BAB II

TINJAUAN PUSTAKA

2.1. Profil Tempat Kerja Praktek

Pusat Penelitian (PUSLIT) Informatika merupakan bagian dari Lembaga Imu Pengetahuan Indonesia (LIPI) Bandung yang menangani riset nasional pada bidang informatika. Pusat Penelitian Informatika mengabdi untuk kepentingan masyarakat dan menyelenggarakan kegiatan di bidang Informatika dan Ilmu Pengetahuan Komputer untuk pembangunan berkelanjutan.

Maknanya bahwa berbagai kegiatan penelitian yang dilakukan di bidang Informatika adalah untuk kepentingan masayarakat, antara lain untuk mencerdaskan kehidupan bangsa dengan menggunakan kemampuan teknologi informasi yang baru. Teknologi ini digunakan untuk pengumpulan, pengolahan, penyimpanan dan penyebarluasan data atau informasi kepada masyarakat dengan cepat, tepat waktu dan akurat. Sehingga akan terjadi percepatan peningkatan kemampuan masyarakat dalam pembangunan yang berkelanjutan.

Lembaga Ilmu Pengetahuan Indonesia (LIPI) Bandung berkedudukan di Jl. Jalan Cisitu No 21/154D Bandung 40135. Untuk Pusat Penelitian (PUSLIT) Informatika sendiri berlokasi di dalam Komplek LIPI Gedung 20 lantai 3.

2.1.1. Sejarah Instansi

Pusat Penelitian Informatika semula bernama Pusat Penelitian Informatika dan Ilmu Pengetahuan Komputer (Puslitbang Inkom) yang didirikan pada Tahun

1986, kemudian melihat peran, urgensi, serta kontribusi dan sejalan dengan kebutuhan organisasi Lembaga Ilmu Pengtahuan Indonesia (LIPI), maka Puslitbang Inkom berubah nama menjadi Pusat Penelitian Informatika yang berada dibawah kedeputian Ilmu Pengetahuan Teknik (Deputi IPT-LIPI). LIPI sendiri merupakan Lembaga Pemerintah Non Departemen (LPND) yang barada dibawah dan bertanggung jawab langsung kepada Presiden Republik Indonesia.

Pusat Penelitian (PUSLIT) Informatika menangani riset pada bidang informatika khususnya mempunyai Visi untuk "Menjadi acuan nasional pada bidang informatika dan Ilmu Pengetahuan Komputer" yang mempertimbangkan eksistensi bangsa dengan terciptanya kamandirian dan keunggulan pada bidang ICT (*Information and Communication Technologi)* dimasa yang akan datang dengan mengabdi untuk kepentingan masyarakat dengan menyelenggarakan kegiatan dibidang Informatika untuk pembangunan yang berkelanjutan.

2.1.2. Logo Instansi

Adapun Logo Pusat Penelitian (PUSLIT) Informatika LIPI bandung ialah :



Gambar 2.1 Logo Instansi

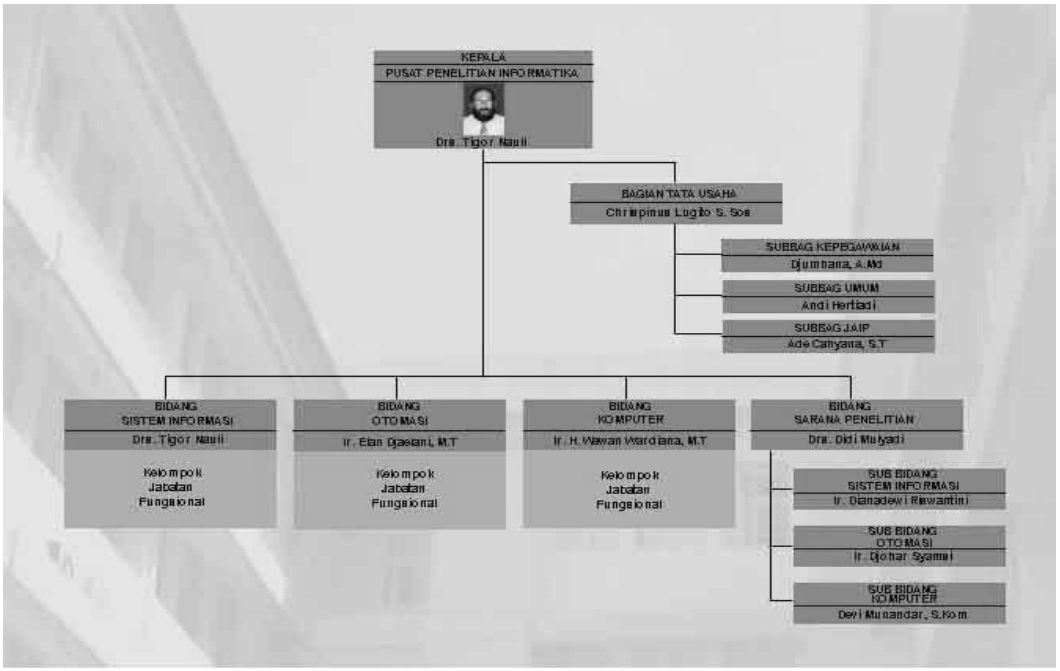
2.1.3 Badan Hukum Instansi

Penetapan Pusat Penelitian Informatika sendiri (Puslit Informatika - LIPI) melalui Surat Keputusan Kepala LIPI No.1151/M/201, tanggal 5 Juni 2001.

2.1.4 Struktur Organisasi dan Job description Instansi

Struktur Organisasi Pusat Penelitian Informatika – LIPI terdiri dari :

1. Bagian Tata Usaha
2. Bagian Sistem Informasi
3. Bidang Otomasi
4. Bidang Komputer
5. Bidang Sarana Penelitian



Gambar 2.2 Struktur Organisasi Pusat Penelitian Informatika

Gambar struktur Organisasi Puslit di atas berdasarkan Sk. Kepala LIPI No.1151/M/2001, tanggal 5 Juni 2001 terdiri dari satu Kepala Puslit, eseion 2a, satu Kepala Tata Usaha, eseion 3a, empat Kepala bidang Komputer dan Kepala Bidang Sistem Informasi, Kepala Bidang Otomasi, Kepala Bidang Komputer dan Kepala Bidang Sarana Penelitian.

Bagian Tata Usaha membawahi tiga Kepala Subbagian, eselon 4 meliputi:

- Subbagian Kepegawaian
- Subbagian Umum
- Subbagian Jasa Dan Informasi.

Bidang Sarana Penelitian membawahi tiga Kepala Subbid, eselon 4a, meliputi:

- Subbid Sarana Sistem Informasi
- Subbid Sarana Otomasi
- Subbid Sarana Komputer

2.2 Landasan Teori

Teori-teori yang bersangkutan dengan sistem yang akan dibangun akan dijelaskan di bawah ini :

2.2.1 Sistem Katalog

2.2.1.1 Definisi

Banyak definisi yang benar tentang sistem, mulai dari yang paling abstrak ke yang paling konkret, tetapi definisi praktisnya yang berguna adalah:

> **"Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu."**

Menurut ilmu perpustakaan Fathmi (2004), katalog perpustakaan dapat didefinisikan sebagai berikut.

> **"Katalog merupakan daftar dari koleksi perpustakaan atau beberapa perpustakaan yang disusun secara sistematis, sehingga memungkinkan pengguna perpustakaan dapat mengetahui dengan mudah koleksi apa yang dimiliki oleh perpustakaan dan dimana koleksi tersebut dapat ditemukan."**

Sehingga Sistem Katalog dapat didefinisikan sebagai berikut :

> **"Sistem katalog adalah suatu sistem pengelompokan bahan pustaka berdasarkan jenis, ukuran (tinggi, pendek, besar, dan kecil), warna, abjad judul, abjad pengarang *(klasifikasi artificial*) serta subyeknya (*klasifikasi fundamental*)."**

2.2.2 Pengembangan Perangkat Lunak

Untuk menyelesaikan masalah aktual di dalam sebuah seting industri, rekayasa perangkat lunak atau tim perekayasa harus menggabungkan strategi pengembangan yang melingkupi lapisan proses, metode, dan alat-alat bantu serta fase-fase generik. Strategi ini sering di acukan sebagai model proses atau paradigm rekayasa perangkat lunak.

Sebuah perangkat lunak yang disimpan atau digunkan dalam jangka waktu yang cukup lama akan menimbulkan masalah apabila sering diabaikan atau tidak diperhatikan perkembangannya. Permasalahan yang akan timbul dari penggunaan perangkat lunak yang cukup lama adalah:

1. Adanya kebutuhan baru, sebagai contoh pada saat ini berkembangnnya perangkat lunak berbasis web.
2. Untuk mengurangi kompleksitas, biaya serta waktu pemasaran. Sebagai contoh pemanfaatan dari perkembangan bahasa pemograman tingkat tinggi.
3. Mengurangi cacat. Sebagai contoh dengan cara membuat standarisasi dari pengkodean yang tidak sesuai.

Untuk dapat menyelesaikan permasalahan yang timbul diatas, maka diperlukannya melakukan evolusi dari perangkat lunak tersebut yang terdiri dari:

1. Pembangunan Perangkat Lunak

   Yang termasuk dalam pembangunan perangkat lunak itu sendiri meliputi dari daur hidup perangkat lunak, yaitu : permintaan (*requirement*), spesifikasi, perancangan (*design*), pengujian (*testing*) dan sebagainya.

2. Perawatan Perangkat Lunak

   Merupakan prosesproses untuk memperpanjang waktu penggunaan sistem perangkat lunak yang ada, sehingga tetap dapat dipergunakan sebagaimana mestinya.

3. Migrasi Perangkat Lunak

   Merupakan proses-proses yang memindahkan sistem yang ada ke sistem yang baru dikarenakan perkembangan dari kebutuhan dari perangkat lunak tersebut.

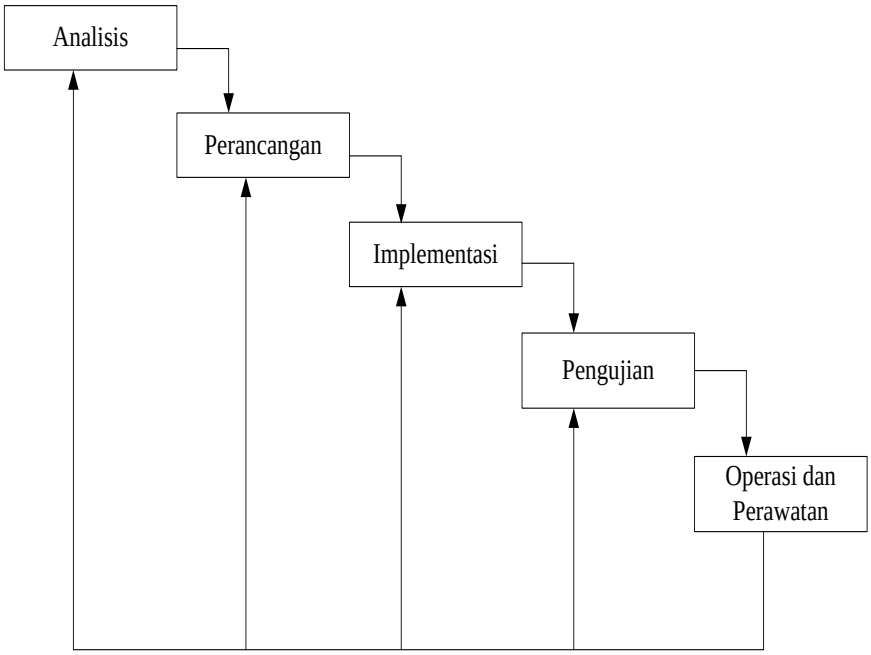
2.2.3 Model Proses Perangkat Lunak

Model proses perangkat lunak merupakan representasi abstrak dari proses perangkat lunak. Setiap model proses merepresentasikan suatu proses dari sudut pandang tertentu sehingga hanya memberikan informasi parsial mengenai proses tersebut.

Racoon mengusulkan sebuah “model Chaos” yang menggambarkan “perkembangan perangkat lunak sebagai kesatuan dari pemakai ke pengembang dan teknologi.” Masing-masing model sudah ditandai dengan cara tertentu sehingga (diharapkan) bisa membantu di dalam control dan koordinasi dari proyek perangkat lunak yang nyata. Dengan demikian, pada intinya, semua model menunjukkan karakteristik dari model Chaos. Adapun model proses perangkat lunak yang dugunakan pada sistem yang dibangun ialah :

2.2.3.1 Model *Waterfall* (Model Air Terjun/Model Sekuensial Linier)

Model *waterfall* adalah paradigm rekayasa perangkat lunak yang paling luas dipakai dan paling tua. Model *waterfall* mengusulkan sebuah pendekatan kepada perkembangan perangkat lunak yang sistematik dan sekuensial yang dimulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, pengujian, dan pemeliharaan. Setelah setiap tahap didefinisikan, tahap tersebut 'diakhiri' (*signed off*) dan pengembangan berlanjut ke tahap berikutnya.



Gambar 2.3 Model *waterfall*

2.2.4 Konsep Merancang Basis Data

2.2.4.1 Definisi

Istilah basis data banyak menimbulkan interpretasi yang berbeda. **Anthoni J. Fabbri** dan **A. Robert Schwab**, mendefinisikan basis data sebagai berikut :

> **"Basis data adalah sistem berkas terpadu yang dirancang terutama untuk menimalkan pengulangan data"**

Menurut **George Tsu-der Chou**, basis data dapat didefinisikan sebagai berikut :

> **"Basis data sebagai kumpulan informasi bermanfaat yang diorganisasikan ke dalam tatacara yang khusus".**

Basis data dimaksudkan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas.

Sistem basis data adalah suatu sistem menyusun dan mengelola record-record menggunakan komputer untuk menyimpan atau merekam serta memelihara data operasional lengkap sebuah organisasi/perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk proses mengambil keputusan.

Untuk mengelola basis data diperlukan perangkat lunak yang disebut DBMS. DBMS adalah perangkat lunak sistem yang memungkinkan para pemakai membuat, memelihara, mengontrol dan mengakses basis data dengan cara yang praktis dan efisien.

2.2.4.2 Bahasa Basis Data

Semua DBMS memiliki dua macam bahasa yang digunakan untuk mengelola dan mengorganisasikan data, yaitu :

1. Bahasa Definisi Data (DDL/Data Definition Language)
2. Bahasa Manipulasi Data (DML/Data Manipulation Language)

Selain kedua bahasa di atas, pada kebanyakan DBMS sekarang juga terdapat bahasa yang tergolong DCL (Data Control Language)yang berkaitan dengan peraturan sekuritas basis data.

A. Bahasa Definisi Data (DDL)

DDL adalah perintah-perintah yang biasa digunakan oleh administrator basis data (DBA) untuk mendefinisikan skema ke DBMS. **Skema** adalah deskripsi lengkap tentang struktur medan, rekaman, dan hubungan data pada basis data pada basis data. Tugas utama skema adalah menjabarkan struktur basis data kepada DBMS. Hasil Kompilasi dari pernyataan-pernyataan DDL disimpan dalam berkas-berkas special yang disebut **katalog sistem**.

Pernyataan SQL seperti CREATE TABLE dan DROP TABLE merupakan contoh perintah yang termasuk dalam kategori DDL.

B. Bahasa Manipulasi Data (DML)

DML adalah perintah-perintah yang digunakan untuk mengubah, memanipulasi, dan mengambil data pada basis data. Tindakan seperti menghapus,

mengubah, dan mengambil data menjadi bagian dari DML. Pada dasarnya DML dibagi menjadi dua:

1. DML Prosedural adalah perintah-perintah yang memungkinkan pengguna menentukan data apa saja yang diperlukan dan bagaimana cara mendapatkannya.
2. DML Nonprosedural adalah perintah-perintah yang memungkinkan pengguna menentukan data apa saja yang diperlukan, tetapi tidak perlu menyebutkan cara mendapatkannya.

2.2.4.3 Tahap Perancangan Basis Data

Perancangan basis data merupakan langkah awal untuk menentukan basis data yang diharapkan dapat mewakili seluruh kebutuhan pengguna. Perancangan basis data terdiri atas perancangan basis data secara konseptual, perancangan basis data secara logis, dan perancangan basis data secara fisis.

Beberapa komponen yang terdapat pada perancangan basis data secara konseptual antara lain:

1. **Entitas**

   Entitas terkadang disebut tipe entitas atau kelas entitas. Entitas adalah objek yang dapat dibedakan dari objek-objek lainnya.

2. **Atribut**

   Atribut adalah item data yang menjadi bagian dari suatu entitas. Istilah lain dari atribut adalah **properti**.

3. Hubungan

Hubungan adalah asosiasi atau kaitan antara dua entitas. Hubungan dapat berupa:

a. 1-1, yakni satu data pada suatu tabel berpasangan dengan hanya satu data pada tabel lain.
b. 1-N, yakni satu data pada suatu tabel berpasangan dengan banyak data pada tabel lain.
c. N-N, yakni banyak data pada suatu tabel berpasangan dengan banyak data pada tabel lain.

4. Kekangan

Kekangan digunakan untuk melindungi integritas data (misalnya, melindungi kesalahan sewaktu pengisian data).

5. Domain

Domain adalah himpunan yang berlaku bagi suatu atribut. Kekangan domain mendefinisikan nama, tipe, format, panjang, dan nilai masing-masing *item* data.

6. **Integritas Referensial**

Integritas referensial adalah aturan-aturan yang mengatur hubungan antara kunci primer dengan kunci tamu milik tabel-tabel yang berbeda dalam suatu basis data relasional untuk menjaga konsistensi data.

2.2.5 Pemodelan Analisis

Model analisis, yang sebenarnya merupakan serangkaian model, merupakan representasi teknis yang pertama dari sistem. Tetapi saat ini ada dua yang mendominasi landasan pemodelan analisis. Yang pertama, analisis terstruktur adalah metode pemodelan klasik, dan analisis berorientasi objek.

Analisis terstruktur adalah aktivitas pembangunan model. Dengan menggunakan notasi yang sesuai dengan prinsip analisis operasional dapat menciptakan model yang menggambarkan muatan dan aliran informasi (data kontrol), membagi sistem secara fungsional dan secara *behavioral*, dan menggambarkan esensi dari apa yang harus dibangun.

Entity-relationship Diagram adalah notasi yang digunakan untuk melakukan aktivitas pemodelan data. Atribut dari masing-masing objek data yang ditulis pada ERD dapat digambarkan dengan menggunakan deskripsi objek data. Sedangkan *data flow diagram* (DFD) memberikan informasi tambahan yang digunakan selama analisis domain informasi dan berfungsi sebagai dasar bagi pemodelan fungsi.

2.2.5.1 Entity Relationship Diagram (ERD)

ERD hanya berfokus pada data, dengan menunjukkan "jaringan data" yang ada untuk suatu sistem yang diberikan. ERD sangat berguna bagi aplikasi di mana data dan hubungan yang mengatur data sangatlah kompleks. ERD pada mulanya di usulkan oleh **Peter Chen** untuk desain sistem *database* relasional dan telah dikembangkan oleh uang lainnya. Sedangkan komponen utama diidentifikasikan

untuk ERD: objek data, atribut, hubungan, dan berbagai tipe indicator. Tujuan utama dari ERD adalah untuk mewakili objek data dan hubungan mereka.

2.2.5.2 Data Flow Diagram (DFD)

Data FlowDiagram (DFD-DAD/Diagram Alir Data) memperlihatkan hubungan fungsional dari nilai yang dihitung oleh sistem, termasuk nilai masukan, nilai keluaran, serta tempat penyimpanan internal. DAD adalah gambaran grafis yang memperlihatkan aliran data dari sumbernya dalam objek kemudian melewati proses yang mentransformasinya ke tujuan lain, yang ada pada objek lain. DAD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir. DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur (*structured analysis and design*). DFD merupakan alat yang cukup popular sekarang ini, karena dapat menggambarkan arus data di dalam sistem dengan terstruktur jelas.

Beberapa symbol yang digunakan dalam *Data Flow Diagram* (DFD) antara lain:

1. ***External Entity*** **(Kesatuan Luar) atau** ***Boundary*** **(Batas Sistem)**

   Setiap sistem pasti mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan sistem luarnya. Sistem akan menerima *input* dan *output* kepada lingkungan luarnya. Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat

berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan *input* atau menerima *output* dari sistem.

2. *Data Flow* (Arus Data)

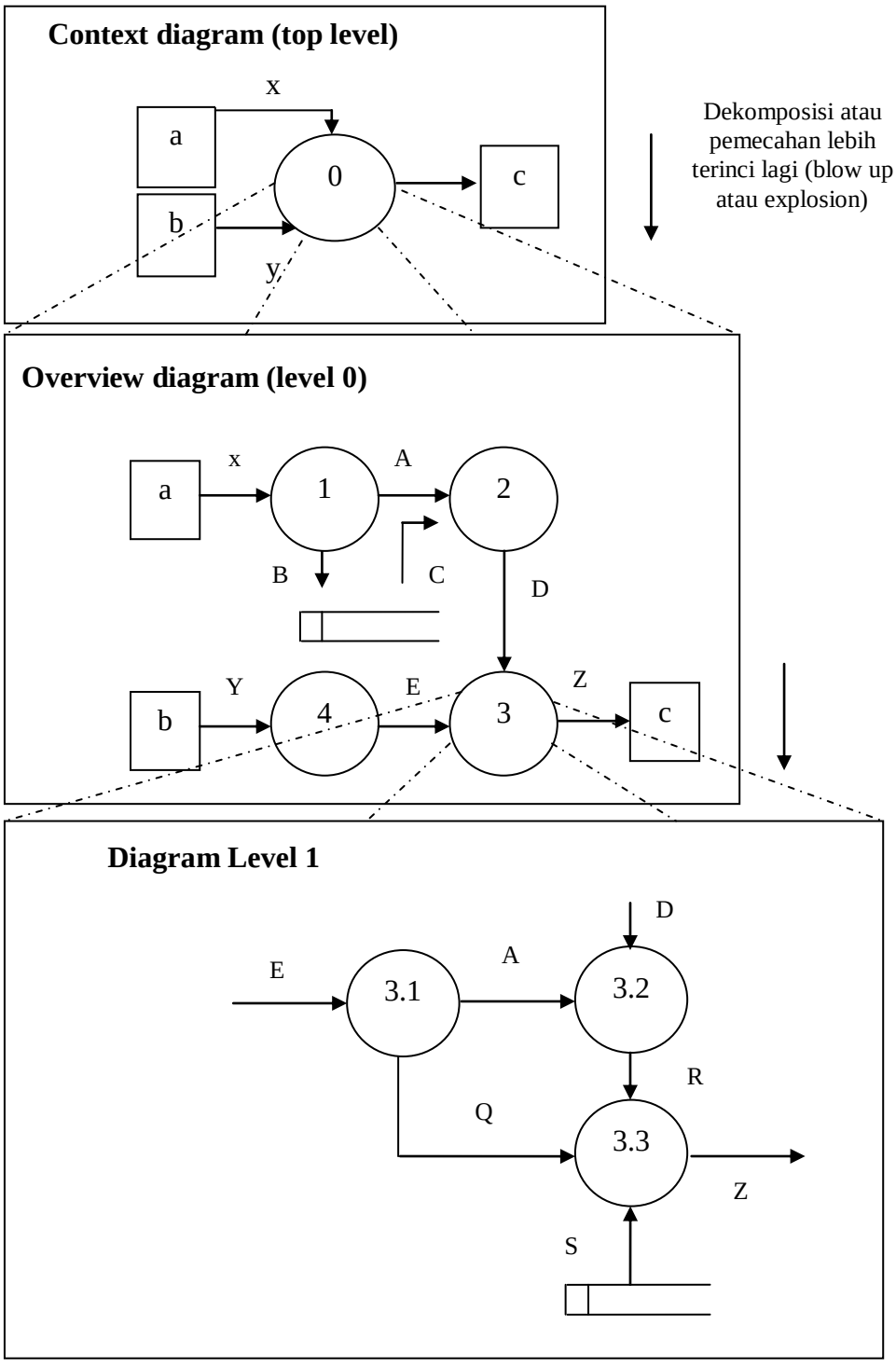
Arus data (*data flow)* di DFD diberi simbol suatu panah. Arus data ini mengalir diantara proses (*process*), simpanan data (*data store*) dan kesatuan luar (*external entity*). Arus data ini menunjukkan arus dari data yang berupa masukan untuk sistem atau hasil dari proses sistem.

3. *Process* (Proses)

Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses. Setiap proses harus diberi penjelasan yang lengkap meliputi identifikasi proses, nama proses dan pemroses.

4. *Data Store* (Simpanan Luar)

Simpanan data (*data store*) merupakan simpanan dari data yang dapat berupa, yaitu suatu file atau *database* di sistem komputer, suatu arsip atau catatan manual, suatu kotak tempat data di meja seseorang, suatu tabel acuan manual, dan suatu agenda atau buku.



Gambar 2.4 Bentuk *Data Flow Diagram* (DFD)

2.2.6 Diagram Konteks

Diagram konteks menggambarkan hubungan antara sistem dengan entitas luarnya. Diagram konteks berfungsi sebagai transformasi dari satu proses yang melakukan transformasi data *inpu* menjadi data *output*. Entitas yang dimaksud adalah entitas yang mempunyai hubungan langsung dengan sistem.

Suatu diagram konteks selalu mengandung satu dan hanya satu proses saja. Proses ini mewakili proses dari seluruh sistem. Diagram konteks ini menggambarkan hubungan *input* atau *output* antara sistem dengan dunia luarnya (kesatuan luar).

2.2.7 Aplikasi Web

Pada awalnya aplikasi *web* dibangun hanya dengan menggunakan bahasa HTML (*HyperText Markup Language*). Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML. Pada saat ini, banyak skrip seperti itu, antara lain yaitu PHP dan ASP, sedangkan contoh yang berupa objek adalah *applet*. Aplikasi *web* itu dapat dibagi menjadi *web* statis dan *web* dinamis.

Web statis dibentuk dengan menggunakan HTML saja. Kekurangan aplikasi seperti ini terletak pada keharusan untuk memelihara program secara terus-menerus untuk mengikuti setiap perubahan yang terjadi. Kelemahan ini diatasi dengan model *web* dinamis.

Dengan menggunakan pendekatan *web* dinamis, dimungkinkan untuk membentuk aplikasi berbasis web. Dari sisi teknologi yang digunakan untuk

membentuk *web* dinamis terdapat dua pengelompokan, yaitu teknologi pada sisi klien dan teknologi pada sisi server.

Teknologi *web* pada sisi klien diimplementasikan dengan mengirimkan kode perluasan HTML atau program tersendiri dan HTML ke klien. Klienlah yang bertanggung jawab dalam melakukan proses terhadap seluruh kode yang diterima. Kelemahan pendekatan seperti ini adalah tedapat kemungkinan bahwa *browser* pada klien tidak mendukung fitur kode perluasan HTML. Kelebihan teknologi pada sisi klien, yaitu memungkinkan penampilan yang bersifat dinamis. Contoh teknologi pada sisi klien, yaitu Kontrol ActiveX, Java Applet, dan Skrip sisi-klien.

Teknologi *web* pada sisi server memungkinkan pemrosesan kode di dalam server sehingga kode yang sampai pada pemakai berbeda dengan kode asli pada server. Contoh teknologi yang berjalan di server, yaitu CGI, ASP, JSP, PHP dan lain sebagainya. Keuntungan penggunaan teknologi pada sisi server adalah sebagai berikut:

1. Mengurangi lalu lintas jaringan dengan cara menghindari percakapan bolak-balik anatara klien dan server.
2. Mengurangi waktu pembuatan kode, mengingat klien hanya mengambil kode HTML saja.
3. Mencegah masalah ketidakkompatibelan *browser*.
4. Klien dapat berinteraksi dengan data yang ada pada server.
5. Mencegah klien mengetahui rahasia kode (mengingat kode yang diberikan ke klien berbeda dengan kode asli pada server).

2.2.8 Personal Home Page (PHP)

2.2.8.1 Sejarah PHP

PHP merupakan bahasa pemrograman untuk *script web server***.** Bahasa pemrograman PHP diciptakan pertama kali oleh **Rasmus Lerdorf**, seorang pemrogram C yang sangat handal. Semula PHP hanya digunakan untuk mencatat seberapa jumlah pengunjung pada *homepage*-nya. **Rasmus** adalah salah seorang pendukung *open source.* Karena itulah ia mengeluarkan *Personal Home Page tools* versi 1.0 secara gratis atau *freeware* pada tahun 1995. Setelah mempelajari YACC dan GNU Bison, **Rasmus** menambah kemampuan pada PHP 1.0 dan menerbitkan PHP 2.0 sebagaipengembangan PHP 1.0 yang telah ada sebelumnya.

PHP 2.0 mampu berhubungan dengan *database* dan dapat diintegrasikan dengan HTML. Pada tahun 1996,PHP telah digunakan oleh banyak website di dunia. Sebuah kelompok pengembang software yang terdiri **Rasmus,Zeew Suraski ,Andi Gutman, Stig Bakken,Shane Carveo** dan **Jim Winstead** bekerja selama tujuh bulan untuk melakukan penyempurnaan PHP 2.0,akhirnya pada tanggal 6 juni 1998,PHP 3.0 resmi dikeluarkan ke dunia pemrograman. Pada pertengahan tahun1999.Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai. Versi ini banyak dipakai karena versi ini mampu dipakai untuk membangun aplikasi *web* kompleks tetapi tetap memiliki kecepatan proses dan stabilitas yang tinggi. PHP 4.0 versi 4.2 telah diterbitkan pada tanggal 22 april 2002 dengan log kelompok fungsi,sampai dengan versi 4.3.7 tercatat 125 kelompok fungsi yang dimiliki oleh PHP. Pada juni 2004, Zend merilis PHP 5.0. Versi ini adalah versi

yang paling mutakhir dari PHP. Dalam versi ini juga dikenalkan model pemrograman berorientasi objek baru untuk menjawab perkembangan bahasa pemrograman berorientasi objek.

2.2.8.2 Kelebihan PHP

PHP memiliki beberapa kelebihan yang tidak dimiliki oleh bahasa-bahasa sejenisnya,yaitu :

1. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan kompilasi dalam penggunaanya.
2. *Web Server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai IIS sampai dengan apache ,dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan *developer* yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman,PHP adalah bahasa *scipting* yang paling mudah karan referensi yang banyak.
5. PHP adalah bahasa *open source* yang dapat di gunakan di berbagai mesin (*linux,unix,windows*) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem.

2.2.9 MySQL

MySQL (*My Structure Query Language*) adalah sebuah program pembuat *database* yang bersifat *open source*. MySQL sebenarnya produk yang berjalan pada *platform Linux*. Karena sifatnya *open source* ,maka MySQL dapat dijalankan pada semua *platform* baik *Windows* maupun *Linux*. Selain itu, MySQL juga merupakan program pengakses *database* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *Multi User* (Banyak Pengguna). Saat ini *database* MySQL telah digunakan hampir oleh semua *programer database*, apalagi dalam pemograman *web.*

Kelebihan lain dari MySQL adalah mengguakan bahasa *Query* standar yang dimiliki SQL (*Structure Query Language*). SQL adalah suatu bahasa permintaan yang terstruktur yang telah di standarkan untuk semuaprogram pengakses *database* seperti Oracle,Posgres SQL ,SQL Server,dan lain-lain.

Sebagai sebuah program penghasil *database*,MySQL tidak dapat berjalan sendiri tanpa sebuah aplikasi lain (*interface*). MySQL dapat didukung oleh hampir semua program aplikasi baik yang *open source* seperti PHP maupun yang tidak,yang ada pada *platform Windows* seperti Visual Basic, Delphi, dan lainnya.

2.2.10 *Universal Decimal Classification* (UDC)

Pengelompokkan atau lazim disebut klasifikasi bahan pustaka dilakukan dengan berpedoman pada suatu bagan klasifikasi, baik yang berbentuk angka (notasi kelas), maupun yang berbentuk kata-kata (Verbal) yang biasa disebut kata kunci untuk yang rumit atau mendalam atau tajuk subjek untuk yang singkat dan sederhana. Cara mengklasifikasi menurut bagan UDC pada umumnya digunakan untuk jenis buku/monograf.

Perpustakaan *Universal Decimal Classification* (UDC) adalah sistem pengklasifikasian buku/bahan pustaka yang dasarnya diambil dari sistem Dewey Decimal Classification (DDC). UDC ini dikelola oleh sebuah organisasi bernama International Federation for Information and Documentation (FID). UDC ini merupakan alat pengindeks dan pencarian kembali untuk mengklasifikasikan informasi atas semua subyek dan dalam berbagai bentuk. Sistem ini juga mencakup semua cabang ilmu pengetahuan yang dibagi menjadi sepuluh cabang.

0 = Karya Umum
1 = Filsafat, metafisika, logika
2 = Agama
3 = Ilmu Sosial
4 = Bahasa/Filologi
5 = Ilmu Murni
6 = Ilmu Terapan
7 = Seni , Olah Raga dan arsitektur
8 = Kesusasteraan
9 = Sejarah , Geografi, dan biografi

2.2.10.1 Tujuan Klasifikasi

Adapun tujuan sistem pengelompokkan ini ialah sebagai berikut :

1. Untuk menetapkan dan menunjukkan isi pokok yang dibahas dalam suatu bahan pustaka
2. Untuk mengumpulkan bahan pustaka yang bidang kajian atau subjeknya sama ke dalam suatu kelompok subjek tertentu.
3. Memudahkan dan memandu pengguna atau pustakawan dalam mencari dan menemukan kembali koleksi atau sekumpulan koleksi pustaka bilamana diperlukan.
4. Untuk menentukan letak dan susunan koleksi pustaka dalam rak.
5. Memandu pengguna menemukan sekumpulan dokumen dalam subjek yang berkaitan satu sama lain sewaktu mereka melakukan pencarian sendiri ke koleksi (browsing).

2.2.10.2 Bagan Klasifikasi UDC

Bagan klasifikasi UDC memiiki beberapa bagian yang perlu diperhatikan pengenalan UDC secara umum, prinsip dasar, struktur UDC, notasi tambahan, bagan utama, dan cara pembentukan nomor klasifikasi.

1. Bagan/ Tabel Utama

Pada bagan utama ini didaftar 10 kelas imu pengetahuan mulai dari 0 untuk ilmu pengetahuan bersifat umum dan sembilan kelompok lainnya mulai dari kelas 1 hingga 9.

2. Tabel Pembantu/ Tambahan

Merupakan tabel-tabel ynag mendaftar notasi tambahan umum dan khusus serta tanda penghubung/ perluasan yang dapat ditambahkan pada tiap notasi utama sesuai keperluan. Notasi dan tanda dalam tabel ini tiadak dpat berdiri sendiri, tetapi harus selalu dicantumkan/ ditambahakan pada notasi pokok yang trdaftar pada tabel utama (kelas utama).

Notasi dan tanda tambahan tersebut terdiri atas sepuluh macam seperti berikut:

Keterangan	Tanda/ Simbol
1. Tanda tambahan dan perluasan	+ dan /
2. Tada yang menghubungkan dua notasi kelas	: atau ::
3. Tanda yang menunjukan bahasa	=
4. Tanda yang menunjukkan bentuk penyajian	(0....)
5. Tanda yang menunjukkan notasi tempat	(1) s/d (9)

6. tanda yang menunjukkan ras dan kebangsaan	(=....)
7. tanda yang menunjukkan petunjuk waktu	“.....”
8. Spesiifikasi alfabetis dan non-UDC untuk mempertajam subjek	A/Z,I,II,III
9. Tambahan untuk menyatakan sudut pandang subjek	.00.....
10. Subdivisi tambahan/khusus	-0/-9,.0 dan’....

Tabel 2.1 Tabel Pembantu/Daftar Notasi

3. Indeks Subjek Alfabetis

Terdiri dari subjek-subjek verbal dalam bahasa Inggris disertai notasi kelasnya, disusun berdasarkan abjad dan mengacu kepad nomor kelas pada bagan utama. Fungsinya untuk menunjukkan notasi kelas dari suatu subjek dalam tabel utama UDC.

Contoh pembagian dalam tabel utama.

0	Umum	61	Ilmu Kedokteran/Kesehatan	630	Kehutanan
1	Ilmu Filsafat			631	Pertanian/ Agronomi
2	Ilmu Agama	62	Keteknikan dan Teknologi	632	Tanaman Pangan
3	Ilmu Sosial	63	Pertanian, Kehutanan, Peternakan, Perburuan, Perikanan	633	Taaman pangan
4	Kosong	64	Ekonomi Rumah tangga, Perhotelan, Catering	634	Tanaman perkebunan
5	Ilmu Murni	65	Managemen organisasi perdaganga, komunikasi, transportasi	635	Tanaman holtikultura
6	Ilmu Terapan	66	Industri dan Teknologi Kimia	636	Peternakan
7	Kesenian	67 /68	Industri/Kerajinan	637	Hasil Peternakan
8	Bahasa, Kesusastraan			638	Peternakan lebah, lak, dll
9	Sejarah, geografi		Industri Bangunan	639	Prikanan