



Smart Storage: Rancang Bangun Sistem Peminjaman dan Pengembalian Barang Otomatis Menggunakan Teknologi RFID dan IoT di Lingkungan Kampus Dengan Metode Unified Modeling Language (UML)

Laporan Tugas Akhir

Oleh:

Mahesha Abiyyi Gifari (4242211029)

Muhammad Ramadhan Pulungan (4242211022)

Program Studi Teknologi Rekayasa Elektronika

Jurusan Teknik Elektro

Politeknik Negeri Batam

2026

Lembar Pengesahan

**Tugas Akhir disusun untuk memenuhi salah satu syarat memperoleh gelar
Sarjana Terapan Teknik (S.Tr.T)**

Disusun oleh:

Mahesha Abiyyi Gifari (4242211029)

Muhammad Ramadhan Pulungan (4242211022)

Tanggal Sidang: 19 Mei 2026

Disetujui oleh :

- 1. Nanta Fakhri Prebianto, S.ST., M.Sc**
NIK : 198903022019031016



- 1. Ridwan, S.ST., M. Tr. T**
NIK : 113113



- 2. Maria Evita Sari, S.Pd., M.Hum**
NIK : 125365



Smart Storage: Rancang Bangun Sistem Peminjaman dan Pengembalian Barang Otomatis Menggunakan Teknologi RFID dan IoT di Lingkungan Kampus Dengan Metode Unified Modeling Language (UML)

Abstrak

Sistem Smart Storage merupakan sistem otomatis berbasis RFID (Radio Frequency Identification) dan IoT (Internet of Things) yang dirancang untuk mempermudah proses peminjaman dan pengembalian barang di lingkungan kampus. Sistem ini menggunakan ESP32 sebagai pengendali utama, modul RFID sebagai media identifikasi pengguna, serta mekanisme conveyor untuk proses pengeluaran dan penempatan barang secara otomatis, yang terintegrasi dengan aplikasi berbasis web untuk monitoring data secara real-time. Pengujian dilakukan untuk mengevaluasi kinerja sistem dari beberapa aspek, meliputi proses peminjaman dan pengembalian barang, pembacaan RFID, pemrosesan data oleh ESP32, serta komunikasi data menggunakan protokol HTTP. Hasil pengujian menunjukkan bahwa sistem peminjaman dan pengembalian barang dapat berjalan dengan stabil setelah dilakukan penyesuaian pada mekanisme konveyor, sehingga pergerakan barang menjadi lebih presisi dan konsisten. Modul RFID mampu membaca kartu secara cepat dan stabil pada jarak optimal hingga ± 3 cm dengan waktu respon yang singkat. Modul ESP32 mampu memproses data dan mengendalikan output sistem secara responsif, serta menampilkan data pada LCD I2C dengan baik. Selain itu, komunikasi data antara ESP32 dan server melalui protokol HTTP menunjukkan waktu respon dibawah 2 detik dengan tingkat akurasi data yang tinggi dan koneksi yang stabil selama pengujian berulang. Berdasarkan hasil tersebut, sistem Smart Storage yang dikembangkan terbukti mampu bekerja secara efisien, stabil, dan terintegrasi, sehingga layak diterapkan sebagai solusi pengelolaan peminjaman dan pengembalian barang otomatis di lingkungan kampus.

Kata kunci: sistem otomatis, RFID, IoT, UML.

Smart Storage: Design and Implementation of an Automated Item Borrowing and Returning System Using RFID and IoT Technology in Campus Environment with Unified Modeling Language (UML) Method

Abstract

The Smart Storage System is an automated system based on RFID (Radio Frequency Identification) and IoT (Internet of Things) designed to streamline the process of borrowing and returning items on campus. The system utilises an ESP32 as the main controller, an RFID module for user identification, and a conveyor mechanism for the automatic retrieval and placement of items, all integrated with a web-based application for real-time data monitoring. Testing was conducted to evaluate the system's performance across several aspects, including the borrowing and returning of items, RFID reading, data processing by the ESP32, and data communication using the HTTP protocol. The test results showed that the item borrowing and returning system operates stably following adjustments to the conveyor mechanism, resulting in more precise and consistent item movement. The RFID module is capable of reading cards quickly and reliably at an optimal distance of up to ± 3 cm with a short response time. The ESP32 module is capable of processing data and controlling system output responsively, as well as displaying data on the I2C LCD effectively. Furthermore, data communication between the ESP32 and the server via the HTTP protocol demonstrated response times of under 2 seconds, with a high level of data accuracy and a stable connection throughout repeated testing. tured, thereby reducing recording errors and facilitating inventory monitoring. Based on these results, the Smart Storage system developed has proven capable of operating efficiently, reliably and seamlessly, making it suitable for implementation as a solution for automated item borrowing and return management on campus.

Keywords: automated system, RFID, IoT, UML.

Daftar Isi

Lembar Pengesahan	1
Abstrak	2
<i>Abstract</i>	3
Daftar Isi	4
Daftar Gambar	6
Daftar Tabel	7
Bab 1. Pendahuluan	8
1.1. Latar Belakang	8
1.2. Rumusan Masalah	9
1.3. Tujuan	9
1.4. Manfaat	10
1.5. Batasan	10
1.6. <i>Work Breakdown Structure</i>	10
Bab 2. Tinjauan Pustaka	11
2.1 Studi Literatur	11
2.2 Gambaran Perkembangan Produk	12
2.2.1 Perangkat Lunak	12
2.2.2 Perangkat Keras	19
Bab 3. Metodologi Penelitian	26
3.1 Perancangan	26
3.1.1 Perancangan Tahapan Penelitian	26
3.1.2 Perancangan Blok Diagram Sistem	27
3.1.3 Perancangan Sistem Dengan Desain UML	28
3.1.4 Komunikasi Data Menggunakan Protokol HTTP	34
3.2 Perancangan Mekanikal	36
3.3 Perancangan Elektrikal	38
3.4 Pengujian	39
3.4.1 Pengujian Sistem Peminjaman Barang	40

3.4.2	Pengujian Sistem Pengembalian Barang.....	40
3.4.3	Pengujian pada Modul RFID.....	41
3.4.4	Pengujian Modul ESP32 dalam Pemrosesan Data	42
3.4.5	Pengujian Sistem Berdasarkan Diagram UML.....	43
3.4.6	Pengujian Sistem IoT Menggunakan HTTP	44
Bab 4.	Hasil dan Pembahasan	46
4.1	Hasil Rancangan Alat.....	46
4.2	Hasil Rancangan Perangkat Lunak	47
4.2.1	Konfigurasi Sistem	48
4.2.2	Konfigurasi Setup Mode	49
4.2.3	Prosedur Penggunaan Sistem	50
4.2.4	Proses Register User Baru.....	53
4.2.5	Proses Inventory Barang.....	53
4.2.6	Proses Peminjaman Barang oleh User.....	55
4.2.7	Proses Pengembalian Barang oleh User	57
4.3	Hasil Pengujian Sistem	59
4.3.1	Pengujian Sistem Peminjaman Barang	60
4.3.2	Pengujian Sistem Pengembalian Barang.....	63
4.3.3	Pengujian Modul RFID	66
4.3.4	Pengujian Modul ESP32 dalam Pemrosesan Data	67
4.3.5	Pengujian Sistem Berdasarkan UML	69
4.3.6	Pengujian Sistem IoT Menggunakan HTTP	71
4.3.7	Pengujian <i>Functional</i> IoT	74
4.3.8	Pengujian <i>Performance</i> IoT.....	75
Bab 5.	Penutup	77
5.1	Kesimpulan	77
5.2	Saran	78
	Daftar Pustaka	79
	Lampiran	81

Daftar Gambar

Gambar 2.1 ESP32	20
Gambar 2.2 LCD 16x2 I2C	21
Gambar 2.3 Konveyor	21
Gambar 2.4 RFID	22
Gambar 2.5 Servo	23
Gambar 2.6 Sensor TCS3200	23
Gambar 2.7 Sensor Infrared	24
Gambar 2.8 Motor Stepper	25
Gambar 3.1 Flowchart Penelitian	27
Gambar 3.2 Blok Diagram Sistem	27
Gambar 3.3 Use Case Diagram	29
Gambar 3.4 <i>Activity Diagram</i>	31
Gambar 3.5 Activity Diagram Gerak Conveyor	33
Gambar 3.6 Desain 3D Mekanikal	36
Gambar 3.7 Desain Umum Mekanikal	37
Gambar 3.8 Desain Elektrikal	38
Gambar 4.1 Tampilan Full Unit	46
Gambar 4.2 Tampilan <i>Box Panel Kontrol</i>	47
Gambar 4.3 Pemilihan Board ESP32	48
Gambar 4.4 Tampilan XAMPP Control Panel pada Sistem Smart Storage	49
Gambar 4.5 Proses Aktivasi XAMPP dan Pembuatan Database	49
Gambar 4.6 Proses Booting dan Aktivasi Setup Mode	50
Gambar 4.7 Koneksi Access Point Smart Storage	51
Gambar 4.8 Tampilan Login dan Konfigurasi Sistem	51
Gambar 4.9 Tampilan Pengaturan Parameter Device	52
Gambar 4.10 Tampilan Manajemen API Key Perangkat	52
Gambar 4.11 Menu Pengguna dan Mode Registrasi RFID	53
Gambar 4.12 Tampilan Menu <i>Inventory</i> pada Sistem <i>Smart Storage</i>	54
Gambar 4.13 Form Penambahan Data Barang pada Sistem	54
Gambar 4.14 Proses Penambahan Barang ke <i>Inventory System</i>	55
Gambar 4.15 Tampilan Menu Peminjaman Barang	55
Gambar 4.16 Proses Peminjaman Barang dan Verifikasi PIN	56
Gambar 4.17 Proses Pengeluaran Barang pada Conveyor	57
Gambar 4.18 Tampilan Menu Pengembalian Barang	58
Gambar 4.19 Proses Pengembalian Barang dan Verifikasi PIN	58
Gambar 4.20 Proses Pengembalian Barang pada Conveyor	59

Daftar Tabel

Tabel 1.1 <i>Work Breakdown Structure</i>	10
Tabel 3.1 Penggunaan Komponen HTTP	35
Tabel 3.2 Pengujian Sistem Peminjaman Barang	40
Tabel 3.3 Pengujian Sistem Pengembalian Barang.....	41
Tabel 3.4 Pengujian pada Modul RFID.....	41
Tabel 3.5 Pengujian Modul ESP32 dalam Pemrosesan Data	42
Tabel 3.6 Pengujian Sistem Berdasarkan Diagram UML.....	43
Tabel 3.7 Pengujian Sistem IoT Menggunakan HTTP.....	44
Tabel 4.1 Pengujian Conveyor Utama (Masalah Motor & Belting).....	60
Tabel 4.2 Pengujian Conveyor Slot (Masalah Presisi & Posisi Barang)	61
Tabel 4.3 Pengujian Presisi & Adjustment Barang.....	61
Tabel 4.4 Pengujian Sistem Peminjaman Barang (Hasil Akhir)	62
Tabel 4.5 Pengujian Sensor Warna Pada Proses Pengembalian Barang.....	63
Tabel 4.6 Pengujian Konveyor Utama (Pengembalian Barang)	64
Tabel 4.7 Pengujian Konveyor Slot (Masalah Penempatan Barang).....	64
Tabel 4.8 Pengujian Servo Mekanisme Pusher.....	65
Tabel 4.9 Pengujian Sistem Pengembalian Barang.....	66
Tabel 4.10 Pengujian Modul RFID	67
Tabel 4.11 Pengujian Awal Modul ESP32 dalam Pemrosesan Data	68
Tabel 4.12 Pengujian Akhir Modul ESP32 dalam Pemrosesan Data	69
Tabel 4.13 Pengujian Sistem Berdasarkan UML	70
Tabel 4.14 Pengujian Kecepatan Respon Sistem IoT Menggunakan HTTP	71
Tabel 4.15 Pengujian Akurasi Data Sistem IoT Menggunakan HTTP	72
Tabel 4.16 Pengujian Stabilitas Koneksi Sistem IoT Menggunakan HTTP.....	73
Tabel 4.17 Pengujian Sistem IoT Menggunakan HTTP	73
Tabel 4.18 Pengujian Functional IoT	74
Tabel 4.19 Pengujian Performance IoT	75

Bab 1. Pendahuluan

1.1. Latar Belakang

Di tengah pesatnya arus modernisasi dan globalisasi, kemajuan teknologi telah mengubah cara manusia hidup. Inovasi digital yang tak henti-hentinya mengubah cara orang berkomunikasi dan sistem kerja di berbagai bidang kehidupan. Dengan kemajuan pesat di bidang komputasi, jaringan, dan otomatisasi, tugas yang sebelumnya memakan waktu berjam-jam kini dapat diselesaikan dalam hitungan detik. Namun, meskipun arus teknologi ini memiliki manfaat yang luar biasa, derasnyanya arus ini juga membawa konsekuensi yang tidak terhindarkan. Sementara limbah elektronik yang terus meningkat menimbulkan ancaman serius bagi kelestarian lingkungan, ketergantungan berlebihan pada perangkat digital berpotensi mengurangi kemampuan dasar manusia. Fenomena ini menunjukkan bahwa di tengah kegembiraan kemajuan, manusia modern menghadapi paradoks baru. Solusi untuk masalah baru yang kompleks muncul bersamaan dengan kemajuan teknologi yang memudahkan hidup.

Proses peminjaman dan pengembalian barang, khususnya di lingkungan kampus seperti laboratorium, sering kali menjadi hambatan dalam efisiensi operasional. Saat ini, banyak institusi masih mengandalkan sistem manual yang rentan terhadap berbagai permasalahan. Pada praktiknya, keterlambatan pencatatan data dan kesalahan dalam pencarian informasi barang menjadi keluhan umum. Hal ini tidak hanya memperlambat proses layanan, tetapi juga membebani staf laboratorium dengan tumpukan pekerjaan administratif yang repetitif. Keterbatasan sistem manual ini juga mempersulit proses monitoring status barang secara real-time, menyebabkan barang yang keluar atau masuk tidak terpantau dengan baik, sehingga berpotensi menimbulkan ketidakjelasan inventaris. [1]

Selain itu, pengelolaan inventaris dalam skala yang lebih luas, seperti di perpustakaan atau gudang penyimpanan barang umum di kampus, juga menghadapi tantangan serupa namun dengan kompleksitas yang berbeda. Sistem manual di area ini seringkali menyebabkan proses layanan yang lambat dan rentan terhadap kesalahan pencatatan data peminjaman maupun pengembalian. Ketika terjadi lonjakan aktivitas, petugas sering kewalahan dalam mengelola data anggota, daftar barang, serta riwayat transaksi. Akibatnya, validitas data menjadi meragukan dan risiko kehilangan data akibat human error meningkat. Kondisi ini

pada akhirnya berdampak pada kesulitan dalam penyusunan laporan yang akurat dan tepat waktu, sehingga menghambat pengambilan keputusan terkait pengelolaan aset kampus. [2]

Berdasarkan permasalahan tersebut, dirancang sebuah sistem Smart Storage yang mengintegrasikan teknologi RFID (Radio Frequency Identification), IoT (Internet of Things), dan pemodelan UML (Unified Modeling Language) sebagai solusi untuk meningkatkan proses peminjaman dan pengembalian barang di lingkungan kampus. Critical consideration dalam penelitian ini adalah tingginya risiko kesalahan pencatatan data peminjaman dan pengembalian barang yang masih dilakukan secara manual, sehingga menyebabkan ketidaksesuaian data inventaris, keterlambatan proses pencarian barang, serta berpotensi menimbulkan kehilangan aset kampus. Oleh karena itu, diperlukan sistem yang mampu meningkatkan akurasi, efisiensi, keamanan, serta pemantauan data secara real-time melalui penerapan teknologi RFID dan IoT. Selain itu, sistem ini juga berpotensi diterapkan pada institusi lain yang membutuhkan sistem manajemen inventaris sehingga relevan dalam mendukung digitalisasi pengelolaan aset di era teknologi saat ini.

1.2. Rumusan Masalah

Berdasarkan penjelasan sebelumnya, masalah yang dibahas dalam tugas akhir ini adalah :

1. Bagaimana cara mengimplementasikan sistem peminjaman dan pengembalian barang otomatis untuk mengatasi masalah pencatatan manual dan meningkatkan efektivitas manajemen aset kampus?
2. Bagaimana memantau status barang secara real-time dengan menerapkan teknologi IoT dan RFID secara terintegrasi dalam sistem?
3. Bagaimana cara menggunakan pendekatan UML untuk merancang sistem dengan pemodelan yang terstruktur?

1.3. Tujuan

Tujuan tugas akhir ini adalah:

1. Membuat sistem mekanik untuk peminjaman dan pengembalian barang kampus yang mudah digunakan dan efektif.
2. Menggunakan sistem IoT untuk memantau dan mengelola barang secara real-time, meningkatkan efisiensi dan ketepatan proses peminjaman dan pengembalian.

3. Merancang sistem secara terstruktur menggunakan pendekatan UML dengan Menyusun diagram seperti *use case diagram*, *activity diagram*, *sequence diagram* dan class diagram, agar pengembangan sistem menjadi lebih terarah dan mudah dipahami.

1.4. Manfaat

Manfaat yang diharapkan dari tugas akhir ini antara lain:

1. Menggunakan sistem otomatis untuk memudahkan peminjaman dan pengembalian barang di kampus.
2. Dengan teknologi IoT, mengelola barang secara real-time menjadi lebih mudah dan mengurangi kesalahan pencatatan.
3. Meningkatkan akurasi dalam memantau barang yang dipinjam

1.5. Batasan

Batasan masalah pada pembuatan alat ini ditujukan untuk mempermudah mahasiswa untuk peminjaman dan pengembalian barang di laboratorium kampus dan mempermudah Laboran Kampus untuk monitoring komponen yang dipinjam dan dikembalikan. Adapun Batasan masalah adalah sebagai berikut :

1. Alat ini hanya difokuskan untuk peminjaman dan pengembalian barang di lab kampus.
2. Rak conveyor dapat menyimpan 5 komponen.
3. Hanya dengan card RFID dapat membuka akses alat.
4. Alat hanya menampung 3 jenis komponen saja.

1.6. Work Breakdown Structure

Tabel 1.1 Work Breakdown Structure

No	Nama	Tugas dan Tanggung Jawab dalam Tim
1	Mahesha Abiyyi Gifari	Programming software website, serta mengerjakan bagian mekanikal secara bersama.
2	Muhammad Ramadhan Pulungan	Programming device ESP32, serta mengerjakan bagian mekanikal secara bersama.

Bab 2. Tinjauan Pustaka

2.1 Studi Literatur

Tinjauan Pustaka merupakan proses mengkaji atau mereview beberapa penelitian sebelumnya yang terkait dengan penelitian ini. Dalam hal ini bertujuan agar penelitian ini memiliki perbandingan terhadap penelitian sebelumnya.

Berdasarkan pembuatan proposal terdapat beberapa jurnal dan makalah referensi yang menjadi inspirasi. Penelitian tentang pembuatan Smart Storage atau tempat peminjaman dan pengembalian barang otomatis. Penelitian membahas sistem peminjaman barang digital berbasis web untuk menggantikan metode manual di lingkungan kampus. Hasilnya menunjukkan peningkatan efisiensi hingga 60% serta solusi terhadap masalah pencatatan dan pelacakan barang. Sistem dirancang menggunakan pendekatan UML dan memperhatikan aspek keamanan, antarmuka pengguna, serta kendala teknis dan non-teknis. Studi ini menjadi landasan penting bagi pengembangan *Smart Storage* berbasis RFID dan IoT. [3]

Penelitian merancang sistem peminjaman dan pengembalian buku berbasis RFID untuk mengatasi permasalahan efisiensi layanan di perpustakaan. Sistem ini memanfaatkan RFID tag yang ditempelkan pada kartu anggota dan buku, sehingga memungkinkan transaksi berlangsung otomatis tanpa input manual dari petugas. Dengan bantuan mikrokontroler dan ethernet shield, data RFID dikirim langsung ke server melalui jaringan LAN, lalu diproses dan ditampilkan melalui sistem web. Hasilnya, proses peminjaman menjadi lebih cepat, akurat, dan mengurangi antrean, sekaligus mengurangi ketergantungan terhadap barcode konvensional yang mudah rusak dan kurang efisien. [4]

Penelitian merancang sistem peminjaman tablet berbasis RFID di lingkungan Pondok Pesantren Ngalah untuk mengatasi permasalahan pencatatan manual yang menyulitkan pengurus asrama. Sistem ini memungkinkan santri meminjam tablet dengan menggunakan Kartu Tanda Santri (KTS) yang telah terintegrasi RFID, dan secara otomatis mencatat transaksi serta memantau kuota peminjaman. Dengan teknologi ini, efisiensi pengelolaan dan keamanan peminjaman meningkat, karena hanya santri terdaftar yang dapat mengakses perangkat. Sistem juga dilengkapi dengan fitur pelaporan dan batas kuota maksimal peminjaman untuk menjaga keteraturan dan pengawasan. [5]

Penelitian mengembangkan sistem peminjaman peralatan laboratorium berbasis RFID dan IoT sebagai solusi terhadap sistem konvensional yang masih

mengandalkan formulir kertas yang rentan rusak, sulit didokumentasikan, dan tidak efisien. Sistem ini memanfaatkan RFID Tag yang dipasang pada setiap peralatan laboratorium dan RFID reader yang terhubung dengan mikrokontroler NodeMCU ESP8266 untuk membaca data peralatan. Informasi hasil pembacaan, termasuk identitas alat dan data peminjam yang diperoleh melalui Kartu Tanda Mahasiswa (KTM), langsung dikirim dan tersimpan secara otomatis dalam Google Sheets. Pendekatan ini tidak hanya meningkatkan efisiensi pencatatan, tetapi juga memungkinkan akses data secara daring, sehingga mendukung sistem dokumentasi digital yang hemat kertas dan mudah dikelola. Hasil pengujian menunjukkan bahwa sistem dapat mendeteksi RFID Tag dengan jarak minimum 3 cm dan seluruh proses pencatatan peminjaman berlangsung secara real-time dan terstruktur. [6]

Berdasarkan referensi penelitian tersebut maka perlu dilakukan penyempurnaan dan penyesuaian fungsi serta penambahan fitur agar project Smart Storage dapat berguna dan lebih efisien dalam penggunaannya.

2.2 Gambaran Perkembangan Produk

Gambaran perkembangan produk menjelaskan proses evolusi dan peningkatan sistem yang dikembangkan dalam penelitian ini. Pada tahap ini diuraikan bagaimana sistem Smart Storage dirancang dan dikembangkan dari konsep awal hingga menjadi sebuah sistem yang terintegrasi antara perangkat keras dan perangkat lunak. Perkembangan produk mencakup perancangan sistem, pemilihan teknologi yang digunakan, serta implementasi fitur-fitur yang mendukung proses peminjaman dan pengembalian barang secara otomatis. Dengan adanya tahapan ini, diharapkan sistem yang dihasilkan dapat memenuhi kebutuhan pengguna serta memberikan solusi yang efektif dalam pengelolaan inventaris barang.

2.2.1 Perangkat Lunak

2.2.1.1 Perancangan Sistem dengan UML

Unified Modeling Language (UML) adalah bahasa pemodelan visual standar yang digunakan untuk memvisualisasikan, menspesifikasikan, membangun, dan mendokumentasikan bagian-bagian penting dalam pengembangan perangkat lunak. UML memungkinkan pengembang untuk menggambarkan struktur, perilaku, serta interaksi sistem dengan cara yang terorganisir dan mudah dipahami oleh semua pihak yang terlibat, baik dari sisi teknis maupun non-teknis. UML tidak hanya membantu dalam tahap desain, tetapi juga bermanfaat dalam mempermudah analisis, komunikasi tim, serta dokumentasi sistem yang baik dan

terstruktur. Dengan UML, pengembangan sistem menjadi lebih terarah karena setiap bagian sistem telah tergambar dengan jelas, sehingga meminimalkan risiko miskomunikasi dan kesalahan implementasi.

Use Case Diagram pada sistem ini memvisualisasikan interaksi antara aktor utama (Admin, User, dan Perangkat IoT) dengan fitur inti. diagram ini mengadopsi pendekatan tiga lapis (input-proses-output) dimana Admin memiliki kendali penuh terhadap manajemen data melalui *dashboard web*, sementara User hanya dapat melakukan peminjaman dan pengembalian barang melalui *RFID Reader*. Implementasi use case "Pinjam Barang" merujuk pada teori identifikasi otomatis Finkenzeller yang menyatakan bahwa pembacaan RFID harus disertai verifikasi dua tahap (kartu pengguna dan tag barang) untuk meminimalisasi kesalahan.

Class Diagram dirancang dengan menerapkan prinsip SOLID khususnya *Single Responsibility Principle*, dimana setiap kelas memiliki tanggung jawab tunggal seperti terlihat pada pemisahan antara kelas Pengguna (bertugas menyimpan data anggota), Barang (mengelola status inventaris), dan Transaksi (menangani logika peminjaman). Relasi *one-to-many* antara Pengguna dan Transaksi sesuai dengan teori model relasional Date tentang integritas referensial, diimplementasikan melalui *foreign key* di tingkat database.

Sequence Diagram menggambarkan alur komunikasi real-time antara komponen fisik dan digital. Tahapan pengiriman data dari ESP32 ke server menggunakan protokol HTTP/1.1 dengan status code 200 OK menunjukkan kesesuaian dengan standar arsitektur RESTful API. Mekanisme *error handling* yang mencakup respon *403 Forbidden* dan *503 Service Unavailable* mengacu pada *best practice* IoT tentang manajemen kegagalan pada sistem embedded.

Activity Diagram memperjelas alur bisnis utama dengan mengkombinasikan teori *finite state machine* dan prinsip *Human-Computer Interaction*. Simbol decision point ({Valid?}) merepresentasikan critical control point dimana sistem harus memberikan feedback visual (LED hijau/merah) dalam waktu <500ms untuk memenuhi prinsip usability.

Dalam sistem peminjaman dan pengembalian barang otomatis berbasis IoT dan RFID yang dirancang, UML berperan penting untuk memodelkan proses secara menyeluruh, mulai dari identifikasi pengguna melalui RFID, pengolahan data oleh ESP32, hingga pengelolaan transaksi pada web server. Penggunaan UML dalam sistem ini membantu menggambarkan seluruh elemen yang terlibat, baik aktor, alur proses, maupun interaksi antar komponen secara logis dan terstruktur. Diagram yang digunakan dalam perancangan sistem meliputi :

1. *Use Case Diagram* : *Use Case Diagram* merupakan diagram yang digunakan untuk menggambarkan interaksi antara aktor (pengguna sistem) dan fungsi-fungsi utama yang tersedia dalam sistem. Diagram ini bertujuan untuk mengidentifikasi kebutuhan fungsional sistem, yaitu apa saja yang harus disediakan oleh sistem agar dapat memenuhi kebutuhan pengguna. Aktor yang terlibat dalam sistem ini misalnya mahasiswa sebagai pengguna dan admin sebagai pengelola data. Contoh use case dalam sistem ini meliputi proses:

- Meminjam barang
- Mengembalikan barang
- Melihat data barang
- Mengelola data pengguna (oleh admin)

Use Case Diagram membantu pengembang memahami batasan sistem dan tanggung jawab setiap aktor, sehingga proses pengembangan dapat fokus pada kebutuhan yang sesuai dengan peran masing-masing pengguna.

2. *Activity Diagram* : *Activity Diagram* merupakan diagram yang digunakan untuk menggambarkan aliran aktivitas atau proses bisnis yang terjadi dalam sistem secara terperinci. Diagram ini membantu memvisualisasikan langkah-langkah yang dilakukan dalam sebuah proses, mulai dari awal hingga akhir. *Activity Diagram* dapat menunjukkan proses yang berlangsung secara berurutan, percabangan keputusan, serta aktivitas yang dapat berjalan secara paralel.

Contoh alur dalam *activity diagram* :

- Sistem membaca ID RFID dan memvalidasi data pengguna.
- Jika pengguna terdaftar → proses dilanjutkan.
- Jika pengguna tidak terdaftar → sistem menampilkan pesan "Akses Ditolak".
- Sistem menampilkan daftar barang yang tersedia pada LCD.
- Mahasiswa memilih barang yang ingin dipinjam.

Activity Diagram penting untuk memperjelas alur operasional sistem, mengidentifikasi keputusan yang mungkin diambil dalam alur proses, serta menggambarkan urutan aktivitas secara paralel atau berurutan dalam sistem.

3. *Sequence Diagram* : *Sequence Diagram* adalah diagram yang menggambarkan urutan interaksi atau komunikasi antar komponen yang terlibat dalam sistem berdasarkan urutan waktu. Diagram ini menunjukkan

bagaimana pesan dikirim antar komponen sistem, dari satu objek ke objek lainnya. Dalam sistem ini, *Sequence Diagram* memvisualisasikan urutan komunikasi mulai dari:

- Pengguna melakukan scan RFID
- *RFID reader* mengirimkan data ke ESP32
- ESP32 memproses data dan berkomunikasi dengan *web server*
- *Web server* mengembalikan respon ke ESP32
- ESP32 mengaktifkan motor konveyor dan memperbarui tampilan pada LCD

Diagram ini membantu memperjelas mekanisme komunikasi *real-time* antar komponen seperti *RFID reader*, ESP32, *server*, *database*, dan antarmuka web, sehingga alur pertukaran data dapat dianalisis dengan baik.

4. *Class Diagram* : *Class Diagram* adalah diagram yang menggambarkan struktur data dan hubungan antar kelas atau entitas dalam sistem. *Class Diagram* membantu menyusun bagaimana data tersimpan dan bagaimana setiap objek dalam sistem saling berhubungan.

Dalam sistem ini, *Class Diagram* mencakup entitas seperti:

- Pengguna: berisi atribut seperti ID pengguna, nama, dan nomor RFID
- Barang: berisi atribut seperti ID barang, nama barang, dan jumlah stok
- Transaksi: berisi data riwayat peminjaman dan pengembalian barang, seperti ID transaksi, tanggal, dan status
- Admin: mengelola data barang dan memantau transaksi sistem

Diagram ini juga menunjukkan hubungan antar entitas, seperti relasi satu-ke-banyak antara pengguna dengan transaksi, atau relasi antara barang dengan transaksi. *Class Diagram* mempermudah proses perancangan basis data dan struktur program yang terorganisir.

Langkah-Langkah Perancangan UML :

1. Analisis kebutuhan sistem :

Tahap awal yang bertujuan untuk mengidentifikasi kebutuhan fungsional dan non-fungsional dari sistem. Contohnya, sistem harus dapat memverifikasi identitas pengguna melalui RFID dan mencatat riwayat peminjaman barang secara real-time.

2. Identifikasi aktor dan *use case* :

Menentukan siapa saja yang berinteraksi dengan sistem, seperti mahasiswa, admin, dan server. Setiap aktor dikaitkan dengan *use case* sesuai perannya, seperti meminjam barang, mengembalikan barang, dan mengelola data.

3. Penyusunan skenario alur kerja :

Membuat skenario yang menggambarkan langkah demi langkah proses yang terjadi, seperti bagaimana mahasiswa meminjam barang: mulai dari scan kartu RFID, validasi data, aktivasi motor konveyor, hingga pencatatan transaksi.

4. Pembuatan diagram UML :

Diagram UML dibuat secara sistematis menggunakan *tools* seperti StarUML atau Draw.io untuk menghasilkan dokumentasi visual yang rapi dan mudah dipahami. Diagram yang dihasilkan harus konsisten dengan kebutuhan sistem dan alur yang telah dianalisis.

5. Validasi dan revisi model UML :

Setelah diagram selesai dibuat, perlu dilakukan validasi untuk memastikan semua kebutuhan sudah tergambarkan dan tidak ada proses yang terlewatkan. Revisi dilakukan jika ditemukan ketidaksesuaian antara model dan kebutuhan sistem.

Manfaat penggunaan UML :

- Membantu semua tim (programmer, analis, dan pengguna) memahami sistem dengan cara yang visual. Dengan diagram UML, sistem dapat dijelaskan melalui gambar yang sederhana dan mudah dipahami. Ini mempermudah komunikasi antar anggota tim, baik dari latar belakang teknis maupun non-teknis, karena semua pihak dapat melihat alur kerja, struktur data, dan interaksi sistem secara langsung tanpa harus memahami detail pemrograman terlebih dahulu.
- Mempermudah identifikasi potensi kesalahan sejak awal pengembangan. Dengan membuat diagram UML sejak tahap perancangan, pengembang dapat menemukan kekurangan atau ketidaksesuaian proses lebih awal sebelum sistem dikembangkan. Hal ini dapat mencegah terjadinya kesalahan desain yang dapat menyebabkan sistem tidak berjalan sesuai kebutuhan atau fungsinya tidak optimal.
- Mempercepat proses dokumentasi dan perancangan sistem yang kompleks. UML menyediakan cara sistematis untuk mendokumentasikan kebutuhan sistem, alur proses, dan struktur data dalam bentuk standar yang diakui secara internasional. Dengan diagram yang jelas, proses perancangan sistem yang kompleks menjadi lebih mudah, terstruktur, dan lebih cepat diselesaikan dibandingkan jika hanya menggunakan deskripsi teks.
- Mempermudah proses *maintenance* di masa depan karena struktur sistem sudah terdokumentasi dengan baik. Setelah sistem berjalan, biasanya akan

ada kebutuhan untuk perbaikan, pengembangan, atau penambahan fitur baru. Dengan adanya dokumentasi UML, pengembang berikutnya dapat memahami bagaimana sistem bekerja tanpa harus memeriksa seluruh kode program secara manual. Diagram UML menjadi referensi utama untuk memahami alur sistem yang sudah dibuat.

Dengan menggunakan UML secara lengkap, sistem yang dikembangkan dapat divisualisasikan dengan jelas mulai dari interaksi pengguna, alur proses, komunikasi antar perangkat keras dan perangkat lunak, hingga struktur data yang digunakan. Hal ini membuat sistem menjadi lebih terstruktur, mudah dipahami, serta mempermudah proses pengembangan dan perawatan di masa depan. [7]

2.2.1.2 Penyimpanan Data SQL

Structured Query Language (SQL) adalah bahasa standar yang digunakan untuk mengakses, mengelola, dan memanipulasi data dalam sistem manajemen basis data relasional (*Relational Database Management System/RDBMS*). SQL menjadi bagian penting dalam pengembangan sistem informasi karena memungkinkan pengelolaan data secara efisien, akurat, dan terstruktur.

Dalam sistem Smart Storage, SQL digunakan untuk menyimpan data transaksi peminjaman dan pengembalian barang, data pengguna, data barang, serta log aktivitas sistem secara otomatis. SQL sangat membantu dalam memastikan bahwa setiap interaksi pengguna tercatat dengan baik dan dapat diakses kembali saat dibutuhkan, baik untuk keperluan monitoring maupun pelaporan.

Karakteristik utama SQL :

1. Struktur Tabel, data disimpan dalam bentuk tabel yang terdiri dari kolom (atribut) dan baris (data), memudahkan pengorganisasian dan pencarian informasi.
2. Bahasa Query – SQL menyediakan perintah-perintah spesifik seperti *SELECT*, *INSERT*, *UPDATE*, dan *DELETE* untuk melakukan manipulasi data.
3. Model Relasional – Setiap tabel saling terhubung melalui kunci (primary key dan foreign key) yang memungkinkan relasi antar data.
4. Transaksi – Mendukung manajemen transaksi (ACID: *Atomicity*, *Consistency*, *Isolation*, *Durability*) yang menjaga integritas data selama proses baca/tulis.
5. Integritas data – SQL mendukung validasi dan batasan (*constraint*) seperti *NOT NULL*, *UNIQUE*, *CHECK*, dan *FOREIGN KEY* untuk memastikan data yang tersimpan benar dan konsisten.

6.Kinerja tinggi – SQL dirancang untuk menangani volume data besar secara cepat melalui indeksasi dan optimasi query.

7.Keamanan data – SQL mendukung pengaturan hak akses pengguna (*privilege*), enkripsi data, dan otorisasi tingkat tabel untuk menjaga kerahasiaan dan integritas data.

Contoh perintah SQL yang umum :

- *Create tabel* : Membuat tabel baru.
- *Insert into* : Menambahkan data baru ke dalam tabel.
- *Select* : Mengambil data dari tabel.
- *Update* : Memperbarui data yang ada dalam tabel.
- *Delete* : Menghapus data dari tabel.

SQL dapat digunakan bersama database populer seperti MySQL, PostgreSQL, atau SQLite, yang dapat terhubung langsung ke mikrokontroler seperti ESP32 melalui API atau server PHP. Dalam proyek *Smart Storage*, SQL bertanggung jawab atas penyimpanan data transaksi secara *real-time* yang dapat ditampilkan kembali melalui antarmuka web atau *dashboard monitoring*.

Dengan pendekatan berbasis SQL, sistem tidak hanya dapat menyimpan data dengan rapi, tetapi juga mampu melakukan analisis data, pencatatan log, serta pelaporan historis yang sangat dibutuhkan dalam manajemen aset kampus. [8]

2.2.1.3 Aplikasi Web

Aplikasi web merupakan antarmuka berbasis browser yang memungkinkan pengguna berinteraksi dengan sistem secara online melalui jaringan internet. Dalam konteks sistem berbasis *Internet of Things* (IoT), aplikasi web berfungsi sebagai media untuk memantau, mengelola, dan menampilkan data secara *real-time* dari perangkat-perangkat yang terhubung, seperti mikrokontroler, sensor, dan actuator.

Pada salah satu penelitian tahun 2023, aplikasi web digunakan untuk memantau suhu mesin secara langsung. ESP32 mengirimkan data dari sensor ke *web dashboard* menggunakan koneksi Wi-Fi, lalu data tersebut divisualisasikan secara *real-time* menggunakan platform seperti Node-RED. Proses ini menunjukkan bagaimana aplikasi web mampu menjembatani komunikasi antara perangkat keras dan pengguna secara efisien.

Fungsi aplikasi web dalam smart storage :

- Mencatat riwayat transaksi : Setiap peminjaman dan pengembalian tercatat secara otomatis dalam database dan dapat diakses melalui halaman web.
- Menampilkan data status barang : Seperti status "dipinjam" atau "tersedia".

- Manajemen pengguna : Admin dapat menambahkan, memantau, atau mengatur akses pengguna yang berhak menggunakan sistem.

Proses pembuatan dan penerapan aplikasi web dalam *smart storage* dilakukan melalui tahapan berikut :

- Perancangan tampilan antarmuka (UI) : Desain halaman web yang memuat informasi penting seperti daftar barang, status, dan log transaksi.
- Pembuatan database online : Untuk menyimpan data transaksi dan informasi barang.
- Koneksi dengan ESP32 : Mikrokontroler ESP32 mengirimkan data ke server web melalui protokol HTTP atau MQTT.

Kriteria aplikasi web untuk smart storage :

- *Real-time Data Sync* : Web harus menampilkan status barang (dipinjam/kembali) secara langsung dari ESP32 tanpa perlu refresh manual.
- Responsif dan *User-Friendly* : Tampilan web dapat diakses dari laptop dan smartphone, serta mudah digunakan oleh admin/laboran.
- Manajemen Transaksi dan Pengguna : Web dapat mencatat aktivitas peminjaman/pengembalian dan mengelola data pengguna yang berwenang.
- Terintegrasi dengan ESP32 : Menerima data dari ESP32 melalui HTTP atau MQTT dan menyimpan ke *database* SQL.
- Keamanan Sistem : Memiliki fitur login dan pembatasan akses agar hanya admin yang bisa melihat dan mengubah data.
- Tersambung dengan Database SQL : Menyimpan data barang, pengguna, dan log transaksi ke dalam sistem manajemen basis data (MySQL/MariaDB).
- Mudah Dikembangkan : Struktur web dibuat modular agar mudah diperbaiki atau dikembangkan lebih lanjut.
- Dibuat dengan Tools Umum Menggunakan : Visual Studio Code untuk coding, PHP/MySQL untuk *backend* dan *database* Draw.io atau Visual Paradigm untuk desain diagram UML

Dengan adanya aplikasi web, sistem *smart storage* tidak hanya berjalan otomatis, tetapi juga dapat dipantau dan dikelola secara digital oleh admin atau pengguna dari mana saja dan kapan saja. [9]

2.2.2 Perangkat Keras

2.2.2.1 ESP32

ESP32 merupakan mikrokontroler utama dalam sistem *Smart Storage* yang berfungsi mengendalikan seluruh komponen seperti RFID, sensor, LCD, dan

aktuator. Modul ini juga bertugas mengirim dan menerima data secara *real-time* melalui jaringan WiFi ke server atau aplikasi web. ESP32 dipilih karena mendukung koneksi nirkabel, memiliki banyak pin input/output, serta efisien dalam mengolah data, sehingga cocok untuk sistem otomatisasi berbasis IoT.

ESP32 membaca data RFID Tag, memproses identitas pengguna, dan secara otomatis mengirimkan data transaksi ke *server web*. Dengan koneksi WiFi yang stabil, ESP32 dapat mengirim informasi ke *database* atau *dashboard monitoring*, sehingga proses peminjaman dan pengembalian barang dapat dicatat dan dipantau secara online tanpa keterlibatan manual.

Mikrokontroler ini dipilih karena memiliki jumlah pin *input* dan *output* yang cukup, daya yang rendah, dan dapat diintegrasikan dengan berbagai sensor dan modul. ESP32 sangat cocok untuk sistem peminjaman barang berbasis IoT yang membutuhkan kontrol otomatis, komunikasi nirkabel, dan pemantauan jarak jauh karena fleksibilitas dan efektif. [9]



Gambar 2.1 ESP32

2.2.2.2 LCD 16x2 I2C

LCD 16x2 I2C digunakan untuk menampilkan informasi seperti nama pengguna, status barang (dipinjam/dikembalikan), dan ID RFID secara langsung. Modul ini hanya memerlukan dua pin (SDA dan SCL), membuat rangkaian lebih rapi dan hemat pin mikrokontroler. LCD I2C sangat cocok digunakan dalam sistem penyimpanan otomatis karena sederhana, cepat dibaca pengguna, dan mendukung feedback visual secara *real-time*.

Modul ini menggunakan protokol komunikasi I2C, yang berarti hanya memerlukan dua pin utama, SDA dan SCL, untuk terhubung dengan mikrokontroler seperti ESP32. Penggunaan I2C juga mengurangi kerumitan pemrograman dan memudahkan integrasi dengan komponen lain yang berbagi pin.

LCD 16x2 I2C dipilih karena bentuknya yang ringkas, hemat daya, mudah diprogram, dan memiliki kemampuan untuk menyajikan informasi secara *real-time* di perangkat penyimpanan. Komponen ini ideal untuk sistem otomatis

berbasis *Internet of Things* yang membutuhkan umpan balik visual yang mudah dan cepat dipahami oleh pengguna. [10]



Gambar 2.2 LCD 16x2 I2C

2.2.2.3 Konveyor

Konveyor berfungsi memindahkan barang secara otomatis dari titik peminjaman ke tempat penyimpanan atau sebaliknya. Dalam sistem ini, konveyor dikendalikan oleh mikrokontroler dan sensor untuk mendeteksi keberadaan barang. Komponen ini dipilih karena dapat meningkatkan efisiensi kerja sistem, meminimalkan keterlibatan manusia, dan mendukung otomatisasi proses secara fisik.

Untuk mengontrol konveyor ini, mikrokontroler (ESP32) bekerja sama dengan sensor infrared untuk mengidentifikasi keberadaan objek di jalur. Jika sensor mendeteksi objek, mikrokontroler akan mengaktifkan atau menghentikan motor penggerak sesuai kebutuhan. Sistem ini memastikan bahwa hanya barang yang telah dikenali atau disetujui yang akan diproses dalam sistem.

Dengan konveyor yang dapat diintegrasikan dengan sistem pemantauan berbasis IoT, Anda dapat memantau status barang secara *real-time* melalui dashboard web. Selain itu, konveyor mampu meningkatkan efisiensi kerja dengan meminimalkan keterlibatan manusia dalam proses pemindahan barang. [11]



Gambar 2.3 Konveyor

2.2.2.4 RFID

RFID digunakan untuk membaca identitas pengguna melalui gelombang radio tanpa kontak langsung. Sistem memanfaatkan RFID Tag pada kartu pengguna, yang dibaca oleh RFID Reader dan dikirim ke sistem untuk diproses. RFID dipilih karena mempercepat proses peminjaman, mengurangi kesalahan input manual, serta mendukung sistem yang efisien dan tanpa kertas.

RFID Reader mendeteksi data dari tag RFID yang ditempelkan dan pengguna dapat diidentifikasi melalui kartu RFID seperti Kartu Tanda Mahasiswa (KTM). Data kemudian diproses oleh mikrokontroler dan dikirim secara otomatis ke aplikasi web untuk dicatat dalam database.

Penggunaan RFID dalam sistem ini dipilih karena mampu mempercepat proses transaksi, mengurangi kesalahan pencatatan manual, dan mendukung sistem kerja yang efisien dan tanpa kertas (paperless). Teknologi ini juga memungkinkan sistem berjalan secara otomatis tanpa harus mengetik atau memindai barcode secara manual, yang membuatnya lebih praktis dan aman untuk digunakan di lingkungan kampus. [12]



Gambar 2.4 RFID

2.2.2.5 Servo

Servo motor digunakan untuk membuka dan menutup laci atau rak penyimpanan secara otomatis. Servo ini dapat digerakkan sesuai sudut yang diatur oleh sistem, misalnya 90° atau 180°, dan langsung berhenti di posisi yang ditentukan. Komponen ini dipilih karena mampu memberikan gerakan presisi, mudah dikontrol, dan sangat sesuai untuk sistem mekanik yang membutuhkan penguncian otomatis.

Mikrokontroler (ESP32) mengontrol servo, yang memiliki kemampuan untuk bergerak sesuai perintah, baik searah maupun berlawanan jarum jam, dan kemudian berhenti di posisi tertentu. Komponen ini sangat cocok untuk sistem yang membutuhkan kontrol gerak yang tepat, seperti pembuka akses loker atau mekanisme penguncian barang.

Servo motor digunakan karena dapat mengatur posisi dengan akurat, mudah diprogram, dan menanggapi perintah dengan cepat. Dengan memasukkannya ke dalam sistem IoT, pergerakan servo dapat dikontrol dan dipantau secara *real-time* melalui aplikasi web. Ini sangat membantu dalam otomatisasi sistem peminjaman dan pengembalian barang. [13]



Gambar 2.5 Servo

2.2.2.6 Sensor TCS3200

Sensor TCS3200 adalah modul pendeteksi warna yang dapat mengenali warna dasar seperti merah, hijau, dan biru (RGB), dan kemudian mengubah intensitas cahaya warna menjadi sinyal digital yang dapat dibaca oleh mikrokontroler. Kemudian, berdasarkan warna yang diterima, sensor menghasilkan frekuensi.

TCS3200 digunakan sebagai sensor tambahan dalam sistem penyimpanan pintar untuk membantu mengenali warna label atau wadah barang. Ini dapat digunakan sebagai identifikasi sekunder selain RFID, terutama jika sistem ingin mengelompokkan barang berdasarkan kategori warna atau menambahkan fitur verifikasi visual.

Sensor TCS3200 dipilih karena akurasi pendeteksian warnanya yang tinggi, kemampuan untuk bekerja dengan baik pada jarak sekitar 6,5 cm, dan kemampuan untuk diintegrasikan dengan mikrokontroler seperti ESP32. Dengan penempatan yang tepat, TCS3200 dapat meningkatkan akurasi pengelompokan barang dan mendukung sistem penyimpanan berbasis warna. [14]



Gambar 2.6 Sensor TCS3200

2.2.2.7 Sensor Infrared

Sensor Infrared adalah komponen yang digunakan untuk mendeteksi keberadaan objek dengan memancarkan dan menerima sinyal inframerah yang tidak terlihat oleh mata manusia. Jika sinyal yang dipancarkan oleh objek terhalang, sensor ini akan merespons dan kemudian mengirimkan sinyal logika ke mikrokontroler.

Sensor infrared dalam sistem penyimpanan pintar dapat dipasang pada konveyor untuk mengetahui apakah barang telah diletakkan atau diambil dari tempat penyimpanan. Ini secara otomatis mendeteksi pergerakan barang tanpa campur tangan manual. Dengan mengirimkan sinyal ke ESP32, sistem dapat mengetahui kapan barang telah diambil atau dikembalikan dan mencatatnya ke dalam sistem secara *real-time*.

Sensor infrared dipilih karena memiliki respons cepat, tidak mengonsumsi banyak daya, dan dapat diintegrasikan dengan mikrokontroler seperti ESP32. Penggunaan sensor infrared secara otomatis meningkatkan keakuratan sistem dan mengurangi kesalahan dalam proses pencatatan barang. [15]



Gambar 2.7 Sensor Infrared

2.2.2.8 Motor Stepper

Motor stepper adalah jenis motor yang bergerak dalam langkah-langkah kecil (step) dan sangat presisi sehingga cocok untuk sistem otomatisasi. Motor ini dapat bergerak dan berhenti pada posisi tertentu tanpa memerlukan sensor posisi tambahan.

Motor stepper digunakan dalam sistem penyimpanan pintar sebagai penggerak konveyor. Fungsi ini memungkinkan sistem mengatur jarak gerak, seperti memindahkan barang ke titik pemindaian RFID atau mengarahkan konveyor ke posisi yang ditetapkan.

Motor stepper dipilih karena kestabilannya dalam gerakan, kemampuan kontrol posisi yang akurat, dan kemudahan integrasi dengan mikrokontroler seperti ESP32 melalui driver seperti A4988. Penggunaan motor ini membantu sistem bekerja secara otomatis, teratur, dan efisien dalam proses peminjaman dan pengembalian barang. [16]



Gambar 2.8 Motor Stepper

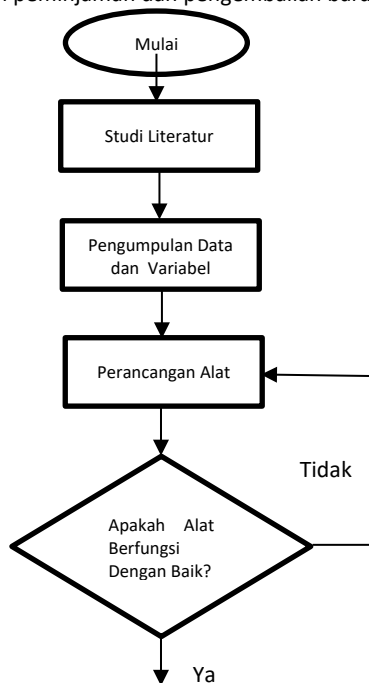
Bab 3. Metodologi Penelitian

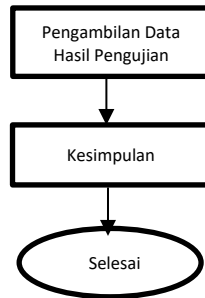
3.1 Perancangan

Pada tahap perancangan ini dilakukan proses perencanaan dan pengembangan sistem Smart Storage yang meliputi perancangan perangkat keras dan perangkat lunak. Perancangan dilakukan untuk memastikan sistem yang dibangun dapat berjalan sesuai dengan kebutuhan, mulai dari proses identifikasi pengguna, peminjaman dan pengembalian barang, hingga pengolahan dan penyimpanan data. Selain itu, tahap ini juga bertujuan untuk menentukan alur kerja sistem secara keseluruhan agar proses implementasi dapat dilakukan secara terstruktur dan sistematis.

3.1.1 Perancangan Tahapan Penelitian

Diagram alur berikut menunjukkan tahapan penelitian yang dilakukan dalam pengembangan sistem peminjaman dan pengembalian barang otomatis.





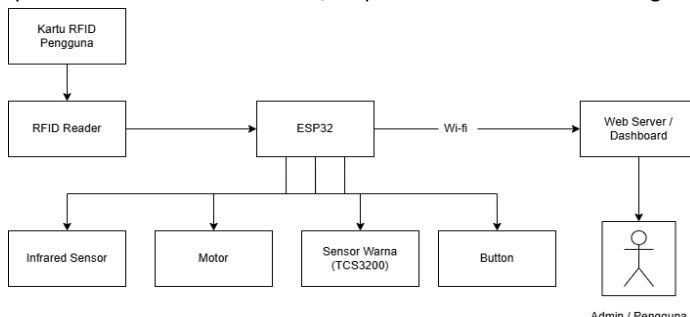
Gambar 3.1 Flowchart Penelitian

Pada gambar 3.1 dijelaskan bahwa sebelum memulai perancangan alat, terdapat beberapa tahapan yang harus dilakukan. Tahap pertama adalah melakukan studi literatur terkait alat yang akan dirancang, seperti mencari referensi jurnal dan literatur yang relevan. Selanjutnya, dilakukan pengumpulan data dan variabel yang diperlukan sebagai dasar dalam perancangan alat. Setelah data dan variabel terkumpul, tahap berikutnya adalah proses perancangan alat.

Jika alat berfungsi dengan baik, langkah berikutnya adalah pengambilan data hasil pengujian untuk dianalisis dan diambil kesimpulannya. Namun, apabila alat belum berfungsi dengan optimal, maka perlu dilakukan evaluasi dan perbaikan pada tahap perancangan hingga alat dapat bekerja dengan baik. Proses ini dilakukan secara berulang sampai diperoleh hasil yang optimal dan data pengujian yang valid.

3.1.2 Perancangan Blok Diagram Sistem

Untuk memberikan gambaran menyeluruh mengenai hubungan antar komponen utama dalam sistem, diperlukan sebuah blok diagram yang



Gambar 3.2 Blok Diagram Sistem

merepresentasikan alur kerja dari sistem secara umum. Blok diagram ini menunjukkan bagaimana input dari pengguna diproses oleh kontroler dan menghasilkan output sesuai fungsi yang diharapkan. Berikut merupakan diagram blok sistem peminjaman dan pengembalian barang otomatis yang dirancang untuk mendukung efisiensi pengelolaan aset berbasis RFID dan IoT.

Sistem peminjaman dan pengembalian barang otomatis berbasis RFID dan IoT ini dirancang untuk memudahkan pengelolaan barang di lingkungan kampus. Sistem ini terdiri dari tiga bagian utama, yaitu input, kontroler, dan output. Bagian input menggunakan kartu RFID yang dimiliki oleh pengguna. Kartu ini dibaca oleh *RFID Reader* untuk proses verifikasi identitas saat melakukan peminjaman atau pengembalian barang. Beberapa komponen tambahan seperti sensor warna (TCS3200), sensor inframerah, dan tombol juga dapat digunakan untuk mendukung proses identifikasi atau interaksi sistem secara manual jika diperlukan.

Kontrol utama sistem berada pada modul ESP32 yang berfungsi sebagai mikrokontroler. ESP32 menerima data dari input, lalu mengirimkannya ke web server melalui koneksi Wi-Fi. Sistem ini terhubung ke *dashboard* berbasis web yang menjadi pusat manajemen data. Pada sistem ini, pengguna hanya memiliki akses untuk melakukan proses peminjaman dan pengembalian barang, tanpa bisa melihat atau mengedit informasi lain dalam sistem. Proses tersebut dapat dilakukan melalui interaksi langsung dengan RFID Reader yang terintegrasi dengan ESP32.

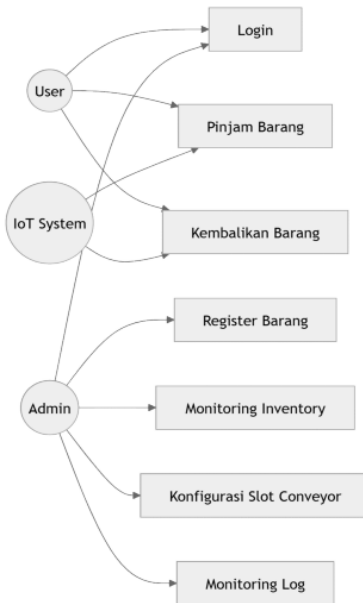
Sementara itu, admin memiliki akses penuh terhadap *dashboard web*. Admin dapat memantau data peminjaman, mengelola informasi barang, melihat status pemakaian, serta memperbarui data inventaris.

3.1.3 Perancangan Sistem Dengan Desain UML

Pada penelitian ini, UML (Unified Modeling Language) digunakan sebagai metode perancangan sistem untuk memodelkan proses dan alur kerja pada sistem Smart Storage. Penggunaan UML bertujuan untuk membantu proses pengembangan sistem agar lebih terstruktur serta mempermudah analisis kebutuhan sistem sebelum tahap implementasi dilakukan. Diagram UML yang digunakan dalam penelitian ini terdiri dari Use Case Diagram dan Activity Diagram. Use Case Diagram digunakan untuk menggambarkan interaksi antara aktor dengan fungsi-fungsi utama pada sistem, sedangkan Activity Diagram digunakan untuk menggambarkan alur proses peminjaman dan pengembalian barang secara terstruktur.

3.1.3.1 Use Case Diagram

Gambar berikut menampilkan Use Case Diagram dari sistem peminjaman dan pengembalian barang otomatis. Diagram ini digunakan untuk menggambarkan interaksi antara aktor (admin dan user) dengan fungsi-fungsi utama yang tersedia dalam sistem.



Gambar 3.3 Use Case Diagram

Use Case Diagram pada penelitian ini digunakan sebagai bagian dari metode *Unified Modeling Language* (UML) untuk menggambarkan hubungan antara aktor dengan fungsi-fungsi yang tersedia pada sistem Smart Storage. Diagram ini digunakan untuk membantu mengidentifikasi kebutuhan fungsional sistem, hak akses pengguna, serta interaksi yang terjadi antara pengguna dengan sistem. Dengan adanya Use Case Diagram, proses perancangan sistem dapat dilakukan secara lebih terstruktur sehingga mempermudah tahap analisis kebutuhan sebelum implementasi sistem dilakukan.

Pada penelitian ini, Use Case Diagram digunakan untuk memodelkan fungsi-fungsi utama yang terdapat pada sistem Smart Storage, seperti proses login, peminjaman barang, pengembalian barang, pengelolaan inventaris, konfigurasi

sistem, serta monitoring aktivitas sistem. Melalui diagram ini, hubungan antara aktor dan fungsi sistem dapat divisualisasikan dengan lebih jelas sehingga mempermudah proses pengembangan sistem.

Proses perancangan Use Case Diagram pada penelitian ini dilakukan menggunakan aplikasi draw.io sebagai media perancangan diagram sistem. Tahapan perancangan dilakukan melalui beberapa langkah sebagai berikut:

1. Analisis kebutuhan sistem

Tahap awal dilakukan dengan mengidentifikasi kebutuhan sistem berdasarkan permasalahan yang ditemukan pada proses peminjaman dan pengembalian barang secara manual. Pada tahap ini dilakukan analisis terhadap fungsi yang dibutuhkan oleh pengguna dan sistem.

2. Identifikasi aktor dan fungsi sistem

Setelah kebutuhan sistem diperoleh, dilakukan identifikasi terhadap aktor yang terlibat pada sistem Smart Storage, yaitu User (Mahasiswa), Admin, dan IoT System. Selain itu ditentukan juga fungsi-fungsi yang dapat dilakukan oleh setiap aktor seperti login, peminjaman barang, pengembalian barang, monitoring inventory, konfigurasi slot, dan monitoring log.

3. Pembuatan Use Case Diagram

Tahap selanjutnya dilakukan perancangan Use Case Diagram menggunakan aplikasi draw.io untuk menggambarkan hubungan antara aktor dengan fungsi yang tersedia pada sistem. Diagram ini digunakan untuk memvisualisasikan kebutuhan fungsional sistem secara keseluruhan.

4. Validasi perancangan sistem

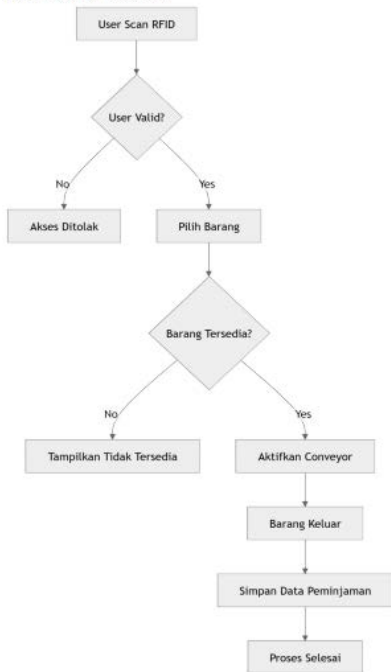
Tahap terakhir dilakukan pengecekan kembali terhadap diagram yang telah dibuat untuk memastikan bahwa seluruh alur sistem sesuai dengan kebutuhan dan implementasi Smart Storage yang dikembangkan.

Dengan menggunakan Use Case Diagram, proses perancangan sistem Smart Storage menjadi lebih terstruktur, mudah dipahami, serta dapat dijadikan sebagai acuan pada tahap implementasi sistem.

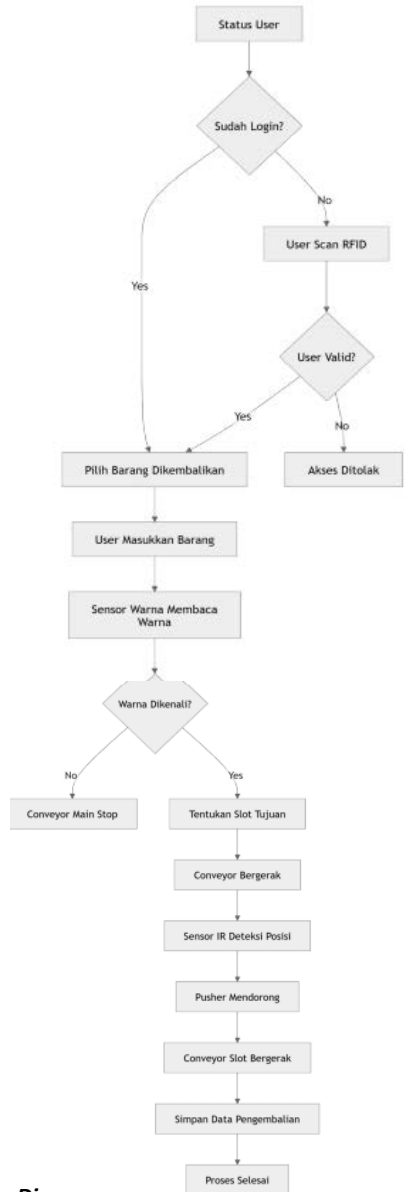
3.1.3.2 Activity Diagram

Gambar berikut menyajikan *Activity Diagram* dari sistem peminjaman dan pengembalian barang otomatis berbasis RFID dan IoT. Diagram ini menggambarkan urutan aktivitas dari proses awal pengguna berinteraksi dengan sistem hingga transaksi selesai dilakukan, baik untuk peminjaman maupun pengembalian barang.

Activity Diagram - Peminjaman Barang



Activity Diagram - Pengembalian Barang



Gambar 3.4 Activity Diagram

Activity Diagram pada penelitian ini digunakan untuk menggambarkan alur aktivitas sistem Smart Storage secara berurutan mulai dari proses awal hingga proses selesai. Diagram ini digunakan untuk memvisualisasikan proses kerja sistem sehingga alur peminjaman dan pengembalian barang dapat dipahami dengan lebih jelas. Selain itu, Activity Diagram juga membantu dalam mengidentifikasi proses validasi, pengambilan keputusan (decision), serta interaksi antar komponen selama sistem berjalan.

Proses perancangan Activity Diagram pada penelitian ini dilakukan menggunakan aplikasi draw.io sebagai media perancangan diagram sistem. Tahapan perancangan Activity Diagram dilakukan melalui beberapa langkah sebagai berikut:

1. Identifikasi proses sistem

Tahap awal dilakukan dengan mengidentifikasi proses utama yang terdapat pada sistem Smart Storage berdasarkan kebutuhan sistem yang telah dianalisis sebelumnya. Proses yang diidentifikasi meliputi autentikasi pengguna menggunakan RFID, peminjaman barang, pengembalian barang, pembacaan sensor, pengendalian conveyor, dan penyimpanan data transaksi.

2. Menentukan urutan aktivitas

Setelah proses utama diidentifikasi, langkah berikutnya adalah menentukan urutan aktivitas berdasarkan mekanisme kerja sistem yang sebenarnya. Aktivitas disusun mulai dari interaksi pengguna dengan sistem hingga proses transaksi selesai dilakukan.

3. Menentukan kondisi keputusan (decision)

Pada tahap ini dilakukan identifikasi terhadap kondisi yang memerlukan pengambilan keputusan dalam sistem, seperti validasi pengguna, ketersediaan barang, dan identifikasi warna barang. Kondisi tersebut digunakan sebagai percabangan proses pada Activity Diagram.

4. Pembuatan Activity Diagram menggunakan draw.io

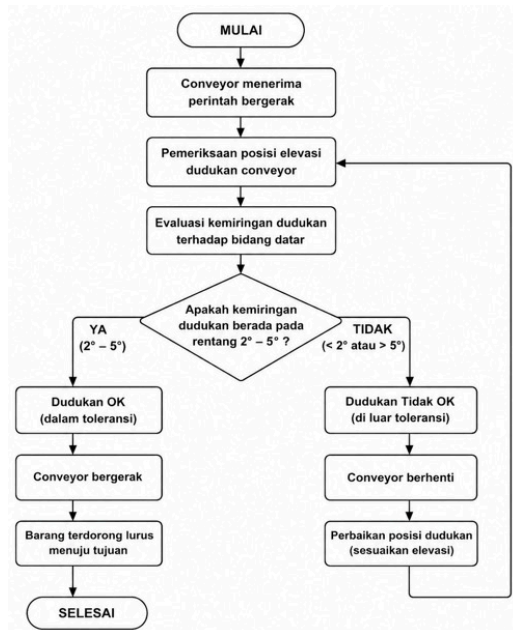
Setelah seluruh aktivitas dan kondisi ditentukan, Activity Diagram dibuat menggunakan aplikasi draw.io dengan menggunakan simbol UML seperti initial node, activity, decision, flow, dan final node. Diagram disusun sesuai dengan alur kerja sistem Smart Storage yang sebenarnya.

5. Validasi diagram

Tahap terakhir dilakukan pengecekan kembali terhadap Activity Diagram yang telah dibuat untuk memastikan bahwa seluruh alur aktivitas telah sesuai dengan kebutuhan sistem dan implementasi perangkat yang digunakan.

Dengan adanya *Activity Diagram*, alur kerja pada sistem *Smart Storage* dapat divisualisasikan secara lebih jelas dan terstruktur, sehingga dapat mempermudah proses analisis, pengembangan, serta implementasi sistem. Selain itu, diagram yang telah dirancang juga dapat digunakan sebagai acuan untuk memastikan bahwa seluruh proses peminjaman dan pengembalian barang berjalan sesuai dengan kebutuhan dan fungsi sistem yang telah ditentukan.

Untuk mendukung pergerakan barang yang stabil pada sistem *Smart Storage*, dilakukan perancangan alur gerak conveyor yang menggambarkan proses perpindahan barang dari titik awal hingga mencapai slot tujuan. Selain itu, aspek mekanik seperti elevasi conveyor juga menjadi faktor yang perlu diperhatikan karena mempengaruhi kelancaran pergerakan barang, kestabilan posisi dudukan, serta mengurangi potensi terjadinya hambatan selama proses peminjaman dan pengembalian barang.



Gambar 3.5 Activity Diagram Gerak Conveyor

Flowchart gerak conveyor digunakan untuk menggambarkan proses kerja conveyor serta hubungan antara posisi elevasi dengan kondisi dudukan conveyor. Pada sistem *Smart Storage*, elevasi berfungsi sebagai dudukan atau penyangga

conveyor agar posisi jalur perpindahan barang tetap sejajar. Kondisi dudukan sangat mempengaruhi pergerakan barang ketika proses peminjaman maupun pengembalian berlangsung.

Proses dimulai ketika conveyor menerima perintah untuk bergerak. Selanjutnya dilakukan pemeriksaan terhadap posisi elevasi dudukan conveyor untuk mengetahui apakah posisi dudukan berada pada kondisi yang sesuai atau tidak. Setelah itu dilakukan evaluasi kemiringan dudukan terhadap bidang datar sebagai parameter untuk menentukan kelayakan posisi dudukan.

Pada tahap pengambilan keputusan, sistem memeriksa apakah kemiringan dudukan berada pada rentang 2° – 5° . Rentang tersebut digunakan sebagai batas toleransi agar posisi dudukan masih dapat menjaga kestabilan pergerakan barang. Apabila kemiringan dudukan berada dalam rentang 2° – 5° , maka kondisi dudukan dinyatakan OK atau masih berada dalam batas toleransi. Pada kondisi tersebut conveyor dapat bergerak secara normal sehingga barang dapat terdorong lurus menuju tujuan tanpa mengalami gangguan selama proses perpindahan.

Sebaliknya, apabila kemiringan dudukan berada di luar rentang toleransi (kurang dari 2° atau lebih dari 5°), maka kondisi dudukan dinyatakan Tidak OK. Kondisi tersebut dapat menyebabkan jalur conveyor tidak sejajar sehingga pergerakan barang menjadi tidak stabil, berpotensi bergeser, tersangkut, atau tidak bergerak sesuai lintasan yang telah ditentukan. Oleh karena itu conveyor dihentikan dan dilakukan penyesuaian posisi elevasi hingga dudukan kembali berada pada kondisi yang sesuai. Setelah posisi dudukan diperbaiki, proses kembali ke tahap pemeriksaan elevasi untuk memastikan conveyor dapat beroperasi secara normal.

3.1.4 Komunikasi Data Menggunakan Protokol HTTP

Pada sistem Smart Storage berbasis IoT ini, komunikasi data antara perangkat mikrokontroler ESP32 dan server dilakukan menggunakan protokol HTTP (Hypertext Transfer Protocol). HTTP merupakan protokol komunikasi berbasis request-response yang umum digunakan pada aplikasi berbasis web. Protokol ini dipilih karena lebih sederhana dan mudah diimplementasikan untuk pengiriman data antara perangkat IoT dan server lokal.

Dalam sistem ini, ketika pengguna menempelkan kartu RFID, ESP32 membaca UID dari kartu tersebut, kemudian mengirimkan data hasil pembacaan ke web server lokal melalui HTTP POST request. Server lokal dijalankan menggunakan aplikasi XAMPP, yang berfungsi sebagai web server dan database penyimpanan

data transaksi. Struktur komunikasi HTTP dalam sistem ini terdiri dari dua komponen utama :

- Client : ESP32 bertindak sebagai pengirim data (request sender) yang mengirimkan data hasil pembacaan RFID ke server.
- Server : Sistem web lokal yang dijalankan menggunakan XAMPP bertugas menerima data dari ESP32, memprosesnya menggunakan skrip PHP, dan menyimpannya ke dalam database MySQL.

Implementasi HTTP pada sistem ini memberikan beberapa keuntungan, antara lain:

- Mudah diimplementasikan : Tidak memerlukan konfigurasi broker seperti pada MQTT, sehingga lebih praktis untuk uji coba lokal.
- Kompatibel dengan web server lokal: HTTP dapat dijalankan langsung di server XAMPP tanpa perlu koneksi internet eksternal.
- Dapat digunakan untuk debugging: Data yang dikirim dan diterima dapat diuji melalui browser atau aplikasi seperti Postman.

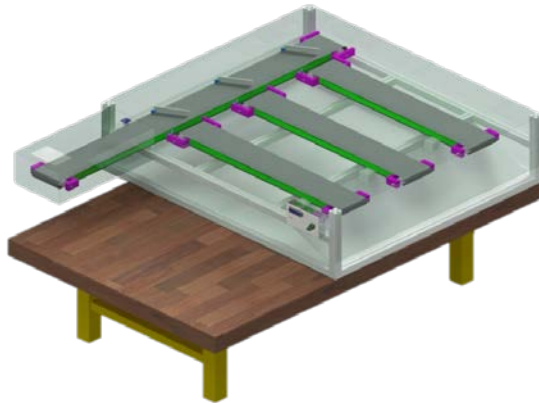
Berikut tabel penggunaan komponen HTTP :

Tabel 3.1 Penggunaan Komponen HTTP

Komponen	Peran	Penjelasan
ESP32	Client	Mengirim data UID pengguna/barang ke web server melalui protokol HTTP menggunakan metode <i>POST request</i> .
Web server	Penghubung data	Menerima data dari ESP32, memprosesnya melalui skrip PHP, dan meneruskannya ke database.
Database (MySQL)	Penyimpanan data	Menyimpan data hasil pengiriman dari ESP32 untuk ditampilkan pada dashboard web.

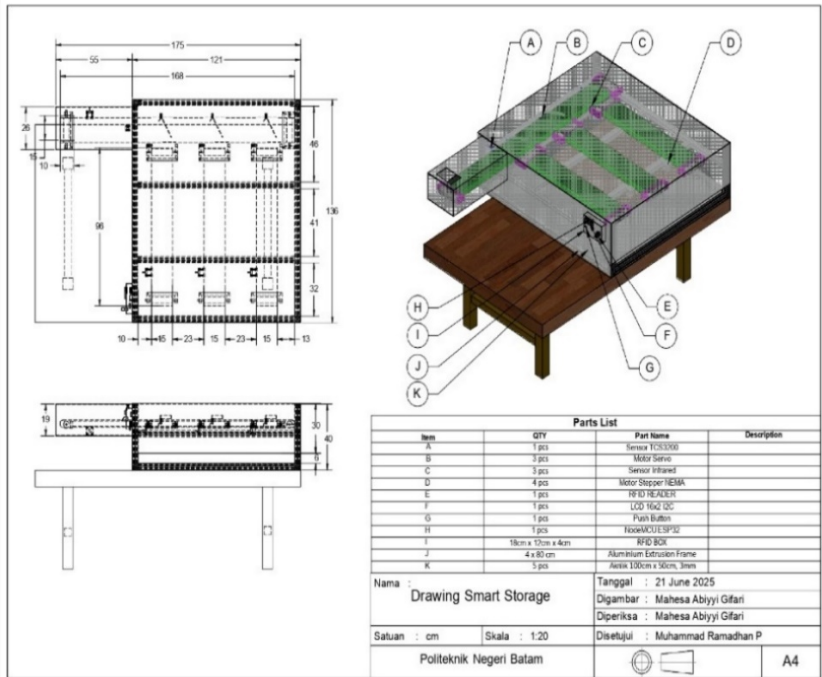
3.2 Perancangan Mekanikal

Perancangan mekanikal dilakukan untuk merancang bentuk fisik dan struktur sistem Smart Storage agar mampu menjalankan fungsi peminjaman dan pengembalian barang secara otomatis. Desain mekanikal ini mencakup penyusunan komponen seperti rangka, konveyor,udukan sensor, serta posisi modul elektronik agar terintegrasi secara fungsional dan efisien dalam lingkungan laboratorium kampus.



Gambar 3.6 Desain 3D Mekanikal

Desain mekanik sistem Smart Storage ini dibuat dengan konsep sederhana untuk mendukung proses peminjaman dan pengembalian barang secara otomatis. Sistem dirancang menggunakan konveyor yang digerakkan oleh motor stepper untuk menggerakkan barang dari titik awal menuju sensor identifikasi (TCS3200). Beberapa komponen elektronik seperti RFID reader, sensor infrared, LCD, dan push button diposisikan secara strategis agar mudah diakses dan terhubung dengan mikrokontroler ESP32. Penempatan komponen mengikuti alur logis kerja sistem, sehingga pengguna cukup meletakkan atau mengambil barang pada area yang telah ditentukan. Seluruh struktur dirancang modular agar praktis dan hemat tempat, namun tetap mampu menjalankan fungsi utama sistem secara optimal.

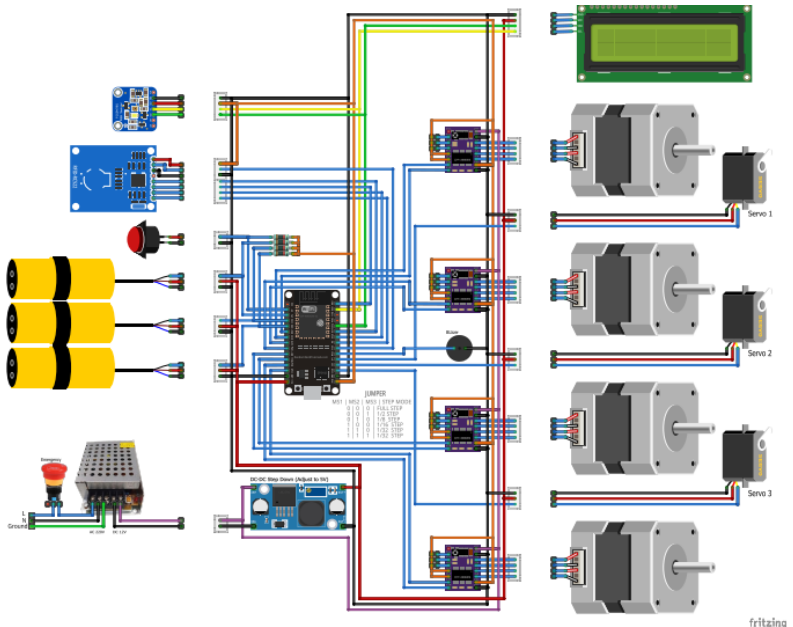


Gambar 3.7 Desain Umum Mekanikal

Berdasarkan gambar teknik, desain ini memiliki ukuran panjang 100 cm dan lebar 50 cm, dengan skala gambar 1:20. Beberapa komponen utama terlihat jelas, seperti sensor TCS3200 untuk deteksi warna barang, servo motor untuk membuka atau menutup laci, serta sensor infrared yang digunakan untuk mendeteksi keberadaan barang di jalur konveyor. Di bagian depan sistem terdapat *RFID Reader* RC522 yang membaca identitas pengguna atau barang, dan LCD 16x2 I2C sebagai media penampil status sistem secara *real-time*. Tambahan seperti push button digunakan untuk input manual, dan semua perangkat dikendalikan oleh NodeMCU ESP32 yang juga menghubungkan data ke *server* melalui koneksi Wi-Fi.

3.3 Perancangan Elektrikal

Pada tahap ini dilakukan perancangan sistem elektrikal yang terdiri dari rangkaian komponen utama seperti mikrokontroler, sensor, pembaca RFID, serta aktuator yang terintegrasi dalam sistem. Gambar berikut menunjukkan skema hubungan antar komponen elektronik yang digunakan dalam sistem peminjaman dan pengembalian barang otomatis berbasis RFID dan IoT.



Gambar 3.8 Desain Elektrikal

Pada desain elektrikal ini, ESP32 berfungsi sebagai pusat kendali yang mengatur seluruh proses sistem, mulai dari pembacaan data RFID, pengolahan data, hingga pengendalian aktuator. Pemilihan ESP32 didasarkan pada kemampuannya dalam mendukung komunikasi berbasis IoT melalui WiFi serta memiliki performa yang cukup untuk menjalankan sistem secara real-time.

Pada bagian aktuator, sistem menggunakan motor stepper sebagai penggerak utama mekanisme conveyor. Motor stepper dipilih karena memiliki keunggulan dalam hal presisi pergerakan dan kontrol posisi yang baik, sehingga sangat sesuai untuk sistem yang membutuhkan pergerakan terarah dan stabil. Motor yang digunakan memiliki torsi yang cukup untuk menggerakkan beban pada sistem

conveyor, sehingga mampu menjaga kestabilan pergerakan tanpa terjadi slip selama proses operasional. Selain itu, digunakan driver motor untuk mengatur arus dan arah putaran motor agar dapat dikontrol secara optimal oleh mikrokontroler.

Selain motor stepper, sistem juga menggunakan servo motor sebagai penggerak mekanisme tambahan yang membutuhkan kontrol sudut tertentu, seperti proses buka-tutup atau pengarah barang. Servo dipilih karena kemampuannya dalam mengatur posisi secara akurat dengan kontrol sederhana berbasis sinyal PWM, serta memiliki torsi yang cukup untuk menggerakkan komponen ringan pada sistem.

Pada bagian sensor, modul RFID digunakan sebagai media identifikasi pengguna dan barang dengan frekuensi kerja 13,56 MHz, yang mampu membaca kartu secara stabil pada jarak dekat. Sensor tambahan seperti sensor IR dan sensor warna digunakan untuk mendeteksi keberadaan dan jenis barang dalam sistem.

Dengan pemilihan komponen yang sesuai serta integrasi rangkaian yang tepat, sistem elektrik yang dirancang mampu mendukung proses peminjaman dan pengembalian barang secara otomatis dengan kinerja yang stabil dan terkontrol.

3.4 Pengujian

Pengujian sistem diawali dengan verifikasi terhadap alur kerja yang telah dimodelkan dalam diagram UML, seperti Use Case, Activity, Sequence, dan Class Diagram. Pengujian dilakukan dengan mencocokkan tiap langkah pada diagram dengan implementasi nyata di alat. Misalnya, pada Use Case Diagram, diuji perbedaan hak akses dan fungsi antara aktor Admin dan User. Pada Activity Diagram, langkah-langkah proses dipastikan sesuai dengan alur aktual peminjaman dan pengembalian barang. Sequence Diagram divalidasi dengan memantau urutan komunikasi data dari RFID Reader ke ESP32 dan ke server, sesuai skenario sistem. Sedangkan Class Diagram dikonfirmasi dengan mengecek struktur dan relasi data di database sistem. Pengujian ini bertujuan memastikan bahwa rancangan sistem secara logika telah diterapkan dengan benar pada implementasi perangkat keras dan lunaknya.

Selanjutnya, pengujian sistem dilaksanakan terhadap seluruh komponen utama, meliputi proses peminjaman barang menggunakan identifikasi RFID, proses pengembalian barang dengan sensor pendeteksi warna, performa modul RFID dalam akurasi pembacaan dan stabilitas deteksi, serta evaluasi kinerja ESP32 dalam pemrosesan data dan konektivitas jaringan. Pengujian dilakukan melalui skenario operasional dengan parameter seperti waktu respons sistem, tingkat

akurasi transaksi, dan konsistensi performa perangkat. Hasil pengujian menjadi acuan dalam mengevaluasi keandalan sistem serta identifikasi area yang perlu dioptimalkan sebelum sistem diimplementasikan secara luas di lingkungan kampus.

Pada aspek Internet of Things (IoT), sistem diuji menggunakan metode komunikasi HTTP (Hypertext Transfer Protocol) sebagai protokol utama dalam pengiriman data dari ESP32 ke server lokal. Pengujian difokuskan pada efisiensi pengiriman data, kestabilan koneksi jaringan Wi-Fi, serta waktu tunda (latency) antara pembacaan data RFID oleh ESP32 dan penerimaan data pada dashboard web. Hasil pengujian menunjukkan bahwa protokol HTTP mampu mengirimkan data dengan cepat dan stabil pada jaringan lokal menggunakan server XAMPP, sehingga metode ini dinilai efektif untuk implementasi sistem Smart Storage.

3.4.1 Pengujian Sistem Peminjaman Barang

Pengujian ini dilakukan untuk memastikan bahwa proses peminjaman barang melalui sistem konveyor dapat berjalan dengan baik sesuai alur yang dirancang. Pengujian melibatkan deteksi kartu RFID oleh pengguna, pembacaan tag barang, serta respons sistem dalam mengaktifkan konveyor untuk mengantarkan barang. Tahapan ini bertujuan untuk mengevaluasi keakuratan sistem dalam mengidentifikasi peminjam dan memastikan barang yang dipinjam tercatat dengan benar.

Tabel 3.2 Pengujian Sistem Peminjaman Barang

No	Nama Alat (Warna)	Pengujian					Tingkat Keberhasilan (%)
		1	2	3	4	5	
1	Arduino Uno (Hijau)	✓	✓	X	✓	✓	80%
2	ESP32 (Merah)	✓	✓	✓	✓	✓	100 %
3	PLC Omron (Biru)	✓	✓	✓	X	✓	80%

3.4.2 Pengujian Sistem Pengembalian Barang

Pengujian difokuskan pada kinerja sensor TCS3200 dalam mendeteksi warna barang saat proses pengembalian. Warna dibaca oleh sensor, lalu diproses oleh ESP32 untuk mencocokkan identitas barang. Pengujian menggunakan beberapa

sampel warna untuk mengevaluasi akurasi, kecepatan, dan konsistensi sensor dalam berbagai kondisi cahaya. Hasilnya digunakan untuk memastikan sistem dapat memvalidasi pengembalian barang secara otomatis dengan kesalahan minimal.

Tabel 3.3 Pengujian Sistem Pengembalian Barang

No	Nama Alat (Warna)	Pengujian					Tingkat Keberhasilan (%)
		1	2	3	4	5	
1	Arduino Uno (Hijau)	X	✓	✓	✓	✓	80%
2	ESP32 (Merah)	✓	✓	X	✓	X	60 %
3	PLC Omron (Biru)	✓	X	✓	✓	✓	80%

3.4.3 Pengujian pada Modul RFID

Pengujian ini mengevaluasi kinerja modul RFID dalam membaca UID dari kartu atau tag pengguna. Pengujian dilakukan dengan berbagai skenario seperti jarak, sudut, dan kecepatan pembacaan. Parameter yang diamati meliputi tingkat keberhasilan, waktu respon, dan kestabilan koneksi dengan ESP32. Tujuannya untuk memastikan modul RFID berfungsi optimal dan konsisten dalam sistem *Smart Storage*.

Tabel 3.4 Pengujian pada Modul RFID

No	Jarak Pembacaan (cm)	Posisi Kartu	Jumlah Percobaan	Jumlah Berhasil	Tingkat Keberhasilan (%)	Keterangan
1	1 cm	Datar di depan reader	5	4	80%	Pembacaan sangat stabil
2	3 cm	Datar sedikit miring	5	2	40%	Pembacaan stabil
3	5 cm	Miring / tidak sejajar	5	0	0%	Kadang perlu penyesuaian posisi

N o	Jarak Pembacaan (cm)	Posisi Kartu	Jumlah Percobaan	Jumlah Berhasil	Tingkat Keberhasilan (%)	Keterangan
4	> 5 cm	Jauh dari reader	5	0	0%	Pembacaan kurang stabil

3.4.4 Pengujian Modul ESP32 dalam Pemrosesan Data

Pengujian ini dilakukan untuk memastikan bahwa modul ESP32 dapat berfungsi sebagai pusat pemrosesan data dalam sistem. Fokus pengujian mencakup kemampuan ESP32 dalam menerima data dari modul RFID dan sensor, kemudian mengirimkan data tersebut ke server lokal melalui jaringan HTTP menggunakan metode POST request.

Hasil pengujian menunjukkan sejauh mana keandalan ESP32 dalam menjaga kestabilan koneksi dan waktu respon selama proses komunikasi data antar komponen sistem.

Tabel 3.5 Pengujian Modul ESP32 dalam Pemrosesan Data

N o	Fitur Pengujian	Deskripsi Pengujian	Akurasi Pemrosesan Data	Waktu Respon (Detik)	Stabilitas Koneksi (%)	Keterangan
1	Pengiriman Data RFID	ESP32 mengirim UID RFID ke server via HTTP	90%	1.5 detik	95%	Data sebagian besar berhasil dikirim
2	Penerimaan Perintah Server	ESP32 menerima instruksi dari server	85%	1.3 detik	90%	Masih terdapat delay pada beberapa percobaan
3	Sinkronisasi Database	ESP32 kirim data transaksi	80%	1.7 detik	90%	Sinkronisasi berjalan namun

N o	Fitur Pengujian	Deskripsi Pengujian	Akurasi Pemrosesan Data	Waktu Respon (Detik)	Stabilitas Koneksi (%)	Keterangan
		ike MySQL				belum optimal
4	Pemrosesan Paralel	ESP32 menjalankan beberapa tugas Bersamaan	75%	2.0 detik	85%	Terjadi penurunan performa saat multitasking

3.4.5 Pengujian Sistem Berdasarkan Diagram UML

Pengujian ini bertujuan untuk memastikan bahwa alur sistem yang telah dirancang menggunakan diagram UML dapat berjalan sesuai pada implementasi alat. Setiap diagram seperti *Use Case*, *Activity*, *Sequence*, dan *Class Diagram* diuji melalui simulasi dan pengamatan langsung untuk melihat kesesuaian proses aktual dengan rancangan yang telah dibuat. Dengan demikian, pengujian ini memberikan validasi terhadap desain sistem secara struktural maupun fungsional.

Tabel 3.6 Pengujian Sistem Berdasarkan Diagram UML

No	Jenis UML	Kasus Uji	Hasil Implementasi Sistem	Kesesuaian dengan UML	Keterangan
1	Use Case Diagram	Pengguna melakukan login untuk mengakses sistem	Sistem dapat menjalankan proses login dengan beberapa kendala kecil	85%	Proses login direncanakan berjalan dengan baik
2	Activity Diagram	Proses peminjaman barang dilakukan oleh pengguna	Alur peminjaman berjalan sesuai tahapan namun masih terdapat delay	80%	Alur sistem sesuai dengan diagram yang dirancang
3	Sequence Diagram	Komunikasi antara ESP32 dan server saat	Data dapat dikirim dan diterima dengan beberapa keterlambatan	80%	Komunikasi sistem direncanakan berjalan cukup baik

No	Jenis UML	Kasus Uji	Hasil Implementasi Sistem	Kesesuaian dengan UML	Keterangan
		pengiriman data RFID			
4	Class Diagram	Relasi antar tabel User, Barang, dan Transaksi	Struktur data sudah saling terhubung namun masih perlu penyempurnaan	75%	Relasi database direncanakan sesuai dengan desain UML

3.4.6 Pengujian Sistem IoT Menggunakan HTTP

Pengujian ini dilakukan untuk mengevaluasi komunikasi data pada sistem Internet of Things (IoT) menggunakan protokol HTTP (Hypertext Transfer Protocol) sebagai media utama pertukaran data antara ESP32 dan server lokal. Tujuan pengujian ini adalah untuk memastikan proses pengiriman dan penerimaan data berjalan dengan baik melalui metode HTTP POST request.

Pengujian difokuskan pada kecepatan respon, akurasi data yang dikirim, dan stabilitas koneksi jaringan Wi-Fi selama proses peminjaman dan pengembalian barang berlangsung. Hasil pengujian menunjukkan bahwa penggunaan protokol HTTP mampu mengirimkan data dengan cepat dan stabil pada jaringan lokal menggunakan server XAMPP, sehingga sistem dapat bekerja secara efisien dan responsif dalam mendukung proses otomatisasi Smart Storage.

Tabel 3.7 Pengujian Sistem IoT Menggunakan HTTP

No	Parameter Pengujian	Deskripsi Pengujian	Hasil Uji	Keterangan
1	Kecepatan Respon	Mengukur waktu antara pembacaan UID RFID oleh ESP32 hingga data diterima server lokal melalui HTTP	±2 detik	Waktu respon direncanakan cukup cepat
2	Akurasi Data	Membandingkan data UID yang dikirim dari ESP32 dengan data yang tersimpan di database	85%	Data direncanakan cukup akurat namun masih terdapat kemungkinan error
3	Stabilitas Koneksi	Pengujian dilakukan selama 10 kali pengiriman	90%	Koneksi direncanakan cukup stabil dengan

No	Parameter Pengujian	Deskripsi Pengujian	Hasil Uji	Keterangan
		data berturut-turut melalui Wi-Fi local		kemungkinan gangguan kecil

Bab 4. Hasil dan Pembahasan

4.1 Hasil Rancangan Alat

Bagian ini menjelaskan hasil rancangan dari alat *Smart Storage* yang telah diselesaikan. Rancangan ini menunjukkan bagaimana setiap komponen diposisikan pada struktur alat agar sistem peminjaman dan pengembalian barang dapat berjalan dengan baik.



Gambar 4.1 Tampilan Full Unit

Pada Gambar 4.1 ditunjukkan hasil implementasi rancangan mekanik sistem *Smart Storage* yang telah direalisasikan sesuai dengan desain sebelumnya. Sistem terdiri dari beberapa jalur *conveyor* yang disusun sejajar sebagai media penyimpanan dan distribusi barang berdasarkan kategori tertentu yang ditandai dengan warna. Struktur utama alat menggunakan rangka berbahan *aluminium profile* yang ringan namun kokoh untuk menopang seluruh komponen seperti motor, sensor, dan modul kontrol. Bagian atas sistem dilengkapi dengan penutup akrilik transparan yang berfungsi sebagai pelindung sekaligus memudahkan proses pengamatan saat pengujian. Pada bagian depan terdapat *conveyor* utama sebagai jalur masuk dan keluar barang yang terintegrasi dengan aktuator seperti motor dan mekanisme *pusher* untuk mengarahkan barang ke slot tujuan secara otomatis. Secara keseluruhan, desain mekanik ini telah mampu mendukung proses peminjaman dan pengembalian barang. Pada penelitian Ariyanto [12], sistem yang

dikembangkan menggunakan RFID berbasis IoT untuk proses dokumentasi dan pencatatan peminjaman peralatan laboratorium secara digital. Sistem tersebut berfokus pada identifikasi pengguna dan penyimpanan data peminjaman. Sedangkan pada penelitian ini dilakukan pengembangan dengan menambahkan conveyor otomatis, sensor warna TCS3200, serta monitoring berbasis web sehingga sistem tidak hanya melakukan pencatatan data, tetapi juga mendukung proses borrow dan return barang secara otomatis.



Gambar 4.2 Tampilan Box Panel Kontrol

Pada Gambar 4.2 ditunjukkan *control panel enclosure* yang berfungsi sebagai pusat kendali sistem *Smart Storage*. Box ini digunakan untuk menempatkan komponen utama seperti mikrokontroler ESP32, modul komunikasi, serta rangkaian pendukung lainnya agar tersusun rapi dan terlindungi. Bagian atas menggunakan penutup akrilik transparan yang memungkinkan pengguna memantau kondisi komponen di dalamnya, serta dilengkapi dengan layar LCD yang menampilkan informasi sistem seperti status operasi dan instruksi penggunaan. Selain itu, terdapat satu *push button* yang berfungsi sebagai tombol konfigurasi untuk mengaktifkan *setup mode*, yaitu mode yang digunakan untuk melakukan pengaturan awal seperti koneksi *WiFi*, serta konfigurasi jaringan.

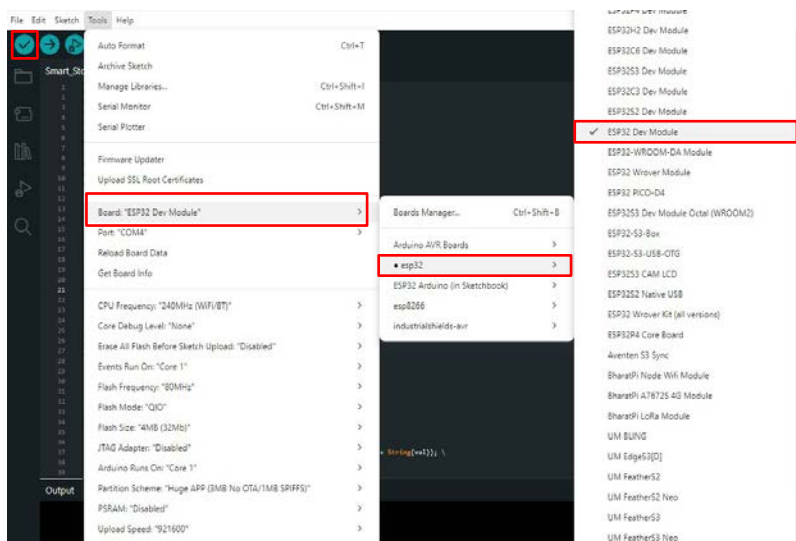
4.2 Hasil Rancangan Perangkat Lunak

Hasil rancangan perangkat lunak pada sistem *Smart Storage* meliputi pengembangan sistem berbasis web dan integrasi dengan perangkat IoT menggunakan ESP32. Sistem ini menyediakan fitur konfigurasi perangkat, manajemen pengguna, pengelolaan *inventory*, serta proses peminjaman dan pengembalian barang secara otomatis. Pada penelitian M. Zakki Nabilah dkk. [3], sistem peminjaman barang digital berbasis web dikembangkan untuk

meningkatkan efisiensi pengelolaan data dan inventaris barang. Selain itu, penelitian Muhammad Nugraha dan Jamaludin Yaskurniaam [1] juga mengembangkan sistem peminjaman barang berbasis web untuk mempermudah proses monitoring dan pengelolaan data barang. Sedangkan pada penelitian ini dilakukan pengembangan dengan mengintegrasikan website dengan sistem IoT berbasis ESP32 sehingga data yang ditampilkan tidak hanya berfungsi sebagai media pengelolaan informasi, tetapi juga terhubung secara langsung dengan perangkat keras untuk mendukung proses borrow dan return secara otomatis.

4.2.1 Konfigurasi Sistem

Konfigurasi sistem pada perangkat *Smart Storage* dilakukan melalui proses pemrograman mikrokontroler ESP32 menggunakan Arduino IDE. Tahapan ini bertujuan untuk menginisialisasi perangkat agar dapat menjalankan fungsi sistem sesuai dengan perancangan yang telah dibuat. Proses pemrograman meliputi pembukaan *source code*, pengaturan *library*, konfigurasi board ESP32, hingga proses kompilasi dan upload program ke perangkat menggunakan kabel USB.



Gambar 4.3 Pemilihan Board ESP32

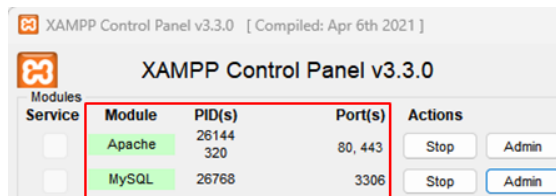
Pada Gambar 4.3 ditunjukkan proses pemilihan board ESP32 pada Arduino IDE, dimana pengguna memilih tipe *board* ESP32 Dev Module melalui menu *Tools* → *Board*. Pemilihan *board* yang sesuai sangat penting agar proses kompilasi dan

upload program dapat berjalan dengan baik. Selain itu, pada tahap ini juga dilakukan pemilihan *port* yang sesuai dengan perangkat yang terhubung ke komputer. Setelah konfigurasi board selesai, proses selanjutnya adalah melakukan *verify/compile* dan upload program ke ESP32 menggunakan kabel USB.

4.2.2 Konfigurasi Setup Mode

Setelah proses pemrograman perangkat selesai, tahap selanjutnya adalah konfigurasi jaringan untuk menghubungkan perangkat *Smart Storage* dengan sistem *web*. Proses ini dilakukan melalui setup mode pada ESP32 yang memungkinkan pengguna mengatur koneksi WiFi dan parameter sistem.

Konfigurasi jaringan bertujuan agar perangkat dapat berkomunikasi dengan server secara *real-time* dalam menjalankan fungsi peminjaman dan pengembalian barang.



Gambar 4.4 Tampilan XAMPP Control Panel pada Sistem Smart Storage

Pada Gambar 4.4 ditunjukkan tampilan *XAMPP Control Panel* yang digunakan sebagai web server dalam sistem *Smart Storage*. Pada tahap ini, layanan *Apache* dan *MySQL* telah berhasil dijalankan yang ditandai dengan status aktif. *Apache* berfungsi sebagai server untuk menjalankan aplikasi *web*, sedangkan *MySQL* digunakan sebagai *database* untuk menyimpan data sistem seperti data pengguna, *inventory*, serta transaksi peminjaman dan pengembalian barang. Tahap ini merupakan bagian penting dalam konfigurasi sistem agar perangkat dapat terhubung dan berkomunikasi dengan server secara *real-time*.



Gambar 4.5 Proses Aktivasi XAMPP dan Pembuatan Database

Pada Gambar 4.5 ditunjukkan proses pembuatan *database* baru pada *php MyAdmin*, dimana pengguna memilih menu New kemudian memasukkan nama *database* yaitu *smart_storage*, selanjutnya menekan tombol *Create* untuk membuat *database*. *Database* ini digunakan sebagai tempat penyimpanan data sistem *Smart Storage* seperti data pengguna, *inventory* barang, serta transaksi peminjaman dan pengembalian.

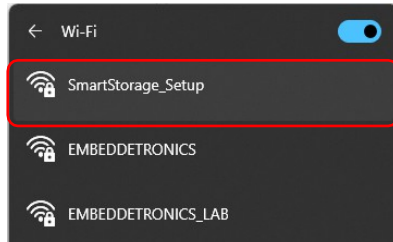
4.2.3 Prosedur Penggunaan Sistem

Bagian ini menjelaskan prosedur penggunaan sistem *Smart Storage* yang dimulai dari tahap konfigurasi perangkat. Prosedur ini bertujuan untuk menghubungkan perangkat dengan jaringan dan sistem web agar dapat berfungsi dengan baik. Pada tahap ini, pengguna melakukan pengaturan parameter sistem melalui halaman konfigurasi yang tersedia sehingga perangkat dapat terhubung dengan server dan siap digunakan untuk proses operasional.



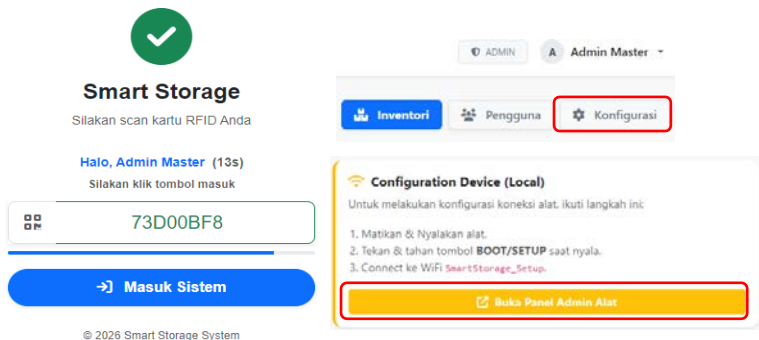
Gambar 4.6 Proses Booting dan Aktivasi Setup Mode

Sebelum melakukan konfigurasi, pengguna perlu menyiapkan jaringan WiFi atau *access point* serta memastikan perangkat dan laptop/PC terhubung pada jaringan yang sama, terutama jika menggunakan local web server (XAMPP), dan menghubungkan perangkat ke sumber daya 12V DC. Pada Gambar 4.6 ditunjukkan proses awal saat perangkat dinyalakan yang ditandai dengan tampilan *System Booting* dan *RTOS Mode ON* pada LCD, kemudian pengguna menekan dan menahan tombol konfigurasi (*push button*) untuk mengaktifkan *setup mode* sebagai langkah awal konfigurasi perangkat.



Gambar 4.7 Koneksi Access Point Smart Storage

Pada Gambar 4.7 ditunjukkan proses menghubungkan perangkat laptop atau PC ke *access point* yang dibuat oleh sistem dengan nama *Smart Storage_Setup*. Setelah perangkat masuk ke *setup mode*, pengguna dapat memilih jaringan tersebut dan menghubungkannya menggunakan *default password* “12345678”. Proses ini dilakukan agar pengguna dapat mengakses halaman konfigurasi perangkat untuk melanjutkan pengaturan sistem.



Gambar 4.8 Tampilan Login dan Konfigurasi Sistem

Pada Gambar 4.8 ditunjukkan tampilan halaman *login* sistem *Smart Storage* serta menu utama setelah pengguna berhasil masuk sebagai admin. Pengguna terlebih dahulu melakukan proses *login* dengan melakukan *scan* RFID, kemudian masuk ke dalam sistem dan memilih menu Konfigurasi. Menu ini digunakan untuk melakukan pengaturan sistem lebih lanjut sesuai kebutuhan operasional.

Device Setup

WiFi SSID

EMBEDDETRONICS_LAB

Password

Lab12345678!@

API Host

https://webhost.embeddetronics.com

API Base

/smart-storage/api

API Key

7880c7c0459fed7985bf42b6d4c98b8b

Device ID

ESP32-01

SAVE & RESTART

Nama WiFi yang akan dihubungkan dengan alat.

Password WiFi yang akan dihubungkan dengan alat.

IP/domain website (contoh seperti gambar) untuk local host silahkan

API Key didapat pada List pada website

Device ID didapat pada List pada website

Isi parameter yang sesuai, lalu Save. Alat akan otomatis restart.

Gambar 4.9 Tampilan Pengaturan Parameter Device

Pada Gambar 4.9 ditunjukkan tampilan halaman *Device Setup* yang digunakan untuk melakukan pengaturan parameter perangkat *Smart Storage*. Pada halaman ini, admin mengisi data seperti SSID dan password WiFi yang akan digunakan, alamat *server* (*API Host*), *API Base*, *API Key*, serta *Device ID*. Setelah seluruh parameter diisi sesuai dengan kebutuhan, pengguna menekan tombol *Save & Restart* sehingga perangkat akan menyimpan konfigurasi dan melakukan *restart* secara otomatis agar dapat terhubung dengan jaringan dan sistem *web*.

Manajemen API Keys

Nama Device (Cth: Gudang-1)

+ Tambah

DEVICE NAME	API KEY	AKSI
ESP32-01		

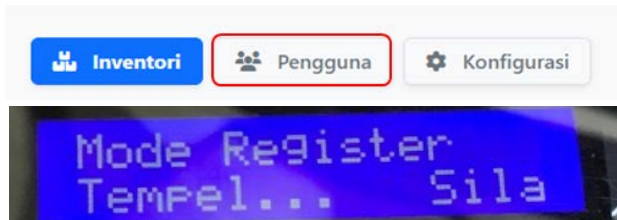
Gambar 4.10 Tampilan Manajemen API Key Perangkat

Pada Gambar 4.10 ditunjukkan tampilan halaman *Manajemen API Keys* yang digunakan untuk mengelola identitas perangkat dalam sistem *Smart Storage*. Pada halaman ini, admin dapat menambahkan perangkat baru dengan memasukkan

nama *device*, kemudian sistem akan menghasilkan *API Key* yang digunakan sebagai identitas komunikasi antara perangkat ESP32 dan *server*. Selain itu, tersedia juga fitur untuk melihat, menyalin, memperbarui, atau menghapus *API Key* sesuai kebutuhan. Penggunaan *API Key* ini bertujuan untuk memastikan keamanan dan validasi data yang dikirim dari perangkat ke sistem.

4.2.4 Proses Register User Baru

Proses registrasi user baru pada sistem *Smart Storage* dilakukan oleh admin melalui antarmuka web untuk menambahkan data pengguna ke dalam sistem. Tahap ini dimulai dengan admin masuk ke menu pengelolaan pengguna, kemudian sistem akan membaca kartu RFID yang akan didaftarkan dengan cara melakukan scan pada perangkat. Setelah kartu berhasil terdeteksi, admin mengisi data pengguna seperti nama dan peran, kemudian menyimpan data tersebut ke dalam *database*. Data RFID yang telah terdaftar akan digunakan sebagai identitas pengguna untuk mengakses sistem, khususnya dalam proses peminjaman dan pengembalian barang. Dengan adanya proses registrasi ini, sistem dapat memastikan bahwa hanya pengguna yang terdaftar yang dapat menggunakan layanan *Smart Storage*.



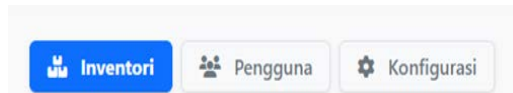
Gambar 4.11 Menu Pengguna dan Mode Registrasi RFID

Pada Gambar 4.11 ditunjukkan tampilan menu “Pengguna” pada sistem web yang digunakan oleh *admin* untuk melakukan registrasi user baru. Selain itu, ditampilkan juga kondisi perangkat saat masuk ke “Mode Register” yang ditandai dengan tampilan “Mode Register” pada LCD. Pada tahap ini, pengguna diminta untuk menempelkan kartu RFID agar dapat dibaca oleh sistem. Setelah kartu terdeteksi, data pengguna dapat disimpan dan digunakan sebagai identitas untuk mengakses sistem *Smart Storage*.

4.2.5 Proses Inventory Barang

Proses inventory barang pada sistem *Smart Storage* dilakukan oleh *admin* melalui antarmuka web untuk menambahkan dan mengelola data barang. *Admin*

memasukkan data item seperti nama dan identitas barang, kemudian barang dimasukkan ke dalam sistem melalui *conveyor*. Selanjutnya, sensor warna akan membaca karakteristik barang dan sistem akan menentukan slot penyimpanan yang sesuai. Data barang tersebut kemudian disimpan ke dalam database sehingga dapat dipantau ketersediaan dan statusnya secara *real-time*. Dengan proses ini, pengelolaan *inventory* menjadi lebih terstruktur dan terintegrasi dengan sistem.



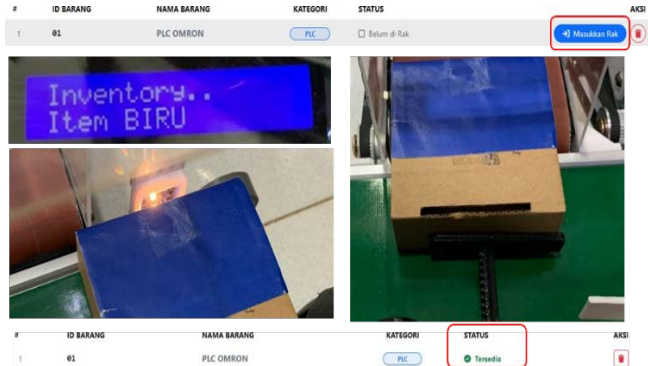
Gambar 4.12 Tampilan Menu *Inventory* pada Sistem *Smart Storage*

Pada Gambar 4.12 ditunjukkan tampilan menu *Inventory* pada antarmuka web *Smart Storage* yang digunakan oleh *admin* untuk mengelola data barang. Melalui menu ini, admin dapat melihat daftar barang yang tersedia, menambahkan item baru, serta memantau status dan ketersediaan barang dalam sistem. Menu ini berperan penting dalam proses pengelolaan *inventory* agar data barang tetap terorganisir dan dapat digunakan dalam proses peminjaman dan pengembalian.

#	ID BARANG	NAMA BARANG	KATEGORI	STATUS	AKSI
1	113314	4141414	PLC	☐ Belum di Rak	<button>Masukkan Rak</button>
2	121334	ESP32 CAM	ESP32	☒ Tersedia	

Gambar 4.13 Form Penambahan Data Barang pada Sistem

Pada Gambar 4.13 ditunjukkan tampilan *form* Tambah Barang Baru pada sistem *Smart Storage* yang digunakan oleh *admin* untuk menambahkan data item ke dalam *inventory*. Pada *form* ini, admin mengisi informasi seperti ID barang (*tag*), nama barang, serta kategori barang, kemudian menekan tombol Simpan untuk menyimpan data ke dalam *database*. Setelah data berhasil disimpan, langkah selanjutnya admin menekan tombol Masukkan Rak pada data barang yang telah ditambahkan untuk memproses penyimpanan fisik barang ke dalam sistem.

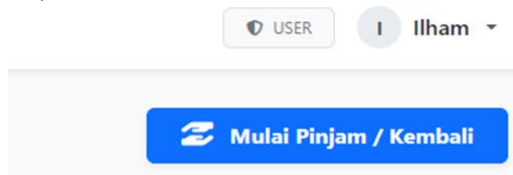


Gambar 4.14 Proses Penambahan Barang ke *Inventory System*

Pada Gambar 4.14 ditunjukkan proses penambahan barang ke dalam sistem Smart Storage melalui menu Masukkan Rak. Setelah data barang diinput, barang dimasukkan ke conveyor dan sistem akan membaca karakteristik barang, seperti warna yang ditampilkan pada LCD sebagai Item BIRU. Selanjutnya, *conveyor* akan mengarahkan barang ke dalam sistem penyimpanan, dan status barang pada *database* akan diperbarui menjadi tersedia. Proses ini menunjukkan bahwa sistem *inventory* mampu mengintegrasikan input data dengan penyimpanan barang secara otomatis.

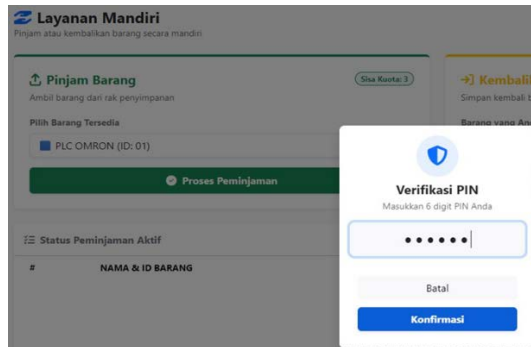
4.2.6 Proses Peminjaman Barang oleh User

Proses peminjaman barang pada sistem *Smart Storage* dilakukan oleh *user* melalui antarmuka web dengan memilih item yang tersedia pada menu peminjaman. Setelah memilih barang, *user* melakukan verifikasi menggunakan PIN yang telah terdaftar. Jika data valid dan barang tersedia, sistem akan memproses permintaan dan mengaktifkan *conveyor* untuk mengeluarkan barang secara otomatis. Seluruh aktivitas peminjaman akan dicatat ke dalam database sehingga dapat dipantau oleh *admin* secara *real-time*.



Gambar 4.15 Tampilan Menu Peminjaman Barang

Pada Gambar 4.15 ditunjukkan tampilan antarmuka *user* pada sistem *Smart Storage* yang digunakan untuk memulai proses peminjaman atau pengembalian barang. Pengguna dapat menekan tombol Mulai Pinjam / Kembali untuk melanjutkan ke proses berikutnya. Menu ini menjadi langkah awal bagi user dalam melakukan transaksi pada sistem, baik untuk mengambil maupun mengembalikan barang secara otomatis.



Gambar 4.16 Proses Peminjaman Barang dan Verifikasi PIN

Pada Gambar 4.16 ditunjukkan tampilan proses peminjaman barang pada sistem *Smart Storage*, dimana pengguna memilih barang yang tersedia kemudian menekan tombol Proses Peminjaman. Selanjutnya, sistem akan menampilkan halaman Verifikasi PIN sebagai langkah autentikasi tambahan dengan memasukkan 6 digit PIN. Jika PIN yang dimasukkan valid, maka sistem akan memproses permintaan dan mengaktifkan *conveyor* untuk mengeluarkan barang secara otomatis. Proses ini bertujuan untuk memastikan keamanan transaksi peminjaman oleh pengguna.

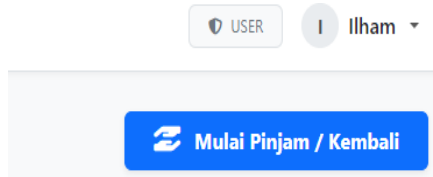


Gambar 4.17 Proses Pengeluaran Barang pada Conveyor

Pada Gambar 4.17 ditunjukkan proses pengeluaran barang pada sistem *Smart Storage* setelah pengguna berhasil melakukan peminjaman. Tampilan pada LCD menunjukkan informasi barang yang dipinjam, yaitu Item BIRU, kemudian *conveyor* akan bergerak untuk mengeluarkan barang tersebut secara otomatis. Barang akan dibawa hingga ke bagian depan sistem sehingga dapat diambil oleh pengguna. Proses ini menunjukkan bahwa sistem mampu mengintegrasikan perintah dari web dengan pergerakan mekanik secara *real-time*.

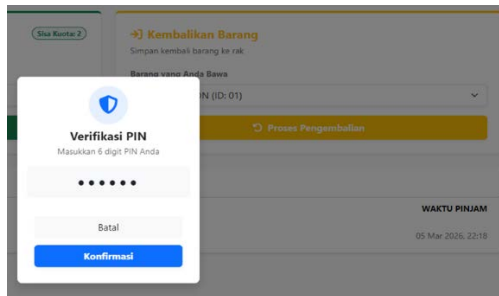
4.2.7 Proses Pengembalian Barang oleh User

Proses pengembalian barang pada sistem *Smart Storage* dilakukan oleh *user* melalui antarmuka web dengan memilih menu pengembalian, kemudian melakukan verifikasi menggunakan PIN. Setelah proses verifikasi berhasil, pengguna memasukkan barang ke *conveyor*. Selanjutnya, sistem akan membaca warna atau identitas barang menggunakan sensor, kemudian menentukan slot penyimpanan yang sesuai. *Conveyor* dan mekanisme pendorong (*pusher*) akan mengarahkan barang ke posisi yang telah ditentukan, serta sistem akan memperbarui status *inventory* secara otomatis pada *database*. Proses ini memastikan pengembalian barang dapat dilakukan secara teratur, aman, dan terintegrasi dengan sistem.



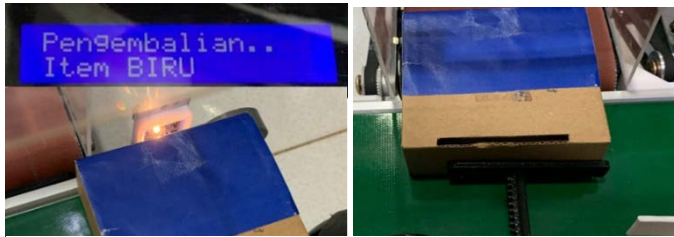
Gambar 4.18 Tampilan Menu Pengembalian Barang

Pada Gambar 4.18 ditunjukkan tampilan antarmuka *user* pada sistem *Smart Storage* yang digunakan untuk memulai proses peminjaman maupun pengembalian barang. Pengguna dapat menekan tombol Mulai Pinjam / Kembali untuk masuk ke menu layanan mandiri. Menu ini menjadi langkah awal dalam proses pengembalian barang sebelum pengguna melakukan verifikasi dan memasukkan barang ke dalam sistem.



Gambar 4.19 Proses Pengembalian Barang dan Verifikasi PIN

Pada Gambar 4.19 ditunjukkan tampilan proses pengembalian barang pada sistem *Smart Storage*, dimana pengguna memilih barang yang akan dikembalikan kemudian menekan tombol Proses Pengembalian. Selanjutnya, sistem akan menampilkan halaman Verifikasi PIN sebagai tahap autentikasi dengan memasukkan 6 digit PIN. Jika PIN yang dimasukkan valid, maka sistem akan memproses pengembalian dan melanjutkan ke tahap berikutnya, yaitu memasukkan barang ke *conveyor* untuk dilakukan penyimpanan secara otomatis.



Gambar 4.20 Proses Pengembalian Barang pada Conveyor

Pada Gambar 4.20 ditunjukkan proses pengembalian barang pada sistem *Smart Storage* setelah pengguna berhasil melakukan verifikasi. Tampilan pada LCD menunjukkan informasi barang yang dikembalikan, yaitu Item BIRU, kemudian sensor akan membaca identitas barang dan sistem menentukan slot penyimpanan yang sesuai. Selanjutnya, *conveyor* dan mekanisme pendorong (*pusher*) akan mengarahkan barang ke posisi yang telah ditentukan. Proses ini menunjukkan bahwa sistem mampu mengelola pengembalian barang secara otomatis dan terintegrasi dengan sistem *inventory*.

4.3 Hasil Pengujian Sistem

Pengujian sistem *Smart Storage* dilakukan untuk memastikan bahwa seluruh komponen perangkat keras dan perangkat lunak dapat bekerja secara optimal sesuai dengan perancangan yang telah dibuat. Pengujian ini mencakup beberapa aspek utama, yaitu konfigurasi sistem, registrasi pengguna, pengelolaan *inventory*, proses peminjaman, serta proses pengembalian barang. Selain itu, dilakukan juga pengujian terhadap komunikasi data berbasis IoT menggunakan protokol HTTP antara ESP32 dan server lokal. Hasil pengujian ini digunakan untuk mengetahui tingkat keberhasilan sistem dalam menjalankan fungsi secara otomatis, akurat, dan real-time.

Pada penelitian Dian Ariyanto [12], sistem peminjaman peralatan laboratorium berbasis RFID dan IoT digunakan untuk meningkatkan proses identifikasi pengguna dan pencatatan data secara digital. Selain itu, penelitian Muhammad Nugraha dan Jamaludin Yaskurniaam [1] mengembangkan sistem peminjaman barang berbasis web untuk meningkatkan efisiensi pengelolaan data, sedangkan J. Susanto dkk. [17] memanfaatkan ESP32 untuk komunikasi dan monitoring data secara real-time. Sedangkan pada penelitian ini dilakukan pengembangan dengan mengintegrasikan RFID, ESP32, konveyor otomatis, sensor warna TCS3200, dan website monitoring sehingga sistem tidak hanya melakukan

pencatatan dan monitoring data, tetapi juga mampu mendukung proses borrow dan return barang secara otomatis dengan komunikasi data yang berjalan secara real-time.

4.3.1 Pengujian Sistem Peminjaman Barang

Pengujian sistem peminjaman barang dilakukan untuk memastikan bahwa seluruh proses peminjaman pada sistem Smart Storage dapat berjalan dengan baik sesuai dengan perancangan yang telah dibuat. Pengujian dilakukan secara bertahap pada beberapa bagian sistem, khususnya pada komponen mekanik seperti konveyor utama dan konveyor slot yang berperan dalam proses pengeluaran barang. Setiap bagian diuji melalui beberapa percobaan untuk mengidentifikasi kendala yang terjadi, seperti performa motor, kondisi belting, serta tingkat presisi dan kesesuaian jalur barang. Hasil dari setiap pengujian dianalisis untuk mengetahui permasalahan yang muncul serta dilakukan perbaikan hingga sistem dapat bekerja secara optimal dalam proses peminjaman barang.

Tabel 4.1 Pengujian Conveyor Utama (Masalah Motor & Belting)

No	Percobaan	Kondisi Pengujian	Hasil	Status	Keterangan
1	1	Conveyor dijalankan tanpa beban	Berjalan normal	Berhasil	Motor bekerja stabil
2	2	Conveyor membawa barang	Barang bergerak lambat	Gagal	Torsi motor kurang
3	3	Conveyor berjalan lama	Belting slip	Gagal	Gesekan kurang kuat
4	4	Setelah penyesuaian motor	Pergerakan lebih cepat	Berhasil	Performa meningkat
5	5	Setelah perbaikan belting	Barang bergerak stabil	Berhasil	Sistem berjalan optimal

Berdasarkan hasil pengujian pada tabel 4.1 yang telah dilakukan, terdapat beberapa kendala pada konveyor utama, terutama pada performa motor dan kondisi belting. Pada pengujian awal, motor belum mampu memberikan torsi yang cukup sehingga pergerakan barang menjadi lambat. Selain itu, belting mengalami slip yang menyebabkan barang tidak bergerak secara optimal. Setelah dilakukan

penyesuaian pada kecepatan motor dan pengencangan belting, sistem menunjukkan peningkatan performa yang signifikan, dimana barang dapat bergerak dengan lebih stabil dan lancar.

Tabel 4.2 Pengujian Conveyor Slot (Masalah Presisi & Posisi Barang)

No	Percobaan	Kondisi Pengujian	Hasil	Status	Keterangan
1	1	Barang diarahkan ke slot	Barang tidak tepat masuk slot	Gagal	Posisi tidak presisi
2	2	Penyesuaian jalur conveyor	Barang mendekati slot	Gagal	Masih melenceng
3	3	Penambahan pembatas	Barang lebih terarah	Berhasil	Presisi meningkat
4	4	Pengujian ulang	Barang masuk slot	Berhasil	Sistem stabil
5	5	Pengujian berulang	Konsisten masuk slot	Berhasil	Presisi sudah baik

Berdasarkan hasil pengujian conveyor slot pada tabel 4.2, ditemukan bahwa presisi posisi barang saat keluar dari slot masih belum optimal pada tahap awal. Barang cenderung melenceng dari jalur yang seharusnya, sehingga tidak tepat masuk ke konveyor utama. Setelah dilakukan penyesuaian pada jalur dan penambahan pembatas, posisi barang menjadi lebih terarah dan presisi. Pengujian ulang menunjukkan bahwa barang dapat keluar dari slot dengan posisi yang sesuai dan konsisten.

Tabel 4.3 Pengujian Presisi & Adjustment Barang

No	Percobaan	Kondisi Pengujian	Hasil	Status	Keterangan
1	1	Barang diproses tanpa pengaturan	Barang miring	Gagal	Tidak stabil
2	2	Penyesuaian jalur	Barang lebih lurus	Gagal	Masih goyang

No	Percobaan	Kondisi Pengujian	Hasil	Status	Keterangan
3	3	Penambahan guide/pembatas	Barang stabil	Berhasil	Presisi meningkat
4	4	Pengujian ulang	Barang lurus dan masuk slot	Berhasil	Sistem optimal
5	5	Pengujian berulang	Konsisten	Berhasil	Presisi baik

Berdasarkan hasil pengujian presisi dan penyesuaian jalur barang pada tabel 4.3, diketahui bahwa posisi barang sangat dipengaruhi oleh kondisi jalur dan penopang mekanik. Pada pengujian awal, barang cenderung tidak stabil dan mengalami kemiringan saat bergerak. Setelah dilakukan penyesuaian pada jalur serta penambahan guide atau pembatas, posisi barang menjadi lebih stabil dan lurus. Hal ini menunjukkan bahwa proses adjustment sangat berpengaruh terhadap keberhasilan sistem dalam menjaga presisi pergerakan barang.

Tabel 4.4 Pengujian Sistem Peminjaman Barang (Hasil Akhir)

No	Nama Alat (Warna)	Pengujian					Tingkat Keberhasilan(%)	Respon LCD I2C	Keterangan
		1	2	3	4	5			
1	Arduino Uno (Hijau)	✓	✓	✓	✓	✓	100%	"Pinjam Barang ...Item Hijau"	Berhasil
2	ESP32 (Merah)	✓	✓	✓	✓	✓	100 %	"Pinjam Barang ...Item Merah"	Berhasil
3	PLC Omron (Biru)	✓	✓	✓	✓	✓	100%	"Pinjam Barang ...Item Biru"	Berhasil

Berdasarkan hasil pengujian pada tabel 4.4, sistem peminjaman barang menunjukkan bahwa seluruh proses telah berjalan dengan baik setelah dilakukan berbagai perbaikan pada tahap pengujian sebelumnya. Komponen yang terlibat, seperti konveyor utama dan konveyor slot, mampu bekerja secara terintegrasi dalam mengantarkan barang dari tempat penyimpanan hingga ke pengguna. Sistem juga berhasil menampilkan informasi pada LCD I2C sesuai dengan proses peminjaman yang dilakukan. Dengan tingkat keberhasilan mencapai 100% pada setiap percobaan, dapat disimpulkan bahwa sistem peminjaman barang telah bekerja secara optimal, stabil, dan sesuai dengan perancangan yang telah dibuat.

4.3.2 Pengujian Sistem Pengembalian Barang

Pengujian sistem pengembalian barang dilakukan untuk memastikan bahwa proses pengembalian melalui sistem Smart Storage dapat berjalan dengan baik sesuai dengan perancangan yang telah dibuat. Pengujian dilakukan secara bertahap pada beberapa bagian sistem, khususnya pada komponen mekanik dan sensor yang terlibat, seperti konveyor utama, konveyor slot, mekanisme servo pusher, serta sensor warna. Setiap komponen diuji melalui beberapa percobaan untuk mengidentifikasi kendala yang terjadi, seperti pergerakan barang, presisi penempatan, serta keakuratan pembacaan warna. Hasil dari setiap pengujian kemudian dianalisis dan dilakukan perbaikan hingga sistem mampu bekerja secara optimal dan mencapai tingkat keberhasilan yang diharapkan dalam proses pengembalian barang.

Tabel 4.5 Pengujian Sensor Warna Pada Proses Pengembalian Barang

No	Percobaan	Kondisi Pengujian	Hasil	Status	Keterangan
1	1	Deteksi warna biru	Terbaca tidak stabil	Gagal	Cahaya mempengaruhi
2	2	Deteksi warna merah	Salah deteksi	Gagal	Kalibrasi kurang
3	3	Kalibrasi sensor	Pembacaan lebih akurat	Berhasil	Sensitivitas meningkat
4	4	Pengujian ulang	Warna terbaca sesuai	Berhasil	Sistem stabil
5	5	Pengujian berulang	Konsisten	Berhasil	Tidak ada kendala

Berdasarkan hasil pengujian sensor warna pada tabel 4.5, diketahui bahwa pada tahap awal pembacaan warna barang belum berjalan secara optimal, dimana terdapat ketidakstabilan dan kesalahan dalam mendeteksi warna akibat pengaruh pencahayaan dan kalibrasi sensor yang belum tepat. Setelah dilakukan proses kalibrasi dan penyesuaian sensitivitas sensor, hasil pembacaan warna menjadi lebih akurat dan konsisten. Pengujian ulang menunjukkan bahwa sensor warna mampu mengidentifikasi jenis barang dengan baik sesuai warna yang telah

ditentukan, sehingga dapat mendukung proses pengembalian barang secara tepat dan sesuai dengan slot penyimpanan yang diinginkan.

Tabel 4.6 Pengujian Konveyor Utama (Pengembalian Barang)

No	Percobaan	Kondisi Pengujian	Hasil	Status	Keterangan
1	1	Barang masuk ke conveyor	Barang bergerak lambat	Gagal	Motor kurang optimal
2	2	Pengujian berulang	Barang tidak stabil	Gagal	Perlu penyesuaian kecepatan
3	3	Setelah pengaturan motor	Barang bergerak lebih baik	Berhasil	Performa meningkat
4	4	Pengujian ulang	Barang bergerak stabil	Berhasil	Sistem mulai optimal

Berdasarkan hasil pengujian konveyor utama pada tabel 4.6, proses pengembalian barang, diketahui bahwa pada tahap awal pergerakan barang masih belum optimal, dimana kecepatan motor dan kestabilan pergerakan belum sesuai dengan yang diharapkan. Hal ini menyebabkan barang bergerak tidak konsisten saat masuk ke jalur konveyor. Setelah dilakukan penyesuaian pada kecepatan motor dan perbaikan pada sistem penggerak, pergerakan barang menjadi lebih stabil dan lancar. Pengujian ulang menunjukkan bahwa konveyor utama mampu mengantarkan barang dengan baik menuju tahap selanjutnya dalam proses pengembalian, sehingga sistem dapat bekerja secara lebih optimal.

Tabel 4.7 Pengujian Konveyor Slot (Masalah Penempatan Barang)

No	Percobaan	Kondisi Pengujian	Hasil	Status	Keterangan
1	1	Barang diarahkan ke slot	Barang tidak masuk slot	Gagal	Posisi tidak presisi

No	Percobaan	Kondisi Pengujian	Hasil	Status	Keterangan
2	2	Penyesuaian jalur	Barang mendekati slot	Gagal	Masih melenceng
3	3	Penambahan guide	Barang masuk slot	Berhasil	Presisi meningkat
4	4	Pengujian ulang	Barang masuk dengan baik	Berhasil	Sistem stabil
5	5	Pengujian berulang	Konsisten	Berhasil	Tidak ada kendala

Berdasarkan hasil pengujian konveyor slot pada pada tabel 4.7, ditemukan bahwa pada tahap awal penempatan barang ke dalam slot masih belum presisi, dimana barang sering tidak tepat masuk ke posisi yang telah ditentukan. Hal ini disebabkan oleh ketidaksesuaian jalur dan kurangnya pengarah pada sistem mekanik. Setelah dilakukan penyesuaian pada jalur konveyor serta penambahan pembatas atau guide, posisi barang menjadi lebih terarah dan tepat masuk ke dalam slot. Pengujian ulang menunjukkan bahwa konveyor slot mampu menempatkan barang dengan lebih presisi dan konsisten, sehingga mendukung proses pengembalian barang berjalan dengan baik.

Tabel 4.8 Pengujian Servo Mekanisme Pusher

No	Percobaan	Kondisi Pengujian	Hasil	Status	Keterangan
1	1	Servo mendorong barang	Barang tidak terdorong	Gagal	Sudut servo kurang
2	2	Penyesuaian sudut servo	Barang mulai bergerak	Gagal	Dorongan belum maksimal
3	3	Penyesuaian posisi pusher	Barang terdorong	Berhasil	Posisi sudah tepat
4	4	Pengujian ulang	Barang masuk slot	Berhasil	Sistem berjalan baik

No	Percobaan	Kondisi Pengujian	Hasil	Status	Keterangan
5	5	Pengujian berulang	Konsisten	Berhasil	Tidak ada kendala

Berdasarkan hasil pengujian mekanisme servo pusher pada tabel 4.8, diketahui bahwa pada tahap awal dorongan yang dihasilkan oleh servo belum optimal, sehingga barang tidak dapat terdorong secara sempurna menuju jalur yang diinginkan. Hal ini disebabkan oleh sudut pergerakan servo dan posisi pusher yang belum sesuai. Setelah dilakukan penyesuaian pada sudut servo dan posisi mekanisme pusher, dorongan terhadap barang menjadi lebih efektif. Pengujian ulang menunjukkan bahwa servo pusher mampu mendorong barang dengan baik dan konsisten, sehingga mendukung proses penempatan barang ke dalam slot secara optimal.

Tabel 4.9 Pengujian Sistem Pengembalian Barang

No	Nama Alat (Warna)	Pengujian					Tingkat Keberhasilan(%)	Respon LCD I2C	Keterangan
		1	2	3	4	5			
1	Arduino Uno (Hijau)	✓	✓	✓	✓	✓	100%	"Pengembalian ...Item Hijau"	Berhasil
2	ESP32 (Merah)	✓	✓	✓	✓	✓	100 %	"Pengembalian ...Item Merah"	Berhasil
3	PLC Omron (Biru)	✓	✓	✓	✓	✓	100%	"Pengembalian ...Item Biru"	Berhasil

Berdasarkan serangkaian pengujian yang telah dilakukan dari berbagai permasalahan pada setiap komponen sistem, seperti konveyor utama, konveyor slot, mekanisme pendorong, serta sensor warna, dapat disimpulkan bahwa seluruh kendala yang muncul pada tahap awal telah berhasil diatasi melalui proses perbaikan dan penyesuaian sistem. Hasil pengujian akhir menunjukkan tingkat keberhasilan mencapai 100% pada setiap percobaan, yang menandakan bahwa sistem pengembalian barang telah bekerja secara optimal, stabil, dan sesuai dengan perancangan. Dengan demikian, sistem mampu menjalankan proses pengembalian barang secara otomatis dengan baik dan siap digunakan dalam kondisi operasional sebenarnya.

4.3.3 Pengujian Modul RFID

Pengujian modul RFID dilakukan untuk mengevaluasi kinerja pembacaan kartu RFID pada sistem Smart Storage. Pengujian ini bertujuan untuk mengetahui

pengaruh jarak dan posisi kartu terhadap keberhasilan pembacaan UID oleh modul RFID. Parameter yang diamati meliputi jumlah percobaan, jumlah keberhasilan, serta tingkat keberhasilan pembacaan. Pengujian dilakukan sebanyak 5 kali percobaan pada setiap variasi jarak dan posisi kartu untuk memastikan kestabilan sistem dalam kondisi yang berbeda.

Tabel 4.10 Pengujian Modul RFID

N o	Jarak Pembacaan (cm)	Posisi Kartu	Jumlah Percobaan	Jumlah Berhasil	Tingkat Keberhasilan (%)	Keterangan
1	1 cm	Datar di depan reader	5	5	100%	Pembacaan sangat stabil
2	2 cm	Datar sedikit miring	5	5	100%	Pembacaan stabil
3	3 cm	Miring / tidak sejajar	5	3	60%	Kadang perlu penyesuaian posisi
4	4 cm	Jauh dari reader	5	0	0%	Pembacaan kurang stabil

Berdasarkan hasil pengujian pada Tabel 4.10, modul RFID menunjukkan performa yang optimal pada jarak dekat, yaitu 1 cm hingga 2 cm dengan tingkat keberhasilan mencapai 100%. Pada jarak 3 cm, tingkat keberhasilan menurun menjadi 60% akibat pengaruh posisi kartu yang tidak sejajar dengan reader. Sedangkan pada jarak 4 cm, kartu tidak dapat terbaca oleh modul RFID. Hal ini menunjukkan bahwa jarak dan posisi kartu sangat mempengaruhi kinerja pembacaan RFID, serta modul RFID yang digunakan memiliki batas efektif pembacaan pada jarak kurang dari 4 cm.

4.3.4 Pengujian Modul ESP32 dalam Pemrosesan Data

Pengujian modul ESP32 dalam pemrosesan data dilakukan untuk memastikan bahwa ESP32 dapat berfungsi dengan baik sebagai pusat pengendali dalam sistem Smart Storage. Pengujian dilakukan secara bertahap dengan beberapa skenario, meliputi pembacaan data dari modul RFID, pemrosesan data oleh sistem, pengendalian output seperti konveyor, serta pengiriman data ke tampilan LCD I2C. Pada tahap awal pengujian, dilakukan beberapa percobaan untuk mengidentifikasi kendala yang terjadi dalam proses pemrosesan data dan komunikasi antar komponen. Hasil dari setiap pengujian kemudian dianalisis dan

dilakukan perbaikan hingga sistem dapat bekerja secara stabil dan optimal sesuai dengan yang diharapkan.

Tabel 4.11 Pengujian Awal Modul ESP32 dalam Pemrosesan Data

No	Fitur Pengujian	Kondisi Pengujian	Hasil	Status	Keterangan
1	Pembacaan RFID	ESP32 membaca UID dari RFID	Data kadang terlambat tampil	Gagal	Delay pada proses pembacaan
2	Pemrosesan Data	ESP32 memproses data RFID	Data diproses namun tidak stabil	Gagal	Proses belum optimal
3	Control Output	ESP32 mengaktifkan conveyor	Conveyor tidak langsung aktif	Gagal	Delay respon output
4	Tampilan LCD	ESP32 mengirim data ke LCD	Data tidak tampil	Gagal	Komunikasi belum sinkron
5	Pengujian Ulang	Setelah perbaikan sistem	Semua fungsi berjalan	Berhasil	Sistem mulai stabil

Berdasarkan hasil pengujian awal pada modul ESP32 dalam pemrosesan data pada tabel 4.11, ditemukan beberapa kendala yang mempengaruhi kinerja sistem, seperti keterlambatan dalam pembacaan data RFID, proses pengolahan data yang belum stabil, serta respon output yang tidak berjalan secara langsung. Selain itu, komunikasi antara ESP32 dengan tampilan LCD I2C juga belum sinkron, sehingga data tidak selalu ditampilkan dengan baik. Kendala-kendala tersebut menunjukkan bahwa sistem masih memerlukan penyesuaian, baik pada sisi program maupun komunikasi antar komponen. Oleh karena itu, dilakukan proses perbaikan dan optimasi sistem untuk meningkatkan stabilitas dan kecepatan respon ESP32 dalam menjalankan seluruh fungsi yang ada.

Tabel 4.12 Pengujian Akhir Modul ESP32 dalam Pemrosesan Data

No	Fitur Pengujian	Deskripsi Pengujian	Jumlah Percobaan	Jumlah Berhasil	Akurasi (%)	Hasil Serial Monitor	Keterangan
1	Pembacaan RFID	ESP32 membaca UID dari kartu RFID	5	5	100 %	UID tampil di serial monitor	Data terbaca dengan baik
2	Pemrosesan Data	ESP32 memproses data dari RFID	5	5	100 %	Status proses ditampilkan	Proses berjalan normal
3	Kontrol Output	ESP32 mengaktifkan conveyor	5	5	100 %	"Conveyor ON" tampil	Output sesuai perintah
4	Tampilan LCD	ESP32 menampilkan data ke LCD	5	5	100 %	Data sesuai dengan serial	Tampilan sinkron

Berdasarkan hasil pengujian pada Tabel 4.12, modul ESP32 dalam pemrosesan data menunjukkan kinerja yang optimal setelah dilakukan proses perbaikan pada tahap sebelumnya. Seluruh fungsi yang diuji, mulai dari pembacaan data RFID, pemrosesan data, pengendalian output, hingga pengiriman data ke tampilan LCD I2C, dapat berjalan dengan baik tanpa kendala. Tingkat keberhasilan mencapai 100% pada setiap percobaan, yang menunjukkan bahwa ESP32 mampu bekerja secara stabil, cepat, dan terintegrasi dalam sistem Smart Storage. Dengan demikian, modul ESP32 dinyatakan telah memenuhi kebutuhan sistem dan siap digunakan dalam kondisi operasional sebenarnya.

4.3.5 Pengujian Sistem Berdasarkan UML

Pengujian sistem berdasarkan diagram UML dilakukan untuk memastikan bahwa alur sistem yang telah dirancang dapat berjalan sesuai dengan implementasi pada sistem Smart Storage. Pengujian ini dilakukan dengan membandingkan proses yang terjadi pada sistem nyata dengan diagram UML yang telah dibuat, yaitu Use Case Diagram, Activity Diagram, Sequence Diagram, dan Class Diagram. Setiap diagram diuji berdasarkan kesesuaian antara fungsi, alur proses, serta komunikasi antar bagian sistem dengan kondisi aktual saat sistem dijalankan. Pengujian ini bertujuan untuk mengetahui tingkat kesesuaian antara perancangan sistem dengan implementasi yang telah direalisasikan.

Tabel 4.13 Pengujian Sistem Berdasarkan UML

N o	Jenis UML	Kasus Uji	Hasil Implementasi Sistem	Kesesuaian dengan UML	Keterangan
1	Use Case Diagram	User melakukan login, peminjaman, dan pengembalian; admin melakukan register barang, monitoring inventory, konfigurasi slot conveyor, dan monitoring log	Semua fitur dapat dijalankan sesuai hak akses	Sesuai	Fitur berjalan sesuai sistem
2	Activity Diagram (Peminjaman)	Proses peminjaman: scan RFID, validasi user, pilih barang, cek ketersediaan, conveyor aktif	Alur berjalan sesuai tahap pada sistem	Sesuai	Tidak ada perbedaan alur
3	Activity Diagram (Pengembalian)	Proses pengembalian: validasi user, input barang, sensor warna membaca, tentukan slot, conveyor bergerak	Sistem menjalankan proses sesuai diagram	Sesuai	Alur pengembalian berjalan baik
4	Sequence Diagram (Peminjaman)	Komunikasi antara RFID, ESP32, server, dan conveyor saat peminjaman	Data terbaca, diproses, dan barang keluar	Sesuai	Urutan proses sesuai
5	Sequence Diagram (Pengembalian)	Komunikasi antara RFID, color sensor, ESP32, server, conveyor	Sistem membaca warna dan menyimpan data	Sesuai	Komunikasi berjalan normal
6	Class Diagram	Relasi data User, Item, Transaction, Inventory	Struktur database sesuai implementasi	Sesuai	Data saling terhubung

Berdasarkan hasil pengujian pada Tabel 4.13, sistem Smart Storage menunjukkan kesesuaian yang baik dengan diagram UML yang telah dirancang. Seluruh proses utama, mulai dari login, peminjaman, pengembalian, hingga

pengelolaan data, dapat berjalan sesuai dengan alur yang terdapat pada diagram. Selain itu, komunikasi antar bagian sistem serta struktur data yang digunakan juga telah sesuai dengan perancangan UML. Hal ini menunjukkan bahwa implementasi sistem telah berhasil merepresentasikan desain sistem yang dibuat sebelumnya dan dapat berjalan secara terstruktur serta terintegrasi.

4.3.6 Pengujian Sistem IoT Menggunakan HTTP

Pengujian sistem IoT menggunakan protokol HTTP dilakukan untuk memastikan bahwa komunikasi data antara ESP32 dan server dapat berjalan dengan baik dalam sistem Smart Storage. Pengujian dilakukan secara bertahap dengan beberapa parameter, meliputi kecepatan respon, akurasi data, dan stabilitas koneksi. Pada tahap awal, dilakukan beberapa percobaan untuk mengidentifikasi kendala yang terjadi dalam proses pengiriman data, seperti keterlambatan respon, ketidaksesuaian data, serta gangguan koneksi jaringan. Hasil pengujian kemudian dianalisis dan dilakukan perbaikan pada sistem komunikasi hingga diperoleh performa yang optimal dan sesuai dengan kebutuhan sistem.

Tabel 4.14 Pengujian Kecepatan Respon Sistem IoT Menggunakan HTTP

No	Percobaan	Kondisi	Waktu Respon	Status	Keterangan
1	1	Kirim data RFID	3 detik	Gagal	Respon lambat
2	2	Kirim data ulang	2.5 detik	Gagal	Masih delay
3	3	Setelah perbaikan	1.5 detik	Berhasil	Respon lebih cepat
4	4	Pengujian ulang	1.2 detik	Berhasil	Stabil
5	5	Pengujian berulang	1.2 detik	Berhasil	Konsisten

Berdasarkan hasil pengujian kecepatan respon pada tabel 4.14, tahap awal sistem masih mengalami keterlambatan dalam proses pengiriman data dari ESP32 ke server, sehingga waktu respon yang dihasilkan belum sesuai dengan yang diharapkan. Hal ini disebabkan oleh proses komunikasi HTTP dan kondisi jaringan

yang belum optimal. Setelah dilakukan penyesuaian pada konfigurasi jaringan dan optimasi proses pengiriman data, waktu respon sistem menjadi lebih cepat dan stabil. Pengujian ulang menunjukkan bahwa data dapat dikirim dan diterima oleh server dengan waktu respon yang lebih singkat dan konsisten, sehingga sistem mampu mendukung komunikasi data secara real-time.

Tabel 4.15 Pengujian Akurasi Data Sistem IoT Menggunakan HTTP

No	Percobaan	Kondisi	Hasil	Status	Keterangan
1	1	Kirim UID RFID	Data tidak masuk	Gagal	Request error
2	2	Kirim ulang	Data tidak sesuai	Gagal	Tidak sinkron
3	3	Perbaikan sistem	Data sesuai	Berhasil	Sinkron
4	4	Pengujian ulang	Data sesuai	Berhasil	Stabil
5	5	Pengujian berulang	Data sesuai	Berhasil	Konsisten

Berdasarkan hasil pengujian akurasi data pada tabel 4.15, tahap awal masih ditemukan ketidaksesuaian antara data UID yang dikirim oleh ESP32 dengan data yang diterima oleh server. Hal ini disebabkan oleh proses pengiriman data yang belum stabil serta kemungkinan kesalahan pada proses parsing data di sisi server. Setelah dilakukan perbaikan pada mekanisme pengiriman dan sinkronisasi data, hasil pengujian menunjukkan bahwa data yang dikirim dapat diterima dengan akurat tanpa adanya perubahan. Pengujian ulang menunjukkan bahwa sistem mampu menjaga konsistensi data secara stabil, sehingga mendukung keandalan komunikasi data dalam sistem Smart Storage.

Tabel 4.16 Pengujian Stabilitas Koneksi Sistem IoT Menggunakan HTTP

No	Percobaan	Kondisi	Hasil	Status	Keterangan
1	1	Kirim data berulang	Terjadi beberapa kegagalan pengiriman	Gagal	Koneksi tidak stabil
2	2	Pengujian ulang	terdapat gangguan pengiriman data	Gagal	Masih ada gangguan
3	3	Perbaikan jaringan	Pengiriman data berhasil	Berhasil	Stabil
4	4	Pengujian ulang	Pengiriman data normal	Berhasil	Konsisten
5	5	Pengujian berulang	Pengiriman data berjalan dengan baik	Berhasil	Sangat stabil

Berdasarkan hasil pengujian stabilitas koneksi pada tabel 4.16, pada tahap awal masih terdapat beberapa kegagalan dalam proses pengiriman data secara berulang, yang menunjukkan bahwa koneksi jaringan belum sepenuhnya stabil. Hal ini dipengaruhi oleh kualitas jaringan *WiFi* serta proses komunikasi HTTP antara ESP32 dan server yang belum optimal. Setelah dilakukan perbaikan pada konfigurasi jaringan dan optimasi sistem komunikasi, hasil pengujian menunjukkan bahwa seluruh data dapat dikirim secara konsisten tanpa adanya kegagalan. Pengujian ulang membuktikan bahwa koneksi sistem menjadi lebih stabil dan mampu mendukung proses komunikasi data secara berkelanjutan dalam sistem *Smart Storage*.

Tabel 4.17 Pengujian Sistem IoT Menggunakan HTTP

No	Parameter Pengujian	Deskripsi Pengujian	Hasil Uji	Keterangan
1	Kecepatan Respon	Mengukur waktu dari ESP32 mengirim data hingga diterima server	Berhasil	Respon cepat
2	Akurasi Data	Membandingkan data UID dari ESP32 dengan data di server	100% sesuai	Data tidak berubah

No	Parameter Pengujian	Deskripsi Pengujian	Hasil Uji	Keterangan
3	Stabilitas Koneksi	Menguji pengiriman data secara berulang untuk melihat apakah koneksi tetap stabil	berhasil (100%)	Koneksi stabil

Dengan demikian, dapat disimpulkan bahwa sistem IoT menggunakan protokol HTTP telah mampu bekerja secara optimal setelah dilakukan proses pengujian dan perbaikan pada tahap sebelumnya. Seluruh parameter pengujian, meliputi kecepatan respon, akurasi data, dan stabilitas koneksi, menunjukkan hasil yang sangat baik dengan tingkat keberhasilan mencapai 100%. Hal ini menandakan bahwa sistem komunikasi antara ESP32 dan *server* berjalan secara cepat, akurat, dan stabil, sehingga sistem dapat diandalkan dalam mendukung proses pertukaran data secara *real-time* pada *Smart Storage*.

4.3.7 Pengujian *Functional IoT*

Functional Testing dilakukan untuk menguji fungsi utama pada sistem *Smart Storage* berbasis IoT agar dapat berjalan sesuai dengan kebutuhan dan perancangan sistem. Pengujian ini bertujuan untuk memastikan bahwa setiap proses, seperti pembacaan RFID, pengiriman data menggunakan protokol HTTP, proses peminjaman dan pengembalian barang, serta monitoring data dapat berfungsi dengan baik. Selain itu, pengujian juga dilakukan untuk mengetahui apakah sistem dapat menangani kondisi kegagalan yang mungkin terjadi selama proses komunikasi data berlangsung.

Tabel 4.18 Pengujian *Functional IoT*

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Status
1	RFID pengguna dipindai	Sistem membaca identitas pengguna	RFID berhasil membaca data pengguna	Berhasil
2	Proses peminjaman barang	Data peminjaman dikirim menggunakan HTTP dan tersimpan pada database	Data berhasil dikirim dan tersimpan	Berhasil
3	Proses pengembalian barang	Data pengembalian dikirim menggunakan HTTP dan	Data berhasil diperbarui	Berhasil

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Status
		diperbarui pada database		
4	Monitoring data pada dashboard	Data ditampilkan secara real-time	Data tampil sesuai pada dashboard	Berhasil
5	Pengiriman data HTTP saat koneksi internet terganggu	Sistem mengirim data ke database	Data gagal dikirim ke database	Gagal

Berdasarkan hasil pengujian Functional Testing, sebagian besar fungsi sistem Smart Storage berbasis IoT telah berjalan sesuai dengan perancangan. Sistem mampu melakukan pembacaan RFID, proses peminjaman dan pengembalian barang, serta menampilkan data secara real-time pada dashboard monitoring. Namun, pada saat koneksi internet mengalami gangguan, proses pengiriman data menggunakan protokol HTTP mengalami kegagalan sehingga data tidak dapat dikirim ke database. Hal ini menunjukkan bahwa kestabilan jaringan mempengaruhi proses komunikasi data pada sistem IoT.

4.3.8 Pengujian *Performance* IoT

Performance Testing dilakukan untuk mengukur kinerja sistem Smart Storage berbasis Internet of Things (IoT) selama proses komunikasi data berlangsung. Pengujian ini bertujuan untuk mengetahui kemampuan sistem dalam mengirim, menerima, dan memperbarui data menggunakan protokol HTTP pada proses peminjaman dan pengembalian barang. Selain itu, pengujian dilakukan untuk mengetahui pengaruh kestabilan jaringan terhadap performa sistem selama proses pertukaran data.

Tabel 4.19 Pengujian Performance IoT

No	Skenario Pengujian	Parameter Pengujian	Hasil Pengujian	Status
1	Pembacaan RFID	Respon sistem	Sistem mampu merespon pembacaan RFID dengan baik	Berhasil

No	Skenario Pengujian	Parameter Pengujian	Hasil Pengujian	Status
2	Pengiriman data peminjaman melalui HTTP	Pengiriman data	Data peminjaman berhasil dikirim dan diterima database	Berhasil
3	Pengiriman data pengembalian melalui HTTP	Pembaruan data	Data pengembalian berhasil diperbarui pada sistem	Berhasil
4	Monitoring data pada dashboard	Tampilan data real-time	Data berhasil ditampilkan pada dashboard	Berhasil
5	Pengiriman data HTTP saat jaringan tidak stabil	Respon sistem	Terjadi keterlambatan dan kegagalan pengiriman data	Gagal

Berdasarkan hasil pengujian Performance Testing, sistem Smart Storage berbasis IoT mampu menjalankan proses komunikasi data menggunakan protokol HTTP dengan baik pada kondisi jaringan yang stabil. Proses pembacaan RFID, pengiriman data peminjaman dan pengembalian, serta monitoring data dapat berjalan sesuai dengan fungsi yang telah dirancang. Namun, pada kondisi jaringan yang tidak stabil, proses pengiriman data mengalami keterlambatan bahkan kegagalan, sehingga dapat mempengaruhi performa sistem secara keseluruhan. Oleh karena itu, kestabilan koneksi internet menjadi salah satu faktor penting dalam mendukung kinerja sistem Smart Storage berbasis IoT.

Bab 5. Penutup

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem yang telah dilakukan, dapat disimpulkan beberapa hal sebagai berikut:

1. Sistem peminjaman barang mampu berjalan dengan baik setelah dilakukan perbaikan pada bagian mekanik, khususnya pada konveyor utama dan konveyor slot, sehingga proses pengeluaran barang menjadi lebih stabil dan sesuai dengan alur yang dirancang.
2. Sistem pengembalian barang menunjukkan kinerja yang optimal, dimana seluruh komponen seperti konveyor utama, konveyor slot, mekanisme pendorong, dan sensor warna dapat bekerja secara terintegrasi dalam mengidentifikasi serta menempatkan barang ke slot yang sesuai.
3. Modul RFID mampu membaca UID kartu dengan baik pada jarak optimal, sehingga proses identifikasi pengguna dapat dilakukan secara cepat dan akurat tanpa kendala yang signifikan.
4. Modul ESP32 sebagai pusat pengendali sistem mampu menjalankan proses pemrosesan data, pengendalian output, serta komunikasi antar perangkat secara stabil setelah dilakukan optimasi pada sistem.
5. Sistem IoT menggunakan protokol HTTP menunjukkan performa yang baik dalam proses komunikasi data antara ESP32 dan server, dengan waktu respon yang cepat, data yang akurat, serta koneksi yang stabil selama pengujian berlangsung.
6. Secara keseluruhan, sistem yang dikembangkan telah mampu menjalankan seluruh fungsi sesuai dengan perancangan, sehingga dapat bekerja secara terintegrasi dan mendukung proses peminjaman serta pengembalian barang secara otomatis.

Dengan demikian, sistem Smart Storage yang dikembangkan telah memenuhi tujuan penelitian dan dapat diimplementasikan sebagai solusi pengelolaan barang yang efisien dan terintegrasi di lingkungan kampus.

5.2 Saran

Berdasarkan hasil perancangan dan pengujian sistem Smart Storage yang telah dilakukan, terdapat beberapa saran yang dapat diberikan untuk pengembangan sistem selanjutnya:

1. Sistem dapat dikembangkan dengan menambahkan metode identifikasi lain seperti QR Code atau biometrik untuk meningkatkan fleksibilitas dan keamanan dalam proses autentikasi pengguna.
2. Sistem dapat dikembangkan dengan menggunakan server berbasis cloud sehingga akses sistem tidak terbatas pada jaringan lokal dan dapat digunakan secara lebih luas.
3. Mekanisme conveyor dan sistem penyimpanan dapat ditingkatkan dengan penambahan sensor yang lebih presisi untuk mengurangi kesalahan dalam penempatan barang.
4. Disarankan penggunaan kerangka atau rangka mekanik yang lebih kokoh dan stabil agar sistem tidak memerlukan penyesuaian ulang secara berulang serta dapat meningkatkan ketahanan alat dalam jangka panjang.
5. Antarmuka web dapat dikembangkan menjadi aplikasi mobile agar memudahkan pengguna dalam mengakses sistem secara lebih praktis.
6. Sistem monitoring dapat ditingkatkan dengan penambahan fitur notifikasi, seperti pemberitahuan saat terjadi peminjaman atau pengembalian barang secara real-time.

Daftar Pustaka

- [1] M. Nugraha dan J. Yaskurniaam, "MIND (Multimedia Artificial Intelligent Networking Database Sistem Informasi Peminjaman Barang Berbasis Web dengan Metode Waterfall," *Journal MIND Journal / ISSN*, vol. 5, no. 1, hlm. 14–23, 2020, doi: 10.26760/mindjournal.v5i1.14.
- [2] Mulyanto Dwi J dan Apriliyanti Windi, "Perancangan Sistem Informasi Peminjaman dan Pengembalian Buku Perpustakaan Berbasis Web di SMK Tujuh Lima 1 Purwokerto," hlm. 117–124, Agu 2023.
- [3] M. Z. Nabilah *dkk.*, "Rancangan Sistem Peminjaman Barang Digital Berbasis Web (Studi Kasus : di Lingkungan Jurusan Teknologi Informasi Politeknik Negeri Samarinda)," 2025.
- [4] Putra Kurniawanto Eko dan Azhari Mairiza, "Perancangan Radio Frequency Identification (RFID) Pada Transaksi Peminjaman dan Pengembalian Buku di Perpustakaan," *Edik Informatika*, vol. 8, no. 2, hlm. 125–134, Apr 2022, doi: 10.22202/ei.2022.v8i2.5566.
- [5] M. Lutfi dan R. Z. Abidin, "Rancang Bangun Sistem Peminjaman Tablet Menggunakan Metode RFID Untuk Meningkatkan Efisiensi Data Dengan Memanfaatkan Kartu Tanda Santri," 2024.
- [6] D. Ariyanto, "Rancang Bangun Sistem Peminjaman Peralatan Laboratorium Menggunakan RFID Berbasis IOT," Online, Sep 2023.
- [7] Putri Aprilia, "Analisis Sistem Dengan 4 Diagram UML Perancangan Aplikasi Penjualan dan Pembelian (Studi Kasus : Rumah Makan Uni Kapau Jaya)".
- [8] Abdi Firdaus Muhammad, Susanto Agung, Utami Ema, dan Kusnawi, "Implementasi Sinkronasi Database SQL Dengan Replikasi Pada Aplikasi Finance PT.XYZ," vol. 12, Apr 2021.
- [9] J. Susanto, M. W. Aji, Q. I. Supardi, dan Y. B. Adyapaka Apatya, "Sistem Monitoring Suhu Block Heater Sealing Pada Mesin Packaging Berbasis IOT Menggunakan Mikrokontroller ESP32," Online, 2023.
- [10] F. Rizakir dan S. A. Sukarno, "Sistem Kunci Otomatis Pada Casing Rokok Berbasis Arduino Nano Dengan LCD I2C," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 1, Jan 2025, doi: 10.23960/jitet.v13i1.5661.
- [11] Halim Ananda, fat Joni, dan Calvinus Yohanes, "Rancangan Kendali Konveyor Otomatis Berbasis Arduino dan Monitoring Melalui Aplikasi Android Untuk Meningkatkan Efisiensi Sistem Produksi," vol. 1, hlm. 81–88, Feb 2021.
- [12] D. Ariyanto, "Rancang Bangun Sistem Peminjaman Peralatan Laboratorium Menggunakan RFID Berbasis IOT," Online.
- [13] T. Informasi, S. Balikpapan, J. Z. A Maulani No, dan B. Balikpapan Selatan, "Rancang Bangun Man-Machine Interface Berbasis Web Untuk Servo

Posisi Motor DC Dengan Menggunakan Mikrokontroler AT89C51,” vol. 1, no. 1, 2017.

- [14] S. F. Athifa dan H. H. Rachmat, “Evaluasi Karakteristik Deteksi Warna RGB Sensor TCS3200 Berdasarkan Jarak dan Dimensi Objek,” *Jetri : Jurnal Ilmiah Teknik Elektro*, hlm. 105–120, Feb 2019, doi: 10.25105/jetri.v16i2.3459.
- [15] A. Akbar Rahmat Ramadhan, W. Indra Bayu, dan H. Yusfi, “Pengembangan Alat Ukur Tes Kelincahan Berbasis Teknologi Sensor Infrared,” vol. 10, hlm. 294–304, 2022, doi: 10.32682/bravos.v10i4/2777.
- [16] S. Nugroho Utomo dan R. Winarso, “Rancang Bangun Conveyor Mesin Planer Kayu Dengan Sistem Penggerak Motor Stepper,” Online, 2019.
- [17] J. Susanto, M. W. Aji, Q. I. Supardi, dan Y. B. Adyapaka Apatya, “Sistem Monitoring Suhu Block Heater Sealing Pada Mesin Packaging Berbasis IOT Menggunakan Mikrokontroler ESP32,” Online, 2023.

Lampiran

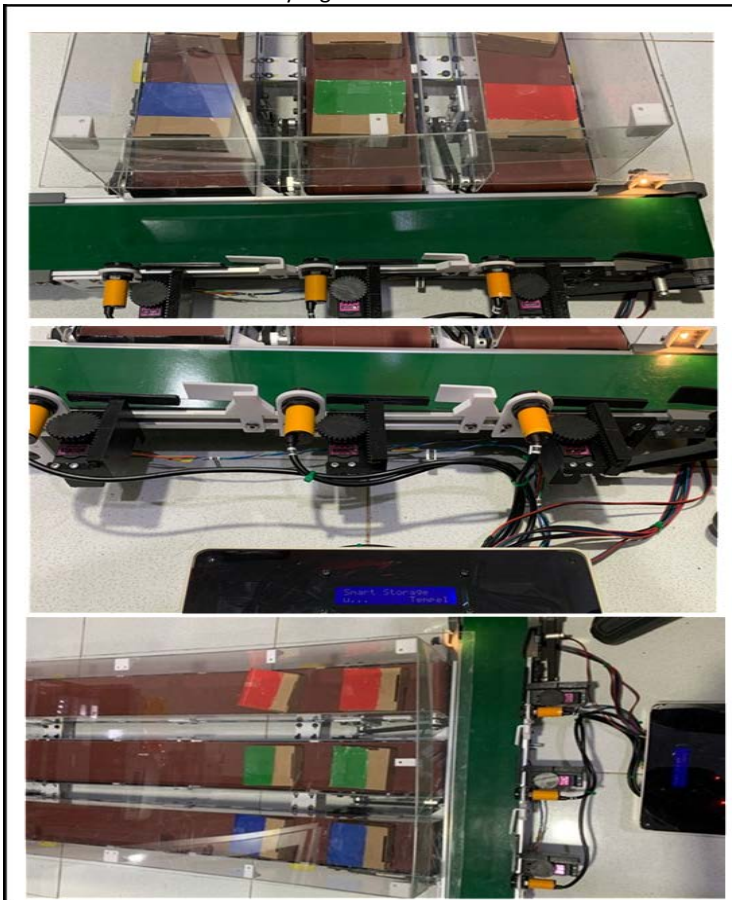
Source code sistem dapat diakses melalui link berikut:

https://drive.google.com/drive/folders/13yZYZkr2u-cl8CxhqHcbWYoNyjKOGGr4V?usp=drive_link

Video demonstrasi sistem dapat diakses melalui:

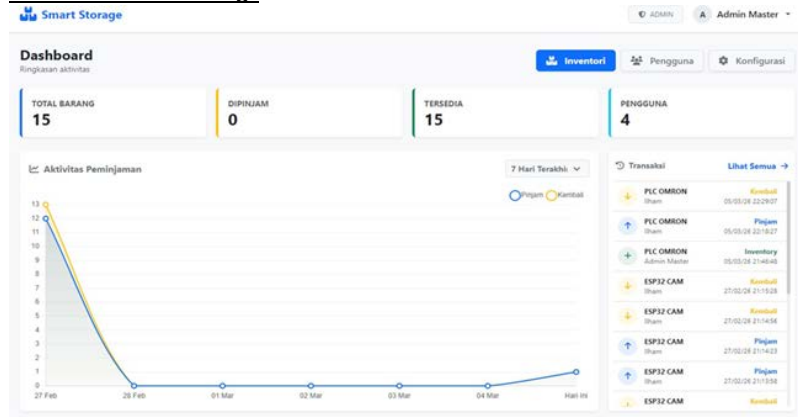
https://drive.google.com/drive/folders/1jntEw3vUAXyiAgJcf9r2bJYycsp3KVI?usp=drive_link

Dokumentasi mekanikal alat yang telah selesai :



Tampilan Sistem Web IoT :

ADMIN - Dashboard Page



ADMIN - Inventory Page

Inventori Barang
Kelola data barang dan stok rak

Tambah Barang Baru

ID Barang (Tag): Nama Barang: Contoh: Arduino Uno Kategori:

Slot Tersedia

0 MERAH 0 HIJAU 0 BIRU

#	ID BARANG	NAMA BARANG	KATEGORI	STATUS	AKSI
1	01	PLC OMRON	PLC	Tersedia	Edit
2	11234454	PLC OMRON	PLC	Tersedia	Edit
3	12134455	PLC OMRON	PLC	Tersedia	Edit
4	1113224424	PLC OMRON	PLC	Tersedia	Edit
5	122314	PLC OMRON	PLC	Tersedia	Edit
6	212333	Arduino Uno	Arduino	Tersedia	Edit
7	1134445	Arduino Uno	Arduino	Tersedia	Edit
8	1232444	Arduino Uno	Arduino	Tersedia	Edit
9	222345	Arduino Uno	Arduino	Tersedia	Edit

Kelola Kategori

Tambah Baru

Daftar Kategori

ID	Warna	Label	Aksi
1	BIRU	PLC	Edit
2	HIJAU	Arduino	Edit
3	MERAH	ESP32	Edit

ADMIN - Pengguna Page

ADMIN Admin Master

Kelola Pengguna

Daftarkan kartu RFID baru & atur akses user

Reset Scan Kembali

Registrasi User Baru

UID Kartu (Scan RFID)

+

Cat: Menunggu kartu ditempatkan pada scanner...

#	UID KARTU	NAMA PENGGUNA	ROLE	STATUS	AKSI
1	039C00F8	A Admin 2	Admin	ON	
2	730000F8	A Admin Master	Admin	ON	Aktif Saya
3	1385090F	I Ilham	User	ON	
4	13A706F8	U User 2	User	ON	

ADMIN - Configuration Page

ADMIN Admin Master

Konfigurasi Sistem

Pengaturan parameter alat dan manajemen API

Kembali

Parameter Motor & Limit

UMUM & KAPASITAS

Kapasitas Per Slot item

Batas Pinjam (User) item

MAIN CONVEYOR (UTAMA)

Kecepatan Maks (Max Speed)

Akselerasi

SLOT CONVEYOR (DISPENSER)

Kecepatan Maks (Max Speed)

Akselerasi

Jarak Steps (Kalibrasi)

SERVO ANGLE CONFIGURATION

RED Retract **RED** Extend
GREEN Retract **GREEN** Extend
BLUE Retract **BLUE** Extend

Simpan Konfigurasi

Maintenance / Troubleshooting

Gunakan fitur ini jika alat mengalami **macet (stuck)**, status terus-menerus **RUNNING**, atau ada slot yang terkunci padahal tidak ada aktivitas.

Efek: Semua transaksi status "running" akan dihapus, slot akan di-reset, dan alat akan dipaksa Restart.

Reset Sistem & Restart Alat

Manajemen API Keys

Nama Device (Cth: Gudang-1) + Tambah

DEVICE NAME	API KEY	AKSI
ESP32-01	*****	

Sensor Warna

LIVE

RED

GREEN

BLUE

Aktifkan switch untuk live data.

Threshold (Batas Bawah)

R > 150
G > 105
B > 75

Update Threshold

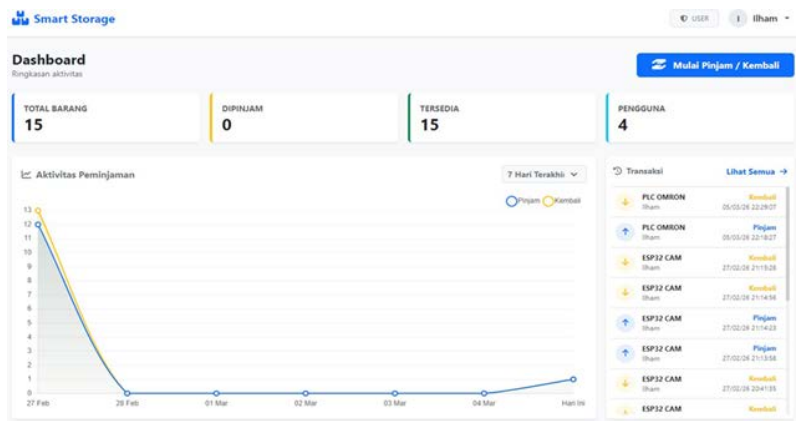
Configuration Device (Local)

Untuk melakukan konfigurasi koneksi alat, ikuti langkah-langkah ini:

- Matikan & Nyalakan alat.
- Tekan & tahan tombol **BOOT/SETUP** saat nyala.
- Connect ke WiFi **smartstorage_123456**.

Buka Panel Admin Alat

USER - Dashboard Page



USER - Pinjam/Kembali Page

Smart Storage

USER | ilham

Layanan Mandiri

Pinjam atau kembalikan barang secara mandiri

[Kembali](#)

Pinjam Barang

Ambil barang dari rak penyimpanan

[Silva Kuntia 3](#)

Pilih Barang Tersedia

-- Pilih Barang --

[Proses Peminjaman](#)

Kembalikan Barang

Simpan kembali barang ke rak

Barang yang Anda Bawa

-- Pilih Barang --

[Proses Pengembalian](#)

Status Peminjaman Aktif

#	NAMA & ID BARANG	KATEGORI	WAKTU PINJAM
Tidak ada barang yang sedang dipinjam.			

Lembar Revisi Sidang Akhir
Program Studi Teknologi Rekayasa Elektronika

Nama : Mahesha Abiyyi Gifari & Muhammad Ramadhan Pulungan
 NIM : 4242211029 & 4242211022
 Judul Proposal : Smart Storage: Rancang Bangun Sistem Peminjaman dan Pengembalian Barang Otomatis Menggunakan Teknologi RFID dan IoT di Lingkungan Kampus Dengan Metode Unified Modeling Language (UML)
 Tanggal Sidang Akhir : 19 Mei 2026

No	Nama Penguji	Saran Perbaikan	Perbaikan	TTD Penguji
1	Nanta Fakih Prebianto,S.ST., M.Sc	1. Apa yang menjadi <i>critical consideration</i> ketika membuat “Solusi” ini? 2. UML ini sebagai metode seperti apa perancangannya? 3. Flowchart gerak conveyor, hubungan antara elevasi dengan tidak ok nya dudukan. 4. Pengujian IoT belum dimunculkan. 5. Gambar pada halaman 32 kurang jelas, bagaimana membuatnya? 6. Halaman 59, Tabel 9 itu yang mana? Perlu diperjelas.	1. Sudah ditambahkan pada sub bab 1.1 latar belakang paragraf terakhir. 2. Sudah diperjelas dan diperkuat untuk UML digunakan sebagai metode pemodelan dan perancangan sistem pada Bab 3 metodologi penelitian, sub bab 3.1.3 3. Sudah ditambahkan flowchart Gerak conveyornya pada bab 3, sub bab 3.1.3.2 4. Sudah ditambahkan tabel pengujian di bab 4 hal 72. 5. Gambar hal.32 sudah dibuat jelas diagramnya, dibuat dengan aplikasi draw.io. 6. Bagian penomoran tabel pada bab 4.3 hasil pengujian sistem dari hal 59-72 sudah diperbaiki.	
2	Maria Evita Sari, S.Pd., M.Hum	1. Perbaikan isi pada Bab 1.3 Tujuan Penelitian 2. Penambahan perbandingan dengan penelitian sebelumnya pada bagian bab 4 hasil dan pembahasan	1. Bagian bab 1.3 hal 9, sudah diperbaiki 2. Sudah ditambahkan perbandingan pada bagian bab 4, sub bab 4.1 - 4.3.	

Dosen Pembimbing



Ridwan, S.ST., M. Tr. T