

BAB II

TINJAUAN PUSTAKA

II.1 Penelitian Terdahulu

Untuk mendukung keberhasilan penelitian ini, penyusun melakukan pendekatan teoritis melalui beberapa literatur yang berhubungan dengan penelitian yang dilakukan. Beberapa uraian penelitian terdahulu yang menjadi acuan dalam penelitian ini yaitu :

Pada penelitian yang dilakukan oleh I Gede Teguh Pribadi dkk, (2017) yang berjudul **“Rancang Bangun Game Tower Defense ‘Defense of Dewata Island’ Berbasis Android”**. Pada penelitian ini *game offline* merupakan suatu metode *single player* yang dikembangkan menggunakan *Corona SDK* berbasis platform Android. *Game Tower Defense* ini dibuat menggunakan *background* dengan konsep budaya Bali serta konsep cerita yang diangkat dari Perang Puputan Badung. *Game Tower Defense* ini telah tergolong baik dari aspek pengalaman *user*, *grafis game*, rekayasa perangkat lunak, entertainment serta *content* dan mampu menyampaikan sejarah perjuangan budaya Bali melalui cerita yang telah diangkat.

Pada penelitian yang dilakukan oleh Doni Indra Sazaly Saragih dkk (2018) yang berjudul **“Rancang Bangun Sistem E-Sport Cybercafe Finder Menggunakan Metode Lbs (*Location Based Service*) Berbasis Android”**. Pada penelitian ini berhasil mempertemukan lokasi antara pengguna/pencari dan *cybercafe* menggunakan android studio, mempermudah bagi para gamers untuk

mencari lokasi *cybercafe* berkat adanya *library map* dari *Google API* sehingga mendapatkan rute terbaik. Integrasi pemrograman *Java* dan *XML* serta *SQLite database* berhasil diimplementasikan pada aplikasi ini dengan lancar.

Pada penelitian yang dilakukan oleh Karllys Samuel Marpaung (2016) dengan judul **“Perancangan Game “Find A way Out Of This Maze!” Berbasis Android di Evo Gaming”**. Penelitian ini menghasilkan aplikasi sistem pembangkit labirin dinamis dimana model pengembangan *GDLC* sangat cocok dalam membantu membuat sebuah *game* dan mengimplementasikan algoritma *Backtraking* pada *game treasure of labyrinth* sebagai *labyrinth generator*. *game treasure of labyrinth* yang dibangun dimainkan di sistem operasi android dan dapat dimainkan pada android minimal versi android *kitkat* dimana *game* ini dapat dimainkan oleh *single player*.

Pada penelitian yang dilakukan oleh Angling Ananditya et all (2020) dengan judul **“Perancangan Sistem Informasi Penjualan Voucher Game Online Berbasis Dekstop Pada Aren.Net Di Depok”**. Pada penelitian ini proses transaksi penjualan *voucher* menjadi efektif, karena di sistem ini sudah otomatis tersedia data *voucher* dan stok *voucher*. Proses transaksi sudah terkomputerisasi, sehingga dapat mempermudah dan mempercepat waktu pekerjaan dalam pengolahan data penjualan *voucher*. Antarmuka yang sederhana dapat mempermudah *user* dalam mempelajari sistem ini.

Pada penelitian yang dilakukan oleh Mukhammad Agus Tri Rakhmawan et al (2017) dengan judul “**Rekayasa Sistem Transaksi Fitur Game Online Dengan Pembayaran Pulsa Berbasis CMS Opencart**”. Pada penelitian ini analisis dan perancangan sistem simulasi website *e-commerce* terdiri dari dua *user* utama, yaitu administrator dan customer. Dimana Administrator memiliki wewenang untuk mengolah website, database website, proses pemesanan item game, proses tradeitem pada game, serta proses penarikan tunai. Sedangkan untuk customer, akan memiliki beberapa fasilitas bila menjadi member, yaitu dapat mengetahui *history* transaksi, keamanan transaksi yang aman, serta kemudahan pembayaran. *Websitee-commerce* telah dirancang dan dibuat dengan *CMS OpenCart* yang telah terintegrasi dengan database. Website *e-commerce* ini telah memiliki fasilitas pembayaran melalui *TCash*. *TCash* memiliki keunggulan fasilitas tarik tunai dari saldo pulsa. Sehingga administrator tidak perlu lagi kesulitan dalam penarikan tunai.

II.2 Landasan Teori

II.2.1 Industri Game Daring

Permainan daring adalah komoditas baru yang mulai populer di dunia semenjak teknologi informasi dan internet berkembang. Industri permainan daring adalah perpanjangan dari industri game offline. Permainan daring dapat didefinisikan sebagai program permainan digital yang memanfaatkan jaringan untuk menghubungkan satu perangkat dengan perangkat yang lainnya. Jaringan

tersebut dapat memanfaatkan jaringan LAN (*Local Area Network*) atau jaringan internet.

Pengembangan permainan daring tidak terbendung dari waktu ke waktu. Perkembangan tersebut menghasilkan jenis - jenis permainan daring yang berbeda karakteristik dan cara memainkannya. (Moh. Adli Ahdiyat. 2018. Jurnal Ilmu Komunikasi. 7(2):105-115.

Jenis-jenis bentuk permainan daring yang populer saat ini adalah sebagai berikut:

1. *Massively Multiplayer Online* (MMO) Jenis permainan daring ini dimainkan melalui jaringan LAN (*Local Area Network*) atau melalui jaringan internet. Pemain menggunakan jaringan tersebut dan berinteraksi dengan pemain lain di dalam dunia virtual. Pemain bahkan dapat bermain dengan ratusan orang di penjuru dunia dalam satu waktu. Contoh dari permainan daring populer dari jenis ini adalah *Player Unknown*
2. *Real-Time Strategy* Jenis game ini adalah *game* yang membutuhkan keahlian dalam menyusun strategi di mana pemain harus membangun sebuah pasukan, bangunan, dan inventaris. Game Jenis ini sering diperuntukkan untuk game bergenre peperangan. Game ini juga dimainkan oleh beberapa orang secara real-time atau dalam waktu yang sama. Contoh populer dari game jenis ini adalah *Age of Empire*.
3. *Role-Playing Games* (RPG) Dalam jenis permainan daring ini, pemain game memainkan sebuah peran dalam sebuah cerita. Pemain dapat menentukan alur cerita yang diinginkannya. Disamping itu, terdapat

petunjuk naratif yang disiapkan oleh program game. Game jenis ini seringkali bergenre fantasy. Contoh game berjenis *RPG* yang populer adalah *Final Fantasy*, *Pokemon*, *Digimon Online*, dan *Ragnarok Online*.

4. *Multiplayer Online Battle Arena* (MOBA). *MOBA* adalah jenis permainan daring yang merupakan sub-genre dari jenis game *RTS*. Pada umumnya *MOBA* adalah game yang dimainkan oleh sepuluh orang yang dibagi menjadi dua tim (5 versus 5). Berbeda dengan *RTS*, di dalam *MOBA* tidak ada penyusunan strategi dengan membangun bangunan dan pasukan. Strategi lebih difokuskan kepada pengembangan skil karakter dan strategi dalam pertempuran antar tim di dalam arena. Contoh populer dari permainan daring jenis ini adalah *DOTA 2*, *Mobile Legends Bang Bang* (*MLBB*), *Vain Glory*, *League of Legends*, dan *Arena of Valor* (*AoV*).
5. *First-Person Shooting* (FPS). *FPS* adalah jenis permainan daring di mana tampilan game adalah gambaran dari penglihatan mata para pemain game. Jenis permainan daring ini sangat populer dikalangan para pecinta game dengan genre *shooting*. *Doom* adalah *game* dengan jenis *FPS* yang pertama kali dirilis pada tahun 1994. Contoh lain dari permainan daring *FPS* adalah *Counter Strike* dan *Point Blank*.

II.3. Pembelian Dalam Bentuk Online

Pembelian *online* adalah suatu bentuk perdagangan menggunakan perangkat elektronik yang memungkinkan konsumen untuk membeli barang atau jasa dari penjual melalui internet. Nama lain kegiatan tersebut adalah: *e-web-shop*, *e-shop*, *e-toko*, toko internet, *web-shop*, *web-store*, toko *online* dan toko virtual. Saat ini belanja *online* sudah semakin canggih dengan adanya perdagangan via ponsel (*m-commerce*). Telepon seluler telah dioptimalkan dengan sebuah aplikasi untuk membeli dari situs *online*.

Kesuksesan sebuah ritel tidak lagi semua tentang fisik bangunan, ini terbukti dengan peningkatan pengecer yang menawarkan antarmuka toko *online* bagi konsumen. Dengan pertumbuhan belanja *online*, banyak peluang pasar baru untuk toko memenuhi permintaan pasar luar negeri dengan persyaratan layanan tertentu. Contoh perusahaan ritel online terbesar adalah *Alibaba*, *Amazon.com*, *eBay*, *GrantonWorld*. Pembeli *online* umumnya menggunakan kartu kredit atau *PayPal* untuk melakukan pembayaran. Namun, beberapa sistem memungkinkan pengguna untuk membuat *account* dan membayar dengan cara alternatif, seperti:

- Penagihan ke ponsel dan sambungan telepon rumah
- *Cash on delivery* (COD)
- Kartu Debit
- Debit langsung di beberapa negara
- Uang elektronik dari berbagai jenis
- Wesel pos

- Wire transfer / pengiriman pembayaran
- *Bitcoin* atau lainnya

II.4. Penjualan Berbasis Online

Penjualan *online* adalah melakukan aktifitas penjualan dari mencari calon pembeli sampai menawarkan produk atau barang dengan memanfaatkan jaringan internet yang didukung dengan seperangkat alat elektronik sebagai penghubung dengan jaringan internet. Berikut uraian dari definisi atau pengertian penjualan *online* diatas.

a. Melakukan aktifitas penjualan

Yaitu menjalankan semua usaha penjualan, Dari mencari calon pembeli sampai dengan menawarkan barang kepada calon pembeli.

b. Mencari calon pembeli

Usaha ini bisa dilakukan dengan cara iklan atau promosi di jaringan internet. Untuk iklan di internet bisa dipilih yang berbayar atau yang gratis. Yaitu dengan memasang materi iklan atau iklan kita pada jaringan internet. Karena saat ini sudah banyak situs atau web yang menyediakan fasilitas iklan atau promosi gratis dengan jangkauan yang luas. Dan untuk hal ini bisa dilakukan dengan sangat mudah.

c. Meawarkan produk atau barang

Hal ini bisa dilakukan oleh penjual dengan cara memasang katalog produk atau barang yang disertai keterangan pendukungnya. Untuk hal ini

diperlukan adanya katalog online yang baik agar penjualan *online* bisa memberikan hasil yang optimal.

d. Menggunakan jaringan internet

Kata internet atau jaringan internet mungkin sudah tidak asing lagi ditelinga kita, Karena saat ini jaringan internet sudah tersebar hampir diseluruh daerah. Dan jaringan internet ini sekarang sudah digunakan oleh penjual untuk melakukan penjualan secara online. Kata *online* berarti terhubung dengan jaringan internet. Dan dengan jaringan internet ini, Penjual bisa melakukan aktifitas mencari calon pembeli sampai pada penawaran produk atau barang.

Penjualan *online* bisa dilakukan dengan banyak cara. Diantaranya bisa menggunakan web, Web *blog* atau dengan *facebook*. Namun untuk itu penjual harus membuat atau mempunyai katalog produk *online* yang bisa menjual, karena pembeli akan memilih dan menentukan keputusan membeli atau tidak berdasarkan katalog produk yang ada pada jaringan internet.

II.5. *Game E-Sport*

E-Sports merupakan singkatan dari *electronic sports*, yaitu olahraga yang menggunakan *game* yang tidak mengharuskan seseorang untuk melakukan aktivitas fisik ataupun merasa lelah karena melakukan kegiatan yang cukup berat. Dalam perkembangannya *E-Sports* mengalami pertumbuhan yang sangat pesat. *E-Sports* bahkan kini menjadi salah satu sumber penghasilan bagi sebagian orang, khususnya bagi para *gamers* yang memang menekuni dunia game online. Mereka

dapat menjadi atlet kemudian mengikuti berbagai perlombaan dan turnamen *E-Sports*, sehingga apabila menang atau berhasil meraih pencapaian tertentu, mereka dapat menghasilkan uang dan bonus tambahan dari kegiatan tersebut. Namun, dibalik itu semua ternyata masih terdapat pandangan yang kurang baik dari berbagai pihak mengenai keberadaan *E-Sports*. Mereka berpandangan bahwa *E-Sports* menyebabkan para pemainnya tidak memperdulikan lingkungan sekitar sehingga terkesan apatis. Bahkan masih banyak yang menganggap kegiatan tersebut hanya untuk kesenangan dan tidak bermanfaat. Padahal sesungguhnya tidak sepenuhnya begitu. Maka dari itu perlu adanya pemahaman lebih lanjut mengenai perkembangan *E-Sports* baik di dunia maupun di Indonesia.

II.6. Pengertian Aplikasi

Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas seperti sistem perniagaan, game, pelayanan masyarakat, periklanan dan hampir semua proses kegiatan. Aplikasi digunakan dalam suatu komputer, instruksi atau pernyataan yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output. Media pembelajaran adalah segala sesuatu yang dapat menyampaikan dan menyalurkan pesan dari sumber secara terencana sehingga tercipta lingkungan belajar yang kondusif dimana penerimanya dapat melakukan proses belajar secara efektif dan efisien. (Nugraha : 2016). Jurnal Manajemen dan Teknik Informatika. 2(1):191-192.

II.7. Android Studio

Android studio merupakan perangkat lunak yang digunakan untuk perangkat mobile (*smartphone*) atau pun perangkat tablet (PDA). Sifat *platform* android yang terbuka bagi para pengembang untuk mengembangkan aplikasi buatan sendiri membuat android menjadi sistem operasi mobile yang populer hingga saat ini. Secara garis besar android memiliki beberapa arsitektur didalamnya diantaranya yaitu:

1. Applications dan Widgets

Applications and Widgets ini adalah layer yang membatasi penggunaannya hanya untuk berhubungan dengan aplikasi saja, seperti halnya saat pengguna mengunduh aplikasi kemudian melakukan instalasi dan menjalankan aplikasi tersebut. Pada layer terdapat aplikasi ini termasuk klien email, program *SMS*, kalender, peta, *browser* atau kontak. Semua aplikasi ditulis dengan menggunakan bahasa pemrograman *Java*.

2. Applications Frameworks

Application Framework adalah layer untuk melakukan pembuatan aplikasi yang akan dijalankan di sistem operasi Android, karena pada layer inilah aplikasi dapat dirancang dan dibuat, seperti *content provider* yang berupa *SMS (Short Message Service)* dan panggilan telepon.

3. Libraries

Libraries ini adalah layer dimana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya. Berjalan di atas kernel *Linux*, layer ini meliputi berbagai *library C/C++* inti seperti *Libc* dan *SSL*, serta:

- a. *libraries* media untuk pemutaran media audio dan video.
- b. *libraries* untuk manajemen tampilan.
- c. *libraries graphics* mencakup *SGL* dan *OpenSGL* untuk grafis
- d. *libraries SQLite* untuk dukungan *database*.

4. Android Run Time

Android Run Time merupakan layer yang membuat aplikasi Android dapat dijalankan yang dalam prosesnya menggunakan implementasi *Linux*. *Android Run Time* dibagi menjadi dua bagian, yaitu.

- a. *Core Libraries*, yaitu aplikasi Android dibangun dalam bahasa *Java*, sementara Dalvik sebagai mesin virtualnya bukan *Virtual Machine Java*, sehingga diperlukan sebuah *libraries* yang berfungsi untuk menerjemahkan bahasa *Java/ C* yang ditangani oleh *Core Libraries*.
- b. *Dalvik Virtual Machine*, yaitu mesin virtual berbasis *register* yang dioptimalkan untuk menjalankan fungsi-fungsi secara efisien, yang mampu membuat *Linux kernel* untuk melakukan *threading* dan manajemen tingkat rendah.

5. Linux Kernel

Linux Kernel adalah layer dimana inti dari operating sistem dari Android itu berada. Berisi *file* sistem yang mengatur sistem *processing*, *memory*, *resource*, *drivers*, dan sistem-sistem operasi Android lainnya.

II.8. MYSQL

MySQL (*My Structure Query Language*) merupakan sebuah program pembuat database yang bersifat *open source*, artinya semua orang dapat menggunakannya dan dapat dijalankan pada semua platform baik *windows* maupun *linux*. *MySQL* juga merupakan sebuah perangkat lunak sistem manajemen basis data *SQL* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi multi user. *MySQL* juga sering dikenal dengan nama sistem manajemen database relasional. Suatu database relasional menyimpan data dalam table yang terpisah. Tabel – table tersebut terhubung oleh suatu relasi terdefinisi yang memungkinkan user memperoleh kombinasi data dari beberapa table dalam suatu permintaan. Untuk administrasi database, seperti pembuatan database, pembuatan tabel, dan sebagainya dapat digunakan aplikasi berbasis *web* seperti *PhpMyAdmin* dengan aplikasi *XAMPP*. (Saipul Anwar, dkk. 2017). *Pilar Nusa Mandiri*. 13(1): 113-114.

II.9. Pengertian Basis Data

Basis data atau Database adalah kumpulan informasi yang disusun dan merupakan suatu kesatuan yang utuh yang disimpan di dalam perangkat keras (komputer) secara sistematis sehingga dapat diolah menggunakan perangkat lunak. Dengan sistem tersebut data yang terhimpun dalam suatu database dapat menghasilkan informasi yang berguna. (Ganda Yoga Swara dkk, 2016). Jurnal TEKNOIF. 4(2): 27-30.

Jenis operasi dasar Basis data terdiri dari beberapa bagian dimana diantaranya yaitu:

1. Pembuatan basis data baru (*create database*), yang identik dengan pembuatan lemari arsip yang baru
2. Penghapusan basis data (*drop database*), yang identik dengan perusakan lemari arsip (sekaligus beserta isinya jika ada).
3. Pembuatan file/tabel baru ke suatu basis data (*create table*), yang identik dengan penambahan map arsip baru ke sebuah lemari arsip yang telah ada.
4. Penghapusan file/tabel dari suatu basis data (*drop table*), yang identik dengan perusakan map arsip lama yang ada di sebuah lemari arsip.
5. Penambahan/pengisian data baru ke sebuah file/tabel di sebuah basis data (*insert*), yang identik dengan penambahan lembaran arsip ke sebuah map arsip.
6. Pengubahan data dari sebuah file/tabel (*update*), yang identik dengan perbaikan isi lembaran arsip yang ada di sebuah map arsip.

7. Penghapusan data dari sebuah file/tabel (*delete*), yang identik dengan penghapusan sebuah lembaran arsip yang ada di sebuah map arsip.

II.10. UML (*Unified Modelling Language*)

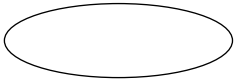
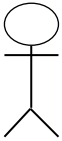

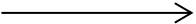
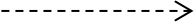
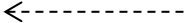
Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. *Unified Modeling Language* (UML) adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan *software* berbasis OO (*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software*. (Suendri:2018). Jurnal Ilmu Komputer dan Informatika. 3(1): 1-2.

II.10.1 Use Case Diagram

Use Case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan *Use Case* digunakan untuk mengetahui fungsi apa saja yang ada di

dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *Use Case* diagram yaitu :

Tabel II.1 Simbol *Use Case* Diagram




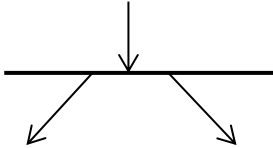
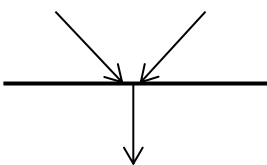
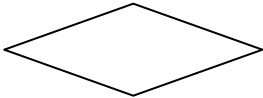

Gambar	Keterangan
	<i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>Use Case</i> .
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>Use Case</i> , tetapi tidak memiliki <i>control</i> terhadap <i>Use Case</i> .
	Asosiasi antara aktor dan <i>Use Case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.
	Asosiasi antara aktor dan <i>Use Case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam <i>Use Case</i> lain (<i>required</i>) atau pemanggilan <i>Use Case</i> oleh <i>Use Case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>Use Case</i> lain jika kondisi atau syarat terpenuhi.

(Sumber : Kusnita Yusmiarti : 2016 : 3

II.10.2 Diagram Aktivitas

Activity diagram menggambarkan *Workflow* (Aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *Activity diagram*, yaitu :

Tabel II.2 Simbol Diagram Aktivitas

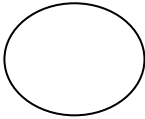
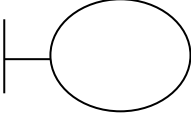
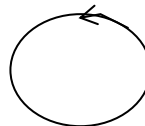

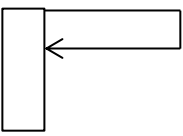


Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

(Sumber : Kusnita Yusmiarti : 2016 : 4)

II.10.3 Sequence Diagram

Diagram urutan menggambarkan kelakuan objek pada *Use Case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam diagram urutan, yaitu:

Tabel II.3 Simbol Diagram Urutan

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>formentry</i> dan <i>form</i> cetak.
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber : Kusnita Yusmiarti : 2016 : 5)

II.10.4 Class Diagram

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class diagram* secara khas meliputi: Kelas (*Class*), Relasi, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/ Method*), *Visibility*, tingkat akses objek *eksternal* kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau kardinaliti.

Tabel II.4. Simbol Class Diagram

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Kusnita Yusmiarti : 2016 : 4)