

## BAB II

### LANDASAN TEORI

#### 2.1 Kajian Pustaka

Penelitian tentang sistem sirkulasi buku pada perpustakaan sudah banyak dilakukan. Penelitian yang dilakukan oleh *Dani Eko Hendrianto (2013)*. Pada penelitian ini, sistem informasi perpustakaan diharap dapat meningkatkan pelayanan dan kinerja petugas perpustakaan dalam hal pengelolaan data administrasi perpustakaan serta mempercepat pendataan buku, data admin, data peminjaman, transaksi peminjaman dan pengembalian buku oleh siswa. Pada sistem informasi tersebut terdapat menu data petugas, data *user*, laporan dan galeri buku. Gambar Sistem Informasi Perpustakaan SMP N 1 Donorojo dapat dilihat pada Gambar 2.1.



Gambar 2.1. Sistem Informasi Perpustakaan SMP N 1 Donorojo

Penelitian sistem informasi perpustakaan yang dilakukan Wafiroh (2016), bertujuan untuk mempermudah pelayanan dan akses informasi serta pengelolaan data perpustakaan, seperti mempermudah pencarian buku/katalog, peminjaman, pengembalian koleksi buku dan pembuatan laporan. Sehingga, akan diperoleh efisiensi pekerjaan staf perpustakaan dalam pengelolaan buku

perpustakaan, penyajian informasi yang lebih mudah dan interaktif, dan memberikan layanan yang lebih baik kepada pengguna layanan perpustakaan. Pada sistem informasi tersebut terdapat menu data anggota, data buku, data peminjaman, data pengembalian dan laporan. Gambar Sistem Informasi Perpustakaan SMAN 4 Kota Solok dapat dilihat pada Gambar 2.2.



Gambar 2.2. Sistem Informasi Perpustakaan SMAN 4 Kota Solok

Penelitian sistem perpustakaan yang dilakukan oleh Abidin (2016) di SMA Negeri 1 Sukoharjo, bertujuan untuk merancang dan membuat program aplikasi perpustakaan dengan Netbeans dan MySQL. Metode penelitian untuk pengambilan data yang digunakan antara lain observasi, wawancara, dan dokumentasi. Penelitian bertujuan untuk membantu dalam penyimpanan dan pengolahan data – data perpustakaan, seperti data peminjaman, data pengembalian, data buku, data anggota, dan data penerbit serta cetak laporan sehingga dapat meningkatkan kualitas layanan perpustakaan di SMA Negeri 1 Sukoharjo. Gambar Sistem Informasi Perpustakaan di SMA Negeri 1 Sukoharjo dapat dilihat pada Gambar 2.3.



Gambar 2.3. Sistem Informasi Perpustakaan SMA Negeri 1 Sukoharjo

Dari ketiga jurnal di atas maka dapat disimpulkan bahwa rata-rata masih menggunakan metode penyampaian informasi secara manual yakni yaitu proses pendaftaran anggota baru, peminjaman buku dan pengembalian buku yang telah dipinjam, semua itu hanya akan dicatat pada buku besar saja yang mana dalam hal ini bisa mengakibatkan kesalahan yang terjadi pada administrasi perpustakaan menjadi besar, data buku, data penerbit, data siswa, data peminjaman, data pengembalian dan laporan-laporan. Untuk itu saran yang diajukan dalam pembuatan sistem sirkulasi buku pada perpustakaan sehingga penyajian informasi yang lebih mudah dan interaktif, dan memberikan layanan yang lebih baik kepada pengguna layanan perpustakaan.

## 2.2 Kerangka Berfikir

Kerangka berfikir yang digunakan dalam membuat Sistem Sirkulasi Buku pada Perpustakaan di SD Negeri Blimbing 3 Sragen dapat dilihat pada Gambar 2.4.

Keterangan Diagram Kerangka Pemikiran (Gambar 2.4) :

1. Latar Belakang Masalah

Mahasiswa membangun sistem sirkulasi buku berbasis *web* di perpustakaan SD Negeri Blimbing 3 Sragen.

2. Penguasaan Dasar (PHP dan MySQL)

Mahasiswa melakukan beberapa percobaan pembuatan aplikasi sederhana dengan PHP dan MySQL agar lebih menguasai bahasa pemrograman tersebut dan mendapatkan hasil yang maksimal.

3. Observasi Aplikasi

Mahasiswa mengamati beberapa aplikasi yang sudah ada, baik dari karya ilmiah, buku atau internet yang dapat dijadikan referensi untuk membangun sistem sirkulasi buku pada perpustakaan di SD Negeri Blimbing 3 Sragen.

4. Analisis dan Perancangan Sistem

Mahasiswa menganalisa dan merancang sistem sirkulasi buku yang akan digunakan untuk perpustakaan di SD Negeri Blimbing 3 Sragen.

5. Implementasi dan Pengembangan Sistem

a. Membuat *database* dengan MYSQL

Mahasiswa membuat *database* dari data-data yang telah didapatkan sesuai dengan kebutuhan sistem sirkulasi buku pada perpustakaan di SD Negeri Blimbing 3 Sragen..

b. Perancangan aplikasi pengolahan data

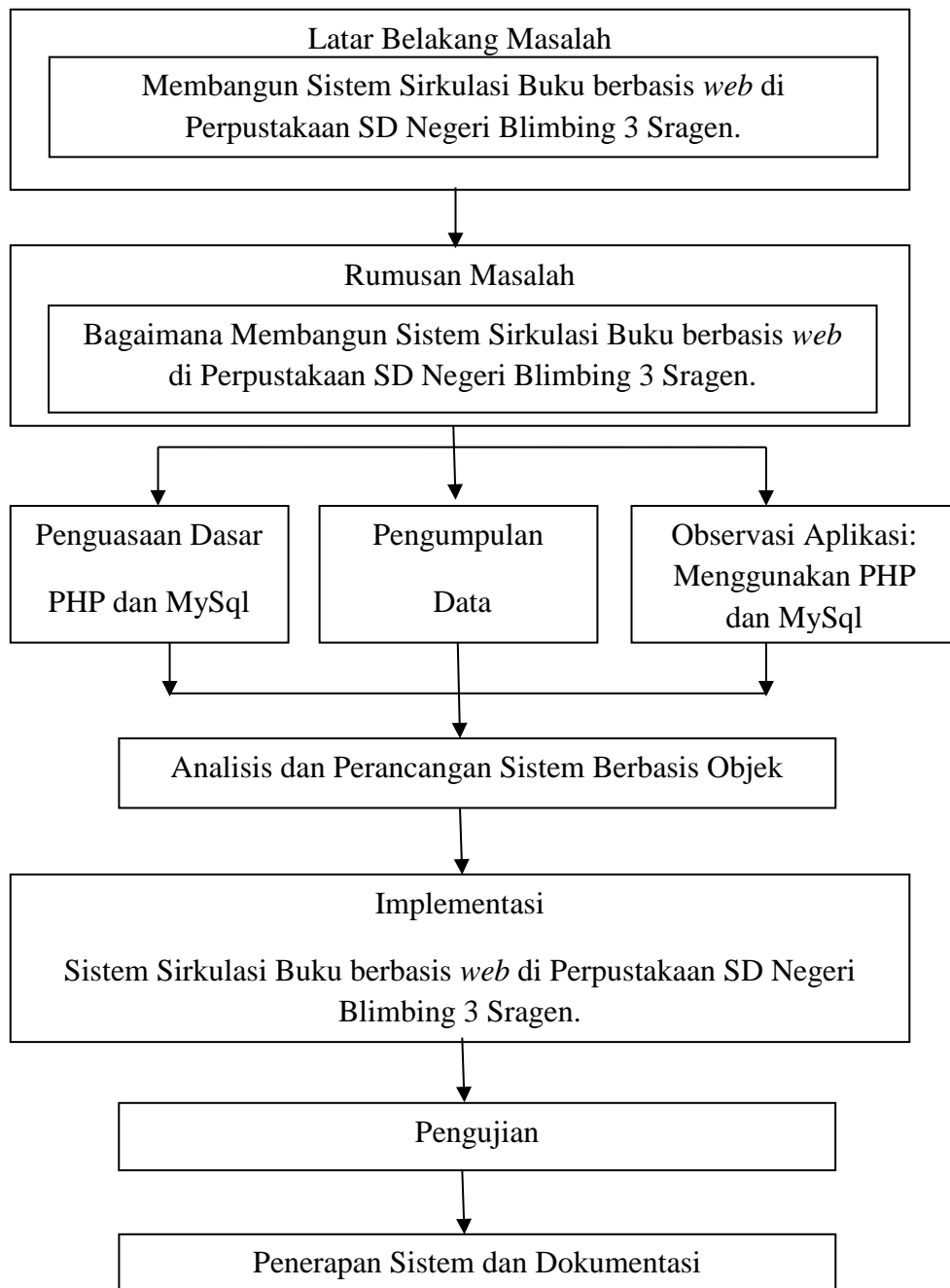
Mahasiswa membuat aplikasi dengan dasar *database* yang telah selesai dibuat.

6. Pengujian Sistem

Pengujian sistem ini dilakukan bertujuan untuk mengetahui jika ternyata masih ada kesalahan atau kekurangan pada sistem sirkulasi buku pada perpustakaan di SD Negeri Blimbing 3 Sragen yang telah selesai dibuat..

7. Penerapan Sistem dan Dokumentasi

Pada tahap akhir, dimana sistem telah siap digunakan setelah melewati tahap pengujian dan membuat dokumentasi dari keseluruhan kegiatan penyusunan tugas akhir.



Gambar 2.4. Diagram Kerangka Pemikiran

## **2.3 Landasan Teori**

### **2.3.1 Sistem**

Menurut Ladjamudin (2006) kata “Sistem” mengandung arti kumpulan dari komponen-komponen yang memiliki keterkaitan antara yang satu dengan yang lainnya. Dari definisi sistem, maka dapat didefinisikan bahwa “Sistem informasi adalah suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi”.

Menurut Azhar Susanto (2013) sistem adalah kumpulan dari sub sistem/bagian/komponen apapun baik fisik maupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu.

### **2.3.2 Perpustakaan**

Menurut Zulaekhah dkk (2009) perpustakaan ialah sumber informasi yang dilakukan oleh pustaka untuk memenuhi kebutuhan pemustaka. Layanan ini merupakan aktivitas perpustakaan dalam memberikan jasa perpustakaan kepada pemustaka. Jenis/divisi layanan pemustaka cukup banyak. Adapun penyelenggaraan semua jenis/divisi layanan tersebut disesuaikan dengan kondisi di setiap perpustakaan dan kebutuhan pemustakanya.

### **2.3.3 Sirkulasi**

Menurut Pawit M. Yusuf (2007) sirkulasi adalah pelayanan yang memutar koleksi, dipinjam keluar, dikembalikan, dipinjam keluar lagi, dikembalikan lagi dan seterusnya. Dalam dunia perpustakaan memiliki arti perputaran buku atau jenis koleksi untuk beberapa waktu lamanya.

### **2.3.4 Buku**

Menurut Tarigan (1986) buku teks adalah buku pelajaran dalam bidang studi tertentu, yang merupakan buku standar, yang disusun oleh pakar dalam bidang itu untuk maksud-maksud dan tujuan instruksional, yang diperlengkapi dengan sarana-sarana pengajaran yang serasi dan mudah dipahami oleh para pemakainya di sekolah-sekolah dan perguruan tinggi sehingga dapat menunjang sesuatu program pengajaran.

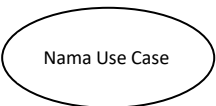


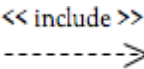
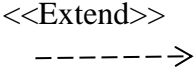

### 2.3.5 UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) adalah ‘bahasa’ pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi object’ (Nugroho : 2010). Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks demikian rupa sehingga lebih mudah dipelajari dan dipahami.

#### 2.3.5.1 *Use Case Diagram*

*Use case diagram* yang menggambarkan *actor*, *use case* dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor. Sebuah *use case* digambarkan sebagai elips horizontal dalam suatu diagram UML *use case*. Simbol–simbol *Use Case Diagram* dapat dilihat pada Tabel 2.1




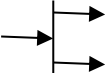
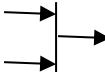

Tabel 2.1 Simbol–simbol *Use Case Diagram*

No	Gambar	Nama	Keterangan
1.		<i>Use Case</i>	Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
2.		<i>Aktor</i>	Orang proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri.
3		<i>Assosiasi</i>	Komunikasi antar aktor dan <i>usecase</i> yang berpartisipasi pada <i>usecase</i> atau sebaliknya.
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit
5		<i>Extend</i>	Menspesifikasikan bahwa perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi
6		<i>Generalisasi</i>	Hubungan generalisai dan spesialis antara dua buah <i>use case</i> .

### 2.3.5.2 Activity Diagram

*Activity Diagram* adalah diagram yang menggambarkan aliran kerja atau aktifitas dari suatu sistem. Perlu diperhatikan bahwa *activity diagram* menggambarkan aktifitas sistem bukan apa yang dilakukan aktor. Simbol–simbol *Activity Diagram* dapat dilihat pada Tabel 2.2.

Tabel 2.2 Simbol–simbol *Activity Diagram*

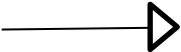
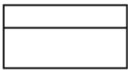
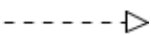
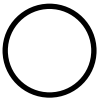



No	Gambar	Nama	Keterangan
1.		<i>Aktivitas</i>	Aktivitas yang dilakukan sistem, aktifitas biasanya diawali dengan kata kerja.
2.		<i>Status Awal</i>	Bagaimana object dibentuk atau diawali.
3.		<i>Status Akhir</i>	Bagaimana object dibentuk dan dihancurkan.
4.		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.
5.		<i>Join Node</i>	Beberapa aliran yang pada tahap tertentu menjadi satu aliran.
6.		<i>Percabangan</i>	Suatu titik/point pada <i>activity diagram</i> yang mengindikasikan suatu kondisi dimana ada kemungkinan perbedaan transisi.

### 2.3.5.3 Class Diagram

*Class Diagram* pada Tabel 2.3 menggambarkan struktur dan deskripsi *class*, *package* dan object beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class diagram* membantu kita dalam visualisasi struktur kelas-kelas dari suatu sistem yang merupakan tipe diagram yang paling banyak dipakai. Simbol–simbol *Class Diagram* dapat dilihat pada Tabel 2.3.




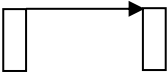




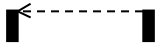
Tabel 2.3 Simbol–simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1.		Generalisasi	Hubungan dimana object anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari object yang ada di atasnya object induk ( <i>ancestor</i> ).
2.		Kelas	Himpunan dari object-object yang berbagi atribut serta operasi yang sama.
3.		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu object.
4.		Antarmuka	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
5.		Agregasi	Relasi antarkelas dengan makna semua bagian
6.		Ketergantungan	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7.		Assosiasi	Apa yang menghubungkan antara object satu dengan object lainnya

#### 2.3.5.4 Sequence Diagram

*Sequence Diagram* pada Tabel 2.4 mendeskripsikan bagaimana entitas dalam sistem berinteraksi, termasuk pesan yang digunakan saat interaksi. Diagram ini juga menunjukkan serangkaian pesan yang diperlukan oleh object-object yang melakukan suatu tugas atau aksi tertentu. Simbol–simbol *Sequence Diagram* dapat dilihat pada Tabel 2.4.

Tabel 2.4 Simbol–simbol *Sequence Diagram*

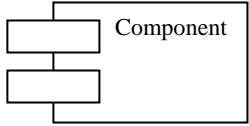
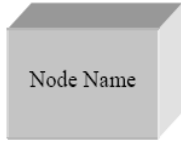

No	Gambar	Nama	Keterangan
1		<i>LifeLine</i>	Object <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar object yang memuat informasi-informasi tentang aktifitas yang terjadi.
3		<i>Actor</i>	Pengguna diluar sistem.
4		<i>Boundary</i>	Boundary biasanya berupa tepi dari sistem, seperti user interface, atau suatu alat yang berinteraksi dengan sistem lain.
5		<i>Control element</i>	Control element mengatur aliran dari informasi untuk sebuah scenario.
6		<i>Entity</i>	Entity biasanya elemen yang bertanggung jawab menyimpan data atau informasi.
7		<i>Return</i>	Spesifikasi dari komunikasi antar object yang memuat informasi-informasi tentang aktifitas yang terjadi

### 2.3.5.5 *Deployment Diagram*

*Deployment Diagram* adalah bahasa yang digunakan untuk memvisualisasikan dan mendokumentasikan proses yang terjadi pada suatu sistem perangkat lunak berbasis obyek. Fungsinya adalah memvisualisasikan atau menggambarkan secara umum proses yang terjadi pada suatu sistem. Pada Tabel 2.5 menggambarkan detail bagaimana komponen di-sebar (*di-deploy*) kedalam infrastruktur sistem, dimana komponen akan terletak (pada mesin, *node*, *server*,

atau piranti kertas apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisik. Simbol–simbol *Deployment Diagram* dapat dilihat pada Tabel 2.5.

Tabel 2.5 Simbol–simbol *Deployment Diagram*

No	Nama	Gambar	Keterangan
1	 Component	<i>Component</i>	Pada <i>deployment diagram</i> , <i>component - component</i> yang ada diletakkan didalam node untuk memastikan keberadaan posisi mereka.
2	 Node Name	<i>Node</i>	<i>Node</i> menggambarkan bagian-bagian <i>hardware</i> dalam sebuah sistem. Notasi untuk node digambarkan sebagai sebuah kubus 3 dimensi.
3		<i>Association</i>	Sebuah <i>association</i> digambarkan sebagai sebuah garis yang menghubungkan dua node yang mengindikasikan jalur komunikasi antara element-element <i>hardware</i> .

### 2.3.6. Software Pendukung

#### 2.3.6.1 Adobe Dreamweaver

Menurut Alexander F.K Sibero (2011), *Adobe dreamweaver* adalah suatu produk *web developer* yang dikembangkan oleh *Adobe System Inc.* ruang kerja atau *workspace* adalah bagian keseluruhan tampilan *adobe dreamweaver* yang terdiri dari *welcome screen*, *menu*, *insert bar*, *document window*, *css panel*, *application panel* yang memiliki fungsi dan aturan masing-masing.

*Adobe Dreamweaver* memudahkan pengembang *website* untuk mengelola halaman *website* dan asset-asetnya, baik gambar , animasi flash, video, suara dan lain sebagainya. Selain itu, *adobe Dreamweaver* juga menyediakan fasilitas untuk melakukan pemrograman scripting, baik ASP, JSP, PHP, JavaScript, Cold Fusion, CSS, XML dan lainnya.

#### 2.3.6.2 XAMPP Control Panel

XAMPP adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolah data MYSQL di komputer lokal”.

XAMPP berperan sebagai *server web* pada komputer lokal. XAMPP juga dapat disebut sebuah *Cpanel server virtual*, yang dapat membantu melakukan *preview* sehingga dapat dimodifikasi *website* tanpa harus *online* atau terakses dengan *internet*.

Kepanjangan dari XAMPP sendiri yaitu Apache, PHP, MySQL dan perl. XAMPP merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstall XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server* Apache, PHP dan MySQL secara manual. XAMPP akan menginstallasi dan mengkonfigurasikannya secara otomatis untuk anda atau auto konfigurasi.

Mengenal bagian XAMPP yang biasa digunakan pada umumnya : *htdocs* adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan skrip lain. *phpMyAdmin* merupakan bagian untuk mengelola basis data MySQL yang ada dikomputer. Kontrol Panel yang berfungsi untuk mengelola layanan *service* XAMPP. Seperti menghentikan *stop* layanan, ataupun memulai *start*.

### **2.3.6.3 MySQL**

Menurut Anhar (2010) MySQL adalah salah satu *databases management system* (DBMS) dari sekian banyak DBMS seperti *Oracle*, *MS SQL*, *Postagre SQL*, dan lainnya. MySQL berfungsi untuk mengolah *database* menggunakan bahasa SQL. MySQL bersifat open source sehingga kita bisa menggunakannya secara gratis. Pemograman PHP juga sangat mendukung/ *support* dengan *database* MySQL.

MySQL memiliki beberapa keistimewaan, antara lain :

- 1) Portabilitas, MySQL dapat berjalan stabil pada berbagai sistem operasi, seperti : *Windows*, *Linux*, *FreeBSD*, *Mac Os X Server*, *Solaris*, *Amiga*.
- 2) Perangkat lunak sumber terbuka, MySQL didistribusikan dibawah lisensi GPL sehingga dapat digunakan secara gratis.
- 3) Multi-*user*, MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

- 4) *Performance tuning*, MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
- 5) Ragam tipe data, MySQL memiliki ragam tipe data yang sangat kaya, seperti *signed/unsigned integer, float, double, char, text, date, timestamp*, dan lain-lain.
- 6) Perintah dan fungsi, MySQL memiliki operator dan fungsi secara penuhyang mendukung perintah *select* dan *where* dalam perintah (*query*).
- 7) Keamanan, MySQL memiliki beberapa lapisan keamanan seperti level *subnetmask*, nama *host* dan izin akses *user* dengan sistem perizinan yang mendetail secara sandi terenskripsi.
- 8) Skalabilitas dan pembatasan, MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman atau *record* lebih dari 50 juta dan 60.000 tabel serta 5 milyar baris. Selain itu batas *indeks* yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
- 9) Konektivitas, MySQL dapat melakukan koneksi dengan *client* menggunakan *protocol* TCP/IP, *unix socket* (UNIX) atau *named pipes* (NT).
- 10) Lokalisasi, MySQL dapat mendeteksi pesan kesalahan pada client dengan menggunakan lebih dari 20 bahasa. Meskipun demikian, Bahasa Indonesia belum termasuk didalamnya.
- 11) Antar muka, MySQL memiliki antar muka (*interface*) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Proramming Interface*).
- 12) *Client* dan peralatan, MySQL dilengkapi dengan berbagai peralatan yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk *online*.
- 13) Struktur tabel, MySQL memiliki struktur tabel yang lebih *fleksibel* dalam menangani *ALTER TABLE*, dibandingkan basis data lainnya semacam *postgre SQL* ataupun *Oracle*.

#### 2.3.6.4 PHP

Menurut Diar Puji (2010) PHP singkatan dari *Hypertext Preprocessor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source*. PHP merupakan script yang terintegrasi dengan HTML dan berada pada *server (server side HTML embedded scripting)*. PHP adalah *script* yang digunakan untuk membuat halaman yang dinamis (*up to date*) (Anhar : 2010).

Kode PHP mempunyai beberapa ciri-ciri khusus, yaitu:

- 1) Hanya dapat dijalankan menggunakan *web server*, misalnya: Apache.
- 2) Kode PHP diletakkan dan dijalankan di *web server*.
- 3) Kode PHP dapat digunakan untuk mengakses *database*, seperti Mysql, PostgreSQL, Oracle, dan lain-lain.
- 4) Merupakan *software* yang bersifat *open source*.
- 5) Gratis untuk di *download* dan digunakan.

Memiliki sifat *multiplatform*, artinya dapat dijalankan menggunakan sistem operasi apapun, seperti *Linux, Unix, Windows*, dan lain-lain.

#### 2.4 Metode Pengujian *Blackbox (Blackbox Testing)*

Menurut Roger S. Pressman (2010) *Black Box Testing* atau Pengujian Kotak Hitam atau juga disebut Behavioral Testing, berfokus pada persyaratan fungsional dari perangkat lunak. Artinya, teknik *Black Box Testing* memungkinkan untuk mendapatkan *set* kondisi masukan yang sepenuhnya akan melaksanakan semua persyaratan fungsional untuk suatu program. *Black Box Testing* bukan merupakan alternatif dari pengujian *White Box Testing*. Sebaliknya, *Black Box Testing* adalah pendekatan *komplementer* yang mungkin untuk mengungkap kelas yang berbeda dari kesalahan daripada metode *White Box Testing*.

*Black Box Testing* mencoba untuk menemukan kesalahan dalam kategori berikut :

1. Fungsi tidak benar atau hilang.
2. Kesalahan *interface* atau antarmuka.
3. Kesalahan dalam struktur data atau akses *database eksternal*.
4. Kesalahan kinerja atau perilaku.
5. Kesalahan inisialisasi dan terminasi.