

BAB III

METODOLOGI PENELITIAN

3.1 Subjek dan Objek Penelitian

Subjek pada penelitian ini yaitu pemilik Ananda Chicken dengan tujuan untuk mengumpulkan informasi yang dibutuhkan. Sedangkan Objek penelitian ini yaitu membangun aplikasi berbasis website Laporan Keuangan pada Ananda Chicken.

3.2 Alat dan Bahan Penelitian

Alat dan Bahan dibutuhkan guna melengkapi proses pada penelitian. Alat yang dibutuhkan dapat berupa *hardware* dan *software*.

3.2.1 Perangkat Keras

Hardware yang digunakan pada penelitian ini yaitu:

1. Laptop Asus X505ZA:
 - a. Prosesor AMD Ryzen 3 3250U
 - b. RAM 12GB
 - c. Storage SSD 256 GB

Laptop Asus X505ZA digunakan dalam melakukan perancangan sistem, *Processor* AMD Ryzen 3 3250U digunakan pada laptop ini, *RAM* 12 GigaByte digunakan pada laptop ini yang berguna dalam menjalankan *tools* yang digunakan pada perancangan sistem, *SSD* 256 GB digunakan untuk menyimpan data sistem dalam penelitian.

3.2.2 Perangkat Lunak

Software yang dipakai guna dapat membuat aplikasi berbasis *Website* pada penelitian ini adalah:

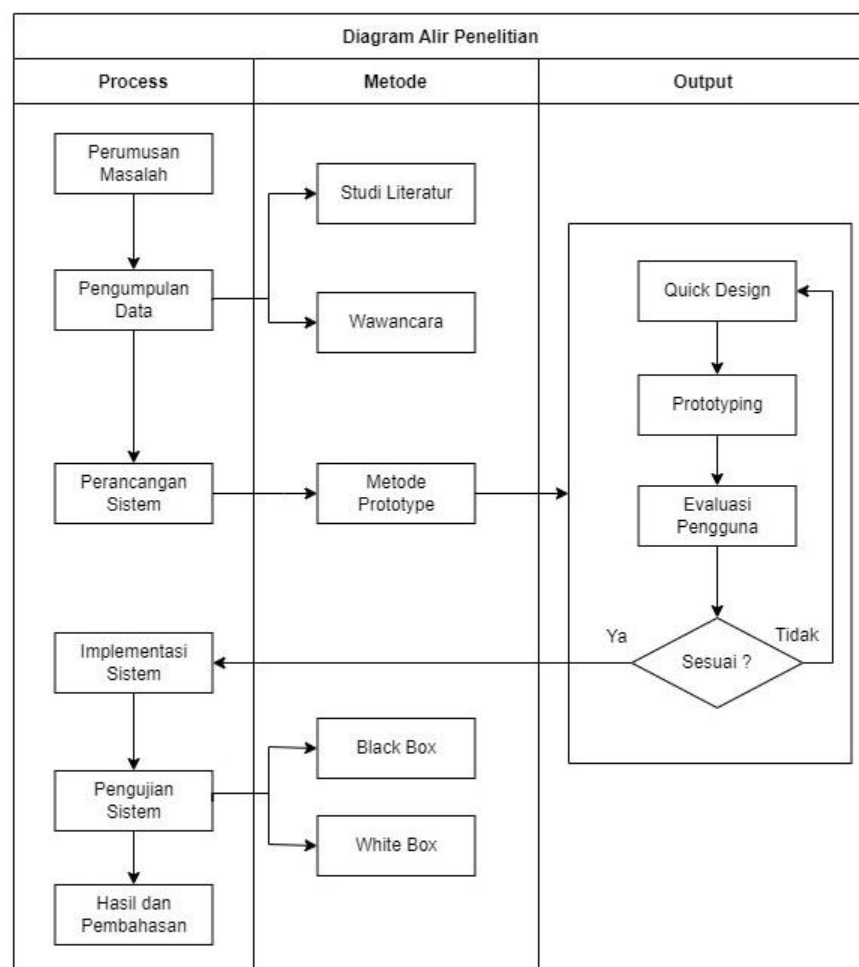
1. Visual Studio Code
2. XAMPP
3. Google Chrome
4. Balsamiq
5. MySQL
6. Microsoft Word

Visual Studio Code versi 1.72.2 digunakan sebagai kode editor, XAMPP versi 3.3.0 dan Google Chrome versi 110.0.5481.104 digunakan untuk menjalankan web pada *localhost*, Balsamiq versi 3.5.17 digunakan untuk membuat desain mockup

aplikasi, MySQL digunakan sebagai manajemen basis data, Microsoft Word versi 2302 (build 16130.20156) digunakan untuk penulisan penyusunan laporan Tugas Akhir.

3.3 Diagram Alir Penelitian

Dalam tahap menyusun proposal penelitian ini terdapat langkah-langkah penelitian yang sudah ditentukan dan menggunakan metode *prototype* pada pengembangan sistem dalam penelitian ini. Diagram alir penelitian ini dapat dilihat pada gambar 3.1 dibawah ini.



Gambar 3. 1 Diagram Alir Penelitian

Gambar 3.1 diagram alir penelitian, menjelaskan tahap-tahap pada penelitian ini yaitu perumusan masalah, pengumpulan data dengan studi literatur dan wawancara, kemudian perancangan sistem dengan pembuatan diagram UML, pengembangan sistem menggunakan metode *prototype* dengan tahapan *quick design*, *prototyping*, evaluasi pengguna lalu implementasi sistem,

kemudian pengujian sistem dalam penelitian menggunakan *blackbox testing* dan *whitebox testing*.

3.3.1 Pengumpulan Data

Pada tahap berikutnya merupakan pengumpulan data dimana peneliti melakukan pengumpulan data berdasarkan permasalahan-permasalahan yang ada. Pengumpulan data yang dilakukan terdapat dua cara, yaitu studi literatur dan wawancara.

Dalam penelitian ini peneliti melakukan studi literatur dengan mengumpulkan referensi dan teori yang membahas tentang *website* laporan keuangan. Data yang diperoleh dari jurnal, artikel, maupun buku. Wawancara akan diperlukan guna mengumpulkan data maupun informasi yang diperlukan guna membangun sistem yang sesuai kebutuhan. Penulis melakukan wawancara dengan pemilik Ananda Chicken. Hasil dari wawancara terdapat pada **Lampiran 1**, dalam pertanyaan nomor 5 bahwa pihak Ananda Chicken terkadang mengalami masalah yaitu hilangnya buku catatan laporan keuangan yang juga menimbulkan terganggunya proses pengolahan data dan rekap keuangan. Serta pada pertanyaan nomor 6 menunjukkan bahwa pihak Ananda Chicken berkenan jika dibuatkan aplikasi laporan keuangan berbasis web.

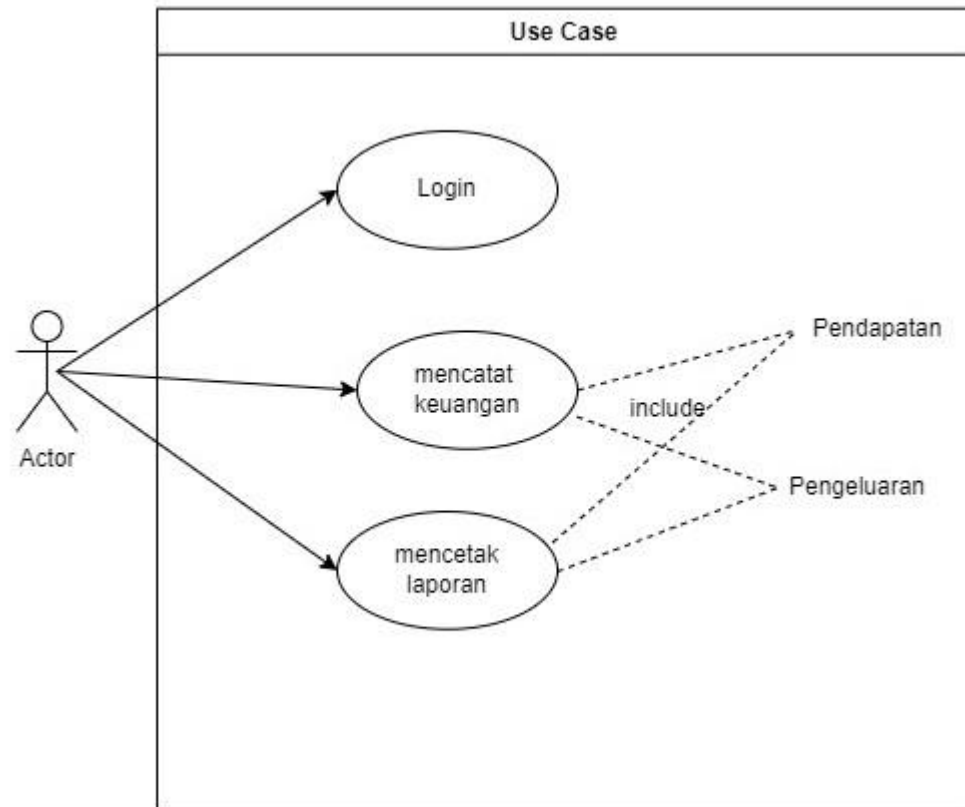
3.3.2 Perancangan Sistem

Tahap perancangan sistem yaitu menggunakan metode *prototype* dengan tahap *quick design* dengan pembuatan diagram UML yang terdiri dari *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram* dan pembuatan *wireframe*. Selanjutnya tahap *prototyping* kemudian evaluasi pengguna mengenai pembangunan prototipe apakah sudah sesuai dengan kebutuhan pengguna maupun perlu adanya perubahan. Setelah melalui tahap evaluasi pengguna selanjutnya ke tahap Implementasi sistem yaitu:

3.3.3.1 Quick Design

Quick Design dilakukan dengan pembuatan *Unified Modeling Language* (UML) sebagai berikut.

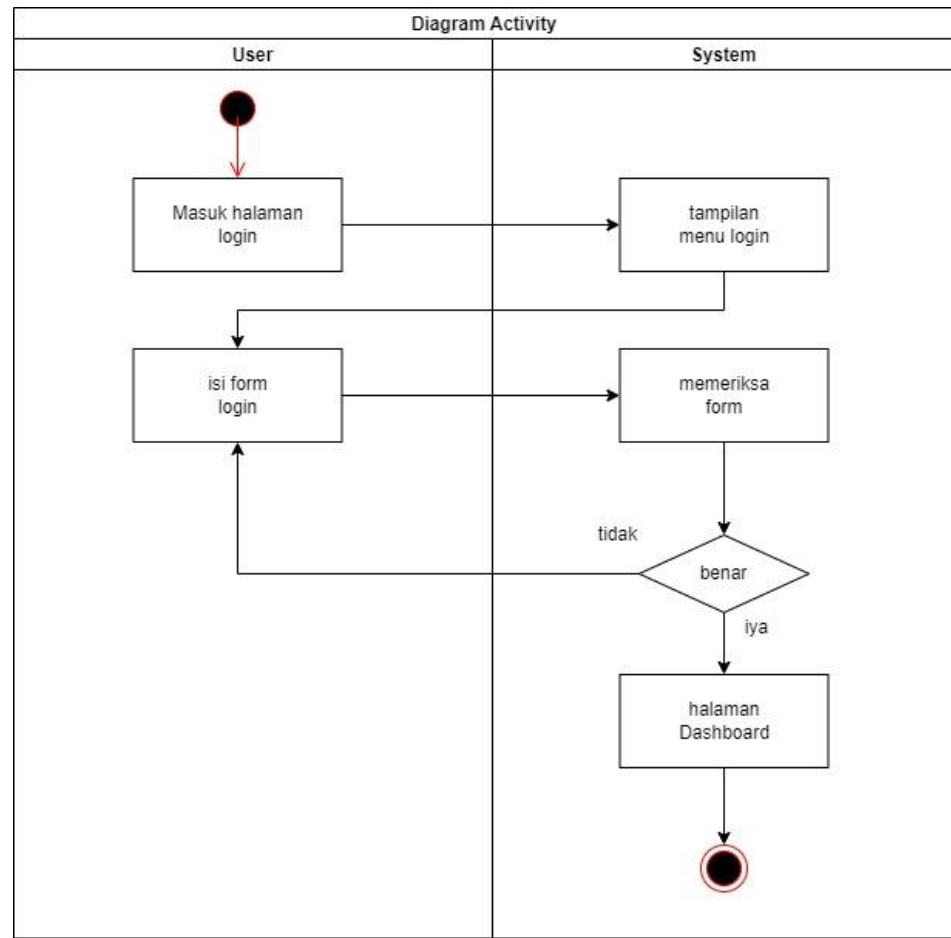
a. *Use Case Diagram*



Gambar 3. 2 *Use Case Diagram Website Laporan Keuangan*

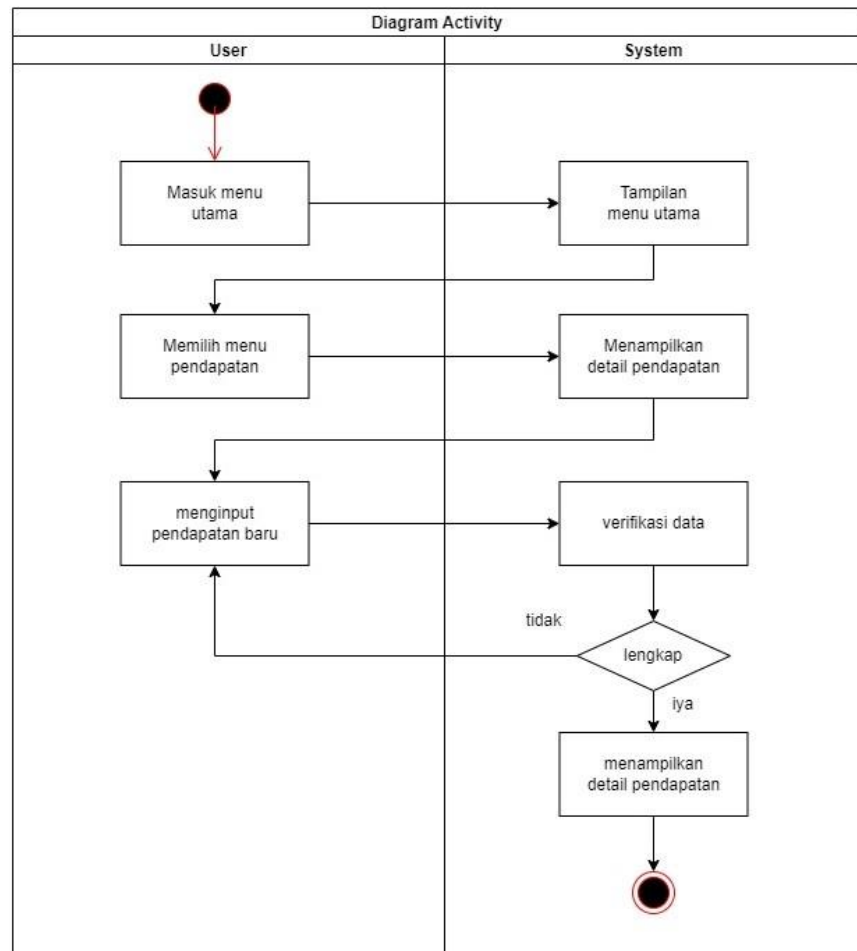
Gambar *Use Case Diagram* diatas dapat dijelaskan bahwa gambar Actor sebagai pengguna. Actor dapat melakukan *Login*, mencatat keuangan, mencetak laporan keuangan, termasuk pendapatan dan pengeluaran.

b. *Activity Diagram*



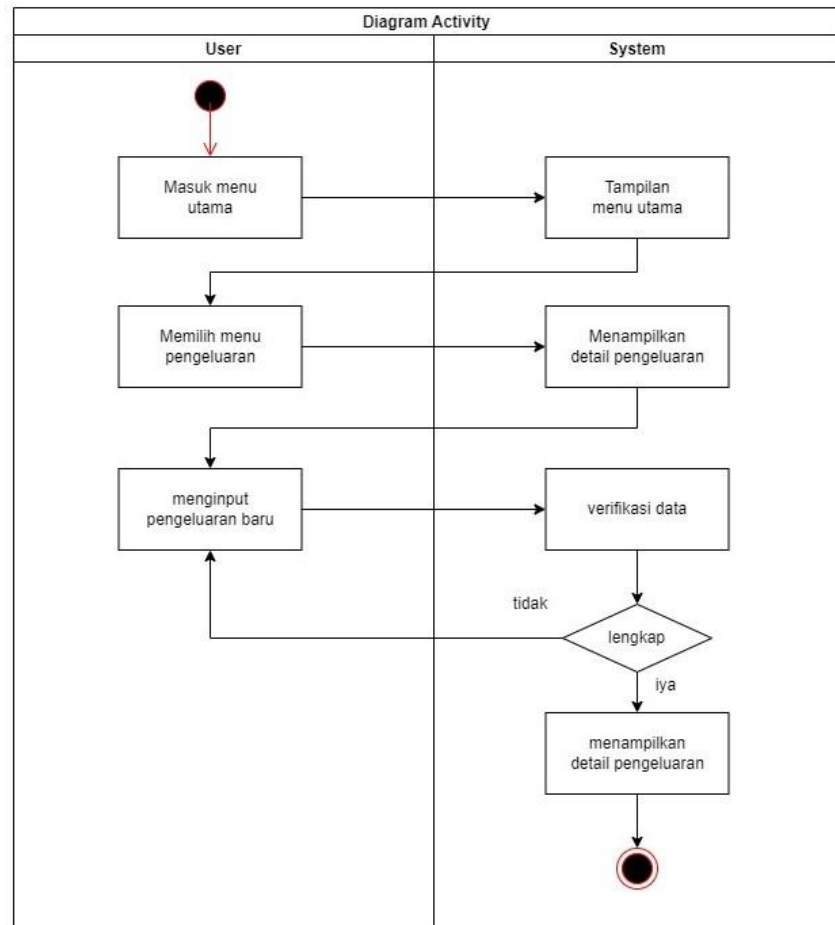
Gambar 3. 3 Activity Diagram Proses Login

Dari gambar Activity Diagram Proses Login diatas menjelaskan User dapat pertama masuk halaman *login*, halaman *login* akan ditampilkan oleh sistem, kemudian *user* mengisi *field* pada halaman *login*, selanjutnya *form* dicocokkan oleh sistem jika data yang dimasukkan sudah benar akan masuk ke halaman *Dashboard*, jika data yang dimasukkan *user* pada *form* salah maka sistem kembali menampilkan *form login* yang diisi oleh *user*.



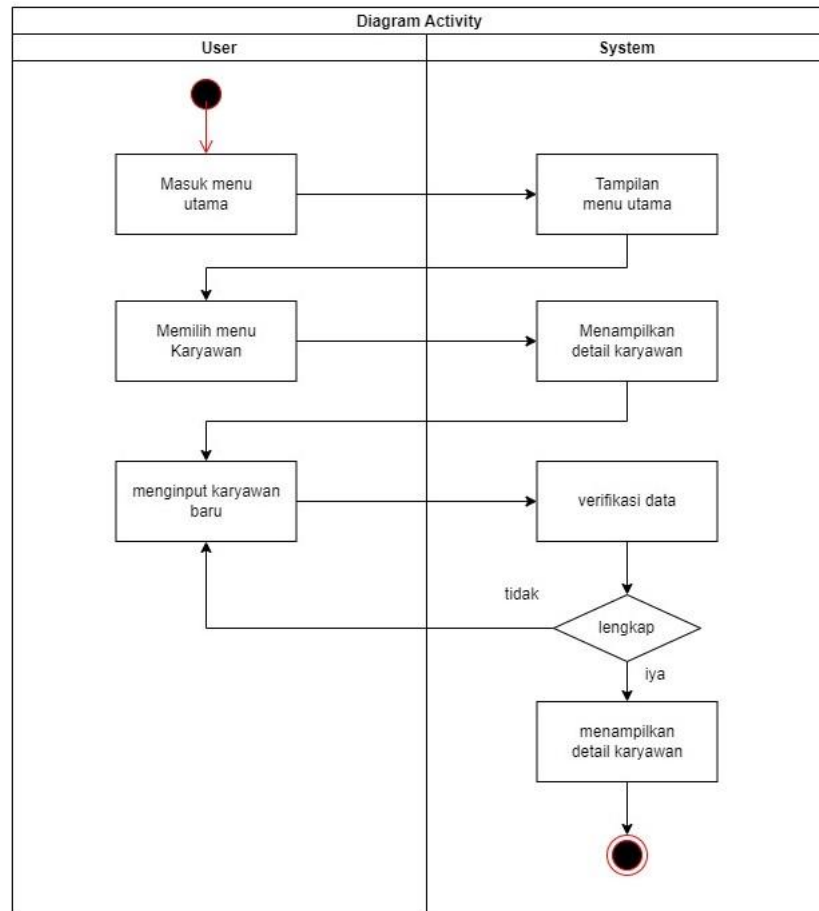
Gambar 3. 4 *Activity Diagram* Pendapatan

Dari gambar *Activity Diagram* Pendapatan diatas menjelaskan aktivitas *user* dimulai dari masuk tampilan halaman utama, menu utama ditampilkan oleh sistem, selanjutnya *user* memilih halaman pendapatan, detail pendapatan ditampilkan oleh sistem, selanjutnya *user* menginput data pendapatan baru, sistem akan mendeteksi apakah *field* yang dibutuhkan saat memasukkan data sudah sesuai, jika sudah sesuai sistem akan menampilkan detail data pendapatan terbaru.



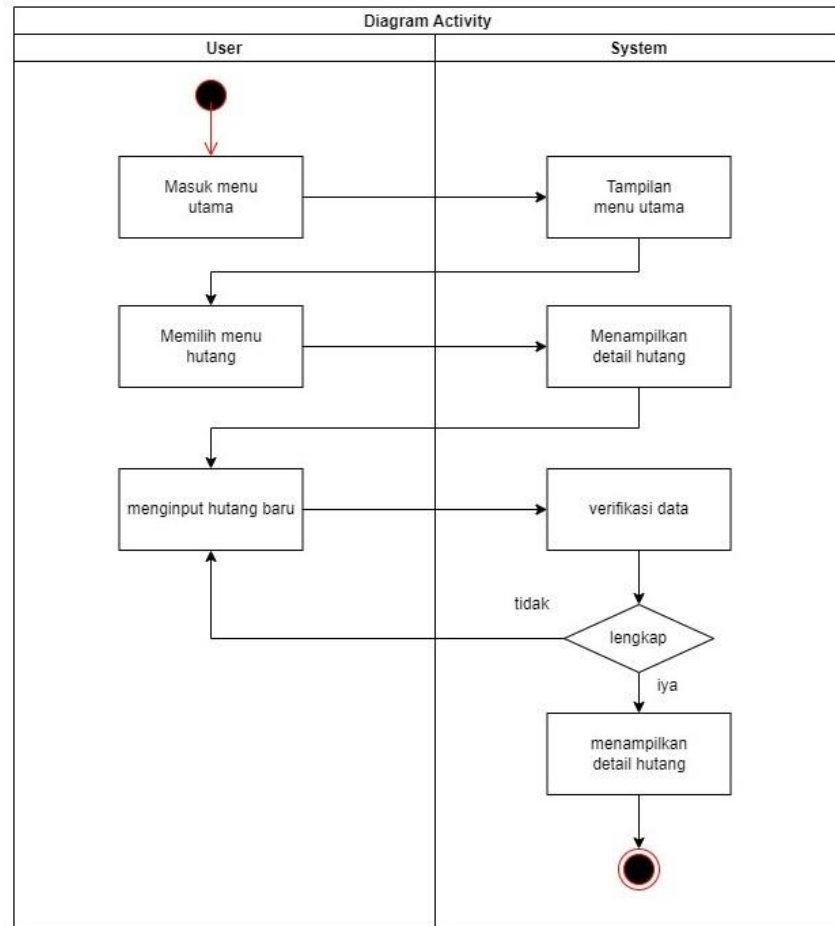
Gambar 3. 5 Activity Diagram Pengeluaran

Dari gambar *Activity Diagram* Pengeluaran diatas menjelaskan aktivitas *user* dimulai dari masuk tampilan halaman utama, menu utama ditampilkan oleh sistem, selanjutnya *user* memilih halaman pengeluaran, detail pengeluaran ditampilkan oleh sistem, selanjutnya *user* menginput data pengeluaran baru, sistem akan mendeteksi apakah *field* yang dibutuhkan saat memasukkan data sudah sesuai, jika sudah sesuai sistem akan menampilkan detail data pengeluaran terbaru.



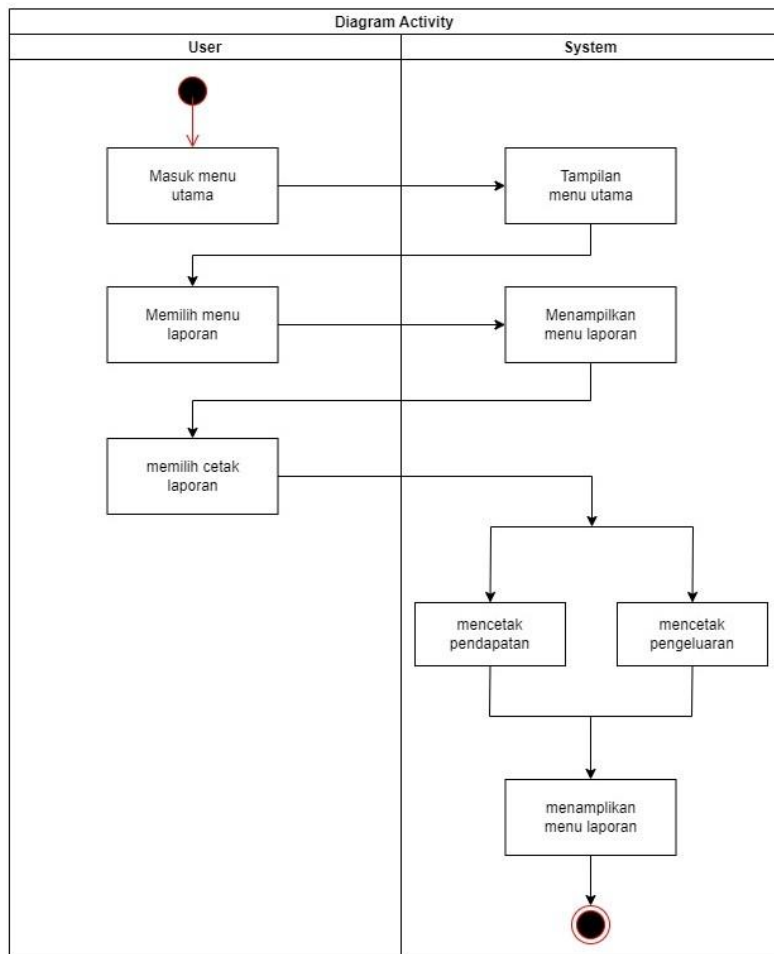
Gambar 3. 6 *Activity Diagram* Karyawan

Dari gambar *Activity Diagram* Karyawan diatas menjelaskan aktivitas *user* dimulai dari masuk tampilan halaman utama, menu utama ditampilkan oleh sistem, selanjutnya *user* memilih halaman karyawan, detail karyawan ditampilkan oleh sistem, selanjutnya *user* menginput data karyawan baru, sistem akan mendeteksi apakah *field* yang dibutuhkan saat memasukkan data sudah sesuai, jika sudah sesuai sistem akan menampilkan detail data karyawan terbaru.



Gambar 3. 7 Activity Diagram Hutang

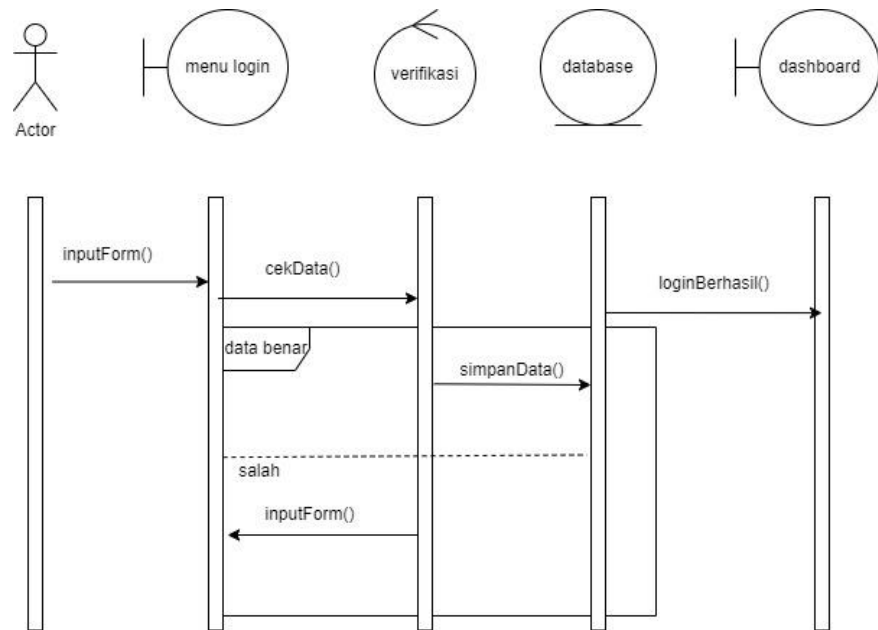
Dari gambar *Activity Diagram* Hutang diatas menjelaskan aktivitas *user* dimulai dari masuk tampilan halaman utama, menu utama ditampilkan oleh sistem, selanjutnya *user* memilih halaman hutang, detail hutang ditampilkan oleh sistem, selanjutnya *user* menginput data hutang baru, sistem akan mendeteksi apakah *field* yang dibutuhkan saat memasukkan data sudah sesuai, jika sudah sesuai sistem akan menampilkan detail data hutang terbaru.



Gambar 3. 8 Activity Diagram Cetak Laporan

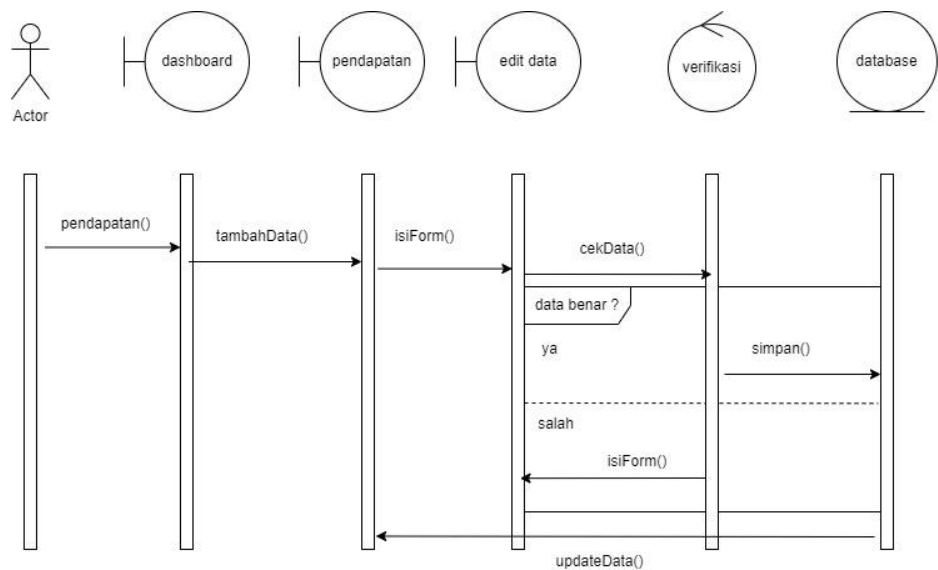
Dari gambar Activity Diagram Cetak Laporan diatas menjelaskan aktivitas *user* ketika ingin mencetak laporan keuangan, dimulai dari *user* masuk tampilan halaman utama, menu utama ditampilkan oleh sistem, selanjutnya *user* memilih menu laporan, menu laporan ditampilkan sistem, selanjutnya *user* mencetak laporan, pada sistem terdapat cetak laporan pendapatan dan laporan pengeluaran, lalu sistem menampilkan menu laporan, *output* cetak laporan dari sistem berupa data excel.

c. Sequence Diagram



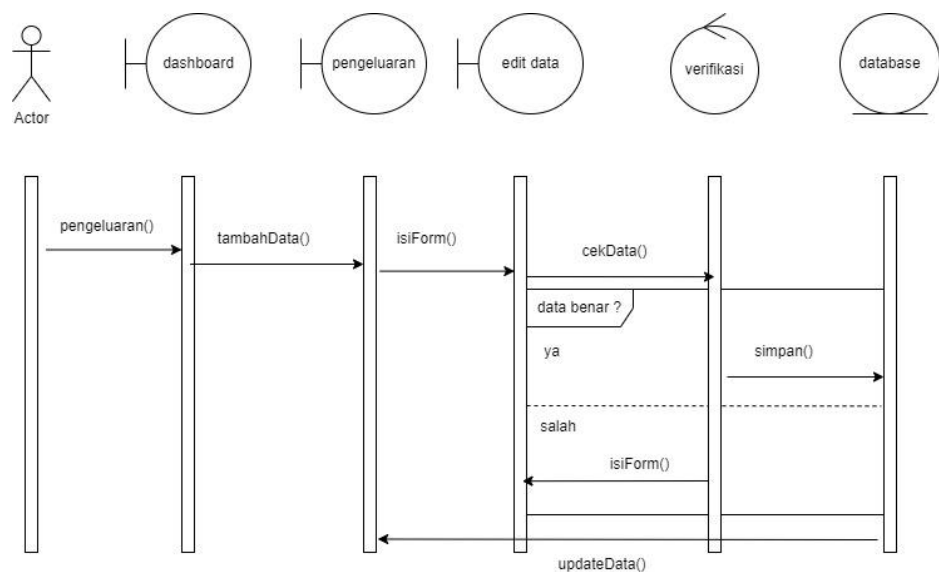
Gambar 3. 9 Sequence Diagram Login

Dari gambar *Sequence Diagram Login* diatas menjelaskan pengguna melakukan login pada halaman login, dengan mengisi *form* pada halaman login, selanjutnya sistem memverifikasi data apakah *field* sudah terisi semua dan sudah sesuai pada basis data. Jika sesuai maka dilanjutkan ke halaman utama, sebaliknya jika data yang diisi belum sesuai maka dikembalikan ke menu login.



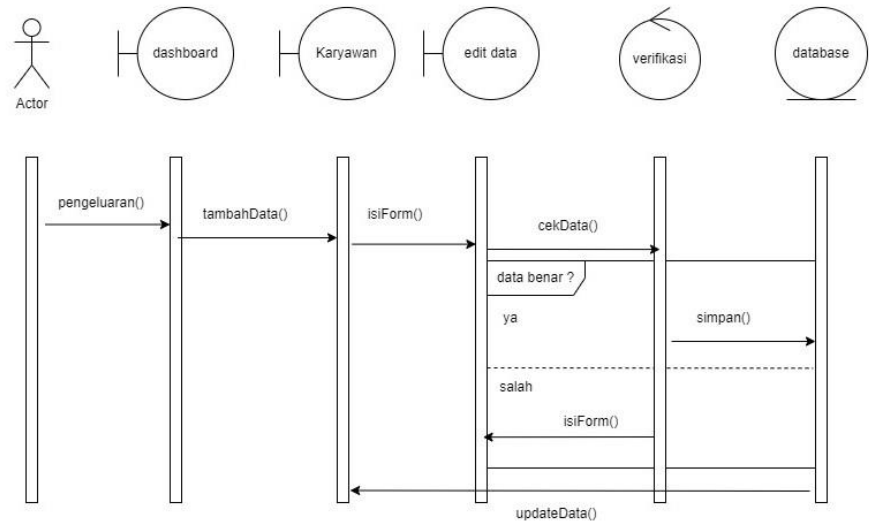
Gambar 3. 10 Sequence Diagram Pendapatan

Dari gambar *Sequence Diagram* Pendapatan diatas menjelaskan pengguna memilih menu pendapatan keuangan pada halaman *dashboard*, selanjutnya menambahkan data pendapatan pada menu tambah pendapatan, kemudian mengisi *form* pada menu edit data. Selanjutnya sistem akan mencocokkan data yang diisi apakah *field* yang dibutuhkan sudah terisi semua atau tidak, jika iya data yang ditambahkan disimpan pada basis data dan sistem memperbarui pendapatan, jika tidak maka dikembalikan pada menu edit data untuk mengisi kembali data dengan benar.



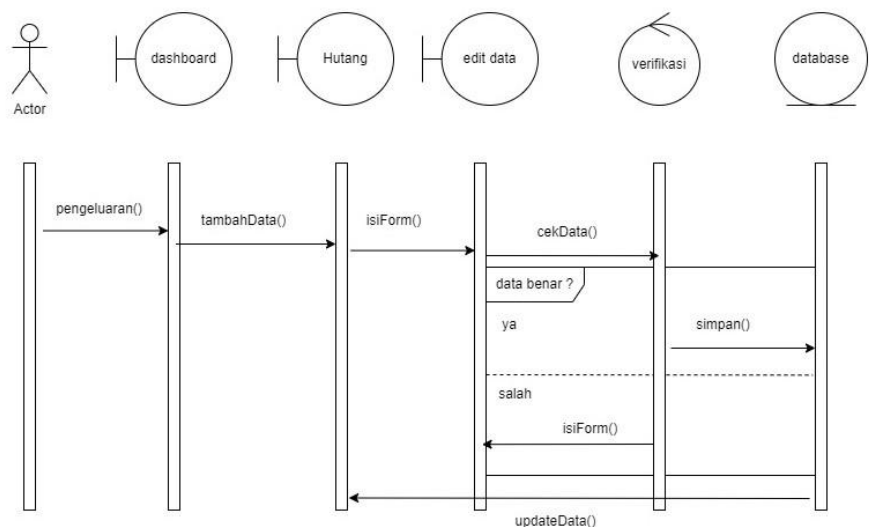
Gambar 3. 11 *Sequence Diagram* Pengeluaran

Dari gambar *Sequence Diagram* Pengeluaran diatas menjelaskan pengguna memilih menu pengeluaran keuangan pada halaman *dashboard*, selanjutnya menambahkan data pengeluaran pada menu tambah pengeluaran, kemudian mengisi *form* pada menu edit data. Selanjutnya sistem akan mencocokkan data yang diisi apakah *field* yang dibutuhkan sudah terisi semua atau belum, jika iya data yang ditambahkan akan tersimpan dalam database dan sistem memperbarui menu pengeluaran, jika tidak maka dikembalikan pada menu edit data untuk mengisi kembali data dengan benar.



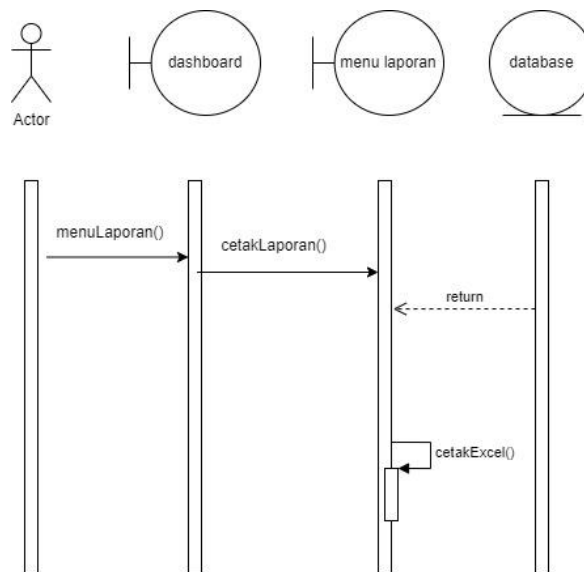
Gambar 3. 12 *Sequence Diagram Karyawan*

Dari gambar *Sequence Diagram Karyawan* diatas menjelaskan pengguna memilih menu karyawan pada halaman *dashboard*, selanjutnya menambahkan data karyawan pada menu tambah karyawan, kemudian mengisi *form* pada menu edit data karyawan. Selanjutnya sistem akan mencocokkan data yang diisi apakah *field* yang dibutuhkan sudah terisi semua atau belum, jika iya data yang ditambahkan akan tersimpan dalam database dan sistem memperbarui menu karyawan, jika tidak maka dikembalikan pada menu edit data untuk mengisi kembali data dengan benar.



Gambar 3. 13 *Sequence Diagram Hutang*

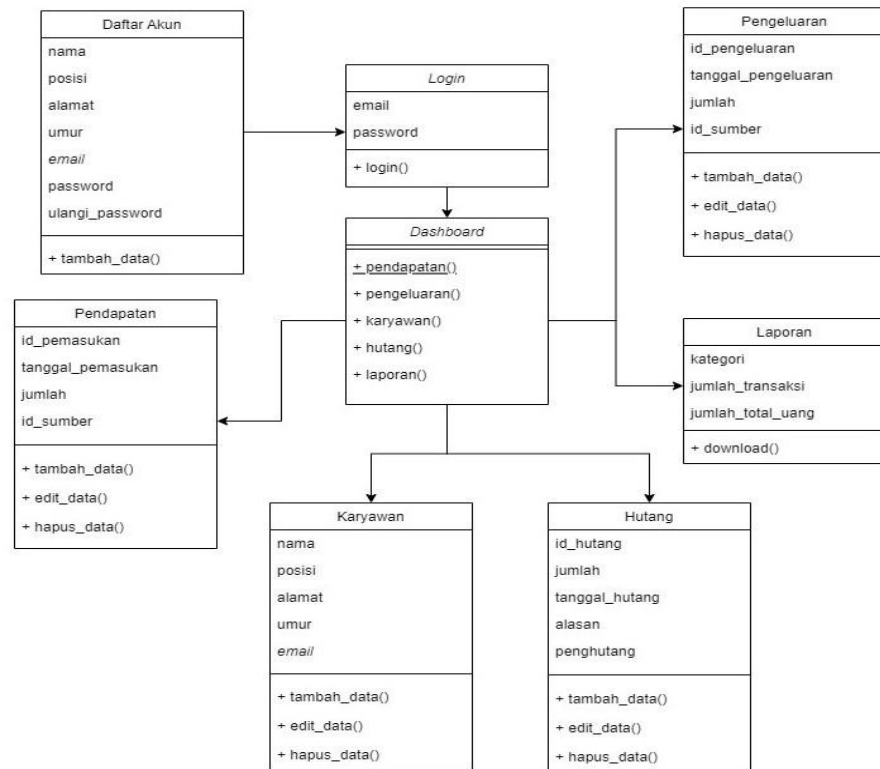
Dari gambar *Sequence Diagram* Hutang diatas menjelaskan pengguna memilih menu hutang keuangan pada halaman *dashboard*, kemudian menambahkan data hutang pada menu tambah hutang, kemudian mengisi *form* pada menu edit data. Selanjutnya sistem akan mencocokkan data yang diisi apakah *field* yang dibutuhkan sudah terisi semua atau belum, jika iya data yang ditambahkan akan tersimpan dalam database dan sistem memperbarui menu hutang, jika tidak maka dikembalikan pada menu edit data untuk mengisi kembali data dengan benar.



Gambar 3. 14 *Sequence Diagram* Cetak Laporan

Dari gambar *Sequence Diagram* Cetak Laporan diatas menjelaskan pengguna memilih menu laporan pada halaman *dashboard*, kemudian pengguna mencetak laporan keuangan pada menu laporan, pada saat memilih cetak, maka sistem akan mengambil data pada *database* untuk dicetak pada menu laporan.

d. Class Diagram

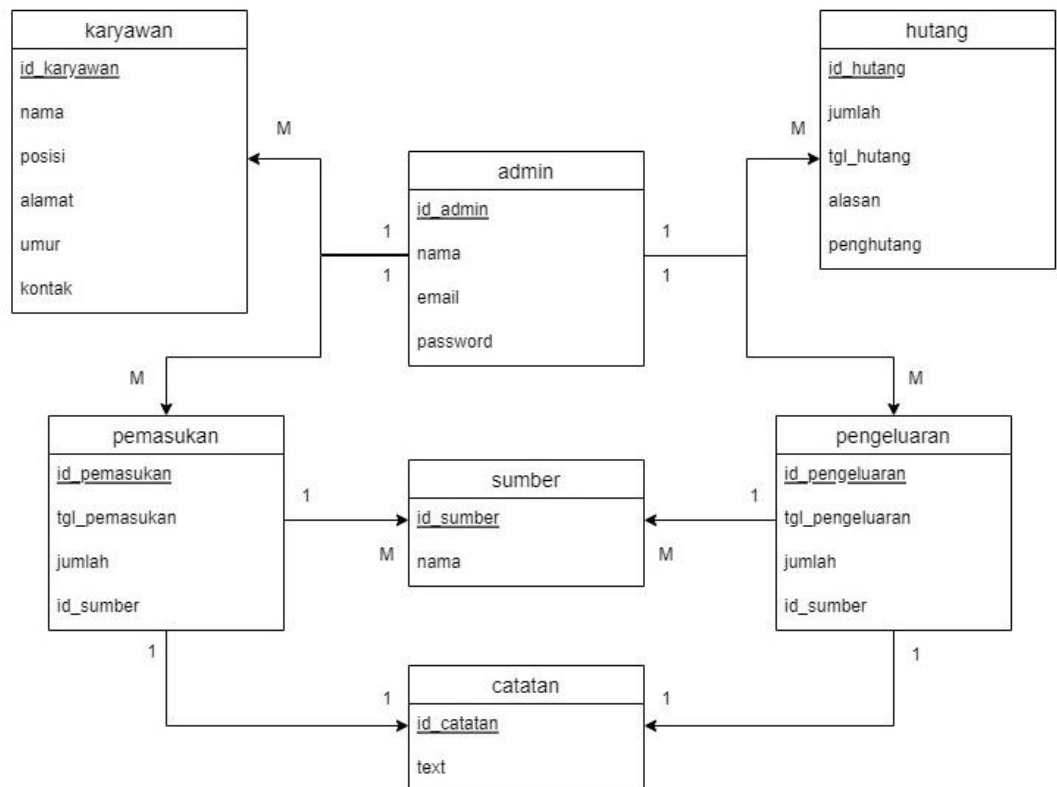


Gambar 3. 15 Class Diagram Website Laporan Keuangan

Class Diagram yang dibuat dari penelitian ini memiliki *Class* Daftar Akun yang berisikan parameter nama, posisi, alamat, umur, *email*, dan *password* dengan function *tambah_data()*. *Class* Daftar Akun memiliki relasi dengan *Class* Login yang berisikan parameter berupa *username* dan *password* dengan function *login()*. *Class* Login memiliki relasi dengan *Class* Dashboard. Dalam *Class* Dashboard memiliki function *pendapatan()*, *pengeluaran()*, *karyawan()*, *hutang()* dan *laporan()*, *Class* Dashboard memiliki relasi dengan *Class* Pendapatan, *Class* Pengeluaran, *Class* Karyawan, *Class* Hutang yang masing-masing memiliki function *tambah_data()*, *edit_data()*, *hapus_data()*, dan *download()* yang terdapat pada *Class* Laporan.

e. Entity Relationship Diagram (ERD)

Peneliti membuat *Entity Relationship Diagram* untuk mengetahui relasi database yang akan dirancang.

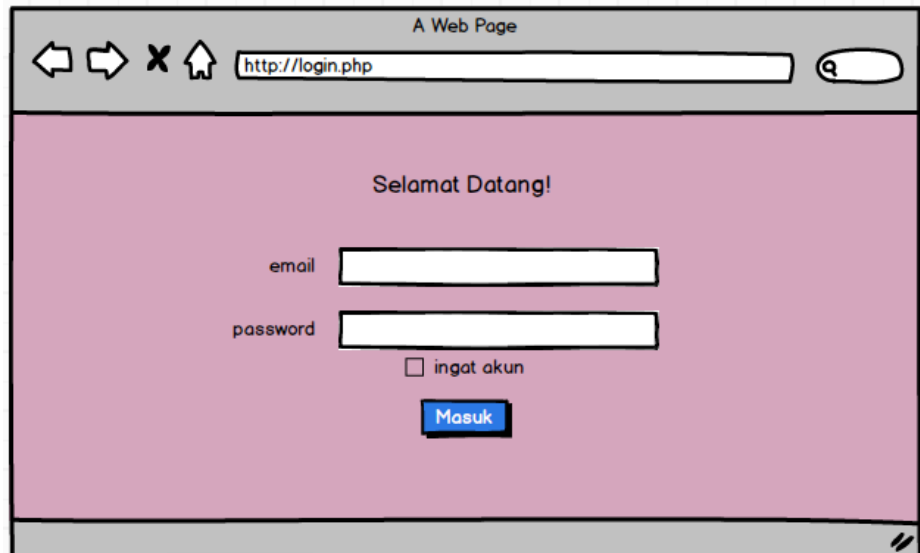


Gambar 3. 16 Desain *Entity Relationship Diagram*

Gambar 3.16 merupakan *Entity Relationship Diagram* laporan keuangan Ananda Chicken yang memiliki entitas admin, karyawan, pemasukan, pengeluaran, hutang, catatan dan sumber. Entitas admin terdiri dari atribut *id_admin*, *password*, *nama*, dan *email* mempunyai relasi *one to many*, dimana *admin* menambahkan karyawan, pemasukan, pengeluaran dan hutang. Entitas pemasukan dan pengeluaran mempunyai relasi *one to one*, dimana pemasukan dan pengeluaran memiliki catatan dan sumber.

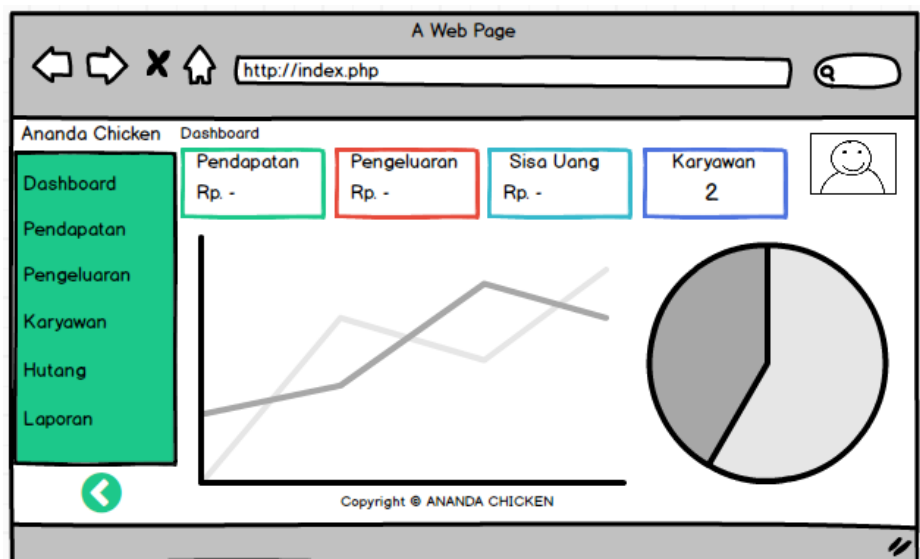
f. Wireframe

Peneliti membuat desain antarmuka atau solusi desain dirancang untuk menggambarkan bentuk aplikasi laporan keuangan yang akan dibangun. *Wireframe* pada penelitian ini dapat dilihat pada Gambar 3.17 berikut.



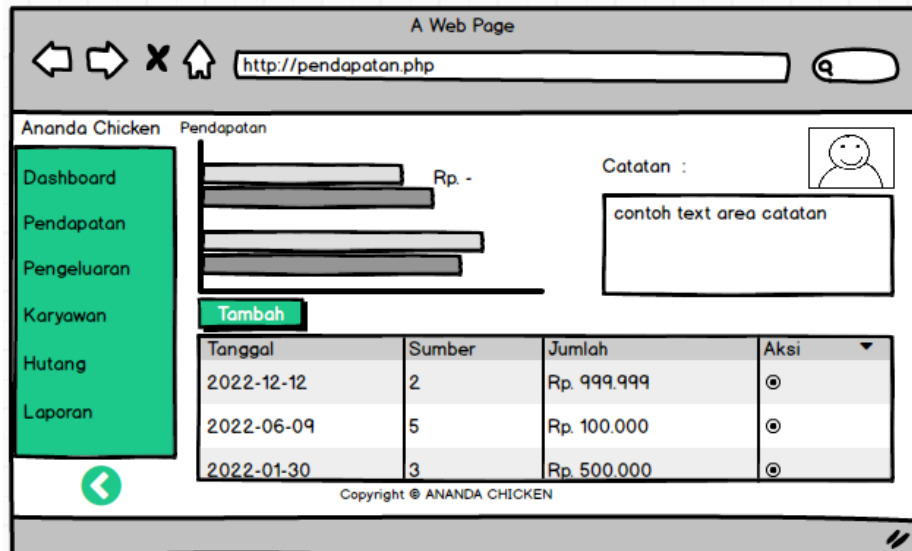
Gambar 3. 17 Wireframe Login

Pada gambar *wireframe* halaman *login* diatas memiliki form *login* dengan berisi *email* dan *password* serta *button* untuk masuk. *wireframe* halaman *dashboard* pada penelitian ini digambarkan pada gambar 3.18 berikut.



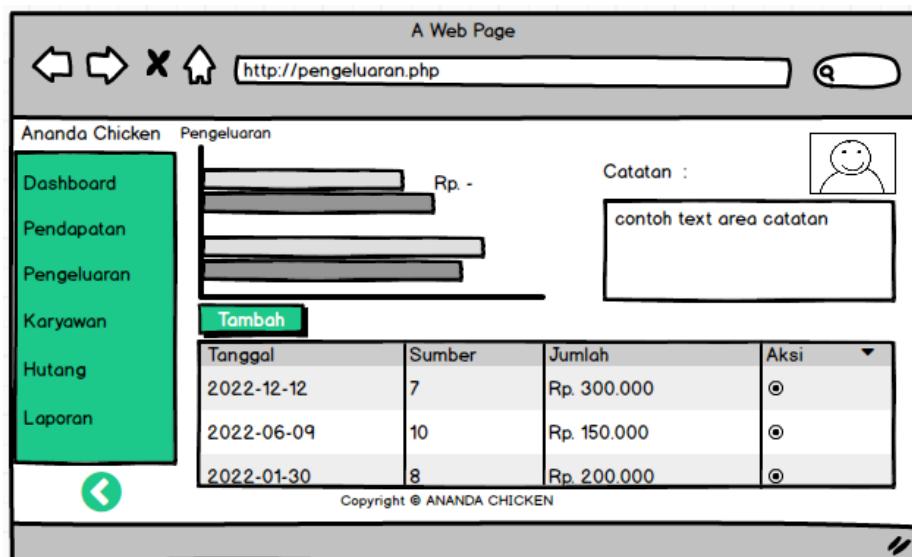
Gambar 3. 18 Wireframe Dashboard

Pada gambar *wireframe* halaman *dashboard* merupakan tampilan utama yang berisi informasi mengenai jumlah pendapatan, jumlah pengeluaran, sisa uang dan jumlah karyawan. *Wireframe* halaman pendapatan pada penelitian ini digambarkan pada gambar 3.19 berikut.



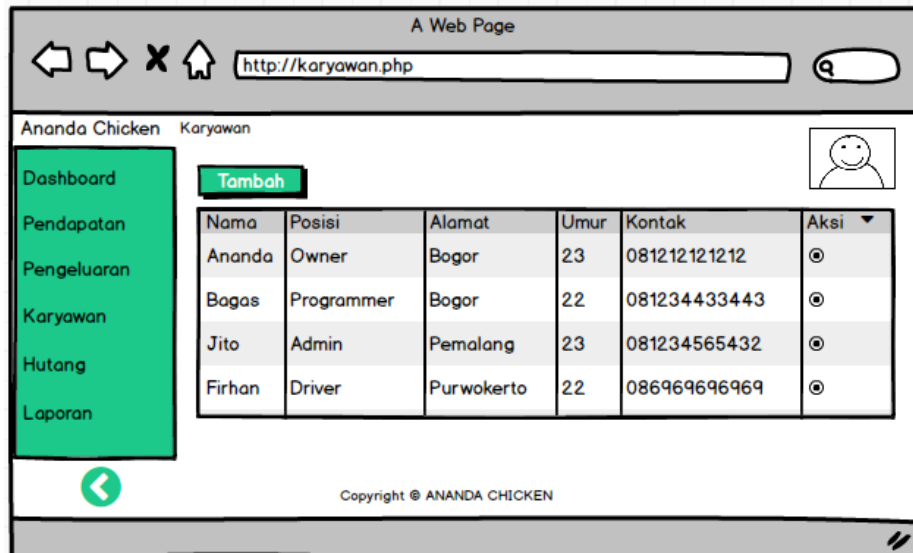
Gambar 3. 19 Wireframe Pendapatan

Pada gambar *wireframe* halaman pendapatan menampilkan informasi mengenai jumlah data pendapatan pada bagan dan tombol untuk menambah data pendapatan. *wireframe* halaman pengeluaran pada penelitian ini digambarkan pada gambar 3.20 berikut.



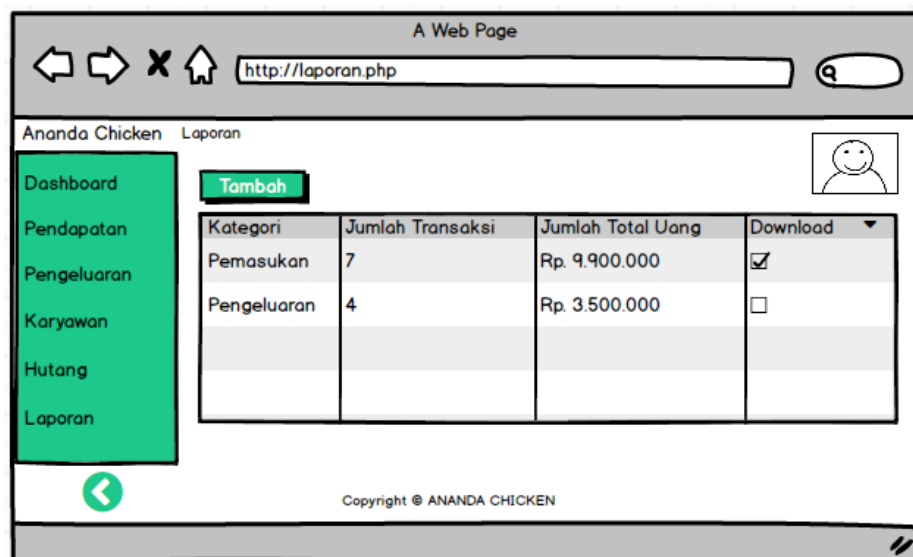
Gambar 3. 20 Wireframe Pengeluaran

Pada gambar *wireframe* halaman pengeluaran menampilkan informasi mengenai jumlah data pengeluaran pada bagan dan tombol untuk menambah data pengeluaran. *Wireframe* halaman karyawan pada penelitian ini digambarkan pada gambar 3.21 berikut



Gambar 3. 21 Wireframe Karyawan

Pada gambar *wireframe* halaman karyawan menampilkan tabel yang berisi daftar karyawan dan informasi karyawan seperti nama, posisi, alamat, umur dan kontak. *Wireframe* halaman laporan pada penelitian ini digambarkan pada gambar 3.22 berikut

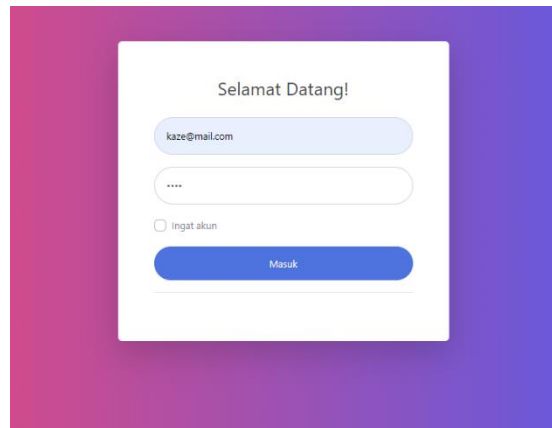


Gambar 3. 22 Wireframe Laporan

Pada gambar *wireframe* halaman laporan menampilkan tabel yang berisi detail laporan dengan kategori laporan pemasukan dan laporan pengeluaran dan tombol *download* laporan untuk menyimpan data laporan.

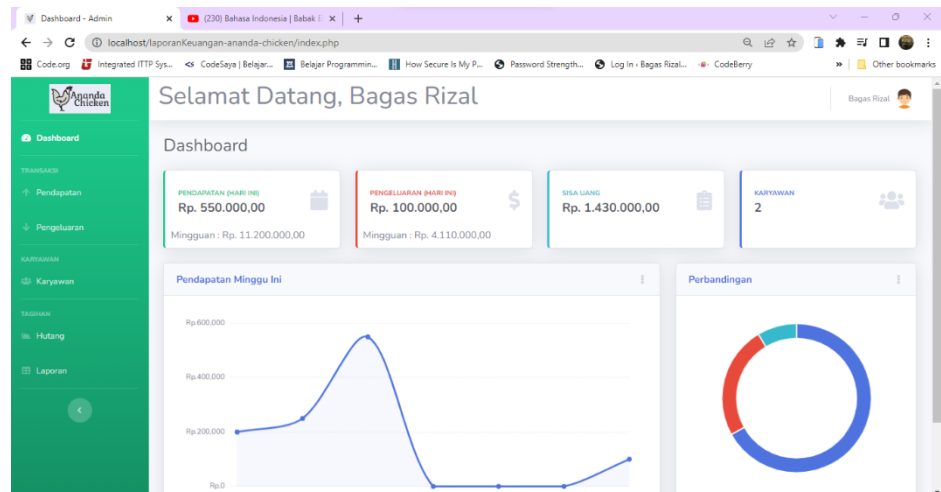
3.3.3.2 Prototyping

Pada proses membangun prototipe, peneliti membuat tahapan implementasi kebutuhan pengguna, pemodelan diagram, dan desain *wireframe* ke dalam bahasa pemrograman, yang kemudian dijadikan sebuah aplikasi.



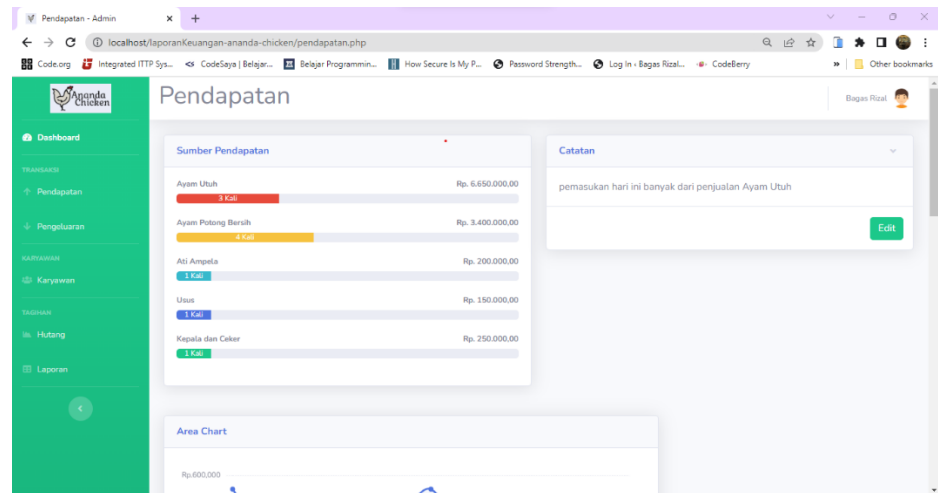
Gambar 3. 23 Prototyping Login

Gambar 3.23 merupakan hasil *prototype* halaman *login* dimana pengguna diwajibkan melengkapi *form* untuk dapat masuk kedalam halaman utama.

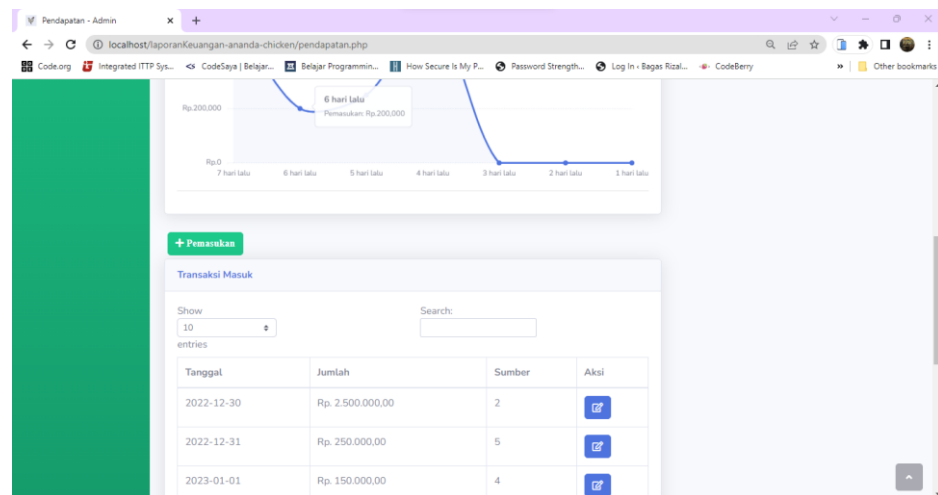


Gambar 3. 24 Prototyping Dashboard

Gambar 3.24 merupakan hasil *prototype* halaman *dashboard* ditampilkan pertama kali ketika user berhasil *login*.

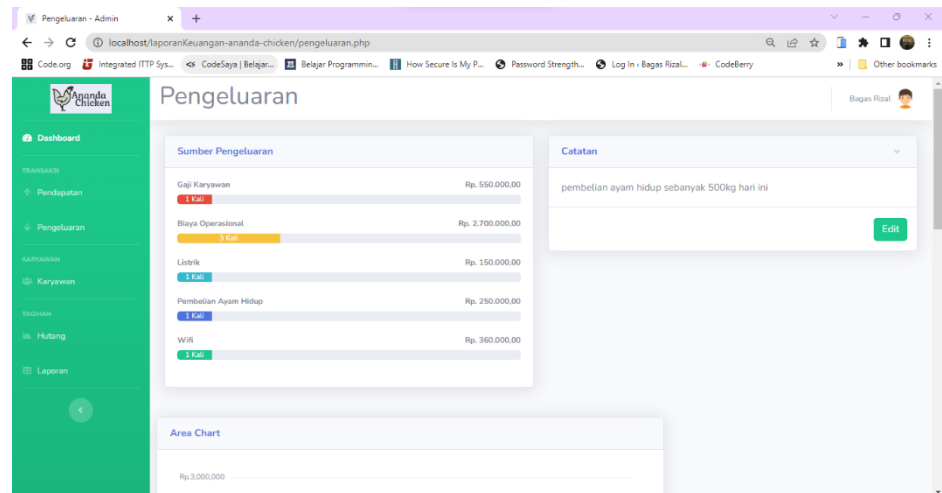


Gambar 3. 25 *Prototyping* Pendapatan

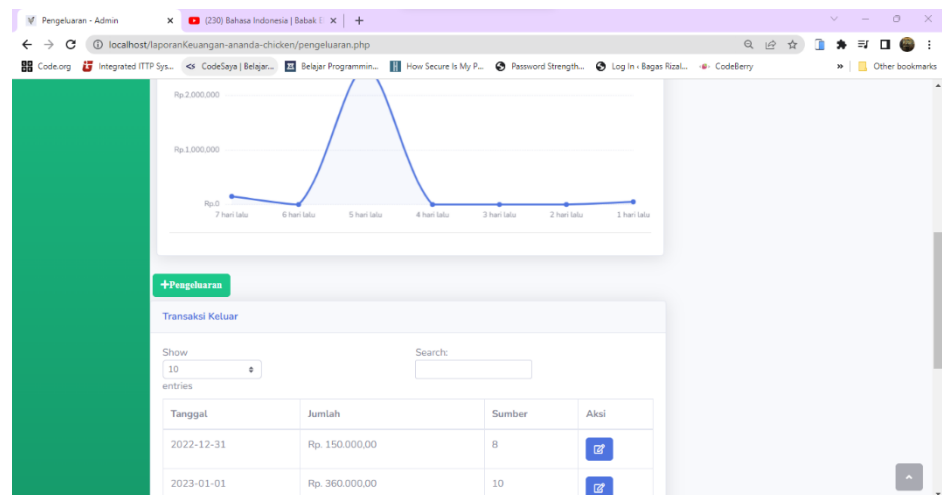


Gambar 3. 26 *Prototyping* Pendapatan 2

Gambar 3.25 dan gambar 3.26 merupakan hasil *prototype* halaman pendapatan yang menampilkan grafik dan pendapatan diterima terbanyak dari sumber tertentu, pada halaman ini pengguna juga dapat menambah data dan mengubah data pendapatan.

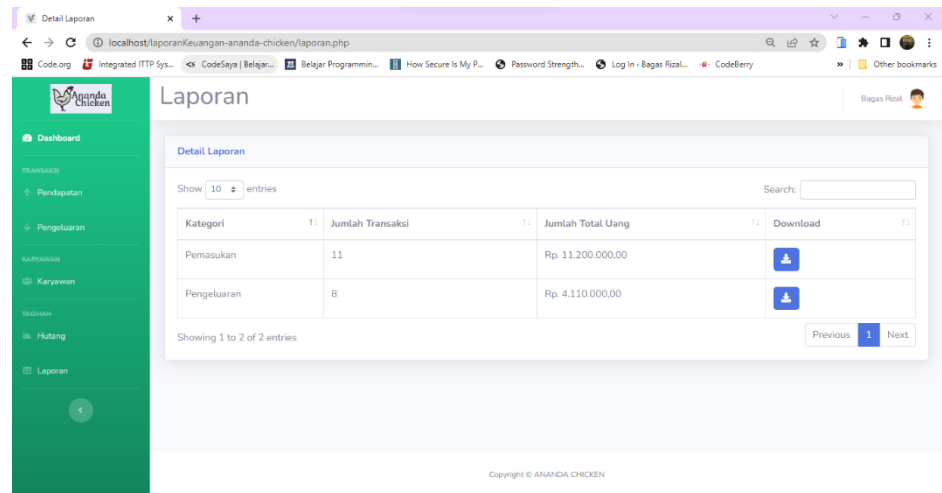


Gambar 3. 27 *Prototyping* Pengeluaran



Gambar 3. 28 *Prototyping* Pengeluaran 2

Gambar 3.27 dan gambar 3.28 merupakan hasil *prototype* halaman pengeluaran yang menampilkan grafik dan pengeluaran diterima terbanyak dari sumber tertentu, pada halaman ini pengguna juga dapat menambah data dan mengubah data pengeluaran.



Gambar 3. 29 *Prototyping* Laporan

Gambar 3.29 merupakan hasil *prototype* halaman laporan yang menampilkan keuangan pendapatan dan pengeluaran, pada halaman laporan juga dapat mendownload laporan ke dalam format excel.

3.3.3.3 Evaluasi Pengguna

Pihak Ananda Chicken mengevaluasi *prototype* yang sudah dibuat dan mendapatkan *feedback* dari pihak Ananda Chicken untuk mendapatkan perbaikan apakah terdapat kelemahan terhadap desain *prototype* atau tidak, untuk selanjutnya akan menjadi bahan evaluasi. Jika pihak Ananda Chicken tidak setuju atau tidak puas dengan prototypenya maka akan dilakukan pengulangan kembali pada tahap perancangan sistem dan perbaikan *prototype*, sedangkan jika pihak Ananda Chicken sudah puas dengan prototypenya selanjutnya ke tahap pembuatan aplikasi. *Feedback* penelitian ini disajikan dalam Tabel 3.1 berikut.

Tabel 3. 1 Evaluasi Pengguna

Komponen	Evaluasi Pengguna	Kesimpulan
Halaman <i>Login</i>	Menambahkan fitur untuk mendaftar akun baru	Selesai
Halaman <i>Dasboard</i>	Menambahkan tombol download laporan dengan hasil yaitu laporan pendapatan dan pengeluaran	Selesai
Navigasi Sidebar	Menyarankan untuk merubah warna dengan yang lebih baru	Selesai

Komponen	Evaluasi Pengguna	Kesimpulan
Fitur Halaman Edit <i>Profile</i>	Menambahkan fitur edit profil untuk mengedit profil pengguna	Selesai
Fitur Halaman Hutang	Menambahkan Halaman hutang untuk membuat daftar hutang	Selesai
Fitur Catatan	Menambahkan fitur tulis catatan pada halaman pendapatan dan pengeluaran	Selesai

3.3.3.4 Implementasi dan Pemeliharaan

Pada fase akhir ini peneliti mengimplementasikan desain ke dalam sistem berdasarkan *prototype* akhir. *Prototype* yang sudah disetujui oleh pihak Ananda Chiken pada tahap Evaluasi dan sudah siap akan mulai dibuatkan, selanjutnya sistem akan diuji dan dikirimkan kepada pengguna. Berikutnya merupakan fase pemeliharaan untuk menjaga agar sistem tetap berjalan lancar dan bebas kendala.

3.3.3 Tahap Pengujian Sistem

Tahap pengujian sistem untuk mengetahui kesesuaian sistem dengan tahapan perencanaan yang telah dibuat menggunakan metode pengujian *black box testing*, yaitu pengujian perangkat lunak yang berfokus pada persyaratan fungsional perangkat lunak, pengujian *black box* menggunakan teknik *Equivalence Partitioning* yaitu melakukan pembagian menjadi beberapa partisi dari *input* data, dengan melakukan skenario pengujian fungsionalitas yang dibagi menjadi beberapa bagian seperti melakukan *login* dan *logout*, *download* laporan, menambahkan data, menghapus data dan menghapus data. Pengujian sistem berikutnya menggunakan *white box testing*, yaitu pengujian perangkat lunak di bagian aliran kode program yang menentukan apakah *input* dan *output* memenuhi standar yang diperlukan. Pengujian *white box* dilakukan menggunakan teknik *basis path testing* dengan menggambarkan aliran kontrol logika program menggunakan *flowchart* dan *flow graph*, selanjutnya untuk menentukan apakah program tersebut sederhana atau kompleks berdasarkan logika menggunakan perhitungan *Cyclomatic Complexity* berdasarkan teknik *basis path*, pengujian dilakukan pada proses *login*, proses pendapatan, proses

pengeluaran dan proses cetak laporan. Skenario pengujian black box dapat dilihat pada tabel 3.2 berikut.

Tabel 3. 2 Skenario Pengujian *Black box*

No.	Skenario Pengujian	Realisasi yang diharapkan	Hasil Pengujian	Kesimpulan
1	Klik Daftar Akun	Memunculkan <i>form</i> daftar akun	<i>Form</i> daftar akun berhasil dimuat	
2	Melakukan <i>login</i> pada website	Masuk ke dalam halaman <i>dashboard</i>	Fungsi <i>login</i> dapat berjalan sesuai yang diharapkan	
3	Melakukan <i>logout</i> pada website	Keluar dari website dan kembali ke halaman login	Fungsi <i>logout</i> dapat berjalan sesuai yang diharapkan	
4	<i>Download</i> laporan pada halaman <i>dashboard</i>	Laporan pendapatan dan pengeluaran terunduh dalam format excel	Laporan pendapatan dan pengeluaran berhasil dicetak	
5	Menambahkan data pendapatan	Data tersimpan kedalam daftar pendapatan	Data pendapatan berhasil ditambahkan	
6	Menambahkan data pengeluaran	Data tersimpan kedalam daftar pengeluaran	Data pengeluaran berhasil ditambahkan	
7	Menambahkan daftar karyawan	Data tersimpan kedalam daftar karyawan	Data karyawan berhasil ditambahkan	
8	Menambahkan data hutang	Data tersimpan kedalam daftar hutang	Data hutang berhasil ditambahkan	
9	Mengedit <i>profile</i>	Data <i>profile</i> dapat dirubah pada halaman edit <i>profile</i>	Data <i>profile</i> berhasil dirubah	
10	<i>Donload</i> laporan	Mengunduh laporan dalam format excel	Laporan berhasil diunduh	