BAB V

IMPLEMENTASI DAN PENGUJIAN

5.1. Implementasi

Implementasi adalah proses penerapan rancangan program yang telah dibuat pada bab sebelumnya atau aplikasi dalam melaksanakan sistem informasi pemrograman yang telah dibuat, hasil dari tahapan implementasi ini adalah suatu sistem pengolahan data yang sudah dapat berjalan dengan baik. Dengan demikian dapat diketahui apakah perangkat lunak ini dapat menghasilkan aplikasi sistem informasi akademik yang sesuai dengan tujuan yang diharapkan.

5.1.1. Batasan Implementasi

Dalam mengimplementasi perangkat lunak pemantauan dan pengendalian aplikasi ini ada beberapa hal yang menjadi batasan implementasi, yaitu:

1. Tidak semua kebutuhan sistem dikerjakan seperti sistem keuangan, tetapi yang dikerjakan hanya subsitem-subsistem yang berkaitan dengan kegiatan belajar mengajar
2. Database yang digunakan dalam pengimplementasian ini adalah Mysql. Pengaksesan database dilakukan dengan mengaktifkan Control wamp.
3. Interface untuk program aplikasi ini menggunakan bahasa Indonesia dan bahasa inggris seperti : Home, Login, Admin, Info, Regristrasi, Open An

Account, Contact Us, Delete, Update, Tambah, dan lain lain agar mudah dimengerti oleh User.

4. Administrator melakukan kelola data master keseluruhan yang berhubungan dengan akademik.

5.1.2. Implementasi Perangkat Lunak

Untuk pengembangan perangkat lunak ini digunakan PHP dan Database Mysql. PHP dipilih sebagai perangkat lunak untuk program aplikasi yang dibuat untuk merancang web secara dinamis, sedangkan Mysql digunakan sebagai perangkat lunak pengembang dalam pembuatan database.

5.1.3. Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus terpenuhi antara lain:

a) Menggunakan minimal processor Intel Pentium II atau yang sekelasnya.

b) Menggunakan memori (RAM) minimal 512 MB.

c) VGA card terpasang rekomendasi minimal 32 MB

d) Printer untuk mencetak laporan-laporan.

e) Harddisk terpasang 320 GB

f) Mouse dan keyboard sebagai peralatan antarmuka.

5.1.4. Implementasi Basis Data (Sintaks SQL)

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

```
-- Table structure for table `agama`

CREATE TABLE IF NOT EXISTS `agama` (
  `kode_agama` tinyint(4) NOT NULL AUTO_INCREMENT,
  `nama_agama` varchar(15) NOT NULL,
  PRIMARY KEY (`kode_agama`),
  KEY `kode_agama` (`kode_agama`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;

--
-- Table structure for table `calon_siswa`

CREATE TABLE IF NOT EXISTS `calon_siswa` (
  `no_registrasi` varchar(15) NOT NULL,
  `nama_lengkap_calon_siswa` varchar(60) NOT NULL,
  `telepon` varchar(16) NOT NULL,
  `tempat_lahir_calon_siswa` varchar(25) NOT NULL,
  `tanggal_lahir_calon_siswa` date NOT NULL,
  `sekolah_asal` varchar(40) NOT NULL,
  `nilai` tinyint(4) NOT NULL DEFAULT '0',
  `nomer_ijazah` varchar(50) NOT NULL,
  `kode_agama` tinyint(4) NOT NULL,
  `alamat_calon_siswa` varchar(100) NOT NULL,
  `nama_orangtua_calon_siswa` varchar(60) NOT NULL,
  `pekerjaan_orang_tua` varchar(35) NOT NULL,
  `pas_photo` varchar(25) NOT NULL,
  `email` varchar(50) NOT NULL,
  `ijazah` varchar(125) NOT NULL,
  PRIMARY KEY (`no_registrasi`),
  KEY `kode_agama` (`kode_agama`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Table structure for table `guru`

CREATE TABLE IF NOT EXISTS `guru` (
  `nip` varchar(25) NOT NULL,
  `nama_guru` varchar(60) NOT NULL,
  `gender` enum('M','F') NOT NULL,
  `kode_agama` tinyint(4) NOT NULL,
  `tempat_lahir_guru` varchar(25) NOT NULL,
  `tanggal_lahir_guru` date NOT NULL,
  `kode_pangkat` tinyint(4) NOT NULL,
  `kode_tingkat` tinyint(4) NOT NULL,
```

```
  `jurusan` varchar(30) NOT NULL,
  `jumlah_jam` int(3) NOT NULL,
  `tugas_tambahan` varchar(35) NOT NULL DEFAULT '-',
  `alamat_rumah` varchar(100) NOT NULL,
  `telepon` varchar(12) NOT NULL,
  `keterangan` varchar(50) DEFAULT NULL,
  `kode_mata_pelajaran` varchar(15) NOT NULL,
  `pas_photo` varchar(25) NOT NULL,
  PRIMARY KEY (`nip`),
  KEY `Kode_Mata_Pelajaran` (`kode_mata_pelajaran`),
  KEY `Kode_Mata_Pelajaran_2` (`kode_mata_pelajaran`),
  KEY `kode_agama` (`kode_agama`),
  KEY `kode_pangkat` (`kode_pangkat`),
  KEY `kode_tingkat` (`kode_tingkat`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Table structure for table `jadwal_kbm`

CREATE TABLE IF NOT EXISTS `jadwal_kbm` (
  `kode_kbm` tinyint(4) NOT NULL AUTO_INCREMENT,
  `hari` varchar(8) NOT NULL,
  `jam` time NOT NULL,
  `tahun_pelajaran` varchar(10) NOT NULL,
  `nip` varchar(25) NOT NULL,
  `kode_kelas` varchar(15) NOT NULL,
  PRIMARY KEY (`kode_kbm`),
  KEY `NIP` (`nip`,`kode_kelas`),
  KEY `NIP_2` (`nip`),
  KEY `Kode_Kelas` (`kode_kelas`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;

-- Table structure for table `kelas`

CREATE TABLE IF NOT EXISTS `kelas` (
  `kode_kelas` varchar(15) NOT NULL,
  `nama_kelas` varchar(12) NOT NULL,
  `keterangan` varchar(50) NOT NULL,
  PRIMARY KEY (`kode_kelas`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Table structure for table `kepangkatan`

CREATE TABLE IF NOT EXISTS `kepangkatan` (
  `kode_pangkat` tinyint(4) NOT NULL AUTO_INCREMENT,
  `pangkat` varchar(20) NOT NULL,
  `golongan` varchar(5) NOT NULL,
  `ruang` char(5) NOT NULL,
  PRIMARY KEY (`kode_pangkat`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=18 ;

-- Table structure for table `login`

CREATE TABLE IF NOT EXISTS `login` (
```

```
  `username` varchar(45) NOT NULL,
  `password` varchar(60) NOT NULL,
  `nama_login` varchar(25) NOT NULL,
  `hak_akses` varchar(8) NOT NULL,
  PRIMARY KEY (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Table structure for table `mata_pelajaran`

CREATE TABLE IF NOT EXISTS `mata_pelajaran` (
  `kode_mata_pelajaran` varchar(15) NOT NULL,
  `nama_mata_pelajaran` varchar(25) NOT NULL,
  `durasi_jam` int(3) NOT NULL,
  PRIMARY KEY (`kode_mata_pelajaran`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Table structure for table `nilai_siswa`

CREATE TABLE IF NOT EXISTS `nilai_siswa` (
  `nilai_mid_semester` float NOT NULL,
  `nilai_ujian_akhir_semester` float NOT NULL,
  `nilai_akhir` float NOT NULL,
  `semester` varchar(6) NOT NULL,
  `nis` varchar(12) NOT NULL,
  `nip` varchar(25) NOT NULL,
  `kode_mata_pelajaran` varchar(15) NOT NULL,
  KEY `NIS` (`nis`,`nip`,`kode_mata_pelajaran`),
  KEY `NIP` (`nip`,`kode_mata_pelajaran`),
  KEY `Kode_Mata_Pelajaran` (`kode_mata_pelajaran`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Table structure for table `siswa`

CREATE TABLE IF NOT EXISTS `siswa` (
  `nis` varchar(12) NOT NULL,
  `nama_siswa` varchar(60) NOT NULL,
  `tempat_lahir_siswa` varchar(25) NOT NULL,
  `tanggal_lahir_siswa` date NOT NULL,
  `gender_siswa` varchar(1) NOT NULL,
  `nama_orangtua_siswa` varchar(60) NOT NULL,
  `anak_ke` int(2) NOT NULL,
  `jml_saudara` int(2) NOT NULL,
  `alamat_1` varchar(100) NOT NULL,
  `alamat_2` varchar(100) NOT NULL,
  `asal_smp` varchar(40) NOT NULL,
  `no_ijazah` varchar(35) NOT NULL,
  `no_skhun` varchar(35) NOT NULL,
  `no_peserta_un` varchar(35) NOT NULL,
  `kode_agama` tinyint(4) NOT NULL,
  `tingkat` varchar(3) NOT NULL,
  `pas_photo` varchar(25) NOT NULL,
  `username` varchar(45) NOT NULL,
  `kode_kelas` varchar(15) NOT NULL,
```

```
  `no_registrasi` varchar(15) NOT NULL,
  PRIMARY KEY (`nis`),
  KEY `Username` (`username`,`kode_kelas`,`no_registrasi`),
  KEY `Kode_Kelas` (`kode_kelas`,`no_registrasi`),
  KEY `No_Registrasi` (`no_registrasi`),
  KEY `kode_agama` (`kode_agama`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Table structure for table `tingkat_pendidikan`

CREATE TABLE IF NOT EXISTS `tingkat_pendidikan` (
  `kode_tingkat` tinyint(4) NOT NULL AUTO_INCREMENT,
  `tingkat_pendidikan` varchar(15) NOT NULL,
  PRIMARY KEY (`kode_tingkat`),
  KEY `kode_tingkat` (`kode_tingkat`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

5.1.5. Implementasi Antar Muka

Berikut ini merupakan implementasi dari setiap halaman aplikasi sistem informasi akademik sma negeri 2 bandung berbasis web dimana setiap halaman dibedakan sesuai dengan hak akses masing-masing yaitu user dan admin.

5.1.5.1. Implementasi Halaman User

Berikut adalah tampilan antar muka untuk user, dimana user harus melakukan login terlebih dahulu untuk dapat mengakses halaman user (siswa atau guru).

Tabel 5.1 Implementasi Halaman User

Sub Menu	Deskripsi	Nama File
Login Siswa	File program untuk halaman login siswa	login.php?vw=lg
Jadwal Pelajaran	File program untuk	profile.php?vw=jdw

	halaman info siswa dan jadwal pelajaran siswa	
Absen	File program untuk halaman info absensi siswa	profile.php?vw=abs
Nilai	File program untuk halaman info nilai siswa	profile.php?vw=nil
Logout	File program untuk keluar (logout)	index.php?vw=hm
Home	File program untuk Halaman Utama Front End	index.php
Profil	File program untuk halaman profil sekolah	careers.php?vw=pr
Artikel	File program untuk halaman artikel	blog1.php?vw=bg
Info	File program untuk halaman informasi artikel terkait dan sekolah	blog-single.php?vw=in
Penerimaan Siswa Baru /	File program untuk	register.php?vw=df-cs

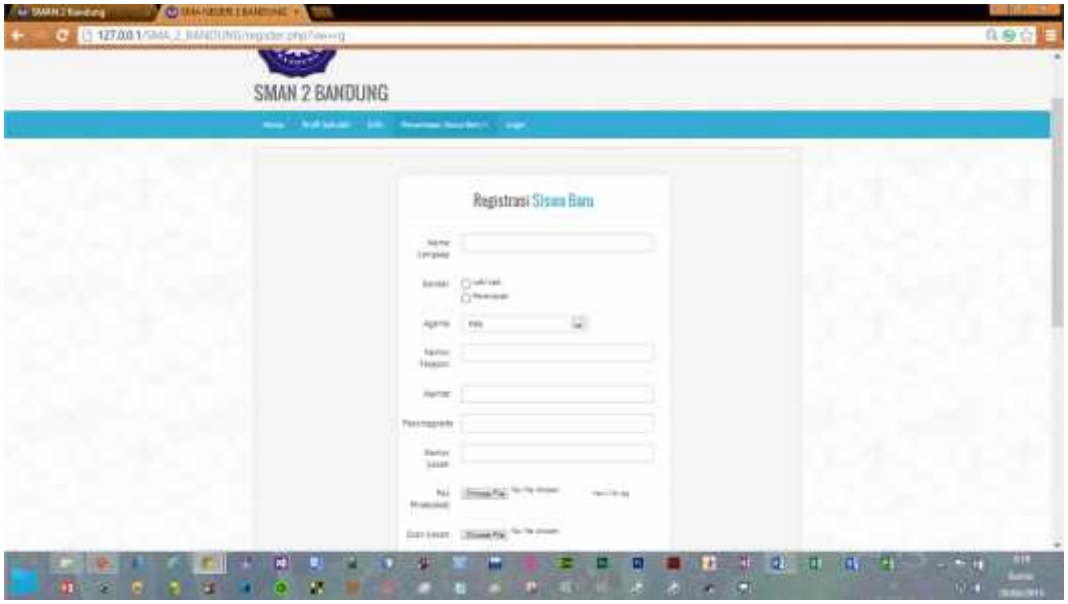
Daftar Calon Siswa	halaman info listing calon siswa	
Penerimaan Siswa Baru / Registrasi Calon Siswa	File program untuk halaman registrasi calon siswa	register.php?vw=rg

Berikut implementasi halaman awal user :



Gambar 5.1. Implementasi Halaman Awal User

Untuk registrasi calon siswa, dapat memilih Menu **Pendaftaran Siswa Baru -> Registrasi**, maka akan tampil halaman sebagai berikut :



Gambar 5.2. Implementasi Halaman Registrasi Calon Siswa

Sedangkan jika ingin melihat list calon siswa SMAN 2 BANDUNG dapat dilihat dengan memilih menu **Penerimaan Siswa Baru -> Daftar Calon Siswa**, maka akan muncul tampilan seperti berikut :



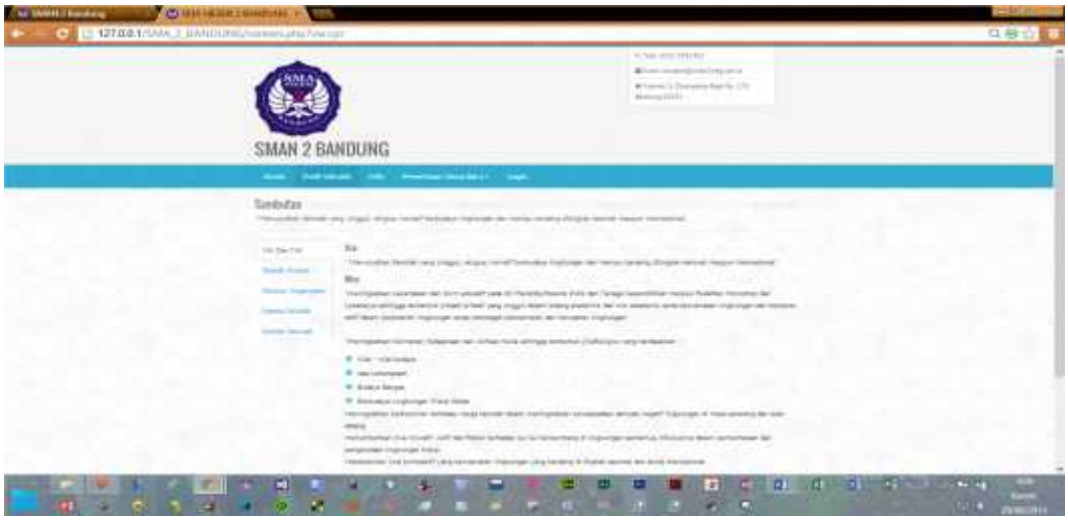
Gambar 5.3. Implementasi Halaman Daftar Calon Siswa

Untuk informasi berita seputar SMAN 2 BANDUNG dapat dilihat dengan memilih menu **Info**, maka akan muncul halaman seperti berikut :



Gambar 5.4. Implementasi Halaman Info Berita

Sedangkan profil SMAN 2 BANDUNG dapat dilihat dengan memilih menu **Profil Sekolah**, maka akan muncul tampilan seperti berikut :



Gambar 5.5. Implementasi Halaman Profil

5.1.5.2. Implementasi Halaman Admin

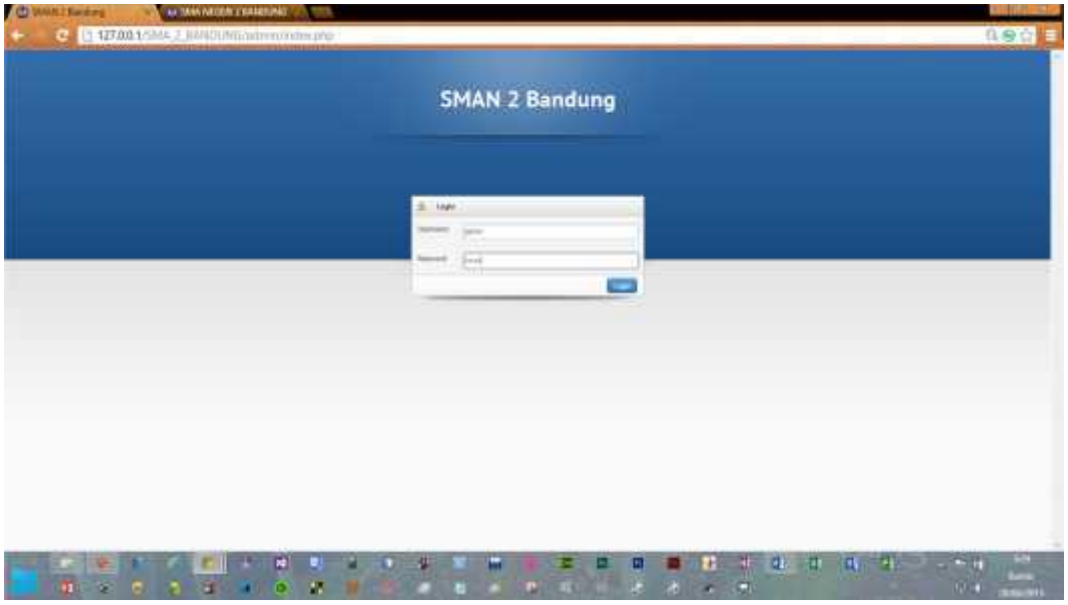
Berikut adalah tampilan antar muka untuk admin, dimana admin harus melakukan login terlebih dahulu untuk dapat mengakses halaman admin.

Tabel 5.2 Implementasi Halaman Admin

Sub Menu	Deskripsi	Nama File
Login Admin	File program untuk halaman login administrator	admin/index.php
Home (Dasboard)	File program untuk Halaman Utama Back End (admin)	admin/home.php
Data Master / Mata Pelajaran	File program untuk halaman data master mata pelajaran	admin/mata_pelajaran.php?act=act&sia=view&vw=mp
Data Master / Guru	File program untuk halaman data master guru	admin/guru.php?act=act&sia=view&vw=gr
Data Master / Kelas	File program untuk halaman data master kelas	admin/kelas.php?act=act&sia=view&vw=kl
Data Master / Siswa	File program untuk halaman data master siswa	admin/siswa.php?act=act&sia=view&vw=sw
Jadwal Pelajaran /	File program untuk halaman	admin/jadwal_kbm.php

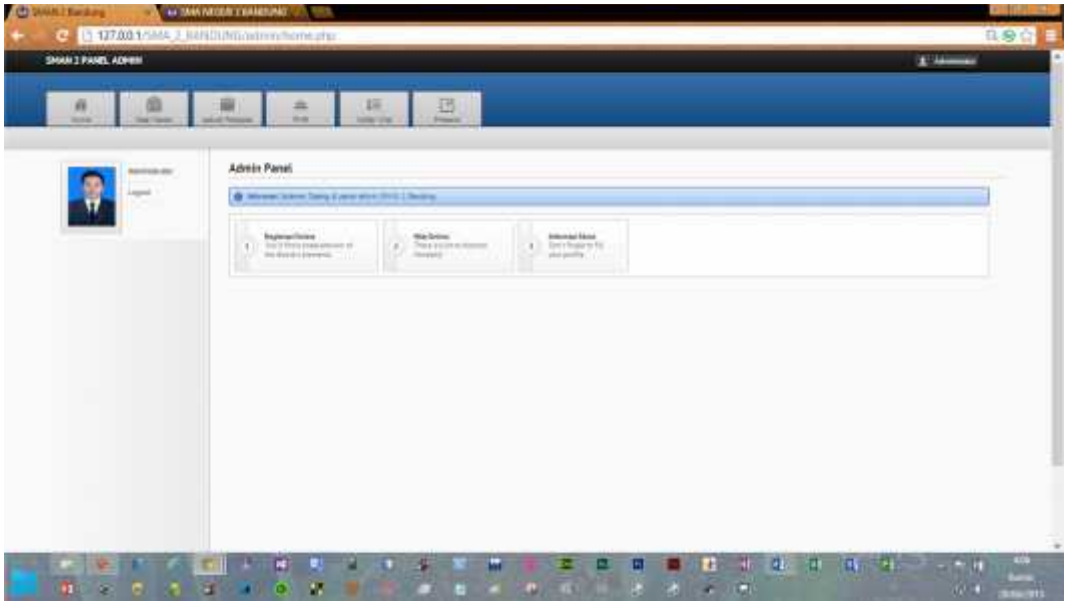
Tambah Jadwal	add jadwal KBM	?act=act&sia=add&vw=jd
Jadwal Pelajaran / Jadwal KBM	File program untuk halaman info jadwal KBM	admin/jadwal_kbm.php?act=act&sia=view&vw=jd
PMB / Daftar Siswa Baru	File program untuk halaman listing calon siswa baru	admin/pmb.php?act=act&sia=view&vw=pmb
PMB / Jadwal KBM	File program untuk halaman info jadwal KBM bagi calon siswa	admin/pmb.php?act=act&sia=add&vw=pmb
Daftar Nilai	File program untuk halaman manipulasi nilai mid semester siswa	admin/nilai.php?act=act&sia=view&vw=z
Tambah Nilai	File program untuk halaman menambahkan data nilai siswa	admin/nilai.php?act=act&sia=add&vw=z
Logout	File program untuk keluar (logout)	admin/index.php

Pada implementasi halaman admin, pertama kali akan muncul halaman permintaan untuk login admin :



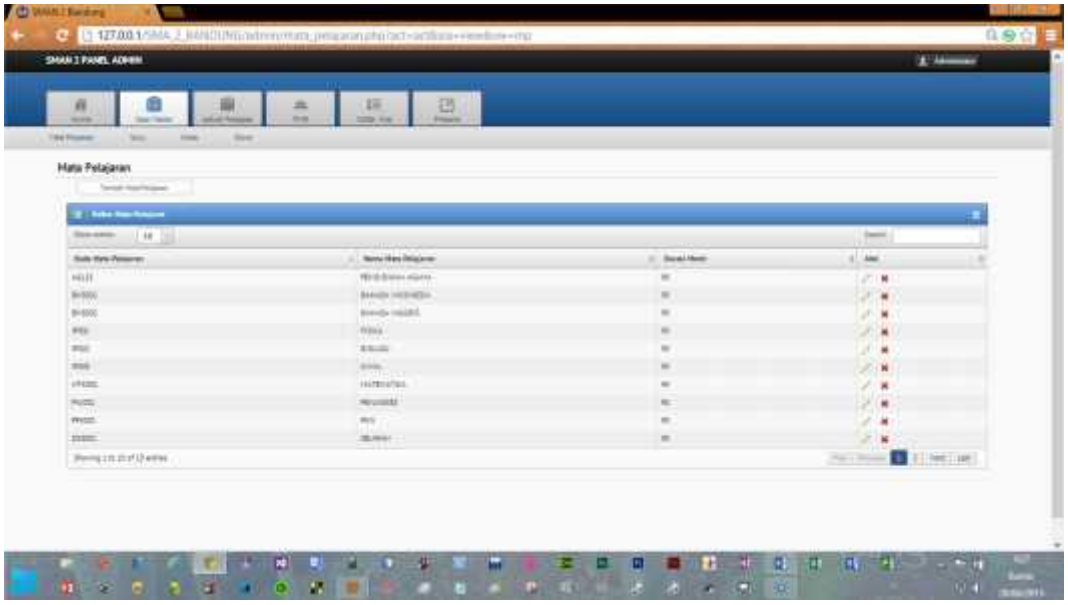
Gambar 5.6. Implementasi Halaman Login Admin

Setelah itu masukan username : **admin** dan password : **admin** maka akan masuk ke halaman Home (Dasboard) Administrator :

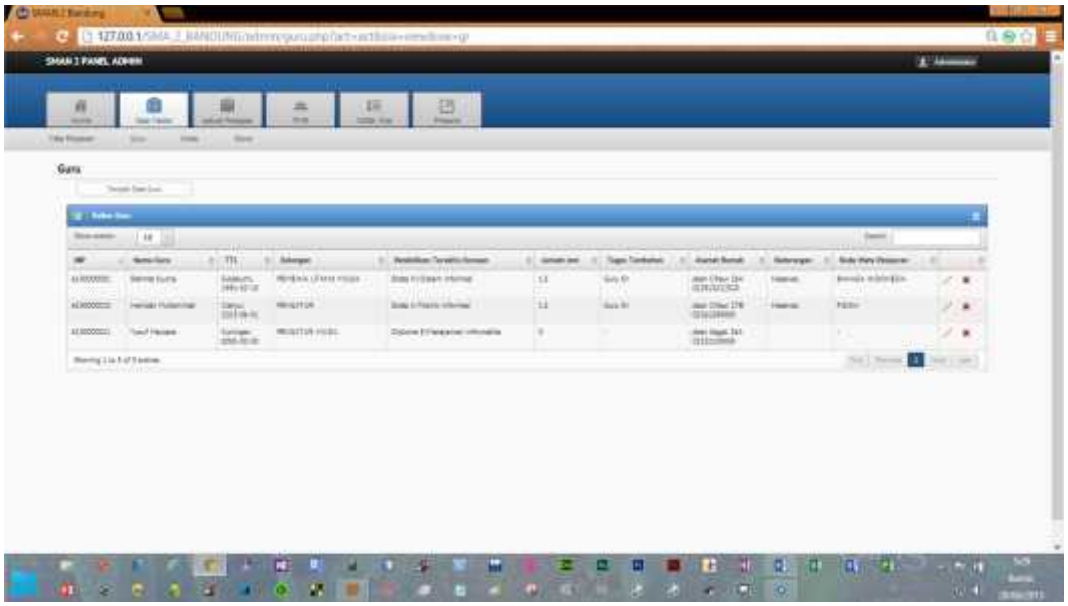


Gambar 5.7. Implementasi Halaman Dasboard Administrator

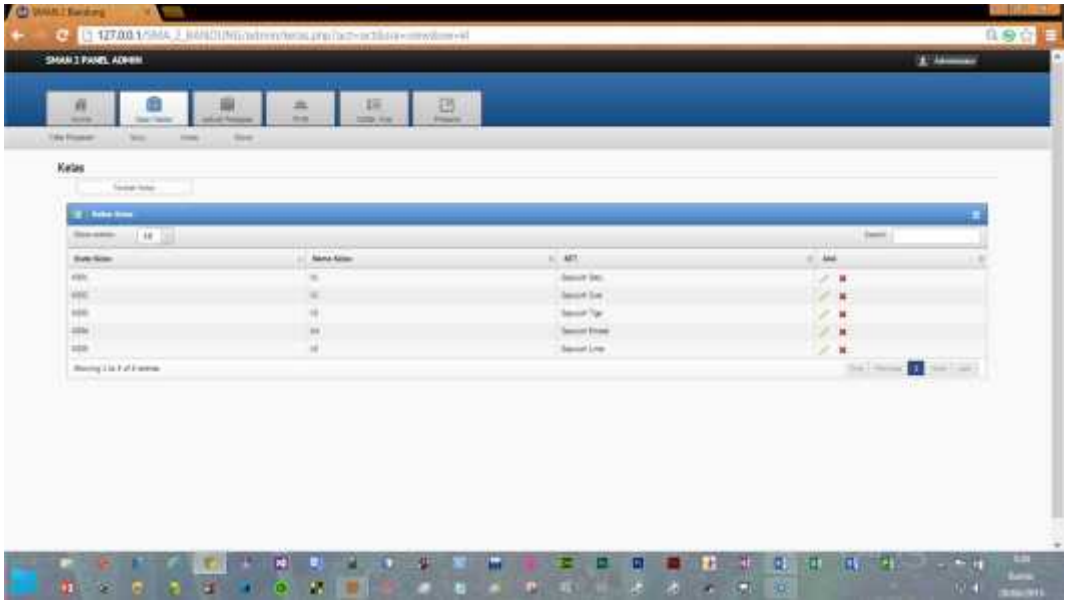
Berikut ini halaman-halaman untuk manipulasi data master :



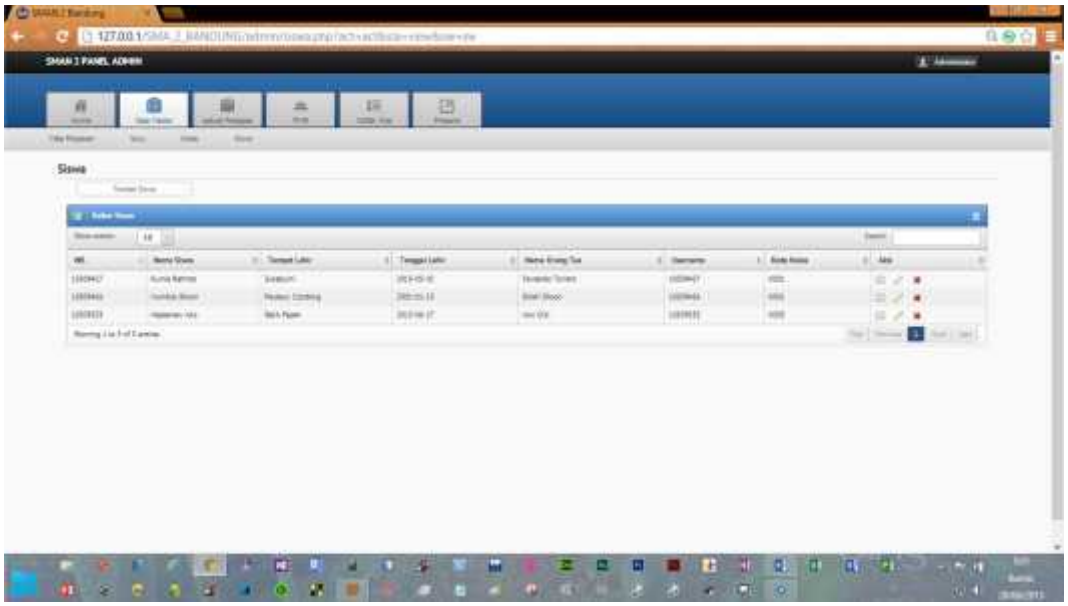
Gambar 5.8. Implementasi Halaman Data Master Mata Pelajaran



Gambar 5.9. Implementasi Halaman Data Master Guru

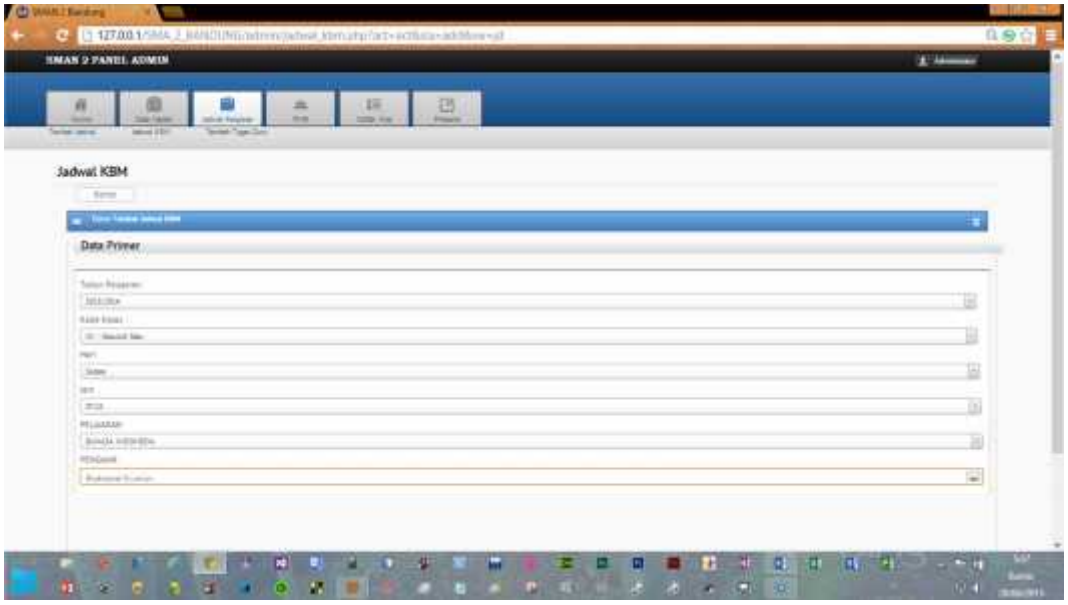


Gambar 5.10. Implementasi Halaman Data Master Kelas



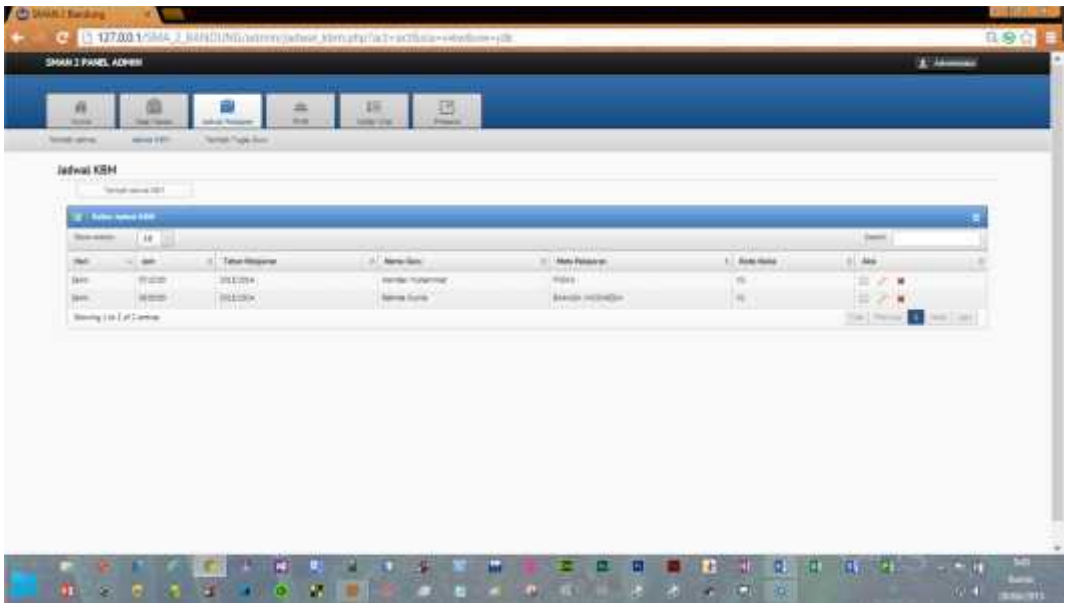
Gambar 5.11. Implementasi Halaman Data Master Siswa

Adapun implementasi penjadwalan kegiatan belajar mengajar pada halaman jadwal KBM dengan memilih menu **Jadwal Pelajaran -> Tambah Jadwal**, maka akan muncul halaman seperti berikut :



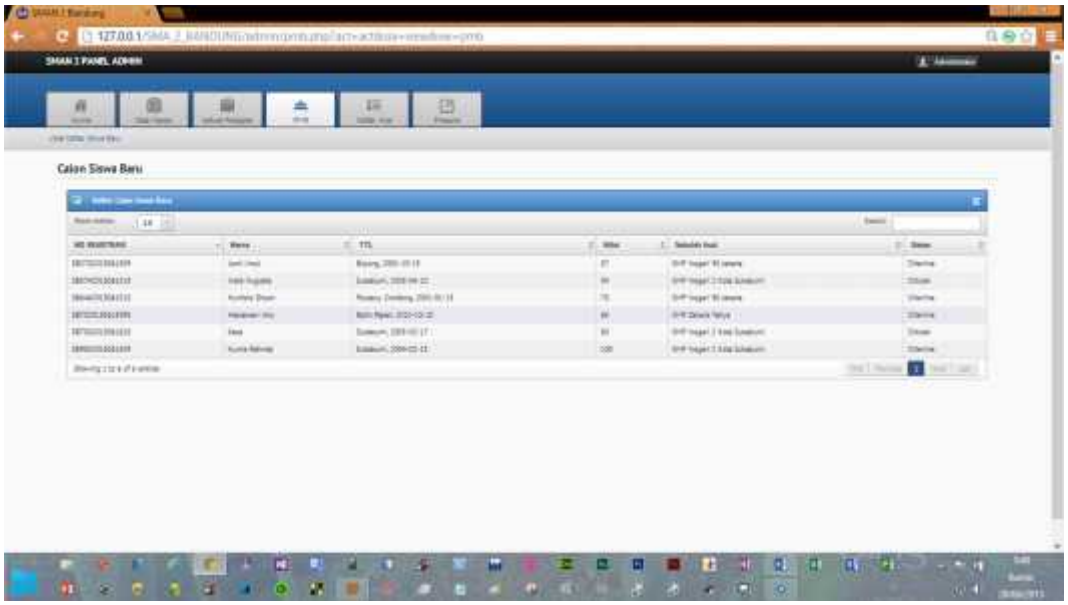
Gambar 5.12. Implementasi Halaman Penjadwalan

Untuk melihat Jadwal KBM terdapat pada halaman Jadwal KBM :



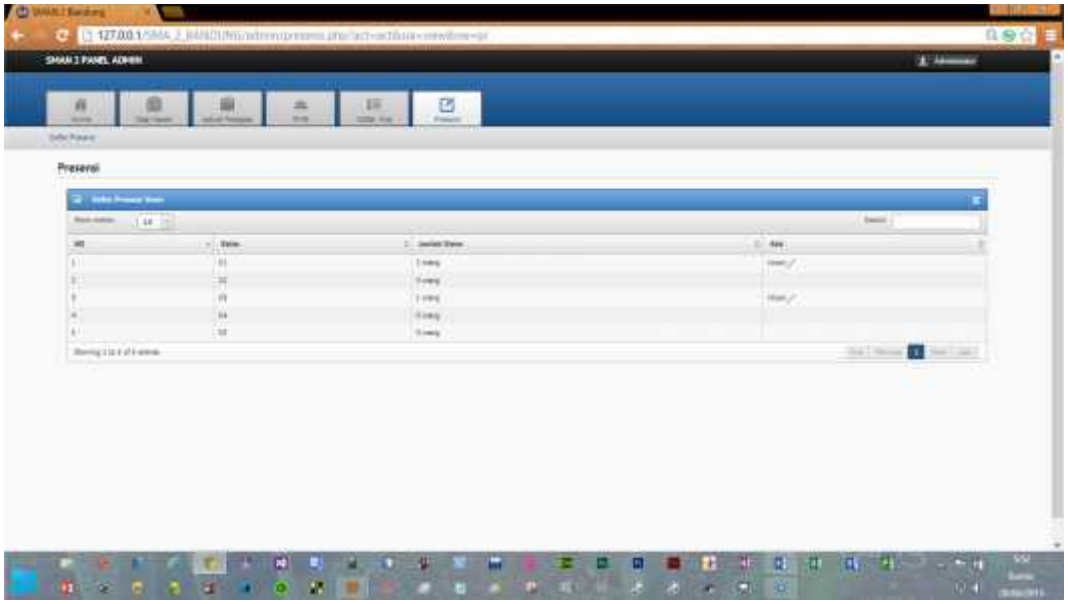
Gambar 5.13. Implementasi Halaman Jadwal KBM

Adapun pengelolaan calon siswa baru yang telah melakukan registrasi online dapat dilihat dengan memilih menu **PMB -> List Daftar Siswa Baru** sebagai berikut :



Gambar 5.14. Implementasi Daftar Calon Siswa Baru

Kemudian untuk data presensi setiap kelas bisa dilihat dengan memilih menu **Presensi -> Daftar Presensi** :



Gambar 5.15. Implementasi Halaman Presensi

5.1.6. Implementasi Instalasi Program

Untuk menguji program aplikasi ini terlebih dahulu ikuti langkah-langkah berikut :

1. Aktifkan WAMP. start->all program->WampServer->startWampServer

2. Simpan Folder Aplikasi didalam Lokasi WAMP/www/SMA_2_BANDUNG.

3. Buka browser, misalnya Mozilla Firefox Portable lalu ketikan alamat url pada bagian address : http://localhost/SMA%202%20BANDUNG, maka akan tampil situs web sistem informasi akademik SMA 2 BANDUNG.

5.2. Pengujian

Pada pengujian ini, metode yang di gunakan adalah metode pengujian Black Box. Pengujian black box adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar. Pengujian black box merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak.

Dalam pengujian black box ini tidak perlu tahu apa yang sesungguhnya terjadi dalam sistem atau perangkat lunak, yang diuji adalah masukan serta keluarannya. Dengan berbagai masukan yang diberikan, apakah sistem atau perangkat lunak memberikan keluaran seperti yang kita harapkan atau tidak.

5.2.1. Rencana Pengujian

Dalam melakukan rencana pengujian pada Sistem Informasi Akademik SMA Negeri 2 Bandung perancangan website ini menggunakan metode black box. Pengujian black box berfokus pada persyaratan fungsional perangkat lunak yang dibuat.

Tabel 5.3 Rencana Pengujian

Kelas Uji	Butir Uji	Tingkat Pengujian	Jenis Pengujian
Login	Pengecekan Login	Sistem	Black Box
Data	Pengisian data siswa	Modul	Black Box
	Pencarian data siswa	Modul	Black Box

5.2.2. Kasus dan Hasil Pengujian

Berdasarkan hasil pengujian dari kasus uji sample di atas, dapat ditarik kesimpulan bahwa perangkat lunak bekerja sesuai dengan harapan dan berjalan dengan sebagaimana mestinya.

1. Pengecekan login admin

Berikut ini kasus dan hasil pengujian login admin:

Tabel 5.4 Pengecekan Login

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Username : admin Password : Admin	Tercantum pada textbox username dan textbox pasword	Dapat mengisi login sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Klik tombol LOGIN	Login berhasil dengan hak aksesnya dan menampilkan halaman utama administrator	Tombol dapat berfungsi sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Username : false Password : False	Tercantum pada textbox username dan textbox password	User tidak dapat login	[X] Diterima [ ] Ditolak
Klik tombol LOGIN	Tidak dapat masuk ke halaman utama administrator dan menampilkan pesan “Ada kesalahan pada username dan password !”	Tombol dapat berfungsi sesuai yang diharapkan	[X] Diterima [ ] Ditolak

2. Pengisian data siswa

Berikut ini kasus dan hasil uji pengujian data siswa :

Tabel 5.5 Pengisian Data Siswa

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol TAMBAH SISWA	Selectbox no_registrasi aktif	Dapat menampilkan data list calon siswa pada selectbox	[X] Diterima [ ] Ditolak
Memilih calon siswa yang registrasi melalui selectbox	Selectbox no_registrasi terpilih	Dapat melakukan fungsi dengan benar	[X] Diterima [ ] Ditolak
Klik tombol CEK	Menampilkan data calon siswa dan field NIS terisi otomatis	Dapat menampilkan data calon siswa dan NIS terisi	[X] Diterima [ ] Ditolak
Mengisi field pas photo, No_SKHU, no_UN	Tercantum pada field pas photo, No_SKHU, no_UN	Dapat melengkapi data siswa sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Klik tombol TAMBAH	Menampilkan data siswa baru dengan NIS baru	Dapat menampilkan data siswa baru dengan NIS baru	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data-data siswa diisi secara tidak lengkap dan ada yang	Penyimpanan tidak dapat dilakukan	Tidak dapat melakukan penyimpanan data	[X] Diterima [ ] Ditolak

kosong			

3. Pencarian data siswa

Berikut ini kasus dan hasil uji pencarian data siswa :

Tabel 5.6 Pencarian Data Siswa

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data diisi sesuai dengan data yang ada	Pencarian data kedalam file data siswa dan muncul hasil pencatian otomatis	Dapat melakukan pencarian data dengan benar dan menampilkan data otomatis	[X] 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 aplikasi sistem informasi akademik SMA Negeri 2 Bandung berbasis *web* ini.