

BAB 3

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Program kerja magang di PT Pabrik Cat dan Tinta Pacific (Pacific Paint) dilaksanakan pada Departemen IT dengan peran sebagai IT Development Intern. Tanggung jawab utama yang diemban selama masa magang adalah merancang dan mengembangkan Sistem Manajemen Gudang Receh. Pelaksanaan tugas ini dilakukan di bawah bimbingan dan arahan Bapak Kukuh Ardy Nugroho selaku IT Development and Integration Lead serta berkoordinasi dengan Bapak Aldi Setiawan selaku supervisor magang untuk keperluan administrasi.

Dalam proses pengembangan, koordinasi dan diskusi secara aktif dilakukan guna menghindari terjadinya miskomunikasi. Evaluasi progres kerja juga dilakukan untuk memastikan pengembangan berjalan sesuai dengan baik.

3.2 Tugas yang Dilakukan

Tugas yang dilakukan selama pelaksanaan program kerja magang di PT Pabrik Cat dan Tinta Pacific (Pacific Paint) berfokus pada perancangan dan pengembangan Sistem Manajemen Gudang Receh.

Tahap perancangan dilakukan dengan menggunakan kebutuhan fungsionalitas yang telah ditetapkan sebagai dasar pembuatan diagram yang digunakan untuk menggambarkan cara kerja sistem. Diagram yang dibuat meliputi *System Architecture Diagram* untuk menggambarkan struktur teknis sistem, *Use Case Diagram* untuk mengidentifikasi aktor dan fungsionalitas sistem, *Activity Diagram* untuk menggambarkan alur proses, *Sequence Diagram* untuk menunjukkan interaksi antar komponen sistem, *Physical Entity Relationship Diagram* (ERD) untuk struktur basis data, dan *mockup* antarmuka pengguna untuk tampilan sistem.

Tahap pengembangan dilakukan secara menyeluruh mencakup bagian *frontend* dan *backend*. Pengujian dilakukan untuk memastikan setiap fitur berfungsi dengan baik dan sesuai dengan kebutuhan pengguna. Kolaborasi dengan pengguna juga dilakukan melalui proses demo untuk mendapatkan umpan balik dan melakukan perbaikan agar sistem yang dikembangkan dapat memberikan solusi yang optimal.

3.3 Uraian Pelaksanaan Magang

Pelaksanaan kerja magang di PT Pabrik Cat dan Tinta Pacific dapat dilihat pada Tabel 3.1.

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu Ke-	Pekerjaan yang dilakukan
1	Menjalani program pelatihan internal untuk karyawan baru bagian pertama.
2	Melanjutkan menjalani program pelatihan internal untuk karyawan baru.
3	Proses <i>onboarding</i> , eksplorasi API sistem perusahaan, dan melakukan perancangan awal untuk proyek PPP.
4	Melanjutkan perancangan dan memulai pengembangan <i>backend</i> proyek PPP yang mencakup pembuatan <i>flowchart</i> , dan membuat logika bisnis.
5	Melanjutkan pengembangan <i>backend</i> proyek PPP, eksplorasi file proyek <i>frontend</i> dan pengembangan tampilan <i>frontend</i> proyek PPP.
6	Integrasi <i>frontend</i> dengan <i>backend</i> yang dibuat untuk proyek PPP.
7	Melanjutkan integrasi <i>frontend</i> dengan <i>backend</i> yang dibuat untuk proyek PPP dan melakukan <i>testing</i> serta <i>debugging</i> fitur.
8	eksplorasi dan setup untuk persiapan proyek WMS, mencakup diskusi, percobaan ambil data dari oracle, dan membangun bagian awal aplikasi <i>mobile</i> dengan <i>framework</i> Flutter.
9	diskusi lebih lanjut untuk proyek WMS, melakukan perancangan dengan membuat diagram UML dan <i>mockup</i> .
10	melanjutkan proses perancangan, lalu masuk ke tahap pengembangan <i>backend</i> proyek WMS untuk proses transfer masuk, keluar, dan pengelolaan rak.
11	melakukan modifikasi pada <i>backend</i> proyek WMS, perbaikan <i>bug</i> , perbaikan penamaan <i>endpoint</i> , dan mengembangkan tampilan aplikasi <i>mobile</i> .
Lanjut pada halaman berikutnya	

Lanjutan Tabel 3.1

Minggu Ke-	Pekerjaan yang dilakukan
12	Implementasi kamera sebagai pemindai <i>barcode</i> dan <i>qrcode</i> , integrasi aplikasi <i>mobile</i> dengan API proyek WMS, menerapkan <i>route guard</i> pada aplikasi, dan penerapan fitur <i>filtering</i> untuk riwayat transfer.
13	melanjutkan pengembangan tampilan aplikasi <i>mobile</i> , lanjut pengembangan tampilan web <i>dashboard</i> proyek WMS dengan fitur <i>filtering</i> dan <i>pagination</i>
14	penambahan fitur <i>searching</i> untuk <i>dashboard</i> WMS, modifikasi cara pemilihan rak saat proses transfer masuk di aplikasi <i>mobile</i> , dan modifikasi tampilan dan kondisi pembuatan struk transfer
15	Penambahan logika pengecekan kapasitas per dus, penambahan halaman <i>transfer report</i> , modifikasi <i>filter</i> , penambahan <i>loading spinner</i> , dan modifikasi halaman buat permintaan transfer
16	Penambahan <i>endpoint</i> dan tampilan untuk atur peran pengguna sistem WMS, melakukan <i>testing</i> manual dan perbaikan <i>bug</i>

3.4 Perangkat Penunjang Pelaksanaan Magang

Bagian ini menjelaskan perangkat yang digunakan selama pelaksanaan kegiatan magang mencakup perangkat keras dan perangkat lunak yang digunakan dalam proses pengembangan.

3.4.1 Perangkat Keras yang Digunakan

Tabel 3.2 memperlihatkan informasi perangkat keras yang digunakan selama pelaksanaan magang, yang mencakup nama perangkat, dan spesifikasi. Perangkat keras ini digunakan dalam mendukung kebutuhan pengembangan, pengujian, dan menjalankan aplikasi yang dibutuhkan.

Tabel 3.2. Daftar perangkat keras

Nama Perangkat	Spesifikasi
<i>Processor</i>	Intel i7-7700
<i>Random Access Memory (RAM)</i>	8GB DDR4
<i>Storage</i>	224 GB SSD
<i>Operating System</i>	Windows 10 Pro
<i>Graphical Processing Unit (GPU)</i>	Intel(R) HD Graphics 630

3.4.2 Perangkat Lunak yang Digunakan

Tabel 3.3 menyajikan daftar perangkat lunak yang digunakan selama pelaksanaan magang, yang mencakup nama perangkat lunak, dan keterangan versi, kegunaan, dan tautan dokumentasi perangkat lunak tersebut dipilih untuk mendukung proses pengembangan sistem *backend*, *frontend* dan aplikasi *mobile*.

Tabel 3.3. Daftar perangkat lunak dan keterangannya

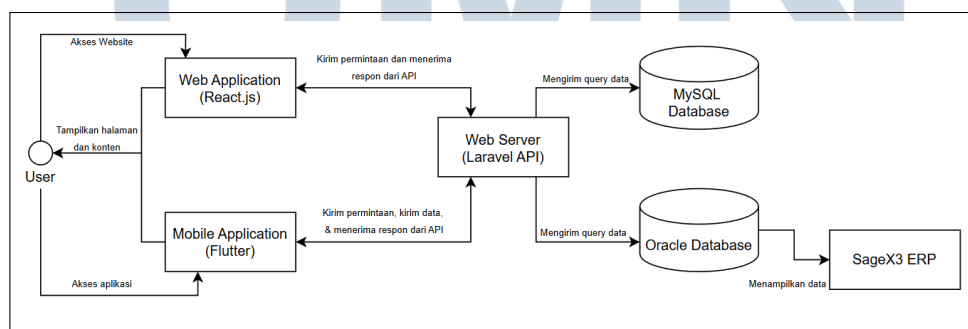
Nama Perangkat	Keterangan
Visual Studio Code	Versi 1.104.2 – <i>Integrated Development Environment (IDE)</i> yang digunakan dalam proses pengembangan untuk menulis, mengelola, dan melakukan <i>debugging</i> kode program [6].
MySQL Workbench	Versi 8.0.43 – Antarmuka grafis yang digunakan untuk mengelola <i>database</i> MySQL [7].
GitHub Desktop	Versi 3.5.4 – Antarmuka grafis yang digunakan untuk mengelola <i>Git repository</i> [8].
Postman	Versi 11.76.6 – Antarmuka grafis yang digunakan untuk menguji <i>endpoint API</i> [9].
DBeaver	Versi 25.2.4 – Antarmuka grafis <i>database</i> umum yang digunakan untuk mengelola database Oracle [10].
PHP	Versi 8.3.25 – Bahasa pemrograman <i>server-side</i> untuk pengembangan <i>backend</i> [11].
Lanjut pada halaman berikutnya	

Nama Perangkat	Keterangan
Laravel	Versi 9.52.16 – <i>Framework</i> untuk pengembangan sistem berbasis PHP [12].
React.js	Versi 18.3.1 – <i>Library</i> JavaScript untuk pengembangan antarmuka pengguna berbasis <i>web</i> [13].
Flutter	Versi 3.35.0 – <i>Framework</i> untuk pengembangan aplikasi <i>mobile</i> lintas <i>platform</i> [14].
MySQL	Versi 8.0.42 – Sistem Manajemen Basis Data Relasional (RDBMS) [15].
Oracle Database	Versi 12c – Sistem Manajemen Basis Data Relasional (RDBMS) yang dikembangkan oleh Oracle [16].

3.5 Perancangan dan Pengembangan Sistem

Bagian ini menjelaskan proses perancangan dan pengembangan sistem manajemen gudang receh di PT Pabrik Cat dan Tinta Pacific (Pacific Paint). Pembahasan mencakup tahap perancangan dengan *System Architecture Diagram*, *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Physical Entity Relationship Diagram* (ERD), *mockup* lalu tahap pengembangan dengan hasil implementasi melalui tangkapan layar tampilan aplikasi *mobile* dan *website*.

3.5.1 System Architecture Diagram



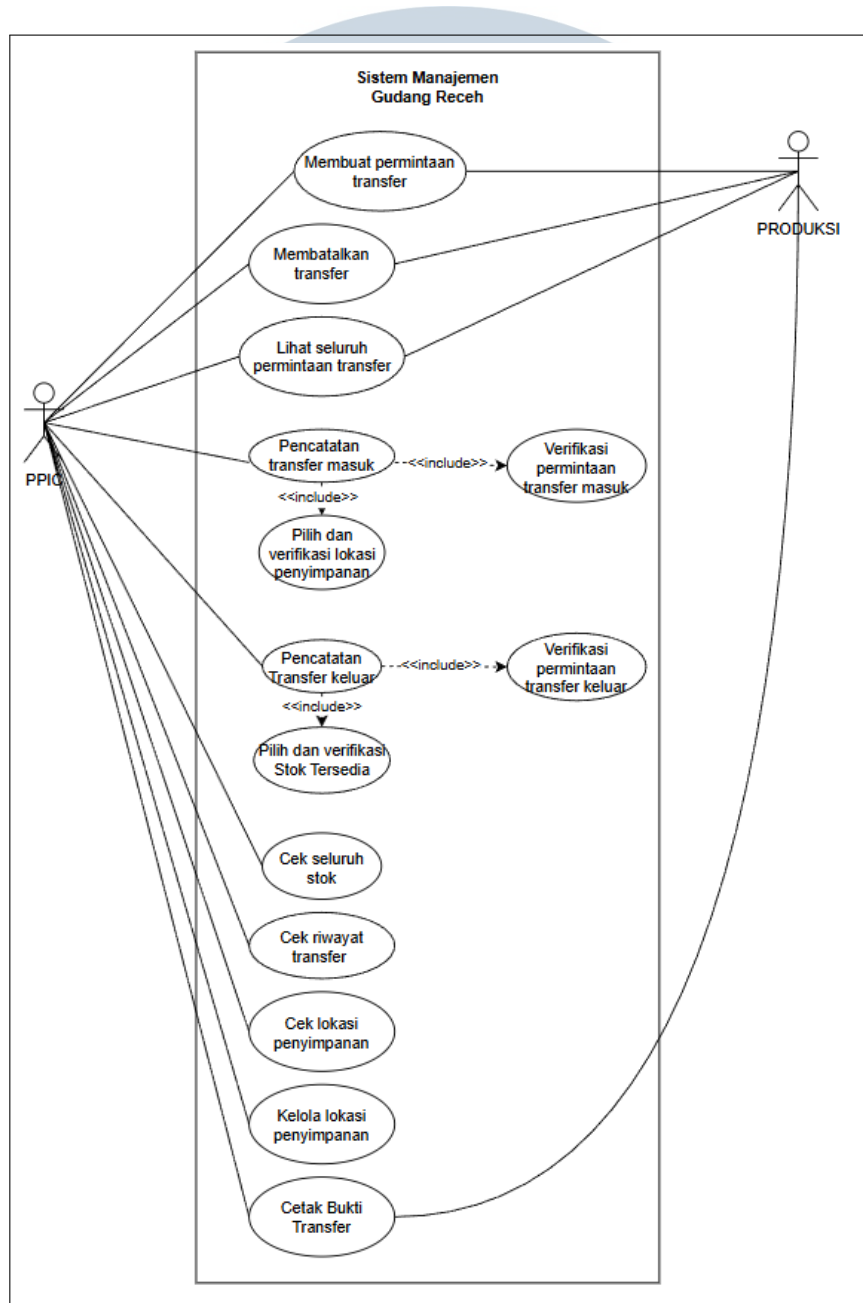
Gambar 3.1. *System Architecture Diagram* Sistem Manajemen Gudang Receh

Gambar 3.1 menampilkan *System Architecture Diagram* yang menggambarkan struktur dan mekanisme interaksi antar komponen yang membangun sistem secara keseluruhan. Sistem ini terdiri dari beberapa komponen utama, yaitu *Web Application*

berbasis React.js yang berfungsi sebagai *dashboard* untuk *monitoring* data; *Mobile Application* berbasis Flutter yang digunakan untuk keperluan proses transfer barang di lapangan; dan API berbasis *Laravel Framework* yang berperan sebagai jembatan komunikasi antara *frontend* dengan basis data. Pengelolaan data dalam sistem ini menggunakan dua basis data yang berbeda, yaitu basis data MySQL untuk menyimpan data Sistem Manajemen Gudang Receh dan basis data Oracle sebagai basis data sistem ERP perusahaan untuk mendapatkan data produk dan Work Order. Integrasi ini memungkinkan Sistem Manajemen Gudang Receh melengkapi data dari sistem ERP yang hanya menyimpan informasi area stok dengan menyimpan data lokasi spesifik setiap barang di gudang receh.



3.5.2 Use Case Diagram



Gambar 3.2. Use Case Diagram Sistem Manajemen Gudang Receh

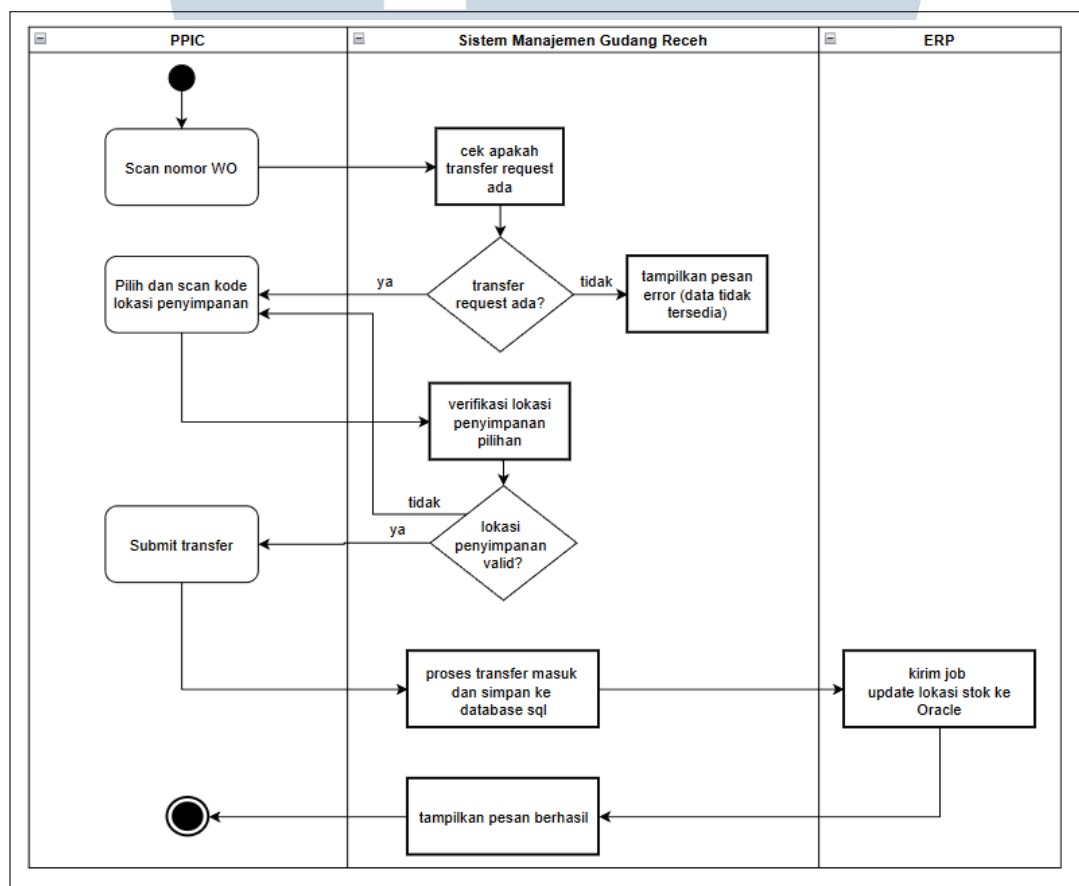
Pada sistem manajemen gudang receh, *Use Case Diagram* menggambarkan interaksi aktor dengan sistem, seperti yang ditunjukkan pada gambar 3.2. terdapat dua aktor, yaitu tim *Production Planning and Inventory Control*(PPIC) dan departemen Produksi. Kedua aktor memiliki akses untuk membuat permintaan transfer, membatalkan permintaan transfer, melihat seluruh permintaan transfer, dan mencetak bukti transfer. Selain itu, PPIC

juga memiliki beberapa fungsi tambahan untuk dapat melakukan pencatatan transfer barang, baik transfer masuk maupun transfer keluar, melihat stok yang tersedia, melihat riwayat transfer, serta mengelola lokasi penyimpanan.

3.5.3 Activity Diagram

Activity Diagram pada Sistem Manajemen Gudang Receh digunakan untuk memodelkan alur aktivitas yang terjadi dalam sistem, sehingga memberikan gambaran mengenai urutan proses serta aliran logika yang terjadi.

A Pencatatan Transfer Masuk

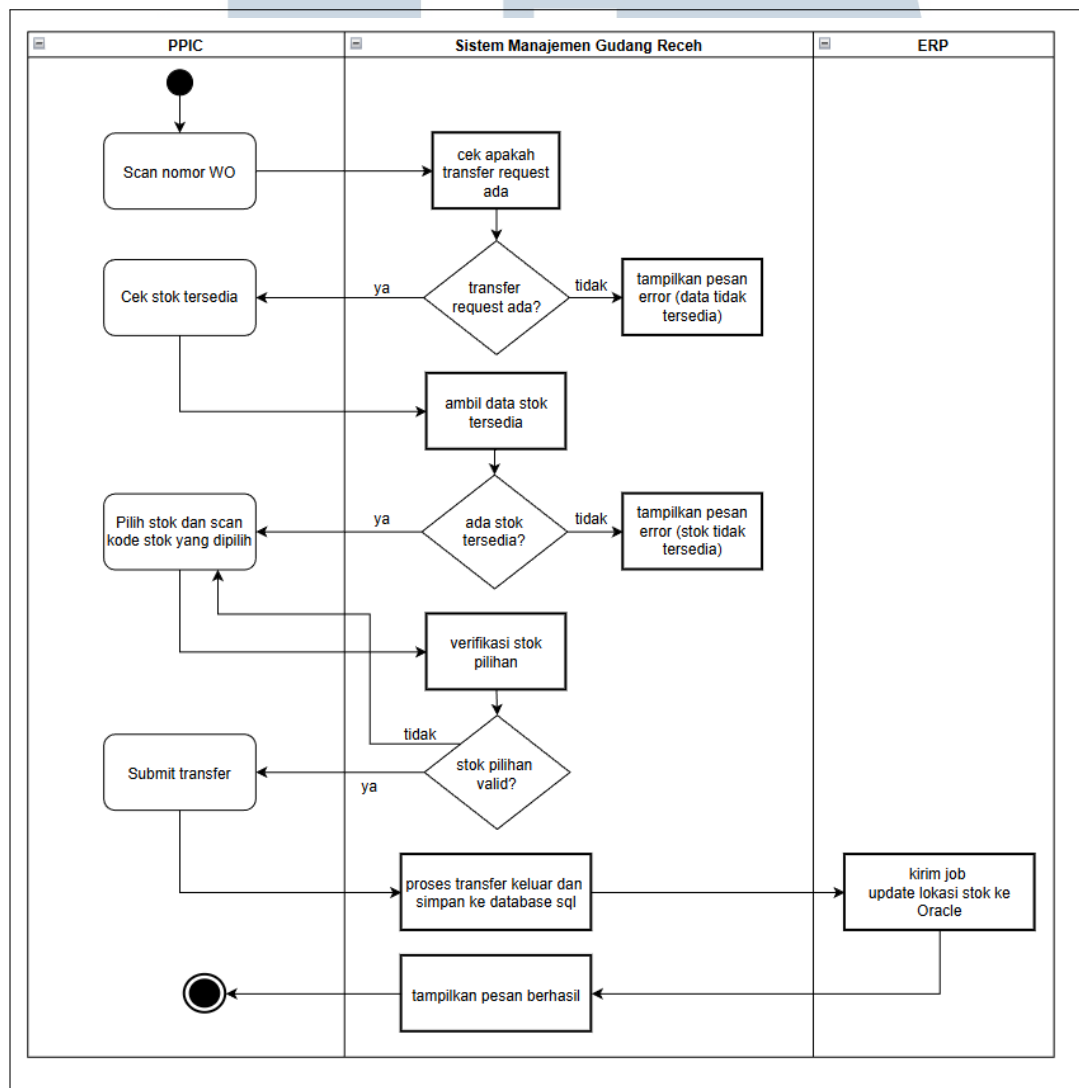


Gambar 3.3. Activity Diagram Proses Transfer Masuk Sistem Manajemen Gudang Receh

Gambar 3.3 menunjukkan *Activity Diagram* proses pencatatan transfer masuk. Proses dimulai dengan PPIC yang akan memindai *barcode/QR code* untuk mendapatkan nomor *Work Order* (WO). Kemudian sistem akan melakukan pengecekan untuk melihat apakah ada permintaan transfer masuk untuk WO tersebut. Jika ada, PPIC akan diminta

untuk memilih lokasi penyimpanan (rak) dan melakukan pemindaian kode rak untuk verifikasi bahwa lokasi yang dipilih sesuai. Setelah rak berhasil diverifikasi, PPIC dapat melakukan klik *submit* dan sistem akan memproses permintaan transfer masuk dengan mengubah status permintaan transfer, membuat data stok, membuat *log* transfer, dan mengirim *job* untuk melakukan *update* lokasi stok di basis data Oracle.

B Pencatatan Transfer Keluar



Gambar 3.4. Activity Diagram Proses Transfer Keluar Sistem Manajemen Gudang Receh

Gambar 3.4 menampilkan Activity Diagram proses pencatatan transfer keluar. Sama seperti proses pencatatan transfer masuk, PPIC akan memindai *barcode/QR code* untuk mendapatkan nomor *Work Order* (WO). Selanjutnya, sistem akan melakukan pengecekan

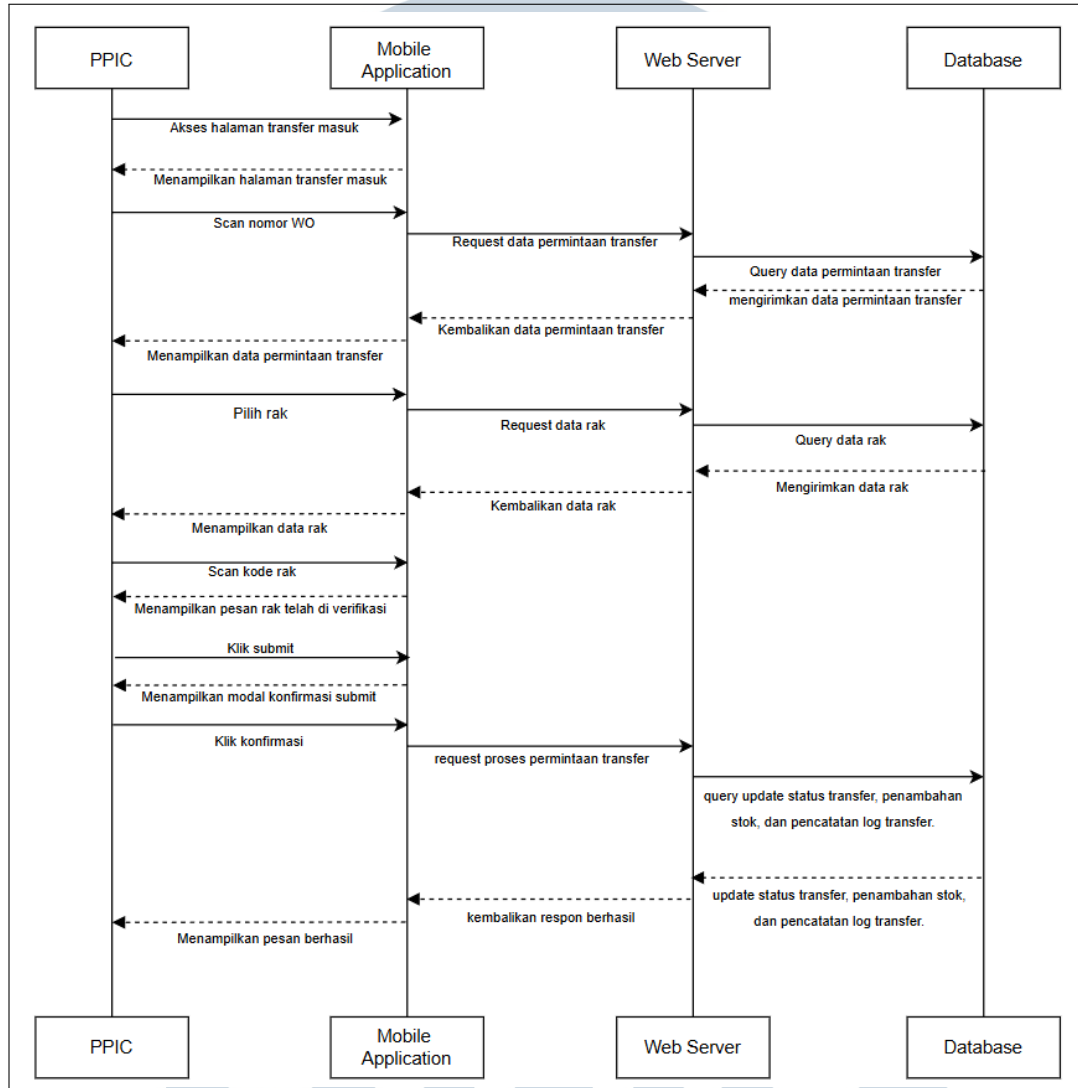
untuk melihat apakah terdapat permintaan transfer keluar untuk WO tersebut. Jika ada, PPIC akan menekan tombol ambil stok tersedia untuk melihat apakah ada stok tersedia untuk produk tersebut. Jika stok tersedia, PPIC akan memilih stok yang akan dikeluarkan kemudian memindai kode stok, yang merupakan kombinasi nomor WO produk dan kode rak untuk verifikasi bahwa stok yang dipilih sesuai. Setelah stok berhasil diverifikasi, PPIC dapat mengklik tombol *submit* dan sistem akan memproses permintaan transfer keluar dengan mengubah status permintaan transfer, mengurangi data stok, membuat *log* transfer, dan mengirim *job* untuk melakukan *update* lokasi stok di basis data Oracle.

3.5.4 *Sequence Diagram*

Sequence Diagram pada Sistem Manajemen Gudang Receh digunakan untuk memodelkan interaksi antar objek dalam sistem secara berurutan berdasarkan waktu. Diagram ini memberikan gambaran mengenai aliran permintaan, dan respon yang didapat.



A Pencatatan Transfer Masuk



Gambar 3.5. *Sequence Diagram* Proses Transfer Masuk Sistem Manajemen Gudang Receh

Gambar 3.5 menunjukkan *Sequence Diagram* proses transfer masuk yang menggambarkan alur interaksi antara PPIC dan Sistem Manajemen Gudang Receh dalam proses penerimaan barang masuk. Proses ini diawali dengan PPIC mengakses halaman transfer masuk melalui aplikasi *mobile*, kemudian sistem menampilkan halaman transfer masuk.

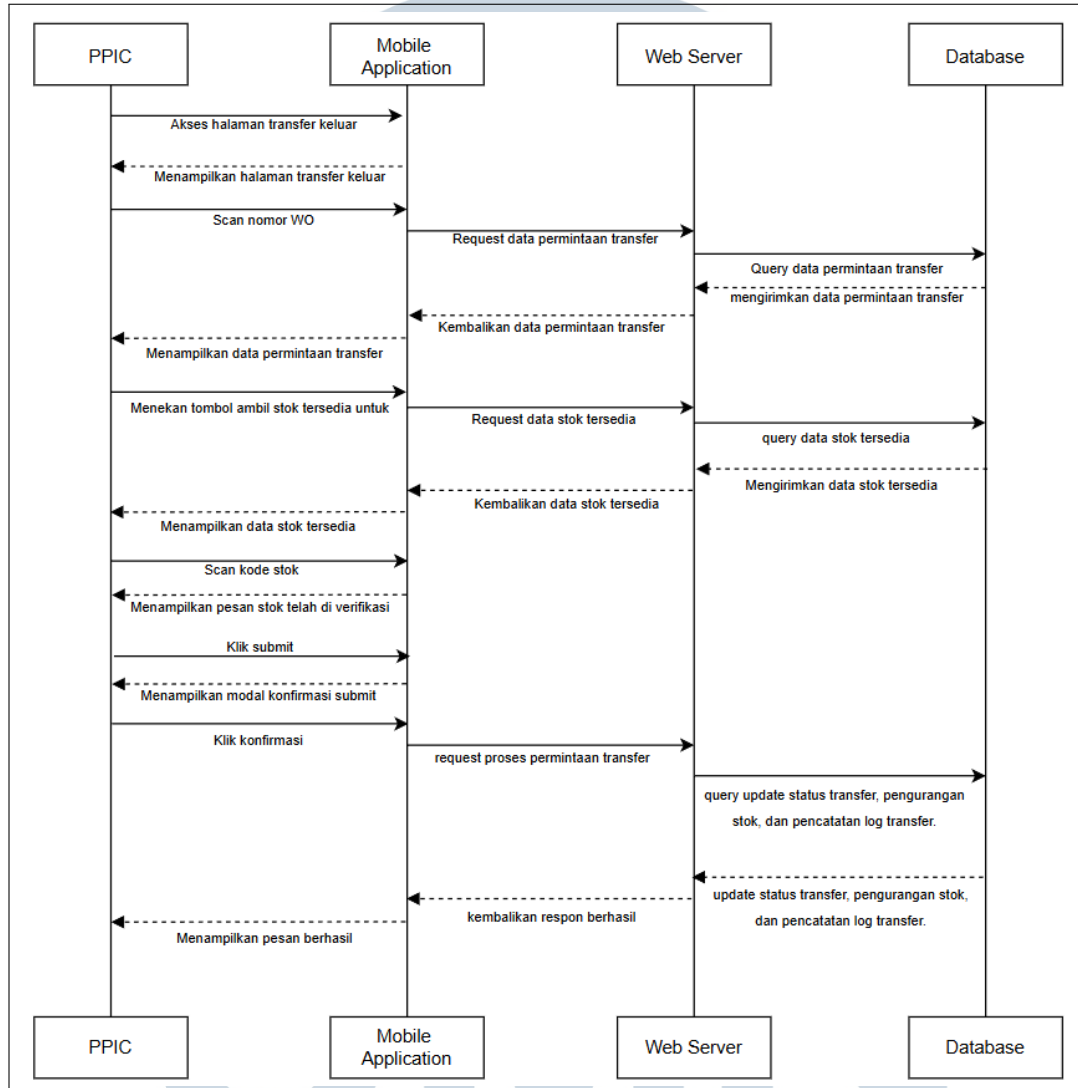
Kemudian, PPIC melakukan pemindaian nomor *Work Order* (WO) yang kemudian diteruskan oleh aplikasi ke *Web Server* untuk mendapatkan data permintaan transfer. *Web Server* akan melakukan query ke basis data dan mengembalikan data permintaan transfer untuk ditampilkan kepada PPIC.

Setelah itu, PPIC memilih rak penyimpanan yang akan digunakan untuk menyimpan barang. Aplikasi akan mengirim permintaan ke *Web Server* untuk mendapatkan data rak yang dipilih. *Web Server* mengambil data dari basis data dan mengembalikannya ke aplikasi untuk ditampilkan. Lalu, PPIC melakukan pemindaian kode rak sebagai proses verifikasi lokasi penyimpanan.

Selanjutnya, PPIC melakukan konfirmasi transfer masuk dengan menekan tombol *submit*. Aplikasi menampilkan modal konfirmasi *submit* kepada PPIC, kemudian PPIC menekan tombol konfirmasi. Aplikasi mengirimkan permintaan proses transfer ke *Web Server* yang akan melakukan *query* untuk memperbarui status permintaan transfer, membuat data stok barang, dan mencatat *log* transfer pada basis data. Setelah proses berhasil, *Web Server* akan mengirimkan respon berhasil ke aplikasi dan pesan berhasil akan ditampilkan kepada PPIC.



B Pencatatan Transfer Keluar



Gambar 3.6. *Sequence Diagram* Transfer Keluar Sistem Manajemen Gudang Receh

Gambar 3.6 menunjukkan *Sequence Diagram* proses transfer keluar yang menggambarkan alur interaksi antara PPIC dan Sistem Manajemen Gudang Receh dalam proses pengeluaran barang. Proses ini diawali dengan PPIC mengakses halaman transfer keluar melalui aplikasi *mobile*, kemudian sistem menampilkan halaman transfer keluar.

Kemudian, PPIC melakukan pemindaian nomor *Work Order* (WO) yang kemudian diteruskan oleh aplikasi ke *Web Server* untuk mendapatkan data permintaan transfer. *Web Server* akan melakukan *query* ke basis data dan mengembalikan data permintaan transfer untuk ditampilkan kepada PPIC.

Setelah itu, PPIC menekan tombol ambil stok tersedia untuk memilih stok yang

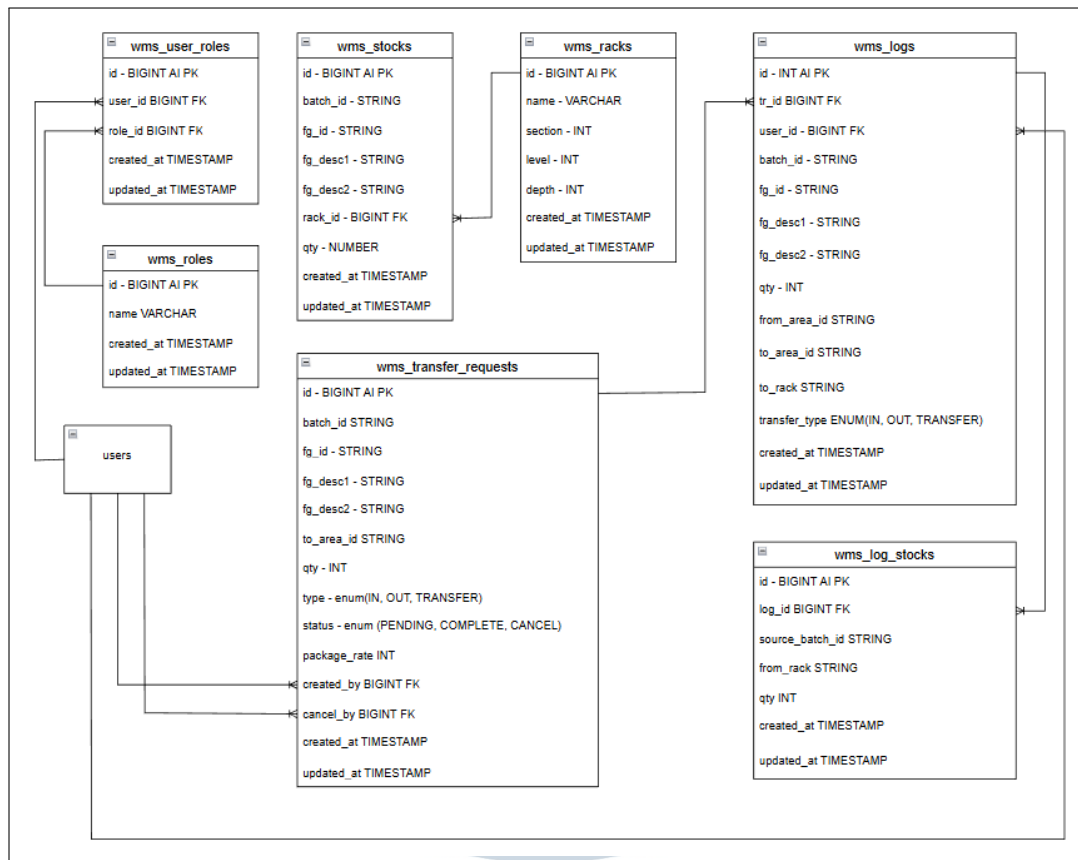
akan dikeluarkan. Aplikasi akan mengirim permintaan ke *Web Server* untuk mendapatkan data stok tersedia. *Web Server* mengambil data dari basis data dan mengembalikannya ke aplikasi untuk ditampilkan. Lalu, PPIC melakukan pemindaian kode stok sebagai proses verifikasi stok yang akan dikeluarkan.

Selanjutnya, PPIC melakukan konfirmasi transfer keluar dengan menekan tombol *submit*. Aplikasi menampilkan modal konfirmasi *submit* kepada PPIC, kemudian PPIC menekan tombol konfirmasi. Aplikasi mengirimkan permintaan proses permintaan transfer ke *Web Server* yang akan melakukan *query* untuk memperbarui status transfer, mengurangi stok barang, dan mencatat *log* transfer pada basis data. Setelah proses berhasil, *Web Server* akan mengirimkan respons berhasil ke aplikasi dan pesan berhasil akan ditampilkan kepada PPIC.

3.5.5 Struktur Tabel

Struktur tabel pada sistem manajemen gudang recek terdiri dari sejumlah entitas tabel yang berfungsi mendukung seluruh alur proses transaksi dalam sistem. Setiap tabel dirancang dengan atribut-atribut berdasarkan kebutuhan fungsional guna menyajikan informasi secara akurat. Keterkaitan antar-tabel tersebut direpresentasikan melalui *Physical Entity Relationship Diagram* (ERD) sebagaimana ditampilkan pada Gambar 3.7, yang selanjutnya diuraikan melalui deskripsi detail untuk setiap tabel.





Gambar 3.7. *Physical Entity Relationship Diagram (ERD)* Sistem Manajemen Gudang Receh

A Tabel **wms_roles**

Tabel 3.4 mendeskripsikan struktur tabel **wms_roles**, yang digunakan untuk menyimpan data peran yang dapat dimiliki oleh pengguna dalam Sistem Manajemen Gudang Receh.

Tabel 3.4. Struktur tabel **wms_roles**

Nama Kolom	Type Data	Constraint
id	BIGINT	PRIMARY KEY, AUTO INCREMENT
name	VARCHAR	
created_at	TIMESTAMP	DEFAULT CURRENT TIMESTAMP
updated_at	TIMESTAMP	ON UPDATE CURRENT TIMESTAMP

Penjelasan atribut tabel **wms_roles**:

1. **id**
Berfungsi sebagai *primary key* untuk setiap data *role*. Kolom diatur dengan *auto increment* agar dapat menghasilkan *id* secara otomatis.
2. **name**
Menyimpan nama peran pengguna yang terdapat dalam sistem. atribut ini digunakan untuk mengidentifikasi jenis peran yang dapat dimiliki oleh user dalam sistem.
3. **created_at**
Mencatat waktu pembuatan data peran. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis saat data dibuat.
4. **updated_at**
Mencatat waktu terakhir kali data peran diperbarui. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis setiap kali terjadi perubahan data.

B Tabel **wms_user_roles**

Tabel 3.5 mendeskripsikan struktur tabel **wms_user_roles**, yang digunakan sebagai tabel pivot relasi *many to many*, untuk menyimpan peran yang dimiliki pengguna di dalam sistem.

Tabel 3.5. Struktur tabel **wms_user_roles**

Nama Kolom	Tipe Data	Constraint
id	BIGINT	PRIMARY KEY, AUTO INCREMENT
user_id	BIGINT	FOREIGN KEY
role_id	BIGINT	FOREIGN KEY
created_at	TIMESTAMP	DEFAULT CURRENT TIMESTAMP
updated_at	TIMESTAMP	ON UPDATE CURRENT TIMESTAMP

Penjelasan atribut tabel **wms_user_roles**:

1. **id**
Berfungsi sebagai *primary key* untuk setiap data peran pengguna. Kolom diatur dengan *auto increment* agar dapat menghasilkan *id* secara otomatis.

2. `user_id`

Foreign key yang terhubung dengan tabel `users`, berfungsi untuk mereferensikan pengguna yang memiliki peran di sistem.

3. `role_id`

Foreign key yang terhubung dengan tabel `wms_roles`, berfungsi untuk mereferensikan peran yang dimiliki pengguna di sistem.

4. `created_at`

Mencatat waktu pembuatan data. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis saat data dibuat.

5. `updated_at`

Mencatat waktu terakhir kali data diperbarui. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis setiap kali terjadi perubahan data.

C Tabel `wms_racks`

Tabel 3.6 mendeskripsikan struktur tabel `wms_racks`, yang digunakan untuk menyimpan data lokasi penyimpanan di dalam sistem.

Tabel 3.6. Struktur tabel `wms_racks`

Nama Kolom	Tipe Data	Constraint
<code>id</code>	BIGINT	PRIMARY KEY, AUTO INCREMENT
<code>name</code>	VARCHAR	
<code>section</code>	INT	
<code>level</code>	INT	
<code>depth</code>	INT	
<code>created_at</code>	TIMESTAMP	DEFAULT CURRENT TIMESTAMP
<code>updated_at</code>	TIMESTAMP	ON UPDATE CURRENT TIMESTAMP

Penjelasan atribut tabel `wms_racks`:

1. `id`

Berfungsi sebagai *primary key* untuk setiap data lokasi penyimpanan (rak). Kolom diatur dengan *auto increment* agar dapat menghasilkan `id` secara otomatis.

2. `name`

Menyimpan informasi nama atau kode dari data rak dengan konvensi penamaan yang telah disetujui oleh pengguna.

3. `section`

Menyimpan informasi *section* dari rak sebagai komponen koordinat lokasi penyimpanan dalam gudang.

4. `level`

Menyimpan informasi tingkat atau dari rak sebagai komponen koordinat vertikal lokasi penyimpanan dalam gudang.

5. `depth`

Menyimpan informasi kedalaman dari rak sebagai komponen koordinat horizontal lokasi penyimpanan dalam gudang.

6. `created_at`

Mencatat waktu pembuatan data rak. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis saat data dibuat.

7. `updated_at`

Mencatat waktu terakhir kali data rak diperbarui. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis setiap kali terjadi perubahan data.

D Tabel `wms_transfer_requests`

Tabel 3.7 mendeskripsikan struktur tabel `wms_transfer_requests`, yang digunakan untuk menyimpan data permintaan transfer di dalam sistem.

Tabel 3.7. Struktur tabel `wms_transfer_requests`

Nama Kolom	Type Data	Constraint
id	BIGINT	PRIMARY KEY, AUTO INCREMENT
batch_id	VARCHAR	
fg_id	VARCHAR	
fg_desc_1	VARCHAR	
fg_desc_2	VARCHAR	
to_area_id	VARCHAR	
qty	INT	
type	ENUM (IN, OUT, TRANSFER)	
status	ENUM (PENDING, COMPLETE, CANCEL)	DEFAULT PENDING
package_rate	INT	
created_by	BIGINT	FOREIGN KEY
cancel_by	BIGINT	FOREIGN KEY
created_at	TIMESTAMP	DEFAULT CURRENT TIMESTAMP
updated_at	TIMESTAMP	ON UPDATE CURRENT TIMESTAMP

Penjelasan atribut tabel `wms_transfer_requests`:

1. id
Befungsi sebagai *primary key* untuk setiap data permintaan transfer. Kolom diatur dengan *auto increment* agar dapat menghasilkan id secara otomatis.
2. batch_id
Menyimpan nomor *batch/Work Order* (WO) yang akan melakukan proses transfer.
3. fg_id
Menyimpan informasi kode produk dari work order yang akan melakukan proses transfer.
4. fg_desc_1
Menyimpan informasi deskripsi utama produk.
5. fg_desc_2
Menyimpan informasi deskripsi tambahan produk.

6. `to_area_id`
Menyimpan area tujuan permintaan transfer, jika transfer masuk tujuan adalah ke area gudang receh, jika transfer keluar tujuan adalah ke area produksi.
7. `qty`
Menyimpan informasi jumlah produk yang akan di transfer, hanya diisi saat pembuatan permintaan transfer masuk.
8. `type`
Menyimpan informasi tipe permintaan transfer, bernilai ENUM dengan opsi IN, OUT, dan TRANSFER.
9. `status`
Menyimpan informasi status permintaan transfer, bernilai ENUM dengan opsi PENDING, COMPLETE, dan CANCEL.
10. `package_rate`
Menyimpan informasi kapasitas per dus setiap produk.
11. `created_by`
foreign key yang terhubung dengan tabel `users`, berguna untuk mereferensikan pengguna yang membuat permintaan transfer.
12. `cancel_by`
foreign key yang terhubung dengan tabel `users`, berguna untuk mereferensikan pengguna yang membatalkan permintaan transfer.
13. `created_at`
Mencatat waktu pembuatan data permintaan transfer. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis saat data dibuat.
14. `updated_at`
Mencatat waktu terakhir kali data permintaan transfer diperbarui. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis setiap kali terjadi perubahan data.

E Tabel `wms_stocks`

Tabel 3.8 mendeskripsikan struktur tabel `wms_stocks`, yang digunakan untuk menyimpan data stok di dalam sistem.

Tabel 3.8. Struktur tabel `wms_stocks`

Nama Kolom	Tipe Data	Constraint
<code>id</code>	BIGINT	PRIMARY KEY, AUTO INCREMENT
<code>batch_id</code>	VARCHAR	
<code>fg_id</code>	VARCHAR	
<code>fg_desc_1</code>	VARCHAR	
<code>fg_desc_2</code>	VARCHAR	
<code>rack_id</code>	BIGINT	
<code>qty</code>	INT	
<code>created_at</code>	TIMESTAMP	DEFAULT CURRENT TIMESTAMP
<code>updated_at</code>	TIMESTAMP	ON UPDATE CURRENT TIMESTAMP

Penjelasan atribut tabel `wms_stocks`:

1. `id`
Berfungsi sebagai *primary key* untuk setiap data stok. Kolom diatur dengan *auto increment* agar dapat menghasilkan `id` secara otomatis.
2. `batch_id`
Menyimpan kode *batch / Work Order (WO)* stok.
3. `fg_id`
Menyimpan informasi kode produk dari stok.
4. `fg_desc_1`
Menyimpan informasi deskripsi utama produk.
5. `fg_desc_2`
Menyimpan informasi deskripsi tambahan produk.
6. `rack_id`
foreign key yang terhubung dengan tabel `racks`, berfungsi mereferensikan lokasi penyimpanan stok.
7. `qty`
Menyimpan informasi jumlah produk yang tersimpan.

8. created_at

Mencatat waktu pembuatan data stok. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis saat data dibuat.

9. updated_at

Mencatat waktu terakhir kali data stok diperbarui. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis setiap kali terjadi perubahan data.

F Tabel **wms_logs**

Tabel 3.9 mendeskripsikan struktur tabel **wms_logs**, yang digunakan untuk menyimpan data riwayat transfer di dalam sistem.

Tabel 3.9. Struktur tabel **wms_logs**

Nama Kolom	Tipe Data	Constraint
id	BIGINT	PRIMARY KEY, AUTO INCREMENT
tr_id	BIGINT	FOREIGN KEY
user_id	BIGINT	FOREIGN KEY
batch_id	VARCHAR	
fg_id	VARCHAR	
fg_desc_1	VARCHAR	
fg_desc_2	VARCHAR	
qty	INT	
from_area_id	STRING	
to_area_id	STRING	
to_rack	STRING	
transfer_type	ENUM(IN, OUT, TRANSFER)	
created_at	TIMESTAMP	DEFAULT CURRENT TIMESTAMP
updated_at	TIMESTAMP	ON UPDATE CURRENT TIMESTAMP

Penjelasan atribut tabel **wms_logs:**

1. `id`
Berfungsi sebagai *primary key* untuk setiap data logs. Kolom diatur dengan *auto increment* agar dapat menghasilkan `id` secara otomatis.
2. `tr_id`
foreign key yang terhubung dengan tabel `wms_transfer_requests`, berguna untuk mereferensikan data dengan permintaan transfer terkait.
3. `user_id`
foreign key yang terhubung dengan tabel `users`, berfungsi mereferensikan pengguna yang melakukan pencatatan transfer.
4. `batch_id`
Menyimpan kode *batch / Work Order (WO)* dari permintaan transfer.
5. `fg_id`
Menyimpan informasi kode produk dari permintaan transfer.
6. `fg_desc_1`
Menyimpan informasi deskripsi utama produk.
7. `fg_desc_2`
Menyimpan informasi deskripsi tambahan produk.
8. `qty`
Menyimpan informasi jumlah produk yang tersimpan.
9. `from_area_id`
Menyimpan informasi area asal transfer.
10. `to_area_id`
Menyimpan informasi area asal tujuan transfer
11. `to_rack`
Menyimpan informasi nama rak tempat stok disimpan, atribut ini hanya digunakan saat transfer masuk.
12. `transfer_type`
Menyimpan informasi tipe permintaan transfer, bernilai ENUM dengan pilihan `IN`, `OUT`, dan `TRANSFER`.
13. `created_at`
Mencatat waktu pembuatan data. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis saat data dibuat.

14. `updated_at`

Mencatat waktu terakhir kali data diperbarui. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis setiap kali terjadi perubahan data.

G Tabel `wms_log_stocks`

Tabel 3.10 mendeskripsikan struktur tabel `wms_log_stocks`, yang digunakan sebagai tabel *pivot* untuk menyimpan informasi stok yang dipilih saat proses transfer keluar.

Tabel 3.10. Struktur tabel `wms_log_stocks`

Nama Kolom	Tipe Data	Constraint
<code>id</code>	BIGINT	PRIMARY KEY, AUTO INCREMENT
<code>log_id</code>	BIGINT	FOREIGN KEY
<code>source_batch_id</code>	VARCHAR	
<code>from_rack</code>	VARCHAR	
<code>qty</code>	INT	
<code>created_at</code>	TIMESTAMP	DEFAULT CURRENT TIMESTAMP
<code>updated_at</code>	TIMESTAMP	ON UPDATE CURRENT TIMESTAMP

Penjelasan atribut tabel `wms_log_stocks`:

1. `id`
Berfungsi sebagai *primary key* untuk setiap data. Kolom diatur dengan *auto increment* agar dapat menghasilkan `id` secara otomatis.
2. `log_id`
foreign key yang terhubung dengan tabel `wms_logs`, digunakan untuk mereferensikan data dengan riwayat transfer keluar.
3. `source_batch_id`
Menyimpan kode *batch / Work Order (WO)* stok yang diambil dari gudang untuk transfer keluar.
4. `from_rack`
Menyimpan informasi rak asal stok yang diambil untuk transfer keluar.

5. qty

Menyimpan informasi jumlah produk yang diambil untuk *batch* terkait saat transfer keluar

6. created_at

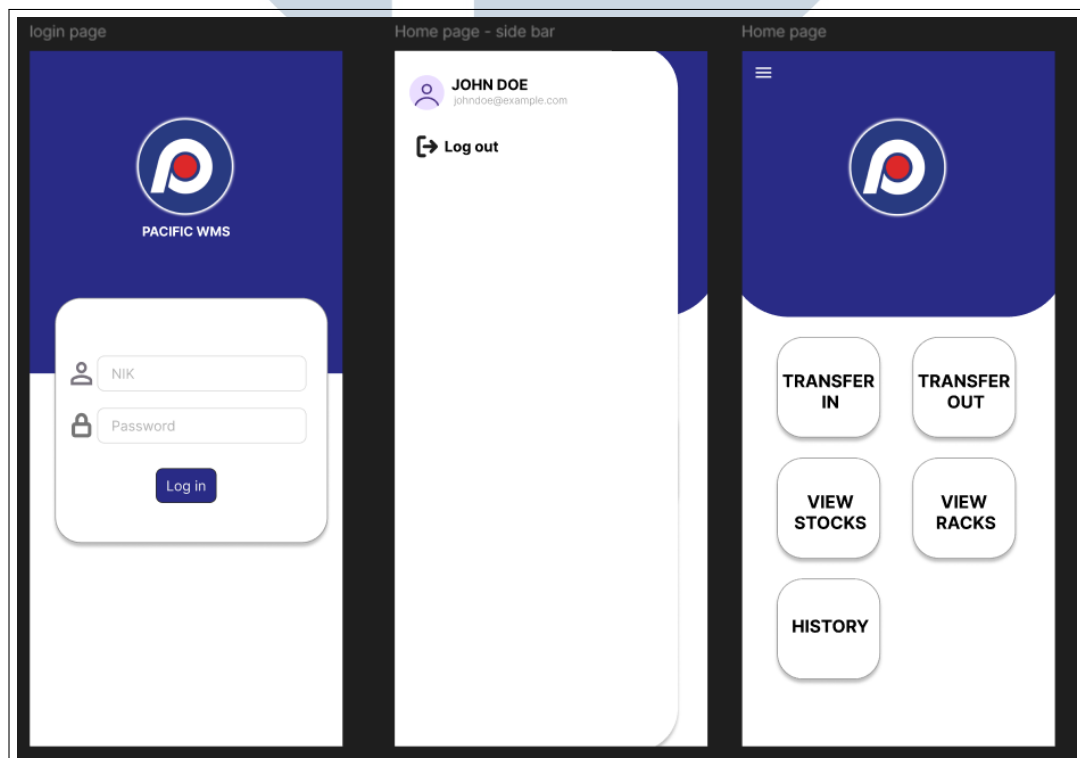
Mencatat waktu pembuatan data. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis saat data dibuat.

7. updated_at

Mencatat waktu terakhir kali data diperbarui. Kolom ini diatur untuk memiliki nilai *current timestamp* secara otomatis setiap kali terjadi perubahan data.

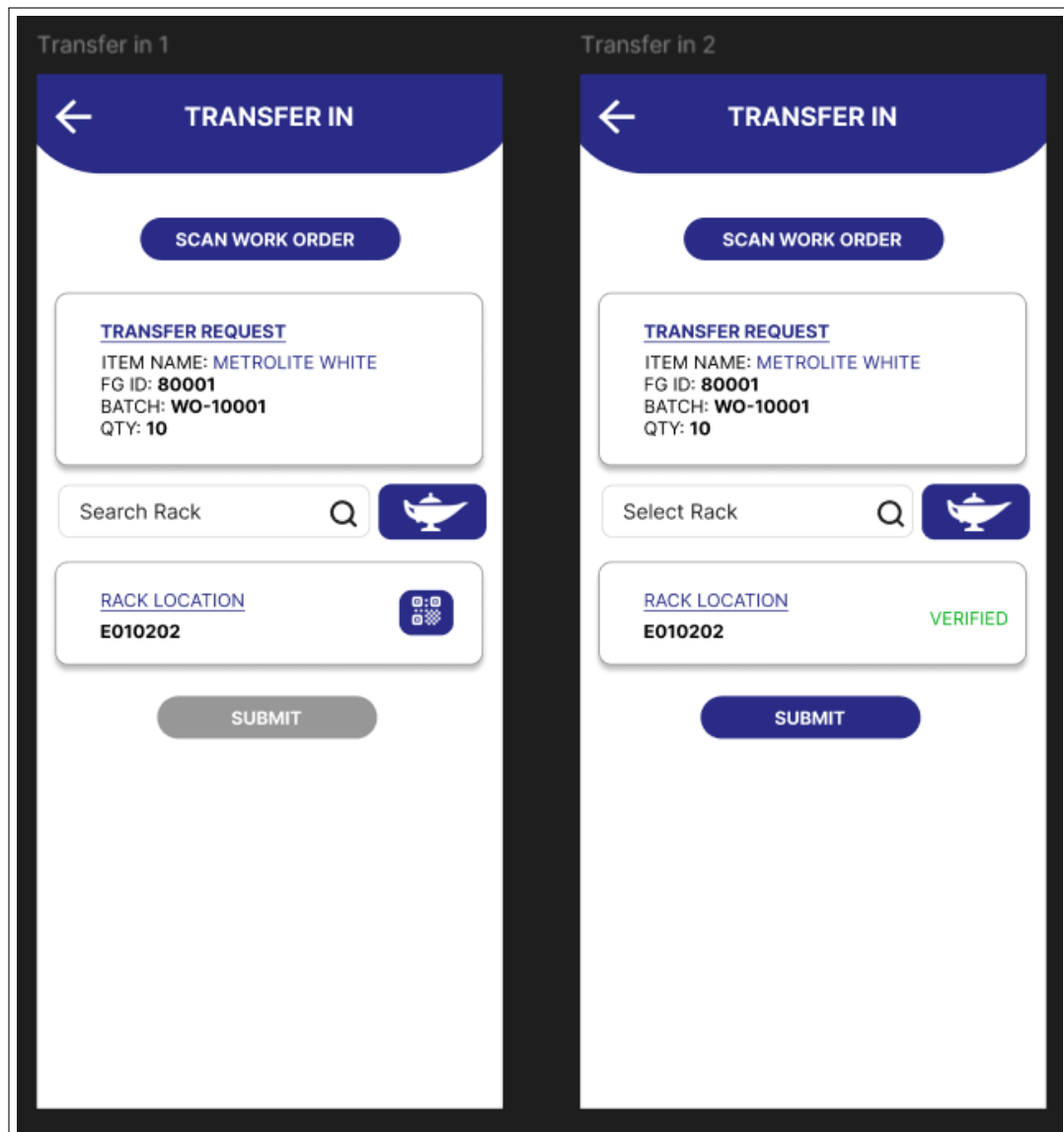
3.5.6 Mockup Antarmuka Pengguna

Bagian ini menampilkan *mockup* antarmuka pengguna Sistem Manajemen Gudang Receh untuk tampilan aplikasi *mobile*. *Mockup* dirancang untuk memberikan gambaran visual mengenai tata letak, navigasi, dan interaksi pengguna dalam menggunakan aplikasi.



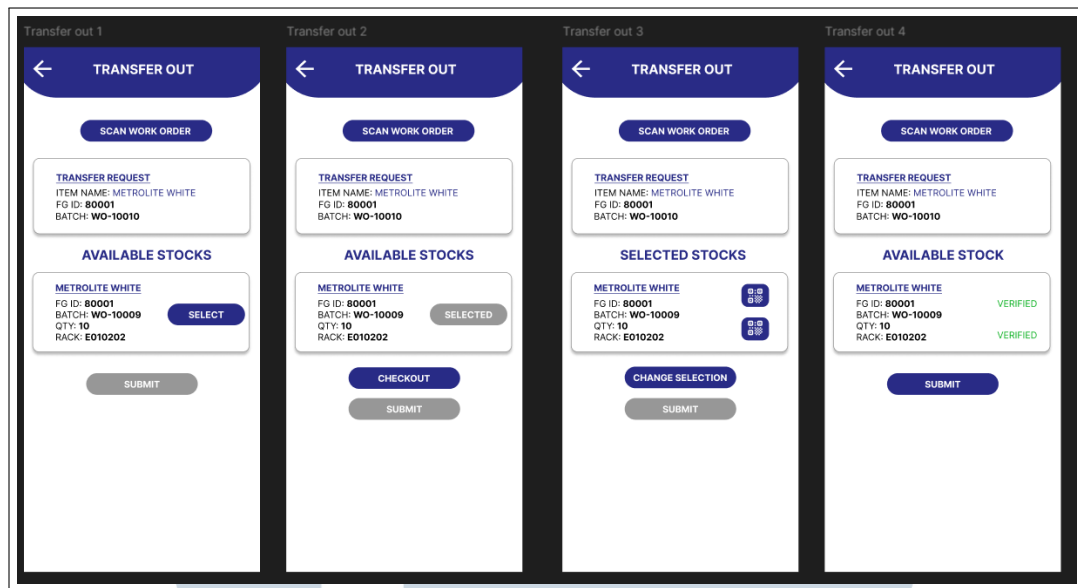
Gambar 3.8. Mockup Halaman Login dan Halaman Utama Aplikasi Mobile

Gambar 3.8 adalah *mockup* dari halaman *login*, dan halaman utama aplikasi *mobile* Sistem Manajemen Gudang Receh.



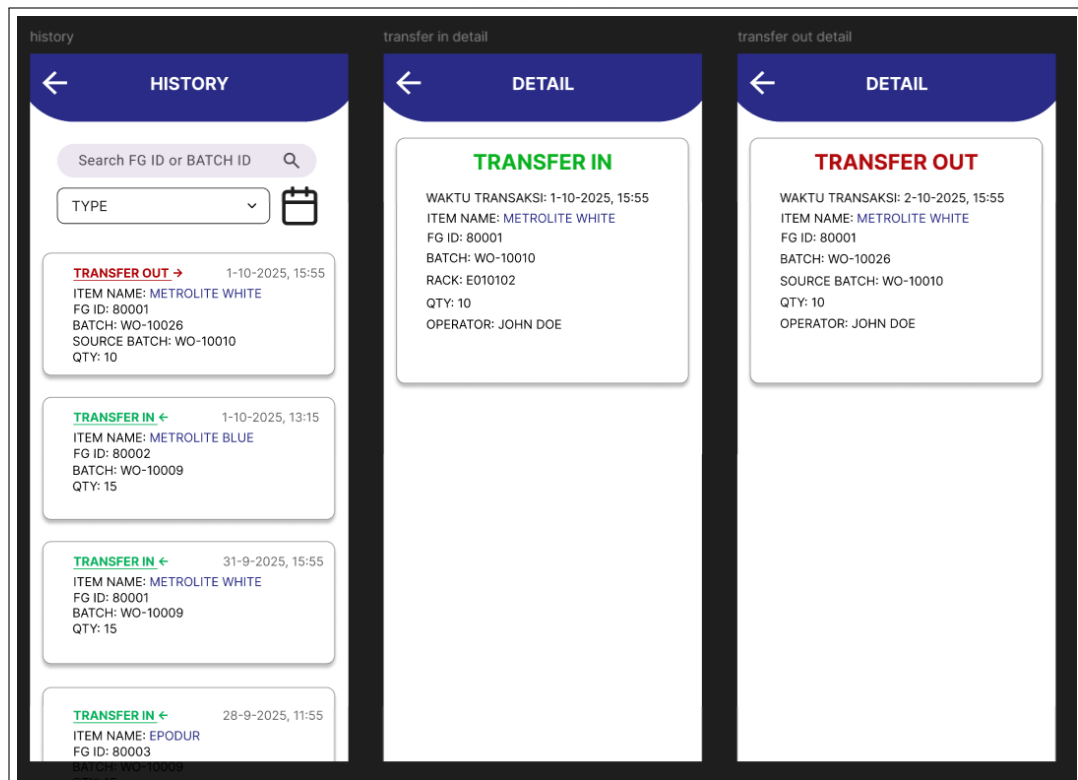
Gambar 3.9. Mockup Halaman Transfer Masuk Aplikasi *Mobile*

Gambar 3.9 adalah *mockup* dari halaman transfer masuk pada aplikasi *mobile* Sistem Manajemen Gudang Recch, yang menggambarkan kondisi saat pengguna melakukan verifikasi lokasi penyimpanan.



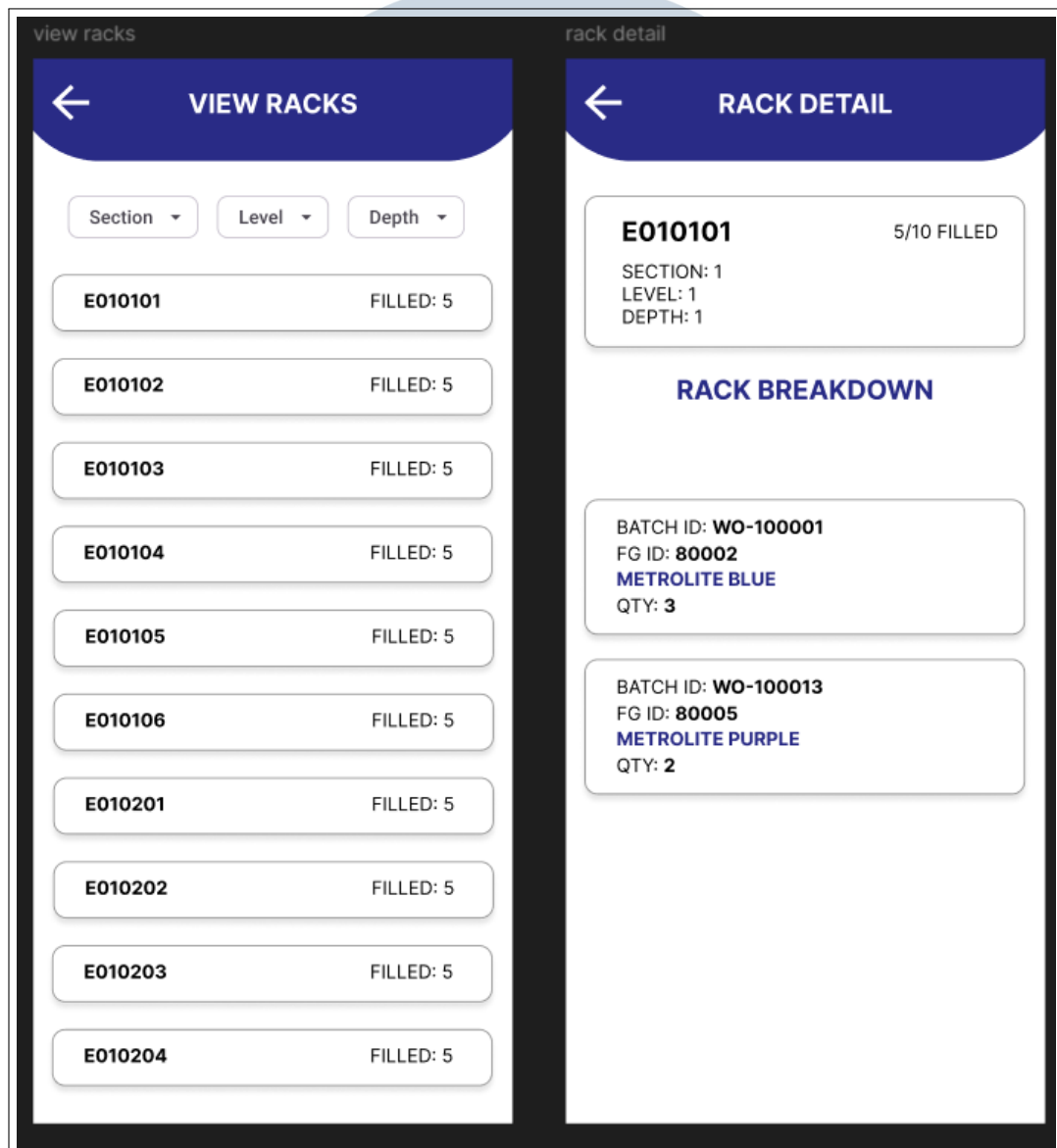
Gambar 3.10. Mockup Halaman Transfer Keluar Aplikasi *Mobile*

Gambar 3.10 adalah *mockup* dari halaman transfer keluar pada aplikasi *mobile* Sistem Manajemen Gudang Receh, yang menggambarkan kondisi saat pengguna memilih stok dan saat pengguna melakukan verifikasi stok.



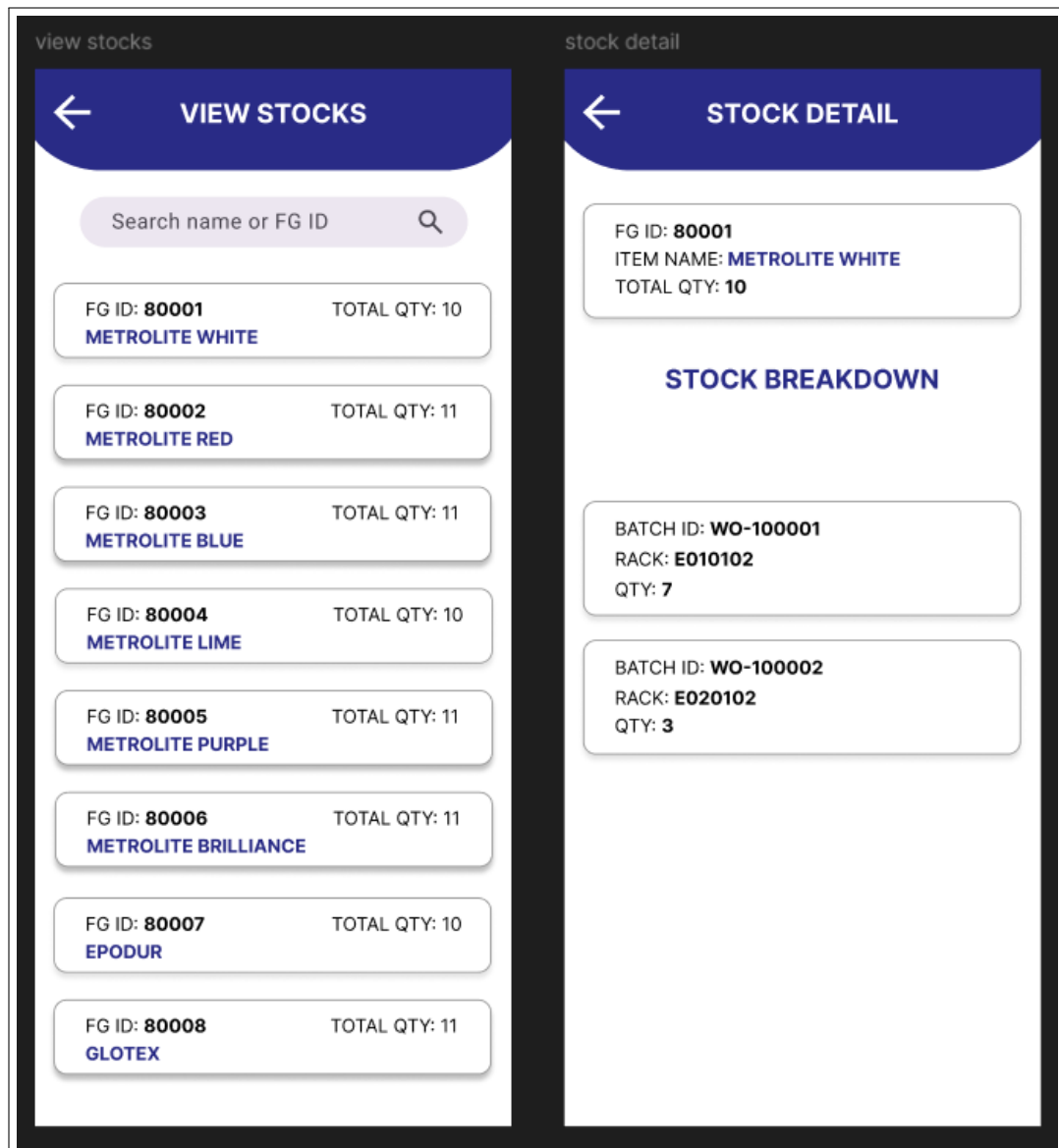
Gambar 3.11. Mockup Halaman Riwayat Transfer Aplikasi *Mobile*

Gambar 3.11 adalah *mockup* dari halaman daftar riwayat transfer beserta detail riwayat transfer pada aplikasi *mobile* Sistem Manajemen Gudang Receh.



Gambar 3.12. Mockup Halaman Rak Aplikasi *Mobile*

Gambar 3.12 adalah *mockup* dari halaman daftar rak beserta detail rincian isi rak pada aplikasi *mobile* Sistem Manajemen Gudang Receh.



Gambar 3.13. Mockup Halaman Stok Aplikasi *Mobile*

Gambar 3.13 adalah *mockup* dari halaman daftar stok beserta detail rincian stok pada aplikasi *mobile* Sistem Manajemen Gudang Receh.

3.5.7 Hasil Implementasi

A *Login*



Gambar 3.14. Halaman *Login* Aplikasi *Mobile* Sistem Manajemen Gudang Receh

Gambar 3.14 menunjukkan tampilan halaman *login* aplikasi *mobile* Sistem Manajemen Gudang Receh. Halaman ini menjadi halaman yang dilihat pertama kali oleh pengguna saat membuka aplikasi.

B Transfer Masuk

← TRANSFER IN

SCAN QR CODE

TRANSFER REQUEST

ITEM DESC 1:
5200.00000-3 LTR

ITEM DESC 2:
METROLITE PUTIH

FG ID:
80005813

BATCH:
WQ25120001

QTY:
3

Input Rack Code
E010101

RACK LOCATION
E010101

SUBMIT

Gambar 3.15. Tampilan Halaman Transfer Masuk Sebelum Verifikasi Lokasi Penyimpanan

Gambar 3.15 menunjukkan tampilan awal halaman transfer masuk pada aplikasi *mobile* Sistem Manajemen Gudang Receh. Pada bagian atas halaman, terdapat tombol yang berfungsi untuk memindai kode QR atau *barcode* menggunakan kamera guna mendapatkan nomor *Work Order* (WO). Apabila permintaan transfer ditemukan, sistem akan menampilkan data permintaan transfer tersebut. Selanjutnya, akan muncul *search bar* dan tombol rekomendasi yang dapat digunakan untuk memilih lokasi penyimpanan (rak). Setelah rak dipilih, sistem akan menampilkan data rak atau lokasi beserta tombol di sebelahnya untuk memindai kode QR rak menggunakan kamera sebagai proses verifikasi rak.

← TRANSFER IN

SCAN QR CODE

TRANSFER REQUEST

ITEM DESC 1:
5200.00000-3 LTR

ITEM DESC 2:
METROLITE PUTIH

FG ID:
80005813

BATCH:
W025120001

QTY:
3

Input Rack Code
E010101

RACK LOCATION
E010101

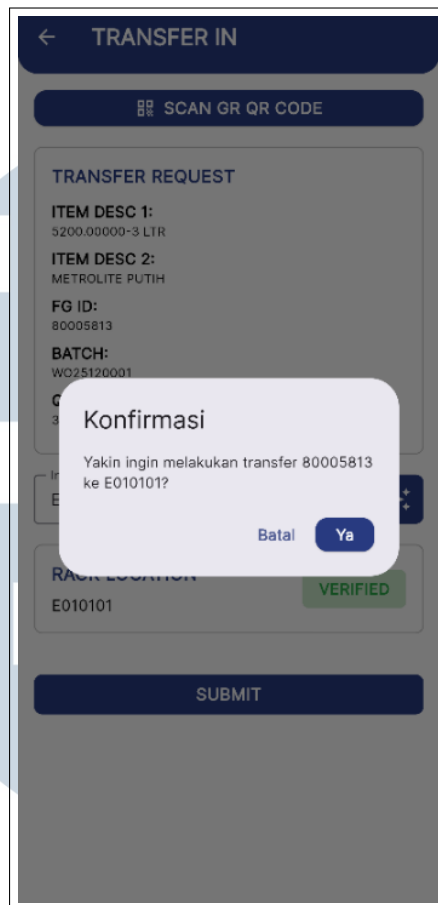
VERIFIED

SUBMIT

Gambar 3.16. Tampilan Halaman Transfer Masuk Setelah Verifikasi Lokasi Penyimpanan

Setelah proses verifikasi berhasil dilakukan, tampilan tombol akan berubah menjadi indikator yang menunjukkan bahwa lokasi penyimpanan telah terverifikasi, seperti yang dapat dilihat pada Gambar 3.16. Pada tahap ini, pengguna dapat melakukan *submit* dengan menekan tombol *submit* yang berubah menjadi aktif.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.17. Tampilan Konfirmasi Proses Transfer Masuk

Aplikasi kemudian akan menampilkan modal konfirmasi untuk memastikan bahwa pengguna ingin melakukan transfer masuk, seperti yang ditunjukkan pada Gambar 3.17.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.18. Pesan Berhasil Proses Transfer Masuk

Setelah pengguna melakukan konfirmasi dan proses transfer berhasil, aplikasi akan menampilkan pesan bahwa transfer masuk telah berhasil dilakukan, seperti yang terlihat pada Gambar 3.18.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

C Transfer Keluar

← TRANSFER OUT

📷 SCAN WO BARCODE

TRANSFER REQUEST

ITEM DESC 1:
5200.00000-3 LTR

ITEM DESC 2:
METROLITE PUTIH

FG ID:
80005813

BATCH:
WO25120002

GET AVAILABLE STOCKS

5200.00000-3 LTR

ITEM DESC 2:
METROLITE PUTIH

FG ID:
80005813

BATCH:
WO25120001

QUANTITY:
3

RACK:
E010101

SELECTED

PROCEED TO SELECTED STOCKS

Gambar 3.19. Tampilan Halaman Proses Transfer Keluar Setelah Memilih Stok Tersedia

Gambar 3.19 menunjukkan tampilan halaman transfer keluar pada aplikasi *mobile* Sistem Manajemen Gudang Receh. Pada bagian atas halaman, layaknya halaman transfer masuk, terdapat tombol yang berfungsi untuk memindai kode QR atau *barcode* guna mendapatkan nomor *Work Order* (WO). Apabila data permintaan transfer keluar ditemukan, sistem akan menampilkan data permintaan transfer terkait. Selanjutnya, terdapat tombol untuk mengambil stok tersedia yang apabila diklik akan menampilkan daftar stok tersedia untuk produk terkait. Daftar stok tersebut memuat informasi jumlah, kode *batch*, dan lokasi penyimpanan (rak). Pengguna dapat memilih stok yang akan dikeluarkan dengan menekan tombol *select* pada stok yang tersedia.

← **TRANSFER OUT**

📷 **SCAN WO BARCODE**

TRANSFER REQUEST

ITEM DESC 1:
5200.00000-3 LTR

ITEM DESC 2:
METROLITE PUTIH

FG ID:
80005813

BATCH:
WO25120002

SELECTED STOCKS

5200.00000-3 LTR

ITEM DESC 2:
METROLITE PUTIH

FG ID:
80005813

BATCH:
WO25120001

QUANTITY:
3

RACK:
E010101

CHANGE SELECTION

SUBMIT

Gambar 3.20. Halaman Proses Transfer Keluar Sebelum Verifikasi Stok Pilihan

Setelah pengguna memilih stok, akan muncul tombol lanjut ke stok pilihan untuk melanjutkan proses transfer keluar. Kemudian, sistem akan menampilkan data stok yang dipilih beserta dua tombol untuk memindai kode QR menggunakan kamera, seperti yang terlihat pada Gambar 3.20. Kedua tombol ini berfungsi untuk melakukan verifikasi nomor WO dan kode rak dari stok yang dipilih.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

TRANSFER OUT

SCAN WO BARCODE

TRANSFER REQUEST

ITEM DESC 1:
5200.00000-3 LTR

ITEM DESC 2:
METROLITE PUTIH

FG ID:
80005813

BATCH:
WO25120002

SELECTED STOCKS

5200.00000-3 LTR	
ITEM DESC 2: METROLITE PUTIH	
FG ID: 80005813	VERIFIED
BATCH: WO25120001	
QUANTITY: 3	
RACK: E010101	VERIFIED

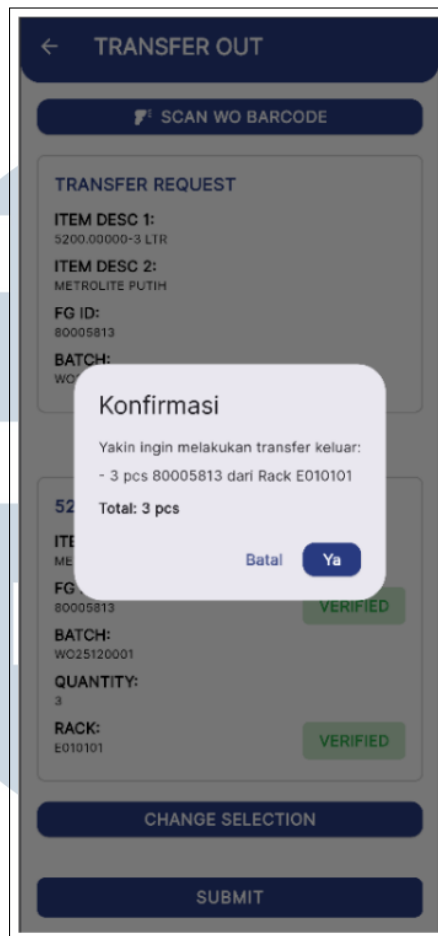
CHANGE SELECTION

SUBMIT

Gambar 3.21. Halaman Proses Transfer Keluar Setelah Verifikasi Stok Pilihan

Setelah proses verifikasi berhasil dilakukan, tampilan kedua tombol akan berubah menjadi indikator yang menunjukkan bahwa nomor WO dan kode rak stok pilihan telah diverifikasi, seperti yang ditampilkan pada Gambar 3.21. Pada tahap ini, pengguna dapat melakukan *submit* dengan menekan tombol *submit* yang telah aktif.

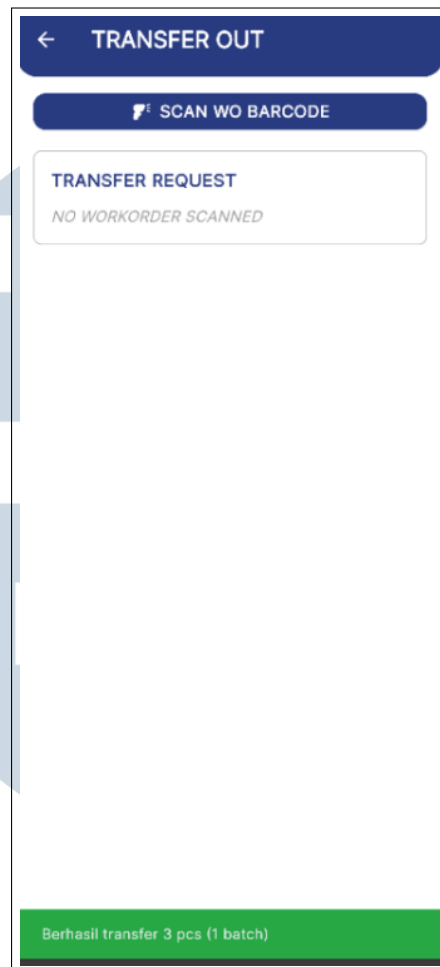
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.22. Tampilan Konfirmasi Proses Transfer Keluar

Aplikasi kemudian akan menampilkan modal konfirmasi untuk memastikan bahwa pengguna ingin melakukan transfer keluar, seperti yang diperlihatkan pada Gambar 3.22.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.23. Pesan Berhasil Proses Transfer Keluar

Setelah pengguna melakukan konfirmasi dan proses transfer berhasil, aplikasi akan menampilkan pesan bahwa transfer keluar telah berhasil dilakukan, seperti yang terlihat pada Gambar 3.23.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

D Create Permintaan Transfer

Tipe permintaan ☒ STOCK IN ☐ STOCK OUT

Tujuan ☐ GUDANG FG ☒ GUDANG RECEH

Nomor WO: W025120001 Check Item

Hasil Cek Workorder W025120001

FG ID	80005813
Deskripsi 1	5200.00000-3 LTR
Deskripsi 2	METROLITE PUTIH
Jumlah per box	4

Jika tujuan ke Gudang FG, jumlah harus merupakan kelipatan dari 4 (jumlah per box).

Jika tujuan ke Gudang Receh, jumlah harus kurang dari 4 (jumlah per box).

Jumlah: 3 Submit

Gambar 3.24. Halaman *Form Create* Permintaan Transfer Masuk

Tipe permintaan ☐ STOCK IN ☒ STOCK OUT

Tujuan ☐ GUDANG FG ☒ GUDANG RECEH

Nomor WO: W025120002 Check Item

Hasil Cek Workorder W025120002

FG ID	80005813
Deskripsi 1	5200.00000-3 LTR
Deskripsi 2	METROLITE PUTIH
Jumlah per box	4

Jika tujuan ke Gudang FG, jumlah harus merupakan kelipatan dari 4 (jumlah per box).

Jika tujuan ke Gudang Receh, jumlah harus kurang dari 4 (jumlah per box).

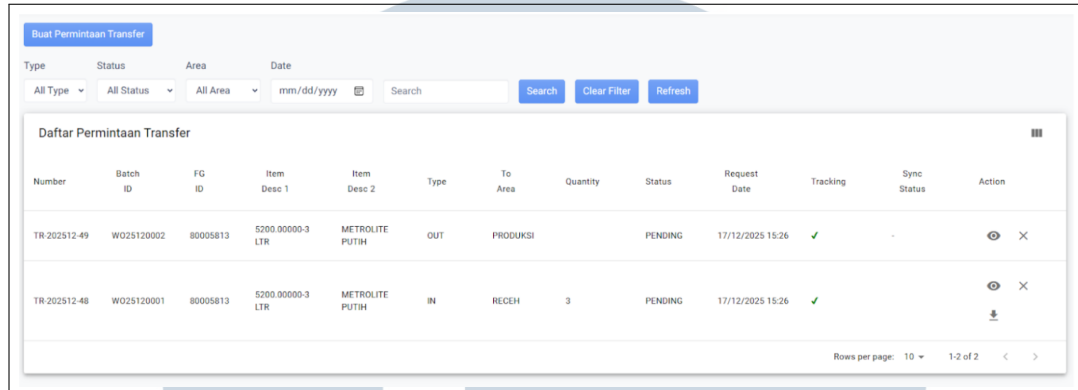
Jumlah: 3 Submit

Gambar 3.25. Halaman *Form Create* Permintaan Transfer Keluar

Gambar 3.24 dan Gambar 3.25 menampilkan halaman *form create* permintaan transfer pada Sistem Manajemen Gudang Receh. Halaman ini digunakan oleh pengguna untuk membuat permintaan transfer, baik transfer masuk maupun transfer keluar. Pengguna dapat memilih tipe permintaan yang diinginkan di bagian awal *form* untuk membedakan kedua jenis transfer tersebut.

Selanjutnya, pengguna dapat memasukkan nomor *Work Order* (WO) dan menekan tombol *check item* untuk memeriksa detail produk terkait. Apabila tipe permintaan transfer adalah transfer masuk, pengguna akan diminta untuk memasukkan jumlah barang yang akan ditransfer masuk. Setelah semua data terisi dengan lengkap, pengguna dapat menekan tombol submit untuk membuat permintaan transfer. Sistem akan memproses data yang telah diinput dan permintaan transfer akan berhasil dibuat.

E Daftar Permintaan Transfer

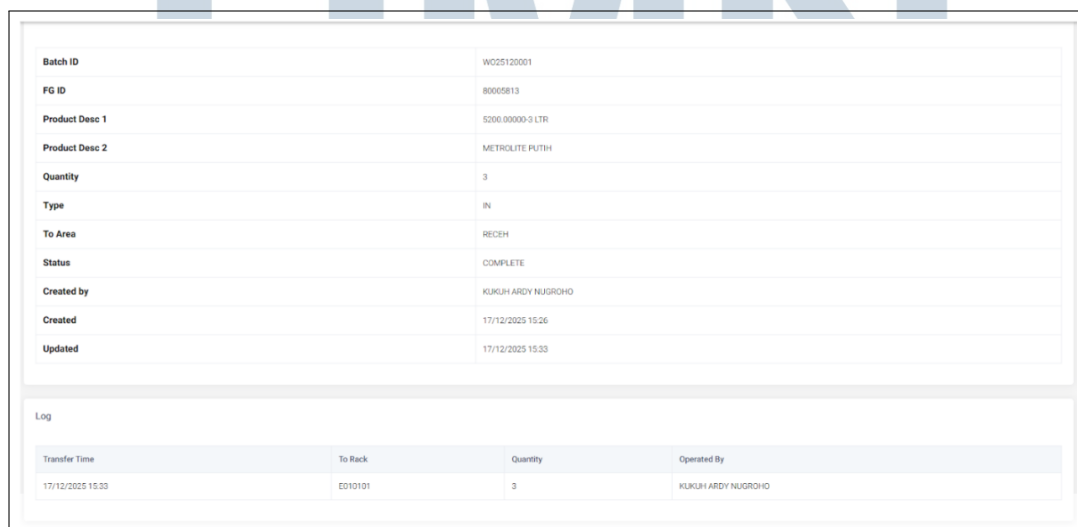


Number	Batch ID	FG ID	Item Desc 1	Item Desc 2	Type	To Area	Quantity	Status	Request Date	Tracking	Sync Status	Action
TR-202512-49	W025120002	80005813	5200.00000-3 LTR	METROLITE PUTIH	OUT	PRODUKSI	1	PENDING	17/12/2025 15:26	✓	-	View, Delete, Download
TR-202512-48	W025120001	80005813	5200.00000-3 LTR	METROLITE PUTIH	IN	RECEH	3	PENDING	17/12/2025 15:26	✓	-	View, Delete, Download

Gambar 3.26. Halaman Daftar Permintaan Transfer

Halaman daftar permintaan transfer berfungsi untuk menampilkan seluruh daftar permintaan transfer yang pernah tercatat di dalam Sistem Manajemen Gudang Receh, seperti yang dapat dilihat pada Gambar 3.26. Tabel data pada halaman ini menampilkan seluruh permintaan transfer beserta informasi yang relevan, disertai dengan beberapa tombol aksi seperti melihat detail, membatalkan permintaan transfer, dan membuat bukti transfer.

Halaman ini juga dilengkapi dengan fitur *filter* dan *searching* untuk memudahkan pengguna dalam menampilkan data yang dibutuhkan. Selain itu, terdapat tombol buat permintaan transfer yang berfungsi untuk membuka halaman *form create* permintaan transfer.



Batch ID	W025120001
FG ID	80005813
Product Desc 1	5200.00000-3 LTR
Product Desc 2	METROLITE PUTIH
Quantity	3
Type	IN
To Area	RECEH
Status	COMPLETE
Created by	KURUJI ARDY NUGROHO
Created	17/12/2025 15:26
Updated	17/12/2025 15:33

Transfer Time	To Rack	Quantity	Operated By
17/12/2025 15:33	ED10101	3	KURUJI ARDY NUGROHO

Gambar 3.27. Halaman Detail Permintaan Transfer Masuk

Detail Permintaan Transfer	
Batch ID	W025120002
FG ID	80005813
Product Desc 1	5200.00000-3 LTR
Product Desc 2	METROLITE PUTIH
Type	OUT
To Area	PRODUKSI
Status	COMPLETE
Created by	KUKUH ARDY NUGROHO
Created	17/12/2025 15:36
Updated	17/12/2025 15:39

Log Transfer Out				
Transfer Time	Batch Taken	From Rack	Quantity	Operated By
17/12/2025 15:39	W025120001	E010101	3	KUKUH ARDY NUGROHO

Gambar 3.28. Halaman Detail Permintaan Transfer Keluar

Saat pengguna menekan tombol lihat detail pada baris data di halaman daftar permintaan transfer, pengguna dapat melihat tampilan detail dari permintaan transfer terkait. di halaman ini, selain dapat melihat informasi lebih lengkap, pengguna juga dapat melihat log waktu permintaan transfer tersebut di proses jika status permintaan transfer sudah *complete*. Tampilan halaman detail permintaan transfer akan memiliki sedikit perbedaan berdasarkan tipe permintaan transfer seperti yang dapat dilihat pada Gambar 3.27 dan Gambar 3.28.



STRUK GR TRANSFER STOCK IN

NO. PERMINTAAN TRANSFER : TR-202512-50

TANGGAL PERMINTAAN TRANSFER : 17 Desember 2025

BATCH ID : W025120001

FG ID : 80005813

FG DESC 1 : 5200.00000-3 LTR

FG DESC 2 : METROLITE PUTIH

TUJUAN : RECEH

TOTAL TRANSFER : 3 PCS

PRODUKSI

PPIC

PENGIRIM

PENERIMA

Gambar 3.29. Hasil *Generate* Bukti Transfer Masuk

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



STRUK GR TRANSFER STOCK OUT

NO. PERMINTAAN TRANSFER : TR-202512-51
TANGGAL PERMINTAAN TRANSFER : 17 Desember 2025
TANGGAL TRANSFER : 17 Desember 2025
BATCH ID : W025120002
FG ID : 80005813
FG DESC 1 : 5200.00000-3 LTR
FG DESC 2 : METROLITE PUTIH
TUJUAN : PRODUKSI

DETAIL STOCK ASAL:

No	Dari Batch	Dari Rak	Jumlah
1	W025120001	E010101	3 PCS

TOTAL TRANSFER KELUAR : 3 PCS

PPIC

PRODUKSI

PENGIRIM

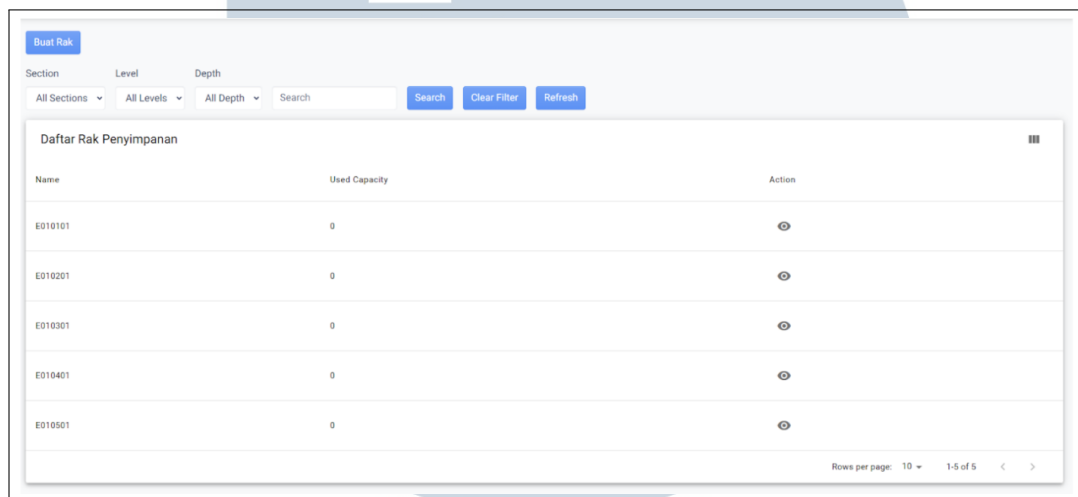
PENERIMA

Gambar 3.30. Hasil *Generate* Bukti Transfer Keluar

Ketika pengguna menekan tombol unduh pada baris data di halaman daftar permintaan transfer, sistem akan mengunduh struk transfer dari permintaan transfer terkait. Khusus untuk permintaan transfer keluar, tombol unduh hanya dapat diakses ketika status permintaan transfer telah berstatus *complete*.

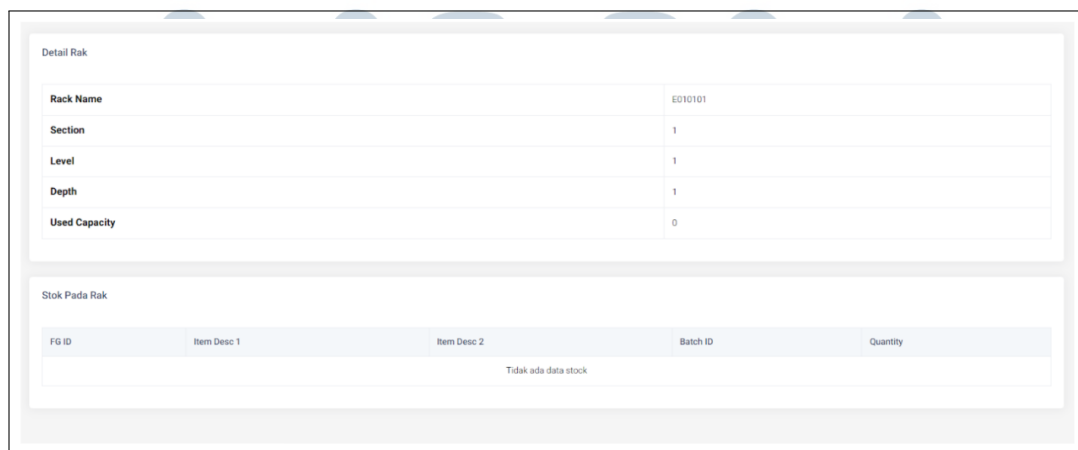
Konten struk transfer memiliki perbedaan berdasarkan tipe permintaan transfer, seperti yang ditunjukkan pada Gambar 3.29 untuk struk transfer masuk dan Gambar 3.30 untuk struk transfer keluar. Fitur pembuatan struk transfer ini digunakan sebagai bukti tertulis dari setiap kegiatan transfer yang terjadi di dalam sistem.

F Daftar Rak



Name	Used Capacity	Action
E010101	0	
E010201	0	
E010301	0	
E010401	0	
E010501	0	

Gambar 3.31. Halaman Daftar Rak



Rack Name	E010101
Section	1
Level	1
Depth	1
Used Capacity	0

FG ID	Item Desc 1	Item Desc 2	Batch ID	Quantity
Tidak ada data stock				

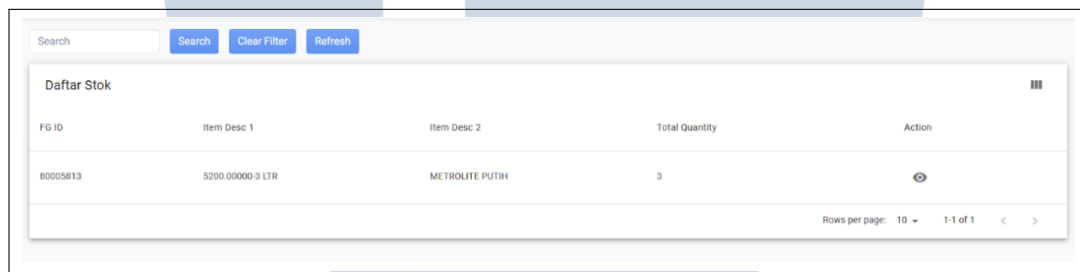
Gambar 3.32. Halaman Detail Rak

Halaman daftar rak berfungsi untuk menampilkan informasi lokasi penyimpanan di dalam Sistem Manajemen Gudang Receh, seperti yang dapat dilihat pada Gambar 3.31. Tabel data pada halaman ini menampilkan daftar kode rak dan jumlah barang yang tersimpan di setiap lokasi penyimpanan, disertai dengan tombol aksi untuk melihat detail

rak. Ketika pengguna menekan tombol detail rak, sistem akan membuka halaman detail rak seperti yang ditunjukkan pada Gambar 3.32. Halaman ini menampilkan informasi rak yang lebih lengkap beserta rincian stok barang yang tersimpan di dalam lokasi penyimpanan tersebut.

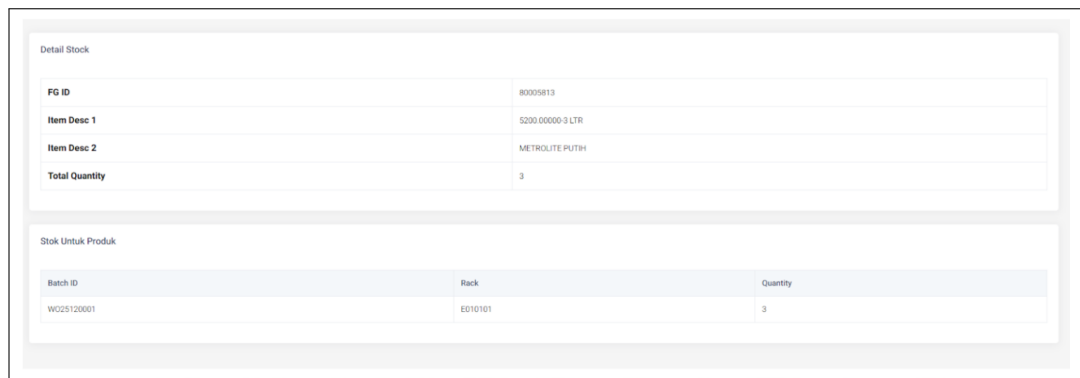
Halaman ini juga dilengkapi dengan fitur *filter* berdasarkan koordinat lokasi (*section, level, depth*) dan *searching* untuk memudahkan pengguna dalam menampilkan data rak yang dibutuhkan. Pada halaman juga terdapat tombol buat rak yang berfungsi untuk membuka halaman *form create* rak.

G Daftar Stok



FG ID	Item Desc 1	Item Desc 2	Total Quantity	Action
80005813	S200.00000-3 LTR	METROLITE PUTIH	3	

Gambar 3.33. Halaman Daftar Stok



FG ID	80005813
Item Desc 1	S200.00000-3 LTR
Item Desc 2	METROLITE PUTIH
Total Quantity	3

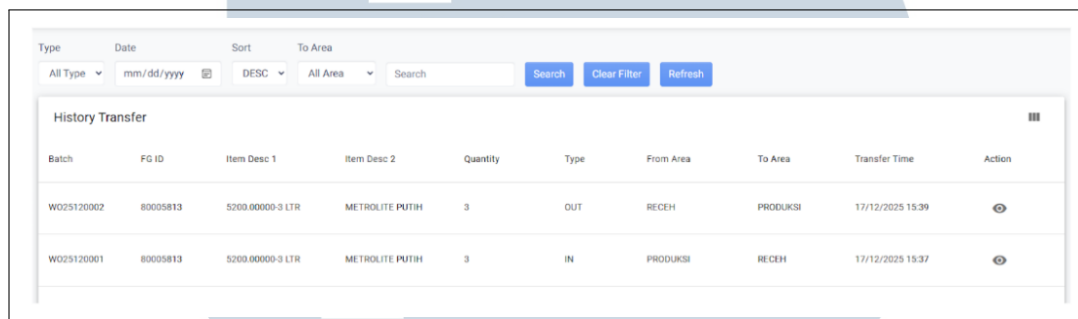
Batch ID	Rack	Quantity
W025120001	E010101	3

Gambar 3.34. Halaman Detail Stok

Halaman daftar stok berfungsi untuk menampilkan informasi stok yang tersedia di dalam Sistem Manajemen Gudang Recoh, seperti yang dapat dilihat pada Gambar 3.33. Tabel data pada halaman ini menampilkan daftar stok yang tersedia berdasarkan nama produk beserta total jumlahnya, disertai dengan tombol aksi untuk melihat detail stok. Halaman ini juga dilengkapi dengan fitur *searching* untuk memudahkan pengguna dalam menampilkan data produk yang dibutuhkan.

Ketika pengguna menekan tombol detail stok, sistem akan membuka halaman detail stok seperti yang ditunjukkan pada Gambar 3.34. Halaman ini menampilkan informasi lengkap produk beserta tabel rincian stok yang memuat kode batch dan lokasi penyimpanan dari setiap pecahan stok produk tersebut.

H Daftar Riwayat Transfer

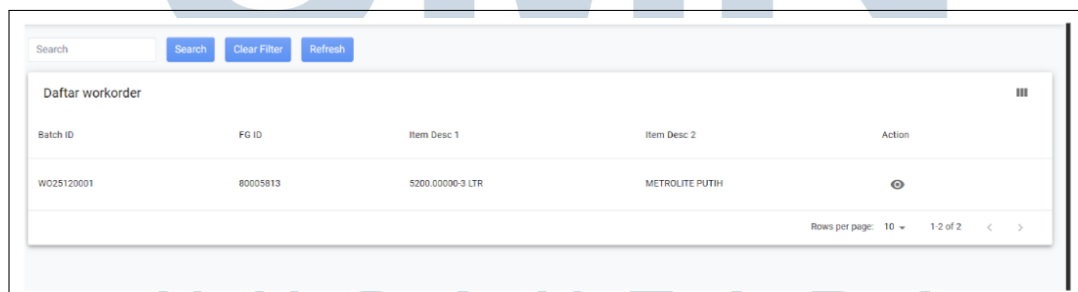


Batch	FG ID	Item Desc 1	Item Desc 2	Quantity	Type	From Area	To Area	Transfer Time	Action
W025120002	80005813	5200.00000-3 LTR	METROLITE PUTIH	3	OUT	RECEH	PRODUKSI	17/12/2025 15:39	
W025120001	80005813	5200.00000-3 LTR	METROLITE PUTIH	3	IN	PRODUKSI	RECEH	17/12/2025 15:37	

Gambar 3.35. Halaman Daftar Riwayat Transfer

Halaman daftar riwayat transfer berfungsi untuk menampilkan informasi riwayat transfer yang telah diproses di aplikasi *mobile*, seperti yang dapat dilihat pada Gambar 3.35. Tabel data pada halaman ini menampilkan daftar riwayat transfer dengan informasi yang relevan, disertai dengan tombol aksi untuk melihat detail riwayat transfer. Halaman ini juga dilengkapi dengan fitur *filter* dan *searching* untuk memudahkan pengguna dalam menampilkan data yang dibutuhkan.

I Daftar Laporan Transfer



Batch ID	FG ID	Item Desc 1	Item Desc 2	Action
W025120001	80005813	5200.00000-3 LTR	METROLITE PUTIH	

Rows per page: 10 1-2 of 2

Gambar 3.36. Halaman Daftar Laporan Transfer

Detail Workorder Transfer

Batch ID	WG25120001
FG ID	80005813
Product Desc 1	S200.00000-3 LTR
Product Desc 2	METROLITE PUTIH

Log Transfer Dari Produksi Ke Gudang Receh

Transfer Time	To Rack	Quantity	Operated By
17/12/2025 15:37	E010101	3 PCS	KUKUH ARDY NUGROHO
TOTAL : 3 PCS			

Log Transfer Dari Gudang Receh Ke Produksi

Transfer Time	From Rack	For Batch	Quantity	Operated By
17/12/2025 15:39	E010101	WG25120002	3 PCS	KUKUH ARDY NUGROHO
TOTAL : 3 PCS				

Gambar 3.37. Halaman Detail Laporan Transfer

Halaman daftar laporan transfer berfungsi untuk menampilkan informasi *Work Order* (WO) yang pernah diproses dalam Sistem Manajemen Gudang Receh, seperti yang ditunjukkan pada Gambar 3.36. Tabel data pada halaman ini menampilkan daftar WO beserta data produk terkait, disertai dengan tombol aksi untuk melihat detail laporan transfer. Halaman ini juga dilengkapi dengan fitur *searching* untuk membantu pengguna dalam menampilkan data yang dibutuhkan.

Ketika pengguna menekan tombol detail, sistem akan membuka halaman detail laporan transfer seperti yang ditunjukkan pada Gambar 3.37. Halaman ini menampilkan data pergerakan produk dari WO tersebut, meliputi aktivitas transfer masuk ke gudang receh dan transfer keluar dari gudang receh.

3.6 Pengujian Sistem

Bagian ini menyajikan hasil pengujian sistem yang dilakukan menggunakan metode *Black Box Testing* untuk memastikan setiap fitur yang dikembangkan berfungsi dengan baik dan sesuai dengan kebutuhan pengguna. Pengujian dilakukan dengan menerapkan beberapa skenario yang telah ditentukan, baik untuk kondisi input yang benar maupun salah. Seluruh pengujian dilakukan secara manual oleh Bapak Christoferus Teguh Aryanto Sudarmadi selaku Kepala Departemen IT.

Tabel 3.11 memperlihatkan daftar fitur yang diuji, skenario tes yang diterapkan, hasil yang diharapkan, hasil pengujian aktual, dan kesimpulan dari pengujian berdasarkan kesesuaian ekspektasi dengan hasil aktual.

Tabel 3.11. Hasil Pengujian Sistem Manajemen Gudang Receh

No	Pengujian (Scope)	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Login dengan kredensial valid (Aplikasi <i>Mobile</i>)	User diarahkan ke halaman utama	Berpindah ke halaman utama	Sesuai
2	Login dengan kredensial tidak valid (Aplikasi <i>Mobile</i>)	Muncul notifikasi error kredensial tidak valid	Muncul pesan error <i>invalid credentials</i>	Sesuai
3	Scan kode QR <i>transfer request</i> terdaftar (<i>Mobile</i> – IN/OUT)	Informasi permintaan transfer muncul	data permintaan transfer muncul	Sesuai
4	Scan kode QR <i>transfer request</i> tidak terdaftar (<i>Mobile</i> – IN/OUT)	Muncul notifikasi error permintaan tidak valid	muncul error permintaan transfer tidak tersedia atau sudah selesai	Sesuai
5	Pencarian rak dengan input benar (<i>Mobile</i> – Transfer In)	Informasi rak muncul	informasi rak muncul	Sesuai
6	Pencarian rak dengan input salah (<i>Mobile</i> – Transfer In)	Muncul notifikasi error rak tidak valid	muncul error bahwa rak tidak ditemukan	Sesuai
7	Fitur rekomendasi rak (<i>Mobile</i> – Transfer In)	Informasi rak tersedia muncul	berhasil menampilkan data rak	Sesuai
8	Scan rak sesuai pilihan (<i>Mobile</i> – Transfer In)	Rak berhasil diverifikasi	status rak berhasil berubah menjadi <i>verified</i>	Sesuai
9	Scan rak tidak sesuai pilihan (<i>Mobile</i> – Transfer In)	Muncul notifikasi error rak tidak sesuai	muncul error rak yang di scan tidak sesuai pilihan	Sesuai

No	Pengujian (Scope)	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
10	<i>Submit Transfer In</i> (<i>Mobile – Transfer In</i>)	Transfer In berhasil disubmit	muncul pesan transfer berhasil	Sesuai
11	Pengambilan data stok tersedia (<i>Mobile – Transfer Out</i>)	Data stok tersedia muncul	Stok tersedia muncul	Sesuai
12	Pemilihan satu stok (<i>Mobile – Transfer Out</i>)	Informasi satu stok terpilih	satu stok berhasil terpilih	Sesuai
13	Pemilihan beberapa stok (<i>Mobile – Transfer Out</i>)	Informasi beberapa stok terpilih	beberapa stok berhasil terpilih	Sesuai
14	Fitur <i>Change Selection</i> (<i>Mobile – Transfer Out</i>)	Tampilan kembali ke stok tersedia	tampilan berubah ke daftar stok tersedia	Sesuai
15	<i>Scan</i> stok sesuai pilihan (<i>Mobile – Transfer Out</i>)	Stok berhasil diverifikasi	status stok berhasil berubah menjadi <i>verified</i>	Sesuai
16	<i>Scan</i> stok tidak sesuai pilihan (<i>Mobile – Transfer Out</i>)	Muncul notifikasi error stok tidak sesuai	muncul error stok yang di scan tidak sesuai pilihan	Sesuai
17	<i>Submit Transfer Out</i> (<i>Mobile – Transfer Out</i>)	Transfer Out berhasil disubmit	muncul pesan transfer keluar berhasil	Sesuai
18	Pembuatan permintaan transfer valid IN/OUT (<i>Dashboard Web</i>)	Permintaan transfer berhasil dibuat	permintaan transfer berhasil dibuat	Sesuai
19	Pembuatan permintaan transfer tidak valid (<i>Dashboard Web</i>)	Muncul notifikasi error data tidak valid	muncul pesan error bahwa data tidak lengkap/valid	Sesuai

No	Pengujian (Scope)	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
20	<i>Cancel</i> permintaan transfer (<i>Dashboard Web</i>)	Permintaan transfer berhasil dibatalkan	status permintaan transfer berubah menjadi <i>cancel</i>	Sesuai
21	Buat struk transfer (<i>Dashboard Web</i>)	Struk transfer berhasil dibuat	berhasil unduh struk transfer	Sesuai
22	Pembuatan rak valid (<i>Dashboard Web</i>)	Data rak berhasil dibuat	berhasil membuat data rak baru	Sesuai
23	Pembuatan rak tidak valid (<i>Dashboard Web</i>)	Muncul notifikasi <i>error</i> data tidak valid	muncul error bahwa data yang di input tidak lengkap/valid	Sesuai
24	<i>Filter</i> dan <i>pagination</i> permintaan transfer (<i>Dashboard Web</i>)	Data berubah sesuai <i>filter</i> dan <i>pagination</i>	data berubah sesuai <i>filter</i> dan bisa berpindah <i>page</i>	Sesuai
25	<i>Filter</i> dan <i>pagination</i> data rak (<i>Dashboard Web</i>)	Data berubah sesuai <i>filter</i> dan <i>pagination</i>	data berubah sesuai <i>filter</i> dan bisa berpindah <i>page</i>	Sesuai
26	<i>Filter</i> dan <i>pagination</i> data log (<i>Dashboard Web</i>)	Data berubah sesuai <i>filter</i> dan <i>pagination</i>	data berubah sesuai <i>filter</i> dan bisa berpindah <i>page</i>	Sesuai
27	<i>Filter</i> dan <i>pagination</i> data stok (<i>Dashboard Web</i>)	Data berubah sesuai <i>filter</i> dan <i>pagination</i>	data berubah sesuai <i>filter</i> dan bisa berpindah <i>page</i>	Sesuai
28	<i>Filter</i> dan <i>pagination</i> data log transfer (<i>Dashboard Web</i>)	Data berubah sesuai <i>filter</i> dan <i>pagination</i>	data berubah sesuai <i>filter</i> dan bisa berpindah <i>page</i>	Sesuai
29	<i>Filter</i> dan <i>pagination</i> data transfer report (<i>Dashboard Web</i>)	Data berubah sesuai <i>filter</i> dan <i>pagination</i>	data berubah sesuai <i>filter</i> dan bisa berpindah <i>page</i>	Sesuai

3.7 Kendala dan Solusi yang Ditemukan

A Kendala

Kendala utama yang ditemukan selama pelaksanaan magang adalah lamanya waktu menunggu proses demo kepada pengguna untuk mendapatkan umpan balik. Apabila terdapat revisi atau permintaan perubahan baru dapat diketahui saat proses demo selanjutnya. Hal ini menyebabkan progres pengembangan sistem terhambat karena tergantung pada ketersediaan waktu demo dengan pengguna.

B Solusi

Untuk mengatasi kendala tersebut, dilakukan koordinasi yang lebih intensif dengan pengguna di luar jadwal demo resmi. Koordinasi dilakukan melalui *Whatsapp* dan Telfon untuk mendapatkan umpan balik secara lebih cepat terkait revisi atau permintaan tambahan fitur. Pendekatan ini memungkinkan proses pengembangan berjalan lebih efisien tanpa harus menunggu jadwal demo berikutnya.

