

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun penelitian terkait yang akan digunakan sebagai sumber acuan yang relevan dan terkini yaitu:

Berdasarkan penelitian yang dilakukan Rizky Putra Fhonna (2021) dengan judul “Sistem Informasi Pemesanan Tiket Pesawat Via Online Berbasis Web Pada Bandara Malikussaleh”. Suatu maskapai penerbangan sangat luas dan juga memudahkan calon penumpang sehingga tidak perlu datang langsung ke kantor cabang, tetapi cukup menggunakan internet dimanapun berada untuk melakukan pemesanan tiket pesawat. Proses pengolahan informasi pada sistem pemesanan pesawat dengan memanfaatkan teknologi web menyebabkan web menjadi media informasi yang dinamis yang dapat memudahkan calon penumpang dalam pemesanan tiket pesawat terbang di Bandar Udara Malikussaleh Aceh Utara.

Berdasarkan penelitian dari Bakhtiar (2018) dengan judul “Sistem Informasi Pemesanan Tiket Berbasis Web pada Perum Damri Palembang Menggunakan Metode Waterfall” Tujuan dari penelitian ini untuk membangun aplikasi berbasis web yang memanfaatkan teknologi internet, yang dapat menyebarluaskan informasi tentang kegiatan layanan-layanan Perum Damri cabang Palembang kepada masyarakat, guna memudahkan pengguna jasa dalam memperoleh informasi, melakukan pemesanan tiket dan mengetahui jadwal keberangkatan secara cepat, akurat dan efisien.

Berdasarkan penelitian dari Causa Prima Wijaya (2019) dengan judul “Perancangan Sistem Informasi Pemesanan Tiket Travel Berbasis Web” Didalam sistem pemesanan tiket travel ini konsumen dapat melakukan pemesanan tiket dengan 2 cara yaitu dengan mendaftar terlebih dahulu sebagai pelanggan dan melakukan pembelian tiket secara deposit atau melakukan pembelian tiket secara langsung tanpa mendaftar sebagai pelanggan, kedua cara tersebut memiliki kelebihan dan kelemahan masing-masing. Setelah dilakukan pengujian sistem diketahui bahwa semua fungsi yang terdapat dalam aplikasi telah berjalan dengan baik seperti mengelola data pemesanan tiket, penyediaan daftar tiket, serta pembelian dan pembatalan tiket sehingga semua fitur yang disediakan telah dapat memudahkan dari pihak konsumen maupun pengelola sistem.

Berdasarkan penelitian dari Muhammad Khathab (2020) dengan judul “Sistem Informasi Pemesanan tiket Pada Indah Travel Berbasis Web” Penelitian ini bertujuan untuk membangun sistem informasi pemesanan tiket pada indah travel berbasis web. Tahap pengembangan sistem meliputi analisis, perancangan, implementasi dan pengujian. Metode pengumpulan data menggunakan metode studi pustaka dan observasi. Sistem informasi pemesanan tiket pada indah travel berbasis web dibangun dengan bahasa pemrograman PHP dan database Mysql. Sistem ini dibuat untuk memberikan informasi pemesanan tiket dan jadwal keberangkatan. Hasil pengujian sistem diketahui bahwa semua fungsi yang terdapat dalam sistem telah berjalan dengan baik seperti mengelola data, penyediaan daftar tiket, serta transaksi pembelian dan pembatalan pemesanan tiket sehingga semua fitur yang disediakan dapat memudahkan bagi pelanggan dan pengelola sistem.

Berdasarkan penelitian dari Roberto Kaban, dkk (2019) dengan judul “Aplikasi Pemesanan Tiket Bus Berbasis *Android* (Study Kasus: PT. Als Terminal Pasar X Tanjung Beringin)” bahwa penelitian yang dilakukan bertujuan untuk untuk merancang aplikasi pemesanan tiket agar memudahkan petugas loket dan calon penumpang dalam proses pemesanan tiket. Dalam tahap pengumpulan data, penulis melakukan studi lapangan, observasi langsung dan wawancara pada petugas loket PT. ALS Terminal Pasar X Tanjung Beringin. Perancangan aplikasi ini menggunakan metode desain berorientasi objek dengan UML (*Unified Modeling Language*).

Berdasarkan penelitian dari Abdul Rohman Amat Rahmat (2018) dengan judul “Aplikasi Pemesanan Tiket Bus Berbasis Web (Studi Kasus Pada PO. Harapan Jaya)” Konsumen sulit mendapatkan informasi reservasi tiket bus di PO. Harapan Jaya serta kurang efesien nya dalam waktu dan biaya karena konsumen harus datang langsung ke PO. harapan jaya. Web adalah salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke internet. web ini menyediakan informasi bagi pemakai komputer yang terhubung ke internet website atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan internet.

Sedangkan berdasarkan penelitian yang dilakukan oleh penulis untuk merancang sebuah sistem pemesanan tiket masuk dengan menggunakan perangkat

Android, sehingga proses pemesanan bisa dilakukan lebih efektif. Sistem yang baru dalam pemesanan tiket masuk dengan sistem *barcode* pada sistem aplikasi yang dirancang. Sistem aplikasi pemesanan tiket masuk berbasis android sangat tepat untuk mengatasi permasalahan tersebut. Karena mayoritas wisatawan telah mempunyai dan mampu mengoperasikan Android, sehingga proses pemesanan tiket masuk dapat dilakukan dengan cepat dan dimana pun berada bisa secara langsung menerima pesanan dari pelanggan dan memperoleh laporan pemesanan tiket masuk yang akurat. Smartphone dengan sistem operasi android semakin banyak tersedia di pasaran dengan harga yang semakin terjangkau.

II.2. Landasan Teori

II.2.1. Aplikasi

Pengertian aplikasi adalah suatu bagian dari perangkat lunak yang dapat digunakan untuk menyelesaikan masalah-masalah yang khusus yang dihadapi *user* dengan menggunakan kemampuan komputer. Sedangkan pengertian penjualan adalah suatu proses seseorang atau organisasi untuk meyakinkan *customer* membeli produk yang ditawarkan. Aplikasi *mobile* dapat diartikan sebagai sebuah produk dari sistem komputasi *mobile*, yaitu sistem komputasi yang dapat dengan mudah dipindahkan secara fisik dan yang komputasi kemampuan dapat digunakan saat mereka sedang dipindahkan. Contohnya adalah *personal digital assistant* (PDA), *smartphone* dan ponsel. (Fergiawan Listianto : 2017)

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan

mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputasi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan. Aplikasi biasanya berupa perangkat lunak yang berbentuk *Software* yang berisi kesatuan perintah atau program yang dibuat untuk melaksanakan sebuah pekerjaan yang diinginkan. (Ari Setiaji : 2020).

II.2.2. Reservasi

Reservasi merupakan proses, pembuatan, cara memesan (tempat dan barang) kepada orang lain. Reservasi adalah sebuah proses perjanjian berupa pemesanan sebuah produk baik barang maupun jasa dimana pada saat itu telah terdapat kesepahaman antara konsumen dengan produsen mengenai produk tersebut namun belum ditutup oleh sebuah transaksi jual-beli. Pada saat reservasi berlangsung biasanya ditandai dengan adanya proses tukar menukar informasi antara konsumen dan produsen agar kesepahaman mengenai produk dapat terwujud. Alasan reservasi menjadi sebuah media yang sangat efektif baik bagi produsen maupun bagi konsumen adalah produsen akan dapat melakukan evaluasi terhadap produk yang akan mereka jual melalui tingkat tinggi rendahnya jumlah reservasi jauh sebelum produk tersebut dijual (barang) ataupun diselenggarakan (jasa), dimana hasil evaluasi tersebut akan membantu produsen untuk menentukan langkah pemasaran yang akan diambil terhadap produk yang akan dijual tersebut. (Roni Ameldi : 2018)

II.2.2.1. Tiket

Tiket adalah salah satu dokumen perjalanan yang dikeluarkan oleh perusahaan untuk menerapkan suatu kontrak tertulis satu pihak yang berisikan ketentuan yang harus dipenuhi oleh penumpang selama memakai jasa tiket dan data penumpang yang mempunyai masa periode waktu tertentu. Penting nya tiket untuk pelanggan sebagai alat bukti pembayaran yang menyatakan adanya perjanjian antara penyedia dan pengguna jasa di dalam pengangkutan ke berbagai tujuan yang telah disepakati kedua belah pihak antara penyedia dan pengguna tiket (Abdul Rohman Amat Rahmat : 2018)

II.2.3. QR Code

QR-Code adalah jenis simbol dua dimensi yang dikembangkan oleh Denso Wave pada tahun 1994. Setiap simbol QR-Code disusun dalam bentuk persegi dan terdiri dari function patterns dan encoding region. Seluruh simbol dikelilingi oleh batas quiet zone pada keempat sisi. Terdapat 4 jenis pola fungsi meliputi finder pattern, separators, timing patterns, dan alignment patterns. Encoding region berisi data, yang mewakili informasi versi, format informasi, data dan koreksi kesalahan. (Afif Priyambodo : 2020)

Kode QR (Quick Response) adalah jenis kode batang matriks atau kode dua dimensi yang dapat menyimpan informasi data dan dirancang untuk dibaca oleh telepon pintar yang menunjukkan bahwa isi kode harus diterjemahkan dengan sangat cepat dengan kecepatan tinggi. Kode terdiri dari modul hitam yang disusun

dalam pola persegi pada latar belakang putih. Informasi yang disandikan mungkin berupa teks, URL, atau data lainnya. QR Code dibuat oleh anak perusahaan Toyota, Denso Wave pada tahun 1994, dan pada awalnya digunakan untuk melacak inventaris dalam pembuatan suku cadang kendaraan. Gagasan di balik pengembangan kode QR adalah keterbatasan kapasitas informasi barcode (hanya dapat menampung 20 karakter alfanumerik). (Suharianto : 2020).

Barcode pada dasarnya adalah susunan garis vertical hitam dan putih dengan ketebalan yang berbeda, sangat sederhana tetapi sangat berguna, dengan kegunaan untuk menyimpan data-data spesifik misalnya kode produksi, tanggal kadaluwarsa dan nomor identitas, teknologi *barcode* tersebut terus berkembang dan bertahan. Sedangkan untuk membaca *barcode* ada banyak pilihan di pasaran dengan harga yang relatif murah mulai dari yang berbentuk pena (wand), slot dan scanner (Jawahir : 2020)

II.2.4. Android

Android merupakan sistem operasi *mobile*. *Android* tidak membedakan antara aplikasi inti dengan aplikasi pihak ketiga. *Application Programming Interface* (API) yang disediakan menawarkan akses ke *hardware*, maupun data data ponsel sekalipun, atau data sistem sendiri. *Android* merupakan sebuah sistem operasi perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi. Beberapa pengertian lain dari *Android*, yaitu :

1. Merupakan *platform* terbuka (*Open Source*) bagi para pengembang (*Programmer*) untuk membuat aplikasi.

2. Merupakan sistem operasi yang dibeli *Google Inc.* dari *Android Inc.*
3. Bukan bahasa pemrograman, tetapi hanya menyediakan lingkungan hidup atau *run time enviroment* yang disebut DVM (*Dalvik Virtual Machine*) yang telah dioptimasi untuk alat/*device* dengan sistem memori yang kecil. (Ni Kadek Ceryna : 2018 : 101)

Android merupakan sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layar sentuh (*mobile*) seperti *smartphone* dan tablet. *Android* adalah sistem operasi yang bersifat *open source* dan google merilis kodenya di bawah *lisensi apache*. *Android* awalnya dikembangkan oleh *Android, Inc.*, dengan dukungan finansial dari *google* dan kemudian *google* membelinya pada tahun 2005. *Android* dibuat menggunakan bahasa pemrograman C(core), C++, Java(UI) dan dukungan berbagai bahasa pemrograman lain. Berdasarkan dashboard developer android, *Android 8.0/1 Oreo* adalah versi *Android* yang paling banyak digunakan, yakni sekitar 28.3% dari keseluruhan perangkat *Android* di seluruh dunia. Hingga saat ini, *Android* sistem operasi berbasis mobile yang paling banyak digunakan di seluruh dunia dan android sudah di rilis versi 10 pada 3 september 2019. (Roberto Kaban : 2019)

II.2.5. Andorid Studio

Android Studio adalah *Integrated Development Enviroment* (IDE) untuk sistem operasi *Android* yang dibangun berdasarkan perangkat lunak IntelliJ IDEA. Selain editor kode dan alat pengembang yang didesain khusus untuk pengembangan android, *Android studio* merupakan pengganti dari *Eclipse Android Development*

Tools (ADT) yang sebelumnya merupakan IDE utama untuk pengembangan aplikasi android. Android studio menyediakan banyak fitur untuk meningkatkan produktivitas dalam pengembangan aplikasi berbasis android seperti tersedianya sistem pembangunan berbasis gradle yang fleksibel, emulator yang memiliki banyak fitur, integrasi dengan GitHub, dukungan bawaan untuk *Google Cloud Platform* sehingga mempermudah pengembang aplikasi untuk mengintegrasikan *Google Cloud Messaging* dan *App Engine*. (Roberto Kaban : 2019)

II.2.6 Tools Pembangunan Android

Untuk membangun sebuah sistem operasi Android dapat menggunakan Mac, Windows PC, ataupun Linux. Tools yang dibutuhkan gratis dan dapat di download dari web. Berikut adalah beberapa tools yang digunakan untuk membangun aplikasi android.

1. JDK (*Java Development Kit*)
2. Android SDK
3. ADT (*Android Development Tools*) (Ni Kadek Ceryna : 2018 : 102)

II.2.7. Basis Data (*Database*)

Basis data atau *database* merupakan koleksi dari data-data yang terorganisir dengan rapi sehingga data dapat dengan mudah disimpan dan dimanipulasi. Kita dapat menjumpai pemanfaatan *database* dalam kehidupan sehari-hari, seperti penggunaan mesin ATM, sistem akademi universitas/sekolah, sistem informasi penjualan. Salah satu tujuan dari *database* adalah memberikan pengguna suatu

pandangan abstrak dari data, yaitu sistem menyembunyikan rincian bagaimana data disimpan dan dipelihara. Sistem *database* harus dibuat semudah mungkin untuk dimengerti karena kebanyakan pengguna sistem *database* adalah orang-orang yang kurang terlatih di bidang teknologi. (Ni Kadek Ceryna : 2018 : 102)

II.2.8 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 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 *PHP MyAdmin* dengan aplikasi *XAMPP*. (Muhammad Faisal Widad : 2017)

II.2.9. PHP

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada sebuah skripsi HTML. Bahasa PHP

dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti C, *Java*, dan *Perl* serta mudah untuk dipelajari. PHP merupakan bahasa *scripting server – side*, dimana pemrosesan datanya dilakukan pada sisi *server*. Sederhananya, *server* yang akan menterjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada *client* yang melakukan permintaan. Adapun pengertian lain PHP adalah *akronim* dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode – kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke *web browser* menjadi kode HTML”. (Muhammad Faisal Widad : 2017).

PHP (singkatan rekursif: *PHP Hypertext Preprocessor*) merupakan script yang dapat disisipkan ke dalam HTML. PHP dipakai untuk membuat sistem berbasis web yang dinamis. PHP bahasa pemrograman yang *server-side* sehingga program tersebut akan dijalankan atau diproses oleh server . Hingga saat ini PHP sudah rilis versi 7.4.1 pada 18 desember 2019. (Roberto kaban : 2019)

II.2.10. UML (*Unified Modelling Language*)

Unified Modelling Language (UML) merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek. UML merupakan salah satu alat bantu yang sangat handal dalam bidang pengembangan sistem berorientasi objek karena UML menyediakan Bahasa pemodelan visual yang memungkinkan pengembang sistem membuat *blue print* atas visinya dalam bentuk yang baku. UML berfungsi sebagai

jembatan dalam mengkomunikasikan beberapa aspek dalam sistem melalui jumlah elemen grafis yang bisa dikombinasikan menjadi.

Unified Modeling Language (UML) biasa digunakan untuk :

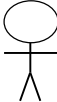
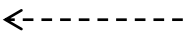
- a. Menggambarkan batasan sistem dan fungsi - fungsi sistem secara umum, dibuat dengan *use case* dan *actor*.
- b. Menggambarkan kegiatan atau proses bisnis yang dilaksanakan secara umum, dibuat dengan *interaction diagrams*.
- c. Menggambarkan representasi struktur *static* sebuah sistem dalam bentuk *class diagrams*.
- d. Membuat model behavior “yang menggambarkan kebiasaan atau sifat sebuah sistem” dengan *state transition diagrams*.
- e. Menyatakan arsitektur implementasi fisik menggunakan *component and development*.
- f. Menyampaikan atau memperluas *fungsi* dengan *stereotypes*. (Omni Alfina dan Fitriana Harahap : 2019)

UML merupakan singkatan dari *Unified Modeling Language* yang merupakan sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis objek. UML juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan. Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat pengembangan karena developer harus melakukan penelusuran dan mempelajari kode program. UML juga dapat menjadi alat bantu untuk transfer ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu developer ke developer lainnya. (Janiver W. Janis : 2020)

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 dapat dilihat pada tabel II.1 dibawah ini:

Tabel II.1. Simbol Use Case

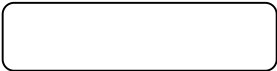
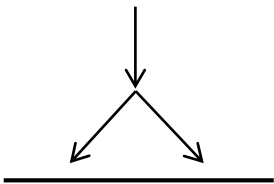
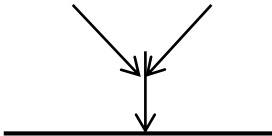
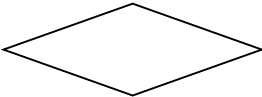
Gambar	Keterangan
	<i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang 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 control 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 : Janiver W. Janis : 2020)

2. Diagram Aktivitas (Activity Diagram)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram* dapat dilihat pada tabel II.2 dibawah ini:

Tabel II.2. Simbol Activity Diagram

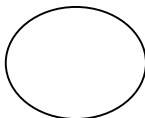
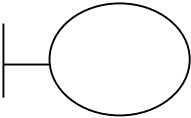
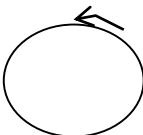
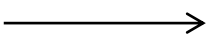
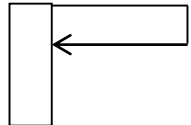
Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau 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 : Janiver W. Janis : 2020)

3. Diagram Urutan (Sequence Diagram)

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram* dapat dilihat pada tabel II.3 dibawah ini :

Tabel II.3. Simbol *Sequence Diagram*

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 formentry 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> , <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 : Janiver W. Janis : 2020)

4. *Class Diagram* (Diagram Kelas)

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 yang dapat dilihat pada tabel II.4 dibawah ini :

Tabel II.4. *Multiplicity 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 : Janiver W. Janis : 2020)