Laporan Ujian Akhir Semester

Pemrograman Berorientasi Objek



oleh

Nama: Putu Dian Shinta Prativi

NIM: 222212822

Kelas: 2KS1

**BAB I**

**PENDAHULUAN**

1. Penjelasan Singkat Aplikasi yang Dikembangkan

Aplikasi yang saya kembangkan dalam rangka memenuhi Ujian Akhir Semester mata kuliah Pemrograman Berorientasi Objek bernama SAMPEL (Sistem Aplikasi Manajemen Pelanggaran). Aplikasi ini merupakan hasil pemikiran saya selaku anggota Satuan Penegak Disiplin (SPD) divisi Absensi Operasi Pengamanan dan Pelaporan (AOPP) yang rutin melakukan pengecekan kedisiplinan mahasiswa dan merekap pelanggaran mahasiswa. Melalui aplikasi yang saya buat ini, harapannya dapat dikembangkan lebih lanjut supaya dapat benar-benar membantu dalam mengerjakan tugas sebagai anggota SPD.

Dalam aplikasi ini terdapat dua *role* yaitu anggota SPD divisi AOPP sebagai admin, dan mahasiswa Politeknik Statistika STIS selain anggota SPD divisi AOPP sebagai user. Admin dapat menginput pelanggaran-pelanggaran yang dilakukan mahasiswa, mengecek *range* perkiraan poin pelanggaran mahasiswa, dan mengekspor rekapan pelanggaran ke dalam fail berekstensi .csv. Sedangkan user dapat mengecek *range* perkiraan total poin, memperbarui poin fiks yang telah diberitahu oleh BAAK, dan mengekspor rekapan pelanggaran miliknya ke dalam fail berekstensi .csv.

Aplikasi yang saya kembangkan dapat digunakan oleh mahasiswa Polstat STIS dengan asumsi database yang terhubung dengan aplikasi saya memiliki informasi NIM dan password yang biasanya digunakan mahasiswa untuk mengakses SIPADU.

Website ini mengharuskan pengguna untuk login terlebih dahulu. Berikut adalah informasi akun yang dapat digunakan untuk login:

Admin:

1. NIM: **222212822**

Password: **1234**

1. NIM: **212212826**

Password: **1235**

User:

1. NIM: **212212825**

Password: **1234**

1. NIM: **222212707**

Password: **2222**

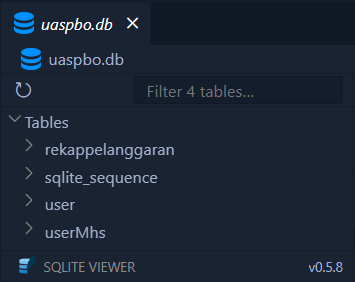
**BAB II**

**ISI**

1. Database yang Digunakan

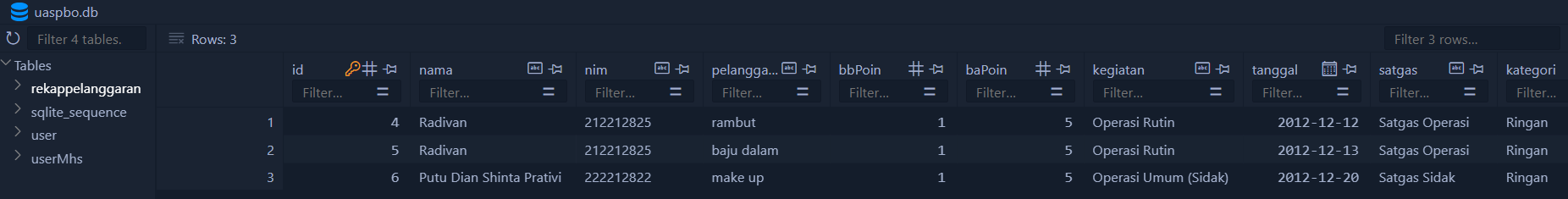
DBMS yang digunakan dalam pembuatan database untuk website ini yaitu SQLite.

Berikut screenshot dari database yang digunakan:



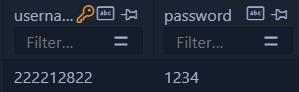
Pada gambar di atas dapat diketahui bahwa database yang digunakan bernama uaspbo.db. Dalam database tersebut terdapat tiga tabel yaitu:

* rekappelanggaran



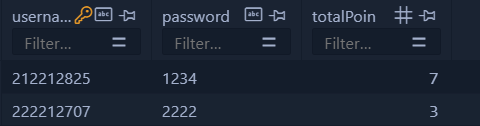
Tabel ini digunakan untuk menyimpan informasi mengenai pelanggaran-pelanggaran yang telah diinput oleh admin. Informasi-informasi tersebut antara lain id, nama mahasiswa, nim mahasiswa, pelanggaran yang dilakukan, batas bawah *range* poin, batas atas range poin, kegiatan saat penindakan pelanggaran, tanggal penindakan, satgas (anggota SPD) yang menindak, dan kategori pelanggaran.. Id tidak diinput oleh pengguna pada aplikasi, tetapi otomatis bertambah (AUTO\_INCREMENT) saat informasi pelanggaran baru ditambahkan ke dalam tabel.

* user



T Tabel ini digunakan untuk menyimpan informasi admin antara lain email dan password untuk log in. Informasi tersebut masing-masing disimpan dalam kolom-kolom bernama email sebagai primary key dan password.

* userMhs

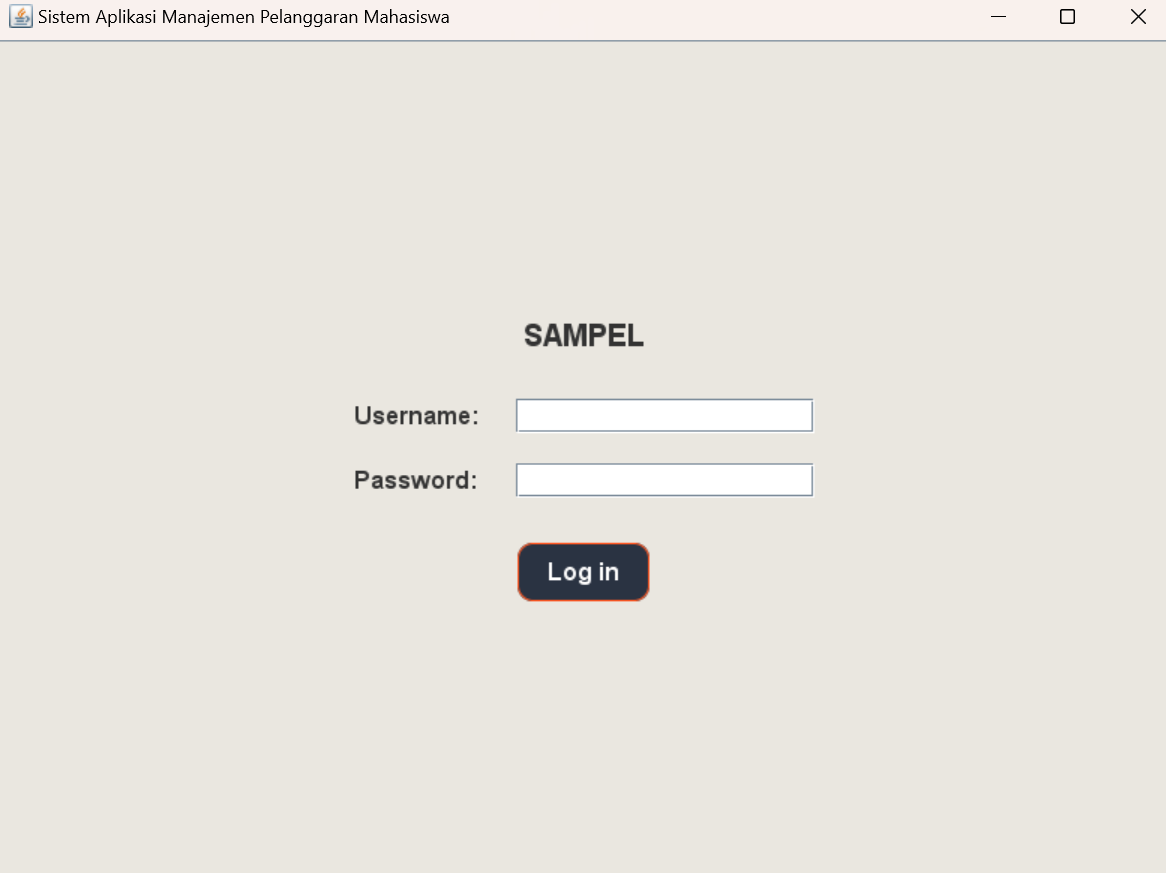


Tabel ini digunakan untuk menyimpan informasi user yang merupakan mahasiswa selain anggota SPD divisi AOPP, antara lain email dan password untuk log in. Informasi tersebut masing-masing disimpan dalam kolom-kolom bernama email sebagai primary key dan password.

1. Screenshot Halaman Web

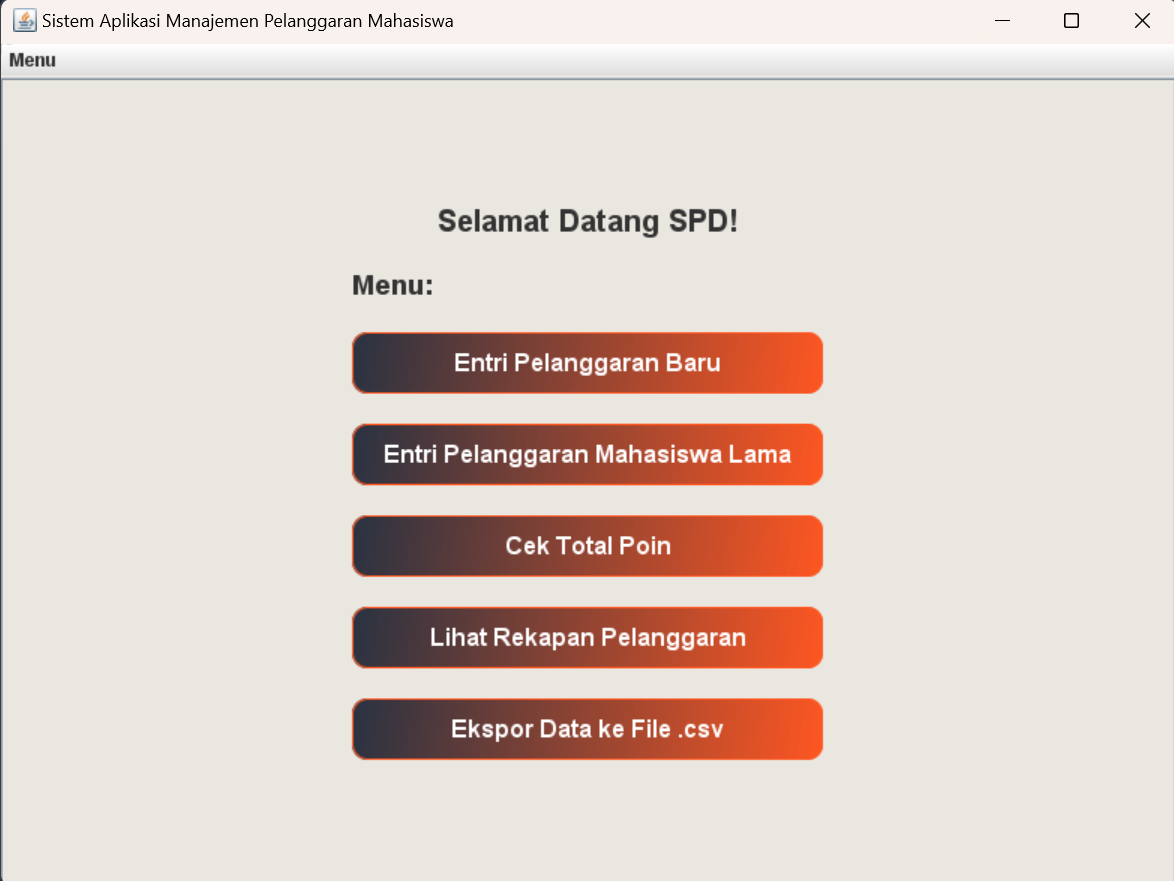
Berikut *screenshot* dari setiap panel untuk admin beserta contoh penggunaannya:

* LoginPanel.java



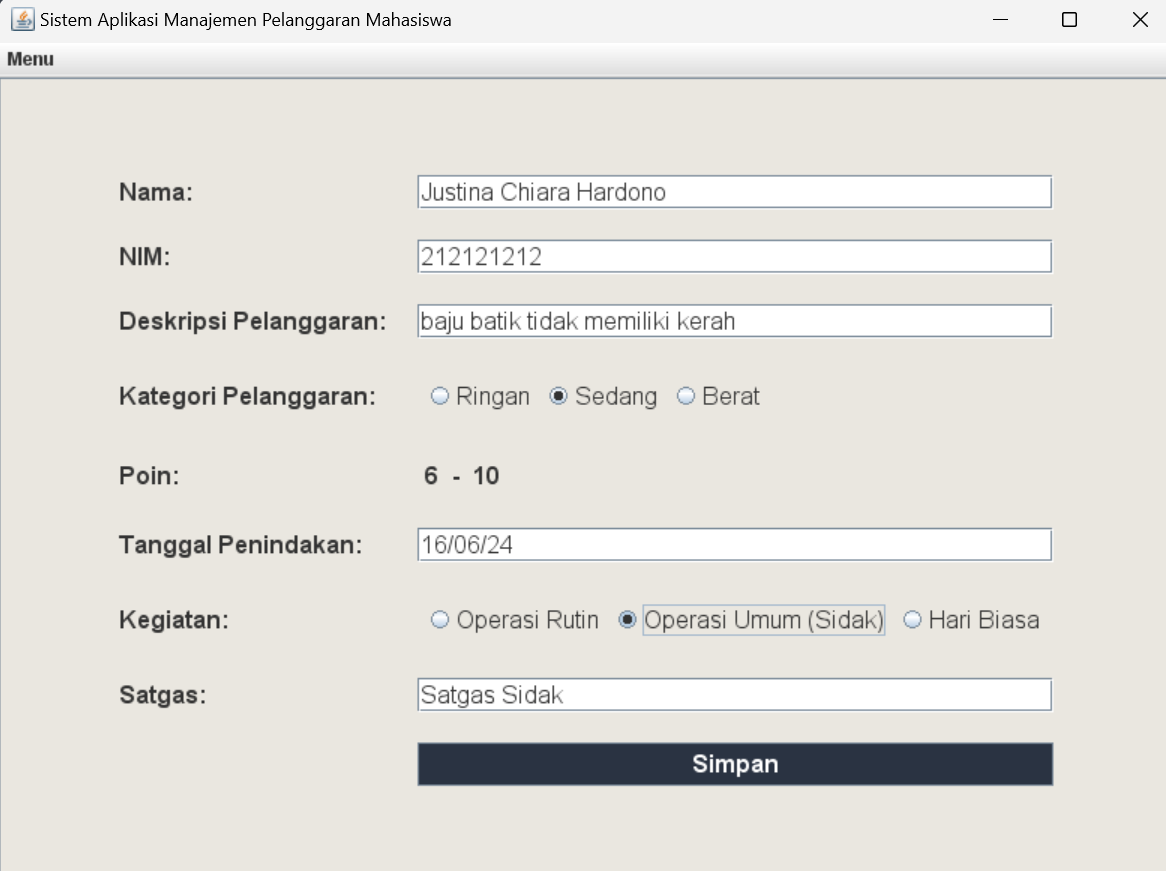
LoginPanel ini berlaku baik untuk admin maupun user. Apabila username dan password yang dimasukkan terdaftar dalam tabel bernama user, maka akan diarahkan ke panel-panel untuk admin. Namun apabila username dan password yang dimasukkan terdaftar dalam tabel bernama userMhs, maka akan diarahkan ke panel-panel untuk user.

* HomePanel.java



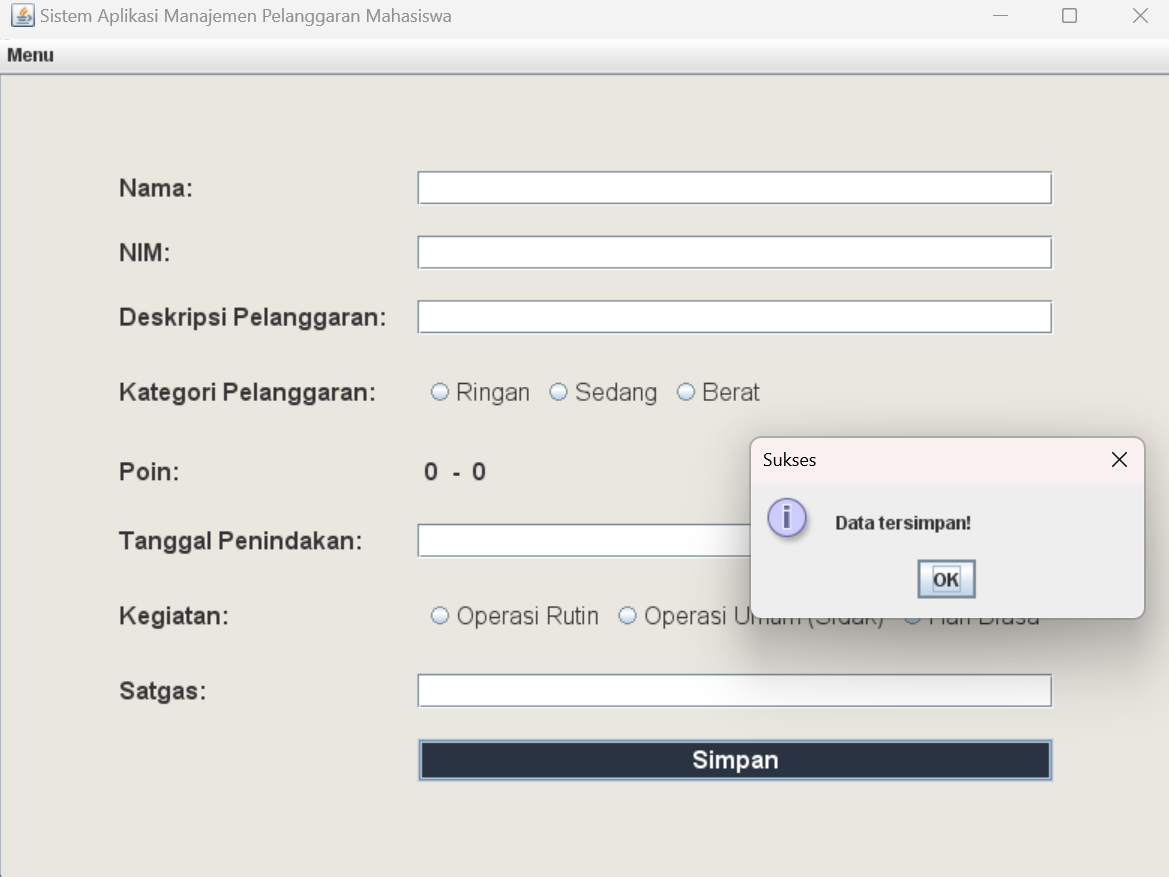
Setelah admin berhasil login, maka akan diarahkan ke HomePanel berisi menu dengan pilihan-pilihan untuk entri pelanggaran baru, entri pelanggaran mahasiswa yang sebelumnya sudah pernah diinput, cek total poin mahasiswa tertentu, melihat rekapan pelanggaran, dan ekspor data ke fail .csv.

* EntryPanel.java



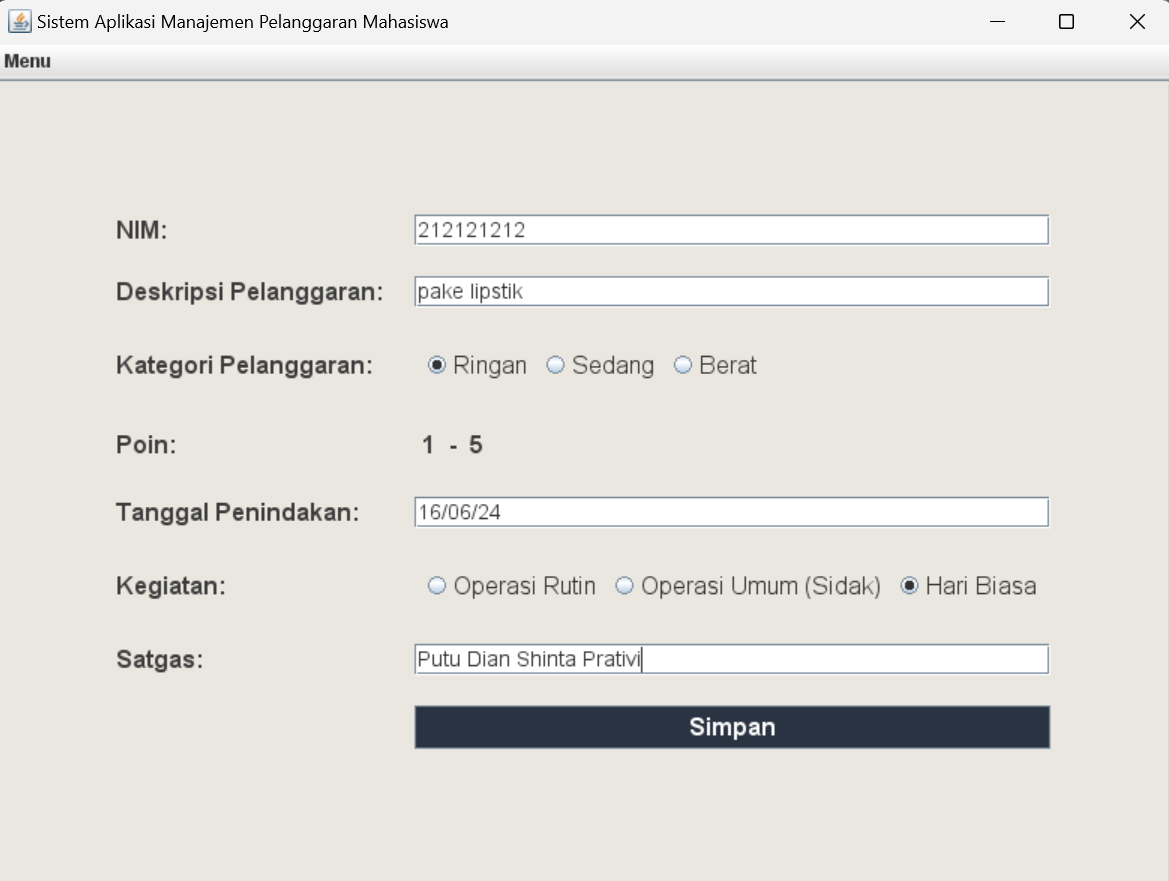
Panel ini akan muncul apabila admin mengklik Entri Pelanggaran Baru pada HomePanel. Pada *screenshot* di atas, telah tertera contoh pengisian formulir untuk mengentri data pelanggaran. Jika admin memilih radio button Ringan maka poin akan otomatis bertuliskan 1 – 5 ( satu sampai lima), sedangkan radio button Sedang jika dipilih akan membuat poin otomatis bertuliskan 6 – 10, dan radio button Berat jika dipilih akan membuat poin otomatis bertuliskan 11 – 40.

Begitu juga jika radio button Operasi Rutin dipilih maka satgas akan otomatis bertuliskan Satgas Operasi Rutin, sedangkan jika Operasi Umum (Sidak) yang terpilih akan membuat satgas otomatis bertuliskan Satgas Sidak, dan jika Hari Biasa yang terpilih maka field satgas akan kosong karena nama satgas harus ditulis sendiri oleh admin.

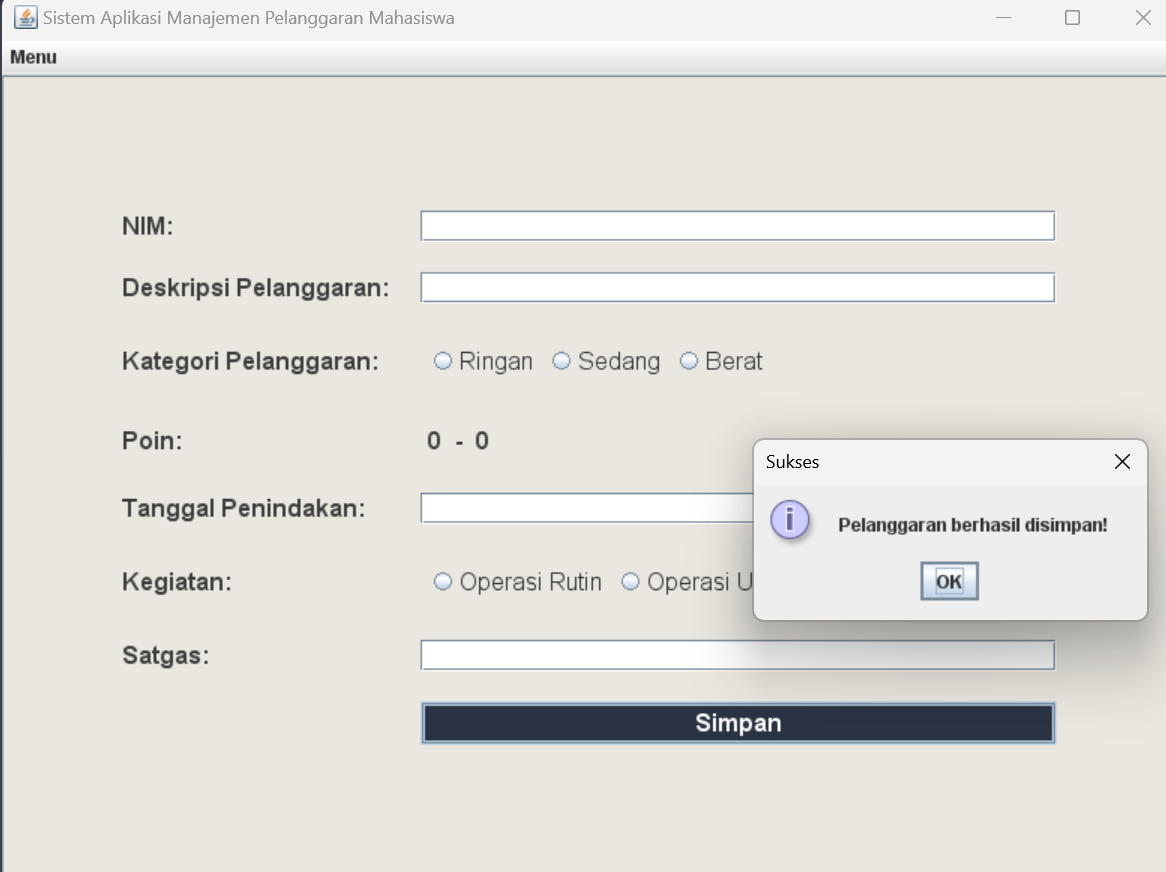


Gambar di atas merupakan kondisi panel apabila data yang diinput tadi berhasil tersimpan setelah admin mengklik tombol Simpan. Data tadi akan otomatis terekam pada tabel rekappelanggaran dalam database.

* EntryExistingPanel.java

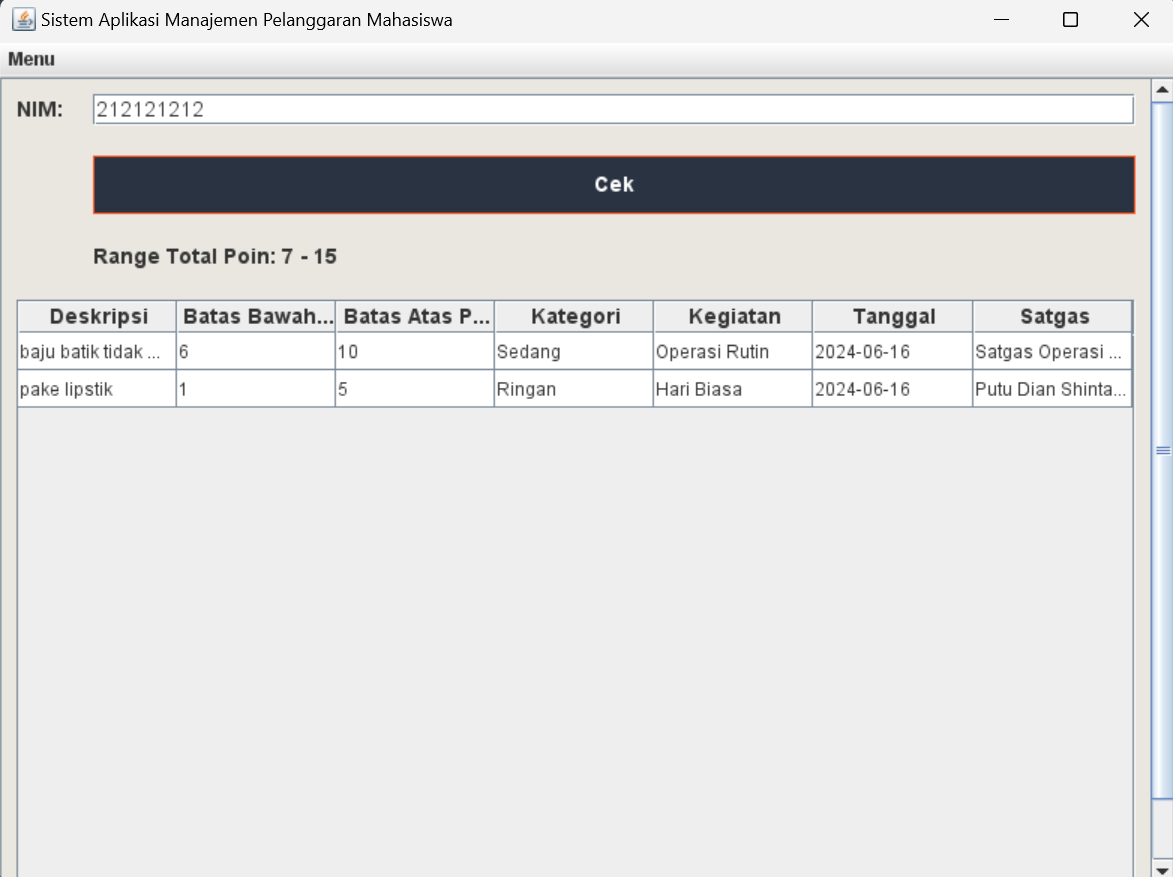


Panel ini akan muncul apabila admin mengklik Entri Pelanggaran Mahasiswa Lama pada HomePanel. Panel ini berfungsi untuk menginput pelanggaran mahasiswa yang sudah pernah tercatat melanggar sebelumnya. Pada *screenshot* di atas tertera contoh pengisian formulir pada panel ini. Fungsional yang ada pada panel ini kurang lebih sama dengan panel EntryPanel.



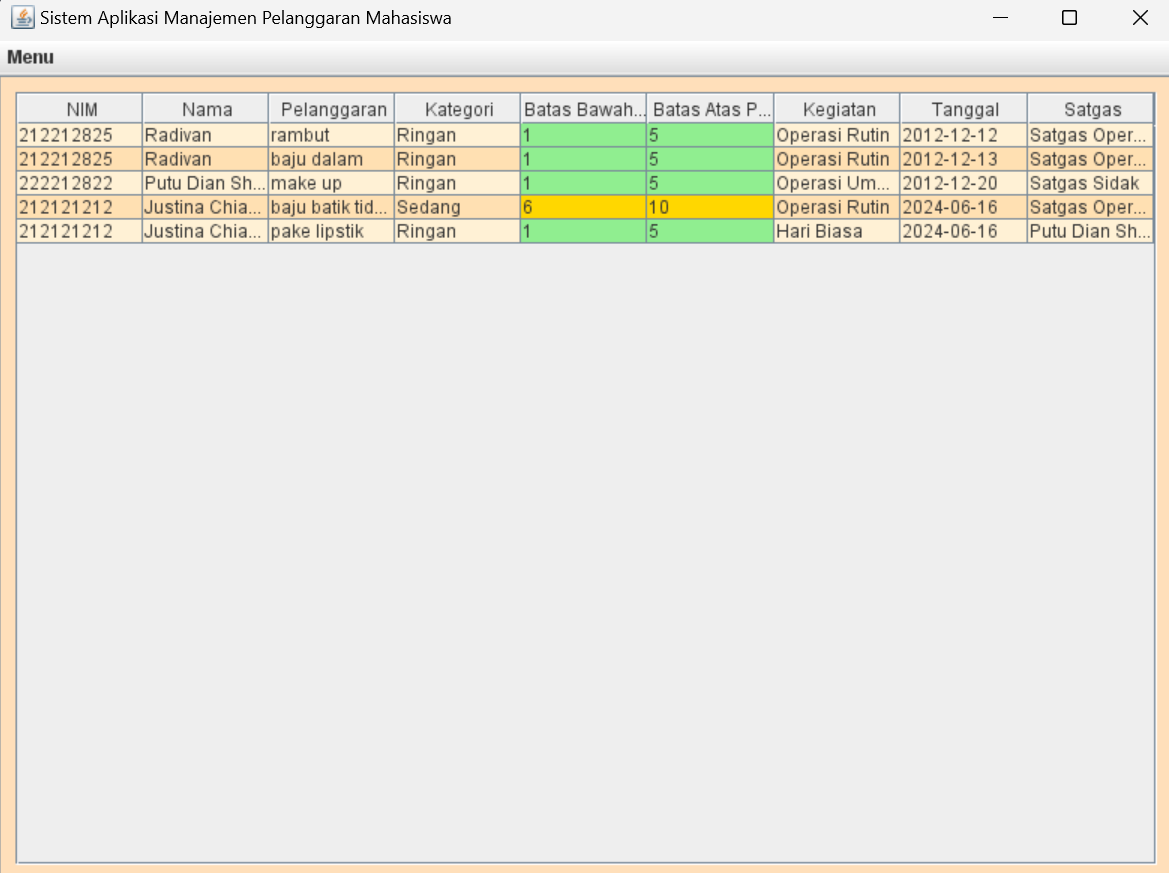
Gambar di atas merupakan tampilan apabila data berhasil tersimpan ke dalam database setelah mengklik tombol Simpan.

* CekPoinPanel.java



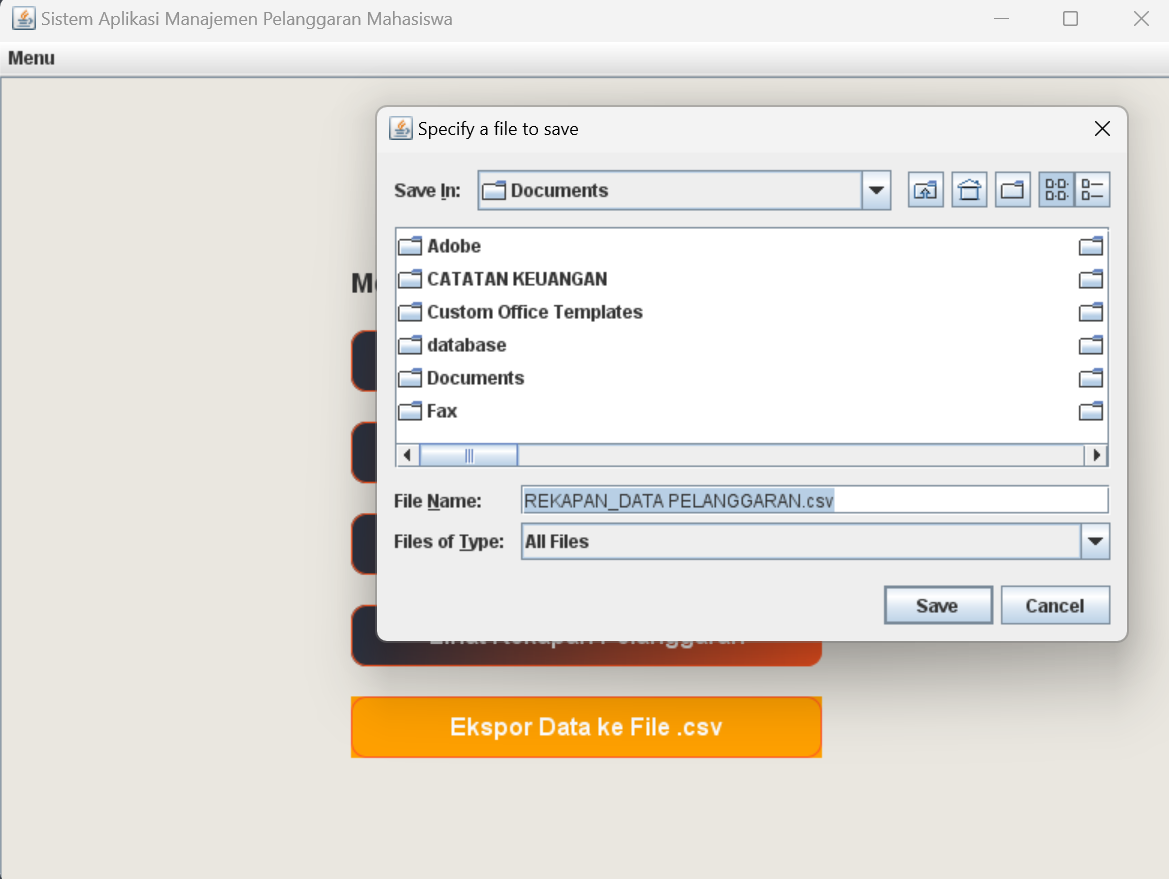
Panel ini akan muncul apabila admin mengklik Cek Total Poin pada HomePanel. Panel ini berfungsi untuk mengecek *range* perkiraan total poin mahasiswa berdasarkan NIM-nya. Total poin yang ditampilkan hanya berupa *range* perkiraan karena yang mengetahui poin fiksnya hanyalah BAAK, yang mana poin fiks ini diinformasikan BAAK kepada mahasiswa dalam jangka waktu tertentu, sehingga SPD hanya dapat menginput *range* perkiraan poin dari suatu pelanggaran berdasarkan kategori pelanggarannya. *Range* tersebut diketahui melalui Perdir. Pada *screenshot* di atas tertera contoh penggunaan panel ini di mana admin akan menginput NIM dari mahasiswa yang ingin diketahui *range* total poinnya lalu setelah mengklik tombol Cek, maka pada tabel akan muncul informasi pelanggaran dari mahasiswa tersebut.

* RekapanDataPanel.java

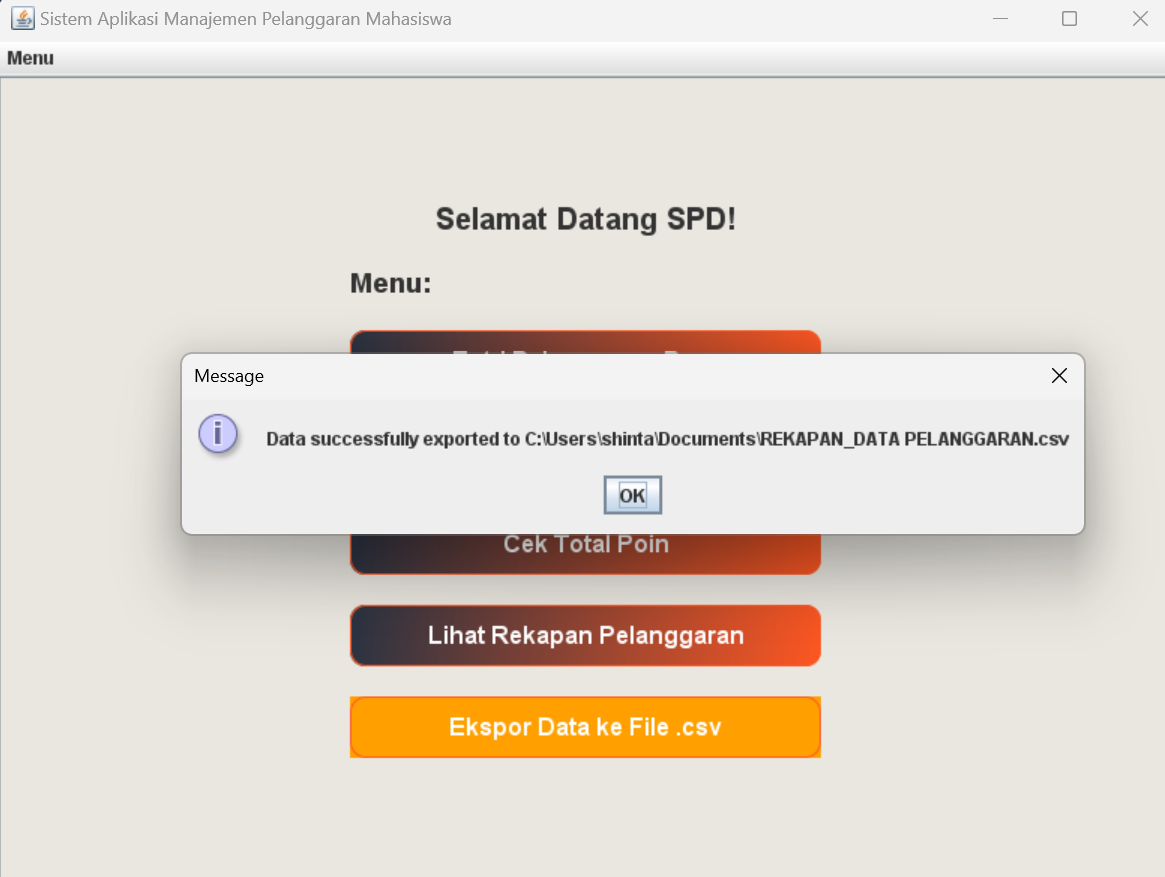


Panel ini akan muncul apabila admin mengklik Lihat Rekapan Pelanggaran pada HomePanel. Panel ini berfungsi untuk melihat pelanggaran-pelanggaran yang selama ini diinput oleh admin dalam bentuk tabel. Nilai-nilai pada kolom Batas Bawah Poin dan Batas Atas Poin diberi warna hijau untuk pelanggaran ringan, kuning untuk pelanggaran sedang, dan merah untuk pelanggaran berat.

* Tombol Ekspor Data ke File .csv

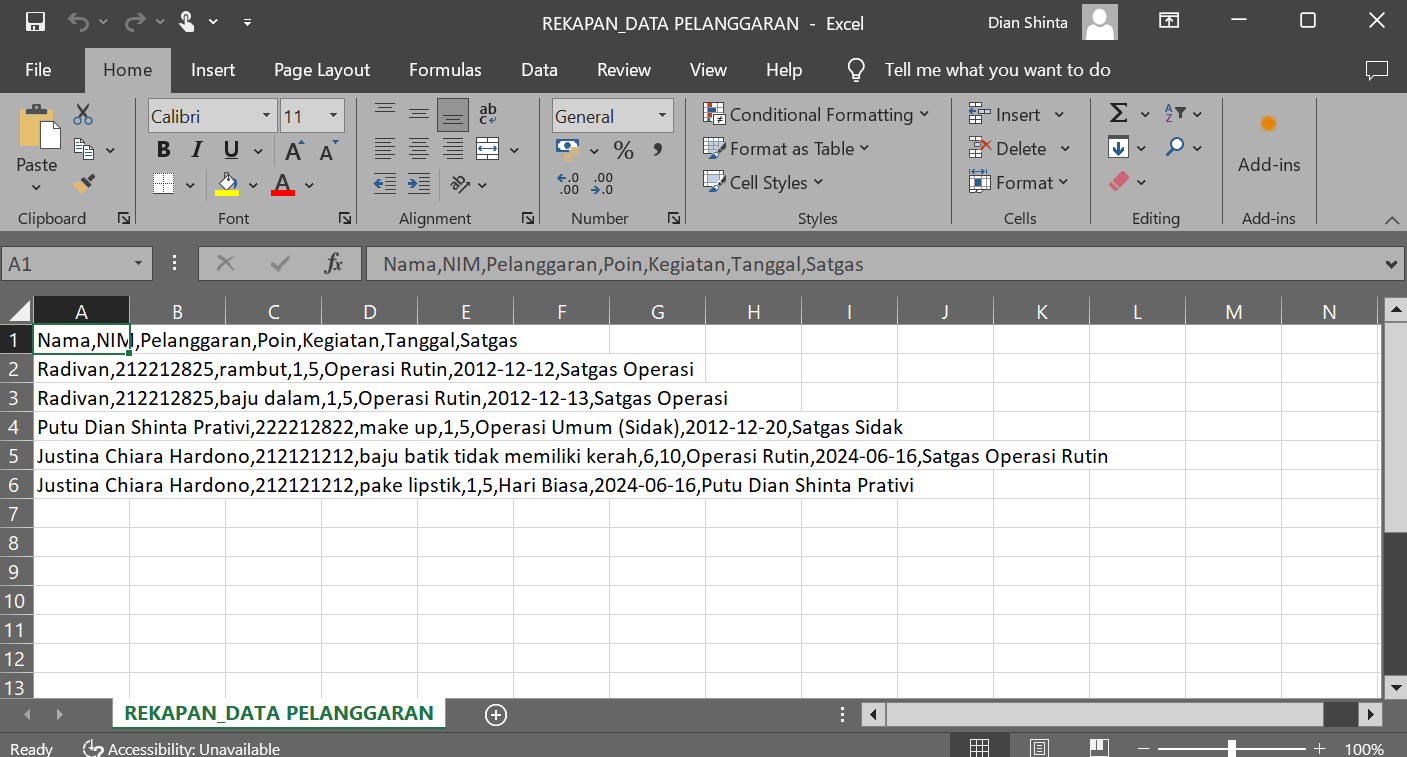


Tombol ini berguna untuk mengekspor data pada RekapanDataPanel tadi ke dalam fail berekstensi .csv yang secara default bernama REKAPAN\_DATA PELANGGARAN.csv.

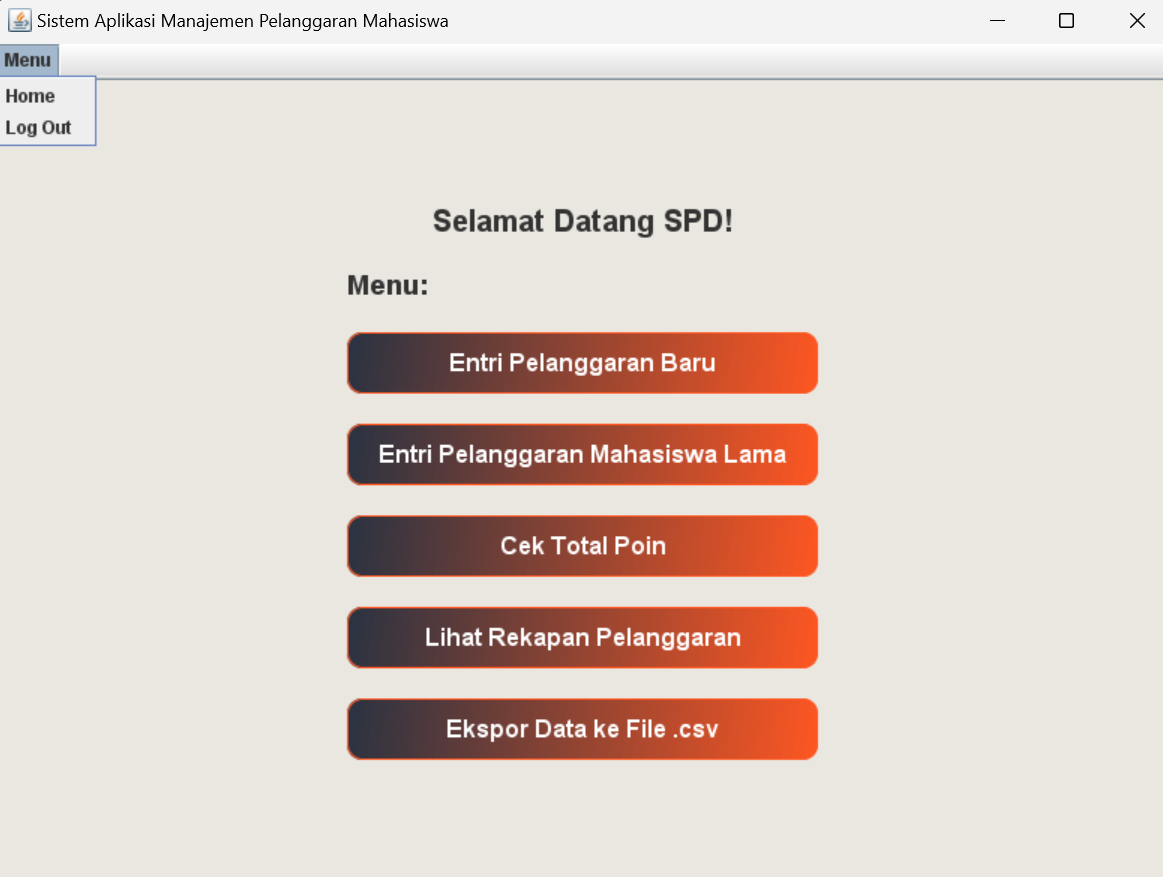


Apabila data berhasil diekspor, maka akan muncul pesan seperti gambar di atas.

Berikut fail REKAPAN\_DATA PELANGGARAN.csv hasil ekspor:



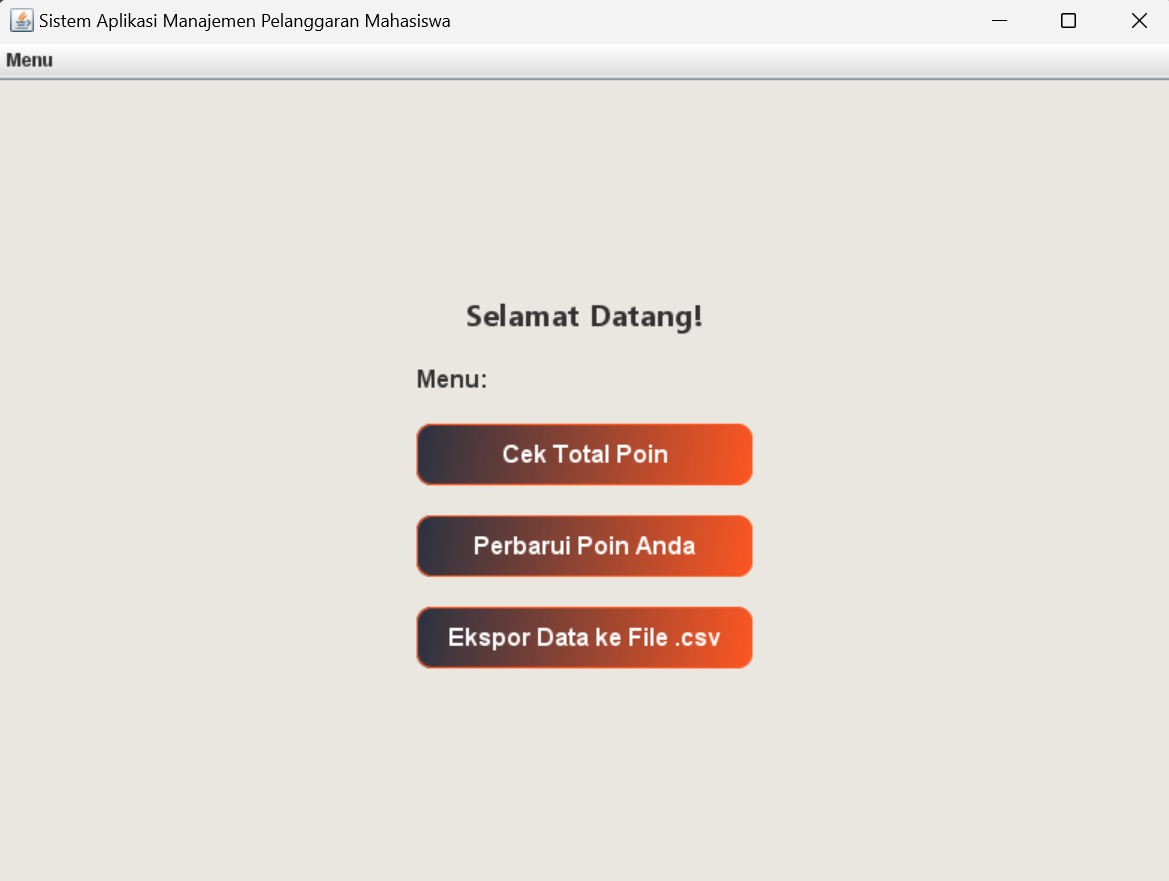
* MenuBar



MenuBar itu selalu muncul pada setiap panel kecuali LoginPanel. Home jika diklik akan muncul HomePanel, sedangkan Log Out jika diklik akan menghapus sesi admin dan akan kembali ke LoginPanel.

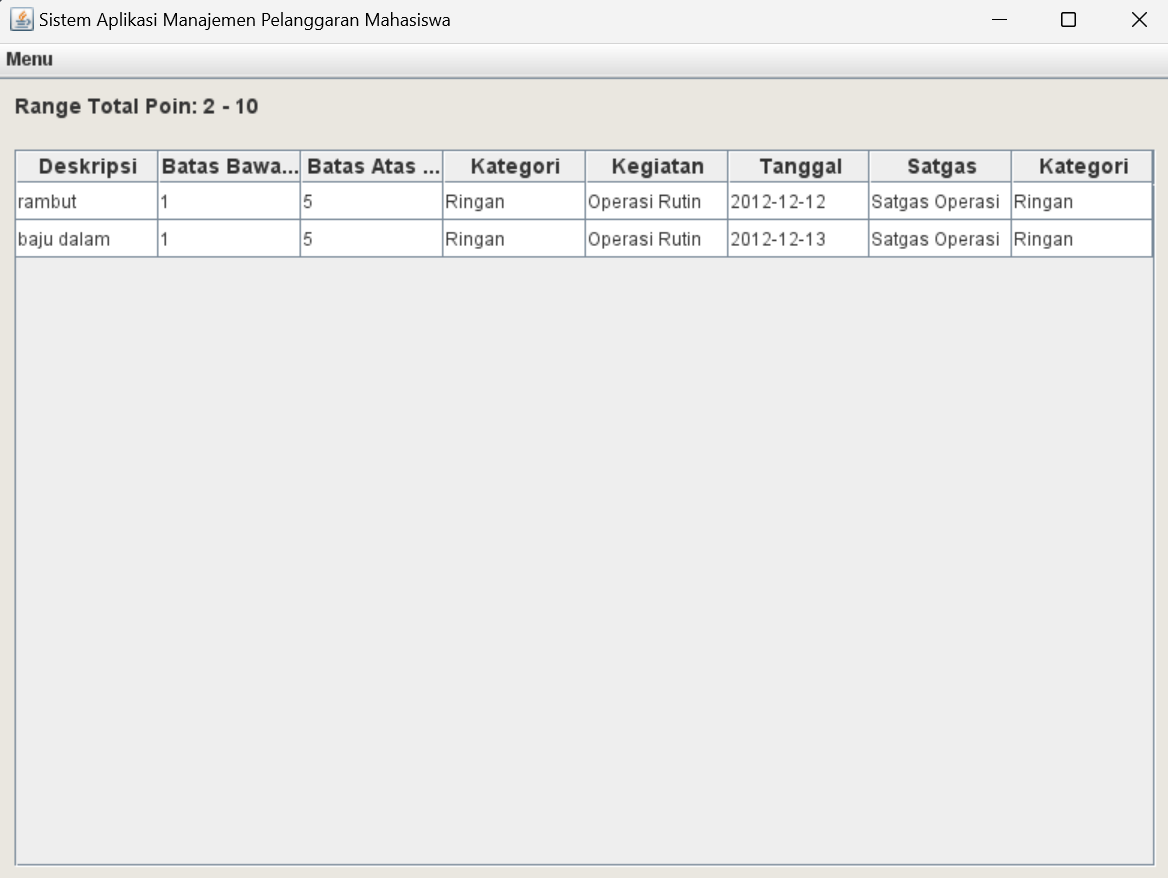
Berikut *screenshot* dari setiap panel untuk user beserta contoh penggunaannya:

* HomeMhsPanel.java



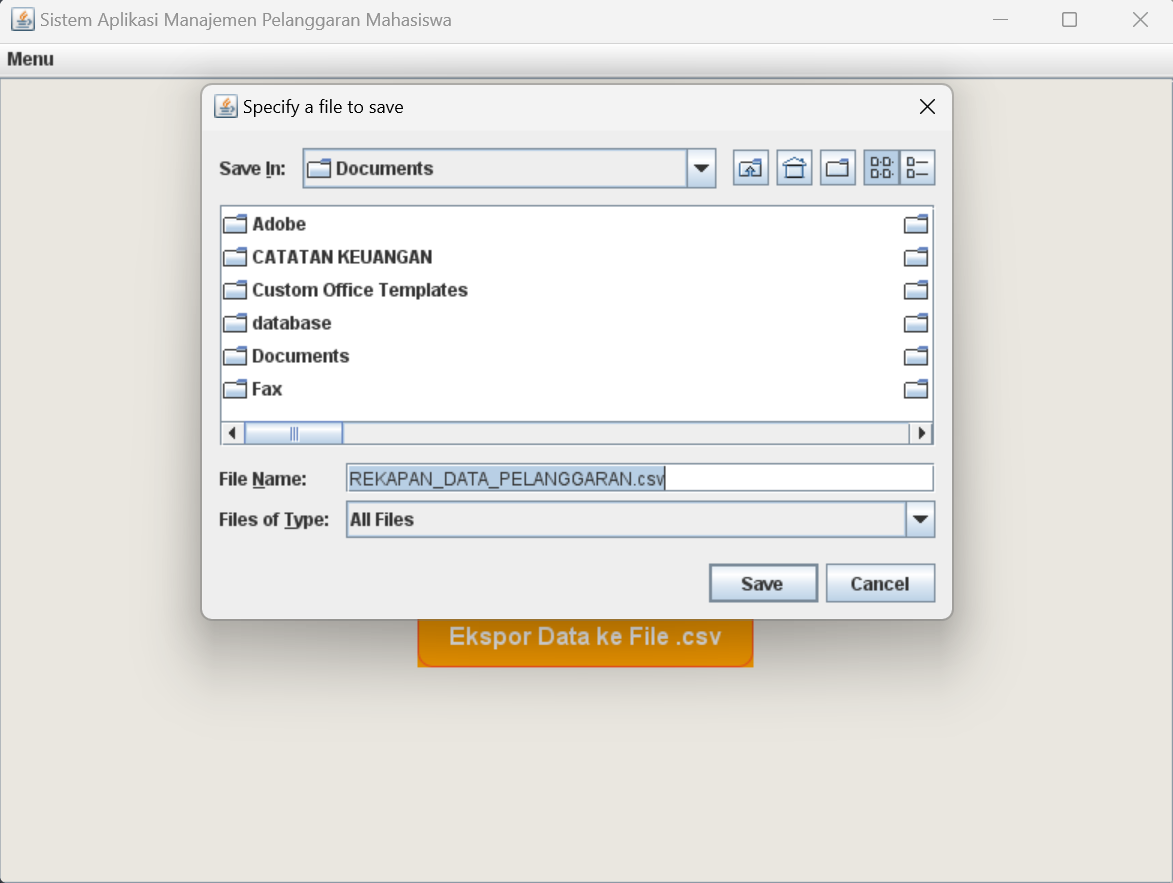
Panel ini akan muncul setelah user berhasil melakukan login. Pada demonstrasi panel untuk user ini, akan digunakan akun milik mahasiswa dengan NIM 212212825 bernama Radivan.

* CekPoinMhsPanel.java

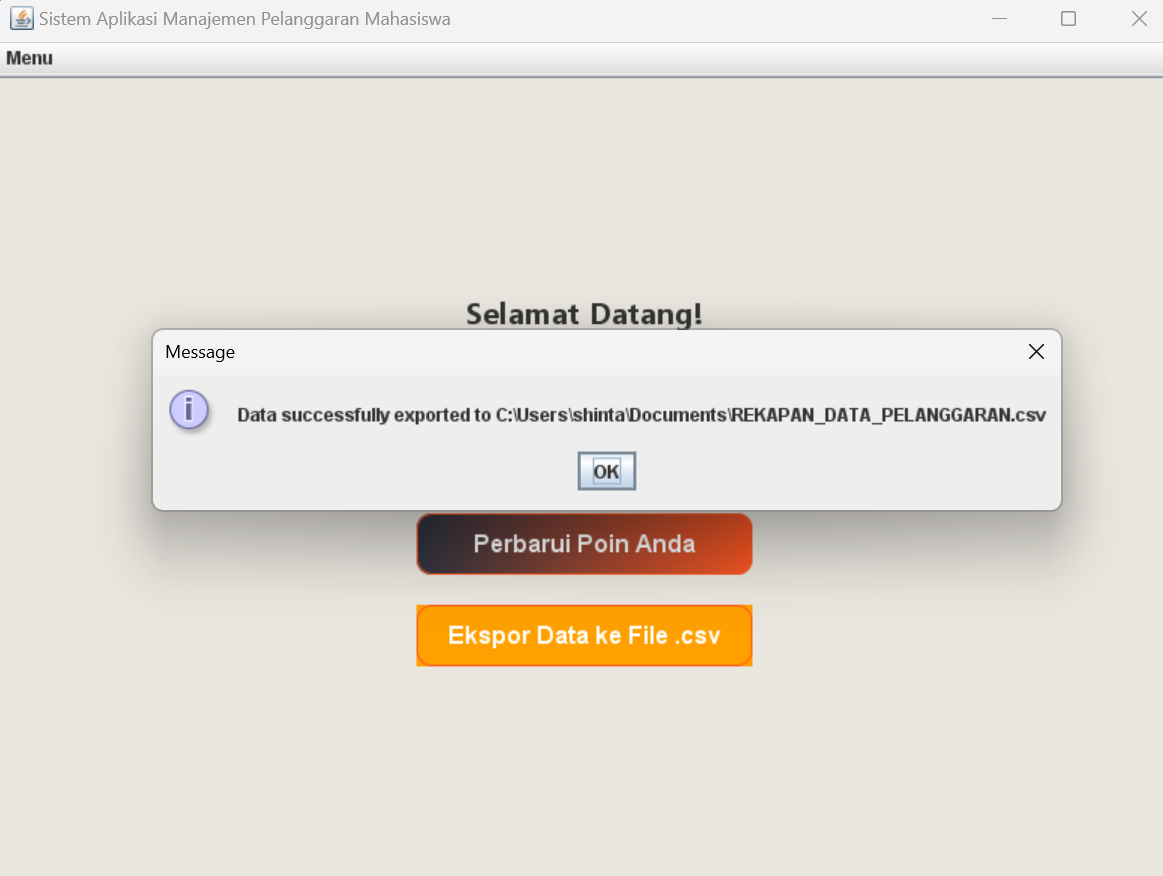


Panel ini akan muncul apabila user mengklik Cek Total Poin pada HomeMhsPanel. Pada panel ini, user dapat melihat *range* total poin beserta rincian pelanggaran miliknya yang telah diinput oleh admin.

* Tombol Ekspor Data ke File .csv

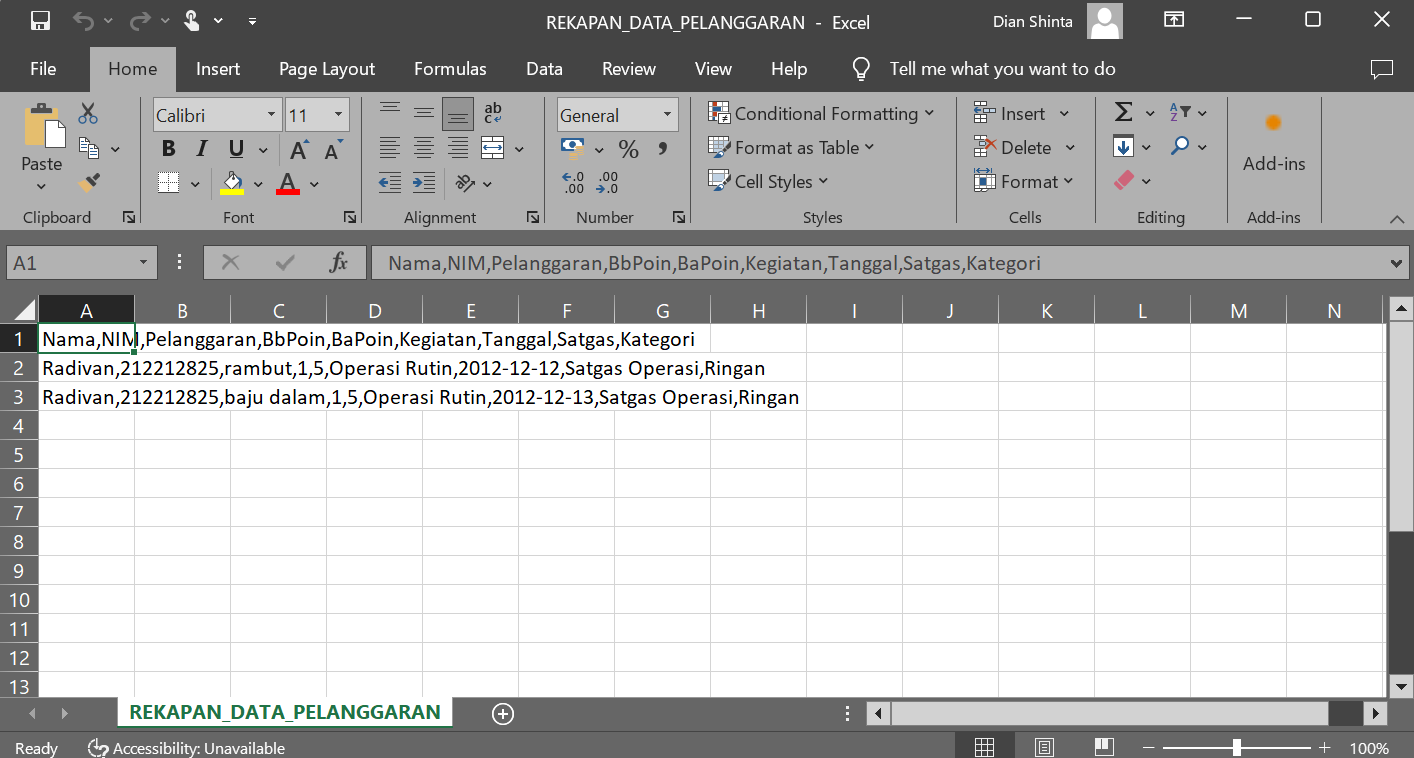


Tombol ini berguna untuk mengekspor data pelanggaran milik user yang sedang login ke dalam fail berekstensi .csv yang secara default bernama REKAPAN\_DATA\_PELANGGARAN.csv.



Apabila data berhasil diekespor, maka akan muncul pesan seperti gambar di atas.

Berikut fail REKAPAN\_DATA\_PELANGGARAN.csv hasil ekspor:



* MenuBar



MenuBar selalu muncul pada setiap panel kecuali LoginPanel. Home jika diklik akan muncul HomeMhsPanel, sedangkan Log Out jika diklik akan menghapus sesi user dan akan kembali ke LoginPanel.

1. Penerapan Konsep MVC

* Model

Berisi fail:

1. Mahasiswa.java
2. Pelanggaran.java

Kedua fail tersebut merupakan class yang merepresentasikan entitas data mahasiswa (user) dan pelanggaran yang digunakan dalam SAMPEL.

* View

Berisi fail:

1. MainFrame.java
2. LoginPanel.java
3. HomePanel.java
4. EntryPanel.java
5. EntryExistingPanel.java
6. CekPoinPanel.java
7. RekapanDataPanel.java
8. HomeMhsPanel.java
9. CekPoinMhsPanel.java
10. UpdateFixPoinPanel.java

Fail-fail pada View bertanggung jawab untuk tampilan GUI dari aplikasi SAMPEL, menerima input dari pengguna, dan menampilkan data kepada pengguna.

* Controller

Berisi fail:

1. Controller.java

Fail tersebut bertindak sebagai pengelola utama yang menghubungkan Model dan View. Controller ini menginisialisasi berbagai panel dan mengatur logika yang menghubungkan tindakan pengguna dengan perubahan data M.

1. Penerapan *Design Pattern*
2. ***Singleton Pattern***

Database.java:

Digunakan untuk memastikan hanya ada satu instance dari kelas Database yang dapat diakses di seluruh aplikasi. Penerapannya dalam SAMPEL yaitu metode getInstance() mengembalikan instance tunggal ini.

1. ***Observer Pattern***

Digunakan secara implisit dalam cara View mengamati perubahan pada Model melalui Controller. Penerapannya dalam SAMPEL yaitu ketika Controller memperbarui data pelanggaran, View diperbarui untuk mencerminkan perubahan data pelanggaran tersebut.

1. ***Strategy Pattern***

Controller.java:

Menggunakan *Strategy Pattern* untuk menentukan bagaimana input dari View diproses dan bagaimana data diperbarui. Penerapannya dalam SAMPEL yaitu Controller memiliki berbagai metode untuk menangani aksi seperti login, cek poin, tambah pelanggaran, dan lain-lain.

1. ***Composite Pattern***

JPanel dan JFrame:

GUI menggunakan Composite Pattern untuk mengatur berbagai komponen GUI menjadi antarmuka yang lebih kompleks. Penerapannya dalam SAMPEL yaitu MainFrame terdiri dari beberapa panel yang dapat diganti-ganti dalam contentScrollPane.

**BAB III**

**PENUTUP**

1. Kesimpulan

Aplikasi yang saya kembangkan yang bernama “SAMPEL”, merupakan aplikasi yang dapat memanajemen dan mencatat data-data pelanggaran yang dilakukan mahasiswa. Bagi admin, terdapat fitur-fitur untuk mengentri data pelanggaran mahasiswa, mengecek poin pelanggaran mahasiswa, melihat rekapan data pelanggaran yang telah diinput, dan mengekspor data tersebut ke dalam fail berekstensi .csv. Sedangkan bagi user yaitu mahasiswa selain anggota SPD divisi AOPP, terdapat fitur untuk mengecek poin, memperbarui poin fiks setelah terdapat pemberitahuan dari BAAK, dan mengekspor data pelanggaran user yang sedang login ke dalam fail berekstensi .csv. Aplikasi ini menerapkan konsep MVC dan berbagai *design patterns*.

1. Saran

Untuk meningkatkan efektivitas dan kenyamanan pengguna aplikasi manajemen pelanggaran mahasiswa, sangat disarankan untuk terus mengembangkan fitur-fitur baru serta memperbaiki bug berdasarkan masukan dari pengguna guna menyederhanakan dan mempermudah penggunaan aplikasi. Optimalkan performa aplikasi agar dapat diakses dengan cepat dan responsif di berbagai perangkat. Mengingat data yang disimpan mencakup informasi pribadi mahasiswa, keamanan data harus menjadi prioritas utama dengan mengimplementasikan mekanisme keamanan canggih seperti enkripsi dan hashing untuk melindungi informasi tersebut. Langkah-langkah ini akan membuat aplikasi lebih handal, aman, dan user-friendly.

**BAB IV**

**LAMPIRAN**

1. Tautan Video Demo Aplikasi

<https://drive.google.com/file/d/1wL6rZbEErsZadrAF2QTe3adZh4UxF62f/view?usp=sharing>

1. Tautan Kode Program dalam Repository Git

<https://git.stis.ac.id/shiinn97/aplikasisampel>