

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Penelitian tentang sistem informasi akademik sudah banyak dilakukan diantaranya: Penelitian yang dilakukan oleh Liatmaja pada tahun 2013 dengan judul “Sistem Informasi Akademik Berbasis *Web* pada Lembaga Bimbingan Belajar Be Exxellent Pacitan” yang mengembangkan suatu sistem pengelolaan akademik meliputi informasi tentang nilai UTS, nilai UAS, nilai *Tryout*, serta jadwal UTS, jadwal UAS, dan jadwal *Tryout*. Hasil penelitian ini adalah dapat mendukung kerja pengelola akademik ataupun sebagai acuan bagi pihak lembaga dalam melakukan perbaikan kinerja layanan terhadap siswa didik.

Penelitian yang dilakukan oleh Utama pada tahun 2011 dengan judul “Sistem Informasi Berbasis Web Jurusan Sistem Informasi Akademik Ilmu Komputer Universitas Sriwijaya” merupakan suatu sistem yang memudahkan mahasiswa dalam melakukan kegiatan pengisian Kartu Rencana Studi, dan mendapatkan informasi laporan nilai, selain itu dengan berbasiskan *web* maka informasi data dapat diakses dengan waktu yang tidak ditentukan. Pada sistem ini, menu hanya dapat diakses oleh user tertentu yaitu siswa dan administrator. Pada hasil penelitian ini telah dikembangkan sebuah Sistem Informasi Berbasis Web dengan studi kasus pada Jurusan Sistem Informasi Fakultas Ilmu Komputer. Dimana dalam membangun sistem ini digunakan alat bantu pengembangan sistem yaitu *Data Flow Diagram* (DFD), *Context Diagram*, *Entity Relationship Diagram* (ERD), dan *Flowchart* serta dengan menggunakan bahasa pemrograman PHP dan HTML dan MySQL sebagai databasenya.

Penelitian yang dilakukan oleh Hasbi pada tahun 2015 melakukan penelitian dengan judul “Perancangan Sistem Informasi Akademik pada SMK Negeri 2 Simbang Maros” untuk memudahkan dalam proses pelayanan akademik disekolah. Data diperoleh dengan teknik observasi dan dokumentasi serta studi literatur. Instrumen penelitian menggunakan lembar observasi dan dokumentasi. Penelitian ini merupakan jenis penelitian pengembangan yang menggunakan yang mengembangkan suatu perangkat lunak (*software development*) dengan model

pengembangan *prototyping*. Analisis data menggunakan analisis deskriptif. Hasil penelitian ini menunjukkan bahwa sistem ini memberikan kemudahan dalam pengaksesan ataupun manajemen data dan informasi akademik sehari-hari, seperti informasi data pelajaran, data guru, data siswa, data kelas dan data nilai siswa.

Dalam penelitian ini peneliti akan merancang sebuah sistem informasi akademik terintegrasi satu sama lain seperti data guru, data siswa, data walimurid, informasi absensi siswa, informasi data mata pelajaran, informasi data nilai dan informasi pengumuman. Peneliti akan membuat sebuah sistem informasi akademik yang memberikan hak akses guru, siswa, dan walimurid sehingga dapat mempermudah untuk mendapatkan informasi akademik secara *realtime* dan *up to date*.

2.2 Landasan Teori

2.2.1. Konsep Dasar Sistem

Sistem adalah sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan atau untuk mengendalikan organisasi (Ladjamudin, 2013).

Sistem adalah kumpulan atau himpunan dari unsur-unsur variabel-variabel yang saling terkait, saling berinteraksi, dan saling tergantung satu sama lain untuk mencapai tujuan. Selain itu, sistem juga didefinisikan sebagai sekumpulan objek-objek yang saling berinteraksi, serta hubungan antar objek bisa dilihat sebagai satu kesatuan yang dirancang untuk mencapai satu tujuan yang telah ditetapkan (Tohari, 2013).

Sistem sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu (Sutabri, 2012).

2.2.1.1 Klasifikasi Sistem

Sistem adalah suatu bentuk integrasi antara satu komponen dengan komponen lain karena memiliki sasaran yang berbeda untuk setiap kasus yang terjadi didalam sistem (Ladjamudin, 2013) Oleh sebab itu, sistem klasifikasi dari beberapa sudut pandang antaranya:

a. Sistem Abstrak dan Sistem Fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik.

b. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang dan malam serta pergantian musim. Sedangkan sistem buatan merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut *Human Machine System*. Salah satu contohnya adalah sistem informasi berbasis komputer, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

c. Sistem Determinasi dan Sistem *Probabilistic*

Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem *deterministic*. Sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur *probabilistic*.

d. Sistem Terbuka dan Sistem Tertutup

Sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk sub sistem lain. Sedangkan sistem tertutup adalah sistem yang tidak berhubungan dan tidak dipengaruhi oleh lingkungan luarnya, sistem ini bekerja secara otomatis tanpa adanya campur dari pihak luar.

2.2.1.2 Karakteristik Sistem

Suatu sistem memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem (Ladjamudin, 2013), yaitu:

a. Komponen Sistem (*Component*)

Komponen sistem adalah terdiri dari sejumlah komponen serta saling berinteraksi yang bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem tersebut dapat berupa bentuk subsistem. Setiap subsistem memiliki sifat-sifat yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat

mempunyai sistem yang lebih besar yang disebut "*supra system*".

b. Batasan Sistem (*Boundary*)

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem lainnya, atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan dan menunjukkan ruang lingkup dari sistem tersebut.

c. Lingkungan Luar Sistem (*Environtment*)

Lingkungan luar adalah bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan 9 lingkungan luar sistem. Lingkungan luar sistem ini dapat menguntungkan, namun dapat juga merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi bagi sistem tersebut, sedangkan lingkungan luar yang merugikan harus dikendalikan, kalau sistem tersebut tidak dikendalikan maka akan mengganggu kelangsungan hidup suatu sistem tersebut.

d. Penghubung Sistem (*interface*)

Media yang menghubungkan antara suatu sistem dengan subsistem yang lainnya disebut dengan penghubung sistem atau interface. Penghubung sistem ini memungkinkan sumber daya yang mengalir dari suatu subsistem yang lainnya. Keluaran suatu subsistem akan menjadi masukan untuk subsistem yang lain, dengan melewati penghubung. Dengan demikian terjadi suatu integrasi sistem yang membentuk satu kesatuan.

e. Masukan Sistem (*input*)

Energi yang dimasukkan kedalam sistem disebut dengan masukan sistem, yang dapat berupa pemeliharaan (*maintance input*) dan sinyal (*signal Input*) sebagai contoh, didalam suatu sistem unit komputer, "*program*" adalah *maintance input* yang digunakan untuk mengoperasikan komputer. Sementara "*data*" adalah *signal input* yang akan diolah menjadi sebuah informasi.

f. Keluaran Sitem (*Output*)

Keluaran Sistem adalah mengolah suatu energi dan di klasifikasikan menjadi suatu keluaran yang berguna, keluaran dapat merupakan masukan

untuk subsistem yang lain seperti sistem informasi.

g. *Pengolahan Sistem (process)*

Pengolahan sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

h. *Sasaran Sistem (Objective)*

Sebuah sistem memiliki tujuan atau sasaran yang pasti dan bersifat *deterministic*.

2.2.2 Konsep Dasar Informasi

Informasi memiliki peranan yang penting dalam organisasi, ibarat darah yang mengalir didalam tubuh suatu organisasi. Suatu sistem yang kurang mendapatkan informasi akan sulit berkembang bahkan dapat menjadi mati.

Sebelum mendefinisikan informasi, penulis memaparkan memaparkan definisi dari data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kesatuan nyata adalah berupa objek nyata seperti tempat, benda, dan orang yang betul-betul ada dan terjadi. Untuk pengambilan keputusan bagi manajemen, maka faktor-faktor tersebut harus diolah lebih lanjut untuk menjadi suatu informasi (Ladjamudin, 2005).

Informasi adalah data yang telah diproses ke dalam suatu bentuk yang mempunyai arti bagi si penerima dan mempunyai nilai nyata dan terasa bagi keputusan saat itu atau keputusan mendatang (Gordon B. Davis, 2012).

Informasi adalah sekumpulan fakta (data) yang diorganisasikan dengan cara tertentu sehingga mereka mempunyai arti bagi si penerima (Sutarman, 2012).

Adapun kualitas informasi sangat dipengaruhi atau ditentukan oleh tiga hal pokok, yaitu akurasi (*accuracy*), relevansi (*relevancy*), dan tepat waktu (*timeliness*) (Mulyanto, 2009).

a. *Akurasi (Accurate)*

Informasi harus bebas dari kesalahan kesalahan dan tidak menyesatkan, dalam hal ini informasi harus jelas mencerminkan maksudnya.

b. *Relevansi (Relevance)*

Informasi harus mempunyai manfaat untuk pemakainya, dimana relevansi informasi untuk tiap-tiap individu berbeda tergantung pada yang menerima dan

yang membutuhkan. Nilai informasi ditentukan oleh dua hal yaitu manfaat dan biaya. Suatu informasi dikatakan bernilai apabila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.

c. Tepat Waktu (*Timeliness*)

Informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usung tidak akan memiliki nilai lagi karena informasi merupakan suatu landasan dalam mengambil sebuah keputusan di mana bila mengambil keputusan terlambat maka akan bersifat fatal untuk organisasi.

2.2.3 Konsep Dasar Sistem Informasi

Secara umum definisi sistem informasi adalah sekelompok elemen-elemen dalam suatu organisasi yang saling berintegrasi dengan menggunakan masukan,, proses, keluaran, dengan maksud yang sama untuk mencapai suatu tujuan dan dapat digunakan untuk membantu pengambilan keputusan yang tepat (Jeffrey L. Whitten, 2006).

Sistem informasi adalah Sistem dapat didefinisikan dengan mengumpulkan, memproses, menyimpan, menganalisis, menyebarkan informasi untuk tujuan tertentu. Seperti sistem lainnya, sebuah sistem informasi terdiri atas input (data, instruksi) dan *output* (laporan, kalkulasi) (Sutarman 2012:13).

Lebih lanjut lagi, sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan-kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan (Sutabri, 2005).

Sistem informasi mempunyai enam buah komponen (Jogiyanto H, 2009), yaitu:

a. Komponen Masukan (*Input*)

Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

b. Komponen Model

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

c. Komponen Keluaran (*Output*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

d. Komponen Teknologi

Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian sistem secara keseluruhan. Pada blok ini terdiri dari 3 bagian utama, yaitu teknisi (*humanware* atau *brainware*), perangkat lunak (*software*) dan perangkat keras (*hardware*).

e. Komponen Basis Data (*Database*)

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut dengan DBMS (*Database Management Systems*).

f. Komponen Kendali

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.2.4 Aplikasi Berbasis Mobile

Aplikasi berbasis *mobile* adalah aplikasi yang telah dirancang khusus untuk *platform mobile* (misalnya *iOS*, *android*, atau *windows mobile*). Dalam banyak kasus, aplikasi *mobile* memiliki *user interface* dengan mekanisme interaksi unik yang disediakan oleh *platform mobile*, interoperabilitas dengan sumber daya berbasis web yang menyediakan akses ke beragam informasi yang relevan dengan aplikasi, dan kemampuan pemrosesan lokal untuk pengumpulan, analisis, dan format informasi dengan cara yang paling cocok untuk *platform*

mobile. Selain itu aplikasi *mobile* menyediakan kemampuan penyimpanan *persisten* dalam *platform*. (Pressman dan Bruce, 2014:9).

Android merupakan perangkat bergerak pada sistem operasi untuk telepon seluler yang berbasis linux (Teguh Arifianto, 2011).

Android adalah sebuah kumpulan perangkat lunak untuk perangkat *mobile* yang mencakup sistem operasi, *middleware* dan aplikasi utama *mobile*. (Safaat, 2010)

Android memiliki empat karakteristik sebagai berikut :

a. Terbuka

Android dibangun untuk benar-benar terbuka sehingga sebuah aplikasi dapat memanggil salah satu fungsi inti ponsel seperti membuat panggilan, mengirim pesan teks, menggunakan kamera dan lain-lain. Android merupakan sebuah mesin virtual yang dirancang khusus untuk mengoptimalkan sumber daya memori dan perangkat keras yang terdapat di dalam perangkat. Android merupakan *open source*, dapat secara bebas diperluas untuk memasukkan teknologi baru yang lebih maju pada saat teknologi tersebut muncul. *Platform* ini akan terus berkembang untuk membangun aplikasi *mobile* yang inovatif.

b. Semua aplikasi dibuat sama

Android tidak memberikan perbedaan terhadap aplikasi utama dari telepon dan aplikasi pihak ketiga (*third-party application*). Semua aplikasi dapat dibangun untuk memiliki akses yang sama terhadap kemampuan sebuah telepon dalam menyediakan layanan dan aplikasi yang luas terhadap para pengguna.

c. Memecahkan hambatan pada aplikasi

Android memecah hambatan untuk membangun aplikasi yang baru dan inovatif. Misalnya, pengembang dapat menggabungkan informasi yang diperoleh dari *web* dengan data pada ponsel seseorang seperti kontak pengguna, kalender atau lokasi geografis.

d. Pengembangan aplikasi yang cepat dan mudah

Android menyediakan akses yang sangat luas kepada pengguna untuk menggunakan aplikasi yang semakin baik. Android memiliki sekumpulan tools yang dapat digunakan sehingga membantu para pengembang dalam meningkatkan produktivitas pada saat membangun aplikasi yang dibuat.

2.2.5 Perangkat Lunak Pendukung

2.2.5.1 Android Studio

Android Studio merupakan sebuah *Integrated Development Environment* (IDE) untuk *platform* Android. Android Studio ini diumumkan pada tanggal 16 Mei 2013 pada Konferensi *Google I/O* oleh Produk Manajer Google, Ellie Powers, Android Studio bersifat *free* dibawah *Apache License* 2.0. Android Studio awalnya dimulai dengan versi 0.1 pada bulan mei 2013, kemudian dibuat versi beta 0.8 yang dirilis pada bulan juni 2014. Yang paling stabil dirilis pada bulan desember 2014, dimulai dari versi 1.0. Berbasiskan *JetBrains' IntelliJ IDEA*, Android Studio didesain khusus untuk *Android Development* dikutip pada situs resminya (<http://developer.android.com/studio/intro/index.html?hl=id>).

Sebagai pengembangan dari Eclipse, Android Studio mempunyai banyak fitur-fitur baru dibandingkan dengan Eclipse IDE. Berbeda dengan Eclipse yang menggunakan Ant, Android Studio menggunakan Gradle sebagai build environment. Fitur-fitur lainnya adalah sebagai berikut :

- a. Menggunakan Gradle-based build system yang fleksibel.
- b. Bisa *mem-build multiple* APK.
- c. *Template support* untuk Google Service dan berbagai macam tipe perangkat.
- d. *Layout editor* yang lebih bagus.
- e. *Built-in support* untuk *Google Cloud Platform*, sehingga mudah untuk integrasi dengan *Google Cloud Messaging* dan *App Engine*.
- f. *Import library* langsung dari *Maven repository*.

Kelebihan Android Studio adalah *software* ini dilengkapi dengan *intelligent code editor* yang mampu mengolah dan menganalisis kode secara lengkap yang menjadikan *developer* semakin produktif. Selain itu pula, Android Studio dilengkapi dengan *Code Templates* dan *Github integration* yang memudahkan *developer* android dalam mengembangkan aplikasi mereka dari sampel-sampel kode yang disediakan ataupun mengimpornya dari *Github*. Android Studio dilengkapi dengan *emulator* yang mencakup semua *devices*, baik ukuran maupun bentuk. Keunggulannya tersebut jelas memudahkan *developer* untuk melihat hasil *project* mereka dari berbagai *devices* yang ada.

2.2.5.2 *Android Software Development Kit (SDK)*

Android SDK adalah *tool API (Application Programming Interface)* yang diperlukan untuk mulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang release oleh *Google*. Saat ini di sediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada *platform Android* menggunakan bahasa pemograman java (Safaat H, 2011). Beberapa fitur-fitur Android yang paling penting adalah:

- a. *Framework* aplikasi yang mendukung penggantian komponen dan *reusable*.
- b. Mesin *Virtual Dalvik* dioptimalkan untuk perangkat mobile.
- c. *Integrated browser* berdasarkan *Engine Open Source WebKit*.
- d. Grafis yang dioptimalkan dan didukung oleh libraries grafis 2D, grafis 3D berdasarkan spesifikasi opengl ES 1,0 (Opsional akselerasi hardware).
- e. SQLite untuk penyimpanan data (*database*).
- f. Media Support yang mendukung audio, video, dan gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF), GSM Telephony (tergantung *hardware*).
- g. Bluetooth, EDGE, 3G, dan WiFi (tergantung hardware) Kamera, GPS, kompas, dan accelerometer (tergantung *hardware*).
- h. Lingkungan development yang lengkap dan kaya termasuk perangkat *emulator*, *tools* untuk *debugging*, profil dan kinerja memori, dan plugin untuk IDE Eclipse.

2.2.5.3 XAMPP

XAMPP adalah paket program web lengkap yang dapat Anda pakai untuk belajar pemrograman web, khususnya PHP dan MySQL (Nugroho 2013:1).

XAMPP adalah perangkat lunak opensource yang diunggah secara gratis dan bisa dijalankan di semua semua operasi seperti windows, linux, solaris, dan mac (Buana, 2014). XAMPP ialah perangkat lunak bebas yang mendukung banyak sistem operasi, merupakan campuran dari beberapa program. Yang mempunyai fungsi sebagai server yang berdiri sendiri (*localhost*), yang terdiri dari program *MySQL database*, *Apache HTTP Server*, dan penerjemah ditulis dalam bahasa

pemrograman *PHP*. Nama XAMPP merupakan singkatan dari X (empat sistem operasi), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia di bawah *GNU General Public License* dan bebas, adalah mudah untuk menggunakan *web server* yang dapat melayani tampilan halaman web yang dinamis. Bagian penting XAMPP adalah:

- a. *htdocs* adalah folder di mana Anda meletakkan file yang akan dijalankan, seperti file *PHP*, *HTML* dan script lainnya.
- b. *phpMyAdmin* adalah bagian untuk mengelola *database MySQL* yang dikomputer. Untuk membukanya, membuka browser dan ketik alamat *http: // localhost / phpMyAdmin*, halaman *phpMyAdmin* akan muncul.
- c. Control Panel yang berfungsi untuk mengelola layanan (*service*) XAMPP. Seperti *stop service* (berhenti), atau mulai (mulai).

2.2.5.4 MySQL

MySQL adalah software atau program Database Server”. Sedangkan SQL adalah bahasa pemrogramannya, bahasa permintaan (*query*) dalam database server termasuk dalam MySQL itu sendiri. SQL juga dipakai dalam software database server lain, seperti SQL Server, Oracle, PostgreSQL dan lainnya (Nugroho 2013:1)

MySQL merupakan *software RDBMS* (atau server *database*) yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user (*multi user*), dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*). (Raharjo, 2011)

MySQL Merupakan database server yang paling sering digunakan dalam pemrograman *PHP*. MySQL digunakan untuk menyimpan data dalam database dan memanipulasi data-data yang diperlukan. Manipulasi data tersebut berupa menambah, mengubah, dan menghapus data yang berada dalam database (Buana, 2014). *PHP MySQL* sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Contoh MySQL adalah *APACHE, XAMPP, WampServer*. Mysql menawarkan berbagai keunggulan dibandingkan database server

lain, berikut ini ada beberapa keunggulan Mysql:

- a. Mampu menangani jutaan user dalam waktu yang bersamaan.
- b. Mampu menangani lebih dari 50.000.000 record.
- c. Sangat cepat mengeksekusi perintah.
- d. Memiliki *User Priveleges System* yang mudah dan efisien.

2.2.5.5 PHP

PHP adalah bahasa *server side scripting* yang menyatu dengan *HTML* untuk membuat halaman web yang dinamis. Karena PHP merupakan *server side scripting* maka sintaks dan perintah-perintah PHP akan dieksekusi di server kemudian hasilnya akan dikirimkan ke *browser* dengan format *HTML* (Arief, 2011).

Personal Home Page merupakan bahasa skrip yang tertanam dalam *HTML* untuk dieksekusi bersifat *server side*". PHP termasuk dalam open source product, sehingga source code PHP dapat diubah dan didistribusikan secara bebas (Nugroho, 2006).

PHP juga dapat berjalan pada berbagai web server seperti IIS (*Internet Information Server*), PWS (*Personal Web Server*), *Apache*, *Xitami*. PHP juga mampu berjalan di banyak sistem operasi yang beredar saat ini, diantaranya : Sistem Operasi Microsoft Windows (semua versi), Linux, Mac Os, Solaris. PHP dapat dibangun sebagai modul *web server Apache* dan sebagai *binary* yang dapat berjalan sebagai CGI (*Common Gateway Interface*).

PHP dapat mengirim *HTTP header*, dapat mengatur *cookies* , mengatur *authentication* dan *redirect user*. Salah satu keunggulan yang dimiliki PHP adalah kemampuannya untuk melakukan koneksi ke berbagai macam *software* sistem manajemen basis data atau *Database Management Sistem* (DBMS), sehingga dapat menciptakan suatu halaman web dinamis. PHP mempunyai koneksitas yang baik dengan beberapa DBMS seperti *Oracle*, *Sybase*, *mSQL*, *MySQL*, *Microsoft SQL Server*, *Solid*, *PostgreSQL*, *Adabas*, *FilePro*, *Velocis*, *dBase*, *Unix dbm*, dan tidak terkecuali semua database *ber-interface ODBC*. Hampir seluruh aplikasi berbasis *web* dapat dibuat dengan PHP. Namun kekuatan utama adalah konektivitas basis

data dengan web. Dengan kemampuan ini kita akan mempunyai suatu sistem basis data yang dapat diakses (Nugroho, 2006).

2.2.6 Perancangan Database

2.2.6.1 Definisi Basis Data

Basis Data atau *Database* adalah sekumpulan *file* yang saling berhubungan dan terorganisasi atau kumpulan *record-record* yang menyimpan dan hubungan diantaranya (Sutarman, 2012, p. 15). Ladjamudin (2013, p. 129) menjelaskan bahwa *Database* adalah sekumpulan data *store* (bisa dalam jumlah yang sangat besar) yang tersimpan dalam *magnetic disk*, *official disk*, *magnetic drum*, atau media penyimpanan sekunder lainnya.

Basis Data atau *Database* adalah sekumpulan *file* yang saling berhubungan dan terorganisasi atau kumpulan *record-record* yang menyimpan dan hubungan diantaranya (Sutarman, 2012, p. 15). Ladjamudin (2013, p. 129) menjelaskan bahwa *Database* adalah sekumpulan data *store* (bisa dalam jumlah yang sangat besar) yang tersimpan dalam *magnetic disk*, *official disk*, *magnetic drum*, atau media penyimpanan sekunder lainnya.

Berdasarkan pendapat para ahli diatas, penulis menyimpulkan bahwa Basis Data adalah sekumpulan *file* atau berkas yang saling berhubungan dan tersimpan dalam sebuah media penyimpanan. Selain berisi data, *database* juga berisi *metadata*, yaitu data yang menjelaskan tentang struktur dari data itu sendiri. Contohnya adalah informasi yang diperoleh tentang nama-nama kolom dan tipe data yang ada pada sebuah tabel, data nama kolom dan tipe yang ditampilkan itu disebut *metadata*.

2.2.6.2 Database Management System (DBMS)

Database Manajemen System (DBMS) adalah sebuah sistem perangkat lunak yang mengizinkan pengguna untuk mendefinisikan, membuat, memelihara dan mengontrol akses ke dalam basis data (Connoly & Begg, 2010).

Fasilitas yang disediakan oleh DBMS adalah sebagai berikut:

- a. Mengizinkan pengguna untuk mendfinisikan basis data, dengan melalui *Data Definition Language* (DDL). DDL mengizinkan pengguna untuk

menentukan tipe, struktur, serta kendala data yang nantinya akan disimpan ke dalam baris data.

- b. Mengizinkan pengguna untuk melakukan menambah, mengubah, menghapus dan mengambil data dari basis data tersebut, dengan menggunakan *Data Manipulation Language* (DML). Standard bahasa dari DBMS ialah *Structured Query Language* (SQL).
- c. Menyediakan akses kontrol ke dalam basis data, seperti:
 1. Sistem keamanan, yang dapat mencegah pengguna yang tidak diberi kuasa untuk mengakses basis data.
 2. Sistem integritas, yang dapat menjaga konsistensi dari data yang tersimpan.
 3. Sistem kontrol konkurensi, yang mengizinkan berbagi akses dengan basis data.
 4. Sistem kontrol pemulihan, jika terjadi kegagalan perangkat keras atau perangkat lunak maka sistem control pemulihan ini dapat mengembalikan basis data ke keadaan yang konsisten dari yang sebelumnya.

2.2.6.3 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya dilakukan oleh sistem analis dalam tahap analisis persyaratan (*requirement analysis*) proyek pengembangan sistem. Sementara seolah-olah teknik diagram atau alat peraga memberikan dasar untuk desain relasi *database* yang mendasari sistem informasi yang dikembangkan (Brady & Loonam, 2010).

ERD menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data, pada dasarnya ada tiga macam notasi yang digunakan yaitu:

- a. Entitas

Adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat. Sebagai contoh pelanggan, pekerja dan lain-lain.

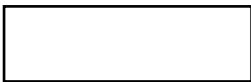
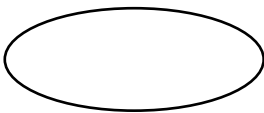
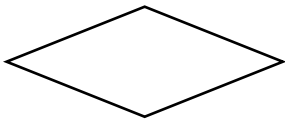

b. Relasi

Hubungan antara dua jenis entitas dan direpresentasikan sebagai garis lurus yang menghubungkan dua entitas. Contohnya adalah mahasiswa mendaftar sebagai anggota suatu organisasi, maka mendaftar adalah sebagai relasinya.

c. Atribut

Atribut memberikan informasi lebih rinci tentang jenis entitas. Contohnya seperti atribut id mahasiswa pada entitas mahasiswa. Setiap entitas bisa memiliki lebih dari satu atribut.

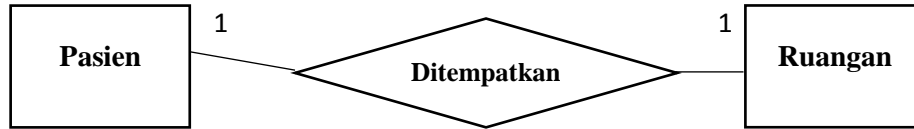
Tabel 2. 1 Simbol ERD

Simbol	Nama	Keterangan
	Entitas	Entitas adalah suatu apa saja yang ada dalam sistem, nyata maupun abstrak dimana data tersimpan atau dimana terdapat data.
	Atribut	Atribut adalah sifat, atau karakteristik, atau elemen tiap entitas maupun <i>relationship</i> .
	<i>Relationship</i>	<i>Relationship</i> adalah hubungan alamiah yang terjadi antara entitas.
	<i>Link</i>	Menghubungkan antara entitas satu dengan entitas lainnya.

Menurut Brady dan Loonam (2010) kardinalitas rasio ERD digunakan untuk menjelaskan jumlah maksimum hubungan antara entitas yang satu dengan yang lainnya. Berikut ini adalah penjelasan mengenai kardinalitas rasio ERD.

a. *One to One* (1:1)

Setiap anggota entitas A hanya boleh berhubungan dengan satu anggota entitas B, begitu pula sebaliknya.



Gambar 2. 1 Kardinalitas Rasio *One to One*

b. *One to Many* (1:M)

Setiap anggota entitas A dapat berhubungan dengan lebih dari satu anggota entitas B, tetapi tidak sebaliknya.



Gambar 2. 2 Kardinalitas Rasio *One to Many*

c. *Many to Many* (M:M)

Setiap entitas A dapat berhubungan dengan banyak entitas himpunan entitas B dan demikian pula sebaliknya.



Gambar 2. 3 Kardinalitas Rasio *Many to Many*

2.2.6.4 Logical Record Structure (LRS)

LRS (*Logical Record Structure*) adalah representasi dari struktur record-record pada tabel-tabel yang terbentuk dari hasil antar himpunan entitas”. *Logical record structure* dibentuk dengan nomor dari *type record*. *Logical record structure*

terdiri dari link-link diantara *tipe record*. Link ini menunjukkan arah dari satu tipe record lainnya. Banyak link dari LRS yang diberi tanda *field-field* yang kelihatan pada kedua link tipe record. Penggambaran LRS mulai dengan menggunakan model yang dimengerti. Dua metode yang dapat digunakan, dimulai dengan hubungan kedua model yang dapat dikonversikan ke LRS. Metode yang lain dimulai dengan ER-diagram dan langsung dikonversikan ke LRS. (Riyanto 2009:22).

2.2.6.5 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek (Rosa & Shalahudin, 2014).

2.2.6.6 Diagram UML

Menurut Rosa dan Shalahudin (2014, p. 140), pada UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Berikut ini penjelasan singkat dari pembagian kategori tersebut.

- a. *Structure diagram*, yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan. *Structure diagram* terdiri dari *Class Diagram*, *Object Diagram*, *Component Diagram*, *Composite Structure Diagram*, *Package Diagram* dan *Deployment Diagram*.
- b. *Behavior diagram*, yaitu kumpulan diagram yang digunakan untuk

menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem. Behavior diagram terdiri dari *Use Case Diagram*, *Activity Diagram*, *State Machine System*.

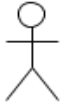
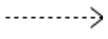
- c. *Interaction diagram*, yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem. *Interaction diagram* terdiri dari *Sequence Diagram*, *Communication Diagram*, *Timing Diagram*, *Interaction Overview Diagram*.









Namun, pada penelitian ini penulis hanya menggunakan 4 diagram, yaitu *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram* dan *Class Diagram*.

2.2.6.7 Use Case Diagram

Use case diagram merupakan permodelan untuk kelakuan (*behavior*) sistem yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Rosa & Shalahudin, 2014).

Tabel 2. 2 Simbol-simbol Use Case Diagram



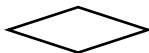


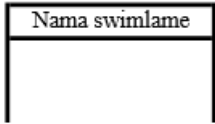


NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri(<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).

3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

2.2.6.8 Activity Diagram

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu di perhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa & Shalahudin, 2014).

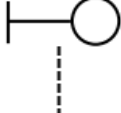


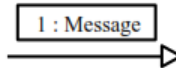

Tabel 2. 3 Simbol – Simbol Activity Diagram



No	Gambar	Nama	Keterangan
1		Status Awal	Status awal aktifitas sistem, sebuah diagram aktifitas memiliki sebuah status awal.
2		Aktivitas	Aktivitas yang dilakukan sistem aktivitas biasanya diawali dengan kata kerja.
3		Percabanga	Menggambarkan cabang suatu keputusan.
4		Penggabungan/ <i>join</i>	Asosiasi penggabungan dimana lebih dari satu aktifitas digabungkan menjadi satu.
5		Status akhir	Status akhir yang dilakukan sistem, sebuah diagram aktifitas memiliki status akhir.
6		<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktifitas yang terjadi.
7		<i>Fork</i>	Digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel.
8		<i>Join</i>	Digunakan untuk menunjukkan kegiatan yang digabungkan.

2.2.6.9 Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dengan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan *sequence diagram* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat *sequence diagram* juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Banyaknya *sequence diagram* yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup dalam *sequence diagram* sehingga semakin banyak *use case* yang didefinisikan maka *sequence diagram* yang harus dibuat juga semakin banyak (Rosa & Shalahudin, 2014).

Tabel 2. 4 Simbol – Simbol Sequence Diagram

No	Gambar	Nama	Keterangan
1		<i>Boundary Lifeline</i>	Menggambarkan hubungan suatu elemen yang berbeda, secara khas merupakan penghubung <i>actor</i> dengan layar.
2		<i>Entity Lifeline</i>	Menggambarkan suatu tempat atau mekanisme yang menangkap pengetahuan atau informasi dalam suatu sistem.
3		<i>Control Lifeline</i>	Menggambarkan suatu pengendalian yang mengorganisir dan menjadwalkan aktivitas elemen-elemen.
4		<i>Message</i>	Perilaku sistem yang menandai adanya suatu alur informasi atau transisi kendali antar elemen.
5		<i>Actor</i>	Menunjukkan seorang pengguna sistem yang memulai alur peristiwa kejadian.

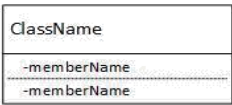



6		<i>Activation Bar</i>	Menggambarkan lamanya suatu pesan diproses.
7		<i>Note</i>	Menunjukkan catatan untuk komentar dari suatu pesan antar elemen.


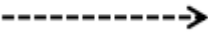

2.2.6.10. Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan *method* atau operasi. Berikut penjelasan atribut dan *method* :

- Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
- Operasi atau *method* adalah fungsi-fungsi yang dimiliki oleh suatu kelas (Rosa & Shalahudin, 2014).

Tabel 2. 5 Simbol-simbol Class Diagram

No	Gambar	Nama	Keterangan
1		<i>Class/Kelas</i>	Kelas pada struktur sistem.
2		<i>Interface</i>	Sama dengan konsep <i>Interface</i> dalam pemrograman berorientasi objek.
3		<i>Association</i>	Relasi antar kelas dengan makna umum, <i>association</i> biasanya juga disertai dengan <i>multiplicity</i> .
4		<i>Direct Association</i>	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, <i>association</i> biasanya juga disertai dengan <i>multiplicity</i> .

5		<i>Generalization</i>	Relasi antar kelas dengan makna generalisasi spesialisasi (umum-khusus).
6		<i>Dependency</i>	Relasi antar kelas dengan makna kebergantungan antar kelas.
7		<i>Aggregation</i>	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>).

2.2.7 Metode Pengembangan Sistem

2.2.7.1 Rapid Application Development (RAD)

Rapid Application Development adalah suatu pendekatan berorientasi objek terhadap pengembangan sistem yang mencakup suatu metode pengembangan serta perangkat-perangkat lunak. RAD bertujuan mempersingkat waktu yang biasanya diperlukan dalam siklus hidup pengembangan sistem tradisional antara perancangan dan penerapan suatu sistem informasi (Kendall, 2010).

Metode yang digunakan oleh penulis adalah Metode *Research and Development*. Dalam bidang pendidikan, bahwa penelitian dan pengembangan (*Research and Development / R&D*) merupakan metode penelitian yang digunakan untuk mengembangkan atau memvalidasi produk-produk yang digunakan dalam pendidikan dan pembelajaran. Penelitian dan Pengembangan merupakan “Jembatan” antara *Basic Research* dan *Applied Research* (Borg and Gall : 1988).

2.2.7.2 Pengembangan Sistem Android

Pengembangan sistem dapat diartikan sebagai sebuah proses pengembangan terstandarisasi yang mendefinisikan satu set aktivitas, metode, praktik terbaik, dan perangkat terotomatisasi yang akan digunakan oleh para pengembangan sistem untuk mengembangkan dan berkesinambungan memperbaiki sistem informasi dan perangkat lunak.

Adapun tahapan-tahapan yang penulis lakukan dalam pengembangan sistem android adalah sebagai berikut:

a. Tahapan perencanaan dan syarat-syarat

Pada tahap ini penulis melakukan pengidentifikasian tujuan aplikasi atau sistem serta untuk mengidentifikasi syarat-syarat informasi yang ditimbulkan dari tujuan-tujuan tersebut yang terdiri dari analisis kebutuhan sistem, tujuan dan syarat-syarat. Proses ini dilakukan untuk mengetahui apa saja syarat-syarat dan kebutuhan yang dibutuhkan dalam pembuatan aplikasi untuk memenuhi tujuan dari pembuatan aplikasi.

b. Tahapan *Design*

Setelah mengetahui definisi aplikasi yang akan dirancang atau dibuat, maka tahapan berikutnya adalah melakukan perancangan (*design*). Perancangan di sini dimaksudkan untuk membuat pemodelan terhadap aplikasi yang mewakili tahapan perencanaan syarat-syarat diatas. Desain yang dimaksud meliputi perancangan proses aplikasi, *database*, dan *interface*. Untuk tampilan antarmuka (*interface*) aplikasi sendiri, penulis melakukan perancangan *Graphical User Interface* (GUI) dari aplikasi ini.

c. Tahapan Implementasi

Setelah tahapan *design* selesai, maka tahap berikutnya adalah mengimplementasikan hasil rancangan tersebut. Implementasi aplikasi dilakukan dengan dua cara pertama pada perangkat lunak *emulator* dan perangkat keras sesungguhnya (ponsel).

2.2.8 Pengujian Testing

Testing adalah proses untuk memeriksa atau mengevaluasi sistem atau komponen sistem secara manual atau terotomatisasi yang bertujuan untuk melakukan verifikasi bahwa sistem tersebut memenuhi persyaratan tertentu atau untuk mengidentifikasikan perbedaan antara *expected result* dan *actual result*. (Singh dan Khan, 2012:146).

Testing melibatkan pengujian dan pencarian *bug* yang terdapat pada *software*, pengujian terhadap performa *website* dan pengujian terhadap *hardware* baru. Bagian penting dalam pengujian adalah mengkaji ulang prototipe dari

tampilan, laporan dan juga hasil lainnya. (O'Brien, George dan Marakas, 2010:435).

2.2.8.1 Black Box Testing

Black Box Testing atau pengujian fungsional, kondisi pengujian dikembangkan berdasarkan fungsionalitas dari program atau sistem yang akan diuji, oleh karena itu, penguji membutuhkan informasi mengenai data input dan output yang diamati, tetapi tidak mengetahui bagaimana program atau sistem tersebut bekerja atau tidak perlu mengetahui bagaimana struktur internal dari program tersebut melakukan eksekusi. Penguji berfokus pada pengujian fungsionalitas dari program terhadap spesifikasi. (Lewis, 2009:39).

2.3 Tinjauan Objek

2.3.1 Profil Sekolah SMK Informatika Citra Bangsa

Sekolah Menengah Kejuruan Informatika Citra Bangsa adalah sebuah lembaga pendidikan formal yang beralamat di jalan. Ciamis - Cirebon, KM. 39, Kecamatan Panawangan, Kabupaten Ciamis, Provinsi Jawa Barat 46255 didirikan pada tahun 2008 dengan Surat Keterangan Pendirian Sekolah Nomor 421.3/9868-Disdik-2008 dan Surat Keterangan Izin Operasional Nomor 421.3/1493-Disdik/2010 yang telah disahkan oleh Kementrian Pendidikan dan Kebudayaan sebagai lembaga pendidikan Sekolah Menengah Kejuruan.

Teknologi Informasi dan Komputer telah menjadi bagian dalam kehidupan masyarakat di era globalisasi dan pasar bebas. Masyarakat saat ini tidak lagi bisa menghindar dari zaman digital. Sebagai jembatan menuju penguasaan teknologi Informasi dan Komunikasi. Yang menjadikan SMK Informatika Citra Bangsa terbentuk. Sekolah Menengah Kejuruan Informatika Citra Bangsa adalah salah satu Sekolah berbasis Teknologi yang mempunyai 3 Program Studi Unggulan Ter-Akreditasi B. Sekolah yang mengedepankan Pendidikan dan Akhlak Mulia Berbasis Penilaian Holistik, menjadikan peserta didik SMK Informatika Citra Bangsa Ciamis Bermartabat, Berkualitas, dan Terpercaya.

Sekolah yang mempunyai letak strategis pada daerah Panawangan Ciamis, yang menjadikan mobilitas Sekolah Menengah Kejuruan Informatika Citra Bangsa

menjadi mudah diakses. Dan di lengkapi pula dengan pendidik yang berkompeten dan ahli pada setiap Program Studinya.

Program Studi pada SMK Informatika Citra Bangsa Ciamis antara lain :

- a. Teknik Komputer Jaringan.
- b. Multimedia
- c. Teknik Kendaraan Ringan

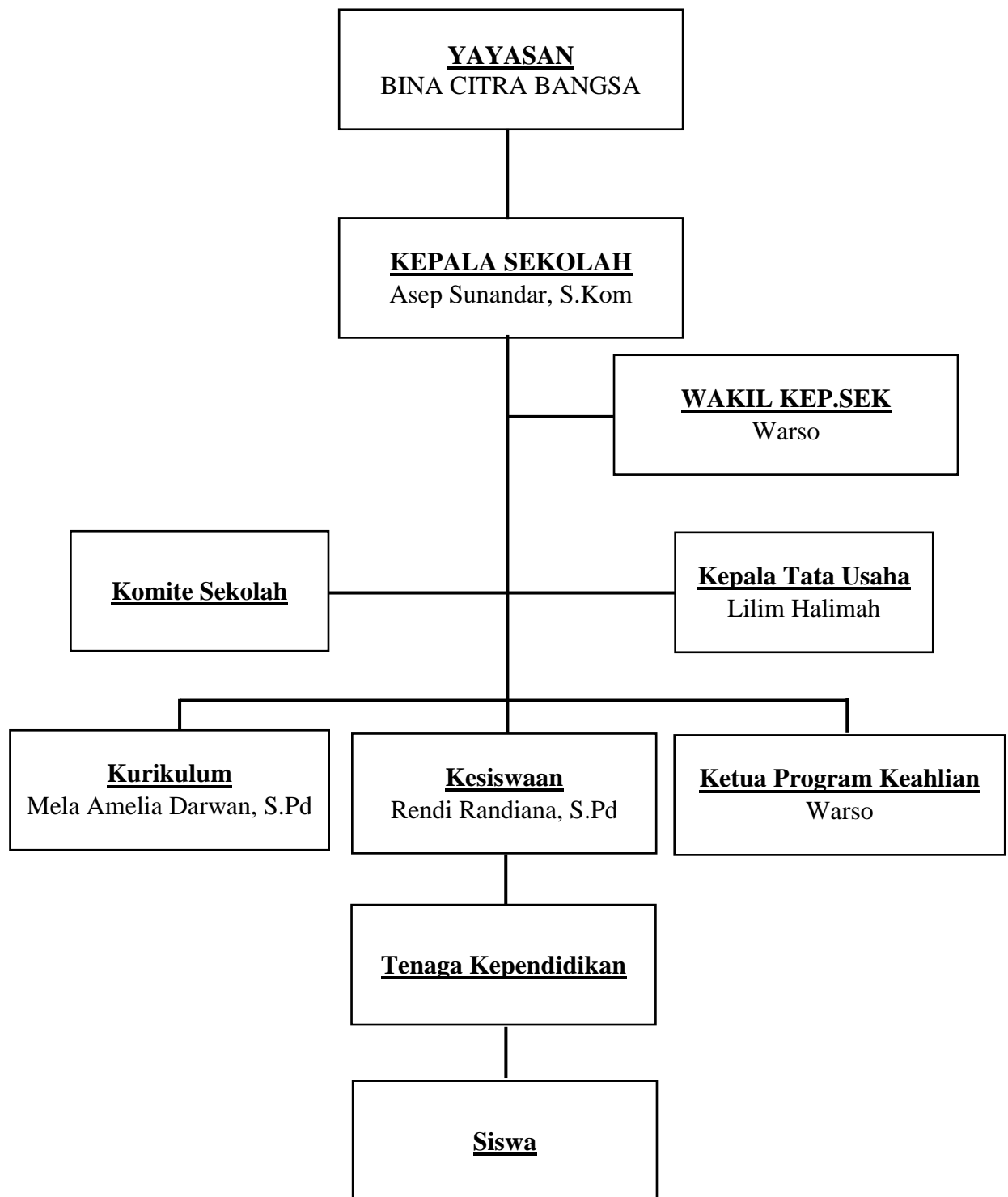
Adapun Visi dan Misi Sekolah Menengah Kejuuruan SMK Informatika Citra Bangsa adalah sebagai berikut:

a. Visi

- 1. menjadi sekolah pusat pendidikan dan latihan berbasis teknologi diwilayah Jawa Barat, untuk menciptakan sumber daya manusia profesional, beriman dan bertaqwa kepada Tuhan yang maha esa.
- 2. Menjadikan penyelenggara pendidikan berkualitas dan terpercaya di daerah Jawa Barat.

b. Misi

- 1. Mewujudkan tata kelola, sistem pengendalian manajemen, dan sistem.pengawasan internal yang modern, efektif, dan efesien
- 2. Menyalurkan dan Mendukung kreativitas peserta didik dengan sarana dan prasarana yang lengkap.
- 3. Mewujudkan budaya religi, jujur, disiplin, beretika, berestetika, pekerja keras,kreatif, inovatif, komptetitif, dan berkualitas.
- 4. Mewujudkan dinamisasi peningkatan kualitas pendidikan barkarakter yang berkesinambungan dan berkelanjutan.
- 5. Mewujudkan produk kompetensi keahlian bernilai Jual Pasar Global.
- 6. Memperluas akses kemitraan dunia kerja yang menjamin lapangan kerja dan prakerin bagi peserta didik dan lulusan SMK Informatika Citra Bangsa.
- 7. Mewujudkan lulusan yang handal di bidangnya sehingga dipercaya oleh segenap dunia kerja pemerintah maupun swasta.



Gambar 2. 4 Struktur Organisasi SMK Informatika Citra Bangsa Ciamis

2.3.2 Sistem Informasi Akademik

Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu, dilihat dari pendekatan sistem yang lebih menekankan pada elemen komponennya. Sedangkan Informasi didefinisikan sebagai data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. (Jogiyanto, 1990).

Wahana Komputer berpendapat mendefinisikan gabungan keduanya. Sistem Informasi adalah kumpulan organisasi yang dirancang untuk mengolah data menjadi informasi atau laporan yang ditujukan kepada pihak *internal* maupun *external*.

Sistem Informasi Akademik (SIA) merupakan sebuah aplikasi yang mengintegrasikan seluruh proses inti sebuah pendidikan ke dalam sebuah sistem informasi yang didukung oleh teknologi terkini. Dengan penerapan SIA akan mempengaruhi mutu layanan secara keseluruhan, yaitu layanan yang berhubungan dengan pihak-pihak diluar lembaga pendidikan (*Front Office*) dan tentu layanan yang berhubungan dengan internal lembaga pendidikan itu sendiri (*Back Office*) (Wahana Komputer, 2003).

Akademik adalah suatu bidang yang mempelajari tentang kurikulum dalam fungsinya untuk meningkatkan pengetahuan dalam segi pendidikan yang dapat dikelola oleh suatu sekolah (Risetyawan, 2010).