BAB 3

ANALISIS MASALAH DAN PERANCANGAN

3.1. Analisis masalah

Analisis masalah merupakan proses memilah-milah suatu permasalahan menjadi elemen-elemen yang lebih kecil untuk dipelajari guna mempermudah permasalahan dari suatu sistem informasi. Hasil akhir dari analisis sistem merupakan cara pemecahan masalah yang terjadi dalam spesifikasi sistem baru.

3.1.1. Analisis sistem yang sedang berjalan

Pada bagian ini akan dibahas bagaimana prosedur dan aliran data yang sedang berjalan dari satu proses ke proses lain yang digambarkan dalam bentuk *activity diagram*. Dari hasil obeservasi dan wawancara pada pihak SMA Negeri 2 Purwakarta, terdapat beberapa proses yang masih dijalankan secara manual, diantaranya adalah :

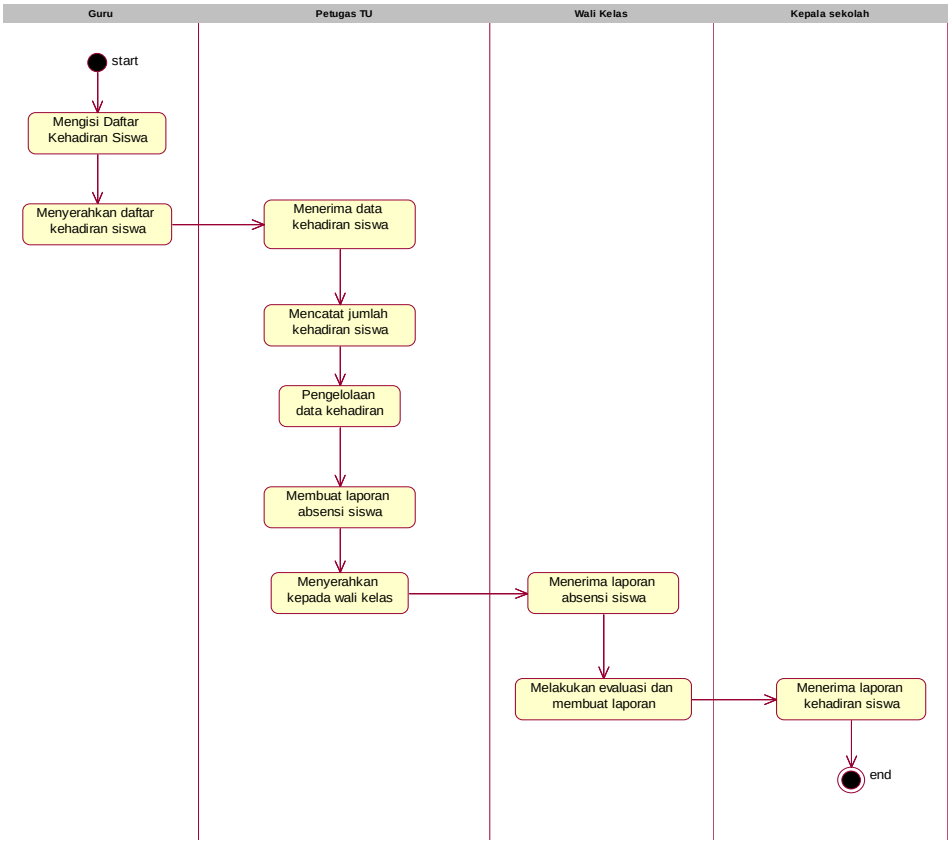
1. Proses absensi siswa
2. Proses nilai siswa
3. Proses pembayaran dan pelaporan iuran sekolah
4. Proses pemilihan beasiswa

3.1.1.1.Proses Absensi Siswa

Proses pengolahan data absensi siswa di SMA Negeri 2 Purwakarta saat ini adalah sebagai berikut :

1. Masing-masing guru melakukan absen setiap melaksanakan kegiatan belajar mengajar.
2. Data kehadiran tersebut diserahkan pada petugas TU.
3. Petugas TU menerima data kehadiran dari setiap kelas.
4. Petugas TU melakukan pencatatan jumlah kehadiran siswa dan keterangan lainnya.
5. Setelah itu, dibuatkan sebuah laporan kehadiran siswa untuk diserahkan pada wali kelas
6. Wali kelas melakukan evaluasi dan melaporkan kepada kepala sekolah.

Untuk lebih jelas, proses absensi dituangkan dalam suatu *activity diagram*.



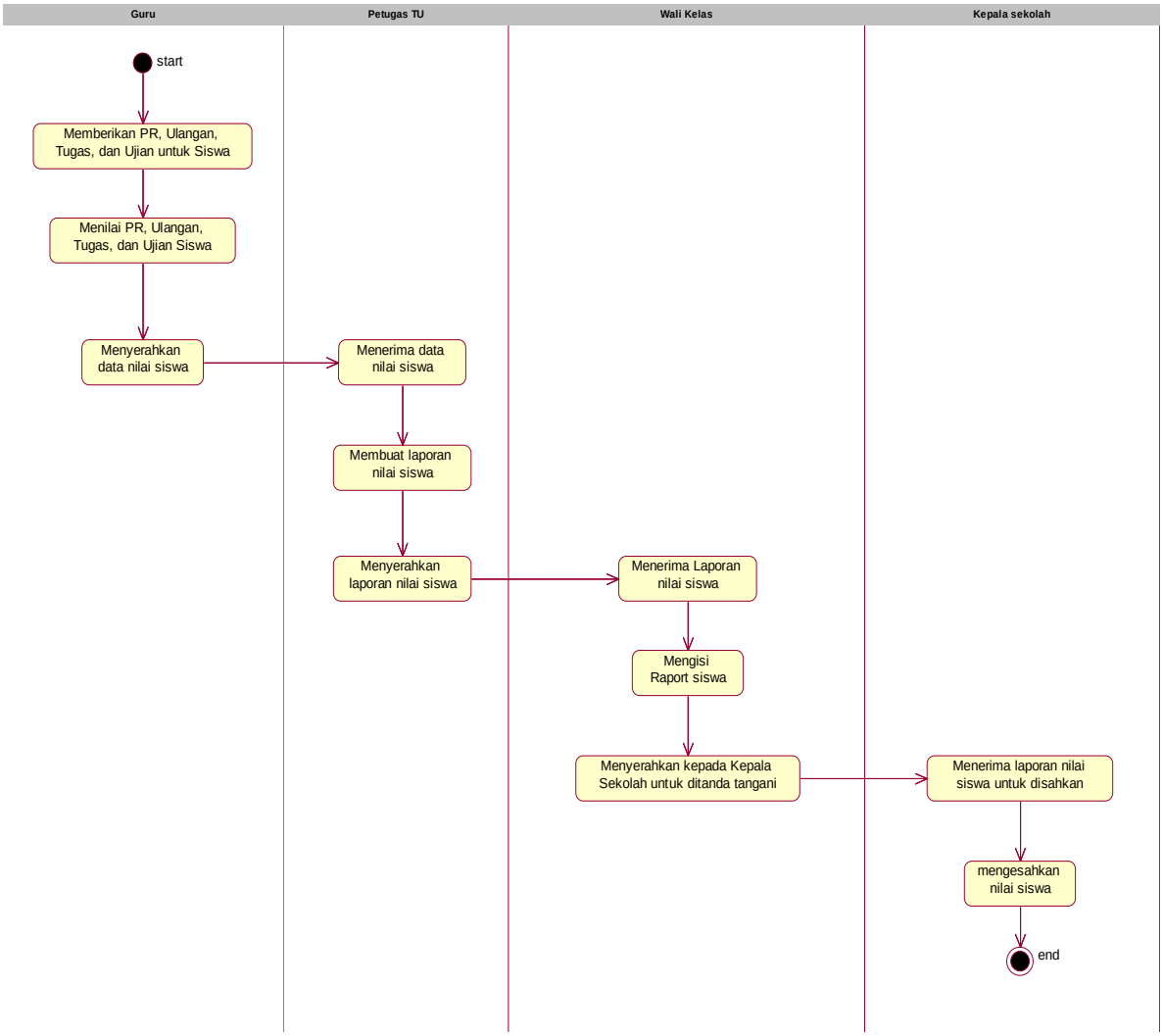
Gambar 3. 1 : Proses Absensi Siswa

3.1.1.2. Proses Pengolahan Nilai Siswa

Proses sistem pengolahan nilai siswa pada SMA Negeri 2 Purwakarta saat ini adalah sebagai bertikut :

1. Guru memberikan PR / tugas, ulangan dan ujian dan menilainya.
2. Hasil penilaian tersebut diserahkan kepada petugas TU.
3. Kemudian nilai tersebut dibuat laporan nilai siswa dan diserahkan kepada wali kelas.
4. Wali kelas menerima laporan nilai siswa.
5. Kemudian, wali kelas mengisi raport siswa dan menyerahkan kepada kepala sekolah untuk ditandatangani.
6. Kepala sekolah menerima laporan nilai siswa dalam bentuk buku raport, kemudian menandatangai sebagai bentuk pengesahan.

Untuk lebih jelasnya, proses pengolahan nilai dituangkan dalam bentuk *activity diagram*.



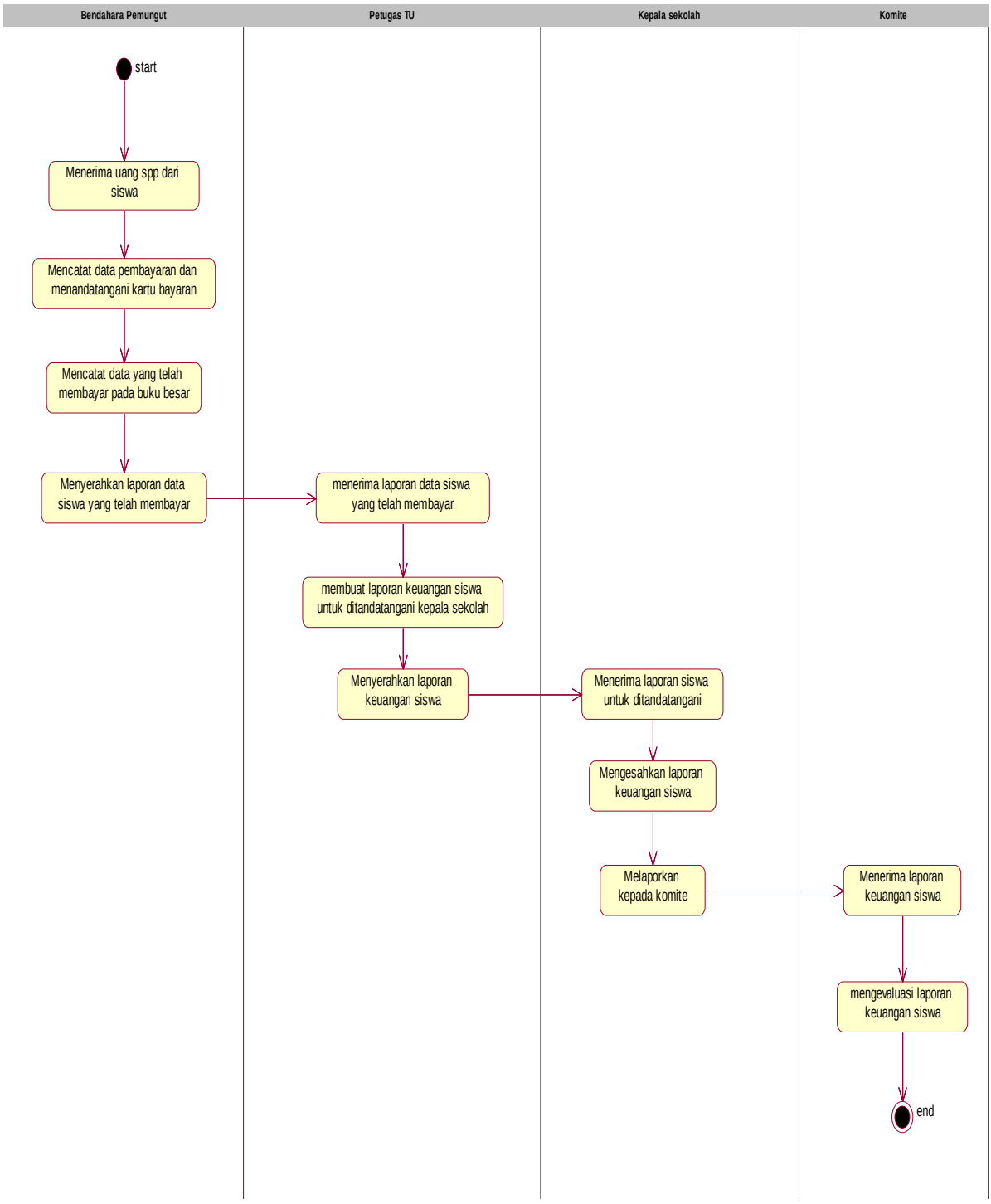
Gambar 3. 2 : Proses Pengolahan Nilai Siswa

3.1.1.3. Proses Pembayaran dan Pelaporan Iuran Sekolah

Proses sistem pembayaran iuran sekolah pada SMA Negeri 2 Purwakarta saat ini adalah sebagai bertikut :

1. Bendahara Pemungut menerima uang spp dari siswa.
2. Kemudian mencatat data pembayaran dan menandatangani kartu bayaran siswa
3. Selanjutnya, bendahara pemungut mencatat data siswa yang sudah membayar pada buku besar.
4. Menyerahkan data siswa yang telah membayar kepada petugas TU.
5. Petugas TU membuat laporan siswa yang telah membayar, dan menyerahkan kepada kepala sekolah untuk ditandatangani.
6. Kepala sekolah menerima laporan keuangan siswa.
7. Kemudian kepala sekolah menandatangani sebagai bentuk pengesahan.
8. Kemudian laporan tersebut dilaporkan kepada komite sekolah untuk dievaluasi.

Untuk lebih jelasnya, proses pembayaran dan pelaporan iuran siswa dituangkan dalam *activity diagram*.



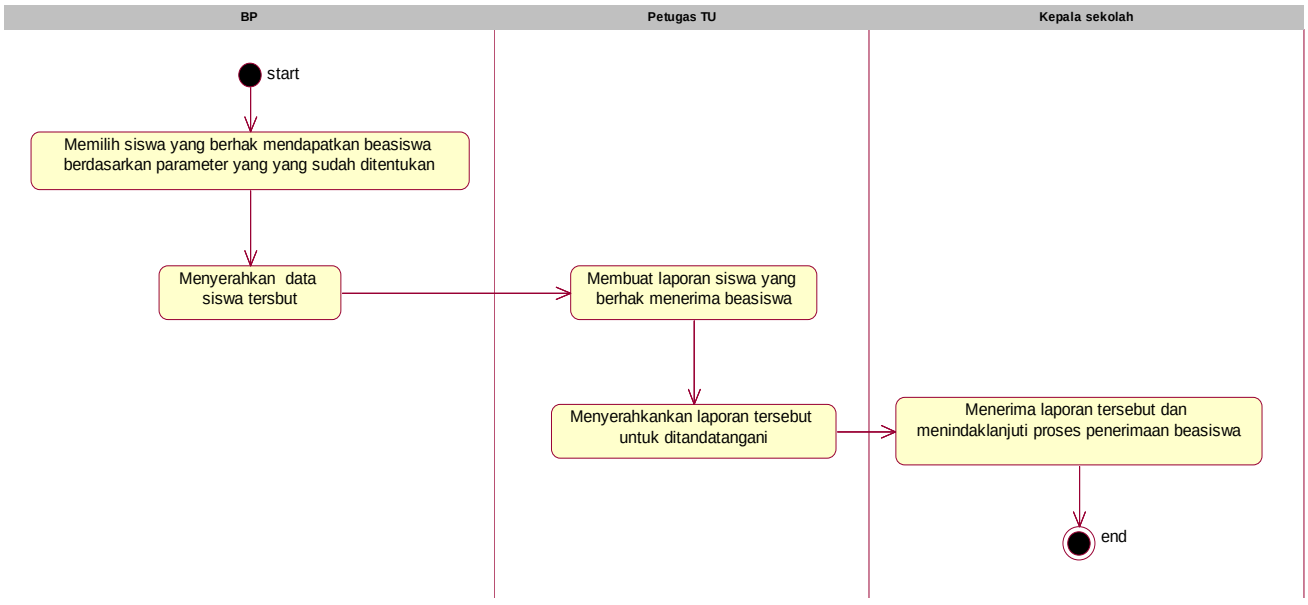
Gambar 3. 3 : Proses Pembayaran Dan Pelaporan Iuran Sekolah

3.1.1.4. Proses Pemilihan Beasiswa

Proses penentuan yang berhak menerima beasiswa di SMA Negeri 2 Purwakarta saat ini adalah sebagai berikut :

1. Yang menentukan siapa yang berhak menerima beasiswa adalah bagian BP. Jadi BP memilih siswa yang berhak menerima beasiswa berdasarkan parameter-parameter tertentu.
2. Kemudian data siswa siswa yang berhak tersebut diserahkan kepada petugas TU untuk dibuat laporan.
3. Kemudian laporan tersebut diserahkan kepada kepala sekolah untuk ditindak lanjuti.

Untuk lebih jelasnya, proses pemilihan beasiswa dituangkan dalam activity diagram.



Gambar 3. 4 : Proses Pemilihan Beasiswa

3.1.2. Analisis kebutuhan *non fungsional*

Analisis kebutuhan *non fungsional* merupakan analisis kebutuhan di luar fungsi dari sistem. Analisis ini terdiri dari analisis *hardware* (perangkat keras), analisis *software* (perangkat lunak), analisis *user* (pengguna), dan analisis jaringan.

3.1.2.1. Analisis Hardware (Perangkat Keras)

Hardware (Perangkat Keras) adalah seluruh komponen atau unsur peralatan yang digunakan untuk menunjang pembangunan suatu sistem.

Berikut ini merupakan spesifikasi perangkat keras yang digunakan dalam membangun aplikasi ini (minimal) :

1. Procesor Intel Pentium 4
2. RAM 1 GB
3. Hard disk 80 GB
4. Monitor
5. Keyboard dan mouse

3.1.2.2. Analisis Software (Perangkat Lunak)

Software (Perangkat Lunak) adalah komponen dalam pengolahan data yang berupa perangkat lunak. Perangkat lunak untuk mengolah data di SMA Negeri 2 Purwakarta saat ini masih menggunakan perangkat lunak yang umum seperti :

a. Sistem Operasi : *Windows XP*

b. Aplikasi pendukung : *Microsoft Office*

Untuk membangun sistem ini maka dibutuhkan spesifikasi minimum perangkat lunak yang harus ada, diantaranya :

a. Sistem operasi : *Microsoft Windows XP*

b. Perangkat pendukung :

1. MySQL

2. Neatbean

3. Rational Rose

4. Microsoft Visio 2003

3.1.2.3. Analisis *User* (Pengguna)

Analisis *user* (pengguna) dimaksudkan untuk mengetahui siapa saja user atau pengguna yang terlibat dalam pengolahan data beserta karakteristiknya sehingga dapat diketahui tingkat pemahaman user terhadap computer.

1. Guru

Education	: S1 / S2
Sistem Operasi yang digunakan	: *Windows XP*
Software	: *Microsoft Word dan Excel*
Knowledge & Experience	: *Medium* (menengah)

Frequency of use : *Medium* (menengah) *to high* (tinggi)

Job Descriction : Memonitoring semua laporan akademik

2. Petugas Tata Usaha / Admin

Education : D3 / S1

Sistem Operasi yang digunakan : *Windows XP*

Software : *Microsoft Word dan Excel*

Knowledge & Experience : *Medium* (menengah)

Frequency of use : *Medium* (menengah) *to high* (tinggi)

Job Descriction : Mengolah seluruh data akademik yang ada di SMA Negeri 2 Purwakarta. Seperti pengelolaan data kehadiran siswa, data tagihan siswa, beasiswa, dan juga nilai siswa. Juga menjadi Administrator pada perangkat lunak yang akan dibangun.

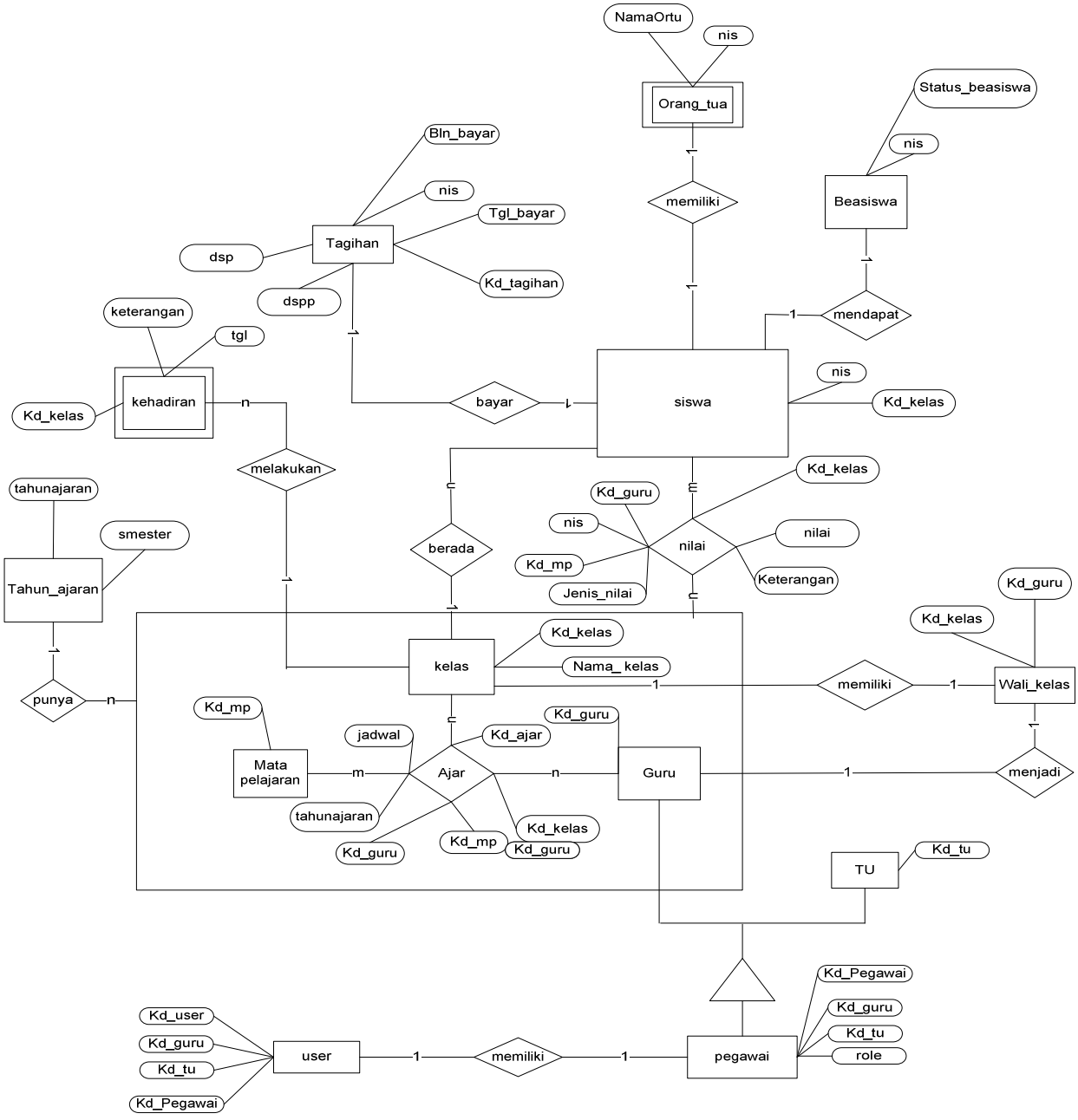
Berdasarkan hasil analisis terhadap user yang akan menggunakan aplikasi, setiap user yang ada di SMA Negeri 2 Purwakarta pada umumnya sudah bisa mengoperasikan komputer, tapi belum memahami benar tentang komputer. Dari data keseluruhan dapat dilihat bahwa setiap user dapat mengoperasikan program aplikasi office seperti

Microsoft Word dan Microsoft Excel. Tetapi masih perlu di berikan pelatihan untuk mengoptimalkan penggunaan aplikasi yang akan dibangun.

3.1.3. Analisis Basis Data

Analisis basis data merupakan kumpulan data yang saling berkaitan satu dengan yang lain yang direalisasikan dengan relasi kunci yang digambarkan dalam *entity relationship diagram* (ERD)

Adapun ERD yang terbentuk adalah :



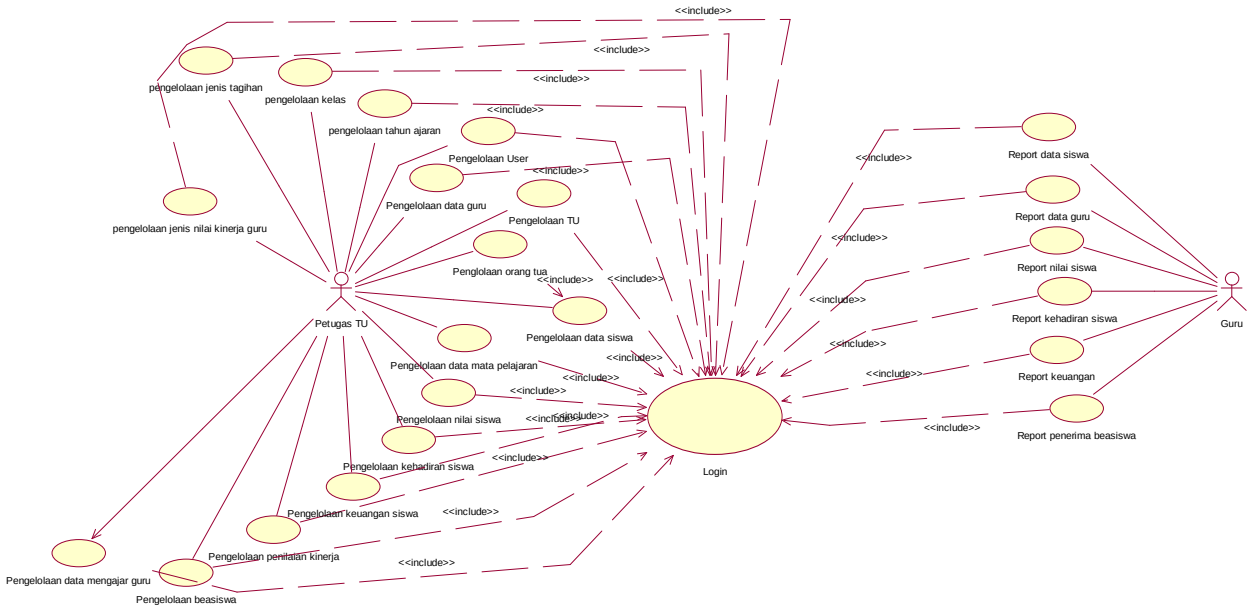
Gambar 3. 5 : ER Diagram

3.1.4. Analisis Kebutuhan *Fungsionalitas*

Analisis kebutuhan *fungsionalitas* suatu sistem bisa digambarkan atau dijelaskan dengan cara membuat diagram UML terlebih dahulu, yang meliputi *Use Case Diagram, sequence Diagram, Activity Diagram* dan *Class Diagram.*

3.1.4.1. Use Case Diagram

Use Case Diagram menggambarkan *fungsionalitas* yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya *login* ke sistem, meng-*create* sebuah daftar belanja, dan sebagaimanya. Seorang / sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.



Gambar 3. 6 : *Use Case Diagram*

Skenario *use case*

Nomor	: 01
Nama use case	: *Login*
Aktor	: Petugas TU, Kepala Sekolah, Wali Kelas
Tujuan	: melakukan *verifikasi* terhahap *user* yang akan menggunakan aplikasi
Deskripsi	: *user* memasukkan *usename* dan *password* sebelum masuk ke halaman utama

Kondisi awal : Halaman *login*	
Kondisi akhir : Halaman utama	
Aktor	Sistem
1. Memasukkan *user name* dan *password* kemudian menekan *button* OK	2. Sistem memeriksa *userename* dan *password* yang di *input* kan oleh user menampilkan form input nilai siswa 3. Jika *username* dan *password* terdaftar, maka secara otomatis akan masuk ke menu utama, jika tidak maka sistem akan kembali ke halaman *login*.

Nomor : 02	
Nama use case : Pengelolaan Tahun Ajaran	
Aktor : Petugas TU	
Tujuan : mencatat, menghapus, dan meng*update* data tahun ajaran	
Deskripsi : 1. Mencatat data tahun ajaran kedalam data base 2. Mencari data tahun ajaran, kemudian dapat menghapus data tersebut atau meng*update*nya	
Proses mencatat tahun ajaran baru	
Kondisi awal : Halaman utama, menu tahun ajaran, form input tahun ajaran	
Kondisi akhir : Tahun ajaran yang baru tersimpan di *database*	
Aktor	Sistem
1. Petugas TU masuk ke menu tahun ajaran, lalu ke form *input* tahun ajaran 2. Petugas TU memasukkan data tahun ajaran yang baru	3. Sistem menampilkan form *input* tersebut 4. Sistem menyimpan data tersebut kedalam database 5. Jika user salah dalam memasukkan

	data, maka akan muncul kotak dialog peringatan.
Proses mencari tahun ajaran baru	
Kondisi awal : Halaman utama, menu tahun ajaran, form cari tahun ajaran	
Kondisi akhir : Menampilkan tahun ajaran yang dicari *user*	
Aktor	Sistem
1. Petugas TU masuk ke menu tahun ajaran, lalu ke form cari tahun ajaran 2. Petugas TU memasukkan data tahun ajaran yang akan dicari	3. Sistem menampilkan form cari tersebut 4. Sistem menampilkan tahun ajaran tersebut 5. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan data yang di cari tidak ditemukan.

Nomor	: 03
Nama use case	: Pengelolaan Kelas
Aktor	: Petugas TU
Tujuan	: mencatat, menghapus, dan meng*update* data kelas
Deskripsi	: 1. Mencatat data kelas kedalam *database* 2. Mencari data kelas, kemudian dapat menghapus data tersebut atau meng*update*nya
Proses mencatat kelas baru	
Kondisi awal	: Halaman utama, menu kelas, form *input* kelas
Kondisi akhir	: Kelas yang baru tersimpan di *database*

Aktor	Sistem
1. Petugas TU masuk ke menu kelas, lalu ke form *input* kelas 2. Petugas TU memasukkan data kelas yang baru	3. Sistem menampilkan form *input* tersebut 4. Sistem menyimpan data tersebut kedalam *database* 5. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog

<table>
<tr><td></td><td>peringatan.</td></tr>
<tr><td colspan="2">Proses mencari kelas</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu kelas, form cari kelas</td></tr>
<tr><td colspan="2">Kondisi akhir : Menampilkan kelas yang dicari user</td></tr>
<tr><td>Aktor</td><td>Sistem</td></tr>
<tr><td>1. Petugas TU masuk ke menu kelas, lalu ke form cari kelas
2. Petugas TU memasukkan kode kelas yang akan dicari</td><td>3. Sistem menampilkan form cari tersebut
4. Sistem menampilkan kelas tersebut
5. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan data yang di cari tidak ditemukan.</td></tr>
</table>

<table>
<tr><td colspan="2">Nomor : 04</td></tr>
<tr><td colspan="2">Nama use case : Pengelolaan data siswa</td></tr>
<tr><td colspan="2">Aktor : Petugas TU</td></tr>
<tr><td colspan="2">Tujuan : mencatat, menghapus, dan mengupdate data siswa</td></tr>
<tr><td colspan="2">Deskripsi : 1. Mencatat data kelas kedalam database
2. Mencari data siswa, kemudian dapat menghapus data tersebut atau mengupdatenya</td></tr>
<tr><td colspan="2">Proses mencatat data siswa</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu siswa, form input siswa</td></tr>
<tr><td colspan="2">Kondisi akhir : Data siswa yang baru tersimpan di database</td></tr>
<tr><td>Aktor</td><td>Sistem</td></tr>
<tr><td>1. Petugas TU masuk ke menu siswa, lalu ke form input data siswa
2. Petugas TU memasukkan data siswa yang baru</td><td>3. Sistem menampilkan form input tersebut
4. Sistem menyimpan data tersebut kedalam database
5. Jika user salah dalam memasukkan</td></tr>
</table>

	data, maka akan muncul kotak dialog peringatan.
Proses mencari data siswa	
Kondisi sebelum : Halaman utama, menu siswa, form cari siswa	
Kondisi sesudah : Menampilkan data siswa yang dicari *user*	
Aktor	Sistem
1. Petugas TU masuk ke menu siswa, lalu ke form cari siswa. 2. Petugas TU memasukkan nis siswa yang akan dicari	3. Sistem menampilkan form cari tersebut 4. Jika nis yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data siswa tersebut. 5. Jika nis tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.
Proses mengubah data siswa	

Kondisi awal : Halaman utama, menu siswa, form ubah siswa	
Kondisi akhir : data siswa telah ter*update*	
Aktor	Sistem
1. Petugas TU masuk ke menu siswa, lalu ke form ubah siswa 2. Petugas TU mencari data siswa yang akan diubah dengan meng*input*kan nis. 3. Petugas TU mengubah data siswa tersebut	4. Sistem menampilkan form ubah tersebut 6. Jika nis yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data siswa tersebut. 7. Jika nis tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan. 8. Sistem memberikan peringatan, apakah data tersebut akan diubah? 9. Jika iya, maka akan sistem menyimpan kembali data siswa yang telah diubah oleh *user*.

	10. Jika user salah dalam mencatat data, maka akan muncul kotak dialog peringatan
Proses hapus data siswa	
Kondisi awal : Halaman utama, menu siswa, form hapus siswa	
Kondisi akhir : Data siswa telah terhapus	
Aktor	Sistem
1. Petugas TU masuk ke menu siswa, lalu ke form hapus siswa 2. Petugas TU mencari data siswa yang akan dihapus dengan meng*input*kan nis. 3. Petugas TU menghapus data siswa tersebut	4. Sistem menampilkan form hapus tersebut 5. Jika nis yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data siswa tersebut. 6. Jika nis tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan. 7. Sistem memberikan peringatan, apakah data tersebut akan dihapus?

	8. Jika iya, maka data akan terhapus.

Nomor : 05	
Nama use case : Pengelolaan Orang Tua	
Aktor : Petugas TU	
Tujuan : mencatat, menghapus, dan meng*update* data orang tua siswa	
Deskripsi : 1. Mencatat data orang tua kedalam *database* 2. Mencari data orang tua, kemudian dapat menghapus data tersebut atau meng*update*nya	
Proses mencatat data orang tua siswa	
Kondisi awal : Halaman utama, menu siswa, form *input* siswa	
Kondisi akhir : Data orang tua siswa yang baru tersimpan di *database*	
Aktor	Sistem

1. Petugas TU masuk ke menu siswa, lalu ke form *input* data siswa 2. Petugas TU memasukkan data orang tua siswa yang baru	6. Sistem menampilkan form *input* tersebut 7. Sistem menyimpan data tersebut kedalam *database* 8. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan.

Proses mencari data orang tua siswa	
Kondisi awal : Halaman utama, menu kelas, form cari siswa	
Kondisi akhir : Menampilkan data orang tua siswa yang dicari *user*	
Aktor	Sistem
1. Petugas TU masuk ke menu siswa, lalu ke form cari siswa. 2. Petugas TU memasukkan nis siswa yang akan dicari data orang tuanya	3. Sistem menampilkan form cari tersebut 4. Jika nis yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data orang tua siswa

	tersebut. 5. Jika nis tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.
Proses mengubah data orang tua siswa	
Kondisi awal : Halaman utama, menu siswa, form ubah siswa	
Kondisi akhir : data orang tua siswa telah ter*update*	
Aktor	Sistem
1. Petugas TU masuk ke menu siswa, lalu ke form ubah siswa 2. Petugas TU mencari data orang tua siswa yang akan diubah dengan meng*input*kan nis. 3. Petugas TU mengubah data orang	4. Sistem menampilkan form ubah tersebut 5. Jika nis yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data orang tua siswa tersebut. 6. Jika nis tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.

tua siswa tersebut	7. Sistem memberikan peringatan, apakah data tersebut akan diubah? 8. Jika iya, maka akan sistem menyimpan kembali data orang tua siswa yang telah diubah oleh *user*. 9. Jika user salah dalam mencatat data, maka akan muncul kotak dialog peringatan
Proses hapus data orang tua siswa	
Kondisi awal : Halaman utama, menu siswa, form hapus siswa	
Kondisi akhir : Data orang tua siswa telah terhapus	
Aktor	Sistem
1. Petugas TU masuk ke menu siswa, lalu ke form hapus siswa 2. Petugas TU mencari data siswa yang akan dihapus dengan	4. Sistem menampilkan form hapus tersebut 5. Jika nis yang dimasukkan *user* terdaftar, maka sistem akan

meng*input*kan nis. 3. Petugas TU menghapus data orang tua tersebut	menampilkan data orang tua siswa tersebut. 6. Jika nis tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan. 7. Sistem memberikan peringatan, apakah data tersebut akan dihapus? 8. Jika iya, maka data akan terhapus.

Nomor	: 06
Nama use case	: Pengelolaan Data Guru
Aktor	: Petugas TU
Tujuan	: mencatat, menghapus, dan meng*update* data guru
Deskripsi	: 1. Mencatat data guru kedalam *database* 2. Mencari data guru, kemudian dapat menghapus data tersebut atau meng*update*nya

<table>
<tr><td colspan="2">Proses mencatat data guru</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu guru, form input guru</td></tr>
<tr><td colspan="2">Kondisi akhir : Data guru yang baru tersimpan di database</td></tr>
<tr><td>Aktor</td><td>Sistem</td></tr>
<tr><td>1. Petugas TU masuk ke menu guru, lalu ke form input data guru
2. Petugas TU memasukkan data guru yang baru</td><td>3. Sistem menampilkan form input tersebut
4. Sistem menyimpan data tersebut kedalam database
5. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan.</td></tr>
<tr><td colspan="2">Proses mencari data guru</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu guru, form cari guru</td></tr>
<tr><td colspan="2">Kondisi akhir : Menampilkan data guru yang dicari user</td></tr>
<tr><td>Aktor</td><td>Sistem</td></tr>
</table>

1. Petugas TU masuk ke menu guru, lalu ke form cari guru. 2. Petugas TU memasukkan kode guru yang akan dicari.	3. Sistem menampilkan form cari tersebut 4. Jika kode guru yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data guru tersebut. 5. Jika guru tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.
Proses mengubah data guru	
Kondisi awal : Halaman utama, menu guru, form ubah guru	
Kondisi akhir : data guru telah ter*update*	
Aktor	Sistem
1. Petugas TU masuk ke menu guru, lalu ke form ubah guru. 2. Petugas TU mencari data guru yang akan diubah dengan meng*input*kan kode guru.	4. Sistem menampilkan form ubah tersebut 5. Jika kode guru yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data guru tersebut.

3. Petugas TU mengubah data guru tersebut	6. Jika kode guru tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan. 7. Sistem memberikan peringatan, apakah data tersebut akan diubah? 8. Jika iya, maka akan sistem menyimpan kembali data guru yang telah diubah oleh *user*. 9. Jika user salah dalam mencatat data, maka akan muncul kotak dialog peringatan
Proses hapus data guru	
Kondisi awal : Halaman utama, menu guru, form hapus guru	
Kondisi akhir : Data guru telah terhapus	

Aktor	Sistem
1. Petugas TU masuk ke menu guru, lalu ke form hapus guru 2. Petugas TU mencari data guru yang akan dihapus dengan meng*input*kan kode guru. 3. Petugas TU menghapus data guru tersebut	4. Sistem menampilkan form hapus tersebut 5. Jika kode guru yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data guru tersebut. 6. Jika kode guru tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan. 7. Sistem memberikan peringatan, apakah data tersebut akan dihapus? 8. Jika iya, maka data akan terhapus.

Nomor : 07
Nama use case : Pengelolaan Data Mata Pelajaran
Aktor : Petugas TU

Tujuan : mencatat, menghapus, dan meng*update* data mata pelajaran	
Deskripsi : 1. Mencatat data mata pelajaran kedalam *database* 2.Mencari data mata pelajaran, kemudian dapat menghapus data tersebut atau meng*update*nya	
Proses mencatat data mata pelajaran	
Kondisi awal : Halaman utama, menu mata pelajaran, form *input* mata pelajaran	
Kondisi akhir : Data mata pelajaran yang baru tersimpan di *database*	
Aktor	Sistem
1. Petugas TU masuk ke menu mata pelajaran, lalu ke form *input* data mata pelajaran 2. Petugas TU memasukkan data mata pelajaran yang baru	3. Sistem menampilkan form *input* tersebut 4. Sistem menyimpan data tersebut kedalam *database* 5. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan.

<table>
<tr><td colspan="2">Proses mencari data mata pelajaran</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu mata pelajaran, form cari mata pelajaran</td></tr>
<tr><td colspan="2">Kondisi akhir : Menampilkan data mata pelajaran yang dicari user</td></tr>
<tr><td>Aktor</td><td>Sistem</td></tr>
<tr><td>1. Petugas TU masuk ke menu mata pelajaran, lalu ke form cari mata pelajaran.
2. Petugas TU memasukkan kode mata pelajaran yang akan dicari.</td><td>3. Sistem menampilkan form cari tersebut
4. Jika kode mata pelajaran yang dimasukkan user terdaftar, maka sistem akan menampilkan data mata pelajaran tersebut tersebut.
5. Jika kode mata pelajaran tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.</td></tr>
<tr><td colspan="2">Proses mengubah data mata pelajaran</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu mata pelajaran, form ubah mata pelajaran</td></tr>
</table>

Kondisi akhir : data mata pelajaran telah ter*update*	
Aktor	Sistem
1. Petugas TU masuk ke menu mata pelajaran, lalu ke form ubah mata pelajaran. 2. Petugas TU mencari data mata pelajaran yang akan diubah dengan meng*input*kan kode mata pelajaran. 3. Petugas TU mengubah data mata pelajaran tersebut	4. Sistem menampilkan form ubah tersebut 5. Jika kode mata pelajaran yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data mata pelajaran tersebut tersebut. 6. Jika kode mata pelajaran tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan. 7. Sistem memberikan peringatan, apakah data tersebut akan diubah? 8. Jika iya, maka akan sistem

	menyimpan kembali data mata pelajaran yang telah diubah oleh *user*. 9. Jika user salah dalam mencatat data, maka akan muncul kotak dialog peringatan
Proses hapus data mata pelajaran	
Kondisi awal : Halaman utama, menu mata pelajaran, form hapus mata pelajaran	
Kondisi akhir : Data mata pelajaran telah terhapus	
Aktor	Sistem
1. Petugas TU masuk ke menu mata pelajaran, lalu ke form hapus mata pelajaran 2. Petugas TU mencari data mata pelajaran yang akan dihapus dengan meng*input*kan kode mata pelajaran.	4. Sistem menampilkan form hapus tersebut 5. Jika kode mata pelajaran yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data mata pelajaran tersebut. 6. Jika kode mata pelajaran tidak

	terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.
3. Petugas TU menghapus data mata pelajaran tersebut	7. Sistem memberikan peringatan, apakah data tersebut akan dihapus? 8. Jika iya, maka data akan terhapus.

Nomor	: 08
Nama use case	: Pengelolaan nilai siswa
Aktor	: Petugas TU
Tujuan	: mencatat, menghapus, dan meng*update* data nilai siswa
Deskripsi	: 1. Mencatat data nilai siswa kedalam *database* 2. Mencari nilai siswa, kemudian dapat menghapus data tersebut atau meng*update*nya
Proses mencatat data nilai siswa	

<table>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu nilai siswa, form input nilai siswa</td></tr>
<tr><td colspan="2">Kondisi akhir : Data nilai siswa tersimpan di database</td></tr>
<tr><td>Aktor</td><td>Sistem</td></tr>
<tr><td>1. Petugas TU masuk ke menu nilai siswa, lalu ke form input nilai siswa.
2. Petugas TU memilih kelas siswa, dan mata pelajaran siswa
3. Petugas TU menginputkan nilai siswa</td><td>4. Sistem menampilkan form input tersebut
5. Sistem menampilkan data siswa berdasarkan kelas dan mata pelajaran yang dipilih user
6. Sistem menyimpan data tersebut kedalam database
7. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan.</td></tr>
<tr><td colspan="2">Proses mencari nilai siswa</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu nilai siswa, form cari nilai siswa</td></tr>
</table>

Kondisi akhir : Menampilkan nilai siswa kinerja yang dicari *user*	
Aktor	Sistem
1. Petugas TU masuk ke menu nilai siswa, lalu ke form cari nilai siswa. 2. Petugas TU memilih kelas siswa, dan mata pelajaran siswa 3. Petugas TU memilih salah 1 siswa	4. Sistem menampilkan form cari tersebut 5. Sistem menampilkan data siswa berdasarkan kelas dan mata pelajaran yang dipilih *user* 6. Sistem menampilkan data nilai siswa tersebut.
Proses mengubah data nilai siswa	
Kondisi awal : Halaman utama, menu nilai siswa, form ubah nilai siswa	
Kondisi akhir : data nilai siswa telah ter*update*	
Aktor	Sistem
1. Petugas TU masuk ke menu nilai siswa, lalu ke form ubah nilai	4. Sistem menampilkan form ubah tersebut

siswa. 2. Petugas TU memilih kelas siswa, dan mata pelajaran siswa 3. Petugas TU memilih salah 1 siswa kemudian mengubah data nilai siswa tersebut	5. Sistem menampilkan data siswa berdasarkan kelas dan mata pelajaran yang dipilih *user* 6. Sistem memberikan peringatan, apakah data tersebut akan diubah? Jika iya, maka sistem akan menyimpan kembali data nilai siswa yang telah diubah oleh *user*. 7. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan.

Nomor : 09
Nama use case : Pengelolaan kehadiran siswa

Aktor	: Petugas TU
Tujuan	: mencatat, menghapus, dan meng*update* data kehadiran siswa
Deskripsi	: 1. Mencatat data kehadiran siswa kedalam *database* 2. Mencari data kehadiran siswa, kemudian dapat menghapus data tersebut atau meng*update*nya
Proses mencatat data kehadiran siswa	
Kondisi awal	: Halaman utama, menu kehadiran siswa, form *input* kehadiran siswa
Kondisi akhir	: Data kehadiran siswa tersimpan di *database*

Aktor	Sistem
1. Petugas TU masuk ke menu kehadiran siswa, lalu ke form *input* kehadiran nilai siswa.	4. Sistem menampilkan form *input* tersebut
2. Petugas TU memilih kelas siswa, dan mata pelajaran siswa	5. Sistem menampilkan data siswa berdasarkan kelas dan mata pelajaran yang dipilih *user*
3. Petugas TU memilih salah 1	

siswa kemudian meng*input*kan data kehadiran siswa tersebut	6. Sistem menyimpan data tersebut kedalam *database* 7. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan.
Proses mencari data kehadiran siswa	
Kondisi awal : Halaman utama, menu kehadiran siswa, form cari kehadiran siswa	
Kondisi akhir : Menampilkan data kehadiran siswa yang dicari *user*	
Aktor	Sistem
1. Petugas TU masuk ke menu kehadiran siswa, lalu ke form cari kehadiran siswa. 2. Petugas TU memilih kelas siswa, dan mata pelajaran siswa 3. Petugas TU memilih salah 1 siswa	4. Sistem menampilkan form cari tersebut 5. Sistem menampilkan data siswa berdasarkan kelas dan mata pelajaran yang dipilih *user*

	6. Sistem menampilkan data kehadiran siswa tersebut.

Nomor : 10	
Nama use case : Pengelolaan keuangan siswa	
Aktor : Petugas TU	
Tujuan : mencatat, menghapus, dan meng*update* data keuangan siswa	
Deskripsi : 1. Mencatat data keuangan siswa kedalam *database* 2. Mencari data keuangan siswa, kemudian dapat menghapus data tersebut atau meng*update*nya	
Proses mencatat data keuangan siswa	
Kondisi awal : Halaman utama, menu keuangan, form *input* keuangan siswa	
Kondisi akhir : Data keuangan siswa tersimpan di *database*	
Aktor	Sistem

1. Petugas TU masuk ke menu keuangan siswa, lalu ke form *input* keuangan nilai siswa. 2. Petugas TU memilih kelas siswa. 3. Petugas TU memilih salah 1 siswa kemudian meng*input*kan data keuangan siswa tersebut	4. Sistem menampilkan form *input* tersebut 5. Sistem menampilkan data siswa berdasarkan kelas 6. Sistem menyimpan data tersebut kedalam *database* 7. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan.
Proses mencari data keuangan siswa	
Kondisi awal : Halaman utama, menu keuangan siswa, form cari keuangan siswa	
Kondisi akhir : Menampilkan data keuangan siswa yang dicari *user*	

Aktor	Sistem
1. Petugas TU masuk ke menu keuangan siswa, lalu ke form cari keuangan siswa. 2. Petugas TU memilih kelas siswa 3. Petugas TU memilih salah 1 siswa	4. Sistem menampilkan form cari tersebut 5. Sistem menampilkan data siswa berdasarkan kelas yang dipilih *user* 6. Sistem menampilkan data kehadiran siswa tersebut.

Nomor	: 11
Nama use case	: Pengelolaan beasiswa
Actor	: Petugas TU
Tujuan	: mencatat, menghapus, dan meng*update* data penerima beasiswa
Deskripsi	: 1. Mencatat data penerima beasiswa kedalam *database* 2. Mencari data penerima besiswa, kemudian dapat menghapus data tersebut atau meng*update*nya

<table>
<tr><td colspan="2">Proses mencatat data penerima beasiswa</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu nilai kinerja, form input nilai kinerja</td></tr>
<tr><td colspan="2">Kondisi akhir : Data siswa yang menerima beasiswa tersimpan di database</td></tr>
<tr><td>Aktor</td><td>Sistem</td></tr>
<tr><td>1. Petugas TU masuk ke menu siswa, lalu ke form input beasiswa.
2. Petugas TU mencari data siswa yang berhak menerima berasiswa.
3. Petugas TU memilih siswa yang berhak menerima beasiswa, kemudian meng input kan data tersebut.</td><td>4. Sistem menampilkan form input tersebut
5. Sistem menampilkan data siswa yang dicari user.
6. Sistem menyimpan data tersebut kedalam database</td></tr>
<tr><td colspan="2">Proses mencari data siswa penerima beasiswa</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu report , form cari penerima beasiswa</td></tr>
</table>

Kondisi akhir : Menampilkan data penerima beasiswa yang dicari *user*	
Aktor	Sistem
1. Petugas TU masuk ke menu report kinerja, lalu ke form Penerima Beasiswa. 2. Petugas TU mencari data penerima beasiswa berdasarkan nis.	3. Sistem menampilkan form cari tersebut 4. Sistem menampilkan data siswa yang menerima beasiswa 5. Jika nis yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data tersebut. 6. Jika nis tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.

Nomor	: 12
Nama use case	: Report Data Siswa
Actor	: Kepala Sekolah
Tujuan	: melihat laporan data siswa

Deskripsi : melaporkan data siswa yang terdaftar di SMA Negeri 2 Purwakarta	
Laporan Data Siswa	
Kondisi sebelum : Halaman utama, menu siswa, form cari siswa	
Kondisi sesudah : Menampilkan data siswa yang dicari *user*	
Aktor	Sistem
1. Kepala sekolah masuk ke menu siswa, lalu ke form cari siswa. 2. Kepala sekolah memasukkan nis siswa yang akan dicari	3. Sistem menampilkan form cari tersebut 4. Jika nis yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data siswa tersebut. 5. Jika nis tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.

Nomor : 13
Nama use case : Report Data Guru

Actor : Guru	
Tujuan : melihat data guru	
Deskripsi : melaporkan data guru yang mengajar di SMA Negeri 2 Purwakarta	
Laporan Data Guru	
Kondisi awal : Halaman utama, menu guru, form cari guru	
Kondisi akhir : Menampilkan data guru yang dicari *user*	
Aktor	Sistem
1. Kepala Sekolah masuk ke menu guru, lalu ke form cari guru. 2. Kepala Sekolah memasukkan kode guru yang akan dicari.	3. Sistem menampilkan form cari tersebut 4. Jika kode guru yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data guru tersebut. 5. Jika guru tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.

Nomor : 14	
Nama use case : Report Nilai Siswa	
Actor : Guru	
Tujuan : melihat nilai siswa	
Deskripsi : mengitung dan melaporkan data nilai siswa	
Laporan Nilai Siswa	
Kondisi awal : Halaman utama, menu nilai siswa, form cari nilai siswa	
Kondisi akhir : Menampilkan nilai siswa kinerja yang dicari *user*	
Aktor	Sistem
1. Kepala Sekolah atau Wali Kelas masuk ke menu nilai siswa, lalu ke form cari nilai siswa. 2. Kepala Sekolah atau Wali Kelas memilih kelas siswa, dan mata pelajaran siswa	4. Sistem menampilkan form cari tersebut 5. Sistem menampilkan data siswa berdasarkan kelas dan mata pelajaran yang dipilih *user*

3. Kepala Sekolah atau Wali Kelas memilih salah 1 siswa	6. Sistem menampilkan data nilai siswa tersebut.

Nomor : 15	
Nama use case : Report Kehadiran siswa	
Actor : Guru	
Tujuan : melihat data kehadiran siswa	
Deskripsi : melaporkan data kehadiran siswa	
Laporan Kehadiran Siswa	
Kondisi awal : Halaman utama, menu kehadiran siswa, form cari kehadiran siswa	
Kondisi akhir : Menampilkan data kehadiran siswa yang dicari *user*	
Aktor	Sistem
1. Kepala Sekolah atau Wali Kelas masuk ke menu kehadiran siswa,	4. Sistem menampilkan form cari tersebut

lalu ke form cari kehadiran siswa. 2. Kepala Sekolah atau Wali Kelas memilih kelas siswa, dan mata pelajaran siswa 3. Kepala Sekolah atau Wali Kelas memilih salah 1 siswa	 5. Sistem menampilkan data siswa berdasarkan kelas dan mata pelajaran yang dipilih *user* 6. Sistem menampilkan data kehadiran siswa tersebut.

Nomor	: 16
Nama use case	: Report Keuangan
Actor	: Guru
Tujuan	: melihat data keuangan siswa siswa
Deskripsi	: melaporkan data keuangan siswa
Laporan Keuangan Siswa	

Kondisi awal : Halaman utama, menu keuangan siswa, form cari keuangan siswa	
Kondisi akhir : Menampilkan data keuangan siswa yang dicari *user*	
Aktor	Sistem
1. Kepala Sekolah atau Wali Kelas masuk ke menu keuangan siswa, lalu ke form cari keuangan siswa. 2. Kepala Sekolah atau Wali Kelas memilih kelas siswa 3. Kepala Sekolah atau Wali Kelas memilih salah 1 siswa	4. Sistem menampilkan form cari tersebut 5. Sistem menampilkan data siswa berdasarkan kelas yang dipilih *user* 6. Sistem menampilkan data kehadiran siswa tersebut.

Nomor : 17
Nama use case : Report Penerima Beasiswa

Actor	: Guru
Tujuan	: melihat data siswa yang berhak menerima beasiswa
Deskripsi	: melaporkan data siswa yang berhak menerima beasiswa
Laporan Penerima Beasiswa	
Kondisi awal	: Halaman utama, menu beasiswa, form cari penerima beasiswa
Kondisi akhir	: Menampilkan data penerima beasiswa yang dicari *user*

Aktor	Sistem
1. Kepala Sekolah masuk ke menu Beasiswa, lalu ke form cari beasiswa. 2. Kepala Sekolah mencari data penerima beasiswa berdasarkan nis.	3. Sistem menampilkan form cari tersebut 4. Jika nis yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data penerima beasiswa tersebut. 5. Jika kode guru tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.

Nomor : 18	
Nama use case : Pengelolaan data mengajar guru	
Aktor : Petugas TU	
Tujuan : mencatat, menghapus, dan meng*update* data mengajar	
Deskripsi : 1. Mencatat data mengajar kedalam *database* 2. Mencari data mengajar, kemudian dapat menghapus data tersebut atau meng*update*nya	
Proses mencatat data mengajar guru	
Kondisi awal : Halaman utama, menu mengajar, form *input* data mengajar.	
Kondisi akhir : Data mengajar guru tersimpan di *database*	
Aktor	Sistem
1. Petugas TU masuk ke menu mengajar, lalu ke form *input* data mengajar. 2. Petugas TU memilih data guru, data kelas, dan data mata pelajaran	4. Sistem menampilkan form *input* tersebut 5. Sistem menampilkan data yang dicari *user*.

3. Petugas TU meng*input*kan kode mengajar guru dan jadwal mengajar guru tersebut	6. Sistem menyimpan data tersebut kedalam *database* 7. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan.
Proses mencari data mengajar guru	
Kondisi awal : Halaman utama, menu mengajar, form cari mengajar	
Kondisi akhir : Menampilkan data mengajar guru yang dicari *user*	
Aktor	Sistem
1. Petugas TU masuk ke menu mengajar , lalu ke form cari mengajar. 2. Petugas TU mencari data nilai kinerja berdasarkan kode	3. Sistem menampilkan form cari tersebut 4. Jika kode mengajar yang dimasukkan *user* terdaftar, maka

mengajar.	sistem akan menampilkan data mengajar tersebut. 5. Jika kode mengajar tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.
Proses mengubah data mengajar guru	
Kondisi awal : Halaman utama, menu mengajar, form ubah data mengajar	
Kondisi akhir : Data mengajar guru telah ter*update*	
Aktor	Sistem
1. Petugas TU masuk ke menu mengajar, lalu ke form ubah data mengajar. 2. Petugas TU mencari data mengajar guru berdasarkan kode mengajar.	4. Sistem menampilkan form ubah tersebut 5. Jika kode mengajar yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data mengajar guru tersebut. 6. Jika kode mengajar tidak terdaftar,

3. Petugas TU kemudian mengubah data mengajar guru tersebut	maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan. 7. Sistem memberikan peringatan, apakah data tersebut akan diubah? Jika iya, maka sistem akan menyimpan kembali data mengajar guru yang telah diubah oleh *user*. 8. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan.

Nomor : 19
Nama use case : Pengelolaan Data TU
Aktor : Petugas TU
Tujuan : mencatat, menghapus, dan meng*update* data TU

<table>
<tr><td colspan="2">Deskripsi : 1. Mencatat data guru kedalam database.
Mencari data guru, kemudian dapat menghapus data tersebut atau mengupdatenya</td></tr>
<tr><td colspan="2">Proses mencatat data TU</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu guru, form input TU</td></tr>
<tr><td colspan="2">Kondisi akhir : Data TU yang baru tersimpan di database</td></tr>
<tr><td>Aktor</td><td>Sistem</td></tr>
<tr><td>1. Petugas TU masuk ke menu TU, lalu ke form input data guru
2. Petugas TU memasukkan data TU yang baru</td><td>3. Sistem menampilkan form input tersebut
4. Sistem menyimpan data tersebut kedalam database
5. Jika user salah dalam memasukkan data, maka akan muncul kotak dialog peringatan.</td></tr>
<tr><td colspan="2">Proses mencari data guru</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu TU, form cari TU</td></tr>
</table>

Kondisi akhir : Menampilkan data TU yang dicari *user*	
Aktor	Sistem
1. Petugas TU masuk ke menu TU, lalu ke form cari TU. 2. Petugas TU memasukkan kode TU yang akan dicari.	3. Sistem menampilkan form cari tersebut 4. Jika kode TU yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data TU tersebut. 5. Jika guru tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.
Proses mengubah data TU	
Kondisi awal : Halaman utama, menu guru, form ubah TU	
Kondisi akhir : data TU telah ter*update*	
Aktor	Sistem
1. Petugas TU masuk ke menu TU, lalu ke form ubah TU. 2. Petugas TU mencari data TU yang akan diubah dengan	4. Sistem menampilkan form ubah tersebut 5. Jika kode TU yang dimasukkan *user* terdaftar, maka sistem akan

meng*input*kan kode TU.	menampilkan data guru tersebut. 6. Jika kode TU tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.
3. Petugas TU mengubah data TU tersebut	7. Sistem memberikan peringatan, apakah data tersebut akan diubah? 8. Jika iya, maka akan sistem menyimpan kembali dataTU yang telah diubah oleh *user*. 9. Jika user salah dalam mencatat data, maka akan muncul kotak dialog peringatan
Proses hapus data TU	
Kondisi awal : Halaman utama, menu TU, form hapus TU	

Kondisi akhir : Data TU telah terhapus	
Aktor	Sistem
1. Petugas TU masuk ke menu TU, lalu ke form hapus TU 2. Petugas TU mencari data TU yang akan dihapus dengan meng*input*kan kode TU. 3. Petugas TU menghapus data TU tersebut	4. Sistem menampilkan form hapus tersebut 5. Jika kode TU yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data guru tersebut. 6. Jika kode TU tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan. 7. Sistem memberikan peringatan, apakah data tersebut akan dihapus? 8. Jika iya, maka data akan terhapus.

Nomor : 20
Nama use case : Pengelolaan Data User

Aktor	: Petugas TU
Tujuan	: mencatat, menghapus, dan meng*update* data user
Deskripsi	: Mencatat data guru kedalam *database.* Mencari data guru, kemudian dapat menghapus data tersebut atau meng*update*nya
Proses mencatat data user	
Kondisi awal	: Halaman utama, menu user, form *input* user
Kondisi akhir	: Data user yang baru tersimpan di *database*

Aktor	Sistem
1. Petugas TU masuk ke menu user, lalu ke form *input* data user	4. Sistem menampilkan form *input* tersebut
2. Petugas TU mencari data pegawai (guru / TU).	5. Sistem menampilkan data pegawai yang dicari petugas TU.
3. Petugas TU memasukkan data user yang baru	6. Sistem menyimpan data tersebut kedalam *database*
	7. Jika user salah dalam memasukkan

<table>
<tr><td></td><td>data, maka akan muncul kotak dialog peringatan.</td></tr>
<tr><td colspan="2">Proses mencari data user</td></tr>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu user, form cari user</td></tr>
<tr><td colspan="2">Kondisi akhir : Menampilkan data user yang dicari user</td></tr>
<tr><td>Aktor</td><td>Sistem</td></tr>
<tr><td>1. Petugas TU masuk ke menu user, lalu ke form cari user.
2. Petugas TU memasukkan kode user yang akan dicari.</td><td>3. Sistem menampilkan form cari tersebut
4. Jika kode user yang dimasukkan user terdaftar, maka sistem akan menampilkan data user tersebut.
5. Jika guru tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.</td></tr>
<tr><td colspan="2">Proses mengubah data user</td></tr>
</table>

<table>
<tr><td colspan="2">Kondisi awal : Halaman utama, menu user, form ubah user</td></tr>
<tr><td colspan="2">Kondisi akhir : data user telah terupdate</td></tr>
<tr><td>Aktor</td><td>Sistem</td></tr>
<tr><td>1. Petugas TU masuk ke menu user, lalu ke form ubah user.
2. Petugas TU mencari data user yang akan diubah dengan menginputkan kode user.

3. Petugas TU mengubah data user tersebut</td><td>4. Sistem menampilkan form ubah tersebut
5. Jika kode user yang dimasukkan user terdaftar, maka sistem akan menampilkan data guru tersebut.
6. Jika kode user tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.
7. Sistem memberikan peringatan, apakah data tersebut akan diubah?
8. Jika iya, maka akan sistem menyimpan kembali data user yang</td></tr>
</table>

	telah diubah oleh *user*. 9. Jika user salah dalam mencatat data, maka akan muncul kotak dialog peringatan
Proses hapus data user	
Kondisi awal : Halaman utama, menu user, form hapus user	
Kondisi akhir : Data user telah terhapus	
Aktor	Sistem
1. Petugas TU masuk ke menu user, lalu ke form hapus user 2. Petugas TU mencari data user yang akan dihapus dengan meng*input*kan kode user. 3. Petugas TU menghapus data user	4. Sistem menampilkan form hapus tersebut 5. Jika kode user yang dimasukkan *user* terdaftar, maka sistem akan menampilkan data guru tersebut. 6. Jika kode user tidak terdaftar, maka akan muncul kotak dialog peringatan data yang dicari tidak ditemukan.

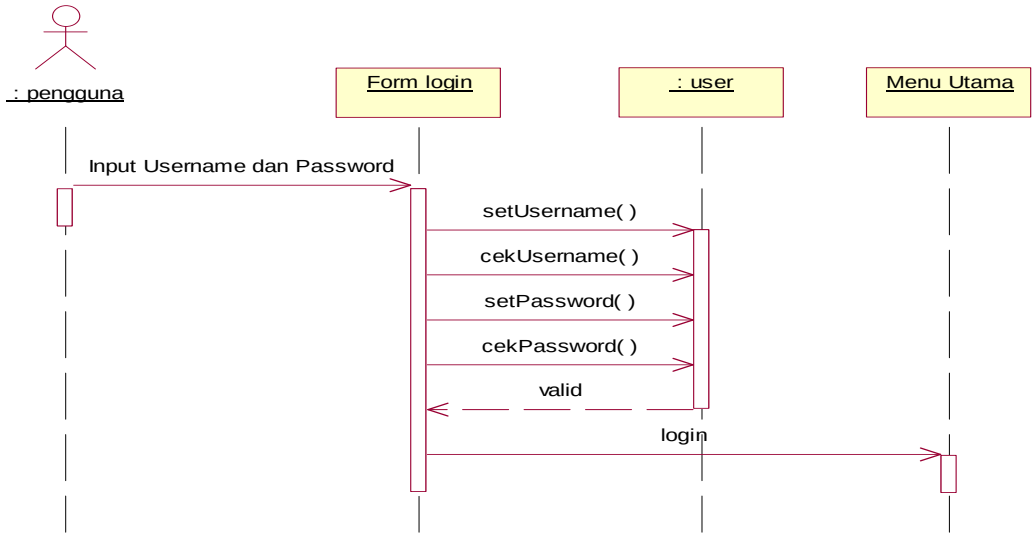
tersebut	7. Sistem memberikan peringatan, apakah data tersebut akan dihapus? 8. Jika iya, maka data akan terhapus.

3.1.4.2. Sequence Diagram

Sequence Diagram menjelaskan interaksi *object* yang disusun dalam suatu urutan waktu. *Sequence diagram* memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan suatu yang dilakukan dalam *use case*.

1. *Sequence Diagram* Proses *Login*

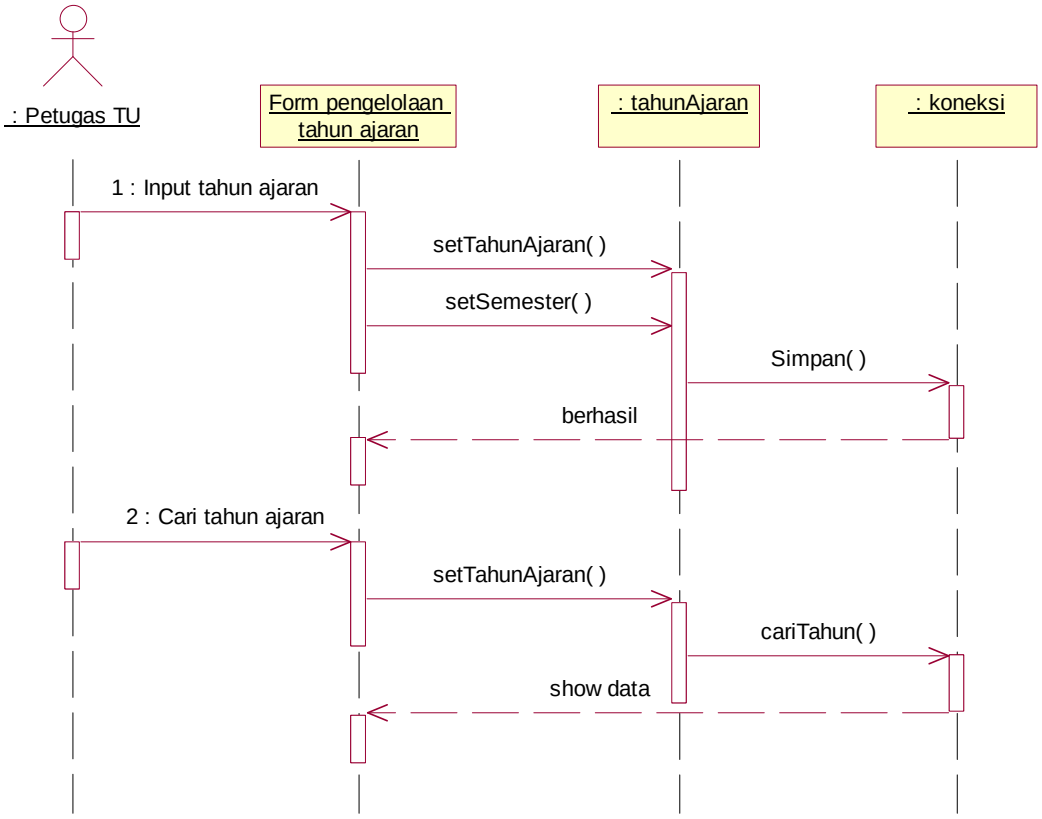
   Pada proses *login*, pengguna dalam hal ini Petugas TU, dan guru masuk ke halaman *login*. Pengguna menmasukkan *username* dan *password*, kemudian sistem memeriksa *username* dan *password* tersebut. Jika username dan password benar / *valid*, maka masuk ke halaman utama.



Gambar 3. 7 : *Sequence Diagram* Proses *Login*

2. *Sequence Diagram* Pengelolaan Tahun Ajaran

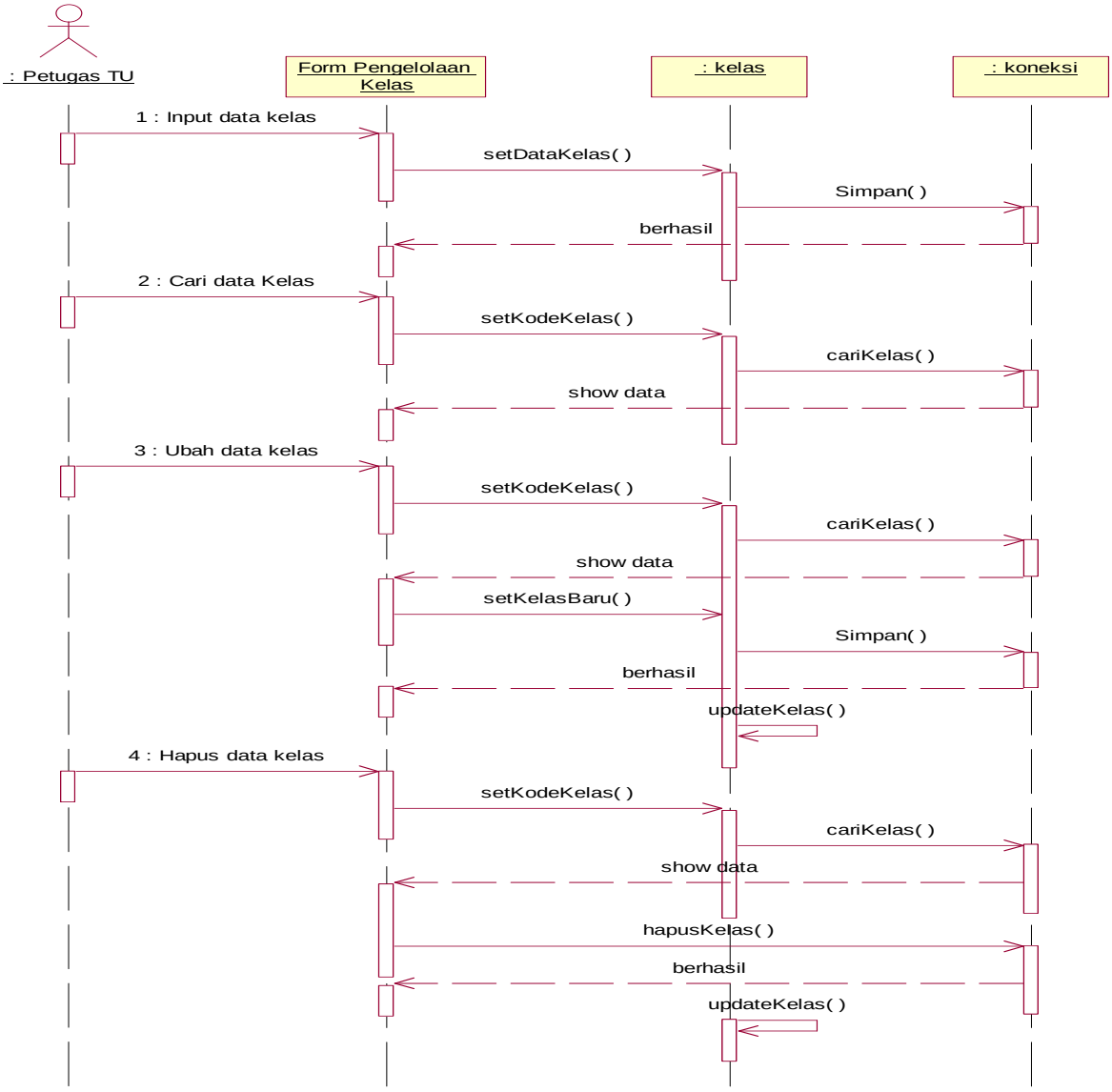
   Proses pengelolaan yang dilakukan meliputi *input* tahun ajaran, dan cari tahun ajaran. Pada proses input tahun ajaran, petugas TU masuk ke menu tahun ajaran lalu ke form input tahun ajaran. Untuk proses cari tahun ajaran, ubah tahun ajaran, dan hapus tahun ajaran, petugas TU masuk ke form cari tahun ajaran terlebih dahulu



Gambar 3. 8 : *Sequence Diagram* Pengelolaan Tahun Ajaran

3. *Sequence Diagram* Pengelolaan Kelas

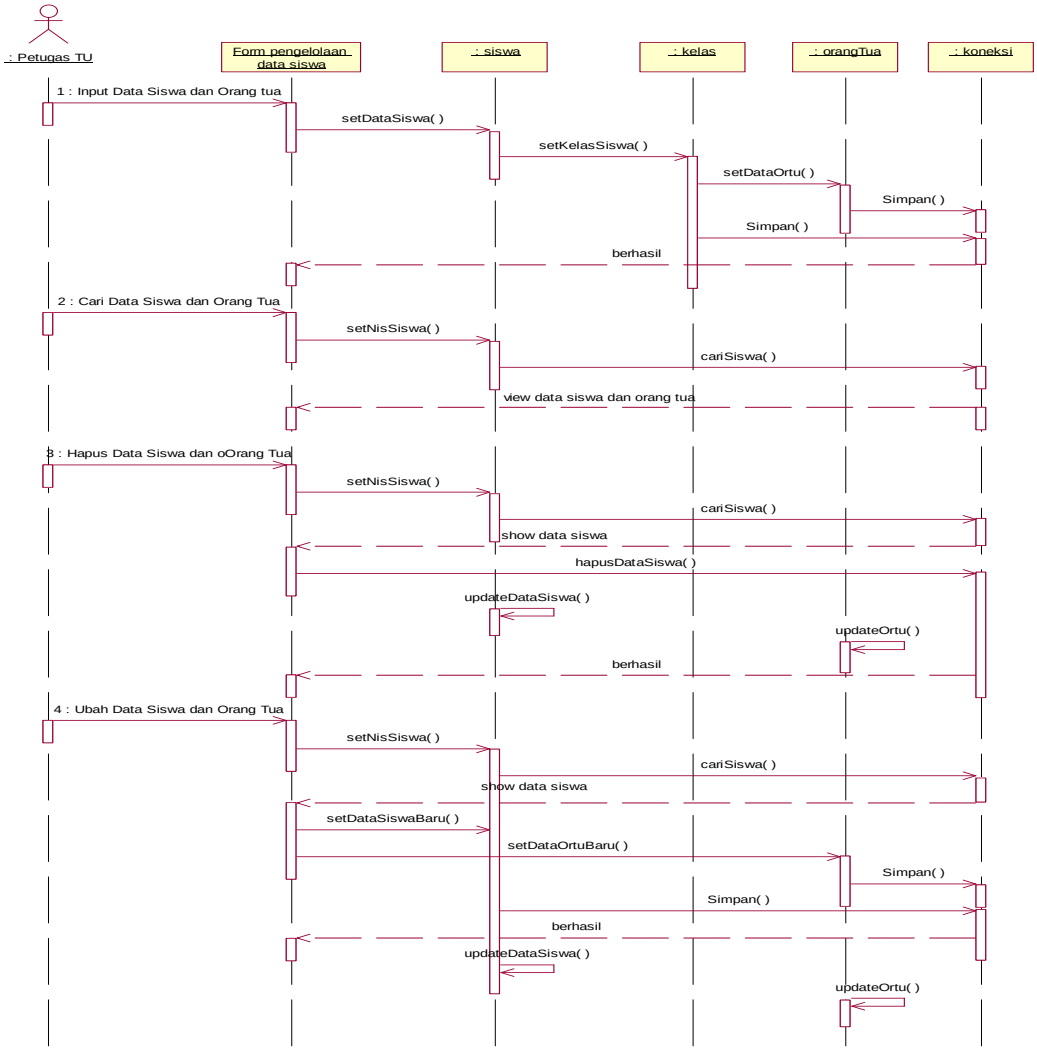
Proses pengelolaan yang dilakukan meliputi *input* kelas, cari kelas, ubah kelas dan hapus kelas. Pada proses input kelas, petugas TU masuk ke menu kelas lalu ke form input kelas. Untuk proses ubah kelas, dan hapus kelas, petugas TU masuk ke form cari kelas ajaran terlebih dahulu.



Gambar 3. 9 : *Sequence Diagram* Pengelolaan Kelas

4. *Sequence Diagram* Pengelolaan data siswa dan orang tua

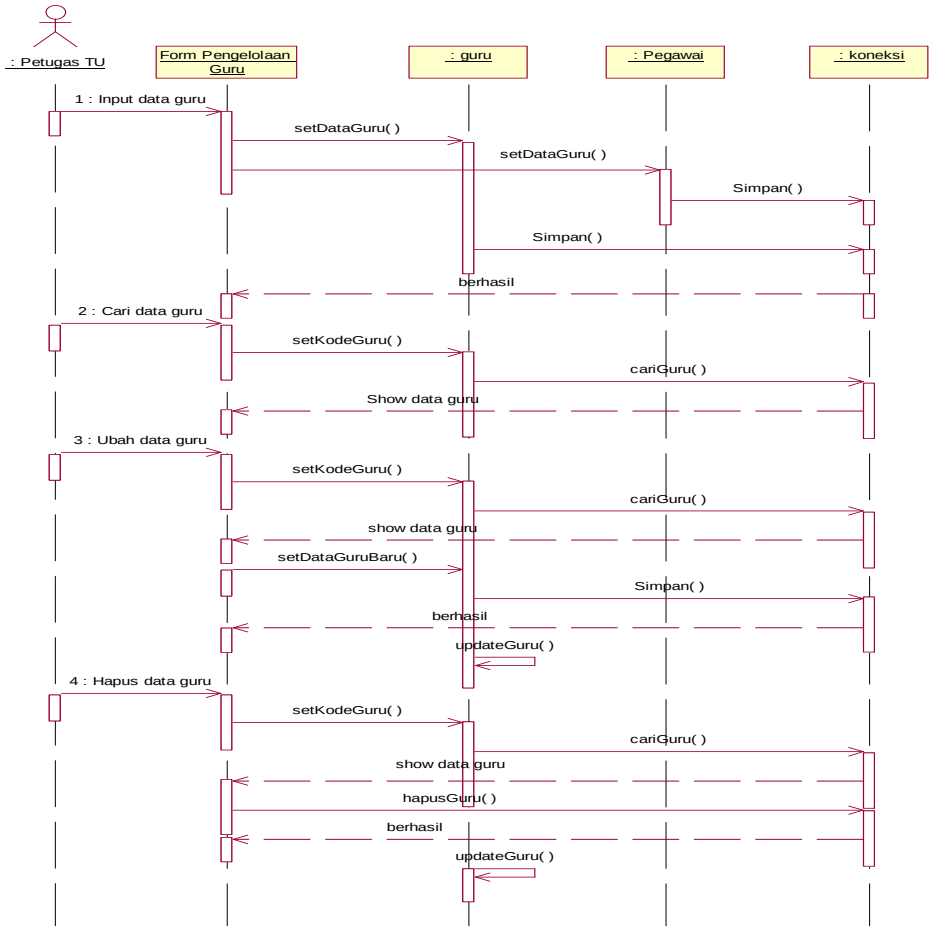
   Proses pengelolaan yang dilakukan meliputi *input* data siswa dan orang tua, cari data siswa dan orang tua, ubah data siswa dan orang tua, dan hapus data siswa dan orang tua. Pada proses input data, petugas TU masuk ke menu siswa lalu ke form input data siswa. Untuk proses ubah data siswa, dan hapus data siswa, petugas TU masuk ke form cari data siswa terlebih dahulu.



Gambar 3. 10 : *Sequence Diagram* Pengelolaan Data Siswa dan Orang Tua

5. *Sequence Diagram* Pengelolaan data guru

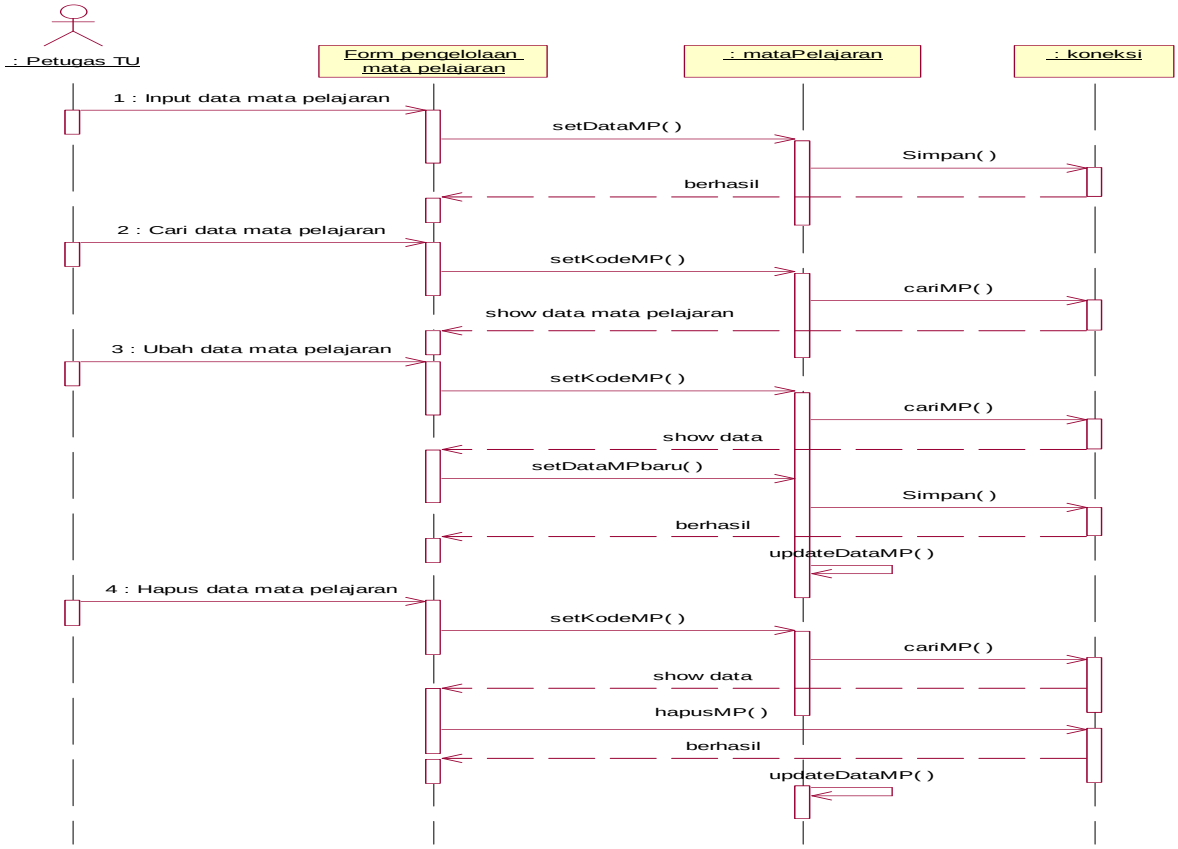
Proses pengelolaan yang dilakukan meliputi *input* data guru, cari data guru, ubah data guru dan hapus data guru. Pada proses input data guru, petugas TU masuk ke menu guru lalu ke form input data guru. Untuk proses ubah data guru, dan hapus data guru, petugas TU masuk ke form cari data guru terlebih dahulu.



Gambar 3. 11 : *Sequence Diagram* Pengelolaan Data Guru

6. *Sequence Diagram* Pengelolaan Data Mata Pelajaran

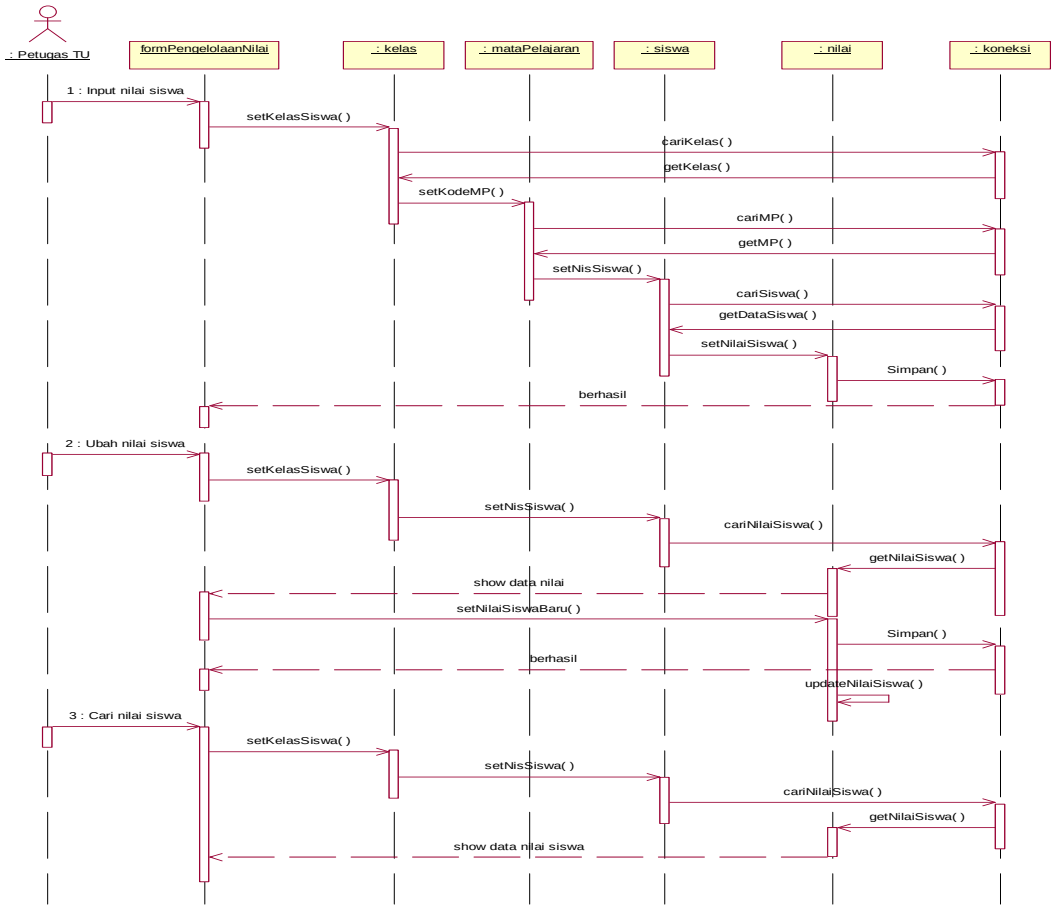
Proses pengelolaan yang dilakukan meliputi *input* data mata pelajaran, cari data mata pelajaran, ubah data mata pelajaran dan hapus data mata pelajaran. Pada proses input data mata pelajaran, petugas TU masuk ke menu mata pelajaran lalu ke form input data mata pelajaran. Untuk proses ubah data mata pelajaran, dan hapus data mata pelajaran, petugas TU masuk ke form cari data mata pelajaran terlebih dahulu.



Gambar 3. 12 : *Sequence Diagram* Pengelolaan Data Mata Pelajaran

7. *Sequence Diagram* Pengelolaan nilai siswa

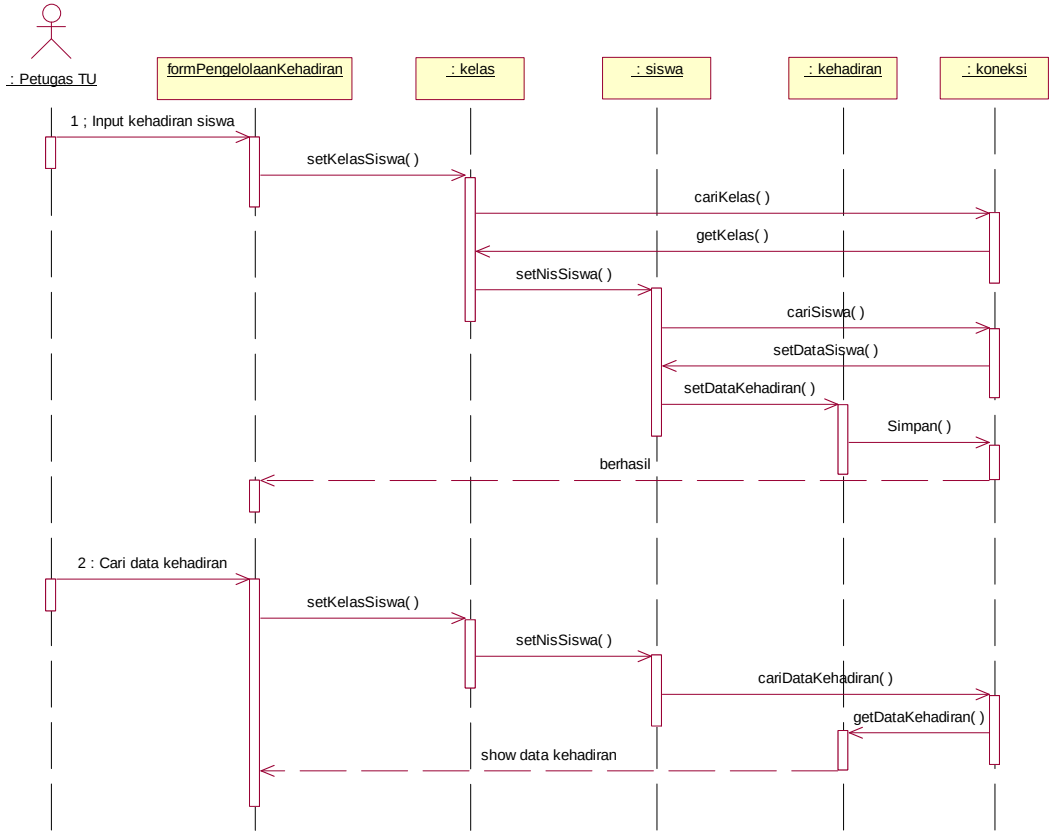
Proses pengelolaan yang dilakukan meliputi *input* nilai siswa, cari nilai siswa, dan ubah nilai. Pada proses input nilai siswa, petugas TU masuk ke menu nilai siswa lalu ke form input nilai siswa. Kemudian cari data siswa berdasarkan kelas dan nis. Untuk proses ubah nilai siswa, dan hapus nilai siswa, petugas TU masuk ke form cari data siswa terlebih dahulu.



Gambar 3. 13 : *Sequence Diagram* Pengelolaan Nilai Siswa

8. *Sequence Diagram* Pengelolaan Kehadiran Siswa

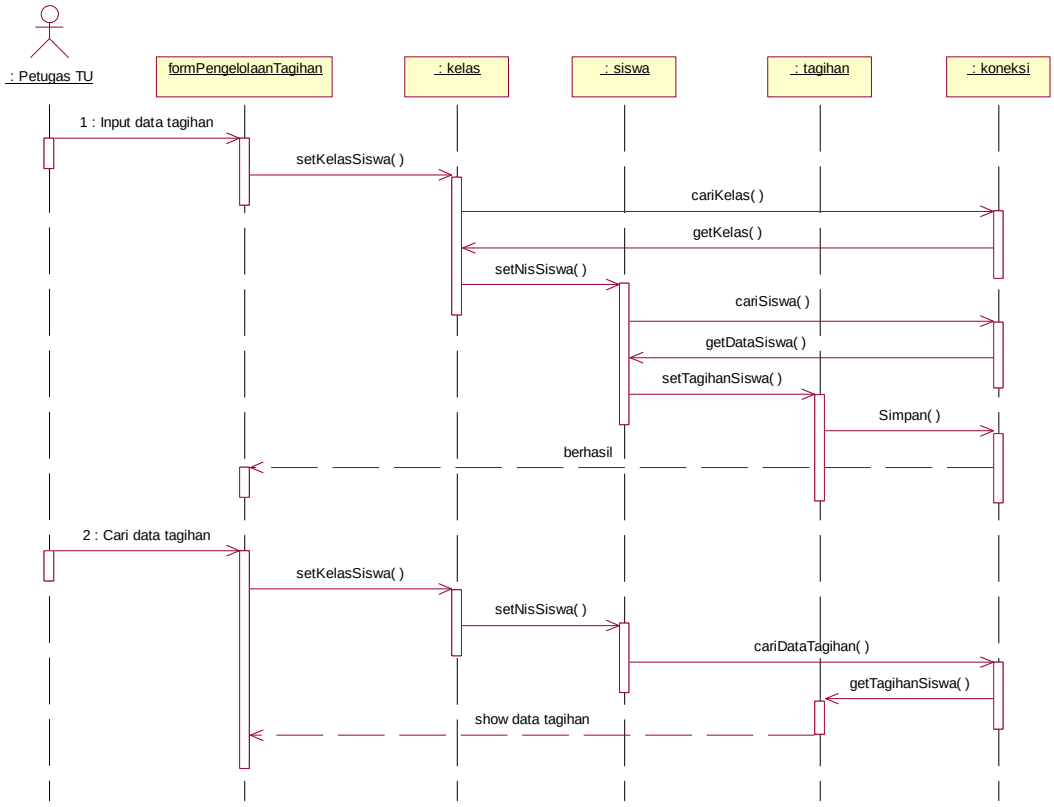
Proses pengelolaan yang dilakukan meliputi *input* kehadiran siswa, dan cari kehadiran siswa. Pada proses input kehadiran siswa, petugas TU masuk ke menu kehadiran siswa lalu ke form kehadiran siswa. Kemudian cari data siswa berdasarkan kelas dan nis. Untuk proses ubah kehadiran siswa, dan hapus kehadiran siswa, petugas TU masuk ke form cari data siswa terlebih dahulu.



Gambar 3. 14 : *Sequence Diagram* Pengelolaan Kehadiran Siswa

9. *Sequence Diagram* Pengelolaan Tagihan Siswa

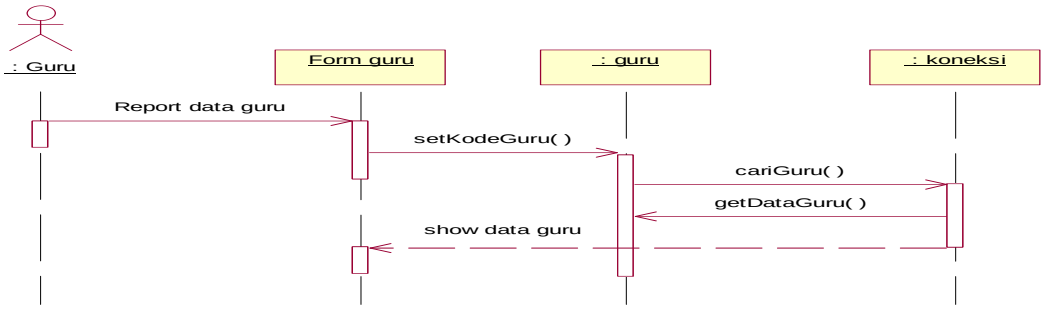
Proses pengelolaan yang dilakukan meliputi *input* tagihan siswa, dan cari tagihan siswa. Pada proses input tagihan siswa, petugas TU masuk ke menu tagihan siswa lalu ke form input tagihan siswa. Kemudian cari data siswa berdasarkan kelas dan nis. Untuk proses ubah tagihan siswa, dan hapus tagihan siswa, petugas TU masuk ke form cari data siswa terlebih dahulu.



Gambar 3. 15 : *Sequence Diagram* Pengelolaan Tagihan Siswa

10. *Sequence Diagram* Report Data Guru

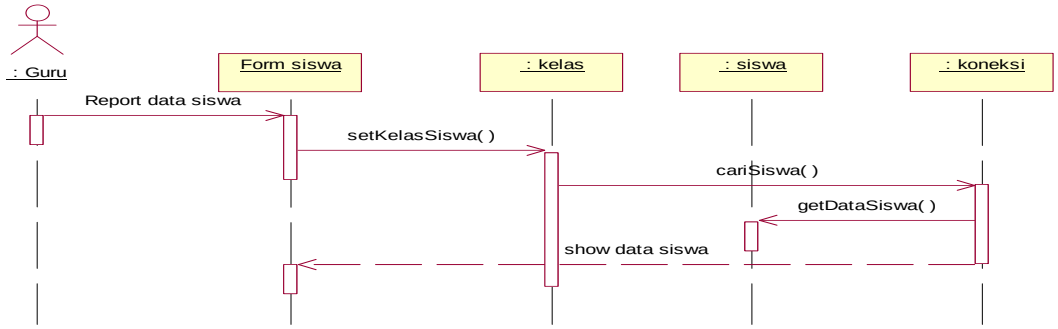
Pada proses ini, dilakukan oleh kepala sekolah. Kepala sekolah dapat melihat data guru-guru yang terdaftar di SMA Negeri 2 Purwakarta. Dalam hal ini, kepala sekolah masuk ke menu guru terlebih dahulu, lalu ke form cari data guru.



Gambar 3. 16 : *Sequence Diagram* Report Data Guru

11. *Sequence Diagram* Report Data Siswa

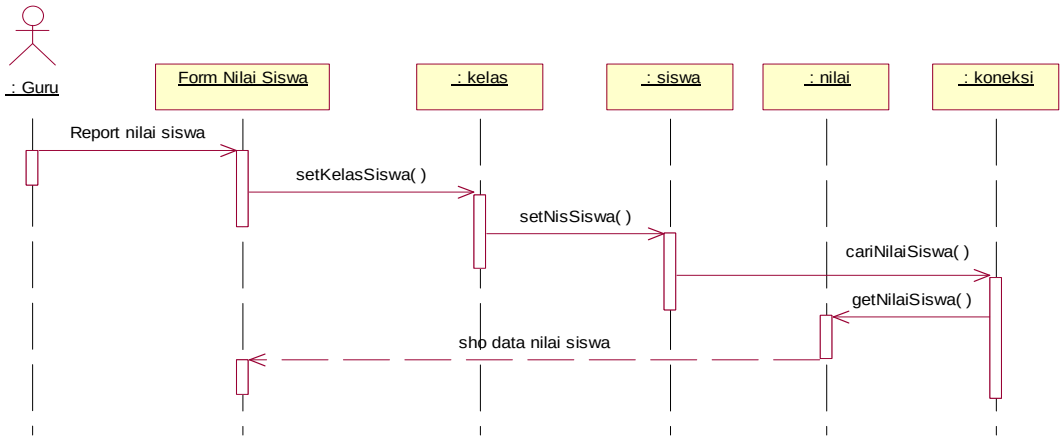
Pada proses ini, dilakukan oleh guru. Guru dapat melihat data siswa-siswa yang ada di SMA Negeri 2 Purwakarta. Dalam hal ini, guru masuk ke menu siswa terlebih dahulu, lalu ke form cari data siswa.



Gambar 3. 17 : Sequence Diagram Report Data Siswa

12. *Sequence Diagram* Report Nilai Siswa

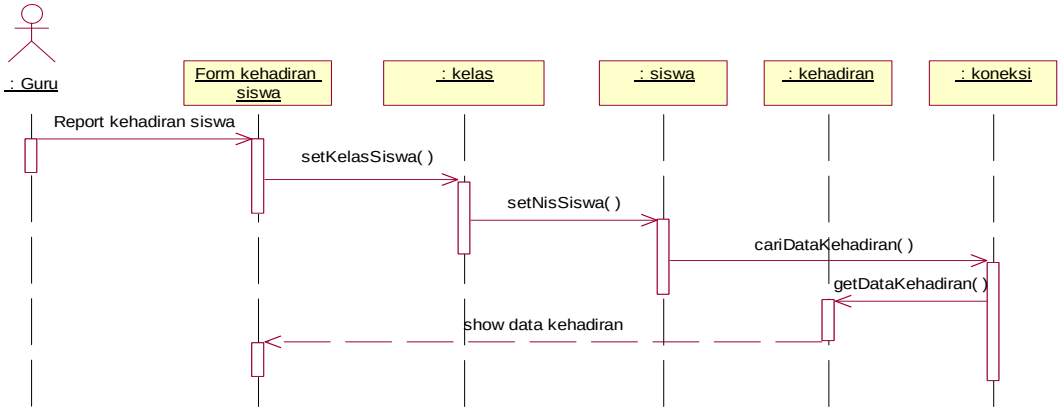
Pada proses ini, dilakukan oleh guru. Guru dapat melihat nilai-nilai siswa yang ada di SMA Negeri 2 Purwakarta. Dalam hal ini, guru masuk ke menu nilai siswa terlebih dahulu, lalu ke form cari nilai siswa.



Gambar 3. 18 : *Sequence Diagram* Report Nilai Siswa

13. Report Kehadiran Siswa

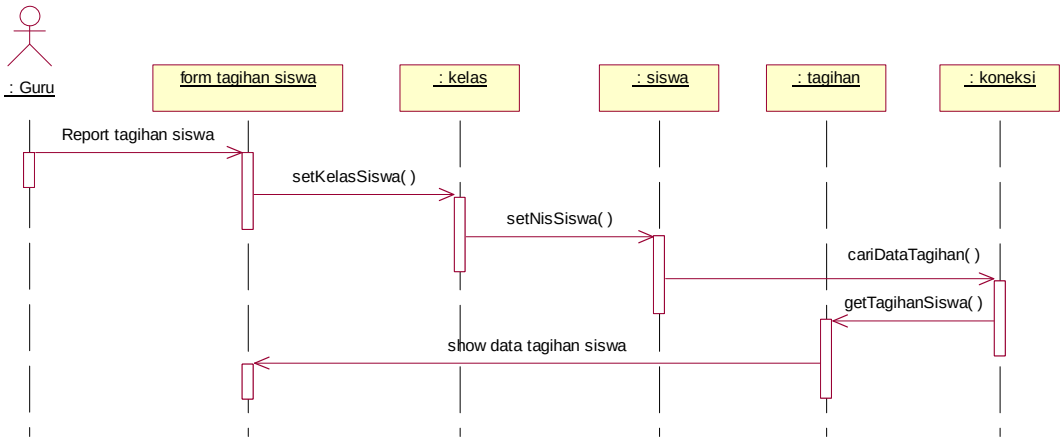
Pada proses ini, dilakukan oleh guru. Guru dapat melihat data kehadiran siswa yang ada di SMA Negeri 2 Purwakarta. Dalam hal ini, Guru masuk ke menu kehadiran siswa terlebih dahulu, lalu ke form cari kehadiran siswa.



Gambar 3. 19 : *Sequence Diagram* Report Kehadiran Siswa

14. *Sequence Diagram* Report Tagihan Siswa

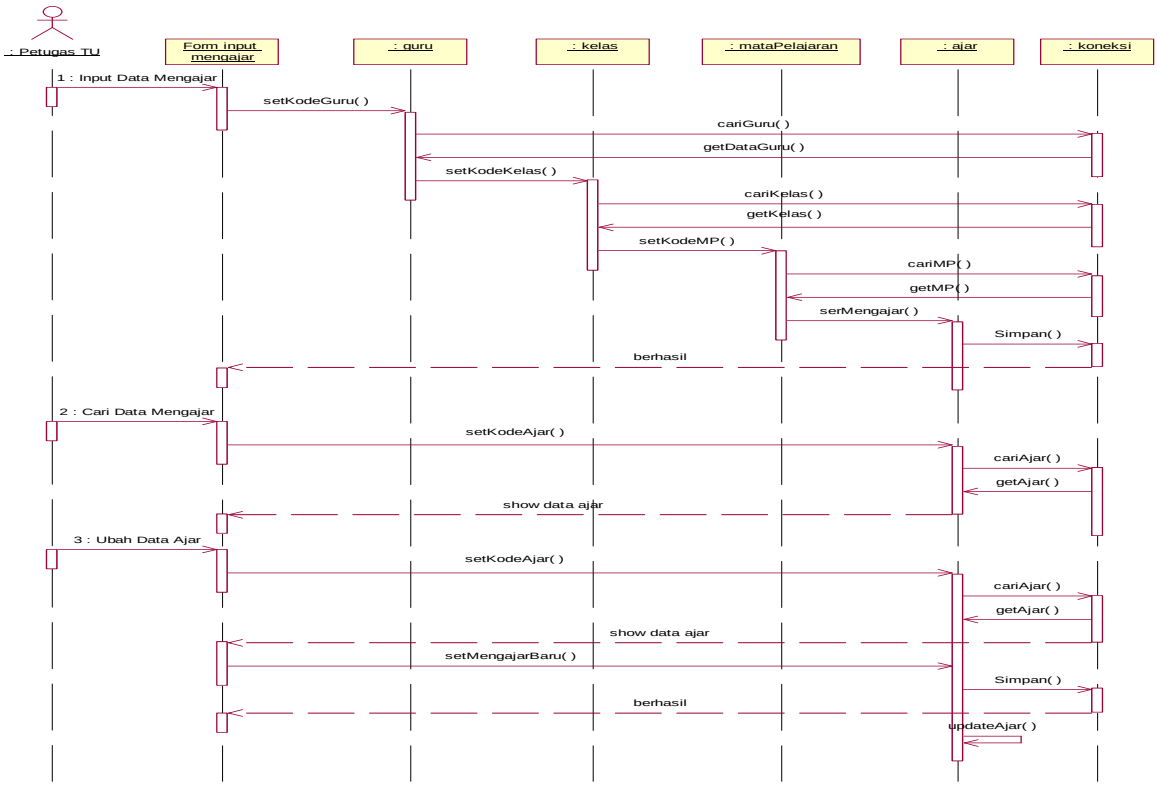
Pada proses ini, dilakukan oleh guru. Guru dapat melihat data tagihan siswa yang ada di SMA Negeri 2 Purwakarta. Dalam hal ini, guru masuk ke menu iuran siswa terlebih dahulu, lalu ke form cari ketagihan siswa.



Gambar 3. 20 : *Sequence Diagram* Report Tagihan Siswa

15. *Sequence Diagram* Pengelolaan Data Mengajar

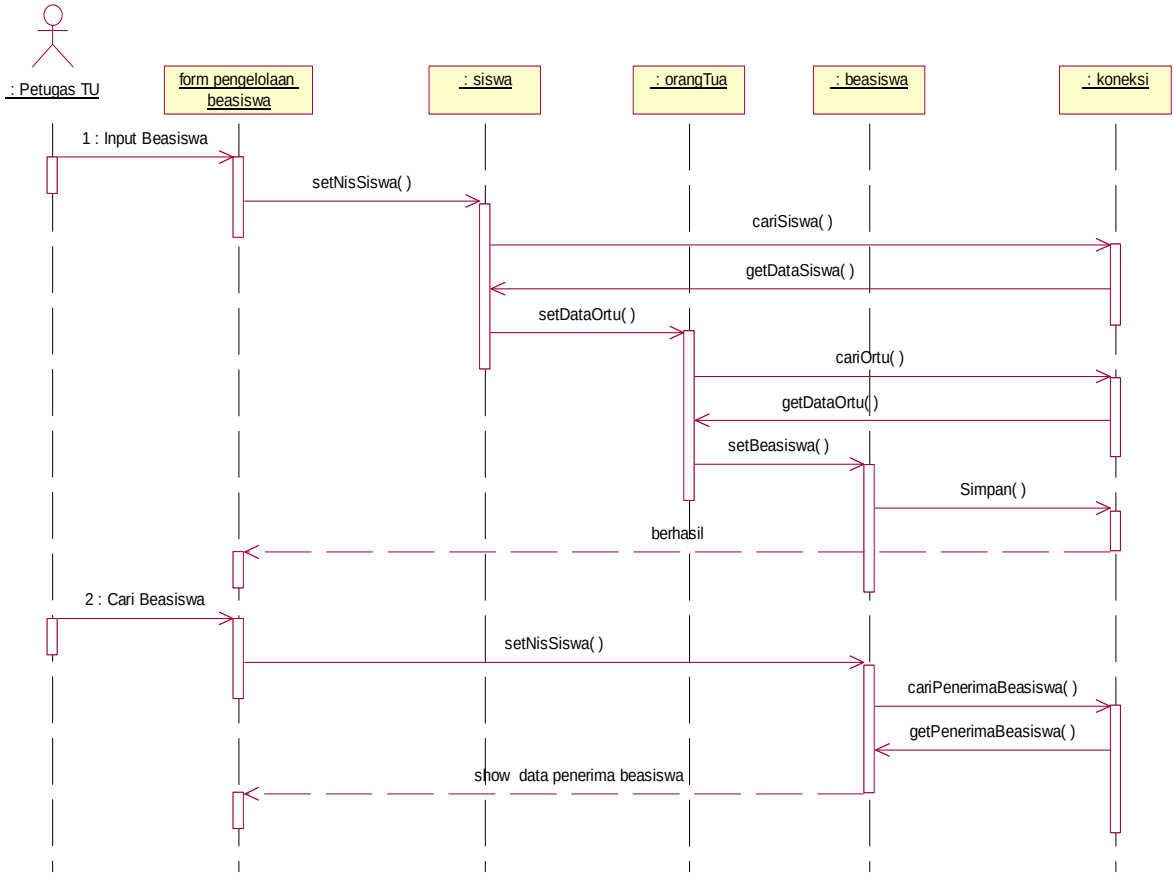
Proses pengelolaan yang dilakukan meliputi *input* data mengajar, cari data mengajar, dan ubah data mengajar. Pada proses input data mengajar, petugas TU masuk ke menu mengajar lalu ke form input mengajar guru. Kemudian cari data guru berdasarkan kode guru, data kelas berdasarkan kode kelas, dan data mata pelajaran berdasarkan kode mata pelajaran. Untuk proses ubah data mengajar dan hapus data mengajar, petugas TU masuk ke form cari data mengajar terlebih dahulu.



Gambar 3. 21 : *Sequence Diagram* Pengelolaan Data Mengajar

16. *Sequence Diagram* Pengelolaan beasiswa

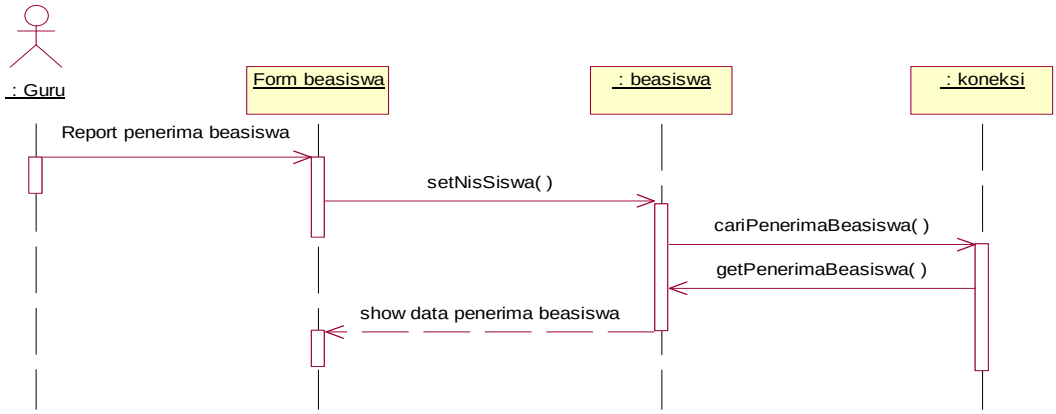
Proses pengelolaan yang dilakukan meliputi *input* data beasiswa dan cari data beasiswa. Pada proses input data beasiswa, petugas TU masuk ke menu beasiswa lalu ke form input beasiswa. Kemudian cari data siswa berdasarkan nis. Untuk proses ubah data beasiswa dan hapus data beasiswa, petugas TU masuk ke form cari beasiswa terlebih dahulu.



Gambar 3. 22 : *Sequence Diagram* Pengelolaan Beasiswa

17. *Sequence Diagram* Report Penerima Beasiswa

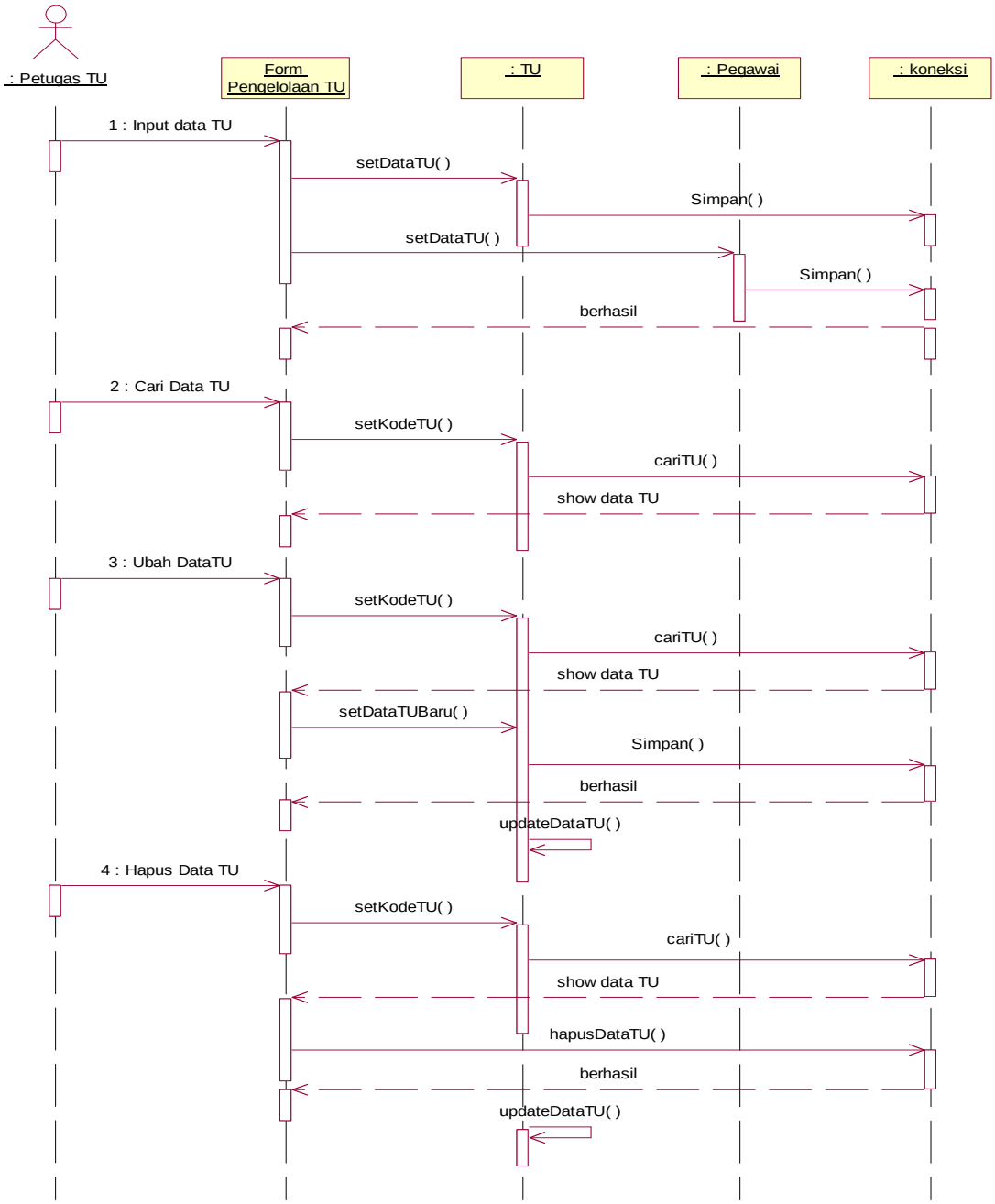
Pada proses ini, dilakukan oleh guru. Guru dapat melihat data penerima beasiswa yang ada di SMA Negeri 2 Purwakarta. Dalam hal ini, guru masuk ke menu beasiswa terlebih dahulu, lalu ke form cari data penerima beasiswa.



Gambar 3. 23 : *Sequence Diagram* Report Penerima Beasiswa

18. *Sequence Diagram* Pengelolaan data TU

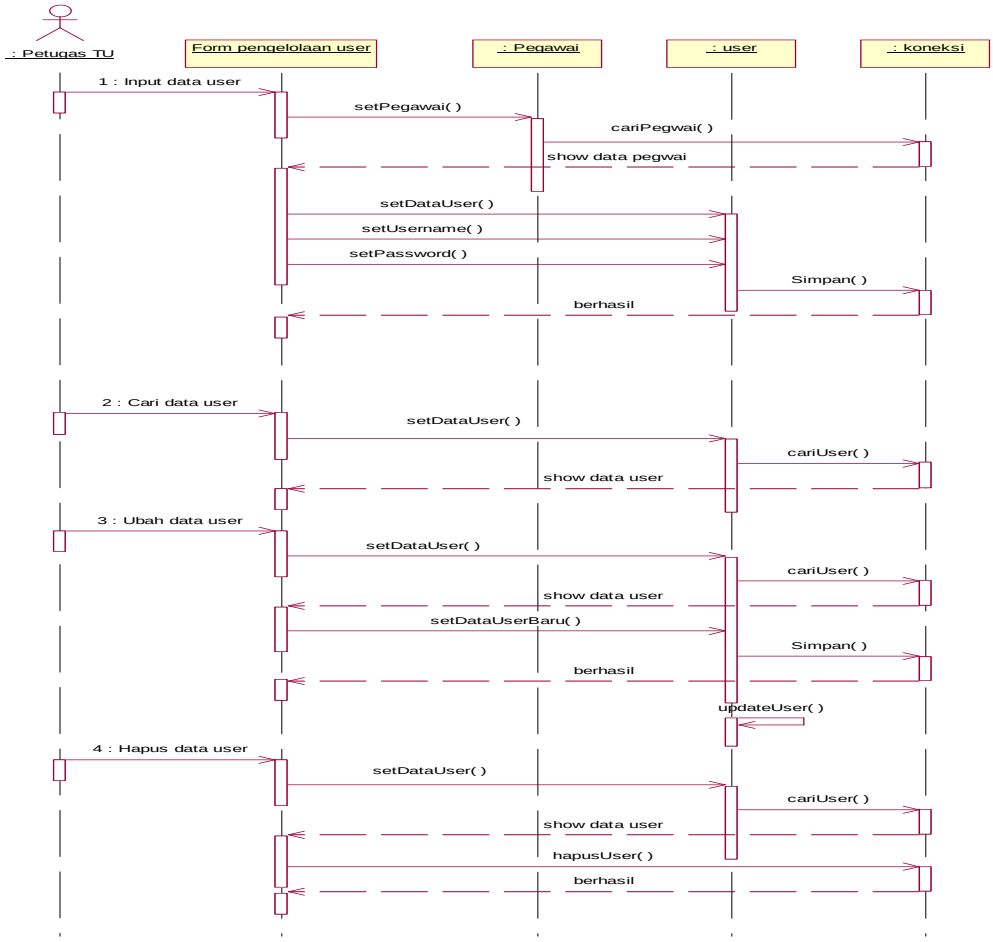
Proses pengelolaan yang dilakukan meliputi *input* data TU, cari data TU, ubah data TU dan hapus data TU. Pada proses input data TU, petugas TU masuk ke menu TU lalu ke form input data TU. Untuk proses ubah data TU, dan hapus data TU, petugas TU masuk ke form cari data TU terlebih dahulu.



Gambar 3. 24 : *Sequence Diagram* Pengelolaan data TU

19. *Sequence Diagram* Pengelolaan data User

Proses pengelolaan yang dilakukan meliputi *input* data user, cari data user, ubah data user dan hapus data user. Pada proses input data user, petugas TU masuk ke menu userr lalu ke form input data user. Untuk proses ubah data user, dan hapus data user, petugas TU masuk ke form cari data user terlebih dahulu.

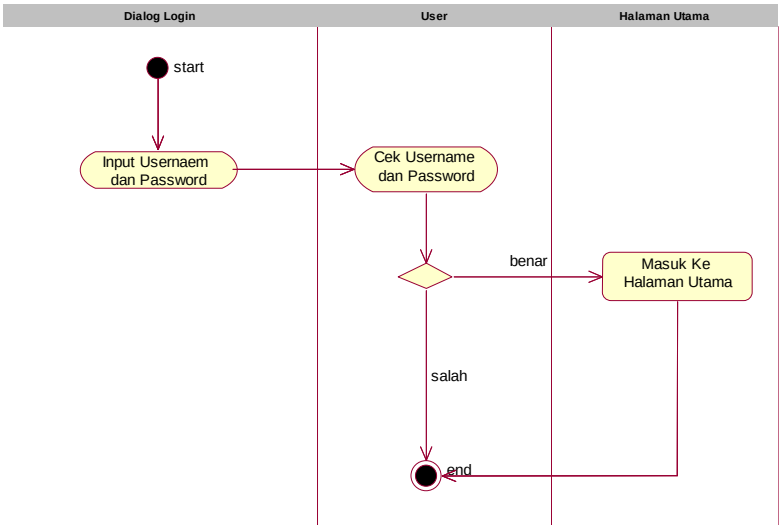


Gambar 3. 25 : *Sequence Diagram* Pengelolaan data User

3.1.4.3. Activity Diagram

Activity diagram memodelkan *workflow* urutan aktifitas dalam sebuah proses.

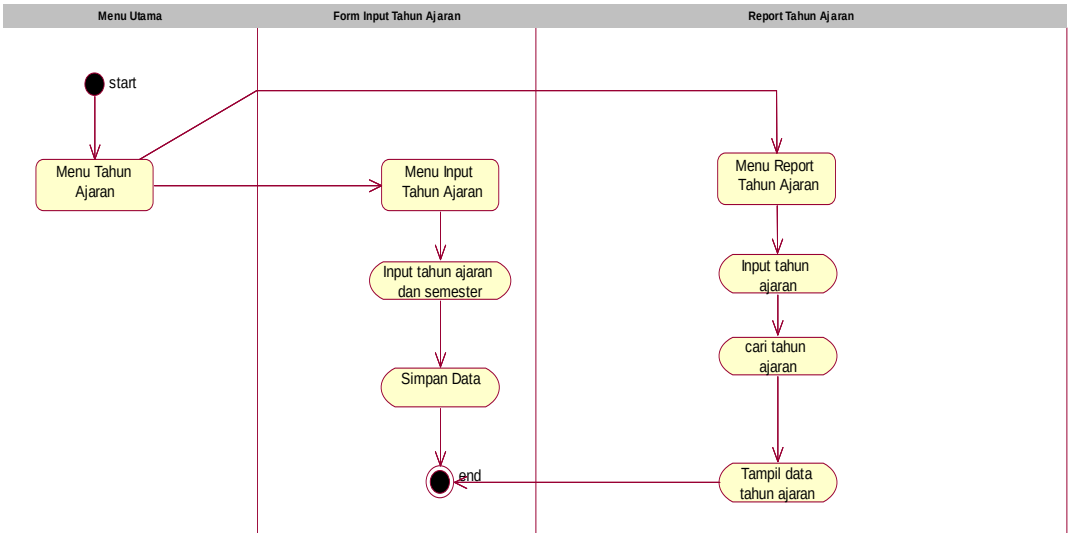
1. *Activity Diagram* Proses *Login*



Gambar 3. 26 : *Activity Diagram* Proses *Login*

2. *Activity Diagram* Pengelolaan Tahun Ajaran
    1. *User* melalukan *login* terlebih dahulu
    2. Kemudian pilih menu tahun ajaran, lalu *form input* tahun ajaran
    3. Kemudian input tahun ajaran, lalu simpan

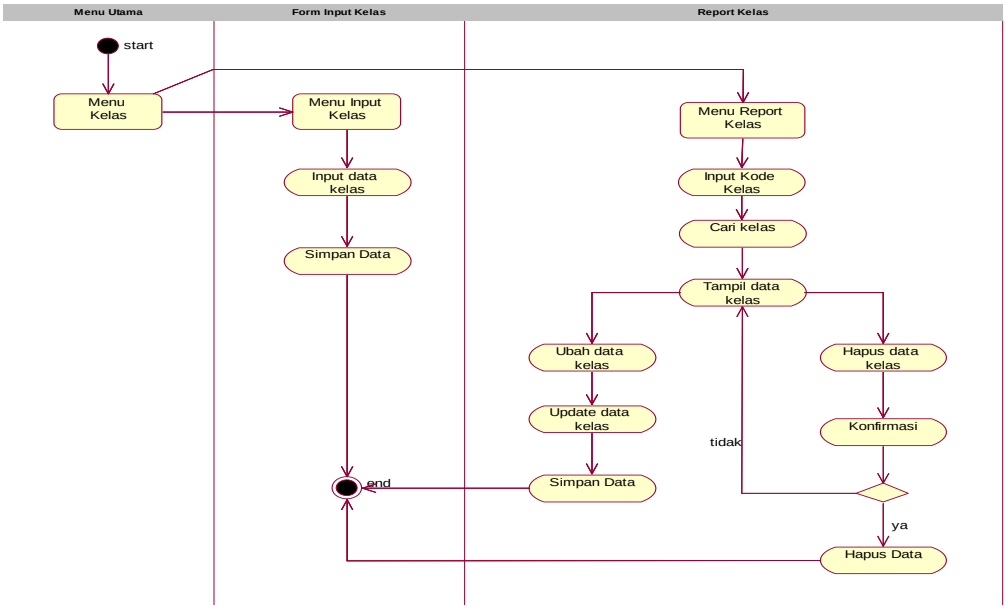
4. Jika ingin mencari data, *user* masuk ke menu report tahun ajaran, kemudian cari tahun ajaran.



Gambar 3. 27 : *Activity Diagram* Pengelolaan Tahun Ajaran

3. *Activity Diagram* Pengelolaan Kelas

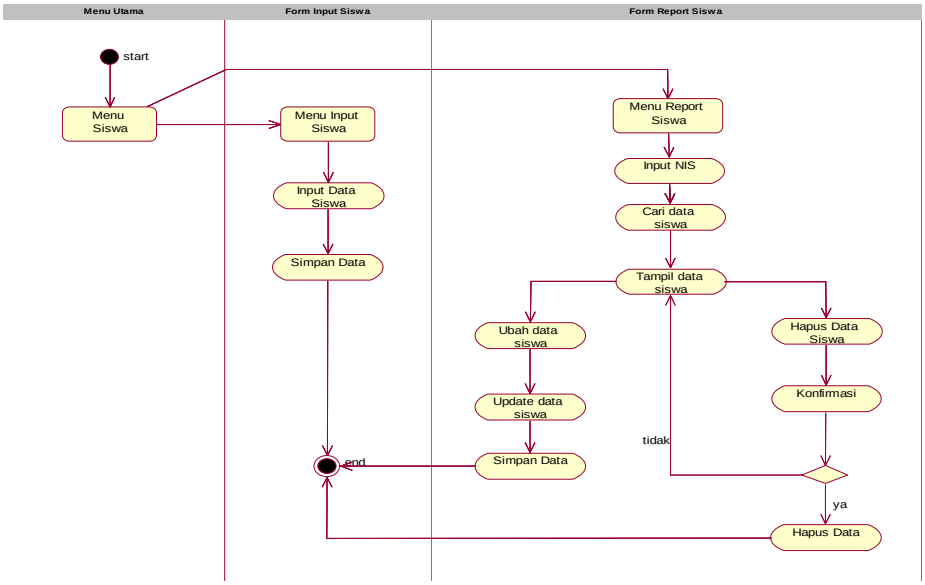
1. *User* melalukan *login* terlebih dahulu
2. Kemudian pilih menu kelas, lalu *form input* kelas
3. Kemudian input data kelas, lalu simpan
4. Jika ingin melakukan manipulasi data (ubah dan hapus), user masuk ke menu report kelas, kemudian cari kelas berdasarkan kode kelas.



Gambar 3. 28 : *Activity Diagram* Pengelolaan Kelas

4. *Activity Diagram* Pengelolaan Data Siswa

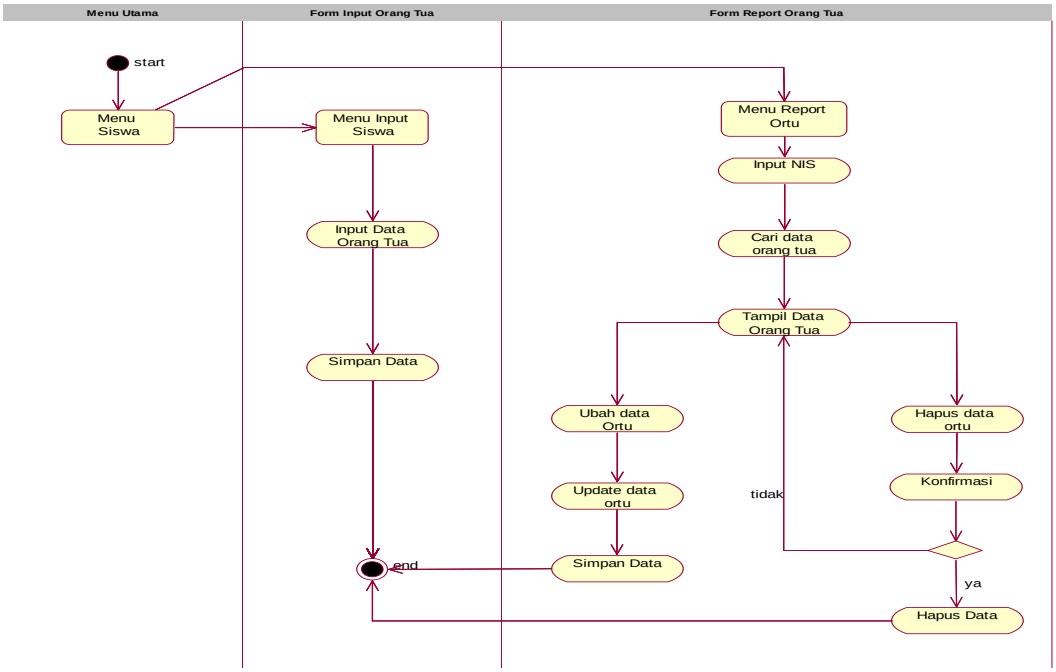
    1. User melalukan *login* terlebih dahulu
    2. Kemudian pilih menu siswa, lalu *form input* siswa
    3. Kemudian input data siswa, lalu simpan
    4. Jika ingin melakukan manipulasi data (ubah dan hapus), user masuk ke menu report siswa, kemudian cari siswa berdasarkan kode nis.



Gambar 3. 30 : *Activity Diagram* Pengelolaan Data Siswa

5. *Activity Diagram* Pengelolaan Data Orang Tua

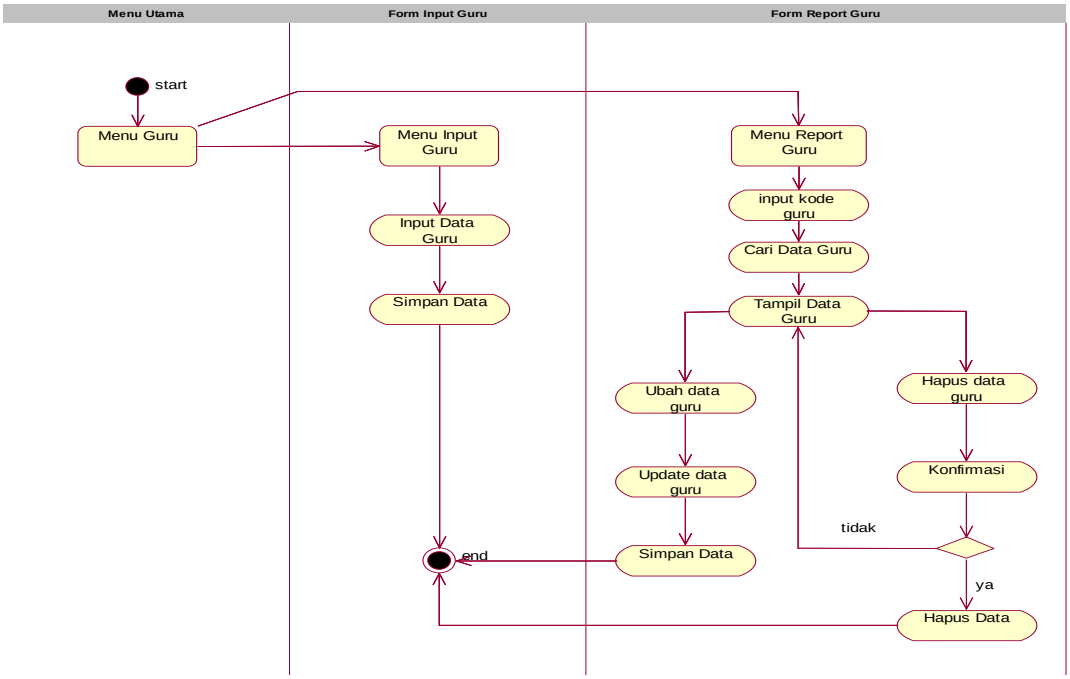
    1. User melalukan *login* terlebih dahulu
    2. Kemudian pilih menu siswa, lalu *form input* siswa
    3. Kemudian input data orangtua,berdasarkan nis siswa lalu simpan
    4. Jika ingin melakukan manipulasi data (ubah dan hapus), user masuk ke menu report siswa, kemudian cari data orang tua berdasarkan nis.



Gambar 3. 31 : *Activity Diagram* Pengelolaan Data Orang Tua

6. *Activity Diagram* Pengelolaan Data Guru

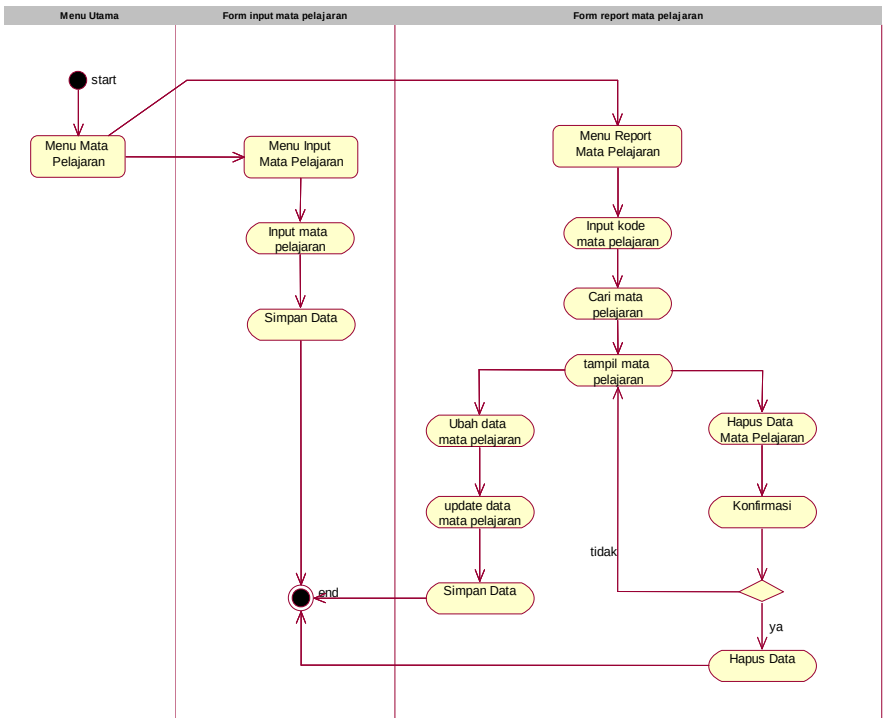
    1. User melalukan *login* terlebih dahulu
    2. Kemudian pilih menu guru, lalu form input guru
    3. Kemudian input data guru, lalu simpan
    4. Jika ingin melakukan manipulasi data (ubah dan hapus), user masuk ke menu report guru, kemudian cari data guru berdasarkan kode guru



Gambar 3. 32 : Activity Diagram Pengelolaan Data Guru

7. *Activity Diagram* Pengelolaan Mata Pelajaran

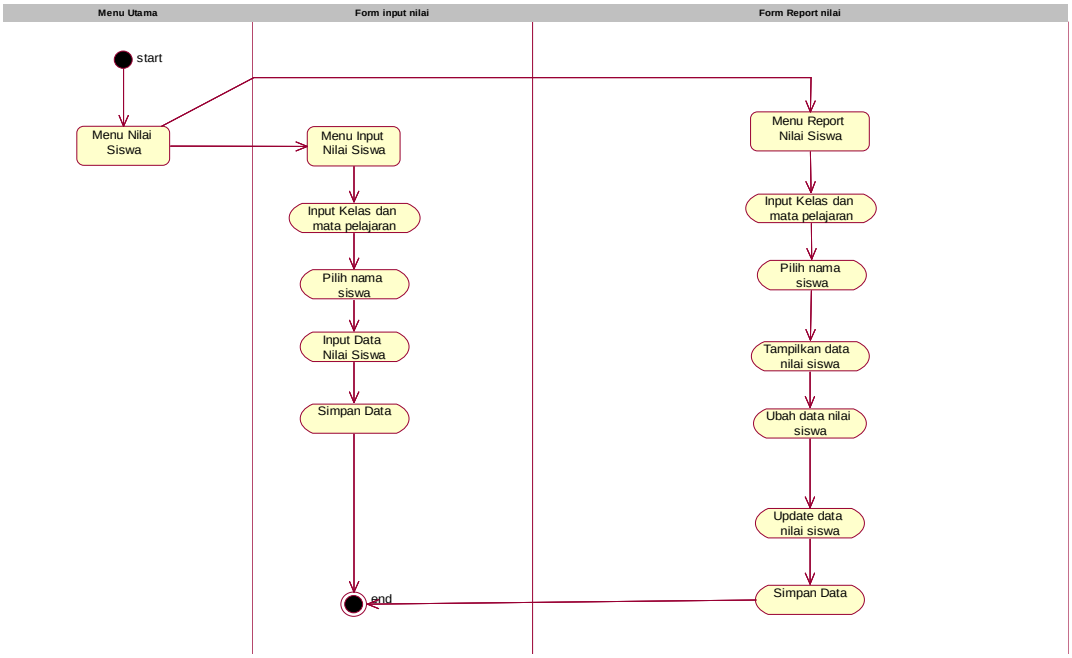
    1. User melalukan *login* terlebih dahulu
    2. Kemudian pilih menu mata pelajaran, lalu *form input* mata pelajaran
    3. Kemudian input data mata pelajarn, lalu simpan
    4. Jika ingin melakukan manipulasi data (ubah dan hapus), user masuk ke menu report mata pelajaran, kemudian cari data mata pelajaran berdasarkan kode mata pelajaran.



Gambar 3. 33 : Activity Diagram Pengelolaan Mata Pelajaran

8. *Activity Diagram* Pengelolaan Nilai Siswa

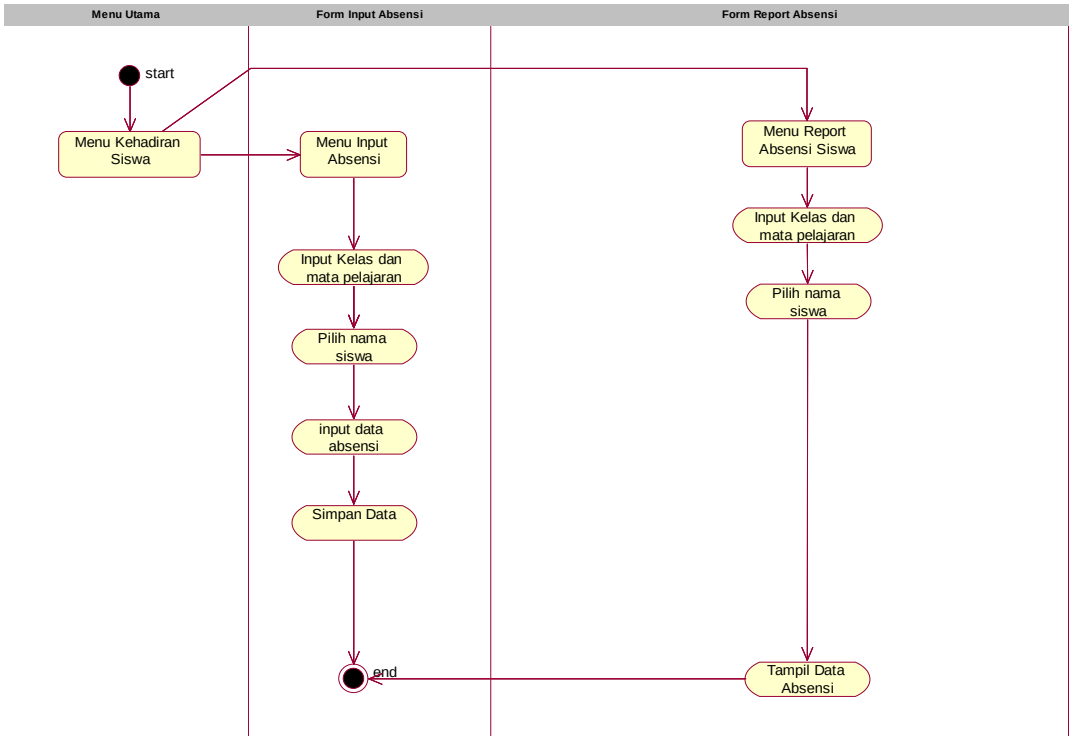
    1. User melalukan *login* terlebih dahulu
    2. Kemudian pilih menu nilai siswa, lalu *form input* nilai siswa
    3. Kemudian cari data siswa berdasarkan nis, lalu input nilai siswa, kemudian simpan
    4. Jika ingin melakukan manipulasi data (ubah), user masuk ke menu report nilai siswa, kemudian cari data nilai siswa berdasarkan kelas / nis siswa.



Gambar 3. 34 : *Activity Diagram* Pengelolaan Nilai Siswa

9. *Activity Diagram* Pengelolaan Kehadiran Siswa

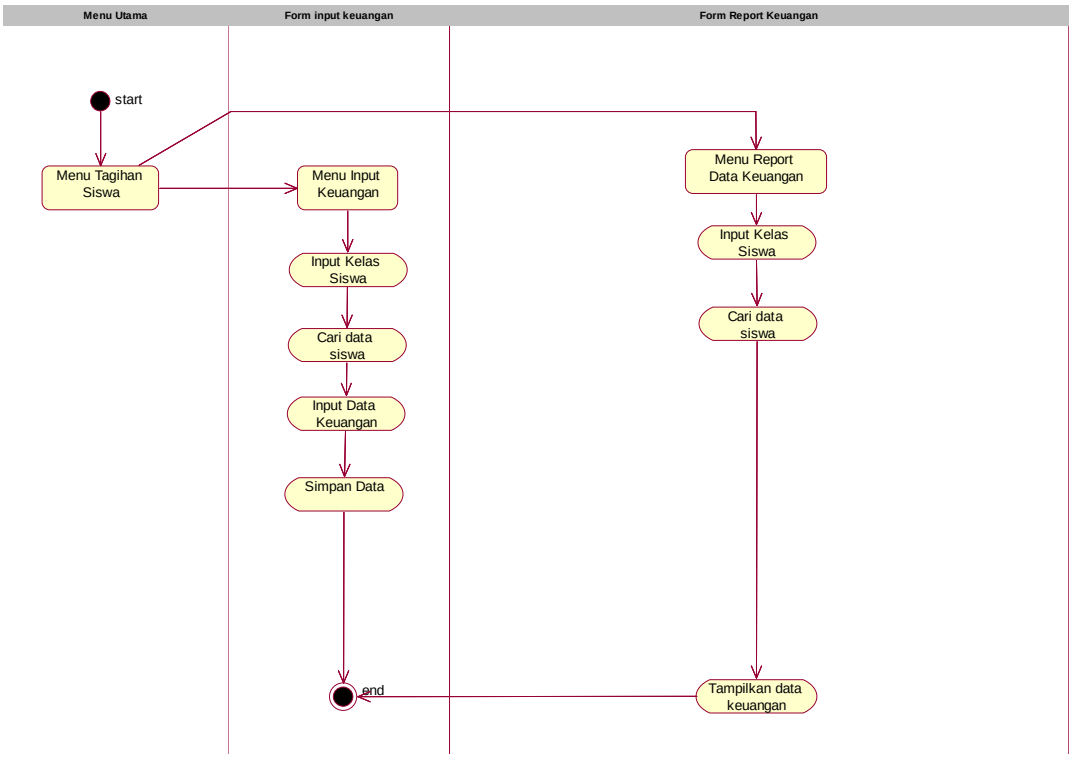
    1. User melalukan login terlebih dahulu
    2. Kemudian pilih menu kehadiran siswa, lalu form input kehadiran siswa
    3. Kemudian cari data siswa berdasarkan nis, lalu input data kehadiran siswa, kemudian simpan
    4. Jika ingin mencari data, user masuk ke menu report kehadiran siswa, kemudian cari data kehadiran siswa berdasarkan nis siswa



Gambar 3. 35 : *Activity Diagram* Pengelolaan Kehadiran Siswa

10. *Acrivity Diagram* Pengelolaan Data Tagihan Siswa

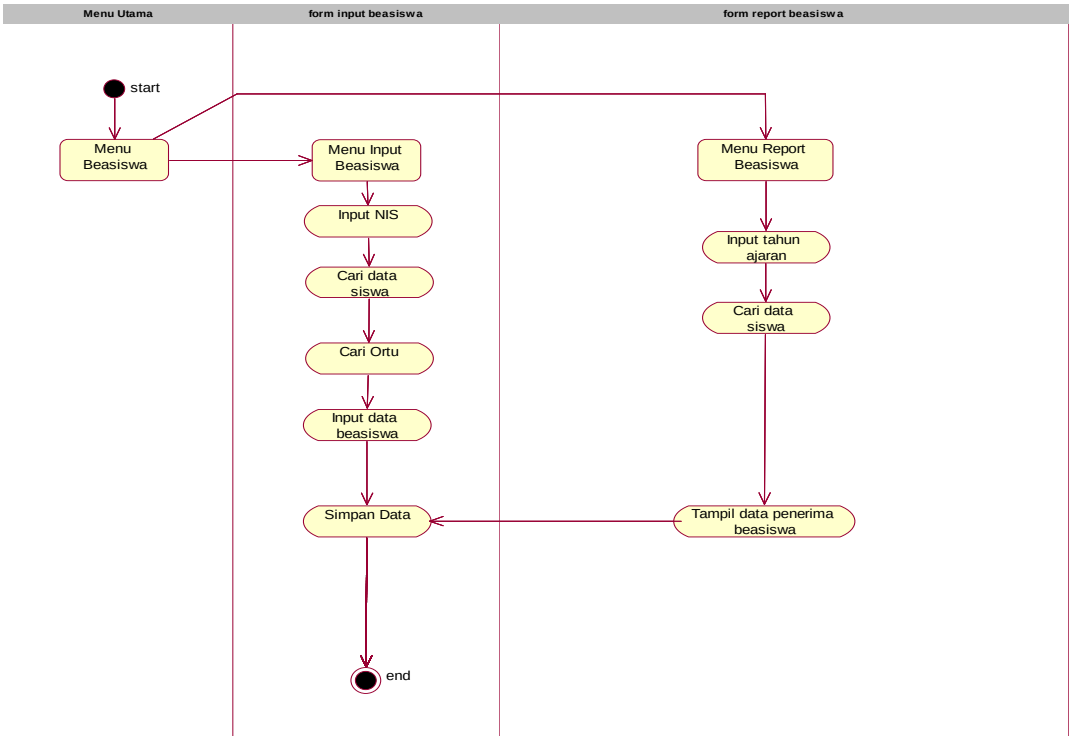
    1. User melalukan *login* terlebih dahulu
    2. Kemudian pilih menu tagihan siswa, lalu *form input* tagihan siswa
    3. Kemudian cari data siswa berdasarkan nis, lalu input data tagihan siswa, kemudian simpan
    4. Jika ingin mencari data user masuk ke menu report tagihan siswa, kemudian cari data kehadiran siswa berdasarkan kelas / nis siswa.



Gambar 3. 36 : *Activity Diagram* Pengelolaan Data Tagihan Siswa

11. *Acrivity Diagram* Pengelolaan Beasiswa

    1. User melalukan *login* terlebih dahulu
    2. Kemudian pilih menu beasiswa, lalu *form input* beasiswa
    3. Kemudian cari data siswa berdasarkan nis dan cari data orang tua, lalu input data beasiswa, kemudian simpan
    4. Jika ingin mencari data, user masuk ke menu report beasiswa, kemudian cari data beasiswa berdasarkan tahun ajaran.

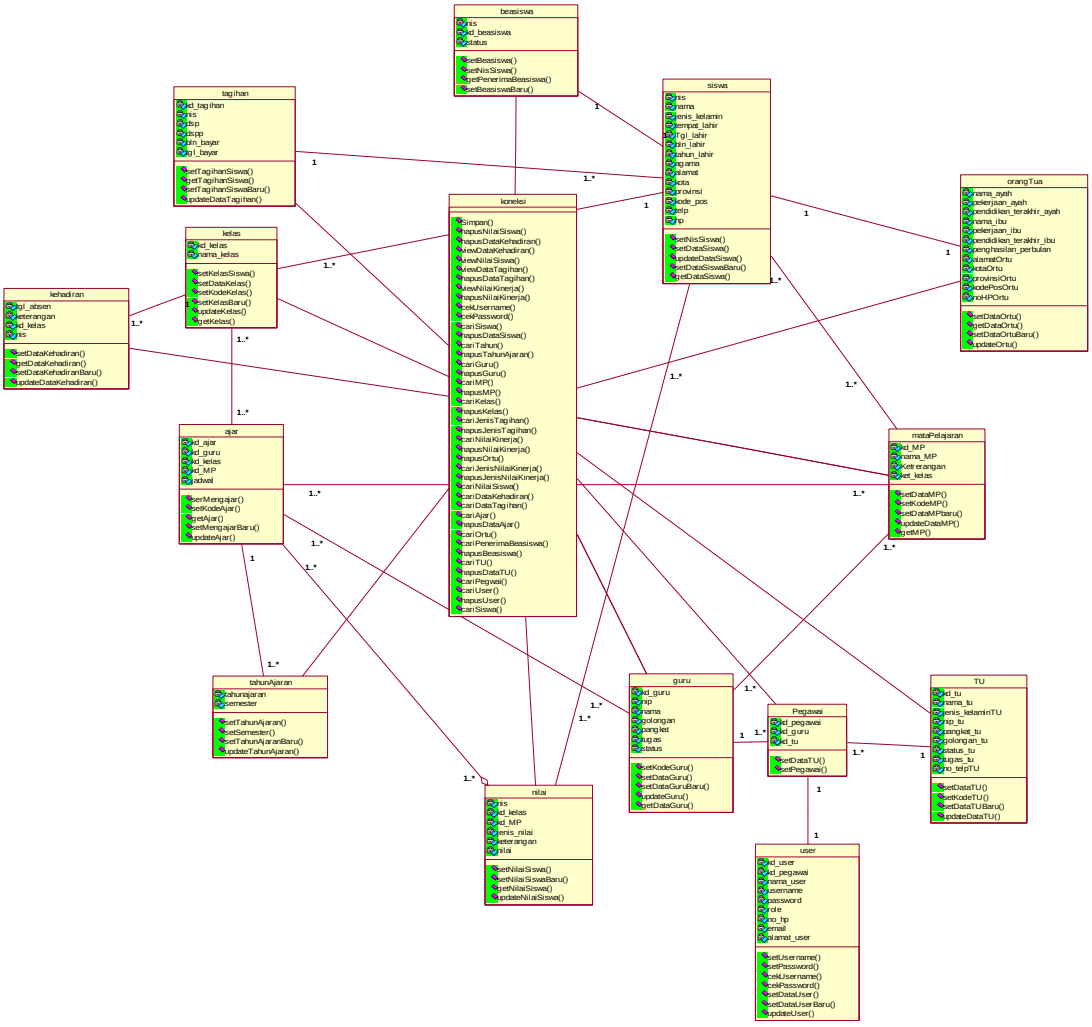


Gambar 3. 37 : *Activity Diagram* Pengelolaan Beasiswa

3.1.4.4. Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut / properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

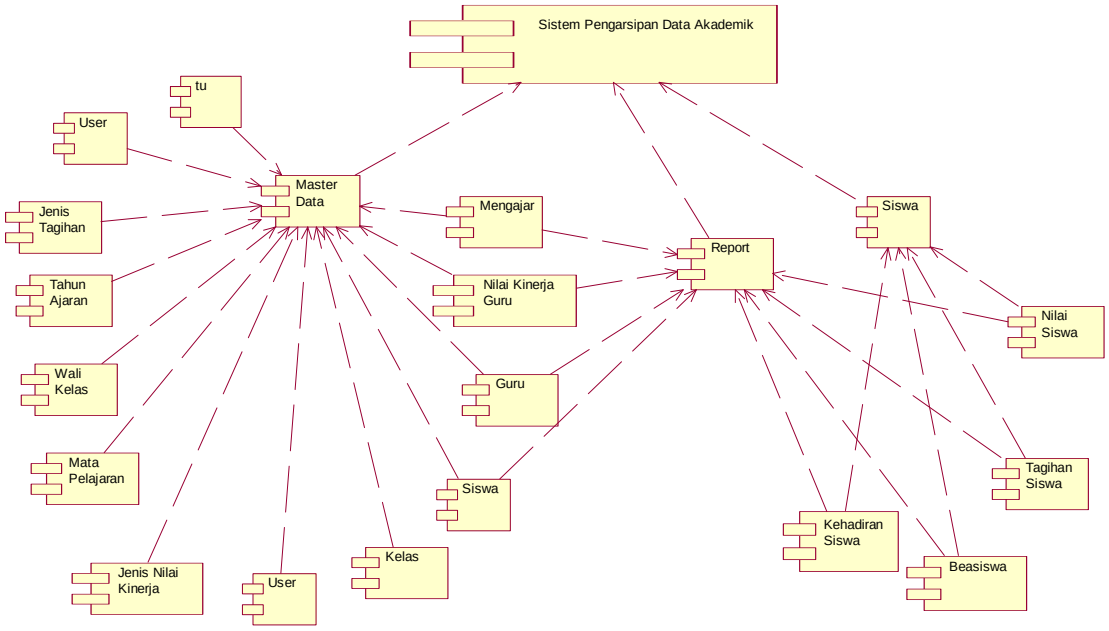
Berikut ini adalah *Class Diagram* dari sistem yang dibangun:



Gambar 3. 38 : *Class Diagram*

3.1.4.5. Component Diagram

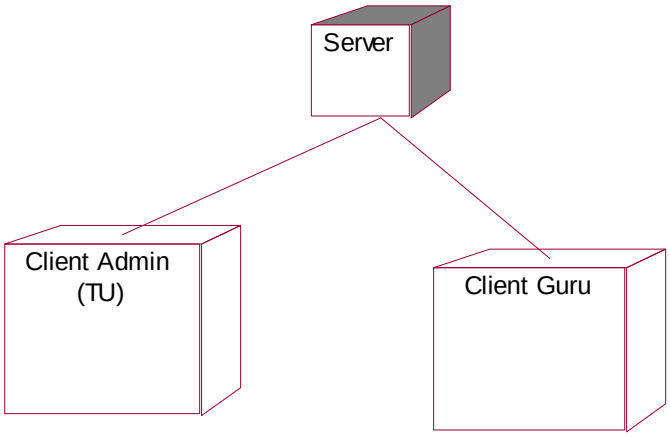
Diagram Komponen (*component diagram)* adalah diagram yang menggambarkan komponen-komponen dalam sistem serta *dependency* antar komponen. Dengan diagram komponen, orang-orang yang bertanggung jawab untuk mengkompilasi dan menyebarkan komponen antar mesin (*deploying*) akan diketahui pustaka kode mana yang sudah ada dan berkas tereksekusi mana yang akan diciptakan saat kode dikompilasi. Adapun Diagram Komponen yang terbentuk adalah sebagai berikut:



Gambar 3. 39 : *Component Diagram*

3.1.4.6. Deployment Diagram

Deployment diagram menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, *server* atau piranti keras), bagaimana kemampuan jaringan pada lokasi tersebut,spesifikasi server, dan hal-hal lain yang bersifat fisikal. Adapun *deployment diagram* yang terbentuk adalah sebagai berikut:



Gambar 3. 40 : *Deployment Diagram*

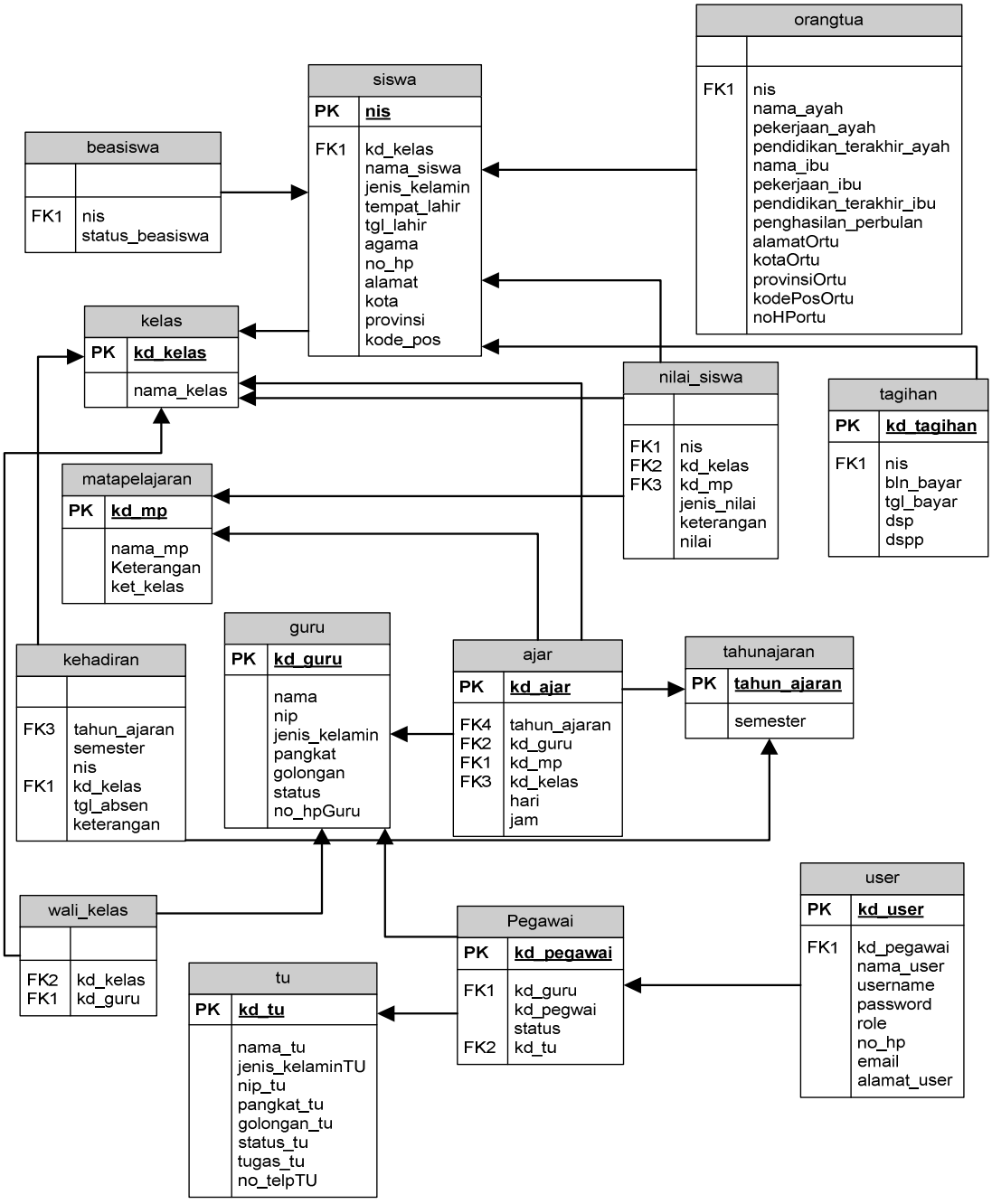
3.2. Perancangan Sistem

3.2.1. Perancangan Basis Data

Perancangan basis data bertujuan untuk mengidentifikasi struktur dari tiap-tiap *file* yang digunakan pada basis data. Pada tahap ini akan dibahas Skema relasi dan struktur tabel.

3.2.1.1. Skema relasi

Skema relasi menggambarkan suatu hubungan antar tabel yang sudah ada dalam keadaan normal. Adapun keterkaitan antar tabel relasi yang digunakan dalam sistem ini dapat digambarkan pada gambar berikut ini :



Gambar 3. 41 : *Skema Relasi*

3.2.1.2. Struktur Tabel

Nama tabel : tahunajaran

Keterangan : Penyimpanan data tahun ajaran

Tabel 3. 1 : Tabel tahunajaran

Nama Field	Type	Size	Keterangan
Tahun_ajaran	varchar	12	Primary key
Semester	varchar	2	

Nama tabel : kelas

Keterangan : Penyimpanan data kelas

Tabel 3. 2 : Tabel kelas

Nama Field	Type	Size	Keterangan
Kd_kelas	varchar	7	Primary key
Nama_kelas	varchar	8	

Nama Tabel : siswa

Keterangan : Penyimpanan data siswa

Tabel 3. 3 : Tabel siswa

Nama Field	Type	Size	Keterangan
nis	varchar	10	Primary key
Kd_kelas	varchar	7	
Nama_siswa	varchar	30	
Jenis_kelamin	varchar	2	
Tempat_lahir	varchar	20	
Tgl_lahir	date		
agama	varchar	10	
No_telp	varchar	15	
No_hp	varchar	15	
alamat	varchar	50	
kota	varchar	20	

provinsi	varchar	30	
Kode_pos	varchar	8	

Nama Tabel : orangtua

Keterangan : Penyimpanan data orang tua

Tabel 3. 4 : Tabel orangtua

Nama Field	Type	Size	Keterangan
nis	varchar	10	Foreign Key
Nama_ayah	varchar	20	
Pekerjaan_ayah	varchar	30	
Pendidikan_terakhir_ayah	varchar	5	
Nama_ibu	varchar	20	
Pekerjaan_ibu	varchar	30	
Pendidikan_terakhir_ibu	varchar	5	
Penghasilan_perbulan	integer		

alamatOrtu	varchar	150	
kotaOrtu	varchar	100	
provinsiOrtu	varchar	100	
kodePosOrtu	varchar	100	
noHPortu	varchar	12	

Nama Tabel : matapelajaran

Keterangan : Penyimpanan data mata pelajaran

Tabel 3. 5 : Tabel matapelajaran

Nama Field	Type	Size	Keterangan
Kd_mp	varchar	5	Primary key
Nama_mp	varchar	25	
Keterangan	varchar	50	
Ket_kelas	varchar	7	

Nama Tabel : guru

Keterangan : Penyimpanan data guru

Tabel 3. 6 : Tabel guru

Nama Field	Type	Size	Keterangan
Kd_guru	varchar	4	Primary key
Nama	varchar	30	
nip	varchar	22	
Jenis_kelamin	varchar	2	
pangkat	varchar	16	
golongan	varchar	5	
status	varchar	5	

Nama Tabel : tu

Keterangan : Penyimpanan data tu

Tabel 3. 7 : Tabel tu

Nama Field	Type	Size	Keterangan
Kd_tu	varchar	4	Primary key
Nama_tu	varchar	30	
Jenis_kelaminTU	varchar	2	
Nip_tu	varchar	22	
Pangkat_tu	varchar	16	
Golongan_tu	varchar	5	
Status_tu	varchar	5	
Tugas_tu	varchar	50	
No_telpTU	varchar	15	

Nama Tabel : pegawai

Keterangan : Penyimpanan data pegawai

Tabel 3. 8 : Tabel pegawai

Nama Field	Type	Size	Keterangan
Kd_pegawai	varchar	5	Primary key
Kd_guru	varchar	4	
Kd_pegawai	varchar	4	
status	varchar	10	

Nama Tabel : user

Keterangan : Penyimpanan data tu

Tabel 3. 9 : Tabel user

Nama Field	Type	Size	Keterangan
Kd_user	varchar	6	Primary key
Kd_pegawai	varchar	6	
Nama_user	varchar	50	

username	varchar	50	
password	varchar	50	
role	varchar	15	
No_hp	varchar	15	
email	varchar	50	
Alamat_user	varchar	50	

Nama Tabel : walikelas

Keterangan : Penyimpanan data wali kelas

Tabel 3. 10 : Tabel walikelas

Nama Field	Type	Size	Keterangan
Kd_guru	varchar	30	Foreign Key
Kd_kelas	varchar	10	Foreign Key

Nama Tabel : ajar

Keterangan : Penyimpanan data mengajar guru

Tabel 3. 11 : Tabel ajar

Nama Field	Type	Size	Keterangan
Kd_ajar	varchar	11	Primary key
Tahun_ajaran	varchar	12	Foreign Key
Kd_guru	varchar	4	Foreign Key
Kd_mp	varchar	5	Foreign Key
Kd_kelas	varchar	7	Foreign Key
jadwal	varchar	20	

Nama Tabel : tagihan

Keterangan : Penyimpanan data tagihan siswa

Tabel 3. 12 : Tabel tagihan

Nama Field	Type	Size	Keterangan
Kd_tagihan	varchar	15	Primary key
nis	varchar	10	Foreign Key
bln_bayar	varchar	15	
Tgl_bayar	date		
dsp	int		
dspp	int		

Nama Tabel : nilai_siswa

Keterangan : Penyimpanan data nilai siswa

Tabel 3. 13 : Tabel nilai_siswa

Nama Field	Type	Size	Keterangan
nis	varchar	10	Foreign Key
Kd_kelas	varchar	7	Foreign Key
Kd_mp	varchar	5	Foreign Key
Jenis_nilai	varchar	10	
keterangan	varchar	2	
nilai	real		

Nama Tabel : kehadiran

Keterangan : Penyimpanan data kehadiran siswa

Tabel 3. 14 : Tabel kehadiran

Nama Field	Type	Size	Keterangan
nis	varchar	10	Foreign Key
Kd_kelas	varchar	7	Foreign Key
Tgl_absen	date		
keterangan	varchar	2	

Nama Tabel : beasiswa

Keterangan : Penyimpanan data penerima beasiswa

Tabel 3. 15 : Tabel beasiswa

Nama Field	Type	Size	Keterangan
nis	varchar	10	Foreign Key
Status_beasiswa	varchar		

3.2.2. Perancangan Kode

1. Tahun ajaran

   Panjang kode : 11 digit

   Tipe kode : String

   Struktur kode : XXXX/YYYY-A

   Perincian :

   XXXX = Menunjukkan tahun 1.

   YYYY = Menunjukkan tahun 2.

   A = Menunjukan semester, yaitu 1 atau 2

   Contoh:

   2010/2011-1 = tahun ajaran 2010/2011 semsester 1.

2. Kode Guru

   Panjang kode : 3 digit

   Tipe kode : String

   Struktur kode : XXX

   Perincian :

   XXX = Menunjukkan singkatan dari nama guru tersebut.

   Contoh:

   GSH = Merupakan kode guru untuk nama H. Gaos Suherman

3. Kode Mata Pelajaran

   Panjang kode : 5 digit

   Tipe kode : String

   Struktur kode : XXX-A

   Perincian :

   XXX = Menunjukkan singkatan dari nama mata pelajaran tersebut tersebut.

   A = Menunjukkan kelas

   Contoh:

   PAI-1 = Merupakan kode mata pelajaran agama islam untuk kelas 1

4. Kode Guru

   Panjang kode : 3 digit

   Tipe kode : String

   Struktur kode : XXX

   Perincian :

   XXX = Menunjukkan singkatan dari nama guru tersebut.

   Contoh:

   GSH = Merupakan kode guru untuk nama H. Gaos Suherman

5. NIS (No Induk Siswa)

   Panjang kode : 8 digit

   Tipe kode : String

   Struktur kode : XXYY1AAA

   Perincian :

   XX = Menunjukkan tahun ajaran masuk

   YY = Menunjukkan tahun ajaran masuk

   AAA = Menunjukkan no urut siswa

   Contoh:

   09101231 = Merupakan NIS siswa yang masuk tahun ajaran 2009/2010 urutan ke 231

6. Kode Tagihan

   Panjang kode : 14 digit

   Tipe kode : String

   Struktur kode : TGH.DMYYYY.AAA

   Perincian :

   D = Menunjukkan tanggal bayar

   M = Menunjukkan bulan bayar

   YYYY = Menunjukkan tahun bayar

   AAA = Menunjukkan urutan bayar

Contoh:

TGH.362012.001 = Merupakan no tagihan yang di bayar siswa pada tanggal 3 juni 2010 urutan ke 1

7. Kode ajar

Panjang kode	: 13 digit
Tipe kode	: String
Struktur kode	: KBMXXXX-Y-AAA
Perincian	:
XXXX	= Menunjukkan tahun ajaran
Y	= Menunjukkan semester
AAA	= Menunjukkan no urut

Contoh:

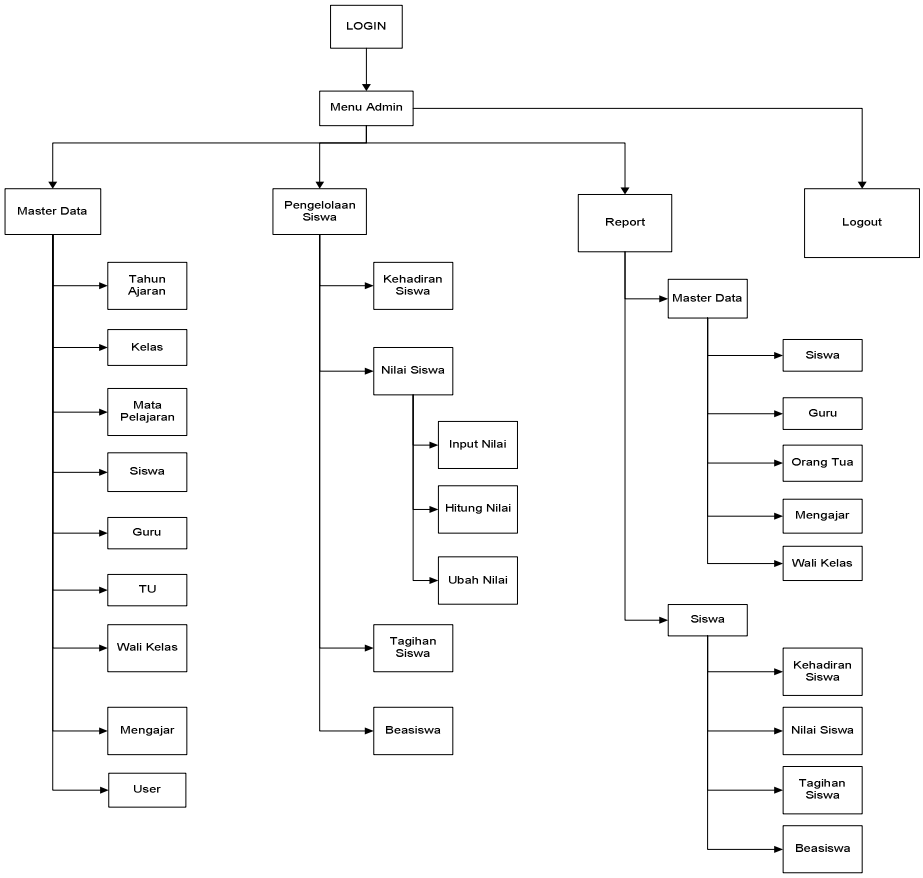
KBM1112-2-001= Merupakan kode mengajar tahun ajaran 2011/2012 semester 2 no 1

3.2.3. Perancangan Struktur Menu

Dalam perancangan sebuah aplikasi dibutuhkan struktur menu yang berisikan menu dan

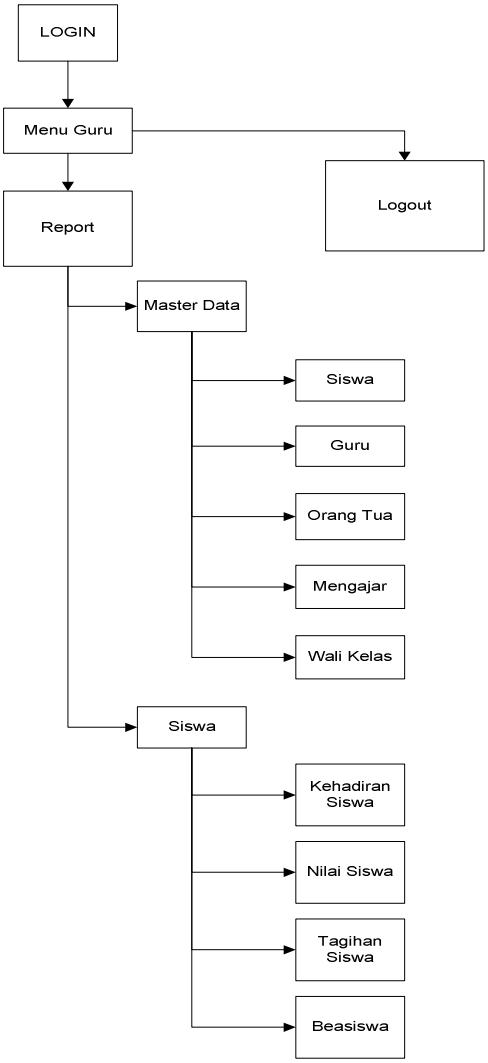
submenu yang berfungsi untuk memudahkan user dalam menggunakan aplikasi tersebut. Berikut ini digambarkan mengenai struktur menu dalam aplikasi ini.

1. Menu Admin (TU)



Gambar 3. 42 : Gambar Struktur Menu TU (Admin)

2. Menu Guru

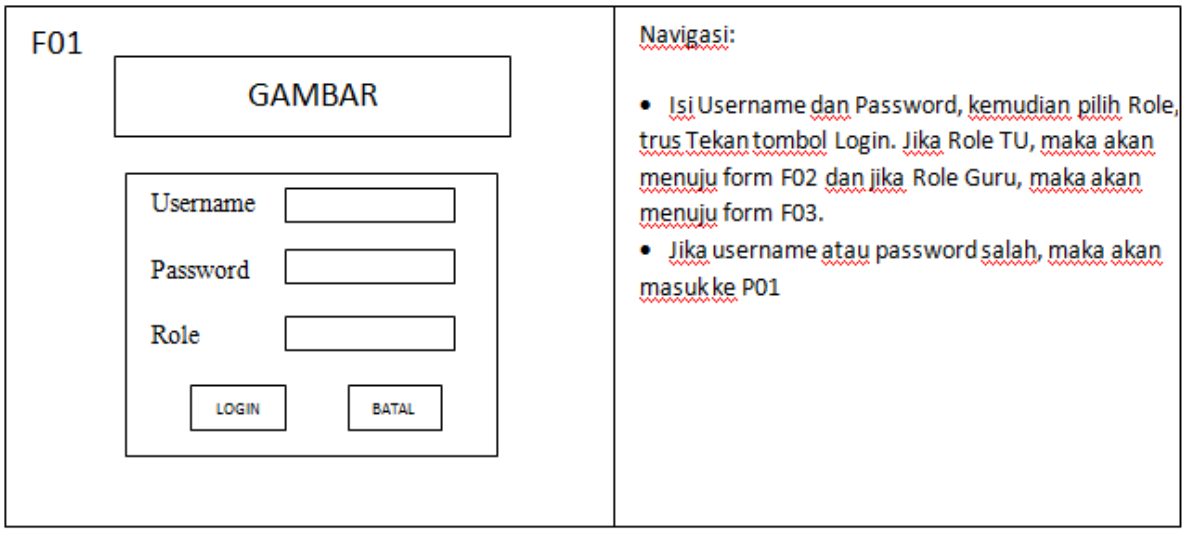


Gambar 3. 43 : Gambar strutur menu guru

3.2.4. Perancangan Antar Muka

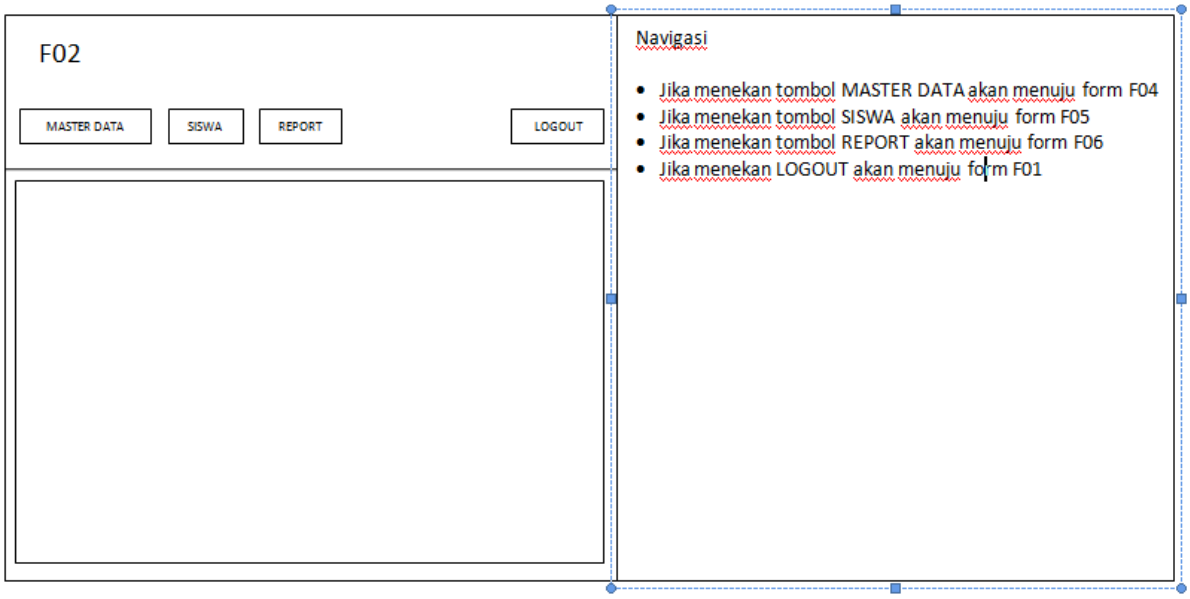
Setelah melakukan perancangan data, maka dibuatlah perancangan antar muka sebagai tempat tampilan data. Perancangan antar muka dilakukan untuk mempermudah untuk mengimplementasikan sistem yang akan dibangun.

1. Login



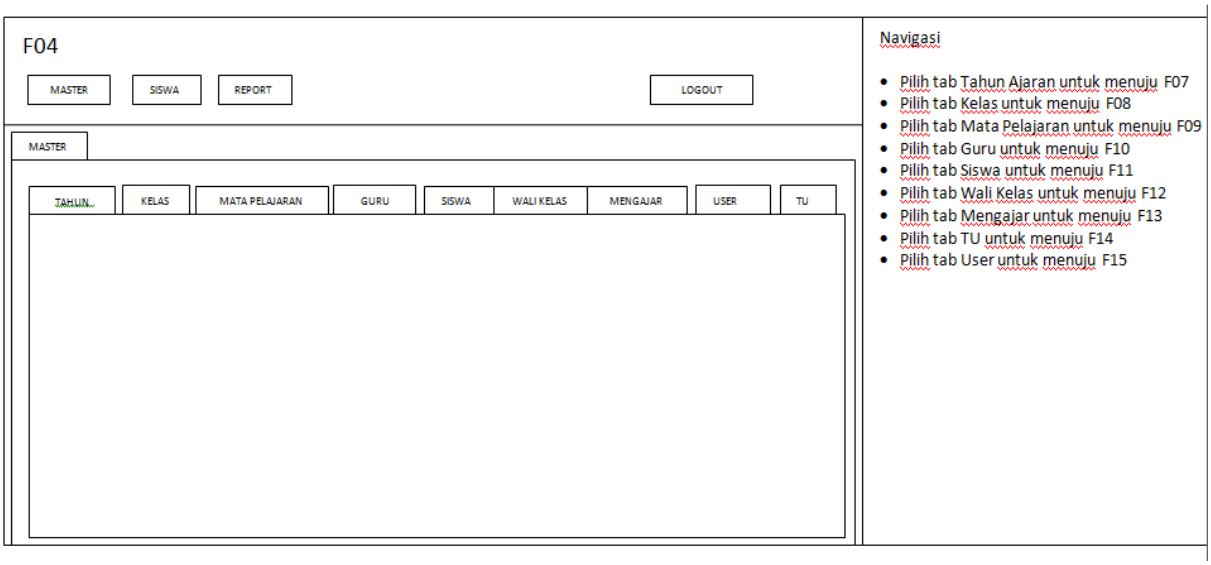
Gambar 3. 44 : Menu login

1. Menu Admin (TU)



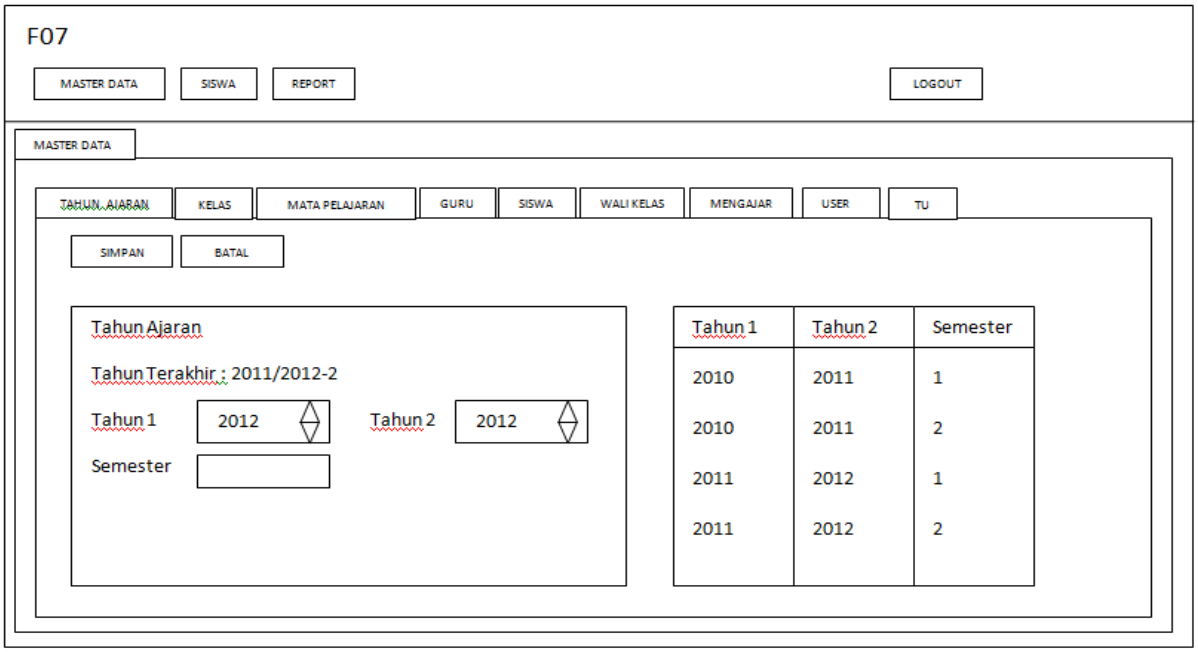
Gambar 3. 45 : Menu Admin (TU)

2. Menu Master data



Gambar 3. 46 : Gambar Menu Master data

3. Tab Tahun Ajaran

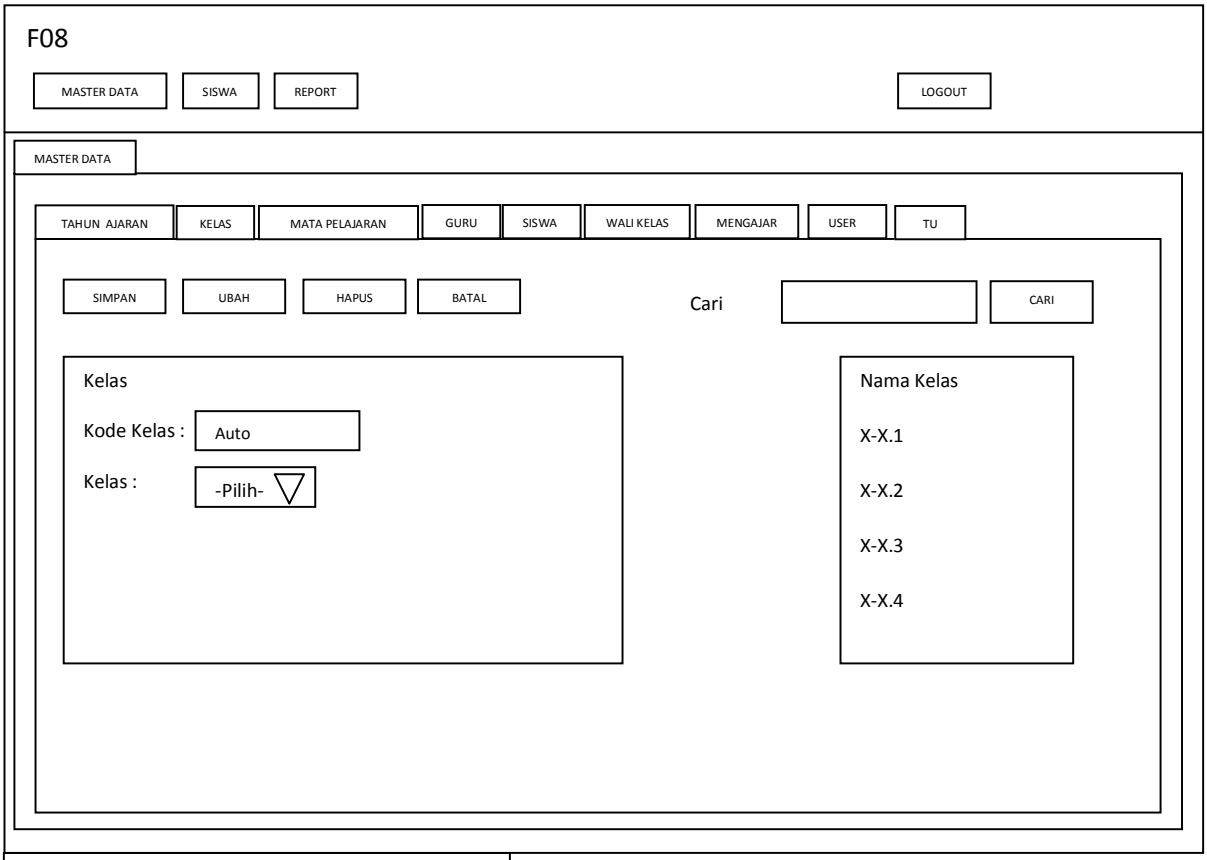


Navigasi

- Pilih tahun 1 dan tahun2, kemudian isi semsester
- Jika Tahun 1 dan tahun 2 sama, akan muncuk P02
- Jika Tahun 1 lebih kecil dari Tahun 2, akan muncul P02
- Jika Kolom semester tidak di isi akan muncul P03
- Jika tekan tombol simpan, akan mucul P04
- Klik tombol batal untuk mengosongkan form inputan

Gambar 3. 47 : Tab Tahun Ajaran

4. Tab kelas

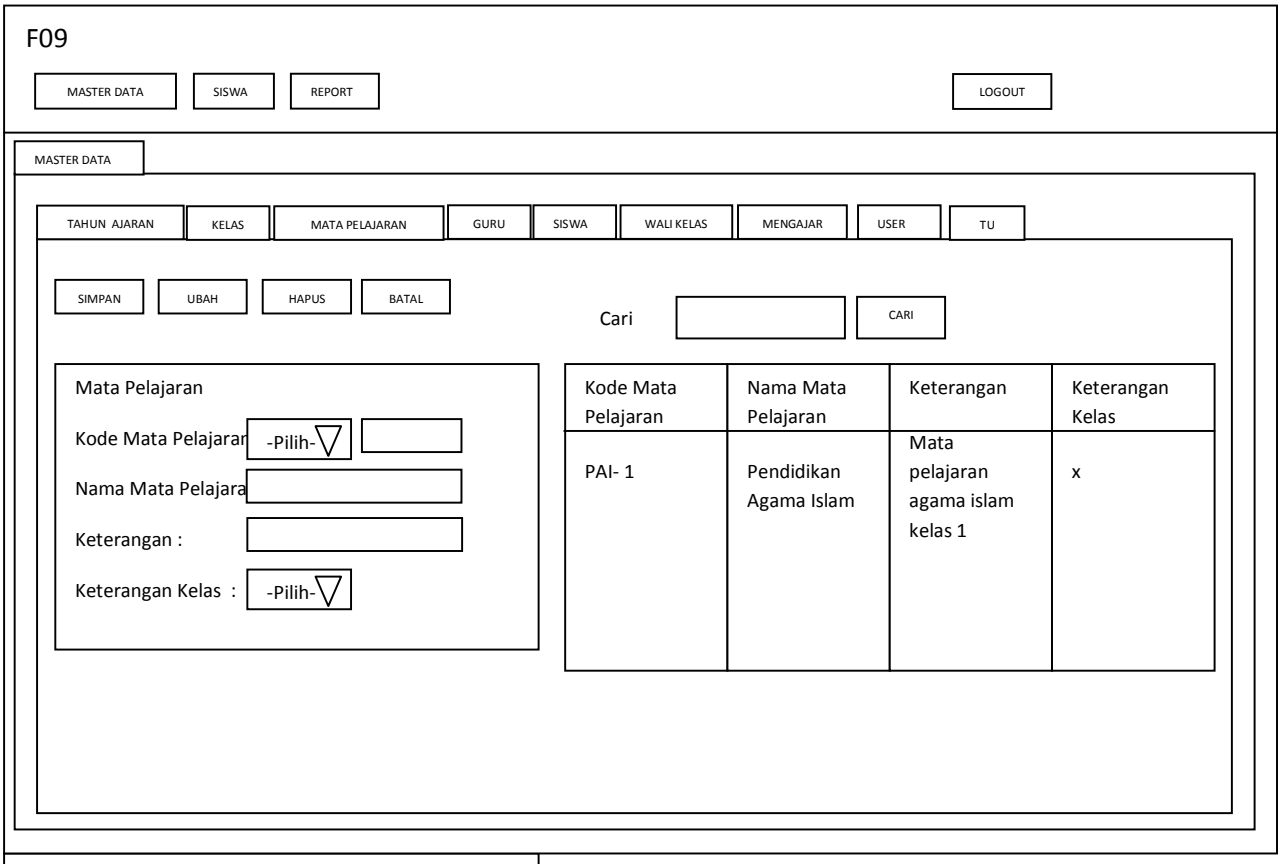


Navigasi

- Pilih kelas, jurusan, dan isi kolom ruangan
- Jika tekan tombol simpan, akan mucul P04
- Klik tombol batal untuk mengosongkan form inputan
- Isi kolom cari untuk mencari data yang diinginkan

Gambar 3. 48 : Tab Kelas

5. Tab Mata Pelajaran

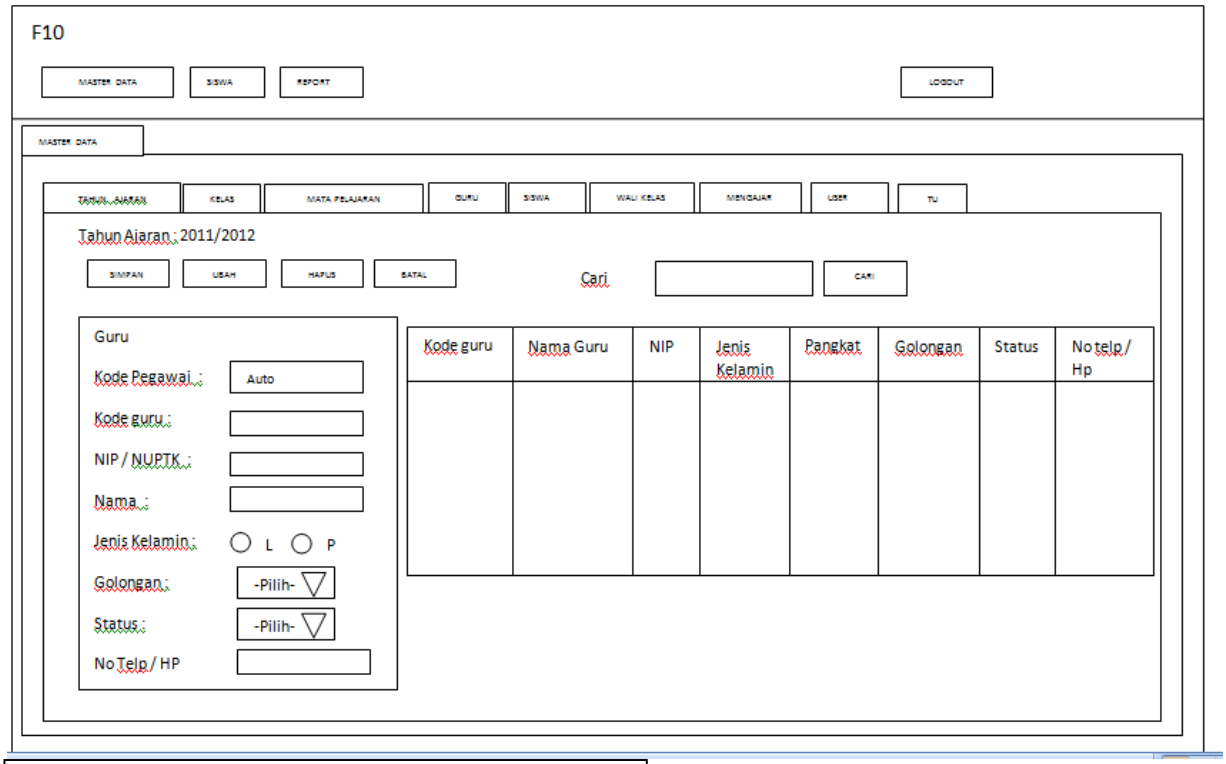


Navigasi

- Isi kode mata pelajaran, nama mata pelajaran, keterangan dan pilih keterangan kelas.
- Jika kode mata pelajaran belum diisi, akan muncul P05
- Jika nama mata pelajaran belum diisi, akan muncul P06
- Jika Keterangan belum diisi, akan muncul P07
- Jika Keterangan Kelas belum dipilih, akan muncul P08
- Jika tekan tombol simpan, akan mucul P04
- Klik tombol batal untuk mengosongkan form inputan
- Isi kolom cari untuk mencari data yang diinginkan
- Klik salah satu data di tabel, untuk mengapus atau mengu data
- Jika tekan tombol ubah, akan muncul P09
- Jika tekan tombol hapus, akan muncul P10

Gambar 3. 49 : Tab Mata Pelajaran

6. Tab Guru

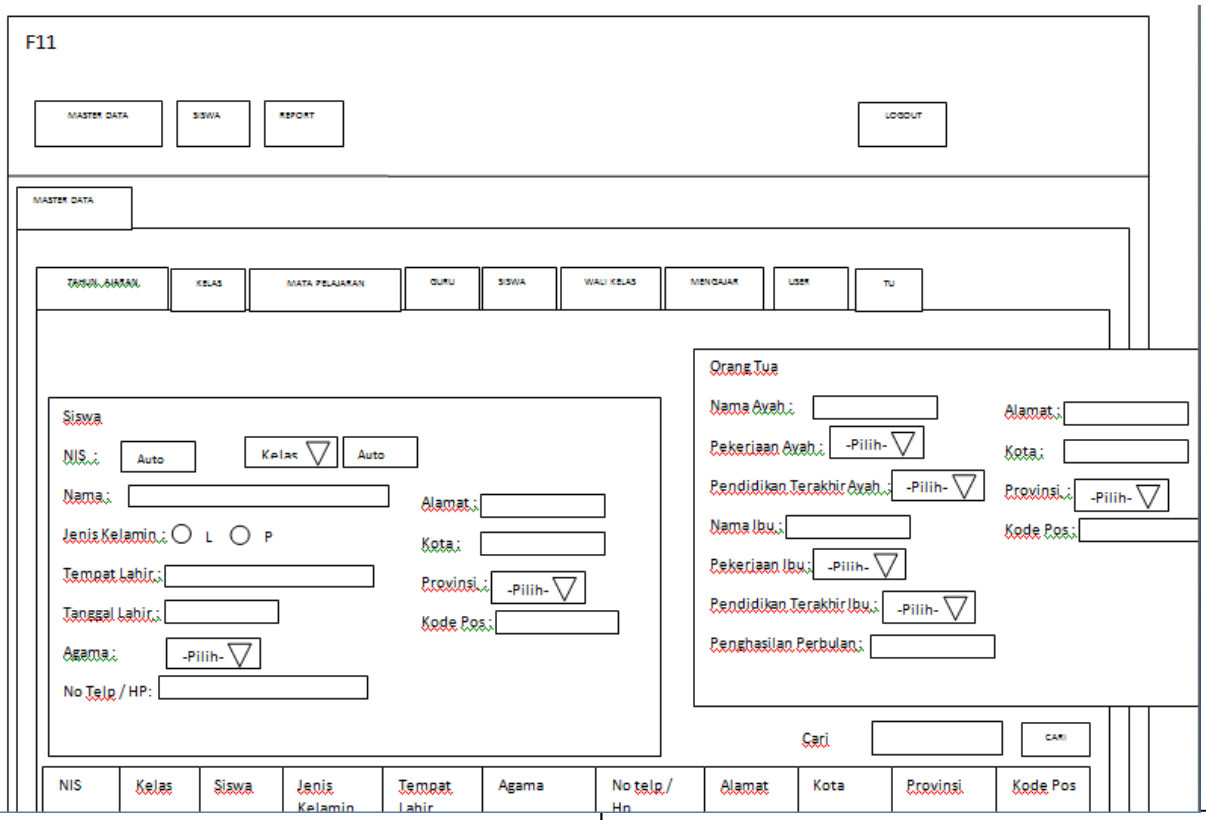


Navigasi

- Isi kode guru, NIP/ NUPTK, Nama, No Telp / HP dan pilih Jenis Kelamin, golongan, status.
- Jika kode guru belum diisi, akan muncul P11
- Jika NIP/ NUPTK belum, akan muncul P12
- Jika Nama belum diisi, akan muncul P13
- Jika Golongan belum dipilih, akan muncul P14
- Jika status belum dipilih, akan muncul P15
- Jika no telp / hp belum diisi, akan muncul P16
- Jika tekan tombol simpan, akan mucul P04
- Klik tombol batal untuk mengosongkan form inputan
- Isi kolom cari untuk mencari data yang diinginkan
- Klik salah satu data di tabel, untuk mengapus atau mengubah data
- Jika tekan tombol ubah, akan muncul P09
- Jika tekan tombol hapus, akan muncul P10

Gambar 3. 50 : Gambar Tab Guru

7. Tab Siswa

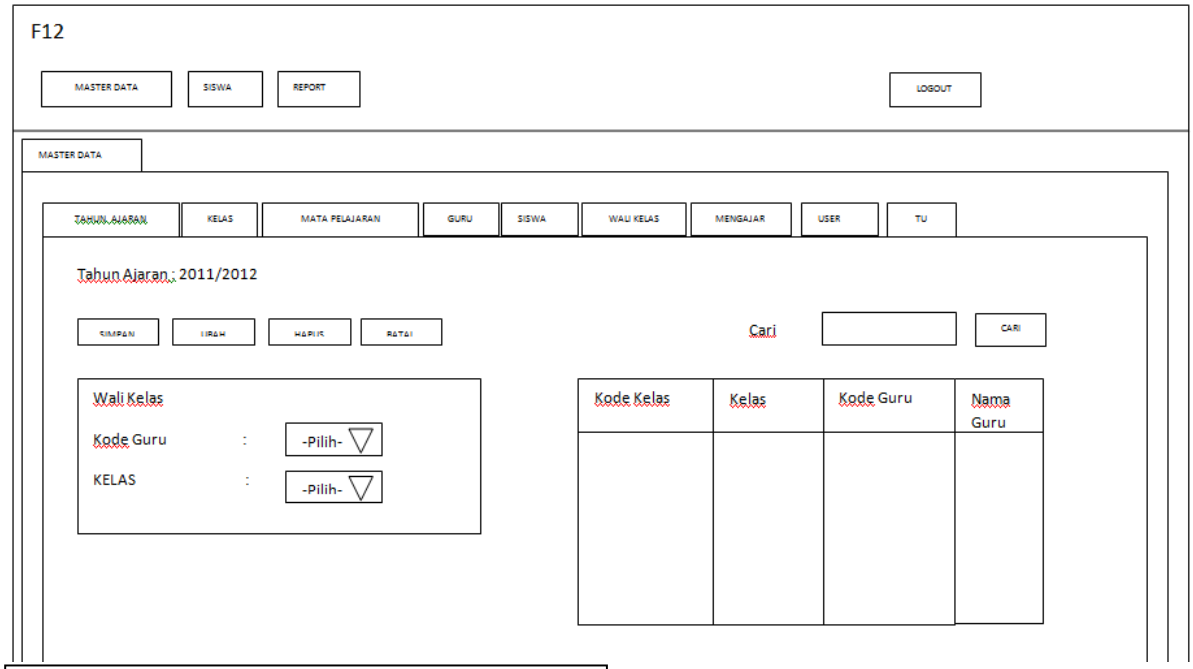


Navigasi

- Isi nis, nama, tempat lahir, tanggal lahir, no telp / hp, nama ayah, nama ibu, alamat, kota, kode pos, dan pilih kelas, jenis kelamin, agama, pekerjaan ayah, pendidikan terakhir ayah, pekerjaan ibu, pendidikan terakhir ibu, provinsi.
- Jika nis belum diisi, akan muncul P17
- Jika nama belum diisi, akan muncul P18
- Jika tempat lahir belum diisi, akan muncul P19
- Jika tanggal lahir belum diisi, akan muncul P20
- Jika Agama belum dipilih, akan muncul P21
- Jika no telp / hp belum diisi, akan muncul P22
- Jika nama ayah belum diisi, akan muncul P23
- Jika Pekerjaan ayah belum dipilih, akan muncul P24
- Jika Pendidikan terakhir ayah belum dipilih, akan muncul P25
- Jika nama ibu belum diisi, akan muncul P26
- Jika Pekerjaan ibu belum dipilih, akan muncul P27
- Jika Pendidikan terakhir ibu belum dipilih, akan muncul P28
- Jika penghasilan perbulan belum diisi, akan muncul P29
- Jika alamat belum diisi, akan muncul P30
- Jika kota belum diisi, akan muncul P31
- Jika Provinsi belum dipilih, akan muncul P32
- Jika kode pos belum diisi, akan muncul P33
- Jika tekan tombol simpan, akan mucul P04
- Klik tombol batal untuk mengosongkan form inputan
- Isi kolom cari untuk mencari data yang diinginkan
- Klik salah satu data di tabel, untuk mengapus atau mengubah data
- Jika tekan tombol ubah, akan muncul P09
- Jika tekan tombol hapus, akan muncul P10

Gambar 3. 51 : Tab siswa

8. Tab Wali Kelas

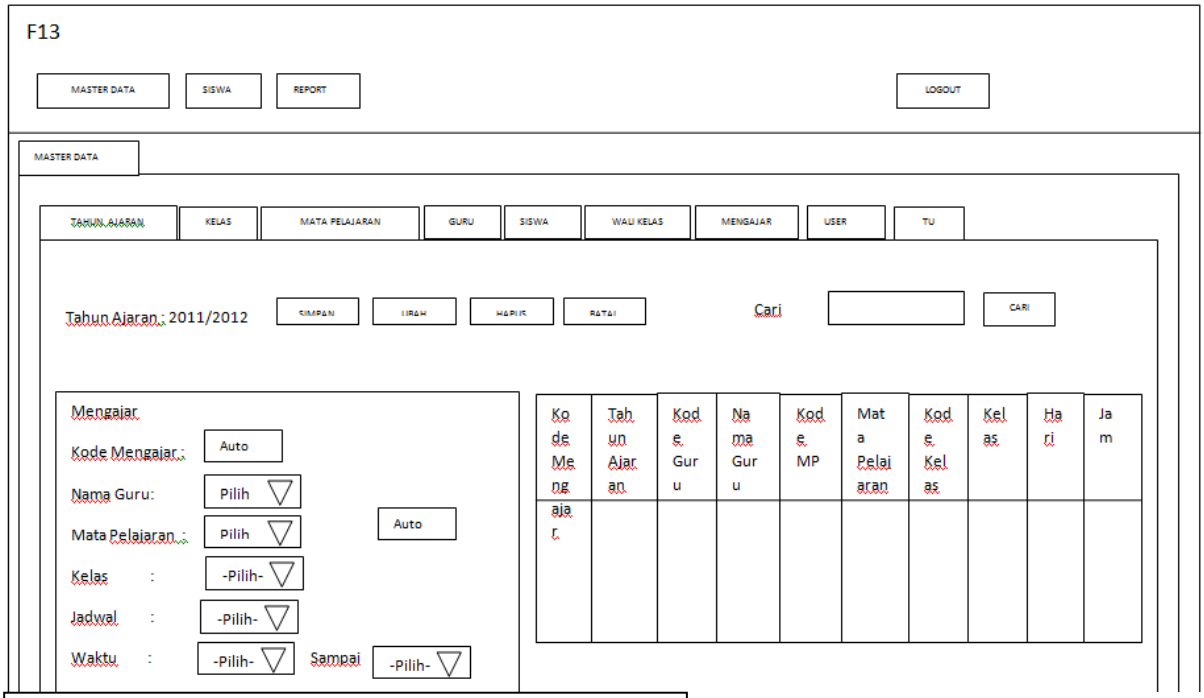


Navigasi

- Pilih kode guru dan kode kelas
- Jika kode guru belum dipilih maka muncul P34
- Jika Kelas belum dipilih maka muncul P35
- Isi kode guru untuk mencari
- Jika ditekan tombol simpan maka muncul P04
- Jika tombol update ditekan maka akan muncul P09
- Jika tombol hapus ditekan maka muncul P10
- Jika ditekan tombol batal, maka akan kembali seperti semula

Gambar 3. 52 Tab Wali Kelas

9. Tab Mengajar

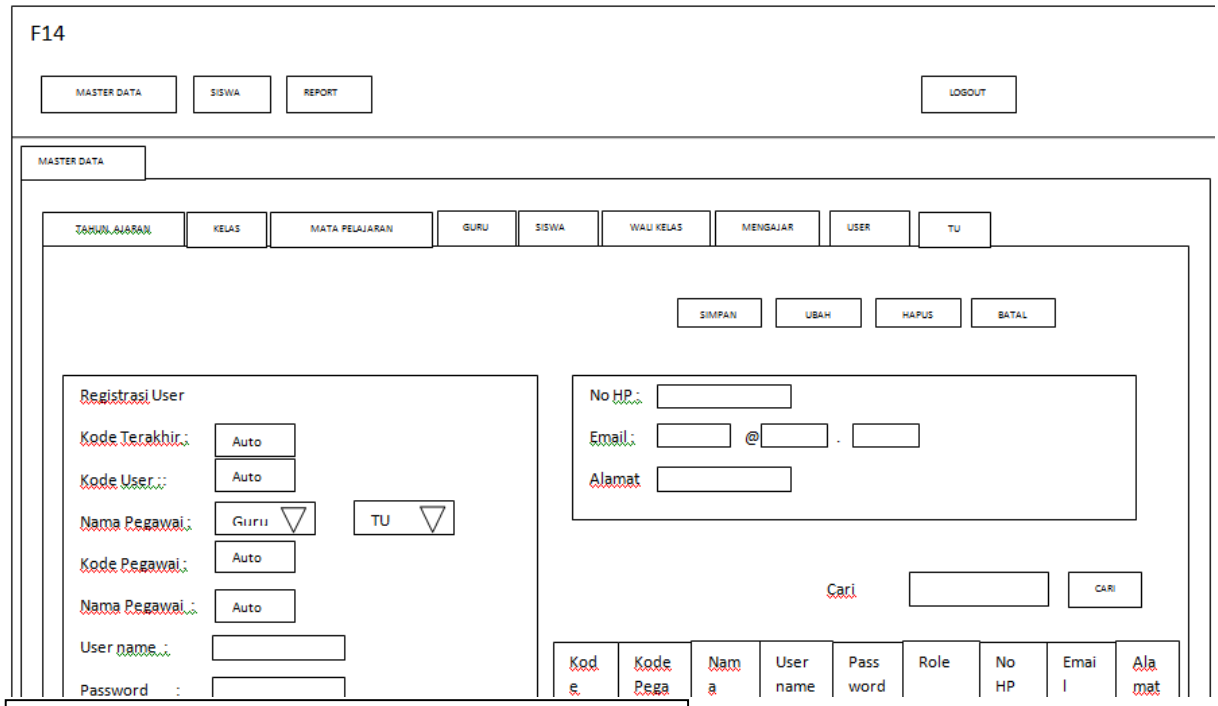


Navigasi

- Pilih kode mengajar, nama guru, mata pelajaran, kelas, jadwal, dan waktu
- Jika nama guru belum dipilih maka muncul P36
- Jika Mata Pelajaran belum dipilih maka muncul P37
- Jika Kelas belum dipilih maka muncul P38
- Jika Jadwal belum dipilih maka muncul P39
- Jika Waktu belum dipilih maka muncul P40
- Isi kode guru untuk mencari
- Jika ditekan tombol simpan maka muncul P04
- Jika tombol update ditekan maka akan muncul P09
- Jika ditekan tombol batal, maka akan kembali seperti semula

Gambar 3. 53 : Tab Mengajar

10. Tab User



Navigasi

- Pilih nama pegawai, username, password, re-password, no hp, email, alama
- Jika nama pegawai belum dipilih maka muncul P41
- Jika username belum diisi maka muncul P42
- Jika password belum diisi maka muncul P43
- Jika re-password belum diisi maka muncul P44
- Jika no hp belum diisi maka muncul P45
- Jika alamat belum diisi maka muncul P46
- Isi kode user untuk mencari
- Jika ditekan tombol simpan maka muncul P04
- Jika tombol update ditekan maka akan muncul P09
- Jika tombol hapus ditekan maka muncul P10
- Jika ditekan tombol batal, maka akan kembali seperti semula

Gambar 3. 54 : Tab User

11. Tab TU

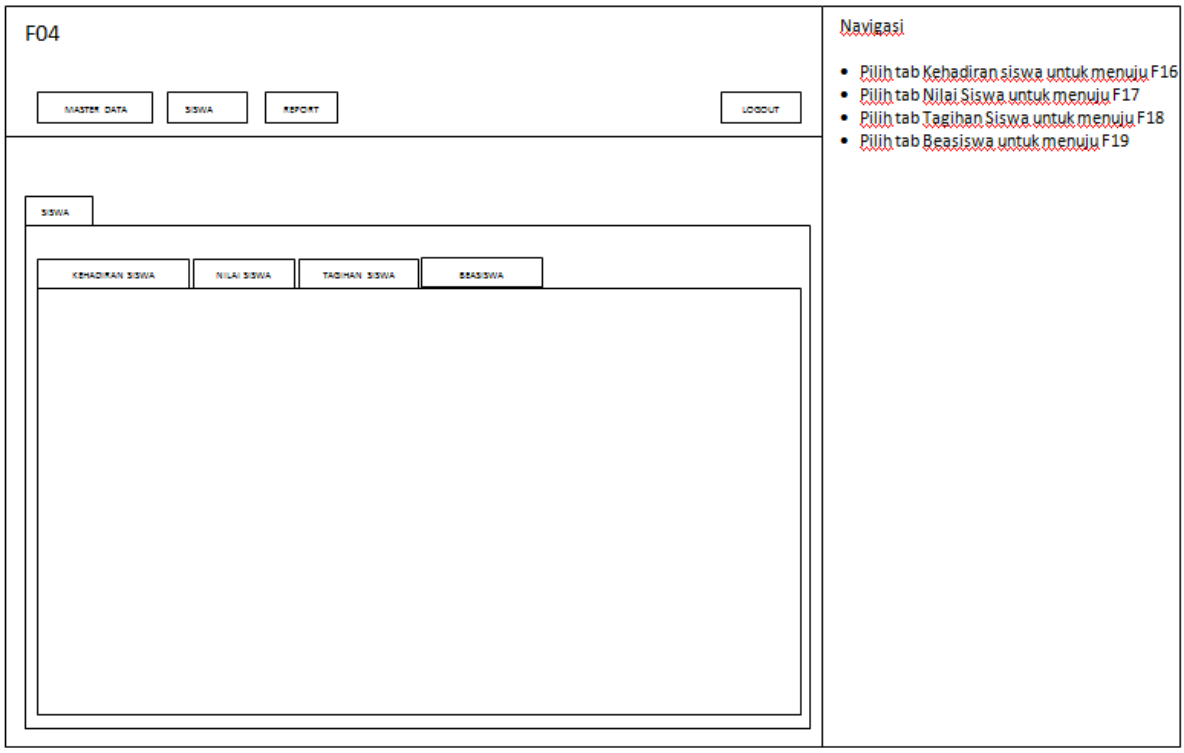


Navigasi

- Isi kode TU, nama pegawai TU, NIP/NUPTK, Jenis kelamin, Golongan, Status, Tugas, No telp/hp
- Jika kode TU belum diisimaka muncul P47
- Jika nama pegawai TU belum diisi maka muncul P48
- Jika NIP belum diisi maka muncul P49
- Jika jenis kelamin belum diisi maka muncul P50
- Jika golongan belum dipilih maka muncul P51
- Jika status belum diisi maka muncul P52
- Jika tugas belum diisi maka muncul P53
- Jika no telp/hpbelum diisi maka muncul P54
- Isi kode user untuk mencari
- Jika ditekan tombol simpan maka muncul P04
- Jika tombol update ditekan maka akan muncul P09
- Jika tombol hapus ditekan maka muncul P10
- Jika ditekan tombol batal, maka akan kembali seperti semula

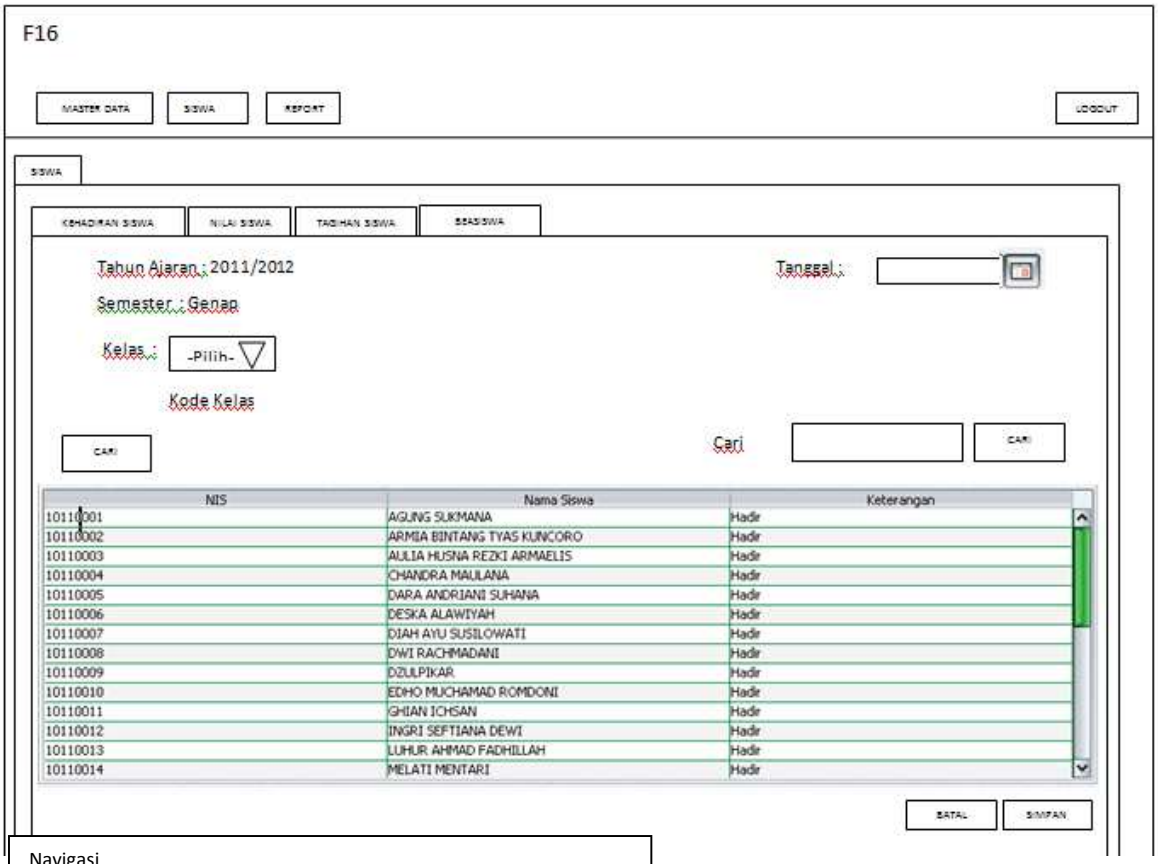
Gambar 3. 55 : Tab TU

12. Menu Siswa



Gambar 3. 56 : Menu Siswa

13. Tab Kehadiran Siswa

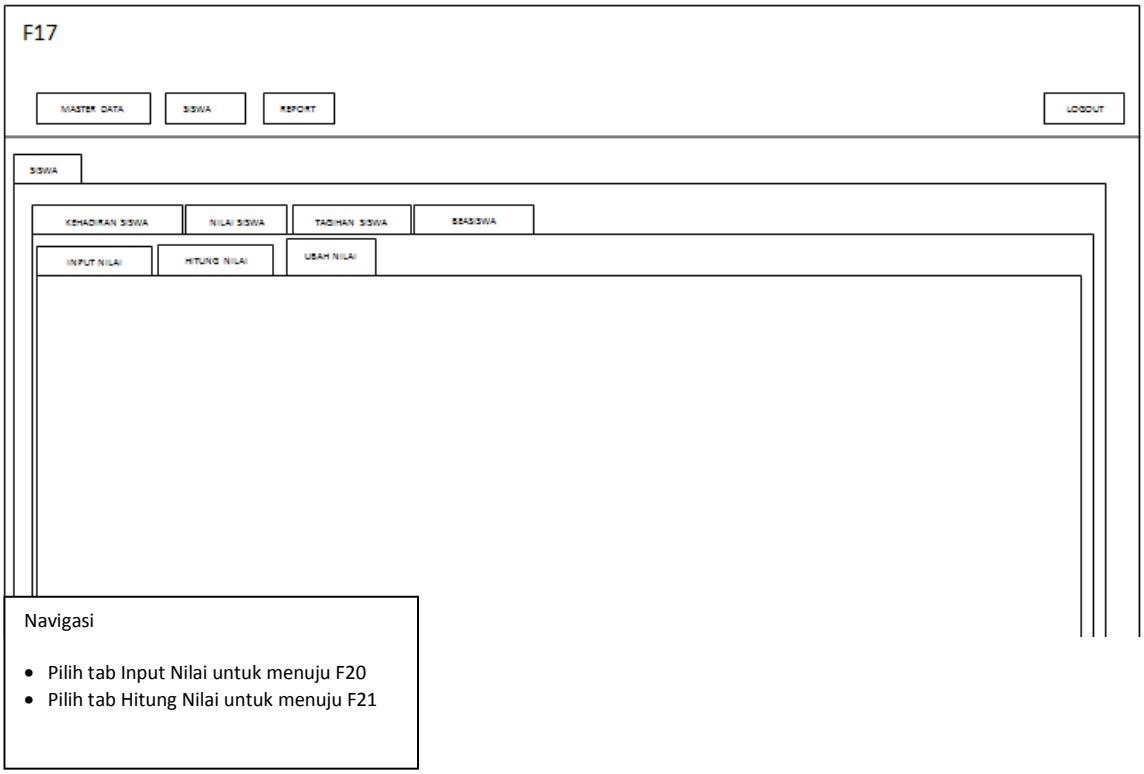


Navigasi

- Pilih Kelas dan mata pelajaran, kemudian isi tanggal absen
- Jika tanggal belum diisi, akan muncul P55
- Jika tekan tombol simpan, akan mucul P04
- Klik tombol batal untuk mengosongkan isi tabel
- Isi kolom cari untuk mencari data yang diinginkan

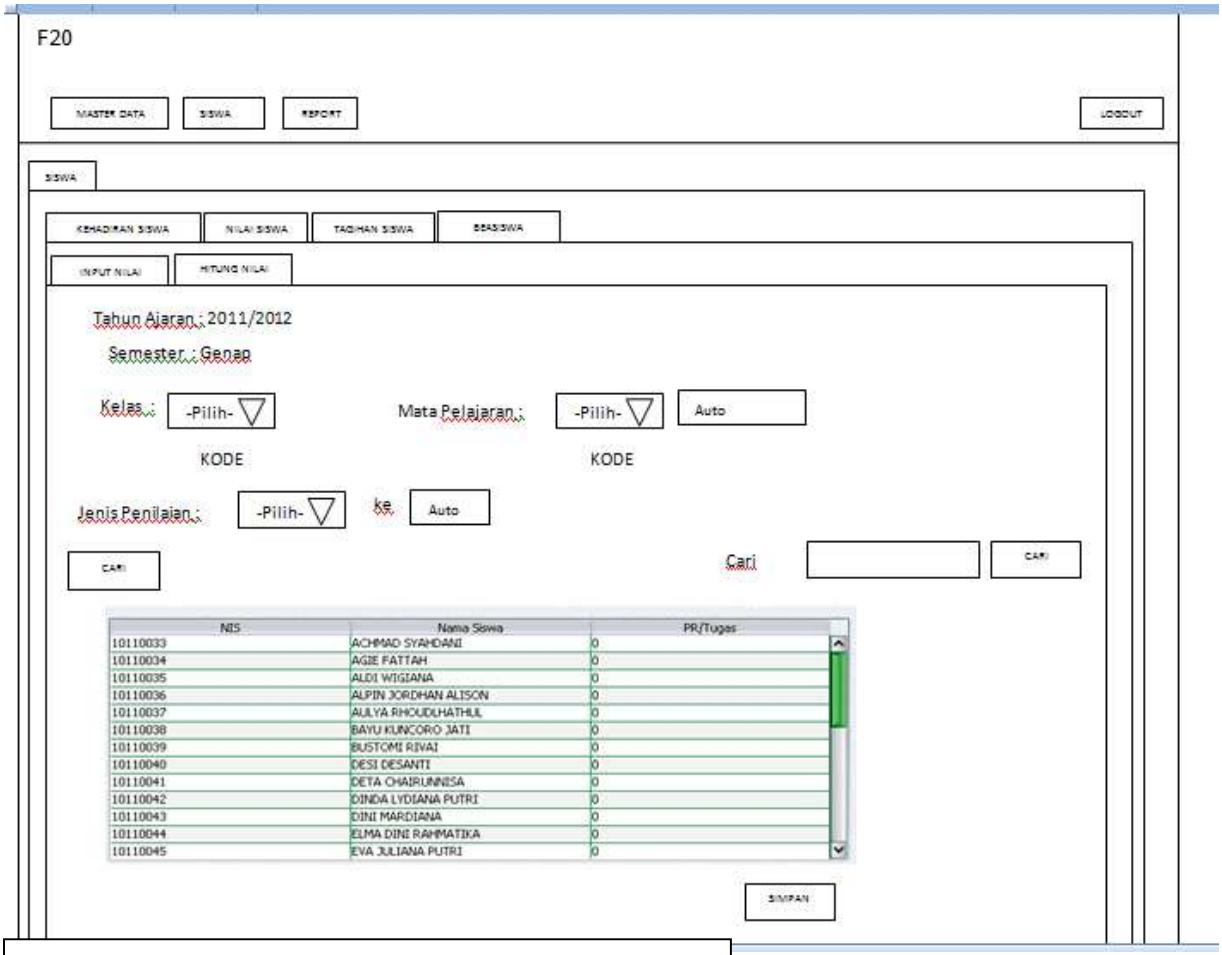
Gambar 3. 57 : Tab kehadiran Siswa

14. Tab Nilai Siswa



Gambar 3. 58 : Gambar Tab Nilai Siswa

15. Tab Inputan Nilai Siswa

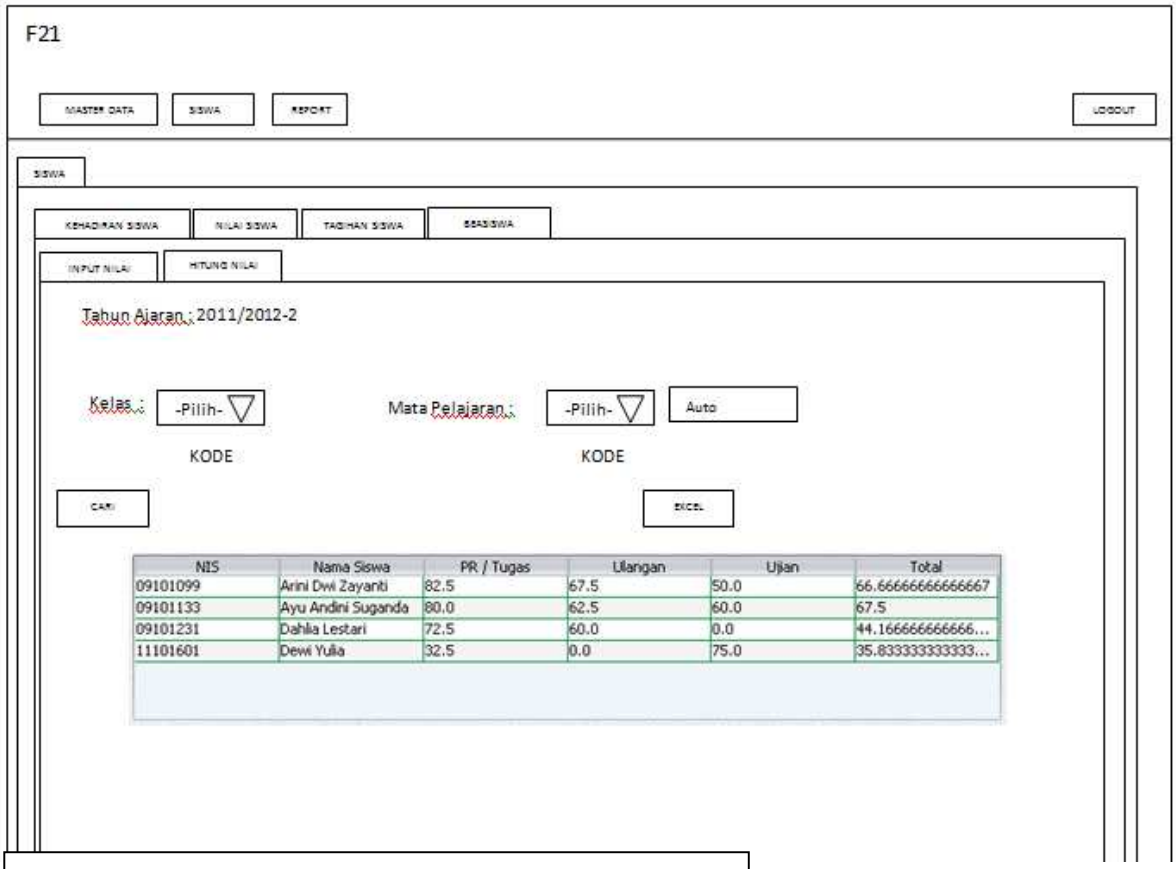


Navigasi

- Pilih Kelas, mata pelajaran, dan jenis penilaian. Jenis penilaian berupa PR / Tugas, Ulangan atau Ujian
- Kemuadian cari data siswa
- Jika tekan tombol simpan, akan mucul P04
- Klik tombol batal untuk mengosongkan isi tabel
- Isi kolom cari untuk mencari data yang diinginkan

Gambar 3. 59 : Tab Inputan Nilai Siswa

16. Tab Hitung Nilai Siswa

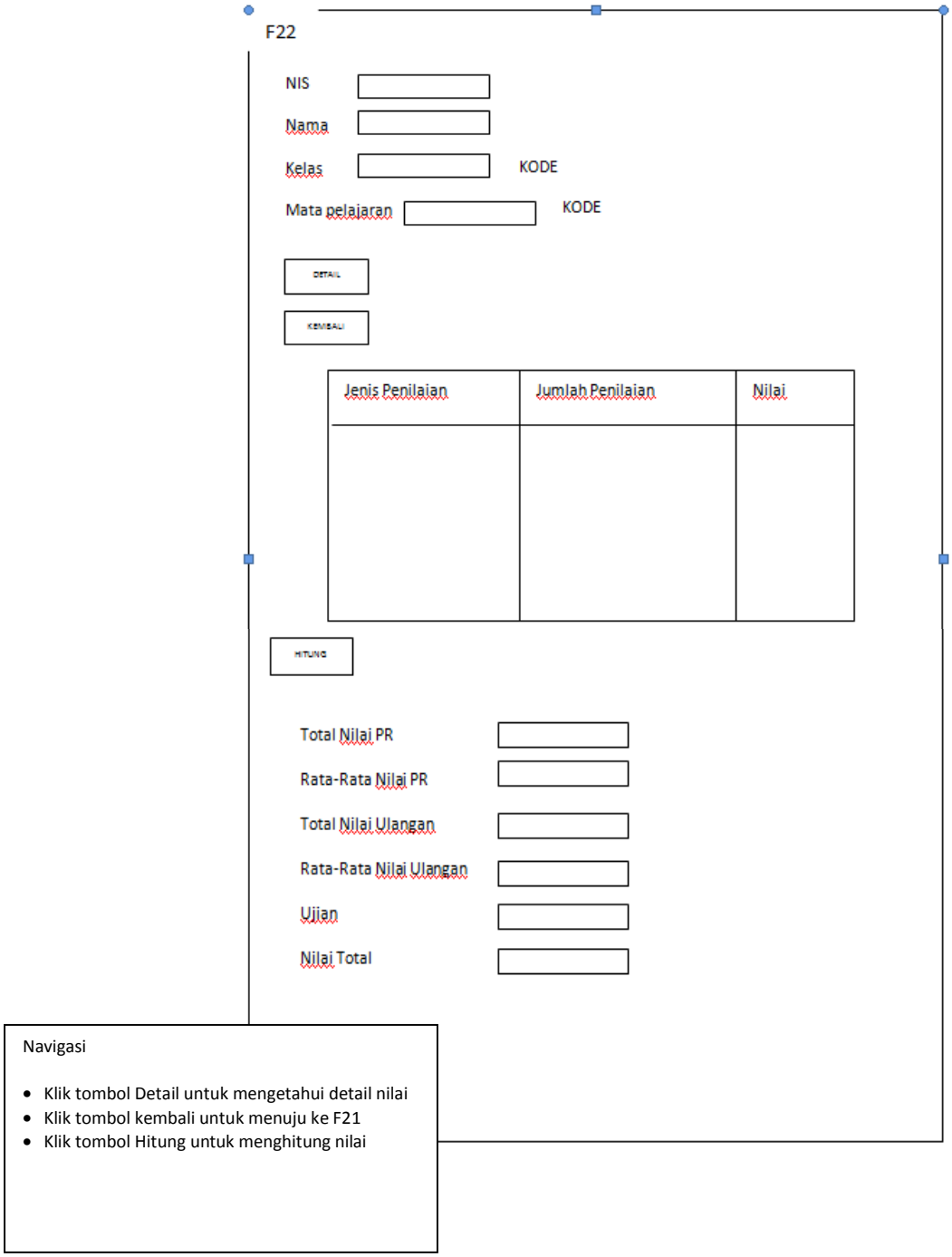


Navigasi

- Pilih Kelas, mata pelajaran
- Kemuadian cari data siswa
- Jika tekan tombol excel,maka file excel akan terbentuk
- Klik salah 1 nama siswa untuk melihat rincian nilai siswa, dan akan menuju ke F22

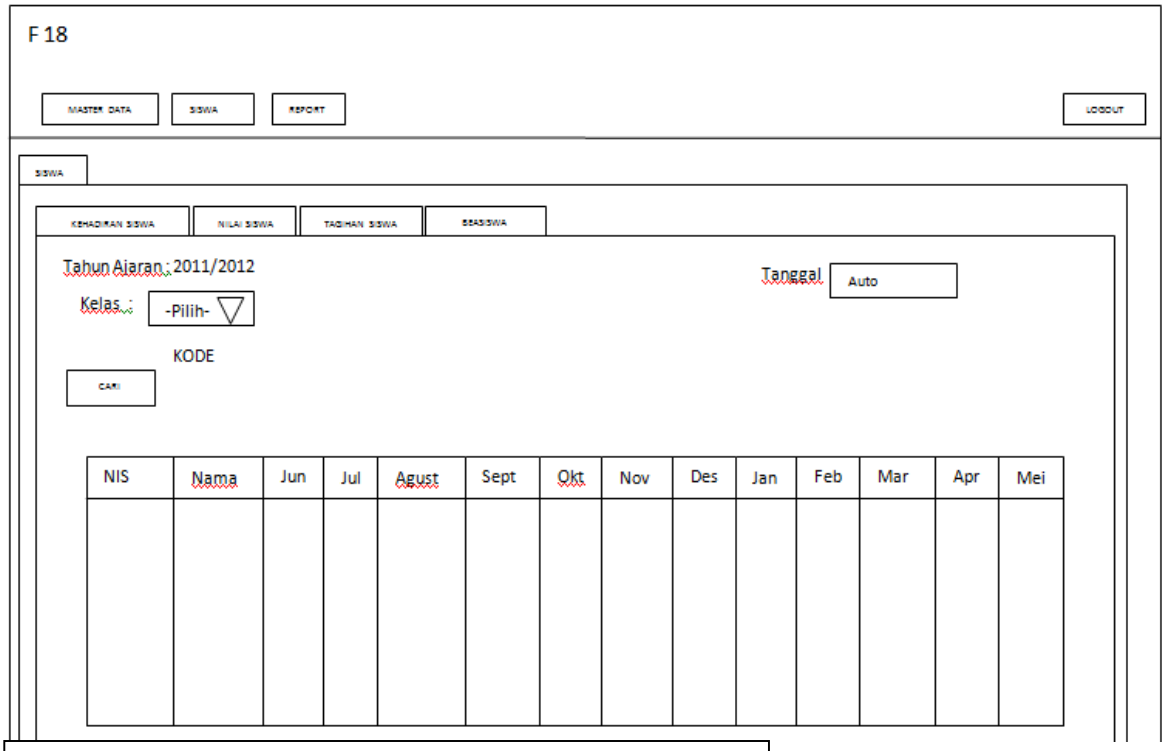
Gambar 3. 60 : Tab Hitung Nilai Siswa

17. Menu Detail Nilai Siswa



Gambar 3. 61 : Menu Detail Nilai Siswa

18. Tab Tagihan

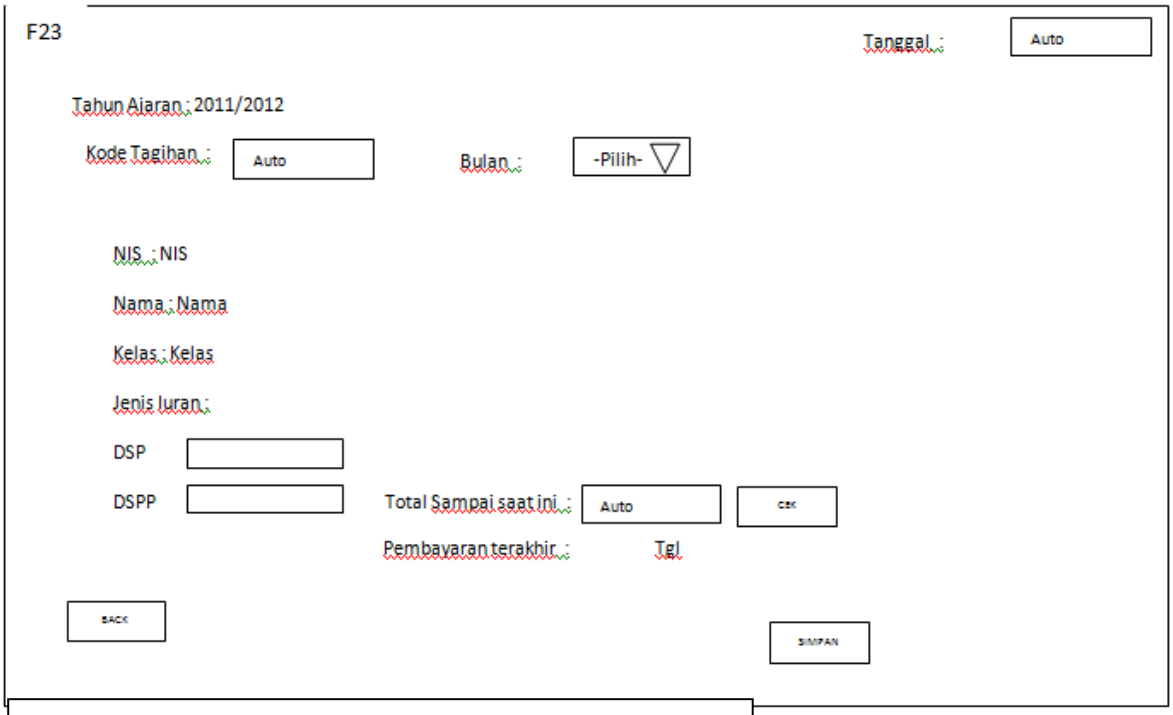


Navigasi

- Pilih Kelas
- Klik tombol cari untuk mencari data siswa
- Klik salah 1 nama siswa untuk menginput data iuran siswa, dan akan menuju ke F23

Gambar 3. 62 : Tab Tagihan

19. Form Input Tagihan

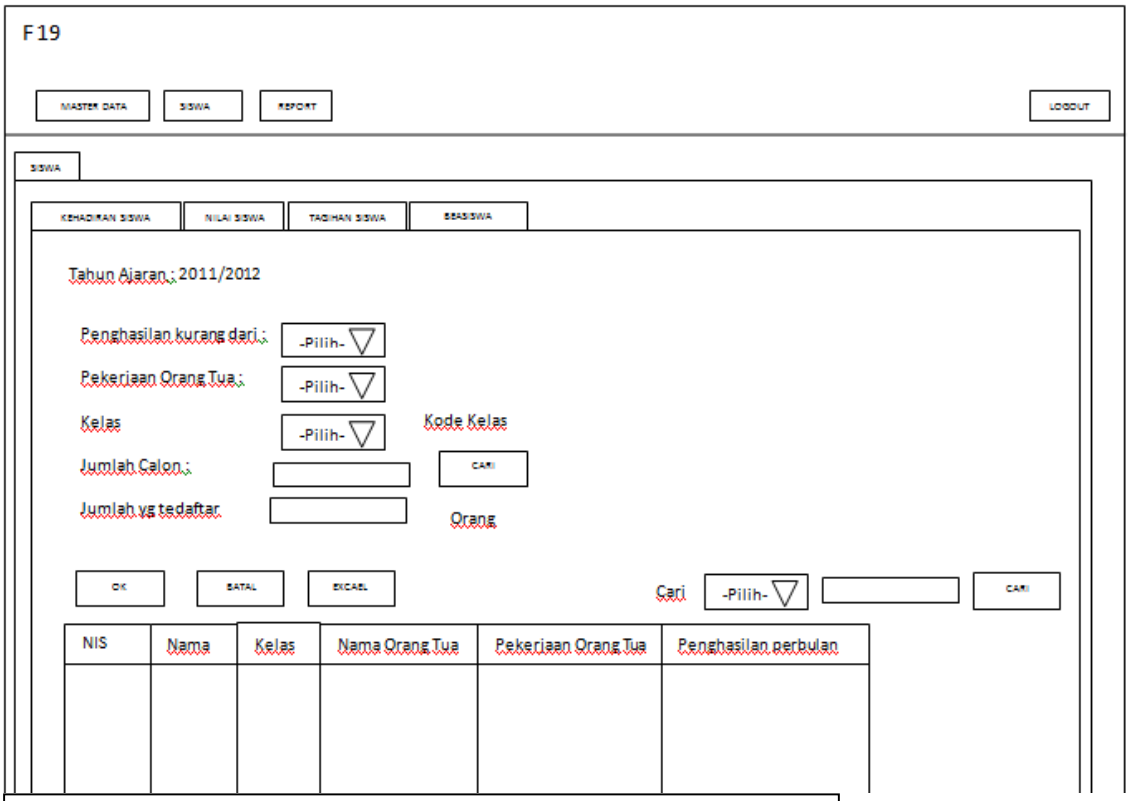


Navigasi

- Pilih bulan pembayaran
- Jika bulan belum dipilih akan muncul P56
- Isi kolom DSP, dan DSPP.
- Jika inputan DSP bernilai minus, akan muncul P57
- Jika inputanDSPP bernilai minus, akan muncul P58
- Klik cek untuk mengetahui total pembayaran iuran DSPP sampai saat ini
- Jika data tidak ada, maka akan muncul P59
- Jika menekan tombol Back akan menuju F18
- Jika menekan tombol simpan, akan muncul P04

Gambar 3. 63 : Form Input Tagihan

20. Tab Beasiswa



Navigasi

- Pilih Penghasilan orang tua dan atau pekerjaan orang tua, dan jumlah calon yang ingin cari
- Kemudian klik tombol cari untuk mencari data calon penerima beasiswa
- Jika jumlah calon belum diisi, muncul P60
- Jika data tidak ditemukan, akan muncul P59
- Jika menekan tombol OK, akan muncul P04

Gambar 3. 64 : Tab Beasiswa

21. Menu Report



Navigasi

- Pilih tab Master Data untuk menuju F24
- Pilih tab Siswa untuk menuju F25

Gambar 3. 65 : Menu Report

22. Tab Master Data



Navigasi

- Pilih tab Siswa untuk menuju F26
- Pilih tab Guru untuk menuju F27
- Pilih tab Orang Tua untuk menuju F28
- Pilih tab Mengajar untuk menuju F29
- Pilih tab Wali Kelas untuk menuju F
-

Gambar 3. 66 : Tab Master Data

23. Tab Siswa

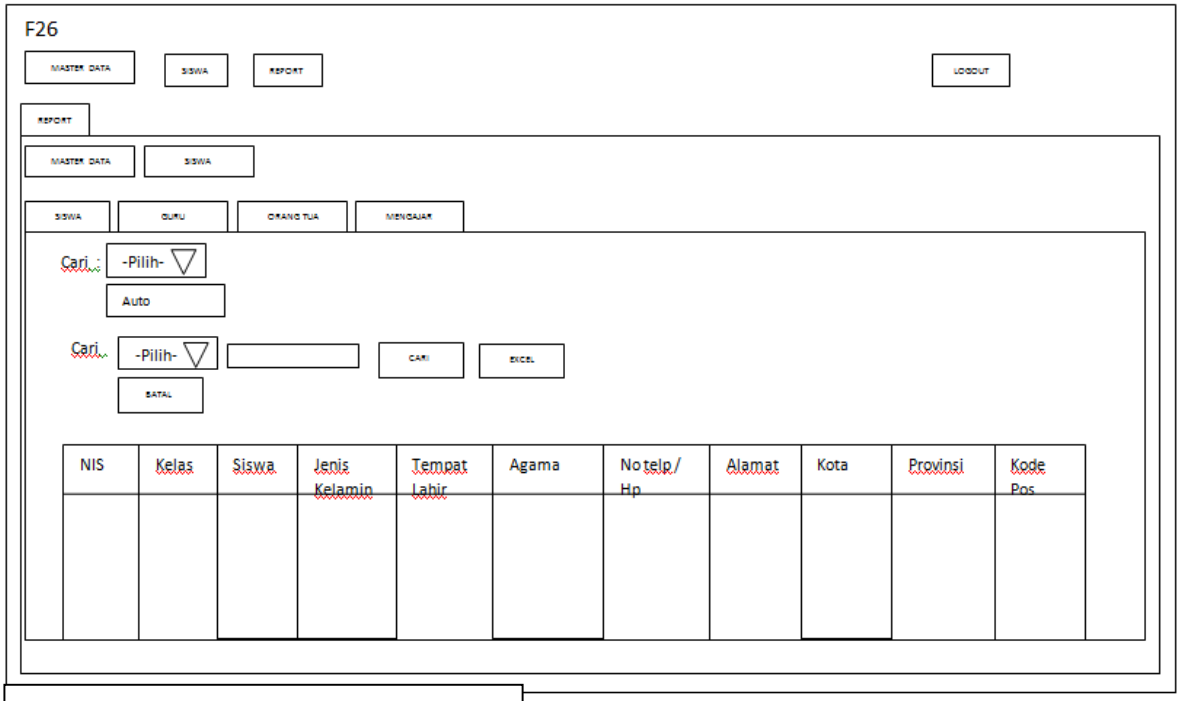


Navigasi

- Pilih tab Kehadiran Siswa untuk menuju F30
- Pilih tab Nilai Siswa untuk menuju F31
- Pilih tab Tagihan Siswa untuk menuju F32
- Pilih tab Beasiswa untuk menuju F33

Gambar 3. 67 : Tab Siswa

24. Tab Siswa

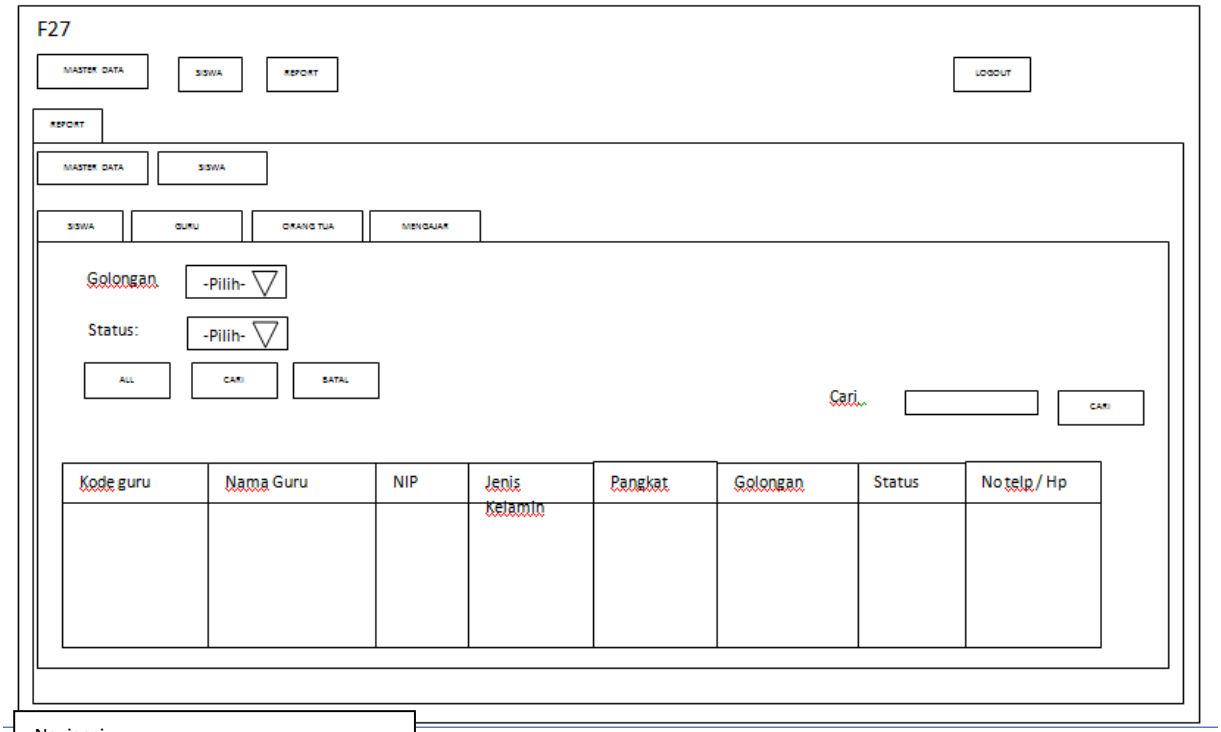


Navigasi

- Pilih cari
- Jika data tidak ada maka akan keluar P59

Gambar 3. 68 : Tab Siswa

25. Tab Guru

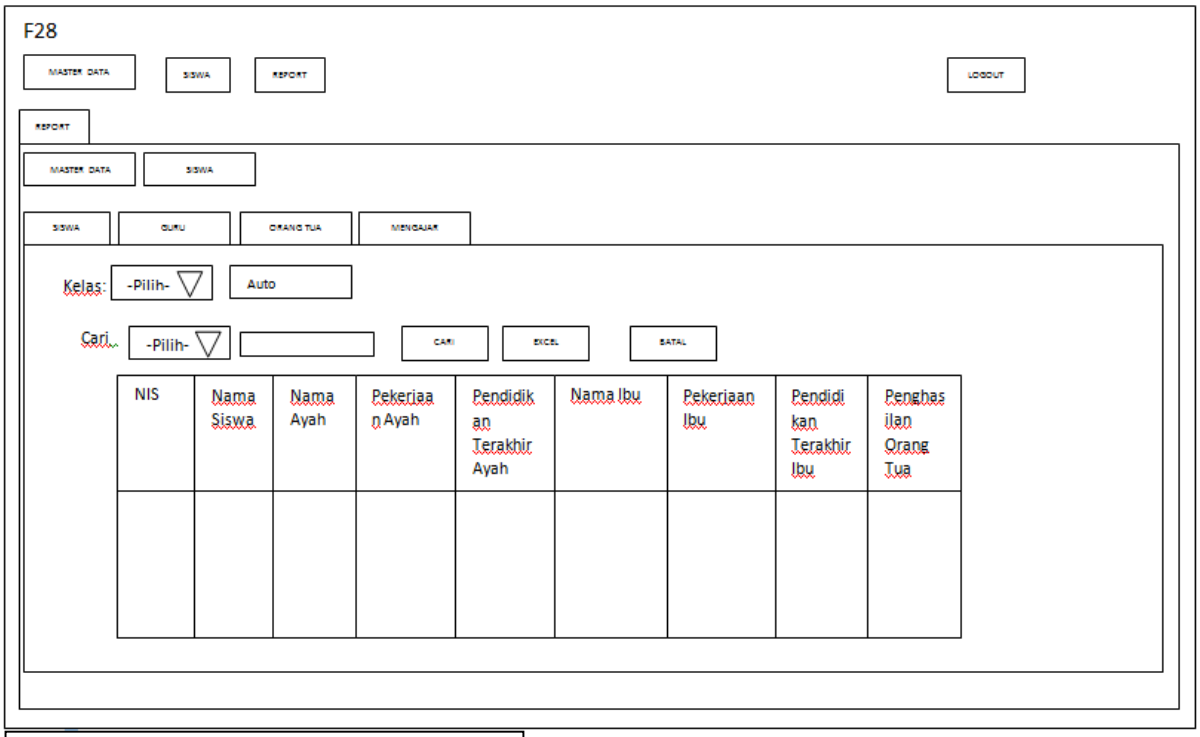


Navigasi

- Pilih golongan dan status
- Jika data tidak ada maka akan keluar P59

Gambar 3. 69 : Tab Guru

26. Tab Orang Tua

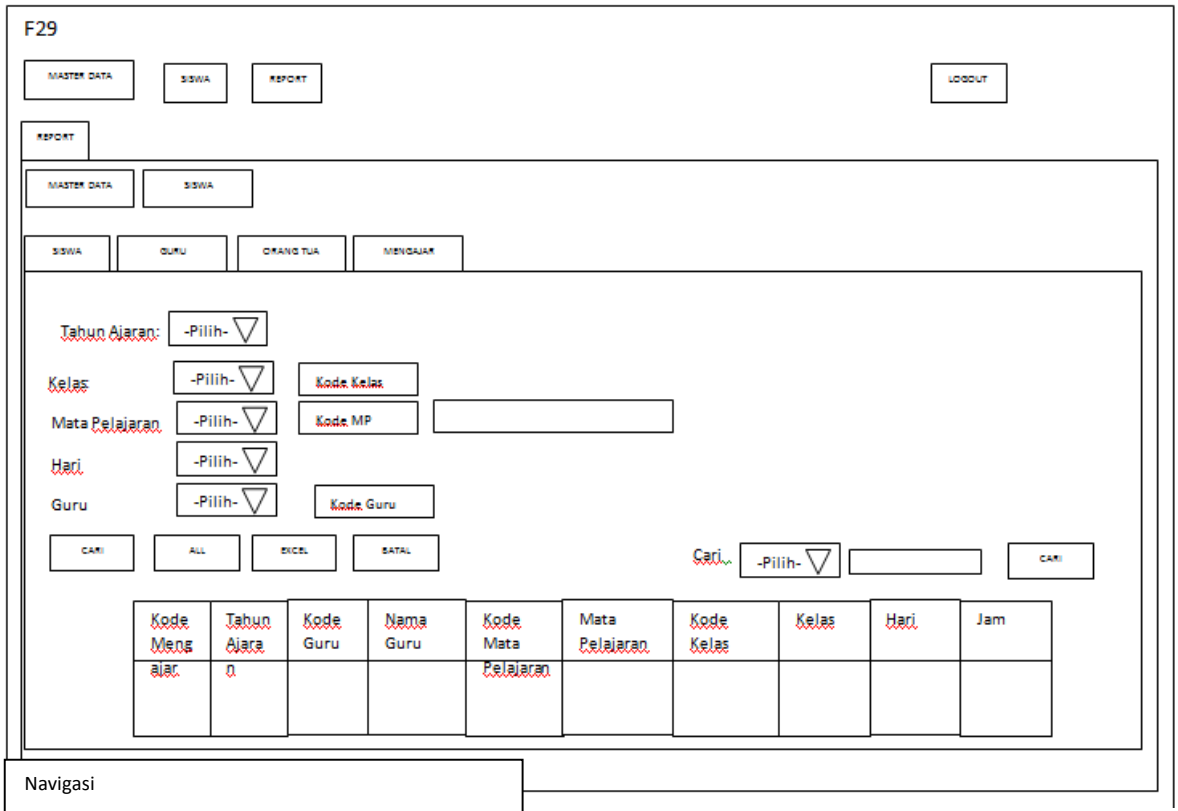


Navigasi

- Pilih kelas
- Jika data tidak ada maka akan keluar P59

Gambar 3. 70 : Tab Orang Tua

27. Tab Mengajar

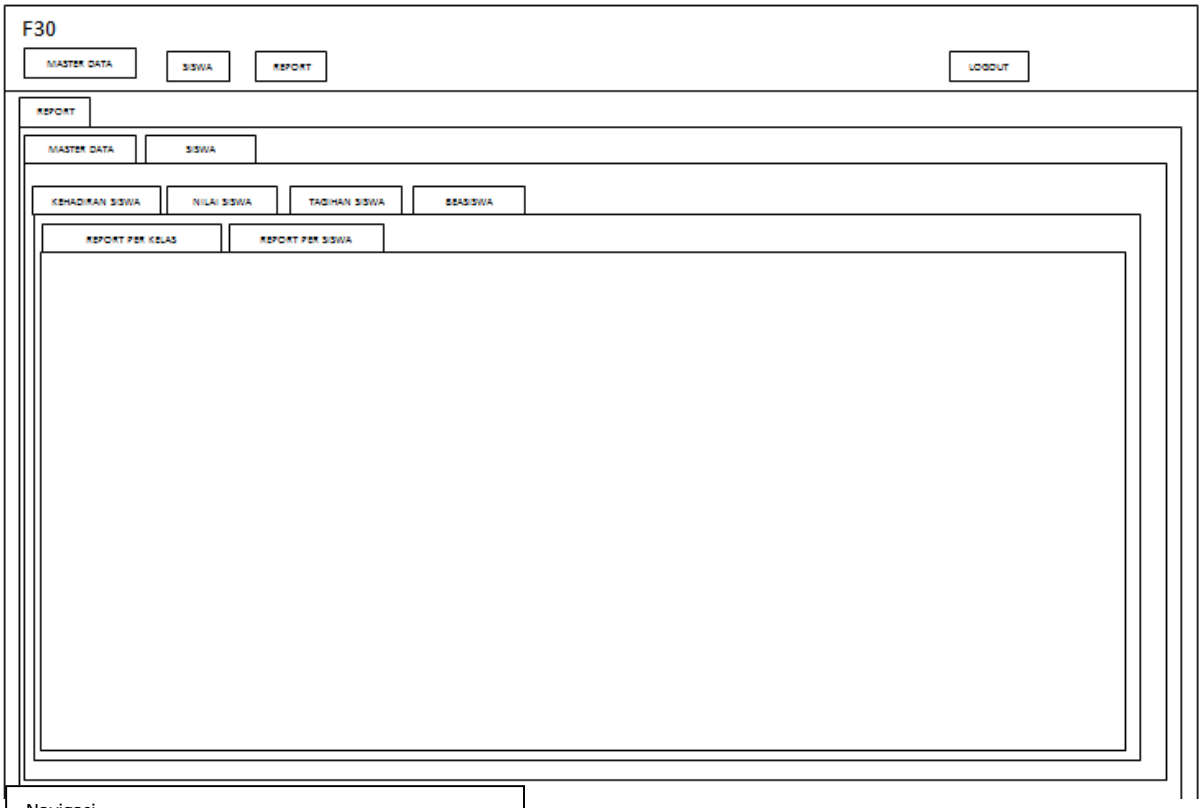


Navigasi

- Pilih kelas atau mata pelajaran, atau hari, atau guru
- Jika data tidak ada maka akan keluar P59

Gambar 3. 71 : Tab Mengajar

28. Tab Kehadiran Siswa

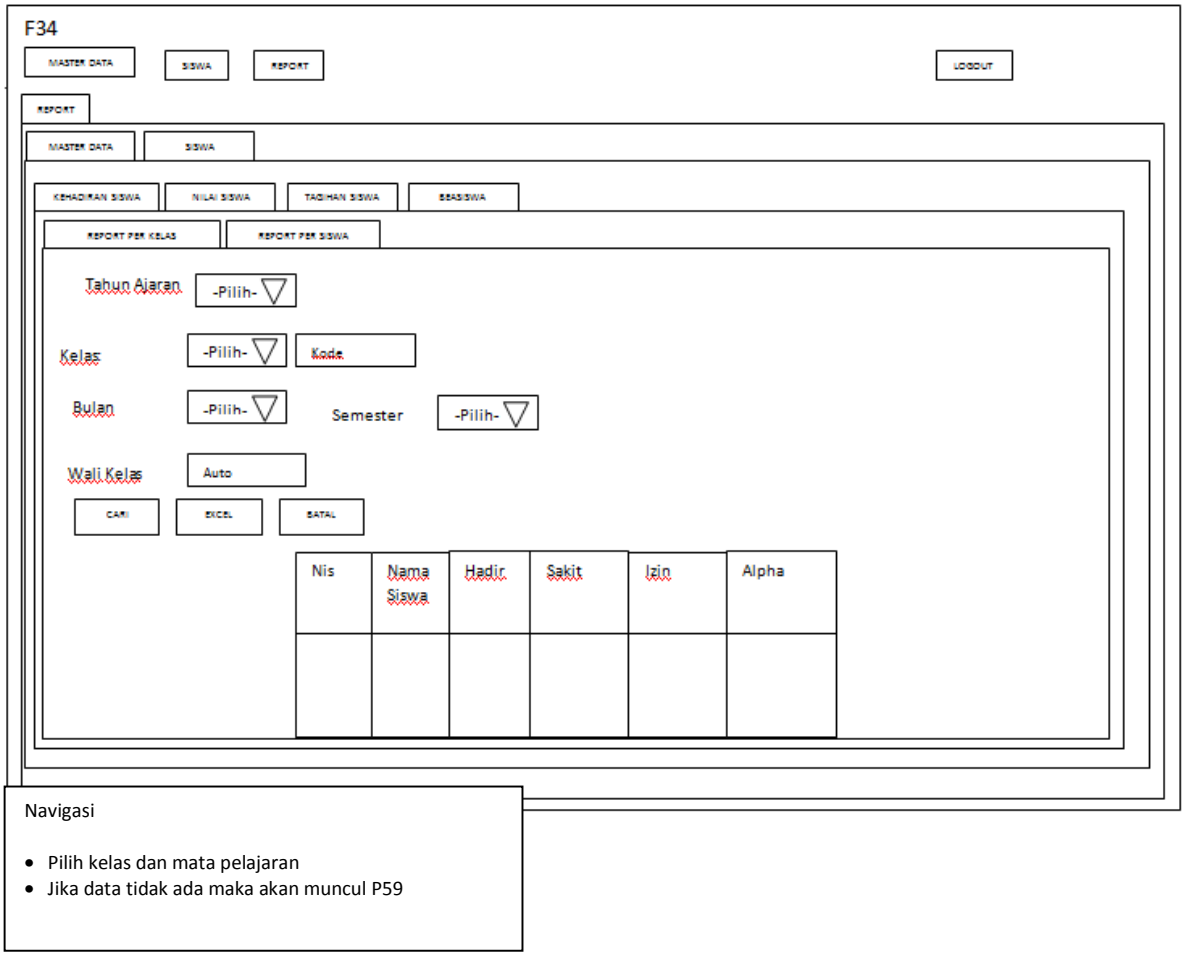


Navigasi

- Pilih tab Report Per Kelas untuk menuju F34
- Pilih tab Report Per Siswa untuk menuju F35

Gambar 3. 72 : Tab Kehadiran Siswa

29. Tab Report Per Kelas

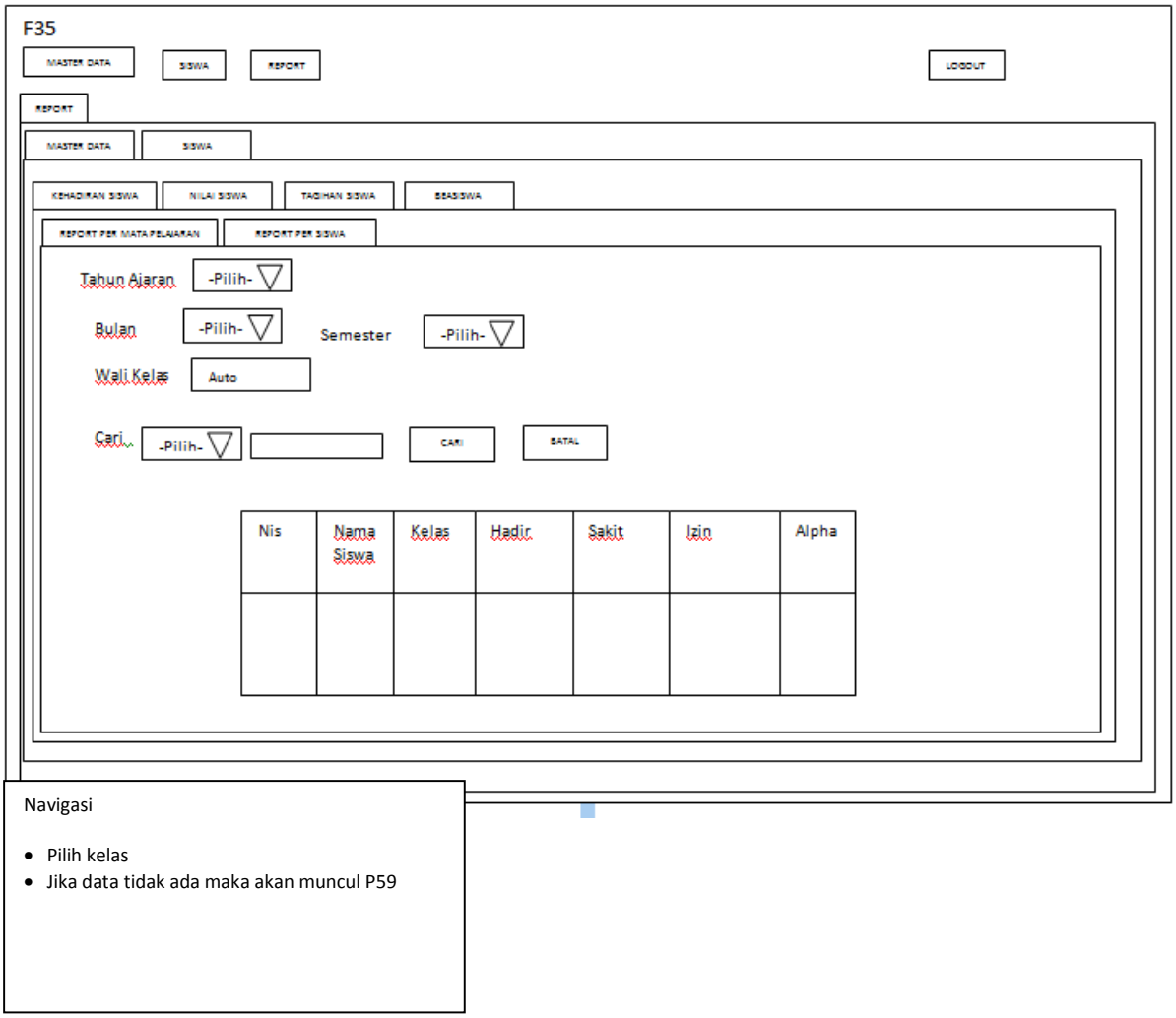


Navigasi

- Pilih kelas dan mata pelajaran
- Jika data tidak ada maka akan muncul P59

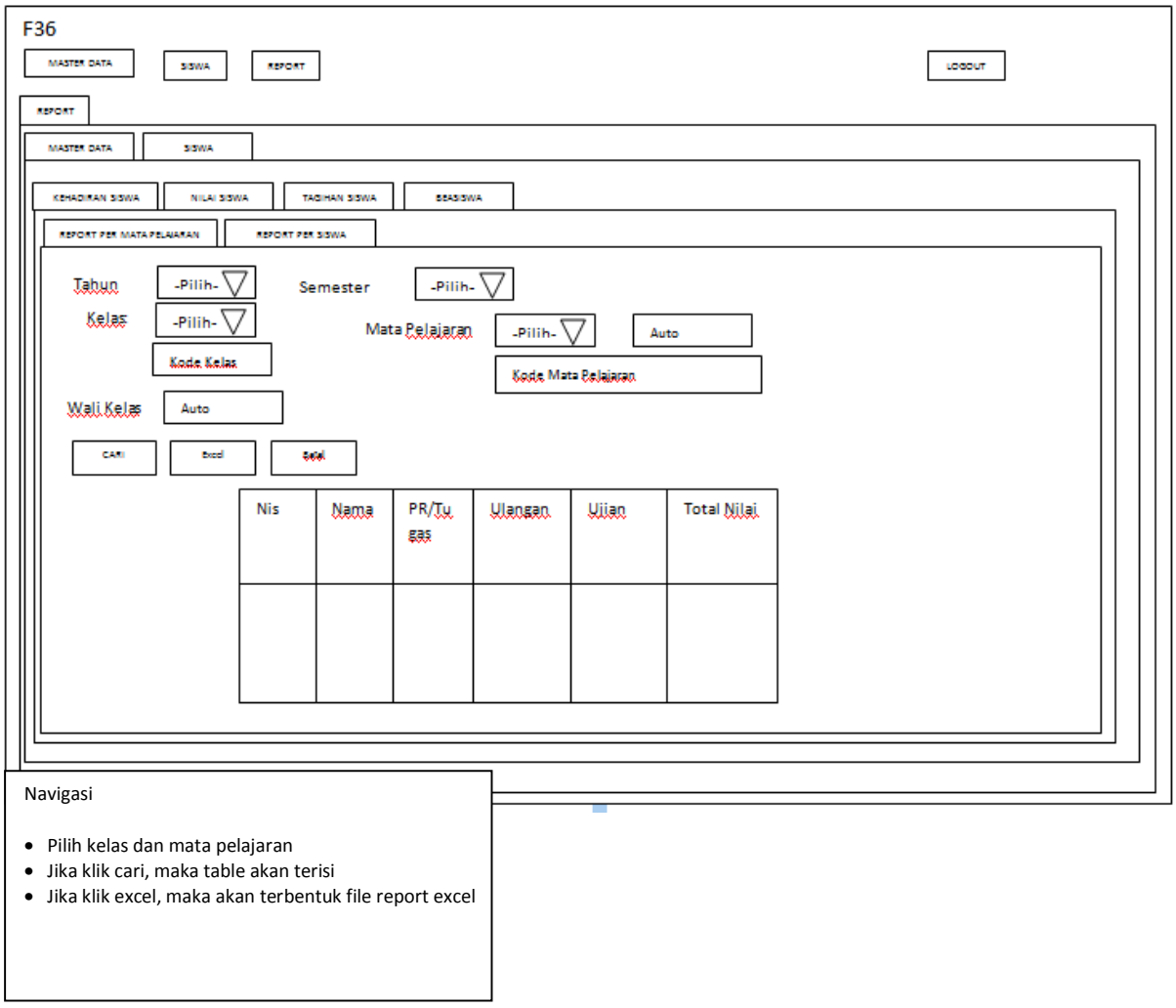
Gambar 3. 73 : Tab Report Per Kelas

30. Tab report persiswa



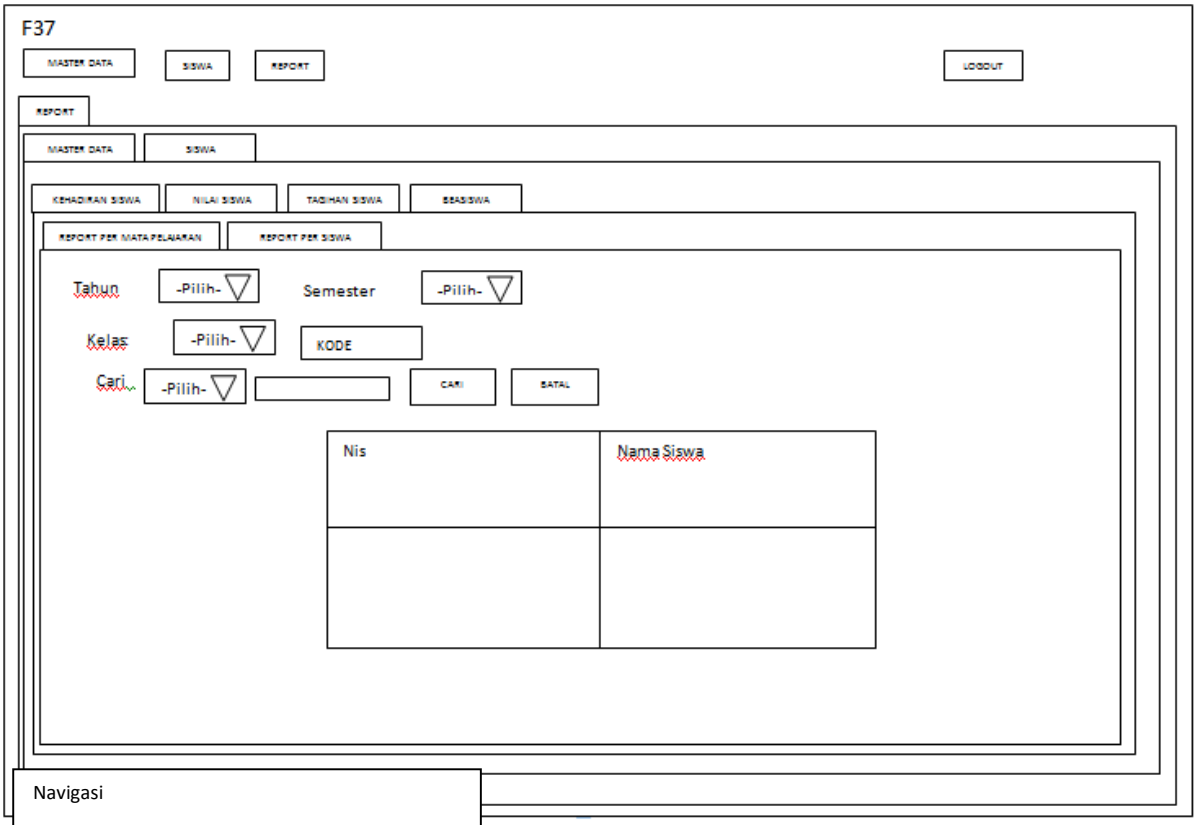
Gambar 3. 74 : Tab Report Persiswa

31. Tab Report Nilai Per mata Pelajaran



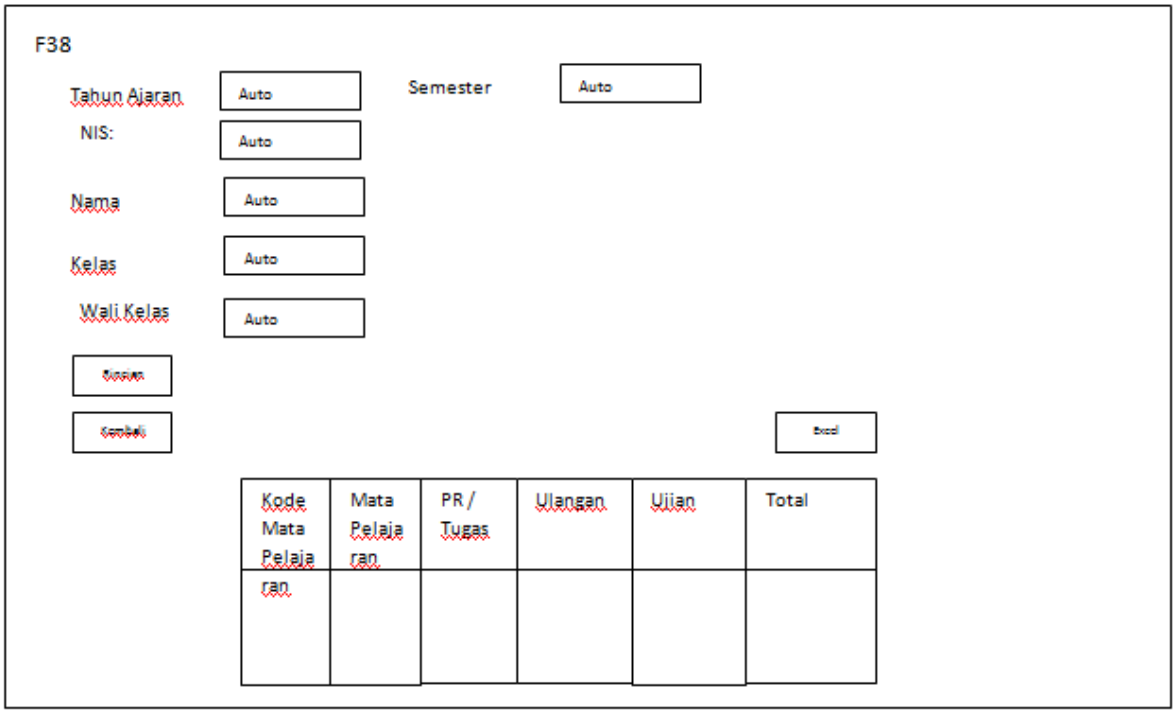
Gambar 3. 75 : Tab Report Per Mata Pelajaran

32. Tab report nilai per siswa



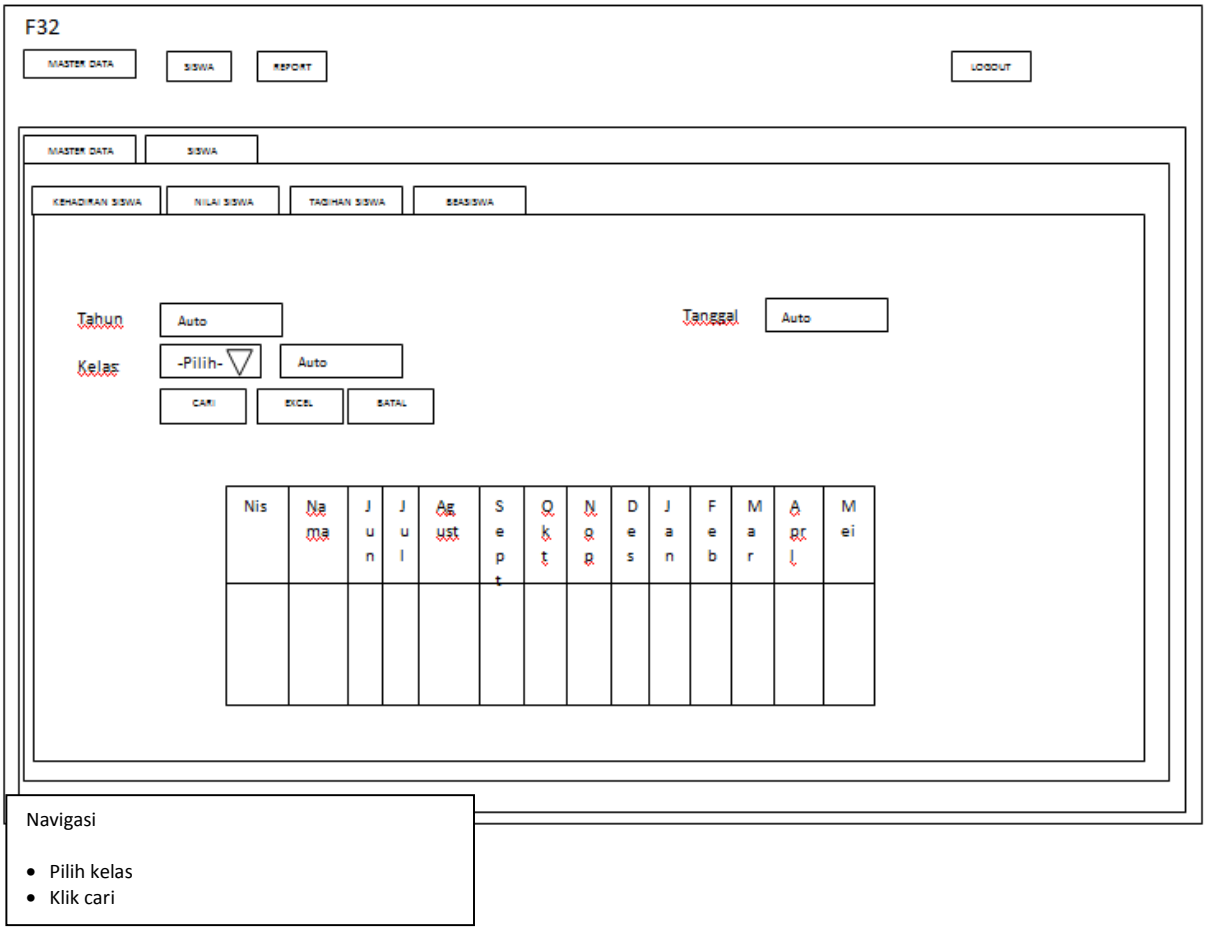
Gambar 3. 76 : Tab Report Persiswa

33. Form Detail Nilai Siswa



Gambar 3. 77 : Detail Nilai Siswa

34. Tab tagihan



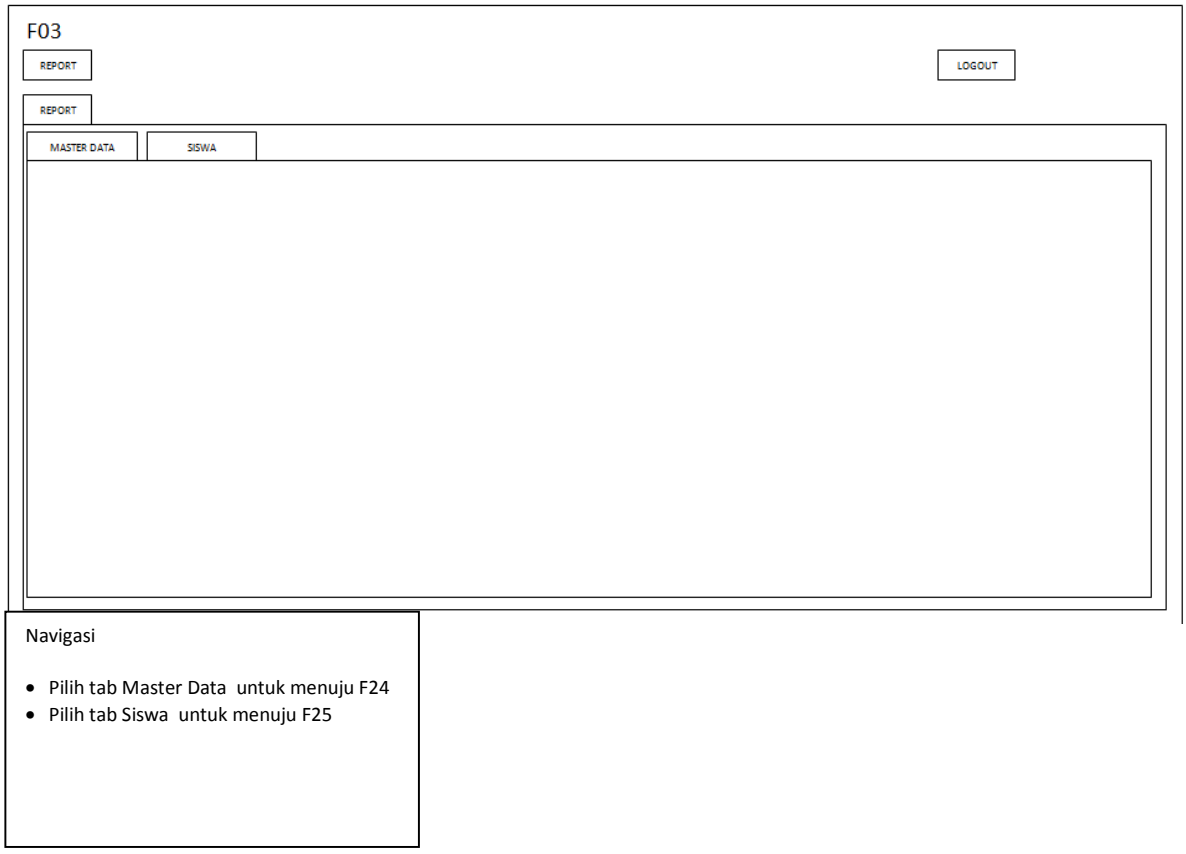
Gambar 3. 78 : Tab Tagihan

35. Tab Beasiswa



Gambar 3. 79 : Tab Beasiswa

36. Menu Guru



Gambar 3. 80 : Menu Guru

37. Tab Master Data

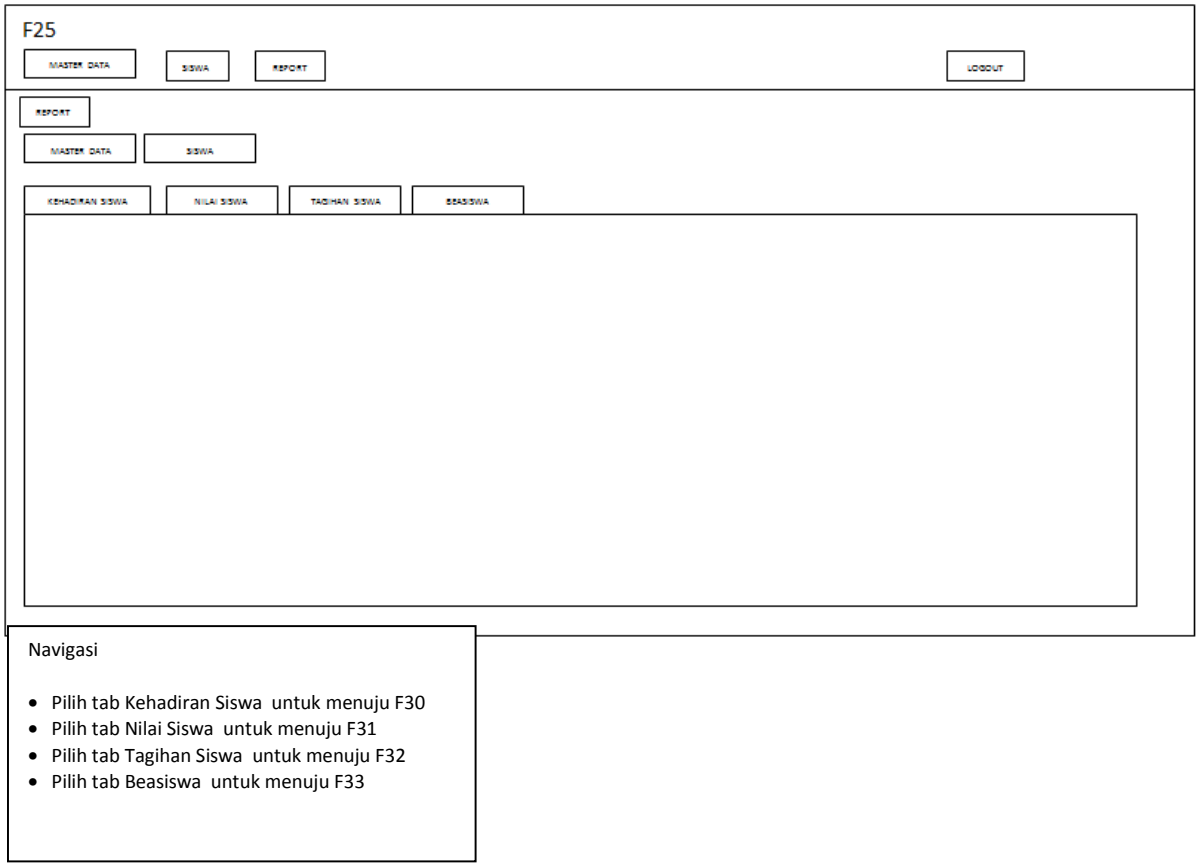


Navigasi

- Pilih tab Siswa untuk menuju F26
- Pilih tab Guru untuk menuju F27
- Pilih tab Orang Tua untuk menuju F28
- Pilih tab Mengajar untuk menuju F29
- Pilih tab Wali Kelas untuk menuju F
-

Gambar 3. 81 : Tab Master Data

38. Tab Siswa

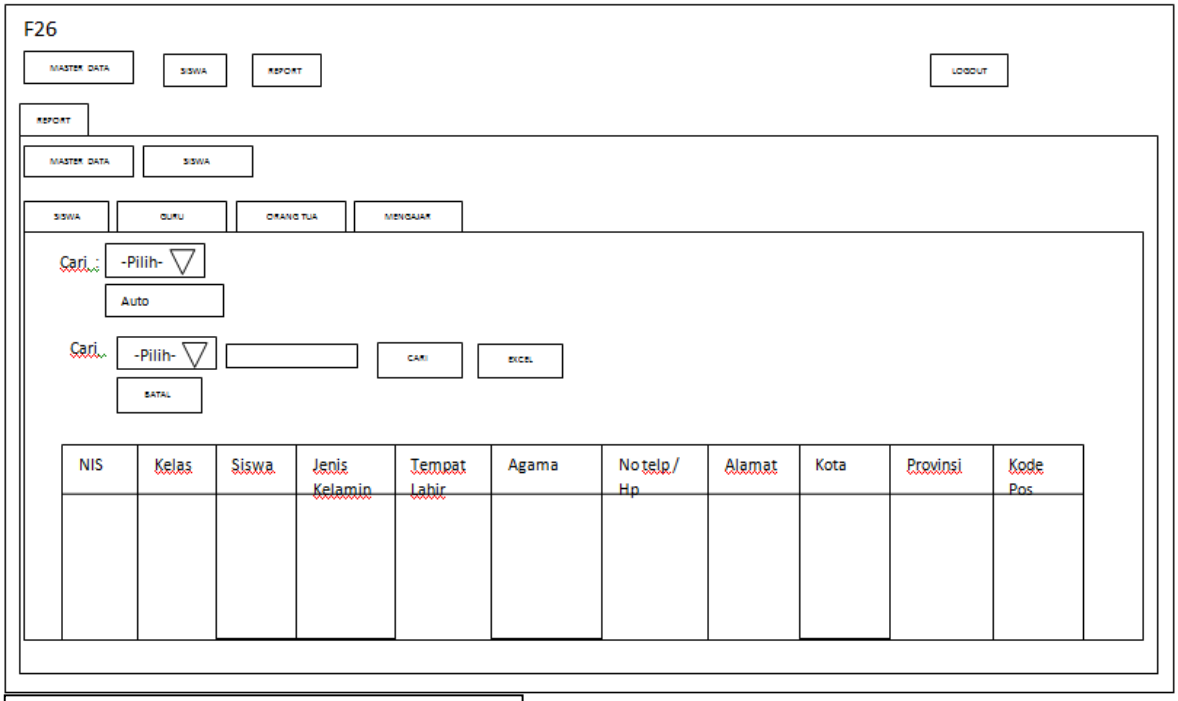


Navigasi

- Pilih tab Kehadiran Siswa untuk menuju F30
- Pilih tab Nilai Siswa untuk menuju F31
- Pilih tab Tagihan Siswa untuk menuju F32
- Pilih tab Beasiswa untuk menuju F33

Gambar 3. 82 : Tab Siswa

39. Tab Siswa

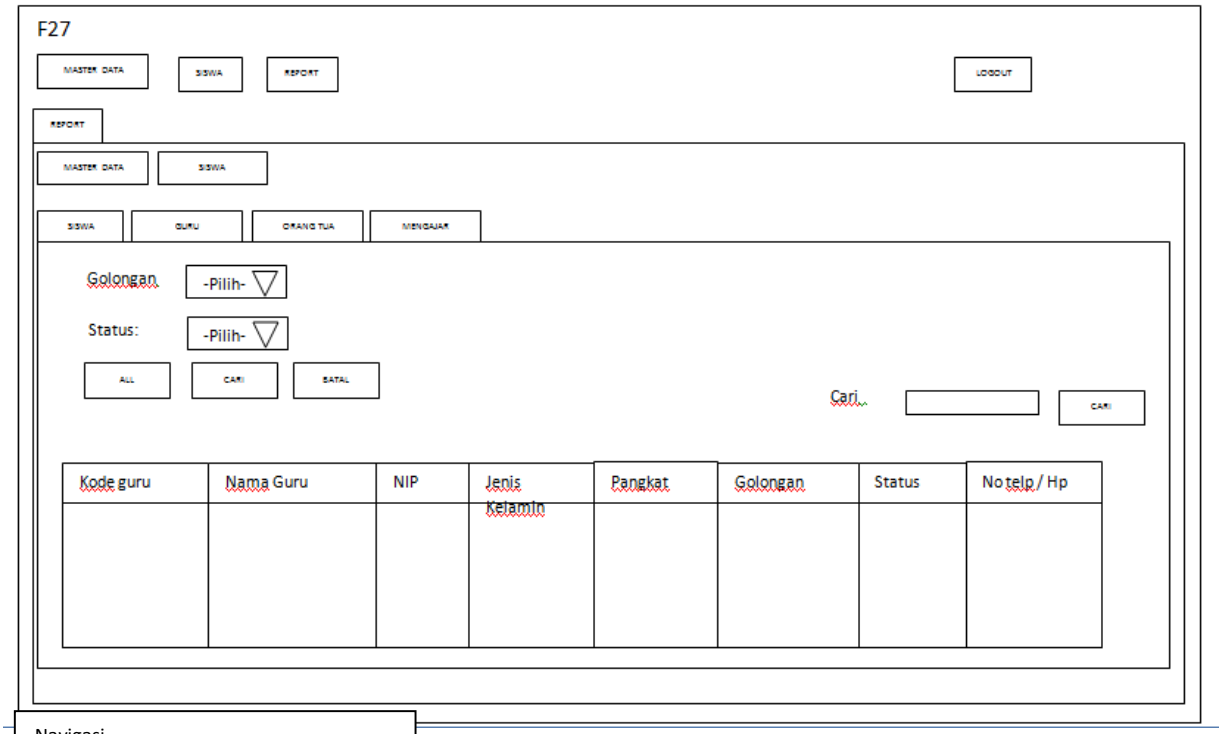


Navigasi

- Pilih cari
- Jika data tidak ada maka akan keluar P59

Gambar 3. 83 : Tab Siswa

40. Tab Guru

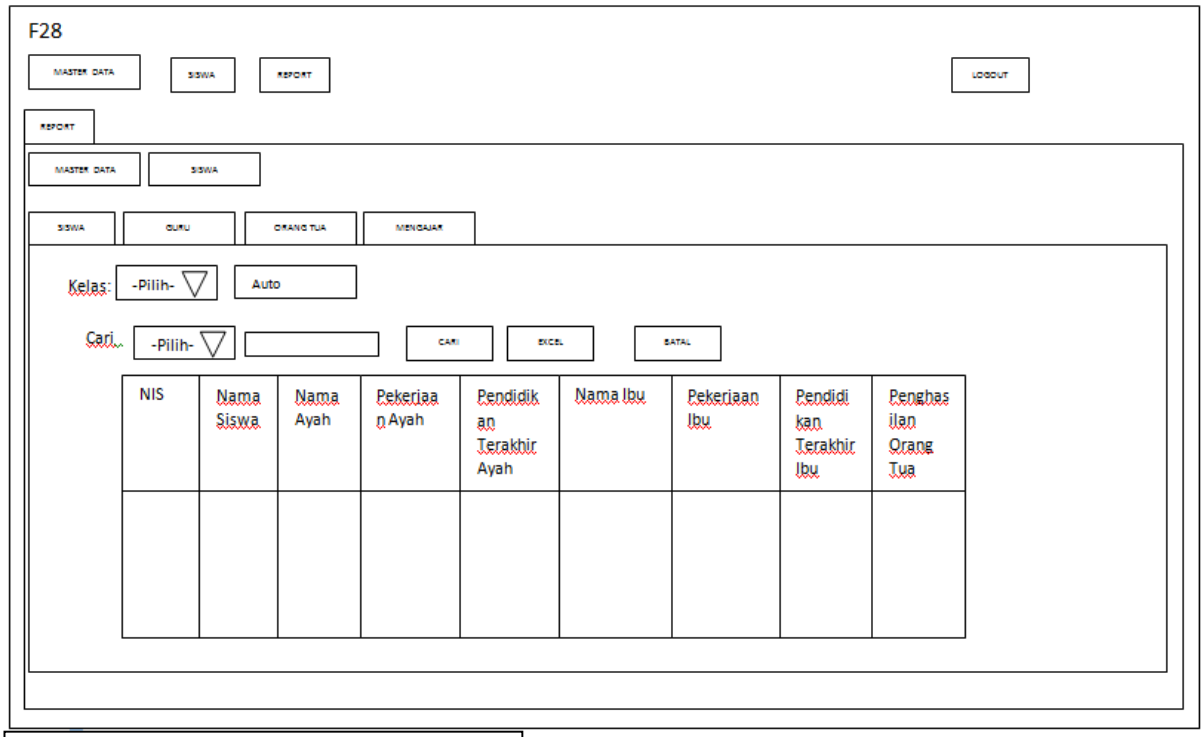


Navigasi

- Pilih golongan dan status
- Jika data tidak ada maka akan keluar P59

Gambar 3. 84 : Tab Guru

41. Tab Orang Tua

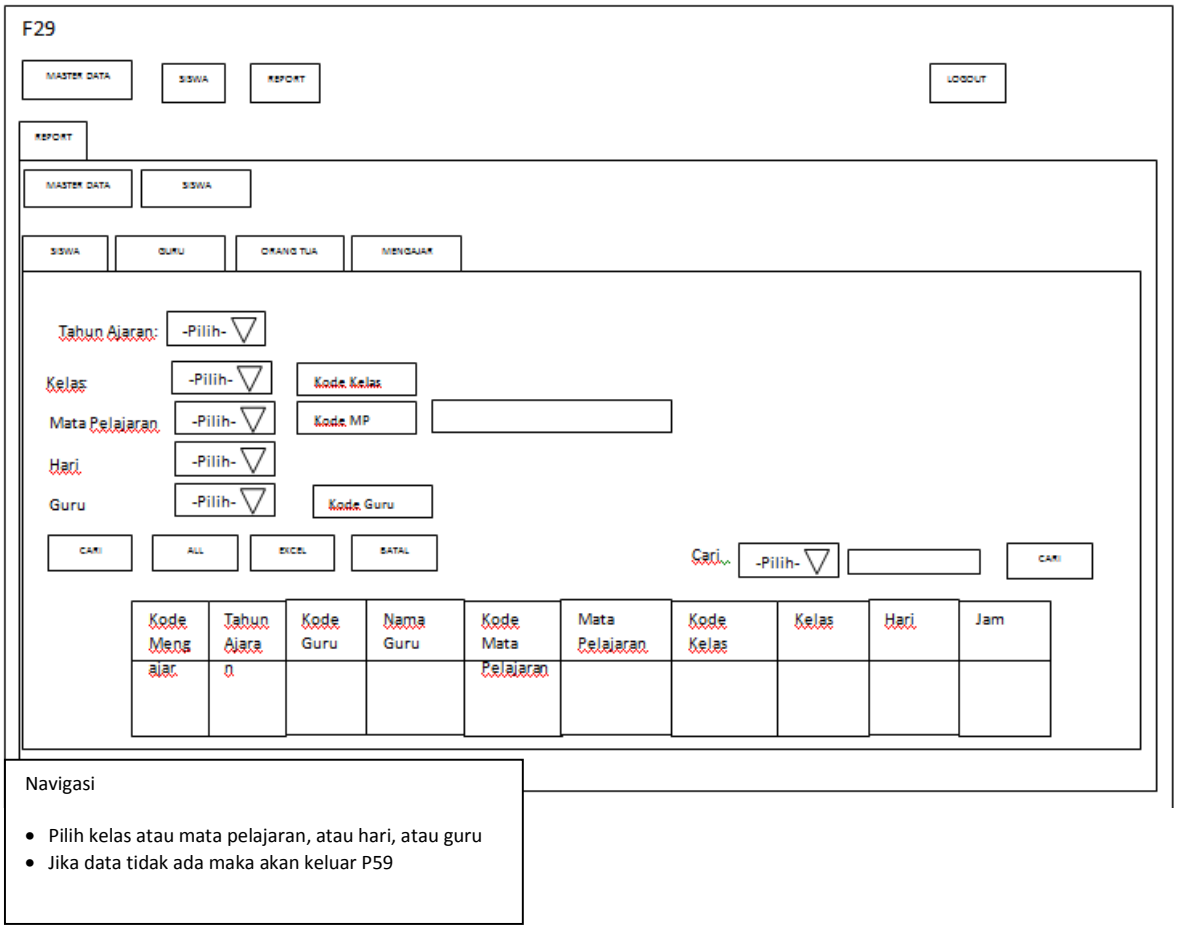


Navigasi

- Pilih kelas
- Jika data tidak ada maka akan keluar P59

Gambar 3. 85 : Tab Orang Tua

42. Tab Mengajar

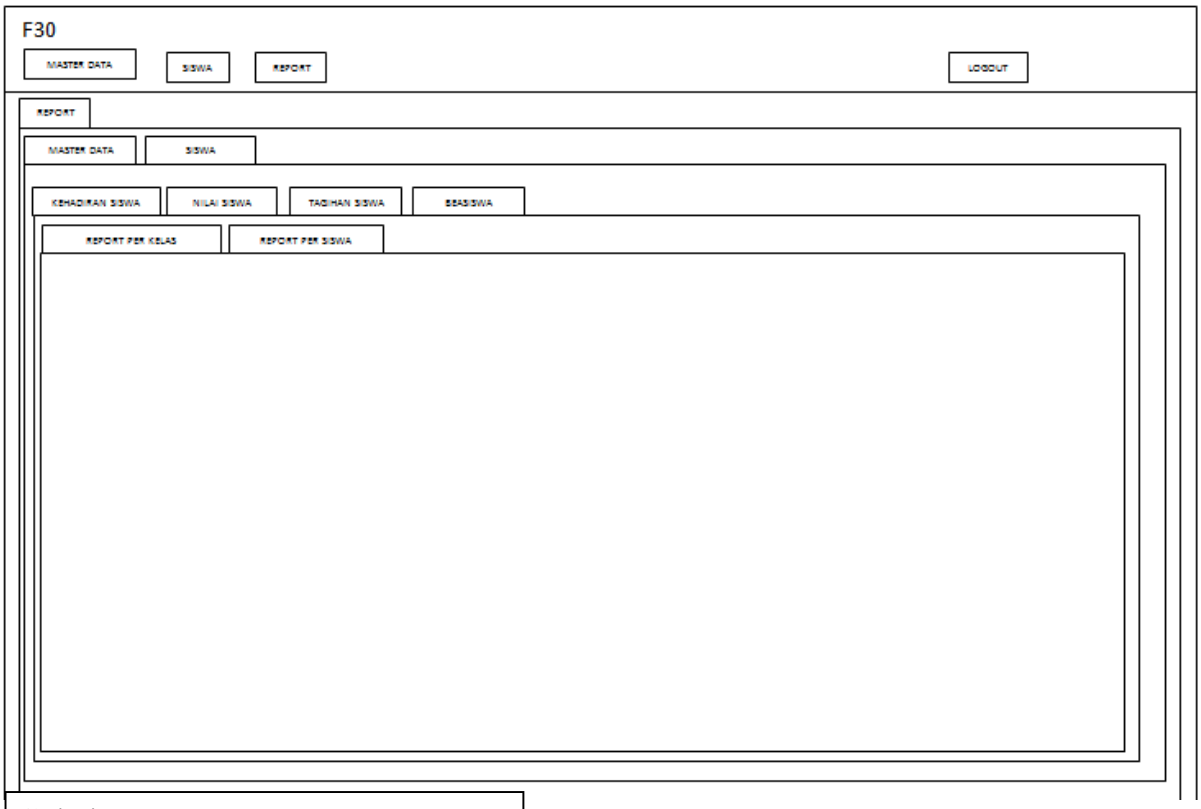


Navigasi

- Pilih kelas atau mata pelajaran, atau hari, atau guru
- Jika data tidak ada maka akan keluar P59

Gambar 3. 86 : Tab Mengajar

43. Tab Kehadiran Siswa

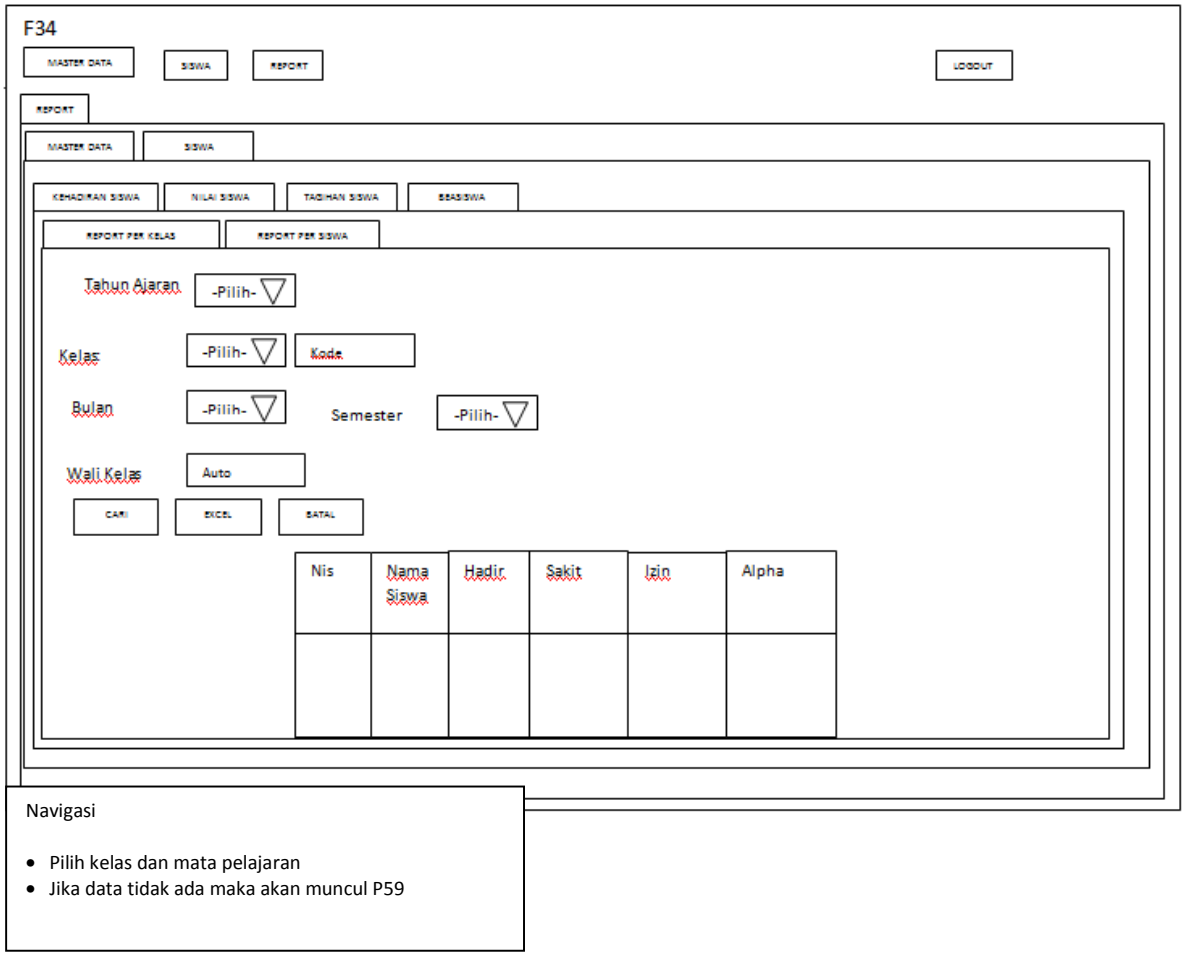


Navigasi

- Pilih tab Report Per Kelas untuk menuju F34
- Pilih tab Report Per Siswa untuk menuju F35

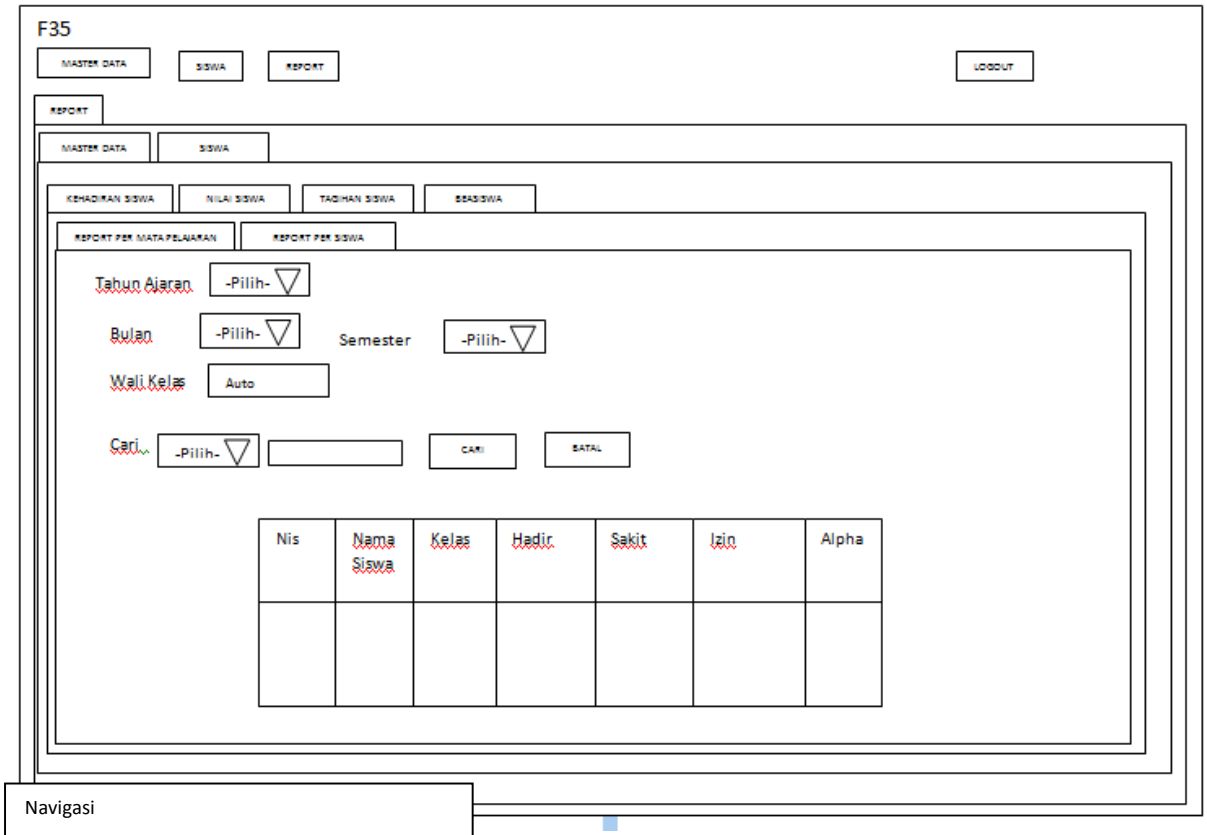
Gambar 3. 87 : Tab Kehadiran Siswa

44. Tab Report Per Kelas



Gambar 3. 88 : Tab Report Per Kelas

45. Tab report persiswa

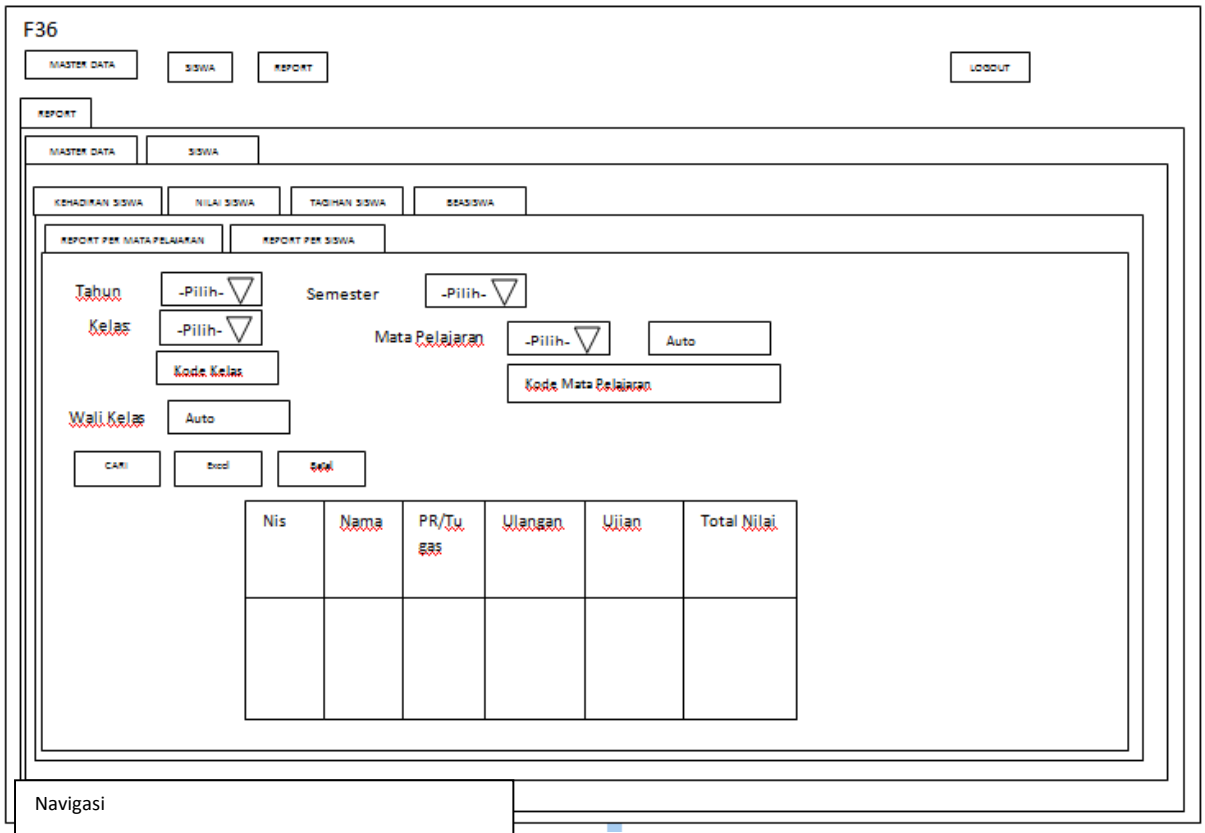


Navigasi

- Pilih kelas
- Jika data tidak ada maka akan muncul P59

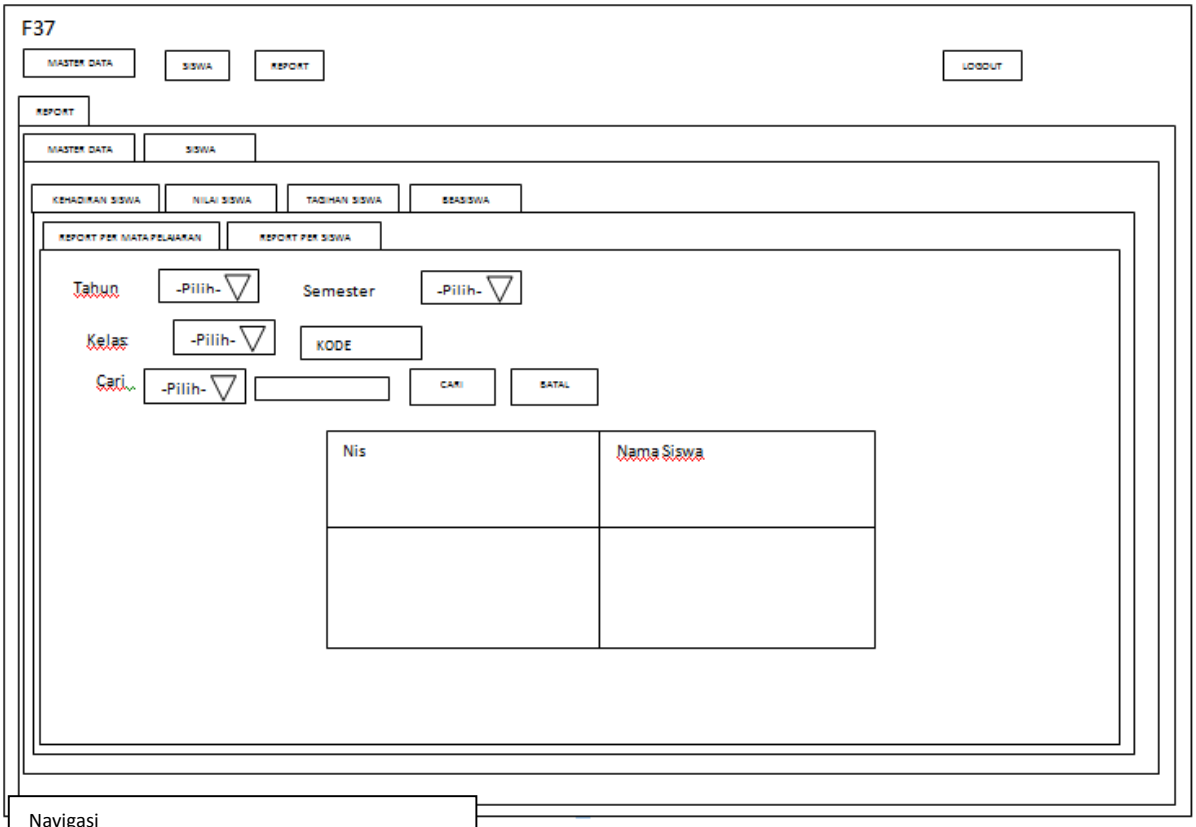
Gambar 3. 89 : Tab Report Persiswa

46. Tab Report Nilai Per mata Pelajaran



Gambar 3. 90 : Tab Report Per Mata Pelajaran

47. Tab report nilai per siswa

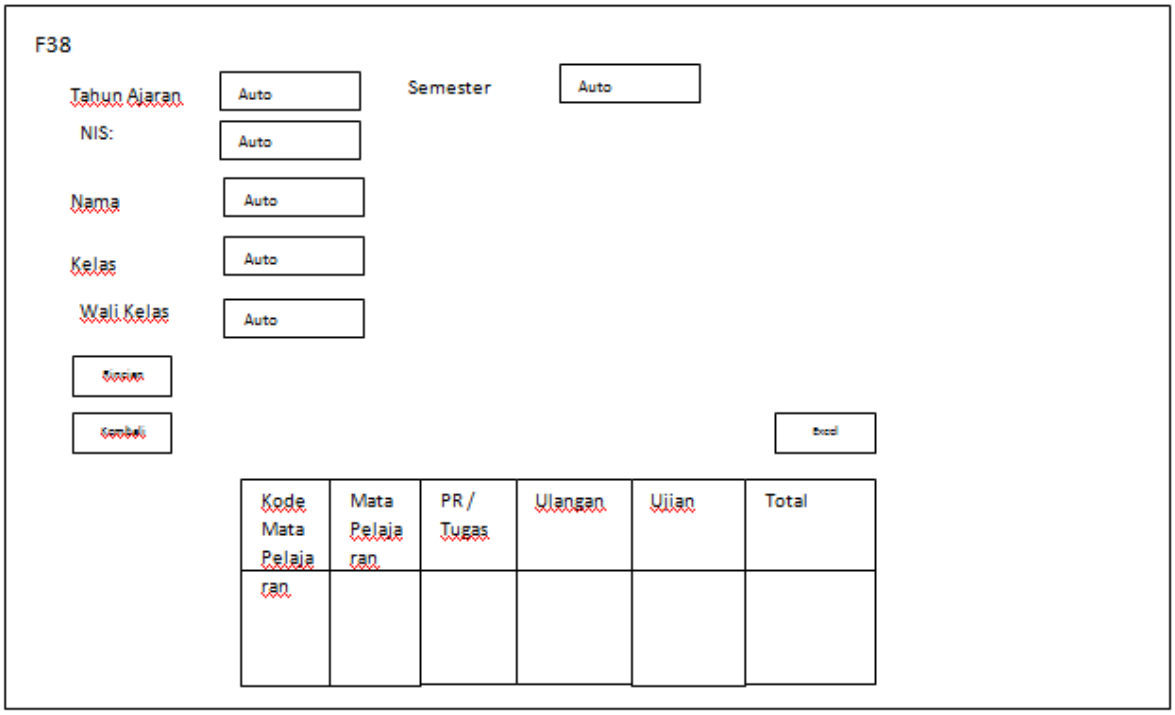


Navigasi

- Pilih kelas
- Jika data tidak ada maka akan muncul P59
- Jika mengklik salah satu nama ditabel maka akan muncul F38

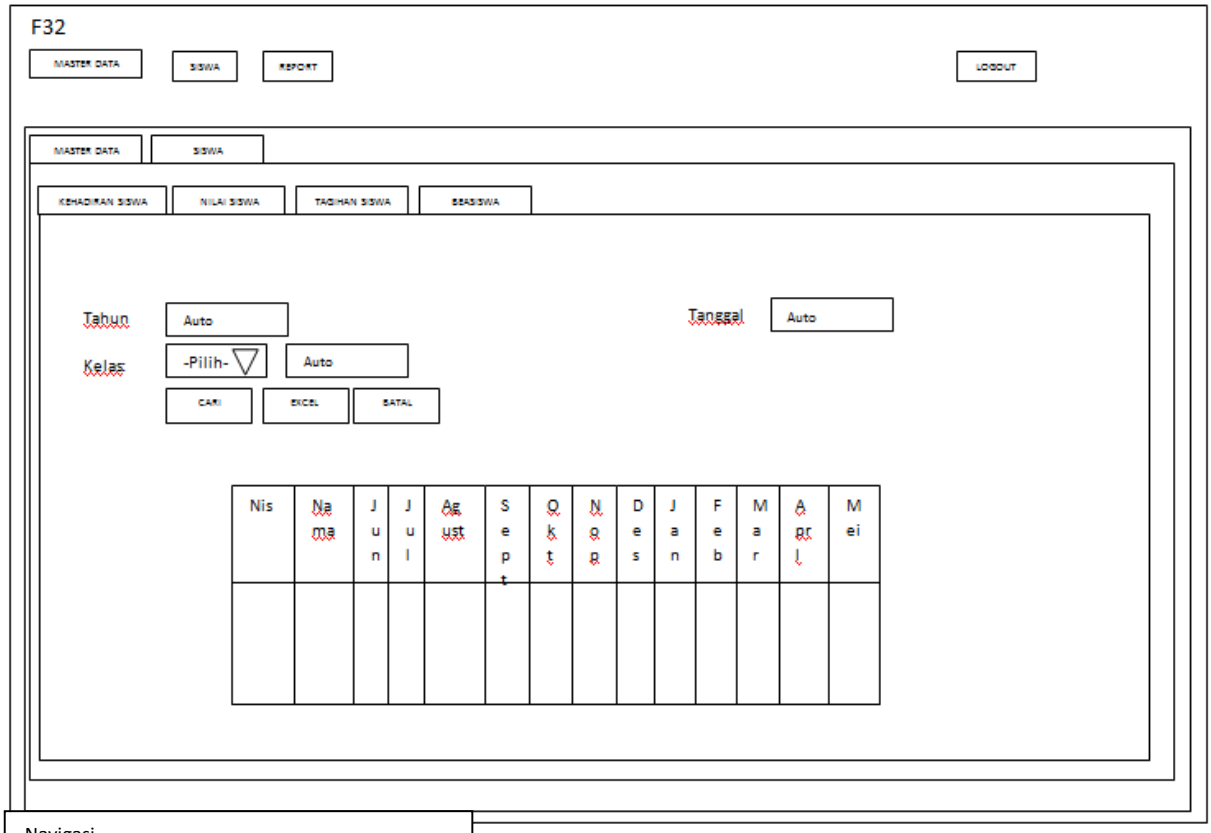
Gambar 3. 91 : Tab Report Persiswa

48. Form Detail Nilai Siswa



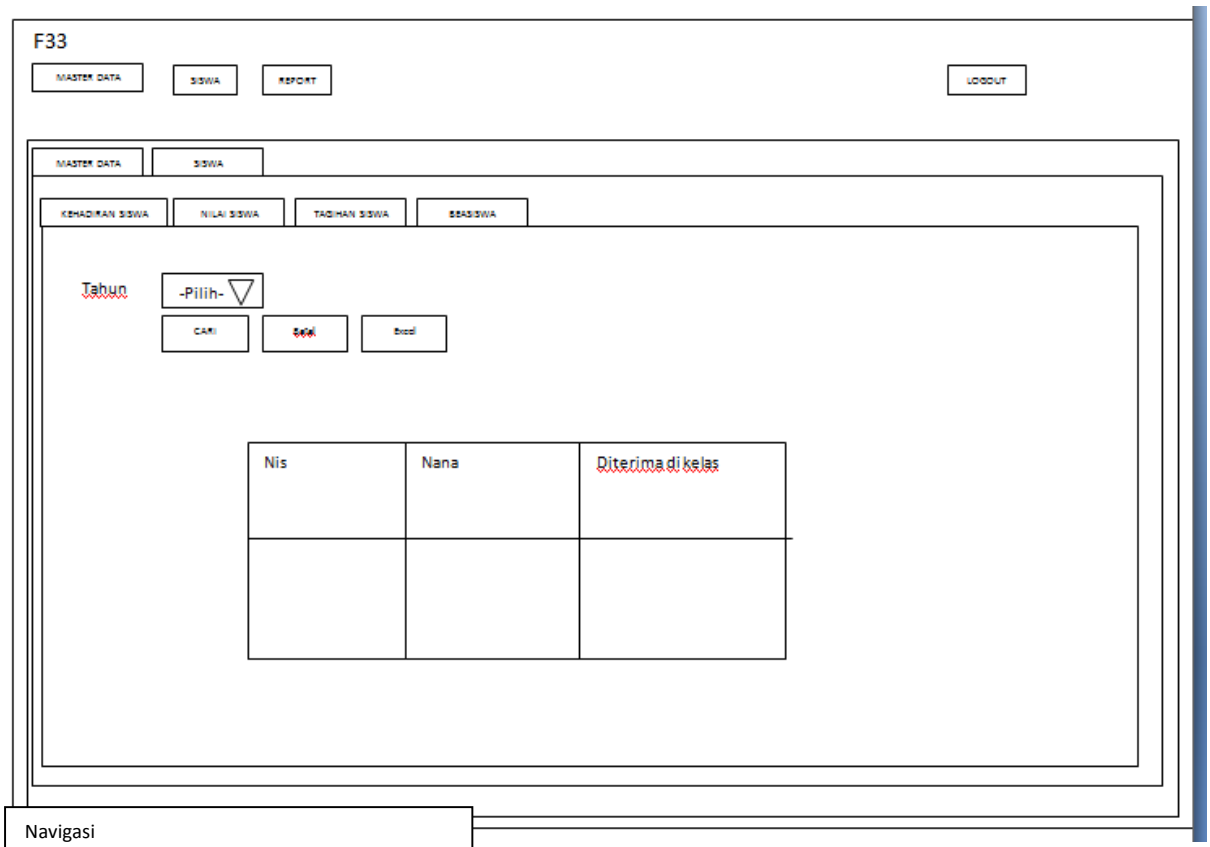
Gambar 3. 92 : Detail Nilai Siswa

49. Tab tagihan



Gambar 3. 93 : Tab Tagihan

50. Tab Beasiswa



Navigasi

- Pilih kelas
- Jika data tidak ada maka akan muncul P59

Gambar 3. 94 : Tab Beasiswa

3.2.5. Perancangan Pesan

Pada gambar di bawah ini merupakan perancangan pesan yang terdapat pada sistem ini. Adapun tampilan pesan yang ada adalah sebagai berikut:



211

P16

Nomor Telp / HP guru belum diisi

OK

P17

nis belum diisi

No

P18

nama belum diisi

No

P19

tempat lahir belum diisi

No

P20

tgl lahir belum diisi

No

P21

Agama belum dipilih

No

P22

No Telp / Hp belum diisi

No

P23

Nama Ayah belum diisi

No

P24

Pekerjaan ayah belum dipilih

No

P25

Pendidikan terakhir ayah belum dipilih

No

P26

Nama Ibu belum diisi

No

P27

Pekerjaan ibu belum dipilih

No

P28

Pendidikan terakhir ibu belum dipilih

No

P29

Penghasilan orang tua belum diisi

No

P30

Alamat belum diisi

No

P31

kota belum diisi

No

P32

Provinsi belum dipilih

No

P33

kode pos belum diisi

No

P34

Kode Guru belum dipilih

Ok

P35

Kelas belum dipilih

ok

P36

Nama guru belum dipilih

Ok

P37

Mata Pelajaran belum dipilih

Ok

P38

Kelas belum dipilih

Ok

P39

Jadwal belum dipilih

Ok

P40

Waktu belum dipilih

Ok

P41

Nama pegawai belum dipilih

Ok

P42

Username belum diisi

Ok

P43

Password belum diisi

Ok

P44

Re-password belum diisi

Ok

P45

No hp belum diisi

Ok

P46

Alamat belum diisi

Ok

P47

Kode TU belum diisi

Ok

P48

Nama pegawai TU belum diisi

Ok

P49

NIP belum diisi

Ok

P50

Jenis kelamin belum dipilih

Ok

P51

Golongan belum dipilih

Ok

P52

Status belum dipilih

Ok

P53

Tugas belum diisi

Ok

P54

No telp/hp belum diisi

Ok

P55

Tanggal Belum diisi

OK

P56

Bulan Belum Di pilih

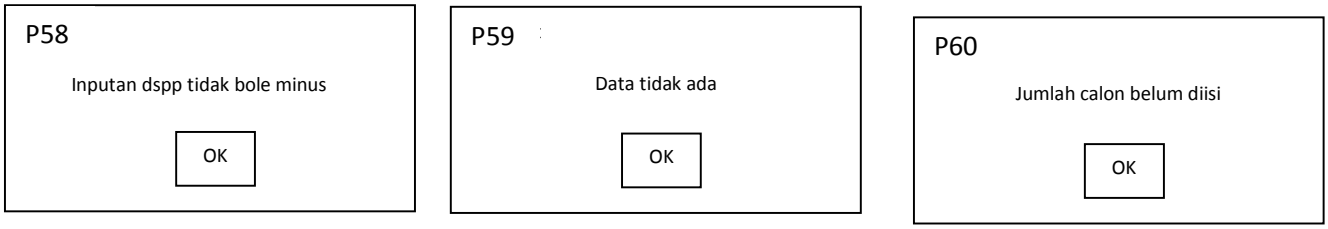
OK

P57

Inputan dsp tidak bole minus

OK

213

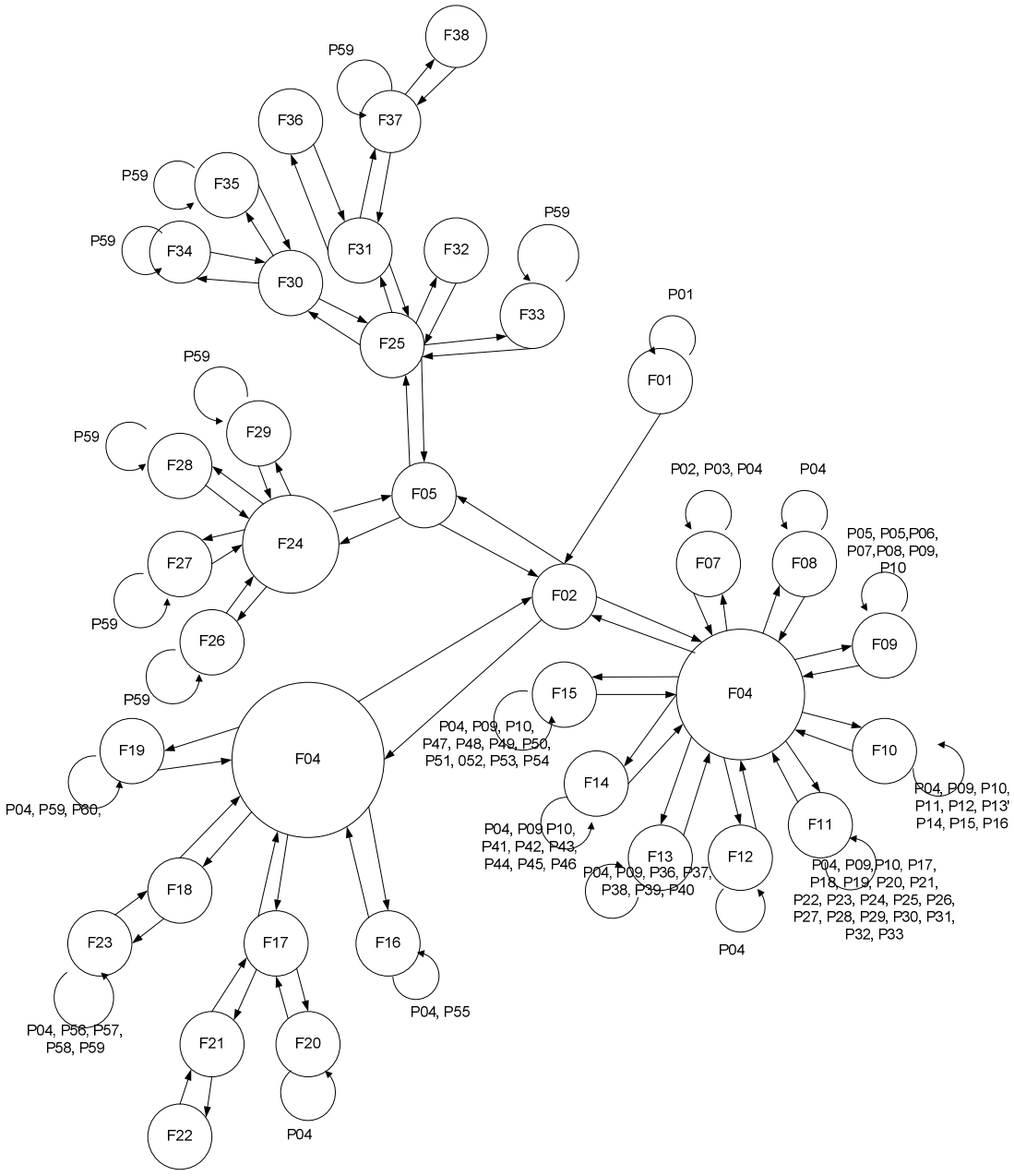


Gambar 3. 95 : Perancangan Pesan

3.2.6. Jaringan Semantik

Jaringan semantik menggambarkan keterhubungan nagivasi menu dari satu halaman ke halaman lainnya. Jaringan semantik pada sistem ini terbagi menjadi 2 bagian yaitu jaringan semantik untuk tampilan TU (admin) dan jaringan semantik untuk Guru.

1. Jaringan Semantik Menu TU (Admin)



Gambar 3. 96 : Jaringan Semantik Menu TU (Admin)

Keterangan

F01 : Login

F02 : Menu Admin

F04 : Menu Master Data

F05 : Menu Siswa

F06 : Menu Report

F07 : Tab Tahun Ajaran

F08 : Tab Kelas

F09 : Tab Mata Pelajaran

F10 : Tab Guru

F11 : Tab Siswa

F12 : Tab Wali Kelas

F13 : Tab Mengajar

F14 : Tab User

F15 : Tab TU

F16 : Tab Kehadiran Siswa

F17 : Tab Nilai Siswa

F18 : Tab Tagihan

F19 : Tab Beasiswa

F20 : Tab Input Nilai

F21 : Tab Hitung Nilai

F22 : Detail Nilai Siswa

F23 : Menu Input iuran siswa

F24 : Tab Master Data

F25 : Tab Siswa

F26 : Tab Siswa

F27 : Tab guru

F28 : Tab Orang Tua

F29 : Tab Mengajar

F30 : Tab Kehadiran Siswa

F31 : Tab Nilai Siswa

F32 : Tab Tagihan

F33 : Tab Beasiswa

F34 : Report Per Kelas

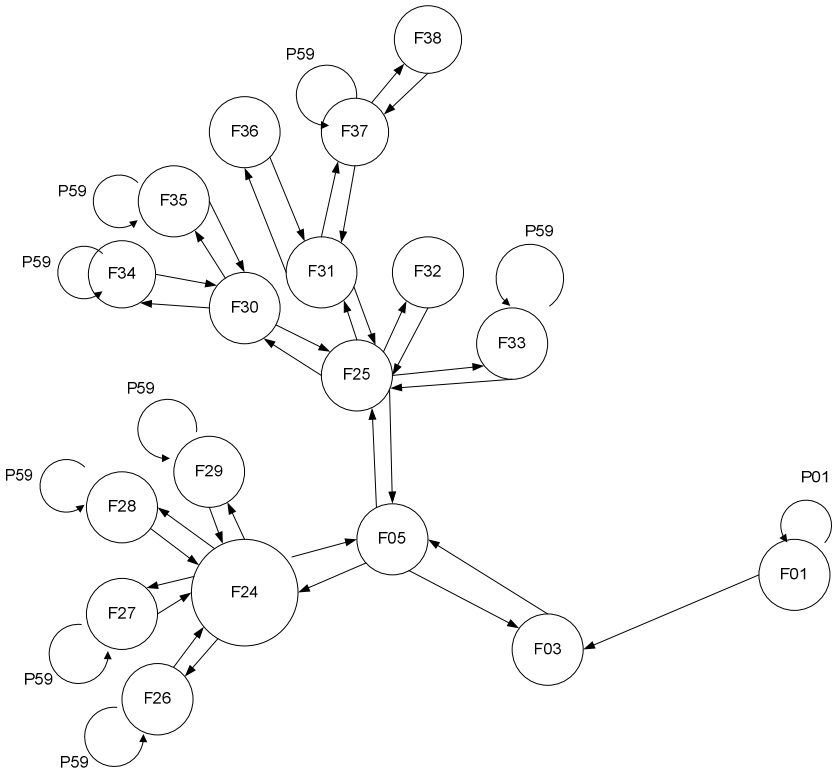
F35 : Report Per Siswa

F36 : Report Nilai Per Mata Pelajaran

F37 : Report Nilai Per Siswa

F38 : Detail Nilai siswa

2. Jaringan Semantik Menu Guru



Gambar 3. 97 : Jaringan Semantik Menu Guru

Keterangan :

F01 : Login

F03 : Menu Guru

F05 : Menu Report

F24 : Tab Master Data

F25 : Tab Siswa

F26 : Tab Siswa

F27 : Tab guru

F28 : Tab Orang Tua

F29 : Tab Mengajar

F30 : Tab Kehadiran Siswa

F31 : Tab Nilai Siswa

F32 : Tab Tagihan

F33 : Tab Beasiswa

F34 : Report Per Kelas

F35 : Report Per Siswa

F36 : Report Nilai Per Mata Pelajaran

F37 : Report Nilai Per Siswa

F38 : Detail Nilai siswa

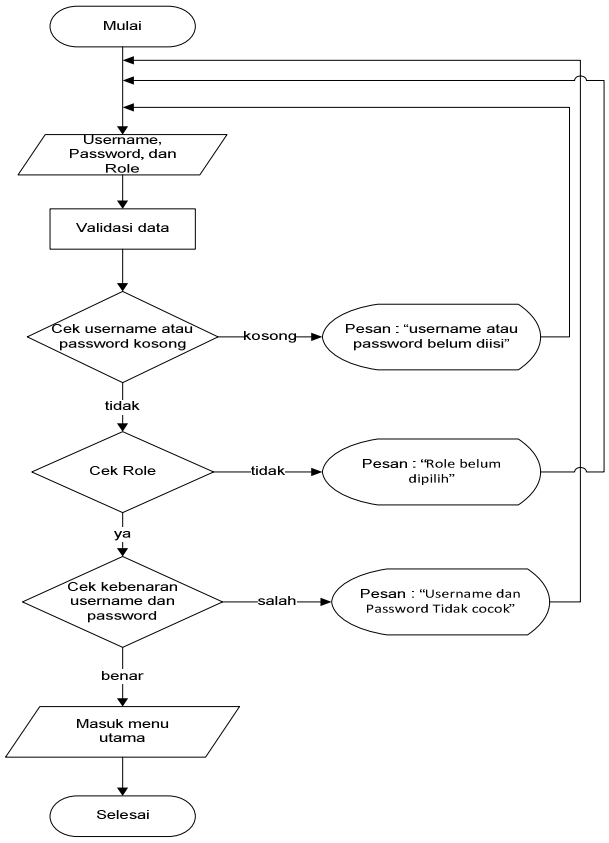
3.2.7. Perancangan *Method*

Sebagai langkah terakhir dalam perancangan yaitu perancangan *method* yang akan diimplementasikan ke dalam sistem. *Method* ini akan digunakan sebagai algoritma dasar dalam

mengkodekan prosedur yang ada. Adapun perancangan *method* yang ada di sistem ini adalah sebagai berikut :

1. Perancangan *Method Login*

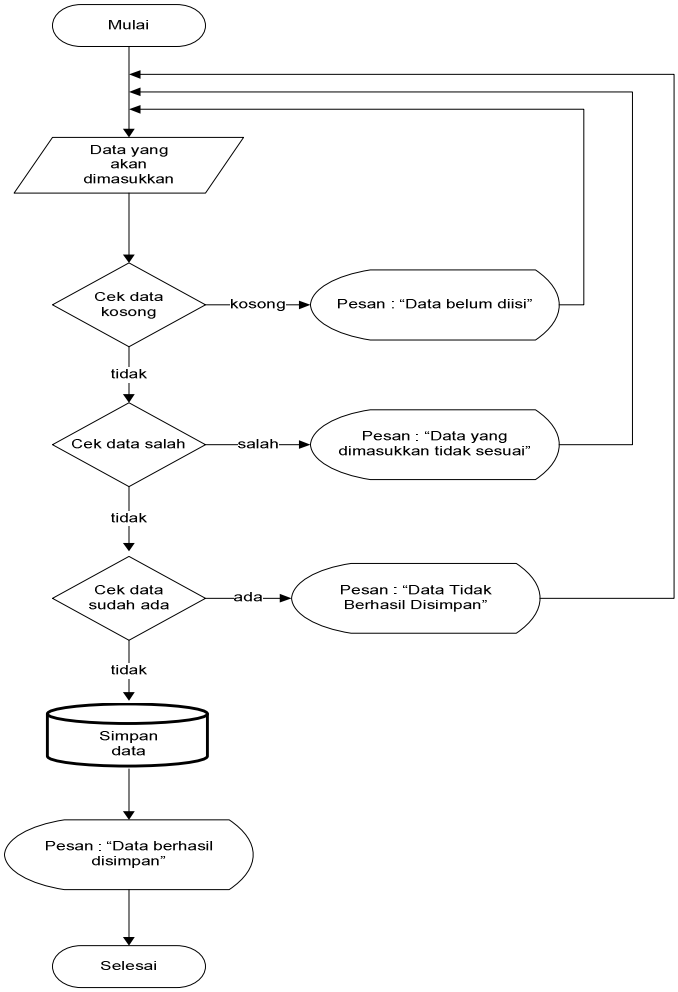
   Method ini dipanggil ketika TU (admin) dan guru akan mengakses aplikasi.



Gambar 3. 98 : *Method Login*

2. Perancangan *Method* Tambah Data

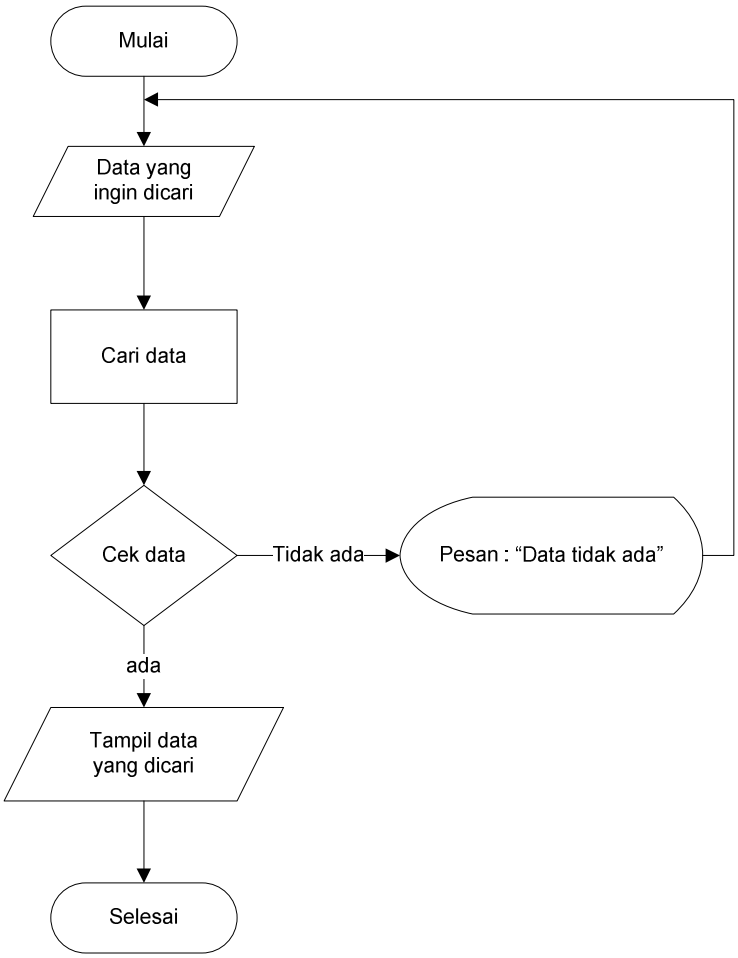
   Method ini dipanggil ketika *user* melakukan penambahan data. Seperti penambahan data tahun ajaran, data kelas, data mata pelajaran, data guru, data TU, data siswa, data orang tua, data mengajar guru, data wali kelas, data nilai siswa, data kehadiran siswa, dan data iuran siswa.



Gambar 3. 99 : *Method* Tambah Data

3. Perancangan Method Cari Data

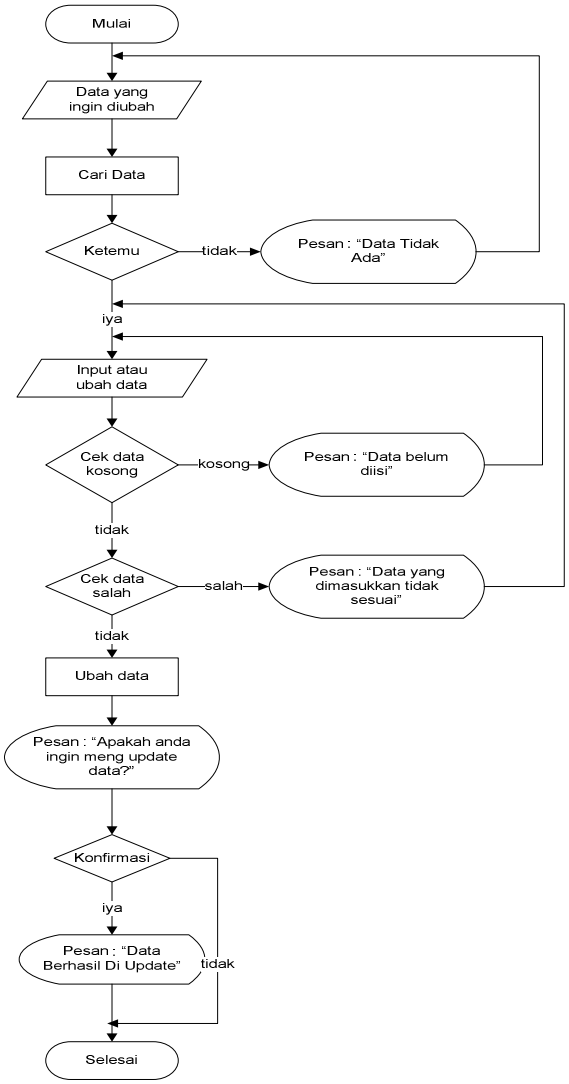
Method ini dipanggil ketika *user* melakukan pencarian data. Seperti pencarian data tahun ajaran, data kelas, data mata pelajaran, data guru, data TU, data siswa, data orang tua, data mengajar guru, data wali kelas, data nilai siswa, data kehadiran siswa, dan data iuran siswa.



Gambar 3. 100 : *Method* Cari Data

4. *Method* Ubah Data

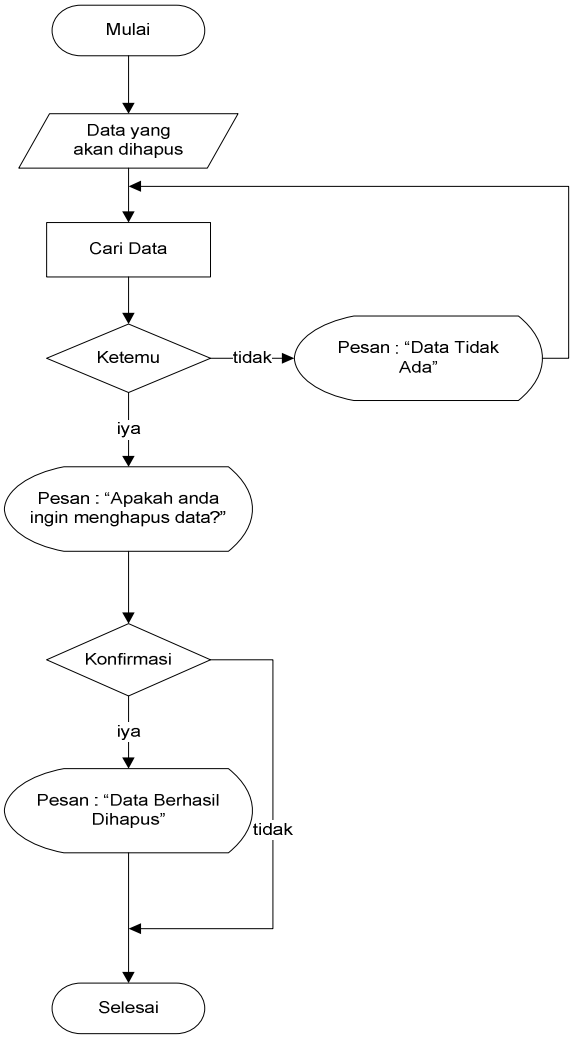
   Method ini dipanggil ketika user melakukan perubahan data. Seperti perubahan data mata pelajaran, data guru, data TU, data siswa, data orang tua, data mengajar guru, data wali kelas dan data nilai siswa.



Gambar 3. 101 : *Method* Ubah Data

5. *Method* Hapus Data

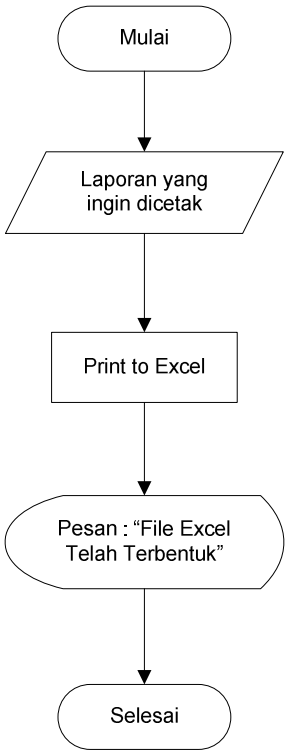
Method ini dipanggil ketika user melakukan penghapusan data. Seperti penghapusan data mata pelajaran, data guru, data TU, data siswa, dan data orang tua.



Gambar 3. 102 : *Method* Hapus Data

6. *Method* Cetak Laporan

Method ini dipanggil ketika user ingin mencetak laporan dalam file *excel*. Seperti laporan data siswa, data orang tua, data mengajar guru, data wali kelas data nilai siswa, data kehadiran, data tagihan siswa, data calon penerima beasiswa, dan data siswa yang telam menerima beasiswa.



Gambar 3. 103 : *Method* Cetak Laporan