

**SISTEM PELAYANAN MANDIRI UNTUK TOKO RITEL DI KOTA
MALANG SEBAGAI SOLUSI PERMASALAHAN
PERBELANJAAN BERBASIS WEB MENGGUNAKAN METODE
PROTOTYPING (STUDI KASUS: PT. PETROLUX ARYA
MANDALA)**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Yonathan Fanuel Mulyadi
NIM: 205150207111022



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2023

PENGESAHAN

**SISTEM PELAYANAN MANDIRI UNTUK TOKO RITEL DI KOTA MALANG SEBAGAI
SOLUSI PERMASALAHAN PERBELANJAAN BERBASIS WEB MENGGUNAKAN
METODE PROTOTYPING (STUDI KASUS: PT. PETROLUX ARYA MANDALA)**

SKRIPSI

**Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer**

**Disusun Oleh :
Yonathan Fanuel Mulyadi
NIM: 205150207111022**

**Skripsi ini telah diuji dan dinyatakan lulus pada
x Desember 2023
Telah diperiksa dan disetujui oleh:**

Dosen Pembimbing I

Dosen Pembimbing 2

**Ir. Adam Hendra Brata, S.Kom., M.T., M.Sc.
NIK: 201607 900105 1 001**

**Reza Andria Siregar, S.T., M.Kom.
NIK: 19790621 200604 1 003**

**Mengetahui
Ketua Jurusan Teknik Informatika**

**Achmad Basuki, S.T., M.MG., Ph.D.
NIK: 19741118 200312 1 002**

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar referensi.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, x November 2023



Yonathan Fanuel Mulyadi

NIM: 205150207111022

PRAKATA

Segala pujian, hormat dan kemuliaan bagi Tuhan Yesus Kristus yang telah memberikan kekuatan, hikmat, dan kesehatan, sehingga penulis dapat menyelesaikan penyusunan skripsi dengan judul SISTEM PELAYANAN MANDIRI UNTUK TOKO RITEL DI KOTA MALANG SEBAGAI SOLUSI PERMASALAHAN PERBELANJAAN BERBASIS WEB MENGGUNAKAN METODE PROTOTYPING (STUDI KASUS: PT. PETROLUX ARYA MANDALA) dan masa studi di Fakultas Ilmu Komputer Universitas Brawijaya. Adapun, penulis memiliki harapan agar skripsi ini dapat menjadi berkat dan manfaat bagi sesama. Bersama dengan itu, penulis menyampaikan ucapan terima kasih kepada:

1. Bapak Wayan Firdaus Mahmudi, S.Si, M.T, Ph.D selaku Dekan Fakultas Ilmu Komputer, Bapak Achmad Basuki, S.T., M.MG., Ph.D selaku Ketua Jurusan Teknik Informatika dan Bapak Adhitya Bhawiyuga, S.Kom., M.Sc selaku Ketua Program Studi Teknik Informatika.
2. Bapak Ir. Adam Hendra Brata, S.Kom., M.T., M.Sc dan Bapak Reza Andria Siregar, S.T., M.Kom, selaku pembimbing skripsi I dan pembimbing skripsi II yang telah memberikan bimbingan kepada penulis dalam menyelesaikan skripsi ini.
3. Seluruh Dosen Fakultas Ilmu Komputer Universitas Brawijaya yang telah mendidik, mengajar, dan menegur penulis selama masa studi.
4. Papi dan Mami tercinta, koko, dan seluruh keluarga yang tanpa henti – hentinya memberikan doa, semangat, dan bimbingan, serta bantuan.
5. Zefania Virgin Andini Sanggor yang selalu mendampingi, mendoakan, memberikan semangat, dan memotivasi penulis dalam menyelesaikan skripsi ini.
6. Seluruh sahabat Boy, Salsa, Fafa, Afif, dan semua teman – teman FILKOM yang telah memberikan dukungan dalam pengerjaan skripsi ini.

Penulis menyadari bahwa, masih terdapat kekurangan dalam penyusunan skripsi ini, baik dalam melakukan pembahasan maupun menyajikan materi. Maka dari itu, penulis mengharapkan kritik dan saran untuk penyempurnaan laporan skripsi ini. Kiranya laporan ini bermanfaat bagi semua pihak yang membutuhkan, Tuhan Yesus memberkati.

Malang, x November 2023

Penulis

Fanomulyadi@gmail.com

ABSTRAK

Yonathan Fanuel Mulyadi, Sistem Pelayanan Mandiri Untuk Toko Ritel Di Kota Malang Sebagai Solusi Permasalahan Perbelanjaan Berbasis Web Menggunakan Metode *Prototyping* (Studi Kasus: PT. Petrolux Arya Mandala)

Pembimbing: Ir. Adam Hendra Brata, S.Kom., M.T., M.Sc. dan Reza Andria Siregar, S.T., M.Kom.

Berbelanja merupakan kegiatan yang sering dilakukan oleh masyarakat dalam kehidupan sehari – hari, yang mana hal ini bertujuan dalam memenuhi kebutuhan hidupnya. Adapun kegiatan berbelanja dapat dilakukan di banyak tempat seperti *supermarket*, *minimarket*, toko ritel, warung, dan lainnya. Salah satu destinasi perbelanjaan bagi masyarakat di Kota Malang merupakan Indomaret, yaitu toko ritel yang menyediakan berbagai macam jenis barang – barang kebutuhan sehari – hari, mulai dari makanan, minuman, bahan – bahan dapur, peralatan rumah tangga, dan lainnya. Akan tetapi, dengan sistem manual yang masih digunakan saat ini masih dinilai kurang efektif, karena dengan sistem yang ada, proses berbelanja menjadi lebih lama. Hal ini dapat dilihat, ketika terjadi *over capacity* pelanggan di dalam toko, maka terdapat antrian yang panjang, sebagai akibat dari sistem manual yang masih dipakai, akibatnya, pelanggan yang melakukan perbelanjaan di dalam toko menjadi tidak nyaman, dan pengalaman berbelanja di Indomaret menjadi tidak baik. Maka dari itu, diperlukan suatu sistem yang mampu memberikan solusi dalam mengatasi permasalahan tersebut, sehingga pengalaman pelanggan dalam berbelanja dapat meningkat, dan antrian didalam toko dapat teratasi. Dalam hal ini, dilakukan penelitian terhadap pengembangan, sistem pelayanan mandiri untuk toko ritel di Kota Malang sebagai solusi permasalahan perbelanjaan berbasis web menggunakan metode *prototyping*. Dari penelitian ini diperoleh **x** aktor, **x** kebutuhan fungsional, dan **x** kebutuhan non-fungsional, yang mana sistem ini telah melalui proses pengujian unit, pengujian validasi, dan pengujian kompatibilitas. Pada proses pengujian unit, dilakukan pengujian pada **x** *method*, sedangkan pada proses pengujian validasi, dilakukan pengujian terhadap seluruh kebutuhan fungsional, dan memperoleh hasil valid, selain itu, pada pengujian kompatibilitas, diperoleh hasil bahwa sistem dapat berjalan dengan baik pada **x** *browser* yang berbeda.

Kata kunci: berbelanja, sistem *self-service*, indomaret, toko ritel, metode *prototyping*.

ABSTRACT

Yonathan Fanuel Mulyadi, Self-Service System for Retail Stores in Malang City as a Web-Based Shopping Solution Using the Prototyping Method (Case Study: PT. Petrolux Arya Mandala)

Supervisors: Ir. Adam Hendra Brata, S.Kom., M.T., M.Sc. dan Reza Andria Siregar, S.T., M.Kom.

Shopping is an activity commonly undertaken by the general society in their daily lives, with the aim to fulfil their basic needs. Shopping can take place in various locations, such as supermarkets, convenience stores, retail shops, and other establishments. One notable shopping destination for the residents of Malang City is Indomaret, which is a retail store that offers a wide range of everyday necessities, including food, beverages, kitchen supplies, household items, and more. However, the current manual system in use is considered to be less effective, as it leads to prolonged shopping processes. It becomes a trouble, when the store is overly capacitated by customers, resulting in long queues. This inconvenience is provoked by the continued use of the manual system, thereby tarnishing the shopping experience at Indomaret. Therefore, it needs a system that could solve these issues, ultimately enhancing the shopping experience for customers and alleviating in-store queues. In this regard, research has been conducted for the development of a self-service system for retail stores in Malang City as a web-based solution for shopping issues using the prototyping method. Through this research, **x** actors, **x** functional requirements, and **x** non-functional requirements have been identified. The system has undergone unit testing, validation testing, and compatibility testing. In the unit testing phase, evaluations have been performed on **x** methods, while validation testing encompassed a comprehensive assessment of all functional requirements, yielding valid results. Furthermore, compatibility testing has confirmed that the system operates effectively across **x** different web browsers.

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
PRAKATA.....	iv
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xiv
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	3
1.5 Batasan Masalah	3
1.6 Sistematika Pembahasan	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Kegiatan Belanja di Indomaret (PT Petrolux Arya Mandala)	6
2.2.1 Transaksi Belanja Pelanggan	6
2.3 Rekayasa Perangkat Lunak	7
2.3.1 Model Pengembangan Perangkat Lunak	7
2.3.2 Pendekatan Berorientasi Objek	10
2.4 Teknologi Pengembangan Sistem.....	16
2.4.1 <i>Laravel</i>	16
2.4.2 PHP	16
2.4.3 HTML	16
2.4.4 CSS.....	17
2.4.5 Javascript.....	17
2.4.6 <i>Bootstrap</i>	17

2.4.7 MySQL	17
2.5 Pengujian Perangkat Lunak.....	18
BAB 3 METODOLOGI PENELITIAN	19
3.1 Studi Literatur	19
3.2 Analisis Kebutuhan Sistem.....	20
3.3 Perancangan Sistem.....	20
3.4 Implementasi Sistem	20
3.5 Pengujian Sistem.....	21
3.6 Kesimpulan dan Saran	21
BAB 4 REKAYASA KEBUTUHAN.....	22
4.1 Deskripsi Umum Sistem	22
4.2 Elisitasi Kebutuhan (tambahkan referensi lampiran).....	22
4.3 Identifikasi Aktor.....	23
4.4 Spesifikasi Kebutuhan	24
4.5 Pemodelan Kebutuhan Sistem.....	38
4.5.1 <i>Use Case Diagram</i>	38
4.5.2 <i>Use Case Scenario</i>	39
BAB 5 PEMBAHASAN.....	57
5.1 Perancangan	57
5.1.1 Perancangan Arsitektur.....	57
5.1.2 <i>Sequence Diagram</i>	58
5.1.3 <i>Class Diagram</i>	60
5.1.4 Perancangan <i>Database</i>	62
5.1.5 Perancangan Antarmuka.....	67
5.1.6 Perancangan Komponen	96
5.2 Implementasi	96
5.2.1 Spesifikasi Sistem	96
5.2.2 Implementasi Antarmuka	97
5.2.3 Subbab Lima Dua Dua	97
5.3 Subbab Lima Tiga	97
5.3.1 Contoh Struktur Penelitian Implementatif Pembangunan	97

5.3.2 Contoh Struktur Penelitian Nonimplementatif Eksperimental.....	98
BAB 6 Penutup	100
6.1 Kesimpulan.....	100
6.2 Saran	100
DAFTAR REFERENSI	101
LAMPIRAN A HASIL WAWANCARA.....	103

DAFTAR TABEL

Tabel 2.1 Kajian Pustaka	5
Tabel 2.2 Hubungan Dalam <i>Class Diagram</i>	11
Tabel 2.3 Hubungan Dalam <i>Use Case Diagram</i>	12
Tabel 2.4 Hubungan Dalam <i>Sequence Diagram</i>	14
Tabel 4.1 Kebutuhan Sistem Hasil Wawancara dengan <i>Stakeholder</i>	22
Tabel 4.2 Identifikasi Aktor	23
Tabel 4.3 Kebutuhan Fungsional	24
Tabel 4.4 Kebutuhan Non-Fungsional	37
Tabel 4.5 Tabel <i>Use Case Scenario Register</i>	39
Tabel 4.6 Tabel <i>Use Case Scenario Login</i>	41
Tabel 4.7 Tabel <i>Use Case Scenario</i> Melakukan Belanja Mandiri	41
Tabel 4.8 Tabel <i>Use Case Scenario</i> Menampilkan Detil Barang Belanja	42
Tabel 4.9 Tabel <i>Use Case Scenario</i> Menampilkan Kategori Barang Belanja	43
Tabel 4.10 Tabel <i>Use Case Scenario</i> Menampilkan Keranjang Belanja	43
Tabel 4.11 Tabel <i>Use Case Scenario</i> Melakukan Pembayaran Barang Belanja Mandiri	44
Tabel 4.12 Tabel <i>Use Case Scenario</i> Menampilkan Riwayat Belanja	44
Tabel 4.13 Tabel <i>Use Case Scenario</i> Menambah Pelaporan Kegiatan Kriminalitas	45
Tabel 4.14 Tabel <i>Use Case Scenario</i> Melakukan Belanja Dengan Bantuan Karyawan	46
Tabel 4.15 Tabel <i>Use Case Scenario</i> Melakukan Pembayaran Barang Belanja dengan Bantuan Karyawan	47
Tabel 4.16 Tabel <i>Use Case Scenario</i> Menampilkan Kategori Barang Belanja dengan Bantuan Karyawan	48
Tabel 4.17 Tabel <i>Use Case Scenario</i> Menampilkan Detil Barang Belanja dengan Bantuan Karyawan	48
Tabel 4.18 Tabel <i>Use Case Scenario</i> Menampilkan Keranjang Belanja dengan Bantuan Karyawan	49
Tabel 4.19 Tabel <i>Use Case Scenario</i> Menambah Data Barang	49
Tabel 4.20 Tabel <i>Use Case Scenario</i> Mengubah Data Barang	50
Tabel 4.21 Tabel <i>Use Case Scenario</i> Menghapus Data Barang	51

Tabel 4.22 Tabel <i>Use Case Scenario</i> Menampilkan Daftar Data Barang.....	52
Tabel 4.23 Tabel <i>Use Case Scenario</i> Menampilkan Daftar Laporan Tindakan Kriminalitas.....	52
Tabel 4.24 Tabel <i>Use Case Scenario</i> Mengubah Data Laporan Tindakan Kriminalitas	53
Tabel 4.25 Tabel <i>Use Case Scenario</i> Menghapus Data Laporan Tindakan Kriminalitas.....	53
Tabel 4.26 Tabel <i>Use Case Scenario</i> Menampilkan Daftar Data Pelanggan	54
Tabel 4.27 Tabel <i>Use Case Scenario</i> Mengubah Data Pelanggan	54
Tabel 4.28 Tabel <i>Use Case Scenario</i> Menghapus Data Pelanggan.....	55
Tabel 5.1 Tabel Detil Entitas <i>General Manager</i> Operasional.....	64
Tabel 5.2 Tabel Detil Entitas Karyawan.....	64
Tabel 5.3 Tabel Detil Entitas Pelanggan	64
Tabel 5.4 Tabel Detil Entitas Barang	65
Tabel 5.5 Tabel Detil Belanja Mandiri	65
Tabel 5.6 Tabel Detil Belanja dengan Bantuan Karyawan	66
Tabel 5.7 Tabel Detil Pelaporan Kegiatan Kriminalitas	66
Tabel 5.8 Tabel Detil Riwayat Belanja	66
Tabel 5.9 Tabel Detil Metode Pembayaran	66
Tabel 5.10 Tabel Detil Metode Pembayaran	96
Tabel 5.11 Tabel Detil Metode Pembayaran	97

DAFTAR GAMBAR

Gambar 2.1 Proses Bisnis Transaksi Belanja Saat Ini	7
Gambar 2.2 Gambar Model <i>Prototyping</i>	9
Gambar 3.1 Strategi Penelitian	19
Gambar 4.1 Proses Bisnis <i>As-Is</i>	23
Gambar 4.2 Proses Bisnis <i>To-Be</i>	23
Gambar 4.3 <i>Use Case Diagram</i>	39
Gambar 5.1 Perancangan Arsitektur Sistem Pelayanan Mandiri Untuk Toko Ritel di Kota Malang	57
Gambar 5.2 <i>Sequence Diagram</i> Melakukan Belanja Mandiri	58
Gambar 5.3 <i>Sequence Diagram</i> Menambah Pelaporan Kegiatan Kriminalitas	59
Gambar 5.4 <i>Sequence Diagram</i> Menambah Data Barang	60
Gambar 5.5 <i>Class Diagram</i>	61
Gambar 5.6 <i>Entity Relational Diagram</i>	63
Gambar 5.7 Perancangan Antarmuka <i>Login</i>	67
Gambar 5.8 Perancangan Antarmuka <i>Register</i>	69
Gambar 5.9 Perancangan Antarmuka Melakukan Belanja Mandiri/Melakukan Belanja dengan Bantuan Karyawan	70
Gambar 5.10 Perancangan Antarmuka Menampilkan Riwayat Belanja	71
Gambar 5.11 Perancangan Antarmuka Menambah Pelaporan Kegiatan Kriminalitas	73
Gambar 5.12 Perancangan Antarmuka Menampilkan Detil Barang Belanja/Menampilkan Detil Barang	74
Gambar 5.13 Perancangan Antarmuka Menampilkan Kategori Barang Belanja/Menampilkan Kategori Barang	75
Gambar 5.14 Perancangan Antarmuka Menampilkan Keranjang Belanja	76
Gambar 5.15 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja	77
Gambar 5.16 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja (2)	78
Gambar 5.17 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja (3)	79
Gambar 5.18 Perancangan Antarmuka Menampilkan Keranjang Belanja dengan Bantuan Karyawan	80

Gambar 5.19 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja dengan Bantuan Karyawan	81
Gambar 5.20 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja dengan Bantuan Karyawan (2).....	82
Gambar 5.21 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja dengan Bantuan Karyawan (3).....	83
Gambar 5.22 Perancangan Antarmuka Menambah Data Barang dan Menampilkan Daftar Data Barang.....	84
Gambar 5.23 Perancangan Antarmuka Mengubah Data Barang	86
Gambar 5.24 Perancangan Antarmuka Menghapus Data Barang.....	87
Gambar 5.25 Perancangan Antarmuka Menghapus Data Barang (2)	88
Gambar 5.26 Perancangan Antarmuka Menghapus Data Barang (3)	89
Gambar 5.27 Perancangan Antarmuka Menampilkan Daftar Laporan Tindakan Kriminalitas.....	90
Gambar 5.28 Perancangan Antarmuka Menampilkan Data Laporan Tindakan Kriminalitas.....	91
Gambar 5.29 Perancangan Antarmuka Menampilkan Daftar Data Pelanggan	92
Gambar 5.30 Perancangan Antarmuka Menampilkan Data Pelanggan	93
Gambar 5.31 Perancangan Antarmuka Mengubah Data Pelanggan	94
Gambar 5.32 Perancangan Antarmuka Menampilkan Daftar Pelanggan Belanja dengan Bantuan Karyawan	95

DAFTAR LAMPIRAN

LAMPIRAN A HASIL WAWANCARA.....	103
---------------------------------	-----

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Berbelanja merupakan aktivitas yang melibatkan pertimbangan pembelian suatu produk maupun jasa, mencari toko yang menyediakan produk ataupun jasa yang terbaik, pencarian produk ataupun jasa yang diinginkan di dalam toko tersebut, serta menentukan keputusan untuk membeli (Huddleston dan Minahan, 2011).

Salah satu tempat melakukan kegiatan berbelanja kebutuhan sehari – hari yaitu, Indomaret. Toko ritel ini merupakan tempat berbelanja yang berdiri sejak tahun 1988 oleh PT Indomarco Prismatama, yang menjual berbagai macam kebutuhan sehari – hari masyarakat. Hal inilah yang menjadikan Indomaret sebagai salah satu destinasi berbelanja kebutuhan sehari – hari oleh masyarakat di Kota Malang.

Akan tetapi, pada zaman sekarang ini, kebutuhan manusia semakin meningkat setiap harinya, dengan jenis dan jumlah kebutuhannya yang semakin tidak menentu, terus meningkat, dan berubah – ubah, maka dari itu, masyarakat dituntut untuk melakukan kegiatan berbelanja setiap harinya, sehingga mengakibatkan terhambatnya kegiatan belanja dan transaksi pembayaran di toko Indomaret sebagai salah satu destinasi berbelanja. Melalui wawancara yang dilakukan pada salah satu toko Indomaret di Kota Malang yaitu PT. Petrolux Arya Mandala, bersama narasumber Mas Okky (salah satu karyawan Indomaret PT. Petrolux Arya Mandala), dilakukan pembahasan mengenai kegiatan berbelanja dan sistem berbelanja yang dipakai saat ini. Beliau mengatakan bahwa, kegiatan berbelanja di Indomaret seringkali menimbulkan penumpukan, karena adanya *over capacity* pengunjung, selain itu, sistem berbelanja yang dipakai saat ini masih dioperasikan oleh manusia, sehingga karyawan/pekerja mengalami kesulitan ketika pelanggan akan melakukan pembayaran se usai berbelanja. Pada saat pelanggan hendak melakukan pembayaran, pelanggan perlu mengantri dan menunggu selama beberapa waktu, sebelum akhirnya dapat melakukan pembayaran, hal ini mengakibatkan terjadinya penumpukan pelanggan dan menyebabkan pelanggan lainnya kesulitan dalam berbelanja serta mengurangi kenyamanan pelanggan dalam melakukan perbelanjaan di toko tersebut. Oleh karena itu, diperlukan sistem berbelanja berbasis *web* yang mampu memberikan kemudahan dan kenyamanan bagi pelanggan, ketika berbelanja di Indomaret.

Adanya sistem berbelanja berbasis web menjadi hal yang sangat penting dalam kegiatan berbelanja, khususnya di era sekarang ini, yang mana transformasi digital terus dilakukan setiap harinya. Sistem berbelanja berbasis web diharapkan mampu mempermudah pelanggan yang akan berbelanja di Indomaret, tanpa harus mengantri ataupun mencari barang belanja yang dibutuhkan secara manual, selain itu, pelanggan dapat dengan mudah membayar barang – barang belanjanya melalui sistem berbelanja yang ada, tanpa harus mengantri di kasir. Adanya penggunaan sistem berbelanja berbasis web ini, maka pengalaman berbelanja pelanggan menjadi lebih menyenangkan dan efisien.

Dalam melakukan pengembangan sistem berbelanja berbasis web bagi Indomaret, digunakan metode *Prototyping*, yang mana metode ini merupakan model fisik kerja suatu sistem dan berfungsi sebagai versi awal dari sebuah sistem (Ogedebe dkk, 2018). Metode *Prototyping* ini berfungsi dalam menggambarkan dan melakukan pengujian pada setiap konsep, fitur, serta fungsi produk kepada *stakeholder*, sebelum akhirnya dilakukan pengembangan secara utuh. Penggunaan metode *Prototyping* dalam penelitian ini didasari pada tingkat pemahaman *stakeholder* mengenai kebutuhan sistem yang diperlukan, yang mana kebutuhan yang dispesifikasikan belum jelas, tidak sepenuhnya dipahami, dan sering terjadi perubahan. Selain itu, penggunaan metode ini juga didasari karena kelebihan dari metode *Prototyping* yang dimiliki, jika dibandingkan dengan metode lainnya, yang mana metode ini memungkinkan *stakeholder* dalam melakukan pengawasan dan pengujian produk yang dikembangkan secara langsung, mampu memperoleh *feedback* lebih awal dari *stakeholder* dalam proses pengembangan produk, memiliki tingkat risiko kesalahan pengembangan yang rendah, menghemat waktu dan biaya yang dibutuhkan, meningkatkan kreativitas dari pengembang produk dengan banyaknya ide – ide yang baru.

Design pattern yang digunakan dalam penelitian ini adalah MVC, alasan penggunaan design pattern ini yaitu, adanya pemisahan *role* yang jelas, memungkinkan pengujian awal pada setiap komponen MVC untuk memastikan semua aspek berfungsi dengan baik, adanya keikutsertaan *stakeholder* dalam melakukan perbaikan dengan memberikan *feedback* yang digunakan sebagai dasar dalam melakukan perbaikan desain UI dan logika bisnisnya (model).

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, maka rumusan masalah yang diperoleh yaitu:

1. Bagaimana mengembangkan sistem pelayanan mandiri berbasis *website* sesuai kebutuhan dari *stakeholder*, dengan menggunakan metode *Prototyping*?
2. Bagaimanakah pengembangan sistem pelayanan mandiri sesuai dengan kebutuhan dari *stakeholder*?
3. Bagaimanakah hasil pengembangan sistem pelayanan mandiri yang sesuai dengan kebutuhan dari *stakeholder*?
4. Bagaimanakah hasil implementasi sistem pelayanan mandiri bagi kinerja karyawan Indomaret dan pengalaman berbelanja pelanggan Indomaret?

1.3 Tujuan

1. Melakukan analisis dan penyusunan kebutuhan *stakeholder* untuk melakukan pengembangan sistem pelayanan mandiri.
2. Melakukan pengembangan sistem pelayanan mandiri sesuai kebutuhan *stakeholder*.
3. Melakukan implementasi sistem pelayanan mandiri yang telah dikembangkan.

4. Melakukan pengujian terhadap sistem pelayanan mandiri yang telah dikembangkan terhadap kebutuhan dari *stakeholder*.

1.4 Manfaat

Manfaat yang dapat diperoleh dari penelitian yang dilakukan yaitu:

1. Manfaat Teoritis

Melalui penelitian yang dilakukan, maka hasilnya diharapkan mampu memberikan ilmu/wawasan dalam melakukan pengembangan sistem pelayanan mandiri yang berbasis *website* bagi Indomaret.

2. Manfaat Praktis

- a. Bagi Peneliti

Dari penelitian yang dilakukan oleh peneliti, maka diperoleh manfaat berupa bertambahnya ilmu dan wawasan dalam melakukan pengembangan sistem pelayanan mandiri yang berbasis *website*.

- b. Bagi Karyawan Indomaret

Dengan dilakukannya penelitian ini, maka hasil yang diperoleh dapat meningkatkan produktivitas karyawan Indomaret, dalam memberikan pelayanan yang lebih baik bagi pelanggan – pelanggan Indomaret.

- c. Bagi Pelanggan Indomaret

Melalui penelitian ini, maka hasilnya dapat meningkatkan pengalaman berbelanja pelanggan Indomaret, sehingga kegiatan berbelanja dapat menjadi lebih efisien dan menyenangkan.

1.5 Batasan Masalah

Batasan masalah yang terdapat di dalam penelitian ini yaitu:

1. Objek dari penelitian ini merupakan Indomaret PT. Petrolux Arya Mandala.
2. Pemodelan kebutuhan dilakukan dengan pendekatan berorientasi obyek.
3. Sistem yang dikembangkan berbasis web.
4. Metode pengembangan sistem menggunakan metode *Prototyping*.
5. Bahasa pemrograman yang digunakan adalah HTML, CSS, Javascript, PHP, dan basis data MySQL, serta menggunakan *framework* Laravel.
6. Sistem digunakan oleh *General Manager* Operasional, karyawan, dan pelanggan.

1.6 Sistematika Pembahasan

Dalam penyusunan penelitian ini, peneliti melakukan pembagian menjadi tiga bagian sistematis, yang terdiri dari:

BAB 1 PENDAHULUAN

Bab ini membahas mengenai latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika pembahasan dalam penelitian pengembangan sistem pelayanan mandiri untuk toko ritel di Kota Malang sebagai solusi permasalahan perbelanjaan berbasis web

BAB 2 LANDASAN KEPUSTAKAAN

Bab ini membahas mengenai kajian pustaka dan dasar – dasar teori yang berhubungan dengan penelitian ini, yang mana kajian pustaka berisikan informasi mengenai penelitian – penelitian sebelumnya yang berkaitan dengan penelitian ini. Dasar teori pada bab ini berisikan informasi mengenai domain masalah, rekayasa perangkat lunak, metode pengembangan *Prototyping*, beserta dengan teknologi – teknologi yang digunakan untuk mengembangkan sistem antara lain, HTML, CSS, Javascript, PHP, MySQL, Laravel.

BAB 3 METODOLOGI PENELITIAN

Bab ini membahas mengenai tahapan – tahapan yang dilakukan pada saat penelitian yang terdiri dari studi literature, kebutuhan, perancangan, implementasi, pengujian, dan kesimpulan penelitian sistem pelayanan mandiri berbasis web.

BAB 4 REKAYASA KEBUTUHAN

Bab ini membahas mengenai visualisasi/gambaran dari sistem yang akan dibentuk, dengan melakukan analisis kebutuhan yang dijadikan sebagai acuan dalam merancang dan membangun sistem.

BAB 5 PERANCANGAN DAN IMPLEMENTASI

Bab ini membahas mengenai hasil dari perancangan dan implementasi sistem yang telah dilakukan, berdasarkan pada rekayasa kebutuhan yang telah didefinisikan.

BAB 6 PENGUJIAN

Bab ini membahas mengenai pemeriksaan terhadap fungsionalitas dari sistem, untuk mengetahui apakah sistem yang dibentuk dapat berjalan dengan baik, dan sesuai sebagaimana telah didefinisikan pada rekayasa kebutuhan.

BAB 7 PENUTUP

Bab ini membahas mengenai evaluasi dari sistem yang telah dibentuk, yang mana berisikan kesimpulan dan saran terhadap penelitian yang telah dilakukan.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Terdapat beberapa penelitian yang membahas mengenai sistem *self-service*, salah satunya merupakan penelitian yang dilakukan oleh Febby Anggara Putra, Riswan Septriayadi Sianturi, dan Agi Putra Kharisma (2023), yang mana dalam penelitian ini dilakukan perancangan *user experience* aplikasi *self-checkout* dengan menggunakan metode *Human Centered Design*. Adapun hasil dari penelitian ini yaitu berupa *wireframe*, *mockup*, dan *prototype*, dengan hasil pengujian yang dilakukan melalui Maze dan Google Form yaitu diperolehnya nilai aspek efektivitas sebesar 94,29%, aspek efisiensi 0,200 *goals* per detiknya, dan aspek kepuasan pengguna 85,25 dengan *adjective rating excellent* dan *grade A*.

Selain itu, terdapat juga penelitian yang dilakukan oleh Suwarno dan Rio Riferro Lim (2021) mengenai perancangan sistem kasir layanan mandiri berbasis web pada *supermarket*, yang mana pada penelitian ini menggunakan metode SDLC *waterfall*. Hasil yang diperoleh dari penelitian ini yaitu adalah sistem kasir layanan mandiri berbasis web yang memberikan pengalaman baru bagi pelanggan, dan meningkatkan daya saing *supermarket*.

Tabel 2.1 Kajian Pustaka

No	Judul Penelitian	Objek Penelitian	Perbedaan	
			Penelitian Terdahulu	Rencana Penelitian
1	Perancangan User Experience Aplikasi Self-Checkout di Supermarket menggunakan Metode Human-Centered Design (Febby Anggara Putra, Riswan Septriayadi Sianturi, & Agi Putra Kharisma, 2022)	Pembangunan sistem self-checkout supermarket berbasis web	Belum adanya fitur detil produk, <i>report a crime</i> , dan hanya terdapat dua aktor yaitu pengunjung <i>supermarket</i> dan admin <i>supermarket</i> .	Melakukan perancangan <i>user experience</i> aplikasi <i>self-checkout</i> setiap kali proses bisnis baru berubah, mengadaptasi metode <i>design thinking</i> , mengimplementasikan perancangan <i>user experience</i> aplikasi <i>self-checkout</i> pada <i>supermarket</i> .
2	Perancangan Sistem Kasir	Pembangunan sistem self-	Belum adanya fitur	Mengimplementasikan perancangan sistem

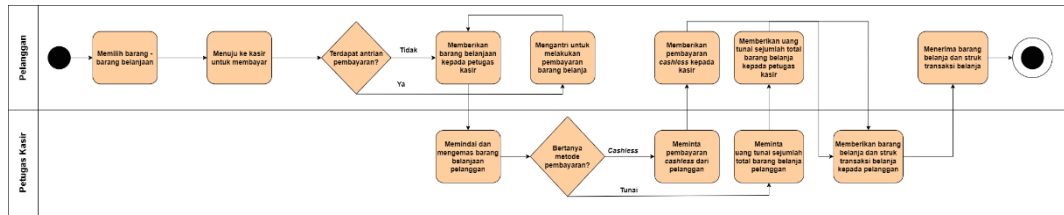
	Layanan Mandiri Berbasis Web Pada Supermarket Bless Dengan Metode Sdlc (Suwarno & Rio Riferro Lim, 2021)	checkout supermarket berbasis web	detil produk, <i>report a crime</i> , dan hanya terdapat dua aktor yaitu admin dan <i>user</i> .	kasir layanan mandiri pada <i>supermarket</i> .
--	--	-----------------------------------	--	---

2.2 Kegiatan Belanja di Indomaret (PT Petrolux Arya Mandala)

Belanja merupakan kegiatan yang dilakukan oleh setiap manusia untuk memenuhi kebutuhannya sehari, yang mana kegiatan ini dilakukan dengan melakukan transaksi jual – beli barang atau jasa antara penjual dengan pelanggan. Belanja juga dapat dikatakan sebagai aktivitas yang melibatkan pertimbangan pembelian suatu produk maupun jasa, mencari toko yang menyediakan produk ataupun jasa yang terbaik, pencarian produk ataupun jasa yang diinginkan di dalam toko tersebut, serta menentukan keputusan untuk membeli (Huddleston dan Minahan, 2011).

2.2.1 Transaksi Belanja Pelanggan

Berdasarkan hasil wawancara yang dilakukan dengan salah satu karyawan Indomaret yaitu Mas Oky, kegiatan belanja di Indomaret dilakukan mulai dari, pelanggan masuk ke dalam Indomaret, kemudian pelanggan memilih barang – barang belanjaan yang dibutuhkan, setelah itu, pelanggan menuju ke kasir untuk melakukan pembayaran barang belanja yang telah dipilih, apabila terdapat antrian pembayaran di kasir, maka pelanggan perlu mengantri terlebih dahulu, jika tidak terdapat antrian, pelanggan dapat langsung memberikan barang belanjaan kepada petugas kasir. Lalu petugas kasir akan memindai dan mengemas barang belanjaan yang telah diberikan pelanggan, setelah itu, petugas kasir akan bertanya kepada pelanggan mengenai metode pembayaran yang akan dipilih, apabila pelanggan memilih *cashless*, maka petugas kasir akan meminta pembayaran secara *cashless* kepada pelanggan, kemudian pelanggan akan memberikan pembayaran secara *cashless* kepada petugas kasir. Akan tetapi, jika pelanggan memilih pembayaran tunai, maka petugas kasir akan meminta uang tunai sejumlah total barang belanja dari pelanggan yang bersangkutan, kemudian pelanggan akan memberikan uang tunai sejumlah total barang belanja kepada petugas kasir. Setelah itu, petugas kasir akan memberikan barang belanja beserta dengan struk transaksi belanja kepada pelanggan, dan terakhir pelanggan akan menerima barang belanja dan struk transaksi belanja dari petugas kasir. Proses transaksi belanja yang dilakukan oleh pelanggan, dapat dilihat pada Gambar 2.1.



Gambar 2.1 Proses Bisnis Transaksi Belanja Saat Ini

2.3 Rekayasa Perangkat Lunak

Rekayasa perangkat lunak adalah suatu pendekatan disiplin dan sistematis untuk mendesain, mengembangkan, menguji, dan melakukan pemeliharaan terhadap sistem perangkat lunak. Adapun aplikasi dari rekayasa perangkat lunak menggunakan konsep rekayasa kepada semua siklus pengembangan perangkat lunak, mulai dari konsep awal dan pengumpulan kebutuhan, hingga peluncuran dan pemeliharaan.

Menurut Roger S, Pressman (2010), rekayasa perangkat lunak merupakan pembuatan dan penggunaan prinsip – prinsip keahlian teknik untuk mendapatkan perangkat lunak yang ekonomis dan handal, serta bekerja secara efisien pada mesin yang sesungguhnya.

Selain itu, menurut Fritz Bauer (1972), rekayasa perangkat lunak merupakan penetapan dan penggunaan prinsip rekayasa yang baik, untuk memperoleh perangkat lunak yang dapat diandalkan dan bekerja pada mesin nyata secara efisien.

Tujuan melakukan rekayasa perangkat lunak adalah merancang, mengembangkan, dan melakukan pemeliharaan sistem perangkat lunak secara sistematis, untuk memenuhi tujuan dan syarat tertentu. Dengan melakukan Rekayasa Perangkat Lunak, maka sistem yang dibangun akan sesuai dengan kebutuhan pengguna dan menyelesaikan masalah di dunia nyata, mampu membentuk perangkat lunak yang bekerja secara efisien, mengontrol biaya pengembangan perangkat lunak supaya lebih murah, memiliki dokumentasi yang baik untuk memahami dan memelihara perangkat lunak.

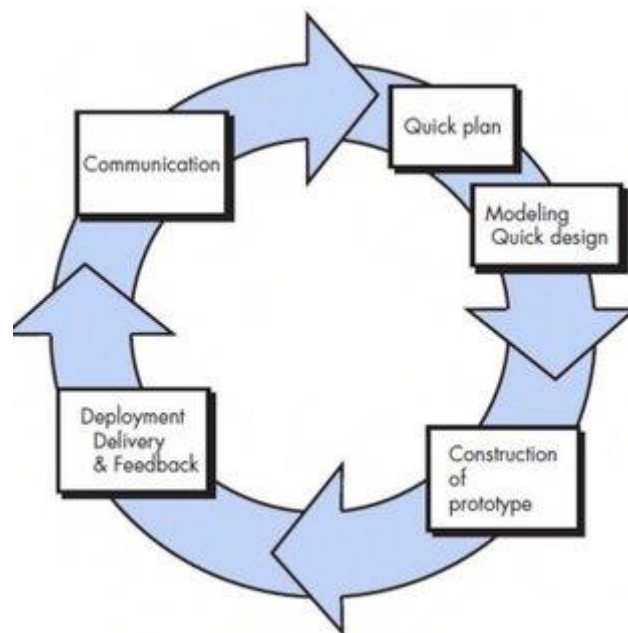
2.3.1 Model Pengembangan Perangkat Lunak

Model Pengembangan Perangkat Lunak atau biasa disebut sebagai *Software Development Life Cycle* (SDLC) merupakan proses yang digunakan oleh pengembang perangkat lunak dalam mendesain, mengembangkan, menguji, dan meluncurkan aplikasi dan perangkat lunak. Menurut Satzinger, Jackson, & Burd (2010, p38), SDLC adalah seluruh proses ruang lingkup sistem yang dimulai pada tahap membangun, menyebarkan, menggunakan, dan memperbarui sistem informasi. Selain itu, menurut Kendall & Kendall (2006), SDLC adalah pendekatan yang dilakukan secara bertahap dalam hal melakukan analisis dan membangun rancangan sistem dengan menggunakan siklus – siklus secara spesifik terhadap kegiatan penggunaannya.

Terdapat beberapa Model Pengembangan Perangkat Lunak antara lain Model *Waterfall*, Model *Spiral*, Model *V-Model*, Model *Prototyping*, Model *Scrum*, Model *Rapid Application Development* (RAD). Model – model pengembangan perangkat lunak tersebut memiliki kelebihan dan kekurangan, serta peruntukan yang berbeda – beda pada setiap modelnya. Pada penelitian ini, model pengembangan yang digunakan adalah model pengembangan *prototyping*, yang mana alasan penggunaannya adalah kebutuhan yang dikemukakan oleh *stakeholder* masih belum jelas dan lengkap. Dengan digunakannya model pengembangan *prototyping*, maka *prototype – prototype* yang dibentuk oleh peneliti mampu membantu *stakeholder* dalam mengeksplorasi kebutuhan yang lebih spesifik dan tepat.

2.3.1.1 Model *Prototyping*

Model *prototyping* merupakan pendekatan pengembangan perangkat lunak, yang mana model/prototipe yang bekerja pada perangkat lunak dibentuk pada siklus awal proyek. Menurut Roger S. Pressman (2002:50), metode *prototyping* adalah metode pengembangan perangkat lunak yang memiliki interaksi antara pengembang dengan pengguna sistem, dengan demikian ketidakserasian antara pengembang maupun pengguna mampu teratasi. Model ini digunakan untuk mengumpulkan kebutuhan dari *stakeholder*, klarifikasi kebutuhan, dan menyempurnakan desain sebelum pengembangan perangkat lunak berskala penuh dimulai. Perlu diperhatikan bahwa, mengatur proses *prototyping* sangat penting untuk menghindari ruang lingkup yang lebih luas, dan melakukan kontrol terhadap jadwal dan anggaran proyek. Adapun model pengembangan *prototyping* dapat dilihat pada gambar 2.2.



Gambar 2.2 Gambar Model *Prototyping*

Sumber: Roger S. Pressman (2012:50)

Tahapan model *prototyping* menurut Roger S. Pressman (2012) yaitu:

1. *Communication* (Komunikasi)

Pada tahapan ini, dilakukan komunikasi antara pengembang dengan *stakeholder*, yang mana tujuan dilakukannya pertemuan ini adalah mengidentifikasi kebutuhan dari *stakeholder* yang nantinya akan digunakan untuk memahami tujuan dan konsep dari perangkat lunak yang akan dikembangkan. Selain itu, melakukan pengumpulan data melalui observasi maupun melalui wawancara dengan *stakeholder*, dengan tujuan yaitu memposisikan diri pengembang sebagai pengguna untuk menemukan kekurangan sistem, dan menerima *feedback* dari *stakeholder* berkaitan dengan masalah yang dialami saat ini. Dalam tahapan ini juga dilakukan analisis proses bisnis yang sedang berlangsung saat ini, sebagai acuan dalam pembuatan perangkat lunak yang akan dikembangkan.

2. *Quick Plan* (Perencanaan)

Tahapan *Quick Plan* merupakan tahapan pembuatan rencana awal untuk melakukan perancangan sistem, yang mana perancangan tersebut bersumber dari proses analisis kebutuhan yang meliputi kebutuhan input, proses, output, dan antarmuka (R. Susanto, 2018). Perlu diketahui bahwa, tahapan ini berisikan hasil dari proses komunikasi, berupa analisis kebutuhan pengguna, beserta dengan format dari keseluruhan sistem yang akan dikembangkan (Pradipta, Prasetyo, & Ambarsari, 2015).

3. *Modeling Quick Design*

Pada tahapan ini dilakukan pemodelan perencanaan dari tahapan yang dilakukan sebelumnya, yang mana pemodelan dilakukan dengan membentuk *Business Processing Model*, yang mana dasar pembentukannya adalah analisis proses bisnis dari hasil wawancara dengan *stakeholder*, berkaitan dengan proses bisnis yang saat ini dilakukan.

4. *Construction of Prototype*

Dalam tahapan ini, pengembang membentuk prototipe yang digunakan untuk merepresentasikan sistem yang akan dikembangkan, dan dijadikan sebagai bentuk awal sistem yang menggambarkan konsep dan rancangan, serta menemukan lebih banyak masalah dan solusinya (Sommerville, 2011). Selain itu, dalam tahapan ini juga dilakukan evaluasi prototipe oleh *stakeholder* yang bertujuan untuk memperoleh ulasan – ulasan berkenaan dengan prototipe yang ada, yang mana ulasan – ulasan tersebut nantinya digunakan oleh pengembang sebagai dasar dalam melakukan perbaikan dan penambahan, apabila dalam prototipe ditemukan kekurangan.

5. *Deployment Delivery and Feedback*

Tahapan ini merupakan tahap terakhir dalam proses model *prototyping*, yang mana dalam tahapan ini, dilakukan pengembangan sistem dengan menggunakan bahasa pemrograman sebagai bentuk realisasi dari prototipe yang sebelumnya sudah dibentuk. Ketika sistem selesai dibangun menjadi *fully-functioned system*, maka selanjutnya adalah evaluasi dari *stakeholder*, yang bertujuan untuk memperoleh ulasan – ulasan dari *stakeholder*, sehingga sistem yang dikembangkan dapat diubah atau diperbaiki, apabila ditemukan kesalahan/kekurangan.

2.3.2 Pendekatan Berorientasi Objek

Pendekatan berorientasi objek adalah metodologi pengembangan perangkat lunak yang berfokus pada mendesain dan mengorganisir perangkat lunak, menggunakan objek dan kelas sebagai dasar pembentukannya. Menurut Sholih (2006), pendekatan berorientasi objek adalah paradigma baru dalam rekayasa perangkat lunak yang memandang sistem sebagai kumpulan objek-objek diskrit yang saling berinteraksi. Dasar pendekatan ini adalah konsep dari memodelkan entitas dunia nyata dan interaksinya dalam sistem perangkat lunak. Adapun penggunaan pendekatan berorientasi objek ini dilakukan dengan melalui *Unified Modeling Language* (UML) (Sommerville, 2011). Menurut Booch (2007) pendekatan berorientasi objek tersusun atas *Object Oriented Analysis* (OOA), *Object Oriented Design* (OOD), dan *Object Oriented Programming* (OOP). OOA merupakan proses pengembangan perangkat lunak yang berfokus pada memahami dan melakukan pemodelan problem domain, sedangkan OOD merupakan proses pembuatan *blueprint* atau prototipe untuk arsitektur sistem perangkat lunak yang diperoleh dari OOA, OOP merupakan proses implementasi pengembangan perangkat lunak yang sebenarnya dengan menggunakan bahasa pemrograman berorientasi objek.

2.3.2.1 Unified Modeling Language (UML)

Unified Modeling Language adalah bahasa pemodelan visual yang digunakan dalam melakukan rekayasa perangkat lunak, dan mendesain sistem untuk mempresentasikan, memvisualisasikan, dan mendokumentasikan berbagai aspek dalam sistem perangkat lunak. Menurut Nugroho (2004) UML adalah sebuah bahasa untuk menspesifikasikan, memvisualisasikan, serta mengkonstruksi bangunan dasar dari sistem perangkat lunak termasuk didalamnya dengan melibatkan pemodelan aturan – aturan bisnis. Adapun beberapa jenis model sistem yaitu:

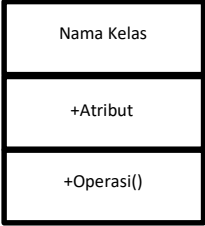
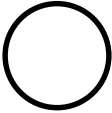


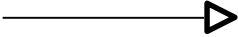

1. *Class Diagram*

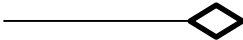
Class Diagram merupakan salah satu diagram UML yang digunakan dalam melakukan rekayasa perangkat lunak untuk merepresentasikan struktur dan hubungan relasi dalam kelas secara visual dalam sebuah sistem perangkat lunak. Menurut Wira (2019) *class diagram* adalah penjelasan secara garis besar mengenai kelas – kelas perancangan sistem dari sudut pandang struktur

sistem yang dapat memperjelas fungsi – fungsinya. Di dalam kelas diagram terdapat atribut dan operasi yang dapat memberikan penjelasan mengenai hubungan antara perancangan dan sistem yang dibentuk.

Tabel 2.2 Hubungan Dalam *Class Diagram*

Sumber: Rosa A.S dan M.Shalahuddin (2013:146)

Simbol	Nama Simbol	Keterangan
	Kelas	Merupakan struktur kelas yang berfungsi dalam melakukan pemetaan dari objek – objek dengan atribut dan operasi yang sama.
	Antarmuka	Merupakan atribut dengan perilaku yang kohesif.
	Asosiasi	Merupakan simbol yang memetakan hubungan antara satu objek dengan objek lainnya.
	Asosiasi Berarah	Merupakan simbol yang menggambarkan hubungan antara dua kelas, mengenai arah ketergantungan antara dua kelas tersebut.
	Generalisasi	Merupakan simbol yang menggambarkan hubungan pewarisan dari kelas induk kepada kelas turunannya.
	Kebergantungan	Merupakan simbol yang menyatakan ketergantungan antara dua kelas, yang mana perubahan yang dilakukan dalam satu

		kelas akan memberikan dampak kepada kelas lainnya.
	Agregasi	Merupakan simbol yang menggambarkan hubungan komposisi antara dua kelas, yang mana kelas utama adalah bagian dari keseluruhan kelas, apabila seluruh kelas dihapus, maka kelas utama akan ikut terhapus.

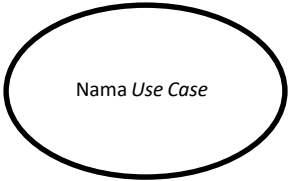
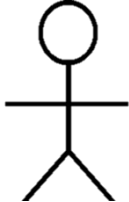


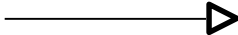

2. Use Case Diagram

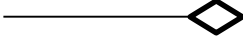
Use Case Diagram merupakan salah satu tipe diagram dari *Unified Modeling Language* (UML), yang digunakan untuk merepresentasikan interaksi antar aktor dan sebuah sistem atau aplikasi perangkat lunak secara visual. Menurut Sukamto dan Shalahuddin (2014:155) *use case diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat, *use case diagram* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem dan yang berhak menggunakan fungsi – fungsi tersebut. Selain itu, menurut Pratama (2014:48) menjelaskan bahwa *use case diagram* merupakan aliran kegiatan dan proses bisnis yang dilakukan oleh pengguna (aktor). Biasanya *use case diagram* digunakan pada tahap awal pengembangan perangkat lunak, untuk mengetahui dan melakukan komunikasi terhadap kebutuhan – kebutuhan fungsional dan perilaku dari sistem. Berikut merupakan simbol – simbol yang akan digunakan dalam membentuk suatu kelas diagram.

Tabel 2.3 Hubungan Dalam Use Case Diagram

Sumber: Rosa A.S dan M.Shalahuddin (2013:156 - 160)

Simbol	Nama Simbol	Keterangan
--------	-------------	------------

	<i>Use Case</i>	<p>Merupakan struktur kelas yang berfungsi dalam melakukan pemetaan dari objek – objek dengan atribut dan operasi yang sama.</p>
	Aktor	<p>Merupakan atribut dengan perilaku yang kohesif.</p>
	Asosiasi	<p>Merupakan simbol yang memetakan hubungan antara satu objek dengan objek lainnya.</p>
	Asosiasi Berarah	<p>Merupakan simbol yang menggambarkan hubungan antara dua kelas, mengenai arah ketergantungan antara dua kelas tersebut.</p>
	Generalisasi	<p>Merupakan simbol yang menggambarkan hubungan pewarisan dari kelas induk kepada kelas turunannya.</p>
	Kebergantungan	<p>Merupakan simbol yang menyatakan ketergantungan antara dua kelas, yang mana perubahan yang dilakukan dalam satu kelas akan memberikan dampak kepada kelas lainnya.</p>

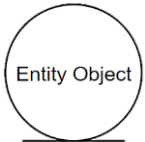
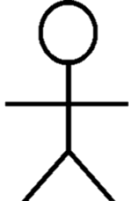
	Agregasi	Merupakan simbol yang menggambarkan hubungan komposisi antara dua kelas, yang mana kelas utama adalah bagian dari keseluruhan kelas, apabila seluruh kelas dihapus, maka kelas utama akan ikut terhapus.
---	----------	--

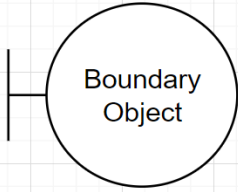
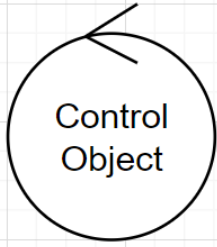
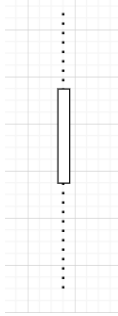



3. *Sequence Diagram*

Sequence Diagram merupakan salah satu tipe diagram interaksi yang ada di dalam UML, yang merepresentasikan interaksi dan komunikasi objek atau komponen secara visual di dalam sebuah sistem atau aplikasi perangkat lunak. Menurut Tohari dalam Tabrani dan Aghniya (2019:46), *sequence diagram* merupakan gambaran interaksi antara sejumlah objek dalam urutan waktu. Selain itu, menurut Valacich dan George (2016) *sequence diagram* menggambarkan interaksi antar objek selama jangka waktu tertentu. Dengan digunakannya *sequence diagram*, maka pemodelan aspek dinamis dalam sistem dapat dengan mudah tergambarkan, dan kolaborasi objek serta aktor dalam menyelesaikan suatu tugas tertentu dapat terlihat.

Tabel 2.4 Hubungan Dalam *Sequence Diagram*

Sumber: Rosa A.S dan M.Shalahuddin (2014)

Simbol	Nama Simbol	Keterangan
	<i>Entity</i>	Merupakan objek, komponen, atau elemen sistem yang memiliki peran dalam melakukan interaksi.
	Aktor	Merupakan entitas eksternal yang merupakan pengguna atau sistem lain, yang mana melakukan interaksi dengan sistem.

	<p><i>Boundary</i></p>	<p>Merupakan antarmuka atau objek pembatas yang menjadi penengah antara aktor dengan sistem.</p>
	<p><i>Control</i></p>	<p>Merupakan sebuah objek yang mengatur aliran dari urutan atau skenario.</p>
	<p><i>Lifeline</i></p>	<p>Merupakan eksistensi dari sebuah objek atau entitas dalam waktu tertentu pada <i>sequence diagram</i>.</p>
	<p><i>Line Message</i></p>	<p>Merupakan baris pesan yang menyatakan komunikasi atau interaksi antar objek atau entitas.</p>
	<p><i>Object</i></p>	<p>Merupakan sebuah instansi dari kelas, komponen, atau entitas yang terlibat dalam interaksi.</p>
	<p>Stimulus</p>	<p>Merupakan aksi, kejadian, atau pesan yang memicu respon dari suatu objek atau aktor</p>

2.4 Teknologi Pengembangan Sistem

Dalam melakukan pengembangan sistem pelayanan mandiri berbasis *web*, maka digunakan beberapa teknologi yang akan membantu dalam mengembangkan sistem tersebut, antara lain, *Laravel*, PHP, HTML, CSS, Javascript, *Bootstrap*, dan MySQL.

2.4.1 *Laravel*

Laravel merupakan salah satu *framework* aplikasi *web* PHP yang sangat populer digunakan dalam membangun aplikasi *web* yang modern. Menurut Awaludin (2016) *Laravel* adalah sebuah *framework* PHP yang dirilis dibawah lisensi MIT dan dikembangkan pertama kali oleh Taylor Otwell, dibangun dengan konsep MVC (Model View Controller). Sedangkan, menurut Aminudin (2016) *Laravel* adalah *framework* PHP MVC yang dikembangkan oleh Taylor Otwell pada tahun 2011 dan sekarang telah mencapai versi 4.2. Dengan digunakannya *framework* ini, maka pengembangan website yang berbasis MVC dapat memiliki kualitas yang tinggi, biaya yang dikeluarkan untuk pengembangan dan pemeliharaan dapat dikurangi, pengalaman bekerja dapat ditingkatkan.

2.4.2 PHP

PHP (PHP: *Hypertext Preprocessor*) adalah suatu bahasa pemrograman *scripting* sisi server, yang utamanya digunakan untuk pengembangan *web*, namun juga digunakan untuk pemrograman secara umum. Menurut Kadir (2014:2), PHP adalah (PHP Hypertext Processor) merupakan salah satu bahasa pemrograman berbentuk skrip yang sangat populer dalam pembuatan aplikasi web. Selain itu, menurut Sibero (dalam Puspitasari, 20 16:227), PHP adalah pemograman (interpreter) adalah proses penerjemahan baris sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan.

2.4.3 HTML

HTML (*HyperText Markup Language*) adalah bahasa *markup* yang mendefinisikan struktur dari konten, yang mana di dalamnya terdapat serangkaian elemen untuk membungkus bagian – bagian tertentu dari konten, agar dapat ditampilkan di laman web dengan sebuah cara. Menurut Sibero (2011) *HTML* adalah “Bahasa yang digunakan pada dokumen web sebagai bahasa pertukaran dokumen antar web. satu sama lain dan lebih penting dengan konten- konten yang kritis”.

Adapun fungsi dari *HTML* yaitu:

- Menstrukturasikan halaman web
- Menavigasikan internet
- Menyematkan foto dan video
- Meningkatkan tempat penyimpanan data dari sisi klien dan kemampuan *offline*
- Berinteraksi dengan *native API*

2.4.4 CSS

CSS (*Cascading Style Sheet*) adalah bahasa pemrograman yang mendeskripsikan bagaimana elemen *HTML* ditampilkan pada layar. Menurut Wahyudi (2017), CSS adalah suatu *“Bahasa pemrograman web yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam web sehingga tampilan web akan lebih rapi, terstruktur, dan seragam”*. CSS ini dapat mengatur susunan dari halaman web secara bersamaan. Dengan adanya CSS, maka tampilan pada web akan lebih menarik dengan berbagai macam variasi yang dimasukkan di dalamnya.

2.4.5 Javascript

Javascript merupakan bahasa pemrograman dinamis yang digunakan untuk pengembangan web, yang mana dengan menggunakan bahasa ini, implementasi fitur dinamis yang tidak dapat diterapkan oleh *HTML* dan CSS dapat dimungkinkan. Menurut Sibero (2013:150) *“Javascript adalah bahasa skrip (Scripting language), yaitu kumpulan intruksi perintah yang digunakan untuk mengendalikan beberapa bagian dari sistem operasi”*.

2.4.6 Bootstrap

Bootstrap adalah *framework* CSS yang paling terkenal untuk melakukan pengembangan situs web yang responsif. Menurut Husein Alatas (2013) *“Bootstrap merupakan framework untuk membangun desain web secara responsif”*. Dengan menggunakan *Bootstrap*, maka pengembangan web dapat menjadi lebih cepat dan mudah, selain itu, *Bootstrap* juga memiliki template desain untuk form, tombol, navigasi, modal, dan lainnya.

2.4.7 MySQL

MySQL adalah sistem manajemen basis data relasional yang digunakan secara luas untuk menyimpan, mengatur, dan mengambil data yang terstruktur. Menurut Lestanti dan Susana (2016), MySQL merupakan sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara garis. Selain itu, Menurut Fransiskus (dalam Akbar & Latifah, 2019), MySQL adalah sistem manajemen database yang sering digunakan bersama PHP. Penggunaan MySQL ini didasari karena fleksibilitas dan penggunaannya yang luas, sehingga MySQL dijadikan sebagai pilihan yang sangat baik untuk berbagai macam aplikasi.

2.4.8 Gopay

Gopay merupakan salah satu metode pembayaran dompet digital yang sering digunakan oleh masyarakat Indonesia, yang mana Gopay adalah produk yang dikeluarkan oleh PT Dompet Anak Bangsa sebagai alat pembayaran digital yang sah.

2.4.9 Ovo

Ovo merupakan dompet digital yang dibuat oleh PT Visionet Internasional, yang mana produk ini biasa digunakan untuk melakukan transaksi pembayaran secara *cashless*.

2.4.10 QRIS

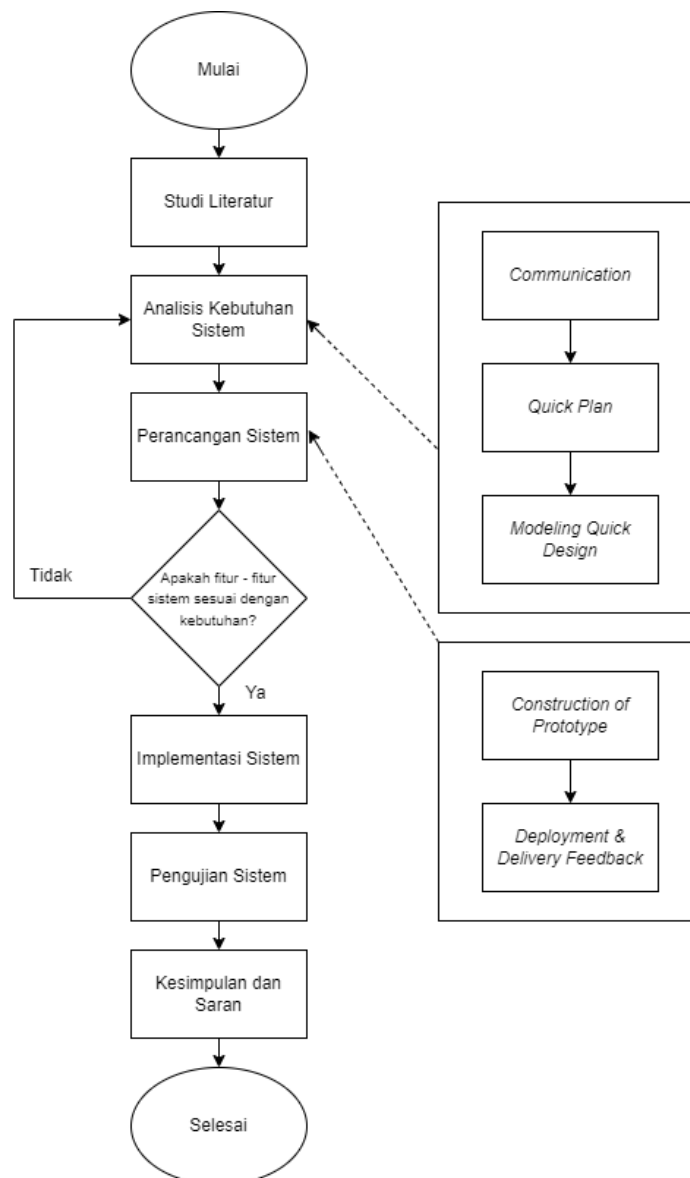
QRIS (*Quick Response Code Indonesian Standard*) merupakan metode pembayaran dengan menggunakan kode QR, yang mana sistem pembayaran ini dikeluarkan oleh Bank Indonesia dan Asosiasi Sistem Pembayaran Indonesia pada tahun 2019.

2.5 Pengujian Perangkat Lunak

Pengujian perangkat lunak adalah proses sistematis dan terkontrol dalam melakukan proses evaluasi aplikasi/sistem perangkat lunak, untuk mengidentifikasi dan menyelesaikan kecacatan, error atau masalah, sehingga fitur – fitur yang ada dapat memenuhi kebutuhan yang telah dikemukakan. Menurut Quadri dan Farooq (2010) pengujian perangkat lunak adalah proses verifikasi dan validasi apakah sebuah aplikasi *software* atau program memenuhi persyaratan bisnis dan persyaratan teknis, yang mengarahkan desain dan pengembangan, serta cara kerjanya seperti yang diharapkan, selain itu, mengidentifikasi kesalahan yang penting dengan adanya golongan berdasarkan tingkat keparahannya pada aplikasi yang perlu diperbaiki. Adapun beberapa jenis – jenis pengujian perangkat lunak seperti pengujian unit, pengujian validasi, dan pengujian integrasi.

BAB 3 METODOLOGI PENELITIAN

Dalam bab 3 ini dilakukan pembahasan mengenai metode penelitian yang dilakukan dalam penelitian ini, berupa langkah – langkah penelitian. Adapun tahapannya terdiri dari studi literatur, analisis kebutuhan sistem, perancangan sistem, implementasi sistem, pengujian sistem, dan melakukan pengambilan kesimpulan dan saran dari penelitian yang dilakukan.



Gambar 3.1 Strategi Penelitian

3.1 Studi Literatur

Pada tahapan studi literatur peneliti melakukan riset mengenai informasi – informasi yang dapat digunakan sebagai dasar dalam melakukan penelitian.

Adapun studi literature yang digunakan dalam penelitian ini yaitu mengenai sistem pelayanan mandiri, pengembangan *web* dengan pendekatan berorientasi objek, *Unified Modeling Language, framework* Laravel, pengujian perangkat lunak.

3.2 Analisis Kebutuhan Sistem

Dalam tahapan ini, dilakukan pengumpulan, pendokumentasian, analisis, dan validasi kebutuhan untuk sistem perangkat lunak yang dikembangkan, yang mana hal – hal tersebut dilakukan dalam beberapa tahapan berdasarkan metode *prototyping* yaitu *communication, quick plan, modeling quick design* (Pressman, 2002).

Adapun terdapat beberapa hal yang dilakukan dalam tahapan analisis kebutuhan sistem penelitian ini yaitu, melakukan diskusi bersama *stakeholder* dengan wawancara untuk melakukan pembahasan mengenai bagaimana sistem nantinya digunakan, melakukan eksplorasi kebutuhan – kebutuhan yang diperlukan untuk sistem yang akan dibangun, yang mana eksplorasi kebutuhan dibantu dengan adanya prototipe – prototipe untuk lebih mengelaborasi kebutuhan dari *stakeholder*. Selain itu, mengidentifikasi aktor dalam sistem, menspesifikasikan kebutuhan – kebutuhan, dan terakhir memodelkan kebutuhan – kebutuhan yang telah terspesifikasikan oleh *stakeholder*.

3.3 Perancangan Sistem

Berdasarkan hasil analisis kebutuhan sistem yang telah dilakukan pada tahap sebelumnya, maka pada tahapan ini dilakukan perancangan sistem, yang mana perancangan sistem merupakan proses sistematis dan komprehensif dalam mendefinisikan, mendesain dan menyiapkan segala sesuatunya untuk kesuksesan eksekusi dari sistem yang dikembangkan. Hal – hal yang dilakukan menurut Pressman (2002) dalam tahapan ini berdasarkan metode *prototyping* yaitu *construction of prototype* dan *deployment and delivery feedback*.

Beberapa hal yang dilakukan dalam tahapan perancangan sistem penelitian ini yaitu, merancang arsitektur dari sistem melalui pembentukan *class diagram* dan *sequence diagram*, merancang struktur basis data, merancang *user interface* sistem melalui pembentukan prototipe, dan terakhir melakukan evaluasi dari *prototype* yang telah dibentuk, yang mana evaluasi ini dilakukan oleh *stakeholder* untuk memperoleh ulasan dan kritik mengenai prototipe yang dibentuk, apabila prototipe yang dikembangkan belum memenuhi kebutuhan atau perlu dilakukan penambahan, maka ulasan dan kritik tersebut dijadikan sebagai dasar perubahannya.

3.4 Implementasi Sistem

Implementasi sistem merupakan tahap eksekusi pengembangan sistem perangkat lunak yang melibatkan pengembangan dan konfigurasi sistem, yang mana dasar implementasi sistem ini adalah prototipe yang telah disetujui pada tahapan sebelumnya.

3.5 Pengujian Sistem

Pengujian sistem merupakan tahapan ketika semua sistem diuji sebagai satu kesatuan untuk memastikan fitur – fitur yang dikembangkan sudah sesuai dengan kebutuhan dan spesifikasi dari pengguna. Tujuan utama pengujian sistem ini adalah melakukan evaluasi terhadap perilaku, fungsionalitas, performa, dan keandalan sistem. Dalam penelitian ini dilakukan tiga pengujian yaitu pengujian validasi untuk mengetahui apakah sistem yang dikembangkan sudah sesuai dengan kebutuhan yang telah dikemukakan melalui pengujian *black box*, pengujian unit untuk mengevaluasi unit – unit individu apakah sudah bekerja secara benar atau belum, yang dilakukan melalui pengujian *white box*, dan terakhir yaitu pengujian *compatibility* untuk menguji apakah sistem yang dikembangkan dapat bekerja pada berbagai macam *platform – platform web* tanpa adanya masalah.

3.6 Kesimpulan dan Saran

Tahapan ini adalah tahapan terakhir dalam penelitian ini, yang mana kesimpulan dan saran diperoleh dari hasil pengembangan sistem yang telah dikerjakan. Kesimpulan berisikan rangkuman mengenai hasil penelitian ini, yang digunakan sebagai dasar dalam menjawab rumusan masalah yang dikemukakan, sedangkan saran berisikan kritik/ulasan yang dapat digunakan untuk memperbaiki/meningkatkan pengembangan sistem kedepannya.

BAB 4 REKAYASA KEBUTUHAN

4.1 Deskripsi Umum Sistem

Sistem pelayanan mandiri pada toko Indomaret (PT. Petrolux Arya Mandala) merupakan sistem yang memungkinkan pelanggan Indomaret dalam melakukan perbelanjaan secara mandiri ataupun dengan bantuan karyawan toko. Adapun sistem ini dapat diakses dalam berbagai macam platform *web browser*, dengan fitur utamanya yaitu *self-service shopping* dan *assisted shopping*.

Dalam sistem ini, terdapat 3 aktor yaitu *General Manager* Operasional, karyawan, dan pelanggan. *General Manager* Operasional dapat melakukan pengolahan data barang, pengolahan data pelanggan, serta menerima laporan tindakan kriminalitas yang dilaporkan oleh pelanggan, sedangkan karyawan dapat melakukan pengolahan data barang, melakukan perbelanjaan untuk membantu pelanggan yang memerlukan bantuan, melihat daftar belanja pelanggan yang dibantu, dan menerima laporan tindakan kriminalitas yang dilaporkan oleh pelanggan. Pelanggan dapat melakukan kegiatan berbelanja secara mandiri atau dengan bantuan karyawan dan melakukan pelaporan kegiatan kriminalitas, serta melihat daftar riwayat belanjanya.

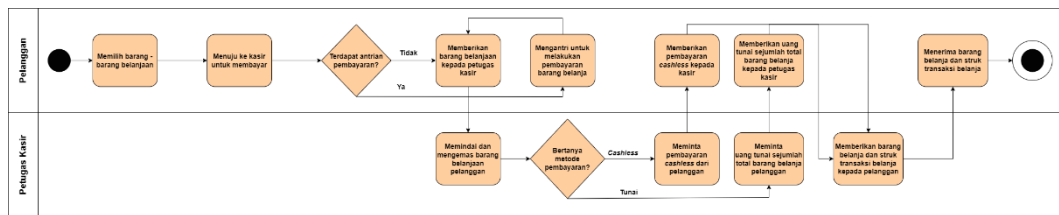
4.2 Elisitasi Kebutuhan (tambahkan referensi lampiran)

Elisitasi kebutuhan dilakukan untuk memperoleh kebutuhan dari sistem, yang nantinya spesifikasi yang dibutuhkan oleh sistem dapat diketahui. Berdasarkan wawancara yang dilakukan dengan *stakeholder*, maka diperoleh beberapa kebutuhan sistem yang terpapar pada Tabel 4.1.

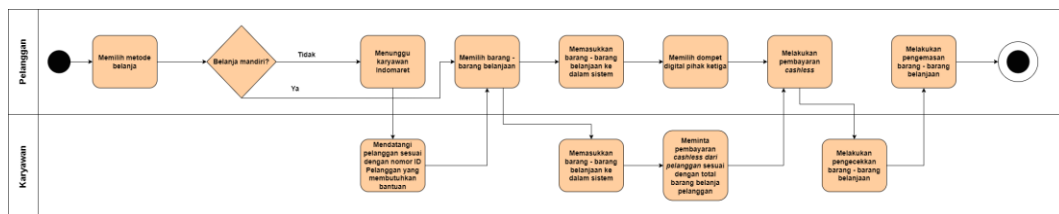
Tabel 4.1 Kebutuhan Sistem Hasil Wawancara dengan *Stakeholder*

No	Kebutuhan
1	Sistem mampu melakukan pengenalan terhadap hak akses General Manager Operasional, karyawan, pelanggan dan menampilkan tampilan halaman sesuai dengan hak aksesnya.
2	Sistem mampu menambahkan, mengurangi, merubah, dan menampilkan barang beserta dengan detail barang.
3	Sistem mampu memproses kegiatan belanja self service dan dengan bantuan karyawan
4	Sistem mampu menampilkan daftar barang beserta dengan deskripsi, gambar, jumlah stok, harga, dan kandungan gizi.
5	Sistem mampu menampilkan kategori barang belanjaan
6	Sistem mampu melakukan pembayaran dengan dompet digital pihak ketiga
7	Sistem mampu menampilkan riwayat belanja dari pelanggan

Berdasarkan hasil wawancara yang telah dilakukan pada Lampiran A Hasil Wawancara, maka diperoleh dasar pembentukan proses bisnis, yang mana proses bisnis ini akan menggambarkan kondisi sebelum dan sesudah diimplementasikannya sistem. Adapun pada Gambar 4.1 diberikan penjelasan mengenai prosedur belanja di Indomaret saat ini, yang divisualisasikan melalui *activity diagram*, dan pada Gambar 4.2 menjelaskan mengenai prosedur belanja di Indomaret ketika sistem diimplementasikan.



Gambar 4.1 Proses Bisnis As-Is



Gambar 4.2 Proses Bisnis To-Be

Dari kedua proses bisnis tersebut terdapat perbedaan yaitu, sebelum penggunaan sistem, Pelanggan hanya dapat melakukan perbelanjaan secara mandiri, akan tetapi, dengan adanya sistem, maka Pelanggan dapat memperoleh bantuan dari Karyawan. Selain itu, sebelum penggunaan sistem, Pelanggan diharuskan mengantri dalam melakukan pembayaran, namun, dengan penggunaan sistem, Pelanggan tidak perlu melakukan antri, namun langsung membayarkannya pada sistem. Terakhir yaitu, sebelum penggunaan sistem, Pelanggan harus menunggu pengemasan barang – barang belanjaan yang dilakukan oleh Karyawan, tetapi, dengan menggunakan sistem, Pelanggan dapat langsung melakukan pengemasan barang tanpa menunggu.

4.3 Identifikasi Aktor

Identifikasi aktor merupakan representasi dari pengguna – pengguna yang akan menggunakan sistem, yang mana identifikasi ini didasari oleh analisis kebutuhan yang sebelumnya telah diinstansiasi, adapun identifikasi aktor dapat dilihat pada Tabel 4.2.

Tabel 4.2 Identifikasi Aktor

No	Nama Aktor	Deskripsi
----	------------	-----------

1	Pelanggan	Aktor yang dapat melakukan kegiatan berbelanja secara mandiri atau dengan bantuan karyawan dan melakukan pelaporan kegiatan kriminalitas, serta melihat daftar riwayat belanjanya.
2	Karyawan	Aktor yang dapat mengolah data barang, melakukan perbelanjaan untuk membantu pelanggan yang memerlukan bantuan dan melihat daftar belanja pelanggan terkait, serta menerima laporan tindakan kriminalitas yang dilaporkan oleh pelanggan.
3	General Manager Operasional	Aktor yang dapat melakukan pengolahan data barang, pengolahan data pelanggan, dan menerima laporan tindakan kriminalitas yang dilaporkan oleh pelanggan.

4.4 Spesifikasi Kebutuhan

Berdasarkan kebutuhan yang telah diintensifikasi sebelumnya, maka kebutuhan – kebutuhan sistem tersebut perlu dispesifikasi. Spesifikasi kebutuhan ini dibagi menjadi dua jenis, yang pertama yaitu kebutuhan fungsional dan yang kedua yaitu kebutuhan non-fungsional. Dari kebutuhan fungsional ini dapat digambarkan seluruh kebutuhan yang dapat dilakukan oleh sistem dan keperluan aktor – aktor terhadap sistem yang dibentuk. Sedangkan, dari kebutuhan non-fungsional digambarkan batasan pada sistem yang secara tidak langsung dibutuhkan oleh aktor dalam menggunakan sistem. Kebutuhan – kebutuhan yang ada ini nantinya akan diberikan kode sebagai penanda antara kebutuhan fungsional dengan kebutuhan non-fungsional. Kode yang diberikan yaitu SSW yang merupakan singkatan dari *Self-Service Website*, kode F yang menyatakan kebutuhan fungsional, NF yang menyatakan kebutuhan non-fungsional, kode AA yang menyatakan nomor yang terdapat pada tiap kebutuhan, dan kode BB yang menyatakan nomor yang terdapat pada spesifikasi kebutuhan. Dengan demikian, struktur kode dari kebutuhan fungsional yaitu SSW-F-AA, spesifikasi kebutuhan fungsional yaitu SSW-F-AA-BB, kebutuhan non-fungsional yaitu SSW-NF-AA. Adapun spesifikasi kebutuhan dapat dilihat pada Tabel 4.3.

Tabel 4.3 Kebutuhan Fungsional

No	Kode	Fungsi	Deskripsi	Aktor
1	SSW-F-01	Login	Sistem mampu mengidentifikasi hak akses sesuai dengan otoritas	Pelanggan, Karyawan, General

			masing – masing aktor.	Manager Operasional
	SSW-F-01		Data yang diperlukan dalam identifikasi hak akses ini adalah <i>username</i> dan <i>password</i> .	Pelanggan, Karyawan, General Manager Operasional
2	SSW-F-02	Logout	Sistem mampu melakukan proses keluarnya aktor dari sistem.	Pelanggan, Karyawan, General Manager Operasional
3	SSW-F-03	Menampilkan Halaman Utama Pelanggan	Sistem mampu menyediakan sarana berupa tampilan halaman utama.	Pelanggan
	SSW-F-03-01		Sistem mampu menampilkan tombol belanja untuk mengakses fitur belanja mandiri dan belanja dengan bantuan karyawan.	
	SSW-F-03-02		Sistem mampu menampilkan tombol riwayat belanja untuk mengakses halaman riwayat belanja.	
	SSW-F-03-03		Sistem mampu menampilkan tombol profil untuk mengakses halaman profil.	
	SSW-F-03-04		Sistem mampu menampilkan tombol laporan kriminalitas untuk	

			mengakses fitur pelaporan kegiatan kriminalitas.	
4	SSW-F-04	Menampilkan Halaman Utama Karyawan	Sistem mampu menyediakan sarana berupa tampilan halaman utama.	Karyawan
	SSW-F-04-01		Sistem mampu menyediakan tombol pengolahan data barang untuk melakukan akses terhadap fitur pengolahan barang.	
	SSW-F-04-02		Sistem mampu menyediakan tombol belanja untuk mengakses fitur belanja untuk membantu pelanggan.	
	SSW-F-04-03		Sistem mampu menyediakan tombol lihat daftar belanja untuk melihat daftar belanja pelanggan yang dibantu.	
	SSW-F-04-04		Sistem mampu menyediakan tombol lihat laporan kriminalitas untuk mengakses daftar laporan kriminalitas dari pelanggan.	
5	SSW-F-05	Menampilkan Utama <i>General Manager</i> Operasional	Sistem mampu menyediakan sarana berupa tampilan halaman utama.	General Manager Operasional

	SSW-F-05-01		Sistem mampu menyediakan tombol pengolahan data barang untuk melakukan akses terhadap fitur pengolahan barang.	
	SSW-F-05-02		Sistem mampu menyediakan tombol validasi data barang untuk melakukan akses terhadap fitur validasi data barang.	
	SSW-F-05-03		Sistem mampu menyediakan tombol lihat data pelanggan untuk melihat data – data yang berhubungan dengan pelanggan terkait.	
	SSW-F-05-04		Sistem mampu menyediakan tombol lihat laporan kriminalitas untuk mengakses daftar laporan kriminalitas dari pelanggan.	
6	SSW-F-06	Melakukan belanja mandiri	Sistem mampu menyediakan mekanisme untuk melakukan belanja mandiri.	Pelanggan
	SSW-F-06-01		Data yang diperlukan adalah data – data barang kebutuhan di Indomaret.	
	SSW-F-06-02		Sistem mampu menyediakan	

			tombol detail barang.	
	SSW-F-06-03		Sistem mampu menyediakan tombol <i>buy</i> untuk memasukkan barang ke dalam keranjang belanja.	
	SSW-F-06-04		Sistem mampu menyediakan tombol tambah untuk menambahkan kuantitas barang yang dipilih.	
	SSW-F-06-05		Sistem mampu menyediakan tombol kurang untuk mengurangi kuantitas barang yang dipilih.	
	SSW-F-06-06		Sistem mampu menyediakan sarana berupa <i>input box</i> untuk memasukkan jumlah barang sesuai dengan keinginan pelanggan.	
	SSW-F-06-07		Sistem mampu menyediakan tombol kembali untuk mengembalikan ke halaman sebelumnya/ ke halaman utama.	
	SSW-F-06-08		Sistem mampu menyediakan tombol keranjang	

			untuk mengakses keranjang belanja.	
	SSW-F-06-09		Sistem mampu menyediakan tombol kategori untuk mengakses kategori barang	
	SSW-F-06-10		Sistem mampu menyediakan sarana berupa <i>search bar</i> untuk melakukan pencarian barang sesuai dengan nama yang dimasukkan.	
7	SSW-F-07	Melakukan belanja dengan bantuan karyawan	Sistem mampu menyediakan mekanisme untuk melakukan belanja mandiri.	Pelanggan dan Karyawan
	SSW-F-07-01		Data yang diperlukan adalah data – data barang kebutuhan di Indomaret.	
	SSW-F-07-02		Sistem mampu menyediakan tombol detail barang.	
	SSW-F-07-03		Sistem mampu menyediakan tombol <i>buy</i> untuk memasukkan barang ke dalam keranjang belanja.	
	SSW-F-07-04		Sistem mampu menyediakan tombol tambah untuk menambahkan	

			kuantitas barang yang dipilih.	
	SSW-F-07-05		Sistem mampu menyediakan tombol kurang untuk mengurangi kuantitas barang yang dipilih.	
	SSW-F-07-06		Sistem mampu menyediakan sarana berupa <i>input box</i> untuk memasukkan jumlah barang sesuai dengan keinginan pelanggan.	
	SSW-F-07-07		Sistem mampu menyediakan tombol kembali untuk mengembalikan ke halaman sebelumnya/ ke halaman utama.	
	SSW-F-07-08		Sistem mampu menyediakan tombol keranjang untuk mengakses keranjang belanja.	
	SSW-F-07-09		Sistem mampu menyediakan tombol kategori untuk mengakses kategori barang	
	SSW-F-07-10		Sistem mampu menyediakan sarana berupa <i>search bar</i> untuk melakukan pencarian barang sesuai dengan	

			nama yang dimasukkan.	
8	SSW-F-08	Menampilkan detail barang	Sistem mampu menyediakan sarana untuk menampilkan detail barang yang dipilih.	Pelanggan dan Karyawan
	SSW-F-08-01		Data yang ditampilkan yaitu kode barang, nama barang, deskripsi, gambar, jumlah stok, harga, dan kandungan gizi.	
9	SSW-F-09	Menampilkan kategori barang	Sistem mampu menyediakan sarana untuk menampilkan kategori – kategori barang.	Pelanggan dan Karyawan
	SSW-F-09-01		Data yang ditampilkan yaitu seluruh kategori barang – barang.	
10	SSW-F-10	Menampilkan keranjang belanja	Sistem mampu menyediakan sarana untuk menampilkan keranjang belanja.	Pelanggan dan Karyawan
	SSW-F-10-01		Data yang ditampilkan berupa nama barang, jumlah, total harga belanja.	
	SSW-F-10-02		Sistem mampu menyediakan tombol tambah untuk melakukan penambahan kuantitas barang.	
	SSW-F-10-03		Sistem mampu menyediakan	

			tombol kurang untuk mengurangi kuantitas barang.	
	SSW-F-10-04		Sistem mampu menyediakan tombol <i>remove</i> untuk menghapus barang dari keranjang	
	SSW-F-10-05		Sistem mampu menyediakan tombol <i>back</i> untuk kembali ke halaman sebelumnya.	
	SSW-F-10-06		Sistem mampu menyediakan tombol <i>checkout</i> untuk mengakses fitur pembayaran	
11	SSW-F-11	Melakukan pembayaran	Sistem mampu menyediakan sarana untuk memproses pembayaran melalui dompet digital pihak ketiga.	Pelanggan dan Karyawan
	SSW-F-11-01		Data – data yang diperlukan yaitu total belanja dan daftar pilihan pembayaran dompet digital pihak ketiga.	
	SSW-F-11-02		Sistem mampu menyediakan tombol <i>pay</i> untuk memproses pembayaran sesuai dengan pilihan metode pembayaran.	

12	SSW-F-12	Menampilkan riwayat belanja	Sistem mampu menyediakan sarana untuk menampilkan daftar riwayat belanja yang dilakukan oleh pelanggan	Pelanggan
	SSW-F-12-01		Data – data yang ditampilkan yaitu tanggal belanja, daftar barang, kuantitas barang, total harga, metode pembayaran, dan total harga belanja.	
	SSW-F-12-02		Sistem mampu menyediakan tombol filter untuk menampilkan daftar riwayat belanja berdasarkan bulan dan tahun	
13	SSW-F-13	Menambah pelaporan kegiatan kriminalitas	Sistem mampu menyediakan sarana untuk memproses pelaporan kegiatan kriminalitas	Pelanggan
	SSW-F-13-01		Data – data yang diperlukan berupa deskripsi kriminalitas dan bukti foto.	
	SSW-F-13-02		Sistem mampu menyediakan tombol <i>submit</i> untuk mengirimkan laporan tindakan kriminalitas	
14	SSW-F-14		Sistem mampu menyediakan	Karyawan dan General

		Menambah data barang	sarana untuk menambah data barang.	Manager Operasional
	SSW-F-14-01		Data – data yang diperlukan yaitu kode barang, nama barang, deskripsi, gambar, jumlah stok, harga, dan kandungan gizi	
	SSW-F-14-02		Sistem mampu menyediakan tombol <i>add</i> untuk menambahkan data barang.	
15	SSW-F-15	Mengubah data barang	Sistem mampu menyediakan sarana untuk melakukan perubahan data pada barang.	Karyawan dan General Manager Operasional
	SSW-F-15-01		Data – data yang diperlukan yaitu kode barang, nama barang, deskripsi, gambar, jumlah stok, harga, dan kandungan gizi.	
	SSW-F-15-02		Sistem mampu menyediakan tombol <i>update</i> untuk melakukan eksekusi perubahan data barang.	
16	SSW-F-16	Menghapus data barang	Sistem mampu menyediakan sarana untuk menghapus data barang.	Karyawan dan General Manager Operasional
17	SSW-F-17	Menampilkan data barang	Sistem mampu menyediakan sarana untuk	Karyawan dan General

			menampilkan data barang	Manager Operasional
	SSW-F-17-01		Data – data yang ditampilkan yaitu kode barang, nama barang, deskripsi, gambar, jumlah stok, harga, dan kandungan gizi.	
18	SSW-F-18	Menampilkan daftar data barang	Sistem mampu menyediakan sarana untuk menampilkan daftar data barang.	Karyawan dan General Manager Operasional
	SSW-F-18-01		Data yang ditampilkan yaitu daftar barang.	
	SSW-F-18-02		Sistem mampu menyediakan tombol <i>edit</i> untuk mengubah data barang.	
	SSW-F-18-03		Sistem mampu menyediakan tombol hapus untuk melakukan penghapusan data barang.	
	SSW-F-18-04		Sistem mampu menyediakan <i>search bar</i> untuk melakukan pencarian sesuai nama barang.	
	SSW-F-18-05		Sistem mampu menyediakan tombol filter untuk mengkategorikan barang sesuai dengan kategori yang dipilih.	

19	SSW-F-19	Menampilkan daftar laporan tindakan kriminalitas	Sistem mampu menyediakan sarana untuk menampilkan laporan tindakan kriminalitas.	Karyawan dan General Manager Operasional
	SSW-F-19-01		Data yang ditampilkan berupa daftar laporan tindakan kriminalitas.	
	SSW-F-19-02		Sistem mampu menyediakan tombol <i>detail</i> untuk melihat detil laporan tindakan kriminalitas.	
20	SSW-F-20	Menampilkan data laporan tindakan kriminalitas	Sistem mampu menyediakan sarana untuk menampilkan data laporan tindakan kriminalitas.	Karyawan dan General Manager Operasional
	SSW-F-20-01		Data – data yang ditampilkan yaitu deskripsi tindakan kriminalitas dan foto bukti kriminalitas.	
	SSW-F-20-02		Sistem mampu menyediakan tombol <i>verify</i> untuk melakukan verifikasi laporan tindakan kriminalitas.	
21	SSW-F-21	Menampilkan daftar data pelanggan	Sistem mampu menyediakan sarana untuk menampilkan daftar data pelanggan.	General Manager Operasional

	SSW-F-21-01		Data yang ditampilkan berupa daftar data pelanggan.	
	SSW-F-21-02		Sistem mampu menyediakan tombol <i>detail</i> untuk melihat detail data pelanggan.	
22	SSW-F-22	Menampilkan data pelanggan	Sistem mampu menyediakan sarana untuk menampilkan data pelanggan.	General Manager Operasional
	SSW-F-22-01		Data yang ditampilkan berupa nama, tanggal lahir, alamat, nomor telepon, alamat email.	
	SSW-F-22-02		Sistem mampu menyediakan tombol <i>status</i> untuk melakukan perubahan status pada data pelanggan.	
	SSW-F-22-03		Sistem mampu menyediakan tombol <i>history</i> untuk melihat daftar riwayat kegiatan belanja pelanggan.	

Tabel 4.4 Kebutuhan Non-Fungsional

No	Kode	Spesifikasi	Parameter
1	SSW-NF-01	Sistem dapat berjalan pada empat platform web, yaitu <i>Google Chrome</i> , <i>Bing</i> , <i>Mozilla Firefox</i> , dan <i>Microsoft Edge</i> .	<i>Compatibility</i>

2	SSW-NF-01	Sistem memiliki tampilan antarmuka yang mudah dipahami.	<i>Usability</i>
---	-----------	---	------------------

4.5 Pemodelan Kebutuhan Sistem

Tahapan pemodelan kebutuhan sistem ini merupakan tahapan penggambaran dari kebutuhan sistem, yang mana spesifikasinya telah dinstansiasi pada tahapan sebelumnya. Adapun visualisasinya diimplementasikan melalui pembentukan *use case diagram* dan *use case scenario*.

4.5.1 Use Case Diagram

Pada *use case diagram*, setiap aktor yang berinteraksi dengan sistem akan divisualisasikan ke dalam *use case diagram*. Dasar pembentukan *use case diagram* ini diperoleh dari hasil elisitasi kebutuhan dan identifikasi aktor yang telah dikemukakan pada tahapan sebelumnya. *Use case diagram* dari Sistem Pelayanan Mandiri untuk Toko Ritel di Kota Malang Berbasis Web dapat dilihat pada Gambar 4.3.

	<p>2. <i>General User</i> memasukkan email, nama, nomor telepon, <i>username</i>, <i>password</i>, <i>confirm password</i>, dan gambar.</p> <p>3. <i>General User</i> menekan tombol register untuk melakukan permintaan pemrosesan registrasi data yang dimasukkan kepada sistem.</p> <p>4. Sistem menampilkan halaman <i>login</i> untuk selanjutnya dilakukan proses <i>login</i>.</p>
Aliran Alternatif	<p>3a. Apabila email kosong, maka sistem menampilkan pesan, "Email kosong, silakan memasukkan alamat email!".</p> <p>3b. Apabila nama kosong, maka sistem menampilkan pesan, "Nama kosong, silakan memasukkan Nama!".</p> <p>3c. Apabila nomor telepon kosong, maka sistem menampilkan pesan, "Nomor Telepon kosong, silakan memasukkan Nomor Telepon!".</p> <p>3d. Apabila <i>username</i> kosong, maka sistem menampilkan pesan, "<i>Username</i> kosong, silakan memasukkan <i>Username</i>!".</p> <p>3e. Apabila <i>password</i> kosong, maka sistem menampilkan pesan, "<i>Password</i> kosong, silakan memasukkan <i>Passsword</i>!".</p> <p>3f. Apabila <i>confirm password</i> kosong, maka sistem menampilkan pesan, "Konfirmasi <i>Password</i> kosong, silakan memasukkan Konfirmasi <i>Password</i>!".</p> <p>3g. Apabila gambar kosong, maka sistem menampilkan pesan, "Gambar kosong, silakan memasukkan Gambar!".</p> <p>3h. Apabila email sudah terdaftar, maka sistem menampilkan pesan "Email terdaftar, silakan pilih Email yang lain!".</p> <p>3i. Apabila nomor telepon sudah terdaftar, maka sistem menampilkan pesan "Nomor telepon terdaftar, silakan pilih Nomor Telepon yang lain!".</p> <p>3j. Apabila <i>username</i> sudah terdaftar, maka sistem menampilkan pesan "<i>Username</i> terdaftar, silakan pilih <i>username</i> yang lain!"</p> <p>3k. Apabila <i>password</i> kurang dari 6 karakter, maka sistem akan menampilkan pesan "Silakan memasukkan <i>password</i> dengan minimal 6 karakter!".</p> <p>3l. Apabila <i>confirm password</i> tidak sama dengan <i>password</i> yang dimasukkan, maka sistem akan menampilkan pesan "<i>Passworrd</i> tidak sesuai!".</p> <p>3m. Apabila tipe <i>file</i> gambar yang dimasukkan tidak berbentuk gambar, maka sistem akan menampilkan pesan "Tipe <i>file</i> harus gambar!".</p>

	<p>3n. Apabila tipe gambar selain JPEG, PNG, JPG, dan GIF, maka sistem akan menampilkan pesan “Hanya tipe gambar JPEG, PNG, JPG, dan GIF yang diperbolehkan!”.</p> <p>3o. Apabila ukuran <i>file</i> gambar yang dimasukkan melebihi 2048KB/2MB, maka sistem akan menampilkan pesan “Ukuran <i>file</i> gambar tidak boleh melebihi 2048KB/2MB!”.</p>
Kondisi Akhir	Halaman <i>login</i> terbuka dengan akun pengguna yang sudah teregistrasi.

Tabel 4.6 Tabel Use Case Scenario Login

Login	
Tujuan	Sistem melakukan rekognisi/pengenalan terhadap otoritas dari <i>General User</i> .
Aktor	<i>General User</i>
Prakondisi	General User berada pada halaman <i>login</i> sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. <i>General User</i> mengakses halaman <i>login</i>. 2. <i>General User</i> memasukkan <i>username</i> dan <i>password</i>. 3. <i>General User</i> melakukan permintaan kepada sistem untuk memproses <i>login</i> yang dilakukan. 4. Sistem menampilkan halaman utama sistem sesuai dengan hak akses.
Aliran Alternatif	<p>3a. Apabila <i>username</i> kosong, maka sistem menampilkan pesan, “<i>Username</i> kosong, silakan memasukkan <i>Username</i>!”.</p> <p>3b. Apabila <i>password</i> kosong, maka sistem menampilkan pesan, “<i>Password</i> kosong, silakan memasukkan <i>Password</i>!”.</p> <p>3c. Apabila <i>username</i> dan/atau <i>password</i> salah, maka sistem akan menampilkan pesan “<i>Username</i> dan/atau <i>Password</i> salah, silakan masukkan <i>Username</i> dan/atau <i>Password</i> yang benar!”.</p>
Kondisi Akhir	<i>Main page</i> sistem terbuka sesuai dengan otoritas dari akun pengguna.

Tabel 4.7 Tabel Use Case Scenario Melakukan Belanja Mandiri

Melakukan Belanja Mandiri	
Tujuan	Pelanggan dapat melakukan kegiatan perbelanjaan secara mandiri melalui sistem.

Aktor	Pelanggan
Prakondisi	Pelanggan berada pada <i>main page</i> sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. Pelanggan memilih menu Belanja pada <i>dashboard</i> Pelanggan. 2. Pelanggan memilih pilihan cara berbelanja. 3. Pelanggan menekan tombol pilihan “Belanja Mandiri”. 4. Sistem menampilkan halaman belanja mandiri. 5. Pelanggan memilih barang – barang belanjaan yang dibutuhkan dan menekan tombol “Beli” pada barang yang dipilih untuk memasukkan ke dalam keranjang. 6. Pelanggan memilih ikon keranjang. 7. Sistem menampilkan halaman keranjang belanja dengan daftar barang belanjaan yang telah dipilih oleh Pelanggan. 8. Pelanggan memilih metode pembayaran. 9. Pelanggan menekan tombol “Pay” untuk melanjutkan pembayaran. 10. Sistem akan menampilkan konfirmasi pembayaran. 11. Pelanggan menekan tombol “Bayar” untuk melakukan konfirmasi pembayaran. 12. Sistem melakukan pemrosesan terhadap pembayaran yang dilakukan oleh Pelanggan. 13. Sistem menampilkan halaman keranjang kosong, dengan transaksi yang sudah terekam pada riwayat transaksi belanja pelanggan.
Aliran Alternatif	<p>3a. Apabila Pelanggan memilih tombol pilihan “Belanja dengan Bantuan Karyawan”, maka halaman yang ditampilkan adalah halaman keranjang, dengan halaman belanja berada pada perangkat karyawan.</p> <p>5a. Apabila stok barang habis, maka sistem akan menampilkan pesan “Stok Habis!” pada barang yang terkait.</p> <p>12a. Apabila pelanggan belum memilih metode pembayaran, maka sistem akan menampilkan pesan “Pilih metode pembayaran!”.</p> <p>12a. Apabila pelanggan memilih metode pembayaran dengan saldo kurang dari total harga barang belanjaan, maka sistem akan menampilkan pesan “Saldo kurang, silakan pilih metode pembayaran yang lain!”.</p>
Kondisi Akhir	Sistem menampilkan halaman keranjang belanja yang kosong.

Tabel 4.8 Tabel Use Case Scenario Menampilkan Detil Barang Belanja

Menampilkan Detil Barang Belanja	
Tujuan	Sistem menampilkan data barang belanja secara detil.
Aktor	Pelanggan
Prakondisi	Pelanggan berada pada halaman Belanja.
Aliran Utama	1. Pelanggan memilih tombol “Detil”, pada barang yang akan dilihat detilnya. 2. Sistem menampilkan <i>pop-up</i> detil barang yang dipilih.
Aliran Alternatif	2a. Apabila tidak terdapat data barang pada <i>database</i> sistem, maka sistem akan menampilkan tulisan “Data tidak ditemukan!”.
Kondisi Akhir	Sistem menampilkan <i>pop-up</i> detil data barang.

Tabel 4.9 Tabel *Use Case Scenario* Menampilkan Kategori Barang Belanja

Menampilkan Kategori Barang Belanja	
Tujuan	Sistem menampilkan kategori barang belanja.
Aktor	Pelanggan
Prakondisi	Pelanggan berada pada halaman belanja.
Aliran Utama	1. Pelanggan memilih tombol “Kategori”. 2. Sistem menampilkan pilihan kategori barang.
Aliran Alternatif	2a. Apabila tidak terdapat data kategori pada <i>database</i> sistem, maka sistem akan menampilkan tulisan “Data tidak ditemukan!”.
Kondisi Akhir	Sistem menampilkan halaman belanja dengan adanya pilihan kategori barang.

Tabel 4.10 Tabel *Use Case Scenario* Menampilkan Keranjang Belanja

Menampilkan Keranjang Belanja	
Tujuan	Sistem menampilkan keranjang barang belanja Pelanggan.
Aktor	Pelanggan
Prakondisi	Pelanggan berada pada halaman belanja.
Aliran Utama	1. Pelanggan menekan ikon “keranjang” pada halaman <i>self-service shopping</i> . 2. Sistem menampilkan halaman keranjang belanja.
Aliran Alternatif	2a. Apabila tidak terdapat data barang belanjaan yang dipilih oleh Pelanggan, maka sistem akan menampilkan halaman keranjang kosong.

Kondisi Akhir	Sistem menampilkan halaman keranjang belanja.

Tabel 4.11 Tabel *Use Case Scenario* Melakukan Pembayaran Barang Belanja Mandiri

Melakukan Pembayaran Barang Belanja Mandiri	
Tujuan	Sistem memproses pembayaran barang belanja yang dilakukan oleh Pelanggan.
Aktor	Pelanggan
Prakondisi	Pelanggan berada pada halaman keranjang belanja sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. Pelanggan memilih ikon keranjang. 2. Sistem menampilkan halaman keranjang belanja dengan daftar barang belanjaan yang telah dipilih oleh Pelanggan. 3. Pelanggan memilih metode pembayaran. 4. Pelanggan menekan tombol “Pay” untuk melanjutkan pembayaran. 5. Sistem akan menampilkan konfirmasi pembayaran. 6. Pelanggan menekan tombol “Bayar” untuk melakukan konfirmasi pembayaran. 7. Sistem melakukan pemrosesan terhadap pembayaran yang dilakukan oleh Pelanggan. 8. Sistem menampilkan halaman keranjang kosong, dengan transaksi yang sudah terekam pada riwayat transaksi belanja pelanggan.
Aliran Alternatif	<ol style="list-style-type: none"> 6a. Apabila pelanggan memilih tombol “Batal”, maka sistem akan menampilkan pesan “Pembayaran dibatalkan!”. 6b. Apabila pelanggan belum memilih metode pembayaran, maka sistem akan menampilkan pesan “Pilih metode pembayaran!”. 6c. Apabila pelanggan memilih metode pembayaran dengan saldo kurang dari total harga barang belanjaan, maka sistem akan menampilkan pesan “Saldo kurang, silakan pilih metode pembayaran yang lain!”.
Kondisi Akhir	Sistem menampilkan halaman keranjang belanja kosong.

Tabel 4.12 Tabel *Use Case Scenario* Menampilkan Riwayat Belanja

Menampilkan Riwayat Belanja	
Tujuan	Sistem menampilkan data riwayat belanja Pelanggan.

Aktor	Pelanggan
Prakondisi	Pelanggan berada pada <i>main page</i> sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. Pelanggan memilih memilih menu Riwayat Transaksi pada <i>dashboard</i> Pelanggan. 2. Sistem menampilkan daftar riwayat belanja Pelanggan.
Aliran Alternatif	2a. Apabila tidak terdapat data riwayat belanja, maka sistem akan menampilkan halaman kosong.
Kondisi Akhir	Sistem menampilkan halaman data riwayat belanja Pelanggan.

Tabel 4.13 Tabel *Use Case Scenario* Menambah Pelaporan Kegiatan Kriminalitas

Menambah Pelaporan Kegiatan Kriminalitas	
Tujuan	Sistem memproses pelaporan kegiatan kriminalitas dari Pelanggan.
Aktor	Pelanggan
Prakondisi	Pelanggan berada pada <i>main page</i> sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. Pelanggan memilih memilih menu Laporan Kriminalitas pada <i>dashboard</i> Pelanggan. 2. Sistem menampilkan halaman Laporan Kriminalitas. 3. Pelanggan memasukkan data pelaporan kegiatan kriminalitas yang terdiri dari deskripsi kriminalitas yang dilakukan dan foto bukti. 4. Pelanggan menekan tombol "<i>submit</i>". 5. Sistem memproses pelaporan yang dilakukan oleh Pelanggan. 6. Sistem menampilkan halaman <i>Report a crime</i> dengan data yang terbaru pada daftar pelaporan.
Aliran Alternatif	<p>2a. Apabila tidak terdapat laporan kriminalitas, maka sistem akan menampilkan form penambahan laporan kriminalitas dan tabel laporan kriminalitas kosong.</p> <p>5a. Apabila tidak terdapat deskripsi laporan, maka sistem akan menampilkan pesan "Deskripsi kosong, silakan mengisikan kolom deskripsi!".</p> <p>5b. Apabila tidak terdapat foto laporan, maka sistem akan menampilkan pesan "Foto kosong, silakan menambahkan foto pada kolom foto!".</p>
Kondisi Akhir	Sistem menampilkan halaman <i>Report a crime</i> dengan data yang <i>ter-update</i> pada daftar pelaporan.

Tabel 4.14 Tabel *Use Case Scenario* Melakukan Belanja Dengan Bantuan Karyawan

Melakukan Belanja Dengan Bantuan Karyawan	
Tujuan	Karyawan melakukan belanja untuk membantu Pelanggan.
Aktor	Pelanggan dan Karyawan
Prakondisi	Pelanggan berada pada <i>main page</i> sistem Pelanggan dan Karyawan berada pada <i>main page</i> sistem Karyawan.
Aliran Utama	<ol style="list-style-type: none"> 1. Pelanggan memilih menu Belanja pada <i>dashboard</i> Pelanggan. 2. Sistem menampilkan pilihan cara berbelanja. 3. Pelanggan memilih tombol pilihan “Belanja dengan Bantuan Karyawan”. 4. Sistem menampilkan halaman keranjang Belanja dengan Bantuan Karyawan. 5. Sistem mengirimkan notifikasi kepada Karyawan untuk membantu melakukan belanja bagi Pelanggan. 6. Karyawan memilih ikon “<i>chat</i>” pada <i>dashboard</i> 7. Sistem akan menampilkan daftar pelanggan yang membutuhkan bantuan dalam berbelanja. 8. Karyawan menekan tombol “Bantu” pada pelanggan yang membutuhkan bantuan dalam berbelanja. 9. Sistem menampilkan halaman belanja pada sistem karyawan. 10. Pelanggan memilih barang belanjaan yang dibutuhkan dan Karyawan menambahkan barang – barang belanjaan yang dipilih oleh Pelanggan pada sistemnya, untuk dimasukkan ke dalam keranjang, dengan menekan tombol “Beli” pada barang yang diperlukan. 11. Karyawan memilih ikon keranjang. 12. Sistem menampilkan daftar barang belanjaan yang telah dipilih oleh Pelanggan pada keranjang belanja sistem Karyawan dan Pelanggan. 13. Pelanggan memilih metode pembayaran pada sistem Karyawan. 14. Karyawan menekan tombol “Pay” untuk melanjutkan pembayaran. 15. Sistem menampilkan konfirmasi pembayaran pada sistem Karyawan 16. Sistem melakukan pemrosesan terhadap pembayaran yang dilakukan.

	17. Sistem menampilkan keranjang belanja kosong, dengan transaksi yang sudah terekam pada riwayat belanja.
Aliran Alternatif	3a. Apabila Pelanggan memilih tombol pilihan “Belanja Mandiri”, maka sistem akan menampilkan halaman belanja mandiri. 10a. Apabila stok barang habis, maka sistem akan menampilkan pesan “Stok Habis”.
Kondisi Akhir	Sistem menampilkan halaman keranjang belanja kosong pada sistem Karyawan dan sistem Pelanggan.

Tabel 4.15 Tabel Use Case Scenario Melakukan Pembayaran Barang Belanja dengan Bantuan Karyawan

Melakukan Pembayaran Barang Belanja dengan bantuan Karyawan	
Tujuan	Sistem memproses pembayaran barang belanja yang dilakukan oleh Pelanggan.
Aktor	Pelanggan dan Karyawan
Prakondisi	Pelanggan berada pada halaman keranjang belanja sistem Pelanggan. Karyawan berada pada halaman keranjang sistem Karyawan.
Aliran Utama	<ol style="list-style-type: none"> 1. Pelanggan memilih ikon keranjang. 2. Sistem menampilkan halaman keranjang belanja dengan daftar barang belanjaan yang telah dipilih oleh Pelanggan. 3. Pelanggan memilih metode pembayaran. 4. Pelanggan menekan tombol “Pay” untuk melanjutkan pembayaran. 5. Sistem akan menampilkan konfirmasi pembayaran. 6. Pelanggan menekan tombol “Bayar” untuk melakukan konfirmasi pembayaran. 7. Sistem melakukan pemrosesan terhadap pembayaran yang dilakukan oleh Pelanggan. 8. Sistem menampilkan halaman keranjang kosong, dengan transaksi yang sudah terekam pada riwayat transaksi belanja pelanggan.
Aliran Alternatif	7a. Apabila pelanggan memilih tombol “Batal”, maka sistem akan menampilkan pesan “Pembayaran dibatalkan!”.

	<p>7b. Apabila pelanggan belum memilih metode pembayaran, maka sistem akan menampilkan pesan “Pilih metode pembayaran!”.</p> <p>7c. Apabila pelanggan memilih metode pembayaran dengan saldo kurang dari total harga barang belanjaan, maka sistem akan menampilkan pesan “Saldo kurang, silakan pilih metode pembayaran yang lain!”.</p> <p>7d. Apabila pelanggan memilih metode pembayaran “QRIS”, maka akan menampilkan kode QR, untuk selanjutnya dilakukan pemindaian oleh pelanggan.</p>
Kondisi Akhir	Sistem menampilkan halaman keranjang belanja kosong pada sistem pelanggan dan karyawan.

Tabel 4.16 Tabel *Use Case Scenario* Menampilkan Kategori Barang Belanja dengan Bantuan Karyawan

Menampilkan Kategori Barang Belanja dengan Bantuan Karyawan	
Tujuan	Sistem menampilkan kategori barang pada sistem Karyawan.
Aktor	Karyawan
Prakondisi	Karyawan berada pada halaman belanja sistem Karyawan.
Aliran Utama	<ol style="list-style-type: none"> 1. Karyawan memilih tombol kategori pada halaman belanja. 2. Sistem menampilkan pilihan kategori barang.
Aliran Alternatif	2a. Apabila tidak terdapat data kategori pada <i>database</i> sistem, maka sistem akan menampilkan tulisan “Data tidak ditemukan”.
Kondisi Akhir	Sistem menampilkan halaman belanja sistem dengan adanya pilihan kategori barang.

Tabel 4.17 Tabel *Use Case Scenario* Menampilkan Detil Barang Belanja dengan Bantuan Karyawan

Menampilkan Detil Barang Belanja dengan Bantuan Karyawan	
Tujuan	Sistem menampilkan detil data barang.
Aktor	Karyawan
Prakondisi	Karyawan berada pada halaman belanja sistem Karyawan.
Aliran Utama	<ol style="list-style-type: none"> 1. Pelanggan memilih tombol “Detil”, pada barang yang akan dilihat detilnya. 2. Sistem menampilkan <i>pop-up</i> detil barang yang dipilih.

Aliran Alternatif	2a. Apabila tidak terdapat data barang pada <i>database</i> sistem, maka sistem akan menampilkan tulisan “Data tidak ditemukan!”.
Kondisi Akhir	Sistem menampilkan <i>pop-up</i> detil data barang.

Tabel 4.18 Tabel *Use Case Scenario* Menampilkan Keranjang Belanja dengan Bantuan Karyawan

Menampilkan Keranjang Belanja dengan Bantuan Karyawan	
Tujuan	Sistem menampilkan keranjang barang belanja Pelanggan pada sistem Karyawan.
Aktor	Karyawan
Prakondisi	Karyawan berada pada halaman belanja sistem Karyawan.
Aliran Utama	<ol style="list-style-type: none"> 1. Karyawan menekan ikon “keranjang” pada halaman <i>self-service shopping</i>. 2. Sistem menampilkan halaman keranjang belanja.
Aliran Alternatif	2a. Apabila tidak terdapat data barang belanjaan yang dipilih, maka sistem akan menampilkan tulisan “Data tidak ditemukan”.
Kondisi Akhir	Sistem menampilkan halaman keranjang belanja.

Tabel 4.19 Tabel *Use Case Scenario* Menambah Data Barang

Menambah Data Barang	
Tujuan	Karyawan melakukan penambahan data barang ke dalam sistem.
Aktor	Karyawan
Prakondisi	Karyawan berada pada <i>main page</i> sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. Karyawan memilih menu “Data Barang”. 2. Sistem menampilkan halaman Data Barang. 3. Karyawan mengisi data barang yang akan dimasukkan pada form penambahan barang, seperti id barang, nama, jenis, harga, deskripsi, komposisi, tanggal kedaluwarsa, foto, jumlah stok barang, kategori. 4. Karyawan memilih tombol “<i>Submit</i>”. 5. Sistem memproses penambahan data barang. 6. Sistem menampilkan halaman data barang dengan data terbaru pada daftar barang.

Aliran Alternatif	<p>2a. Apabila tidak terdapat data barang, maka sistem akan menampilkan halaman data barang dengan tabel data barang kosong.</p> <p>5a. Apabila tidak terdapat id barang, maka sistem akan menampilkan pesan “ID Barang Kosong!”.</p> <p>5b. Apabila tidak terdapat nama barang, maka sistem akan menampilkan pesan “Nama Barang Kosong!”.</p> <p>5c. Apabila tidak terdapat jenis barang, maka sistem akan menampilkan pesan “Jenis Barang Kosong!”.</p> <p>5d. Apabila tidak terdapat harga barang, maka sistem akan menampilkan pesan “Harga Barang Kosong!”.</p> <p>5e. Apabila tidak terdapat deskripsi barang, maka sistem akan menampilkan pesan “Deskripsi Barang Kosong!”.</p> <p>5f. Apabila tidak terdapat komposisi barang, maka sistem akan menampilkan pesan “Komposisi Barang Kosong!”.</p> <p>5g. Apabila tidak terdapat kategori barang yang dipilih, maka sistem akan menampilkan pesan “Silakan pilih kategori!”.</p> <p>5h. Apabila tidak terdapat tanggal kedaluwarsa barang, maka sistem akan menampilkan pesan “Tanggal Kedaluwarsa Barang Kosong!”.</p> <p>5i. Apabila tidak terdapat foto barang, maka sistem akan menampilkan pesan “Foto Barang Kosong!”.</p> <p>5j. Apabila tidak terdapat jumlah stok barang, maka sistem akan menampilkan pesan “Jumlah Stok Barang Kosong!”.</p> <p>5k. Apabila id barang sudah terdaftar, maka sistem akan menampilkan pesan “ID Barang sudah terdaftar!”.</p>
Kondisi Akhir	Sistem menampilkan halaman data barang dengan data terbaru pada daftar barang.

Tabel 4.20 Tabel *Use Case Scenario* Mengubah Data Barang

Mengubah Data Barang	
Tujuan	Karyawan melakukan perubahan data barang ke dalam sistem.
Aktor	Karyawan
Prakondisi	Karyawan berada pada <i>main page</i> sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. Karyawan memilih menu “Data Barang”. 2. Sistem menampilkan halaman Data Barang. 3. Karyawan memilih tombol “<i>Edit</i>” pada data barang yang akan dilakukan perubahan. 4. Sistem menampilkan form yang berisikan data barang.

	<p>5. Karyawan mengisi data barang yang ingin diperbaharui, seperti nama, jenis, harga, deskripsi, komposisi, tanggal kedaluwarsa, jumlah stok barang, foto, kategori.</p> <p>6. Karyawan memilih tombol <i>"Submit"</i>.</p> <p>7. Sistem memproses perubahan data barang.</p> <p>8. Sistem menampilkan halaman <i>"Data Barang"</i> dengan data terbaru pada daftar barang.</p>
Aliran Alternatif	<p>2a. Apabila tidak terdapat data barang, maka sistem akan menampilkan halaman data barang dengan tabel data barang kosong.</p> <p>7a. Apabila tidak terdapat nama barang, maka sistem akan menampilkan pesan <i>"Nama Barang Kosong!"</i>.</p> <p>7b. Apabila tidak terdapat jenis barang, maka sistem akan menampilkan pesan <i>"Jenis Barang Kosong!"</i>.</p> <p>7c. Apabila tidak terdapat harga barang, maka sistem akan menampilkan pesan <i>"Harga Barang Kosong!"</i>.</p> <p>7d. Apabila tidak terdapat deskripsi barang, maka sistem akan menampilkan pesan <i>"Deskripsi Barang Kosong!"</i>.</p> <p>7e. Apabila tidak terdapat komposisi barang, maka sistem akan menampilkan pesan <i>"Komposisi Barang Kosong!"</i>.</p> <p>7f. Apabila tidak terdapat kategori barang yang dipilih, maka sistem akan menampilkan pesan <i>"Silakan pilih kategori!"</i>.</p> <p>7g. Apabila tidak terdapat tanggal kedaluwarsa barang, maka sistem akan menampilkan pesan <i>"Tanggal Kedaluwarsa Barang Kosong!"</i>.</p> <p>7h. Apabila tidak terdapat foto barang, maka sistem akan menampilkan pesan <i>"Foto Barang Kosong!"</i>.</p> <p>7i. Apabila tidak terdapat jumlah stok barang, maka sistem akan menampilkan pesan <i>"Jumlah Stok Barang Kosong!"</i>.</p>
Kondisi Akhir	Sistem menampilkan halaman data barang dengan data terbaru pada daftar barang.

Tabel 4.21 Tabel *Use Case Scenario* Menghapus Data Barang

Menghapus Data Barang	
Tujuan	Karyawan melakukan penghapusan data barang dalam sistem.
Aktor	Karyawan
Prakondisi	Karyawan berada pada <i>main page</i> sistem.
Aliran Utama	1. Karyawan memilih menu <i>"Data Barang"</i> .

	<ol style="list-style-type: none"> 2. Sistem menampilkan halaman Data Barang. 3. Karyawan memilih tombol “Hapus” pada data barang yang akan dihapus. 4. Sistem menampilkan form konfirmasi penghapusan. 5. Karyawan memilih tombol “Ok”. 6. Sistem memproses penghapusan data barang. 7. Sistem menampilkan halaman Data Barang dengan data terbaru pada daftar barang.
Aliran Alternatif	<p>2a. Apabila tidak terdapat data barang, maka sistem akan menampilkan halaman data barang dengan tabel data barang kosong.</p> <p>6a. Apabila Karyawan memilih tombol “Cancel”, maka sistem akan menampilkan pesan “Penghapusan Data Dibatalkan!”.</p>
Kondisi Akhir	Sistem menampilkan halaman data barang dengan data terbaru pada daftar barang.

Tabel 4.22 Tabel Use Case Scenario Menampilkan Daftar Data Barang

Menampilkan Daftar Data Barang	
Tujuan	Sistem menampilkan daftar data barang pada sistem Karyawan.
Aktor	Karyawan
Prakondisi	Karyawan berada pada <i>main page</i> sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. Karyawan memilih menu “Data Barang”. 2. Sistem menampilkan halaman data barang dengan daftar data barang – barang.
Aliran Alternatif	2a. Apabila tidak terdapat data barang, maka sistem akan menampilkan halaman data barang dengan tabel data barang kosong.
Kondisi Akhir	Sistem menampilkan halaman daftar data barang.

Tabel 4.23 Tabel Use Case Scenario Menampilkan Daftar Laporan Tindakan Kriminalitas

Menampilkan Daftar Laporan Tindakan Kriminalitas	
Tujuan	Sistem menampilkan daftar laporan tindakan kriminalitas pada sistem Karyawan.
Aktor	Karyawan
Prakondisi	Karyawan berada pada <i>main page</i> sistem.
Aliran Utama	1. Karyawan memilih menu “Laporan Kriminalitas”.

	2. Sistem menampilkan halaman laporan kriminalitas dengan daftar laporan tindakan kriminalitas.
Aliran Alternatif	2a. Apabila tidak terdapat data laporan tindakan kriminalitas, halaman kriminalitas dengan tabel kosong.
Kondisi Akhir	Sistem menampilkan halaman laporan kriminalitas dengan daftar laporan kriminalitas.

Tabel 4.24 Tabel *Use Case Scenario* Mengubah Data Laporan Tindakan Kriminalitas

Mengubah Data Laporan Tindakan Kriminalitas	
Tujuan	Sistem melakukan perubahan detil laporan tindakan kriminalitas pada sistem Karyawan.
Aktor	Karyawan
Prakondisi	Karyawan berada pada <i>main page</i> sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. Karyawan memilih menu “Laporan Kriminalitas”. 2. Sistem menampilkan halaman laporan kriminalitas. 3. Karyawan memilih tombol “Edit” pada laporan yang akan dilakukan perubahan data. 4. Karyawan memilih status laporan. 5. Karyawan menekan tombol “Update”. 6. Sistem melakukan perubahan data laporan tindakan kriminalitas yang dilakukan. 7. Sistem menampilkan halaman laporan kriminalitas dengan data pada tabel laporan kriminalitas yang diperbaharui.
Aliran Alternatif	2a. Apabila tidak terdapat data laporan tindakan kriminalitas, halaman kriminalitas dengan tabel kosong.
Kondisi Akhir	Sistem menampilkan halaman laporan kriminalitas dengan data pada tabel laporan kriminalitas yang terbaru.

Tabel 4.25 Tabel *Use Case Scenario* Menghapus Data Laporan Tindakan Kriminalitas

Mengubah Data Laporan Tindakan Kriminalitas	
Tujuan	Sistem melakukan penghapusan laporan tindakan kriminalitas pada sistem Karyawan.
Aktor	Karyawan
Prakondisi	Karyawan berada pada <i>main page</i> sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. Karyawan memilih menu “Laporan Kriminalitas”. 2. Sistem menampilkan halaman laporan kriminalitas.

	3. Karyawan memilih tombol “Hapus” pada laporan yang akan dilakukan perubahan data. 4. Sistem melakukan menampilkan konfirmasi penghapusan data laporan tindakan kriminalitas. 5. Karyawan memilih tombol “Ok” untuk melakukan konfirmasi penghapusan data laporan kriminalitas. 6. Sistem memproses penghapusan data laporan kriminalitas. 7. Sistem menampilkan halaman laporan kriminalitas dengan data pada tabel laporan kriminalitas yang diperbaharui.
Aliran Alternatif	2a. Apabila tidak terdapat data laporan tindakan kriminalitas, halaman kriminalitas dengan tabel kosong. 6a. Apabila karyawan memilih tombol “Cancel”, maka sistem akan menampilkan pesan “Penghapusan data dibatalkan!”
Kondisi Akhir	Sistem menampilkan halaman laporan kriminalitas dengan data pada tabel laporan kriminalitas yang terbaru.

Tabel 4.26 Tabel Use Case Scenario Menampilkan Daftar Data Pelanggan

Menampilkan Daftar Data Pelanggan	
Tujuan	Sistem menampilkan daftar data Pelanggan pada sistem General Manager Operasional.
Aktor	General Manager Operasional
Prakondisi	General Manager Operasional berada pada <i>main page</i> sistem.
Aliran Utama	1. General Manager Operasional memilih menu “Data Pelanggan”. 2. Sistem menampilkan halaman data pelanggan dengan tabel data pelanggan.
Aliran Alternatif	2a. Apabila tidak terdapat data Pelanggan, maka sistem akan menampilkan tabel data pelanggan kosong.
Kondisi Akhir	Sistem menampilkan halaman data pelanggan dengan daftar pelanggan.

Tabel 4.27 Tabel Use Case Scenario Mengubah Data Pelanggan

Mengubah Data Pelanggan	
Tujuan	Sistem melakukan perubahan data pelanggan pada sistem General Manager Operasional.

Aktor	General Manager Operasional
Prakondisi	General Manager Operasional berada pada <i>main page</i> sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. General Manager Operasional memilih menu “Data Pelanggan”. 2. Sistem menampilkan halaman data pelanggan dengan adanya daftar Pelanggan. 3. Sistem menekan tombol “<i>Edit</i>” pada data Pelanggan yang akan dilakukan perubahan. 4. Sistem menampilkan halaman data Pelanggan yang dipilih. 5. General Manager Operasional memilih status belanja dengan bantuan karyawan. 6. General Manager Operasional menekan tombol “<i>Submit</i>” 7. Sistem memproses perubahan data pelanggan. 8. Sistem menampilkan halaman data pelanggan dengan tabel daftar pelanggan yang telah diperbaharui
Aliran Alternatif	2a. Apabila tidak terdapat data Pelanggan, maka sistem akan menampilkan tabel data pelanggan kosong.
Kondisi Akhir	Sistem menampilkan halaman Data Pelanggan dengan data yang terbaru.

Tabel 4.28 Tabel Use Case Scenario Menghapus Data Pelanggan

Menghapus Data Pelanggan	
Tujuan	Sistem melakukan penghapusan data pelanggan pada sistem General Manager Operasional.
Aktor	General Manager Operasional
Prakondisi	General Manager Operasional berada pada <i>main page</i> sistem.
Aliran Utama	<ol style="list-style-type: none"> 1. General Manager Operasional memilih menu “Data Pelanggan”. 2. Sistem menampilkan halaman data pelanggan dengan adanya daftar Pelanggan. 3. Sistem menekan tombol “Hapus” pada data Pelanggan yang akan dilakukan dihapus. 4. Sistem menampilkan konfirmasi penghapusan data pelanggan. 5. General Manager Operasional memilih tombol “Ok”. 6. Sistem memproses penghapusan data pelanggan.

	7. Sistem menampilkan halaman data pelanggan dengan tabel daftar pelanggan yang telah diperbaharui
Aliran Alternatif	2a. Apabila tidak terdapat data Pelanggan, maka sistem akan menampilkan tabel data pelanggan kosong. 6a. Apabila General Manager Operasional memilih tombol <i>"Cancel"</i> , maka sistem akan menampilkan pesan <i>"Penghapusan data dibatalkan!"</i> .
Kondisi Akhir	Sistem menampilkan halaman Data Pelanggan dengan data yang terbaru.

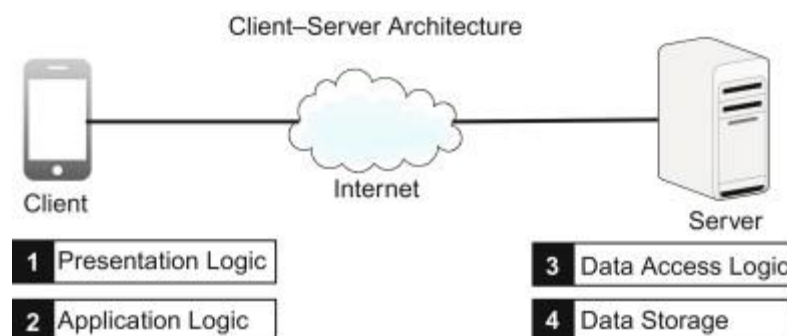
BAB 5 PEMBAHASAN

5.1 Perancangan

Tahapan selanjutnya setelah melakukan analisis kebutuhan sistem adalah melakukan perancangan sistem, yang mana dalam perancangan sistem ini, dilakukan perancangan antarmuka atau prototipe. Dasar tahapan perancangan sistem ini, diperoleh dari analisis kebutuhan sistem yang diperoleh pada tahapan sebelumnya, yang selanjutnya akan divisualisasikan dalam model *prototype*. Adapun hal – hal yang akan dibahas dalam tahapan perancangan sistem ini adalah *sequence diagram*, perancangan *class diagram*, perancangan *database*, perancangan algoritma, dan perancangan *interface*.

5.1.1 Perancangan Arsitektur

Pada perancangan arsitektur sistem pelayanan mandiri untuk toko ritel di Kota Malang ini terdapat dua bagian, yaitu bagian *client side* dan *server side*. Di *client side*, maka terdapat komponen – komponen tampilan *website*, interaksi pada *website* yang dilakukan oleh pengguna. Sedangkan, pada *server side*, terdapat *application server* yang menyimpan data – data HTML, CSS, dan Javascript, selain itu, terdapat *database server* yang menyimpan data – data yang akan ditampilkan pada *website*. Adapun visualisasi perancangan arsitektur sistem pelayanan mandiri untuk toko ritel di Kota Malang dapat dilihat pada Gambar 5.1.

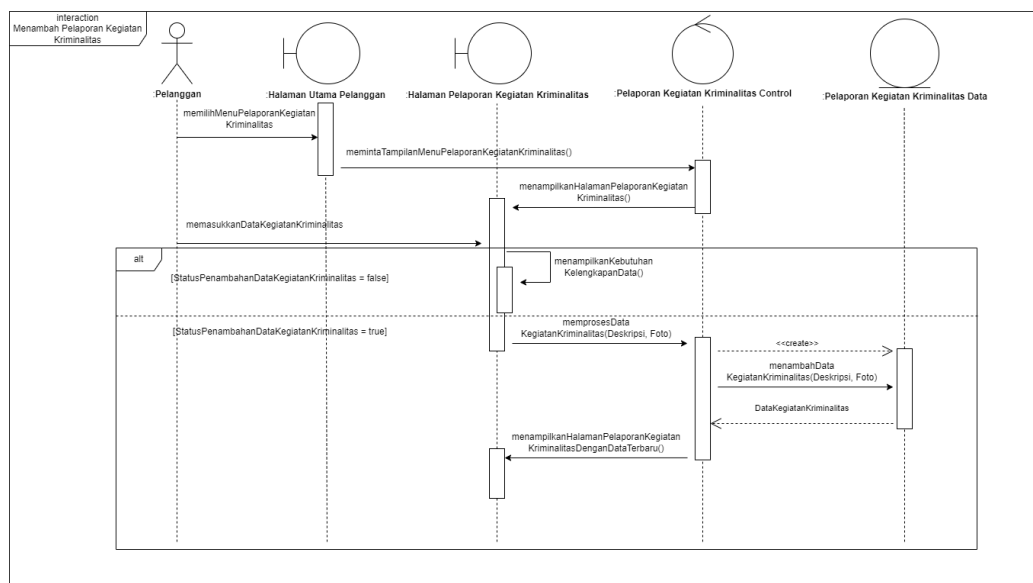


Gambar 5.1 Perancangan Arsitektur Sistem Pelayanan Mandiri Untuk Toko Ritel di Kota Malang

Dalam perancangan arsitektur ini, *design pattern* yang digunakan adalah MVC (*Model, View, Controller*), yaitu pola dalam desain perangkat lunak yang sering digunakan untuk mengimplementasikan antarmuka pengguna, data, dan logika kontroler. Arsitektur dari sistem ini, dibentuk dengan pola, ketika *client* melakukan *request*, maka *file index.php* akan dieksekusi, yang kemudian dilanjutkan kepada *file router.php*, apabila dalam *file* tersebut tidak memiliki *cache*, maka akan dilanjutkan kepada kontroler, dan terakhir kontroler akan mengambil data dari *model* dan menampilkan isi dari *view* kepada *client*.

membayarkan barang – barang belanjanya dengan menekan ikon keranjang, maka sistem akan menampilkan halaman keranjang dengan daftar barang – barang belanja yang sudah dipilih, setelah itu, pelanggan dapat memilih metode pembayaran untuk membayar barang belanja yang dipilih, lalu memilih tombol bayar untuk melanjutkan proses pembayaran, apabila saldo pada metode pembayaran kurang, maka sistem akan menampilkan pesan saldo kurang, namun apabila saldo metode pembayaran cukup, maka sistem akan memproses pembayaran, dan setelah proses pembayaran selesai, sistem akan menampilkan resi belanja.

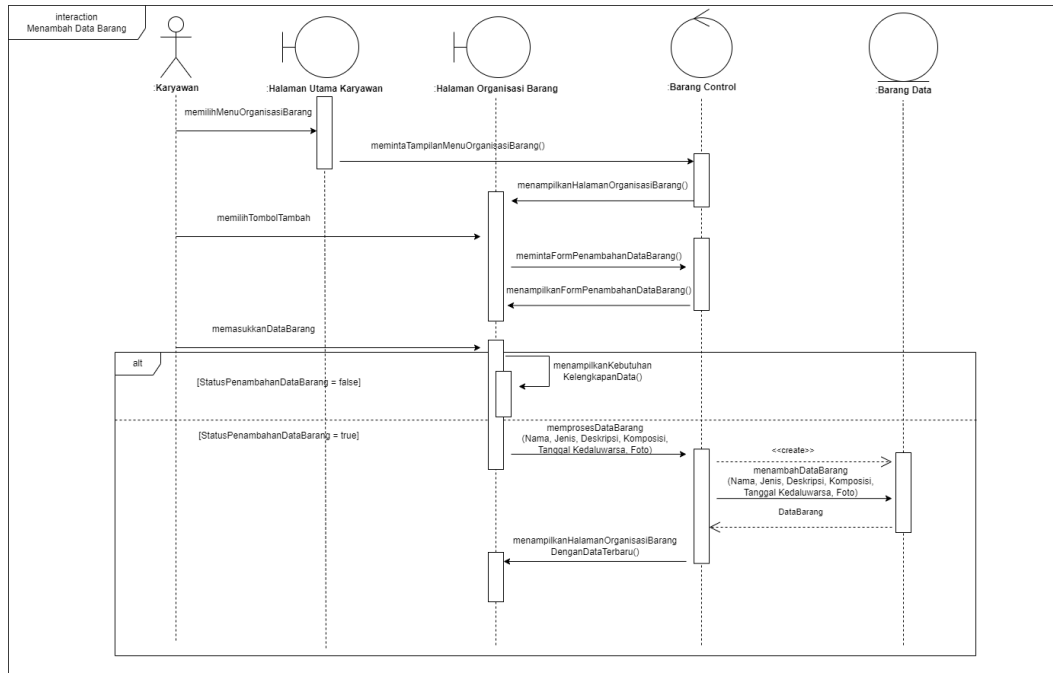
5.1.2.2 Sequence Diagram Menambah Pelaporan Kegiatan Kriminalitas



Gambar 5.3 Sequence Diagram Menambah Pelaporan Kegiatan Kriminalitas

Sequence Diagram diatas merupakan *sequence diagram* yang menjelaskan alur kerja dari fitur menambah pelaporan kegiatan kriminalitas, yang mana alur dimulai dengan pelanggan memilih menu pelaporan kegiatan kriminalitas pada halaman utama pelanggan, kemudian pada halaman pelaporan kegiatan kriminalitas, pelanggan dapat memasukkan data kegiatan kriminalitas yang akan dilaporkan yang berisikan deskripsi dan foto. Apabila data yang dimasukkan belum lengkap, maka sistem akan menampilkan pesan untuk meminta pelanggan melengkapi data yang belum lengkap, akan tetapi, jika data sudah lengkap, maka sistem akan memproses masukan data yang diberikan oleh pelanggan, dan kemudian sistem akan menampilkan halaman pelaporan kegiatan kriminalitas dengan data yang sudah terbaharui.

5.1.2.3 Sequence Diagram Menambah Data Barang



Gambar 5.4 Sequence Diagram Menambah Data Barang

Sequence Diagram diatas merupakan *sequence diagram* yang menjelaskan alur kerja dari fitur menambah data barang, yang mana alur dimulai dengan karyawan memilih menu organisasi barang, dilanjutkan dengan karyawan memilih tombol tambah untuk melakukan penambahan data barang. Kemudian, karyawan memasukkan data barang yang terdiri dari nama, jenis, deskripsi, komposisi, tanggal kedaluwarsa, dan foto barang, apabila terdapat data yang belum lengkap, maka sistem akan meminta karyawan untuk melengkapi kekurangan data, akan tetapi, jika data barang sudah lengkap, maka sistem akan memproses penambahan data, setelah itu, sistem akan menampilkan halaman organisasi barang dengan data yang terbaru.

5.1.3 Class Diagram

Pada *class diagram* ini dijelaskan gambaran dari struktur kelas yang ada pada sistem, yang mana struktur perancangan menggunakan konsep *framework MVC (Model, View, Controller)*. Kelas model digunakan sebagai perantara sistem dengan basis data, sedangkan kelas kontroler digunakan sebagai perantara model dengan kelas *view*, dan kelas *view* digunakan sebagai antarmuka yang berhubungan langsung dengan pengguna sistem untuk menangani interaksi pengguna. Adapun perancangan diagram kelas dapat dilihat pada Gambar 5.5.

5.1.4 Perancangan *Database*

Adapun perancangan *database* dari sistem pelayanan mandiri untuk toko ritel di Kota Malang dibentuk dengan diagram ERD (*Entity Relational Diagram*) pada Gambar 5.6

Berdasarkan pada Gambar 5.6, maka berikut merupakan detail dari setiap entitasnya yaitu:

A. Detail Entitas *General Manager* Operasional

Tabel 5.1 Tabel Detail Entitas *General Manager* Operasional

Atribut	Tipe Data	Fungsi
<u>id_generalmanageroperasional</u>	Int(10)	Primary key
email	Varchar(70)	
nama	Varchar(70)	
nomor_telepon	Varchar(70)	
username	Varchar(70)	
password	Varchar(70)	
jabatan	Varchar(70)	
status	Varchar(70)	
gambar	Varchar(70)	

B. Detail Entitas Karyawan

Tabel 5.2 Tabel Detail Entitas Karyawan

Atribut	Tipe Data	Fungsi
<u>id_karyawan</u>	Int(10)	Primary key
email	Varchar(70)	
nama	Varchar(70)	
nomor_telepon	Varchar(70)	
username	Varchar(70)	
password	Varchar(70)	
jabatan	Varchar(70)	
status	Varchar(70)	
status_belanja_dengan_bantuan_karyawan	Varchar(70)	
gambar	Varchar(70)	

C. Detail Entitas Pelanggan

Tabel 5.3 Tabel Detail Entitas Pelanggan

Atribut	Tipe Data	Fungsi
<u>id_pelanggan</u>	Int(10)	Primary key
email	Varchar(70)	
nama	Varchar(70)	
nomor_telepon	Varchar(70)	
username	Varchar(70)	
password	Varchar(70)	
status	Varchar(70)	
status_belanja_dengan_bantuan_karyawan	Varchar(70)	
gambar	Varchar(70)	

D. Detil Entitas Barang

Tabel 5.4 Tabel Detil Entitas Barang

Atribut	Tipe Data	Fungsi
<u>id_barang</u>	Int(10)	Primary key
namabarang	Varchar(100)	
jenisbarang	Varchar(70)	
harga	Int(10)	
deskripsi	Varchar(500)	
komposisi	Varchar(500)	
tanggalkedaluwarsa	Varchar(70)	
foto	Varchar(500)	
jumlahstokbarang	Int(10)	
kategori_id	Int(10)	

E. Detil Entitas Belanja Mandiri

Tabel 5.5 Tabel Detil Belanja Mandiri

Atribut	Tipe Data	Fungsi
<u>id_belanjamandiri</u>	Int(10)	Primary key
namabarang	Varchar(100)	
jumlahbarang	Int(10)	

metodepembayaran	Varchar(100)	
statuspembayaran	Varchar(30)	

F. Detil Entitas Belanja dengan Bantuan Karyawan

Tabel 5.6 Tabel Detil Belanja dengan Bantuan Karyawan

Atribut	Tipe Data	Fungsi
<u>id_belanjadenganbantuankaryawan</u>	Int(10)	Primary key
namabarang	Varchar(100)	
jumlahbarang	Int(10)	
metodepembayaran	Varchar(100)	
statuspembayaran	Varchar(30)	

G. Detil Entitas Pelaporan Kegiatan Kriminalitas

Tabel 5.7 Tabel Detil Pelaporan Kegiatan Kriminalitas

Atribut	Tipe Data	Fungsi
<u>id_pelaporankegiatankriminalitas</u>	Int(10)	Primary key
username	Varchar(70)	
deskripsi	Varchar(500)	
foto	Varchar(500)	
statuspelaporan	Varchar(30)	

H. Detil Entitas Riwayat Belanja

Tabel 5.8 Tabel Detil Riwayat Belanja

Atribut	Tipe Data	Fungsi
<u>id_riwayatbelanja</u>	Int(10)	Primary key
id_pelanggan	Int(10)	
username	Varchar(70)	
totalbelanja	int (30)	
tanggaltransaksi	Timestamp	

I. Detil Entitas Metode Pembayaran

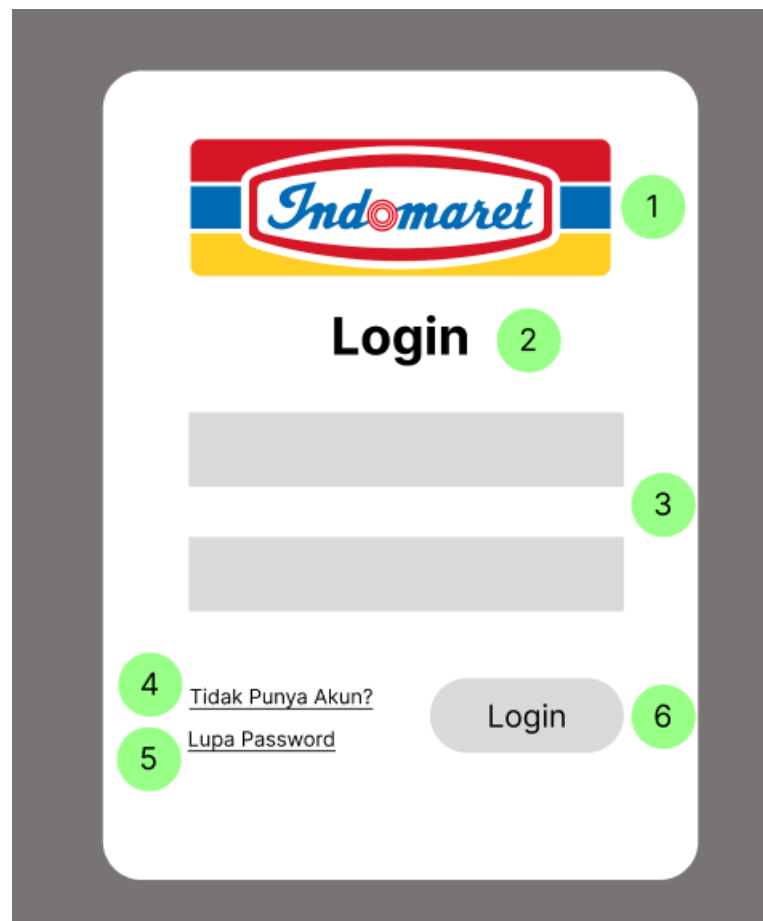
Tabel 5.9 Tabel Detil Metode Pembayaran

Atribut	Tipe Data	Fungsi
<u>id_metodepembayaran</u>	Int(10)	Primary key
namametodepembayaran	Varchar(100)	
saldometodepembayaran	int (30)	

5.1.5 Perancangan Antarmuka

5.1.5.1 Perancangan Antarmuka *Login*

Berikut merupakan perancangan dari antarmuka *login* yang dapat dilihat pada Gambar 5.7.



Gambar 5.7 Perancangan Antarmuka *Login*

Adapun detail dari perancangan antarmuka *login* yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.

2. Nomor 2 melambangkan nama halaman yang diakses, dalam hal ini halaman *login*.
3. Nomor 3 melambangkan form *login* untuk melakukan otorisasi pengguna.
4. Nomor 4 melambangkan *link* untuk mengarahkan pengguna ke halaman *register*, apabila pengguna tidak memiliki akun yang terdaftar pada sistem.
5. Nomor 5 melambangkan *link* untuk mengarahkan pengguna ke halaman lupa *password*, sehingga pengguna dapat memperoleh *password* sementara untuk akun yang dimiliki.
6. Nomor 6 melambangkan tombol *login* untuk memproses otorisasi *username* dan *password* yang dimasukkan oleh pengguna.

5.1.5.2 Perancangan Antarmuka *Register*

Berikut merupakan perancangan dari antarmuka *register* yang dapat dilihat pada Gambar 5.8.

The image shows a mobile app wireframe for the 'Indomaret Register' page. At the top is the Indomaret logo, labeled with a green circle containing the number 1. Below the logo is the title 'Register', labeled with a green circle containing the number 2. The main area contains seven horizontal input fields for registration details, with the third field labeled by a green circle containing the number 3. At the bottom left is a link 'Punya Akun?' labeled with a green circle containing the number 4. To its right is a 'Register' button, labeled with a green circle containing the number 5. The entire interface is set against a dark gray background.

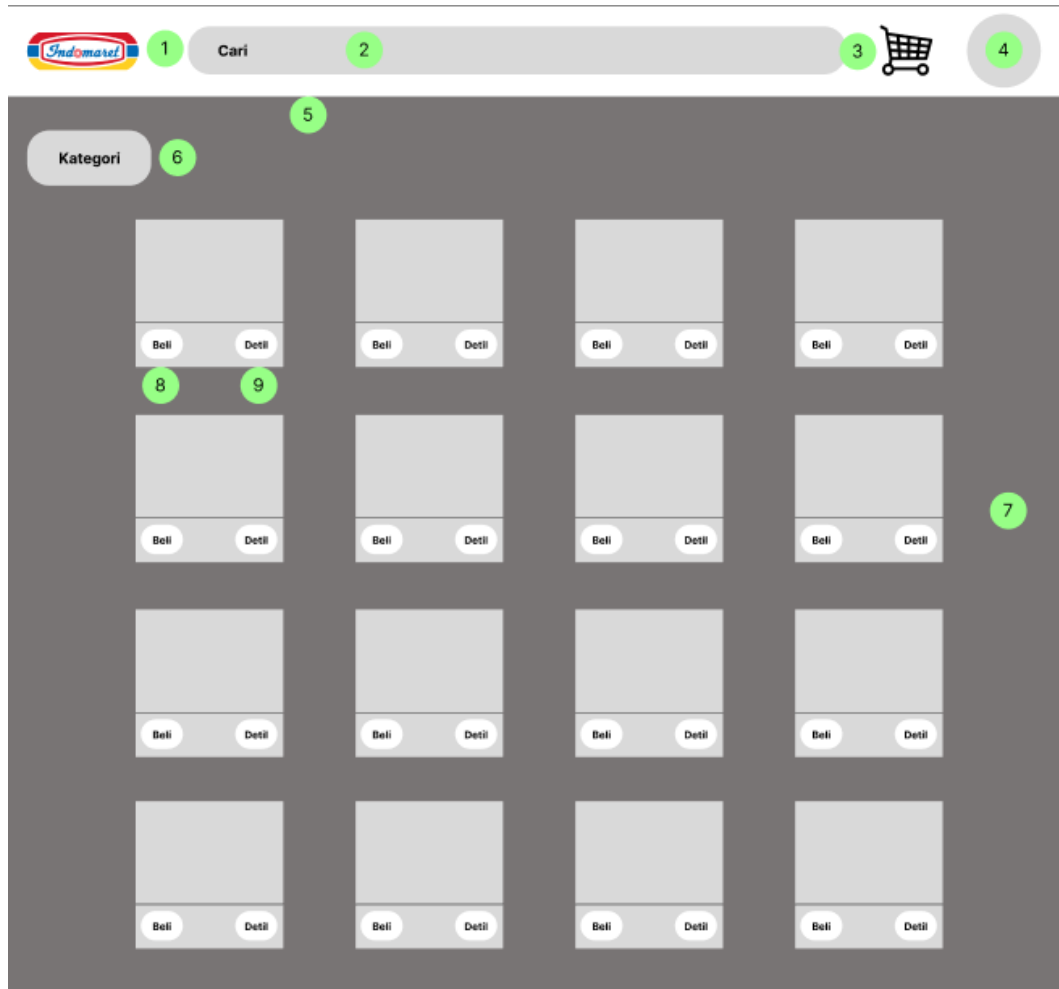
Gambar 5.8 Perancangan Antarmuka *Register*

Adapun detail dari perancangan antarmuka *register* yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan nama halaman yang diakses, dalam hal ini halaman *register*.
3. Nomor 3 melambangkan form *register* untuk melakukan pendaftaran akun pengguna.
4. Nomor 4 melambangkan *link* untuk mengarahkan pengguna ke halaman *register*, apabila pengguna tidak memiliki akun yang terdaftar pada sistem.
5. Nomor 5 melambangkan *link* untuk mengarahkan pengguna ke halaman lupa *password*, sehingga pengguna dapat memperoleh *password* sementara untuk akun yang dimiliki.

5.1.5.3 Perancangan Antarmuka Melakukan Belanja Mandiri/Melakukan Belanja dengan Bantuan Karyawan

Berikut merupakan perancangan dari antarmuka Melakukan Belanja Mandiri/Melakukan Belanja dengan Bantuan Karyawan yang dapat dilihat pada Gambar 5.9.



Gambar 5.9 Perancangan Antarmuka Melakukan Belanja Mandiri/Melakukan Belanja dengan Bantuan Karyawan

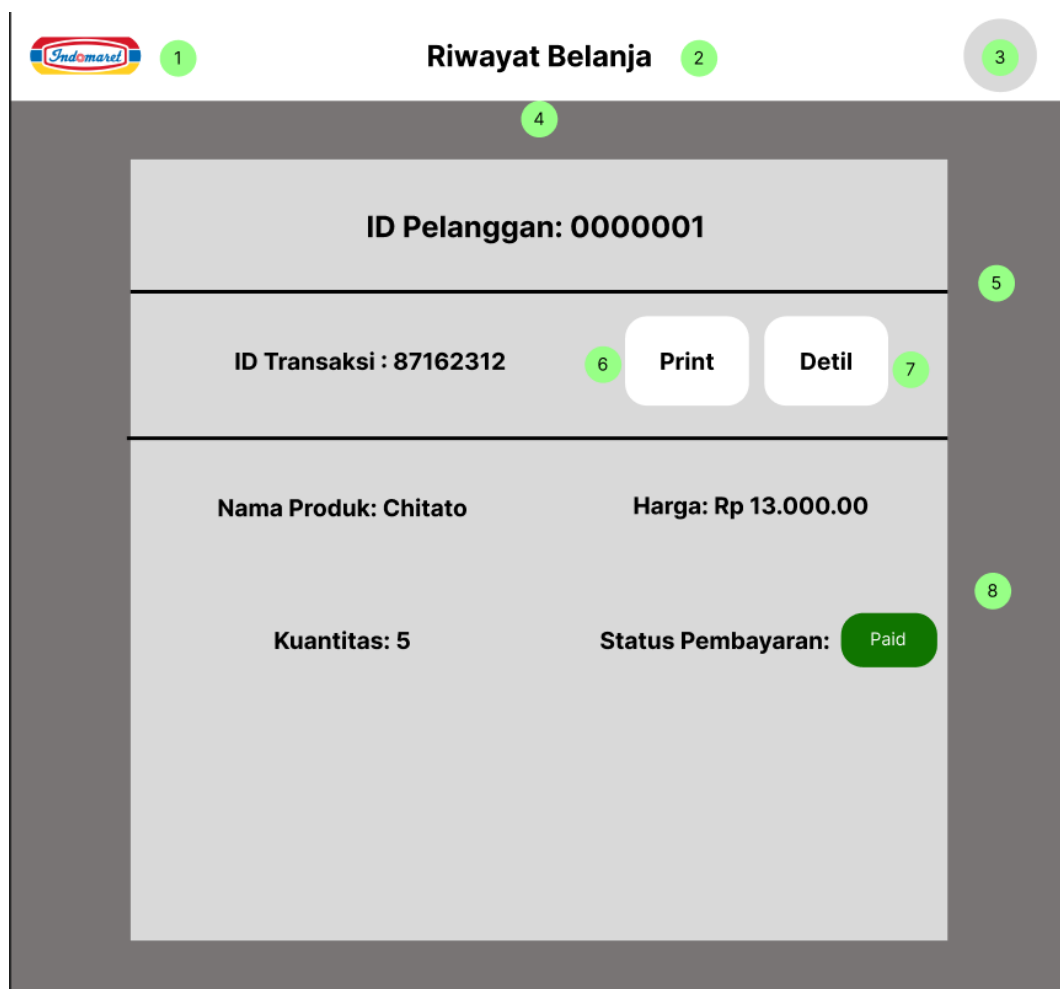
Adapun detail dari perancangan antarmuka Melakukan Belanja Mandiri/Melakukan Belanja dengan Bantuan Karyawan yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan *search bar* untuk melakukan pencarian barang.
3. Nomor 3 melambangkan ikon keranjang untuk mengakses halaman keranjang.
4. Nomor 4 melambangkan ikon profil pengguna/karyawan.
5. Nomor 5 melambangkan *navbar* sistem.

6. Nomor 6 melambangkan tombol kategori untuk mengakses halaman kategori barang.
7. Nomor 7 melambangkan daftar semua barang yang terdaftar dalam sistem.
8. Nomor 8 melambangkan tombol beli, untuk menambah barang ke dalam keranjang.
9. Nomor 9 melambangkan tombol detail untuk mengakses detail barang.

5.1.5.4 Perancangan Antarmuka Menampilkan Riwayat Belanja

Berikut merupakan perancangan dari antarmuka Menampilkan Riwayat Belanja yang dapat dilihat pada Gambar 5.10.



Gambar 5.10 Perancangan Antarmuka Menampilkan Riwayat Belanja

Adapun detail dari perancangan antarmuka Menampilkan Riwayat Belanja yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan nama halaman yang diakses

3. Nomor 3 melambangkan ikon profil pengguna.
4. Nomor 4 melambangkan *navbar* sistem.
5. Nomor 5 melambangkan form riwayat belanja dari pengguna.
6. Nomor 6 melambangkan tombol *print* untuk mencetak riwayat belanja.
7. Nomor 7 melambangkan tombol *view* untuk melihat detil riwayat belanja.
8. Nomor 8 melambangkan data – data riwayat belanja dari id transaksi yang tertera, dengan berisikan nama produk, harga, kuantitas, dan status pembayaran produk tersebut.

5.1.5.5 Perancangan Antarmuka Menambah Pelaporan Kegiatan Kriminalitas

Berikut merupakan perancangan dari antarmuka Menambah Pelaporan Kegiatan Kriminalitas yang dapat dilihat pada Gambar 5.11.

The wireframe shows a web interface for reporting criminal activities. At the top, there is a header bar containing a logo (1), the title 'Pelaporan Kegiatan Kriminalitas' (2), and a user profile icon (3). The main content area (4) features a form with a 'Deskripsi' label (5) above a text input field, a 'Foto' label above another text input field, and a 'Submit' button (6). Below the form is a table with three columns: 'Deskripsi', 'Deskripsi', and 'Status Laporan'.

Deskripsi	Deskripsi	Status Laporan

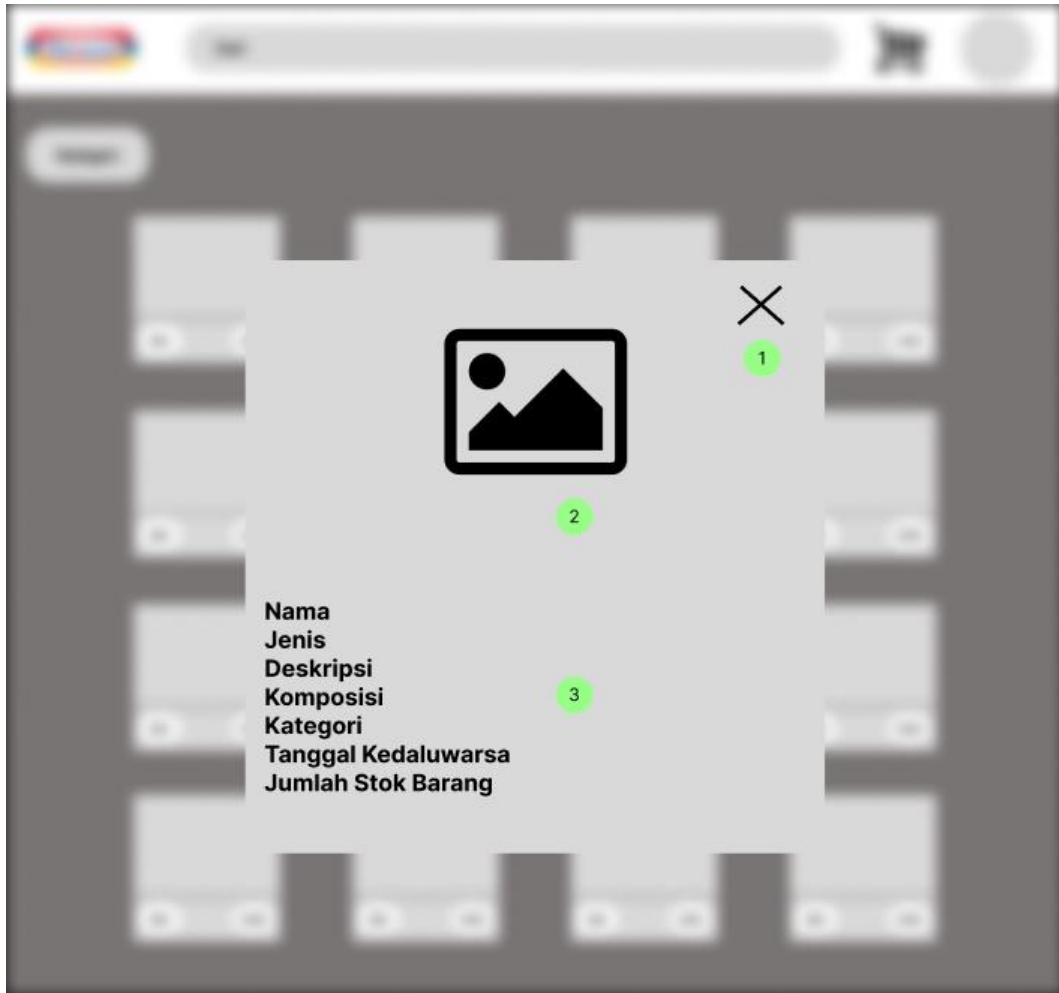
Gambar 5.11 Perancangan Antarmuka Menambah Pelaporan Kegiatan Kriminalitas

Adapun detil dari perancangan antarmuka Menambah Pelaporan Kegiatan Kriminalitas yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan nama halaman yang diakses
3. Nomor 3 melambangkan ikon profil pengguna.
4. Nomor 4 melambangkan *navbar* sistem.
5. Nomor 5 melambangkan form pelaporan kegiatan kriminalitas yang berisikan form untuk menambah laporan dan daftar laporan yang sudah dilakukan pengguna.
6. Nomor 6 melambangkan tombol *submit* untuk mengirimkan laporan pada sistem.

5.1.5.6 Perancangan Antarmuka Menampilkan Detil Barang Belanja/Menampilkan Detil Barang

Berikut merupakan perancangan dari antarmuka Menampilkan Detil Barang Belanja/Menampilkan Detil Barang yang dapat dilihat pada Gambar 5.12.



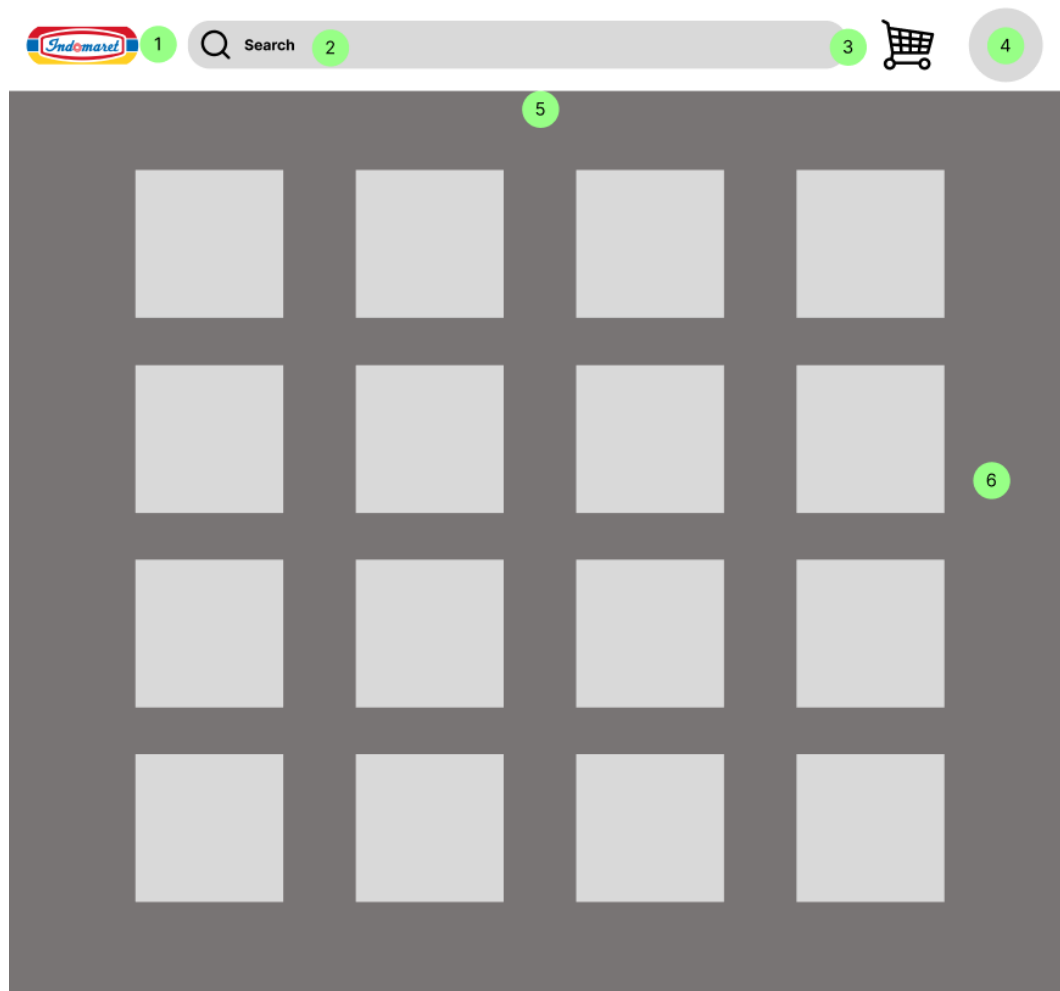
Gambar 5.12 Perancangan Antarmuka Menampilkan Detil Barang Belanja/Menampilkan Detil Barang

Adapun detil dari perancangan antarmuka Menampilkan Detil Barang Belanja/Menampilkan Detil Barang yaitu:

1. Nomor 1 melambangkan ikon 'X' untuk menutup *pop up*.
2. Nomor 2 melambangkan gambar dari produk barang yang dipilih.
3. Nomor 3 melambangkan detil dari barang yang dilihat.

5.1.5.7 Perancangan Antarmuka Menampilkan Kategori Barang Belanja/Menampilkan Kategori Barang

Berikut merupakan perancangan dari antarmuka Menampilkan Kategori Barang Belanja/Menampilkan Kategori Barang yang dapat dilihat pada Gambar 5.13.



Gambar 5.13 Perancangan Antarmuka Menampilkan Kategori Barang Belanja/Menampilkan Kategori Barang

Adapun detail dari perancangan antarmuka Menampilkan Kategori Barang Belanja/Menampilkan Kategori Barang yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan *search bar* untuk mencari kategori barang yang diinginkan.
3. Nomor 3 melambangkan ikon keranjang untuk mengakses keranjang belanja.
4. Nomor 4 melambangkan ikon profil pengguna/karyawan.
5. Nomor 5 melambangkan *navbar* sistem.

6. Nomor 6 melambangkan daftar kategori barang.

5.1.5.8 Perancangan Antarmuka Menampilkan Keranjang Belanja

Berikut merupakan perancangan dari antarmuka Menampilkan Keranjang Belanja yang dapat dilihat pada Gambar 5.14.



Gambar 5.14 Perancangan Antarmuka Menampilkan Keranjang Belanja

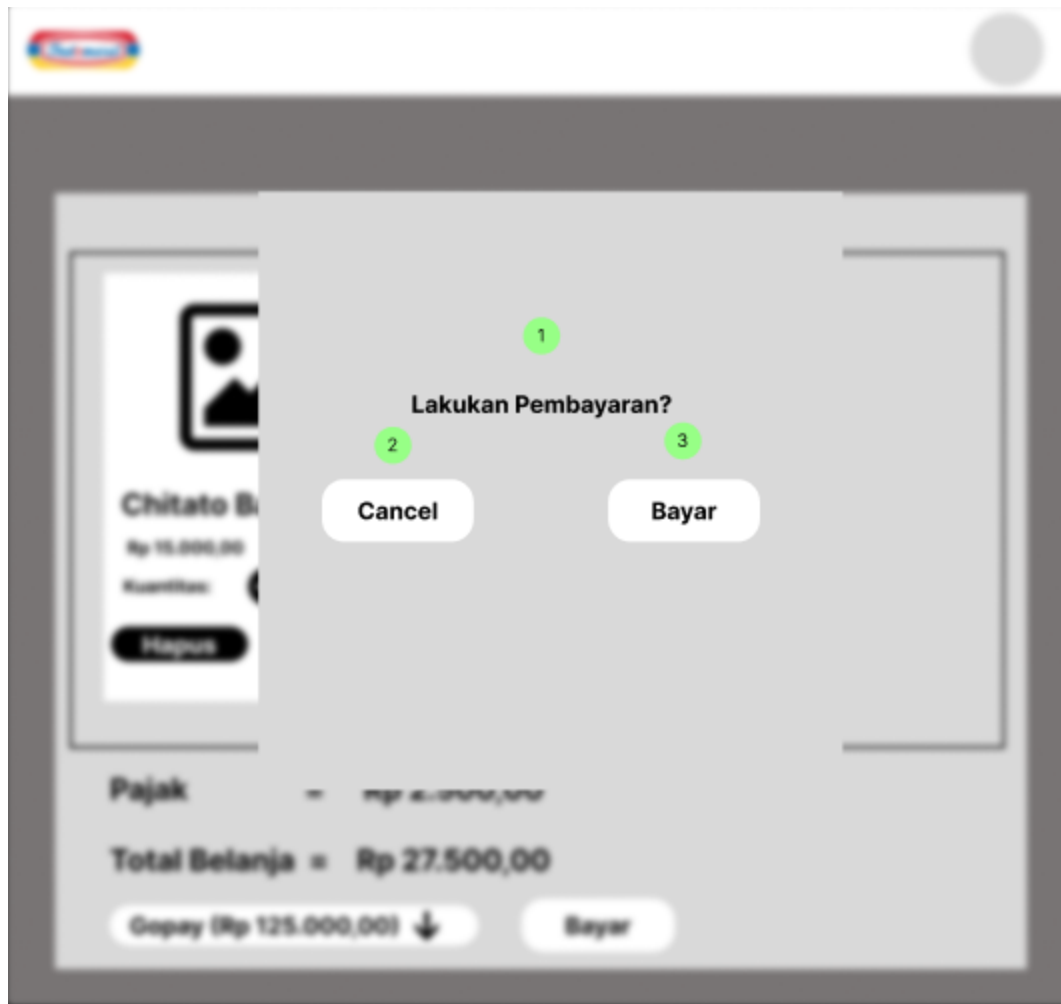
Adapun detail dari perancangan antarmuka Menampilkan Keranjang Belanja yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan ikon profil pengguna.
3. Nomor 3 melambangkan *navbar* sistem.
4. Nomor 4 melambangkan daftar barang dalam keranjang.
5. Nomor 5 melambangkan tombol “Hapus”, untuk menghapus barang dari keranjang.

6. Nomor 6 melambangkan ikon '-' untuk mengurangi jumlah barang sebanyak 1.
7. Nomor 7 melambangkan ikon '+' untuk menambah jumlah barang sebanyak 1.
8. Nomor 8 melambangkan pilihan metode pembayaran.
9. Nomor 9 melambangkan tombol bayar, untuk membayarkan barang belanja.

5.1.5.9 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja

Berikut merupakan perancangan dari antarmuka Melakukan Pembayaran Barang Belanja yang dapat dilihat pada Gambar 5.15, Gambar 5.16, Gambar 5.17.



Gambar 5.15 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja



Gambar 5.16 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja
(2)



Gambar 5.17 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja (3)

Adapun detil dari perancangan antarmuka Melakukan Pembayaran Barang Belanja yaitu:

1. Nomor 1 melambangkan pertanyaan konfirmasi.
2. Nomor 2 melambangkan tombol *cancel* untuk membatalkan pembayaran.
3. Nomor 3 melambangkan tombol bayar, untuk mengkonfirmasi pembayaran.
4. Nomor 4 melambangkan status pembayaran yang berhasil.
5. Nomor 5 melambangkan status pembayaran yang gagal.

5.1.5.10 Perancangan Antarmuka Menampilkan Keranjang Belanja dengan Bantuan Karyawan

Berikut merupakan perancangan dari antarmuka Menampilkan Keranjang Belanja dengan Bantuan Karyawan yang dapat dilihat pada Gambar 5.18.



Gambar 5.18 Perancangan Antarmuka Menampilkan Keranjang Belanja dengan Bantuan Karyawan

Adapun detail dari perancangan antarmuka Menampilkan Keranjang Belanja dengan Bantuan Karyawan yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan ikon profil karyawan.
3. Nomor 3 melambangkan *navbar* sistem.
4. Nomor 4 melambangkan daftar barang dalam keranjang.
5. Nomor 5 melambangkan tombol “Hapus”, untuk menghapus barang dari keranjang.

6. Nomor 6 melambangkan ikon '-' untuk mengurangi jumlah barang sebanyak 1.
7. Nomor 7 melambangkan ikon '+' untuk menambah jumlah barang sebanyak 1.
8. Nomor 8 melambangkan tombol bayar, untuk membayarkan barang belanja.

5.1.5.11 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja dengan Bantuan Karyawan

Berikut merupakan perancangan dari antarmuka Melakukan Pembayaran Barang Belanja dengan Bantuan Karyawan yang dapat dilihat pada Gambar 5.19, Gambar 5.20 dan Gambar 5.21.



Gambar 5.19 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja dengan Bantuan Karyawan



Gambar 5.20 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja dengan Bantuan Karyawan (2)



Gambar 5.21 Perancangan Antarmuka Melakukan Pembayaran Barang Belanja dengan Bantuan Karyawan (3)

Adapun detail dari perancangan antarmuka Melakukan Pembayaran Barang Belanja dengan Bantuan Karyawan yaitu:

1. Nomor 1 melambangkan nama dari kode QR yang ditampilkan.
2. Nomor 2 melambangkan kode QR pembayaran.
3. Nomor 3 melambangkan kode *virtual account* untuk pelanggan yang membayar melalui transfer.
4. Nomor 4 melambangkan estimasi waktu pembayaran.
5. Nomor 5 melambangkan ikon 'X' untuk menutup *pop up* pembayaran.
6. Nomor 6 melambangkan status pembayaran berhasil.
7. Nomor 7 melambangkan status pembayaran gagal.

5.1.5.12 Perancangan Antarmuka Menambah Data Barang dan Menampilkan Daftar Data Barang

Berikut merupakan perancangan dari antarmuka Menambah Data Barang dan Menampilkan Daftar Data Barang yang dapat dilihat pada Gambar 5.22.

The interface is titled "Data Barang". It features a form for adding new goods and a table for displaying existing goods.

Form Fields:

- 1. Logo of Indomaret
- 2. Page title "Data Barang"
- 3. ID Barang
- 4. Nama Barang
- 5. Jenis Barang
- 6. Harga (Rp)
- 7. Deskripsi
- 8. Komposisi
- 9. Tanggal Kedaluwarsa
- 10. Jumlah Stok Barang
- 11. Foto
- 12. Kategori
- 13. Submit button

Table:

ID Barang	Nama Barang	Jenis Barang	Harga	Deskripsi	Komposisi	Tanggal Kedaluwarsa	Jumlah Stok Barang	Gambar	Kategori	Aksi
11111	Soklin									<div>Edit</div> <div>Hapus</div>

Gambar 5.22 Perancangan Antarmuka Menambah Data Barang dan Menampilkan Daftar Data Barang

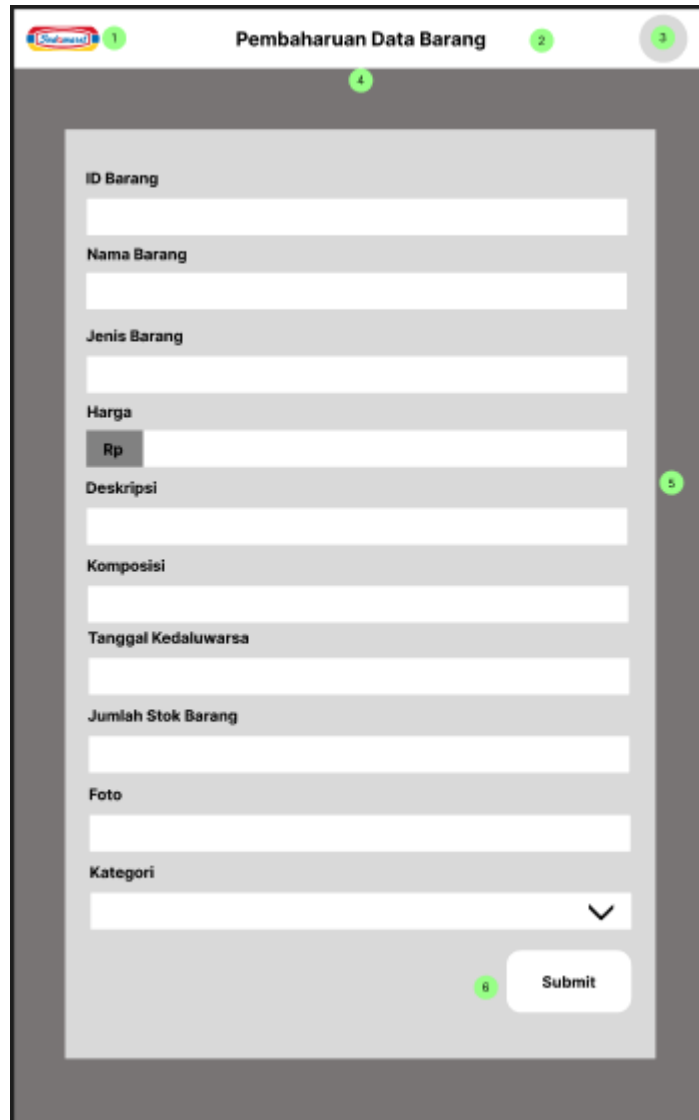
Adapun detail dari perancangan antarmuka Menambah Data Barang dan Menampilkan Daftar Data Barang yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan nama halaman yang diakses.

3. Nomor 3 melambangkan ikon profil karyawan.
4. Nomor 4 melambangkan *navbar* sistem.
5. Nomor 5 melambangkan form penambahan barang.
6. Nomor 6 melambangkan tombol *submit* untuk menambahkan barang pada sistem.
7. Nomor 7 melambangkan tabel barang yang terdaftar dalam sistem.
8. Nomor 8 melambangkan tombol *Edit* untuk melakukan perubahan data barang.
9. Nomor 9 melambangkan tombol *Hapus* untuk melakukan penghapusan data barang dari sistem.

5.1.5.13 Perancangan Antarmuka Mengubah Data Barang

Berikut merupakan perancangan dari antarmuka Mengubah Data Barang yang dapat dilihat pada Gambar 5.23.



The screenshot shows a mobile application interface for updating item data. At the top, there is a header bar with a logo (1), the title 'Pembaharuan Data Barang' (2), and a profile icon (3). Below the header is a dark grey sidebar (4) containing a list of items. The main content area (5) is a form with the following fields: 'ID Barang', 'Nama Barang', 'Jenis Barang', 'Harga' (with a 'Rp' prefix), 'Deskripsi', 'Komposisi', 'Tanggal Kedaluwarsa', 'Jumlah Stok Barang', 'Foto', and 'Kategori' (a dropdown menu). A 'Submit' button (6) is located at the bottom right of the form.

Gambar 5.23 Perancangan Antarmuka Mengubah Data Barang

Adapun detail dari perancangan antarmuka Mengubah Data Barang yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan nama halaman yang diakses.
3. Nomor 3 melambangkan ikon profil karyawan.
4. Nomor 4 melambangkan *navbar* sistem.
5. Nomor 5 melambangkan form perubahan data barang.
6. Nomor 6 melambangkan tombol *submit* untuk menambahkan perubahan data barang pada sistem.

5.1.5.14 Perancangan Antarmuka Menghapus Data Barang

Berikut merupakan perancangan dari antarmuka Menghapus Data Barang yang dapat dilihat pada Gambar 5.24, Gambar 5.25, dan Gambar 5.26.

The screenshot shows a web application interface for managing products. At the top, there is a header bar with a logo on the left and the text "Data Barang" in the center. Below the header, there is a form for adding or editing a product. The form has the following fields: "ID Barang", "Nama Barang", "Jenis Barang", "Harga" (with a currency symbol "Rp"), "Deskripsi", "Komposisi", "Tanggal Kadaluarsa", "Jumlah Stok Barang", "Foto", and "Kategori" (with a dropdown arrow). A "Submit" button is located at the bottom right of the form. In the center of the screen, a modal dialog is displayed with the text "Apakah Anda Yakin Menghapus Barang 101?". The dialog has two buttons: "Cancel" and "Hapus". The "Hapus" button is highlighted with a green circle and a green number 2. The "Cancel" button is highlighted with a green circle and a green number 3. The "Hapus" button is also highlighted with a green circle and a green number 1. Below the form, there is a table with the following columns: "ID Barang", "Nama Barang", "Jenis Barang", "Harga", "Deskripsi", "Komposisi", "Tanggal Kadaluarsa", "Jumlah Stok Barang", "Gambar", "Kategori", and "Aksi". The first row of the table shows "11111" in the "ID Barang" column, "Suklin" in the "Nama Barang" column, and "Edit" and "Hapus" buttons in the "Aksi" column.

ID Barang	Nama Barang	Jenis Barang	Harga	Deskripsi	Komposisi	Tanggal Kadaluarsa	Jumlah Stok Barang	Gambar	Kategori	Aksi
11111	Suklin									Edit Hapus

Gambar 5.24 Perancangan Antarmuka Menghapus Data Barang


Data Barang

ID Barang

Nama Barang

Jenis Barang

Harga

Rp

Deskripsi

Komponen

Tanggal Ka

Jumlah Stok barang

Foto

Kategori

Submit

4

Data Berhasil Dihapus

ID Barang	Nama Barang	Jenis Barang	Harga	Deskripsi	Komponen	Tanggal Bedaformase	Jumlah Stok Barang	Gambar	Kategori	Aksi
11111	Seklon									<div>Edit</div> <div>Hapus</div>

Gambar 5.25 Perancangan Antarmuka Menghapus Data Barang (2)

Data Barang

ID Barang
 Nama Barang
 Jenis Barang
 Harga
 Rp
 Deskripsi
 Komposisi
 Tanggal K
 Jumlah St
 Foto
 Kategori
 Submit

Penghapusan Data Dibatalkan

ID Barang	Nama Barang	Jenis Barang	Harga	Deskripsi	Komposisi	Tanggal Kadaluarsa	Jumlah Stok Barang	Gambar	Kategori	Aksi
11111	Sukin									Edit Hapus

Gambar 5.26 Perancangan Antarmuka Menghapus Data Barang (3)

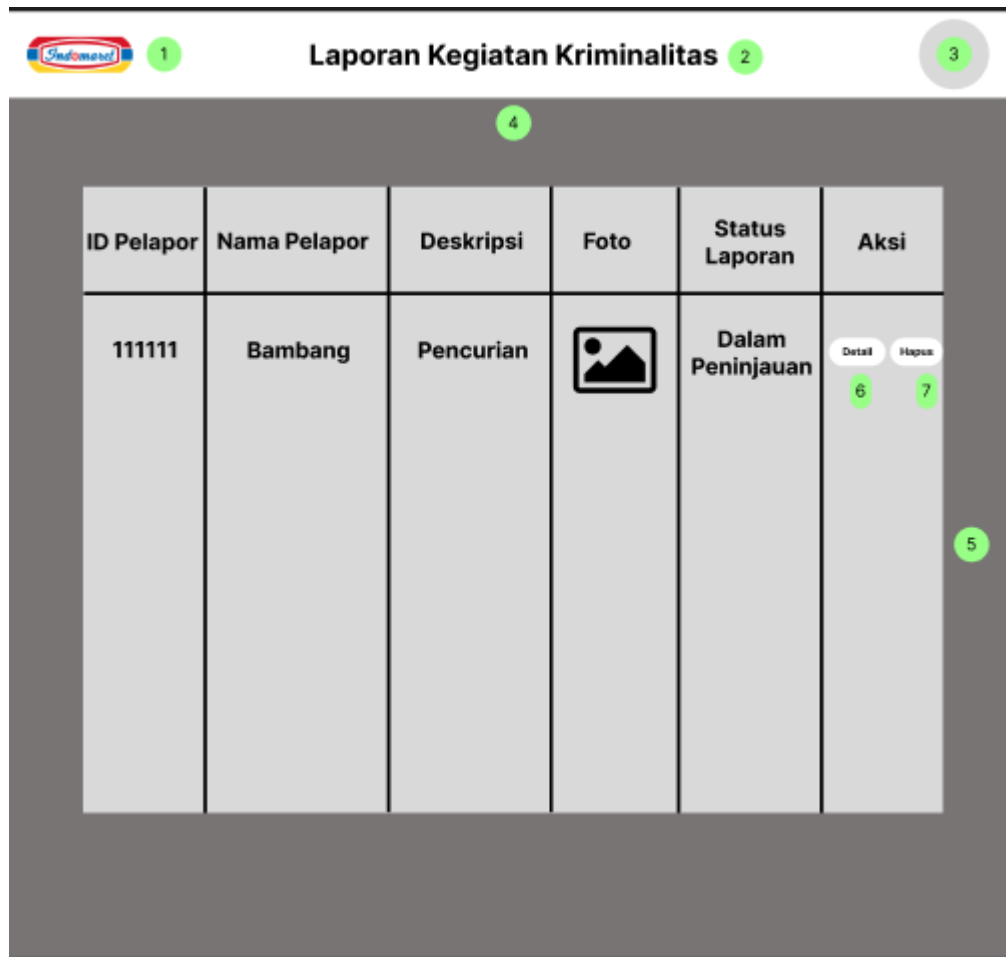
Adapun detil dari perancangan antarmuka Menghapus Data Barang yaitu:

1. Nomor 1 melambangkan konfirmasi penghapusan data barang.
2. Nomor 2 melambangkan tombol *cancel* untuk membatalkan penghapusan data barang.
3. Nomor 3 melambangkan tombol hapus untuk mengkonfirmasi penghapusan data barang.
4. Nomor 4 melambangkan status penghapusan data barang yang berhasil.

5. Nomor 5 melambangkan status penghapusan data barang yang dibatalkan.

5.1.5.15 Perancangan Antarmuka Menampilkan Daftar Laporan Tindakan Kriminalitas

Berikut merupakan perancangan dari antarmuka Menampilkan Daftar Laporan Tindakan Kriminalitas yang dapat dilihat pada Gambar 5.27.



Gambar 5.27 Perancangan Antarmuka Menampilkan Daftar Laporan Tindakan Kriminalitas

Adapun detil dari perancangan antarmuka Menampilkan Daftar Laporan Tindakan Kriminalitas yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan nama halaman yang diakses.
3. Nomor 3 melambangkan ikon profil karyawan.
4. Nomor 4 melambangkan *navbar* sistem.
5. Nomor 5 melambangkan daftar laporan tindakan kriminalitas.

6. Nomor 6 melambangkan tombol *Detail* untuk melihat detail laporan.
7. Nomor 7 melambangkan tombol Hapus, untuk menghapus laporan tindakan kriminalitas.

5.1.5.16 Perancangan Antarmuka Menampilkan Data Laporan Tindakan Kriminalitas

Berikut merupakan perancangan dari antarmuka Menampilkan Data Laporan Tindakan Kriminalitas yang dapat dilihat pada Gambar 5.28.

The screenshot shows a web form titled "Laporan Kegiatan Kriminalitas". At the top left is the Indomaret logo (1). The title "Laporan Kegiatan Kriminalitas" is at the top center (2). A grey circle with the number 3 is at the top right. The form contains several input fields: "ID Pelapor", "Nama Pelapor", and "Deskripsi" (4). Below these is a "Foto" section with a placeholder image icon. At the bottom of the form is a "Status Laporan" dropdown menu (5) currently set to "Dalam Peninjauan", and an "Update" button (6).

Gambar 5.28 Perancangan Antarmuka Menampilkan Data Laporan Tindakan Kriminalitas

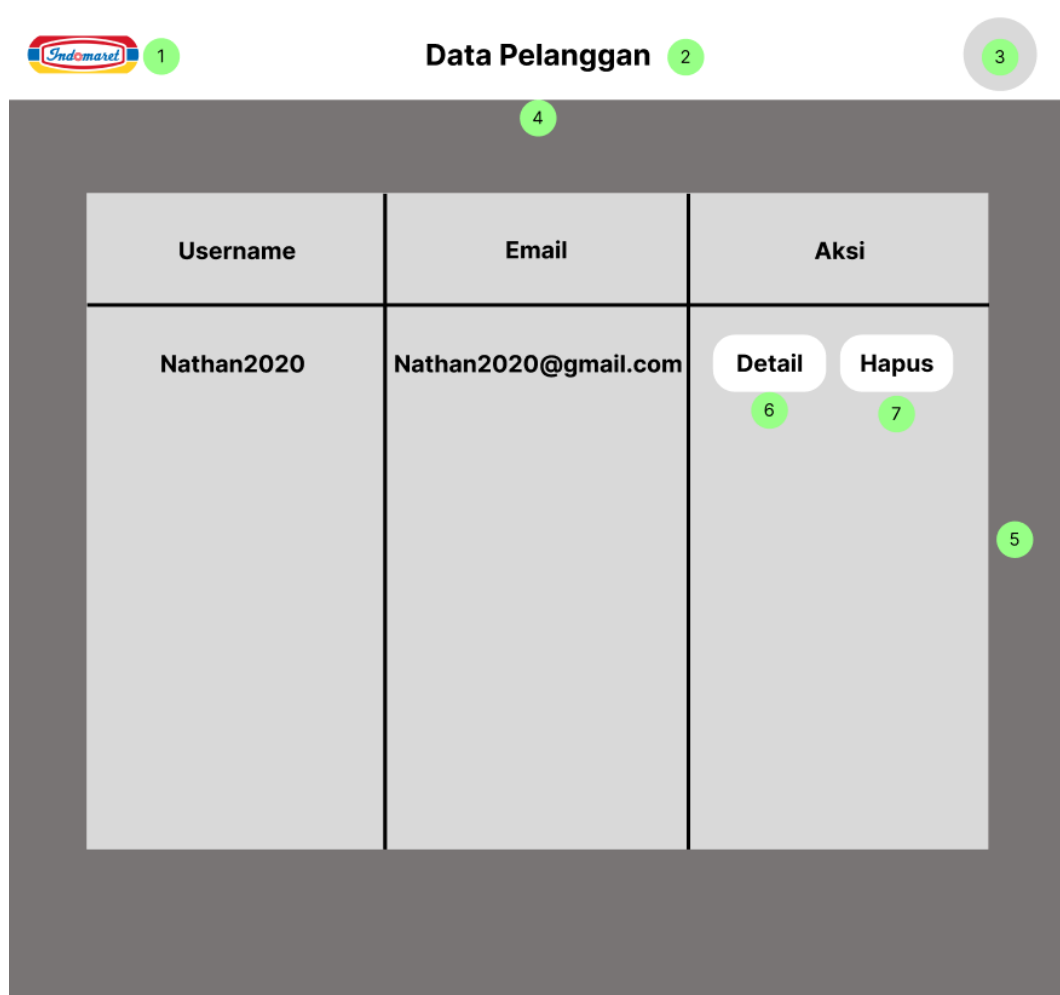
Adapun detail dari perancangan antarmuka Menampilkan Data Laporan Tindakan Kriminalitas yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan nama halaman yang diakses.

3. Nomor 3 melambangkan ikon profil karyawan.
4. Nomor 4 melambangkan form detil laporan tindakan kriminalitas.
5. Nomor 5 melambangkan status laporan.
6. Nomor 6 melambangkan tombol kembali, untuk kembali ke halaman daftar laporan kegiatan kriminalitas.
7. Nomor 7 melambangkan tombol *Update* untuk memperbaharui status dari laporan.

5.1.5.17 Perancangan Antarmuka Menampilkan Daftar Data Pelanggan

Berikut merupakan perancangan dari antarmuka Menampilkan Daftar Data Pelanggan yang dapat dilihat pada Gambar 5.29.



Gambar 5.29 Perancangan Antarmuka Menampilkan Daftar Data Pelanggan

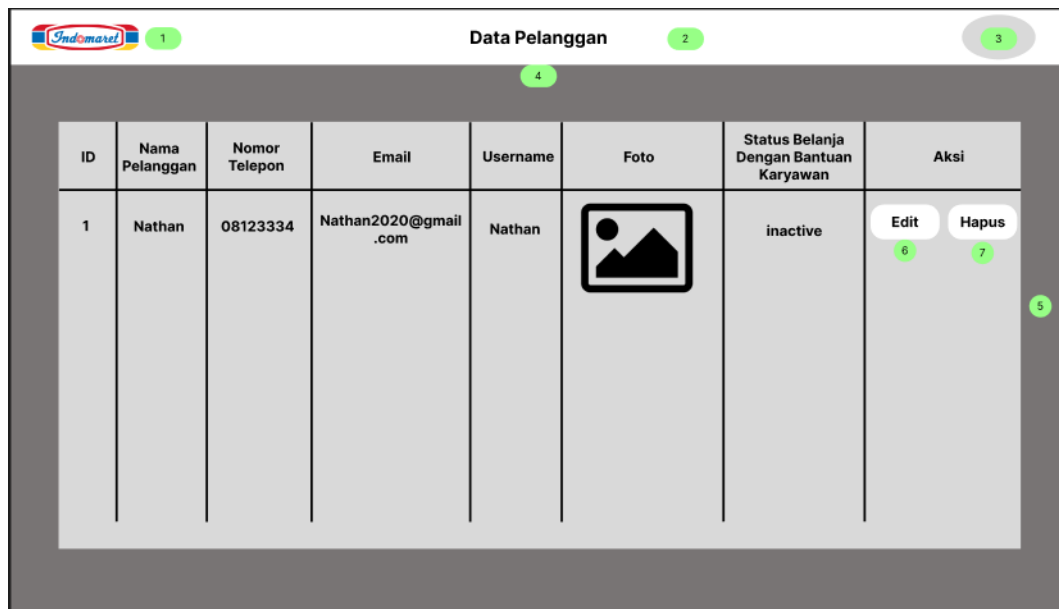
Adapun detil dari perancangan antarmuka Menampilkan Daftar Data Pelanggan yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.

2. Nomor 2 melambangkan nama halaman yang diakses.
3. Nomor 3 melambangkan ikon profil karyawan.
4. Nomor 4 melambangkan *navbar* sistem.
5. Nomor 5 melambangkan daftar data pelanggan.
6. Nomor 6 melambangkan tombol *Detail*, untuk melihat detil data pelanggan.
7. Nomor 7 melambangkan tombol Hapus untuk menghapus data pelanggan.

5.1.5.18 Perancangan Antarmuka Menampilkan Data Pelanggan

Berikut merupakan perancangan dari antarmuka Menampilkan Data Pelanggan yang dapat dilihat pada Gambar 5.30.



Gambar 5.30 Perancangan Antarmuka Menampilkan Data Pelanggan

Adapun detil dari perancangan antarmuka Menampilkan Daftar Data Pelanggan yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan nama halaman yang diakses.
3. Nomor 3 melambangkan ikon profil karyawan.
4. Nomor 4 melambangkan *navbar* sistem.
5. Nomor 5 melambangkan detil data pelanggan.
6. Nomor 6 melambangkan tombol *"Edit"*, untuk melakukan perubahan data pelanggan.

7. Nomor 7 melambangkan tombol “Hapus”, untuk melakukan penghapusan data pelanggan.

5.1.5.19 Perancangan Antarmuka Mengubah Data Pelanggan

The screenshot shows a web form titled "Data Pelanggan". At the top left is the Indomaret logo (1). The title "Data Pelanggan" is at the top center (2). A circular profile picture placeholder is at the top right (3). The form contains several input fields: "ID Pelanggan", "Nama Pelanggan", "Nomor Telepon", "Email", and "Username". Below these is a "Foto" section with a picture icon. At the bottom is a dropdown menu for "Status Belanja Dengan Bantuan Karyawan" with a downward arrow. A "Submit" button is at the bottom left (6). A green circle with the number 4 is at the top center of the form area, and a green circle with the number 5 is on the right side of the form area.

Gambar 5.31 Perancangan Antarmuka Mengubah Data Pelanggan

Adapun detail dari perancangan antarmuka Mengubah Data Pelanggan yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan nama halaman yang diakses.

3. Nomor 3 melambangkan ikon profil karyawan.
4. Nomor 4 melambangkan *navbar* sistem.
5. Nomor 5 melambangkan form data pelanggan.
6. Nomor 6 melambangkan tombol “*Submit*”, untuk menyimpan perubahan data pelanggan yang dilakukan.

5.1.5.20 Perancangan Antarmuka Menampilkan Daftar Pelanggan Belanja dengan Bantuan Karyawan



Gambar 5.32 Perancangan Antarmuka Menampilkan Daftar Pelanggan Belanja dengan Bantuan Karyawan

Adapun detail dari perancangan antarmuka Menampilkan Daftar Pelanggan Belanja dengan Bantuan Karyawan yaitu:

1. Nomor 1 melambangkan logo dari Indomaret.
2. Nomor 2 melambangkan nama halaman yang diakses.
3. Nomor 3 melambangkan ikon profil karyawan.
4. Nomor 4 melambangkan *navbar* sistem.

5. Nomor 5 melambangkan tabel data pelanggan yang membutuhkan bantuan belanja.
6. Nomor 6 melambangkan tombol “B
7. antu”, untuk melakukan perbelanjaan bagi pelanggan yang membutuhkan bantuan.

5.1.6 Perancangan Komponen

5.2 Implementasi

Pada bagian implementasi ini, dilakukan dengan menerapkan seluruh hasil rancangan sistem, yang mana hal – hal yang dibahas pada bagian ini meliputi spesifikasi sistem, implementasi antarmuka, implementasi *database*, dan implementasi algoritma.

5.2.1 Spesifikasi Sistem

Spesifikasi sistem memiliki dua bagian yaitu, spesifikasi perangkat lunak dan spesifikasi perangkat keras yang digunakan untuk mengembangkan sistem pelayanan mandiri untuk toko ritel di Kota Malang. Adapun berikut merupakan detail perangkat keras dan perangkat lunak yang digunakan.

5.2.1.1 Spesifikasi Perangkat Keras

Berikut merupakan spesifikasi perangkat keras yang digunakan untuk mengembangkan sistem pelayanan mandiri untuk toko ritel di Kota Malang.

Tabel 5.10 Tabel Detil Metode Pembayaran

Komponen	Spesifikasi
Laptop	Nama : ROG Strix G G531 (G531GT-AL541T)
	Prosesor : Intel® Core™ i7-9750H <i>Processor</i> 2.6 GHz (12M <i>Cache</i> , 4.5 GHz. 6 <i>Cores</i>)
	<i>Memory</i> : 16GB DDR4-2666 SO-DIMM x 2
	<i>Storage</i> : 2TB Samsung QVO SSD + 512GB PCIe® 3.0 NVMe™ M.2 SSD
	<i>Graphics</i> : NVIDIA® GeForce® GTX 1650 4GB GDDR5

5.2.1.2 Spesifikasi Lunak

Berikut merupakan spesifikasi perangkat lunak yang digunakan untuk mengembangkan sistem pelayanan mandiri untuk toko ritel di Kota Malang.

Tabel 5.11 Tabel Detil Metode Pembayaran

Komponen	Spesifikasi
Sistem Operasi	Windows 11
Bahasa Pemrograman	HTML, CSS, Javascript, PHP
Editor Kode	Visual Studio Code
Editor Perancangan	Draw.io
<i>Framework</i>	Laravel versi 10.6.1
DBMS	MySQL

5.2.2 Implementasi Antarmuka

5.2.3 Subbab Lima Dua Dua

Cara ketiga biasanya digunakan jika data, analisis, dan penafsirannya sulit dipisahkan. Pemisahannya terkadang justru membuat laporan penelitian sulit dibaca. Hal ini dapat berlaku pada tipe penelitian yang bersifat kualitatif, baik itu deskriptif ataupun analitik/eksplanatori.

Pada dasarnya peletakan dan jumlah bab untuk hasil dan pembahasan sebaiknya disesuaikan karakter penelitian masing-masing. Judul bab pun tidak harus secara eksplisit “Hasil” dan “Pembahasan” tetapi dapat digantikan dengan nama yang lebih deskriptif dan tematik.

5.3 Subbab Lima Tiga

Contoh struktur skripsi untuk implementatif pembangunan dan nonimplementatif eksperimental dapat dilihat pada kedua subbab berikut.

5.3.1 Contoh Struktur Penelitian Implementatif Pembangunan

Berikut ini adalah contoh bab-bab yang terdapat pada penelitian implementatif pembangunan sistem perangkat lunak.

- Bab 1 Pendahuluan
- Bab 2 Landasan Kepustakaan
- Bab 3 Metodologi Penelitian
- Bab 4 Persyaratan
- Bab 5 Perancangan dan Implementasi
- Bab 6 Pengujian
- Bab 7 Penutup

Bab 1 sampai Bab 3 memuat informasi yang sesuai dengan panduan sebelumnya. Isi dari bab-bab berikutnya:

- Bab 4 Persyaratan:
 - Pernyataan masalah (problem statement), yang lebih elaboratif daripada yang di Pendahuluan.
 - Identifikasi pemangku kepentingan (stakeholders) dan aktor (actors) sistem.
 - Daftar terstruktur persyaratan/kebutuhan perangkat lunak, secara fungsional, data, dan non-fungsional
 - Use cases, use case diagrams, dan use case specifications, dan sebagainya.
- Bab 5 Perancangan dan Implementasi:
 - Rancangan arsitektur: deskripsi struktur dan setiap komponen utama
 - Representasi data dalam model data dan basis data
 - Detil implementasi dari fungsi-fungsi utama yang menjadi fokus
- Bab 6 Pengujian dan Evaluasi
 - Strategi, rencana, kasus, dan data pengujian
 - Ringkasan hasil pengujian perangkat lunak, termasuk data dan analisisnya (detilnya di Lampiran)
 - Evaluasi hasil proyek secara keseluruhan, misalkan
- Bab 7 Penutup
 - Ringkasan dari capaian proyek
 - Saran pengembangan lebih lanjut

Pada contoh struktur ini “hasil” tersebar di beberapa bab mulai Bab 4 Persyaratan sampai Bab 6, sedangkan “pembahasan” secara keseluruhan terhadap masalah penelitian terdapat di Bab 6. Yang dimaksud dengan pengujian dalam Bab 6 terfokus pada pengujian persyaratan perangkat lunak, sedangkan evaluasi berfungsi sebagai “pembahasan” secara keseluruhan, yaitu menentukan apakah “hasil” sudah menjawab masalah penelitian yang dirumuskan pada Bab 1.

Sebagai catatan, Bab 3 Metodologi umumnya menjelaskan model proses perangkat lunak yang digunakan. Jika strategi untuk setiap aktivitasnya (analisis persyaratan, perancangan, dan seterusnya) sudah dijelaskan di Bab 3 ini juga, maka bab-bab lainnya yang berhubungan dengan aktivitas-aktivitas ini masing-masing langsung dapat menjelaskan hasil pelaksanaan metodenya.

5.3.2 Contoh Struktur Penelitian Nonimplementatif Eksperimental

Berikut ini adalah contoh bab-bab yang terdapat pada penelitian implementatif pembangunan sistem perangkat lunak.

Bab 1 Pendahuluan

Bab 2 Landasan Kepustakaan

Bab 3 Metodologi Penelitian

Bab 4 Hasil

Bab 5 Pembahasan

Bab 6 Penutup

Isi dari setiap bab dapat menyesuaikan dengan panduan yang telah dijelaskan sebelumnya. Jika diperlukan, Bab 4 dapat digabungkan dengan Bab 5, menjadi Hasil dan Pembahasan.

Struktur dasar ini cukup universal sehingga dapat digunakan juga untuk tipe-tipe penelitian lainnya, khususnya jika belum ada struktur lain yang lebih tematik dan cocok untuk penelitian yang bersangkutan.

BAB 6 PENUTUP

Bagian ini memuat kesimpulan dan saran terhadap skripsi. Kesimpulan dan saran disajikan secara terpisah, dengan penjelasan sebagai berikut:

6.1 Kesimpulan

Kesimpulan merupakan pernyataan-pernyataan yang singkat, jelas, dan tepat tentang hasil penelitian yang diperoleh berdasarkan tujuannya. Bagian ini merupakan penegasan dari yang telah dijelaskan pada bagian Pembahasan dan tidak memuat informasi yang baru. Bagian ini juga mencerminkan jawaban dari rumusan masalah (pertanyaan penelitian).

6.2 Saran

Saran berisi pernyataan-pernyataan yang ringkas dan jelas tentang masalah-masalah atau hal-hal yang dapat dilakukan untuk mengembangkan penelitian ini lebih lanjut. Saran itu dapat diarahkan pada aspek metode, instrumen, populasi/sampel, dan sebagainya.

DAFTAR REFERENSI

- ANHAR, A. and PUTRA, R.A., 2023. Perancangan dan Implementasi Self-Checkout System pada Toko Ritel menggunakan Convolutional Neural Network (CNN). *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 11(2), p.466.
- Arizal, A., Puteri, A.N., Zakiyabarsi, F. and Priambodo, D.F., 2022. Metode Prototype pada Sistem Informasi Manajemen Tugas Akhir Mahasiswa Berbasis Website. *Jurnal Teknologi Informasi dan Komunikasi (TIKoSIN)*, 10(1), pp.1-8.
- Chengjun, S. and Pilian, H., 2000. Applied research on developing web database with PHP [J]. *Computer engineering*, 9, pp.184-185.
- Dewi, N.R., Hartati, R.S. and Divayana, Y., 2021. Penerapan Metode Prototype dalam Perancangan Sistem Informasi Penerimaan Karyawan Berbasis Website pada Berlian Agency. *Maj. Ilm. Teknol. Elektro*, 20(1), p.147.
- Di, N., 2014. Design of hospital website based on PHP technology [J]. *China health vision: medical edition*, 3, pp.175-176.
- Duckett, J., 2014. *Web design with HTML, CSS, JavaScript and jQuery set*. Wiley Publishing.
- Firma Sahrul, B., Safi'ie, M.A. and Wa, O.D., 2016. Implementasi Sistem Informasi Akademik Berbasis Web Menggunakan Framework Laravel. *TRANSFORMASI*, 12(1).
- Firmansyah, Y., Maulana, R. and Maulana, M.S., 2021. Implementasi Metode SDLC Prototype Pada Sistem Informasi Indeks Kepuasan Masyarakat (IKM) Berbasis Website Studi Kasus Dinas Kependudukan Dan Catatan Sipil. *JUSTIN (Jurnal Sistem dan Teknologi Informasi)*, 9(3), pp.315-323.
- Fridayanthie, E.W., Haryanto, H. and Tsaibitah, T., 2021. Penerapan Metode Prototype Pada Perancangan Sistem Informasi Penggajian Karyawan (Persis Gawan) Berbasis Web. *Jurnal Khatulistiwa Informatika*, 23(2), p.472897.
- Gellersen, H.W. and Gaedke, M., 1999. Object-oriented web application development. *IEEE Internet Computing*, 3(1), pp.60-68.
- Huda, M.K. and Rozi, N.F., 2022, December. Rancang Bangun Sistem Gudang Berbasis Web Dengan Model Pengembangan Rapid Prototyping. In *Prosiding Seminar Nasional Sains dan Teknologi Terapan*.
- Matthews, M., 2014. *PHP and MySQL Web Development: A Beginner's Guide*. McGraw-Hill Education Group.
- Mueller, J.P., 2015. *Security for Web Developers: Using JavaScript, HTML, and CSS*. " O'Reilly Media, Inc."

- Naramore, E., Gerner, J., Le Scouarnec, Y., Stolz, J. and Glass, M.K., 2005. *Beginning PHP5, Apache, and MySQL web development*. John Wiley & Sons.
- Nazihovna, Y.G., 2022. CREATING A PLATFORM USING HTML, CSS AND JAVA SCRIPT METHODS AND STRENGTHENING EDUCATION WITH THIS STEAM. *Confrencea*, 5(5), pp.17-38.
- Robbins, J.N., 2012. *Learning web design: A beginner's guide to HTML, CSS, JavaScript, and web graphics*. " O'Reilly Media, Inc."
- Saputra, D., 2016. Perbandingan teknologi system software, application software, embedded software dan web applications. *Jurnal Bangkit Indonesia*, 5(1), pp.74-74.
- Schifreen, R., 2009. How to create Web sites and applications with HTML, CSS, Javascript, PHP and MySQL.
- Schwabe, D. and Rossi, G., 1998. An object oriented approach to web-based applications design. *TAPOS*, 4(4), pp.207-225.
- Suwarno, S. and Lim, R.R., 2021, March. Perancangan Sistem Kasir Layanan Mandiri Berbasis Web Pada Supermarket Bless Dengan Metode Sdlc. In *CoMBInES-Conference on Management, Business, Innovation, Education and Social Sciences* (Vol. 1, No. 1, pp. 860-873).
- Valade, J., 2011. *PHP and MySQL Web Development All-in-One Desk Reference For Dummies*. John Wiley & Sons.
- Welling, L. and Thomson, L., 2003. *PHP and MySQL Web development*. Sams Publishing.

LAMPIRAN A HASIL WAWANCARA

Narasumber: Sdr. Okky

1. Bagaimana proses berbelanja di Indomaret?

Proses berbelanja di Indomaret dimulai dengan pelanggan memasuki area toko, kemudian pelanggan dapat memilih barang – barang apa saja yang diperlukan, setelah itu, pelanggan dapat membawa barang – barang tersebut ke kasir untuk dibayar, akan tetapi, apabila terdapat pelanggan di depannya, maka pelanggan tersebut perlu mengantri. Ketika sudah saatnya pelanggan tersebut membayar, maka pelanggan dapat memberikan barangnya kepada petugas kasir untuk dilakukan pemindaian barang belanjaan dan pengemasan barang ke dalam kantong plasti/belanja, lalu petugas kasir menanyakan kepada pelanggan berkaitan dengan metode pembayaran yang dipilih, apabila pembayaran dilakukan secara *cashless*, maka pelanggan dapat melakukannya melalui Qris, Kartu Debit, Kartu Kredit, namun untuk pembayaran tunai, maka pelanggan perlu memberikan uang sejumlah total belanjanya. Setelah itu, petugas kasir akan memberikan kuitansi belanja dan barang belanjanya, kemudian pelanggan dapat meninggalkan area toko Indomaret.

2. Bagaimana sistem yang digunakan di Indomaret saat ini?

Sistem yang digunakan masih bersifat konvensional, yang mana sistem dilakukan oleh manusia.

3. Apa saja kendala yang dialami saat ini?

Kendala yang dialami saat ini berupa *overcapacity* pengunjung ketika jam sibuk, selain itu, tidak efisiennya pembayaran di kasir, karena menimbulkan antrian pelanggan yang hendak melakukan pembayaran barang belanjaan, yang mana rata – rata waktu pelayanan di kasir adalah 5 – 10 menit.

4. Apakah terdapat kejadian – kejadian tidak terduga selama proses berbelanja di Indomaret selama ini?

Ya, terdapat kejadian seperti pencurian barang, meskipun sudah diawasi oleh karyawan – karyawan yang bekerja.

5. Apa yang menjadi kebutuhan khusus agar diimplementasikan dalam sistem?

Untuk kebutuhan yang khususnya yaitu fitur laporan pencurian yang dapat dilakukan oleh pengunjung lainnya, karena dengan adanya fitur itu, maka dapat membantu karyawan yang bekerja dalam menjaga keamanan toko.