

BAB I

PENDAHULUAN

A. Latar Belakang

Bisnis penyedia jasa atau layanan serta penjualan produk saat ini semakin berkembang pesat dan menjadi salah satu sektor bisnis yang menjanjikan keuntungan (Aprianto, 2021). Dalam kondisi persaingan yang semakin ketat, pelaku usaha harus mampu memberikan pelayanan yang lebih baik dan efisien untuk para pelanggannya guna mempertahankan pangsa pasar. Dalam rangka meningkatkan pelayanan dan efisiensi bisnis tersebut, pengembangan aplikasi pemesanan produk menjadi sangat penting. Aplikasi tersebut akan memberikan kemudahan dan kepraktisan dalam melakukan pemesanan, pembayaran, serta memudahkan pelanggan dalam mengetahui informasi produk yang tersedia dan status pemesanan (Suri & Hasibuan, 2021).

Dalam hal manajemen bisnis, aplikasi tersebut dapat membantu meningkatkan efisiensi, antara lain dalam hal manajemen stok produk, pengelolaan pembayaran, dan pelacakan pemesanan. Penggunaan aplikasi juga dapat menghemat biaya dan waktu yang diperlukan dalam proses bisnis, serta meningkatkan kualitas pelayanan yang diberikan. Namun, pengembangan aplikasi pemesanan produk tidak dapat diabaikan faktor keamanan dan privasi data pelanggan. Oleh karena itu, aplikasi tersebut harus dirancang dengan memperhatikan standar keamanan yang tinggi dan dilengkapi dengan sistem peng

manan yang memadai. Hal ini bertujuan untuk menjaga kepercayaan pelanggan dan mencegah adanya tindakan kejahatan siber yang dapat membahayakan bisnis dan pengguna aplikasi.

Untuk memenuhi kebutuhan tersebut, penelitian dan pengembangan aplikasi pemesanan produk perlu dilakukan dengan serius. Penelitian dan pengembangan tersebut diharapkan dapat menghasilkan aplikasi yang berkualitas dan dapat memberikan manfaat bagi pengguna dan bisnis tersebut utamanya bagi mitra. Sehingga dapat meningkatkan kinerja bisnis dan memberikan kepuasan kepada para pelanggan.

B. Rumusan Masalah

Adapun rumusan masalah dari latar belakang yang ada yaitu :

1. Bagaimana merancang aplikasi pemesanan produk berbasis web ?
2. Bagaimana konsumen dapat menggunakan aplikasi serta pemesanan produk pada web ?

C. Tujuan Penelitian

Adapun tujuan yang hendak dicapai dalam penelitian ini yaitu :

1. Merancang dan mengembangkan sebuah aplikasi berbasis web yang memungkinkan pelanggan PT. Mitra Makassar untuk melakukan pemesanan dan pembelian produk.
2. Meningkatkan efisiensi pelayanan dan pemesanan produk pada PT. Mitra Makassar

D. Batasan Masalah

Agar pembahasan penelitian lebih terarah dan tidak menyimpang dari tujuan penelitian, maka dibuat batasan-batasan masalah diantaranya :

1. Sistem yang melibatkan tahapan pemesanan produk pada PT. Mitra Makassar.
2. Sistem yang memudahkan *customer* dalam melakukan pemesanan produk dimana saja dan kapan saja.
3. Terdapat *output* berupa data pemesanan produk dan data produk.

E. Manfaat Penelitian

Adapun manfaat yang akan didapatkan dari pelaksanaan penelitian ini adalah:

1. Terciptanya aplikasi web pemesanan produk yang dapat membantu PT. Mitra Makassar dalam hal peningkatan efisiensi bisnis.
2. Efisiensi pelayanan dan pemesanan produk yang meningkat dan memberi dampak diantaranya peningkatan kualitas operasional, peningkatan kepuasan pelanggan, dan mendorong pertumbuhan bisnis yang berkelanjutan khususnya pada PT. Mitra Makassar.

BAB II

TINJAUAN PUSTAKA

A. Definisi Perancangan

Perancangan merupakan tahapan awal dalam suatu proses di mana segala elemen direncanakan sebelumnya. Perancangan berperan dalam menggambarkan, merencanakan, serta membuat sketsa elemen-elemen yang terpisah menjadi satu kesatuan utuh yang memiliki fungsi yang terdefinisi dengan baik (Triyono, Safitri, & Gunawan, 2018). Kemudian, perancangan sistem adalah langkah dalam siklus pengembangan sistem yang dapat dijelaskan sebagai fase dimana kebutuhan fungsional didefinisikan dan diilustrasikan bagaimana struktur suatu sistem terbentuk (Sitepu & Tanjung, 2020).

B. Pemesanan Produk

Pemesanan produk adalah prosedur formal yang digunakan oleh pembeli untuk mendapatkan produk dari vendor. Prosedur ini membutuhkan sejumlah langkah yang direncanakan dengan cermat. Pelanggan pertama-tama harus memilih barang yang ingin mereka beli berdasarkan kebutuhan dan keinginan mereka. Mereka kemudian memutuskan berapa banyak persediaan yang perlu dipesan. Selanjutnya, pelanggan memasukkan rincian yang diperlukan untuk menyelesaikan proses pemesanan, termasuk nama, alamat, dan informasi kartu kredit. Tahap berikutnya adalah pelanggan mengkonfirmasi pesanan mereka, setelah itu mereka harus membayar barang yang dipesan. Pada akhirnya, penjual memikul tanggung

jawab untuk menyediakan produk yang dipesan kepada pelanggan (Jannah & Mulyono, 2021).

Pelanggan memiliki kemampuan untuk memesan berbagai jenis produk. Salah satunya adalah pemesanan online, ketika pelanggan melakukan pemesanan menggunakan situs web atau aplikasi khusus di internet. Selain itu, pemesanan offline dapat dilakukan melalui telepon atau secara langsung di toko. Ada juga jenis pesanan khusus lainnya, seperti *pre-order*, di mana pelanggan mengirimkan pesanan sebelum produk dirilis, dan *backorder*, di mana pelanggan memesan barang yang tidak tersedia saat ini tetapi akan dipesan oleh penjual di kemudian hari (Supriyatna, 2017).

Terdapat beberapa aspek yang dapat memengaruhi cara pelanggan memesan produk. Ini termasuk harga yang kompetitif, produk berkualitas tinggi, akses cepat ke barang dagangan, promosi yang memikat, dan umpan balik pelanggan yang mendorong yang dapat meningkatkan kepercayaan pembeli terhadap produk. Selain itu, ada sejumlah ide yang berkaitan dengan pemesanan produk dalam pengaturan teori. Contoh teori tersebut termasuk perilaku konsumen, yang menjelaskan bagaimana konsumen membuat keputusan tentang apa yang akan dibeli, teori pemasaran, yang berbicara tentang bagaimana bisnis menjual produk mereka kepada konsumen, dan teori logistik, yang menjelaskan bagaimana bisnis menjalankan rantai pasokan mereka untuk menjamin bahwa pelanggan dapat memperoleh produk mereka (Fristanto, 2013).

Terakhir, terdapat juga beberapa model pemesanan produk yang digunakan oleh konsumen. Model-model ini mencakup model tradisional di mana konsumen

memesan barang langsung di toko, model *e-commerce* di mana pemesanan dilakukan secara online, serta model marketplace di mana konsumen memesan produk melalui *platform* online yang menghubungkan mereka dengan berbagai penjual. Dengan pemahaman yang baik tentang konsep, jenis, faktor, teori, dan model terkait pemesanan produk, konsumen dapat membuat keputusan pembelian yang lebih tepat sesuai dengan kebutuhan dan preferensi mereka (Hayami, Harianto, & Unik, 2019).

C. *Website*

Website dideskripsikan sebagai sekumpulan halaman yang digunakan untuk menampilkan informasi tertulis, gambar diam atau bergerak, animasi, suara, atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis, untuk membentuk suatu jaringan struktur yang saling berhubungan. *Website* merupakan kumpulan halaman web yang saling berhubungan yang menggunakan nama domain yang sama. Hasilnya, siapa pun yang memiliki koneksi internet dapat menjelajahi situs web kapan pun dan dari lokasi mana pun (Rusli, Ahmar, & Rahman, 2019).

Situs web melayani berbagai macam tujuan, mulai dari informasi dasar hingga berita terkini, serta konten khusus seperti blog, forum, atau *platform e-commerce*. Situs web menyediakan keterlibatan langsung antara pengguna dan pemilik situs melalui elemen interaktif seperti formulir kontak, komentar, dan alat obrolan. Akses ke situs web juga dapat dilakukan melalui PC, laptop, atau ponsel yang dilengkapi dengan peramban web seperti Chrome, Firefox, atau Safari. Situs web pribadi atau blog, situs web berita, situs web *e-commerce*, situs web instruksional, dan sebagainya adalah contoh-contoh situs web (Triyono et al., 2018)

D. Pemograman

Program adalah kumpulan pernyataan komputer, sedangkan algoritma adalah metode dan tahapan sistematis yang ditulis menggunakan bahasa pemograman. Struktur data dan algoritma berhubungan sangat erat pada sebuah program. Pemilihan struktur data yang kurang tepat akan membuat program menjadi kurang baik. Pemograman tidak hanya sebatas pada penulisan kode semata, tetapi juga mencakup aspek-aspek lain seperti analisis kebutuhan, desain sistem, serta pengujian dan debugging untuk memastikan bahwa perangkat lunak yang dihasilkan memenuhi spesifikasi dan berfungsi dengan baik dalam berbagai kondisi (Ade Hastuty Hasyim, 2021).

E. Algoritma

Algoritma merupakan urutan langkah-langkah penyelesaian masalah yang disusun secara sistematis dan logis. Pemilihan langkah langkah dalam algoritma harus logis dan dapat ditentukan nilai salah dan benarnya. Didalam pemilihan algoritma terdapat beberapa pertimbangan seperti algoritma dapat memberikan *output* yang dikehendaki dari jumlah *input* yang diberikan, algoritma harus memberikan hasil yang sedekat mungkin dengan nilai yang sebenarnya serta efisiensi algoritma tersebut yang ditinjau dari waktu komputasi dan memori yang dipakai. (Ade Hastuty Hasyim, 2021).

F. Next.js

Next.js adalah sebuah *framework* perangkat lunak bersifat *open-source* yang di bangun diatas lingkungan proses *Node.js*. *Framework* ini memungkinkan

aplikasi *website* berbasis *React.js* untuk menerjemahkan kontennya terlebih dahulu di *back-end* sebelum ditampilkan atau membuat aplikasi tersebut menjadi sebuah *website* statis. Dengan menggunakan *Next.js*, penulis dapat membangun aplikasi web yang interaktif, cepat, dan ramah terhadap algoritma pengindeksian mesin pencari seperti Google (Yudhanto & Prasetyo, 2019).

G. JavaScript

JavaScript adalah salah satu bahasa pemrograman yang berdasarkan standar *ECMAScript*. *JavaScript* merupakan bahasa pemrograman Tingkat tinggi yang banyak digunakan oleh programmer untuk membangun aplikasi *front-end* yang dinamis, mudah digunakan dan memiliki tampilan yang menarik. Dalam pengembangan web modern memungkinkan pembuatan aplikasi web yang tidak hanya fungsional dan cepat, tetapi juga estetis dan mudah diakses oleh pengguna dari berbagai perangkat Selain itu *JavaScript* juga populer digunakan sebagai bahasa pemrograman untuk membangun server *back-end* yang menerima dan memproses data yang dimasukkan oleh pengguna (Rusli, Ahmar, & Rahman, 2019).

H. Database

Database atau basis data adalah kumpulan informasi yang disimpan secara sistematis di dalam komputer sehingga dapat diperiksa dengan menggunakan program komputer untuk mengekstrak informasi dari basis data. Sistem manajemen basis data (*Database Management System*, DBMS) adalah perangkat lunak yang digunakan untuk memelihara dan melakukan kueri terhadap basis data. Ilmu

informasi mempelajari sistem basis data. Tujuan utama dari *Database* adalah untuk memungkinkan pengguna membuat abstraksi tampilan data. Hal ini bertujuan untuk memfasilitasi interaksi pengguna dengan sistem dan *Database*, yang dapat memberikan tampilan yang berbeda kepada pengguna, pemrogram, dan administrator (Andaru, 2018).

I. *Unified Modelling Language (UML)*

Unified Modeling Language (UML) adalah sekumpulan struktur dan metode untuk mendokumentasikan desain dan penggunaan program berorientasi objek (OOP). UML adalah teknik untuk mendesain sistem OOP yang didukung oleh seperangkat alat untuk membantunya. *Unified Modeling Language (UML)* adalah sekumpulan notasi grafis berdasarkan meta-model tunggal yang digunakan untuk menggambarkan dan membangun sistem perangkat lunak, khususnya yang dibangun dengan pemrograman berorientasi objek (Ade Hastuty Hasyim, 2021). Adapun bagian-bagian UML yang digunakan pada penelitian ini diantaranya *use case diagram*, *Class Diagram*, *Sequence Diagram* serta *Activity Diagram*.

Use case diagram mendefinisikan perilaku sistem informasi selanjutnya. Diagram ini menggambarkan interaksi pemain atau aktor dengan sistem informasi yang sedang dikembangkan. *Use case* digunakan untuk mengidentifikasi fungsi-fungsi yang ada di dalam sistem dan aktor yang menggunakannya. *Use case diagram* memainkan peran penting dalam menentukan, merinci, dan mendokumentasikan persyaratan perilaku sistem. Sedangkan, *Class Diagram*

digunakan untuk menggambarkan struktur objek sistem. Diagram ini menggambarkan objek kelas sistem serta hubungan antara objek kelas ini. Diagram kelas adalah jenis diagram yang menggambarkan kelas-kelas dalam sistem dan hubungan logisnya (Ade Hastuty Hasyim, 2021).

Activity Diagram, yaitu diagram yang digunakan untuk menggambarkan aliran aktivitas atau langkah-langkah dalam suatu proses atau sistem. Diagram ini menggambarkan urutan aktivitas sistem, yang meliputi keputusan, perulangan, dan konkurensi. *Activity Diagram* membantu dalam desain logika bisnis sistem dan alur kerja. Grafik ini menggambarkan aliran aktivitas dengan menggunakan simbol-simbol seperti aktivitas (*Activity*), panah, pilihan (*decision*), dan *fork/join* (Ade Hastuty Hasyim, 2021).

Sequence Diagram adalah diagram yang digunakan untuk menunjukkan bagaimana item berinteraksi dalam sistem perangkat lunak. Grafik ini menggambarkan urutan pesan yang dikirimkan di antara berbagai objek, serta bagaimana pesan-pesan tersebut ditangani dari waktu ke waktu. *Sequence Diagram* berguna untuk merepresentasikan aliran logis dan komunikasi antara item sistem. Diagram ini membantu dalam memahami bagaimana item berinteraksi satu sama lain dan bagaimana pesan dikirim dan diterima di antara mereka (Ade Hastuty Hasyim, 2021).

Diagram kelas atau *class diagram* adalah jenis diagram *statis* yang menunjukkan banyak komponen sistem secara grafis. Umumnya para pengembang menggunakannya untuk menguraikan bagaimana sebuah sistem dapat terlihat dalam bentuk *statis*. *Class diagram* memiliki sifat statis artinya tidak menjelaskan

mengenai apa yang terjadi apabila kelas-kelasnya berhubungan, tetapi menjelaskan hubungan apa yang terjadi. Hal ini sangat sesuai apabila diimplementasikan pada proyek yang memakai konsep *objek oriented*. Alasannya, gambaran dari *diagram* kelas sangat mudah digunakan.

Tabel 2.1 *Table Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan

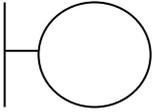
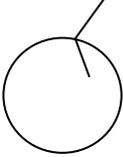
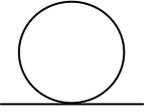
			prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2.2 *Symbol Activity Diagram*

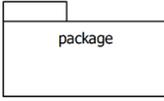
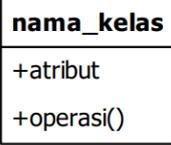
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi.
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

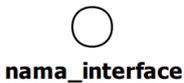
Tabel 2.3 *Symbol Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Objekt</i> (<i>Partisipan</i>)	<i>Objekt</i> atau biasa juga disebut partisipan merupakan <i>instance</i> dari sebuah <i>class</i> dan dituliskan tersusun secara <i>horizontal</i> . Digambarkan sebagai sebuah <i>class</i> (kotak) dengan nama objek di dalamnya yang diawali dengan sebuah titik koma.
2		<i>Actor</i>	<i>Actor</i> juga dapat berkomunikasi dengan <i>object</i> , maka <i>actor</i> juga dapat diurutkan sebagai kolom.

3		<i>Life line</i>	<i>Life line</i> mengidentifikasi keberadaan sebuah object dalam basis waktu. Notasi untuk <i>lifeline</i> adalah garis putus-putus <i>vertical</i> yang di tarik oleh sebuah <i>object</i>
4		<i>Collaborations</i>	<i>Activation</i> dinotasikan sebagai kotak segi empat yang digambar pada sebuah <i>lifeline</i> <i>action</i> mengidentifikasi sebuah <i>object</i> yang akan beraksi.
5		<i>Boundary</i>	<i>Boundary</i> terletak diantara system dengan dunia di sekelilingnya. Semua <i>form</i> , laporan-laporan, antara muka ke perangkat keras seperti <i>printer</i> atau <i>scenner</i> dan antara muka ke system lainnya adalah termasuk dalam kategori.
6		<i>Control</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Entity</i>	<i>Entity</i> digunakan untuk menengani informasi yang akan disimpan secara permanen.

Tabel 2.4 *Symbol Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Package</i>	<i>Package</i> merupakan sebuah bungkusan dari satu atau lebih <i>class</i> .
2		<i>Class</i>	Simbol <i>class</i> berfungsi sebagai pemetaan pada himpunan yang berasal dari objek yang mana terdapat operasi dan atribut yang tidak begitu berbeda.

3		<i>Antarmuka (Interface)</i>	Sekumpulan atribut yang mendefinisikan perilaku yang saling berkaitan. Komponen ini mirip dengan kelas, tetapi harus dimiliki oleh setidaknya satu kelas untuk diimplementasikan.
4		<i>Association</i>	Simbol <i>association</i> bermakna sebagai relasi antara suatu objek dengan objek yang lain sebagai pemetaan.
5		<i>directed association</i>	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
6		<i>Generalization</i>	Pada simbol <i>generalization</i> menunjukkan simbol dengan bentuk garis lurus sebagai perilaku serta struktur data yang berasal dari objek yang berada pada ancestor atau objek induk.
7		<i>Dependency</i>	relasi antar kelas dengan makna kebergantungan antar kelas.
8		<i>Aggregation</i>	relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>).

J. *Flowchart*

Flowchart adalah representasi visual berupa grafik dari urutan langkah-langkah yang perlu diikuti untuk menyelesaikan sebuah masalah. *Flowchart* terdiri dari serangkaian simbol, dimana setiap simbol mewakili aktivitas tertentu. *Flowchart* biasanya dimulai dengan penerimaan input, dilanjutkan dengan pemrosesan input, dan diakhiri dengan menampilkan output. Dengan kata lain, *flowchart* adalah diagram yang menunjukkan alur: mulai dari pembacaan data, pengolahan data, pengambilan keputusan terhadap data, hingga penyajian hasil pengolahan data (Ade Hastuty Hasyim, 2021).

Tabel 2. 5 Simbol Flowchart

NO	GAMBAR	NAMA	KETERANGAN
1.		Simbol <i>Input</i>	Simbol ini melambangkan kegiatan <i>input</i> . Dalam simbol ini kita dapat menuliskan <i>input</i> -an yang diperlukan pada suatu waktu.
2.		Simbol Proses	Simbol ini melambangkan kegiatan pemrosesan input. Dapat dituliskan dengan operasi yang digunakan oleh input.
3.		Simbol <i>Output</i>	Simbol ini melambangkan kegiatan penampilan output. Dapat dituliskan dengan semua output yang harus ditampilkan oleh program

K. Pengujian Aplikasi

Pengujian aplikasi adalah salah satu langkah penting dalam proses pengembangan perangkat lunak yang bertujuan untuk memastikan bahwa aplikasi tersebut berfungsi sesuai dengan spesifikasi yang telah ditetapkan dan bebas dari kesalahan atau bug (Rusli, Ahmar, & Rahman, 2019).

1. Pengujian *White box* (*White box Testing*)

Pengujian *white box*, juga dikenal sebagai pengujian struktural, adalah metode pengujian di mana penguji memiliki pengetahuan tentang struktur internal atau kode sumber dari aplikasi yang diuji. Dalam pengujian ini, penguji memeriksa jalur logika, aliran data, kondisi, dan keputusan yang dibuat dalam kode. Fokus utama dari pengujian *white box* adalah untuk memastikan bahwa semua jalur eksekusi dalam program telah diuji dan berfungsi dengan benar.

2. Pengujian *Black Box* (*Black box Testing*)

Pengujian *black box*, atau pengujian fungsional, adalah metode pengujian di mana penguji tidak memiliki pengetahuan tentang struktur internal atau kode sumber dari aplikasi yang diuji. Pengujian ini berfokus pada pengujian fungsionalitas aplikasi berdasarkan spesifikasi atau persyaratan yang telah ditentukan. Penguji hanya mengetahui input yang masuk dan output yang dihasilkan, tanpa memperhatikan bagaimana aplikasi tersebut mencapai hasil tersebut

L. Kajian Hasil Penelitian Terdahulu

Dalam menyusun penelitian ini, penulis mendapatkan referensi kajian penelitian terdahulu yang berkaitan dengan penelitian ini tentunya. Adapun penelitian terdahulu yang berkaitan dengan penelitian penulis diantaranya :

1. Penelitian dengan judul “Implementasi Sistem Informasi *Booking Service Online* Pada PT. Riau Argo Perkasa Berbasis Web” oleh (Simatupang & Gomal Juni Yanris, 2020). Persamaan penelitian ini yaitu kedua-duanya memiliki fitur *booking service* dan fitur pengelolaan data yang akan dikelola oleh administrator. Sedangkan perbedaannya ialah, penelitian ini tidak memiliki fitur produk maupun penjualan produk.
2. Penelitian dengan judul “Pembangunan Aplikasi Mobile Toko *Online* Dan *Booking Service* Di Sentra Anugrah Motor Memanfaatkan *API Payment Gateway*” oleh (Hidayatulloh, 2020). Persamaan penelitian ini dengan penelitian penulis yaitu fitur *booking service*, fitur penjualan produk serta fitur transaksi. Adapun perbedaannya adalah penelitian kedua ini menggunakan *API Payment Gateway* dalam transaksinya.
3. Penelitian ketiga dengan judul “Perancangan Sistem Aplikasi *Management Booking Service* Bengkel Berbasis *Website* Dengan Metode *Prototype* (Studi Kasus Bengkel Motor Rido Racing Rempoa Tangerang Selatan)” oleh (Pangalila & Ardhiansyah, 2022). Persamaan penelitian ini dengan penelitian penulis yaitu kedua-duanya menerapkan fitur *booking service*. Adapun perbedaannya ialah pada penggunaan metode penelitian yang digunakan.

Penelitian ketiga ini menggunakan metode *Rapid Application Development* (RAD).

4. Penelitian keempat dengan judul “Sistem Informasi Penyedia Jasa Penjualan Pakaian Berbasis Android” (Herawan, Alam, Zainal, & Herawan, 2021).
Persamaan penelitian ini yaitu menerapkan sistem informasi penjualan suatu produk. Sedangkan perbedaan penelitian ini adalah pada penelitian ini penulis tidak menerapkan layanan *booking*. Sedangkan penelitian yang diusulkan menerapkan layanan booking serta penjualan produk.
5. Penelitian kelima dengan judul “Sistem Pengambilan Nomor Antrian Dan Pengingat Jadwal Kunjungan Berbasis Whatsapp Gateway (Study Kasus Klinik Sitti Khadijah)” (Zainal, 2022). Adapun persamaan penelitian ini adalah kedua-dua menerapkan sistem pengambilan nomor antrian namun dalam hal penelitian yang akan diajukan yaitu layanan booking yang keduanya hampir mirip. Sedangkan perbedaannya adalah penelitian yang akan diajukan juga menerapkan sistem penjualan produk pada PT. Mitra Makassar

BAB III

METODE PENELITIAN

A. Jenis Penelitian

Adapun jenis penelitian yang digunakan penelitian kualitatif dengan menggunakan metode pendekatan penelitian yaitu *Research and Development* (R&D) meliputi studi literatur, pengumpulan data, analisis sistem, perancangan sistem, pembuatan aplikasi, pengujian aplikasi (*testing*), serta implementasi.

B. Lokasi dan Waktu

Lokasi penelitian ini bertempat pada PT. Mitra Makassar yang beralamatkan di Jl. Bambu Runcing, Bumi Harapan, Kec. Bacukiki Barat. Dan waktu penelitian dilakukan selama 4 bulan dimulai pada bulan Januari sampai dengan April 2024.

Tabel 3. 1 Jadwal Pelaksanaan Penelitian

No.	Uraian Kegiatan	Tahun 2024			
		Januari	Februari	Maret	April
1.	Studi Literatur				
2.	Pengumpulan Data (Observasi & Wawancara)				
3.	Analisis Sistem				
4.	Perancangan Sistem				
5.	Pembuatan Aplikasi (<i>Coding</i>)				
6.	Pengujian Aplikasi				
7.	Implementasi				

Keterangan :

	= Telah dilaksanakan
	= Belum dilaksanakan

C. Alat dan Bahan

Untuk mengimplementasikan desain sistem ini, penelitian ini tentunya menggunakan alat dan bahan pada penelitian. Pada penelitian ini digunakan perangkat keras (*hardware*) serta perangkat lunak (*software*).

Adapun perangkat keras (*hardware*) yang digunakan diantaranya *Personal Computer* (PC/Laptop) dengan spesifikasi *processor* intel core i7, RAM 8GB, SSD 256GB. Untuk perangkat lunak (*software*) yang digunakan yaitu sistem operasi Windows 10, Visual Studio Code, *Browser*, dan XAMPP.

D. Teknik Pengumpulan Data

Dalam penelitian ini, teknik pengumpulan data yang digunakan yaitu Alur dan prosedur proses penelitian ini dilakukan dengan bertahap diantaranya :

1. Studi Literatur

Langkah pertama dalam proses penelitian adalah studi literatur, yang meliputi pencarian, pengumpulan, dan menilai bahan bacaan yang relevan yang berkaitan dengan topik penelitian yang akan dilakukan. Tujuan utama dari langkah studi literatur adalah untuk memahami apa yang sebelumnya telah diketahui tentang masalah penelitian, mengidentifikasi kesenjangan pengetahuan, dan meletakkan dasar teori untuk penelitian yang akan dilakukan.

2. Pengumpulan Data (Observasi dan Wawancara)

Pada tahap ini dilakukan pengumpulan data-data relevan yang berkaitan dengan penelitian *booking service* dan penjualan produk seperti data-data produk yang dijual PT. Mitra Makassar, layanan yang ada, dan sebagainya.

3. Analisis Sistem

Pada langkah ini, analisis sistem dilakukan dengan berfokus pada kesulitan dan tujuan pengembangan aplikasi.

4. Perancangan Sistem

Pada tahap ini dilakukan perancangan sistem yang akan dikembangkan berdasarkan hasil analisis sistem sebelumnya.

5. Pembuatan Aplikasi (*Coding*)

Pada tahap ini, membuat desain antarmuka dan aplikasi *dengan* menggunakan bahasa pemrograman *JavaScript* dengan *framework Next Js*.

6. Pengujian Aplikasi

Pada tahap ini, pengujian aplikasi dilakukan untuk menguji fungsi-fungsi aplikasi yang telah dirancang untuk menunjukkan kelayakan sistem untuk digunakan.

7. Implementasi

Pada tahap ini, implementasi dilakukan untuk menjalankan aplikasi yang telah dirancang selama proses pengujian system

E. Jenis Data

Data yang digunakan dalam penelitian ini berupa data-data yang telah dikumpulkan melalui Penelitian Pustaka (*Library Research*). Adapun jenis data primer dan data sekunder yang relevan dengan masalah yang akan dibahas.

a. Data primer

Data Primer adalah data yang berasal atau data yang diperoleh langsung dari sumber data dan pengetahuan.

b. Data Sekunder

Data sekunder adalah data yang diperoleh tidak secara langsung dari objek penelitian. Peneliti mendapatkan data yang sudah jadi dari internet, *Website* dan jurnal.

F. Metode Pengujian Aplikasi

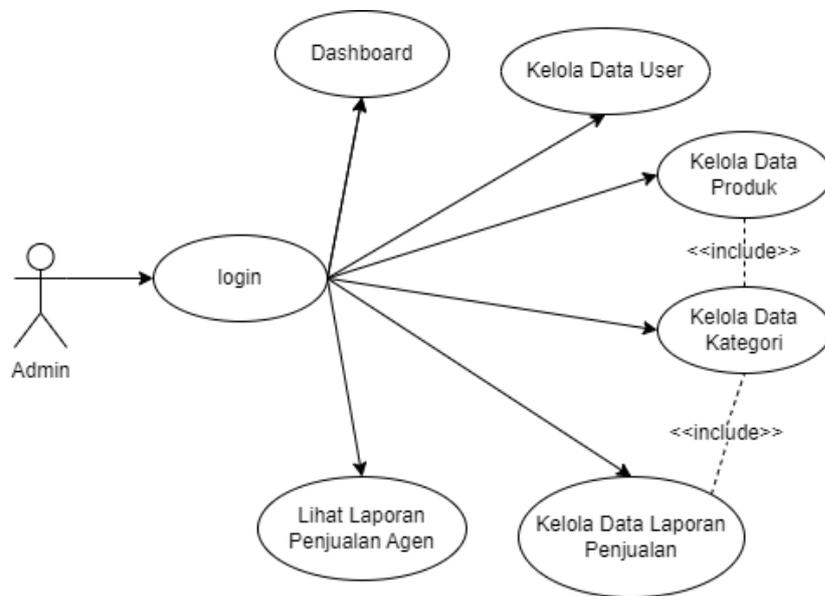
Dalam penelitian ini, metode pengujian yang digunakan untuk mengevaluasi aplikasi adalah secara eksklusif menggunakan pendekatan *Black box testing*. Pendekatan ini memungkinkan kami untuk menguji aplikasi tanpa perlu mengetahui struktur internal atau mekanisme kerja dari kode program yang terlibat. Pengujian ini fokus pada fungsi dan respons aplikasi berdasarkan input yang diberikan dan output yang dihasilkan, memastikan bahwa aplikasi beroperasi sesuai dengan spesifikasi yang ditentukan tanpa mempertimbangkan aspek internal system.

BAB IV

HASIL DAN PEMBAHASAN

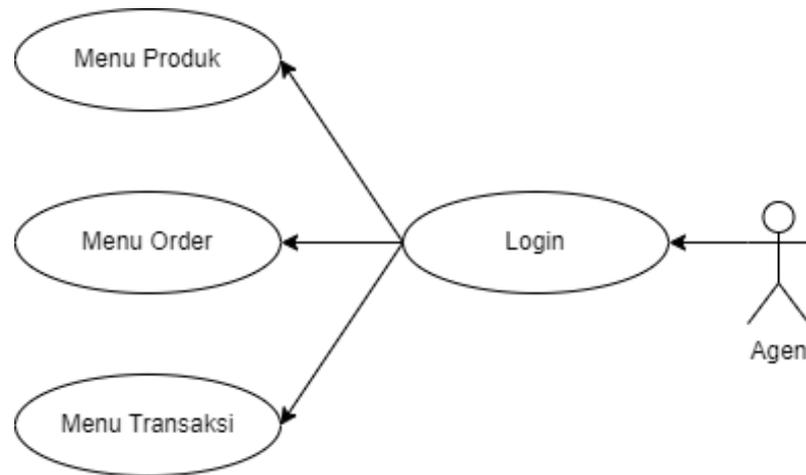
A. Analisis Sistem

1. Usecase Diagram



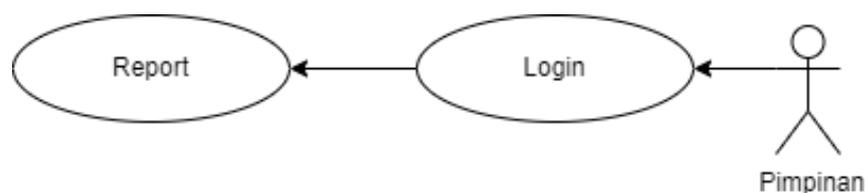
Gambar 4. 1 Usecase Diagram Admin

Pada rancangan *usecase* diatas menggambarkan fitur yang dapat diakses oleh admin setelah *login* ke *Dashboard* aplikasi. Fungsi-fungsi ini termasuk mengelola data pengguna, produk, kategori produk, kelola laporan penjualan dan melihat laporan penjualan agen. Semua operasi ini dapat diakses dari *Dashboard* utama setelah proses *login* berhasil, memungkinkan admin untuk melakukan pengaturan, pembaruan, dan analisis data terkait dengan aplikasi.



Gambar 4. 2 *Usecase Diagram Agen*

Pada rancangan *usecase* diatas, untuk role Agen memiliki fitur untuk *login*, menu produk, menu order dan menu transaksi. Semua menu ini terhubung langsung ke proses *login*, menunjukkan bahwa akses ke menu-menu ini membutuhkan autentikasi terlebih dahulu. Diagram ini menggambarkan cara agen berinteraksi dengan aplikasi untuk mengelola produk, order, dan transaksi.



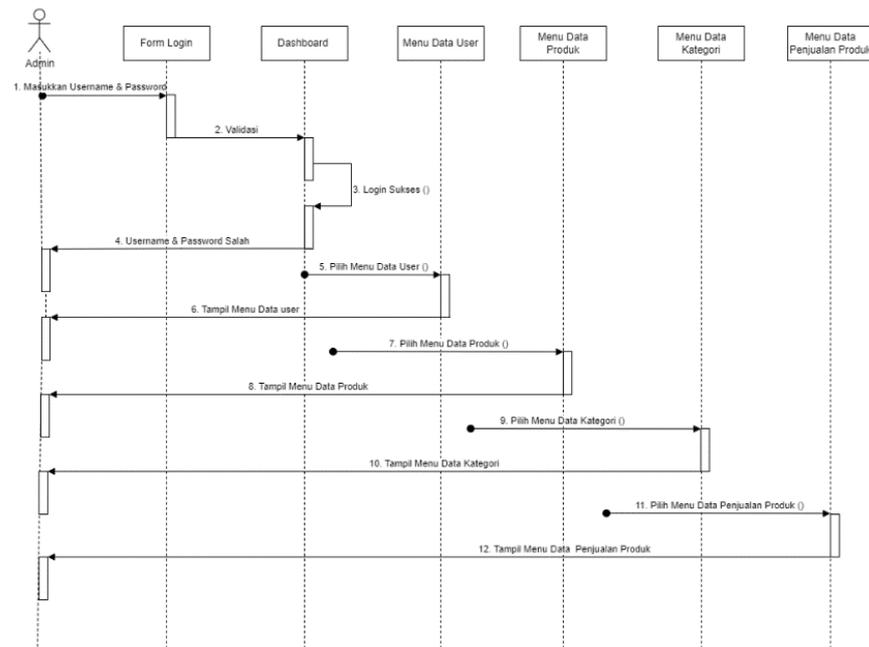
Gambar 4. 3 *Usecase Diagram Pimpinan*

Pada rancangan diagram *usecase* ini menggambarkan interaksi seorang pimpinan dengan sistem untuk mengakses laporan. Proses ini dimulai ketika pimpinan melakukan *login* ke dalam sistem. Setelah berhasil masuk, pimpinan kemudian memiliki akses untuk melihat dan mengelola laporan. Diagram ini

menunjukkan bahwa autentikasi melalui *login* adalah syarat mutlak untuk pimpinan dapat mengakses dan berinteraksi dengan fungsi laporan dalam sistem. Ini menggarisbawahi pentingnya proses *login* dalam mengamankan akses ke data sensitif dan penting seperti laporan.

2. Sequence Diagram

a. Sequence Diagram Admin

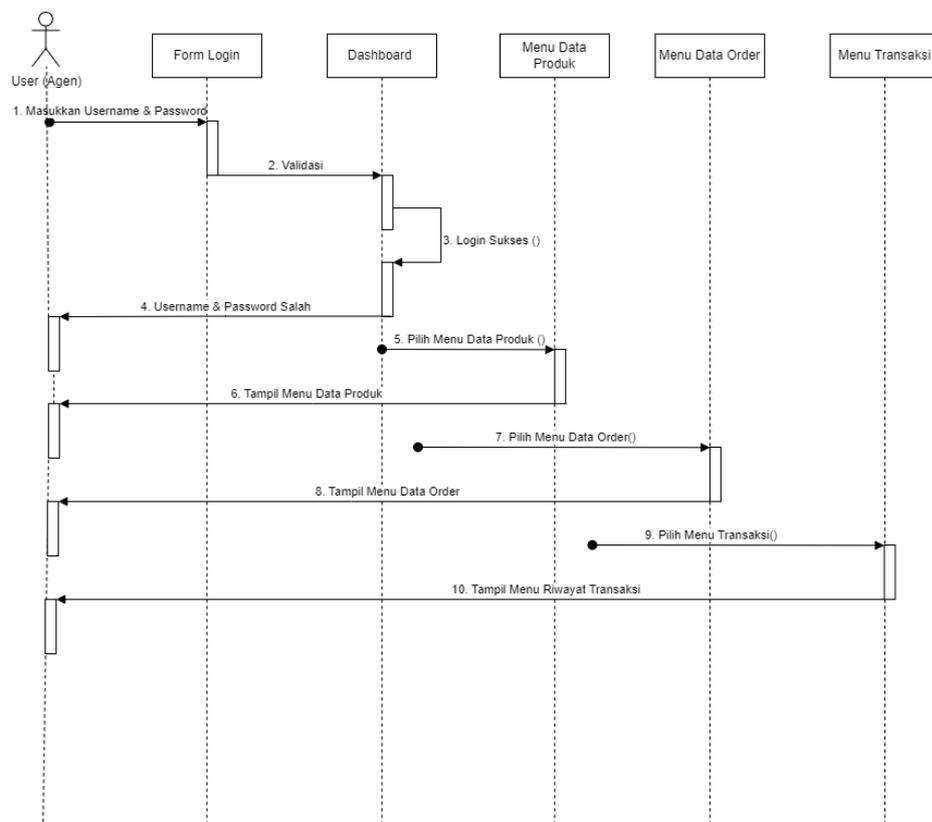


Gambar 4. 4 Sequence Diagram Admin

Pada gambar 4.4 terdapat *Sequence Diagram* admin. Dimana, terdapat 1 aktor yaitu admin serta 7 object yaitu form *login*, *Dashboard*, menu data *User*, data data produk, data kategori, dan data penjualan produk. Pertama, aktor akan masuk ke form *login* dan memasukkan *Username & password*. Setelah itu, akan dilakukan

validasi. Jika valid, maka akan masuk ke *Dashboard*. Jika salah satu dari *Username* atau *password* salah, maka *login* gagal. Selanjutnya, admin akan dapat mengakses menu-menu yang ada pada *Dashboard* setelah *login* berhasil sesuai dengan *Sequence Diagram* yang ada pada gambar.

b. *Sequence Diagram User*

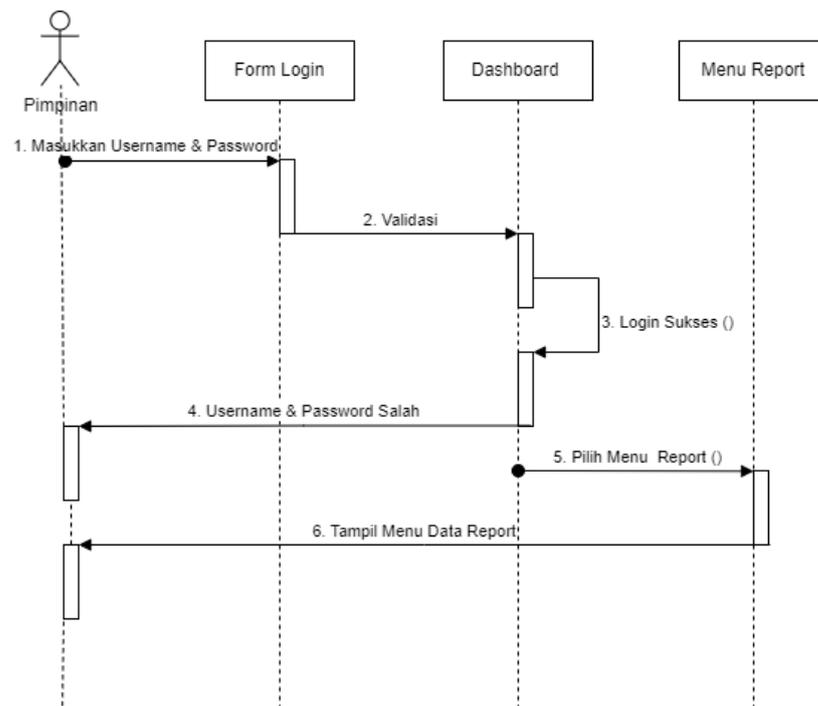


Gambar 4. 5 *Sequence Diagram User*

Pada gambar 4.5 terdapat *Sequence Diagram User*. Dimana, terdapat 1 aktor yaitu *User (Agen)* serta 5 object yaitu *form login*, *Dashboard*, menu produk, menu data order, serta menu transaksi. Aktor akan memulai dengan memasuki form

login dan menginputkan *usrname* serta *password*. Tahap selanjutnya melibatkan proses validasi, di mana sistem akan memeriksa keabsahan informasi yang dimasukkan. Apabila informasi valid, *User* akan diarahkan ke *Dashboard*. Namun, jika validasi tidak berhasil, yang dapat disebabkan oleh kesalahan pada *Username* atau *password*, proses *login* akan mengalami kegagalan. Setelah berhasil *login*, aktor akan dapat mengakses berbagai menu yang tersedia di dalam *Dashboard User*, sejalan dengan urutan langkah yang terdapat pada *Sequence Diagram* yang terlihat pada gambar 4.5.

c. *Sequence Diagram* Pimpinan

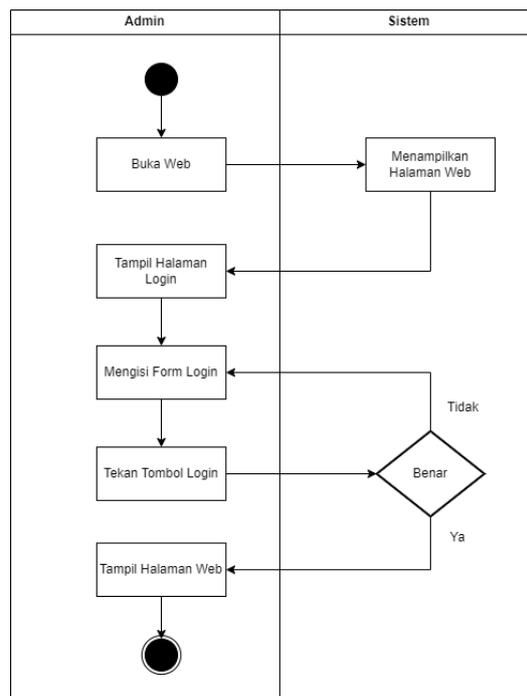


Gambar 4. 6 *Sequence Diagram* Pimpinan

Untuk gambar 4.6 terdapat *Sequence Diagram* pimpinan. Dalam hal ini, pimpinan hanya memiliki 3 object diantaranya form *login*, *Dashboard*, menu report. Sama seperti sebelumnya, pimpinan terlebih dahulu akan masuk pada form *login*. Dan setelah validasi berhasil, maka pimpinan dapat mengakses 2 objek yang ada sesuai dengan gambar.

3. Activity Diagram

a. Activity Diagram Login

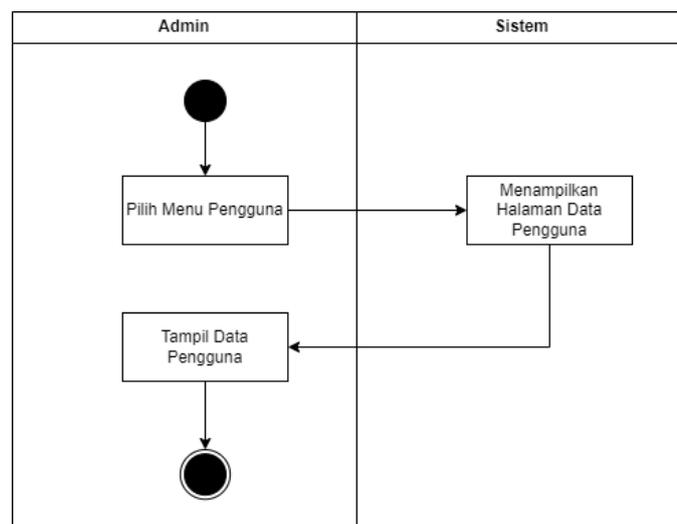


Gambar 4. 7 Activity Diagram Login

Pada gambar 4.7 terdapat diagram aktivitas *login* yang dilakukan oleh semua role, yaitu admin dan *User*. Diagram aktivitas *login* ini dimulai ketika *User* membuka situs web, yang kemudian direspon oleh sistem dengan

menampilkan halaman web berisi formulir *login*. Selanjutnya, *User* mengisi formulir dengan informasi *login* seperti *Username* dan *password*. Langkah berikutnya melibatkan validasi informasi *login* oleh sistem. Jika informasi yang dimasukkan *User* dikonfirmasi kebenarannya oleh sistem, maka aktivitas dilanjutkan dengan menampilkan halaman *Dashboard*, menandakan bahwa *login* berhasil dan *User* dapat mengakses *Dashboard*. Namun, jika terdapat ketidakcocokan atau kesalahan dalam informasi *login*, maka diambil langkah "*Login Gagal*". Diagram aktivitas *login* diakhiri dengan simbol akhir yang menunjukkan penyelesaian proses *login*. Dengan menggunakan diagram ini, alur kerja dari membuka situs web hingga mencapai halaman *Dashboard* atau kembali ke formulir *login* dapat dipahami dengan jelas dan divisualisasikan.

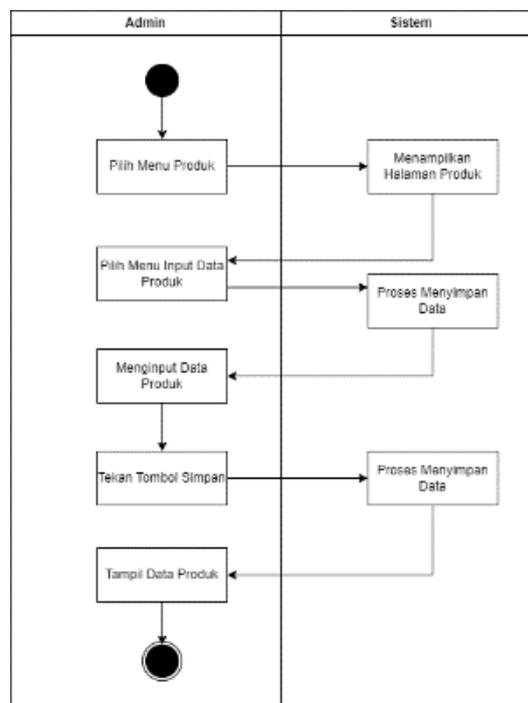
b. *Activity Diagram Data User*



Gambar 4. 8 *Activity Diagram Data User*

Diagram aktivitas admin untuk menu data *User* dimulai ketika admin memilih menu "Data *User*". Sistem kemudian merespons dengan menampilkan halaman data pengguna. Admin selanjutnya melihat tampilan halaman data pengguna menandakan bahwa proses telah selesai

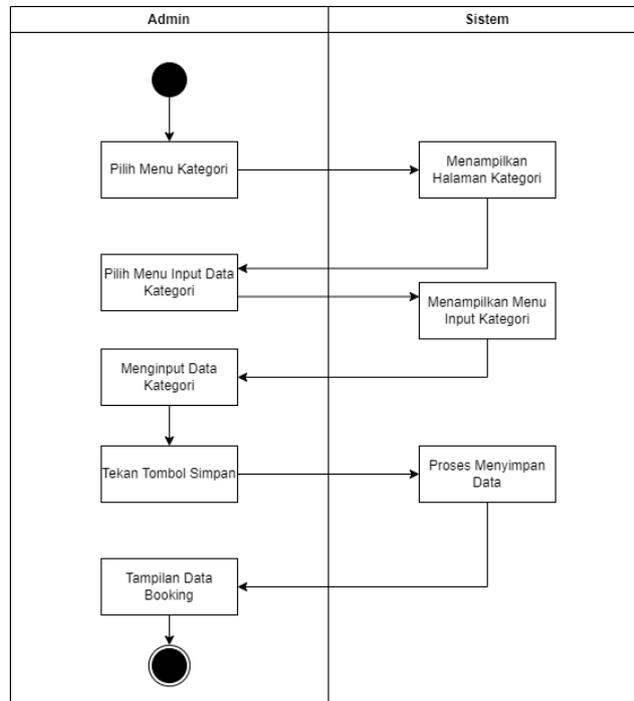
c. *Activity Diagram Data Produk*



Gambar 4. 9 *Activity Diagram Data Produk*

Diagram aktivitas data produk melibatkan beberapa langkah mulai dari pemilihan menu, tampilan dan input data produk, penyimpanan data oleh sistem, dan tampilan data produk telah selesai, sesuai dengan gambar 4.9.

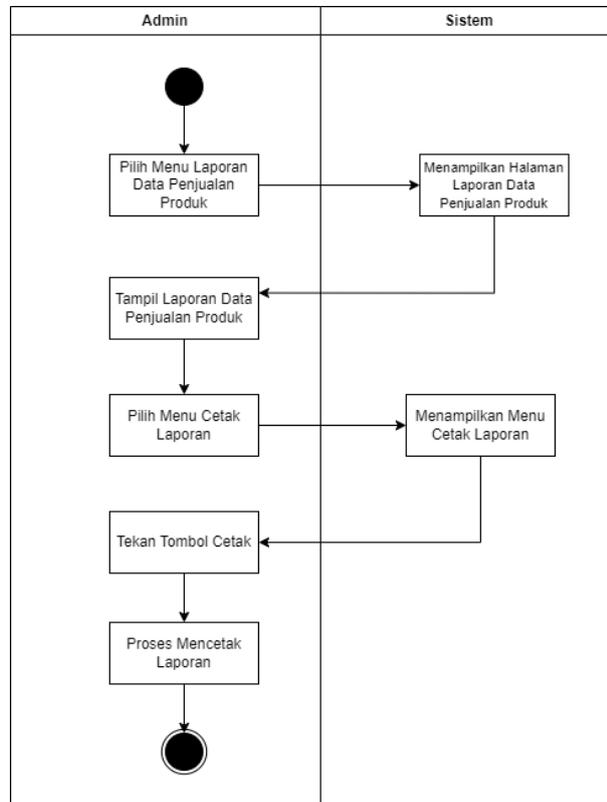
d. *Activity Diagram* Kategori



Gambar 4. 10 *Activity Diagram* Kategori

Activity Diagram admin untuk menu data kategori dimulai dengan admin memilih menu kategori. Sistem merespons dengan menampilkan halaman data kategori. Admin melihat tampilan halaman data kategori dan melakukan input data kategori. Sistem kemudian menampilkan form input kategori dan admin akan menginput data kategori. Setelah input data kategori selesai, admin menekan tombol "Simpan". Sistem kemudian memproses dan menyimpan data kategori. Setelah itu, terdapat simbol object destruction yang menunjukkan bahwa proses tampilan data kategori selesai

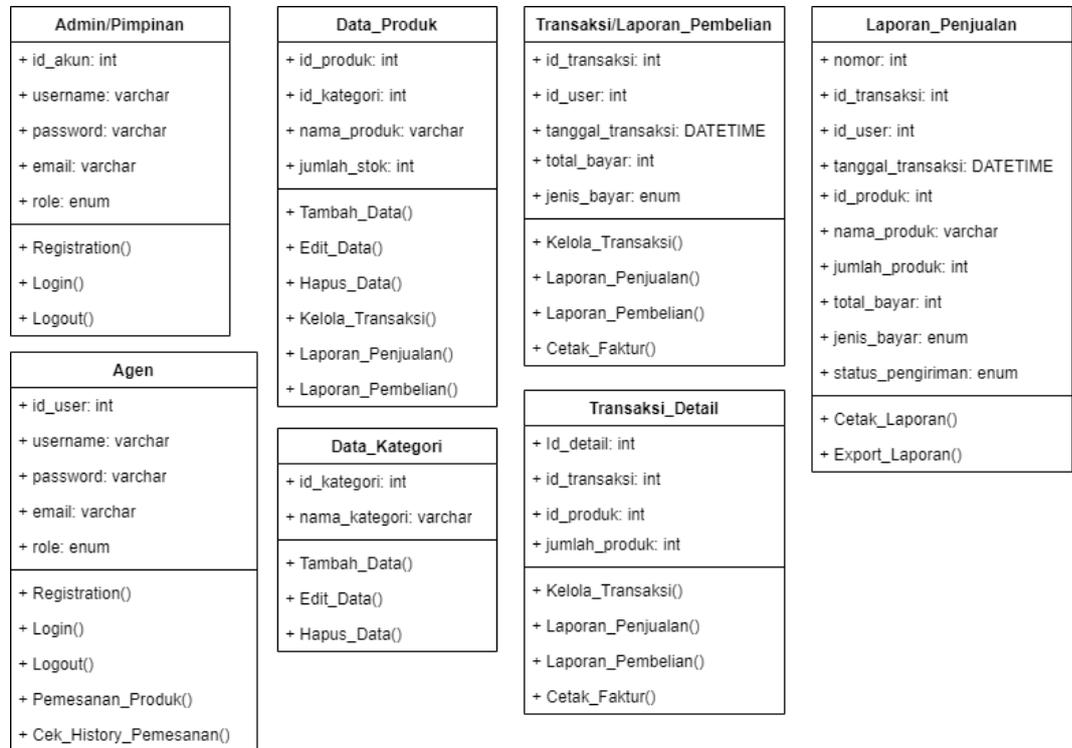
e. *Activity Diagram* Data Penjualan Produk



Gambar 4. 11 *Activity Diagram* Data Penjualan Produk

Activity Diagram data laporan penjualan produk (gambar 4.11) prosesnya sama. *Activity Diagram* dimulai dengan admin memilih menu laporan data penjualan produk. Sistem merespons dengan menampilkan halaman data laporan penjualan produk. Admin melihat tampilan halaman data laporan penjualan produk. Admin dapat melakukan tindakan mencetak laporan. Sistem kemudian memproses dan mencetak data laporan penjualan produk. Setelah itu, terdapat simbol object destruction yang menunjukkan bahwa proses selesai.

4. Class Diagram



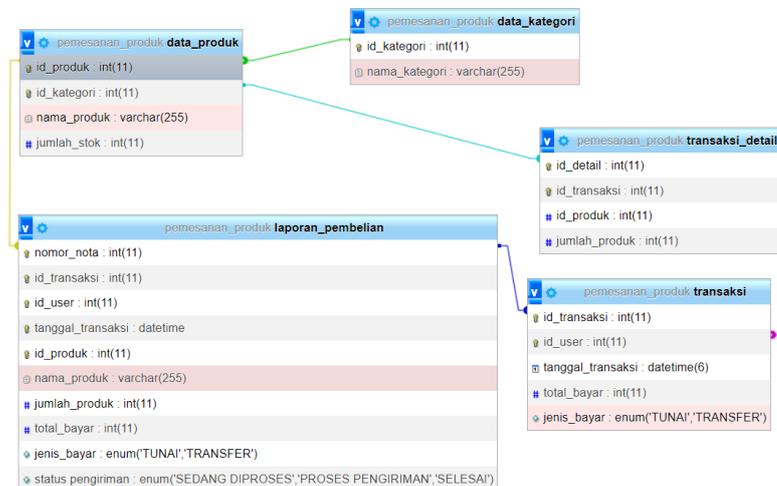
Gambar 4. 12 Class Diagram

Class Diagram adalah jenis diagram dalam UML (*Unified Modeling Language*) yang digunakan untuk menggambarkan struktur dan hubungan antara kelas dalam suatu sistem perangkat lunak. *Class Diagram* adalah salah satu diagram UML yang paling umum digunakan dan memberikan pandangan yang kuat tentang elemen-elemen kunci dalam desain sistem perangkat lunak.

Pada Gambar 4.12 terdapat rancangan *Class Diagram* untuk aplikasi yang nantinya akan dibuat. Untuk Admin/Pimpinan, kelas ini mewakili pengguna dengan peran admin atau pimpinan yang memiliki akses untuk mengelola data

produk, transaksi, dan laporan. Sedangkan, Agen, kelas ini dapat melakukan pemesanan barang serta mengecek history atau riwayat pemesanan. Selain itu, terdapat kelas-kelas lainnya yaitu data produk, data kategori, data transaksi/laporan pembelian, data transaksi detail, serta laporan penjualan.

Selain itu, terdapat diagram relasi atau *Entity Relationship Diagram* (ERD) untuk merancang dan memahami *Database* untuk sistem yang nantinya akan dibuat. Diagram ini memvisualisasikan hubungan antar tabel dalam *Database*, dan juga membantu dalam memahami struktur data dan bagaimana data terhubung satu sama lain. Diagram ini dapat dilihat pada gambar berikut.



Gambar 4. 13 *Diagram Entity Relationship Diagram* (ERD)

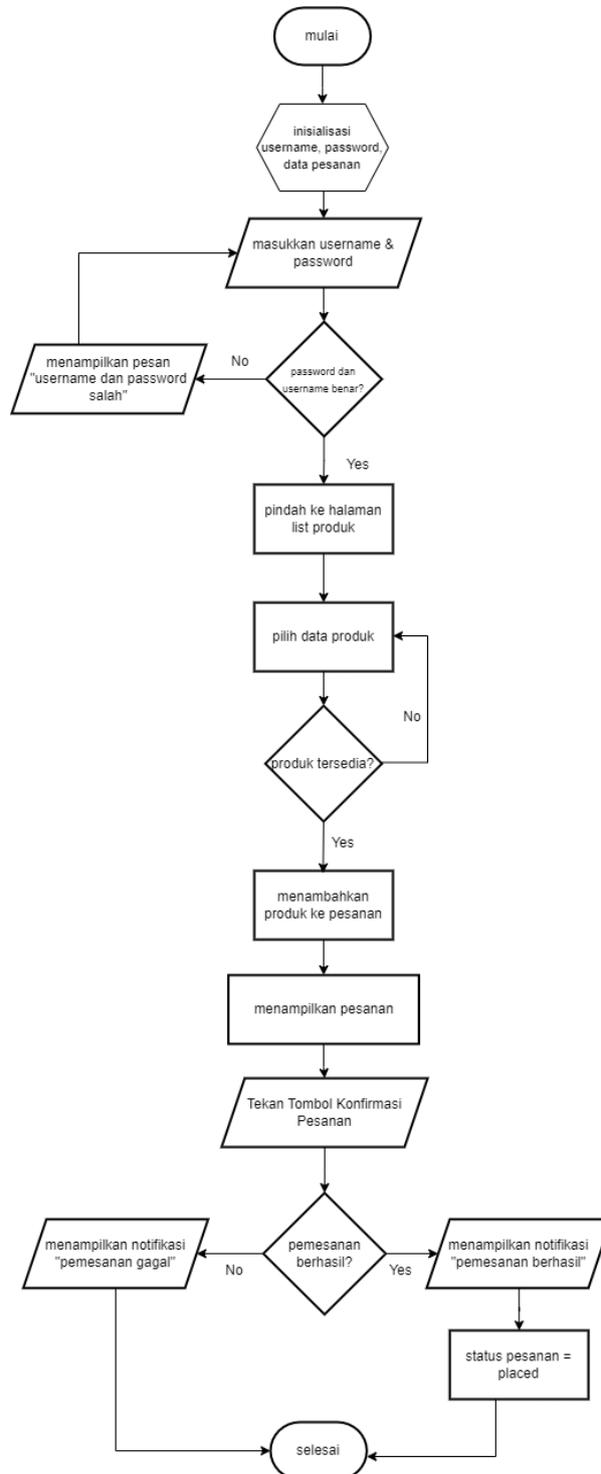
Pada gambar 4.13 terdapat ERD dari sistem yang akan dibuat. Dimana, pada gambar tersebut terdapat masing-masing tabel yang berelasi dengan masing-masing *primary key*. Untuk tabel *data_produk* terdapat *primary key* yaitu

id_produk. Kemudian, data_kategori, *primary key*-nya adalah id_kategori. Selanjutnya, tabel_transaksi yang memiliki *primary key* yaitu id_transaksi, tabel_transaksi_detal *primary key*-nya yaitu id_detail, dan laporan_pembelian yang *primary key*-nya adalah nomor_nota. Untuk data produk yang nantinya akan dimasukkan dalam sistem, dapat dilihat pada gambar dibawah ini.

PT. Mitra Makassar			
No.	Brand Produk	Jenis Produk	Nama Produk
1	Paseo	Tisu Wajah	Paseo Smart Facial Softpack 250'S
2	Paseo	Tisu Wajah	Paseo Smart Facial Softpack 540'S
3	Paseo	Tisu Wajah	Paseo Smart Facial Travel Pack 50'S
4	Paseo	Tisu Wajah	Paseo Smart Facial Hanky Standard 6 Pack 12'S
5	Paseo	Tisu Dapur	Paseo Cooking Towel 3 Rolls 70'S
6	Paseo	Tisu Dapur	Paseo Towell Roll White Emboss 2 Rolls 70'S
7	Paseo	Tisu Dapur	Paseo Cooking Towel 1 Rolls 70'S
8	Paseo	Tisu Bayi	Paseo Baby Pure Soft Soft Pack 130'S
9	Paseo	Tisu Bayi	Paseo Baby Wipes Gazette 50'S Jojoba Oil
10	Paseo	Tisu Bayi	Paseo Baby Wipes Gazette 50'S Chamomile Oil
11	Nice	Tisu Wajah	Nice Facial Living 250'S
12	Nice	Tisu Wajah	Nice Facial Living 360'S
13	Nice	Tisu Wajah	Nice Facial Living 180'S
14	Nice	Tisu Dapur	Nice Kitchen Roll 150'S
15	Nice	Tisu Dapur	Nice Kitchen Roll 238'S
16	Jolly	Tisu Wajah	Jolly Facial Softpack 250'S
17	Jolly	Tisu Wajah	Jolly Facial Softpack 560'S
18	Jolly	Tisu Wajah	Jolly Facial Pop Up 150'S
19	Jolly	Tisu Dapur	Jolly Handtowel Napkins 250'S
20	Jolly	Tisu Dapur	Jolly Coreless 10 Roll
21	Topy	Tisu Dapur	Topy Napkins 50'S

Gambar 4. 14 Data Produk

5. Flowchart



Gambar 4. 15 Flowchart

B. Rancangan Database

1. Tabel data user

Tabel 4. 1 Data User

Field	Type	Null	Key	Default	Ekstra
id	Int		primary		
email	Varchar(255)	Tidak			
username	Varchar(255)	Tidak			
password	Varchar(255)	Tidak			
role	enum (agen, admin, pimpinan)			'agen'	
nomorTelepon	Varchar(255)	Tidak			
createdAt	timestamp				
updateAt	timestamp				

2. Tabel data produk

Tabel 4. 2 Data Produk

Field	Type	Null	Key	Default	Ekstra
id	Char(36)		primary		
id_kategori	Int	Tidak			
nama_product	Varchar(255)	Tidak			
jumlah_stok	Int	Tidak			
harga	Int	Tidak			
image_url	Varchar(255)				

3. Tabel da kategori

Tabel 4. 3 Data Kategori

Field	Type	Null	Key	Default	Ekstra
id	Int		primary		
nama_kategori	Varchar(255)	Tidak			

4. Tabel transaksi

Tabel 4. 4 Data transaksi

Field	Type	Null	Key	Default	Ekstra
id	Char(36)		primary		
id_user	Int	Tidak			
kode_transaksi	Varchar(255)	Tidak			
total_bayar	Int	Tidak			
status_pengiriman	enum (placed, rejected, shipped, received)	Tidak		'agen'	
jenis_bayar	enum (cash, transfer)	Tidak			
keterangan	Text				
createdAt	timestamp				
updateAt	timestamp				
alamat	text				

5. Tabel detail transaksi

Tabel 4. 5 Detail Transaksi

Field	Type	Null	Key	Default	Ekstra
Id	Char(36)		primary		
id_transaksi	Char(36)	Tidak			
id_produk	Char(36)	Tidak			
qty	Int	Tidak			
harga_satuan	Int	Tidak		'agen'	
total_harga	int	Tidak			
createdAt	Datetime				
updateAt	datetime				

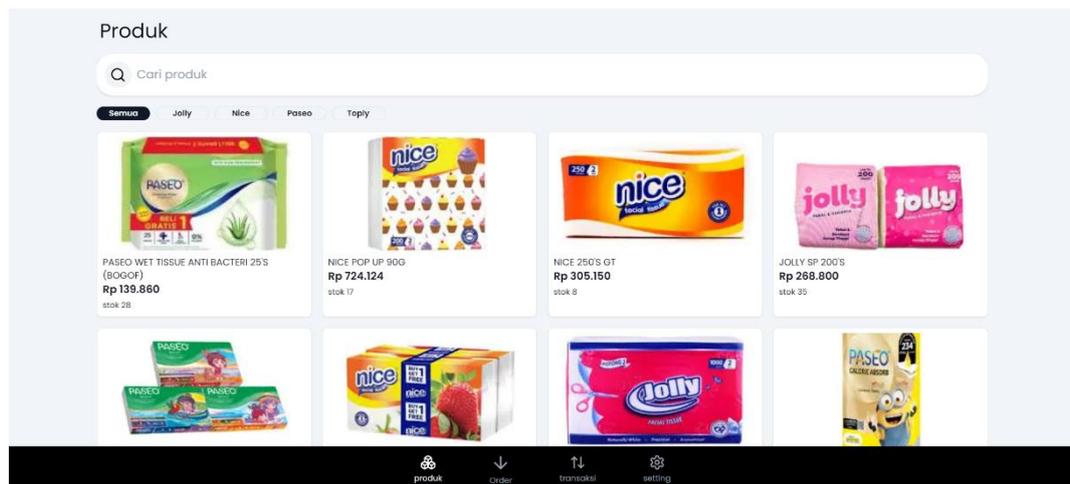
6. Tabel setting

Tabel 4. 6 Tabel Setting

Field	Type	Null	Key	Default	Ekstra
Id	Int		primary		
nama_brand	Varchar(255)	Tidak			
alamat	Text	Tidak			
nomorTelepon	Varchar(20)	Tidak			
email	Varchar(255)	Tidak		'agen'	
nomorRekening	Varchar(255)	Tidak			
pemilikRekening	Varchar(255)	Tidak			
bank	Varchar(255)				
createdAt	Datetime				
updateAt	datetime				

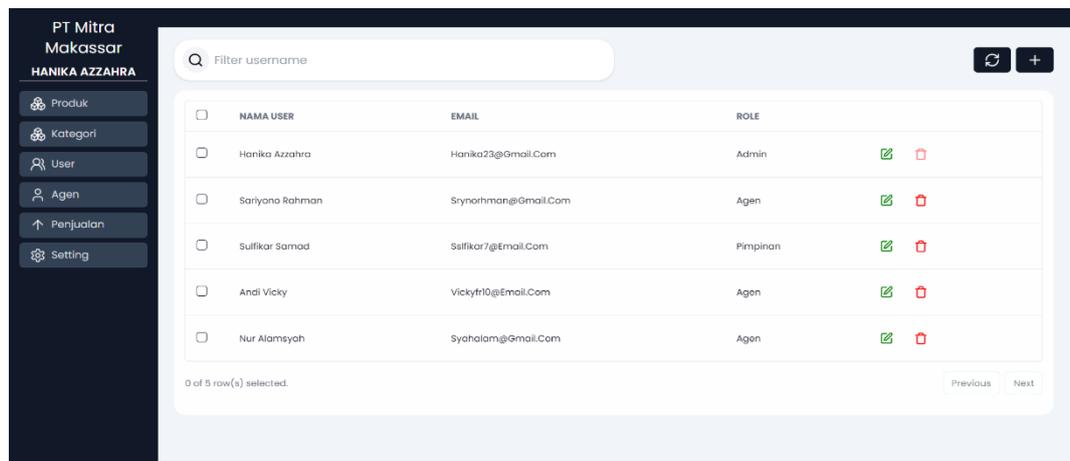
C. Detail Aplikasi

1. Tampilan Utama Aplikasi



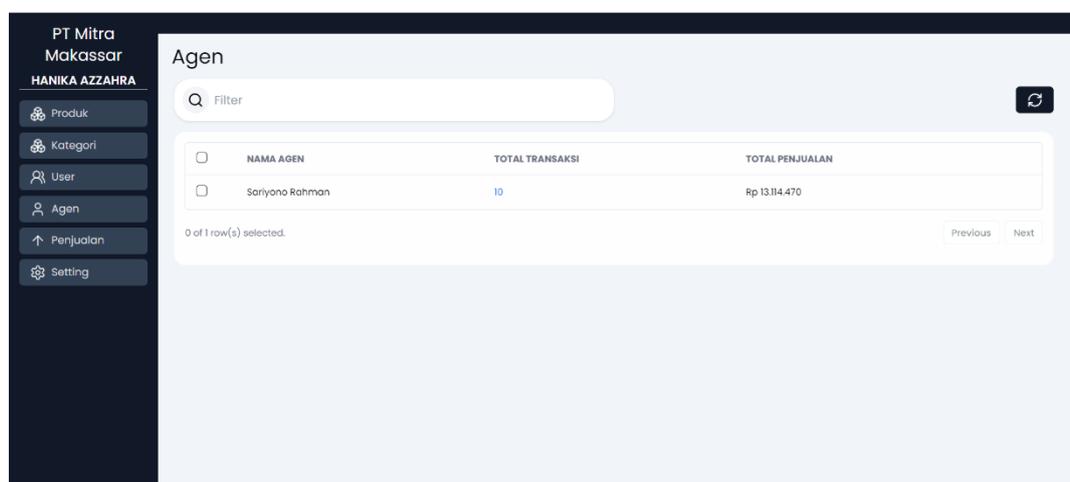
Gambar 4. 16 Tampilan Utama Aplikasi

Gambar diatas merupakan tampilan utama dari PT Mitra Makassar untuk mengelola. Pengguna dapat melihat berbagai produk yang tersedia, lengkap dengan informasi seperti nama produk, harga, dan stok yang tersedia. Di bagian atas, terdapat fitur pencarian yang memungkinkan pengguna untuk mencari produk berdasarkan kata kunci. Pengguna juga dapat memfilter produk berdasarkan merek dengan menggunakan tombol filter yang tersedia.



Gambar 4. 17 Tampilan Menu *User* pada *Dashboard*

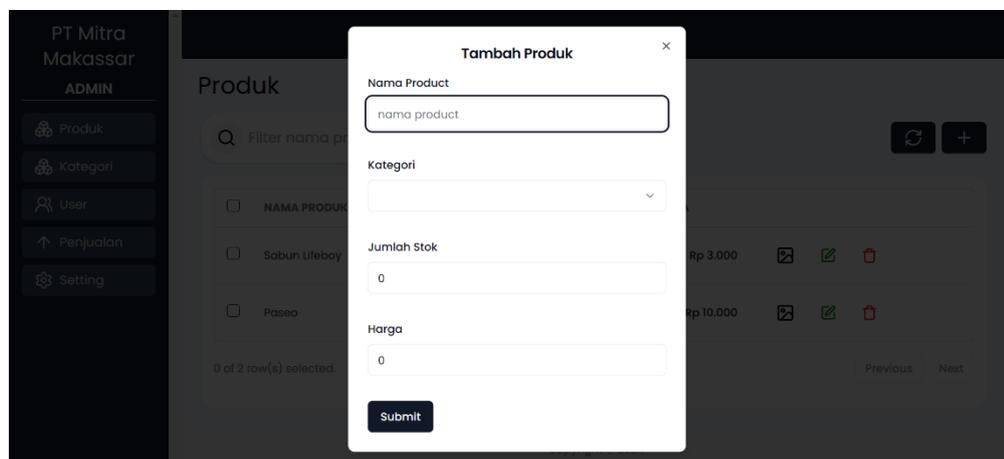
Gambar 4.16 merupakan menu *User* dalam *Dashboard* aplikasi PT Mitra Makassar. Pada menu ini, pengguna dapat melihat daftar individu yang memiliki akses ke aplikasi, lengkap dengan informasi seperti nama pengguna nama *user*, alamat email, dan peran (role) masing-masing pengguna. Selain itu, pengguna dapat mengelola informasi setiap individu dengan menggunakan ikon edit untuk memperbarui data, atau ikon hapus untuk menghapus akses pengguna tersebut dari sistem.



Gambar 4. 18 Tampilan Menu *Agen* pada *Dashboard*

Gambar di atas menunjukkan menu agen dalam aplikasi PT Mitra Makassar. Pada menu ini, pengguna dapat melihat daftar agen yang terdaftar beserta rincian transaksi mereka. Setiap agen ditampilkan dengan informasi seperti Nama Agen, Total Transaksi, dan Total Penjualan yang telah mereka capai.

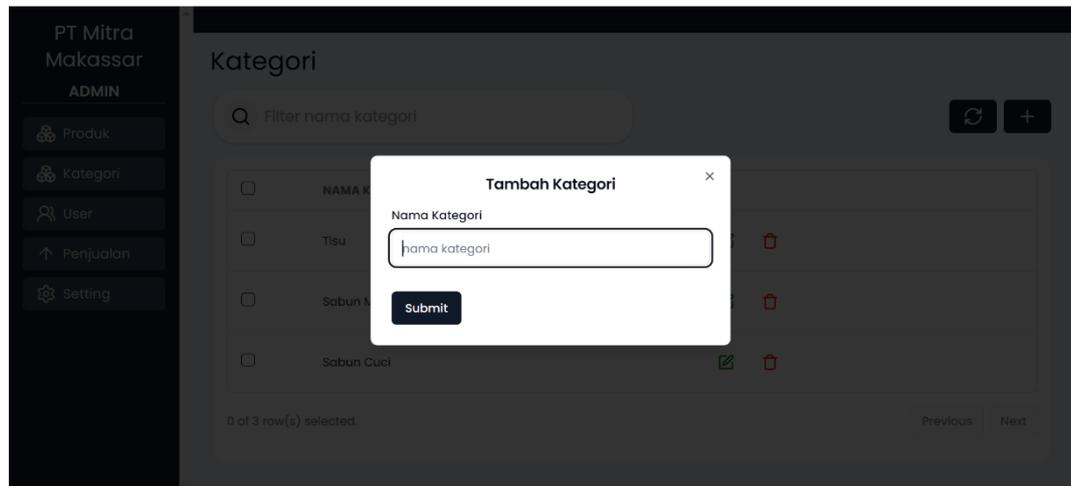
2. Input Data Produk



Gambar 4. 19 Tampilan Input Data Produk pada *Dashboard*

Pada gambar diatas menunjukkan tampilan pop-up untuk menambahkan produk baru. Dalam penginputan produk terdapat informasi yang harus diisi seperti nama produk, kategori, jumlah stok dan harga. Dibagian bawah terdapat tombol “*submit*” untuk menyimpan data produk yang telah diinput.

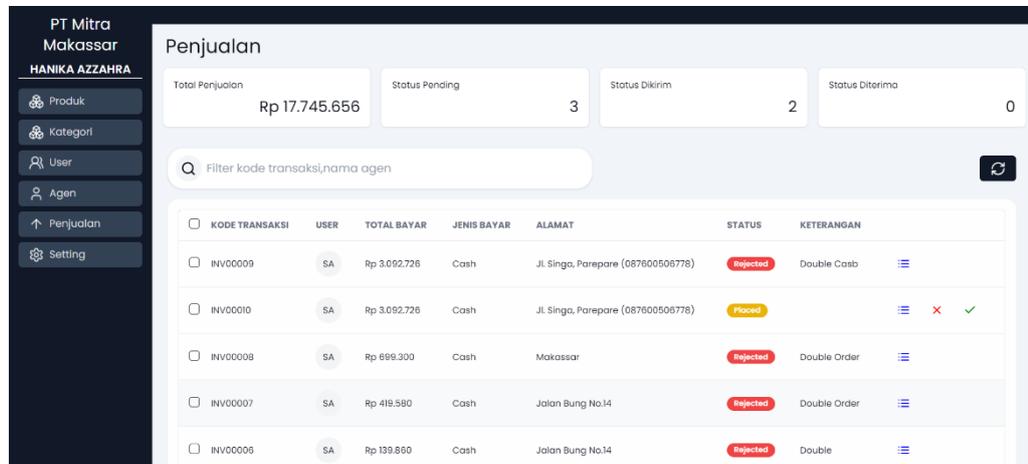
3. Input Data Kategori



Gambar 4. 20 Tampilan Input Data Kategori pada Dashboard

Pada gambar diatas menunjukkan tampilan pop-up untuk menambahkan kategori berdasarkan jenis produk. Di bagian bawah form, terdapat tombol "Submit" untuk menyimpan data kategori yang telah diinput. Pop-up ini muncul di atas halaman utama yang menampilkan daftar kategori yang sudah ada, dengan fitur pencarian dan opsi untuk mengedit atau menghapus kategori. Antarmuka ini dirancang untuk memudahkan admin dalam menambah kategori baru secara cepat dan efisien.

4. Output Aplikasi



The screenshot shows the 'Penjualan' (Sales) dashboard for PT Mitra Makassar. The user is HANIKA AZZAHRA. The dashboard displays a summary of sales with the following data:

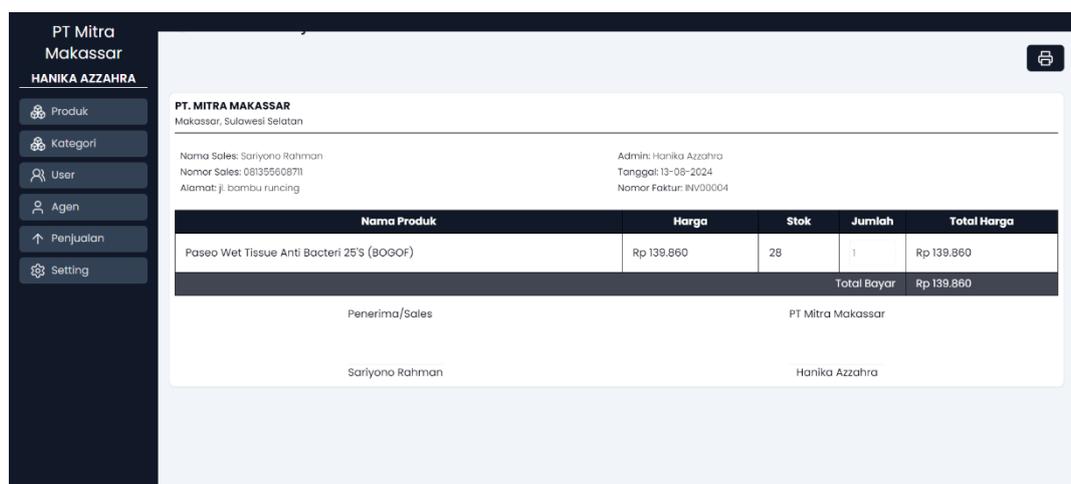
Total Penjualan	Status Pending	Status Dikirim	Status Diterima
Rp 17.745.656	3	2	0

Below the summary is a search bar with the placeholder text 'Filter kode transaksi, nama agen' and a refresh button. The main part of the dashboard is a table of transactions:

KODE TRANSAKSI	USER	TOTAL BAYAR	JENIS BAYAR	ALAMAT	STATUS	KETERANGAN
INV0009	SA	Rp 3.092.726	Cash	Jl. Singa, Parepare (087600506778)	Rejected	Double Cash
INV0010	SA	Rp 3.092.726	Cash	Jl. Singa, Parepare (087600506778)	Placed	
INV0008	SA	Rp 699.300	Cash	Makassar	Rejected	Double Order
INV0007	SA	Rp 419.580	Cash	Jalan Bung No.14	Rejected	Double Order
INV0006	SA	Rp 139.860	Cash	Jalan Bung No.14	Rejected	Double

Gambar 4. 21 Tampilan Rekap Penjualan pada *Dashboard*

Pada gambar diatas menunjukkan halaman rekap penjualan. Bagian utama menampilkan tabel transaksi penjualan. Tabel ini juga dilengkapi dengan ikon untuk melihat detail, menghapus, dan mengonfirmasi transaksi. Di atas tabel terdapat kotak pencarian untuk memfilter transaksi berdasarkan kode, serta tombol refresh untuk memuat ulang data. Antarmuka ini memudahkan admin dalam mengelola dan memantau transaksi penjualan.



The screenshot shows the 'Faktur' (Invoice) page for PT. MITRA MAKASSAR. The user is HANIKA AZZAHRA. The page displays a summary of the invoice with the following data:

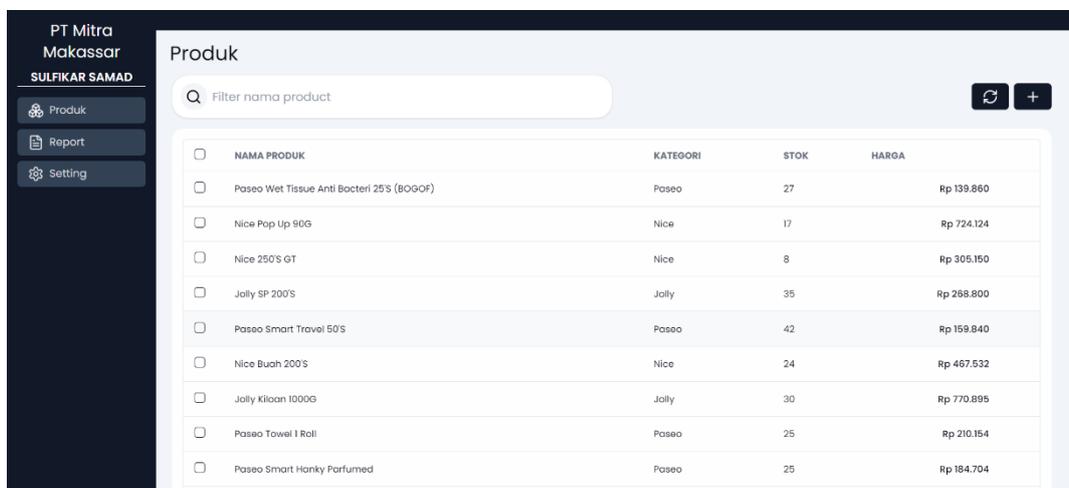
Nama Produk	Harga	Stok	Jumlah	Total Harga
Paseo Wet Tissue Anti Bacteri 25'S (BOGOR)	Rp 139.860	28	1	Rp 139.860
Total Bayar				Rp 139.860

Below the table, the recipient and sender information is displayed:

Penerima/Sales	PT Mitra Makassar
Sariyono Rahman	Hanika Azzahra

Gambar 4. 22 Tampilan Faktur pada *Dashboard*

Gambar tersebut menunjukkan halaman faktur pada aplikasi web admin PT Mitra Makassar. Halaman ini menampilkan informasi detail penjualan termasuk nama dan nomor sales, alamat, admin yang bertugas, tanggal, dan nomor faktur. Tabel di bagian tengah menampilkan daftar produk yang dipesan, harga, stok, jumlah yang dipesan, dan total harga. Antarmuka ini memudahkan admin untuk melihat detail setiap order dengan jelas dan terstruktur.



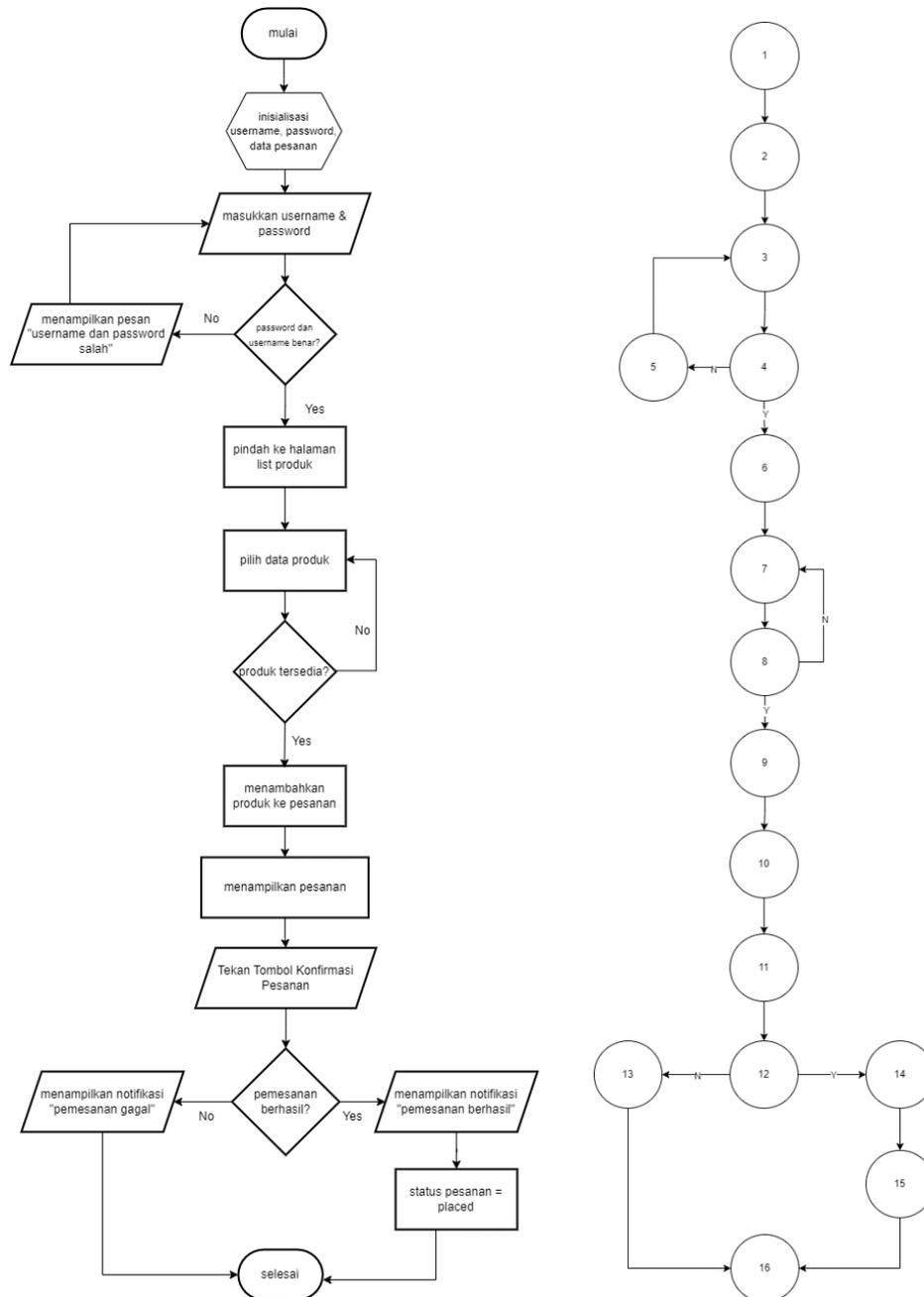
<input type="checkbox"/>	NAMA PRODUK	KATEGORI	STOK	HARGA
<input type="checkbox"/>	Paseo Wet Tissue Anti Bacteri 25'S (BOGOF)	Paseo	27	Rp 139.860
<input type="checkbox"/>	Nice Pop Up 50G	Nice	17	Rp 724.124
<input type="checkbox"/>	Nice 250'S GT	Nice	8	Rp 305.150
<input type="checkbox"/>	Jolly SP 200'S	Jolly	35	Rp 268.800
<input type="checkbox"/>	Paseo Smart Travel 50'S	Paseo	42	Rp 159.840
<input type="checkbox"/>	Nice Buah 200'S	Nice	24	Rp 467.532
<input type="checkbox"/>	Jolly Kilan 1000G	Jolly	30	Rp 770.895
<input type="checkbox"/>	Paseo Towel 1 Roll	Paseo	25	Rp 210.154
<input type="checkbox"/>	Paseo Smart Hanky Perfumed	Paseo	25	Rp 184.704

Gambar 4. 23 Tampilan Report Pimpinan pada *Dashboard*

Gambar diatas merupakan tampilan rekapan laporan penjualan dan faktur penjualan. Output tersebut nantinya dapat dicetak melalui print, pdf ataupun di *export* ke berbagai macam file

D. Pengujian Sistem

Pengujian aplikasi dilakukan dengan cara yaitu Pengujian *Black Box* dan *whitebox*:

1) Pengujian *White Box*

Gambar 4. 24 Flowchart dan Flowgraph Sistem Pemesanan Produk

Cyclomatic Complexity:

$$V(G) = 18 - 16 + 2$$

$$V(G) = 2 + 2$$

$$V(G) = 4$$

Tabel 4. 7 *Independent Path*

No	<i>Independent Path</i>
1.	1-2-3-4-5-3 (User menginputkan username dan password yang salah)
2.	1-2-3-4-6-7 (User menginputkan username dan password yang benar)
3.	1-2-3-4-6-7-8-7 (User memilih produk yang tidak tersedia)
4.	1-2-3-4-6-7-8-9-10-11 (User memilih produk yang tersedia dan berhasil menambahkan produk ke pesanan)
5.	1-2-3-4-6-7-8-9-10-11-12-13-16 (User gagal melakukan pemesanan produk)
6.	1-2-3-4-6-7-8-9-10-11-12-14-15-16 (User berhasil melakukan pemesanan produk)

Tabel 4. 8 *Test Case* Fitur Login

Path	1
Jalur	1-2-3-4-5-3
Skenario	<ol style="list-style-type: none"> 1. Mulai 2. Inisialisasi data 3. Input username dan password 4. Proses verifikasi yang sudah diinputkan 5. Input yang dimasukkan tidak sesuai dan menampilkan pesan “username dan password salah” 3 Input username dan password
Hasil Pengujian	Berhasil

Tabel 4. 9 *Test Case* Fitur Login

Path	2
Jalur	1-2-3-4-6-7
Skenario	<ol style="list-style-type: none"> 1. Mulai 2. Inisialisasi data 3. Input username dan password 4. Proses verifikasi yang sudah diinputkan 6 Username dan password berhasil di inputkan dan pindah ke halaman produk dan pindah ke halaman list produk 7 Pilih data produk

Hasil Pengujian	Berhasil
------------------------	----------

Tabel 4. 10 *Test Case* Fitur Pilih Produk

Path	3
Jalur	1-2-3-4-6-7-8-7
Jalur Skenario	<ol style="list-style-type: none"> 1. Mulai 2. Inisialisasi data 3. Input username dan password 4. Proses verifikasi yang sudah diinputkan 6 Username dan password berhasil di inputkan dan pindah ke halaman produk dan pindah ke halaman list produk 7 Pilih data produk 8 Proses verifikasi yang sudah diinputkan 7 Input yang dimasukkan tidak sesuai dan Kembali kehalaman list produk
Hasil Pengujian	Berhasil

Tabel 4. 11 *Test Case* Fitur Pilih Produk

Path	4
Jalur	1-2-3-4-6-7-8-9-10-11
Jalur Skenario	<ol style="list-style-type: none"> 1. Mulai 2. Inisialisasi data 3. Input username dan password 4. Proses verifikasi yang sudah diinputkan 6 Username dan password berhasil di inputkan dan pindah ke halaman produk 7 Pilih data produk 8 Proses verifikasi yang sudah diinputkan 9 Input berhasil dan menambahkan produk kepesanan 10 Menampilkan pesan 11 Tekan tombol konfirmasi pesanan
Hasil Pengujian	Berhasil

Tabel 4. 12 *Test Case* Fitur Pemesanan Produk

Path	5
Jalur	1-2-3-4-6-7-8-9-10-11-12-13-16
Jalur Skenario	<ol style="list-style-type: none"> 1. Mulai 2. Inisialisasi data 3. Input username dan password 4. Proses verifikasi yang sudah di inputkan 6 Username dan password berhasil di inputkan dan pindah ke halaman produk 7 Pilih data produk 8 Proses verifikasi yang sudah diinputkan 9 Input berhasil dan menambahkan produk kepesanan 10 Menampilkan pesanan 11 Tekan tombol kofirmasi pesanan 12 Proses verifikasi yang sudah diinputkan 13 Input yang dimasukkan tidak sesuai dan menampilkan notifikasi “pemesanan gagal” 16 Selesai
Hasil Pengujian	Berhasil

Tabel 4. 13 *Test Case* Fitur Pemesanan Produk

Path	6
Jalur	1-2-3-4-6-7-8-9-10-11-12-14-15-16
Jalur Skenario	<ol style="list-style-type: none"> 1. Mulai 2. Inisialisasi data 3. Input username dan password 4. Proses verifikasi yang sudah di inputkan 6 Username dan password berhasil di inputkan dan pindah ke halaman produk 7 Pilih data produk 8 Proses verifikasi yang sudah diinputkan 9 Input berhasil dan menambahkan produk kepesanan 10 Menampilkan pesanan 11 Tekan tombol kofirmasi pesanan 12 Proses verifikasi yang sudah diinputkan 14 Input yang dimasukkan sesuai dan menampilkan notifikasi “pemesanan berhasil”

	15 Status pesanan="placed" 16 Selesai
Hasil Pengujian	Berhasil

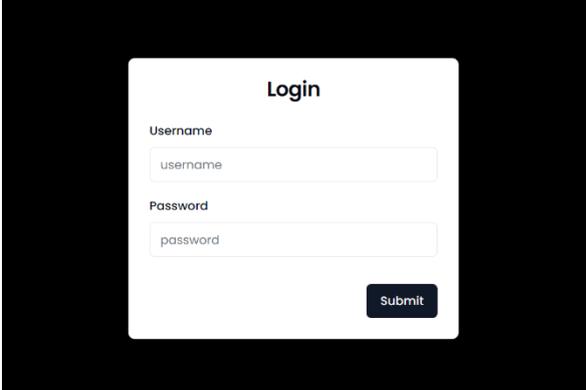
Tabel 4. 14 *Scenario Test*

Aktivitas	Memasukkan inputan salah dan tidak terverifikasi
Expected Result	Menampilkan pemesanan yang berhasil
Result	Menampilkan pemesanan yang berhasil
Nilai	Username: Sariyono Rahman Password: agen1
Keterangan	Valid

2) Pengujian *Black Box*

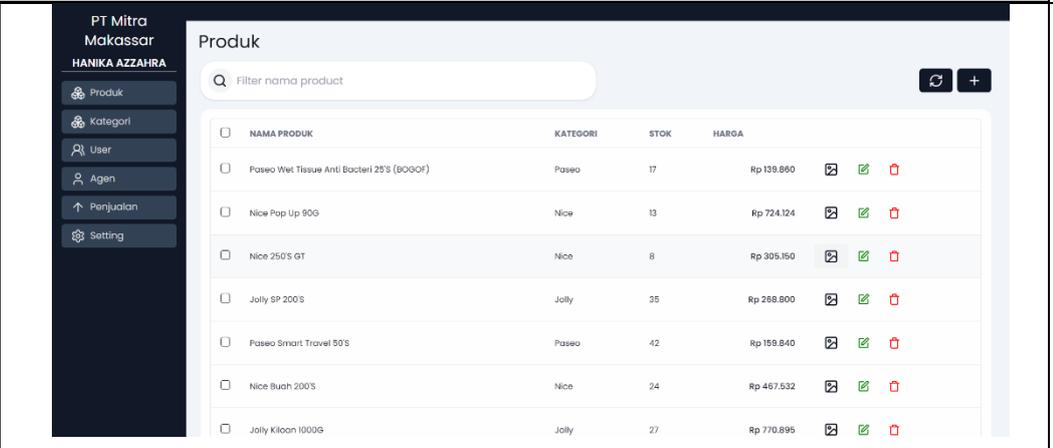
- a. *Black box* melakukan proses *login* pada aplikasi

Tabel 4. 15 *Black box* melakukan proses *login* pada aplikasi

<i>Test Factor</i>	Hasil	Kesimpulan
Jika tombol <i>submit</i> pada <i>login</i> di tekan akan menampilkan halaman <i>Dashboard</i>	✓	Berhasil, karena <i>Dashboard</i> dapat di tampilkan
Screenshot		
		

b. *Black box* melakukan proses pada input produk

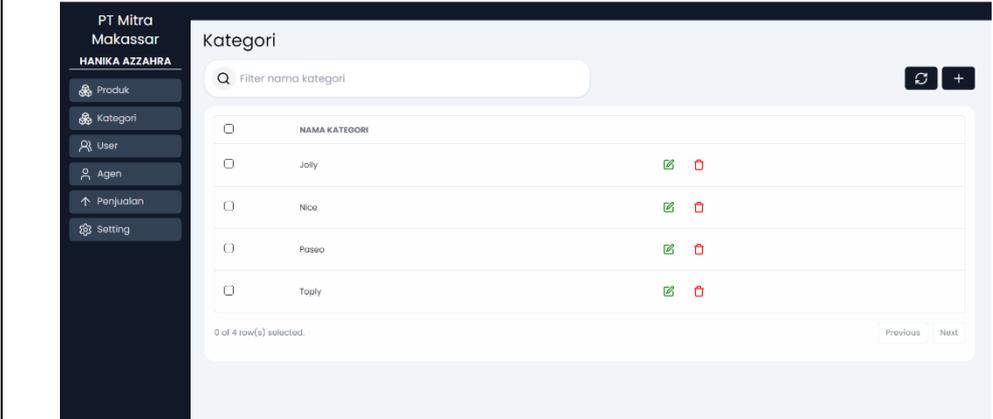
Tabel 4. 16 *Black box* melakukan proses input produk

<i>Test Factor</i>	Hasil	Kesimpulan
Jika tombol input produk di tekan akan menampilkan produk yang akan di pilih untuk ditampilkan dan melakukan proses simpan perangkat.	✓	Berhasil, karena produk dapat di tampilkan data detail pada sistem.
Screenshot		
		

c. *Black box* melakukan proses pada input kategori

Tabel 4. 17 *Black box* melakukan proses input kategori

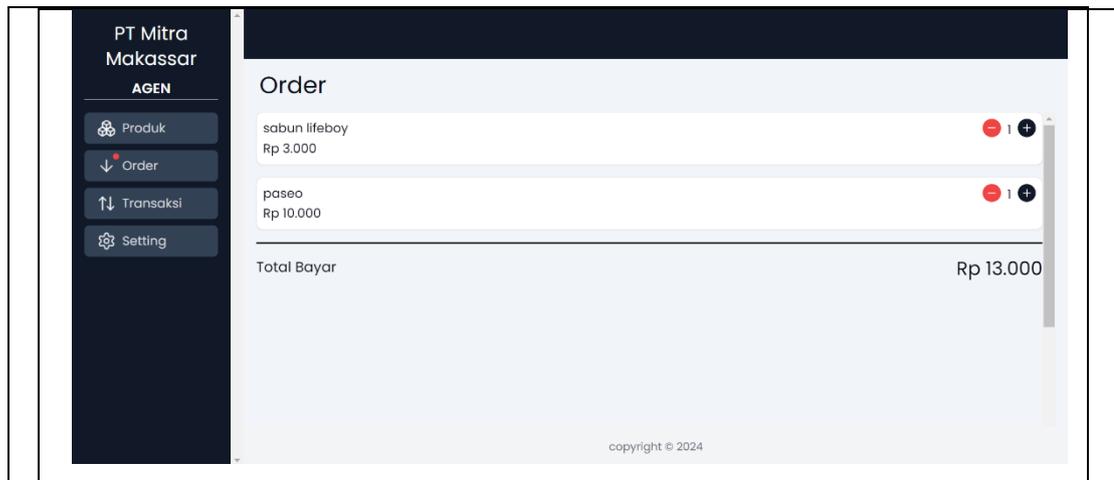
<i>Test Factor</i>	Hasil	Kesimpulan
Jika tombol input kategori di tekan akan menampilkan kategori	✓	Berhasil, karena kategori dapat di tampilkan data detail pada sistem.

yang akan di pilih untuk ditampilkan dan melakukan proses simpan perangkat.		
Screenshot		
		

d. *Black box* melakukan proses pada order produk

Tabel 4. 18 *Black box* melakukan proses *login* pada aplikasi

<i>Test Factor</i>	Hasil	Kesimpulan
Jika produk di tekan akan menampilkan halaman order untuk melakukan proses transaksi	✓	Berhasil, karena produk yang di order dapat di tampilkan data detail pada sistem.
Screenshot		



E. Evaluasi Kinerja Aplikasi

Penilaian kinerja aplikasi pemesanan produk PT. Mitra Makassar menggunakan data yang dikumpulkan melalui Google Form. Formulir tersebut dirancang untuk mengumpulkan data dari agen yang secara aktif menggunakan aplikasi dalam kegiatan mereka. Tujuan pengumpulan data ini adalah untuk memahami seberapa efisiensi aplikasi tersebut dalam memenuhi kebutuhan operasional pengguna.

Tabel 4. 19 Pertanyaan Kuisisioner

No	Pertanyaan	Kode
1	Seberapa mudah Anda menggunakan aplikasi pemesanan produk ini?	P1
2	Bagaimana Anda menilai kecepatan respons aplikasi saat digunakan?	P2
3	Apakah anda mengalami kesulitan saat melakukan <i>login</i> atau akses ke aplikasi?	P3
4	Bagaimana kualitas dan relevansi informasi produk yang di tampilkan dalam aplikasi?	P4

No	Pertanyaan	Kode
5	Apakah instruksi dan pesan kesalahan yang diberikan oleh aplikasi mudah dimengerti?	P5
6	Seberapa besar kemungkinan anda merekomendasikan aplikasi ini kepada orang lain?	P6

Tabel 4. 20 Hasil Efisiensi Aplikasi

No	Nama Responden	P1	P2	P3	P4	P5	P6	total
1	Ainan Mart	5	5	5	4	5	5	96%
2	77 Mart	5	5	5	5	5	4	96%
3	Tk.Abdullah	4	4	5	5	5	5	93%
4	24 Mart	5	5	4	5	3	5	90%
5	Cahaya Matahari	4	5	5	5	5	5	96%
6	Swiss Mart	5	5	5	5	5	5	100%
7	Toko Jingga	5	5	5	5	5	5	96%
8	Kupa One	4	5	5	5	5	5	100%
9	Diana Mart	5	5	5	5	5	5	100%
10	Haura Mart	4	5	5	5	5	5	96%
TOTAL								96,3%

Dengan memberikan bobot di setiap jawaban responden dengan skor 1-5 (skor paling tinggi adalah jawaban paling terbaik), maka didapatkan hasil presentase efisiensi sebesar 96,3% yang didapatkan dari 10 agen dimana persentase tersebut menyatakan bahwa aplikasi yang dibuat memiliki tingkat efisiensi yang tinggi dalam memudahkan para agen untuk melakukan pemesanan barang.

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan oleh penulis maka dapat ditarik kesimpulan yaitu :

1. Penelitian ini telah berhasil menghasilkan dan mengembangkan aplikasi pemesanan berbasis web yang memudahkan pelanggan PT. Mitra Makassar untuk melakukan pemesanan dan pembelian produk.
2. Berdasarkan survei dari 10 agen, didapatkan hasil 96,3% yang menyatakan tingkat efisiensi dengan menggunakan aplikasi pemesanan produk sangat bagus dan memudahkan agen dalam melakukan transaksi pemesanan produk. Hasil ini menunjukkan bahwa aplikasi tersebut mampu meningkatkan efisiensi dan kemudahan dalam proses pemesanan produk bagi konsumen.

B. Saran

Setelah melakukan penelitian, maka penulis memberikan saran sebagai berikut:

1. Penulis berharap agar aplikasi ini dapat diterapkan, dimanfaatkan dan dikembangkan dikemudian hari dan menyarankan agar pengembang aplikasi ini dapat menambahkan beberapa fitur-fitur lainnya sehingga aplikasi ini bisa lebih lengkap.

2. Penulis menyadari bahwa sistem yang dibangun masih membutuhkan penyempurnaan yang lebih baik. Oleh karena itu, penulis menyarankan agar skripsi ini dapat dijadikan sebagai bahan referensi untuk mengembangkan sistem yang lebih sempurna.
3. Pemakai disarankan untuk memperhatikan kekurangan dan kelemahan aplikasi ini, agar dapat segera mencari penyelesaian masalahnya dan segera memperbaikinya.

DAFTAR PUSTAKA

- Andaru, A. (2018). Pengertian *Database* Secara Umum. *OSF Prepr*, 2.
- Aprianto, N. E. K. (2021). Peran Teknologi Informasi dan Komunikasi dalam Bisnis. *International Journal Administration Business & Organization*, 2(1), 8–15.
- Aripin, S., & Somantri, S. (2021). Implementasi Progressive Web Apps (PWA) pada Repository E-Portofolio Mahasiswa. *Jurnal Eksplora Informatika*, 10(2), 148–158. <https://doi.org/10.30864/eksplora.v10i2.486>
- Fristanto, H. T. (2013). Pembuatan *Website* Promosi Dan Pemesanan Produk Pada Home Industri Agro Santoso Jamur Punung Pacitan. *Seruni-Seminar Riset Unggulan Nasional Informatika Dan Komputer*, 2(1).
- Hasyim, Ade Hastuty. 2021. *DASAR PEMROGRAMAN*. CV Bangun Bumitama, Makassar
- Hayami, R., Harianto, R., & Unik, M. (2019). Rancang Bangun Aplikasi Katalog Online Dan Sistem Pemesanan Produk. *Jurnal Fasilkom*, 9(2), 440–447.
- Herawan, R., Alam, S., Zainal, M., & Herawan, R. (2021). *SISTEM INFORMASI PENYEDIA JASA PENJUALAN PAKAIAN BERBASIS ANDROID*. 1(3), 3–6.
- Hidayatulloh, R. (2020). *Pembangunan Aplikasi Mobile Toko Online Dan Booking Service Di Sentra Anugrah Motor Memanfaatkan Api Payment Gateway*. Universitas Komputer Indonesia.
- Jannah, D. N., & Mulyono, H. (2021). Analisis Dan Perancangan Sistem Informasi Pemesanan Produk Pengantin Berbasis Web Pada Wedding Organizer MeyMey Cantik. *Jurnal Manajemen Sistem Informasi*, 6(2), 209–220.
- Pangalila, A. J. F., & Ardhiansyah, M. (2022). Perancangan Sistem Aplikasi Management Booking Service Bengkel Berbasis *Website* Dengan Metode Prototype (Studi Kasus Bengkel Motor Rido Racing Rempoa Tangerang Selatan). *INFORMATIKA*, 3(2).
- Rusli, M. S., Ahmar, A. S., & Rahman, A. (2019). *Pemrograman Website dengan PHP-MySQL untuk Pemula*. Yayasan Ahmar Cendekia Indonesia.
- Simatupang, J., & Gomal Juni Yanris, S. (2020). Implementasi Sistem Informasi Booking Service Online Pada Pt. Riau Argo Perkasa Berbasis Web. *Jurnal Intra Tech*, 4(2), 69–80.
- Sitepu, A. I. B., & Tanjung, D. Y. H. (2020). Rancang Bangun Aplikasi Pemesanan dan Penjualan Berbasis Web dan Android pada Toko YT. Wall Interior. *Jurnal Mahasiswa Fakultas Teknik Dan Ilmu Komputer*, 1(1), 816–828.
- Supriyatna, A. (2017). Sistem Informasi Pemesanan Jasa Percetakan Berbasis Web. *Swabumi (Suara Wawasan Sukabumi): Ilmu Komputer, Manajemen, Dan Sosial*, 5(1), 65–73.
- Suri, G. P., & Hasibuan, Z. S. (2021). Sistem Informasi Penjualan Online Berbasis Web Di Tassia Store. *Engineering and Technology International Journal*, 3(01), 55–65.

- Triyono, T., Safitri, R., & Gunawan, T. (2018). Perancangan Sistem Informasi Absensi Guru Dan Staff Pada SMK Pancakarya Tangerang Berbasis Web. *Journal Sensi*, 4(2), 153–167.
- Yudhanto, Y., & Prasetyo, H. A. (2019). *Mudah Menguasai Framework Laravel*. Elex Media Komputindo.
- Zainal, M. (2022). *SISTEM PENGAMBILAN NOMOR ANTRIAN DAN PENGINGAT JADWAL KUNJUNGAN BERBASIS WHATSAPP GATEWAY (STUDY KASUS KLINIK SITTI KHADIJAH)*. 1(1), 1–7.