**Laporan Tugas Proyek**

**Pemrograman Berorientasi Objek**

**SiKKoLa (School Management)**



**Disusun Oleh :**

|  |  |
| --- | --- |
| 11321009 | Cici Yanti Lubis |
| 11321021  11321025  11321029  11321037 | Luana Breka Banjarnahor  Theresia Herlita Sinaga  Novrael Gabriel L Marbun  Samuel Jefri S Siahaan |

**Institut Teknologi Del**

**Fakultas Vokasi**

**D3 Teknologi Informasi**

**Laguboti**

**2022**

RUBRIK PENILAIAN PROYEK

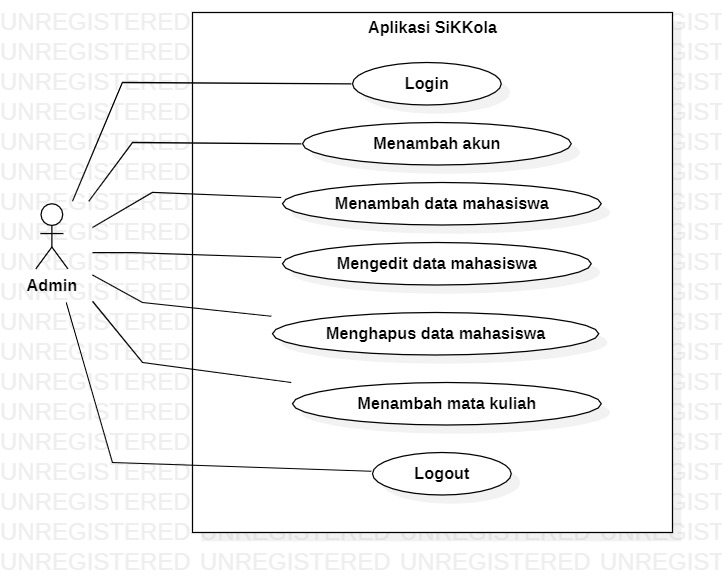
1. Beri deskripsi secara singkat ada berapa dan uraian mengenai fungsi yang anda buat dalam proyek ini.
2. Jelaskan penerapan konsep OOP yang anda terapkan pada proyek anda kemudian sertakan screenshot dari sistem yang anda bangun berdasarkan point berikut:
   1. Abstraction
   2. Encapsulation
   3. Inheritance
   4. Polymorphism
   5. Interface
   6. Exception handling
3. Rekam video penjelasan dari No 1 dan No 2. Upload pada Youtube dan masukkan link tersebut kedalam laporan ini.
4. Pengumpulan laporan dan *sourcecode* adalah hari sabtu, 5 november 2022, 23.59 WIB di E-Course. Template *filename*:

3digitbelakangNIM1\_3digitbelakangNIM2\_3digitbelakangNIM3\_3digitbelakangNIM4-Proyek PBO\_Topik.zip

Contoh: 011\_022\_033\_044-Proyek PBO\_Sistem Perpustakaan.zip.  
Keterlambatan pengumpulan akan mendapatkan pengurangan point sebanyak berapa menit anda terlambat.

1. **FUNCTIONAL DESCRIPTION**

Pada aplikasi yang dibangun, terdapat fungsi-fungsi yang dapat dilakukan oleh admin. Deskripsi fungsional aplikasi digambarkan pada use case diagram yang dapat dilihat pada gambar 1.



Gambar 1. Use Case Diagram

Fungsi-fungsi dari aplikasi yang dibangun dijelaskan dibawah ini.

1. Fungsi Menambah Akun

Fungsi ini digunakan admin untuk dapat mengakses aplikasi dengan membuat akun terlebih dahulu

1. Fungsi Menambah Mata Kuliah

Fungsi ini digunakan admin untuk menambah mata kuliah terbaru ke dalam aplikasi.

1. Fungsi Menambah Data Mahasiswa

Fungsi ini digunakan admin untuk menambahkan data mahasiswa ke dalam aplikasi.

1. Fungsi Mengedit Data Mahasiswa

Funngsi ini digunakan admin untuk mengedit data mahasiswa yang ingin di perbaharui pada dalam aplikasi.

1. Fungsi Menghapus Data Mahasiswa

Fungsi ini digunakan admin untuk menghapus data mahasiswa pada aplikasi.

1. Fungsi *Login*

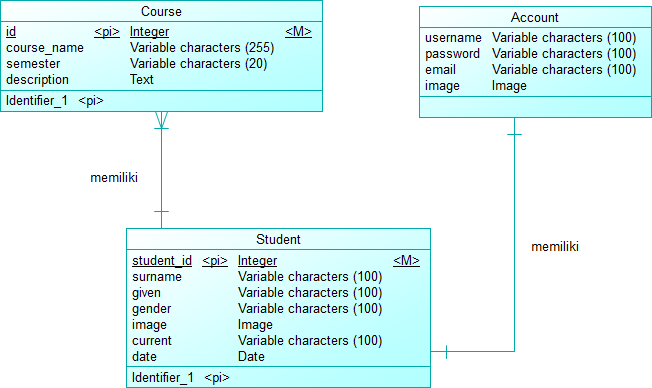
Fungsi ini digunakan admin untuk melakukan *login* ke dalam aplikasi. Admin harus *login* terlebih dahulu untuk dapat mengakses aplikasi

1. Fungsi *Logout*

Fungsi ini digunakan admin untuk keluar dari aplikasi ketika telah selesai.

1. **DESIGN CONCEPTUAL DATA MODEL**

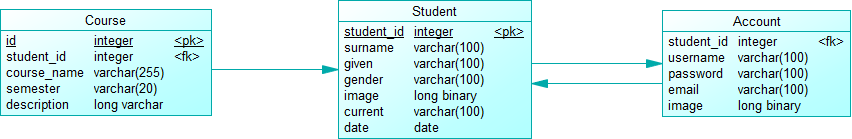
*Conceptual data model* dari Aplikasi SiKKola dapat dilihat pada Gambar 2.

****

Gambar 2. Conceptual Data Model

1. **DESIGN PHYSICAL DATA MODEL**

*Physical data model* dari Aplikasi SiKKola dapat dilihat pada Gambar 3.

****

Gambar 3. Physical Data Model

1. **DESIGN TABLE STRUCTURE**

Pada bagian ini berisi tabel pada rancangan global yang dirinci satu per satu. Untuk setiap tabel mengandung nama tabel, jenis, volume, dan primary key.

* 1. **Tabel Student**

Identifikasi/Nama : Student

Deskripsi Isi : Tabel ini berisi tentang data mahasiswa yang akan

menggunkaan aplikasi

Jenis : Tabel master

Volume : 7 *field*

*Primary Key*  : student\_id

**Tabel 1. Tabel Student**

| **Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Default** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| Student\_id | Berisi id dari student | Int | NO | None | *Primary key* |
| surname | Berisi username dari student | VarChar(100) | NO | None | *Attribute*  *Non Key* |
| Given | Berisi data student | VarChar(100) | NO | None | *Attribute*  *Non Key* |
| Gender | Berisi jenis kelamin dari student | VarChar(100) | NO | None | *Attribute*  *Non Key* |
| Image | Berisi gambar profil dari student | Image | NO | None | *Attribute*  *Non Key* |
| Current | Berisi data student | VarChar(100) | No | None | *Attribute*  *Non Key* |
| Date | Berisi tanggallahir dari student | Date | Yes | Null | *Attribute*  *Non Key* |

* 1. **Tabel Student**

Identifikasi/Nama : Course

Deskripsi Isi : Tabel ini berisi data mengenai mata kuliah

Jenis : Tabel master

Volume : 4 *field*

*Primary Key*  : id

**Tabel 2. Tabel Course**

| **Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Default** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| id | Berisi id\_ course yang tersimpan dalam aplikasi | Integer | NO | None | *Primary key* |
| Course\_name | Berisi nama mata kuliah | VarChar(255) | NO | None | *Attribute*  *Non Key* |
| semester | Berisi semester berapa yang ditempuh | VarChar(255) | NO | None | *Attribute*  *Non Key* |
| deskription | Berisi deskripsi dari mata kuliah | Long VarChar | NO | None | *Attribute*  *Non Key* |

* 1. **Tabel Student**

Identifikasi/Nama : Account

Deskripsi Isi : Tabel ini berisi data akun

Jenis : Tabel master

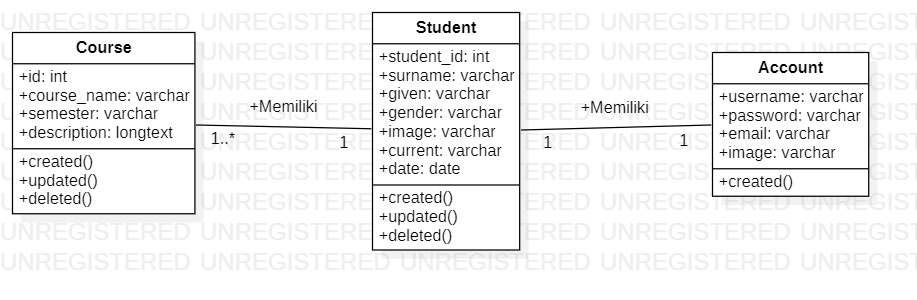
Volume : 4 *field*

*Primary Key*  : id

**Tabel 3. Tabel Account**

| **Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Default** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| username | Berisi username pada akun | VarChar(100) | NO | None | *Primary key* |
| password | Berisi password dari akun | VarChar(100) | NO | None | *Foreign Key* |
| email | Berisi email dari akun | VarChar(100) | NO | None | *Attribute*  *Non Key* |
| image | Berisi image dari akun | VarChar(255) | NO | None | *Attribute*  *Non Key* |

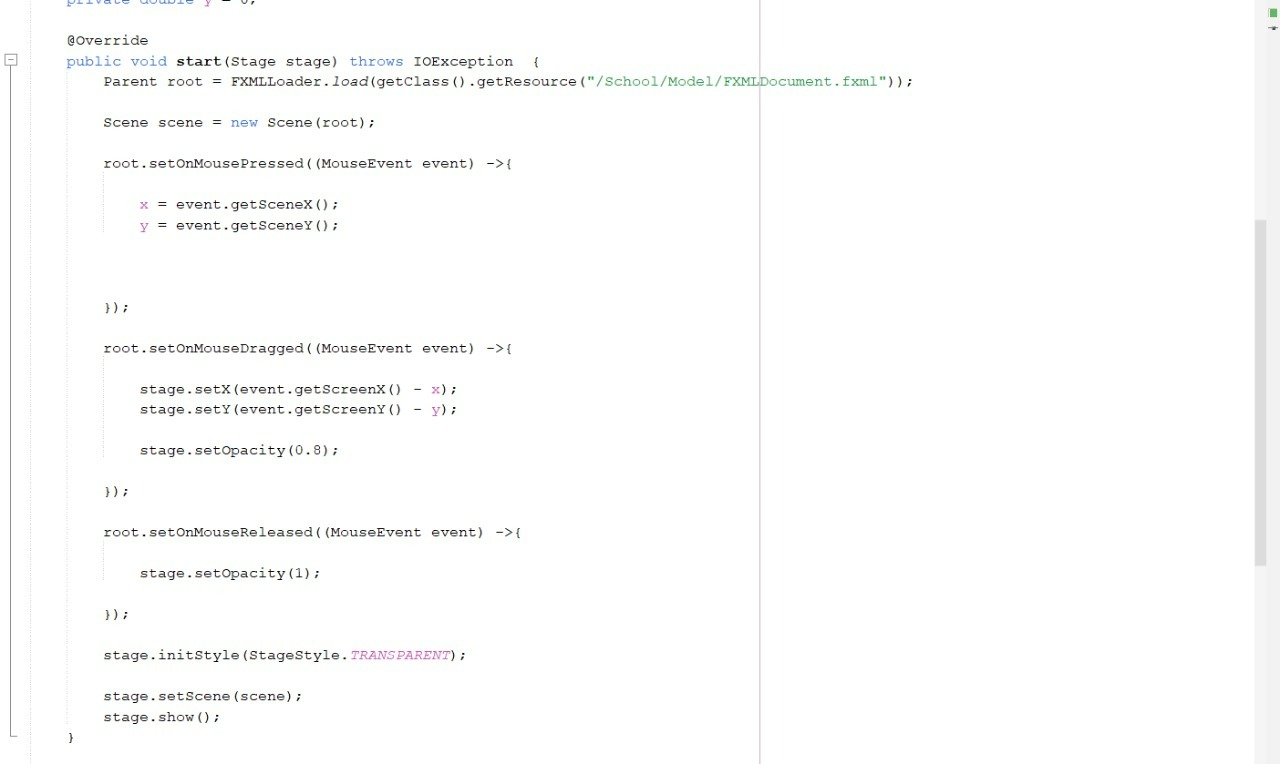
1. **DESIGN CLASS DIAGRAM**

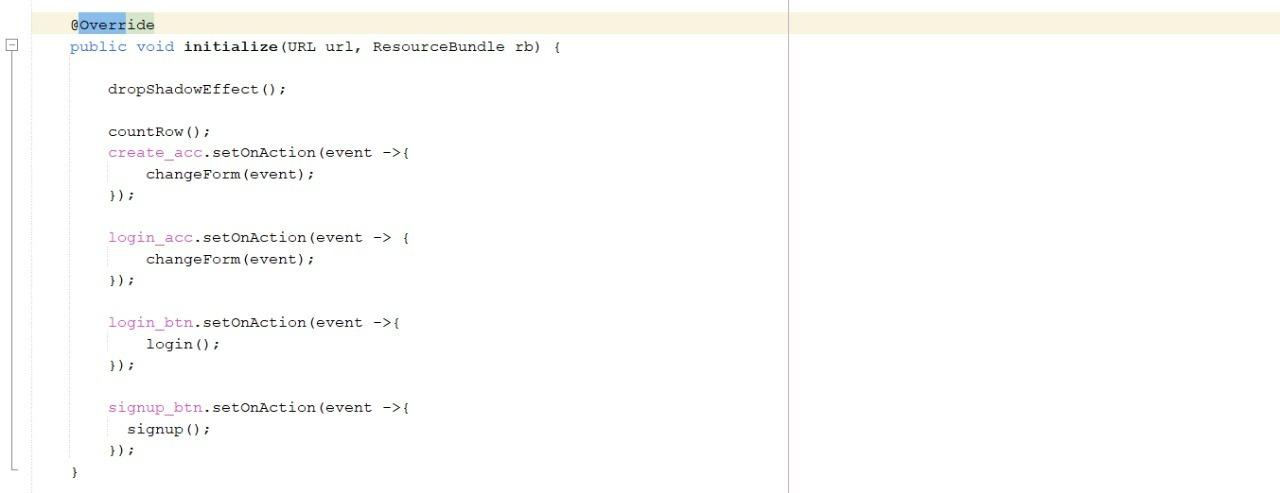
****

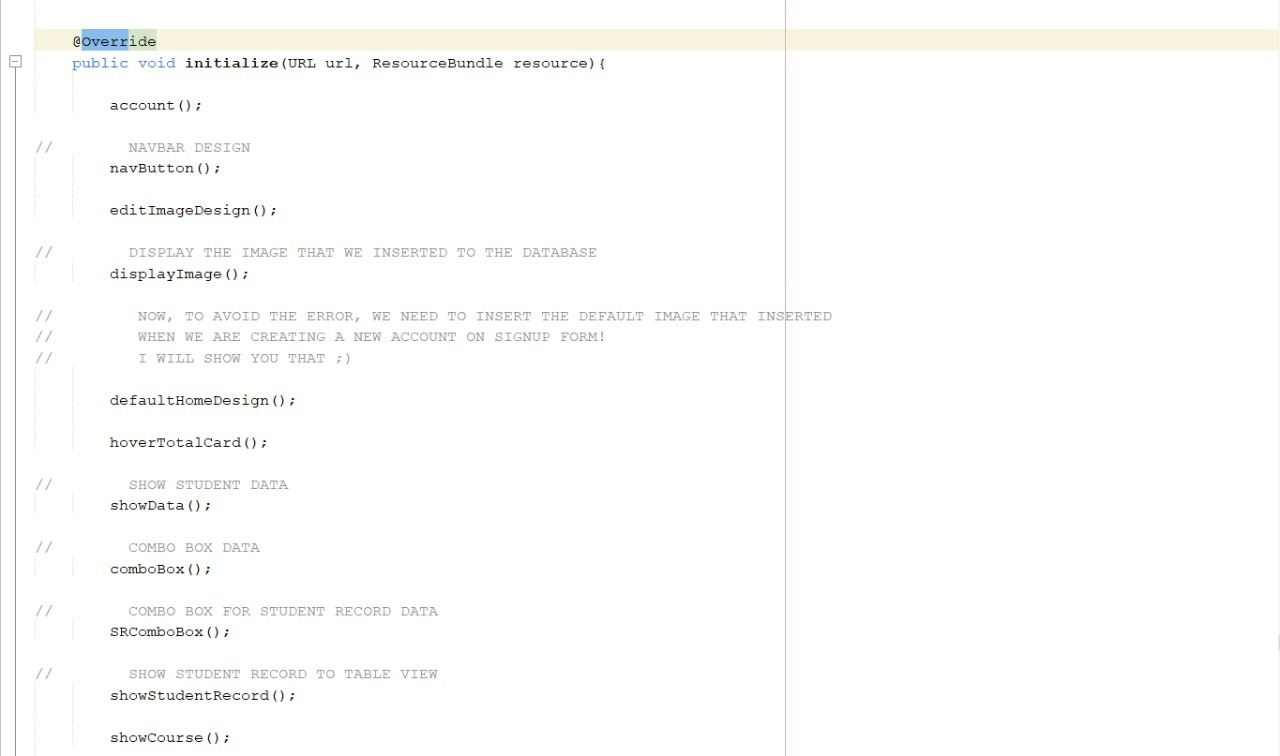
Gambar 4. Diagram Class

1. **OOP IMPLEMENTATION**
   1. **Polymorphism**

Dibawah ini merupakan contoh pengimplementasian konsep OOP yaitu Polymorphism. Pada program ini menjalankan program untuk login, membuat akun serta logout dari aplikasi. Kemudian pada konsep ini dapat mengimplementasikan CRUD pada akun yang telah dibuat.

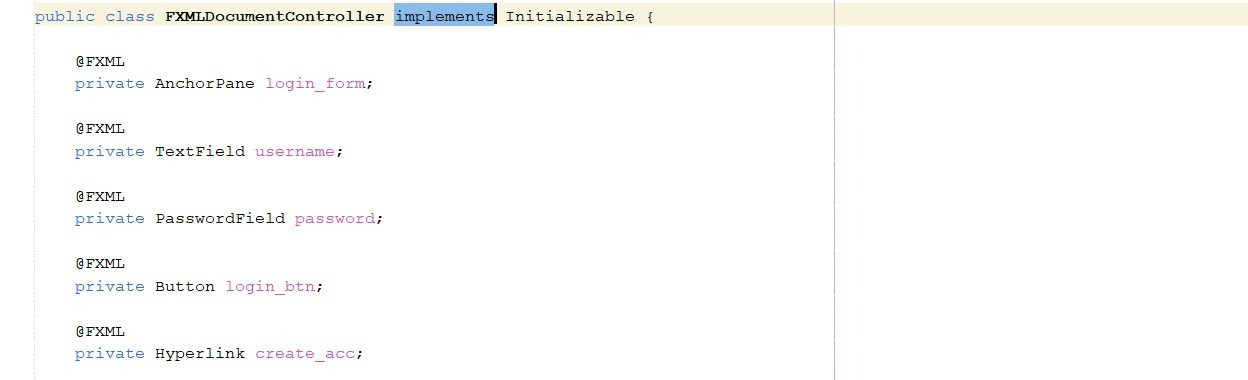


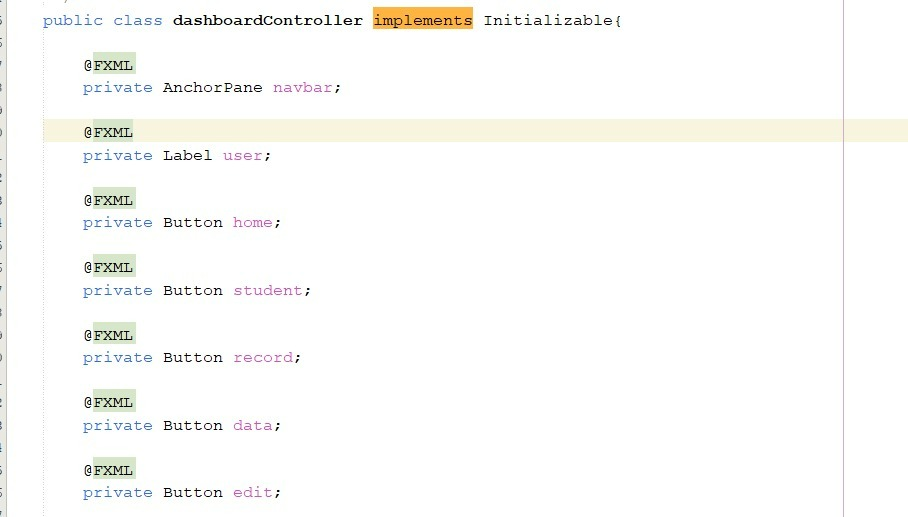




* 1. **Interface**

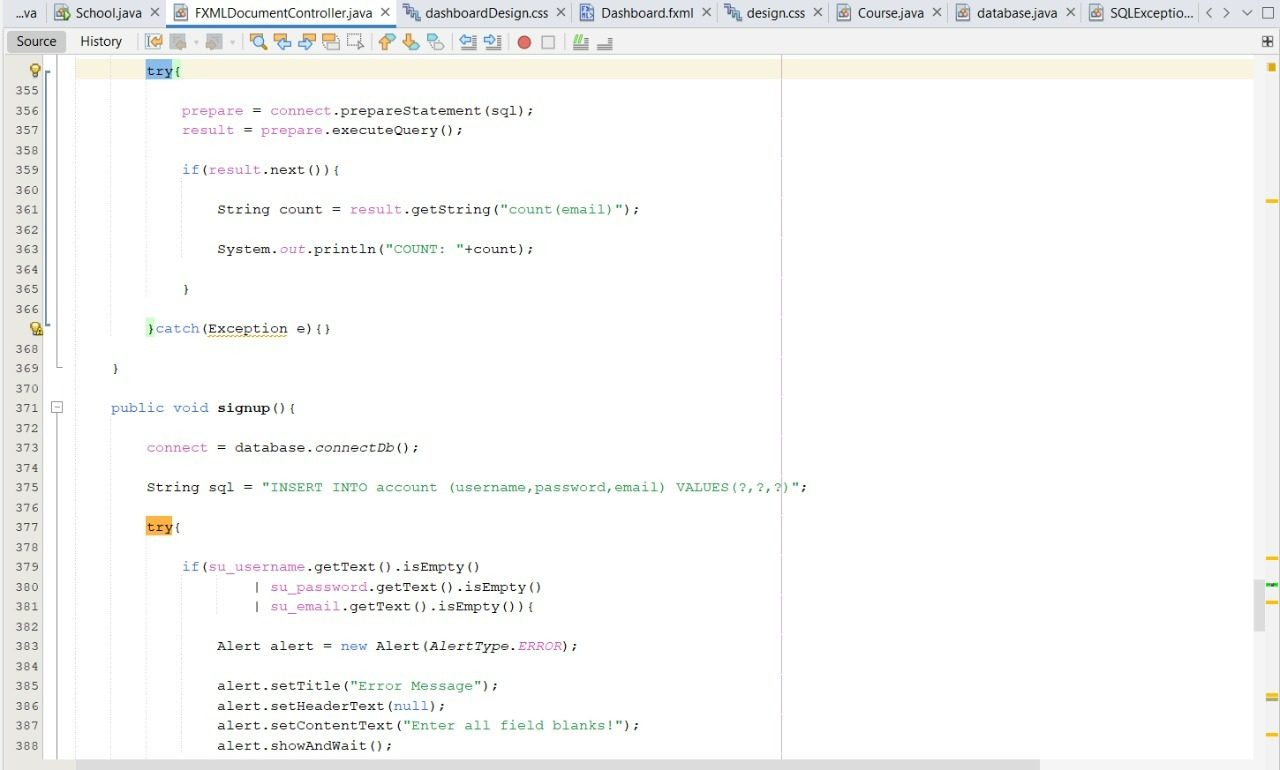
Dibawah ini merupakan contoh pengimplementasian konsep OOP yaitu Interface. Penggunaan interface dituliskan setelah keyword implements. Pada program ini menampilkan form untuk login ke dalam aplikasi. Selain itu setelah melakukan login admin akan otomasi ke halaman utama yang menampilkan berbagai menu pada aplikasi.

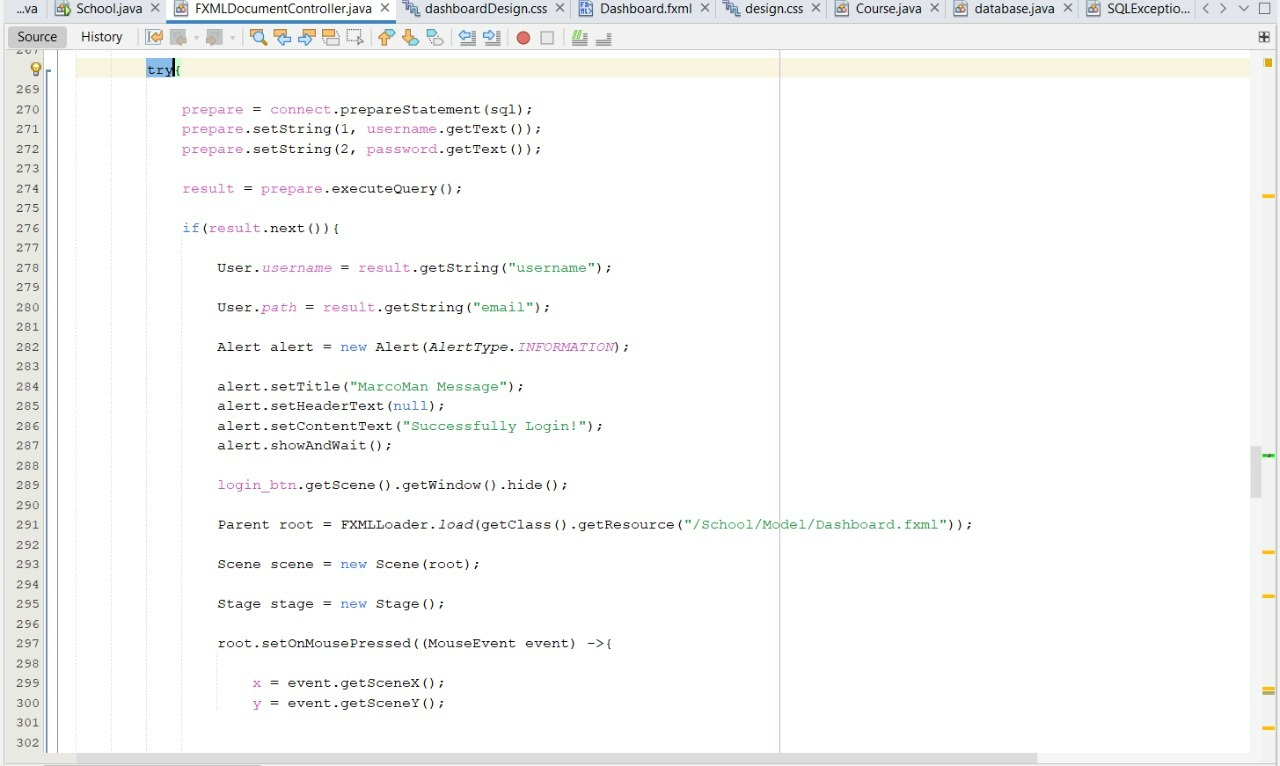


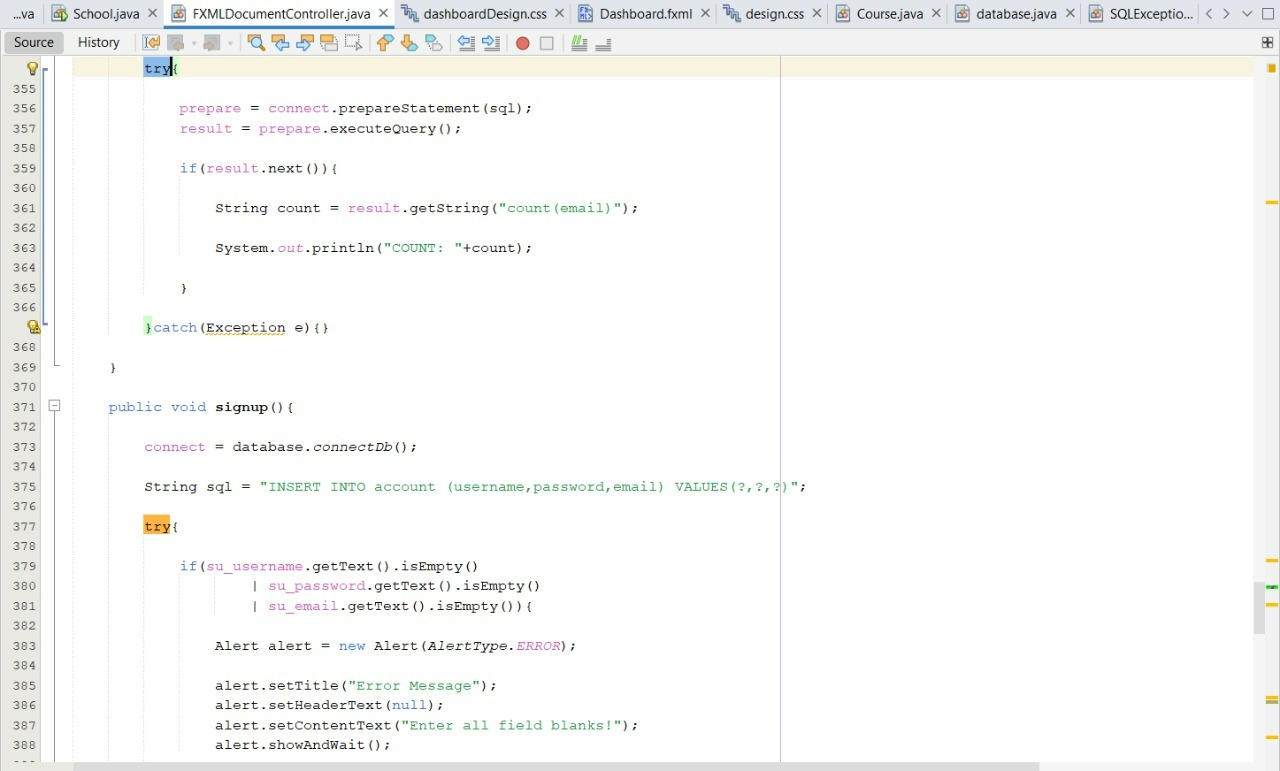


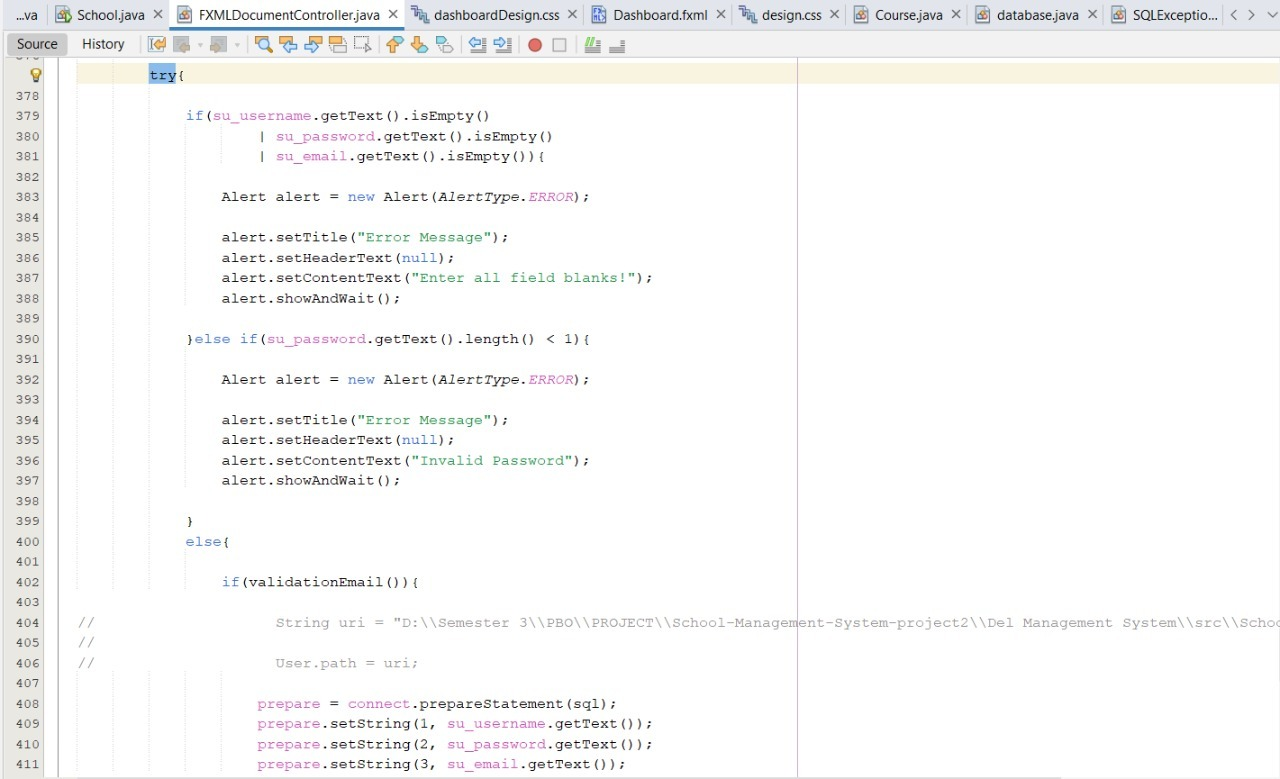
* 1. **Exception handling**

Dibawah ini merupakan contoh pengimplementasian konsep OOP yaitu Exception Handling. Pada program ini menjalankan fungsi dalam menambah akun. Dalam kode program ini telah terbuat suatu alert yang berfungsi untuk memberi pemberitahuan bahwa data yang telah di input benar atau salah. Dengan adanya alert tersebut dapat membantu user dalam menggunakan aplikasi.









1. **LINK YOUTUBE**

[**https://youtu.be/s01ChOhjEyQ**](https://youtu.be/s01ChOhjEyQ)

1. **LINK SOURCE CODE**[**https://drive.google.com/drive/folders/1Bvn1hOpD7JPBZ85N01xpOLhnx10\_2-iW?usp=share\_link**](https://drive.google.com/drive/folders/1Bvn1hOpD7JPBZ85N01xpOLhnx10_2-iW?usp=share_link)