

**LAPORAN PRAKTEK KERJA LAPANGAN**  
**SISTEM INFORMASI MANAJEMEN ASET PADA TOKO MYNnE**  
**COLLECTION BERBASIS WEBSITE**



**Oleh:**

**R. M. Ridho Vernanda Padantyo**  
**F1D016071**

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS MATARAM**  
**MARET 2021**

**HALAMAN PENGESAHAN**  
**LAPORAN PRAKTEK KERJA LAPANGAN**  
**SISTEM INFORMASI MANAJEMEN ASET PADA TOKO MYNE**  
**COLLECTION BERBASIS WEBSITE**

Oleh:

**R. M. Ridho Vernanda Padantyo**  
**F1D016071**

Telah disetujui oleh:

1. Dosen Pembimbing



Tanggal: 25 / 03 / 2021

**Andy Hidayat Jatmika, S.T., M.Kom.**  
**198312092012121001**

2. Pembimbing Lapangan



Tanggal: 25 / 03 / 2021

**Faisal, S.E., M.M.**  
**5201082603760002**

Mengetahui,  
Sekretaris Program Studi Teknik Informatika  
Fakultas Teknik  
Universitas Mataram



**Andy Hidayat Jatmika, S.T., M.Kom.**  
**19831209201212100**

## KATA PENGANTAR

Dengan penuh rasa hikmat, penulis mengucapkan puji syukur kepada Allah Yang Maha Esa atas segala kasih dan karunia-Nya sehingga penulis dapat menyelesaikan praktek kerja lapangan di Toko Myne Collection, Mataram.

Praktek Kerja Lapangan (PKL) dilakukan sebagai salah satu persyaratan kelulusan Sarjana S1 di Fakultas Teknik, Universitas Mataram. Penulis melakukan praktek kerja lapangan khusus mengenai sistem informasi manajemen aset pada Toko Myne Collection berbasis website..

Pada pelaksanaan kerja lapangan ini, penulis telah banyak mendapatkan bantuan dari berbagai pihak. Oleh karena itu, penulis ingin mengucapkan terima kasih kepada :

1. Bapak Andy Hidayat Jatmika, S.T., M.Kom. selaku dosen pembimbing praktek kerja lapangan.
2. Bapak Faisal, S.E., M.M. selaku pembimbing lapangan dan pemilik Toko Myne Collection.
3. Orang tua tercinta yang telah memberikan semangat, motivasi, dan dorongan agar praktek kerja lapangan ini dapat berjalan lancar hingga selesai.
4. Saudara Romi dan Ragil yang telah memberikan kontribusi besar di dalam pengerjaan sistem dan laporan praktek kerja lapangan ini.
5. Teman-teman HMB yang telah mensupport dalam aktivitas praktek kerja lapangan.
6. Shifa yang selalu memberikan motivasi, dorongan, dan dukungan dalam menyelesaikan praktek kerja lapangan ini.

Dengan tersusunnya laporan ini, semoga bisa bermanfaat bagi semua pembaca pada umumnya. Laporan ini mungkin jauh dari kesempurnaan, oleh karena itu, penulis sangat mengharapkan saran dan kritik dari pihak pembaca yang sifatnya membangun untuk kesempurnaan lebih lanjut dari laporan ini.

Penulis

# DAFTAR ISI

HALAMAN PENGESAHAN .....	ii
KATA PENGANTAR .....	iii
DAFTAR ISI.....	iv
DAFTAR TABEL.....	vi
DAFTAR GAMBAR.....	vii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	2
1.3. Batasan Masalah.....	2
1.4. Tujuan.....	2
1.5. Manfaat.....	2
BAB II TINJAUAN INSTANSI TEMPAT PKL .....	3
2.1. Profil Toko Myne Collection .....	3
2.2. Visi dan Misi Toko Myne Collection.....	3
2.3. Struktur Organisasi.....	4
BAB III LANDASAN TEORI.....	5
3.1. Sistem Informasi .....	5
3.2. Basis Data.....	5
3.3. Structured Query Language(SQL) .....	6
3.4. Aplikasi Web dan Web Server .....	6
3.5. PHP .....	7
3.6. UML.....	7
3.6.1. Use Case Diagram.....	7
3.6.2. Activity Diagram .....	9
3.6.3. Data Flow Diagram.....	10
3.6.4. Entity Relation Diagram .....	10
3.7. Sublime Text 3 .....	12
3.8. XAMPP .....	12
BAB IV PEMBAHASAN .....	13
4.1. Tahapan Pengembangan/Pembuatan Sistem Informasi Manajemen Aset .....	13
4.1.1. Analisa Sistem yang Ada .....	14
4.1.2. Analisa Kebutuhan Sistem.....	14

4.2.	Perancangan Desain .....	16
4.2.1.	Use case Diagram .....	16
4.2.2.	Entity Relationship Diagram (ERD).....	17
4.2.3.	Activity Diagram .....	19
4.3.	Desain Interface Sistem.....	23
4.4.	Implementasi Sistem .....	26
4.4.1.	Implementasi Database .....	26
4.4.2.	Implementasi Desain Interface .....	28
BAB V PENUTUP .....		34
5.1.	Kesimpulan.....	34
5.2.	Saran.....	34
DAFTAR PUSTAKA .....		35

## DAFTAR TABEL

Tabel 3.1. Tabel Simbol <i>Use Case Diagram</i> .....	8
Tabel 3.2 Tabel Simbol <i>Activity Diagram</i> .....	9
Tabel 3.3 Tabel Simbol <i>Data Flow Diagram</i> .....	10
Tabel 3.4. Tabel Simbol ERD.....	11
Tabel 4.1 Tabel admin .....	17
Tabel 4.2 tbl_transaksi.....	18
Tabel 4.3 Tbl_barang.....	18

## DAFTAR GAMBAR

<b>Gambar 2.1</b> Struktur Organisasi Myne Collection.....	4
<b>Gambar 3.1</b> Hubungan antara <i>client</i> dan server .....	7
<b>Gambar 4.1</b> Tahapan pengembangan sistem.....	14
<b>Gambar 4.2</b> <i>Use case diagram admin</i> .....	16
<b>Gambar 4.3</b> ER <i>Diagram</i> Sistem Kepegawaian.....	17
<b>Gambar 4.4</b> <i>Activity diagram</i> proses <i>login</i> .....	19
<b>Gambar 4.5</b> <i>Activity diagram</i> proses <i>input data</i> .....	20
<b>Gambar 4.6</b> <i>Activity diagram</i> proses edit data .....	20
<b>Gambar 4.7</b> <i>Activity diagram</i> proses hapus data .....	21
<b>Gambar 4.8</b> <i>Activity diagram</i> proses lihat detail data .....	22
<b>Gambar 4.9</b> <i>Activity diagram</i> proses tambah transaksi .....	22
<b>Gambar 4.10</b> Desain halaman <i>login</i> .....	23
<b>Gambar 4.11</b> Desain halaman <i>home</i> .....	23
<b>Gambar 4.12</b> Desain halaman <i>form</i> tambah barang .....	24
<b>Gambar 4.13</b> Desain halaman daftar aset.....	24
<b>Gambar 4.14</b> Desain halaman edit data.....	25
<b>Gambar 4.15</b> Desain halaman <i>form</i> transaksi.....	25
<b>Gambar 4.16</b> Desain halaman <i>profile</i> .....	26
<b>Gambar 4.17</b> Halaman <i>form</i> edit profile .....	26
<b>Gambar 4.18</b> Struktur tabel <i>admin</i> .....	27
<b>Gambar 4.19</b> Struktur tabel <i>tbl_transaksi</i> .....	27
<b>Gambar 4.20</b> Struktur tabel <i>tbl_barang</i> .....	27
<b>Gambar 4.21</b> Implementasi tampilan halaman <i>login</i> .....	28
<b>Gambar 4.22</b> Implementasi tampilan halaman <i>home</i> .....	29
<b>Gambar 4.23</b> Implementasi tampilan halaman <i>form</i> tambah barang.....	29
<b>Gambar 4.24</b> Implementasi tampilan halaman daftar aset .....	30
<b>Gambar 4.25</b> Implementasi tampilan halaman lihat data .....	31
<b>Gambar 4.26</b> Implementasi tampilan halaman <i>edit</i> aset.....	31
<b>Gambar 4.27</b> Implementasi tampilan halaman transaksi.....	32

<b>Gambar 4.28</b> Implementasi tampilan halaman <i>profile</i> .....	32
<b>Gambar 4.29</b> Implementasi tampilan halaman edit <i>profile</i> .....	33



# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Teknologi yang telah berkembang dengan pesat pada saat ini berdampak pada kehidupan sehari-hari, salah satunya pengolahan data dalam bidang perekonomian. Badan usaha milik pribadi merupakan salah satu contoh badan yang bergerak dalam bidang ekonomi. Dalam suatu badan usaha perlu adanya pengolahan data dan informasi yang tepat dan cepat. Salah satu solusi untuk memecahkan masalah dalam mengelola data dan informasi dalam suatu badan usaha adalah Sistem Informasi.

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan. Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan pun dibutuhkan. Sistem ini menyimpan, mengambil, mengubah, mengolah dan mengkomunikasikan informasi yang diterima dengan menggunakan sistem informasi atau peralatan sistem lainnya.

Toko Myne Collection merupakan badan usaha milik pribadi yang menjual boneka serta aksesoris yang menjadi kebutuhan tersier masyarakat dalam kehidupan sehari-hari seperti topi, sabuk, jam tangan dan lain-lain. Dalam menjalankan usaha tersebut banyak kegiatan perniagaan yang dilakukan dan salah satu contohnya yaitu pencatatan aset yang tersedia dalam toko tersebut. Pada pelaksanaannya Toko Myne Collection masih melakukan pencatatan aset secara manual dimana hal tersebut memakan banyak waktu dan sumber daya yang berdampak pada penjualan dan kinerja yang tidak maksimal para pegawainya. Informasi data barang masih sering tidak *valid* dan proses yang dijalankan masih secara manual dengan menggunakan aplikasi *Microsoft office excel*, sedangkan data barang yang dikelola sangat banyak. Berdasarkan permasalahan yang dihadapi di atas, maka dibuatlah sebuah sistem yaitu “Sistem Informasi Manajemen Aset pada Toko Myne Collection berbasis *Website*” untuk sistem internal dari toko tersebut sehingga mempermudah pengelolaan aset barang dari usaha tersebut.

## **1.2. Rumusan Masalah**

Bagaimana merancang dan membuat sistem informasi manajemen aset berbasis *website* yang dapat digunakan untuk mempermudah kinerja di bidang aset Toko Myne Collection?

## **1.3. Batasan Masalah**

Berdasarkan rumusan masalah di atas terdapat batasan-batasan masalah sebagai berikut:

1. Sistem informasi aset dibuat berdasarkan keperluan Toko Myne Collection.
2. Sistem informasi aset dibuat dengan berbasis *website*.
3. Pengguna dari sistem informasi antara lain adalah administrator yang dalam hal ini adalah pegawai dari bagian aset.
4. Sistem hanya sebatas pengolahan data aset.

## **1.4. Tujuan**

Merancang dan membuat sistem informasi manajemen aset berbasis *website* yang dapat digunakan untuk mempermudah dan mempercepat kinerja para pegawai dalam mengolah data aset.

## **1.5. Manfaat**

Manfaat yang didapatkan dari Praktek Kerja Lapangan ini dapat berguna bagi mahasiswa dan instansi yang bersangkutan.

1. Bagi Mahasiswa
  - a. Memberikan pengalaman kepada mahasiswa bagaimana menyelesaikan suatu pekerjaan dengan menerapkan keilmuan yang didapatkan pada bangku kuliah.
  - b. Memberikan wawasan serta pengalaman kepada mahasiswa mengenai dunia kerja sehingga mahasiswa mampu menerapkan pengalaman yang didapatkan di dunia kerja yang sebenarnya.
2. Bagi Badan Usaha
  - a. Memudahkan dan menunjang proses kerja khususnya sub bagian aset pada Toko Myne Collection dalam mengolah data aset.

## **BAB II**

### **TINJAUAN INSTANSI TEMPAT PKL**

#### **2.1. Profil Toko Myne Collection**

Toko Myne Collection adalah badan usaha milik pribadi yang berdiri sejak tahun 2017 yang menyediakan berbagai aksesoris seperti topi, ikat pinggang, jam tangan dan lain-lain. Usaha ini bersifat komersil, artinya barang-barang yang disediakan merupakan barang sehari-hari yang diperjual belikan secara umum pula. Jenis badan usaha ini yaitu CV (*Commanditaire Vennootschap*) atau Persekutuan Komoditer dimana usaha ini didirikan oleh dua orang atau lebih yang memiliki tanggung jawab masing-masing. Usaha ini didaftarkan dengan menggunakan akta yang telah disahkan pada kantor notaris di wilayah setempat. Toko Myne Collection terletak di Mataram, Nusa Tenggara Barat.

#### **2.2. Visi dan Misi Toko Myne Collection**

Adapun visi dan misi dari Toko Myne Collection adalah sebagai berikut :

a. Visi

“Menjadikan Toko Myne Collection sebagai referensi toko boneka dan aksesoris yang berkualitas dan terjangkau dengan mengedepankan pelayanan bagi konsumen”.

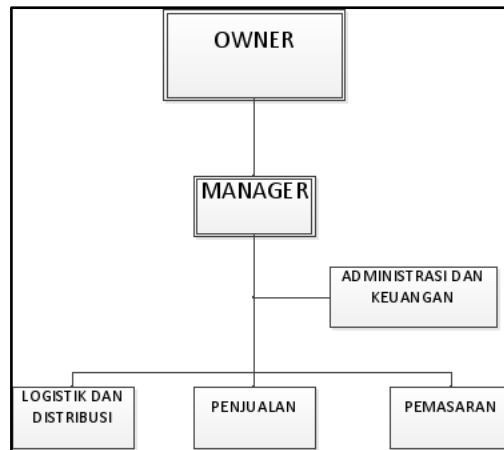
b. Misi

Misi adalah suatu rumusan pernyataan umum sebagai pemandu dalam mencapai tujuan inti organisasi. Misi dari Toko Myne Collection adalah:

1. Menyediakan barang-barang berkualitas dan terpilih bagi konsumen.
2. Memberikan standar pelayanan 3S (Senyum, Sapa, Salam) demi kenyamanan konsumen.
3. Membantu memenuhi kebutuhan tersier masyarakat lokal dengan menyediakan barang-barang dengan harga terjangkau.
4. Menyediakan produk-produk yang dapat diterima konsumen lokal dari berbagai kalangan.

### 2.3. Struktur Organisasi

Struktur organisasi Myne Collection dalam pelaksanaannya sebagai badan usaha memiliki struktur jabatan yang dapat dilihat pada Gambar 2.1.



**Gambar 2.1** Struktur Organisasi Myne Collection

## **BAB III**

### **LANDASAN TEORI**

#### **3.1. Sistem Informasi**

Sistem diartikan sebagai kumpulan dari komponen yang saling berkaitan untuk secara bersama-sama menghasilkan satu tujuan. Mengenai hirarki pengelompokkannya, dapat dikemukakan bahwa apabila suatu komponen didalam suatu sistem membentuk sistem sendiri maka komponen ini dinamakan subsistem dan seterusnya sehingga akan ada nama-nama modul, submodul, aplikasi dan subaplikasi. Hirarki ini berlaku relatif, tergantung dari jenjang manajerial manakah dimulainya. Informasi dilihat dari sumbernya dibagi menjadi dua jenis: internal dan eksternal. Informasi internal adalah informasi yang menggambarkan keadaan (profile), dan informasi eksternal adalah informasi yang menggambarkan ada tidaknya perubahan di luar organisasi itu. Informasi eksternal lebih banyak digunakan oleh kegiatan manajerial tingkat atas. Jenis informasi dibagi menjadi informasi insidental dan rutin. Informasi rutin digunakan secara periodic terjadwal dan digunakan untuk penanggulangan masalah-masalah rutin. Informasi insidental diperlukan untuk penanggulangan masalah-masalah khusus.

Pengertian sistem informasi dapat dilihat dari segi fisik dan fungsinya. Dari segi fisiknya dapat diartikan susunan yang terdiri dari perangkat keras, perangkat lunak dan tenaga pelaksanaannya yang secara bersama-sama saling mendukung untuk menghasilkan suatu produk. Sedangkan dari segi fungsi informasi merupakan suatu proses berurutan dimulai dari pengumpulan data dan diakhiri dengan komunikasi/desiminasi. Selanjutnya sistem informasi dikatakan berdaya guna jika mampu menghasilkan informasi yang baik, tinggi akurasi, tepat waktu, lengkap dan ringkas isinya. Akurasi adalah ukuran berupa rasio antara jumlah informasi yang benar dan tidak benar. Suatu sistem dikatakan mempunyai akurasi tinggi apabila akurasi sebesar 95%. Namun akurasi tinggi tidak akan berguna apabila kedatangannya terlambat dan tidak teratur. Oleh karena itu sistem informasi dituntut untuk lengkap, ringkas dan teratur sehingga tidak memusingkan pengguna informasi tersebut.

#### **3.2. Basis Data**

Basis data terdiri dari 2 kata, yaitu Basis dan Data. Basis kurang lebih diartikan sebagai markas atau gudang, tempat bersarang/berkumpul. Sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang,

hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi/suara, atau kombinasinya[1].

### 3.3. *Structured Query Language*(SQL)

Bahasa basis data terdiri atas *Data Definition Language* (DDL) dan *Data Manipulation Language* (DML). DDL merujuk pada kumpulan perintah yang dapat digunakan untuk mendefinisikan objek-objek basis data, seperti membuat tabel basis data atau indeks. Sedang DML mengacu pada kumpulan perintah yang dapat digunakan untuk melakukan manipulasi data, seperti penyimpanan data ke suatu tabel, lalu kemudian mengubahnya atau menghapusnya atau hanya sekedar menampilkannya kembali. DML inilah yang sebenarnya lebih dekat dengan *Query Language* (QL)[1].

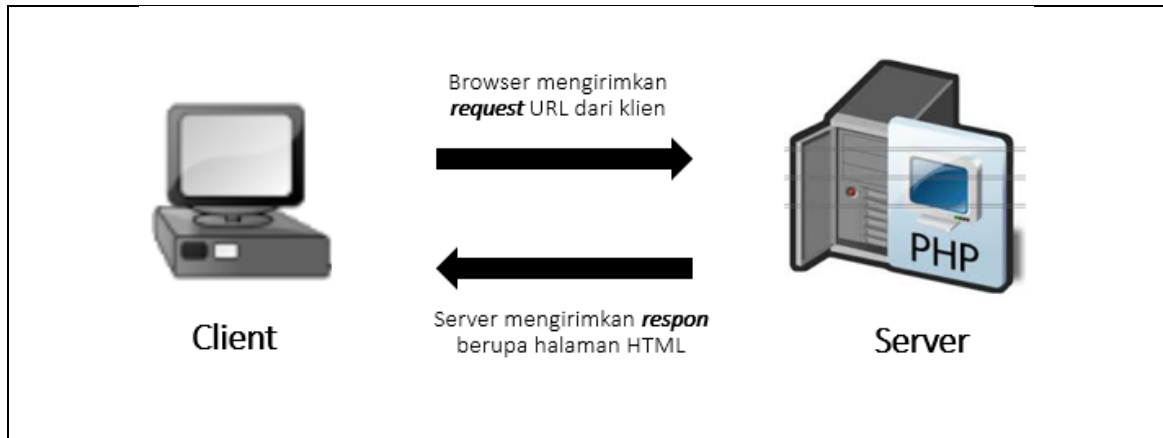
Kendati SQL merujuk pada *Query Language*, tidak berarti perintah-perintah yang menjadi standar hanyalah perintah-perintah yang berhubungan dengan query data saja. Selain mencakup DML, SQL juga dilengkapi dengan berbagai perintah yang tergolong DDL dan perintah kontrol transaksi (*transaction control*). Jadi SQL dapat diartikan sebagai bahasa standar yang digunakan untuk mengakses data di dalam database relasional.

### 3.4. Aplikasi Web dan Web Server

*World Wide Web* atau yang biasa disebut *web*, merupakan salah satu sumber daya internet yang berkembang pesat. Teknologi *web* memanfaatkan bahasa yang disebut HTML (*HyperText Markup Language*) dan protokol yang digunakan dinamakan HTTP (*HyperText Transfer Protocol*). Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML, seperti PHP, ASP, dan *applet* (Java)[2].

HTML merupakan sebuah bahasa *markup* (tanda) yang digunakan dalam membuat sebuah halaman *web*, serta menampilkan berbagai informasi di dalam sebuah *web browser*. HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman *web*.

*Webserver* merupakan *software* yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan *webbrowser*, serta mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen HTML. Dengan kata lain, sebuah *webserver* akan menunggu seorang klien untuk meminta *request* melalui *webbrowser* seperti Mozilla, Opera, Google Chrome, Internet Explorer dan program *browser* lainnya.



Gambar 3.1 Hubungan antara *client* dan server

### 3.5. PHP

PHP adalah salah satu bahasa pemrograman skrip yang dirancang untuk membangun aplikasi *web*. Aplikasi *web* adalah aplikasi yang disimpan dan dieksekusi (oleh PHP Engine) di lingkungan *webserver*. Setiap permintaan yang dilakukan oleh *user* melalui aplikasi *client* (*webbrowser*) akan direspon oleh aplikasi dan hasilnya akan dikembalikan lagi ke hadapan user. Dengan aplikasi *web*, halaman yang ditampilkan di layar *web browser* dapat bersifat dinamis, tergantung dari nilai data atau parameter yang dikirimkan oleh user ke *webserver*[3].

PHP disebut juga bahasa pemrograman *serverside* karena PHP diproses pada komputer *server*. Hal ini berbeda dibandingkan dengan bahasa pemrograman *client-side* seperti *JavaScript* yang diproses pada *web browser (client)*.

### 3.6. UML

UML (*Unified Modelling Language*) adalah sekumpulan pemodelan konvensi yang digunakan untuk menentukan atau menggambarkan sebuah system perangkat lunak dalam kaitannya dengan objek[2]. UML dapat juga diartikan sebuah bahasa grafik standar yang digunakan untuk memodelkan perangkat lunak berbasis objek.

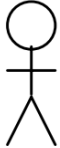


#### 3.6.1. Use Case Diagram

Suatu use case diagram menampilkan sekumpulan use case dan aktor (pelaku) dan hubungan diantara use case dan aktor tersebut. Use case diagram digunakan untuk penggambaran use case statik dari suatu sistem[4].




Use case diagram penting dalam mengatur dan memodelkan kelakuan dari suatu sistem. Use case menjelaskan apa yang dilakukan sistem (atau subsistem) tetapi tidak menspesifikasikan

cara kerjanya. Flow of event digunakan untuk menspesifikasikan kelakuan dari use case. Flow of event menjelaskan use case dalam bentuk tulisan dengan sejelas-jelasnya, diantaranya bagaimana, kapan use case dimulai dan berakhir, ketika use case berinteraksi dengan aktor, obyek apa yang digunakan, alur dasar dan alur alternatif. Tabel 3.1 menjelaskan simbol-simbol dan keterangan yang ada pada use case diagram :

Tabel 3.1. Tabel Simbol *Use Case Diagram*

Simbol	Nama	Keterangan
	<p><i>Actor</i></p>	<p><i>Actor</i> tersebut mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. Sebuah <i>actor</i> mungkin hanya memberikan informasi <i>input</i>-an pada sistem, hanya menerima informasi dari sistem atau keduanya menerima, dan memberi informasi pada sistem. <i>Actor</i> hanya berinteraksi dengan <i>use case</i>, tetapi tidak memiliki kontrol atas <i>use case</i>. <i>Actor</i> biasanya digambarkan dengan <i>stick man</i>. <i>Actor</i> dapat digambarkan secara umum atau spesifik, dimana untuk membedakannya digunakan suatu hubungan atau <i>relationship</i>.</p>
	<p><i>Use Case</i></p>	<p>Gambaran fungsionalitas dari suatu sistem, sehingga <i>customer</i> atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun.</p>
	<p><i>Association</i></p>	<p>Menghubungkan <i>link</i> antar <i>element</i>.</p>







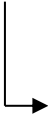
	<i>Include</i>	Kelakuan yang harus terpenuhi agar sebuah <i>event</i> dapat terjadi, dimana pada kondisi ini sebuah <i>use case</i> adalah bagian dari <i>use case</i> lainnya.
	<i>Dependency</i>	Sebuah <i>element</i> bergantung dalam beberapa cara ke <i>element</i> lainnya.
	<i>Generalization</i>	Biasa disebut juga <i>inheritance</i> atau pewarisan sebuah elemen dapat merupakan spesialisasi dari elemen lainnya.

### 3.6.2. Activity Diagram

Activity diagram adalah cara untuk menggambarkan aktivitas – aktivitas yang terdapat pada suatu sistem yang akan dibuat. Dengan activity diagram, alur dari kerja suatu sistem dapat dibuat dari awal hingga akhir. Activity diagram memperlihatkan siapa saja yang terlibat baik aktor maupun sistem serta memberikan keterangan proses dari tiap bagian tersebut. Pada Activity Diagram terdapat beberapa simbol dan beberapa cara penulisan diantaranya terlihat pada Tabel 3.2 sebagai berikut :

Tabel 3.2 Tabel Simbol *Activity Diagram*


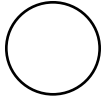

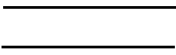
<b>Simbol</b>	<b>Nama</b>	<b>Keterangan</b>
	<i>Activity</i>	Menunjukkan sebuah aktivitas yang dilakukan
	<i>Initial Node</i>	Awal dari aktivitas suatu sistem atau program
	<i>Activity Final Node</i>	Akhir dari suatu aktivitas

	<i>Decision</i>	Menunjukkan sebuah pilihan
	<i>Line Connector</i>	Menghubungkan antara simbol yang satu dan yang lainnya

### 3.6.3. Data Flow Diagram

*Data flow diagram* adalah Data Flow Diagram (DFD) merupakan suatu cara atau metode untuk membuat rancangan sebuah sistem yang mana berorientasi pada alur data yang bergerak pada sebuah sistem nantinya. Simbol-simbol dan keterangan yang digunakan pada *data flow* diagram terlihat pada Tabel 3.3 sebagai berikut :

Tabel 3.3 Tabel Simbol *Data Flow Diagram*


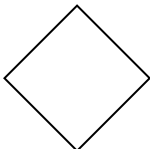

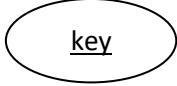

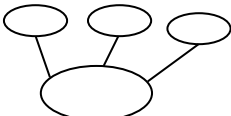

<b>Simbol</b>	<b>Nama</b>	<b>Keterangan</b>
	<i>External Entity</i>	Digunakan untuk menggambarkan asal atau tujuan data.
	Proses	Digunakan untuk memproses pengolahan data.
	<i>Data flow</i>	Digunakan untuk menggambarkan aliran data yang berjalan.
	<i>Data store</i>	Digunakan untuk menggambarkan data yang telah disimpan.

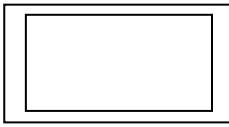
### 3.6.4. Entity Relation Diagram

ERD (Entity Relation Diagram) adalah cara untuk menggambarkan gambaran dari dunia nyata yang akan diterapkan pada suatu database sebuah system. ERD melihat objek nyata dapat sebagai sebuah entitas - entitas yang memiliki relasi antara entitas yang satu ataupun yang lain.

Dengan ERD sendiri dapat membantu mengurangi kesalahan – kesalahan dalam melakukan perancangan database dari gambaran dunia nyata dan struktur database seperti redundansi data, hubungan – hubungan antara entitas, dan lain sebagainya. Simbol-simbol dan keterangan yang digunakan pada ERD seperti yang ditunjukkan pada Tabel 3.5 berikut :

Tabel 3.4. Tabel Simbol ERD

Simbol	Nama	Keterangan
	<i>Entitiy</i>	<i>Entitiy</i> adalah objek dalam dunia nyata yang akan digambarkan dalam lingkungan pemakai
	<i>Relationship</i>	<i>Relationship</i> menunjukkan adanya hubungan antara entitas yang berbeda
	<i>Atribute</i>	<i>Atribute</i> memberikan deskripsi dari entitas
	<i>Key Atribute</i>	<i>Atribute</i> unik yang mewakili dari sebuah entitas
	<i>Multi Value Atribute</i>	<i>Atribute</i> yang dapat diisi dengan lebih satu nilai dengan jenis yang sama
	<i>Composite Atribute</i>	<i>Atribute</i> yang terdiri dari beberapa nilai yang dapat dipecah sesuai kebutuhan
	<i>Derived Atribute</i>	<i>Atribute</i> yang dihasilkan dari <i>atribute</i> yang lain

	<p><i>Weak Entity</i></p>	<p><i>Weak Entity</i> (Entitas Lemah) adalah entitas yang keberadaannya sangat bergantung dengan entitas lain dan bisa dihadirkan ataupun tidak.</p>
---	---------------------------	--

### 3.7. Sublime Text 3

*Sublime Text 3* adalah *text editor* yang sangat mudah digunakan dan didukung oleh banyak *plugin* yang mempermudah untuk menulis kode. Banyak sejumlah bahasa program yang ada pada aplikasi ini. Diantaranya PHP, CSS, C, C++, HTML, ASP, Java, dan sebagainya. *Sublime Text 3* dapat lebih memudahkan pekerjaan pengguna saat membuat sebuah program.

### 3.8. XAMPP

XAMPP adalah perangkat lunak bebas yang mendukung banyak sistem informasi, 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*[5].

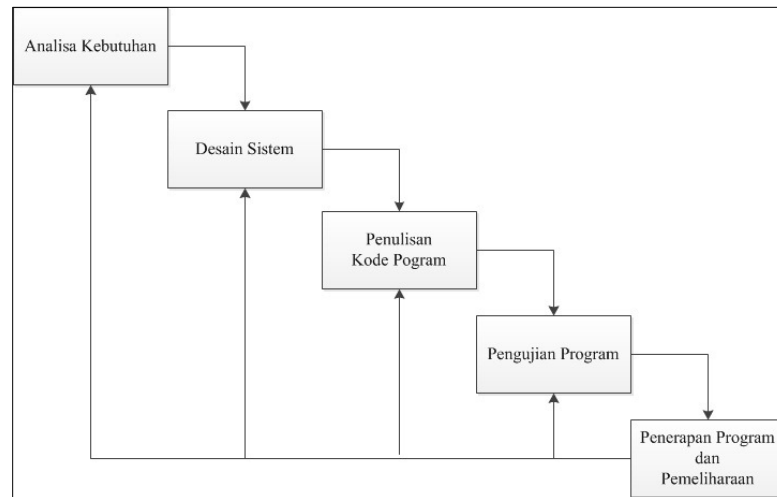
## BAB IV

### PEMBAHASAN

#### 4.1. Tahapan Pengembangan/Pembuatan Sistem Informasi Manajemen Aset

Model pengembangan yang digunakan pada pembuatan sistem informasi manajemen aset ini adalah model *Waterfall*. Model *waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software*[5]. Model ini melakukan pendekatan secara sistematis dan berurutan. Disebut dengan *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Model *waterfall* atau yang sering disebut model *classic life cycle* menunjukkan pengembangan perangkat lunak secara berurutan dan sistematis dimulai dari tahap analisis kebutuhan sistem lalu menuju ke tahap analisis, desain, *coding*, *testing/verification*, dan *maintenance*.

Dalam sebuah *project* yang bersifat OOP (*Object Oriented Programming*) perancangan sistem yang seperti *class* diagram, *use case* diagram, *activity* diagram dan *sequence* diagram harus ada. Untuk membuat perancangan sistem dilakukan analisis terhadap sistem yang akan dibuat. Hal yang pertama kali dilakukan adalah menganalisis *Use case* diagram untuk interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri. Selanjutnya menganalisis *activity* diagram pada sebuah sistem yang akan dirancang agar alur dari kerja suatu sistem dari awal hingga akhir dapat dimengerti oleh perancang sistem. Setelah itu buat perancangan untuk *sequence* diagram yang harus sesuai dengan *use case* yang ada pada sistem. Kemudian merancang *class* diagram untuk menjelaskan struktur dari program yang akan dibuat, biasanya *class* diagram dirancang pada sistem yang menggunakan konsep OOP. Perancangan dari segi *database* menggunakan penggambaran ERD juga erlu, untuk menggambarkan data-data yang ada dalam sistem. Gambar 4.1 menggambarkan tahapan pengembangan/ pembuatan dari sistem informasi manajemen aset pada Toko Myne *Collection* :



**Gambar 4.1**Tahapan pengembangan sistem

#### **4.1.1. Analisa Sistem yang Ada**

Proses pembuatan dan penyimpanan data yang berjalan saat ini khususnya ketika melakukan penambahan, pencarian, dan mengolah data barang atau aset yang ada di Toko Myne *Collection* masih menggunakan teknik manual yaitu dengan menggunakan aplikasi *Microsoft Excel* atau pun pembukuan dalam mengolah data barang sehingga proses pengolahan data masih berjalan lambat, data-data yang tidak beraturan, dan memakan waktu yang lama sehingga berpengaruh pada efisiensi pendataan barang itu sendiri.

#### **4.1.2. Analisa Kebutuhan Sistem**

Dalam menganalisa sebuah sistem dibutuhkan dua jenis kebutuhan. Kebutuhan fungsional dan kebutuhan non fungsional. Kebutuhan fungsional adalah kebutuhan yang berisi proses-proses apa saja yang nantinya dilakukan oleh sistem. Sedangkan kebutuhan non fungsional adalah kebutuhan yang berisi pada properti perilaku yang dimiliki oleh sistem.

##### **4.1.2.1. Kebutuhan Fungsional**

Kebutuhan fungsional adalah kebutuhan yang berisi proses-proses apa saja yang nantinya dilakukan oleh sistem. Kebutuhan fungsional sistem informasi manajemen aset pada Toko Myne *Collection* adalah sebagai berikut:

## **1. Data**

Data yang dibutuhkan selama proses perancangan/pembuatan sistem informasi manajemen aset adalah data barang-barang yang masuk, barang-barang yang terjual dan catatan transaksi.

## **2. Proses/ Fungsi**

Kegunaan dari pembuatan sistem manajemen aset ini adalah untuk memudahkan pegawai dalam melakukan pencatatan barang-barang yang ada di toko dalam mengolah dan mencari data barang. Di dalam sistem terdapat beberapa menu diantaranya menu *login*, *dashboard*, daftar aset, tambah barang, transaksi, dan *logout*. Sistem manajemen aset hanya dapat dikelola oleh pegawai pada Toko Myne *Collection*.

### **4.1.2.2. Kebutuhan Non-Fungsional**

Kebutuhan non-fungsional adalah kebutuhan yang berisi pada properti perilaku yang dimiliki oleh sistem. Kebutuhan non fungsional sistem informasi manajemen aset pada Toko Myne *Collection* adalah sebagai berikut:

#### **1. Alat**

Alat-alat yang digunakan dalam proses perancangan/ pembuatan sistem informasi manajemen aset adalah sebagai berikut:

- a. Sublime text 3
- b. Xampp
- c. Apache
- d. MySQL
- e. Web Browser (Google Chrome)
- f. Microsoft Visio 2016
- g. Microsoft Office 2016

#### **2. Bahan**

Bahan-bahan yang digunakan dalam proses perancangan/ pembuatan sistem informasi manajemen aset adalah sebagai berikut:

- a. HTML
- b. CSS
- c. Java Script

- d. PHP
- e. SQL

#### 4.2. Perancangan Desain

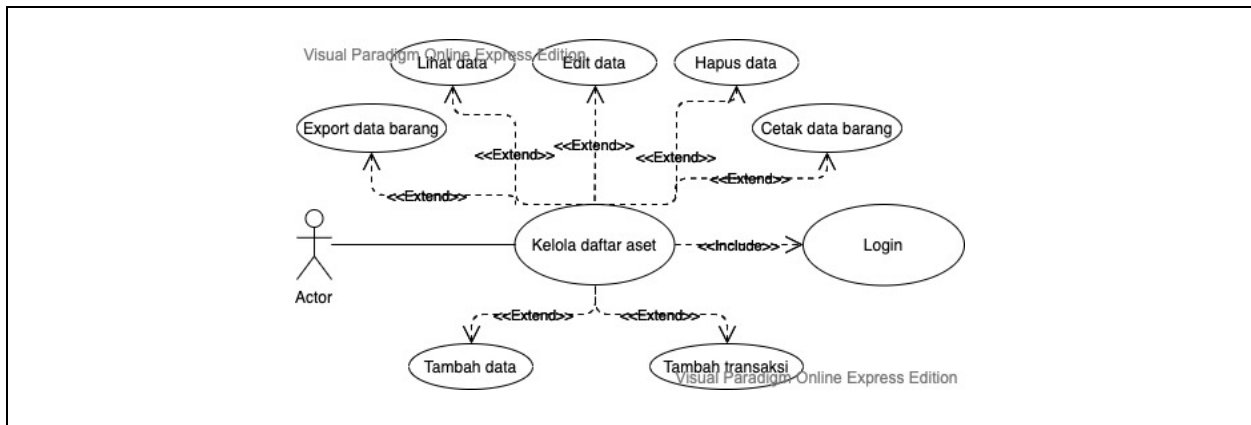
Perancangan desain sistem informasi manajemen aset terdiri dari *entity relationship diagram*, *use case diagram*, *activity diagram*, dan desain *interface* sistem.

##### 4.2.1. Use case Diagram

*Use case Diagram* dalam sistem informasi manajemen aset bertujuan untuk menjelaskan apa saja yang akan diperbuat atau dilakukan seorang *user* di dalam sistem atau dengan kata lain *use case* berfungsi merepresentasikan sebuah interaksi antara *user* dengan sistem. *User* dari sistem informasi manajemen aset pada Toko Myne *Collection* adalah:

###### 1. Admin (web)

Berikut *Use Case Diagram* dari sistem informasi manajemen aset dari pegawai:



**Gambar 4.2** Use case diagram admin

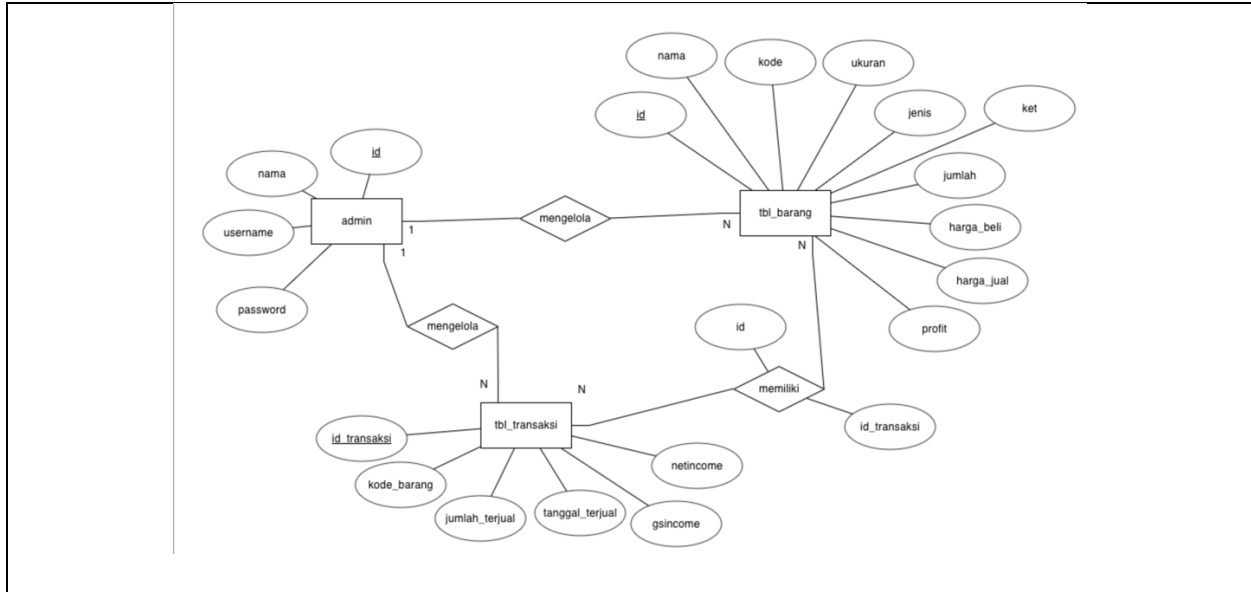
*Use case* diagram pada Gambar 4.2 merupakan gambaran dari aktifitas yang dapat dilakukan *admin* pada sistem informasi manajemen aset, bahwa *admin* dapat melakukan proses pengolahan data seperti berikut :

- a. Melakukan *login* dan *logout* dengan akun yang telah didaftarkan pada basis data.
- b. Mengelola data aset pada Toko Myne *Collection* meliputi menambah data aset, mengedit, menghapus, melihat data detail setiap aset, mencetak dan meng-*export* data aset dalam bentuk Excel. Selain itu sistem dapat menambah transaksi yang dilakukan, melihat pendapatan, dan mencetak laporan pendapatan.



#### 4.2.2. Entity Relationship Diagram (ERD)

Salah satu komponen utama dalam suatu sistem informasi adalah adanya suatu data atau basis data tidak terkecuali dalam Sistem Informasi Manajemen Aset Toko Myne *Collection*. Basis data dalam sistem ini dirancang dengan tabel – tabel yang saling berelasi satu sama lain atau biasa disebut *Entity Relationship Diagram*. *Entity relationship diagram* dari Sistem Informasi Manajemen Aset ini dapat digambarkan seperti pada Gambar 4.3 dibawah ini:



**Gambar 4.3** ER Diagram Sistem Kepegawaian

Berdasarkan Gambar 4.3, berikut ini adalah spesifikasi tabel dari keseluruhan *Entity Relationship Diagram* Sistem Informasi Manajemen Aset sebagai berikut:

1. Admin

Tabel 4.1 Tabel admin

	<b>Field</b>	<b>Tipe Data</b>	<b>Panjang</b>	<b>Key</b>
1	<u>Id</u>	Int	10	PK
2	Nama	Varchar	30	
2	Username	Varchar	30	
3	Password	Int	20	

## 2. Tbl\_transaksi

Tabel 4.2 tbl\_transaksi

	<i>Field</i>	<b>Tipe Data</b>	<b>Panjang</b>	<i>Key</i>
1	<u>Id_transaksi</u>	Int	11	PK
2	Kode_barang	Varchar	50	
3	Jumlah_terjual	Int	10	
4	Tanggal_terjual	Varchar	10	
5	Gsincome	Varchar	255	
6	Netincome	Varchar	255	

## 3. Tbl\_barang

Tabel 4.3 Tbl\_barang

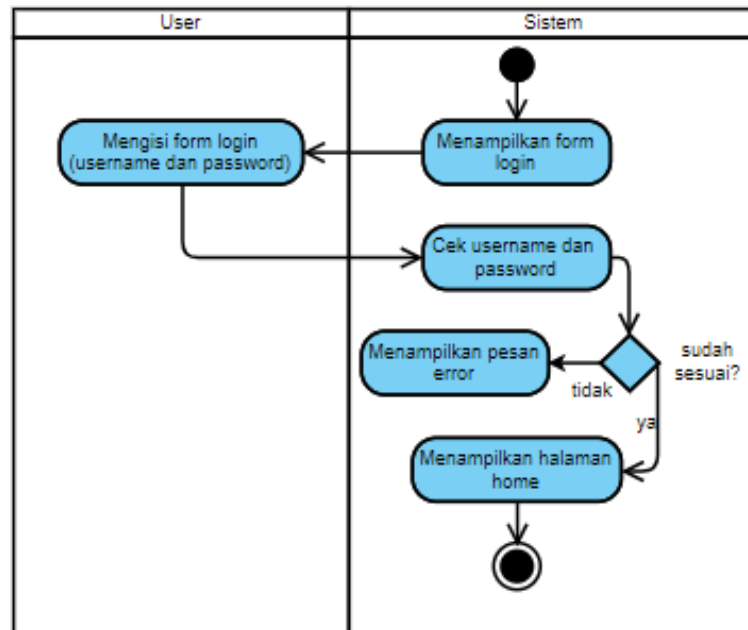
	<i>Field</i>	<b>Tipe Data</b>	<b>Panjang</b>	<i>Key</i>
1	<u>Id</u>	Int	10	PK
2	Nama	Varchar	50	
3	Kode	Varchar	50	
4	Ukuran	Varchar	20	
5	Jenis	Varchar	255	
6	Jumlah	Int	10	
7	Harga_beli	Int	20	
8	Harga_jual	Int	20	

9	Profit	Int	20	
10	Ket	Varchar	255	

### 4.2.3. Activity Diagram

#### 1. Activity Diagram Proses Login

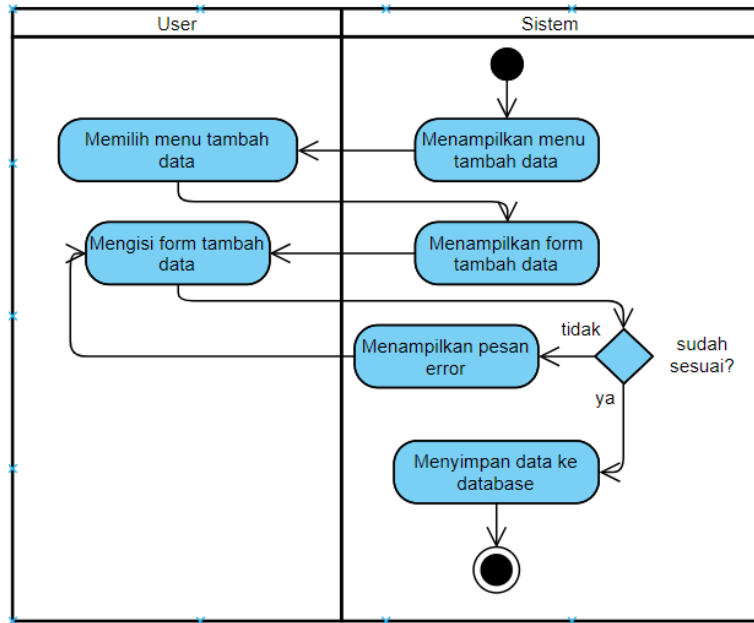
Pada sistem informasi aset ini *user (admin)* dapat melakukan *login* terlebih dahulu sebelum melakukan aktivitas yang lainnya terhadap sistem. Gambar 4.4 menjelaskan bagaimana aktivitas untuk masuk ke dalam sistem atau *login* sistem. Proses pertama adalah sistem akan menampilkan halaman *login*, kemudian *user (admin)* akan memasukkan *username* dan *password* dengan akun yang telah terdaftar didalam *database* sistem. Jika *login* berhasil, *user (admin)* akan dialihkan ke halaman utama/*home* sistem, jika gagal *user (admin)* sistem akan menampilkan pesan kesalahan kepada *user (admin)*.



**Gambar 4.4** Activity diagram proses login

#### 2. Menambah data

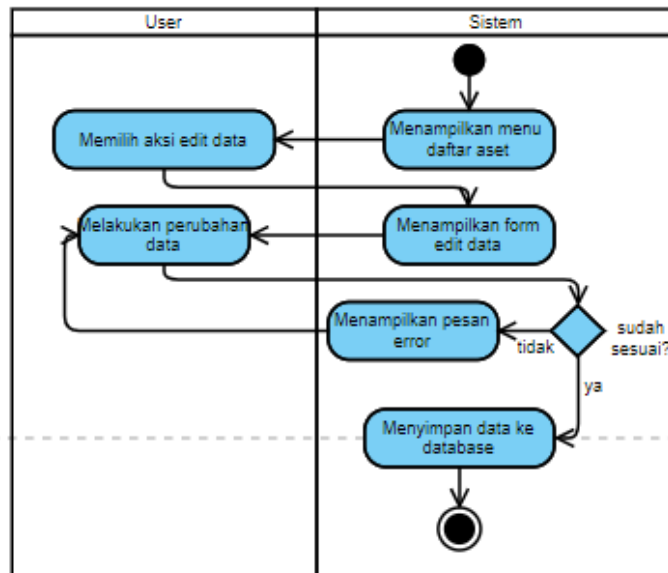
Pada sistem informasi aset ini, *user (admin)* dapat melakukan aktivitas penambahan data. Gambar 4.5 menjelaskan bagaimana aktivitas untuk menambah data pada sistem.



**Gambar 4.5** Activity diagram proses input data

### 3. Mengedit data

Pada sistem informasi aset ini, *user (admin)* dapat melakukan aktivitas edit data. Gambar 4.6 menjelaskan bagaimana aktivitas untuk mengedit data pada sistem.

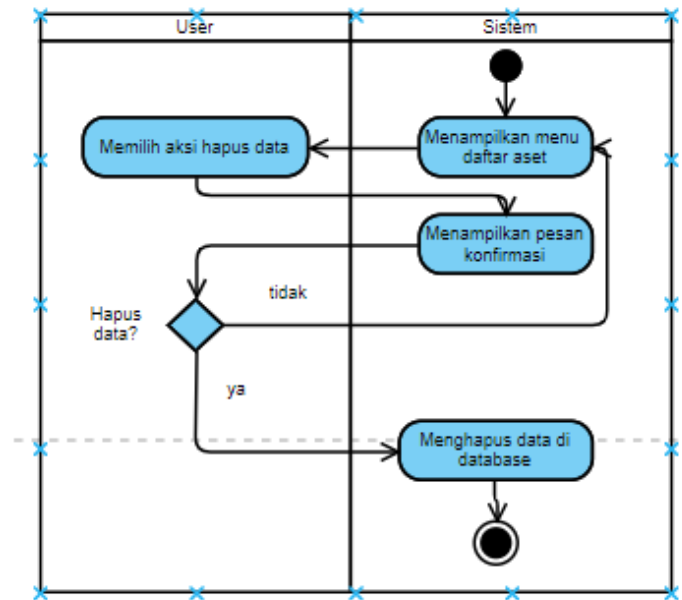


**Gambar 4.6** Activity diagram proses edit data

#### 4. Menghapus data

Pada sistem informasi aset, *user (admin)* dapat melakukan aktivitas hapus data.

Gambar 4.7 menjelaskan bagaimana aktivitas untuk menghapus data pada sistem.

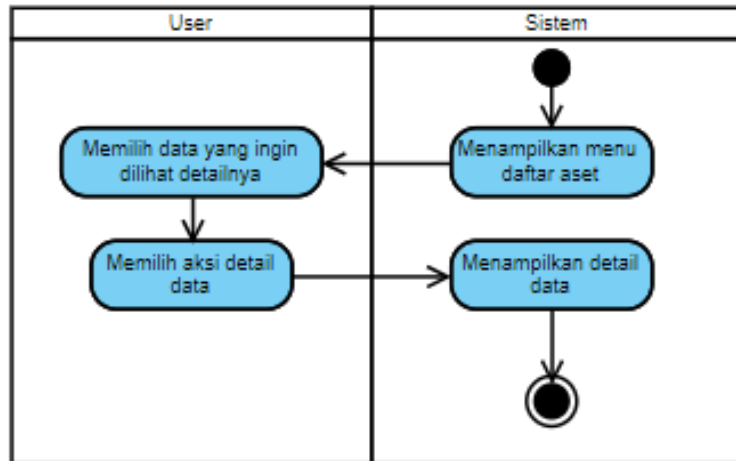


**Gambar 4.7** Activity diagram proses hapus data

#### 5. Melihat detail data

Pada sistem informasi aset, *user (admin)* dapat melakukan aktivitas melihat detail data.

Gambar 4.8 menjelaskan bagaimana aktivitas untuk melihat detail data pada sistem.

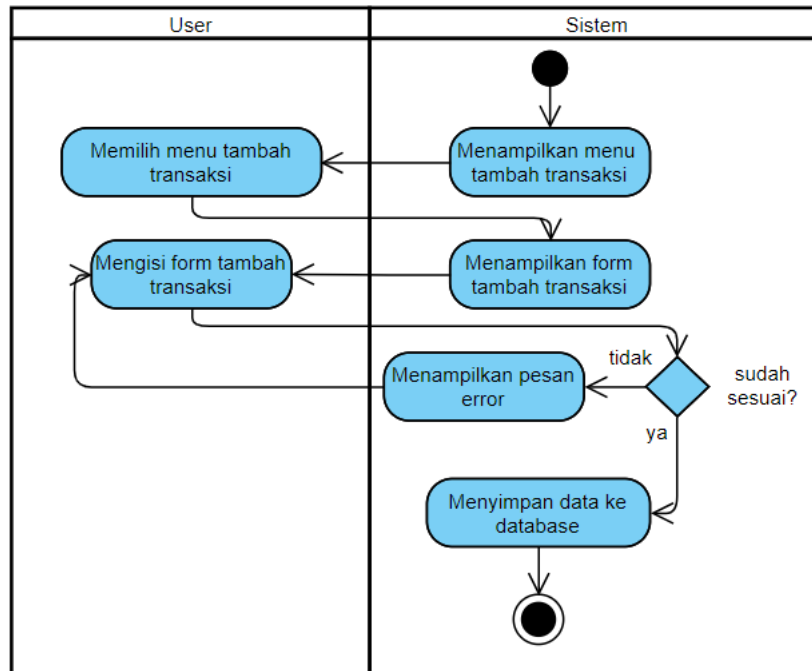


**Gambar 4.8** Activity diagram proses lihat detail data

#### 6. Menambah transaksi

Pada sistem informasi aset, *user (admin)* dapat melakukan aktivitas menambah transaksi.

Gambar 4.9 menjelaskan bagaimana aktivitas untuk melihat detail data pada sistem.



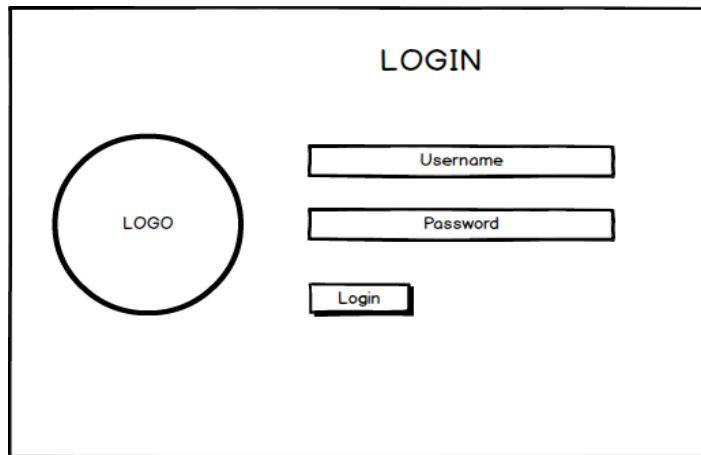
**Gambar 4.9** Activity diagram proses tambah transaksi

### 4.3. Desain Interface Sistem

#### 1. Tampilan Halaman Web

##### a. Halaman Login

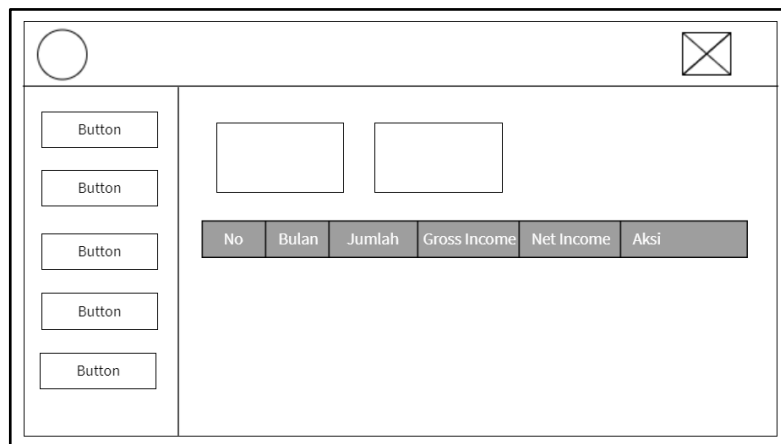
Halaman *login* merupakan halaman yang digunakan *user (admin)* untuk masuk ke dalam sistem. Halaman *login* terlihat seperti Gambar 4.9, pada halaman *login* terdapat 2 *field* yaitu *username/email* dan *password* yang harus diisi oleh *user (admin)*.



Gambar 4.10 Desain halaman *login*

##### b. Halaman Awal/Home

Halaman *home* menampilkan jumlah aset, jumlah transaksi, jumlah user, dan total transaksi setiap bulan seperti yang dapat dilihat pada Gambar 4.11. Pada halaman *home* di bagian sisi sebelah kiri terdapat *side bar* yang berisi menu-menu dari sistem ini.



Gambar 4.11 Desain halaman *home*

c. Halaman *Form* Tambah Barang

Halaman *form* tambah barang digunakan untuk menambahkan data aset ke dalam sistem. Pada halaman *form* tambah barang terdapat beberapa *field*/kolom isian yang harus diisi oleh *admin* seperti yang terlihat pada Gambar 4.12.

**Gambar 4.12** Desain halaman *form* tambah barang

d. Halaman Daftar Aset

Halaman daftar aset digunakan untuk menampilkan informasi tentang data aset. Pada halaman daftar aset seperti yang terlihat pada gambar 4.13, terdapat satu *field*/kolom aksi yang berisi beberapa tombol yang masing-masing tombol memiliki fungsi untuk melihat detail data, edit data, dan hapus data pada setiap aset yang telah dimasukkan. Pada bagian atas kanan terdapat dua fitur tambahan, yang pertama cetak yang berfungsi untuk mencetak semua data aset, dan yang kedua data lengkap yang berfungsi untuk meng-*export* data lengkap aset ke dalam *Microsoft Excel*.

**Gambar 4.13** Desain halaman daftar aset



e. Halaman *Edit Data*

Halaman *edit* data digunakan untuk mengubah data suatu aset. Pada halaman *edit* data, *form* akan sudah terisi sesuai dengan data yang terdapat di dalam basis data. Pengguna hanya perlu mengganti data ingin dirubah.

The wireframe shows a window with a title bar containing a circle icon on the left and a close button (X) on the right. The main content area is divided into two sections. On the left is a sidebar with five buttons, each labeled 'Button'. On the right is a form area with a title bar labeled 'Title'. Below the title bar are four text input fields, each labeled 'Text', arranged vertically. At the bottom of the form area is a button labeled 'Button'.

**Gambar 4.14** Desain halaman edit data

f. Halaman *Form Transaksi*

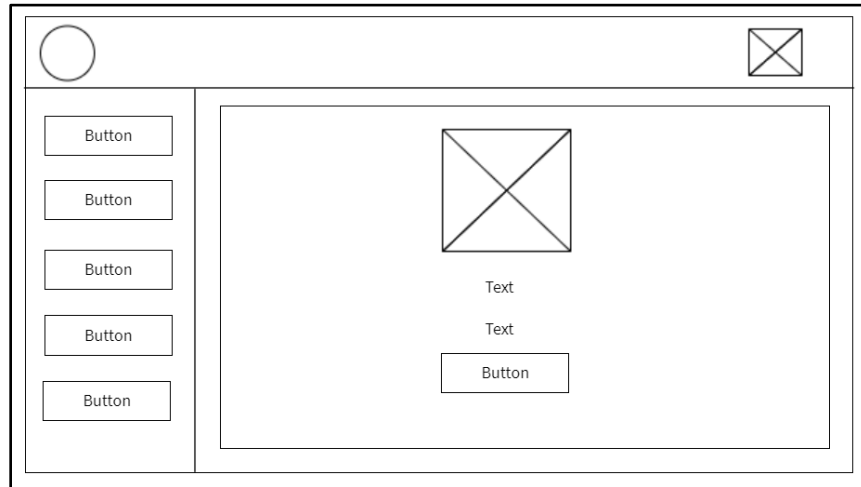
Halaman *form* transaksi digunakan untuk memasukkan transaksi pembelian barang. Pada halaman *form* transaksi, terdapat tabel yang berisi kode barang, nama barang, *stock*, harga, jumlah terjual, dan tanggal terjual.

The wireframe shows a window with a title bar containing a circle icon on the left and a close button (X) on the right. The main content area is divided into two sections. On the left is a sidebar with five buttons, each labeled 'Button'. On the right is a form area with a title bar labeled 'Title'. Below the title bar are four text input fields, each labeled 'Text', arranged vertically. At the bottom of the form area is a button labeled 'Button'.

**Gambar 4.15** Desain halaman *form* transaksi

g. Halaman *Profile*

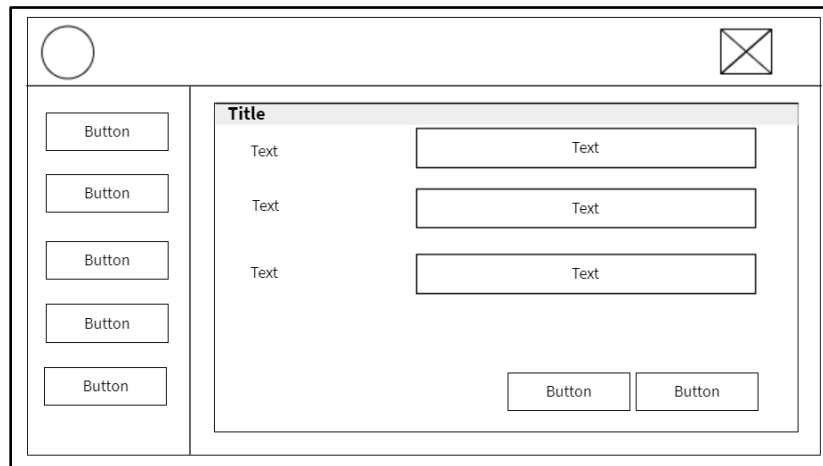
Halaman *profile* ini menampilkan informasi mengenai profil *admin* sistem. Pada bagian bawah halaman terdapat tombol edit profile yang digunakan untuk melakukan perubahan pada profile *account* dari *admin* sistem.



**Gambar 4.16** Desain halaman *profile*

h. Halaman *Form Edit Profile*

Pada halaman *edit profile* ini terdapat *form* yang digunakan untuk mengedit data *account* pada sistem diantaranya nama, *username*, dan *password* seperti yang terlihat pada Gambar 4.17.



**Gambar 4.17** Halaman *form* edit profile

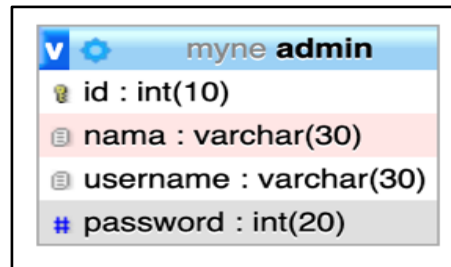
## 4.4. Implementasi Sistem

### 4.4.1. Implementasi Database

Berikut adalah hasil implementasi *database* dari Sistem Informasi Manajemen Aset Toko Myne Collection. *Database* dibuat dengan menggunakan aplikasi MySQL (*xampp*), nama

database adalah “myne” dan terdiri dari 3 tabel yaitu tabel “admin”, tabel “tbl\_transaksi”, dan tabel “tbl\_barang”. Berikut hasil implementasi database sistem :

1. Tabel *Admin*



The screenshot shows the structure of the 'myne admin' table. It has four columns: 'id' (int(10), primary key), 'nama' (varchar(30)), 'username' (varchar(30)), and 'password' (int(20)).

Field	Type
id	int(10)
nama	varchar(30)
username	varchar(30)
password	int(20)

Gambar 4.18 Struktur tabel *admin*

2. Tabel *Tbl\_transaksi*

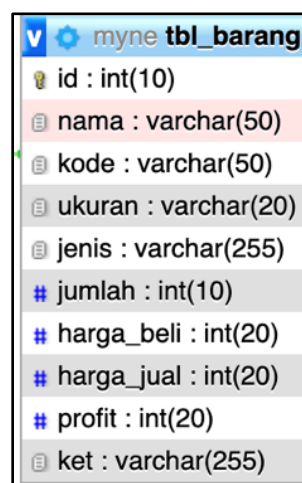


The screenshot shows the structure of the 'myne tbl\_transaksi' table. It has six columns: 'id\_transaksi' (int(11), primary key), 'kode\_barang' (varchar(50)), 'jumlah\_terjual' (int(10)), 'tanggal\_terjual' (varchar(10)), 'gsincome' (varchar(255)), and 'netincome' (varchar(255)).

Field	Type
id_transaksi	int(11)
kode_barang	varchar(50)
jumlah_terjual	int(10)
tanggal_terjual	varchar(10)
gsincome	varchar(255)
netincome	varchar(255)

Gambar 4.19 Struktur tabel *tbl\_transaksi*.

3. Tabel *tbl\_barang*



The screenshot shows the structure of the 'myne tbl\_barang' table. It has ten columns: 'id' (int(10), primary key), 'nama' (varchar(50)), 'kode' (varchar(50)), 'ukuran' (varchar(20)), 'jenis' (varchar(255)), 'jumlah' (int(10)), 'harga\_beli' (int(20)), 'harga\_jual' (int(20)), 'profit' (int(20)), and 'ket' (varchar(255)).

Field	Type
id	int(10)
nama	varchar(50)
kode	varchar(50)
ukuran	varchar(20)
jenis	varchar(255)
jumlah	int(10)
harga_beli	int(20)
harga_jual	int(20)
profit	int(20)
ket	varchar(255)

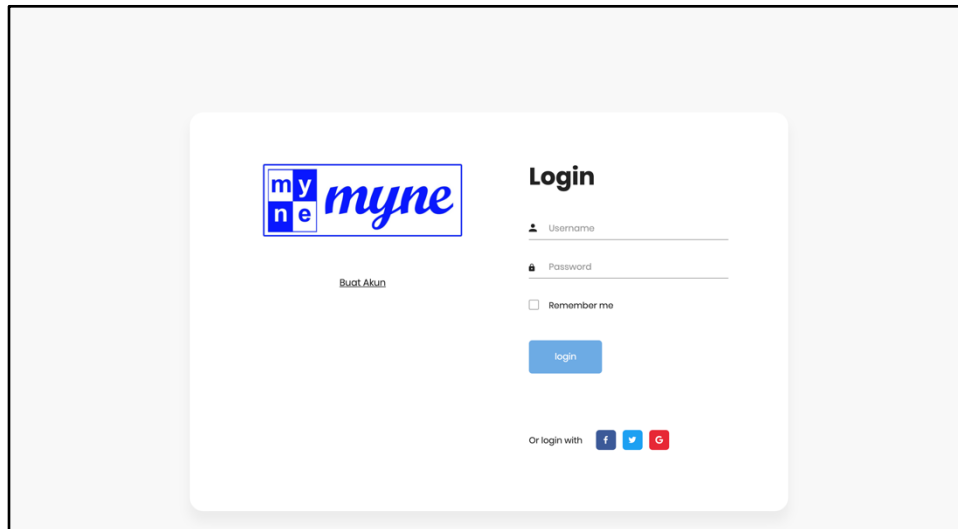
Gambar 4.20 Struktur tabel *tbl\_barang*

## 4.4.2. Implementasi Desain Interface

### 1. Tampilan Halaman *Web*

#### a. Halaman *Login*

Halaman *login* digunakan untuk *user (admin)* untuk masuk ke dalam sistem. Pada halaman *login* terdapat 2 *field*/kolom isian yaitu *username* dan *password* yang harus diisi oleh *user (admin)*. Setelah *user* mengklik tombol *Login*, jika akun yang dimasukkan benar, maka *user* akan diarahkan ke halaman *home* seperti pada Gambar 4.28, jika salah maka akan ditampilkan pesan kesalahan.

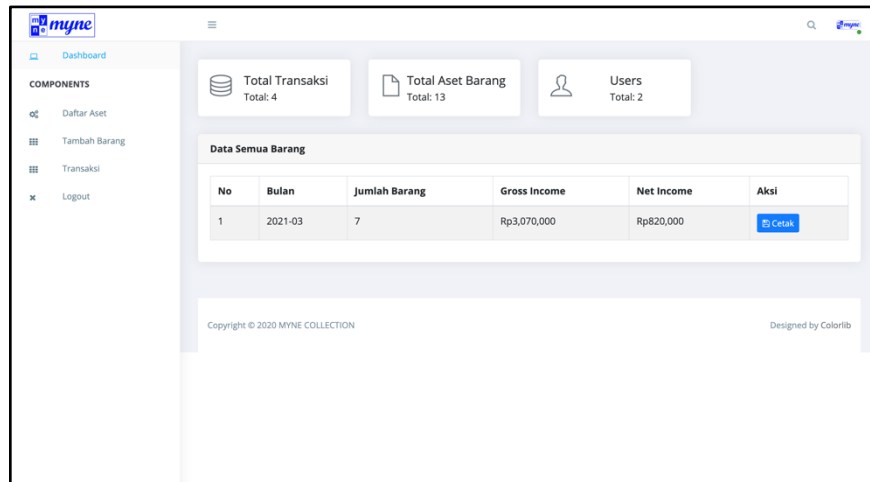


**Gambar 4.21** Implementasi tampilan halaman *login*

#### b. Halaman Awal/*Home*

Halaman *home* pada sisi *admin* menampilkan data penjualan barang setiap bulan. Terdapat tiga kotak kecil yaitu total aset lengkap, total transaksi, dan jumlah user seperti yang terlihat pada Gambar 4.22. Pada bagian kiri terdapat lima buah menu yaitu *dashboard*, daftar aset, tambah barang, transaksi, dan *logout*. Jika *admin* memilih menu *Dashboard* maka *admin* akan dialihkan ke halaman *home* seperti Gambar 4.22. Jika *admin* memilih menu daftar aset maka *admin* akan dialihkan ke halaman Daftar aset. Jika *admin* memilih Tambah barang, maka *admin* akan dialihkan ke halaman *form* tambah barang. Jika *admin* memilih menu transaksi maka *admin* akan dialihkan ke halaman *form* transaksi. Jika *admin* memilih pada atas kanan sistem *profile*, maka *admin* akan diberikan satu sub menu pada *profile* yaitu *profile* dan *logout*. *Profile* berfungsi untuk melihat *profile* lengkap dari *user*.

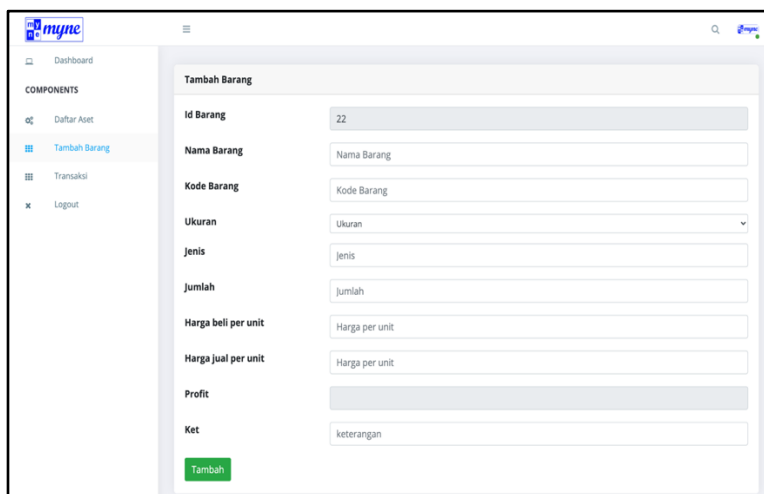
Tabel data semua barang menampilkan informasi penjualan barang setiap bulannya. Informasi yang ditampilkan yaitu total barang terjual, *gross income*, dan *net income*. Selain itu, terdapat juga tombol cetak yang berfungsi untuk mencetak data penjualan di tiap bulan.



**Gambar 4.22** Implementasi tampilan halaman *home*

c. Halaman Tambah Barang

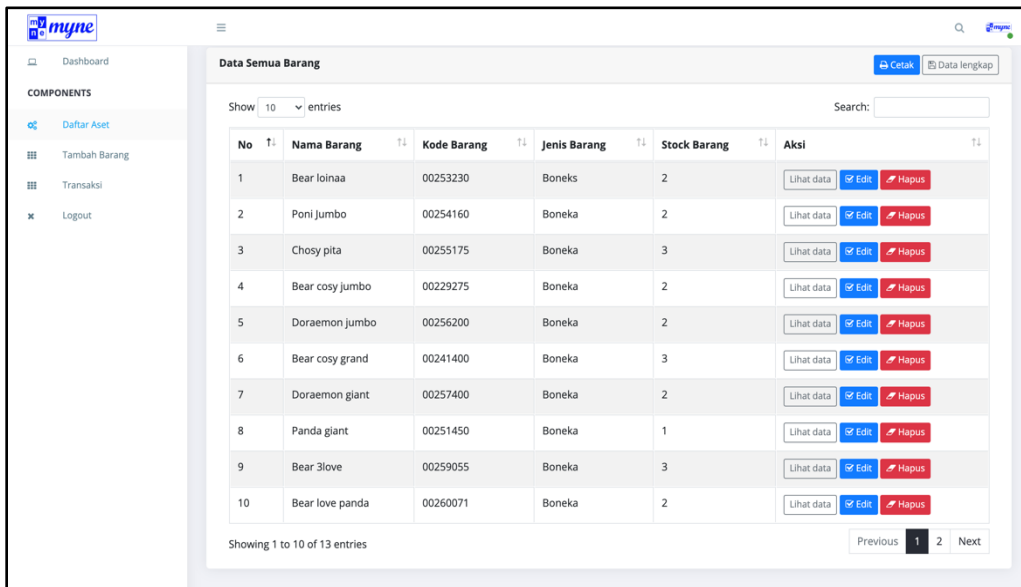
Halaman tambah barang digunakan untuk menambahkan data-data aset dimana pada *form* tambah barang terdapat total 10 *field*. Tujuh diantaranya harus diisi oleh *admin*, dua *field* otomatis terisi oleh sistem, dan satu *field* bersifat opsional. Apabila terdapat salah satu *field* pada *field* yang wajib terisi namun belum terisi, maka akan diberikan notifikasi bahwa masih ada *field* yang kosong atau belum terisi. Sebaliknya apabila semua *field* telah terisi, *admin* dapat meng-klik tombol simpan untuk menyimpan data-data yang telah diisi ke *database*.



**Gambar 4.23** Implementasi tampilan halaman *form* tambah barang

#### d. Halaman Data Aset

Halaman Daftar Aset digunakan untuk menampilkan data-data aset dalam bentuk tabel. Data-data yang ditampilkan hanya beberapa data saja seperti nama, kode, jenis dan jumlah stok barang. Di kolom aksi terdapat tombol lihat data, edit dan hapus yang dimana pada masing-masing tombol memiliki fungsi. Pada tombol lihat data jika *admin* memilihnya, maka *admin* akan dialihkan ke halaman detail dari aset yang dipilih. Pada tombol edit jika *admin* memilihnya, maka *admin* akan dialihkan ke halaman *form* edit aset. Dan pada tombol hapus jika *admin* memilihnya, maka *record* data barang yang dipilih akan terhapus dari *database* sistem. Pada pojok kiri atas tabel terdapat *numbering*, jika admin memilih jumlah 10, maka data yang ditampilkan sebanyak 10 data. Pada pojok kanan atas terdapat tombol cetak untuk mencetak seluruh data aset dalam sistem, dan tombol data lengkap untuk meng-*export* semua data aset ke dalam *Microsoft Excel*. Pada kanan bawah tabel terdapat *paging* tabel, dimana jumlah data yang ada dibagi sesuai jumlah data yang diinginkan untuk ditampilkan. Ketika *admin* telah memilih jumlah data yang ingin ditampilkan pada fitur yang ada di kiri atas tabel contoh admin memilih 10 data untuk ditampilkan, maka jumlah data yang ada akan dibagi menjadi 10 data sesuai yang dipilih. Sehingga pada *paging* tabel ini *admin* dapat melakukan *next* atau *previous* data. Jika *admin* memilih *next* maka akan ditunjukan ke data selanjutnya. Jika admin memilih *previous*, maka akan ditunjukan ke data sebelumnya.

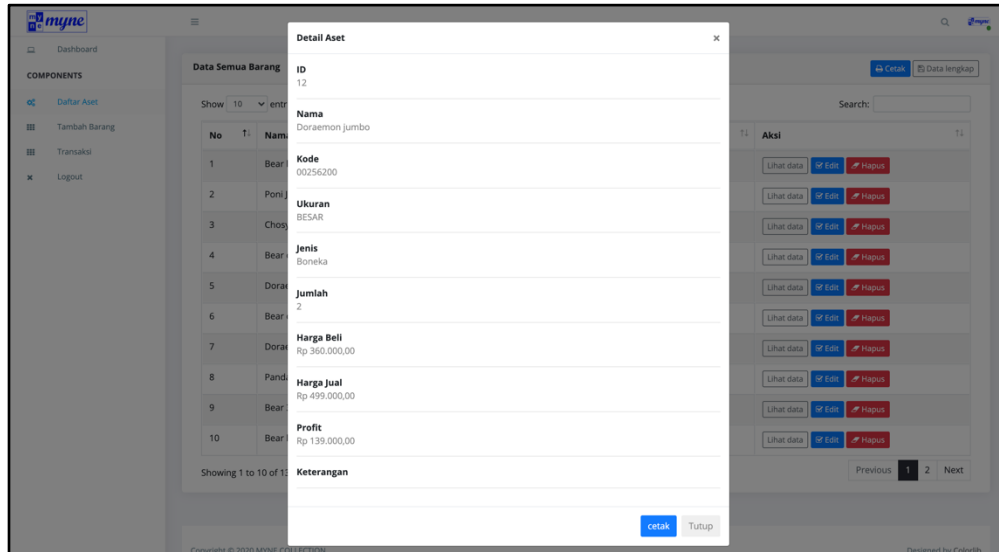


No	Nama Barang	Kode Barang	Jenis Barang	Stock Barang	Aksi
1	Bear Iolnaa	00253230	Boneks	2	Lihat data Edit Hapus
2	Poni Jumbo	00254160	Boneka	2	Lihat data Edit Hapus
3	Chosy pita	00255175	Boneka	3	Lihat data Edit Hapus
4	Bear cosy jumbo	00229275	Boneka	2	Lihat data Edit Hapus
5	Doraemon jumbo	00256200	Boneka	2	Lihat data Edit Hapus
6	Bear cosy grand	00241400	Boneka	3	Lihat data Edit Hapus
7	Doraemon giant	00257400	Boneka	2	Lihat data Edit Hapus
8	Panda giant	00251450	Boneka	1	Lihat data Edit Hapus
9	Bear 3love	00259055	Boneka	3	Lihat data Edit Hapus
10	Bear love panda	00260071	Boneka	2	Lihat data Edit Hapus

Gambar 4.24 Implementasi tampilan halaman daftar aset

e. Halaman Lihat Data

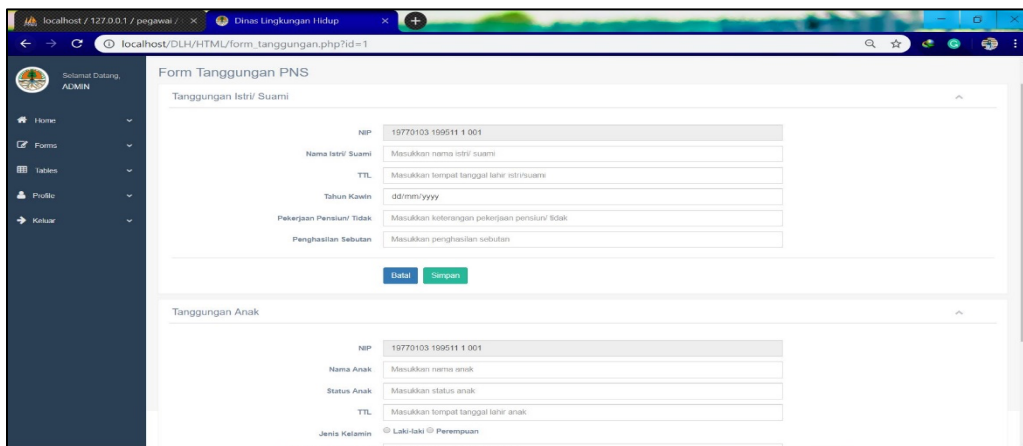
Halaman Lihat Data digunakan untuk menampilkan informasi lengkap dari data aset yang dipilih. Halaman ini ditampilkan dalam bentuk *pop-up* di halaman yang sama yaitu halaman daftar aset.



Gambar 4.25 Implementasi tampilan halaman lihat data

f. Halaman *Edit* Data

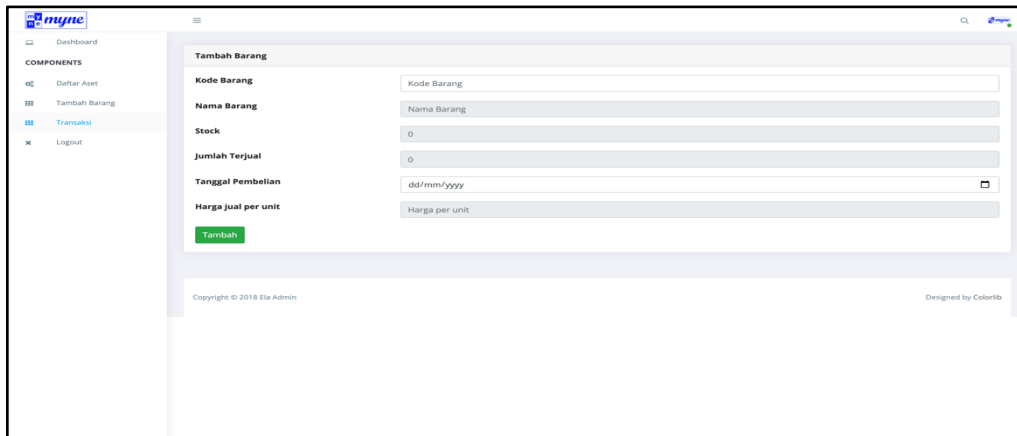
Halaman edit aset digunakan untuk mengubah/memperbarui data aset. Pada halaman tersebut terdapat beberapa *field* yang sudah terisi dan bisa diubah oleh *admin*. Terdapat tombol simpan yang jika *admin* memilih tombol simpan maka data akan diperbarui ke dalam *database* sistem serta *admin* akan dialihkan ke halaman tabel data aset.



Gambar 4.26 Implementasi tampilan halaman *edit* aset

#### g. Halaman Transaksi

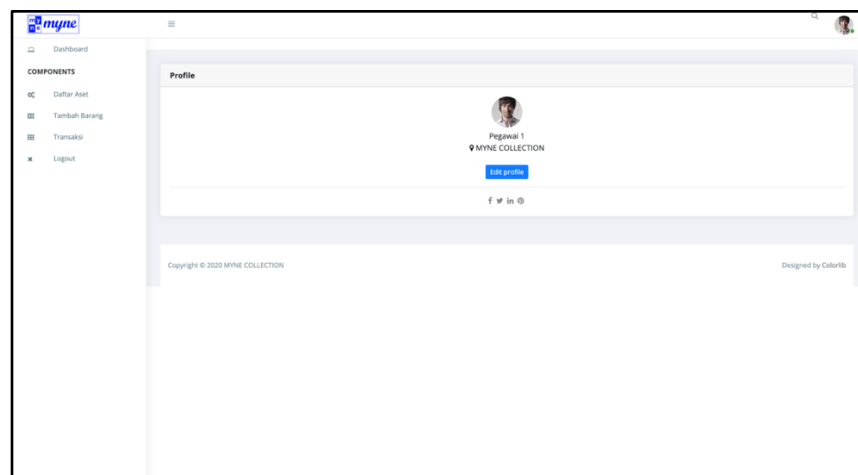
Halaman transaksi digunakan untuk menambahkan data transaksi penjualan barang dimana pada *form* transaksi terdapat total 6 *field*. *Admin* hanya perlu memasukkan kode barang ke dalam *field* yang tersedia, lalu sistem akan menampilkan secara otomatis nama barang, jumlah stok yang tersedia, dan harga jual per unit. Setelah itu, *admin* memasukkan jumlah barang yang terjual beserta tanggal transaksinya. Jumlah barang yang terjual tidak bisa melebihi jumlah stok barang yang tersedia yang telah diatur oleh sistem. Apabila semua *field* telah terisi, *admin* dapat mengklik tombol simpan untuk menyimpan data-data yang telah diisi ke *database*.

The image shows a web application interface for adding a new item. On the left is a sidebar with a 'COMPONENTS' menu containing 'Dashboard', 'Daftar Asat', 'Tambah Barang', 'Transaksi', and 'Logout'. The main content area is titled 'Tambah Barang' and contains a form with the following fields: 'Kode Barang' (text input), 'Nama Barang' (text input), 'Stock' (text input with '0'), 'Jumlah Terjual' (text input with '0'), 'Tanggal Pembelian' (date picker set to 'dd/mm/yyyy'), and 'Harga jual per unit' (text input). A green 'Tambah' button is located below the form. At the bottom of the page, there is a copyright notice 'Copyright © 2018 Ela Admin' and 'Designed by Colorlib'.

**Gambar 4.27** Implementasi tampilan halaman transaksi

#### h. Halaman Profile

Halaman *profile* berfungsi untuk menampilkan informasi mengenai *profile* dari *user*. Pada bagian bawah halaman terdapat tombol edit profile yang digunakan untuk melakukan perubahan pada akun profil *user*.

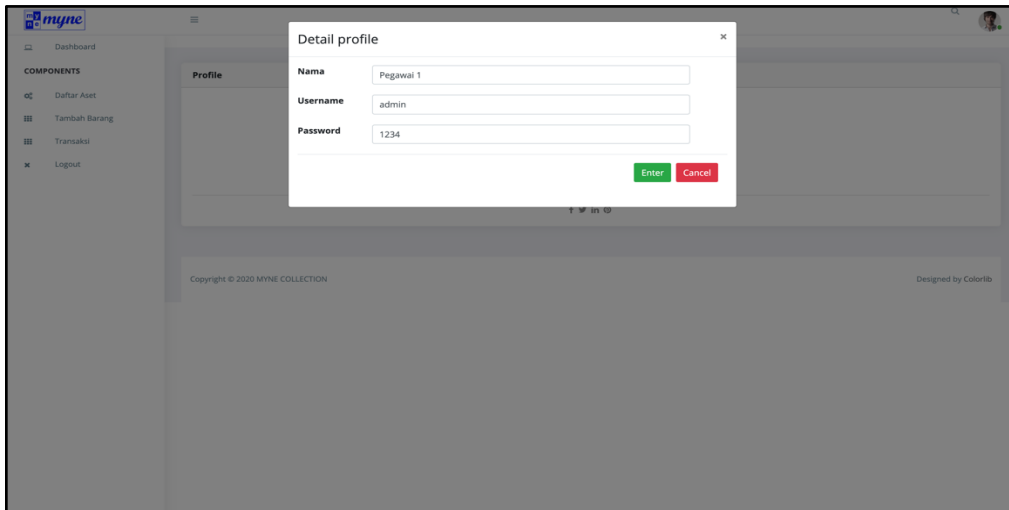
The image shows a user profile page. The sidebar is the same as in the previous screenshot. The main content area is titled 'Profile' and features a user profile card. The card includes a circular profile picture, the name 'Pegawai 1', and the organization 'MYNE COLLECTION'. Below the name is a blue 'Edit profile' button and social media icons for Facebook, WhatsApp, and LinkedIn. At the bottom of the page, there is a copyright notice 'Copyright © 2020 MYNE COLLECTION' and 'Designed by Colorlib'.

**Gambar 4.28** Implementasi tampilan halaman *profile*



i. Halaman *Edit Profile*

Halaman *edit profile* berfungsi untuk melakukan perubahan pada akun profil *user*. Halaman ini *ditampilkan* dalam bentuk *pop-up* dan terdapat tiga *field* di dalamnya yaitu *field* nama, *username*, dan *password*. Ketiga *field* tersebut dapat diganti sesuai dengan kebutuhan. Tombol *Enter* pada bagian bawah berfungsi untuk menyimpan data yang telah dirubah ke dalam *database*.



**Gambar 4.29** Implementasi tampilan halaman edit *profile*

## **BAB V**

### **PENUTUP**

#### **5.1. Kesimpulan**

Kesimpulan yang dapat diambil dari sistem informasi manajemen aset Toko Myne Collection ini antara lain sebagai berikut:

1. Rancangan Sistem Informasi Manajemen Aset Toko Myne Collection berbasis *web* dibuat berdasarkan hasil analisis terhadap kebutuhan toko untuk yang kemudian akan disajikan dalam bentuk yang lebih informatif kepada pegawai untuk dapat dimanfaatkan guna menunjang kebutuhan operasional.
2. Sistem informasi aset berbasis web ini digunakan oleh *admin* untuk mengelola data aset baik itu melihat, memasukkan, merubah, dan menghapus.

#### **5.2. Saran**

Sistem Informasi ini masih belum bisa dikatakan sempurna, terdapat beberapa saran yang diharapkan dapat menjadi perhatian untuk pengembangan kedepannya.

1. Pada sistem informasi aset berbasis *web* sekarang ini hanya dikembangkan dengan menggunakan PHP *native*, kedepannya diharapkan dapat dikembangkan dengan menggunakan *framework* seperti *Code Igniter, Laravel*, atau lainnya.
2. Diharapkan sistem ini dibuat lebih kompleks, tidak hanya pengelolaan aset tetapi juga pengelolaan data yang lain.
3. Dalam melakukan pengembangan sistem alangkah baiknya untuk menganalisa secara mendetail kebutuhan terlebih dahulu sehingga sistem yang dibangun sesuai dengan spesifikasi dan mudah dalam melakukan *maintenance* kedepannya seperti melakukan normalisasi *tabel* dalam *database*.

## DAFTAR PUSTAKA

- [1] Fathansyah, *Basis Data*. Bandung: Informatika, 2007.
- [2] H. Hidayat, *Cara Instan Menguasai Pemrograman Website Secara Instan*. Jakarta Barat: Agogos Publishing, 2011.
- [3] M. Fowler, *UML Distilled*, 3rd ed. Yogyakarta: Andi, 2005.
- [4] G. Booch, J. Rumbaugh, and I. Jacobson, *The Unified Modeling Language User Guide Second Edition*. Addison-Wesley Professional, 2012.
- [5] S. Dharwiyanti and R. S. Wahono, “Pengantar Unified Modeling Language (UML),” *IlmuKomputer.com*, pp. 1–13, 2003, [Online]. Available: <http://www.unej.ac.id/pdf/yanti-uml.pdf>.