# BAB II

# LANDASAN TEORI

## 2.1 Aplikasi

Merupakan perangkat lunak yang dijalankan oleh para pengguna untuk mendapat suatu tujuan tertentu.

### 2.1.1 Pengerian Aplikasi

Pengertian Aplikasi menurut Barry Pratama dalam artikel di *http://ilmukomputer.org/category/aplikasi-server/*, yaitu:

1. *Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas.*
2. *Aplikasi adalah sistem lengkap yang mengerjakan tugas spesifik*.
3. *Aplikasi basis data terdiri atas sekumpulan menu, formulir, laporan dan program yang memenuhi kebutuhan suatu fungsional unit bisnis/organisasi/instansi*.

Aplikasi merupakan program siap pakai. Program yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain. Contoh-contoh aplikasi ialah program pemproses kata dan Web Browser. Aplikasi akan menggunakan sistem operasi (OS) komputer dan aplikasi yang lainnya yang mendukung.Aplikasi secara umum adalah suatuproses dari cara manual yang

ditransformasikan ke komputer dengan membuat sistem atauprogram agar data diolah lebih berdaya guna secara optimal.

Jadi dapat disimpulkan bahwa aplikasi merupakan subkelas perangkat lunak komputeryang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna agar lebih berguna secara optimal.

### 2.1.2 Karakteristik Aplikasi

Salah satu karakteristik yang menonjol dari mobile user ini selalu dilengkapi oleh perangkat portabel yang memungkinkan bisa terkoneksi dengan internet atau bisa terhubung dengan pengguna lainnya. Perangkat-perangkat tersebut pada umumnya berbentuk kecil bisa berupa notebook/netbook, smartphone, PDA, tablet PC, atau mobile devices yang lain. Mereka menggunakan perangkat tersebut untuk melakukan aktivitas bisnis, belajar-mengajar, interaksi sosial ataupun sekedar untuk hiburan.Selain oleh perangkat portabel, untuk melakukan berbagai macam aktifitasnya tentu pengguna mobile ini menggunakan aplikasi yang terinstall di perangkat tersebut.

### 2.1.3 Komponen Aplikasi

Fungsionalitas aplikasi dapat dikustomisasi secara mudah dan diperkaya dengan arsitektur komponennya yang fleksibel. Aplikasi mengatur satu set komponen aplikasi, masing-masing mengimplementasi fitur tertentu.Dengan mengkonfigurasi properti komponen aplikasi, kita bisa mengkustomisasi kelasi dan nilai properti setiap komponen aplikasi yang dipakai dalam sebuah aplikasi.

## 2.2 Aplikasi *Mobile*

Pemanfaatan aplikasi *mobile* untuk hiburan paling banyak digemari oleh hampir 70% pengguna telepon seluler, karena dengan memanfaatkan adanya fitur game, music player, sampai video player membuat kita menjadi semakin mudah menikmati hiburan kapan saja dan dimanapun.

### 2.2.1 Pengertian Aplikasi *Mobile*

Aplikasi *Mobile* merupakan sebuah aplikasi yang memungkinkan untuk melakukan mobilitas dengan menggunakan perlengkapan seperti PDA, telepon seluler atau *Handphone*. Dengan menggunakan aplikasi *mobile*, User dapat dengan mudah melakukan berbagai macam aktifitas mulai dari hiburan, berjualan, belajar, mengerjakan pekerjaan kantor, *browsing* dan lain sebagainya.

### 2.2.2 Karakteristik Perangkat *Mobile*

Karakteristik yang menonjol dari perangkat mobile, ukuran yang kecil, memory yang terbatas, daya proses yang terbatas, mengkonsumsi daya yang rendah, kuat dan dapat diandalkan, konektivitas yang terbatas, dan masa hidup yang pendek.

## 2.3 *HandPhone*

Telepon selular atau *handphone* merupakan alat telekomunikasi elektronik yang mempunyai kemampuan dasar yang sama dengan telepon konvensional saluran tetap, namun dapat dibawa ke mana-mana (*portabel, mobile*) dan tidak perlu disambungkan dengan jaringan telepon menggunakan kabel (nirkabel; wireless). Selain berfungsi untuk melakukan dan menerima panggilan telepon, ponsel umumnya juga mempunyai fungsi pengiriman dan penerimaan pesan singkat *Short Message*

*Service* (SMS). Mengikuti perkembangan teknologi, kini ponsel dilengkapi dengan berbagai pilihan fitur, seperti bisa menangkap siaran televisi, perangkat lunak pemutar audio dan video, kamera digital, game, dan layanan internet. Ada pula penyedia jasa telepon genggam di beberapa negara yang menyediakan layanan generasi ketiga atau 3G dengan menambahkan jasa videophone, sebagai alat pembayaran, maupun untuk televisi online pada telepon genggam mereka. Sekarang, telepon genggam menjadi gadget yang multifungsi. Selain fitur-fitur tersebut, ponsel sekarang sudah ditanamkan fitur komputer. Jadi pada ponsel tersebut, orang bisa mengubah fungsi ponsel tersebut menjadi mini komputer. Di dunia pendidikan, fitur ini sangat membantu bagi para pelajar untuk melakukan pembelajaran di satu alat.

## 2.4 Java

Menurut definisi Sun Microsystem, di dalam buku M. Shalahuddin dan Rosa A.S. (2010 : 1) Java adalah nama sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer yang berdiri sendiri (standalone) ataupun pada lingkungan jaringan.

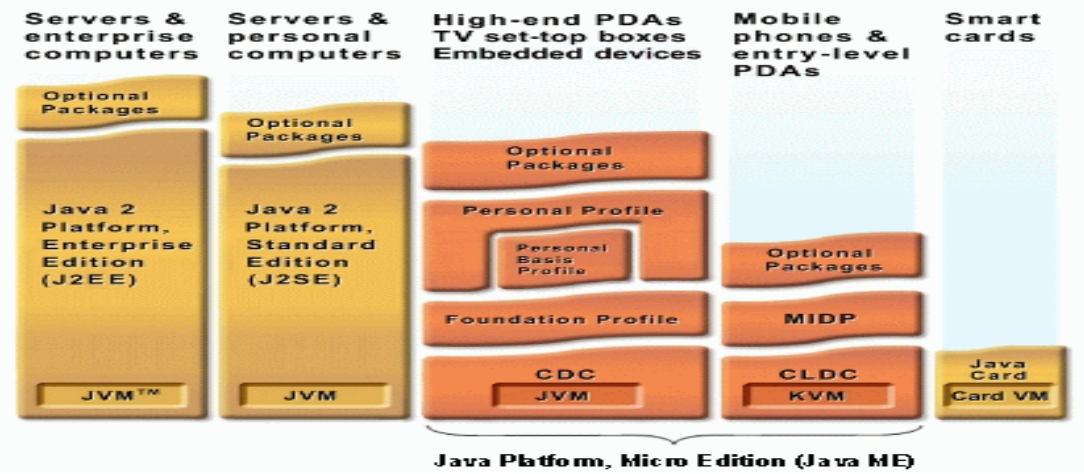
### 2.4.1 Pengertian Java

Java2 adalah generasi kedua dari Java *platform* (generasi awalnya adalah *Java Development Kit*). Java berdiri diatas sebuah mesin *interpreter* yang di beri nama*Java Virtual Machine* (JVM). JVM inilah yang akan membaca *bytecode* dalam file **.class** dari suatu program sebagai representasi langsung program yang berisi bahasa mesin. Oleh karena itu, bahasa Java disebut sebagai bahasa pemrogramaan

yang *portable* karena dapat dijalankan pada berbagai sistem operasi, asalkan sistem operasi tersebut terdapat JVM.

*Platform* Java terdiri dari kumpulan *library*, JVM, kelas-kelas *loader* yang dipaket dalam sebuah lingkungan rutin Java, dan sebuah *compiler*, *debuger*, dan perkakas lain yang dipaketkan dalam *Java Development Kit* (JDK). Java2 adalah generasi sekarang yang sedang berkembang dari *platform* Java. Agar sebuah program Java dapat dijalankan, maka file dengan eksistensi .**java**harus dikompilasi menjadi file *bytecode*. Untuk menjalankan *bytecode* tersebut dibutuhkan *Java Runtime Environment* (JRE) yang memungkinkan pemakai untuk menjalankan program Java, hanya menjalankan, tidak untuk membuat kode baru lagi. JRE berisi JVM dan *library* Java yang digunakan.

*Platform* Java memiliki tiga buah edisi yang berbeda, yaitu J2EE (Java2 *Enterprise Edition*), J2ME (Java2 *Micro Edition*), dan J2SE (Java2 *Second Edition*). J2EE adalah kelompok dari beberapa API dari Java dan teknologi selain Java.J2EE dibuat untuk membuat aplikasi yang kompleks.J2EE sering dianggap sebagai *middleware* atau teknologi yang berjalan di *server*, namun sebenarnya J2EE tidak hanya terbatas untuk itu.J2EE juga mencakup teknologi yang dapat digunakan disemua lapisan dari sebuah sistem informasi.Implementasi J2EE menyediakan kelas dasar dan API dari Java yang mendukung pengembangan dan rutin standar untuk aplikasi *client* maupun *server*, termasuk aplikasi yang berjalan di *web browser*.J2SE adalah lingkungan dasar dari Java.Ruang lingkup keterhubungan J2EE, J2SE, dan J2ME.




**Gambar 2.1 Keterhubungan J2SE, J2EE dan J2ME**
**Sumber : Jurnal Aplikasi Mobile Learning Organizer Mahasiswa**

### 2.4.2 J2ME

Menurut Candra (2010:32), J2ME adalah platform-independen dan membuat penciptaan mobile, aplikasi berbasis internet. Menurut sarwosri (2010:184), Java ME adalah lingkungan pengembangan yang didesain untuk menggunakan aplikasi java pada peralatan elektronik kecil, seperti telepon seluler, PDA, dan sejenisnya. Java ME dibuat untuk mengatasi keterbatasanyang berhubungan dengan pembuatan aplikasi pada peralatan elektronik kecil

J2ME merupakan satu set spesifikasi dan teknologi yang fokus kepada perangkat konsumen. Perangkat ini memiliki jumlah memori yang terbatas, menghabiskan sedikit daya dari baterei, layar yang kecil dan bandwith jaringan yang rendah.Karateristik dari perangkat mobile, dan bagaimana hal tersebut mempengaruhi

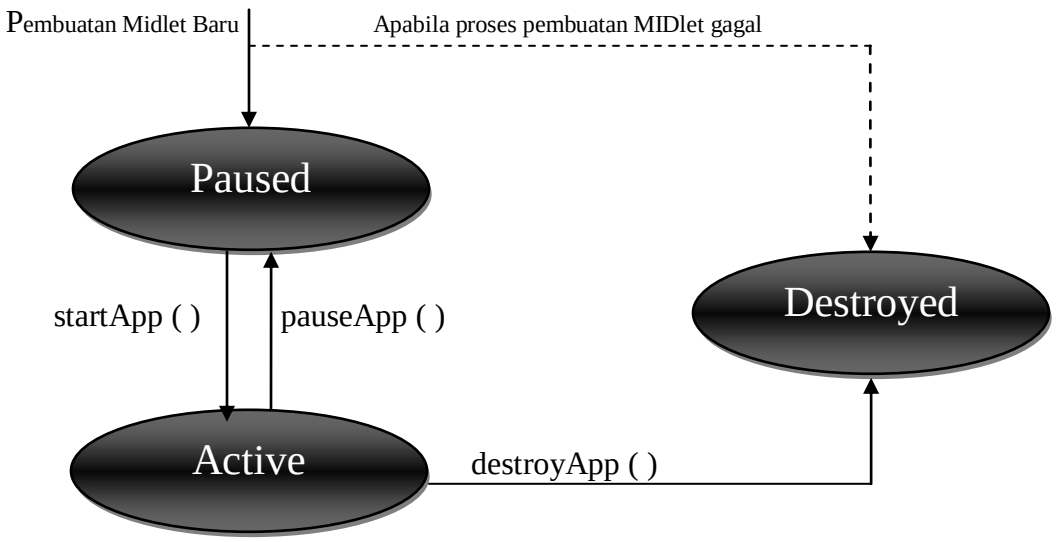
pengembangan program untuk perangkat ini. Kita akan diperkenalkan kepada Java 2 Mobile Edition (J2ME), termasuk pentingnya configuration dan profilenya.

### 2.4.3 MIDlet

MIDlet merupakan bagian dari package javax.microedition.midlet. Sebuah MIDlet harus di-extend dengan class MIDlet. Dan dapat meminta parameter dari AMS seperti dirumuskan dalam application descriptor (JAD).Suatu MIDlet tidak harus memiliki (dan memang harus tidak mempunyai) sebuah method public static void main(String[] argv).Method tersebut tidak akan dikenal lagi oleh AMS sebagai titik awal sebuah program.

Aplikasi-aplikasi MIDlet dibungkus dan dikirim kedalam perangkat sebagai MIDlet suites.Sebuah MIDlet suite terdiri dari Java Archive (JAR) dan sebuah tambahan Java Application Descriptor (JAD). File JAD adalah suatu file teks yang berisi satu set atribut-atribut, beberapa dibutuhkan.Terdapat Software yang Digunakan Dalam mobile Programming.

Setiap MIDlet dapat berada dalam salah satu keadaan (state) berikut:**Pause**, **Active**, maupun **Destroyed**




**Gambar 2.2 Siklus Hidup MIDlet**
**Sumber : Tuntunan Pemograman JAVA untuk Handphone**

Pada gambar di atas bahwa pada saat pembuatan MIDlet baru, mula-mula MIDlet akan berada dalam keadaan Paused. Apabila proses pembuatan MIDlet gagal atau mengakibatkan kesalahan (menimbulkan eksepsi). Maka MIDlet akan langsung berada dalam keadaan Destroyed. Namun apabila proses pembuatan MIDlet berjalan dengan baik, maka setelah MIDlet dijalankan, maka AMS secara otomatis akan mengeksekusi method startApp( ) dan hal ini akan mengubah MIDlet untuk dalam keadaaan Active. MIDlet yang dalam keadaan active dapat diubah kembali menjadi keadaan Pause melalui pemanggilan methode pauseApp( ) atau diubah menjadi keadaan Destroyed melalui pemanggilan methode destroyedApp( ).

#### 2.4.3.1 Method startApp()

AMS akan memanggil method startApp() untuk memerintahkan MIDlet agar memperoleh fokus dan menjadikan MIDlet berada dalam keadaan Active. Hal ini terjadi ketika MIDlet baru saja dibuat atau MIDlet yang akan kembali diaktifkan dari keadaan paused. Bentuk umum deklarasi method startApp() adalah sebagai berikut:

**public static void startApp()**

#### 2.4.3.2 Method pauseApp()

AMS akan memanggil method pauseApp() untuk memerintahkan MIDlet agar tidak memiliki fokus dan akan menjadikan MIDlet berada dalam keadaan paused. Dalam keadaan ini, aplikasi tidak dapat memiliki satu pun tampilan UI (User Interface). Apabila aplikasi yang yang dibuat mengandung objek *Thread*maupun *Timer,* maka objek-objek tersebut tidak akan dihentikan secara otomatis. Artinya, harus dihentikan secara manual melalui penulisan kode. Aplikasi akan kembali berada dalam keadaan Active bila diaktivasi ulang. Bentuk umum deklarasi method pauseApp() adalah sebagai berikut:

**public static void pauseApp()**

#### 2.4.3.3 Method destroyApp()

AMS memanggil method destroyApp() untuk memerintahkan MIDlet agar membuang atau membebaskan semua resource (biasanya berupa file) yang digunakan sekaligus menutup dan menghentikan aplikasi sesegera mungkin. Ini berarti harus menutup semua stream yang masih terbuka serta menghentikan semua thread dan timer yang digunakan. Pemanggilan method destroyApp() akan mengakibatkan

MIDlet berada dalam keadaan Destroyed sehingga pada saat tersebut tidak dapat lagi melakukan pengaksesan terhadap obek Display. Bentuk umum deklarasi methd destroyApp() adalah sebagai berikut:

**public static void destroyApp(boolean unconditional)**

### 2.4.4 Connected Limited Device Configuration(CLDC)

Didalam buku Tuntunan Pemograman java unuk Handphone CLDC adalah sebuah konfigurasi yang terdapat didalam J2ME untuk alat-alat yang memiliki keterbatasan ruang memori atau RAM (kurang dari 512KB) dan pada umumnya dioperasikan dengan menggunakan baterai, serta memiliki bandwith yang kecil

### 2.4.5 KVM (Kilobyte *Virtual Machine*)

KVM adalah paket JVM yang didesain untuk perangkat yang kecil.KVM mendukung sebagian dari fitur-fitur JVM, seperti misalnya KVM tidak mendukung operasi *floating-point* dan finalisasi objek.KVM diimplemen-tasikan dengan menggunakan C sehingga sangat mudah beradaptasi pada tipe *platform* yang berbeda.

### 2.4.6 CVM (C-*Virtual Machine*)

CVM adalah paket JVM optimal yang digunakan pada CDC.CVM mempunyai seluruh fitur dari *virtual machine* yang didesain untuk perangkat yang memerlukan fitur-fitur *Java2 virtual machine.*

## 2.5 Panca Indera

Panca indera adalah alat tubuh yang berguna untuk mengetahui keadaan di luar tubuh. Alat indera ada lima, yaitu mata, telinga, hidung, lidah, dan kulit. Kelima alat indra itu disebut panca indera.

### 2.5.1 Pengertian Panca Indera

Pada setiap alat indera terdapat saraf. Saraf ini akan menerima rangsang dari luar tubuh. Kemudian, saraf mengirim rangsang itu ke otak. Saat rangsang diterima otak dengan baik, maka kita dapat melihat, mendengar, membau, mengecap, atau meraba. Alat indera harus dirawat dengan baik. Jika alat indera rusak, tubuh kita tidak dapat bekerja dengan baik. Akibatnya kita tidak dapat menikmati keadaaan sekitar.

### 2.5.2 Mata

Mata terdiri dari otot mata, bola mata dan saraf mata serta alat tambahan mata yaitu alis, kelopak mata, dan bulu mata. Alat tambahan mata ini berfungsi melindungi mata dari gangguan lingkungan. Alis mata berfungsi untuk melindungi mata dari keringat, kelopak mata melindungi mata dari benturan dan bulu mata melindungi mata dari cahaya yang kuat, debu dan kotoran.

Fungsi bagian - bagian indra penglihatan adalah sebagai berikut :

a. Kornea mata berfungsi untuk menerima rangsang cahaya dan meneruskannya ke bagian mata yang lebih dalam.

b. Lensa mata berfungsi meneruskan dan memfokuskan cahaya agar bayangan benda jatuh ke lensa mata.

c. Iris berfungsi mengatur banyak sedikitnya cahaya yang masuk ke mata

d. Pupil berfungsi sebagai saluran masuknya cahaya.

e. Retina berfungsi untuk membentuk bayangan benda yang kemudian dikirim oleh oleh saraf mata ke otak

f. Otot mata berfungsi mengatur gerakan bola mata

g. Saraf mata berfungsi meneruskan rangsang cahaya dari retina ke otak

### 2.5.3 Telinga

Indra pendengar adalah telinga yang terdiri dari :

1). Telinga bagian luar yaitu daun telinga, lubang telinga dan liang pendengaran

2). Telinga bagian tengah terdiri dari gendang telinga, 3 tulang pendengar ( martil, landasan dan sanggurdi) dan saluran eustachius.

3). Telinga bagian dalam terdiri dari alat keseimbangan tubuh, tiga saluran setengah lingkaran, tingkap jorong, tingkap bundar dan rumah siput (koklea)

Fungsi bagian-bagian indra pendengar :

a. Daun telinga, lubang telinga dan liang pendengaran berfungsi menangkap dan mengumpulkan gelombang bunyi.

b. Gendang telinga berfungsi menerima rangsang bunyi dan meneruskannya ke bagian yang lebih dalam.

c. Tiga tulang pendengaran ( tulang martil, landasan dan sanggurdi) berfungsi memperkuat getaran dan meneruskannya ke koklea atau rumah siput.

d. Tingkap jorong, tingkap bundar, tiga saluran setengah lingkaran dan koklea (rumah siput) berfungsi mengubah impuls dan diteruskan ke otak. Tga saluran setengah

lingkaran juga berfungsi menjaga keseimbangan tubuh.

e. Saluran eustachius menghubungkan rongga mulut dengan telinga bagian luar.

### 2.5.4 Lidah

Bagian lidah yang berbintil-bintil disebut papila adalah ujung saraf pengecap. Setiap bintil-bintil saraf pengecap tersebut mempunyai kepekaan terhadap rasa tertentu berdasarkan letaknya pada lidah.Pangkal lidah dapat mengecap rasa pahit, tepi lidah mengecap rasa asin dan asam serta ujung lidah dapat mengecap

### 2.5.5 Hidung

Fungsi bagian-bagian indra pembau/hidung :

a. Lubang hidung berfungsi untuk keluar masuknya udara

b. Rambut hidung berfungsi untuk menyaring udara yang masuk ketika bernapas

c. Selaput lendir berfungsi tempat menempelnya kotoran dan sebagai indra pembau

d. Serabut saraf berfungsi mendeteksi zat kimia yang ada dalam udara pernapasan

e. Saraf pembau berfungsi mengirimkan bau-bauan yang ke otak

### 2.5.6 Kulit

Dengan kulit kita dapat merasakan sentuhan. Bagian indra peraba yang paling peka adalah ujung jari, telapak tangan, telapak kaki, bibir dan alat kemaluan.

Fungsi bagian-bagian kulit :

a. Kulit ari berfungsi mencegah masuknya bibit penyakit dan mencegah penguapan air dari dalam tubuh.

b. Kelenjar keringat berfungsi menghasilkan keringat

c. Lapisan lemak berfungsi menghangatkan tubuh

d. Otot penggerah rambut berfungsi mengatur gerakan rambut

e. Pembuluh darah berfungsi mengalirkan darah keseluruh tubuh.

## 2.6 Pendidikan Anak Usia Dini

Pendidikan anak usia dini (PAUD) adalah jenjang pendidikan sebelum jenjang pendidikan dasar yang merupakan suatu upaya pembinaan yang ditujukan bagi anak sejak lahir sampai dengan usia enam tahun yang dilakukan melalui pemberian rangsangan pendidikan untuk membantu pertumbuhan dan perkembangan jasmani dan rohani agar anak memiliki kesiapan dalam memasuki pendidikan lebih lanjut, yang diselenggarakan pada jalur formal, nonformal, dan informal. Pendidikan anak usia dini merupakan salah satu bentuk penyelenggaraan pendidikan yang menitikberatkan pada peletakan dasar ke arah pertumbuhan dan perkembangan fisik (koordinasi motorik halus dan kasar), kecerdasan (daya pikir, daya cipta, kecerdasan emosi, kecerdasan spiritual), sosio emosional (sikap dan perilaku serta agama) bahasa dan komunikasi, sesuai dengan keunikan dan tahap-tahap perkembangan yang dilalui oleh anak usia dini. Ada dua tujuan diselenggarakannya pendidikan anak usia dini yaitu:

Tujuan utama: untuk membentuk anak Indonesia yang berkualitas, yaitu anak yang tumbuh dan berkembang sesuai dengan tingkat perkembangannya sehingga memiliki kesiapan yang optimal di dalam memasuki pendidikan dasar serta mengarungi kehidupan di masa dewasa.

Tujuan penyerta: untuk membantu menyiapkan anak mencapai kesiapan belajar (akademik) di sekolah.

Rentangan anak usia dini menurut Pasal 28 UU Sisdiknas No.20/2003 ayat 1 adalah 0-6 tahun. Sementara menurut kajian rumpun keilmuan PAUD dan penyelenggaraannya di beberapa negara, PAUD dilaksanakan sejak usia 0-8 tahun.

## 2.7 UML

UML singkatan dari *Unified Modeling Language* yang berarti bahasa pemodelan standar. (chonoles, 2003: bab 1) mengatakan sebagai bahasa, berarti UML memiliki sintaks dan semantik. Intinya, UML merupakan alat komunikasi yang konsisten dalam mensuport para pengembang sistem saat ini. Sebagai perancang sistem, mau tidak mau pasti akan menjumpai UML, baik kita sendiri yang membuat atau sekedar membaca diagram UML buatan orang lain (Pilone, 2005: bab1), dalam buku Menggunakan UML.

Unified Modelling Language (UML) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak.UML menawarkan sebuah standard.untuk merancang model sebuah sistem. Ada beberapa diagram yang disediakan dalam UML antara lain:

### 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 actor dengan sistem.

(Pooley, 2003:15) mengatakan bahwa model use case dapat dijabarkan dalam diagram use case, tetapi yang perlu diingat, diagram tidak identik dengan model karena model lebih luas dari diagram, sumber buku Menggunkan UML.

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.

**2. Diagram Aktivitas**

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 untukmenggammbarkan aliran kerja (*floe of events*) dalam *use case.*

**3. Diagram Sekuensial**

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

**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.

***5. Component Diagram***

Menggambarkan arsitektur fisik dari perangkat keras dan perangkat lunak sistem, menunjukkan hubungan komputer dengan perangkat (*nodes*) satu sama lain

dan jenis hubungannya. Di dalam *nodes*, *executeable component* dan *object*yang dialokasikan untuk memperlihatkan unit perangkat lunak yang dieksekusi oleh *node* tertentu dan ketergantungan komponen.

***6. Deployment diagram***

Menggambarkan struktur fisik kode dari komponent. Komponent dapat berupa *source code*, komponent biner, atau *executable component*. Sebuah komponent berisi informasi tentang logic class atau class yang diimplementasikansehingga membuat pemetaan dari *logical view* ke *component view.*