

BAB III

METODOLOGI PENELITIAN

3.1. Jenis dan Pendekatan Penelitian

Jenis penelitian yang digunakan dalam penelitian ini adalah penelitian Perancangan atau sering disebut sebagai Penelitian Pengembangan R&D. Fokus utama dari penelitian ini adalah merancang dan membangun sebuah produk perangkat lunak berupa sistem informasi manajemen gudang yang dapat diimplementasikan secara nyata pada PT. Plastindo Pack Printing. Penelitian ini bersifat aplikatif, di mana solusi teknis dikembangkan untuk menjawab permasalahan operasional yang ada di lapangan.

3.2. Lokasi dan Waktu Penelitian

3.2.1. Lokasi Penelitian

Penelitian ini dilaksanakan di PT. Plastindo Pack Printing yang berlokasi di Jl. Raya Kutruk Rancaiyuh no 89, RT.003/RW.004, Pasir Bar., Kec. Jambe, Kabupaten Tangerang, Banten 15720. Mengingat efisiensi dan aksesibilitas, pengambilan data dilakukan secara *Hybrid* atau Virtual dan Observasi Lapangan.

Peneliti melakukan Observasi Virtual menggunakan platform Zoom Meeting dengan bantuan informan staf pergudangan yang berada di lokasi. Melalui metode ini, peneliti mengamati secara langsung alur masuk-keluar barang, penempatan raw material, serta demonstrasi cara kerja karyawan di rantai produksi melalui transmisi video. Hal ini dilakukan untuk memastikan perancangan sistem informasi tetap akurat sesuai dengan kondisi fisik dan operasional pabrik.

3.2.2. Waktu Penelitian

Penelitian ini direncanakan berlangsung selama terhitung mulai dari bulan april 2026 sampai dengan Juli 2026. Jadwal pengambilan data melalui Zoom Meeting dilakukan secara rutin berdasarkan kesepakatan dengan pihak perusahaan guna melakukan tahapan evaluasi pada metode Prototype.

3.3. Objek dan Subjek Penelitian

3.3.1. Objek Penelitian

Objek dalam penelitian ini difokuskan pada manajemen gudang dan bahan baku (*Raw Material*) pada PT. Plastindo Pack Printing. Hal ini mencakup seluruh siklus digitalisasi data yang meliputi pencatatan sirkulasi barang masuk, aktivitas penggunaan material dasar pada mesin produksi, hingga mekanisme pelaporan ketersediaan inventaris yang mengintegrasikan aplikasi *mobile* dengan *backend server*.

3.3.2. Subjek Penelitian

Subjek penelitian merepresentasikan para pemangku kepentingan yang bertindak sebagai penyedia data sekaligus calon pengguna (*end-user*) dari sistem yang dikembangkan. Adapun komposisi subjek dalam penelitian ini meliputi:

1. Informan Personel, bagian stok dan pergudangan di PT. Plastindo Pack Printing yang berperan sebagai fasilitator dalam pemetaan data operasional awal.
2. Karyawan operasional di rantai produksi dan fasilitas gudang, keterlibatan mereka sangat krusial dalam memberikan umpan balik (*feedback*) terhadap prototipe aplikasi, di mana evaluasi terkait aspek antarmuka (UI) dan fungsionalitas menjadi landasan utama dalam iterasi perancangan sistem.
3. Manajemen PT. Plastindo Pack Printing, pihak pengambil kebijakan yang melakukan validasi terhadap akurasi laporan manajerial yang diproduksi oleh sistem.

3.4. Alat dan Bahan Penelitian

3.4.1 Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam penelitian ini bertujuan untuk mendukung proses pengembangan serta pengujian aplikasi agar berjalan secara optimal pada berbagai kondisi perangkat pengguna. Adapun spesifikasi perangkat keras yang digunakan adalah sebagai berikut:

1. Laptop digunakan sebagai perangkat utama dalam proses pengembangan aplikasi berbasis Laravel dan Flutter secara bersamaan. Spesifikasi yang

digunakan disesuaikan untuk mendukung kebutuhan komputasi yang tinggi selama proses coding, debugging, dan build aplikasi.:

- a) Prosesor: Intel Core i7-10750H
 - b) Memory (RAM): 32GB DDR4
 - c) Penyimpanan: 1536GB
2. Perangkat mobile digunakan untuk melakukan pengujian fungsionalitas dan kompatibilitas aplikasi pada berbagai tingkat spesifikasi perangkat. Pengujian dilakukan pada dua kategori perangkat, yaitu kelas bawah (low-end) dan kelas atas (high-end), untuk memastikan aplikasi dapat berjalan dengan baik pada kondisi perangkat yang berbeda diantaranya.:
- a) Digunakan untuk menguji performa aplikasi pada perangkat dengan spesifikasi terbatas, sehingga dapat diketahui apakah aplikasi tetap berjalan dengan baik pada kondisi minimum dengan spesifikasi berikut:
 - i. Model: Redmi 10 5G
 - ii. Prosesor: Mediatek Dimensity 700
 - iii. Ram: 6GB
 - iv. Penyimpanan: 128GB
 - b) Digunakan untuk menguji performa aplikasi pada perangkat dengan spesifikasi tinggi, sehingga dapat dibandingkan dengan hasil pengujian pada perangkat low-end.:
 - i. Model: Sony Xperia 1 II
 - ii. Prosesor: Snapdragon 865 5G
 - iii. Ram: 8GB
 - iv. Penyimpanan: 192GB
3. Server Testing: Cloud Hosting (DomaiNesia paket 25GB) sebagai infrastruktur untuk testing deployment backend Laravel 12 dan database MySQL agar dapat diakses secara daring.

3.4.2. Perangkat Lunak (Software)

Perangkat lunak yang digunakan sebagai bahan penunjang pengerjaan sistem adalah:

1. Sistem operasi Windows sebagai lingkungan kerja pengembangan.
2. XAMPP untuk menjalankan proyek secara lokal
3. Bahasa pemrograman dan framework PHP 8.3+ (Laravel 12) untuk backend dan *Dart Flutter* untuk pengembangan mobile *hybrid*.
4. Basis data menggunakan MySQL/MariaDB sebagai media penyimpanan data stok dan bahan baku.
5. Editor kode menggunakan Visual Studio Code
6. Alat Perancangan dan Dokumentasi menggunakan Figma untuk desain UI/UX dan prototyping
7. Alat Komunikasi dan observasi menggunakan Discord dan Zoom Meeting untuk melakukan observasi virtual dan koordinasi dengan pihak PT. Plastindo Pack Printing.

3.5. Teknik Pengumpulan Data

Untuk mendapatkan data yang akurat dan relevan dalam mendukung perancangan sistem ini, peneliti menggunakan beberapa teknik pengumpulan data sebagai berikut

3.5.1. Observasi

Peneliti melakukan pengamatan secara daring kepada personel yang terkait terhadap alur kerja operasional di PT. Plastindo Pack Printing. Mengingat kondisi tertentu, observasi dilakukan secara *hybrid*:

1. Observasi visual dilakukan melalui platform *Zoom Meeting*, di mana informan kunci menunjukkan tata letak gudang, proses perpindahan bahan baku ke mesin, serta cara karyawan mencatat data secara manual.
2. Tujuannya Memahami hambatan fisik seperti tangan karyawan yang kotor atau area minim sinyal yang mempengaruhi desain antarmuka aplikasi Flutter.

3.5.2. Wawancara

Peneliti melakukan wawancara semi-terstruktur dengan dua pihak utama dimana:

1. Pihak Manajemen, untuk mengetahui kebutuhan informasi strategis, seperti laporan stok bulanan dan monitoring efisiensi bahan baku.
2. Staf gudang, untuk menelusuri kendala teknis di lapangan, alasan kegagalan sistem website sebelumnya, dan fitur apa yang paling dibutuhkan oleh karyawan yang memiliki keterbatasan literasi digital.

3.5.3. Dokumentasi

Peneliti mengumpulkan data sekunder berupa dokumen-dokumen yang selama ini digunakan oleh perusahaan dalam mengelola gudang, antara lain

1. Catatan manual stok barang masuk dan keluar.
2. Data inventaris mesin produksi.
3. Struktur organisasi dan profil PT. Plastindo Pack Printing.

3.6. Prosedur Penelitian

Prosedur penelitian ini dijalankan secara sistematis dengan menerapkan model pengembangan Prototype. Tahapan penelitian digambarkan melalui alur kerja sebagai berikut:

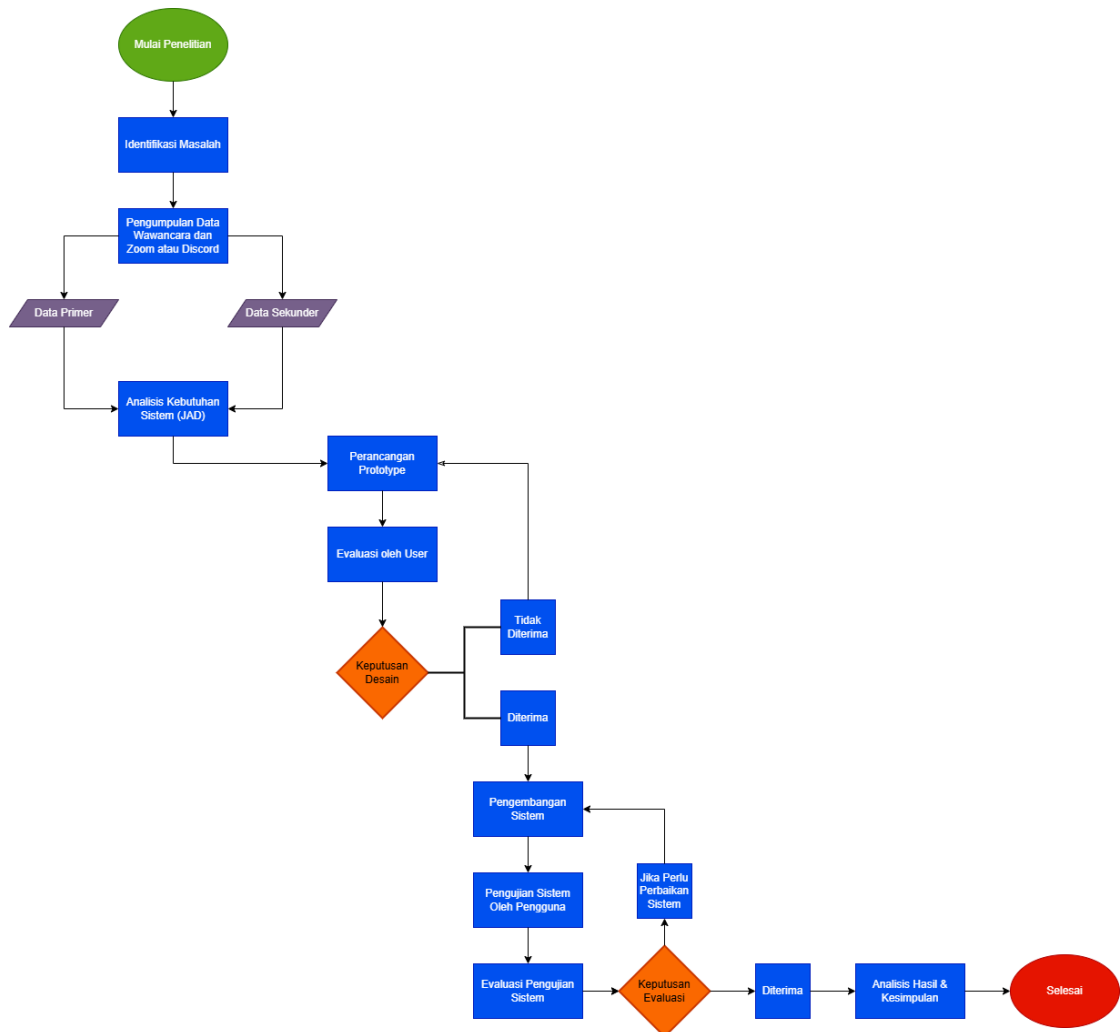
1. Identifikasi Masalah, memulai penelitian dengan menentukan akar permasalahan di PT. Plastindo Pack Printing, yaitu ketidakefektifan sistem inventaris lama dan kendala literasi digital karyawan dalam mengelola data stok serta bahan baku.
2. Pengumpulan Data Wawancara dan Zoom/Discord, melakukan pengambilan data secara *hybrid*. Proses komunikasi dan observasi dilakukan melalui tatap muka virtual menggunakan platform Zoom atau Discord. Data yang dikumpulkan terbagi menjadi dua kategori,
 - a. Data Primer, hasil wawancara langsung dengan pemilik dan staf gudang mengenai alur kerja riil.
 - b. Data Sekunder, dokumen internal perusahaan, daftar stok fisik, serta aset foto produk jika ada.
3. Berdasarkan data yang terkumpul, peneliti melakukan analisis terhadap kebutuhan fungsional dan non-fungsional sistem dengan menerapkan metode *Joint Application Design* (JAD). Melalui pendekatan kolaboratif

JAD ini, peneliti berdiskusi secara langsung dengan pihak manajemen dan staf operasional gudang guna mendefinisikan masalah serta merumuskan solusi bersama-sama secara akurat. Tahap ini secara spesifik menentukan fitur dan spesifikasi ruang lingkup apa saja yang harus dibangun pada aplikasi *mobile* berbasis Flutter (untuk sisi karyawan lapangan) dan *web dashboard* berbasis Laravel 12 (untuk sisi manajemen).

4. Perancangan Prototype Awal Menggunakan Figma atau soft coding, merancang antarmuka (*User Interface*) dan alur pengguna (*User Experience*) menggunakan alat bantu Figma. Fokus perancangan adalah pada kemudahan akses visual untuk meminimalisir kesalahan input oleh pengguna, Tahap perancangan antarmuka melalui desain *mockup* dilakukan untuk memberikan gambaran visual kepada pengguna sebelum masuk ke tahap pengkodean, guna memastikan setiap fitur mudah dipahami oleh pengguna awam (Puspita et al., 2024)..
5. Evaluasi oleh User dan keputusan desain, prototipe yang telah dibuat di Figma didemonstrasikan kepada pengguna (*user*) di PT. Plastindo. Pada tahap ini, terdapat Keputusan Desain:
 - a. Tidak Diterima, jika pengguna merasa desain masih sulit atau ada fitur yang kurang, peneliti akan kembali ke tahap perancangan di Figma atau pengkodean awal yang dapat dirubah (iterasi).
 - b. Diterima, jika desain telah memenuhi ekspektasi pengguna, penelitian berlanjut ke tahap berikutnya.
6. Pengembangan sistem setelah desain disetujui, melakukan proses pengkodean (*coding*) secara utuh. Tahap ini meliputi pembangunan backend menggunakan Laravel 12 dan aplikasi mobile menggunakan Flutter.
7. Pengujian Sistem oleh Pengguna Sistem yang telah dikembangkan kemudian diuji coba langsung oleh karyawan di lingkungan kerja PT. Plastindo Pack Printing untuk memastikan semua fungsi berjalan dengan baik dan stabil.

8. Melakukan evaluasi mendalam terhadap hasil pengujian sistem bersama pihak manajemen dan karyawan. Jika pada tahap ini masih ditemukan *bug* atau pengguna merasa ada fitur teknis yang perlu disesuaikan dengan kondisi lapangan, maka akan dilakukan kembali (*loopback*) ke tahap Pengembangan Sistem untuk proses perbaikan dan penyempurnaan kode. Jika sistem telah dinilai valid dan memenuhi seluruh kebutuhan, maka proses dilanjutkan ke tahap akhir.
9. Analisis Hasil & Kesimpulan, Melakukan analisis akhir terhadap efektivitas sistem yang telah diimplementasikan penuh dalam menyelesaikan masalah manajemen stok di lapangan. Pada tahap ini, dilakukan penyebaran instrumen *User Satisfaction Questionnaire* untuk mengukur dan mengevaluasi tingkat kepuasan pengguna akhir terhadap antarmuka serta fungsionalitas aplikasi. Hasil analisis dari data kuesioner dan pengujian ini kemudian dirangkum secara komprehensif menjadi kesimpulan tugas akhir, beserta rumusan saran untuk pengembangan sistem di masa depan.

3.6.1. Flowchart Prosedur Penelitian



Gambar 3.7. Prosedur Penelitian.

3.7. Pengolahan dan Analisis Data

Tahap pengolahan dan analisis data bertujuan untuk mengubah data mentah hasil observasi dan wawancara menjadi informasi yang dapat diimplementasikan ke dalam arsitektur sistem.

3.7.1. Pengolahan Data

Data yang diperoleh dari PT. Plastindo Pack Printing diolah melalui beberapa tahapan:

1. Reduksi Data, menyaring hasil wawancara dan observasi virtual (Zoom/Discord) untuk mengambil poin-poin penting terkait alur stok barang dan kebutuhan fungsional sistem.
2. Kategorisasi Data, mengelompokkan data menjadi entitas database, seperti data barang, data *raw material*, data pengguna, dan data aktivitas mesin jika ada.
3. Digitalisasi Aset, Mengolah foto produk menjadi format digital yang siap diunggah ke dalam sistem melalui Laravel *storage*.

3.7.2. Metode Analisis

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah metode Prototype. Pada tahap analisis kebutuhan, digunakan pendekatan *Joint Application Design* (JAD) untuk mengumpulkan kebutuhan sistem melalui keterlibatan langsung pengguna. Dengan demikian, kebutuhan sistem yang diperoleh menjadi lebih akurat dan sesuai dengan kondisi di lapangan. JAD merupakan teknik pengembangan sistem yang melibatkan komitmen penuh dari pengguna (*end-users*) dan pengelola dalam menentukan kebutuhan serta merancang antarmuka sistem secara bersama-sama dalam sebuah forum diskusi atau workshop yang terstruktur (Aldisa et al., 2022).

Implementasi JAD dalam penelitian manajemen stok di PT. Plastindo Pack Printing akan difokuskan untuk:

1. Identifikasi Persyaratan Kolaboratif: Melibatkan pihak manajemen (pemilik) dan staf operasional (pengelola gudang) untuk mengidentifikasi variabel bahan baku karung dan kemasan secara akurat agar tidak terjadi kesalahan input data.
2. Validasi desain visual, yang dihadapi pengguna, sesi JAD digunakan untuk memvalidasi mockup antarmuka berbasis mobile yang intuitif, sehingga sistem mudah dioperasikan tanpa pelatihan yang rumit.
3. Efisiensi alur kerja dimana menentukan prioritas menu pada aplikasi, mulai dari *dashboard* ketersediaan stok hingga laporan penggunaan bahan baku produksi, untuk memastikan sinergi antara aplikasi *mobile* dengan *backend* Laravel (Rahmi et al., 2023).

3.7.3. Perangkat Lunak

yang digunakan Untuk mendukung proses pengolahan dan analisis, peneliti menggunakan:

1. Microsoft Excel atau Google Sheets, untuk mentabulasi data stok fisik sebelum diimpor ke database.
2. Draw.io, untuk memodelkan logika data melalui ERD (*Entity Relationship Diagram*) dan DFD (*Data Flow Diagram*).

3.7.4. Teknik Pengujian (Validasi)

Mengingat penelitian ini tidak menggunakan hipotesis statistik seperti uji regresi, teknik pengujian yang digunakan sebagai pengganti untuk memvalidasi keberhasilan sistem adalah:

1. *Black-box Testing*, menguji seluruh fungsionalitas fitur aplikasi tanpa melihat kode program, memastikan setiap tombol dan input berjalan sesuai yang direncanakan.
2. *User Acceptance Test (UAT)*, Melakukan pengujian penerimaan sistem secara langsung dengan melibatkan pihak manajemen PT. Plastindo Pack Printing. Pengujian ini tidak menggunakan metode kuesioner statistik, melainkan menerapkan metode *live testing* yang dikombinasikan dengan wawancara langsung melalui *platform Zoom Meeting, discord* atau *Google Meet*. Dalam pelaksanaannya, dalam skenario ini, pihak manajemen operasional tidak hanya menilai kelancaran fitur untuk kebutuhannya sendiri (di Web), tetapi juga menilai kelayakan dan kemudahan aplikasi mobile untuk nantinya diterapkan kepada bawahan/karyawan lapangan., efisiensi pencatatan stok, serta memastikan bahwa fungsi sistem telah benar-benar menjawab kendala literasi digital dan kebutuhan operasional di lapangan.

3.7.5. Skenario Teknik Pengujian *Black Box* dan *UAT*

Pada skenario teknik pengujian, di bagi menjadi 2 bagian, yaitu bagian skenario pengujian web, dan skenario pengujian mobile

1. Skenario *Black-Box Testing Web*

a. Skenario *Testing admin*

Tabel 3.7. Skenario *Testing Admin*

Admin	Sistem
Login Admin	
1. Melakukan <i>login</i> dengan menginputkan email dan password.	2. Menampilkan halaman utama atau <i>dashboard</i> .
Logout Admin	
3. Klik tombol <i>logout</i> pada <i>Navigation Bar</i>	4. Menampilkan halaman <i>login</i>
Daftar Sparepart/Bahan Baku	
5. menekan tombol tambah sparepart.	6. menampilkan menu penambahan sparepart/bahan baku.
7. Input dan simpan data <i>sparepart</i> /bahan baku baru.	8. Data tersimpan ke database
9. Menambah stok <i>sparepart</i> /bahan baku	10. Stok <i>sparepart</i> /bahan baku bertambah dan disimpan ke database
11. Edit Sparepart yang tersimpan di database	12. Menyimpan data ke database
13. Impor data <i>sparepart</i>	14. Data terimpor ke database
15. Ekspor data <i>sparepart</i>	16. Data <i>sparepart</i> akan di <i>download</i> secara otomatis
History Log/Riwayat	
17. menekan tombol menu Riwayat/ <i>History Log</i>	18. menampilkan menu Riwayat/ <i>history log</i>
19. Impor data riwayat	20. Data Riwayat akan di <i>download</i> secara otomatis
Manajemen User/Pengguna	
21. menekan tombol menu manajemen <i>user</i> /pengguna	22. menampilkan menu tambah <i>user</i> dan daftar <i>user</i>
23. input dan simpan <i>user</i> /pengguna baru.	24. Data pengguna baru tersimpan ke database
25. Menekan menu edit <i>user</i> /pengguna	26. Menampilkan menu edit <i>user</i> /pengguna
27. Edit dan simpan data	28. Menyimpan data ke database
Profil	
29. Menekan tombol profil	30. Menampilkan menu profil
31. Mengganti Password	32. Password baru akan tersimpan ke database

b. Skenario *Testing* Teknisi

Tabel 3.8. Skenario *Testing* teknisi

Teknisi	Sistem
Login Teknisi	
1. Melakukan login dengan memasukkan email dan password.	2. Menampilkan halaman utama atau <i>dashboard</i> .
Logout Teknisi	
3. Klik tombol logout pada <i>Navigation Bar</i>	4. Menampilkan halaman login
Daftar Sparepart/Bahan Baku	
5. Menambah stok sparepart/bahan baku	6. Stok sparepart/bahan baku bertambah dan disimpan ke database
7. Ekspor data sparepart	8. Data sparepart akan di <i>download</i> secara otomatis
History Log/Riwayat	
9. menekan tombol menu Riwayat/ <i>History Log</i>	10. menampilkan menu Riwayat/ <i>history log</i>
11. Impor data riwayat	12. Data Riwayat akan di <i>download</i> secara otomatis
Profil	
13. Menekan tombol profil	14. Menampilkan menu profil
15. Mengganti Password	16. Password baru akan tersimpan ke database

2. Skenario *Black-Box Testing Mobile*

Skenario *Testing* Mobile

Tabel 3.9. Skenario *Testing* Mobile

Teknisi	Sistem
Login Teknisi	
1. Melakukan login dengan memasukkan email dan password.	2. Menampilkan halaman utama atau <i>dashboard</i> .
Logout Teknisi	
3. Klik tombol logout pada <i>Navigation Bar</i>	4. Menampilkan halaman login
Daftar Sparepart/Bahan Baku	
5. Menambah stok <i>sparepart</i> /bahan baku	6. Stok <i>sparepart</i> /bahan baku bertambah dan disimpan ke database

Teknisi	Sistem
History Log/Riwayat	
7. menekan tombol menu Riwayat/ <i>History Log</i>	8. menampilkan menu Riwayat/ <i>history log</i>
Profil	
9. Menekan tombol profil	10. Menampilkan menu profil
11. Mengganti Password	12. Password baru akan tersimpan ke database

3. Skenario UAT Live Testing Pihak Manajemen Operasional

Pengujian *User Acceptance Test* (UAT) dilakukan secara komprehensif melalui demonstrasi interaktif menggunakan platform Zoom Meeting. Pengujian ini dipusatkan kepada pihak Manajemen Operasional Stok PT. Plastindo Pack Printing yang memiliki otoritas untuk memvalidasi kelayakan sistem. Pihak manajemen menguji dua platform sekaligus, yakni web-admin untuk mengevaluasi fungsi pelaporan dan pengawasan, serta aplikasi mobile untuk menilai kelayakan antarmuka sebelum didistribusikan kepada karyawan operasional di lapangan. Parameter pengujian dan wawancara dirangkum pada tabel berikut.

Tabel 3.10. Skenario *Live Testing* UAT

Sistem yang Diuji	Skenario Pengujian
Aplikasi Web-Admin	Pemantauan dashboard, melihat indikator sisa stok dan peringatan stok menipis.
Aplikasi Web-Admin	Kelancaran Manajemen data, seperti menambah, mengedit, menghapus data <i>sparepart</i> dan kategori.
Aplikasi Web-Admin	Kelancaran Laporan dan mutasi barang, mengekspor data ke Excel dan melihat <i>History Log</i> .
Aplikasi Web-Admin	Manajemen akses pembuatan akun untuk teknisi lapangan.
Aplikasi Mobile	Simulasi Operasional Lapangan Uji coba antarmuka aplikasi <i>mobile</i> .
Aplikasi Mobile	Simulasi input mutasi stok barang (Menggunakan tombol tambah/kurang stok di HP secara instan)

Sistem yang Diuji	Skenario Pengujian
Aplikasi Mobile	Melihat keakuratan dan kecepatan sinkronisasi masuk data dari aplikasi mobile ke web

3.8. Desain Sistem

Desain Sistem menggunakan beberapa jenis diagram UML atau *Unified Language Model*, selain itu juga, pembuatan desain website menggunakan wireframe untuk membantu pengembangan proyek, mengeksplorasi potensi desain, dan memvalidasi desain arsitektur perangkat lunak dan fisik yang ada, berikut beberapa diagram untuk desain sistem manajemen stok barang

3.8.1. Kebutuhan Fungsional

Kebutuhan fungsional mendefinisikan layanan, interaksi, dan operasi spesifik yang harus mampu dilakukan oleh sistem untuk memenuhi kebutuhan proses bisnis di PT. Plastindo Pack Printing. Perancangan fungsionalitas sistem informasi manajemen stok ini dibagi menjadi dua ruang lingkup aplikasi, yaitu platform *web* untuk kebutuhan manajerial dan platform *mobile* untuk kebutuhan operasional di rantai produksi.

Rincian spesifikasi kebutuhan fungsional sistem didefinisikan pada tabel berikut:

Tabel 3.11. Kebutuhan Fungsional

Aktor	Kebutuhan
Admin	1. Sistem harus menyediakan fitur autentikasi (<i>login</i> dan <i>logout</i>) untuk menjaga keamanan akses <i>dashboard</i> berbasis <i>web</i>. 2. Sistem harus menampilkan <i>dashboard</i> manajerial yang memuat ringkasan data statistik secara <i>real-time</i>, seperti total barang, stok menipis, dan angka mutasi harian. 3. Sistem harus menyediakan pengelolaan data induk <i>sparepart</i> (tambah, baca, ubah, dan hapus). 4. Sistem harus menyediakan utilitas untuk mengimpor dan mengeksport data <i>sparepart</i> serta riwayat transaksi dalam format dokumen Excel.

Aktor	Kebutuhan
	5. Sistem harus menampilkan laporan rekam jejak (<i>history log</i>) seluruh aktivitas mutasi barang beserta fitur filter pencarian. 6. Sistem harus menyediakan fitur manajemen pengguna untuk mengelola data akun dan menetapkan hak akses (<i>role</i>). 7. Sistem harus memberikan hak akses penuh kepada Admin untuk mengoperasikan seluruh fitur operasional yang dimiliki oleh Teknisi apabila diperlukan.
Teknisi	1. Sistem harus menyediakan fitur autentikasi (<i>login</i> dan <i>logout</i>) pada aplikasi <i>mobile</i> menggunakan otorisasi token. 2. Sistem harus menyediakan fitur kustomisasi <i>dashboard</i> agar teknisi dapat menyematkan (<i>pin</i>) dan mengurutkan daftar barang sesuai preferensi operasional. 3. Sistem harus menyediakan antarmuka pencatatan mutasi stok fisik (barang masuk dan keluar) secara <i>real-time</i>. 4. Sistem harus mampu melakukan validasi transaksi dan menampilkan peringatan jika jumlah pengeluaran yang diinput melebihi sisa stok di gudang. 5. Sistem harus menyediakan menu riwayat khusus untuk melihat rekam jejak mutasi barang yang dilakukan oleh akun teknisi bersangkutan. 6. Sistem harus menyediakan fitur pengelolaan profil pengguna agar teknisi dapat memperbarui kata sandi secara mandiri.

3.8.2. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan spesifikasi yang mendefinisikan atribut kualitas, kinerja, dan batasan operasional dari sistem yang dibangun.

Kebutuhan ini tidak berfokus pada apa yang dilakukan sistem, melainkan pada bagaimana sistem tersebut bekerja dalam melayani pengguna dengan baik, aman, dan efisien.

Berdasarkan analisis terhadap lingkungan operasional gudang dan manajemen, spesifikasi kebutuhan non-fungsional sistem didefinisikan pada tabel berikut:

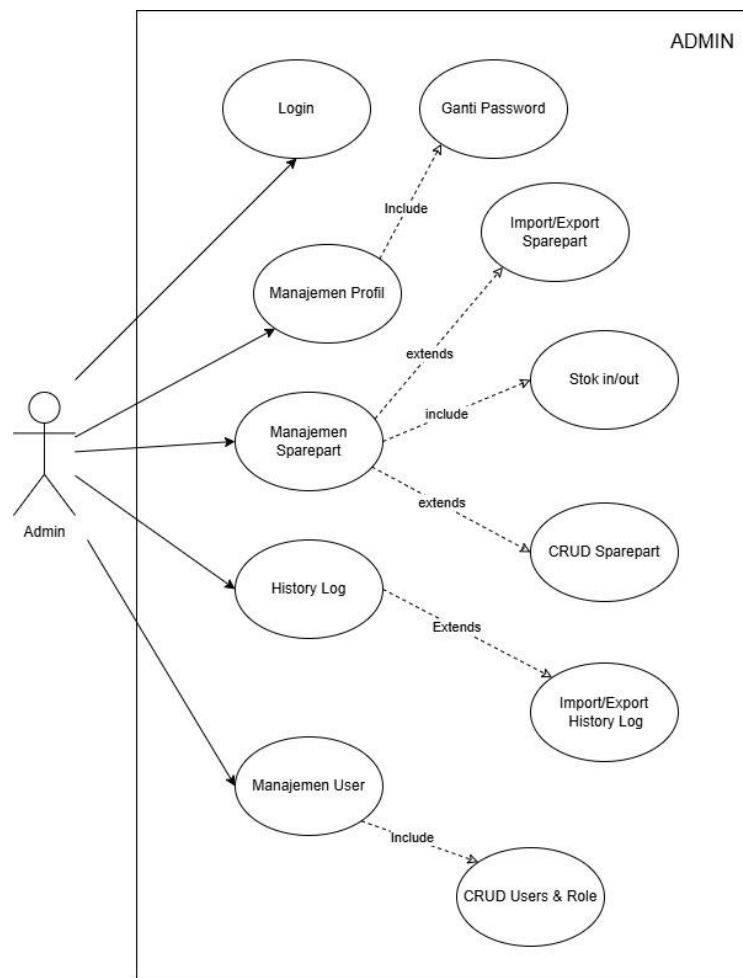
Tabel 3.12. Tabel Analisis Kebutuhan non-Fungsional

Aspek	Kebutuhan
Kinerja/ <i>Performance</i>	1. Sistem harus mampu memproses permintaan data dan menampilkan pembaruan stok dengan waktu respons kurang dari 30 detik pada kondisi jaringan internet yang stabil. 2. Aplikasi <i>mobile</i> harus dirancang ringan dan tidak membebani memori (<i>RAM</i>) perangkat genggam teknisi saat memuat daftar <i>sparepart</i> dalam jumlah besar.
Keamanan/ <i>Security</i>	1. Sistem harus mengenkripsi kata sandi seluruh pengguna secara otomatis (<i>hashing</i>) ke dalam basis data sehingga tidak dapat dibaca oleh pihak mana pun. 2. Komunikasi data antara aplikasi <i>mobile</i> dan <i>server backend</i> harus menggunakan sistem otorisasi token yang aman untuk mencegah akses dari perangkat yang tidak sah. 3. Sistem harus menerapkan pembatasan hak akses halaman (<i>middleware</i>) untuk memastikan teknisi tidak dapat membuka halaman manajemen milik admin.
Keandalan/ <i>Reliability</i>	1. Sistem harus tersedia dan dapat diakses 24 jam penuh setiap harinya guna mendukung operasional pencatatan gudang tanpa henti. 2. Sistem harus memiliki penanganan galat (<i>error handling</i>) yang baik, agar jika terjadi kegagalan transaksi, sistem tidak rusak (<i>crash</i>), melainkan menampilkan pesan peringatan yang jelas kepada pengguna.

Aspek	Kebutuhan
Kemudahan Penggunaan (<i>Usability</i>)	1. Antarmuka aplikasi <i>mobile</i> harus berpusat pada pengguna (<i>user-friendly</i>) dengan ukuran tombol dan teks yang proporsional agar mudah ditekan oleh teknisi di lapangan. 2. Antarmuka sistem <i>web admin</i> harus bersifat responsif (<i>responsive design</i>), sehingga tata letak laporan dan tabel tetap rapi ketika dibuka melalui monitor PC resolusi tinggi maupun layar laptop standar.

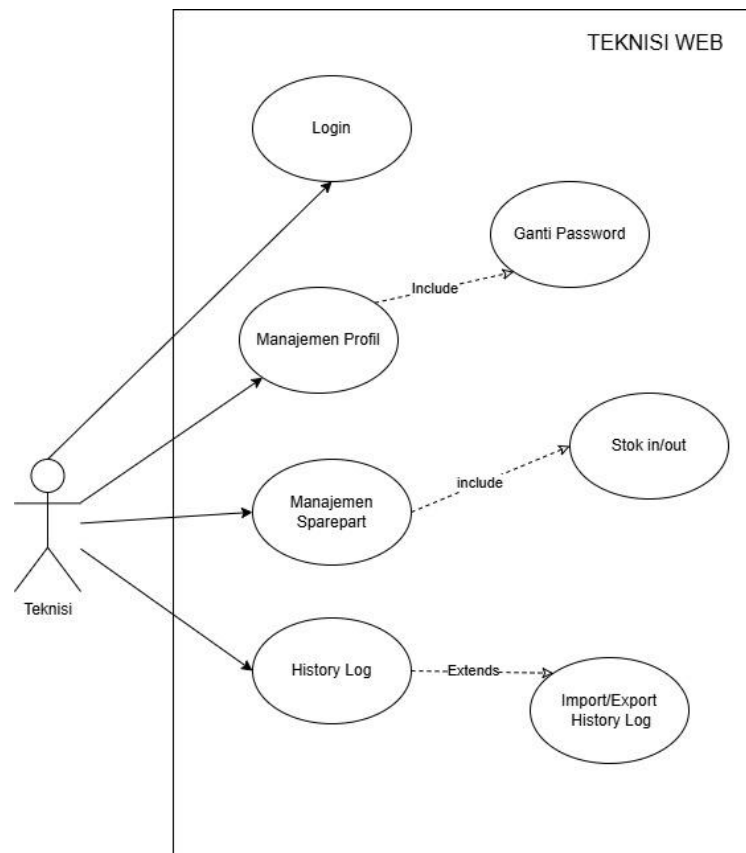
3.8.3. Use Case Diagram

Menggambarkan Aktor (Pengguna) yang terlibat di dalam fungsi suatu sistem yang dapat di lakukan, seperti input data baru, menambah atau mengurangi stok di dalam data, akses khusus untuk admin ke fungsionalitas tertentu, dan aktor pengguna *mobile*.



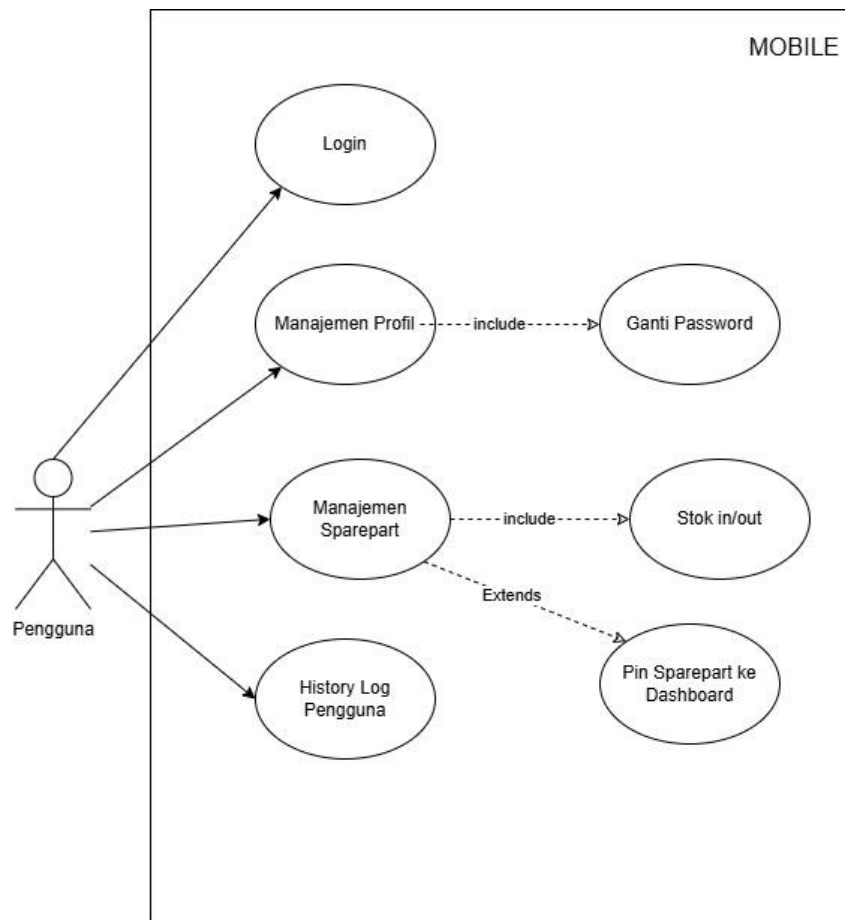
Gambar 3.8. Use Case diagram Admin Web

Gambar diatas menggambarkan interaksi fungsionalitas aktor Admin pada sistem informasi berbasis web. Admin diwajibkan melakukan *Login* untuk mengakses sistem. Setelah masuk, Admin memiliki hak akses penuh terhadap sistem, meliputi Manajemen Profil, Manajemen *Sparepart* yang mencakup *include* stok in/out, serta *extends* pada CRUD Sparepart dan *Import/Export* Sparepart, mengakses *History Log* dengan fitur untuk *Import/Export History Log*, serta mengakses Manajemen User dimana mencakup CRUD Users dan Role untuk mengatur akun karyawan.



Gambar 3.9. Use Case diagram Teknisi Web

Gambar diatas menjelaskan batasan fungsionalitas aktor Teknisi ketika mengakses sistem melalui platform web. Berbeda dengan Admin, hak akses Teknisi lebih dibatasi. Teknisi dapat melakukan Login dan Manajemen Profil seperti ganti password pada bagian operasional, Teknisi hanya dapat mengakses Manajemen Sparepart untuk melakukan pembaruan persediaan mutasi stok barang, serta melihat *history log* dengan opsi untuk mengekspor data riwayat mutasi. Teknisi tidak memiliki akses untuk menambah/menghapus data utama barang maupun mengelola akun.



Gambar 3.10. Use Case diagram Mobile

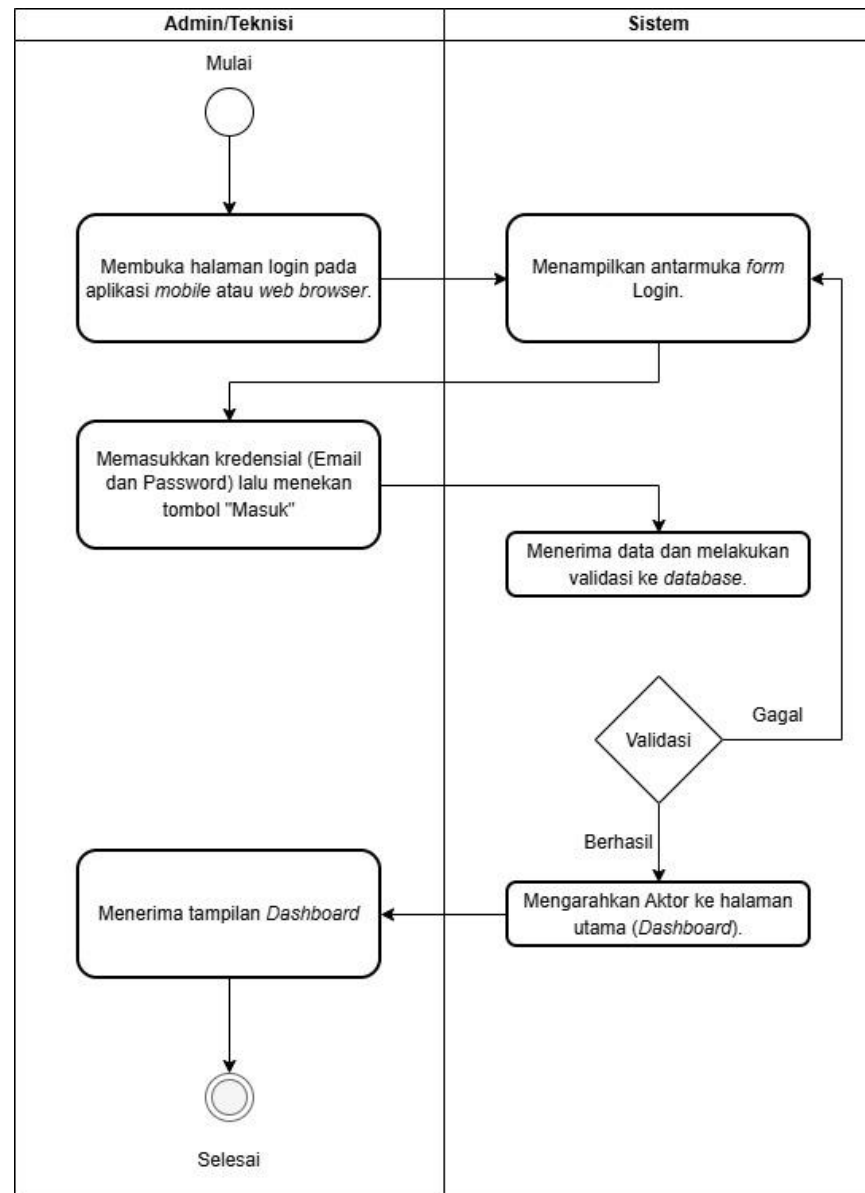
Gambar diatas melihat interaksi aktor Pengguna (Teknisi Lapangan) yang mengakses sistem melalui aplikasi *mobile*. Alur ini dirancang khusus untuk memfasilitasi mobilitas staf gudang. Setelah melakukan *Login*, pengguna dapat mengakses Manajemen Profil untuk mengonfirmasi dan mengganti kata sandi. Fungsi utama di lapangan adalah Manajemen Sparepart untuk melakukan pembaruan stok secara instan yang memiliki fungsi tambahan berupa fitur Pin *Sparepart* ke *Dashboard* agar barang tertentu mudah dicari. Selain itu, pengguna dapat memantau jejak aktivitas pribadinya melalui menu History Log Pengguna.

3.8.4. Activity Diagram

Digunakan untuk memodelkan alur kerja (*workflow*) atau langkah-langkah logika dalam suatu sistem atau program, dimana *activity* diagram

menggambarkan bagaimana sebuah sistem mengeksekusi fungsi atau tugas secara berurutan.

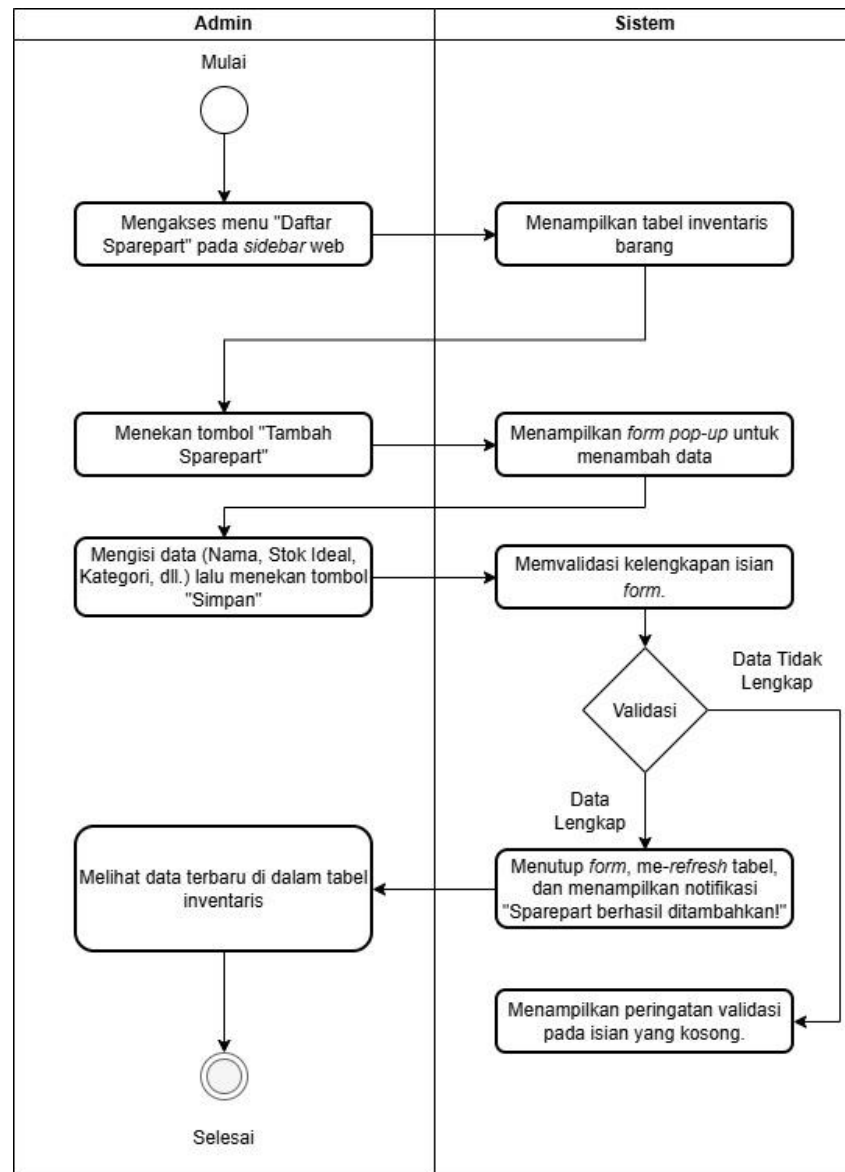
1. Activity Diagram Login



Gambar 3.11. Activity Diagram login

Diagram diatas ini menggambarkan alur pengguna (baik Teknisi di *mobile* maupun Admin di *web*) saat *login* ke dalam sistem.

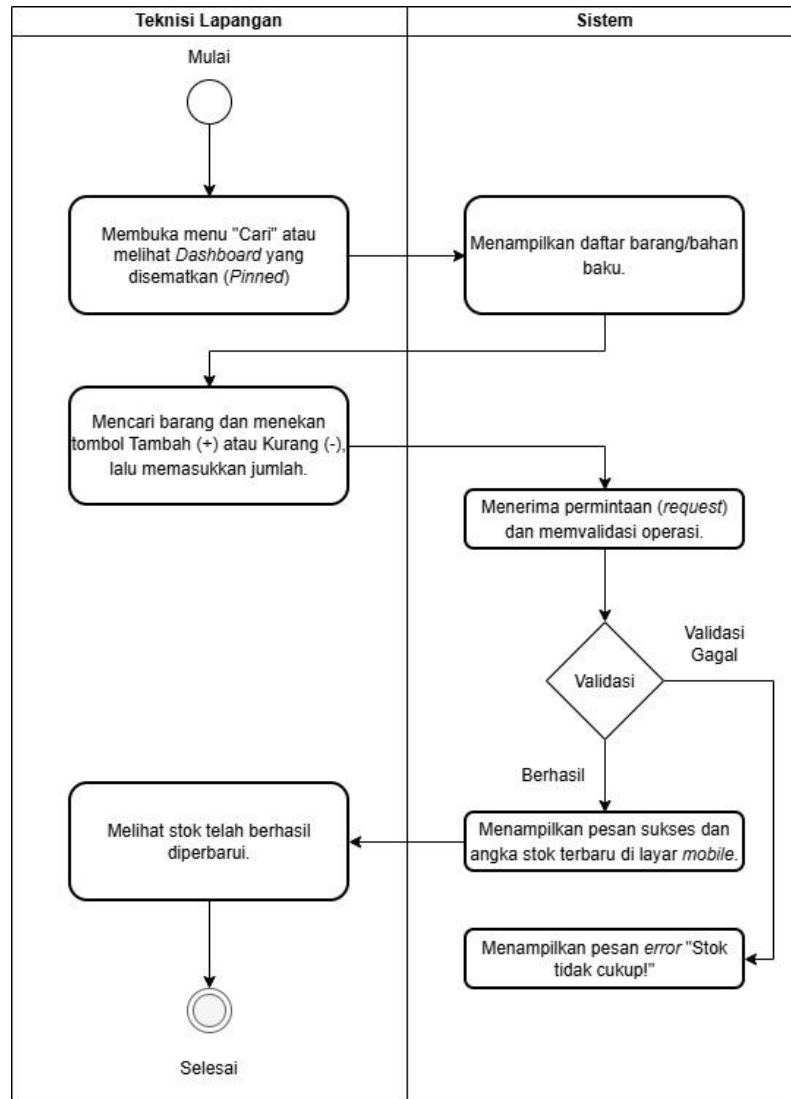
2. Activity Diagram Kelola Sparepart



Gambar 3.12. Activity Diagram Tambah bahan baku baru

Diagram diatas ini Diagram ini menggambarkan alur aktor admin dalam melakukan penambahan data *sparepart* baru ke dalam sistem.

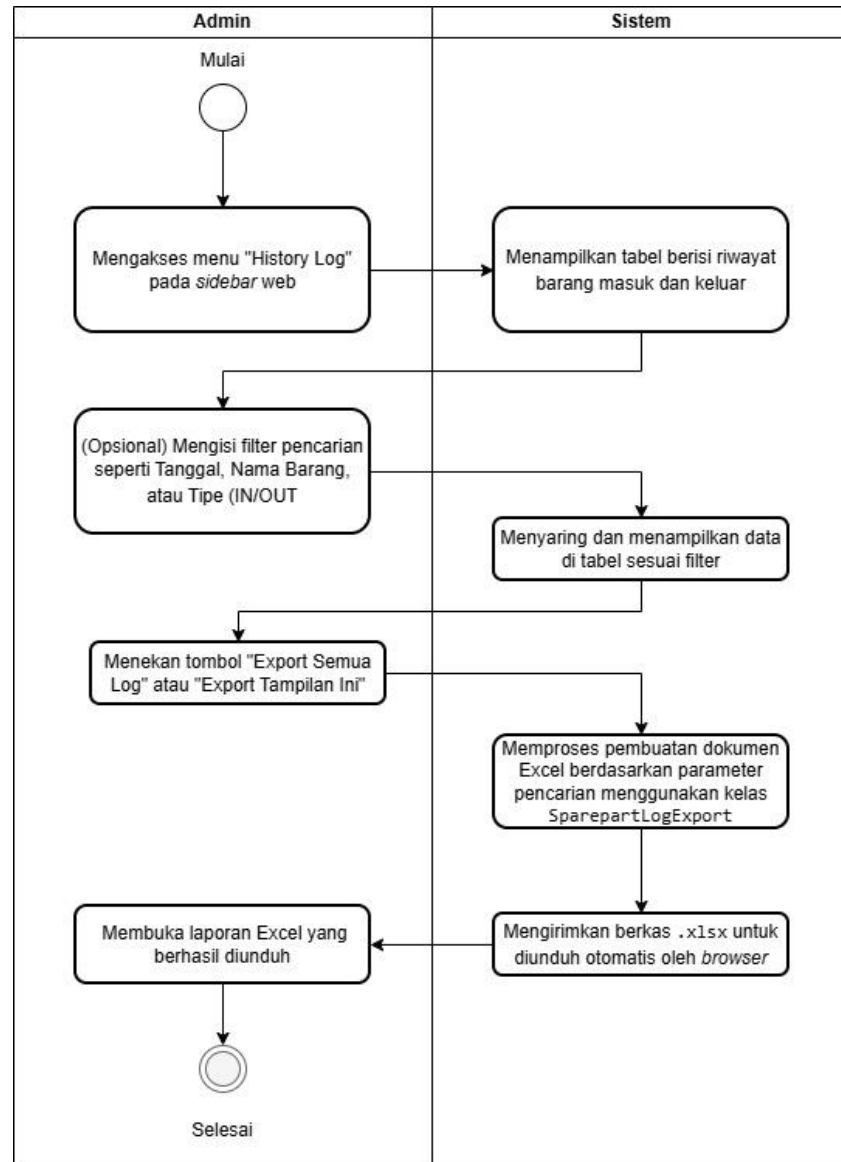
3. Activity Diagram Update Stok



Gambar 3.13. Activity Diagram update stok

Diagram diatas ini menggambarkan alur Teknisi di lapangan saat melakukan pembaruan stok fisik secara *real-time*.

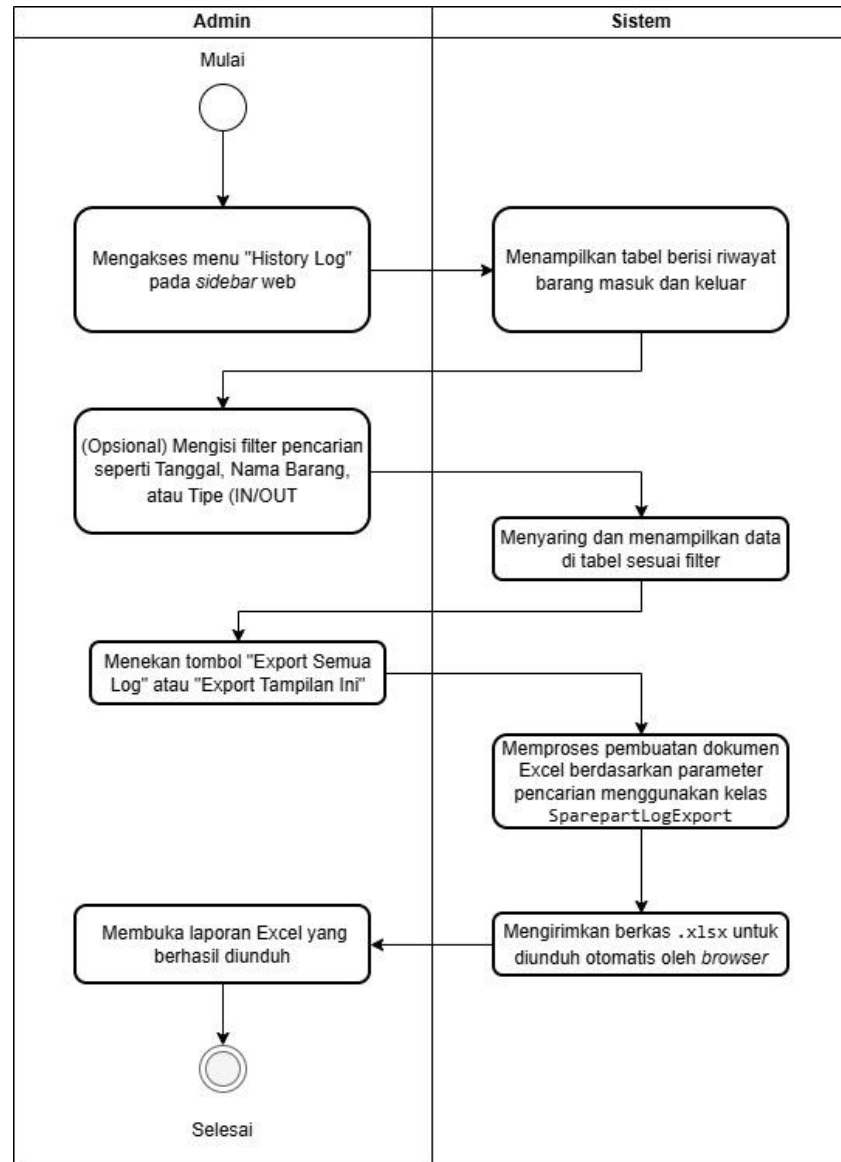
4. Activity Diagram Cetak Laporan



Gambar 3.14. Activity Diagram Cetak Laporan

Diagram diatas ini menggambarkan alur Admin saat akan mengekspor riwayat mutasi barang menjadi format Excel.

5. Activity Diagram Pembuatan Akun

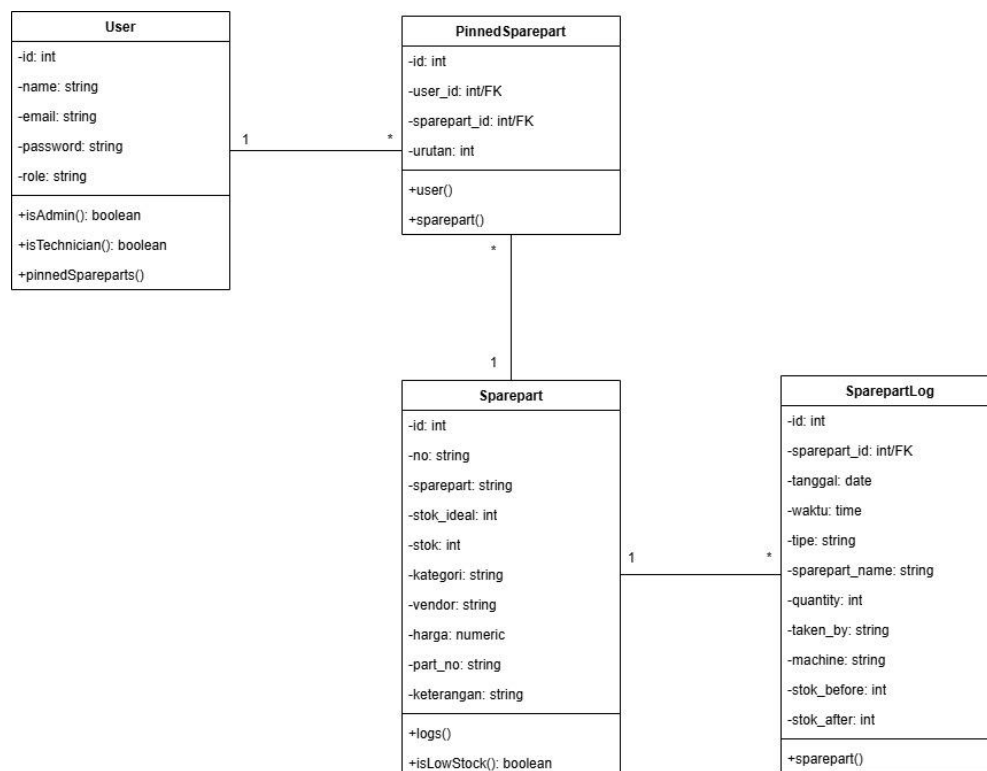


Gambar 3.15. Activity Diagram Pembuatan Akun

Diagram ini menggambarkan alur Admin dalam membuat akun baru (baik untuk admin lain maupun untuk teknisi lapangan) agar dapat mengakses sistem.

3.8.5. Class diagram

Class Diagram memodelkan struktur statis dari sistem perangkat lunak, menunjukkan kelas-kelas penyusun sistem, atribut (data), metode (operasi/fungsi), serta hubungan antar kelas tersebut. Pada sistem manajemen stok PT. Plastindo Pack Printing yang dibangun dengan kerangka kerja Laravel 12, *Class Diagram* difokuskan pada kelas *Model* yang merepresentasikan tabel-tabel di dalam basis data melalui *Eloquent ORM*.

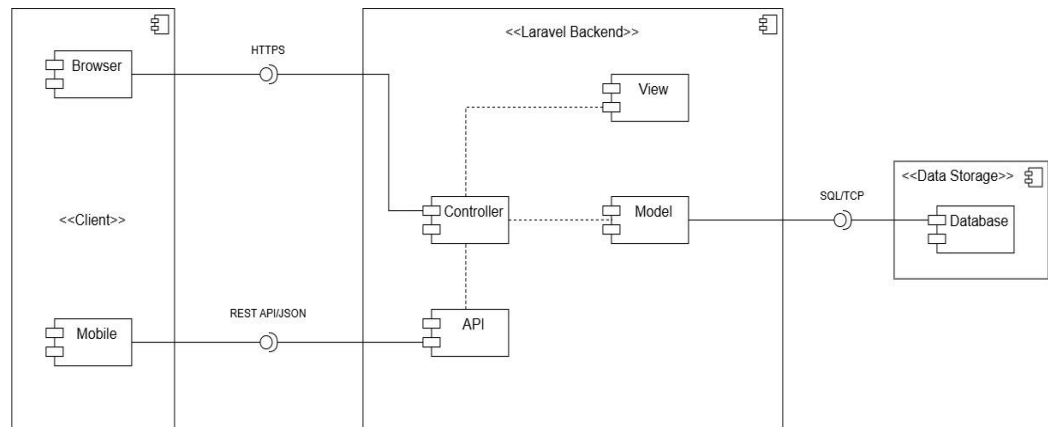


Gambar 3.16. *Class Diagram*

3.8.6. Component diagram

menggambarkan struktur fisik dan arsitektur dari sebuah sistem perangkat lunak. Diagram ini berfokus pada komponen-komponen perangkat lunak dan bagaimana mereka saling terhubung dan bergantung satu sama lain untuk membentuk sebuah sistem. Dalam konteks sistem terdistribusi, seperti sistem informasi manajemen stok yang dibangun dalam

penelitian ini, *component* diagram sangat krusial untuk memvisualisasikan pemisahan antara aplikasi klien (*frontend*), logika *backend*, dan penyimpanan data (*database*).



Gambar 3.17. *Component* Diagram sistem

Pada gambar diatas, arsitektur sistem informasi manajemen stok ini memiliki alur komunikasi yang terstruktur berdasarkan pemisahan antarmuka pengguna. Secara umum, aliran data (*data flow*) dan pemanggilan fungsi antar komponen diklasifikasikan menjadi dua alur utama, yaitu alur operasional melalui perangkat *mobile* dan alur manajerial *browser*. Berikut adalah rincian fungsionalitas dan ketergantungan (*dependency*) dari masing-masing komponen:

Tabel 3.13. Rincian fungsionalitas *Component* Diagram

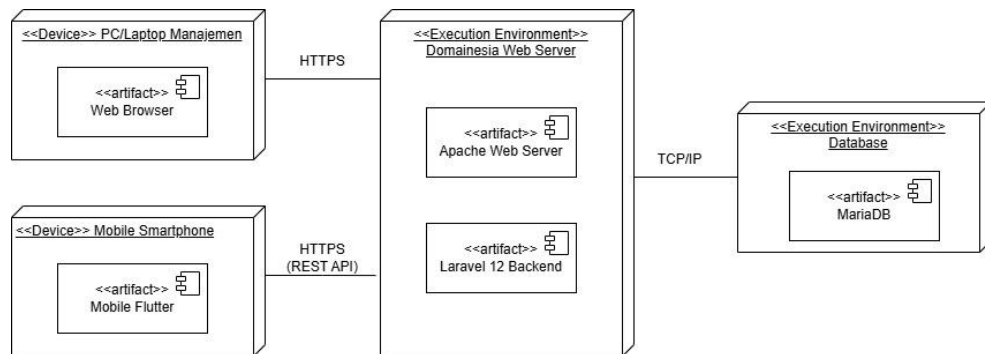
Nama Komponen	Fungsi dan Keterangan Arsitektur
<<Client>> • Browser • Mobile	Node ini mewakili perangkat pengguna akhir. 1. Browser digunakan oleh pihak manajemen (admin) untuk mengakses halaman web. Komponen ini memiliki <i>required interface</i> untuk meminta halaman web ke <i>backend</i> melalui protokol komunikasi HTTPS. 2. Mobile adalah aplikasi berbasis Flutter yang digunakan oleh teknisi atau karyawan gudang di lapangan. Komponen ini tidak berinteraksi dengan

Nama Komponen	Fungsi dan Keterangan Arsitektur
	antarmuka web, melainkan mengonsumsi data mentah menggunakan protokol komunikasi REST API/JSON.
<<Laravel Backend>> • API • Controller • Model • View	Node ini adalah pusat pemrosesan logika bisnis yang dibangun menggunakan framework Laravel 12. Arsitekturnya mematuhi pola Model-View-Controller (MVC) 1. API menyediakan jalur komunikasi khusus bagi aplikasi <i>mobile</i>. Permintaan dari API kemudian diteruskan secara internal ke <i>Controller</i>. 2. <i>Controller</i> berfungsi sebagai otak sistem yang mengatur lalu lintas data. Jika menerima permintaan dari <i>Browser</i>, <i>Controller</i> akan memproses data dengan bantuan dari Model lalu mengirimkannya ke View untuk dirakit menjadi halaman antarmuka pengguna. 3. Model merupakan representasi dari tabel <i>database</i>. Semua akses untuk membaca atau menulis data stok harus melewati komponen Model ini, yang terhubung ke luar <i>node</i> melalui protokol SQL/TCP.
<<Data Storage>> • Database	Node terpisah ini menampung komponen <i>Database</i> (menggunakan MariaDB). Pemisahan <i>database</i> sebagai <i>node</i> tersendiri menunjukkan bahwa penyimpanan data bersifat persisten dan independen dari logika <i>backend</i> , serta memastikan keamanan dan stabilitas sistem.

3.8.7. Deployment diagram

jenis diagram UML yang menggambarkan arsitektur fisik tempat sistem perangkat lunak diimplementasikan. Jika component diagram berfokus pada logika modular software, maka deployment diagram berfokus pada infrastruktur perangkat keras (*hardware*) atau lingkungan eksekusi (*server*) yang menampung komponen-komponen tersebut.

Diagram ini sangat penting dalam perancangan sistem berbasis cloud dan mobile di PT. Plastindo Pack Printing untuk memetakan bagaimana perangkat pekerja lapangan, komputer manajemen, dan server hosting saling terhubung melalui jaringan internet.



Gambar 3.18. Deployment diagram sistem

Berdasarkan rancangan sistem yang telah disesuaikan dengan kebutuhan operasional, arsitektur fisik sistem dibagi menjadi empat *Node* utama. Setiap *node* mewakili lingkungan komputasi yang menampung artefak perangkat lunak (*artifact*) tertentu. Rincian distribusi *node* dan artefaknya pada tabel dibawah ini:

Tabel 3.14. Rincian Fungsionalitas *Deployment* Diagram

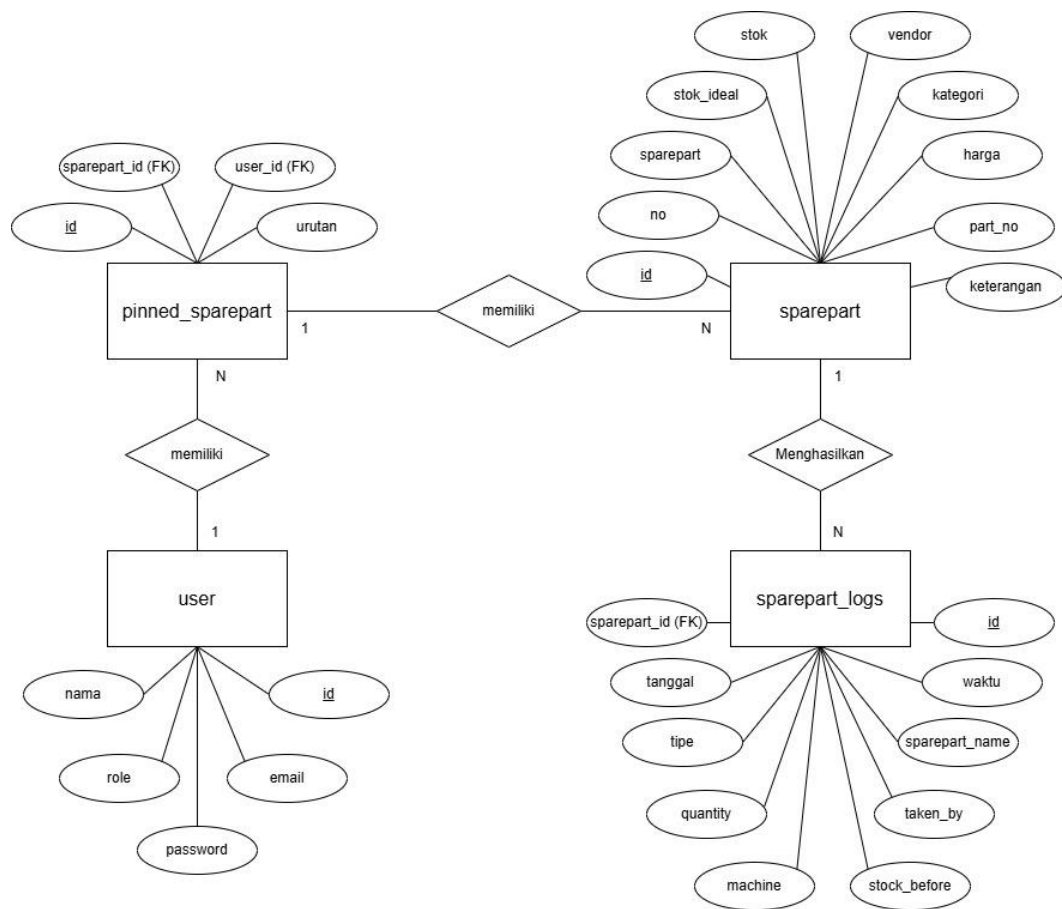
Nama Node	Perangkat Lunak (Artifact)	Keterangan dan fungsi
<<Device>> Smartphone	<<artifact>> Aplikasi Mobile Flutter (.apk)	Perangkat <i>mobile</i> (seperti HP Android) yang dipegang langsung oleh staf gudang di lantai produksi. Berfungsi sebagai klien utama untuk pemindaian dan penginputan sirkulasi barang secara <i>mobile</i> .
<<Device>> PC/Laptop Manajemen	<<artifact>> Aplikasi Mobile Flutter (.apk)	Komputer fisik yang berada di ruang kantor manajemen. Digunakan oleh administrator untuk mengakses halaman utama laporan stok secara visual, menambahkan bahan baku baru, dan mengelola akun pengguna.
<<Execution Environment>> Web Server	<<artifact>> Laravel 12 Backend ————— <<artifact>> Web Server Service (Apache)	Lingkungan cloud hosting (dalam penelitian ini menggunakan layanan DomaiNesia) yang menyajikan logika sistem utama. Menampung seluruh controller dan API yang memproses permintaan dari klien <i>mobile</i> dan web.
<<Execution Environment>> Database	<<artifact>> MariaDB Service	Lingkungan basis data relasional yang berfungsi sebagai pusat penyimpanan seluruh rekaman data stok, riwayat aktivitas barang masuk/keluar, dan data autentikasi pengguna secara permanen.

Ketiga perangkat tersebut saling terhubung membentuk arsitektur *client-server* yang solid. Untuk menghubungkan entitas fisik tersebut, didefinisikan jalur komunikasi sebagai berikut:

1. HTTPS / REST API merupakan jalur komunikasi nirkabel (internet) yang menghubungkan *Node Smartphone* Pengguna dengan *Node Web Server*. Jalur ini digunakan aplikasi Flutter untuk mengirim dan menarik data JSON secara aman dan cepat.
2. HTTPS merupakan jalur internet standar yang menghubungkan *Node PC Manajemen* dengan *Node Web Server*, memungkinkan administrator untuk memuat halaman antarmuka web (HTML) secara terenkripsi.
3. TCP/IP (SQL) merupakan jalur komunikasi internal (*server*) yang menghubungkan *Web Server* dengan *Database Server*. Jalur ini hanya dapat diakses oleh sistem *backend* (Laravel) untuk mengeksekusi kueri manajemen data (CRUD) secara privat, memastikan basis data terlindungi dari akses langsung dari luar.

3.8.8. ERD

Entity Relationship Diagram (ERD) adalah salah satu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. ERD digunakan untuk memodelkan struktur data dan hubungan antar data dalam basis data MySQL yang digunakan oleh sistem. Pada sistem informasi manajemen stok ini, relasi antar entitas dirancang untuk mendukung pencatatan sirkulasi barang yang terintegrasi antara aplikasi *mobile* dan *web admin*.



Gambar 3.19. *Entity Relationships* Diagram

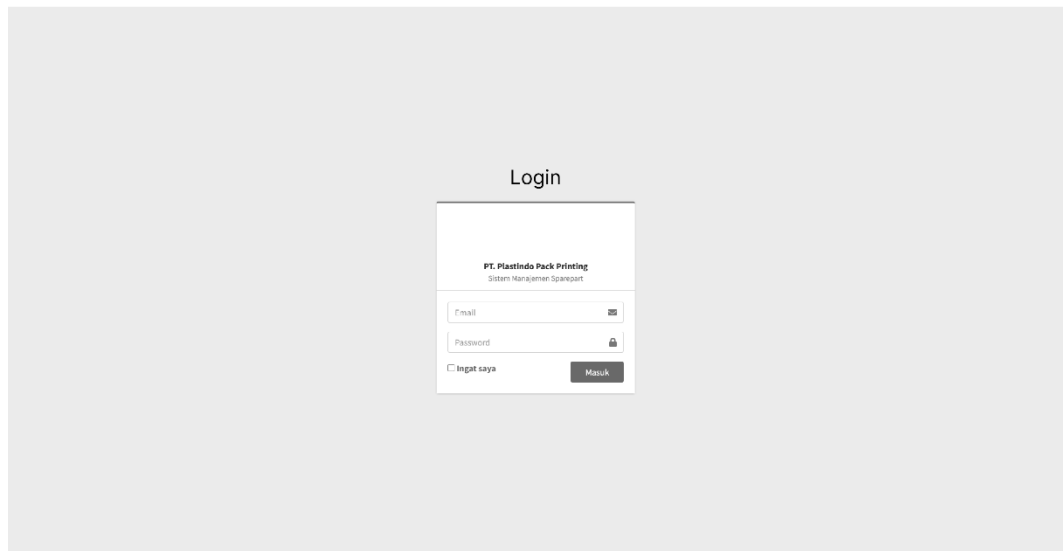
Gambar diatas menunjukkan rancangan ERD yang terdiri dari empat entitas utama, yaitu:

1. Entitas user menyimpan data profil dan kredensial pengguna.
2. Entitas sparepart berfungsi untuk menyimpan master data persediaan bahan baku secara detail.
3. Entitas pinned_sparepart dimana dia berfungsi sebagai tabel perantara yang menghubungkan entitas user dengan sparepart untuk memfasilitasi fitur penyematan atau pin barang pada aplikasi mobile.
4. Entitas sparepart_logs: Bertindak sebagai tabel riwayat operasional yang mencatat setiap aktivitas mutasi pada entitas sparepart.

3.8.9. Perancangan Antarmuka (*Mockup*)

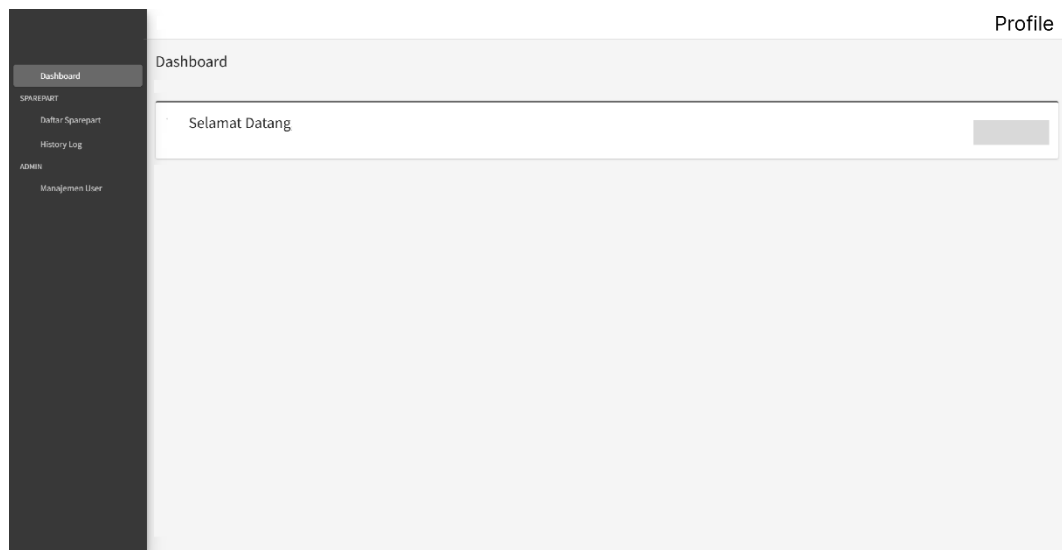
1. *Mockup* Sistem Berbasis Web

Berikut adalah rancangan tampilan antarmuka dari Sistem pergudangan PT Plastindo Pack Printing Berbasis Web:



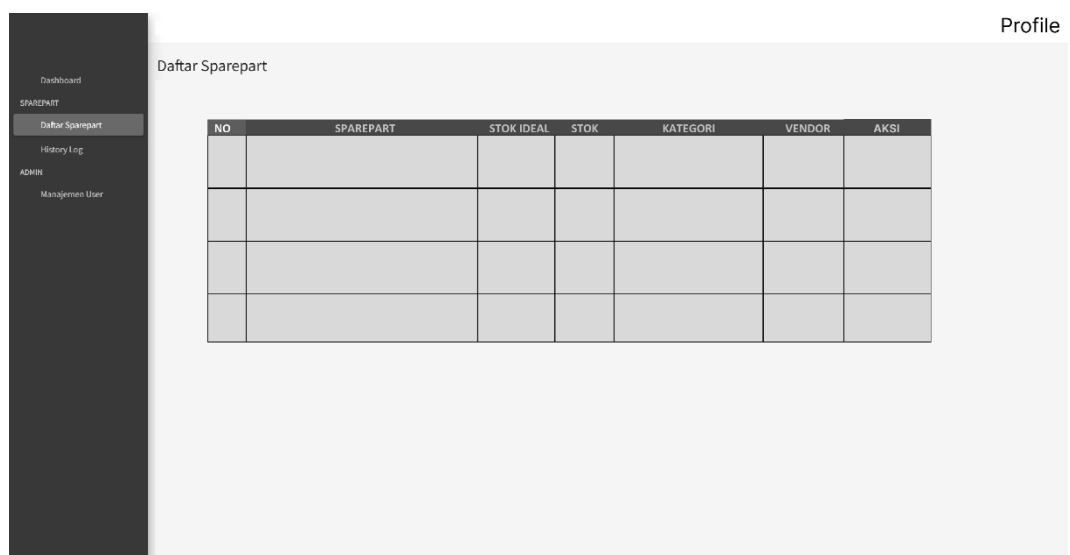
Gambar 3.20. Rancangan Halaman Login

Antarmuka halaman *login* dirancang sebagai pintu gerbang utama untuk menjaga keamanan akses *dashboard* berbasis *web*. Pada halaman ini, pengguna diwajibkan untuk memasukkan kredensial berupa *email* dan *password* yang valid agar dapat diotentikasi dan masuk ke dalam sistem manajemen.



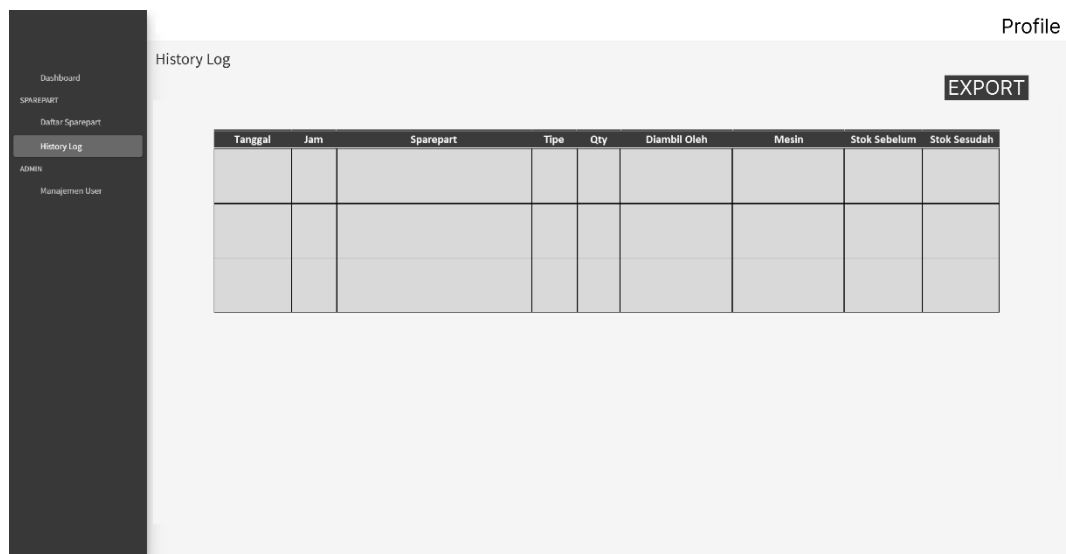
Gambar 3.21. Rancangan Halaman Dashboard

Halaman *dashboard* manajerial berfungsi sebagai pusat pemantauan utama bagi administrator. Antarmuka ini dirancang untuk menampilkan ringkasan data statistik secara *real-time*, yang meliputi total item keseluruhan, peringatan visual untuk stok yang menipis (*low stock*), serta metrik angka mutasi barang masuk dan keluar pada hari tersebut guna mempermudah pengambilan keputusan.



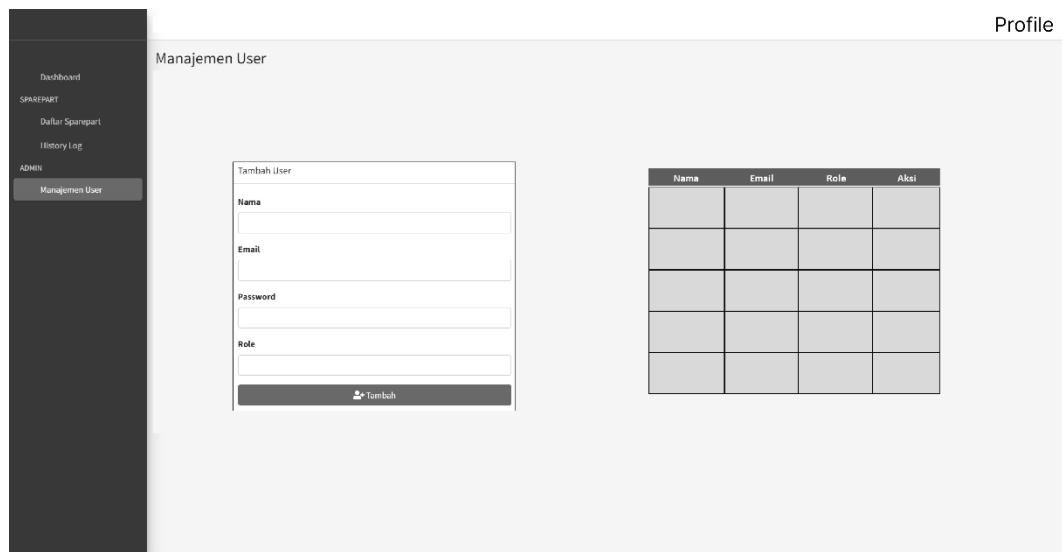
Gambar 3.22. Rancangan Halaman Manajemen Sparepart

Antarmuka ini dikembangkan untuk memfasilitasi pengelolaan data induk *sparepart* dan bahan baku secara menyeluruh. Melalui halaman ini, administrator dapat melakukan operasi penambahan data, pengubahan, penghapusan, memfilter pencarian berdasarkan kategori, serta mengeksekusi proses *import* dan *export* data menggunakan format dokumen Excel.



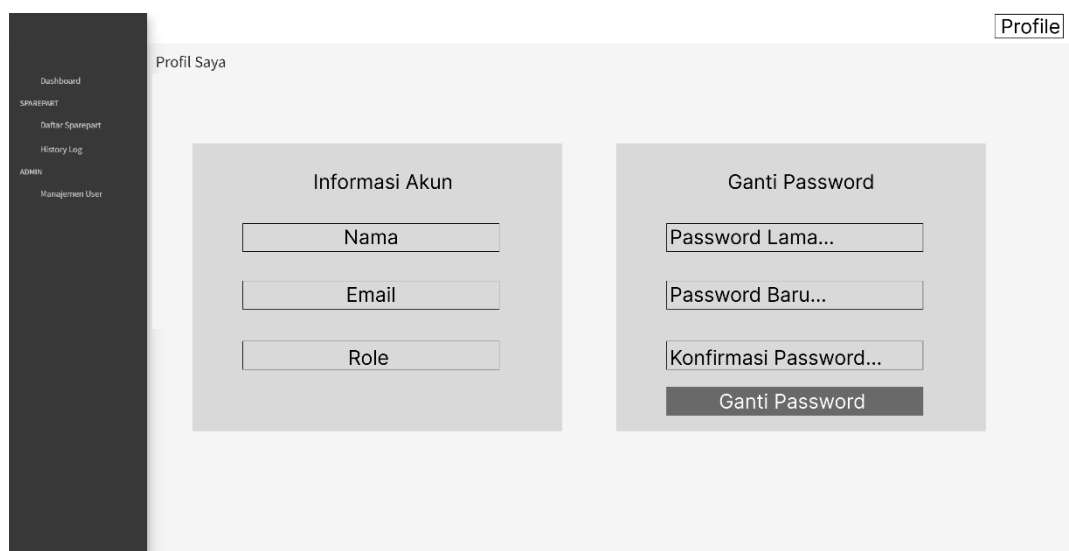
Gambar 3.23. Rancangan Halaman Riwayat

Halaman *History Log* menyajikan antarmuka berupa tabel buku besar (*ledger*) yang mencatat secara detail seluruh riwayat aktivitas sirkulasi barang masuk (IN) maupun keluar (OUT).



Gambar 3.24. Rancangan Halaman Manajemen Akun

Antarmuka manajemen *user* merupakan halaman dengan hak akses eksklusif yang hanya dapat dibuka oleh administrator tingkat atas atau pihak manajemen. Halaman ini digunakan untuk mengontrol sistem dengan cara menambah akun pengguna baru, mengedit data kredensial, serta menetapkan hak akses (*role*) apakah pengguna tersebut bertindak sebagai admin atau teknisi lapangan.

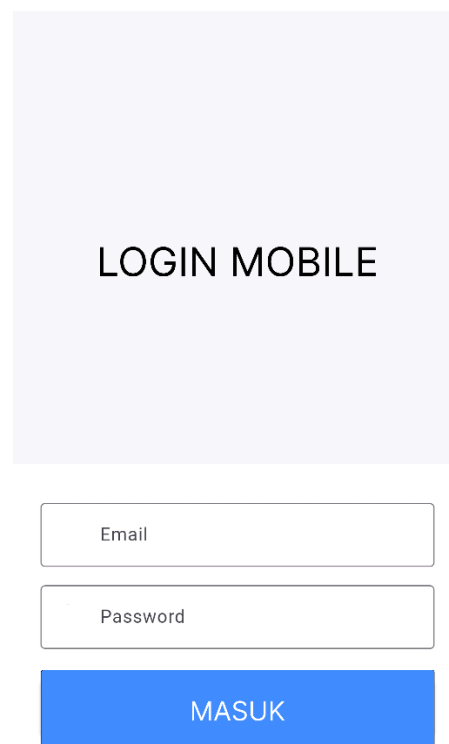


Gambar 3.25. Rancangan Halaman Profil

Halaman profil berfungsi untuk menampilkan rincian informasi dari akun yang sedang aktif, seperti nama, *email*, dan status peran (*role*) pengguna. Selain itu, antarmuka ini menyediakan formulir khusus bagi pengguna untuk melakukan pembaruan kata sandi (*password*) secara mandiri demi menjaga keamanan sistem.

2. *Mockup* Sistem Berbasis Mobile

Berikut adalah rancangan tampilan antarmuka dari Sistem pergudangan PT Plastindo Pack Printing Berbasis Mobile



LOGIN MOBILE

Email

Password

MASUK

Gambar 3.26. Rancangan Halaman Login Mobile

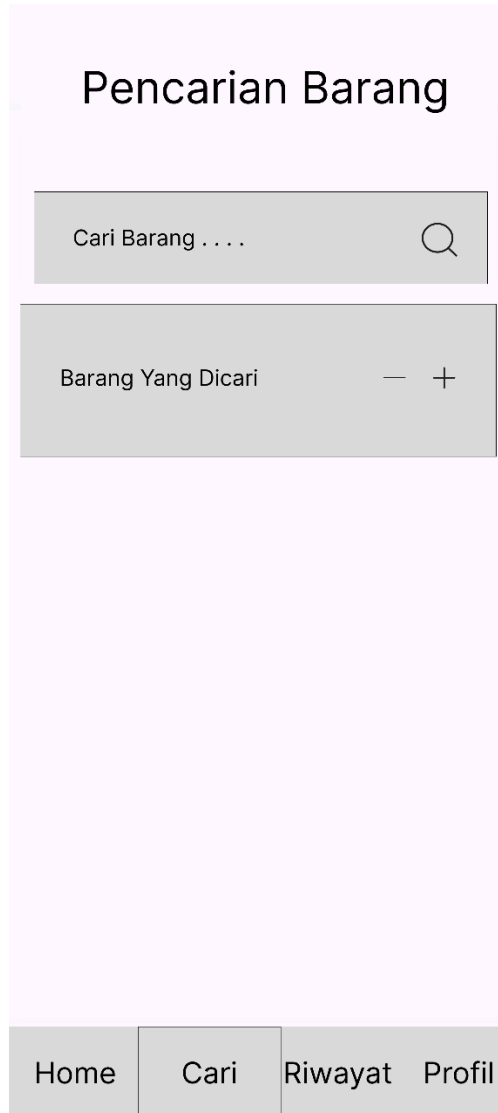
Antarmuka *login* pada aplikasi *mobile* dirancang dengan tampilan yang bersih, berpusat pada pengguna, dan sangat sederhana. Teknisi lapangan hanya perlu memasukkan *email* dan *password* untuk mendapatkan otorisasi akses berbasis token ke dalam sistem operasional aplikasi.



Gambar 3.27. Rancangan Halaman Utama Mobile

Antarmuka pencarian dirancang untuk membantu pekerja lapangan menemukan detail *sparepart* atau bahan baku secara cepat di dalam area gudang. Melalui halaman ini pula, teknisi dapat memilih dan menyematkan

barang-barang tertentu agar tampil sebagai akses cepat di halaman *dashboard* utama mereka.



Gambar 3.28. Rancangan Halaman Cari Sparepart Mobile

Antarmuka pencarian dirancang untuk membantu pekerja lapangan menemukan detail *sparepart* atau bahan baku secara cepat di dalam area gudang.



Gambar 3.29. Rancangan Halaman Riwayat Mobile

Halaman riwayat pada perangkat *mobile* menyajikan rekam jejak aktivitas mutasi stok yang bersifat personal. Antarmuka ini difokuskan untuk menampilkan riwayat barang masuk dan keluar yang hanya dieksekusi oleh akun teknisi yang saat itu sedang masuk ke dalam sistem.

The image shows a mobile application interface for a profile page. At the top, the word "Profil" is centered. Below it is a grey box containing the text "Nama User Login". Further down is a larger grey box titled "Ganti Password" which contains three input fields: "Password Lama", "Password Baru", and "Konfirmasi Password Baru". Below these fields is a blue button labeled "Simpan". At the bottom of the screen is a grey navigation bar with four options: "Home", "Cari", "Riwayat", and "Profil", with "Profil" being the active page.

Gambar 3.30. Rancangan Halaman Profil Mobile

Antarmuka profil pada aplikasi *mobile* merangkum identitas dan informasi kredensial dari teknisi. Halaman ini memberikan fleksibilitas fungsional bagi teknisi lapangan untuk mengganti dan mengonfirmasi kata sandi baru mereka secara mudah langsung melalui perangkat genggam.

3.9 Jadwal Penelitian

Jadwal penelitian disusun sebagai pedoman waktu untuk memastikan setiap tahapan, mulai dari identifikasi masalah hingga sidang tugas akhir, dapat diselesaikan secara terorganisir dan tepat waktu. Penelitian ini direncanakan

berlangsung selama empat bulan, terhitung mulai dari bulan April 2026 hingga Juli 2026.

Rincian agenda kegiatan penelitian disajikan dalam tabel berikut:

Tabel 3.15. Jadwal Penelitian.

No	Kegiatan	Apr	Mei	Jun	Jul	
1	Penyusunan & Seminar Proposal					
2	Identifikasi Masalah & Pengumpulan Data Awal					
3	Analisis Kebutuhan & Perancangan Prototype Awal					
4	Evaluasi Iterasi Prototype & Pengembangan Sistem					
5	Pengujian Sistem (Black-box & UAT)					
6	Penyusunan Laporan Tugas Akhir & Seminar Hasil					
7	Sidang Tugas Akhir					