

## **BAB II**

### **LANDASAN TEORI**

#### **II.1. Android**

Menurut Safaat (2012 : 1), Android adalah sistem operasi berbasis Linux bagi telepon seluler seperti telepon pintar dan komputer tablet. Android juga menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang akan digunakan untuk berbagai macam piranti gerak. Awalnya, *Google Inc.* membeli *Android Inc.*, pendatang baru yang membuat piranti lunak untuk ponsel. kemudian dalam pengembangan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan piranti keras, piranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia.

Dalam sub bab ini akan membahas mengenai berbagai macam hal mengenai Android, serta tools apa saja yang akan digunakan untuk mengembangkan aplikasi yang akan di rancang.

##### **II.1.1. Versi Android**

Safaat (2012 : 10) memaparkan penjelasan berbagai macam versi Android seperti di bawah ini :

1. Android 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 pembaharuan untuk 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 indikator baterai 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.

Agar dapat bergerak cepat dalam persaingan perangkat generasi yang berikutnya, 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 bertambahnya jumlah *handset* Android, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka dalam sistem operasi Android. Aplikasi terkenal yang diubah ke dalam sistem operasi Android adalah *Shazam*, *Background*, 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 Yoghurt*)

Pada 20 Mei 2010, Android versi 2.2 (*Froyo*) diluncurkan. Perubahan-perubahan umum terhadap versi-versi sebelumnya antara lain dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, 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*)

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) yang sudah 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* 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 multiprocessor dan juga akselerasi perangkat keras (hardware) untuk grafis. Tablet pertama kali yang dibuat dengan

menjalankan *Honeycomb* adalah *Motorola Xoom*. Perangkat tablet tersebut bernama *Eee Pad Transformer* yang merupakan produk dari Asus yang masuk ke pasar Indonesia pada Mei 2011.

8. Android versi 4.0 (*ICS: Ice Cream Sandwich*)

Pada tanggal 19 Oktober 2011, diperkenalkannya Android versi 4.0 yang membawa fitur *Honeycomb* untuk smartphone, menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, kontak jaringan sosial terpadu, perangkat tambahan fotografi, pencarian 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* yang diluncurkan pada acara *Google I/O* membawa sejumlah keunggulan dan fitur baru. Adapun penambahan fitur baru diantaranya yaitu meningkatkan input keyboard, desain baru fitur pencarian, UI yang baru dan pencarian melalui *Voice Search* yang lebih cepat. *Google Now* yang juga menjadi bagian yang diperbarui pun tak ketinggalan. *Google Now* memberikan informasi yang tepat pada waktu yang tepat pula. Salah satu kemampuannya adalah dapat mengetahui informasi cuaca, lalu-lintas, atau pun hasil pertandingan olahraga. Sistem operasi Android *Jelly Bean 4.1* muncul pertama kali dalam produk tablet *Asus*, yakni *Google Nexus 7*.

### II.1.2 Fitur Android

Adapun beberapa fitur – fitur menurut Sifaat (2012:5) yang tersedia di Android adalah sebagai berikut ini

- a. Kerangka aplikasi: memungkinkan penggunaan dan penghapusan komponen yang tersedia.
- b. *Dalvik* mesin *virtual*: mesin *virtual* dioptimalkan untuk perangkat telepon seluler.
- c. Grafik: grafik di 2D dan grafis 3D berdasarkan pustaka *OpenGL*.
- d. *SQLite*: untuk penyimpanan data.
- e. Mendukung media: audio, video, dan berbagai format gambar (*MPEG4*, *H.264*, *MP3*, *AAC*, *AMR*, *JPG*, *PNG*, *GIF*)
- f. *GSM*, *Bluetooth*, *EDGE*, *3G*, *4G* dan WiFi (tergantung piranti keras)
- g. Kamera, *Global Positioning System (GPS)*, kompas, *NFC* dan *accelerometer* (tergantung piranti keras).

### II.1.3 Android SDK

Menurut Sifaat (2012:5), Android SDK adalah *tools API*(*Application Programming Interface*) yang diperlukan untuk mengembangkan aplikasi pada *platform* Android yang menggunakan bahasa pemrograman Java. Android merupakan *subset* perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang di-*release* oleh *Google*. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk mengembangkan aplikasi pada platform Android menggunakan bahasa

pemrograman Java. Sebagai *platform* Android aplikasi – netral, Android memberi Anda kesempatan untuk membuat Aplikasi yang kita butuhkan yang bukan merupakan aplikasi bawaan *Handphone/Smartphone*.

#### **II.1.4. ADT (*Android Development Tools*)**

Safaat (2012:6) juga memaparkan bahwa *Android Development Tools (ADT)* adalah *plug-in* yang didesain untuk *IDE Eclipse* yang memberi kita kemudahan dalam mengembangkan aplikasi Android dengan menggunakan *IDE Eclipse*. Dengan menggunakan *ADT* untuk *Eclipse*, ini akan memudahkan kita dalam membuat aplikasi *project Android*, membuat *GUI aplikasi*, dan menambahkan komponen–komponen yang lainnya. Selain itu kita juga dapat melakukan *running* aplikasi menggunakan *Android SDK* melalui *Eclipse*. Dengan *ADT* kita juga dapat melakukan pembuatan *package Android (.apk)* yang digunakan untuk distribusi aplikasi Android yang kita rancang.

Mengembangkan aplikasi Android dengan menggunakan *ADT* di *Eclipse* sangat mudah dan baik untuk memulai mengembangkan aplikasi Android. Semakin tinggi *platform* Android yang kita gunakan, maka penggunaan *ADT* dianjurkan untuk memakai versi yang lebih baru, karena munculnya *platform* baru diikuti oleh munculnya versi *ADT* yang terbaru. Versi *ADT* yang kami gunakan untuk *Eclipse* dalam pembuatan aplikasi ini adalah versi terbaru *ADT* yang telah dirilis, yaitu *ADT 20.0.3* (Agustus 2012).

### **II.1.5. AVD (*Android Virtual Device*)**

Menurut Safaat (2012:19), *AVD* merupakan emulator yang digunakan untuk menjalankan program aplikasi Android yang telah dirancang. *AVD* dapat dikonfigurasi agar dapat menjalankan berbagai macam versi Android yang telah diinstal. Dalam perancangan aplikasi ini, kami menggunakan konfigurasi Android versi 4.0 (*Ice Cream Sandwich*).

### **II.1.6. IDE Eclipse**

Menurut Fatimah (2011), Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak agar dapat di jalankan di semua platform (*platform-independent*). Berikut ini adalah sifat dari Eclipse:

- a. *Multi-platform* : Target sistem operasi Eclipse adalah *Microsoft Windows, Linux, Solaris, AIX, HP-UX* dan *Mac OS X*.
- b. *Multilanguage* : Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti *C/C++, Java, Cobol, Python, Perl, PHP*, dan lain sebagainya.
- c. *Multi-role* : Selain sebagai *IDE* untuk pengembangan aplikasi, Eclipse pun dapat 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 bersifat *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari *Eclipse* adalah kemampuannya untuk dapat dikembangkan oleh *user* dengan komponen yang dinamakan *plug-in*. *Eclipse* yang kami gunakan untuk mengembangkan aplikasi ini adalah *Eclipse Classic 4.2.1*

### **II.1.7. JDK (*Java Development Kit*)**

Menurut DeCoster (2012), *Java* adalah sebuah teknologi yang diperkenalkan oleh *Sun Microsystems* pada pertengahan tahun 1990. Menurut definisi *Sun*, *Java* adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada *computer standalone* ataupun pada lingkungan jaringan. Untuk membuat program *Java* dibutuhkan kompiler dan interpreter untuk program *Java* berbentuk *Java Development Kit (JDK)* yang diproduksi oleh *Sun Microsystems*. Sebelum memulai instalasi *Android SDK*, terlebih dahulu kita harus melakukan instalasi *JDK* di komputer. *JDK* yang kami gunakan untuk dapat mengompilasi aplikasi android yang kami rancang ini adalah *Java SE Development Kit 7*.

## **II.2. Kamus**

### **II.2.1 Pengertian Kamus**

Pengertian Kamus adalah sejenis buku rujukan yang menerangkan makna kata-kata. berfungsi untuk membantu seseorang mengenal perkataan baru. Selain

menerangkan maksud kata, kamus juga mungkin mempunyai pedoman sebutan, asal-usul (etimologi) sesuatu perkataan dan juga contoh penggunaan bagi sesuatu perkataan. Untuk memperjelas kadang kala terdapat juga ilustrasi di dalam kamus. Dewasa ini kamu merupakan khazanah yang memuat perbendaharaan kata suatu bahasa, yang secara ideal tidak terbatas jumlahnya. Setiap kebudayaan besar di dunia bangga akan kamus bahasanya. Dalam kenyataannya kamus itu tidak hanya menjadi lambang kebanggaan suatu bangsa, tetapi juga mempunyai fungsi dan mafaat praktis.(Dep. Pendidikan dan Kebudayaan, 2002:499).

### **II.2.2 Kamus Elektronik**

Pada saat ini kamus telah memiliki berbagai bentuk mulai dari kamus dalam bentuk buku serta dalam bentuk elektronik, kamus dalam bentuk buku memiliki kelebihan dalam jumlah kosa kata yang banyak, tetapi juga memiliki kelemahan dalam hal pencarian kosa kata dan arti kata yang membutuhkan waktu yang lama(Balbeid, 2008).

Media kamus dalam bentuk elektronik (kamus elektronik, *aplikasi desktop*, web dan mobile kamus) mengatasi masalah dalam pencarian kata-kata yang cepat. Dengan di gunakan alat-alat tersebut, pencarian kosa kata dan arti kata menjadi cepat dan praktis, tetapi yang menjadi masalah adalah alat-alat tersebut tidak dapat digunakan kapan saja dan dimana saja, kecuali kamus elektronik dan aplikasi mobile kamus. Pada kamus elektronik dan aplikasi mobile kamus memiliki kelemahan dalam hal keterbatasan jumlah data kata.

## II.3 Defenisi MYSQL

MYSQL adalah DBMS yang di distribusikan secara gratis dibawah lisensi dari *General Public License (GPL)*, dimana setiap orang bebas untuk menggunakannya tetapi tidak boleh untuk dijadikan program induk turunan bersifat *close source* (komersial).

MYSQL sebenarnya merupakan turunan dari salah satu konsep utama dalam basis data sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basis data terutama proses seleksi, pemasukan, pengubahan, dan penghapusan data yang dimungkinkan dapat dikerjakan dengan mudah dan otomatis.

### II.3.1 Keunggulan MYSQL

Berikut beberapa keunggulan dari MYSQL, diantaranya :

#### 1. *Portability*

Dapat berjalanstabil pada berbagai sistem operasi

#### 2. *Open source*

Didistribusikan secara gratis dibawah lisensi dari *General Public License (GPL)* dimana setiap orang bebas untuk menggunakannya tetapi tidak boleh menggunakan MYSQL untuk dijadikan induk turunan yang bersifat *close source* (komersial).

#### 3. *Multi User*

Dapat digunakan oleh beberapa user dalam waktu yang bersamaan.

#### 4. *Performance tuning*

Mempunyai kecepatan yang tinggi dalam menangani *query*.

5. *Column types*

Memiliki tipe data yang sangat kompleks.

6. *Command dan function*

Memiliki operator dan fungsi penuh yang mendukung *select* dan *where* dalam *query*.

7. *Security*

Memiliki beberapa lapisan sekuritas seperti tingkat subnet mask, hostname, privilege user dengan sistem perijinan yang mendetail serta password yang terenkripsi.

8. *Scability dan limits*

Mampu menangani basis data dalam jumlah besar, dengan jumlah field lebih dari 50 juta, 60 ribu tabel, dan 5 milyar record. Batas indeks mencapai 32 buah per tabel.

9. *Localization*

Dapat mendeteksi pesan kesalahan (*error code*) pada klien dengan menggunakan lebih dari 20 bahasa.

10. *Connectivity*

Dapat melakukan koneksi dengan klien menggunakan protocol TCP/IP, Unix Socket, Named Pipes.

11. *Interface*

Memiliki antar muka terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API.

## 12. *Clients dan Tools*

Dilengkapi dengan berbagai tool yang dapat digunakan untuk administrasi basis data sekaligus dokumen petunjuk *online*.

## 13. Struktur tabel

Memiliki struktur tabel yang lebih fleksibel dalam menangani *alter tabel* dibandingkan dengan PostgreSQL dan Oracle.

## II.4 Database (Basis data)

Menurut Hengky W. Pramana (2002 : 3) ”Basis data adalah sebuah kumpulan dari sekelompok informasi yang diorganisasikan dengan beberapa cara logik dan saling berhubungan”.

Menurut Aryanto (2001 : 19) “*Database* merupakan keterangan mengenai kumpulan sejumlah tabel, prosedur tersimpan (*stored procedure*) dan hubungan relasi antar tabel yang saling berhubungan dalam membentuk suatu program aplikasi”.

### II.4.1 Komponen - komponen Sistem Basis Data

Menurut Hariyanto (2004 : 56) ”sistem basis data merupakan sistem penyusunan data yang saling terpadu, yang mempunyai komponen – komponen sebagai berikut” :

1. *Database* (basis data) adalah kumpulan *file-file* yang saling berhubungan atau berelasi untuk menghasilkan suatu informasi.

2. *Software* (perangkat lunak) adalah perangkat lunak yang digunakan dalam suatu sistem basis data. Misalnya: *SQL, Microsoft Access*.
3. *Hardware* (perangkat keras), merupakan perangkat keras yang digunakan dalam sistem basis data yaitu unit pusat pengolahan (*Central Processing Unit* atau *CPU*), unit penyimpanan (*Storage Unit*), *keyboard*, monitor, printer dan lain-lain.
4. *Brainware* (manusia), merupakan elemen penting pada sistem basis data yang terdiri dari sistem engineer, administrator basis data, *programmer* dan pemakai akhir.

## **II.5 UML (*Unified Modeling Language*)**


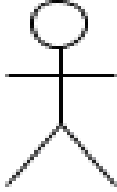


Menurut Fowler Martin (2004 : 2) *Unified Modeling Language* (UML) adalah keluarga notasi grafis yang didukung oleh meta model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (OO).


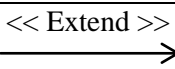
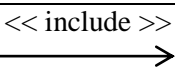
Selain itu UML adalah bahasa pemodelan yang menggunakan konsep orientasi objek. UML dibuat oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson dibawah bendera Rational Software Corp, UML menyediakan notasi – notasi yang membantu memodelkan sistem dari berbagai perspektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. UML dideskripsikan oleh beberapa diagram, diantaranya :

### II.5.1.Uses Case Diagram

Use Case Diagram adalah suatu representasi / model yang digunakan pada rekayasa perangkat lunak yang menunjukkan sekumpulan use case dan aktor serta hubungan diantara keduanya. Simbol-simbol Uses Case Diagram dapat dilihat pada tabel 2.1 berikut ini :

**Tabel 2.1 Simbol Use Case Diagram**

Nama Komponen	Deskripsi	Gambar
<i>Use Case</i>	Menerangkan “apa” yang dikerjakan sistem, bukan “bagaimana” sistem mengerjakannya.	
<i>Actor</i>	Menggambarkan orang, sistem atau <i>external</i> entitas / stakeholder yang menyediakan atau menerima informasi dari sistem.	
<i>Sistem Boundary</i>	Menggambarkan jangkauan sistem.	
<i>Association</i>	Menggambarkan bagaimana <i>actor</i> terlibat dalam <i>use case</i> .	

<i>Generalization</i>	Dibuat ketika ada sebuah keadaan yang lain / perlakuan khusus.	
<i>Extend</i>	Perluasan dari use case lain jika kondisi atau syarat terpenuhi.	
<i>Include</i>	Menjelaskan bahwa use case termasuk didalam use case lain.	

Sumber : Buku Panduan Belajar STMIK Triguna Dharma



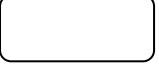
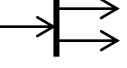
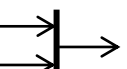

## II.5.2. Activity Diagram

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

Activity diagram tidak menggambarkan behaviour internal sebuah sistem (interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur - jalur aktivitas dari level atas secara umum, proses bisnis dan urutan aktivitas dalam sebuah proses.

Activity diagram merupakan *state* diagram khusus, dimana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu activity diagram tidak menggambarkan behaviour *internal* sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Simbol-simbol activity diagram dapat dilihat pada tabel II.2 berikut :

**Tabel II.2 Simbol Activity Diagram**

Simbol	Keterangan
 	Start Point  End Point
	Activitas
	Fork (Percabangan)
	Join (Penggabungan)
	Decision
Swimlane	Sebuah cara untuk mngelompokkan aktivitty berdasarkan Actor (mengelompokkan aktivitty

	dalam sebuah urutan yang sama).
--	---------------------------------

Sumber : Buku Panduan Belajar STMIK Triguna Dharma





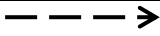
### II.5.3. Class Diagram

Object adalah gambaran dari entity, baik dunia nyata atau konsep dengan batasan-batasan dan pengertian yang tepat. Objek biasa mewakili sesuatu yang nyata seperti computer, mobil atau dapat berupa konsep seperti proses kimia, transaksi bank, permintaan pembelian, dan lain-lain. Setiap objek dalam sistem memiliki tiga karakteristik yaitu State (Status), Behaviour (Sifat), dan Identity (Identitas).

Class adalah deskripsi sekelompok objek dari property (atribut), sifat (operasi), relasi antar objek dan semantik yang umum. Class merupakan template untuk membentuk objek. Setiap objek merupakan contoh lebih dari satu class. Class adalah sebuah spesifikasi yang jika di instansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (*metoda / fungsi*). Simbol-simbol yang digunakan pada *class diagram* dapat dilihat pada tabel II.3 berikut ini :

**Tabel II.3 Simbol Class Diagram**

Indikator / Gambar	Deskripsi
0..1	Kosong atau satu
0..*	Lebih dari sama dengan kosong

0..n	Lebih dari sama dengan n, dimana n lebih dari 1
1	Hanya 1
1..*	Lebih dari sama dengan 1
1..n	Lebih dari sama dengan 1 dimana n lebih dari 1
*	Banyak atau <i>Many</i>
N	Hanya N, dimana N lebih dari 1
n..*	Lebih dari sama dengan n dimana n lebih dari 1
n..m	Lebih dari sama dengna n dan kurang dari sama dengan m. Dimana m dan n lebih dari 1
	Agregasi ( <i>Aggregation</i> )
	Kunci gabungan ( <i>Composite Key</i> )
	Turunan ( <i>Inheritance</i> )
	Relasi ( <i>Relation</i> )
	<i>Dependencies</i>

<b>--- --&gt;</b>	<i>Realizations</i>
<b>+</b>	<i>Public</i>
<b>#</b>	<i>Protected</i>
<b>-</b>	<i>Private</i>
<b>\$</b>	<i>Static</i>
<b>/</b>	<i>Drived</i>
<b>*</b>	<i>Abstrak</i>

*Sumber : Buku Panduan Belajar STMIK Triguna Dharma*

#### **II.5.4 Tujuan Penggunaan UML**

1. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
2. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan.
3. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
4. UML bisa juga berfungsi sebagai sebuah (blue print) cetak biru karena sangat lengkap dan detail.

### II.5.5 Defenisi Flowchart


Flowchart adalah penggambaran secara grafik dari langkah – langkah dan urutan – urutan prosedur dari suatu program. Flowchart menolong analis dan programmer untuk memecahkan masalah kedalam segmen – segmen yang lebih kecil dan menolong dalam menganalisis alternatif – alternatif lain dalam pengoperasian. Flowchart biasanya memepermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut.






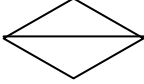

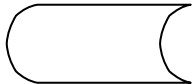
Tujuan utama penggunaan flowchart adalah untuk menyederhanakan rangkaian proses atau prosedur untuk memudahkan pemahaman pengguna terhadap informasi tersebut. Oleh karena itu, design sebuah flowchart harus ringkas, jelas, dan logis.

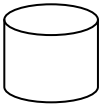




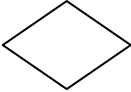
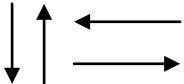

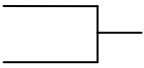
Flowchart memperlihatkan urutan dan hubungan antar proses beserta instruksinya. Gambaran ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan hubungan antar proses digambarkan dengan garis penghubung.



Simbol - simbol flowchart dapat dilihat pada tabel II.4 berikut :

**Tabel II.4 Simbol Flowchart**

Simbol	Pengertian	Keterangan
	Document	Menggamburkan semua jenis dokumen yang merupakan formulir yang digunakan untuk data terjadinya suatu transaksi

	Manual Operation	Digunakan untuk menggambarkan kegiatan manual
	Punched Card	Menunjukkan input atau output yang menggunakan kartu plong ( <i>Punched Card</i> )
	Process	Menunjukkan kegiatan proses dari operasi program komputer
	Simpanan (arsip)	Digunakan untuk menyimpan data sebagai arsip secara manual. Untuk menunjukkan urutan pengarsipan dokumen, digunakan simbol berikut: "A"= menurut abjad "N"= menurut nomor "C"= kronologis, menurut tanggal.
	Predefined Process	Rincian proses berada ditempat / halaman berbeda
	Pengurutan	Menunjukkan proses pengurutan data diluar proses komputer
	Magnetic Tape	Menunjukkan input/output menggunakan pita magnetik
	<i>Stored Data</i>	Menunjukkan input/output menggunakan penyimpanan akses

		langsung
	Magnetic Disk	Menunjukkan input/output menggunakan <i>Disk Magnetic</i>
	Magnetic Drum	Menunjukkan input/output menggunakan drum magnetic
	Punched Tape	Menunjukkan input/output menggunakan pita kertas berlubang
	Keying (typing, verifying)	Menggambarkan pemasok data kedalam komputer melalui on-line terminal
	Input / Output	Merepresentasikan input / output data
	Decision	Menggambarkan keputusan yang harus dibuat dalam proses pengolahan data
	Flow line	Menggambarkan arah dari proses
	Connector	Menunjukkan penghubung kehalaman yang masih sama atau ke halaman lain
	Keterangan, komentar	Digunakan untuk memperjelas pesan yang disampaikan dalam

		bagian alir
	Preparation	Pemberian harga awal
	Terminator	Menggambarkan awal dan akhir suatu sistem akuntansi