BAB 2
TINJAUAN PUSTAKA

2.1. Profil Sekolah

2.1.1. Sejarah Sekolah

SMK BSC merupakan sekolah kejuruan yang usia berdirinya masih relative muda yaitu tanggal 1 Januari 2009 bernaung dibawah Yayasan Teknologi Sarana Utama (YTSU). SMK BSC memiliki beberapa jurusan atau program keahlian yaitu :

1. Rekayasa Perangkat Lunak (RPL)

   Jurusan ini mengajarkan mengenai pemahaman perangkat lunak yang bertujuan agar siswa dapat dengan mudah menyesuaikan diri dengan dunia kerja. Materi yang dipelajari adalah semua sektor materi yang dibutuhkan pada industri – industri maupun perkantoran.
2. Teknik Komputer Jaringan (TKJ)

   Jurusan ini mengajarkan mengenai pemahaman tentang perangkat keras komputer , perakitan dan perbaikan komputer. Semua akan dipelajari di jurusan ini dengan harapan siswa dapat bersaing di dunia industri maupun berwirausaha.
3. Farmasi

   Indonesia sehat membuat kelulusan farmasi berperan utama dalam hal peracikan obat, perencanaan pembuatan obat, dll. Membuat kelulusan farmasi mempunyai peluang yang sangat besar untuk kedunia industry farmasi dan department kesehatan.
4. Otomotif

   Teknik atau langkah-langkah perbaikan dalam hal kendaran otomotif (mobil, motor) dll. Sehingga diharapkan siswa kelulusan Otomotif dapat bersaing serta merta dapat membuka lapangan pekerjaan.
5. Administrasi Perkantoran

   Siswa akan mempelajari teknik-teknik dalam hal administrasi, pembukuan, surat menyurat dll, sehingga siswa dapat menjadi bagian administrasi yang handal.

SMK BSC juga memiliki sarana dan prasarana demi menunjang aktifitas dalam kegiatan siswa seperti ruang kelas, aula, lapangan Olahraga, mushola, laboratorium, gudang, kantin dan beberapa bengkel sebagai tempat praktek yang digunakan pada beberapa jurusan.

2.1.2. Visi dan Misi

2.1.2.1. Visi

Menjadi Sekolah ungulan dengan basis teknologi dalam mengembangankan generasi muda Indonesia, khusunya di Jawa Barat agar menjadi manusia yang cerdas, berkualitas, ungul, dan berakhlak mulia dalam memasuki pasar kerja lokal, global, serta memiliki jiwa Enterpreneur untuk pengembangan usaha.

.

2.1.2.2. Misi

Menyiapkan lulusan yang terdepan dan mudah bekerja serta mempunyai prestasi dengan mendepankan perpaduan IMTAQ (Imam dan Taqwa) dan IPTEK (Ilmu Pengetahuan dan Teknologi)

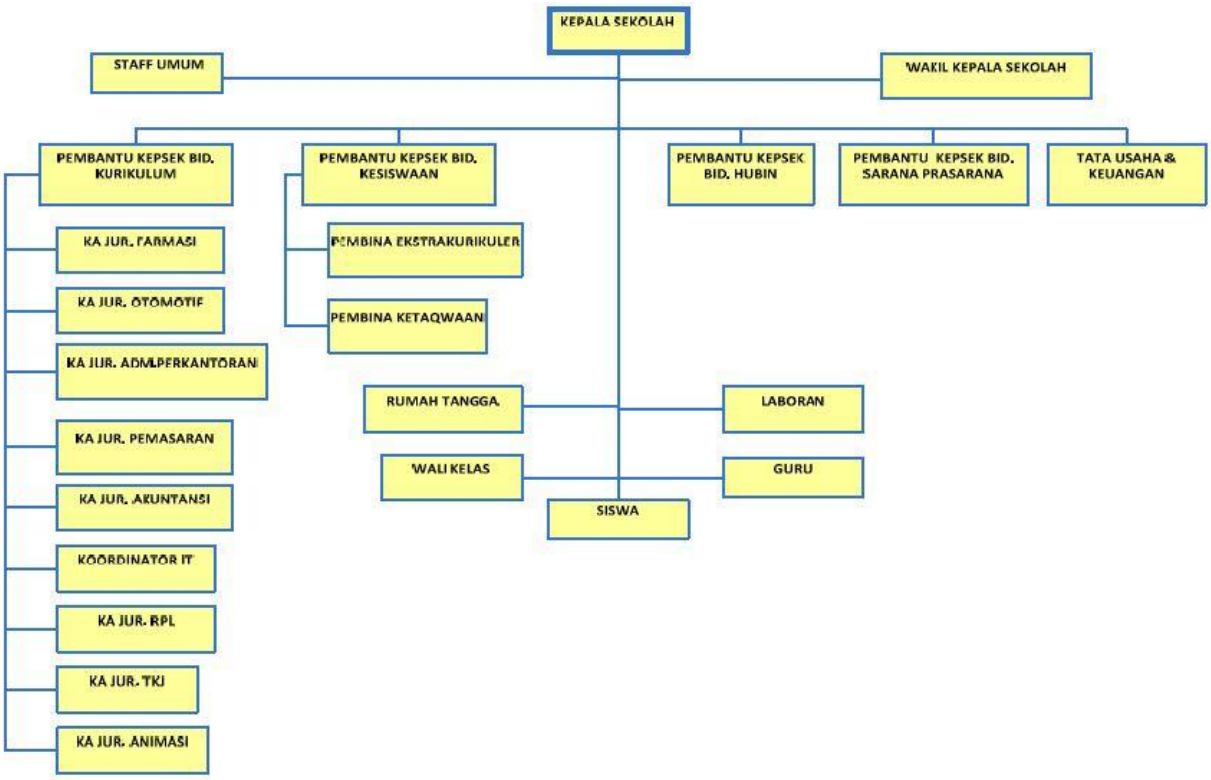
2.1.3. Logo SMK BSC



Gambar 2.1 Logo SMK Bina Sarana Cendika[1]

2.1.4. Struktur Organisasi

Struktur organisasi SMK BSC Bandung adalah sebagai berikut :



Gambar 2.2. Struktur Organisasi SMK BSC[1]

2.1.5. Deskripsi Tugas

Adapun Deskripsi tugas struktru organisasi pada SMK BSC adalah sebagai berikut :

1. Kepala Sekolah.
    a. Melakukan evaluasi terhadapkegiatan belajar mengajar.
    b. Penanggungjawab segala kegiatan sekolah kepada Komite sekolah dan Dinas Pendidikan.
    c. Penanggungjawab manajemen sekolah.
    d. Melaksanakan pengawasan terhadap kegiatan akademik sekolah.
    e. Menetukan kebijakan sekolah.

Membawahi dalam pengambilan segala keputusan yang berhubungan dengan kegiatan sekolah.

2. Wakil Kepala Sekolah.
    a. Membantu kepala sekolah dalam melaksankan program atau tugas – tugasnya.

3. Pembantu Kepala Sekolah Bidang Kurikulum.
    a. Menyusun jadwal pengajaran.
    b. Mengaudit buku sumber dicocokan dengan pengembangan kurikulum.
    c. Pengurusan kegiatan belajar baik kurikuler maupun estrakulikuler.
    d. Mengontrol target kurikulum, kompetisi dan daya serap siswa.
    e. Mengembangkan kurikulum dan mutu pendidikan.
    f. Mempertangung jawabkan pencapaian hasil dan mutu pendidikan.
4. Pembantu Kepala Sekolah Bidang Kesiswaan.
    a. Pembina organisasi siswa intra sekolah (OSIS).
    b. Pembina terhadap prestasi siswa.
    c. Pengarahan dan pngendalian siswa dalam rangka mengekan disiplin siswa dan tata tertib sekolah.
5. Pembantu Kepala Sekolah Bidang Hubin.
    a. Membantu kepala sekolah di bidang hubungan masyarakat.
    b. Menjalin komunikasi dengan dinas, dunia kerja dan industry.
6. Pembantu Kepala Sekolah Bidang Sarana dan Prasarana.
    a. Membantu kepala sekolah dalam memimpin , mengawasi seluruh kegiatan ketatalaksanan & ketatausahaan sekolah.
    b. Memelihara dan mengawasi penggunaan sarana dan prasarana sekolah.
    c. Memelihara dan mencatat laporan tentang semua hal ketaatalaksanaan dan tatausaha.
7. Bidang Tata Usaha dan Keuangan
    a. Penyususnan program tatausaha sekolah.
    b. Pengurusan adminstrasi pegawai, guru dan siswa.
    c. Pembinaan & pengembangan karier pegawai usaha.
    d. Penyusunan laporan pelaksanaan kegiatan pengurusan ketatausahaan secara berkala.
    e. Penyususnan dan penyajian data / statistic sekolah.
8. Kepala Jurusan.
    a. Bertanggung jawab berjalannya kegiatan akademik di masing – masing setiap jurusan.

9. Guru
    a. Bertanggung jawab kepada kepala sekolah dan mempunyai tugas melaksanakan proses belajar mengajar secara efektif dan efisien.
10. Wali Kelas
    a. Mengkoordinasi kegiatan dikelas yang di asuhnya.
    b. Mengadakan kunjungan/ silaturahmi kepada orang tua / wali siswa.
    c. Memanggil orang tua / wali siswa yang bermasalah.
    d. Melaksanakan adminstrasi kelas.
    e. Mengatasi siswa yang bermasalah bersama guru BP dan GDN.
    f. Membina siswa yang menjadi bimbingannya secara kontinyu.
    g. Membantu kelancaran penagihan BP-3 dan administrasi sekolah lainnya.
    h. Mengajukan usulan dan saran kepada kepala sekolah untuk pembinaan dan pengembangan serta memupuk bakat dan minat siswa baik secara individu maupun secara kelompok.
    i. Mengusulkan beasiswa kepada Kaprogram keahlian bagi siswa yang memeuhi syarat yang ditentukan.
    j. Tugas lain yang relevan.
11. Laboran
    a. Bertugas mengelola, menyiapkan fasilitas yang ada dilaboratorium pratikum siswa.

2.2. Landasan Teori

Landasan teori dari penulisan laporan kerja praktek ini menguraikan proses pembangunan sistem informasi Penerimaan Siswa Baru di SMK BSC Berbasis Web.

2.2.1. Pengertian Sistem

Terdapat beberapa pendapat yang menjelaskan mengenai definisi dari pengembangan sistem, diantaranya :

1. Menurut Jogiyanto [2], sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata.

Kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi.

2. Menurut Budi Sutejo [3], sistem merupakan kumpulan elemen yangsaling berhubungan satu sama lain yang membetuntuk satu kesatuan dalam usaha mencapai suatu tujuan.
3. Menurut Tata Sutabri [4], sistem adalah suatu kumpulan atau himpunan dari unsur, komponen atau variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu.

Dari uraian diatas dapat disimpulkan bahwa sistem adalah satu kesatuan yang terdiri dari dua atau lebih komponen atau subsistem yang berinteraksi untuk mencapai suatu tujuan

2.2.2. Elemen – elemen Sistem

Menurut Jogiyanto[2] ,Elemen sistem adalah suatu sistem terdiri dari sejumlah elemen yang saling berinteraki yang artinya saling kerjasama membentuk satu kesatuan. Pendekatan suatu sistem yang merupakan suatu jaringan prosedur lebih menekankan pada urutan – urutan operasi di dalam sistem, sedangkan pendekatan yang menekan kan pada elemn – elemen atau komponen merupakan interaksi antar elemen – elemen atau endapai sasaran atau tujuan sistem.

2.2.3. Karateristik Sistem

Menurut Jogiyanto[2], Suatu sistem mempunyai karakteristik atau sifat – sifat tertentu yaitu mempunyai komponen – komponen (*componen*), batasan sistem (*boundary*), lingkup luar sistem (*enviroments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolahan(*processo*) dan sasaran (*objective*) atau tujuan (*goal*) :

1. Komponen Sistem (*components*)

   Suatu sistem terdiri dari kumpulan komponen atau subsitem yang saling berinteraksi atau bekerja satu sama lain untuk membentuk suatu kesatuan. Setiap komponen atau subsistem tersebut memiliki sifat – sifat dan fungsi tertentu yang mempengaruhi proses sistem secara keseluruhan.
2. Batasan Sistem (*boundary*)

Batasan sistem yaitu daerah yang memebatasi antara suatu sistem dengan lingkungan luarnya atau dengan sistem yang lainnya. Batas suatu sistem merupakan ruang lingkup (*scope*) dari suatu sistem tersebut. Dan batas sistem ini memungkinkan suatu sistem di pandalang dalam sebagi satu kesatuan.

3. Lingkup Luar Sistem (*Environments*)

   Enviroments dari suatu sistem adalah apapun yang berada diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat menguntungkan dan dpaat juga bersifat merugikan sitem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem yang harus dijaga dan dipelihara. Sedangkan luar sistem yang bersifat merugikan harus ditahan dan dikendalikan, kalau tidak maka akan menggangu kelangsungan hidup dari sistem.

4. Penghubung Sistem (*Interface*)

   Merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Keluaran (output) dari satu subsistem akan menjadi masukan (input) untuk subsistem yang lain dengan melalui penghubung. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

5. Masukan (I*nput*)

   Merupakan energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (maintenance input) dan masukan sinyal (signal input). Maintenance input adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Signal input adalah energi yang diproses untuk didapatkan keluaran.

   Sebagai contoh di dalam sistem komputer, program adalah maintaenance input yang digunakan untuk mengoperasikan komputer dan data adalah signal input untuk diolah menjadi informasi.

6. Keluaran (*Output*)

   Keluaran merupakan hasil dari energi yang diolah. Keluaran ini dapat berupa keluaran yang berguna atau keluaran yang tidak berguna. Sebagai contoh panas yang dikeluarkan dari sistem komputer merupakan keluaran yang tidak berguna

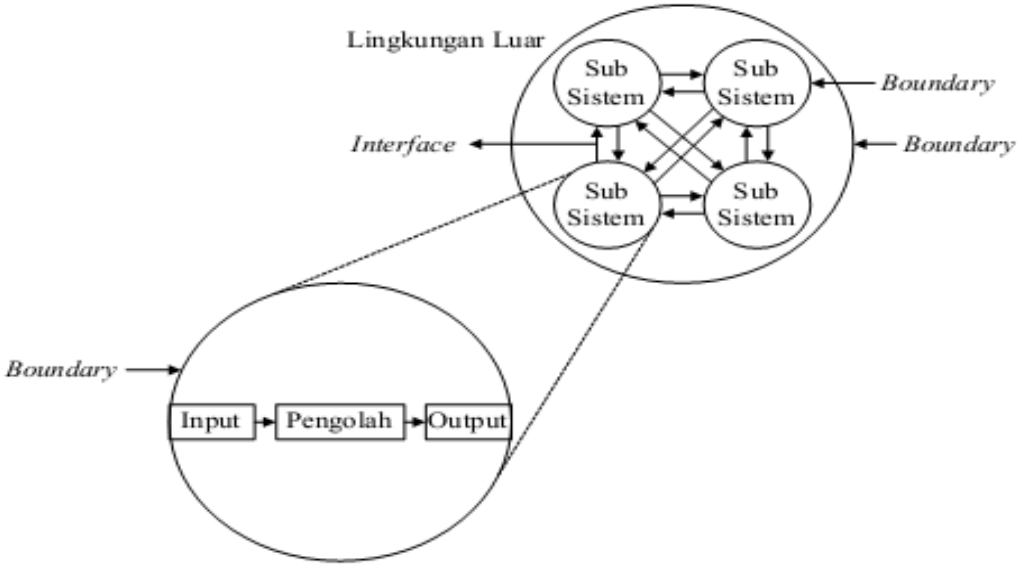
dan merupakan sisa pembuangan. sedangkan informasi adalah keluaran yang berguna dan dibutuhkan.

7. Proses (*Process*)

   Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran. Sistem akuntansi akan mengolah data-data transaksi menjadi laporan-laporan keuangan dan laporanlaporan yang dibutuhkan oleh manajemen.

8. Sasan Sistem (*Objective*)

   Suatu sistem pasti mempunyai tujuan (goal) atau sasaran (objective). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.



Gambar 2.3 Karakteristik Sistem

2.2.4. Kualitas Sistem

Menurut Roger S.Pressman [5], “Kualitas sistem, aplikasi atau produk merupakan persyaratan yang menjelaskan masalah, desain model solusi, kode yang membuat program dapat dieksekusi dan pengujian yang menguji perangkat lunak untuk menemukan kesalahan”.

Adapun indikator yang menentukan kualitas suatu sistem informasi atau kualitas suatu perangkat lunak menurut Roger S. Pressman[5] disingkat FURPS (*Functionality*, *Usability*, *Performance*,*Reliability* dan *Supportability*), namun dalam penelitian ini hanya 4 (empat) yangdijadikan indikator untuk menentukan kualitas suatu perangkat lunak yaitusebagai berikut :

1. Bekerja sesuai fungsinya (*Functionality*) Dinilai melalui evaluasi bentuk himpunan dan kemampuan program, generalitas fungsi-fungsi yang disampaikan, dan keamanan keseluruhan sistem.
2. Kemampuan (*Usability*) Dinilai dengan mempertimbangkan factor manusia, keseluruhan estetika, konsistensi, dan dokumentasi.
3. Kinerja Sistem (*Performance*) Diukur melalui kecepatan pemrosesan, waktu respon, konsumsi kode sumber, *throughput* dan efesiensi.

Keandalan (*Reliability*) Dievaluasi melalui pengukuran frekuensi dan besarnya kegagalan, akurasi hasil output, *Mean Time Between Failure* (MTBF), kemampuan untuk pulih dari kegagalan, dan prediktabilitas program.

2.2.5. Pengertian Informasi

Informasi ibarat darah yang mengalir di dalam tubuh suatu organisasi, sehingga informasi ini sangat penting di dalam suatu organisasi. Informasi (*information*) dapat didefinisikan sebagai berikut:

1. Informasi merupakan data yang telah diproses, transformasi data, sehingga berubah bentuk menjadi informasi. Informasi adalah data yang telah diambil kembali, diolah atau sebaliknya digunakan untuk tujuan *informative*, argumentasi ataupun sebagai dasar untuk pengambilan keputusan. Informasi adalah hasil proses data yang bentuknya kurang berguna menjadi data yang berguna. Menurut Jogiyanto Hartono [2].
2. Informasi merupakan data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunkan data tersebut. Abdul Kadir [6].
3. Informasi merupakan hasil pemrosesan data yang diperoleh dari setiap elemen sistem tersebut menjadi bentuk yang mudah dipahami dan merupakan

pengetahuan yang relevan yang dibutukkan oleh orang untuk menambah pemahamannya terhadap fakta-fakta yang ada. Budi Sutejo [3].

2.2.6. Siklus Informasi

Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model untuk dihasilkan informasi. Data yang diolah untuk menghasilkan informasi menggunakan suatu model proses tertentu.

Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lainnya yang akan membuat sejumlah data kembali. Data tersebut akan ditanggap sebagai input, diproses kembali lewat suatu model dan membentuk suatu siklus. Menurut John Burch disebut dengan siklus informasi (*information cycle*) atau siklus ini disebut juga dengan siklus pengolahan data (*data processing cycles*).

2.2.7. Kualitas Informasi

Kualitas Informasi digunakan untuk mengambil keputusan serta tindakan sehingga informasi yang diterima akurat dan tepat waktu serta relevan, yaitu :

1. Setiap informasi yang dipakai dalam proses pengambilan keputusan,informasi tersebut harus memiliki kualitas informasi. Menurut Abdul Kadir [6], kualitas informasi tersebut adalah sebagai berikut :
    a. Akurat *(Accuracy)* berarti informasi harus bebas dari kesalahan-kesalahan dan tidak biasa atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai berubah atau merusak informasi tersebut
    b. Tepat pada waktu *(TimeLinnes),* berarti informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan didalam pengambilan keputusan. Bila pengambilan keputusan terlambat, maka dapat berakibat fatal untuk organisasi.

c. Relevan *(Relevancy),* berarti informasi tersebut mempunyai manfaat untuk pemakainya, relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda.

2. Menurut Jogiyanto [2], Kualitas dari suatu informasi (*Quality of Information*) tergantung dari tiga hal, yaitu informasi harus akurat (*Accurate*), tepat pada waktunya (*Timeliness*), dan relevan (*Relevance*).

Dari definisi kualitas informasi diatas maka peneliti mengambil kesimpulan bahwa kualitas informasi dapat dilihat berdasarkan tiga hal, yaitu informasi harus akurat (*Accurate*), tepat waktu (*Timeliness*), dan relevan (*Relevance*).

2.2.8. Nilai Informasi

Menurut Jogiyanto[2], Nilai dari informasi (*Value of Information*) ditentukan dari dua hal, yaitu manfaat dan biaya mendapatkannya.

Berdasarkan definisi nilai informasi diatas, maka penyusun menyimpulkan bahwa nilai informasi ditentukan oleh dua hal, yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.

2.2.9. Sistem Informasi

Terdapat berbagai macam pengertian Sistem Informasi menurut beberapa ahli, diantaranya :

1. Sistem Informasi adalah suatu sistem didalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung oprasi, bersifat manajerial dan kegiatan strategis dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. Jogiyanto [2].
2. Sistem Informasi adalah kumpulan elemen yang saling berhubungan satu sama lain untuk membentuk suatu kesatuan untuk mengintegrasi data, memproses dan menyimpan serta mendistribusikan informsi tersebut. Budi Sutejo [3]

3. Sistem informasi adalah suatu sistem buatan manusia yang secara umum terdiri atas sekumpulan komponen berbasis komputer dan manual yang dibuat untuk menghimpun, menyimpan, dan mengolah data serta menyediakan informasi keluaran kepada pemakai. Abdul Kadir [6]

Jadi defenisi sistem informasi adalah suatu kumpulan sumber daya manusia atau alat yang terpadu serta modal yang bertanggung jawab untuk mengumpulkan data dan mengolah data demi menghasilkan suatu informasi yang berguna bagi seluruh tingkat operasi untuk kegiatan perencanaan, pelaksanaan, pekerjaan, pengendalian, dan pengambilan keputusan dalam sebuah organisasi.

2.2.10. Komponen Sitem Informasi

Sistem informasi tidak akan berjalan atau bekerja apabila tidak ada sesuatu yang menggerakkannya. Oleh karena itu diperlihatkan beberapa komponen untuk menggerakkan sistem informasi tersebut.

1. Perangkat Keras

   Perangkat keras adalah komputer yang berperan sebagai alat input, dan alat output.
2. Perangkat lunak

   Perangkat lunak merupakan alat yang dipakai untuk menjalankan perangkat keras. Tanpa adanya perangkat lunak maka perangkat keras yang telah ada tidak akan dapat digunakan. Perangkat lunak ini dapat berupa sistem operasi maupun program aplikasi yang berhubungan dengan sistem informasi.
3. Data

   Data merupakan bagian yang penting dari suatu sitem informasi karena data adalah bahan baku untuk menghasilkan informasi.
4. Prosedur

   Prosedur adalah suatu urutan dari suatu pekerjaan agar setiap pekerjaan dapat diselesaikan sesuai dengan rencana yang telah ditetapkan sebelumnya.
5. Personal

Personal dalam hal ini adalah manusia yang merupakan bagian yang terpenting karena hanya manusia yang dapat menangani semua komponen yang telah disebutkan di atas.

2.2.11. Flowmap

Flowmap merupakan diagram alir yang menunjukan aliran suatu dokumen, aliran data fisik entitas sistem informasi dan kegiatan operasi yang berhubungan dengan sistem informasi. Penggambaran biasanya diawali dengan mengamati dokumen apa yang menjadi media data atau informasi dan selanjutnya ditelusuri bagaimana dokumen termasuk ke bagian entitas mana dokumen tersebut, proses apa yang terjadi terhadap dokumen tersebut dan seterusnya.

2.2.12. Basis Data

Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan [1]. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.

Sistem informasi tidak dapat dipisahkan dengan kebutuhan akan basis data apa pun bentuknya, entah berupa *file* teks ataupun *Database Management System* (DMBS). Kebutuhan basis data dalam sistem informasi meliputi :

1. Masukkan, menyimpan, dan mengambil data.
2. Membuat laporan berdasarkan data yang telah disimpan.

2.2.13. DBMS

DBMS (*Database Management System*) atau dalam bahasa Indonesia sering disebut sebagai Sistem Manajemen Basis Data adalah suatu sistem aplikasi yang digunakan untuk menyimpan, mengelola, dan menampilkan data [1]. Suatu sistem aplikasi disebut DBMS jika memenuhi persyaratan minimal sebagai berikut :

1. Menyediakan fasilitas untuk mengelola akses data
2. Mampu menangani integritas data
3. Mampu menangani akses data yang dilakukan secara bersamaan
4. Mampu menangani *backup* data

Karena pentingnya data bagi suatu organisasi/perusahaan, maka hampir sebagian besar perusahaan memanfaatkan DBMS dalam mengelola data yang mereka miliki. Pengelolaan DBMS sendiri biasanya ditangani oleh tenaga ahli yang spesialis menangani DBMS yang disebut sebagai DBA (*Database Administrator*).

DBMS sudah mulai berkembang sejak tahun 1960an. Kemudian sekitar tahun 1970an mulai berkembang teknologi *Relational DBMS* yaitu DBMS berbasis relasional model. Relasional model pertama kali dikembangkan oleh Edgar J. Codd pada tahun 1970. Secara sederhana reasional model dapat dipahami sebagai suatu model yang memandang data sebagai sekumpulan tabel yang saling terkait. Hampir semua DBMS komersial dan *open source* saat ini berbasis *Relational* DMBS atau RDBMS.

Pada tahun 1980-an mulai berkembang *Object Oriented* DBMS (OODBMS). OODBMS berkembang seiring dengan perkembangan teknologi pemrograman berorientasi objek. Secara umum dapat diartikan bahwa OODBMS merupakan DBMS yang memandang data sebagai suatu objek. Saat ini OODBMS juga cukup berkembang namun belum dapat menggeser kepopuleran RDBMS.

Berikut ini adalah 4 macam DBMS versi komersial yang paling banyak digunakan di dunia saat ini, yaitu :

1. Oracle
2. Microsoft SQL Server
3. IBM DB2
4. Microsoft Access

Sedangkan DBMS versi *open source* yang cukup berkembang dan paling banyak digunakan saat ini adalah sebagai berikut:

1. MySQL
2. PostgreSQL
3. Firebird
4. SQLite

Hampir semua DBMS mengadopsi SQL sebagai bahasa untuk mengelola data pada DBMS.

2.2.14. UML (Unified Modeling Language)

UML (Unified Modeling Language) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, danpendokumentasian dari sebuah sistempengembangan software berbasis OO (*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelaskelas dalam bahasa program yang spesifik, skema database, dan komponenkomponen yang diperlukan dalam sistem software.[8]

2.2.14.1 Artifact UML

UML menyediakan beberapa notasi dan *artifact* standar yang bisa digunakan sebagai alat komunikasi bagi para pelaku dalam proses analisis dan desain. *Artifact* didalam UML didefinisikan sebagai informasi dalam bentuk yang digunakan atau dihasilkan dalam proses pengembangan perangkat.

Contohnya adalah *source code* yang dihasilkan oleh proses pemrograman. Yang harus diperhatikan untuk menjaga konsistensi antar *artifact* selama proses analisis dan desain adalah bahwa setiap perubahan yang terjadi pada satu *artifact* harus juga dilakukan pada *atifact* sebelumnya.

Untuk membuat suatu model, UML memiliki diagram grafis sebagai berikut :

1. *Use Case Diagram*
2. *Class Diagram*
3. *Statechart Diagram*
4. *Activity Diagram*
5. *Sequence Diagram*
6. *Collaboratorion Diagram*
7. *Component Diagran*
8. *Deployment Diagram*

Dibuatnya berbagai jenis diagram dikarenakan :

1. Setiap sistem yang kompleks selalu paling baik jika didekati melalui himpunan berbagai sudut pandang yang kecil yang satu sama lain hampir

saling bebas (*independent*). Sudut pandang tunggal senantiasa tidak mencukupi untuk melihat sistem yang besar dan kompleks.

2. Diagram yang berbeda – beda tersebut dapat menyatakan tingkatan yang berbeda-beda dalam proses rekayasa.
3. Diagram-diagram tersebut dibuat agar model yang dibuat semakin mendekati realitas.

2.2.14.2 Tujuan UML

Tujuan UML diantaranya untuk :

1. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
2. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
3. Menyatukan praktek-praktek terbaik yang terdapat dalam bahasa pemodelan.

2.2.14.3 Notasi dalam UML

1. Actor

Actor menggambarkan segala pengguna *software* aplikasi (*user*). *Actor* memberikan suatu gambaran jelas tentang apa yang harus dikerjakan *software* aplikasi. Sebagai contoh sebuah *actor* dapat memberikan *input* ke dalam dan menerima informasi dari *software* aplikasi, perlu dicatat bahwa sebuah *actor* berinteraksi dengan *use case*, tetapi tidak memiliki kontrol atas *use case*. Sebuah *actor* mungkin seorang manusia, satu *device*, *hardware* atau sistem informasi lainnya.

2. Use Case

Use case menjelaskan urutan kegiatan yang dilakukan *actor* dan sistem untuk mencapai suatu tujuan tertentu. Walaupun menjelaskan kegiatan, namun *use case* hanya menjelaskan apa yang dilakukan oleh *actor* dan sistem bukan bagaimana *actor* dan sistem melakukan kegiatan tersebut.

a. *Use-case Konkret* adalah *use case* yang dibuat langsung karena keperluan *actor*. *Actor* dapat melihat dan berinisiatif terhadapnya.
b. *Use-case Abstrak* adalah *use case* yang tidak pernah berdiri sendiri. *Use case abstrak* senantiasa termasuk didalam (*include*), diperluas dari (*extend*) atau memperumum (*generalize*) *use case* lainnya.

Untuk menggambarkannya dalam *use case* model biasanya digunakan *association relationship* yang memiliki *stereotype include*, *extend* atau *generalization relationship*. Hubungan *include* menggambarkan bahwa suatu *use case* seluruhnya meliputi fungsionalitas dari *use case* lainnya. Hubungan *extend* antar *use case* berarti bahwa satu *use case* merupakan tambahan fungsionalitas dari *use case* yang lain jika kondisi atau syarat tertentu terpenuhi.

2.2.14.4 Class

Class merupakan pembentuk utama dari sistem berorientasi *obyek*, karena *class* menunjukkan kumpulan *obyek* yang memiliki atribut dan operasi yang sama. *Class* digunakan untuk mengimplementasikan *interface.*

Class digunakan untuk mengabstraksikan elemen-elemen dari sistem yang sedang dibangun. *Class* bisa merepresentasikan baik perangkat lunak maupun perangkat keras, baik konsep maupun benda nyata. Notasi *class* berbentuk persegi panjang berisi 3 bagian: persegi panjang paling atas untuk nama *class*, persegi panjang paling bawah untuk operasi, dan persegi panjang ditengah untuk *atribut*.

Atribut digunakan untuk menyimpan informasi. Nama *atribut* menggunakan kata benda yang bisa dengan jelas merepresentasikan informasi yang tersimpan didalamnya. Operasi menunjukkan sesuatu yang bisa dilakukan oleh *obyek* dan menggunakan kata kerja.

2.2.14.5 Interface

Interface merupakan kumpulan operasi tanpa implementasi dari suatu *class*. Implementasi operasi dalam *interface* dijabarkan oleh operasi di dalam

class. Oleh karena itu keberadaan *interface* selalu disertai oleh *class* yang mengimplementasikan operasinya. *Interface* ini merupakan salah satu cara mewujudkan prinsip *enkapsulasi* dalam *obyek.*

2.2.14.6 Interaction

Interaction digunakan untuk menunjukkan baik aliran pesan atau informasi antar *obyek* maupun hubungan antar *obyek*. Biasanya *interaction* ini dilengkapi juga dengan teks bernama *operation signature* yang tersusun dari nama operasi, parameter yang dikirim dan tipe parameter yang dikembalikan.

2.2.14.7 Note

Note digunakan untuk memberikan keterangan atau komentar tambahan dari suatu elemen sehingga bisa langsung terlampir dalam model. *Note* ini bisa disertakan ke semua elemen *notasi* yang lain.

2.2.14.8 Dependency

Dependency merupakan *relasi* yang menunjukan bahwa perubahan pada salah satu elemen memberi pengaruh pada elemen lain. Elemen yang ada di bagian tanda panah adalah elemen yang tergantung pada elemen yang ada dibagian tanpa tanda panah.

Terdapat 2 *stereo type* dari *dependency*, yaitu *include* dan *extend*. *Include* menunjukkan bahwa suatu bagian dari elemen (yang ada digaris tanpa panah) memicu eksekusi bagian dari elemen lain (yang ada di garis dengan panah). *Extend* menunjukkan bahwa suatu bagian dari elemen di garis tanpa panah bisa disisipkan kedalam elemen yang ada di garis dengan panah.

2.2.14.9 Association

Association menggambarkan navigasi antar *class* (*navigation*), berapa banyak *obyek* lain yang bisa berhubungan dengan satu *obyek* (*multiplicity* antar *class*) dan apakah suatu *class* menjadi bagian dari *class* lainnya (*aggregation*).

Navigation dilambangkan dengan penambahan tanda panah di akhir garis. *Bidirectional navigation* menunjukkan bahwa dengan mengetahui salah satu *class* bisa didapatkan informasi dari *class* lainnya. Sementara *Uni Directional navigation* hanya dengan mengetahui *class* diujung garis *association* tanpa panah.

Aggregation mengacu pada hubungan "*has-a*", yaitu bahwa suatu *class* memiliki *class* lain, misalnya Rumah memiliki *class* Kamar.

2.2.14.10 Generalization

Generalization menunjukkan hubungan antara elemen yang lebih umum ke elemen yang lebih spesifik. Dengan *generalization*, *class* yang lebih spesifik (*subclass*) akan menurunkan *atribut* dan operasi dari *class* yang lebih umum (*superclass*) atau "*subclass is superclass*". Dengan menggunakan *notasi generalization* ini, konsep *inheritance* dari prinsip *hirarki* dapat dimodelkan.

2.2.14.11 Realization

Realization menunjukkan hubungan bahwa elemen yang ada di bagian tanpa panah akan merealisasikan apa yang dinyatakan oleh elemen yang ada di bagian dengan panah. Misalnya *class* merealisasikan *package*, *component* merealisasikan *class* atau *interface*

2.2.15. HML dan PHP

2.2.15.1.HTML

HTML (H*ypetext Markup Language*) adalah sebuah bahsa makup yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi di dalam sebuah pejajah web Internet dan formating hypertext sederhana yag ditulis kedalam berkas format ASCII agar dapat mengahsilkan tampilan wujud yang terintegrasi. Dengan kata lain, berkas yang di buat dalam perangkat lunak pengolahan kata dan disimpan kedalam format ASCII normal sehingga menjadi home page dengan perintah – perintah HTML. Bermula dari sebuah bahasa yang sebelumnya banayak digunakan di dunia penerbitan dan percetakan yang disebut

dengan SGML (*Standard Generalized Markup Language*), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saat ini merupkan standar Internet yang didefinisikan dan dikendalikan penggunanaya oleh World Wide Web Consortium (W3C).

HTML berupa kode – kode tag yang menginstrusikan penjelajah web untuk menghasilkan tampilan sesuai dengan yang diinginkan. Sebuah file yang merupkan file HTML dapat dibuka dengan menggunakan penjajahan web seperti Mozila Firefox atau Microsoft Internet Expoler. HTNL juga dapat dikenal oleh aplikasi pembuka surel ataupun dari PDA dan perangkat lunak lain yang memiliki kemampuan browser. Dengan menggunakan perintah – perintah HTML memungkinkan pengguna untuk melakukan tugas- tugas berikut:

1. Menentukan ukuran data dan alur tulisan
2. Mengintegrasi gambar dengan tulisan
3. Membuat pranala
4. Mengintegrasi berkas suara dan rekaman gambar hidup

Kode HTML diawali dengan huruf <html> dan diakhiri denga </html?. Namun, sebelum <html> diaharuskan untuk diberikan deklarasi !DOCTYPE. Fungsi deklarasi ini adalah sebagai *Public Text Identifer*. Beberapa hal penting dalam kode HTML :

1. Tanda <> menyatakan sebuah tag.
2. Pada umumnya tag berpasangan
3. Tag yang tidak berpasangan antara lain adalah <br/> dan <hr/>
4. Pada tag yang berpasangan, tag yang berkedudukan dan sebagai tag penutup mempunyai bentuk berupa</>.

2.2.15.2.PHP

PHP (*Hyperteks Processor)*, merupakan bahasa pemograman web bersifat serverside. Hal tersebut mengandung arti bahswa PHP merupakan bahasa berbentuk *script* yang disimpan dan fdijalankan dikomputer server () sedangkan hasilnya dikirimkan ke komputer client (*web browsing*) dalam bentuk *script*HTML. Keuntungan penggunaan PHP, kode yang menyusun program tidak perlu dibagikan ke pemakai, yang berarti bahwakerahasiaan kode dapat dilindungi.

Pengunaan PHP memungkinkan web dapat dibuat dinamis sehingga perawatan situs web tersebut menjadi lebih mudah dan efisien. Hal menarik ini yang didukung oleh PHP ada kenyataan bahwa PHP bisa digunakan untuk mengakses berbagi macam database seperti Access, Oracle, MySQL dan lain – lain. PHP merupakan perangkat lunak Open Source yang disebarkan dan dilisesnsikan secara gratis serta dapat diunduh secara bebas dari situs resminya.

Fungsi yang dimiliki PHP sanggat lengkap sehingga dapat dikatakan tidak perlumembuat fungsi sendiri. Hal tersebut dikarenakan daftar fungsi PHP yang lengkap menjadikan baris perintah semakin efisien. Selain itu juga, seperti yang telah disebutkan sebelumya bahwa kelebihan PHP adalah mampu berintegrasi dengan berbagai macam databases. Salah satu dabases yan paling umum diintegrasikan adalah MySQL.

Adapun karateristik *script* PHP dapat diuraikan sebagai berikut:

1. File PHP disimpan dengan eketensi file seperti *.php3, *.php4, *.php.
2. Script PHP biasanya diawali dengan *tag* <? atau <?php dan ditutup dengan *tag* ?>.
3. File PHP dapat menginduk atau disisispkan ada bahasa *script* lainya atau dapat berdiri sendiri.

2.2.16. Apche dan MySQL

2.2.16.1.Apache Web Server

Apache merupakan *web server* yang paling banyak digunakan di *internet*. Sever HTTP Apache atau server web/www apache adalah sever web yang dapat dijalankan dibanyak sistem operasi (Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta platform lainnya) yang berguna melayani dan memfungsikan situs web. Protokol yang digunakan untuk melayani fasilitas web/www ini menggunakan HTTP.

Apache memiliki program pendukung yang cukup banyak. Hal ini memeberikan layanann yang cukup lengkap bagi penggunanaya. Beberapa dukungan tersebut diantaranya:

1. Kontrol Aksses

   Kontrol ini dapat dijalankan berdasarkan nama *host* atau nomor IP

2. CGI (*Common Gateway Interface*)

   CGI yang paling terkenal adalah Perl (*Pratical Extraction and Report Language*), didukung oleh Apache dengan menempatkannya sebagi odul (mod-perl).

3. PHP (*Personal Homepage / Hypertext Processor*)

   Apache mendukung PHP dengan menempatkannya sebagai salah satu modulnya (mod-php). Hal ini menjadikan kinerja PHP lebih baik.

4. SSI (*Server Side Includes*)

   Selain itu kelebhan lainnya dari Apache diantaranya yaitu Apache termasuk dalam katagori *freewar* dan Apache mudah dalam proses instalasi dibandingkan dengan web browser lainya.

2.2.16.2.MySQL

MySQL sebenrnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya yaitu SQL (Structured Query Language). SQL adalah konsep pengoperasian basisdata, terutama untuk pemilihan atau selekasi data pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Kehandalan suatu sistem basisdata (DBMS) dapat diketahui dari cara kerja pengoptimasi-nya dalam melakukan proses perintah – perintah SQL yang dibuat oleh pengguna maupun program – program aplikasi yang memanfaatkanya. Sebgai peladen basis data, MySQL mendukung operasi basisdata transakional maupun operasi basisdata non-transaksional.

Sebagi databases server yang memiliki konsep databases modern, MySQL memiliki banyak sekeali keistimewaan. Berikut ini beberapa keistimewaan yang dimiliki oleh MySQL:

a. Probility

   MySQL dapat berjalan stabil pada berbagai OS seperti Windows, Linux, Unix, Mac OS, Solaris, Unix, Amiga, HP-UX, Symbian.

b. Open Source “Limited”

   Dahulu MySQL didistribuskan secara open source (gratis), dibawah lisensi GPL sehingga kita dapat menggunakanya secara Cuma – Cuma tanpa dipungut biay. Namun, saat ini karena MySQL telah dibeli oleh SUN, maka kita tidak

dapat lagi menikmati fitur – fitur baru ini hanya user yang membeli lisensinya. Sehinnga MySQL tidak lagi sebuah opensource yang benar – benar gratis lagi. MySQL sekarang hanya menyediakan fitur – fitur "dasar" saja yang saat ini menggunakan versi 5.1. untuk ,emdownloadnya silahkan download disini dan dicari versi MySQL dengan OS kita.

c. Multiuser

MySQL dapat digunakan oleh beberapa user dalam waktu bersamaan tanpa mengalami konflik. Hal ini memungkinkan sebuah database server MySQL dapat diakses klien secara bersamaan.

d. Perfomance Tuning

MySQL memiliki kecepatan yang menajubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL persatuan waktu.

e. Column Types

MySQL memiliki tipe kolom yang sangat komplek, seperti signed/ unsiged interger, float, double, char, varchar, text, blob, date, time, datetime, timestamp, year, set serta enum.

f. Command and Functions

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah SELECT dan WHERE dalam query.

g. Security

MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses user dan sistem perizinan yang mendetail serta passwrd terenkripsi.

h. Scalability dan Limits

MySQL mampu menangani database dalam skala besar dengan jumlah records lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selain itu, batas index yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

i. Connectivity

MySQL dapat melakukan koneksi dengan klien menggunakan TCP/IP, Unix soket (Unix), atau Named Pipes (NT).

j. Localization

MySQL dapat mendeteksi pesan kesalahan (error code) pada klien dengan menggunakan lebih dari dua puluh bahasa.

k. Interface

MySQL memiliki interface terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface)

l. Clients and Tools

MySQL dilengkapi dengan berbagai tool yang dapat digunakan untuk adminstrasi dataase, dan pada setiap tool yang ada disertakan petunjuk online.

m. Strukrture Table

MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE dibandngkan database lainnya.

2.2.16.3. JQuery

JQuery adalah Javascript Library atau kumpulan kode/fungsi Javascript siap pakai, sehingga mempermudah kita untuk membuat kode Javascript. Atau dalam kesimpulannya jQuery menyederhanakan kode Javascript. Hal ini sesuai dengan slogannya"*Writeless, do more*". Ada beberapa alasan mengapa lebih baik menggunakan jQuery daripada library lainnya, antara lain:

a. Kompatibel dengan hampir seluruh browser
b. jQuery telah digunakan oleh website-website raksasa
c. Kompatibel dengan seluruh versi CSS (dari CSS 1 sampai CSS 3)
d. Dokumentasi dan tutorialnya lengkap
e. Didukung oleh banyak komunitas
f. Disupport oleh plugin yang lengkap
g. Filenya hanya satu dan ukurannya relatif kecil, sekitar 20kb
h. Open source atau Free
i. jQuery lebih diminati oleh para developer web saat ini

Berikut beberapa kemampuan yang dimiliki:

a. Mempermudah akses dan manipulasi elemen tertentu pada dokumen.
b. Mempermudah modifikasi/perubahan tampilan halaman web.

c. Mempersingkat Ajax (Asynchronous Javacsript and XML).
d. Memiliki API (Application Programming Interface).
e. Mampu merespon interaksi antara user dengan halaman web dengan lebihcepat.
f. Menyediakan fasilitas untuk membuat animasi sekelas Flash denganmudah.

2.2.17. Penerimaan Siswa baru

Penerimaan siswa baru merupakan sebuah kegiatan rutin pada awal tahun ajaran di setiap lembaga penididkan khusunya Sekolah Menengah Kejuruan untuk menyeleksi calon siswa baru. Pada umumnya alur penerimaan siswa baru adalah setiap calon siswa diharuskan mendaftarkan diri secara pribadi atau kolektif dengan mengisi formulir dan melengkapi berkas persyaratan yang telah ditetapkan.