

LEMBAR KONTRIBUSI PROYEK UAS

PEMROGRAMAN BERBASIS OBJEK

1. Identitas Mahasiswa

Nama Lengkap	NIM	Tugas
Widari Dwi Hayati	2311102060	CSS, mysql
Anisa Yasaroh	2311102063	Class BookDAO, Searchable, BookUtils
Fitri Kusumaningtyas	2311102068	Class Database, Library, Book
Trie Nabilla Farhah	2311102071	Html, Jsp
Anisah Syifa Mustika Riyanto	2311102080	Web.xml, Class Item dan BookServlet

2. Judul Proyek

LITERATA PERPUSTAKAAN – Sistem Manajemen Perpustakaan Digital Berbasis Java

3. Deskripsi Singkat Proyek

Literata Perpustakaan adalah sistem manajemen perpustakaan digital berbasis Java yang dirancang untuk membantu pengguna dalam mengelola data buku. Aplikasi ini dibangun menggunakan bahasa pemrograman Java dengan arsitektur MVC (Model-View-Controller), dan terintegrasi dengan database MySQL untuk penyimpanan data buku dan transaksi peminjaman. Literata juga dilengkapi dengan fitur CRUD (Create, Read, Update, Delete) untuk memudahkan pengelolaan koleksi buku dan mendukung otomatisasi administrasi perpustakaan.

4. Fitur-fitur utama

Aplikasi ini bersifat sederhana namun fungsional, ditujukan untuk mendukung pengelolaan data buku dalam perpustakaan digital. Fitur yang tersedia antara lain.

- **Form Penambahan Buku**

Pengguna dapat menambahkan data buku baru ke sistem melalui halaman form.

Input Data Buku. Data yang dimasukkan meliputi:

- ID Buku
- Judul Buku
- Nama Penulis
- Tahun Terbit

- **Simpan ke Database**

Data buku yang sudah diisi akan disimpan langsung ke database MySQL melalui koneksi JDBC.

- **Navigasi Sederhana**

Pengguna dapat berpindah antar halaman JSP seperti BookList.jsp (daftar buku) dan editBook.jsp (edit data buku).

5. Fitur yang Disediakan

No	Fitur yang Dikerjakan	Deskripsi Tugas	Kode yang Digunakan
1	Dashboard	Menyajikan ringkasan visual atau akses cepat ke fitur utama aplikasi, seperti daftar buku dan input buku baru, untuk memudahkan navigasi pengguna.	dashboard.html style.css
2	Daftar Buku	Menampilkan seluruh data buku yang tersimpan dalam database dalam bentuk tabel, lengkap dengan informasi detail dan opsi untuk mengedit atau menghapus setiap buku.	BookList.jsp Book.java BookServlet.java BookDAO.java DatabaseConnection.java Library.java style.css BookUtils.java

3	Input Buku Baru	Menyediakan formulir input untuk menambahkan data buku baru ke dalam sistem seperti Nomor Buku, Judul Buku, Penulis, dan Tahun Terbit. Kemudian data disimpan dalam database untuk ditampilkan di daftar buku.	newBook.html Book.java BookServlet.java BookDAO.java DatabaseConnection.java Library.java
4	Edit Buku	Menampilkan data buku yang dipilih dan memberikan akses untuk mengedit Nomor Buku, Judul Buku, Penulis, dan Tahun Terbit.	editBook.jsp Book.java BookServlet.java BookDAO.java DatabaseConnection.java Library.java Searchable.java BookUtils.java Item.java style.css
5	Hapus	Menghapus data buku tertentu dari database secara permanen berdasarkan ID buku yang dipilih pengguna.	BookList.jsp Book.java BookServlet.java BookDAO.java DatabaseConnection.java Library.java Searchable.java BookUtils.java Item.java style.css

6. Kendala yang Dihadapi & Solusi

Kendala:

- Kesalahan koneksi ke database MySQL karena pengaturan port atau driver JDBC belum terpasang dengan benar.
- Error saat melakukan operasi CRUD, seperti data tidak tersimpan atau tidak tampil karena query atau pemetaan data yang kurang tepat.
- Struktur proyek yang belum konsisten, menyebabkan kesulitan saat navigasi antar halaman JSP atau saat memanggil Servlet.
- Styling antarmuka pengguna (UI) kurang menarik dan membutuhkan penyesuaian CSS agar tampilan lebih rapi.

Solusi:

- Melakukan pengecekan dan konfigurasi ulang DatabaseConnection dengan memastikan driver `com.mysql.cj.jdbc.Driver` ter-load dan port default MySQL (3306) tersedia.
- Melakukan debugging menggunakan `System.out.println()` dan try-catch untuk melacak error query, serta memperbaiki syntax SQL dan pemetaan atribut Java.
- Menata ulang struktur file berdasarkan prinsip MVC (Model-View-Controller), agar kode lebih modular dan mudah dipahami.
- Menyesuaikan file CSS dan struktur HTML agar antarmuka lebih responsif dan user-friendly.

7. Refleksi dan Pembelajaran

Melalui proyek ini, saya belajar secara langsung penerapan konsep Pemrograman Berbasis Objek (OOP) dalam pengembangan aplikasi berbasis Java. Saya memahami pentingnya pemisahan logika aplikasi ke dalam model, DAO, dan servlet agar lebih terstruktur. Saya juga mempelajari cara menghubungkan aplikasi Java ke MySQL menggunakan JDBC serta menerapkan operasi CRUD dengan aman menggunakan PreparedStatement. Selain itu, saya menyadari pentingnya desain antarmuka yang sederhana namun fungsional, serta pentingnya kolaborasi tim dan dokumentasi dalam pengembangan perangkat lunak.