

**APLIKASI SISTEM INFORMASI AKADEMIK SMA  
KATOLIK TARUNA JAYA SAMPIT**

Oleh:

Alexander Setiawan

Liliana

Bryan Andhika Wijaya

PROGRAM STUDI TEKNIK INFORMATIKA



**FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS KRISTEN PETRA  
SURABAYA**

**2013**

**LAPORAN PENELITIAN**  
**NO: 135/Pen/Informatika/I/2013**

**APLIKASI SISTEM INFORMASI AKADEMIK SMA**  
**KATOLIK TARUNA JAYA SAMPIT**

Oleh:  
Alexander Setiawan  
Liliana  
Bryan Andhika Wijaya

PROGRAM STUDI TEKNIK INFORMATIKA



**FAKULTAS TEKNOLOGI INDUSTRI**  
**UNIVERSITAS KRISTEN PETRA**  
**SURABAYA**

**2013**

**LEMBAR IDENTITAS DAN PENGESAHAN**  
**LAPORAN HASIL PENELITIAN**

1. a. Judul Penelitian : **APLIKASI SISTEM INFORMASI  
AKADEMIK SMA KATOLIK  
TARUNA JAYA SAMPIT**
- b. Nomor Penelitian :  
c. Jalur Penelitian : I / ~~II~~ / III / IV
2. Ketua Peneliti
- a. Nama lengkap dan Gelar : Alexander Setiawan, M.T.  
b. Jenis Kelamin : Laki-laki  
c. Pangkat/Golongan/NIP : Pembina / IV-A / 04021  
d. Bidang Ilmu yang diteliti : Sistem Informasi  
e. Jabatan Akademik : Lektor  
f. Fakultas/Program Studi : Fakultas Teknologi Industri / Teknik  
Informatika  
g. Universitas : Universitas Kristen Petra
3. Anggota Tim Peneliti (I)
- a. Nama lengkap dan Gelar : Liliana, M.Eng.  
b. Jenis Kelamin : Perempuan  
c. Pangkat/Golongan/NIP : Pembina / IV-A / 03024  
d. Bidang Ilmu yang diteliti : Sistem Informasi  
e. Jabatan Akademik : Lektor  
f. Fakultas/Program Studi : Fakultas Teknologi Industri / Teknik  
Informatika  
g. Universitas : Universitas Kristen Petra
- Anggota Tim Peneliti (II)
- a. Nama lengkap dan Gelar : Bryan Andhika Wijaya, S.Kom  
b. Jenis Kelamin : Laki-laki  
c. Pangkat/Golongan/NIP : -  
d. Bidang Ilmu yang diteliti : Sistem Informasi  
e. Jabatan Akademik : -  
f. Fakultas/Program Studi : Fakultas Teknologi Industri / Teknik  
Informatika  
g. Universitas : Universitas Kristen Petra
4. Lokasi Penelitian : Surabaya
5. Kerjasama dengan Instansi lain  
Nama Instansi : -  
Alamat : -
6. Tanggal Penelitian : Agustus 2012 s/d Januari 2013
7. Biaya : Rp. 10.000.000,-

Surabaya, 5 Februari 2013

Mengetahui,  
Ketua Program Studi

Ketua Peneliti

Yulia, M.Kom.  
NIP. 99036

Alexander Setiawan, M.T.  
NIP. 04021

Menyetujui,  
Dekan Fakultas Teknologi Industri

Djoni Haryadi Setiabudi, M.Eng.  
NIP. 85009

## ABSTRAK

SMA Katolik Taruna Jaya Sampit, didirikan pada pada tahun 1985, merupakan salah satu SMA swasta yang dikenal kualitasnya dan termasuk favorit di Kota Sampit. Hal ini didasari tingkat kedisiplinan yang tinggi, cara pengajaran yang baik, dan fasilitas yang memadai. Pada sistem yang ada di SMA Katolik Taruna Jaya Sampit sebelumnya, semua data (data siswa, data nilai, data orang tua, data staf pengajar, dan data pendaftaran siswa baru) disimpan dalam bentuk *hard copy*. Penyimpanan ini beresiko akan kehilangan data, karena ruangan penyimpanan data menjadi satu bercampur dengan berbagai arsip data-data yang lain. SMA Katolik Taruna Jaya Sampit membutuhkan sebuah sistem informasi untuk memenuhi kebutuhan peningkatan layanan pendidikan dalam hal kinerja dan kualitas pelayanan.

Perancangan desain sistem menggunakan *Data Flow Diagram* dan *Entity Relationship Diagram*. Aplikasi ini dibuat menggunakan bahasa pemrograman *php* dan menggunakan *Database MySQL*. Proses yang disediakan pada aplikasi ini diantara lain melakukan pengolahan data-data akademik seperti data pendaftaran siswa baru, biodata siswa, data orang tua siswa, data nilai siswa, data mata pelajaran, data jadwal pelajaran, data jadwal ujian, data pelanggaran siswa, dan data staf pengajar. Sistem informasi juga memiliki *tools* yang dapat membantu dalam penyusunan jadwal akademik di sekolah tersebut.

Dari hasil pengujian, program yang telah dibuat dapat memproses data akademik sehari-hari serta menghasilkan laporan yang membantu memantau perkembangan kegiatan akademik sekolah. Laporan yang disediakan berupa laporan nilai, laporan pelanggaran siswa, laporan jadwal, dan laporan penerimaan siswa..

Kata kunci:

*Sistem Informasi, Akademik*

## **ABSTRACT**

Taruna Jaya Catholic Senior High School Sampit, founded in 1985, is a private school that known for its quality and includes the favorite in Sampit City. This is based on a high level of discipline, a good way of teaching, and adequate facilities. In the previous existing system, all data (student data, grade data, student's parent data, teacher data, and new student registration data) are stored in hard copy form. This storage will risk losing data, because data storage is mixed with a variety of another data archive. Taruna Jaya Catholic Senior High School Sampit requires an information system to meet the increasing needs of education in terms of performance and service quality.

The design of the system use Data Flow Diagram and Entity Relationship Diagram. This application is created using the programming language PHP and uses MySQL Database. Processes in this application are doing academic data such as data of new student registration, student data, student's parent data, grade data, subject data, timetable data, exam schedule data, student's fouls data, and teacher data.

From the test results, the program has been created to process the data of daily academic and generate reports that help monitor the progress of academic activities of the school. The report provided grade report, fouls report of the student, timetable reports, and student registration reports.

Keyword:

Information System, Academic

## **KATA PENGANTAR**

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat dan penyertaan-Nya, sehingga penulis dapat menyelesaikan penelitian ini dengan baik.

Penulis berharap semoga penelitian ini dapat memberikan kontribusi bagi perkembangan ilmu pengetahuan dalam bidang sistem informasi.

Akhir kata, penulis sadar bahwa penelitian ini masih jauh dari sempurna dan penulis akan menerima berbagai saran dan kritik yang membangun dari pembaca agar dapat menjadi lebih baik lagi dari sebelumnya. Penulis juga berharap agar pelaksanaan kegiatan penelitian ini dapat membantu semua pihak yang membutuhkannya. Terima kasih.

Surabaya, 5 Februari 2013

Penulis

## 1. PENDAHULUAN

### 1.1 Latar Belakang Masalah

Teknologi informasi yang berkembang pesat pada saat ini, membuat solusi baru dalam penyelesaian suatu masalah. Perusahaan, lembaga, atau organisasi memanfaatkan teknologi informasi sehingga kegiatan kesehariannya lebih efektif dan efisien.

Bidang pendidikan, saat ini, juga tidak ketinggalan memanfaatkan teknologi informasi. SMA Katolik Taruna Jaya Sampit merupakan salah satu sekolah yang memerlukan suatu sistem informasi akademik. Pada sistem sebelumnya, proses pencatatan, pengolahan, pencarian, dan administrasi data menggunakan data yang disimpan dalam bentuk *hard copy*.

Proses yang masih menggunakan *hard copy* ini memperlambat proses akademik yang berlangsung. Proses penghitungan nilai siswa memerlukan waktu yang cukup lama, karena data nilai yang jumlahnya banyak dan memerlukan ketelitian. Adanya kemungkinan data *hard copy* hilang akibat kesalahan penyimpanan atau kerusakan *hard copy*. Proses penyusunan jadwal akademik memerlukan waktu cukup lama, karena menyesuaikan jumlah guru yang tidak terlalu banyak, sehingga diperlukan *tools* yang membantu dalam penjadwalan akademik. Selain itu, diperlukan sistem informasi yang dapat menginformasikan nilai akademik, pelanggaran siswa, ataupun berita sekolah dengan cepat bagi orang tua siswa, dan membantu dalam proses penerimaan siswa baru.

Berdasarkan uraian sebelumnya, diperlukan sebuah sistem informasi akademik, khususnya menyangkut akademik kesiswaan dan administrasi data. Sistem informasi akademik dirancang untuk memenuhi kebutuhan peningkatan layanan pendidikan dalam hal kinerja dan kualitas pelayanan. Sistem informasi akademik membantu mengolah data yang sebelumnya masih berbentuk *hard copy* dan diharapkan mampu mengefektifkan waktu dan menekan biaya operasional.

## **1.2 Perumusan Masalah**

Berdasarkan latar belakang masalah yang telah dijelaskan, maka perumusan masalah yang dihadapi adalah:

1. Bagaimana membangun sistem informasi akademik yang dapat melakukan pengolahan data, yang berbentuk *hard copy*, seperti data pendaftaran siswa baru, data siswa, data orang tua siswa, data nilai siswa, data mata pelajaran, data jadwal pelajaran, data jadwal ujian, data pelanggaran siswa, dan data staf pengajar.
2. Bagaimana membuat informasi data akademik yang dapat diakses dengan baik dan informatif terhadap *user*.
3. Bagaimana membuat *tools* yang dapat membantu dalam penyusunan jadwal akademik di sekolah tersebut.

## **1.3 Tujuan Penelitian**

Tujuan dari penelitian ini adalah untuk membuat desain dan implementasi sistem informasi akademik pada SMA Katolik Taruna Jaya Sampit yang dapat membantu proses kegiatan akademik kesiswaan dan administrasi data sehari-hari lebih efektif dan efisien, termasuk proses penerimaan siswa baru, membantu para orang tua dalam memantau pendidikan siswa, dan membantu penjadwalan akademik di SMA Katolik Taruna Jaya Sampit dengan *tools* yang dikembangkan.

## **1.4 Manfaat Penelitian**

Hasil dari penelitian ini diharapkan dapat membantu dalam administrasi kegiatan akademik SMA Katolik Taruna Jaya Sampit.

## **1.5 Ruang Lingkup**

Ruang lingkup dibatasi pada:

1. Pembuatan bagian utama sistem informasi akademik  
Bagian utama sistem akademik ini berupa halaman *website* yang berisi tentang proses pendaftaran siswa baru, profil sekolah, staf pengajar, dan informasi atau berita (*news*) mengenai sekolah.
2. Pembuatan bagian administrasi sistem informasi akademik

Bagian ini berupa halaman *website* yang hanya dapat diakses administrator. Proses *input* dan *maintenance* data seperti data nilai, data jadwal pelajaran, jadwal mata pelajaran, data siswa, data orang tua, data staf pengajar, data pelanggaran siswa, serta data lainnya dilakukan pada bagian ini.

3. Pembuatan *tools* yang membantu penjadwalan akademik dengan menyesuaikan jumlah guru yang tidak terlalu banyak (guru tetap dan guru tidak tetap), banyak jam pelajaran setiap minggu, hari dimana guru berhalangan untuk mengajar.
4. Sistem informasi akademik ini dikembangkan dengan PHP dan HTML.
5. Sistem informasi akademik ini menggunakan MySQL sebagai penyimpanan *database*.
6. Pembuatan forum yang dapat diakses oleh siswa, guru, dan orang tua. Sehingga adanya *feedback* dari berbagai pihak mengenai proses akademik di sekolah.
7. Adanya menu profil lulusan atau alumni. Menu alumni berisi data mengenai kelanjutan pendidikan atau karir yang ditempuh setelah menyelesaikan studi di sekolah tersebut.

Fasilitas yang disediakan di dalam aplikasi *website* adalah:

- ❖ *Login* dan hak akses sebagai siswa atau orang tua siswa:
  1. *View* nilai akademik siswa.
  2. *View* jadwal pelajaran dan jadwal ujian siswa.
  3. *View* data siswa dan orang tua.
  4. *View* riwayat pelanggaran yang telah dilakukan oleh siswa.
  5. *Posting* suatu topik di dalam forum sekolah, *reply* post dari suatu topik di dalam forum sekolah.
  
- ❖ *Login* dan hak akses sebagai administrator:
  1. *Input* dan *maintenance* data pendaftaran siswa baru.
  2. *Input* dan *maintenance* data orang tua.
  3. *Input* dan *maintenance* data nilai siswa.
  4. *Input* dan *maintenance* data mata pelajaran.

5. *Input* dan *maintenance* data staf pengajar.
  6. *Input* dan *maintenance* data ujian.
  7. *Input* dan *maintenance* data jadwal pelajaran.
  8. *Manage* (*posting*, *edit*, atau *delete*) suatu topik dan *post* di dalam forum sekolah.
- ❖ *Login* dan hak akses sebagai guru:
1. *Input* data nilai pelajaran yang diajar ke dalam sistem.
  2. *Posting* suatu topik di dalam forum sekolah, *reply* post dari suatu topik di dalam forum sekolah.
- ❖ *Login* dan hak akses sebagai pendaftar atau calon siswa baru:
1. *Input* data pendaftaran siswa baru.
  2. Cek status dari proses pendaftaran. Jika diterima maka status berubah menjadi lulus, sedangkan jika ditolak status berubah menjadi tidak lulus dan data dihapus dari *database* pada saat proses pendaftaran siswa baru berakhir.

Laporan yang dihasilkan adalah laporan nilai akademik siswa, laporan jadwal akademik siswa, laporan pelanggaran siswa, laporan penerimaan siswa baru.

## **1.6 Sistematika Penyusunan Laporan**

Garis besar penyusunan laporan penelitian ini adalah sebagai berikut:

### **Bab I : PENDAHULUAN**

Bab ini berisi tentang latar belakang penulisan penelitian, perumusan masalah, tujuan, manfaat, ruang lingkup, metodologi yang digunakan serta sistematika penyusunan laporan.

### **Bab II : TINJAUAN PUSTAKA**

Bab ini berisi tentang teori-teori, metode dan konsep yang digunakan penulis dalam pembuatan penelitian.

### **Bab III: METODE PENELITIAN**

Bab ini berisi tentang metode penelitian yang dilakukan serta perancangan pembuatan perangkat lunak.

**Bab IV: HASIL PENELITIAN DAN PEMBAHASAN**

Bab ini berisi tentang hasil dari penelitian, berupa perangkat lunak yang telah dikembangkan beserta dengan pengujian perangkat lunak tersebut.

**Bab VI : KESIMPULAN DAN SARAN**

Bab ini berisi tentang kesimpulan dan saran yang muncul setelah pembahasan materi dan pengujian aplikasi.

## **2. LANDASAN TEORI**

### **2.1 Profil SMA Katolik Taruna Jaya Sampit**

SMA Katolik Taruna Jaya Sampit didirikan pada tahun 1985. SMA Katolik Taruna Jaya Sampit berlokasi di Jalan Letjen S. Parman no. 22, Kompleks Gereja Katolik, Sampit, Kalimantan Tengah. SMA ini merupakan salah satu SMA swasta yang dikenal baik kualitasnya oleh masyarakat dan termasuk favorit di Kota Sampit, Kalimantan Tengah. Hal ini didasarkan oleh tingkat kedisiplinan yang tinggi, cara pengajaran yang baik, dan fasilitas yang cukup memadai.

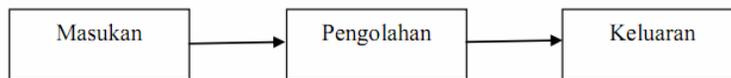
SMA Katolik Taruna Jaya Sampit merupakan salah satu lembaga pendidikan yang tergabung dalam Yayasan Siswarta, yang juga mengelola TK, SD, dan SMP, pada Kompleks Gereja Katolik, Sampit. Persentase tingkat kelulusan yang mencapai 100% pada Ujian Akhir Nasional (UAN) membuatnya menjadi salah satu sekolah pilihan yang cukup diminati para calon siswa baru tiap tahun ajaran.

Setiap tahun, SMA Katolik Taruna Jaya Sampit menerima siswa baru kurang lebih berjumlah 105 orang. SMA Katolik Taruna Jaya Sampit memiliki sembilan ruangan kelas. Dengan kapasitas kelas sebanyak tiga kelas untuk kelas X, tiga kelas untuk kelas XI, dan tiga kelas untuk kelas XII. Tiap-tiap kelas memiliki kapasitas untuk kurang lebih 35 orang siswa.

### **2.2 Sistem**

Menurut Jogiyanto (2005), definisi sistem:

Sistem dapat didefinisikan dengan pendekatan prosedur dan dengan pendekatan komponen. Dengan pendekatan prosedur, sistem dapat didefinisikan sebagai kumpulan dari prosedur-prosedur yang mempunyai tujuan tertentu. Sedangkan dengan pendekatan komponen didefinisikan sebagai kumpulan yang saling berhubungan satu dengan yang lainnya membentuk satu kesatuan untuk mencapai tujuan tertentu.



Gambar 2.1 Bentuk Umum Sistem

Sumber: Jogiyanto (2005)

Suatu sistem mempunyai karakteristik atau sifat-sifat yang khusus (Jogiyanto, 2005), yaitu:

1. Komponen sistem (*component*)  
Sistem terdiri dari sejumlah komponen yang saling berinteraksi atau bekerja bersama-sama yang membentuk satu kesatuan.
2. Batas sistem (*boundary*)  
Batas sistem merupakan daerah yang membatasi antara satu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem menunjukkan ruang lingkup (*scope*) dari suatu sistem.
3. Lingkungan luar sistem (*environment*)  
*Environment* adalah segala hal yang berada di luar batas sistem, yang mempengaruhi operasi sistem. Hal ini dapat bersifat menguntungkan atau merugikan.
4. Penghubung sistem (*interface*)  
*Interface* ialah media penghubung antara satu subsistem dengan subsistem yang lainnya. Sumber daya dapat mengalir antar tiap subsistem melalui *interface*. *Output* satu subsistem dapat menjadi input bagi subsistem lainnya melalui *interface*. Satu subsistem saling berintegrasi membentuk suatu kesatuan melalui *interface*.
5. Masukan sistem (*input*)  
*Input* merupakan energi yang dimasukkan ke dalam sistem. *Input* bisa berbentuk *maintenance input* atau *signal input*. *Maintenance input* adalah energi yang dimasukkan agar sistem beroperasi. *Signal input* adalah energi yang diproses untuk menjadi *output*.
6. Keluaran sistem (*output*)

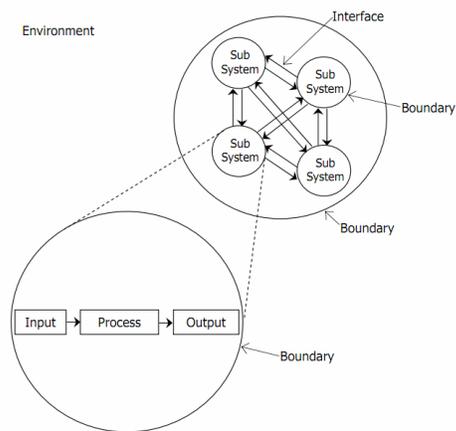
*Output* merupakan hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna (informasi) dan sisa pembuangan. Keluaran dapat menjadi masukan untuk suatu subsistem yang lain atau kepada supra sistem.

7. Pengolahan sistem (*process*)

Pengolahan merupakan bagian dari sistem yang mengubah suatu *input* menjadi suatu *output* (transformasi).

8. Sasaran sistem (*goal/objective*)

Suatu sistem pasti memiliki sasaran atau tujuan (*goal/objective*). Sasaran sistem sangat menentukan jenis *input* yang diperlukan oleh sistem dan *output* yang dihasilkan sistem. Sistem bisa dikatakan sukses atau berhasil jika dapat mencapai *goal/objective* yang telah ditentukan sebelumnya.



Gambar 2.2 Karakteristik Sistem

Sumber: Jogiyanto (2005)

### 2.3 Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya (Jogiyanto, 2005). Kualitas suatu informasi dapat ditentukan oleh beberapa hal, yaitu:

a. Akurat

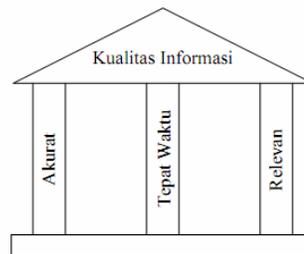
Informasi harus jelas mencerminkan maksud yang disampaikan, bebas dari kesalahan-kesalahan, dan tidak bias. Ukuran keakuratan informasi amat bervariasi dan bergantung pada sifat informasi yang dihasilkan.

b. Tepat waktu

Umur informasi ialah faktor penting dalam penentuan kegunaan informasi. Tepat waktu berarti informasi tidak boleh lebih lama dari periode waktu yang diperlukan. Informasi yang usang akan mempunyai nilai lebih rendah.

c. Relevan

Informasi dikatakan relevan apabila mempunyai suatu manfaat untuk penggunaannya. Relevansi informasi bagi tiap penerima informasi bisa berbeda-beda.



Gambar 2.3 Kualitas Informasi

Sumber: Jogiyanto (2005)

## 2.4 Sistem Informasi

Sistem informasi adalah suatu susunan dari orang, data, proses, dan teknologi informasi yang berinteraksi untuk mengumpulkan, memproses, menyimpan, dan menyediakan *output* informasi yang diperlukan untuk mendukung sebuah organisasi (Bentley & Whitten, 2007).

Sedangkan menurut Al-Bahra Bin Ladjamudin (2005), sistem informasi adalah suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.

Raymond McLeod, Jr. & George P. Schell (2004) berpendapat bahwa proses pengembangan sistem memiliki lima tahapan, yaitu:

1. Perencanaan (*Planning*)

Perencanaan adalah tahapan awal pengembangan suatu sistem. Tindakan pada tahap ini adalah identifikasi dan mendefinisikan masalah, pembuatan tujuan, membuat studi kelayakan, persiapan usulan penelitian, menyetujui atau menolak penelitian, dan penetapan mekanisme kontrol.

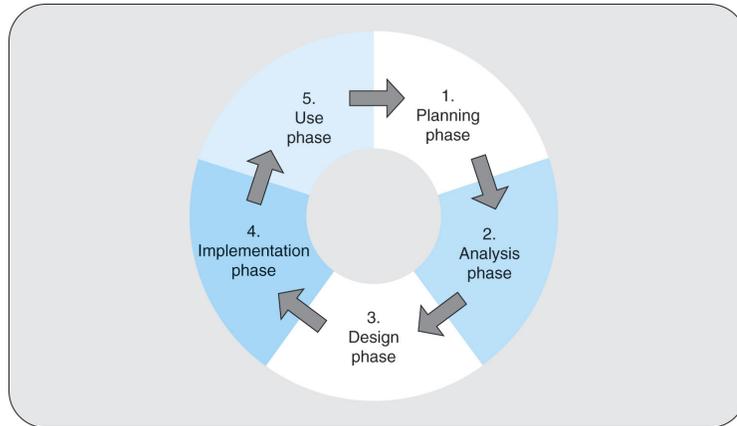
2. Analisis (*Analysis*)

Analisis adalah penelitian sistem yang telah ada untuk pengembangan sistem yang baru. Pada tahap analisis, tindakan yang dilakukan adalah mendefinisikan kebutuhan informasi, mendefinisikan kriteria kinerja sistem, menyiapkan usulan rancangan sistem baru, dan menyetujui atau menolak rancangan proyek.
3. Rancangan (*Design*)

Rancangan adalah penentuan proses dan data yang dibutuhkan sistem baru. Pada proses *design*, *Data Flow Diagram* dipakai untuk menggambarkan rancangan sistem yang akan dikerjakan. Tindakan pada tahap ini adalah persiapan rincian rancangan, identifikasi pengaturan sistem, evaluasi pengaturan sistem, pemilihan konfigurasi yang paling sesuai, persiapan usulan implemmentasi, dan menyetujui atau menolak penerapan sistem.
4. Implementasi (*Implementation*)

Implementasi ialah kegiatan memperoleh dan integrasi sumber daya (fisik dan konseptual) dalam pembuatan sistem baru. Dalam tahap ini, kegiatan yang dilakukan adalah perencanaan implementasi, mempersiapkan perangkat keras (*hardware*) dan perangkat lunak (*software*), persiapan dan pembuatan database, persiapan fasilitas fisik, pelatihan *user*, dan penerapan sistem baru.
5. Penggunaan (*Use*)

Pada tahap ini, kegiatan yang dilakukan ialah penggunaan sistem, audit sistem, dan perawatan (*maintenance*) sistem. Pada proses penggunaan sistem, *user* diharapkan menggunakan sistem sesuai tujuan (*objective*). Sedangkan audit sistem, dilakukan studi untuk mengetahui sampai mana sistem dapat memenuhi tujuan yang ditentukan. Selain *maintenance* sistem, modifikasi sistem juga diperlukan agar sistem tetap dapat mendukung proses penyelesaian pekerjaan.



Gambar 2.4 Tahapan Pengembangan Sistem

Sumber: McLeod, Jr & Schell (2004)

## 2.5. *Hypertext Preprocessor (PHP)*

Menurut Bachtiar (2008), PHP adalah sebuah bahasa pemrograman yang didesain agar dapat disisipkan dengan mudah ke halaman HTML (*HyperText Markup Language*). PHP memberikan solusi yang murah (gratis digunakan) dan dapat berjalan pada berbagai *platform*. Sebagai bahasa *web programming*, PHP membutuhkan *web server* untuk menunjang pengoperasiannya. *Web server* ialah suatu perangkat lunak *server* yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan *web browser* dan mengirimkan kembali hasilnya berupa halaman *website* yang berformat HTML (Kurniawan, 2008).

Menurut Sibero (2011), terdapat dua cara penulisan baris kode PHP, yaitu penulisan baris kode menggunakan format PHP dan penulisan baris kode menggunakan format gabungan antara PHP dan HTML. Beberapa hal yang perlu diketahui mengenai cara penulisan baris kode PHP yaitu :

- Penulisan baris kode PHP dimulai dengan *tag* `<?php` dan diakhiri dengan *tag* `?>`.
- Penulisan baris kode PHP harus diakhiri dengan tanda semikolon atau titik koma (;).

- Penulisan pada PHP bersifat *case-sensitive*, contohnya huruf b dan huruf B merupakan cara penulisan yang berbeda.
- Komentar yang dituliskan pada baris kode PHP tidak akan diproses untuk ditampilkan pada halaman *website*. Cara penulisan komentar pada PHP, apabila hanya sepanjang satu baris, bisa menggunakan tanda // kemudian diikuti baris komentar. Sedangkan penulisan komentar lebih dari satu baris, *syntax* penulisannya ialah pada awal baris komentar diberikan tanda /\* dan pada akhir penulisan baris komentar ditutup dengan tanda /\*.
- Penulisan teks pada baris kode PHP diapit dengan tanda ‘ atau “
- Penulisan tanda baca (‘, ”, \, /, dsb.) diawali tanda \ kemudian diikuti dengan tanda baca yang akan dituliskan.

Variabel pada PHP digunakan untuk media penyimpanan atau penampung nilai atau hasil proses yang mempunyai suatu jenis tipe data tertentu. Pada PHP, penulisan variabel diawali dengan karakter '\$'. Variabel pada PHP tidak perlu dideklarasikan dan ditetapkan jenis datanya sebelum variabel itu digunakan. Nama variabel bersifat *case-sensitive*.

## 2.6 *My Structured Query Language (MySQL)*

MySQL (*My Structured Query Language*) adalah *database* yang menghubungkan *script* PHP menggunakan perintah *query* dan *escape character* yang sama dengan PHP (Kurniawan, 2008). MySQL mempunyai tampilan *client* yang mempermudah dalam mengakses *database* dengan kata sandi untuk menyetujui proses yang boleh dilakukan.

Perintah yang dapat digunakan dalam *database* MySQL adalah:

### 1. *Create Database*

Digunakan untuk membuat *database* baru.

Sintaks:

```
CREATE DATABASE nama_database;
```

### 2. *Use Database*

Digunakan untuk mengaktifkan *database* yang digunakan.

Sintaks:

```
USE DATABASE nama_database;
```

3. *Show Databases*

Digunakan untuk menampilkan *database* yang telah ada.

Sintaks:

```
SHOW DATABASES;
```

4. *Show Tables*

Digunakan untuk melihat daftar tabel yang ada di dalam *database*.

Sintaks:

```
SHOW TABLES FROM nama_database;
```

5. *Drop Database*

Digunakan untuk menghapus *database*.

Sintaks:

```
DROP DATABASE nama_database;
```

*Data Definition Language* (DDL) merupakan sub bahasa SQL yang berfungsi mendefinisikan *database* dan tabel. Perintah yang digunakan ialah *create*, *alter*, dan *drop* (Nugroho, 2008).

1. *Create Table*

Digunakan untuk membuat tabel baru pada suatu tabel.

Sintaks:

```
CREATE TABLE nama_tabel (nama_kolom tipe_kolom (panjang), ...);
```

2. *Alter Table*

Digunakan untuk menangani perubahan struktur yang ada di dalam tabel.

Sintaks:

```
ALTER TABLE nama_tabel [jenis_perubahan];
```

Jenis\_perubahan dapat berupa *change*, *add*, *drop*, dan *rename*.

3. *Drop Table*

Digunakan untuk menghapus tabel.

Sintaks:

```
DROP TABLE nama_tabel;
```

Menurut Nugroho (2008), *Data Manipulation Language* (DML) ialah sub bahasa SQL yang berfungsi memanipulasi data yang ada di dalam *database*. Perintah yang termasuk di dalam DML ialah *insert*, *select*, *update*, dan *delete*.

1. *Insert*

Digunakan untuk memasukkan baris data pada tabel.

Sintaks:

```
INSERT INTO nama_tabel (nama_kolom) VALUES (nilai_kolom);
```

2. *Select*

Digunakan untuk menyeleksi atau memilih baris data pada table.

Sintaks:

```
SELECT nama_kolom FROM nama_tabel WHERE kondisi;
```

3. *Update*

Digunakan untuk mengubah baris data pada tabel.

Sintaks:

```
UPDATE nama_tabel SET nama_kolom = nilai_kolom WHERE kondisi;
```

4. *Delete*

Digunakan untuk menghapus baris data pada tabel.

Sintaks: DELETE FROM nama\_tabel WHERE kondisi;

## 2.7 *Data Flow Diagram (DFD)*

*Data Flow Diagram* ialah sebuah grafik representatif dari sebuah sistem yang menggunakan 4 buah bentuk simbol untuk menggambarkan bagaimana aliran data melalui proses yang saling berhubungan (McLeod, Jr. & Schell, 2004).

Sedangkan menurut Jogiyanto (2005), *Data Flow Diagram* ialah diagram yang menggunakan notasi-notasi atau simbol-simbol untuk menggambarkan sistem jaringan kerja antar fungsi-fungsi yang berhubungan satu sama lain dengan aliran dan penyimpanan data (Jogiyanto, 2005).

*Data Flow Diagram* terdiri dari empat buah simbol (McLeod, Jr. & Schell, 2004), yaitu:

1. *Terminator*

*Terminator* seringkali digunakan untuk mendeskripsikan bagian dari lingkungan, karena merupakan batas dari sebuah sistem. *Terminator*, disebut juga *entity*, memberikan masukan (*input*) ke dalam sistem dan menerima keluaran (*output*) dari sistem. Sebuah *terminator* dapat berupa seseorang, sebuah organisasi, ataupun sistem yang lainnya. *Terminator* biasanya digambarkan dalam bentuk kotak atau segiempat.

2. Proses

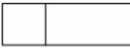
Proses adalah sesuatu yang merubah *input* menjadi *output*. Simbol dari proses adalah lingkaran, sebuah segiempat horizontal, atau persegi panjang tegak dengan ujung yang bundar. Setiap proses diidentifikasi dengan label. Teknik paling umum dari pemberian label ialah menggunakan kata kerja dan sebuah objek, tapi bisa juga menggunakan nama dari sistem atau program komputer.

3. *Data Flow*

Sebuah *data flow* berisi sekelompok elemen data yang berhubungan secara logis yang mengalir dari sebuah poin atau proses dari satu ke lainnya. Simbol *data flow* berupa panah yang menunjukkan aliran data. Data yang dibawa oleh *data flow* harus disebutkan dan diletakkan di atas lambang *data flow*.

4. *Data Store*

*Data store* adalah sebuah repositori data, tempat penyimpanan data. *Data flow* yang menuju *data store* adalah kegiatan menulis data, sehingga isi data akan berubah. Sedangkan *data flow* yang menuju ke proses dari *data store*, adalah kegiatan membaca data, sehingga isi data tidak berubah. Simbol *data store* adalah sebuah segiempat yang ujungnya terbuka, oval, atau sepasang garis paralel.

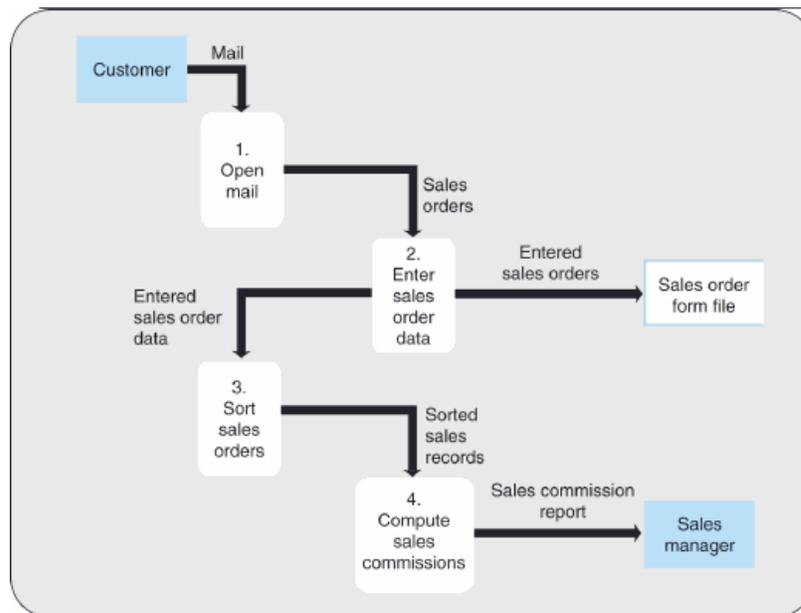
Simbol	Arti
	<i>Terminator</i>
	Proses
	<i>Data Flow</i>
	<i>Data Store</i>

Gambar 2.5 Simbol DFD

Dalam pembuatan DFD, ada beberapa aturan yang ditetapkan, yaitu:

1. Antar *terminator (entity)* tidak diijinkan terjadi hubungan atau relasi.

2. Tidak boleh ada *data flow* antara *entity* eksternal dengan *data store*.
3. Untuk kerapian pembuatan (menghindari aliran data yang bersilangan), *entity* eksternal atau *data store* boleh digambar beberapa kali dengan tanda khusus, misalnya diberi nomor.
4. Satu aliran data boleh mengalirkan beberapa paket data.
5. Bentuk anak panah aliran data boleh bervariasi.
6. Semua objek harus mempunyai nama.
7. Aliran data selalu diawali atau diakhir dengan proses.
8. Semua aliran data harus mempunyai tanda arah.
9. Jumlah proses tidak lebih dari sembilan proses dalam sistem, jika melebihi maka sebaiknya dikelompokkan beberapa proses yang bekerja bersama-sama di dalam suatu subsistem.



Gambar 2.6 Contoh DFD

Sumber: McLeod, Jr. & Schell (2004)

Langkah-langkah pembuatan DFD yaitu :

1. *Context Diagram*

Diagram yang berisi sebuah simbol proses tunggal yang menggambarkan keseluruhan sistem (McLeod, Jr. & Schell, 2004).

2. *Figure n Diagram*

Diagram yang mendokumentasikan sebuah detail yang lebih jelas dari sebuah proses yang terdapat pada DFD (McLeod,. Jr. & Schell, 2004).

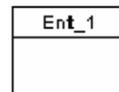
## 2.8 Entity Relationship Diagram (ERD)

*Entity Relational Diagram* termasuk dalam jenis pemodelan data konseptual yang digunakan dalam proses pembuatan basis data bertipe relasional. ERD adalah rincian yang merupakan representasi logika dari data pada suatu organisasi atau area bisnis tertentu. (Fathansyah, 2007).

Model ER memiliki beberapa komponen dasar :

### 1. Entitas (*entity*)

Entitas adalah individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain.



Gambar 2.7 Entity

### 2. Atribut

Atribut adalah karakteristik (*property*) yang dideskripsikan oleh entitas.



Gambar 2.8 Atribut

### 3. Relasi (*relationship*)

Relasi adalah hubungan antara suatu himpunan entitas dengan himpunan entitas yang lain.



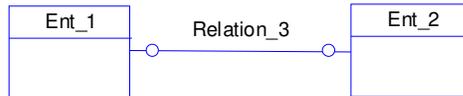
Gambar 2.9 Relasi

### 4. Kardinalitas atau Derajat Relasi

Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Kardinalitas relasi yang terjadi di antara dua himpunan entitas dapat berupa:

- Satu ke satu (*one to one*)

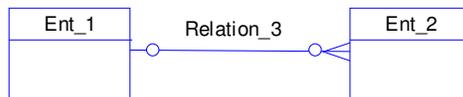
Setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, begitupun sebaliknya.



Gambar 2.10 *One to One Relationship*

- Satu ke banyak (*one to many*)

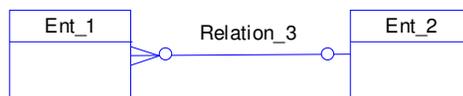
Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.



Gambar 2.11 *One to Many Relationship*

- Banyak ke satu (*many to one*)

Setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, tetapi tidak sebaliknya dengan entitas B.



Gambar 2.12 *Many to One Relationship*

- Banyak ke banyak (*many to many*)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian sebaliknya, dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.



Gambar 2.13 *Many to Many Relationship*

### 3. METODOLOGI PENELITIAN

#### 3.1. Metodologi Penelitian

Langkah-langkah dalam pengerjaan penelitian ini adalah:

1. Studi literatur
  - 1.1. Mempelajari literatur yang berkaitan dengan:
    - PHP
    - MySQL
    - HTML
2. Pengumpulan data
  - 2.1. Melakukan wawancara tanya jawab dengan kepala sekolah, dewan guru, serta tata usaha.
  - 2.2. Survei langsung mengamati sistem yang ada dan mencari masalah-masalah yang timbul.
  - 2.3. Pencatatan data yang terdapat di *hard copy*.
3. Perancangan sistem dan *database*
  - 3.1. Pembuatan ERD dan DFD.
  - 3.2. Pembuatan *database*.
  - 3.3. Pembuatan *interface*.
  - 3.4. Pembuatan *tools* penjadwalan.
4. Pengujian dan evaluasi program
  - 4.1. Pengujian program yang telah dibuat, sesuai dengan tujuan awal pembuatan program.
  - 4.2. Analisis *output* program.
5. Pelatihan aplikasi  
Pelatihan *user* yang menggunakan sistem terutama bagian administrasi.
6. Pengambilan kesimpulan  
Pengambilan kesimpulan dari hasil aplikasi yang telah dibangun.
7. Pembuatan laporan  
Laporan mengenai penelitian yang telah dibuat.

### **3.2. Pengumpulan Data**

Tahap pertama yang dilakukan dalam proses pengumpulan data ialah melakukan wawancara dengan pihak SMA Katolik Taruna Jaya Sampit. Wawancara dilakukan dengan kepala sekolah, dewan guru, dan tata usaha. Wawancara bertujuan agar mengetahui lebih jelas mengenai sistem yang terdapat di sekolah tersebut. Bagian sistem yang ditanyakan pada saat wawancara ialah proses pendaftaran siswa baru, perhitungan nilai akademik, administrasi data (siswa, orang tua, dan guru), pembuatan jadwal akademik, kegiatan akademik siswa, dan pelanggaran siswa.

Tahap yang dilakukan selanjutnya, setelah wawancara, adalah pelaksanaan survei secara langsung mengamati sistem yang telah ada dan mencari masalah-masalah yang timbul pada sistem. Setelah pelaksanaan survei, dilakukan proses pencatatan data yang terdapat pada *hard copy* seperti data siswa, data guru, data mata pelajaran, data jadwal pelajaran, data pelanggaran siswa, data pendaftaran siswa baru, data absensi, data orang tua, dan data nilai akademik.

### **3.3. Analisis Sistem**

Langkah pertama yang dilakukan sebelum merancang sebuah sistem ialah analisa permasalahan dari sistem yang sudah ada. Proses tersebut bertujuan menemukan masalah pada sistem yang sudah ada. Setelah mengetahui masalah yang terdapat pada sistem, pembuatan solusi sistem dapat dilakukan dengan lebih terarah untuk memperbaiki kekurangan pada sistem sebelumnya. Berikut ini merupakan sistem yang terdapat di SMA Katolik Taruna Jaya Sampit:

#### **3.3.1. Administrasi Data**

Sistem administrasi data di SMA Katolik Taruna Jaya Sampit sebelumnya masih menggunakan bentuk *hard copy*. Proses pencatatan data ini dilakukan oleh bagian tata usaha sekolah. Proses penyimpanan data akademik, seperti data siswa, data nilai, data orang tua, data guru, data pelanggaran siswa, dan data pendaftaran siswa baru disimpan dalam suatu buku. Penyimpanan menggunakan cara ini beresiko besar kehilangan data akibat kerusakan *hard copy*, kehilangan *hard copy* karena ruangan penyimpanan data dijadikan satu bercampur dengan berbagai arsip

data yang lain. Proses penyimpanan ini membuat pencarian data menjadi lebih lama karena bercampur dengan arsip data yang lain.

### **3.3.2. Pendaftaran Siswa Baru**

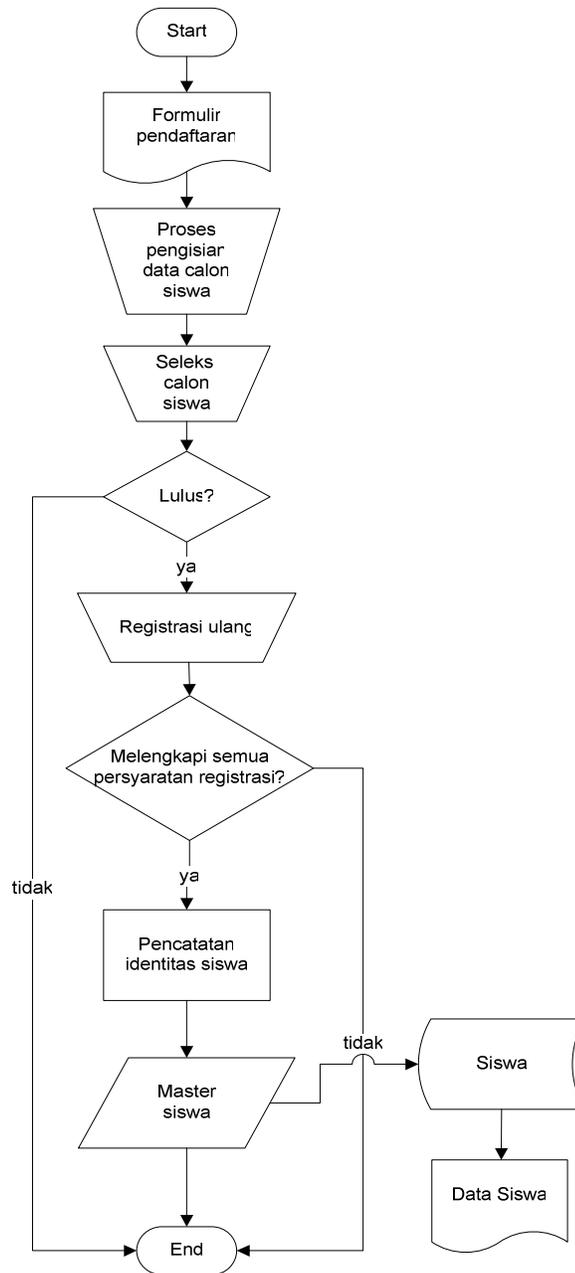
SMA Katolik Taruna Jaya Sampit memiliki kuota maksimum sembilan ruangan kelas yang dapat dipergunakan. Kuota untuk kelas X sebanyak tiga kelas, kuota untuk kelas XI sebanyak tiga kelas, dan kuota untuk kelas XII sebanyak tiga kelas. Setiap tahun, SMA Katolik Taruna Jaya Sampit menerima siswa baru kurang lebih berjumlah 105 orang.

Proses penerimaan siswa baru dimulai dengan proses pengambilan formulir di bagian tata usaha sekolah. Panitia penerimaan siswa baru kemudian menjelaskan informasi tentang persyaratan dan kewajiban yang harus dilengkapi.

Formulir pendaftaran berisi pertanyaan tentang data siswa dan data orang tua siswa yang harus dilengkapi. Data siswa terdiri dari nama, tanggal dan tempat lahir, asal sekolah, alamat, agama, dan data keterangan lainnya. Data orang tua siswa berisi nama orang tua (ayah, ibu, atau wali), alamat, dan data keterangan lainnya.

Setelah calon siswa melakukan pendaftaran, dilakukan proses seleksi penerimaan siswa baru. Proses seleksi ini dilakukan berdasarkan pada nilai yang dimiliki para calon siswa. Panitia kemudian mengumumkan hasil penerimaan siswa baru pada papan pengumuman di sekolah.

Apabila calon siswa tersebut dinyatakan lulus, maka diwajibkan untuk registrasi kembali. Sedangkan apabila calon siswa tersebut dinyatakan tidak lulus penerimaan atau siswa tersebut sudah dinyatakan lulus tetapi tidak melakukan registrasi ulang, maka calon siswa tersebut tidak diterima dan datanya tidak akan dicatat. Setelah melakukan registrasi ulang, maka calon siswa tersebut telah resmi menjadi siswa SMA Katolik Taruna Jaya Sampit. Siswa tersebut kemudian memiliki nomer induk, yang format nomer induknya meneruskan nomer induk yang terdapat pada data *hard copy* siswa SMA Katolik Taruna Jaya Sampit dan kemudian dicatat ke dalam data siswa, yang dapat dilihat pada Gambar 3.1.



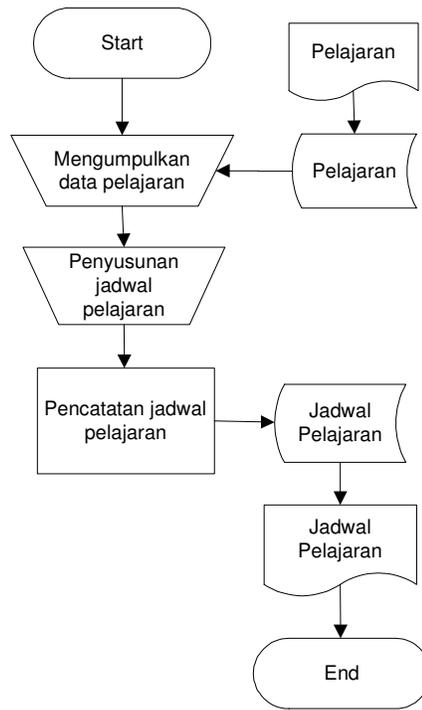
Gambar 3.1 Pendaftaran Siswa Baru

### 3.3.3. Penjadwalan Akademik

Penjadwalan sering menjadi masalah dalam setiap tahun ajaran. Hal ini diakibatkan adanya masalah seperti kekurangan jumlah guru (guru tetap dan guru tidak tetap), banyak jam pelajaran yang wajib dilaksanakan tiap minggu, atau ada hari dimana guru berhalangan untuk mengajar. Sehingga, walau jadwal akademik

telah dibuat, sewaktu-waktu dapat dipindah kembali sesuai dengan kondisi jumlah tenaga pengajar.

Penyusunan jadwal pelajaran ini dimulai dengan pengumpulan data pelajaran yang akan disusun. Setelah itu dilakukan penyusunan jadwal pelajaran oleh bagian kurikulum dan akademik. Data jadwal pelajaran tersebut kemudian akan disimpan dalam bentuk dokumen data jadwal pelajaran, yang dapat dilihat pada Gambar 3.2.



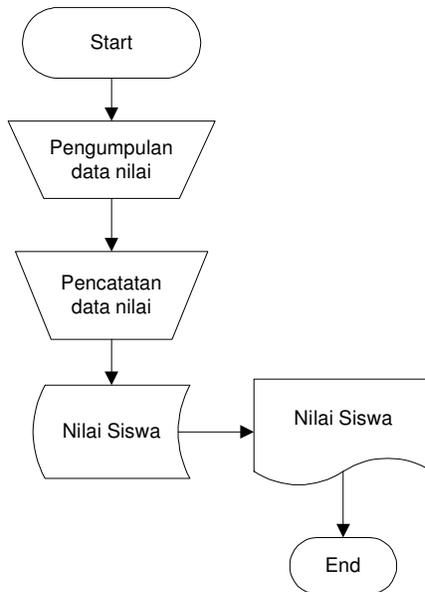
Gambar 3.2 Penjadwalan Akademik

#### 3.3.4. Nilai Siswa

Proses penghitungan nilai siswa bergantung dari nilai harian, tugas, dan ulangan umum. Banyaknya pengambilan nilai harian dan tugas dapat ditentukan oleh guru yang bersangkutan. Namun, tetap ada suatu rumus penghitungan nilai akhir yang telah ditetapkan oleh sekolah.

Nilai setiap siswa dalam suatu kelas akan dikumpulkan oleh wali kelas, dan kemudian dicatat dalam bentuk *hard copy* oleh bagian tata usaha. Nilai akademik

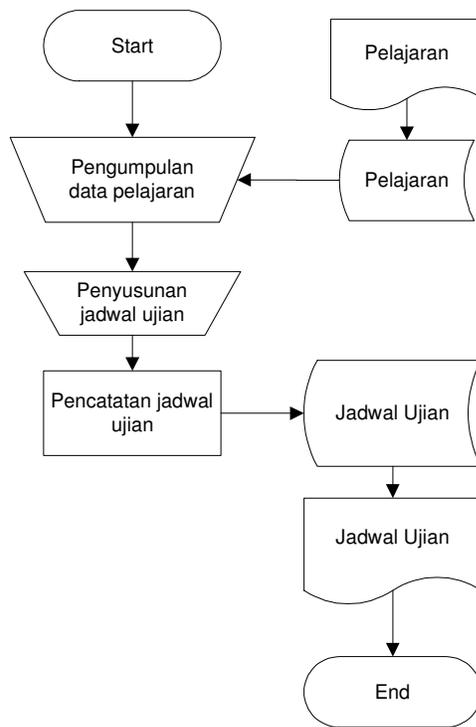
siswa ini kemudian disimpan menjadi nilai rapor pada akhir semester, yang dapat dilihat pada Gambar 3.3.



Gambar 3.3 Penilaian Akademik Siswa

### 3.3.5. Penjadwalan Ujian

Sama halnya seperti penjadwalan akademik, penjadwalan ujian juga disusun oleh bagian kurikulum dan akademik. Data pelajaran dikumpulkan untuk kemudian disusun menjadi jadwal ujian. Jadwal tersebut diberikan kepada bagian tata usaha untuk diketik dan dibagikan kepada seluruh siswa baik, baik dengan cara ditempelkan pada papan pengumuman atau diberikan langsung pada tiap siswa melalui wali kelas masing-masing. Data jadwal ujian akan disimpan dalam bentuk dokumen jadwal ujian, yang dapat dilihat pada Gambar 3.4.

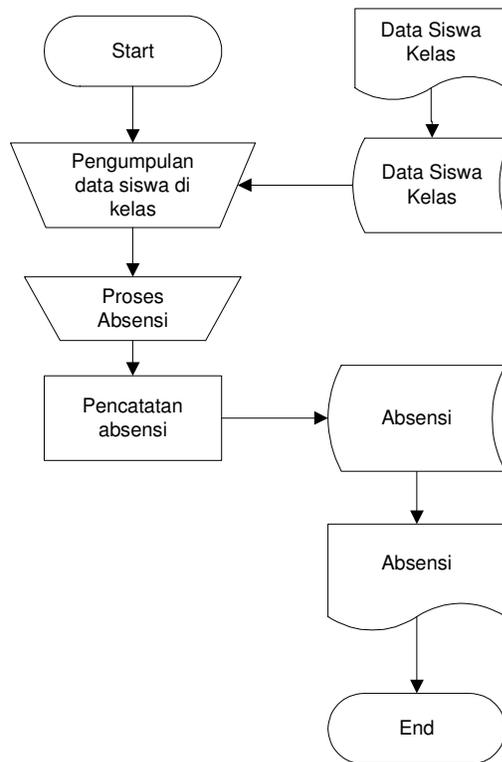


Gambar 3.4 Penjadwalan Ujian

### 3.3.6. Absensi

Proses absensi dilakukan secara manual pada jam pelajaran pertama setiap harinya. Proses absensi dilakukan sesuai dengan data siswa-siswi yang berada di suatu kelas.

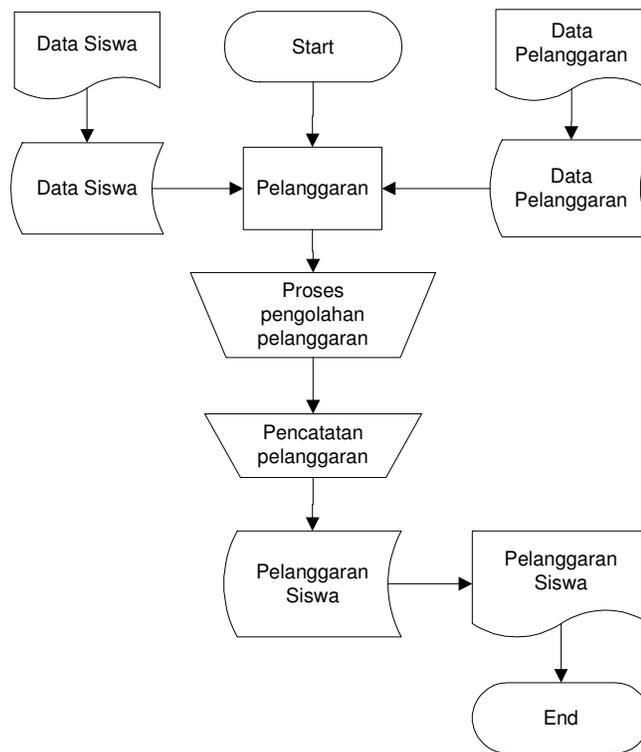
Proses absensi dilakukan oleh wali kelas. Status absensi terdiri dari masuk, sakit, ijin, atau alpha. Proses absensi ini akan direkap dalam sebuah buku absensi. Proses absensi ini dapat dilihat lebih jelas pada Gambar 3.5.



Gambar 3.5 Absensi

### 3.3.7. Pelanggaran Siswa

Proses pelanggaran siswa diatur dalam buku tata tertib pelanggaran siswa. Buku ini memiliki catatan tipe jenis pelanggaran dan poin pelanggarannya. Buku pelanggaran siswa mencatat berbagai pelanggaran yang dilakukan oleh siswa selama masa studi. Buku ini dipegang oleh wali kelas masing-masing siswa. Apabila terjadi pelanggaran, maka dilakukan pencatatan pelanggaran yang berisi jenis pelanggaran, tanggal terjadi pelanggaran, dan poin pelanggaran. Proses pelanggaran siswa ini dapat dilihat pada Gambar 3.6.



Gambar 3.6 Pelanggaran Siswa

### 3.4. Analisis Permasalahan

Berdasarkan analisa sistem yang telah dilakukan, terdapat beberapa permasalahan sebagai berikut:

- Proses pencatatan data yang dilakukan oleh bagian tata usaha SMA Katolik Taruna Jaya Sampit bisa dikatakan lengkap, walaupun kurang tertata. Hal ini disebabkan oleh kurangnya tenaga kerja di bagian tata usaha.
- Penyimpanan dengan *hard copy* dapat memiliki resiko kerusakan data, kehilangan data.
- Pembuatan jadwal pelajaran membutuhkan waktu yang lama.
- Informasi nilai, absensi, jadwal pelajaran, pelanggaran, dan jadwal ujian susah dipantau oleh pihak orang tua murid.
- Pihak sekolah memerlukan waktu yang lama dalam akses informasi nilai, pelanggaran, dan absensi siswa.

### 3.5. Analisis Kebutuhan

Berdasarkan analisa permasalahan yang telah dilakukan, maka SMA Katolik Taruna Jaya membutuhkan pengembangan sistem sebagai berikut:

- Sistem informasi yang mengatur administrasi data akademik sehingga lebih tertata.
- Sistem informasi yang dapat memberikan informasi nilai, absensi, jadwal ujian, jadwal pelajaran, dan pelanggaran siswa bagi orang tua atau siswa.
- Sistem informasi yang membantu penilaian akademik, pelanggaran siswa, dan absensi siswa.
- Sistem informasi yang membantu penjadwalan.
- Sistem informasi yang membantu pendaftaran siswa baru.

### 3.6. Desain Sistem Informasi Akademik

Dari hasil wawancara, survei sistem yang telah ada, serta pencatatan data, maka dirancang sistem informasi akademik berbasis *web* untuk membantu kegiatan akademik di sekolah SMA Katolik Taruna Jaya. Perancangan sistem yang baru dibuat dengan DFD (*Data Flow Diagram*) dan ERD (*Entity Relational Diagram*).

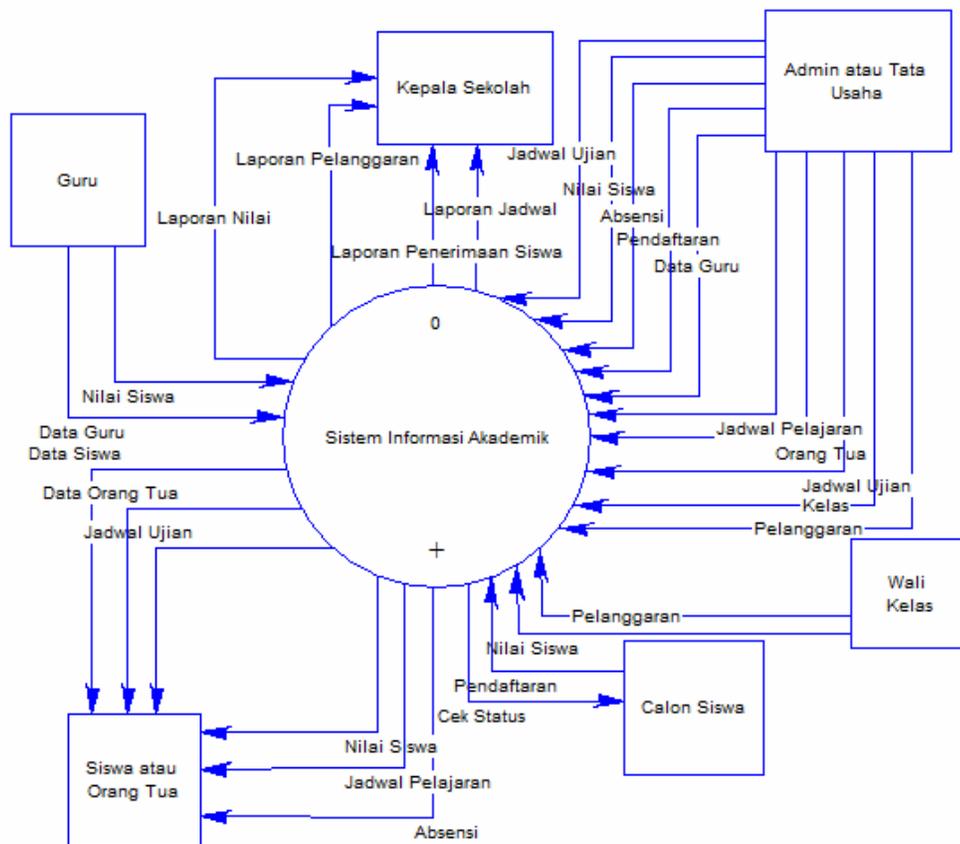
#### 3.6.1. *Data Flow Diagram*

Berdasarkan analisa yang telah dilakukan, dilakukan pembuatan DFD yang terdiri dari *Context Diagram*, DFD Level 1, DFD Administrasi, DFD Nilai.

##### 3.6.1.1. *Context Diagram*

*Context Diagram* digunakan mempermudah melihat aliran data yang ada di dalam sistem. Pertama-tama dibuat *context diagram* yang memberikan gambaran sistem keseluruhan, dan setelah itu *decompose* proses menjadi bagian-bagian yang lebih detail. Pada *Context Diagram* yang dapat dilihat pada Gambar 3.7. terdapat 6 *entity* yaitu administrator atau tata usaha, guru, siswa atau orangtua siswa, kepala sekolah, wali kelas, dan calon siswa. Tiap *entity* memiliki *data flow* dalam sistem informasi akademik. Admin atau tata usaha memberikan *data flow* berupa

data administrasi sekolah. Guru memberikan *data flow* berupa nilai siswa dan data guru. Wali kelas memberikan *data flow* berupa pelanggaran dan nilai siswa. Murid atau orang tua siswa menerima *data flow* berupa berbagai *display* data akademik siswa. Calon siswa memberikan *data flow* berupa data calon siswa dan menerima *data flow* berupa status pendaftaran. Kepala sekolah menerima *data flow* berupa laporan akademik.



Gambar 3.7 Context Diagram

### 3.6.1.2. DFD Level 0

Terdapat 4 proses dalam DFD Level 0 yaitu pendaftaran, administrasi, nilai, dan pelanggaran. Proses-proses tersebut ialah detail sistem dari *context diagram* sistem informasi akademik.

Pada proses pendaftaran, calon siswa dapat melakukan *input* data pendaftaran yang berupa data diri calon siswa. Setelah itu calon siswa dapat

melakukan *check* status untuk mengetahui status pendaftarannya, yaitu diterima atau tidak diterima. Admin dapat melakukan pengelolaan data pada proses pendaftaran siswa baru. Hasil proses pendaftaran ini kemudian disimpan dalam *data store* pendaftaran. Proses pendaftaran ini menghasilkan laporan penerimaan yang dapat dilihat oleh kepala sekolah.

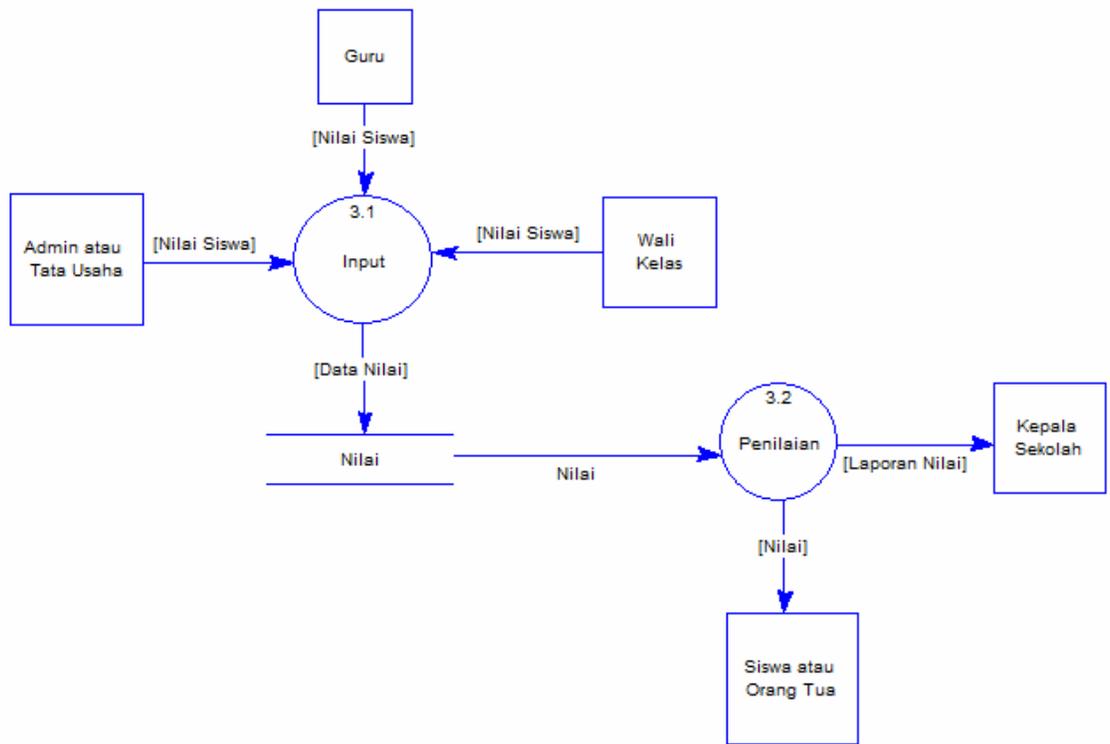
Proses administrasi data mengatur proses berbagai data akademik seperti jadwal pelajaran, jadwal ujian, data absensi, siswa, kelas, dan orang tua. Pada proses ini dilakukan penyimpanan data dan *display* laporan yang berkaitan dengan data akademik tersebut. *Display* ini berupa laporan akademik berkaitan dengan siswa yang dapat dilihat oleh siswa atau orang tua siswa.

Pada proses nilai, dilakukan proses *input* nilai. Setelah itu, data nilai yang telah disimpan akan melalui proses perhitungan nilai untuk menghasilkan nilai akhir dan laporan nilai. Hasil proses perhitungan nilai ini dapat dilihat oleh kepala sekolah berupa laporan nilai dan dapat dilihat oleh siswa atau orang tua siswa berupa data nilai akademik siswa.

Pada proses pelanggaran, mengatur berbagai proses berkaitan dengan proses pelanggaran yang dilakukan oleh siswa pada masa studi. Data pelanggaran ini akan disimpan pada *data store* pelanggaran. *History* pelanggaran ini dapat dilihat oleh kepala sekolah sebagai laporan pelanggaran dan dapat dilihat juga oleh siswa atau orang tua siswa berupa data pelanggaran siswa. Proses ini dapat dilihat pada Gambar 3.8.







Gambar 3.10 DFD *Level 1* Nilai

### 3.5.2. *Entity Relationship Diagram*

*Entity Relationship Diagram* (ERD) yang terdapat pada desain sistem dapat dibedakan dua, yaitu *Conceptual Data Model* dan *Physical Data Model*. *Conceptual Data Model* dapat dilihat pada Gambar 3.11.





### 3.7.1. Tabel absensikelas

Digunakan untuk menyimpan data absensi kelas pada suatu tahun ajaran.

*Field-field* yang terdapat pada tabel absensikelas dapat dilihat pada Tabel 3.1.

Tabel 3.1. Tabel absensikelas

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
noinduk	varchar	4	*	Menyimpan nomer induk siswa
tahunajaran	varchar	9	*	Menyimpan tahun ajaran
tgl	date		*	Menyimpan tanggal absensi
status	varchar	5		Menyimpan status absensi siswa (absen, sakit, ijin, masuk)
idkelas	varchar	5	*	Menyimpan id kelas

### 3.7.2. Tabel alumni

Digunakan untuk menyimpan data-data dari kuesioner alumni. *Field-field* yang terdapat pada tabel alumni dapat dilihat pada Tabel 3.2.

Tabel 3.2. Tabel alumni

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
idkuesioner	integer		*	Menyimpan id kuesioner alumni
rencana	varchar	25		Menyimpan jawaban rencana
lainnya	varchar	15		Menyimpan rencana lainnya
namajurusan	varchar	40		Menyimpan nama jurusan
tingkatpendidikan	varchar	5		Menyimpan tingkat pendidikan
namalembaga	varchar	70		Menyimpan nama lembaga
jurusan	varchar	10		Menyimpan jurusan
tempatkerja	varchar	50		Menyimpan tempat kerja
namatempat	varchar	40		Menyimpan nama tempat
bidangkerja	varchar	35		Menyimpan bidang kerja
nama	varchar	40		Menyimpan nama responden

Tabel 3.2 Tabel alumni (lanjutan)

kelas	varchar	15		Menyimpan kelas responden kuesioner
bidanglainnya	varchar	40		Menyimpan bidang lainnya
tahunajaran	varchar	9		Menyimpan tahun ajaran dari responden kuesioner

### 3.7.3. Tabel berita

Digunakan untuk menyimpan data-data berita. *Field-field* yang terdapat pada tabel berita dapat dilihat pada Tabel 3.3.

Tabel 3.3. Tabel berita

Nama <i>Field</i>	Tipe Data	Ukuran	PK	Keterangan
idberita	integer		*	Menyimpan id berita
tanggal	date			Menyimpan tanggal berita
judul	varchar	100		Menyimpan judul berita
isi	text			Menyimpan isi berita
topik	varchar	40		Menyimpan topik berita

### 3.7.4. Tabel detailnilai

Digunakan untuk menyimpan data-data detail nilai. *Field-field* yang terdapat pada tabel pelanggaran dapat dilihat pada Tabel 3.4.

Tabel 3.4 Tabel detailnilai

Nama <i>Field</i>	Tipe Data	Ukuran	PK	Keterangan
idkelas	varchar	5	*	Menyimpan id kelas
tahunajar	varchar	10	*	Menyimpan tahun ajaran
idpel	varchar	5	*	Menyimpan id pelajaran
noinduk	varchar	4	*	Menyimpan nomor induk
jenis	varchar	10	*	Menyimpan jenis nilai
nourut	int		*	Menyimpan urutan nilai
nilai	float			Menyimpan nilai
tingkat	varchar	5		Menyimpan tingkat

### 3.7.5. Tabel gambarberita

Digunakan untuk menyimpan data-data gambar yang terdapat berita. *Field-field* yang terdapat pada tabel gambarberita dapat dilihat pada Tabel 3.5.

Tabel 3.5 Tabel gambarberita

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
idgambar	integer		*	Menyimpan id gambar
idberita	integer			Menyimpan id berita
gambar1	varchar	200		Menyimpan <i>link</i> gambar pertama
gambar2	varchar	200		Menyimpan <i>link</i> gambar kedua

### 3.7.6. Tabel guru

Digunakan untuk menyimpan data-data guru. *Field-field* yang terdapat pada tabel guru dapat dilihat pada Tabel 3.6.

Tabel 3.6. Tabel guru

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
idguru	integer		*	Menyimpan id guru
nama	varchar	60		Menyimpan nama guru
nuptk	varchar	25		Menyimpan nuptk
tempatlahir	varchar	40		Menyimpan tempat lahir
tanggallahir	date			Menyimpan tanggal lahir
status	varchar	10		Menyimpan status dari guru
pendidikan	varchar	5		Menyimpan data pendidikan guru
pelajaran	varchar	40		Menyimpan info pelajaran
masakerja	varchar	15		Menyimpan masa kerja
jabatan	varchar	25		Menyimpan jabatan
colorcode	varchar	7		Menyimpan <i>colorcode</i>

### 3.7.7. Tabel jampelajaran

Digunakan untuk menyimpan data-data jam pelajaran. *Field-field* yang terdapat pada tabel jampelajaran dapat dilihat pada Tabel 3.7.

Tabel 3.7. Tabel jampelajaran

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
id	integer		*	Menyimpan id
hari	varchar	10		Menyimpan nama hari
jam	integer			Menyimpan jumlah jam

### 3.7.8. Tabel kelas

Digunakan untuk menyimpan data-data kelas. *Field-field* yang terdapat pada tabel kelas dapat dilihat pada Tabel 3.8.

Tabel 3.8. Tabel kelas

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
idkelas	varchar	5	*	Menyimpan id kelas
namakelas	varchar	15		Menyimpan nama kelas
lokasi	varchar	15		Menyimpan lokasi kelas
tingkat	varchar	5		Menyimpan tingkat

### 3.7.9. Tabel kelastahun

Digunakan untuk menyimpan data-data kelas tiap tahun ajaran. *Field-field* yang terdapat pada tabel kelastahun dapat dilihat pada Tabel 3.9.

Tabel 3.9. Tabel kelastahun

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
idkelas	varchar	5	*	Menyimpan id kelas
tahunajar	varchar	10	*	Menyimpan tahun ajaran
walikelas	integer			Menyimpan walikelas

### 3.7.10. Tabel login

Digunakan untuk menyimpan data-data *login*. *Field-field* yang terdapat pada tabel *login* dapat dilihat pada Tabel 3.10.

Tabel 3.10. Tabel *login*

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
username	varchar	15	*	Menyimpan <i>username</i>
password	varchar	30		Menyimpan <i>password</i>
status	varchar	20		Menyimpan status
id	varchar	5		Menyimpan id

### 3.7.11. Tabel nilai

Digunakan untuk menyimpan data-data nilai. *Field-field* yang terdapat pada tabel nilai dapat dilihat pada Tabel 3.11.

Tabel 3.11. Tabel nilai

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
idkelas	varchar	5	*	Menyimpan id kelas
tahunajaran	varchar	15	*	Menyimpan tahun ajaran
idpel	varchar	15	*	Menyimpan id pelajaran
noinduk	varchar	4	*	Menyimpan no induk
harian	float			Menyimpan nilai harian
tugas	float			Menyimpan nilai tugas
ulanganumum	float			Menyimpan nilai ulangan umum
nilaiakhir	float			Menyimpan nilai akhir
praktik	float			Menyimpan nilai praktik
sikap	varchar	1		Menyimpan nilai sikap
tingkat	varchar	5		Menyimpan tingkat

### 3.7.12. Tabel orangtua

Digunakan untuk menyimpan data-data orang tua. *Field-field* yang terdapat pada tabel orangtua dapat dilihat pada Tabel 3.12.

Tabel 3.12. Tabel orangtua

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
idortu	integer		*	Menyimpan id orang tua
namaayah	varchar	60		Menyimpan nama ayah
namaibu	varchar	60		Menyimpan nama ibu
namawali	varchar	60		Menyimpan nama wali
alamatortu	varchar	70		Menyimpan alamat orang tua
pekerjaanayah	varchar	30		Menyimpan pekerjaan ayah
pekerjaanibu	varchar	30		Menyimpan pekerjaan ibu
pekerjaanwali	varchar	30		Menyimpan pekerjaan wali
teleponortu	varchar	30		Menyimpan nomer telepon orang tua
alamatwali	varchar	70		Menyimpan alamat wali
teleponwali	varchar	30		Menyimpan telepon wali

### 3.7.13. Tabel pelajaran

Digunakan untuk menyimpan data-data pelajaran. *Field-field* yang ada pada tabel pelajaran dapat dilihat pada Tabel 3.13.

Tabel 3.13. Tabel pelajaran

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
idpel	integer	5	*	Menyimpan id pelajaran
namapel	varchar	20		Menyimpan nama pelajaran
tingkat	varchar	5	*	Menyimpan tingkat

### 3.7.14. Tabel pelajaranguru

Digunakan untuk menyimpan data-data pelajaran guru *Field-field* yang ada pada tabel pelajaran guru dapat dilihat pada Tabel 3.14.

Tabel 3.14 Tabel pelajaranguru

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
idguru	integer		*	Menyimpan id guru
idpel	varchar	5	*	Menyimpan id pelajaran
jam	integer			
tingkat	varchar	5	*	Menyimpan tingkat

### 3.7.15. Tabel pelajarakelas

Digunakan untuk menyimpan data-data pelajaran kelas. *Field-field* yang ada pada tabel pelajarakelas dapat dilihat pada Tabel 3.15.

Tabel 3.15. Tabel pelajarakelas

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
idkelas	varchar	5	*	Menyimpan id kelas
tahunajar	varchar	10	*	Menyimpan tahun ajaran
idpel	varchar	5		Menyimpan id pelajaran
tingkat	varchar	5		Menyimpan tingkat
idguru	integer			Menyimpan id guru
hari	varchar	10	*	Menyimpan hari
jam	integer		*	Menyimpan jam

### 3.7.16. Tabel pendaftaran

Digunakan untuk menyimpan data-data pendaftaran. *Field-field* yang ada pada tabel pendaftaran dapat dilihat pada Tabel 3.16.

Tabel 3.16 Tabel pendaftaran

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
noform	integer		*	Menyimpan nomer form
nama	varchar	60		Menyimpan nama
jeniskelamin	varchar	10		Menyimpan jenis kelamin
tempatlahir	varchar	40		Menyimpan tempat lahir
tanggallahir	date			Menyimpan tanggal lahir
agama	varchar	10		Menyimpan agama

Tabel 3.16 Tabel pendaftaran (lanjutan)

alamat	varchar	70		Menyimpan alamat
asalsekolah	varchar	50		Menyimpan asal sekolah
tahunajaran	varchar	10		Menyimpah tahun ajar
status	varchar	15		Menyimpan status
kuota	integer			Menyimpan kuota
nilai	float			Menyimpan nilai
telepon	varchar	30		Menyimpan telepon
tanggaldaftar	date			Menyimpan tanggal daftar

### 3.7.17. Tabel *point*

Digunakan untuk menyimpan data-data *point* yang ada. *Field-field* yang terdapat pada tabel *point* dapat dilihat pada Tabel 3.17.

Tabel 3.17. Tabel *point*

Nama <i>Field</i>	Tipe Data	Ukuran	PK	Keterangan
id	varchar	5	*	Menyimpan id pelanggaran
jenis	varchar	150		Menyimpan jenis pelanggaran
point	integer			Menyimpan nilai pelanggaran

### 3.7.18. Tabel *siswa*

Digunakan untuk menyimpan data-data siswa yang ada. *Field-field* yang terdapat pada tabel *siswa* dapat dilihat pada Tabel 3.18.

Tabel 3.18. Tabel *siswa*

Nama <i>Field</i>	Tipe Data	Ukuran	PK	Keterangan
noinduk	varchar	4		Menyimpan no induk
nama	varchar	60		Menyimpan nama siswa
jeniskelamin	varchar	10		Menyimpanan jenis kelamin
tempatlahir	varchar	40		Menyimpan tempat lahir
tanggallahir	date			Menyimpan tanggal lahir
agama	varchar	10		Menyimpan agama

Tabel 3.18 Tabel *siswa* (lanjutan)

status	varchar	10		Menyimpan status siswa dalam keluarga
urutan	integer			Menyimpan urutan siswa sebagai anak
alamat	varchar	70		Menyimpan alamat
telepon	varchar	30		Menyimpan telepon siswa
kelasterima	varchar	3		Menyimpan kelas ketika siswa diterima
tanggalterima	date			Menyimpan tanggal penerimaan siswa
semesterterima	varchar	10		Menyimpan semester penerimaan siswa
asasekolah	varchar	50		Menyimpan asal sekolah siswa
alamatsekolah	varchar	70		Menyimpan alamat sekolah siswa sebelumnya
tahunsttb	int			Menyimpan tahun sttb
nosttb	varchar	25		Menyimpan nomer sttb
tingkat	varchar	3		Menyimpan tingkat
idortu	integer			Menyimpan id ortu

### 3.7.19. Tabel siswakelas

Digunakan untuk menyimpan data-data siswa kelas yang ada. *Field-field* yang ada pada tabel siswa kelas dapat dilihat pada Tabel 3.19.

Tabel 3.19. Tabel siswakelas

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
-------------------	------------------	---------------	-----------	-------------------

idkelas	varchar	5	*	Menyimpan id kelas
tahunajar	varchar	10	*	Menyimpan tahun ajar
noinduk	varchar	4	*	Menyimpan nomer induk
total	float			Menyimpan total
rangking	integer			Menyimpan ranking
status	varchar	10		Menyimpan status

### 3.7.20. Tabel siswapoint

Digunakan untuk menyimpan data-data siswapoint yang ada. *Field-field* yang ada pada tabel siswapoint dapat dilihat pada Tabel 3.20.

Tabel 3.20. Tabel siswapoint

Nama <i>Field</i>	Tipe Data	Ukuran	PK	Keterangan
idpelanggaran	integer		*	Menyimpan id pelanggaran
idpoint	varchar	4		Menyimpan id point
noinduk	varchar	4		Menyimpan nomer induk
tanggal	date			Menyimpan tanggal

### 3.7.21. Tabel tahunajaran

Digunakan untuk menyimpan data-data tahun ajaran. *Field-field* yang ada pada tabel tahunajaran dapat dilihat pada Tabel 3.21.

Tabel 3.21. Tabel tahunajaran

Nama <i>Field</i>	Tipe Data	Ukuran	PK	Keterangan
tahunajar	varchar	9	*	Menyimpan tahun ajaran
status	varchar	5		Menyimpan status tahun ajaran
statusjadwal	varchar	5		Menyimpan status jadwal pelajaran

### 3.7.22. Tabel ujian

Digunakan untuk menyimpan data-data ujian yang ada. *Field-field* yang ada pada tabel ujian dapat dilihat pada Tabel 3.22.

Tabel 3.22. Tabel ujian

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran</b>	<b>PK</b>	<b>Keterangan</b>
idujian	integer		*	Menyimpan id ujian
idpel	varchar	5		Menyimpan id pelajaran
idkelas	varchar	5		Menyimpan id kelas
tanggal	date			Menyimpan tanggal
ruangan	varchar	30		Menyimpan ruangan

### 3.7.23. Relasi Tabel

Tabel-tabel yang terdapat pada bagian desain tabel dapat memiliki relasi antar satu tabel dengan tabel yang lain. Relasi antar tiap tabel tersebut dapat dilihat pada Tabel 3.23.

Tabel 3.23 Relasi Tabel

No	<i>Entity 1</i>	<i>Entity 2</i>	Relasi
1	Orang tua	Siswa	<i>One to many</i>
2	Guru	Kelas Tahun	<i>One to many</i>
3	Guru	Pelajaran Guru	<i>One to many</i>
4	Pelajaran Guru	Pelajaran	<i>Many to one</i>
5	Pelajaran Guru	Pelajaran Kelas	<i>One to many</i>
6	Pelajaran	Ujian	<i>One to many</i>
7	Ujian	Kelas	<i>Many to one</i>
8	Pelajaran Kelas	Kelas Tahun	<i>One to many</i>
9	Pelajaran	Nilai	<i>One to many</i>
10	Kelas	Kelas Tahun	<i>One to many</i>
11	Nilai	Detail Nilai	<i>One to many</i>
12	Nilai	Siswa Kelas	<i>Many to one</i>
13	Point	Siswa Point	<i>One to many</i>
14	Siswa	Siswa Point	<i>One to many</i>

Tabel 3.23 Relasi Tabel (lanjutan)

15	Kelas Tahun	Tahun Ajaran	<i>One to many</i>
----	-------------	--------------	--------------------

16	Kelas Tahun	Siswa Kelas	<i>One to many</i>
17	Siswa	Siswa Kelas	<i>One to many</i>
18	Absensi Kelas	Siswa Kelas	<i>Many to one</i>
19	Kuesioner	Tahun Ajaran	<i>Many to one</i>
20	Pendaftaran	Tahun Ajaran	<i>Mamy to one</i>
21	Berita	Gambar Berita	<i>One to many</i>

### **3.8. Desain Menu**

#### **3.8.1. Menu *User Umum***

*User umum* ialah *user* yang belum melakukan *login*. Menu *user umum* terdiri dari:

- *Home*
- *News*
- *About*
- *Login*

#### **3.8.2. Menu *User Siswa atau Orang Tua***

*User siswa atau orang tua* dapat melakukan view pada berbagai data siswa, antara lain nilai, absensi, pelanggaran, jadwal, dan biodata. Menu *user siswa atau orang tua* terdiri dari:

- *Home*
- *News*
- Nilai
- Absensi
- Pelanggaran
- Jadwal
- Biodata
- Forum

#### **3.8.3. Menu *User Admin***

*User* admin dapat mengatur administrasi data akademik, antara lain proses pendaftaran, tahun ajaran, siswa, orang tua, guru, pelajaran, pelanggaran siswa, dan ujian. Menu *user* admin terdiri dari:

- *Home*
- Pendaftaran
  - Pendaftaran
  - *Check* Status Pendaftaran
  - Proses Pendaftaran
  - Registrasi Ulang
- Kelas
  - Tahun Ajaran
  - Kelas
  - Akademik
- Siswa
- Guru
- Orang Tua
- Kuesioner
  - Kuesioner
  - Hasil Kuesioner
- Pelajaran
  - Pelajaran
  - Pelajaran Guru
- Pelanggaran
  - Pelanggaran
  - Pelanggaran Siswa
- Ujian
- *News*
  - *News*
  - *List News*
- Forum
- Laporan
  - Laporan Nilai dan Jadwal

- Laporan Penerimaan
- Laporan Pelanggaran

#### **3.8.4. Menu *User Kepala Sekolah***

*User* kepala sekolah dapat melihat laporan berbagai data, antara lain nilai, jadwal, pelanggaran, dan penerimaan. Menu *user* kepala sekolah terdiri dari:

- *Home*
- *News*
- Forum
- Laporan
  - Laporan Nilai dan Jadwal
  - Laporan Penerimaan
  - Laporan Pelanggaran

#### **3.8.5. Menu *User Guru***

*User* guru dapat mengakses jadwal pelajaran, *news*, forum, dan nilai. Menu *user* guru terdiri dari:

- *Home*
- *News*
- Forum
- Nilai
- Jadwal

### **3.9. Desain *Interface***

Desain aplikasi untuk *user* umum dapat dilihat pada Gambar 3.13. Pada bagian kiri atas terdapat logo dan nama sekolah. Di bawah logo dan nama sekolah terdapat berbagai menu yang dapat diakses oleh *user*. Menu tersebut terdiri dari *home*, *news*, *about*, dan *login*. Pada bagian tengah tampilan terdapat *list news* mengenai sekolah.

Logo Nama

Sekolah			
<i>Home</i>	<i>News</i>	<i>About</i>	<i>Login</i>
<i>List News</i>			

Gambar 3.13 Desain Umum

Desain aplikasi untuk *user* orang tua dapat dilihat pada Gambar 3.14. Perbedaan yang terdapat pada desain *user* orang tua ialah menu yang dapat diakses. Menu tersebut terdiri dari *home*, *news*, nilai, absensi, pelanggaran, jadwal, biodata, dan forum.

Logo Nama Sekolah				
<i>Home</i>	<i>News</i>	<i>Nilai</i>	<i>Absensi</i>	<i>Pelanggaran</i>
Jadwal	Biodata	Forum		
<i>List News</i>				

Gambar 3.14 Desain Orang Tua

Desain aplikasi untuk *user* admin dapat dilihat pada Gambar 3.15. Perbedaan yang terdapat ialah menu yang lebih lengkap. Menu tersebut meliputi semua administrasi data akademik yang terdiri dari *home*, pendaftaran, kelas, siswa, guru, orangtua, kuesioner, pelajaran, pelanggaran, ujian, *news*, forum, dan laporan.

Logo Nama Sekolah			
<i>Home</i>	<i>Pendaftaran</i>	<i>Kelas</i>	<i>Siswa</i>
<i>Guru</i>	<i>Orang tua</i>	<i>Kuesioner</i>	<i>Pelajaran</i>
<i>Pelanggaran</i>	<i>Ujian</i>	<i>News</i>	<i>Forum</i>
<i>Laporan</i>			
<i>List News</i>			

Gambar 3.15 Desain Admin

Desain aplikasi untuk *user* kepala sekolah dapat dilihat pada Gambar 3.16. *User* kepala sekolah dapat melihat berbagai laporan akademik. Menu yang dapat diakses terdiri dari *home*, *news*, forum, dan laporan.

Logo Nama Sekolah			
<i>Home</i>	<i>News</i>	<i>Forum</i>	<i>Laporan</i>
List News			

Gambar 3.16 Desain Kepala Sekolah

Desain aplikasi untuk *user* guru dapat dilihat pada Gambar 3.17. *User* guru dapat melihat jadwal dan berbagai berita tentang sekolah. *User* guru juga dapat membuka forum dan mengakses penilaian. Menu tersebut terdiri dari *home*, *news*, forum, nilai, dan jadwal.

Logo Nama Sekolah				
<i>Home</i>	<i>News</i>	<i>Forum</i>	<i>Nilai</i>	<i>Jadwal</i>
List News				

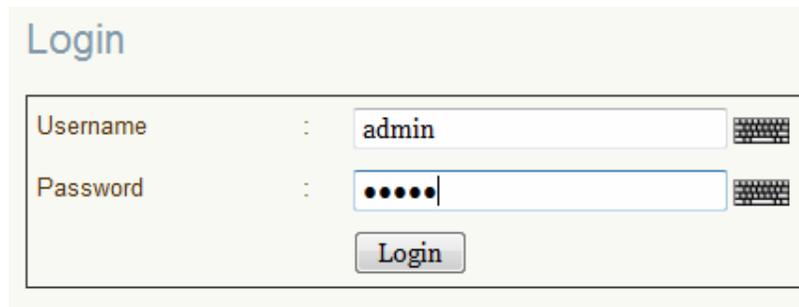
Gambar 3.17 Desain Guru

## 4. PENGUJIAN SISTEM

Pada bab ini dilakukan pengujian sistem dan pengoperasian dari fitur aplikasi yang telah dibuat. Hal ini dilakukan untuk mengetahui fitur aplikasi dapat berjalan dengan baik dan memperbaiki kesalahan yang ditemukan.

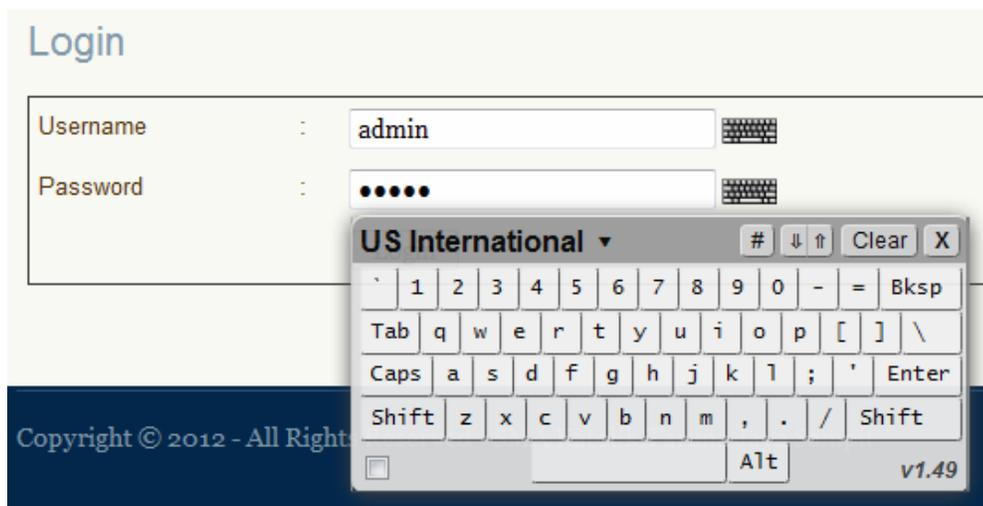
### 4.1. Login

*Login* merupakan proses yang dilakukan pertama kali untuk menggunakan fasilitas aplikasi sesuai dengan tipe *user*. *User* memasukkan *username* dan *password* pada *form login*. *Form login* dapat dilihat pada Gambar 4.1.

The image shows a web form titled "Login". It contains two input fields: "Username" with the text "admin" and "Password" with five dots. Below the fields is a "Login" button. Small keyboard icons are visible to the right of each input field.

Gambar 4.1. *Form Login*

Pada halaman *login* terdapat *keyboard virtual* untuk *input username* dan *password*. Halaman *login* dapat dilihat pada Gambar 4.2.



Gambar 4.2. *Keyboard Virtual*

Pada proses *login*, jika *username* atau *password* yang dimasukkan kosong maka akan muncul *warning*. *Warning* ini memberitahukan bahwa *username* atau *password* harus diisi. Hal ini dapat dilihat pada Gambar 4.3.

The image shows a web form titled "Login" with a light green background. At the top, there is a red warning message that reads "Username harus diisi" and "Password harus diisi". Below the warning, there are two input fields: "Username" and "Password", each followed by a colon and a small keyboard icon. A "Login" button is positioned below the "Password" field.

Gambar 4.3. *Warning Login Kosong*

Apabila *username* atau *password* yang dimasukkan tidak sesuai, maka muncul *warning* yang memberitahukan bahwa *username* atau *password* yang dimasukkan salah. Hal ini dapat dilihat pada Gambar 4.4.

The image shows a web form titled "Login" with a light green background. At the top, there is a red warning message that reads "Username atau password yang dimasukkan salah". Below the warning, there are two input fields: "Username" and "Password", each followed by a colon and a small keyboard icon. A "Login" button is positioned below the "Password" field.

Gambar 4.4. *Warning Login Salah*

## 4.2. Tahun Ajaran

Halaman Tahun Ajaran digunakan untuk mengatur data tahun ajaran. Pada halaman ini dapat dilakukan penambahan data tahun ajaran, penjadwalan, *delete* data tahun ajaran, *lock* jadwal pelajaran, dan *lock* tahun ajaran. Halaman tahun ajaran dapat dilihat pada Gambar 4.5.

Tahun Ajaran

[Add Tahun Ajaran](#)

Tahun Ajaran	Status Tahun Ajaran	Status Lock Jadwal Pelajaran	
2010/2011	aktif	aktif	<a href="#">Penjadwalan</a> <a href="#">Delete</a> <a href="#">Lock Jadwal</a>
2011/2012	aktif	aktif	<a href="#">Penjadwalan</a> <a href="#">Delete</a> <a href="#">Lock Jadwal</a>

Gambar 4.5. Halaman Tahun Ajaran

Tahun ajaran yang akan digunakan, bila belum terdapat di *database*, dapat ditambahkan pada proses *add* tahun ajaran. Halaman *add* tahun ajaran dapat dilihat pada Gambar 5.6.

Add Tahun Ajaran

\* wajib diisi

Tahun Ajaran \* :

Gambar 4.6. Halaman Add Tahun Ajaran

Apabila *input* data tahun ajaran kosong, maka muncul *warning* yang menunjukkan bahwa tahun ajaran tidak boleh kosong. Hal ini dapat dilihat pada Gambar 5.7.

Add Tahun Ajaran

\* wajib diisi

Tahun Ajaran \* :  Format tahun  
Tahun ajaran tidak boleh kosong

Gambar 4.7. Warning Tahun Ajaran Kosong

Jika *input* data tahun ajaran telah terdapat dalam *database*, maka muncul *warning* yang menunjukkan bahwa tahun ajaran sudah terdapat sebelumnya di dalam *database*. Hal ini dapat dilihat pada Gambar 4.8.

### Add Tahun Ajaran

\* wajib diisi

Tahun Ajaran \* :  Format tahun ajaran dip  
Tahun ajaran sudah ada di dalam database

Gambar 4.8 Warning Tahun Ajaran Telah Ada

Hasil tahun ajaran yang telah dimasukkan ke dalam *database* akan tampil pada halaman tahun ajaran. Hasil penambahan data ini dapat dilihat pada Gambar 4.9.

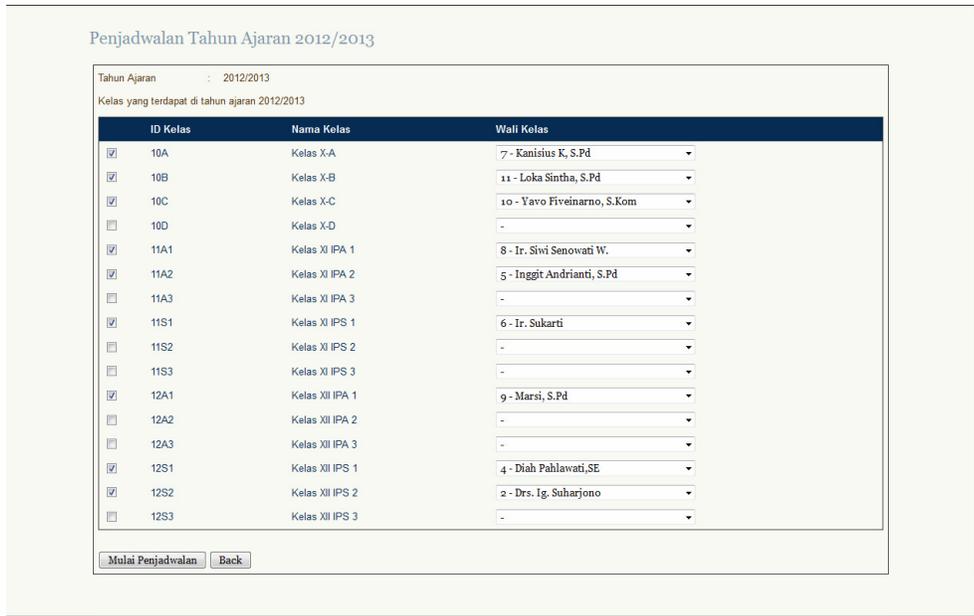
Tahun Ajaran

[+ Add Tahun Ajaran](#)

Tahun Ajaran	Status Tahun Ajaran	Status Lock Jadwal Pelajaran	
2010/2011	aktif	aktif	<input type="button" value="Penjadwalan"/> <input type="button" value="Delete"/> <input type="button" value="Lock Jadwal"/>
2011/2012	aktif	aktif	<input type="button" value="Penjadwalan"/> <input type="button" value="Delete"/> <input type="button" value="Lock Jadwal"/>
2012/2013	aktif	aktif	<input type="button" value="Penjadwalan"/> <input type="button" value="Delete"/> <input type="button" value="Lock Jadwal"/>

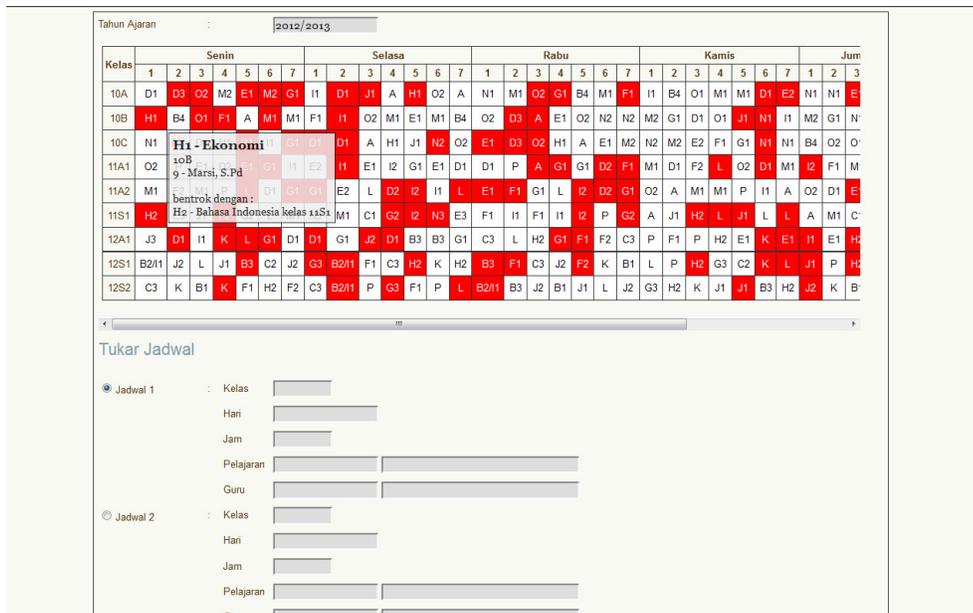
Gambar 4.9. Hasil Add Tahun Ajaran

Setelah memasukkan tahun ajaran, dapat dilakukan pengaturan kelas-kelas yang terdapat pada tahun ajaran tersebut. Hal ini dilakukan dengan memberikan tanda centang pada *check box* yang terdapat di sisi kiri. Pengaturan kelas ini juga termasuk dalam penentuan wali kelasnya. Wali kelas ini dipilih melalui *combo box* yang tersedia. *Combo box* ini berisi *list* nama guru yang memiliki status sebagai guru tetap yayasan. Hal ini dapat dilihat pada Gambar 4.10.



Gambar 4.10. Halaman Penjadwalan

Setelah memasukkan pengaturan kelas pada tahun ajaran beserta wali kelasnya, maka dilakukan proses penjadwalan akademik. Proses penjadwalan akademik ini dilakukan secara *random*. Apabila terjadi bentrok pada hasil penjadwalan akademik maka akan diberikan warna merah dan keterangan jadwal yang saling bentrok. Hal ini dapat dilihat pada Gambar 4.11.



Gambar 4.11. Penjadwalan Akademik

Untuk mengatasi bentrokan yang terjadi pada jadwal pelajaran digunakan fitur tukar jadwal. Fitur tukar jadwal ini menukar posisi antara satu jadwal dengan jadwal yang lainnya. Pada Gambar 4.11. dapat dilihat jadwal Hari Senin jam pertama pada kelas 10B mata pelajaran Ekonomi, yang memiliki kode H1. Mata pelajaran ini berbentrok dengan jadwal mata pelajaran Bahasa Indonesia yang memiliki kode H2 pada kelas 11S1.

Pada pengujian fitur tukar jadwal, dilakukan pertukaran mata pelajaran Ekonomi yang memiliki kode H1 dengan jadwal mata pelajaran Biologi, yang memiliki kode E1, pada hari Senin jam kelima di kelas 10A. Hal ini dapat dilihat pada Gambar 4.12.

Kelas	Senin							Selasa							Rabu							Kamis							Jum		
	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3
10A	D1	D3	O2	M2	E1	M2	G1	I1	D1	J1	A	H1	O2	A	N1	M1	O2	G1	B4	M1	F1	I1	B4	O1	M1	M1	D1	E2	N1	N1	E
10B	H1	B4	O1	F1	A	E1	E1	O2	M1	E1	M1	B4	O2	D3	A	E1	O2	N2	N2	M2	G1	D1	O1	J1	N1	I1	M2	G1	N		
10C	N1	M1	D1	N1	B4	E1	E1	O2	M1	E1	M1	B4	O2	D3	A	E1	O2	N2	N2	M2	G1	D1	O1	J1	N1	I1	M2	G1	N		
11A1	O2	P	E1	D2	E1	E1	E1	O2	M1	E1	M1	B4	O2	D3	A	E1	O2	N2	N2	M2	G1	D1	O1	J1	N1	I1	M2	G1	N		
11A2	M1	F2	M1	P	L	L	L	O2	M1	E1	M1	B4	O2	D3	A	E1	O2	N2	N2	M2	G1	D1	O1	J1	N1	I1	M2	G1	N		
11S1	H2	E3	J1	F2	G2	L	M1	N3	M1	C1	G2	I2	N3	E3	F1	I1	I1	I2	P	G2	A	J1	H2	L	J1	L	A	M1	C		
12A1	J3	D1	I1	K	L	G1	D1	D1	G1	J2	D1	B3	B3	G1	C3	L	H2	G1	F1	F2	C3	P	F1	P	H2	E1	K	E1	I1	E1	H
12S1	B2/1	J2	L	J1	B3	C2	J2	G3	B2/1	F1	C3	H2	K	H2	B3	F1	C3	J2	F2	K	B1	L	P	H2	G3	C2	K	L	J1	P	H
12S2	C3	K	B1	K	F1	H2	F2	C3	B2/1	P	G3	F1	P	L	B2/1	B3	J2	B1	J1	L	J2	G3	H2	K	J1	J1	B3	H2	J2	K	B

Tukar Jadwal

Jadwal 1 : Kelas

Hari

Jam

Pelajaran

Guru

Jadwal 2 : Kelas

Hari

Jam

Pelajaran

Guru

Gambar 4.12. Pertukaran Jadwal

Setelah dilakukan pertukaran jadwal antara mata pelajaran Ekonomi yang memiliki kode H1 dengan mata pelajaran Biologi yang memiliki kode E1, maka dapat terlihat warna putih pada jadwal pelajaran Hari Senin jam pertama. Hal ini berarti sudah tidak terjadi bentrok jadwal pada jam tersebut. Proses ini dapat dilihat pada Gambar 4.13.

Tahun Ajaran : 2012/2013

Kelas	Senin							Selasa							Rabu							Kamis							Jum		
	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3
10A	D1	D3	O2	M2	H1	M2	G1	I1	D1	J1	A	H1	O2	A	N1	M1	O2	G1	B4	M1	F1	I1	B4	O1	M1	M1	D1	E2	N1	N1	E
10B	E1	B4	O1	F1	A	M1	M1	F1	I1	O2	M1	E1	M1	B4	O2	D3	A	E1	O2	N2	N2	M2	G1	D1	O1	J1	N1	I1	M2	G1	N
10C	N1	M1	D1	N1	B4	I1	G1	D1	D1	A	H1	J1	N2	O2	E1	O2	H1	A	E1	M2	N2	M2	E2	F1	G1	N1	N1	B4	O2	O	
11A1	O2	P	E1	D2	E1	G1	I1	E2	I1	E1	I2	G1	E1	D1	D1	P	A	G1	G1	D2	F1	M1	D1	F2	L	O2	D1	M1	I2	F1	M
11A2	M1	F2	M1	P	L	D1	G1	G1	E2	L	D2	I2	I1	L	E1	F1	G1	L	I2	D2	G1	O2	A	M1	M1	P	I1	A	O2	D1	E
11S1	H2	E3	J1	F2	G2	L	M1	N3	M1	C1	G2	I2	N3	E3	F1	I1	F1	I1	I2	P	G2	A	J1	H2	L	J1	L	L	A	M1	C
12A1	J3	D1	I1	K	L	G1	D1	D1	G1	J2	D1	B3	B3	G1	C3	L	H2	G1	F1	F2	C3	P	F1	P	H2	E1	K	E1	I1	E1	H
12S1	B2/I1	J2	L	J1	B3	C2	J2	G3	B2/I1	F1	C3	H2	K	H2	B3	F1	C3	J2	F2	K	B1	L	P	H2	G3	C2	K	L	J1	P	H
12S2	C3	K	B1	K	F1	H2	F2	C3	B2/I1	P	G3	F1	P	L	B2/I1	B3	J2	B1	J1	L	J2	G3	H2	K	J1	J1	B3	H2	J2	K	B

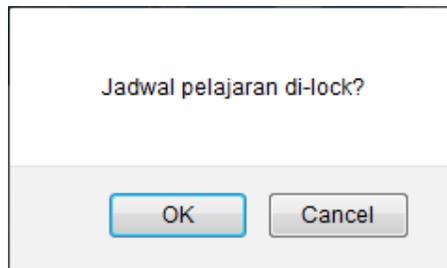
Tukar Jadwal

Jadwal 1 : Kelas   
 Hari   
 Jam   
 Pelajaran   
 Guru

Jadwal 2 : Kelas   
 Hari   
 Jam   
 Pelajaran   
 Guru

Gambar 4.13. Hasil Pertukaran Jadwal

Fitur *lock* jadwal pelajaran digunakan sebagai verifikasi dari suatu jadwal pelajaran, sehingga data jadwal pelajaran tidak bisa di-*edit* lagi. Ketika *user* akan melakukan *lock* jadwal pelajaran, terlebih dahulu *user* akan diberikan konfirmasi ulang bahwa jadwal pelajaran akan di-*lock*. Hal ini dapat dilihat pada Gambar 4.14.

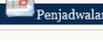


Gambar 4.14 Konfirmasi *Lock* Jadwal Pelajaran

Jika jadwal pelajaran telah di-*lock*, maka jadwal pelajaran tidak bisa di-*generate* kembali dan statusnya pun berubah menjadi *lock*. Hal ini dapat dilihat pada Gambar 4.15.

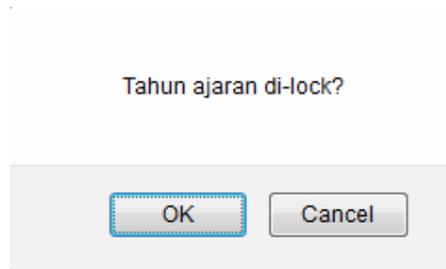
Tahun Ajaran



Tahun Ajaran	Status Tahun Ajaran	Status Lock Jadwal Pelajaran	
2010/2011	aktif	lock	
2011/2012	aktif	aktif	  
2012/2013	aktif	aktif	  

Gambar 4.15 Perubahan Status *Lock* Jadwal Pelajaran

Fitur *lock* tahun ajaran digunakan untuk verifikasi dari suatu tahun ajaran, yang menyatakan bahwa tahun ajaran tersebut telah selesai. Fitur *lock* tahun ajaran muncul setelah melakukan *lock* jadwal pelajaran terlebih dahulu. Ketika *user* akan melakukan *lock* tahun ajaran, terlebih dahulu *user* akan diberikan konfirmasi ulang bahwa tahun ajaran akan di-*lock*. Hal ini dapat dilihat pada Gambar 4.16.



Gambar 4.16. Konfirmasi *Lock* Tahun Ajaran

Setelah verifikasi, tahun ajaran tersebut hanya dapat di-*view* oleh *user*. Fitur *delete* tahun ajaran tidak bisa digunakan setelah status tahun ajaran berubah menjadi *lock*. Hal ini dapat dilihat pada Gambar 4.17.

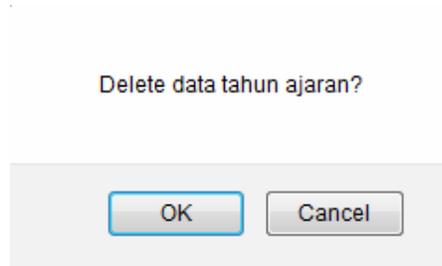
Tahun Ajaran



Tahun Ajaran	Status Tahun Ajaran	Status Lock Jadwal Pelajaran	
2010/2011	lock	lock	
2011/2012	aktif	aktif	  
2012/2013	aktif	aktif	  

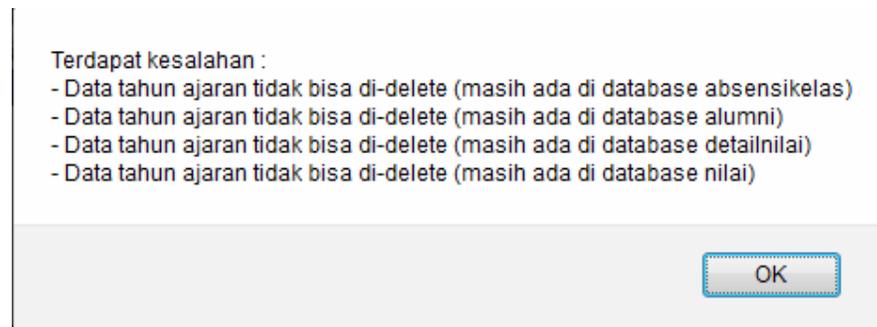
Gambar 4.17. Status Tahun Ajaran *Lock*

Fitur *delete* tahun ajaran digunakan untuk menghapus tahun ajaran yang statusnya masih aktif. Sebelum melakukan *delete* tahun ajaran, *user* akan diberikan konfirmasi ulang. Hal ini dapat dilihat pada Gambar 4.18.



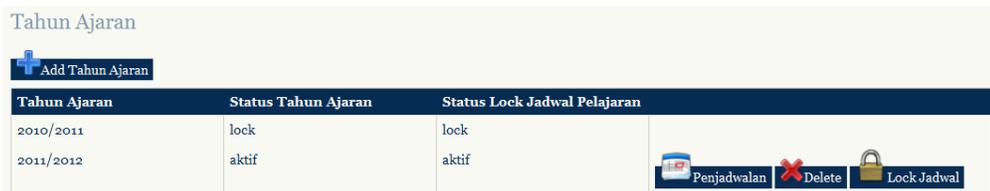
Gambar 4.18. *Delete* Tahun Ajaran

Namun, tahun ajaran tidak bisa di *delete* jika menjadi *foreign key* di dalam *table* yang lain pada *database*. Hal ini dapat dilihat pada Gambar 4.18.



Gambar 4.19. *Warning Delete* Tahun Ajaran

Apabila tahun ajaran tidak menjadi *foreign key* di *table* lain, *delete* tahun ajaran dapat dilakukan. Pengujian *delete* dilakukan pada tahun ajaran 2012/2013 yang baru ditambahkan ke dalam *database*. Hasil *delete* tahun ajaran dapat dilihat pada Gambar 4.20.



Tahun Ajaran

[Add Tahun Ajaran](#)

Tahun Ajaran	Status Tahun Ajaran	Status Lock Jadwal Pelajaran
2010/2011	lock	lock
2011/2012	aktif	aktif

Penjadwalan Delete Lock Jadwal

Gambar 4.20. Hasil *Delete* Tahun Pelajaran

### 4.3. Add, View, Edit, dan Delete

Proses *add*, *view*, *edit*, dan *delete* yang dilakukan pada pengujian sistem ialah proses *add*, *edit*, dan *delete* di halaman pendaftaran siswa. Proses administrasi data ini (*add*, *view*, *edit*, dan *delete*) digunakan untuk mengelola data yang terdapat pada *table* pendaftaran. Halaman pendaftaran siswa dapat dilihat pada Gambar 4.21.



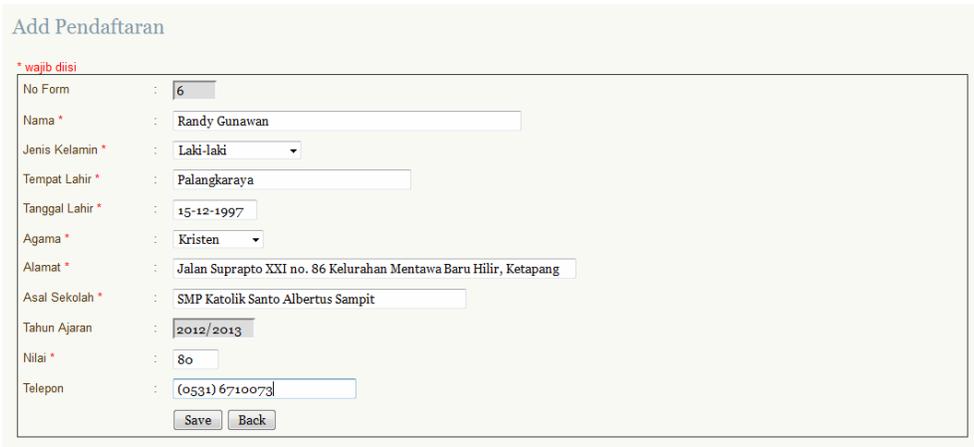
Pendaftaran

[+ Add Pendaftaran](#)

No Form	Nama	Asal Sekolah	Tahun Ajaran	Nilai	
1	Denny Setiawan	SMP Katolik Santo Albertus Sampit	2012/2013	40	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
2	Adi Cipta Nugroho	SMP Katolik St. Albertus	2012/2013	70	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
3	Stevani Susanto	SMP Katolik Santo Albertus Sampit	2012/2013	70	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
4	Prajoko	SMP Kristen Maranatha Sampit	2012/2013	65	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
5	Hindarto Laytno	SMP Katolik St. Albertus	2012/2013	78	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>

Gambar 4.21. Halaman Pendaftaran Siswa

Proses *add* yang dilakukan pada pengujian sistem ialah proses *add* pendaftaran. Proses *add* pendaftaran digunakan untuk menambahkan data pendaftaran ke dalam *table* pendaftaran di dalam *database*. Halaman *add* pendaftaran dapat dilihat pada Gambar 4.22.



Add Pendaftaran

\* wajib diisi

No Form : 6

Nama \* : Randy Gunawan

Jenis Kelamin \* : Laki-laki

Tempat Lahir \* : Palangkaraya

Tanggal Lahir \* : 15-12-1997

Agama \* : Kristen

Alamat \* : Jalan Suprpto XXI no. 86 Kelurahan Mentawa Baru Hilir, Ketapang

Asal Sekolah \* : SMP Katolik Santo Albertus Sampit

Tahun Ajaran : 2012/2013

Nilai \* : 80

Telepon : (0531) 6710073

Gambar 4.22. Halaman Add Pendaftaran

Jika ada data pendaftaran yang wajib diisi, namun data pendaftaran tersebut kosong. Maka akan diberikan warning yang menjelaskan bahwa data tersebut kosong. Hal ini dapat dilihat pada Gambar 4.23.

Add Pendaftaran

\* wajib diisi

No Form : 6

Nama \* : Randy Gunawan

Jenis Kelamin \* : Laki-Laki

Tempat Lahir \* : Palangkaraya

Tanggal Lahir \* : 15-12-1997

Agama \* : Kristen

Alamat \* : Jalan Suprpto XXI no. 86 Kelurahan Mentawa Baru Hilir, Ketapang

Asal Sekolah \* :  
Asal sekolah tidak boleh kosong

Tahun Ajaran : 2012/2013

Nilai \* : 80

Telepon : (0531) 6710073

Save Back

Gambar 4.23. Warning Add Pendaftaran

Setelah data pendaftaran dilengkapi, maka proses *add* pendaftaran siswa dapat dilakukan. Hasil penambahan data tersebut dapat dilihat pada Gambar 4.24.

Pendaftaran

[+ Add Pendaftaran](#)

No Form	Nama	Asal Sekolah	Tahun Ajaran	Nilai	
1	Denny Setiawan	SMP Katolik Santo Albertus Sampit	2012/2013	40	
2	Adi Cipta Nugroho	SMP Katolik St. Albertus Sampit	2012/2013	70	
3	Stevani Susanto	SMP Katolik Santo Albertus Sampit	2012/2013	70	
4	Prajoko	SMP Kristen Maranatha Sampit	2012/2013	65	
5	Hindarto Laytno	SMP Katolik St. Albertus Sampit	2012/2013	78	
6	Randy Gunawan	SMP Katolik Santo Albertus Sampit	2012/2013	80	

Gambar 4.24. Penambahan Data Pendaftaran

Halaman *edit* pendaftaran digunakan untuk mengubah data pendaftaran yang ada pada *database*. Halaman *edit* pendaftaran ditunjukkan pada Gambar 4.25.

Edit Pendaftaran

\* wajib diisi

No Form	:	6
Nama *	:	Randy Gunawan
Jenis Kelamin *	:	Laki-laki
Tempat Lahir *	:	Palangkaraya
Tanggal Lahir *	:	15-12-1997
Agama *	:	Kristen
Alamat *	:	Jalan Suprpto XXI no. 86 Kelurahan Mentawa Baru Hilir, Ketapang
Asal Sekolah *	:	SMP Kristen Maranatha Sampit
Tahun Ajaran	:	2012/2013
Nilai *	:	80
Telepon	:	(0531) 6710073

Save Back

Gambar 4.25 Halaman *Edit* Pendaftaran

Pada pengujian *edit* data pendaftaran, dilakukan *edit* data asal sekolah pada data pendaftaran. Hasil *edit* data pendaftaran dapat dilihat pada Gambar 4.26.

Pendaftaran

+ Add Pendaftaran

No Form	Nama	Asal Sekolah	Tahun Ajaran	Nilai	
1	Denny Setiawan	SMP Katolik Santo Albertus Sampit	2012/2013	40	View Edit Delete
2	Adi Cipta Nugroho	SMP Katolik St. Albertus Sampit	2012/2013	70	View Edit Delete
3	Stevani Susanto	SMP Katolik Santo Albertus Sampit	2012/2013	70	View Edit Delete
4	Prajoko	SMP Kristen Maranatha Sampit	2012/2013	65	View Edit Delete
5	Hindarto Laytno	SMP Katolik St. Albertus Sampit	2012/2013	78	View Edit Delete
6	Randy Gunawan	SMP Kristen Maranatha Sampit	2012/2013	80	View Edit Delete

Gambar 4.26 Hasil *Edit* Pendaftaran

Halaman *view* pendaftaran digunakan untuk melihat data pendaftaran yang ada pada *table* pendaftaran di *database*. Halaman *view* pendaftaran dapat dilihat pada Gambar 4.27.

View Pendaftaran

No Form	: 6
Nama	: Randy Gunawan
Jenis Kelamin	: Laki-laki
Tempat Lahir	: Palangkaraya
Tanggal Lahir	: 15-12-1997
Agama	: Kristen
Alamat	: Jalan Suprpto XXI no. 86 Kelurahan Mentawa Baru Hilir, Ketapang
Asal Sekolah	: SMP Kristen Maranatha Sampit
Tahun Ajaran	: 2012/2013
Nilai	: 80
Telepon	: (0531) 6710073

Gambar 4.27 Halaman View Pendaftaran

Halaman *delete* pendaftaran digunakan untuk menghapus data pendaftaran yang ada pada *database*. Sebelum melakukan *delete* data pendaftaran, *user* akan diberikan konfirmasi ulang *delete* data pendaftaran. Hal ini dapat dilihat pada Gambar 4.28.

Delete data pendaftaran?

Gambar 4.28 Konfirmasi *Delete* Pendaftaran

Pada proses *delete* pendaftaran yang dilakukan pengujian sistem, data pendaftaran dengan nomer *form* enam akan dihapus dari *database*. Hasil *delete* data pendaftaran ini dapat dilihat pada Gambar 4.29.

Pendaftaran

No Form	Nama	Asal Sekolah	Tahun Ajaran	Nilai	
1	Denny Setiawan	SMP Katolik Santo Albertus Sampit	2012/2013	40	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>
2	Adi Cipta Nugroho	SMP Katolik St. Albertus Sampit	2012/2013	70	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>
3	Stevani Susanto	SMP Katolik Santo Albertus Sampit	2012/2013	70	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>
4	Prajoko	SMP Kristen Maranatha Sampit	2012/2013	65	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>
5	Hindarto Laytno	SMP Katolik St. Albertus Sampit	2012/2013	78	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>

Gambar 4.29 Hasil *Delete* Pendaftaran

#### 4.4. Proses Pendaftaran

Pada proses pendaftaran, pertama kali dilakukan *input* data berupa jumlah kuota pendaftaran. Pada halaman proses pendaftaran, terdapat *button* pendaftaran pada sisi kiri yang terhubung dengan halaman pendaftaran (*add*, *edit*, *view*, dan *delete*). Hal ini dapat dilihat pada Gambar 4.30.



The screenshot shows a web form titled "Proses Pendaftaran". At the top left, there is a "Pendaftaran" button. Below it, the form contains two input fields: "Tahun Ajaran" with the value "2012/2013" and "Kuota" with the value "100". A "Proses Pendaftaran" button is located at the bottom right of the form area.

Gambar 4.30 Halaman Proses Pendaftaran

Apabila *input* data jumlah kuota kosong atau bukan bernilai numerik, maka akan keluar *warning* yang menyatakan bahwa kuota tidak boleh kosong atau kuota harus bernilai numerik. Hal ini dapat dilihat pada Gambar 4.31.



The screenshot shows the same "Proses Pendaftaran" form as in Gambar 4.30. However, the "Kuota" input field is empty. Below the input field, there is a red warning message that reads: "Kuota tidak boleh kosong" and "Kuota harus bernilai numerik". The "Proses Pendaftaran" button is still visible at the bottom.

Gambar 4.31 Warning Kuota

Setelah memasukkan kuota pendaftaran, misalnya berjumlah tiga, maka akan ditampilkan semua *list* data pendaftaran yang ada di dalam *database*. *List* data pendaftaran diurutkan berdasarkan nilai dan nomer *form*. Dari semua *list* pendaftaran tersebut, tiga urutan teratas dari *list* akan secara otomatis statusnya menjadi "Diterima". Status ini dapat diganti melalui *combo box* yang tersedia sesuai dengan pertimbangan dari pihak sekolah. Hal ini dapat dilihat pada Gambar 4.32.

**Pendaftaran**

Tahun Ajaran : 2012/2013

Kuota : 3

No	Nama	No Form	Asal Sekolah	Jenis Kelamin	Nilai	Status
1	Hindarto Laytno	5	SMP Katolik St. Albertus Sampit	L	78	Diterima
2	Adi Cipta Nugroho	2	SMP Katolik St. Albertus Sampit	L	70	Diterima
3	Stevani Susanto	3	SMP Katolik Santo Albertus Sampit	P	70	Diterima
4	Prajoko	4	SMP Kristen Maranatha Sampit	L	65	Tidak Diterima
5	Denny Setiawan	1	SMP Katolik Santo Albertus Sampit	L	40	Tidak Diterima

Gambar 4.32 Proses Pemilihan Pendaftaran

Sebelum menyimpan data proses pendaftaran ke dalam *database*, akan muncul dialog konfirmasi sebagai verifikasi. Apabila data pendaftaran telah benar, *list* proses pendaftaran ini kemudian disimpan ke dalam *database*. Hal ini dapat dilihat pada Gambar 4.33.

Apakah data sudah benar?

Gambar 4.33 Konfirmasi Data Proses Pendaftaran

#### 4.5. Registrasi Ulang

Halaman ini digunakan untuk proses registrasi ulang pendaftaran. Pada proses registrasi ulang, ditampilkan semua *list* data pendaftaran yang telah diterima sebelumnya pada proses pendaftaran. Pada sisi kanan *list* tersebut terdapat sebuah *check box* untuk menentukan calon siswa, yang telah diterima sebelumnya, yang telah melakukan registrasi ulang. Hal ini dapat dilihat pada Gambar 4.34.

Registrasi Ulang

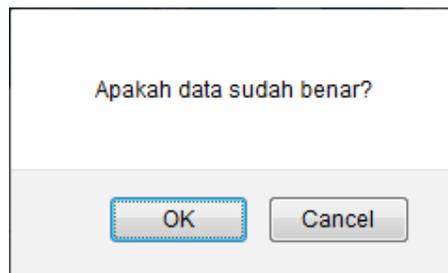
Tahun Ajaran : 2012/2013

No	Nama	No Form	Asal Sekolah	Jenis Kelamin	Nilai	Registrasi Ulang
1	Hindarto Laytno	5	SMP Katolik St. Albertus Sampit	L	78	<input checked="" type="checkbox"/>
2	Adi Cipta Nugroho	2	SMP Katolik St. Albertus Sampit	L	70	<input checked="" type="checkbox"/>
3	Stevani Susanto	3	SMP Katolik Santo Albertus Sampit	P	70	<input checked="" type="checkbox"/>

Save Hasil Registrasi Ulang

Gambar 4.34 Halaman Registrasi Ulang

Sebelum menyimpan data registrasi ulang ke dalam *database*, akan muncul dialog konfirmasi sebagai verifikasi. Hal ini dapat dilihat pada Gambar 4.35.



Gambar 5.35 Konfirmasi Data Registrasi Ulang

Apabila data registrasi ulang tersebut telah benar, maka data calon siswa tersebut akan disimpan ke dalam *database*. Data calon siswa yang telah melakukan registrasi ulang akan dimasukkan ke dalam *database* siswa. Hal ini dapat dilihat pada Gambar 4.36.

1757	Maria Yuliana Marfin	Perempuan	Desa V	28-06-1992	-	  
1762	Merry Florida	Perempuan	Sampit	22-11-1996	Jalan Sampit P.Bun Km. 26	  
1773	Pandu Kariwian Tua	Laki-laki	Mahajandau	16-06-1996	Jalan H.M Arsyad / Jalan Teratai 5 Jalur 4 No.73 R	  
1784	Rita Ria Safitri	Perempuan	Tanjung Jorong	11-02-1996	Asrama Katolik Sampit	  
1788	Rupinarika	Perempuan	Tanjung Harapan	17-08-1995	-	  
1792	Silvi Pasaribu	Perempuan	Luwuk Bunter	02-04-1997	-	  
1794	Simeon Ardi Mulyadi	Laki-laki	Sampit	18-04-1995	Jalan Merpati No.129 Perumnas Sampit	  
1801	Tommy Soegiarto	Laki-laki	Surabaya	11-04-1996	Jalan DI Panjaitan No. 22 RT. 33 RW. 02	  
1809	Yenni Tria Wulandary	Perempuan	Kristen	28-03-1996	Asrama Katolik	  
1811	Yolanda	Perempuan	Tumbang Lahang	18-05-1997	Jl. Cristopel Mihing Gg. Agus Salim No. 19	  
1812	Hindarto Laytno	Laki-laki	Surabaya	16-02-1998	Jalan Pangeran Antasari	  
1813	Adi Cipta Nugroho	Laki-laki	Surabaya	11-05-1998	Jalan Merdeka	  
1814	Stevani Susanto	Perempuan	Surabaya	06-04-1998	Jalan Jenderal Sudirman	  

Gambar 4.36 Data Siswa Baru

#### 4.6. Siswa Kelas

Halaman siswa kelas digunakan untuk menambahkan siswa ke dalam suatu kelas pada suatu tahun ajaran atau menghapus siswa dari suatu kelas pada suatu tahun ajaran. Untuk menambahkan siswa, dapat menggunakan *button add* siswa yang berada di sisi kiri atas halaman. Halaman siswa kelas dapat dilihat pada Gambar 4.37.

Daftar Siswa Kelas X-A Tahun Ajaran 2012/2013		
		
No.	No Induk	Nama Siswa
		

Gambar 4.37 Halaman Siswa Kelas

Pada pengujian sistem, di proses *add* siswa ini terdapat *list* siswa yang berada pada kelas X. Untuk memilih siswa yang dimasukkan ke dalam kelas, digunakan *checkbox* pada sisi kiri *list* siswa. Proses *add* siswa dapat dilihat pada Gambar 4.38.

No.	No Induk	Nama Siswa
<input type="checkbox"/> 1	1707	Afrida Karolina Ndoa
<input type="checkbox"/> 2	1709	Alfin Giovani
<input type="checkbox"/> 3	1717	Clara Setia Lencana
<input type="checkbox"/> 4	1725	Dominika Odok
<input type="checkbox"/> 5	1731	Euneke Leonora Giovanie Jemadut
<input type="checkbox"/> 6	1734	Feronika Pangaribuan
<input type="checkbox"/> 7	1735	Gardena
<input type="checkbox"/> 8	1740	Hervina Wulandari
<input type="checkbox"/> 9	1750	Lerina Gelin
<input type="checkbox"/> 10	1757	Maria Yuliana Marfin
<input type="checkbox"/> 11	1762	Merry Florida
<input type="checkbox"/> 12	1773	Pandu Kariwian Tua
<input type="checkbox"/> 13	1784	Rita Ria Safitri
<input type="checkbox"/> 14	1788	Rupinarika
<input type="checkbox"/> 15	1792	Silvi Pasaribu
<input type="checkbox"/> 16	1794	Simeon Ardi Mulyadi
<input type="checkbox"/> 17	1801	Tommy Soegiarto
<input type="checkbox"/> 18	1809	Yenni Tria Wulandary
<input type="checkbox"/> 19	1811	Yolanda
<input checked="" type="checkbox"/> 20	1812	Hindarto Laytno
<input checked="" type="checkbox"/> 21	1813	Adi Cipta Nugroho
<input checked="" type="checkbox"/> 22	1814	Stevani Susanto

Gambar 4.38 Proses Add Siswa

Setelah menyimpan hasil proses penambahan siswa, maka hasil *input* tersebut dapat dilihat pada halaman siswa kelas. Hal ini dapat dilihat pada Gambar 4.39.

Daftar Siswa Kelas X-A Tahun Ajaran 2012/2013

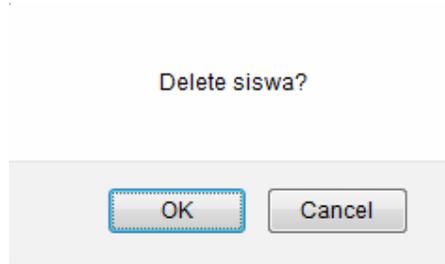
[+ Add Siswa](#)

No.	No Induk	Nama Siswa	
1	1812	Hindarto Laytno	
2	1813	Adi Cipta Nugroho	
3	1814	Stevani Susanto	

[Back](#)

Gambar 5.39 Hasil Penambahan Siswa

Halaman *delete* siswa kelas digunakan untuk menghapus data siswa dari kelas. Sebelum melakukan *delete* data siswa dari kelas, *user* akan diberikan konfirmasi ulang *delete* data siswa kelas. Hal ini dapat dilihat pada Gambar 4.40.



Gambar 4.40 Konfirmasi *Delete* Siswa

Pada proses *delete* siswa kelas yang dilakukan pengujian sistem, data siswa dengan nomer induk 1813 akan dihapus dari kelas X-A. Hasil *delete* data siswa kelas ini dapat dilihat pada Gambar 4.41.

Daftar Siswa Kelas X-A Tahun Ajaran 2012/2013

+ Add Siswa

No.	No Induk	Nama Siswa	
1	1812	Hindarto Laytno	Delete
2	1814	Stevani Susanto	Delete

Back

Gambar 4.41 Hasil *Delete* Siswa

#### 4.7. Add Absensi

Halaman *add* absensi digunakan untuk menambahkan data absensi ke dalam *database*. Halaman absensi dapat dilihat pada Gambar 4.42.

Daftar Absensi Kelas X-A Tahun Ajaran 2012/2013 Tanggal 17-09-2012

<< Tanggal : 17-09-2012 Ubah Tanggal >>

No.	No Induk	Nama Siswa	Masuk	Sakit	Ijin	Alpha
1	1812	Hindarto Laytno	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2	1814	Stevani Susanto	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Save Back

Gambar 4.42 Halaman *Add Absensi*

#### 4.8. Jadwal Pelajaran

Jadwal pelajaran yang telah di-generate sebelumnya pada penjadwalan dapat dilihat pada halaman jadwal pelajaran. Hal ini dapat dilihat pada Gambar 4.43.

The screenshot shows a lesson schedule for three days: Senin, Selasa, and Rabu. Each day's schedule is presented in a table with four columns: Jam, Pelajaran, Guru, and Penilaian. The 'Penilaian' column contains a button labeled 'Nilai'.

Senin			
Jam	Pelajaran	Guru	Penilaian
1	D1 - Kimia	5 - Inggit Andrianti, S.Pd	Nilai
2	D3 - Bimbingan Konseling	5 - Inggit Andrianti, S.Pd	Nilai
3	O2 - Bahasa Indonesia	13 - Tommy Roeska Amrosius Silalahi	Nilai
4	M2 - Muatan Lokal	12 - Marjoko	Nilai
5	H1 - Ekonomi	9 - Marsi, S.Pd	Nilai
6	M2 - Muatan Lokal	12 - Marjoko	Nilai
7	G1 - Fisika	8 - Ir. Sivi Senowati W.	Nilai

Selasa			
Jam	Pelajaran	Guru	Penilaian
1	I1 - TIKOM	10 - Yavo Fiveirarno, S.Kom	Nilai
2	D1 - Kimia	5 - Inggit Andrianti, S.Pd	Nilai
3	J1 - Geografi	4 - Diah Pahlawati, S.E	Nilai
4	A - PKn	1 - Drs.Arkedeus BT, MSM	Nilai
5	H1 - Ekonomi	9 - Marsi, S.Pd	Nilai
6	O2 - Bahasa Indonesia	13 - Tommy Roeska Amrosius Silalahi	Nilai
7	A - PKn	1 - Drs.Arkedeus BT, MSM	Nilai

Rabu			
Jam	Pelajaran	Guru	Penilaian
1	N1 - Bahasa Inggris	14 - Imelda	Nilai
2	M1 - Matematika	12 - Marjoko	Nilai

Gambar 4.43 Jadwal Pelajaran

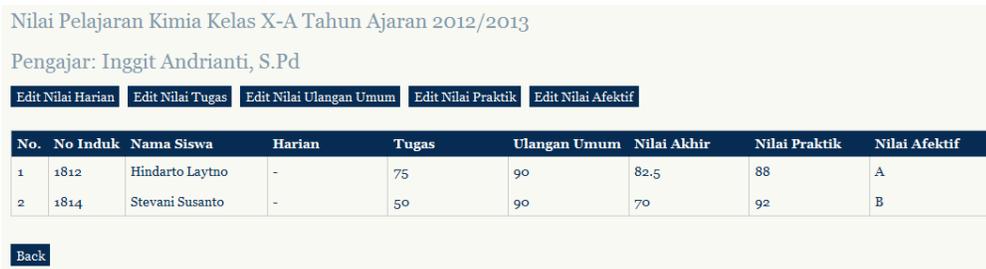
Pada fitur tukar jadwal pelajaran versi yang kedua, tiap mata pelajaran dibuat berwarna sesuai dengan *color code* guru yang telah ditentukan. Tanda jadwal yang berbentrok dapat dilihat melalui tulisan kode pelajaran yang berwarna merah dan keterangannya. Hal ini dapat dilihat pada Gambar 4.44.



Gambar 4.44 Tukar Jadwal

#### 4.9. Penilaian

Nilai murid dimasukkan ke dalam *database* melalui halaman penilaian. Hal ini dapat dilihat pada Gambar 4.45.



Gambar 4.45 Penilaian

Pada pengujian sistem, dilakukan penambahan data nilai harian. Data nilai harian ini dapat diatur jumlahnya. Untuk menambah jumlah data nilai harian dapat dilakukan melalui *button* tambah jumlah nilai harian. Untuk mengurangi jumlah data nilai harian dapat dilakukan melalui *button* kurangi jumlah nilai harian. Hal ini dapat dilihat pada Gambar 4.46.

Edit Nilai Harian Pelajaran Kimia Kelas X-A Tahun Ajaran 2012/2013  
 Pengajar: Inggit Andrianti, S.Pd

Jumlah Ulangan Harian : 0

No.	No Induk	Nama Siswa
1	1812	Hindarto Laytno
2	1814	Stevani Susanto

Gambar 4.46 Nilai Harian

Pada pengujian *input* nilai harian, terlebih dahulu jumlah nilai harian ditambahkan. Hal ini dapat dilihat pada Gambar 4.47.

Edit Nilai Harian Pelajaran Kimia Kelas X-A Tahun Ajaran 2012/2013  
 Pengajar: Inggit Andrianti, S.Pd

Jumlah Ulangan Harian : 1

No.	No Induk	Nama Siswa	1
1	1812	Hindarto Laytno	-
2	1814	Stevani Susanto	-

Gambar 4.47 Penambahan Jumlah Nilai Harian

Setelah jumlah nilai harian ini ditambahkan, maka akan terlihat kolom untuk mengisi nilai harian. Pada pengujian sistem, kolom ini akan dimasukkan *input* nilai harian pada masing-masing siswa. Hasil ini akan disimpan dengan menekan *button save* yang terdapat di sisi kiri bawah halaman. Hal ini dapat dilihat pada Gambar 4.48.

Edit Nilai Harian Pelajaran Kimia Kelas X-A Tahun Ajaran 2012/2013  
 Pengajar: Inggit Andrianti, S.Pd

Jumlah Ulangan Harian : 1

No.	No Induk	Nama Siswa	1
1	1812	Hindarto Laytno	75
2	1814	Stevani Susanto	80

Gambar 4.48 *Input* Nilai Harian

Hasil *input* nilai harian ini secara otomatis akan menjadi salah satu nilai dalam perhitungan nilai akhir. Hal ini dapat dilihat pada Gambar 4.49.

Nilai Pelajaran Kimia Kelas X-A Tahun Ajaran 2012/2013

Pengajar: Inggit Andrianti, S.Pd

[Edit Nilai Harian](#)
[Edit Nilai Tugas](#)
[Edit Nilai Ulangan Umum](#)
[Edit Nilai Praktik](#)
[Edit Nilai Afektif](#)

No.	No Induk	Nama Siswa	Harian	Tugas	Ulangan Umum	Nilai Akhir	Nilai Praktik	Nilai Afektif
1	1812	Hindarto Laytno	75	75	90	80	88	A
2	1814	Stevani Susanto	80	50	90	73-3333	92	B

[Back](#)

Gambar 4.49 Hasil *Input* Nilai Harian

#### 4.10. Kenaikan Kelas

Kenaikan kelas dari seorang siswa dilihat berdasarkan nilai akhir dimilikinya. Halaman kenaikan kelas mengatur proses kenaikan kelas. Pada proses kenaikan kelas, siswa yang naik kelas akan diberi tanda centang pada *check box* yang tersedia, dan dilakukan pengaturan jurusan yang dipilih (IPA/IPS) pada *combo box* yang terletak di sebelah kanannya. Jika proses kenaikan kelas ini belum dilaksanakan, maka statusnya ialah “Belum diproses”. Hal ini dapat dilihat pada Gambar 4.50.

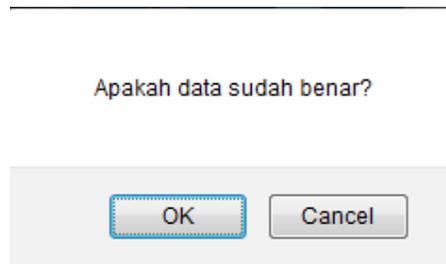
Proses Kenaikan Kelas X-A Tahun Ajaran 2012/2013

No.	No Induk	Nama Siswa	Nilai Akhir	Status	Naik ke tingkat 11	IPA/IPS
1	1812	Hindarto Laytno	75,00	Belum diproses	<input checked="" type="checkbox"/>	IPA ▾
2	1814	Stevani Susanto	80,00	Belum diproses	<input checked="" type="checkbox"/>	IPA ▾

[Save](#)
[Back](#)

Gambar 4.50 Kenaikan Kelas

Sebelum melakukan *input* data kenaikan kelas, *user* akan diberikan konfirmasi ulang. Hal ini dapat dilihat pada Gambar 5.51.



Gambar 4.51 Konfirmasi Data Kenaikan Kelas

Setelah dilakukan *input* data kenaikan kelas ke kelas XI IPA, status siswa pada proses kenaikan kelas ini berubah menjadi “Naik ke tingkat 11A”. Hal ini dapat dilihat pada Gambar 4.52.

Proses Kenaikan Kelas X-A Tahun Ajaran 2012/2013

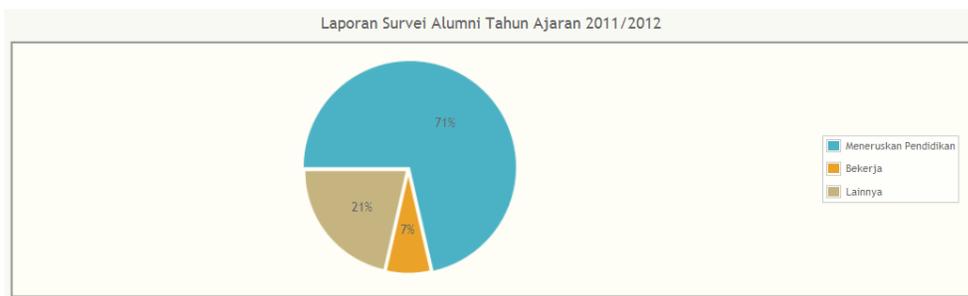
No.	No Induk	Nama Siswa	Nilai Akhir	Status	Naik ke tingkat 11	IPA/IPS
1	1812	Hindarto Laytno	75,00	Naik ke tingkat 11A	<input checked="" type="checkbox"/>	IPA
2	1814	Stevani Susanto	80,00	Naik ke tingkat 11A	<input checked="" type="checkbox"/>	IPA

Save Back

Gambar 4.52 Hasil Kenaikan Kelas

#### 4.11. Laporan Kuesioner Alumni

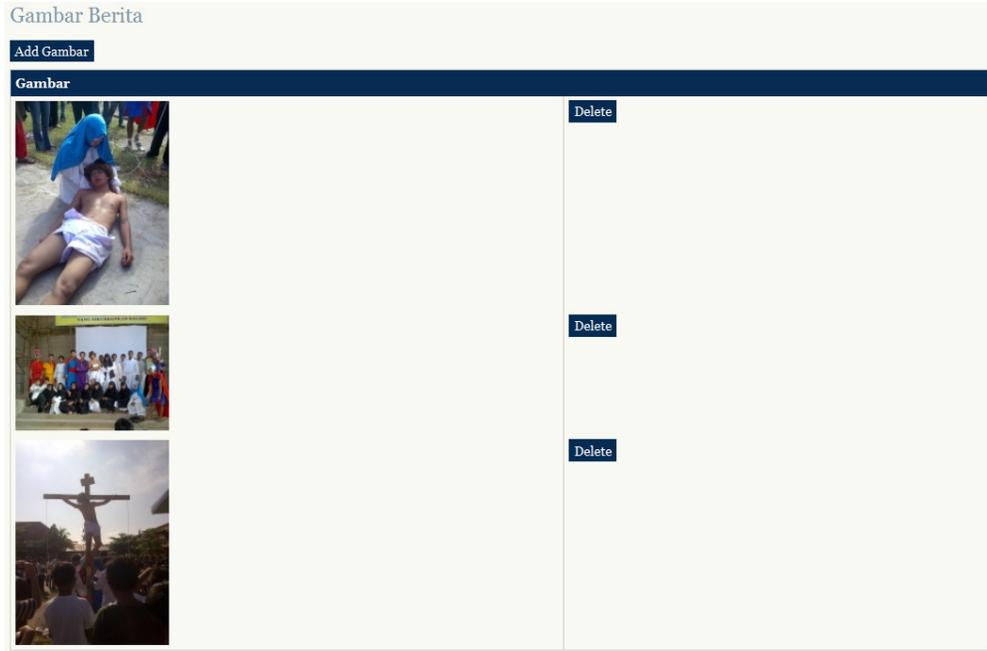
*Input* kuesioner alumni yang telah disimpan ke dalam *database* sebelumnya, akan diproses dalam bentuk laporan. Halaman laporan kuesioner alumni dapat dilihat pada Gambar 4.53.



Gambar 4.53 Laporan Kuesioner

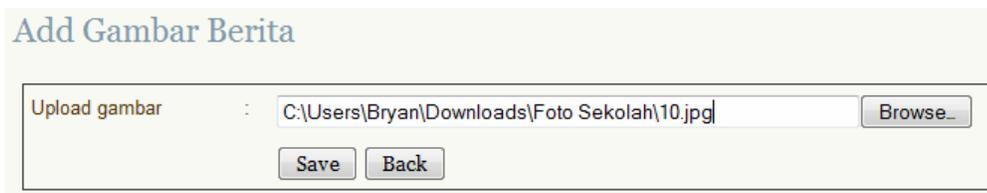
#### 4.12. Gambar Berita

Berita atau *news* yang telah disimpan di dalam *database* dapat diberikan gambar pendukung. Gambar ini dapat disimpan atau di-*delete* melalui halaman gambar berita sesuai keinginan *user*. Hal ini dapat dilihat pada Gambar 4.54.



Gambar 4.54 Halaman Gambar Berita

Gambar berita yang akan disimpan ke dalam *database* perlu terlebih dahulu di-*upload*. Hal ini dapat dilihat pada Gambar 4.55.



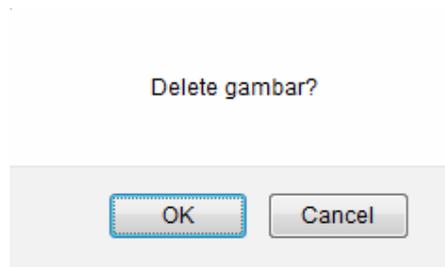
Gambar 4.55 Gambar Berita *Upload*

Setelah proses *upload* selesai, maka proses *add* gambar dapat dilihat pada halaman gambar berita. Hasil *upload* gambar tersebut dapat dilihat pada Gambar 4.56.



Gambar 4.56 Hasil *Upload*

*Delete* gambar berita digunakan untuk menghapus gambar berita yang terdapat pada *database*. Sebelum melakukan *delete* data gambar berita, *user* akan diberikan konfirmasi ulang. Hal ini dapat dilihat pada Gambar 4.57.



Gambar 4.57 Konfirmasi *Delete* Gambar Berita

Pada proses *delete* gambar berita yang dilakukan pengujian sistem, data gambar berita yang baru di-*upload*, pada proses *add* gambar berita sebelumnya, akan di-*delete*. Hasil *delete* data pendaftaran ini dapat dilihat pada Gambar 4.58.



Gambar 4.58 Delete Gambar Berita

#### 4.13. Laporan Nilai

Hasil proses penilaian yang dilakukan sebelumnya dapat dilihat dalam bentuk suatu laporan. Halaman laporan penilaian ini dapat dilihat pada Gambar 4.59

Nilai Pelajaran Kelas X-A Tahun Ajaran 2012/2013

No.	No Induk	Nama Siswa	Keterangan	Mata pelajaran											
				Bahasa Inggris	Biologi	Pendidikan Agama	TIKOM	Sosiologi	Matematika	Kimia	Sejarah	Bahasa Indonesia	Fisika	Penjaskes	
1	1812	Hindarto Laytno	K	87	75	90	85	92	70	82.5	88	80	80	82	
			P	90	80	75	76	83	80	88	77	80	80	88	
			A	A	B	A	B	A	A	A	B	A	B	A	
2	1814	Stevani Susanto	K	79	90	80	70	87	82	70	76	79.8333	85	87	
			P	88	80	77	89	75	80	92	87	80	75	94	
			A	B	A	B	B	A	A	B	C	B	B	A	

Back Print Excel Print

Gambar 4.59 Laporan Nilai

Laporan nilai ini juga dapat di-convert dalam bentuk file .xls (*Microsoft Excel*) dengan menggunakan *button print excel*. Hasil laporan nilai dalam bentuk file .xls ini dapat dilihat pada Gambar 4.60

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Nilai	Pelajaran Kelas X-A: Tahun Ajaran 2012/2013												
2	No	No Induk	Nama Siswa	Keterangan	Bahasa Inggris	Biologi	Pendidikan Agama	TIKOM	Sosiologi	Matematika	Kimia	Sejarah	Bahasa Indonesia	Fisika
3	1	1812	Hindarto Laytno	K	87	75	90	85	92	70	82.5	89	80	80
4				P	90	80	75	76	83	80	88	77	80	80
5				A	A	B	A	B	A	A	A	B	A	B
6				K	79	90	80	70	87	82	70	76	79.8333	85
7	2	1814	Stevani Susanto	P	88	80	77	89	75	80	92	87	80	75
8				A	B	A	B	B	A	A	B	C	B	B

Gambar 4.60 Laporan Nilai *Microsoft Excel*

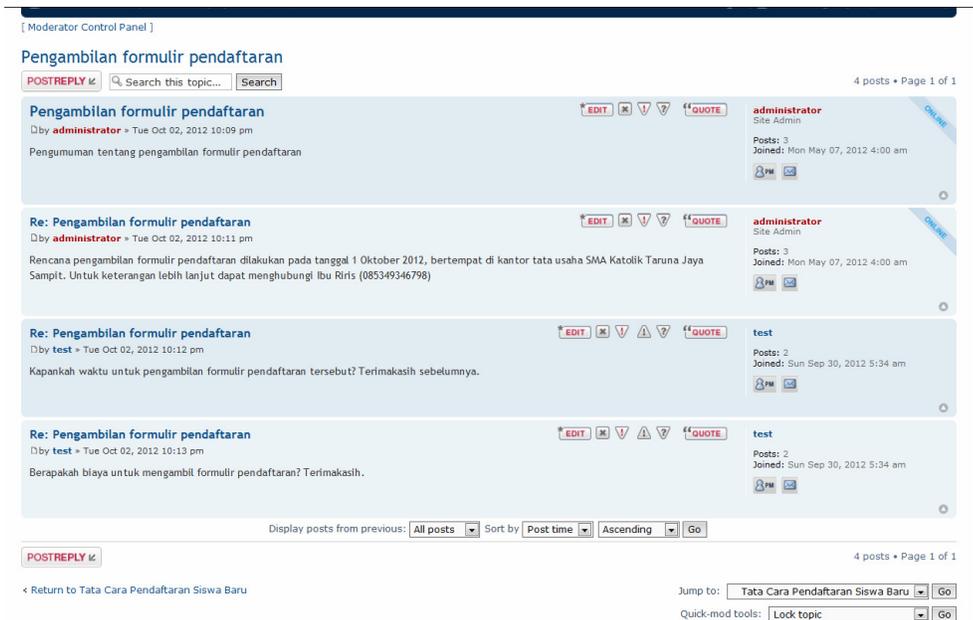
#### 4.14. Forum

Fitur forum digunakan untuk mempermudah komunikasi antara pihak sekolah dan pihak siswa atau orang tua siswa. Forum ini dapat dilihat pada Gambar 4.61.

The screenshot shows the forum interface for SMA Katolik Taruna Jaya. The header includes the forum logo and name. Below the header, there are navigation links, a search bar, and a list of forum topics. The topics listed include 'Pemberitahuan Kegiatan Registrasi Ulang' and 'Pengambilan formulir pendaftaran'. The page also shows user statistics and forum permissions.

Gambar 4.61 Forum

Pada forum ini, berbagai topik mengenai akademik sekolah dapat di-*post* dan dapat diberikan tanggapan oleh *user*, baik *user* administrator atau *user* siswa. Hal ini dapat dilihat pada Gambar 4.62.



Gambar 4.62 Forum Post

#### 4.15. Survei

Tabel 5.1 menjelaskan mengenai survei yang diadakan di SMA Katolik Taruna Jaya Sampit pada tanggal 2 Juni 2012. Subyek yang mengisi survei ini berjumlah 15 orang. Responden kuesioner ini terdiri dari pihak kepala sekolah dan pihak guru-guru yang berjumlah 14 orang. Survei memiliki skala penilaian dari 1 (sangat buruk) hingga 5 (sangat baik). Hasil pengisian survei ini dapat dilihat pada Tabel 4.1.

Tabel 4.1 Survei

No	Evaluasi	Peserta														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	Tingkat kemudahan dalam pemakaian aplikasi	4	4	4	4	4	5	5	4	4	4	4	4	4	4	4
2	Desain user interface atau tampilan dari aplikasi	4	5	4	5	5	5	5	4	4	4	5	4	4	4	4
3	Keakuratan informasi yang dihasilkan oleh aplikasi	3	4	5	5	5	5	5	4	5	4	4	4	4	4	5
4	Kesesuaian aplikasi dengan sistem yang ada	3	5	4	5	4	5	5	4	5	4	5	4	4	4	5
5	Kesesuaian aplikasi dengan kebutuhan dari pihak sekolah	4	5	5	5	5	5	5	4	5	5	5	4	5	4	5
6	Kelengkapan fitur-fitur pada aplikasi	4	4	4	5	5	5	5	4	4	5	4	4	4	4	4
7	Penggunaan bahasa dalam aplikasi	4	4	4	4	5	5	5	5	4	4	5	4	5	4	5
8	Petunjuk yang diberikan kepada user dalam penggunaan aplikasi	3	5	5	5	4	5	5	4	5	4	4	4	5	4	4
9	Aplikasi secara keseluruhan	4	5	4	5	5	5	5	4	4	4	4	4	4	4	4

Pada Tabel 4.2 dijelaskan mengenai rekapitulasi hasil survei Tabel 4.1.

Tabel 4.2 Rekapitulasi Hasil

No	Evaluasi	1 (Sangat Buruk)		2 (Buruk)		3 (Cukup)		4 (Baik)		5 (Sangat Baik)	
		0	0%	0	0%	0	0%	0	0%	13	86,6%
1	Tingkat kemudahan dalam pemakaian aplikasi	0	0%	0	0%	0	0%	13	86,6%	2	13,3%
2	Desain user interface atau tampilan dari aplikasi	0	0%	0	0%	0	0%	9	60	6	40
3	Keakuratan informasi yang dihasilkan oleh aplikasi	0	0%	0	0%	1	0,6	7	46,6	7	46,6
4	Kesesuaian aplikasi dengan sistem yang ada	0	0%	0	0%	1	0,6	7	46,6	7	46,6
5	Kesesuaian aplikasi dengan kebutuhan dari pihak sekolah	0	0%	0	0%	0	0%	4	26,6	11	73,3
6	Kelengkapan fitur-fitur pada aplikasi	0	0%	0	0%	0	0%	10	66,6	5	33,3
7	Penggunaan bahasa dalam aplikasi	0	0%	0	0%	0	0%	8	53,3	7	46,6
8	Petunjuk yang diberikan kepada user dalam penggunaan aplikasi	0	0%	0	0%	1	0,6	7	46,6	7	46,6
9	Aplikasi secara keseluruhan	0	0%	0	0%	0	0%	10	66,6	5	33,3

Setelah melalui proses perhitungan diketahui bahwa:

- 86, 6% dari *user* berpendapat bahwa tingkat kemudahan dalam pemakaian aplikasi ini baik dan 13, 3% dari *user* berpendapat bahwa tingkat kemudahan dalam pemakaian aplikasi ini sangat baik.
- 60% dari *user* berpendapat bahwa desain *user interface* atau tampilan aplikasi baik dan 40% dari *user* berpendapat bahwa desain *user interface* atau tampilan aplikasi sangat baik.
- 0, 6% dari *user* berpendapat bahwa keakuratan aplikasi dengan sistem yang ada cukup, 46, 6% dari *user* berpendapat bahwa keakuratan aplikasi dengan sistem yang ada baik, dan 46, 6% dari *user* berpendapat bahwa keakuratan aplikasi dengan sistem yang ada sangat baik.
- 0, 6% dari *user* berpendapat bahwa kesesuaian aplikasi dengan sistem yang ada sudah cukup, 46, 6% *user* berpendapat bahwa kesesuaian aplikasi dengan sistem yang ada baik, dan 46, 6% dari *user* berpendapat bahwa kesesuaian aplikasi dengan sistem yang ada sangat baik.
- 26, 6% dari *user* berpendapat bahwa kesesuaian aplikasi dengan kebutuhan dari pihak sekolah baik dan 73, 3% dari *user* berpendapat

bahwa kesesuaian aplikasi dengan kebutuhan dari pihak sekolah sangat baik.

- 66, 6% dari *user* berpendapat bahwa kelengkapan fitur pada aplikasi baik dan 33, 3% dari *user* berpendapat bahwa kelengkapan fitur pada aplikasi sangat baik.
- 53, 3% dari *user* berpendapat bahwa penggunaan bahasa dalam aplikasi baik dan 46, 6% dari *user* berpendapat bahwa penggunaan bahasa dalam aplikasi sangat baik.
- 0, 6% dari *user* berpendapat bahwa petunjuk yang diberikan kepada user dalam penggunaan cukup, 46, 6% dari *user* berpendapat petunjuk yang diberikan kepada user dalam penggunaan baik, dan 46, 6% dari *user* berpendapat bahwa petunjuk yang diberikan kepada user dalam penggunaan sangat baik.
- 66, 6% dari *user* berpendapat bahwa aplikasi secara keseluruhan baik dan 33, 3 % dari *user* berpendapat bahwa aplikasi secara keseluruhan sangat baik.

## 5. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Dari hasil pembuatan aplikasi Sistem Informasi Akademik SMA Katolik Taruna Jaya Sampit didapat beberapa kesimpulan, yaitu:

- Dari evaluasi yang telah dilakukan, penilaian untuk aplikasi ini membantu proses akademik dalam kesehariannya, dengan tingkat penilaian aplikasi secara keseluruhan sebesar 66, 6% dari *user* berpendapat baik dan 33, 3% dari *user* berpendapat sangat baik.
- Aplikasi memiliki kesesuaian dengan sistem yang ada dan dapat menghasilkan informasi yang baik. Hal ini dapat dilihat dari hasil perhitungan survei yang menyatakan 0, 6% dari *user* berpendapat bahwa keakuratan aplikasi dengan sistem yang ada cukup, 46, 6% dari *user* berpendapat bahwa keakuratan aplikasi dengan sistem yang ada baik, dan 46, 6% dari *user* berpendapat bahwa keakuratan aplikasi dengan sistem yang ada sangat baik. Kemudahan aplikasi dinilai dari hasil perhitungan survei dengan hasil 0, 6% dari *user* berpendapat bahwa kesesuaian aplikasi dengan sistem yang ada sudah cukup, 46, 6% *user* berpendapat bahwa kesesuaian aplikasi dengan sistem yang ada baik, dan 46, 6% dari *user* berpendapat bahwa kesesuaian aplikasi dengan sistem yang ada sangat baik.

### 5.2 Saran

Terdapat beberapa saran yang diharapkan dapat mendukung pengembangan lebih lanjut, yaitu :

- Pemberian menu bantuan atau *help* dalam penggunaan aplikasi.
- *Tools* penjadwalan masih diproses dengan sederhana melalui proses *random* dan pemberian *warning* bentrok antar jadwal pelajaran, sehingga

bagian penjadwalan ini dapat dikembangkan lebih baik dengan algoritma lainnya. Hal ini untuk menghasilkan jadwal pelajaran yang lebih baik.

## DAFTAR REFERENSI

- Al-Bahra Bin Ladjamudin (2005). *Analisis dan Desain Sistem Informasi*. Yogyakarta: Graha Ilmu.
- Bahtiar, Agus (2008). *PHP Script Most Wanted*. Yogyakarta: Penerbit Andi.
- Bentley & Whitten (2007). *System Analysis & Design for the Global Enterprise*. New York: McGraw-Hill Companies, Inc.
- Fathansyah (2007). *Basis Data*. Bandung: Penerbit Informatika.
- Hartono, Jogiyanto. (2009). *Sistem Teknologi Informasi*. Yogyakarta: Penerbit Andi.
- Jogiyanto, HM. (2005). *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Yogyakarta: Penerbit Andi.
- Kadir, Abdul. (2003). *Pengenalan Sistem Informasi*. Yogyakarta: Penerbit Andi.
- Kenneth E. Kendall & Julie E. Kendall (2010). *Analisis dan Perancangan Sistem Edisi Kelima Jilid 1*. Jakarta: Indeks.
- Kurniawan, Rudianto (2008). *Membangun Situs dengan PHP untuk Orang Awam*. Palembang: Maxikom.
- Nugroho, Bunafit. (2008). *Panduan Lengkap Menguasai Perintah SQL*. Jakarta: Media Kita.
- Raymond McLeod, Jr. & George P. Schell (2004). *Management Information System, 9th edition*. Pearson Education: New Jersey.
- Sibero, Alexander F.K. (2011). *Kitab Suci Web Programming*. Yogyakarta: Mediakom.