

## BAB II LANDASAN TEORI

### 1.1 TinjauanPustaka

Dalam penelitian ini akan digunakan lima tinjauan pustaka yang nantinya dapat mendukung penelitian, berikut ini merupakan tinjauan pustaka yang diambil yaitu:

**Tabel 1.1** Tinjauan Studi

No	Nama & Tahun	Judul	Metode	Hasil Dari Penelitian
1	(Saputro, Aryadita dan Priyambadha), 2018)	Pengembangan Sistem Informasi <i>Electronic Supply Chain Management</i> (Studi Kasus: CV. Baiducha Technology)	Waterfall	perusahaan masih melakukan proses pemesanan bahan baku hanya melalui jaringan telepon, Berdasarkan hasil pengujian fungsionalitas yang telah dilakukan menunjukkan bahwa sistem ini telah memenuhi kebutuhan sesuai dengan spesifikasi yang ditentukan, dan juga hasil dari pengujian kompatibilitas menunjukkan bahwa sistem dapat berjalan dengan baik pada berbagai peramban yang berbeda

**Tabel 2.1** Tinjauan Studi (Lanjutan)

2	(Noor Azizah dan Virliana Dicky Pramandari, 2017)	IMPLEMENTASI <i>SUPPLY CHAIN MANAGEMENT</i> PADA UMKM TENUN TROSO JEPARA	Pengumpulan data	Dengan adanya sistem yang telah terkomputerisasi, diharapkan dapat meringankan pekerjaan dan efisiensi waktu manager dalam menyeleksi pemasok dan distributor yang ada. Serta dengan adanya sistem terkomputerisasi tersebut dapat memonitor pemasok dan distributo
3	(Vistasusiyanti, 2017)	<i>ANALYSIS SUPPLY CHAIN MANAGEMEN T (SCM) SPRING BED AT PT. MASSINDO SINAR PRATAMA KOTA MANADO</i>	Analisis data kualitatif	penerapan proses manajemen rantai pasokan pada PT. Massindo Sinar Pratama adalah terdapat pada fungsi manajemen yang mendukung kegiatan rantai pasokan spring bed dimanabagian perencanaan, pengorganisasian, pelaksanaan dan pengawasan. Pada rantai pasokan yang baik yaitu lancar, memenuhi kualitas, serta tepat waktu akan berdampak pada persediaan bahan baku yang dibutuhkan oleh pihak perusahaan baik untuk persediaan jangka pendek, jangka menengah, maupun jangka panjang.

**Tabel 2.1** Tinjauan Studi (Lanjutan)

4	(Silfia Andini, 2017)	Perancangan Dan Implementasi <i>Supply Chain Management</i> (SCM) Pada CV Hayati Padang	Pengumpulan data dan observasi	Dengan adanya sistem supply chain management CV hayati dapat mengendalikan stok sepeda motor sehingga permintaan konsumen dapat terpenuhi. Sebelumnya produk yang ingin dibeli konsumen sering kosong namun setelah adanya aplikasi SCM ini stok produk dikendalikan secara terkomputerisasi sehingga petugas dapat mengetahui jika stok produk hampir habis sehingga petugas dapat segera memesan produk kepada pemasok untuk menghindari kekosongan stok.
5	(Sari, Renaldi, & Umbara, 2017)	Pembangunan sistem informasi manajemen rantai pasokan Dalam produksi barang di pt.cisangkan purwakarta	Waterfall	Kesulitan utama adalah manager yang membutuhkan waktu untuk melakukan pengecekan. Karena pengecekan dilakukan dengan mendatangi setiap bagian dan menanyakannya satu persatu, tentu hal ini akan mengganggu pekerjaan, mengintegrasikan keseluruhan data yang berhubungan dengan proses produksi barang

1. (Saputro, Aryadita dan Priyambadha), 2018) meneliti tentang Pengembangan Sistem Informasi *Electronic Supply Chain Management* (Studi Kasus: CV. Baiducha

Technology), dengan hasil penelitian perusahaan masih melakukan proses pemesanan bahan baku hanya melalui jaringan telepon, Berdasarkan hasil pengujian fungsionalitas yang telah dilakukan menunjukkan bahwa sistem ini telah memenuhi kebutuhan sesuai dengan spesifikasi yang ditentukan, dan juga hasil dari pengujian kompatibilitas menunjukkan bahwa sistem dapat berjalan dengan baik pada berbagai peramban yang berbeda.

2. (Noor Azizah dan Virliana Dicky Pramandari, 2017) meneliti tentang IMPLEMENTASI *SUPPLY CHAIN MANAGEMENT* PADA UMKM TENUN TROSO JEPARA, dengan hasil penelitian Dengan adanya sistem yang telah terkomputerisasi, diharapkan dapat meringankan pekerjaan dan efisiensi waktu manager dalam menyeleksi pemasok dan distributor yang ada. Serta dengan adanya sistem terkomputerisasi tersebut dapat memonitor pemasok dan distributor.
3. (Vistasusiyanti, 2017) meneliti tentang *ANALYSIS SUPPLY CHAIN MANAGEMENT (SCM) SPRING BED AT PT. MASSINDO SINAR PRATAMA KOTA MANADO* dengan hasil penelitian penerapan proses manajemen rantai pasokan pada PT. Massindo Sinar Pratama adalah terdapat pada fungsi manajemen yang mendukung kegiatan rantai pasokan spring bed dimanabagian perencanaan,pengorganisasian, pelaksanaan dan pengawasan. Pada rantai pasokan yang baik yaitu lancar,memenuhi kualitas, serta tepat waktu akan berdampakpada persediaan bahan baku yang dibutuhkan oleh pihak perusahaan baik untuk persediaan jangka pendek,jangka menengah, maupun jangka panjang.
4. (Silfia Andini, 2017) meneliti tentang Perancangan Dan Implementasi *Supply Chain Management (SCM)* Pada CV Hayati Padang dengan hasil penelitian Dengan adanya sistem supply chain management CV hayati dapat mengendalikan stok sepeda motor sehingga permintaan konsumen dapat terpenuhi. Sebelumnya produk yang ingin dibeli

konsumen sering kosong namun setelah adanya aplikasi SCM ini stok produk dikendalikan secara terkomputerisasi sehingga petugas dapat mengetahui jika stok produk hampir habis sehingga petugas dapat segera memesan produk kepada pemasok untuk menghindari kekosongan stok.

5. (Sari, Renaldi, & Umbara, 2017) meneliti tentang Pembangunan sistem informasi manajemen rantai pasokan Dalam produksi barang di pt.cisangkan purwakarta dengan hasil penelitian Kesulitan utama adalah manager yang membutuhkan waktu untuk melakukan pengecekan. Karena pengecekan dilakukan dengan mendatangi setiap bagian dan menanyakannya satu persatu, tentu hal ini akan mengganggu pekerjaan bagian tersebut dan manager akan mengalokasikan waktunya untuk melakukan hal itu, mengintegrasikan keseluruhan data yang berhubungan dengan proses produksi barang.

Berdasarkan penelitian terdahulu seperti diatas menunjukkan bahwa dengan menerapkan *supply chain management* menggunakan *web* dapat melakukan permintaan pasokan menjadi lebih mudah, serta dapat dilihat perbedaannya yaitu :

1. Metode pengembangan sistem dan alat pengembangan sistem yang dipakai yaitu *Extreme Programming* dan *UML (Unit Modelling Language)* sebagai alat pengembangan sistem serta metode pengujian sistem menggunakan *Acceptent Test Criteria*.
2. Sistem yang akan dibuat berbasis *web* menggunakan *adobe dreamweaver cs 8* dan *database MySQL* sebagai penyimpanan data.
3. Hasil penelitian yaitu merancang dan membangun sistem informasi rantai pasokan dengan tujuan untuk mempermudah proses pendistribusian barang dan retur barang serta proses pengolahan persediaan barang.

## **1.2 Sistem Informasi**

Pengertian menurut Krismiaji (2015) Sistem informasi adalah cara-cara yang diorganisasi untuk mengumpulkan, memasukkan, dan mengolah serta menyimpan data, dan cara-cara yang diorganisasi untuk menyimpan, mengelola, mengendalikan, dan melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah ditetapkan.

Definisi menurut Nas (2018) Sistem informasi adalah sistem pemrosesan data, merupakan sistem buatan manusia yang biasanya terdiri dari sekumpulan komponen (baik manual maupun berbasis komputer) yang terintegrasi untuk mengumpulkan, menyimpan, dan mengelola data serta menyediakan informasi mengenai saldo persediaan.

Jadi berdasarkan pengertian diatas dapat disimpulkan bahwa sistem informasi adalah kumpulan data yang terintegrasi dan saling melengkapi dengan menghasilkan *output* yang baik guna untuk memecahkan masalah dan pengambilan keputusan.

## **1.3 Pengertian *Supply Chain Management***

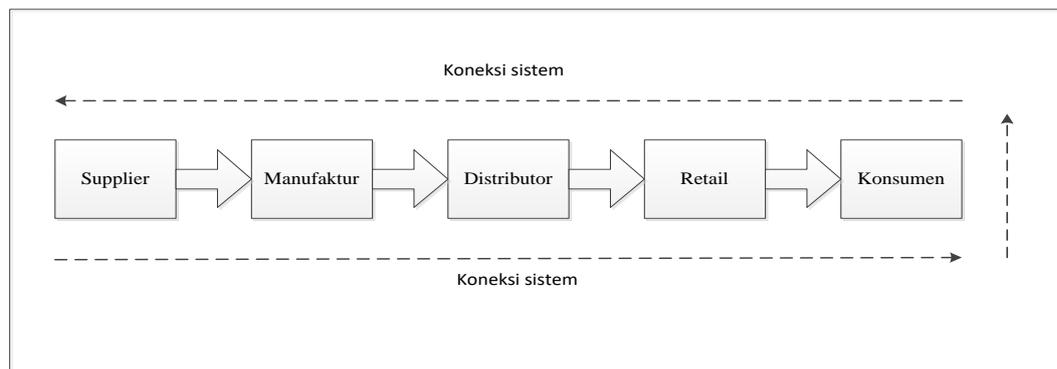
Supply Chain menurut Sakti (2016) adalah sebuah kelompok dari partisipasi perusahaan yang saling terkait yang menambahkan nilai pada aliran dari perubahan input dari Sumber asal mereka ke produk akhir atau jasa yang dituntut dari konsumen akhir yang dituju.

*Supply Chain management* menurut Akhiar (2016) adalah Integrasi aktivitas untuk mendapatkan material dan servis, mengubahnya menjadi barang setengah jadi dan barang jadi, dan mengirimkannya kepada konsumen.

Dari beberapa definisi *Supply Chain Management* diatas, maka dapat disimpulkan *Supply Chain Management* adalah Suatu metode yang mengintegrasikan pengelolaan aliran

informasi, produk, barang dan jasa dalam fungsi *supply chain* dengan pendekatan yang terintegrasi.

Pemodelan *supply chain management* merupakan penggambaran prosedur pada proses pasokan suatu perusahaan, berikut adalah model SCM pada Gambar 2.1 :



**Gambar 1.1** Konsep SCM  
Sumber : (Hamed, 2018)

Berdasarkan model *supply chain management* tersebut dapat dilihat proses pasokan dimulai dari pemasok diproses ke manufaktur diproses pada distributor hingga ke retail lalu di terima oleh konsumen, pada kasus yang dikembangkan peneliti dan berdasarkan SCM model tersebut, proses permintaan barang yang dilakukan oleh retail terhadap distributor.

### 1.3.1 Pemasok

Pemasoks menurut (Dawei, 2011) adalah Jaringan SC ini diberi nama pemasok yang merupakan penyedia bahan pertama dalam bentuk bahan baku, bahan mentah, bahan penolong, bahan dagangan, suku cadang atau sebagainya.

### 1.3.2 Manufaktur

Manufaktur menurut (Dawei, 2011) adalah Rantai pertama dihubungkan dengan rantai kedua, yaitu manufaktur atau bentuk lain yang melakukan pekerjaan membuat, memfabrikasi, mengasembling, merakit ataupun menyelesaikan barang (*finishing*). Hubungan dengan mata

rantai pertama ini sudah mempunyai potensi untuk melakukan penghematan. Misalnya, persediaan bahan baku, bahan setengah jadi, dan bahan jadi yang berada di pihak pemasok, manufaktur, dan tempat-tempat transit merupakan target untuk penghematan ini.

### **1.3.3 Distributor**

Distributor menurut (Dawei, 2011) adalah merupakan barang sudah jadi yang dihasilkan oleh manufaktur sudah harus mulai disalurkan kepada pelanggan. Walaupun sudah tersedia, banyak cara untuk menyalurkan barang kepada pelanggan, yang umumnya adalah melalui distributor dan ini biasanya ditempuh oleh sebagian supply chain. Barang dari pabrik melalui gudangnya disalurkan ke gudang distributor atau *wholesaler* atau pedagang besar dalam jumlah besar, dan pada waktunya pedagang besar menyalurkan dalam jumlah yang lebih kecil kepada retail atau pengecer.

### **1.3.4 Retail**

Retail menurut (Dawei, 2011) adalah *Retail Outlets Wholesales* biasanya mempunyai gudang yang digunakan untuk menimbun barang sebelum disalurkan ke pihak pengecer. Dalam jaringan ini ada kesempatan untuk memperoleh penghematan dalam bentuk jumlah *inventories* dan biaya gedung, dengan cara melakukan *re-design* pola pengiriman barang baik dari gudang manufaktur maupun ke toko pengecer.

### **1.3.5 Consumers**

Konsumen menurut (Dawei, 2011) adalah *Consumers* Dari rak-raknya, para pengecer ini menawarkan barangnya langsung kepada pelanggan atau konsumen, yang termasuk *outlets* adalah toko, warung, toko serba ada, pasar swalayan, mal, dan sebagainya. Walaupun secara fisik ini dapat dikatakan bahwa merupakan mata rantai terakhir, sebetulnya masih ada satu mata rantai lagi, yaitu dari pembeli (yang mendatangi *retail outlet*) ke *realcustomers* atau *realuser*, karena pembeli belum tentu pengguna sesungguhnya. Mata rantai supply beru

betul-betul berhenti setelah barang yang bersangkutan tiba di pemakai langsung (pemakai yang sebenarnya) barang atau jasa.

#### **1.4 Web Based**

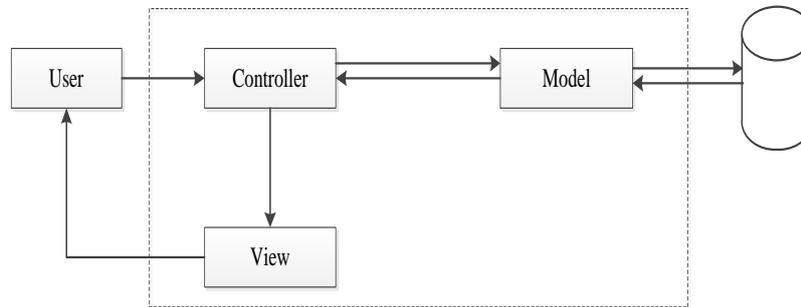
Menurut Urbietta *et al.* (2019), *Web Based* adalah aplikasi yang dibuat berbasis *web* yang membutuhkan *web server* dan *browser* untuk menjalankannya. Dengan membuat sistem berbasis *web based* ada beberapa hal yang penting dan harus kita pikirkan sebelum membangun sistem tersebut, diantaranya:

1. Tidak membutuhkan *hardware* dengan spesifikasi yang tangguh untuk menjalankan aplikasinya.
2. Server yang dibutuhkan cukup diinstallkan *tools* pendukung saja agar klien mudah menjalankan aplikasi
3. Infrastruktur jaringan yang dibutuhkan juga cukup besar karena aplikasi yang dibuat dapat diakses dari jaringan luar (internet).
4. Aplikasi berbasis *web based* dapat diakses dari berbagai perangkat dengan syarat menggunakan *web browser* saja sudah dapat mengaksesnya.
5. Jika aplikasi yang sudah jadi ingin di *update*, sangat mudah untuk melakukannya karena tidak membutuhkan membuka keseluruhan aplikasi.

##### **1.4.1 CodeIgniter**

Menurut Raharjo (2018), *CodeIgniter* adalah *Framework* untuk bahasa pemrograman PHP, yang dibuat Rick Ellis pada tahun 2006. *CodeIgniter* memiliki banyak fitur yang membantu para pengembang PHP untuk dapat membuat aplikasi secara mudah dan cepat serta memiliki sifat yang fleksibel dapat mengembangkan dalam perangkat *web*, *desktop* maupun *mobile*.

*CodeIgniter* memiliki konsep atau pola *Model-View-Controller* (MVC) sehingga kode-kode dapat di sederhanakan.



**Gambar 1.2** Arsitektur MVC

### 1.4.2 PHP

Menurut Subagja (2018), PHP adalah bahasa *server-side-scripting* yang menyatudengan HTML untuk membuat halaman *web* yang dinamis.

Menurut Aryani *et al.*, (2015), berpendapat bahwa *PHP Hypertext Preprocessor* adalah bahasa pemrograman *web server-side* yang bersifat *open source*. PHP merupakan *script* yang terintegrasi dengan HTML dan berada pada *server* (*server side HTML embedded scripting*). PHP adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru/*up to date*. Semua *script* PHP dieksekusi pada *server* dimana *script* tersebut dijalankan. Dengan menggunakan program PHP, sebuah *website* akan lebih interaktif dan dinamis.

Sehingga PHP merupakan bahasa pemrograman yang digunakan oleh pengembang untuk membuat sistem *website* dengan kumpulan bahasa HTML dan *script* lainnya.

### 1.4.3 MySql

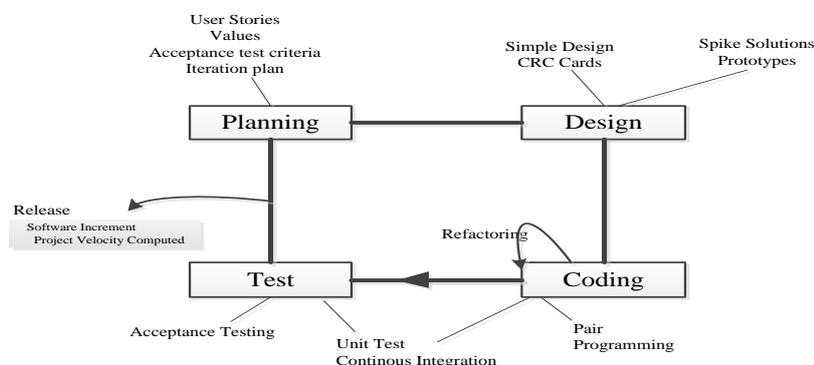
Menurut MySQL (2018), *MySQL* adalah singkatan dari *Structue Query Language* yang digunakan untuk mendefinisikan *structure* data, memodifikasi data pada basis data, menspesifikasi batasan keamanan (*security*), hingga pemeliharaan data.

Menurut Amin (2018) mendefinisikan *mysql* adalah RDBMS yang cepat dan mudah digunakan, serta sudah banyak digunakan untuk berbagai kebutuhan.

*MySQL* merupakan bahasa standar yang paling banyak digunakan untuk mengakses *data base* relasional dan merupakan aplikasi yang dapat dipergunakan secara bebas.

### 1.5 Extreme Programming

Menurut Suryantara (2017) *extreme programming* merupakan salah satu metodologi rekayasa perangkat lunak yang banyak digunakan untuk mengembangkan aplikasi oleh para developer dan merupakan pengembangan proyek yang memerlukan adaptasi yang cepat dalam perubahan-perubahan yang terjadi selama pengembangan aplikasi. Tujuan penggunaan metode tersebut untuk menghasilkan perangkat lunak yang berkualitas serta produktif dengan siklus pengembangan yang cukup singkat. Berikut merupakan konsep *extreme programming*.



**Gambar 1.3** *Extreme Programming*  
Sumber: (Suryantara, 2017)

a. *Kelebihan model Extreme Programming*

Komunikasi dalam XP dibangun dengan melakukan pemrograman berpasangan (*pair programming*). *Developer* didampingi oleh pihak klien dalam melakukan *coding* dan *unit testing* sehingga klien bisa terlibat langsung dalam pemrograman sambil berkomunikasi dengan *developer*. Selain itu perkiraan beban tugas juga diperhitungkan.

1. Menekankan pada kesederhanaan dalam pengkodean: “*What is the simplest thing that could possibly work?*” Lebih baik melakukan hal yang sederhana dan mengembangkannya besok jika diperlukan. Komunikasi yang lebih banyak mempermudah, dan rancangan yang sederhana mengurangi penjelasan.
2. Setiap *feed back* ditanggapi dengan melakukan tes, *unit test* atau *system integration* dan jangan menunda karena biaya akan membengkak (uang, tenaga, waktu).
3. Banyak ide baru dan berani mencobanya, berani mengerjakan kembali dan setiap kali kesalahan ditemukan, langsung diperbaiki.

b. *Kelemahan model Extreme Programming*

Kelemahan pada pengembangan tersebut dapat dilihat berdasarkan kesesuaian pengembangan yang dilakukan, berikut adalah kelemahan metode pengembang sistem:

1. *Developer* harus selalu siap dengan perubahan karena perubahan akan selalu diterima.
2. Tidak bisa membuat kode yang detail di awal (*prinsip simplicity* dan juga anjuran untuk melakukan apa yang diperlukan hari itu juga).

### **1.5.1 Tahapan Penelitian**

Tahapan dalam penelitian sebagai langkah-langkah penelitian yang harus dikerjakan, berikut adalah tahapan penelitian *extreme programming*.

a. Tahap Perencanaan

Peneliti atau pengembang memutuskan bagaimana hasil *story* dari pengguna dibangun dengan komitmen telah disepakati, adapun *story-story* yang dilakukan dengan cara :

1. Pengguna menceritakan apa permasalahan pada sistem yang digunakan dan sistem seperti apa yang akan dibangun.
2. Berdasarkan hasil cerita pengguna maka peneliti menentukan poin pada bagian *value* untuk memutuskan apa saja yang akan dibangun.
3. Dari hasil kesepakatan tersebut maka peneliti menentukan *acceptance criteria test* yaitu menentukan kriteria-kriteria apa saja yang nantinya sebagai acuan terhadap sistem yang akan di uji.
4. Sehingga hasil peneliti menyimpulkan berapa kali akan dilakukan *realies* dan perbaikan pada tahap *iteration plan* merencanakan berapa kali akan dilakukan uji terhadap sistem yang dibangun.

b. Tahap Perancangan

*Extreme programming* pada proses pembuatan desain di lakukan untuk memberikan informasi gambaran sistem yang akan dibangun, berikut adalah beberapa desain yang akan dilakukan oleh peneliti:

1. *CRC card* untuk mengenali dan mengatur *object oriented class* yang sesuai dengan pengembangan. Jika pada saat perancangan terdapat ketidak sesuaian maupun perbaikan maka akan dilakukan
2. *Spike solution* yang dilakukan kepada pengguna untuk mendapatkan kesesuaian antara ke inginan pengguna dengan pengembangan yang dilakukan.
3. *Prototype* adalah bagian perancangan berupa *user interface* dalam bentuk *wireframing* untuk mempermudah pengguna melihat desain sistem.

c. Tahap Pengkodean

Pada proses pengkodean peneliti menyesuaikan terhadap *story* pengguna sehingga sistem yang dibangun sesuai, proses pengkodean yang dilakukan yaitu:

1. *Pair Programming* merupakan tahap sistem dibangun dengan bahasa pemrograman dan media penyimpanan yang telah disepakati.
2. *Refactory* merupakan tahapan yang dilakukan ketika terjadi ketidak sesuaian kode program sehingga dilakukan perbaikan guna mendapatkan hasil yang sesuai.

d. Tahap Pengujian

Tahap pengujian dilakukan oleh pengguna sebagai user dengan melakukan uji sesuai dengan *acceptance test* yang telah ditentukan dan disetujui. Unit *test* yang telah dibuat fokus pada keseluruhan fitur dan fungsional sistem. Sehingga sistem dapat disimpulkan telah sesuai dan dapat di *realies*.

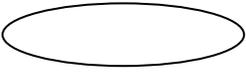
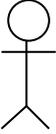
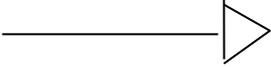
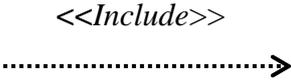
## **1.6 Unified Modelling Language (UML)**

Menurut Rosa and Shalahuddin (2019) UML (*unified Modelling Language*) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. Berikut ini merupakan penjelasan tentang masing-masing diagram yang ada pada UML (*Unified Modelling Language*).

### **1.6.1 Use Case Diagram**

Menurut Rosa and Shalahuddin (2019) *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Use Case Diagram* dapat dilihat pada tabel 2.2.

**Tabel 1.2** Simbol *Use Case Diagram*

No	Simbol	Deskripsi
1.		<i>Usecase</i> Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .
2.		Aktor Aktor seseorang/sesuatu yang berinteraksi dengan yang akan dibuat. diluar sistem informasi. Biasanya dinyatakan menggunakan kata benda
3.		Asosiasi/association merupakan komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.
4.		Generalisasi ( <i>generalization</i> ) merupakan hubungan (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum
5.		Include berarti use case yang ditambahkan akan dipanggil saat use case tambahan dijalankan.
6.		Ekstensi ( <i>extend</i> ) merupakan use case tambahan ke sebuah use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu.

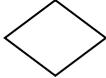
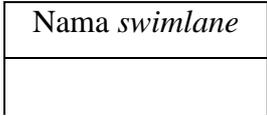
Sumber :(Rosa and Shalahuddin, 2019)

### 1.6.2 Activity Diagram

Menurut Rosa and Shalahuddin (2019) *activity* diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Berikut

simbol-simbol yang akan digunakan dalam menggambarkan *activity diagram* dapat dilihat pada tabel 2.3 berikut ini :

**Tabel 1.3** Simbol *Activity Diagram*

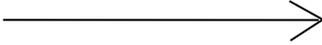
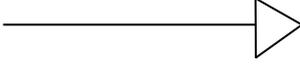
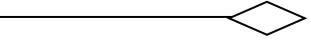
No.	Simbol	Keterangan
1.		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		Percabangan ( <i>Decision</i> ) merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.		Penggabungan ( <i>Join</i> ) merupakan asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.		Swimlane Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas.
6.		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

Sumber : (Rosa and Shalahuddin, 2019)

### 1.6.3 Class Diagram

Menurut Rosa and Shalahuddin (2019) *Class diagram* mengembangkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Class Diagram* dapat dilihat pada tabel 2.4.

**Tabel 1.4** Simbol *Class Diagram*

No.	Simbol	Deskripsi
1.		Kelas pada struktur sistem.
2.	<p>Antar Muka/<i>Interface</i></p>  <p>Nama_<i>Interface</i></p>	Sama dengan konsep interface dalam pemrograman berorientasi objek.
3.	<p>Asosiasi / <i>Association</i></p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan symbol
4.	<p>Asosiasi Berarah / <i>Directed Association</i></p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan symbol.
5.	<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
6.	<p>Agregasi / <i>aggregation</i></p> 	Relasi antar kelas dengan maksna semua bagian ( <i>whole-part</i> )

**Sumber:** (Rosa and Shalahuddin, 2019)

### 1.7 *Balsamiq Mockups*

Menurut Enggar (2016) *Balsamiq Mockups* adalah salah satu *software* yang digunakan dalam pembuatan desain atau *prototyping* dalam pembuatan tampilan *user interface* sebuah aplikasi. Dengan menggunakan *Balsamiq Mockup* kita dimudahkan dalam pembuatan *user interface* karena *Balsamiq Mockup* sudah menyediakan *tools* yang dapat memudahkan dalam membuat desain *prototyping* aplikasi yang akan kita buat. *Software* ini berfokus pada konten yang ingin digambar dan fungsionalitas yang dibutuhkan oleh pengguna.

*Balsamiq* merupakan aplikasi yang disediakan untuk para *designer* guna mendesign *mockups*, dimana *mockups* adalah sebagai sebuah model dari suatu struktur atau alat baik *full*

*size* ataupun berupa miniatur yang digunakan untuk pembelajaran, demo, *test desain*, promosi, dsb.

## 1.8 Black Box Testing

Menurut Dondeti (2012)

“Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black-box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program”.

Pendekatan pengujian *Black-Box* adalah metode pengujian di mana data tes berasal dari persyaratan fungsional yang ditentukan tanpa memperhatikan struktur program akhir. Karena hanya fungsi dari modul perangkat lunak yang menjadi perhatian, pengujian *Black-Box* juga mengacu pada uji fungsional, metode pengujian menekankan pada menjalankan fungsi dan pemeriksaan inputan dan data output (Howden, 2017)

Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut (Howden, 2017):

- a. Fungsi – fungsi yang tidak benar atau hilang,
- b. Kesalahan interface
- c. Kesalahan dalam struktur data atau akses eksternal
- d. Kesalahan kinerja
- e. Inisialisasi dan kesalahan terminasi

Menurut Howden (2017) mengemukakan ciri-ciri *black box testing*, diantaranya sebagai berikut:

1. *Black box testing* berfokus pada kebutuhan fungsional pada software, berdasarkan pada spesifikasi kebutuhan dari software.

2. *Black box testing* bukan teknik alternatif daripada *white box testing*. Lebih dari pada itu, ia merupakan pendekatan pelengkap dalam mencakup error dengan kelas yang berbeda dari metode *white box testing*.

3. *Black box testing* melakukan pengujian tanpa pengetahuan detail struktur internal dari sistem atau komponen yang dites. Juga disebut sebagai *behavioral testing*, *specification-based testing*, *input/output testing* atau *functional testing*.

Pada *black box testing* terdapat jenis teknik design tes yang dapat dipilih berdasarkan pada tipe testing yang akan digunakan, diantaranya sebagai berikut:

1. *Equivalence Class Partitioning*
2. *Boundary Value Analysis*
3. *State Transitions Testing*
4. *Cause-Effect Graphing*

Kategori kesalahan/error yang akan diketahui melalui *black box testing*:

1. Fungsi yang hilang atau tak benar/salah
2. Error dari antar-muka/*interface*
3. Error dari struktur data atau akses eksternal *database*
4. Error dari kinerja atau tingkah laku/*perform*
5. Error dari inisialisasi dan terminasi

Adapun kerangka yang akan digunakan untuk melakukan pengujian dapat dilihat pada

Tabel 2.5

**Tabel 1.5** Kerangka Pengujian

<b>Kasus dan Hasil Uji</b>			
<b>Data Masukan</b>	<b>Data Diharapkan</b>	<b>Pengamatan</b>	<b>Kesimpulan</b>
			Diterima ( ) Ditolak ( )
			Diterima ( ) Ditolak ( )

Rumus dari kuesioner pengujian *black box* menggunakan analisa deskriptif, ditunjukkan sebagai berikut:

$$\% Skor = \frac{Skor\ Aktual}{Skor\ Ideal} \times 100\% \dots\dots\dots(1)$$

Keterangan:

Skor Aktual : Jawaban diterima responden

Skor Ideal : Total jumlah butir soal yang telah diujikan kepada responden Adapun kriteria hasil perhitungan kelayakan sistem dari pengujian ini dapat dilihat pada Tabel 2.6.

**Tabel 1.6** Kriteria Persentase Skor Tanggapan Responden

<b>Jumlah Skor</b>	<b>Kriteria</b>
0.00 – 36.00	Tidak Layak
36.01 – 52.00	Kurang Layak
52.01 – 68.00	Cukup Layak
68.01 – 84.00	Layak
84.01 – 100	Sangat Layak