

**APLIKASI *POINT OF SALE* BERBASIS WEB
(STUDI KASUS: TOKO MULIA)**

LAPORAN KERJA PRAKTIK

Disusun oleh:

William James Olii

18013029



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO
2023**

**APLIKASI *POINT OF SALE* BERBASIS WEB
(STUDI KASUS: TOKO MULIA)**

KERJA PRAKTIK

Ditulis untuk Memenuhi Persyaratan Mata Kuliah Kerja Praktik
(INF2217401)

Disusun oleh:

William James Ollii

18013029



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO
2023**

LEMBAR PENGESAHAN

LAPORAN KERJA PRAKTIK

Judul:

APLIKASI *POINT OF SALE* BERBASIS WEB

Telah disetujui dan disahkan pada tanggal:

28 Juli 2023

Oleh:

Toko Mulia



Yan Tumiwa

LEMBAR PERNYATAAN

Yang bertanda tangan di bawah ini:

Nama : William James Oliy
NIM : 18013029
Tempat/Tanggal Lahir : Manado, 24 Oktober 2000
Fakultas/Jurusan : Teknik/Teknik Informatika

Menyatakan dengan ini laporan kerja praktik yang berjudul “*APLIKASI POINT OF SALE BERBASIS WEB*” telah disusun oleh saya sendiri dan tidak ditulis oleh pihak lain. Tetapi ada beberapa hal kutipan milik orang lain, telah ditulis sumbernya.

Dengan ini saya membuat pernyataan, apabila didapati kalimat yang tidak sesuai dengan seharusnya, maka saya bersedia untuk menjalani sanksi sebagaimana yang telah ditentukan, yaitu pembatalan Laporan Kerja Praktik.

Manado, Juli 2023
Yang menyatakan,



William James Oliy

Menyetujui,

Dosen Pembimbing I

Dosen Pembimbing II

Steven Pandelaki, S.T., M.Sc.

Lanny Sitanayah, S.T., M.Sc., Ph.D.

Ketua Program Studi

Mengetahui,

Dekan Fakultas Teknik

Vivie Deyby Kumenap, S.T., M.Cs.



Ronald Albert Rachmadi, S.T., M.T.



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO**

FORM KP - 003

FORMULIR DATA UMUM PERUSAHAAN

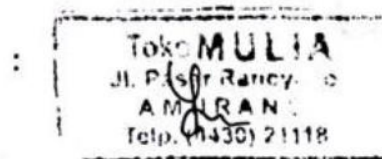
NAMA MAHASISWA : William James Oliy
NIM : 18013029
NAMA PERUSAHAAN : Toko Mulia
ALAMAT PERUSAHAAN : Jln. Pasar Ranoyapo Amurang

DIDIRIKAN TAHUN : 1985
IJIN USAHA : 44/18.02/PDN/AS
BIDANG BISNIS : Toko bangunan
JUMLAH KARWAYAN : 5
PEMILIK : Yan Tumiwa
DEWAN DIREKTUR : -

WAKIL PERUSAHAAN

Tanggal : -
Nama : -
Jabatan : -

(Tanda Tangan dan
Cap Perusahaan)





**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO**

FORM KP - 004

FORMULIR KEMAJUAN KERJA PRAKTIK





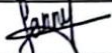
A. UMUM

Nama Mahasiswa : William James Ollie
NIM Mahasiswa : 18013029
Program Studi : Teknik Informatika
Dosen Pembimbing Akademik : Steven Pandelaki, S.T., M.Sc.
Topik/Rencana Bidang : Aplikasi *Point of Sale* Berbasis Web

Pembimbing 1 : Steven Pandelaki, S.T., M.Sc.
Pembimbing 2 : Lanny Sitanayah, S.T., M.Sc., Ph.D.
Terhitung Mulai : 28 Januari 2023
Target Selesai : Juli 2023

B. KEGIATAN PELAKSANAAN KERJA PRAKTIK

No	Tanggal	Jenis Kegiatan	Paraf Pembimbing
1	3 April 2023	Konsultasi mengenai aplikasi yang dibangun (DP1).	
2	4 April 2023	Konsultasi proposal (DP2).	
3	10 April 2023	Konsultasi revisi proposal (DP1).	
4	10 April 2023	Konsultasi mengenai proposal dan aplikasi (DP2).	
5	03 Mei 2023	Konsultasi mengenai aplikasi (DP1).	
6	20 Mei 2023	Konsultasi mengenai laporan (DP1).	
7	3 Juli 2023	Konsultasi Aplikasi (DP1).	
8	20 Juli 2023	Konsultasi bab 1, 2, dan 3 (DP2).	
9	22 Juli 2023	Revisi bab 1 dan 2 (DP2).	
10	24 Juli 2023	Konsultasi mengenai aplikasi (DP1).	
11	24 Juli 2023	Konsultasi bab 4 (DP2).	

12	24 Juli 2023	Konsultasi Bab 4(DP1).	
13	25 Juli 2023	Konsultasi aplikasi (DP2).	
14	26 Juli 2023	Konsultasi Aplikasi (DP1).	
15	26 Juli 2023	Revisi Bab 4 (DP2).	
16	27 Juli 2023	Revisi Seluruh Laporan (DP2).	

Manado, 28 Juli 2023

Dosen Pembimbing KP



Steven Pandelaki, S.T., M.Sc.



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO**

FORM KP - 005

FORMULIR PENILAIAN PELAKSANAAN KERJA PRAKTIK

Mohon diisi dan dicek seperlunya,

NAMA MAHASISWA : William James Olli
NIM : 18013029
NAMA PERUSAHAAN : Toko Mulia
ALAMAT PERUSAHAAN : Jln. Pasar Ranoyapo Amurang
TGL KERJA PRAKTIK : 28 Januari 2023
TOPIK YANG DIBAHAS : Aplikasi *Point of Sale* Berbasis Web

Nilai = _____
Sikap 50 60 70 80 90 100

Nilai = _____
50 60 70 80 90 100

Prestasi = _____
50 60 70 80 90 100

NILAI RATA-RATA : 96,6
TANGGAL : 28 Maret 2023
NAMA PENILAI : Yan Tumiwa
JABATAN : Pemilik Toko

(Tanda Tangan dan
Cap Perusahaan)



KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa, yang telah menganugerahkan banyak hikmat sehingga penulis dapat menyusun laporan Kerja Praktik ini dengan baik. Laporan ini berisi tentang uraian hasil Kerja Praktik mengenai “*APLIKASI POINT OF SALE BERBASIS WEB*”. Oleh karena itu, melalui tulisan ini, penulis ingin berterima kasih kepada pihak-pihak yang sudah senantiasa membantu penulis. mereka adalah:

1. Prof. Dr. Johanis Ohoitumur selaku Rektor Universitas Katolik De La Salle Manado.
2. Bapak Ronald Albert Rachmadi, S.T., M.T. selaku Dekan Fakultas Teknik.
3. Ibu Vivie Deyby Kumenap, S.T., M.Cs. selaku Ketua Program Studi Teknik Informatika.
4. Bapak Steven Pandelaki, S.T., M.Sc. selaku Dosen Pembimbing Akademik sekaligus Dosen Pembimbing I yang telah membimbing penulis selama masa perkuliahan dan Kerja Praktik.
5. Ibu Lanny Sitanayah, S.T., M.Sc., Ph.D. selaku Dosen Pembimbing II yang telah membantu penulis selama Kerja Praktik.
6. Mama, papa, dan adik-adik yang selalu mendukung, memberi semangat, dan mendoakan.
7. Bapak Yan Tumiwa selaku pemilik Toko Mulia yang telah memberikan kesempatan kepada penulis untuk melakukan Kerja Praktik.
8. Felicia, Frendy, Dimas, Octavia, Luis, Matthew, Naldy, dan Jose yang telah menyemangati dan memberikan bantuan selama proses pembuatan laporan Kerja Praktik.

Banyak lagi pihak lainnya yang tidak dapat penulis tuliskan satu-persatu, namun telah membantu penulis. Akhir kata, bisa terdapat kesalahan dalam pengerjaan Kerja Praktik ini. Penulis mohon maaf sebesar-besarnya atas kesalahan yang telah diperbuat.

Manado, Juli 2023

Penulis

DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN	ii
LEMBAR PERNYATAAN.....	iii
FORMULIR DATA UMUM PERUSAHAAN (FORM KP-003).....	iv
FORMULIR KEMAJUAN KERJA PRAKTIK (FORM KP-004).....	v
FORMULIR PENILAIAN PELAKSANAAN KERJA PRAKTIK (FORM KP-005).....	vii
KATA PENGANTAR	viii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN.....	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Kerja Praktik	2
1.4 Manfaat Kerja Praktik	2
1.5 Batasan Masalah.....	2
1.6 Sistematika Penulisan.....	3
BAB II DATA UMUM PERUSAHAAN	4
2.1 Sejarah Singkat Perusahaan.....	4
2.2 Ruang Lingkup Perusahaan.....	4
2.1.1 Lingkup Pekerjaan Perusahaan	4
2.1.2 Visi Perusahaan	4
2.1.3 Misi Perusahaan	5
2.1.4 Logo Perusahaan	5
2.1.5 Struktur Organisasi	5
2.1.6 Tugas Pokok dan Fungsi Bagian Umum dan Kepegawaian	6
2.3 Lingkup Pekerjaan Yang Dilakukan	6
BAB III LANDASAN TEORI.....	7
3.1 Teori Pendukung	7
3.1.1 Aplikasi Web.....	7
3.1.2 <i>Point of Sales</i>	7
3.1.3 <i>Cascading Style Sheet (CSS)</i>	7
3.1.4 <i>Hypertext Markup Language (HTML)</i>	8
3.1.5 <i>Framework Laravel</i>	8
3.1.6 <i>Hypertext Preprocessor (PHP)</i>	9
3.1.7 <i>Database</i>	9
3.1.8 <i>JavaScript</i>	10
3.1.9 <i>Unified Modeling Language (UML)</i>	11
3.2 Metodologi Pengembangan Aplikasi	14
3.3 Prosedur Pengumpulan Data	14

BAB IV PEMBAHASAN.....	16
4.1 Analisis Persyaratan	16
4.1.1 Pengumpulan Data	16
4.1.2 Komunikasi dan Perencanaan Proyek.....	17
4.1.3 Mengidentifikasi Masalah dan Solusi	18
4.1.4 Daftar Spesifikasi Persyaratan	18
4.1.5 Identifikasi Sumber Daya Perangkat Lunak	20
4.2 Desain.....	20
4.2.1 Use Case Diagram.....	21
4.2.2 <i>Use Case Table</i>	21
4.2.3 <i>Class Diagram</i>	27
4.2.4 <i>Activity Diagram</i>	28
4.2.5 <i>Storyboard</i>	30
4.3 Pembuatan Kode Program.....	33
4.3.1 Implementasi Basis Data.....	33
4.3.2 Implementasi Antarmuka	37
4.3.3 Melakukan Pemrograman	43
4.4 Pengujian	50
 BAB V KESIMPULAN DAN SARAN.....	 69
5.1 Kesimpulan.....	69
5.2 Saran.....	69
 DAFTAR PUSTAKA	 70
LAMPIRAN	

DAFTAR TABEL

Tabel 3.1	Komponen <i>Use Case Diagram</i>	11
Tabel 3.2	Komponen <i>Class Diagram</i>	12
Tabel 3.3	Komponen <i>Activity Diagram</i>	13
Tabel 4.1	Komunikasi dan Perencanaan Proyek.....	17
Tabel 4.2	Mengidentifikasi Masalah dan Solusi	18
Tabel 4.3	Sumber Daya Perangkat Keras Dan Perangkat Lunak.....	20
Tabel 4.4	<i>Use Case Table</i> Masuk Aplikasi.....	21
Tabel 4.5	<i>Use Case Table</i> Melakukan Lupa Kata Sandi	22
Tabel 4.6	<i>Use Case Table</i> Mengelola Data Barang	22
Tabel 4.7	<i>Use Case Table</i> Mengelola Data <i>Sales</i>	23
Tabel 4.8	<i>Use Case Table</i> Membuat Transaksi Barang.....	24
Tabel 4.9	<i>Use Case Table</i> Memilih Barang Dibeli.....	24
Tabel 4.10	<i>Use Case Table</i> Membayar Barang Dibeli	25
Tabel 4.11	<i>Use Case Table</i> Mencetak Struk.....	25
Tabel 4.12	<i>Use Case Table</i> Melihat Riwayat Terjual	26
Tabel 4.13	Pelaksanaan Pengujian.....	51

DAFTAR GAMBAR

Gambar 2.1	Logo Perusahaan	5
Gambar 2.2	Struktur Organisasi Toko Mulia.....	5
Gambar 4.1	<i>Use Case Diagram</i>	21
Gambar 4.2	<i>Class Diagram</i>	27
Gambar 4.3	<i>Activity Diagram</i>	28
Gambar 4.4	<i>Storyboard</i> Halaman Barang	30
Gambar 4.5	<i>Storyboard</i> Halaman Sales	31
Gambar 4.6	<i>Storyboard</i> Halaman Transaksi	32
Gambar 4.7	<i>Storyboard</i> Halaman Laporan	33
Gambar 4.8	Implementasi Keseluruhan Tabel.....	34
Gambar 4.9	Tabel Barang	34
Gambar 4.10	Tabel Keranjang	35
Gambar 4.11	Tabel Sales	35
Gambar 4.12	Tabel Transaksi	36
Gambar 4.13	Tabel Transaksi Detail	36
Gambar 4.14	Tabel <i>User</i>	37
Gambar 4.15	Implementasi Halaman masuk	37
Gambar 4.16	Implementasi Lupa Kata Sandi	38
Gambar 4.17	Implementasi Halaman Barang	38
Gambar 4.18	Implementasi Halaman Sales	39
Gambar 4.19	Implementasi Halaman Transaksi	39
Gambar 4.20	Implementasi Halaman Pilih barang	40
Gambar 4.21	Implementasi Halaman Cetak Nota.....	40
Gambar 4.22	Implementasi Halaman Nota.....	41
Gambar 4.23	Implementasi Halaman Laporan Pengujian	41
Gambar 4.24	Implementasi Halaman Ganti Pertanyaan	42
Gambar 4.25	Pengujian pada Google chrome.....	51
Gambar 4.26	Pengujian di Browser Edge	52
Gambar 4.27	Implementasi Pada Browser firefox.....	52
Gambar 4.28	Pengujian Halaman Login	53
Gambar 4.29	Pengujian Menampilkan Halaman Barang setelah Log In.....	53
Gambar 4.30	Pengujian Menampilkan Halaman Data Barang	54
Gambar 4.31	Pengujian Halaman Isi Data Barang	54
Gambar 4.32	Pengujian Data Barang Berhasil Diisi.....	55
Gambar 4.33	Pengujian Data Barang Sebelum Diubah	55
Gambar 4.34	Pengujian Data Barang Setelah Diubah	56
Gambar 4.35	Pengujian Data Barang Berhasil Diubah.....	56
Gambar 4.36	Pengujian Data Setelah Dihapus	57
Gambar 4.37	Pengujian Halaman Data Sales	57
Gambar 4.38	Pengujian Halaman Mengisi Data <i>Sales</i>	58
Gambar 4.39	Pengujian Data <i>Sales</i> Berhasil Ditambah.....	58
Gambar 4.40	Pengujian Halaman Data <i>Sales</i> Sebelum Diganti	59
Gambar 4.41	Pengujian Halaman Data <i>Sales</i> Setelah Diubah.....	59
Gambar 4.42	Pengujian Halaman Sales Setelah Ditekan Tombol Ubah	59
Gambar 4.43	Pengujian Halaman Sales Setelah Dihapus Data Sales.....	60

Gambar 4.44 Pengujian Halaman Transaksi Sebelum Memilih Barang.....	60
Gambar 4.45 Pengujian Halaman Transaksi Pilih Barang.....	61
Gambar 4.46 Pengujian Halaman Transaksi Setelah Memilih Barang.....	61
Gambar 4.47 Pengujian Halaman Transaksi Pada Saat Mengisi Tanggal Dan Jumlah Uang	62
Gambar 4.48 Pengujian Halaman Transaksi Setelah Menekan Tombol Bayar	62
Gambar 4.49 Pengujian Halaman Transaksi Bagian Riwayat Transaksi.....	63
Gambar 4.50 Halaman Cetak Struk	63
Gambar 4.51 Pengujian Halaman Laporan	64
Gambar 4.52 Pengujian Pencarian Keuntungan Berdasarkan Harian.....	64
Gambar 4.53 Pengujian Pencarian Keuntungan Berdasarkan Bulanan	65
Gambar 4.54 Pengujian Pencarian Keuntungan Berdasarkan Tahunan.....	66
Gambar 4.55 Pengujian Halaman Ganti Pertanyaan.....	66
Gambar 4.56 Pengujian Berhasil Ganti Pertanyaan.....	67
Gambar 4.57 Pengujian Halaman Ganti Akun Kasir	67
Gambar 4.58 Pengujian Berhasil Ganti username/kata sandi	67

DAFTAR LAMPIRAN

LAMPIRAN A TRANSKRIP WAWANCARA.....	A-1
LAMPIRAN B <i>USER ACCEPTANCE TESTING</i>	B-1

BAB I

PENDAHULUAN

1.1 Latar Belakang

Saat ini dunia telah memasuki era teknologi yang seiring berjalannya waktu berkembang dengan pesat. Teknologi saat ini telah diaplikasikan dalam kehidupan sehari-hari untuk mempermudah suatu hal sehingga tidak dapat dipungkiri lagi bahwa teknologi saat ini telah menjadi sesuatu yang sangat penting. Salah satu contoh pihak yang dapat menggunakan teknologi untuk mempermudah pekerjaan mereka adalah para pemilik toko, salah satunya adalah pemilik toko bahan bangunan.

Toko Mulia adalah toko bahan bangunan yang terletak di Kecamatan Amurang yang merupakan ibukota dari Kabupaten Minahasa Selatan, Provinsi Sulawesi Utara. Toko Mulia merupakan toko bahan bangunan yang telah lama berdiri dan telah dikenal oleh banyak orang. Toko Mulia sebagai toko bahan bangunan menjual barang material bangunan yang memiliki banyak jenis dan ukuran. Pembeli juga sering datang dalam jumlah yang banyak dan membeli banyak jenis barang sekaligus. Hal ini membuat pemilik dan karyawan toko kesusahan dalam memberikan pelayanan yang baik. Pemilik dan karyawan mengalami kesulitan dalam menghafal harga dari segala barang yang ada. Selain itu, informasi mengenai stok barang yang dimiliki toko harus dicek setiap dicari pembeli. Hal ini mengakibatkan kerugian bagi toko dalam segi waktu. Hal lain yang mengakibatkan proses penjualan dalam toko menjadi lebih lama adalah pembuatan nota yang masih menggunakan metode konvensional yaitu ditulis di kertas, yang selain memakan waktu lama juga rentan akan terjadinya kesalahan karena tulisan maupun coretan. Hal lain yang juga menjadi masalah adalah tidak adanya rekapan transaksi sehingga pemilik tidak memiliki informasi yang jelas mengenai keuntungan yang didapat [1].

Berdasarkan masalah tersebut, maka Toko Mulia membutuhkan suatu aplikasi *point of sale* agar dapat mempermudah pemilik toko dalam proses transaksi yang terjadi baik untuk menghemat waktu yang terbuang dalam pengecekan *stock*

barang dan pembuatan nota, juga dapat memberikan rekapan penjualan yang jelas kepada pemilik.

1.2 Rumusan Masalah

Bagaimana cara membuat suatu aplikasi *point of sale* yang dapat membantu pemilik Toko Mulia dalam menjalankan usahanya?

1.3 Tujuan Kerja Praktik

Tujuan dari Kerja Praktik ini adalah untuk mendesain dan membangun suatu aplikasi *point of sale* yang dapat mempermudah proses bisnis dalam Toko Mulia.

1.4 Manfaat Kerja Praktik

Manfaat dari kerja praktik ini adalah sebagai berikut:

- A. Toko Mulia
 - 1. Aplikasi diharapkan dapat mempermudah pemilik dan karyawan Toko Mulia dalam memeriksa harga dan persediaan barang.
 - 2. Aplikasi dapat meminimalisir waktu yang terbuang saat pembuatan nota.
 - 3. Aplikasi dapat mempermudah proses pengawasan yang dilakukan oleh pemilik toko untuk mengetahui segala transaksi yang terjadi dengan laporan penjualan.
- B. Mahasiswa
 - 1. Sebagai penambahan pengalaman kerja dan sebagai latihan agar lebih siap untuk masa kerja nanti.
 - 2. Sebagai media pengalaman dalam penerapan ilmu yang telah dipelajari selama masa perkuliahan.

1.5 Batasan Masalah

Batasan masalah dalam Kerja Praktik ini adalah sebagai berikut:

- 1. Aplikasi yang dibuat tidak memiliki laporan laba rugi.
- 2. Aplikasi tidak memiliki fitur pengembalian barang.

3. Aplikasi yang dibuat merupakan aplikasi web yang dijalankan melalui *localhost*.
4. Aplikasi akan memiliki 2 level pengguna yang berupa admin dan kasir
5. Aplikasi tidak memiliki fitur riwayat barang masuk.

1.6 Sistematika Penulisan

Sistematika penulisan laporan Kerja Praktik yang akan digunakan adalah:

1. Bab I Pendahuluan; memuat latar belakang, rumusan masalah, tujuan Kerja Praktik, manfaat, batasan masalah, dan sistematika penulisan.
2. Bab II Data Umum Perusahaan; memuat sejarah singkat perusahaan, lingkup kerja perusahaan, dan lingkup pekerjaan yang dilakukan.
3. Bab III Landasan Teori; memuat teori pendukung, metodologi pengembangan perangkat lunak, dan prosedur pengumpulan data.
4. Bab IV Pembahasan; memuat pengumpulan dan pengolahan data, analisis dan pemecahan masalah yang mengikuti metodologi pengembangan perangkat lunak.
5. Bab V Kesimpulan dan Saran; memuat kesimpulan dan saran.

BAB II

DATA UMUM PERUSAHAAN

2.1 Sejarah Singkat Perusahaan

Toko Mulia adalah suatu toko bahan bangunan yang terletak di Kecamatan Ranoyapo, Kabupaten Minahasa Selatan, Sulawesi Utara. Toko Mulia merupakan toko bangunan yang telah ada sejak lama dan telah memiliki Surat Izin Usaha yang resmi [2].

Toko Mulia merupakan bisnis keluarga yang dimiliki oleh keluarga Tumiwa – Ratuwongo. Toko Mulia seringkali dijalankan oleh para anggota keluarga yaitu istri dan anak dari pemilik dan memiliki tiga orang karyawan yang di mana dua dari karyawan tersebut bertugas membantu di toko dan seorang karyawan untuk bagian gudang. Toko Mulia terus berusaha untuk berkembang dengan menambah jenis barang yang dijual agar dapat menjadi toko yang lengkap agar dapat memberikan pelayanan yang terbaik bagi para pembeli [2].

2.2 Ruang Lingkup Perusahaan

Pada bagian ini akan membahas aktivitas yang ada pada perusahaan, visi dan misi, logo perusahaan, struktur organisasi, serta tugas pokok dan fungsi dari perusahaan.

2.1.1 Lingkup Pekerjaan Perusahaan

Lingkup pekerjaan dari Toko Mulia adalah menjual berbagai bahan bangunan seperti perlengkapan cat, besi, semen, mesin, perlengkapan bor, dan berbagai bahan bangunan lainnya.

2.1.2 Visi Perusahaan

Toko Mulia tentunya memiliki visi, yaitu “Menjadi toko bahan bangunan terlengkap dan terpercaya sehingga dapat menjadi pilihan utama bagi para konsumen dalam mencari keperluan bahan bangunan” [2].

Penjelasan mengenai Misi Perusahaan berada dihalaman selanjutnya.

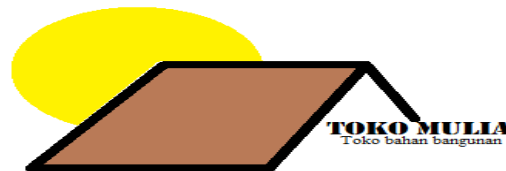
2.1.3 Misi Perusahaan

Toko Mulia memiliki misi berupa [2]:

1. Memberikan pelayanan berkualitas tinggi kepada para pembeli.
2. Memberikan kepuasan dan kenyamanan sehingga pembeli memiliki pengalaman belanja yang menyenangkan.
3. Menjadi tempat berbelanja bahan bangunan yang lengkap dengan harga terjangkau.

2.1.4 Logo Perusahaan

Berikut merupakan logo yang digunakan oleh Toko Mulia beserta arti dari komponen di dalamnya [2]:



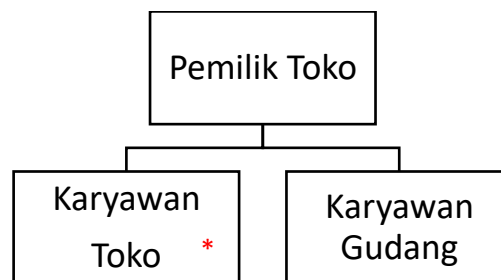
Gambar 2.1 Logo Perusahaan [2]

Arti dari bagian-bagian yang ada dalam logo perusahaan, yaitu [2]:

1. Tulisan yang memberitahu mengenai nama toko yaitu Toko Mulia beserta dengan penjelasan tentang apa yang dijual oleh Toko Mulia.
2. Warna coklat pada logo tersebut yang memberikan kesan kenyamanan dan kehangatan seperti pelayanan yang diberikan dalam Toko Mulia.
3. Matahari di bagian belakang logo memiliki arti masa depan yang cerah seperti harapan pemilik akan Toko Mulia.

2.1.5 Struktur Organisasi

Berikut merupakan struktur organisasi dari Toko Mulia, yaitu [2]:



Keterangan: (*) merupakan penempatan penulis saat Kerja Praktik.

Gambar 2.2 Struktur Organisasi Toko Mulia [2]

2.1.6 Tugas Pokok dan Fungsi Bagian Umum dan Kepegawaian

Berikut merupakan tugas dari masing-masing bagian dalam Toko Mulia [2].

A. Pemilik Toko

1. Mengurus segala hal mengenai kelengkapan *supply* produk dengan cara pembelian kepada *sales*.
2. Melayani proses pembayaran yang terjadi dan menjadi satu-satunya posisi dengan akses kasir.
3. Memeriksa barang masuk dan mengontrol persediaan barang yang ada.
4. Mengatur jumlah dan memberikan gaji karyawan.

B. Karyawan Toko

1. Membantu pembeli mencari barang yang diperlukan.
2. Memberikan kepada pembeli barang yang dicari.
3. Membantu memeriksa barang yang masuk.
4. Membersihkan toko.
5. Memberi saran pada pembeli mengenai barang yang akan dibeli.

C. Karwayan Gudang

1. Melakukan pemeriksaan mengenai stok barang yang ada di gudang.
2. Mengatur letak dan posisi barang yang ada di gudang.
3. Melaporkan keadaan gudang seperti habis pulsa listrik atau genteng bocor.
4. Memeriksa barang yang masuk dan mengecek jika barang yang masuk tersebut dalam keadaan baik.

2.3 Lingkup Pekerjaan Yang Dilakukan

Pada Toko Mulia penulis ditempatkan untuk membantu karyawan toko dengan lingkup pekerjaan berupa:

1. Mengorganisir dan merapikan letak barang-barang yang ada di toko.
2. Membantu pembeli dalam mencari dan menyiapkan barang yang diinginkan.
3. Membantu memeriksa barang yang masuk sesuai nota yang ada.

BAB III

LANDASAN TEORI

3.1 Teori Pendukung

Bagian ini berisi tentang teori pendukung yang berkaitan dengan pembuatan Aplikasi *Point of Sales*. Informasi-informasi yang ada berasal dari berbagai sumber yang berupa jurnal ilmiah maupun buku yang ada di Internet.

3.1.1 Aplikasi Web

Aplikasi web merupakan suatu jenis aplikasi yang dapat diakses kapanpun dan dimanapun selama pengguna memiliki jaringan Internet. Aplikasi web sendiri dapat diakses melalui *browser* seperti Mozilla Firefox, Google Chrome, Microsoft Edge dan berbagai aplikasi *browser* lainnya [3].

Web *browser* sendiri adalah sebuah aplikasi yang dapat digunakan agar bisa menjelajahi situs-situs yang ada di dunia maya atau yang dapat juga disebut *website*. Tujuan dari *browser-browser* yang ada adalah untuk membaca dokumen HTML dan kemudian menampilkannya ke dalam *User Interface* yang dapat dibaca dan dimengerti oleh pengguna [4].

3.1.2 Point of Sales

Aplikasi point of sales adalah suatu aplikasi yang dapat membantu pemilik toko untuk dapat memeriksa persediaan barang dengan lebih baik dan dapat melakukan proses transaksi dengan fitur-fitur yang ada di dalam aplikasi tersebut [5].

Ada berbagai keuntungan yang akan didapat jika menggunakan *point of sales* seperti meningkatnya kualitas dari pelayanan yang ada, meningkatkan citra usaha karena konsumen dapat melihat bahwa perusahaan dikelola secara baik dan profesional. Selain itu, perusahaan juga dapat memiliki pengelolaan data dan penyediaan informasi yang lebih baik dari perusahaan lain yang belum memanfaatkan aplikasi *point of sales* [5] .

Penjelasan mengenai CSS berada dihalaman berikutnya.

3.1.3 *Cascading Style Sheet (CSS)*

CSS adalah suatu bahasa *style sheet* yang memiliki kegunaan untuk membuat dan mengubah tampilan dokumen. CSS umumnya digunakan untuk mengatur tampilan dari halaman web yang dibuat dengan bahasa HTML dan XHTML. Untuk lebih jelasnya, CSS memiliki fungsi untuk mengatur ukuran gambar, warna teks dan tabel, ukuran dan warna border, dan parameter lainnya [6].

CSS juga bisa dikatakan sebagai penjelasan dari elemen-elemen HTML yang ditampilkan agar terlihat lebih baik. CSS memiliki 3 cara yang dapat dilakukan agar dapat ditambahkan ke elemen HTML yaitu *inline* yang menggunakan atribut *style* dari elemen HTML, *internal* yang menggunakan elemen `<style>` pada area `<head>`, dan *external* menggunakan file CSS *external* [4].

3.1.4 *Hypertext Markup Language (HTML)*

HTML adalah suatu bahasa pemrograman terstruktur yang dirancang dengan tujuan membuat halaman web yang dapat diakses atau dilihat dengan *web browser*. HTML dibuat pada tahun 1989 oleh Tim Berners Lee yang kemudian dikembangkan oleh *World Wide Web Consortium (W3C)* [4].

HTML memiliki fungsi sebagai tulisan yang dapat memuat instruksi yang dapat memiliki berbagai efek yang berbeda-beda pada kode tersebut. HTML memungkinkan seseorang agar bisa menyunting format atau tampilan yang akan dikirimkan melalui media daring [4].

3.1.5 *Framework Laravel*

Framework atau dapat juga disebut kerangka kerja adalah sekumpulan *script* yang dapat mempermudah *developer* dalam menyelesaikan berbagai masalah-masalah yang ada dalam pemrograman, seperti koneksi ke *database*, pemanggilan variabel, dan sebagainya, sehingga aplikasi dapat terbangun dengan lebih mudah dan cepat [7].

Framework dapat dengan diambil dan digunakan pada saat pembuatan aplikasi, sehingga keberadaannya sangat membantu karena para *programmer* tidak perlu lagi untuk melakukan segala sesuatu dari awal karena telah disediakan oleh *framework* [8].

Laravel adalah satu dari sekian banyak jenis *framework* PHP yang ada dan dapat digunakan secara gratis. Laravel pada mulanya dikembangkan oleh seorang *programmer* yang berasal dari Amerika Serikat bernama Taylor Otwell di tahun 2011. Laravel kemudian naik menjadi salah satu *framework* yang paling disukai oleh seluruh *programmer* di dunia melebihi *framework* pendahulunya [8].

Laravel memiliki banyak keunggulan yang membuatnya dapat menjadi salah satu *framework* terbaik. Beberapa keunggulannya berupa lebih banyaknya fitur yang tidak dimiliki *framework* lain, merupakan suatu *framework* yang ekspresif sehingga mudah dimengerti, dan didukung oleh *composer* sehingga *library-library* Laravel dapat ditemukan dengan mudah oleh *composer* [8].

3.1.6 Hypertext Preprocessor (PHP)

PHP merupakan suatu bahasa pemrograman yang bersifat universal yang memiliki kegunaan dalam pembuatan dan pengembangan suatu situs web dan dapat digunakan secara bersamaan dengan HTML dan juga mediawiki adalah contoh dari aplikasi program PHP [9].

Fungsi utama dari PHP adalah untuk mengevaluasi dan juga mengumpulkan hasil survei atau bentuk apa saja ke dalam *server database* dan kemudian menciptakan efek beruntun yang berupa tindakan komunikasi dari skrip lain dan *database* yang kemudian mengumpulkan dan mengelompokkan informasi yang akan ditampilkan saat dibutuhkan [9].

3.1.7 Database

Database merupakan suatu aplikasi terpisah yang menjadi tempat penyimpanan data-data yang dikoleksi. Data terkait yang telah dikoleksi tersimpan dalam kumpulan tabel di mana data-data tersebut dapat diakses dan dikelola dalam *database* tersebut [10].

Perangkat lunak yang dapat digunakan untuk mengelola basis data disebut sistem manajemen basis data atau dapat juga disingkat DBMS. DBMS inilah yang kemudian digunakan untuk melakukan pengaturan dan mengelola koleksi data berjumlah besar yang ada dalam *database* tersebut dengan mudah [11].

Penjelasan mengenai XAMPP berada dihalaman berikutnya.

3.1.7.1 XAMPP

XAMPP merupakan suatu perangkat yang menjadi penghubung dari tiga aplikasi yaitu Apache, MySQL, dan phpMyAdmin. XAMPP sangat mempermudah pekerjaan yang harus dilakukan karena dapat menginstalasi dan mengkonfigurasi ketiga aplikasi tersebut dengan otomatis [10].

XAMPP merupakan aplikasi *open source* yang dapat digunakan secara gratis, sehingga menjadi salah satu aplikasi pengelolaan *server* paling populer dan sering digunakan. Selain gratis, XAMPP juga bersifat *cross platform* yang artinya mendukung berbagai macam *platform* baik Windows, MacOS, maupun Linux [11].

3.1.7.2 MariaDB

MariaDB merupakan versi perkembangan dari MySQL oleh Oracle pada tahun 2010. MariaDB merupakan manajemen basis data relasional (RDBMS) yang digunakan untuk penyimpanan data aplikasi web. MariaDB memiliki fitur yang lebih baik daripada MySQL dan memiliki kompatibilitas yang baik dengan MySQL sehingga dalam pengelolaan data memiliki kemiripan dengan MySQL [12].

Beberapa contoh kelebihan yang dimiliki oleh MariaDB adalah portabilitas yang baik karena mampu dipasang dan berjalan di semua sistem operasi. MariaDB adalah *open source*, sehingga dapat digunakan secara gratis, dapat digunakan oleh banyak *user* dalam waktu bersamaan, memiliki banyak pilihan data, dan keamanan yang dibuat dengan detail [12].

3.1.8 JavaScript

JavaScript merupakan suatu bahasa yang digunakan dalam pembuatan program agar dokumen HTML yang ditampilkan selain indah dapat juga dilakukan proses interaksi. JavaScript dapat menyediakan lebih dari satu pilihan fungsionalitas ke dalam halaman web yang nantinya akan ditampilkan pada halaman antarmuka web [13].

JavaScript merupakan bahasa *script* yang ringan dan mudah untuk digunakan. Selain itu, JavaScript juga tidak memerlukan adanya lisensi agar dapat menggunakannya. Jika *browser* yang digunakan mendukung JavaScript, maka *JavaScript* dapat langsung dijalankan [13].

3.1.9 Unified Modeling Language (UML)

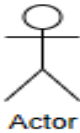
UML merupakan sebuah sebutan untuk sebuah teknik pengembangan sistem yang memakai bahasa grafis sebagai sarana untuk dokumentasi dan membuat spesifikasi pada sistem [14]. UML memiliki liputan berupa teknik notasi grafik yang dapat digunakan untuk membuat suatu model yang abstrak dari suatu sistem tertentu. Singkatnya UML adalah suatu metode terbuka yang dapat digunakan untuk memberikan spesifikasi, memvisualisasikan, membuat, dan mendokumentasikan perangkat lunak yang sedang dibuat. UML menyediakan suatu cara standar untuk membuat *blueprint* dari sistem yang sedang dibuat. Selain itu, UML juga memperhatikan kalimat bahasa pemrograman, skema dari *database*, dan komponen perangkat lunak yang dapat digunakan kembali. UML memiliki tujuan untuk menjadi suatu bahasa pemodelan standar yang dapat membuat model *concurrent* dan sistem terdistribusi [15].



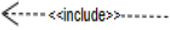
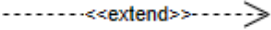
Dalam pembuatan Aplikasi *Point Of Sale* Berbasis Web dengan studi kasus Toko Mulia, jenis diagram yang akan digunakan kaka pemodelan yang akan digunakan adalah *use case diagram*, *class diagram*, dan *activity diagram*.

3.1.9.1 Use Case Diagram

Diagram use case adalah diagram yang mendeskripsikan fungsi yang diharapkan dari sistem yang akan dikembangkan. Diagram ini meliputi keseluruhan fungsi yang akan dijalankan saat menggunakan sistem yang akan dibuat. *Use case* akan memberikan suatu manfaat nyata untuk satu atau lebih aktor yang berkomunikasi dengan *use case* yang dibuat [16].

Tabel 3.1 Komponen *Use Case Diagram* [16].

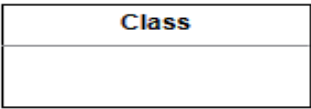

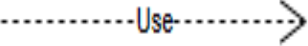
Nama	Gambar	Keterangan
<i>Actor</i>		Aktor adalah representasi dari peran yang dimiliki oleh pengguna sistem.
<i>Use Case</i>		<i>Use case</i> adalah interaksi antara aktor dan sistem.


Nama	Gambar	Keterangan
		
<i>Association</i>		Penghubung antara aktor dan <i>use case</i> yang menunjukkan aktor mana yang berinteraksi dengan fitur yang ada dalam <i>use case</i> tersebut.
<i>Include</i>		Menjadi penanda bahwa <i>use case</i> tersebut adalah termasuk bagian dari <i>use case</i> yang ditandai.
<i>Extend</i>		Menjadi penanda bahwa <i>use case</i> tersebut memiliki hubungan dengan <i>case</i> yang ditandai.

3.1.9.2 Class Diagram

Class diagram adalah diagram yang digunakan untuk memodelkan struktur statis dari suatu sistem, sehingga dapat mendeskripsikan mengenai elemen-elemen yang ada dalam sistem dan hubungan antara mereka [16].

Tabel 3.2 Komponen *Class Diagram* [16]




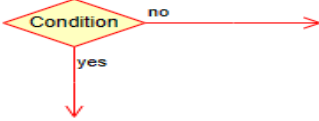

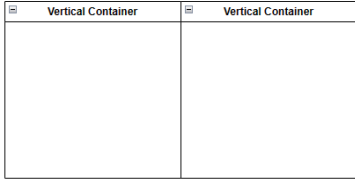
Nama	Gambar	Keterangan
<i>Class</i>		Menunjukkan atribut dan metode.
<i>Association</i>		Untuk menghubungkan satu objek dengan yang lain.
<i>Dependency</i>		Hubungan antar kelas di mana suatu kelas menggunakan kelas lain.

Nama	Gambar	Keterangan
<i>Aggregation</i>		Hubungan antar kelas di mana <i>class</i> tersebut adalah bagian dari <i>class</i> lainya.

3.1.9.3 Activity Diagram

Activity diagram adalah diagram yang lebih fokus kepada pemodelan dari aspek pemrosesan prosedural dari sistem tersebut. *Activity diagram* memberikan spesifikasi *control flow* dan *data flow* yang ada di antara berbagai langkah yang ada atau aksi yang diperlukan untuk mengimplementasikan suatu aktivitas [16].

Tabel 3.3 Komponen *Activity Diagram* [16]

Nama	Gambar	Keterangan
<i>Initial State</i>		Menandai awal dari aliran kerja.
<i>Final State</i>		Menandai akhir dari aliran kerja.
<i>Activity</i>		Merupakan aktivitas yang dilakukan oleh sistem dan biasanya diawali dengan kata kerja.
<i>Decision</i>		Menandai percabangan yang terjadi saat aktivitas yang dapat dipilih lebih dari satu.
<i>Merge</i>		Menyatukan <i>decision path</i> sehingga menjadi satu aktivitas.
<i>swimlane</i>		Untuk memberitahu siapa yang bertanggung jawab atas aktivitas yang terjadi.

3.2 Metodologi Pengembangan Aplikasi

Metodologi yang digunakan dalam pembangunan Aplikasi *Point of Sales* Berbasis Web adalah dengan menggunakan metodologi *Rapid Application Development* (RAD). Metode RAD merupakan suatu pendekatan berorientasi objek terhadap pembuatan suatu sistem yang meliputi suatu metode pengembangan serta perangkat lunak. RAD adalah suatu model proses sekuensial linier untuk pengembangan perangkat lunak yang menekankan siklus pengembangan yang sangat singkat. RAD memungkinkan bagi para *developer* agar dapat membuat suatu sistem fungsional yang utuh dengan waktu yang singkat. Berikut merupakan tahapan-tahapan yang ada dalam RAD [17]:

1 *Requirement Planning*

Tahapan ini adalah tahapan di mana segala hal yang menjadi kebutuhan sistem akan diketahui dengan cara mengidentifikasi kebutuhan informasi dan segala jenis masalah yang akan dihadapi agar dapat menentukan tujuan, batasan dari sistem, kendala, dan alternatif yang dapat diambil untuk pemecahan masalah. Analisis digunakan agar dapat mengetahui perilaku dan aktivitas sistem [17].

2 *Design Workshop*

Tahapan ini adalah tahapan di mana segala solusi alternatif akan dilakukan identifikasi sehingga solusi terbaik dapat dipilih. Selanjutnya adalah membuat suatu desain proses bisnis dan desain pemrograman bagi data-data yang telah ada kemudian dimodelkan dalam arsitektur sistem informasi. Bagian ini biasanya menggunakan *tools* UML [17].

3 *Implementation*

Setelah tahapan *design workshop*, maka langkah selanjutnya adalah implementasi (*coding*) ke dalam suatu bentuk yang dapat dimengerti oleh mesin sehingga dapat tercipta suatu unit program. Tahapan ini merupakan tahap peletakan sistem agar siap untuk digunakan [17].

3.3 Prosedur Pengumpulan Data

Dalam proses pengumpulan data yang terjadi Toko Mulia untuk mendapatkan kriteria aplikasi yang sesuai dengan kebutuhan pengguna, dilakukanlah beberapa metode yang berupa:

1. Observasi

Observasi merupakan salah satu metode yang dilakukan untuk mendapatkan informasi yang diinginkan. Observasi dilakukan untuk mengetahui SOP yang ada dalam toko serta tugas apa saja yang dimiliki oleh pegawai dan pemilik toko.

2. Wawancara

Wawancara juga merupakan salah satu metode dalam mendapatkan informasi. Wawancara dilakukan untuk mengetahui fitur dan bentuk aplikasi yang diinginkan oleh pemilik toko.

BAB IV

PEMBAHASAN

4.1 Analisis Persyaratan

Tahapan ini adalah tahap di mana dilakukan proses pengumpulan informasi dengan metode yang telah dijelaskan sebelumnya yaitu metode observasi dan wawancara. Data yang telah didapatkan akan digunakan untuk menganalisis batasan, bentuk dan tujuan dari aplikasi yang akan dibuat.

4.1.1 Pengumpulan Data

Pada tahapan ini data dikumpulkan dengan menggunakan metode wawancara kepada Bapak Yan Tumiwa untuk mendapatkan informasi mengenai data-data yang diperlukan untuk membuat aplikasi yang sesuai dengan kebutuhan toko. Berikut merupakan kumpulan pertanyaan dan jawaban yang diberikan oleh pemilik.

A. Pertanyaan Wawancara

Berikut adalah daftar pertanyaan yang diajukan dalam proses pengumpulan data:

1. Seperti apa struktur organisasi yang dimiliki Toko Mulia?
2. Apa saja produk-produk yang dijual di Toko Mulia?
3. Kapan pengecekan persediaan barang terjadi?
4. Apakah pemilik mengingat harga dari barang-barang yang ada?
5. Apakah pemilik mengingat *stock* dari barang-barang yang ada?
6. Apakah karyawan diberi kuasa untuk menangani bagian kasir?
7. Apakah pemilik merasa kesulitan dalam menulis nota?

B. Hasil Wawancara

Toko Mulia selayaknya toko bangunan pada biasanya memiliki produk-produk jualan berupa bahan-bahan bangunan yang beragam dengan berbagai macam jenis dan ukuran. Toko Mulia sendiri memiliki 3 pembagian kerja, yaitu bagian toko, bagian gudang, dan pemilik toko sendiri. Bagian toko bertugas untuk melayani para pembeli yang datang ke toko dan menguasai barang-barang yang ada

di toko baik tempat dan perkiraan *stock*. Bagian gudang bertugas saat pembeli mencari barang yang ditempatkan di gudang dan menjaga proses *stock* barang. Pemilik memiliki tugas sebagai satu-satunya bagian yang memiliki akses akan kasir dan pembuatan nota. Akan tetapi pemilik juga sering melakukan pekerjaan bagian toko jika tidak ada karyawan di toko atau keadaan toko sangat ramai.

Karena jumlah produk yang ada sangat banyak dan beragam, maka pemilik mengalami kesulitan untuk mengingat harga-harga dari produk yang ada. Hal ini diakibatkan karena selain jumlah barang yang banyak dan beragam, harga-harga dari barang-barang tersebut kerap berubah sehingga membuat pemilik kesulitan dalam menghafalnya.

Selain itu, pengecekan *stock* barang memakan waktu sangat lama dikarenakan harus dihitung satu persatu oleh bagian gudang sehingga sangat tidak efisien waktu. Selain itu, pengecekan tidak terjadi setiap hari, sehingga biasanya saat pembeli datang dan membeli dengan jumlah besar maka harus dicek lagi. Hal ini memakan waktu lama. Pembuatan nota juga menjadi salah satu hal yang memakan waktu dikarenakan masih dilakukan dengan cara ditulis tangan yang selain rawan akan kesalahan, para pembeli juga biasa berubah pikiran yang menyebabkan nota harus dicoret atau ditulis kembali dari awal.

4.1.2 Komunikasi dan Perencanaan Proyek

Pada tahapan ini dilakukanlah identifikasi mengenai masalah yang ada dalam Toko Mulia, sehingga dapat menemukan solusi yang sesuai atas permasalahan yang dihadapi tersebut untuk dapat menentukan daftar kebutuhan pengguna.

Tabel 4.1 Komunikasi dan Perencanaan Proyek

Nama Perusahaan	Toko Mulia Amurang
Nama Proyek	Aplikasi <i>Point of Sale</i> Berbasis Web
Manajer Proyek	William J. Olii
Pemilik Proyek	Toko Mulia
Latar Belakang Masalah	Lihat Bab I
Tujuan Kerja Praktik	Lihat Bab I
Manfaat Kerja Praktik	Lihat Bab I

Batasan Masalah	Lihat Bab I
------------------------	-------------

4.1.3 Mengidentifikasi Masalah dan Solusi

Tabel 4.2 Mengidentifikasi Masalah dan Solusi

Masalah	Solusi
Proses transaksi yang memakan waktu karena pemilik mengalami kesulitan mengingat harga dan <i>stock</i> barang.	Membangun suatu aplikasi POS yang akan membantu pemilik toko dalam melakukan proses transaksi. Aplikasi akan memperlihatkan harga barang beserta <i>stock</i> yang ada. Aplikasi juga nantinya akan menghitung jumlah yang harus dibayar dan dikembalikan.
Pembuatan nota yang masih ditulis dalam kertas yang memakan banyak waktu.	Aplikasi akan dapat membuat nota berdasarkan barang yang telah dibeli yang kemudian bisa dicetak sesuka hati oleh pemilik.
Toko tidak memiliki cara perhitungan laporan yang efektif dan pasti.	Aplikasi akan menyimpan data data pembelian yang terjadi yang kemudian bisa dilihat oleh pemilik. Aplikasi akan menunjukkan hasil keuntungan sesuai dengan perbedaan harga jual dan beli yang diberikan oleh pemilik.

4.1.4 Daftar Spesifikasi Persyaratan

Pada bagian ini dijelaskan fitur-fitur yang akan dimiliki oleh aplikasi *point of sale* yang akan dibuat

1. *Login*

Pengguna dapat masuk ke dalam aplikasi hanya jika berhasil memasukkan nama pengguna serta kata sandi yang benar sesuai dengan akun yang telah didaftarkan. Kemudian aplikasi akan memeriksa apakah akun yang dimasukan adalah akun admin atau kasir.

2. Lupa Kata Sandi

Aplikasi akan memiliki fitur lupa kata sandi yang dapat digunakan saat admin melupakan kata sandi dari akun yang dimiliki. Cara yang akan diberikan aplikasi adalah dengan memberikan pertanyaan keamanan. Jika pertanyaan berhasil dijawab dengan benar maka admin dapat mengganti kata sandi tersebut dengan kata sandi baru.

3. Mengganti Pertanyaan Keamanan

Pertanyaan keamanan yang digunakan saat pengguna melupakan kata sandi yang telah dibuat dapat diganti dengan pertanyaan dan jawaban sesuai keinginan pemilik.

4. Mengelola Data Barang

Pengguna dapat mengatur data-data barang yang ada baik membuat daftar barang yang baru maupun mengubah dan menghapusnya.

5. Mengelola Data *Sales*

Pengguna dapat mengatur data-data *sales* yang ada baik membuat data *sales* yang baru maupun mengubah dan menghapusnya.

6. Melakukan Transaksi

Pengguna dapat melakukan proses transaksi dengan cara memasukkan barang-barang yang diinginkan oleh pembeli beserta jumlahnya yang di mana aplikasi akan langsung menghitung harga yang harus dibayar beserta kembaliannya.

7. Mencetak Nota

Pengguna dapat mencetak nota dari transaksi yang telah terjadi sebelumnya.

8. Melihat Laporan

Pengguna dapat melihat keuntungan yang didapat dari hasil penjualan yang telah terjadi. Pengguna juga bisa memilih untuk melihat keuntungan yang didapatkan dalam hari, bulan, maupun tahun.

9. Akun Kasir

Admin akan dapat membuat dan mengubah akun kasir. Akun tersebut hanya akan memiliki akses pada fitur transaksi.

4.1.5 Identifikasi Sumber Daya Perangkat Lunak

Berikut merupakan identifikasi dari sumber daya perangkat lunak yang digunakan dalam pembuatan aplikasi.

Tabel 4.3 Sumber Daya Perangkat Keras Dan Perangkat Lunak

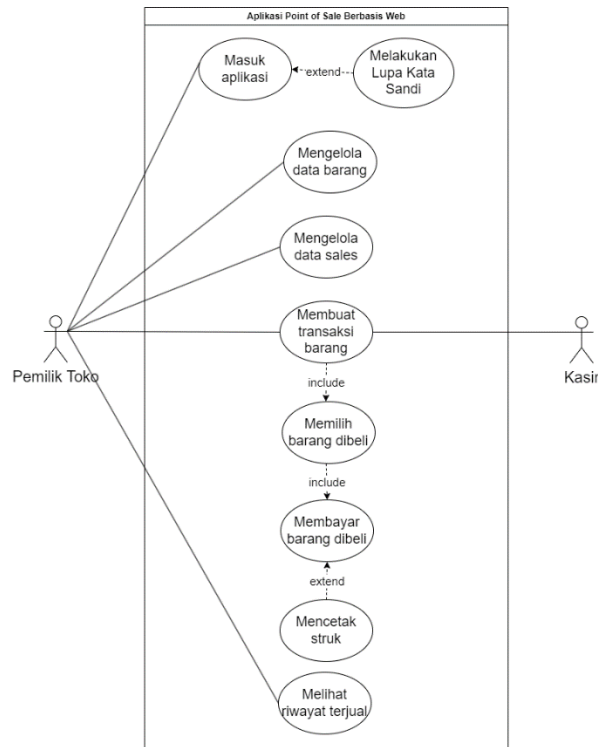
No	Sumber Daya	Keterangan
1	Perangkat Keras 1. RAM 2. <i>Processor</i> 3. <i>Harddisk</i> 4. Peralatan Lain	8 GB Ryzen 5 1 TB <i>Mouse, keyboard, printer</i>
2	Perangkat Lunak 1. <i>Desain Interface</i> 2. <i>Pemodelan Aplikasi</i> 3. DBMS 4. <i>Text Editor</i>	Figma Draw.io MySQL Visual Studio Code

4.2 Desain

Tahapan ini adalah bagian di mana desain rencana dari aplikasi dibuat. Pada tahapan ini persyaratan spesifikasi dan model dari aplikasi akan dihasilkan.

Penggambaran dari perancangan desain berada di halaman berikutnya.

4.2.1 Use Case Diagram



Gambar 4.1 Use Case Diagram

4.2.2 Use Case Table

Tabel 4.4 Use Case Table Masuk Aplikasi

Nama Use Case	Masuk Aplikasi.	
Aktor	Pemilik Toko.	
Deskripsi	Aktor memasukkan nama pengguna dan kata sandi yang jika benar maka aktor akan dapat mengakses aplikasi.	
Normal Course	Pengguna	Sistem
	Pengguna dapat mengisi nama pengguna dan kata sandi pada form login. Setelah mengisi form, pengguna dapat menekan <i>button</i> masuk dan berhasil masuk ke aplikasi	Sistem akan mengecek nama pengguna dan kata sandi jika sudah benar beserta <i>role</i> dari pengguna. Jika sudah sesuai, sistem akan mengarah ke halaman sesuai dengan role masing-masing.

<i>Alternate Course</i>	Nama pengguna dan kata sandi yang dimasukkan aktor salah.
<i>Pre-Condition</i>	Aktor telah terdaftar dalam aplikasi.
<i>Post-Condition</i>	Aktor dibawa ke halaman data barang.
<i>Assumption</i>	-

Tabel 4.5 Use Case Table Melakukan Lupa Kata Sandi

Nama Use Case	Melakukan Lupa Kata sandi.	
Aktor	Pemilik Toko.	
Deskripsi	Aktor menekan pilihan lupa kata sandi yang berada di halaman masuk.	
<i>Normal Course</i>	Pengguna	Sistem
	Pengguna dapat menekan button lupa kata sandi.	Sistem dapat mengarahkan pengguna ke halaman lupa kata sandi.
<i>Alternate Course</i>	Aktor menjawab dengan salah.	
<i>Pre-Condition</i>	Sudah ada kata sandi dan pertanyaan keamanan.	
<i>Post-Condition</i>	Aktor dapat mengganti kata sandi.	
<i>Assumption</i>	-	

Tabel 4.6 Use Case Table Mengelola Data Barang

Nama Use Case	Mengelola Data Barang.	
Aktor	Pemilik Toko.	
Deskripsi	Aktor dapat melihat data-data barang yang ada sekaligus melihat pilihan yang dapat dilakukan pada data barang tersebut.	
<i>Normal Course</i>	Pengguna	Sistem
	Pengguna dapat melihat, mengubah dan menghapus data barang yang ada di toko.	Sistem dapat menampilkan seluruh data barang, menyimpan data barang yang diubah oleh pengguna, dan menghapus data barang yang dipilih.

<i>Alternate Course</i>	-
<i>Pre-Condition</i>	Aktor berhasil masuk aplikasi.
<i>Post-Condition</i>	Mengolah data barang.
<i>Assumption</i>	-

Tabel 4.7 Use Case Table Mengelola Data Sales

Nama Use Case	Melihat Data <i>Sales</i> .	
Aktor	Pemilik Toko.	
Deskripsi	Aktor dapat melihat data-data <i>sales</i> yang ada sekaligus melihat pilihan yang dapat dilakukan pada data <i>sales</i> tersebut.	
<i>Normal Course</i>	Pengguna	Sistem
	Pengguna dapat melihat, mengubah dan menghapus data <i>sales</i> .	Sistem dapat menampilkan seluruh data <i>sales</i> , menyimpan data <i>sales</i> yang diubah oleh pengguna, dan menghapus data <i>sales</i> yang dipilih.
<i>Alternate Course</i>	-	
<i>Pre-Condition</i>	Berhasil masuk aplikasi.	
<i>Post-Condition</i>	Mengolah data <i>sales</i> .	
<i>Assumption</i>	-	

Tabel *use case table* membuat transaksi barang berada di halaman berikutnya pada Tabel 4.13.

Tabel 4.8 Use Case Table Membuat Transaksi Barang

Nama Use Case	Membuat Transaksi Barang.	
Aktor	Pemilik Toko dan Kasir.	
Deskripsi	Aktor dapat melakukan pencarian barang yang ingin dibeli dan memasukkannya kedalam keranjang serta melihat berapa harga yang harus dibayarkan.	
Normal Course	Pengguna	Sistem
	Pengguna melakukan proses transaksi barang hingga <i>checkout</i> .	sistem melakukan proses transaksi barang hingga data pembelian barang berhasil <i>checkout</i> dan tersimpan di basis data.
Alternate Course	-	
Pre-Condition	Data barang telah diisi dengan barang yang diinginkan.	
Post-Condition	Aplikasi akan menghitung harga barang yang ada.	
Assumption	-	

Tabel 4.9 Use Case Table Memilih Barang Dibeli

Nama Use Case	Memilih Barang Dibeli.	
Aktor	Pemilik Toko dan Kasir.	
Deskripsi	Aktor dapat memilih barang yang akan dibeli oleh pembeli yang di mana data barang akan muncul lengkap dengan informasi harga juga <i>stock</i> yang tersisa.	
Normal Course	Pengguna	Sistem
	Pengguna dapat melihat dan memilih <i>list</i> barang yang tersedia.	Sistem dapat menampilkan dan menyimpan list barang yang dipilih oleh pengguna.
Alternate Course	<i>Stock</i> barang tersebut tidak cukup.	
Pre-Condition	Barang telah didaftarkan ke data barang dengan <i>stock</i> yang mencukupi.	
Post-Condition	Barang yang dibeli akan muncul di layar transaksi	

Assumption	-
-------------------	---

Tabel 4.10 Use Case Table Membayar Barang Dibeli

Nama Use Case	Membayar Barang Dibeli.	
Aktor	Pemilik Toko dan Kasir.	
Deskripsi	Barang yang telah dipilih dapat dibayar yang akan menyimpan data barang yang telah dibeli dan mengurangi <i>stock</i> nya pada data barang sesuai jumlah barang yang dibeli.	
Normal Course	Pengguna	Sistem
	Pengguna mengisi tanggal transaksi dan jumlah uang yang diberikan <i>customer</i> , kemudian menekan <i>button</i> bayar untuk melakukan proses <i>checkout</i> .	Sistem menampilkan <i>form</i> untuk mengisi tanggal dan jumlah uang. Pada saat pengguna menekan <i>button</i> bayar, sistem dapat menyimpan barang keranjang dan form yang diisi oleh pengguna.
Alternate Course	Uang pelanggan kurang yang akan mengakibatkan aplikasi memberikan pesan bahwa transaksi tidak dapat dijalankan.	
Pre-Condition	Barang telah dipilih.	
Post-Condition	Data barang yang dibeli akan disimpan sehingga bisa dicetak nota dan masuk ke data laporan.	
Assumption	-	

Tabel 4.11 Use Case Table Mencetak Struk

Nama Use Case	Mencetak Struk.	
Aktor	Pemilik Toko dan Kasir.	
Deskripsi	Aktor dapat memilih untuk mencetak nota transaksi yang pernah terjadi jika diinginkan.	
Normal Course	Pengguna	Sistem

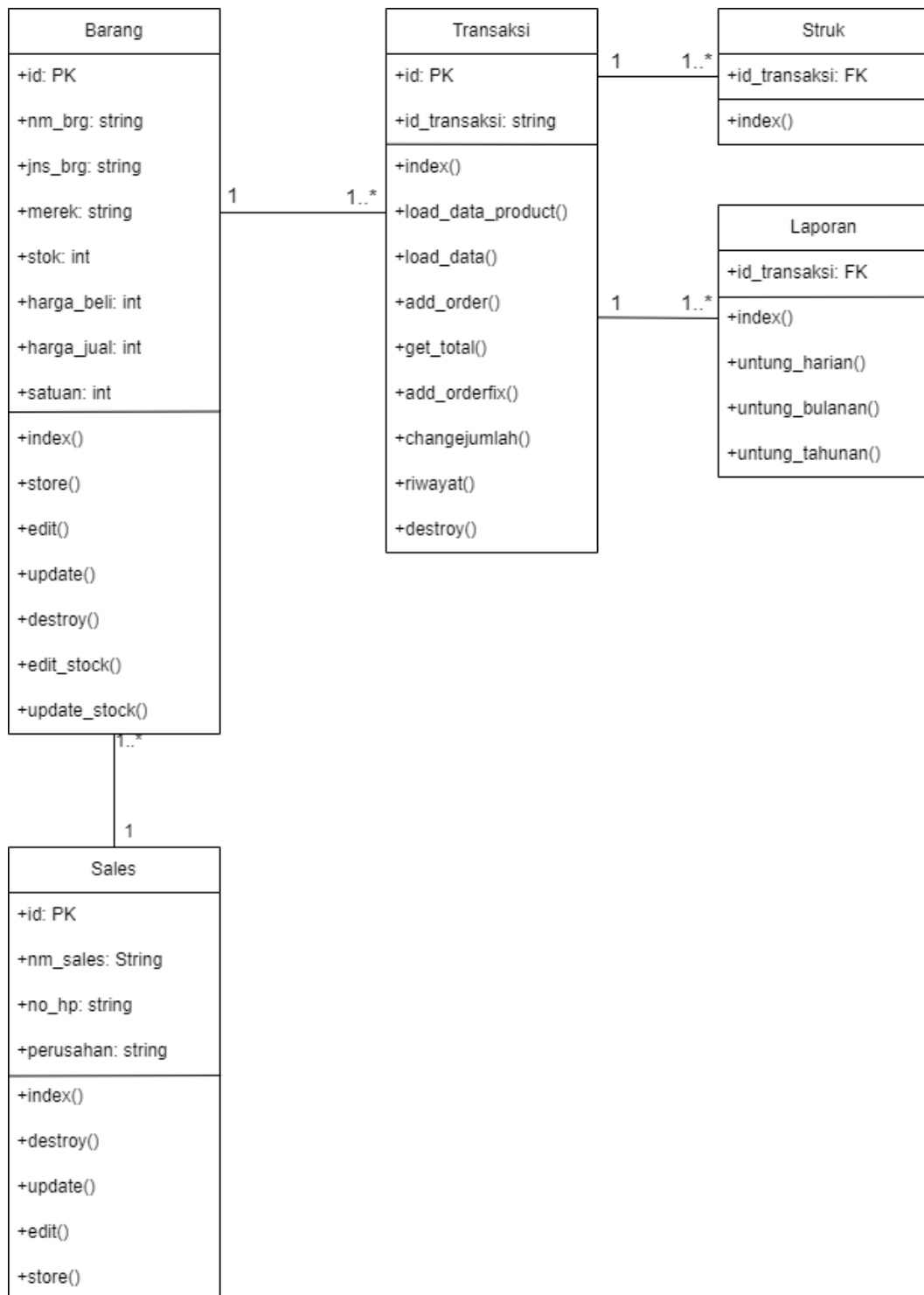
	Pengguna dapat melihat riwayat transaksi dan mencetak struk.	Sistem dapat menampilkan riwayat transaksi dan menampilkan struk dari transaksi yang dipilih oleh pengguna.
Alternate Course	-	
Pre-Condition	Telah terjadi transaksi sebelumnya.	
Post-Condition	Nota dicetak dengan bantuan <i>printer</i> .	
Assumption	-	

Tabel 4.12 Use Case Table Melihat Riwayat Terjual

Nama Use Case	Melihat Riwayat Terjual.	
Aktor	Pemilik Toko.	
Deskripsi	Aktor dapat melihat daftar barang-barang yang telah terjual beserta dengan keuntungan yang didapat baik pada hari, bulan maupun tahun yang diinginkan.	
Normal Course	Pengguna	Sistem
	Pengguna dapat melihat barang yang telah berhasil terjual dan dapat melihat keuntungan yang didapatkan baik itu harian, bulanan, atau tahunan.	Sistem menampilkan list barang terjual dan menampilkan total keuntungan yang dipilih oleh pengguna baik itu harian, bulanan, atau tahunan.
Alternate Course	-	
Pre-Condition	Telah terjadi transaksi.	
Post-Condition	-	
Assumption	-	

Penggambaran *class diagram* berada di halaman berikutnya pada sub bab 4.2.3 *class diagram*.

4.2.3 Class Diagram

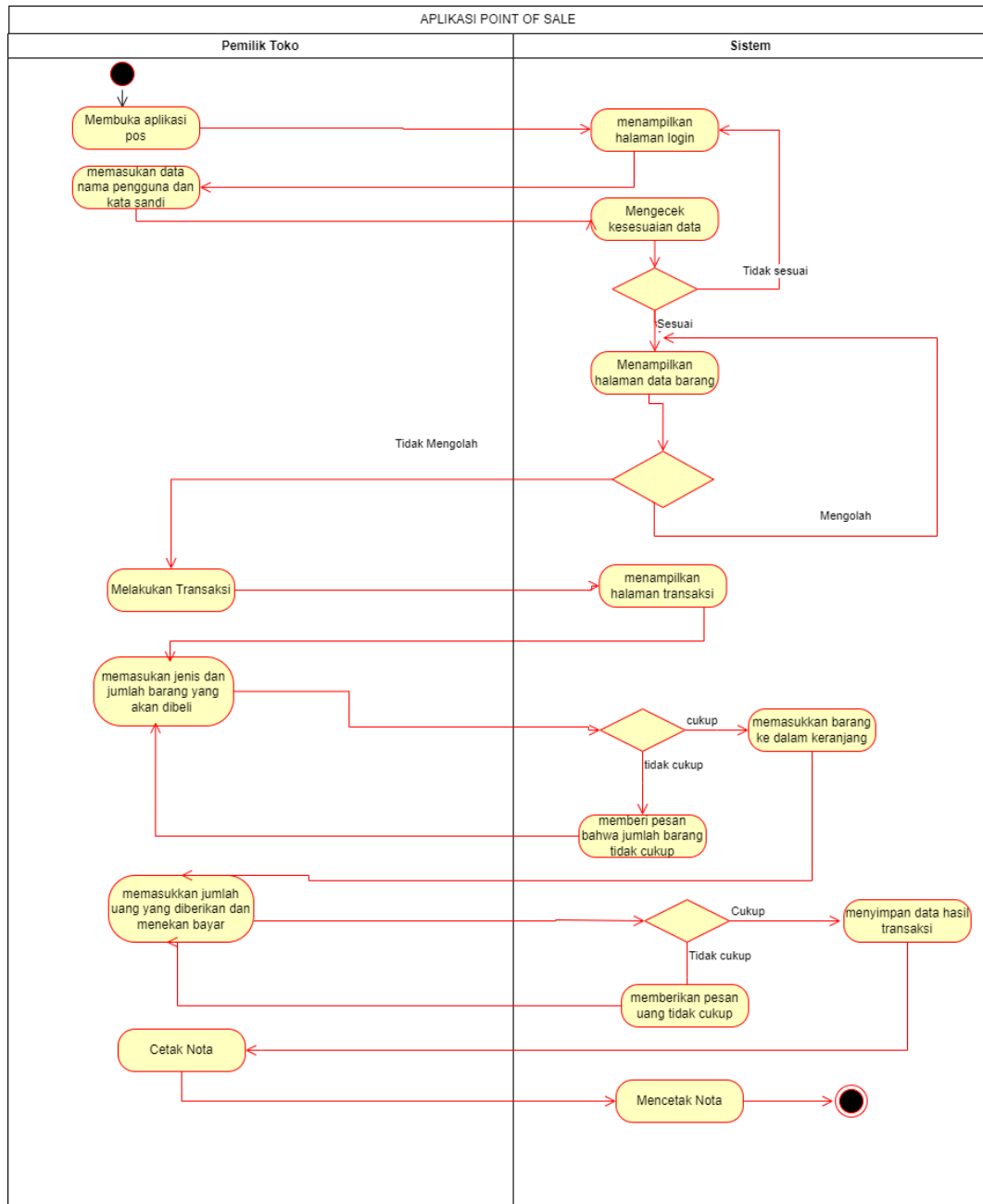


Gambar 4.2 Class Diagram

Gambar 4.2 merupakan gambar dari fitur utama yang digambarkan dalam bentuk *class diagram*. Hasil dari *class diagram* tersebut akan diimplementasikan ke dalam aplikasi.

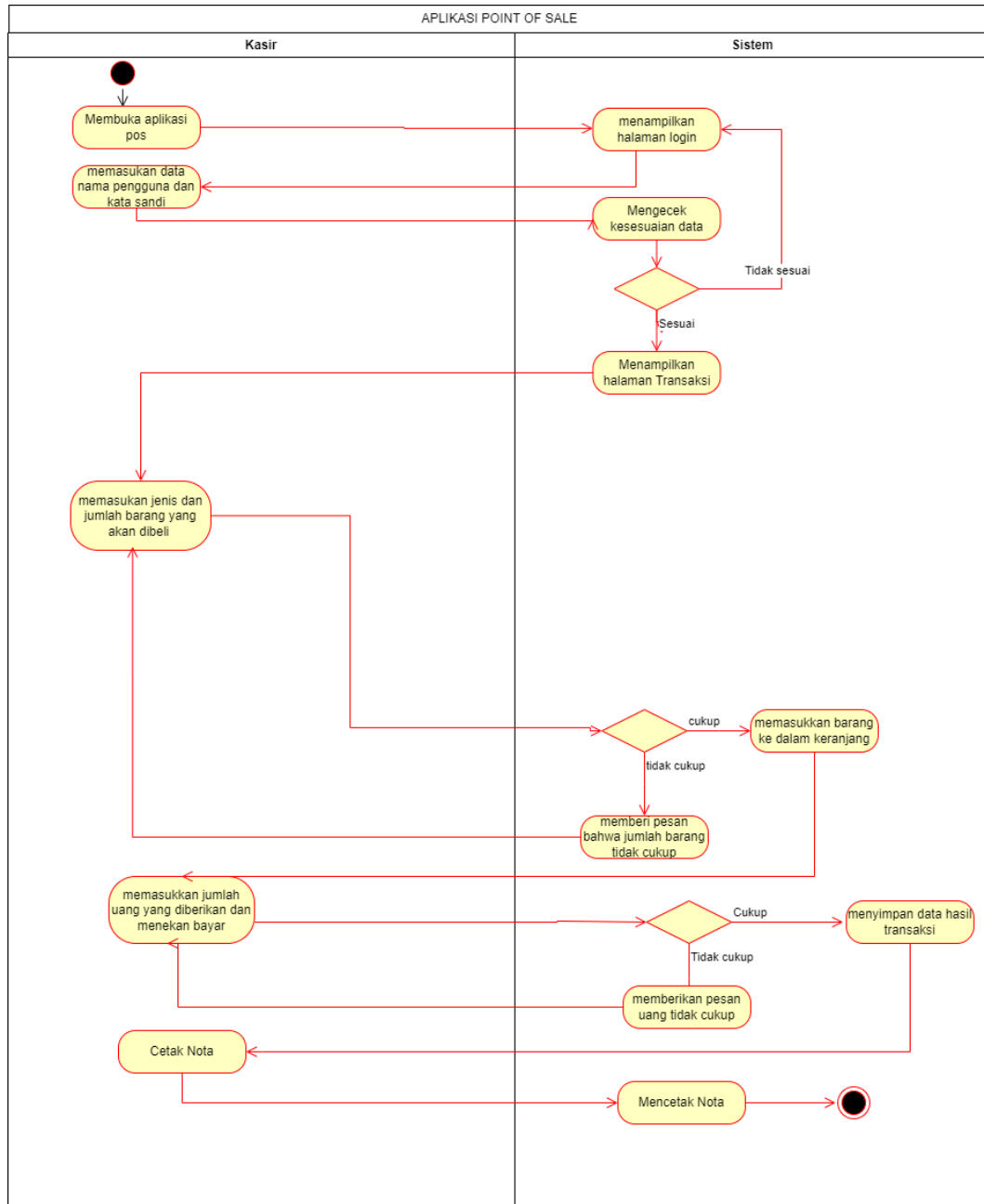
4.2.4 Activity Diagram

Berikut adalah *activity diagram* yang terjadi pada saat transaksi yang di mana aktor yang bertanggung jawab hanya pemilik toko.



Gambar 4.3 Activity Diagram Pemilik Toko

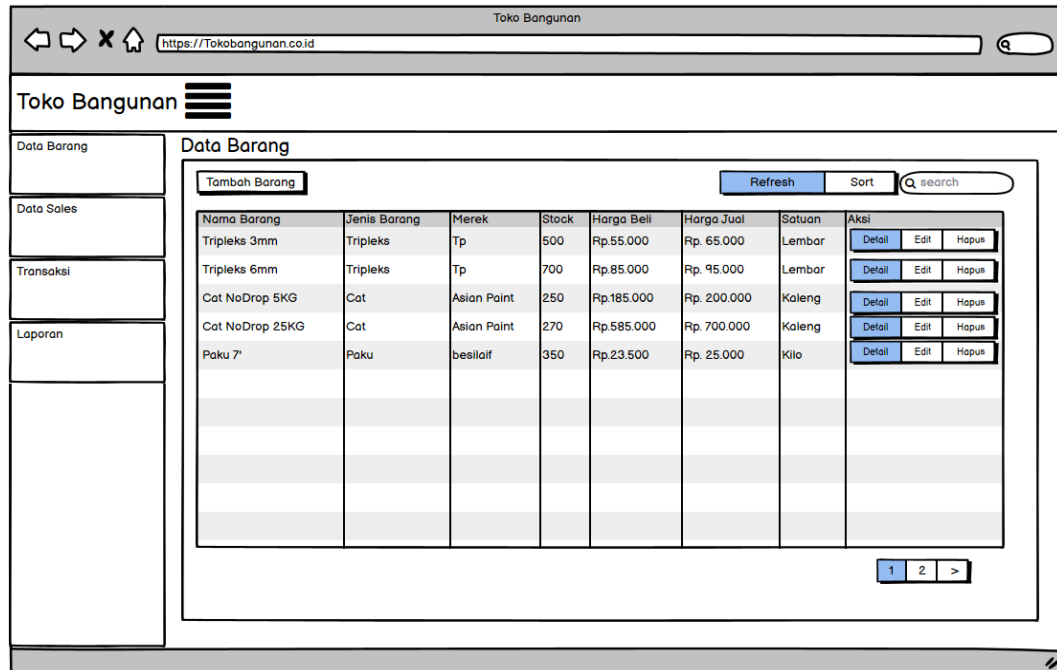
Gambar *activity diagram* kasir berada dihalaman berikutnya pada gambar 4.4.



Gambar 4.4 Activity Diagram Kasir

Gambar *storyboard* berada di halaman berikutnya pada sub bab 4.2.5.

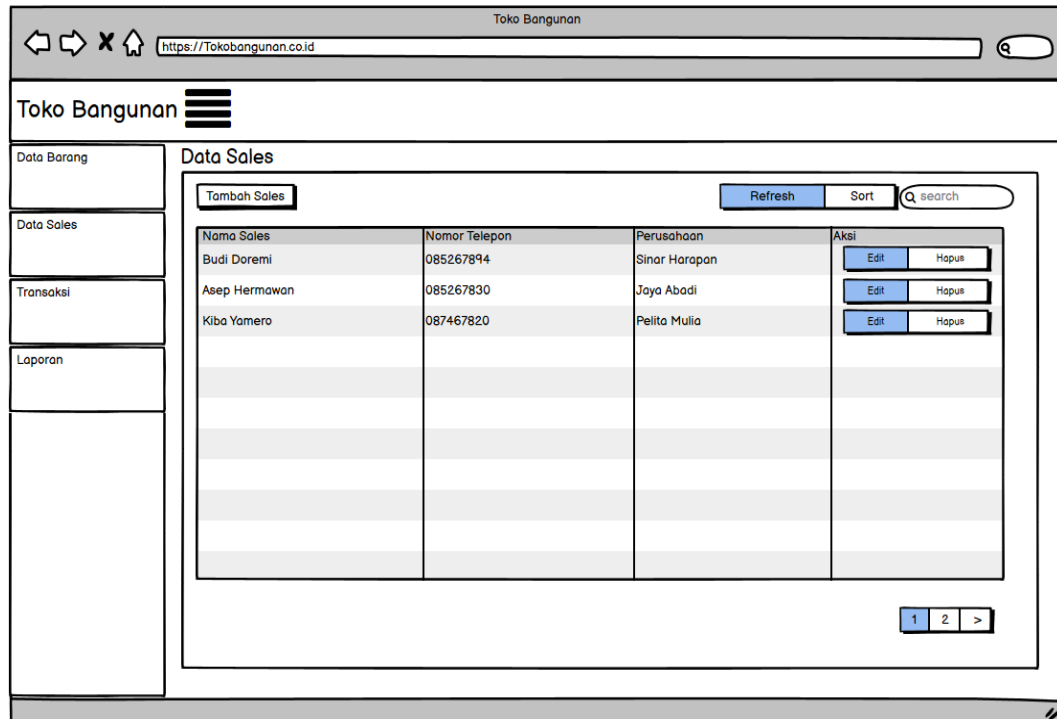
4.2.5 Storyboard



Gambar 4.5 Storyboard Halaman Barang

Fitur ini adalah fitur data barang. dalam fitur ini pengguna dapat menambah dan melihat data barang. Pengguna juga dapat mengedit dan menghapus data-data yang ada (CRUD). Dalam halaman ini pengguna juga bisa mencari data barang dan bisa melakukan sortir berdasarkan abjad.

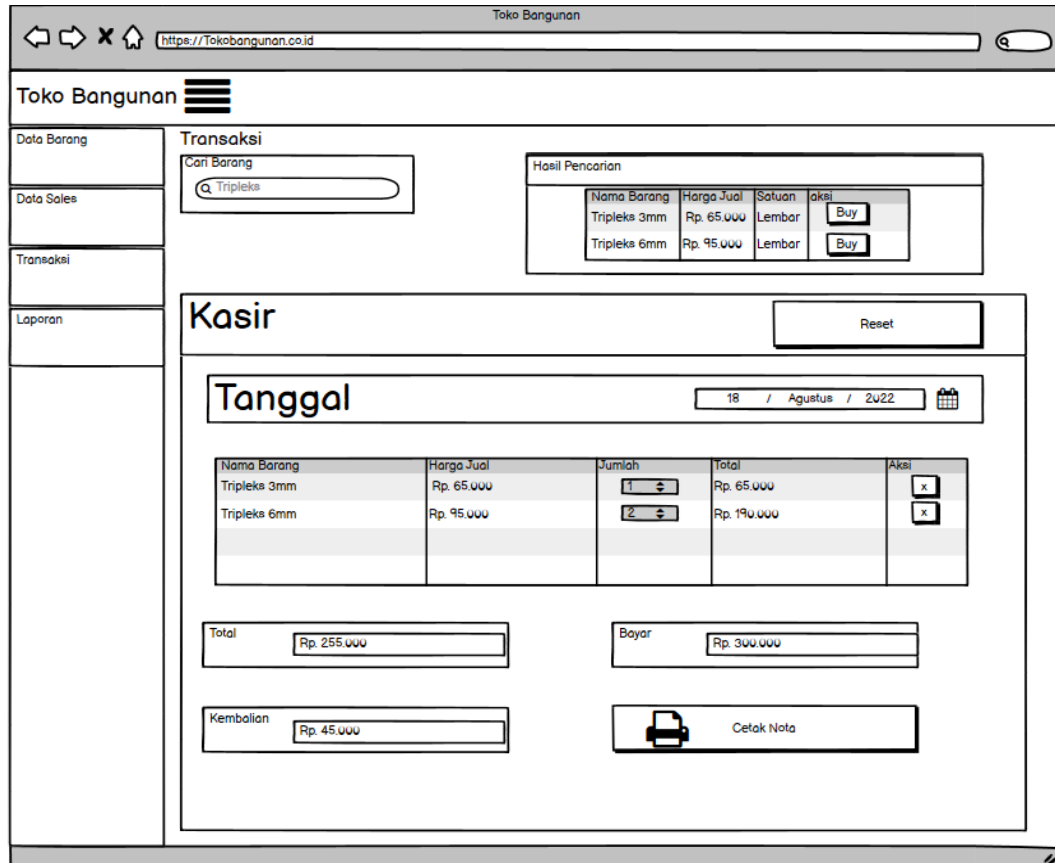
Gambar dari *storyboard* halaman *sales* berada dihalaman berikutnya pada Gambar 4.6.



Gambar 4.6 Storyboard Halaman Sales

Fitur ini adalah fitur data *sales* dalam fitur ini pengguna dapat melakukan CRUD pada data *sales* yang ada. Pengguna juga dapat melakukan pencarian dan sortir berdasarkan abjad.

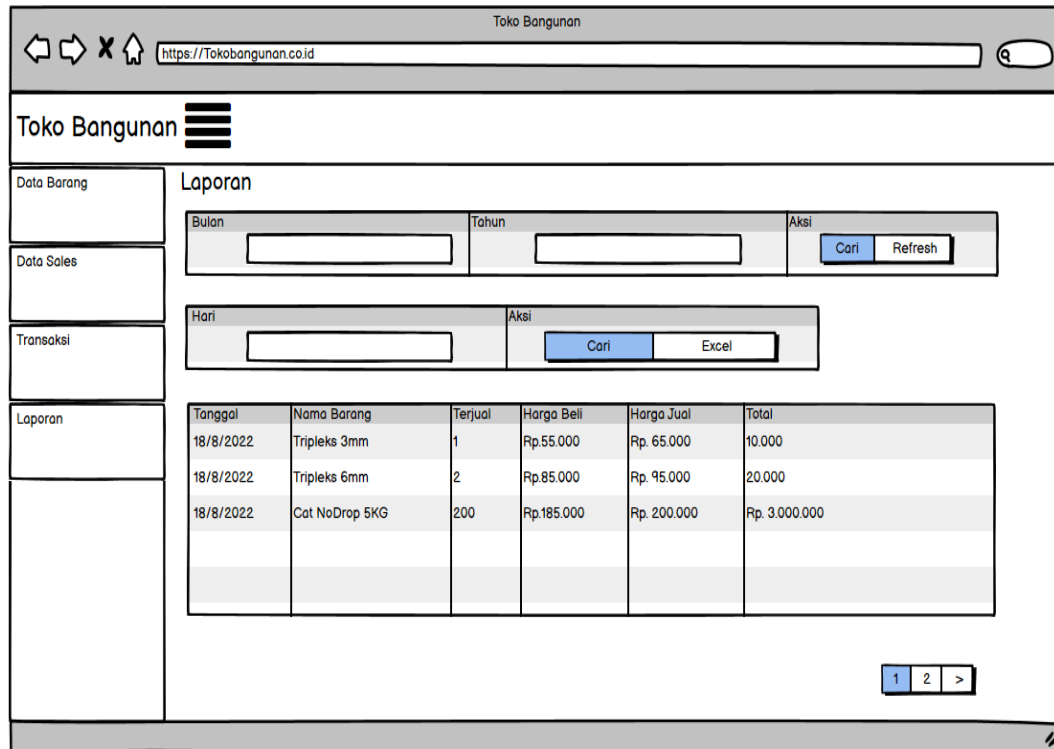
Gambar dari *storyboard* halaman transaksi berada dihalaman berikutnya pada Gambar 4.7.



Gambar 4.7 Storyboard Halaman Transaksi

Fitur ini adalah fitur transaksi. Pengguna dapat melakukan proses transaksi di mana user dapat mencari barang yang akan dibeli yang akan muncul di bagian samping hasil pencarian. Jika Pengguna memilih barang tersebut maka barang tersebut akan ditambahkan ke daftar barang yang akan dibeli. Pengguna dapat mengganti jumlah barang yang akan dibeli yang kemudian harganya akan dihitung oleh aplikasi secara otomatis. Pengguna juga dapat menghapus barang tersebut dari daftar barang yang akan dibeli. Pengguna dapat mengatur tanggal pembelian yang nantinya akan muncul di nota. Pembeli kemudian dapat memasukkan jumlah uang yang diberikan pembeli yang nantinya kembalian akan dihitung secara otomatis oleh aplikasi. jika cetak nota ditekan maka pembelian akan langsung berhasil dan aplikasi akan langsung melakukan *print* nota dengan bantuan alat cetak nota. Di saat yang sama juga *stock* di data barang juga akan berkurang sesuai barang yang dibeli.

Gambar dari *storyboard* halaman laporan berada dihalaman berikutnya pada Gambar 4.6.



Gambar 4.8 Storyboard Halaman Laporan

Fitur ini adalah fitur laporan. Dalam fitur ini, pengguna dapat melihat laporan penjualan barang yang terjadi. Dengan fitur ini pengguna dapat melihat keuntungan yang didapat. Pengguna juga dapat memilih tanggal data yang dapat ditunjukkan.

4.3 Pembuatan Kode Program

Dalam tahap ini dilakukan implementasi basis data dan *interface* aplikasi. Pada tahap ini juga akan dilakukan pemrograman serta pengujian dari aplikasi yang dibuat.

4.3.1 Implementasi Basis Data

A. Database Keseluruhan

Di bawah ini adalah gambar dari keseluruhan basis data yang digunakan dalam aplikasi POS yang dibuat.

Gambar dari implementasi basis data berada dihalaman berikutnya pada Gambar 4.9.

Tabel	Tindakan	Baris	Jenis	Penyortiran	Ukuran	Beban
<input type="checkbox"/> barang	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	4	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
<input type="checkbox"/> failed_jobs	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
<input type="checkbox"/> keranjangs	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
<input type="checkbox"/> migrations	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	11	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
<input type="checkbox"/> password_resets	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
<input type="checkbox"/> password_reset_tokens	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
<input type="checkbox"/> personal_access_tokens	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
<input type="checkbox"/> sales	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	1	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
<input type="checkbox"/> transaksi	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	8	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
<input type="checkbox"/> transaksi_detail	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	13	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
<input type="checkbox"/> users	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	1	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
11 tabel	Jumlah	38	InnoDB	utf8mb4_general_ci	368.0 KB	0 B

Gambar 4.9 Implementasi Keseluruhan Tabel

B. Tabel Barang

Di bawah ini adalah gambar dari tabel yang menyimpan data daftar barang.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	nm_brg	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada		
3	jns_brg	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada		
4	merek	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada		
5	stock	int(11)			Tidak	Tidak ada		
6	hrg_beli	int(11)			Tidak	Tidak ada		
7	hrg_jual	int(11)			Tidak	Tidak ada		
8	satuan	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada		
9	created_at	timestamp			Ya	NULL		
10	updated_at	timestamp			Ya	NULL		
11	sales_id 🔑	bigint(20)		UNSIGNED	Ya	NULL		

Gambar 4.10 Tabel Barang

Penjelasan dari tabel keranjang berada di halaman berikutnya.

C. Tabel Keranjang

Di bawah ini merupakan gambar dari tabel yang menyimpan data dari barang-barang yang dimasukkan ke dalam keranjang pembelian.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCRE
2	barang_id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		
3	jumlah	int(11)			Tidak	Tidak ada		
4	total_harga	int(11)			Tidak	Tidak ada		
5	created_at	timestamp			Ya	NULL		
6	updated_at	timestamp			Ya	NULL		
7	nm_brg	varchar(20)	utf8mb4_unicode_ci		Tidak	Tidak ada		
8	hrng_jual	int(11)			Tidak	Tidak ada		
9	hrng_beli	int(11)			Tidak	Tidak ada		

Gambar 4.11 Tabel Keranjang

D. Tabel Sales

Di bawah ini adalah gambar dari tabel yang menyimpan data para *sales*.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	nm_sales	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada		
3	no_tlp	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada		
4	perusahaan	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada		

Gambar 4.12 Tabel Sales

Penjelasan dari tabel transaksi berada di halaman berikutnya.

E. Tabel Transaksi

Di bawah ini adalah gambar dari tabel yang menyimpan data dari transaksi yang terjadi.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	tgl_transaksi	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada		
3	jam_transaksi	varchar(11)	utf8mb4_unicode_ci		Tidak	Tidak ada		
4	total	int(11)			Tidak	Tidak ada		
5	uang	int(11)			Tidak	Tidak ada		
6	kembalian	int(11)			Tidak	Tidak ada		

Gambar 4.13 Tabel Transaksi

F. Tabel Transaksi Detail

Di bawah ini adalah gambar dari tabel yang menyimpan detail barang-barang transaksi.



#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCRE
2	barang_id 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		
3	transaksi_id 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		
4	jumlah	int(11)			Tidak	Tidak ada		
5	untung	int(11)			Tidak	Tidak ada		
6	harian	varchar(50)	utf8mb4_unicode_ci		Tidak	Tidak ada		
7	bulan	varchar(50)	utf8mb4_unicode_ci		Tidak	Tidak ada		
8	tahun	varchar(50)	utf8mb4_unicode_ci		Tidak	Tidak ada		
9	total_harga	int(11)			Tidak	Tidak ada		

Gambar 4.14 Tabel Transaksi Detail

Penjelasan dari tabel user berada dihalaman berikutnya.

G. Tabel *User*

Di bawah ini adalah gambar dari tabel yang menyimpan data dari *user* yang merupakan pemilik Toko Mulia.

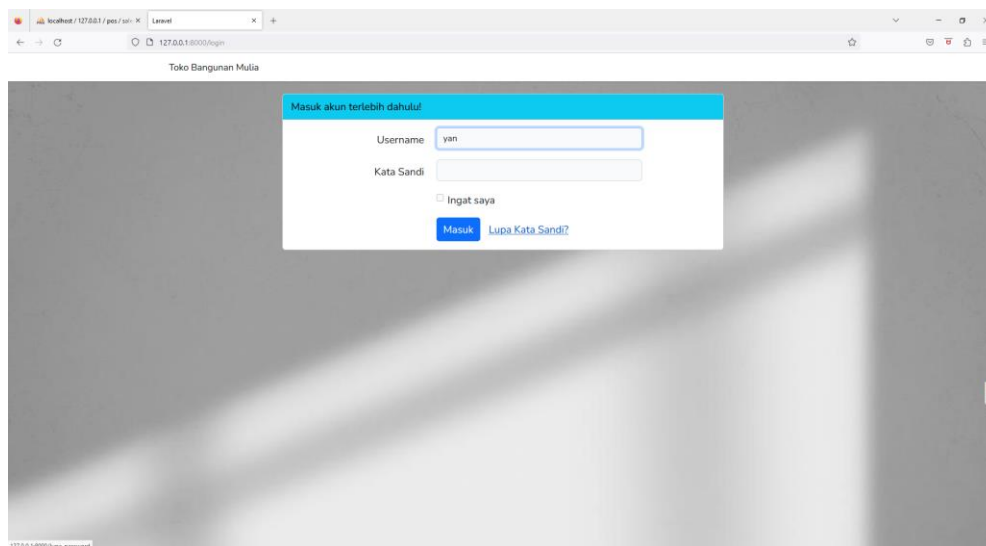
#	Name	Type	Collation	Attributes
1	id 	bigint(20)		UNSIGNED
2	name	varchar(255)	utf8mb4_unicode_ci	
3	email 	varchar(255)	utf8mb4_unicode_ci	
4	email_verified_at	timestamp		
5	password	varchar(255)	utf8mb4_unicode_ci	
6	pertanyaan	varchar(100)	utf8mb4_unicode_ci	
7	jawaban	varchar(100)	utf8mb4_unicode_ci	
8	role	varchar(11)	utf8mb4_unicode_ci	

Gambar 4.15 Tabel *User*

4.3.2 Implementasi Antarmuka

Berikut merupakan tampilan dari aplikasi *Point of Sale* yang telah dibuat beserta penjelasan mengenai kegunaan dan fitur-fiturnya.

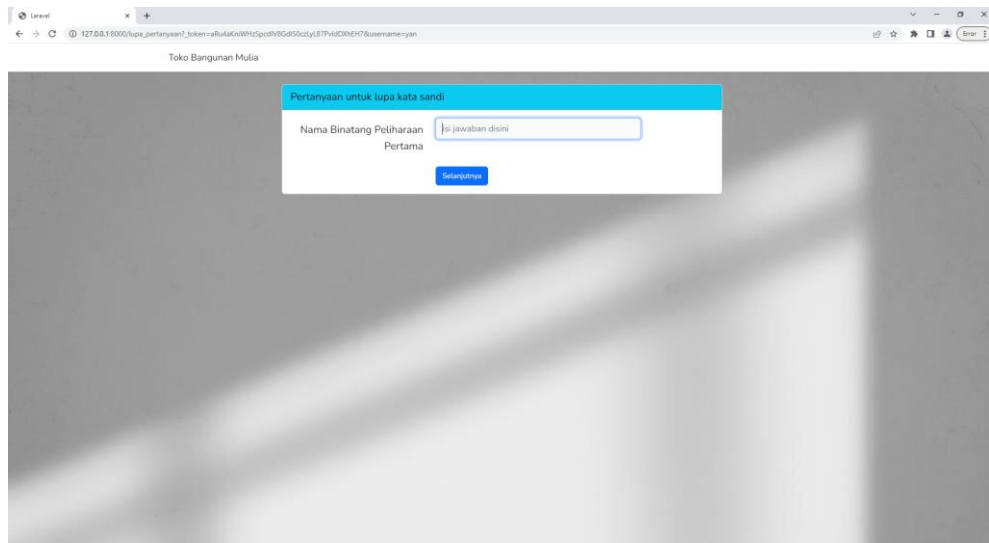
A. Halaman Masuk



Gambar 4.16 Implementasi Halaman masuk

Halaman yang terbuka pertama kali saat aplikasi dibuka. Pengguna dapat memasukkan nama pengguna dan kata sandi yang jika benar akan mengarahkan pengguna ke dalam aplikasi.

B. Halaman Lupa Kata Sandi



Gambar 4.17 Implementasi Lupa Kata Sandi

Ini adalah halaman yang akan terbuka jika pengguna menekan lupa kata sandi pada bagian halaman masuk aplikasi. Di sini pengguna akan diberikan pertanyaan keamanan yang jika benar maka pengguna akan diberikan kesempatan untuk membuat kata sandi baru.

C. Halaman Data Barang

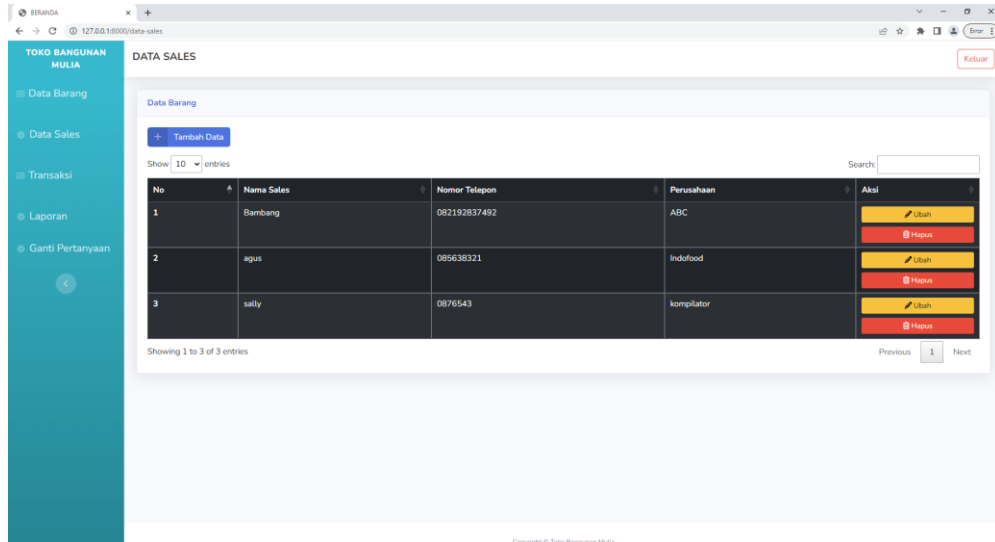
No	Nama Barang	Jenis Barang	Merek	Stock	Harga Beli	Harga Jual	Satuan	Sales	Aksi
1	Tripleks 3mm	Tripleks	Tp	453	55000	65000	Lembarr	Bambang	[Data] [Stock] [Hapus]
2	Tripleks 6mm	Tripleks	Tp	682	85000	95000	Lembar	Bambang	[Data] [Stock] [Hapus]
3	Cat NoDrop 25KG	Cat	Asian Paint	257	585000	700000	Kaleng	Bambang	[Data] [Stock] [Hapus]
4	Paku 7"	Paku	Pakubesi	493	25000	27500	Kg	agus	[Data] [Stock] [Hapus]
5	bib	mkm	opko	12	67000	72500	mm	sally	[Data] [Stock] [Hapus]

Gambar 4.18 Implementasi Halaman Barang

Ini adalah halaman yang akan pertama diarahkan jika pengguna berhasil memasukkan nama pengguna dan kata sandi yang benar. Di sini pengguna dapat melihat data-data barang yang ada mulai dari nama barang sampai nama *sales* yang

menjual barang tersebut. Data tersebut bisa ditambahkan dan di ubah sesuai kemauan pengguna.

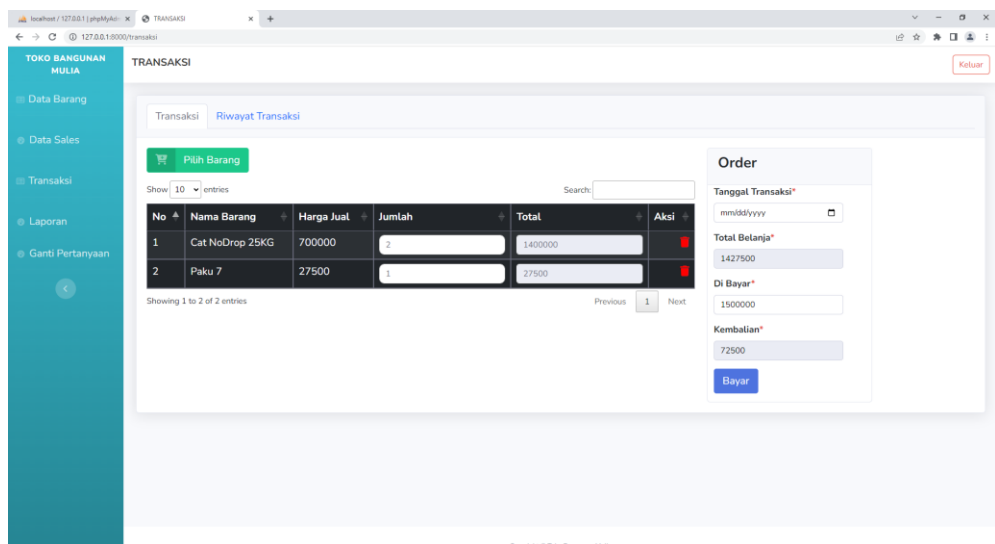
D. Halaman Data Sales



Gambar 4.19 Implementasi Halaman Sales

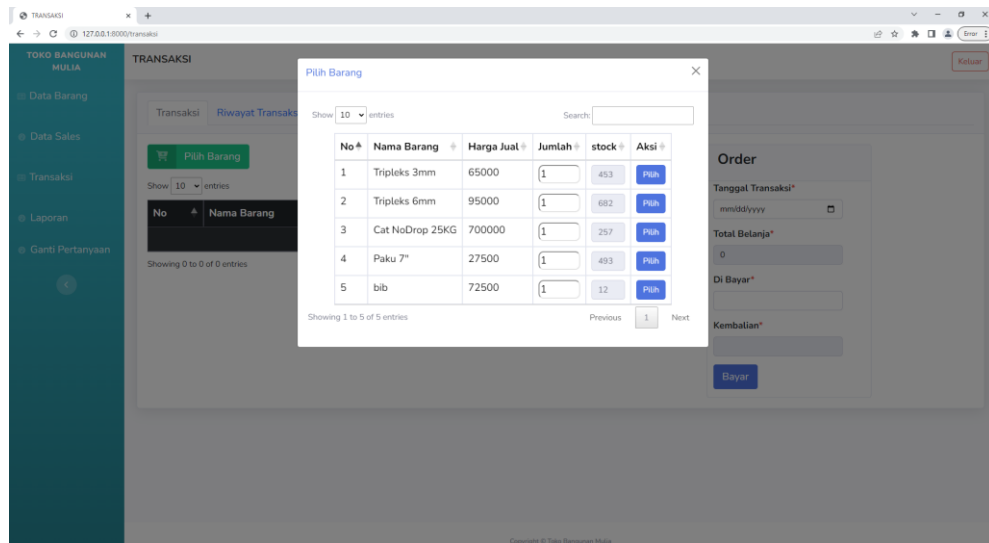
Pada halaman ini terdapat data dari para *sales* yang berupa nama, Nomor telepon, dan perusahaan asal mereka. Data yang ada pada halaman ini juga bisa ditambah dan di ubah sesuai dengan kemauan pengguna.

E. Halaman Transaksi



Gambar 4.20 Implementasi Halaman Transaksi

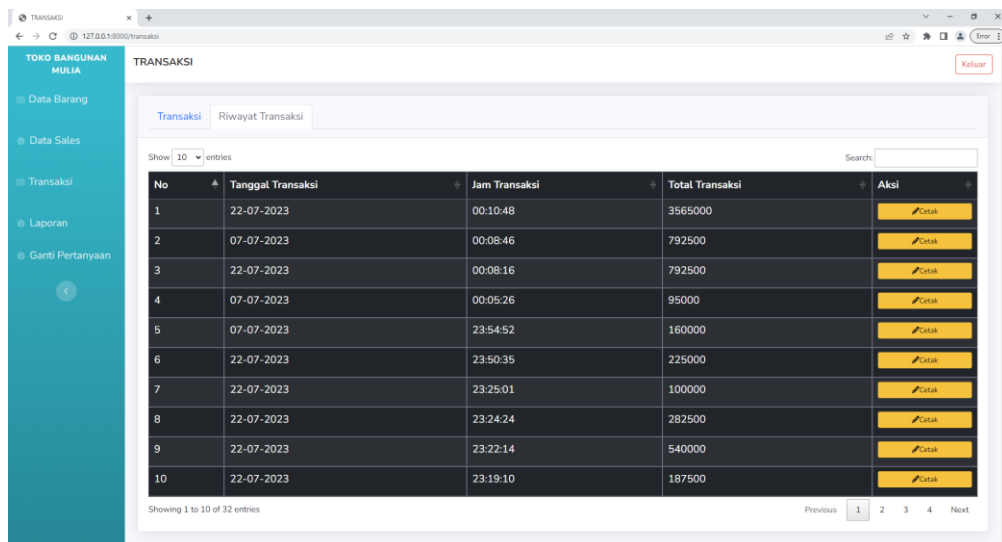
Gambar dari antarmuka halaman pilih barang berada dihalaman berikutnya pada gambar 4.5.



Gambar 4.21 Implementasi Halaman Pilih barang

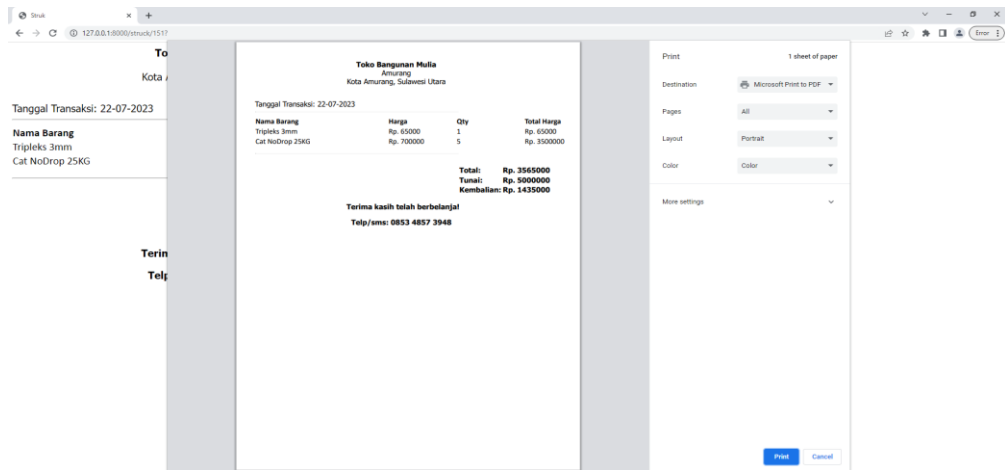
Berikut adalah halaman transaksi yang ada dalam aplikasi ini. Pada bagian ini pengguna dapat memilih barang yang akan di beli oleh pembeli beserta jumlah barang yang akan di beli. Pada bagian pemilihan barang maka akan muncul jumlah *stock* barang tersebut yang di mana *stock* tersebut akan berkurang secara otomatis pada data barang saat transaksi terjadi dan pemilik menekan tombol bayar.

F. Halaman Cetak Nota



Gambar 4.22 Implementasi Halaman Cetak Nota

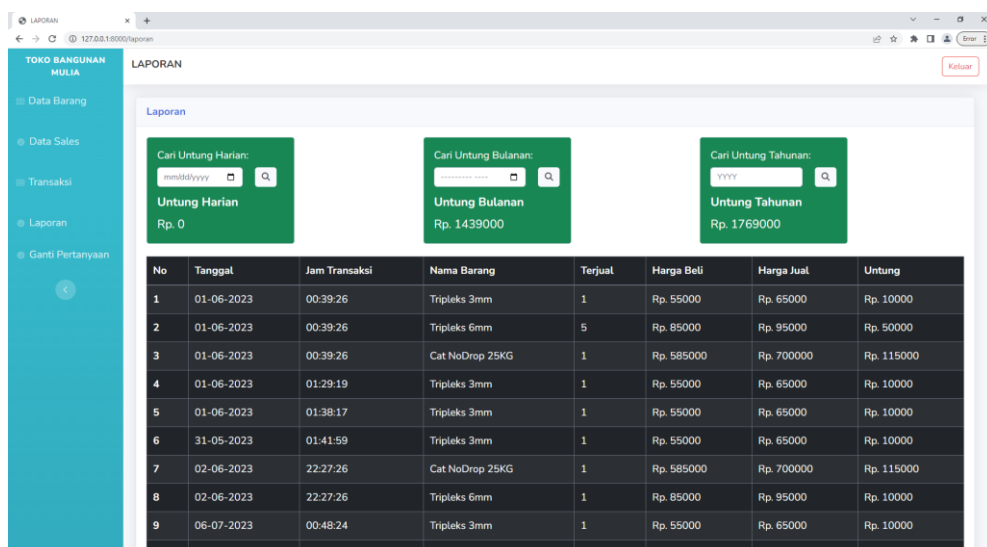
Gambar dari antarmuka halaman pilih barang berada dihalaman berikutnya pada gambar 4.5.



Gambar 4.23 Implementasi Halaman Nota

Jika pengguna telah menekan tombol bayar pada halaman transaksi maka data dari barang yang telah di beli tersebut akan di simpan pada bagian riwayat transaksi. Pengguna dapat memilih untuk mencetak nota dari transaksi yang terjadi jika diinginkan oleh pengguna.

G. Halaman Laporan Pengujian



Gambar 4.24 Implementasi Halaman Laporan Pengujian

Pada halaman ini pengguna dapat melihat laporan dari barang yang telah dijual disini pengguna dapat melihat daftar barang-barang yang telah terjadi transaksi. Pengguna juga dapat melihat berapa keuntungan yang didapat pada saat yang diinginkan baik itu pada hari,tanggal, maupun tahun manapun. Jumlah keuntungan

yang diberikan adalah sesuai selisih dari harga jual dan harga beli dari barang-barang tersebut.

H. Halaman Ganti Pertanyaan

The screenshot shows a web browser window with the URL 127.0.0.1:8000/pertanyaan. The page title is 'GANTI PERTANYAAN'. On the left is a sidebar for 'TOKO BANGUNAN MULIA' with menu items: Data Barang, Data Sales, Transaksi, Laporan, and Ganti Pertanyaan. The main content area is titled 'Ganti Pertanyaan' and contains a form with the following fields: 'Pertanyaan Sekarang' (containing 'Nama Bintang Peliharaan Pertama'), 'Pertanyaan Baru', and 'Jawaban Baru'. A blue 'Perbarui' button is at the bottom of the form. A 'Keluar' button is in the top right corner.

Gambar 4.25 Implementasi Halaman Ganti Pertanyaan

Pada halaman ini pengguna dapat mengganti pertanyaan keamanan yang ada dengan pertanyaan baru.

I. Halaman Ganti Akun Kasir

The screenshot shows a web browser window with the URL 127.0.0.1:8000/pertanyaan. The page title is 'GANTI AKUN KASIR'. On the left is a sidebar for 'TOKO BANGUNAN MULIA' with menu items: Data Barang, Data Sales, Transaksi, Laporan, Ganti Pertanyaan, and Akun Kasir. The main content area is titled 'Ganti Akun Kasir' and contains a form with the following fields: 'Username' (containing 'kasir') and 'Kata Sandi' (containing '12345678'). A blue 'Ganti' button is at the bottom of the form.

Gambar 4.26 Implementasi Halaman Ganti Akun Kasir

Pada halaman ini pengguna dapat mengganti akun kasir baik itu username atau kata sandi.

Tabel pemrograman berada di halaman berikutnya pada sub bab 4.3.3.

4.3.3 Melakukan Pemrograman

Berikut merupakan hasil pemrograman dari aplikasi *Point of Sale*

A. Fitur Barang

Menampilkan data barang
<pre>public function index() { \$barang = Barang::all(); return view('halaman.index', compact('barang')); }</pre>
Menyimpan data barang
<pre>public function store(Request \$request) { \$request -> validate(['nm_brg' => 'required', 'jns_brg' => 'required', 'merek' => 'required', 'stock' => 'required', 'harga_beli' => 'required', 'harga_jual' => 'required', 'satuan' => 'required',], ['nm_brg.required' => 'Nama barang tidak boleh kosong', 'jns_brg.required' => 'Jenis barang tidak boleh kosong', 'merek.required' => 'Merek tidak boleh kosong', 'stock.required' => 'Stock tidak boleh kosong', 'harga_beli.required' => 'Harga beli tidak boleh kosong', 'harga_jual.required' => 'Harga jual tidak boleh kosong', 'satuan.required' => 'Satuan tidak boleh kosong',]); \$barang = new Barang; \$barang -> nm_brg = \$request -> nm_brg; \$barang -> jns_brg = \$request -> jns_brg; \$barang -> merek = \$request -> merek; \$barang -> stock = \$request -> stock; \$barang -> hrg_beli = \$request -> harga_beli; \$barang -> hrg_jual = \$request -> harga_jual; \$barang -> satuan = \$request -> satuan; \$barang -> save(); return redirect('/') ->with('success_store_data', \$barang->nm_barang . " berhasil ditambahkan"); }</pre>
Mengubah data barang
<pre>public function edit(string \$id) { \$barang = Barang::find(\$id);</pre>

```

return response() -> json([
    'status' => 200,
    'barang' => $barang,
]);
}

public function update(Request $request)
{
    $request -> validate([
        'ubah_nm' => 'required',
        'ubah_jns' => 'required',
        'ubah_merek' => 'required',
        'ubah_stock' => 'required',
        'ubah_beli' => 'required',
        'ubah_jual' => 'required',
        'ubah_satuan' => 'required',
    ],
    [
        'ubah_nm.required' => 'Nama barang tidak boleh kosong',
        'ubah_jns.required' => 'Jenis barang tidak boleh
kosong',
        'ubah_merek.required' => 'Merek tidak boleh kosong',
        'ubah_stock.required' => 'Stock tidak boleh kosong',
        'ubah_beli.required' => 'Harga beli tidak boleh kosong',
        'ubah_jual.required' => 'Harga jua tidak boleh kosong',
        'ubah_satuan.required' => 'Satuan tidak boleh kosong',
    ]);

    $id_brg = $request -> brg_id;
    $barang = Barang::find($id_brg);

    $barang -> nm_brg = $request -> ubah_nm;
    $barang -> jns_brg = $request -> ubah_jns;
    $barang -> merek = $request -> ubah_merek;
    $barang -> stock = $request -> ubah_stock;
    $barang -> hrg_beli = $request -> ubah_beli;
    $barang -> hrg_jual = $request -> ubah_jual;
    $barang -> satuan = $request -> ubah_satuan;

    $barang -> save();

    return redirect('/')
        ->with('success_edit_data', $barang->nm_brg . "
berhasil Diubah");
}

```

Menghapus data barang

```

public function destroy(string $id)
{
    $brg = Barang::findorfail($id);
    $brg -> delete();

    return redirect('/')
        ->with('success_delete_data', $brg->judul . "
berhasil dihapus");
}

```

B. Fitur Sales

Menampilkan data sales
<pre>public function index() { \$sales = Sales::all(); return view('halaman.sales_sales', compact('sales')); }</pre>
Menyimpan data sales
<pre>public function store(Request \$request) { \$request -> validate(['nm_sales' => 'required', 'no_hp' => 'required', 'perusahaan' => 'required',], ['nm_sales.required' => 'Nama sales tidak boleh kosong', 'no_hp.required' => 'Nomor telepon tidak boleh kosong', 'perusahaan.required' => 'Perusahaan tidak boleh kosong',]); \$sales = new Sales; \$sales -> nm_sales = \$request -> nm_sales; \$sales -> no_tlp = \$request -> no_hp; \$sales -> perusahaan = \$request -> perusahaan; \$sales -> save(); return redirect('/data-sales') ->with('success_store_data', 'Sales ' . \$sales- >nm_sales . " berhasil ditambahkan"); }</pre>
Mengubah data sales
<pre>public function edit(string \$id) { \$sales = Sales::find(\$id); return response() -> json(['status' => 200, 'sales' => \$sales,]); } public function update(Request \$request) { \$request -> validate(['ubah_nm' => 'required', 'ubah_no' => 'required', 'ubah_perusahaan' => 'required',], ['ubah_nm.required' => 'Nama barang tidak boleh kosong', 'ubah_no.required' => 'Jenis barang tidak boleh kosong',]); }</pre>

```

    'ubah_perusahaan.required' => 'Merek tidak boleh
kosong',
    ]);

    $id_sales = $request -> sales_id;
    $barang = Sales::find($id_sales);

    $barang -> nm_sales = $request -> ubah_nm;
    $barang -> no_tlp = $request -> ubah_no;
    $barang -> perusahaan = $request -> ubah_perusahaan;

    $barang -> update();

    return redirect('/data-sales')
        ->with('success_edit_data', $barang->nm_sales . "
berhasil Diubah");
}

```

Menghapus data sales

```

public function destroy(string $id)
{
    $sales = Sales::findorfail($id);
    $sales -> delete();

    return redirect('/data-sales')
        ->with('success_delete_data', $sales->nm_sales . "
berhasil dihapus");
}

```

C. Fitur Transaksi

Menampilkan data barang untuk dipilih

```

public function load_data_product()
{
    $barang_load = Barang::all();
    echo json_encode($barang_load);
}

```

Menampilkan data barang yang telah dipilih

```

public function load_data()
{
    $barang_load = Keranjang::all();
    echo json_encode($barang_load);
}

```

menambahkan barang ke keranjang

```

public function add_order(request $request)
{
    try {

        $keranjang = new Keranjang;
        $keranjang -> barang_id = $request -> id_barang;
        $keranjang -> nm_brg = $request -> nama_barang;
        $keranjang -> hrg_jual = $request -> hrg_jual;
        $keranjang -> jumlah = $request -> jumlah;
        $keranjang -> hrg_beli = $request -> hrg_beli;
        $keranjang -> total_harga = $request -> total;
    }
}

```

```

    $keranjang -> save();

    $this->load_data();
} catch (Exception $e) {
    return response([
        'success' => false,
        'msg'      => 'nintau kapoa'
    ]);
}
}

```

Menampilkan total harga belanja

```

public function get_total(Request $request)
{
    try {
        $data = Keranjang::sum('total_harga');
        echo json_encode($data);

    } catch (Exception $e) {
        return response([
            'success' => false,
            'msg'      => 'Error : ' . $e->getMessage() .
' Line : ' . $e->getLine() . ' File : ' . $e->getFile()
        ]);
    }
}

```

Menambahkan barang yang akan dibeli

```

public function add_orderdfix(Request $request)
{
    // simpan transaksi
    $tgl_fix = Carbon::parse($request->tanggal_transaksi)-
>format('d-m-Y');

    $transaksi = new Transaksi;
    $transaksi -> tgl_transaksi = $tgl_fix;
    $transaksi -> jam_transaksi = Carbon::now('Singapore') ->
toTimeString();

    $bulan = Carbon::now()->format('F')."-" . Carbon::now()-
>format('Y');

    $transaksi -> total = $request->jumlah;
    $transaksi -> uang = $request->cash;
    $transaksi -> kembalian = $request->cashback;
    $transaksi -> save();

    // simpan transaksi detail
    $barang = Keranjang::all();
    $transaksi_terakhir = Transaksi::all() -> last();
    $ambil_order = [];

    foreach($barang as $a) {
        $total_hrg_beli = $a -> jumlah * $a -> hrg_beli;

        $untung = $a -> total_harga - $total_hrg_beli;

        if(!empty($a)) {

```

```

        $ambil_order[] = [
            'barang_id' => $a -> barang_id,
            'transaksi_id' => $transaksi_terakhir -> id,
            'jumlah' => $a -> jumlah,
            'bulan' => $bulan,
            'tahun' => Carbon::now()->format('Y'),
            'harian' => Carbon::now()->format('d-m-Y'),
            'untung' => $untung,
            'total_harga' => $a -> total_harga,
        ];
    }

    // mengurangi stok
    $brg_stok = Barang::where('id', '=', $a -> barang_id)
-> first();
    $update_stok = $brg_stok -> stock - $a -> jumlah;
    $brg_stok -> stock = $update_stok;
    $brg_stok -> update();
}

Transaksi_detail::insert($ambil_order);
Keranjang::truncate();

$transaksi_detail = Transaksi::all() -> last();

// return redirect() ->
to('/'.struck/'.$transaksi_detail->id);
$this -> load_data();

}

```

Mengganti jumlah barang setelah dipilih

```

public function changejumlah(Request $request)
{
    try {
        $id = $request->id;

        $keranjang = Keranjang::find($id);
        $keranjang -> jumlah = $request -> jml_order;
        $keranjang -> total_harga = $request -> total;

        $keranjang -> save();

        $this->load_data();
    } catch (Exception $e) {
        return response([
            'success' => false,
            'msg' => 'Error : ' . $e->getMessage() . ' Line : ' .
            $e->getLine() . ' File : ' . $e->getFile()
        ]);
    }
}

```

Menampilkan riwayat transaksi

```

public function riwayat()
{
    $transaksi = Transaksi::orderBy('id', 'desc')->get();
    echo json_encode($transaksi);
}

```

D. Fitur Laporan

Menampilkan data barang yang telah terjual
<pre> public function index() { \$transaksi = Transaksi_detail::all(); \$bulan = Carbon::now()->format('F')."-" .Carbon::now()- >format('Y'); \$untung_bulan = Transaksi_detail::where('bulan', '=', \$bulan) -> sum('untung'); \$untung_tahun = Transaksi_detail::where('tahun', '=', Carbon::now()->format('Y')) -> sum('untung'); \$untung_harian = Transaksi_detail::where('harian', '=', Carbon::now()->addDays(1)->format('d-m-Y')) -> sum('untung'); return view('halaman.laporan', compact('transaksi', 'untung_tahun', 'untung_bulan', 'untung_harian')); } </pre>
Menampilkan hasil pencarian harian
<pre> public function untung_harian(Request \$request) { \$tanggal_old = \$request -> tanggal_harian; \$tanggal_new = Carbon::createFromFormat('Y-m-d', \$tanggal_old)->format('d-m-Y'); \$bulan = Carbon::now()->format('F')."-" .Carbon::now()- >format('Y'); \$untung_harian = Transaksi_detail::where('harian', '=', \$tanggal_new) -> sum('untung'); \$untung_bulan = Transaksi_detail::where('bulan', '=', \$bulan) -> sum('untung'); \$untung_tahun = Transaksi_detail::where('tahun', '=', Carbon::now()->format('Y')) -> sum('untung'); \$transaksi = Transaksi_detail::where('harian', '=', \$tanggal_new) -> get(); return view('halaman.laporan', compact('transaksi', 'untung_harian', 'untung_bulan', 'untung_tahun')); } </pre>
Menampilkan hasil pencarian bulanan
<pre> public function untung_bulanan(Request \$request) { \$tanggal_old = \$request -> tanggal_bulanan; \$tanggal_new = Carbon::createFromFormat('Y-m', \$tanggal_old)->format('F-Y'); \$hari = Carbon::now()->format('d-m-Y'); \$untung_harian = Transaksi_detail::where('harian', '=', \$hari) -> sum('untung'); </pre>

```

        $untung_bulan = Transaksi_detail::where('bulan',
        '=', $tanggal_new) -> sum('untung');
        $untung_tahun = Transaksi_detail::where('tahun',
        '=', Carbon::now()->format('Y')) -> sum('untung');

        $transaksi = Transaksi_detail::where('bulan', '=',
        $tanggal_new) -> get();
        return view('halaman.laporan', compact('transaksi',
        'untung_harian', 'untung_bulan', 'untung_tahun'));
    }

```

Menampilkan hasil pencarian tahunan

```

public function untung_tahunan(Request $request)
    {
        $tanggal_old = $request -> tanggal_tahunan;
        // $tanggal_new = Carbon::createFromFormat('Y-m',
        $tanggal_old)->format('F-Y');
        $hari = Carbon::now()->format('d-m-Y');
        $bulan = Carbon::now()->format('F')." -
        ".Carbon::now()->format('Y');
        // dd($tanggal_old);

        $untung_harian = Transaksi_detail::where('harian',
        '=', $hari) -> sum('untung');
        $untung_bulan = Transaksi_detail::where('bulan',
        '=', $bulan) -> sum('untung');
        $untung_tahun = Transaksi_detail::where('tahun',
        '=', $tanggal_old) -> sum('untung');

        $transaksi = Transaksi_detail::where('tahun', '=',
        $tanggal_old) -> get();
        return view('halaman.laporan', compact('transaksi',
        'untung_harian', 'untung_bulan', 'untung_tahun'));
    }

```

4.4 Pengujian

untuk mengetahui apakah Aplikasi *Point of Sale* sudah berjalan dengan baik, maka perlu dilakukan pengujian terhadap fitur-fitur yang ada di website.

A. Tujuan pengujian

Tujuan pengujian yaitu untuk menguji dan memastikan apakah Aplikasi *Point of Sale* sudah sesuai dengan requirements atau belum.

B. Kriteria pengujian

1. Aplikasi telah memenuhi persyaratan pengguna dan aplikasi.
2. Aplikasi telah berjalan dengan baik dan sesuai dengan yang di harapkan.

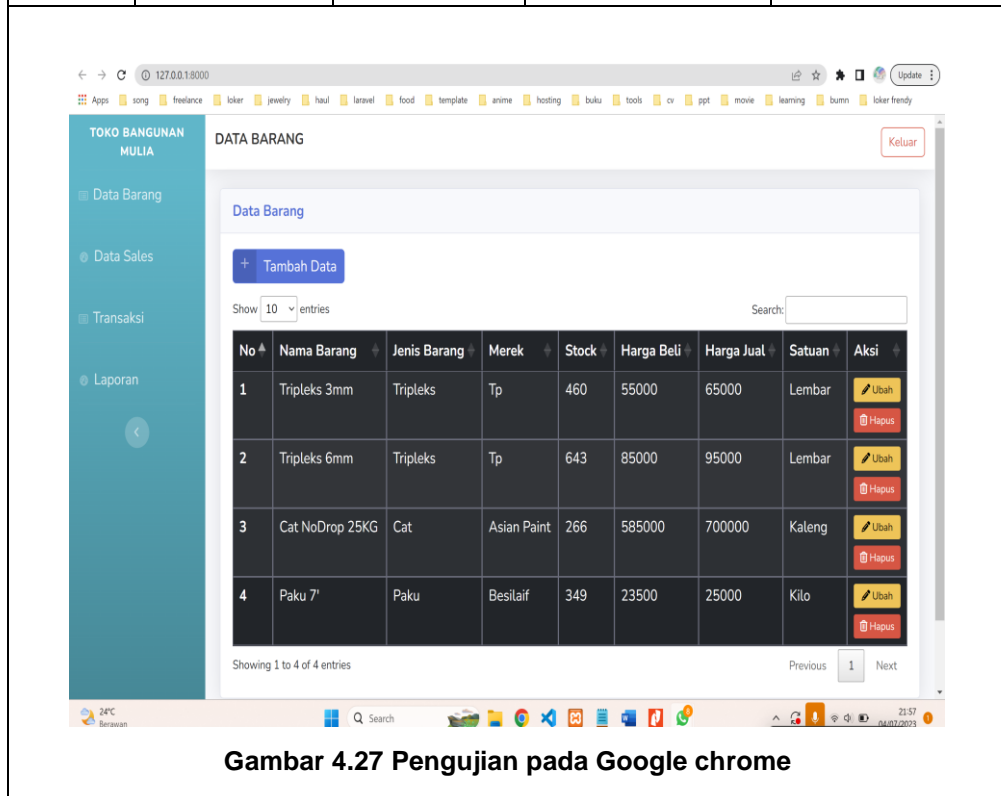
C. Kasus pengujian

1. Apakah fitur barang berjalan dengan baik?
2. Apakah aplikasi dapat dijalankan lebih dari 1 browser?
3. Apakah fitur sales berjalan dengan baik?
4. Apakah fitur transaksi beserta cetak struk berjalan dengan baik?
5. Apakah fitur Riwayat transaksi berjalan dengan baik?
6. Apakah fitur *log in* beserta fitur lupa kata sandi berjalan dengan baik ?

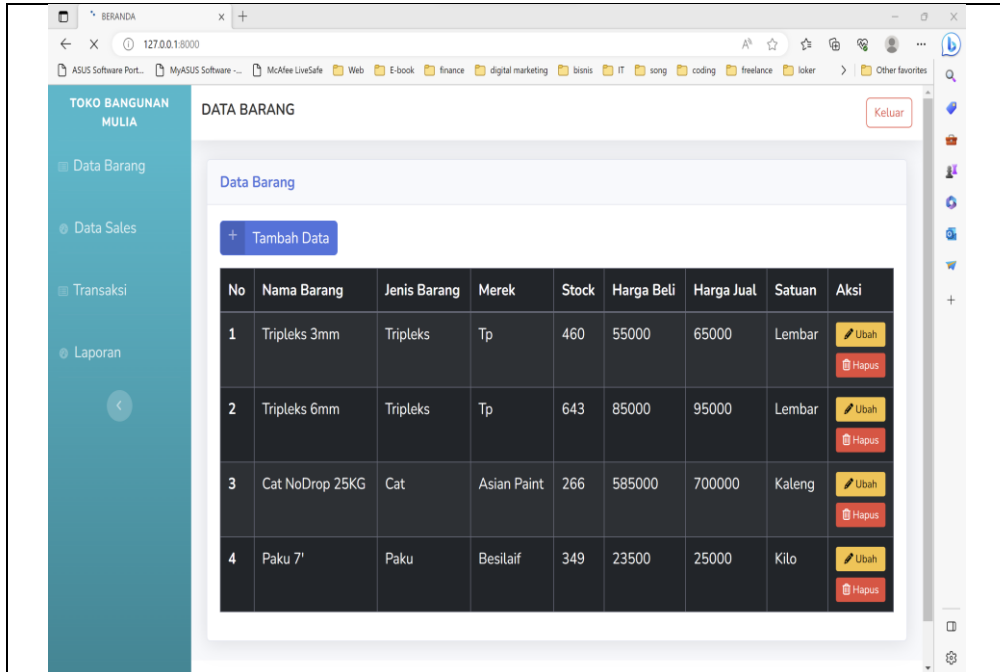
D. Pelaksanaan Pengujian

Tabel 4.13 Pelaksanaan pengujian

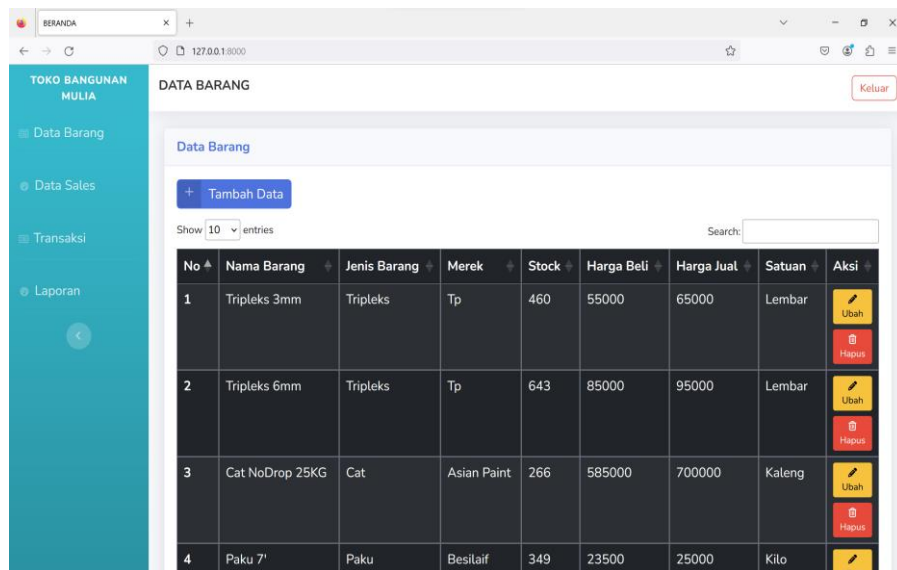
No	Kegiatan Pengujian	Pengujian	Hasil yang diharapkan	Hasil didapatkan
1	Menjalankan aplikasi di browser chrome, edge, dan firefox	Membuka aplikasi di masing-masing browser	Aplikasi dapat terbuka dengan baik di masing-masing browser	Aplikasi berhasil dijalankan dengan baik di masing-masing browser



Gambar 4.27 Pengujian pada Google chrome



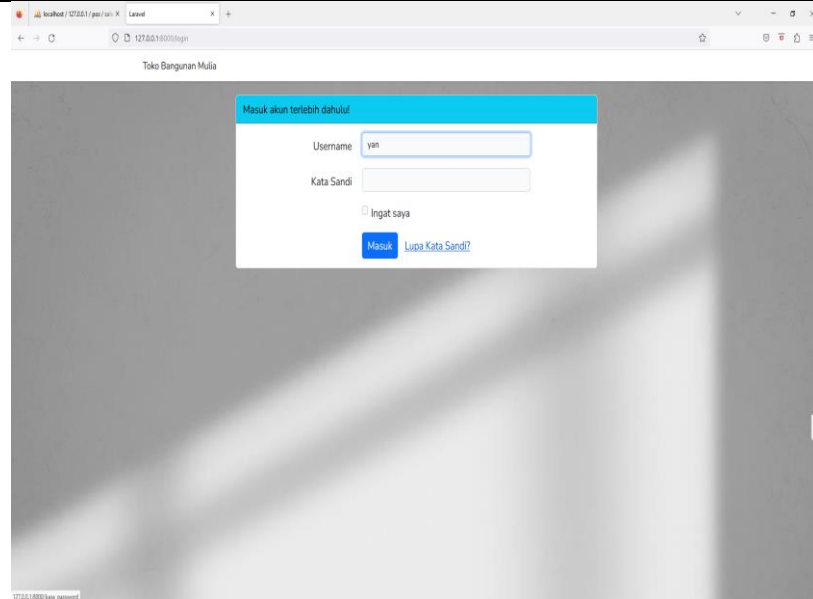
Gambar 4.28 Pengujian di Browser Edge



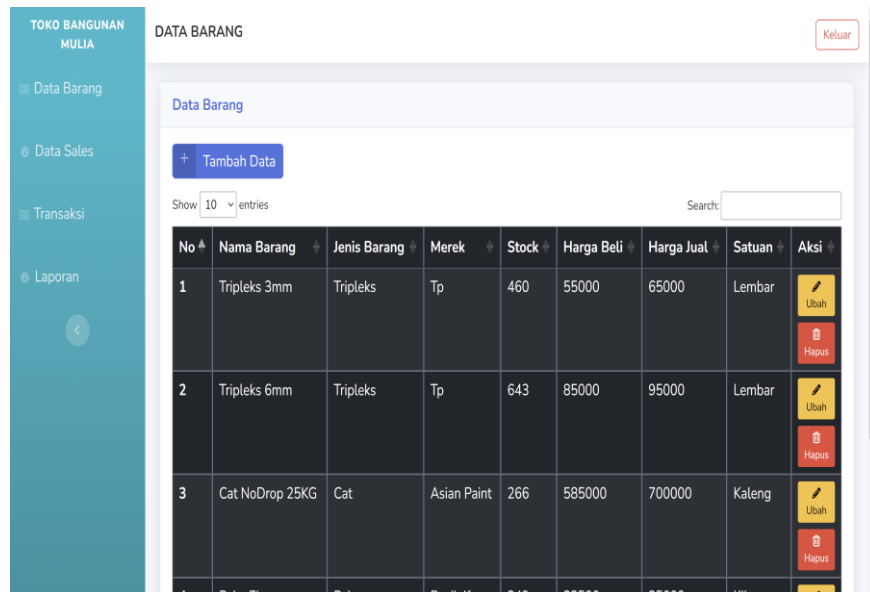
Gambar 4.29 Implementasi Pada Browser firefox

2	Melakukan proses log in	Mengisi username dan kata sandi lalu tekan	Bisa masuk ke halaman beranda setelah log in.	Berhasil masuk ke halaman beranda setelah log in.
---	-------------------------	--	---	---

		tombol masuk		
--	--	-----------------	--	--



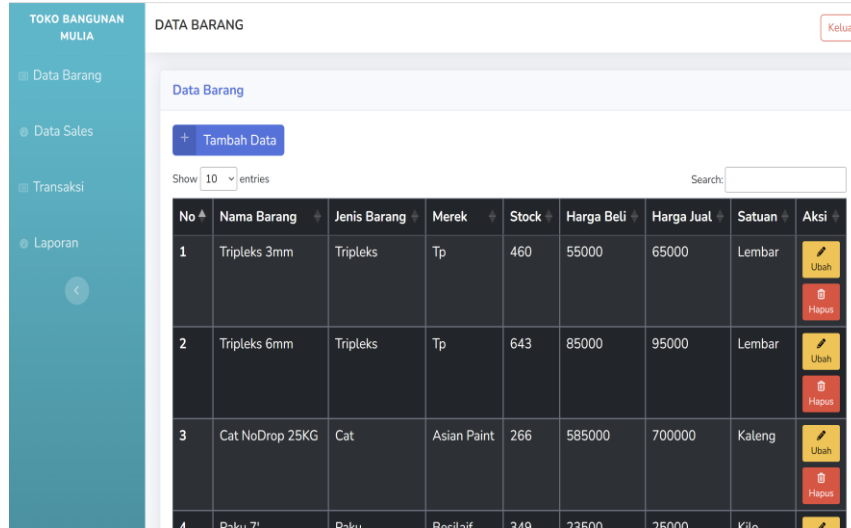
Gambar 4.30 Pengujian Halaman Login



Gambar 4.31 Pengujian Menampilkan Halaman Barang setelah *Log In*

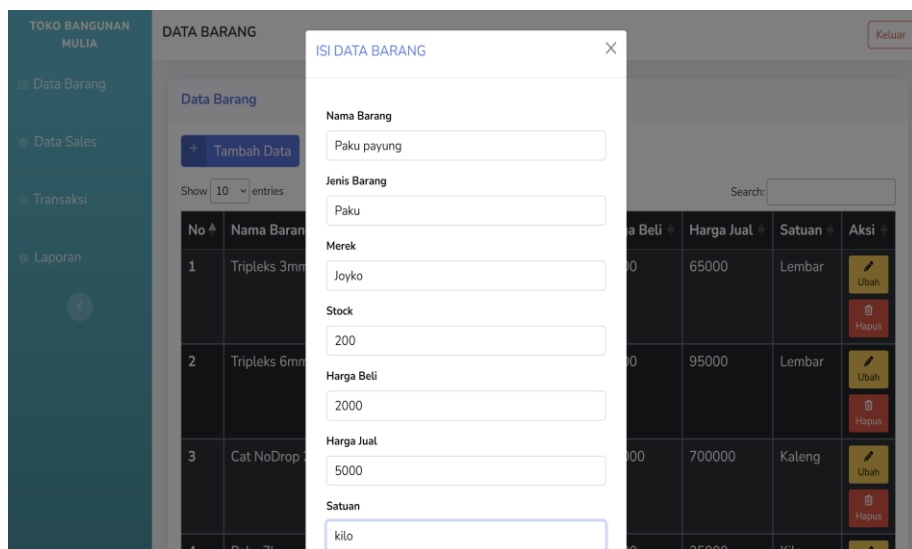
3	Menampilka n data barang pada	Membuka halaman data barang.	Halaman data barang dapat menampilkan	Halaman data barang berhasil ditampilkan.
---	-------------------------------------	------------------------------------	---	---

	halaman data barang.		seluruh data barang.	
--	----------------------	--	----------------------	--

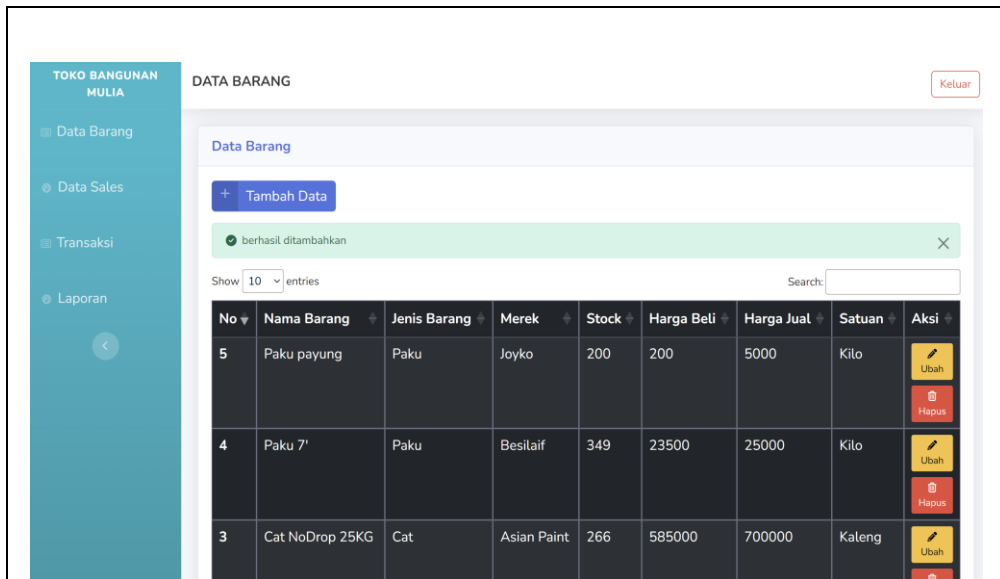


Gambar 4.32 Pengujian Menampilkan Halaman Data Barang

4	Melakukan proses tambah data pada halaman data barang.	Menekan tombol tambah data dan mengisi data barang.	Aplikasi dapat mengisi data barang.	Aplikasi berhasil mengisi data barang.
---	--	---	-------------------------------------	--

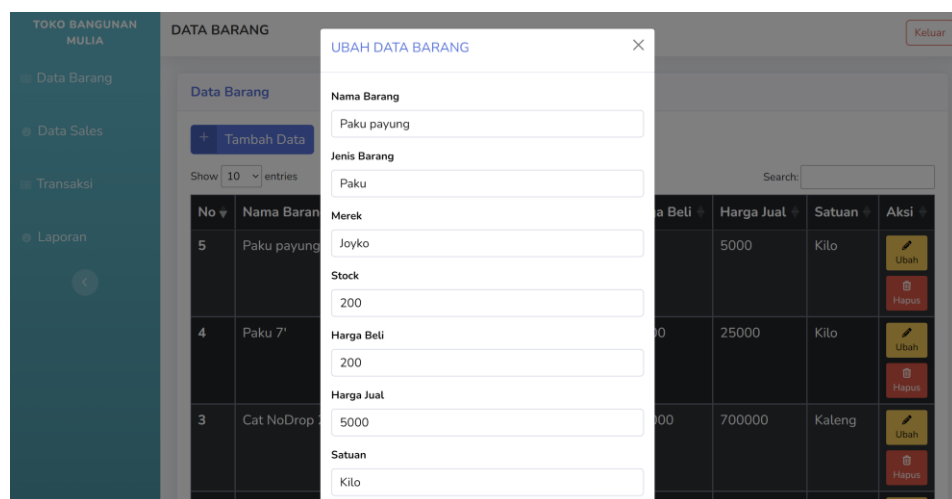


Gambar 4.33 Pengujian Halaman Isi Data Barang

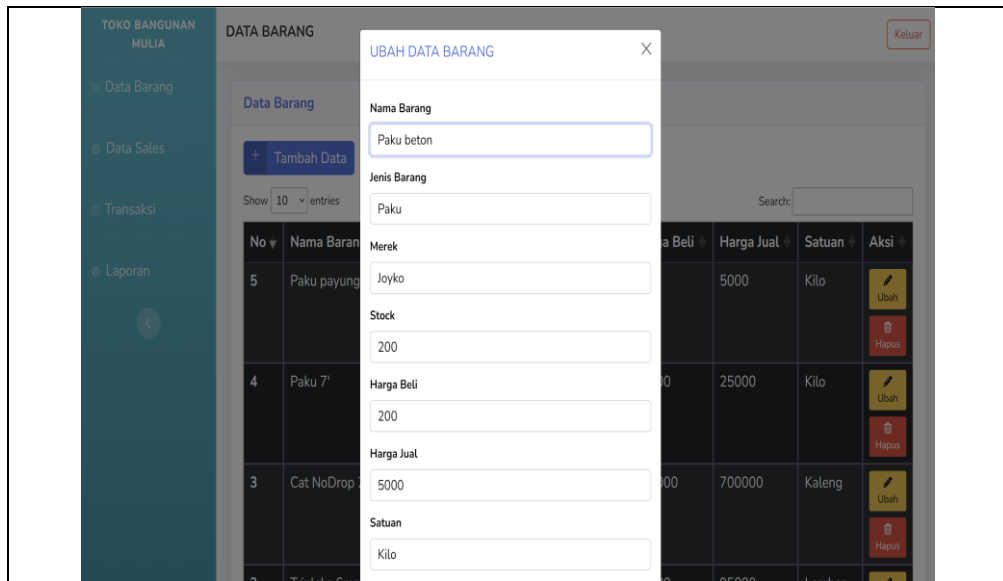


Gambar 4.34 Pengujian Data Barang Berhasil Diisi

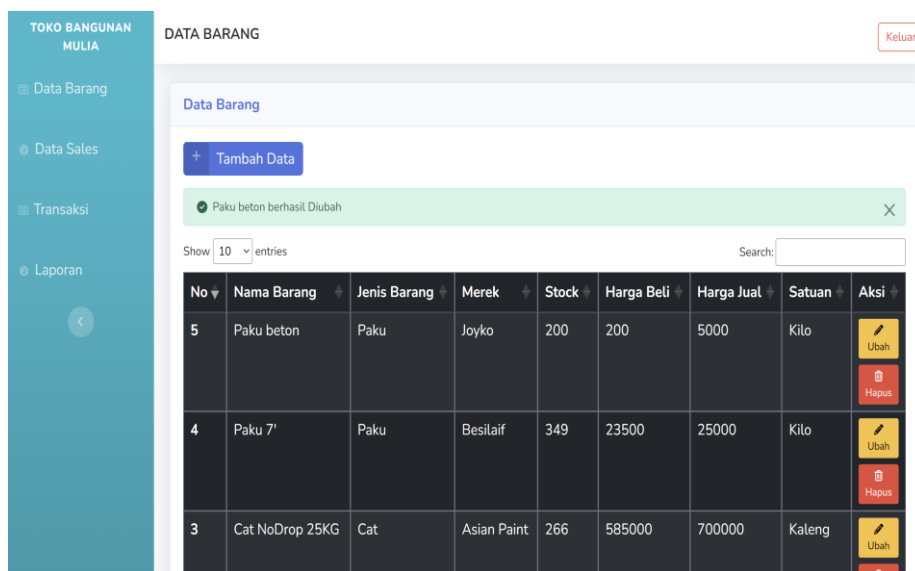
5	Melakukan proses mengubah data pada halaman data barang.	Menekan tombol mengubah data dan mengganti data barang yang ingin diganti.	Aplikasi dapat mengubah data barang.	Aplikasi berhasil mengubah data barang.
---	--	--	--------------------------------------	---



Gambar 4.35 Pengujian Data Barang Sebelum Diubah



Gambar 4.36 Pengujian Data Barang Setelah Diubah



Gambar 4.37 Pengujian Data Barang Berhasil Diubah

6	Melakukan proses menghapus data barang.	Menekan tombol hapus.	Aplikasi dapat berhasil menghapus data barang.	Aplikasi berhasil menghapus data barang.
---	---	-----------------------	--	--

No	Nama Barang	Jenis Barang	Merek	Stock	Harga Beli	Harga Jual	Satuan	Aksi
4	Paku 7'	Paku	Besilaif	349	23500	25000	Kilo	Ubah Hapus
3	Cat NoDrop 25KG	Cat	Asian Paint	266	585000	700000	Kaleng	Ubah Hapus
2	Tripleks 6mm	Tripleks	Tp	643	85000	95000	Lembar	Ubah Hapus

Gambar 4.38 Pengujian Data Setelah Dihapus

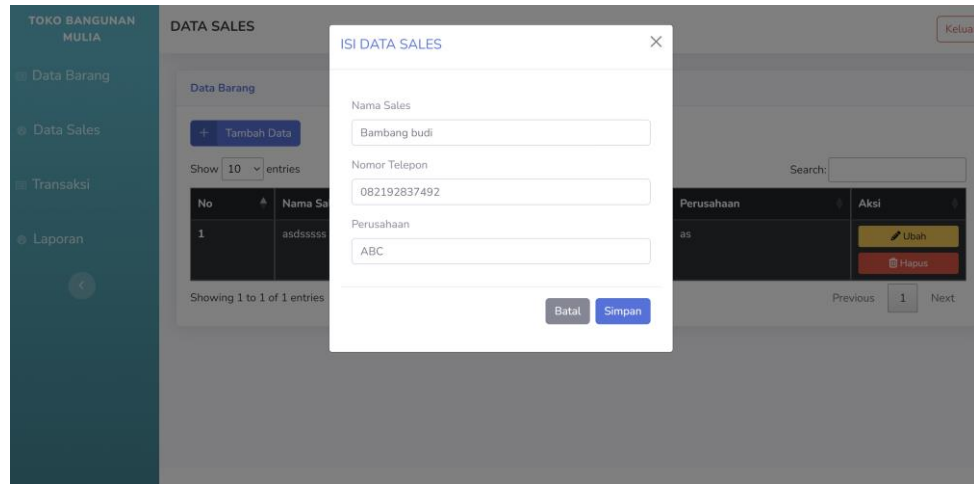
7	Menampilkan data sales pada halaman data sales.	Membuka halaman data sales.	Halaman data sales dapat menampilkan seluruh data sales.	Halaman data sales berhasil ditampilkan.
---	---	-----------------------------	--	--

No	Nama Sales	Nomor Telepon	Perusahaan	Aksi
1	asdsdsss	1212	as	Ubah Hapus

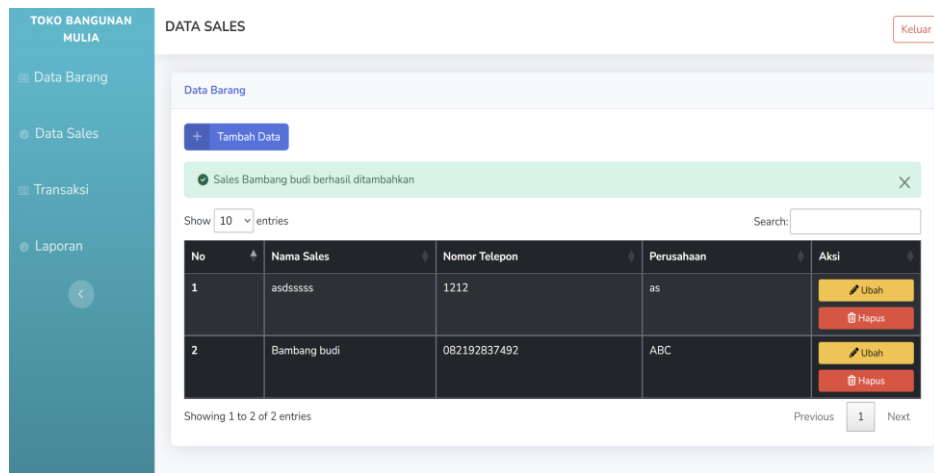
Gambar 4.39 Pengujian Halaman Data Sales

8	Melakukan proses tambah data pada	Menekan tombol tambah data	Aplikasi dapat mengisi data barang.	Aplikasi berhasil mengisi data barang.
---	-----------------------------------	----------------------------	-------------------------------------	--

	halaman data barang.	dan mengisi data barang.		
--	----------------------	--------------------------	--	--

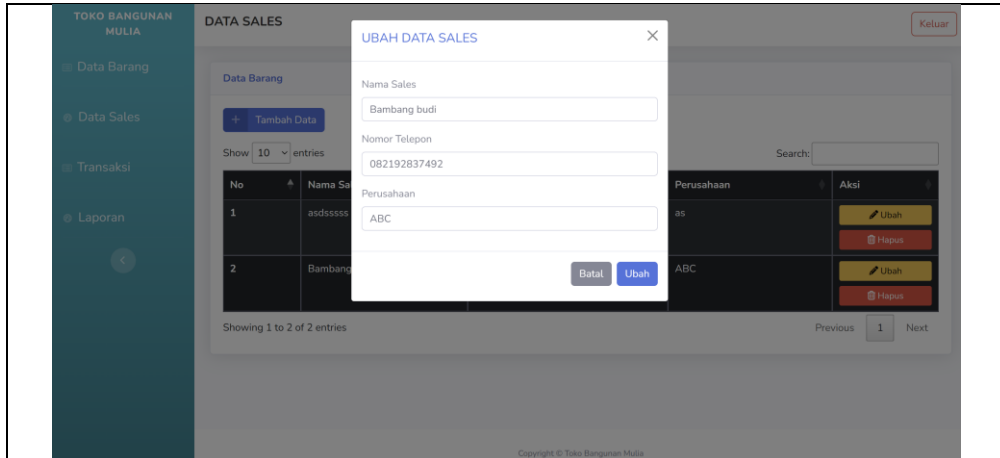


Gambar 4.40 Pengujian Halaman Mengisi Data Sales

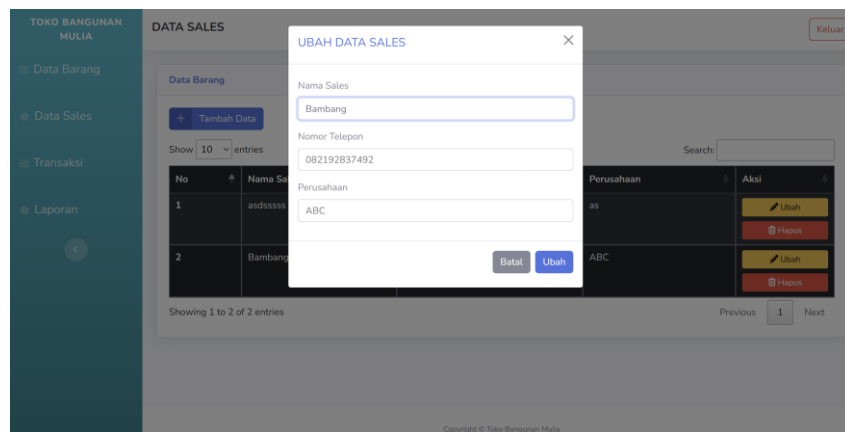


Gambar 4.41 Pengujian Data Sales Berhasil Ditambah

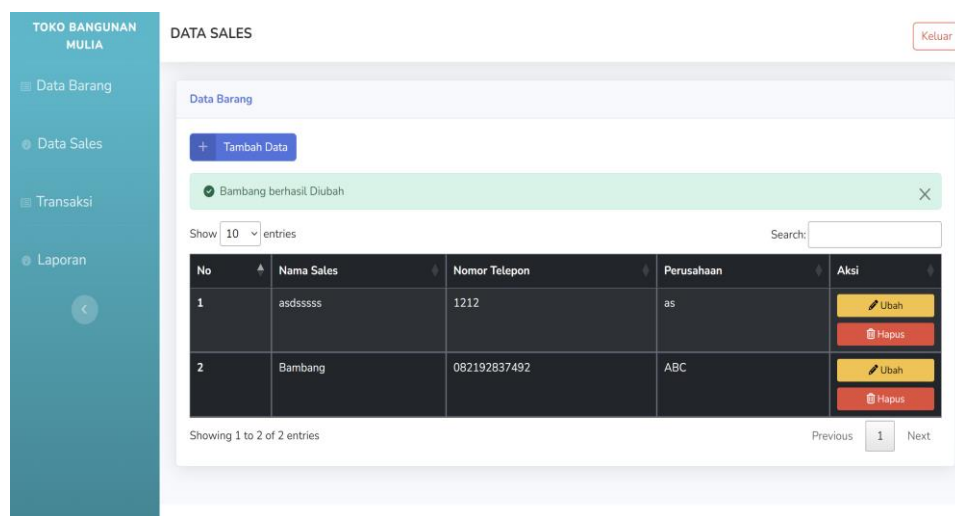
9	Melakukan proses mengubah data pada halaman data sales.	Menekan tombol mengubah data dan mengganti data sales yang ingin diganti.	Aplikasi dapat mengubah data sales.	Aplikasi berhasil mengubah data sales.
---	---	---	-------------------------------------	--



Gambar 4.42 Pengujian Halaman Data Sales Sebelum Diganti

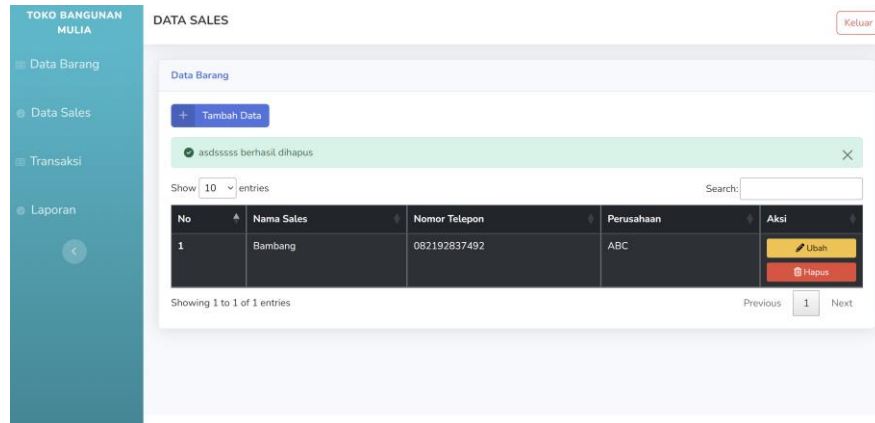


Gambar 4.43 Pengujian Halaman Data Sales Setelah Diubah



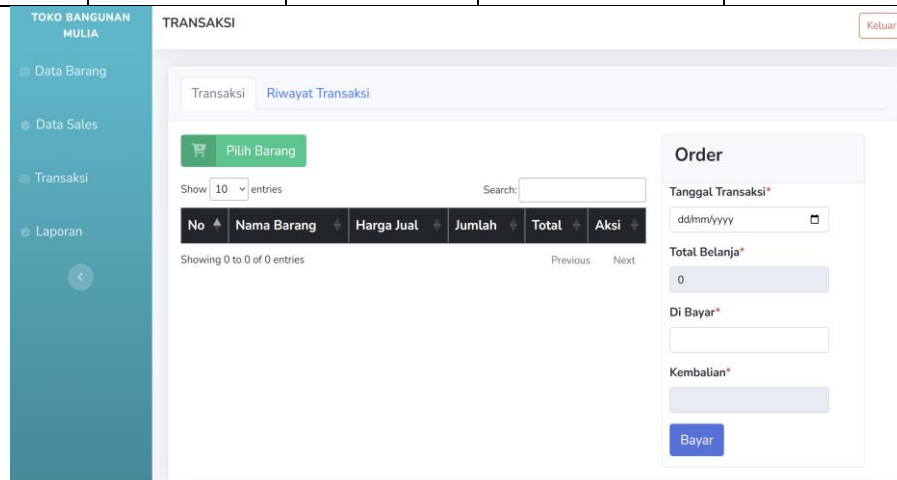
Gambar 4.44 Pengujian Halaman Sales Setelah Ditekan Tombol Ubah

10	Melakukan proses menghapus data sales.	Menekan tombol hapus.	Aplikasi dapat berhasil menghapus data sales.	Aplikasi berhasil menghapus data sales.
----	--	-----------------------	---	---

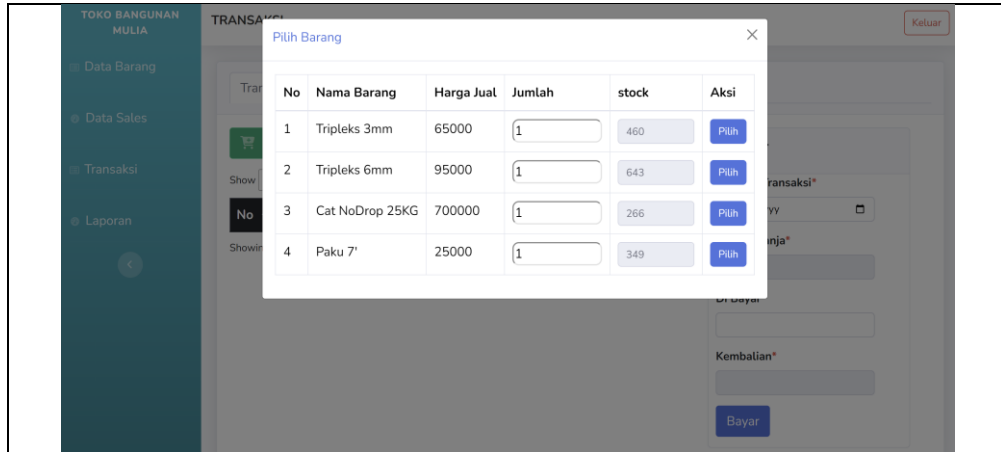


Gambar 4.45 Pengujian Halaman Sales Setelah Dihapus Data Sales

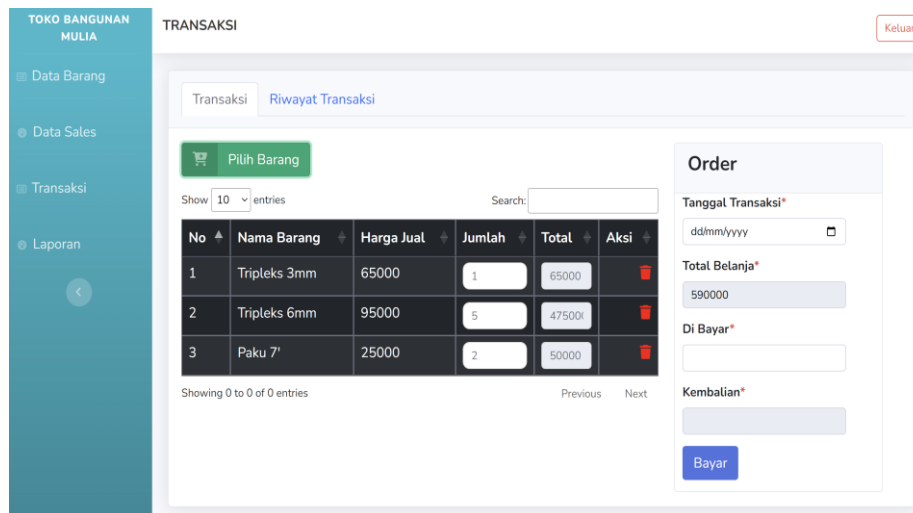
11	Melakukan proses memilih barang yang akan dibeli.	Menekan tombol pilih barang lalu dimasukkan jumlah barang dan tekan tombol pilih.	Pengguna dapat memilih barang dan aplikasi dapat menampilkan barang yang dipilih.	Pengguna berhasil memilih barang dan aplikasi berhasil menampilkan barang yang dipilih.
----	---	---	---	---



Gambar 4.46 Pengujian Halaman Transaksi Sebelum Memilih Barang

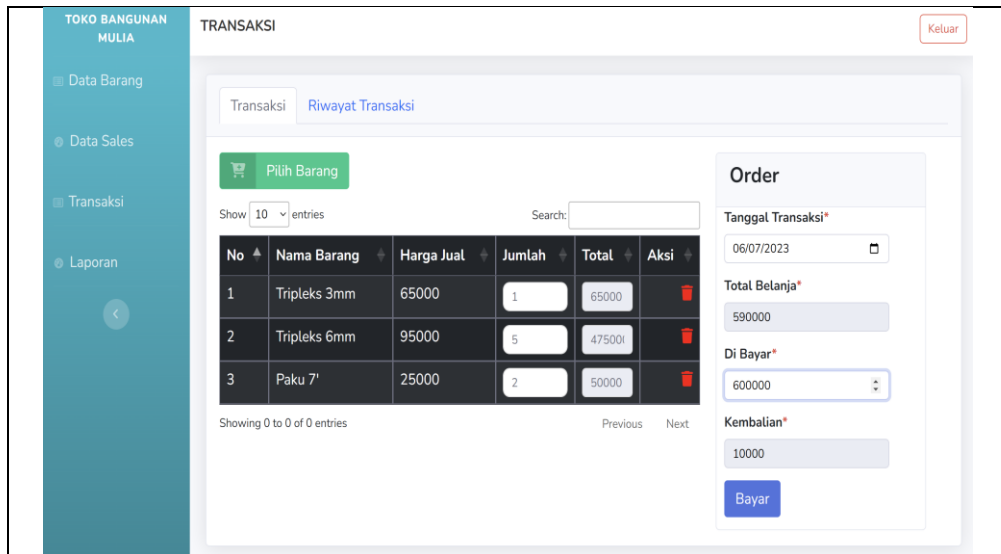


Gambar 4.47 Pengujian Halaman Transaksi Pilih Barang

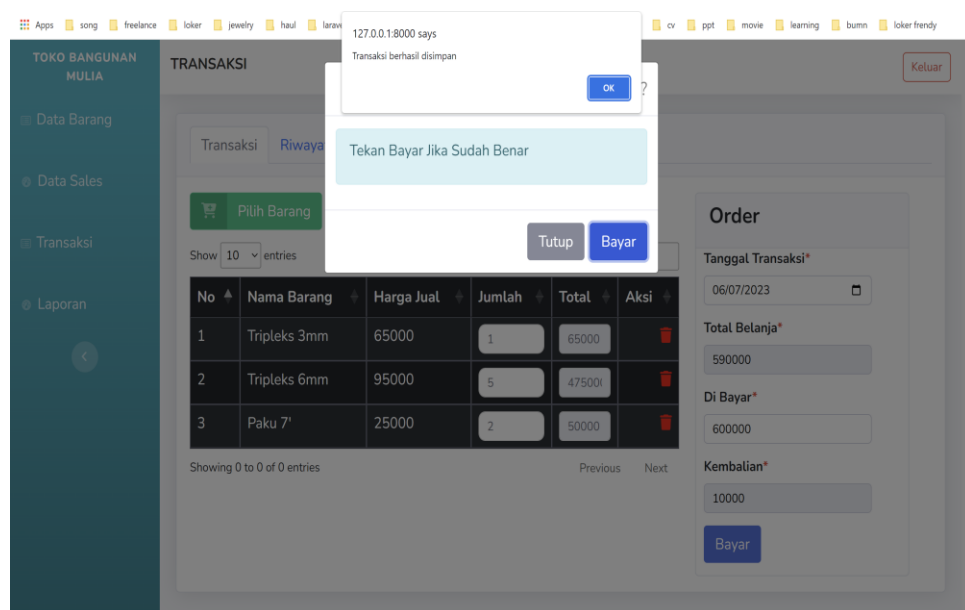


Gambar 4.48 Pengujian Halaman Transaksi Setelah Memilih Barang

12	Melakukan bayar pada barang yang ingin dibeli.	Memilih tanggal transaksi dan mengisi uang yang dikasih oleh pembeli kemudian tekan tombol bayar	Pengguna dapat memilih tanggal, mengisi jumlah uang, dan dapat menekan tombol bayar.	Pengguna berhasil memilih tanggal, mengisi jumlah uang, dan dapat menekan tombol bayar.
----	--	--	--	---



Gambar 4.49 Pengujian Halaman Transaksi Pada Saat Mengisi Tanggal Dan Jumlah Uang



Gambar 4.50 Pengujian Halaman Transaksi Setelah Menekan Tombol Bayar

13	Melakukan proses mencetak struk.	Membuka <i>tab</i> riwayat transaksi kemudian menekan tombol cetak.	Aplikasi dapat mencetak struk.	Aplikasi berhasil mencetak struk.
----	----------------------------------	---	--------------------------------	-----------------------------------

No	Tanggal Transaksi	Jam Transaksi	Total Transaksi	Aksi
1	06-07-2023	00:48:24	590000	Cetak
2	02-06-2023	22:27:26	795000	Cetak
3	01-06-2023	02:11:33	25000	Cetak
4	31-05-2023	01:41:59	65000	Cetak
5	01-06-2023	01:38:17	65000	Cetak
6	01-06-2023	01:29:19	65000	Cetak
7	01-06-2023	00:39:26	1240000	Cetak

Gambar 4.51 Pengujian Halaman Transaksi Bagian Riwayat Transaksi

Nama Barang	Harga	Qty	Total Harga
Tripleks 3mm	Rp. 65000	1	Rp. 65000
Tripleks 6mm	Rp. 95000	5	Rp. 475000
Paku 7'	Rp. 25000	2	Rp. 50000

Total: Rp. 590000
Tunai: Rp. 600000
Kembalian: Rp. 10000

Terima kasih telah berbelanja!
Telp/sms: 0853 4857 3948

Gambar 4.52 Halaman Cetak Struk

14	Menampilkan halaman Riwayat barang terjual.	Aplikasi menampilkan seluruh barang yang terjual.	Aplikasi dapat menampilkan seluruh barang yang terjual.	Aplikasi berhasil menampilkan seluruh barang yang terjual.
----	---	---	---	--

No	Tanggal	Jam Transaksi	Nama Barang	Terjual	Harga Beli	Harga Jual	Untung
1	01-06-2023	00:39:26	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
2	01-06-2023	00:39:26	Tripleks 6mm	5	Rp. 85000	Rp. 95000	Rp. 50000
3	01-06-2023	00:39:26	Cat NoDrop 25KG	1	Rp. 585000	Rp. 700000	Rp. 115000
4	01-06-2023	01:29:19	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
5	01-06-2023	01:38:17	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
6	31-05-2023	01:41:59	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
7	01-06-2023	02:11:33	Paku 7'	1	Rp. 23500	Rp. 25000	Rp. 1500

Gambar 4.53 Pengujian Halaman Laporan

15	Mencari keuntungan penjualan berdasarkan harian.	Pengguna memilih tanggal dan aplikasi dapat menampilkan total keuntungan.	Pengguna dapat memilih tanggal dan aplikasi dapat menampilkan total keuntungan.	Pengguna berhasil memilih tanggal dan aplikasi berhasil menampilkan total keuntungan.
----	--	---	---	---

No	Tanggal	Jam Transaksi	Nama Barang	Terjual	Harga Beli	Harga Jual	Untung
1	06-07-2023	00:48:24	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
2	06-07-2023	00:48:24	Tripleks 6mm	5	Rp. 85000	Rp. 95000	Rp. 50000
3	06-07-2023	00:48:24	Paku 7'	2	Rp. 23500	Rp. 25000	Rp. 3000

Gambar 4.54 Pengujian Pencarian Keuntungan Berdasarkan Harian

16	Mencari keuntungan penjualan	Pengguna memilih bulan dan	Pengguna dapat memilih bulan dan aplikasi	Pengguna berhasil memilih bulan
----	------------------------------	----------------------------	---	---------------------------------

	berdasarkan bulan.	aplikasi dapat menampilkan total keuntungan.	dapat menampilkan total keuntungan.	dan aplikasi berhasil menampilkan total keuntungan.
--	--------------------	--	-------------------------------------	---

Laporan

Cari Untung Harian:

dd/mm/yyyy

Untung Harian

Rp. 0

Cari Untung Bulanan:

Untung Bulanan

Rp. 321500

Cari Untung Tahunan:

YYYY

Untung Tahunan

Rp. 394500

No	Tanggal	Jam Transaksi	Nama Barang	Terjual	Harga Beli	Harga Jual	Untung
1	01-06-2023	00:39:26	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
2	01-06-2023	00:39:26	Tripleks 6mm	5	Rp. 85000	Rp. 95000	Rp. 50000
3	01-06-2023	00:39:26	Cat NoDrop 25KG	1	Rp. 585000	Rp. 700000	Rp. 115000
4	01-06-2023	01:29:19	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
5	01-06-2023	01:38:17	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
6	01-06-2023	02:11:33	Paku 7'	1	Rp. 23500	Rp. 25000	Rp. 1500
7	02-06-2023	22:27:26	Cat NoDrop 25KG	1	Rp. 585000	Rp. 700000	Rp. 115000

Gambar 4.55 Pengujian Pencarian Keuntungan Berdasarkan Bulanan

17	Mencari keuntungan penjualan berdasarkan tahunan.	Pengguna mengisi tahun dan aplikasi dapat menampilkan total keuntungan.	Pengguna dapat mengisi tahun dan aplikasi dapat menampilkan total keuntungan.	Pengguna berhasil mengisi tahun dan aplikasi berhasil menampilkan total keuntungan.
----	---	---	---	---

Laporan

Cari Untung Harian:

dd/mm/yyyy

Untung Harian

Rp. 0

Cari Untung Bulanan:

Untung Bulanan

Rp. 63000

Cari Untung Tahunan:

yyyy

Untung Tahunan

Rp. 394500

No	Tanggal	Jam Transaksi	Nama Barang	Terjual	Harga Beli	Harga Jual	Untung
1	01-06-2023	00:39:26	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
2	01-06-2023	00:39:26	Tripleks 6mm	5	Rp. 85000	Rp. 95000	Rp. 50000
3	01-06-2023	00:39:26	Cat NoDrop 25KG	1	Rp. 585000	Rp. 700000	Rp. 115000
4	01-06-2023	01:29:19	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
5	01-06-2023	01:38:17	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
6	31-05-2023	01:41:59	Tripleks 3mm	1	Rp. 55000	Rp. 65000	Rp. 10000
7	01-06-2023	02:11:33	Paku 7'	1	Rp. 23500	Rp. 25000	Rp. 1500

Gambar 4.56 Pengujian Pencarian Keuntungan Berdasarkan Tahunan

18	Menampilkan dan mengubah pertanyaan & jawaban halaman ganti pertanyaan.	Aplikasi menampilkan halaman ganti pertanyaan dan dapat mengubah pertanyaan dan jawaban.	Aplikasi dapat menampilkan mengubah pertanyaan serta jawaban.	Aplikasi berhasil menampilkan dan mengubah pertanyaan pada halaman ganti pertanyaan.
----	---	--	---	--

GANTI PERTANYAAN

Ganti Pertanyaan

Pertanyaan Sekarang

Nama Bintang Peliharaan Pertama

Pertanyaan Baru

Kota paling disukai?

Jawaban Baru

Manado

Gambar 4.57 Pengujian Halaman Ganti Pertanyaan

Gambar 4.58 Pengujian Berhasil Ganti Pertanyaan

19	Menampilkan dan mengubah username dan kata sandi akun kasir pada halaman akun kasir.	Aplikasi menampilkan halaman ganti akun kasir dan dapat mengubah username dan kata sandi.	Aplikasi dapat menampilkan dan mengubah username dan kata sandi.	Aplikasi berhasil menampilkan dan mengubah username dan kata sandi.
----	--	---	--	---

Gambar 4.59 Pengujian Halaman Ganti Akun Kasir

Gambar 4.60 Pengujian Berhasil Ganti *username*/kata sandi

E. Hasil Pengujian

Setelah melakukan pengujian terhadap seluruh fitur yang ada pada aplikasi, maka dapat disimpulkan bahwa:

1. Semua fitur dan fungsi pada aplikasi dapat berjalan dengan baik.
2. Antarmuka aplikasi dapat berjalan dengan baik dan sesuai dengan semestinya.
3. Aplikasi yang dibangun sudah memenuhi persyaratan fungsional dan kebutuhan pengguna.

F. Kesimpulan *User Acceptance Testing*

Hasil dari UAT yang dilakukan oleh pemilik toko, Seperti pada Lampiran B dapat diambil kesimpulan bahwa tiap fitur yang ada dapat berjalan dengan baik dan sesuai dengan kebutuhan dari pengguna.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan pembahasan mengenai aplikasi yang dibuat maka dapat diambil kesimpulan berupa:

1. Aplikasi telah memenuhi tuntutan pemilik toko mulia dan telah memenuhi kebutuhan yang ada.
2. aplikasi yang dibangun dapat menjalankan fungsi-fungsi yang ada dengan baik sehingga dapat membantu Toko Mulia dalam proses transaksi jual beli, pengontrolan *stock* barang, pembuatan nota, dan laporan atas transaksi transaksi yang terjadi.

5.2 Saran

Saran yang dapat diberikan untuk pengembangan selanjutnya adalah:

1. Diharapkan bagi pengembang selanjutnya untuk dapat menambahkan fitur *barcode scanner*.
2. Menambahkan fitur penggajian karyawan.

DAFTAR PUSTAKA

- [1] Y. Tumiwa, Interviewee, *Wawancara SOP Transaksi di Toko Mulia*. [Interview]. April 2023.
- [2] Y. Tumiwa, Interviewee, *Wawancara Mengenai Toko Mulia*. [Interview]. April 2023.
- [3] A. Kadir, *MEMBUAT APLIKASI WEB DENGAN PHP DAN DATABASE MYSQL (Edisi Revisi)*, 1 ed., M. kika, Ed., Yogyakarta: Penerbit Andi, 2023, p. 1.
- [4] A. Saputra, *Buku Sakti HTML, CSS & Javascript*, Anak Hebat Indonesia, 2019.
- [5] E. ., H. Y. Cornelius De, "Rancang Bangun Perangkat Lunak Point Of Sales Berbasis Web CV. Sumber Baru Di Ketapang," *ENTER*, vol. 1, pp. 417-428, 2018.
- [6] H. Irsyad, *Aplikasi Android dalam 5 Menit*, Revisi ed., Jakarta: Elex Media Komputindo, 2016.
- [7] Y. Y and H. A. Prasetyo, *Panduan Mudah Belajar Framework Laravel*, Jakarta: Elex Media Komputindo, 2018.
- [8] R. Abdulloh, *Membuat Aplikasi Point Of Sale Dengan Laravel dan Ajax*, Jakarta: Elex Media Komputindo, 2017.
- [9] M. MF, *Buku Sakti Pemrograman Web Seri PHP*, S. Adams, Ed., Yogyakarta: Anak Hebat Indonesia, 2018.
- [10] F. S, A. F, S. S, A. E and D. G, *Fundamental Basis Data*, R. R. Rerung, Ed., Bandung: Media Sains Indonesia, 2020.
- [11] R. Fitri, *Pemrograman Basis Data Menggunakan MySQL*, R. Fauzan, Ed., Banjarmasin: Poliban Press, 2020.
- [12] A. S. Roni Habibi, *Aplikasi Prediksi Jumlah Kebutuhan Perusahaan*, Bandung: Kreatif Industri Nusantara, 2020.
- [13] A. A, I. S, D. K and Riyanto, *Panduan Pemilu Desa Berbasis Website (Teknologi Sistem Cerdas Dan Implementasi Di Masyarakat)*, R. Supriyanti, Ed., Yogyakarta: Deepublish, 2019.
- [14] S. Mulyani, *Analisis dan Perancangan Sistem Informasi Manajemen Keuangan Daerah*, edisi ke 2 ed., Bandung: Abdi Sistematika, 2017.
- [15] R. F. Sari and A. U. S, *REKAYASA PERANGKAT LUNAK BERORIENTASI OBJEK MENGGUNAKAN PHP*, A. A. Christian, Ed., Yogyakarta: Penerbit Andi, 2021.
- [16] M. S., M. S., C. H. and G. Kappel, *UML @ Classroom An Introduction to Object-Oriented Modeling*, I. Mackie, Ed., Heidelberg: Springer International Publishing AG., 2015.
- [17] R. Habibi and R. Aprilian, *Tutorial dan penjelasan aplikasi e-office berbasis web menggunakan metode RAD*, R. Habibi, Ed., Bandung: Kreatif Industri Nusantara, 2020.

- [18] Subianto, "Penerapan Metode Rapid Application Development dalam Perancangan Sistem Informasi Pendataan," *INFOKAM*, vol. XVI, p. 46, MARET 2020.

LAMPIRAN A

TRANSKRIP WAWANCARA

Pertanyaan yang telah ditanyakan di Toko Mulia.

1. Apa visi toko mulia?

Jawaban: Toko Mulia tentunya memiliki visi, yaitu “Menjadi toko bahan bangunan terlengkap dan terpercaya sehingga dapat menjadi pilihan utama bagi para konsumen dalam mencari keperluan bahan bangunan”.

2. Apa misi toko mulia?

Jawaban: Toko Mulia memiliki misi berupa:

- a. Memberikan pelayanan berkualitas tinggi kepada para pembeli.
- b. Memberikan kepuasan dan kenyamanan sehingga pembeli memiliki pengalaman belanja yang menyenangkan.
- c. Menjadi tempat berbelanja bahan bangunan yang lengkap dengan harga terjangkau.

3. Jelaskan alur/sop dari transaksi yang dilakukan oleh kasir?

Jawaban: setelah pembeli yang datang dilayani oleh bagian toko maka barang-barang tersebut di cek oleh bagian kasir. Kemudian kasir akan memberitahukan harga yang harus dibayar untuk barang-barang

4. Apa tugas dan fungsi dari kasir?

Jawaban: Kasir bertanggung jawab melayani sales yang datang membawa produk dan *customer* yang ingin melakukan transaksi.

5. apa saja barang-barang yang dijual di Toko Mulia?

Jawaban: Toko Mulia menjual bahan bahan bangunan seperti cat,kuas,tehel,semen,paku Dll.

6. Apakah pemilik menghafal seluruh harga barang yang ada?

Jawaban: dengan sangat banyak jenis yang ada maka sangat sulit menghafal seluruh harga barang yang ada. Oleh karena itu pemilik memiliki buku yang bertuliskan harga-harga barang yang ada.

7. Apakah ada kesulitan yang dialami dengan menulis harga-harga tersebut pada buku?

Jawaban : banyak mulai dari saat harga barang berubah maka harus dicoret dan ditulis lagi harga baru, buku terkena air dan banyak lagi.

8. Apakah pemilik mengingat semua *sales* yang ada?

Jawaban: pemilik mengingat sebagian besar dari mereka akan tetapi biasanya ada sales baru yang datang ataupun terjadi pergantian sales. Hal ini mengakibatkan pemilik mengalami kesulitan dalam mengingat nama maupun nomor dari sales tersebut.

9. Apakah ada masalah yang terjadi saat pembuatan nota?

Jawaban : masalah dalam pembuatan nota biasanya terjadi saat pembeli mengubah barang yang dibeli entah jumlah atau jenis barang yang diambil. Ini mengakibatkan nota menjadi penuh coretan. Selain itu saat pembeli lama dalam memilih barang yang diambil maka nota akan lama selesai.

10. Siapa saja yang memiliki akses kasir?

Jawaban: Hanya pemilik.

11. Apakah ada rekapan penjualan?

Jawaban: tidak ada.

12. apakah dibutuhkan aplikasi yang dapat membantu mempermudah masalah yang terjadi dalam toko?

Jawaban: iya. Karena dengan berkembangnya toko maka mulai ada kesulitan melayani pembeli dengan jumlah banyak dengan cara yang ada sekarang.

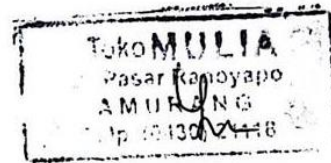
13. siapakah yang dapat mengakses aplikasi itu?

Jawaban: hanya pemilik saja.

14. Apakah aplikasi memerlukan fitur *barcode scanner*?

Jawaban: untuk saat ini pemilik merasa fitur tersebut tidak diperlukan selain itu akan cukup sulit menerapkannya dengan keadaan toko saat ini.

Mengetahui,



Yan Tumiwa

Pemilik Toko

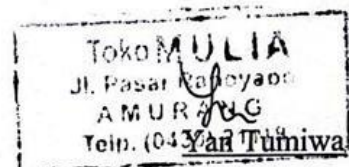
LAMPIRAN B

USER ACCEPTANCE TESTING

Berikut ini merupakan hasil dari pengujian pada user yang akan menggunakan Aplikasi *Point of Sale* Berbasis Web yang akan dilakukan oleh pemilik toko.

No	Pertanyaan	Jawaban	
		Ya	Tidak
1	Apakah aplikasi sudah sesuai dengan kebutuhan di toko?	✓	
2	Apakah masalah yang ada di toko dapat terselesaikan dengan baik?	✓	
3	Apakah aplikasi ini dapat digunakan dengan baik?	✓	
4	Apakah pengguna dapat membuat struk?	✓	
5	Apakah pengguna dapat melihat serta mencari data seperti data barang atau transaksi?	✓	
6	Apakah fitur transaksi yang diberikan sesuai dengan keperluan dan dapat berjalan dengan baik?	✓	
7	Apakah fitur laporan sudah berfungsi dengan baik?	✓	
Saran: —			

Mengetahui,



Pemilik Toko

Berikut ini merupakan hasil dari pengujian pada user yang akan menggunakan Aplikasi *Point of Sale* Berbasis Web yang akan dilakukan oleh pemilik toko.

No	Pertanyaan	Jawaban	
		Ya	Tidak
1	Apakah aplikasi sudah sesuai dengan kebutuhan di toko?	✓	
2	Apakah masalah yang ada di toko dapat terselesaikan dengan baik?	✓	
3	Apakah aplikasi ini dapat digunakan dengan baik?	✓	
4	Apakah pengguna dapat membuat struk?	✓	
5	Apakah fitur transaksi yang diberikan sesuai dengan keperluan dan dapat berjalan dengan baik?	✓	
Saran: —			

Mengetahui,



Pemilik Toko