

BAB II

TINJAUAN PUSTAKA

II.1. Sistem Informasi

II.1.1. Sistem

Suatu sistem pada dasarnya adalah kumpulan dari unsur atau elemen-elemen yang saling memengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan tertentu.

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu:

1. Komponen (*Components*)

Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, bekerja sama membentuk suatu kesatuan.

2. Batas Sistem (*Boudary*)

Ruang lingkup merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan yang tidak dapat dipisah-pisahkan.

3. Lingkungan Luar (*Environtment*)

Segala sesuatu di luar batas sistem yang memengaruhi operasi dari suatu sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

4. Penghubung Sistem (*Interface*)

Media penghubung antara satu subsistem dengan subsistem lainnya untuk membentuk satu kesatuan sehingga sumber-sumber daya mengalir dari subsistem yang satu ke subsistem lainnya.

5. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

6. Keluaran Sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi pengeluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain.

7. Pengolahan Sistem (*Proses*)

Merupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan.

8. Sasaran Sistem (*Objective*)

Setiap sistem pasti mempunyai tujuan ataupun sasaran yang memengaruhi *input* yang dibutuhkan dan *output* yang dihasilkan (Asbon Hendra : 2012 : 157).

II.1.2. Informasi

Informasi adalah data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta, suatu nilai yang bermanfaat. Jadi, pada suatu proses transformasi data menjadi suatu informasi = *input* - proses – *output* (Asbon Hendra : 2012 : 157).

II.1.3. Sistem Informasi

Sistem informasi adalah sekumpulan komponen-komponen yang saling berhubungan dan bekerja sama untuk mengumpulkan, memproses, menyimpan dan mendistribusikan informasi terkait untuk mendukung proses pengambilan keputusan, koordinasi, dan pengendalian (S. Nofan Maulana Rachman : 2012 : 2).

II.2. Sistem Informasi Geografis

Sistem informasi geografis adalah sistem berbasis komputer yang digunakan untuk menyimpan dan memanipulasi informasi-informasi geografis. Sistem informasi geografi diciptakan untuk mengumpulkan, menyimpan dan menganalisis obyek atau fenomena dimana lokasi geografis menjadi karakteristik atau kritik penting untuk analisi. Sistem informasi geografi adalah sistem berbasis komputer yang memiliki kemampuan dalam menangani data berefrensi dalam:

1. Masukkan data
2. Manajemen data
3. Manipulasi
4. Analisis

Pada awalnya data geografi hanya disajikan di atas peta dengan menggunakan simbol, garis, dan warna. Elemen-elemen geometri ini dideskripsikan di dalam legenda-nya misalnya, garis hitam tebal untuk jalan utama, garis hitam tipis untuk jalan sekunder dan jalan-jalan yang berikutnya. Selain itu, berbagai data juga di dapat di-overlay-kan berdasarkan sistem koordinat yang sama. Akibatnya, sebuah peta menjadi media yang efektif baik

sebagai alat presentasi maupun sebagai bank tempat penyimpanan data geografis. Tetapi, media peta masih mengandung kelemahan atau keterbatasan. Informasi-informasi yang tersimpan, diproses dan dipresentasikan dengan suatu cara tertentu, dan biasanya untuk tujuan tertentu pula. Tidak mudah untuk mengubah bentuk presentasi ini, sebuah peta selalu menyediakan gambar atau symbol unsure geografi dengan bentuk yang tetap atau statis meskipun diperlukan untuk kebutuhan yang berbeda. (S. Nofan Maulana Rachman : 2012 : 2)

II.3. Android

Android adalah sistem operasi untuk perangkat selular yang berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. (Nazruddin Saffat : 2014 : 3).

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-Mobile, dan Nvidia. Pada saat perilisan perdana Android, 5 November 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Dilain pihak, Google merilis kode-kode Android di bawah lisensi *Apache*, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler. Di dunia ini terdapat dua

jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau *Google Mail Services* (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai *Open Handset Distribution* (OHD).

II.3.1 Fitur dan Arsitektur Android

Fitur yang tersedia pada Android adalah:

1. Framework Aplikasi : memungkinkan penggunaan dan pemindahan dari komponen yang tersedia.
2. Dalvik Virtual Machine : virtual machine yang dioptimalkan untuk perangkat mobile.
3. Grafik : grafik 2D dan grafik 3D yang didasarkan pada library OpenGL.
4. SQLite : untuk menyimpan data.
5. Mendukung Media : audio, video, dan berbagai format gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF).
6. GSM, Bluetooth, Edge, 3G, WiFi, Camera, Global Positioning System (GPS), compass, dan accelerometer (tergantung hardware).
7. Lingkungan pengembangan yang kaya, termasuk emolator, peralatan debugging, dan plugin untuk Eclipse IDE. (Zara Zulfariana dan Ernastuti : 2012 : 1)

Sistem operasi Android dibangun berdasarkan kernel Linux, dan memiliki arsitektur sebagai berikut:

1. Applications

Lapisan ini adalah lapisan aplikasi, serangkaian aplikasi akan terdapat pada perangkat mobile. Aplikasi inti yang telah terdapat pada Android termasuk kalender, kontak, SMS (Short Message Service), dan lain sebagainya. Aplikasi-aplikasi ini ditulis dengan bahasa pemrograman Java.

2. Application Framework

Pengembang aplikasi memiliki akses penuh ke Android sama dengan aplikasi inti yang telah tersedia. Pengembang dapat mudah mengakses informasi lokasi, mengatur alarm, menambah pemberitahuan ke status bar dan lainnya sebagainya. Arsitektur aplikasi ini dirancang untuk menyederhanakan penggunaan kembali komponen, aplikasi apa pun yang dapat memublikasikan kemampuan dan aplikasi lainnya dapat menggunakan kemampuan mereka sesuai batasan keamanan. Dasar dari aplikasi adalah seperangkat layanan sistem, yaitu berbagai view yang digunakan untuk membangun *user* interface, content provider yang memungkinkan aplikasi berbagi data, ResourceManager menyediakan akses bukan kode seperti grafik, string, dan layout, NotificationManager yang akan membuat aplikasi dapat menampilkan tanda pada status bar dan ActivityManager yang berguna mengatur daur hidup dari aplikasi.

3. Libraries

Satu set libraries dalam bahasa C/C++ yang digunakan oleh berbagai komponen pada sistem multimedia.

4. Android Runtime

Satu set libraries inti yang menyediakan sebagian besar fungsi yang tersedia di libraries inti dari bahasa pemrograman Java. Setiap aplikasi akan berjalan sebagai proses sendiri pada Dalvik Virtual Machine.

5. Linux Kernel

Android bergantung pada Linux versi 2.6 untuk layanan sistem inti seperti keamanan, manajemen memori, manajemen proses, network stack, dan model driver. Kernel juga bertindak sebagai lapisan antara hardware dan seluruh software. (Zara Zulfariana dan Ernastuti : 2012 : 1).

II.3.2 Jenis-Jenis Versi Android

1. Android versi 1.1

Android versi 1.1 di rilis pada 9 Maret 2009 oleh Google. Android versi ini dilengkapi disupport oleh Google Mail Service dengan pembaruan estetis pada aplikasi, jam alarm, voice search (pencarian suara), pengiriman pesan dengan Gmail, dan pemberitahuan email.

2. Android versi 1.5 Cup Cake

Android Cup Cake di rilis pada pertengahan Mei 2009, masih oleh Google Inc. Android ini dilengkapi software development kit dengan berbagai pembaharuan termasuk penambahan beberapa fitur antara lain yakni kemampuan merekam dan menonton video dengan modus kamera, mengunggah video ke

Youtube, upload gambar ke Picasa langsung dari telepon, serta mendapat dukungan Bluetooth A2DP.

3. Android versi 1.6 Donut

Android Donut di rilis pada September 2009 menampilkan proses pencarian yang lebih baik dibandingkan versi-versi sebelumnya. Selain itu Android Duut memiliki fitur-fitur tambahan seperti galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus; kamera, camcorder dan galeri yang dintegrasikan; Text-to-speech engine; kemampuan dial kontak; teknologi text to change speech. Android Donut juga dilengkapi baterai indikator, dan kontrol applet VPN.

4. Android versi 2.0/2.1 Eclair

Android Eclair dirilis pada 3 Desember 2009. Perubahan yang ada antara lain 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. Android Eclair merupakan Adroid pertama yang mulai dipakai oleh banyak smartphone, fitur utama Eclair yaitu perubahan total struktur dan tampilan *user interface*.

5. Android versi 2.2 Froyo (Frozen Yogurt)

Android Froyo dirilis pada 20 mei 2012. Adroid versi ini memiliki kecepatan kinerja dan aplikasi 2 sampai 5 kali dari versi-versi sebelumnya. Selain itu ada penambahan fitur-fitur baru seperti dukungan Adobe Flash 10.1, integrasi

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

Android Gingerbread di rilis pada 6 Desember 2010. 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 3.0/3.1 Honeycomb

Android Honeycomb di rilis pada awal 2012. Merupakan versi Android yang dirancang khusus untuk device dengan layar besar seperti Tablet PC. Fitur baru yang ada pada Android Honeycomb antara lain yaitu dukungan terhadap prosessor multicore dan grafis dengan hardware acceleration. *User Interface* pada Honeycomb juga berbeda karena sudah didesain untuk tablet. Tablet pertama yang memakai Honeycomb adalah tablet Motorola Xoom yang dirilis bulan Februari 2011. Selain itu sebuah perangkat keras produksi Asus bernama Eee Pad Transformer juga menggunakan OS Android honeycomb dan diharapkan akan masuk ke pasaran Indonesia pada Mei 2011.

8. Android versi 4.0 ICS (Ice Cream Sandwich)

Android Ice Cream Sandwich diumumkan secara resmi pada 10 Mei 2011 di ajang Google I/O Developer Conference (San Francisco), pihak Google mengklaim Android Ice Cream Sandwich akan dapat digunakan baik di smartphone ataupun tablet. Android Ice Cream Sandwich membawa fitur Honeycomb untuk smartphone serta ada penambahan fitur baru seperti membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, terpadu kontak jaringan sosial, perangkat tambahan fotografi, mencari email secara offline, dan berbagi informasi dengan menggunakan NFC. Ponsel pertama yang menggunakan sistem operasi ini adalah Samsung Galaxy Nexus.

9. Android versi 4.1 Jelly Bean

Android Jelly Bean juga diluncurkan pada acara Google I/O 10 Mei 2011 yang lalu. Android versi ini membawa sejumlah keunggulan dan fitur baru, diantaranya meningkatkan input keyboard, desain baru fitur pencarian, UI yang baru dan pencarian melalui Voice Search yang lebih cepat. Versi ini juga dilengkapi Google Now yang dapat memberikan informasi yang tepat pada waktu yang tepat pula. Salah satu kemampuannya adalah dapat mengetahui informasi cuaca, lalu-lintas, ataupun hasil pertandingan olahraga. Sistem operasi Android Jelly Bean 4.1 pertama kali digunakan dalam produk tablet Asus, yakni Google Nexus 7.

10. Android versi 4.2 Jelly Bean

Fitur photo sphere untuk panorama, daydream sebagai screensaver, power control, lock screen widget, menjalankan banyak *user* (dalam tablet saja), widget terbaru. Android 4.2 Pertama kali dikenalkan melalui LG Google Nexus 4. (Alfa Satyaputra, M.Sc : 2014 : 5).

II.4. *Google Map API*

Google Maps adalah layanan pemetaan berbasis web service yang disediakan oleh Google dan bersifat gratis, yang memiliki kemampuan terhadap banyak layanan pemetaan berbasis web. *Google Maps* juga memiliki sifat server side, yaitu peta yang tersimpan pada server Google dapat dimanfaatkan oleh pengguna. *Google Maps API* adalah suatu library yang berbentuk javascript yang berguna untuk memodifikasi peta yang ada di *Google Maps* sesuai kebutuhan. Untuk membangun aplikasi yang memanfaatkan *Google Maps* di desktop dan mobile device maka akan digunakan *Google Maps Javascript API v3* yang memiliki keunggulan lebih cepat dari versi sebelumnya. (Alqod Elian, dkk : 2012 : 1)

II.5. Sekilas Sejarah Java

Bahasa pemrograman Java dimulai dari sebuah tim pengembangan software dari Sun Microsystem yang dipimpin oleh James Gosling dan Patrick Naughton. Pada tahun ini 1991, Sun Microsystem mengembangkan sebuah bahasa pemrograman yang berukuran kecil untuk diimplementasikan pada alat elektronik rumah tangga seperti switchbox TV kabel. Berhubung alat tersebut tidak memiliki

banyak memori, maka bahasa yang digunakan harus sangat kecil dan menghasilkan kode yang kecil pula. Permasalahan lainnya adalah alat-alat tersebut memiliki CPU yang berbeda-beda karena dibuat oleh manufaktur yang berbeda.

Karena keharusan sebuah bahasa pemrograman yang kecil pula dan harus *platform* independen (tidak terikat pada platform) membuat tim pada proyek tersebut terinspirasi ide pemrograman yang sama yang telah ditemukan oleh Niklaus Wirth, penemu Pascal. Dengan pemikiran tersebutlah maka terpikirlah prinsip yang sama ingin diterapkan tim Sun Microsystem, hanya saja pada Java mesin virtualnya disebut Java Virtual Machine (JVM). Bahasa ini pada awalnya disebut "Oak" oleh James Gosling. Akan tetapi menyadari sudah ada bahasa pemrograman yang menggunakan nama tersebut dan mereka menggantinya dengan nama "Java".

Sun sendiri merilis java pertama kalinya pada awal tahun 1996, kemudian diikuti dengan versi 1.02 beberapa bulan kemudian. Pada awalnya Java masih belum mampu memenuhi kebutuhan para pengembang untuk membangun sebuah software secara profesional. Baru pada tahun 1998 muncul versi java 1.2 yang dirilis pada bulan Desember dan beberapa hari kemudian namanya diganti dengan Java 2. (*Jubilee Enterprise*, 2014)

II.5.1. Sintaks Java

Berikut contoh sintaks java dalam pemrograman sederhana.

```
Public class hallo{
```

```
Public static void main(String[] args)

{

System.out.print(“Hallo, selamat datang di Java”);

}

}
```

II.6. Eclipse

Sejak berkurangnya standarisasi pengujian aplikais pada Android, membangun aplikasi pada *operating system* tersebut menjadi lebih mudah dan fleksibel dibandingkan sebelumnya. Sehingga semakin banyak programmer – programmer baru yang lahir dengan beragam kreatifitas dalam mengembangkan aplikasi. Tampilan logo Eclipse kepler terlihat seperti terlihat pada Gambar II.1.



Gambar II.1. Logo Eclipse Kepler
Sumber : (Alfa Satyaputra : 2014 : 12)

Dalam mengembangkan sebuah aplikasi sendiri, dibutuhkan *software* pendukung. Begitu pula dalam mengembangkan aplikasi pada Android, anda juga

memerlukan software seperti *Eclipse* sebagai IDE (*Integrated Development Environment*) atau program computer dengan beberapa fasilitas yang diperlukan dalam pengembangan perangkat lunak. *Eclipse* pertama kali diluncurkan tanggal 21 Juni 2004, sejak itu, setiap tahun pada bulan Juni diluncurkan versi baru dari *Eclipse*. Umumnya versi Eclipse menggunakan nama yang berhubungan dengan astronomi. Pada saat buku ini ditulis, versi terbaru dari *Eclipse* adalah Eclipse Kepler yang dirilis tanggal 26 Juni 2013.

II.7. Konsep UML (*Unified Modelling Language*)

Unified Modelling Language (UML) adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML dapat dibuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan class dan operation dalam konsep dasarnya, maka lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek seperti C++, Java, atau VB. NET. (Prastuti Sulistyorini : 2009 : 1).

II.7.1 Diagram – diagram UML

1. *Use Case Diagram*

Use-case adalah konstruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata pengguna potensial. Use-case terdiri dari sekumpulan skenario

yang dilakukan oleh seorang aktor (orang, perangkat keras, urutan waktu atau sistem yang lain). Sedangkan use-case diagram memfasilitasi komunikasi di antara analis dan pengguna serta diantara analis dan klien. Diagram *Use Case* menunjukkan 3 aspek dari sistem yaitu : actor, use-case, dan system boundary. Actor adalah pengguna sistem, biasanya mewakili peran orang, sistem yang lain atau alat yang berkomunikasi dengan use-case. *Use Case* adalah tugas yg dilakukan oleh *actor*. Sekumpulan use-case biasanya dikelompokkan dalam suatu group yang disebut System Boundary. (Prastuti Sulistyorini : 2009 : 1). Simbol usecase ditunjukkan pada gambar II.5.







Gambar II.2. Actor

Sumber : (Prabowo Pudjo Widodo dan Herlawati : 2010 : 20).

2. *Activity Diagram*

Activity Diagram 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 (Radiant Victor Imbar dan Yuliusman Kurniawan : 2012 : 5).

Simbol	Keterangan
	Titik Awal
	Titik Akhir
	Activity
	Pilihan untuk pengambilan keputusan

Gambar II.3. Simbol-simbol pada *Activity Diagram*

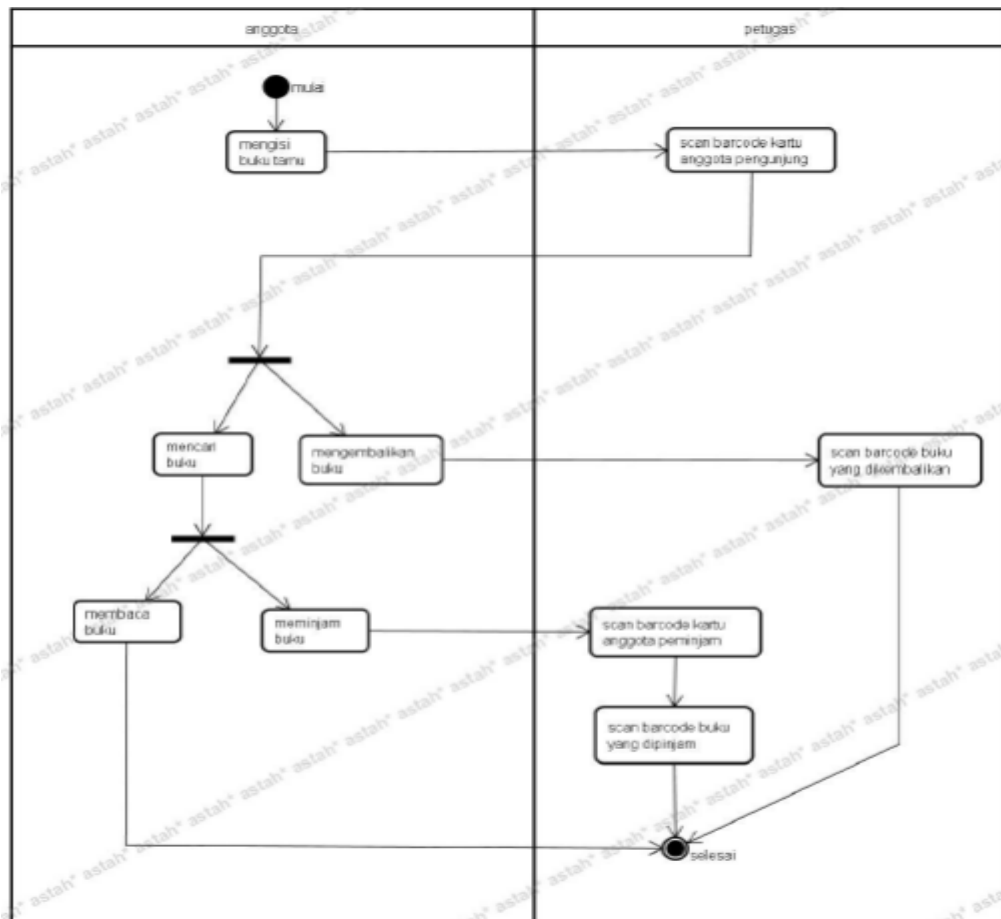
Sumber : (Radiant Victor Imbar dan Yuliusman Kurniawan : 2012 : 5).

3. *Class Diagram*

Class Diagram membantu dalam visualisas istruktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang palingbanyak. *Class Diagram* memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain(dalam logical view) dari suatu sistem. Selama proses analisis, *Class Diagram* memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Selama proses analisis, *Class Diagram* memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Selama tahap decían, *Class Diagram* berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat. *Class Diagram* juga merupakan pondasi untuk component diagram dan deployment diagram. (Prastuti Sulistyorini : 2009 : 4).

Class Diagram menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut, operasi dan hubungan antara kelas. *Class Diagram* membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Selama tahap

desain, *Class Diagram* berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat. (Haviluddin : 2011 : 3).



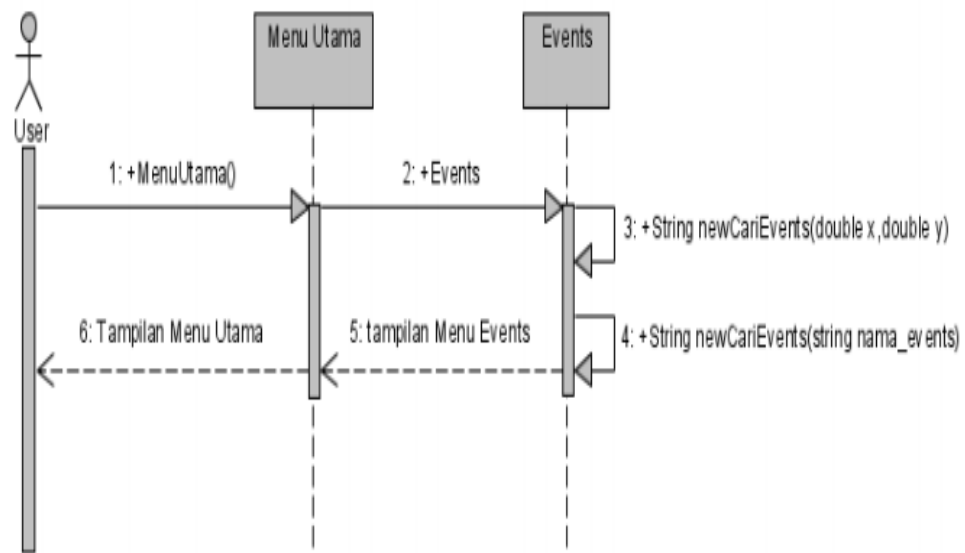
Gambar II.4. Contoh Simbol *Class Diagram*

Sumber : (Haviluddin : 2013 : 3).

4. *Sequence Diagram*

Sequence Diagram menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara khusus berasosiasi dengan usecase. *Sequence Diagram* memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu didalam *Use Case*. Diagram sequence sebaiknya

digunakan diawal tahap desain atau analisis karena kesederhanaannya dan mudah untuk dimengerti. *Sequence Diagram* menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya *Sequence Diagram* adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan usecase diagram. (Haviluddin : 2011 : 5)



Gambar II.5. Contoh Sequence Diagram

Sumber : (S. Nofan Maulana Rachman : 2012 : 9)