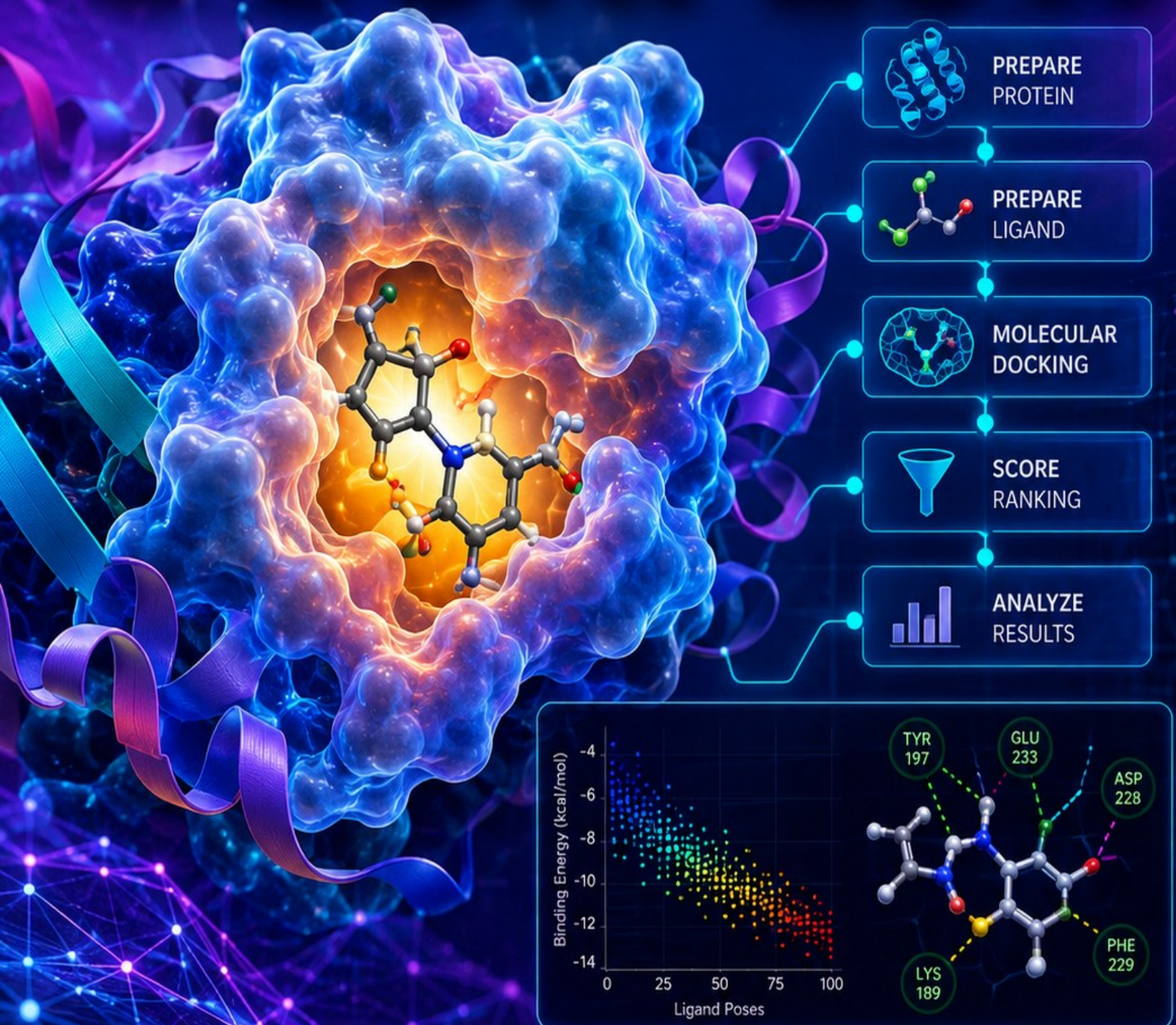


APLIKASI DOCKFLOW

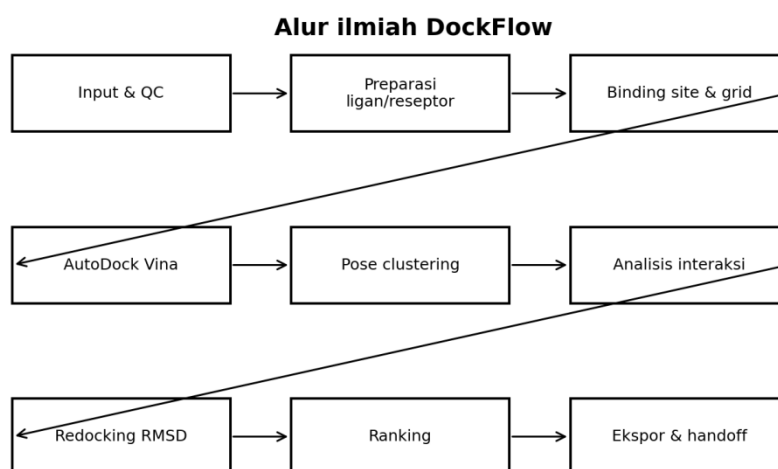


KASMUI

APLIKASI DOCKFLOW

MOLECULAR DOCKING DAN ANALISIS INTERAKSI LIGAN-PROTEIN

*Panduan Konseptual, Praktikum, Validasi, Analisis Hasil, dan Integrasi
Workflow*



Gambar Sampul. Ringkasan workflow ilmiah yang menjadi kerangka Aplikasi DockFlow.

Berbasis DockFlow V1.1.4
2026

CATATAN EDISI

Buku ini disusun berdasarkan source code DockFlow V1.1.4 yang diberikan pengguna sebagai sumber teknis primer. Uraian fungsi, parameter default, arsitektur, alur tutorial, dan batas interpretasi diselaraskan dengan implementasi tersebut. Literatur eksternal digunakan untuk memberi landasan konsep molecular docking, preparasi struktur, AutoDock Vina, RDKit/ETKDG, validasi, dan struktur benchmark.

DockFlow adalah workflow manager untuk molecular docking dan analisis interaksi ligan–protein. Hasil docking diperlakukan sebagai hipotesis pose dan alat prioritasasi dalam protokol yang terdokumentasi. Skor komputasi tidak diposisikan sebagai pengukuran klinis atau afinitas eksperimen absolut. Penggunaan hasil harus mempertimbangkan kualitas struktur, preparasi, sampling, validasi, dan metode lanjutan yang relevan.

Contoh molekul pada buku dipakai sebagai objek pembelajaran komputasi dan benchmark struktur. Buku ini tidak memberikan rekomendasi penggunaan obat atau keputusan terapi.

KATA PENGANTAR

Perkembangan kimia komputasi membuat molecular docking semakin mudah diakses melalui perangkat lunak terbuka, Python, dan basis data struktur tiga dimensi. Kemudahan tersebut merupakan peluang besar bagi pembelajaran dan riset, tetapi sekaligus membawa tantangan metodologis. Pengguna dapat memperoleh skor docking dalam waktu singkat, namun kualitas kesimpulan sangat bergantung pada cara menyiapkan ligan dan reseptor, menetapkan search space, memilih parameter sampling, memvalidasi protokol, membaca pose, dan menyimpan jejak komputasi. Buku APLIKASI DOCKFLOW disusun untuk menjembatani kebutuhan teknis dan kebutuhan penalaran ilmiah tersebut.

DockFlow V1.1.4 tidak dirancang sekadar sebagai antarmuka untuk menjalankan AutoDock Vina. Source code menunjukkan bahwa aplikasi menggabungkan input multi-format, katalog ligan persisten, preparasi struktur dengan RDKit dan Meeko/Open Babel, pembersihan reseptor, beberapa cara penentuan grid, eksekusi Vina melalui Python API atau CLI, replikasi random seed, cache, queue, stop aman, parsing pose, clustering RMSD, analisis interaksi, interaction fingerprint, redocking, ranking multikriteria, SQLite provenance, serta ekspor CSV/XLSX/JSON/PDF/ZIP. Dengan demikian, pembahasan buku mengikuti alur data dari awal sampai handoff ke QM, MD, atau QSAR.

Pendekatan buku ini sengaja menempatkan validasi sebelum interpretasi. Skor docking dibahas sebagai keluaran scoring function yang berguna untuk perbandingan dalam protokol konsisten, bukan sebagai padanan otomatis konstanta disosiasi atau energi bebas eksperimen. Pose yang terlihat meyakinkan secara visual tetap perlu dibaca bersama cluster, residu kontak, keadaan protonasi, ruang pencarian, dan redocking. Sebaliknya, hasil yang tidak memenuhi quality control tidak diselamatkan hanya karena memiliki angka skor yang rendah.

Buku ini juga dirancang sebagai bahan ajar. Setiap bab memuat capaian pembelajaran, uraian konsep, pemetaan ke implementasi DockFlow, tabel keputusan, ilustrasi workflow, latihan, pembahasan, dan soal pengembangan. Model ini diharapkan membantu mahasiswa bergerak dari kemampuan menjalankan aplikasi menuju kemampuan merancang protokol dan bertanggungjawabkan keputusan komputasinya. Penilaian seharusnya tidak berhenti pada "siapa mendapat score paling negatif", tetapi mencakup ketepatan input, konsistensi parameter, kualitas validasi, keterlacakan data, dan kekuatan argumentasi ilmiah.

Studi kasus ER α /3ERT–4-hydroxytamoxifen yang sudah tertanam pada tutorial DockFlow dipakai sebagai contoh integratif. Struktur 3ERT menyediakan receptor dan ligan ko-kristal untuk latihan redocking, sedangkan artikel Muchtaridi dan kolega memberi konteks penelitian turunan chalcone. Buku

menekankan bahwa artikel tersebut menggunakan AutoDock 4.2, sementara DockFlow menggunakan AutoDock Vina; karena itu reproduksi yang bertanggung jawab menilai pose, RMSD, residu interaksi, konsistensi protokol, dan tren, bukan memaksa skor absolut kedua engine menjadi identik.

Penyusunan buku mengikuti format buku profesional pada master prompt yang diberikan pengguna: A4, margin kiri 4 cm serta margin atas, kanan, dan bawah 3 cm; paragraf Times New Roman 12 pt, justify, spasi 1,15 tanpa first-line indent; hierarki Heading 1–3; awal bab pada halaman baru; subbab mengalir tanpa page break manual; caption gambar di bawah dan caption tabel di atas; rumus menggunakan notasi teks/Math; serta daftar isi yang ditautkan ke heading. Glosarium disusun A–Z tanpa menjadikan huruf abjad sebagai heading, sehingga hanya judul GLOSARIUM yang masuk daftar isi.

Keterbukaan metodologis menjadi prinsip akhir buku. Bila pembaca mengubah receptor, protonasi, grid, exhaustiveness, seed, scoring, atau metode preparasi, perubahan tersebut harus diperlakukan sebagai bagian dari desain eksperimen dan dicatat. Dengan kebiasaan ini, DockFlow dapat menjadi sarana pembelajaran yang baik untuk memahami bukan hanya cara "menjalankan docking", tetapi juga cara membangun workflow komputasi yang dapat diperiksa, diulang, dan dikritisi.

Semarang, Agustus 2026

Penulis,



KASMUI

PETUNJUK PENGGUNAAN BUKU

Buku dapat dibaca berurutan dari Bab 1 sampai Bab 10 atau dipakai secara modular dalam praktikum. Bab 1–2 membangun fondasi dan arsitektur; Bab 3–5 membahas preparasi input dan search space; Bab 6–8 membahas eksekusi, analisis, dan validasi; Bab 9 membahas ranking/provenance/handoff; Bab 10 mengintegrasikan semuanya dalam studi kasus dan troubleshooting.

Setiap praktikum sebaiknya dimulai dengan job specification yang dibekukan sebelum run. Mahasiswa kemudian menyimpan input, konfigurasi, log, hasil mentah, hasil QC, dan laporan interpretasi. Bila hasil tidak sesuai ekspektasi, lakukan troubleshooting pada tahap yang paling awal menunjukkan ketidaksesuaian, bukan langsung mengubah banyak parameter sekaligus.

Pada Microsoft Word, Daftar Isi dapat diperbarui setelah perubahan tata letak dengan memilih seluruh dokumen melalui Ctrl+A kemudian menekan F9. Hanya paragraf dengan style Heading 1, Heading 2, dan Heading 3 yang dimaksudkan masuk Daftar Isi; soal, jawaban, caption, tabel, gambar, dan huruf A–Z pada glosarium tidak menggunakan style heading.

DAFTAR ISI

KATA PENGANTAR	4
PETUNJUK PENGGUNAAN BUKU	6
DAFTAR ISI	7
BAB 1. LANDASAN MOLECULAR DOCKING DAN FILOSOFI DOCKFLOW	16
1.1 Capaian Pembelajaran	16
1.2 Molecular docking sebagai masalah pencarian dan penilaian	17
1.2.1 Catatan Quality Control	17
1.3 Hipotesis pose dan batas inferensi	18
1.3.1 Catatan Quality Control	19
1.4 Workflow tervalidasi	20
1.4.1 Catatan Quality Control	21
1.5 Peran ligan dan reseptor	22
1.5.1 Catatan Quality Control	23
1.6 Reprodusibilitas komputasi	24
1.6.1 Catatan Quality Control	24
1.7 Clustering dan konsensus pose	25
1.7.1 Catatan Quality Control	26
1.8 Analisis interaksi sebagai deskripsi geometrik	27
1.8.1 Catatan Quality Control	28
1.9 Handoff ke metode lanjutan	29
1.9.1 Catatan Quality Control	29
1.10 Rumus dan Indikator yang Relevan	31
1.11 Praktikum Terpandu	31
1.11.1 Analisis Praktikum 1	31
1.11.2 Analisis Praktikum 2	32
1.11.3 Analisis Praktikum 3	32
1.12 Troubleshooting dan Kesalahan yang Sering Terjadi	33
1.13 Latihan: 20 Soal	34
1.14 Pembahasan 20 Soal	36
1.15 Soal Pengembangan	42

BAB 2. ARSITEKTUR APLIKASI, INSTALASI, DAN MANAJEMEN WORKSPACE.....	44
2.1 Capaian Pembelajaran	44
2.2 Lapisan arsitektur aplikasi	45
2.2.1 Catatan Quality Control.....	45
2.3 Bootstrap dependensi Python.....	46
2.3.1 Catatan Quality Control.....	47
2.4 Backend docking dan fallback	48
2.4.1 Catatan Quality Control.....	49
2.5 Workspace lintas sistem operasi	50
2.5.1 Catatan Quality Control.....	50
2.6 Port dan peluncuran Streamlit.....	52
2.6.1 Catatan Quality Control.....	52
2.7 SQLite dan audit trail	53
2.7.1 Catatan Quality Control.....	54
2.8 Hash, cache, dan identitas data	55
2.8.1 Catatan Quality Control.....	56
2.9 Monitoring, stop aman, dan riwayat.....	57
2.9.1 Catatan Quality Control.....	57
2.10 Rumus dan Indikator yang Relevan	58
2.11 Praktikum Terpandu	59
2.11.1 Analisis Praktikum 1	59
2.11.2 Analisis Praktikum 2	59
2.11.3 Analisis Praktikum 3	60
2.12 Troubleshooting dan Kesalahan yang Sering Terjadi	60
2.13 Latihan: 20 Soal	62
2.14 Pembahasan 20 Soal	63
2.15 Soal Pengembangan	70
BAB 3. INPUT, QUALITY CONTROL, DAN PREPARASI LIGAN	71
3.1 Capaian Pembelajaran	71
3.2 Representasi SMILES dan InChI.....	72
3.2.1 Catatan Quality Control.....	73

3.3 Input multi-format.....	74
3.3.1 Catatan Quality Control.....	74
3.4 Sanitasi dan identitas molekul	76
3.4.1 Catatan Quality Control.....	76
3.5 Deduplikasi ligan	77
3.5.1 Catatan Quality Control.....	78
3.6 Pemilihan fragmen dan penambahan hidrogen.....	79
3.6.1 Catatan Quality Control.....	80
3.7 Generasi 3D ETKDGv3	81
3.7.1 Catatan Quality Control.....	81
3.8 Optimasi MMFF94 dan UFF	82
3.8.1 Catatan Quality Control.....	83
3.9 Meeko dan Open Babel	84
3.9.1 Catatan Quality Control.....	85
3.10 Rumus dan Indikator yang Relevan	86
3.11 Praktikum Terpandu	86
3.11.1 Analisis Praktikum 1	86
3.11.2 Analisis Praktikum 2	87
3.11.3 Analisis Praktikum 3	87
3.12 Troubleshooting dan Kesalahan yang Sering Terjadi	88
3.13 Latihan: 20 Soal	89
3.14 Pembahasan 20 Soal.....	90
3.15 Soal Pengembangan	97
BAB 4. AKUISISI, PEMBERSIHAN, DAN PREPARASI RESEPTOR.....	98
4.1 Capaian Pembelajaran	98
4.2 Sumber struktur reseptor	99
4.2.1 Catatan Quality Control.....	99
4.3 PDB versus mmCIF	100
4.3.1 Catatan Quality Control.....	101
4.4 Pemilihan chain.....	102
4.4.1 Catatan Quality Control.....	103
4.5 Kebijakan water	104

4.5.1 Catatan Quality Control.....	104
4.6 HETATM, ion, kofaktor, dan ligan ko-kristal.....	105
4.6.1 Catatan Quality Control.....	106
4.7 Penambahan hidrogen dan charge reseptor	107
4.7.1 Catatan Quality Control.....	108
4.8 Reseptor PDBQT siap pakai.....	109
4.8.1 Catatan Quality Control.....	109
4.9 Provenance cleaning.....	110
4.9.1 Catatan Quality Control.....	111
4.10 Rumus dan Indikator yang Relevan	112
4.11 Praktikum Terpandu.....	112
4.11.1 Analisis Praktikum 1	113
4.11.2 Analisis Praktikum 2	113
4.11.3 Analisis Praktikum 3	114
4.12 Troubleshooting dan Kesalahan yang Sering Terjadi	114
4.13 Latihan: 20 Soal	116
4.14 Pembahasan 20 Soal.....	117
4.15 Soal Pengembangan	125
BAB 5. BINDING SITE, GRID, DAN DESAIN SEARCH SPACE	126
5.1 Capaian Pembelajaran	126
5.2 Makna center dan size	127
5.2.1 Catatan Quality Control.....	127
5.3 Grid manual	128
5.3.1 Catatan Quality Control.....	129
5.4 Grid dari ligan referensi.....	130
5.4.1 Catatan Quality Control.....	131
5.5 Grid dari residu situs aktif	132
5.5.1 Catatan Quality Control.....	132
5.6 Padding dan ruang orientasi.....	133
5.6.1 Catatan Quality Control.....	134
5.7 Validasi ukuran dan volume	135
5.7.1 Catatan Quality Control.....	135

5.8 Directed versus blind docking	137
5.8.1 Catatan Quality Control.....	137
5.9 Visualisasi grid.....	138
5.9.1 Catatan Quality Control.....	139
5.10 Rumus dan Indikator yang Relevan	140
5.11 Praktikum Terpandu	140
5.11.1 Analisis Praktikum 1	141
5.11.2 Analisis Praktikum 2	141
5.11.3 Analisis Praktikum 3	141
5.12 Troubleshooting dan Kesalahan yang Sering Terjadi	142
5.13 Latihan: 20 Soal	143
5.14 Pembahasan 20 Soal.....	144
5.15 Soal Pengembangan	152
BAB 6. AUTODOCK VINA, REPLIKASI, BATCH, DAN POSE CLUSTERING	153
6.1 Capaian Pembelajaran	153
6.2 Scoring function Vina	154
6.2.1 Catatan Quality Control.....	154
6.3 Exhaustiveness.....	155
6.3.1 Catatan Quality Control.....	156
6.4 Number of modes dan energy range	157
6.4.1 Catatan Quality Control.....	158
6.5 Random seed dan replikasi.....	159
6.5.1 Catatan Quality Control.....	159
6.6 Python API versus CLI.....	161
6.6.1 Catatan Quality Control.....	161
6.7 Parsing pose PDBQT.....	162
6.7.1 Catatan Quality Control.....	163
6.8 RMSD antar-pose.....	164
6.8.1 Catatan Quality Control.....	165
6.9 Pose clustering dan consensus	166
6.9.1 Catatan Quality Control.....	166

6.10 Rumus dan Indikator yang Relevan	167
6.11 Praktikum Terpandu	168
6.11.1 Analisis Praktikum 1	168
6.11.2 Analisis Praktikum 2	168
6.11.3 Analisis Praktikum 3	169
6.12 Troubleshooting dan Kesalahan yang Sering Terjadi	169
6.13 Latihan: 20 Soal	171
6.14 Pembahasan 20 Soal	172
6.15 Soal Pengembangan	179
BAB 7. ANALISIS INTERAKSI LIGAN-PROTEIN DAN INTERACTION FINGERPRINT	180
7.1 Capaian Pembelajaran	180
7.2 Kontak polar dan kandidat H-bond	181
7.2.1 Catatan Quality Control	181
7.3 Kontak hidrofobik	182
7.3.1 Catatan Quality Control	183
7.4 Interaksi aromatik π - π	184
7.4.1 Catatan Quality Control	185
7.5 Kation- π	186
7.5.1 Catatan Quality Control	186
7.6 Salt bridge	187
7.6.1 Catatan Quality Control	188
7.7 Interaction fingerprint	189
7.7.1 Catatan Quality Control	190
7.8 Frekuensi residu	191
7.8.1 Catatan Quality Control	191
7.9 Key residues	192
7.9.1 Catatan Quality Control	193
7.10 Rumus dan Indikator yang Relevan	194
7.11 Praktikum Terpandu	194
7.11.1 Analisis Praktikum 1	195
7.11.2 Analisis Praktikum 2	195

7.11.3 Analisis Praktikum 3	195
7.12 Troubleshooting dan Kesalahan yang Sering Terjadi	196
7.13 Latihan: 20 Soal	197
7.14 Pembahasan 20 Soal	199
7.15 Soal Pengembangan	205
BAB 8. VALIDASI REDOCKING, RMSD, DAN REPRODUKSIBILITAS PROTOKOL.....	206
8.1 Capaian Pembelajaran	206
8.2 Tujuan redocking	207
8.2.1 Catatan Quality Control	207
8.3 Pose referensi	208
8.3.1 Catatan Quality Control	209
8.4 Threshold RMSD	210
8.4.1 Catatan Quality Control	211
8.5 Direct heavy-atom RMSD	212
8.5.1 Catatan Quality Control	212
8.6 Sumber kegagalan redocking	213
8.6.1 Catatan Quality Control	214
8.7 Redocking dan grid referensi	215
8.7.1 Catatan Quality Control	216
8.8 Replikasi validasi	217
8.8.1 Catatan Quality Control	217
8.9 Dokumentasi validasi	218
8.9.1 Catatan Quality Control	219
8.10 Rumus dan Indikator yang Relevan	220
8.11 Praktikum Terpandu	220
8.11.1 Analisis Praktikum 1	221
8.11.2 Analisis Praktikum 2	221
8.11.3 Analisis Praktikum 3	222
8.12 Troubleshooting dan Kesalahan yang Sering Terjadi	222
8.13 Latihan: 20 Soal	224
8.14 Pembahasan 20 Soal	225

8.15 Soal Pengembangan	232
BAB 9. RANKING MULTIKRITERIA, PROVENANCE, EKSPOR, DAN INTEGRASI.....	233
9.1 Capaian Pembelajaran	233
9.2 Ligand efficiency	234
9.2.1 Catatan Quality Control	235
9.3 Pose consensus.....	236
9.3.1 Catatan Quality Control	236
9.4 Bobot ranking.....	237
9.4.1 Catatan Quality Control	238
9.5 Normalisasi fitur	239
9.5.1 Catatan Quality Control	240
9.6 QC penalty	241
9.6.1 Catatan Quality Control	241
9.7 Ekspor multi-format	242
9.7.1 Catatan Quality Control	243
9.8 SQLite provenance.....	244
9.8.1 Catatan Quality Control	245
9.9 Integration manifest.....	246
9.9.1 Catatan Quality Control	246
9.10 Rumus dan Indikator yang Relevan	247
9.11 Praktikum Terpandu	248
9.11.1 Analisis Praktikum 1	248
9.11.2 Analisis Praktikum 2	249
9.11.3 Analisis Praktikum 3	249
9.12 Troubleshooting dan Kesalahan yang Sering Terjadi	249
9.13 Latihan: 20 Soal	251
9.14 Pembahasan 20 Soal	252
9.15 Soal Pengembangan	259
BAB 10. STUDI KASUS ERA/3ERT, PRAKTIK PEMBELAJARAN, DAN TROUBLESHOOTING.....	260
10.1 Capaian Pembelajaran	260
10.2 Struktur PDB 3ERT	261

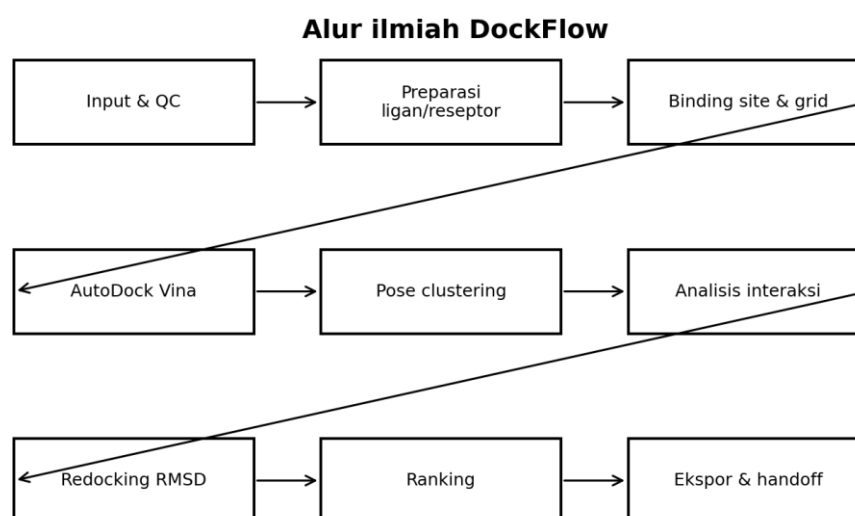
10.2.1 Catatan Quality Control	261
10.3 4-hydroxytamoxifen sebagai ligan referensi.....	262
10.3.1 Catatan Quality Control	263
10.4 Artikel chalcone sebagai konteks reproduksi	264
10.4.1 Catatan Quality Control	265
10.5 Langkah redocking terarah	266
10.5.1 Catatan Quality Control	266
10.6 Perbandingan residu kunci.....	268
10.6.1 Catatan Quality Control	268
10.7 Perbedaan engine dan parameter.....	269
10.7.1 Catatan Quality Control	270
10.8 Troubleshooting berbasis tahap	271
10.8.1 Catatan Quality Control	272
10.9 Desain praktikum dan laporan ilmiah	273
10.9.1 Catatan Quality Control	273
10.10 Rumus dan Indikator yang Relevan	274
10.11 Praktikum Terpandu	275
10.11.1 Analisis Praktikum 1.....	275
10.11.2 Analisis Praktikum 2.....	275
10.11.3 Analisis Praktikum 3.....	276
10.12 Troubleshooting dan Kesalahan yang Sering Terjadi	276
10.13 Latihan: 20 Soal	278
10.14 Pembahasan 20 Soal	279
10.15 Soal Pengembangan	286
GLOSARIUM	287
DAFTAR PUSTAKA	292
Catatan Pemanfaatan Literatur	295

BAB 1. LANDASAN MOLECULAR DOCKING DAN FILOSOFI DOCKFLOW

Bab ini menempatkan docking sebagai metode komputasi untuk menghasilkan hipotesis pose dan memprioritaskan ligan dalam protokol yang terdefinisi, bukan sebagai pengganti pengukuran eksperimen. Kerangka ini penting karena seluruh fitur DockFlow dibangun untuk menjaga jejak parameter, validasi, dan interpretasi hasil.

1.1 Capaian Pembelajaran

- Menjelaskan konsep utama Bab 1 dengan istilah yang tepat dan membedakan data komputasi dari inferensi.
- Menghubungkan konsep dengan parameter, file, dan artefak yang benar-benar dihasilkan DockFlow.
- Menyusun quality-control dan kriteria keputusan sebelum menjalankan batch.
- Menganalisis kegagalan secara bertahap dan mendokumentasikan perubahan protokol.
- Menyusun laporan yang dapat diperiksa dan direproduksi.



Gambar 1.1. Skema utama pembahasan Bab 1: Landasan Molecular Docking Dan Filosofi Dockflow.

Tabel 1.1. Perbedaan keluaran dan batas interpretasinya

Keluaran	Makna yang wajar	Batas interpretasi
Vina score	Skor model untuk pose/kandidat dalam protokol sama	Bukan Kd/Ki atau ΔG eksperimen absolut
Pose cluster	Konsistensi geometri hasil sampling	Bukan satu struktur biologis pasti

Keluaran	Makna yang wajar	Batas interpretasi
Redocking RMSD	Kemampuan mereproduksi pose referensi	Bukan validasi afinitas
Interaction fingerprint	Representasi kontak untuk perbandingan	Bukan mekanisme biologis final
Composite rank	Prioritisasi berbobot	Bukan urutan potensi universal

1.2 Molecular docking sebagai masalah pencarian dan penilaian

Docking memadukan pencarian ruang konformasi dengan scoring function. Kualitas pose dan kualitas ranking adalah dua tujuan yang berhubungan tetapi tidak identik. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow mengorganisasi pencarian melalui AutoDock Vina, replikasi seed, parsing pose, clustering, dan ranking multikriteria. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menyamakan skor terendah dengan afinitas eksperimen; mengabaikan protonasi, fleksibilitas, dan ruang pencarian. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

1.2.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait molecular docking sebagai masalah pencarian dan penilaian.

- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menyamakan skor terendah dengan afinitas eksperimen; mengabaikan protonasi, fleksibilitas, dan ruang pencarian.
- Keputusan yang dianjurkan: Pisahkan pertanyaan pose prediction, virtual screening, dan interpretasi biologis.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Pisahkan pertanyaan pose prediction, virtual screening, dan interpretasi biologis. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Molecular docking sebagai masalah pencarian dan penilaian” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “molecular docking sebagai masalah pencarian dan penilaian”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

1.3 Hipotesis pose dan batas inferensi

Pose docking adalah model yang harus diuji konsistensinya terhadap struktur referensi, residu kunci, dan konteks kimia-fisika. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada

protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Jika dibaca sebagai bagian dari workflow, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Aplikasi menampilkan pose, cluster, interaksi, dan redocking agar pengguna tidak berhenti pada satu angka skor. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengklaim mekanisme atau potensi biologis hanya dari skor komputasi. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

1.3.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait hipotesis pose dan batas inferensi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengklaim mekanisme atau potensi biologis hanya dari skor komputasi.
- Keputusan yang dianjurkan: Gunakan bahasa "mendukung hipotesis" dan dokumentasikan keterbatasan.

Dari sisi quality control, keputusan yang disarankan adalah: Gunakan bahasa "mendukung hipotesis" dan dokumentasikan keterbatasan. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina

menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Hipotesis pose dan batas inferensi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “hipotesis pose dan batas inferensi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 1.2. Skenario keputusan untuk hipotesis pose dan batas inferensi

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Mengklaim mekanisme atau potensi biologis hanya dari skor komputasi.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Gunakan bahasa "mendukung hipotesis" dan dokumentasikan keterbatasan.

1.4 Workflow tervalidasi

Workflow yang baik menyatukan input, parameter, output mentah, QC, dan jejak reproduksibilitas. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Untuk keperluan reproduksibilitas, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menulis job specification, environment snapshot, SQLite provenance, raw output, dan artefak ekspor. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengubah parameter antar-ligan tanpa catatan sehingga ranking menjadi tidak sebanding. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

1.4.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait workflow tervalidasi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengubah parameter antar-ligan tanpa catatan sehingga ranking menjadi tidak sebanding.
- Keputusan yang dianjurkan: Tetapkan protokol sebelum batch dan ubah hanya melalui eksperimen terkontrol.

Dalam konteks pembelajaran, keputusan yang disarankan adalah: Tetapkan protokol sebelum batch dan ubah hanya melalui eksperimen terkontrol. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Workflow tervalidasi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “workflow tervalidasi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

1.5 Peran ligan dan reseptor

Kualitas struktur input menentukan validitas pencarian docking. Struktur 2D, 3D, protonasi, chain, water, kofaktor, dan heteroatom memiliki konsekuensi. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sisi quality control, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menyediakan QC ligan dan pembersihan reseptor dengan kebijakan yang dicatat. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menghapus seluruh hetatm secara mekanis atau memakai struktur ligan tanpa pemeriksaan valensi. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

1.5.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait peran ligan dan reseptor.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menghapus seluruh HETATM secara mekanis atau memakai struktur ligan tanpa pemeriksaan valensi.
- Keputusan yang dianjurkan: Tentukan objek biologis dan alasan setiap langkah preparasi.

Dari sudut pandang metodologi, keputusan yang disarankan adalah: Tentukan objek biologis dan alasan setiap langkah preparasi. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Peran ligan dan reseptor” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “peran ligan dan reseptor”: apakah struktur dan state identik, apakah reseptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 1.3. Skenario keputusan untuk peran ligan dan reseptor

Elemen	Uraian
--------	--------

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menghapus seluruh HETATM secara mekanis atau memakai struktur ligan tanpa pemeriksaan valensi.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Tentukan objek biologis dan alasan setiap langkah preparasi.

1.6 Reprodusibilitas komputasi

Reprodusibilitas memerlukan seed, versi engine, parameter sampling, file input, dan output mentah. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam konteks pembelajaran, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Random seed, exhaustiveness, jumlah mode, backend, dan snapshot lingkungan dicatat. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah hanya menyimpan tabel akhir tanpa konfigurasi dan file sumber. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

1.6.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait reprodusibilitas komputasi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.

- Tandai secara eksplisit risiko: Hanya menyimpan tabel akhir tanpa konfigurasi dan file sumber.
- Keputusan yang dianjurkan: Simpan paket proyek lengkap bersama manifest dan log.

Pada tahap ini, keputusan yang disarankan adalah: Simpan paket proyek lengkap bersama manifest dan log. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Reproduksibilitas komputasi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “reproduksibilitas komputasi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

1.7 Clustering dan konsensus pose

Banyak pose dapat merepresentasikan mode yang serupa. Clustering membantu menilai apakah pencarian berulang konvergen ke wilayah geometri yang konsisten. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa

pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sudut pandang metodologi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow mengelompokkan pose dengan RMSD heavy-atom dan memilih representative pose. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap jumlah pose sebagai bukti statistik tanpa mempertimbangkan cara sampling. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

1.7.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait clustering dan konsensus pose.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap jumlah pose sebagai bukti statistik tanpa mempertimbangkan cara sampling.
- Keputusan yang dianjurkan: Gunakan cluster sebagai indikator konsistensi internal, bukan probabilitas biologis absolut.

Dalam praktik komputasi, keputusan yang disarankan adalah: Gunakan cluster sebagai indikator konsistensi internal, bukan probabilitas biologis absolut. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow

menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Clustering dan konsensus pose” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “clustering dan konsensus pose”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 1.4. Skenario keputusan untuk clustering dan konsensus pose

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menganggap jumlah pose sebagai bukti statistik tanpa mempertimbangkan cara sampling.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Gunakan cluster sebagai indikator konsistensi internal, bukan probabilitas biologis absolut.

1.8 Analisis interaksi sebagai deskripsi geometrik

Kontak ligan-protein memperkaya interpretasi pose, tetapi beberapa klasifikasi hanya screen geometrik. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Pada tahap ini, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow melabeli H-bond sebagai candidate karena sudut

donor-H-akseptor tidak selalu tersedia dari PDBQT. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengubah kontak jarak menjadi klaim ikatan pasti. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

1.8.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait analisis interaksi sebagai deskripsi geometrik.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengubah kontak jarak menjadi klaim ikatan pasti.
- Keputusan yang dianjurkan: Bedakan kandidat geometrik, interaksi terverifikasi, dan interpretasi mekanistik.

Jika dibaca sebagai bagian dari workflow, keputusan yang disarankan adalah: Bedakan kandidat geometrik, interaksi terverifikasi, dan interpretasi mekanistik. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Analisis interaksi sebagai deskripsi geometrik” dalam tiga lapis bukti. Lapis pertama adalah

konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “analisis interaksi sebagai deskripsi geometrik”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

1.9 Handoff ke metode lanjutan

Docking paling kuat ketika menjadi tahap awal yang mengarahkan analisis lanjutan sesuai pertanyaan riset. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: `integration_manifest.json` menyiapkan pose dan fitur untuk QM, MD, atau QSAR. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah meneruskan semua kandidat tanpa prioritas atau tanpa menyimpan protokol asal. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

1.9.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait handoff ke metode lanjutan.

- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Meneruskan semua kandidat tanpa prioritas atau tanpa menyimpan protokol asal.
- Keputusan yang dianjurkan: Gunakan handoff yang terlacak dan jelaskan alasan pemilihan kandidat.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Gunakan handoff yang terlacak dan jelaskan alasan pemilihan kandidat. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Handoff ke metode lanjutan” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “handoff ke metode lanjutan”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 1.5. Skenario keputusan untuk handoff ke metode lanjutan

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.

Elemen	Uraian
Risiko utama	Meneruskan semua kandidat tanpa prioritas atau tanpa menyimpan protokol asal.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Gunakan handoff yang terlacak dan jelaskan alasan pemilihan kandidat.

1.10 Rumus dan Indikator yang Relevan

Keputusan ilmiah = bukti input + QC + parameter + hasil + validasi + keterbatasan

Formula konseptual tersebut mengingatkan bahwa satu indikator tidak cukup untuk menopang kesimpulan workflow. Setiap bagian menyediakan jenis bukti yang berbeda dan saling melengkapi.

1.11 Praktikum Terpandu

1. Buat folder proyek baru dan tulis tujuan eksperimen dalam satu paragraf sebelum menjalankan aplikasi.
2. Catat sumber receptor, ligan, dan semua parameter yang relevan dengan bab ini.
3. Jalankan QC input; bila ada warning, tuliskan keputusan menerima atau memperbaiki beserta alasannya.
4. Jalankan satu baseline terlebih dahulu. Simpan konfigurasi, log, raw output, dan ringkasan hasil.
5. Lakukan satu perubahan terkontrol pada parameter yang sedang dipelajari, lalu bandingkan keluaran dengan baseline.
6. Tulis kesimpulan yang memisahkan observasi komputasi, interpretasi, dan keterbatasan.

1.11.1 Analisis Praktikum 1

Analisis praktikum 1 pada Bab 1 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

1.11.2 Analisis Praktikum 2

Analisis praktikum 2 pada Bab 1 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

1.11.3 Analisis Praktikum 3

Analisis praktikum 3 pada Bab 1 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

1.12 Troubleshooting dan Kesalahan yang Sering Terjadi

Tabel 1.6. Troubleshooting ringkas Bab 1

Area	Gejala/Risiko	Tindakan
Molecular docking sebagai masalah pencarian dan penilaian	Menyamakan skor terendah dengan afinitas eksperimen; mengabaikan protonasi, fleksibilitas, dan ruang pencarian.	Pisahkan pertanyaan pose prediction, virtual screening, dan interpretasi biologis.
Hipotesis pose dan batas inferensi	Mengklaim mekanisme atau potensi biologis hanya dari skor komputasi.	Gunakan bahasa "mendukung hipotesis" dan dokumentasikan keterbatasan.
Workflow tervalidasi	Mengubah parameter antar-ligan tanpa catatan sehingga ranking menjadi tidak sebanding.	Tetapkan protokol sebelum batch dan ubah hanya melalui eksperimen terkontrol.
Peran ligan dan reseptor	Menghapus seluruh HETATM secara mekanis atau memakai struktur ligan tanpa pemeriksaan valensi.	Tentukan objek biologis dan alasan setiap langkah preparasi.
Reproduksibilitas komputasi	Hanya menyimpan tabel akhir tanpa konfigurasi dan file sumber.	Simpan paket proyek lengkap bersama manifest dan log.
Clustering dan konsensus pose	Menganggap jumlah pose sebagai bukti statistik tanpa mempertimbangkan cara sampling.	Gunakan cluster sebagai indikator konsistensi internal, bukan probabilitas biologis absolut.

Jika masalah berada pada area molecular docking sebagai masalah pencarian dan penilaian, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menyamakan skor terendah dengan afinitas eksperimen; mengabaikan protonasi, fleksibilitas, dan ruang pencarian. Tindakan awal yang lebih kuat adalah Pisahkan pertanyaan pose prediction, virtual screening, dan interpretasi biologis. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area hipotesis pose dan batas inferensi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengklaim mekanisme atau potensi biologis hanya dari skor komputasi. Tindakan awal yang lebih kuat adalah Gunakan bahasa "mendukung hipotesis" dan dokumentasikan keterbatasan. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area workflow tervalidasi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengubah parameter antar-

ligan tanpa catatan sehingga ranking menjadi tidak sebanding. Tindakan awal yang lebih kuat adalah Tetapkan protokol sebelum batch dan ubah hanya melalui eksperimen terkontrol. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area peran ligan dan reseptor, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menghapus seluruh hetatm secara mekanis atau memakai struktur ligan tanpa pemeriksaan valensi. Tindakan awal yang lebih kuat adalah Tentukan objek biologis dan alasan setiap langkah preparasi. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area reproduksibilitas komputasi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah hanya menyimpan tabel akhir tanpa konfigurasi dan file sumber. Tindakan awal yang lebih kuat adalah Simpan paket proyek lengkap bersama manifest dan log. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area clustering dan konsensus pose, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap jumlah pose sebagai bukti statistik tanpa mempertimbangkan cara sampling. Tindakan awal yang lebih kuat adalah Gunakan cluster sebagai indikator konsistensi internal, bukan probabilitas biologis absolut. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

1.13 Latihan: 20 Soal

1. Jelaskan mengapa molecular docking sebagai masalah pencarian dan penilaian perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
2. Jelaskan mengapa hipotesis pose dan batas inferensi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
3. Jelaskan mengapa workflow tervalidasi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.

4. Jelaskan mengapa peran ligan dan reseptor perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
5. Jelaskan mengapa reproduksibilitas komputasi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
6. Jelaskan mengapa clustering dan konsensus pose perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
7. Jelaskan mengapa analisis interaksi sebagai deskripsi geometrik perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
8. Jelaskan mengapa handoff ke metode lanjutan perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
9. Susun checklist minimum sebelum menjalankan batch pada Bab 1.
10. Sebuah hasil terlihat sangat baik pada satu indikator, tetapi gagal pada indikator QC lain. Bagaimana seharusnya hasil tersebut diperlakukan?
11. Mengapa perubahan satu parameter sebaiknya diuji secara terkontrol?
12. Jelaskan perbedaan antara data mentah, data turunan, dan interpretasi dalam workflow DockFlow.
13. Apa fungsi provenance ketika dua run menghasilkan hasil berbeda?
14. Kritik pernyataan: “Ligan dengan docking score paling rendah pasti merupakan kandidat terbaik.”
15. Bagaimana Anda menulis bagian metode agar run dapat direproduksi?
16. Mengapa visualisasi tidak boleh menggantikan pemeriksaan numerik?
17. Rancang satu eksperimen sensitivitas yang relevan dengan bab ini.
18. Apa yang harus dilakukan bila hasil tidak dapat direproduksi oleh kelompok lain?
19. Mengapa buku ini menekankan redocking sebelum screening besar?
20. Buat format ringkas laporan hasil yang baik.

1.14 Pembahasan 20 Soal

Pembahasan 1. Docking memadukan pencarian ruang konformasi dengan scoring function. Kualitas pose dan kualitas ranking adalah dua tujuan yang berhubungan tetapi tidak identik. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow mengorganisasi pencarian melalui AutoDock Vina, replikasi seed, parsing pose, clustering, dan ranking multikriteria. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menyamakan skor terendah dengan afinitas eksperimen; mengabaikan protonasi, fleksibilitas, dan ruang pencarian. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Pisahkan pertanyaan pose prediction, virtual screening, dan interpretasi biologis.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 2. Pose docking adalah model yang harus diuji konsistensinya terhadap struktur referensi, residu kunci, dan konteks kimia-fisika. Pada DockFlow, aspek ini diwujudkan melalui: Aplikasi menampilkan pose, cluster, interaksi, dan redocking agar pengguna tidak berhenti pada satu angka skor. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengklaim mekanisme atau potensi biologis hanya dari skor komputasi. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan bahasa "mendukung hipotesis" dan dokumentasikan keterbatasan.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 3. Workflow yang baik menyatukan input, parameter, output mentah, QC, dan jejak reproduksibilitas. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menulis job specification, environment snapshot, SQLite provenance, raw output, dan artefak ekspor. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengubah parameter antar-ligan tanpa catatan

sehingga ranking menjadi tidak sebanding. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Tetapkan protokol sebelum batch dan ubah hanya melalui eksperimen terkontrol.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 4. Kualitas struktur input menentukan validitas pencarian docking. Struktur 2D, 3D, protonasi, chain, water, kofaktor, dan heteroatom memiliki konsekuensi. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menyediakan QC ligan dan pembersihan reseptor dengan kebijakan yang dicatat. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menghapus seluruh hetatm secara mekanis atau memakai struktur ligan tanpa pemeriksaan valensi. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Tentukan objek biologis dan alasan setiap langkah preparasi.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 5. Reprodusibilitas memerlukan seed, versi engine, parameter sampling, file input, dan output mentah. Pada DockFlow, aspek ini diwujudkan melalui: Random seed, exhaustiveness, jumlah mode, backend, dan snapshot lingkungan dicatat. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah hanya menyimpan tabel akhir tanpa konfigurasi dan file sumber. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Simpan paket proyek lengkap bersama manifest dan log.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika

terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 6. Banyak pose dapat merepresentasikan mode yang serupa. Clustering membantu menilai apakah pencarian berulang konvergen ke wilayah geometri yang konsisten. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow mengelompokkan pose dengan RMSD heavy-atom dan memilih representative pose. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap jumlah pose sebagai bukti statistik tanpa mempertimbangkan cara sampling. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan cluster sebagai indikator konsistensi internal, bukan probabilitas biologis absolut.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 7. Kontak ligan-protein memperkaya interpretasi pose, tetapi beberapa klasifikasi hanya screen geometrik. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow melabeli H-bond sebagai candidate karena sudut donor-H-akseptor tidak selalu tersedia dari PDBQT. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengubah kontak jarak menjadi klaim ikatan pasti. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Bedakan kandidat geometrik, interaksi terverifikasi, dan interpretasi mekanistik.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 8. Docking paling kuat ketika menjadi tahap awal yang mengarahkan analisis lanjutan sesuai pertanyaan riset. Pada DockFlow, aspek ini diwujudkan melalui: integration_manifest.json menyiapkan pose dan fitur untuk QM, MD, atau QSAR. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari

adalah meneruskan semua kandidat tanpa prioritas atau tanpa menyimpan protokol asal. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan handoff yang terlacak dan jelaskan alasan pemilihan kandidat.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 9. Checklist minimum harus memeriksa identitas input, kelengkapan struktur, parameter yang relevan dengan bab ini, ketersediaan backend, serta lokasi artefak yang akan disimpan. Tambahkan kriteria gagal/lanjut yang jelas. Prinsipnya: jangan memulai batch bila masalah dasar sudah terlihat pada QC. Setelah run, simpan konfigurasi, log, dan keluaran mentah agar keputusan dapat diaudit.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 10. Indikator yang baik tidak menghapus kegagalan QC. Hasil perlu ditandai bermasalah, penyebabnya ditelusuri pada tahap input/preparasi/eksekusi, dan bila perlu run diulang dengan protokol yang diperbaiki. Dalam DockFlow, ranking bukan pengganti QC; hasil invalid bahkan mendapat penalty pada composite score. Laporan harus menyebutkan kegagalan dan tidak menjadikan satu angka sebagai pembenaran.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 11. Perubahan satu faktor memudahkan atribusi sebab-akibat. Bila beberapa parameter diubah sekaligus, pengguna tidak dapat mengetahui faktor mana yang menyebabkan perubahan pose, score, atau interaksi. Praktik yang baik adalah membuat baseline, mengubah satu parameter, menjaga input lain identik,

membandingkan output, lalu mencatat keputusan. Pendekatan ini juga mempermudah reproduksi oleh kelompok lain.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 12. Data mentah meliputi struktur sumber dan output engine; data turunan meliputi pose terpilih, cluster, fingerprint, frekuensi residu, atau ranking; interpretasi adalah argumen ilmiah yang dibangun dari data tersebut. Ketiganya harus dipisahkan agar pembaca dapat menilai apakah transformasi data masuk akal dan apakah kesimpulan lebih kuat daripada bukti yang tersedia.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 13. Provenance menyediakan rekaman input, versi, seed, parameter, backend, waktu, dan artefak yang memungkinkan perbedaan dilacak. Tanpa provenance, perbedaan hasil mudah dianggap sebagai perilaku molekul padahal mungkin berasal dari perubahan preparasi atau konfigurasi. Dengan provenance, pengguna dapat menyusun perbandingan terkontrol dan mengulang kondisi secara lebih tepat.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 14. Pernyataan itu terlalu kuat. Docking score adalah salah satu keluaran model; kualitas pose, cluster, interaksi, redocking, protonasi, search space, dan sampling juga perlu diperiksa. DockFlow sengaja menyediakan composite ranking dan guardrail agar score tidak berdiri sendiri. Bahkan composite score tetap hanya alat prioritisasi internal, bukan bukti aktivitas atau afinitas eksperimen.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file

atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 15. Tuliskan sumber receptor dan ligan, metode preparasi, kebijakan water/hetero/chain, center dan size grid, backend Vina, scoring, exhaustiveness, num modes, energy range, seed, jumlah replikasi, threshold clustering, cutoff interaksi, metode redocking, versi perangkat lunak, serta cara ekspor/analisis. Sertakan file konfigurasi atau manifest bila tersedia.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 16. Visualisasi sangat baik untuk mendeteksi masalah besar seperti grid di luar pocket atau pose terbalik, tetapi tampilan dapat menyesatkan karena perspektif, skala, dan seleksi objek. Pemeriksaan numerik—koordinat, ukuran box, distance, RMSD, score, dan log—memberikan bukti yang lebih eksplisit. Keduanya harus dipakai secara komplementer.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 17. Mulailah dengan baseline yang telah lolos QC. Pilih satu parameter yang benar-benar terkait dengan bab, misalnya padding grid, exhaustiveness, threshold clustering, atau kebijakan preparasi. Jalankan minimal dua atau tiga tingkat parameter dengan input dan seed/replikasi yang terkontrol. Bandingkan pose, cluster, score, interaksi, dan validasi; jangan hanya membandingkan ranking akhir.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 18. Bandingkan checksum/hash input, versi engine, metode preparasi, konfigurasi lengkap, seed, receptor PDBQT, grid, dan output mentah. Pastikan tidak ada langkah manual yang tidak tercatat. Bila perbedaan tetap ada, laporkan lingkungan komputasi dan backend aktual. Reprodusibilitas adalah diagnosis berbasis bukti, bukan sekadar mencoba ulang perintah yang sama.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 19. Jika ligan referensi tersedia, redocking memberi uji awal apakah kombinasi receptor preparation, grid, sampling, dan pose mapping mampu mereproduksi struktur yang diketahui. Kegagalan pada tahap ini memberi sinyal untuk memperbaiki protokol sebelum biaya komputasi dan interpretasi diperluas ke banyak ligan.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 20. Laporan minimal memuat tujuan, input dan sumber struktur, parameter, status backend, QC, hasil redocking bila ada, tabel pose/cluster, analisis interaksi, ranking, keterbatasan, serta lampiran provenance dan raw output. Nyatakan dengan jelas mana data komputasi, mana interpretasi, dan mana fakta dari sumber eksperimen atau literatur.

Penerapan pada Bab 1: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

1.15 Soal Pengembangan

1. Rancang mini-proyek yang menguji satu keputusan metodologis utama Bab 1 dengan baseline dan minimal dua kondisi uji.
2. Susun rubrik penilaian yang memberi bobot pada QC, reprodusibilitas, validasi, interpretasi, dan dokumentasi.

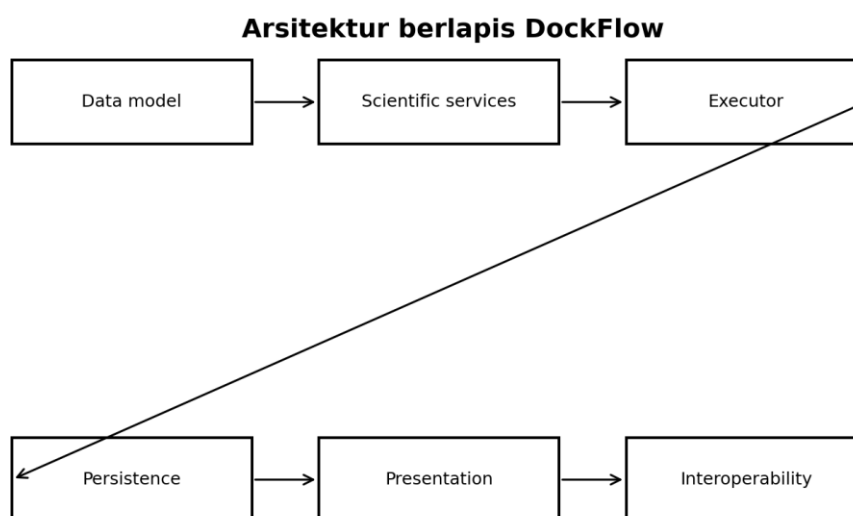
3. Tulis satu contoh klaim yang terlalu kuat dari hasil docking, lalu perbaiki menjadi klaim yang sesuai kekuatan bukti.
4. Buat diagram data lineage dari input sampai satu keluaran akhir dan tunjukkan di mana provenance harus disimpan.
5. Susun daftar informasi yang harus diberikan kepada peneliti lain agar ia dapat mengulang run tanpa bertanya kembali.

BAB 2. ARSITEKTUR APLIKASI, INSTALASI, DAN MANAJEMEN WORKSPACE

DockFlow V1.1.4 adalah aplikasi satu-file berbasis Streamlit dengan dependensi ilmiah dan lapisan persistence. Bab ini membahas arsitektur data, dependensi, workspace, port, backend, dan desain fail-safe yang menentukan keterandalan aplikasi.

2.1 Capaian Pembelajaran

- Menjelaskan konsep utama Bab 2 dengan istilah yang tepat dan membedakan data komputasi dari inferensi.
- Menghubungkan konsep dengan parameter, file, dan artefak yang benar-benar dihasilkan DockFlow.
- Menyusun quality-control dan kriteria keputusan sebelum menjalankan batch.
- Menganalisis kegagalan secara bertahap dan mendokumentasikan perubahan protokol.
- Menyusun laporan yang dapat diperiksa dan direproduksi.



Gambar 2.1. Skema utama pembahasan Bab 2: Arsitektur Aplikasi, Instalasi, Dan Manajemen Workspace.

Tabel 2.1. Komponen teknologi DockFlow

Komponen	Fungsi
Streamlit	Antarmuka aplikasi web lokal
RDKit	Representasi molekul, deskriptor, 3D
Meeko	Preparasi ligan PDBQT

Komponen	Fungsi
Open Babel	Fallback konversi dan preparasi receptor
AutoDock Vina	Engine docking
SQLite	Persistence dan audit
py3Dmol/Plotly	Visualisasi 3D dan grid
pandas/XlsxWriter/ReportLab	Ekspor hasil

2.2 Lapisan arsitektur aplikasi

Pemisahan data model, scientific service, executor, persistence, presentation, dan interoperability membuat workflow lebih mudah diaudit. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menyusun receptor, ligand, grid, job, pose, interaction, validation dalam lapisan yang saling terhubung. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mencampur logika ui dengan hasil ilmiah tanpa state dan provenance. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

2.2.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait lapisan arsitektur aplikasi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.

- Tandai secara eksplisit risiko: Mencampur logika UI dengan hasil ilmiah tanpa state dan provenance.
- Keputusan yang dianjurkan: Baca keluaran berdasarkan lapisan: input, eksekusi, hasil, dan handoff.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Baca keluaran berdasarkan lapisan: input, eksekusi, hasil, dan handoff. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Lapisan arsitektur aplikasi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “lapisan arsitektur aplikasi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

2.3 Bootstrap dependensi Python

Aplikasi memeriksa paket inti dan dapat memasang paket yang hilang dari pemetaan internal. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai

angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Jika dibaca sebagai bagian dari workflow, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Streamlit, NumPy, pandas, RDKit, Plotly, psutil, openpyxl, XlsxWriter, ReportLab, py3Dmol adalah stack inti; vina dan meeko opsional agar GUI tetap terbuka. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap auto-install menjamin backend sistem seperti open babel tersedia. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

2.3.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait bootstrap dependensi python.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap auto-install menjamin backend sistem seperti Open Babel tersedia.
- Keputusan yang dianjurkan: Bedakan paket Python, executable sistem, dan fallback.

Dari sisi quality control, keputusan yang disarankan adalah: Bedakan paket Python, executable sistem, dan fallback. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow

menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Bootstrap dependensi Python” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “bootstrap dependensi python”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 2.2. Skenario keputusan untuk bootstrap dependensi python

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menganggap auto-install menjamin backend sistem seperti Open Babel tersedia.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Bedakan paket Python, executable sistem, dan fallback.

2.4 Backend docking dan fallback

Reliabilitas meningkat bila workflow dapat memilih Python API atau CLI secara eksplisit. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Untuk keperluan reproduksibilitas, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Mode Auto mencoba Vina Python API lalu CLI; error masing-masing dikumpulkan untuk diagnosis. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan

desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah fallback diam-diam tanpa log dapat membuat protokol sulit direproduksi. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

2.4.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait backend docking dan fallback.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Fallback diam-diam tanpa log dapat membuat protokol sulit direproduksi.
- Keputusan yang dianjurkan: Catat backend aktual dan versi engine.

Dalam konteks pembelajaran, keputusan yang disarankan adalah: Catat backend aktual dan versi engine. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Backend docking dan fallback” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai

ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “backend docking dan fallback”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

2.5 Workspace lintas sistem operasi

Lokasi data harus writable dan konsisten agar run, cache, katalog, serta database tidak bercampur dengan file program. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sisi quality control, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memakai DOCKFLOW_HOME bila tersedia, lalu lokasi user-specific pada Windows, macOS, atau XDG data pada Linux. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menulis hasil ke /opt atau folder aplikasi yang tidak writable. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

2.5.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait workspace lintas sistem operasi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.

- Tandai secara eksplisit risiko: Menulis hasil ke /opt atau folder aplikasi yang tidak writable.
- Keputusan yang dianjurkan: Pisahkan program, data pengguna, dan artefak run.

Dari sudut pandang metodologi, keputusan yang disarankan adalah: Pisahkan program, data pengguna, dan artefak run. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Workspace lintas sistem operasi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “workspace lintas sistem operasi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 2.3. Skenario keputusan untuk workspace lintas sistem operasi

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menulis hasil ke /opt atau folder aplikasi yang tidak writable.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.

Elemen	Uraian
Tindakan	Pisahkan program, data pengguna, dan artefak run.

2.6 Port dan peluncuran Streamlit

Aplikasi web lokal memerlukan port yang tidak bentrok dan address yang sesuai. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam konteks pembelajaran, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memakai `DEFAULT_PORT` 8802 dan dapat membaca `DOCKFLOW_PORT` ketika diluncurkan sebagai skrip Python. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengira port adalah bagian dari metode ilmiah; port hanya konfigurasi layanan. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

2.6.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait port dan peluncuran streamlit.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengira port adalah bagian dari metode ilmiah; port hanya konfigurasi layanan.
- Keputusan yang dianjurkan: Gunakan port konsisten dan dokumentasikan cara akses lokal.

Pada tahap ini, keputusan yang disarankan adalah: Gunakan port konsisten dan dokumentasikan cara akses lokal. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Port dan peluncuran Streamlit” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “port dan peluncuran streamlit”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

2.7 SQLite dan audit trail

Database ringan membantu mengikat proyek, job, hasil, serta tindakan audit. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sudut pandang metodologi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memakai WAL, foreign keys, tabel projects, jobs, results, dan audit. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan

hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengandalkan database saja tanpa raw files atau sebaliknya. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

2.7.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait sqlite dan audit trail.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengandalkan database saja tanpa raw files atau sebaliknya.
- Keputusan yang dianjurkan: Pertahankan database dan artefak file sebagai bukti komplementer.

Dalam praktik komputasi, keputusan yang disarankan adalah: Pertahankan database dan artefak file sebagai bukti komplementer. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “SQLite dan audit trail” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal.

Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “sqlite dan audit trail”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 2.4. Skenario keputusan untuk sqlite dan audit trail

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Mengandalkan database saja tanpa raw files atau sebaliknya.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Pertahankan database dan artefak file sebagai bukti komplementer.

2.8 Hash, cache, dan identitas data

Hash membantu mendeteksi duplikasi dan menghubungkan input dengan hasil. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Pada tahap ini, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: SHA-256 dipakai untuk teks/bytes dan cache key menggabungkan receptor, ligand, konfigurasi, serta seed. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap hash sebagai ukuran kemiripan kimia. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input,

job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

2.8.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait hash, cache, dan identitas data.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap hash sebagai ukuran kemiripan kimia.
- Keputusan yang dianjurkan: Gunakan hash untuk identitas byte/teks, bukan similarity molekuler.

Jika dibaca sebagai bagian dari workflow, keputusan yang disarankan adalah: Gunakan hash untuk identitas byte/teks, bukan similarity molekuler. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Hash, cache, dan identitas data” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “hash, cache, dan identitas data”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah

QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

2.9 Monitoring, stop aman, dan riwayat

Job panjang perlu status, log, progress, dan titik berhenti yang tidak merusak data. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Worker thread menulis `run_status`, log, dan berhenti pada checkpoint antar tahap/replikasi. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mematikan proses secara paksa saat file sedang ditulis. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

2.9.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait monitoring, stop aman, dan riwayat.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mematikan proses secara paksa saat file sedang ditulis.
- Keputusan yang dianjurkan: Gunakan Stop aman, baca log, lalu verifikasi artefak parsial.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Gunakan Stop aman, baca log, lalu verifikasi artefak parsial. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal,

perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Monitoring, stop aman, dan riwayat” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “monitoring, stop aman, dan riwayat”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 2.5. Skenario keputusan untuk monitoring, stop aman, dan riwayat

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Mematikan proses secara paksa saat file sedang ditulis.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Gunakan Stop aman, baca log, lalu verifikasi artefak parsial.

2.10 Rumus dan Indikator yang Relevan

Keputusan ilmiah = bukti input + QC + parameter + hasil + validasi + keterbatasan

Formula konseptual tersebut mengingatkan bahwa satu indikator tidak cukup untuk menopang kesimpulan workflow. Setiap bagian menyediakan jenis bukti yang berbeda dan saling melengkapi.

2.11 Praktikum Terpandu

1. Buat folder proyek baru dan tulis tujuan eksperimen dalam satu paragraf sebelum menjalankan aplikasi.
2. Catat sumber receptor, ligan, dan semua parameter yang relevan dengan bab ini.
3. Jalankan QC input; bila ada warning, tuliskan keputusan menerima atau memperbaiki beserta alasannya.
4. Jalankan satu baseline terlebih dahulu. Simpan konfigurasi, log, raw output, dan ringkasan hasil.
5. Lakukan satu perubahan terkontrol pada parameter yang sedang dipelajari, lalu bandingkan keluaran dengan baseline.
6. Tulis kesimpulan yang memisahkan observasi komputasi, interpretasi, dan keterbatasan.

2.11.1 Analisis Praktikum 1

Analisis praktikum 1 pada Bab 2 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

2.11.2 Analisis Praktikum 2

Analisis praktikum 2 pada Bab 2 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu

parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

2.11.3 Analisis Praktikum 3

Analisis praktikum 3 pada Bab 2 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

2.12 Troubleshooting dan Kesalahan yang Sering Terjadi

Tabel 2.6. Troubleshooting ringkas Bab 2

Area	Gejala/Risiko	Tindakan
Lapisan arsitektur aplikasi	Mencampur logika UI dengan hasil ilmiah tanpa state dan provenance.	Baca keluaran berdasarkan lapisan: input, eksekusi, hasil, dan handoff.
Bootstrap dependensi Python	Menganggap auto-install menjamin backend sistem seperti Open Babel tersedia.	Bedakan paket Python, executable sistem, dan fallback.
Backend docking dan fallback	Fallback diam-diam tanpa log dapat membuat protokol sulit direproduksi.	Catat backend aktual dan versi engine.
Workspace lintas sistem operasi	Menulis hasil ke /opt atau folder aplikasi yang tidak writable.	Pisahkan program, data pengguna, dan artefak run.

Area	Gejala/Risiko	Tindakan
Port dan peluncuran Streamlit	Mengira port adalah bagian dari metode ilmiah; port hanya konfigurasi layanan.	Gunakan port konsisten dan dokumentasikan cara akses lokal.
SQLite dan audit trail	Mengandalkan database saja tanpa raw files atau sebaliknya.	Pertahankan database dan artefak file sebagai bukti komplementer.

Jika masalah berada pada area lapisan arsitektur aplikasi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mencampur logika ui dengan hasil ilmiah tanpa state dan provenance. Tindakan awal yang lebih kuat adalah Baca keluaran berdasarkan lapisan: input, eksekusi, hasil, dan handoff. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area bootstrap dependensi python, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap auto-install menjamin backend sistem seperti open babel tersedia. Tindakan awal yang lebih kuat adalah Bedakan paket Python, executable sistem, dan fallback. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area backend docking dan fallback, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah fallback diam-diam tanpa log dapat membuat protokol sulit direproduksi. Tindakan awal yang lebih kuat adalah Catat backend aktual dan versi engine. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area workspace lintas sistem operasi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menulis hasil ke /opt atau folder aplikasi yang tidak writable. Tindakan awal yang lebih kuat adalah Pisahkan program, data pengguna, dan artefak run. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area port dan peluncuran streamlit, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengira port adalah bagian dari metode ilmiah; port hanya konfigurasi layanan. Tindakan awal yang lebih kuat adalah Gunakan port konsisten dan dokumentasikan cara akses lokal. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan

bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area sqlite dan audit trail, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengandalkan database saja tanpa raw files atau sebaliknya. Tindakan awal yang lebih kuat adalah Pertahankan database dan artefak file sebagai bukti komplementer. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

2.13 Latihan: 20 Soal

1. Jelaskan mengapa lapisan arsitektur aplikasi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
2. Jelaskan mengapa bootstrap dependensi python perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
3. Jelaskan mengapa backend docking dan fallback perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
4. Jelaskan mengapa workspace lintas sistem operasi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
5. Jelaskan mengapa port dan peluncuran streamlit perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
6. Jelaskan mengapa sqlite dan audit trail perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
7. Jelaskan mengapa hash, cache, dan identitas data perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
8. Jelaskan mengapa monitoring, stop aman, dan riwayat perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
9. Susun checklist minimum sebelum menjalankan batch pada Bab 2.

10. Sebuah hasil terlihat sangat baik pada satu indikator, tetapi gagal pada indikator QC lain. Bagaimana seharusnya hasil tersebut diperlakukan?
11. Mengapa perubahan satu parameter sebaiknya diuji secara terkontrol?
12. Jelaskan perbedaan antara data mentah, data turunan, dan interpretasi dalam workflow DockFlow.
13. Apa fungsi provenance ketika dua run menghasilkan hasil berbeda?
14. Kritik pernyataan: “Ligan dengan docking score paling rendah pasti merupakan kandidat terbaik.”
15. Bagaimana Anda menulis bagian metode agar run dapat direproduksi?
16. Mengapa visualisasi tidak boleh menggantikan pemeriksaan numerik?
17. Rancang satu eksperimen sensitivitas yang relevan dengan bab ini.
18. Apa yang harus dilakukan bila hasil tidak dapat direproduksi oleh kelompok lain?
19. Mengapa buku ini menekankan redocking sebelum screening besar?
20. Buat format ringkas laporan hasil yang baik.

2.14 Pembahasan 20 Soal

Pembahasan 1. Pemisahan data model, scientific service, executor, persistence, presentation, dan interoperability membuat workflow lebih mudah diaudit. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menyusun receptor, ligand, grid, job, pose, interaction, validation dalam lapisan yang saling terhubung. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mencampur logika ui dengan hasil ilmiah tanpa state dan provenance. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Baca keluaran berdasarkan lapisan: input, eksekusi, hasil, dan handoff.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 2. Aplikasi memeriksa paket inti dan dapat memasang paket yang hilang dari pemetaan internal. Pada DockFlow, aspek ini diwujudkan melalui: Streamlit, NumPy, pandas, RDKit, Plotly, psutil, openpyxl, XlsxWriter,

ReportLab, py3Dmol adalah stack inti; vina dan meeko opsional agar GUI tetap terbuka. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap auto-install menjamin backend sistem seperti open babel tersedia. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Bedakan paket Python, executable sistem, dan fallback.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 3. Reliabilitas meningkat bila workflow dapat memilih Python API atau CLI secara eksplisit. Pada DockFlow, aspek ini diwujudkan melalui: Mode Auto mencoba Vina Python API lalu CLI; error masing-masing dikumpulkan untuk diagnosis. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah fallback diam-diam tanpa log dapat membuat protokol sulit direproduksi. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Catat backend aktual dan versi engine.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 4. Lokasi data harus writable dan konsisten agar run, cache, katalog, serta database tidak bercampur dengan file program. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memakai DOCKFLOW_HOME bila tersedia, lalu lokasi user-specific pada Windows, macOS, atau XDG data pada Linux. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menulis hasil ke /opt atau folder aplikasi yang tidak writable. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Pisahkan program, data pengguna, dan artefak run.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file

atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 5. Aplikasi web lokal memerlukan port yang tidak bentrok dan address yang sesuai. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memakai DEFAULT_PORT 8802 dan dapat membaca DOCKFLOW_PORT ketika diluncurkan sebagai skrip Python. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengira port adalah bagian dari metode ilmiah; port hanya konfigurasi layanan. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan port konsisten dan dokumentasikan cara akses lokal.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 6. Database ringan membantu mengikat proyek, job, hasil, serta tindakan audit. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memakai WAL, foreign keys, tabel projects, jobs, results, dan audit. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengandalkan database saja tanpa raw files atau sebaliknya. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Pertahankan database dan artefak file sebagai bukti komplementer.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 7. Hash membantu mendeteksi duplikasi dan menghubungkan input dengan hasil. Pada DockFlow, aspek ini diwujudkan melalui: SHA-256 dipakai untuk teks/bytes dan cache key menggabungkan receptor, ligand, konfigurasi, serta seed. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah

menganggap hash sebagai ukuran kemiripan kimia. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan hash untuk identitas byte/teks, bukan similarity molekuler.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 8. Job panjang perlu status, log, progress, dan titik berhenti yang tidak merusak data. Pada DockFlow, aspek ini diwujudkan melalui: Worker thread menulis run_status, log, dan berhenti pada checkpoint antar tahap/replikasi. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mematikan proses secara paksa saat file sedang ditulis. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan Stop aman, baca log, lalu verifikasi artefak parsial.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 9. Checklist minimum harus memeriksa identitas input, kelengkapan struktur, parameter yang relevan dengan bab ini, ketersediaan backend, serta lokasi artefak yang akan disimpan. Tambahkan kriteria gagal/lanjut yang jelas. Prinsipnya: jangan memulai batch bila masalah dasar sudah terlihat pada QC. Setelah run, simpan konfigurasi, log, dan keluaran mentah agar keputusan dapat diaudit.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 10. Indikator yang baik tidak menghapus kegagalan QC. Hasil perlu ditandai bermasalah, penyebabnya ditelusuri pada tahap input/preparasi/eksekusi,

dan bila perlu run diulang dengan protokol yang diperbaiki. Dalam DockFlow, ranking bukan pengganti QC; hasil invalid bahkan mendapat penalty pada composite score. Laporan harus menyebutkan kegagalan dan tidak menjadikan satu angka sebagai pembenaran.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 11. Perubahan satu faktor memudahkan atribusi sebab-akibat. Bila beberapa parameter diubah sekaligus, pengguna tidak dapat mengetahui faktor mana yang menyebabkan perubahan pose, score, atau interaksi. Praktik yang baik adalah membuat baseline, mengubah satu parameter, menjaga input lain identik, membandingkan output, lalu mencatat keputusan. Pendekatan ini juga mempermudah reproduksi oleh kelompok lain.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 12. Data mentah meliputi struktur sumber dan output engine; data turunan meliputi pose terpilih, cluster, fingerprint, frekuensi residu, atau ranking; interpretasi adalah argumen ilmiah yang dibangun dari data tersebut. Ketiganya harus dipisahkan agar pembaca dapat menilai apakah transformasi data masuk akal dan apakah kesimpulan lebih kuat daripada bukti yang tersedia.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 13. Provenance menyediakan rekaman input, versi, seed, parameter, backend, waktu, dan artefak yang memungkinkan perbedaan dilacak. Tanpa provenance, perbedaan hasil mudah dianggap sebagai perilaku molekul padahal mungkin berasal dari perubahan preparasi atau konfigurasi. Dengan provenance, pengguna dapat menyusun perbandingan terkontrol dan mengulang kondisi secara lebih tepat.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 14. Pernyataan itu terlalu kuat. Docking score adalah salah satu keluaran model; kualitas pose, cluster, interaksi, redocking, protonasi, search space, dan sampling juga perlu diperiksa. DockFlow sengaja menyediakan composite ranking dan guardrail agar score tidak berdiri sendiri. Bahkan composite score tetap hanya alat prioritas internal, bukan bukti aktivitas atau afinitas eksperimen.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 15. Tuliskan sumber receptor dan ligan, metode preparasi, kebijakan water/hetero/chain, center dan size grid, backend Vina, scoring, exhaustiveness, num modes, energy range, seed, jumlah replikasi, threshold clustering, cutoff interaksi, metode redocking, versi perangkat lunak, serta cara ekspor/analisis. Sertakan file konfigurasi atau manifest bila tersedia.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 16. Visualisasi sangat baik untuk mendeteksi masalah besar seperti grid di luar pocket atau pose terbalik, tetapi tampilan dapat menyesatkan karena perspektif, skala, dan seleksi objek. Pemeriksaan numerik—koordinat, ukuran

box, distance, RMSD, score, dan log—memberikan bukti yang lebih eksplisit. Keduanya harus dipakai secara komplementer.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 17. Mulailah dengan baseline yang telah lolos QC. Pilih satu parameter yang benar-benar terkait dengan bab, misalnya padding grid, exhaustiveness, threshold clustering, atau kebijakan preparasi. Jalankan minimal dua atau tiga tingkat parameter dengan input dan seed/replikasi yang terkontrol. Bandingkan pose, cluster, score, interaksi, dan validasi; jangan hanya membandingkan ranking akhir.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 18. Bandingkan checksum/hash input, versi engine, metode preparasi, konfigurasi lengkap, seed, receptor PDBQT, grid, dan output mentah. Pastikan tidak ada langkah manual yang tidak tercatat. Bila perbedaan tetap ada, laporkan lingkungan komputasi dan backend aktual. Reprodusibilitas adalah diagnosis berbasis bukti, bukan sekadar mencoba ulang perintah yang sama.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 19. Jika ligan referensi tersedia, redocking memberi uji awal apakah kombinasi receptor preparation, grid, sampling, dan pose mapping mampu mereproduksi struktur yang diketahui. Kegagalan pada tahap ini memberi sinyal untuk memperbaiki protokol sebelum biaya komputasi dan interpretasi diperluas ke banyak ligan.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang

berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 20. Laporan minimal memuat tujuan, input dan sumber struktur, parameter, status backend, QC, hasil redocking bila ada, tabel pose/cluster, analisis interaksi, ranking, keterbatasan, serta lampiran provenance dan raw output. Nyatakan dengan jelas mana data komputasi, mana interpretasi, dan mana fakta dari sumber eksperimen atau literatur.

Penerapan pada Bab 2: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

2.15 Soal Pengembangan

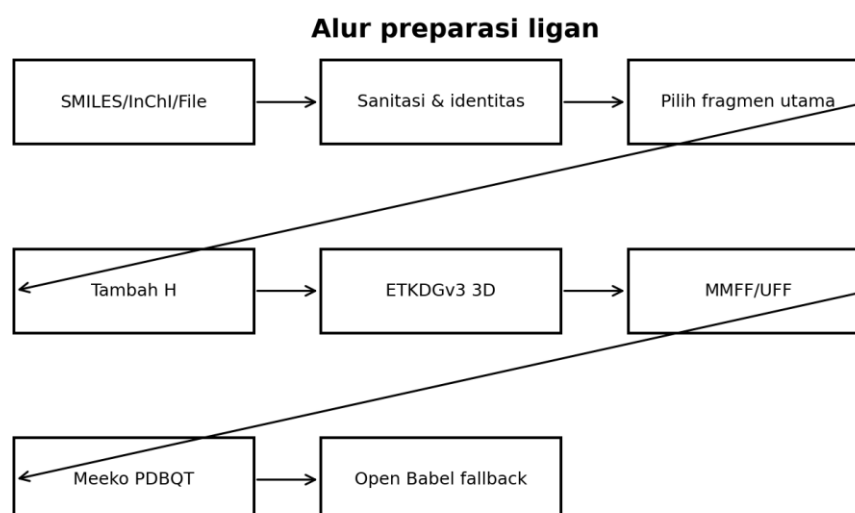
1. Rancang mini-proyek yang menguji satu keputusan metodologis utama Bab 2 dengan baseline dan minimal dua kondisi uji.
2. Susun rubrik penilaian yang memberi bobot pada QC, reproduksibilitas, validasi, interpretasi, dan dokumentasi.
3. Tulis satu contoh klaim yang terlalu kuat dari hasil docking, lalu perbaiki menjadi klaim yang sesuai kekuatan bukti.
4. Buat diagram data lineage dari input sampai satu keluaran akhir dan tunjukkan di mana provenance harus disimpan.
5. Susun daftar informasi yang harus diberikan kepada peneliti lain agar ia dapat mengulang run tanpa bertanya kembali.

BAB 3. INPUT, QUALITY CONTROL, DAN PREPARASI LIGAN

Bab ini membahas representasi molekul, validasi input, deduplikasi, deskriptor awal, generasi struktur 3D, serta konversi ke PDBQT. Fokusnya adalah memastikan bahwa struktur yang masuk ke docking memiliki identitas dan geometri yang dapat dipertanggungjawabkan.

3.1 Capaian Pembelajaran

- Menjelaskan konsep utama Bab 3 dengan istilah yang tepat dan membedakan data komputasi dari inferensi.
- Menghubungkan konsep dengan parameter, file, dan artefak yang benar-benar dihasilkan DockFlow.
- Menyusun quality-control dan kriteria keputusan sebelum menjalankan batch.
- Menganalisis kegagalan secara bertahap dan mendokumentasikan perubahan protokol.
- Menyusun laporan yang dapat diperiksa dan direproduksi.



Gambar 3.1. Skema utama pembahasan Bab 3: Input, Quality Control, Dan Preparasi Ligan.

Tabel 3.1. Format input ligan yang didukung

Format	Catatan penggunaan
SMILES	Nama=SMILES atau baris SMILES
InChI	Dapat ditulis dengan prefix InChI=
CSV	Memerlukan kolom SMILES atau InChI; nama opsional
SDF/SD	Dapat memuat multi-record dan nama molekul

Format	Catatan penggunaan
MOL/MOL2/PDB	Dibaca sebagai struktur molekul
PDBQT	Dipertahankan sebagai input docking siap pakai
TXT	Diinterpretasi sebagai baris SMILES/InChI

Tabel 3.2. Deskriptor QC awal ligan

Deskriptor	Peran
Formula	Pemeriksaan komposisi
Heavy atoms	Ukuran non-hidrogen dan denominator ligand efficiency
Formal charge	Peringatan protonasi/charge state
MW	Konteks ukuran molekul
TPSA	Konteks polaritas topologis
HBD/HBA	Kapasitas donor/akseptor H-bond
Rotatable bonds	Konteks fleksibilitas

3.2 Representasi SMILES dan InChI

Representasi line notation menyimpan konektivitas dan, pada tingkat tertentu, informasi stereokimia; keduanya bukan koordinat 3D docking. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menerima baris Nama=SMILES atau InChI serta membangun objek RDKit. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap smiles sama dengan konformer 3d final. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

3.2.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait representasi smiles dan inchi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap SMILES sama dengan konformer 3D final.
- Keputusan yang dianjurkan: Pisahkan identitas molekul dari geometri yang akan didocking.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Pisahkan identitas molekul dari geometri yang akan didocking. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Representasi SMILES dan InChI” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan

adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “representasi smiles dan inchi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

3.3 Input multi-format

File SDF, MOL, MOL2, PDB, PDBQT, CSV, dan TXT membawa tingkat informasi yang berbeda. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Jika dibaca sebagai bagian dari workflow, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow membaca multi-record SDF, kolom SMILES/InChI pada CSV, dan mempertahankan PDBQT siap pakai. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mencampur koordinat dan bond order yang tidak konsisten antar-format. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

3.3.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait input multi-format.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mencampur koordinat dan bond order yang tidak konsisten antar-format.
- Keputusan yang dianjurkan: Pilih format yang mempertahankan informasi yang dibutuhkan tahap berikutnya.

Dari sisi quality control, keputusan yang disarankan adalah: Pilih format yang mempertahankan informasi yang dibutuhkan tahap berikutnya. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Input multi-format” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “input multi-format”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 3.2. Skenario keputusan untuk input multi-format

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Mencampur koordinat dan bond order yang tidak konsisten antar-format.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Pilih format yang mempertahankan informasi yang dibutuhkan tahap berikutnya.

3.4 Sanitasi dan identitas molekul

Sanitasi RDKit memeriksa valensi dan membantu mencegah struktur tidak masuk akal masuk ke workflow. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Untuk keperluan reproduksibilitas, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: `mol_identity` menghasilkan canonical SMILES, InChIKey/hash, formula, heavy atoms, charge, MW, TPSA, HBD, HBA, dan rotatable bonds. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menggunakan deskriptor awal sebagai bukti aktivitas biologis. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

3.4.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait sanitasi dan identitas molekul.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menggunakan deskriptor awal sebagai bukti aktivitas biologis.
- Keputusan yang dianjurkan: Gunakan deskriptor untuk QC dan konteks, bukan kesimpulan efek.

Dalam konteks pembelajaran, keputusan yang disarankan adalah: Gunakan deskriptor untuk QC dan konteks, bukan kesimpulan efek. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan

sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Sanitasi dan identitas molekul” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “sanitasi dan identitas molekul”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

3.5 Deduplikasi ligan

Duplikasi meningkatkan beban komputasi dan dapat membiasakan interpretasi frekuensi. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sisi quality control, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow membandingkan InChIKey atau hash struktur untuk menyisihkan struktur identik. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai

oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap stereoisomer atau protonation state berbeda sebagai duplikat tanpa pemeriksaan. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

3.5.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait deduplikasi ligan.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap stereoisomer atau protonation state berbeda sebagai duplikat tanpa pemeriksaan.
- Keputusan yang dianjurkan: Tinjau identitas kimia, bukan nama file saja.

Dari sudut pandang metodologi, keputusan yang disarankan adalah: Tinjau identitas kimia, bukan nama file saja. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Deduplikasi ligan” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “deduplikasi ligan”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 3.3. Skenario keputusan untuk deduplikasi ligan

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menganggap stereoisomer atau protonation state berbeda sebagai duplikat tanpa pemeriksaan.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Tinjau identitas kimia, bukan nama file saja.

3.6 Pemilihan fragmen dan penambahan hidrogen

Garam atau struktur multi-fragmen memerlukan kebijakan eksplisit sebelum docking. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam konteks pembelajaran, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Default preparasi dapat memilih fragmen dengan heavy atom terbanyak lalu menambah H dengan RDKit. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menghapus komponen bermakna hanya karena lebih kecil. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan

catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

3.6.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait pemilihan fragmen dan penambahan hidrogen.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menghapus komponen bermakna hanya karena lebih kecil.
- Keputusan yang dianjurkan: Putuskan kebijakan fragmen berdasarkan tujuan eksperimen.

Pada tahap ini, keputusan yang disarankan adalah: Putuskan kebijakan fragmen berdasarkan tujuan eksperimen. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Pemilihan fragmen dan penambahan hidrogen” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “pemilihan fragmen dan penambahan hidrogen”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

3.7 Generasi 3D ETKDGV3

Distance geometry berbasis pengetahuan membantu menghasilkan konformer awal yang masuk akal dari struktur 2D. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sudut pandang metodologi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memakai ETKDGV3, random seed, dan useRandomCoords bila struktur belum 3D. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap satu konformer hasil embedding mewakili ensemble larutan. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

3.7.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait generasi 3d etkdgv3.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap satu konformer hasil embedding mewakili ensemble larutan.
- Keputusan yang dianjurkan: Perlakukan geometri awal sebagai input sampling, bukan keadaan tunggal yang pasti.

Dalam praktik komputasi, keputusan yang disarankan adalah: Perlakukan geometri awal sebagai input sampling, bukan keadaan tunggal yang pasti. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai.

Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Generasi 3D ETKDGV3” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “generasi 3d etkdgv3”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 3.4. Skenario keputusan untuk generasi 3d etkdgv3

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menganggap satu konformer hasil embedding mewakili ensemble larutan.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Perlakukan geometri awal sebagai input sampling, bukan keadaan tunggal yang pasti.

3.8 Optimasi MMFF94 dan UFF

Optimasi force field dapat merapikan geometri lokal sebelum konversi PDBQT. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap

residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Pada tahap ini, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow mencoba MMFF94, lalu UFF sebagai fallback; `preserve_coords` dipakai untuk ligan referensi redocking. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengoptimasi ligan referensi kristalografi sehingga koordinat validasi berubah. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

3.8.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait optimasi mmff94 dan uff.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengoptimasi ligan referensi kristalografi sehingga koordinat validasi berubah.
- Keputusan yang dianjurkan: Gunakan `preserve_coords` ketika tujuan adalah membandingkan pose dengan referensi.

Jika dibaca sebagai bagian dari workflow, keputusan yang disarankan adalah: Gunakan `preserve_coords` ketika tujuan adalah membandingkan pose dengan referensi. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow

menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Optimasi MMFF94 dan UFF” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “optimasi mmff94 dan uff”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

3.9 Meeko dan Open Babel

PDBQT membawa atom type, charge, dan informasi torsional yang diperlukan Vina. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow mengutamakan Meeko; bila gagal dan obabel tersedia, konversi SDF ke PDBQT memakai Gasteiger charge. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mencampur metode preparasi antar-ligan tanpa mencatatnya. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

3.9.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait meeko dan open babel.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mencampur metode preparasi antar-ligan tanpa mencatatnya.
- Keputusan yang dianjurkan: Pastikan metode PDBQT konsisten atau dokumentasikan fallback per ligan.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Pastikan metode PDBQT konsisten atau dokumentasikan fallback per ligan. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Meeko dan Open Babel” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “meeko dan open babel”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 3.5. Skenario keputusan untuk meeko dan open babel

Elemen	Uraian
--------	--------

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Mencampur metode preparasi antar-ligan tanpa mencatatnya.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Pastikan metode PDBQT konsisten atau dokumentasikan fallback per ligan.

3.10 Rumus dan Indikator yang Relevan

Keputusan ilmiah = bukti input + QC + parameter + hasil + validasi + keterbatasan

Formula konseptual tersebut mengingatkan bahwa satu indikator tidak cukup untuk menopang kesimpulan workflow. Setiap bagian menyediakan jenis bukti yang berbeda dan saling melengkapi.

3.11 Praktikum Terpandu

1. Buat folder proyek baru dan tulis tujuan eksperimen dalam satu paragraf sebelum menjalankan aplikasi.
2. Catat sumber receptor, ligan, dan semua parameter yang relevan dengan bab ini.
3. Jalankan QC input; bila ada warning, tuliskan keputusan menerima atau memperbaiki beserta alasannya.
4. Jalankan satu baseline terlebih dahulu. Simpan konfigurasi, log, raw output, dan ringkasan hasil.
5. Lakukan satu perubahan terkontrol pada parameter yang sedang dipelajari, lalu bandingkan keluaran dengan baseline.
6. Tulis kesimpulan yang memisahkan observasi komputasi, interpretasi, dan keterbatasan.

3.11.1 Analisis Praktikum 1

Analisis praktikum 1 pada Bab 3 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa

apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

3.11.2 Analisis Praktikum 2

Analisis praktikum 2 pada Bab 3 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

3.11.3 Analisis Praktikum 3

Analisis praktikum 3 pada Bab 3 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

3.12 Troubleshooting dan Kesalahan yang Sering Terjadi

Tabel 3.6. Troubleshooting ringkas Bab 3

Area	Gejala/Risiko	Tindakan
Representasi SMILES dan InChI	Menganggap SMILES sama dengan konformer 3D final.	Pisahkan identitas molekul dari geometri yang akan didocking.
Input multi-format	Mencampur koordinat dan bond order yang tidak konsisten antar-format.	Pilih format yang mempertahankan informasi yang dibutuhkan tahap berikutnya.
Sanitasi dan identitas molekul	Menggunakan deskriptor awal sebagai bukti aktivitas biologis.	Gunakan deskriptor untuk QC dan konteks, bukan kesimpulan efek.
Deduplikasi ligan	Menganggap stereoisomer atau protonation state berbeda sebagai duplikat tanpa pemeriksaan.	Tinjau identitas kimia, bukan nama file saja.
Pemilihan fragmen dan penambahan hidrogen	Menghapus komponen bermakna hanya karena lebih kecil.	Putuskan kebijakan fragmen berdasarkan tujuan eksperimen.
Generasi 3D ETKDGV3	Menganggap satu konformer hasil embedding mewakili ensemble larutan.	Perlakukan geometri awal sebagai input sampling, bukan keadaan tunggal yang pasti.

Jika masalah berada pada area representasi smiles dan inchi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap smiles sama dengan konformer 3d final. Tindakan awal yang lebih kuat adalah Pisahkan identitas molekul dari geometri yang akan didocking. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area input multi-format, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mencampur koordinat dan bond order yang tidak konsisten antar-format. Tindakan awal yang lebih kuat adalah Pilih format yang mempertahankan informasi yang dibutuhkan tahap berikutnya. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area sanitasi dan identitas molekul, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menggunakan deskriptor awal sebagai bukti aktivitas biologis. Tindakan awal yang lebih kuat adalah Gunakan deskriptor untuk QC dan konteks, bukan kesimpulan efek. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan

artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area deduplikasi ligan, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap stereoisomer atau protonation state berbeda sebagai duplikat tanpa pemeriksaan. Tindakan awal yang lebih kuat adalah Tinjau identitas kimia, bukan nama file saja. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area pemilihan fragmen dan penambahan hidrogen, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menghapus komponen bermakna hanya karena lebih kecil. Tindakan awal yang lebih kuat adalah Putuskan kebijakan fragmen berdasarkan tujuan eksperimen. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area generasi 3d etkdgv3, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap satu konformer hasil embedding mewakili ensemble larutan. Tindakan awal yang lebih kuat adalah Perlakukan geometri awal sebagai input sampling, bukan keadaan tunggal yang pasti. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

3.13 Latihan: 20 Soal

1. Jelaskan mengapa representasi smiles dan inchi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
2. Jelaskan mengapa input multi-format perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
3. Jelaskan mengapa sanitasi dan identitas molekul perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
4. Jelaskan mengapa deduplikasi ligan perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.

5. Jelaskan mengapa pemilihan fragmen dan penambahan hidrogen perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
6. Jelaskan mengapa generasi 3d etkdgv3 perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
7. Jelaskan mengapa optimasi mmff94 dan uff perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
8. Jelaskan mengapa meeko dan open babel perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
9. Susun checklist minimum sebelum menjalankan batch pada Bab 3.
10. Sebuah hasil terlihat sangat baik pada satu indikator, tetapi gagal pada indikator QC lain. Bagaimana seharusnya hasil tersebut diperlakukan?
11. Mengapa perubahan satu parameter sebaiknya diuji secara terkontrol?
12. Jelaskan perbedaan antara data mentah, data turunan, dan interpretasi dalam workflow DockFlow.
13. Apa fungsi provenance ketika dua run menghasilkan hasil berbeda?
14. Kritik pernyataan: “Ligan dengan docking score paling rendah pasti merupakan kandidat terbaik.”
15. Bagaimana Anda menulis bagian metode agar run dapat direproduksi?
16. Mengapa visualisasi tidak boleh menggantikan pemeriksaan numerik?
17. Rancang satu eksperimen sensitivitas yang relevan dengan bab ini.
18. Apa yang harus dilakukan bila hasil tidak dapat direproduksi oleh kelompok lain?
19. Mengapa buku ini menekankan redocking sebelum screening besar?
20. Buat format ringkas laporan hasil yang baik.

3.14 Pembahasan 20 Soal

Pembahasan 1. Representasi line notation menyimpan konektivitas dan, pada tingkat tertentu, informasi stereokimia; keduanya bukan koordinat 3D docking. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menerima baris Nama=SMILES atau InChI serta membangun objek RDKit. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat.

Kesalahan yang perlu dihindari adalah menganggap smiles sama dengan konformer 3d final. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Pisahkan identitas molekul dari geometri yang akan didocking.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 2. File SDF, MOL, MOL2, PDB, PDBQT, CSV, dan TXT membawa tingkat informasi yang berbeda. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow membaca multi-record SDF, kolom SMILES/InChI pada CSV, dan mempertahankan PDBQT siap pakai. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mencampur koordinat dan bond order yang tidak konsisten antar-format. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Pilih format yang mempertahankan informasi yang dibutuhkan tahap berikutnya.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 3. Sanitasi RDKit memeriksa valensi dan membantu mencegah struktur tidak masuk akal masuk ke workflow. Pada DockFlow, aspek ini diwujudkan melalui: mol_identity menghasilkan canonical SMILES, InChIKey/hash, formula, heavy atoms, charge, MW, TPSA, HBD, HBA, dan rotatable bonds. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menggunakan deskriptor awal sebagai bukti aktivitas biologis. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan deskriptor untuk QC dan konteks, bukan kesimpulan efek.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang

berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 4. Duplikasi meningkatkan beban komputasi dan dapat membiasakan interpretasi frekuensi. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow membandingkan InChIKey atau hash struktur untuk menyisihkan struktur identik. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap stereoisomer atau protonation state berbeda sebagai duplikat tanpa pemeriksaan. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Tinjau identitas kimia, bukan nama file saja.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 5. Garam atau struktur multi-fragmen memerlukan kebijakan eksplisit sebelum docking. Pada DockFlow, aspek ini diwujudkan melalui: Default preparasi dapat memilih fragmen dengan heavy atom terbanyak lalu menambah H dengan RDKit. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menghapus komponen bermakna hanya karena lebih kecil. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Putuskan kebijakan fragmen berdasarkan tujuan eksperimen.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 6. Distance geometry berbasis pengetahuan membantu menghasilkan konformer awal yang masuk akal dari struktur 2D. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memakai ETKDGv3, random seed, dan useRandomCoords bila struktur belum 3D. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap satu konformer hasil embedding

mewakili ensemble larutan. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Perlakukan geometri awal sebagai input sampling, bukan keadaan tunggal yang pasti.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 7. Optimasi force field dapat merapikan geometri lokal sebelum konversi PDBQT. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow mencoba MMFF94, lalu UFF sebagai fallback; `preserve_coords` dipakai untuk ligan referensi redocking. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengoptimasi ligan referensi kristalografi sehingga koordinat validasi berubah. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan `preserve_coords` ketika tujuan adalah membandingkan pose dengan referensi.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 8. PDBQT membawa atom type, charge, dan informasi torsional yang diperlukan Vina. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow mengutamakan Meeko; bila gagal dan obabel tersedia, konversi SDF ke PDBQT memakai Gasteiger charge. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mencampur metode preparasi antar-ligan tanpa mencatatnya. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Pastikan metode PDBQT konsisten atau dokumentasikan fallback per ligan.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika

terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 9. Checklist minimum harus memeriksa identitas input, kelengkapan struktur, parameter yang relevan dengan bab ini, ketersediaan backend, serta lokasi artefak yang akan disimpan. Tambahkan kriteria gagal/lanjut yang jelas. Prinsipnya: jangan memulai batch bila masalah dasar sudah terlihat pada QC. Setelah run, simpan konfigurasi, log, dan keluaran mentah agar keputusan dapat diaudit.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 10. Indikator yang baik tidak menghapus kegagalan QC. Hasil perlu ditandai bermasalah, penyebabnya ditelusuri pada tahap input/preparasi/eksekusi, dan bila perlu run diulang dengan protokol yang diperbaiki. Dalam DockFlow, ranking bukan pengganti QC; hasil invalid bahkan mendapat penalty pada composite score. Laporan harus menyebutkan kegagalan dan tidak menjadikan satu angka sebagai pembenaran.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 11. Perubahan satu faktor memudahkan atribusi sebab-akibat. Bila beberapa parameter diubah sekaligus, pengguna tidak dapat mengetahui faktor mana yang menyebabkan perubahan pose, score, atau interaksi. Praktik yang baik adalah membuat baseline, mengubah satu parameter, menjaga input lain identik, membandingkan output, lalu mencatat keputusan. Pendekatan ini juga mempermudah reproduksi oleh kelompok lain.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 12. Data mentah meliputi struktur sumber dan output engine; data turunan meliputi pose terpilih, cluster, fingerprint, frekuensi residu, atau ranking; interpretasi adalah argumen ilmiah yang dibangun dari data tersebut. Ketiganya harus dipisahkan agar pembaca dapat menilai apakah transformasi data masuk akal dan apakah kesimpulan lebih kuat daripada bukti yang tersedia.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 13. Provenance menyediakan rekaman input, versi, seed, parameter, backend, waktu, dan artefak yang memungkinkan perbedaan dilacak. Tanpa provenance, perbedaan hasil mudah dianggap sebagai perilaku molekul padahal mungkin berasal dari perubahan preparasi atau konfigurasi. Dengan provenance, pengguna dapat menyusun perbandingan terkontrol dan mengulang kondisi secara lebih tepat.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 14. Pernyataan itu terlalu kuat. Docking score adalah salah satu keluaran model; kualitas pose, cluster, interaksi, redocking, protonasi, search space, dan sampling juga perlu diperiksa. DockFlow sengaja menyediakan composite ranking dan guardrail agar score tidak berdiri sendiri. Bahkan composite score tetap hanya alat prioritisasi internal, bukan bukti aktivitas atau afinitas eksperimen.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 15. Tuliskan sumber receptor dan ligan, metode preparasi, kebijakan water/hetero/chain, center dan size grid, backend Vina, scoring, exhaustiveness, num modes, energy range, seed, jumlah replikasi, threshold

clustering, cutoff interaksi, metode redocking, versi perangkat lunak, serta cara ekspor/analisis. Sertakan file konfigurasi atau manifest bila tersedia.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 16. Visualisasi sangat baik untuk mendeteksi masalah besar seperti grid di luar pocket atau pose terbalik, tetapi tampilan dapat menyesatkan karena perspektif, skala, dan seleksi objek. Pemeriksaan numerik—koordinat, ukuran box, distance, RMSD, score, dan log—memberikan bukti yang lebih eksplisit. Keduanya harus dipakai secara komplementer.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 17. Mulailah dengan baseline yang telah lolos QC. Pilih satu parameter yang benar-benar terkait dengan bab, misalnya padding grid, exhaustiveness, threshold clustering, atau kebijakan preparasi. Jalankan minimal dua atau tiga tingkat parameter dengan input dan seed/replikasi yang terkontrol. Bandingkan pose, cluster, score, interaksi, dan validasi; jangan hanya membandingkan ranking akhir.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 18. Bandingkan checksum/hash input, versi engine, metode preparasi, konfigurasi lengkap, seed, receptor PDBQT, grid, dan output mentah. Pastikan tidak ada langkah manual yang tidak tercatat. Bila perbedaan tetap ada, laporkan lingkungan komputasi dan backend aktual. Reprodusibilitas adalah diagnosis berbasis bukti, bukan sekadar mencoba ulang perintah yang sama.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang

berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 19. Jika ligan referensi tersedia, redocking memberi uji awal apakah kombinasi receptor preparation, grid, sampling, dan pose mapping mampu mereproduksi struktur yang diketahui. Kegagalan pada tahap ini memberi sinyal untuk memperbaiki protokol sebelum biaya komputasi dan interpretasi diperluas ke banyak ligan.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 20. Laporan minimal memuat tujuan, input dan sumber struktur, parameter, status backend, QC, hasil redocking bila ada, tabel pose/cluster, analisis interaksi, ranking, keterbatasan, serta lampiran provenance dan raw output. Nyatakan dengan jelas mana data komputasi, mana interpretasi, dan mana fakta dari sumber eksperimen atau literatur.

Penerapan pada Bab 3: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

3.15 Soal Pengembangan

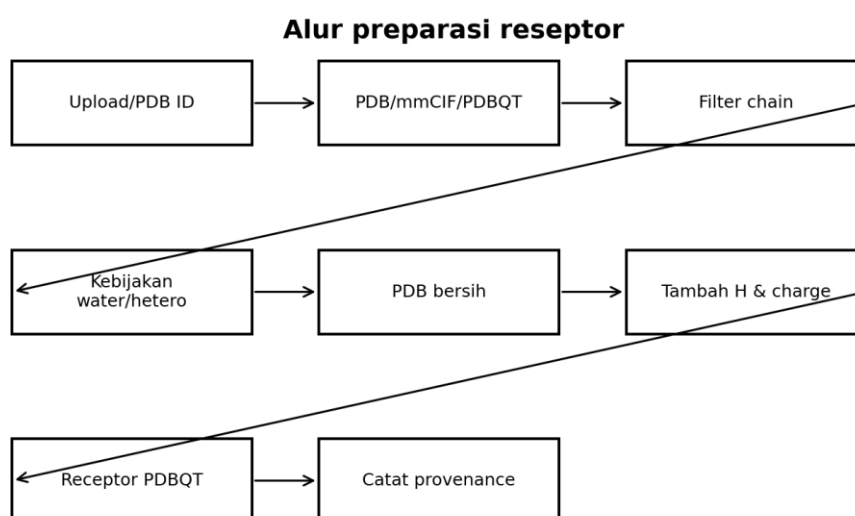
1. Rancang mini-proyek yang menguji satu keputusan metodologis utama Bab 3 dengan baseline dan minimal dua kondisi uji.
2. Susun rubrik penilaian yang memberi bobot pada QC, reproduksibilitas, validasi, interpretasi, dan dokumentasi.
3. Tulis satu contoh klaim yang terlalu kuat dari hasil docking, lalu perbaiki menjadi klaim yang sesuai kekuatan bukti.
4. Buat diagram data lineage dari input sampai satu keluaran akhir dan tunjukkan di mana provenance harus disimpan.
5. Susun daftar informasi yang harus diberikan kepada peneliti lain agar ia dapat mengulang run tanpa bertanya kembali.

BAB 4. AKUISISI, PEMBERSIHAN, DAN PREPARASI RESEPTOR

Kualitas reseptor sering menjadi sumber kegagalan docking yang paling sulit dilacak. Bab ini membahas PDB/mmCIF/PDBQT, pemilihan chain, air kristal, heteroatom, hidrogen, charge, dan provenance cleaning.

4.1 Capaian Pembelajaran

- Menjelaskan konsep utama Bab 4 dengan istilah yang tepat dan membedakan data komputasi dari inferensi.
- Menghubungkan konsep dengan parameter, file, dan artefak yang benar-benar dihasilkan DockFlow.
- Menyusun quality-control dan kriteria keputusan sebelum menjalankan batch.
- Menganalisis kegagalan secara bertahap dan mendokumentasikan perubahan protokol.
- Menyusun laporan yang dapat diperiksa dan direproduksi.



Gambar 4.1. Skema utama pembahasan Bab 4: Akuisisi, Pembersihan, Dan Preparasi Reseptor.

Tabel 4.1. Keputusan preparasi reseptor

Keputusan	Pertanyaan QC
Chain	Chain mana yang merepresentasikan target/pocket?
Water	Apakah ada water terstruktur yang relevan?
HETATM	Kofaktor/ion mana yang harus dipertahankan?
Hydrogen	Metode dan protonation state apa yang digunakan?

Keputusan	Pertanyaan QC
Charge	Bagaimana charge ditetapkan?
PDBQT	Tool dan versi apa yang menghasilkan file final?

4.2 Sumber struktur reseptor

Struktur eksperimen membawa informasi resolusi, ligan ko-kristal, chain, alternate location, ion, dan komponen kristal yang perlu dibaca sebelum preparasi. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menerima upload PDB/PDBQT/mmCIF atau mengunduh PDB ID empat karakter. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengambil struktur hanya berdasarkan nama protein tanpa memeriksa konstruksi dan kualitas eksperimen. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

4.2.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait sumber struktur reseptor.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengambil struktur hanya berdasarkan nama protein tanpa memeriksa konstruksi dan kualitas eksperimen.
- Keputusan yang dianjurkan: Catat identitas entry, chain, metode, resolusi, dan komponen penting.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Catat identitas entry, chain, metode, resolusi, dan komponen penting. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Sumber struktur reseptor” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “sumber struktur reseptor”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

4.3 PDB versus mmCIF

PDB adalah format historis yang mudah dibaca, sedangkan mmCIF lebih kaya dan sesuai untuk struktur kompleks. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Jika dibaca sebagai bagian dari workflow, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow dapat

mengonversi mmCIF ke PDB melalui Open Babel sebelum pembersihan. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap konversi selalu tanpa kehilangan informasi metadata. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

4.3.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait pdb versus mmcif.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap konversi selalu tanpa kehilangan informasi metadata.
- Keputusan yang dianjurkan: Simpan file sumber asli bersama file turunan.

Dari sisi quality control, keputusan yang disarankan adalah: Simpan file sumber asli bersama file turunan. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “PDB versus mmCIF” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai

ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “pdb versus mmCIF”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 4.2. Skenario keputusan untuk pdb versus mmCIF

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menganggap konversi selalu tanpa kehilangan informasi metadata.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Simpan file sumber asli bersama file turunan.

4.4 Pemilihan chain

Chain menentukan objek biologis yang benar untuk docking dan dapat memengaruhi pocket. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Untuk keperluan reproduksibilitas, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Pengguna dapat menyaring chain sebelum receptor PDBQT dibuat. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah memilih chain secara arbitrer pada multimer atau interface binding site. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian

informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

4.4.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait pemilihan chain.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Memilih chain secara arbitrer pada multimer atau interface binding site.
- Keputusan yang dianjurkan: Gunakan informasi struktur dan tujuan biologis untuk memilih chain.

Dalam konteks pembelajaran, keputusan yang disarankan adalah: Gunakan informasi struktur dan tujuan biologis untuk memilih chain. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Pemilihan chain” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “pemilihan chain”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus.

Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

4.5 Kebijakan water

Molekul air dapat berupa artefak kristal, mediator interaksi, atau bagian struktural pocket. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sisi quality control, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menyediakan opsi `remove_waters` dan mencatat statistik cleaning. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menghapus atau mempertahankan semua water tanpa alasan. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

4.5.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait kebijakan water.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menghapus atau mempertahankan semua water tanpa alasan.
- Keputusan yang dianjurkan: Tentukan kebijakan water sebagai parameter protokol yang dapat dipertanggungjawabkan.

Dari sudut pandang metodologi, keputusan yang disarankan adalah: Tentukan kebijakan water sebagai parameter protokol yang dapat dipertanggungjawabkan. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol

baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Kebijakan water” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “kebijakan water”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 4.3. Skenario keputusan untuk kebijakan water

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menghapus atau mempertahankan semua water tanpa alasan.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Tentukan kebijakan water sebagai parameter protokol yang dapat dipertanggungjawabkan.

4.6 HETATM, ion, kofaktor, dan ligan ko-kristal

HETATM mencakup banyak jenis entitas; tidak semuanya harus diperlakukan sama. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak

terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam konteks pembelajaran, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow dapat `remove_hetero` dengan daftar `keep_hetero` tertentu. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menghapus kofaktor esensial atau logam katalitik secara otomatis. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

4.6.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait hetatm, ion, kofaktor, dan ligan ko-kristal.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menghapus kofaktor esensial atau logam katalitik secara otomatis.
- Keputusan yang dianjurkan: Buat daftar komponen yang dipertahankan dan alasan ilmiahnya.

Pada tahap ini, keputusan yang disarankan adalah: Buat daftar komponen yang dipertahankan dan alasan ilmiahnya. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur

eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “HETATM, ion, kofaktor, dan ligan ko-kristal” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “hetatm, ion, kofaktor, dan ligan ko-kristal”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

4.7 Penambahan hidrogen dan charge reseptor

Docking memerlukan representasi atom dan charge yang kompatibel dengan engine. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sudut pandang metodologi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Untuk PDB, DockFlow menggunakan Open Babel dengan opsi rigid, add H, dan Gasteiger charge menuju PDBQT. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengabaikan protonation state residu situs aktif. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan

catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

4.7.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait penambahan hidrogen dan charge reseptor.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengabaikan protonation state residu situs aktif.
- Keputusan yang dianjurkan: Tinjau kondisi pH dan residu yang peka terhadap protonasi bila relevan.

Dalam praktik komputasi, keputusan yang disarankan adalah: Tinjau kondisi pH dan residu yang peka terhadap protonasi bila relevan. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Penambahan hidrogen dan charge reseptor” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “penambahan hidrogen dan charge reseptor”: apakah struktur dan state identik, apakah reseptor dan grid sama, apakah backend/seed/exhaustiveness

sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 4.4. Skenario keputusan untuk penambahan hidrogen dan charge reseptor

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Mengabaikan protonation state residu situs aktif.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Tinjau kondisi pH dan residu yang peka terhadap protonasi bila relevan.

4.8 Reseptor PDBQT siap pakai

PDBQT yang sudah dipreparasi dapat menghindari langkah konversi internal, tetapi kualitasnya tetap harus diketahui. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Pada tahap ini, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Jika input PDBQT, DockFlow menyalin file sebagai reseptor final dan menghitung hash. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap format siap pakai berarti preparasi ilmiahnya benar. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

4.8.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait reseptor pdbqt siap pakai.

- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap format siap pakai berarti preparasi ilmiahnya benar.
- Keputusan yang dianjurkan: Dokumentasikan tool dan parameter yang menghasilkan PDBQT.

Jika dibaca sebagai bagian dari workflow, keputusan yang disarankan adalah: Dokumentasikan tool dan parameter yang menghasilkan PDBQT. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Reseptor PDBQT siap pakai” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “reseptor pdbqt siap pakai”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

4.9 Provenance cleaning

Provenance memungkinkan pengguna menjelaskan perbedaan hasil antar-run atau antar-laboratorium. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan

kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menyimpan source, cleaning stats, hash, clean PDB, dan receptor PDBQT. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah hanya menyimpan file akhir sehingga keputusan preparasi hilang. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

4.9.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait provenance cleaning.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Hanya menyimpan file akhir sehingga keputusan preparasi hilang.
- Keputusan yang dianjurkan: Simpan input mentah, file bersih, dan metadata perubahan bersama-sama.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Simpan input mentah, file bersih, dan metadata perubahan bersama-sama. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur

eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Provenance cleaning” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “provenance cleaning”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 4.5. Skenario keputusan untuk provenance cleaning

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Hanya menyimpan file akhir sehingga keputusan preparasi hilang.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Simpan input mentah, file bersih, dan metadata perubahan bersama-sama.

4.10 Rumus dan Indikator yang Relevan

Keputusan ilmiah = bukti input + QC + parameter + hasil + validasi + keterbatasan

Formula konseptual tersebut mengingatkan bahwa satu indikator tidak cukup untuk menopang kesimpulan workflow. Setiap bagian menyediakan jenis bukti yang berbeda dan saling melengkapi.

4.11 Praktikum Terpandu

1. Buat folder proyek baru dan tulis tujuan eksperimen dalam satu paragraf sebelum menjalankan aplikasi.

2. Catat sumber receptor, ligan, dan semua parameter yang relevan dengan bab ini.
3. Jalankan QC input; bila ada warning, tuliskan keputusan menerima atau memperbaiki beserta alasannya.
4. Jalankan satu baseline terlebih dahulu. Simpan konfigurasi, log, raw output, dan ringkasan hasil.
5. Lakukan satu perubahan terkontrol pada parameter yang sedang dipelajari, lalu bandingkan keluaran dengan baseline.
6. Tulis kesimpulan yang memisahkan observasi komputasi, interpretasi, dan keterbatasan.

4.11.1 Analisis Praktikum 1

Analisis praktikum 1 pada Bab 4 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

4.11.2 Analisis Praktikum 2

Analisis praktikum 2 pada Bab 4 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat

praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

4.11.3 Analisis Praktikum 3

Analisis praktikum 3 pada Bab 4 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

4.12 Troubleshooting dan Kesalahan yang Sering Terjadi

Tabel 4.6. Troubleshooting ringkas Bab 4

Area	Gejala/Risiko	Tindakan
Sumber struktur reseptor	Mengambil struktur hanya berdasarkan nama protein tanpa memeriksa konstruksi dan kualitas eksperimen.	Catat identitas entry, chain, metode, resolusi, dan komponen penting.
PDB versus mmCIF	Menganggap konversi selalu tanpa kehilangan informasi metadata.	Simpan file sumber asli bersama file turunan.
Pemilihan chain	Memilih chain secara arbitrer pada multimer atau interface binding site.	Gunakan informasi struktur dan tujuan biologis untuk memilih chain.
Kebijakan water	Menghapus atau mempertahankan semua water tanpa alasan.	Tentukan kebijakan water sebagai parameter protokol yang dapat dipertanggungjawabkan.
HETATM, ion, kofaktor, dan ligan ko-kristal	Menghapus kofaktor esensial atau logam katalitik secara otomatis.	Buat daftar komponen yang dipertahankan dan alasan ilmiahnya.
Penambahan hidrogen dan charge reseptor	Mengabaikan protonation state residu situs aktif.	Tinjau kondisi pH dan residu yang peka terhadap protonasi bila relevan.

Jika masalah berada pada area sumber struktur reseptor, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengambil struktur hanya berdasarkan nama protein tanpa memeriksa konstruksi dan kualitas eksperimen. Tindakan awal yang lebih kuat adalah Catat identitas entry, chain, metode, resolusi, dan komponen penting. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area pdb versus mmcif, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap konversi selalu tanpa kehilangan informasi metadata. Tindakan awal yang lebih kuat adalah Simpan file sumber asli bersama file turunan. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area pemilihan chain, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah memilih chain secara arbitrer pada multimer atau interface binding site. Tindakan awal yang lebih kuat adalah Gunakan informasi struktur dan tujuan biologis untuk memilih chain. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area kebijakan water, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menghapus atau mempertahankan semua water tanpa alasan. Tindakan awal yang lebih kuat adalah Tentukan kebijakan water sebagai parameter protokol yang dapat dipertanggungjawabkan. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area hetatm, ion, kofaktor, dan ligan ko-kristal, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menghapus kofaktor esensial atau logam katalitik secara otomatis. Tindakan awal yang lebih kuat adalah Buat daftar komponen yang dipertahankan dan alasan ilmiahnya. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area penambahan hidrogen dan charge reseptor, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengabaikan protonation state residu situs aktif. Tindakan awal yang lebih kuat adalah Tinjau kondisi pH dan residu yang peka terhadap protonasi bila relevan. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan

artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

4.13 Latihan: 20 Soal

1. Jelaskan mengapa sumber struktur reseptor perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
2. Jelaskan mengapa pdb versus mmCIF perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
3. Jelaskan mengapa pemilihan chain perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
4. Jelaskan mengapa kebijakan water perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
5. Jelaskan mengapa hetatm, ion, kofaktor, dan ligan ko-kristal perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
6. Jelaskan mengapa penambahan hidrogen dan charge reseptor perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
7. Jelaskan mengapa reseptor pdbqt siap pakai perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
8. Jelaskan mengapa provenance cleaning perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
9. Susun checklist minimum sebelum menjalankan batch pada Bab 4.
10. Sebuah hasil terlihat sangat baik pada satu indikator, tetapi gagal pada indikator QC lain. Bagaimana seharusnya hasil tersebut diperlakukan?
11. Mengapa perubahan satu parameter sebaiknya diuji secara terkontrol?
12. Jelaskan perbedaan antara data mentah, data turunan, dan interpretasi dalam workflow DockFlow.
13. Apa fungsi provenance ketika dua run menghasilkan hasil berbeda?

14. Kritik pernyataan: “Ligan dengan docking score paling rendah pasti merupakan kandidat terbaik.”
15. Bagaimana Anda menulis bagian metode agar run dapat direproduksi?
16. Mengapa visualisasi tidak boleh menggantikan pemeriksaan numerik?
17. Rancang satu eksperimen sensitivitas yang relevan dengan bab ini.
18. Apa yang harus dilakukan bila hasil tidak dapat direproduksi oleh kelompok lain?
19. Mengapa buku ini menekankan redocking sebelum screening besar?
20. Buat format ringkas laporan hasil yang baik.

4.14 Pembahasan 20 Soal

Pembahasan 1. Struktur eksperimen membawa informasi resolusi, ligan ko-kristal, chain, alternate location, ion, dan komponen kristal yang perlu dibaca sebelum preparasi. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menerima upload PDB/PDBQT/mmCIF atau mengunduh PDB ID empat karakter. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengambil struktur hanya berdasarkan nama protein tanpa memeriksa konstruksi dan kualitas eksperimen. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Catat identitas entry, chain, metode, resolusi, dan komponen penting.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 2. PDB adalah format historis yang mudah dibaca, sedangkan mmCIF lebih kaya dan sesuai untuk struktur kompleks. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow dapat mengonversi mmCIF ke PDB melalui Open Babel sebelum pembersihan. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap konversi selalu tanpa kehilangan informasi metadata. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Simpan file sumber asli bersama file turunan.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 3. Chain menentukan objek biologis yang benar untuk docking dan dapat memengaruhi pocket. Pada DockFlow, aspek ini diwujudkan melalui: Pengguna dapat menyaring chain sebelum receptor PDBQT dibuat. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah memilih chain secara arbitrer pada multimer atau interface binding site. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan informasi struktur dan tujuan biologis untuk memilih chain.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 4. Molekul air dapat berupa artefak kristal, mediator interaksi, atau bagian struktural pocket. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menyediakan opsi `remove_waters` dan mencatat statistik cleaning. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menghapus atau mempertahankan semua water tanpa alasan. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Tentukan kebijakan water sebagai parameter protokol yang dapat dipertanggungjawabkan.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 5. HETATM mencakup banyak jenis entitas; tidak semuanya harus diperlakukan sama. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow dapat `remove_hetero` dengan daftar `keep_hetero` tertentu. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menghapus kofaktor esensial atau logam katalitik secara otomatis. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Buat daftar komponen yang dipertahankan dan alasan ilmiahnya.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 6. Docking memerlukan representasi atom dan charge yang kompatibel dengan engine. Pada DockFlow, aspek ini diwujudkan melalui: Untuk PDB, DockFlow menggunakan Open Babel dengan opsi rigid, add H, dan Gasteiger charge menuju PDBQT. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengabaikan protonation state residu situs aktif. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Tinjau kondisi pH dan residu yang peka terhadap protonasi bila relevan.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 7. PDBQT yang sudah dipreparasi dapat menghindari langkah konversi internal, tetapi kualitasnya tetap harus diketahui. Pada DockFlow, aspek ini diwujudkan melalui: Jika input PDBQT, DockFlow menyalin file sebagai receptor final dan menghitung hash. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap format siap pakai berarti preparasi ilmiahnya benar. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Dokumentasikan tool dan parameter yang menghasilkan PDBQT.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 8. Provenance memungkinkan pengguna menjelaskan perbedaan hasil antar-run atau antar-laboratorium. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menyimpan source, cleaning stats, hash, clean PDB, dan receptor PDBQT. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah hanya menyimpan file akhir sehingga keputusan preparasi hilang. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Simpan input mentah, file bersih, dan metadata perubahan bersama-sama.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 9. Checklist minimum harus memeriksa identitas input, kelengkapan struktur, parameter yang relevan dengan bab ini, ketersediaan backend, serta lokasi artefak yang akan disimpan. Tambahkan kriteria gagal/lanjut yang jelas. Prinsipnya: jangan memulai batch bila masalah dasar sudah terlihat pada QC. Setelah run, simpan konfigurasi, log, dan keluaran mentah agar keputusan dapat diaudit.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 10. Indikator yang baik tidak menghapus kegagalan QC. Hasil perlu ditandai bermasalah, penyebabnya ditelusuri pada tahap input/preparasi/eksekusi, dan bila perlu run diulang dengan protokol yang diperbaiki. Dalam DockFlow, ranking bukan pengganti QC; hasil invalid bahkan mendapat penalty pada composite score. Laporan harus menyebutkan kegagalan dan tidak menjadikan satu angka sebagai pembenaran.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 11. Perubahan satu faktor memudahkan atribusi sebab-akibat. Bila beberapa parameter diubah sekaligus, pengguna tidak dapat mengetahui faktor mana yang menyebabkan perubahan pose, score, atau interaksi. Praktik yang baik adalah membuat baseline, mengubah satu parameter, menjaga input lain identik, membandingkan output, lalu mencatat keputusan. Pendekatan ini juga mempermudah reproduksi oleh kelompok lain.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 12. Data mentah meliputi struktur sumber dan output engine; data turunan meliputi pose terpilih, cluster, fingerprint, frekuensi residu, atau ranking; interpretasi adalah argumen ilmiah yang dibangun dari data tersebut. Ketiganya harus dipisahkan agar pembaca dapat menilai apakah transformasi data masuk akal dan apakah kesimpulan lebih kuat daripada bukti yang tersedia.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 13. Provenance menyediakan rekaman input, versi, seed, parameter, backend, waktu, dan artefak yang memungkinkan perbedaan dilacak. Tanpa provenance, perbedaan hasil mudah dianggap sebagai perilaku molekul padahal mungkin berasal dari perubahan preparasi atau konfigurasi. Dengan provenance, pengguna dapat menyusun perbandingan terkontrol dan mengulang kondisi secara lebih tepat.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 14. Pernyataan itu terlalu kuat. Docking score adalah salah satu keluaran model; kualitas pose, cluster, interaksi, redocking, protonasi, search space, dan sampling juga perlu diperiksa. DockFlow sengaja menyediakan composite ranking dan guardrail agar score tidak berdiri sendiri. Bahkan

composite score tetap hanya alat prioritisasi internal, bukan bukti aktivitas atau afinitas eksperimen.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 15. Tuliskan sumber receptor dan ligan, metode preparasi, kebijakan water/hetero/chain, center dan size grid, backend Vina, scoring, exhaustiveness, num modes, energy range, seed, jumlah replikasi, threshold clustering, cutoff interaksi, metode redocking, versi perangkat lunak, serta cara ekspor/analisis. Sertakan file konfigurasi atau manifest bila tersedia.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 16. Visualisasi sangat baik untuk mendeteksi masalah besar seperti grid di luar pocket atau pose terbalik, tetapi tampilan dapat menyesatkan karena perspektif, skala, dan seleksi objek. Pemeriksaan numerik—koordinat, ukuran box, distance, RMSD, score, dan log—memberikan bukti yang lebih eksplisit. Keduanya harus dipakai secara komplementer.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 17. Mulailah dengan baseline yang telah lolos QC. Pilih satu parameter yang benar-benar terkait dengan bab, misalnya padding grid, exhaustiveness, threshold clustering, atau kebijakan preparasi. Jalankan minimal dua atau tiga tingkat parameter dengan input dan seed/replikasi yang terkontrol. Bandingkan pose, cluster, score, interaksi, dan validasi; jangan hanya membandingkan ranking akhir.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang

berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 18. Bandingkan checksum/hash input, versi engine, metode preparasi, konfigurasi lengkap, seed, receptor PDBQT, grid, dan output mentah. Pastikan tidak ada langkah manual yang tidak tercatat. Bila perbedaan tetap ada, laporkan lingkungan komputasi dan backend aktual. Reprodusibilitas adalah diagnosis berbasis bukti, bukan sekadar mencoba ulang perintah yang sama.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 19. Jika ligan referensi tersedia, redocking memberi uji awal apakah kombinasi receptor preparation, grid, sampling, dan pose mapping mampu mereproduksi struktur yang diketahui. Kegagalan pada tahap ini memberi sinyal untuk memperbaiki protokol sebelum biaya komputasi dan interpretasi diperluas ke banyak ligan.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 20. Laporan minimal memuat tujuan, input dan sumber struktur, parameter, status backend, QC, hasil redocking bila ada, tabel pose/cluster, analisis interaksi, ranking, keterbatasan, serta lampiran provenance dan raw output. Nyatakan dengan jelas mana data komputasi, mana interpretasi, dan mana fakta dari sumber eksperimen atau literatur.

Penerapan pada Bab 4: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

4.15 Soal Pengembangan

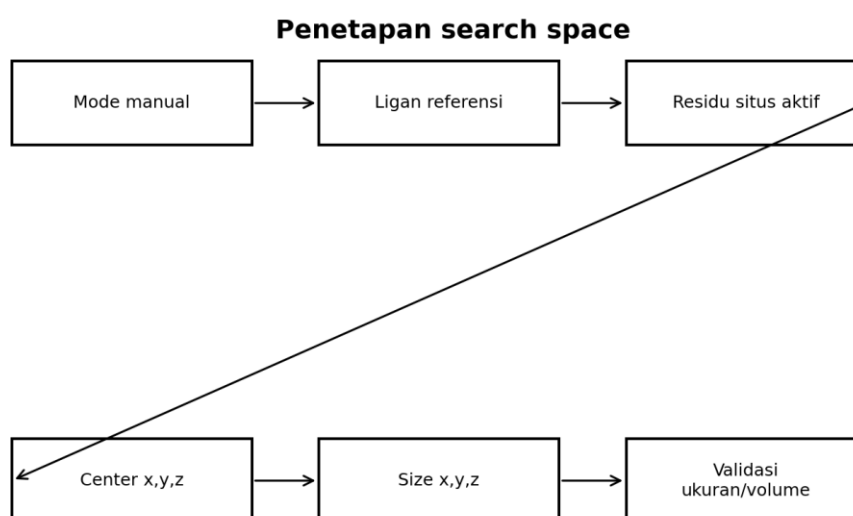
1. Rancang mini-proyek yang menguji satu keputusan metodologis utama Bab 4 dengan baseline dan minimal dua kondisi uji.
2. Susun rubrik penilaian yang memberi bobot pada QC, reproduksibilitas, validasi, interpretasi, dan dokumentasi.
3. Tulis satu contoh klaim yang terlalu kuat dari hasil docking, lalu perbaiki menjadi klaim yang sesuai kekuatan bukti.
4. Buat diagram data lineage dari input sampai satu keluaran akhir dan tunjukkan di mana provenance harus disimpan.
5. Susun daftar informasi yang harus diberikan kepada peneliti lain agar ia dapat mengulang run tanpa bertanya kembali.

BAB 5. BINDING SITE, GRID, DAN DESAIN SEARCH SPACE

Search space menentukan wilayah yang dapat dijelajahi docking. Grid yang terlalu kecil memotong ruang orientasi, sedangkan grid terlalu besar memperberat sampling dan dapat menurunkan fokus pencarian.

5.1 Capaian Pembelajaran

- Menjelaskan konsep utama Bab 5 dengan istilah yang tepat dan membedakan data komputasi dari inferensi.
- Menghubungkan konsep dengan parameter, file, dan artefak yang benar-benar dihasilkan DockFlow.
- Menyusun quality-control dan kriteria keputusan sebelum menjalankan batch.
- Menganalisis kegagalan secara bertahap dan mendokumentasikan perubahan protokol.
- Menyusun laporan yang dapat diperiksa dan direproduksi.



Gambar 5.1. Skema utama pembahasan Bab 5: Binding Site, Grid, Dan Desain Search Space.

Tabel 5.1. Warning validasi grid pada DockFlow

Kondisi	Makna praktis
Sisi > 40 Å	Search space sangat besar; sampling dapat tidak efisien
Volume > 100000 Å ³	Pertimbangkan docking terarah bila pocket diketahui
Sisi < 8 Å	Ligan mungkin tidak memiliki ruang orientasi cukup
Size ≤ 0	Konfigurasi tidak valid

Kondisi	Makna praktis
Nilai non-finite	Konfigurasi tidak valid

5.2 Makna center dan size

Search space Vina dinyatakan oleh pusat x-y-z dan panjang sisi x-y-z dalam angstrom. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menyimpan center dan size sebagai bagian job specification. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menyalin grid dari studi lain tanpa memastikan frame koordinat sama. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

5.2.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait makna center dan size.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menyalin grid dari studi lain tanpa memastikan frame koordinat sama.
- Keputusan yang dianjurkan: Pastikan grid berada pada frame reseptor yang digunakan.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Pastikan grid berada pada frame reseptor yang digunakan. Keputusan tersebut sebaiknya

ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Makna center dan size” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “makna center dan size”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

5.3 Grid manual

Grid manual berguna ketika situs aktif sudah diketahui dari literatur atau inspeksi struktur. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Jika dibaca sebagai bagian dari workflow, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Pengguna memasukkan center dan size secara eksplisit. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai

oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah memilih angka berdasarkan perkiraan visual tanpa dokumentasi. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

5.3.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait grid manual.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Memilih angka berdasarkan perkiraan visual tanpa dokumentasi.
- Keputusan yang dianjurkan: Catat sumber koordinat dan alasan ukuran setiap sisi.

Dari sisi quality control, keputusan yang disarankan adalah: Catat sumber koordinat dan alasan ukuran setiap sisi. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Grid manual” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “grid manual”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 5.2. Skenario keputusan untuk grid manual

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Memilih angka berdasarkan perkiraan visual tanpa dokumentasi.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Catat sumber koordinat dan alasan ukuran setiap sisi.

5.4 Grid dari ligan referensi

Ligan ko-kristal memberi petunjuk geometri pocket yang kuat untuk redocking dan docking terarah. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Untuk keperluan reproduksibilitas, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menghitung centroid dan bounding box ligan referensi dengan padding. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menggunakan referensi yang tidak berada pada pocket target. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT,

output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

5.4.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait grid dari ligan referensi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menggunakan referensi yang tidak berada pada pocket target.
- Keputusan yang dianjurkan: Verifikasi identitas dan posisi ligan sebelum mengotomatisasi grid.

Dalam konteks pembelajaran, keputusan yang disarankan adalah: Verifikasi identitas dan posisi ligan sebelum mengotomatisasi grid. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Grid dari ligan referensi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “grid dari ligan referensi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

5.5 Grid dari residu situs aktif

Daftar residu dapat dipakai untuk menghitung centroid ketika ligan referensi tidak tersedia. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sisi quality control, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow membaca token residu dan menghitung pusat dari atom yang cocok. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah daftar residu salah chain atau numbering menghasilkan pusat yang keliru. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

5.5.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait grid dari residu situs aktif.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Daftar residu salah chain atau numbering menghasilkan pusat yang keliru.
- Keputusan yang dianjurkan: Uji token residu terhadap struktur yang benar.

Dari sudut pandang metodologi, keputusan yang disarankan adalah: Uji token residu terhadap struktur yang benar. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina

menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Grid dari residu situs aktif” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “grid dari residu situs aktif”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 5.3. Skenario keputusan untuk grid dari residu situs aktif

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Daftar residu salah chain atau numbering menghasilkan pusat yang keliru.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Uji token residu terhadap struktur yang benar.

5.6 Padding dan ruang orientasi

Padding memberi ruang translasi dan rotasi di sekitar objek referensi. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam konteks pembelajaran, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Nilai padding disimpan sebagai

grid_padding dan dipakai ketika box diturunkan dari ligan. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah padding terlalu sempit atau terlalu besar tanpa uji sensitivitas. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

5.6.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait padding dan ruang orientasi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Padding terlalu sempit atau terlalu besar tanpa uji sensitivitas.
- Keputusan yang dianjurkan: Gunakan uji skala dan periksa apakah seluruh pocket tercakup.

Pada tahap ini, keputusan yang disarankan adalah: Gunakan uji skala dan periksa apakah seluruh pocket tercakup. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Padding dan ruang orientasi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung

tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “padding dan ruang orientasi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

5.7 Validasi ukuran dan volume

Pemeriksaan geometri mencegah search space non-finite, negatif, terlalu kecil, atau sangat besar. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sudut pandang metodologi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memberi warning jika sisi $>40 \text{ \AA}$, volume $>100000 \text{ \AA}^3$, atau sisi $<8 \text{ \AA}$. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengabaikan warning sebagai pesan kosmetik. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

5.7.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait validasi ukuran dan volume.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.

- Tandai secara eksplisit risiko: Mengabaikan warning sebagai pesan kosmetik.
- Keputusan yang dianjurkan: Tafsirkan warning sebagai pemicu pemeriksaan protokol.

Dalam praktik komputasi, keputusan yang disarankan adalah: Tafsirkan warning sebagai pemicu pemeriksaan protokol. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Validasi ukuran dan volume” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “validasi ukuran dan volume”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 5.4. Skenario keputusan untuk validasi ukuran dan volume

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Mengabaikan warning sebagai pesan kosmetik.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Tafsirkan warning sebagai pemicu pemeriksaan protokol.

5.8 Directed versus blind docking

Directed docking berfokus pada pocket yang diketahui; blind docking memperluas ruang pencarian dan beban sampling. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Pada tahap ini, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow lebih cocok untuk search space eksplisit dan tervalidasi. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menyebut grid besar sebagai blind docking yang pasti menemukan semua situs. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

5.8.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait directed versus blind docking.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menyebut grid besar sebagai blind docking yang pasti menemukan semua situs.
- Keputusan yang dianjurkan: Jelaskan cakupan pencarian dan keterbatasannya.

Jika dibaca sebagai bagian dari workflow, keputusan yang disarankan adalah: Jelaskan cakupan pencarian dan keterbatasannya. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan

parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Directed versus blind docking” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “directed versus blind docking”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

5.9 Visualisasi grid

Visualisasi membantu mendeteksi center yang berada di luar protein atau box yang tidak mencakup pocket. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menyediakan Plotly box dan py3Dmol bila struktur tersedia. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan

demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menggunakan visualisasi tanpa memeriksa koordinat numerik. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

5.9.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait visualisasi grid.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menggunakan visualisasi tanpa memeriksa koordinat numerik.
- Keputusan yang dianjurkan: Gabungkan inspeksi visual dan validasi numerik.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Gabungkan inspeksi visual dan validasi numerik. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Visualisasi grid” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan

adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “visualisasi grid”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 5.5. Skenario keputusan untuk visualisasi grid

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menggunakan visualisasi tanpa memeriksa koordinat numerik.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Gabungkan inspeksi visual dan validasi numerik.

5.10 Rumus dan Indikator yang Relevan

Volume search space merupakan hasil perkalian tiga sisi box. Rumus ini membantu pengguna memahami bahwa sedikit perubahan tiap sisi dapat memperbesar volume pencarian secara signifikan.

$$V_{\text{grid}} = S_x \times S_y \times S_z$$

5.11 Praktikum Terpandu

1. Buat folder proyek baru dan tulis tujuan eksperimen dalam satu paragraf sebelum menjalankan aplikasi.
2. Catat sumber receptor, ligan, dan semua parameter yang relevan dengan bab ini.
3. Jalankan QC input; bila ada warning, tuliskan keputusan menerima atau memperbaiki beserta alasannya.
4. Jalankan satu baseline terlebih dahulu. Simpan konfigurasi, log, raw output, dan ringkasan hasil.
5. Lakukan satu perubahan terkontrol pada parameter yang sedang dipelajari, lalu bandingkan keluaran dengan baseline.
6. Tulis kesimpulan yang memisahkan observasi komputasi, interpretasi, dan keterbatasan.

5.11.1 Analisis Praktikum 1

Analisis praktikum 1 pada Bab 5 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

5.11.2 Analisis Praktikum 2

Analisis praktikum 2 pada Bab 5 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

5.11.3 Analisis Praktikum 3

Analisis praktikum 3 pada Bab 5 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

5.12 Troubleshooting dan Kesalahan yang Sering Terjadi

Tabel 5.6. Troubleshooting ringkas Bab 5

Area	Gejala/Risiko	Tindakan
Makna center dan size	Menyalin grid dari studi lain tanpa memastikan frame koordinat sama.	Pastikan grid berada pada frame reseptor yang digunakan.
Grid manual	Memilih angka berdasarkan perkiraan visual tanpa dokumentasi.	Catat sumber koordinat dan alasan ukuran setiap sisi.
Grid dari ligan referensi	Menggunakan referensi yang tidak berada pada pocket target.	Verifikasi identitas dan posisi ligan sebelum mengotomatisasi grid.
Grid dari residu situs aktif	Daftar residu salah chain atau numbering menghasilkan pusat yang keliru.	Uji token residu terhadap struktur yang benar.
Padding dan ruang orientasi	Padding terlalu sempit atau terlalu besar tanpa uji sensitivitas.	Gunakan uji skala dan periksa apakah seluruh pocket tercakup.
Validasi ukuran dan volume	Mengabaikan warning sebagai pesan kosmetik.	Tafsirkan warning sebagai pemicu pemeriksaan protokol.

Jika masalah berada pada area makna center dan size, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menyalin grid dari studi lain tanpa memastikan frame koordinat sama. Tindakan awal yang lebih kuat adalah Pastikan grid berada pada frame reseptor yang digunakan. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area grid manual, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah memilih angka berdasarkan perkiraan visual tanpa dokumentasi. Tindakan awal yang lebih kuat adalah Catat sumber koordinat dan alasan ukuran setiap sisi. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area grid dari ligan referensi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menggunakan

referensi yang tidak berada pada pocket target. Tindakan awal yang lebih kuat adalah Verifikasi identitas dan posisi ligan sebelum mengotomatisasi grid. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area grid dari residu situs aktif, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah daftar residu salah chain atau numbering menghasilkan pusat yang keliru. Tindakan awal yang lebih kuat adalah Uji token residu terhadap struktur yang benar. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area padding dan ruang orientasi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah padding terlalu sempit atau terlalu besar tanpa uji sensitivitas. Tindakan awal yang lebih kuat adalah Gunakan uji skala dan periksa apakah seluruh pocket tercakup. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area validasi ukuran dan volume, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengabaikan warning sebagai pesan kosmetik. Tindakan awal yang lebih kuat adalah Tafsirkan warning sebagai pemicu pemeriksaan protokol. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

5.13 Latihan: 20 Soal

1. Jelaskan mengapa makna center dan size perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
2. Jelaskan mengapa grid manual perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
3. Jelaskan mengapa grid dari ligan referensi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
4. Jelaskan mengapa grid dari residu situs aktif perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.

5. Jelaskan mengapa padding dan ruang orientasi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
6. Jelaskan mengapa validasi ukuran dan volume perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
7. Jelaskan mengapa directed versus blind docking perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
8. Jelaskan mengapa visualisasi grid perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
9. Susun checklist minimum sebelum menjalankan batch pada Bab 5.
10. Sebuah hasil terlihat sangat baik pada satu indikator, tetapi gagal pada indikator QC lain. Bagaimana seharusnya hasil tersebut diperlakukan?
11. Mengapa perubahan satu parameter sebaiknya diuji secara terkontrol?
12. Jelaskan perbedaan antara data mentah, data turunan, dan interpretasi dalam workflow DockFlow.
13. Apa fungsi provenance ketika dua run menghasilkan hasil berbeda?
14. Kritik pernyataan: “Ligan dengan docking score paling rendah pasti merupakan kandidat terbaik.”
15. Bagaimana Anda menulis bagian metode agar run dapat direproduksi?
16. Mengapa visualisasi tidak boleh menggantikan pemeriksaan numerik?
17. Rancang satu eksperimen sensitivitas yang relevan dengan bab ini.
18. Apa yang harus dilakukan bila hasil tidak dapat direproduksi oleh kelompok lain?
19. Mengapa buku ini menekankan redocking sebelum screening besar?
20. Buat format ringkas laporan hasil yang baik.

5.14 Pembahasan 20 Soal

Pembahasan 1. Search space Vina dinyatakan oleh pusat x-y-z dan panjang sisi x-y-z dalam angstrom. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menyimpan center dan size sebagai bagian job specification. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menyalin grid dari studi lain

tanpa memastikan frame koordinat sama. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Pastikan grid berada pada frame reseptor yang digunakan.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 2. Grid manual berguna ketika situs aktif sudah diketahui dari literatur atau inspeksi struktur. Pada DockFlow, aspek ini diwujudkan melalui: Pengguna memasukkan center dan size secara eksplisit. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah memilih angka berdasarkan perkiraan visual tanpa dokumentasi. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Catat sumber koordinat dan alasan ukuran setiap sisi.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 3. Ligan ko-kristal memberi petunjuk geometri pocket yang kuat untuk redocking dan docking terarah. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menghitung centroid dan bounding box ligan referensi dengan padding. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menggunakan referensi yang tidak berada pada pocket target. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Verifikasi identitas dan posisi ligan sebelum mengotomatisasi grid.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 4. Daftar residu dapat dipakai untuk menghitung centroid ketika ligan referensi tidak tersedia. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow membaca token residu dan menghitung pusat dari atom yang cocok. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah daftar residu salah chain atau numbering menghasilkan pusat yang keliru. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Uji token residu terhadap struktur yang benar.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 5. Padding memberi ruang translasi dan rotasi di sekitar objek referensi. Pada DockFlow, aspek ini diwujudkan melalui: Nilai padding disimpan sebagai `grid_padding` dan dipakai ketika box diturunkan dari ligan. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah padding terlalu sempit atau terlalu besar tanpa uji sensitivitas. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan uji skala dan periksa apakah seluruh pocket tercakup.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 6. Pemeriksaan geometri mencegah search space non-finite, negatif, terlalu kecil, atau sangat besar. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memberi warning jika sisi $>40 \text{ \AA}$, volume $>100000 \text{ \AA}^3$, atau sisi $<8 \text{ \AA}$. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengabaikan warning sebagai pesan kosmetik. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Tafsirkan warning sebagai pemicu pemeriksaan protokol.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 7. Directed docking berfokus pada pocket yang diketahui; blind docking memperluas ruang pencarian dan beban sampling. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow lebih cocok untuk search space eksplisit dan tervalidasi. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menyebut grid besar sebagai blind docking yang pasti menemukan semua situs. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Jelaskan cakupan pencarian dan keterbatasannya.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 8. Visualisasi membantu mendeteksi center yang berada di luar protein atau box yang tidak mencakup pocket. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menyediakan Plotly box dan py3Dmol bila struktur tersedia. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menggunakan visualisasi tanpa memeriksa koordinat numerik. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gabungkan inspeksi visual dan validasi numerik.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 9. Checklist minimum harus memeriksa identitas input, kelengkapan struktur, parameter yang relevan dengan bab ini, ketersediaan backend, serta lokasi artefak yang akan disimpan. Tambahkan kriteria gagal/lanjut yang jelas. Prinsipnya: jangan memulai batch bila masalah dasar sudah terlihat pada QC. Setelah run, simpan konfigurasi, log, dan keluaran mentah agar keputusan dapat diaudit.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 10. Indikator yang baik tidak menghapus kegagalan QC. Hasil perlu ditandai bermasalah, penyebabnya ditelusuri pada tahap input/preparasi/eksekusi, dan bila perlu run diulang dengan protokol yang diperbaiki. Dalam DockFlow, ranking bukan pengganti QC; hasil invalid bahkan mendapat penalty pada composite score. Laporan harus menyebutkan kegagalan dan tidak menjadikan satu angka sebagai pembenaran.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 11. Perubahan satu faktor memudahkan atribusi sebab-akibat. Bila beberapa parameter diubah sekaligus, pengguna tidak dapat mengetahui faktor mana yang menyebabkan perubahan pose, score, atau interaksi. Praktik yang baik adalah membuat baseline, mengubah satu parameter, menjaga input lain identik, membandingkan output, lalu mencatat keputusan. Pendekatan ini juga mempermudah reproduksi oleh kelompok lain.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 12. Data mentah meliputi struktur sumber dan output engine; data turunan meliputi pose terpilih, cluster, fingerprint, frekuensi residu, atau ranking; interpretasi adalah argumen ilmiah yang dibangun dari data tersebut. Ketiganya harus dipisahkan agar pembaca dapat menilai apakah transformasi data masuk akal dan apakah kesimpulan lebih kuat daripada bukti yang tersedia.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 13. Provenance menyediakan rekaman input, versi, seed, parameter, backend, waktu, dan artefak yang memungkinkan perbedaan dilacak. Tanpa provenance, perbedaan hasil mudah dianggap sebagai perilaku molekul padahal mungkin berasal dari perubahan preparasi atau konfigurasi. Dengan provenance, pengguna dapat menyusun perbandingan terkontrol dan mengulang kondisi secara lebih tepat.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 14. Pernyataan itu terlalu kuat. Docking score adalah salah satu keluaran model; kualitas pose, cluster, interaksi, redocking, protonasi, search space, dan sampling juga perlu diperiksa. DockFlow sengaja menyediakan composite ranking dan guardrail agar score tidak berdiri sendiri. Bahkan

composite score tetap hanya alat prioritisasi internal, bukan bukti aktivitas atau afinitas eksperimen.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 15. Tuliskan sumber receptor dan ligan, metode preparasi, kebijakan water/hetero/chain, center dan size grid, backend Vina, scoring, exhaustiveness, num modes, energy range, seed, jumlah replikasi, threshold clustering, cutoff interaksi, metode redocking, versi perangkat lunak, serta cara ekspor/analisis. Sertakan file konfigurasi atau manifest bila tersedia.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 16. Visualisasi sangat baik untuk mendeteksi masalah besar seperti grid di luar pocket atau pose terbalik, tetapi tampilan dapat menyesatkan karena perspektif, skala, dan seleksi objek. Pemeriksaan numerik—koordinat, ukuran box, distance, RMSD, score, dan log—memberikan bukti yang lebih eksplisit. Keduanya harus dipakai secara komplementer.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 17. Mulailah dengan baseline yang telah lolos QC. Pilih satu parameter yang benar-benar terkait dengan bab, misalnya padding grid, exhaustiveness, threshold clustering, atau kebijakan preparasi. Jalankan minimal dua atau tiga tingkat parameter dengan input dan seed/replikasi yang terkontrol. Bandingkan pose, cluster, score, interaksi, dan validasi; jangan hanya membandingkan ranking akhir.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang

berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 18. Bandingkan checksum/hash input, versi engine, metode preparasi, konfigurasi lengkap, seed, receptor PDBQT, grid, dan output mentah. Pastikan tidak ada langkah manual yang tidak tercatat. Bila perbedaan tetap ada, laporkan lingkungan komputasi dan backend aktual. Reprodusibilitas adalah diagnosis berbasis bukti, bukan sekadar mencoba ulang perintah yang sama.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 19. Jika ligan referensi tersedia, redocking memberi uji awal apakah kombinasi receptor preparation, grid, sampling, dan pose mapping mampu mereproduksi struktur yang diketahui. Kegagalan pada tahap ini memberi sinyal untuk memperbaiki protokol sebelum biaya komputasi dan interpretasi diperluas ke banyak ligan.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 20. Laporan minimal memuat tujuan, input dan sumber struktur, parameter, status backend, QC, hasil redocking bila ada, tabel pose/cluster, analisis interaksi, ranking, keterbatasan, serta lampiran provenance dan raw output. Nyatakan dengan jelas mana data komputasi, mana interpretasi, dan mana fakta dari sumber eksperimen atau literatur.

Penerapan pada Bab 5: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

5.15 Soal Pengembangan

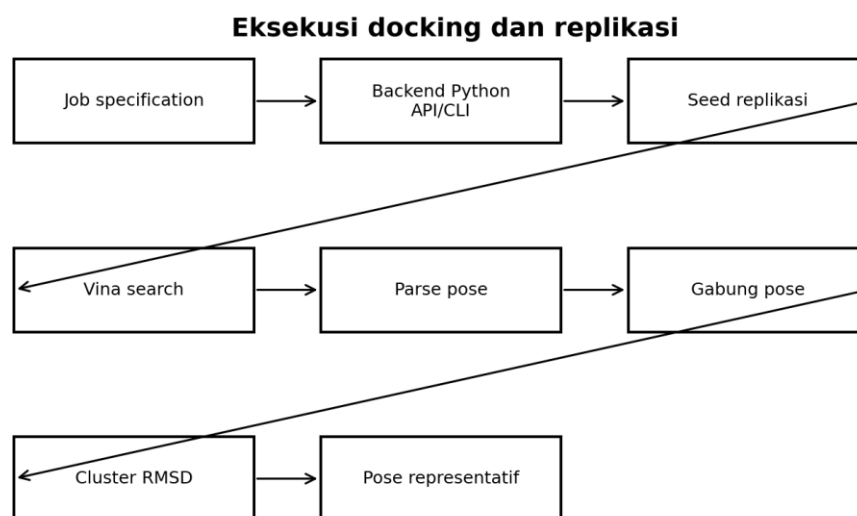
1. Rancang mini-proyek yang menguji satu keputusan metodologis utama Bab 5 dengan baseline dan minimal dua kondisi uji.
2. Susun rubrik penilaian yang memberi bobot pada QC, reproduksibilitas, validasi, interpretasi, dan dokumentasi.
3. Tulis satu contoh klaim yang terlalu kuat dari hasil docking, lalu perbaiki menjadi klaim yang sesuai kekuatan bukti.
4. Buat diagram data lineage dari input sampai satu keluaran akhir dan tunjukkan di mana provenance harus disimpan.
5. Susun daftar informasi yang harus diberikan kepada peneliti lain agar ia dapat mengulang run tanpa bertanya kembali.

BAB 6. AUTODOCK VINA, REPLIKASI, BATCH, DAN POSE CLUSTERING

Bab ini menjelaskan bagaimana DockFlow menjalankan Vina melalui Python API atau CLI, mengelola seed dan replikasi, mem-parsing pose, serta mengelompokkan pose berdasarkan RMSD untuk menilai konsistensi sampling.

6.1 Capaian Pembelajaran

- Menjelaskan konsep utama Bab 6 dengan istilah yang tepat dan membedakan data komputasi dari inferensi.
- Menghubungkan konsep dengan parameter, file, dan artefak yang benar-benar dihasilkan DockFlow.
- Menyusun quality-control dan kriteria keputusan sebelum menjalankan batch.
- Menganalisis kegagalan secara bertahap dan mendokumentasikan perubahan protokol.
- Menyusun laporan yang dapat diperiksa dan direproduksi.



Gambar 6.1. Skema utama pembahasan Bab 6: Autodock Vina, Replikasi, Batch, Dan Pose Clustering.

Tabel 6.1. Parameter utama Vina dalam DockFlow

Parameter	Fungsi
backend	Auto, Python API, atau CLI
scoring	Pilihan scoring backend
exhaustiveness	Intensitas pencarian
num_modes	Jumlah pose keluaran

Parameter	Fungsi
energy_range	Rentang energi pose keluaran
replicates	Jumlah replikasi seed
random_seed	Dasar reproduksibilitas sampling
timeout_sec	Batas waktu eksekusi

6.2 Scoring function Vina

Scoring function menilai kompatibilitas pose dalam model Vina dan terutama berguna untuk perbandingan internal protokol. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menyimpan score untuk setiap pose dan menggunakan best score sebagai salah satu fitur ranking. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengonversi score menjadi kd/ki eksperimen secara langsung. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

6.2.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait scoring function vina.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.

- Tandai secara eksplisit risiko: Mengonversi score menjadi Kd/Ki eksperimen secara langsung.
- Keputusan yang dianjurkan: Gunakan score bersama pose, cluster, interaksi, dan validasi.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Gunakan score bersama pose, cluster, interaksi, dan validasi. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Scoring function Vina” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “scoring function vina”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

6.3 Exhaustiveness

Exhaustiveness mengatur intensitas pencarian; nilai lebih besar biasanya menambah sampling dan waktu komputasi. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah

diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Jika dibaca sebagai bagian dari workflow, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Parameter disimpan dalam job specification sehingga studi sensitivitas dapat direproduksi. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengubah exhaustiveness antar-ligan dalam satu ranking. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

6.3.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait exhaustiveness.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengubah exhaustiveness antar-ligan dalam satu ranking.
- Keputusan yang dianjurkan: Gunakan parameter konsisten dan uji konvergensi pada subset.

Dari sisi quality control, keputusan yang disarankan adalah: Gunakan parameter konsisten dan uji konvergensi pada subset. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow

menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Exhaustiveness” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “exhaustiveness”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 6.2. Skenario keputusan untuk exhaustiveness

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Mengubah exhaustiveness antar-ligan dalam satu ranking.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Gunakan parameter konsisten dan uji konvergensi pada subset.

6.4 Number of modes dan energy range

Jumlah mode menentukan berapa pose yang diminta, sedangkan energy range membatasi rentang energi keluaran. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Untuk keperluan reproduksibilitas, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow meneruskan `num_modes` dan `energy_range` ke backend. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk

menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menafsirkan semua mode sebagai keadaan biologis independen. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

6.4.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait number of modes dan energy range.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menafsirkan semua mode sebagai keadaan biologis independen.
- Keputusan yang dianjurkan: Gunakan mode sebagai kandidat geometri untuk dianalisis dan dikelompokkan.

Dalam konteks pembelajaran, keputusan yang disarankan adalah: Gunakan mode sebagai kandidat geometri untuk dianalisis dan dikelompokkan. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Number of modes dan energy range” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai

ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “number of modes dan energy range”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

6.5 Random seed dan replikasi

Sampling stokastik memerlukan seed agar run dapat diulang dan variasi sampling dapat dinilai. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sisi quality control, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menjalankan replicates dengan seed yang terkontrol dan menyimpan metadata replikasi. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap satu seed cukup untuk menilai stabilitas pencarian. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

6.5.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait random seed dan replikasi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.

- Tandai secara eksplisit risiko: Menganggap satu seed cukup untuk menilai stabilitas pencarian.
- Keputusan yang dianjurkan: Gunakan replikasi sebagai pemeriksaan konsistensi, bukan pengganti data eksperimen.

Dari sudut pandang metodologi, keputusan yang disarankan adalah: Gunakan replikasi sebagai pemeriksaan konsistensi, bukan pengganti data eksperimen. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Random seed dan replikasi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “random seed dan replikasi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 6.3. Skenario keputusan untuk random seed dan replikasi

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menganggap satu seed cukup untuk menilai stabilitas pencarian.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.

Elemen	Uraian
Tindakan	Gunakan replikasi sebagai pemeriksaan konsistensi, bukan pengganti data eksperimen.

6.6 Python API versus CLI

Dua jalur backend memberi fleksibilitas deployment tetapi harus menghasilkan provenance yang jelas. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam konteks pembelajaran, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Mode Auto mencoba API, lalu CLI bila diperlukan. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah membandingkan hasil tanpa mencatat backend dan versi. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

6.6.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait python api versus cli.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Membandingkan hasil tanpa mencatat backend dan versi.
- Keputusan yang dianjurkan: Sertakan backend dalam laporan metode.

Pada tahap ini, keputusan yang disarankan adalah: Sertakan backend dalam laporan metode. Keputusan tersebut sebaiknya ditulis sebelum batch besar

dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Python API versus CLI” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “python api versus cli”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

6.7 Parsing pose PDBQT

Pose Vina dipisahkan dari blok MODEL/ENDMDL dan score dibaca dari REMARK VINA RESULT. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sudut pandang metodologi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow mengubah output menjadi struktur data pose, atom, score, dan PDBQT. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk

menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah parser gagal tetapi pengguna tetap memakai file parsial. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

6.7.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait parsing pose pdbqt.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Parser gagal tetapi pengguna tetap memakai file parsial.
- Keputusan yang dianjurkan: Verifikasi jumlah atom dan keberadaan score sebelum analisis.

Dalam praktik komputasi, keputusan yang disarankan adalah: Verifikasi jumlah atom dan keberadaan score sebelum analisis. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Parsing pose PDBQT” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai

ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “parsing pose pdbqt”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 6.4. Skenario keputusan untuk parsing pose pdbqt

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Parser gagal tetapi pengguna tetap memakai file parsial.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Verifikasi jumlah atom dan keberadaan score sebelum analisis.

6.8 RMSD antar-pose

RMSD memberikan ukuran jarak geometri untuk atom yang dipetakan secara konsisten. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Pada tahap ini, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memakai direct heavy-atom RMSD berbasis urutan atom. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menggunakan rmsd order-based saat atom order berbeda atau simetri penting. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT,

output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

6.8.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait rmsd antar-pose.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menggunakan RMSD order-based saat atom order berbeda atau simetri penting.
- Keputusan yang dianjurkan: Gunakan symmetry-aware mapping eksternal bila kasus memerlukannya.

Jika dibaca sebagai bagian dari workflow, keputusan yang disarankan adalah: Gunakan symmetry-aware mapping eksternal bila kasus memerlukannya. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “RMSD antar-pose” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “rmsd antar-pose”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

6.9 Pose clustering dan consensus

Clustering merangkum pose serupa dan ukuran cluster memberi indikator konsistensi sampling. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Default threshold 2,0 Å dapat mengelompokkan pose terhadap representative yang lebih baik score-nya. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menyebut cluster terbesar sebagai probabilitas binding. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

6.9.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait pose clustering dan consensus.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menyebut cluster terbesar sebagai probabilitas binding.
- Keputusan yang dianjurkan: Gunakan consensus sebagai fitur internal dan evaluasi sensitivitas threshold.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Gunakan consensus sebagai fitur internal dan evaluasi sensitivitas threshold. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini

mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Pose clustering dan consensus” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “pose clustering dan consensus”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 6.5. Skenario keputusan untuk pose clustering dan consensus

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menyebut cluster terbesar sebagai probabilitas binding.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Gunakan consensus sebagai fitur internal dan evaluasi sensitivitas threshold.

6.10 Rumus dan Indikator yang Relevan

DockFlow menggunakan direct heavy-atom RMSD ketika jumlah dan urutan atom sesuai. Rumus berikut menggambarkan bentuk umum ukuran yang dihitung dari selisih koordinat atom.

$$\text{RMSD} = \sqrt{[(1/N) \times \sum ||r_i - r_i(\text{ref})||^2]}$$

RMSD yang kecil menunjukkan kedekatan geometri pada mapping atom yang digunakan. Ia tidak secara otomatis menilai kesamaan kimia, simetri, atau kualitas afinitas.

6.11 Praktikum Terpandu

1. Buat folder proyek baru dan tulis tujuan eksperimen dalam satu paragraf sebelum menjalankan aplikasi.
2. Catat sumber receptor, ligan, dan semua parameter yang relevan dengan bab ini.
3. Jalankan QC input; bila ada warning, tuliskan keputusan menerima atau memperbaiki beserta alasannya.
4. Jalankan satu baseline terlebih dahulu. Simpan konfigurasi, log, raw output, dan ringkasan hasil.
5. Lakukan satu perubahan terkontrol pada parameter yang sedang dipelajari, lalu bandingkan keluaran dengan baseline.
6. Tulis kesimpulan yang memisahkan observasi komputasi, interpretasi, dan keterbatasan.

6.11.1 Analisis Praktikum 1

Analisis praktikum 1 pada Bab 6 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

6.11.2 Analisis Praktikum 2

Analisis praktikum 2 pada Bab 6 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu

parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

6.11.3 Analisis Praktikum 3

Analisis praktikum 3 pada Bab 6 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

6.12 Troubleshooting dan Kesalahan yang Sering Terjadi

Tabel 6.6. Troubleshooting ringkas Bab 6

Area	Gejala/Risiko	Tindakan
Scoring function Vina	Mengonversi score menjadi Kd/Ki eksperimen secara langsung.	Gunakan score bersama pose, cluster, interaksi, dan validasi.
Exhaustiveness	Mengubah exhaustiveness antarligan dalam satu ranking.	Gunakan parameter konsisten dan uji konvergensi pada subset.
Number of modes dan energy range	Menafsirkan semua mode sebagai keadaan biologis independen.	Gunakan mode sebagai kandidat geometri untuk dianalisis dan dikelompokkan.
Random seed dan replikasi	Menganggap satu seed cukup untuk menilai stabilitas pencarian.	Gunakan replikasi sebagai pemeriksaan konsistensi, bukan pengganti data eksperimen.

Area	Gejala/Risiko	Tindakan
Python API versus CLI	Membandingkan hasil tanpa mencatat backend dan versi.	Sertakan backend dalam laporan metode.
Parsing pose PDBQT	Parser gagal tetapi pengguna tetap memakai file parsial.	Verifikasi jumlah atom dan keberadaan score sebelum analisis.

Jika masalah berada pada area scoring function vina, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengonversi score menjadi kd/ki eksperimen secara langsung. Tindakan awal yang lebih kuat adalah Gunakan score bersama pose, cluster, interaksi, dan validasi. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area exhaustiveness, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengubah exhaustiveness antarligan dalam satu ranking. Tindakan awal yang lebih kuat adalah Gunakan parameter konsisten dan uji konvergensi pada subset. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area number of modes dan energy range, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menafsirkan semua mode sebagai keadaan biologis independen. Tindakan awal yang lebih kuat adalah Gunakan mode sebagai kandidat geometri untuk dianalisis dan dikelompokkan. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area random seed dan replikasi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap satu seed cukup untuk menilai stabilitas pencarian. Tindakan awal yang lebih kuat adalah Gunakan replikasi sebagai pemeriksaan konsistensi, bukan pengganti data eksperimen. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area python api versus cli, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah membandingkan hasil tanpa mencatat backend dan versi. Tindakan awal yang lebih kuat adalah Sertakan backend dalam laporan metode. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area parsing pose pdbqt, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah parser gagal tetapi pengguna tetap memakai file parsial. Tindakan awal yang lebih kuat adalah Verifikasi jumlah atom dan keberadaan score sebelum analisis. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

6.13 Latihan: 20 Soal

1. Jelaskan mengapa scoring function vina perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
2. Jelaskan mengapa exhaustiveness perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
3. Jelaskan mengapa number of modes dan energy range perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
4. Jelaskan mengapa random seed dan replikasi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
5. Jelaskan mengapa python api versus cli perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
6. Jelaskan mengapa parsing pose pdbqt perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
7. Jelaskan mengapa rmsd antar-pose perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
8. Jelaskan mengapa pose clustering dan consensus perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
9. Susun checklist minimum sebelum menjalankan batch pada Bab 6.
10. Sebuah hasil terlihat sangat baik pada satu indikator, tetapi gagal pada indikator QC lain. Bagaimana seharusnya hasil tersebut diperlakukan?
11. Mengapa perubahan satu parameter sebaiknya diuji secara terkontrol?

12. Jelaskan perbedaan antara data mentah, data turunan, dan interpretasi dalam workflow DockFlow.
13. Apa fungsi provenance ketika dua run menghasilkan hasil berbeda?
14. Kritik pernyataan: “Ligan dengan docking score paling rendah pasti merupakan kandidat terbaik.”
15. Bagaimana Anda menulis bagian metode agar run dapat direproduksi?
16. Mengapa visualisasi tidak boleh menggantikan pemeriksaan numerik?
17. Rancang satu eksperimen sensitivitas yang relevan dengan bab ini.
18. Apa yang harus dilakukan bila hasil tidak dapat direproduksi oleh kelompok lain?
19. Mengapa buku ini menekankan redocking sebelum screening besar?
20. Buat format ringkas laporan hasil yang baik.

6.14 Pembahasan 20 Soal

Pembahasan 1. Scoring function menilai kompatibilitas pose dalam model Vina dan terutama berguna untuk perbandingan internal protokol. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menyimpan score untuk setiap pose dan menggunakan best score sebagai salah satu fitur ranking. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengonversi score menjadi kd/ki eksperimen secara langsung. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan score bersama pose, cluster, interaksi, dan validasi.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 2. Exhaustiveness mengatur intensitas pencarian; nilai lebih besar biasanya menambah sampling dan waktu komputasi. Pada DockFlow, aspek ini diwujudkan melalui: Parameter disimpan dalam job specification sehingga studi sensitivitas dapat direproduksi. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengubah exhaustiveness antar-ligan dalam satu ranking. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan

alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan parameter konsisten dan uji konvergensi pada subset.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 3. Jumlah mode menentukan berapa pose yang diminta, sedangkan energy range membatasi rentang energi keluaran. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow meneruskan `num_modes` dan `energy_range` ke backend. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menafsirkan semua mode sebagai keadaan biologis independen. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan mode sebagai kandidat geometri untuk dianalisis dan dikelompokkan.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 4. Sampling stokastik memerlukan seed agar run dapat diulang dan variasi sampling dapat dinilai. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menjalankan replicates dengan seed yang terkontrol dan menyimpan metadata replikasi. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap satu seed cukup untuk menilai stabilitas pencarian. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan replikasi sebagai pemeriksaan konsistensi, bukan pengganti data eksperimen.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 5. Dua jalur backend memberi fleksibilitas deployment tetapi harus menghasilkan provenance yang jelas. Pada DockFlow, aspek ini diwujudkan melalui: Mode Auto mencoba API, lalu CLI bila diperlukan. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah membandingkan hasil tanpa mencatat backend dan versi. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Sertakan backend dalam laporan metode.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 6. Pose Vina dipisahkan dari blok MODEL/ENDMDL dan score dibaca dari REMARK VINA RESULT. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow mengubah output menjadi struktur data pose, atom, score, dan PDBQT. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah parser gagal tetapi pengguna tetap memakai file parsial. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Verifikasi jumlah atom dan keberadaan score sebelum analisis.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 7. RMSD memberikan ukuran jarak geometri untuk atom yang dipetakan secara konsisten. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memakai direct heavy-atom RMSD berbasis urutan atom. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menggunakan rmsd order-based saat atom order berbeda atau simetri penting. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan symmetry-aware mapping eksternal bila kasus memerlukannya.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 8. Clustering merangkum pose serupa dan ukuran cluster memberi indikator konsistensi sampling. Pada DockFlow, aspek ini diwujudkan melalui: Default threshold 2,0 Å dapat mengelompokkan pose terhadap representative yang lebih baik score-nya. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menyebut cluster terbesar sebagai probabilitas binding. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan consensus sebagai fitur internal dan evaluasi sensitivitas threshold.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 9. Checklist minimum harus memeriksa identitas input, kelengkapan struktur, parameter yang relevan dengan bab ini, ketersediaan backend, serta lokasi artefak yang akan disimpan. Tambahkan kriteria gagal/lanjut yang jelas. Prinsipnya: jangan memulai batch bila masalah dasar sudah terlihat pada QC. Setelah run, simpan konfigurasi, log, dan keluaran mentah agar keputusan dapat diaudit.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika

terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 10. Indikator yang baik tidak menghapus kegagalan QC. Hasil perlu ditandai bermasalah, penyebabnya ditelusuri pada tahap input/preparasi/eksekusi, dan bila perlu run diulang dengan protokol yang diperbaiki. Dalam DockFlow, ranking bukan pengganti QC; hasil invalid bahkan mendapat penalty pada composite score. Laporan harus menyebutkan kegagalan dan tidak menjadikan satu angka sebagai pembenaran.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 11. Perubahan satu faktor memudahkan atribusi sebab-akibat. Bila beberapa parameter diubah sekaligus, pengguna tidak dapat mengetahui faktor mana yang menyebabkan perubahan pose, score, atau interaksi. Praktik yang baik adalah membuat baseline, mengubah satu parameter, menjaga input lain identik, membandingkan output, lalu mencatat keputusan. Pendekatan ini juga mempermudah reproduksi oleh kelompok lain.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 12. Data mentah meliputi struktur sumber dan output engine; data turunan meliputi pose terpilih, cluster, fingerprint, frekuensi residu, atau ranking; interpretasi adalah argumen ilmiah yang dibangun dari data tersebut. Ketiganya harus dipisahkan agar pembaca dapat menilai apakah transformasi data masuk akal dan apakah kesimpulan lebih kuat daripada bukti yang tersedia.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 13. Provenance menyediakan rekaman input, versi, seed, parameter, backend, waktu, dan artefak yang memungkinkan perbedaan dilacak. Tanpa

provenance, perbedaan hasil mudah dianggap sebagai perilaku molekul padahal mungkin berasal dari perubahan preparasi atau konfigurasi. Dengan provenance, pengguna dapat menyusun perbandingan terkontrol dan mengulang kondisi secara lebih tepat.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 14. Pernyataan itu terlalu kuat. Docking score adalah salah satu keluaran model; kualitas pose, cluster, interaksi, redocking, protonasi, search space, dan sampling juga perlu diperiksa. DockFlow sengaja menyediakan composite ranking dan guardrail agar score tidak berdiri sendiri. Bahkan composite score tetap hanya alat prioritas internal, bukan bukti aktivitas atau afinitas eksperimen.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 15. Tuliskan sumber receptor dan ligan, metode preparasi, kebijakan water/hetero/chain, center dan size grid, backend Vina, scoring, exhaustiveness, num modes, energy range, seed, jumlah replikasi, threshold clustering, cutoff interaksi, metode redocking, versi perangkat lunak, serta cara ekspor/analisis. Sertakan file konfigurasi atau manifest bila tersedia.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 16. Visualisasi sangat baik untuk mendeteksi masalah besar seperti grid di luar pocket atau pose terbalik, tetapi tampilan dapat menyesatkan karena perspektif, skala, dan seleksi objek. Pemeriksaan numerik—koordinat, ukuran box, distance, RMSD, score, dan log—memberikan bukti yang lebih eksplisit. Keduanya harus dipakai secara komplementer.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 17. Mulailah dengan baseline yang telah lolos QC. Pilih satu parameter yang benar-benar terkait dengan bab, misalnya padding grid, exhaustiveness, threshold clustering, atau kebijakan preparasi. Jalankan minimal dua atau tiga tingkat parameter dengan input dan seed/replikasi yang terkontrol. Bandingkan pose, cluster, score, interaksi, dan validasi; jangan hanya membandingkan ranking akhir.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 18. Bandingkan checksum/hash input, versi engine, metode preparasi, konfigurasi lengkap, seed, receptor PDBQT, grid, dan output mentah. Pastikan tidak ada langkah manual yang tidak tercatat. Bila perbedaan tetap ada, laporkan lingkungan komputasi dan backend aktual. Reprodusibilitas adalah diagnosis berbasis bukti, bukan sekadar mencoba ulang perintah yang sama.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 19. Jika ligan referensi tersedia, redocking memberi uji awal apakah kombinasi receptor preparation, grid, sampling, dan pose mapping mampu mereproduksi struktur yang diketahui. Kegagalan pada tahap ini memberi sinyal untuk memperbaiki protokol sebelum biaya komputasi dan interpretasi diperluas ke banyak ligan.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 20. Laporan minimal memuat tujuan, input dan sumber struktur, parameter, status backend, QC, hasil redocking bila ada, tabel pose/cluster, analisis interaksi, ranking, keterbatasan, serta lampiran provenance dan raw output. Nyatakan dengan jelas mana data komputasi, mana interpretasi, dan mana fakta dari sumber eksperimen atau literatur.

Penerapan pada Bab 6: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

6.15 Soal Pengembangan

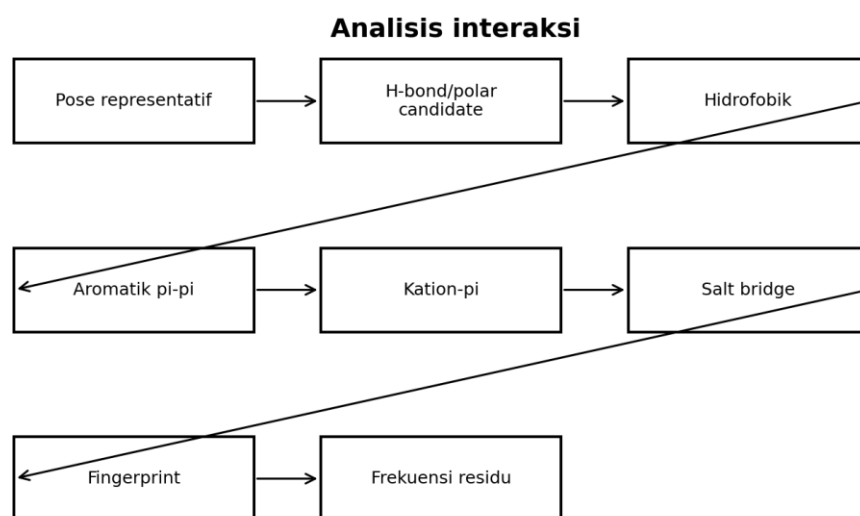
1. Rancang mini-proyek yang menguji satu keputusan metodologis utama Bab 6 dengan baseline dan minimal dua kondisi uji.
2. Susun rubrik penilaian yang memberi bobot pada QC, reproduksibilitas, validasi, interpretasi, dan dokumentasi.
3. Tulis satu contoh klaim yang terlalu kuat dari hasil docking, lalu perbaiki menjadi klaim yang sesuai kekuatan bukti.
4. Buat diagram data lineage dari input sampai satu keluaran akhir dan tunjukkan di mana provenance harus disimpan.
5. Susun daftar informasi yang harus diberikan kepada peneliti lain agar ia dapat mengulang run tanpa bertanya kembali.

BAB 7. ANALISIS INTERAKSI LIGAN–PROTEIN DAN INTERACTION FINGERPRINT

Setelah docking, interpretasi beralih dari skor ke geometri kontak. Bab ini menguraikan cara DockFlow menyaring kontak polar, hidrofobik, aromatik, kation- π , salt bridge, serta membangun fingerprint dan frekuensi residu.

7.1 Capaian Pembelajaran

- Menjelaskan konsep utama Bab 7 dengan istilah yang tepat dan membedakan data komputasi dari inferensi.
- Menghubungkan konsep dengan parameter, file, dan artefak yang benar-benar dihasilkan DockFlow.
- Menyusun quality-control dan kriteria keputusan sebelum menjalankan batch.
- Menganalisis kegagalan secara bertahap dan mendokumentasikan perubahan protokol.
- Menyusun laporan yang dapat diperiksa dan direproduksi.



Gambar 7.1. Skema utama pembahasan Bab 7: Analisis Interaksi Ligan–Protein Dan Interaction Fingerprint.

Tabel 7.1. Interaksi dan cutoff default

Interaksi	Cutoff/pendekatan	Catatan
H-bond/polar candidate	3,5 Å	Screen jarak; angle tidak dibuktikan
Hidrofobik	4,5 Å	Kontak geometrik
Salt bridge	4,0 Å	Bergantung assignment charge
π - π	5,5 Å	Memakai geometri cincin

Interaksi	Cutoff/pendekatan	Catatan
Kation- π	6,0 Å	Bergantung pusat kation dan cincin

7.2 Kontak polar dan kandidat H-bond

Jarak antar-atom polar dapat menyaring kemungkinan ikatan hidrogen, tetapi geometri sudut donor-H-akseptor tetap penting. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memakai cutoff default 3,5 Å dan memberi label H-bond/polar candidate. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menyebut semua kontak n/o/s dalam cutoff sebagai h-bond pasti. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

7.2.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait kontak polar dan kandidat h-bond.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menyebut semua kontak N/O/S dalam cutoff sebagai H-bond pasti.
- Keputusan yang dianjurkan: Laporkan sebagai kandidat dan verifikasi geometri lengkap bila tersedia.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Laporkan sebagai kandidat dan verifikasi geometri lengkap bila tersedia. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Kontak polar dan kandidat H-bond” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “kontak polar dan kandidat h-bond”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

7.3 Kontak hidrofobik

Kontak atom nonpolar membantu menggambarkan packing ligan di pocket. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Jika dibaca sebagai bagian dari workflow, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memakai cutoff default 4,5 Å pada atom hidrofobik yang teridentifikasi. Setiap

operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menghitung kontak sebagai ukuran energi tanpa model tambahan. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

7.3.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait kontak hidrofobik.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menghitung kontak sebagai ukuran energi tanpa model tambahan.
- Keputusan yang dianjurkan: Gunakan count sebagai deskriptor geometri, bukan energi bebas.

Dari sisi quality control, keputusan yang disarankan adalah: Gunakan count sebagai deskriptor geometri, bukan energi bebas. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Kontak hidrofobik” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung

tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “kontak hidrofobik”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 7.2. Skenario keputusan untuk kontak hidrofobik

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menghitung kontak sebagai ukuran energi tanpa model tambahan.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Gunakan count sebagai deskriptor geometri, bukan energi bebas.

7.4 Interaksi aromatik π – π

Interaksi aromatik bergantung pada jarak centroid dan orientasi bidang cincin. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Untuk keperluan reproduksibilitas, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memperkirakan ring geometry dan membandingkan cincin aromatik ligan dengan residu aromatik. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menggunakan satu jarak atom sebagai bukti π – π stacking. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh

sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

7.4.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait interaksi aromatik π - π .
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menggunakan satu jarak atom sebagai bukti π - π stacking.
- Keputusan yang dianjurkan: Baca centroid, normal, dan orientasi bersama konteks residu.

Dalam konteks pembelajaran, keputusan yang disarankan adalah: Baca centroid, normal, dan orientasi bersama konteks residu. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Interaksi aromatik π - π ” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “interaksi aromatik π - π ”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah

QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

7.5 Kation- π

Kation- π melibatkan pusat bermuatan positif dan sistem aromatik. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sisi quality control, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menyaring geometri dengan cutoff yang disimpan dalam konfigurasi. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengabaikan protonation state pusat kationik. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

7.5.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait kation- π .
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengabaikan protonation state pusat kationik.
- Keputusan yang dianjurkan: Verifikasi keadaan muatan sebelum menafsirkan interaksi.

Dari sudut pandang metodologi, keputusan yang disarankan adalah: Verifikasi keadaan muatan sebelum menafsirkan interaksi. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan

parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Kation- π ” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “kation- π ”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 7.3. Skenario keputusan untuk kation- π

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Mengabaikan protonation state pusat kationik.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Verifikasi keadaan muatan sebelum menafsirkan interaksi.

7.6 Salt bridge

Salt bridge memerlukan pasangan muatan berlawanan dalam geometri yang sesuai. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa

pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam konteks pembelajaran, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memakai cutoff default 4,0 Å pada pasangan yang teridentifikasi. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengklaim salt bridge saat assignment charge tidak tepat. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

7.6.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait salt bridge.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengklaim salt bridge saat assignment charge tidak tepat.
- Keputusan yang dianjurkan: Pastikan representasi charge dan protonasi konsisten.

Pada tahap ini, keputusan yang disarankan adalah: Pastikan representasi charge dan protonasi konsisten. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Salt bridge” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “salt bridge”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

7.7 Interaction fingerprint

Fingerprint mengubah daftar kontak menjadi fitur vektor untuk perbandingan pose dan analisis data. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sudut pandang metodologi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menghasilkan baris fingerprint per ligan dan pose untuk ekspor CSV/XLSX. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap fingerprint sebagai mekanisme biologis final. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

7.7.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait interaction fingerprint.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap fingerprint sebagai mekanisme biologis final.
- Keputusan yang dianjurkan: Gunakan sebagai representasi komparatif dan fitur downstream.

Dalam praktik komputasi, keputusan yang disarankan adalah: Gunakan sebagai representasi komparatif dan fitur downstream. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Interaction fingerprint” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “interaction fingerprint”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 7.4. Skenario keputusan untuk interaction fingerprint

Elemen	Uraian
--------	--------

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menganggap fingerprint sebagai mekanisme biologis final.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Gunakan sebagai representasi komparatif dan fitur downstream.

7.8 Frekuensi residu

Frekuensi menunjukkan seberapa sering residu muncul pada pose yang dianalisis. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Pada tahap ini, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menghitung Pose count dan Frequency menurut tipe interaksi. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah membandingkan frekuensi dari jumlah pose berbeda tanpa normalisasi. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

7.8.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait frekuensi residu.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Membandingkan frekuensi dari jumlah pose berbeda tanpa normalisasi.
- Keputusan yang dianjurkan: Perhatikan denominator dan kebijakan top poses.

Jika dibaca sebagai bagian dari workflow, keputusan yang disarankan adalah: Perhatikan denominator dan kebijakan top poses. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Frekuensi residu” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “frekuensi residu”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

7.9 Key residues

Residu kunci dapat berasal dari struktur, mutagenesis, atau literatur dan berguna untuk prioritisasi. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menghitung `key_residue_hits` dan memasukkannya ke composite ranking. Setiap operasi menghasilkan artefak

atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah memilih residu setelah melihat hasil hanya untuk menaikkan kandidat tertentu. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

7.9.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait key residues.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Memilih residu setelah melihat hasil hanya untuk menaikkan kandidat tertentu.
- Keputusan yang dianjurkan: Tentukan key residues secara a priori bila digunakan dalam ranking.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Tentukan key residues secara a priori bila digunakan dalam ranking. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Key residues” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal.

Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “key residues”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 7.5. Skenario keputusan untuk key residues

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Memilih residu setelah melihat hasil hanya untuk menaikkan kandidat tertentu.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Tentukan key residues secara a priori bila digunakan dalam ranking.

7.10 Rumus dan Indikator yang Relevan

Keputusan ilmiah = bukti input + QC + parameter + hasil + validasi + keterbatasan

Formula konseptual tersebut mengingatkan bahwa satu indikator tidak cukup untuk menopang kesimpulan workflow. Setiap bagian menyediakan jenis bukti yang berbeda dan saling melengkapi.

7.11 Praktikum Terpandu

1. Buat folder proyek baru dan tulis tujuan eksperimen dalam satu paragraf sebelum menjalankan aplikasi.
2. Catat sumber receptor, ligan, dan semua parameter yang relevan dengan bab ini.
3. Jalankan QC input; bila ada warning, tuliskan keputusan menerima atau memperbaiki beserta alasannya.
4. Jalankan satu baseline terlebih dahulu. Simpan konfigurasi, log, raw output, dan ringkasan hasil.

5. Lakukan satu perubahan terkontrol pada parameter yang sedang dipelajari, lalu bandingkan keluaran dengan baseline.
6. Tulis kesimpulan yang memisahkan observasi komputasi, interpretasi, dan keterbatasan.

7.11.1 Analisis Praktikum 1

Analisis praktikum 1 pada Bab 7 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

7.11.2 Analisis Praktikum 2

Analisis praktikum 2 pada Bab 7 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

7.11.3 Analisis Praktikum 3

Analisis praktikum 3 pada Bab 7 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila

tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

7.12 Troubleshooting dan Kesalahan yang Sering Terjadi

Tabel 7.6. Troubleshooting ringkas Bab 7

Area	Gejala/Risiko	Tindakan
Kontak polar dan kandidat H-bond	Menyebut semua kontak N/O/S dalam cutoff sebagai H-bond pasti.	Laporkan sebagai kandidat dan verifikasi geometri lengkap bila tersedia.
Kontak hidrofobik	Menghitung kontak sebagai ukuran energi tanpa model tambahan.	Gunakan count sebagai deskriptor geometri, bukan energi bebas.
Interaksi aromatik π - π	Menggunakan satu jarak atom sebagai bukti π - π stacking.	Baca centroid, normal, dan orientasi bersama konteks residu.
Kation- π	Mengabaikan protonation state pusat kationik.	Verifikasi keadaan muatan sebelum menafsirkan interaksi.
Salt bridge	Mengklaim salt bridge saat assignment charge tidak tepat.	Pastikan representasi charge dan protonasi konsisten.
Interaction fingerprint	Menganggap fingerprint sebagai mekanisme biologis final.	Gunakan sebagai representasi komparatif dan fitur downstream.

Jika masalah berada pada area kontak polar dan kandidat h-bond, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menyebut semua kontak n/o/s dalam cutoff sebagai h-bond pasti. Tindakan awal yang lebih kuat adalah Laporkan sebagai kandidat dan verifikasi geometri lengkap bila tersedia. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area kontak hidrofobik, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menghitung kontak sebagai ukuran energi tanpa model tambahan. Tindakan awal yang lebih kuat adalah Gunakan count sebagai deskriptor geometri, bukan energi bebas. Setelah

perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area interaksi aromatik π - π , jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menggunakan satu jarak atom sebagai bukti π - π stacking. Tindakan awal yang lebih kuat adalah Baca centroid, normal, dan orientasi bersama konteks residu. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area kation- π , jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengabaikan protonation state pusat kationik. Tindakan awal yang lebih kuat adalah Verifikasi keadaan muatan sebelum menafsirkan interaksi. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area salt bridge, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengklaim salt bridge saat assignment charge tidak tepat. Tindakan awal yang lebih kuat adalah Pastikan representasi charge dan protonasi konsisten. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area interaction fingerprint, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap fingerprint sebagai mekanisme biologis final. Tindakan awal yang lebih kuat adalah Gunakan sebagai representasi komparatif dan fitur downstream. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

7.13 Latihan: 20 Soal

1. Jelaskan mengapa kontak polar dan kandidat h-bond perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
2. Jelaskan mengapa kontak hidrofobik perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
3. Jelaskan mengapa interaksi aromatik π - π perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.

4. Jelaskan mengapa kation- π perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
5. Jelaskan mengapa salt bridge perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
6. Jelaskan mengapa interaction fingerprint perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
7. Jelaskan mengapa frekuensi residu perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
8. Jelaskan mengapa key residues perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
9. Susun checklist minimum sebelum menjalankan batch pada Bab 7.
10. Sebuah hasil terlihat sangat baik pada satu indikator, tetapi gagal pada indikator QC lain. Bagaimana seharusnya hasil tersebut diperlakukan?
11. Mengapa perubahan satu parameter sebaiknya diuji secara terkontrol?
12. Jelaskan perbedaan antara data mentah, data turunan, dan interpretasi dalam workflow DockFlow.
13. Apa fungsi provenance ketika dua run menghasilkan hasil berbeda?
14. Kritik pernyataan: “Ligan dengan docking score paling rendah pasti merupakan kandidat terbaik.”
15. Bagaimana Anda menulis bagian metode agar run dapat direproduksi?
16. Mengapa visualisasi tidak boleh menggantikan pemeriksaan numerik?
17. Rancang satu eksperimen sensitivitas yang relevan dengan bab ini.
18. Apa yang harus dilakukan bila hasil tidak dapat direproduksi oleh kelompok lain?
19. Mengapa buku ini menekankan redocking sebelum screening besar?
20. Buat format ringkas laporan hasil yang baik.

7.14 Pembahasan 20 Soal

Pembahasan 1. Jarak antar-atom polar dapat menyaring kemungkinan ikatan hidrogen, tetapi geometri sudut donor-H-akseptor tetap penting. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memakai cutoff default 3,5 Å dan memberi label H-bond/polar candidate. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menyebut semua kontak n/o/s dalam cutoff sebagai h-bond pasti. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Laporkan sebagai kandidat dan verifikasi geometri lengkap bila tersedia.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 2. Kontak atom nonpolar membantu menggambarkan packing ligan di pocket. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memakai cutoff default 4,5 Å pada atom hidrofobik yang teridentifikasi. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menghitung kontak sebagai ukuran energi tanpa model tambahan. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan count sebagai deskriptor geometri, bukan energi bebas.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 3. Interaksi aromatik bergantung pada jarak centroid dan orientasi bidang cincin. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memperkirakan ring geometry dan membandingkan cincin aromatik ligan dengan residu aromatik. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menggunakan satu jarak atom sebagai bukti π - π stacking. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa

keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Baca centroid, normal, dan orientasi bersama konteks residu.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 4. Kation- π melibatkan pusat bermuatan positif dan sistem aromatik. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menyaring geometri dengan cutoff yang disimpan dalam konfigurasi. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengabaikan protonation state pusat kationik. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Verifikasi keadaan muatan sebelum menafsirkan interaksi.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 5. Salt bridge memerlukan pasangan muatan berlawanan dalam geometri yang sesuai. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memakai cutoff default 4,0 Å pada pasangan yang teridentifikasi. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengklaim salt bridge saat assignment charge tidak tepat. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Pastikan representasi charge dan protonasi konsisten.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 6. Fingerprint mengubah daftar kontak menjadi fitur vektor untuk perbandingan pose dan analisis data. Pada DockFlow, aspek ini diwujudkan

melalui: DockFlow menghasilkan baris fingerprint per ligan dan pose untuk ekspor CSV/XLSX. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap fingerprint sebagai mekanisme biologis final. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan sebagai representasi komparatif dan fitur downstream.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 7. Frekuensi menunjukkan seberapa sering residu muncul pada pose yang dianalisis. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menghitung Pose count dan Frequency menurut tipe interaksi. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah membandingkan frekuensi dari jumlah pose berbeda tanpa normalisasi. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Perhatikan denominator dan kebijakan top poses.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 8. Residu kunci dapat berasal dari struktur, mutagenesis, atau literatur dan berguna untuk prioritasasi. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menghitung key_residue_hits dan memasukkannya ke composite ranking. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah memilih residu setelah melihat hasil hanya untuk menaikkan kandidat tertentu. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Tentukan key residues secara a priori bila digunakan dalam ranking.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file

atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 9. Checklist minimum harus memeriksa identitas input, kelengkapan struktur, parameter yang relevan dengan bab ini, ketersediaan backend, serta lokasi artefak yang akan disimpan. Tambahkan kriteria gagal/lanjut yang jelas. Prinsipnya: jangan memulai batch bila masalah dasar sudah terlihat pada QC. Setelah run, simpan konfigurasi, log, dan keluaran mentah agar keputusan dapat diaudit.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 10. Indikator yang baik tidak menghapus kegagalan QC. Hasil perlu ditandai bermasalah, penyebabnya ditelusuri pada tahap input/preparasi/eksekusi, dan bila perlu run diulang dengan protokol yang diperbaiki. Dalam DockFlow, ranking bukan pengganti QC; hasil invalid bahkan mendapat penalty pada composite score. Laporan harus menyebutkan kegagalan dan tidak menjadikan satu angka sebagai pembenaran.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 11. Perubahan satu faktor memudahkan atribusi sebab-akibat. Bila beberapa parameter diubah sekaligus, pengguna tidak dapat mengetahui faktor mana yang menyebabkan perubahan pose, score, atau interaksi. Praktik yang baik adalah membuat baseline, mengubah satu parameter, menjaga input lain identik, membandingkan output, lalu mencatat keputusan. Pendekatan ini juga mempermudah reproduksi oleh kelompok lain.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika

terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 12. Data mentah meliputi struktur sumber dan output engine; data turunan meliputi pose terpilih, cluster, fingerprint, frekuensi residu, atau ranking; interpretasi adalah argumen ilmiah yang dibangun dari data tersebut. Ketiganya harus dipisahkan agar pembaca dapat menilai apakah transformasi data masuk akal dan apakah kesimpulan lebih kuat daripada bukti yang tersedia.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 13. Provenance menyediakan rekaman input, versi, seed, parameter, backend, waktu, dan artefak yang memungkinkan perbedaan dilacak. Tanpa provenance, perbedaan hasil mudah dianggap sebagai perilaku molekul padahal mungkin berasal dari perubahan preparasi atau konfigurasi. Dengan provenance, pengguna dapat menyusun perbandingan terkontrol dan mengulang kondisi secara lebih tepat.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 14. Pernyataan itu terlalu kuat. Docking score adalah salah satu keluaran model; kualitas pose, cluster, interaksi, redocking, protonasi, search space, dan sampling juga perlu diperiksa. DockFlow sengaja menyediakan composite ranking dan guardrail agar score tidak berdiri sendiri. Bahkan composite score tetap hanya alat prioritisasi internal, bukan bukti aktivitas atau afinitas eksperimen.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 15. Tuliskan sumber receptor dan ligan, metode preparasi, kebijakan water/hetero/chain, center dan size grid, backend Vina, scoring,

exhaustiveness, num modes, energy range, seed, jumlah replikasi, threshold clustering, cutoff interaksi, metode redocking, versi perangkat lunak, serta cara ekspor/analisis. Sertakan file konfigurasi atau manifest bila tersedia.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 16. Visualisasi sangat baik untuk mendeteksi masalah besar seperti grid di luar pocket atau pose terbalik, tetapi tampilan dapat menyesatkan karena perspektif, skala, dan seleksi objek. Pemeriksaan numerik—koordinat, ukuran box, distance, RMSD, score, dan log—memberikan bukti yang lebih eksplisit. Keduanya harus dipakai secara komplementer.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 17. Mulailah dengan baseline yang telah lolos QC. Pilih satu parameter yang benar-benar terkait dengan bab, misalnya padding grid, exhaustiveness, threshold clustering, atau kebijakan preparasi. Jalankan minimal dua atau tiga tingkat parameter dengan input dan seed/replikasi yang terkontrol. Bandingkan pose, cluster, score, interaksi, dan validasi; jangan hanya membandingkan ranking akhir.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 18. Bandingkan checksum/hash input, versi engine, metode preparasi, konfigurasi lengkap, seed, receptor PDBQT, grid, dan output mentah. Pastikan tidak ada langkah manual yang tidak tercatat. Bila perbedaan tetap ada, laporkan lingkungan komputasi dan backend aktual. Reprodusibilitas adalah diagnosis berbasis bukti, bukan sekadar mencoba ulang perintah yang sama.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file

atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 19. Jika ligan referensi tersedia, redocking memberi uji awal apakah kombinasi receptor preparation, grid, sampling, dan pose mapping mampu mereproduksi struktur yang diketahui. Kegagalan pada tahap ini memberi sinyal untuk memperbaiki protokol sebelum biaya komputasi dan interpretasi diperluas ke banyak ligan.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 20. Laporan minimal memuat tujuan, input dan sumber struktur, parameter, status backend, QC, hasil redocking bila ada, tabel pose/cluster, analisis interaksi, ranking, keterbatasan, serta lampiran provenance dan raw output. Nyatakan dengan jelas mana data komputasi, mana interpretasi, dan mana fakta dari sumber eksperimen atau literatur.

Penerapan pada Bab 7: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

7.15 Soal Pengembangan

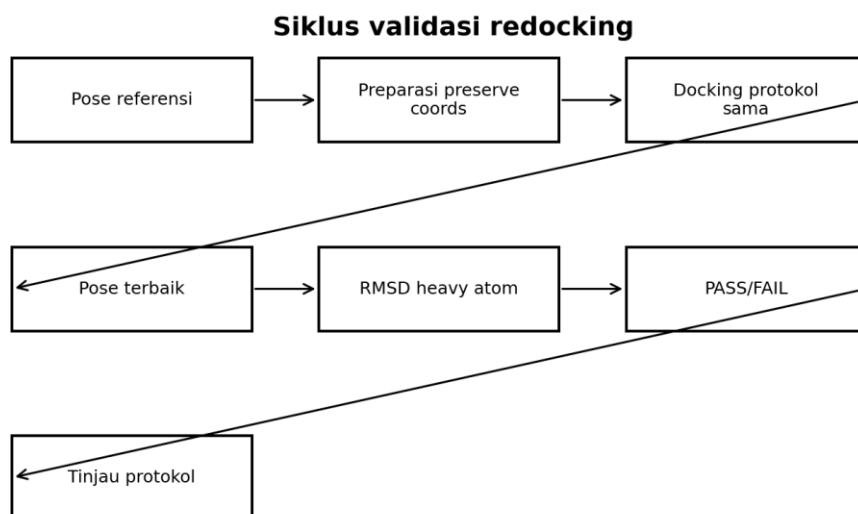
1. Rancang mini-proyek yang menguji satu keputusan metodologis utama Bab 7 dengan baseline dan minimal dua kondisi uji.
2. Susun rubrik penilaian yang memberi bobot pada QC, reproduksibilitas, validasi, interpretasi, dan dokumentasi.
3. Tulis satu contoh klaim yang terlalu kuat dari hasil docking, lalu perbaiki menjadi klaim yang sesuai kekuatan bukti.
4. Buat diagram data lineage dari input sampai satu keluaran akhir dan tunjukkan di mana provenance harus disimpan.
5. Susun daftar informasi yang harus diberikan kepada peneliti lain agar ia dapat mengulang run tanpa bertanya kembali.

BAB 8. VALIDASI REDOCKING, RMSD, DAN REPRODUKSIBILITAS PROTOKOL

Redocking menguji apakah protokol mampu mereproduksi pose referensi dalam kondisi yang ditetapkan. Bab ini menekankan pemilihan referensi, preserve coordinates, threshold RMSD, keterbatasan atom mapping, serta keputusan PASS/FAIL.

8.1 Capaian Pembelajaran

- Menjelaskan konsep utama Bab 8 dengan istilah yang tepat dan membedakan data komputasi dari inferensi.
- Menghubungkan konsep dengan parameter, file, dan artefak yang benar-benar dihasilkan DockFlow.
- Menyusun quality-control dan kriteria keputusan sebelum menjalankan batch.
- Menganalisis kegagalan secara bertahap dan mendokumentasikan perubahan protokol.
- Menyusun laporan yang dapat diperiksa dan direproduksi.



Gambar 8.1. Skema utama pembahasan Bab 8: Validasi Redocking, Rmsd, Dan Reprodusibilitas Protokol.

Tabel 8.1. Checklist sebelum menyatakan redocking PASS

Butir	Pertanyaan
Referensi	Apakah pose referensi berasal dari frame receptor yang sama?
Koordinat	Apakah koordinat referensi dipertahankan?
Grid	Apakah pocket dan ruang orientasi tercakup?

Butir	Pertanyaan
Parameter	Apakah seed/exhaustiveness/modes terdokumentasi?
RMSD	Apakah metode mapping sesuai untuk molekul?
Threshold	Apakah kriteria ditentukan sebelum melihat hasil?

8.2 Tujuan redocking

Redocking adalah quality control pose prediction terhadap ligan dengan pose eksperimen yang diketahui. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menyiapkan ligan referensi dengan `preserve_coords` lalu menjalankan protokol docking. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap redocking sebagai validasi afinitas. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

8.2.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait tujuan redocking.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap redocking sebagai validasi afinitas.

- Keputusan yang dianjurkan: Gunakan redocking untuk kemampuan reproduksi pose, bukan pembuktian energi binding.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Gunakan redocking untuk kemampuan reproduksi pose, bukan pembuktian energi binding. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Tujuan redocking” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “tujuan redocking”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

8.3 Pose referensi

Referensi harus berasal dari koordinat yang relevan dengan receptor yang sama atau frame yang konsisten. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Jika dibaca sebagai bagian dari workflow, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow dapat memakai ligan referensi dari PDBQT atau koordinat molekul yang dipertahankan. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menggunakan geometri referensi yang telah dioptimasi bebas sehingga bukan pose kristalografi asli. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

8.3.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait pose referensi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menggunakan geometri referensi yang telah dioptimasi bebas sehingga bukan pose kristalografi asli.
- Keputusan yang dianjurkan: Simpan pose asli dan metode ekstraksi.

Dari sisi quality control, keputusan yang disarankan adalah: Simpan pose asli dan metode ekstraksi. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Pose referensi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah

interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “pose referensi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 8.2. Skenario keputusan untuk pose referensi

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menggunakan geometri referensi yang telah dioptimasi bebas sehingga bukan pose kristalografi asli.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Simpan pose asli dan metode ekstraksi.

8.4 Threshold RMSD

Threshold adalah kriteria keputusan yang harus dinyatakan sebelum screening. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Untuk keperluan reproduksibilitas, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Default konfigurasi redocking_rmsd_threshold adalah 2,0 Å dan status PASS diberikan bila $\text{RMSD} \leq \text{threshold}$. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengubah threshold setelah melihat hasil. Masalah tersebut sering tidak terlihat pada tabel ranking

akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

8.4.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait threshold rmsd.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengubah threshold setelah melihat hasil.
- Keputusan yang dianjurkan: Tetapkan threshold dan metode RMSD sebelum evaluasi.

Dalam konteks pembelajaran, keputusan yang disarankan adalah: Tetapkan threshold dan metode RMSD sebelum evaluasi. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Threshold RMSD” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “threshold rmsd”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus.

Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

8.5 Direct heavy-atom RMSD

RMSD order-based sederhana berguna bila atom mapping dan urutan konsisten. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sisi quality control, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menghitung RMSD tanpa alignment atau symmetry remapping pada daftar heavy atom yang berurutan sama. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menggunakan nilai ini pada molekul simetris atau atom order berbeda. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

8.5.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait direct heavy-atom rmsd.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menggunakan nilai ini pada molekul simetris atau atom order berbeda.
- Keputusan yang dianjurkan: Gunakan metode symmetry-aware eksternal bila diperlukan.

Dari sudut pandang metodologi, keputusan yang disarankan adalah: Gunakan metode symmetry-aware eksternal bila diperlukan. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi

diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Direct heavy-atom RMSD” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “direct heavy-atom rmsd”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 8.3. Skenario keputusan untuk direct heavy-atom rmsd

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menggunakan nilai ini pada molekul simetris atau atom order berbeda.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Gunakan metode symmetry-aware eksternal bila diperlukan.

8.6 Sumber kegagalan redocking

Kegagalan dapat berasal dari receptor prep, protonasi, grid, sampling, koordinat referensi, atau mapping. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan

kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam konteks pembelajaran, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow mengarahkan pengguna meninjau komponen tersebut sebelum screening besar. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah meningkatkan exhaustiveness secara otomatis tanpa diagnosis. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

8.6.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait sumber kegagalan redocking.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Meningkatkan exhaustiveness secara otomatis tanpa diagnosis.
- Keputusan yang dianjurkan: Lakukan troubleshooting berurutan dan ubah satu faktor per eksperimen.

Pada tahap ini, keputusan yang disarankan adalah: Lakukan troubleshooting berurutan dan ubah satu faktor per eksperimen. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur

eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Sumber kegagalan redocking” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “sumber kegagalan redocking”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

8.7 Redocking dan grid referensi

Grid yang diturunkan dari ligan referensi biasanya memberikan search space terarah untuk uji reproduksi. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sudut pandang metodologi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow dapat memakai reference ligand untuk center dan size dengan padding. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah membiarkan ligan referensi masih berada di receptor docking. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan

catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

8.7.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait redocking dan grid referensi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Membiarkan ligan referensi masih berada di receptor docking.
- Keputusan yang dianjurkan: Pisahkan ligan dari receptor sesuai desain protokol.

Dalam praktik komputasi, keputusan yang disarankan adalah: Pisahkan ligan dari receptor sesuai desain protokol. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Redocking dan grid referensi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “redocking dan grid referensi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 8.4. Skenario keputusan untuk redocking dan grid referensi

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Membiarkan ligan referensi masih berada di receptor docking.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Pisahkan ligan dari receptor sesuai desain protokol.

8.8 Replikasi validasi

Replikasi membantu melihat apakah keberhasilan redocking stabil terhadap seed. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Pada tahap ini, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memiliki redocking_replicates dan metadata seed. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menyebut satu pass tunggal sebagai bukti protokol robust. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

8.8.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait replikasi validasi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menyebut satu PASS tunggal sebagai bukti protokol robust.

- Keputusan yang dianjurkan: Periksa konsistensi pose dan score antar-replikasi bila sumber daya memungkinkan.

Jika dibaca sebagai bagian dari workflow, keputusan yang disarankan adalah: Periksa konsistensi pose dan score antar-replikasi bila sumber daya memungkinkan. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Replikasi validasi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “replikasi validasi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

8.9 Dokumentasi validasi

Validasi hanya berguna bila parameter, input, metode RMSD, dan outcome dapat dilacak. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: `redocking_result.json` menyimpan status, `rmsd_A`, `threshold_A`, catatan, dan hasil docking. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah hanya menulis "valid" tanpa angka dan metode. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

8.9.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait dokumentasi validasi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Hanya menulis "valid" tanpa angka dan metode.
- Keputusan yang dianjurkan: Laporkan kriteria, nilai, metode, dan keterbatasan.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Laporkan kriteria, nilai, metode, dan keterbatasan. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Dokumentasi validasi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang

direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “dokumentasi validasi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 8.5. Skenario keputusan untuk dokumentasi validasi

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Hanya menulis "valid" tanpa angka dan metode.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Laporkan kriteria, nilai, metode, dan keterbatasan.

8.10 Rumus dan Indikator yang Relevan

DockFlow menggunakan direct heavy-atom RMSD ketika jumlah dan urutan atom sesuai. Rumus berikut menggambarkan bentuk umum ukuran yang dihitung dari selisih koordinat atom.

$$\text{RMSD} = \sqrt{[(1/N) \times \sum ||r_i - r_i(\text{ref})||^2]}$$

RMSD yang kecil menunjukkan kedekatan geometri pada mapping atom yang digunakan. Ia tidak secara otomatis menilai kesamaan kimia, simetri, atau kualitas afinitas.

8.11 Praktikum Terpandu

1. Buat folder proyek baru dan tulis tujuan eksperimen dalam satu paragraf sebelum menjalankan aplikasi.
2. Catat sumber receptor, ligan, dan semua parameter yang relevan dengan bab ini.

3. Jalankan QC input; bila ada warning, tuliskan keputusan menerima atau memperbaiki beserta alasannya.
4. Jalankan satu baseline terlebih dahulu. Simpan konfigurasi, log, raw output, dan ringkasan hasil.
5. Lakukan satu perubahan terkontrol pada parameter yang sedang dipelajari, lalu bandingkan keluaran dengan baseline.
6. Tulis kesimpulan yang memisahkan observasi komputasi, interpretasi, dan keterbatasan.

8.11.1 Analisis Praktikum 1

Analisis praktikum 1 pada Bab 8 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

8.11.2 Analisis Praktikum 2

Analisis praktikum 2 pada Bab 8 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

8.11.3 Analisis Praktikum 3

Analisis praktikum 3 pada Bab 8 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

8.12 Troubleshooting dan Kesalahan yang Sering Terjadi

Tabel 8.6. Troubleshooting ringkas Bab 8

Area	Gejala/Risiko	Tindakan
Tujuan redocking	Menganggap redocking sebagai validasi afinitas.	Gunakan redocking untuk kemampuan reproduksi pose, bukan pembuktian energi binding.
Pose referensi	Menggunakan geometri referensi yang telah dioptimasi bebas sehingga bukan pose kristalografi asli.	Simpan pose asli dan metode ekstraksi.
Threshold RMSD	Mengubah threshold setelah melihat hasil.	Tetapkan threshold dan metode RMSD sebelum evaluasi.
Direct heavy-atom RMSD	Menggunakan nilai ini pada molekul simetris atau atom order berbeda.	Gunakan metode symmetry-aware eksternal bila diperlukan.
Sumber kegagalan redocking	Meningkatkan exhaustiveness secara otomatis tanpa diagnosis.	Lakukan troubleshooting berurutan dan ubah satu faktor per eksperimen.
Redocking dan grid referensi	Membiarkan ligan referensi masih berada di receptor docking.	Pisahkan ligan dari receptor sesuai desain protokol.

Jika masalah berada pada area tujuan redocking, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap redocking sebagai validasi afinitas. Tindakan awal yang lebih kuat adalah Gunakan

redocking untuk kemampuan reproduksi pose, bukan pembuktian energi binding. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area pose referensi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menggunakan geometri referensi yang telah dioptimasi bebas sehingga bukan pose kristalografi asli. Tindakan awal yang lebih kuat adalah Simpan pose asli dan metode ekstraksi. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area threshold rmsd, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengubah threshold setelah melihat hasil. Tindakan awal yang lebih kuat adalah Tetapkan threshold dan metode RMSD sebelum evaluasi. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area direct heavy-atom rmsd, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menggunakan nilai ini pada molekul simetris atau atom order berbeda. Tindakan awal yang lebih kuat adalah Gunakan metode symmetry-aware eksternal bila diperlukan. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area sumber kegagalan redocking, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah meningkatkan exhaustiveness secara otomatis tanpa diagnosis. Tindakan awal yang lebih kuat adalah Lakukan troubleshooting berurutan dan ubah satu faktor per eksperimen. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area redocking dan grid referensi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah membiarkan ligan referensi masih berada di receptor docking. Tindakan awal yang lebih kuat adalah Pisahkan ligan dari receptor sesuai desain protokol. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

8.13 Latihan: 20 Soal

1. Jelaskan mengapa tujuan redocking perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
2. Jelaskan mengapa pose referensi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
3. Jelaskan mengapa threshold rmsd perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
4. Jelaskan mengapa direct heavy-atom rmsd perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
5. Jelaskan mengapa sumber kegagalan redocking perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
6. Jelaskan mengapa redocking dan grid referensi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
7. Jelaskan mengapa replikasi validasi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
8. Jelaskan mengapa dokumentasi validasi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
9. Susun checklist minimum sebelum menjalankan batch pada Bab 8.
10. Sebuah hasil terlihat sangat baik pada satu indikator, tetapi gagal pada indikator QC lain. Bagaimana seharusnya hasil tersebut diperlakukan?
11. Mengapa perubahan satu parameter sebaiknya diuji secara terkontrol?
12. Jelaskan perbedaan antara data mentah, data turunan, dan interpretasi dalam workflow DockFlow.
13. Apa fungsi provenance ketika dua run menghasilkan hasil berbeda?
14. Kritik pernyataan: “Ligan dengan docking score paling rendah pasti merupakan kandidat terbaik.”
15. Bagaimana Anda menulis bagian metode agar run dapat direproduksi?

16. Mengapa visualisasi tidak boleh menggantikan pemeriksaan numerik?
17. Rancang satu eksperimen sensitivitas yang relevan dengan bab ini.
18. Apa yang harus dilakukan bila hasil tidak dapat direproduksi oleh kelompok lain?
19. Mengapa buku ini menekankan redocking sebelum screening besar?
20. Buat format ringkas laporan hasil yang baik.

8.14 Pembahasan 20 Soal

Pembahasan 1. Redocking adalah quality control pose prediction terhadap ligan dengan pose eksperimen yang diketahui. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menyiapkan ligan referensi dengan `preserve_coords` lalu menjalankan protokol docking. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap redocking sebagai validasi afinitas. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan redocking untuk kemampuan reproduksi pose, bukan pembuktian energi binding.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 2. Referensi harus berasal dari koordinat yang relevan dengan receptor yang sama atau frame yang konsisten. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow dapat memakai ligan referensi dari PDBQT atau koordinat molekul yang dipertahankan. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menggunakan geometri referensi yang telah dioptimasi bebas sehingga bukan pose kristalografi asli. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Simpan pose asli dan metode ekstraksi.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika

terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 3. Threshold adalah kriteria keputusan yang harus dinyatakan sebelum screening. Pada DockFlow, aspek ini diwujudkan melalui: Default konfigurasi `redocking_rmsd_threshold` adalah 2,0 Å dan status PASS diberikan bila $RMSD \leq threshold$. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengubah threshold setelah melihat hasil. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Tetapkan threshold dan metode RMSD sebelum evaluasi.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 4. RMSD order-based sederhana berguna bila atom mapping dan urutan konsisten. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menghitung RMSD tanpa alignment atau symmetry remapping pada daftar heavy atom yang berurutan sama. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menggunakan nilai ini pada molekul simetris atau atom order berbeda. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan metode symmetry-aware eksternal bila diperlukan.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 5. Kegagalan dapat berasal dari receptor prep, protonasi, grid, sampling, koordinat referensi, atau mapping. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow mengarahkan pengguna meninjau komponen tersebut sebelum screening besar. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah meningkatkan exhaustiveness secara otomatis tanpa diagnosis. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan

alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Lakukan troubleshooting berurutan dan ubah satu faktor per eksperimen.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 6. Grid yang diturunkan dari ligan referensi biasanya memberikan search space terarah untuk uji reproduksi. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow dapat memakai reference ligand untuk center dan size dengan padding. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah membiarkan ligan referensi masih berada di receptor docking. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Pisahkan ligan dari receptor sesuai desain protokol.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 7. Replikasi membantu melihat apakah keberhasilan redocking stabil terhadap seed. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memiliki redocking_replicates dan metadata seed. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menyebut satu pass tunggal sebagai bukti protokol robust. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Periksa konsistensi pose dan score antar-replikasi bila sumber daya memungkinkan.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 8. Validasi hanya berguna bila parameter, input, metode RMSD, dan outcome dapat dilacak. Pada DockFlow, aspek ini diwujudkan melalui: `redocking_result.json` menyimpan status, `rmsd_A`, `threshold_A`, catatan, dan hasil docking. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah hanya menulis "valid" tanpa angka dan metode. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Laporkan kriteria, nilai, metode, dan keterbatasan.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 9. Checklist minimum harus memeriksa identitas input, kelengkapan struktur, parameter yang relevan dengan bab ini, ketersediaan backend, serta lokasi artefak yang akan disimpan. Tambahkan kriteria gagal/lanjut yang jelas. Prinsipnya: jangan memulai batch bila masalah dasar sudah terlihat pada QC. Setelah run, simpan konfigurasi, log, dan keluaran mentah agar keputusan dapat diaudit.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 10. Indikator yang baik tidak menghapus kegagalan QC. Hasil perlu ditandai bermasalah, penyebabnya ditelusuri pada tahap input/preparasi/eksekusi, dan bila perlu run diulang dengan protokol yang diperbaiki. Dalam DockFlow, ranking bukan pengganti QC; hasil invalid bahkan mendapat penalty pada composite score. Laporan harus menyebutkan kegagalan dan tidak menjadikan satu angka sebagai pembenaran.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 11. Perubahan satu faktor memudahkan atribusi sebab-akibat. Bila beberapa parameter diubah sekaligus, pengguna tidak dapat mengetahui faktor mana yang menyebabkan perubahan pose, score, atau interaksi. Praktik yang baik adalah membuat baseline, mengubah satu parameter, menjaga input lain identik, membandingkan output, lalu mencatat keputusan. Pendekatan ini juga mempermudah reproduksi oleh kelompok lain.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 12. Data mentah meliputi struktur sumber dan output engine; data turunan meliputi pose terpilih, cluster, fingerprint, frekuensi residu, atau ranking; interpretasi adalah argumen ilmiah yang dibangun dari data tersebut. Ketiganya harus dipisahkan agar pembaca dapat menilai apakah transformasi data masuk akal dan apakah kesimpulan lebih kuat daripada bukti yang tersedia.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 13. Provenance menyediakan rekaman input, versi, seed, parameter, backend, waktu, dan artefak yang memungkinkan perbedaan dilacak. Tanpa provenance, perbedaan hasil mudah dianggap sebagai perilaku molekul padahal mungkin berasal dari perubahan preparasi atau konfigurasi. Dengan provenance, pengguna dapat menyusun perbandingan terkontrol dan mengulang kondisi secara lebih tepat.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 14. Pernyataan itu terlalu kuat. Docking score adalah salah satu keluaran model; kualitas pose, cluster, interaksi, redocking, protonasi, search space, dan sampling juga perlu diperiksa. DockFlow sengaja menyediakan composite ranking dan guardrail agar score tidak berdiri sendiri. Bahkan

composite score tetap hanya alat prioritisasi internal, bukan bukti aktivitas atau afinitas eksperimen.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 15. Tuliskan sumber receptor dan ligan, metode preparasi, kebijakan water/hetero/chain, center dan size grid, backend Vina, scoring, exhaustiveness, num modes, energy range, seed, jumlah replikasi, threshold clustering, cutoff interaksi, metode redocking, versi perangkat lunak, serta cara ekspor/analisis. Sertakan file konfigurasi atau manifest bila tersedia.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 16. Visualisasi sangat baik untuk mendeteksi masalah besar seperti grid di luar pocket atau pose terbalik, tetapi tampilan dapat menyesatkan karena perspektif, skala, dan seleksi objek. Pemeriksaan numerik—koordinat, ukuran box, distance, RMSD, score, dan log—memberikan bukti yang lebih eksplisit. Keduanya harus dipakai secara komplementer.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 17. Mulailah dengan baseline yang telah lolos QC. Pilih satu parameter yang benar-benar terkait dengan bab, misalnya padding grid, exhaustiveness, threshold clustering, atau kebijakan preparasi. Jalankan minimal dua atau tiga tingkat parameter dengan input dan seed/replikasi yang terkontrol. Bandingkan pose, cluster, score, interaksi, dan validasi; jangan hanya membandingkan ranking akhir.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang

berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 18. Bandingkan checksum/hash input, versi engine, metode preparasi, konfigurasi lengkap, seed, receptor PDBQT, grid, dan output mentah. Pastikan tidak ada langkah manual yang tidak tercatat. Bila perbedaan tetap ada, laporkan lingkungan komputasi dan backend aktual. Reprodusibilitas adalah diagnosis berbasis bukti, bukan sekadar mencoba ulang perintah yang sama.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 19. Jika ligan referensi tersedia, redocking memberi uji awal apakah kombinasi receptor preparation, grid, sampling, dan pose mapping mampu mereproduksi struktur yang diketahui. Kegagalan pada tahap ini memberi sinyal untuk memperbaiki protokol sebelum biaya komputasi dan interpretasi diperluas ke banyak ligan.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 20. Laporan minimal memuat tujuan, input dan sumber struktur, parameter, status backend, QC, hasil redocking bila ada, tabel pose/cluster, analisis interaksi, ranking, keterbatasan, serta lampiran provenance dan raw output. Nyatakan dengan jelas mana data komputasi, mana interpretasi, dan mana fakta dari sumber eksperimen atau literatur.

Penerapan pada Bab 8: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

8.15 Soal Pengembangan

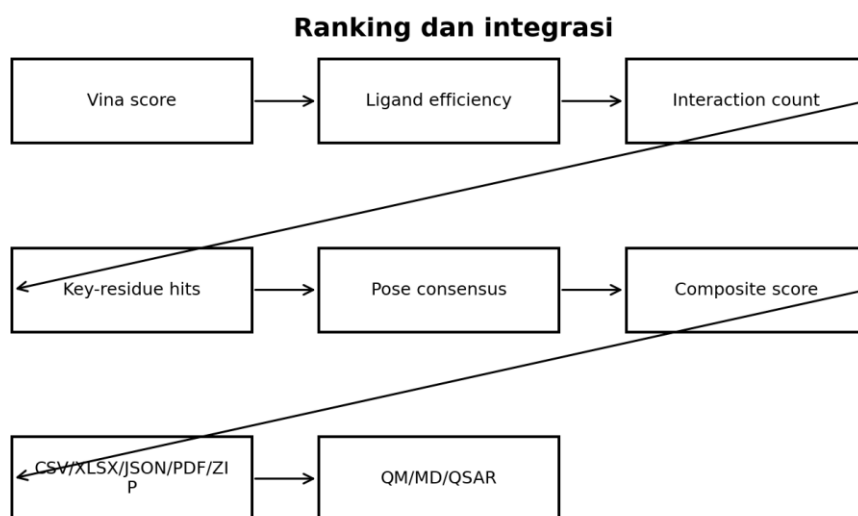
1. Rancang mini-proyek yang menguji satu keputusan metodologis utama Bab 8 dengan baseline dan minimal dua kondisi uji.
2. Susun rubrik penilaian yang memberi bobot pada QC, reproduksibilitas, validasi, interpretasi, dan dokumentasi.
3. Tulis satu contoh klaim yang terlalu kuat dari hasil docking, lalu perbaiki menjadi klaim yang sesuai kekuatan bukti.
4. Buat diagram data lineage dari input sampai satu keluaran akhir dan tunjukkan di mana provenance harus disimpan.
5. Susun daftar informasi yang harus diberikan kepada peneliti lain agar ia dapat mengulang run tanpa bertanya kembali.

BAB 9. RANKING MULTIKRITERIA, PROVENANCE, EKSPOR, DAN INTEGRASI

DockFlow menggabungkan beberapa fitur untuk prioritisasi internal dan mengekspor artefak yang dapat ditelusuri. Bab ini membahas normalisasi, bobot, composite score, SQLite provenance, serta kontrak data ke QM, MD, dan QSAR.

9.1 Capaian Pembelajaran

- Menjelaskan konsep utama Bab 9 dengan istilah yang tepat dan membedakan data komputasi dari inferensi.
- Menghubungkan konsep dengan parameter, file, dan artefak yang benar-benar dihasilkan DockFlow.
- Menyusun quality-control dan kriteria keputusan sebelum menjalankan batch.
- Menganalisis kegagalan secara bertahap dan mendokumentasikan perubahan protokol.
- Menyusun laporan yang dapat diperiksa dan direproduksi.



Gambar 9.1. Skema utama pembahasan Bab 9: Ranking Multikriteria, Provenance, Ekspor, Dan Integrasi.

Tabel 9.1. Bobot default composite ranking

Komponen	Bobot
Docking score	0,40
Ligand efficiency	0,15
Interaction count	0,20
Key-residue hits	0,15

Komponen	Bobot
Pose consensus	0,10

Tabel 9.2. Artefak ekspor proyek

Artefak	Tujuan
ranking.csv	Prioritas ligan dan fitur ringkas
interaction_fingerprint.csv	Matriks fitur interaksi
residue_frequency.csv	Frekuensi residu/tipe interaksi
results.json	Hasil terstruktur lengkap
DockFlow_results.xlsx	Workbook untuk inspeksi tabular
DockFlow_report.pdf	Laporan ringkas proyek
DockFlow_Project.zip	Paket artefak run
integration_manifest.json	Kontrak handoff QM/MD/QSAR

9.2 Ligand efficiency

Ligand efficiency menormalkan skor terhadap jumlah heavy atom untuk memberi konteks ukuran molekul. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menghitung $LE = -\text{best_score}/\text{heavy_atoms}$ ketika kedua nilai tersedia. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah membandingkan le lintas target atau protokol berbeda tanpa konteks. Masalah tersebut sering tidak

terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

9.2.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait ligand efficiency.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Membandingkan LE lintas target atau protokol berbeda tanpa konteks.
- Keputusan yang dianjurkan: Gunakan sebagai fitur internal, bukan ukuran universal.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Gunakan sebagai fitur internal, bukan ukuran universal. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Ligand efficiency” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “ligand efficiency”: apakah struktur dan state identik, apakah receptor dan

grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

9.3 Pose consensus

Consensus dihitung dari ukuran cluster teratas relatif terhadap jumlah pose. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Jika dibaca sebagai bagian dari workflow, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memakai `top_cluster_size/n_poses` sebagai satu komponen ranking. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap consensus sebagai peluang statistik binding. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

9.3.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait pose consensus.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap consensus sebagai peluang statistik binding.
- Keputusan yang dianjurkan: Baca sebagai konsistensi sampling internal.

Dari sisi quality control, keputusan yang disarankan adalah: Baca sebagai konsistensi sampling internal. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam

terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Pose consensus” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “pose consensus”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 9.2. Skenario keputusan untuk pose consensus

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menganggap consensus sebagai peluang statistik binding.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Baca sebagai konsistensi sampling internal.

9.4 Bobot ranking

Bobot menyatakan prioritas analisis dan harus transparan. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran

komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Untuk keperluan reproduksibilitas, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Default: docking 0,40; ligand efficiency 0,15; interaction count 0,20; key residues 0,15; consensus 0,10. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengubah bobot untuk menghasilkan urutan yang diinginkan. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

9.4.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait bobot ranking.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengubah bobot untuk menghasilkan urutan yang diinginkan.
- Keputusan yang dianjurkan: Lakukan analisis sensitivitas dan jelaskan rasional bobot.

Dalam konteks pembelajaran, keputusan yang disarankan adalah: Lakukan analisis sensitivitas dan jelaskan rasional bobot. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow

menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Bobot ranking” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “bobot ranking”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

9.5 Normalisasi fitur

Fitur dengan skala berbeda perlu normalisasi sebelum digabung. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sisi quality control, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow memakai min-max high-is-better; docking dibalik tanda karena score lebih negatif lebih baik. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengabaikan outlier dan distribusi sempit. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

9.5.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait normalisasi fitur.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengabaikan outlier dan distribusi sempit.
- Keputusan yang dianjurkan: Tinjau rentang fitur serta stabilitas ranking.

Dari sudut pandang metodologi, keputusan yang disarankan adalah: Tinjau rentang fitur serta stabilitas ranking. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Normalisasi fitur” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “normalisasi fitur”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 9.3. Skenario keputusan untuk normalisasi fitur

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Mengabaikan outlier dan distribusi sempit.

Elemen	Uraian
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Tinjau rentang fitur serta stabilitas ranking.

9.6 QC penalty

Hasil yang gagal QC tidak seharusnya bersaing setara dengan hasil valid. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam konteks pembelajaran, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Composite score hasil QC invalid dikalikan 0,25 sebelum sorting. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap penalty memperbaiki data yang gagal. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

9.6.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait qc penalty.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap penalty memperbaiki data yang gagal.
- Keputusan yang dianjurkan: Perbaiki sumber kegagalan, jangan hanya mengandalkan penalty.

Pada tahap ini, keputusan yang disarankan adalah: Perbaiki sumber kegagalan, jangan hanya mengandalkan penalty. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “QC penalty” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “qc penalty”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

9.7 Ekspor multi-format

Format berbeda melayani analisis, audit, berbagi, dan dokumentasi. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sudut pandang metodologi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow mengeksport `ranking.csv`, `interaction_fingerprint.csv`, `residue_frequency.csv`, `results.json`, `XLSX`, `PDF`, dan `ZIP` proyek. Setiap operasi menghasilkan artefak atau metadata yang dapat

diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menyebarkan hanya screenshot tabel tanpa raw data. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

9.7.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait ekspor multi-format.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menyebarkan hanya screenshot tabel tanpa raw data.
- Keputusan yang dianjurkan: Simpan minimal data tabular, konfigurasi, raw output, dan manifest.

Dalam praktik komputasi, keputusan yang disarankan adalah: Simpan minimal data tabular, konfigurasi, raw output, dan manifest. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Ekspor multi-format” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai

ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “ekspor multi-format”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 9.4. Skenario keputusan untuk ekspor multi-format

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menyebarkan hanya screenshot tabel tanpa raw data.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Simpan minimal data tabular, konfigurasi, raw output, dan manifest.

9.8 SQLite provenance

Database menyimpan proyek, job, result, dan audit sehingga status dapat ditelusuri. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Pada tahap ini, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Setiap job dan hasil terkait `project_id`, sedangkan audit mencatat tindakan dan detail JSON. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menghapus database saat memindahkan proyek tanpa backup. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber:

struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

9.8.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait sqlite provenance.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menghapus database saat memindahkan proyek tanpa backup.
- Keputusan yang dianjurkan: Paketkan database atau artefak ekuivalen bersama run.

Jika dibaca sebagai bagian dari workflow, keputusan yang disarankan adalah: Paketkan database atau artefak ekuivalen bersama run. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “SQLite provenance” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “sqlite provenance”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus.

Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

9.9 Integration manifest

Handoff terstruktur mengurangi kehilangan konteks ketika data masuk ke workflow berikutnya. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Manifest menyebut `receptor_pdbqt`, ranking, fingerprint, serta target modul QM, MD, QSAR. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah meneruskan pose tanpa informasi receptor dan protokol asal. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

9.9.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait integration manifest.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Meneruskan pose tanpa informasi receptor dan protokol asal.
- Keputusan yang dianjurkan: Gunakan manifest sebagai kontrak data dan catat transformasi downstream.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Gunakan manifest sebagai kontrak data dan catat transformasi downstream. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus

diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Integration manifest” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “integration manifest”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 9.5. Skenario keputusan untuk integration manifest

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Meneruskan pose tanpa informasi receptor dan protokol asal.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Gunakan manifest sebagai kontrak data dan catat transformasi downstream.

9.10 Rumus dan Indikator yang Relevan

$$\text{Ligand efficiency} = -(\text{Best Vina score}) / N_{\text{heavy}}$$

$$\text{Pose consensus} = N_{\text{top_cluster}} / N_{\text{all_poses}}$$

$$\text{Composite} = (\sum w_k \times s_k) / (\sum w_k)$$

Dalam implementasi DockFlow, docking score dibalik tanda sebelum normalisasi karena score yang lebih negatif dianggap lebih baik dalam konteks ranking internal. Composite score tetap harus dibaca sebagai alat prioritisasi, bukan ukuran biologis absolut.

9.11 Praktikum Terpandu

1. Buat folder proyek baru dan tulis tujuan eksperimen dalam satu paragraf sebelum menjalankan aplikasi.
2. Catat sumber receptor, ligan, dan semua parameter yang relevan dengan bab ini.
3. Jalankan QC input; bila ada warning, tuliskan keputusan menerima atau memperbaiki beserta alasannya.
4. Jalankan satu baseline terlebih dahulu. Simpan konfigurasi, log, raw output, dan ringkasan hasil.
5. Lakukan satu perubahan terkontrol pada parameter yang sedang dipelajari, lalu bandingkan keluaran dengan baseline.
6. Tulis kesimpulan yang memisahkan observasi komputasi, interpretasi, dan keterbatasan.

9.11.1 Analisis Praktikum 1

Analisis praktikum 1 pada Bab 9 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

9.11.2 Analisis Praktikum 2

Analisis praktikum 2 pada Bab 9 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

9.11.3 Analisis Praktikum 3

Analisis praktikum 3 pada Bab 9 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

9.12 Troubleshooting dan Kesalahan yang Sering Terjadi

Tabel 9.6. Troubleshooting ringkas Bab 9

Area	Gejala/Risiko	Tindakan
Ligand efficiency	Membandingkan LE lintas target atau protokol berbeda tanpa konteks.	Gunakan sebagai fitur internal, bukan ukuran universal.
Pose consensus	Menganggap consensus sebagai peluang statistik binding.	Baca sebagai konsistensi sampling internal.

Area	Gejala/Risiko	Tindakan
Bobot ranking	Mengubah bobot untuk menghasilkan urutan yang diinginkan.	Lakukan analisis sensitivitas dan jelaskan rasional bobot.
Normalisasi fitur	Mengabaikan outlier dan distribusi sempit.	Tinjau rentang fitur serta stabilitas ranking.
QC penalty	Menganggap penalty memperbaiki data yang gagal.	Perbaiki sumber kegagalan, jangan hanya mengandalkan penalty.
Ekspor multi-format	Menyebarkan hanya screenshot tabel tanpa raw data.	Simpan minimal data tabular, konfigurasi, raw output, dan manifest.

Jika masalah berada pada area ligand efficiency, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah membandingkan le lintas target atau protokol berbeda tanpa konteks. Tindakan awal yang lebih kuat adalah Gunakan sebagai fitur internal, bukan ukuran universal. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area pose consensus, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap consensus sebagai peluang statistik binding. Tindakan awal yang lebih kuat adalah Baca sebagai konsistensi sampling internal. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area bobot ranking, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengubah bobot untuk menghasilkan urutan yang diinginkan. Tindakan awal yang lebih kuat adalah Lakukan analisis sensitivitas dan jelaskan rasional bobot. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area normalisasi fitur, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengabaikan outlier dan distribusi sempit. Tindakan awal yang lebih kuat adalah Tinjau rentang fitur serta stabilitas ranking. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area qc penalty, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap penalty memperbaiki data yang gagal. Tindakan awal yang lebih kuat adalah Perbaiki sumber

kegagalan, jangan hanya mengandalkan penalty. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area ekspor multi-format, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menyebarkan hanya screenshot tabel tanpa raw data. Tindakan awal yang lebih kuat adalah Simpan minimal data tabular, konfigurasi, raw output, dan manifest. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

9.13 Latihan: 20 Soal

1. Jelaskan mengapa ligand efficiency perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
2. Jelaskan mengapa pose consensus perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
3. Jelaskan mengapa bobot ranking perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
4. Jelaskan mengapa normalisasi fitur perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
5. Jelaskan mengapa qc penalty perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
6. Jelaskan mengapa ekspor multi-format perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
7. Jelaskan mengapa sqlite provenance perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
8. Jelaskan mengapa integration manifest perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
9. Susun checklist minimum sebelum menjalankan batch pada Bab 9.

10. Sebuah hasil terlihat sangat baik pada satu indikator, tetapi gagal pada indikator QC lain. Bagaimana seharusnya hasil tersebut diperlakukan?
11. Mengapa perubahan satu parameter sebaiknya diuji secara terkontrol?
12. Jelaskan perbedaan antara data mentah, data turunan, dan interpretasi dalam workflow DockFlow.
13. Apa fungsi provenance ketika dua run menghasilkan hasil berbeda?
14. Kritik pernyataan: “Ligan dengan docking score paling rendah pasti merupakan kandidat terbaik.”
15. Bagaimana Anda menulis bagian metode agar run dapat direproduksi?
16. Mengapa visualisasi tidak boleh menggantikan pemeriksaan numerik?
17. Rancang satu eksperimen sensitivitas yang relevan dengan bab ini.
18. Apa yang harus dilakukan bila hasil tidak dapat direproduksi oleh kelompok lain?
19. Mengapa buku ini menekankan redocking sebelum screening besar?
20. Buat format ringkas laporan hasil yang baik.

9.14 Pembahasan 20 Soal

Pembahasan 1. Ligand efficiency menormalkan skor terhadap jumlah heavy atom untuk memberi konteks ukuran molekul. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menghitung $LE = -best_score/heavy_atoms$ ketika kedua nilai tersedia. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah membandingkan le lintas target atau protokol berbeda tanpa konteks. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan sebagai fitur internal, bukan ukuran universal.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 2. Consensus dihitung dari ukuran cluster teratas relatif terhadap jumlah pose. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memakai $top_cluster_size/n_poses$ sebagai satu komponen ranking. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat.

Kesalahan yang perlu dihindari adalah menganggap consensus sebagai peluang statistik binding. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Baca sebagai konsistensi sampling internal.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 3. Bobot menyatakan prioritas analisis dan harus transparan. Pada DockFlow, aspek ini diwujudkan melalui: Default: docking 0,40; ligand efficiency 0,15; interaction count 0,20; key residues 0,15; consensus 0,10. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengubah bobot untuk menghasilkan urutan yang diinginkan. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Lakukan analisis sensitivitas dan jelaskan rasional bobot.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 4. Fitur dengan skala berbeda perlu normalisasi sebelum digabung. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow memakai min-max high-is-better; docking dibalik tanda karena score lebih negatif lebih baik. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengabaikan outlier dan distribusi sempit. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Tinjau rentang fitur serta stabilitas ranking.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 5. Hasil yang gagal QC tidak seharusnya bersaing setara dengan hasil valid. Pada DockFlow, aspek ini diwujudkan melalui: Composite score hasil QC invalid dikalikan 0,25 sebelum sorting. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap penalty memperbaiki data yang gagal. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Perbaiki sumber kegagalan, jangan hanya mengandalkan penalty.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 6. Format berbeda melayani analisis, audit, berbagi, dan dokumentasi. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow mengekspor ranking.csv, interaction_fingerprint.csv, residue_frequency.csv, results.json, XLSX, PDF, dan ZIP proyek. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menyebarkan hanya screenshot tabel tanpa raw data. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Simpan minimal data tabular, konfigurasi, raw output, dan manifest.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 7. Database menyimpan proyek, job, result, dan audit sehingga status dapat ditelusuri. Pada DockFlow, aspek ini diwujudkan melalui: Setiap job dan hasil terkait project_id, sedangkan audit mencatat tindakan dan detail JSON. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menghapus database saat memindahkan proyek tanpa backup. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Paketkan database atau artefak ekuivalen bersama run.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file

atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 8. Handoff terstruktur mengurangi kehilangan konteks ketika data masuk ke workflow berikutnya. Pada DockFlow, aspek ini diwujudkan melalui: Manifest menyebut `receptor_pdbqt`, ranking, fingerprint, serta target modul QM, MD, QSAR. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah meneruskan pose tanpa informasi receptor dan protokol asal. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan manifest sebagai kontrak data dan catat transformasi downstream.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 9. Checklist minimum harus memeriksa identitas input, kelengkapan struktur, parameter yang relevan dengan bab ini, ketersediaan backend, serta lokasi artefak yang akan disimpan. Tambahkan kriteria gagal/lanjut yang jelas. Prinsipnya: jangan memulai batch bila masalah dasar sudah terlihat pada QC. Setelah run, simpan konfigurasi, log, dan keluaran mentah agar keputusan dapat diaudit.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 10. Indikator yang baik tidak menghapus kegagalan QC. Hasil perlu ditandai bermasalah, penyebabnya ditelusuri pada tahap input/preparasi/eksekusi, dan bila perlu run diulang dengan protokol yang diperbaiki. Dalam DockFlow, ranking bukan pengganti QC; hasil invalid bahkan mendapat penalty pada composite score. Laporan harus menyebutkan kegagalan dan tidak menjadikan satu angka sebagai pembenaran.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file

atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 11. Perubahan satu faktor memudahkan atribusi sebab-akibat. Bila beberapa parameter diubah sekaligus, pengguna tidak dapat mengetahui faktor mana yang menyebabkan perubahan pose, score, atau interaksi. Praktik yang baik adalah membuat baseline, mengubah satu parameter, menjaga input lain identik, membandingkan output, lalu mencatat keputusan. Pendekatan ini juga mempermudah reproduksi oleh kelompok lain.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 12. Data mentah meliputi struktur sumber dan output engine; data turunan meliputi pose terpilih, cluster, fingerprint, frekuensi residu, atau ranking; interpretasi adalah argumen ilmiah yang dibangun dari data tersebut. Ketiganya harus dipisahkan agar pembaca dapat menilai apakah transformasi data masuk akal dan apakah kesimpulan lebih kuat daripada bukti yang tersedia.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 13. Provenance menyediakan rekaman input, versi, seed, parameter, backend, waktu, dan artefak yang memungkinkan perbedaan dilacak. Tanpa provenance, perbedaan hasil mudah dianggap sebagai perilaku molekul padahal mungkin berasal dari perubahan preparasi atau konfigurasi. Dengan provenance, pengguna dapat menyusun perbandingan terkontrol dan mengulang kondisi secara lebih tepat.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 14. Pernyataan itu terlalu kuat. Docking score adalah salah satu keluaran model; kualitas pose, cluster, interaksi, redocking, protonasi, search space, dan sampling juga perlu diperiksa. DockFlow sengaja menyediakan composite ranking dan guardrail agar score tidak berdiri sendiri. Bahkan composite score tetap hanya alat prioritisasi internal, bukan bukti aktivitas atau afinitas eksperimen.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 15. Tuliskan sumber receptor dan ligan, metode preparasi, kebijakan water/hetero/chain, center dan size grid, backend Vina, scoring, exhaustiveness, num modes, energy range, seed, jumlah replikasi, threshold clustering, cutoff interaksi, metode redocking, versi perangkat lunak, serta cara ekspor/analisis. Sertakan file konfigurasi atau manifest bila tersedia.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 16. Visualisasi sangat baik untuk mendeteksi masalah besar seperti grid di luar pocket atau pose terbalik, tetapi tampilan dapat menyesatkan karena perspektif, skala, dan seleksi objek. Pemeriksaan numerik—koordinat, ukuran box, distance, RMSD, score, dan log—memberikan bukti yang lebih eksplisit. Keduanya harus dipakai secara komplementer.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 17. Mulailah dengan baseline yang telah lolos QC. Pilih satu parameter yang benar-benar terkait dengan bab, misalnya padding grid, exhaustiveness, threshold clustering, atau kebijakan preparasi. Jalankan minimal dua atau tiga tingkat parameter dengan input dan seed/replikasi yang terkontrol.

Bandungkan pose, cluster, score, interaksi, dan validasi; jangan hanya membandingkan ranking akhir.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 18. Bandungkan checksum/hash input, versi engine, metode preparasi, konfigurasi lengkap, seed, receptor PDBQT, grid, dan output mentah. Pastikan tidak ada langkah manual yang tidak tercatat. Bila perbedaan tetap ada, laporkan lingkungan komputasi dan backend aktual. Reprodusibilitas adalah diagnosis berbasis bukti, bukan sekadar mencoba ulang perintah yang sama.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 19. Jika ligan referensi tersedia, redocking memberi uji awal apakah kombinasi receptor preparation, grid, sampling, dan pose mapping mampu mereproduksi struktur yang diketahui. Kegagalan pada tahap ini memberi sinyal untuk memperbaiki protokol sebelum biaya komputasi dan interpretasi diperluas ke banyak ligan.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 20. Laporan minimal memuat tujuan, input dan sumber struktur, parameter, status backend, QC, hasil redocking bila ada, tabel pose/cluster, analisis interaksi, ranking, keterbatasan, serta lampiran provenance dan raw output. Nyatakan dengan jelas mana data komputasi, mana interpretasi, dan mana fakta dari sumber eksperimen atau literatur.

Penerapan pada Bab 9: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika

terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

9.15 Soal Pengembangan

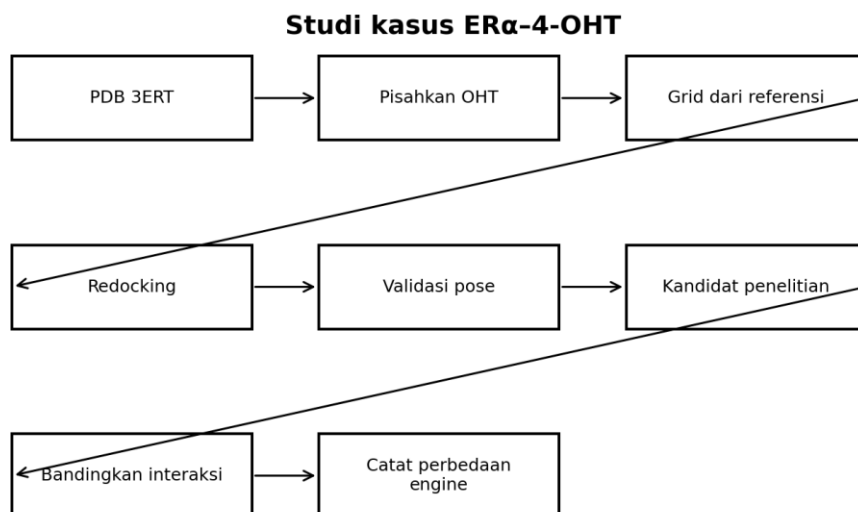
1. Rancang mini-proyek yang menguji satu keputusan metodologis utama Bab 9 dengan baseline dan minimal dua kondisi uji.
2. Susun rubrik penilaian yang memberi bobot pada QC, reproduksibilitas, validasi, interpretasi, dan dokumentasi.
3. Tulis satu contoh klaim yang terlalu kuat dari hasil docking, lalu perbaiki menjadi klaim yang sesuai kekuatan bukti.
4. Buat diagram data lineage dari input sampai satu keluaran akhir dan tunjukkan di mana provenance harus disimpan.
5. Susun daftar informasi yang harus diberikan kepada peneliti lain agar ia dapat mengulang run tanpa bertanya kembali.

BAB 10. STUDI KASUS ERA/3ERT, PRAKTIK PEMBELAJARAN, DAN TROUBLESHOOTING

Bab penutup mengintegrasikan seluruh workflow melalui studi kasus publik ER α /3ERT–4-hydroxytamoxifen dan artikel turunan chalcone yang sudah tertanam dalam tutorial DockFlow. Fokus utamanya adalah reproduksi metodologis, bukan pemaksaan skor absolut antar-engine.

10.1 Capaian Pembelajaran

- Menjelaskan konsep utama Bab 10 dengan istilah yang tepat dan membedakan data komputasi dari inferensi.
- Menghubungkan konsep dengan parameter, file, dan artefak yang benar-benar dihasilkan DockFlow.
- Menyusun quality-control dan kriteria keputusan sebelum menjalankan batch.
- Menganalisis kegagalan secara bertahap dan mendokumentasikan perubahan protokol.
- Menyusun laporan yang dapat diperiksa dan direproduksi.



Gambar 10.1. Skema utama pembahasan Bab 10: Studi Kasus Era/3Ert, Praktik Pembelajaran, Dan Troubleshooting.

Tabel 10.1. Data struktur benchmark yang digunakan tutorial

Komponen	Keterangan
Target	Human estrogen receptor alpha ligand-binding domain
PDB	3ERT
Ligan referensi	4-hydroxytamoxifen / OHT

Komponen	Keterangan
Metode struktur	X-ray diffraction
Resolusi	1,90 Å
Artikel konteks	Muchtaridi dkk., Pharmaceuticals 2017;10(4):81
Engine artikel	AutoDock 4.2
Engine DockFlow	AutoDock Vina

10.2 Struktur PDB 3ERT

3ERT adalah domain ligand-binding estrogen receptor alpha manusia dalam kompleks dengan 4-hydroxytamoxifen dan memiliki struktur X-ray resolusi 1,90 Å. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Tutorial DockFlow memakai 3ERT sebagai sumber receptor dan pose referensi OHT. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengabaikan bahwa entry membawa ligan ko-kristal yang harus dipisahkan untuk redocking. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

10.2.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait struktur pdb 3ert.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.

- Tandai secara eksplisit risiko: Mengabaikan bahwa entry membawa ligan ko-kristal yang harus dipisahkan untuk redocking.
- Keputusan yang dianjurkan: Gunakan struktur sebagai benchmark pose dan dokumentasikan chain/komponen.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Gunakan struktur sebagai benchmark pose dan dokumentasikan chain/komponen. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Struktur PDB 3ERT” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “struktur pdb 3ert”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

10.3 4-hydroxytamoxifen sebagai ligan referensi

Ligan ko-kristal menyediakan koordinat target untuk redocking dan referensi pocket. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa

pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Jika dibaca sebagai bagian dari workflow, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow mengekstrak komponen OHT dan dapat membuat receptor tanpa OHT. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mencampur pose referensi dengan kandidat hasil docking dalam analisis rmsd. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

10.3.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait 4-hydroxytamoxifen sebagai ligan referensi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mencampur pose referensi dengan kandidat hasil docking dalam analisis RMSD.
- Keputusan yang dianjurkan: Pertahankan salinan referensi asli yang tidak berubah.

Dari sisi quality control, keputusan yang disarankan adalah: Pertahankan salinan referensi asli yang tidak berubah. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow

menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “4-hydroxytamoxifen sebagai ligan referensi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “4-hydroxytamoxifen sebagai ligan referensi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 10.2. Skenario keputusan untuk 4-hydroxytamoxifen sebagai ligan referensi

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Mencampur pose referensi dengan kandidat hasil docking dalam analisis RMSD.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Pertahankan salinan referensi asli yang tidak berubah.

10.4 Artikel chalcone sebagai konteks reproduksi

Studi Muchtaridi dkk. 2017 melaporkan docking turunan chalcone pada ER α menggunakan AutoDock 4.2 dan data redocking. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Untuk keperluan reproduksibilitas, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow menyimpan metadata artikel, nilai yang dilaporkan, dan catatan reproduksi. Setiap operasi menghasilkan

artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah memaksa score vina sama dengan autodock 4.2. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

10.4.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait artikel chalcone sebagai konteks reproduksi.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Memaksa score Vina sama dengan AutoDock 4.2.
- Keputusan yang dianjurkan: Bandingkan pose, RMSD, residu, konsistensi protokol, dan tren; jelaskan perbedaan engine.

Dalam konteks pembelajaran, keputusan yang disarankan adalah: Bandingkan pose, RMSD, residu, konsistensi protokol, dan tren; jelaskan perbedaan engine. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Artikel chalcone sebagai konteks reproduksi” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan

aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “artikel chalcone sebagai konteks reproduksi”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

10.5 Langkah redocking terarah

Sebelum screening, receptor disiapkan, OHT dipisahkan, grid diturunkan dari referensi, lalu redocking dijalankan. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sisi quality control, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Tutorial aplikasi menempatkan redocking sebelum kandidat. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah melakukan screening besar ketika validasi awal gagal. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

10.5.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait langkah redocking terarah.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.

- Tandai secara eksplisit risiko: Melakukan screening besar ketika validasi awal gagal.
- Keputusan yang dianjurkan: Jadikan redocking sebagai gerbang QC protokol.

Dari sudut pandang metodologi, keputusan yang disarankan adalah: Jadikan redocking sebagai gerbang QC protokol. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Langkah redocking terarah” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “langkah redocking terarah”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 10.3. Skenario keputusan untuk langkah redocking terarah

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Melakukan screening besar ketika validasi awal gagal.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Jadikan redocking sebagai gerbang QC protokol.

10.6 Perbandingan residu kunci

Residues seperti Glu353 dan Arg394 sering menjadi perhatian dalam studi ERA dan dapat dipakai untuk evaluasi pose. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam konteks pembelajaran, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow dapat memasukkan key residues ke analisis interaksi dan ranking. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap kontak dengan satu residu membuktikan aktivitas. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

10.6.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait perbandingan residu kunci.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap kontak dengan satu residu membuktikan aktivitas.
- Keputusan yang dianjurkan: Gunakan residu sebagai konteks struktur, bukan klaim efek biologis.

Pada tahap ini, keputusan yang disarankan adalah: Gunakan residu sebagai konteks struktur, bukan klaim efek biologis. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi

diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Perbandingan residu kunci” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “perbandingan residu kunci”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

10.7 Perbedaan engine dan parameter

AutoDock 4.2 dan AutoDock Vina memiliki algoritme serta scoring berbeda. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dari sudut pandang metodologi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: DockFlow secara eksplisit memperingatkan bahwa angka absolut tidak wajib identik dengan artikel asli. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang

benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menganggap selisih score sebagai bug tanpa melihat engine dan preparasi. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

10.7.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait perbedaan engine dan parameter.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menganggap selisih score sebagai bug tanpa melihat engine dan preparasi.
- Keputusan yang dianjurkan: Bandingkan aspek yang memang dapat direproduksi dan catat variabel.

Dalam praktik komputasi, keputusan yang disarankan adalah: Bandingkan aspek yang memang dapat direproduksi dan catat variabel. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Perbedaan engine dan parameter” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai

ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “perbedaan engine dan parameter”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 10.4. Skenario keputusan untuk perbedaan engine dan parameter

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menganggap selisih score sebagai bug tanpa melihat engine dan preparasi.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Bandingkan aspek yang memang dapat direproduksi dan catat variabel.

10.8 Troubleshooting berbasis tahap

Error paling efisien diatasi dengan menentukan tahap gagal: input, preparasi, grid, backend, parsing, interaksi, atau ekspor. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Pada tahap ini, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Log, run_status, folder prepared, ligands, redocking, dan exports menyediakan jejak diagnosis. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah mengubah banyak dependensi sekaligus tanpa membaca log. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh

sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

10.8.1 Catatan Quality Control

- Periksa bukti yang mendukung keputusan terkait troubleshooting berbasis tahap.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Mengubah banyak dependensi sekaligus tanpa membaca log.
- Keputusan yang dianjurkan: Isolasi masalah dan verifikasi artefak per tahap.

Jika dibaca sebagai bagian dari workflow, keputusan yang disarankan adalah: Isolasi masalah dan verifikasi artefak per tahap. Keputusan tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Troubleshooting berbasis tahap” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “troubleshooting berbasis tahap”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan

apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

10.9 Desain praktikum dan laporan ilmiah

Workflow dapat dijadikan praktikum komputasi yang menilai kemampuan menyusun protokol, menjalankan QC, dan menafsirkan data. Konsep ini perlu diletakkan di dalam tujuan penelitian yang eksplisit. Pengguna harus mengetahui apakah ia sedang menguji kemampuan reproduksi pose, membandingkan kandidat pada protokol yang sama, memetakan kontak terhadap residu tertentu, atau menyiapkan pose untuk analisis lanjutan. Tanpa pertanyaan yang jelas, keluaran komputasi mudah diperlakukan sebagai angka berdiri sendiri dan kehilangan hubungan dengan asumsi yang membentuknya.

Dalam praktik komputasi, implementasi DockFlow menerjemahkan prinsip tersebut menjadi operasi yang tercatat: Buku ini mengarahkan mahasiswa menghasilkan job specification, hasil, validasi, dan refleksi keterbatasan. Setiap operasi menghasilkan artefak atau metadata yang dapat diperiksa kembali. Keuntungan desain semacam ini bukan hanya kenyamanan pengguna, melainkan kemampuan untuk menghubungkan keputusan di antarmuka dengan file yang benar-benar dipakai oleh backend. Dengan demikian, perubahan hasil dapat ditelusuri ke perubahan input, parameter, versi, atau jalur eksekusi.

Sumber kesalahan yang paling umum pada topik ini adalah menilai mahasiswa hanya dari ligand dengan score paling negatif. Masalah tersebut sering tidak terlihat pada tabel ranking akhir karena tabel hanya merangkum sebagian informasi. Oleh sebab itu, pemeriksaan harus dimulai dari data paling dekat dengan sumber: struktur input, job specification, log preparasi, file PDBQT, output engine, dan catatan QC. Prinsip ini sejalan dengan praktik ilmiah yang memisahkan data mentah dari interpretasi.

10.9.1 Catatan *Quality Control*

- Periksa bukti yang mendukung keputusan terkait desain praktikum dan laporan ilmiah.
- Simpan parameter aktual dan jangan hanya mengandalkan ingatan atau screenshot antarmuka.
- Tandai secara eksplisit risiko: Menilai mahasiswa hanya dari ligand dengan score paling negatif.
- Keputusan yang dianjurkan: Nilai kualitas desain eksperimen, ketertelusuran, validasi, dan argumentasi.

Untuk keperluan reproduksibilitas, keputusan yang disarankan adalah: Nilai kualitas desain eksperimen, ketertelusuran, validasi, dan argumentasi. Keputusan

tersebut sebaiknya ditulis sebelum batch besar dimulai. Bila parameter harus diubah setelah uji awal, perubahan diperlakukan sebagai iterasi protokol baru dan bukan sebagai koreksi diam-diam terhadap hasil yang kurang sesuai. Cara ini mencegah pemilihan parameter secara retrospektif hanya untuk mendapatkan ranking atau pose yang diinginkan.

Hubungan dengan literatur penting karena perangkat lunak mengotomatisasi langkah, tetapi tidak menggantikan pertimbangan kimia struktur. AutoDock Vina menyediakan algoritme pencarian dan scoring; RDKit dan Meeko/Open Babel menangani representasi serta preparasi; RCSB PDB menyediakan banyak struktur eksperimen. Masing-masing memiliki asumsi dan domain penggunaan. DockFlow menyatukannya dalam workflow, sehingga pengguna tetap perlu memahami batas metode di balik setiap tombol.

Untuk praktikum, mahasiswa dapat mendokumentasikan topik “Desain praktikum dan laporan ilmiah” dalam tiga lapis bukti. Lapis pertama adalah konfigurasi yang direncanakan. Lapis kedua adalah artefak aktual yang dihasilkan aplikasi. Lapis ketiga adalah interpretasi yang menjelaskan apakah artefak tersebut mendukung tujuan awal. Bila ketiganya tidak konsisten, laporan harus menandai ketidaksesuaian dan menjelaskan tindakan perbaikan, bukan menyembunyikan hasil yang tidak ideal.

Sebagai latihan reflektif, bayangkan dua kelompok memperoleh urutan ranking berbeda meskipun memakai daftar ligan yang sama. Sebelum menyimpulkan adanya perbedaan sifat molekul, periksa terlebih dahulu hal-hal yang terkait dengan “desain praktikum dan laporan ilmiah”: apakah struktur dan state identik, apakah receptor dan grid sama, apakah backend/seed/exhaustiveness sama, dan apakah QC lulus. Perbandingan baru bermakna setelah kondisi yang membentuk hasil benar-benar sebanding.

Tabel 10.5. Skenario keputusan untuk desain praktikum dan laporan ilmiah

Elemen	Uraian
Kondisi awal	Input/parameter tersedia dan dapat diperiksa.
Risiko utama	Menilai mahasiswa hanya dari ligand dengan score paling negatif.
Bukti yang dicari	Konfigurasi, file antara, log, dan hasil QC.
Tindakan	Nilai kualitas desain eksperimen, ketertelusuran, validasi, dan argumentasi.

10.10 Rumus dan Indikator yang Relevan

Keputusan ilmiah = bukti input + QC + parameter + hasil + validasi +
keterbatasan

Formula konseptual tersebut mengingatkan bahwa satu indikator tidak cukup untuk menopang kesimpulan workflow. Setiap bagian menyediakan jenis bukti yang berbeda dan saling melengkapi.

10.11 Praktikum Terpandu

1. Buat folder proyek baru dan tulis tujuan eksperimen dalam satu paragraf sebelum menjalankan aplikasi.
2. Catat sumber receptor, ligan, dan semua parameter yang relevan dengan bab ini.
3. Jalankan QC input; bila ada warning, tuliskan keputusan menerima atau memperbaiki beserta alasannya.
4. Jalankan satu baseline terlebih dahulu. Simpan konfigurasi, log, raw output, dan ringkasan hasil.
5. Lakukan satu perubahan terkontrol pada parameter yang sedang dipelajari, lalu bandingkan keluaran dengan baseline.
6. Tulis kesimpulan yang memisahkan observasi komputasi, interpretasi, dan keterbatasan.

10.11.1 Analisis Praktikum 1

Analisis praktikum 1 pada Bab 10 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

10.11.2 Analisis Praktikum 2

Analisis praktikum 2 pada Bab 10 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu

parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

10.11.3 Analisis Praktikum 3

Analisis praktikum 3 pada Bab 10 sebaiknya menggunakan tabel perbandingan yang memuat baseline dan kondisi uji. Kolom minimal adalah input, parameter yang berubah, status QC, indikator utama, perubahan pose/cluster/interaksi bila tersedia, dan komentar metodologis. Jangan menyatakan bahwa perubahan satu parameter "meningkatkan kualitas" hanya karena satu skor membaik; periksa apakah validasi, konsistensi sampling, dan interpretasi struktur juga mendukung pernyataan tersebut.

Jika hasil baseline dan kondisi uji sangat berbeda, pertama-tama pastikan perbedaan bukan berasal dari file input, seed yang tidak tercatat, atau backend yang berbeda. Setelah kontrol tersebut terpenuhi, barulah perubahan dapat dibahas sebagai sensitivitas terhadap parameter. Dokumentasi semacam ini membuat praktikum menjadi latihan desain eksperimen komputasi, bukan sekadar latihan mengoperasikan aplikasi.

10.12 Troubleshooting dan Kesalahan yang Sering Terjadi

Tabel 10.6. Troubleshooting ringkas Bab 10

Area	Gejala/Risiko	Tindakan
Struktur PDB 3ERT	Mengabaikan bahwa entry membawa ligan ko-kristal yang harus dipisahkan untuk redocking.	Gunakan struktur sebagai benchmark pose dan dokumentasikan chain/komponen.
4-hydroxytamoxifen sebagai ligan referensi	Mencampur pose referensi dengan kandidat hasil docking dalam analisis RMSD.	Pertahankan salinan referensi asli yang tidak berubah.
Artikel chalcone sebagai konteks reproduksi	Memaksa score Vina sama dengan AutoDock 4.2.	Bandingkan pose, RMSD, residu, konsistensi protokol, dan tren; jelaskan perbedaan engine.
Langkah redocking terarah	Melakukan screening besar ketika validasi awal gagal.	Jadikan redocking sebagai gerbang QC protokol.

Area	Gejala/Risiko	Tindakan
Perbandingan residu kunci	Menganggap kontak dengan satu residu membuktikan aktivitas.	Gunakan residu sebagai konteks struktur, bukan klaim efek biologis.
Perbedaan engine dan parameter	Menganggap selisih score sebagai bug tanpa melihat engine dan preparasi.	Bandingkan aspek yang memang dapat direproduksi dan catat variabel.

Jika masalah berada pada area struktur pdb 3ert, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mengabaikan bahwa entry membawa ligan ko-kristal yang harus dipisahkan untuk redocking. Tindakan awal yang lebih kuat adalah Gunakan struktur sebagai benchmark pose dan dokumentasikan chain/komponen. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area 4-hydroxytamoxifen sebagai ligan referensi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah mencampur pose referensi dengan kandidat hasil docking dalam analisis rmsd. Tindakan awal yang lebih kuat adalah Pertahankan salinan referensi asli yang tidak berubah. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area artikel chalcone sebagai konteks reproduksi, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah memaksa score vina sama dengan autodock 4.2. Tindakan awal yang lebih kuat adalah Bandingkan pose, RMSD, residu, konsistensi protokol, dan tren; jelaskan perbedaan engine. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area langkah redocking terarah, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah melakukan screening besar ketika validasi awal gagal. Tindakan awal yang lebih kuat adalah Jadikan redocking sebagai gerbang QC protokol. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area perbandingan residu kunci, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap kontak dengan satu residu membuktikan aktivitas. Tindakan awal yang lebih kuat adalah Gunakan residu sebagai konteks struktur, bukan klaim efek biologis. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan

artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

Jika masalah berada pada area perbedaan engine dan parameter, jangan langsung mengubah semua parameter. Gejala yang perlu dicurigai adalah menganggap selisih score sebagai bug tanpa melihat engine dan preparasi. Tindakan awal yang lebih kuat adalah Bandingkan aspek yang memang dapat direproduksi dan catat variabel. Setelah perubahan dibuat, ulangi hanya tahap yang diperlukan dan bandingkan artefak sebelum dan sesudah perubahan. Catat hasil diagnosis pada log praktikum atau audit proyek.

10.13 Latihan: 20 Soal

1. Jelaskan mengapa struktur pdb 3ert perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
2. Jelaskan mengapa 4-hydroxytamoxifen sebagai ligan referensi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
3. Jelaskan mengapa artikel chalcone sebagai konteks reproduksi perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
4. Jelaskan mengapa langkah redocking terarah perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
5. Jelaskan mengapa perbandingan residu kunci perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
6. Jelaskan mengapa perbedaan engine dan parameter perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
7. Jelaskan mengapa troubleshooting berbasis tahap perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
8. Jelaskan mengapa desain praktikum dan laporan ilmiah perlu diperlakukan sebagai bagian dari protokol ilmiah, bukan sekadar pengaturan teknis. Kaitkan jawaban dengan perilaku DockFlow.
9. Susun checklist minimum sebelum menjalankan batch pada Bab 10.

10. Sebuah hasil terlihat sangat baik pada satu indikator, tetapi gagal pada indikator QC lain. Bagaimana seharusnya hasil tersebut diperlakukan?
11. Mengapa perubahan satu parameter sebaiknya diuji secara terkontrol?
12. Jelaskan perbedaan antara data mentah, data turunan, dan interpretasi dalam workflow DockFlow.
13. Apa fungsi provenance ketika dua run menghasilkan hasil berbeda?
14. Kritik pernyataan: “Ligan dengan docking score paling rendah pasti merupakan kandidat terbaik.”
15. Bagaimana Anda menulis bagian metode agar run dapat direproduksi?
16. Mengapa visualisasi tidak boleh menggantikan pemeriksaan numerik?
17. Rancang satu eksperimen sensitivitas yang relevan dengan bab ini.
18. Apa yang harus dilakukan bila hasil tidak dapat direproduksi oleh kelompok lain?
19. Mengapa buku ini menekankan redocking sebelum screening besar?
20. Buat format ringkas laporan hasil yang baik.

10.14 Pembahasan 20 Soal

Pembahasan 1. 3ERT adalah domain ligand-binding estrogen receptor alpha manusia dalam kompleks dengan 4-hydroxytamoxifen dan memiliki struktur X-ray resolusi 1,90 Å. Pada DockFlow, aspek ini diwujudkan melalui: Tutorial DockFlow memakai 3ERT sebagai sumber receptor dan pose referensi OHT. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengabaikan bahwa entry membawa ligan ko-kristal yang harus dipisahkan untuk redocking. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan struktur sebagai benchmark pose dan dokumentasikan chain/komponen.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 2. Ligan ko-kristal menyediakan koordinat target untuk redocking dan referensi pocket. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow

mengekstrak komponen OHT dan dapat membuat receptor tanpa OHT. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mencampur pose referensi dengan kandidat hasil docking dalam analisis rmsd. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Pertahankan salinan referensi asli yang tidak berubah.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 3. Studi Muchtaridi dkk. 2017 melaporkan docking turunan chalcone pada ER α menggunakan AutoDock 4.2 dan data redocking. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow menyimpan metadata artikel, nilai yang dilaporkan, dan catatan reproduksi. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah memaksa score vina sama dengan autodock 4.2. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Bandingkan pose, RMSD, residu, konsistensi protokol, dan tren; jelaskan perbedaan engine.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 4. Sebelum screening, receptor disiapkan, OHT dipisahkan, grid diturunkan dari referensi, lalu redocking dijalankan. Pada DockFlow, aspek ini diwujudkan melalui: Tutorial aplikasi menempatkan redocking sebelum kandidat. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah melakukan screening besar ketika validasi awal gagal. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Jadikan redocking sebagai gerbang QC protokol.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file

atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 5. Residues seperti Glu353 dan Arg394 sering menjadi perhatian dalam studi ER α dan dapat dipakai untuk evaluasi pose. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow dapat memasukkan key residues ke analisis interaksi dan ranking. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap kontak dengan satu residu membuktikan aktivitas. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Gunakan residu sebagai konteks struktur, bukan klaim efek biologis.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 6. AutoDock 4.2 dan AutoDock Vina memiliki algoritme serta scoring berbeda. Pada DockFlow, aspek ini diwujudkan melalui: DockFlow secara eksplisit memperingatkan bahwa angka absolut tidak wajib identik dengan artikel asli. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menganggap selisih score sebagai bug tanpa melihat engine dan preparasi. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Bandingkan aspek yang memang dapat direproduksi dan catat variabel.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 7. Error paling efisien diatasi dengan menentukan tahap gagal: input, preparasi, grid, backend, parsing, interaksi, atau ekspor. Pada DockFlow, aspek ini diwujudkan melalui: Log, run_status, folder prepared, ligands, redocking, dan exports menyediakan jejak diagnosis. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah mengubah banyak dependensi sekaligus

tanpa membaca log. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Isolasi masalah dan verifikasi artefak per tahap.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 8. Workflow dapat dijadikan praktikum komputasi yang menilai kemampuan menyusun protokol, menjalankan QC, dan menafsirkan data. Pada DockFlow, aspek ini diwujudkan melalui: Buku ini mengarahkan mahasiswa menghasilkan job specification, hasil, validasi, dan refleksi keterbatasan. Karena itu, keputusan pengguna memengaruhi data yang masuk ke tahap berikutnya dan harus tercatat. Kesalahan yang perlu dihindari adalah menilai mahasiswa hanya dari ligand dengan score paling negatif. Jawaban yang baik menyebutkan tujuan analisis, artefak yang perlu diperiksa, dan alasan mengapa keputusan harus konsisten antar-run. Rekomendasi praktisnya adalah Nilai kualitas desain eksperimen, ketertelusuran, validasi, dan argumentasi.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 9. Checklist minimum harus memeriksa identitas input, kelengkapan struktur, parameter yang relevan dengan bab ini, ketersediaan backend, serta lokasi artefak yang akan disimpan. Tambahkan kriteria gagal/lanjut yang jelas. Prinsipnya: jangan memulai batch bila masalah dasar sudah terlihat pada QC. Setelah run, simpan konfigurasi, log, dan keluaran mentah agar keputusan dapat diaudit.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 10. Indikator yang baik tidak menghapus kegagalan QC. Hasil perlu ditandai bermasalah, penyebabnya ditelusuri pada tahap input/preparasi/eksekusi,

dan bila perlu run diulang dengan protokol yang diperbaiki. Dalam DockFlow, ranking bukan pengganti QC; hasil invalid bahkan mendapat penalty pada composite score. Laporan harus menyebutkan kegagalan dan tidak menjadikan satu angka sebagai pembenaran.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 11. Perubahan satu faktor memudahkan atribusi sebab-akibat. Bila beberapa parameter diubah sekaligus, pengguna tidak dapat mengetahui faktor mana yang menyebabkan perubahan pose, score, atau interaksi. Praktik yang baik adalah membuat baseline, mengubah satu parameter, menjaga input lain identik, membandingkan output, lalu mencatat keputusan. Pendekatan ini juga mempermudah reproduksi oleh kelompok lain.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 12. Data mentah meliputi struktur sumber dan output engine; data turunan meliputi pose terpilih, cluster, fingerprint, frekuensi residu, atau ranking; interpretasi adalah argumen ilmiah yang dibangun dari data tersebut. Ketiganya harus dipisahkan agar pembaca dapat menilai apakah transformasi data masuk akal dan apakah kesimpulan lebih kuat daripada bukti yang tersedia.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 13. Provenance menyediakan rekaman input, versi, seed, parameter, backend, waktu, dan artefak yang memungkinkan perbedaan dilacak. Tanpa provenance, perbedaan hasil mudah dianggap sebagai perilaku molekul padahal mungkin berasal dari perubahan preparasi atau konfigurasi. Dengan provenance, pengguna dapat menyusun perbandingan terkontrol dan mengulang kondisi secara lebih tepat.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 14. Pernyataan itu terlalu kuat. Docking score adalah salah satu keluaran model; kualitas pose, cluster, interaksi, redocking, protonasi, search space, dan sampling juga perlu diperiksa. DockFlow sengaja menyediakan composite ranking dan guardrail agar score tidak berdiri sendiri. Bahkan composite score tetap hanya alat prioritas internal, bukan bukti aktivitas atau afinitas eksperimen.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 15. Tuliskan sumber receptor dan ligan, metode preparasi, kebijakan water/hetero/chain, center dan size grid, backend Vina, scoring, exhaustiveness, num modes, energy range, seed, jumlah replikasi, threshold clustering, cutoff interaksi, metode redocking, versi perangkat lunak, serta cara ekspor/analisis. Sertakan file konfigurasi atau manifest bila tersedia.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 16. Visualisasi sangat baik untuk mendeteksi masalah besar seperti grid di luar pocket atau pose terbalik, tetapi tampilan dapat menyesatkan karena perspektif, skala, dan seleksi objek. Pemeriksaan numerik—koordinat, ukuran

box, distance, RMSD, score, dan log—memberikan bukti yang lebih eksplisit. Keduanya harus dipakai secara komplementer.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 17. Mulailah dengan baseline yang telah lolos QC. Pilih satu parameter yang benar-benar terkait dengan bab, misalnya padding grid, exhaustiveness, threshold clustering, atau kebijakan preparasi. Jalankan minimal dua atau tiga tingkat parameter dengan input dan seed/replikasi yang terkontrol. Bandingkan pose, cluster, score, interaksi, dan validasi; jangan hanya membandingkan ranking akhir.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 18. Bandingkan checksum/hash input, versi engine, metode preparasi, konfigurasi lengkap, seed, receptor PDBQT, grid, dan output mentah. Pastikan tidak ada langkah manual yang tidak tercatat. Bila perbedaan tetap ada, laporkan lingkungan komputasi dan backend aktual. Reprodusibilitas adalah diagnosis berbasis bukti, bukan sekadar mencoba ulang perintah yang sama.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 19. Jika ligan referensi tersedia, redocking memberi uji awal apakah kombinasi receptor preparation, grid, sampling, dan pose mapping mampu mereproduksi struktur yang diketahui. Kegagalan pada tahap ini memberi sinyal untuk memperbaiki protokol sebelum biaya komputasi dan interpretasi diperluas ke banyak ligan.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang

berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

Pembahasan 20. Laporan minimal memuat tujuan, input dan sumber struktur, parameter, status backend, QC, hasil redocking bila ada, tabel pose/cluster, analisis interaksi, ranking, keterbatasan, serta lampiran provenance dan raw output. Nyatakan dengan jelas mana data komputasi, mana interpretasi, dan mana fakta dari sumber eksperimen atau literatur.

Penerapan pada Bab 10: jawaban tersebut perlu dibuktikan dengan artefak yang sesuai, bukan hanya dijelaskan secara lisan. Mahasiswa sebaiknya menunjuk file atau tampilan QC yang mendukung argumen, menyebut parameter yang berpengaruh, dan menuliskan apa yang belum dapat disimpulkan dari data. Jika terdapat lebih dari satu keputusan yang masuk akal, bandingkan konsekuensi metodologis masing-masing dan jelaskan pilihan akhir.

10.15 Soal Pengembangan

1. Rancang mini-proyek yang menguji satu keputusan metodologis utama Bab 10 dengan baseline dan minimal dua kondisi uji.
2. Susun rubrik penilaian yang memberi bobot pada QC, reproduksibilitas, validasi, interpretasi, dan dokumentasi.
3. Tulis satu contoh klaim yang terlalu kuat dari hasil docking, lalu perbaiki menjadi klaim yang sesuai kekuatan bukti.
4. Buat diagram data lineage dari input sampai satu keluaran akhir dan tunjukkan di mana provenance harus disimpan.
5. Susun daftar informasi yang harus diberikan kepada peneliti lain agar ia dapat mengulang run tanpa bertanya kembali.

GLOSARIUM

Glosarium disusun A–Z. Huruf abjad pada bagian ini menggunakan paragraf biasa, bukan Heading 1/2/3, sehingga tidak masuk Daftar Isi. Istilah dipilih untuk mewakili konsep aplikasi, molecular docking, format data, validasi, dan reproducibility.

A

Affinity — Kecenderungan termodinamik suatu ligan berasosiasi dengan target; tidak boleh disamakan langsung dengan docking score.

AutoDock Vina — Engine docking yang digunakan DockFlow melalui Python API atau CLI.

B

Backend — Jalur eksekusi komputasi, misalnya Vina Python API atau Vina CLI.

Binding site — Wilayah reseptor yang menjadi ruang pencarian docking.

C

Cache — Penyimpanan hasil antara berbasis identitas input dan konfigurasi untuk mengurangi komputasi berulang.

Cluster — Kelompok pose yang dianggap serupa berdasarkan ambang RMSD.

Consensus — Proporsi pose pada cluster teratas sebagai indikator konsistensi sampling internal.

D

Docking — Prosedur komputasi untuk mencari pose ligan dalam situs target dan memberi skor pada pose tersebut.

Descriptor — Nilai numerik yang menggambarkan sifat atau karakteristik molekul.

Deduplikasi — Proses menghilangkan struktur input identik agar tidak dihitung berulang.

E

Energy range — Parameter Vina yang membatasi rentang energi pose keluaran.

ETKDG — Metode distance geometry berbasis pengetahuan torsional untuk generasi konformer 3D.

F

Fingerprint interaksi — Vektor fitur yang menandai interaksi/residu tertentu pada pose.

Formal charge — Muatan formal struktur molekul yang perlu diperiksa selama QC.

G

Grid — Kotak search space yang mendefinisikan center dan size docking.

Gasteiger charge — Metode penetapan partial charge yang digunakan pada jalur konversi Open Babel dalam DockFlow.

H

H-bond candidate — Kontak polar yang memenuhi screen jarak dan masih memerlukan verifikasi geometri donor-akseptor.

Hash — Sidik digital seperti SHA-256 untuk identitas data dan cache.

I

InChI — Representasi tekstual struktur kimia yang dapat menjadi input DockFlow.

Interaction count — Jumlah kontak yang terdeteksi pada pose-pose yang dianalisis.

J

Job specification — Kumpulan parameter terstruktur yang mendefinisikan suatu run DockFlow.

JSON — Format data terstruktur yang dipakai untuk konfigurasi, hasil, dan manifest.

K

Key residue — Residu target yang diprioritaskan berdasarkan pengetahuan sebelumnya dan dapat dipakai sebagai fitur ranking.

Kofaktor — Komponen non-protein yang dapat berperan penting dalam struktur/fungsi target dan tidak selalu boleh dihapus.

L

Ligand — Molekul yang ditempatkan pada binding site dalam proses docking.

Ligand efficiency — Fitur yang menormalkan besarnya skor docking terhadap jumlah heavy atom.

M

Manifest — Dokumen terstruktur yang menghubungkan input, konfigurasi, hasil, dan handoff.

Meeko — Perangkat lunak preparasi PDBQT yang menjadi jalur utama preparasi ligan dalam DockFlow.

MMFF94 — Force field molekul yang dicoba DockFlow untuk optimasi geometri ligan.

N

Num modes — Jumlah pose keluaran yang diminta dari Vina.

Normalization — Transformasi fitur agar beberapa skala dapat digabung pada ranking.

O

Open Babel — Toolkit kimia terbuka yang dipakai sebagai converter dan fallback preparasi.

Output mentah — File asli hasil engine yang sebaiknya dipertahankan untuk audit.

P

PDB — Format historis koordinat struktur biomakromolekul.

PDBQT — Format docking yang menambahkan informasi charge dan atom type pada data koordinat.

Pose — Konfigurasi posisi dan orientasi ligan terhadap reseptor.

Provenance — Jejak asal data, parameter, versi, dan tindakan yang memungkinkan hasil ditelusuri.

Q

QC — Quality control; pemeriksaan validitas input, konfigurasi, dan hasil.

Queue — Mekanisme antrean/monitoring job komputasi.

R

Random seed — Nilai awal generator acak yang membantu reproduksibilitas sampling.

Redocking — Docking ulang ligan referensi untuk menguji kemampuan reproduksi pose.

RMSD — Root-mean-square deviation; ukuran perbedaan koordinat antar-pose dengan mapping atom tertentu.

S

Scoring function — Fungsi numerik yang menilai pose dalam model docking.

Search space — Wilayah koordinat yang diizinkan untuk pencarian pose.

SMILES — Notasi tekstual struktur molekul.

Salt bridge — Kontak antara pusat bermuatan berlawanan dalam geometri yang sesuai.

T

TPSA — Topological polar surface area; deskriptor polaritas topologis.

Tautomer — Bentuk struktur yang berbeda karena perpindahan proton dan ikatan yang dapat memengaruhi docking.

U

UFF — Universal Force Field; fallback optimasi geometri pada preparasi ligan.

UUID — Identifier unik yang dapat digunakan untuk membedakan project, job, atau result.

V

Validation — Proses menguji kelayakan protokol terhadap kriteria yang ditetapkan.

Vina score — Skor yang diberikan AutoDock Vina pada pose; bukan pengukuran afinitas eksperimen absolut.

W

Water policy — Keputusan protokol mengenai molekul air pada reseptor.

WAL — Write-Ahead Logging pada SQLite untuk meningkatkan ketahanan akses database.

X

XLSX — Format spreadsheet hasil DockFlow yang memuat ranking dan tabel analisis.

X-ray diffraction — Metode eksperimen struktur yang menghasilkan banyak entry PDB, termasuk 3ERT.

Y

Yield komputasi — Istilah praktis untuk jumlah job yang selesai valid dibanding job yang dijalankan; bukan yield sintesis kimia.

Z

ZIP proyek — Paket terkompresi yang menyatukan artefak run untuk pemindahan dan arsip.

Zero-padding — Konsep penambahan margin ruang pada box referensi; dalam DockFlow direpresentasikan sebagai grid padding, bukan padding nol.

DAFTAR PUSTAKA

Daftar pustaka berikut mencakup literatur metodologis dan dokumentasi perangkat lunak yang digunakan sebagai dasar penjelasan. Referensi perangkat lunak perlu dibaca bersama versi yang benar-benar dipakai pada lingkungan komputasi pengguna.

1. Berman, H. M., Westbrook, J., Feng, Z., et al. (2000). The Protein Data Bank. *Nucleic Acids Research*, 28(1), 235–242. DOI: 10.1093/nar/28.1.235.
2. Eberhardt, J., Santos-Martins, D., Tillack, A. F., & Forli, S. (2021). AutoDock Vina 1.2.0: New Docking Methods, Expanded Force Field, and Python Bindings. *Journal of Chemical Information and Modeling*, 61(8), 3891–3898. DOI: 10.1021/acs.jcim.1c00203.
3. Forli, S., Huey, R., Pique, M. E., Sanner, M. F., Goodsell, D. S., & Olson, A. J. (2016). Computational protein-ligand docking and virtual drug screening with the AutoDock suite. *Nature Protocols*, 11(5), 905–919. DOI: 10.1038/nprot.2016.051.
4. Morris, G. M., Huey, R., Lindstrom, W., et al. (2009). AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility. *Journal of Computational Chemistry*, 30(16), 2785–2791. DOI: 10.1002/jcc.21256.
5. O’Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., & Hutchison, G. R. (2011). Open Babel: An open chemical toolbox. *Journal of Cheminformatics*, 3, 33. DOI: 10.1186/1758-2946-3-33.
6. Pagadala, N. S., Syed, K., & Tuszynski, J. (2017). Software for molecular docking: a review. *Biophysical Reviews*, 9(2), 91–102. DOI: 10.1007/s12551-016-0247-1.
7. Riniker, S., & Landrum, G. A. (2015). Better Informed Distance Geometry: Using What We Know To Improve Conformation Generation. *Journal of Chemical Information and Modeling*, 55(12), 2562–2574. DOI: 10.1021/acs.jcim.5b00654.
8. Trott, O., & Olson, A. J. (2010). AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of Computational Chemistry*, 31(2), 455–461. DOI: 10.1002/jcc.21334.
9. Wang, S., Witek, J., Landrum, G. A., & Riniker, S. (2020). Improving Conformer Generation for Small Rings and Macrocycles Based on Distance Geometry and Experimental Torsional-Angle Preferences. *Journal of*

- Chemical Information and Modeling, 60(4), 2044–2058. DOI: 10.1021/acs.jcim.0c00025.
10. Warren, G. L., Andrews, C. W., Capelli, A.-M., et al. (2006). A critical assessment of docking programs and scoring functions. *Journal of Medicinal Chemistry*, 49(20), 5912–5931. DOI: 10.1021/jm050362n.
 11. Muchtaridi, M., Syahidah, H. N., Subarnas, A., Yusuf, M., Bryant, S. D., & Langer, T. (2017). Molecular Docking and 3D-Pharmacophore Modeling to Study the Interactions of Chalcone Derivatives with Estrogen Receptor Alpha. *Pharmaceuticals*, 10(4), 81. DOI: 10.3390/ph10040081.
 12. Shiau, A. K., Barstad, D., Loria, P. M., et al. (1998). The structural basis of estrogen receptor/coactivator recognition and the antagonism of this interaction by tamoxifen. *Cell*, 95(7), 927–937.
 13. RCSB Protein Data Bank. Entry 3ERT: Human estrogen receptor alpha ligand-binding domain in complex with 4-hydroxytamoxifen. PDB DOI: 10.2210/pdb3ERT/pdb.
 14. RDKit Documentation. The RDKit Book, bagian conformer generation dan ETKDG. Dokumentasi perangkat lunak yang digunakan sebagai landasan teknis preparasi molekul.
 15. Meeko Documentation. Basic ligand preparation, receptor preparation, dan basic docking tutorials. Dokumentasi perangkat lunak untuk preparasi PDBQT dan integrasi AutoDock Vina.
 16. AutoDock Vina Documentation. Dokumentasi resmi penggunaan command line, Python bindings, search space, scoring, dan docking workflow.
 17. Streamlit Documentation. Dokumentasi framework antarmuka web Python yang digunakan oleh DockFlow.
 18. Python Software Foundation. Python Documentation. Dokumentasi bahasa pemrograman dan pustaka standar yang mendasari aplikasi.
 19. SQLite Documentation. Dokumentasi SQLite untuk transaction, WAL, foreign keys, dan persistence data lokal.
 20. NumPy Documentation. Dokumentasi array numerik dan operasi linear algebra yang mendukung pengolahan koordinat.
 21. pandas Documentation. Dokumentasi DataFrame, CSV, dan integrasi ekspor data tabular.
 22. Plotly Python Documentation. Dokumentasi visualisasi interaktif yang mendukung tampilan grid dan grafik hasil.

23. py3Dmol Documentation. Dokumentasi visualisasi struktur molekul 3D berbasis 3Dmol.js.
24. ReportLab User Guide. Dokumentasi pembuatan PDF programatik untuk ringkasan hasil DockFlow.
25. Office Open XML / ECMA-376. Spesifikasi field, heading, hyperlink, dan struktur dokumen yang relevan untuk format buku ini.
26. DockFlow V1.1.4 source code. Molecular Docking & Ligand–Protein Interaction Workflow Manager. Berkas pengguna yang menjadi sumber primer fungsi, parameter default, arsitektur, dan tutorial buku ini.

Catatan Pemanfaatan Literatur

AutoDock Vina: Gunakan artikel Trott & Olson untuk landasan Vina generasi awal dan Eberhardt dkk. untuk perkembangan Vina 1.2, Python bindings, serta fitur docking yang diperluas.

AutoDock Suite: Artikel Morris dkk. dan protokol Forli dkk. membantu memahami preparasi, asumsi docking, dan praktik virtual screening di luar antarmuka DockFlow.

RDKit/ETKDG: Riniker & Landrum serta Wang dkk. menjelaskan dasar conformer generation yang relevan dengan penggunaan ETKDGv3 dalam preparasi ligan.

Protein Data Bank: Berman dkk. dan entry RCSB 3ERT memberi konteks sumber struktur eksperimen dan identitas sistem benchmark.

Kritik scoring: Warren dkk. serta review Pagadala dkk. penting untuk memahami mengapa docking score tidak boleh diperlakukan sebagai estimator afinitas universal.

Studi kasus: Muchtaridi dkk. menjadi konteks tutorial ER α /chalcone yang tertanam dalam aplikasi, tetapi perbedaan engine harus dipertimbangkan saat reproduksi.

- KASMUI -
Dosen Kimia, Komputasi, IT, AI UNNES

<https://hisabmu.com/>
<https://kasmui.cloud/>

Ketua PCM Gunungpati 2	Anggota Majelis Tabligh PDM Kota Semarang & PWM Jawa Tengah	Tim Pengembang Software KHGT MTT PP Muhammadiyah	Praktisi Ilmu Falak
---------------------------	---	--	------------------------