

Sistem Informasi Perpustakaan Kampus

Berbasis Java OOP

Laporan ini disusun untuk memenuhi tugas akhir mata kuliah Pemrograman Berorientasi Obyek yang dibina oleh : Ir. Galih Wicaksono, S.kom. M.Cs.



Disusun oleh :

Nindi Laura Salsabila	202410370110300
Nabiila Izzati Zatadini	202410370110325
Karina Putri Ani	202410370110354

PRODI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MALANG
2025

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perpustakaan merupakan salah satu fasilitas penting di lingkungan kampus. Namun, pengelolaan data buku, anggota, dan transaksi secara manual seringkali menimbulkan masalah seperti keterlambatan layanan, kesalahan pencatatan, dan kurangnya transparansi data. Untuk mengatasi hal tersebut, dibutuhkan sistem informasi perpustakaan berbasis Java dengan konsep Pemrograman Berorientasi Objek (OOP) dan antarmuka grafis (JavaFX) agar proses pengelolaan menjadi lebih efektif, efisien, dan terintegrasi.

1.2 Tujuan

1. Mengimplementasikan seluruh materi java OOP dalam proyek nyata.
2. Membuat sistem perpustakaan digital yang user-friendly dan mudah dikelola.
3. Melatih Kerjasama tim, dokumentasi, dan penggunaan tools pengembangan modern seperti GitHub.

BAB II

ANALISIS KEBUTUHAN

2.1 Fitur Utama

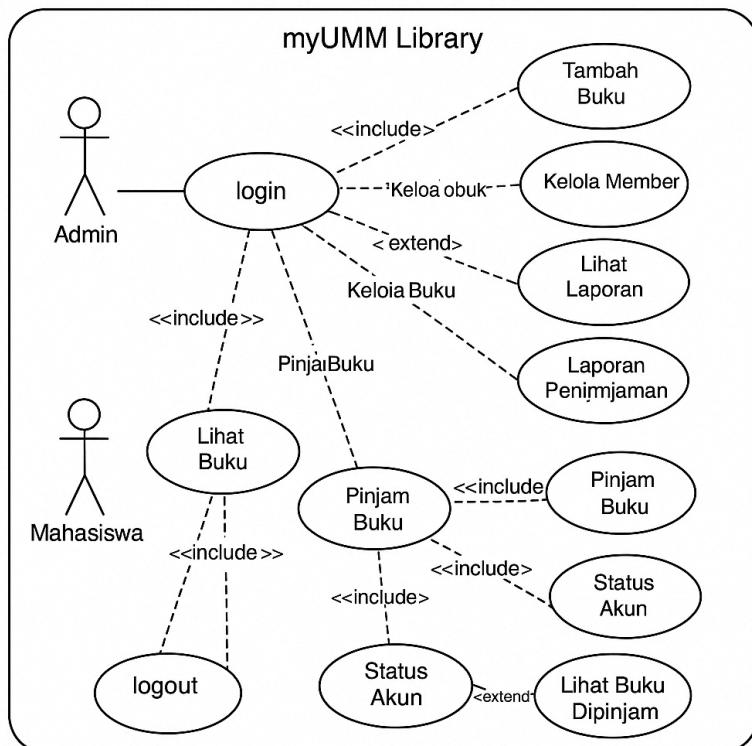
1. **Otentikasi Pengguna** : Login untuk anggota Mahasiswa dan Admin (perpustakaan).
2. **Manajemen Buku** : Operasi CRUD (Create, Read, Update, Delete) pada data buku, pencarian berdasarkan ISBN, judul, penulis, atau kategori.
3. **Registrasi Anggota**: Formulir pendaftaran anggota baru dengan validasi ID/ email ganda.
4. **Laporan & Stastik** : Laporan buku yang sedang dipinjam, statistic bulanan peminjaman, pengembalian, dan denda.

2.2 Analisis Pengguna

Admin : Mengelola data buku, anggota , transaksi, dan laporan.

Member : Melihat katalog buku, melakukan peminjaman, dan melihat riwayat peminjaman , Mencari Buku,

2.3 Use Case Diagram



BAB III

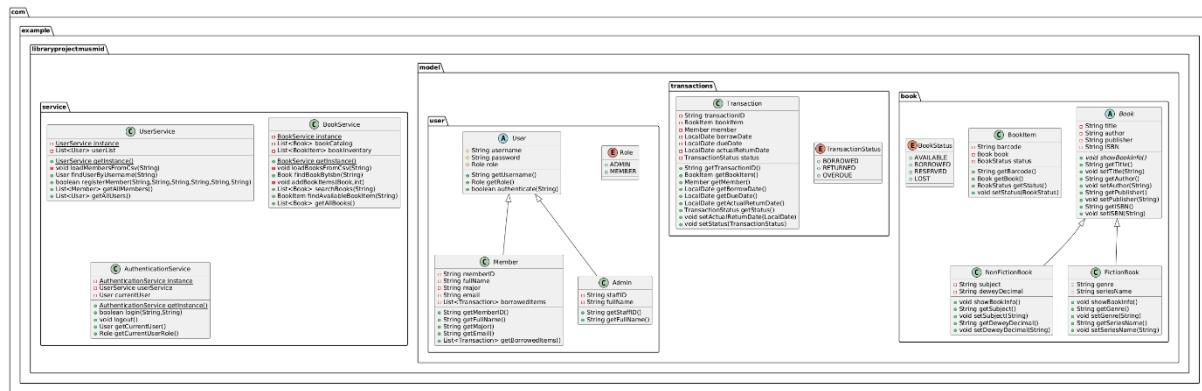
PERANCANGAN SISTEM

3.1 Arsitektur MVC

Sistem dibangun dengan Model-View-Controller (MVC) agar terstruktur dan mudah dikembangkan.

- **Model:** Kelas Book, Member, Transaction, LibraryManager yang menangani data dan logika bisnis.
- **View:** Antarmuka pengguna berbasis JavaFX (FXML), TableView, Form, dan Dialog.
- **Controller:** Menghubungkan event user ke model dan memperbarui tampilan.

3.2 Diagram Kelas

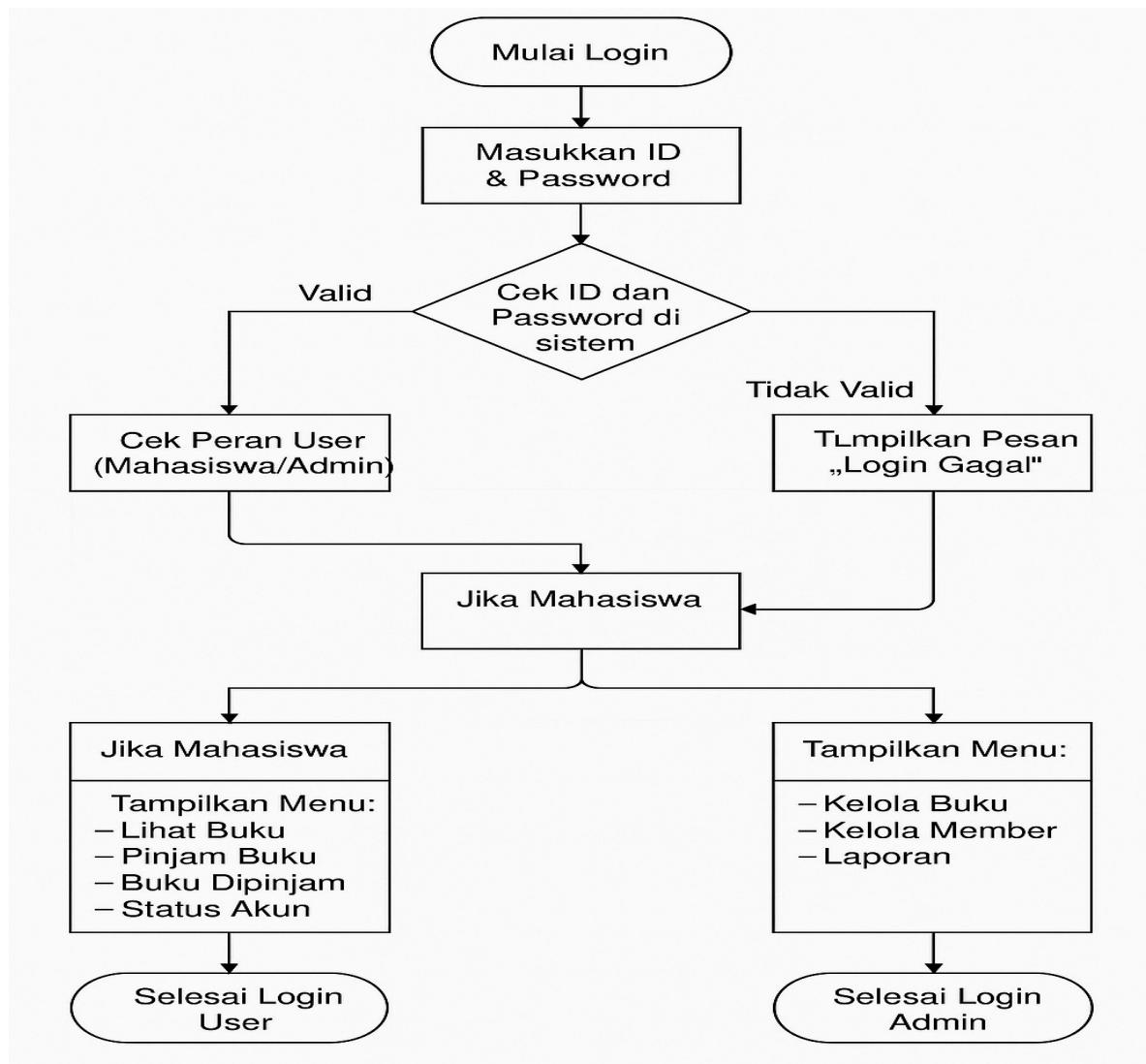


3.3 Contoh Data

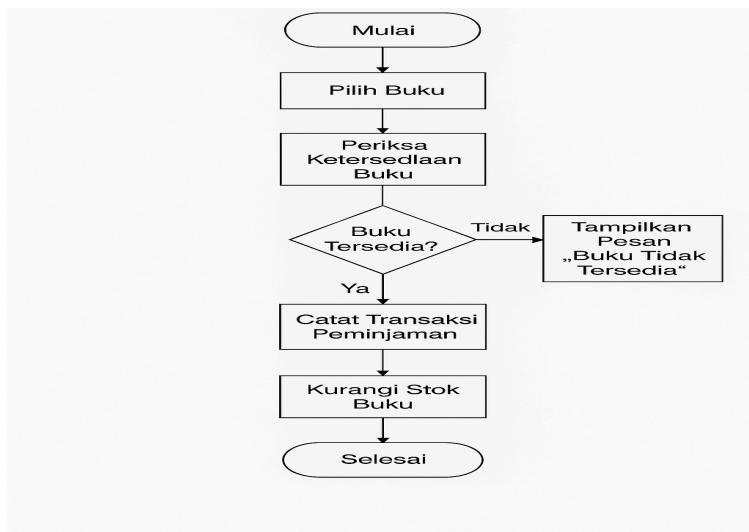
- **Buku:**
ISBN: 978-602-73156-1-0, Judul: "Pemrograman Java Dasar", Penulis: "Agus Salim", Jumlah: 5
- **Anggota:**
ID: M001, Nama: "Budi Santoso", Jurusan: "Informatika", Email: "budi@umm.ac.id"
- **Transaksi:**
ID: T1001, MemberID: M001, ISBN: 978-602-73156-1-0, TglPinjam: 2025-01-15, TglKembali: 2025-01-22, Status: Dipinjam

3.4 Flowchart Proses Utama

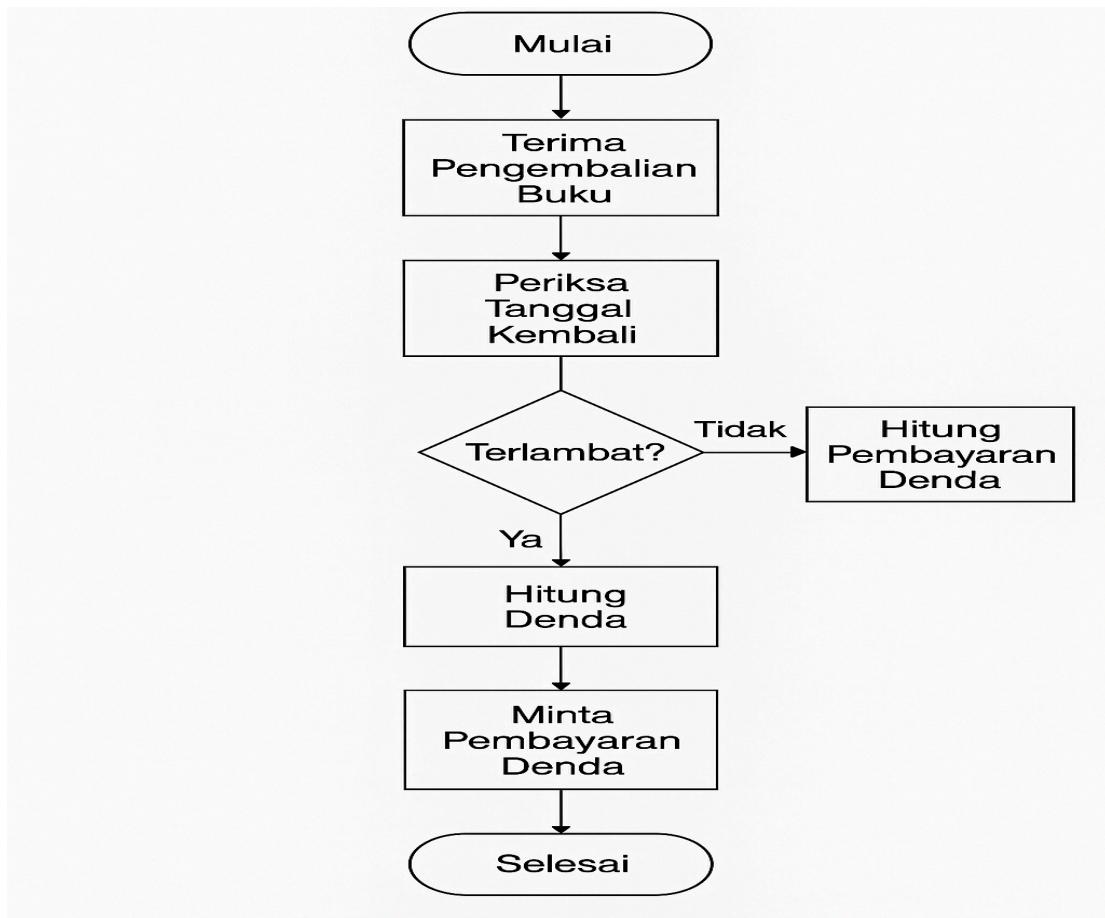
1. Proses Login



2. Peminjaman Buku



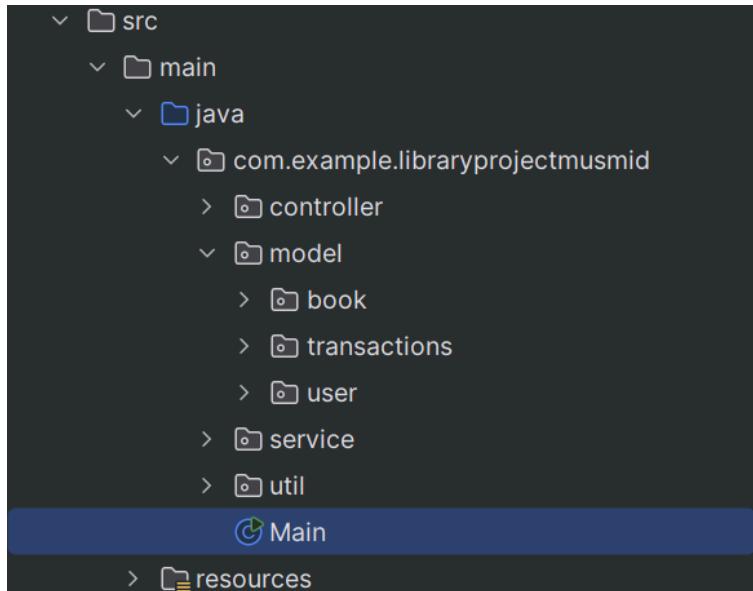
3. Proses pengembalian & perhitungan denda



BAB IV

IMPLEMENTASI

4.1 Struktur Folder



4.2 Penjelasan Kode

a. Model

- **Book:** Menyimpan data buku (ISBN, judul, penulis, kategori, jumlah).
- **Member:** Menyimpan data anggota (ID, nama, jurusan, email).
- **Transaction:** Menyimpan data transaksi (ID, memberID, ISBN, tanggal pinjam/kembali, status).
- **LibraryManager:** Mengelola operasi CRUD, transaksi, dan logika bisnis.

b. View

- Menggunakan JavaFX dengan FXML untuk antarmuka grafis.
- Komponen utama: menu utama, form input, TableView untuk data buku/anggota/transaksi, dialog konfirmasi.

c. Controller

- Menghubungkan event pada UI ke metode di model, misal saat klik tombol "Pinjam Buku", controller akan memanggil LibraryManager.borrowBook().

4.3 Contoh Potongan Kode

```

package com.example.libraryprojectmusmid.model.user;
import com.example.libraryprojectmusmid.model.user.*;

6 usages
public class Admin extends User {
    2 usages
    private String staffID;
    2 usages
    private String fullName;

    7 usages
    public Admin(String username, String password, String staffID, String fullName) {
        super(username, password);
        this.role = Role.ADMIN; // Contoh: Role enum diasumsikan ada
        this.staffID = staffID;
        this.fullName = fullName;
    }

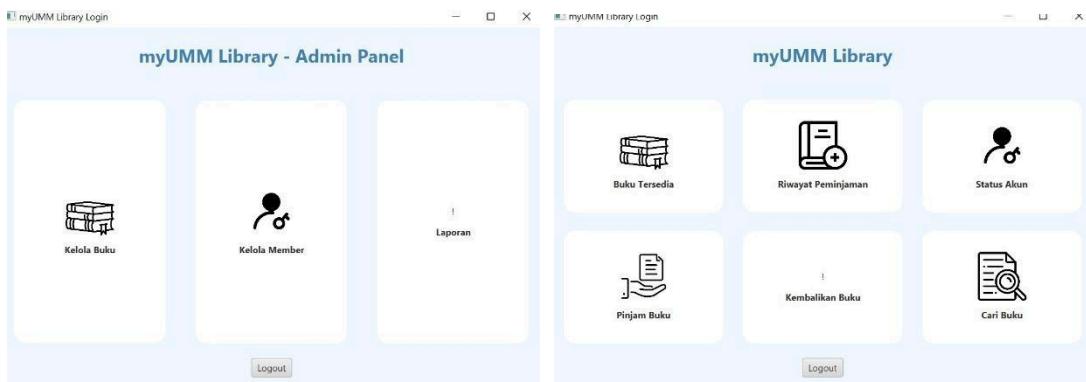
    no usages
    >     public String getStaffID() { return staffID; }

    no usages
    >     public String getFullName() { return fullName; }
}

```

4.4 Integrasi GUI

- Menu utama menampilkan pilihan



- Form input CRUD dengan validasi



- TableView menampilkan data buku.



myUMM Library Login

Daftar Buku Tersedia

Judul	Penulis	Penerbit	ISBN
Pemrograman Java Dasar	Agus Salim	Penerbit Informatika	978-602-7
Laskar Pelangi	Andrea Hirata	Bentang Pustaka	978-979-1
Bumi Manusia	Pramoedya Ananta Toer	Hasta Mitra	978-979-3
Cantik Itu Lukas	Eka Kurniawan	Gramedia	978-602-0
Filosofi Teras	Henry Manampiring	Kompas	978-602-4

< Kembali ke Dashboard

- Dialog konfirmasi sebelum menghapus data atau mengembalikan buku.

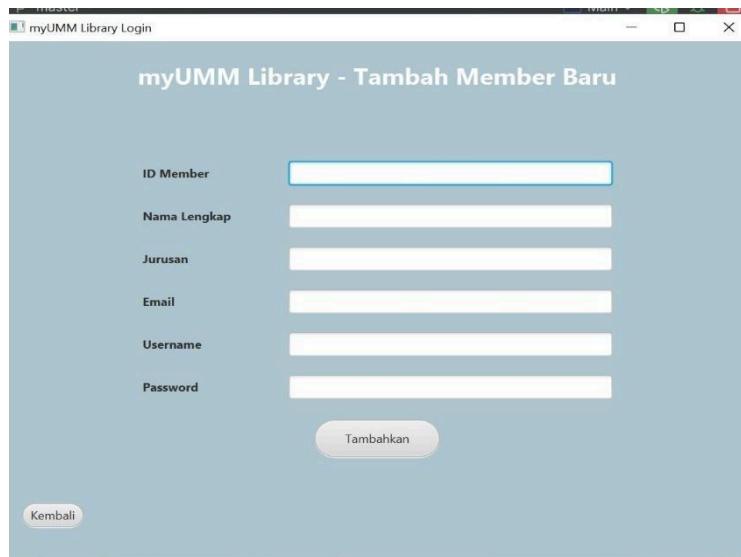


BAB V

PENGUJIAN

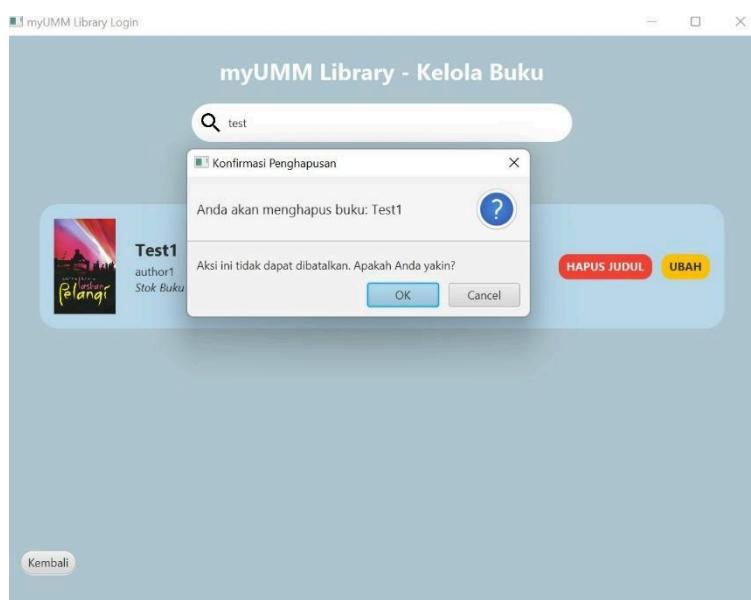
5.1 Unit Testing

- Pengujian CRUD



5.2 Hasil Pengujian

- Peminjaman tidak bisa dilakukan jika admin ingin menghapus buku maka ada validasi

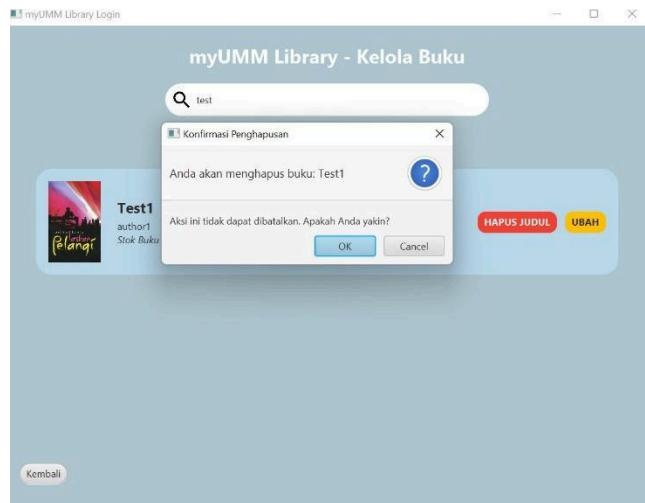


5.3 Pengujian GUI

- Semua tombol dan form berjalan lancar.



- Dialog konfirmasi muncul pada aksi penting.



BAB IV

DOKUMENTASI & PRESENTASI

- Dokumentasi Teknis: Menjelaskan arsitektur sistem, diagram kelas, flowchart, dan panduan penggunaan aplikasi.
- Video Demo: Menampilkan proses instalasi, fitur utama (CRUD, pinjam/kembali), dan alur penggunaan aplikasi.
- Slide Presentasi: Meliputi latar belakang, arsitektur, demo fitur, hasil pengujian, dan rencana pengembangan lanjutan.

BAB VII

KESIMPULAN

Proyek akhir ini berhasil mengintegrasikan seluruh konsep dasar Java OOP, pengelolaan data dengan file CSV, serta antarmuka grafis JavaFX. Sistem yang dibangun mampu meningkatkan efisiensi pengelolaan perpustakaan, meminimalisir kesalahan input, dan memberikan pengalaman digital yang lebih baik bagi pengguna. Kedepannya, sistem dapat dikembangkan dengan fitur tambahan seperti autentikasi lebih aman, ekspor laporan ke PDF, dan statistik dalam bentuk grafik.