

BAB II

LANDASAN TEORI

2.1 Perancangan

Definisi perancangan menurut bin Ladjamudin (2005:39)” Perancangan adalah tahapan (design) memiliki tujuan untuk merancang sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternative sistem yang terbaik”.

Definisi perancangan menurut Syifaun Nafisah (2003:2)” Perancangan adalah, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam suatu kesatuan yang utuh dan berfungsi. Perancangan sistem dapat dirancang dalam bentuk bagan alir sistem (*system flowchart*), yang merupakan alat bentuk grafik yang dapat digunakan untuk menunjukkan urutan-urutan proses dari sistem”.

2.2 Media Pembelajaran

Media pembelajaran adalah alat bantu dalam proses belajar mengajar. Sesuatu yang dapat dipergunakan untuk merangsang pikiran, perhatian, perasaan, dan kemampuan atau ketrampilan pembelajar tersebut sehingga dapat mendorong terjadinya proses belajar atau kegiatan pembelajaran.

Secara umum dapat dikatakan bahwa substansi dari media pembelajaran adalah bentuk saluran, yang digunakan untuk menyalurkan pesan, informasi atau bahan pelajaran kepada penerima pesan atau pembelajar dapat dikatakan bahwa media pembelajaran adalah berbagai jenis komponen dalam lingkungan pembelajar yang dapat merangsang pembelajar untuk belajar.

2.2.1. Fungsi dan Manfaat Media Pembelajaran

Arsyad (2014. P.31) mengemukakan bahwa “Pemakaian media pembelajaran dalam proses belajar mengajar dapat membangkitkan keinginan dan minat yang baru,

membangkitkan motivasi dan rangsangan kegiatan belajar, dan bahkan membawa pengaruh-pengaruh psikologis terhadap siswa”. Secara umum manfaat yang dapat diperoleh adalah proses pembelajaran menjadi lebih menarik dan tidak membosankan, lebih interaktif, dan dapat meningkatkan kualitas belajar siswa.

2.2.2. Jenis Media Pembelajaran

Dengan berkembangnya teknologi, media pembelajaran dapat dikembangkan dengan memanfaatkan teknologi. Berdasarkan teknologi tersebut, Azhar Arsyad (2011) mengklasifikasikan media atas empat kelompok, yaitu :

- a. Media hasil teknologi cetak.
- b. Media hasil teknologi audio-visual.
- c. Media hasil teknologi yang berdasarkan komputer.
- d. Media hasil gabungan teknologi cetak dan komputer.

Seels dan Glasgow (dalam Azhari Arsyad 2011: 33) membagi media kedalam dua kelompok besar, yaitu :

- a. Media tradisional
 - 1) Visual diam yang diproyeksikan, yaitu proyeksi overhead, slides.
 - 2) Visual yang tak diproyeksikan, yaitu gambar, poster, foto, charts, grafik, diagram, papan info.
 - 3) Audio, yaitu rekaman piringan, pita kaset, reel, cartridge.
 - 4) Penyajian multimedia, yaitu slide plus suara (tape).
 - 5) Visual dinamis yang diproyeksikan, yaitu film, televisi, video.
 - 6) Media cetak, yaitu buku teks, modul, majalah ilmiah, workbook.
 - 7) Permainan, yaitu teka-teki, simulasi, permainan papan.
 - 8) Media replika, yaitu model, specimen (contoh), manipulatif (peta, boneka).
- b. Media teknologi mutakhir
 - 1) Media berbasis telekomunikasi, yaitu telekonferen, kuliah jarak jauh.
 - 2) Media berbasis mikroprosesor, yaitu permainan komputer, computer assisted instruction, compact (video) disc.

2.3 Android

Android adalah sistem operasi untuk telepon seluler berbasis linux sebagai karnelnya. Android menyediakan platform terbuka (open source) bagi para pengembang untuk menciptakan aplikasi mereka sendiri. Awalnya, perusahaan search engine terbesar saat ini, yaitu Google Inc, membeli Android Inc, pendatang baru yang membuat perangkat lunak untuk ponsel. Android Inc. Didirikan oleh Andy Rubin, Rich Milner, Nick Sears dan Chris White pada tahun 2003. Pada Agustus 2005 Google membeli Android Inc. Dimulai pada tahun 2005, Android Inc. dibawah naungan Google Inc. Berusaha membuat sebuah operating system mobile baru. Sejak saat itulah mulai beredar rumor bahwa Google akan melakukan ekspansi bisnis ke industri seluler. Akhirnya pada bulan September 2007 Google mengajukan hak paten atas produknya yang dinamai Nexus One. Kemudian pada akhir tahun 2008, dibentuk sebuah tim kerja sama yang dinamai Open Handset Alliance (OHA). OHA ini terdiri dari beberapa produsen perangkat telekomunikasi ternama dunia, antara lain ASUS, Toshiba, Sony Ericsson (sekarang Sony), Garmin, Vodafone, dan Softbank. OHA bekerja sama untuk mengembangkan sebuah kernel Linux yang akan dijadikan sebuah program untuk perangkat seluler. Hingga akhirnya OHA berhasil dan mengumumkan produk operating system mobile yang diberi nama Android. Ponsel yang mendapat kehormatan untuk mencoba pertama kali sistem operasi Android adalah HTC Dream. HTC Dream dirilis pada bulan Oktober tahun 2008. Sejak saat itu banyak perusahaan perangkat seluler di dunia ikut menggunakan Android sebagai operating system ponsel mereka.

2.3.1. Komponen Android

Android SDK (Software Development Kit) merupakan sebuah tools yang diperlukan untuk mengembangkan aplikasi berbasis Android menggunakan bahasa pemrograman Java. Pada saat ini Android SDK telah menjadi alat bantu dan API (Application Programming Interface) untuk mengembangkan aplikasi berbasis android. Android SDK dapat anda lihat dan unduh pada situs resminya, yaitu <http://developer.android.com/>, Android SDK bersifat gratis dan bebas anda distribusikan karena Android bersifat open source.

Berikut adalah garis besar pada arsitektur yang Android miliki :

1. Applications dan widget
2. Applications Frameworks
3. Libraries
4. Android Run Time
5. Linux Kernel

2.3.2. Jenis dan Versi Android

Seiring berkembangnya teknologi khususnya dalam bidang pemrograman Android maka terciptalah beberapa versi dari Android. Berikut adalah perkembangan versi Android :

- 1) Android versi awal (2007 - 2008)

Android Beta, pertama kali di rilis tanggal 5, November, 2007, dan versi SDK (Software developer Kit) beta di rilis tanggal 12, November di tahun yang sama.

- 2) 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 e-mail.

- 3) Android versi 1.5 (Cupcake)

Android Cup Cake di rilis pada pertengahan Mei 2009, masih oleh Google Inc. Adroid 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.

- 4) 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.

5) 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 smartphne, fitur utama Eclair yaitu perubahan total struktur dan tampilan user interface.

6) 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, intergrasi V8 JavaScript engine yang dipakai Google Chrome yang mempercepat kemampuan rendering pada browser, pemasangan aplikasi dalam SD Card, kemampuan WiFi Hotspot portabel, dan kemampuan auto update dalam aplikasi AndroidMarket.

7) Android versi 2.3 Gingerbread

Andoid 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.

8) 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.

9) 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 e-mail secara offline, dan berbagi informasi dengan menggunakan NFC. Ponsel pertama yang menggunakan sistem operasi ini adalah Samsung Galaxy Nexus.

10) Android versi 4.1.x/4.2.x/4.3 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.

11) Android versi 4.4 KitKat

Android KitKat ini diluncurkan pada tanggal 14 Oktober 2013, menurut beberapa sumber di versi ini akan membawakan beberapa fitur-fitur baru dari versi sebelumnya yang diantaranya pergantian fragmentation atau ukuran pada layar, boot animasi yang baru yaitu pergantian warna logo android, hadirnya notifikasi widget baru, updatenya gallery baru miracast, dan firmware yang dapat di gunakan untuk smartphone yang lama.

12) Android versi 5.0.x/5.1.x Lollipop

Android Lollipop adalah versi stabil terbaru dari sistem operasi Android yang dikembangkan oleh Google, yang pada saat ini mencakup versi antara 5.0 dan 5.1. Diresmikan pada 25 Juni 2014 saat Google I / O, dan tersedia secara resmi melalui over-the-air (OTA) update pada tanggal 12 November 2014, untuk memilih perangkat yang menjalankan distribusi Android dilayani oleh Google (seperti perangkat Nexus dan Google Play edition). Kode sumbernya dibuat tersedia pada 3 November 2014.

13) Android versi 6.0 Marshmallow

Android Marshmallow merupakan pemutakhiran yang akan datang untuk sistem operasi telepon genggam Android, kemungkinan besar akan dirilis pada Q3 2015 ("sementara dijadwalkan untuk September"), dengan pratayang ketiga dan terakhir dirilis pada tanggal 17 Agustus 2015. Pertama diperkenalkan di Google I/O pada tanggal 28 Mei 2015, Marshmallow terutama akan berfokus pada perbaikan inkremental dan penambahan fitur lainnya.

2.4 Perangkat Lunak yang Digunakan

2.4.1 Adobe Flash professional CS6

Adobe Flash Professional CS6 merupakan versi terbaru dari Adobe Flash yang dirilis tahun 2012 sebagai penyempurna dari versi adobe flash sebelumnya, yaitu adobe flash CS5 dimana pada versi ini telah mampu mengolah teks maupun objek dengan efek 3dimensi sehingga tampak lebih menarik (Madcoms 2012,p.2). Hasil penyimpanan file dari Adobe Flash Professional CS6 tidak dapat dibuka pada program flash versi sebelumnya. Apabila hasil file ingin dibuka pada versi sebelumnya, maka pada format penyimpanannya pilih tipe flash pada bagian save as a type. Area kerja Adobe flash Professional CS6 dirancang secara khusus agar ruang kerja yang digunakan dapat diatur dan lebih mudah dipahami oleh pemakai pemula maupun para desainer flash yang telah berpengalaman. Adobe Flash Professional CS6 adalah program yang cukup kompleks sehingga tidak mungkin untuk

menjabarkannya secara lengkap. Disini hanya akan diberikan sedikit pengertian dan fungsi dasar dari Adobe Flash Professional CS6.

2.4.2 Adobe Photoshop CS

Photoshop CS adalah versi terbaru photoshop, merupakan program terbaik editing gambar Adobe. Photoshop CS adalah bagian dari perangkat kreatif Adobe, sebuah paket program desain untuk orang-orang yang bekerja, baik dengan gambar cetak atau grafis jaringan. Diantara fitur-fitur baru di Photoshop CS adalah jangkauan perbaikan produktivitas dan pilihan menarik untuk Fotografer. Photoshop CS juga memasukkan program ImageReady yang terintegrasi sangat dekat dengan Photoshop CS untuk memfasilitasi pembuatan grafis jaringan.

2.4.3 Adobe Premiere Pro

Semenjak diluncurkan pertama kali, Adobe Premiere telah menjadi standar pilihan dalam bidang pengolahan dan pengeditan video dan animasi oleh para pelaku multimedia. Hal ini dikarenakan adanya berbagai kemudahan dan fasilitas tool editor yang ditawarkan dan kemudahan bagi pengguna profesional dan pemula untuk menuangkan kreasi mereka dalam bereksperimen dengan file-file video. Hal ini telah membuat apa yang dahulu memerlukan biaya yang banyak dan perangkat keras yang bernilai mahal kini dapat dilakukan dengan computer pribadi di rumah.

2.5 Metode Prototype

Proses pengembangan sistem seringkali menggunakan pendekatan prototype (*prototyping*). Metode ini sangat baik digunakan untuk menyelesaikan masalah kesalahpahaman antara *user* dan analis yang timbul akibat *user* tidak mampu mendefinisikan secara jelas kebutuhannya (Mulyanto, 2009).

Prototyping adalah pengembangan yang cepat dan pengujian terhadap model kerja (prototipe) dari aplikasi baru melalui proses interaksi dan berulang-ulang yang biasa digunakan ahli sistem informasi dan ahli bisnis. *Prototyping* disebut juga desain aplikasi cepat (*rapid application design/RAD*) karena menyederhanakan dan

mempercepat desain sistem (O'Brien, 2005). Metode prototyping dibagi dalam 5 sistem antara lain:

1) Analisis dan definisi persyaratan

Analisis dan definisi persyaratan adalah proses pengumpulan kebutuhan dilakukan secara intensif untuk memesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

2) Perancangan sistem

Perancangan sistem adalah proses multistep yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antar muka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

3) Implementasi dan pengujian unit

Perancangan sistem dan perangkat lunak ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4) Integrasi dan pengujian sistem

Pengujian fokus pada perangkat lunak secara dari segi logis dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5) Operasi dan pemeliharaan

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap

pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, namun tidak

untuk membuat perangkat lunak baru. Berikut contoh sistem model *prototyping*, dijelaskan pada Gambar 2.1:



Gambar 2.1 Sistem Model *prototyping*

Kelebihan dari metode *prototyping* ini sebagai berikut :

- 1) Adanya komunikasi yang baik antara pengembang dan pelanggan
- 2) Pengembangan dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan.
- 3) Lebih menghemat waktu dalam pengembangan sistem.
- 4) Penerapan menjadi lebih mudah karna pemakaian mengetahui apa yang diharapkannya.

Kekurangan dari metode prototype ini sebagai berikut :

- 1) Resiko tinggi yaitu untuk masalah-masalah yang tidak terstruktur dengan baik, ada perubahan yang besar dari waktu ke waktu, dan adanya persyaratan data yang tidak menentu.
- 2) Interaksi pemakai penting. Sistem harus menyediakan dialog online antara pelanggan dan komputer.
- 3) Hubungan pelanggan dengan komputer yang disediakan mungkin tidak menceritakan teknik perancangan yang baik.

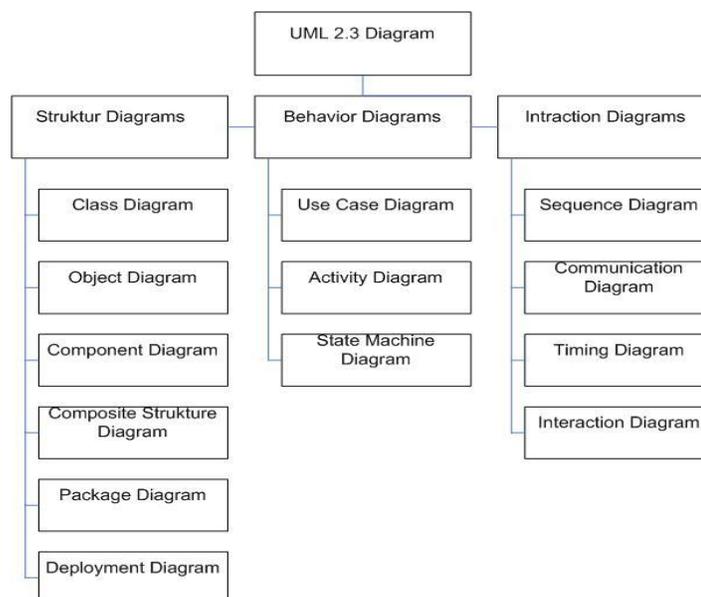
2.6 UML (*Unified Modeling Language*)

Rosa dan Shalahuddin (2013, p.137) *Unified Modelling Language* atau biasa disingkat UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks – teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

Seperti bahasa - bahasa lainnya, UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti perangkat lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk – bentuk tersebut dapat dikombinasikan.

2.7 Diagram – Diagram UML

Rosa dan Shalahuddin (2013, p.140) pada UML 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam – macam diagram tersebut dapat dilihat Gambar 2.2 dibawah ini :



Gambar 2.2 Diagram UML

Berikut ini penjelasan singkat dari pembagian kategori tersebut

- 1) *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
- 2) *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
- 3) *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

2.8 Class Diagram

Rosa dan Shalahuddin (2013, p.141) diagram kelas atau *class diagram* merupakan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- 1) Atribut merupakan variable – variable yang dimiliki oleh suatu kelas
- 2) Operasi atau metode adalah fungsi – fungsi yang dimiliki oleh suatu kelas

Pada kelas dibuat agar pembuat program atau *programmer* membuat kelas – kelas sesuai rancangan di dalam kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron. Kelas – kelas yang ada pada struktur sistem harus dapat melakukan fungsi – fungsi sesuai dengan kebutuhan sistem sehingga pembuat perangkat lunak atau *programmer* dapat membuat kelas – kelas di dalam program perangkat lunak sesuai dengan perancangan diagram kelas. Susunan kelas yang baik pada diagram kelas sebaiknya memiliki jenis – jenis berikut :

- 1) Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan

- 2) Kelas yang menangani tampilan sistem (*view*)

Kelas yang mendefinisikan dan mengatur tampilan ke pengguna.

3) Kelas yang diambil dari pendefinisian *use case* (*Controller*)

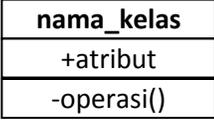
Kelas yang menangani fungsi – fungsi yang harus ada diambil dari pendefinisian *use case*, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

4) Kelas yang diambil dari pendefinisian data (*model*)

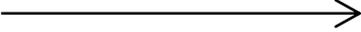
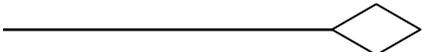
Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data. Semua tabel yang dibuat di basis data dapat dijadikan kelas, namun untuk tabel dari hasil relasi atau atribut multivalued ERD dapat dijadikan kelas tersendiri dapat juga tidak asalkan pengaksesannya dapat dipertanggung jawabkan atau tetap didalam perancangan kelas.

Simbol – simbol yang ada pada diagram kelas (*class diagram*) dapat dilihat pada tabel 2.1 berikut :

Tabel 2.1 Simbol *Class Diagram*

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka/ <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi/ <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>

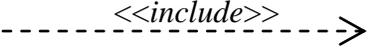
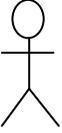
Lanjutan **Tabel 2.1** Simbol *Class Diagram*

Asosiasi berarah/ <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
Kebergantungan/ <i>dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi/ <i>aggregation</i> 	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>)

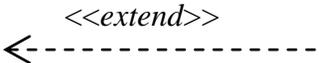
2.9 Use Case Diagram

Rosa dan Shalahuddin (2013, p.155) *use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi satu atau lebih aktor dengan sistem informasi yang akan dibuat. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*. Simbol-simbol yang ada pada diagram *use case* dapat dilihat pada Tabel 2.2 berikut.

Tabel 2.2 Simbol *Use Case*

Simbol	Deskripsi
<p>Menggunakan <i>include</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i> :</p> <ol style="list-style-type: none"> 1) <i>include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, 2) <i>include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan.
<p>Aktor/<i>Actor</i></p>  <p>Nama Aktor</p>	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.</p>

Lanjutan **Tabel 2.2** Simbol *Use Case*

<p><i>Use Case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>.</p>
<p>Asosiasi/<i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.</p>
<p>Generalisasi/<i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya</p>
<p>Ekstensi/<i>extend</i></p> 	<p>Relasi <i>use case</i> tambahan ke <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.</p>

Use case nantinya akan menjadi kelas proses pada diagram kelas sehingga perlu dipertimbangkan penamaan yang dilakukan apakah sudah layak menjadi kelas atau belum sesuai dengan aturan pendefinisian kelas yang baik.

2.10 Activity Diagram

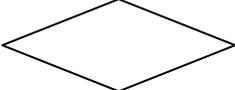
Rosa dan Shalahuddin (2013, p.161) diagram aktivitas atau *activity diagram* menggambarkan *workflow* (alir kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

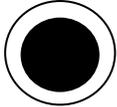
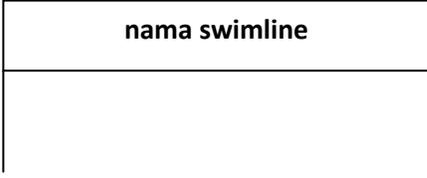
- 1) Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- 2) Urutan atau pengelompokan tampilan dari sistem/*uses interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
- 3) Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
- 4) Rancangan menu yang ditampilkan pada perangkat lunak.

Simbol-simbol yang ada pada diagram aktivitas dapat dilihat pada Tabel 2.3 berikut ini.

Tabel 2.3 Simbol *Activity Diagram*

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah sistem awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.

Lanjutan **Tabel 2.3** Simbol *Activity Diagram*

Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
Swimlane 	Memisahkan organisasi bisnis yang bertanggungjawab terhadap aktivitas yang terjadi.

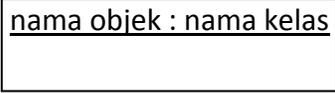
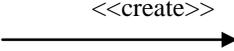
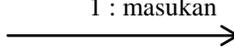
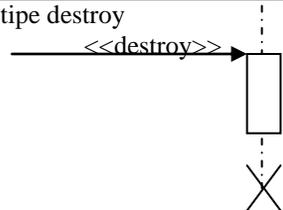
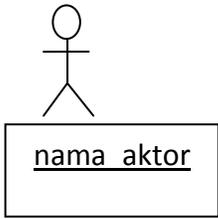
2.11 *Sequence Diagram*

Rosa dan Shalahuddin (2013, p.165) diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat scenario yang ada pada *use case*.

Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.

Simbol-simbol yang ada pada diagram sekuen dapat dilihat pada Tabel 2.4 berikut.

Tabel 2.4 Simbol *Sequence Diagram*

Simbol	Deskripsi
Garis hidup/ <i>lifeline</i> 	Menyatakan kehidupan suatu objek.
Objek 	Menyatakan objek yang berinteraksi pesan.
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
Pesan tipe create 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
Pesan tipe call 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
Pesan tipe send 	Merupakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
Pesan tipe return 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
Pesan tipe destroy 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri sebaliknya jika ada create maka ada destroy.
Aktor atau 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.

Penomoran pesan berdasarkan urutan interaksi pesan. Penggambaran letak pesan harus berurutan, pesan yang lebih atas dari lainnya adalah pesan yang berjalan terlebih dahulu. Semua metode di dalam kelas harus ada di dalam diagram kolaborasi atau sekuen, jika tidak ada berarti perancangan metode di dalam kelas itu kurang baik. Hal ini dikarenakan ada metode yang tidak dapat dipertanggungjawabkan kegunaannya.