BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Sekolah

2.1.1 Sejarah Berdirinya TamanSiswa

Berbagai realisasi dari tugas menyelenggarakan pendidikan untuk anak – anak, Suwardi Suryaningrat mendirikan Perguruan Nasional Tamansiswa pada tanggal 3 Juli 1922.

Maksud dan tujuan mendirikan Tamansiswa adalah sebagai berikut :

a. Pendidikan dan pengajaran bagi bangsa sebagai perwujudan pemeliharaan dan pengembangan turunan bangsa, agar tumbuh dengan sehat lahir batinnya untuk mencapai masyarakat yang berkebudayaan.

b. Pendidikanan pengajaran bagi tiap pribadi (individu) harus mengembangkan jiwa dan raganya.

c. Segala peralatan yang digunakan hendaklah berdasarkan adat istiadat rakyat, agar dapat tumbuh dengan cepat menurut kodratnya sesuai dengan kemajuan bangsanya.

d. Pengajaran barat tidak luput dari pengaruh kolonial, ialah seorang didikan untuk keperluan mendidiknya.

Adapun para pendiri Tamansiswa adalah :

1. Ki Suwardi Suryaningrat
2. Ki Sutatmo Suryokusumo
3. Ki Sutatmo Suryokusumo
4. Ki Cokrodirjo

5. Ki pronowidigdo
6. Nyi sutartinah Suwardi
7. Ki sutopo Wonoboyo
8. Ki Subono

Menurut kenyataan Perguruan Tamansiswa mendapatkan sambutan positif dari masyarakat, sehingga tahun1923 perlu membentuk organisasi untuk menyelenggarakannnya, dengan ditandai candraserigkala " Suci Tata Ngesti Tunggal " artinya " Dengan suci tertib mencapai persatuan", semboyan tersebut merupakan janji bagi para anggotanya.

Persatuan Tamansiswa yang baru dibentuk itu diimpin oleh pengurus yang disebut " *Instituutraad* " dengan susunan sebagai berikut :

- Panitera Umum : Ki Suwardi Suryaningrat
- Ketua I : Ki Sutatmo Suryokusumo
- Ketua II : Ki Sutatmo Suryokusumo
- Anggota : Ki pronowidigdo, Ki sutopo Wonoboyo, Ki Subono, Ki Rujito, Ki Wiryodiharjo.

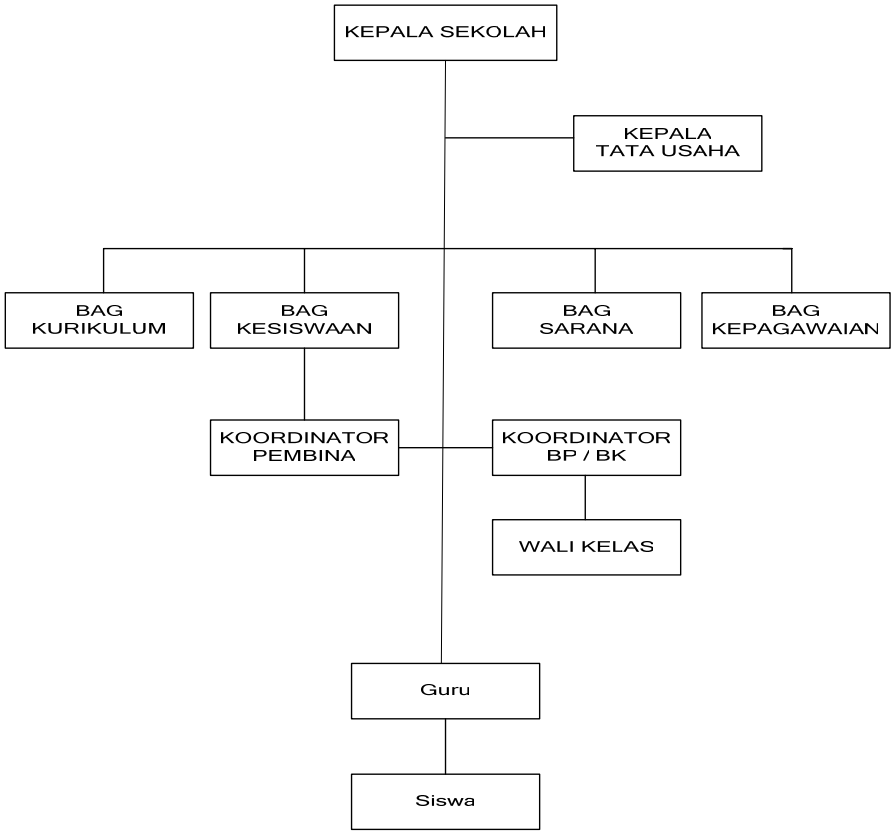
2.1.2 Perkembangan Tamansiswa

SMA Tamansiswa (Taman Madya) cabang Bandung berdiri sejak tahun 1951, status sekolah terakreditasi berdasarkan SK Badan Akreditasi Sekolah Provinsi Jawa Barat No. 420/599-Dikmenti

Kualifikasi Staf Pengajar 85 % Sarjana IKIP (UPI) Bandung, 10 % Universitas Tamansiswa Yogyakarta dan 5 % Lembaga Pendidikan Tinggi

Lainnya (ITB, UNPAD, IAIN). Untuk fasilitas gedung yang digunakkan dalam kegiatan belajar mengajar merupakan milik sendiri yang berdiri di atas tanah seluas 4.000 meter persegi terdiri atas 12 lokal bangunan tingkat tiga dan 5 lokal bangunan tingkat dua serta bangunan kantor Majelis Cabang yang bertingkat dua.

2.1.3 Struktur Organisasi



Gambar 2.1 Struktur Organisasi SMA Tamansiswa Bandung

2.1.4 Uraian Tugas

Pembagian tugas yang ada di SMA Tamansiswa bandung adalah sebagai berikut :

2.1.4.1 Tugas Kepala Sekolah

a) Rumusan umum tugas dan fungsi

Menyusun dan melaksanakan program kerja, mengarahkan, membina, memimpin, mengawasi serta mengkoordinasikan pelaksanaan tugas bidang administrasi dan keuangan sekolah, ketenagaan, kesiswaan, sarana dan prasarana, pencapaian kurikulum, kerjasama dengan dunia yang relevan serta memasarkan tamatan SMA

b) Uraian tugas Kepala Sekolah

1. Mengatur penyelenggaraan pendidikan dan pengajaran
2. Mengatur penyelenggaraan tata usaha sekolah
3. Merencanakan dan membina pengembangan profesi, karir guru dan staf.
4. Memonitor dan mengevaluasi kegiatan program kerja sekolah
5. Membina dan mengawasi pengelolaan penyesuaian dan pelaksanaan kurikulum.
6. Merencanakan dan mengawasi pelaksanaan penerimaan siswa baru.
7. Membina kesiswaan.
8. Membina kegiatan penelusuran kelulusan
9. Memelihara dan mengembangkan organisasi dan manajemen sekolah.

2.1.4.2 Tugas Jabatan Kepala Tata Usaha

a) Rumusan Umum Tugas dan Fungsi

Membantu Kepala Sekolah dalam urusan kesiswaan, yaitu dalam menyusun program kerja pembinaan kesiswaan, kegiatan luar sekolah dan mengkoordinir pelaksnaannya.

b) Uraian Tugas bagian Kesiswaan

1. Menyusun program kerja pembinaan siswa (mingguan, semesteran dan tahunan) dan mengkoordinir pelaksanaannya.
2. Mengkoordinasikan pelaksaanan pemilihan penyusunan OSIS, Pramuka, Paskibra, PMR dan lain-lain.
3. Mengkoordinasikan perencanaan dan pelaksanaan kegiatan luar sekolah / ekstra kurikuler
4. Membimbing dan mengawasi kegiatan OSIS, Pramuka, Paskibra, PMR dan lain-lain.
5. Membimbing dan mengawasi pengembangan hubungan siswa dengan siswa sekolah lain melalui organisasi sekolah yang ada.
6. Mengawasi dan mengevaluasi pelaksanaan pembinaan kesiswaan
7. Mengkoordinir kegiatan upacara-upacara sekolah
8. Membuat laporan berkala dan insidentil kepada Kepala Sekolah.

2.1.4.3 Tugas Jabatan Bagian Kurikulum

a) Rumusan Umum Tugas dan Fungsi

Membantu Kepala Sekolah dalam pelaksanaan kegiatan kurikulum dan ekstra kurikuler.

b) Uraian Tugas bagian Kurikulum

1. Menyusun program kerja tahunan (*Action Plan*)
2. Mengkoordinir pemasyarakatan dan pengembangan kurikulum.
3. Menyusun program pembelajaran (mingguan, bulanan, semesteran, dan tahunan) dan mengkoordinasikan pelaksanaannya.
4. Mengkoordinasikan persiapan dan pelaksanaan UNAS, uji profesi tes kejuruan, dan sebagainya
5. Menganalisis ketercapaian target kurikulum dan daya serap
6. Menyusun laporan berkala dan insidentil tentang kegiatan kurikuler dan ekstra kurikuler.
7. Mengkoordinir pelaksanaan Penerimaan Siswa Baru
8. Mengkoordinir pelaksanaan tugas Wali Kelas dan Bimbingan Karir Kejuruan.
9. Mewakili Sekolah dalam kegiatan yang berhubungan dengan pengembangan kurikulum.
10. Mengkoordinir pelaksanaan kegiatan Pokja Kurikulum Sekolah
11. Menyusun laporan kegiatan berkala dan akhir tahun pelajaran Kepala Sekolah.

2.1.4.4 Tugas Bagian Kesiswaan

a. Rumusan Umum Tugas dan Fungsi

Membantu Kepala Sekolah dalam urusan kesiswaan, yaitu dalam menyusun program kerja pembinaan kesiswaan, kegiatan luar sekolah dan mengkoordinir pelaksnaannya.

b. Uraian Tugas Jabatan bagian kesiswaan

1. Mengkoordinasikan pelaksaanan pemilihan penyusunan OSIS, Pramuka, Paskibra, PMR dan lain-lain.
2. Mengkoordinasikan perencanaan dan pelaksanaan kegiatan luar sekolah / ekstra kurikuler.
3. Membimbing dan mengawasi kegiatan OSIS, Pramuka, Paskibra, PMR dan lain-lain.
4. Membina kepengurusan OSIS, Pramuka, Paskibra, PMR dan lain-lain.
5. Mengkoordinir pelaksanaan pemilihan calon siswa teladan, penerima beasiswa, dan Paskibra.
6. Membimbing dan mengawasi pengembangan hubungan siswa dengan siswa sekolah lain melalui organisasi sekolah yang ada.

   Mengawasi dan mengevaluasi pelaksanaan pembinaan kesiswaan
7. Mengkoordinir kegiatan upacara-upacara sekolah.

   Membuat laporan berkala dan insidentil kepada Kepala Sekolah

2.1.4.5 Tugas Jabatan Wali Kelas

a. Rumusan Umum Tugas dan Fungsi

Membantu siswa, guru BK, bendahara sekolah dalam memecahkan masalah siswa dengan cara memahami karakter siswa dan membina hubungan baik orang tua murid/wali, serta melaksanakan tugas administrasi kelas yang diasuhnya.

b. Uraian Tugas Jabatan Wali Kelas

1. Menyusun program kerja wali kelas.
2. Mengatur tempat duduk siswa di kelas dan membuat *layout* kelas.
3. Menjalin hubungan dengan orang tua siswa.
4. Menghubungi orang tua/wali apabila perlu.
5. Membantu bendahara sekolah dalam pengumpulan pembayaran SPP dan sumbangan lainnya.
6. Memahami karakter siswa dari kelas yang diasuhnya.
7. Mengumpulkan nilai dari para guru dan memasukkannya ke dalam buku/ Daftar Kumpulan Nilai (DKN).
8. Mengisi dan membagi raport.
9. Membantu guru BK dalam menangani siswa yang menghadapi masalah.
10. Membina budi pekerti siswa.
11. Membantu siswa dalam memecahkan masalahnya

2.1.4.6 Tugas Jabatan Guru

a. Rumusan Umum Tugas dan Fungsi

Memberikan pendidikan / kepengajaran teori dan praktek kepada siswa dan melaksanakan tugas teknik kependidikan lainnya yang dibebankan oleh Kepala Sekolah.

b. Uraian Tugas Jabatan Guru

1. Menyiapkan perangkat semester, analisa program, stuan pelajaran dan kisis-kisi berikut perangkat evaluasi.
2. Melaksanakan administrasi siswa (daftar nilai, daftar hadir, dan daftar kemajuan siswa).
3. Melaksanakan kegiatan proses belajar mengajar 24 jam pelajaran

2.2 Landasan Teori

Pada landasan teori ini diuraikan sejumlah teori untuk membantu dan memecahkan permasalahan yang ada. Beberapa landasan teori tersebut meliputi konsep dasar dan definisi-definisi yang berkaitan dengan perangkat lunak yang digunakan serta faktor-faktor pendukung dalam pelaksanaan perancangannya

2.2.1 Definisi Sistem Informasi

Sistem informasi didefinisikan oleh Robert A. Leitch dan K. Roscoe Davis sebagai berikut:[1]

> *Sistem informasi adalah suatu sistem didalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.*

2.2.2 Komponen Sistem Informasi

John Burch dan Gary Grudnitski mengemukakan bahwa sistem informasi terdiri dari komponen-komponen yang disebutnya dengan istilah blok bangunan (*building block*), yaitu blok masukan (*input block*), blok model (*model block*),

blok keluaran (*output block*), blok teknologi (*technologi block*), blok basis data (*database block*), dan blok kendali (*komputers block*). Sebagai suatu sistem, keenam blok tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasarannya.[1]

1. Blok Masukan

   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.

2. Blok 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.

3. Blok Keluaran

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

4. Blok Teknologi

   Teknologi merupakan “kotak alat” (*tool box*) dalam sistem informasi. Teknologi digunakan untuk menerima output, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 bagian utama, yaitu teknisi (*humanware* atau *brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*). Teknisi dapat berupa

orang-orang yang mengetahui teknologi dan membuatnya dapat beroperasi. Misalnya teknisi adalah operator komputer, pemrogram, operator pengolah kata, spesialis telekomunikasi, analis sistem, penyimpanan data dan lain sebagainya.

5. Blok Basis Data

   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. Data perlu disimpan di dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa, supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut dengan DBMS (*Database Management Systems*).

6. Blok Kendali

   Banyak hal yang dapat merusak sistem informasi, seperti misalnya bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, kesalahan-kesalahan, ketidak efisienan, sabotase dan lain sebagainya. 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.

2.2.3 Arsitektur Informasi

Arsitektur Informasi adalah suatu pemetaan atau rencana kebutuhan-kebutuhan informasi di dalam organisasi. Arsitektur ini berguna sebagi penuntun bagi operasi sekarang atau menjadi *blue print* untuk arahan di masa mendatang.[2]

Tujuan dari arsitektur ini adalah agar bagian teknologi informasi memenuhi kebutuhan-kebutuhan bisnis strategi organisasi oleh karena itu arsitektur informasi memadukan kebutuhan informasi, komponen sistem informasi, dan teknologi pendukung.[2]

2.3 Rekayasa Perangkat Lunak

2.3.1 Definisi Rekayasa Perangkat Lunak

Rekayasa perangkat lunak adalah pengembangan dari penggunaan prinsip pengembangan suara untuk memperoleh perangkat lunak secara otomatis yang reliabel dalam bekerja secara efisien pada mesin nyata.[3]

2.3.2 Model Proses Rekayasa Perangkat Lunak

Model proses untuk rekayasa perangkat lunak dipilih berdasarkan sifat aplikasi dan proyeknya, metode dan alat-alat bantu yang akan dipakai, dan kontrol serta penyampaian yang dibutuhkan. Perkembangan perangkat lunak bisa dianggap sebagai lingkaran pemecahan masalah dimana terdapat empat keadaan berbeda, yaitu status quo, definisi masalah, perkembangan teknis memecahkan masalah di keseluruhan aplikasi dari banyak aplikasi, dan integrasi pemecahan menyampaikan hasil kepada siapa yang membutuhkan pertama kali. Bermacam-macam model proses yang berbeda pada perangkat lunak sebagai berikut:[3]

1. Model Sekuensial Linier atau *Waterfall*

   Sekuensial linier mengusulkan sebuah pendekatan kepada perkembangan perangkat lunak yang sistematik dan sekuensial yang mulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, pengujian, dan pemeliharaan.

2. Model Prototipe

   Prototyping paradigma dimulai dengan pengumpulan kebutuhan. Pengembang dan pelanggan bertemu dan mendefinisikan obyektif keseluruhan dari perangkat lunak, mengidentifikasi segala kebutuhan yang diketahui, dan area garis besar dimana definisi lebih jauh merupakan keharusan kemudian dilakukan “perancangan kilat”. Perancangan kilat berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut,yang akan nampak bagi pelanggan / pemakai (contohnya pendekatan input dan format output).

   Pada model ini dimulai dengan pengumpulan informasi mengenai kebutuhan, dimana :

   1. Pengembang dan pemesan bertemu dan mendefinisikan sasaran-sasaran umum;
   2. Mengidentifikasikan kebutuhan yang telah diketahui, dan
   3. Mencari bidang-bidang yang masih memerlukan pendefinisian.

   Setelah itu pengembang melakukan “perancangan kilat” terhadap kebutuhan yang telah teridentifikasi pada pertemuan. Perancangan berfokus pada representasi yang tampak oleh pengguna. Perancangan ini menuntun pembangunan *Prototype* perangkat lunak yang akan diberikan pembeli/pemakai. *Prototype* itu dievaluasi olah pemakai dan digunakan

sebagai landasan untuk memperbaiki spesifikasi kebutuhan. Proses ini berulang sampai *Prototype* yang dikembangkan memenuhi kebutuhan pemakai, dan pengembang telah memahami permasalahan dengan lebih baik.[3]

3. Model RAD

   Rapid Application Development (RAD) adalah sebuah model proses perkembangan perangkat lunak sekuensial linier yang menekankan siklus perkembangan yang sangat pendek. Model RAD ini merupakan sebuah adaptasi “kecepatan tinggi” dari model sekuensial linier di mana perkembangan cepat dicapai dengan menggunakan pendekatan konstruksi berbasis komponen.

4. Model Proses Perangkat Lunak Evolusioner

   Model evolusioner adalah model iteratif. Model itu ditandai dengan tingkah laku yang memungkinkan perekayasa perangkat lunak mengembangkan versi perangkat lunak yang lebih lengkap sedikit demi sedikit.

   a. Model Pertambahan

      Model inkeremental menggabungkan elemen-elemen model sekuensial linier (diaplikasikan secara berulang) dengan filosofi prototipe iteratif. Model pertambahan memakai urutan-urutan linier di dalam model yang membingungkan, seiring dengan laju waktu kalender. Model pertambahan berfokus pada penyampaian produk operasional dalam setiap pertambahannya.

b. Model Spiral

Model spiral (*spiral model*) adalah model proses perangkat lunak yang evolusioner yang merangkai sifat iteratif dari prototipe dengan cara komputer dan aspek sistematis dari model sekuensial linier.

c. Model Rakitan Komponen

Model ratikan komponen menggabungkan beberapa karakteristik model spiral. Model ini bersifat evolusioner, sehingga membutuhkan pendekatan iteratif untuk mencapai perangkat lunak. Tetapi model rakitan komponen merangkai aplikasi dari komponen perangkat lunak sebelum dipaketkan (kadang-kadang disebut “kelas”).

d. Model Perkembangan Konkuren

Model proses yang konkuren dapat disajikan secara skematis sebagai sederetan aktivitas teknik mayor, tugas-tugas, dan keadaannya yang lain. Contohnya aktivitas rekayasa yang dibatasi untuk model spiral dipenuhi dengan melakukan tugas-tugas sebagai berikut*: prototyping* dan atau pemodelan analisis, spesifikasi kebutuhan, dan rancangan.

5. Model Formal

Model metode formal mencakup sekumpulan aktivitas yang membawa kepada spesifikasi matematis perangkat lunak komputer. Metode formal memungkinkan perekayasa perangkat lunak untuk mengkhususkan, mengembangkan, dan memverifikasi sistem berbasis komputer dengan menggunakan notasi matematis yang tepat.

2.4 Konsep Merancang Basis Data

2.4.1 Definisi Basis data

Basis data dimaksudkan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas.[4]

Sistem basis data adalah suatu sistem menyusun dan mengelola record-record menggunakan komputer untuk menyimpan atau merekam serta memelihara data opersional lengkap sebuah organisasi/perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk proses mengambil keputusan.[4]

Untuk mengelola basis data diperlukan perangkat lunak yang disebut DBMS. DBMS adalah perangkat lunak sistem yang memungkinkan para pemakai membuat, memelihara, mengontrol, dan mengakses basis data dengan cara yang praktis dan efisien.[4]

2.4.2 Tahap Perancangan Basis Data

Perancangan basis data merupakan langkah untuk menentukan basis data yang diharapkan dapat mewakili seluruh kebutuhan pengguna. Perancangan basis data terdiri atas perancangan basis data secara konseptual, perancangan basis data secara logis, dan perancangan basis data secara fisis.[4]

2.4.3 Entity Relationship Diagram

ERD hanya berfokus pada data, dengan menunjukkan “jaringan data” yang ada untuk suatu sistem yang diberikan. ERD sangat berguna bagi aplikasi di mana data dan hubungan yang mengatur data sangatlah kompleks. ERD pada mulanya

diusulkan oleh Peter Chen untuk desain sistem *database* relasional dan telah dikembangkan oleh yang lainnya. Serangkaian komponen utama diidentifikasikan untuk ERD : objek data, atribut, hubungan dan berbagai tipe indikator. Tujuan utama dari ERD adalah untuk mewakili objek data dan hubungan mereka.[4]

Kardinalitas model data harus dapat merepresentsikan jumlah peristiwa dari objek di dalam hubungan yang diberikan. Tillmann mendefinisikan kardinalitas dari *object-relationship pair* dengan cara sebagai berikut: kardinalitas merupakan spesifikasi dari sejumlah peristiwa dari satu [objek] yang dapat dihubungkan ke sejumlah peristiwa dari [objek] yang lain. [4]

Secara skematik, basis data dapat dideskripsikan secara grafis dengan ERD yang memiliki komponen utama, yaitu :[4]

1. Entitas

   Entitas adalah sesuatu atau objek pada dunia nyata yang dapat dibedakan satu terhadap yang lainnya, yang bermanfaat bagi aplikasi yang sedang kita kembangkan. Entitas digambarkan dengan empat persegi panjang.

2. Atribut

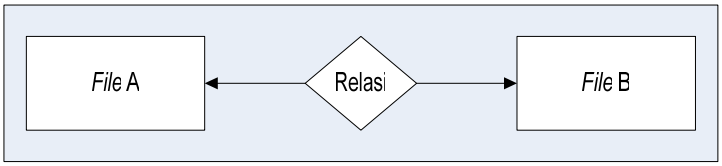
   Atribut adalah komponen untuk mendeskripsikan entitas yang digambarkan dengan bentuk *elips*.

3. Relasi

   Relasi adalah hubungan antara beberap entitas yang digambarkan dengan belah ketupat. Relasi antar tabel dapat dikategorikan menjadi tiga macam, yaitu :

a. *One to One Relationship*

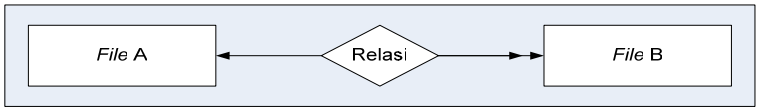
Hubungan *one to one relationship* adalah antara *file* pertama dengan *file* kedua adalah satu banding satu.



Gambar 2.2 *One to One Relationship*

b. *One to Many Relationship*

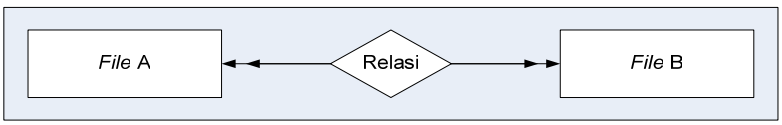
Hubungan *one to many relationship* adalah sebuah relasi merupakan tahap dimana hubungan antara *file* pertama dengan *file* kedua adalah satu banding banyak.



Gambar 2.3 One to Many Relationship

c. *Many to Many Relationship*

Hubungan *many to many relationship* adalah sebuah relasi antara suatu *file* satu dengan *file* yang kedua memepunyai relasi banyak berbanding banyak.



Gambar 2.4 Many to Many Relationship

4. Garis

   Garis adalah yang menyatukan atribuut-atribut pada entitas tertentu serta menyatukan entitas-entitas dalam suatu relasi tertentu.

2.5 Metode Analisis Sistem

2.5.1 Diagram Konteks

Diagram konteks menggambarkan hubungan antara sistem dengan entitas luarnya. Diagram konteks berfungsi sebagai transformasi dari satu proses yang melakukan transformasi data *input* menjadi data *output*. Entitas yang dimaksud adalah entitas yang mempunyai hubungan langsung dengan sistem.[3]

Suatu diagram konteks selalu mengandung satu dan hanya satu proses saja. Diagram konteks ini menggambarkan hubungan *input* atau *output* antara sistem dengan dunia luarnya (kesatuan luar).[3]

2.5.2 Entity- Relationship Diagram

ERD hanya berfokus pada data, dengan menunjukkan “jaringan data” yang ada untuk suatu sistem yang diberikan. ERD sangat berguna bagi aplikasi di mana data dan hubungan yang mengatur data sangatlah kompleks. ERD pada mulanya diusulkan oleh **Peter Chen** untuk desain sistem *database* relasional dan telah dikembangkan oleh yang lainnya. Serangkaian komponen utama diindentifikasikan untuk ERD: objek data, atribut, hubungan, dan berbagai tipe indikator. Tujuan utama dari ERD adalah untuk mewakili objek data dan hubungan mereka.

Kardinalitas model data harus dapat merepresentsikan jumlah peristiwa dari objek di dalam hubungan yang diberikan. **Tillmann** mendefinisikan kardinalitas dari **object-relationship pair** dengan cara sebagai berikut: kardinalitas merupakan spesifikasi dari sejumlah peristiwa dari satu [objek] yang dapat dihubungkan ke sejumlah peristiwa dari [objek] yang lain. Dengan mempertimbangkan semua kombinasi dari ‘satu’ dan ‘banyak’, dua [objek] dapat dihubungkan sebagai:

1. Satu-ke-satu (1:1)
2. Satu-ke-banyak (1:N)
3. Banyak-ke-satu (N:1)
4. Banyak-ke-banyak (M:N)

2.5.3 *Data Flow Diagram* (*DFD*)

Data Flow Diagram (DFD–DAD/Diagram Alir Data) memperlihatkan hubungan fungsional dari nilai yang dihitung oleh sistem, termasuk nilai masukan, nilai keluaran, serta tempat penyimpanan internal. DFD adalah gambaran grafis yang memperlihatkan aliran data dari sumbernya dalam objek kemudian melewati proses yang mentransformasinya ke tujuan yang lain, yang ada pada objek lain. DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangan lingkungan fisik dimana data tersebut mengalir.[3]

DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur (*structured analysis and design*). DFD merupakan alat

yang cukup populer sekarang ini, karena dapat menggambarkan arus data di dalam sistem dengan terstruktur jelas.[3]

Beberapa simbol yang digunakan dalam *Data Flow Diagram* (DFD) antara lain:[3]

1. *External Entity* (kesatuan luar) atau *boundary* (batas sistem)

   Setiap sistem pasti mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima input dan menghasilkan output kepada lingkungan luarnya. Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan *input* atau menerima *output* dari sistem.

2. *Data Flow* (arus data)

   Arus data (*data flow*) di DFD diberi simbol suatu panah. Arus data ini mengalir diantara proses (*process*), simpanan data (*data strore*) dan kesatuan luar (*external entity*). Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

3. *Process* (proses)

   Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang , mesin atau kompuiter dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses. Untuk *physical data flow diagram* (PDFD), proses dapat dilakukan oleh orang, mesin atau komputer, sedangkan untuk *logical data flow diagram* (LDFD), suatu proses hanya menunjukkan proses dari komputer. Setiap proses harus diberi penjelasan yang lengkap meliputu identifikasi proses, nama proses dan pemroses.

4. *Data Store* (simpanan luar)

   Simpanan data (*data store*) merupakan simpanan dari data yang dapat berupa, yaitu suatu file atau *database* di sistem komputer, suatu arsip atau catatan manual, suatu kotak tempat data di meja seseorang, suatu tabel acuan manual, dan suatu agenda atau buku.

2.5.4 Kamus Data

Kamus data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data, analisis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap. Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem. [3]

Pada tahap analisis dan perancangan, kamus data dapat digunakan sebagai alat komunikasi antara analsisis sistem dengan pemakai sistem tentang data yang mengalir di sistem, yaitu tentang data yang masuk ke sistem dana tentang informasi yang dibutuhkan oleh pemakai sistem.[3]

Pada tahap perancangan, kamus data digunakan untuk merancang *input*, merancang laporan-laporan dan *database*. Kamus data dibuat berdasarkan arus data yang ada di DFD. Arus data di DFD sifatnya global, hanya ditunjukkan nama arus datanya saja. Keterangan lebih lanjut tentang struktur data dari arus data di DFD secara lebih rinci dapat dilihat di kamus data. Kamus data harus dapat mencerminkan keterangan yang jelas tentang data yang dicatatnya, maka kamus data harus memuat nama arus data, alias, bentuk data, arus data, penjelasan, periode, volume, dan struktur data.[3]

2.6 Intranet dan Aplikasi *Web*

Intranet (*Internal Network*) mulai terdengar pada pertengahan tahun 1995 oleh beberapa penjual produk jarinagn yang mengacu pada kebutuhan informasi dalam bentuk *web* dalam perusahaan. Intranet adalah jaringan komputer dalam perusahaan yang menggunakan komunikasi data standar seperti dalam internet. Artinya, kita dapat menggunakan semua fasilitas internet untuk kebutuhan dalam perusahaan. Dengan kata lain intranet dapat dikatakan berinternet dalam lingkungan perusahaan. Standar internet yang digunakan adalah standar protokol TCP/IP (*Transport Control Protocol*, ada yang menyebutnya *Transmission Control Protocol/Internetworking Protocol*), TCP/IP memungkinkan protokol jaringan untuk berkomunikasi, menerima dan mengirimkan data ke terminal yang lain. Standar yang lain adalah FTP (*File Transfer Protocol*) yang merupakan pelayanan *resource sharing*, sebuah fasilitas untuk dapat mengambil file yang ada di internet. SMTP (*Simple Mail Transfer Protocol*) yang merupakan dasar dari e-mail untuk berkomunikasi serta MIME (*Multipurpose internet Mail Extensions*) yang merupakan standar untuk mendefinisikan format biner, grafik dan suara agar dapat ditransmisikan dengan e-mail. Protokol lainnya adalah NNTP (*Network News Transfer Protocol*) dan POP (*Post Office Protocol*).[5]

Pada awalnya aplikasi *Web* dibangun hanya dengan menggunakan bahasa yang disebut HTML (*HyperText Markup Language)*. Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML. Pada saat ini, banyak skrip seperti itu, antara lain yaitu PHP dan ASP, sedangkan contoh yang berupa objek adalah *applet*. Aplikasi *Web* itu dapat dibagi menjadi *Web* statis dan *Web* dinamis.

Web statis dibentuk dengan menggunakan HTML saja. Kekurangan aplikasi seperti ini terletak pada keharusan untuk memelihara program secara terus-menerus untuk mengikuti setiap perubahan yang terjadi. Kelemahan ini diatasi dengan model *Web* dinamis.

Dengan menggunakan pendekatan Web dinamis, dimungkinkan untuk membentuk sistem informasi berbasis web. Dari sisi teknologi yang digunakan untuk membentuk *web* dinamis terdapat dua pengelompokan, yaitu teknologi pada sisi klien dan teknologi pada sisi server.

Teknologi *Web* pada sisi klien diimplementasikan dengan mengirimkan kode perluasan HTML atau program tersendiri dan HTML ke klien. Klienlah yang bertanggung jawab dalam melakukan proses terhadap seluruh kode yang diterima. Kelemahan pendekatan seperi ini adalah terdapat kemungkinan bahwa browser pada klien tidak mendukung fitur kode perluasan HTML. Kelebihan teknologi pada sisi klien, yaitu memungkinkan penampilan yang bersifat dinamis. Contoh teknologi pada sisi klien, yaitu Kontrol ActiveX, Java Applet, dan Skrip sisi-klien

Teknologi *Web* pada sisi server memungkinkan pemrosesan kode di dalam server sehingga kode yang sampai pada pemakai berbeda dengan kode asli pada server. Contoh teknologi yang berjalan di server, yaitu CGI, ASP, JSP, PHP dan lain sebagainya. Keuntungan penggunaan teknologi pada sisi server adalah sebagai berikut:

1. Mengurangi lalu lintas jaringan dengan cara menghindari percakapan bolak-balik antara klien dan server.
2. Mengurangi waktu pemuatan kode, menginat klien hanya mengambil kode HTML saja.

3. Mencegah masalah ketidakkompatibelan browser.
4. Klien dapat berinteraksi dengan data yang ada pada server.
5. Mencegah klien mengetahui rahasia kode (mengingat kode yang diberikan ke klien berbeda dengan kode asli pada server).[5]

2.7 PHP

PHP merupakan bahasa pemrogramana untuk *script web server-side*. Bahasa pemrograman PHP diciptakan pertama kali oleh **Rasmus Lerdorf**, seorang pemrogram C yang sangat handal. Semula PHP hanya digunakan untuk mencatat seberapa jumlah pengunjung pada *hompage*-nya. **Rasmus** adalah salah seorang pendukung *open source*. Karen itulah ia mengeluarkan *Personal Home Page Tools* versi 1.0 secara gratis atau *freeware* pada tahun 1995. Setelah mempelajari YACC dan GNU Bison, **Rasmus** menambah kemampuan pada PHP 1.0 dan menertbitkan PHP 2.0 sebagai pengembangan dari PHP 1.0 yang telah ada sebelumnya.

PHP 2.0 mampu berhubungan dengan *database* dan dapat diintegrasikan dengan HTML. Pada tahun 1996, PHP telah digunkana oleh banyak *website* di dunia. Sebuah kelompok pengembang software yang terdiri dari **Rasmus**, **Zeew Suraski**, **Andi Gutman**, **Stig Bakken**, **Shane Carveo** dan **Jim Winstead** bekerja selama tujuh bulan untuk melakukan penyempurnan terhadap PHP 2.0, akhirnya pada tanggal 6 juni 1998, PHP 3.0 resmi dikeluarkan ke dunia pemrograman. Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai. Versi ini banyak dipakai sebab versi ini mampu dipakai untuk membangun aplikasi *web* kompleks tetapi tetap memiliki kecepatan proses dan stabilitas yang

tinggi. PHP versi 4.2 telah diterbitkan pada tanggal 22 April 2002 dengan log kelompok fungsi, sampai dengan versi 4.3.7 tercatat 125 kelompok fungsi yang dimiliki oleh PHP. Pada juni 2004, Zend merilis PHP 5.0. Versi ini adalah versi yang paling mutakhir dari PHP.

2.7.1 Kelebihan PHP

PHP memiliki beberapa kelebihan yang tidak dimiliki oleh bahasa-bahasa sejenisnya, yaitu:

1. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaanya.
2. *Web Server* yang mendukung PHP dapat ditemukan dimana - mana dari mulai IIS sampai dengan apache, dengan configurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan *developer* yang siap membantu dalam pengembangan.
4. Dalam sisi pemahamanan, PHP adalah bahasa *scripting* yang paling mudah karena referensi yang banyak.
5. PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (*linux*, *unix*, *windows*) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem.

2.7.2 Dasar – Dasar PHP

Browser – browse sekarang ini umumnya telah sanggup mengeksekusi *client-side script*, yaitu *script* yang dijalankan disisi client. PHP menggunakan *script* C, dan biasanya dieksekusi *di server*sehingga disebut *client-side script*,

sehingga untuk membedakannya dari *servert-side script* perlu ada suatu mekanisme tertentu.[6]

2.7.3 Variabel pada PHP

PHP mendukung beberapa jenis variabel, diantaranya adalah:

1. *String*, merupakan jenis data karakter
2. *Integer*, digunakan untuk menyimpan bilangan bulat
3. *Double*, digunakan untuk menyimpan bilangan bernilai pecahan dan bilangan bernilai pemangkat.
4. *Array*, digunakan untuk menyimpan beberapa elemen dalam satu variabel.
5. *Object*, digunakan berdasarkan gambaran objek pada dunia nyata, yang mempunyai ”status” dan ”tingkah laku”. Sebuah variabel objek menyimpan statusnya dalam bentuk variabel dan tingkah lakunya berupa parameter.
6. *Pdfdoc*, digunakan jika dukungan *pdf* diaktifkan.
7. *pdfinfo*, sama halnya dengan *pdfdoc*, *Pdfinfo* juga dapat digunakan jika dukungan *pdf* diaktifkan.

Jenis variabel biasanya tidak ditentukan oleh pemrograman, tetapi ditentukan oleh PHP saat *runtime*, tergantung pada konteks apa variabel itu digunakan. Misalnya variabel coba berisikan nilai string, maka variabel coba berisi string atau integer bila berisi nilai integer.[6]

2.8 MYSQL

MySQL (*My Structure Query Language*) adalah sebuah program pembuat *database* yang bersifat *open source*. MySQL sebenarnya produk yang berjalan pada *platform Linux*. Karen sifatnya *open source*, maka MySQL dapat dijalankan pada semua *platform* baik *Windows* maupun *Linux*. Selain itu, MySQL juga merupakan program pengakses *database* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *Multi User* (Banyak Pengguna). Saat ini *database* MySQL telah digunakan hampir oleh semua *programer database*, apalagi dalm pemrograman *web*.

Kelebihan lain dari MySQL adalah menggunakan bahasa *Query* standar yang dimiliki SQL (*Structure Query Language*). SQL adalah suatu bahasa permintaan yang terstruktu yang telah distandarkan untuk semua program pengakses *database* seperti Oracle, Posgres SQL, SQL Server, dan lain-lain.

Sebagai sebuah program penghasil *database*, MySQL tidak dapat berjalan sendiri tanpa adanya sebuah aplikasi lain (*interface*). MySQL dapat didukung oleh hampir semua program aplikasi baik yang *open source* seperti PHP maupun yang tidak, yang ada pada *platform Windows* seperti Visual Basic, Delphi, dan lainnya.[6]