

## **BAB II**

### **KAJIAN PUSTAKA**

#### **2.1. Teori Umum**

##### **2.1.1. Sistem**

Menurut Ali (2010: 8) Sistem adalah kumpulan dari sub –sub sistem, elemen – elemen, prosedur – prosedur, yang saling berintergrasi untuk mencapai tujuan tertentu, seperti informasi, target atau goal. Karakter suatu sistem terdiri dari: Komponen (*Components*), Batas Sistem (*Boundary*), Lingkungan luar sistem (*Environments*), Penghubung (*Interface*), Input, proses, dan output, Sasaran (*Objectives*), Tujuan (*Goal*).

##### **2.1.1.1. Komponen Sistem**

Masing – masing komponen sistem memiliki fungsi yang berbeda – beda. Komponen tersebut adalah sebagai berikut (Novita dan Sari, 2015: 2):

1. *Input*, menggambarkan kegiatan untuk menyediakan data yang akan diproses.
2. *Proses*, menggambarkan bagaimana suatu data diproses untuk menghasilkan suatu informasi yang bernilai tambah.
3. *Output*, suatu kegiatan untuk menghasilkan laporan dari proses diatas.

4. Penyimpanan, suatu kegiatan untuk memelihara dan menyimpan data.
5. Kontrol, suatu aktifitas untuk menjamin bahwa sistem informasi tersebut berjalan sesuai dengan yang diharapkan.

### **2.1.2. Informasi**

Menurut Ali (2010 : 10) Informasi adalah data yang telah diolah menjadi bentuk yang penting bagi sipenerima dan mempunyai nilai nyata atau dapat dirasakan manfaatnya dalam keputusan – keputusan yang akan datang. Data adalah representasi suatu objek.

### **2.1.3. Sistem Informasi**

Berdasarkan penelitian Watung, dkk (2014: 1) Sistem informasi dapat didefinisikan sebagai kumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan untuk mengintegrasikan data memproses dan menyimpan serta mendistribusikan informasi. Dengan kata lain SI merupakan kesatuan elemen-elemen yang saling berinteraksi secara sistematis dan teratur untuk menciptakan dan membentuk aliran informasi yang akan mendukung pembuatan keputusan dan melakukan kontrol terhadap jalannya perusahaan. Menurut Tantra (2012: 2) Sistem informasi adalah cara yang terorganisir untuk mengumpulkan, memasukan, dan memroses data dan menyimpannya, mengelola, mengontrol dan melaporkannya sehingga dapat mendukung perusahaan atau organisasi untuk mencapai tujuan.

### 2.1.3.1. Konsep Dasar Sistem Informasi

Berdasarkan penelitian Novita dan Sari (2015: 2), Sistem informasi memberikan nilai tambah terhadap proses, produksi, kualitas, manajemen, pengambila keputusan, dan pemecahan masalah serta keunggulan kompetitif yang tentu saja sangat berguna bagi kegiatan bisnis. Kegiatan yang terdapat pada sistem informasi antara lain:

- a. *Input*, menggambarkan suatu kegiatan untuk menyediakan data yang akan diproses.
- b. Proses, menggambarkan bagaimana suatu data diproses untuk menghasilkan suatu informasi yang bernilai tambah.
- c. *Output*, suatu kegiatan untuk menghasilkan laporan dari proses diatas.
- d. Penyimpanan, suatu kegiatan untk memelihara dan menyimpan data.
- e. Kontrol, suatu aktifitas untuk menjamin bahwa sistem informasi tersebut berjalan sesuai dengan yang diharapkan.

### 2.1.4. UML

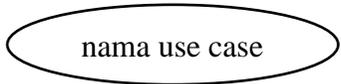
Menurut (Rosa & Shalahuddin, 2011) UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks – teks pendukung. UML hanya berfungsi untuk melakukan pemodelan, jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

Berikut merupakan jenis-jenis UML yang akan digunakan dalam penelitian ini:

### 1. Use Case Diagram

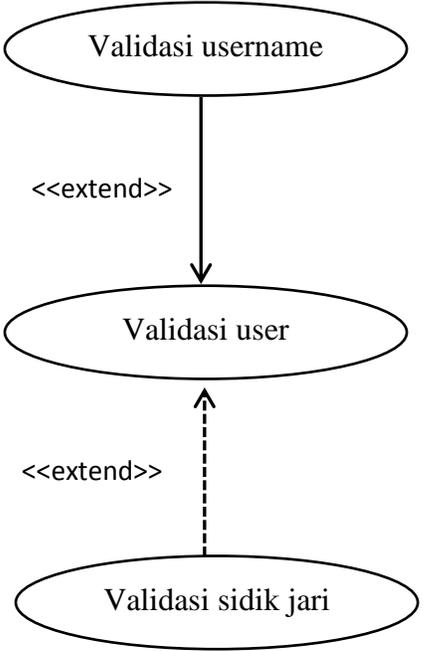
Menurut A.S, Rosa dan M. Shalahuddin (2011: 130), *Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Berikut adalah simbol-simbol yang ada pada diagram *use case*:

**Tabel 2.1** Simbol-simbol diagram *use case*

Simbol	Deskripsi
<p><i>Use case</i></p>  <p>nama use case</p>	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i></p>
<p>Aktor / <i>actor</i></p>  <p>nama actor</p>	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang; tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.</p>
<p>Asosiasi / <i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.</p>

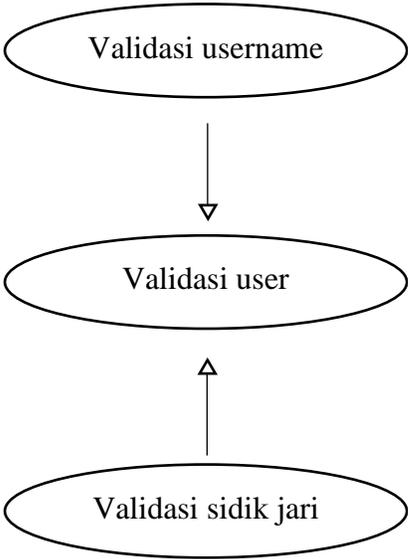
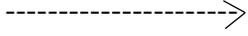
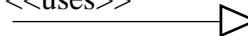
Sumber: A.S Rosa dan M.Shalahuddin

Tabel 2.1 Lanjutan

Simbol	Deskripsi
<p>Ekstensi / <i>extend</i></p> <p>&lt;&lt;extend&gt;&gt;</p> <p>-----&gt;</p>	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.</p>  <pre> graph TD     A([Validasi user]) -- "&lt;&lt;extend&gt;&gt;" --&gt; B([Validasi username])     C([Validasi sidik jari]) -.-&gt; "&lt;&lt;extend&gt;&gt;" --&gt; A   </pre>

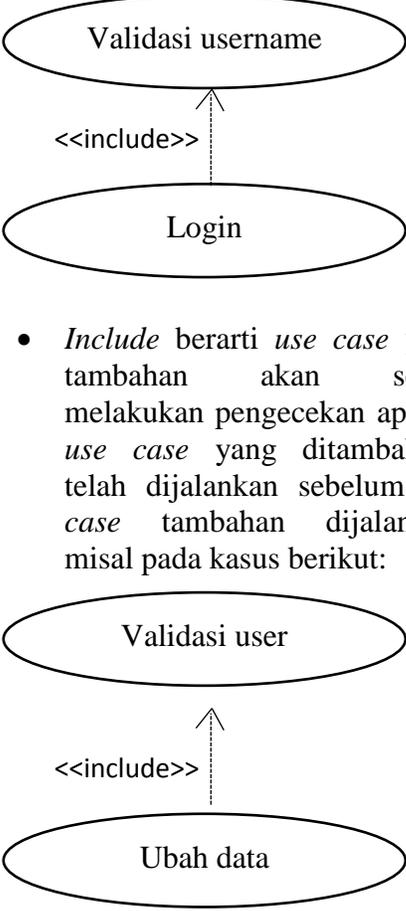
Sumber: A.S Rosa dan M.Shalahuddin

Tabel 2.1 Lanjutan

Simbol	Deskripsi
<p>Generalisasi / <i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dan lainnya, misalnya:</p>  <p>Arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum)</p>
<p>Menggunakan / <i>include / uses</i></p> <p>&lt;&lt;include&gt;&gt;</p>  <p>&lt;&lt;uses&gt;&gt;</p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p> <p>Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i>:</p> <ul style="list-style-type: none"> <li>• <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, misal pada kasus berikut:</li> </ul>

Sumber: A.S Rosa dan M.Shalahuddin

Tabel 2.1 Lanjutan

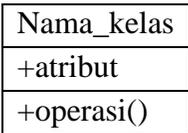
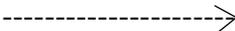
Simbol	Deskripsi
	 <pre> graph TD     Login(Login) -.-&gt; &lt;&lt;include&gt;&gt;  Validasi_username(Validasi username)     Ubah_data(Ubah data) -.-&gt; &lt;&lt;include&gt;&gt;  Validasi_user(Validasi user)   </pre> <p>• <i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan, misal pada kasus berikut:</p> <p>Kedua interpretasi diatas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan</p>

Sumber: A.S Rosa dan M.Shalahuddin

## 2. Class Diagram

Menurut A.S, Rosa dan M. Shalahuddin (2011: 122), Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

**Tabel 2. 2** Simbol-simbol diagram kelas

Simbol	Deskripsi
Kelas 	Kelas pada struktur system
Antarmuka/ <i>interface</i>  <b>Nama_interface</b>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi / <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Asosiasi berarah / <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan / <i>dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi / <i>aggregation</i> 	Relasi antar kelas dengan makna semua-bagian (whole-part)

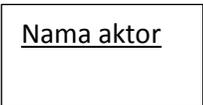
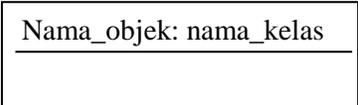
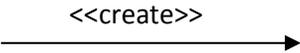
Sumber: A.S Rosa dan M.Shalahuddin

### 3. *Sequence Diagram*

Menurut A.S, Rosa dan M. Shalahuddin (2011: 137), Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antarobjek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek

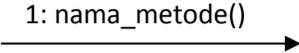
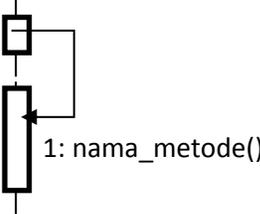
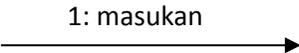
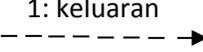
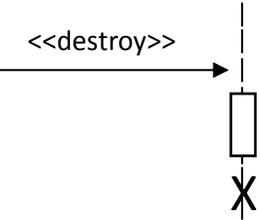
yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki jelas yang diinstansi menjadi objek itu.

**Tabel 2. 3** Simbol-simbol diagram sekuen

Simbol	Deskripsi
<p>Aktor</p>  <p>Atau</p>  <p>Tanpa waktu aktif</p>	<p>Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun <i>symbol</i> dari aktor adalah gambar orang, biasanya dinyatakan menggunakan kata benda diawal frase nama actor</p>
<p>Garis hidup /<i>lifeline</i></p> 	<p>Menyatakan garis hidup objek</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan.</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi pesan</p>
<p>Pesan tipe <i>create</i></p> 	<p>Objek yang lain, arah panah mengarah pada objek yang dibuat</p>

Sumber: A.S Rosa dan M.Shalahuddin

Tabel 2.3 Lanjutan

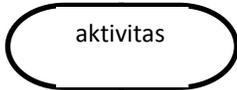
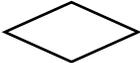
Simbol	Deskripsi
<p>Pesan tipe <i>call</i></p> 	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,</p>  <p>Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi</p>
<p>Pesan tipe <i>send</i></p> 	<p>Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek yang lainnya, arah panah mengarah pada objek yang dikirim</p>
<p>Pesan tipe <i>return</i></p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>
<p>Pesan tipe <i>destroy</i></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy</p>

Sumber: A.S Rosa dan M.Shalahuddin

#### 4. Activity Diagram

Menurut A.S, Rosa dan M. Shalahuddin (2011: 134), Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Berikut adalah simbol-simbol yang ada pada diagram aktivitas:

**Tabel 2. 4** Simbol-simbol diagram aktivitas

Simbol		Deskripsi		
Status awal		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal		
Aktivitas		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja		
Percabangan / <i>decision</i>		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu		
Penggabungan / <i>join</i>		Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu		
Status akhir		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir		
Swimlane	<table border="1" data-bbox="444 1577 716 1818"> <tr> <td>Nama swimlane</td> </tr> <tr> <td> </td> </tr> </table>	Nama swimlane		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi
Nama swimlane				

Sumber: A.S Rosa dan M.Shalahuddin

### **2.1.5. Perancangan atau Desain**

Menurut A.S, Rosa dan M.Shalahuddin (2011:21), Desain atau perancangan dalam pembangunan perangkat lunak merupakan upaya untuk mengonstruksi sebuah sistem yang memberikan kepuasan (mungkin informal) akan spesifikasi kebutuhan fungsional, memenuhi target, memenuhi kebutuhan implisit atau eksplisit dari segi performansi maupun penggunaan sumber daya, kepuasan batasan pada proses desain dari segi biaya, waktu dan perangkat. Kualitas perangkat lunak biasanya dinilai dari segi kepuasan pengguna perangkat lunak terhadap perangkat lunak yang digunakan.

## **2.2. Teori Khusus**

### **2.2.1. Penjualan**

Berdasarkan penelitian Viantoni, dkk (2009: 60) penjualan dapat diartikan sebagai prosedur pengeluaran barang dan penerimaan uang oleh perusahaan. Kegiatan penjualan terdiri dari transaksi penjualan barang atau jasa baik secara tunai maupun kredit. Dalam transaksi penjualan tidak semua penjualan berhasil mendatangkan pendapatan bagi perusahaan, adakalanya pembeli mengembalikan barang yang telah dibelinya kepada perusahaan atau dengan kata lain retur penjualan. Penjualan merupakan aktivitas utama perusahaan yang sangat penting sebab penjualan merupakan sasaran terakhir dari seluruh aktivitasnya perusahaan yaitu mendapatkan keuntungan.

Pembahasan sistem penjualan dibagi kedalam dua macam sistem, yaitu:

1. Sistem penjualan tunai

Pada sistem ini, apabila pembeli sudah memilih barang yang akan dibeli, pembeli diharuskan membayar ke bagian kasir. Selain itu barang yang dibeli dibawa langsung oleh pembeli. Oleh karena itu tidak perlu ada prosedur pengiriman barang pada toko penjualan.

2. Sistem penjualan kredit

Menurut Mulyadi (2008) dalam jurnal Riswya (2013: 110), Adapun prosedur – prosedur penjualan kredit adalah sebagai berikut:

- a. Prosedur Order Penjualan

Dalam prosedur ini bagian penjualan menerima order dari pembeli dan menambahkan informasi penting pada surat order dari pembeli. Bagian penjualan kemudian membuat faktur penjualan dan mengirimkannya kepada bagian yang lain untuk memungkinkan bagian tersebut memberikan kontribusi dalam melayani order dari pembeli.

- b. Prosedur Pengiriman

Dalam prosedur ini bagian gudang menyiapkan barang yang diperlukan oleh pembeli dan bagian pengiriman mengirimkan barang kepada pembeli sesuai dengan informasi yang tercantum dalam faktur penjualan yang diterima dari bagian gudang.

c. Prosedur Pencatatan Piutang

Dalam prosedur ini bagian akuntansi mencatat tebusan faktur faktur penjualan kedalam kartu piutang.

d. Prosedur Penagihan

Dalam prosedur ini bagian penagihan menerima faktur penjualan dan mengarsipkannya menurut abjad. Secara periodik bagian penagihan membuat surat tagihan dan mengirimkannya kepada pembeli tersebut yang dilampiri dengan faktur penjualan.

e. Prosedur Pencatatan Penjualan

Dalam prosedur ini bagian akuntansi mencatat transaksi penjualan kedalam jurnal penjualan.

### **2.2.1.1. Prosedur Penjualan**

Berdasarkan penelitian Riandy, dkk (2011: 2) Sejak dulu sampai sekarang penjualan masih tetap berperan paling penting diantara aktivitas lainnya. Penjualan merupakan sumber hidup suatu perusahaan, karena dari penjualan dapat diperoleh laba. Banyak faktor selain usaha promosi termasuk segi pelayanan atau pemberian *service*, harga yang cocok, juga bagian pimpinan perusahaan dapat mengikuti *order* penjualan yang masih luas/terbuka. Maka aktivitas penjualan dapat dikronologiskan sebagai berikut:

1. Penerimaan pesanan.
2. Penegasan pesanan.
3. Pengiriman barang.

4. Pembuatan faktur.
5. Pembuatan laporan operasional penjualan.

Prosedur penjualan adalah urutan-urutan sejak diterimanya pesanan dari pembeli pengiriman barang, pembuatan faktur (penagihan) dan pencatatan penjualan. Prosedur penjualan melibatkan beberapa bagian dalam perusahaan dengan maksud agar penjualan yang terjadi dapat diawasi dengan baik. Bagian-bagian yang terkait dalam penjualan adalah bagian pesanan penjualan, bagian kredit, bagian gudang, bagian pengiriman dan bagian *filling*.

### **2.2.2. Sistem Informasi Penjualan**

Menurut Sutabri (2004) dalam jurnal Viantoni, dkk (2009: 61) Sistem Informasi Penjualan adalah suatu sistem informasi yang mengorganisasikan serangkaian prosedur dan metode yang dirancang untuk menghasilkan, menganalisa, menyebarkan dan memperoleh informasi guna mendukung pengambilan keputusan mengenai penjualan.

Menurut Achun (2007) dalam jurnal Murniati, dkk (2013: 2-3) Sistem informasi penjualan adalah sub sistem informasi bisnis yang mencakup kumpulan prosedur yang melaksanakan, mencatat, mengkalkulasi, membuat dokumen dan informasi penjualan untuk keperluan manajemen dan bagian lain yang berkepentingan, mulai dari diterimanya order penjualan sampai mencatat timbulnya tagihan/piutang dagang.

### 2.2.3.PHP

Berdasarkan penelitian Reza, dkk (2013: 3) PHP merupakan sebuah bahasa skrip yang dapat ditanamkan atau disisipkan pada HTML. Penggunaan PHP ini sudah menjadi hal umum dalam membuat web yang dinamis. Bahkan *Framework* dan CMS (*Content Management System*) juga menggunakan PHP dalam pembuatannya.

Berdasarkan penelitian Watung, dkk (2014: 2) PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman open source yang sangat cocok atau dikhususkan untuk pengembangan Web dan dapat di tanamkan pada sebuah skrip HTML. Bahasa PHP dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti C, Java, dan Perl serta mudah untuk dipelajari.

Sistem kerja dari PHP diawali dengan permintaan yang berasal dari halaman website oleh browser. Berdasarkan URL atau alamat website dalam jaringan internet, browser akan menemukan sebuah alamat dari webserver, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan oleh webserver.

Selanjutnya webserver akan mencarikan berkas yang diminta dan menampilkan isinya di browser. Browser yang mendapatkan isinya segera menerjemahkan kode HTML dan menampilkannya. Lalu bagaimana apabila yang dipanggil oleh user adalah halaman yang mengandung script PHP ? Pada prinsipnya sama dengan memanggil kode HTML, namun pada saat permintaan dikirim ke web-server, web-server akan memeriksa tipe file yang diminta user. Jika tipe file yang diminta adalah PHP, maka akan memeriksa isi script dari halaman PHP tersebut.

#### 2.2.4. Apache Web Server

Berdasarkan penelitian Reza, dkk (2013: 3) Apache adalah *web server* kuat yang tersedia benar – benar gratis serta memiliki fitur yang lengkap. Apache bersifat gratis dan bebas untuk dipergunakan atau dimodifikasi untuk memperluas fungsional apache. Apache juga dikemas untuk sebagian besar distribusi Linux.

#### 2.2.5. jQuery

Berdasarkan penelitian Yatini (2014: 2) jQuery adalah JavaScript *library* yang dirancang untuk meringkas kode-kode JavaScript, sehingga dapat menyederhanakan penulisan skrip program, sesuai dengan slogan “*write less, do more*”. jQuery pertama kali dirilis oleh John Resig pada tahun 2006, pada perkembangannya jQuery tidak hanya sebagai *framework* JavaScript, namun memiliki kelebihan antara lain.

1. Kemudahan mengakses dan memanipulasi elemen-elemen HTML.
2. Memanipulasi CSS.
3. Penanganan *event* HTML.
4. Efek-efek JavaScript dan animasi.
5. Memodifikasi elemen HTML DOM.

Sintak dasar jQuery  $$(selector).action()$ , tanda \$ untuk mendefinisikan jQuery, jQuery selector digunakan untuk mendapatkan elemen HTML, action adalah tindakan yang dilakukan jQuery pada elemen ().

Contoh penggunaan jQuery untuk menyembunyikan elemen dengan id “*test*” sebagai berikut.

```
$("#test").hide()
```

Semua metode jQuery berada di dalam fungsi *document.ready()* yaitu perintah inisialisasi yang menunjukkan dokumen telah siap ditampilkan dan sekaligus menjalankan perintah yang terdapat didalam fungsi.

### 2.2.6. JavaScript

Berdasarkan penelitian Yatini (2014: 2) JavaScript adalah bahasa *scripting* kecil, ringan, berorientasi objek yang ditempelkan pada kode HTML dan di proses di sisi *client*. JavaScript digunakan dalam pembuatan *website* agar lebih interaktif dengan memberikan kemampuan tambahan terhadap HTML melalui eksekusi perintah di sisi *browser*. JavaScript dapat merespon perintah *user* dengan cepat dan menjadikan halaman web menjadi responsif.

JavaScript memiliki struktur sederhana, kodenya dapat disisipkan pada dokumen HTML atau berdiri sebagai satu kesatuan aplikasi. Struktur penulisan JavaScript adalah sebagai berikut.

```
<script language = “javascript”>
```

```
<!--
```

Penulisan kode javascript

```
//-->
```

```
</script>
```

### 2.2.7. Bahasa Pemrograman HTML5

Berdasarkan penelitian Yatini (2014: 3) HTML 5 merupakan teknologi inti dari Internet adalah bahasa *markup* untuk penataan dan penyajian konten *world wide web*. Tujuan utama HTML 5 adalah meningkatkan bahasa dengan dukungan multimedia yang tetap mudah dibaca dan dimengerti.

Fitur baru HTML 5 antara lain.

1. Unsur `<canvas>` untuk menggambar 2D.
2. Unsur `<video>` dan `<audio>` untuk media pemutaran.
3. Dukungan untuk penyimpanan lokal.
4. Konten baru dengan elemen spesifik seperti `<article>`, `<footer>`, `<header>`, `<nav>`, `<section>`.

### 2.2.8. MySQL

Berdasarkan penelitian Reza, dkk (2013: 3) MySQL Merupakan jenis *software* yang cukup populer, karena memiliki kehandalan, kecepatan serta kemudahan dalam penggunaannya. Beberapa perusahaan besar seperti Google, Yahoo atau Youtube juga merupakan pengguna dari perangkat lunak ini karena memiliki kelebihan antara lain:

1. Dapat mengeksekusi dengan cepat dalam menangani query sederhana.
2. Memiliki beberapa lapisan keamanan dan izin hak user secara mendetail.
3. Mampu menangani basis data dalam jumlah skala besar.

Berdasarkan penelitian Watung, dkk (2014: 2) MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management*

*system*) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. 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, dan masih banyak lagi.
2. Open Source. MySQL distribusikan secara *open source*, dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.
3. ‘Multiuser’. MySQL dapat digunakan oleh beberapa user 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. Jenis Kolom. MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed / unsigned integer, float, double, char, text, date, timestamp*, dan lain-lain.
6. Perintah dan Fungsi. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (*query*).
7. Keamanan. MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta sandi terenkripsi.

8. Skalabilitas dan Pembatasan. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu 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 klien menggunakan protokol TCP/IP, Unix soket (UNIX), atau *Named Pipes* (NT).
10. Lokalisasi. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.
11. Antar Muka. MySQL memiliki interface (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).
12. Klien dan Peralatan. MySQL dilengkapi dengan berbagai peralatan (*tool*) 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 PostgreSQL ataupun Oracle.

### 2.2.9. CSS

Berdasarkan penelitian Reza, dkk (2013: 3) *Cascading Style Sheets*, atau CSS, adalah cara yang direkomendasikan untuk mengendalikan lapisan presentasi dalam dokumen *web*. Keuntungan dari CSS adalah, lebih mengendalikan presentasi HTML yang disimpan terpisah dari konten. Sebagai contoh, mungkin untuk menyimpan semua gaya presentasi untuk situs web 10.000 halaman dalam sebuah file CSS tunggal. CSS juga menyediakan kontrol yang jauh lebih baik atas presentasi dari sebuah halaman situs daripada melakukan modifikasi elemen presentasi dalam HTML.