BAB II

LANDASAN TEORI

2.1. Aplikasi

Aplikasi merupakan sekumpulan elemen yang saling berinteraksi dan saling berketerkaitan antara satu dengan yang lainnya dalam melakukan suatu kegiatan secara bersama-sama untuk mencapai tujuan tertentu. Sebagai contoh Aplikasi komputer yang terdiri dari *Software*, *Hardware*, dan *Brainware*. Ketiga unsur tersebut saling berkaitan dan bertanggung jawab dalam memproses *Input* (masukan) dan menghasilkan *Output* (keluaran).

2.2. Bencana Alam

Bencana alam adalah konsekuensi dari kombinasi aktivitas alami (suatu peristiwa fisik, seperti letusan gunung, gempa bumi, tanah longsor) dan aktivitas manusia. Karena ketidakberdayaan manusia, akibat kurang baiknya manajemen keadaan darurat, sehingga menyebabkan kerugian dalam bidang keuangan dan struktural, bahkan sampai kematian. Kerugian yang dihasilkan tergantung pada kemampuan untuk mencegah atau menghindari bencana dan daya tahan mereka. Pemahaman ini berhubungan dengan pernyataan: "*bencana muncul bila ancaman bahaya bertemu dengan ketidakberdayaan*". Dengan demikian, aktivitas alam yang berbahaya tidak akan menjadi bencana alam didaerah tanpa ketidakberdayaan manusia, misalnya gempa bumi di wilayah tidak berpenghuni. Konsekuensinya, pemakaian istilah "*alam*" juga ditentang karena peristiwa tersebut bukan hanya bahaya atau malapetaka tanpa keterlibatan manusia.

Besarnya potensi kerugian juga tergantung pada bentuk bahayanya sendiri, mulai dari kebakaran, yang mengancam bangunan individual, sampai peristiwa tubrukan meteor besar yang berpotensi mengakhiri peradaban umat manusia.

Namun demikian pada daerah yang memiliki tingkat bahaya tinggi (*hazard*) serta memiliki kerentanan/kerawanan (*vulnerability*) yang juga tinggi tidak akan memberi dampak yang hebat/luas jika manusia yang berada disana memiliki ketahanan terhadap bencana (*disaster resilience*). Konsep ketahanan bencana merupakan valuasi kemampuan sistem dan infrastruktur-infrastruktur untuk mendeteksi, mencegah dan menangani tantangantantangan serius yang hadir. Dengan demikian meskipun daerah tersebut rawan bencana dengan jumlah penduduk yang besar jika diimbangi dengan ketetahanan terhadap bencana yang cukup (Khairul Munadi : 2011).

2.3. *Android*

Android adalah sistem operasi untuk telepon seluler yang berbasis *Linux*. *Android* menyediakan platform terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Awalnya, *Google Inc*. membeli *Android Inc*., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan *Android*, dibentuklah *Open Handset Alliance, konsorsium* dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk *Google, HTC, Intel, Motorola, Qualcomm, T-Mobil*e, dan *Nvidia*

Beberapa versi dari *android* yang sudah diluncurkan diantaranya:

1. **Andorid versi 1.1**

   Pada 9 Maret 2009, Google merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam alarm, *voice search* (pencarian suara), pengiriman pesan dengan Gmail, dan pemberitahuan email.

2. **Android versi 1.5 (*Cupcake*)**

   Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (*Software Development Kit*) dengan versi 1.5 (*Cupcake*). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengunggah video ke Youtube dan gambar ke Picasa langsung dari telepon, dukungan Bluetooth A2DP, kemampuan terhubung secara otomatis ke *headset* Bluetooth, animasi layar, dan keyboard pada layar yang dapat disesuaikan dengan sistem.

3. **Android versi 1.6 (*Donut*)**

   Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus; kamera, camcorder dan galeri yang dintegrasikan; CDMA / EVDO, 802.1x, VPN, Gestures, dan *Text-to-speech engine*; kemampuan dial

kontak; teknologi *text to change speech* (tidak tersedia pada semua ponsel; pengadaan resolusi VWGA.

4. **Android versi 2.0/2,1 (*Eclair*)**

   Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (*Eclair*), perubahan yang dilakukan adalah pengoptimalan hardware, peningkatan Google Maps 3.1.2, perubahan UI dengan browser baru dan dukungan HTML5, daftar kontak yang baru, dukungan *flash* untuk kamera 3,2 MP, digital Zoom, dan Bluetooth 2.1. Untuk bergerak cepat dalam persaingan perangkat generasi berikut, Google melakukan investasi dengan mengadakan kompetisi aplikasi mobile terbaik (*killer apps* - aplikasi unggulan). Kompetisi ini berhadiah $25,000 bagi setiap pengembang aplikasi terpilih. Kompetisi diadakan selama dua tahap yang tiap tahapnya dipilih 50 aplikasi terbaik. Dengan semakin berkembangnya dan semakin bertambahnya jumlah handset Android, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka kepada sistem operasi Android. Aplikasi terkenal yang diubah ke dalam sistem operasi Android adalah *Shazam, Backgrounds*, dan *WeatherBug*. Sistem operasi Android dalam situs Internet juga dianggap penting untuk menciptakan aplikasi Android asli, contohnya oleh MySpace dan Facebook.

5. **Android versi 2.2 (*Froyo: Frozen Youghurt*)**

   Pada 20 Mei 2010, Android versi 2.2 (*Froyo*) diluncurkan. Perubahan-perubahan umumnya terhadap versi-versi sebelumnya antara lain

dukungan *Adobe Flash 10.1*, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, *intergrasi V8 JavaScript engine* yang dipakai Google Chrome yang mempercepat kemampuan rendering pada *browser*, pemasangan aplikasi dalam SD Card, kemampuan WiFi Hotspot portabel, dan kemampuan auto update dalam aplikasi *Android Market*.

6. **Android versi 2.3 (*Gingerbread*)**

   Pada 6 Desember 2010, Android versi 2.3 (*Gingerbread*) diluncurkan. Perubahan-perubahan umum yang didapat dari Android versi ini antara lain peningkatan kemampuan permainan (*gaming*), peningkatan fungsi copy paste, layar antar muka (*User Interface*) didesain ulang, dukungan format video VP8 dan WebM, efek audio baru (*reverb, equalization, headphone virtualization,* dan *bass boost*), dukungan kemampuan *Near Field Communication (NFC)*, dan dukungan jumlah kamera yang lebih dari satu

7. **Android versi 3.0 (*Honeycomb*)**

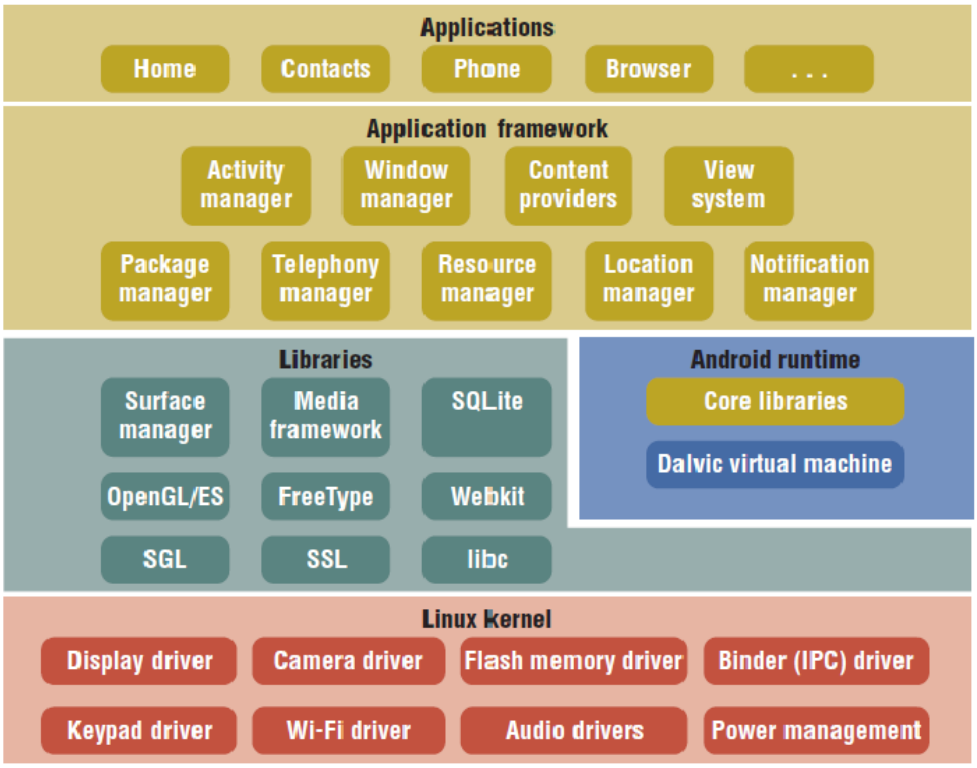
   Android Honeycomb dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. *User Interface* pada Honeycomb juga berbeda karena sudah didesain untuk *tablet*. Honeycomb juga mendukung multi prosesor dan juga akselerasi perangkat keras (*hardware*) untuk grafis. Tablet pertama yang dibuat dengan menjalankan Honeycomb adalah Motorola Xoom.

2.3.1. Features Android

Features yang tersedia pada platform Android saat ini antara lain :

a. Framework Aplikasi yang mendukung penggantian komponen dan reusable.
b. Mesin Virtual Dalvik berjalan diatas Linux kernel dan dioptimalkan untuk perangkat mobile
c. Integrated browser berdasarkan open source engine WebKit
d. Grafis yang dioptimalkan dan didukung oleh library grafis 2D yang terkostumisasi, grafis 3D berdasarkan spesifikasi openGL ES 1,0 (*Opsional akselarasi hardware*)
e. SQLite untuk penyimpanan data
f. Media support yang mendukung audio, video, dan gambar (MPGE, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
g. GSM Telephony (tergantung *hardware*)
h. Bluetooth, EDGE, 3G, dan WIFI (tergantung *hardware*)
i. *Multi-touch* : kemampuan layaknya handset modern yang dapat menggunakan dua jari atau lebih untuk berinteraksi dengan perangkat
j. Lingkungan Development yang lengkap dan kaya termasuk perangkat emulator, tools untuk debugging, profil, dan kinerja memori, dan plugin untuk Eclips IDE
k. *Market* : Seperti kebanyakan Handphone yang memiliki tempat penjualan aplikasi, market pada android merupakan katalog aplikasi yang dapat di download dan di install pada Handphone melalui internet

2.3.2 Arsitektur Android



Gambar 2. 1 Arsitektur Andoid

2.3.2.1. Linux Kernel

Android bukan Linux, akan tetapi android dibangun diatas Linux Kernel yaitu versi 2.6 sehingga kehandalannya dapat di percaya. Untuk inti sistem service linux yang digunakan seperti keamanan, manajemen memori, proses manajemen, network, dan drive model. Seperti yang terlihat digambar, Linux Kernel menyediakan Driver Layer, Kamera, Keypad, Kamera, Wifi, Flash Memory, Audio, dan IPC (*Interprocess Communication*) untuk mengatur aplikasi dan keamanan. Kernel juga bertindak sebagai lapisan abstrak antara hardware dan software stacknya 20

2.3.2.2. Libraries

Android menyertakan libraries C/C++ yang digunakan oleh berbagai komponen dari sestem Android. Kemampuan ini disediakan kepada Developer aplikasi melalui framework aplikasi Android. beberapa inti libraries tercantum di bawah ini :

a. System C library – Variasi dari implementasi BSD-berasal pelaksanaan sistem standar C library (libe), sesuai untuk perangkat embedded berbasis linux.
b. Media libraries – PaketVideo berdasarkan OpenCORE; library mendukung pemutaran rekaman dan populer banyak format audio dan video, serta file gambar, termasuk MPEG, H.264, MP3, AAC, AMR, JPG. Dan PNG
c. Surface Manager – mengelola akses ke subsistem layar
d. LibWebCore – mesin web modern yang powerfull yang baik browser Android embeddable web.
e. SGL – mesin grafis 2D
f. 3D libraries – sebuah pelaksanaan berdasarkan openGL ES 1,0 API; perpustakaan baik menggunakan perangkat keras akselarasi 3D (apabila tersedia) atau yang disertakan, sangat optimal 3D software rasterizer
g. FreeType – bitmap dan vektor font rendering
h. SQLite – mesin database yang kuat dan ringan, dan penghubung tersedia untuk semua aplikasi.

2.3.2.3. Android-Runtime

Android terdiri satu set core libraries yang menyediakan sebagian besar fungsi yang sama dengan yang terdapat pada core libraries bahasa pemograman java. Setiap aplikasi menjalankan prosesnya sendiri dalam Android. Setiap alikasi menjalankan prosesnya sendiri dalam Android, dengan masing-masing instan dari mesin virtual Dalvik (Dalvik VM). Dalvik dirancang agar perangkat dapat menjalankan multiple VMs secara efisien. Mesin Virtual Dalvik mengeksekusi file dalam Dalvik executable (.dex), sebuah format yang dioptimalkan untuk memori yang kecil. Dalvik VM berbasis, berjalan dan dikompilasi oleh compiler bahasa Java yang telah ditransformasikan ke dalam .dex format oleh tool "dx" yang telah di sertakan, Dalvik VM bergantung pada kernel Linux untuk berfungsi seperti threading dan manajemen memori tingkat rendahnya.

2.3.2.4. Framework-Aplikasi

Pengembang memiliki akses penuh menuju API framework yang sama, dengan yang digunakan oleh aplikasi inti. Arsitektur aplikasi dirancang agar komponen dapat digunakan kembali (reuse) denga mudah. Setiap aplikasi dapat memanfaatkan kemampuan ini dan aplikasi yang lain mungkin akan memanfaatkan kemampuan ini (sesuai denganbatasan keamanan yang didefinisikan oleh framework). Mekanisme yang sama memungkinkan komponen untuk diganti oleh pengguna.

Semua pengguna merupakan rangkaian set layanan dan sistem, termasuk:

a. *View* yang kaya dan *extensible* yang dapat digunakan untuk membangun aplikasi, termasuk list, grid, kotak teks, tombol, dan bahkan sebuah embeddable web
b. *Content provider* yang memungkinkan aplikasi untuk mengakses data (seperti dari daftar kontak help) atau membagi data tersebut.
c. Resource Manager, yang menyediakan akses ke kode non sumber lokal seperti string, gambar, dan tata letak file
d. *Notification Manager* yang memungkinkan semua aplikasi menampilkan alert yang bisa dokostumisasi didalam status bar.
e. *Activity Manager* yang mengelola siklus hidup aplikasi dan menyediakan navigasi umum backstack.

2.3.2.5. Application

Android telah menyertakan aplikasi inti seperti email elient, SMS, kalender, peta, browser, kontak, dan lain-nya. Semua aplikasi tersebut ditulis dengan menggunakan bahasa pemrograman Java. Pada layer inilah developer atau kita menempatkan aplikasi yang dibuat. Yang istimewa adalah pada android semua aplikasi baik aplikasi inti (*native*) maupun aplikasi pihak ketiga berjalan pada layer aplikasi yang dibuat untuk android akan memiliki akses yang setara dalam mengakses seluruh kemampuan handset, tanpa membedakan apakah itu merupakan aplikasi inti atau aplikasi pihak ketiga. Dalam kata lain dengan platform android ini, Programmer atau developer secara penuh akan bisa mengkostumisasi perangkat androidnya.

2.4. Java

Menurut Abdul Kadir (2004:2) Java adalah bahasa pemrograman. Menurut Fredy Purnomo dkk (2006:166) Java adalah duatu platform teknologi yang dikembangkan oleh Sun Microsystem sekitar tahun90-an.

Alasan utama pembentukan bahasa java adalah untuk membuat aplikasiaplikasi yang dapat diletakkan diberbagai macam perangkat elektronik, seperti microwave oven dan remote control, sehingga Java harus bersifat portable atau yang sering disebut dengan platform independent (tidak tergantung pada *platform*). Itulah yang menyebabkan dalam dunia pemrograman Java, dikenal adanya istilah '*write once, run everywhere*', yang berarti kode program hanya ditulis sekali, namun dapat dijalankan dibawah platform manapun, tanpa harus melakukan perubahan kode program.

2.5. Eclipse

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platform-independent*). Berikut ini adalah sifat dari Eclipse:

1. Multi-platform: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
2. Mulit-language: Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.

3. Multi-role: Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari Eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in*.

2.6. *Software Development Kit* (SDK)

Software Development Kit (SDK) adalah suatu kit atau library dari bahasa pemrograman untuk pengembangan atau pembangunan suatu perangkat lunak dan biasanya SDK terdiri dari kumpulan tools yang dibutuhkan. Misalnya bahasa pemrograman java, mempunyai SDK yang berisi suatu *library* yang dapat digunakan untuk membuat suatu aplikasi berbasis java.

2.7. *Java Development Kit* (JDK)

Java Development Kit (JDK) adalah sekumpulan perangkat lunak yang dapat kamu gunakan untuk mengembangkan perangkat lunak yang berbasis Java, Sedangkan JRE adalah sebuah implementasi dari *Java Virtual Machine* yang benar-benar digunakan untuk menjalankan program java.Biasanya, setiap JDK berisi satu atau lebih JRE dan berbagai alat pengembangan lain seperti sumber kompiler java, bundling, debuggers, development libraries dan lain sebagainya.

Perbedaan JDK dengan SDK *(Software Development Kit)* yaitu JDK adalah sebuah SDK tetapi sebuah SDK tidak harus menjadi sebuah JDK.

2.8. ADT

Android Development Tools (*ADT*) adalah *plugin* untuk Eclipse *Intergrated Development Environment* (*IDE*) yang dirancang untuk memberikan lingkungan yang terpadu di mana untuk membangun aplikasi Android.

ADT memperluas kemampuan Eclipse untuk membiarkan para *developer* lebih cepat dalam membuat proyek baru Android, membuat aplikasi UI, menambahkan komponen berdasarkan Android *Framework* API, *debug* aplikasi dalam pengunaan Android SDK, dan membuat file APK untuk mendistribusikan aplikasi.

Mengembangkan aplikasi di Eclipse dengan ADT sangat dianjurkan dan merupakan cara tercepat untuk memulai membuat aplikasi android, karena banyak kemudahan-kemudahan sebagai tools yang terintegrasi seperti, custom XML editor, dan *debug panel ouput*. Selain itu ADT memberikan dorongan luar biasa dalam mengembangkan aplikasi Android.

2.9. XML (*Extensible Markup Language*)

XML (*eXtensible Markup Language*) dikembangkan mulai tahun 1996 dan mendapatkan pengakuan dari W3C pada bulan Februari 1998. Teknologi yang digunakan pada XML sebenarnya bukan teknologi baru, tapi merupakan turunan dari SGML yang telah dikembangkan pada awal 80-an dan telah banyak digunakan pada dokumentasi teknis bebagai proyek berskala besar. Ketika HTML dikembangkan pada tahun 1990, para penggagas XML mengadopsi bagian paling

penting pada SGML dan dengan berpedoman pada pengembangan HTML menghasilkan markup language yang tidak kalah hebatnya dengan SGML.

Seperti halnya HTML, XML juga menggunakan elemen yang ditandai dengan tag pembuka (diawali dengan „<‟ dan diakhiri dengan „>‟), tagpenutup (diawali dengan „</ „diakhiri „>‟) dan atribut elemen (parameter yang dinyatakan dalam tag pembuka misal <form name=”isidata”>). Hanya bedanya, HTML medefinisikan dari awal tag dan atribut yang dipakai didalamnya, sedangkan pada XML kita bisa menggunakan tag dan atribut sesuai kehendak kita.

XML untuk saat ini bukan merupakan pengganti HTML. Masing-masing dikembangkan untuk tujuan yang berbeda.Kalau HTML digunakan untuk menampilkan informasi dan berfokus pada bagaimana informasi terlihat, XML mendeskripsikan susunan informasi dan berfokus pada informasi itu sendiri.XML terutama dibutuhkan untuk menyusun dan menyajikan informasi dengan format yang tidak mengandung format standard layaknya heading, paragraf, tabel dan lain sebagainya.

2.10. Google Maps

Google Maps merupakan sebuah layanan peta dunia virtual berbasis web yang disediakan oleh Google. Layanan ini gratis dan dapat ditemukan di http://maps.google.com.

Google Maps menawarkan peta yang dapat digeser (*panned*), diperbesar (*zoom in*), diperkecil (*zoom out*), dapat diganti dalam beberapa mode (*map, satelit, hybrid*, dan lain-lain), fitur pencarian rute (*routing*), penunjuk arah dari satu objek peta ke objek yang lain (*direction*) dan juga pencarian tempat (*place*). Sampai saat

ini, Google Maps masih berada dalam tahap beta, dan masih terus dikembangkan dengan data yang selalu diperbarui secara berkala.

Google maps merupakan hak cipta Google secara *propriety*, sehingga dalam menggunakannya memerlukan adanya perjanjian, Google membuat mekanisme untuk dapat mengakses *Google Maps* dengan coding aplikasi dengan sebuah kunci yang dikenal dengan *API Key.*

2.11. MySQL

MySQL adalah salah satu jenis database server yang sangat terkenal, kepopulerannya disebabkan MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya. Selain itu , ia bersifat Open source (Anda tidak perlu membayar untuk menggunakannya) pada pelbagai platform (kecuali untuk jenis enterprise, yang bersifat komersial). Perangkat lunak MySQL sendiri bisa di download dari *http://www.mysql.com*

MySQL termasuk jenis RDBMS (*Relational Database Management*). Itulah sebabnya, istilah seperti tabel, baris, dan kolom digunakan pada MySQL. Pada MySQL, sebuah database mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom.

2.12. PHP

Menurut dokumen resmi PHP, PHP merupakan singkatan dari PHP Hypertext Preprocessor. Ia merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses di server. Hasilnya yang dikirim ke klien tempat pemakai menggunakan *browser*.

Secara khusus, PHP dirancang untuk membentuk aplikasi web dinamis. Artinya, ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, Anda bisa menampilkan isi *database* ke halaman web. Pada prinsipnya PHP mempunyai fungsi yang sama dengan skrip-skrip seperti ASP (*Active Server Page*), *Cold Fusion*, ataupun *Perl*. Namun, perlu diketahui bahwa PHP sebenernya

bisa dipakai secara command line. Artinya, skrip PHP dapat dijalankan tanpa melibatkan *web server* maupun *browser*.

Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah skrip Perl yang dapat mengamati siapa saja yang melihat-lihat daftar riwayat hidupnya, yakni pada tahun 1994. Skrip-skrip ini selanjutnya dikemas menjadi *tool* yang disebut "*Personal home page*". Paket inilah yang menjadi cikal-bakal PHP. Pada tahun 1995, Rasmus menciptakan PHP/FI Versi 2. Pada versi inilah pemrograman dapat menempelkan kode terstruktur di dalam tag HTML. Yang menarik, kode PHP juga bisa berkomunikasi dengan database dan melakukan perhitungan-perhitungan kompleks sambil jalan.

Pada saat ini PHP cukup popular sebagai peranti pemrograman Web, terutama di lingkungan Linux. Walaupun demikian, PHP sebenarnya juga dapat berfungsi pada *server-server* yang berbasis *UNIX*, *Windows*, dan *Machintosh*.

Pada awalnya, PHP dirancang untuk diintegrasikan dengan web server Apache. Namun, belakangan PHP juga dapat bekerja dengan web server seperti PWS (*Personal Web Server*), IIS (*Internet Information Server*), dan Xitami.

2.13. Internet

Internet merupakan media yang akan digunakan sebagai media interaksi anatara pihak merchant dan konsumen. Untuk dapat menjalankan bisnis online, maka merchant harus melakukan koneksi terhadap internet. Demikian pula 24 konsumen yang akan membeli produk atau jasa secara elektronik harus menyambungkan komputernya ke internet melaui ISP (*Internet Service Provider*) tertentu.

2.13.1. Pengertian Internet

Internet adalah sistem informasi global berbasis komputer. Internet terbentuk dari jaringan-jaringan komputer-komputer yang terkoneksi satu sama lain. Setia jaringa mungkin menghubungkan puluhan, ratusan, hingga ribuan komputer, yang memungkinkan komputer-komputer itu saling berbagi data dan informasi satu sama lain dan untuk saling berbagi sumber daya komputasi yang mahal dan langka seperti komputer-komputer super (*SuperComputer*) yang sangat kuat dan sistem-sistem basis data yang berisi informasi-informasi yang bermanfaat.

2.13.2. Perkembangan Internet

Penggunaan internet berkembang pesat sejak penemuannya. Alih-alih menghubungkan jaringan-jaringan secara terbatas pada tipe komputer tertentu, teknologi internet memungkinkan koneksi terjadi diantara berbagai jenis komputer. Antar berbagai sistem operasi (baca:tidak terikat *platform* perangkat keras maupun perangkat lunak). Tidak ada jaringan yang terlalu cepat atau lambat, terlalu besar atau terlalu kecil sehingga tidak bisa dikoneksikan. Internet dapat

menghubungkan jaringan-jaringan sederhana yang terdiri dari beberapa komputer di suatu ruangan hingga jaringan-jaringan canggih yang merentang antar benua dan menghubungkan ribuan (bahkan jutaan) komputer.

2.14. UML

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. UML adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat *tool* untuk mendukung pengembangan sistem tersebut. UML mulai diperkenalkan oleh *Object Management Group*, sebuah organisasi yang telah mengembangkan model, teknologi, dan standar OOP sejak tahun 1980-an. Sekarang UML sudah mulai banyak digunakan oleh para praktisi OOP. UML merupakan dasar bagi perangkat (*tool*) desain berorientasi objek dari IBM.UML adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh *Grady Booch*, *Jim Rumbaugh*, dan *Ivar Jacobson*. Namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi. Penggunaan UML dalam industri terus meningkat. Ini merupakan standar terbuka yang menjadikannya sebagai bahasa pemodelan yang umum dalam industri peranti lunak dan pengembangan sistem.

2.14.1. Sejarah UML

Sampai era tahun 1990 puluhan metodologi pemodelan berorientasi objek telah bermunculan di dunia. Diantaranya adalah: metodologi *booch*, metodologi *coad*, metodologi *OOSE*, metodologi *OMT*, metodologi *shlaer-mellor*, metodologi *wirfs-brock*, dan sebagainya. Masa itu terkenal dengan masa perang metodologi (*method war*) dalam pendesainan berorientasi objek. Masing-masing metodologi membawa notasi sendiri-sendiri, yang mengakibatkan timbul masalah baru apabila kita bekerjasama dengan kelompok/perusahaan lain yang menggunakan metodologi yang berlainan.

Dimulai pada bulan Oktober 1994 *Booch*, *Rumbaugh* dan *Jacobson*, yang merupakan tiga tokoh yang boleh dikata metodologinya banyak digunakan mempelopori usaha untuk penyatuan metodologi pendesainan berorientasi objek. Pada tahun 1995 *direlease draft* pertama dari UML (versi 0.8). Sejak tahun 1996 pengembangan tersebut dikoordinasikan oleh *Object Management Group*.

2.14.2. Diagram UML

UML menyediakan 10 macam diagram untuk memodelkan aplikasi berorientasi objek, yaitu:

1. *Use Case Diagram* untuk memodelkan proses bisnis.
2. *Conceptual Diagram* untuk memodelkan konsep-konsep yang ada di dalam aplikasi.
3. *Sequence Diagram* untuk memodelkan pengiriman pesan (*message*) antar *objects*
4. *Collaboration Diagram* untuk memodelkan interaksi antar *objects*.

5. *State Diagram* untuk memodelkan perilaku *objects* di dalam sistem.
6. *Activity Diagram* untuk memodelkan perilaku *Use Cases* dan *objects* di dalam *system*.
7. *Class Diagram* untuk memodelkan struktur kelas.
8. *Object Diagram* untuk memodelkan *struktur object*.
9. *Component Diagram* untuk memodelkan komponen object.
10. *Deployment Diagram* untuk memodelkan distribusi aplikasi.

Berikut akan dijelaskan empat macam diagram yang paling sering digunakan dalam pembangunan aplikasi berorientasi object, yaitu *use case diagram, sequence diagram, collaboration diagram,* dan *class diagram.*

1. Use Case Diagram

Use case diagram digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. *Use case* diagram terdiri atas diagram untuk *use case* dan *actor*. *Actor* merepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi.

Use case merepresentasikan operasi-operasi yang dilakukan oleh aktor. *Use case* digambarkan berbentuk elips dengan nama operasi dituliskan di dalamnnya. *Actor* yang melakukan operasi dihubungkan dengan garis lurus ke usecase.

2. Sequence Diagram

Sequence diagram menjelaskan secara detil urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari *use case*: interaksi yang terjadi antar

class, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi.

3. Collaboration Diagram

Collaboration diagram dipakai untuk memodelkan interaksi antar objek di dalam sistem. Berbeda dengan sequence diagram yang lebih menonjolkan kronologis dari operasi-operasi yang dilakukan, *collaboration diagram* lebih fokus pada pemahaman atas keseluruhan operasi yang dilakukan oleh objek.

4. Class Diagram

Class diagram merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. *Class diagram* menunjukkan hubungan antar class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan.