BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1. Implementasi

Implementasi adalah suatu proses penerapan rancangan program yang telah dibuat ke dalam sebuah aplikasi pemrograman sesuai dengan tujuan yang diharapkan dari program aplikasi tersebut. Kegiatan implementasi dilakukan dengan dasar yang Ptelah direncanakan dalam tahap perancangan.

5.1.1. Batasan Implementasi

Dalam pengimplementasian perangkat lunak sistem informasi akademik ini terdapat beberapa hal yang menjadi batasan implementasi, yaitu :

1. Tidak semua kebutuhan sistem informasi akademik dikerjakan, tetapi hanya subsistem-subsistem yang mendukung.
2. *Database* yang digunakan dalam *client-server* menggunakan heidisql. Adapun jembatan penghubung dari aplikasi ke *database* menggunakan ODBC.
3. Alamat IP dari komputer yang digunakan merupakan alamat IP statis.

5.1.2. Implementasi Perangkat Lunak

Perangkat lunak (*Software*) digunakan untuk mendukung sistem operasi (SO) dan aplikasi *Database*. Perangkat lunak yang dibutuhkan oleh Komputer

server maupun Komputer *client*, adapun kebutuhan perangkat lunak yang digunakan sebagai berikut :

1. Komputer *Server*

   Komputer *server* adalah komputer yang mempunyai kemampuan yang lebih dari komputer *client*, dimana didalamnya tersimpan data-data yang akan didistribusikan. Dimana software yang dibutuhkan diantaranya :

   a. Sistem Operasi Windows XP SP 2
   b. Aplikasi SQL Server 2000
   c. Bahasa Pemrograman Java
   d. ODBC dan Heidisql
   e. iReport-3.7.5

2. Komputer *Client*

   Komputer *Client* adalah komputer yang didalamnya tersimpan data-data yang dapat digunakan secara perorangan, mengumpulkan dan menampilkan data, serta menyimpan data ke *server*. Dimana software yang dibutuhkan diantaranya :

   a. Sistem Operasi Windows XP SP 2
   b. ODBC dan Heidisql

5.1.3. Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus terpenuhi untuk dapat menjalankan sistem aplikasi ini antara lain:

1. Komputer *Server*

a. Menggunakan minimal processor Intel Pentium 4 atau yang sekelasnya.
b. RAM minimal 128Mb.
c. *Hard Drive* untuk media penyimpanan, minimal 10Mb untuk *server* diluar basisdata.
d. *Mouse, keyboard* dan Monitor sebagai alat antar muka.

2. Komputer *Client*
   a. Menggunakan minimal processor Intel Pentium 4 atau yang sekelasnya
   b. RAM minimal 64Mb
   c. *Mouse, keyboard,* dam monitor sebagai alat antar muka

5.1.4. Implementasi Basis Data

Implementasi basis data dilakukan dengan menggunakan bahasa SQL, dimana aplikasi pemrograman yang digunakan adalah MySQL, implementasi basis datanya dalam bahas SQL adalah sebagai berikut :

```
-----------------------------------------------------
--
-- Table structure for table `daftar`
--
CREATE TABLE IF NOT EXISTS `daftar` (
`NoPendaftar` varchar(7) NOT NULL,  `nama` varchar(30) NOT NULL,
`AsalSekolah` varchar(30) NOT NULL,  `NoSTTB` varchar(12) NOT NULL,
```

```
`Nem` varchar(5) NOT NULL, `TempatLahir` varchar(25) NOT NULL, `TanggalLahir` varchar(10) NOT NULL, `GolonganDarah` varchar(2) NOT NULL, `AgamaCPD` varchar(20) NOT NULL, `Alamat` varchar(50) NOT NULL, `TelpCPD` varchar(15) NOT NULL, `PendidikanTerakhir` varchar(10) NOT NULL, `AlamatAS` varchar(50) NOT NULL, `NamaAyah` varchar(25) NOT NULL, `AlamatAyah` varchar(50) NOT NULL, `TelpAyah` varchar(15) NOT NULL, `PekerjaanAyah` varchar(15) NOT NULL, `AgamaAyah` varchar(20) NOT NULL, `NamaIbu` varchar(25) NOT NULL, `AlamatIbu` varchar(50) NOT NULL, `TelpIbu` varchar(15) NOT NULL, `PekerjaanIbu` varchar(15) NOT NULL, `AgamaIbu` varchar(20) NOT NULL, `NamaWali` varchar(25) NOT NULL, `PekerjaanWali` varchar(50) NOT NULL, `AlamatWali` varchar(50) NOT NULL, `AgamaWali` varchar(20) NOT NULL, `Pilihan1` varchar(15) NOT NULL, `Pilihan2` varchar(15) NOT NULL, `JenisKelamin` varchar(15) NOT NULL,
PRIMARY KEY (`NoPendaftar`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--------------------------------------------------------

--

-- Table structure for table `seleksi`

--

CREATE TABLE IF NOT EXISTS `seleksi` (
`NoPendaftar` varchar(8) NOT NULL, `nama` varchar(40) NOT NULL, `AsalSekolah` varchar(20) NOT NULL, `NoSTTB` varchar(8) NOT NULL,
```

```
`Nem` varchar(5) NOT NULL, `TempatLahir` varchar(15) NOT NULL, `TanggalLahir` varchar(15) NOT NULL, `GolonganDarah` varchar(2) NOT NULL, `AgamaCPD` varchar(25) NOT NULL, `Alamat` varchar(30) NOT NULL, `TelpCPD` varchar(15) NOT NULL, `PendidikanTerakhir` varchar(5) NOT NULL, `AlamatAS` varchar(50) NOT NULL, `NamaAyah` varchar(25) NOT NULL, `AlamatAyah` varchar(50) NOT NULL, `TelpAyah` varchar(15) NOT NULL, `PekerjaanAyah` varchar(50) NOT NULL, `AgamaAyah` varchar(25) NOT NULL, `NamaIbu` varchar(25) NOT NULL, `AlamatIbu` varchar(50) NOT NULL, `TelpIbu` varchar(15) NOT NULL, `PekerjaanIbu` varchar(25) NOT NULL, `AgamaIbu` varchar(20) NOT NULL, `NamaWali` varchar(25) NOT NULL, `PekerjaanWali` varchar(25) NOT NULL, `AlamatWali` varchar(25) NOT NULL, `AgamaWali` varchar(25) NOT NULL, `Pilihan1` varchar(25) NOT NULL, `Pilihan2` varchar(25) NOT NULL, `JenisKelamin` varchar(25) NOT NULL, `Status` varchar(20) NOT NULL, FOREIGN KEY (`NoPendaftar`))
ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
----------------------------------------------------
--
-- Table structure for table `biodatasiswa`
--
CREATE TABLE IF NOT EXISTS `biodatasiswa` (
```

`NoPendaftar` varchar(10) NOT NULL, `nama` varchar(50) NOT NULL, `TempatLahir` varchar(20) NOT NULL, `TanggalLahir` varchar(10) NOT NULL, `JenisKelamin` varchar(15) NOT NULL, `AgamaCPD` varchar(10) NOT NULL, `AnakKe` varchar(5) NOT NULL, `SaudaraKandung` varchar(5) NOT NULL, `StatusAnak` varchar(20) NOT NULL, `Alamat` varchar(50) NOT NULL, `KodePos` varchar(8) NOT NULL, `Telp` varchar(15) NOT NULL, `TanggalDiterima` varchar(10) NOT NULL, `AsalSekolah` varchar(20) NOT NULL, `NoSKHUN` varchar(25) NOT NULL, `LamaBelajar` varchar(10) NOT NULL, `NilaiSKHUN` varchar(10) NOT NULL, `TahunAjaran` varchar(10) NOT NULL, `NIS` varchar(9) NOT NULL, `GolonganDarah` varchar(3) NOT NULL, `NamaAyah` varchar(25) NOT NULL, `AlamatAyah` varchar(50) NOT NULL, `TelpAyah` varchar(15) NOT NULL, `PekerjaanAyah` varchar(25) NOT NULL, `AgamaAyah` varchar(20) NOT NULL, `NamaIbu` varchar(25) NOT NULL, `AlamatIbu` varchar(50) NOT NULL, `TelpIbu` varchar(15) NOT NULL, `PekerjaanIbu` varchar(50) NOT NULL, `AgamaIbu` varchar(25) NOT NULL, `Kwn` varchar(5) NOT NULL, `SaudaraTiri` varchar(5) NOT NULL, `SaudaraAngkat` varchar(5) NOT NULL, `StatusYP` varchar(10) NOT NULL, `Bahasa` varchar(50) NOT NULL, `PenanggungBiaya` varchar(20) NOT NULL, `TinggalDi` varchar(25) NOT NULL, `Jarak` varchar(5) NOT NULL, `TahunLulus` varchar(5) NOT NULL, `KelainanJasmani` varchar(100) NOT NULL, `Tinggi` varchar(5) NOT NULL, `Berat` varchar(5) NOT NULL, `Riwayat` varchar(100) NOT NULL, `Hobi` varchar(50) NOT NULL,

```
`TempatAyah` varchar(20) NOT NULL, `TanggalAyah` varchar(20) NOT NULL, `KwnAyah` varchar(10) NOT NULL, `GolonganAyah` varchar(50) NOT NULL, `PendidikanAyah` varchar(25) NOT NULL, `PenghasilanAyah` varchar(25) NOT NULL, `StatusAyah` varchar(25) NOT NULL, `NikahAyah` varchar(25) NOT NULL, `MeninggalAyah` varchar(5) NOT NULL, `TempatIbu` varchar(20) NOT NULL, `TanggalIbu` varchar(20) NOT NULL, `KwnIbu` varchar(10) NOT NULL, `PendidikanIbu` varchar(25) NOT NULL, `GolonganIbu` varchar(50) NOT NULL, `PenghasilanIbu` varchar(20) NOT NULL, `StatusIbu` varchar(15) NOT NULL, `NikahIbu` varchar(10) NOT NULL, `MeninggalIbu` varchar(10) NOT NULL)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
--------------------------------------------------------
--
-- Table structure for table `detailsiswa`
--
CREATE TABLE IF NOT EXISTS `detailsiswa` (
`NIS` varchar(15) NOT NULL,  `nama_siswa` varchar(40) NOT NULL, `ta` varchar(11) NOT NULL, `kelas` varchar(11) NOT NULL)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
--------------------------------------------------------
--
-- Table structure for table `pengajar`
--
```

```
CREATE TABLE IF NOT EXISTS `pengajar` (
`NIP` varchar(25) NOT NULL, `nama_guru` varchar(50) NOT NULL, `TempatLahir` varchar(20) NOT NULL, 'TanggalLahir` varchar(20) NOT NULL, `TahunLahir` varchar(20) NOT NULL, `alamat` varchar(50) NOT NULL, `telp` varchar(15) NOT NULL, `TanggalP` varchar(20) NOT NULL, `TahunP` varchar(5) NOT NULL, `TahunMasuk` varchar(5) NOT NULL, `PendidikanTerakhir` varchar(25) NOT NULL, `BidangStudy` varchar(50) NOT NULL, `JenisKelamin` varchar(20) NOT NULL)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
--------------------------------------------------------
--
-- Table structure for table `matpel`
--
CREATE TABLE IF NOT EXISTS `matpel` (
`TahunAjaran` varchar(12) NOT NULL, `MataPelajaran` varchar(50) NOT NULL, `KodePelajaran` varchar(6) NOT NULL)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
--------------------------------------------------------
--
-- Table structure for table `guru`
--
CREATE TABLE IF NOT EXISTS `guru` (
```

```
`NIP` varchar(11) NOT NULL, `nama_guru` varchar(25) NOT NULL, `alamat` varchar(30) NOT NULL, `telp` varchar(12) NOT NULL, `JenisKelamin` varchar(11) NOT NULL, `BidangStudy` varchar(30) NOT NULL, `KodePelajaran` varchar(20) NOT NULL, `TahunMasuk` varchar(11) NOT NULL, PRIMARY KEY (`NIP`))
ENGINE=InnoDB DEFAULT CHARSET=latin1;
--------------------------------------------------------
--
-- Table structure for table `jambelajar`
--
CREATE TABLE IF NOT EXISTS `jambelajar` (
`Hari` varchar(8) NOT NULL, `Ke` varchar(3) NOT NULL, `Jam` varchar(12) NOT NULL) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--------------------------------------------------------
--
-- Table structure for table `jadwal`
--
CREATE TABLE IF NOT EXISTS `jadwal` (
`NIP` varchar(11) NOT NULL, `nama` varchar(30) NOT NULL, `kode_kelas` varchar(4) NOT NULL, `hari` varchar(6) NOT NULL, `jamKe` varchar(3) NOT NULL, `lamajam` varchar(13) NOT NULL, `bid_study` varchar(50) NOT NULL) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--------------------------------------------------------
```

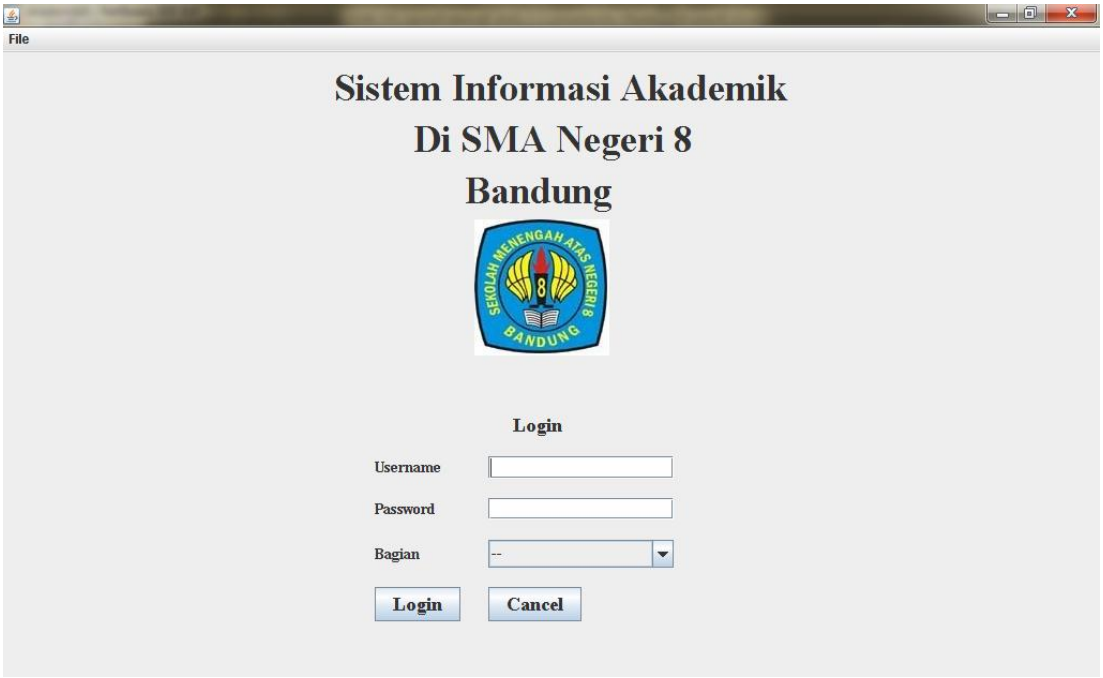
```
--
-- Table structure for table `nilai`
--
CREATE TABLE IF NOT EXISTS `nilai` (
`ta` varchar(11) NOT NULL, `NIS` varchar(11) NOT NULL, `nama_siswa` varchar(80) NOT NULL, `kode_kelas` varchar(6) NOT NULL, `NIP` varchar(11) NOT NULL, `nama_guru` varchar(80) NOT NULL, `mata_pelajaran` varchar(11) NOT NULL, `Semester` varchar(6) NOT NULL, `tugas` varchar(11) NOT NULL, `ulangan_harian` double NOT NULL, `UTS` double NOT NULL, `UAS` double NOT NULL, `NA` double NOT NULL, `sikap` varchar(5) NOT NULL, `PemahamanKonsep` double NOT NULL, `Komentar` varchar(200) NOT NULL)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
--------------------------------------------------------
--
-- Table structure for table `walikelas`
--
CREATE TABLE IF NOT EXISTS `walikelas` (
`TahunAjaran` varchar(15) NOT NULL, `kelas` varchar(3) NOT NULL, `nama_guru` varchar(20) NOT NULL, `NIP` varchar(8) NOT NULL, PRIMARY KEY (`NIP`))
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.5 Implementasi Program Untuk User

Berikut merupakan implementasi program untuk user, berikut ini adalah tampilan dari program untuk interface user:

1. Proses Login Untuk Pengguna

   Halaman ini merupakan halaman login untuk semua user. Sebagai contoh user login sebagai bagian Tata Usaha.



Gambar 5.1 Tampilan Login Pengguna

2. Halaman Sebagai Tata Usaha

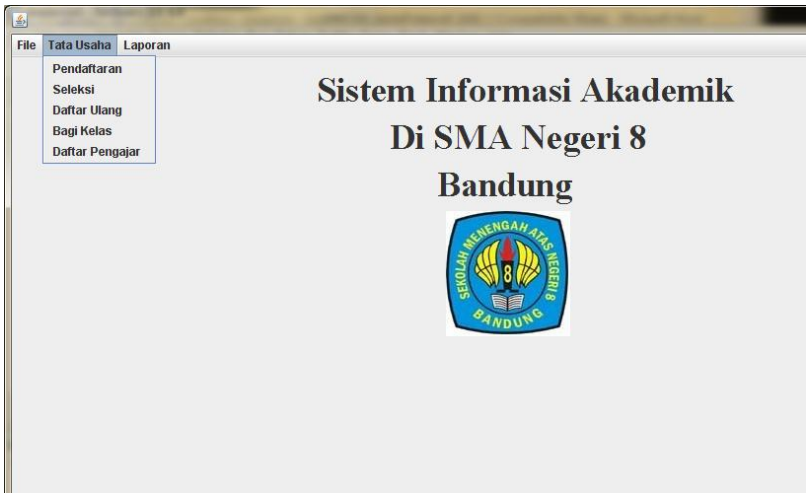
   Setelah user berhasil login sebagai Tata Usaha maka akan tampil halaman yang dapat di akses oleh bagian Tata Usaha, seperti gambar di bawah ini.



Gambar 5.2 Home Utama Tata Usaha

3. Halaman Menu Utama Sebagai Tata Usaha

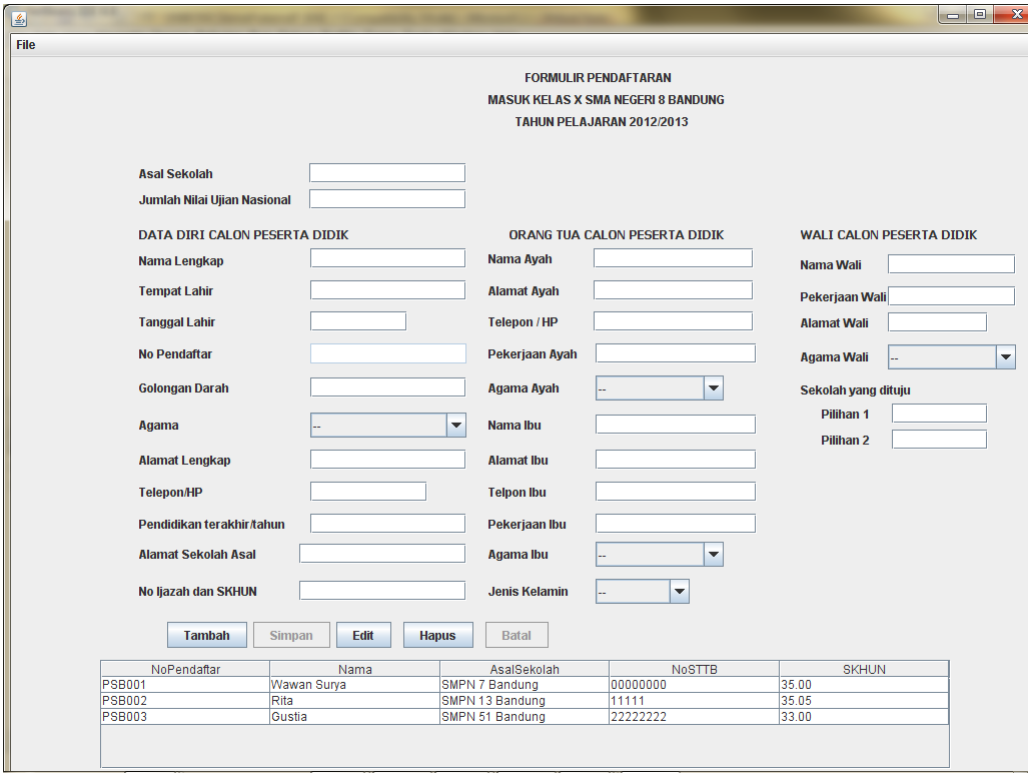
   Gambar 5.3 menampilkan menu-menu yang dapat diakses oleh Bagian Tata Usaha. Menu yang berada dalam tampilan Bagian Tata Usaha pada menu bar yaitu Pendaftaran, Seleksi, Daftar Ulang, Bagi Kelas, dan Data Pengajar. Dan di menu bar Laporan ada Nem untuk menentukan Nem yang menyeleksi calon siswa.



Gambar 5.3 Pilihan Menu Bag.Tata Usaha

4. Tampilan Pengisian Pendaftar Calon Siswa

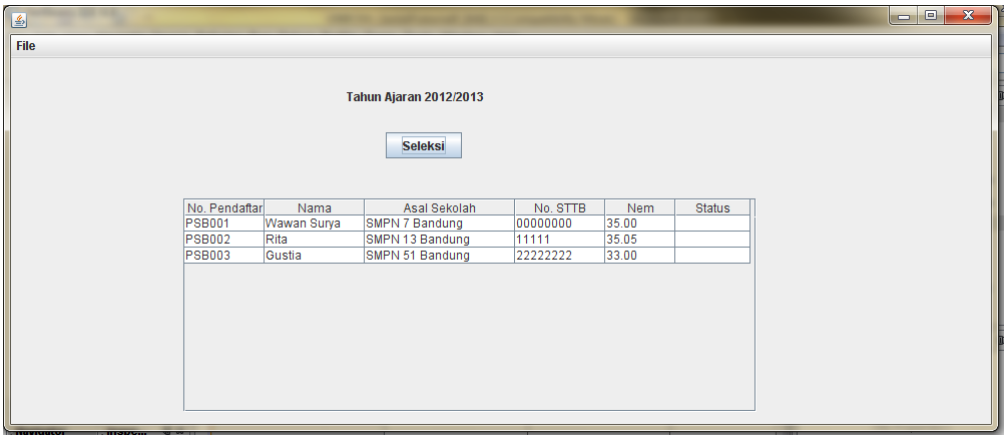
Gambar 5.4 merupakan tampilan untuk mengisi data calon siswa. Pertama-tama harus meng-klik tombol “Tambah” untuk mendapatkan no pendaftaran. Lalu melengkapi data pendaftar dan klik “Simpan”



Gambar 5.4 Proses Pendaftaran Calon Siswa Baru

5. Tampilan Seleksi Calon Siswa

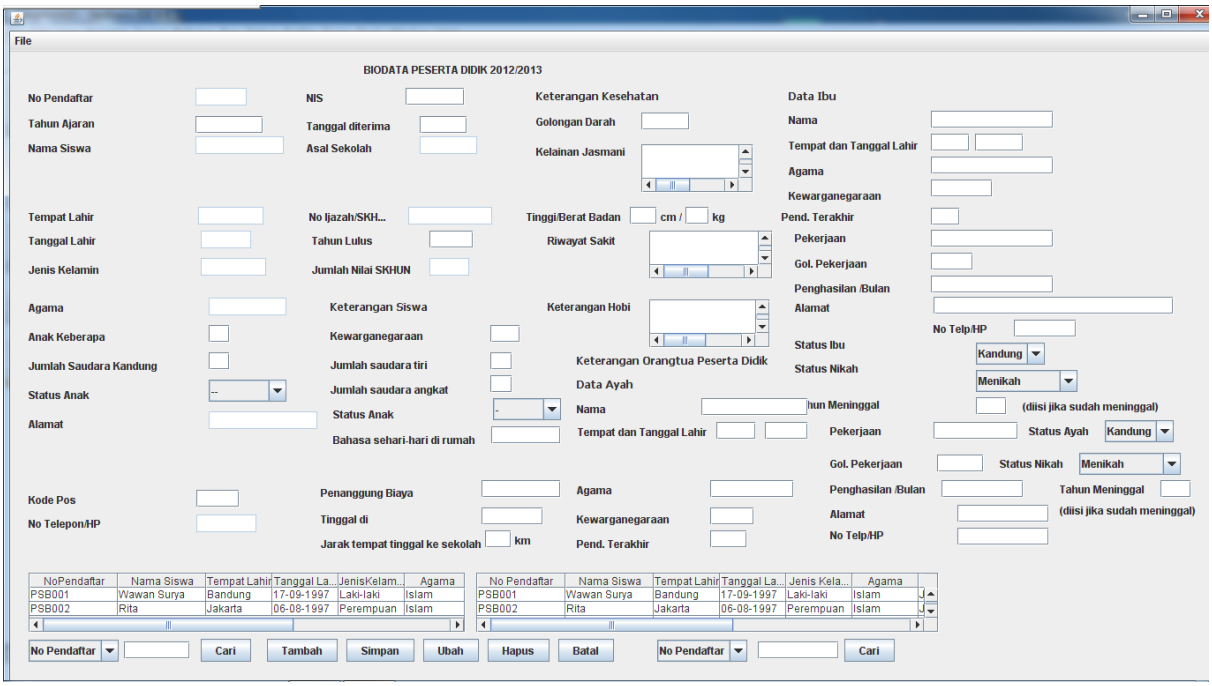
Tampilan pada gambar 5.5 adalah tampilan untuk menyeleksi calon siswa. Pada tampilan tersebut sudah ada beberapa atribut dan tabel calon siswa yang sudah terdaftar, pertama-tama hanya mengklik siswa yang sudah berada pada tabel. Maka data siswa yang sudah berada pada tabel dengan otomatis akan mengisi kolom-kolom yang sudah tersedia, lalu tinggal mengklik tombol seleksi dan mengklik tombol simpan.



Gambar 5.5 Proses Seleksi Calon Siswa

6. Tampilan Pendaftaran Ulang

Gambar 5.6 Proses Pendaftaran Ulang, proses dan tampilan disini yang berada pada tabel sebelah kanan atas adalah data siswa yang telah terseleksi diterima. Di tampilan sini pun pengguna tinggal mengklik data siswa yang sudah terpapar dalam tabel, maka beberapa kolom akan terisi dan beberapa kolom harus diisi kembali. Pada tampilan pendaftaran ulang ada beberapa tombol seperti tambah, simpan, ubah, hapus, dan batal.

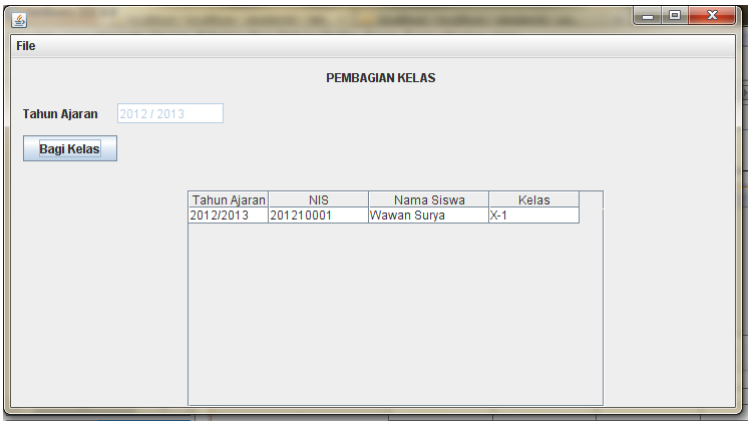


Gambar 5.6 Proses Pendaftaran Ulang

7. Tampilan Bagi Kelas

Pada tampilan bagi kelas, user tinggal mengklik tombol “Bagi Kelas”.

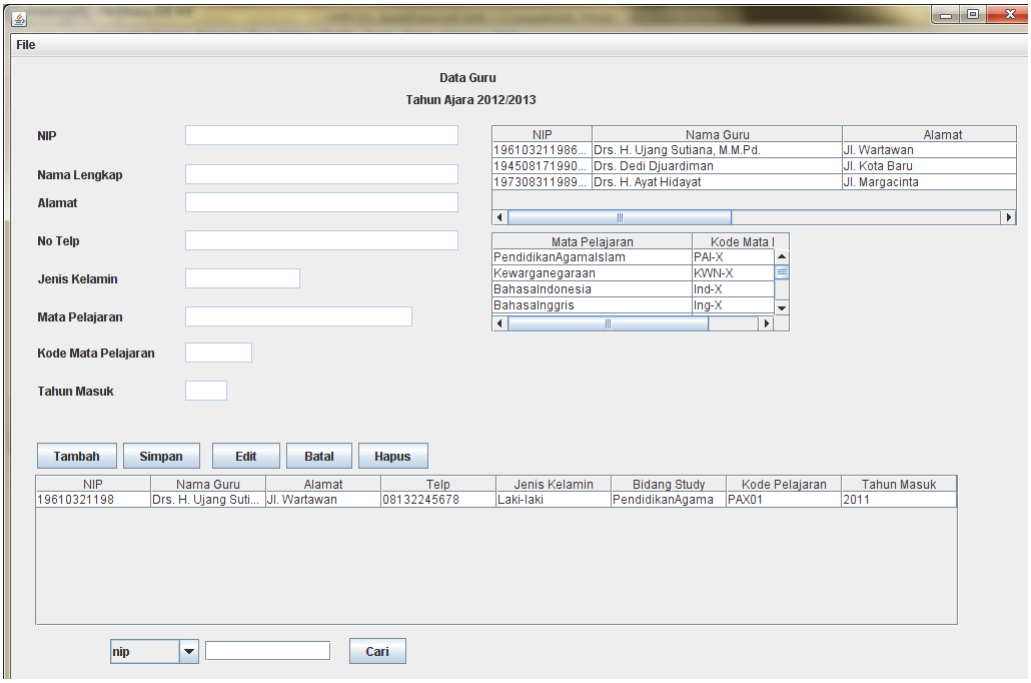
Tetapi siswa yang sudah mempunya NIS atau sudah terdaftar resmi.



Gambar 5.6 Tampilan Proses Bagi Kelas

8. Tampilan Data Pengajar

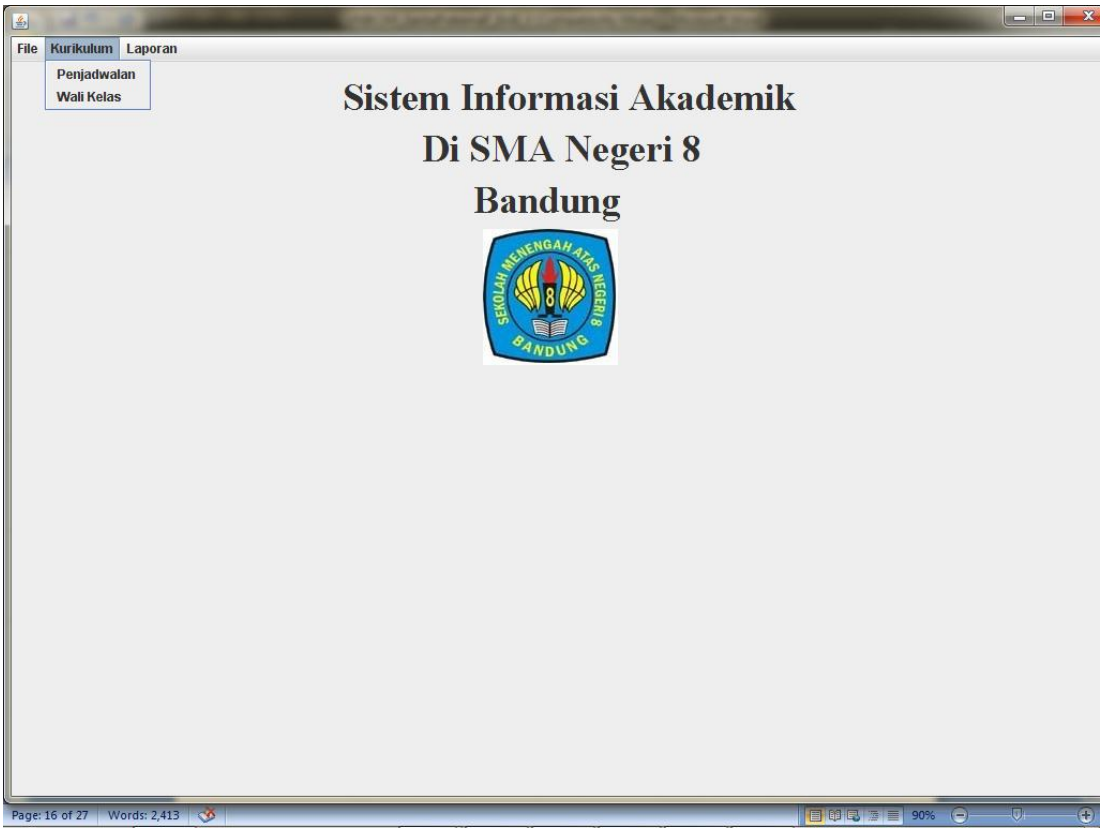
Pada tampilan ini, Bagian Tata Usaha dapat menginputkan data pengajar dengan lengkap.



Gambar 5.7 Tampilan Data Pengajar

9. Tampilan Kurikulum

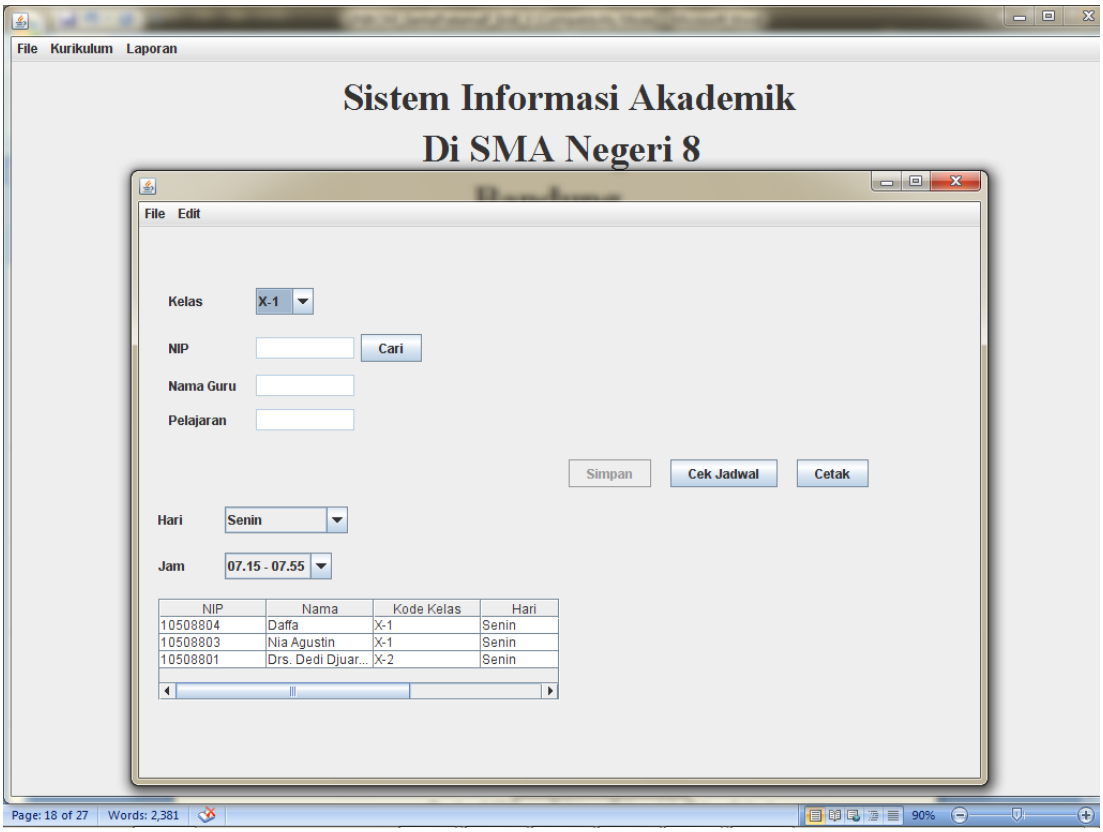
Tampilan berikut ini merupakan tampilan sebagai kurikulum. Bag. Kurikulum memiliki menu seperti penjadwalan dan wali kelas.



Gambar 5.8 Tampilan Sebagai Kurikulum

10. Tampilan Penjadwalan

Halaman ini merupakan tampilan untuk menginputkan jadwal pelajaran yang merupakan hak akses dari bagian kurikulum. Pertama, Bag. Kurikulum menginputkan semua field yang harus terisi, kemudian memilih tombol “Cek Jadwal“



Gambar 5.9 Halaman Proses Penjadwalan

Ketika menekan Tombol “Cek Jadwal” maka akan muncul pesan seperti berikut ini :

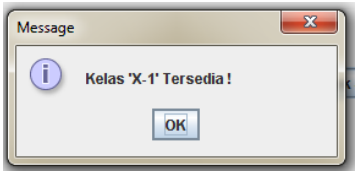


Gambar 5.10 Pesan Pelajaran Belum Ada Dalam Jadwal

Setelah itu, Jika pelajaran bisa diisi maka sistem akan menampilkan pesan Guru bisa mengajar atau tidak , dan selanjutnya akan mengecek ketersediaan kelas.



Gambar 5.11 Pesan Belum Mengajar

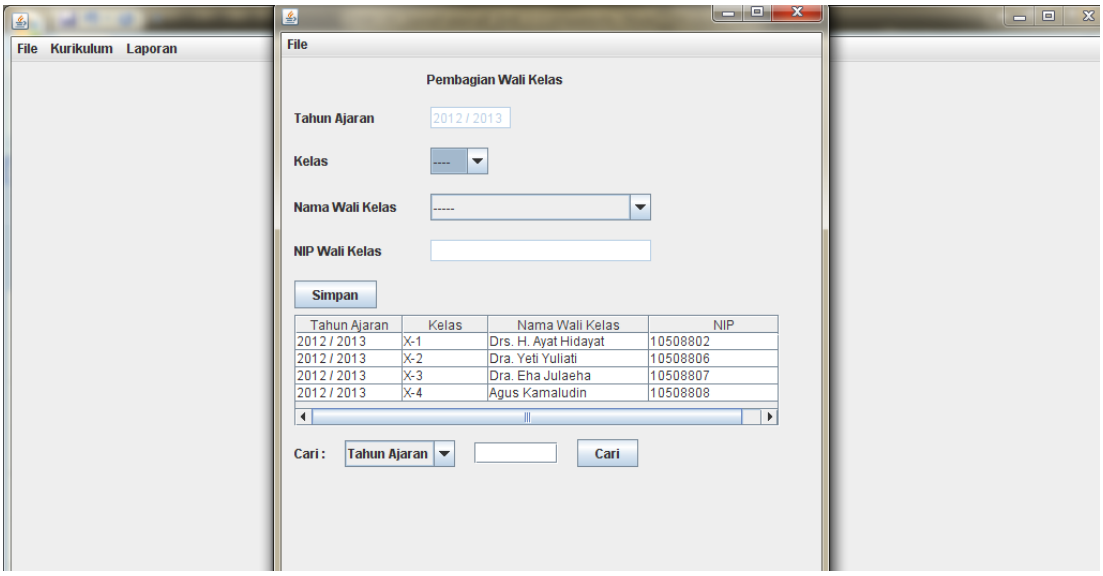


Gambar 5.12 Pesan Kelas Tersebut Belum Ada Yang Mengajar

Jika muncul pesan dialog “Kelas Tersedia“ seperti pada gambar diatas, maka user Bag. Kurikulum dapat menyimpan jadwal pelajaran dengan mengklik “Simpan”. Setelah tersimpan maka jadwal yang sudah dibuat akan ditampilkan di tabel.

11. Tampilan Wali Kelas

Pada tampilan gambar 5.13 proses wali kelas, kolom tahun ajaran akan otomatis mengikuti tahun ajaran yang akan berjalan. Dan tombol pilihan kelas adalah untuk memilih kelas, juga tombol nama guru yang akan dijadikan wali kelas. Jika kelas belum memiliki wali kelas maka tinggal mengklik tombol simpan.



Gambar 5.13 Proses Wali Kelas

12. Tampilan Menu Data Guru

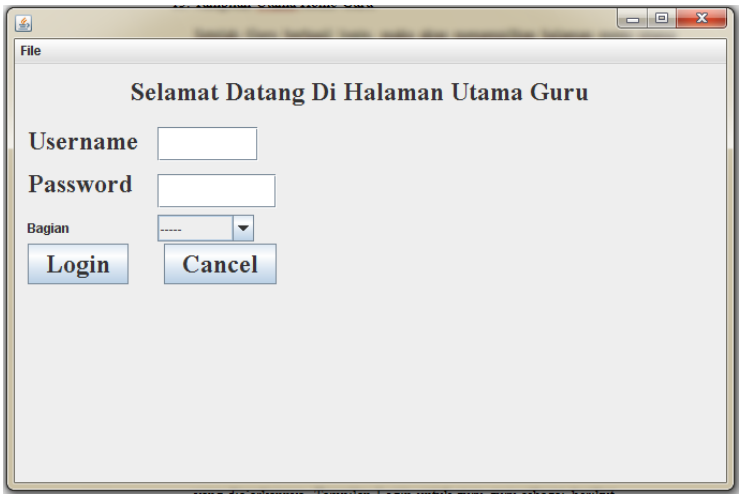
Halaman ini untuk mencari data guru yang aktif serta matapelajarannya, untuk menyusun jadwal pelajaran.



Gambar 5.14 Halaman Data Guru

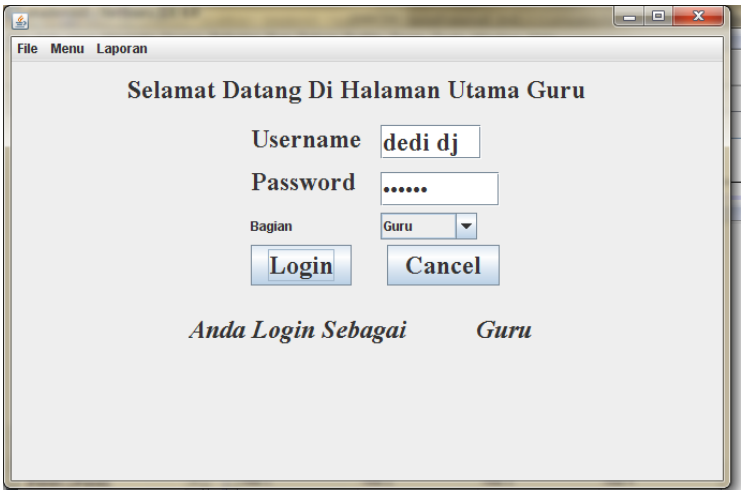
13. Tampilan Utama Home Guru

Setelah Guru berhasil login, maka akan menampilkan halaman menu utama guru seperti dibawah ini.

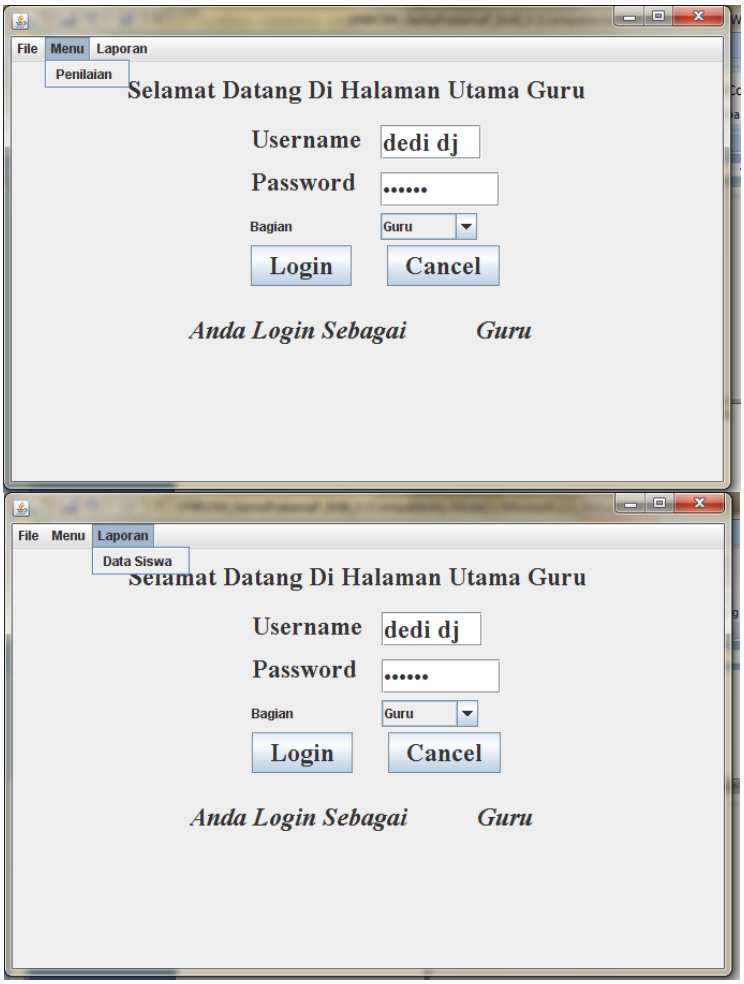


Gambar 5.15 Halaman Menu Utama Guru

Pada halaman utama guru, guru diharuskan login dan memilih bagian karena disini ada 2 pilihan yaitu menjadi guru dan wali kelas. Terlebih dahulu untuk dapat mengakses kelas mana dengan mata pelajaran sesuai dengan mata pelajaran yang diajarkannya. Tampilan Login untuk guru-guru sebagai berikut.



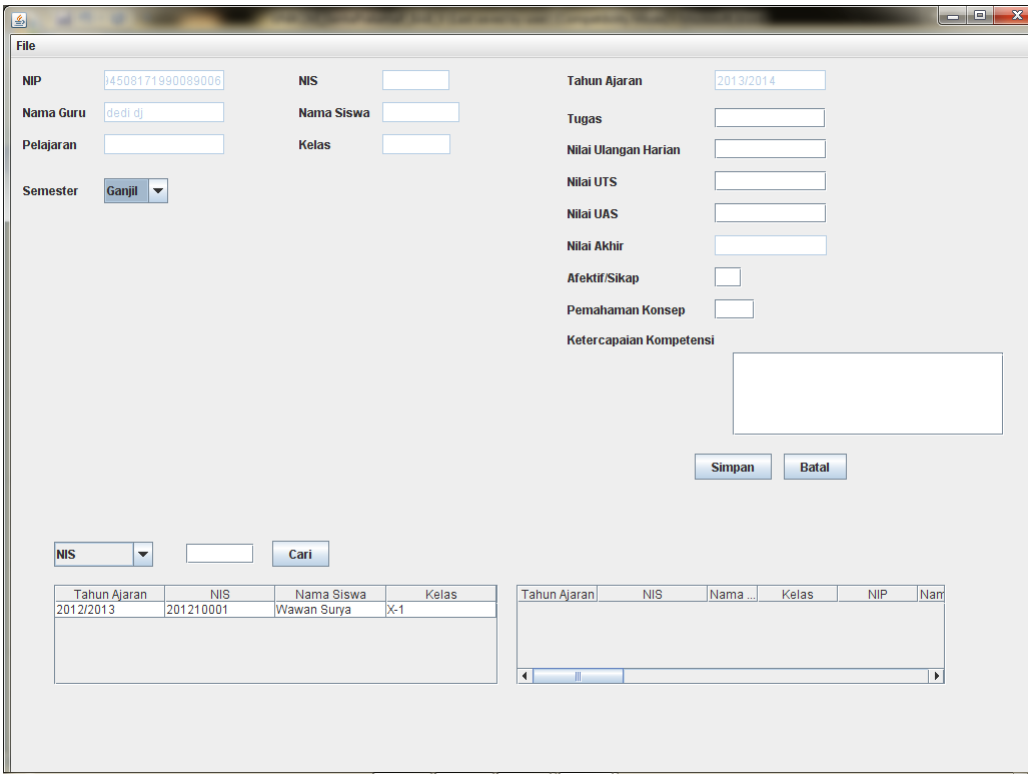
Gambar 5.16 Halaman Menu Pilihan Guru



Gambar 5.17 Tampilan Menu Bagian Guru

14. Tampilan Penilaian

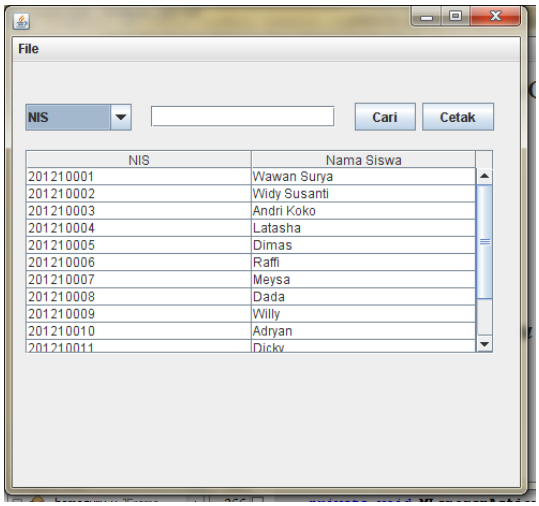
Pada Gambar 5.18 untuk memasukan nilai, guru hanya perlu mengklik 2 kali pada nama siswa yang akan diberikan nilainya, setelah itu masukan nilai pada field nilai yang tersedia di sebelah kanan. Setelah itu “Enter” dan “Simpan”.



Gambar 5.18 tampilan Penilaian

15. Halaman Data Siswa

Ditampilan data siswa ini, dapat mencari dan mencetak data siswa. Terlebih dahulu siswa dicari melalui NIS atau Nama, lalu klik “Cari” maka akan muncul NIS dan Nama Siswa.



Gambar 5.19 Tampilan Halaman Data Siswa

16. Tampilan Wali Kelas

Tampilan wali kelas pada gambar 5.20. wali kelas hanya dapat melakukan pilihan menu untuk mencetak rapot siswa.



Gambar 5.20 Tampilan Wali Kelas

5.2 Pengujian

Pengujian merupakan bagian yang penting dalam siklus pengembangan perangkat lunak. Pengujian selain untuk mengetahui kualitas dan juga mengetahui kelemahan dari perangkat lunak. Pengujian terhadap program itu sendiri yang bertujuan agar program dapat berjalan dengan baik sesuai dengan kebutuhan pengguna dan tanpa mengalami gangguan atau *error*, dan memungkinkan untuk dilakukannya pengembangan sistem lebih lanjut. Pengujian perangkat lunak ini penulis memilih menggunakan metode pengujian *black box*. Pengujian *black box* ini tidak perlu tahu apa yang sesungguhnya terjadi dalam sistem atau perangkat lunak, yang diuji adalah masukan serta keluarannya. Berikut ini adalah rencana pengujian input/output Sistem Informasi Akademik.

5.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem. Pengujian Sistem Informasi Akademik berikut menggunakan data uji berupa masukan dari Bag. Tata Usaha, Wak. Kurikulum, dan Guru. Rencana pengujian selengkapnya terlihat pada tabel dibawah ini :

Tabel 5.1 Rencana Pengujian

Kelas Uji	Detail Pengujian	Jenis Pengujian
1. *Login* Tata Usaha, Wak. Kurikulum, Guru	Verifikasi data login masing-masing pengguna dengan memasukkan nama pengguna dan kata sandi oleh pelanggan.	*Black Box*
2. Pengujian pengisian data siswa pada form	Proses pengisian data siswa pada form	*Black Box*

pendaftaran ulang siswa baru.	pendaftaran ulang siswa baru sekaligus proses simpan.	
3. Pengujian pengisian pembagian kelas siswa	Proses pengisian pembagian kelas siswa sekaligus proses simpan.	*Black Box*
4. Pengujian pengisian pembuatan jadwal pelajaran	Proses pengisian pembuatan jadwal pelajaran sekaligus proses simpan	*Black Box*
5. Pengujian pengisian data nilai siswa	Proses pengisian data nilai siswa sekaligus proses simpan.	*Black Box*

5.2.2 Kasus dan Hasil Pengujian

Kasus dan hasil pengujian dibuat untuk mengetahui apakah Sistem Informasi Akademik pada SMA Negeri 8 Bandung dapat berjalan dengan baik atau tidak.

1. Kelas Uji Login Admin

Berikut ini adalah tabel pengujian *login* pengguna Tata Usaha, Waka. Bid. Kurikulum dan Guru sebagai pengecekkan pengguna yang telah terdaftar:

Tabel 5.2 Kelas Uji Login

Kasus dan Hasil Uji (Data Normal)			
Data Masuk	Yang Diharapkan	Pengamatan	Kesimpulan
Username : admin *Password :* admin	Dapat mengakses halaman sesuai login *user*	Dapat mengakses halaman sesuai login *user*	[ ] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			

Data Masuk	Yang Diharapkan	Pengamatan	Kesimpulan
Username : salah *Password :* salah	Tidak dapat *Login* dan Menampilkan pesan kesalahan	*User* tidak dapat *login* jika ada kesalahan pada *Username* dan atau *Password*	[ ] Diterima [ ] Ditolak

2. Kelas Uji Pengisian Data Siswa Baru

Berikut ini adalah tabel pengujian input data siswa baru:

Tabel 5.3 Kelas Uji Pengisian Data Siswa Baru

Kasus dan Hasil Uji (Data Normal)			
Data Masuk	Yang Diharapkan	Pengamatan	Kesimpulan
Isi lengkap data siswa	Data sesuai dan dapat tersimpan	Pengisian sesuai dengan yang diharapkan	[ ] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masuk	Yang Diharapkan	Pengamatan	Kesimpulan
Data siswa tidak lengkap	Penyimpanan gagal dan memunculkan pesan kesalahan	Petugas harus memasukan kembali data siswa dengan lengkap	[ ] Diterima [ ] Ditolak

3. Kelas Uji Pembagian Kelas Siswa

Berikut ini adalah tabel pengujian pembagian kelas siswa:

Tabel 5.4 Kelas Uji Pembagian Kelas Siswa

Kasus dan Hasil Uji (Data Normal)			
Data Masuk	Yang Diharapkan	Pengamatan	Kesimpulan
Data siswa yang sudah daftar ulang	Dapat menampilkan data kelas baru siswa	Pengisian sesuai dengan yang diharapkan	[ ] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masuk	Yang Diharapkan	Pengamatan	Kesimpulan
Data siswa yang sudah daftar ulang	Tidak dapat menampilkan data kelas baru siswa dan menampilkan pesan kesalahan	Petugas harus memasukan kembali data siswa yang sesuai	[ ] Diterima [ ] Ditolak

4. Kelas Uji Pembuatan Jadwal

Berikut ini adalah tabel pengujian input jadwal pelajaran:

Tabel 5.5 Kelas Uji Pembuatan Jadwal

Kasus dan Hasil Uji (Data Normal)			
Data Masuk	Yang Diharapkan	Pengamatan	Kesimpulan
Data siswa yang sudah daftar ulang	Dapat menampilkan data kelas baru siswa	Pengisian sesuai dengan yang diharapkan	[ ] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masuk	Yang Diharapkan	Pengamatan	Kesimpulan
Data siswa yang sudah daftar ulang	Tidak dapat menampilkan data kelas baru siswa dan menampilkan pesan kesalahan	Petugas harus memasukan kembali data siswa yang sesuai	[ ] Diterima [ ] Ditolak

5. Kelas Uji Penilaian Siswa

Berikut ini adalah tabel pengujian penilaian siswa:

Tabel 5.6 Kelas Uji Penilaian Siswa

Kasus dan Hasil Uji (Data Normal)			
Data Masuk	Yang Diharapkan	Pengamatan	Kesimpulan
Data nilai siswa	Dapat memproses nilai siswa dan dapat tersimpan	Pengisian sesuai dengan yang diharapkan	[ ] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masuk	Yang Diharapkan	Pengamatan	Kesimpulan
Data nilai siswa	Tidak dapat memproses nilai siswa.	Petugas harus memasukan kembali data nilai	[ ] Diterima [ ] Ditolak

5.2.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus *sample* uji yang telah dilakukan memberikan kesimpulan bahwa perangkat lunak ini dapat digunakan dengan baik, namun pengujian tersebut dapat dikatakan belum sempurna,

karena hanya dilakukan pada satu sisi pengujian. Dari semua yang telah dilakukan dalam pengujian ini diharapkan dapat mewakili pengujian fungsi yang lain dalam Sistem Informasi Akademik ini.