

BAB II

TINJAUAN PUSTAKA

2.1 Pengertian Sistem Informasi

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran tertentu [3]. Informasi adalah data yang telah diolah menjadi suatu bentuk yang penting bagi si penerima dan mempunyai nilai nyata atau yang dapat dirasakan dalam keputusan-keputusan yang sekarang atau keputusan-keputusan yang akan datang. Administrasi keuangan adalah langkah pengolahan keuangan sekolah yang dimulai dari penerimaan hingga bagaimana keuangan yang digunakan dipertanggungjawabkan secara obyektif. Langkah tersebut sangat penting untuk diperhatikan, sebab pembiayaan merupakan sarana vital yang akan mempengaruhi keberlangsungan proses pendidikan di suatu sekolah [4]. Dengan demikian dapat disimpulkan bahwa sistem informasi administrasi keuangan merupakan data yang telah diolah menjadi suatu bentuk yang penting dalam pengolahan data keuangan sekolah agar lebih efektif dan efisien.

2.2 Perancangan Sistem

2.2.1 Database

Database adalah sekumpulan file data yang satu sama lainnya saling berhubungan yang diorganisasi sedemikian rupa sehingga memudahkan untuk mendapatkan dan memproses data tersebut. Lingkungan sistem database menekankan pada data yang tidak tergantung (*independent*) pada aplikasi yang akan menggunakan data tersebut [4]. Sistem basis data adalah terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media penyimpanan data.

Database adalah proses yang akan mendukung operasional dan tujuan perusahaan. Tujuan basis data adalah:

- Menggambarkan relasi data antar data yang dibutuhkan oleh aplikasi dan *user view*.
- Menyediakan model data yang mendukung seluruh transaksi yang diperlukan.
- Menspesifikasikan desain dengan struktur yang sesuai dengan kebutuhan sistem.

Ada beberapa pendekatan yang dapat digunakan dalam basis data yaitu:

- *Top-down*

Diawali dengan membuat data model. Pendekatan top-down dapat diilustrasikan menggunakan *entity-relationship (ER)* model yang *high level*, kemudian mengidentifikasi *entity*, dan *relationship* antar *entity* organisai. Pendekatan ini sesuai bagi basis data yang kompleks.

- *Bottom-up*

Dimulai dari level dasar attribute (*property entity* dan *relationship*), menganalisa hubungan antar-attribute, mengelompokkannya dalam suatu relasi yang menggambarkan tipe *entity* dan *relasi antar-entity* [6].

Database Planning (Perencanaan basis data) merupakan aktivitas manajemen untuk merealisasi tahapan database *Application Lifecycle* secara efektif dan efisien. perencanaan basisdata mencakup cara pengumpulan data, format data, dokumentasi yang diperlukan, cara membuat desain dan implementasi.

2.2.2 Aturan merancang database

Dalam pembuatan aplikasi perlu memperhatikan rancangan database agar aplikasi yang dibuat dapat berjalan sesuai konsep yang direncanakan, Perancangan database yang baik, hendaknya mengikuti aturan seperti berikut .

- Tidak boleh ada dua data atau lebih yang sama atau *redundancy* data.
- Setiap table harus memiliki kolom yang unik yang disebut *primary key*, dimana kolom tersebut tidak dapat berisi data yang sama.
- Besar dan ukuran table dibuat seminimal mungkin dengan pengaturan tipe data yang tepat.
- Pembuatan table dan kolom pada table hendaknya disesuaikan dengan kebutuhan aplikasi.

2.2.3 HTML (*Hyper Text Markup Language*)

HTML(*Hyper Text Markup Language*) adalah format standar yang digunakan Untuk membuat halaman web.file HTML berupa file teks yang terdiri dari 2 bagian: Content (isi) yaitu berupa teks yang akan ditampilkan oleh browser dan markup atau tags yang menjelaskan bagaimana teks tersebut diinterpretasi oleh browser. Tag tagyang digunakan HTML mempunyai format sebagai berikut:<tag pembuka>teks </tag penutup> [9].

2.2.4 PHP (*Hypertext Pre-processor*)

PHP merupakan bahasa pemrograman yang digunakan untuk membuat website dinamis dan interaktif. Dinamis artinya, website tersebut bisa berubah-ruhahtampilan dan kontennya sesuai kondisi tertentu [10]. *PHP* adalah penerjemahan baris kode yang bisa dibaca atau dimengerti oleh komputer karena *PHP* bisa diletakkan pada *scriptHTML* atau sebaliknya.*PHP* dikhususkan untuk pengembangan web dinamis [11].

Agar dapat menjalankan *PHP* harus menyediakan perangkat lunak berikut :

- *Web server (Apache, IIS, Personal Web Server/PWS)*
- *PHP Server*
- *Database Server (MySQL, Itebase, MS SQL)*

2.2.5 Fungsi *PHP*

- Fungsi *PHP* dalam pemrograman web diantaranya sebagai berikut :
- Menghasilkan halaman dinamis pada website.
- Membuat, membuka, menulis, membaca, menghapus dan menutup file pada *server*.
- Memproses data yang dikirim dari form.
- Mengirim dan mengakses *cookie*.
- Modifikasi data pada *database*.
- Mengontrol akses *user*.
- Mengenkripsi data.

2.2.6 XAMPP

XAMPP merupakan software stack untuk pengembangan aplikasi berbasis web yang paling populer. *XAMPP* awalnya adalah *MySQL*, tetapi sekarang bisa berarti juga *MariaDB*, karena paket *XAMPP* dari *Apache Friends* menggunakan *MariaDB* sebagai bagian dari *software Stacknya* [10].

PHPMysqlAdmin adalah alat yang dibuat dengan *PHP* untuk administrasi database *MySQL*, seperti database, tabel, indeks, trigger, user, hak akses, dan lain lain

2.2.7 MySQL

MySQL merupakan *server* yang melayani database. Untuk membuat dan mengolah database, kita dapat mempelajari pemrograman khusus yang disebut *query*

(perintah) *SQL*. Database sendiri dibutuhkan jika kita ingin menginput data dari user menggunakan *form HTML* untuk kemudian diolah PHP agar bisa disimpan ke dalam database *MySQL*. *MySQL* adalah sebuah database server, dapat juga berperan sebagai client sehingga sering disebut database client/server, yang open source dengan kemampuan dapat berjalan baik di *OS (Operating System)* maupun dengan platform windows maupun linux. Kelebihan *MySQL* adalah mampu menerima query yang bertumpuk dalam satu permintaan atau yang disebut Multi-Treading [12].

2.2.8 CSS (*Cascading Style Sheet*)

CSS (Cascading Style Sheet) merupakan sebuah standar penggunaan dan pembuatan *style* pada suatu dokumen halaman web.

Cara yang digunakan untuk meletakkan (*embedding*) CSS kedalam file *HTML*.

```
<Linkrel="stylesheet" type="text/css" href="filr.css">
```

Tag ini berguna untuk mengimpor style dari file eksternal ke dalam *HTML*. Agar file tersebut berhasil di-load, tag ini sebaiknya diletakkan antara tag *<head></head>*

Keuntungan menggunakan CSS antara lain:

- *Rich styling*, CSS menawarkan lebih banyak alternatif style yang bisa ditambahkan pada webpage daripada HTML.
- *Ease of use*, CSS dapat mengurangi beban seorang webmaster dalam coding dengan meletakkan pendefinisian style di satu tempat dari pada tersebar dalam tag-tag diseluruh dokumen.
- *Using style for multiple pages*, informasi style suatu webpage dapat disimpan terpisah dari file HTML dan diaplikasikan ke banyak halaman web sehingga menjaga kesatuan semua halaman web.
- *Cascading*, CSS menangani konflik yang terjadi bila ada sebuah style rule yang diaplikasikan pada elemen yang sudah memiliki style tertentu.
- *Compact file size*, setiap halaman web dapat berkurang ukuran filenya karena informasi mengenai style disimpan di tempat lain [7].

2.2.9 Web Server

Web server merupakan perangkat lunak yang berfungsi untuk menerima permintaan melalui *protocol HTTP* atau *HTTPS* dari *client* kemudian mengirimkan kembali dalam bentuk halaman-halaman web. Contoh yang termasuk *web server* adalah *apache*. Dalam penggunaannya, biasanya sudah jadi satu paket dengan PHP dan *MySQL*. Contoh paket yang sudah berisi *apache*, *PHP* dan *MySQL* diantaranya *Xampp* dan *Appserv*. Server merupakan computer khusus dengan kecepatan dan kapasitas penyimpanan di atas komputer biasa digunakan sebagai tempat penyimpanan file-file website [12].

2.2.10 Website

Website adalah suatu aplikasi yang berisikan dokumen-dokumen yang tersimpan dalam *server* serta untuk mengaksesnya dibutuhkan perangkat lunak yang disebut browser. Selain itu website dapat didefinisikan sebagai berikut: Web adalah sistem dengan informasi yang disajikan dalam bentuk teks, gambar, suara, dan lainnya yang tersimpan dalam sebuah *server web* internet yang disajikan dalam bentuk *hypertext*. *Website* adalah salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, animasi, video) didalamnya yang menggunakan protokol *HTTP (Hypertext Transfer Protocol)* dan untuk mengaksesnya menggunakan perangkat lunak yang disebut browser.

Jenis-jenis web berdasarkan sifat atau style-nya sebagai berikut:

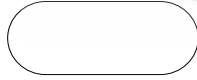
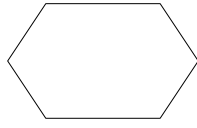

1. Website Dinamis Merupakan sebuah website yang menyediakan content atau isi yang selalu berubah ubah setiap saat. Bahasa pemrograman yang digunakan antara lain *PHP*, *ASP*, *.NET* dan memanfaatkan database *MySQL* atau *MS SQL*. Misalnya website www.artikel-it.com, www.detik.com, dan lain-lain.
2. Website Statis Website yang content-nya sangat jarang diubah. Bahasa pemrograman yang digunakan adalah *HTML* dan belum memanfaatkan database. Misalnya: web profile organisasi dan lain-lain.

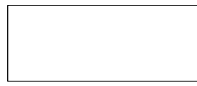
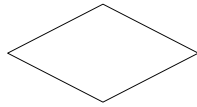

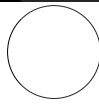




3. Fungsi Web Berdasarkan pada fungsinya, website terbagi atas:
 - a. *Personal Website* yang berisi informasi pribadi seseorang.
 - b. *Commercial Website* yang dimiliki oleh sebuah perusahaan yang bersifat bisnis.
 - c. *Government Website* yang dimiliki oleh instansi pemerintah, pendidikan yang bertujuan memberikan pelayanan kepada pengguna.
 - d. *Non-Profit Organization Website* Dimiliki oleh organisasi yang bersifat non-profit atau tidak bersifat bisnis [14].

2.4.4 Flowchart

Flowchart merupakan sebuah bagian dengan simbol (sandi) tertentu yang menjelaskan dan menggambarkan langkah-langkah proses secara mendetail, dan hubungan antara proses (metode) dengan proses lainnya pada suatu program. Flowchart adalah representasi secara simbolik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan flowchart akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu flowchart juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek [16].

Tabel 1.1 Simbol Flowchart

No	Simbol	Fungsi
1		<i>Terminal</i> yaitu awal atau akhir sebuah program
2		<i>Preparation</i> atau persiapan yaitu pemberian harga awal dan deklarasi variabel dan lain-lain
3		<i>Input / output</i> yaitu proses pemasukan atau pengeluaran data




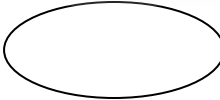

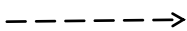
4		<i>Process</i> yaitu proses pengolahan data yang dilakukan oleh computer
5		<i>Decision</i> yaitu proses pengambilan keputusan untuk memilih satu diantara dua alternative
6		<i>Subrutine</i> yaitu kumpulan langkah-langkah
7		<i>On Page Connector</i> yaitu tanda gabung dalam halaman yang sama
8		<i>Off Page Connector</i> yaitu tanda gabung lain halaman
9		<i>Arrow</i> yaitu penunjuk arah aliran program
10		<i>Document</i> , menyatakan simbol untuk data yang berbentuk kertas maupun untuk informasi
11		<i>Simbol</i> untuk output, yang ditunjukkan ke suatu device, seperti printer

2.3 Basis Data

2.3.1 Usecase Diagram

Usecase Diagram adalah diagram yang menggambarkan interaksi antara pengguna dengan sistem informasi yang dibuat [12]. *Usecase diagram* Merupakan gambaran dari fungsionalitas yang diharapkan dari sebuah sistem, dan merepresentasikan sebuah interaksi antara aktor dan sistem. Didalam *use case* terdapat aktor yang merupakan sebuah gambaran entitas dari manusia atau sebuah sistem yang melakukan pekerjaan di sistem [13].

Tabel 1.2 Simbol Usecase Diagram

Symbol	Penjelasan
 ACTOR	<i>Actor</i> Mendefinisikan seperangkat peranan yang user sistem dapat diperankan ketika berinteraksi dengan use case.
	<i>Association</i> Menggambarkan interaksi antara actor dan use case.
	<i>Generalization</i> Relasi antar use case, dimana salah satunya dalam bentuk yang lebih umum dari yang lain.
	<i>Use Case</i> Sebuah deskripsi dari seperangkat aksi berurutan yang ditampilkan pada sebuah sistem.
	<i>System</i> Tempat seluruh aktivitas-aktivitas sistem yang sedang berjalan.
	<i>Dependency</i> Untuk menggambarkan ketergantungan sebuah use

	case dengan use case lainnya.
<pre><<Include>> - - - - -></pre>	<p><i>Include</i></p> <p>Menggambarkan bahwa keseluruhan dari sebuah use case merupakan fungsionalitas use case lainnya.</p>
<pre><<Extend>> - - - - -></pre>	<p><i>Extend</i></p> <p>Menggambarkan hubungan antara use case dimana sebuah use case merupakan fungsionalitas use case lainnya apabila kondisi tertentu dipenuhi.</p>

2.3.2 UML (*Unified Modelling Language*)

Unified Modeling Language merupakan salah satu metode pemodelan visual yang digunakan dalam perancangan dan pembuatan sebuah software yang berorientasikan pada objek. UML merupakan sebuah standar penulisan atau semacam blue print dimana didalamnya termasuk sebuah bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik [14].

UML menyediakan banyak sekali diagram yang diperlukan untuk menjelaskan sistem yang sedang dikembangkan, baik dari aspek statis maupun dinamisnya. Salah satu diagram penting yang digunakan untuk mengilustrasikan kebutuhan (*requirements*) dari sistem adalah *use case (UC)* diagram, yang menjelaskan secara visual konteks dari interaksi antara aktor dengan sistem [15].

2.3.3 Aktor

Aktor mempresentasikan orang atau obyek yang akan berinteraksi dengan aplikasi. Aplikasi ini mendefinisikan actor yang akan berinteraksi dengan aplikasi, yaitu Admin, Kepala Sekolah, Siswa.

Tabel 1.3 Aktor

Aktor	Defenisi
Admin	Admin dapat melakukan proses pembayaran dan melakukan manajemen laporan pembayaran.
Kepalasekolah	Kepala Sekolah dapat melihat laporan pembayaran.
Siswa	Siswa dapat mencetak laporan pembayaran spp dan mengupload bukti pembayaran jika pembayaran melalui rekening sekolah.