

NIM : 2201010109

NAMA : I Gusti Ayu Putu Widhya Iswari

Project : Form Pendaftaran Mahasiswa

Deskripsi :

Aplikasi ini adalah aplikasi desktop yang digunakan untuk mengelola data pendaftaran mahasiswa baru. Aplikasi ini dikembangkan menggunakan bahasa pemrograman Java dengan konsep pemrograman berorientasi objek (OOP). Aplikasi ini memungkinkan pengguna untuk menambahkan, mengedit, menghapus, dan menampilkan data mahasiswa yang telah mendaftar. Tujuan dari aplikasi ini adalah untuk memudahkan pengelolaan data pendaftaran mahasiswa baru sehingga proses administrasi menjadi lebih efisien dan terorganisir.

Fitur:

1. Tambah Data Mahasiswa
Form untuk memasukkan nama, alamat, asal sekolah, dan nomor telepon.
2. Edit Data Mahasiswa
Form untuk memperbarui informasi yang telah dimasukkan.
3. Hapus Data Mahasiswa
Fitur untuk menghapus data mahasiswa dari daftar.
4. Simpan Data Mahasiswa
Fitur untuk menyimpan data mahasiswa yang telah dimasukkan ke dalam basis data sementara atau permanen.
5. Selesai
Menyelesaikan proses pendaftaran dan keluar dari aplikasi.

6. Teknik Pemrograman OOP yang Digunakan

1 Encapsulation

- Penggunaan Modifier Access: Variabel dan metode yang digunakan dalam kelas `'FormDMhs'` sebagian besar menggunakan modifier `'private'` untuk melindungi data dan mencegah akses langsung dari luar kelas.
- Metode untuk Mengelola Data: Data mahasiswa diakses dan dimanipulasi melalui metode seperti `'StoreData()'`, `'UpdateData()'`, dan `'destroyData()'`, yang mengenkapsulasi logika bisnis aplikasi.

2 Inheritance

- Pewarisan dari Kelas JFrame: Kelas `FormDMhs` mewarisi dari kelas `javax.swing.JFrame` untuk mengimplementasikan antarmuka grafis. Ini memungkinkan `FormDMhs` untuk menggunakan dan memperluas fungsionalitas dari `JFrame`.

3 Polymorphism

- Overriding Methods: Metode seperti ` initComponents()` dan event-handler methods (` ActionPerformed`, `MouseClicked`) di-override untuk menyesuaikan perilaku standar dari komponen GUI.

4 Abstraction

- Penggunaan Antarmuka dan Kelas Abstrak: Meskipun tidak secara eksplisit terlihat dalam kode ini, konsep abstraksi diterapkan dengan menggunakan metode yang menyembunyikan implementasi detail dari logika database (seperti koneksi database di metode `newkoneksi()` dari kelas `Koneksi`).

7. Alasan Penggunaan Teknik OOP

- Encapsulation: Untuk menjaga integritas data dan memastikan data hanya bisa diakses dan dimodifikasi melalui metode yang telah didefinisikan.
- Inheritance: Untuk memanfaatkan fungsionalitas yang sudah ada dan mengurangi du

Aplikasi kode.

- Polymorphism: Untuk memungkinkan fleksibilitas dalam pengelolaan event dan interaksi GUI.
- Abstraction: Untuk menyederhanakan kompleksitas dengan menyembunyikan detail implementasi dan fokus pada fungsionalitas utama.