

## BAB II

### LANDASAN TEORI

#### 2.1. Tinjauan Pustaka

##### 1. Data Warehouse

Data *warehouse* adalah sebuah sistem yang mengintegrasikan data dari berbagai sumber operasional untuk menyediakan lingkungan yang stabil untuk pelaporan, kueri, dan analisis [6]. Fungsi utama dari data *warehouse* adalah menyediakan satu sumber data yang terpadu, yang memungkinkan analisis dan pengambilan keputusan yang lebih efektif. Data *warehouse* menyimpan data dalam format yang dioptimalkan untuk analisis, termasuk data historis untuk memungkinkan analisis tren dan pola. Umumnya, sebelum data dimasukkan ke dalam data *warehouse*, dilakukan proses pembersihan dan transformasi untuk memastikan kualitas dan konsistensi data [7].

Perusahaan yang bergerak di bidang komersial perlu melakukan analisis kinerja penjualan [1]. Data *warehouse* adalah sebuah basis data komprehensif yang mendukung semua analisis keputusan yang diperlukan oleh suatu perusahaan dengan menyediakan ringkasan dan rincian informasi [2]. *Extract-Transform-Load* (ETL) menjadi dasar dari penerapan data *warehouse* [3].

##### 2. *Extract, Transform, Load* (ETL)

ETL adalah proses yang terdiri dari tiga tahap utama untuk memindahkan data dari berbagai sumber ke dalam data *warehouse*. *Extract* adalah tahapan di mana data diekstrak atau diambil dari sumbernya. *Transform* adalah tahapan di mana data yang telah diekstrak kemudian diubah atau dikonversi ke format yang

sesuai dengan kebutuhan data *warehouse*, yang melibatkan pembersihan data, penggabungan data, dan standarisasi format. *Load* adalah tahapan terakhir di mana data yang telah diubah dimuat ke dalam data *warehouse* [6].

### 3. *Business Intelligence (BI)*

*Business Intelligence* adalah serangkaian strategi, proses, dan teknologi yang digunakan untuk menganalisis data bisnis. *Business Intelligence* melibatkan proses mengubah data menjadi wawasan yang dapat ditindaklanjuti untuk membantu pengambilan keputusan bisnis [7]. Dalam konteks ini, *dashboard Business Intelligence* menjadi alat yang krusial untuk memantau kinerja bisnis melalui *Key Performance Indicators (KPI)* [8].

### 4. *Apache Superset*

*Superset* adalah platform eksplorasi dan visualisasi data modern yang dapat menggantikan atau melengkapi alat *business intelligence proprietary* untuk banyak tim. *Superset* terintegrasi dengan baik dengan berbagai sumber data [9].

Platform ini menawarkan antarmuka yang intuitif dan mudah digunakan, memungkinkan pengguna untuk membuat *dashboard* interaktif dan visualisasi data yang kaya tanpa memerlukan keterampilan teknis yang mendalam. Selain itu, *Superset* bersifat *open-source*, sehingga dapat dikustomisasi sesuai kebutuhan spesifik organisasi dan menghindari biaya lisensi yang tinggi

### 5. *Apache Airflow*

*Apache Airflow* adalah sebuah platform untuk mengatur alur kerja (*workflow*) secara terprogram. Wahyudi et al. menunjukkan bahwa *Apache Airflow* dapat

digunakan untuk akuisisi data secara periodik [4]. Platform ini memungkinkan pengguna untuk menjadwalkan, memonitor, dan mengelola alur kerja data yang kompleks melalui antarmuka berbasis *web* yang intuitif. Selain itu, *Apache Airflow* mendukung integrasi dengan berbagai sistem eksternal, seperti *database*, *cloud storage*, dan layanan analitik, sehingga sangat fleksibel untuk berbagai kebutuhan pemrosesan data

## 6. *Python*

*Python* adalah bahasa pemrograman interpretatif yang berfokus pada kemudahan untuk membaca kode yang dihasilkan. *Python* memiliki beberapa keunggulan seperti sintaks yang lebih singkat serta mudah dipahami, memiliki banyak *library* pendukung, dan sistem yang dibangun dapat berjalan dengan baik pada sistem operasi *Linux*, *Windows*, maupun *Mac* [10]. Bahasa ini sangat populer di berbagai bidang seperti pengembangan *web*, analisis data, kecerdasan buatan, dan otomasi karena fleksibilitas dan komunitas pengembang yang besar. Selain itu, *Python* mendukung pemrograman berorientasi objek, fungsional, dan prosedural, sehingga cocok untuk berbagai jenis proyek pengembangan perangkat lunak.

## 7. *PostgreSQL*

*PostgreSQL* merupakan sistem basis data yang bersifat sumber terbuka, memiliki dukungan komunitas yang besar, dan mendukung penggunaan lintas platform. *PostgreSQL* dapat digunakan untuk standar *SQL* seperti *complex queries*, *transactional integrity*, dan *multiversion concurrency control* (MVCC), serta lebih efisien dalam pengelolaan data dalam skala besar

dibandingkan dengan beberapa sistem basis data lainnya [11]. Sistem ini juga mendukung berbagai tipe data canggih seperti *JSON*, *XML*, dan *array*, serta memiliki fitur *full-text search* yang powerful untuk kebutuhan pencarian teks. Selain itu, *PostgreSQL* dikenal dengan skalabilitasnya yang tinggi, mendukung replikasi data, partisi tabel, dan dapat diintegrasikan dengan berbagai bahasa pemrograman melalui *driver* yang tersedia, menjadikannya solusi ideal untuk aplikasi *enterprise* yang membutuhkan performa dan keandalan tinggi.

#### 8. *Design Thinking*

*Design Thinking* merupakan suatu pendekatan metodologis yang diterapkan sebagai strategi inovasi dalam proses perancangan solusi berbasis pengguna. Metode ini menekankan pendekatan empati untuk memahami kebutuhan pengguna secara mendalam, dengan fokus pada aspek bentuk, relasi, perilaku, interaksi, serta respons emosional manusia guna menghasilkan solusi yang optimal. Proses *Design Thinking* terdiri atas lima tahapan yaitu *emphatize*, *define*, *ideate*, *prototype*, dan *test*. Melalui penerapan kelima tahap ini, masalah kompleks yang dihadapi pengguna dapat diurai secara sistematis dan diselesaikan secara efektif [12].

#### 9. *System Usability Scale*

*System Usability Scale* (SUS) merupakan teknik penilaian kegunaan sistem yang praktis namun akurat untuk mengevaluasi tingkat kemudahan penggunaan suatu antarmuka atau sistem. Pendekatan ini menerapkan skala *Likert* lima tingkat dalam menilai sepuluh aspek yang berkaitan dengan kenyamanan penggunaan, estetika tampilan, serta tingkat kepuasan pengguna.

Kuesioner SUS memiliki kelebihan sebagai instrumen evaluasi yang telah terbukti andal dalam mengukur kualitas suatu perangkat lunak. Kuesioner SUS banyak dimanfaatkan dalam mengukur tingkat kepuasan pengguna karena keunggulannya yang memungkinkan proses pengumpulan data dilakukan secara cepat dan hemat biaya. SUS dianggap sebagai salah satu metode paling efisien untuk memperoleh data yang valid secara statistik serta menghasilkan skor yang jelas dan cukup akurat, sehingga sering disebut sebagai "*Quick and Dirty test*" [13].

Metode SUS banyak diaplikasikan dalam riset dan pengembangan produk digital seperti platform *web*, aplikasi *mobile*, maupun perangkat lunak komputer untuk mendapatkan pemahaman objektif tentang pengalaman pengguna sekaligus menyediakan acuan perbaikan yang terukur [14].

Kuesioner SUS terdiri dari 10 pernyataan dengan skala Likert 1–5 (1 = Sangat Tidak Setuju hingga 5 = Sangat Setuju). Perhitungan skor dilakukan dengan cara: untuk pernyataan bernomor ganjil (1, 3, 5, 7, 9), nilai kontribusi diperoleh dari skor jawaban dikurangi 1, sedangkan untuk pernyataan bernomor genap (2, 4, 6, 8, 10), nilai kontribusi dihitung dari 5 dikurangi skor jawaban. Seluruh skor kontribusi kemudian dijumlahkan dan hasilnya dikalikan dengan 2,5 untuk mendapatkan skor akhir SUS yang berada pada rentang 0 hingga 100.

Proses ini dapat dirumuskan sebagai berikut [13] :

$$SUS = 2,5 \times \left[ \sum_{n=1}^5 (U_{2n-1} - 1) + (5 - U_{2n}) \right]$$

Keterangan:

- $n$ = indeks set pernyataan (1 sampai 5)

- $U_{2n-1}$  = skor jawaban responden untuk pernyataan ganjil ke- $(2n - 1)$
- $U_{2n}$  = skor jawaban responden untuk pernyataan genap ke- $(2n)$
- Faktor pengali 2,5 digunakan agar skor akhir berada pada rentang 0–100.

#### 10. *Star Schema*

Star schema adalah model arsitektur data *warehouse* yang populer karena desainnya yang sederhana dan kemampuannya mengoptimalkan analisis data. Model ini memiliki struktur inti berupa tabel fakta sebagai pusatnya, dikelilingi oleh beberapa tabel dimensi yang terhubung langsung, yang berisi referensi data [15]. Tabel fakta menyimpan data numerik terukur seperti nilai transaksi atau volume penjualan, sementara tabel dimensi berisi atribut deskriptif seperti periode waktu, lokasi geografis, atau kategori produk yang memberikan konteks terhadap data fakta. Dinamakan star karena bentuk diagramnya menyerupai pola bintang dengan tabel fakta sebagai pusat dan tabel dimensi sebagai cabang-cabangnya.

Keunggulan utama *star schema* terletak pada kemudahan pembuatan *query*, peningkatan kinerja operasi agregasi data, serta dukungannya terhadap sistem *Business Intelligence*. Dalam implementasinya pada penelitian pemantauan harga pokok, tabel fakta berisi catatan harga komoditas, sedangkan tabel dimensi mencakup variabel waktu pengamatan, lokasi pasar, jenis barang, dan satuan ukuran. Pendekatan ini memungkinkan analisis multidimensi yang efisien, seperti pelacakan tren harga berdasarkan wilayah atau periode tertentu.

## 2.2. Penelitian Terkait

Dalam penelitian terkait dengan ETL data *warehouse* dengan *apache airflow* dan *python* terkait, terdapat beberapa studi yang relevan dan memberikan wawasan yang berguna dalam pengembangan solusi ETL. Beberapa penelitian terkait tersebut diuraikan sebagai berikut:

1. Penelitian pertama adalah "Akuisisi Data Prediksi Curah Hujan Secara Periodik Menggunakan *Apache Airflow*". Penelitian ini mengimplementasikan *Apache Airflow* untuk mengotomatisasi akuisisi data prediksi curah hujan dari BMKG ke komputer lokal. Metode yang digunakan meliputi desain *DAG (Directed Acyclic Graph)* dengan dua *task* utama: sensor ketersediaan data dan pengambilan data. Relevansi penelitian ini terletak pada penggunaan *Apache Airflow* untuk penjadwalan tugas periodik (setiap 12 jam) dan manajemen *log* untuk pelacakan kegagalan. Hasilnya menunjukkan bahwa *Airflow* efektif dalam menangani data *semi-real-time* dan menyediakan mekanisme *error handling* yang robust. Temuan ini dapat menjadi referensi untuk pengembangan ETL yang membutuhkan penjadwalan dan pemantauan status tugas [4].
2. Penelitian kedua adalah "Implementasi ETL pada *Data Warehouse* Penjualan Menggunakan *Tools Pentaho*". Studi ini berfokus pada proses ETL untuk integrasi data penjualan dari file *Excel* ke *data warehouse* menggunakan *Pentaho*. Transformasi data meliputi pembersihan duplikasi, standarisasi format, dan pembuatan dimensi (*Customer, Home Office*). Korelasi dengan topik ini adalah pendekatan modular dalam transformasi data. Hasil penelitian memperkuat pentingnya *data cleaning* dan konsistensi struktur sebelum dimuat ke *data warehouse* [1].

3. Penelitian ketiga adalah “Implementasi ETL *Data Warehouse* Laporan Harian *Pool*”. Penelitian ini mengintegrasikan data dari sumber heterogen (*SQL Server* dan *MySQL*) ke *data warehouse* menggunakan pendekatan skema *snowflake*. Proses ETL dijalankan dengan *SQL Server Integration Services* (SSIS) dan *Pentaho*, menekankan pada *incremental load* untuk efisiensi penyimpanan. Relevansinya terletak pada solusi untuk masalah integrasi data multi-sumber, yang dapat diadopsi dalam desain ETL berbasis *Airflow* dengan *Python* untuk kasus serupa [3].
4. Penelitian keempat adalah “Implementasi *Data Warehouse* dan BI untuk *E-Commerce* Menggunakan *Pentaho* dan *Metabase* untuk Membuat *Dashboard* Visualisasi Kinerja Penjualan *E-Commerce Wish*”. Studi ini menggabungkan ETL (*Pentaho*) dan visualisasi (*Metabase*) untuk analisis kinerja penjualan *e-commerce*. Transformasi data mencakup penghitungan *profit* dan seleksi atribut kunci. Penelitian ini relevan karena menunjukkan bagaimana ETL dapat mendukung *business intelligence*, dengan *Python* sebagai alternatif untuk logika transformasi yang lebih fleksibel [6].
5. Penelitian kelima adalah “Pengembangan *Business Intelligence Dashboard* untuk Monitoring *Key Performance Indicator* Perusahaan di WWMusik Malang”. Penelitian ini mengusulkan ETL menggunakan Talend untuk memuat data ke *data warehouse PostgreSQL*, dengan penjadwalan mingguan. Metode *four-step design* Kimball yang digunakan (pemilihan proses, *grain*, dimensi, dan fakta) dapat diadaptasi dalam desain ETL berbasis *Python*. Hasilnya menekankan pentingnya *pipeline* yang terukur dan terotomatisasi [8].
6. Penelitian keenam adalah “Sistem Propagasi Anotasi pada Metadata *Lineage* untuk Manajemen *Data Warehouse*”. Studi ini mengembangkan sistem



propagasi anotasi berbasis *lineage* untuk memudahkan pelacakan perubahan struktur data dalam ETL. Penelitian ini relevan untuk memahami manajemen metadata dan *dependency tracking* dalam *pipeline* ETL yang kompleks [16].

### 2.3. Tinjauan Organisasi/Objek Penelitian

Kementerian Perdagangan Republik Indonesia merupakan lembaga pemerintahan yang memiliki peran strategis dalam mengatur, mengembangkan, dan mengawasi kegiatan perdagangan dalam negeri dan luar negeri. Sebagai bagian dari sistem pemerintahan pusat, Kementerian Perdagangan bertugas merumuskan dan melaksanakan kebijakan nasional di bidang perdagangan, termasuk pengendalian harga barang kebutuhan pokok, perlindungan konsumen, serta fasilitasi ekspor dan impor.

Kementerian ini juga bertanggung jawab atas pengumpulan dan pengolahan data perdagangan yang mendukung pengambilan keputusan berbasis data (*data-driven decision making*). Salah satu sistem penting yang dimiliki adalah Sistem Pemantauan Pasar dan Kebutuhan Pokok (SP2KP), yang digunakan untuk memantau harga barang kebutuhan pokok di seluruh wilayah Indonesia secara harian.

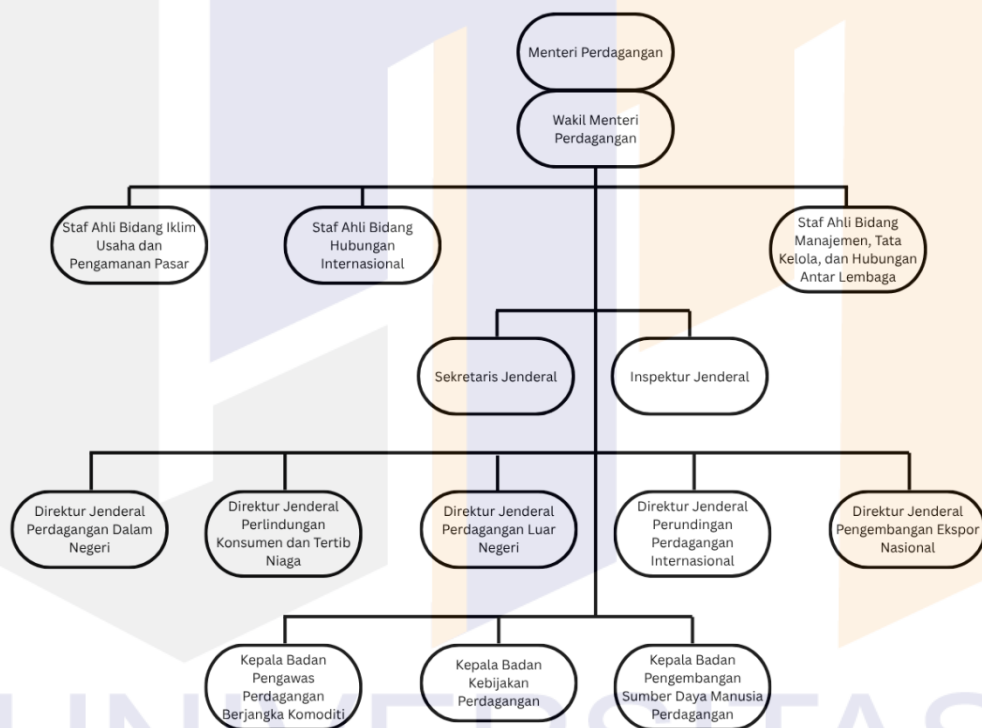
#### 2.3.1. Visi dan Misi

Visi Kementerian Perdagangan tahun 2020–2024 adalah "Perdagangan yang adil untuk mendukung pertumbuhan ekonomi yang berkualitas." [17]. Untuk mewujudkan visi tersebut, Kementerian Perdagangan menetapkan beberapa misi, antara lain:

- a. Meningkatkan daya saing dan nilai tambah produk ekspor.

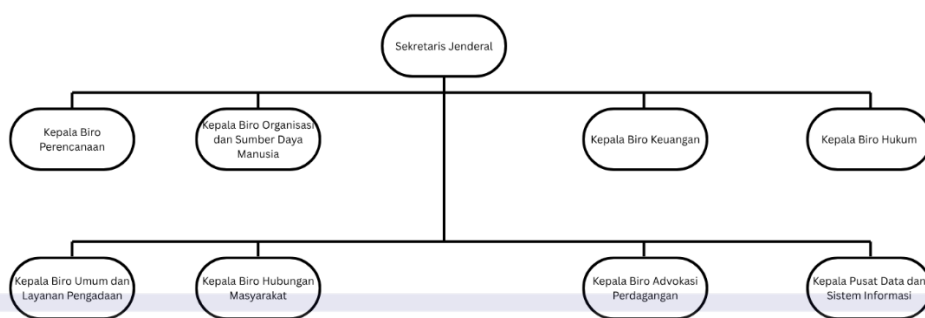
- b. Meningkatkan efisiensi dan efektivitas sistem distribusi dan logistik perdagangan.
- c. Meningkatkan perlindungan konsumen dan pengawasan perdagangan.
- d. Mendorong perdagangan dalam negeri yang sehat, adil, dan transparan.

### 2.3.2. Struktur Organisasi



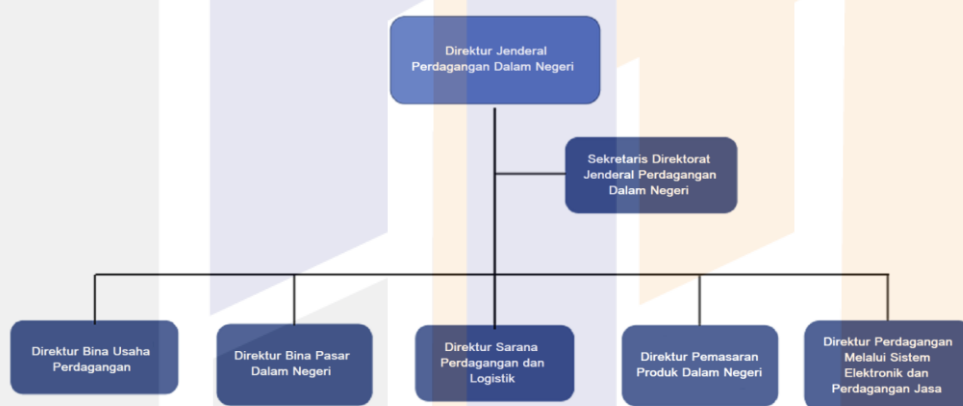
Sumber [18]

Gambar II. 1. Struktur Organisasi Kementerian Perdagangan



Sumber [18]

Gambar II. 2. Struktur Organisasi Sekretariat Jenderal



Sumber [18]

Gambar II. 3. Struktur Organisasi Direktorat Jenderal Perdagangan Dalam Negeri

Salah satu sasaran strategis Kementerian Perdagangan adalah “Terwujudnya stabilisasi harga dan ketersediaan pasokan barang kebutuhan pokok”. Untuk mendukung pencapaian sasaran tersebut, Kementerian melakukan upaya pemantauan harga secara rutin guna memastikan kestabilan dan keterjangkauan harga bagi masyarakat [17]. Pemantauan ini dilakukan melalui aplikasi Sistem Pemantauan Pasar dan Kebutuhan Pokok (SP2KP) yang menyediakan data harga harian dari berbagai daerah sebagai dasar dalam pengambilan kebijakan yang tepat dan responsif.

Aplikasi SP2KP berada di bawah pengawasan langsung Direktorat Bina Pasar Dalam negeri (Dit. Binpasdagri) sebagai pihak yang bertanggung jawab atas proses bisnis dan validitas data yang dihimpun. Direktorat ini memastikan bahwa data harga yang masuk dari seluruh daerah telah melalui verifikasi dan dapat digunakan sebagai dasar analisis kebijakan stabilisasi harga. Di sisi lain, aspek teknis seperti ketersediaan sistem, performa *server*, dan keamanan aplikasi SP2KP menjadi tanggung jawab Pusat Data dan Sistem Informasi (PDSI). PDSI bertugas memantau infrastruktur TI yang mendukung operasional SP2KP agar aplikasi dapat berjalan secara optimal dan dapat diakses dengan baik oleh petugas entri maupun pengguna internal Kementerian Perdagangan di seluruh Indonesia. Kolaborasi antara Dit. Bapokting dan PDSI ini menjadi kunci dalam menjaga keberlangsungan dan keandalan sistem pemantauan harga nasional.

Data SP2KP digunakan oleh berbagai pemangku kepentingan seperti pemerintah pusat, daerah, dan masyarakat umum untuk:

- a. Mengambil kebijakan stabilisasi harga.
- b. Menentukan kebutuhan distribusi dan pasokan barang pokok.
- c. Melakukan evaluasi ketersediaan dan keterjangkauan harga.

Sistem ini juga menjadi salah satu sumber data utama untuk *dashboard* pemantauan harga yang dibangun menggunakan *Tableau*, yang saat ini sedang dalam proses migrasi ke *Apache Superset* untuk meningkatkan efisiensi dan aksesibilitas data secara luas.