

BAB III

METODE PENELITIAN DAN PERANCANGAN

3.1 Metode Pengembangan Sistem

Pada sistem informasi monitoring usaha dalam pengembangan sistem menggunakan metode *waterfall*. Metode *waterfall* adalah suatu proses pemodelan sistem secara terstruktur dan berurutan dimulai dari analisis kebutuhan, perancangan, implementasi dan pengujian sistem. Pemodelan sistem dengan metode ini dipilih karena sangat cocok dilakukan untuk kualitas sistem yang tetap terjaga karena pengembangannya yang terstruktur dan sekuensial. Berikut adalah tahapan mengenai tahapan siklus *waterfall* pada sistem informasi monitoring usaha. Pengembangan mulai dari analisis, dimana kebutuhan-kebutuhan sistem diperlukan akan dijelaskan ditahapan ini. Tahap perancangan dan hasil atau pembuatan, pengujian.

3.2 Analisis Kebutuhan

3.2.1 Metode Pengumpulan Data

Metode pengumpulan data dilakukan dengan mencari informasi terkait aplikasi yang akan dibuat yaitu sistem informasi monitoring usaha budidaya sarang walet CV. Walet Borneo. Metode yang digunakan adalah wawancara dan observasi.

- a. Wawancara dilakukan terhadap pemilik usaha dan melakukan tanya jawab terkait masalah apa yang dihadapi dalam melakukan monitoring pada bisnisnya. Pada saat wawancara pemilik mengeluhkan terlalu banyak waktu yang dibutuhkan untuk bisa melakukan monitoring ke seluruh cabang usaha sehingga menyebabkan pekerjaan tidak efisien dan proses penyimpanan data hasil panen serta penjualan masih sering terjadi kesalahan dikarenakan masih menggunakan pembukuan secara manual.
- b. Observasi dilakukan untuk mengamati secara langsung di lokasi penelitian untuk mengetahui kondisi apa yang terjadi. Pada saat observasi peneliti melihat langsung sulitnya mengelola informasi seluruh cabang usaha yang hanya di monitoring oleh satu orang yaitu pemilik, hal ini mengakibatkan banyaknya keterlambatan penerimaan informasi dari seluruh cabang sehingga penjualan menjadi lebih lama dan beberapa hasil panen sudah mengalami penurunan kualitas. Maka dibuatlah suatu aplikasi yang akan memberikan informasi hasil panen dan penjualan langsung kepada pemilik.

3.2.2 Hasil Analisis

a. User Requirement

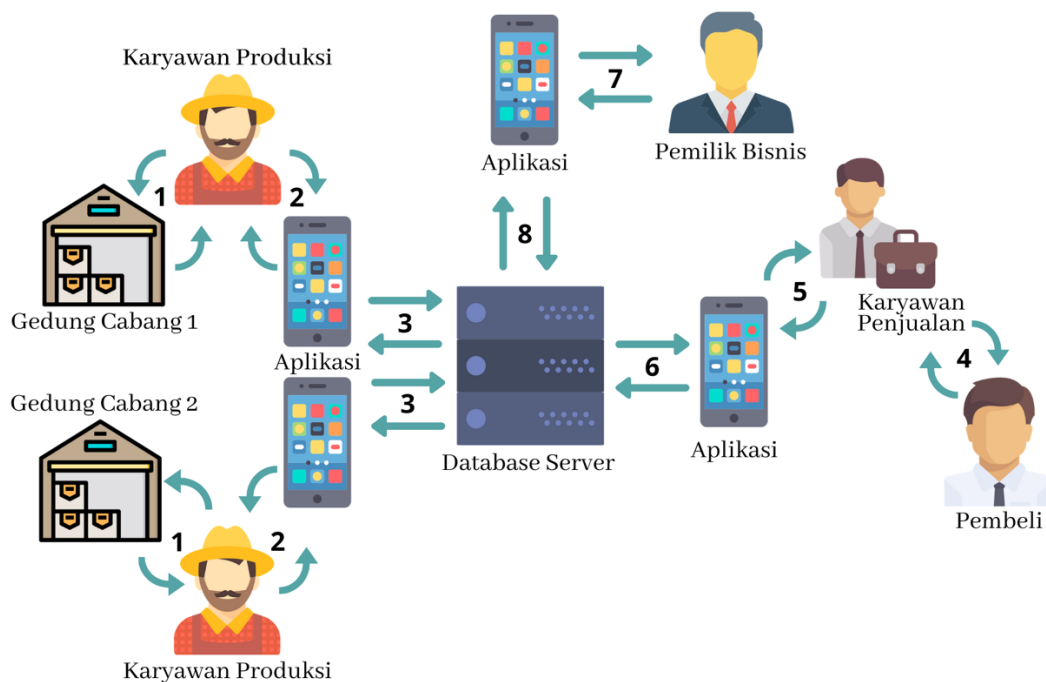
Dengan mengetahui kebutuhan awal dari sudut pandang pengguna, maka diperlukan *user requirement* yang didapatkan dari hasil wawancara penulis dengan pemilik CV. Walet Borneo. Pengguna akan dibagi menjadi dua pemilik dan karyawan yang akan dijelaskan pada Tabel 3.1.

Table 3.1 *User Requirement*

No	Kebutuhan	Skenario
1	Pemilik ingin seluruh pengguna <i>login</i> ke dalam sistem	- Masukan <i>username</i> dan <i>password</i>
2	Pemilik ingin halaman utama menampilkan seluruh menu pada aplikasi	- Setelah <i>login</i> , <i>user</i> akan langsung diarahkan ke halaman utama - Halaman utama akan menampilkan menu produksi, penjualan, karyawan dan aduan
3	Pemilik ingin melihat seluruh hasil panen dari masing-masing cabang	- Halaman produksi akan menampilkan hasil panen setiap cabang yang di <i>input</i> oleh karyawan - Halaman produksi menampilkan statistik hasil panen dari seluruh cabang
4	Pemilik ingin melihat seluruh hasil penjualan seluruh cabang	- Halaman penjualan menampilkan data penjualan dari masing-masing cabang usaha yang di <i>input</i> oleh karyawan
6	Pemilik ingin setiap terjadi masalah pada setiap cabang bisa di laporkan melalui sistem	- Sistem memiliki halaman aduan yang digunakan oleh karyawan untuk mengirimkan laporan apabila terjadi masalah pada setiap cabang
7	Pemilik ingin mengelola data karyawan	- Sistem memiliki halaman karyawan yang digunakan oleh pemilik untuk mengelola data karyawan yang berkerja di setiap cabang usaha

3.3 Alur Pengelolaan Bisnis pada Aplikasi

Alur sistem bisnis pada aplikasi untuk mengetahui bagaimana alur sistem ini bekerja ataupun runtutanya. Mulai dari pengguna sistem, manajemen *database*, data yang disajikan. Alur proses penggunaan sistem pengelolaan dan aplikasi monitoring usaha mulai dari data panen, data penjualan hingga hingga ke sistem informasi pengelolaan dapat dilihat pada Gambar 3.1.



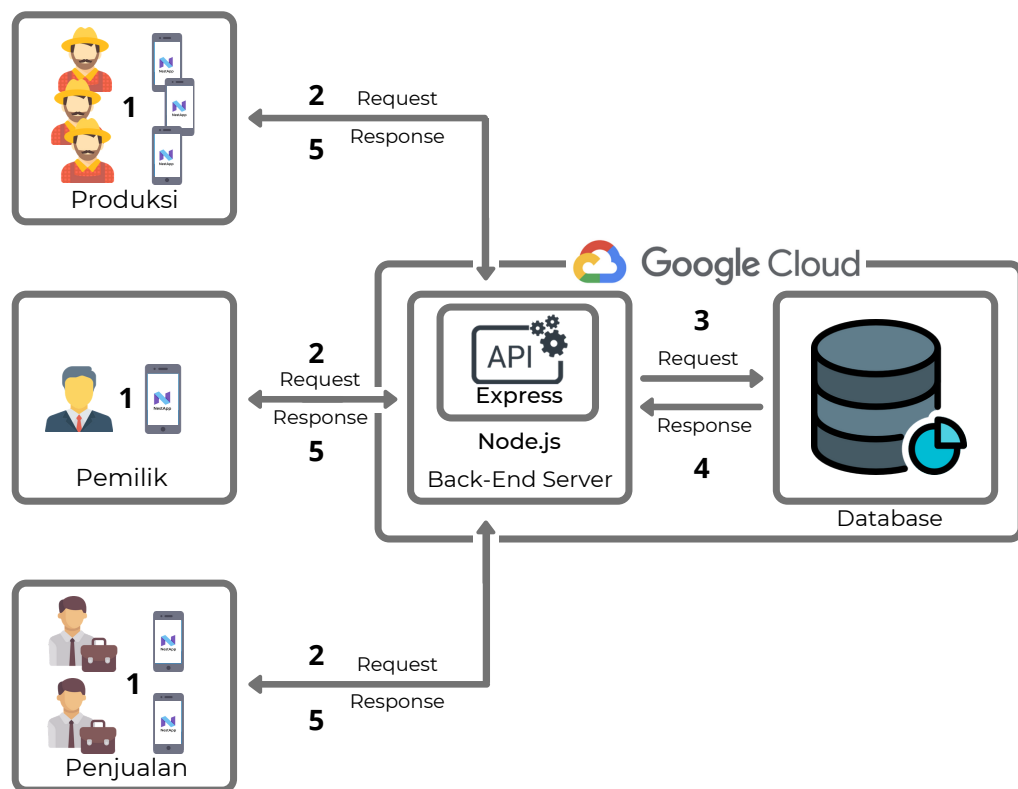
Gambar 3.1 Alur Sistem Aplikasi

Keterangan dari Gambar 3.1, yaitu:

1. Proses karyawan produksi melakukan panen pada gedung cabang.
2. Karyawan memasukkan data panen ke aplikasi.
3. Data panen yang dimasukkan oleh karyawan akan diproses oleh sistem untuk dimasukkan ke dalam *database server*.
4. Proses penjualan sarang antara karyawan dan pembeli.
5. Karyawan penjualan melakukan *update* hasil penjualan pada aplikasi.
6. Data penjualan yang telah di *update* akan diproses oleh sistem untuk dimasukkan ke dalam *database server*.
7. Pemilik bisnis melakukan monitoring pada panen dan penjualan dengan mengakses aplikasi.
8. Data panen dan penjualan yang telah diproses di *database* dan dikirimkan melalui aplikasi monitoring usaha.

3.4 Alur *Back-end* pada Aplikasi

Alur *back-end* pada aplikasi untuk mengetahui bagaimana data pada aplikasi berjalan hingga *database*. Alur *back-end* yang digambarkan seperti pada gambar 3.2.



Gambar 3.2 Alur *Back-end* Aplikasi

Keterangan dari Gambar 3.2, yaitu:

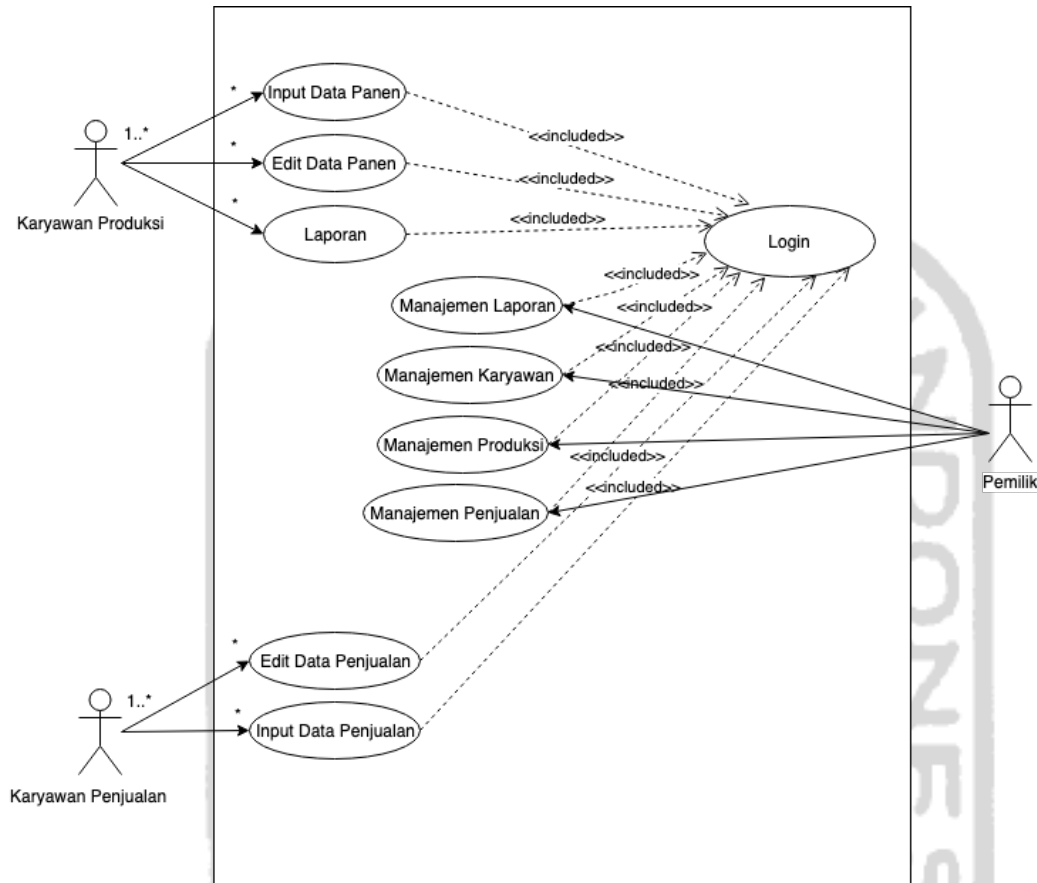
1. Proses *input* data yang dilakukan seluruh *user* kepada aplikasi, data tersebut merupakan data yang di *input* oleh karyawan produksi, karyawan penjualan dan pemilik.
2. Data yang telah di *input* lalu dikirim dan *request* oleh *back-end server* / *node.js* aplikasi ke service API menggunakan *express.js* yang telah dibuat agar data yang di *request* dapat eksekusi oleh banyak *user* secara bersamaan.
3. *Request* yang dikirimkan oleh *service API* terhadap *database* untuk mengambil data sesuai dengan *request user*.
4. *Response* data dari *database* sesuai dengan *request* yang dikirimkan oleh API
5. *Response* data dari API sesuai dengan *request* yang dikirimkan oleh *user*

3.5 Perancangan Proses Bisnis

Pada tahap ke dua dari metode *waterfall* adalah perancangan desain sistem sesuai dengan kebutuhan analisis yang telah dilakukan sebelumnya. Dari hasil analisis yang telah dirangkum, perancangan desain sistem baru bisa dibuat. Pada perancangan sistem ini terdiri dari beberapa tahapan perancangan yaitu perancangan *Use Case Diagram*, *Activity Diagram*, perancangan basisdata, dan perancangan antarmuka.

3.5.1 Use Case Diagram

Use case Diagram adalah proses menggambarkan fungsionalitas pada sistem dengan interaksinya. Rancangan *use case diagram* yang digambarkan seperti pada Gambar 3.3.



Gambar 3.3 Use Case Diagram Sistem

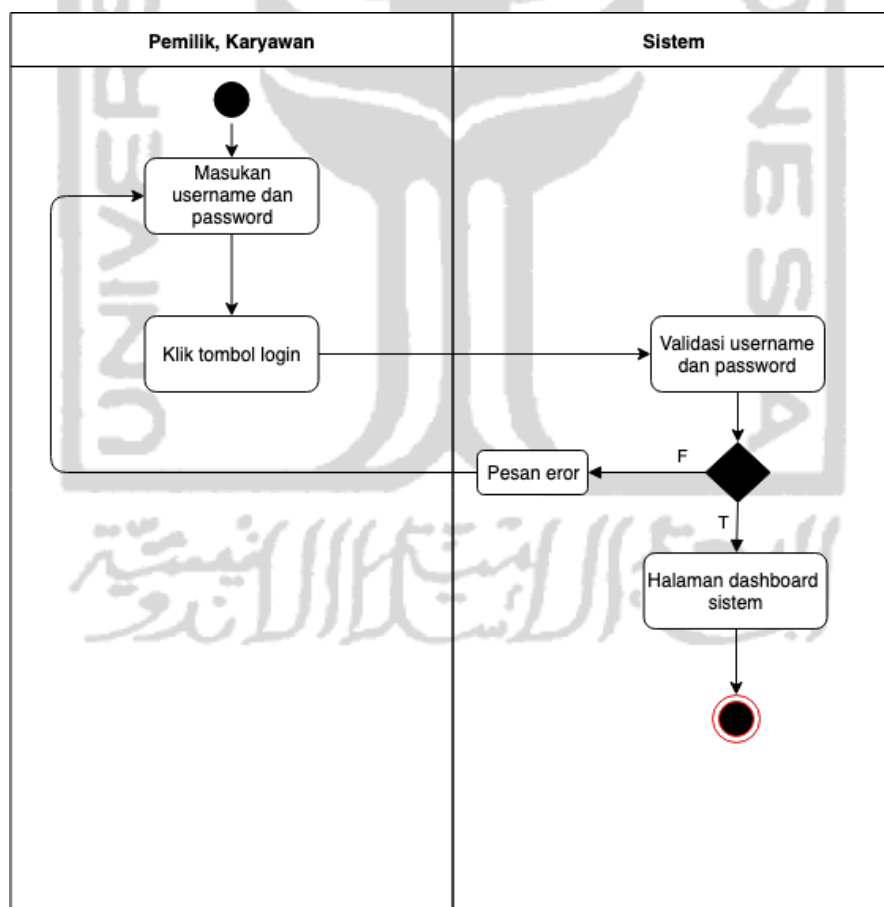
Pada Gambar 3.3 menjelaskan rancangan fungsi dari sistem yang akan diimplementasikan ke dalam bab selanjutnya. Dari gambar diatas dapat diketahui ada terdapat 3 aktor yang terlibat di dalam sistem ini, yaitu pemilik dan karyawan pengelola panen dan karyawan penjualan. Terdapat 10 proses pada sistem, dengan 3 aktor mempunyai masing-masing proses yaitu aktor pemilik mempunyai 4 proses yang meliputi manajemen produksi, manajemen penjualan, manajemen laporan, manajemen karyawan. Karyawan produksi mempunyai 3 proses yaitu proses *input* data panen, *edit* data panen dan laporan, seluruh proses dilakukan oleh masing-masing karyawan yang ditempatkan di setiap cabang usaha dan *login* sesuai dengan akun yang telah dibuat oleh admin, selanjutnya karyawan penjualan memiliki 2 proses yaitu proses *input* penjualan dan *edit* data penjualan. Seluruh proses harus melakukan *login* terlebih dahulu sebelum bisa menggunakan masing-masing proses.

3.5.2 Activity Diagram

Agar memahami lebih lanjut fungsi dari yang telah digambarkan pada *use case*, maka dibuatlah rangkaian *activity diagram* sebagai berikut:

1. Login

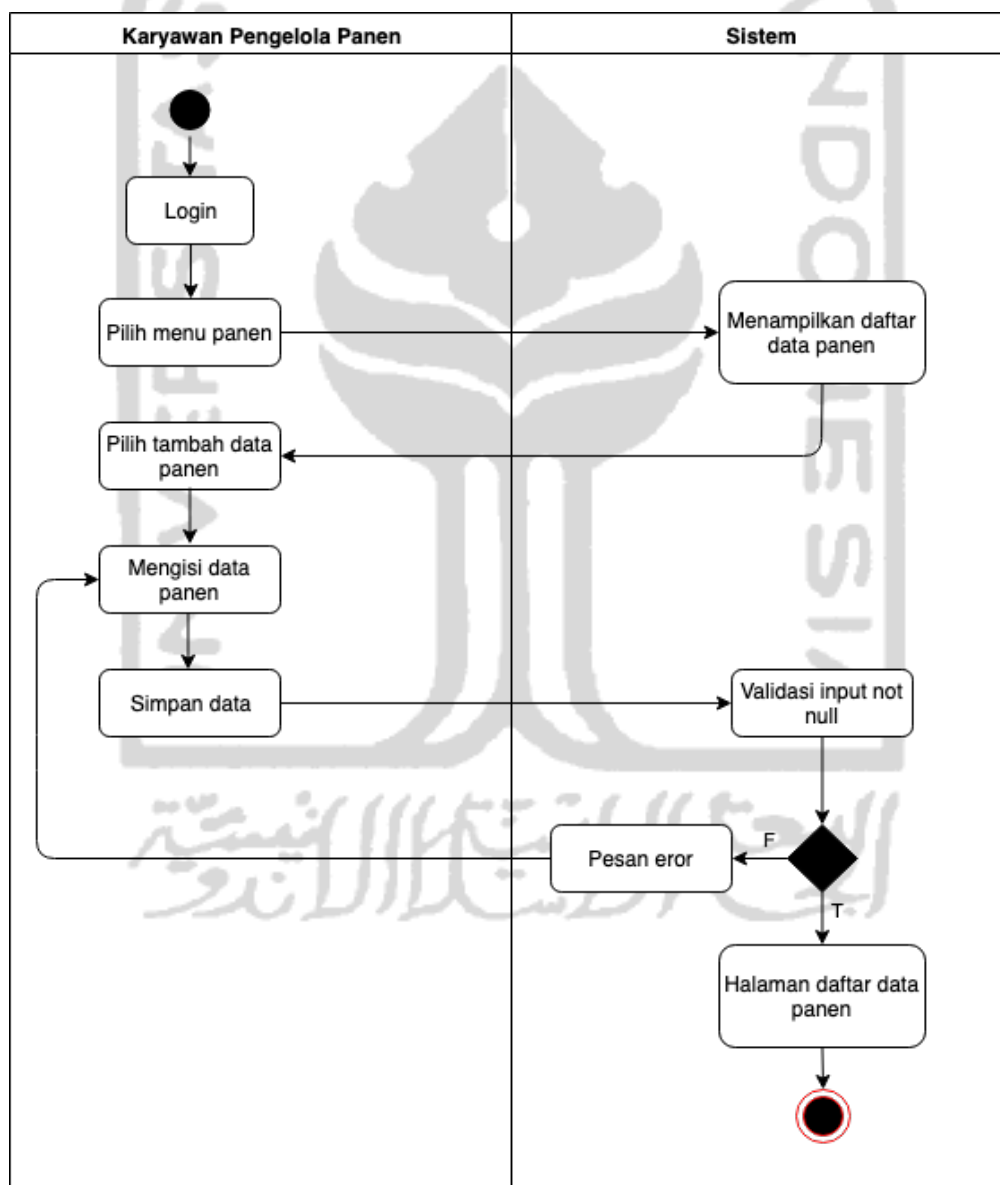
Pada diagram ini akan menampilkan aliran aktivitas seluruh *user* pada saat melakukan *login*. Untuk melakukan *login* pemilik harus mengisi *user name* dan *password*, sedangkan karyawan harus mengisi *user name* dan *password* yang sebelumnya harus dibuat terlebih dahulu oleh pemilik melalui menu manajemen karyawan. Apabila data yang di *input* sudah benar maka pemilik akan masuk ke halaman utama aplikasi, karyawan pengelola panen akan masuk ke halaman utama pengelolaan panen dan karyawan penjualan akan masuk ke halaman utama penjualan. Dan apabila *input* data tidak sesuai, maka akan menampilkan pesan bahwa data yang di *input* salah atau belum terdaftar dan akan tetap di halaman *login*. Untuk memperjelas dapat dilihat pada Gambar 3.4.



Gambar 3.4 Activity Diagram Login

2. *Input Data Panen*

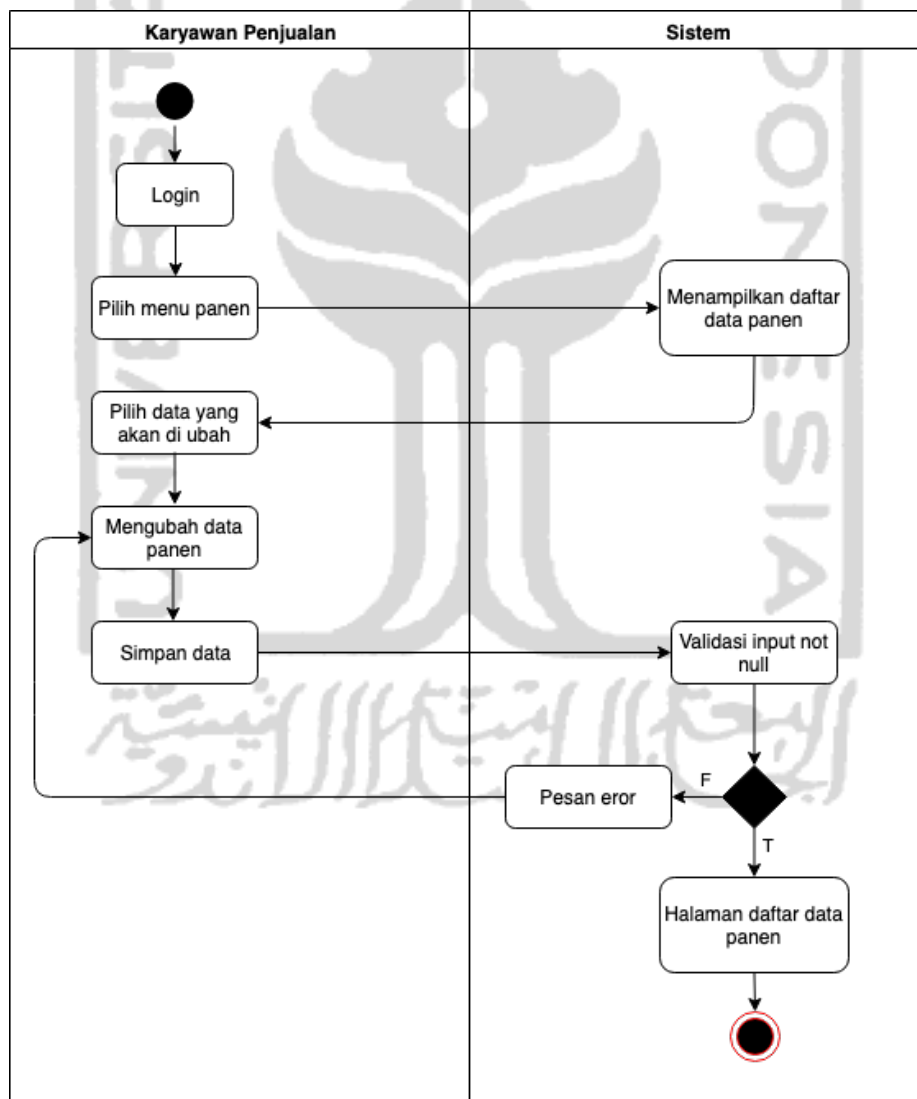
Pada diagram ini akan menampilkan aliran aktivitas yang dilakukan oleh karyawan pengelola panen untuk menyimpan data panen ke dalam sistem. Karyawan harus melakukan *login* terlebih dahulu untuk bisa mengakses sistem, karyawan akan memilih menu panen dan sistem akan menampilkan daftar data panen, selanjutnya memilih tambah data panen, mengisi data panen dan menyimpan data panen. Dan apabila data yang di *input* tidak diisi, maka akan menampilkan pesan bahwa data belum diisi. Data yang di *input* berdasarkan jenis grade pada sarang walet sesuai klasifikasi dari CV. Walet Borneo Untuk memperjelas dapat dilihat pada Gambar 3.5.



Gambar 3.5 *Activity Diagram Input Data Panen*

3. *Edit Data Panen*

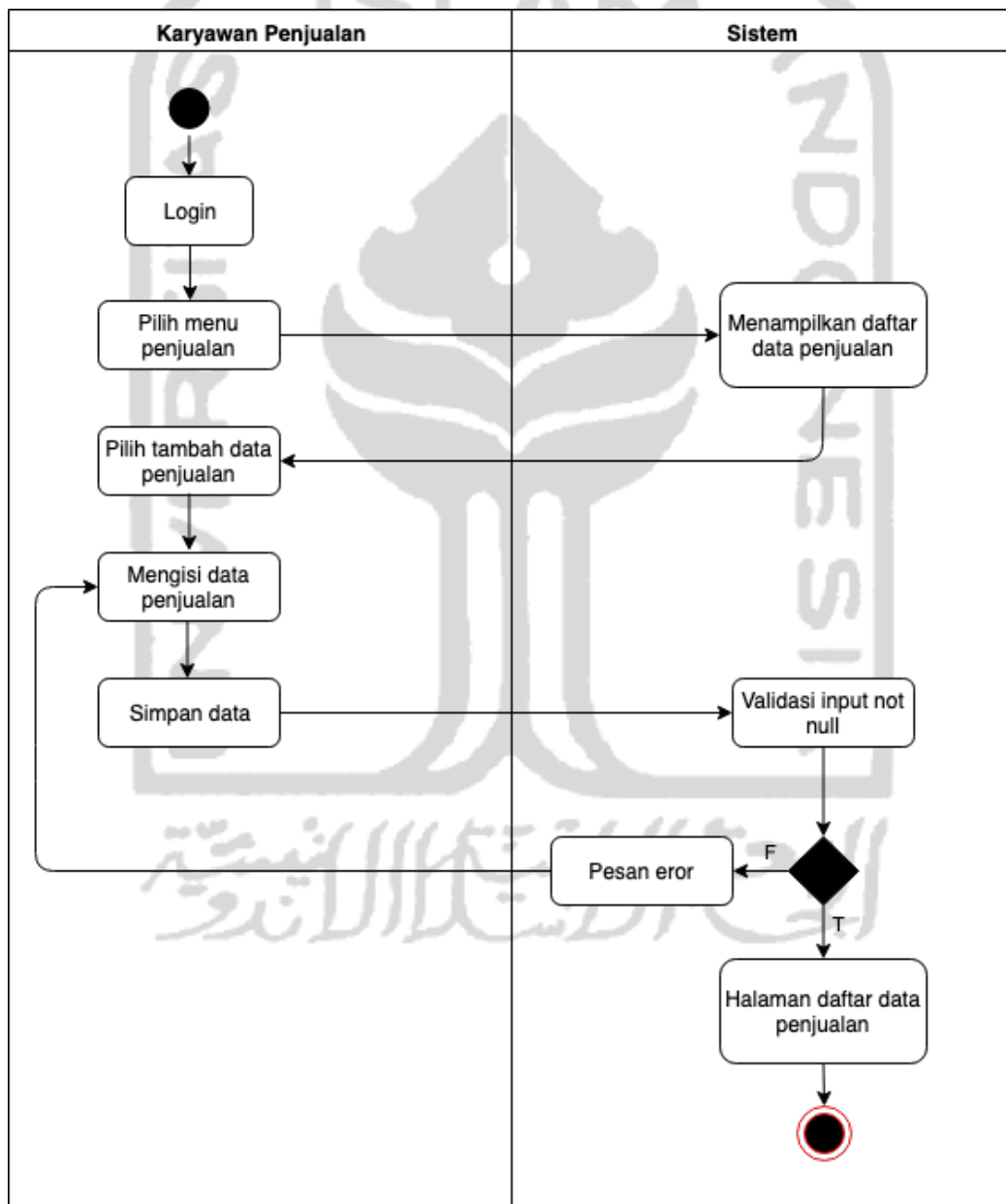
Pada diagram ini akan menampilkan aliran aktivitas yang dilakukan karyawan pengelola panen untuk mengubah data panen. Setelah *login* karyawan akan memilih menu panen dan sistem akan menampilkan seluruh daftar data panen, selanjutnya akan memilih data yang akan diubah, setelah mengubah dan menyimpan data panen, sistem akan memastikan bahwa tidak ada data yang kosong. Apabila data yang di *input* kosong maka akan muncul pesan bahwa data belum diisi. *Edit* data panen dilakukan ketika dalam kondisi hasil panen yang di *input* oleh karyawan produksi mengalami perbedaan grade pada proses penyortiran yang dilakukan oleh pemilik usaha, kesalahan tersebut terjadi akibat kurangnya pengalaman yang dimiliki oleh karyawan dalam melakukan penyortiran terhadap *grade* sarang. Untuk memperjelas dapat dilihat pada Gambar 3.6.



Gambar 3.6 *Activity Diagram Edit Data Panen*

4. *Input Data Penjualan*

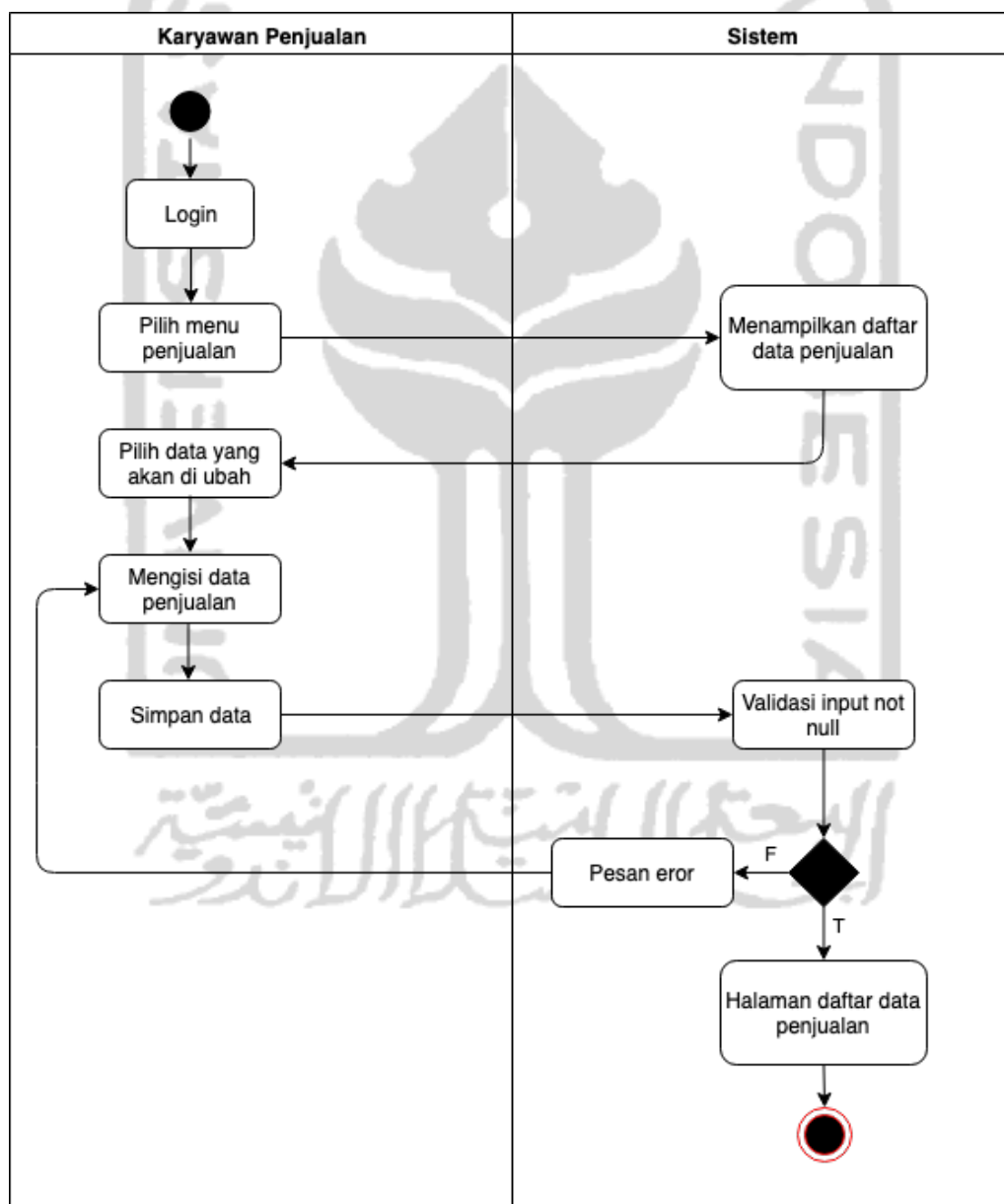
Pada diagram ini akan menampilkan aliran aktivitas yang dilakukan oleh karyawan penjualan untuk menyimpan data penjualan ke dalam sistem. Karyawan harus melakukan *login* terlebih dahulu untuk bisa mengakses sistem, karyawan akan memilih menu penjualan dan sistem akan menampilkan daftar data penjualan, selanjutnya memilih tambah data penjualan, mengisi data penjualan dan menyimpan data penjualan. Dan apabila data yang di *input* tidak di isi, maka akan menampilkan pesan bahwa data belum diisi. Untuk memperjelas dapat dilihat pada Gambar 3.7.



Gambar 3.7 *Activity Diagram Input Data Penjualan*

5. *Edit Data Penjualan*

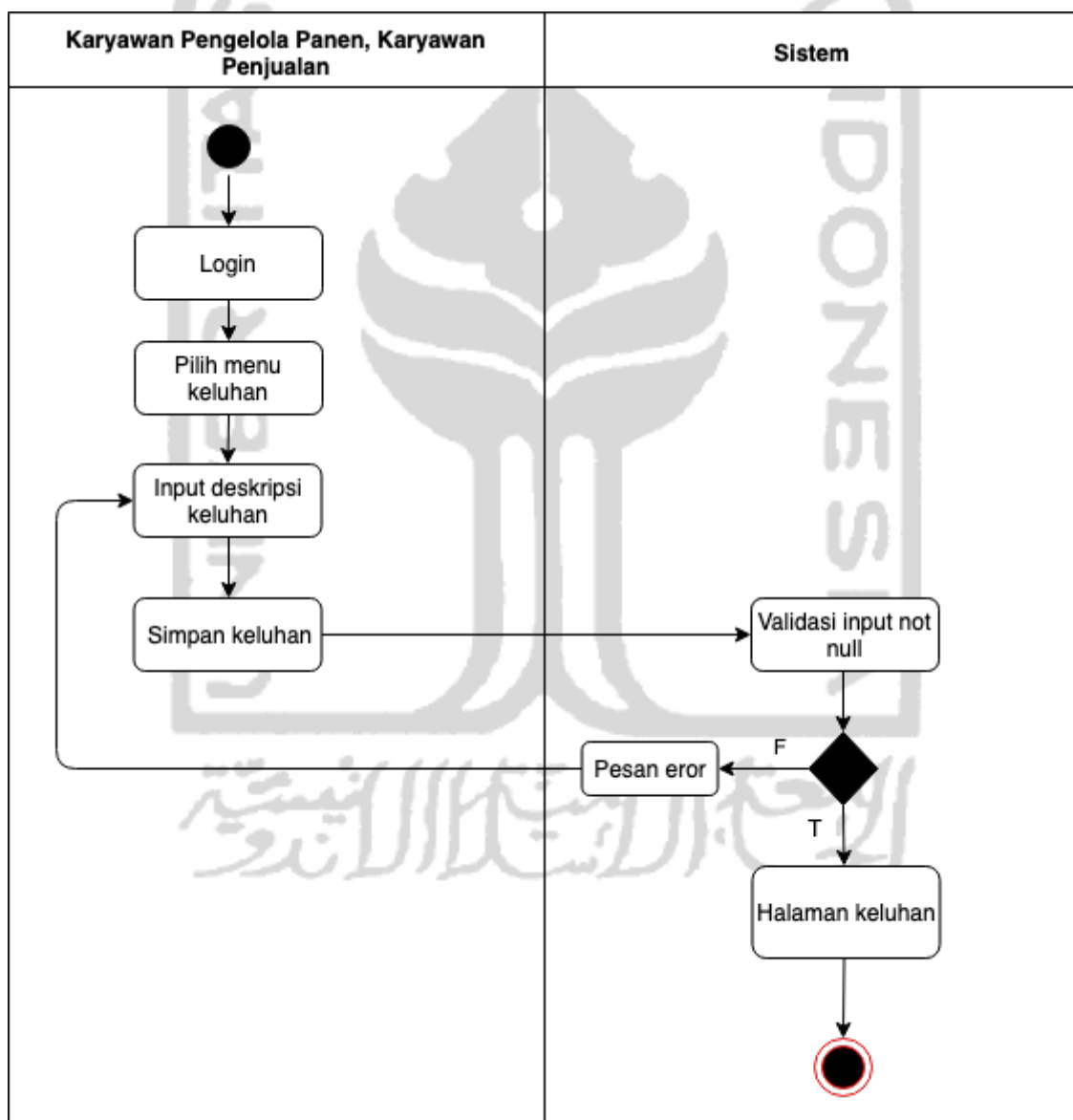
Pada diagram ini akan menampilkan aliran aktivitas yang dilakukan karyawan penjualan untuk mengubah data penjualan. Setelah *login* karyawan akan memilih menu penjualan dan sistem akan menampilkan seluruh daftar data penjualan, selanjutnya akan memilih data yang akan diubah, setelah mengubah dan menyimpan data penjualan maka sistem akan memastikan bahwa data tidak ada yang kosong. Apabila data yang di *input* kosong maka akan muncul pesan bahwa data belum diisi. *Edit data* penjualan dilakukan ketika terjadi kesalahan dalam menginputkan berat dan harga dengan nota penjualan. Untuk memperjelas dapat dilihat pada Gambar 3.8.



Gambar 3.8 *Activity Diagram Edit Data Penjualan*

6. Laporan

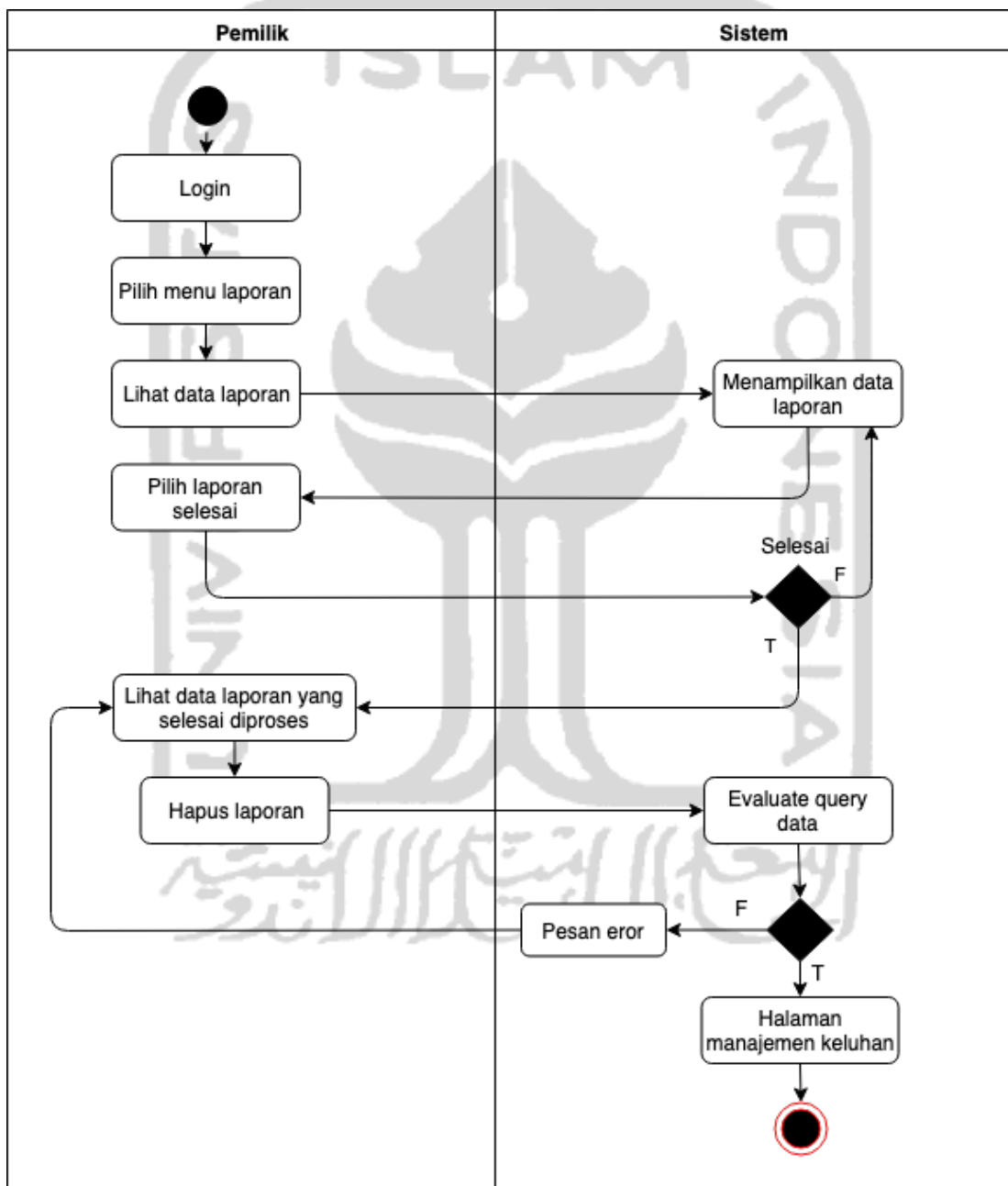
Pada diagram ini akan menampilkan aliran aktivitas yang dilakukan karyawan pengelola panen dan karyawan penjualan untuk melaporkan apabila terjadi masalah pada apa yang dikelola. Setelah login karyawan akan memilih menu keluhan, selanjutnya mengisi deskripsi keluhan dan mengirim data keluhan, sistem akan memastikan bahwa pesan tersebut tidak kosong. Dan apabila data yang di *input* kosong maka akan muncul pesan bahwa data keluhan belum diisi. Laporan yang dikirim hanya bisa dihapus oleh pemilik dan tidak bisa dihapus oleh karyawan yang mengirimkan data laporan tersebut. Untuk memperjelas dapat dilihat pada Gambar 3.9.



Gambar 3.9 Activity Diagram Input Laporan

7. Manajemen Laporan

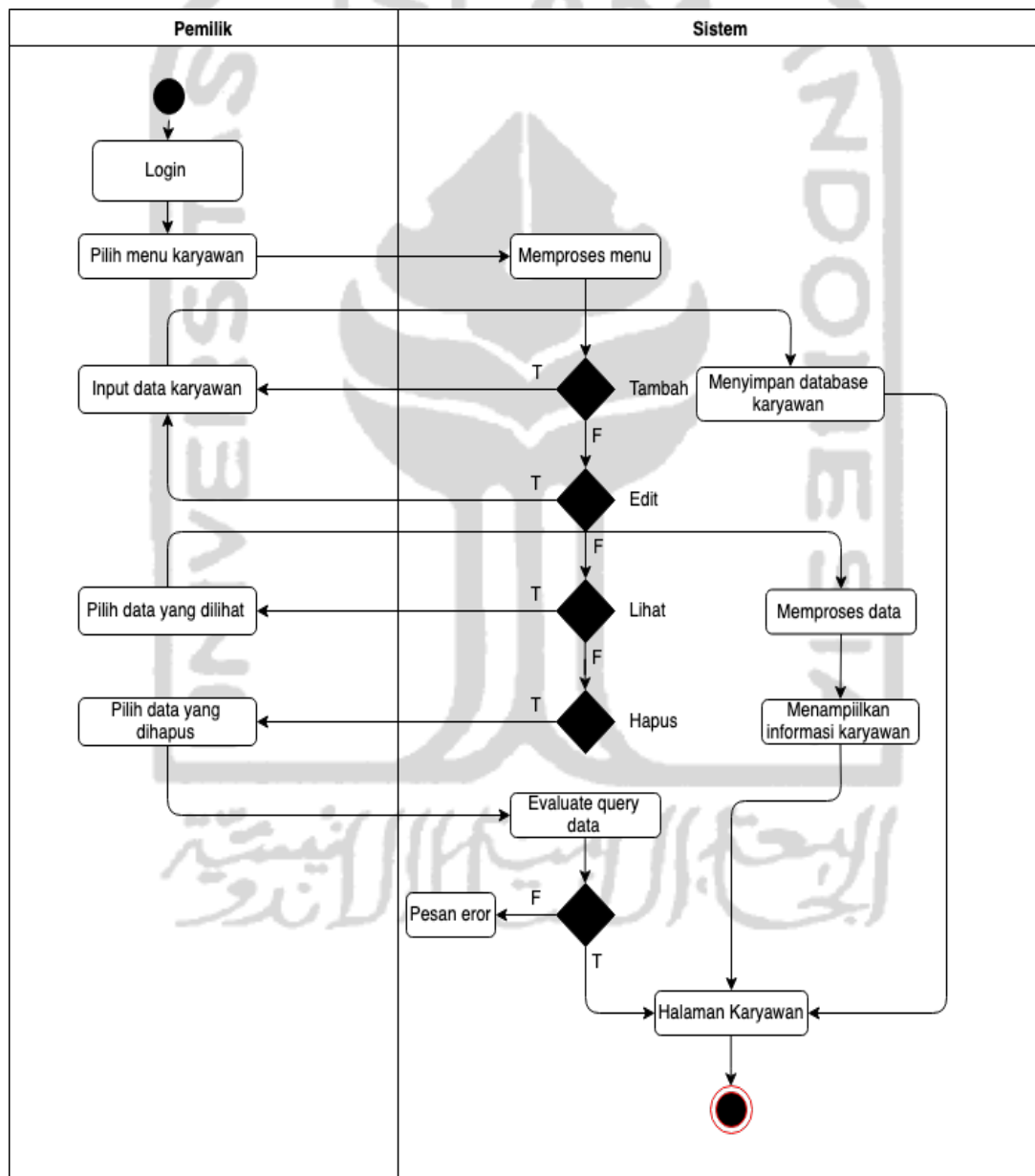
Pada diagram ini akan menampilkan aliran aktivitas yang dilakukan pemilik untuk mengelola laporan yang di *input* oleh karyawan. Setelah *login* pemilik akan memilih menu laporan dan memilih data keluhan yang akan diproses. Apabila laporan tersebut telah diproses maka status pada laporan tersebut diubah menjadi selesai, selanjutnya data yang sudah selesai diproses akan siap untuk dihapus apabila ingin dihapus.. Untuk memperjelas dapat dilihat pada Gambar 3.10.



Gambar 3.10 Activity Diagram Manajemen Laporan

8. Manajemen Karyawan

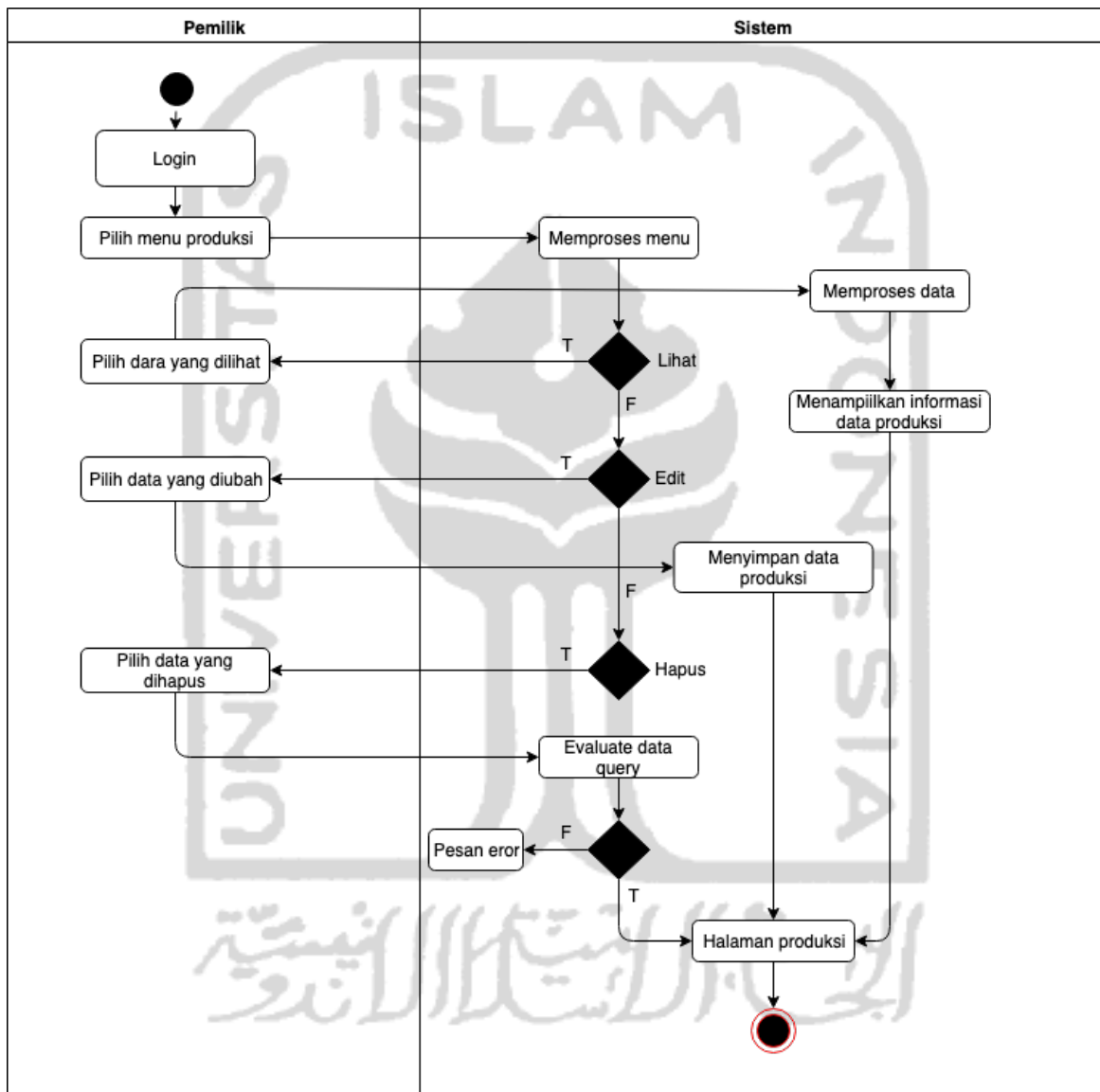
Pada diagram ini akan menampilkan aliran aktivitas yang dilakukan pemilik untuk mengelola data karyawan. Setelah *login* pemilik akan memilih menu karyawan, selanjutnya pemilik dapat mengelola seluruh data karyawan, data karyawan bisa ditambah diubah dan dihapus. Pada menu ini pemilik akan membuat akun yang akan digunakan oleh seluruh karyawan produksi dan penjualan *login* ke sistem, pemilik juga bisa menentukan *role* apa yang akan dikerjakan oleh karyawan saat menggunakan sistem, role terbagi menjadi dua yaitu produksi dan penjualan. Untuk memperjelas dapat dilihat pada Gambar 3.11.



Gambar 3.11 Activity Diagram Manajemen Karyawan

9. Manajemen Produksi

Pada diagram ini akan menampilkan aliran aktivitas yang dilakukan pemilik untuk memonitoring dan manajemen data hasil panen. Pada diagram ini sistem akan menampilkan halaman produksi. Halaman ini digunakan untuk melihat data hasil panen seluruh cabang usaha, Data panen bisa diubah dan dihapus. Untuk memperjelas dapat dilihat pada Gambar 3.12.

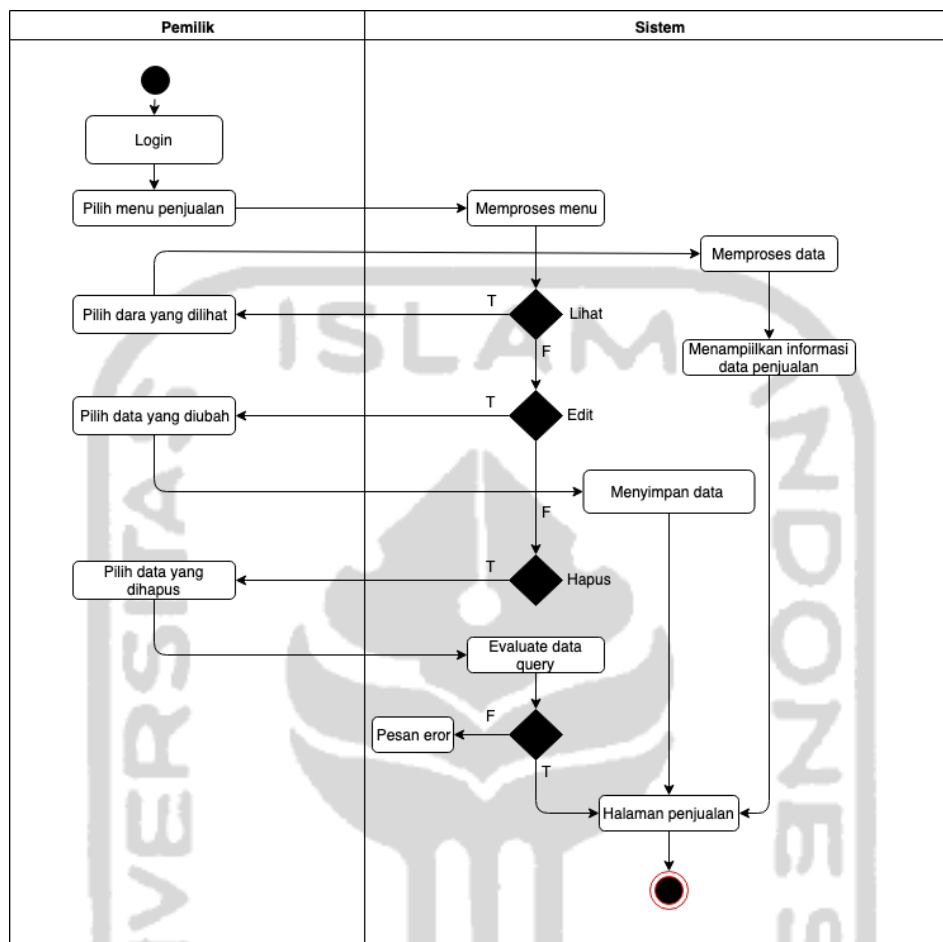


Gambar 3.12 Activity Diagram Manajemen Produksi

10. Manajemen Penjualan

Pada diagram ini akan menampilkan aliran aktivitas yang dilakukan pemilik untuk manajemen data hasil penjualan. Pada *diagram* ini sistem akan menampilkan halaman

penjualan. Halaman ini digunakan untuk melihat data hasil penjualan seluruh cabang usaha. Data penjualan bisa ubah dan dihapus. Untuk memperjelas dapat dilihat pada Gambar 3.13.



Gambar 3.13 Activity Diagram Manajemen Penjualan

3.6 Perancangan Basisdata

3.6.1 Struktur Tabel

Struktur tabel pada sistem informasi ini menjelaskan secara detail struktur dari setiap tabel yang terdapat pada *database*. Berikut merupakan tabel–tabel tersebut, yaitu:

a. Tabel Gedung

Nama tabel: Gedung

Primary_key: id_gedung

Foreign_key:

Fungsi: Tabel berfungsi untuk menyimpan data gedung yang dikelola oleh pemilik

Tabel Building terdapat 5 kolom atribut. Berikut adalah penjelasan dari atribut tersebut:

- id_gedung: Memiliki tipe data Integer dengan batasan karakter 11, *Constrain Primary_key* dan *Auto Increment*.

- nama: Memiliki tipe data Varchar dengan batasan karakter 255
- alamat: Memiliki tipe data Varchar dengan batasan karakter 255
- createdAt: Memiliki tipe data date
- updateAt: Memiliki tipe data date

Untuk lebih jelas dapat dilihat pada Tabel 3.2.

Table 3.2 Gedung

No	Atribut	Tipe Data	Constrain	Keterangan
1	id_gedung	Integer(11)	Primery_key	Not Null
2	nama	Varchar(255)		Not Null
3	alamat	Varchar(255)		Not Null
4	createdAt	Date		Not Null
5	updateAt	Date		Not Null

b. Tabel Users

Nama tabel: Users

Primary_key: id_users

Foreign_key: id_gedung

Fungsi: Tabel berfungsi untuk menyimpan data karyawan yang dikelola pemilik

Tabel Karyawan terdapat 8 kolom atribut. Berikut adalah penjelasan dari atribut tersebut:

- id_users: Memiliki tipe data Integer dengan batasan karakter 11, *Constrain Primary_key* dan *Auto Increment*.
- id_gedung: Memiliki tipe data Integer dengan batasan karakter 11, *Constrain Foreign_key* dari tabel Gedung
- nama: Memiliki tipe data Varchar dengan batasan karakter 255
- alamat: Memiliki tipe data Varchar dengan batasan karakter 255
- no_hp: Memiliki tipe data Varchar dengan batasan karakter 12
- role: Memiliki tipe data Varchar dengan batasan karakter 255
- username : Memiliki tipe data Varchar dengan batasan karakter 255
- password : Memiliki tipe data Varchar dengan batasan karakter 255
- createdAt: Memiliki tipe data date
- updateAt: Memiliki tipe data date

Untuk lebih jelas dapat dilihat pada Tabel 3.1.

Table 3.3 Users

No	Atribut	Tipe Data	Constrain	Keterangan
1	id_users	Integer(11)	Primery_key	Not Null

No	Atribut	Tipe Data	Constrain	Keterangan
2	id_gedung	Integer(11)	<i>Foreign key</i>	Tabel Gedung
3	nama	Varchar(255)		<i>Not Null</i>
4	alamat	Integer(12)		<i>Not Null</i>
5	no_hp	Integer(12)		<i>Not Null</i>
6	role	Varchar(255)		<i>Not Null</i>
7	username	Varchar(255)		<i>Not Null</i>
8	password	Varchar(255)		<i>Not Null</i>
9	createdAt	Date		<i>Not Null</i>
10	updateAt	Date		<i>Not Null</i>

c. Tabel Produksi

Nama tabel: Produksi

Primary_key: id_produk

Foreign_key: id_users
: id_gedung

Fungsi: Tabel berfungsi untuk menyimpan data hasil panen yang di *input* oleh masing masing karyawan di setiap gedung.

Tabel Item terdapat 9 kolom atribut. Berikut adalah penjelasan dari atribut tersebut:

- id_produk: Memiliki tipe data Integer dengan batasan karakter 11, *Constrain Primary_key* dan *Auto Increment*.
- id_users: Memiliki tipe data Integer dengan batasan karakter 11, *Constrain Foreign_key* dari tabel Karyawan.
- id_gedung: Memiliki tipe data Integer dengan batasan karakter 11, *Constrain Foreign_key* dari tabel Gedung.
- mangkuk: Memiliki tipe data float dengan batasan karakter 10.
- sudut: Memiliki tipe data float dengan batasan karakter 10.
- patahan: Memiliki tipe data float dengan batasan karakter 10.
- strip: Memiliki tipe data float dengan batasan karakter 10.
- keterangan: Memiliki tipe data Varchar dengan batasan karakter 255.
- createdAt: Memiliki tipe data date
- updateAt: Memiliki tipe data date

Untuk lebih jelas bisa dilihat pada Tabel 3.4.

Table 3.4 Produksi

No	Atribut	Tipe Data	Constrain	Keterangan
1	id_produk	Integer(11)	<i>Primery_key</i>	<i>Not Null</i>

No	Atribut	Tipe Data	Constrain	Keterangan
2	id_users	Integer(11)	<i>Foreign key</i>	Tabel Users
3	id_gedung	Integer(11)	<i>Foreign key</i>	Tabel Gedung
4	mangkuk	Float(10)		<i>Not Null</i>
5	sudut	Float(10)		<i>Not Null</i>
6	patahan	Float(10)		<i>Not Null</i>
7	strip	Float(10)		<i>Not Null</i>
8	keterangan	Varchar(255)		<i>Not Null</i>
9	createdAt	Date		<i>Not Null</i>
10	updateAt	Date		<i>Not Null</i>

d. Tabel Penjualan

Nama tabel: Penjualan

Primary_key: id_penjualan

Foreign_key: id_users

id_gedung

Fungsi: Tabel berfungsi untuk menyimpan data penjualan dari barang yang di *input* oleh karyawan penjualan

Tabel Jual terdapat 9 kolom atribut, Berikut adalah penjelasan dari atribut tersebut:

- id_penjualan: Memiliki tipe data Integer dengan batasan karakter 11, *Constrain Primary_key* dan *Auto Increment*.
- id_users: Memiliki tipe data Integer dengan batasan karakter 11, *Constrain Foreign_key* dari tabel Karyawan.
- id_gedung : Memiliki tipe data Integer dengan batasan karakter 11, *Constrain Foreign_key* dari tabel Gedung.
- pembeli: Memiliki tipe data Varchar dengan batasan karakter 255.
- mangkuk: Memiliki tipe data float dengan batasan karakter 10.
- sudut: Memiliki tipe data float dengan batasan karakter 10.
- patahan: Memiliki tipe data float dengan batasan karakter 10.
- strip: Memiliki tipe data float dengan batasan karakter 10.
- harga_mangkuk: Memiliki tipe data float dengan batasan karakter 10.
- harga_sudut: Memiliki tipe data float dengan batasan karakter 10.
- harga_patahan: Memiliki tipe data float dengan batasan karakter 10.
- harga_strip: Memiliki tipe data float dengan batasan karakter 10.
- createdAt: Memiliki tipe data date
- updateAt: Memiliki tipe data date

Untuk lebih jelas dapat dilihat pada Tabel 3.5.

Table 3.5 Penjualan

No	Atribut	Tipe Data	Cosntrain	Keterangan
1	id_penjualan	Integer(11)	Primary key	Not Null
2	id_users	Integer(11)	Foreign key	Tabel Users
3	id_gedung	Integer(11)	Foreign key	Tabel Gedung
4	pembeli	Varchar(255)		Not Null
5	mangkuk	Float(10)		Not Null
6	sudut	Float(10)		Not Null
7	patahan	Float(10)		Not Null
8	strip	Float(10)		Not Null
9	harga_mangkuk	Float(10)		Not Null
10	harga_sudut	Float(10)		Not Null
11	harga_patahan	Float(10)		Not Null
12	harga_strip	Float(10)		Not Null
13	createdAt	Date		Not Null
14	updateAt	Date		Not Null

e. Tabel Laporan

Nama tabel: Laporan

Primary_key: id_laporan

Foreign_key: id_users

Fungsi: Tabel berfungsi untuk menyimpan data keluhan yang di *input* oleh karyawan dan dikelola oleh pemilik usaha.

Tabel Complain terdapat 3 kolom atribut. Berikut adalah penjelasan dari atribut tersebut:

- id_laporan: Memiliki tipe data Integer dengan batasan karakter 11, *Constrain Primary_key* dan *Auto Increment*.
- id_users: Memiliki tipe data Integer dengan batasan karakter 11, *Constrain Foreign_key* dari tabel Karyawan.
- keterangan: Memiliki tipe data Varchar dengan batasan karakter 255
- createdAt: Memiliki tipe data date
- updateAt: Memiliki tipe data date

Untuk lebih jelas dapat dilihat pada Tabel 3.6.

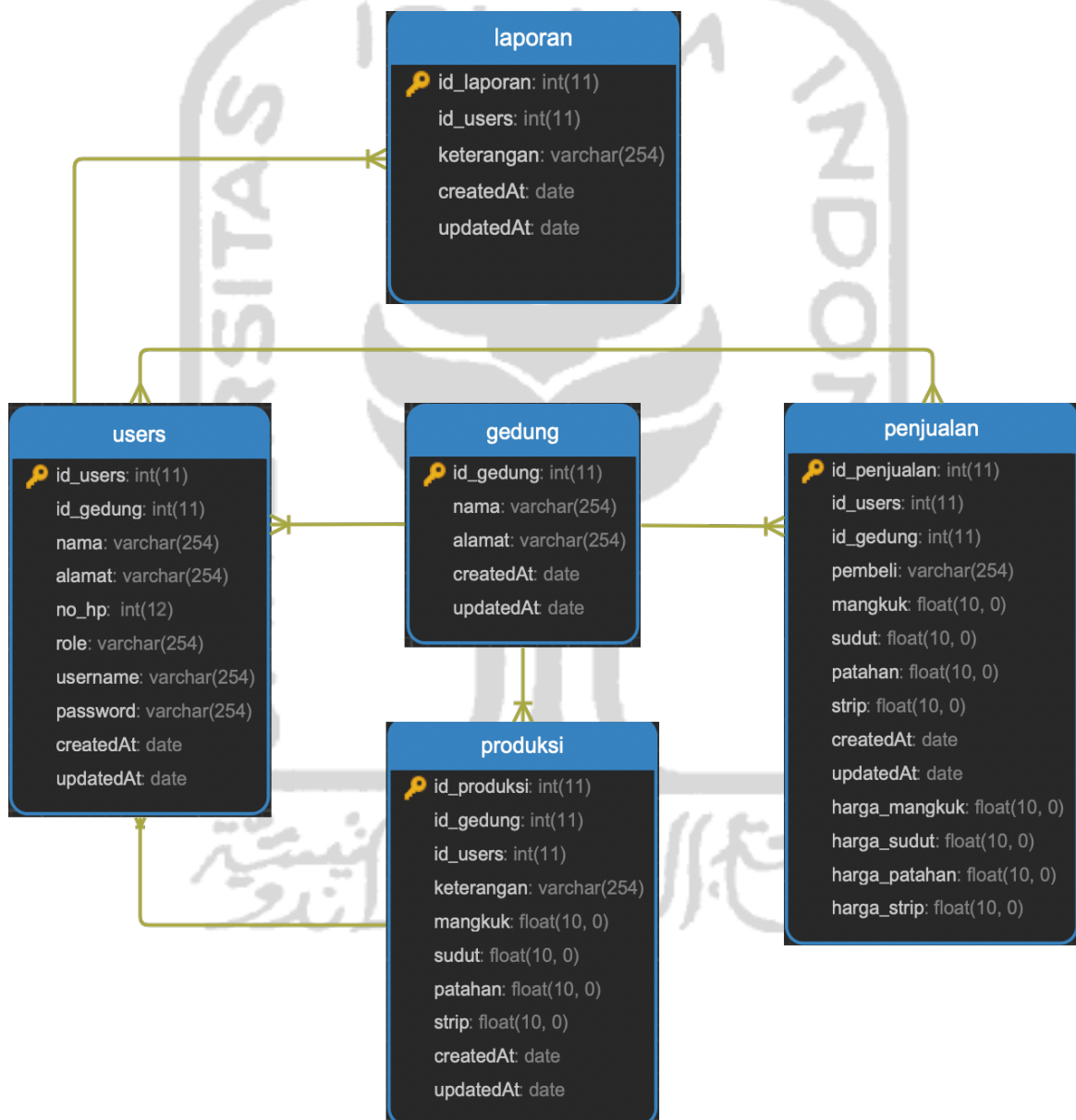
Table 3.6 Laporan

No	Atribut	Tipe Data	Constrain	Keterangan
1	id_laporan	Integer(11)	Primary key	Not Null
2	id_users	Integer(11)	Foreign key	Tabel Users
3	keterangan	Varchar(255)		Not Null
4	createdAt	Date		Not Null

No	Atribut	Type Data	Constrain	Keterangan
5	updateAt	Date		<i>Not Null</i>

3.6.2 Relasi Antar Tabel

Relasi antar tabel yang dibuat sesuai dengan analisa kebutuhan sistem, pada perancangan *database* pada sistem ini, terdapat 5 tabel yang memiliki relasi antar satu dengan yang lain yaitu tabel Gedung, Users, Produksi, Penjualan dan Laporan. Adapun rancangan relasi antar tabel *database* dapat dilihat pada Gambar 3.14 .

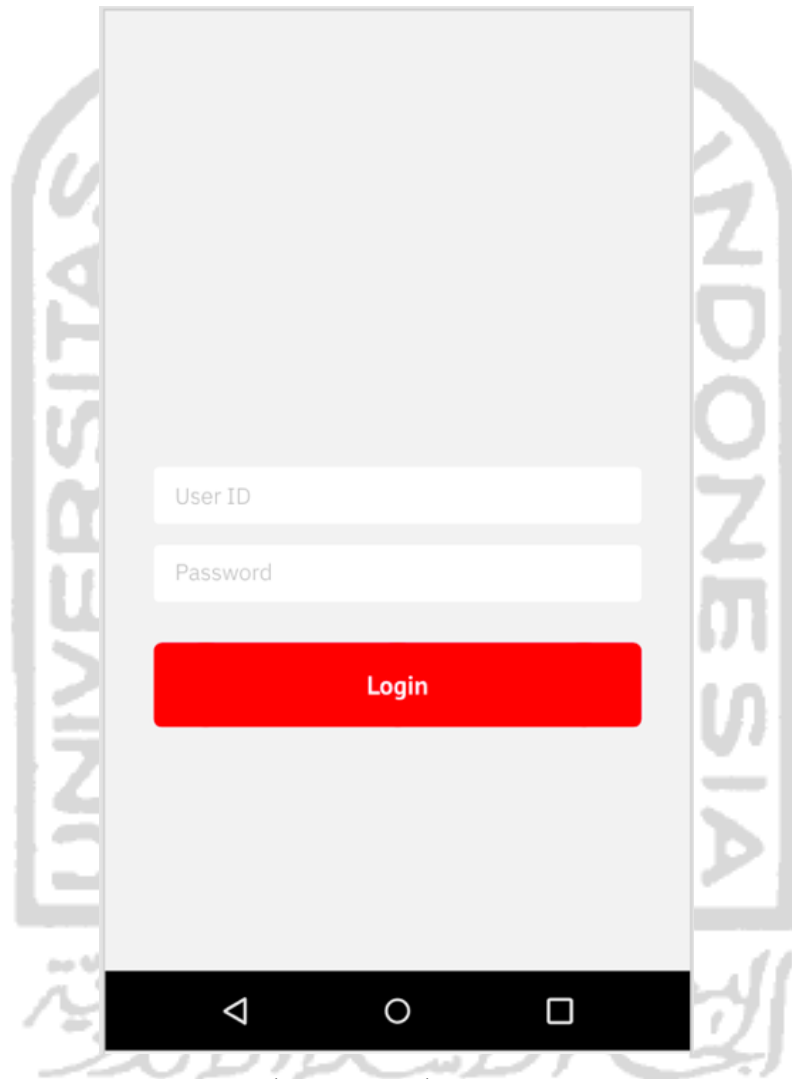


Gambar 3.14 Relasi Antar Tabel *Database*

3.7 Perancangan Antarmuka

a. Tampilan halaman *Login*

Halaman ini merupakan halaman awal dari aplikasi monitoring usaha pada bisnis sarang walet, *User* harus memasukan *username* dan *password* yang sudah terdaftar sebelumnya dan menekan tombol *Login* untuk mengakses aplikasi. Untuk lebih jelas dapat dilihat tampilan halaman *login* pada Gambar 3.15.

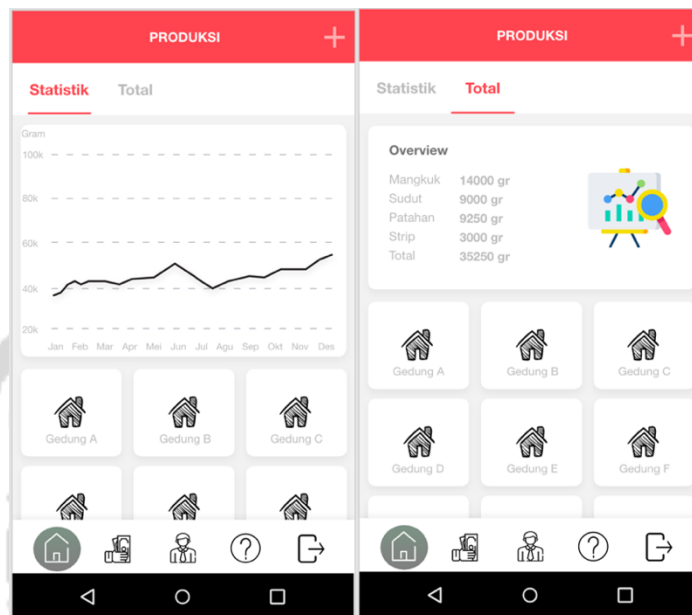


Gambar 3.15 Halaman *Login*

b. Halaman Manajemen Produksi

Halaman ini merupakan halaman yang digunakan pemilik usaha untuk memonitoring hasil panen pada setiap cabang dengan menampilkan statistik panen dan total panen seluruh cabang maupun setiap cabang. Halaman ini juga bisa melakukan penambahan cabang ketika perusahaan memiliki cabang usaha baru dengan memilih *button* yang memiliki *icon plus* hanya

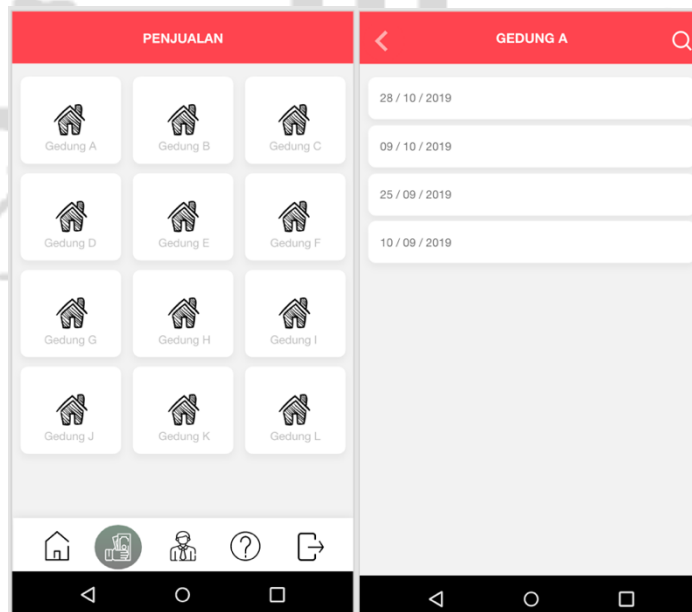
bisa diakses oleh pemilik setelah melakukan *login* terlebih dahulu. Untuk lebih jelas dapat dilihat tampilan halaman manajemen produksi pada Gambar 3.16.



Gambar 3.16 Halaman Manajemen Produksi

c. Halaman Manajemen Penjualan

Halaman ini merupakan halaman yang digunakan oleh pemilik untuk memonitoring seluruh hasil penjualan yang di *input* oleh karyawan. Halaman ini bisa diakses oleh pemilik setelah melakukan *login* terlebih dahulu. Untuk lebih jelas dapat dilihat tampilan halaman manajemen penjualan pada Gambar 3.17.



Gambar 3.17 Halaman Manajemen Penjualan

d. Halaman Manajemen Karyawan

Halaman ini merupakan halaman yang digunakan oleh pemilik untuk mengelola data karyawan yang bekerja dan ditempatkan dimasing-masing cabang usaha. Pada halaman karyawan pemilik akan membuat akun karyawan dan menentukan posisi pekerjaan karyawan. Untuk lebih jelas dapat dilihat tampilan halaman manajemen karyawan pada Gambar 3.18.



Gambar 3.18 Halaman Manajemen Karyawan

e. Halaman *Input* Data Karyawan

Halaman ini merupakan halaman yang digunakan oleh pemilik untuk membuat akun karyawan dan menentukan posisi pekerjaan yang akan dilakukan karyawan. Untuk lebih jelas dapat dilihat tampilan halaman *input* data karyawan pada Gambar 3.19.

Gambar 3.19 Halaman *Input* Data Karyawan

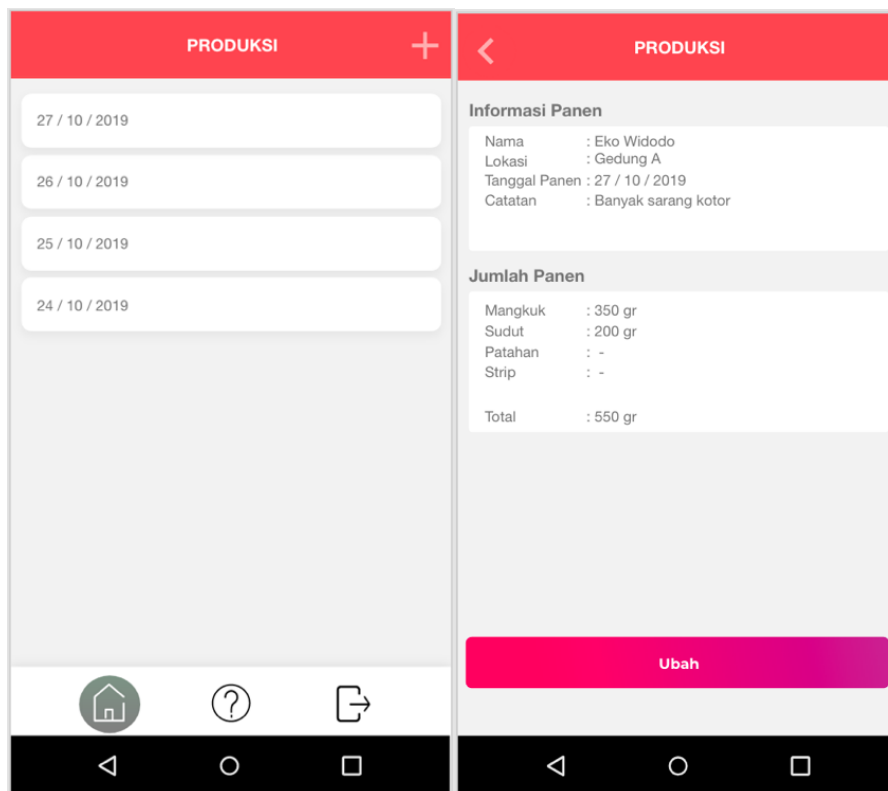
f. Halaman Manajemen Laporan

Halaman ini merupakan halaman yang digunakan oleh pemilik untuk mengelola data keluhan yang di *input* oleh karyawan dari masing - masing cabang usaha. Untuk lebih jelas dapat dilihat tampilan halaman manajemen laporan pada Gambar 3.20.

Gambar 3.20 Halaman Manajemen Laporan

g. Halaman Karyawan Produksi

Halaman ini merupakan halaman yang digunakan oleh karyawan untuk memasukan data hasil panen. Pada halaman karyawan juga dapat melihat hasil panen yang sudah dilakukan sebelumnya dengan melihat tanggal pada halaman tersebut. Untuk lebih jelas dapat dilihat tampilan halaman karyawan produksi pada Gambar 3.21.



Gambar 3.21 Halaman Karyawan Produksi

h. Halaman *Input* Data Panen

Halaman ini merupakan *form* yang digunakan karyawan untuk menyimpan data hasil panen. Untuk lebih jelas dapat dilihat tampilan halaman *input* data panen pada Gambar 3.22.

Gambar 3.22 Halaman *Input* Data Panen

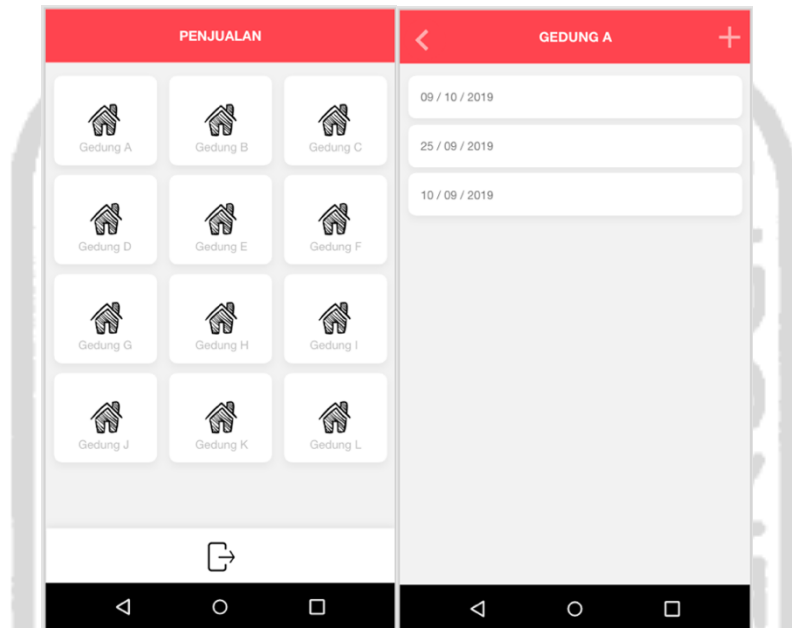
i. Halaman *Input* Laporan Karyawan

Halaman ini merupakan halaman yang digunakan karyawan untuk mengirimkan laporan ketika terjadi masalah pada cabang yang dikelola. Karyawan mengirimkan laporan dengan mengisi *form* sesuai dengan apa yang ingin dilaporkan. Untuk lebih jelas dapat dilihat tampilan halaman *input* laporan karyawan pada Gambar 3.23.

Gambar 3.23 Halaman *Input* Laporan Karyawan

j. Halaman Karyawan Penjualan

Halaman ini merupakan halaman yang digunakan karyawan untuk memasukan data penjualan sarang yang didapatkan dari hasil panen. Pada halaman ini karyawan juga dapat melihat penjualan yang telah dilakukan sebelumnya dengan memilih gedung mana yang ingin dilihat dan setelah itu memilih berdasarkan tanggal penjualan yang dilakukan sebelumnya. Untuk lebih jelas dapat dilihat tampilan halaman karyawan penjualan pada Gambar 3.24.



Gambar 3.24 Halaman Karyawan Penjualan

k. Halaman *Input* Penjualan

Halaman ini merupakan *form* yang digunakan karyawan untuk memasukan data penjualan. Pada halaman ini karyawan akan memasukan data berupa tanggal jual, *grade* yang akan dijual, berat sarang dalam satuan gram, dan harga jual yang ditelah disepakati bersama, nama pembeli dan catatan tambahan jika ada. Untuk lebih jelas dapat dilihat tampilan halaman *input* penjualan pada Gambar 3.25.

The image displays two side-by-side screenshots of a mobile application interface for a sales input page. Both screens have a red header with a back arrow and the text "GEDUNG A".

Left Screenshot (Input Form):

- Tanggal Jual:** 28 / 10 / 2019
- Grade Mangkuk:** Input fields for "Isi Berat" and "Isi Harga".
- Grade Sudut:** Input fields for "Isi Berat" and "Isi Harga".
- Grade Patahan:** Input fields for "Isi Berat" and "Isi Harga".
- Grade Strip:** Input fields for "Isi Berat" and "Isi Harga".
- Total :** A field for the total value.
- Nama Pembeli:** Input field for the buyer's name.
- Buttons:** "Kirim" (Send) and "Ubah" (Change).

Right Screenshot (Summary Table):

Informasi Penjualan

Karyawan : Agus Sofyan
 Tanggal Jual : 28 / 10 / 2019
 Pembeli : Hermansyah

Hasil Penjualan

Grade	Berat	Harga Jual
Mangkuk	800 gr	12.000.000
Sudut	600 gr	7.600.000
Patahan	400 gr	2.200.000
Strip	150 gr	1.800.000
Total	1950 gr	23.600.000

Buttons: "Ubah" (Change)

Gambar 3.25 Halaman *Input* Penjualan

