

**UAS
PEMROGRAMAN
BERORIENTASI OBJEK
+ LAB**



Oleh:

Martogi Jekson C. Siagian 2105541054

PROGRAM STUDI TEKNIK ELEKTRO

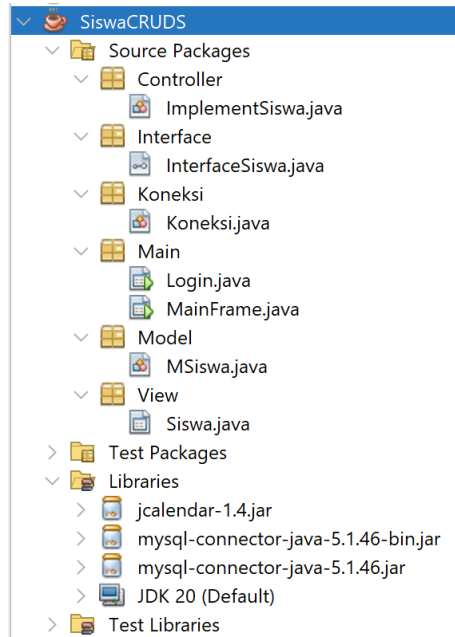
FAKULTAS TEKNIK

UNIVERSITAS UDAYANA

2023

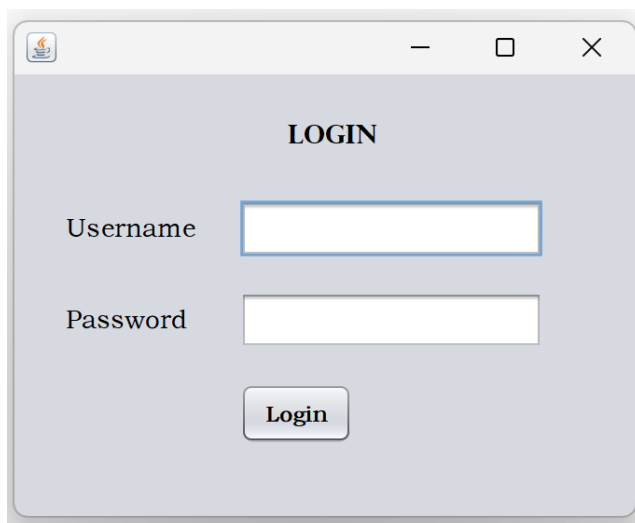
Aplikasi Pendaftaran Siswa Baru (CRUDS)

Pada Proyek Uas Pembuatan Aplikasi Pendaftaran Siswa baru ini saya membuat Proyek baru di NetBeans dengan nama SiswaCRUDS, Dimana saya membagi menjadi 6 Package yaitu Main, koneksi, Model, Controller, View, dan Interface.

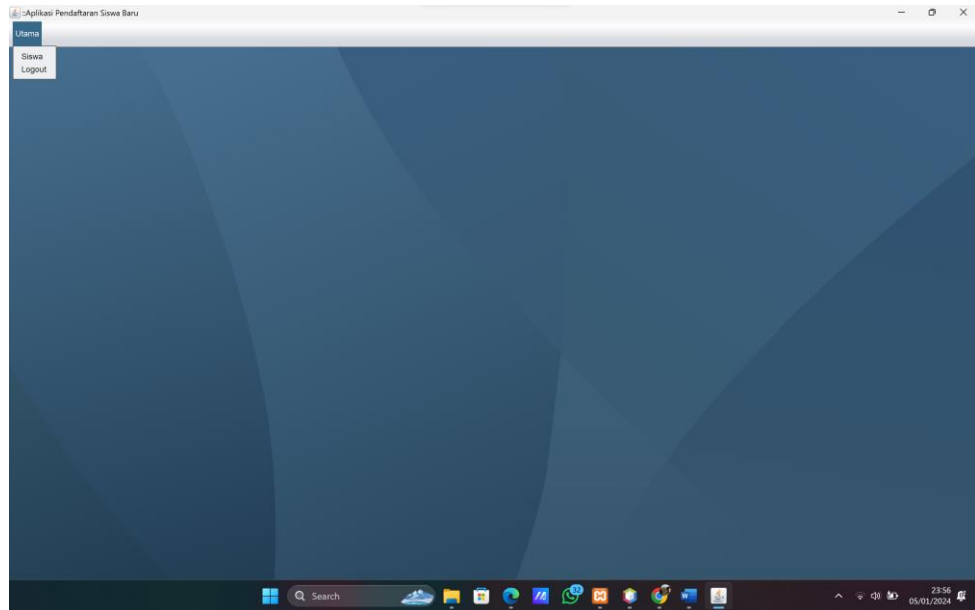


1. Main

Pada Package Main ini terdapat dua jform yaitu form Login.java dan form MainFrame.Java sebagai menu utama berikut tampilan nya

A screenshot of a Java Swing window titled 'LOGIN'. The window has a light gray background and a standard title bar with minimize, maximize, and close buttons. It contains two text input fields: one for 'Username' and one for 'Password'. Below the password field is a 'Login' button. The labels 'Username' and 'Password' are positioned to the left of their respective input fields.

Gambar 1. Form Login

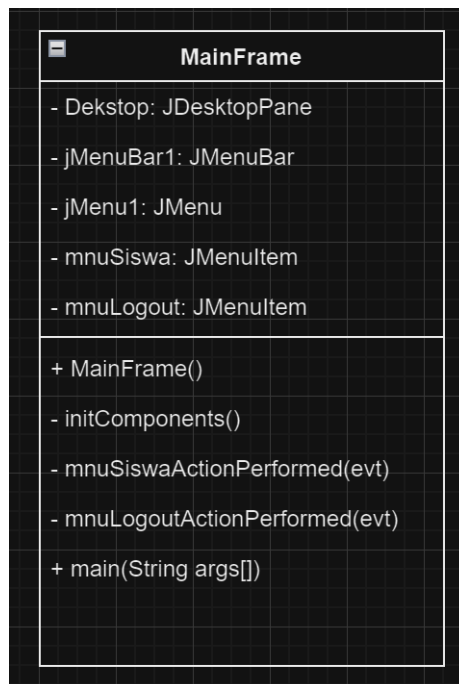


Gambar 2. Form Menu Utama

Pada Form login saya menambahkan table baru pada database saya sebagai berikut:

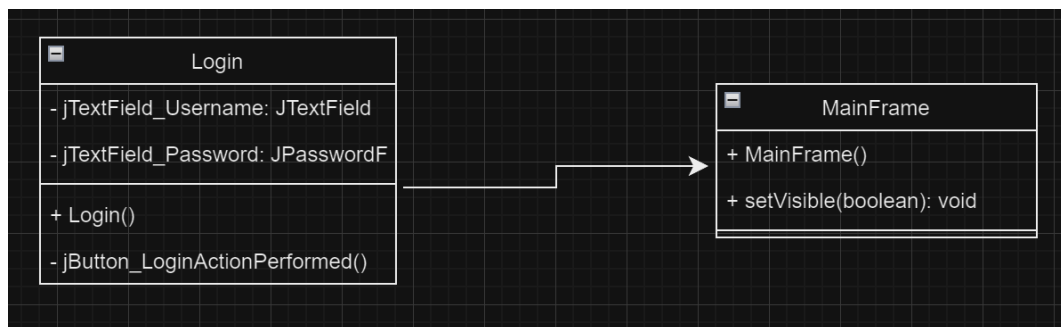
```
CREATE TABLE login (
  id INT AUTO_INCREMENT PRIMARY KEY,
  username VARCHAR(50) NOT NULL,
  password VARCHAR(100) NOT NULL
);
```

Class Diagram



Penjelasan:

- MainFrame: Kelas ini merupakan kelas utama yang mewakili frame utama dari aplikasi.
- MainFrame memiliki hubungan dengan Siswa. MainFrame menciptakan dan menambahkan instance Siswa ke dalam JDesktopPane.
- Atribut-atribut dan method-method yang dimiliki oleh kelas MainFrame ditunjukkan dengan simbol +.
- Hubungan antara MainFrame dan Siswa ditunjukkan dengan panah yang menunjukkan MainFrame menciptakan Siswa.
- Beberapa atribut dan method dihilangkan untuk tujuan kejelasan dan fokus pada struktur kelas.

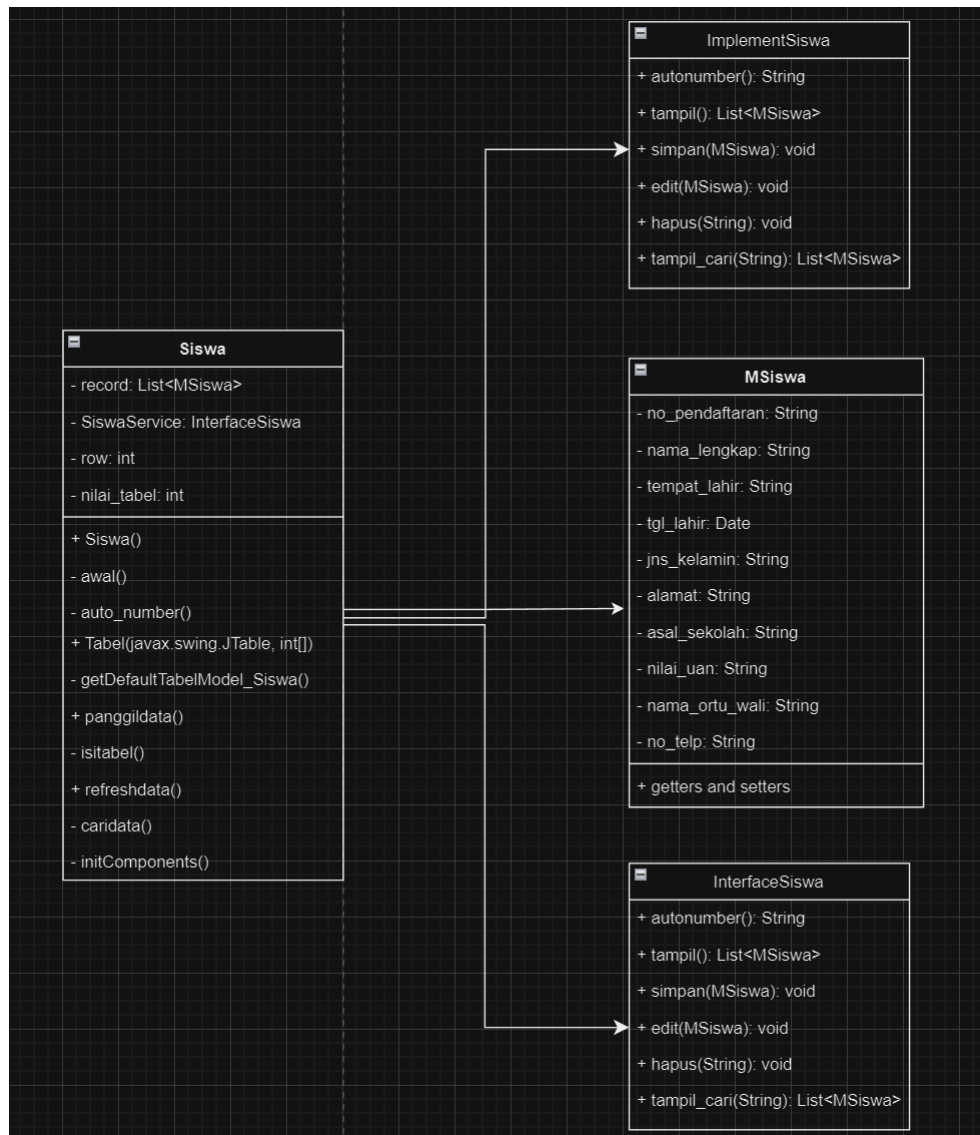


Penjelasan:

- Kelas Login memiliki dua atribut yang tidak terlihat secara eksplisit dalam kode, yaitu `jTextField_Username` (sebuah `JTextField`) dan `jTextField_Password` (sebuah `JPasswordField`).
- Kelas MainFrame tidak memiliki atribut yang terlihat dalam kode yang diberikan, tetapi menunjukkan adanya fungsi `setVisible(boolean)` yang digunakan untuk menampilkan atau menyembunyikan frame.
- Tidak ada keterkaitan (relasi) yang ditunjukkan dalam diagram ini. Ini adalah representasi yang sangat sederhana berdasarkan informasi yang disediakan dalam potongan kode.

2. View

Pada Package View terdapat satu form yaitu JinternalFrame Form dan berikut class diagramnya

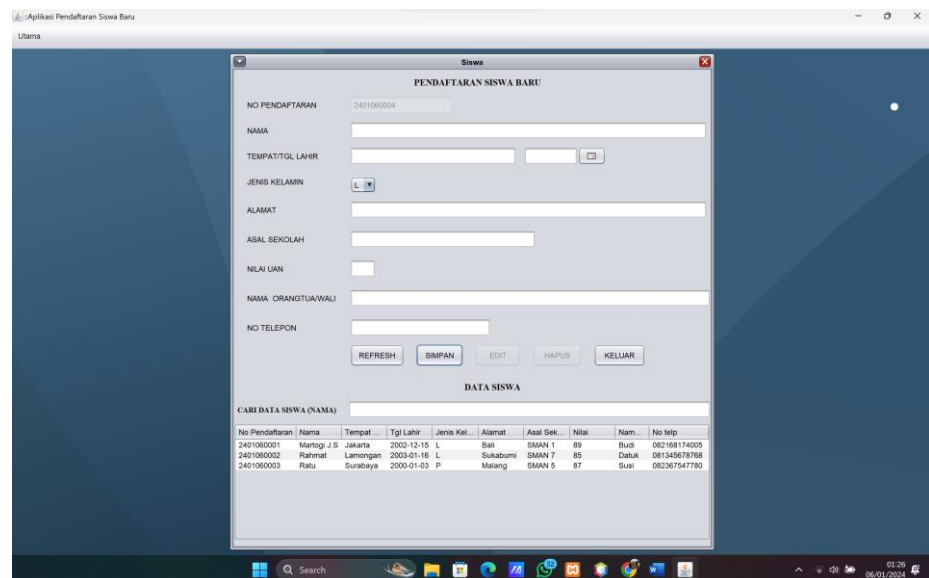


Penjelasan:

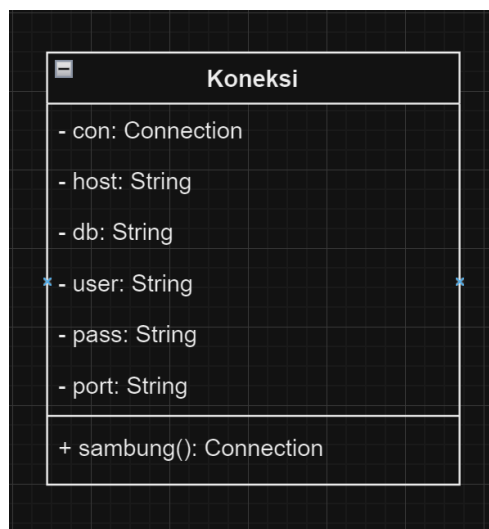
- Kelas Siswa adalah kelas utama yang berfungsi sebagai internal frame dalam antarmuka pengguna. Kelas ini memiliki atribut seperti `record`, `SiswaService`, `row`, dan `nilai_tabel` yang digunakan untuk mengelola data siswa, mengatur layanan siswa, dan mengontrol tampilan antarmuka.
- Kelas MSiswa mungkin merepresentasikan model data untuk entitas siswa. Kelas ini memiliki atribut-atribut seperti `no_pendaftaran`, `nama_lengkap`, `tgl_lahir`, dan lainnya yang merepresentasikan informasi tentang siswa.

- Interface `InterfaceSiswa` mendefinisikan kontrak atau blueprint untuk layanan yang berkaitan dengan data siswa, seperti pengambilan, penyimpanan, pengeditan, dan penghapusan data siswa.
- Kelas `ImplementSiswa` adalah implementasi dari `InterfaceSiswa` yang berisi logika untuk melakukan operasi-operasi terkait data siswa, seperti pengambilan dari database, penyimpanan, dan lainnya.

Berikut tampilan Menu Pendaftaran pada Package View dengan `InternalFrame` dengan nama `Form MSiswa`



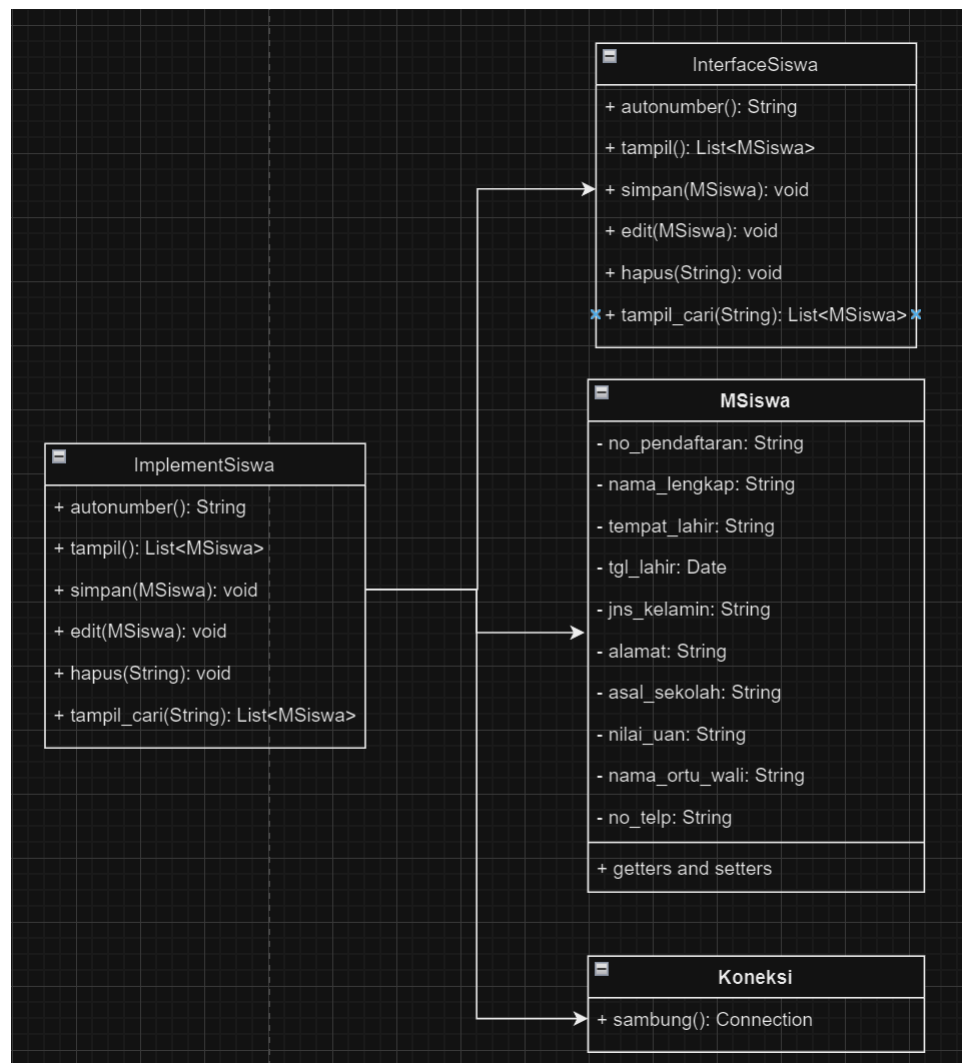
3. Koneksi



Penjelasan:

- Koneksi adalah kelas yang berfungsi sebagai kelas untuk mengelola koneksi ke database.
- con, host, db, user, pass, dan port adalah variabel kelas.
- sambung() adalah metode kelas yang mengembalikan objek Connection untuk koneksi ke database.

4. Controller



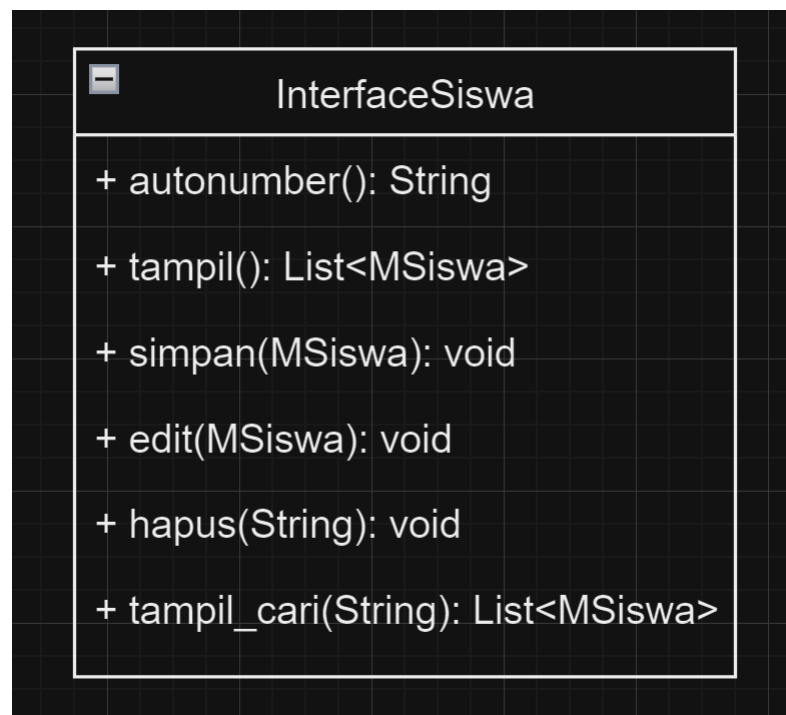
Pemjelasan:

- ImplementSiswa: Kelas yang mengimplementasikan InterfaceSiswa. Ini memiliki metode untuk menyimpan, mengedit, dan menghapus data siswa dari database, serta metode untuk

menampilkan data siswa dan mencari siswa berdasarkan kriteria tertentu.

- MSiswa: Kelas yang mewakili entitas siswa dengan atribut-atribut seperti nomor pendaftaran, nama lengkap, tempat lahir, tanggal lahir, jenis kelamin, alamat, dan lainnya. Kelas ini memiliki getter dan setter untuk atribut-atributnya.
- InterfaceSiswa: Sebuah antarmuka yang mendefinisikan operasi-operasi yang harus diimplementasikan oleh kelas lain yang ingin melakukan operasi pada data siswa, seperti menyimpan, mengedit, dan menghapus data, serta menampilkan data siswa.
- Koneksi: Kelas yang memfasilitasi koneksi ke database untuk menjalankan operasi-operasi database yang diperlukan oleh kelas-kelas lain dalam aplikasi.

5. Interface



Penjelasan Singkat:

- `simpan(MSiswa o): MSiswa`: Metode ini menerima objek dari kelas `MSiswa` dan kemungkinan melakukan operasi penyimpanan ke

dalam database atau media penyimpanan lainnya, mengembalikan objek MSiswa yang disimpan.

- `edit(MSiswa o): void`: Metode ini menerima objek MSiswa dan kemungkinan melakukan operasi pengeditan data siswa yang ada di dalam database atau media penyimpanan lainnya tanpa mengembalikan nilai.
- `hapus(String no_pendaftaran): void`: Metode ini menerima nomor pendaftaran siswa sebagai parameter dan kemungkinan melakukan operasi penghapusan data siswa dari database atau media penyimpanan lainnya tanpa mengembalikan nilai.
- `autonumber(): String`: Metode ini mungkin bertugas menghasilkan nomor pendaftaran siswa secara otomatis dan mengembalikan nomor unik sebagai string.
- `tampil(): List<MSiswa>`: Metode ini kemungkinan mengembalikan daftar seluruh data siswa yang ada dalam bentuk list objek MSiswa.
- `tampil_cari(String cari): List<MSiswa>`: Metode ini kemungkinan mengembalikan daftar data siswa yang sesuai dengan kriteria pencarian yang diberikan, misalnya, nama atau nomor pendaftaran, dalam bentuk list objek MSiswa.

6. Model

```

-----
|          MSiswa          |
-----
| - no_pendaftaran: String |
| - nama_lengkap: String  |
| - tempat_lahir: String  |
| - tgl_lahir: String      |
| - jns_kelamin: String    |
| - alamat: String         |
| - asal_sekolah: String   |
| - nilai_uan: String      |
| - nama_ortu_wali: String  |
| - no_telp: String        |
-----
| + getNo_pendaftaran(): String |
| + setNo_pendaftaran(String): void |
| + getNama_lengkap(): String |
| + setNama_lengkap(String): void |
| + getTempat_lahir(): String |
| + setTempat_lahir(String): void |
| + getTgl_lahir(): String |
| + setTgl_lahir(String): void |
| + getJns_kelamin(): String |
| + setJns_kelamin(String): void |
| + getAlamat(): String |
| + setAlamat(String): void |
| + getAsal_sekolah(): String |
| + setAsal_sekolah(String): void |
| + getNilai_uan(): String |
| + setNilai_uan(String): void |
| + getNama_ortu_wali(): String |
| + setNama_ortu_wali(String): void |
| + getNo_telp(): String |
| + setNo_telp(String): void |
-----

```

Penjelasan:

- no_pendaftaran: Menyimpan nomor pendaftaran siswa.
- nama_lengkap: Menyimpan nama lengkap siswa.
- tempat_lahir: Menyimpan tempat lahir siswa.
- tgl_lahir: Menyimpan tanggal lahir siswa.
- jns_kelamin: Menyimpan jenis kelamin siswa.
- alamat: Menyimpan alamat siswa.
- asal_sekolah: Menyimpan nama sekolah asal siswa.
- nilai_uan: Menyimpan nilai Ujian Akhir Nasional (UAN) siswa.
- nama_ortu_wali: Menyimpan nama orang tua atau wali siswa.
- no_telp: Menyimpan nomor telepon siswa atau orang tua/wali.
- Metode get dan set, Metode-metode tersebut mengimplementasikan prinsip-prinsip akses pada atribut-atribut kelas MSiswa untuk mendapatkan nilai dan mengatur nilai dari setiap atributnya.