

LAPORAN TUGAS BESAR

PEMROGRAMAN BERORIENTASI OBJEK

Laporan ini disusun untuk memenuhi tugas akhir mata kuliah Pemrograman Berorientasi Objek
Yang dibimbing oleh Bapak Ir. Galih Wasis Wicaksono, S.Kom., M.Cs.



Disusun Oleh:

Ismail Dwi Muh. Anugerah	202410370110013
Abi Danadyaksa	202410370110047
Mahligai Arsyah Nanda	202410370110056

PRODI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MALANG
2025

Arsitektur Sistem

1. Model

Bagian model merupakan representasi dari data dan logika bisnis dalam sistem informasi perpustakaan. Di dalamnya terdapat kelas-kelas entitas seperti User, Book, Borrowing, BorrowingDetail, dan Member yang mencerminkan tabel-tabel utama dalam database. Kelas-kelas ini berisi atribut dan konstruktor yang sesuai dengan data riil, serta getter dan setter untuk mengelola data objek. Selain itu, model mencakup kelas DAO (Data Access Object) seperti UserDAO, BookDAO, dan BorrowingDAO yang bertugas menjalankan operasi CRUD (Create, Read, Update, Delete) terhadap database MySQL menggunakan JDBC.

DAO bertindak sebagai penghubung antara aplikasi dan database. Semua query SQL dan logika pengambilan atau penyimpanan data diletakkan di dalam DAO, sehingga controller tidak berurusan langsung dengan SQL. Pendekatan ini memisahkan tanggung jawab, membuat kode lebih rapi dan mudah diuji. Model menjadi pusat penyimpanan data dan menjadi sumber kebenaran (source of truth) bagi seluruh sistem.

2. View

View merupakan antarmuka pengguna yang dibangun menggunakan JavaFX. Tampilan aplikasi dirancang dengan file .fxml yang memungkinkan pemisahan antara kode logika dan desain visual. Di dalam View, pengguna dapat berinteraksi melalui elemen-elemen grafis seperti tombol, label, dan form. View hanya bertanggung jawab untuk menampilkan data ke pengguna dan menangkap input, tanpa mengetahui bagaimana data tersebut diproses.

Setiap tampilan (seperti LoginView, RegisterView, Dashboard Mahasiswa, dan Admin) dipisahkan agar mudah dimodifikasi tanpa mempengaruhi bagian lain dari sistem. View akan memanggil controller untuk menangani aksi pengguna. Struktur ini memudahkan pengembangan antarmuka yang konsisten dan fleksibel di berbagai modul, baik untuk mahasiswa maupun admin.

3. Controller

Controller bertanggung jawab sebagai pengatur alur logika aplikasi. Setiap controller menghubungkan input dari View dengan proses yang ada di Model. Misalnya, LoginController memproses validasi login dengan mengambil data dari View dan memverifikasinya melalui UserDAO. Setelah proses berhasil atau gagal, controller akan mengatur tampilan mana yang akan ditampilkan berikutnya. Dengan kata lain, controller adalah jembatan aktif yang menghubungkan View dan Model secara real-time.

Controller juga memegang peran penting dalam memastikan input yang diterima dari pengguna valid sebelum diteruskan ke Model atau database. Struktur seperti AdminDashboardController atau StudentController menangani logika spesifik tergantung pada

peran pengguna. Dengan memisahkan controller dari logika tampilan dan data, sistem menjadi lebih modular dan setiap bagian bisa diuji secara terpisah.

4. Database (MySQL)

Sistem ini menggunakan MySQL sebagai tempat penyimpanan seluruh data permanen, mulai dari informasi user, buku, peminjaman, hingga pengembalian. Setiap entitas yang ada di Model berelasi langsung dengan tabel dalam database. Struktur tabel dirancang sesuai kebutuhan operasional perpustakaan digital, dengan relasi seperti peminjaman dan detail peminjaman yang terkait secara satu-ke-banyak.

Koneksi antara aplikasi Java dan MySQL dikelola melalui utilitas khusus (DatabaseUtil) dan dipanggil oleh setiap kelas DAO. Hal ini memastikan bahwa proses pengambilan dan penyimpanan data dilakukan dengan aman dan efisien. Dengan penggunaan database relasional, integritas dan konsistensi data dapat terjaga, serta memudahkan pembuatan laporan dan analisis data di tahap pengembangan lanjutan.

Kelas Diagram

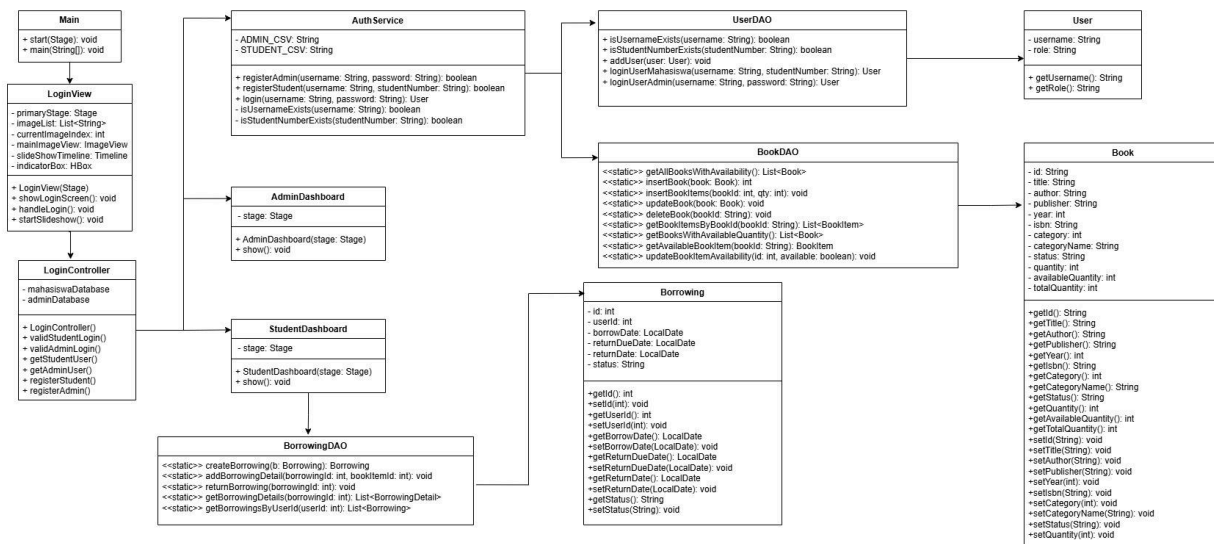
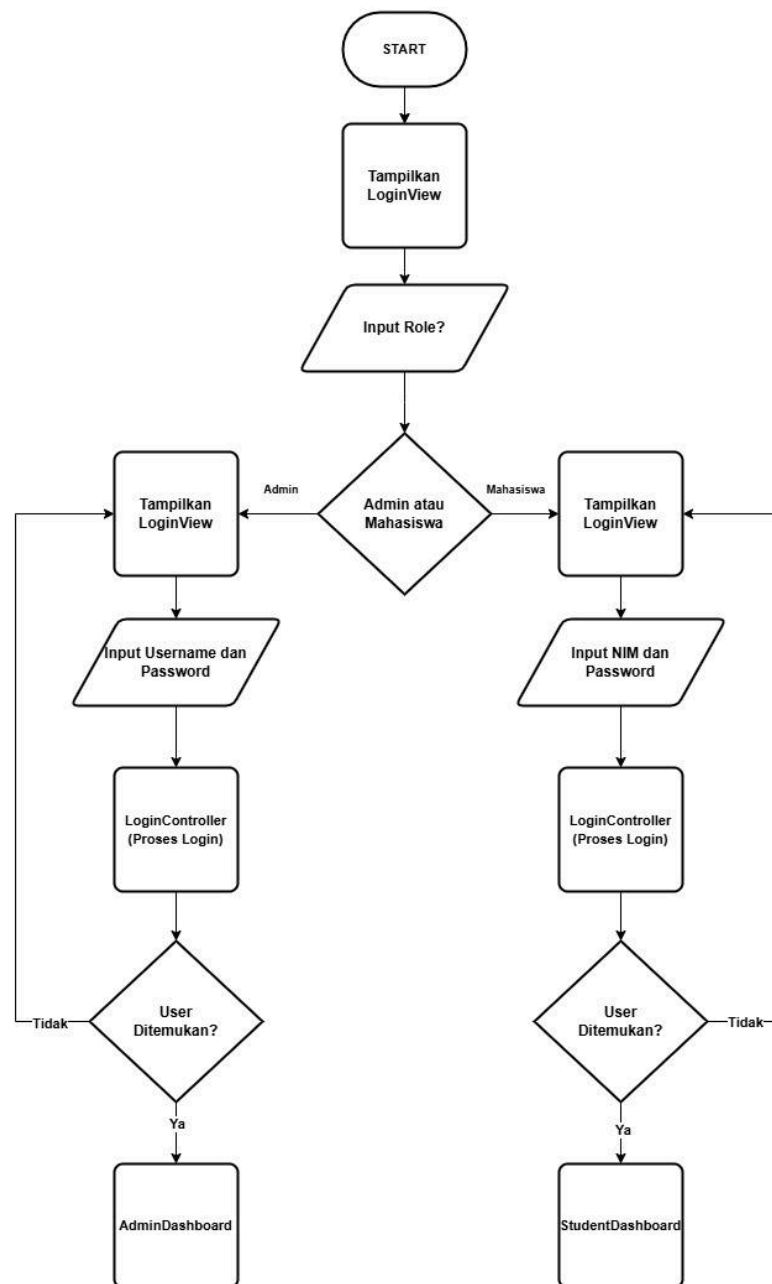


Diagram Alir



User Instruction Instalasi GUI & JavaFX di IntelliJ

1. Persiapan

Sebelum memulai, pastikan:

- JDK (Java Development Kit) **minimal versi 17** sudah terpasang.
- JavaFX SDK versi 17 sudah diunduh dari situs resmi.
- IntelliJ IDEA sudah terinstal di perangkat.
- Koneksi internet aktif (jika perlu unduh library tambahan).

2. Langkah Instalasi

Unduh JavaFX SDK

- Kunjungi situs: <https://gluonhq.com/products/javafx/>
- Pilih **JavaFX SDK versi 17** sesuai sistem operasi.
- Ekstrak file .zip ke direktori yang mudah diakses (misal: C:\javafx-sdk-17).

Buat Project Baru

- Buka IntelliJ IDEA → **File > New > Project**
- Pilih template **Java**
- Pastikan JDK yang digunakan mendukung JavaFX
- Klik **Next** dan beri nama project sesuai keinginan → **Finish**

Tambahkan Library JavaFX

- Masuk ke **File > Project Structure (Ctrl + Alt + Shift + S)**
- Pilih tab **Libraries** → klik tanda + → **Java**
- Arahkan ke folder lib di dalam folder JavaFX SDK (misalnya C:\javafx-sdk-17\lib)
- Pilih semua file .jar → klik **OK** → **Apply** → **OK**

Konfigurasi VM Options

- Buka **Run > Edit Configurations**
- Pada bagian **VM Options**, masukkan:
--module-path "C:\javafx-sdk-17\lib" --add-modules javafx.controls,javafx.fxml
- (Sesuaikan path dengan lokasi penyimpanan JavaFX SDK di perangkatmu)
- Tekan **Apply** → **OK**

User Instruction Instalasi MySQL Community Edition

1. Persiapan Awal

Pastikan perangkat telah memenuhi kebutuhan berikut:

- Sistem operasi Windows / macOS / Linux.
- Koneksi internet stabil.
- Ruang penyimpanan kosong minimal 500 MB.
- Hak akses administrator (untuk instalasi).

2. Langkah Instalasi MySQL (Windows/macOS/Linux)

1. Unduh Installer MySQL

- Buka situs resmi: <https://dev.mysql.com/downloads/mysql/>
- Pilih versi **MySQL Community Server**.
- Pilih sistem operasi yang sesuai (Windows, macOS, atau Linux).
- Klik **Download** dan simpan installer ke perangkat.

2. Jalankan Installer

- Buka file installer yang telah diunduh.
- Pilih tipe instalasi:
 - **Developer Default** (disarankan, mencakup MySQL Server, Workbench, Shell, dll).
- Klik **Next** untuk melanjutkan.

3. Konfigurasi MySQL Server

- Pilih port (default: 3306).
- Pilih mode autentikasi: **Use Strong Password Encryption**.
- Buat dan simpan **root password** yang mudah diingat dan aman.

4. Selesaikan Instalasi

- Klik **Execute** untuk memulai proses instalasi dan konfigurasi.
- Tunggu hingga semua status bertanda **"Complete"**.
- Klik **Finish**.

5. Verifikasi Instalasi

- Buka **MySQL Workbench** (jika ikut diinstal).

- Lakukan koneksi ke `localhost` menggunakan user `root` dan password yang telah dibuat.
- Jika berhasil login, instalasi selesai.