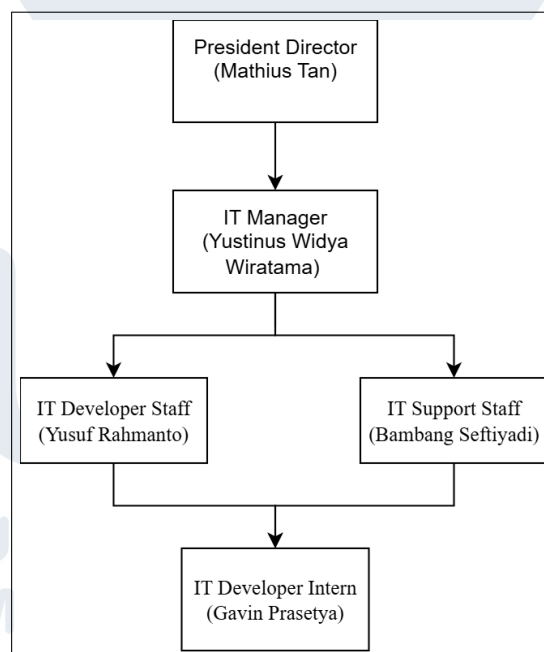


BAB 3

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Selama pelaksanaan kerja magang, posisi yang ditempati adalah *IT Developer Intern* pada divisi *IT*. Divisi *IT* berada di bawah koordinasi *IT Manager*, yaitu Yustinus Widya Wiratama. Bapak Yusuf Rahmanto sebagai *Supervisor IT* bertindak sebagai pembimbing yang memberikan arahan dan menugaskan kegiatan terkait proyek pengembangan sistem, serta melakukan pemantauan proses secara berkala. Dalam pelaksanaan-nya, *Intern* berkoordinasi dengan saudara Yustinus Widya Wiratama *IT Manager* yang memberikan masukan dan arahan terkait antarmuka pengguna, serta Saudara Bambang Seftiyadi *IT Staff* yang turut memberikan dukungan dalam berbagai kegiatan divisi.



Gambar 3.1. Kedudukan *IT Development Intern* dalam Struktur Divisi *IT*

Mekanisme koordinasi dilakukan melalui pertemuan harian untuk meninjau perkembangan proyek dan menyelaraskan hasil kerja dengan kebutuhan perusahaan. Selain fokus pada pengembangan website terdapat kerjaan lainnya yaitu membantu merapihkan dan menginputkan data serial number (SN) ke dalam mySQL (Database), turut mendukung kegiatan divisi *IT Support Staff* seperti

membantu packing PC dan juga backup file. Struktur koordinasi ini memastikan bahwa seluruh kegiatan magang berjalan terarah dan selaras dengan tujuan divisi maupun perusahaan secara keseluruhan.

3.2 Tugas yang Dilakukan

Selama pelaksanaan kegiatan magang di PT. Caturaga Tiara Persada pada divisi *IT* dalam pengembangan website *Sales Force Application*. tugas-tugas yang dilakukan meliputi :

1. Analisis kebutuhan pengguna dan sistem untuk pengembangan *website* berbasis *PHP*.
2. Perancangan desain antarmuka pengguna *UI* untuk proyek *Sales Force Application* menggunakan *Figma*, dengan memperlihatkan aspek tampilan, tata letak, serta kemudahan penggunaan agar sesuai dengan kebutuhan pengguna yaitu sales.
3. consume API dari *Epicollect5* untuk mengambil data dan token yang ditampilkan dalam bentuk table menggunakan localhost *Laravel*.
4. Implementasi fitur *Sales Force Application* menggunakan *framework Laravel* untuk *website*.
5. Penginputan data serial number dari file *Microsoft Excel* ke dalam database *SN* menggunakan *PHP*, guna memastikan data tersimpan dengan rapi, terstruktur, dan dapat diakses melalui sistem yang dikembangkan
6. Perbaikan *bug* dan revisi fitur berdasarkan *feedback* dari *supervisor*.
7. Pengujian website untuk memastikan fungsionalitas, dan performa *platform* berjalan dengan baik.
8. Pelaporan progres harian dan mingguan kepada *supervisor* mengenai perkembangan proyek pengembangan aplikasi.

3.3 Uraian Pelaksanaan Magang

Pelaksanaan magang dilaksanakan dari hari senin hingga jumat pada pukul 08:00 hingga 17:00, dengan total 40 jam kerja per minggu. Rincian aktivitas dan jadwal pelaksanaan magang di jabarkan pada tabel 3.3.

Pelaksanaan kerja magang diuraikan seperti pada Tabel 3.3.

Tabel 3.1. Aktivitas magang yang dilaksanakan per minggu

Minggu	Tanggal	Pekerjaan yang dilakukan
1	2-5 Sept 2025	Pembelajaran dan penerapan API menggunakan Postman dan Laravel, pengambilan data dari API <i>Epicollect5</i> , pembuatan fungsi untuk menampilkan data absensi dan lokasi pada website, membuat fitur untuk menampilkan dan menyimpan gambar melalui API ke sistem lokal laravel, dan instalasi software yang digunakan untuk perusahaan.
2	8-12 Sept 2025	Pembelajaran dan penerapan API menggunakan Postman dan Laravel, pengambilan data dari API <i>Epicollect5</i> , pembuatan fungsi untuk menampilkan data absensi dan lokasi pada website, membuat fitur untuk menampilkan dan menyimpan gambar melalui API ke sistem lokal laravel, dan instalasi software yang digunakan untuk perusahaan.
3	15-19 Sept 2025	Penyusunan presentasi, Mempelajari API Google Maps, perancangan dan revisi desain proyek Sales Force Application <i>SFA</i> menggunakan <i>Figma</i> .
4	22-26 Sept 2025	Pengembangan desain antarmuka <i>UI</i> untuk proyek Sales Force Application di fitur Check-In dan Check-Out, revisi desain fitur Check-In, Customer Profile, penyempurnaan elemen UI seperti tombol, font, dan tampilan peta agar lebih menarik dan fungsional.
5	29 Sept - 3 Oct 2025	pengembangan proyek sales force Application <i>SFA</i> di modul Check-In dan Check-Out, menyempurnakan desain antarmuka <i>UI</i> agar lebih menarik, membuat fitur lokasi dan kamera, memperbaiki kode pada sistem, dan melakukan perevisian tabel dalam database <i>mySQL</i> .

lanjutan di halaman berikutnya

Minggu	Tanggal	Pekerjaan yang dilakukan
6	6 - 10 okt 2025	penyempurnaan desain antarmuka <i>UI</i> Sales Force Application , revisi desain antarmuka <i>UI</i> ,penginputan dan merapihkan data SN dari file excel ke dalam database <i>mySQL</i> menggunakan <i>PHP</i> .
7	13 okt - 17 okt 2025	Penginputan dan perapihan data SN dari file excel ke dalam database <i>mySQL</i> menggunakan <i>PHP</i> , pengembangan desain antarmuka <i>UI</i> website Sales Force Application pada fitur Check-In dan Customer Visit untuk meningkatkan dan fungsionalitas sistem.
8	20 - 24 okt 2025	Penginputan data SN produk dari file excel ke dalam database <i>mySQL</i> menggunakan <i>PHP</i> , melakukan sorting dan listing ke data-data yang tidak lengkap.
9	27 - 31 okt 2025	pengembangan proyek Sales Force Application (SFA) dengan membuat dan penyempurnaan desain antarmuka (<i>UI</i>) menggunakan <i>Figma</i> serta implementasi sistem menggunakan <i>PHP</i> , membuat fitur mobile Check-In, location,camera, dan memperbaiki kode untuk meningkatkan tampilan dan fungsionalitas website/aplikasi.
10	3 - 7 Nov 2025	Pengerjaan project Sales Force application difokuskan pada pengembangan fitur mobile check-in serta dashboard untuk pencatatan toko yang dikunjungi oleh sales. Selain itu, dilakukan perancangan desain bagian <i>invoice monitoring</i> dengan mencari referensi yang relevan, lalu diimplementasikan ke dalam project. Fitur invoice monitoring juga mulai dikembangkan menggunakan <i>PHP</i> dan ditampilkan melalui website agar dapat digunakan secara langsung.

lanjutan di halaman berikutnya

Minggu	Tanggal	Pekerjaan yang dilakukan
11	10 - 14 Nov 2025	Pekerjaan yang dilakukan mencakup penginputan serial number terbaru pada Oktober 2025 sekaligus mempelajari konsep serta implementasi <i>PWA (Progressive Web Apps)</i> . Fitur PWA dikembangkan agar aplikasi dapat diunduh langsung dari website, kemudian dilanjutkan dengan proses deployment ke browser yang membutuhkan HTTPS serta file pendukung seperti <i>manifest.json</i> , <i>serviceworker.json</i> , dan <i>pwa-install.js</i> .
12	17 - 21 Nov 2025	Pengembangan <i>PWA</i> agar website dapat ditampilkan secara optimal di perangkat mobile. Selain itu, fokus diarahkan pada fitur invoice monitoring dalam <i>project SFA</i> , termasuk menghubungkan database yang telah dipersiapkan agar data dapat muncul langsung di website. Pekerjaan ini bertujuan untuk menyempurnakan fungsi invoice monitoring sehingga dapat digunakan secara efektif dalam aplikasi.
13	24 - 28 Nov 2025	Pengerjaan project magang difokuskan pada pengembangan fitur <i>Invoice Monitoring</i> , dimulai dengan melakukan filter data agar invoice berstatus Closed tidak ditampilkan di web. Selain itu, dilakukan penyesuaian agar tampilan menjadi mobile responsive, sekaligus mempelajari dan mengimplementasikan konsep SSO menggunakan Laravel Passport. Pekerjaan juga mencakup pembelajaran cara consume API ke dalam project serta revisi desain tampilan web dan mobile agar lebih optimal.

lanjutan di halaman berikutnya

Minggu	Tanggal	Pekerjaan yang dilakukan
14	1 - 5 Des 2025	Melakukan revisi menyeluruh pada proyek Invoice Monitoring, khususnya pada bagian dashboard agar tampil lebih eye-catching dan informatif. Selain itu, mengerjakan modul Product dengan melakukan mapping terhadap database yang telah disediakan. Aset berupa foto-foto produk Reiwa dan Arisa juga telah dimasukkan dan ditampilkan dalam bentuk katalog pada halaman web. Page Invoice Monitoring turut direvisi sesuai arahan dari Manager, termasuk perubahan struktur isi invoice agar lebih jelas dan terstruktur. Desain keseluruhan proyek juga diperbagus untuk meningkatkan daya tarik visual dan kenyamanan pengguna.
15	8 - 12 Des 2025	Melakukan pengembangan pada proyek Invoice Monitoring dengan fokus pada beberapa bagian penting. Pertama, mengimplementasikan fitur login menggunakan SSO (Single-Sign-On) yang diuji melalui Postman untuk memastikan integrasi berjalan dengan baik. Selanjutnya, menjalankan proyek Visit-Sales yang diberikan oleh supervisor sebagai referensi untuk memahami alur proses yang akan diterapkan. Pada sisi database, menambahkan tabel yang diperlukan yaitu ocrd-plan dan cities agar mendukung kebutuhan data aplikasi. Selain itu, dikembangkan fitur baru berupa Sales Visit yang berfungsi untuk menampilkan riwayat kunjungan sales ke berbagai lokasi, sehingga sistem menjadi lebih komprehensif dan informatif.

lanjutan di halaman berikutnya

Minggu	Tanggal	Pekerjaan yang dilakukan
16	15 - 19 Des 2025	Melanjutkan pengembangan fitur Sales dengan menambahkan tombol Add Sales Plan serta merapikan tabel Sales Visit agar lebih terstruktur. Pada bagian Home Page, ditambahkan fitur zoom produk sehingga pengguna dapat melihat detail produk dengan lebih jelas. Selain itu, dilakukan perapihan kode sekaligus pengecekan tampilan untuk memastikan konsistensi dan kualitas UI. Pada modul Sales Visit, ditambahkan fitur search untuk mencari kunjungan ke toko tertentu, sorting tabel, serta penerapan DataTable agar pengelolaan data lebih interaktif dan mudah digunakan.
17	22 - 23 Des 2025	Melanjutkan pengembangan fitur Sales dengan menambahkan tombol Add Sales Plan serta merapikan tabel Sales Visit agar lebih terstruktur. Pada bagian Home Page, ditambahkan fitur zoom produk sehingga pengguna dapat melihat detail produk dengan lebih jelas. Selain itu, dilakukan perapihan kode sekaligus pengecekan tampilan untuk memastikan konsistensi dan kualitas UI. Pada modul Sales Visit, ditambahkan fitur search untuk mencari kunjungan ke toko tertentu, sorting tabel, serta penerapan DataTable agar pengelolaan data lebih interaktif dan mudah digunakan.

3.4 Perangkat Lunak dan Perangkat Keras yang Digunakan

Selama pelaksanaan magang, digunakan berbagai perangkat lunak dan perangkat keras untuk mendukung proses perancangan, pengembangan, integrasi, serta pengujian website berbasis *PHP* dengan *framework Laravel* dan pengelolaan database *mysql*. Perangkat-perangkat tersebut mencakup tools pengembangan aplikasi, lingkungan pengujian, serta perangkat pendukung lainnya yang digunakan secara langsung selama kegiatan magang. Penjelasan mengenai perangkat lunak dan perangkat keras yang digunakan disajikan sebagai berikut :

3.4.1 Perangkat Lunak

Tabel 3.2. Daftar Perangkat Lunak yang Digunakan

No.	Perangkat Lunak	Fungsi
1	Visual Studio Code	Digunakan sebagai <i>Integrated Development Environment (IDE)</i> utama dalam proses pengembangan website untuk proyek Sales Force dengan dukungan framework <i>Laravel</i> , dan <i>PHP</i> .
2	HeidiSQL (Database)	Digunakan untuk mengelola dan mengedit data di server database seperti <i>mySQL</i> dan juga dihubungkan ke database server untuk melihat dan mengelola database, tabel, dan data di dalamnya.
3	Postman	Digunakan untuk menguji dan mengelola API pada website seperti mengirim permintaan <i>request</i> dan melihat respon <i>response</i> dari server.
4	Microsoft Excel	Digunakan untuk menaruh data serial number dan produk dari perusahaan dan di kelola menggunakan <i>Visual Studio Code</i> .
5	Epicollect5	Digunakan untuk mengumpulkan dan mengelola data perusahaan melalui form digital di perangkat mobile atau web.
6	WAMPP Server	Digunakan sebagai server lokal untuk menjalankan aplikasi web yang menggunakan <i>PHP</i> , <i>mySQL</i> di laptop.

3.4.2 Perangkat Keras

Tabel 3.3. Daftar Perangkat Keras yang Digunakan

No.	Perangkat Keras	Fungsi
1	Laptop (Lenovo LOQ 15ARP9)	Digunakan sebagai perangkat utama untuk pengembangan website <i>Sales Force</i> , penggunaan database, dan juga desain UI menggunakan <i>Figma</i> dengan spesifikasi Processor AMD Ryzen 7 7435HS @ 3.1 GHz, RAM 24 GB DDR5, dan operating system Windows 11 Home Single Language 64-Bit.

3.5 Proses Pelaksanaan Kerja Magang

Proses pelaksanaan kerja magang di PT Caturaga Tiara Persada (CTP) dalam mengerjakan proyek ini menggunakan pendekatan *Software Development Life Cycle (SDLC)* sebagai kerangka kerja agar kegiatan dapat berjalan secara terstruktur, terarah, dan efektif. Dalam pendekatan *SDLC* tersebut, metodologi yang diterapkan adalah *Agile*. Keunggulan utama *Agile* terletak pada kemampuannya untuk diterapkan secara berkala dan iteratif, memungkinkan proses pengembangan yang fleksibel, responsif terhadap perubahan, serta didukung umpan balik dari berbagai pemangku kepentingan [4]. Pendekatan *Agile* bersifat adaptif sehingga dapat diterapkan baik dalam perancangan dan pengembangan website berbasis *PHP Laravel* serta memungkinkan penyesuaian selama proses berlangsung tanpa mengganggu keselarasan dan efisiensi pengembangan.

3.5.1 Perencanaan Proyek: Mengidentifikasi Kebutuhan Bisnis dan Mengidentifikasi Fitur Website

Proses pengembangan aplikasi web ini diawali dengan tahap perencanaan yang menekankan identifikasi kebutuhan bisnis dan teknis secara menyeluruh. Dalam diskusi bersama supervisor dijelaskan bahwa aplikasi dirancang untuk mendukung pengelolaan data invoice perusahaan secara terpusat dengan memanfaatkan basis data internal yang dikelola melalui HeidiSQL. Integrasi ke database ini dipandang penting karena memungkinkan pencatatan transaksi yang lebih rapi, konsisten, dan mudah diaudit, sehingga setiap informasi yang ditampilkan kepada pengguna selalu akurat dan terkini. Selain itu, aspek keamanan serta kenyamanan akses pengguna juga menjadi perhatian utama. Oleh karena itu,

diputuskan untuk menerapkan sistem autentikasi Single Sign-On (SSO) berbasis OAuth2, yang memungkinkan pengguna masuk ke aplikasi menggunakan akun perusahaan yang sudah ada tanpa perlu membuat akun baru.

Pendekatan ini tidak hanya meningkatkan efisiensi, tetapi juga memperkuat kontrol akses dan mengurangi risiko duplikasi data pengguna. Dengan adanya kombinasi integrasi database dan SSO, aplikasi diharapkan mampu memberikan pengalaman penggunaan yang lebih sederhana, intuitif, dan aman bagi tim lapangan maupun manajemen. Tahap perencanaan ini kemudian menjadi fondasi bagi proses desain antarmuka, implementasi kode, serta pengujian sistem yang dilakukan secara bertahap sesuai kerangka kerja Software Development Life Cycle (SDLC).

3.5.2 Perancangan Sistem : Diagram UML dan Mockup UI

Tahap perancangan dalam pengembangan aplikasi web dilakukan untuk memvisualisasikan bagaimana alur kerja sistem pada setiap modul dan fitur, sehingga aplikasi dapat berfungsi dengan baik dan efisien. Penggunaan diagram UML menjadi salah satu keunggulan karena mampu mendokumentasikan arsitektur serta proses sistem secara terstruktur dan memiliki standar yang jelas. Hal ini memudahkan tim dalam berkomunikasi maupun melaksanakan proses pengembangan [5].

Pada tahap ini, dibuat use case diagram untuk menunjukkan bentuk interaksi antara pengguna dan sistem, serta activity diagram yang menggambarkan alur proses dari masing-masing use case. Selain itu, disusun pula mockup antarmuka UI untuk menampilkan rancangan tampilan serta cara kerja aplikasi pada setiap modul dan fitur yang dikembangkan.

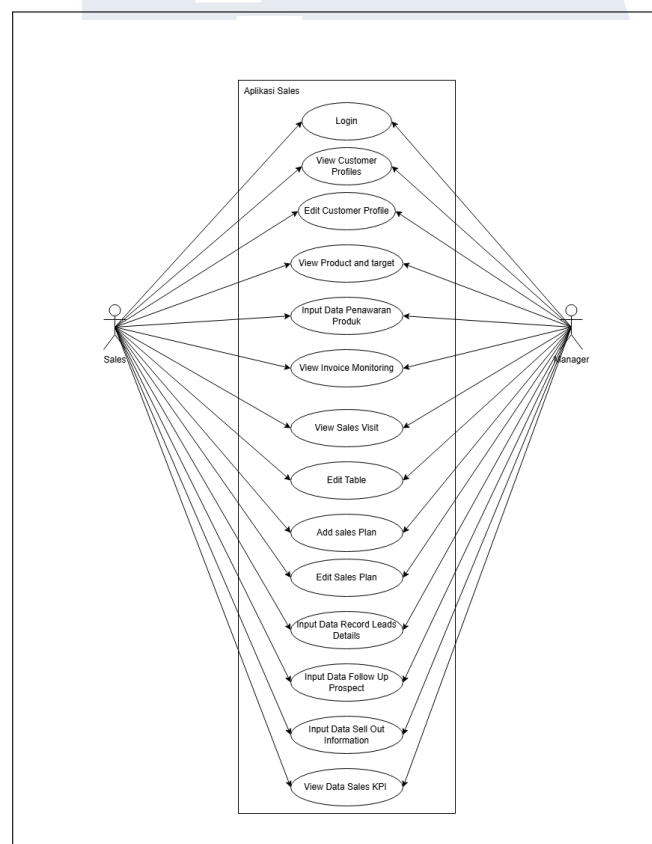
3.5.3 Use case diagram Web Application

Hubungan antara pengguna dan web aplikasi yang dibangun digambarkan dalam use case diagram di Gambar 3.2. Diagram ini menunjukkan berbagai cara pengguna berinteraksi dengan sistem. Ini termasuk cara mereka mengakses fitur, melakukan fungsi tertentu, dan mengetahui batasan kewenangan mereka. Diagram ini menunjukkan alur kerja aplikasi, yang memungkinkan proses pengembangan untuk memastikan bahwa setiap fungsi sesuai dengan kebutuhan pengguna.

Selain itu, use case ini menunjukkan siapa yang memiliki hak untuk melihat atau mengontrol data tertentu, sehingga pengaturan akses dalam aplikasi dapat

dirancang dengan lebih terorganisir dan aman. Dengan pemetaan ini, pengembang dapat memperoleh pemahaman menyeluruh tentang proses yang terjadi dalam aplikasi sebelum memulai tahap implementasi.

Dalam diagram tersebut terlihat bahwa *User Sales* dan *Manager* dapat mengakses semua use case, seperti Login, View Customer profiles, Edit Customer, View Product and target, Input Data penawaran Produk, View Invoice Monitoring, View Sales Visit, Edit Sales Plan, Add Sales Plan, Input Data Record Leads Details, Input Data Follow Up Prospect, Input Data Sell Out Information, dan View Data Sales KPI.



Gambar 3.2. Use Case Diagram Web Application

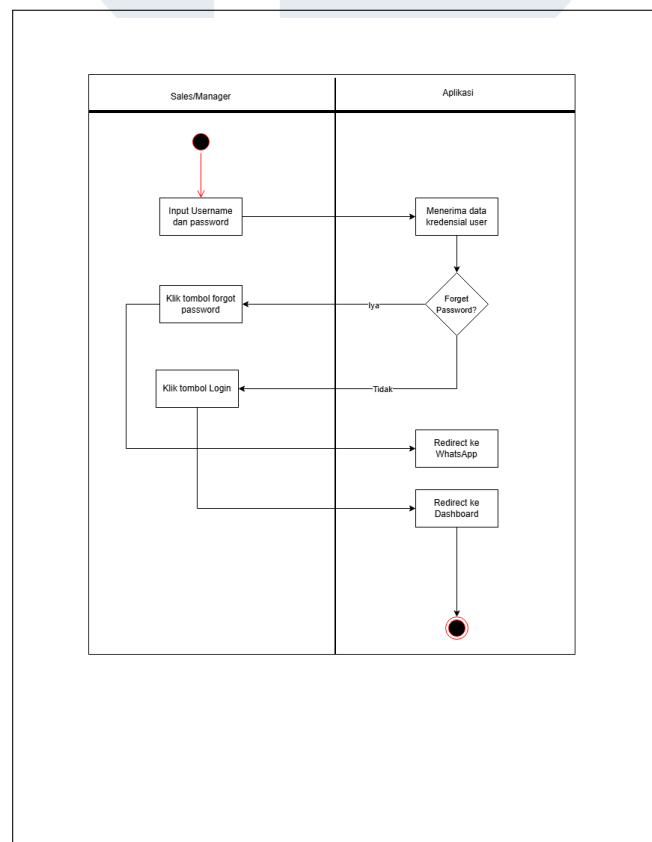
A Activity Diagram Login

Alur interaksi pengguna dengan aplikasi ini digambarkan melalui sebuah *Activity Diagram* yang ditampilkan pada Gambar 3.3. Diagram tersebut menjelaskan bagaimana proses dimulai dari tahap *Start*, yaitu ketika pengguna diminta untuk memasukkan informasi dasar berupa nama pengguna (username) dan kata sandi (password). Tahapan ini menjadi pintu masuk utama agar sistem

dapat mengenali identitas pengguna sebelum mereka melanjutkan ke fitur-fitur berikutnya.

Apabila pengguna mengalami kendala, misalnya lupa nama pengguna atau kata sandi, aplikasi telah menyediakan *tombol Forget Password*. Tombol ini berfungsi sebagai jalur bantuan cepat yang akan langsung mengarahkan pengguna untuk menghubungi admin melalui *WhatsApp*. Dengan adanya mekanisme ini, proses pemulihan akun menjadi lebih sederhana dan tidak memerlukan prosedur yang rumit.

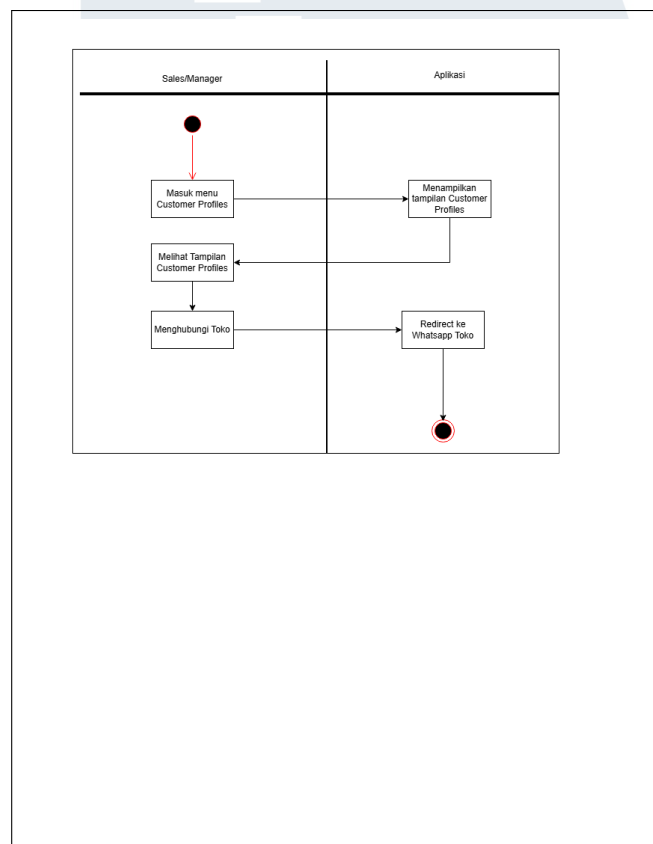
Selain itu, karena tim pengembang sudah menyiapkan data berupa *username* dan *password* untuk setiap pengguna, khususnya bagi tim Sales, mereka tidak perlu lagi membuat akun secara mandiri. Hal ini bertujuan untuk mempermudah proses login, mengurangi potensi kesalahan input, serta memastikan bahwa setiap anggota tim dapat langsung menggunakan aplikasi sesuai dengan peran dan kebutuhan mereka. Dengan cara ini, interaksi pengguna menjadi lebih efisien, terarah, dan mendukung kelancaran operasional di lapangan.



Gambar 3.3. Activity Diagram Login Aplikasi

B Activity Diagram Masuk Modul (Customer Profiles)

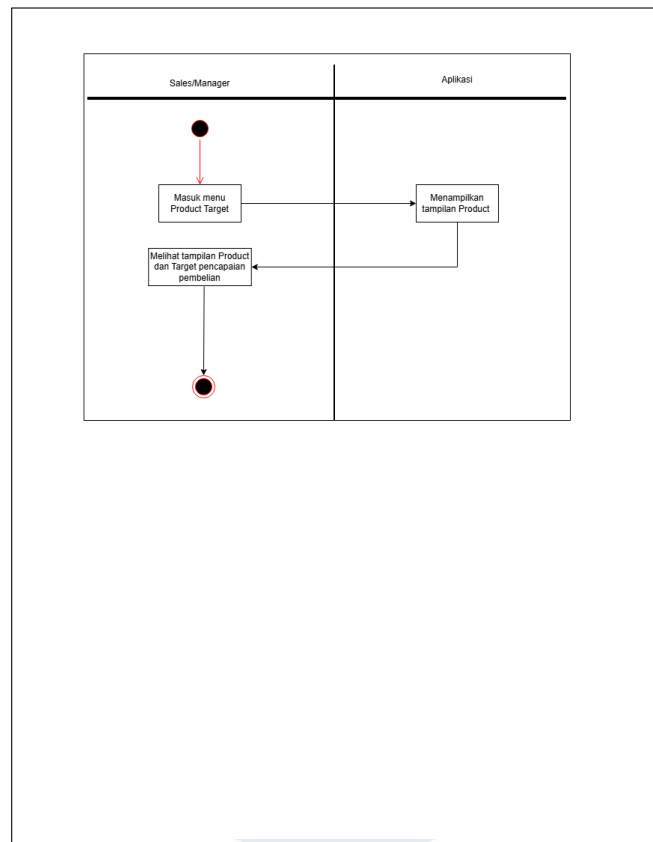
Alur interaksi pengguna dengan aplikasi ini digambarkan melalui sebuah *Activity Diagram* yang ditampilkan pada Gambar 3.4. Diagram tersebut menjelaskan bagaimana proses dimulai dari tahap *Start*, yaitu sesampainya di tampilan *Dashboard* terdapat Beberapa Modul di dalamnya. Salah satunya adalah *Customer Profiles*. Dalam Modul *Customer Profiles*, Sales dapat *View Customer* siapa saja yang ia kunjungi, dan juga terdapat sebuah tombol yaitu *Hubungi Toko* yang dimaksud dapat menghubungi toko customer dan redirect ke *Whatsapp*.



Gambar 3.4. Activity Diagram Aplikasi Modul(Customer Profiles)

C Activity Diagram Modul(Product Targets)

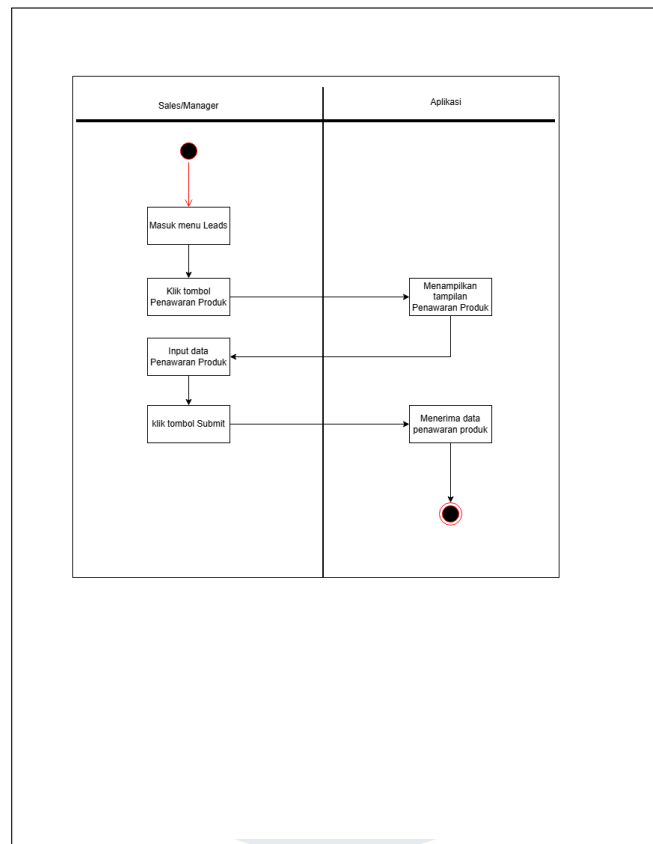
Pada gambar 3.5 menampilkan *Activity Diagram* yang menggambarkan proses ketika sudah masuk dalam tampilan *Dashboard* terdapat modul *ProductTarget* yang berfungsi sebagai penampilan produk apa saja yang terdapat dalam perusahaan ini dan juga target dalam pencapaian pembelian produk tersebut sudah mencapai berapa persen.



Gambar 3.5. Activity Diagram Modul(Product Targets)

D Activity Diagram Modul(Leads)

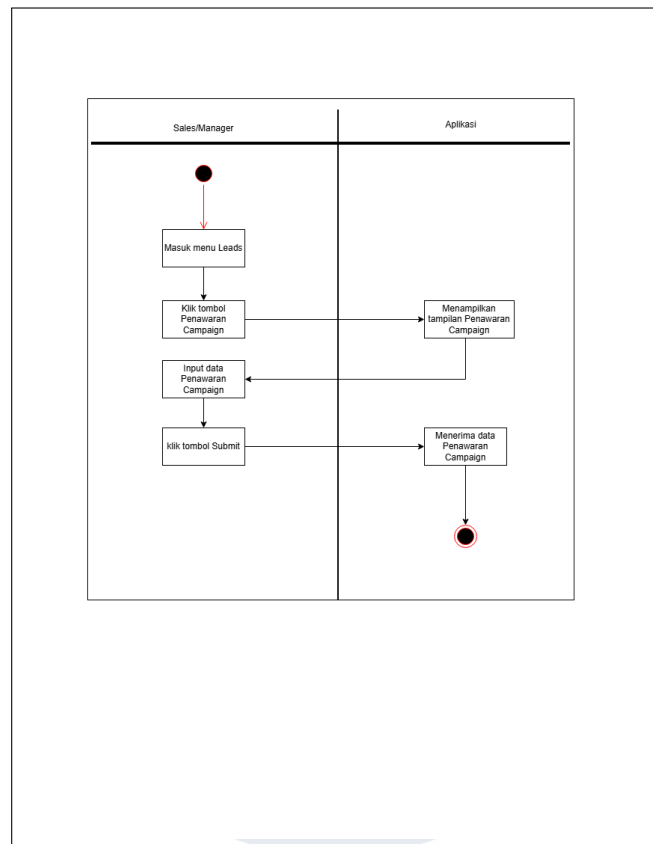
Pada Gambar 3.6 menampilkan *Activity Diagram* yang menggambarkan proses ketika sudah dalam tampilan dashboard terdapat banyak modul di tampilannya, salah satunya adalah modul *Leads*. Ketika pengguna menekan tombol *Penawaran Produk*, setelah itu Pengguna mengisi data penawaran sesuai kebutuhan dan menekan tombol submit untuk mengirimkan data tersebut ke dalam sistem.



Gambar 3.6. Activity Diagram Modul(Leads)

E Activity Diagram Penawaran Campaign Modul(Leads)

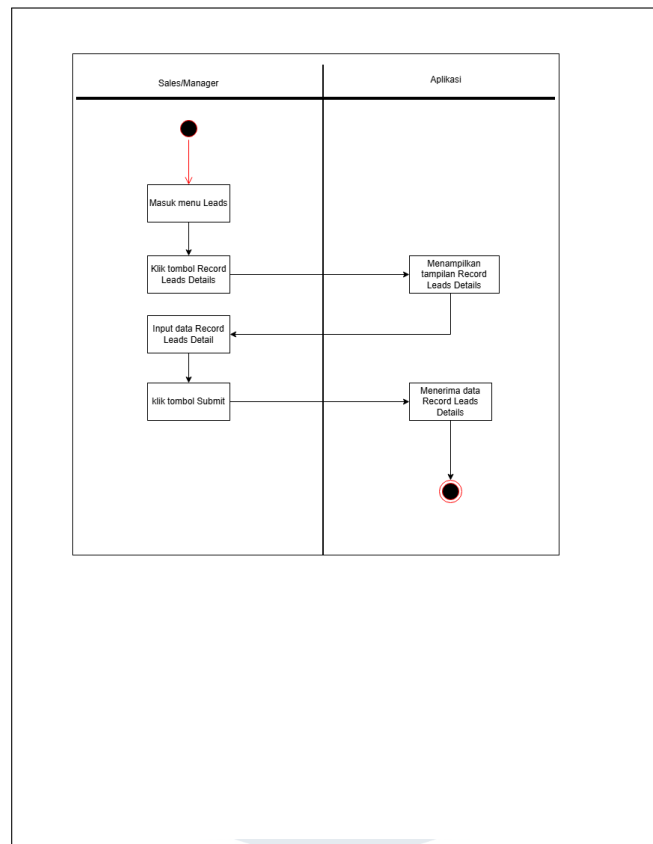
Pada Gambar 3.7 menampilkan *Activity Diagram* yang menggambarkan proses yang dimana dimulai dari *pengguna* masuk ke modul *Leads*, kemudian menekan tombol *Penawaran Campaign*, lalu aplikasi merespon dengan menampilkan khusus untuk pegisian form *data campaign*. *Pengguna* mengisi data penawaran sesuai kebutuhan dan menekan tombol submit untuk mengirimkan data penawaran tersebut.



Gambar 3.7. Activity Diagram Modul(Penawaran Campaign)

F Activity Diagram Record Leads Detail Modul(Leads)

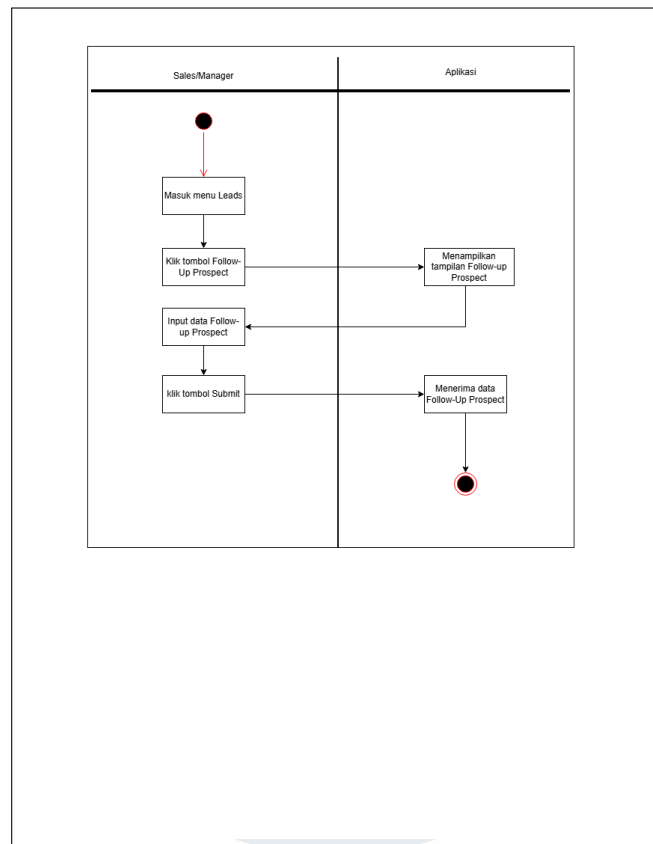
Pada Gambar 3.8 menampilkan *Activity Diagram* yang menggambarkan proses pencatatan *Record Leads Details*. Pengguna masuk ke modul *Leads*, lalu menekan tombol *Record Leads Details*. Aplikasi kemudian menampilkan tampilan untuk pengisian data. Setelah pengguna mengisi data *detail Leads*, lalu menekan tombol submit untuk mengirimkan data tersebut ke dalam sistem.



Gambar 3.8. Activity Diagram Modul(Record Leads Details)

G Activity Diagram Follow-Up Prospect Modul(leads)

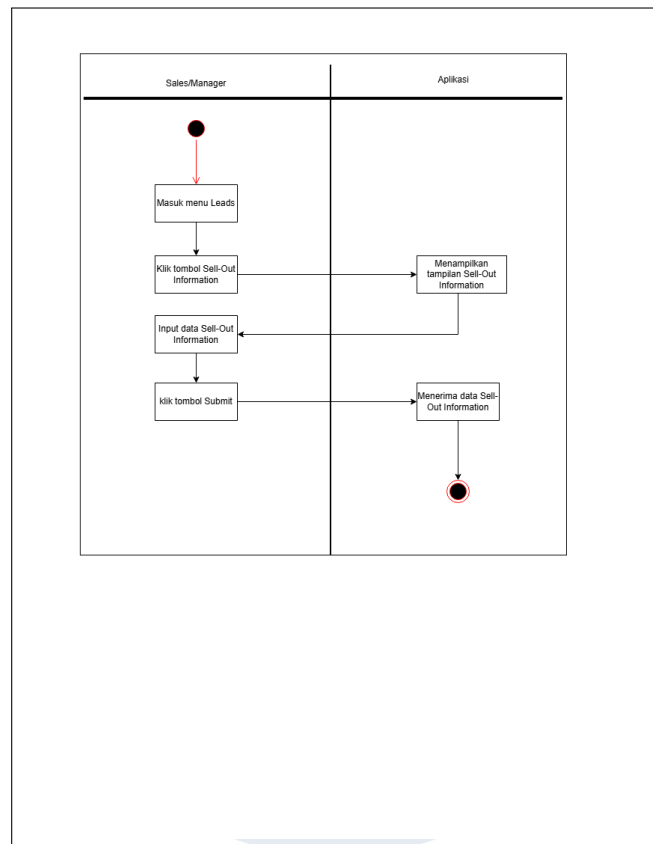
Pada Gambar 3.9 menampilkan *Activity Diagram* yang menggambarkan proses pengelolaan Follow-Up Prospect. Proses dimulai ketika pengguna masuk ke modul *Leads*, lalu menekan tombol Follow-up Prospect. Aplikasi kemudian menampilkan tampilan untuk pengisian data follow-up. Setelah tampilan muncul, pengguna mengisi data prospect follow-up sesuai kebutuhan dan menekan tombol Submit untuk mengirimkan data. Data yang dimasukkan diterima oleh sistem aplikasi dan disimpan untuk diproses



Gambar 3.9. Activity Diagram Modul(Follow-Up)

H Activity Diagram Sell-Out Information Modul(leads)

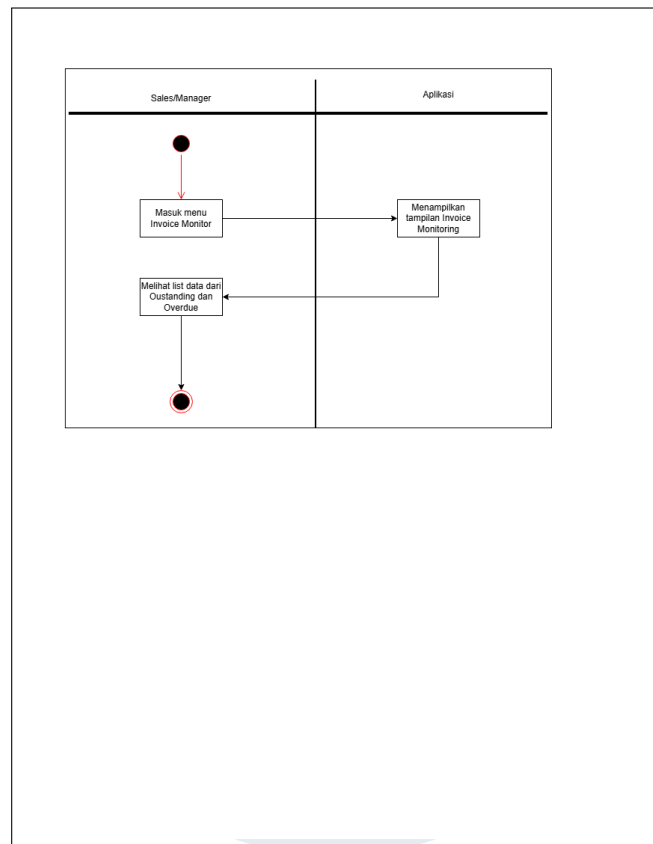
Pada Gambar 3.10 menunjukkan *Activity Diagram* antara pengguna dan sistem aplikasi. Proses dimulai dengan pengguna masuk ke menu Leads dan menekan tombol Sell-Out Information. Aplikasi menanggapi dengan menampilkan tampilan khusus untuk pengisian data tersebut. Setelah tampilan muncul, pengguna mengisi data Informasi Penjualan sesuai kebutuhan, lalu menekan tombol Kirim untuk mengirimkan data ke sistem.



Gambar 3.10. Activity Diagram Modul (Sell-Out Information)

I Activity Diagram Modul (Invoice Monitoring)

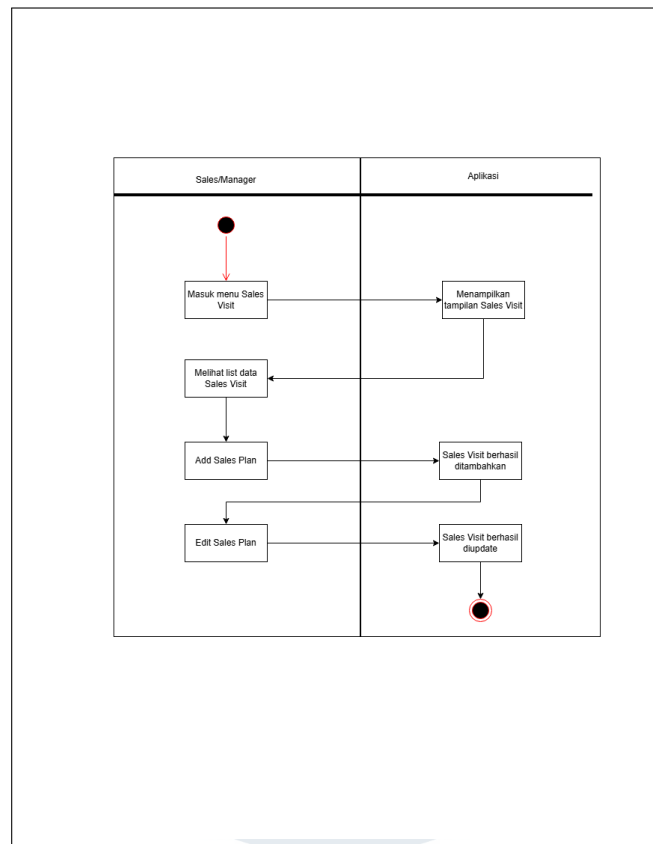
Pada Gambar 3.11 menunjukkan *Activity Diagram* yang menggambarkan interaksi antara pengguna dan sistem aplikasi proses Invoice Monitoring. Proses dimulai dari pengguna masuk ke modul *Invoice Monitor*, kemudian melihat daftar data invoice yang berstatus Outstanding dan Overdue. Sebagai respons, aplikasi menampilkan tampilan untuk monitoring invoice. Proses berjalan secara pasif, di mana pengguna hanya melakukan viewing terhadap data yang telah tersedia tanpa perlu melakukan input tambahan.



Gambar 3.11. Activity Diagram Modul (Invoice Monitoring)

J Activity Diagram Modul (Sales Visit)

Pada Gambar 3.12 menunjukan *Activity Diagram* yang menggambarkan interaksi antara pengguna dan sistem aplikasi proses Sales Visit. Proses dimulai dari pengguna masuk ke modul *Sales Visit*, kemudian melihat tabel data nama sales yang berhasil atau sudah mengunjungi ke toko. Pengguna dapat mengedit tabel data untuk mengecek apakah data tersebut sudah benar atau tidak dan jika sudah benar dapat di save lalu muncul notifikasi "Sales plan berhasil diupdate". lalu pengguna dapat mengklik button "Add Sales Plan" untuk menambahkan data secara manual.

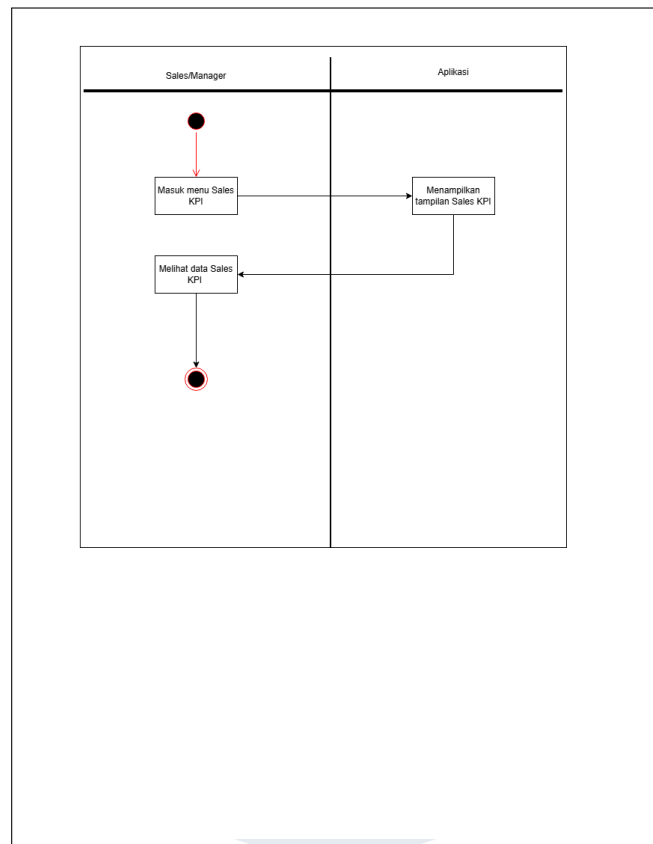


Gambar 3.12. Activity Diagram Modul(Sales Visit)

K Activity Diagram Modul (Sales KPI)

Dalam proses akses dan pemantauan data KPI penjualan, pengguna dan sistem aplikasi berinteraksi seperti yang ditunjukkan dalam Activity Diagram Gambar 3.12. Proses dimulai ketika pengguna masuk ke menu KPI penjualan, dan sistem aplikasi menanggapi dengan menampilkan tampilan khusus yang berisi informasi tentang indikator kinerja penjualan.

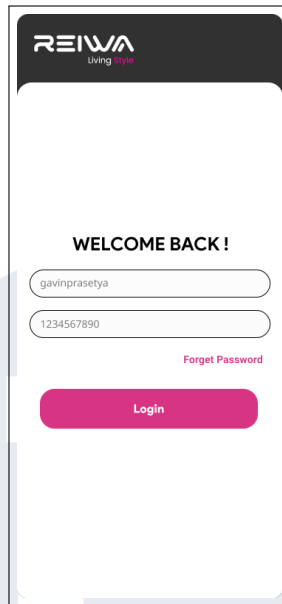
Setelah tampilan muncul, pengguna dapat melihat data KPI penjualan seperti pencapaian target, performa individu, dan statistik penjualan lainnya. Diagram ini menunjukkan bahwa proses ini secara informatif dan pasif dan pengguna tidak perlu memberikan input tambahan.



Gambar 3.13. Activity Diagram Modul(Sales KPI)

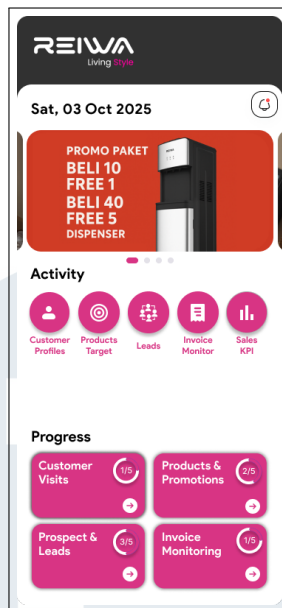
L Mockup UI Aplikasi

Salah satu tahapan penting setelah penyusunan diagram UML adalah pembuatan *Mockup UI*, yang dimaksudkan untuk memberikan gambaran visual mengenai tata letak dan interaksi *UI* sesuai dengan alur fitur yang telah dirancang. Mockup ini berfungsi sebagai panduan dalam implementasi *UI* selama pengembangan aplikasi, membantu pengembang memahami struktur dan alur penggunaan aplikasi. Berikut ini adalah beberapa contoh mock-up *UI*:



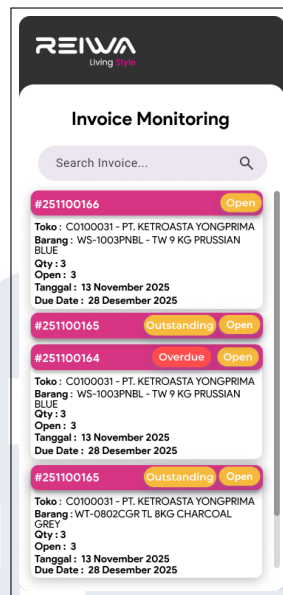
Gambar 3.14. Mockup UI Login

Gambar 3.13 menampilkan *Mockup UI* halaman awal aplikasi, yaitu tampilan *Login*. Warna yang digunakan pada *mockup* pada tombol submit sesuai dengan warna dari Reiwa itu tersendiri, dan tampilan halaman awal bersifat sementara dan bertujuan sebagai gambaran awal dan perubahan warna akan dilakukan setelah mendapat konfirmasi dari *supervisor*. Pada *mockup* tersebut terdapat kotak berisi sejumlah *field* yang digunakan untuk input yang harus diisi user untuk melakukan autentikasi dan kemudian akan dialihkan ke halaman *dashboard*.



Gambar 3.15. Mockup UI Login

Pada gambar 3.14 menampilkan *Mockup UI* halaman *Dashboard* yang muncul setelah *username* dan *password* berhasil melalui halaman pengisian data sebelumnya. Pada tampilan ini, *user* dapat melihat modul *Activity* yang disajikan dalam bentuk lingkaran sederhana agar tampilan lebih simpel dan eye-catching. Terdapat juga carousel di atasnya yang berguna sebagai mengasih tau informasi ada promo dari produk apa saja. Terdapat juga modul *Progress* yang berfungsi sebagai tracker bahwa sudah berapa jauh progress atau target.



Gambar 3.16. Mockup UI Login

Pada gambar 3.15 menampilkan *Mockup UI* halaman *Invoice Monitoring* yang muncul ketika mengklik salah satu modul yaitu modul *Invoice Monitoring*. Pada tampilan ini, *Invoice Monitoring* dapat melihat invoice-invoice yang terdapat dari sales ketika *Login*. Terdapat fitur search box untuk mencari status dari invoice tersebut yaitu *Overdue* atau *Outstanding*. Data-data tersebut bersifat *accordion*, jadi ketika di klik *accordion* tersebut akan keluar data-data seperti *DocumentNumber*, *nama Customer*, *CardCode*, *Item*, dan *Quantity/Qty*.

3.5.4 Implementasi Web Application

A Implementasi Integrasi Data Excel ke Database MySQL

```

1 $filePath = __DIR__ . "/generate sn arisa from yono \ Desember
  2025\SN_99999_9999.xlsx";
2 $sheet = $spreadsheet->getSheet(1);
3 if (!$sheet) {
4     return [];
5 }
6 $sheetsData[] = [
7     "sheet" => $sheetName,
8     "no_dn" => $deliverynumber,
9     "branch" => $cabang_toko,
10    "models" => array_map(function ($sn_list,
    $modelName) {

```



```

11         return [
12             "model" => $modelName,
13             "nomor produk" => $nomor_produk,
14             "jumlah" => count($nomor_produk)
15         ];
16     }, $models, array_keys($models))
17 ];
18 }
19 function convertToYmd($str)
20 {
21     $day = substr($str, 0, 2);
22     $month = substr($str, 2, 2);
23     $year = substr($str, 4, 2);
24
25     // Anggap semua 2 digit tahun itu 2000-an
26     $year = 2000 + (int)$year;
27
28     return sprintf("%04d-%02d-%02d", $year, $month, $day);
29 }
30 $fileName = basename($filePath);
31 $data = readExcelFile($filePath);

```

Kode 3.1: Potongan kode implementasi integrasi PHP dengan excel dan mySQL

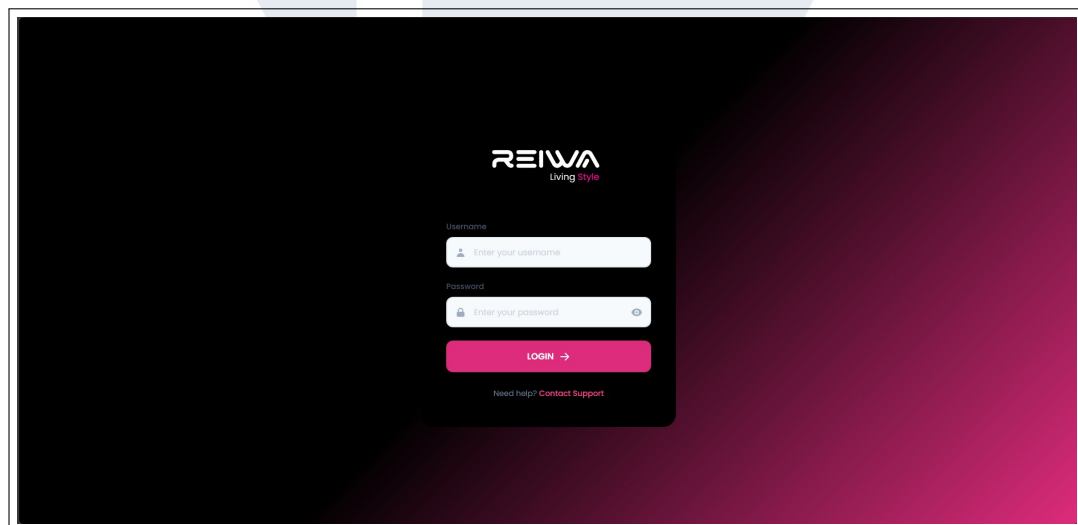
Integrasi PHP dengan excel dan mySQL dilakukan menggunakan *PHP* dengan framework Laravel seperti kode pada kode 3.1, protokol yang digunakan adalah *ETL*. *ETL* meliputi proses ekstraksi dari data dari sumber, transformasi sesuai kebutuhan dan pemuatan ke dalam data dan dapat diimplementasikan melalui berbagai teknologi dan platform [6]. Data-data dari excel di input dalam line *filepath* yang berfungsi untuk menjalankan excel tersebut dan juga line *sheet* yang berfungsi sebagai jumlah sheet yang terdapat dalam file excel tersebut. Program PHP tersebut akan memproses data excel tersebut lalu data dari excel akan masuk ke dalam mySQL dengan tabel dan kolom yang sudah dibuat terdapat pada gambar 3.16.

3.5.5 Tampilan hasil dari data excel ke dalam database

#	1 id	2 tanggal	3 delivery_note	4 preorder	5 design	6 jumlah	7 nomor_produk	8 cabang_toko	9 deskripsi
1	1	2025-12-12	99999	99999	WS-0851PGN	50	9999999999	AFRIKA	(NULL)
2	2	2025-12-10	99999	99999	WS-0851PGN	50	9999999999	JEPANG	(NULL)
3	3	2025-12-10	99999	99999	WS-0851PGN	50	9999999999	JAKARTA	(NULL)
4	4	2025-12-10	99999	99999	WS-0851PGN	50	9999999999	JAKARTA	(NULL)
5	5	2025-12-10	99999	99999	WS-0851PGN	50	9999999999	JAKARTA	(NULL)
6	6	2025-12-10	99999	99999	WS-0851PGN	50	9999999999	JAKARTA	(NULL)
7	7	2025-12-10	99999	99999	WS-0851PGN	50	9999999999	JAKARTA	(NULL)
8	8	2025-12-10	99999	99999	WS-0851PGN	50	9999999999	JAKARTA	(NULL)
9	9	2025-12-10	99999	99999	WS-0851PGN	50	9999999999	JAKARTA	(NULL)
10	10	2025-12-10	99999	99999	WS-0851PGN	50	9999999999	JAKARTA	(NULL)

Gambar 3.17. Tampilan data excel dalam database

3.5.6 Tampilan Login



Gambar 3.18. Tampilan Login

Gambar 3.16 menampilkan tampilan halaman *login Web Application*, di mana pengguna wajib mengisi *field*: *Username*, dan *Password*. Data username dan password didapat dari aplikasi Postman. Desain antarmuka menggunakan palet warna hitam dan pink yang disesuaikan dengan identitas visual logo Reiwa, dan juga sesuai persetujuan *Supervisor*, demi menjaga konsistensi *branding*.

```

1 public function loginSSO(Request $request)
2 {
3     $request->validate([
4         'username' => 'required|string',

```

```

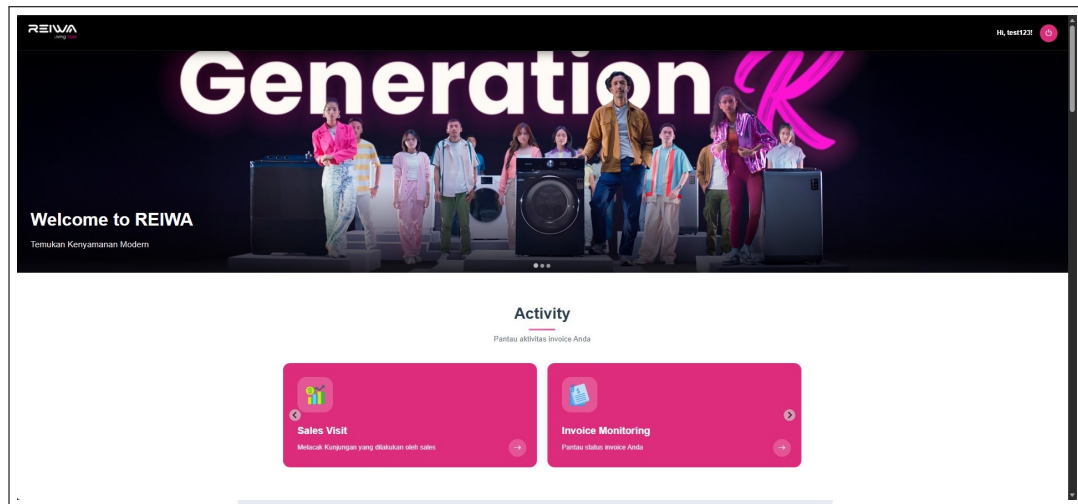
5         'password' => 'required|string',
6     ]);
7
8     try {
9         $response = Http::post(env('SSO_BASE_URL').'/oauth/token',
10         [
11             'grant_type' => 'password',
12             'client_id' => env('SSO_CLIENT_ID'),
13             'client_secret' => env('SSO_CLIENT_SECRET'),
14             'username' => $request->username,
15             'password' => $request->password,
16         ]);
17
18         if ($response->successful()) {
19             $data = $response->json();
20             Session::put('sso_access_token', $data['access_token
21             ']);
22             Session::put('login_method', 'sso');
23             return redirect()->route('dashboard');
24         }
25
26         return back()->with('error', 'Login SSO gagal!');
27     } catch (\Exception $e) {
28         return back()->with('error', 'Tidak dapat terhubung ke
server SSO');
29     }
30 }

```

Kode 3.2: Potongan kode Single-Sign On

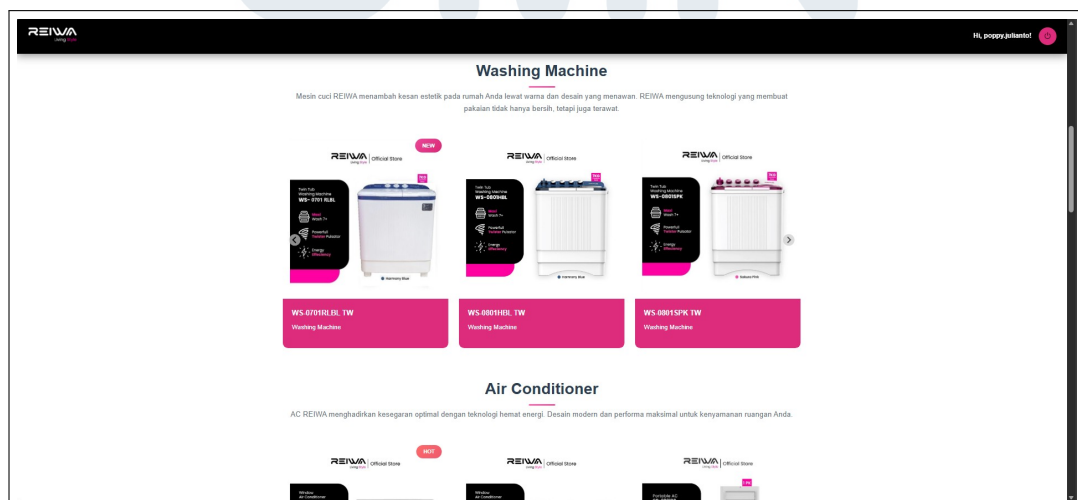
Fitur *login* aplikasi web dirancang untuk memudahkan pengguna dengan menggunakan Single-Sign On *SSO* yang membuat user tidak perlu mengingat username dan password cukup dengan sekali login untuk akses ke semua sistem yang terhubung. Single-Sign On *SSO* dapat memberikan otentikasi ke user tanpa harus mengungkapkan identitas atau informasi sensitif kepada provider [7]. Potongan kode 3.2 terdapat grant-type, client-id, client-secret, username, dan password yang didapat dalam aplikasi *Postman* agar Single-Sign On berfungsi. `Http::post` merupakan fungsi yang penting, karena harus memanggil dari file `.env` yang berisi link ip tokennya.

3.5.7 Tampilan Homepage



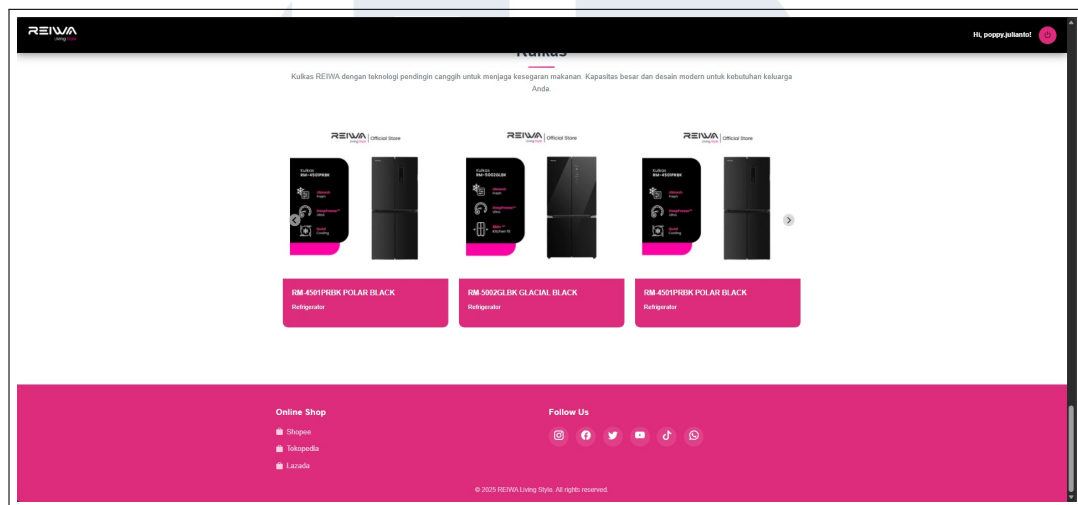
Gambar 3.19. Tampilan Homepage

Setelah pengguna menyelesaikan proses login, homepage aplikasi ditunjukkan pada Gambar 3.19. Tampilan ini pada dasarnya mengikuti rancangan awal yang sudah ditentukan dalam mockup UI, tetapi beberapa elemen diubah karena dianggap tidak penting atau tidak diperlukan untuk implementasi akhir [8]. Beberapa bagian penting, seperti Activity, yang menampilkan modul *Invoice Monitoring*, dan modul *Sales Visit*, menyambut pengguna pada halaman utama ini. Di bagian atas halaman terdapat navigasi bar yang sederhana tetapi berguna yang terdiri dari logo Reiwa sebagai identitas aplikasi, dan tombol untuk keluar dari akun.



Gambar 3.20. Tampilan Homepage

Produk digambarkan secara lebih rinci di Gambar 3.20. Mesin cuci, AC, fryer, kabinet, dispenser air, dan kulkas adalah beberapa produk yang ditampilkan. Untuk melihat semua produk ini, pengguna dapat melihat daftar produk PT Caturaga Tiara Persada. Tampilan produk dirancang sesuai dengan desain katalog, sehingga pelanggan lebih mudah memahami dan membedakan setiap kategori produk. Carousel juga membuat tampilan lebih interaktif dan responsif terhadap ukuran layar di bagian ini [9].



Gambar 3.21. Tampilan Homepage

Gambar 3.21. Pada bagian bawah, terdapat berbagai tautan ke kanal informasi dan media sosial perusahaan, seperti Instagram, Facebook, Twitter, YouTube, TikTok, dan WhatsApp. Selain media sosial, footer juga menawarkan akses langsung ke toko online perusahaan di beberapa platform populer, seperti Shopee, Tokopedia, dan Lazada. Tujuan penempatan tautan ini adalah untuk membuat pengguna lebih mudah menemukan informasi tambahan dan membeli barang secara online. Aplikasi dengan footer yang lengkap membantu perusahaan berkomunikasi dan menjual di luar.

```
1 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/
  @splidejs/splide@4.1.4/dist/css/splide.min.css" />
2 document.addEventListener("DOMContentLoaded", function() {
3   new Splide("#splideCarousel", {
4     type: "loop",
5     perPage: 1,
6     autoplay: true,
7     interval: 4000,
8     arrows: false,
```

```

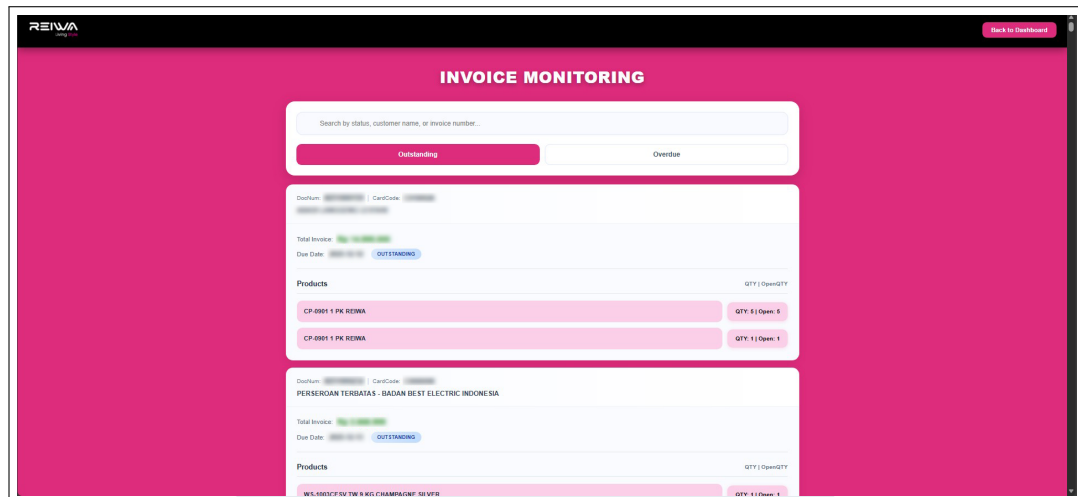
9      pagination: true,
10     pauseOnHover: true,
11     speed: 800,
12   }).mount();
13
14   const productConfig = {
15     type: "loop",
16     perPage: 3,
17     gap: "1.5rem",
18     arrows: true,
19     rewind: true,
20     pagination: false,
21     breakpoints: { 1024: { perPage: 2 }, 768: { perPage: 1.5
22 }, 480: { perPage: 1 } }
23   };
24   ["#produkMesinCuci", "#produkAC", "#produkCabinet", "#
25 produkWaterDispenser", "#produkRefrigerator"]
26     .forEach(id => new Splide(id, productConfig).mount());
27
28   new Splide("#produkAirFryer", productConfig).mount();
29
30   new Splide("#invoiceCarousel", {
31     type: "loop", perPage: 2, gap: "1.5rem", arrows: true,
32     rewind: true, pagination: false,
33     breakpoints: { 768: { perPage: 1 } }
34   }).mount();

```

Kode 3.3: Potongan Kode splideJs

Pada halaman *homepage*, dirancang agar pengguna dapat melihat produk yang ditawarkan serta aktivitas yang tersedia. Aktivitas tersebut dapat di-slide ke sebelah, seperti fitur carousel yang berjalan otomatis dan berulang (autoplay dan loop) sesuai dengan mekanisme micromodifications pada adaptive user interfaces. Section produk juga dapat di-slide untuk melihat berbagai produk yang disediakan oleh perusahaan, seperti AC, Washing Machine, Air Fryer, Cabinet, Water Dispenser, dan Kulkas. Pada potongan kode 3.3 merupakan kode dari *SplideJS* yang berfungsi sebagai carousel dengan autoplay dan loop pada section Activity dan Product [10]. Fitur ini mendukung konsep antarmuka adaptif dan dinamis, memungkinkan pengguna memperoleh pengalaman yang personal dan interaktif sesuai pendekatan dalam kerangka kerja multivariant UI.

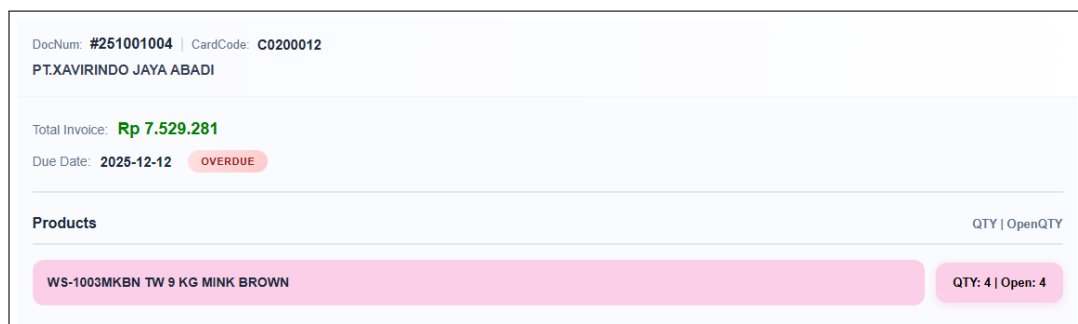
3.5.8 Tampilan modul Invoice Monitoring



Gambar 3.22. Tampilan Menu Modul

Gambar 3.22 menampilkan halaman *Invoice Monitoring* yang muncul setelah pengguna memilih modul activity dari halaman *homepage*. Halaman ini menyediakan informasi viewing seperti *DokumenNumber*, *CodeProduk*, *namaCustomer*, *Jatuh tempo*, *Produk*, *jumlahproduk*. *Overdue* yang berarti melewati jatuh tempo dan *Outstanding* sebelum jatuh tempo.

Perbedaan *Quantity* dan *OpenQTY* yaitu *quantity* merupakan jumlah total barang yang tercantum dalam invoice atau purchase order secara keseluruhan sejak awal transaksi dibuat, dan *OpenQTY* merupakan jumlah barang yang masi belum dikirim, belum diterima, atau belum diselesaikan pembayarannya.



Gambar 3.23. contoh validasi Perhitungan Overdue

Gambar 3.23 menampilkan contoh validasi perhitungan Total invoice yang bersifat overdue didapat dari jumlah quantity yaitu 4 dari database diambil price

produk dengan harga $\text{Rp.}1.695.784 \times 4 = \text{Rp. } 6.783.136$, dan terdapat PPN/VAT sebesar 11 persen sekitar $\text{Rp.}746.145$ dari SAP dengan ini total menjadi $\text{Rp. } 7.529.281$.

DocNum: #251100216 | CardCode: C0000098
 PERSEROAN TERBATAS - BADAN BEST ELECTRIC INDONESIA

Total Invoice: **Rp 2.668.999**
 Due Date: 2026-01-16 **OUTSTANDING**

Products QTY | OpenQTY

WS-1003MKBN TW 9 KG MINK BROWN	QTY: 1 Open: 1
--------------------------------	------------------

Gambar 3.24. contoh validasi Perhitungan Outstanding

Gambar 3.24 menampilkan contoh validasi perhitungan Total invoice yang bersifat Outstanding didapat dari jumlah quantity yaitu 1 dari database diambil price produk dengan harga $\text{Rp. } 2.404.504 \times 1 = \text{Rp. } 2.404.504$, dan terdapat PPN/VAT sebesar 11 persen sekitar $\text{Rp.}264.495$ dari SAP dengan ini total menjadi $\text{Rp. } 2.668.999$.

3.5.9 Tampilan modul Sales Visit

Sales Visit Monitoring Add Sales Plan

Show: 10 entries

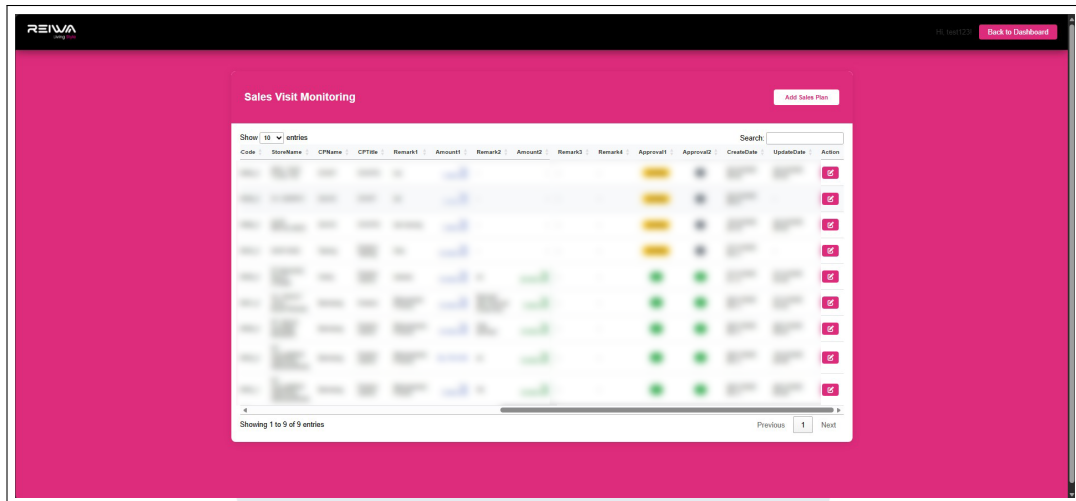
#	ID	SpCode	SpName	PlantDate	BranchCode	BranchName	City	CardCode	CardName	CardType	StoreCode	StoreName	CPName	CPTitle
1	1						KOTA TANGERANG	-	-	Calon				
2	2						JAKARTA TIMUR	-	-	Calon				
3	3						JAKARTA TIMUR	-	-	Calon				
4	4						JAKARTA SELATAN	-	-	Calon				
5	5						JAKARTA SELATAN	-	-	Calon				
6	6						JAKARTA BARAT	-	-	Calon				
7	7						JAKARTA SELATAN	-	-	Calon				
8	8						JAKARTA SELATAN	-	-	Calon				
9	9						JAKARTA SELATAN	-	-	Calon				

Showing 1 to 9 of 9 entries Previous 1 Next

Gambar 3.25. Tampilan Modul Sales Visit

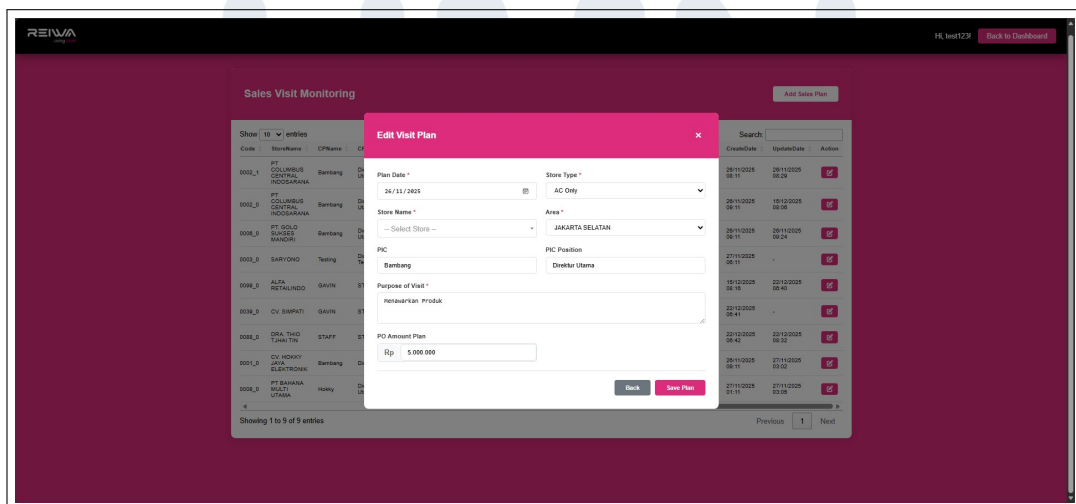
Gambar 3.23 menampilkan halaman *Sales Visit* yang muncul setelah pengguna memilih modul activity dari halaman *homepage*. Halaman ini

menampilkkan list informasi seperti *ID*, *SLP Code*, *PlanDate* dll yang bersifat dummy.



Gambar 3.26. Tampilan Modul Sales Visit

Gambar 3.24 menampilkan halaman *Sales Visit* yang muncul setelah pengguna memilih modul activity dari halaman *homepage*. Terdapat modul action yang berfungsi sebagai edit data, jadi pengguna dapat mengedit data jika terdapat data-data yang tidak benar atau typo.



Gambar 3.27. Tampilan Modul Sales Visit (Edit Visit Plan)

Gambar 3.25 menampilkan halaman *Sales Visit Edit Sales Plan*, Pada halaman tersebut muncul sebuah pop-up yang bernama edit sales plan yang muncul

ketika edit button di tekan. Pengguna dapat mengedit *Plan Date*, *Store Type*, *Store Name*, *Area*, *PIC*, *PIC Position*, *Purpose of Visit*, dan *PO Amount plan*.

The screenshot displays the 'Create Visit Plan' modal window in the REIVA Sales Visit Monitoring application. The modal is a pink overlay with a close button (X) in the top right corner. It contains the following fields:

- Plan Date ***: A date input field with a calendar icon, showing 'dd/mm/yyyy'.
- Store Type ***: A dropdown menu with the option 'Select Store Type'.
- Store Name ***: A dropdown menu with the option 'Select Store'.
- Area ***: A dropdown menu with the option 'Select Area'.
- PIC**: A text input field.
- PIC Position**: A text input field.
- Purpose of Visit ***: A text area.
- PO Amount Plan**: A text input field with a currency symbol 'Rp' and a value '0'.

At the bottom of the modal are two buttons: 'Back' and 'Save Plan'. The background shows a table with columns 'Code', 'StoreName', 'CPName', and 'CP'. The table is titled 'Sales Visit Monitoring' and has a search bar and 'Add Sales Plan' button at the top right. The table shows 9 entries, with the first one being '159'.

Gambar 3.28. Tampilan Modul Sales Visit (Create Visit Plan)

Gambar 3.26 menampilkan halaman *Sales Visit Create Visit Plan*. pada halaman tersebut, pengguna dapat membuat *Visit Plan* secara manual dengan mengisi *Plan Date*, *Store Type*, *Store Name*, *Area*, *PIC*, *PIC Position*, *Purpose of Visit*, dan *PO Amount plan*.

3.5.10 Summary Sales Visit

Berdasarkan data Sales Visit Plan terdapat 9 rencana kunjungan dalam periode November-Desember 2025. Kunjungan dilakukan oleh 2 sales yaitu Hendiarto Sentosa (SLP Code 159) dengan 7 kunjungan dan Wilson Julian (SLP Code 150) dengan 2 kunjungan, seluruhnya di wilayah JABODETABEK.

Distribusi lokasi terbanyak di Jakarta Selatan (4 kunjungan) dan Jakarta Timur (3 kunjungan), sisanya Jakarta Barat (1) dan Kota Tangerang (1). Produk yang dipromosikan didominasi oleh AC (6 kunjungan atau 67 persen), kemudian Cabinet (2) dan Hybrid (1). Target kunjungan meliputi retail modern (ALFA RETAILINDO, PT Columbus), toko elektronik (CV Hokky Jaya), dan distributor dengan PIC mayoritas pada level Direktur Utama dan Staff.

3.5.11 Pengujian Aplikasi Web Application

Proses pengujian dilakukan untuk mengevaluasi fungsionalitas aplikasi, termasuk penggunaan setiap modul yang tersedia, serta verifikasi bahwa fitur-fitur seperti pengiriman email, perubahan status data, dan penambahan data ke dalam daftar berjalan sesuai harapan. Pengujian dilakukan dengan menggunakan metode *Black box testing*. Metode pengujian *Black box testing* merupakan proses yang dilakukan tanpa pengetahuan mengenai detail internal sistem yang diuji dan *testers* hanya dapat mengumpulkan informasi melalui saluran publik dan melakukan pengujian dari luar sistem, tanpa akses ke kode sumber maupun dokumentasi internal [11].

Tabel 3.4. Hasil Pengujian *Black Box Testing* Aplikasi Web

Test ID	Skenario Pengujian	Fitur yang Diuji	syarat	Test Data / Input	Ekspektasi Hasil	Status
TC-001	Input kredensial valid	Login	User memiliki akun aktif	Username: gavin.prasetya Password: Pass123! SSO Name: Gavin	Aplikasi berhasil login dan masuk ke halaman homepage	Berhasil
TC-002	Input kredensial tidak valid	Login	-	Username: gavin.wrong Password: Pass123!	Muncul notifikasi error "Login Gagal"	Berhasil
TC-003	Klik tombol Contact Support	Contact Support	-	Klik button "Contact Support"	Redirect ke Whatsapp Reiwa	Berhasil
TC-004	Tampilkan password	Show Password	-	Password: Pass123! Aksi: Klik icon mata	Password ditampilkan dalam bentuk teks	Berhasil

Bersambung ke halaman berikutnya

Tabel 3.4. Hasil Pengujian *Black Box Testing* Aplikasi Web (lanjutan)

Test ID	Skenario Pengujian	Fitur yang Diuji	Prasyarat	Test Data / Input	Ekspektasi Hasil	Status
TC-005	Akses halaman homepage	Homepage	User sudah login	-	Menampilkan menu utama dengan benar	Berhasil
TC-006	Logout dari aplikasi	Logout	User sudah login	Klik button "Logout"	Redirect ke halaman Login	Berhasil
TC-007	Interaksi dengan product card	Products Card	User di homepage	Klik salah satu product card	Product dapat di-zoom in dan zoom out	Berhasil
TC-008	Akses halaman Invoice Monitoring	Invoice Monitoring	User sudah login	Klik module "Invoice Monitoring"	Menampilkan list data invoice dengan detail	Berhasil
TC-009	Pencarian invoice	Search Invoice	Ada data invoice	Input: "Toko ABC" atau "INV-001"	Data invoice yang sesuai ditampilkan	Berhasil
TC-010	Filter invoice Outstanding	Filter Outstanding	Ada invoice Outstanding	Klik filter "Outstanding"	Hanya menampilkan invoice sebelum jatuh tempo	Berhasil
TC-011	Filter invoice Overdue	Filter Overdue	Ada invoice Overdue	Klik filter "Overdue"	Hanya menampilkan invoice melewati jatuh tempo	Berhasil

Bersambung ke halaman berikutnya

Tabel 3.4. Hasil Pengujian *Black Box Testing* Aplikasi Web (lanjutan)

Test ID	Skenario Pengujian	Fitur yang Diuji	Prasyarat	Test Data / Input	Ekspektasi Hasil	Status
TC-012	Akses halaman Sales Visit	Sales Visit	User sudah login	Klik module "Sales Visit"	Menampilkan tabel data sales secara lengkap	Berhasil
TC-013	Input tanggal kunjungan	Plan Date	Modal create terbuka	Tanggal: 2025-01-20	Tanggal berhasil dipilih menggunakan date picker	Berhasil
TC-014	Pilih tipe toko	Store Type	Modal create terbuka	Pilihan: "Modern Trade"	Tipe toko berhasil dipilih dari dropdown	Berhasil
TC-015	Pilih nama toko	Store Name	Store Type dipilih	Pilihan: "Indomaret Sunter"	Nama toko sesuai tipe dan area berhasil dipilih	Berhasil
TC-016	Pilih area kunjungan	Area	Modal create terbuka	Pilihan: "Jakarta Utara"	Area berhasil dipilih dari dropdown	Berhasil
TC-017	Input nama PIC	PIC Name	Modal create terbuka	Input: "Budi Santoso"	Nama PIC berhasil disimpan	Berhasil
TC-018	Input jabatan PIC	PIC Position	Modal create terbuka	Input: "Store Manager"	Jabatan PIC berhasil disimpan	Berhasil

Bersambung ke halaman berikutnya


Tabel 3.4. Hasil Pengujian *Black Box Testing* Aplikasi Web (lanjutan)

Test ID	Skenario Pengujian	Fitur yang Diuji	Prasyarat	Test Data / Input	Ekspektasi Hasil	Status
TC-019	Input tujuan kunjungan	Purpose of Visit	Modal create terbuka	Input: "Demo produk AC Reiwa"	Tujuan kunjungan berhasil disimpan	Berhasil
TC-020	Input nominal PO	PO Amount Plan	Modal create terbuka	Input: 10000000	Nominal berhasil diformat dan disimpan	Berhasil
TC-021	Simpan visit plan (lengkap)	Save Plan	Semua field diisi	Klik "Save Plan"	Data tersimpan ke database dan form ditutup	Berhasil
TC-022	Simpan visit plan (tidak lengkap)	Save Plan Validation	Modal create terbuka	Field wajib kosong Klik "Save"	Sistem menampilkan pesan error validasi	Berhasil
TC-023	Batalan pembuatan plan	Back Button	Modal create terbuka	Klik "Back"	Kembali ke halaman Sales Visit tanpa menyimpan	Berhasil
TC-024	Ubah jumlah entries	Show Entries	Ada data >10	Pilih: "25" entries	Menampilkan 25 data per halaman	Berhasil
TC-025	Navigasi pagination	Pagination	Ada data >10	Klik "Next" atau "Previous"	Halaman data berubah sesuai navigasi	Berhasil

Tabel 3.5. Ringkasan Hasil Pengujian *Black Box Testing*

Kategori	Total Test Cases	Passed	Failed	Pass Rate
Login & Authentication	4	4	0	100%
Homepage & Navigation	4	4	0	100%
Invoice Monitoring	3	3	0	100%
Sales Visit	14	14	0	100%
TOTAL	25	25	0	100%

Tabel 3.6. Informasi Lingkungan Pengujian

Parameter	Detail
Tester	Gavin Prasetya (NIM: 00000068873)
Tanggal Testing	22–23 Desember 2024
Browser	Microsoft Edge (64-bit)
Operating System	Windows 11 Pro 64-bit
Network	Local Development (localhost:8000)
Database	MySQL 8.0.30
PHP Version	8.3.14 CLI
Laravel Framework Version	12.34.0
Testing Method	Manual Black Box Testing
Reviewed by	 Yusuf Rahmanto (IT Supervisor)

Berdasarkan hasil pengujian *black box* yang dilakukan terhadap aplikasi *Web Application*, seluruh fitur fungsional pada setiap modul telah berhasil berjalan sesuai harapan. Tidak ditemukan *bug* atau *error* yang berasal dari sisi web. Fitur *Invoice Monitoring*, dan *Sales Visit*. Aplikasi akan tetap dilakukan pemeliharaan secara berkala untuk memastikan kinerja optimal dan adaptasi terhadap kebutuhan pengguna di masa mendatang[11] .

3.6 Kendala dan Solusi yang Ditemukan

A Kendala

Selama pelaksanaan kerja magang di PT. Caturaga Tiara Persada, beberapa kendala yang memengaruhi efisiensi dan kualitas kerja, antara lain:

1. Kesulitan dalam mencari referensi teknis terkait desain *UI* dan implementasi fitur digunakan, serta seringnya perubahan kebutuhan fungsional dari *supervisor*. Hal ini disebabkan oleh kurangnya dokumentasi atau tutorial spesifik untuk pendekatan yang digunakan, serta seringnya perubahan kebutuhan fungsional dari *supervisor*.
2. Jarak tempuh antara tempat tinggal dengan lokasi magang yang cukup jauh. Kondisi ini menyebabkan waktu perjalanan lebih lama, sehingga berpotensi menimbulkan kelelahan dan mengurangi efektivitas dalam menjalani aktivitas magang sehari-hari. Selain itu, faktor jarak juga menuntut pengaturan waktu yang lebih disiplin agar tidak terjadi keterlambatan.
3. Kesulitan dalam berkomunikasi dengan rekan kerja maupun atasan disebabkan karena malu dan kurang percaya diri ketika menyampaikan pendapat atau bertanya mengenai pekerjaan.

B Solusi

Untuk mengatasi kendala-kendala tersebut, beberapa solusi yang diterapkan adalah:

1. Berusaha memaksimalkan kemampuan diri sendiri untuk mencari referensi desain *UI* atau dengan menanyakan *Supervisor* mengenai desain yang bagus seperti apa. Selain itu, pencarian referensi desain *UI* dilakukan melalui website lain di internet yang memiliki tampilan sederhana namun menarik, sesuai dengan preferensi *Supervisor* dan *IT Manager* untuk dijadikan inspirasi dalam merancang antarmuka aplikasi.
2. Manajemen waktu yang lebih baik seperti berangkat lebih awal agar perjalanan tidak tergesa-gesa, memanfaatkan mode transportasi yang lebih efisien dengan menggunakan transportasi umum yang lebih cepat dan terjangkau, dan menjaga kondisi fisik dengan istirahat yang cukup.

3. Melatih keberanian secara bertahap dan membangun hubungan baik dengan rekan kerja melalui interaksi informal seperti berbincang ringan di luar pekerjaan agar komunikasi lebih cair.



UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA