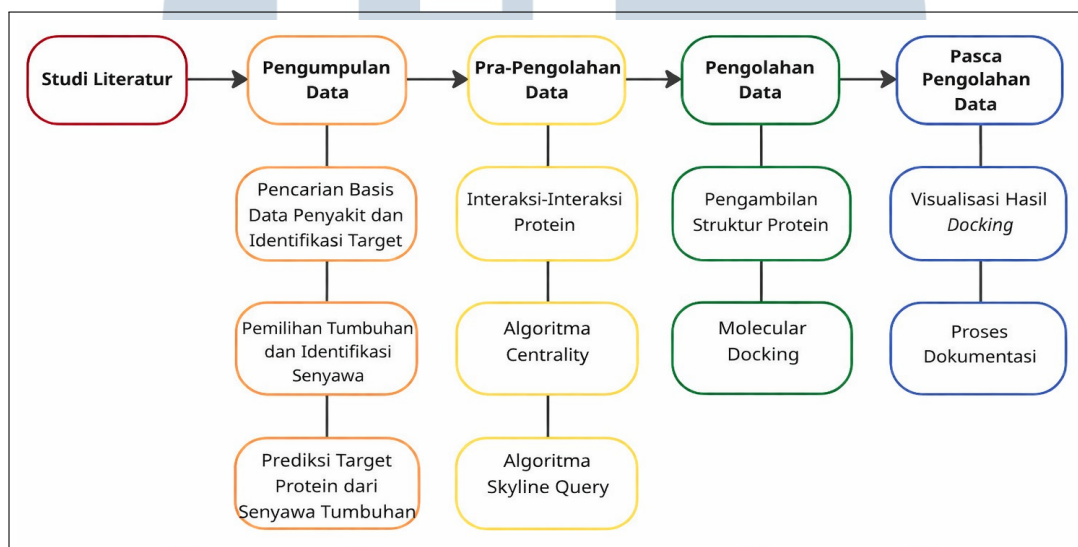


### BAB 3

## METODOLOGI PENELITIAN

Bab ini menjelaskan metode penelitian yang sistematis digunakan dalam studi ini, dengan penjelasan langkah demi langkah mulai dari pengumpulan data awal hingga pembuatan dokumen akhir. Metode ini mengikuti proses yang terorganisir untuk mengidentifikasi penyakit yang dituju serta melakukan analisis senyawa yang berasal dari tumbuhan.



Gambar 3.1. Alur Metodologi Penelitian.

Gambar 3.1 Menjelaskan mengenai proses kerja penelitian yang dimulai dengan tinjauan literatur untuk membentuk dasar teoritis. Selanjutnya, masuk ke tahap pengumpulan data yang melibatkan pencarian sumber informasi mengenai penyakit serta pemilihan tumbuhan. Pada fase pra-pengolahan data, data dilakukan pemurnian melalui analisis interaksi protein dan metode algoritma centrality, serta menggunakan algoritma skyline query untuk mendapatkan lima kandidat target utama. Pada tahap pengolahan data utama, dilakukan pengambilan struktur protein dan pelaksanaan simulasi docking molekuler untuk mengevaluasi interaksi antara senyawa dan protein. Sebagai penutup, dilakukan proses pasca-pengolahan data yang berupa visualisasi hasil guna memudahkan pemahaman, penyampaian, serta dokumentasi secara sistematis.

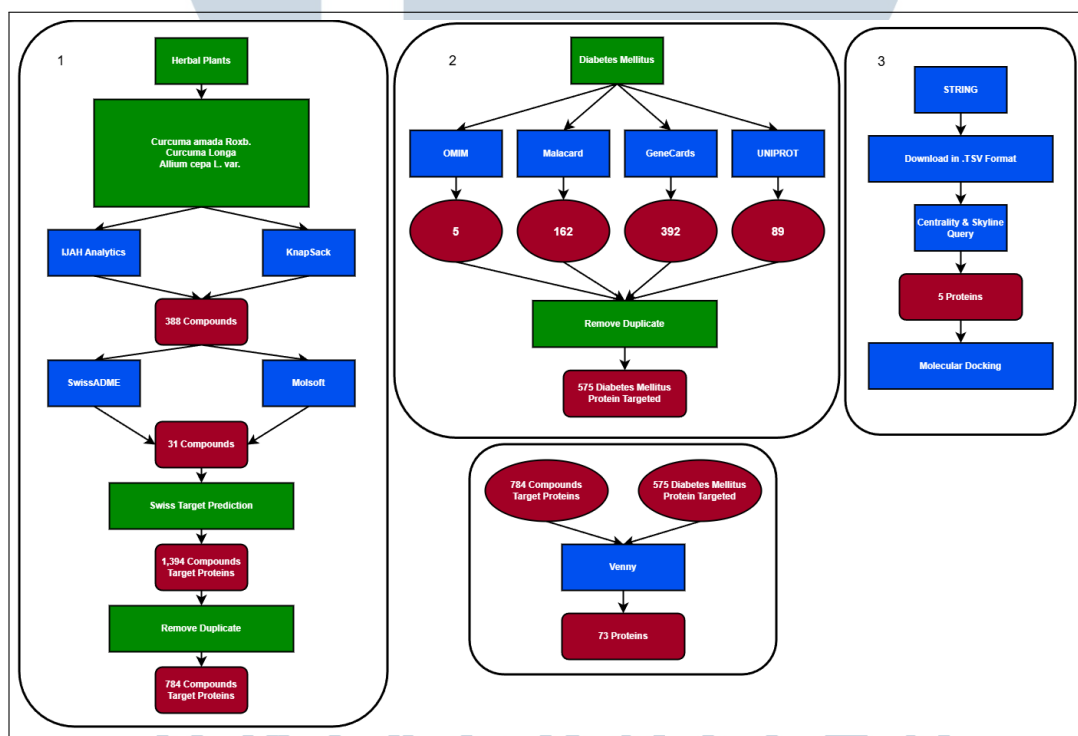
### 3.1 Studi Literatur

Studi literatur ini menjelaskan jurnal utama yang digunakan sebagai acuan dalam penelitian ini. Penelitian yang berjudul "Unveiling the anti-obesity potential of *Kemuning*" menggunakan pendekatan farmakologi jaringan terintegrasi dan molecular docking untuk mengidentifikasi mekanisme anti-obesitas dari tanaman *Murraya paniculata*. Para peneliti menggunakan teknik kueri Skyline untuk mengidentifikasi EP300 dan PPARG sebagai target utama, yang kemudian diverifikasi melalui simulasi dinamika molekuler dan uji in vitro. Hasil penelitian menunjukkan bahwa metabolit aktif dari tanaman tersebut dapat secara efektif menghambat proses maturasi sel lemak dan metabolisme lemak, memberikan dasar ilmiah yang kuat untuk penggunaan tanaman ini dalam pengobatan obesitas [31]. Sementara itu, penelitian "Network Pharmacology and Molecular Mechanism of Traditional Indonesian Medicine in Hypertension Treatment" berhasil mengungkap mekanisme farmakologis dari formulasi herbal Indonesia yang digunakan dalam pengobatan hipertensi. Melalui analisis farmakologi jaringan, peneliti mengidentifikasi 44 senyawa aktif yang berinteraksi dengan jalur sinyal utama, seperti PI3K/MAPK dan HIF-1. Studi ini menunjukkan bahwa efektivitas ramuan tradisional ini berasal dari kemampuannya untuk menargetkan beberapa jalur molekuler secara bersamaan, terutama yang terkait dengan fungsi endotel vaskular, memberikan dasar ilmiah untuk penggunaan Jamu dalam terapi kardiovaskular [32].

Integrasi pengobatan tradisional Indonesia ke dalam kerangka kerja terapeutik modern semakin mendapat dukungan dari farmakologi jaringan, seperti yang ditunjukkan dalam penelitian Setiani dkk. (2024) dan Fatriani dkk. (2024). Setiani dkk. menjelaskan mekanisme multi-target Jamu dalam pengobatan hipertensi melalui analisis jalur (HIF-1, PI3K), sementara Fatriani dkk. mengembangkan pendekatan komputasional ini dengan menerapkan dinamika molekuler dan uji in vitro untuk memvalidasi potensi anti-obesitas *Murraya paniculata*. Secara keseluruhan, kedua studi tersebut menunjukkan perkembangan metodologis yang konsisten, yaitu berpindah dari pembuatan model jaringan yang luas ke validasi struktur yang lebih tepat. Transisi ini membuktikan pentingnya penggunaan teknik penyaringan yang ketat, seperti Algoritma skyline query yang digunakan dalam penelitian ini, untuk mengatasi kesenjangan antara pengetahuan etnobotani dan kedokteran berbasis bukti [31, 32].

### 3.2 Pengumpulan Data

Fase pengumpulan data menjadi dasar penting dalam studi penemuan obat komputasional ini, yang dilakukan melalui pengumpulan dan validasi informasi biologis serta kimia secara sistematis dari sumber data yang dapat dipercaya. Proses ini terdiri dari dua komponen utama. Komponen pertama adalah identifikasi dan validasi target penyakit, yang mencakup pencarian target penyakit utama, pengujian lintas basis data, serta identifikasi protein sebagai target. Komponen kedua adalah pemilihan tanaman obat yang memiliki potensi terapeutik dan pengumpulan data senyawa bioaktifnya. Pendekatan yang digunakan menggabungkan pengetahuan dari pengobatan tradisional dengan sumber daya bioinformatika modern, sehingga memastikan karakterisasi yang menyeluruh terhadap target biologis dan senyawa berasal dari produk alami sebelum dilakukan analisis komputasional.



Gambar 3.2. Alur Pengumpulan Data.

#### 3.2.1 Pengumpulan Data

Fase pertama melibatkan eksplorasi komprehensif basis data penyakit menggunakan sumber-sumber otoritatif:

##### A. Pencarian Target Penyakit Utama

Akses basis data *Online Mendelian Inheritance in Man* (OMIM), UniProt, MalaCards, GeneCards untuk mengidentifikasi penyakit target dan memperoleh informasi penyakit yang relevan.

#### **B. Validasi dan Perbandingan Penyakit**

Gunakan basis data IJAH dengan memasukkan OMIM-ID yang diperoleh dari langkah sebelumnya untuk memvalidasi informasi penyakit.

### **3.2.2 Pemilihan Tumbuhan dan Identifikasi Senyawa**

Fase kedua berfokus pada identifikasi tumbuhan obat yang sesuai dan senyawa bioaktifnya:

#### **A. Pemilihan Spesies Tumbuhan**

Menggunakan basis data IJAH kemudian memasukkan kode OMIM dan pilih salah satu dari sekian banyak tumbuhan yang nantinya keluar dari hasil pencarian.

#### **B. Pengumpulan Data Senyawa Tumbuhan**

Senyawa aktif tumbuhan beserta protein terkait dikumpulkan untuk setiap tanaman (temu mangga, kunyit, dan bawang merah) dari basis data IJAH Analytics dan KNApSACk, dengan menggunakan nama ilmiah masing-masing tanaman sebagai kata kunci pencarian (*Curcuma amada* Roxb. rimpang, *Curcuma longa*, dan *Allium cepa* L. var. *aggregatum*). IJAH Analytics terutama menyediakan data mengenai tanaman obat Indonesia, sedangkan KNApSACk menyediakan informasi senyawa yang dikumpulkan secara global. Setelah daftar senyawa diperoleh, entri duplikat dihapus menggunakan Venny, yaitu alat berbasis web untuk pembuatan diagram Venn.

### **3.3 Pra-Pengolahan Data**

Fase persiapan sebelum pengolahan data dilakukan secara teratur untuk menganalisis dan memperbaiki data biologis yang telah dikumpulkan, dengan tujuan mengidentifikasi protein target yang paling relevan untuk studi docking molekuler. Proses ini mencakup tiga komponen utama, yaitu analisis interaksi protein yang mencakup prediksi target serta perpotongan antara protein penyakit

dan tumbuhan, penerapan algoritma sentralitas untuk menilai tingkat pentingnya jaringan protein, serta penggunaan algoritma skyline query untuk memilih lima protein target yang paling menjanjikan. Pendekatan komputasional ini memastikan eksperimen docking berikutnya berfokus pada target yang memiliki relevansi biologis tinggi, sehingga mengoptimalkan efisiensi dalam penelitian dan meningkatkan kemungkinan menemukan obat.

### 3.3.1 Interaksi-Interaksi Protein

Tahapan selanjutnya adalah penyaringan senyawa setelah penghapusan duplikasi. Senyawa dengan nilai bioavailabilitas oral  $\geq 0.5$  dan nilai drug-likeness  $> 0$  dipilih untuk analisis lebih lanjut serta digunakan dalam proses identifikasi protein terkait. Evaluasi bioavailabilitas oral dilakukan menggunakan SwissADME, sedangkan penilaian drug-likeness dilakukan menggunakan Molsoft.

Protein yang berhubungan dengan setiap senyawa terpilih diprediksi menggunakan SwissTargetPrediction. Setelah target protein diperoleh, target duplikat dihapus kembali menggunakan Venny.

Untuk mengidentifikasi target kandidat yang akan dianalisis lebih lanjut, diagram Venn digunakan untuk menentukan irisan antara target yang terkait dengan tanaman dan target yang berhubungan dengan diabetes. Target yang saling beririsan tersebut kemudian dianggap sebagai target potensial dalam penelitian ini.

Data interaksi protein-protein (protein-protein interaction, PPI) dihasilkan menggunakan STRING versi 12.0. Pada penelitian ini, tingkat kepercayaan menengah (medium confidence) dipilih, dan jaringan interaksi diunduh dalam format TSV.

Berkas TSV yang diperoleh dari STRING selanjutnya digunakan untuk membangun matriks ketetanggaan (adjacency matrix) sebagai dasar dalam analisis jaringan.

### 3.3.2 Algoritma Centrality

Tahap ini bertujuan untuk mencari degree, betweenness, closeness, dan eigenvector dengan menggunakan algoritma sentralitas:

```
1 # Hitung semua centrality
2 degree Centrality = nx.degree_centrality(G)
3 betweenness Centrality = nx.betweenness_centrality(G)
4 closeness Centrality = nx.closeness_centrality(G)
```

```

5 eigenvector_centrality = nx.eigenvector_centrality(G, max_iter
    =1000)
6
7 # Gabungkan hasil ke satu DataFrame
8 centrality_df = pd.DataFrame({
9     'Degree Centrality': degree_centrality,
10    'Betweenness Centrality': betweenness_centrality,
11    'Closeness Centrality': closeness_centrality,
12    'Eigenvector Centrality': eigenvector_centrality
13 })
14
15 print("\n=== CENTRALITY RESULTS ===")
16 print(centrality_df.sort_values(
17     by='Degree Centrality', ascending=False).head(10))

```

### Kode 3.1: Implementasi Perhitungan Centrality Menggunakan NetworkX

Pada lampiran Kode 3.3.2 merupakan kode dengan bahasa Python menggunakan library NetworkX untuk mencari nilai dari ke-empat centrality.

### 3.3.3 Pencarian 5 Protein Teratas dengan Algoritma Skyline Query

```

1 data = {
2     "Display_Name": ["Nama_nama_protein"],
3     "Degree_Centrality": ["nilai_nilai_degree"],
4     "Betweenness_Centrality": ["nilai_nilai_betweenness"],
5     "Closeness_Centrality": ["nilai_nilai_closeness"],
6     "Eigenvector_Centrality": ["nilai_nilai_eigenvector"]
7 }
8
9 df = pd.DataFrame(data)
10
11 def dominates(a, b):
12     return all(x >= y for x, y in zip(a, b)) and any(x > y for x,
13     y in zip(a, b))
14
15 def skyline_query(df, criteria_cols): #Algoritma Skyline Query
16     skyline = []
17     for _, row in df.iterrows():
18         point = row[criteria_cols].values
19         dominated = False
20         to_remove = []
21         for s in skyline:

```



```

21         if dominates(s[criteria_cols].values, point):
22             dominated = True
23             break
24         elif dominates(point, s[criteria_cols].values):
25             to_remove.append(s)
26     if not dominated:
27         for r in to_remove:
28             skyline.remove(r)
29         skyline.append(row)
30     return pd.DataFrame(skyline).reset_index(drop=True)
31
32 criteria = ["Degree_Centrality", "Betweenness_Centrality", "
33            Closeness_Centrality", "Eigenvector_Centrality"]
34 skyline_result = skyline_query(df, criteria)
35
36 top10 = df.sort_values(by=criteria, ascending=False).head(10)
37 print(top10[["Display_Name"] + criteria])

```

Kode 3.2: Implementasi Algoritma Skyline Query Menggunakan NetworkX

Pencarian 5 protein teratas menggunakan algoritma skyline query, algoritma skyline query sendiri merupakan algoritma yang dapat memberikan hasil output dari banyaknya kriteria. Pada penelitian ini kriteria yang digunakan ada 4 *centrality*, antara lain betweenness, closeness, degree, eigenvector. Lampiran kode diatas merupakan kode untuk SkyLine Query menggunakan bahasa Python.

### 3.4 Pengolahan Data

Strategi pengolahan data ini berfokus pada pelaksanaan analisis molecular docking menggunakan protein-protein teratas yang diidentifikasi dari langkah sebelumnya

#### 3.4.1 Pengambilan Struktur Protein

Reseptor dan ligan dipilih berdasarkan hasil Skyline query, di mana target protein dengan peringkat terbaik diidentifikasi. Setelah target protein ditentukan, pencarian manual dilakukan untuk mengidentifikasi senyawa yang relevan dan berasosiasi dengan protein tersebut. Struktur reseptor (protein) diperoleh dari RCSB Protein Data Bank dan diunduh dalam format PDB legacy. Struktur ligan (senyawa) diperoleh dari PubChem dan diunduh dalam format SDF. Selanjutnya,

berkas ligan dikonversi dari format SDF ke PDB menggunakan OpenBabel GUI, yaitu perangkat lunak untuk konversi format berkas molekuler.

### 3.4.2 Molecular Docking

#### A. Pengaturan Konfigurasi Grid

Lakukan grid menggunakan AutoDockTools, pada AutoDockTools buka pengaturan "Grid > Grid Box", atur *center* menjadi *center* pada ligan "Center > Center on Ligand", atur juga *spacing* menjadi 1.000, terakhir sesuaikan box agar ligan berada dalam box dan jangan terlalu besar atau terlalu kecil. Simpan hasil *grid* ke dalam berkas konfigurasi berformat .txt, dengan berisikan:

```
1 center_x = (nilai_koordinat_X)
2 center_y = (nilai_koordinat_Y)
3 center_z = (nilai_koordinat_Z)
4
5 size_x = (nilai_besar_box_sumbu_X)
6 size_y = (nilai_besar_box_sumbu_Y)
7 size_z = (nilai_besar_box_sumbu_Z)
8
9 spacing = 1.000
```

Kode 3.3: File Grid Config

#### B. Pelaksanaan Molecular Docking

Dalam tahap persiapan molecular docking, AutoDockTools digunakan untuk preparasi reseptor dan ligan kemudian di docking dengan AutoDock Vina versi 1.2.7. Setiap reseptor didocking dengan masing-masing ligan, dan hasil terbaik dipilih berdasarkan nilai afinitas ikatan tertinggi, yaitu skor docking yang paling negatif. Pertama buka CMD pada komputer, gunakan command "CD.." untuk memundurkan direktori, gunakan command "CD (nama file di direktori yang sama)" untuk berpindah ke direktori yang dituju, pada penelitian ini file yang dituju adalah "vina" maka "CD vina". Terakhir jalankan command dibawah ini :

```
1 vina.exe --receptor (nama_reseptor).pdbqt --ligand (nama_ligand)
   .pdbqt --config (nama_konfigurasi).txt --exhaustiveness (
   berkelipatan 8 dimulai dari 8) --out (hasil_docking).pdbqt
```

Kode 3.4: Command Vina



### C. Pengumpulan Hasil

Output dari docking menggunakan AutoDock Vina 1.2.7 dapat disimpan ke dalam file .txt dari CMD.

### D. Hasil *Top 3 Ligand*

Setelah molecular docking selesai dilakukan, pilih 3 ligan terbaik berdasarkan nilai afinitas (semakin negatif hasil, berarti keterikatannya semakin kuat) pada setiap reseptor.

## 3.5 Pasca Pengolahan Data

Tahap berikutnya setelah mendapatkan hasil 3 ligan terbaik dari setiap reseptor ialah visualisasi ikatan-ikatan yang terjadi pada protein.

### 3.5.1 Visualisasi Hasil Docking

Visualisasi hasil docking akan terbentuk dalam 2 tipe, tipe pertama yaitu 3 Dimensi, dan tipe yang kedua 2 Dimensi.

#### A. Pengolahan Berkelompok

Proses visualisasi dilakukan dengan memasangkan reseptor dan ligan. Dalam proses ini reseptor dan ligan tidak boleh dipisahkan untuk memastikan kekomprehensifan dari hasil *molecular docking*.

#### B. Visualisasi 3D Menggunakan PyMol

Buka file reseptor pada perangkat lunak PyMol, masukkan *file* ligan, setelah hasil dapat dilihat ekspor dengan format .pdb. Setelah itu dapat di save dengan format file .png

#### C. Visualisasi 2D Menggunakan LigPlot

Visualisasi 2D lebih mudah jika sudah melakukan 3D terlebih dahulu, tinggal memasukkan file .pdb yang di ekspor dari perangkat lunak PyMol, kemudian hasil akan langsung keluar dalam bentuk 2D. Lalu screenshot dan simpan dalam format .png