**DOKUMENTASI PROYEK**

**Nama Proyek:** CampusLibrarySystem

**Nama:** Abdul Halim (016)

Muhammad Idris Dwi (018)

Muhammad Arif Rahman (033)

**Mata Kuliah:** Pemrograman Berorientasi Objek (OOP)  
**Kelas:** 2B  
**Dosen Pengampu:** Ir. Galih Wasis Wicaksono, S.Kom., M.Cs.  
**Tanggal Selesai:** 19 Juni 2025

**1. Latar Belakang**

Perpustakaan Universitas Muhammadiyah Malang saat ini masih mengalami kendala dalam sistem peminjaman dan pengembalian buku yang sebagian besar dilakukan secara manual. Hal ini menyebabkan proses yang lambat dan rawan kesalahan pencatatan. Untuk mengatasi masalah tersebut, tim kami mengembangkan sistem informasi perpustakaan kampus berbasis Java dengan antarmuka JavaFX dan pengolahan data berbasis file CSV. Sistem ini diharapkan mampu mempermudah proses pelayanan peminjaman buku serta meminimalkan kesalahan data yang terjadi karena pencatatan manual.

**2. Tujuan Proyek**

Tujuan dari proyek ini adalah membuat aplikasi **CampusLibrarySystem** yang mampu:

* Mengelola data buku dan anggota perpustakaan (CRUD)
* Memproses peminjaman dan pengembalian buku
* Menyediakan autentikasi login untuk admin dan anggota
* Menampilkan antarmuka pengguna yang ramah menggunakan JavaFX
* Menyimpan dan membaca data menggunakan file CSV sebagai alternatif database sederhana

**3. Arsitektur Sistem**

Sistem dibangun dengan menerapkan arsitektur **Model-View-Controller (MVC)**, yang terdiri dari:

* **Model:** Menyimpan struktur data utama seperti Book, Member, Transaction
* **View:** Tampilan antarmuka pengguna dengan JavaFX
* **Controller (Service):** Logika aplikasi yang mengatur alur data antara model dan tampilan
* **Repository:** Mengelola proses penyimpanan dan pembacaan data CSV
* **Auth:** Autentikasi pengguna yang memisahkan akses antara admin dan anggota

Struktur proyek terdiri dari:

* model/ : berisi struktur data dan entitas utama
* repository/ : pengolahan data melalui file I/O
* service/ : berisi logika bisnis aplikasi
* gui/ dan view/ : mengelola tampilan antarmuka pengguna
* auth/ : berfungsi untuk proses login dan otorisasi
* resources/ : menyimpan file gambar seperti logo dan background

**4. Dokumentasi Harian (Day-by-Day Progress)**

**Day 1–3: Setup Awal & Perencanaan**

Hari-hari pertama kami manfaatkan untuk menyusun konsep dan fondasi proyek. Kami berdiskusi untuk menyepakati alur kerja, membagi peran di dalam tim, serta memilih arsitektur proyek. Kami memutuskan menggunakan struktur MVC agar lebih terorganisir dan scalable. UI kami rancang terlebih dahulu di Canva sebagai prototipe awal yang akan diterapkan ke JavaFX. Tidak lupa, kami menginisialisasi repository di GitHub untuk memudahkan kolaborasi. Hari-hari awal ini terasa cukup ringan namun sangat krusial untuk kesuksesan tahap-tahap selanjutnya.

**Day 4–6: Pengembangan Backend (CRUD)**

Memasuki hari keempat, kami mulai terjun ke bagian coding, khususnya pada backend. Kami membangun class-class utama seperti Book, Member, dan Transaction. Kami juga membuat sistem CRUD menggunakan file CSV sebagai media penyimpanan. Tantangan utama pada tahap ini adalah memastikan setiap perubahan data dapat disimpan dan dimuat ulang tanpa error, serta menghindari duplikasi data. Kami juga belajar banyak tentang penanganan file dan exception seperti FileNotFoundException. Hari-hari ini cukup melelahkan karena penuh debugging, tetapi menyenangkan saat fungsi CRUD pertama kami berhasil berjalan dengan baik.

**Day 7–10: Frontend & Fitur Login + Peminjaman**

Di pertengahan proyek, fokus kami beralih ke bagian frontend. Dengan bantuan referensi dari desain Canva dan arahan dari ChatGPT, kami mengimplementasikan tampilan antarmuka menggunakan JavaFX. Mulai dari halaman login, dashboard admin, katalog buku, hingga form peminjaman, semua kami rancang dengan perhatian terhadap kemudahan penggunaan. Selain itu, kami juga menyambungkan data dengan TableView agar pengguna dapat melihat daftar buku secara langsung dari antarmuka. Fitur peminjaman dan pengembalian pun kami kerjakan, termasuk logika perhitungan denda keterlambatan. Meski sempat mengalami kesulitan dalam menangani event handling dan validasi input, hari-hari ini menjadi titik di mana sistem kami mulai terasa nyata dan fungsional.

**Day 11–13: Testing & Dokumentasi**

Tiga hari terakhir kami gunakan untuk melakukan penyempurnaan sistem. Kami melakukan banyak testing, memperbaiki bug yang tersisa, serta memastikan semua fitur bekerja sesuai alur. Beberapa error seperti data tidak tersimpan, alert yang tidak muncul, hingga logika denda yang salah sempat muncul, namun dapat diatasi dengan debugging bersama. Selain itu, kami mulai menyusun dokumentasi teknis dan membuat slide presentasi untuk persiapan demo. Hari-hari terakhir ini cukup padat dan sedikit menegangkan, namun kami merasa puas melihat hasil kerja keras selama hampir dua minggu mulai terlihat lengkap dan siap dipresentasikan.

**5. Fitur Utama**

1. **Login**
   * Pengguna: Admin & Anggota
   * Validasi data login melalui file CSV yang berisi daftar akun terdaftar
2. **Manajemen Buku**
   * Tambah, edit, hapus, dan cari buku berdasarkan ISBN atau judul buku
   * Data disimpan dan dimuat dari file CSV
3. **Manajemen Anggota**
   * Registrasi anggota baru dengan validasi ID dan email agar tidak duplikat
4. **Peminjaman & Pengembalian**
   * Mencatat tanggal pinjam, estimasi pengembalian, dan status peminjaman
   * Menghitung denda otomatis berdasarkan selisih hari keterlambatan
5. **Notifikasi & Validasi**
   * Menggunakan JavaFX Alert untuk memberikan feedback berupa konfirmasi atau peringatan kepada pengguna

**6. Kesimpulan & Evaluasi**

CampusLibrarySystem berhasil mengintegrasikan berbagai konsep dasar dan lanjutan dari Pemrograman Berorientasi Objek menggunakan bahasa Java. Seluruh fitur dirancang untuk menangani alur kerja perpustakaan sederhana, mulai dari manajemen buku hingga pengembalian. Dengan pembagian tugas yang jelas serta dukungan dari sumber eksternal seperti ChatGPT dan Canva, proses pengembangan dapat berjalan efektif dan efisien. Proyek ini menjadi pengalaman berharga dalam bekerja secara tim, membangun aplikasi nyata, dan menerapkan prinsip OOP secara praktikal.

**7. Saran Pengembangan**

Beberapa fitur tambahan yang dapat dikembangkan di masa depan:

* Ekspor laporan transaksi ke PDF
* Statistik peminjaman bulanan dalam bentuk grafik menggunakan JavaFX Charts
* Sistem notifikasi email atau pop-up pengingat pengembalian buku
* Penyimpanan data menggunakan database relasional seperti MySQL untuk skalabilitas lebih tinggi

**Disusun oleh:**  
Tim CampusLibrarySystem  
Mahasiswa Informatika  
Universitas Muhammadiyah Malang

Foto Hasil Dari Dokumentasi

**Day 1–3: Setup Awal & Perencanaan**





**Day 4–6: Pengembangan Backend (CRUD)**





**Day 7–10: Frontend & Fitur Login + Peminjaman**



**Day 11–13: Testing & Dokumentasi**

