LAPORAN PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK

MEMBUAT FUNGSI CRUD (Create, Read, Update, Delete) USER

DENGAN DATABASE MYSQL

OLEH :

Abdullah Al Ramadhani

2411533016

DOSEN PENGAMPU:

NURFIAH, S.ST, M.Kom,

FAKULTAS TEKNOLOGI INFORMASI

DEPARTEMEN INFORMATIKA

UNIVERSITAS ANDALAS

PADANG

2025

A. Pendahuluan

Pemrograman Berorientasi Objek (PBO) adalah sebuah paradigma yang berpusat pada konsep objek, di mana setiap objek mempunyai atribut (data) dan metode (perilaku). Paradigma ini memfasilitasi developer dalam menciptakan aplikasi yang terorganisir, modular, dan gampang dirawat. Pada praktikum ini, PBO diaplikasikan untuk membuat aplikasi laundry yang menjalankan empat operasi data fundamental, yaitu CRUD (Create, Read, Update, Delete).

CRUD merupakan serangkaian operasi inti untuk manajemen data yang memungkinkan aplikasi untuk membuat data baru, membaca data yang ada, memperbarui, serta menghapusnya dari database. Proses ini didukung oleh beberapa komponen utama. XAMPP digunakan sebagai server lokal yang sudah mencakup Apache dan MySQL. Untuk menghubungkan aplikasi Java ke database MySQL, digunakan MySQL Connector/J sebagai driver-nya.

Guna menjaga agar kode tetap rapi dan terstruktur, digunakan pola desain DAO (Data Access Object) yang berfungsi memisahkan logika bisnis dari logika akses data. Pemisahan ini membuat kode lebih mudah digunakan kembali (reusable) dan dikembangkan. Selain itu, Interface dalam Java dimanfaatkan untuk menetapkan "kontrak" atau standar metode yang wajib dimiliki oleh sebuah kelas, sehingga struktur program menjadi lebih konsisten.

Dengan mengintegrasikan PBO dan fungsionalitas CRUD, kita bisa memahami alur pengolahan data dalam sebuah aplikasi sekaligus menerapkan prinsip-prinsip pengembangan perangkat lunak seperti modularitas, enkapsulasi, dan reusabilitas.

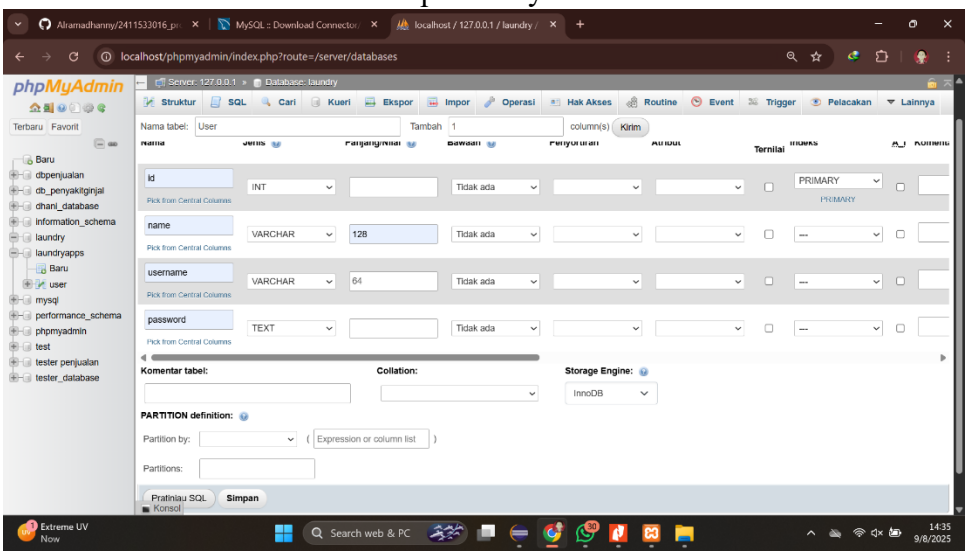
B. Tujuan

- Mahasiswa dapat mendesain dan membangun tabel user dalam sistem database MySQL.
- Mahasiswa bisa menghubungkan aplikasi Java dengan database MySQL secara efektif.
- Mahasiswa mampu merancang antarmuka grafis (GUI) untuk operasi Create, Read, Update, dan Delete pada user.
- Mahasiswa dapat mengembangkan serta menerapkan interface dalam pemrograman.
- Mahasiswa mampu membuat serta menjalankan fungsi DAO (Data Access Object) untuk manajemen data.
- Mahasiswa dapat mengaplikasikan prinsip Pemrograman Berorientasi Objek dalam pembuatan fungsi CRUD.

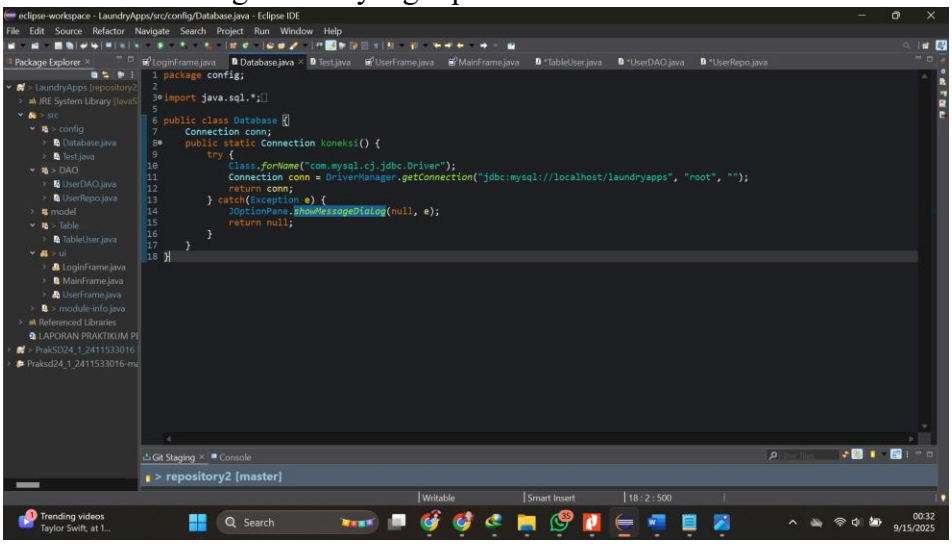
C. Langkah-Langkah

1) Pasang MySQL Connector dan tambahkan library connector tersebut ke dalam package proyek.

2) Buat database dan tabel user melalui http://localhost/phpmyadmin dengan mengklik opsi new, buat database dengan nama laundry_apps, lalu buat tabel user dengan menentukan nama kolom serta tipe datanya.



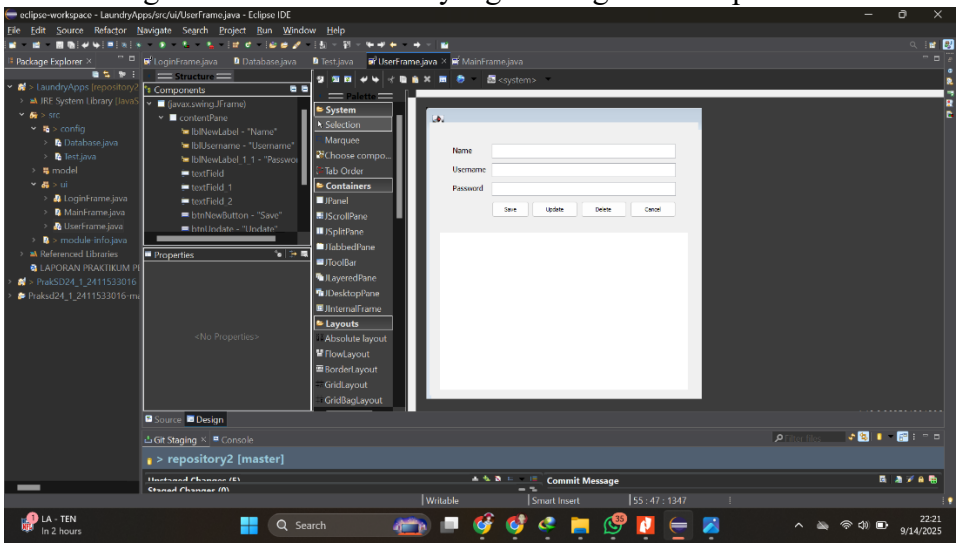
3) Koneksi ke Database MySQL
Setelah MySQL Connector terpasang, buat koneksi ke database dengan membuat package baru bernama config, kemudian buat kelas baru bernama Database.java. Isi kelas tersebut dengan kode yang diperlukan:



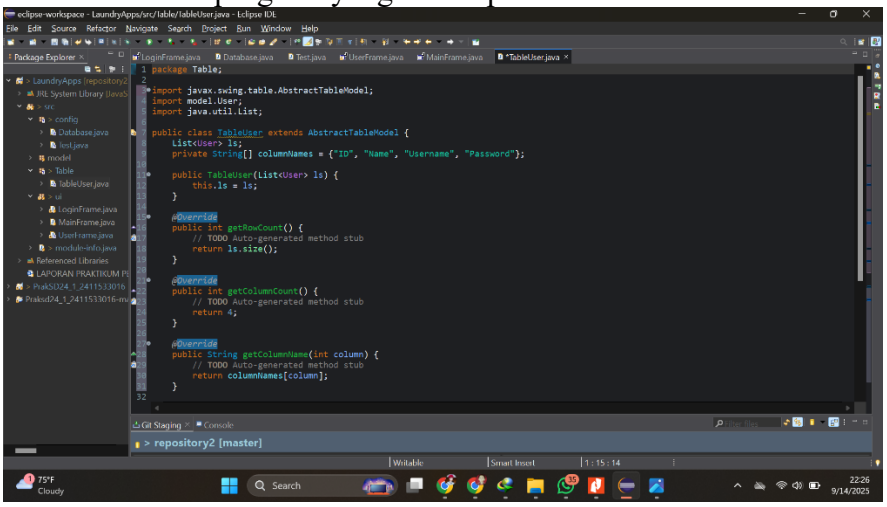
Import java.sql.* untuk menggunakan semua fungsi SQL, Gunakan method Connection untuk membuka koneksi ke database. Jika koneksi berhasil, method ini akan mengembalikan objek Connection, jika gagal, akan menampilkan pesan kesalahan.

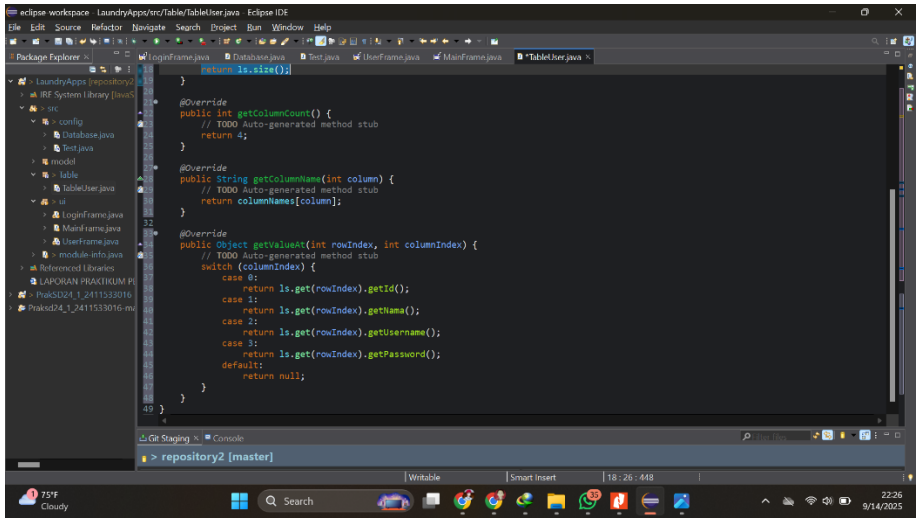
4) Desain Tampilan CRUD
Buatlah file JFrame baru pada package ui dengan nama UserFrame. Tambahkan komponen-komponen berikut:

a) JTextField dengan variabel txtName untuk input Nama, txtUsername untuk Username, dan txtPassword untuk Password.
b) JButton dengan variabel btnSave, btnUpdate, btnDelete, dan btnCancel.
c) JTable dengan variabel tableUser yang berfungsi menampilkan daftar User.



5) Pembuatan Table Model
Buat package baru bernama table dan buat kelas TableUser yang berfungsi sebagai model tabel untuk mengambil data dari database dan menampilkannya di tabel. Masukkan kode program yang sesuai pada kelas tersebut.

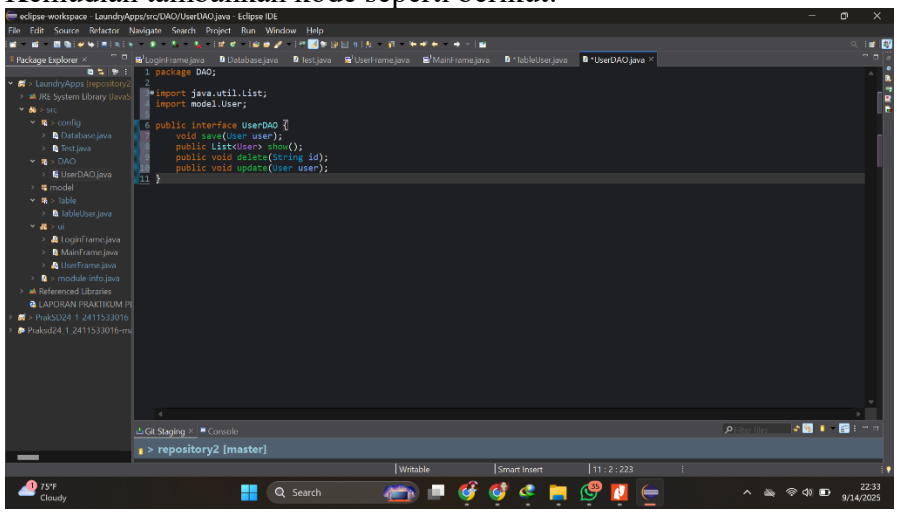




a) Kelas TableUser dibuat dengan mewarisi AbstractTableModel agar dapat berfungsi sebagai model data untuk komponen JTable.
b) Variabel ls yang bertipe List<User> berperan sebagai tempat penyimpanan seluruh objek user yang akan ditampilkan dalam tabel.
c) Array columnNames berisi daftar nama kolom tabel, yaitu ID, Name, Username, dan Password.
d) Konstruktor TableUser menerima parameter berupa list user kemudian menyimpannya ke dalam variabel ls.
e) Method getRowCount() mengembalikan total baris berdasarkan ukuran list ls, getColumnCount() mengembalikan jumlah kolom sesuai dengan panjang array columnNames, getColumnName(int column) mengembalikan nama kolom berdasarkan indeks, dan getValue di gunakan untuk mengambil data pada setiap sel tabel.

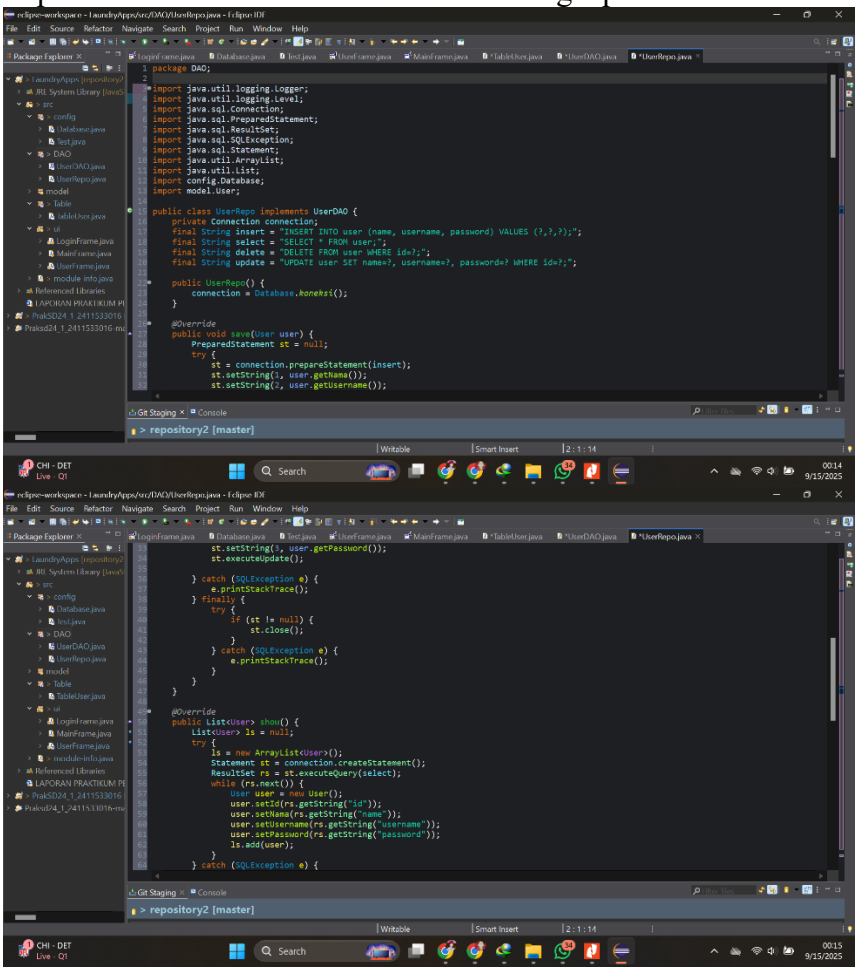
6) Membuat Fungsi DAO

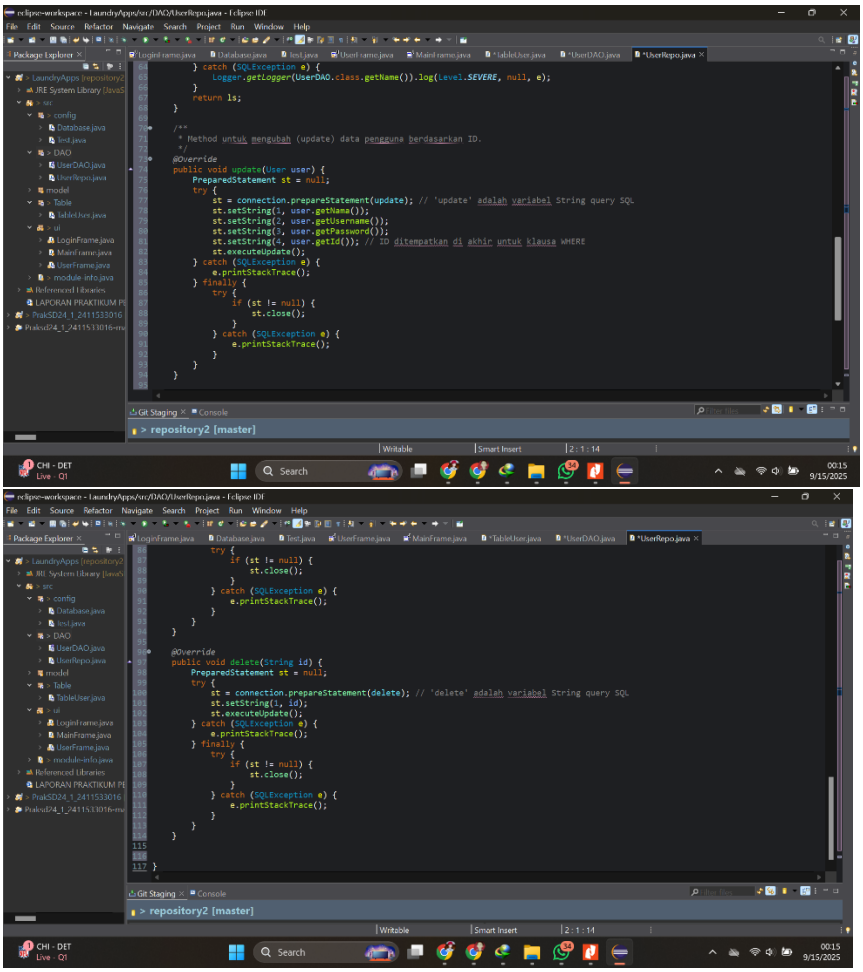
Buat package baru bernama DAO dan buat interface UserDAO di dalamnya. Kemudian tambahkan kode seperti berikut:



a) UserDAO merupakan interface yang menetapkan ketentuan dasar untuk operasi CRUD pada tabel user.
b) Method save(User user) berfungsi untuk menyimpan data baru, method show() mengembalikan seluruh daftar data user dalam bentuk List<User>. Method delete() digunakan untuk menghapus data user berdasarkan ID, sedangkan method update() bertugas memperbarui data user yang sudah tersedia.

7) Mengimplementasikan Fungsi DAO
Buat kelas baru bernama UserRepo dalam package DAO yang berfungsi sebagai implementasi dari interface UserDAO. Lengkapi kelas tersebut dengan kode berikut:





a) Variabel connection menyimpan objek koneksi yang didapatkan dari DataBase.koneksi().
b) Berbagai query SQL tersedia untuk mengelola data dalam database.
c) Pada konstruktor UserRepo(), koneksi ke database diinisialisasi sehingga dapat digunakan di seluruh metode CRUD.
d) Method save(User user) memakai PreparedStatement untuk menjalankan perintah INSERT, dimana data diambil dari objek User dan disimpan ke dalam database.
e) Method show() menggunakan Statement untuk mengeksekusi query SELECT, hasilnya berupa ResultSet yang diubah baris per baris menjadi objek User, kemudian dimasukkan ke dalam List<User> dan dikembalikan.
f) Method update(User user) melaksanakan query UPDATE menggunakan PreparedStatement, memperbarui data lama sesuai dengan nilai baru dari objek User.
g) Method delete menjalankan query DELETE dengan PreparedStatement menggunakan parameter ID, sehingga data dengan ID tersebut akan dihapus dari database setelah eksekusi.

D. Kesimpulan

Praktikum ini menyimpulkan bahwa Pemrograman Berorientasi Objek (PBO) memberikan dasar penting dalam pengelolaan data pada sebuah aplikasi. Konsep DAO memisahkan antara logika akses data dan logika bisnis, sedangkan interface menyediakan kerangka kerja agar implementasi menjadi lebih konsisten. Penggunaan MySQL Connector memungkinkan aplikasi Java untuk berkomunikasi dengan database secara efisien, mulai dari proses penyimpanan hingga penghapusan data.