BAB II
TINJAUAN PUSTAKA

2.1 Profil Perusahaan [1]

Profil dari perusahaan yang akan dibahas pada bab II adalah mengenai profil dari Lembaga Ilmu Pengetahuan Indonesia (LIPI). Profil perusahaan meliputi sejarah perkembangan, visi dan misi, logo perusahaan, badan hukum, struktur organisasi serta tugas dan fungsi dari Lembaga Ilmu Pengetahuan Indonesia (LIPI).

2.1.1 Sejarah Perkembangan Lembaga Ilmu Pengetahuan Indonesia (LIPI)

Berdirinya Pusat Penelitian Informatika tidak dapat dipisahkan dengan sejarah terbentuknya Lembaga Elektroteknika Nasional (LEN). Lembaga Elektroteknika Nasional sebagai lembaga penelitian dalam bidang elektronika diresmikan pada tanggal 10 Juni 1965. Bermula di kampus ITB dengan ruang seluas 30 meter persegi yang merupakan pinjaman dari ITB. Pada tahun 1967, berdasarkan Keputusan Presiden nomor 128 tahun 1967, Lembaga Elektroteknika Nasional (LEN) bernaung dibawah koordinasi Lembaga Ilmu Pengetahuan Indonesia. Pada tahun 1969, Prusahaan Negara Pertambangan Minyak dan Gas Bumi Nasional (Pertamina), memberikan pinjaman gedung seluas 1000 meter persegi di Jl. Sawunggaling No. 14 Bandung, sehingga ruangan dikomplek ITB ditinggalkan. Pada awal tahun 1978, komplek gedung perkantoran dan laboratorium LEN-LIPI sudah tersebar di 9 lokasi di kota Bandung, dengan luas lantai sekitar 2500 meter persegi.

Tahun 1983 LEN-LIPI pindah ke Jl. Soekarno-Hatta No.407 250A Bandung, serta gedung tersebut diresmikan pemakaiannya oleh Presiden Republik Indonesia, Bapak Soeharto. pada tanggal 31 Mei 1983. Dengan dilakukannya reorganisasi di Lembaga Ilmu Pengetahuan Indonesia (LIPI) berdasarkan Keppres No. l tahun 1986 tanggal 13 Januari 1986, Lembaga Elektroteknika Nasional berkembang menjadi tiga Puslitbang dan satu Unit Pelaksana Teknis, yaitu:

1. Pusat Penelitian dan Pengembangan Telekomunikasi, Elektronika Strategis, Komponen dan Material (Puslitbang Telkoma-LIPI).
2. Pusat Penelitian dan Pengembangan Informatika dan Ilmu Pengetahuan Komputer (Puslitbang INKOM LIPI).
3. Pusat Penelitian dan Pengembangan Tenaga Listrik dan Mekatronika (Puslitbang Telimek-LIPI)

4. UPT Pusat Laboratorium Enjiniring Nasional (UPT Pusat LEN LIPI).

Dalam struktur organisasi LIPI, keempat Puslibang tersebut diatas berada dibawah koordinasi Deputi Bidang Ilmu Pengetahuan Teknik -LIPI.

Pada tanggal 8 Maret 1990 seluruh fasilitas UPT LEN-LIPI diserahkan ke BPIS. Kantor Puslitbang INKOM LIPI pindah ke Jl. Ranggamalela No. 11 Bandung, sedangkan laboratoriumnya menempati salah satu bagian gedung di kompleks LIPI Jl. Cisitu 21/154D Bandung. Pada tahun 2001 dalam Reorganisasi baru Lembaga Ilmu Pengetahuan Indonesia, Pusat Penelitian dan Pengembangan Informatika dan Ilmu Pengetahuan Komputer berubah nama dan struktur organisasinya menjadi Pusat Penelitian Informatika LIPI.

2.1.2 Visi dan Misi Lembaga Ilmu Pengetahuan Indonesia (LIPI)

1. Visi

Visi dari pusat penelitian informatika LIPI yaitu menjadi acuan nasional pada bidang informatika dan ilmu pengetahuan komputer.

2. Misi

Misi dari pusat penelitian informatika LIPI yaitu melaksanakan kegiatan penelitian, pengembangan dan pembinaan, pelayanan jasa, dan saran kebijakan di bidang informatika dan ilmu pengetahuan komputer.

2.1.3 Logo Lembaga Ilmu Pengetahuan Indonesia (LIPI)

Logo Lembaga Ilmu Pengetahuan Indonesia (LIPI) berdasarkan SK Ketua LIPI Nomor 37/KEP/C.3/1975 (Gambar 2.1) dengan penjelasan sebagai berikut:

a. Logo dan Penjelasan Logo

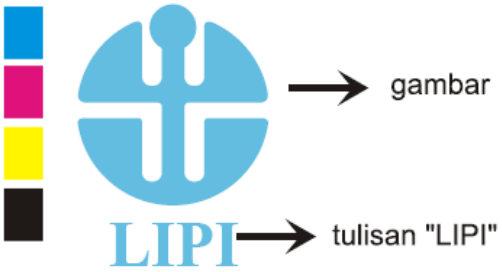
Logo LIPI adalah huruf dan atau lambang yang mengandung suatu makna sebagai lambang lembaga/identitas yang melekat dalam mengikat batin dan ksatuan jiwa seluruh pegawai LIPI.

b. Bentuk Logo dan Jenis Warna

Logo LIPI adalah berbentuk bulat dan merupakan satu kesatuan dari lambang dan huruf LIPI dibawahnya.

Jenis warna adalah “TURQUISE BLUE” dengan spesifikasi warna campuran sebagai berikut :

- CYAN = 47%
- MAGENTA = 1%
- YELLOW = 6%
- BLACK = 0%



Gambar 2.1 Logo LIPI

Logo LIPI terdiri dari gambar dan tulisan LIPI sebagai satu kesatuan yang tidak dipisahkan.

Penempatan Logo LIPI ditempatkan pada tempat yang layak dan baik bagi pandangan mata.

Penggunaan Logo LIPI dilakukan pada Kantor LIPI, Ruang Rapat, Kop Surat Dinas, Cap Instansi, Amplop Dinas, Pakaian Resmi, Jaket, Kartu Nama, dll.

Penanggung jawab penggunaan Logo LIPI antara lain adalah sebagai berikut.

a. Inspektorat LIPI
b. Kepala Satuan Kerja
c. Biro Umum dan Perlengkapan LIPI

Dan barang siapa yang tidak mengindahkan tata cara dalam penggunaan Logo LIPI, akan dikenakan sanksi menurut peraturan yang berlaku.

2.1.4 Badan Hukum Lembaga Ilmu Pengetahuan Indonesia (LIPI)

Pusat Penelitian Informatika Lembaga Ilmu Pengetahuan Indonesia didirikan pada tanggal 13 Januari 1986. Semula bernama Pusat Penelitian dan Pengmbangan Informatika dan Ilmu Pengetahuan Komputer. Berdasarkan SK kepala LIPI nomor : 115/m/2001, tanggal 5 juni 2001 tentang organisasi dan tata kerja sebagaimana telah diubah dengan keputusan kepala LIPI nomor : 3213/m/2004.

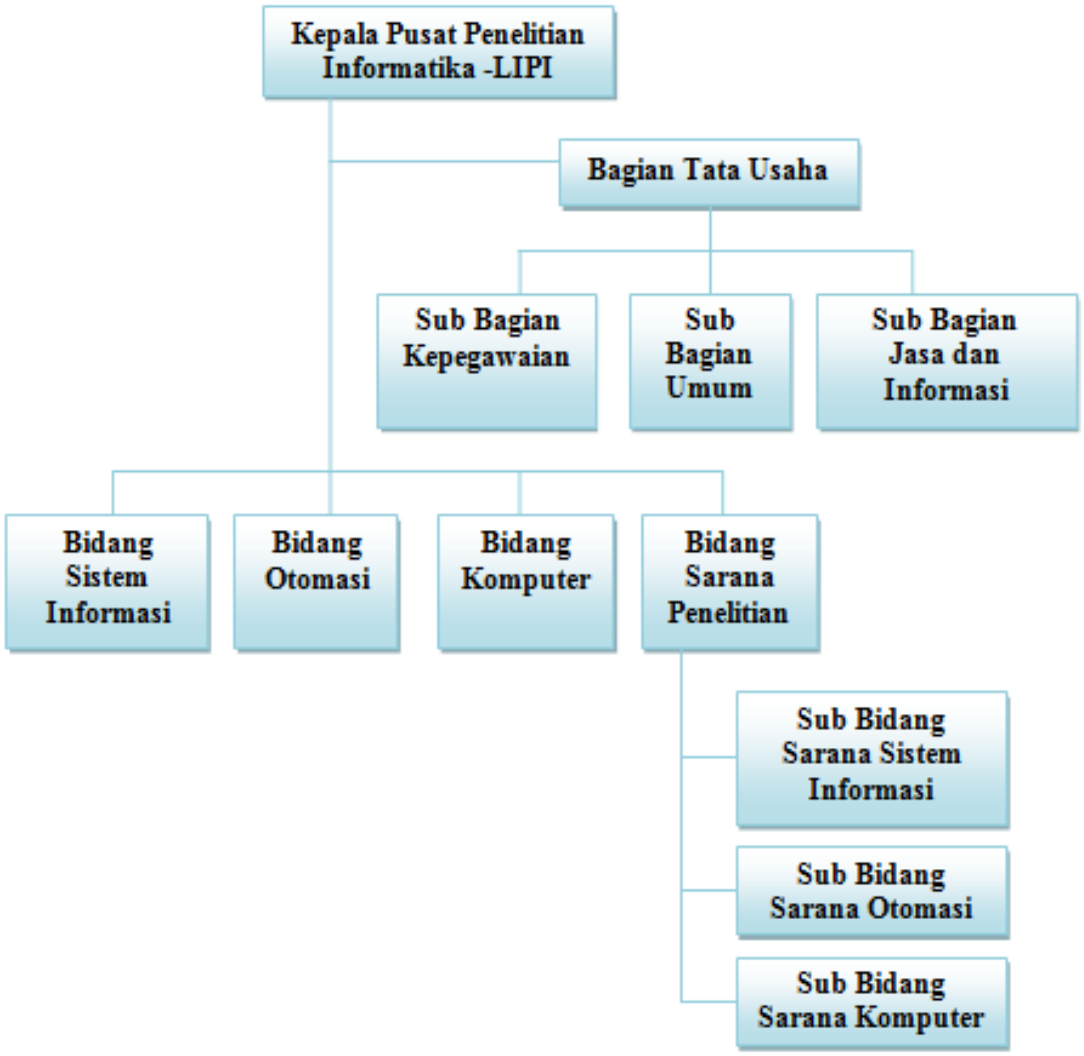
Pusat Penelitian Informatika berada dibawah deputi bidang Ilmu Pengetahuan Teknik yang berlokasi di jalan Cisitu no.21/154D Bandung. Untuk melaksanakan tugas dan fungsinya sebagaimana dituangkan dalam PPK LIPI. Berdasarkan rumusan hasil Rapat Kerja LIPI dan rumusan hasil rapat kerja kedeputian bidang IPT-LIPI serta hasil Rapat Kerja Pusat Penelitian Informatika LIPI. Maka perlu menyusun buku panduan kerja untuk melaksanakan kegiatan Pusat Penelitian informatika-LIPI. Buku panduan kerja ini merupakan penjabaran/turunan dan PPK LIPI, senigga diharapkan dapat menjadi acuan dan pedoman kerja untuk semua pegawai di Pusat Penelitian Informatika-LIPI.

2.1.5 Struktur Organisasi Lembaga Ilmu Pengetahuan Indonesia (LIPI)

Struktur Organisasi Pusat Penelitian Informatika-LIPI, berdasarkan SK Kepala LIPI Nomor 1151/M/2001 tanggal 5 Juni 2001 tentang Organisasi dan Tata Kerja sebagaimana telah diubah dengan Keputusan Kepala LIPI Nomor 3212/M/2004 dimana struktur organisasi terdiri dari :

1. Kepala Pusat Penelitian Informatika LIPI
2. Bagian Tata Usaha, yang terdiri dari.
    a. Sub Bagian Kepegawaian
    b. Sub Bagian Umum
    c. Sub Bagian Jasa dan Informasi
3. Bidang Sistem Informasi
4. Bidang Otomasi
5. Bidang Komputer
6. Bidang Sarana Penelitian, yang terdiri dari.
    a. Sub Bidang Sarana Sistem Informasi
    b. Sub Bidang Sarana Otomasi
    c. Sub Bidang Sarana Komputer

Struktur organisasi dari Lembaga Ilmu Pengetahuan Indonesia (LIPI) dapat dilihat pada Gambar 2.2 .



Gambar 2.2 Struktur Organisasi LIPI

2.1.6 Tugas dan Fungsi Lembaga Ilmu Pengetahuan Indonesia (LIPI)

Pusat Penelitian Informatika LIPI mempunyai tugas melaksanakan kegiatan penelitian. meningkatkan kemampuan masyarakat ilmiah dan industri, serta mendayagunakan hasil penelitian dan pengembangan di bidang informatika, sesuai dengan kebijaksanaan yang telah ditetapkan oleh Kepala LIPI. Untuk menyelenggarakan tugas tersebut, Puslit Informatika LIPI mempunyai fungsi:

1. Mempersiapkan program penelitian. di bidang Informatika, kendali, dan komputer.
2. Melaksanakan penelitian di bidang Informatika, kendali dan komputer

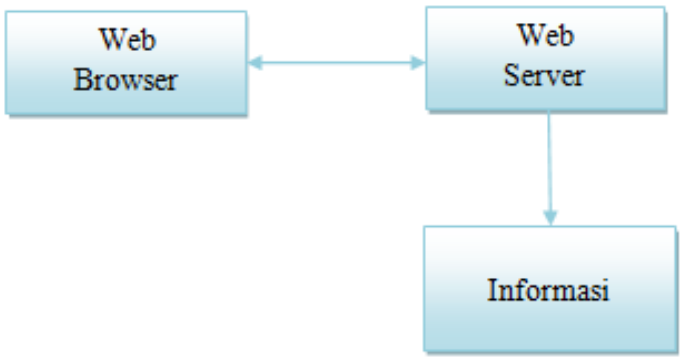
3. Melaksanakan pengalihan dan penguasaan teknologi dibidang informatika, kendali dan komputer. Melaksanakan pengembangan standarisasi, metoda dan prosedur pengukuran dan pengujian di bidang informatika, kendali dan komputer.
4. Melaksanakan pengujian teknik di bidang informatika, kendali dan komputer terhadap hasil industri. Melaksanakan pengamatan, pengumpulan, analisa dan penyebarluasan data yang menunjang penelitian dan pengembangan bidang informatika, kendali dan komputer.
5. Melaksanakan kerjasama penelitian dan pengembangan, dan perekayasaan di bidang informatika, kendali dan komputer dengan lembaga/instansi di dalam dan luar negeri.
6. Melaksanakan peningkatan keterampilan tenaga penelitian dan pengembangan di bidang informatika, kendali dan komputer.
7. Mengadakan evaluasi hasil penelitian dan pengembangan, perekyasaan. penerapan, pengalihan dan penguasaan teknologi dibidang informatika, kendali dan komputer, serta menyusun laporan.
8. Melaksanakan urusan tata usaha.

2.2 Landasan Teori

Dalam pengembangan sistem jurnal online INKOM LIPI terdapat beberapa landasan teori yang mendukung diantaranya aplikasi berbasis web, media jurnal, jurnal online, *open journal system* (OJS), PHP, MySQL dan Joomla!.

2.2.1 Aplikasi Berbasis web [2]

Aplikasi berbasis web adalah aplikasi yang dapat diakses melalui jaringan LAN atau Internet. Aplikasi berbasis web dibangun diatas diatas HTTP. HTTP merupakan salah satu protokol yang berjalan diatas TCP/IP (protokol internet). HTTP adalah protokol yang *stateless,* web server hanya memberikan informasi yang diminta, setelah itu koneksi diputus. Layanan informasi ini disediakan oleh *web server* dan diakses oleh *web browser*, untuk lebih jelasnya, dapat dilihat pada Gambar (2.3) yakni sebagai berikut.



Gambar 2.3 Aplikasi Berbasis Web

2.2.1.1 Web Browser

Web browser digunakan untuk memperoleh informasi dengan format *hypertext*. Web browser akan mengirimkan *request* ke web server, dan menampilkan hasilnya ke pengguna. Selain itu web browser juga dapat mengirimkan form ke web server untuk diproses. Contoh dari web browser adalah NCSA Mosaic, Netscape Navigator, MacWeb, IE, Opera, Conqueror dan lain-lain.

2.2.1.2 Web Server

Web server memberikan jawaban/*response* dari permintaan/*request* web browser. Web server juga dapat memproses form yang dikirimkan oleh web browser. Contoh dari Web server adalah, Apache, NCSA HTTPD (Unix), Webstar (Mac), IIS/PWS (Windows).

2.2.1.3 Informasi

Antarmuka web bisa mengirimkan informasi dalam bentuk text terformat (HTML), grafik/animasi, dan suara. HTTP mengirimkan informasi dengan format HTML (*Hypertext Markup Language*). HTML sendiri merupakan format sederhana dari standar SGML. Selain itu HTTP juga dapat mengirimkan informasi dalam bentuk lain.

Aplikasi web terletak pada server, yang merupakan perpanjangan dari webserver untuk mengirim dan memporses informasi. Jadi informasi yang dikirim ke browser menjadi dinamis dengan adanya aplikasi tersebut. Proses tersebut dapat dilakukan oleh CGI (*Common Gateway Interface)* yang dapat dikembangkan dengan berbagai bahasa pemrograman dalam berbagai platform selama menggunakan standar yang ditetapkan.

Aplikasi berbasis web dapat juga dijalankan sebagai script oleh processor semacam ASP atau PHP. Script dan program tersebut akan memberikan informasi dalam format HTML ke webserver untuk diteruskan ke browser dan juga memproses informasi dari web

server, misalnya menyimpan data ke basisdata. Perlu disadari bahwa aplikasi berbasis web kebayakan menggunakan database (*database-driven application*). Contoh aplikasi web yang sering digunakan adalah webmail yaitu aplikasi web untuk membaca dan mengirimkan email lewat browser.

2.2.1.4 Keunggulan Aplikasi Berbasis Web

Keunggulan-keunggulan yang terdapat dari aplikasi berbasis web adalah sebagai berikut:

1. Klien yang kecil, tidak membutuhkan *resource* yang besar untuk menjalankan aplikasinya. Cukup dengan komputer yang sanggup menjalakan web browser.
2. Kemudahan untuk *depolyment* atau penyebaran aplikasi
3. *Cross platform*, karena menggunakan standard umum HTTP/HTML. Bisa dijalankan dengan berbagai macam sistem operasi.
4. *Learning curve*, khususnya yang telah mengenal web sebelumnya. Biasanya aplikasi web dirancang secara intuitif.
5. Kemudahan untuk digabungkan dengan layanan internet lainnya, misalnya email, ftp dan lain-lain.
6. Banyak tersedia modul/source untuk komponen pengembangan aplikasi
7. Mudah diakses darimana saja, dari intranet maupun internet.
8. Dengan web yang dibangun, dapat mengunakan modul-modul yang telah dikembangkan sebelumnya. Terdapat komunitas yang kuat untuk mensupport pengembangan aplikasi opensource misalnya Sourceforge.net

2.2.1.5 Kelemahan Aplikasi Berbasis Web

Kelemahan-kelemahan yang terdapat dari aplikasi berbasis web adalah sebagai berikut:

1. User Interface tidak sekaya aplikasi platform lain, karena keterbatasan standar HTML. Beberapa bisa ditingkatkan dengan javascript tetapi menimbulkan masalah cross browser.
2. Aplikasi yang *stateless*, diakibatkan protokol HTTP.
3. Banyak faktor yang mempengaruhi performansi aplikasi. Seperti jaringan, server dan lain-lain.
4. Semua proses dilakukan diserver, ini akan memberatkan server jika proses yang dilakukan terlalu banyak.
5. Sangat tergantung pada jaringan

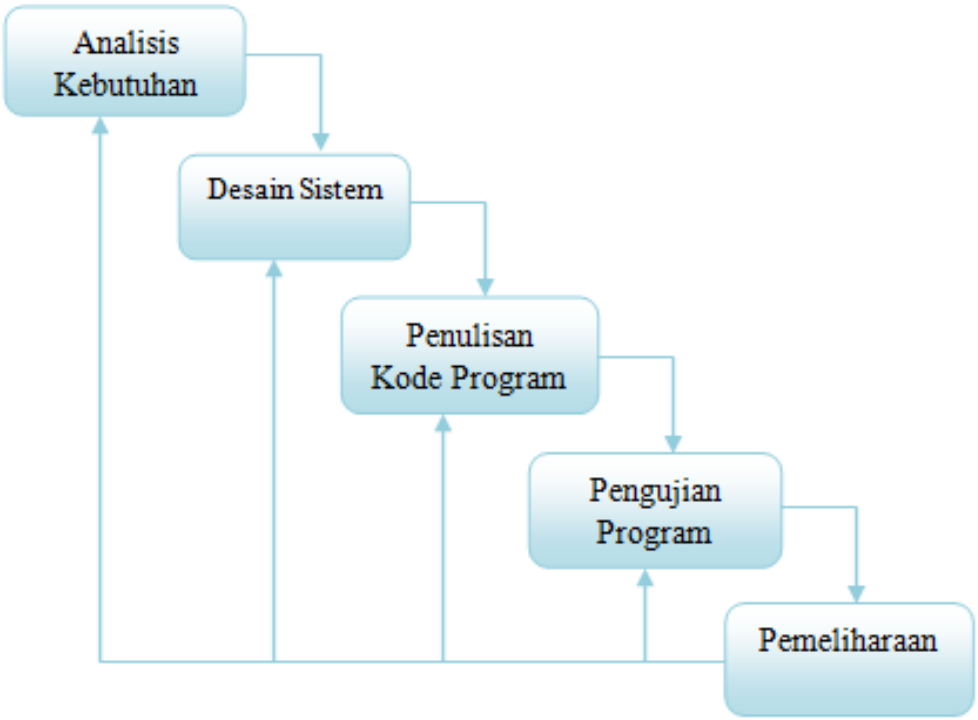
2.2.2 Metode Pengembangan Sistem [3]

Metode yang digunakan dalam pengembangan sistem informasi pengolahan data pasien di puskesmas sempaja adalah metode *waterfall*. Metode ini juga sering disebut sebagai metode dengan pendekatan air terjun atau lebih dikenal dengan istilah metode *waterfall approach*.

2.2.2.1 Metode Waterfall

Paradigma metode waterfall terkadang juga disebut dengan model *classic life cycle*, model ini terhubung secara sistematik dengan memulai teknis pengembangan perangkat lunak yang dimulai dari level sistem dengan tahapan-tahapan selanjutnya yaitu *system engginering, software analysis, software design, coding, testing dan maintenance*. Dimana konsep dari metode ini adalah bagaimana melihat suatu masalah secara sistematis dan terstruktur dari sistem yang telah ada sebelumnya.

Alur dari metode waterfall atau metode waterfall approach dapat dilihat pada Gambar 2.4.



Gambar 2.4 Alur Metode Waterfall

Penjelasan dari Gambar 2.4 mengenai metode *waterfall* adalah sebagai berikut :

1. Rekayasa dan Pemodelan (*System Engginering*), pada tahap ini, rekayasa informasi mencakup pengumpulan kebutuhan aplikasi perangkat lunak yang akan dibangun.
2. Analisis perangkat Lunak (*Software Analysis*), menentukan apakah kegiatan dari sistem engineering dapat diimplementasikan menjadi sebuah sistem informasi atau tidak dan menentukan prosedur-prosedur yang bekerja. Adapun fungsi-fungsi tersebut meliputi fungsi masukan, fungsi proses dan fungsi keluaran.
3. Perancangan perangkat Lunak (*Software Design*), perancangan perangkat lunak merupakan perancangan perangkat lunak yang dilakukan berdasarkan data-data yang telah dikumpulkan pada tahap sebelumnya. Perancangan tersebut meliput perancangan struktur file, stuktur menu, stuktur program, format masukan (*input*) dan format keluaran (*output*).
4. Implementasi perangkat lunak (*Coding*), implementasi perangkat lunak yaitu kegiatan yang mengimplementasikan hasil dari perancangan perangkat lunak ke dalam kode program yang dimengerti oleh bahasa mesin.
5. Pengujian perangkat lunak (*Testing*), memfokuskan pada logika internal dari perangkat lunak, fungsi eksternal, dan mencari segala kemungkinan kesalahan, memeriksa apakah input sesuai dengan hasil yang diinginkan setelah proses.
6. Pemeliharaan (*Maintenance*), penerapan secara keseluruhan disertai pemeliharaan jika terjadi perubahan struktur baik dari segi perangkat lunak maupun maupun perangkat kerass.

2.2.2.2 Pemodelan Berorientasi Objek [4]

Pada metode pengembangan sistem ini akan menggunakan *object oriented Progaming* (OOP) yaitu paradigma dalam rekayasa perangkat lunak yang didasarkan pada objek dan kelas. *Object-oriented progaming* merupakan metode terbaik dalam rekayasa perangkat lunak dan mencakup bidang aplikasi yang sangat luas. Karena luasnya cakupan *object-oriented progaming* maka terdapat beberapa hal yang membuat bingung mengenai beberapa istilah dan konsep *object-oriented progaming*. Istilah dan konsep yang berkenaan dengan *object-oriented progaming* ini adalah :

1. *Object-oriented Analysis* adalah metode analisis yang memeriksa *requirement* (syarat / keperluan yang harus dipenuhi suatu sistem) dari sudut pandang kelas-kelas dan objek-objek yang ditemukan dalam ruang lingkup permasalahan.

2. *Object-oriented Design* adalah metode untuk mengarahkan arsitektur perangkat lunak yang didasarkan pada manipulasi objek-objek sistem atau subsistem.

Object Oriented Analysis and Design (OOAD) merupakan pendekatan yang menekankan pada solusi logis berbasis objek. Beberapa konsep dasar dalam OOAD adalah sebagai berikut :

a. Objek

Objek adalah "benda", secara fisik atau konseptual dapat ditemui dalam kehidupan sehari-hari. Perangkat keras, pernagkat lunak, dokumen, manusia bahkan konsep semuanya adalah konsep objek. Sebuah objek memiliki keadaaan sesaat (*state*) dan perilaku (*behaviour*). *State* dari sebuah objek adalah kondisi dari objek tersebut atau himpunan dari keadaan yang menggambarkan objek tersebut.

b. Kelas

Kelas adalah definisi umum untuk himpunan objek sejenis. Kelas menetapkan spesifikasi prilaku dan atribut objek-objek tersebut.

Object oriented Programing merupakan metode yang paling baik dalam rekayasa perangkat lunak diantaranya *procedure-oriented, object-oriented, data struktur-oriented, data flow-oriented, dan constraint-oriented*. Sehingga dengan metode *object-oriented progaming* dapat diaplikasikan dalam seluruh ruang lingkup rekayasa perangkat lunak.

Untuk memhami keunggulan OOAD (*Object Oriented Analysis and Design*) maka perlu memahami masalah yang dihadapi, antara lain sebagai berikut:

1. Software sulit untuk dimodifikasi bila memerlukan pengembangan.
2. Proses pembuatan software memerlukan waktu yang cukup lama sehingga kadang kala melebihi anggaran dalam pembuatannya.
3. Para programmer selalu membuat software dari dasar karena tidak adanya kode yang bisa digunakan ulang (reuse).

Dari beberapa keunggulan diatas, dengan menggunakan *object oriented progaming* maka sangat menguntungkan bagi *programmer* karena *programmer* dapat mendesign program dalam bentuk objek-objek dan hubungan antara objek-objek tersebut untuk kemudian dimodelkan dalam sistem nyata. Keuntungan yang lain adalah proses pembuatan perangkat lunak dapat dilakukan dengan lebih cepat karena perangkat lunak dibangun dalam objek-objek standar, sehingga dapat digunakan secara berulang-ulang.

Bahasa pemograman yang menggunakan *object oriented progaming* diantaranya yang terkenal C++, Visual Basic dan Java.

2.2.3 *Unified Modeling Language* (UML) [5]

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak.

2.2.3.1 Definisi UML

Unified Modeling Language (UML) merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek (Whitten L. Jeffery et al, 2004).

Sedangkan menurut Henderi (2007: 4) Unified Modeling Language (UML) adalah sebuah bahasa pemodelan yang telah menjadi standar dalam industri software untuk visualisasi, merancang, dan mendokumentasikan sistem perangkat lunak.

Bahasa Pemodelan UML lebih cocok untuk pembuatan perangkat lunak dalam bahasa pemrograman berorientasi objek (C+ , Java, VB.NET), namun demikian tetap dapat digunakan pada bahasa pemrograman prosedural (Ziga Turck, 2007)

Unified Modeling Language (UML) biasa digunakan untuk beberapa hal berikut ini, diantaranya.

1. Menggambarkan batasan sistem dan fungsi-fungsi sistem secara umum, dibuat dengan *use case dan actor.*
2. Menggambarkan kegiatan atau proses bisnis yang dilaksanakan secara umum, dibuat dengan *interaction diagrams.*
3. Menggambarkan representasi struktur statik sebuah sistem dalam bentuk *class diagrams.*
4. Membuat model behavior "yang menggambarkan kebiasaan atau sifat sebuah sistem" dengan *state transition diagrams.*
5. Menyatakan arsitektur implementasi fisik menggunakan *component and development diagrams*
6. Menyampaikan atau memperluas fungsionalitas dengan *stereotypes* (Ziga Turck, 2007)

UML merupakan salah satu alat bantu yang sangat handal dalam bidang pengembangan sistem berorientasi objek karena UML menyediakan bahasa pemodelan visual yang memungkinkan pengembang sistem membuat *blue print* atas visinya dalam bentuk yang baku.

UML berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dalam sistem melalui sejumlah elemen grafis yang bisa dikombinasikan menjadi diagram. UML mempunyai banyak diagram yang dapat mengakomodasi berbagai sudut pandang dari suatu

perangkat lunak yang akan dibangun. Diagram-diagram tersebut digunakan untuk hal-hal sebagai berikut (Henderi et al, 2008:71):

1. Mengkomunikasikan ide
2. Melahirkan ide-ide baru dan peluang-peluang baru
3. Menguji ide dan membuat prediksi
4. Memahami struktur dan relasi-relasinya

2.2.3.2 Konsep Pemodelan Menggunakan Unified Modeling Language (UML)

Pemodelan menggunakan *Unified Modeling Language* (UML) merupakan metode pemodelan berorientasi objek dan berbasis visual. Karenanya pemodelan menggunakan UML merupakan pemodelan objek yang fokus pada pendefinisian struktur statis dan model sistem informasi yang dinamis daripada mendefinisikan data dan model proses yang tujuannya adalah pengembangan tradisional.

UML menawarkan diagram yang dikelompokan menjadi lima perspektif berbeda untuk memodelkan suatu sistem. Seperti satu set *blue print* yang digunakan untuk membangun sebuah rumah. Jadi, hanya dibutuhkan sebuah UML agar dapat dibangun sebuah sistem.

2.2.3.3 Diagram Dasar dalam Unified Modeling Language (UML)

Diagram dasar *Unified Modelling Language* (UML) serta tujuannya dijelaskan seperti berikut :

a. Model Use Case Diagram

Use Case Diagram secara grafis menggambarkan interaksi antara sistem, sistem eksternal, dan pengguna. Dengan kata lain Use Case diagram secara grafis mendeskripsikan siapa yang akan menggunakan sistem dan dalam cara apa pengguna (user) mengharapkan interaksi dengan sistem itu. Use Case secara naratif digunakan untuk secara tekstual menggambarkan sekuensi langkah-langkah dari setiap interaksi.

b. Diagram Struktur Statis

UML menawarkan dua macam diagram untuk memodelkan struktur statis sistem informasi, yaitu:

a. ***Class Diagram*:** menggambarkan struktur objek sistem. Diagram ini menunjukan *class object* yang menyusun sistem dan juga hubungan antara *class object* tersebut.
b. ***Object Diagram*:** serupa dengan *class diagram*, tetapi *object diagram* memodelkan *isntance object actual* dengan menunjukan nilai-nilai saat ini dari atribut instance. Object

Diagram menyajikan “snapshot/potret” tentang objek sistem pada point waktu tertentu. Diagram ini tidak digunakan sesering *Class Diagram*, tetapi saat digunakan dapat membantu seorang developer memahami struktur sistem secara lebih baik.

c. Diagram Interaksi

Diagram interaksi memodelkan sebuah interaksi, terdiri dari satu set objek, hubungan-hubungannya, dan pesan yang terkirim di antara objek. Model diagram ini memodelkan behavior (kelakuan) sistem yang dinamis dan UML memiliki dua diagram untuk tujuan ini, yaitu:

a. **Diagram rangkaian/Sequence Diagram:** secara grafis menggambarkan bagaimana objek berinteraksi dengan satu sama lain melalui pesan pada sekuensi sebuah use case atau operasi. Diagram ini mengilustrasikan bagaimana pesan terkirim dan diterima di antara objek dan dalam sekuensi atau timing apa.
b. **Diagram kolaborasi/Collaboration Diagram:** serupa dengan diagram rangkaian/sekuensi, tetapi tidak fokus pada timing atau sekuensi pesan. Diagram ini justru menggambarkan interaksi (atau kolaborasi) antara objek dalam sebuah format jaringan.
c. **Diagram rangkaian maupun diagram kolaborasi** merupakan isomorphic artinya kita dapat mengubah dari satu diagram ke diagram lain.

d. Diagram State/State Diagram

UML memiliki sebuah diagram untuk memodelkan *behavior* objek khusus yang kompleks (*statecahrt*) dan sebuah diagram untuk memodelkan *behavior* dari sebuah use case atau sebuah metode, yaitu:

a. ***Diagram statechart:*** digunakan untuk memodelkan behavior objek khusus yang dinamis. Diagram ini mengilustrasikan siklus hidup objek-berbagai keadaan yang dapat diasumsikan oleh objek dan event-event (kejadian) yang menyebabkan objek beralih dari satu state ke state lain.
b. ***Diagram aktivitas/Activity Diagram:*** secara grafis digunakan untuk menggambarkan rangkaian aliran aktivitas baik proses bisnis maupun use case. Activity diagram dapat juga digunakan untuk memodelkan action yang akan dilakukan saat sebuah operasi dieksekusi, dan memodelkan hasil dari action tersebut.

e. Diagram Implementasi

Diagram implementasi juga memodelkan struktur sistem informasi, yaitu sebagai berikut:

a. ***Diaram komponen/Component Diagram:*** digunakan untuk menggambarkan organisasi dan ketergantungan komponen-komponen pernagkat lunak sistem. Komponen diagram dapat digunakan untuk menunjukan bagaimana kode pemrograman dibagi menjadi modul-modul (atau komponen).

b. ***Diagram penguraian/Deployment:*** digunakan untuk mendeskripsikan arsitektur fisik dalam istilah ”node” untuk hardware dan software dalam sistem. Diagram ini menggambarkan konfigurasi komponen-komponen software *real-time*, prosesor, dan peralatan yang membentuk arsitektur sistem.

2.2.3.4 Definisi Use Case Diagram

Use Case diagram adalah model fungsional sebuah sistem yang menggunakan *actor* dan *use case*. *Use Case* adalah layanan (*services)* atau fungsi-fungsi yang disediakan oleh sistem untuk pengguna-penggunanya (Henderi et al, 2008). Use Case adalah suatu pola atau gambaran yang menunjukan kelakukan atau kebiasaan sistem. Setiap Use Case adalahsuatu urut-urutan (*sequence*) transaksi yang saling berhubungan dan dilakukan oleh sebuah actor dan sistem dalam bentuk sebuah dialog (Henderi, 2007). Use Case Diagram dibuat untuk memvisualisasikan/ menggambarkan hubungan antara Actor dan Use Case. Use Case diagram mempresentasikan kegunaan atau fungsi-fungsi sistem dari perspektif pengguna.

2.2.3.5 Definisi Class Diagram

Class adalah kumpulan objek-objek dengan dan yang mempunyai struktur umum, behavior umum, relasi umum, dan semantic/kata yang umum. *Class-class* ditentukan/ditemukan dengan cara memeriksa objek-objek dalam sequence diagram dan collaboration diagram. Sebuah class digambarkan seperti sebuah bujur sangkar dengan tiga bagian ruangan. Class sebaiknya diberi nama menggunakan kata benda sesuai dengan domain/bagian/kelompoknya (Whitten L. Jeffery et al, 2004).

Class Diagram adalah diagram yang menunjukan class-class yang ada dari sebuah sistem dan hubungannya secara logika. Class diagram menggambarkan struktur statis dari sebuah sistem. Karena itu class diagram merupakan tulang punggung atau kekuatan dasar dari hampir setiap metode berorientasi objek termasuk UML (Henderi, 2008). Sementara menurut (Whitten L. Jeffery et al 2004:432) class diagram adalah gambar grafis mengenai

struktur objek statis dari suatu sistem, menunjukan class-class objek yang menyusun sebuah sistem dan juga hubungan antara class objek tersebut.

Elemen-eleman class diagram dalam pemodelan UML terdiri dari: Class-class, struktur class, sifat class (class behavior), perkumpulan/gabungan (association), pengumpulan/kesatuan (agregation), ketergantungan (dependency), relasi-relasi turunannya, keberagaman dan indikator navigasi, dan role name (peranan/tugas nama).

2.2.3.6 Definisi Activity Diagram

Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Activity diagram merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

Sama seperti *state*, standar UML menggunakan segiempat dengan sudut pembulat untuk menggambarkan aktivitas. *Decision* digunakan untuk menggambarkan behaviour pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (*fork* dan *join*) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal. *Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.

2.2.3.7 Definisi Sequence Diagram

Sequence diagram adalah suatu diagram yang menggambarkan interaksi antar obyek dan mengindikasikan komunikasi diantara obyek-obyek tersebut. Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh obyek-obyek yang melakukan suatu tugas atau aksi tertentu. Obyek-obyek tersebut kemudian diurutkan dari kiri ke kanan, aktor yang menginisiasi interaksi biasanya ditaruh di paling kiri dari diagram.

Pada diagram ini, dimensi vertikal merepresentasikan waktu. Bagian paling atas dari diagram menjadi titik awal dan waktu berjalan ke bawah sampai dengan bagian dasar dari diagram. Garis Vertical, disebut *lifeline,* dilekatkan pada setiap obyek atau aktor. Kemudian, *lifeline* tersebut digambarkan menjadi kotak ketika obyek melakukan suatu operasi , kotak tersebut disebut *activation box*. Obyek dikatakan mempunyai *live activation* pada saat tersebut.

2.2.3.8 Definisi State Diagram

Statechart diagram menggambarkan transisi dan perubahan keadaan (dari satu *state* ke *state* lainnya)suatu objek pada sistem sebagai akibat dari *stimuli* yang diterima. Pada u mumnya *statechart diagram* menggambarkan *class* tertentu (satu *class* dapat memiliki lebih dari satu *statechart diagram*).

Dalam UML, *state* digambarkan berbentuk segiempat dengan sudut membulat dan memiliki nama sesuai kondisinya saat itu. Transisi antar *state* umumnya memiliki kondisi *guard* yang merupakan syarat terjadinya transisi yang bersangkutan, dituliskan dalam kurung siku. *Action* yang dilakukan sebagai akibat dari *event* tertentu dituliskan dengan diawali garis miring.

Titik awal dan akhir digambarkan berbentuk lingkaran berwarna penuh dan berwarna setengah.

2.2.4 Basis Data [6]

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasi. *Database* merupakan salah satu komponen yang paling penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi bagi para pemakai. Penerapan *database* dalam sistem informasi disebut *database system.*

Sistem basis data adalah suatu sistem informasi yang mengintegrasikan dari kumpulan data yang saling berhubungan satu dengan yang lainnya dan membuat tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi.

a. Field

   Suatu *field* mengambarkan suatu atribut dari *record* yang menunjukan suatu item dari data, seperti misalnya nama, alamat, dan lain sebagainya. Kumpulan dari *field* membentuk suatu *record.*

b. Record

Kumpulan dari *field* membentuk suatu record. Record menggambarkan suatu unit data individu yang tertentu. Kumpulan dari record membentuk suatu file.

c. File

File terdiri dari record-record yang menggambarkan satu kesatuan dari data-data yang sejenis atau sama.

d. Database

Kumpulan dari file membentuk suatu database. Penyusunan suatu database digunakan untuk mengatasi suatu masalah penyusunan data, yaitu:

1. Kesulitan pengaksesan
2. Banyak pemakai (*multiple user*)
3. Masalah keamanan (*security*)
4. Masalah kesatuan (*integration*)
5. Masalah kebebasan data (*independence*)

2.2.5 Media Jurnal

Jurnal adalah terbitan berkala yang berbentuk pamflet berseri berisi bahan yang sangat diminati orang saat diterbitkan . Bila dikaitkan dengan kata ilmiah di belakang kata jurnal dapat terbitan berarti berkala yang berbentuk pamflet yang berisi bahan ilmiah yang sangat diminati orang saat diterbitkan. (Buku Pegangan Gaya Penulisan, penyunting dan penerbitan Karya Ilmiah Pegangan Gaya Penulisan, Penyunting dan Penerbitan Karya Ilmiah Indonesia, karya Mien A. Rifai, Gajah Mada Uneversity, 1995, h.57-95). [7]

Ada beberapa jenis penerbitan berkala, selain jurnal , yaitu sebagai berikut: [8]

1. Majalah adalah terbitan berkala yang bukan harian, setiap keluar diberi halaman terpisah, biasanya diidentifikasi dengan tanggal dan bukan nomor berseri.
2. Bulletin adalah berkala resmi yang dikeluarkan lembaga atau organisasi profesi ilmiah serta memuat berita, hasil dan laporan kegiatan dalam satu bidang.
3. Warkat Warta, adalah terbitan pendek berisi berita, termasuk kemejuan keilmuan yang berisi catatan singkat yang mengutarakan materi secara umum dan tidak mendalam.

Selain itu, dari sisi teknis isi ada tiga macam berkala ilmiah yaitu pertama majalah teknis ilmiah, kedua berkala semi ilmiah dan ketiga berkala sekunder. Untuk lebih jelasnya akan dijelaskan sebagai berikut.

1. Majalah teknis ilmiah merupakan majalah yang memuat hasil dan temuan baru penelitihan. Berkala ini biasanya sebagai sarana untuk komonikasi para pakar yang terspesialisasi.
2. Berkala semi ilmiah, yaitu berkala yang memuat tulisan teknis dengan cakupan yang bersifat siklopedia dan ditujukan bagi mereka yang bukan ahli atau spesialis dalam bidang yang dimaksud.
3. Berkala sekunder berisi abstrak atau ringkasan majalah primer yang sering disebut pula berkala penyari (abstracting Jurnal).

2.2.6 Jurnal Online [9]

Jurnal Online adalah sarana berbasis *web* untuk mengelola sebuah jurnal ilmiah maupun non-ilmiah. Sarana ini disediakan sebagai wadah bagi pengelola, penulis dan pembaca karya-karya ilmiah.

Sarana jurnal online diadakan dengan motivasi awal untuk membantu komunitas ilmiah Indonesia dalam menerbitkan jurnal-jurnal ilmiah lokal. Karena komunitas yang kecil dan sebagian besar tidak aktif melakukan kajian ilmiah, ditambah biaya penerbitan yang tidak sesuai dengan pangsa pasar, kegiatan enerbitan karya ilmiah lokal menghadapi kendala yang sangat serius dan seringkali berakhir dengan kondisi terbengkalai. Dilain pihak keinginan untuk terus mempertahankan jurnal-jurnal ilmiah lokal sangat tinggi.

2.2.7 *Open Journal System* (OJS) [10]

*Open Journal System*s atau OJS adalah sebuah sistem manajeman konten berbasis web yang khusus dibuat untuk menangani keseluruhan proses manajemen publikasi ilmiah dari proses call for paper, peer review, hingga penerbitan dalam bentuk online. OJS memudahkan peran pengelola jurnal, administrator, kontributor, editor, reviewer, penulis, dan pembaca. OJS seperti dimaksudkan oleh konseptornya adalah menggantikan atau sebagai inovasi model pengelolaan jurnal dari konvesional atau tradisional menjadi model jurnal online.

Open Journal System (OJS) terinstal secara lokal dan lokal dikendalikan.

1. Editor mengkonfigurasi persyaratan, bagian, proses review, dan lain-lain.
2. pengajuan dan pengelolaan semua konten.
3. Berlangganan tertunda modul dengan pilihan akses terbuka.
4. Komprehensif pengindeksan bagian isi dari sistem global.
5. Alat untuk membaca konten, berdasarkan lapangan dan pilihan editor.

6. Email pemberitahuan dan kemampuan untuk komentar pembaca.
7. Lengkapi online konteks-sensitif Bantuan dukungan.

OJS membantu dengan setiap tahap proses penerbitan naskahnya, dari pengiriman melalui publikasi online dan pengindeksan. Melalui sistem manajemen, indexing berbutir halus penelitian, dan menyediakan konteks bagi penelitian, OJS berusaha untuk meningkatkan baik kualitas ilmiah dan umum penelitian naskahnya.

OJS merupakan perangkat lunak open source tersedia bebas untuk jurnal di seluruh dunia untuk tujuan membuat akses terbuka menerbitkan sebuah pilihan yang layak untuk jurnal lebih, seperti akses terbuka dapat meningkatkan jurnal pembaca yang juga sebagai kontribusi untuk kepentingan publik dalam skala global.

2.2.8 Perangkat Lunak Pendukung

Perangkat lunak pendukung yang digunakan untuk membangun sitem jurnal online INKOM LIPI diantaranya yaitu PHP, MySQL, Joomla!, dan WampServer 2.0i.

2.2.8.1 PHP [11]

PHP adalah singkatan dari PHP *Hypertext Prepocessor*, yaitu bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah web dan bisa digunakan pada HTML.

Sebagian besar sintaks mirip dengan bahasa C, Java, asp dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama bahasa ini adalah untuk memungkinkan perancang web untuk menulis halaman web dinamik dengan cepat.PHP banyak dipakai untuk memrogram situs web dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain.Contoh terkenal dari aplikasi PHP adalah phpBB dan MediaWiki. PHP juga dapat dilihat sebagai pilihan lain dari ASP.NET/C#/VB.NET Microsoft, ColdFusion Macromedia, JSP/Java Sun Microsystems, dan CGI/Perl. Contoh aplikasi lain yang lebih kompleks berupa CMS yang dibangun menggunakan PHP adalah Mambo, Joomla!, Postnuke, Xaraya, dan lain-lain.Seluruh aplikasi berbasis web dapat dibuat dengan PHP. Namun kekuatan yang paling utama PHP adalah pada konektivitasnya dengan sistem database yang ada di dalam web. Sistem database yang dapat didukung oleh PHP adalah oracle, MySQL, Ms Access, Sybase dan PostgreSQL.

PHP dapat berjalan di berbagai system operasi seperti windows 98/NT, UNIX/LINUX, solaris maupun macintosh. Keunggulan lainnya dari PHP adalah PHP juga

mendukung komunikasi dengan layanan seperti protocol IMAP, SNMP, NNTP, POP3 bahkan HTTP

1. Sejarah PHP [12]

PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama FI (Form Interpreted), yang wujudnya berupa sekumpulan script yang digunakan untuk mengolah data form dari web.Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI, kependekan dari *Hypertext Preprocessing*/Form Interpreter. Dengan perilisan kode sumber ini menjadi open source, maka banyak programmer yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0.Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek.

2. Kelebihan PHP dari bahasa pemrograman lain

a. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaanya.
b. Web Server yang mendukung PHP dapat ditemukan dimana-mana dari mulai IIS sampai dengan apache, dengan configurasi yang relatif mudah.
c. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan developer yang siap membantu dalam pengembangan.
d. Dalam sisi pemahamanan, PHP adalah bahasa *scripting* yang paling mudah karena referensi yang banyak.
e. PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (linux, unix, windows) dan dapat dijalankan secara runtime melalui *console* serta juga dapat menjalankan perintah-perintah sistem.

2.2.8.2 MySQL [13]

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL.

Tidak sama dengan proyek-proyek seperti Apache, dimana perangkat lunak dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing-masing, MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia MySQL AB, dimana memegang hak cipta hampir atas semua kode sumbernya. Kedua orang Swedia dan satu orang Finlandia yang mendirikan MySQL AB adalah: David Axmark, Allan Larsson, dan Michael "Monty" Widenius.

MySQL adalah Relational Database Management System (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat closed source atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem database (DBMS) dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan database server lainnya dalam query data. Hal ini terbukti untuk query yang dilakukan oleh single user, kecepatan query MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan Interbase. Selain itu MySQL juga memiliki beberapa keistimewaan, antara lain :

a. Portability

MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux,FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.

b. OpenSource

MySQL didistribusikan secara *opensource* (gratis), dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.

c. Multiuser

MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

d. Performance tuning

MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

e. Column types

MySQL memiliki tipe kolom yang sangat kompleks, seperti signed/ unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.

f. Command dan functions

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam query.

g. Security

MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta password terenkripsi.

h. Scalability dan limits

MySQL mampu menangani database dalam skala besar, dengan jumlah records lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

i. Connectivity

MySQL dapat melakukan koneksi dengan client menggunakan protokol TCP/IP, Unix soket (UNIX), atau Named Pipes (NT).

j. Localisation

MySQL dapat mendeteksi pesan kesalahan pada client dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk didalamnya.

k. Interface

MySQL memiliki interface (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface).

l. Clients dan tools

MySQL dilengkapi dengan berbagai tool yang dapat digunakan untuk administrasi database, dan pada setiap tool yang ada disertakan petunjuk online.

m. Struktur table

MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan database lainnya semacam PostgreSQL ataupun Oracle.

MySQL sangat populer dalam aplikasi web seperti MediaWiki (perangkat lunak yang dipakai Wikipedia dan proyek-proyek sejenis) dan PHP-Nuke dan berfungsi sebagai komponen basis data dalam LAMP. Popularitas sebagai aplikasi web dikarenakan kedekatannya dengan popularitas PHP, sehingga seringkali disebut sebagai Dynamic Duo.

Untuk melakukan administrasi dalam basis data MySQL, dapat menggunakan modul yang sudah termasuk yaitu command-line (perintah: mysql dan mysqladmin). Juga dapat diunduh dari situs MySQL yaitu sebuah modul berbasis grafik (GUI): MySQL Administrator dan MySQL Query Browser. Selain itu terdapat juga sebuah perangkat lunak gratis untuk administrasi basis data MySQL berbasis web yang sangat populer yaitu phpMyAdmin. Untuk perangkat lunak untuk administrasi basis data MySQL yang dijual secara komersial antara lain: MySQL front, Navicat dan EMS SQL Manager for MySQL.

2.2.8.3 JOOMLA! [14]

Joomla! adalah Sistem Manajemen Konten (SMK atau CMS) yang bebas dan terbuka (*free opensource*) ditulis menggunakan PHP dan basisdata MySQL untuk keperluan di internet maupun intranet. Joomla! pertama kali dirilis dengan versi 1.0.0. Joomla menyertakan banyak fitur untuk meningkatkan kinerja termasuk *page caching* untuk peningkatan performansi, *web indexing*, RSS freed, halaman untuk dicetak, web side searchingdan dukungan Internasionalisasi, Joomla! menggunakan lisensi GPL.

Untuk membuat suatu Informasi Electronik yang mulai berkembang saat ini diperlukan suatu CMS *Content Management System* / server php yang digunakan untuk pembuatan/desain grafis, dan juga memiliki penulis yang kreatif untuk mengisi halaman-halaman web site dengan Mambo ataupun Joomla, tidak menjadi masalah karena seluruh tool memiliki fungsi yang sama. Meskipun program ini menggunakan bahasa php, bukan berarti harus paham seluruhnya tentang kode php. Bagi mereka tidak memahami kode HTML, Javascript, ASP, CGI, , maupun php tidak perlu takut untuk mempelajarinya karena meskipun tidak tahu tentang kode-kode tersebut, orang bisa membuat sebuah web secara cepat dengan beberapa fitur yang sangat kompleks yang dihubungkan dengan dalam suatu jaringan local atau luas (internet).

Sistem manajemen konten (*Content Management System*, disingkat CMS), adalah perangkat lunak yang memungkinkan seseorang untuk menambahkan dan/atau memanipulasi atau mengubah isi dari suatu situs *Web*.

Joomla sendiri berasal dari kata Swahili yang mengandung arti "kebersamaan". Joomla dibangun sebagai fork dari mambo dengan rilis perdana Joomla 1.0.0 pada tanggal 16

september 2005 mengembangkan versi dari mambo. Menurut roadmap Joomla 2.0 code base akan ditulis ulang sekelompok dibangun mengadopsi GNU General Publik Licence. Secara garis besar Joomla terdiri dari 3 elemen dasar, yaitu server web (webserver), skrip PHP dan basisdata MySQL. Server web diasumsikan terhubung dengan Internet/Intranet yang berfungsi sebagai penyedia layanan situs. Skrip PHP terdiri dari kode program dalam bahasa PHP dan basisdata merupakan tempat penyimpanan konten. Joomla menggunakan Apache sebagai server web dan MySQL untuk basisdatanya.

Pertama kali, pengguna meminta akses terhadap halaman Joomla dengan mengeksekusi URL pada browser web yang kemudian terhubung dengan server web. Permintaan ini yang dalam istilah teknis lebih dikenal dengan query string selain terdapat URL juga mengandung parameter konten (section, category, ID article dan lain-lain). Berdasarkan parameter tersebut, sistem skrip Joomla melakukan kontak dengan basisdata dan mengambil konten yang dimaksud berdasarkan parameternya. Terakhir, konten dan templat (template) digabung bersama dan kembali sebagai halaman html, gambar, css dan javascript.

Paket Joomla terdiri dari beberapa bagian yang terpisah dan termodul yang sangat fleksibel, dapat dengan mudah dikembangkan dan diintegrasikan. Sebagai contoh "plugins" WikiBot yang memperbolehkan penulis di Joomla menggunakan "Wikitags" pada bagian artikel yang fungsinya secara otomatis membuat pranala dinamis ke artikel Wikipedia pada saat ditampilkan. Ada lebih tersedia 1,700 "plugins" yang secara resmi didelegasikan oleh OpenSourceMatters tersedia di http://extensions.joomla.org/ dan saat ini secara resmi dipindahkan ke http://joomlacode.org dengan dukungan server yang lebih lengkap.

Pada tanggal 17 September 2005, diluncurkan Joomla versi awal yaitu 1.0.0 yang merupakan versi alias dari Mambo 4.5.2.3 yang dikembangkan oleh "Pengembang Inti Mambo hingga Agustus 2005" yang hengkang dari Mambo.

Joomla! dapat digunakan untuk membuat beberapa macam situs, yakni sebagai situs portal, komersial, non-profit dan pribadi.

Aplikasi yang beragam dapat dibuat menggunakan Joomla karena memiliki beberapa kemampuan, diantaranya:

1. Dapat melakukan update halaman utama, halaman berita, halaman artikel, maupun halaman arsip secara cepat.
2. Dapat melihat file Microsoft Word, Excel, dan PDF.
3. Dapat menambahkan menu baru pada main menu, other menu, maupun top menu dengan mudah
4. Dapat mengatur weblink dengan cepat dan mudah.

5. Dapat mengatur banner (iklan).
6. Dapat mengatur FAQ (Frequently Asked Questions).
7. Dapat mengatur Newsflashes
8. Membuat arsip dan menampilkannya kembali
9. Dapat mencetak, mengirim artikel lewat email, dan membaca artikel dengan format PDF.
10. Dapat mengatur user dengan tingkat akses level yang berbeda.
11. Dapat menambah komponen, modul, dan template dari pihak ketiga (seperti: forum, calendar, reminder, agenda, guestbook (buku tamu), shopping cart, dan sebagainya).

Desain Joomla berbeda dengan aplikasi sejenis seperti Microsoft Frontpage maupun Macromedia Dreamweaver. Joomla sangat fleksibel, mudah digunakan, dan bentuknya sederhana, menjadikan software ini sangat friend user karena:

1. Mudah melakukan setup/instalasi.
2. Memiliki interface administrator yang sederhana untuk mengatur isi situs (website).
3. Mudah digunakan untuk mengedit (menambah, mengubah, dan menghapus) content dan gambar.
4. Fleksibel untuk mengatur tampilan front end.
5. Fleksibel untuk menambah komponen dan modul yang baru dari pihak ketiga.