

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sistem informasi dan teknologi komputer berkembang sangat pesat dengan besarnya kebutuhan terhadap informasi. Perkembangan teknologi informasi tidak lepas dari pesatnya perkembangan teknologi komputer, karena komputer merupakan media yang dapat memberikan kemudahan bagi manusia diiringi dengan perkembangan cara penyampaian suatu informasi, mulai dari gambar-gambar yang ada di dinding-dinding gua, perletakan tonggak sejarah dalam bentuk prasasti sampai diperkenalkannya dunia dengan *internet*. Perubahan masyarakat yang semakin cepat seiring dengan perkembangan zaman dan teknologi sehingga memerlukan kualitas informasi yang akurat, cepat, dan tepat. Untuk menyediakan informasi tersebut, diperlukan suatu alat bantu atau media untuk mengolah beraneka ragam data agar dapat disajikan menjadi sebuah informasi yang bermanfaat dengan kemasan yang menarik dan berpedoman pada kriteria informasi yang berkualitas.

Dengan perkembangan teknologi informasi yang sangat pesat, teknologi informasi sekarang telah merambah di berbagai aspek kehidupan, tidak terkecuali pada dunia pendidikan. Saat ini bisa dijumpai hampir semua sekolah hingga mulai dari SMP hingga ke Perguruan Tinggi telah memiliki *website* masing-masing, yang digunakan untuk memperkenalkan profil sekolah tersebut. Dengan adanya *website* tersebut, sekarang masyarakat dapat lebih mudah mendapatkan informasi tentang sekolah yang mereka minati. Begitu pulda dengan SMA YPI Tunas

Bangsa Palembang yang sangat membutuhkan teknologi komputer untuk memudahkan pekerjaan di sekolah tersebut.

SMA YPI Tunas Bangsa Palembang merupakan salah satu Sekolah Menengah Atas swasta di Kota Palembang yang memiliki akreditasi A, yang terletak di jalan Residen H Abdul Rozak No 4 RT. 49/ 8 ilir/ Ilir Timur II Palembang. Sistem yang berjalan saat ini di SMA YPI Tunas Bangsa Palembang masih menggunakan komputer secara manual, yaitu menggunakan *Microsoft word* dan *Microsoft excel* sebagai alat bantu dalam proses pengolahan data (akademik). Proses pengolahan data tersebut adalah, data kelas, data guru, data siswa, data jadwal, data nilai, dan data mata pelajaran.

SMA YPI Tunas Bangsa Palembang ini masih menggunakan kertas sebagai media untuk memberikan informasi dan pengumuman. Informasi mulai dari nilai-nilai Subsumatif, mulai dari nilai-nilai Subsumatif, nilai Mid semester, dan nilai UAS, serta pengumuman-pengumuman seperti pengumuman pemberitahuan acara, dan pengumuman kalender sekolah. Semua pengumuman dan informasi itu dicetak di kertas dan di tempel di sebuah mading yang terletak di dinding kantor guru, hal itu membuat selalu ramainya kantor guru ketika waktu istirahat dan membuat keadaan menjadi sangat tidak terkendali karena semua siswa-siswa ingin melihat pengumuman, informasi dan nilai-nilai yang tercantum di mading tersebut. Masalah lain timbul adalah jika ada pengumuman mendadak dari pihak sekolah sedangkan keadaan sekolah lagi libur, seperti pengumuman batas akhir SPP dan pelunasan uang buku atau LKS (Lembar Kerja Siswa). Hal

itu membuat siswa-siswi tidak mengetahui jika di sekolah mereka ada pengumuman terbaru.

Berdasarkan permasalahan yang ada di SMA YPI Tunas Bangsa Palembang, penulis mencoba memberikan solusi dengan suatu rancangan baru yaitu, dengan pemanfaatan teknologi informasi. Berdasarkan hal tersebut maka dalam penyusunan skripsi ini mengambil judul **“SISTEM INFORMASI AKADEMIK DI SMA YPI TUNAS BANGSA PALEMBANG MENGGUNAKAN METODE MVC (*MODEL VIEW CONTROLLER*) BERBASIS WEB”**.

1.2 Rumusan Masalah

Sesuai dengan masalah yang diangkat pada latar belakang di atas, maka masalah yang dibahas dapat dirumuskan sebagai berikut :

1. Bagaimana sistem informasi akademik ini dapat membantu proses pencatatan data guru, siswa, kelas, jadwal pengajaran, mata pelajaran, dan penilaian dengan menggunakan sistem yang berbasis web.

1.3 Batasan Masalah

Pada penelitian ini penulis ingin membatasi masalah agar tidak meluas dari permasalahan yang ada.

1. Pendaftaran siswa baru tidak dibahas di dalam sistem ini.
2. Proses absensi siswa dan guru tidak dibahas di sistem ini
3. Proses pengisian rapor tidak dibahas di sistem ini, hanya sistem ini dapat melakukan pengolahan nilai siswa, mulai dari nilai Subsumatif, MID semester, nilai tugas, dan UAS.

4. Sistem informasi ini hanya dapat menginputkan jadwal mata pelajaran

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini yaitu :

1. Penelitian ini bertujuan untuk terciptanya sebuah sistem informasi akademik di SMA YPI Tunas Bangsa Palembang berbasis web.
2. Membangun sebuah sistem informasi yang dapat dimanfaatkan untuk menyajikan informasi yang baik dengan cepat dan mudah serta memberikan kemudahan dalam pembuatan laporan. Informasi disini maksudnya adalah seperti informasi mengenai nilai yaitu penginputan nilai yang dilakukan oleh guru dan pencetakan nilai yang dilakukan oleh siswa, jadwal pelajaran yang berfungsi untuk siswa dan guru, dan berita-berita seputar sekolah.

1.5 Manfaat Penelitian

Sesuai dengan permasalahan dan tujuan penelitian yang telah disebutkan di atas, maka manfaat penelitian dapat dirumuskan sebagai berikut :

1. Bagi Penulis

- a. Menambah wawasan penulis tentang teknologi pemograman PHP dan basis data *MySQL* yang dapat diterapkan langsung dengan mengembangkan aplikasi berbasis Web.
- b. Menciptakan sebuah aplikasi berbasis web yang dapat digunakan untuk mengelola sistem akademik di sekolah serta melatih sumber daya yang ada

untuk memanfaatkan teknologi yang sedang berkembang pesat sekarang ini.

2. Bagi Sekolah

- a. Aktivitas akademik di SMA YPI Tunas Bangsa Palembang dapat tercatat dengan rapi dan efisien dalam bentuk media maupun tenaga.
- b. Membantu yayasan dan sekolah untuk memanfaatkan teknologi informasi yang harus dipakai pada era sekarang ini.
- c. Membuat sistem pengarsipan data sekolah lebih rapi dari pada menggunakan kertas.
- d. Membantu para siswa-siswi lebih mudah untuk mendapatkan informasi dan pengumuman sekitar tentang sekolah mereka.
- e. Lebih memudahkan para siswa dan guru untuk mendapatkan informasi kapanpun dan dimanapun, tanpa harus datang ke sekolah.

1.6 Metodologi Penelitian

Teknik pengumpulan data yang digunakan penulis ini dalam penelitian adalah :

Metode Pengumpulan Data

a. Metode Observasi (Pengamatan)

Observasi adalah metode pengumpulan informasi dengan cara pengamatan atau peninjauan langsung terhadap obyek penelitian, yaitu melakukan pengamatan terhadap proses ujian semester pada sekolah SMA YPI Tunas Bangsa Palembang

b. Metode Wawancara

Merupakan proses tanya jawab secara langsung dengan dua atau beberapa orang pengumpulan data dan informasi dengan cara melakukan wawancara ini dilakukan dengan pihak instansi terkait. Dalam hal ini dilakukan wawancara dengan Humas sekolah SMA YPI Tunas Bangsa Palembang.

c. Metode Studi Pustaka

Pengumpulan data yang bersumber dari berbagai buku yang menjadi referensi data dan pencarian dengan media *internet* untuk memperoleh data tambahan dalam rangka melengkapai skripsi.

Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan yaitu RUP (*Rational Unified Process*). RUP (*Rational Unified Process*) merupakan pendekatan pengembangan perangkat lunak yang dilakukan berulang-ulang, fokus pada arsitektur, lebih di arahkan berdasarkan penggunaan kasus khusus untuk pemograman berorientasi objek dikembangkan oleh *Rational Software* (Rosa dan Shalahuddin, 2014: 124).

1.7 Sistematika Penulisan

Dalam penulisan dan tersusunnya pembuatan skripsi ini penulis akan membagi sistematika penulisan dalam lima bab, dimana satu dan yang lainnya saling berhubungan, maka penting sekali adanya sistematika penulisan.

Adapun sistematika penulisannya adalah sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini menjelaskan mengenai latar belakang masalah, rumusan masalah, tujuan dan manfaat, dan metodologi penelitian.

BAB II LANDASAN TEORI

Pada bab ini berisi tentang landasan teori, yaitu penjelasan dari hal-hal yang berhubungan dengan penulisan skripsi, sistem, informasi, akademik, *software-software* pendukung, metode pembangunan sistem, metode pengembangan sistem, dan tinjauan pustaka.

BAB III ANALISIS DAN DESAIN

Pada bab ini menjelaskan mengenai analisis kebutuhan dan rancangan desain dari *flowchart*, *database*, desain menu admin dan *user*.

BAB IV IMPLEMENTASI DAN TESTING

Pada bab ini menjelaskan implementasi dari sistem tersebut dan melakukan testing kepada sistem itu menggunakan metode-metode tertentu untuk menentukan sistem tersebut telah layak atau belum layak digunakan.

BAB V PENUTUP

Pada bab ini merupakan bab terakhir dalam penulisan skripsi yang berisi tentang hasil dari kesimpulan berdasarkan analisa dan desain sistem yang telah dilakukan. Bab ini juga berisi saran-saran secara keseluruhan sehingga sistem yang telah dibuat dapat dikemhangkan menjadi sistem yang lebih baik atau sistem yang lebih besar.

BAB II

LANDASAN TEORI

2.1 Sistem

Menurut I Putu Agus Eka Pratama sistem dapat didefinisikan sebagai sekumpulan prosedur yang saling berkaitan dan saling terhubung untuk melakukan suatu tugas bersama-sama. (I Putu Agus Eka Pratama, 2013:7)

Menurut Jogiyanto sistem dapat didefinisikan dengan prosedur dan dengan pendekatan komponen. Dengan pendekatan prosedur, sistem dapat didefinisikan sebagai kumpulan prosedur yang mempunyai tujuan tertentu. (Jogiyanto, 2008:34)

Maka dari itu dapat disimpulkan, sistem adalah sekumpulan prosedur dan komponen yang memiliki suatu tujuan dan tugas tertentu yang berguna untuk pemakai.

2.2 Informasi

Informasi adalah salah satu sumber daya penting dalam suatu organisasi, digunakan sebagai bahan pengambilan keputusan, sehubungan dengan hal itu, informasi haruslah berkualitas. (Abdul Kadir dan Terra CH. Triwahyuni, 2003:546)

Informasi merupakan hasil pengolahan data dari satu atau berbagai sumber, yang kemudian diolah, sehingga memberikan nilai, arti, dan manfaat. (I Putu Agus Eka Pratama, 2013:9).

Dari definisi-definisi tersebut maka dapat disimpulkan bahwa informasi adalah salah satu sumber daya yang berupa kumpulan data-data yang telah diolah

atau diproses menjadi sesuatu yang berguna dan bermanfaat bagi para penggunanya.

2.3 Sistem Informasi

Menurut Alter (1992) sistem informasi adalah kombinasi antar prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai suatu tujuan dalam sebuah organisasi. (Abdul Kadir dan Terra CH. Triwahyuni, 2003:546)

Menurut Abdul Kadir (2003: 10) sistem informasi mencakup sejumlah komponen (manusia, komputer, teknologi informasi, dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan.

Berdasarkan uraian di atas, maka penulis dapat menyimpulkan definisi dari sistem informasi adalah sekumpulan elemen-elemen yang saling berkaitan satu sama lain dalam mengolah data sehingga menjadi suatu informasi yang bernilai dan bermanfaat.

2.4 Akademik

Pengertian Akademik di dalam jurnal Andri Setiyawan dkk (ISSN: 2302-5700) adalah kegiatan yang dilakukan di dalam lingkungan dunia pendidikan yang berhubungan dengan proses belajar mengajar.

2.5 Sistem Informasi Akademik

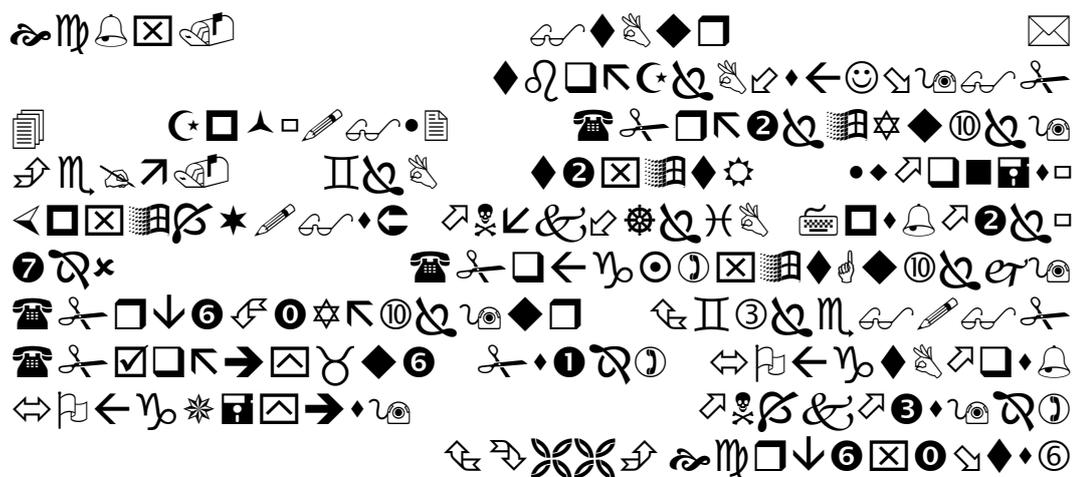
Sistem informasi akademik (SIA) di dalam jurnal Etin Indryani (April 2011 ISSN 1412-565X) adalah himpunan dari berbagai macam data yang dikelola dan

diproses se-otomatis mungkin dengan alat dan metoda sehingga menghasilkan informasi yang diperlukan bagi terlaksananya kegiatan akademis.

Di dalam jurnal Andri Setiyawan dkk (ISSN: 2302-5700) pengertian sistem informasi akademik adalah sistem yang memberikan layanan informasi yang berupa data dalam hal yang berhubungan dengan data akademik.

Maka dapat disimpulkan sistem informasi akademik adalah sebuah sistem yang dapat membantu dan memudahkan pengguna dalam mengakses data dalam hal yang berhubungan dengan akademik.

Di dalam islam mengenai akademik atau pendidikan telah disebutkan dalam Al-qur'an surat At- Taubah ayat 122:



Artinya: "Tidak sepatutnya bagi mukminin itu pergi semuanya (ke medan perang). mengapa tidak pergi dari tiap-tiap golongan di antara mereka beberapa orang untuk memperdalam pengetahuan mereka tentang agama dan untuk memberi peringatan kepada kaumnya apabila mereka telah kembali kepadanya, supaya mereka itu dapat menjaga dirinya."

Ayat ini memberi anjuran tegas kepada umat islam agar ada sebagian dari umat islam untuk memperdalam agama. Dikatakan juga bahwa yang dimaksud

kata *tafaqquh di al-din* adalah menjadi seorang yang mendalam ilmunya dan selalu memiliki tanggung jawab dalam pencarian ilmu Allah.

2.6 MVC (*Model View Controller*)

MVC (*Model-View-Controller*) adalah pola desain perangkat lunak yang dibangun di sekitar interkoneksi dari tiga tipe komponen utama, dalam bahasa pemrograman seperti *PHP*, sering dengan fokus yang kuat pada pemrograman berorientasi objek (OOP) paradigma perangkat lunak. Ketiga tipe komponen yang diistilahkan dengan *Model*, *View*, dan *Controller*.

Model ini di mana semua logika bisnis dari sebuah aplikasi disimpan. Logika bisnis dapat sesuatu yang spesifik untuk bagaimana aplikasi menyimpan data, atau menggunakan jasa pihak ketiga, dalam rangka memenuhi kebutuhan bisnisnya. Jika aplikasi harus mengakses informasi dalam *database*, kode untuk melakukan itu akan disimpan dalam *model*. Jika itu diperlukan, misalnya, untuk mengambil data saham atau tweet (memposting) tentang produk baru, bahwa kode yang juga akan disimpan di *model*.

View di mana semua pengguna elemen antarmuka dari aplikasi kita simpan. Hal ini dapat mencakup mark up *HTML*, *CSS style sheet*, dan file *JavaScript*. Pengguna apa pun melihat atau berinteraksi dengan dapat disimpan dalam sebuah gambaran, dan kadang-kadang apa yang pengguna melihat sebenarnya kombinasi dari banyak pandangan yang berbeda dalam permintaan yang sama.

Controller merupakan komponen yang menghubungkan model dan *view* bersama-sama. *Controller* mengisolasi logika bisnis model dari pengguna elemen antarmuka *view*, dan menangani bagaimana aplikasi akan merespon pengguna

interaksi dalam *view*. *Controller* merupakan poin pertama masuk ke komponen trio ini, karena permintaan pertama dilewatkan ke *Controller*. (Chris Pitt, 2012: 1)

2.7 PHP (*Hypertext Preprocessor*)

Menurut Wibowo (2007:2) dengan buku yang berjudul 16 Aplikasi PHP Gratis untuk Pengembangan Situs Web, PHP (*PHP Hypertext Preprocessor*) adalah bahasa *scripting server-side* bagi pemrograman *web*. Secara sederhana, PHP merupakan *tool* bagi pengembangan *web* dinamis. PHP sangat populer karena memiliki fungsi *built-in* lengkap, cepat, mudah dipelajari, dan bersifat gratis.

2.8 Database

Database adalah kumpulan data yang terkait. Secara teknis, yang berada dalam sebuah *database* adalah sekumpulan tabel atau objek lain (*indeks, view, dan lain-lain*). Tujuan utama pembuatan *database* adalah untuk memudahkan dalam mengakses data. Data dapat ditambahkan, diubah, dihapus, atau dibaca relative lebih mudah. (Abdul Kadir, 2008: 14)

2.9 MySQL

MySQL merupakan software yang tergolong database server dan bersifat open source. Open source menyatakan bahwa software ini dilengkapi dengan *sourcecode* (kode yang dipakai untuk membuat *MySQL*), selain tentu saja bentuk kode yang dapat dijalankan secara langsung dalam sistem operasi, dan bisa diperoleh dengan cara mengunduh di internet secara gratis. Hal yang menarik lainnya adalah *MySQL* dapat dijalankan pada berbagai sistem operasi. (Abdul Kadir, 2009 :25)

Lain lagi menurut Sadeli (2014:10) *MySQL* adalah *database* yang menghubungkan script php menggunakan perintah *query* dan *escaps character* yang sama dengan php. *MySQL* mempunyai tampilan client yang mempermudah anda dalam mengakses *database* dengan kata sandi untuk mengizinkan proses yang bisa anda lakukan.

Menurut Kadir (2010:10) dalam bukunya yang berjudul Mudah Mempelajari Database *MySQL*, *MySQL* tergolong sebagai *DBMS (Database Management System)*. Perangkat lunak ini bermanfaat untuk mengelola data dengan cara yang sangat fleksibel dan cepat.

2.10 Metode Pengembangan Sistem

2.10.1 RUP (*Rational Unifed Proses*)

Metode pengembangan sistem menggunakan metodologi yang berorientasi objek karena metodologi berorientasi objek didasarkan pada penerapan prinsip-prinsip pengelolaan kompleksitas dan pendekatan objek secara sistematis yang meliputi rangkaian aktivitas analisis berorientasi objek, perancangan berorientasi objek, pemrograman berorientasi objek dan pengujian berorientasi objek (Rosa dan Shalahuddin, 2014:100).

Metode pengembangan sistem yang digunakan yaitu RUP (*Rational Unifed Process*). RUP (*Rational Unifed Process*) merupakan pendekatan pengembangan perangkat lunak yang dilakukan berulang-ulang, fokus pada arsitektur, lebih di arahkan berdasarkan penggunaan kasus khusus untuk pemrograman berorientasi objek dikembangkan oleh *Rational Software*. RUP merupakan proses rekayasa perangkat lunak dengan pendefinisian yang baik dan

penstrukturan yang baik. RUP menyediakan pendefinisian struktur yang baik untuk alur hidu proyek perangkat lunak. RUP adalah sebuah produk proses perangkat lunak yang dikembangkan oleh *Rational Software* yang diakuisisi oleh IBM di bulan Februari 2003.(Rosa dan Shalahuddin, 2014: 124).

Fase-fase dalam Metodologi RUP terdiri dari (Rosa A.S dan M.Shalahuddin, 2014: 125):

1. *Inception* (Permulaan)

Tahap ini lebih pada memodelkan proses bisnis yang dibutuhkan dan mendefinisikan kebutuhan akan sistem yang akan dibuat. Berikut adalah tahap yang dibutuhkan pada tahap ini yaitu, memahami ruang lingkup dari proyek dan membangun kasus bisnis yang dibutuhkan.Jika pada akhir tahap ini target yang diinginkan tidak dicapai maka dapat dibatalkan atau diulang kembali setelah dirancang ulang agar kriteria yang diinginkan dapat dicapai. Batas/tonggak objektif digunakan untuk mendeteksi apakah sebuah kebutuhan akan sistem dapat diimplementasikan atau tidak.

Proses yang dilakukan pada tahap *inception* yaitu, memahami ruang lingkup di SMA YPI Tunas Bangsa Palembang dengan cara mengidentifikasi masalah yang terjadi disana dan mendefinisikan kebutuhan akan sistem yang akan dibuat.

2. *Elaboration* (Perluasan/perencanaan)

Tahap ini lebih difokuskan pada perencanaan arsitektur sistem. Tahap ini juga dapat mendeteksi apakah arsitektur sistem yang diinginkan dapat dibuat atau tidak. Mendeteksi resiko yang mungkin terjadi dan arsitektur yang dibuat. Tahap ini lebih pada analisis dan desain sistem serta implementasi sistem yang fokus

purwarupa sistem. Jika pada akhir tahap ini target yang diinginkan tidak dicapai maka dapat dibatalkan atau diulang kembali.

Batas/tonggak arsitektur digunakan untuk mendeteksi apakah sebuah kebutuhan akan sistem dapat diimplemenasikan atau tidak melalui pembuatan arsitektur. Proses yang dilakukan di tahap *elaboration* ini yaitu, membuat rancangan sistem yang akan diusulkan dengan menggunakan UML.

3. *Construction* (Konstruksi)

Tahap ini fokus pada pengembangan komponen dan fitur-fitur sistem. Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada implementasi perangkat lunak pada kode program. Tahap ini menghasilkan produk perangkat lunak dimana menjadi syarat dari *Initial Operational Capability* atau batas/tonggak kemampuan operasional awal.

Proses yang dilakukan di tahap *construction* ini yaitu, melakukan implementasi ke dalam bentuk kode program berdasarkan sistem yang telah di rancang pada tahap *elaboration*.

4. *Transition* (Transisi)

Tahap ini lebih pada *deployment* atau instalasi sistem agar dapat dimengerti oleh user. Tahap ini menghasilkan produk perangkat lunak dimana menjadi syarat dari *Initial Operational Capability Milestone* atau batas/tonggak kemampuan operasional awal. Aktifitas pada tahap ini termasuk pada pelatihan user, pemeliharaan dan pengujian sistem apakah sudah memenuhi harapan user. Produk perangkat lunak juga disesuaikan dengan kebutuhan yang didefinisikan pada tahap *inception*. Jika semua kriteria objektif terpenuhi maka dianggap sudah memenuhi

batas/tonggak peluncuran produk dan pengembangan perangkat lunak selesai dilakukan.

Proses yang dilakukan pada tahap *transition* ini yaitu, melakukan instalasi sistem dengan cara menghostingnya dan memberikan pelatihan *user*, pemeliharaan dan pengujian kepada Humas SMA YPI Tunas Bangsa Palembang selaku admin sistem.

2.11 Metode Pengujian Sistem

2.11.1 *White-Box Testing* (Pengujian Kotak Putih)

White box testing sering disebut juga pengujian kotak kaca (*glass-box testing*) merupakan sebuah filosofi perancangan *test case* yang menggunakan struktur kontrol yang dijelaskan sebagai bagian dari perancangan perangkat komponen untuk menghasilkan *test case*.

2.11.2 *Black-Box Testing* (Pengujian Kotak Hitam)

Black-box testing adalah sebuah metode pengujian sistem yang menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. (Rosa dan Shalahuddin, 2014:275).

Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. (Rosa dan Shalahuddin, 2014:276).

2.12 Desain Perancangan Sistem

2.12.1 UML (*Unified Modeling Language*)

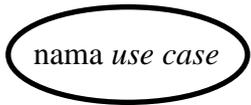
UML adalah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemograman berorientas objek. (Rosa dan Shalahuddin, 2014:137).

Dalam UML terdapat 13 macam diagram seperti *Class Diagram*, *Object Diagram*, *Component Diagram*, *Composite Diagram*, *Composite Structure Diagram*, *Package Diagram*, *Deployment Diagram*, *Use Case Diagram*, *Activity Diagram*, *State Machine Diagram*, *Sequence Diagram*, *Communication Diagram*, *Timing Diagram*, dan *Interaction Overview Diagram* (Rosa A.S dan M. Shalahuddin, 2013: 139). Disini hanya akan menggunakan 4 macam diagram, yaitu :

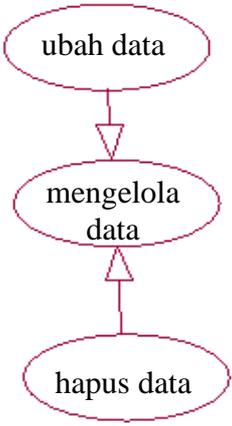
2.12.1.1 *Use Case Diagram*

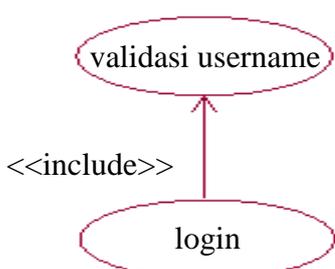
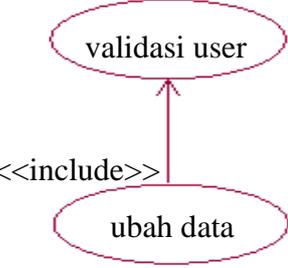
Use Case atau diagram *use case* merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. (Rosa dan Shalahuddin, 2013: 155). Berikut ini adalah simbol-simbol yang ada pada diagram use case :

Tabel 2.1 Simbol-Simbol *Use Case Diagram*

No	Simbol	Deskripsi
	<p><i>Use case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama <i>use case</i>
2	Aktor atau <i>actor</i>	Orang, proses, atau sistem lain



		yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama actor
3	Asosiasi atau <i>association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor
4	Ekstensi atau <i>extend</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan
5	Generalisasi atau <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya:  arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum)

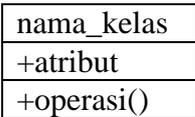
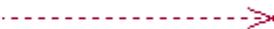
6	<<include>>	<p>Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i>:</p> <ol style="list-style-type: none"> 1. <i>include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, misal pada kasus berikut:  <pre> graph BT login((login)) -- <<include>> --> validasi_username((validasi username)) </pre> <ol style="list-style-type: none"> 2. <i>include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan, misalnya pada kasus berikut:  <pre> graph BT ubah_data((ubah data)) -- <<include>> --> validasi_user((validasi user)) </pre>
---	-------------	--

2.12.1.2 Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas dan metode atau operasi adalah fungsi-fungsi yang dimiliki oleh suatu kelas. Diagram kelas dibuat agar

programmer membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.

Tabel 2.2 Simbol-simbol *Diagram Class*

No	Simbol	Deskripsi
1	Kelas 	Kelas pada struktur sistem
2	antarmuka atau <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
3	asosiasi atau <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
4	asosiasi berarah atau <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
5	generalisasi 	Relasi antarkelas dengan makna generalisasi
6	kebergantungan atau <i>dependency</i> 	Kebergantungan antarkelas
7	Agregasi atau <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>)

2.12.1.3 *Activity Diagram*

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram

aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2013: 161).

Berikut adalah simbol-simbol yang ada pada diagram aktivitas:

Tabel 2.4 Simbol-simbol *Activity Diagram*

No	Simbol	Deskripsi
1	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
2	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3	Percabangan atau <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
4	Penggabungan atau <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
5	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir

6	<p><i>Swimlane</i></p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <p style="text-align: center;">nama swimlane</p> </div> <p>atau</p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <table border="1" style="width: 100%; height: 100%;"> <tr> <td style="width: 20%; text-align: center; vertical-align: middle;">nama swimlane</td> <td></td> </tr> </table> </div>	nama swimlane		<p>Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi</p>
nama swimlane				

2.13 Tinjauan Pustaka

Penelitian sebelumnya dilakukan oleh Anjar Priyadna dan Berliana Kusuma Riasti dalam sebuah jurnal yang berjudul Pembuatan Sistem Informasi Nilai Akademik Berbasis SMS Gateway pada SMP Negeri 3 Pringkuku Pacitan. Penelitian ini menghasilkan sebuah sistem informasi yang bisa dimanfaatkan oleh orang tua siswa untuk mengakses informasi nilai akademik siswa melalui sms sehingga diharapkan orang tua siswa bisa merespon nformasi yang diterima dengan memperhatikan pendidikan putra putrinya.

Penelitian selanjutnya dilakukan oleh Bambang Nurdiansyah dalam sebuah skripsi yang berjudul Sistem Informasi Akademik Berbasis Web Pada SMP Negeri 6 Purwodadi. Peneliti menggunakan metodologi prototype sebagai

metodologi pengembangan sistem. Penelitian ini menghasilkan suatu sistem informasi yang berisi jadwal mengajar, jadwal pelajaran, tugas siswa dan jadwal ujian siswa yang dapat diakses secara online dan memperoleh informasi sekolah secara lengkap dengan cepat melalui website online ini.

Adapula penelitian yang dilakukan oleh Nendy Subhansyah dalam sebuah skripsi yang berjudul perancangan sistem akademik sekolah berbasis teknologi *mobile web* (studi kasus : SMA Muhammadiyah 3 Tangerang). Peneliti menggunakan metodologi *incremental* yang merupakan metode pengembangan dari metode *waterfall* itu sendiri dengan sifat pengulangan. Sistem informasi ini berguna untuk membantu para siswa mendapatkan informasi tentang sekolah mereka dengan lebih cepat karena dapat diakses melalui *handphone* siswa masing-masing. Sistem informasi ini berisi data-data sekolah seperti data kepala sekolah, data guru, data pegawai, data siswa, nilai-nilai, dan pengumuman-pengumuman penting dari pihak sekolah.

Penelitian lain yang dilakukan oleh Jihan Ali Ahmad dalam sebuah jurnal yang berjudul Reayasa Sistem Informasi Akademik Sekolah Berdasarkan Kurikulum 2013 dengan menggunakan Framework CodeIgniter. Peneliti ini menggunakan metode Model Sekuensial Linier sebagai metode pengembangan sistem. Sistem informasi ini berguna untuk menerapkan kurikulum 2013 yang mempunyai aspek penilaian yang lebih banyak dari pada kurikulum sebelumnya.

Lain lagi penelitian yang dilakukan oleh Suryana Wijaya dalam sebuah skripsi yang berjudul Sistem Informasi Akademik Universitas Islam Negeri (UIN) Sunan Kalijaga Yogyakarta Berbasis Android. Peneliti ini menggunakan SDLC

(*Software Development Life Cycle*) sebagai metode pengembangan sistem. Sistem informasi ini menghasilkan sebuah aplikasi yang dapat digunakan untuk mengakses informasi akademik mahasiswa, seperti jadwal kuliah, jadwal ujian, kartu hasil studi, dan indeks prestasi (IP).

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Gambaran Umum SMA YPI Tunas Bangsa Palembang (*Inception/perencanaan*)

3.1.1 Sejarah SMA YPI Tunas Bangsa

Yayasan Pendidikan Islam (YPI) Tunas Bangsa Palembang, saat ini mengelola SMP dan SMA yang didirikan pada tahun 1986 (Akte notaris No. 57/2610/1886, 26 Oktober 1986) yang mengelola SMP/SMA YPI Tunas Bangsa Palembang. Sejak didirikan hingga saat ini telah mengalami perkembangan pesat diberbagai bidang, baik prestasi akademik, tenaga edukatif, sarana, dan prasarana maupun jumlah siswanya.

Yayasan Pendidikan Islam (YPI) Tunas Bangsa Palembang didirikan pada tanggal 5 Nopember 1985 dengan akte notaris No. 57 Tahun 1985, dengan susunan pengurus sebagai berikut :

Ketua : Ir. Ahmad Dimiati, MBA
Wakil Ketua : M. Oktaruddin, S.T
Sekretaris I : Yuni Andriyani, S.T
Bendahara : Dra. Dwi Septinawati, M.Psi

Dasar didirikannya SMA YPI Tunas Bangsa adalah untuk membantu pemerintahan mendidik dan membina siswa yang tidak tertampung di sekolah negeri dan meningkatkan mutu pendidikan pada umumnya. Terletak di atas tanah seluas 11017 m², di jalan Residen H. Abdul Rozak No. 4 RT. 49 RW. 10

Kelurahan 8 Ilir Kecamatan ILIR TIMUR II Palembang. Mulai beroperasi pada tahun ajaran 1986/1987 dengan piagam pengesahan dari Kantor Wilayah Departemen Pendidikan dan Kebudayaan Propinsi Sumatera Selatan No. 26/I/11/2B/F4e/1986, tanggal 26 Mei 1986.

Pada akhir tahun 1999 diadakan akreditasi oleh tim dari Kantor Wilayah Depdikbud Sumatera Selatan dengan hasil status “Diakui” berdasarkan piagam jenjang Akreditasi No 92/F.11/I/1990. Akreditasi akhir SMA YPI Tunas Bangsa Palembang pada tahun 2011 dengan predikat “TERAKREDITASI A” berdasarkan surat keputusan atau SK. Depdikbud RI No. 350/BAP-SM/TU/XI/2011.Ma.011046 tanggal 9 November 2011.

Keseluruhan itu ditujukan untuk membentuk sekolah yang bermutu, menjadi sekolah bercirikan Islam yang terkemuka di Palembang, memiliki keunggulan terintegrasi keilmuan pengetahuan, teknologi dan ilmu agama, menghasilkan manusia islam yang unggul, tangguh dan serasi yang bertaqwa kepada Tuhan Yang Maha Esa, berakhlak baik, berjiwa sosial dan berbudaya. Bahkan untuk kegiatan-kegiatan tertentu siswa SMA YPI Tunas Bangsa ini telah mampu berkiprah pada tingkat nasional mewakili Sumatera Selatan bahkan pernah sampai pada tingkat nasional.

Keberhasilan dan perkembangan yang begitu pesat tersebut terlaksana berkat dukungan semua pihak dan semua ini berlandaskan pada program kami untuk mencetak para lulusan yang sanggup menghadapi tantangan pembangunan nasional jangka panjang dengan landasan iman, taqwa kepada Allah SWT, untuk mewujudkan cita-cita bangsa.

SMA YPI Tunas Bangsa Palembang merupakan sekolah lanjutan tingkat atas yang terletak di Jalan Residen H. Abdul Rozak Rt.49 No.4 RW.10 Kelurahan 8 ILIR Kecamatan ILIR TIMUR II Palembang Kode Pos 30114 Nomor Telepon 0711 713859.

SMA YPI Tunas Bangsa Palembang dipimpin oleh Bapak Fahrurrozi, S.Pd., M.M. sebagai kepala sekolah, wakil kepala sekolah Bidang kurikulum Bapak Riffan Afandi, S.Pd., M.M dan wakil kepala sekolah bidang Kesiswaan Bapak Pauzan, S.Ag., serta Wakil Kepala Sekolah Bidang Humas Bapak Purwadi Susilo, S.Pd. Guru yang mengajar berjumlah 56 orang yang berasal dari berbagai alumni perguruan tinggi ternama baik di kota Palembang maupun luar Sumatera Selatan. SMA YPI Tunas Bangsa Palembang merupakan Sekolah yang berada dibawah naungan Yayasan Pendidikan Islam Tunas Bangsa yang didirikan oleh Bapak Drs. Zainal Abidin Ali (Suami Ibu Hj. Sumiati, B.A.) pada tahun 1986. SMA YPI Tunas Bangsa hingga kini terus berbenah guna meningkatkan mutu dan kualitas lulusannya. Sesuai dengan motto sekolah “kualitas adalah tradisi”.

3.1.2 Keadaan SMA YPI Tunas Bangsa

I. KEADAAN YAYASAN

1. Nama Yayasan : Yayasan Pendidikan Islam Tunas Bangsa
2. Alamat Yayasan : Jl. Residen H. Abdul Rozak Rt. 49 No. 04
Palembang
3. Nama Ketua Yayasan : Ir. Ahmad Dimiati, MBA
4. Telp/Hp : 0711 713859/ 0711 710148

II. KEADAAN SEKOLAH

1. Nama Sekolah : SMA YPI Tunas Bangsa Palembang
2. N D S : 3011100074
3. N I S : 10609719
4. N S S : 302116002086
5. S. K Izin Pendirian Nomor : No. 25/ii.2B/F4e
6. Akreditasi Tahun : 2011/ 9 November
7. Alamat Sekolah : Jl. Residen H. Abdul Rozak RT 49 RW 10
No. 04 Kec. Ilir Timur II Kotamadya
Palembang
8. Pelaksanaan Belajar : Pagi
9. Sekolah Induk : SMA Negeri 14 Palembang

3.1.3 Visi dan Misi SMA YPI Tunas Bangsa

Visi SMA YPI Tunas Bangsa

Menjadi sekolah bercirikan islam yang terkemuka di Palembang, memiliki keunggulan terintegrasi keilmuan, teknologi, dan ilmu agama, menghasilkan manusia islam yang unggul, tangguh, dan serasi yang bertaqwa kepada Tuhan Yang Maha Esa, berakhlak baik, berjiwa sosial, dan berbudaya.

Misi SMA YPI Tunas Bangsa

1. Menyelenggarakan pendidikan umum yang dijiwai ilmu agama.

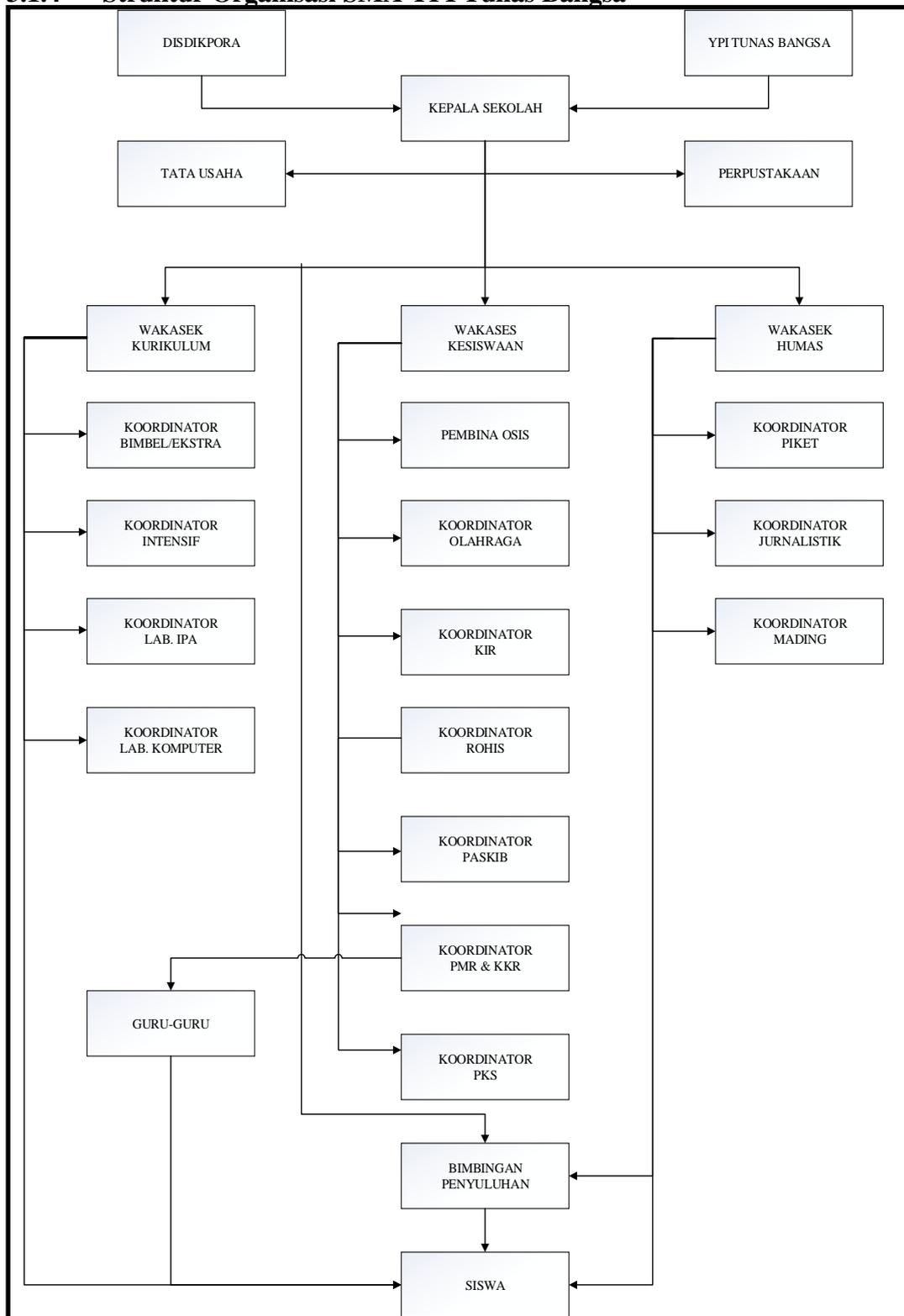
2. Melaksanakan pendidikan dengan disiplin.
3. Membentuk siswa yang berjiwa sosial dan berbudaya yang baik.
4. Membentuk siswa yang sadar akan tanggung jawab, jujur, dan mandiri.
5. Melaksanakan penguasaan teknologi informasi yang baik.
6. Membina peserta didik menjadi pribadi islam yang unggul, tangguh, dan serasi.
7. Melatih siswa untuk biasa hidup sehat baik fisik maupun mental.

TUJUAN

Tujuan pendidikan SMA YPI Tunas Bangsa Palembang adalah:

1. Menghasilkan siswa yang berilmu mandiri, disiplin, jujur, dan bertanggung jawab, berbudi luhur, dan bermoral baik untuk dirinya, agama maupun lingkungannya.
2. Menghasilkan siswa yang ulet dan mampu terjun ke masyarakat.
3. Menghasilkan siswa yang tahu ilmu pengetahuan dan teknologi yang berlandaskan ilmu agama.
4. Menghasilkan siswa yang biasa hidup sehat fisik dan mental.

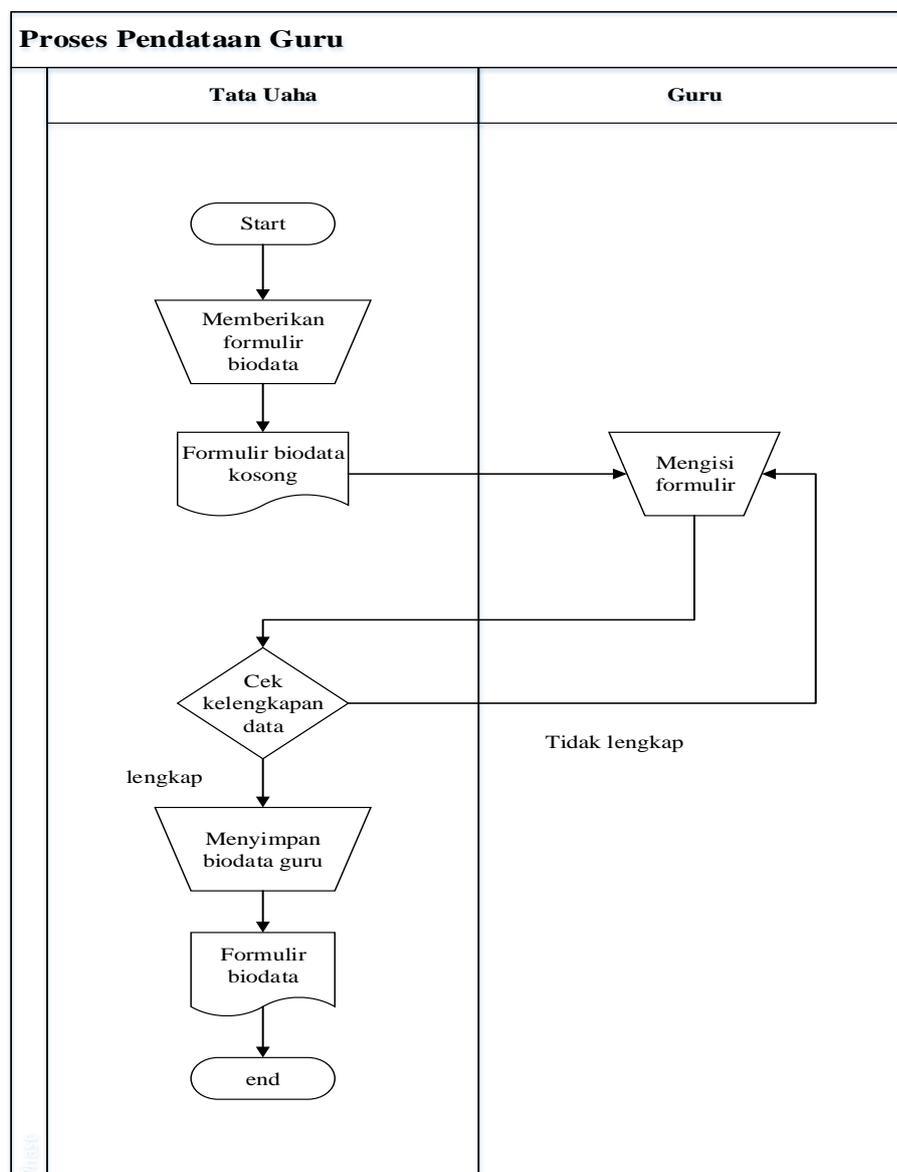
3.1.4 Struktur Organisasi SMA YPI Tunas Bangsa



Gambar 3.1 Struktur Organisasi SMA YPI Tunas Bangsa Palembang

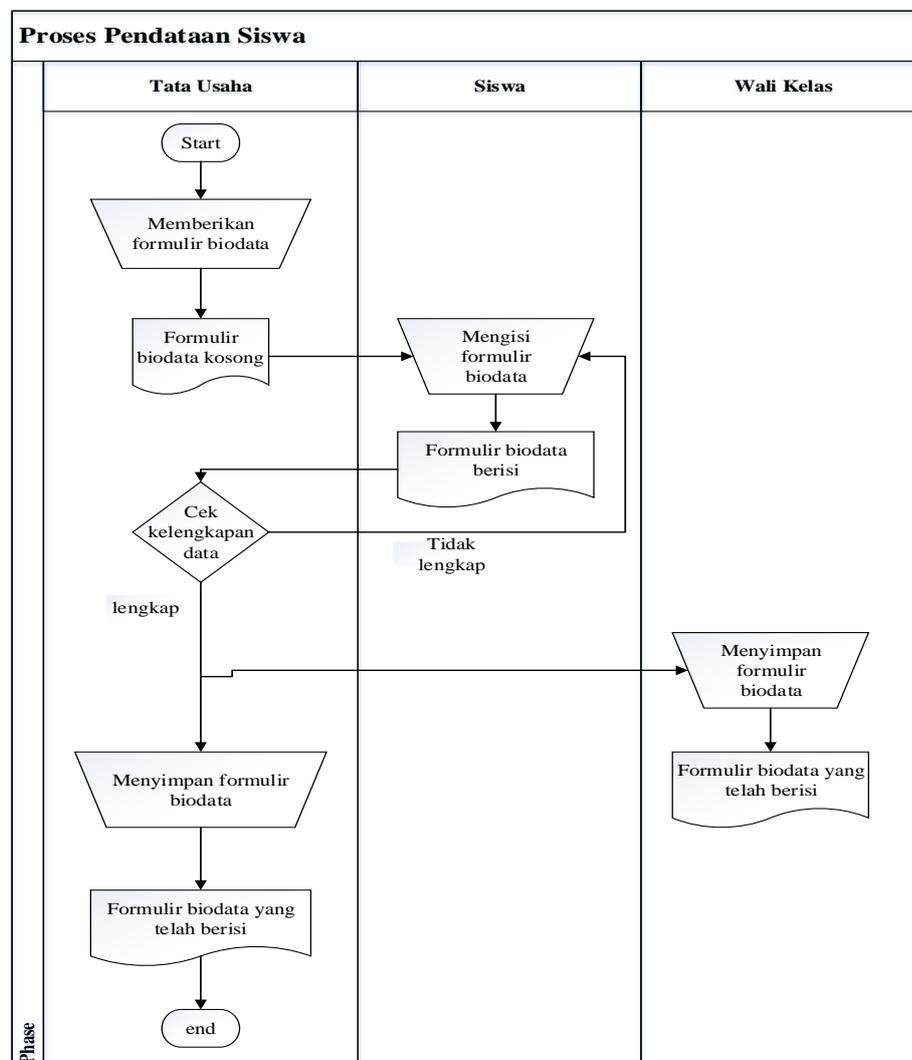
3.2 Analisis Sistem Akademik yang Sedang Berjalan di SMA YPI Tunas Bangsa Palembang

Setelah melakukan penelitian di SMA YPI Tunas Bangsa Palembang dapat diketahui bahwa sistem yang sedang berjalan di sana belum memanfaatkan teknologi yang sepenuhnya. Berikut adalah gambaran alur sistem yang sedang berjalan di SMA YPI Tunas Bangsa Palembang :



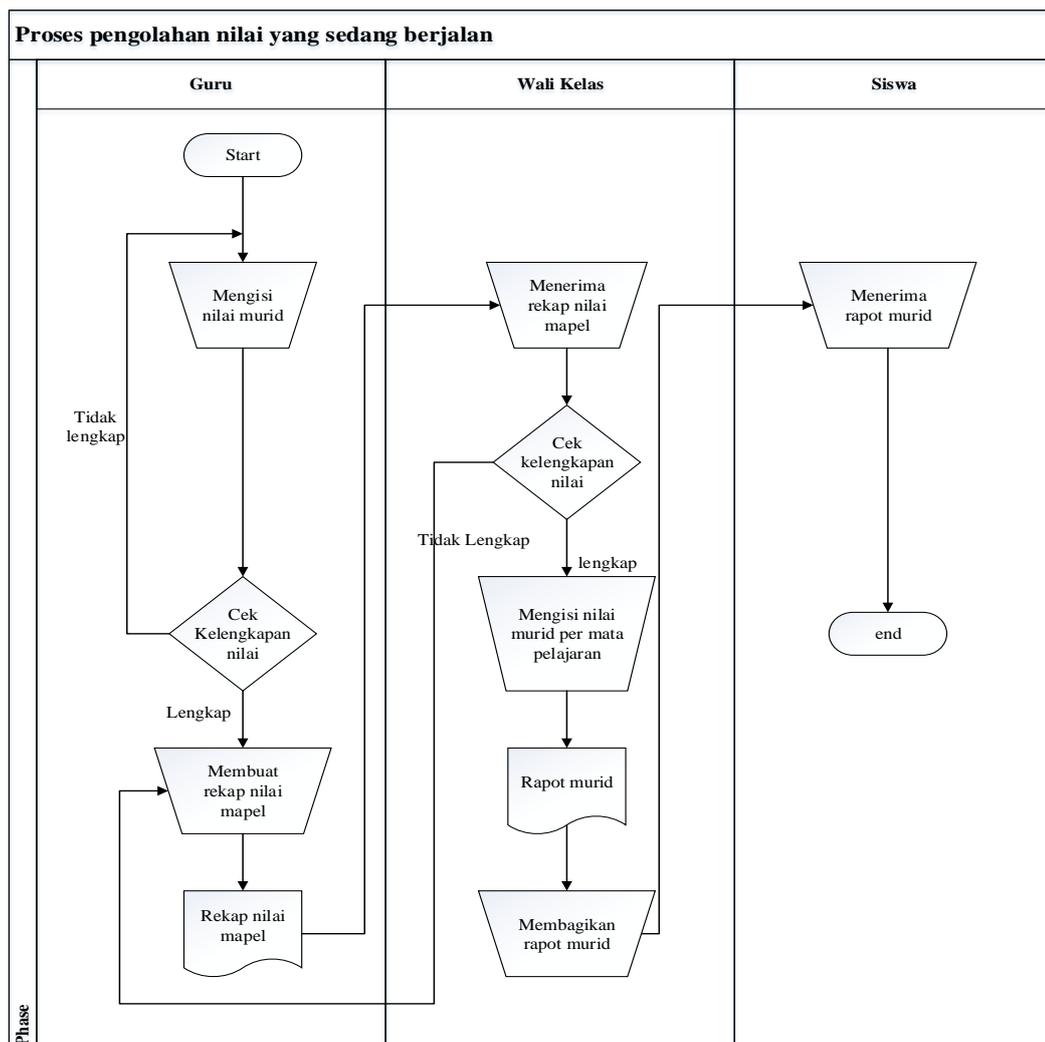
Gambar 3.2 Proses pendataan data guru yang sedang berjalan

Gambar 3.2 menjelaskan tentang proses pendataan guru yang sedang berjalan di SMA YPI Tunas Bangsa Palembang yaitu, TU memberikan formulir biodata kosong kepada guru untuk diisi, seperti yang ada di gambar 3.2. Kemudian, guru mengisi formulir tersebut dan mengembalikannya lagi ke TU. Lalu, TU memeriksa formulir tersebut, jika formulir telah benar maka TU akan menyimpan formulir biodata guru tersebut, jika formulir tersebut tidak lengkap maka TU akan mengembalikannya lagi ke guru untuk diperbaiki.



Gambar 3.3 Proses pendataan siswa yang sedang berjalan

Gambar 3.3 menjelaskan tentang proses pendataan guru yang sedang berjalan di SMA YPI Tunas Bangsa Palembang yaitu, TU memberikan formulir biodata kepada siswa untuk diisi seperti yang ada di gambar 3.3. Kemudian, siswa mengisi formulir tersebut dan mengumpulkannya ke TU. Kemudian, TU akan memeriksa formulir tersebut jika formulir tersebut telah lengkap maka TU akan memberikan kepada wali kelas untuk disimpan sebagai arsip dan TU menyimpan formulir biodata siswa yang telah diisi tersebut dan jika tidak lengkap maka formulir itu akan dikembalikan lagi ke siswa untuk diperbaiki.



Gambar 3.4 Proses pengolahan nilai siswa yang sedang berjalan

Gambar 3.4 menjelaskan tentang proses pendataan guru yang sedang berjalan di SMA YPI Tunas Bangsa Palembang yaitu, Guru mata pelajaran akan mengisi semua nilai para siswa mulai dari nilai subsumatif 1 dan 2, nilai Ujian Akhir Semester (UAS) 1 dan 2, serta nilai MID Semester 1 dan 2 seperti di gambar 3.4. Kemudian guru akan mengecek lagi nilai para siswa tersebut. Jika tidak lengkap maka guru akan memperbaiki lagi dan jika nilai telah lengkap semua maka guru akan membuatkan rekapian nilai para siswa untuk diberikan kepada wali kelas. Lalu, wali kelas akan menerima dan memeriksa kelengkapan data tersebut. Jika data tidak lengkap maka akan dikembalikan lagi ke guru untuk diperbaiki dan jika lengkap maka wali kelas akan mencetak rapot dan membagikannya kepada setiap siswa berdasarkan nama para siswa.

3.3 Analisis Kebutuhan Sistem

Untuk membuat sebuah sistem informasi dibutuhkan adanya perangkat keras dan perangkat lunak. Kemudian, sistem yang akan diusulkan juga mengikuti kebutuhan pengguna. Berikut ini adalah kebutuhan-kebutuhan yang diperlukan untuk membuat sistem informasi akademik di SMA YPI Tunas Bangsa Palembang ini:

a. Kebutuhan Perangkat Keras

Perangkat keras yang digunakan untuk memenuhi sistem dalam mengelola data. Spesifikasi minimum perangkat keras yang diperlukan untuk membuka sistem informasi ini yaitu, PC, laptop, atau smartphone dengan catatan harus terhubung dengan koneksi internet.

b. Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan untuk mendukung kegiatan dari sistem komputer dalam pembuatan sistem ini, perangkat lunak yang digunakan adalah *Dreamweaver CS6*, *Photoshop CS6*, *XAMPP*, dan *MySQL*.

c. Kebutuhan Pengguna

Admin merupakan orang yang bertugas menjalankan sistem yang merupakan bagian dari staff atau guru yang berada di kantor sekolah. Kemudian guru merupakan orang yang nantinya akan mengolah nilai-nilai dari mata pelajaran, mulai dari nilai subsumatif 1, subsumatif 2, MID Semester, dan UAS. Selanjutnya, kepala sekolah adalah orang yang memiliki akses untuk melihat semua laporan mengenai keadaan di sekolah. Dan yang terakhir adalah siswa yang akan menggunakan sistem informasi ini untuk mengetahui informasi-informasi seputar sekolah mereka dan melihat nilai mereka.

3.4 Sistem yang Diusulkan (*Elaboration/Perencanaan*)

Dengan melihat masalah yang ada, maka diperlukan sebuah sistem yang dapat memberikan informasi dengan cepat dan praktis di Sekolah SMA YPI Tunas Bangsa Palembang dengan memanfaatkan Sistem Informasi Akademik berbasis website.

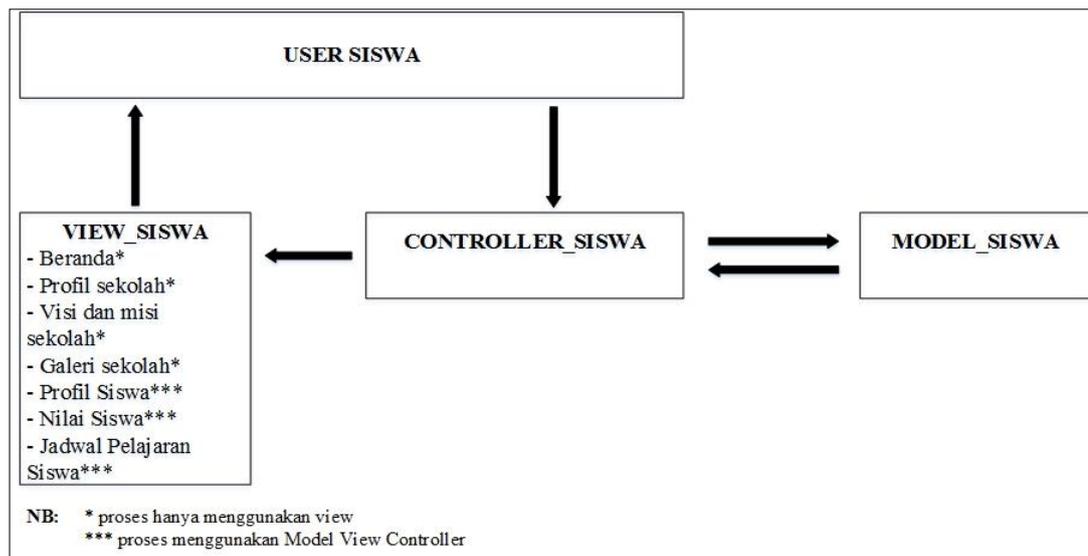
Pada sistem yang akan diusulkan ini terdapat 4 pengguna sistem, yaitu:

- a. Admin adalah orang memiliki akses untuk mengola data, seperti mengentry data guru, mengentry data siswa, mengentry data jadwal mata pelajaran, dan

mengentry berita seputar sekolah dan memiliki akses juga untuk mencetak laporan data guru, data siswa, dan data jadwal.

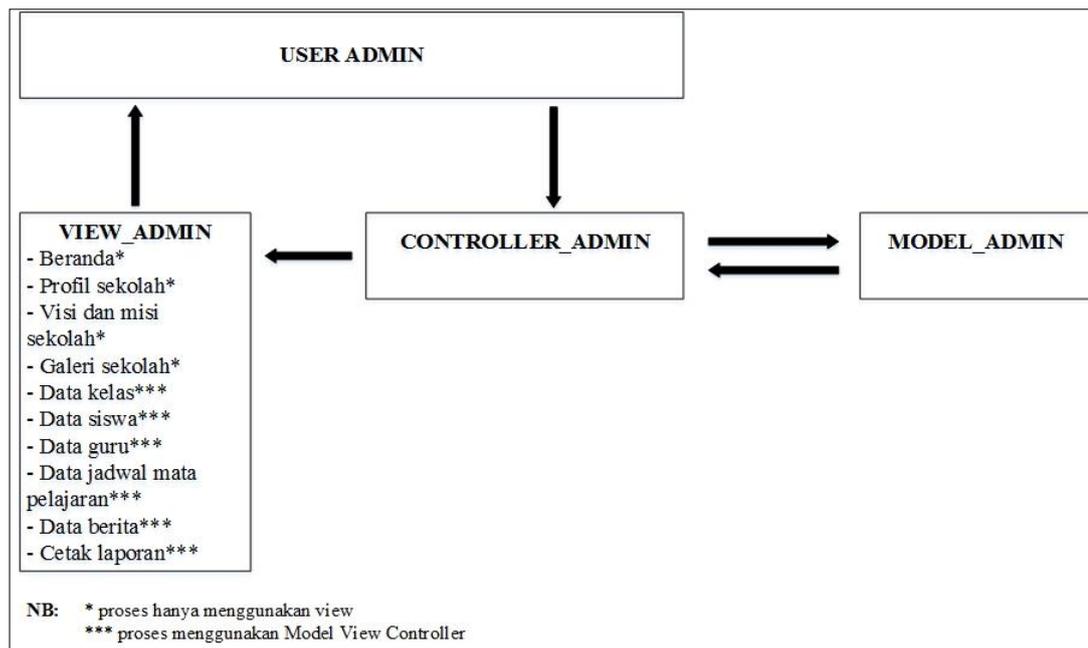
- b. Guru adalah orang yang memiliki akses untuk mengola nilai setiap mata pelajaran, mulai dari nilai subsumatif 1, subsumatif 2, MID Semester, dan Ujian Akhir Semester (UAS), guru juga dapat melihat jadwal mengajar di jadwal mata pelajaran. Guru juga dapat mencetak nilai dan jadwal.
- c. Siswa adalah orang yang dapat melihat nilai-nilai, jadwal mata pelajaran dan informasi yang ada di website. Siswa juga dapat mencetak jadwal dan nilainya.
- d. Kepala sekolah adalah orang yang memiliki hak akses untuk melihat laporan-laporan, seperti laporan keadaan sekolah mulai dari jumlah siswa, jumlah kelas, jumlah guru, serta laporan jadwal mata pelajaran. Kepala sekolah juga dapat mencetak laporan-laporan seperti data siswa, data guru, dan data jadwal.

Dengan menggunakan metode MVC (*Model View Controller*) sistem yang diusulkan dapat dibagi berdasarkan *model*, *view*, dan *controller*. Berikut ini adalah pembagiannya:



Gambar 3.5 Proses *MVC* sistem yang diusulkan untuk *user* sebagai siswa

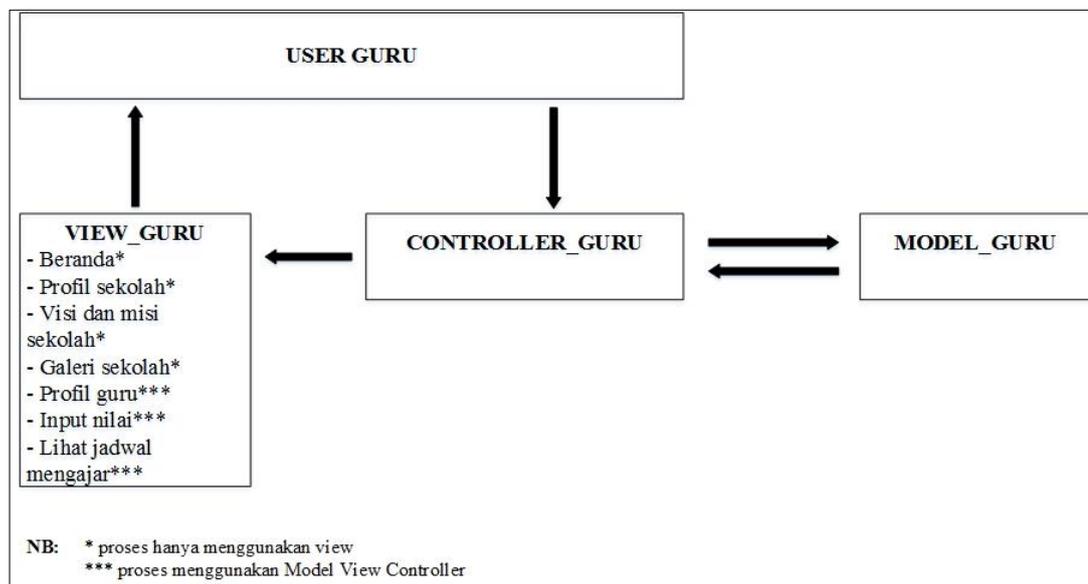
Gambar 3.5 menjelaskan bagaimana proses *MVC* pada sistem yang diusulkan untuk *user* sebagai siswa. *MVC* sendiri terbagi menjadi 3 bagian yaitu, *model*, *view*, dan *controller*. Di bagian menu untuk siswa ini, tidak semua menu ditampilkan melalui proses *model view controller* melainkan hanya ada proses yang dilakukan pada bagian *view* saja. Seperti, beranda, halaman galeri sekolah, halaman visi dan misi, dan halaman profil sekolah hanya menggunakan proses di bagian *view* saja, karena di halaman tersebut tidak membutuhkan data yang dipanggil di halaman lain. Menu yang menggunakan proses *model view controller* yaitu, profil siswa, nilai siswa, dan jadwal mata pelajaran siswa. Semua menu tersebut menggunakan metode *MVC* dalam memenuhi aktifitasnya. Profil siswa, nilai siswa, dan jadwal mata pelajaran siswa menggunakan *model_siswa* sebagai tempat menyimpan datanya, kemudian *controller_siswa* yang mengatur data agar dapat tampil di *view_siswa*.



Gambar 3.6 Proses *MVC* sistem yang diusulkan untuk *user* sebagai admin

Gambar 3.6 menjelaskan bagaimana proses *MVC* pada sistem yang diusulkan untuk *user* sebagai guru. *MVC* sendiri terbagi menjadi 3 bagian yaitu, *model*, *view*, dan *controller*. Di bagian menu untuk admin ini, tidak semua menu ditampilkan melalui proses *model view controller* melainkan hanya ada proses yang dilakukan pada bagian *view* saja. Seperti, beranda, halaman galeri sekolah, halaman visi dan misi, dan halaman profil sekolah hanya menggunakan proses di bagian *view* saja, karena di halaman tersebut tidak membutuhkan data yang dipanggil di halaman lain. Menu yang menggunakan proses *model view controller* yaitu, data kelas, data jadwal, data mapel, data guru, data siswa, data berita, serta cetak laporan. Semua menu tersebut menggunakan metode *MVC* dalam memenuhi aktifitasnya. Beranda, data kelas, data jadwal, data mapel, data guru, data siswa, data berita, serta cetak laporan menggunakan *model_admin* sebagai tempat

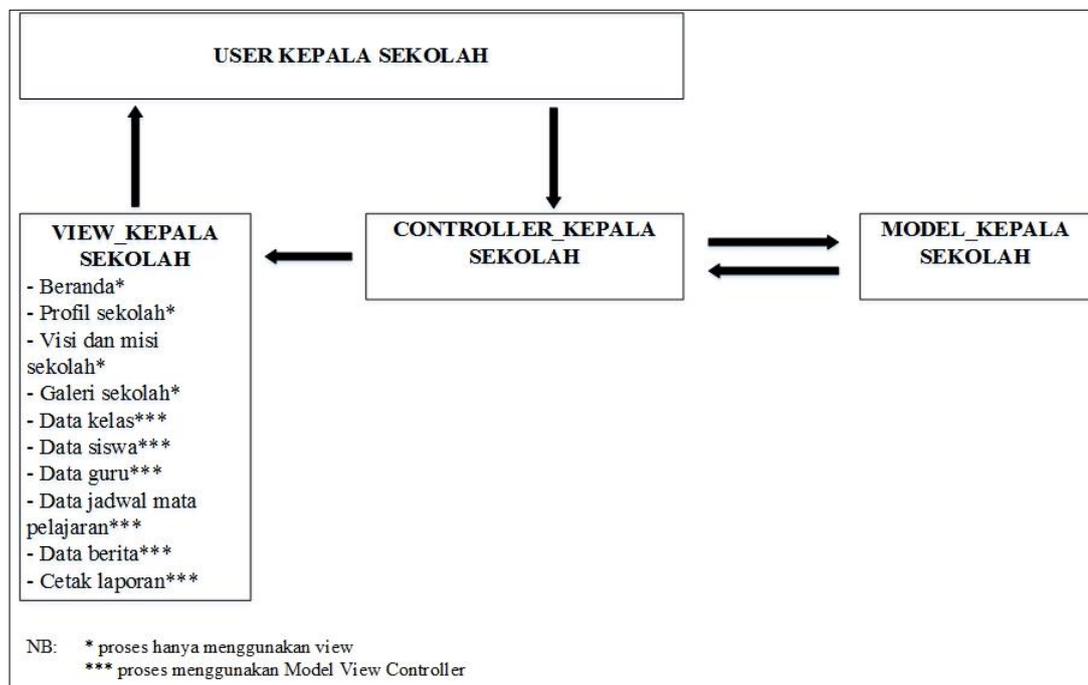
menyimpan datanya, kemudian *controller_admin* yang mengatur data agar dapat tampil di *view_admin*.



Gambar 3.7 Proses MVC sistem yang diusulkan untuk *user* sebagai guru

Gambar 3.7 menjelaskan bagaimana proses MVC pada sistem yang diusulkan untuk *user* sebagai guru. MVC sendiri terbagi menjadi 3 bagian yaitu, *model*, *view*, dan *controller*. Di bagian menu untuk guru ini, tidak semua menu ditampilkan melalui proses *model view controller* melainkan hanya ada proses yang dilakukan pada bagian *view* saja. Seperti, beranda, halaman galeri sekolah, halaman visi dan misi, dan halaman profil sekolah hanya menggunakan proses di bagian *view* saja, karena di halaman tersebut tidak membutuhkan data yang di panggil di halaman lain. Menu yang menggunakan proses *model view controller* yaitu, profil guru, input nilai, dan lihat jadwal mengajar. Semua menu tersebut menggunakan metode MVC dalam memenuhi aktifitasnya. Beranda, profil guru, input nilai, dan lihat jadwal mengajar menggunakan *model_guru* sebagai tempat

menyimpan datanya, kemudian *controller_guru* yang mengatur data agar dapat tampil di *view_guru*.



Gambar 3.8 Proses *MVC* sistem yang diusulkan untuk *user* sebagai kepala sekolah

Gambar 3.8 menjelaskan bagaimana proses *MVC* pada sistem yang diusulkan untuk *user* sebagai kepala sekolah. *MVC* sendiri terbagi menjadi 3 bagian yaitu, *model*, *view*, dan *controller*. Di bagian menu untuk kepala sekolah ini, tidak semua menu ditampilkan melalui proses *model view controller* melainkan hanya ada proses yang dilakukan pada bagian *view* saja. Seperti, beranda, halaman galeri sekolah, halaman visi dan misi, dan halaman profil sekolah hanya menggunakan proses di bagian *view* saja, karena di halaman tersebut tidak membutuhkan data yang dipanggil di halaman lain. Menu yang menggunakan proses *model view controller* yaitu, data kelas, data jadwal, data mapel, data guru, data siswa, data nilai, data berita, serta cetak laporan. Semua

menu tersebut menggunakan metode *MVC* dalam memenuhi aktifitasnya. Data kelas, data guru, data siswa, data mapel, data jadwal, data nilai, dan cetak laporan menggunakan *model_kepala_sekolah* sebagai tempat menyimpan datanya, kemudian *controller_kepala_sekolah* yang mengatur data agar dapat tampil di *view_kepala_sekolah*.

3.5 Rancangan Sistem Informasi Akademik di SMA YPI Tunas Bangsa Palembang yang akan diusulkan.

Perancangan sistem informasi akademik di SMA YPI Tunas Bangsa Palembang yang diusulkan ini menggunakan sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemograman berorientas objek. (Rosa dan Shalahuddin, 2014:137). Sistem informasi akademik di SMA YPI Tunas Bangsa Palembang ini sendiri akan dirancang hanya menggunakan 3 macam diagram dari UML (*unified modelling language*) yaitu, *use case diagram*, *class diagram*, dan *activity diagram*.

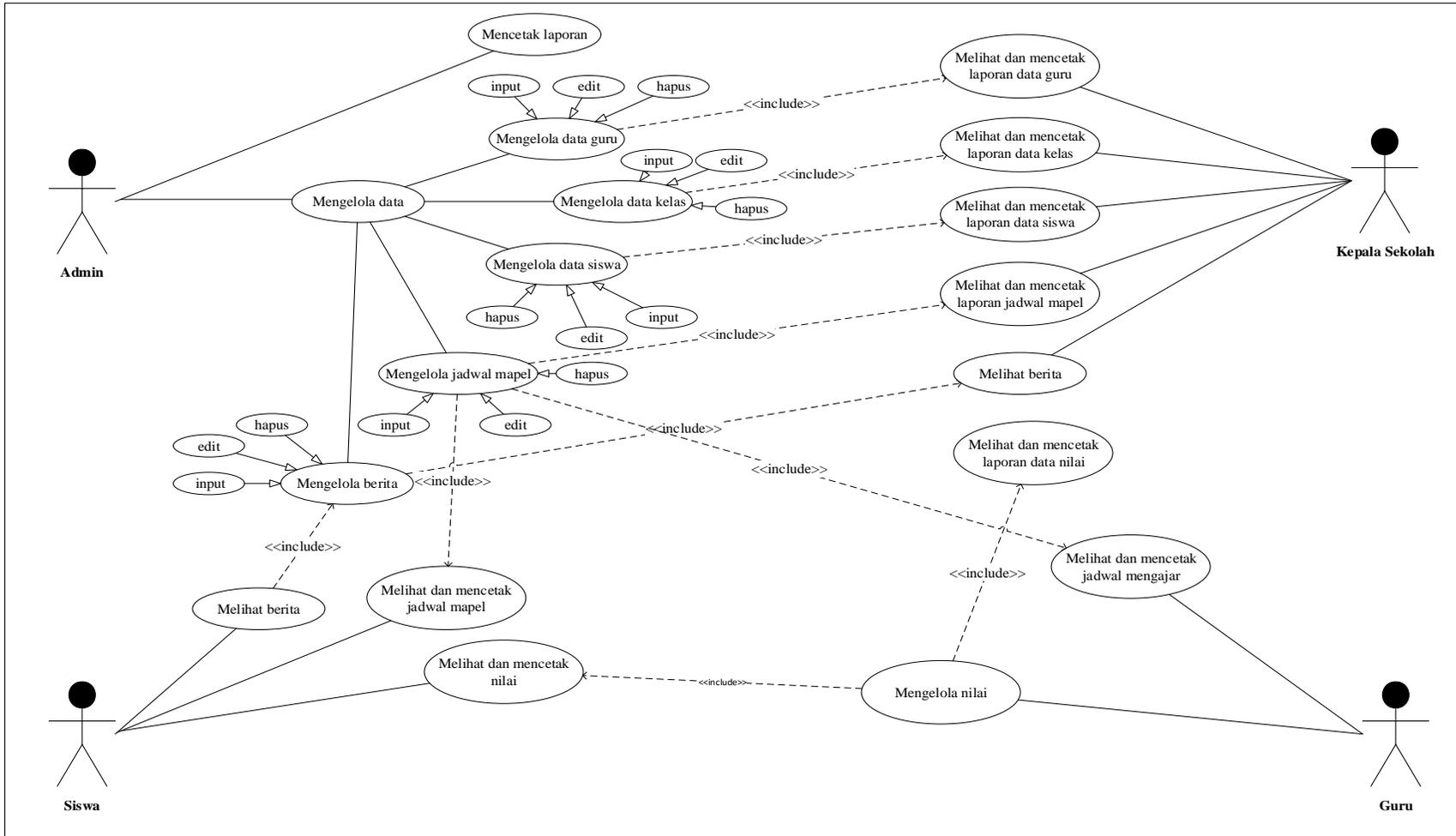
3.5.1 Perancangan Use Case Diagram Sistem Informasi Akademik SMA YPI Tunas Bangsa Palembang.

Use case akan memberikan hasil tertentu bagi actor, hal ini dilakukan untuk mennetukan kebutuhan fungsional dan inforasi yang diperlukan oleh pengguna. Suatu *use case* mempresentasikan sebuah interaksi antara aktor dengan sistem. Setiap *use case* yang diusulkan dapat dilihat pada table sehingga dapat mempermudah identifikasi pada implementasi dan penyusunan dokumen. Berikut daftar identifikasi aktor sistem yang diusulkan dapat dilihat pada table 3.1.

Tabel 3.1 Tabel Identifikasi aktor

No	Aktor	Deskripsi
1.	Admin	Orang yang bertanggung jawab mengelola data-data dalam system
2.	Guru	Orang yang menggunakan sistem untuk keperluan mengolah dan menyimpan nilai para siswa serta melihat dan mencetak jadwal pelajaran
3.	Kepala sekolah	Orang yang menggunakan sistem untuk keperluan untuk melihat dan mencetak laporan-laporan, seperti laporan keadaan sekolah mulai dari jumlah siswa, jumlah kelas, jumlah guru, laporan nilai serta laporan jadwal mata pelajaran.
4.	Siswa	Orang yang menggunakan sistem untuk keperluan melihat informasi seputar sekolah dan mencetak nilai dan jadwal pelajaran mereka

Setelah aktor dan *use case* teridentifikasi, diagram *use case* dapat digunakan untuk menggambarkan secara grafis interaksi aktor dan *use case* yang terlihat pada gambar 3.9



Gambar3.9 Use case diagram Sistem informasi akademik SMA YPI Tunas Bangsa Palembang

Tabel 3.2 Tabel Definisi *use case*

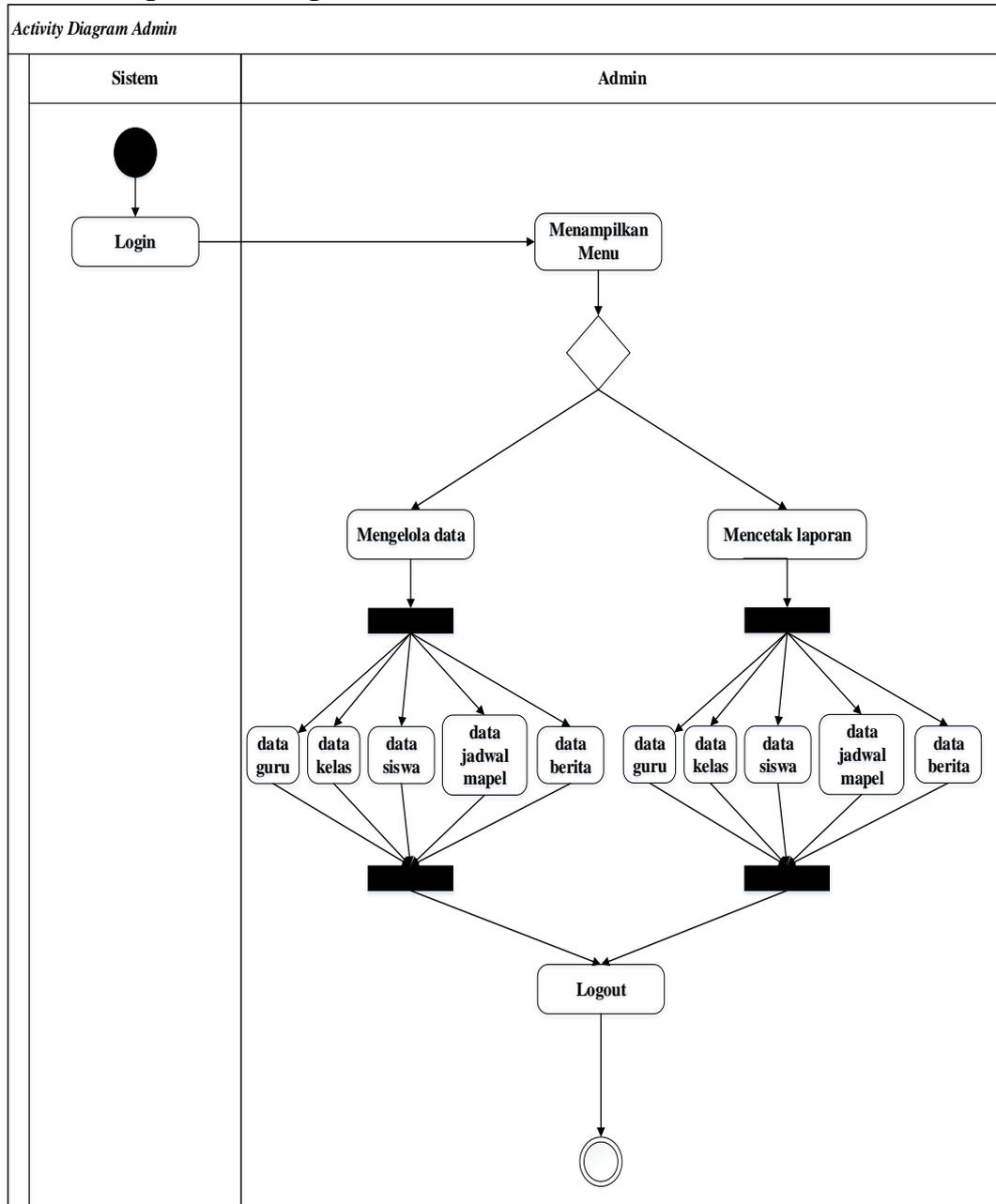
No	<i>Use Case</i>	Deskripsi	Aktor
1.	Entry Data	Merupakan proses memasukkan data guru, siswa, kepala sekolah, data kelas, jadwal mata pelajaran dan nilai ke dalam basis data	Admin dan guru
2.	Merubah data	Merupakan proses merubah data guru, siswa, kepala sekolah, jadwal pelajaran, data kelas dan nilai	Admin dan guru
3.	Menghapus data	Merupakan proses menghapus data guru, siswa, kepala sekolah, jadwal mata pelajaran, data kelas dan nilai	Admin dan guru
4.	Mengelola data guru	Merupakan proses memasukkan, mengedit, menghapus, dan menambahkan data guru ke dalam database	Admin
5.	Mengelola data kelas	Merupakan proses memasukkan, mengedit, menghapus, dan menambahkan data kelas ke dalam database	Admin
6.	Mengelola data siswa	Merupakan proses memasukkan, mengedit, menghapus, dan menambahkan data siswa ke dalam database	Admin
7.	Mengelola jadwal mapel	Merupakan proses memasukkan, mengedit, menambahkan, dan menghapus jadwal mata pelajaran ke dalam sistem untuk dapat dilihat oleh siswa, guru, dan kepala sekolah.	Admin
8.	Mengelola berita	Merupakan proses memasukkan, menambahkan, mengedit, dan menghapus data berita ke dalam basis data	Admin
9.	Melihat berita	Merupakan proses melihat	Admin, siswa,

		berita yang ada di dalam basis data.	guru, dan kepala sekolah
10.	Melihat jadwal pelajaran	Merupakan proses melihat jadwal pelajaran yang ada di dalam basis data.	Guru, siswa, dan kepala sekolah.
11.	Melihat nilai	Merupakan proses melihat nilai akhir yang ada di dalam basis data	Siswa
12.	Melihat dan mencetak jadwal mengajar	Merupakan proses melihat jadwal pelajaran yang ada di dalam basis data. Kemudian, dapat mencetak jadwal mengajar tersebut.	Guru dan kepala sekolah
13.	Mengelola nilai	Merupakan proses memasukkan, menambahkan, menghapus, dan mengedit nilai siswa ke dalam database	Guru
14.	Melihat laporan guru	Merupakan proses melihat data guru yang ada di dalam basis data. Kemudian, mencetak data guru tersebut.	Kepala sekolah
15.	Melihat laporan dan mencetak kelas	Merupakan proses melihat data kelas yang ada di dalam basis data. Kemudian, dapat mencetak data kelas yang ada	Kepala sekolah
16.	Melihat dan mencetak laporan siswa	Merupakan proses melihat data siswa yang ada di dalam basis data. Kemudian dapat mencetak data siswa tersebut	Kepala sekolah
17.	Melihat dan mencetak laporan jadwal pelajaran	Merupakan proses melihat jadwal mata pelajaran yang ada di dalam basis data. Kemudian kepala sekolah juga dapat mencetak jadwal pelajaran tersebut	Kepala sekolah

3.5.2 Perancangan *Class Diagram* Sistem Informasi Akademik SMA YPI Tunas Bangsa Palembang

Pada Sistem Informasi Akademik ini terdapat *class controller_siswa*, *controller_guru*, *controller_admin*, *controller_kepsek*, *model_nilai*, *model_guru*, *model_mapel*, *model_berita*, *model_jadwal*, *model_siswa*, *model_kelas*, *view* untuk melihat data-data seperti data jadwal, data nilai, data guru, data kelas, data mapel, data berita, data siswa. Notasi *class* berbentuk persegi panjang dengan 3 bagian yaitu, bagian paling atas yaitu untuk nama *class*, bagian tengah yaitu untuk nama *method*, dan bagian paling bawah untuk *atribute*. Berikut adalah *class diagram* sistem informasi akademik yang dibangun oleh peneliti.

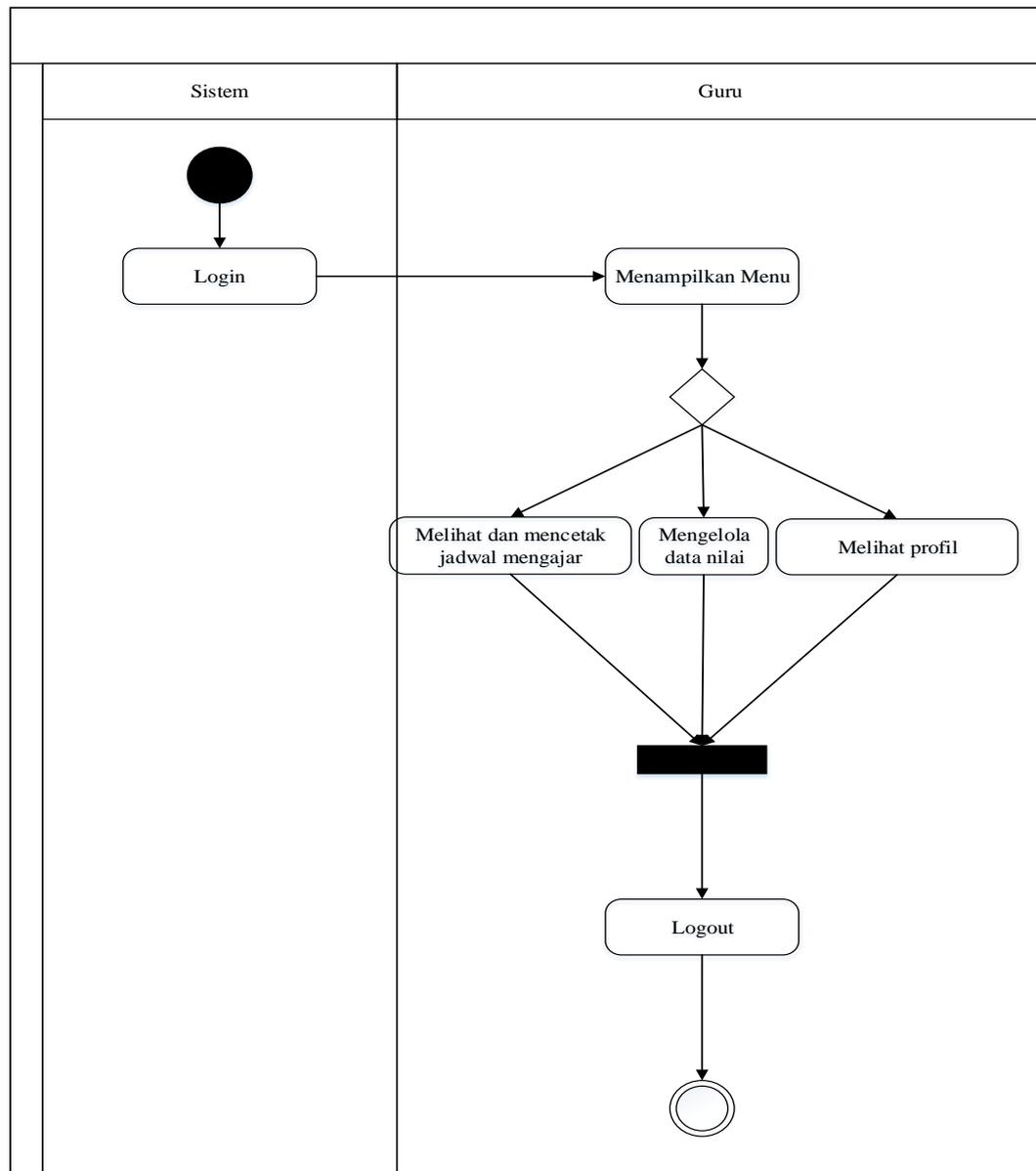
3.5.3 Perancangan *Activity Diagram* Sistem Informasi Akademik SMA YPI Tunas Bangsa Palembang



Gambar 3.11 *Activity diagram* sistem informasi akademik *user* admin

Gambar 3.11 menjelaskan proses *activity diagram* sistem informasi akademik *user* admin. Setelah admin melakukan *login* ke dalam sistem, sistem akan merespon dengan menampilkan pilihan menu untuk admin, mulai dari menu data kelas, data guru, data siswa, data mata pelajaran, data jadwal, data berita, dan

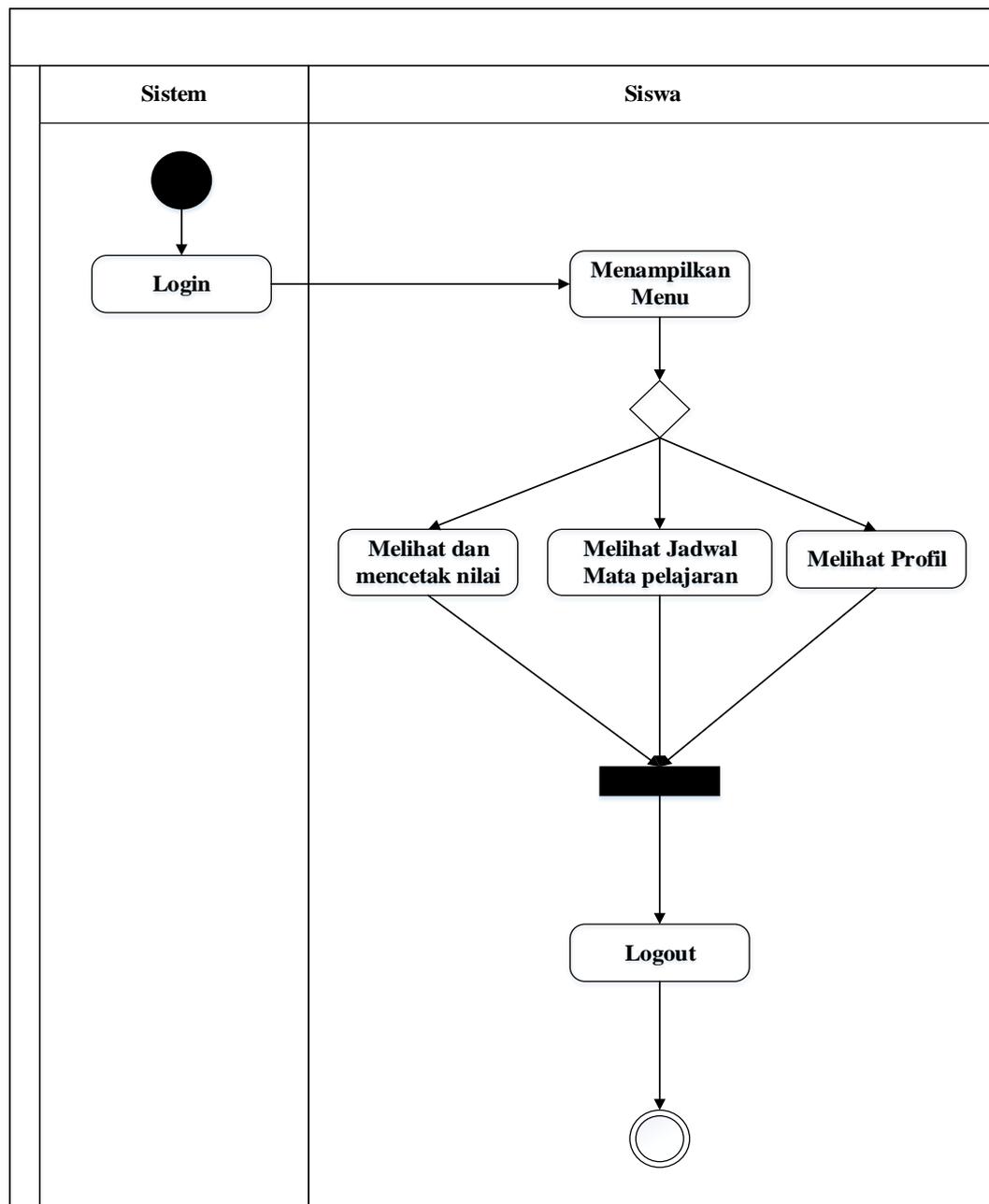
mencetak laporan. Jika admin telah selesai melakukan aktifitasnya, admin dapat *logout* dari sistem dengan cara menekan tombol *logout*.



Gambar 3.12 Activity diagram sistem informasi akademik user guru

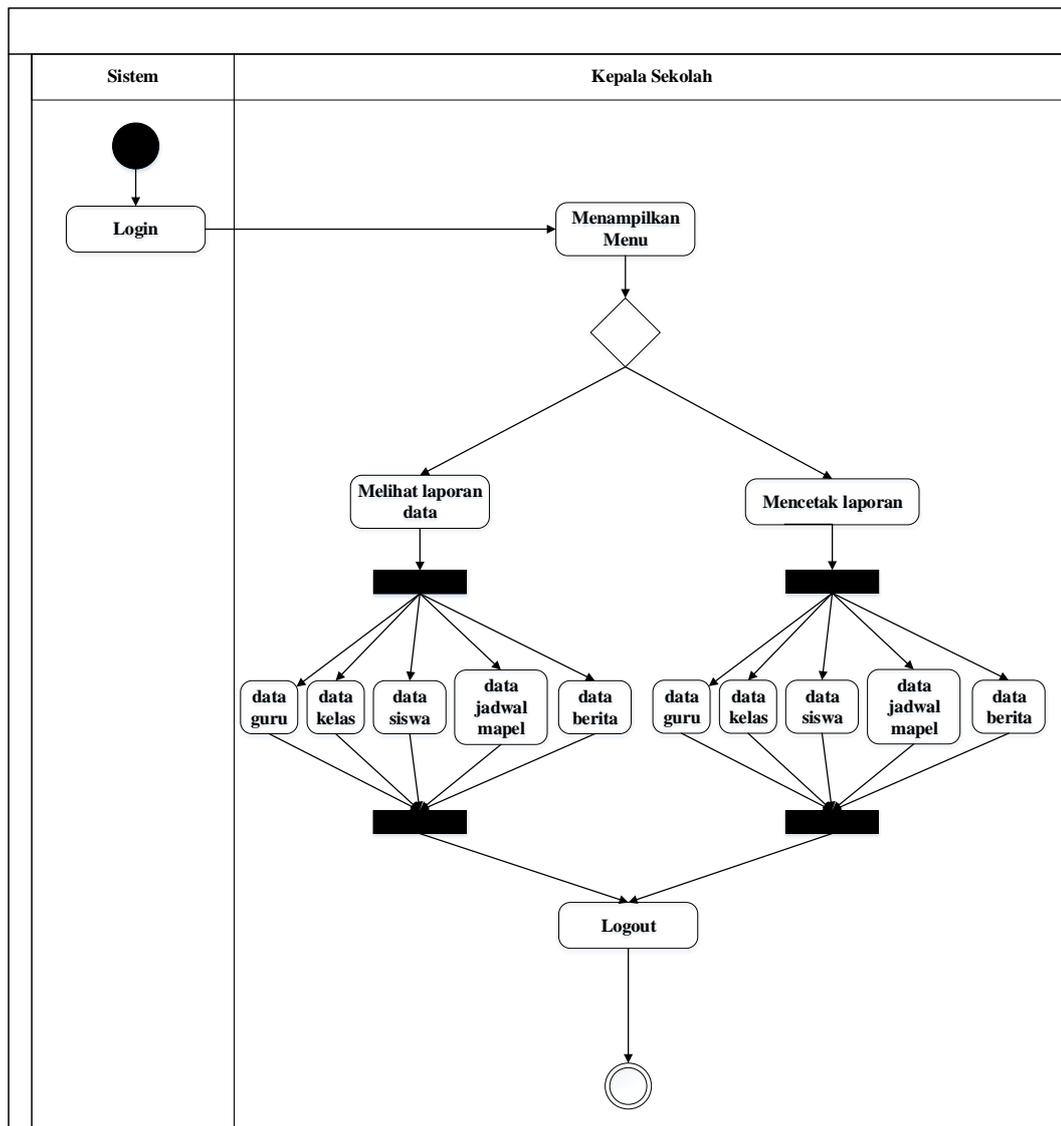
Gambar 3.12 menjelaskan proses *activity diagram* sistem informasi akademik *user* guru. Setelah guru melakukan *login* ke sistem dengan cara memasukkan *username* dan *password* dengan benar, maka sistem akan merespon

dengan cara menampilkan menu-menu untuk admin. Menu-menu admin tersebut adalah, menu mengelola data nilai, melihat jadwal, dan melihat profil. Guru dapat menginputkan nilai siswa di bagian menu data nilai, setelah selesai guru dapat *logout* dari sistem.



Gambar 3.13 Activity diagram sistem informasi akademik user siswa

Gambar 3.13 menjelaskan proses *activity diagram* sistem informasi akademik *user* siswa. Setelah siswa melakukan *login* dengan cara memasukkan *username* dan *password* dengan benar, maka sistem akan merespon dengan menampilkan menu-menu untuk siswa. Mulai dari menu melihat dan mencetak nilai, melihat dan mencetak jadwal serta melihat profil siswa itu sendiri.



Gambar 3.14 *Activity diagram* sistem informasi akademik *user* kepala sekolah

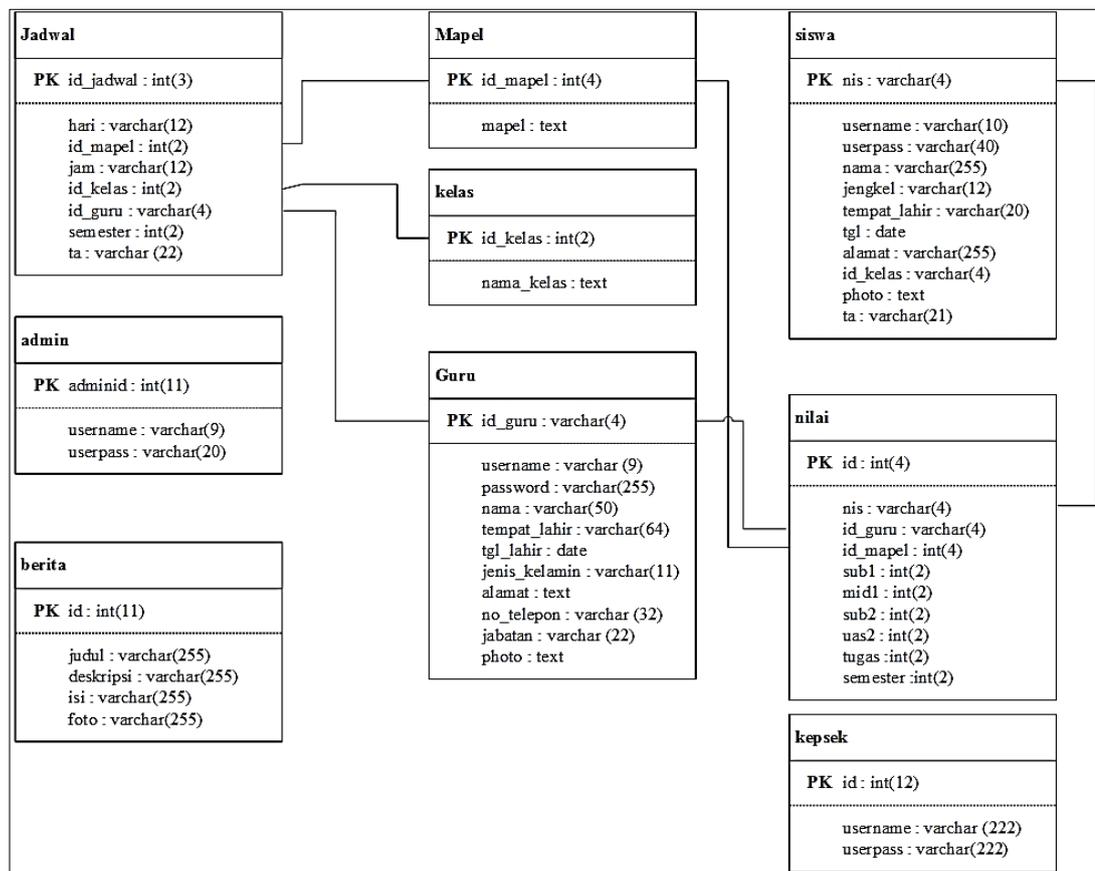
Gambar 3.14 menjelaskan proses *activity diagram* sistem informasi akademik *user* kepala sekolah. Setelah kepala sekolah melakukan *login* ke dalam

sistem dengan memasukkan *username* dan *password* dengan benar, sistem akan merespon dengan cara menampilkan menu-menu untuk kepala sekolah. Sistem akan menampilkan menu, mulai dari menu data guru, data kelas, data siswa, data jadwal, data berita, serta kepala sekolah dapat mencetak semua laporan data-data tersebut.

3.5.4 Perancangan Struktur *Database* Sistem Informasi Akademik di SMA YPI Tunas Bangsa Palembang

3.5.4.1 Rancangan Relasi Antar Tabel

Perancangan relasi antar tabel digunakan untuk melihat relasi-relasi antar tabel yang ada di dalam *database*. *Database* digunakan untuk menampung tabel-tabel yang akan dibuat.



Gambar 3.15 Relasi tabel *database* Sistem Informasi Akademik SMA YPI Tunas Bangsa Palembang

3.5.4.2 Rancangan Tabel *Database*

Membuat sebuah *database* dengan nama akademik dengan 8 tabel merupakan tabel rancangan sistem. Berikut ini adalah tabel yang akan dibuat untuk pembuatan sistem informasi yang sedang dirancang di SMA YPI Tunas Bangsa Palembang:

a. Tabel Admin

Tabel admin adalah tabel yang digunakan untuk menyimpan data admin yang mana isi *field* dari tabel admin dapat dilihat di tabel 3.4.

Tabel 3.4 Tabel Admin

Nama Field	Tipe Data	Keterangan
Adminid	Int (11)	<i>Primary Key</i>
Username	Varchar (9)	
Userpass	Varchar (255)	

b. Tabel Mapel

Tabel mapel adalah tabel yang digunakan untuk menyimpan data mapel yang mana isi *field* dari tabel mapel dapat dilihat di tabel 3.5.

Tabel 3.5 Tabel Mapel

Nama Field	Tipe Data	Keterangan
Id_mapel	Int (4)	<i>Primary Key</i>
Mapel	Text	

c. Tabel kelas

Tabel kelas adalah tabel yang digunakan untuk menyimpan data kelas yang mana isi *field* dari tabel kelas dapat dilihat di tabel 3.6.

Tabel 3.6 Tabel kelas

Nama Field	Tipe Data	Keterangan
Id_kelas	Int (2)	<i>Primary Key</i>
Nama_kelas	Text	

d. Tabel Guru

Tabel guru adalah tabel yang digunakan untuk menyimpan data guru yang mana isi *field* dari tabel kelas dapat dilihat di tabel 3.7.

Tabel 3.7 Tabel guru

Nama Field	Tipe Data	Keterangan
Id_guru	Varchar (4)	<i>Primary Key</i>
Username	Varchar (9)	
Password	Varchar (255)	
Nama	Varchar (50)	
Tempat_lahir	Varchar (64)	
Tgl_lahir	Date	
Jenis_kelamin	Varchar (11)	
Alamat	Text	
No_telpon	Varchar (32)	
Pend_terakhir	Varchar (22)	
Jabatan	Varchar (22)	

e. Tabel siswa

Tabel siswa adalah tabel yang digunakan untuk menyimpan data siswa yang mana isi *field* dari tabel siswa dapat dilihat di tabel 3.8.

Tabel 3.8 Tabel siswa

Nama Field	Tipe Data	Keterangan
NIS	Varchar (4)	<i>Primary Key</i>
Username	Varchar (10)	
Userpass	Varchar (40)	
Nama	Varchar (255)	
Id_kelas	Int (2)	
Tempat_lahir	Varchar (20)	
Tgl	Date	
Jengkel	Varchar (12)	
Alamat	Varchar (255)	
Ta	Varchar (22)	
Photo	Text	

f. Tabel nilai

Tabel nilai adalah tabel yang digunakan untuk menyimpan data nilai yang mana isi *field* dari tabel kelas dapat dilihat di tabel 3.9.

Tabel 3.9 Tabel nilai

Nama Field	Tipe Data	Keterangan
NIS	Varchar (4)	
Id	Int (4)	<i>Primary Key</i>
Id_guru	Varchar (4)	
Id_mapel	Int (4)	
Semester	Int (2)	
Sub1	Int (2)	
Mid1	Int (2)	
Sub 2	Int (2)	
Uas2	Int (2)	
Tugas	Int (2)	

g. Tabel Jadwal

Tabel jadwal adalah tabel yang digunakan untuk menyimpan data jadwal yang mana isi *field* dari tabel kelas dapat dilihat di tabel 3.10.

Tabel 3.10 Tabel jadwal

Nama Field	Tipe Data	Keterangan
Id_jadwal	Varchar (3)	<i>Primary Key</i>
Hari	Varchar (12)	
Id_mapel	Int (2)	
Id_kelas	Int (2)	
Jam	Varchar (12)	
Id_guru	Varchar (4)	
Semester	Int (2)	
Ta	Varchar(22)	

h. Tabel Kepsek

Tabel kepek adalah tabel yang digunakan untuk menyimpan data kepek yang mana isi *field* dari tabel kelas dapat dilihat di tabel 3.11.

Tabel 3.11 Tabel kepek

Nama Field	Type Data	Keterangan
Id	Int (12)	<i>Primary Key</i>
Username	Varchar (222)	
Userpass	Varchar (222)	

i. Tabel Berita

Tabel berita adalah tabel yang digunakan untuk menyimpan data berita yang mana isi *field* dari tabel kelas dapat dilihat di tabel 3.12.

Tabel 3.12 Tabel Berita

Nama Field	Type Data	Keterangan
Id	Int (11)	<i>Primary Key</i>
Judul	Varchar (255)	
Deskripsi	Varchar (255)	
Isi	Varchar (255)	
Foto	Varchar (55)	

3.5.5 Perancangan Antarmuka

The wireframe shows a web interface with the following structure:

- HEADER:** A horizontal bar containing navigation links: HOME, PROFIL, SEJARAH, VISI DAN MISI, GALERI.
- Main Content Area:**
 - On the left side, there is a login form with input fields for USERNAME and PASSWORD, a HAK AKSES section, and buttons for LOGIN and RESET.
 - The right side of the main content area is a large white box labeled BERITA, intended for displaying news content.
- FOOTER:** A horizontal bar at the bottom of the page.

Gambar 3.16 Rancangan *Interface* halaman beranda

Gambar 3.16 adalah gambar rancangan untuk halaman beranda. Di halaman beranda tersebut terdapat menu home, profil, sejarah, visi dan misi, serta

galeri tentang sekolah. Terdapat juga tempat untuk login ke dalam website tersebut dengan memasukkan *username* dan *password* yang benar. Ada juga form tempat menginformasikan berita tentang sekolah tersebut.

HEADER								
PROFIL	NILAI	JADWAL	LOGOUT					
NIS : NAMA : KELAS :								
SEMESTER		TAMPILKAN						
		SEMESTER					TOTAL	RATA-RATA
		SUB 1	MID 1	SUB 2	TUGAS	UAS 1		
CETAK								
FOOTER								

Gambar 3.17 rancangan *interface* halaman nilai siswa

Gambar 3.17 adalah gambar rancangan *interface* untuk halaman nilai siswa. Di halaman tersebut siswa dapat melihat nilainya berdasarkan semester. Kemudian, siswa tersebut dapat mencetak nilainya dengan cara menekan tombol cetak.

HEADER					
PROFIL	NILAI	JADWAL	LOGOUT		
NO	HARI	JAM	MATA PELAJARAN	GURU	KELAS
XXX	XXXXX	XXXXXX	XXXXX	XXXXXXXX	XX
XXX	XXXXX	XXXXXX	XXXXX	XXXXXXXX	XX
XXX	XXXXX	XXXXXX	XXXXX	XXXXXXXX	XX
<input type="button" value="CETAK"/>					
FOOTER					

Gambar 3.18 rancangan *interface* halaman jadwal pelajaran siswa

Gambar 3.18 adalah gambar rancangan *interface* halaman jadwal pelajaran untuk siswa. Di halaman tersebut siswa dapat melihat jadwal pelajarannya dan kemudian mencetaknya jika ingin mencetaknya dengan cara menekan tombol cetak.

HEADER									
			PROFIL	NILAI	JADWAL	LOGOUT			
<input type="button" value="PILIHAN KELAS"/>				<input type="button" value="TAMPILKAN"/>					
NO	NIS	NAMA	NILAI SUB 1	NILAI MID	NILAI SUB 2	NILAI TUGAS	NILAI UJIAN	SEMESTER	RATA-RATA
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	XXXX
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	XXXX
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	XXXX
<input type="button" value="SIMPAN"/>									
FOOTER									

Gambar 3.19 gambar rancangan *interface* halaman guru input nilai siswa

Gambar 3.19 adalah gambar rancangan *interface* halaman guru untuk menginputkan nilai siswa. Setelah guru memilih kelas, maka akan tampil semua siswa yang ada di dalam kelas tersebut, kemudian guru dapat menginputkan nilai untuk semua siswa yang ada berdasarkan semester yang di pilih guru.

HEADER					
	PROFIL	NILAI	JADWAL	LOGOUT	
DATA JADWAL					
SELAMAT DATANG XXXXX (NAMA GURU)					
NO	HARI	JAM	MATA PELAJARAN	GURU	KELAS
XXX	XXXXX	XXXXXX	XXXXX	XXXXXXXX	XX
XXX	XXXXX	XXXXXX	XXXXX	XXXXXXXX	XX
<div style="display: inline-block; border: 1px solid black; padding: 2px 5px;">CETAK</div>					
FOOTER					

Gambar 3.20 gambar rancangan *interface* halaman jadwal mengajar guru

Gambar 3.20 adalah gambar rancangan *interface* untuk halaman jadwal mengajar guru. Setelah guru login, guru dapat melihat jadwal mengajarnya sendiri, di hari, kelas, jam, mana dia mengajar. Kemudian, guru juga dapat mencetaknya dengan cara menekan tombol cetak.

HEADER																											
KELAS	DATA GURU	DATA SISWA	LIST JADWAL	LIST MAPEL	BERITA	LAPORAN	LOGOUT																				
DATA KELAS <input type="button" value="TAMBAH KELAS"/>																											
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>NO</th> <th>ID KELAS</th> <th>NAMA KELAS</th> <th colspan="2">ACTION</th> </tr> </thead> <tbody> <tr> <td>XX</td> <td>XXX</td> <td>XXX</td> <td><input type="button" value="EDIT"/></td> <td><input type="button" value="HAPUS"/></td> </tr> <tr> <td>XX</td> <td>XXX</td> <td>XXX</td> <td><input type="button" value="EDIT"/></td> <td><input type="button" value="HAPUS"/></td> </tr> <tr> <td>XX</td> <td>XXX</td> <td>XXX</td> <td><input type="button" value="EDIT"/></td> <td><input type="button" value="HAPUS"/></td> </tr> </tbody> </table>								NO	ID KELAS	NAMA KELAS	ACTION		XX	XXX	XXX	<input type="button" value="EDIT"/>	<input type="button" value="HAPUS"/>	XX	XXX	XXX	<input type="button" value="EDIT"/>	<input type="button" value="HAPUS"/>	XX	XXX	XXX	<input type="button" value="EDIT"/>	<input type="button" value="HAPUS"/>
NO	ID KELAS	NAMA KELAS	ACTION																								
XX	XXX	XXX	<input type="button" value="EDIT"/>	<input type="button" value="HAPUS"/>																							
XX	XXX	XXX	<input type="button" value="EDIT"/>	<input type="button" value="HAPUS"/>																							
XX	XXX	XXX	<input type="button" value="EDIT"/>	<input type="button" value="HAPUS"/>																							
FOOTER																											

Gambar 3.21 gambar rancangan *interface* halaman kelas untuk admin
 Gambar 3.21 adalah gambar rancangan *interface* halaman kelas untuk

admin. Di halaman tersebut admin dapat menginputkan kelas baru, mengedit data kelas lama dan dapat juga menghapusnya.

HEADER																																																	
KELAS	DATA GURU	DATA SISWA	LIST JADWAL	LIST MAPEL	BERITA	LAPORAN	LOGOUT																																										
DATA GURU <input type="button" value="PILIH DATA BERDASARKAN KELAS"/> <input type="button" value="TAMPILKAN"/>																																																	
<input type="button" value="TAMBAH SISWA"/>																																																	
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>NO</th> <th>NIS</th> <th>NAMA</th> <th>USERNAME</th> <th>TEMPAT LAHIR</th> <th>TANGGAL LAHIR</th> <th>JENIS KELAMIN</th> <th>ALAMAT</th> <th>KELAS</th> <th>ACTION</th> </tr> </thead> <tbody> <tr> <td>XX</td> <td>XX</td> <td>XXXX</td> <td>XXX</td> <td>XXX</td> <td>XXX</td> <td>XXX</td> <td>XXX</td> <td>XXXX</td> <td><input type="button" value="EDIT"/> <input type="button" value="HAPUS"/></td> </tr> <tr> <td>XX</td> <td>XX</td> <td>XXXX</td> <td>XXX</td> <td>XXX</td> <td>XXX</td> <td>XXX</td> <td>XXX</td> <td>XXXX</td> <td><input type="button" value="EDIT"/> <input type="button" value="HAPUS"/></td> </tr> <tr> <td>XX</td> <td>XX</td> <td>XXXX</td> <td>XXX</td> <td>XXX</td> <td>XXX</td> <td>XXX</td> <td>XXX</td> <td>XXXX</td> <td><input type="button" value="EDIT"/> <input type="button" value="HAPUS"/></td> </tr> </tbody> </table>										NO	NIS	NAMA	USERNAME	TEMPAT LAHIR	TANGGAL LAHIR	JENIS KELAMIN	ALAMAT	KELAS	ACTION	XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	<input type="button" value="EDIT"/> <input type="button" value="HAPUS"/>	XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	<input type="button" value="EDIT"/> <input type="button" value="HAPUS"/>	XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	<input type="button" value="EDIT"/> <input type="button" value="HAPUS"/>
NO	NIS	NAMA	USERNAME	TEMPAT LAHIR	TANGGAL LAHIR	JENIS KELAMIN	ALAMAT	KELAS	ACTION																																								
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	<input type="button" value="EDIT"/> <input type="button" value="HAPUS"/>																																								
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	<input type="button" value="EDIT"/> <input type="button" value="HAPUS"/>																																								
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	<input type="button" value="EDIT"/> <input type="button" value="HAPUS"/>																																								
FOOTER																																																	

Gambar 3.22 gambar rancangan *interface* halaman data siswa untuk admin

Gambar 3.22 adalah gambar rancangan *interface* halaman data siswa untuk admin. Di halaman tersebut admin dapat menginputkan data siswa dengan cara menekan tombol tambah siswa, lalu admin juga dapat mengedit dan menghapus data siswa.

HEADER										
KELAS	DATA GURU	DATA SISWA	LIST JADWAL	LIST MAPEL	BERITA	LAPORAN	LOGOUT			
DATA GURU <div style="background-color: #cccccc; display: inline-block; padding: 5px 20px; margin: 5px 0;">TAMBAH GURU</div>										
NO	NIP	NAMA	USER NAME	TEMPAT LAHIR	TANGGAL LAHIR	JENIS KELAMIN	ALAMAT	NO TELP	JABATAN	ACTION
XX	XX	XX	XX	XX	XXX	XXX	XX	XXX	XXX	<div style="display: inline-block; background-color: #cccccc; padding: 2px 5px; margin-right: 5px;">EDIT</div> <div style="display: inline-block; background-color: #cccccc; padding: 2px 5px;">HAPUS</div>
XX	XX	XX	XX	XX	XXX	XXX	XX	XXX	XXX	<div style="display: inline-block; background-color: #cccccc; padding: 2px 5px; margin-right: 5px;">EDIT</div> <div style="display: inline-block; background-color: #cccccc; padding: 2px 5px;">HAPUS</div>
XX	XX	XX	XX	XX	XXX	XXX	XX	XXX	XXX	<div style="display: inline-block; background-color: #cccccc; padding: 2px 5px; margin-right: 5px;">EDIT</div> <div style="display: inline-block; background-color: #cccccc; padding: 2px 5px;">HAPUS</div>
FOOTER										

Gambar 3.23 gambar rancangan *interface* halaman data guru untuk admin

Gambar 3.23 adalah gambar rancangan *interface* halaman data guru untuk admin. Di halaman tersebut admin dapat menambahkan data guru dengan cara menekan tombol tambah guru, admin juga dapat mengedit dan menghapus data guru yang lama.

HEADER							
KELAS	DATA GURU	DATA SISWA	LIST JADWAL	LIST MAPEL	BERITA	LAPORAN	LOGOUT
DATA JADWAL TAMBAH JADWAL							
KELAS			TAMPILKAN				
NO	HARI	JAM	GURU	MAPEL	KELAS	SEMESTER	ACTION
XX	XX	XXX	XXX	XX	XXX	XXX	HAPUS
XX	XX	XXX	XXX	XX	XXX	XXX	HAPUS
XX	XX	XXX	XXX	XX	XXX	XXX	HAPUS
FOOTER							

Gambar 3.24 gambar rancangan *interface* halaman data jadwal untuk admin

Gambar 3.24 adalah gambar rancangan *interface* halaman data jadwal untuk admin. Di halaman tersebut admin dapat menginputkan jadwal pelajaran untuk seluruh kelas dengan cara menekan tombol tambah jadwal

HEADER							
KELAS	DATA GURU	DATA SISWA	LIST JADWAL	LIST MAPEL	BERITA	LAPORAN	LOGOUT
DATA MAPEL TAMBAH MAPEL							
NO	ID MAPEL	NAMA MAPEL	ACTION				
XX	XXX	XXX	EDIT	HAPUS			
XX	XXX	XXX	EDIT	HAPUS			
XX	XXX	XXX	EDIT	HAPUS			
FOOTER							

Gambar 3.25 gambar rancangan halaman data mapel untuk admin

Gambar 3.25 adalah gambar rancangan halaman data mapel untuk admin. Di halaman tersebut admin dapat menambahkan mata pelajaran, mengeditnya dan menghapusnya.

HEADER							
KELAS	DATA GURU	DATA SISWA	LIST JADWAL	LIST MAPEL	BERITA	LAPORAN	LOGOUT
EDIT BERITA							
NO	JUDUL	DESKRIPSI	ISI BERITA	FOTO	PENGATURAN		
XX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXXXXXXXXXX	XXXXXXXXXX	<input type="button" value="EDIT"/> <input type="button" value="HAPUS"/>		
XX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXXXXXXXXXX	XXXXXXXXXX	<input type="button" value="EDIT"/> <input type="button" value="HAPUS"/>		
FOOTER							

Gambar 3.26 gambar rancangan *interface* halaman edit berita untuk admin

Gambar 3.26 adalah gambar rancangan *interface* halaman edit berita untuk admin. Admin dapat mengedit berita dan menghapusnya dengan cara menekan tombol di kolom pengaturan.

HEADER									
KELAS	GURU	SISWA	BERITA	JADWAL	MAPEL	LOGOUT			
DATA SGURU									
<input type="button" value="PILIH DATA BERDASARKAN JENIS KELAMIN"/>						<input type="button" value="TAMPILKAN"/>			
NO	NIP	NAMA	USERNAME	TEMPAT LAHIR	TANGGAL LAHIR	JENIS KELAMIN	ALAMAT	NO TELEPON	JABATAN
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	XXXX
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	XXXX
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX	XXXX
<input type="button" value="CETAK"/>									
FOOTER									

Gambar 3.27 gambar rancangan *interface* halaman data guru untuk kepala sekolah

Gambar 3.27 adalah gambar rancangan *interface* untuk halaman data guru untuk kepala sekolah. Di halaman tersebut kepala sekolah dapat mencetak data guru berdasarkan jenis kelamin yang ingin dipilihnya.

HEADER						
KELAS	GURU	SISWA	BERITA	JADWAL	MAPEL	LOGOUT
DATA JADWAL						
PILIH DATA BERDASARKAN KELAS	PILIH DATA BERDASARKAN HARI	PILIH DATA BERDASARKAN SEMESTER	TAMPILKAN			
NO	HARI	JAM	MAPEL	KELAS	SEMESTER	
XX	XX	XXXX	XXX	XXX	XXX	
XX	XX	XXXX	XXX	XXX	XXX	
XX	XX	XXXX	XXX	XXX	XXX	
CETAK						
FOOTER						

Gambar 3.28 gambar rancangan *interface* halaman jadwal untuk kepala sekolah

Gambar 3.28 adalah gambar rancangan *interface* untuk halaman jadwal kepala sekolah. Di halaman tersebut kepala sekolah dapat mencetak jadwal pelajaran berdasarkan kelas, hari, semester yang dipilihnya.

HEADER								
KELAS	GURU	SISWA	BERITA	JADWAL	MAPEL	LOGOUT		
DATA SISWA								
PILIH DATA BERDASARKAN KELAS			PILIH DATA BERDASARKAN JENIS KELAMIN				TAMPILKAN	
NO	NIS	NAMA	USERNAME	TEMPAT LAHIR	TANGGAL LAHIR	JENIS KELAMIN	ALAMAT	KELAS
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX
XX	XX	XXXX	XXX	XXX	XXX	XXX	XXX	XXXX
CETAK								
FOOTER								

Gambar 3.29 gambar rancangan *interface* halaman data siswa untuk kepala sekolah

Gambar 3.29 adalah gambar rancangan *interface* halaman data siswa untuk kepala sekolah. Di halaman tersebut kepala sekolah dapat mencetak data siswa berdasarkan kelas dan jenis kelamin yang dipilihnya dengan cara menekan tombol cetak.

SMA YPI TUNAS BANGSA PALEMBANG									
NO	NIP	NAMA	USERNAME	TEMPAT LAHIR	TANGGAL LAHIR	JENIS KELAMIN	ALAMAT	NO TELEPON	JABATAN
XX	XX	XXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX
XX	XX	XXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX
XX	XX	XXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX
XX	XX	XXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	XXXX
PALEBANG, XX-XX-XXXX MENGETAHUI KEPALA SEKOLAH									
XXXXX									

Gambar 3.30 gambar rancangan *interface* cetak laporan data guru

Gambar 3.30 adalah gambar rancangan *interface* hasil cetak laporan data guru. Di laporan tersebut terdapat tanggal otomatis berdasarkan waktu ketika laporan itu dicetak dan dapat langsung di tandatangani oleh kepala sekolah. Hanya admin dan kepala sekolah yang dapat mencetak laporan ini.

		SMA YPI TUNAS BANGSA PALEMBANG						
NIS : XXX								
NAMA : XXXX								
KELAS : XXX								
NO	MAPEL	SEMESTER					TOTAL	RATA-RATA
		SUB 1	MID 1	SUB 2	TUGAS	UAS 1		
XX	XXX	XX	XX	XX	XX	XX	XXX	XXX
XX	XXX	XX	XX	XX	XX	XX	XXX	XXX
XX	XXX	XX	XX	XX	XX	XX	XXX	XXX
PALEBANG, XX-XX-XXXX MENGETAHUI KEPALA SEKOLAH								XXXXX

Gambar 3.31 gambar rancangan *interface* laporan nilai untuk siswa

Gambar 3.31 adalah rancangan *interface* laporan nilai untuk siswa. Setelah siswa menekan tombol cetak maka laporan tersebut akan tercetak. Di laporan tersebut terdapat list mata pelajaran, nilai, total, dan rata-rata. Total dan rata-rata akan terisi otomatis melalui perhitungan yang ada di sistem.

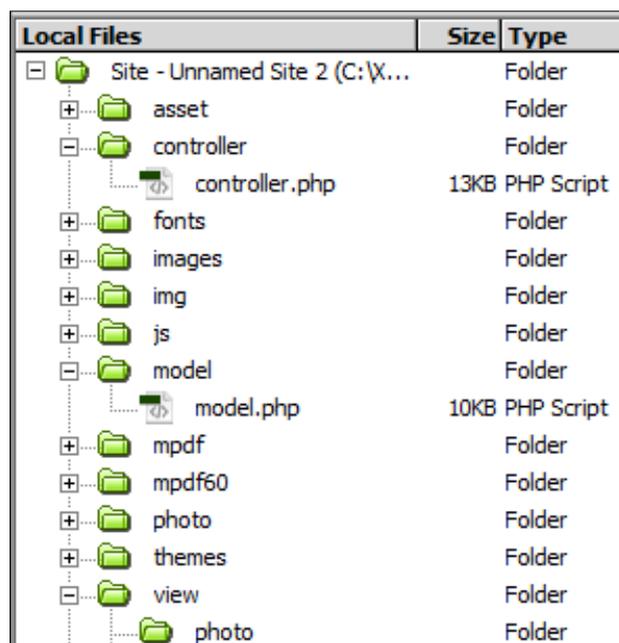
BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil

Setelah melalui proses perancangan dan pengkodean program maka dihasilkan suatu sistem informasi akademik di SMA YPI Tunaas Bangsa Palembang, adapun sistem yang dihasilkan merupakan hasil rancangan yang dilakukan penulis pada bab-bab sebelumnya untuk mencapai tujuan yang telah ditentukan, berikut ini merupakan hasil yang dicapai:

- a. Sistem informasi ini dapat melakukan pengolahan nilai, baik penginputan nilai yang dilakukan oleh guru maupun pencetakan nilai yang dilakukan oleh siswa
- b. Sistem informasi ini dapat menampilkan jadwal pelajaran yang dan mengolah jadwal pelajaran agar tidak terjadi tabrakan antar jadwal.
- c. Sistem informasi ini dapat menyimpan data guru, siswa, jadwal dengan rapi dan aman, tidak seperti selama ini data guru yang disimpan di kertas dan dijadikan arsip.
- d. Sistem informasi ini dapat menampilkan berita-berita seperti pengumuman jadwal ujian, pengumuman acara, dan pengumuman libur sekolah.



Gambar 4.1 Folder-folder komponen program SIA SMA YPI Tunas Bangsa

Pengkodean program dilakukan dengan menggunakan metode MVC (*model view controller*) yang memisahkan program menjadi 3 bagian yaitu bagian, *model*, *view*, dan *controller*.

4.2 Pembahasan

Sistem informasi akademik ini dapat melakukan pengolahan nilai, pengolahan jadwal, dan pengolahan data guru serta siswa. Pengolahan nilai disini dilakukan oleh guru dengan cara menginputkan nilai-nilai siswa, mulai dari nilai subsumatif 1 dan 2, nilai mid semester, nilai tugas dan nilai ujian akhir sekolah, guna mendapatkan nilai rata-rata siswa, dengan begitu perhitungan nilai siswa lebih akurat. Pengolahan jadwal maksudnya jadwal pelajaran antar kelas disekolah tersebut dapat tersusun rapi, tanpa ada jadwal yang tidak valid, sistem ini akan otomatis mengeluarkan peringatan jika ada jadwal yang sama, mulai dari hari, kelas, jam, mata pelajaran, dan guru. Pengolahan data disini maksudnya

data-data seperti data guru, data siswa, data jadwal, dan data nilai dapat disimpan rapi dan aman. Selain itu sistem informasi akademik ini dapat mencetak nilai siswa, jadwal pelajaran, data guru, data siswa dengan menekan tombol cetak yang berguna untuk siswa, guru, dan kepala sekolah.

4.2.1 Halaman Utama

View :

```
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>Sistem Informasi Akademik </title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="description" content="">
  <meta name="author" content="">
  <link href="asset/css/featured_slide.css" rel="stylesheet" type="text/css">
  <script src="themes/js/jquery.min.js" type="text/javascript"></script>
  <script src="themes/js/berita.js" type="text/javascript"></script>
  <link id="callCss" rel="stylesheet" href="themes/current/bootstrap.min.css"
  type="text/css" media="screen"/>
  <link href="themes/css/bootstrap-responsive.min.css" rel="stylesheet"
  type="text/css">
  <link href="themes/css/font-awesome.css" rel="stylesheet" type="text/css">
  <link href="themes/css/base.css" rel="stylesheet" type="text/css">
  <style type="text/css" id="enject"></style>
  <link rel="shortcut icon" href="themes/images/logo.png" />
<body>
  <div class="container" style="width: 87%; text-align: left;">
  <div class="raw">
  <table width="100%" border="1">
    <tr>
      <th colspan="5" scope="col"> <a class="brand" href="..">
   </a></th>
    </tr>
    <tr>
      <td colspan="5">
      <div class="menu-wrapper">
  <ul class="menu-horizontal" style="margin: -11px -130px; width: 1197px;">
  <li><a class="brand" href="index.php?page=beranda">Home</a></li>
  <li><a class="brand" href="index.php?page=Profil">Profil</a></li>
  <li><a class="brand" href="index.php?page=sejarah">Sejarah</a></li>
  <li><a class="brand" href="index.php?page=visi-misi">Visi-Misi</a></li>
  <li><a class="brand" href="index.php?page=galleri">Galleri</a></li> </ul>
```

```

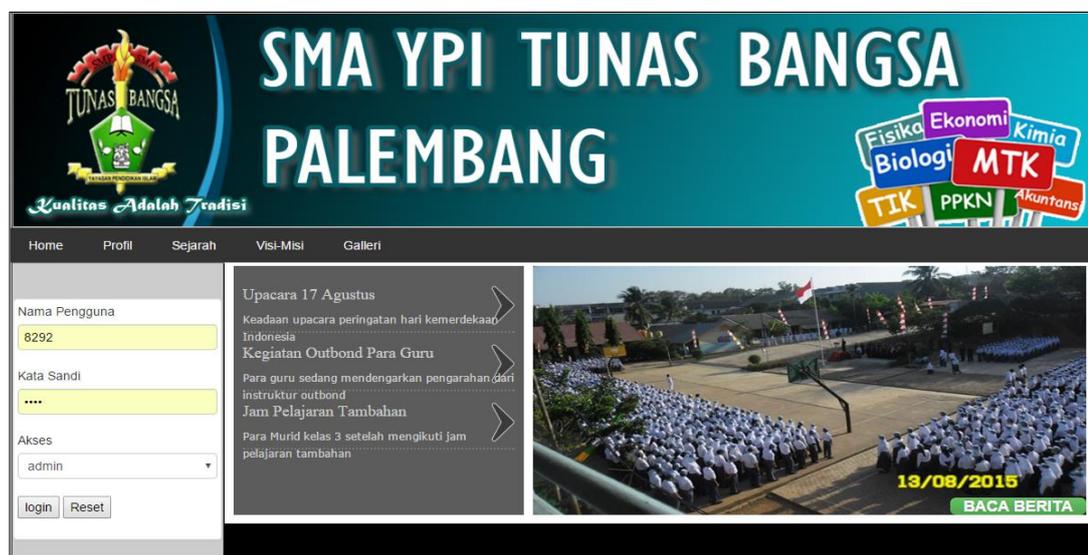
</div> </td> </tr>
<?php
session_start();
include "controller/controller.php";
    $main = new controller();
    $page=$_GET['page'];
    switch($page){
    case "Profil";
        $main->profil();
        break;
    case "sejarah";
        $main->sejarah();
        break;
    case "visi-misi";
        $main->visimisi();
        break;
    case "galleri";
        $main->galleri();
        break;
    default ;
        $main->beranda();
        break; }
?>
<section id="footerSection" style="width: 1197px;">
<div class="container" >
<footer class="footer well well-small">
<center>&copy; 2015, <a href="">SMA YPI Tunas Bangsa Palembang</a>.
M.11 &copy;copyright</center>
<center>Madzanni@gmail.com|| Sistem Informasi 2011 || UIN Raden Fatah
Palembang</center> </footer>
</div>
</section>
</div>
</div>
<script src="themes/js/jquery-1.8.3.min.js"></script>
<script src="themes/js/bootstrap.min.js"></script>
<script src="themes/js/bootstrap-tooltip.js"></script>
<script src="themes/js/bootstrap-popover.js"></script>
</body>
</html>

```

Gambar 4.2 Source code pada halaman utama

Gambar 4.2 berisi *source code* pada halaman utama. Ketika *source code* tersebut dijalankan maka *browser* akan menampilkan halaman utama, dimana *user*

dapat memasukkan *username* dan *password* agar dapat masuk ke dalam menu sesuai dengan kebutuhan dan hak akses masing-masing.



Gambar 4.3 Tampilan Halaman Utama

Gambar 4.3 merupakan hasil eksekusi *source view* beranda yang ada di dalam gambar 4.3. Halaman utama ini dapat menghubungkan *link* ke menu dan halaman lain dengan cara memasukkan *username* dan *password* dengan benar dan memilih hak akses sesuai *username*. Adapun fungsi dari halaman utama ini adalah sebagai pintu masuk bagi pengguna untuk masuk ke sistem sesuai dengan kebutuhannya.

4.2.2 Halaman Nilai Siswa

Model

```
function nilaisiswa($semester){
$query = "SELECT * FROM nilai,mapel where
nilai.id_mapel=mapel.id_mapel and nilai.nis='$_SESSION[username]' and
nilai.semester='$semester'";
return $this->execute($query);}

```

Controller

```
function nilaisiswa(){ //fungsi untuk melihat nilai
$semester = $_POST['semester'];
$datapofil = $this->model->profilsiswa();
}

```

```
$data = $this->model->nilaisiswa($semester);
include"view/nilai_siswa.php"; }
```

View

```
<div class="row">
<div class="span4">
<div class="well well-small">
<?php $row = $this->model->fetch($datapofil)?>
<table align="center">
  <tr> <td><b>NIS</b></td>
  <td><b>:</b></td>
  <td><?php echo $_SESSION['username'];?></td> </tr>
  <tr> <td><b>Nama</b></td>
  <td><b>:</b></td>
  <td><?php echo $row['nama'];?></td> </tr>
  <tr> <td><b>Kelas</b></td>
  <td><b>:</b></td>
  <td><?php echo $row['nama_kelas'];?></td> </tr>
</table>
</div>
</div>
</div>
<form action="" method="POST">
  Semester = <select name="semester" id="select">
    <option value="1">Ganjil</option>
    <option value="2">Genap</option>
  </select>
  <input type="submit" name="tampil" value="Tampilkan">
</form>
</center><hr>
<left><h3><b>NILAI SEMESTER </b></h3>
<table align="center" border="1">
  <tr>
    <th align="center">NO.</th>
    <th align="center">Mapel</th>
    <th align="center">SUB 1</th>
    <th align="center">MID 1</th>
    <th align="center">SUB 2</th>
    <th align="center">TUGAS</th>
    <th align="center">UAS 1</th>
    <th align="center">TOTAL</th>
    <th align="center"><font color="green">RATA-RATA</font></th>
  </tr>
  <?php if(isset($_POST['tampil'])){
    echo'<b>SEMESTER</b>'; echo $_POST['semester'];
  }else{
    echo'<b>SEMESTER</b>';
```

```

}??>
<?php
    $no=1;
    while($result = $this->model->fetch($data)){
        $total = (($result['sub1']*1.5) + ($result['mid1']*2) + ($result['sub
2']*1.5) + ($result['Tugas']*1) + ($result['uas2']*4));
        $total_nilai = $total/10;
    }?>
<tr>
<td align="center"><?php echo $no;?>.</td>
<td align="center"><?php echo $result['mapel'];?></td>
<td align="center"><?php echo $result['sub1'];?></td>
<td align="center"><?php echo $result['mid1'];?></td>
<td align="center"><?php echo $result['sub 2'];?></td>
<td align="center"><?php echo $result['Tugas'];?></td>
<td align="center"><?php echo $result['uas2'];?></td>
<td align="center"><?php echo $total;?></td>
<td align="center"><?php echo $total_nilai;?></td>
</tr>
<?php $no++;}
?>
</table>
<form action="view/cetak_nilai.php" target="_blank" method="POST">
<input type="hidden" id="kelas" name="nama" value="<?php echo
$row['nama'];?>" />
<input type="hidden" id="kelas" name="kelas" value="<?php echo
$row['nama_kelas'];?>" />
<input type="hidden" id="kelas" name="semester" value="<?php echo
$_POST['semester'];?>" />
<input type="submit" name="cetak" value="Cetak">
</form>

```

Gambar 4.4 *Source Code* pada Halaman Nilai Siswa

Gambar 4.4 berisi *source code* pada halaman nilai siswa. Ketika *source code* tersebut dijalankan maka *browser* akan menampilkan halaman nilai siswa, dimana siswa dapat mencetak nilai mereka. *Model* disini berfungsi sebagai tempat menyimpan *username*, *password* dan nilai siswa. Kemudian, *controller* akan memanggil data-data apa saja yang akan ditampilkan di *view*.

The screenshot shows a web interface titled "Halaman Siswa". At the top, there are navigation links: "Profil", "Nilai", "Data Jadwal", and "Logout". Below this, a profile box displays the following information:

- NIS : 8175
- Nama : Muhammad Iqbal Tawakkal
- Kelas : X 2

Below the profile box, there is a "Semester =" dropdown menu set to "Ganjil" and a "Tampilkan" button. Underneath, the section "NILAI SEMESTER" is displayed in red. Below this, the text "SEMESTER1" is shown. A table with 9 columns (NO., Mapel, SUB 1, MID 1, SUB 2, TUGAS, UAS 1, TOTAL, RATA-RATA) contains the following data:

NO.	Mapel	SUB 1	MID 1	SUB 2	TUGAS	UAS 1	TOTAL	RATA-RATA
1.	Bahasa Inggris	88	88	88	88	88	880	88
2.	Penjaskes	66	66	66	66	66	660	66
3.	Sejarah	77	77	77	77	77	770	77

At the bottom left of the table area, there is a "Cetak" button.

Gambar 4.5 Tampilan Halaman Nilai Siswa

Gambar 4.5 merupakan hasil eksekusi *source code* yang ada di dalam *model*, *controller*, dan *view*. Halaman nilai ini hanya dapat diakses oleh siswa dengan cara memasukkan *username* dan *password* dengan benar di halaman utama.

4.2.3 Halaman Input Nilai

Model:

```
function insert_nilai(){
    $kls=$_GET['id_kelas'];
    $smtr=$_GET['semester'];
    $id = $_SESSION['id_guru'];
    $no =1;
    $total=0;
    $biru = mysql_query("select * from siswa where
id_kelas=$_GET[id_kelas]");
    while($bale = mysql_fetch_array($biru)){
    $ns1=$_POST['s1'].$bale['nis'];
    $nm=$_POST['m'].$bale['nis'];
    $s2=$_POST['s2'].$bale['nis'];
    $nt=$_POST['t'].$bale['nis'];
```

```

    $nu=$_POST['u'].$bale['nis'];
    $nsm=$_POST['sm'].$bale['nis'];
    mysql_query("insert                into                nilai
values(',$bale[nis]','$_SESSION[username]','$_SESSION[mapel]','$ns1','$nm','
$s2','$nt','$nu','$nsm)") or die ('mysql error'); }

```

Controller:

```

function insert_nilai(){
$insert = $this->model->insert_nilai();
echo    "<meta    charset='UTF-8'>    <meta    http-equiv='refresh'
content='0;url=guru.php?id_kelas=$_GET[id_kelas]&semester=$_GET[semest
er]&page=data-nilai&kelas=Tampilkan'>"; }

```

View:

```

<?php $hasil= $this->model->fetch($profil); ?>
<center><h3>SELAMAT                DATANG                <?php                echo
$hasil['nama']?><br/>Silahkan Menginput Nilai</h3></center>
<form action="" method="GET">
Kelas = <select name="id_kelas" id="select">
<?php while($row = $this->model->fetch($data1)){ ?>
<option value="<?php echo $row['id_kelas'] ?>"><?php echo
$row['nama_kelas'] ?></option> <?php }?>
</select>
Semester = <select name="semester" id="semester" style="width: 46px;">
<option value="1">1</option>
<option value="2">2</option>
</select>
<input type="hidden" name="page" value="data-nilai&kelas=Tampilkan"/>
<input type="submit" id="kelas" name="kelas" value="Tampilkan">
</form></center><hr>
<form action="" method="post"/>
<?php if(isset($_GET['kelas'])){ ?>
<table border="1" align="center">
    <tbody>
        <tr>
            <th align="center">No.</th>
            <th align="center">Nis</th>
            <th align="center">Nama</th>
            <th align="center">Nilai Sub1</th>
            <th align="center">Nilai mid1</th>
            <th align="center">Nilai Sub2</th>
            <th align="center">Nilai Tugas</th>
            <th align="center">Nilai Uas1</th>
            <th align="center">Semester</th>
            <th align="center">Rata-rata</th>
        </tr>
    </tbody>
</table>
<?php
    $no=1;

```



```

<?php $no++;} ?> </tbody> </table>
<input type="hidden" name="semester" value="<?php echo $smtr ?>"/>
<input type="hidden" name="kelas" value="<?php echo $id_ks ?>"/>
<input type="hidden" name="id_kelas" value="<?php echo $_GET['id_kelas']
?>"/>
<?php
echo "<center><br><input type='submit' name='simpan' value='simpan'
/></center>";
?></form>
<?php }if(isset($_POST['simpan'])){
    $main = new controller();
    $main->insert_nilai(); }?>

```

Gambar 4.6 Source Code pada halaman input nilai siswa

Gambar 4.6 berisi *source code* pada halaman input nilai siswa. Ketika *source code* tersebut dijalankan maka *browser* akan menampilkan halaman input nilai siswa. *Model* disini berfungsi sebagai tempat menyimpan *username*, *password* dan nilai siswa. Kemudian, *controller* akan memanggil data-data apa saja yang akan ditampilkan di *view*.

No.	Nis	Nama	Nilai Sub1	Nilai mid1	Nilai Sub2	Nilai Tugas	Nilai Uas1	Semester	Rata-rata
1.	8555	Gilang Ramdhan	90	90	90	90	90	1	90
2.	8595	Muhammad Fadil Azka	90	90	90	90	90	1	90
3.	8517	Fauzan Akabar	99	99	99	99	99	1	99
4.	8496	Arief Purnama Putra	99	99	99	99	99	1	99
5.	8298	Dhita Ayu Wandira O	99	99	99	99	99	1	99
6.	8291	Devi Lestari Ramadhani	99	99	99	99	99	1	99
7.	8278	Ayu Yuliani	99	99	99	99	99	1	99
8.	8263	Anisah Meidella	99	99	99	9	99	1	90
9.	8265	Anita Hamini	99	99	99	99	99	1	99
10.	8254	Allisia Fransiska	99	99	99	99	99	1	99

Gambar 4.7 Tampilan Halaman Input Nilai

Gambar 4.7 merupakan hasil eksekusi *source code* yang ada di dalam *model*, *controller*, dan *view*. Halaman input nilai ini hanya dapat di akses oleh guru dengan memasukkan *username* dan *password* dengan benar.

4.2.4 Halaman Input Jadwal

Model:

```
function insertjadwal($hari, $jam, $mapel, $kelas, $guru, $kelas, $ta,
$semester){
$query =mysql_fetch_array(mysql_query("select * from guru where
id_guru='$guru'"));
$query2 =mysql_fetch_array(mysql_query("select * from kelas where
id_kelas='$kelas'"));
$query3 =mysql_fetch_array(mysql_query("select * from mapel where
id_mapel='$mapel'"));
$sql = mysql_query("select * from jadwal where hari ='$hari' and jam='$jam'
and id_mapel='$mapel' and id_kelas='$kelas' and id_guru='$guru' and
semester='$semester' and ta='$ta'");
    $jumlah = mysql_num_rows($sql);
    if($jumlah == 0){
        mysql_query("insert into jadwal values (',$hari', '$jam', '$mapel',
'$kelas', '$guru', '$kelas', '$ta)");
        echo "<meta charset='UTF-8'> <meta http-equiv='refresh'
content='0;url=?page=input-jadwal'>";
        echo "<script>alert('Jadwal Berhasil Disimpan');
window.location.href=?page=input-jadwal'; </script>"; }
}
```

Controller:

```
function insertjadwal(){
    $hari = $_POST['hari'];
    $jam = $_POST['jam'];
    $mapel = $_POST['mapel'];
    $kelas = $_POST['kelas'];
    $guru = $_POST['guru'];
    $kelas = $_POST['kelas'];
    $ta = $_POST['ta'];
    $semester = $_POST['semester'];
    $data = $this->model->insertjadwal($hari, $jam, $mapel, $kelas,
$guru, $kelas, $ta, $semester); }
}
```

View:

```
<form action="" method="post">
<table class="form">
<tr> <td valign="top">Hari : </td>
    <td>
        <select name="hari" id="select">
        <option value="senin">Senin</option>
```

```

<option value="selasa">Selasa</option>
<option value="rabu">Rabu</option>
<option value="Kamis">Kamis</option>
<option value="jumat">Jumat</option>
<option value="sabtu">Sabtu</option>
</select> </td> </tr>
<tr> <td valign="top">Jam : </td>
<td> <select name="jam" id="select">
<option value="(1)7.00-7.45">(1)7.00-7.45</option>
<option value="(2)7.45-8.30">(2)7.45-8.30</option>
<option value="(3)8.30-9.15">(3)8.30-9.15</option>
<option value="(4)09.325-10.10">(4)9.25-10.10</option>
<option value="(5)10.10-10.55">(5)10.10-10.55</option>
<option value="(6)11.05-11.50">(6)11.05-11.50</option>
<option value="(7)11.50-12.35">(7)11.50-12.35</option>
<option value="(8)12.35-13.05">(8)12.35-13.05</option>
</select> </td> </tr>
<tr> <td valign="top">Kelas : </td>
<td> <select name="kelas" id="select">
<?php while($row = $this->model->fetch($data1)){ ?>
<option value="<?php echo $row['id_kelas'] ?>"><?php echo
$row['nama_kelas'] ?></option>
<?php }?>
</select> </td> </tr>
<tr> <td valign="top">Guru : </td>
<td> <select name="guru" id="select">
<?php while($row = $this->model->fetch($data2)){ ?>
<option value="<?php echo $row['id_guru'] ?>"><?php echo
$row['nama'] ?></option>
<?php }?>
</select> </td> </tr>
<tr> <td valign="top">Mapel : </td>
<td> <select name="mapel" id="select">
<?php while($row = $this->model->fetch($data3)){ ?>
<option value="<?php echo $row['id_mapel'] ?>"><?php echo
$row['mapel'] ?></option>
<?php }?> </select> </td> </tr>
<tr> <td valign="top">Semester : </td>
<td> <select name="semester" id="select">
<option value="1">Ganjil</option>
<option value="2">Genap</option>
</select> </td> </tr>
<tr> <td valign="top">Tahun Akademik : </td>
<td> <input type="text" name="ta" placeholder="Tahun Akademik">
</td> </tr>
<tr>

```

```
<td><input type="submit" name="simpan" value="simpan"/></td> </tr>
</table>
</form>
```

Gambar 4.8 Source Code pada Halaman Admin Input Jadwal

Gambar 4.8 berisi *source code* pada halaman admin input jadwal. Ketika *source code* tersebut dijalankan maka *browser* akan menampilkan halaman admin input jadwal. *Model* disini berfungsi sebagai tempat menyimpan *username*, *password* dan nilai siswa. Kemudian, *controller* akan memanggil data-data apa saja yang akan ditampilkan di *view*.

Gambar 4.9 Tampilan Halaman Input Jadwal

Gambar 4.9 merupakan hasil dari eksekusi *source code* yang terdapat di dalam *model*, *view*, dan *controller* yang ada di gambar 4.8. Halaman ini dapat diakses hanya oleh admin dengan memasukkan *username* dan *password* dengan benar.

4.2.5 Halaman Input Data Siswa

Model:

```
function insertsiswa($username, $password, $nis, $nama, $jengkel,
$tempat_lahir, $tgl, $alamat, $kelas, $ta){
$query = "insert into siswa values('$username', '$password', '$nis', '$nama',
'$jengkel', '$tempat_lahir', '$tgl', '$alamat', '$kelas', '$ta)";
return $this->execute($query); }
```

Controller:

```

function insertsiswa(){
    $username = $_POST['username'];
    $password = $_POST['password'];
    $nis = $_POST['nis'];
    $nama = $_POST['nama'];
    $jengkel = $_POST['jengkel'];
    $tempat_lahir = $_POST['tempat_lahir'];
    $tgl = $_POST['tgl'];
    $alamat = $_POST['alamat'];
    $kelas = $_POST['kelas'];
    $ta = $_POST['ta'];
    $insert = $this->model->insertsiswa($username, $password, $nis,
    $nama, $jengkel, $tempat_lahir, $tgl, $alamat, $kelas,$ta);
    echo "<meta charset='UTF-8'> <meta http-equiv='refresh'
    content='0;url=?page=data-siswa&&kelas=$kelas'>"; }

```

View:

```

<h3>INPUT SISWA</h3>
<form action="" method="post">
<table class="form">
<tr> <td valign="top">NIS : </td>
    <td><input type="text" name="nis" required/></td>
</tr>
<tr> <td valign="top">Nama : </td>
    <td><input type="text" name="nama" required/></td>
</tr>
<tr> <td valign="top">Username : </td>
    <td><input type="text" name="username" required/></td>
</tr>
<tr><td valign="top">Password : </td>
    <td><input type="password" name="password" required/></td>
</tr>
<tr><td valign="top">Kelas : </td>
    <td><select name="kelas" id="select">
<?php while($row = $this->model->fetch($data1)){ ?>
<option value="<?php echo $row['id_kelas'] ?>"><?php echo
$row['nama_kelas'] ?></option>
<?php }?> </select> </td> </tr>
<tr><td valign="top">Tempat lahir : </td>
    <td><input type="text" name="tempat_lahir" required/></td>
</tr>
<tr><td valign="top">tgl Lahir: </td>
    <td><input type="date" name="tgl" required/></td>
</tr>
<tr><td valign="top">Jenis kelamin : </td>
    <td><select name="jengkel" >
        <option value="Laki-Laki">Laki-Laki</option>

```

```

<option value="Perempuan">Perempuan</option>
</select> </td> </tr>
<tr> <td valign="top">Alamat : </td>
<td><input type="text" value="Isi Sendiri"
name="alamat" placeholder="isi sendiri" > </td> </tr>
<tr> <td valign="top">Tahun Ajaran : </td>
<td><input type="text" name="ta" required/> </td>
</tr>
<tr> <td> <input type="submit" name="simpan" value="simpan"/></td></tr>
</table>
</form>

```

Gambar 4.10 *Source code* pada halaman input data siswa

Gambar 4.10 merupakan *source code* pada halaman input siswa. Ketika *source code* yang tersebut dijalankan maka *browser* akan menampilkan halaman input data siswa. *Model* disini berfungsi sebagai tempat menyimpan *username*, *password* dan nilai siswa. Kemudian, *controller* akan memanggil data-data apa saja yang akan ditampilkan di *view*.

Gambar 4.11 Tampilan Halaman Input Data Siswa

Gambar 4.11 merupakan hasil dari eksekusi *source code* yang ada di gambar 4.10. Halaman ini hanya dapat diakses oleh admin dengan cara memasukkan *username* dan *password* dengan benar pada halaman beranda.

4.2.6 Halaman Laporan Data Siswa

```

Model:
function adminsiswa($kelas) {
$query = "select * from siswa,kelas where siswa.id_kelas='$kelas' and
siswa.id_kelas=kelas.id_kelas ";
return $this->execute($query);}

Controller:
function adminsiswa(){
$data = $this->model->seluruhsiswa();
include"view/cetaklaporanseluruhsiswa.php"; }

View:
<table id="iqbal" border="1" class="table table-bordered table-striped">
<thead> <tr>
    <th>No.</th>
    <th>NIS</th>
    <th>Nama</th>
    <th>Username</th>
    <th>Tempat Lahir</th>
    <th>Tanggal lahir</th>
    <th>Jenis Kelamin</th>
    <th>Alamat</th>
    <th>Kelas</th>
    <th>Tahun Akademik</th>
</tr> </thead>
<tbody>
    <?php
    $no=1;
    while($row = $this->model->fetch($data)){
    ?>
    <tr>
        <td><?php echo $no;?></td>
        <td><?php echo $row['nis'];?></td>
        <td><?php echo $row['nama']; ?></td>
        <td><?php echo $row['username']; ?></td>
        <td><?php echo $row['tempat_lahir']; ?></td>
        <td><?php echo $row['tgl']; ?></td>
        <td><?php echo $row['jengkel']; ?></td>
        <td><?php echo $row['alamat']; ?></td>
        <td><?php echo $row['nama_kelas']; ?></td>
        <td><?php echo $row['ta']; ?></td>
    </tr>
        <?php $no++;} ?>
    </tbody>
</table>

```

Gambar 4.12 Source code Halaman hasil cetak laporan data siswa

Gambar 4.12 merupakan *source code* pada halaman hasil cetak laporan data siswa. Ketika *source code* yang tersebut dijalankan maka *browser* akan menampilkan halaman hasil cetak laporan yang user dapat memilih untuk mengunduh atau sekedar melihat laporan saja.

No.	NIS	Nama	Username	Tempat Lahir	Tanggal lahir	Jenis Kelamin	Alamat	Kelas
1.	8254	Allina Fransiska	8254	Palembang	1999-08-14	Perempuan		X.1
2.	8263	Anisah Mezdella	8263	palembang	2000-05-24	Perempuan		X.1
3.	8265	Anisa Harmini	8265	Palembang	1999-05-02	Perempuan		X.1
4.	8266	Anissa Indah Pertika	8266	Palembang	1999-12-09	Perempuan		X.2
5.	8278	Ayu Yuliani	8278	Palembang	1999-10-24	Perempuan		X.1
6.	8288	Desiana Siregar	8282	Maba	1998-01-26	Perempuan		X.2
7.	8291	Dewi Lestari Ramadhani	8291	Palembang	1999-12-26	Perempuan		X.1
8.	8298	Dhita Ayu Wandira O	8298	Palembang	1999-10-28	Perempuan		X.1
9.	8347	Shaga Pratiwi	8347	Palembang	2000-05-15	Perempuan		X.2
10.	8391	Hani Elvina Putri	8391	Palembang	2000-01-27	Perempuan		X.2
11.	8496	Artief Purmana Putra	8496	Palembang	2000-10-22	Laki-Laki		X.1
12.	8499	Roya Saputra	8499	Tanjung Enim	1999-07-05	Laki-Laki		X.2
13.	8517	Rauan Akbar	8517	Tamparan	1999-06-12	Laki-Laki		X.1
14.	8555	Dilang Hamdhan	8555	Palembang	1999-12-15	Laki-Laki		X.1
15.	8595	Muhammad Rafid Aka	8595	Bandung	1999-07-25	Laki-Laki		X.1

Palembang, 13-12-2015
Kepala Sekolah

Fahrurrozi, S.Pd., M.M

Gambar 4.13 Tampilan Halaman Laporan Data Siswa

Gambar 4.13 merupakan hasil dari eksekusi *source code* yang ada di gambar 4.12. Halaman ini hanya dapat diakses oleh kepala sekolah dengan cara memasukkan *username* dan *password* dengan benar.

4.2.7 Halaman Lihat Jadwal Siswa

Model:

```
function jadwalsiswa(){
$query = "SELECT * FROM siswa,jadwal,kelas,mapel where
jadwal.id_kelas=kelas.id_kelas and jadwal.id_mapel=mapel.id_mapel
and
siswa.nis='$_SESSION[username]' and
siswa.id_kelas=jadwal.id_kelas";
return $this->execute($query); }
```

Controller:

```
function viewjadwalsiswa(){
$data = $this->model->jadwalsiswa();
$profil = $this->model->nama_siswa();
include "view/siswa_jadwal.php"; }
```

```

View:
<?php $hasil= $this->model->fetch($profil); ?>
<center><h3>SELAMAT          DATANG          <?php          echo
$hasil['nama']?></h3></center>
<center><h3>Jadwal Mata Pelajaran </h3></center>
<center>  <table  class="form"  border="1"  cellpadding="5"
cellspacing="0" align="center">
  <tr align="center">
    <td>No</td>
    <th align="center">Hari</th>
    <th align="center">Jam</th>
    <th align="center">Mata Pelajaran</th>
    <th align="center">Kelas</th>
    <th align="center">Semester</th>
    <th align="center">Tahun Akademik</th> </tr>
<?php $no=1;
while($row = $this->model->fetch($data)){
?>
  <tr><td align="center" id="no"><?php echo $no;?>.</td>
  <td align="center"><?php echo $row['hari']?></td>
  <td align="left" id="nama"><?php echo $row['jam']; ?></td>
  <td align="left" id="nama"><?php echo $row['mapel']; ?></td>
  <td align="left" id="nama"><?php echo $row['nama_kelas'];
?></td>
  <td align="left" id="nama"><?php echo $row['semester'];
?></td>
  <td align="left" id="nama"><?php echo $row['ta']; ?></td>
  </tr>
<?php $no++;}?>
</table>
<form action="cetak.php?page=siswajadwal" target="_blank"
method="POST">
<input type="submit" id="kelas" name="nis" value="Cetak">
<input type="hidden" id="kelas" name="nama" value="<?php
echo $row['nama'];?>">
<input type="hidden" id="kelas" name="kelas" value="<?php
echo $row['nama_kelas'];?>">
</form>

```

Gambar 4.14 *Source code* halaman lihat jadwal siswa

Gambar 4.14 merupakan *source code* pada halaman lihat jadwal siswa. Ketika *source code* yang tersebut dijalankan maka *browser* akan menampilkan halaman halaman lihat jadwal siswa.

Halaman Siswa						
Profil	Nilai	Data Jadwal	Logout			
Jadwal Mata Pelajaran						
No	Hari	Jam	Mata Pelajaran	Kelas	Semester	Tahun Akademik
1.	senin	(1)7.00-7.45	Matematika	X 1	1	2013-2014
2.	senin	(4)09.325-10	Sejarah	X 1	1	2013-2014
3.	Kamis	(1)7.00-7.45	Kewarganegaraan	X 1	1	2013-2014
4.	sabtu	(4)09.325-10	Bahasa Inggris	X 1	1	2013-2014
<input type="button" value="Cetak"/>						

Gambar 4.15 Tampilan halaman jadwal siswa

Gambar 4.15 merupakan hasil dari eksekusi *source code* yang ada di gambar 4.14. Halaman ini hanya dapat diakses oleh siswa dengan cara memasukkan *username* dan *password* dengan benar.

4.2.8 Halaman Lihat Jadwal Mengajar Guru

Model:

```
function jadwalguru(){
$query = "SELECT * FROM jadwal,kelas,mapel where
jadwal.id_kelas=kelas.id_kelas and jadwal.id_mapel=mapel.id_mapel
and jadwal.id_guru='$_SESSION[username]'";
return $this->execute($query); }
```

Controller:

```
function viewjadwalguru(){
$hasil =$_GET['nama'];
$data = $this->model->jadwalguru();
$profil = $this->model->nama_guru();
include "view/jadwal_guru.php"; }
```

View:

```
<?php $hasil= $this->model->fetch($profil); ?>
<center><h3>SELAMAT DATANG <?php echo
$hasil['nama']?></h3></center>
<table class="form" border="1" cellpadding="5" cellspacing="0"
align="center">
<tr align="center">
<td>No</td>
<th align="center">Hari</th>
```

```

<th align="center">Jam</th>
<th align="center">Mata Pelajaran</th>
<th align="center">Kelas</th> </tr>
<?php $no=1;
    while($row = $this->model->fetch($data)){ ?>
<tr>
<td align="center" id="no"><?php echo $no;?>.</td>
<td align="center"><?php echo $row['hari'];?></td>
<td align="left" id="nama"><?php echo $row['jam']; ?></td>
<td align="left" id="nama"><?php echo $row['mapel']; ?></td>
<td align="left" id="nama"><?php echo $row['nama_kelas']; ?></td>
</tr>
<?php $no++;} ?>
</table>
<form      action="cetak.php?page=gurujadwal"      target="_blank"
method="POST">
<input type="submit" id="kelas" name="nis" value="Cetak">
</form>

```

Gambar 4.16 Source code halaman lihat jadwal mengajar guru

Gambar 4.16 merupakan *source code* pada halaman lihat jadwal siswa. Ketika *source code* yang tersebut dijalankan maka *browser* akan menampilkan halaman halaman lihat jadwal siswa.

Halaman Guru

[Profil](#) [Data Nilai](#) [Data Jadwal](#) [Logout](#)

SELAMAT DATANG Pauzan, S.Ag

No	Hari	Jam	Mata Pelajaran	Kelas
1.	selasa	(3)8.30-9.15	Pendidikan Agama Islam	X 2
2.	sabtu	(1)7.00-7.45	Pendidikan Agama Islam	X 3
3.	jumat	(6)11.05-11.	Pendidikan Agama Islam	XI IPA 1
4.	jumat	(5)10.10-10.	Pendidikan Agama Islam	X 4
5.	rabu	(8)12.35-13.	Pendidikan Agama Islam	XII IPS 1

Gambar 4.17 Tampilan halaman lihat jadwal mengajar guru

Gambar 4.16 merupakan hasil dari eksekusi *source code* yang ada di gambar 4.15. Halaman ini hanya dapat diakses oleh guru dengan cara memasukkan *username* dan *password* dengan benar.

4.3 Pengujian Sistem

Pengujian yang digunakan untuk menguji sistem informasi ini adalah menggunakan metode pengujian *black box*.

Black Box Testing

Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. (Rosa dan Shalahuddin, 2014:276).

Tabel 4.1 Tabel Pengujian yang dilakukan oleh Admin

Fungsi yang diuji	Cara Pengujian	Hasil yang diharapkan	Hasil Pengujian
Login	Admin memasukkan <i>username</i> dan <i>password</i>	Admin masuk ke halaman admin	Berhasil
Input data guru	Pilih menu data guru, pilih tambah data	Data guru bertambah	Berhasil
Ubah data guru	Pilih menu data guru, pilih edit data	Data guru telah berubah	Berhasil
Hapus data guru	Pilih menu data guru, pilih hapus data	Data guru telah dihapus	Berhasil
Input data siswa	Pilih menu data siswa, pilih tambah data	Data siswa telah bertambah	Berhasil
Hapus data siswa	Pilih menu data siswa, pilih hapus data	Data siswa telah dihapus	Berhasil
Ubah data siswa	Pilih menu data siswa, pilih edit data siswa	Data siswa telah berubah	Berhasil
Input data kelas	Pilih menu data kelas, pilih tambah data	Data kelas telah bertambah	Berhasil
Hapus data kelas	Pilih menu data kelas, pilih hapus data	Data kelas telah dihapus	Berhasil
Ubah data siswa	Pilih menu data kelas, pilih edit data	Data kelas telah berubah	Berhasil
Input jadwal	Pilih menu jadwal, pilih tambah jadwal	Data jadwal telah bertambah	Berhasil
Edit jadwal	Pilih menu jadwal, pilih	Data jadwal telah	Berhasil

	edit jadwal	berubah	
Hapus jadwal	Pilih menu jadwal, pilih hapus jadwal	Data jadwal telah dihapus	Berhasil
Input berita	Pilih menu berita, pilih tambah berita	Data berita telah ditambah	Berhasil
Hapus berita	Pilih menu berita, pilih hapus berita,	Data berita telah dihapus	Berhasil
Edit berita	Pilih menu berita, pilih edit berita	Data berita telah berubah	Berhasil
Cetak laporan kelas	Pilih menu laporan, pilih laporan kelas, pilih cetak laporan	Laporan telah dicetak	Berhasil
Cetak laporan guru	Pilih menu laporan, pilih cetak laporan guru, pilih cetak laporan	Laporan telah dicetak	Berhasil
Cetak laporan siswa	Pilih menu laporan, pilih cetak laporan siswa, pilih kelas siswa, pilih cetak laporan	Laporan telah dicetak	Berhasil
Cetak laporan jadwal	Pilih menu laporan, pilih cetak laporan jadwal, pilih cetak laporan	Laporan telah dicetak	Berhasil
Cetak laporan nilai	Pilih menu laporan, pilih cetak laporan nilai, pilih kelas, pilih cetak laporan	Laporan telah dicetak	Berhasil

Tabel 4.2 Tabel Pengujian yang dilakukan oleh Siswa

Fungsi yang diuji	Cara Pengujian	Hasil yang diharapkan	Hasil Pengujian
Login	Siswa memasukkan <i>username</i> dan <i>password</i>	Siswa masuk ke halaman siswa	Berhasil
Lihat jadwal	Pilih menu jadwal, pilih semester	Siswa dapat melihat jadwal pelajarannya berdasarkan semester yang dipilih	Berhasil
Lihat nilai	Pilih menu nilai, pilih semester	Siswa dapat melihat nilainya berdasarkan semester yang dipilih	Berhasil
Edit biodata	Pilih menu biodata, pilih edit biodata	Siswa dapat mengedit biodatanya jika	Berhasil

		terdapat data yang kurang sesuai	
Cetak laporan nilai	Pilih menu nilai, pilih semester, pilih cetak	Siswa dapat mencetak nilainya berdasarkan semester yang dipilihnya	Berhasil
Cetak laporan jadwal	Pilih menu jadwal, pilih semester, pilih cetak	Siswa dapat mencetak jadwal berdasarkan semester yang dipilihnya	Berhasil
Logout	Tekan tombol logout	Siswa dapat logout dari halaman siswa dan kembali ke halaman awal	berhasil

Tabel 4.3 Tabel Pengujian yang dilakukan oleh Guru

Fungsi yang diuji	Cara Pengujian	Hasil yang diharapkan	Hasil Pengujian
Login	Guru memasukkan <i>username</i> dan <i>password</i>	Guru berhasil masuk ke halaman guru	Berhasil
Lihat jadwal	Pilih menu jadwal, pilih semester	Guru dapat melihat jadwal pelajarannya berdasarkan semester yang dipilih	Berhasil
input nilai	Pilih menu nilai, pilih kelas, pilih siswa	Guru dapat menginputkan nilai siswa berdasarkan kelas dan siswa yang dipilihnya	Berhasil
Edit biodata	Pilih menu biodata, pilih edit biodata	Guru dapat mengedit biodatanya jika terdapat data yang kurang sesuai	Berhasil
Cetak laporan jadwal	Pilih menu jadwal, pilih semester, pilih cetak	Guru dapat mencetak jadwal berdasarkan semester yang	Berhasil

		dipilihnya	
Logout	Tekan tombol logout	Guru dapat logout dari halaman siswa dan kembali ke halaman awal	berhasil

Tabel 4.4 Tabel Pengujian yang dilakukan oleh Kepala Sekolah

Fungsi yang diuji	Cara Pengujian	Hasil yang diharapkan	Hasil Pengujian
Login	Kepala sekolah memasukkan <i>username</i> dan <i>password</i>	Kepala sekolah berhasil masuk ke halaman kepala sekolah	Berhasil
lihat data kelas	Pilih menu kelas	Kepala sekolah berhasil melihat data kelas	Berhasil
Lihat data guru	Pilih menu guru	Kepala sekolah dapat melihat data guru	Berhasil
Lihat data siswa	Pilih menu siswa, pilih kelas	Kepala sekolah dapat melihat data siswa	Berhasil
Lihat data jadwal	Pilih menu jadwal, pilih kelas, pilih hari, pilih semester	Kepala sekolah dapat melihat data jadwal berdasarkan kelas, hari, dan semester yang dipilihnya	Berhasil
Lihat data berita	Pilih menu berita	Kepala sekolah dapat melihat berita	Berhasil
Lihat data nilai	Pilih menu nilai, pilih kelas, pilih siswa, pilih semester	Kepala sekolah dapat melihat data nilai berdasarkan kelas, siswa, dan semester yang dipilihnya	Berhasil
Cetak data guru	Pilih menu guru, pilih cetak	Kepala sekolah dapat mencetak laporan data guru	Berhasil
Cetak data siswa	Pilih menu siswa, pilih cetak	Kepala sekolah dapat mencetak laporan data siswa berdasarkan kelas yang dipilihnya	Berhasil

Cetak data jadwal	Pilih menu jadwal, pilih, kelas, pilih hari, pilih semester, pilih tampilkan, pilih cetak	Kepala sekolah dapat mencetak laporan jadwal berdasarkan kelas, hari, semester yang dipilihnya	Berhasil
Cetak data nilai	Pilih menu nilai, pilih kelas, pilih siswa, pilih tampilkan, pilih cetak	Kepalas sekolah dapat mencetak laporan nilai berdasarkan kelas, hari, siswa, dan semester yang dipilihnya	Berhasil
Logout	Tekan tombol logout	Kepala sekolah berhasil keluar dan kembali ke halaman awal	Berhasil

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pengujian sistem informasi akademik di SMA YPI Tunas Bangsa yang telah dilakukan oleh penulis, maka dapat diambil beberapa kesimpulan, yaitu:

1. Sistem Informasi Akademik di SMA YPI Tunas Bangsa Palembang dengan Metode MVC (*Model View Controller*) berbasis web merupakan proses menganalisa data pada SMA YPI Tunas Bangsa Palembang untuk mengetahui proses akademik, guna merancang sistem akademik *online* dan untuk mengetahui kekurangan yang ada di sistem yang sedang berjalan.
2. Sistem Informasi Akademik di SMA YPI Tunas Bangsa Palembang dengan Metode MVC (*Model View Controller*) berbasis web merupakan rancangan pembuatan program untuk keperluan akademik di SMA YPI Tunas Bangsa Palembang yang mempunyai beberapa modul diantaranya admin, guru, siswa, dan kepala sekolah. Modul admin merupakan orang yang dapat menginput, mengedit, dan menghapus data guru, data jadwal, data siswa, dan data berita. Modul siswa adalah orang yang dapat melihat dan mencetak nilai dan jadwal mata pelajarannya. Modul guru adalah orang yang dapat menginputkan, menghapus, dan mengubah nilai siswa serta dapat mencetak jadwal mengajarnya. Modul kepala sekolah adalah

orang yang dapat melihat segala data yang ada mulai dari data guru, siswa, nilai, jadwal, berita, dan kelas.

3. Siswa dapat melihat dan mencetak nilai serta jadwal mata pelajarannya secara *online*.
4. Guru dapat melakukan pengolahan nilai secara *online*.

5.2 Saran

Pada Sistem Informasi Akademik di SMA YPI Tunas Bangsa Palembang dengan Metode MVC (*Model View Controller*) berbasis Web masih terdapat kekurangan, oleh karena itu perlu pengembangan lebih lanjut dalam merancang sistem, agar sistem yang dibangun lebih sempurna. Adapun beberapa hal yang perlu disarankan untuk penelitian selanjutnya adalah:

1. Dalam perancangan sistem informasi akademik yang diusulkan seperti pemberitahuan pembayaran SPP dari bank masih belum tersedia, sehingga dalam pengecekan siswa yang sudah membayar SPP masih dicetak secara manual.
2. Dapat ditambahkan sistem *e-learning* atau pembelajaran *online*.
3. Pendataan alumni dapat ditambahkan dengan mengumpulkan semua data siswa yang telah tamat/lulus sebelum program ini digunakan.
4. Dapat dikembangkan lagi untuk *sms gateway*, agar para orang tua dapat mengetahui nilai para anaknya.