

BAB III PELAKSANAAN KERJA

3.1 Bidang Kerja

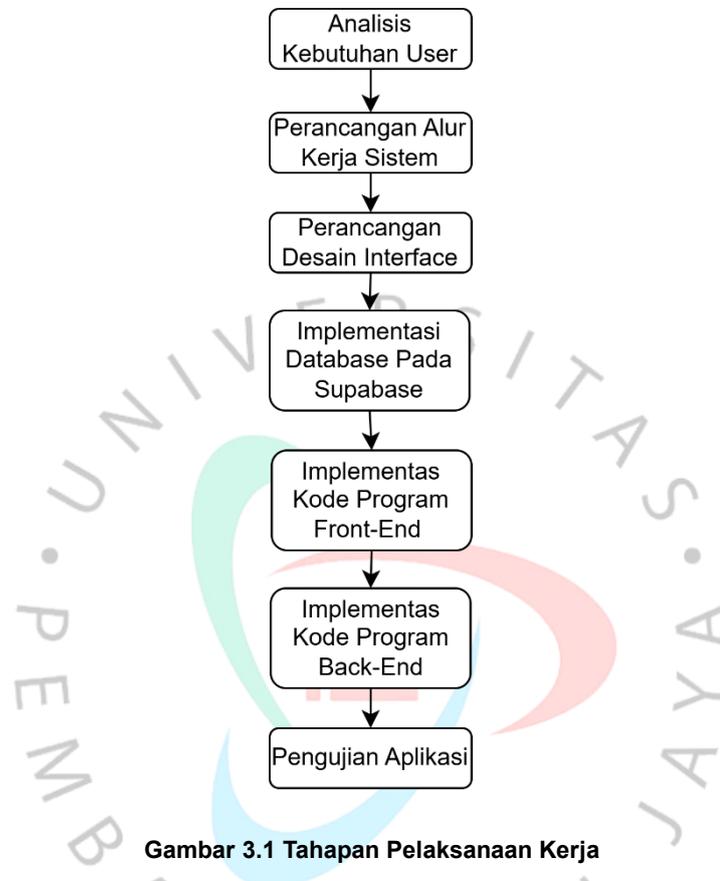
Selama menjalankan program kerja profesi di PT Brainlens, ditempatkan sebagai *programmer* bagian *Front-End* di divisi pengembangan. Divisi ini memiliki peran mengembangkan dan mengimplementasikan hasil analisis kode program untuk memastikan aplikasi atau website dapat berjalan dengan baik dalam melakukan pengujian pada aplikasi atau website, salah satunya dengan pengembangan aplikasi pengolahan data barang logistic event berbasis website.

Aplikasi ini dikembangkan dengan tujuan untuk mengelola data barang logistik event berbasis *website*. Aplikasi ini menggunakan *framework Next.js* untuk mengembangkan web yang cepat dan efisien, serta menggunakan *Tailwind CSS* untuk desain antarmuka pengguna yang responsif dan modern. Untuk penyimpanan data, aplikasi ini menggunakan *Supabase* sebagai *Back-end* database yang memungkinkan manajemen data secara *real-time* dan mudah diintegrasikan dengan *Front-end* yang ada. Pada aplikasi berbasis *website* tersebut diminta untuk mengadakan beberapa fitur, seperti fitur untuk menyimpan data barang yang dibeli ataupun sewa, mengelola detail barang tersebut, dan *scanning QR Code* untuk pencarian data barang.

Dalam pengembangan aplikasi ini, praktikan berperan aktif dalam pembuatan dan mengintegrasikan berbagai fitur *Front-end*, serta memastikan tampilan aplikasi data digunakan dengan baik oleh pengguna. Praktikan bersama dengan rekan kerja diminta untuk mempelajari *framework* serta platform pendukung perancangan aplikasi yang nantinya akan digunakan, agar proyek yang dikerjakan menggunakan *framework* ataupun *platform* pendukung yang biasa digunakan oleh perusahaan.

3.2 Pelaksanaan Kerja

Berikut adalah tahap pelaksanaan kerja yang digambarkan melalui diagram alir.



Gambar 3.1 Tahapan Pelaksanaan Kerja

Dalam pelaksanaan kerja proyek aplikasi berbasis website, terdapat beberapa langkah yang harus dilakukan secara terarah untuk mencapai hasil yang terbaik. Berdasarkan tahap pelaksanaan kerja diatas memperlihatkan seluruh tahapan pelaksanaan pembuatan aplikasi berbasis *website*. Untuk menyelesaikan proyek tersebut, praktikan bekerja sama dengan rekan kerja saling membagi dan berdiskusi mengenai pembagian pekerjaan dan hasil diskusi tersebut menetapkan bahwa praktikan akan bekerja di bagian analisis, serta implementasi kode program *Front-End*.

3.2.1 Analisis Kebutuhan User

Pada tahap ini, fokus utama adalah mengidentifikasi kebutuhan pengguna aplikasi untuk memastikan aplikasi yang dikembangkan dapat memenuhi tujuan dan permintaan user. Tahapan ini terdiri dari beberapa langkah penting, sebagai berikut:

1) Identifikasi Stakeholder

Aplikasi pengelola data barang logistik ini akan digunakan oleh panitia *event* khususnya di divisi logistik, tim logistik akan menggunakan aplikasi ini sebagai tempat penyimpanan data barang yang akan dibeli ataupun sewa serta bisa mengelolanya, tujuannya adalah untuk memudahkan tim logistik menyimpan dan mengelola data barang dengan lebih efisien.

2) Spesifikasi Kebutuhan User

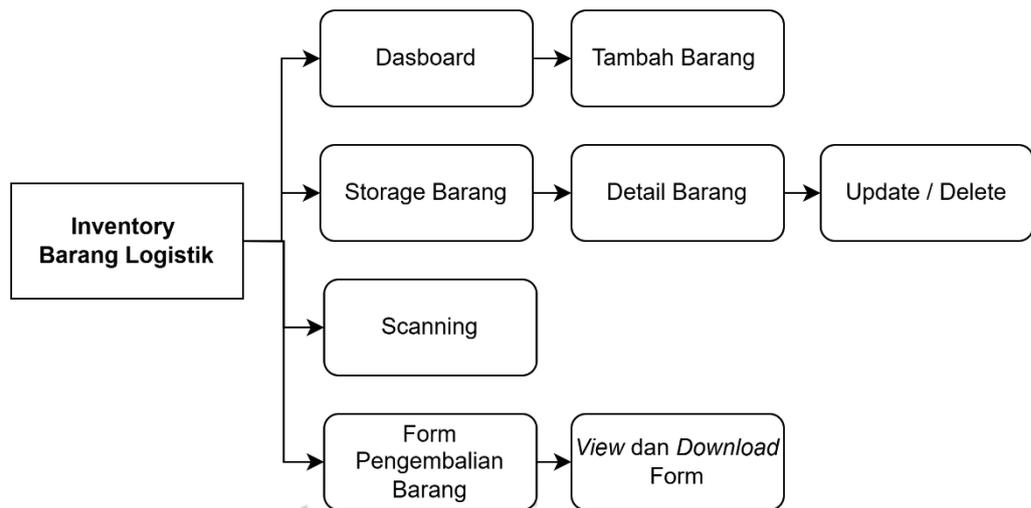
User diharuskan untuk dapat menambahkan, mengedit, dan menghapus data barang, serta dapat melihat informasi barang secara rinci. *User* juga dapat melakukan *scanning* pada *QR Code* yang telah dibuat dan ditempelkan pada barang untuk mempermudah pencarian barang di aplikasi.

3) Spesifikasi Kebutuhan User Interface

Pada saat membangun aplikasi berbasis *website*, praktikan diharapkan dapat membuat antarmuka yang mudah digunakan dan *mobile friendly* untuk memudahkan *user* dalam menggunakan aplikasi tersebut di lapangan.

3.2.2 Perancangan Alur Kerja Sistem

Setelah melakukan analisis kebutuhan untuk pengembangan aplikasi berbasis *website*, praktikan merancang fitur serta sistem dari aplikasi tersebut sesuai dengan hasil analisis yang telah dilakukan. Perancangan ini akan menjadi dasar untuk pengembangan sistem berikutnya.

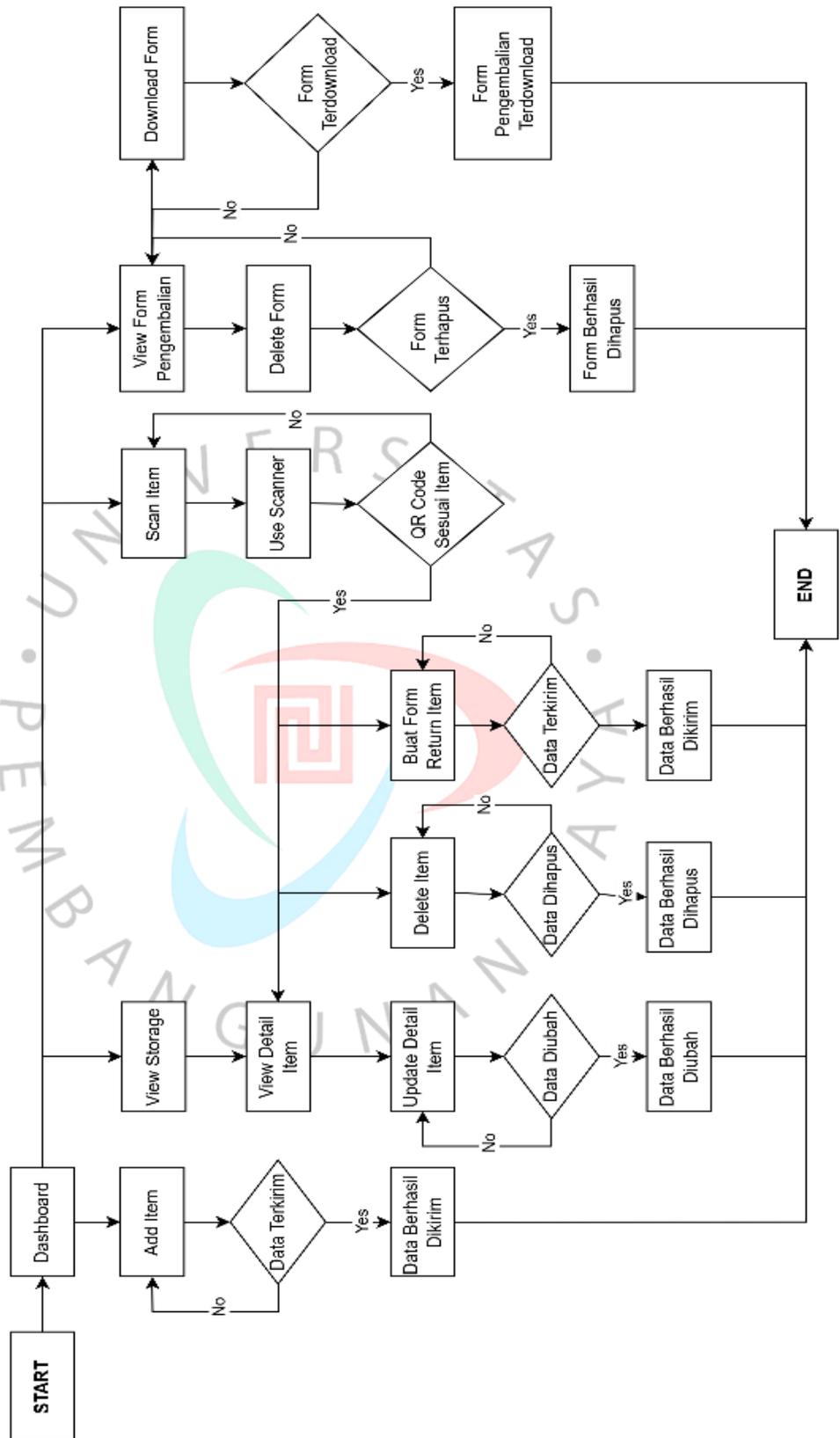


Gambar 3.2 Struktur Aplikasi

Gambar di atas merupakan gambaran struktur aplikasi yang terdapat pada aplikasi pengelolaan data barang logistik. Pada struktur aplikasi ini, terdapat beberapa halaman yang dirancang untuk dikembangkan menjadi desain *flowchart*, desain *Mockup*, dan perancangan *database*. *Flowchart* adalah diagram yang digunakan untuk menggambarkan alur atau proses dari suatu aplikasi secara grafis. Desain *Mockup* merupakan representasi visual yang menampilkan tampilan dan fungsi utama antarmuka dari aplikasi tersebut. Perancangan *database* adalah proses untuk menetapkan struktur, tabel, hubungan, dan elemen data lainnya untuk memenuhi kebutuhan aplikasi tersebut.

3.2.2.1 *Flowchart*

Pada fase perancangan, setelah praktikan melakukan analisis seluruh kebutuhan yang diperlukan dalam membangun aplikasi ini, langkah selanjutnya yaitu merancang sebuah *flowchart* agar dapat memvisualisasikan alur atau proses kerja aplikasi secara grafis.



Gambar 3.3 Desain *Flowchart* Aplikasi

Gambar di atas menunjukkan desain flowchart aplikasi yang akan menjadi panduan dalam pengembangan aplikasi. Sebelum memasuki tahap pembuatan Mockup dan implementasi, flowchart berperan sebagai dasar dalam menentukan alur kerja aplikasi. Berdasarkan flowchart tersebut, terdapat beberapa aktivitas yang dapat dilakukan oleh pengguna:

1. Menambahkan Data Barang

Ketika pengguna ingin menambahkan barang, mereka akan diarahkan ke halaman pengisian data barang. Setelah semua data diisi dengan lengkap dan benar, informasi tersebut akan disimpan ke dalam database dan ditampilkan pada halaman *storage*.

2. Mengubah atau Menghapus Data Barang

Jika pengguna ingin mengedit data barang yang telah tersimpan, mereka dapat memilih data tersebut di halaman *storage*. Selanjutnya, pengguna akan diarahkan ke halaman detail barang, di mana mereka dapat mengubah atau menghapus data sesuai kebutuhan.

3. Mencari Data Barang dengan Fitur Scan Item

Untuk mencari data barang, pengguna dapat menggunakan fitur *scan item*. Pada halaman ini, pengguna harus memberikan izin penggunaan kamera perangkat mereka. Setelah itu, mereka dapat memindai QR Code pada barang yang datanya ingin diketahui. Jika pemindaian berhasil, pengguna akan diarahkan ke halaman detail barang terkait.

4. Membuat Formulir Pengembalian Barang Sewa

Ketika pengguna ingin membuat formulir pengembalian barang sewa, mereka dapat menekan tombol formulir pengembalian pada halaman detail barang. Pengguna kemudian diarahkan ke halaman pengisian data untuk mencatat tanggal pengembalian. Setelah data diisi

dengan lengkap dan benar, informasi tersebut akan disimpan di database.

5. Melihat dan Mengelola Folmulir Pengembalian

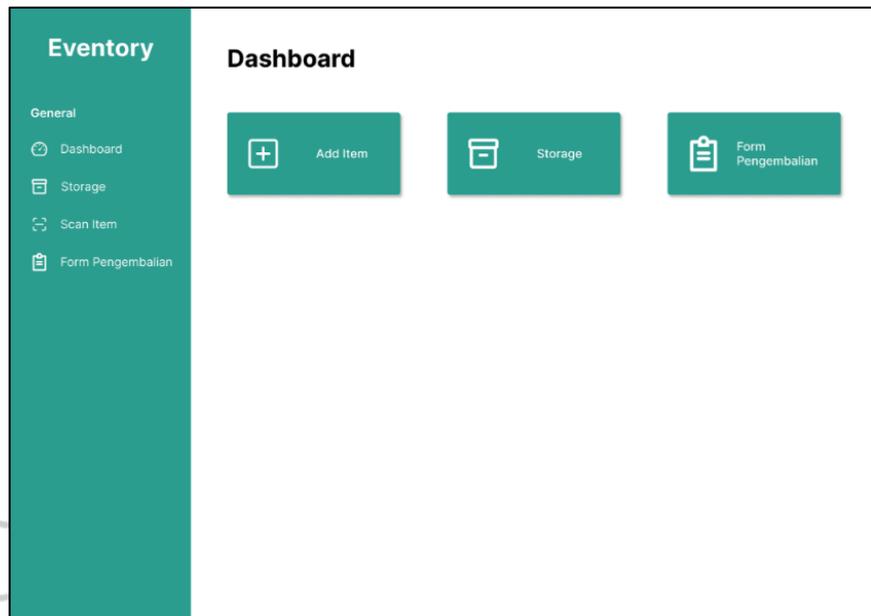
Pada halaman formulir pengembalian, pengguna dapat melihat data barang yang telah dikirim sebelumnya. Selain itu, mereka juga dapat mengunduh atau menghapus formulir sesuai kebutuhan.

Flowchart ini memastikan semua alur proses aplikasi berjalan dengan terstruktur, memudahkan pengguna dalam mengelola data barang secara efektif.



3.2.2.2 Desain *Mockup*

Desain mockup dirancang untuk memberikan gambaran visual mengenai alur kerja dan tampilan aplikasi, sekaligus membantu praktikan memahami sistem secara keseluruhan sebelum tahap implementasi kode dimulai.



Gambar 3.4 Mockup *dashboard*

Gambar di atas menunjukkan desain halaman dashboard, yang berfungsi sebagai halaman awal ketika aplikasi dibuka. Halaman ini menyediakan tombol navigasi ke halaman *Add Item*, *Storage*, dan *Form Pengembalian*.

Eventory

General

- Dashboard
- Storage
- Scan Item
- Form Pengembalian

Add Item

Data Pengaju

Nama PIC: Divisi: Nomor Telepon/WhatsApp:

Masukkan Nama PIC Masukkan Divisi Masukkan Nomor Telepon

Data Barang

Kategori Barang: Nama Barang: Status Barang: Tanggal Pengembalian Barang:

Masukkan Kategori Barang Masukkan Nama Barang Masukkan Status Barang MM/DD/YYYY

Jumlah Barang: Nama Vendor: Catatan Barang:

Masukkan Jumlah Barang Masukkan Nama Vendor Masukkan Catatan Barang

Foto Bukti Barang

Gambar 3.5 Mockup Add Item

Gambar di atas menunjukkan halaman *Add Item* didesain untuk memungkinkan pengguna mengisi data barang, termasuk informasi pengaju dan detail barang. Data yang diinput nantinya akan disimpan di dalam *Storage*.

Eventory

General

- Dashboard
- Storage
- Scan Item
- Form Pengembalian

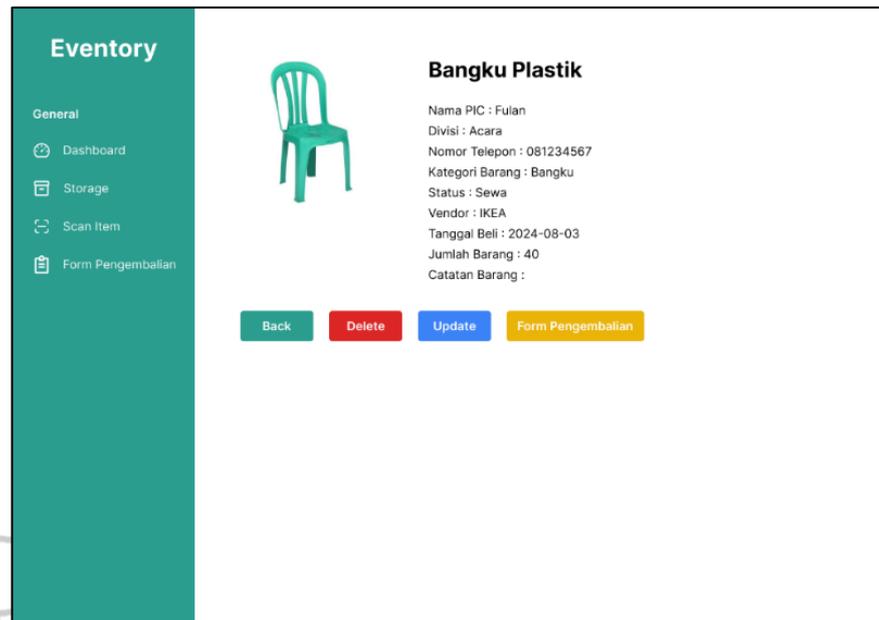
Storage

Nama PIC	Divisi	Kategori	Barang	Status	Tanggal Beli	Detail
Fulan	Acara	Bangku	Bangku Plastik	Sewa	2024-08-03	<input type="button" value="🔍"/>

Gambar 3.6 Mockup Storage Utama

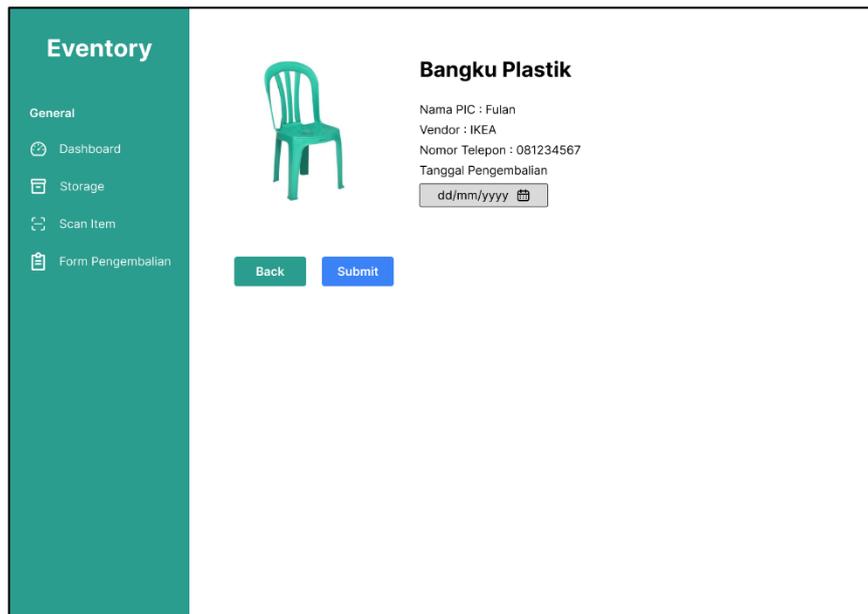
Gambar di atas menunjukkan halaman *Storage* utama menampilkan tabel berisi informasi barang, seperti nama pengaju

atau PIC, divisi, kategori barang, nama barang, status barang, tanggal beli atau sewa, serta tombol *Detail* yang mengarahkan ke halaman informasi lebih lengkap tentang barang tersebut. Halaman ini menampilkan data yang berhasil disimpan setelah penambahan barang.



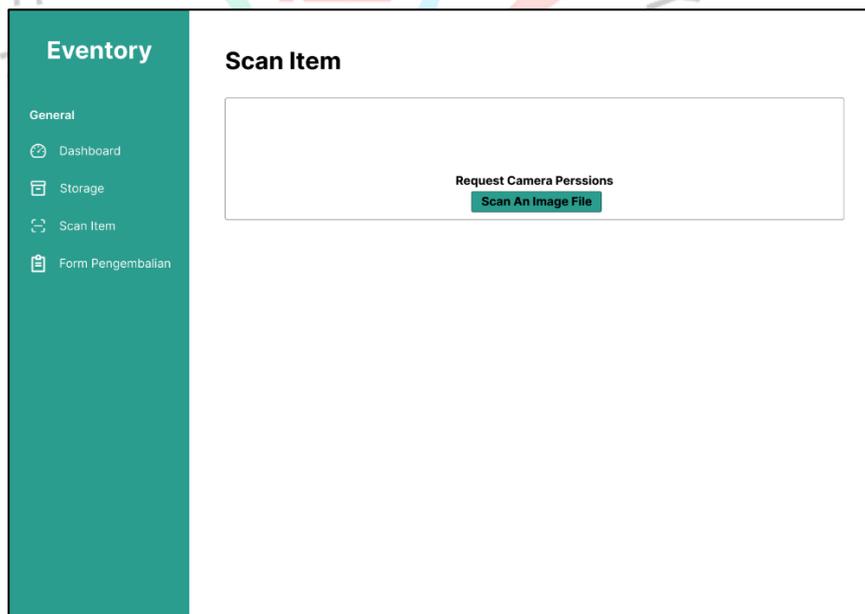
Gambar 3.7 Mockup Detail Storage

Gambar di atas menunjukkan halaman detail *Storage* menampilkan informasi lengkap terkait barang yang telah disimpan. Fitur yang disediakan meliputi penghapusan dan perubahan data barang, serta tombol untuk menuju halaman pengisian *Form Pengembalian*..



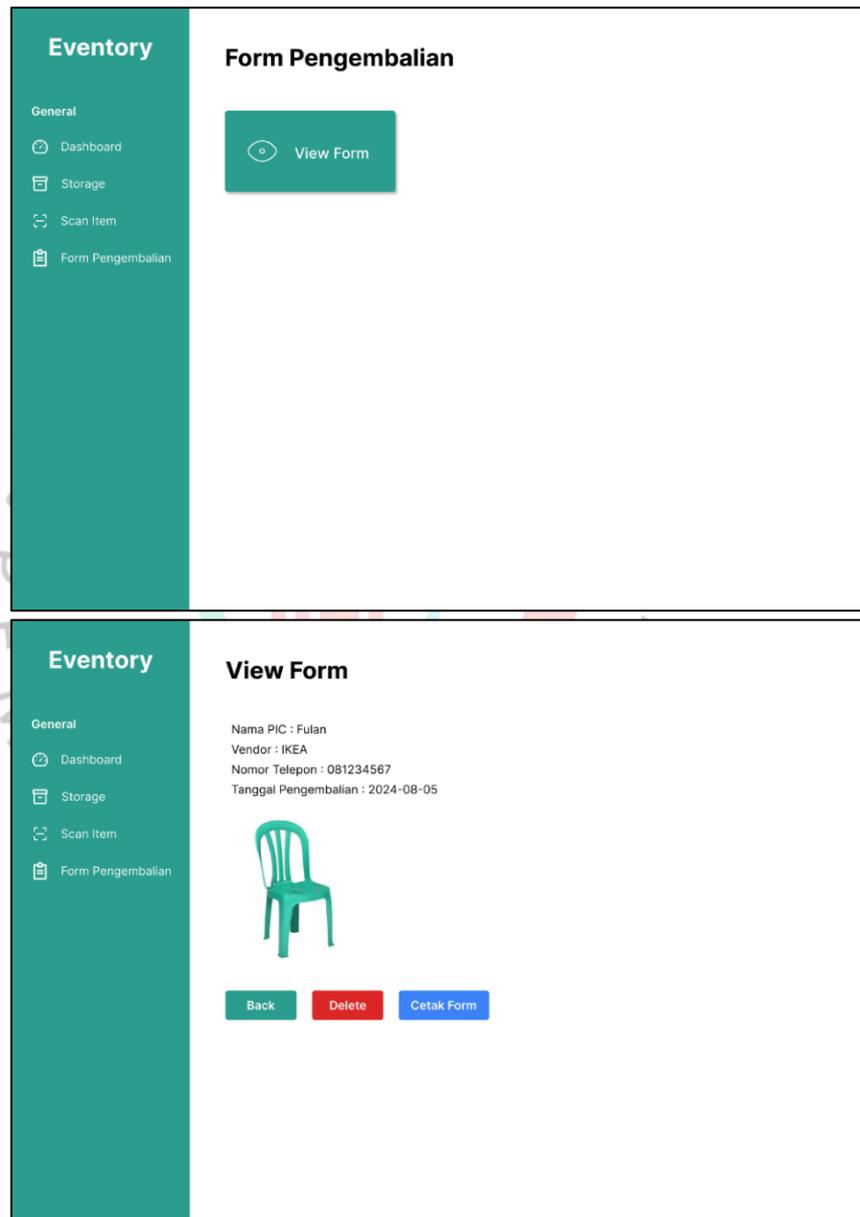
Gambar 3.8 Mockup Input Form Pengembalian

Gambar di atas menunjukkan halaman ini memungkinkan pengguna untuk mengisi data terkait pengembalian barang, seperti tanggal pengembalian sesuai perjanjian peminjaman. Data yang diisi akan disimpan di tabel *return* dalam database..



Gambar 3.9 Mockup Scan Item

Gambar di atas menunjukkan halaman ini dirancang untuk memindai QR Code pada barang. Jika proses pemindaian berhasil, pengguna akan diarahkan ke halaman detail barang sesuai QR Code yang dipindai.



Gambar 3.10 Mockup Form Pengembalian

Gambar di atas menunjukkan halaman *Form Pengembalian* menampilkan informasi barang sewa dan menyediakan fitur untuk menghapus data atau mencetaknya ke dalam format PDF.

Halaman ini membantu pengguna mengelola formulir pengembalian dengan lebih efisien.

3.2.3 Implementasi

Dalam tahap implementasi ini, akan dijelaskan mengenai kerangka kerja yang digunakan, spesifikasi kebutuhan perangkat keras, spesifikasi kebutuhan perangkat lunak, serta antarmuka dari hasil realisasi aplikasi.

Spesifikasi Kebutuhan Perangkat Lunak

Dalam spesifikasi kebutuhan perangkat lunak pengembangan aplikasi berbasis *website* praktikan menggunakan beberapa perangkat lunak yang mendukung proses Pembangunan aplikasi ini berikut.

Tabel 3.3 Spesifikasi Kebutuhan Perangkat Lunak

No	Perangkat Lunak	Kebutuhan Perangkat Lunak
1	<i>Operation System</i>	Windows 11
2	<i>IDE</i>	Visual Studio Code
3	<i>Front-End Framework</i>	Next.js
4	<i>Back-End Framework</i>	Next.js
5	<i>Database</i>	Supabase
6	<i>Scripting Language</i>	Javascript
7	<i>CSS Framework</i>	Tailwind

3.2.3.1 Spesifikasi Kebutuhan Perangkat Keras

Dalam spesifikasi kebutuhan perangkat keras yang praktikan gunakan dalam pembangunan aplikasi ini.

Tabel 3.4 Spesifikasi Kebutuhan Perangkat Keras

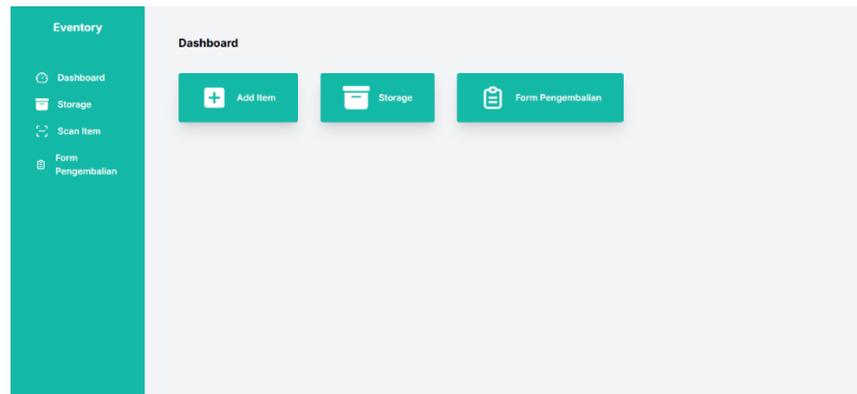
No	Perangkat Keras	Kebutuhan Perangkat Keras
1	Processor	Intel Core i5 7200U
2	VGA	Intel UHD Graphics 620
3	Memory	8 GB

3.2.3.2 Tampilan dan Realisasi Aplikasi

Aplikasi berbasis web ini dikembangkan menggunakan *framework* yang biasa digunakan oleh perusahaan, *framework* yang digunakan praktikan untuk membangun aplikasi ini adalah *Next.js*. *Next.js* adalah *framework React* yang dirancang untuk mempermudah pembangunan aplikasi web dengan berbagai fitur-fitur yang memungkinkan pengembangan aplikasi web dilakukan dengan lebih efisien dan memberikan performa yang optimal.

Framework ini dipilih karena keunggulannya dalam menyediakan fitur modern untuk pengembangan aplikasi secara simultan di sisi front-end dan back-end. Selain itu, untuk mendukung desain antarmuka, digunakan framework CSS Tailwind yang membantu dalam pengelolaan gaya pada halaman web. Tailwind juga sudah terintegrasi dengan baik dalam ekosistem Next.js, sehingga memudahkan proses pengembangan di sisi front-end.

Dalam pengelolaan database, digunakan platform Supabase yang sering dimanfaatkan oleh perusahaan untuk kebutuhan serupa. Pemilihan Supabase didasarkan pada kompatibilitasnya dengan Next.js, kemudahan integrasi, serta fitur bucket storage yang memungkinkan pengelolaan file seperti gambar dan dokumen. Fitur-fitur ini menjadikan Supabase sebagai pilihan yang tepat untuk mendukung pengelolaan database dalam pembangunan aplikasi ini.



Gambar 3.13 Realisasi Halaman *Dashboard*

Gambar di atas merupakan hasil realisasi dari halaman *dashboard* yang telah disesuaikan dengan rancangan sistem dan desain sebelumnya. Pada gambar halaman *dashboard* tersebut, terdapat *button* yang masing-masing memiliki fungsi salah satunya *button add item* yang akan mengarahkan *user* langsung kepada halaman *add item*. Hal ini dilakukan agar *user* dapat dengan langsung menambahkan data barang yang ingin disimpan secara cepat.

Gambar 3.14 Realisasi Halaman *Add Item*

Gambar di atas merupakan hasil realisasi dari halaman *Add Item*. Pada halaman tersebut, terdapat beberapa data yang harus diisi dengan benar oleh *user* agar data tersebut berhasil disimpan ke *database* dan dapat dilihat pada halaman *storage*.

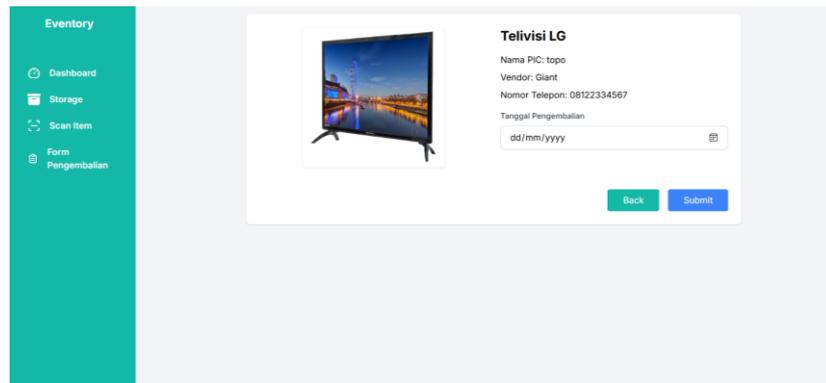
Nama PIC	Divisi	Kategori	Barang	Status	Tanggal Beli	Detail
Zahid	acara	Alat tulis	Pulpen	beli	2024-10-23	
untung	perkap	lampu	lampu tembak	beli	2024-11-16	
topo	acara	televisi	Televisi LG	sewa	2024-11-16	
panjul	humas	Alat tulis	Pulpen Kenko	beli	2024-11-17	

Gambar 3.15 Realisasi Halaman Storage

Gambar di atas merupakan realisasi dari halaman utama *Storage*. Sesuai dengan rancangan desain sebelumnya, pada halaman ini terdapat tabel yang berisikan beberapa data barang yang berhasil disimpan ke dalam *database* pada tahap add item sebelumnya. Selain itu, pada kolom detail terdapat *button* untuk mengarahkan *user* ke halaman data barang tersebut yang lebih lengkap.

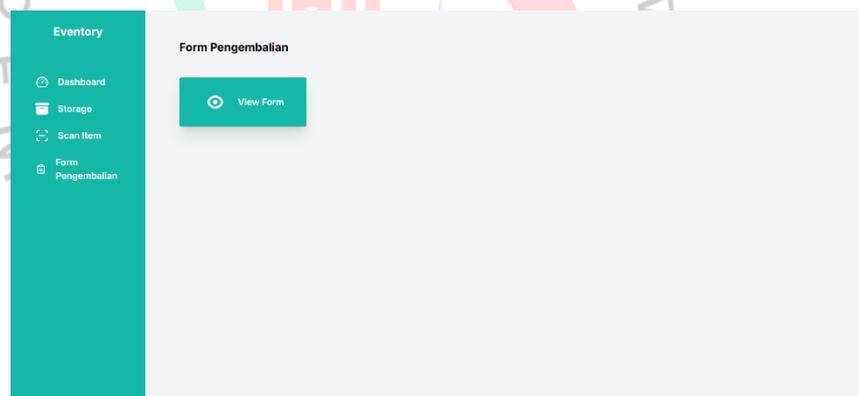
Gambar 3.16 Realisasi Halaman Detail Barang

Gambar di atas adalah realisasi dari halaman detail barang. Masih pada halaman *storage*, halaman detail barang ini menyediakan informasi lengkap terkait barang yang dipilih, selain itu, *user* juga dapat mengubah ataupun menghapus data. Halaman ini bertujuan untuk memperlihatkan kepada *user* data barang yang berhasil disimpan, dan mengelola lebih lanjut apakah data telah sesuai atau tidak.



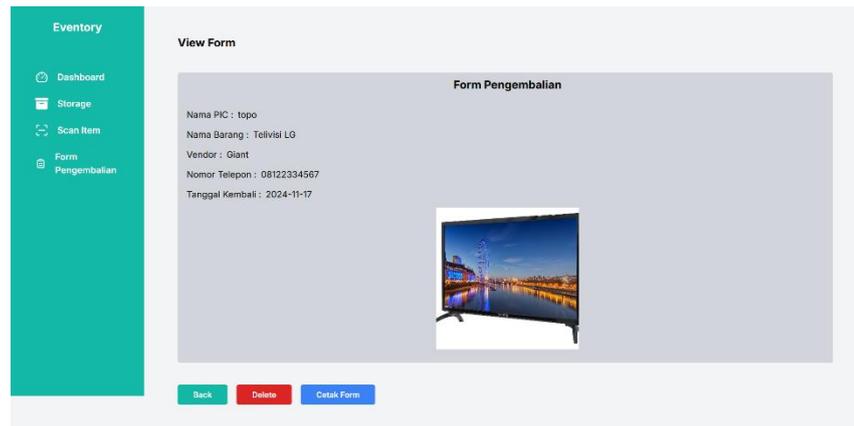
Gambar 3.17 Realisasi Halaman Input Form Pengembalian

Gambar di atas merupakan realisasi dari halaman input form pengembalian pada halaman storage. Halaman ini berfungsi untuk pembuatan formulir pengembalian yang berisi beberapa data barang. Setelah data diisi dengan benar, maka data akan disimpan ke dalam tabel *return* pada *database*. Halaman ini akan muncul ketika *user* menekan *button* form pengembalian pada halaman detail barang.



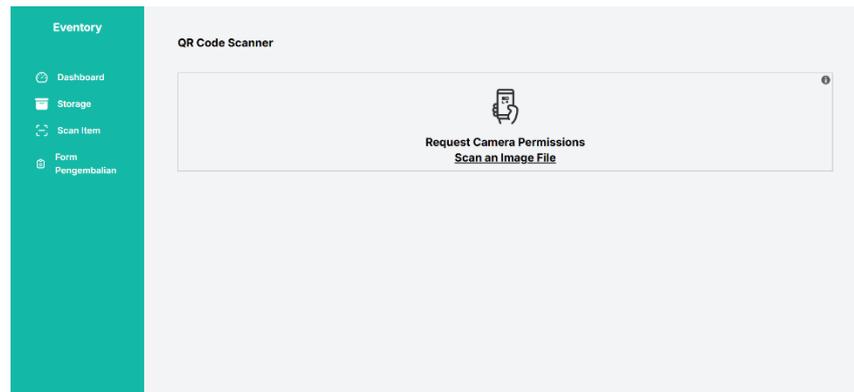
Gambar 3.18 Realisasi Halaman Utama Form Pengembalian

Gambar di atas merupakan realisasi dari halaman utama form pengembalian. Halaman ini berisi *button* untuk mengarahkan *user* ke halaman *preview* formulir pengembalian barang yang telah berhasil disimpan.

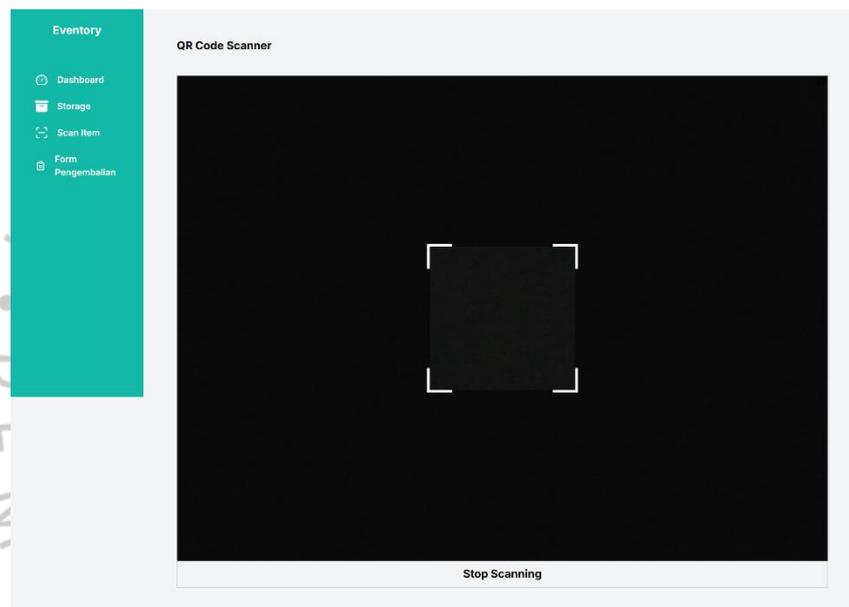


Gambar 3.19 Realisasi Halaman Preview Form Pengembalian

Gambar di atas merupakan realisasi dari halaman *preview* form pengembalian. Halaman ini berfungsi untuk memperlihatkan data barang sewa hingga waktu pengembaliannya yang dibentuk layaknya formulir, data yang terdapat pada formulir tersebut antara lain adalah nama PIC, nama barang, nama vendor, nomor telepon PIC, tanggal pengembalian barang, dan foto barang. Data tersebut adalah data yang berhasil disimpan ke dalam *database* tabel *return*, dan diperlihatkan di halaman *preview* form pengembalian ini. Selain itu terdapat fitur untuk mencetak form ke dalam format *.pdf* dan fitur untuk menghapus *preview* form apabila *user* telah mencetak form pengembalian tersebut.



Gambar 3.20 Realisasi Halaman Scan Item



Gambar 3.21 Realisasi Fitur Kamera Untuk Scan Item

Gambar di atas merupakan realisasi dari halaman *scan item*. Pada halaman ini, terdapat fitur *scan item* yang berfungsi untuk melakukan *scanning* pada *QR Code* barang. Fitur *scanning* bisa dilakukan secara langsung lewat kamera device atau lewat file foto *QR Code*.

3.3 Kendala yang Dihadapi

Praktikan telah menjalani peran sebagai karyawan magang di divisi developer PT Brainlens Rekayasa Konsultan dengan cukup baik, namun terdapat beberapa kendala yang dihadapi selama proses pembangunan aplikasi berbasis website. Berikut adalah kendala yang dialami:

- 1) Implementasi bahasa pemrograman yang terbatas, terutama penggunaan *framework Next.js* dan platform pengelola *database Supabase*, menyebabkan beberapa tantangan dan perlu waktu yang lebih.
- 2) Kurangnya rekan kerja magang menyebabkan pembangunan aplikasi tersebut memakan waktu cukup lama dan terbatasnya inovasi baru yang didapatkan dari rekan kerja.
- 3) Sulitnya berkomunikasi secara tatap muka ataupun *virtual* dengan mentor yang menyebabkan inovasi baru atau pengalaman dunia kerja yang didapatkan dari mentor tidak banyak.

3.4 Cara mengatasi Kendala

Dalam menghadapi berbagai kendala selama kegiatan magang, praktikan berupaya menemukan solusi yang dapat membantu meningkatkan kemampuan dan pengalaman kerja. Berikut adalah langkah-langkah yang dilakukan:

- 1) Praktikan aktif mengeksplorasi informasi terkait penggunaan *framework* dan platform melalui internet, serta berkonsultasi secara berkala dengan mentor untuk memastikan pembangunan aplikasi berjalan sesuai rencana.
- 2) Praktikan mengadakan diskusi langsung dengan rekan kerja ketika ada waktu luang, membahas perkembangan pekerjaan, membagi jadwal tugas, dan saling berbagi pengetahuan terkait pengembangan aplikasi.
- 3) Setiap kesempatan berdiskusi dengan mentor dimanfaatkan secara maksimal, dengan mengajukan pertanyaan yang relevan untuk memperoleh wawasan dan pemahaman yang lebih mendalam guna mendukung pelaksanaan pekerjaan.

3.5 Pembelajaran yang Diperoleh dari Kerja Profesi

Selama praktikan bekerja di perusahaan PT Brainlens Rekayasa Konsultan, praktikan mendapatkan beberapa pembelajaran dari pembangunan aplikasi berbasis *website* ini. Berikut adalah pembelajaran yang diperoleh praktikan dari kerja profesi.

- 1) Praktikan mendapatkan pembelajaran tentang bagaimana membangun aplikasi pengelolaan data logistik berbasis *website*.
- 2) Praktikan mendapatkan pembelajaran terkait penggunaan *framework Next.js*, *Tailwind CSS*, dan platform pengelolaan *database Supabase* yang biasa digunakan perusahaan untuk mengembangkan sebuah aplikasi atau *website*.
- 3) Praktikan mendapatkan wawasan baru tentang apa saja tantangan yang akan ditemui di dunia kerja nanti terutama bagi seorang yang bekerja sebagai *developer*.

