

BAB II

LANDASAN TEORI

2.1 Database

Database adalah susunan record data operasional lengkap dari suatu organisasi atau perusahaan, yang diorganisir dan disimpan secara terintegrasi dengan menggunakan metode tertentu sehingga mampu memenuhi informasi yang optimal yang dibutuhkan oleh para pengguna.

Database memiliki entitas, atribut, dan relasi. Entitas adalah objek yang berbeda yang terdapat pada sebuah database (orang, tempat, benda, konsep, atau peristiwa). Atribut adalah properti yang menjelaskan beberapa aspek dari sebuah objek yang ingin direkam. Relasi adalah hubungan antara suatu entitas dengan entitas lainnya.

2.2 Data

Data adalah sumber informasi yang bentuknya masih mentah. Menurut Jogianto (1990), data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Data merupakan bentuk jamak dari datum, berasal dari bahasa Latin yang berarti “sesuatu yang diberikan”. Dalam penggunaan sehari-hari data berarti suatu pernyataan yang diterima secara apa adanya. Pernyataan ini adalah hasil pengukuran atau pengamatan suatu variabel yang bentuknya dapat berupa angka, kata-kata, atau citra.

Dalam keilmuan (ilmiah), fakta dikumpulkan menjadi data. Data kemudian diolah sehingga dapat diutarakan secara jelas dan tepat agar dapat dimengerti oleh orang lain yang tidak langsung mengalaminya sendiri, hal ini dinamakan deskripsi.

Secara konseptual, data adalah deskripsi tentang benda, kejadian, aktivitas, dan transaksi yang tidak mempunyai makna atau tidak berpengaruh secara langsung kepada pemakai.

2.3 Informasi

Informasi adalah suatu koleksi fakta yang didapat dari data yang telah terorganisir dengan beberapa cara sehingga dapat memberikan suatu arti yang dapat dipahami oleh penerima (Turban & Rainer)

2.4 Visualisasi Data

Visualisasi data adalah teknik penyajian data berbentuk visual yang dapat memudahkan pengguna untuk memahami atau menganalisis data yang ditampilkan.

Informasi yang disajikan dalam bentuk visual mempermudah pengguna memahami informasi dari data yang tersedia, sehingga akan mempermudah proses pengambilan keputusan didalam sebuah perusahaan.

2.5 Basis data

Menurut O'Brien (2002, p.166) basis data merupakan kumpulan dari beberapa file dokumen yang terhubung secara logis.

Menurut Date (2000, p.10) basis data merupakan kumpulan data tetap (data yang hampir tidak mengalami perubahan), yang digunakan oleh sistem aplikasi di beberapa perusahaan.

Menurut Conolly & Begg (2002, p.14) basis data adalah kumpulan data yang terhubung secara logis, dan merupakan deskripsi dari data itu sendiri, yang dirancang untuk mempermudah proses pencarian informasi yang dibutuhkan oleh perusahaan.

Dari beberapa definisi tentang basis data yang telah disebutkan diatas, dapat disimpulkan bahwa basis data merupakan kumpulan data yang terhubung secara logis dan digunakan pada sistem aplikasi perusahaan, yang dapat mempermudah proses pencarian informasi.

2.5.1 Pembagian basis data menurut jenisnya:

1. Basis data flat-file.

Basis data flat-file ideal untuk data berukuran kecil dan dapat dirubah dengan mudah. Pada dasarnya, mereka tersusun dari sekumpulan string dalam satu atau lebih file yang dapat diurai untuk mendapatkan informasi yang disimpan. Basis data flat-file baik digunakan untuk menyimpan daftar atau data yang sederhana dan dalam jumlah kecil. Basis data flat-file akan menjadi sangat rumit apabila digunakan untuk menyimpan data dengan struktur kompleks walaupun dimungkinkan pula untuk menyimpan data semacam itu. Beberapa kendala dalam menggunakan basis data jenis ini adalah rentan pada korupsi data karena tidak adanya penguncian yang melekat ketika data digunakan atau dimodifikasi dan juga adanya duplikasi data yang mungkin sulit dihindari. Salah satu tipe basis data flat-file adalah file CSV yang menggunakan pemisah koma untuk setiap nilainya.

2. Basis data relasional.

Basis data ini mempunyai struktur yang lebih logis terkait cara penyimpanan. Kata "relasional" berasal dari kenyataan bahwa tabel-tabel yang berada di basis data dapat dihubungkan satu dengan lainnya. Basis data relasional menggunakan sekumpulan tabel dua dimensi yang masing-masing tabel tersusun atas baris (tupel) dan kolom (atribut). Untuk membuat hubungan antara dua atau lebih tabel, digunakan key (atribut kunci) yaitu **primary key** di salah satu tabel dan **foreign key** di tabel yang lain. Saat ini, basis data relasional menjadi pilihan karena keunggulannya. Beberapa kelemahan yang mungkin dirasakan untuk basis data jenis ini adalah implementasi yang lebih sulit untuk data dalam jumlah besar dengan tingkat kompleksitasnya yang tinggi dan proses pencarian informasi yang lebih lambat karena perlu menghubungkan tabel-tabel terlebih dahulu apabila datanya

tersebar di beberapa tabel. Beberapa contoh basis data relasional adalah Microsoft Access, MySQL, Oracle, Microsoft SQL Server dan PostgreSQL.

2.6 Data Warehouse

Menurut (Kimball, Ross, 2002) Data warehouse adalah sebuah sarana yang dapat meningkatkan kualitas pengambilan keputusan, data warehouse berisi data yang dapat mendukung proses DSS.

Menurut (O'Brien, James A, 2007), Data warehouse merupakan sumber utama data dari database lain yang telah dibersihkan, diubah, dan dikategorikan untuk analisis bisnis dan mendukung proses DSS.

Dari keterangan di atas dapat disimpulkan bahwa data warehouse adalah sebuah wadah yang dapat menampung data-data yang dibutuhkan untuk melakukan analisis suatu kondisi dalam perusahaan yang akan menghasilkan sebuah informasi yang bermanfaat yang dapat mendukung proses DSS. Data yang digunakan dalam data warehouse dapat berasal dari data internal dan eksternal

2.6.1 Karakteristik

Karakteristik data warehouse menurut Inmon yaitu :

1. Subject Oriented (Berorientasi Subjek)

Data warehouse berorientasi subjek artinya data warehouse didesain untuk menganalisa data berdasarkan subjek-subjek tertentu dalam organisasi, bukan pada proses atau fungsi aplikasi tertentu.

Data warehouse diorganisasikan disekitar subjek-subjek utama dari perusahaan (customers, products dan sales) dan tidak diorganisasikan pada area-area aplikasi utama (customer invoicing, stock control dan product sales). Hal ini dikarenakan kebutuhan dari data warehouse untuk menyimpulkan data-data yang bersifat sebagai penunjang suatu keputusan, dari pada aplikasi yang berorientasi terhadap data.

Jadi dengan kata lain, data yang disimpan adalah berorientasi kepada subjek bukan terhadap proses. Secara garis besar perbedaan antara data operasional dan data warehouse yaitu :

Tabel 2. 1 Perbedaan Data Operasional dan Data Warehouse

Data Operasional	Data Warehouse
Dirancang berorientasi hanya pada aplikasi dan fungsi tertentu.	Dirancang berdasarkan pada subjek-subjek tertentu (utama).
Fokusnya pada desain database dan proses.	Fokusnya pada pemodelan data dan desain data.
Berisi rincian atau detail data.	Berisi data-data history yang akan dipakai dalam proses analisis.
Relasi antar tabel berdasar aturan terkini (selalu mengikuti rule(aturan) terbaru).	Banyak aturan bisnis tersaji antara tabel-tabel.

2. Integrated (Terintegrasi)

Data warehouse dapat menyimpan data-data yang berasal dari sumber-sumber yang terpisah kedalam suatu format yang konsisten dan saling terintegrasi satu dengan lainnya. Dengan demikian data tidak bisa dipecah-pecah karena data yang ada merupakan suatu kesatuan yang menunjang keseluruhan konsep data warehouse itu sendiri.

Syarat integrasi sumber data dapat dipenuhi dengan berbagai cara seperti konsisten dalam penamaan variabel, konsisten dalam ukuran variabel, konsisten struktur pengkodean dan konsisten dalam atribut fisik dari data.

3. Time Variant (Rentang Waktu)

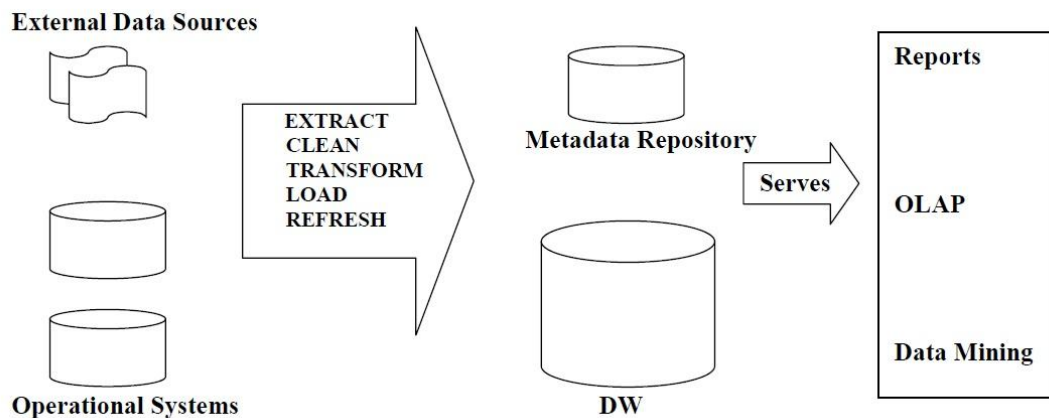
Seluruh data pada data warehouse dapat dikatakan akurat atau valid pada rentang waktu tertentu. Untuk melihat interval

waktu yang digunakan dalam mengukur keakuratan suatu data warehouse, kita dapat menggunakan cara antara lain :

- a. Cara yang paling sederhana adalah menyajikan data warehouse pada rentang waktu tertentu, misalnya 5 sampai 10 tahun ke depan.
 - b. Cara yang kedua, dengan menggunakan variasi atau perbedaan waktu yang disajikan dalam data warehouse baik implisit maupun eksplisit secara eksplisit dengan unsur waktu dalam hari, minggu, bulan dsb. Secara implisit misalnya pada saat data tersebut diduplikasi pada setiap bulan, atau per tiga bulan. Unsur waktu akan tetap ada secara implisit didalam data tersebut.
 - c. Cara ketiga, variasi waktu yang disajikan data warehouse melalui serangkaian snapshot yang panjang. Snapshot merupakan tampilan dari sebagian data tertentu sesuai keinginan pemakai dari keseluruhan data yang ada bersifat read-only.
4. Non Volatile

Karakteristik keempat dari data warehouse adalah non volatile, maksudnya data pada data warehouse tidak di-update secara real time tetapi di refresh dari sistem operasional secara regular. Data yang baru selalu ditambahkan sebagai suplemen bagi database itu sendiri dari pada sebagai sebuah perubahan. Database tersebut secara kontinu menyerap data baru ini, kemudian secara incremental disatukan dengan data sebelumnya.

2.6.2 Arsitektur Data Warehouse



Gambar 2. 1 Arsitektur Data Warehouse

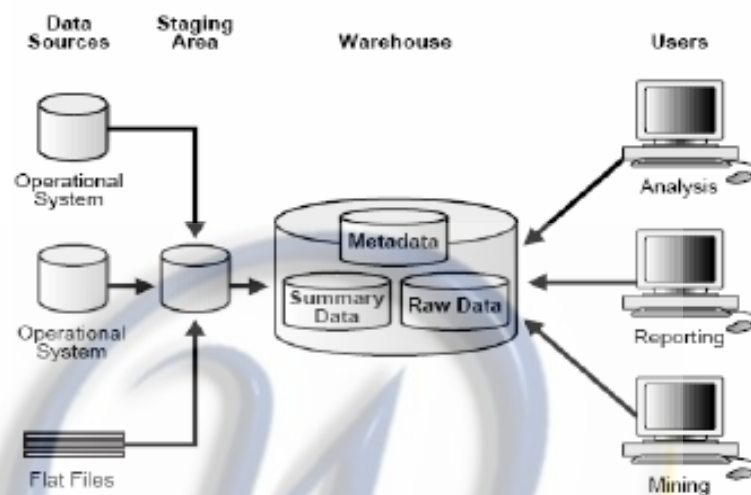
Sumber :

lessy.lecturer.pens.ac.id/kuliah/datawarehouse/materiUASdw.pdf/arsitekturdw.jpg

Data yang pada data warehouse berasal dari sistem operasional dan juga dari sumber lain dari luar (external sources). Semua sumber data ini disebut dengan source systems. Data diekstrak dari source system lalu disimpan dalam suatu tempat yang dinamakan data staging area, dimana data dibersihkan, ditransformasikan, dikombinasikan, diduplikasi untuk dapat siap di-load kedalam data warehouse. Pada data staging area juga terdapat aktifitas seperti pengurutan. Bagian dari sistem yang menyediakan query dan layanan presentasi disebut dengan presentastion server. Merupakan mesin target dimana data di-load dari data staging area kemudian disimpan untuk dilakukan query secara langsung oleh end users, penyiapan laporan dan aplikasi yang lain. Pada dasarnya data warehouse membutuhkan tiga sistem yang berbeda yaitu :

1. Source systems
2. Data staging area
3. Presentation

Fungsi dari tiap sistem telah dijelaskan sebelumnya.



Gambar 2. 2 Arsitektur Data Warehouse dengan Staging Area (Paul Lane, 2002)

Pada gambar 2.2 terlihat proses pengolahan data operasional sebelum dimasukkan ke dalam data warehouse melalui staging area terlebih dahulu. Staging area digunakan untuk memudahkan dalam melakukan integrasi dan pembersihan data sehingga dapat menghasilkan data yang berkualitas. Karena di dalam staging area terdapat proses untuk melakukan penggabungan data, pembersihan, dan standarisasi data.

2.6.3 Pemodelan Dimensional

Pemodelan dimensional adalah suatu nama dari teknik desain logika yang digunakan untuk data warehouse. Berbeda dengan entity relational modeling (ER Modeling) yang digunakan untuk menyimpan data transaksi pada system OLTP.

Tujuan dari dimensional modelling ini adalah untuk mempresentasikan suatu set pengukuran bisnis dalam kerangka standar sehingga dapat lebih mudah dipahami oleh end user. Dimensional model

berisi informasi yang mirip dengan ER Model akan tetapi data dalam format symmetric yang didesain untuk :

- User understandability
- Query performance
- Resilience to change

Komponen utama dari dimensional modelling ini adalah tabel fakta dan tabel dimensi. Tabel fakta adalah primary tabel yang dalam beberapa dimensi model yang dimaksudkan untuk pengukuran suatu bisnis.

Suatu fakta bergantung pada beberapa faktor, sebagai contoh sale amount, suatu fakta, yang bergantung pada produk, lokasi dan waktu. Faktor faktor ini adalah yang disebut dimensi.

2.6.4 Skema Star

Struktur skema bintang adalah suatu struktur yang dapat dengan mudah dipahami dan digunakan oleh pengguna. Dalam skema bintang terdapat dua jenis tabel, yaitu tabel fakta dan tabel dimensi yang berisi referensi data.

Tabel fakta pada skema bintang dihasilkan dari event-event yang terjadi dimasa lampau, fakta yang paling berguna dalam tabel fakta adalah perhitungan numeric, atau fakta yang terjadi dari setiap record. Sedangkan tabel dimensi mengandung informasi tekstual yang deskriptif.

2.6.5 Perancangan Data Warehouse

(Connolly dan Begg, 2002) Terdapat sembilan tahap metodologi dalam membangun data warehouse yang dikenal dengan nine-step methodology yang terdiri dari:

1. Memilih Proses (Choosing The Process)
2. Memilih Sumber (Choosing The Grain)
3. Mengidentifikasi dan Penyesuaian Dimensi (Identifying and Conforming Dimensions)
4. Memilih Fakta (Choosing The Fact)
5. Menyimpan Perhitungan Awal dalam Tabel Fakta (Storing Pre-calculation in The Fact Table)
6. Melihat Kembali Table Dimensi (Rounding Out The Dimension Tables)
7. Memilih Durasi Database (Choosing The Duration of Database)
8. Menelusuri Perubahan dari Dimensi secara Perlahan (Tracking Slowly Changing Dimension)
9. Memutuskan Prioritas Query dan Tipe Query (Deciding The Query Prorities and The Query Models)

2.6.6 Data Mart

(inmon, 2005) yang dimaksud dengan data mart adalah struktur data per departement yang berasal dari data warehouse dilakukan denormalisasi berdasarkan kebutuhan informasi dari setiap departement.

(poniah, 2001) data mart adalah data warehouse kecil yang dirancang untuk tiap departement. Keuntungan dari data mart adalah biayanya yang murah dan waktu penerapannya relatif cepat.

2.7 Dashboard

Dashboard adalah tampilan visual dari informasi yang paling penting yang diperlukan untuk memahami dan mengelola satu atau lebih bidang organisasi yang diatur pada satu layar komputer sehingga dapat dipantau dengan mudah (Stephen Few, 2006).

Dashboard adalah sebuah koleksi data informasi yang digunakan untuk melakukan evaluasi performa dan memastikan tercapainya tujuan dari sebuah operasi (Ronald R, 2010).

Dari pengertian di atas, maka sebuah dashboard harus dapat mendukung proses monitoring secara visual seperti:

1. Melihat gambaran besar informasi.
2. Fokus pada item tertentu yang memiliki informasi penting.
3. Dapat melakukan drill ke informasi tambahan yang diperlukan untuk proses pengambilan keputusan.

2.8 Online Transaction Processing (OLTP)

OLTP adalah sebuah sistem yang dirancang untuk dapat melakukan pemrosesan transaksi yang secara umum dapat melakukan perubahan kecil pada data operasional sebuah organisasi, yang bertujuan untuk menangani operasi sehari-hari (Connolly dan Begg, 2005).

2.9 UML

Nugroho (2010:6), UML (Unified Modeling Language) adalah ‘bahasa’ pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi objek’. Pemodelan (modeling) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

Nugroho (2009:4), UML (Unified Modeling Language) adalah Metodologi kolaborasi antara metoda-metoda Booch, OMT (Object Modeling Technique), serta OOSE (Object Oriented Software Engineering) dan beberapa metoda lainnya, merupakan metodologi yang paling sering digunakan saat ini untuk analisa dan perancangan sistem dengan metodologi berorientasi objek mengadaptasi maraknya penggunaan bahasa “pemrograman berorientasi objek” (OOP).

2.9.1 Model UML (Unified Modeling Language)

Menurut Widodo (2011:10), “Beberapa literature menyebutkan bahwa UML menyediakan sembilan jenis diagram, yang lain menyebutkan delapan karena ada beberapa diagram yang digabung,

misanya diagram komunikasi, diagram urutan dan diagram pewaktuan digabung menjadi diagram interaksi”. Namun demikian model-model itu dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis. Jenis diagram itu antara lain:

1. Diagram kelas (Class Diagram)

Bersifat statis, Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif.

2. Diagram paket (Package Diagram)

Bersifat statis. Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian dari diagram komponen.

3. Diagram use-case (Usecase Diagram)

Bersifat statis. Diagram ini memperlihatkan himpunan use-case dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.

4. Diagram interaksi dan sequence (Sequence Diagram)

Bersifat dinamis. Diagram urutan adalah iterasiksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.

5. Diagram komunikasi (Communication Diagram)

Bersifat dinamis. Diagram sebagai pengganti diagram kolaborasi UML yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.

6. Diagram statechart (Statechart Diagram)

Bersifat dinamis. Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status (state), transisi, kejadian serta aktivitas.

7. Diagram aktivitas (Activity Diagram)

Bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek.

8. Diagram komponen (Component Diagram)

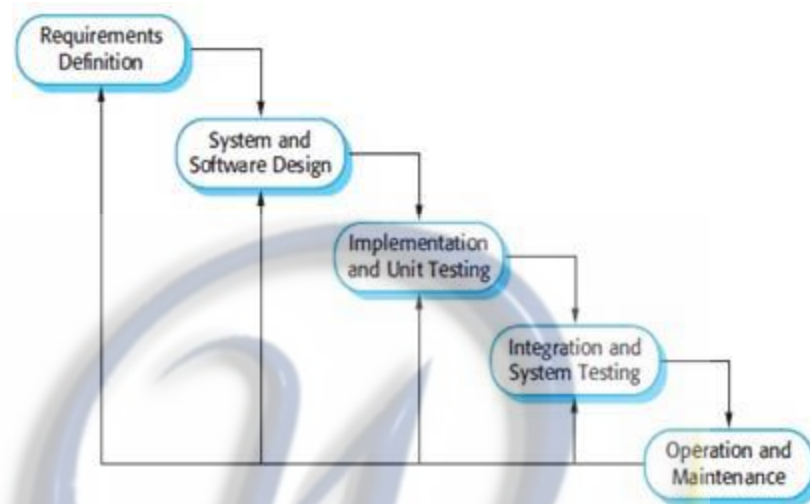
Bersifat statis. Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya.

9. Diagram deployment (deployment diagram)

Bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (run-time). Memuat simpul-simpul beserta komponen-komponen yang di dalamnya. Kesembilan diagram ini tidak mutlak harus digunakan dalam pengembangan perangkat lunak, semuanya dibuat sesuai kebutuhan. Pada UML dimungkinkan kita menggunakan diagram-diagram lainnya misalnya data flow diagram, entity relationship diagram, dan sebagainya.

2.10 Waterfall Model

Menurut (I Sommerville, 2011) terdapat 5 tahapan pada waterfall model, yaitu requirement analysis and definition, system and software design, implementation and unit testing, integration and system testing, dan operation and maintenance.



Gambar 2. 3 metode waterfall (I Sommerville, 2011)

1. Requirement Analysis and Definition

Merupakan tahapan penetapan fitur, kendala, serta tujuan sistem melalui proses wawancara yang dilakukan dengan pengguna sistem. Proses ini bertujuan untuk menentukan spesifikasi sistem yang akan dibangun.

2. Sytem and Software Design

Tahapan ini merupakan tahapan pembentukan suatu arsitektur sistem berdasarkan yang telah ditentukan pada proses sebelumnya. Tahapan ini berfungsi sebagai gambaran abstraksi dasar sistem perangkat lunak dan hubungan-hubungannya.

3. Implementation and System Testing

Pada tahapan ini dilakukan proses implementasi desain yang telah dibuat pada tahapan sebelumnya kedalam bentuk program dengan menggunakan bahasa pemrograman. Setiap unit akan dilakukan pengujian untuk memastikan semua fungsi dengan yang diharapkan.

a. Integration and System Testing

Pada tahapan ini setiap unit program akan dilakukan pengintegrasian antara satu dengan yang lainnya, serta dilakukan proses ujicoba sebagai satu sistem yang utuh untuk memastikan sistem yang dibangun telah memenuhi persyaratan yang ada.

b. Operation and Maintenance

Tahapan ini merupakan tahapan terakhir dimana pada tahapan ini dilakukan proses penggunaan sistem. Pada tahapan ini dapat dilakukan perbaikan apabila ada error ditemukan.

2.11 PHP (PHP Hypertext Preprocessor)

(A Saputra, 2011) yang dimaksud dengan PHP adalah suatu bahasa pemrograman yang berfungsi untuk membangun suatu website dinamis. PHP bersifat open source sehingga dapat dipakai secara cuma-cuma dan mampu berjalan pada sistem operasi windows maupun linux. Adapun keunggulan yang dimiliki PHP yaitu :

1. Mudah untuk dipelajari
2. PHP dapat diaplikasikan ke berbagai Operasi Sistem dan dapat diakses oleh banyak web browser
3. Memiliki tingkat akses yang cepat
4. Bersifat open source
5. Didukung oleh beberapa web server, seperti apache, IIS, lighttod, xitami.

6. Dapat mendukung database, baik yang gratis maupun yang berbayar, seperti MySQL, PostgreSQL, mSQL, Infomix, SQL Server, Oracle

2.12 High chart

High chart adalah sebuah charting library yang ditulis dengan menggunakan bahasa pemrograman javascript . High chart menawarkan grafik yang interaktif yang dapat digunakan pada aplikasi berbasis web. Saat ini highchart mendukung grafik tipe line, spline, area, arespline, column, bar, pie, dan scatter. Highchart dapat berjalan di hampir semua browser.

2.13 Query

Query adalah kemampuan untuk menampilkan suatu data dari database dimana data diambil dari tabel-tabel yang ada di dalam database sesuai dengan kebutuhan. Bahasa query merupakan bahasa khusus yang digunakan untuk melakukan query pada basis data. Contoh penggunaan bahasa query adalah `SELECT * FROM SERVICE WHERE KD_TIME=2`. Query tersebut meminta untuk menampilkan semua record yang terdapat pada tabel service dengan `KD_TIME=2`.

2.14 Pentaho

Pada tahun 2001, matt casters membuat sebuah perangkat ETL yang diberi nama kettle yang merupakan singkatan dari KDE ETTL Environment, karena pada awalnya perangkat ini ditujukan untuk dapat dijalankan di atas K Desktop Environment (KDE) linux. ETTL pada kettle merupakan singkatan dari Extraction, Transpormation, Transformation, dan Loading.

Karena kettle memiliki beragam keunggulan disbanding perangkat ETL open source lainnya, pada april 2006, kode sumber kettle resmi diambil oleh pentaho. Oleh pentaho, kettle diberi nama pentaho data integration (PDI).

2.12.1 Step

Step merupakan blok bangunan inti dari transformation. Masing-masing step memiliki fungsi dan tugas tertentu, misalkan ada step yang berfungsi membaca file excel, ada step yang berfungsi untuk memilih field, ada step yang berfungsi untuk melakukan proses insert/update data ke dalam database, dan lain sebagainya.

Kettle memiliki ratusan step siap pakai, dalam spoon setiap step diawali oleh ikon-ikon unik. Antara step yang satu dengan step yang lainnya dihubungkan menggunakan transformation hop.

