

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Kajian Pustaka**

Penelitian tentang program sistem pendukung keputusan ini pernah dilakukan oleh Jamaliah (2011) dalam skripsi dengan judul “Sistem Informasi Akademik Berbasis *Client Server*”. Pada penelitian tersebut membahas bagaimana meningkatkan performance basis data, menjamin ketersediaan data dan *respon time* yang tinggi sehingga sekolah dapat memperoleh data yang diinginkan dengan cepat, efisien, akurat dan optimal.

Adapun penelitian tentang pemilihan perguruan tinggi juga pernah dilakukan Lidya Wulandari dan Delviani (2011) dalam skripsi dengan judul oleh “Sistem Informasi Akademik Berbasis Berbasis Desktop Pada SMA Bina Cipta Palembang”. Pada penelitian tersebut membahas bagaimana mempermudah, mempercepat dan mengurangi tingkat kesalahan dalam pengelolaan data siswa, data guru, nilai siswa dan laporan data-data siswa, guru serta nilai yang akan disajikan ke Kepala Sekolah pada SMA Bina Cipta Palembang.

#### **2.2 Dasar Teori**

##### **2.2.1 Sistem Informasi Akademik**

Sistem informasi akademik adalah sebuah perangkat lunak yang mengolah proses akademik sekolah secara komputerisasi dengan tujuan agar proses pengolahan data akademik lebih cepat, tepat dan aman.

##### **2.2.1.1 Definisi Sistem**

Menurut Jogianto (2005: 2) “Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata, seperti tempat, benda dan orang-orang yang ada dan terjadi.”

Sedangkan menurut Oetomo dkk (2006: 41) “Sistem merupakan sekumpulan elemen yang saling berhubungan satu dengan lainnya dan membentuk satu kesatuan untuk menyelesaikan satu tujuan yang spesifik atau menjalankan seperangkat fungsi.”

Berdasarkan kutipan Jogianto dan Oetomo, dapat disimpulkan bahwa sistem adalah kumpulan dari elemen yang saling berinteraksi antara satu sama lain untuk mendapat tujuan tertentu.

### **2.2.1.2 Definisi Informasi**

Menurut Jogiyanto HM., (1999: 692) “Informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (*event*) yang nyata (*fact*) dan digunakan untuk pengambilan keputusan.”

Sedangkan menurut Anton M. Meliono (1990: 331) “Informasi adalah data yang telah diproses untuk suatu tujuan tertentu. Tujuan tersebut adalah untuk menghasilkan sebuah keputusan.”

Berdasarkan kutipan Jogiyanto dan Anton M. Meliono, dapat disimpulkan bahwa informasi adalah data yang berhasil diproses agar mendapatkan sebuah keputusan berdasarkan data tersebut.

### **2.2.2 Web Dan Web Browser**

*Web* adalah suatu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, suara, animasi, dan video) di dalamnya digunakan protocol HTTP (*HyperText Transfer Protocol*) dan untuk mengaksesnya menggunakan perangkat lunak yang disebut dengan *web browser* (Rudyanto Arief, M, 2011).

Sedangkan *web browser* adalah program yang berfungsi untuk menampilkan dokumen-dokumen *web* dalam format HTML (*HyperText Markup Language*).

### **2.2.3 Web Server**

*Web server* adalah program aplikasi yang memiliki fungsi sebagai tempat menyimpan dokumen-dokumen *web* (Rudyanto Arief, M, 2011). Semua dokumen *web*

baik yang ditulis menggunakan *client side scripting* maupun *server side scripting* tersimpan dalam direktori utama *web server (document root)*. Salah satu contoh *web server* adalah *apache* yang mendukung PHP. Biasanya *web server* ini sudah tersedia secara *bundle* dengan *database server* dalam satu aplikasi untuk membantu proses *develop* program.

Salah satu program yang sering digunakan dalam proses *develop web* PHP dengan menggunakan *web server* apache adalah XAMPP. XAMPP merupakan paket PHP yang berbasis *open source* yang dikembangkan oleh komunitas *open source*, (Bunafit Nugroho, 2009). Dengan menggunakan XAMPP sudah ada paket lengkap untuk *dovelop* sebuah *website*. Berikut adalah beberapa paket yang telah disediakan:

1. Apache
2. MySql
3. PHP
4. FileZilla FTP Server
5. Php My Admin

Dengan adanya dukungan dari paket-paker tersebut, proses *develop* web akan lebih mudah, didukung dengan banyaknya forum yang membahas tentang paket tersebut, sehingga pengguna dapat mencari solusi sendiri ketika menemukan kesulitan. Program ini juga aman dan legal untuk di pakai karena berbasis *open source*.

#### 2.2.4 CSS

CSS adalah singkatan dari *Cascading Style Sheet* yang berguna untuk mengatur tampilan dokumen HTML dari jarak antar baris, warna dan format *border* bahkan penampilan gambar (jayan 2010, 2). CSS dikembangkan oleh W3C, organisasi yang mengembangkan teknologi internet. Tujuannya tidak lain untuk mempermudah proses penataan halaman *web*.

CSS hanyalah berupa kumpulan *script* yang tujuamya bukan menggantikan HTML, melainkan pelengkap agar dokumen HTML bisa tampil lebih cantik dan

dinamis. Sejak awal ditemukannya CSS pada awal dekade 90an, CSS terus dikembangkan dan diserap oleh *web developer*. Kode CSS bersifat lintas *platform*, yang berarti *script* ini dapat dibaca oleh berbagai macam sistem operasi dan *browser*.

Dengan berkembangnya teknologi CSS saat ini munculah alat bantu untuk membuat tampilan *web* dengan lebih mudah dan cepat, yaitu dengan *Twitter Bootstrap*. *Twitter Bootstrap* adalah sebuah alat bantu untuk membuat *website* yang dapat mempercepat seorang pengembang *website* ataupun pendesain halaman *website* (Wikipedia.org).

*Twitter bootstrap* dibangun dengan teknologi HTML dan CSS yang dapat membangun sebuah *layout* halaman *website*, *table*, tombol, navigasi, dan komponen lainnya didalam *website* hanya dengan mengambil fungsi CSS dalam berkas HTML yang telah didefinisikan. Selain itu juga terdapat komponen-komponen lainnya yang dibangun dengan *java script*.

### 2.2.5 HTML

HTML atau (*HyperText Markup Language*) merupakan salah satu format yang digunakan dalam pembuatan dokumen dan aplikasi yang berjalan di halaman *web*. Dokumen ini dikenal sebagai *web page*. Dokumen HTML merupakan dokumen yang disajikan pada *web browser*. (Rudyanto Arief, M, 2011).

```
<HTML>

</HTML>
```

**Gambar 2.1** Struktur HTML

#### 2.2.5.1 Pengenalan kode HTML

Masing-masing baris diatas disebut tag. Tag adalah kode yang digunakan untuk *me-markup* (memoles) teks ASCII menjadi file HTML. Setiap tag diapit dengan tanda kurung runcing. Ada tag pembuka yaitu `<HTML>` dab ada tag penutup yaitu `</HTML>`

yang ditandai dengan tanda *slash* di depan awal tulisannya. Tag diatas memberikan kaidah bahwa yang akan ditulis diantara kedua tag tersebut adalah dari dokumen HTML.

### 2.2.5.2 Definisi Elemen dan *Tag* HTML

Sebuah dokumen HTML disusun oleh beberapa elemen atau lebih dikenal dengan komponen-komponen dasar. Elemen dapat berupa teks murni, atau bukan teks atau bukan keduanya. Elemen atau komponen tersebut misal *head*, *body*, *paragraph*, *list*, dan lain-lain.

Untuk menandai sebuah elemen dalam suatu dokumen HTML digunakan tag. Tag HTML terdiri dari sebuah [*(<)(nama tag)(>)*] contoh *<h1>*, *tag* pada umumnya berpasangan (misalnya *<h1>* dengan *</h1>*), tanda / pada *tag* pasangan memberikan tanda bahwa *tag* tersebut merupakan pembatas akhir elemen yang dibuka oleh *tag* awal. Jadi pada umumnya penulisan *tag* adalah *<nama tag>....</nama tag>* selain itu dalam penamaan *tag* tidak menganut *case sensitive*

Elemen dasar yang harus dimiliki dalam pembuatan dokumen HTML adalah sebagai berikut:

```
<html>
<head>
  ....informasi dokumen...
</head>
<body>
  ...informasi yang ditampilkan pada web browser....
</body>
</html>
```

**Gambar 2. 2** Elemen HTML

### 2.2.6 PHP

PHP (*HyperText Preprocessor*) adalah Bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat suatu halaman *web* yang dinamis. Karena PHP merupakan *server-side scripting* maka sintaks dan perintah-perintah PHP dieksekusi di *server* kemudian hasilnya dikirim ke *browser* dalam format HTML. Dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh *user* sehingga keamanan dalam *web* lebih terjamin. PHP dirancang untuk membentuk sebuah halaman *web* yang dinamis, yaitu halaman *web* yang mampu membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman *web*. (Rudyanto Arief, M, 2011).

PHP termasuk dalam *Open Source Product*, sehingga *source code* PHP dapat diubah dan distribusikan secara bebas. Versi terbaru PHP dapat diunduh secara gratis disitus resmi PHP: <http://www.php.net>. PHP juga dapat berjalan diberbagai *web server* seperti ISS (*Internet Information Server*), PWS (*Personal Web Server*), Apache, Xitami. PHP juga mampu lintas *platform*. Artinya PHP dapat berjalan dibanyak sistem operasi yang beredar saat ini, diantara: Sistem Operasi Microsoft Windows (semua versi), Linux, MacOS, Solaris. PHP dapat dibangun sebagai modul pada *web server* Apache dan sebagai *binary* yang dapat berjalan sebagai CGI (*Common Gateway Interface*). PHP dapat mengirim HTML *header*, dapat mengatur *cookies*, mengatur *authentication* dan *redirect users*.

Salah satu keunggulan yang dimiliki oleh PHP adalah kemampuannya untuk melakukan koneksi ke berbagai macam *software* sistem manajemen basis data / *Database Manajemen System* (DBMS), sehingga dapat menciptakan halaman *web* yang dinamis. PHP mempunyai koneksifitas terhadap beberapa DBMS antara lain Oracle, Sybase, mSQL, MySQL, Microsoft SQL Server, Solid, PostgreSQL, Adabas, FilePro, Velocis, dBase, Unix dbm, dan tidak terkecuali semua database berinterface ODBC. PHP juga memiliki integrasi dengan beberapa *library* eksternal yang dapat membuat pengguna dapat melakukan segalanya dari dokumen PDF hingga mem-*parse* XML.

PHP juga mendukung komunikasi dengan layanan lain melalui IMAP, SNMP, NNTP, POP3 atau bahkan HTTP. Hampir seluruh aplikasi berbasis *web* dapat dibuat dengan PHP. Namun kekuatan utama adalah koneksifitas basis data dengan *web*. Dengan kemampuan ini kita akan mempunyai suatu sistem basis data yang dapat diakses dari *web*.

### 2.2.7 JavaScript

JavaScript adalah bahasa *scripting* yang populer di internet dan dapat bekerja di sebagian besar *browser* populer seperti Internet Explorer (IE), Mozilla FireFox, Netscape dan Opera. Kode JavaScript dapat disisipkan dalam halaman *web* menggunakan *tag SCRIPT*.

Beberapa hal tentang JavaScript:

1. JavaScript didesain untuk menambah interaktif suatu *web*.
2. JavaScript merupakan bahasa *scripting*.
3. Bahasa *scripting* merupakan bahasa pemrograman yang ringan.
4. JavaScript berisi baris kode yang dijalankan di komputer (*Web Browser*).
5. JavaScript biasanya disisipkan (*embedded*) dalam HTML.
6. JavaScript adalah bahasa *interpreter* (yang berarti *script* dieksekusi tanpa proses kompilasi).
7. Setiap orang dapat menggunakan JavaScript tanpa membayar lisensi.

(Sunyoto, Andi, 2007).

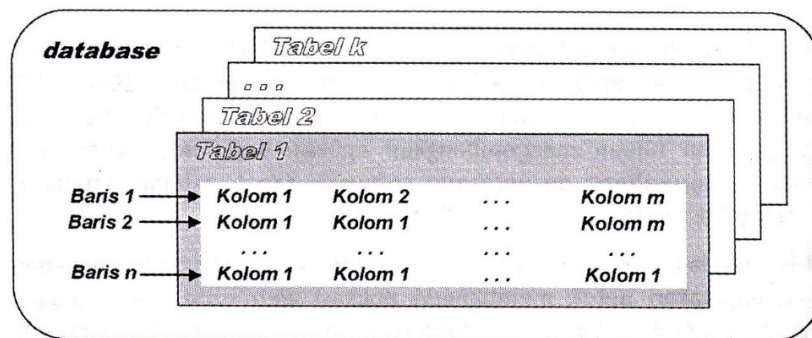
### 2.2.8 MySql

MySql (*My Structured Query Language*) atau yang sering disebut dengan *Mail-Sequel* adalah sebuah program pembuat *database* atau yang sering disebut dengan DBMS (*Database Management System*), sifat dari DBMS adalah *open source*. (Bunafit Nugroho, 2009).

MySQL termasuk RDBMS (*Relational Database Management System*). Itulah sebabnya istilah *table*, baris, dan kolom digunakan pada MySQL. Pada MySQL,

sebuah *database* mengantung satu atau sejumlah tabel. Tabel terdiri atas sejumlah kolom dan baris, dimana setiap kolom berisi sekumpulan data yang memiliki tipe yang sejenis, dan baris merupakan sekumpulan data yang berkaitan dan membentuk informasi. Kolom biasanya juga disebut sebagai *field* dan informasi yang disimpan dalam setiap baris disebut *record*.

### Konsep *database*

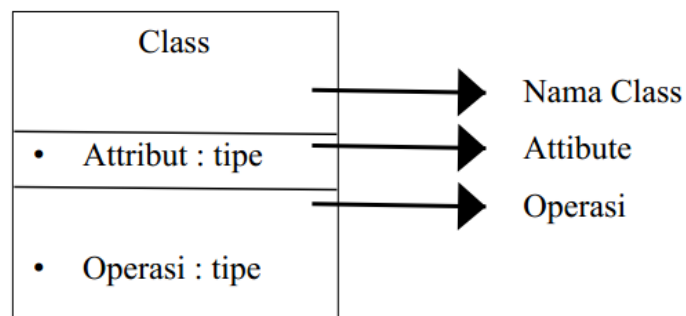


**Gambar 2.3** Konsep *Database*

### 2.2.9 Class Diagram

*Class* diagram merupakan salah satu diagram utama dari UML untuk menggambarkan sebuah *class* atau *blueprint object* pada sebuah sistem. Analisis pembentukan *class* diagram merupakan aktivitas inti yang sangat mempengaruhi perancangan *software* yang dirancang ke tahap *coding*. Berikut merupakan gambar notasi dari *class* diagram:





**Gambar 2. 4** Notasi *Class* Diagram

Berikut merupakan penjelasan dari notasi *class* diagram diatas:

1. *Class Name*

Berfungsi untuk membedakan antara satu kelas dengan kelas yang lain.

Contoh: Manusia, Dosen, Karyawan, Mahasiswa

2. *Attribute*

Berfungsi untuk menyimpan *state* atau apa yang dimiliki oleh sebuah *object*.

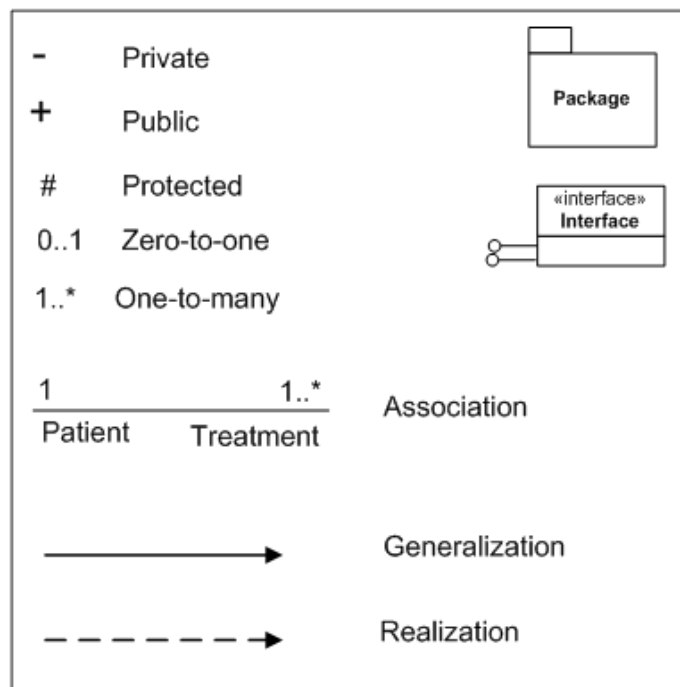
Contoh: NIM, nama, alamat, tempat lahir, tanggal lahir

3. *Method* atau Operasi

Method merupakan kumpulan program yang memiliki nama. Method juga berfungsi untuk memecah program menjadi bagian-bagian yang kecil agar menjadi lebih kompleks sehingga dapat digunakan secara berulang-ulang.

Contoh: ambilRasa, kupasKulit

Gambar di bawah menunjukkan berbagai hubungan antar tiap *class* diagram:



**Gambar 2. 5** Hubungan Antar *Class* Diagram

	Agregasi ( <i>Aggregation</i> )	
	Kunci Gabungan ( <i>Composite Key</i> )	
	Turunan ( <i>Inheritance</i> )	Specialisasi/Generalisasi
	Relasi ( <i>Relation</i> )	
	<i>Dependencies</i>	
	<i>Realizations</i>	

**Gambar 2. 6** Simbol Hubungan Antar *Class*

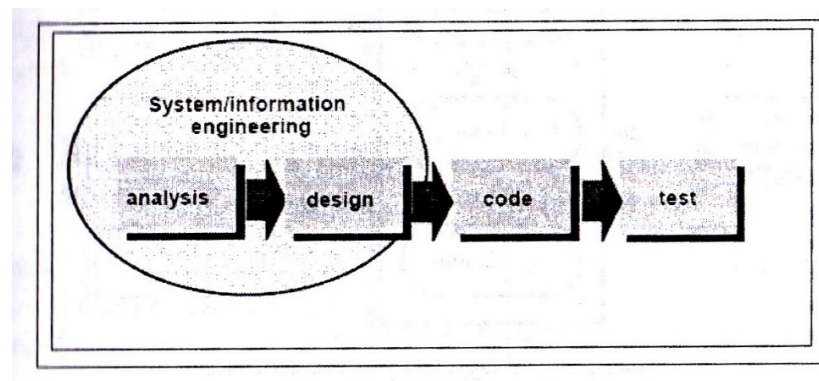
Berikut merupakan hubungan antar tiap class diagram:

1. Agregasi yang menunjukkan sebuah *class* merupakan bagian dari *class* yang lain.
2. Komposit merupakan relasi yang menunjukkan sebuah *class* yang memiliki ketergantungan terhadap *class* lain, dimana *class* ini tidak dapat terbentuk tanpa adanya *class* lain.
3. Turunan atau generalisasi merupakan relasi yang menunjukkan beberapa *class* yang memiliki induk dari *class* yang lain.  
Contoh: mahasiswa *parent* (induk) memiliki turunan yaitu *class* mahasiswa aktif, mahasiswa cuti.
4. Relasi merupakan bentuk hubungan umum antar tiap *class*
5. Dependensi adalah relasi yang menjelaskan hubungan dari suatu *class* diagram yang membentuk sebuah *class* baru.

#### 2.2.10 SDLC (*Software Development Life Cycle*)

Beberapa ahli membagi proses-proses pengembangan sistem ke dalam sejumlah urutan yang berbeda-beda. Pressman membagi tahapan SDLC dalam 4 tahap berikut.

Tahapan SDLC



**Gambar 2.7** Tahapan SDLC

SDLC dengan model seperti ini mempunyai beberapa kelemahan, yaitu:

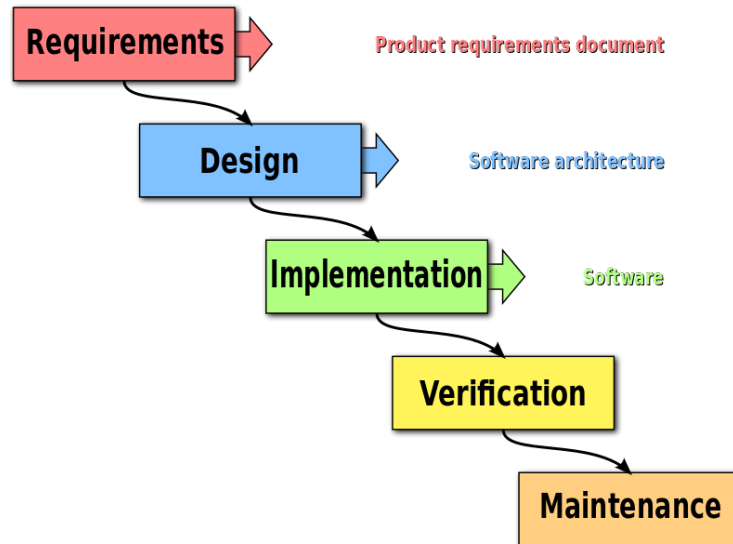
1. Terjadinya pembagian proyek menjadi tahap-tahap yang tidak fleksibel, karena komitmen harus dilakuakn pada tahap awal proses.
2. Dapat mengakibatkan sulitnya merespon perubahan kebutuhan pengguna.
3. Model SDLC harus digunakan hanya ketika persyaratan dipahami dengan baik.

Pada perkembangannya, proses-proses standar tadi dituangkan dalam satu metode yang dikenal dengan nama *System Development Life Cycle* (SDLC) yang merupakan metodologi umum dalam dalam pengembangan sistem yang menandai kemandirian usaha analisis dan desain. SDLC meliputi fase-fase sebagai berikut.

1. Identifikasi dan seleksi proyek
2. Inisiasi dan perencanaan proyek
3. Analisis
4. Desain
5. Implementasi
6. Pemeliharaan

Salah satu metodologi yang terdapat dalam SDLC ada *waterfall* , Model *Waterfall* adalah proses desain berurutan, yang digunakan dalam proses pengembangan perangkat lunak, di mana kemajuan dipandang sebagai mengalir terus ke bawah (seperti air terjun) melalui fase pembuahan, inisiasi, analisis, Desain, konstruksi, pengujian, produksi pelaksanaan dan pemeliharaan.

Bagan Metode *Waterfall*



Gambar 2.8 Bagan Metode *Waterfall*