

## **BAB 2**

### **TINJAUAN PUSTAKA**

Bagian ini berisi mengenai profil perusahaan beserta teori-teori yang menjadi pendukung dalam proses analisis dan implementasi Pemanfaatan *Augmented Reality* Untuk Pembuatan Katalog Interaktif Pameran Busana Berbasis Android (Studi Kasus AMU Fashion Design).

#### **2.1 Profil AMU Fashion Design**

Berikut ini merupakan profil dari AMU Fashion Design yang menjadi tempat penelitian penulis.

##### **2.1.1 Sejarah AMU Fashion Design**

Berawal dari senang merancang pakaian dan kecintaan *owner* terhadap memadugayakan pakaian tradisional Minangkabau dengan mengikuti *trend* zaman sekarang, menjadi awal berdirinya sebuah usaha di bidang busana dengan nama AMU Fashion Design. AMU Fashion Design berdiri sejak 13 Januari 2019, di prakarsai oleh Atika Mutiarani sekaligus *owner* AMU Fashion Design.

Toko yang beralamatkan di Jalan Raya Batusangkar No.400, Ampang Gadang, Kec. Ampek Angkek, Kabupaten Agam, Sumatera Barat ini menjual dan menerima pesanan berbagai jenis model busana *ready to wear* maupun busana dengan rancangan sesuai minat konsumen. Koleksi busana antara lain Bumi Batik, Millenial Songket, Batik Romanticism, Fatimah Seribu Bukit, In Dignityd dan busana lainnya. Saat ini dalam mengembangkan usaha, AMU Fashion Design sudah banyak mendapatkan penghargaan dari kompetisi pameran busana yang telah di ikuti.

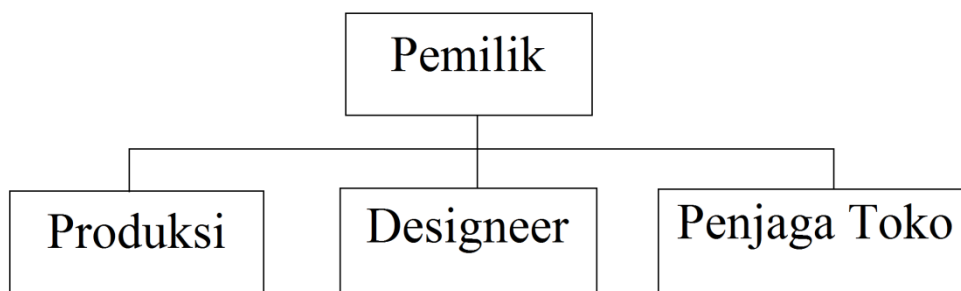
##### **2.1.2 Logo Perusahaan**

Logo Perusahaan AMU Fashion Design diambil dari singkatan nama sang *owner* itu sendiri yaitu Atika Mutiarani. Pada Gambar 2.1 merupakan logo dari AMU Fashion Design.



**Gambar 2.1 Logo AMU Fashion Design**

### 2.1.3 Struktur Organisasi Perusahaan



**Gambar 2.2 Struktur Organisasi**

Adapun uraian pada struktur organisasi pada AMU Fashion Design sebagai berikut :

1. Pemilik  
Merancang dan menyediakan biaya untuk pembelian bahan-bahan yang digunakan untuk memproduksi busana.
2. Penjaga Toko  
Melayani pelanggan dalam proses transaksi.
3. Designer  
Bertanggung jawab atas rancangan yang akan dibuat.
4. Bagian Produksi  
Memproduksi busana sesuai rancangan.

## **2.2 Landasan Teori**

Landasan teori merupakan pengertian dan teori-teori yang akan digunakan untuk dijadikan landasan dalam melakukan penelitian.

### **2.2.1 Promosi**

Promosi ialah suatu cara untuk memberitahukan suatu barang atau produk agar konsumen tertarik dan membeli produk yang ditawarkan. Promosi tidak hanya menginformasikan, tetapi untuk membedakan produk satu dengan yang lain [3].

### **2.2.2 Katalog**

Katalog merupakan media promosi yang berisikan informasi dari benda atau barang yang ditawarkan dalam jumlah secara bersamaan [3].

### **2.2.3 Sistem**

Sistem merupakan beberapa komponen yang memiliki fungsi berbeda dan saling berkerja sama untuk mencapai satu tujuan [4]. Jika didalam sebuah sistem yang tidak dapat memberikan manfaat dalam mencapai suatu tujuan, maka elemen tersebut bukan dari suatu sistem. Berikut adalah beberapa elemen yang membentuk sebuah sistem :

#### **1. Input**

Input merupakan suatu yang masuk ke dalam database dan selanjutnya akan diproses. Input dapat berupa tampak secara fisik maupun tidak tampak.

#### **2. Output**

Output merupakan hasil pemrosesan. Pada sistem informasi, output bisa berupa informasi, laporan cetak dan sebagainya.

#### **3. Proses**

Proses merupakan bagian yang melakukan perubahan dari input menjadi output.

#### **4. Tujuan**

Setiap sistem memiliki tujuan. Tujuan ini mengarahkan sebuah sistem dalam proses kerjanya. Tanpa tujuan, sistem menjadi tak terkendali.

## 5. Mekanisme Pengendalian dan Umpan Balik

Mekanisme pengendalian diwujudkan dengan menggunakan umpan balik, yang mencuplik output. Umpan balik digunakan untuk mengendalikan input maupun output. Tujuannya ialah untuk mengatur sistem berjalan dengan baik sesuai tujuan.

### 2.2.4 Aplikasi

Aplikasi penerapan, penggunaan atau penambahan [5]. Dapat disimpulkan aplikasi merupakan perangkat lunak yang berfungsi untuk menjalankan perintah-perintah dari pengguna dengan tujuan mendapatkan hasil yang akurat.

Menurut Buyens, aplikasi perangkat lunak yang dirancang atau dibuat untuk mempermudah kebutuhan pengguna. Jadi dapat disimpulkan aplikasi merupakan perangkat lunak untuk mempermudah pengguna dalam pekerjaan seperti penambahan data, penerapan. Salah satu contoh aplikasi ialah Web Browser. Klasifikasi aplikasi dapat dibagi menjadi dua yaitu :

1. Aplikasi Paket, dengan dokumentasi yang dirancang untuk masalah tertentu.
2. Aplikasi software spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu.

### 2.2.5 Multimedia

Multimedia merupakan penggunaan komputer untuk menampilkan animasi, suara, teks, gambar, video dan lain sebagainya, sehingga pengguna dengan mudah berkomunikasi dan berinteraksi. Multimedia sering digunakan untuk dunia hiburan seperti *Game*. Menurut Iwan Binanto multimedia dapat dibagi berdasarkan cara menjalankannya [6]. Berikut jenis multimedia yaitu :

1. Multimedia Interaktif

Pengguna dapat mengontrol apa dan kapan konten multimedia akan ditampilkan atau dikirimkan.

2. Multimedia Hiperaktif

Multimedia hiperaktif mempunyai struktur elemen-elemen terkait yang hanya dapat diarah oleh pengguna. Dengan kata lain, multimedia hiperaktif mempunyai banyak tautan yang menghubungkan elemen-elemen multimedia yang ada.

### 3. Multimedia Linear

Dalam penggunaan multimedia linear, pengguna hanya menonton produk multimedia yang disajikan.

#### 2.2.6 Augmented Reality

*Augmented Reality* ialah Suatu Teknologi yang dapat menggabungkan lingkungan tampak dengan dunia virtual yang berjalan secara interaktif dalam waktu nyata. *Augmented Reality* dikembangkan pertama kali pada tahun 1965 di Sutherland dan sampai sekarang masih terus berkembang. Sedangkan tujuan *Augmented Reality* adalah menyederhanakan objek nyata dengan membawa objek maya sehingga informasi tidak hanya untuk pengguna secara langsung tetapi juga untuk setiap pengguna yang tidak langsung terhubung dengan *user interface* dari objek nyata [1].

Teknologi *Augmented Reality* ada beberapa jenis, yaitu :

##### 1. Marker Based Tracking Augmented Reality

Sebuah metode yang menggunakan *marker* berupa ilustrasi hitam dan putih berbentuk persegi atau lainnya. *Marker Based Tracking* pertama kali dikembangkan sejak tahun 1980. Cara kerja *Marker Based Tracking* ialah menggunakan kamera yang diarahkan ke *marker*, kemudian akan menampilkan objek virtual 2D atau 3D.

##### 2. Markeless Augmented Reality

*Markeless Augmented Reality* kebalikan dari *Marker Based Tracking*. Metode ini tidak perlu menggunakan *marker* unik sebagai acuan untuk menampilkan objek 2D atau 3D. Sekalipun masih dinamakan *markeless*, tetapi aplikasi berkerja dengan melakukan pemindaian ke objek, namun ruang lingkup *markeless* lebih jauh dari pada metode *Marker Based Tracking*. Teknik *Markeless*

*Augmented Reality* memiliki teknik seperti *Face Tracking*, *3D Object Tracking*, *Image Target* dan *Motion Tracking*.

a. *Face Tracking*

Menggunakan berbagai jenis algoritma yang telah dikembangkan, komputer dapat mengenali bentuk wajah manusia secara umum dengan cara mengenali struktur wajah, hidung, mata dan mulut. Objek-objek lain seperti pohon, kursi, meja di sekitar akan diabaikan.

b. *3D Object Tracking*

Berbeda dengan *Face Tracking* yang hanya mengenali bentuk struktur wajah, metode *3D Object Tracking* dapat mengenali semua bentuk benda yang ada disekitar, seperti pohon, kursi, mobil dan lain-lain.

c. *Image Target*

*Image Target* adalah sebuah metode yang akan dibaca oleh sistem yaitu gambar seperti gambar berwarna maupun gambar hitam putih.

d. *Motion Tracking*

*Motion Tracking* ialah teknik yang dapat menangkap gerakan.

3. *Projection Based*

*Projection Based* yaitu teknik yang akan memproyeksikan cahaya sintetis ke permukaan fisik, dan dalam beberapa kasus memungkinkan untuk berinteraksi dengannya.

Penggunaan *Augmented Reality* sudah banyak merambah diberbagai bidang. Bidang-Bidang yang suda menerapkan teknologi *Augmented Reality* :

1. *Industri*

Dalam dunia industri saat ini, penggunaan AR sangat efektif dalam memperkenalkan suatu trobosan atau produk, sehingga konsumen akan lebih tertarik akan nilai produk yang ditampilkan. Salah satu contohnya dalam industri properti sekarang yang sudah menggunakan teknologi AR untuk menampilkan prabotan rumah tangga seperti kursi, meja, lemari dan lain sebagainya.

2. *Dunia Medis*

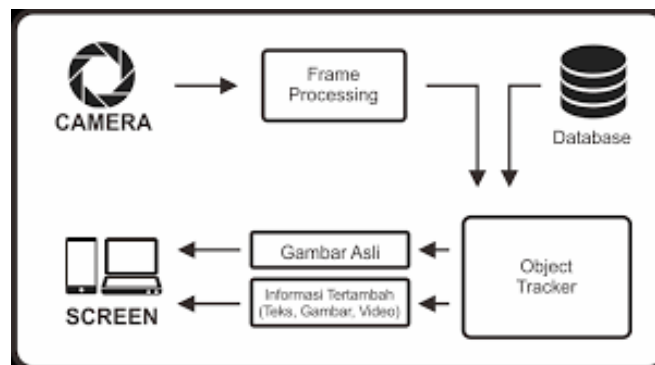
Dalam dunia medis saat ini juga sudah menggunakan teknologi AR sebagai pendukung, seperti simulasi operasi. Dengan menerapkan teknologi AR, akan membantu dokter menganalisis sebelum melakukan operasi.

### 3. Pelatihan Pilot

Saat ini berbagai maskapai penerbangan sudah menerapkan teknologi AR untuk pelatihan penerbangan. Sebagai contoh, maskapai menggunakan AR untuk membuat *game* simulasi penerbangan untuk melatih pilot-pilot muda yang siap untuk langsung memulai menerbangkan pesawat secara nyata.

#### 2.2.7 Cara Kerja Augmented Reality

Secara umum cara kerja *Augmented Reality* dimulai dari pengambilan gambar *marker* dengan webcam atau kamera ponsel. *Marker* tersebut dapat dikenali berdasarkan fitur yang dimiliki dan telah terdaftar didalam database, kemudian masuk ke dalam objek yang akan dilacak oleh *Software Development Kit (SDK)*. *Object Tracker* akan melacak dan mencocokkan *marker* tersebut agar dapat menampilkan objek ataupun informasi yang telah dimasukkan. Hasil *output* dari pelacakan *marker* segera ditampilkan ke layar komputer atau layar ponsel.



**Gambar 2.3 Alur kerja Augmented Reality**

1. Kamera  
Digunakan untuk menscan *marker*.
2. Frame Processing  
Proses perhitungan pixel yang digunakan untuk pendeteksian objek.
3. Database  
Sebagai tempat menyimpan data *marker*.

#### 4. Object Tracker

Objek yang dilacak akan ditampilkan ke layar berdasarkan data marker yang ada didalam database.

#### 5. Informasi

Menampilkan informasi berdasarkan dengan *object tracker* yang sudah dikenali didatabse.

#### 6. Screen

Menampilkan ke layar komputer berupa teks, gambar, vidio, objek 2D atau 3D.

### 2.2.8 Vuforia

*Augmented Reality* memiliki library bernama Vuforia yang dikembangkan oleh Qualcomm, menggunakan sumber yang konsisten mengenai computer vision yang fokus pada image recognition. Vuforia banyak memiliki fitur-fitur yang dapat membantu developer untuk mewujudkan ide kreatif mereka tanpa adanya batasan.

Dengan *support* untuk sistem operasi Windows, Android, iOS dan Unity, Vuforia mendukung para developer dalam mengembangkan aplikasi yang bisa digunakan diberbagai jenis ponsel dan *tablet*. Developer diberikan kebebasan untuk merancang aplikasi yang mempunyai kemampuan antara lain :

1. Teknologi computer vision tingkat tinggi yang mengijinkan developer untuk membuat elemen-elemen khusus pada device.
2. Tracking dan detection tingkat lanjut.
3. Pengaturan database gambar yang fleksibel.
4. Dapat mengenali multiple image secara terus-terusan.

Target pada vuforia ialah objek pada dunia nyata yang dapat dideteksi oleh webcam atau kamera, kemudian menampilkan objek virtual. Beberapa jenis targer pada vuforia adalah :

1. *Image Targets*, contohnya sampul buku, poster, papan permainan dan lainnya sebagainya. Jenis target ini menampilkan objek sederhana dari Augmented.



2. *Multi Target*, contohnya kemasan produk yang memiliki sudut atau persegi. Jenis targer ini menampilkan objek sederhana Augmented.
3. *Virtual Buttons*, dapat membuat tombol sebagai sasaran gambar, teks, dan lainnya sebagainya.
4. *Frame Markers*, tipe frame gambar 2D dengan pattern khusus yang dapat digunakan sebagai potongan permainan pada papan.

### 2.2.9 Arsitektur Vuforia

Vuforia memerlukan beberapa komponen penting agar dapat berjalan dengan baik. Komponen tersebut sebagai berikut :

1. Kamera
 

Kamera atau webcam dibutuhkan untuk memastikan bahwa setiap frame ditangkap dan diteruskan secara efisien ke *tracker*. Developer tinggal memberi perintah kapan kamera mulai menangkap dan berhenti.
2. *Image Converter*

Mengkonfersi format kamera seperti YUI12 ke format yang dapat dideteksi oleh Open GL untuk tracking.
3. *Tracker*

Memakai algoritma *computer vision* untuk melakukan pelacakan objek nyata yang diambil dari kamera. Objek tersebut dipertimbangkan dan hasilnya akan disimpan dalam *state object* yang akan dipakai oleh *video background renderer* dan dapat diakses dari *application code*.
4. *Target Resource*

Menggunakan *Target Management System*. Assets di unduh berisi sebuah konfigurasi xml `config.xml` yang memungkinkan developer untuk mengkonfigurasi beberapa fitur dalam trackable dan binary file yang berisi database trackable.
5. *Application Code*

Selama objek yang diinginkan untuk diubah prosesnya, maka *application code* harus diubah berdasarkan lokasi objek virtual.
6. *Vidio Background Render*

*Render* gambar dari kamera yang tersimpan di dalam *state object*. Kecepatan dari *render* sangat tergantung pada device yang digunakan.

### **2.2.10 Unity**

Unity merupakan *software development* yang terintegrasi untuk menciptakan video *game* atau konten yang memiliki visualisasi arsitektural 3D. Unity dapat dijalankan pada sistem operasi Windows, Linux maupun macOS. Permainan yang dihasilkan dapat dijalankan Xbox 360, PlayStation, Nintendo Wii, iPad, Android, iOS dan lainnya sebagai. Unity juga dapat menghasilkan permainan untuk web browser dengan menggunakan plugin Unity Web Player.

### **2.2.11 Blender**

Studio animasi Belanda Neo Geo mengembangkan aplikasi in-house yang diberi nama Blender. Hal ini terutama ditulis oleh Ton Roosendaal, sebelumnya menulis sebuah pelacak sinar disebut Jejak untuk Amiga pada tahun 1989. Nama Blender terinspirasi oleh sebuah lagu Yello dari album bayi.

Blender ialah sebuah *software* design grafis open-source yang digunakan untuk membuat aplikasi 3D, efek visual, animasi atau video game. Blender memiliki fitur pemodelan 3D, *unwrapping UV*, *texturing*, *rigging*, pelacakan kamera, *rendering*, video editing dan *compositing*.

### **2.2.12 Android**

Android merupakan sebuah sistem operasi untuk perangkat mobile berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* yang terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Android merupakan generasi baru *platform mobile*, *platform* yang memberikan pengembang untuk melakukan pengembangan sesuai dengan yang diharapkannya.

Sistem operasi yang mendasari Android dilisensikan dibawah GNU, *General Public Lisence* Versi 2 (GPLv2), yang sering dikenal dengan istilah “copyleft” lisensi dimana setiap perbaikan pihak ketiga harus terus jatuh di bawah terms. Android didistribusikan di bawah Lisensi *Apache Software*

(ASL/Apache2), yang memungkinkan untuk distribusi kedua dan seterusnya. Komersialisasi pengembang (produsen handset khususnya) dapat memilih untuk meningkatkan *platform* tanpa harus memberikan perbaikan mereka ke masyarakat *open source*.

Sebaliknya, pengembang dapat keuntungan dari perangkat tambahan seperti perbaikan dan mendistribusikan ulang pekerjaan mereka di bawah lisensi apapun yang mereka inginkan. Pengembang aplikasi Android diperbolehkan untuk mendistribusikan aplikasi mereka di bawah skema lisensi apapun yang mereka inginkan [7].

### **2.2.13 Java Development Kit (JDK)**

Java Development Kit (JDK) adalah *software development* kit merupakan library, form dan paket yang berguna untuk membuat aplikasi. Didalam JDK terdapat Java Runtime Environment (JRE) yang akan memproses *byte code* yang dihasilkan dari *interpreter java*. *Interpreter java* bertugas sebagai compile *source.java* menjadi *source.class* yang selanjutnya *source.class* inilah yang akan dijalankan JRE.

### **2.2.14 Software Development Kit (SDK)**

Software Development Kit (SDK) merupakan kumpulan dari *library* dan *tool kit* seperti emulator untuk mengembangkan aplikasi untuk suatu perangkat lunak, perangkat komputer dan sistem operasi. SDK terdapat *tools* yang dibutuhkan dalam pengembangan android, seperti adb shell yang merupakan singkatan dari *Android Development Bridge* yang dapat menjalankan terminal android seperti terminal pada sistem operasi linux dan command yang terdapat pada adb shell sendiri seperti command linux pada umumnya.

### **2.2.15 Pemrograman Berorientasi Objek**

Pemrograman Berorientasi Objek (PBO) merupakan konsep pemrograman untuk pengembangan suatu perangkat lunak, dimana dalam struktur perangkat lunak berdasarkan kepada interaksi objek dalam penyelesaian suatu proses [8].

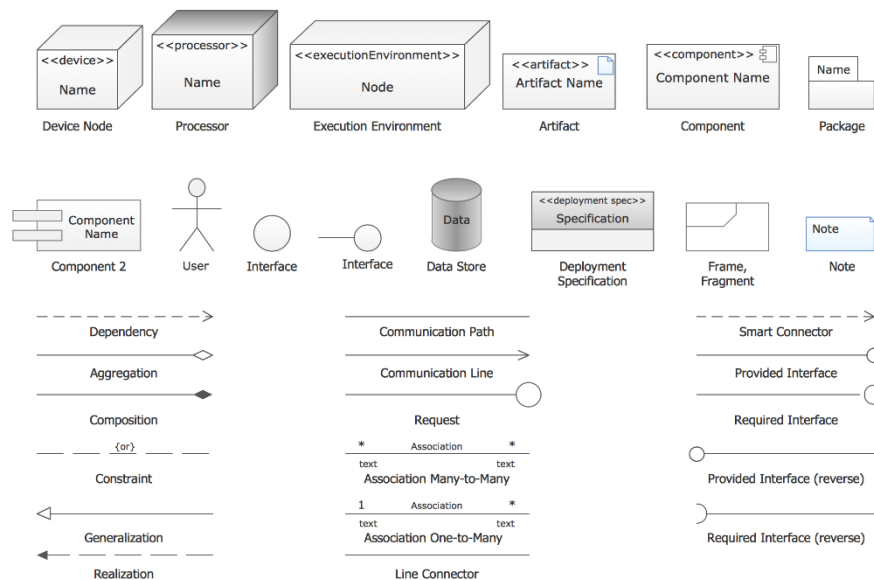
Setiap komponen dalam sistem tersebut dapat mewarisi atribut dan sifat komponen lainnya, serta saling berinteraksi satu sama lain.

Metode berorientasi objek didasarkan pada penerapan prinsip-prinsip pengolahan kompleksitas. Metode ini meliputi rangkaian aktivitas, analisis berorientasi objek, perancangan berorientasi objek, pemrograman berorientasi objek, dan pengujian berorientasi objek.

Analisis dan desain berorientasi objek adalah suatu cara dalam memecahkan masalah dengan model yang dibuat menurut konsep. Model berorientasi objek bermanfaat untuk memahami suatu masalah, komunikasi dengan pengembang aplikasi, dokumentasi serta perancangan program.

### 2.2.16 Unified Modeling Language (UML)

Unified Modelling Language (UML) adalah metode permodelan secara visual untuk sarana perancangan sistem berorientasi objek, atau definisi UML yaitu sebagai suatu bahasa yang sudah menjadi standar pada visualisasi, perancangan dan juga pendokumentasian sistem *software* [9].









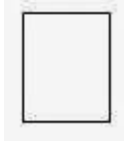

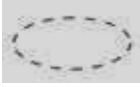
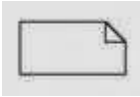
**Gambar 2.4 UML Notations**

### 2.2.17 Usecase Diagram

*Usecase* merupakan metode yang digunakan dalam pengembangan sebuah perangkat lunak untuk menangkap kebutuhan fungsional dari sistem yang bersangkutan. *Use Case* mendeskripsikan interaksi yang terjadi antara satu atau lebih aktor dengan sistem yang ada, sebuah *Use Case* direpresentasikan dengan urutan langkah yang sederhana.




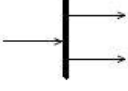
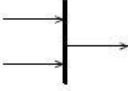

**Tabel 2.1 Usecase Notations**

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
3		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

### 2.2.18 Activity Diagram

*Activity diagram* adalah rangkaian aliran kerja dari sebuah sistem atau proses bisnis. Digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case*. Perlu diperhatikan adalah bahwa *Activity Diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan aktor.

Simbol	Keterangan
	Start Point
	End Point
	Activities
	Fork (Percabangan)
	Join (Penggabungan)
	Decision
Swimlane	Sebuah cara untuk mengelompokkan activity berdasarkan Actor (mengelompokkan activity dalam sebuah urutan yang sama)



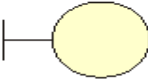



**Gambar 2.5 Activity Diagram**

### 2.2.19 Class Diagram

*Class diagram* merupakan diagram yang menggambarkan struktur *system* dengan menjelaskan keterkaitan antara suatu kelas dengan kelas yang lain yang terdapat pada sistem tersebut. Kelas memiliki 3 bagian utama yaitu *attribute*, *operation*, dan *name*. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem [12].

### 2.2.20 Sequence Diagram

*Sequence diagram* merupakan diagram yang menggambarkan interaksi objek pada *usecase* dengan menjelaskan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Banyaknya *Sequence diagram* yang harus digambar sebanyak *usecase* yang telah didefinisikan interaksinya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *usecase* yang ada maka diagram sekuen yang dibuat semakin banyak [12].

- a. An Actor  
 Menggambarkan orang yang sedang berinteraksi dengan sistem
- b. Entity Class  
 Menggambarkan hubungan kegiatan yang akan dilakukan
- c. Boundary Class  
 Menggambarkan sebuah penggambaran dari form
- d. Control Class  
 Menggambarkan penghubung antara boundary dengan tabel
- e. A focus Of Control & A life line  
 Menggambarkan tempat mulai dan berakhirnya sebuah message
- f. A message  
 Menggambarkan Pengiriman Pesan

**Gambar 2.6 Sequence Diagram**