

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Penelitian Terdahulu**

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

Berdasarkan penelitian yang dilakukan Devita (2020) dengan judul “Perancangan Sistem Informasi Penjualan Kopi Pada CV Halani Kopi Medan” Penelitian ini bertujuan untuk membangun sebuah *website* penjualan produk kopi pada CV.Halani Kopi. Sistem yang akan dirancang oleh peneliti bertujuan untuk mempermudah proses penjualan kopi dalam skala besar, dan membantu proses promosi produk kopi kepada masyarakat penikmat kopi. Sistem yang akan dirancang oleh peneliti adalah sistem aplikasi yang berbasis online sehingga proses promosi dan penjualan produk kopi dapat dilakukan dengan online.

Berdasarkan Yessy Fadillah (2019) dengan judul “Sistem Informasi Penjualan Produk Krupuk Berbasis *Web Responsive* (Studi Kasus : UD. Sumber Makmur)” Penelitian ini bertujuan menghasilkan sistem informasi untuk pengolahan data penjualan produk krupuk dan untuk memperluas pangsa pasar penjualan produk krupuk di UD. Sumber Makmur. Selain itu *customer* dapat melihat informasi dan melakukan transaksi pada sistem ini. Dalam membangun sistem informasi ini peneliti

menggunakan metode *web responsive* dengan menggunakan Bahasa pemrograman *PHP* serta *MySQL* untuk databasenya.

Berdasarkan penelitian dari Asfinoza (2018) dengan judul “Sistem Informasi Penjualan Pupuk Berbasis Web pada PT. Sri Aneka Karyatama” Berdasarkan penelitian wrriter di PT. Sri Aneka Karyatama penelitian ini mengembangkan sistem informasi penjualan untuk PT. Sri Aneka Karyatama dimana beberapa masalah seperti jarak antara pelanggan dan perusahaan dalam proses pemesanan pupuk, kesalahan dalam mencatat pesanan, membutuhkan waktu sedikit lebih lama dalam membuat laporan, dan begitu banyak pesaing sehingga perusahaan ini membutuhkan media dan alat yang sangat besar. Tentang sebuah perusahaan dan produk di perusahaan ini, penulis menggunakan metodologi penelitian Iteratif metodologi ini disusun oleh 4 tahap, analisis pertama, desain kedua, pengkodean ketiga dan yang terakhir adalah sebuah ujian.

Berdasarkan penelitian dari Dadan Zaliluddin (2018) dengan judul “Perancangan Sistem Informasi Penjualan Berbasis Web (Studi Kasus Pada *Newbiestore*)” Dengan adanya *e-commerce* ini banyak sekali keuntungan yang dihasilkannya. Aplikasi perangkat lunak komputer dan *Internet* telah berkembang untuk *E-Commerce* dapat membantu mendongkrak penjualan dengan pasar yang lebih luas. Dalam jurnal ini diuraikan Penerapan *e-commerce* dalam penjualan sebuah distro pakaian menggunakan metodologi *waterfall* dan penggambaran data dengan UML.

Berdasarkan Fajar Ramadhan (2020) dengan judul “Perancangan Sistem Informasi Penjualan Ikan Air Tawar Berbasis Web (Studi Kasus : N’cex Aquatic)” Pada saat ini N’Cex *Aquatic* melakukan penjualan ikan pernah menggunakan komputerisasi berupa *Microsoft Office*. Proses pengelolaan data seperti ini sangat tidak efektif dan efisien. Apabila data yang diolah jumlahnya besar mungkin bisa terjadi kesalahan dalam pencatatan data yang disebabkan dokumen hilang ataupun rusak, menyebabkan data menjadi tidak akurat.

## **II.2. Landasan Teori**

### **II.2.1. Perancangan Sistem**

Perancangan sistem dapat didefinisikan yaitu merumuskan bagaimana cara membentuk suatu struktur. Perancangan sistem juga dapat diartikan sebagai berikut : Tahap setelah analisis dari siklus pengembangan sistem.

1. Pendefinisian dari kebutuhan-kebutuhan fungsional.
2. Persiapan untuk rancang bangun implementasi.
3. Menggambarkan bagaimana suatu sistem dibentuk.
4. Yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam suatu kesatuan yang utuh dan berfungsi.

Termasuk menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem. (Andy Victor, 2018).

## **II.2.2. Sistem**

Sistem dapat terdiri dari sistem-sistem bagian (*subsystem*). Sebagai misal, sistem komputer dapat terdiri dari subsistem perangkat keras dan subsistem perangkat lunak. Masing-masing subsistem dapat terdiri dari subsistem-subsistem yang lebih kecil atau terdiri dari komponen-komponen. Subsistem perangkat keras (*hardware*) dapat terdiri dari alat masukan, alat pemroses, alat keluaran dan simpanan luar, dan kemudian subsistem-subsistem tersebut akan berinteraksi sedemikian rupa sehingga dapat mencapai satu kesatuan yang terpadu. Dalam buku Analisa dan Design Sistem Informasi Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis. Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu (M. Syariat, 2020)

### **II.2.2.1. Karakteristik Sistem**

Sistem mempunyai karakteristik atau sifat-sifat tertentu, diantaranya adalah sebagai berikut :

#### **1. Komponen Sistem (*Components*)**

Sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem.

#### **2. Batas Sistem (*Boundary*)**

Batas sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

### 3. Lingkungan Luar Sistem (*Environments*)

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga merugikan suatu sistem.

### 4. Penghubung Sistem (*Interface*)

Penghubung merupakan media penghubung antara suatu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari suatu subsistem ke subsistem lainnya.

### 5. Masukan Sistem (*Input*)

Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*).

### 6. Keluaran sistem (*Output*)

Keluaran adalah hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat berupa masukan untuk subsistem yang lain.

### 7. Pengolah Sistem (*Process*)

Sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolah. Pengolah yang akan merubah masukan menjadi keluaran.

#### 8. Sasaran Sistem (*Objectives*)

Sistem mempunyai tujuan atau sasaran yang akan menentukan masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. (M. Syariat, 2020)

### II.2.3. Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

- (1) Akurat (*Accurate*) Berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya.
- (2) Tepat waktu (*Timelines*) Berarti informasi yang datang pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai logika karena informasi merupakan landasan dalam pengambilan keputusan.
- (3) Relevan (*Relevance*) Berarti informasi tersebut bermanfaat bagi pemakainya. (Fajar Ramadhan, 2020).

### II.2.4. Sistem Informasi

Sistem informasi adalah sekumpulan komponen-komponen sistem yang berada didalam suatu ruang lingkup organisasi, saling berinteraksi untuk

menghasilkan sebuah informasi yang bertujuan untuk pihak manajemen tertentu dan untuk mencapai tujuan tertentu. Sistem informasi adalah menggambarkan campuran dari empat faktor pokok keempat komponen utama tersebut mencakup perangkat keras (*hardware*), perangkat lunak (*software*) prasarana dan sumber daya manusia (SDM) yang berkompeten. Sementara implementasi sistem informasi bisa berbentuk sebuah mainframe, sebuah server dari kebanyakan komputer maupun hosting dalam internet di suatu komputer *server*. (Fajar Ramadhan, 2020).

Faktor-faktor yang menentukan kehandalan dari suatu sistem informasi atau informasi dapat dikatakan baik jika memenuhi kriteria-kriteria sebagai berikut :

1. Keunggulan (*Usefulness*)

Yaitu suatu sistem yang harus dapat menghasilkan informasi yang tepat dan relevan untuk mengambil keputusan manajemen dan personil operasi dalam organisasi.

2. Ekonomis

Kemampuan sistem yang mempengaruhi sistem harus bernilai manfaat minimal, sebesar biayanya.

3. Kehandalan (*Reliability*)

Keluaran dari sistem harus mempunyai tingkat ketelitian tinggi dan sistem tersebut harus beroperasi secara efektif.

4. Pelayanan (*Customer Service*)

Yakni suatu sistem memberikan pelayanan yang baik dan efisien kepada para pengguna sistem pada saat berhubungan dengan organisasi.

5. Kapasitas (*Capacity*)

Setiap sistem harus mempunyai kapasitas yang memadai untuk menangani setiap periode sesuai yang dibutuhkan.

6. Sederhana dalam kemudahan (*Simplicity*)

Sistem tersebut lebih sederhana (umum) sehingga struktur dan operasinya dapat dengan mudah dimengerti dan prosedur mudah diikuti.

7. Fleksibel (*Flexibility*)

Sistem informasi ini harus dapat digunakan dalam kondisi yang bagaimana yang diinginkan oleh organisasi tersebut atau pengguna tertentu.

8. Komponen Sistem Informasi

Istilah dalam komponen sistem informasi adalah blok bangunan (building block) yang dapat di bagi menjadi enam blok yaitu :

a. Blok masukan (*Input block*)

Blok *input* merupakan data-data yang masuk ke dalam sistem informasi, yang dapat berupa *document-document* dasar yang dapat diolah menjadi suatu informasi tertentu.

b. Blok model (*Model block*)



Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan mengolah data *input* untuk menghasilkan suatu informasi yang dibutuhkan.

c. Blok keluaran (*Output block*)

Merupakan informasi yang menghasilkan sekumpulan data yang nantinya akan disimpan berupa data cetak laporan.

d. Blok teknologi (*Technologi block*)

Blok teknologi merupakan penunjang utama dalam berlangsungnya sistem informasi. Yang memiliki beberapa komponen yaitu diantaranya alat memasukkan data (*input device*), alat untuk menyimpan dan mengakses data (*storege device*), alat untuk menghasilkan dan mengirimkan keluaran (*output divice*) dan alat untuk membentuk pengendalian sistem secara keseluruhan (*control device*). Teknologi informasi terdiri dari 3 (tiga) bagian utama, yaitu teknisi (*humanware* atau *braiware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

e. Blok basis data (*Database block*)

Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu di simpan dan perlu di organisasi sedemikian rupa, supaya informasi yang dihasilkan berkualitas.

f. Blok kendali (*Control block*)

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat di cegah bila terlanjur terjadi. (Fajar Ramadhan, 2020).

#### **II.2.5. Penjualan**

Penjualan merupakan syarat mutlak keberlangsungan suatu usaha, karena dengan penjualan maka akan didapatkan keuntungan. Semakin tinggi penjualan maka keuntungan yang akan didapatpun akan semakin maksimal. Untuk mencapai tujuan ini maka sangat diperlukan usaha-usaha agar konsumen mempunyai daya tarik dan sifat loyal dalam berbelanja disuatu unit usaha. Suatu perusahaan tidak akan berkembang apa bila tidak mampu menjual produk yang dihasilkan, sebaliknya suatu perusahaan mampu untuk terus meningkatkan penjualan maka perusahaan tersebut akan mampu untuk eksis dalam persaingan usaha. (Rogi Gusrizaldi, 2018).

Penjualan merupakan suatu kegiatan yang dilakukan sebagian manusia dalam menjual barang dagangan yang dimiliki baik itu barang ataupun jasa kepada pasar agar mencapai suatu tujuan yang diinginkan. Penjualan adalah suatu transaksi yang bertujuan untuk mendapatkan suatu keuntungan, dan merupakan suatu jantung dari suatu perusahaan. Dengan pengertian lain definisi penjualan adalah pemindahan hak milik atas barang atau pemberian jasa yang dilakukan penjualan kepada pembeli dengan harga yang disepakati bersama dengan jumlah yang dibebankan kepada pelanggan dalam penjualan barang atau jasa dalam suatu periode akuntansi (Andri Prasetyo, 2018)

### **II.2.6. *E-Commerce***

*E-Commerce* merupakan kegiatan atau aktivitas yang berkaitan dengan proses penjualan atau pembelian produk, jasa ataupun informasi antara perusahaan, rumah tangga, individu, pemerintah, dan masyarakat atau organisasi swasta lainnya, yang dilakukan dengan memanfaatkan komputer pada media jaringan misalnya internet. Jadi dapat disimpulkan *e-commerce* merupakan segala bentuk transaksi perdagangan atau perniagaan barang dan jasa baik itu proses pembelian, penjualan, pentransferan, atau pertukaran produk, jasa atau informasi yang merupakan suatu set dinamis teknologi, aplikasi, dan proses bisnis yang menghubungkan perusahaan, konsumen, dan komunitas tertentu yang dilakukan secara elektronik. (Andi Ridho Rachman, 2017).

### **II.2.7 Jenis-Jenis Transaksi *E-Commerce***

Saat ini ada beberapa jenis *e-commerce* yang dapat dibedakan berdasarkan transaksi yang dilakukan oleh pengguna nya, mulai dari usaha kecil sampai usaha yang besar sekali pun dapat dibedakan berdasarkan transaksi *e-commerce* nya. Jenis-jenis transaksi *E-Commerce*, yaitu:

1. *Collaborative Commerce (e-commerce)*

Kerja sama secara elektronik antara rekan bisnis. Kerja sama ini biasanya terjadi antara rekan bisnis yang berada pada jalur penyediaan barang (supply chain).

2. *Business to Business (B2B)*

Tipe ini meliputi transaksi antar organisasi yang dilakukan di electronic market.

3. *Business to Consumer (B2C)*

Penjual adalah suatu organisasi dan pembeli adalah individu.

4. *Consumer to Business (C2B)*

Konsumen memberitahukan kebutuhan atas suatu produk atau jasa tertentu, dan para pemasok bersaing untuk menyediakan produk atau jasa tersebut ke konsumen.

5. *Consumer to Consumer (C2C)*

Dimana konsumen menjual secara langsung ke konsumen lain atau mengiklankan jasa pribadi di internet. Dapat juga disebut sebagai pelanggan ke pelanggan, yaitu orang yang menjual produk dan jasa ke satu sama lain. (Andi Ridho Rachman, 2018).

## **II.2.8 Website**

*Website* adalah halaman informasi yang disediakan melalui jalur internet dan memiliki fasilitas hiperteks untuk menampilkan data berupa teks, gambar, suara, animasi, dan data multimedia lain nya sehingga bisa diakses di seluruh dunia selama terkoneksi dengan jaringan internet. *Website* dapat diartikan sekumpulan halaman

situs yang saling berhubungan menampilkan berbagai macam informasi maupun data berupa teks, gambar diam ataupun bergerak, data animasi, suara, video maupun gabungan dari semuanya yang bisa diakses dimana saja kapan saja diseluruh dunia. Dimana *website* dikategorikan menjadi dua, yaitu:

1. *Web Statis*

Adalah web yang berisi/menampilkan informasi-informasi yang sifatnya *statis* (tetap). Disebut statis karena pengguna tidak dapat berinteraksi dengan *web* tersebut. Pengguna hanya dapat melihat isi dokumen pada halaman web dan apabila diklik akan berpindah kehalaman web yang lain. Interaksi pengguna terbatas hanya melihat informasi yang ditampilkan, tetapi tidak bisa mengolah informasi yang dihasilkan. Biasanya merupakan *HTML* yang ditulis pada editor teks dan disimpan dalam bentuk *.html* atau *.htm*.

2. *Web Dinamis*

Adalah *web* yang menampilkan informasi serta dapat berinteraksi dengan pengguna. *Web* yang dinamis memungkinkan pengguna untuk berinteraksi dengan *form* sehingga dapat mengolah informasi yang ditampilkan. Bersifat teraktif, tidak kaku, dan terlihat lebih indah. (Andi Ridho Rachman, 2018).

## **II.2.5 Basis Data (*Database*)**

Basis data atau *database* merupakan koleksi dari data-data yang terorganisir dengan rapi sehingga data dapat dengan mudah disimpan dan dimanipulasi. 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.2.11. XAMPP**

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Nama *XAMPP* merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam *GNU General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis. Untuk mendapatkannya dapat *men-download* langsung dari *web* resminya. (Randi Palit, 2018 ).

#### **II.2.12. 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 : 2018)

### **II.2.13.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”. 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.2.14. 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 sitem 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)

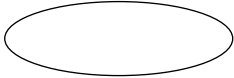
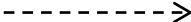
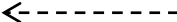
UML merupakan singkatan dari *Unified Modeling Language* yang merupakan sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis objek. UML juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan. Aplikasi atau sistem 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 melakukan (*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**

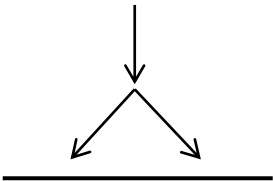
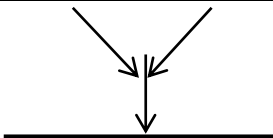
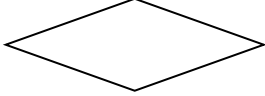
| Gambar                                                                              | Keterangan                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | <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 mengidentifikasikan 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**

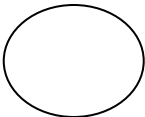
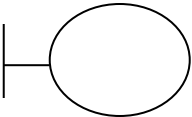
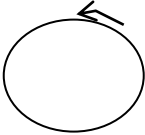
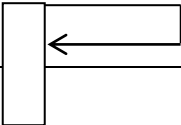
| Gambar                                                                              | Keterangan                                                                                                                                                 |
|-------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | <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 pararel 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**

| <b>Multiplicity</b> | <b>Penjelasan</b>                 |
|---------------------|-----------------------------------|
| 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)***