BAB II

LANDASAN TEORI

Pada Bab II akan menjelaskan beberapa konsep yang berhubungan dengan permasalahan di dalam penelitian yang dimana akan digunakan sebagai dasar pemahaman dalam mengimplementasikan konsep-konsep kedalam semua kegiatan pembangunan sebuah aplikasi

2.1 Sistem Informasi

Pengertian Sistem Informasi dibangun oleh dua unsur utama yaitu sistem dan informasi. Untuk lebih memudahkan pemahaman mengenai Sistem Informasi, maka pengertian mengenai sistem, informasi, dan sistem informasi akan diuraikan sebagai berikut :

2.1.1 Pengertian Sistem

Menurut Jimmy L.Goal (2008:9), “sistem adalah hubungan satu unit dengan unit-unit lainnya yang saling berhubungan satu sama lainnya dan yang tidak dapat dipisahkan serta menuju satu kesatuan dalam rangka mencapai tujuan yang telah ditetapkan. Apabila suatu unit macet atau terganggu, unit lainnya pun akan terganggu untuk mencapai tujuan yang telah ditetapkan tersebut”.

2.1.2 Karakteristik Sistem

Menurut Agus Mulyanto (2009:2), sistem mempunyai karakteristik sebagai berikut:

1. **Mempunyai Komponen Sistem (*Components Sistem*)**

Suatu sistem tidak berada dalam lingkungan yang kosong, tetapi sebuah sistem berada dan berfungsi di dalam lingkungan yang berisi sistem lainnya. Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan. Apabila suatu sistem merupakan salah satu dari komponen sistem lain yang lebih besar, maka akan disebut dengan *subsistem* , sedangkan sistem yang lebih besar tersebut adalah lingkungannya.

2. **Mempunyai Batasan Sistem (*Boundary*)**

Batas sistem merupakan pembatas atau pemisah antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya.

3. **Mempunyai Lingkungan (*Environment*)**

Lingkungan luar adalah apa pun di luar batas dari sistem yang dapat mempengaruhi operasi sistem, baik pengaruh yang menguntungkan ataupun yang merugikan. Pengaruh yang menguntungkan ini tentunya harus dijaga sehingga akan mendukung kelangsungan operasi sebuah sistem. Sedangkan lingkungan yang merugikan harus ditahan dan dikendalikan agar tidak mengganggu kelangsungan sebuah sistem.

4. Mempunyai Penghubung (*interface*) Antar Komponen

Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Penghubung inilah yang akan menjadi media yang digunakan data dari masukan (*input*) hingga keluaran (*output*). Dengan adanya penghubung, suatu subsistem dapat berinteraksi dan berintegrasi dengan subsistem yang lain membentuk satu kesatuan.

5. Mempunyai Masukan (*input*)

Masukan atau input merupakan energi yang dimasukan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*), yaitu bahan yang dimasukkan agar sistem tersebut dapat beroperasi dan masukan sinyal (*signal input*), yaitu masukan yang diproses untuk mendapatkan keluaran.

6. Mempunyai Pengolahan (*processing*)

Pengolahan (*process*) merupakan bagian yang melakukan perubahan dari masukan untuk menjadi keluaran yang diinginkan.

7. Mempunyai Sasaran (*Objective*) dan Tujuan

Suatu sistem pasti memiliki sasaran (*objective*) atau tujuan *(goal*). Apabila sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Tujuan inilah yang mengarahkan suatu sistem. Tanpa adanya tujuan, sistem menajdi tidak terarah dan terkendali.

8. Mempunyai Keluaran (*output*)

Keluaran (*output*) merupakan hasil dari pemrosesan. Keluaran dapat berupa informasi sebagai masukan pada sistem lain atau hanya sebagai sisa pembuangan.

9. Mempunyai Umpan Balik (*Feed Back*)

Umpan balik diperlukan oleh bagian kendali (*Control*) sistem untuk mengecek terjadinya penyimpangan proses dalam sistem dan mengembalikannya ke dalam kondisi normal.

2.1.3 Klasifikasi Sistem

Menurut Agus Mulyanto (2009:8), "Sistem dapat di klasifikasikan berbagai sudut pandang, diantaranya adalah sebagai berikut:

1. Sistem abstak (*abstract system*) dan sistem fisik (*physical system*).

   Sistem abstrak (*abstract system*) adalah sistem yang berupa pemikiran atau gagasan yang tidak tampak secara fisik. Sedangkan sistem fisik (*physical system)* adalah sistem yang ada secara fisik dan dapat dilihat dengan mata.
2. Sistem alamiah (*natural system*) dan sistem buatan manusia (*human made system*).

   Sistem alamiah adalah sistem yang keberadaannya terjadi karena proses alam, bukan buatan manusia. Sedangkan sistem buatan manusia (*human*

made systems) adalah sistem yang terjadi melalui rancangan atau campur tangan manusia.

3. Sistem tertentu (*deterministic system*) dan sistem tak tentu (*probabilistic system*)

Sistem tertentu (*deterministic systems*) yaitu sistem yang operasinya dapat diprediksi secara cepat dan interaksi diantara bagian-bagiannya dapat dideteksi dengan pasti. Sedangkan sistem tidak tentu (*probabilistic systems*) yaitu sistem yang hasilnya tidak dapat diprediksi karena mengandung unsur probabilitas.

4. Sistem tertutup (*closed system*) dan sistem terbuka (*open system*).

Sistem tertutup (*closed systems*) yaitu sistem yang tidak berhubungan dengan lingkungan di luar sistem. Sistem ini tidak berinteraksi dan tidak dipengaruhi oleh lingkungan luar. Sistem ini juga bekerja secara otomatis tanpa adanya campur tangan dari pihak luar. Dalam kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada hanyalah sistem yang relatif tertutup (*relative closed system*). Sistem relatif tertutup biasanya mempunyai masukan dan keluaran yang tertentu serta tidak terpengaruh oleh keadaan di luar sistem. Sedangkan sistem terbuka (*open system*) adalah sistem yang berhubungan dengan lingkungan luar dan dapat terpengaruh dengan keadaan lingkunga luar. Sistem terbuka menerima *input* dari subsistem lain dan menghasilkan *output* untuk subsistem lain. Sistem ini mampu beradaptasi

dan memiliki sistem pengendalian yang baik karena lingkungan luar yang bersifat merugikan dapat mengganggu jalannya proses di dalam sistem.

2.1.4 Pengertian lnformasi

Telah diketahui bahwa informasi merupakan hal yang sangat penting bagi manajemen dalam pengambilan keputusan. Beberapa ahli mendefinisikan informasi sebagai berikut:

Menurut Agus Mulyanto (2009 : 12), “Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya, sedangkan data merupakan sumber informasi yang menggambarkan suatu kejadian yang nyata”.

2.1.5 Siklus Informasi

Data merupakan fakta atau kejadian yang belum berguna bagi penerimanya, sehingga perlu diolah lebih lanjut. Data diolah melalui dengan suatu metode pendekatan dan pengembangan tertentu untuk dihasilkan suatu informasi.

Data diolah sehingga menghasilkan informasi, kemudian penerima menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat suatu data kembali. Data tersebut akan ditangkap sebagai input, diproses kembali lewat sebuah model dan seterusnya membentuk suatu siklus. Siklus ini oleh John Burch disebut dengan siklus

informasi (*information cycle*), siklus ini disebut juga dengan siklus pengolahan data (*data processing cycle*).

2.1.6 Kualitas dari Informasi

Menurut Agus Mulyanto (2009 :20), Kualitas informasi bergantung pada 3 (tiga) hal yang sangat domain yaitu:

1. Informasi harus akurat.

   Sebuah informasi harus akurat karena dari sumber informasi hingga penerima informasi kemungkinan banyak terjadi gangguan yang dapat mengubah atau merusak informasi tersebut. Informasi dikatakan akurat apabila informasi tersebut tidak bias atau menyesatkan, bebas dari kesalahan-kesalahan dan harus jelas mencerminkan maksudnya.

2. Informasi harus tepat waktu.

   Informasi yang dihasilkan dari suatu proses pengolahan data, datangnya tidak boleh terlambat (usang). Informasi yang terlambat tidak akan mempunyai nilai yang baik, karena informasi merupakan landasan dalam pengambilan keputusan.

3. Informasi harus relevan.

   Informasi dikatakan berkualitas jika relevan bagi pemakainya. Hal ini berarti bahwa informasi tersebut harus bermanfaat bagi pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda.

2.1.7 Nilai Informasi

Nilai dari suatu informasi ditentukan oleh dua hal yaitu manfaat dan biaya pendapatannya, suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Kegunaan informasi adalah untuk mengurangi hal ketidakpastian didalam proses pengambilan keputusan tentang suatu keadaan.

2.1.8 Pengertian Sistem Informasi

Menyangkut pemahaman tentang pengertian sistem informasi ini, dalam bukunya Agus Mulyanto (2009:29) mengutipkan beberapa pendapat para ahli, diantaranya:

1. Menurut James alter, sistem informasi adalah "kombinasi antar prosedur kerja, informasi, orang dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi".

2. Menurut Bodnar dan Hopwood, sistem informasi adalah "kumpulan perangkat keras dan perangkat lunak yang dirancang untuk mentransformasikan data ke dalam bentuk informasi yang berguna".

3. Menurut Gelinas, Oram dan Wiggins, sistem informasi adalah "suatu sistem buatan manusia yang secara umum terdiri atas sekumpulan komponen berbasis komputer dan manual yang dibuat untuk menghimpun, menyimpan, dan mengelola data serta menyediakan informasi keluaran kepada pemakai".

4. Menurut Turban, McLean dan Waterbe, sistem informasi adalah "sistem yang mengumpulkan, memproses, menyimpan, menganalisis, dan mneyebarkan informasi untuk tujuan spesifik".

5. Menurut Joseph Wilkinson, sistem informasi adalah "kerangka kerja yang mengkoordinasikan sumber daya (manusia, komputer) untuk mengubah masukan (*input*) menjadi keluaran (informasi), guna mencapai sasaran-sasaran perusahaan".

Dari beberapa definisi diatas, dapat disimpulkan bahwa sistem informasi adalah suatu sistem yang terdiri dari kumpulan komponen sistem, yaitu *software, hardware* dan *brainware* yang memproses informasi menjadi sebuah *output* yang berguna untuk mencapai suatu tujuan tertentu dalam suatu organisasi.

2.2 Sistem Informasi Nilai Raport

Nilai diartikan sebagai harga, dalam hal ini adalah suatu angka kepandaian siswa dalam pembelajaran. Sedangkan Raport adalah buku yang berisi keterangan mengenai nilai kepandaian dan prestasi belajar murid di sekolah, yang biasanya dipakai sebagai laporan guru kepada orang tua siswa atau wali murid. Di dalam buku raport terdiri dari halaman profil siswa, dan halaman nilai siswa dimana ddi dalam nilai raport itu terdapat nilai mata pelajaran , nilai sikap ,ketidakhadiran, catatan pengembangan diri dan catatan dari wali kelas .

Jadi bisa di simpulkan bahwa Sistem Informasi Nilai Raport adalah suatu sistem layanan yang memberikan informasi mengenai nilai raport yang berisi keterangan prestasi, baik itu akademik, sikap, maupun ekstrakulikuler siswa di suatu sekolah .

2.3 Pengertian Android

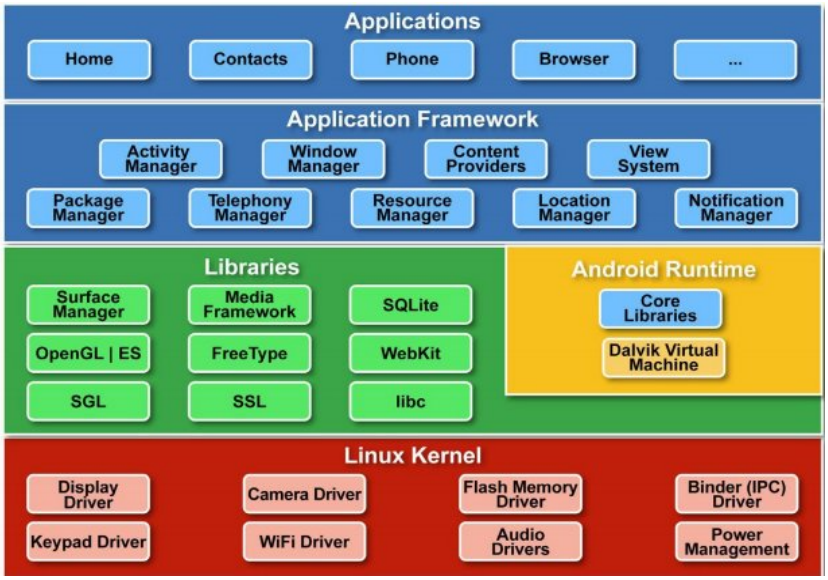
Menurut Nazrudin safaat H (2012:1) menyatakan bahwa android adalah sebuah sistem oprasi untuk perangkat *mobile* berbasis linux yang mencangkup sistem oprasi, *middleware* dan aplikasi. Android menyiapkan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka.

Menurut Nazrudin safaat H (2012:3) menyatakan bahwa Android dipuji sebagai "*platform mobile* pertama yang lengkap, terbuka, dan bebas".

1. Lengkap (C*omplete Platform*) :para desainer dapat melakukan pendekatan komprehensif ketika mereka sedang mengembangkan *platform* Android. Android merupakan sistem operasi yang aman dan menyediakan *tools* dalam membangun *software* dan memungkinkan untuk peluang pengembangan aplikasi.
2. Terbuka (*Open Source Platform*) :*Platform* Android disediakan melalui lisensi *open source*. Pengembang dapat dengan bebas untuk mengembangkan aplikasi. Android sendiri menggunakan Linux Kernel 2.6.
3. Free (*Free Platform*) : Android adalah *platform/*aplikasi yang bebas untuk develop. Tidak ada lisensi atau biaya royalty untuk dikembangkan pada *platform*

android. Tidak ada kontrak yang diperlukan. Aplikasi untuk android dapat di distribusikan dan diperdagangkan dalam bentuk apa pun.

Android mempunyai Arsitektur sebagai berikut :



Gambar 2.2 Arsitektur Android

1. *Application*: Android telah menyediakan beberapa aplikasi inti seperti email, sms, kalender, peta, *browser*, kontak, dan lainnya. Semua aplikasi tersebut ditulis dengan menggunakan bahasa pemograman Java.
2. *Application Framework*: Pengembang aplikasi mendapatkan akses penuh ke seluruh *application framework*. Arsitektur aplikasi didesain agar komponen-komponennya dapat dengan mudah digunakan kembali. Suatu aplikasi dapat mengumumkan segala macam kebutuhannya lalu aplikasi lain akan menyiapkan kebutuhan itu.
3. *Libraries*: Pada Android terdapat *library* dari C/C++ yang digunakan oleh berbagai macam komponen pada sistem Andorid. Pengembang aplikasi dapat

menggunakannya dengan melalui *Application Framwork*. 4) Android *Runtime*: Android menyediakan sejumlah *library* inti yang sebagian besar ada pada *library* inti bahasa pemograman Java.

4. Linux *Kernel*: Android menggunakan Linux 2.6 (sampai versi Gingerbread) sebagai sistem layanan inti seperti keamanan, manajemen memori, manajemen proses, *stack* jaringan, dan model *driver*. Kernel bertindak sebagai layar abstraksi antara perangkat keras dan perangkat lunak.

2.3.1 Sejarah Android

Android merupakan sistem operasi yang dikembangkan untuk perangkat lunak mobile berbasis linux. Pada awalnya sisitem operasi ini dikembangkan oleh Android Inc. yang dikemudian dibeli oleh Google pada tahun 2005.

Dalam suatu usaha mengembangkan android, pada tahun 2007 dibentuklah Open Handset Alliance (OHA), sebuah konsorsium dari beberapa perusahaan, yaitu Texas Instruments, Broadcom Corporation, Google, HTC ,Intel ,LG , Marvel Tecnologi Group, Motorola, NVidia, Qualcomm, Samsusng Electronic , Sprint Nextel, dan T-Mobile dengan tujuan untuk mengembangkan statndar terbuka perangkat mobile. Pada tanggal 9 desember 2008 ia diumumkan bahwa 14 anggota baru akan bergabung pada proyek android .

2.3.2 Versi Android

Android memiliki berbagai versi , penamaannya pun unik yaitu menggambil dari nama makanan dan berurutan sesuai abjad . Dapat kita liat di bawah ini :

Tabel 2.1 Versi Android

Versi Android	Tanggal Rilis	Kode Nama
1.1	9 februari 2009	Astro- Bender
1.5	30 April 2009	Cupcake
1.6	15 September 2009	Donut
2.0/2.1	26 Oktober 2009	Éclair
2.2	20 mei 2010	Froyo
2.3	6 Desember 2010	Gingerbred
3.0/3.1/3.2	22 Februari 2011	Honeycomb
4.0	19 Oktober 2011	Ice cream sandwich
4.1-4.2.2		Jelly Bean
		KitKat

2.3.3 Android *Software Development Kit* (SDK)

Android SDK adalah tools API (*Application Programing Interfaces*) Yang diperlukan untuk mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java.

2.3.4 Android *Development Tools* (ADT)

Android ADT adalah plugin yang didesain untuk *Integrated Development Environment (*IDE) Eclipse yang memberikan kemudahan dalam mengembangkan aplikasi android.

2.4 Pengertian PHP (Hypertext Processor)

Menurut Mochamad Joko Adi Wiraman (2009:1) menyatakan bahwa PHP merupakan bahasa pemograman yang paling popular dan banyak digunakan untuk pemograman web. PHP disebut juga permograman *server side*, artinya program yang dijalankan pada server. PPH sebenarnya merupakan program yang berjalan di LINUX.Program ini bersifat freewere.

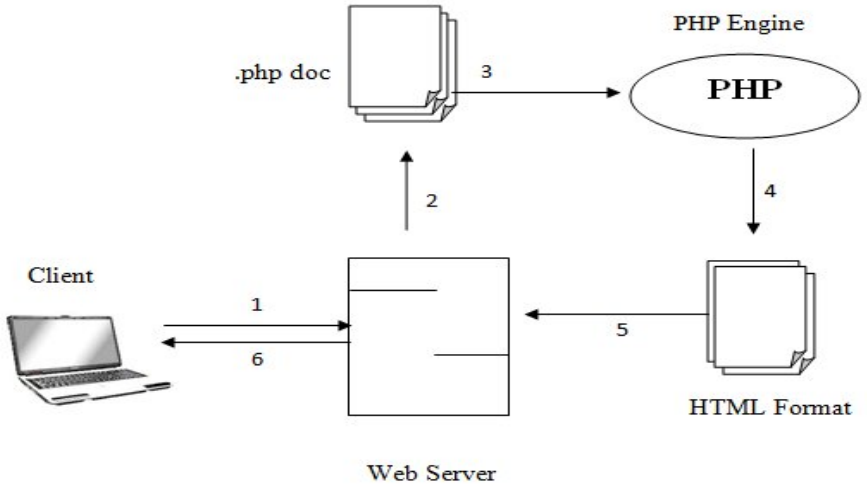
Dalam pembangunan suatu website, PHP memiliki aturan penulisan, yaitu bagaimana memulai PHP dan mengakhiri PHP. Untuk memulai program PHP, harus menggunakan tanda **<?php,** setelah tanda tersebut baru dilanjutkan penulisan kode program isi di dalamnya dengan menggunakan tag untuk menuliskan kode PHPnya. Sedangkan untuk mengakhiri penulisan PHP dapat menutupnya menggunakan tanda **?>,** atau tanda lain seperti **</script>**jika tanda yang digunakan untuk mengawali PHPnya adalah **<script language=”php”>**, atau **%>**jika tanda yang mengawaliPHPnya adalah **<%**.

Menurut Sutarman (2007:95) menyatakan bahwa PHP adalah salah satu bahasa *Server-side* yang di desain khusus untuk aplikasi web. PHP dapat disisipkan diantara bahasa HTML dan karena bahasa *Server side*, maka bahasa PHP akan dieksekusi di server, sehingga yang dikirimkan ke browser adalah ”hasil jadi” dakam bentuk HTML, dan kode PHP anda tidak akan terlihat.

Contoh script sederhana PHP yang disisipkan ke dalam HTML (disimpan dengan ekstensi .php) :

```
<html>
<head>
<title>embeded dalam HTML</title>
<body>
    <?php
    Echo ” Selamat belajar PHP ”;
    ?>
</body>
</html>
```

Prinsipkerjapada PHP sama seperti halnya pada HTML. Hanya saja, ketika berkas PHP yang diminta didapatkan oleh *web server,* isinya segera dikirimkan ke mesin PHP dan mesin ini lah yang memproses dan memberikan hasilnya (berupa kode HTML) ke*web server.*Selanjutnya, *web server* menyampaikan ke klien.



Gambar 2.3 Skema Proses PHP

Sumber : Wirawan, Mochamad J.A. *Amazing News Website with PHP, AJAX, dan MySQL* (2009)

PHP termasuk ke dalam*Open Source Product*. Jadi pengguna dapat merubah *source code* dan mendistribusikannya secara bebas.PHP juga diedarkan secara gratis, maka semua orang bisa memperolehnya secara gratis.

2.4.1 DataBase MySQL

Menurut Syahrul Mubarak Abdullah (2010:223) MySQL (My Structure Query Language) adalah sebuah program pembuat database yang bersifat open source, artinya siapa saja boleh menggunakannya dan tidak dicekal. MySQL sebenarnya produk yang berjalan pada platform Linux. Karena sifatnya yang open source, dia 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 pemrograman web. Kelebihan lain dari MySQL adalah ia menggunakan bahasa Query standar yang dimiliki SQL (Structure Query Language). SQL adalah suatu bahasa permintaan yang terstruktur yang telah distandarkan untuk semua program pengakses database. Sebagai sebuah program penghasil database, MySQL tidak dapat berjalan sendiri tanpa adanya sebuah aplikasi lain. MySQL dapat didukung oleh hampir semua program aplikasi baik yang open source maupun yang tidak. Berikut ini beberapa keistimewaan MySQL, antara lain :

1. Portability

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

2. Multi User

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

3. Security

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

4. 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.

2.5 Java PlatForm Android

Merurut Wahana Komputer (2012:3) menyatakan bahwa Pada bahasa pemograman, proses kompilasi program harus dilakukan di mesin computer tempat kode program itu ditulis sehingga hanya dapat dijalankan pada arsitektur computer tersebut. Permasalahan terjadi ketika program tersebut harus berjalan pada arsitektur komputer yang tidak sama dengan lingkungan program tersebut dibangun maka program tersebut tidak akan dapat dijalankan. Pada java, program yang telah dibangun dapat dijalankan pada berbagai *platform* dan sistem operasi. Java memiliki dua buah *platform* penting untuk membangun perangkat lunak.

Beberapa produk yang dikeluarkan untuk membantu dalam membuat aplikasi, yaitu:

1. *Java Virtual Machine*

   Java Virtual Machine (JVM) adalah penerjemah kode program yang telah dikompilasi menjadi kode mesin. JVM dapat diterapkan pada berbagai sistem operasi dan berbagai *platform*. Hal ini membuat program java dapat dengan

mudah dijalankan pada computer bersistem oprasi apa pun tanpa harus mengubah kode program yang telah ditulis.

2. *Java Runtime Environment*

   Java Runtime Environment (JRE) merupakan lingkungan yang mendukung proses eksekusi program ketika JVM menerjemahkan program itu ke dalam bahasa mesin. Pada JRE juga telah terdapat kelas inti untuk java API dan berbagai file pendukungnya.

3. Java Development Kit (JDK)

   Java Development Kit (JDK) merupakan perangkat lunak yang digunakan untuk menajemen dan membangun berbagai aplikasi java.JDK merupakan superset dari JRE, berisikan segala sesuatu yang ada di JRE ditambahkan *compiler* dan *debugger* yang diperlukan untuk mengembangkan *applet* dan aplikasi.JDK yang mendukung aplikasi android adalah JDK1.6.0_24 (JDK 1.6 *update* 24).

2.6 Konsep Perancangan Berorientasi Objek

2.6.1 Pengenalan UML

Menurut Prabowo Pudjo Widodo & Herlawati (2011) UML (Unified Modeling Language) merupakan pengganti dari metode analisis berorientasi object dan design berorientasi object (OOA & OOD) yang dimunculkan sekitar akhir tahun 80-an dan awal tahun 90-an. UML merupakan gabungan dari metode Grady Booch (Booch Method), James Rumbaugh (OMT) dan Ivar Jacobson (OOSE). Tetapi UML ini akan mencakup lebih luas daripada OOA&D. Pada pertengahan pengembangan UML dilakukan standarisasi proses dengan OMG (*Object Management Group*)

dengan harapan UML akan menjadi bahasa standar pemodelan pada masa yang akan datang.

UML disebut sebagai bahasa pemodelan bukan metode. Kebanyakan metode terdiri paling sedikit prinsip, bahasa pemodelan dan proses. Bahasa pemodelan (sebagian besar grafik) merupakan notasi dari metode yang digunakan untuk mendesain secara cepat.

Bahasa pemodelan merupakan bagian terpenting dari metode. Ini merupakan bagian kunci tertentu untuk komunikasi. Jika anda ingin berdiskusi tentang desain dengan seseorang, maka Anda hanya membutuhkan bahasa pemodelan bukan proses yang digunakan untuk mendapatkan desain. UML merupakan bahasa standar untuk penulisan Blueprint Software yang digunakan untuk Visualisasi (*Visualize*), Spesifikasi (*Specify*), Pembentukan (*Construct*) dan Pendokumentasian (*Documentation*) alat-alat dari sistem perangkat lunak.

2.6.2 Sejarah UML

UML dimulai secara resmi pada oktober 1994, ketika Rumbaugh bergabung dengan Booch pada Relational Software Corporation. Proyek ini memfokuskan pada penyatuan metode Booch dan OMT. UML versi 0.8 merupakan metode penyatuan yang dirilis pada bulan Oktober 1995. Dalam waktu yang sama, Jacobson bergabung dengan Relational dan cakupan dari UML semakin luas sampai diluar perusahaan OOSE. Dokumentasi UML versi 0.9 akhirnya dirilis pada bulan Juni 1996. Meskipun pada tahun 1996 ini melihat dan menerima feedback dari komunitas Software Engineering . Dalam waktu tersebut, menjadi lebih jelas bahwa beberapa organisasi

perangkat lunak melihat UML sebagai strategi dari bisnisnya. Kemudian dibangunlah UML Consortium dengan beberapa organisasi yang akan menyumbangkan sumber dayanya untuk bekerja, mengembangkan, dan melengkapi UML.

Di sini beberapa partner yang berkontribusi pada UML 1.0, diantaranya Digital Equipment Corporation, Hewlett-Packard, I-Logix, Intellicorp, IBM, ICON Computing, MCI Systemhouse, Microsoft, Oracle, Relational, Texas Instruments dan Unisys. Dari kolaborasi ini dihasilkan UML 1.0 yang merupakan bahasa pemodelan yang ditetapkan secara baik, expressive, kuat, dan cocok untuk lingkungan masalah yang luas. UML 1.0 ditawarkan menjadi standarisasi dari Object Management Group (OMG). Dan pada Januari 1997 dijadikan sebagai standar bahasa pemodelan.

Antara Januari–Juli 1997 gabungan group tersebut memperluas kontribusinya sebagai hasil respon dari OMG dengan memasukkan Adersen Consulting, Ericsson, ObjectTimeLimeted, Platinum Technology, Ptech, Reich Technologies, Softeam, Sterling Software dan Taskon. Revisi dari versi UML (versi 1.1) ditawarkan kepada OMG sebagai standarisasi pada bulan Juli 1997.

Dan pada bulan September 1997, versi ini dierima oleh OMG Analysis dan Design Task Force (ADTF) dan OMG ArchitectureBoard. Dan Akhirnya pada Juli 1997 UML versi 1.1 menjadi standarisasi.

Pemeliharaan UML terus dipegang oleh OMG Revision Task Force (RTF) yang dipimpin oleh Cris Kobryn. RTP merilis editorial dari UML 1.2 pada Juni 1998. Dan pada tahun 1998 RTF juga merilis UML 1.3 disertai dengan user guide dan memberikan technical cleanup.

2.6.3 Pengertian UML

Menurut Prabowo Pudjo Widodo & Herlawati (2011) UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan artifacts (bagian dari informasi yang digunakan atau dihasilkan oleh proses pembuatan perangkat lunak, artifact tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya. Selain itu UML adalah bahasa pemodelan yang menggunakan konsep orientasi object.

UML dibuat oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson di bawah bendera Rational Software Corp. UML menyediakan notasi-notasi yang membantu memodelkan sistem dari berbagai perspektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

2.6.4 Diagram UML

Untuk merancang model sebuah sistem. Ada beberapa diagram yang disediakan dalam UML antara lain:

2.6.4.1 Diagram Use Case

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. Konsep dasar pemodelan *use case* meliputi : *use case,* actor, relasi,

diagram aktivitas dan diagram *use case.* Aktor merupakan bagian dari *use case* yang bertindak sebagai subjek (pelaku) dalam suatu proses. *Use case* adalah proses-proses yang terjadi dalam suatu *software*. *Use case* juga menggambarkan apa yang sedang dilakukan oleh seorang *actor*. Relasi menggambarkan hubungan antara *actor* dan *use case.*

2.6.4.2 Diagram Aktifitas

Diagram aktivitas atau *Activity diagram* menggambarkan aliran fungsionalitas sistem. Pada tahap pemodelan bisnis, diagram aktivitas dapat digunakan untuk menunjukkan aliran kerja bisnis. Diagram ini dapat digunakan untuk menggammbarkan aliran kerja (*floe of events*) dalam *use case.*

2.6.4.3 Diagram Sekuensial

Diagram sekuensial atau *sequence diagram* digunakan untuk menggambarkan aliran fungsionalitas dalam use case. Diagram ini disusun berdasarkan urutan waktu.

2.6.4.4 Diagram Kelas

Diagram kelas atau *class diagram* menunjukkan interaksi antar kelas dalam sistem. Diagram kelas digunakan untuk menampilkan kelas-kelas atau paket-paket di dalam sistem dan relasi antar mereka. Diagram ini menunjukkan gambaran sistem secara statis. Satu diagram kelas menampilkan *subset* dari kelaskelas dan relasinya.

2.6.4.5 Diagram Deployment

Diagram deployment bagian – bagian perangkat lunak suatu sistem ke perangkat keras yang akan mengeksekusinya. Elemen – elemen perangkat lunak seperti komponen, kelas, paket dan sebagainya dimanifestasikan menggunakan artifak serta dipetakan ke perangkat keras yang akan menjalankannya dengan nodes.

2.7 Eclipse

Menurut Nasruddin Safaat H (Pemrograman aplikasi *mobile smartphone* dan tablet PC berbasis android)(2012:16) **Eclipse** adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platformindependent*). 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 lain 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, pengujian perangkat lunak, pengembangan web, dan lain sebagainya.

Pada saat ini, Eclipse merupakan salah satu IDE favorit karena gratis dan *open source*. *Open source* 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 membuat komponen yang disebut *plugin*.

Konsep dari Eclipse sendiri ada 3 yaitu IDE adalah

1. terbuka (*open*),
2. mudah diperluas (*extensible*) untuk apa saja, dan
3. tidak untuk sesuatu yang spesifik.