

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Sistem Informasi**

Menurut Al-Bahra (2013:13) Sistem informasi dapat didefinisikan sebagai berikut:

- a. Suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.
- b. Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan/atau untuk mengendalikan organisasi.
- c. Suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Menurut Tata Sutabri (2012:46), sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategis dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan.

Dari kedua kutipan pengertian sistem informasi penulis mengambil kesimpulan bahwa sistem informasi adalah kumpulan dari sub-sub sistem yang saling berhubungan dan bekerja sama untuk mendukung pengambilan keputusan dan untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berguna.

#### **2.2 Perizinan**

Perizinan adalah pemberian legalitas kepada seseorang atau pelaku usaha/kegiatan tertentu, baik dalam bentuk izin maupun tanda daftar usaha. Izin ialah salah satu instrumen yang paling banyak digunakan dalam hukum administrasi, untuk mengemudikan tingkah laku para warga. Selain itu izin juga dapat diartikan sebagai

dispensasi atau pelepasan/pembebasan dari suatu larangan. Terdapat juga pengertian izin dalam arti sempit maupun luas :

- a. Izin dalam arti luas yaitu semua yang menimbulkan akibat kurang lebih sama, yakni bahwa dalam bentuk tertentu diberi perkenaan untuk melakukan sesuatu yang mesti dilarang.
- b. Izin dalam arti sempit yaitu suatu tindakan dilarang, terkecuali diperkenankan, dengan tujuan agar ketentuan-ketentuan yang disangkutkan dengan perkenaan dapat dengan teliti diberikan batas-batas tertentu bagi tiap kasus.

Pada umumnya sistem izin terdiri dari 3 :

1. Larangan.
2. Persetujuan yang merupakan dasar kekecualian (izin).
3. Ketentuan-ketentuan yang berhubungan dengan izin.

### **2.3 Jenis-Jenis Izin Usaha Perdagangan**

Beberapa jenis izin usaha yang dikeluarkan oleh pemerintah yang menyangkut izin usaha perdagangan, yaitu:

1. SIUP (Surat Izin Usaha Perdagangan) Merupakan surat izin yang diberikan oleh menteri atau pejabat yang ditunjuk kepada pengusaha untuk melaksanakan kegiatan usaha di bidang perdagangan dan jasa. Surat izin usaha perdagangan (SIUP) diberikan kepada para pengusaha, baik perseorangan, firma, CV, PT, koperasi, maupun BUMN sesuai Peraturan Menteri perdagangan Republik Indonesia Nomor 46/M-DAG/PER/9/2009. Kewajiban pemegang SIUP yaitu melaporkan kepada kepala kantor wilayah Departemen Perdagangan dan Industri atau kantor Departemen Perdagangan yang menerbitkan SIUP apabila perusahaan tidak melakukan lagi kegiatan perdagangan atau menutup perusahaan disertai dengan pembelian SIUP.
2. SITU (Surat Izin Tempat Usaha) Setiap perusahaan perdagangan yang ada perlu dan harus mengurus SITU, demi keamanan dan kelancaran usahanya. SITU dikeluarkan oleh pemerintah Kota atau Kabupaten sepanjang ketentuan-ketentuan Undang-Undang Gangguan mewajibkannya.

## 2.4 Metodologi Pengembangan Sistem

Metodologi pengembangan sistem adalah suatu kerangka kerja yang digunakan untuk menstrukturkan, merencanakan, dan mengendalikan proses pengembangan suatu sistem informasi. Didalam penulisan skripsi ini menggunakan metodologi *waterfall* untuk digunakan pada pengembangan sistem. Metodologi ini menggunakan pendekatan pengembangan sistem terstruktur yang dilengkapi dengan alat-alat (*tools*) dan teknik-teknik (*techniques*) yang dibutuhkan dalam pengembangan sistem, sehingga hasil akhir dari sistem yang dikembangkan akan didapatkan sistem yang strukturnya didefinisikan dengan baik dan jelas.

### 2.4.1 Perlunya Pengembangan Sistem

Beberapa hal yang menjadi alasan mengapa pentingnya dilakukan perubahan atau pengembangan sistem lama dan perlu diperbaiki atau diganti menurut **Mardi** (2014: 119) adalah sebagai berikut:

1. Sering Menimbulkan Gangguan pada Sistem Lama
  - a. Ketidakberesan sistem lama

Sistem lama sudah melewati usia produktif, tidak sesuai dengan perkembangan zaman, sehingga lebih sering menghambat kelancaran pekerjaan.
  - b. Perkembangan organisasi

Organisasi bisnis yang semakin berkembang menuntut kebutuhan informasi yang semakin tinggi, volume pengolahan data semakin meningkat menyebabkan harus disusun sistem baru.
2. Untuk Meraih Kesempatan

Untuk meraih kesempatan dan peluang usaha, dengan tetap mengandalkan sistem lama dapat merugikan perusahaan sebab kecepatan informasi atau efisiensi waktu sangat menentukan berhasil atau tidaknya strategi dan rencana yang telah disusun untuk meraih kesempatan dan peluang besar. Penggunaan teknologi informasi merupakan kebutuhan mutlak untuk menghadapi pesaing, karena itu jika tidak bisa membeli yang baru maka kembangkan sistem yang sudah ada.

### 3. Adanya Instruksi dari Pimpinan

Menindaklanjuti keputusan direksi, apabila hasil rapat dewan direksi telah menetapkan bahwa segera dilakukan pengembangan sistem yang ada maka manajemen akan membuat perencanaan membuat sistem baru. Dalam penerapannya, suatu sistem yang sudah dikembangkan perlu dilakukan uji coba, yang dilakukan oleh ahli sistem dan setelah itu, sistem baru dapat berjalan normal.

#### 2.4.2 Prinsip Pengembangan Sistem

Beberapa prinsip dalam proses pengembangan sistem adalah sebagai berikut.

- a. Sistem yang dikembangkan adalah untuk manajemen.
- b. Sistem yang dikembangkan adalah investasi modal yang besar.
- c. Sistem yang dikembangkan memerlukan orang terdidik.
- d. Tahapan kerja dan tugas-tugas yang harus dilakukan dalam proses pengembangan sistem.
- e. Proses pengembangan sistem tidak harus urut.
- f. Jangan takut membatalkan proyek.

#### 2.4.3 Siklus Hidup Pengembangan Sistem

Menurut Rosa A.S dan M. Shalahuddin (2016) SDLC atau *Software Development Life Cycle* atau sering disebut juga *System Development Life Cycle* adalah proses pengembangan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya.

Tahapan-tahapan yang ada pada SDLC secara global adalah sebagai berikut :

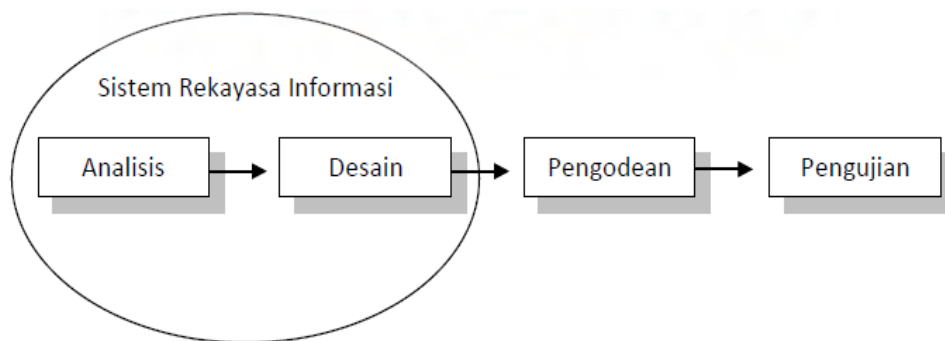
- a. Inisiasi (*Initiation*)  
Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.
- b. Pengembangan konsep sistem (*system concept development*)  
Mendefinisikan lingkup konsep termasuk dokumen lingkup sistem, analisis manfaat biaya, manajemen rencana, dan pembelajaran kemudahan sistem.

- c. Perencanaan (*planning*)  
Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resources*) yang dibutuhkan untuk memperoleh solusi.
- d. Analisis kebutuhan (*requirements analysis*)  
Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan user. Membuat dokumen kebutuhan fungsional.
- e. Desain (*design*)  
Mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.
- f. Pengembangan (*development*)  
Mengonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan; membuat basis data dan mempersiapkan prosedur kasus pengujian; mempersiapkan berkas atau *file* pengujian, pengodean, pengompilasian, memperbaiki dan membersihkan program; peninjauan pengujian.
- g. Integrasi dan pengujian (*integration and test*)  
Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional. Dengan diarahkan oleh staf penjamin kualitas (*quality assurance*) dan *user*. Menghasilkan laporan analisis pengujian.
- h. Implementasi (*implementation*)  
Termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan produksi (lingkungan pada *user*) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.

- i. Operasi dan pemeliharaan (*operations and maintenance*)  
Mendeskripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan produksi (lingkungan pada *user*), termasuk implementasi akhir dan masuk pada proses peninjauan.
- j. Disposisi (*disposition*)  
Mendeskripsikan aktifitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai dengan aktifitas *user*.

#### 2.4.4 Model Waterfall

Menurut Rosa A.S dan M. Shalahuddin (2016) Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*), model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*). Berikut adalah gambar model air terjun :



Gambar 2.1 Ilustrasi model waterfall

- a. Analisis kebutuhan perangkat lunak  
Proses pengumpulan kebutuhan dilakukan secara intensif untuk memesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.
- b. Desain  
Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur

perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c. Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (error) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.


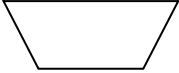



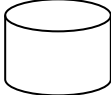

e. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

## 2.5 Alat Pengembangan Sistem

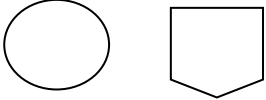

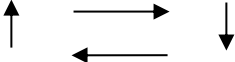
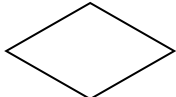
Agar dapat melakukan langkah-langkah sesuai dengan metodologi pengembangan analisis dan desain sistem terstruktur, maka dibutuhkan alat dan teknik untuk melaksanakannya. Adapun alat pengembangan sistem yang digunakan yaitu bagan alir dokumen. Bagan alir dokumen (*Document Flowchart*) untuk menelusuri alur form dan laporan sistem dari satu bagian ke bagian lain baik bagaimana alur form dan laporan diproses, dicatat dan disimpan. Bagan alir dokumen ini menggunakan simbol-simbol seperti pada Tabel 2.1 berikut.

Tabel 2.1 Bagan Alir Dokumen

Simbol	Keterangan
<p><b>Dokumen</b></p> 	<p>Menunjukkan dokumen yang digunakan untuk <i>input</i> dan <i>output</i> baik secara manual maupun komputerisasi.</p>
<p><b>Proses manual</b></p> 	<p>Menunjukkan pekerjaan yang dilakukan secara manual.</p>
<p><b>Proses komputerisasi</b></p> 	<p>Menunjukkan proses dari operasi program komputer.</p>
<p><b>Simpanan</b></p> 	<p>Menunjukkan arsip.</p>
<p><b>Keyboard</b></p> 	<p>Menunjukkan input yang menggunakan <i>keyboard</i>.</p>
<p><b>Hard disk</b></p> 	<p>Media penyimpanan, menggunakan perangkat <i>hard disk</i>.</p>
<p><b>Keterangan</b></p> 	<p>Digunakan untuk memberikan keterangan yang lainnya.</p>

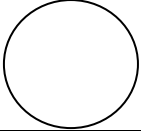
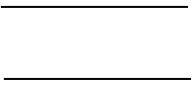
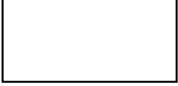
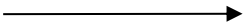


Tabel 2.1 Bagan Alir Dokumen (lanjutan)

Simbol	Keterangan
<p data-bbox="336 356 517 389"><b>Penghubung</b></p> 	<p data-bbox="850 356 1257 611">Simbol yang digunakan untuk menunjukkan sambungan dari bagan alir yang terputus dihalaman yang sama maupun dihalaman yang lain.</p>
<p data-bbox="336 719 504 752"><b>Terminator</b></p> 	<p data-bbox="850 719 1257 804">Digunakan untuk memberikan awal dan akhir suatu proses.</p>
<p data-bbox="336 938 475 972"><b>Garis alir</b></p> 	<p data-bbox="850 938 1257 1023">Digunakan untuk menunjukkan arus dari proses.</p>
<p data-bbox="336 1144 456 1178"><b>Decision</b></p> 	<p data-bbox="850 1144 1257 1288">Digunakan untuk suatu penyeleksian kondisi di dalam program.</p>



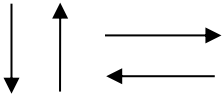

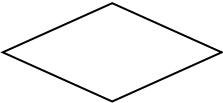
Menurut Rosa A.S dan M. Shalahuddin (2016:69) *Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*). DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur karena pemrograman terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur. Notasi-notasi pada DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut :

Tabel 2.2 Simbol *Data Flow Diagram*



Simbol	Keterangan
<p data-bbox="352 454 576 488"><b>Proses (<i>process</i>)</b></p> 	<p data-bbox="683 342 1299 432">proses atau fungsi yang mentransformasikan data masukan menjadi keluaran</p> <p data-bbox="683 488 1299 633">Catatan: Nama yang diberikan pada sebuah proses biasanya berupa kata kerja</p>
<p data-bbox="352 656 620 745"><b>Penyimpanan data (<i>data store</i>)</b></p> 	<p data-bbox="683 656 1299 853">Notasi ini yang dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data.</p> <p data-bbox="683 909 1299 1055">Catatan: Nama yang diberikan pada sebuah penyimpanan biasanya kata benda</p>
<p data-bbox="352 1070 579 1104"><b><i>External Entity</i></b></p> 	<p data-bbox="683 1070 1299 1267">orang yang berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan</p> <p data-bbox="683 1323 1299 1469">Catatan: Nama yang digunakan pada masukan atau keluaran biasanya berupa kata benda</p>
<p data-bbox="352 1529 620 1619"><b>Aliran data (<i>data flow</i>)</b></p> 	<p data-bbox="683 1529 1299 1675">Data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan atau keluaran.</p> <p data-bbox="683 1731 1299 1877">Catatan : Nama yang digunakan pada aliran data berupa kata benda</p>

Bagan alir program (*Program Flowchart*) merupakan keterangan yang lebih rinci tentang bagaimana setiap langkah program atau prosedur sesungguhnya dilaksanakan. *Flowchart* ini menunjukkan setiap langkah program atau prosedur dalam urutan yang tepat saat terjadi. Untuk simbol-simbol *program flowchart* hampir sama dengan bagan alir dokumen, hanya terdapat perbedaan pada simbol *process*, penyimpanan dan *input-output* data. Adapun simbol-simbol *program flowchart* sebagai berikut :

Tabel 2.3 Simbol Bagan Alir Program

Simbol	Keterangan
<p><b>Input / Output</b></p> 	<p>Simbol <i>input/output</i> digunakan untuk mewakili data <i>input/output</i></p>
<p><b>Proses</b></p> 	<p>Simbol proses (<i>processing symbol</i>) atau simbol pengolah yang digunakan suatu proses.</p>
<p><b>Garis alir</b></p> 	<p>Simbol garis alir (<i>flow lines symbol</i>), digunakan untuk menunjukkan arus dari proses</p>
<p><b>Penghubung</b></p> 	<p>Simbol penghubung (<i>connector symbol</i>), digunakan untuk menunjukkan sambungan dari bagan alir yang terputus dihalaman yang sama / dihalaman yang lain.</p>
<p><b>Keputusan</b></p> 	<p>Simbol keputusan (<i>decision symbol</i>), digunakan untuk suatu penyelesaian kondisi didalam program.</p>

Tabel 2.3 Simbol Bagan Alir Program (lanjutan)

Simbol	Keterangan
<p data-bbox="363 472 619 506"><b>Proses Terdefinisi</b></p> 	<p data-bbox="678 465 1262 607">Simbol proses terdefinisi digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan ketempat lain.</p>
<p data-bbox="363 725 496 759"><b>Terminal</b></p> 	<p data-bbox="678 707 1262 851">Simbol terminal digunakan untuk menunjukkan awal dan akhir dari suatu program.</p>

### 2.5.1 Kamus Data

Menurut Rosa A.S dan M. Shalahuddin (2016:73), Kamus data (*data dictionary*) dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan dan keluaran dapat dipahami secara umum. Kamus data dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur. Kamus data biasanya berisi:

- Nama – nama dari data
- Digunakan pada – merupakan proses-proses yang terkait data
- Deskripsi – merupakan deskripsi data
- Informasi tambahan – seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data

Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut:

Tabel 2.4 Simbol kamus data

Simbol	Keterangan
=	disusun atau terdiri dari
+	dan
[]	baik...atau...
{ }"	n kali diulang/ bernilai banyak
()	data opsional
*...*	batas komentar

### 2.5.2 Basis Data

Menurut Winarno dkk (2013:117) mendefinisikan bahwa “Basis data (database) secara bahasa memiliki arti tempat atau wadah sebuah data. Secara istilah database yaitu kumpulan informasi yang disimpan didalam komputer secara sistematis sehingga dapat diambil datanya atau diolah menggunakan suatu program.” Penyusunan basis data digunakan untuk mengatasi masalah-masalah pada penyusunan data antara lain sebagai berikut :

a. Redudansi dan Inkonsistensi data

Yaitu jika file-file dan program aplikasi diciptakan oleh programmer yang berbeda pada waktu yang berselang cukup panjang, maka ada beberapa bagian data yang mengalami penggandaan (redundancy) pada file yang berbeda. Penyimpanan data yang sama berulang-ulang di beberapa file dapat mengakibatkan juga inkonsistensi atau tidak konsisten, hal ini terjadi satu file record diubah tanpa mengubah file yang sama pada record yang lain.

b. Kesulitan Pengaksesan Data

Kesulitan pengaksesan data timbul bila suatu saat terjadi pengolahan data yang kompleks dan dalam jumlah yang besar, sementara belum tersedia program untuk menunjang hal itu.

c. Isolasi Data Standarisasi

Jika data terbesar beberapa file dalam format yang tidak sama, maka ini akan menyulitkan dalam menulis program aplikasi untuk mengambil dan menyimpan data, maka data harus dalam satu database yang dibuat satu format sehingga mudah dibuat program aplikasi.

d. Banyak Pemakaian (Multiple User)

Dalam rangka mempercepat daya guna sistem dan mendapat respon waktu cepat, beberapa sistem mengizinkan banyak pemakaian untuk meng-update data nantinya, akan dipakai dalam waktu yang berbeda.

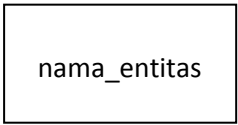
e. Masalah Keamanan (Security)

Setiap merancang suatu database, masalah keamanan atau security harus sangat diperhatikan agar setiap pemakai (User) tidak dapat mengakses semua data.

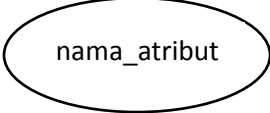
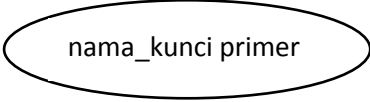
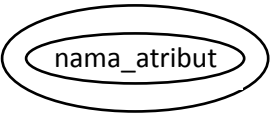
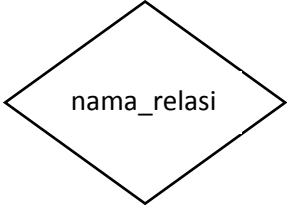
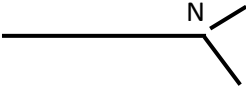
### 2.5.3 Entity Relationship Diagram (ERD)

Menurut Rosa A.S dan M. Shalahuddin (2016:50) Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram* (ERD). ERD digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow's Foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah notasi dari Chen. Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi Chen:

Tabel 2.4 Simbol *Entity Relationship Diagram* (ERD)

Simbol	Keterangan
<p><i>Entity</i></p> 	<p>Merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel</p>

Tabel 2.4 Simbol *Entity Relationship Diagram* (ERD)

Simbol	Keterangan
Atribut  nama_atribut	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
Atribut kunci primer  nama_kunci primer	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama)
Atribut multivalai/ <i>multivalue</i>  nama_atribut	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai dari satu
Relasi  nama_relasi	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
Asosiasi/ <i>association</i> 	Penghubung antar relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian

#### 2.5.4 Relasi Antar Tabel

Relasi adalah hubungan antara tabel yang mempresentasikan hubungan antar objek di dunia nyata. Relasi merupakan hubungan yang terjadi pada suatu tabel dengan lainnya yang mempresentasikan hubungan antar objek di dunia nyata dan berfungsi untuk mengatur operasi suatu database. Database dengan struktur data hubungan dapat digambarkan dalam bentuk dua dimensi. Kolom dari tabel menunjukkan atribut dari file. Atribut ini menunjukkan item data atau field. Kumpulan nilai dari field atau item data disebut dengan istilah domain. Masing-masing baris dari record di dalam tabel disebut dengan istilah tuple. Suatu tuple (record) yang mempunyai dua domain disebut dengan 2 tuple. Suatu tuple yang mempunyai 3 domain disebut dengan 3-tuple dan seterusnya. Tiap-tiap tuple atau record ini dapat mempunyai suatu kunci yang unik dengan cara mana tuple ini dapat diidentifikasi. Field yang menjadi kunci yang unik ini disebut dengan field kunci (key field). Database didukung oleh beberapa komponen antara lain :

a. Tabel

Pengelompokan secara logika dari sistem yang berhubungan, table- table dibentuk dari baris- baris dan kolom -kolom, baris-baris disebut sebagai record -record, sedangkan kolom-kolom disebut sebagai field.

b. Record

Suatu record mengandung informasi dari suatu isian tunggal dari suatu table.

c. Field

Suatu record dibentuk oleh banyak field dalam satu record mengandung suatu potongan tunggal dari informasi tentang record.

d. Primary key

Primary key adalah suatu field atau kombinasi dari field-field yang mana nilainya bersifat untuk tiap baris atau record dalam table.

e. Foreign key

Atribut yang menjadi atribut penghubung antara satu file dengan file lainnya.



## 2.6 Perangkat Lunak Pendukung

Perangkat lunak pendukung yaitu *software* yang akan digunakan untuk menganalisis dan merancang sistem yang diusulkan, diantaranya seperti berikut.

### 2.6.1 PHP

Sidik (2014:4) menjelaskan, PHP merupakan secara umum dikenal sebagai bahasa pemrograman *script* yang membuat dokumen HTML secara *on the fly* yang dieksekusi di *server web*, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML.

Langkah- langkah menggunakan PHP :

- File **php** yang akan dijalankan harus berada di dalam folder **D:\xampp\htdocs**, baik itu di dalam folder tersebut, atau folder-folder dibawahnya. Untuk menjalankan di dalam browser, anda tinggal mengganti alamat **D:\xampp\htdocs** menjadi **localhost**. Namun Jika file php tersebut berada di luar folder **htdocs**, maka web server tidak bisa mengaksesnya.
- Setiap file **PHP** harus ditulis menggunakan akhiran **.php** (walaupun hal ini bisa dirubah pada settingan modul php).
- Penamaan file PHP, harus tanpa spasi dan sebaiknya menggunakan huruf kecil. Anda dapat menggunakan **underscore** (**\_**) sebagai pengganti spasi.

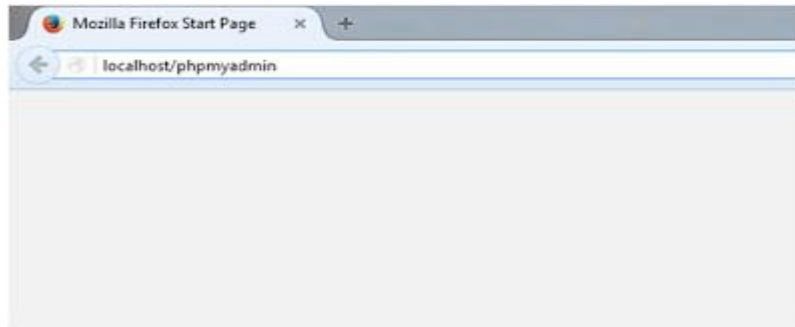
### 2.6.2 MySQL

Kadir (2013:15), “MySQL adalah nama *database server*. *Database server* adalah *server* yang berfungsi untuk menangani *database*. *Database* adalah suatu pengorganisasian data dengan tujuan memudahkan penyimpanan dan pengaksesan data. Dengan menggunakan MySQL, kita bisa menyimpan data dan kemudian data bisa diakses dengan cara yang mudah dan cepat.”

Langkah – langkah menggunakan MySQL

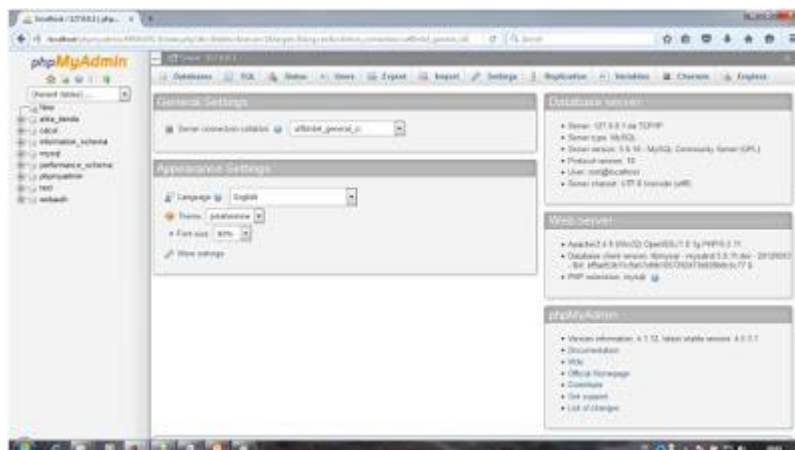
1. Langkah pertama, tentunya kita harus menjalankan Xampp terlebih dulu, lalu aktifkan web server (Apache) dan database MySql nya.

2. Lalu jika sudah berhasil menjalankan xampp nya, langkah selanjutnya kita buka browser kita, boleh mozilla firefox atau Google chrome (sesuai keinginan anda).
3. Kemudian ketikkan `http://localhost/phpmyadmin/` kemudian tekan enter. Gambarnya seperti dibawah ini.



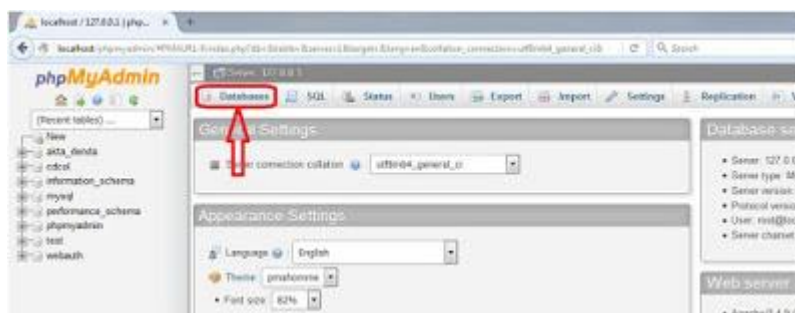
Gambar 2.2 localhost

4. Jika sudah, maka anda akan masuk ke dalam halaman PhpMyadmin tampilannya seperti dibawah ini



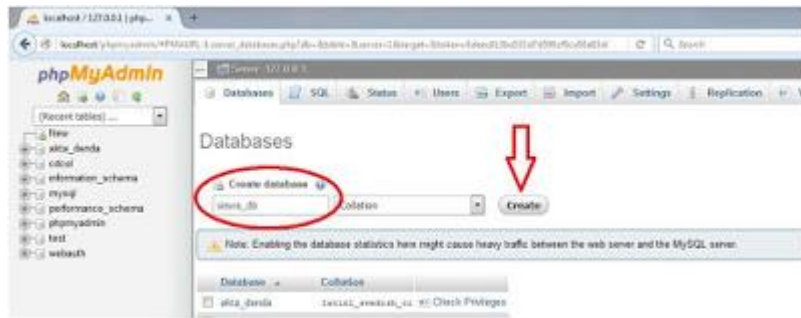
Gambar 2.3 Tampilan PhpMyadmin

5. Kemudian kita klik database, dan isikan nama database sesuai dengan keinginan anda, disini saya akan menggunakan nama siswa\_db.



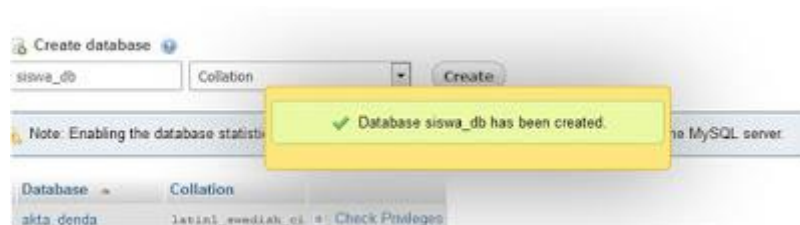
Gambar 2.4 Tampilan PhpMyadmin 2

6. Setelah mengisi nama database selanjutnya klik create.



Gambar 2.5 Membuat database

7. Jika sudah muncul tampilan seperti dibawah ini, berarti anda telah berhasil membuat database.



Gambar 2.6 Tampilan database berhasil dibuat

8. Berikut langkah – langkah dalam membuat *table*
- Ketikan nama *table* pada kolom *create new table* pada *database* yang telah dibuat sebelumnya seperti gambar 2.7 dibawah ini



Gambar 2.7 membuat tabel

- Setelah itu isikan *field* sesuai dengan kebutuhan, lalu klik *go*,
- Kemudian isikan nama *field*, tipe data, lenght, primary key dengan sesuai, lalu klik *button* simpan.

### **2.6.3 Apache**

Rulianto Kurniawan (2010:9), Apache web server atau server HTTP Apache adalah server web yang dapat dijalankan di banyak sistem operasi seperti Linux, Unix, Microsoft Windows dan Novell Netware serta platform lainnya yang berguna untuk melayani dan mengfungsikan situs web. Apache merupakan perangkat lunak sumber terbuka (open source) yang dikembangkan oleh komunitas terbuka.