

## BAB II TINJAUAN PUSTAKA

### 2.1 Penelitian Terkait

Dalam penelitian ini, penulis terinspirasi dan mereferensi dari penelitian-penelitian sebelumnya yang berkaitan dengan latar belakang masalah pada skripsi ini. Berikut adalah penelitian terdahulu yang berhubungan dengan skripsi ini antara lain:

Penelitian yang dilakukan oleh (Wardana, 2015), "Aplikasi Penjualan Pakaian Anak Online Berbasis Web Pada Toko Amanda". Memberikan hasil akhir aplikasi *online* untuk proses penjualan pada toko yang diharapkan dapat menunjang pemasaran dan dapat membangun *brand building* sehingga customer mengenal produk yang dipasarkan. Metode penulisan yang digunakan yaitu analisa dan perancangan berorientasi objek.

Penelitian yang dilakukan oleh (Effendi, 2017), "Aplikasi Penjualan Pakaian di Toko MJB Collection Berbasis Web". Hasil dari penelitian ini dapat membantu konsumen dalam mengakses informasi produk yang dijual serta melakukan pemesanan produk, mengimplementasikan sistem informasi yang meliputi implementasi perangkat lunak, perangkat keras, basis data dan antarmuka dari aplikasi yang dihasilkan.

Penelitian yang dilakukan oleh (Hasyim, 2012), "Sistem Informasi Penjualan Pakaian Jadi Pada Distro Blindwear Bandung". Dapat mempermudah kinerja *administrator* ataupun pemilik dalam proses pengolahan data karena sudah mempunyai *database* yang terintegrasi. Pada aplikasi ini tidak tersedia kontak *admin* dan fitur *chat* sehingga sulit untuk customer dalam menanyakan informasi detail atau kesediaan produk.

Tabel 2.1 Penelitian Terkait

No	Penulis	Keterangan
1	Yoga Kusuma Wardana (2015)	<p>- Ada 5 (lima) level pengguna pada aplikasi ini yaitu <i>owner</i>, <i>adm&amp;finance</i>, <i>cashier</i>, <i>marketing</i> dan <i>customer</i>.</p> <p>- Hasil dari penulisan ini berupa aplikasi <i>online</i> sebagai bentuk peralihan proses penjualan pada Toko Hobi Amanda agar meningkatkan profit serta efisiensi waktu dalam pembuatan laporan penjualan. Penulisan ini menggunakan metode analisa dan perancangan berorientasi objek, dimulai dari analisa sistem berjalan menggunakan UML, elisitasi kebutuhan sistem, serta penggambaran sistem yang diusulkan melalui UML.</p>
2	Mohammad Efendi (2017)	<p>- Pengguna atau <i>user</i> dari aplikasi ini adalah pemilik dari toko <i>mjb collection</i>, bagian penjualan, bagian produksi dan member.</p> <p>- Hasil penelitian menunjukkan bahwa perancangan sistem informasi penjualan dan pemesanan berbasis web dapat membantu konsumen dalam mengakses informasi mengenai produk yang dijual dan dalam melakukan pemesanan produk, mengimplementasikan sistem informasi yang meliputi implementasi perangkat lunak, perangkat keras, basis data serta antarmuka dari aplikasi yang dihasilkan. Selain itu, manfaat lain dari sistem yang dibangun ini adalah produsen dapat menentukan Harga Pokok Penjualan (HPP) dari setiap barang yang dijual berdasarkan harga bahan, biaya produksi, dan biaya pengeluaran yang lain.</p>
3	Faisal Ibnu Hasyim (2012)	<p>- Pengguna dari sistem ini adalah pemilik distro <i>blindwear bandung</i>, bagian kasir, bagian gudang dan konsumen.</p> <p>- Sistem informasi penjualan ini menjadi metode alternatif dalam melakukan promosi dan transaksi pembelian yang lebih mudah, efektif dan efisien. Sistem ini terfokus pada user (pelanggan) dan admin. Fitur yang dapat diakses oleh</p>

No	Penulis	Keterangan
		pelanggan setelah melakukan pendaftaran dan <i>login</i> yaitu melakukan proses pemesanan. Pembuatan sistem penjualan ini memudahkan pelanggan dalam melakukan proses pembelian serta memudahkan pemilik toko dalam melakukan promosi produknya lebih efektif dan efisien.

**Tabel 2.2** Penelitian yang Dilakukan

No	Penulis	Judul	Keterangan
1	Nur Rizki Fachrullah	Aplikasi Penjualan Pakaian <i>Vintage Second Hand</i> Berbasis E- <i>Commerce</i> Pada Wakaiko.90s	<ul style="list-style-type: none"> <li>- Pengguna aplikasi yaitu admin dari <i>owner</i> Wakaiko.90s itu sendiri yang mengelola setiap produk dan mengkonfirmasi orderan didalam sistem dan <i>customer</i> yang melakukan pemesanan pada aplikasi.</li> <li>- Aplikasi dikhususkan untuk admin dari pihak Wakaiko.90s dalam mengelola penjualan produk serta mengelola produk-produk di dalam sistem.</li> <li>- Aplikasi memiliki kategori untuk memilah kumpulan pakaian <i>vintage</i> sesuai tahun produksi.</li> <li>- Terdapat opsi fitur <i>chat</i> pada sistem yang akan mengarahkan customer ke WhatsApp Admin Wakaiko.90s untuk memudahkan komunikasi antara <i>customer</i> dengan admin Wakaiko.90s.</li> <li>- Terdapat fitur melihat <i>tracking</i> produk yang dipesan melalui detail pengiriman di aplikasi.</li> </ul>

### **2.1.1 Kajian Pustaka**

### **2.1.2 Vintage**

#### **2.1.2.1 Pengertian Vintage**

Menurut kamus Oxford kata “*vintage*” itu dapat diartikan sebagai “*old and of very high quality*”. Jika diartikan secara global, *vintage* bisa diartikan sebagai barang-barang yang diproduksi di masa lampau ataupun masa kini, tetapi dianggap memiliki model klasik dan antik di masa sekarang. Seiring perkembangannya, pemaknaan *vintage* sendiri lebih akrab bila dikaitkan dengan dunia *fashion*.

Menurut Katherine Sorrell (2011), *vintage* style tidak terpaku pada satu gaya desain atau style, tetapi Vintage style adalah penggabungan dua atau lebih style yang berbeda menjadi satu. Dimana style atau gaya desain yang digunakan mencakup gaya desain abad 18, Georgian, Victorian, Art Deco, 1950s, 1960s, 1970s, dan contemporary dipadukan juga dengan “Global” styles seperti English Country, American Shakers, Scandivanian dan Ethic (Sorrell, 2011).

#### **2.1.2.2 Jenis-Jenis Pakaian Vintage**

Berikut ciri-ciri umum dari pakaian *vintage*:

##### **1. Build Up**

Yaitu pakaian *vintage* yang dimana tidak memiliki jahitan samping dari bagian bawah lengan hingga paling bawah bagian pinggang.

##### **2. Tahun Produksi**

Pada umumnya pakaian *vintage* memiliki tahun produksi pakaian, semakin lama umur pakaian tersebut, maka semakin tinggi nilai atau harga pakaian tersebut. Pakaian yang sudah berumur lebih dari 15 tahun dapat dikategorikan kedalam pakaian *vintage*.



Gambar 2.1 Tahun Produksi/ Keluaran Pakaian Vintage

### 3. Single Stich

*Single Stich* dapat diartikan sebagai jahitan tunggal. Pakaian yang hanya memiliki satu jahitan pada bagian ujung lengan dan bagian ujung bawah juga dapat dikategorikan sebagai pakaian *vintage*. Pakaian-pakaian tersebut biasanya diproduksi pada tahun 90an kebawah.

### 4. Tag Pakaian

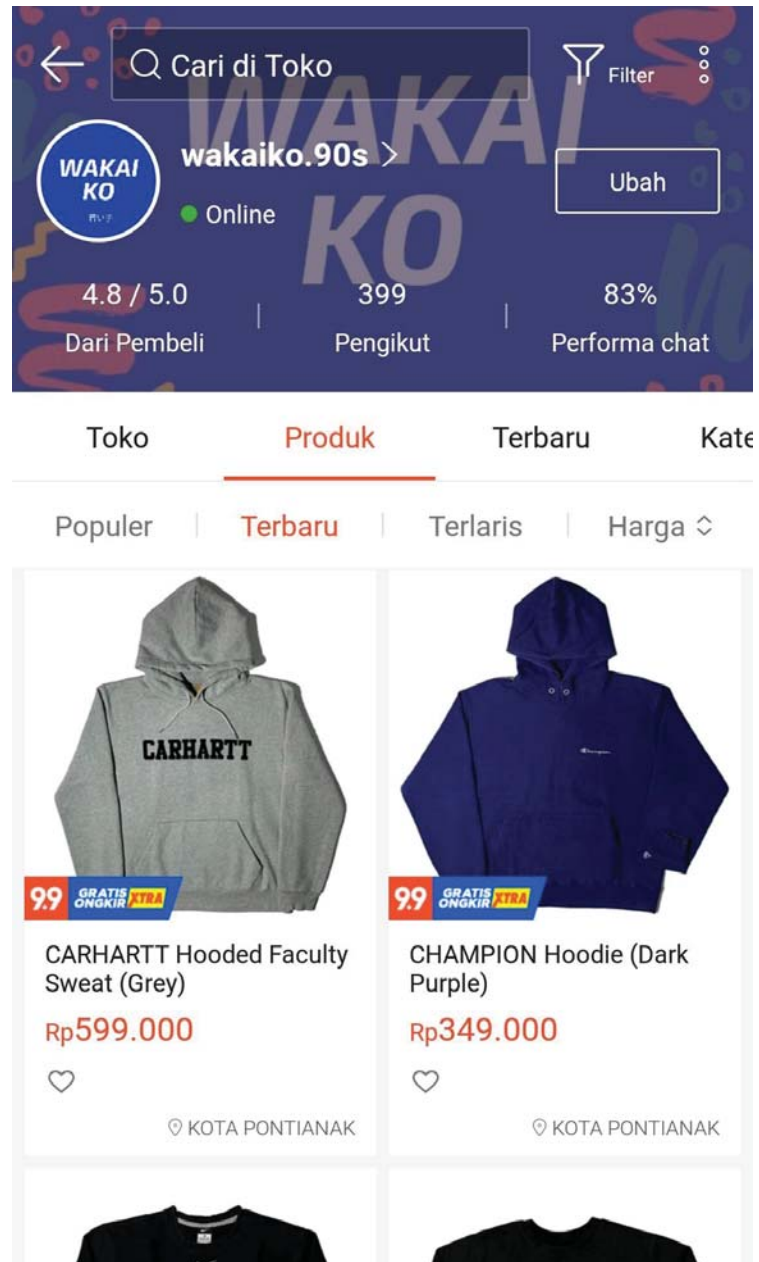
*Tag* Pakaian dapat dilihat pada bagian dalam kerah pakaian. Dengan melihat *tag* pakaian, kita dapat mengetahui apakah pakaian tersebut termasuk pakaian *vintage* atau tidak. *Tag* pakaian ini berperan penting untuk mengetahui *value* dari pakaian *vintage* tersebut. Berikut adalah contoh *tag* dari pakaian *vintage champion reverse weave*:



Gambar 2.2 Tag Pakaian Vintage Champion Reverse Weave

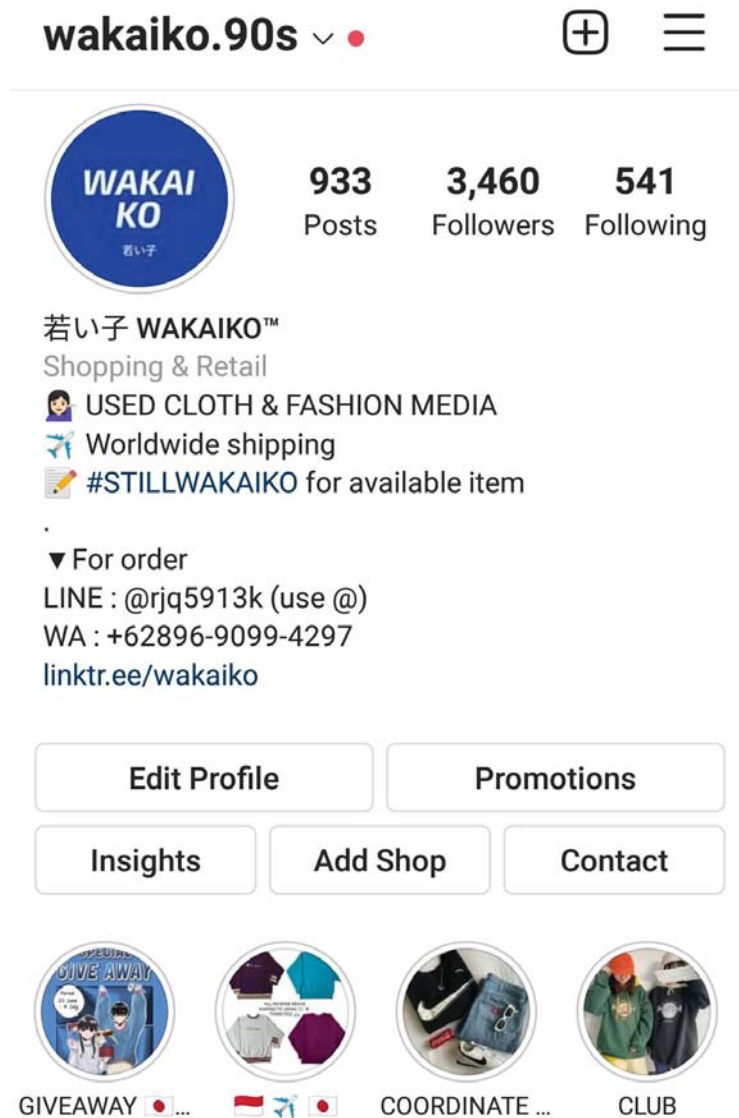
### 2.1.3 Profil Wakaiko.90s

Berikut adalah profil atau gambaran pada penjualan *online store* Wakaiko.90s pada sosial media dan *market place* meliputi penjualan melalui shopee, profil Instagram dan postingan instagram.



**Gambar 2.3** Penjualan melalui Shopee

Berdasarkan gambar 2.3 dapat dilihat bahwa Wakaiko.90s menjual beberapa produk pada *market place* shopee. Terdapat informasi pada *market place* Shopee seperti nama produk, harga, detail produk, lokasi dan informasi lainnya.



**Gambar 2.4** Profil Instagram

Berdasarkan gambar 2.4 dapat dijelaskan terdapat Profil akun Instagram Wakaiko.90s. Pada profil tersebut mencakup informasi mengenai *online store* Wakaiko.90s seperti worldwide shipping dalam artian bisa mengirim pakaian kesemua negara, lalu terdapat hastag “#STILLWAKAIKO” yaitu barang-barang yang masih tersedia/ belum terjual, lalu terdapat informasi kontak melalui aplikasi Line dan Whatsapp dan ada link untuk mempermudah customer dalam mengakses informasi serta pembelian.





**Gambar 2.5** Postingan Instagram

Berdasarkan gambar 2.5 dapat dilihat terdapat beberapa postingan pada akun Instagram Wakaiko.90s. Di sosial media Instagram Wakaiko.90s selain menjual produk juga terdapat postingan sebagai referensi berpakaian dengan menggunakan pakaian bekas/ *second hand*.



## **2.1.4 E-Commerce**

### **2.1.4.1 Pengertian E-Commerce**

Menurut Husda (2012:169) *E-commerce* adalah kegiatan-kegiatan bisnis yang menyangkut Konsumen (*Consumers*), Manufaktur (*Manufactures*), *Service Providers* dan pedagang perantara (*intermediaries*) dengan menggunakan jaringan-jaringan komputer (*Computer Networks*) yaitu internet.

Menurut Ding (2005:120) dalam Bukunya *E-commerce: Law & Practice*, mengemukakan bahwa *E-commerce* sebagai suatu konsep yang tidak didefinisikan. *E-commerce* memiliki arti yang berbeda bagi orang yang berbeda.

Jadi *E-commerce* digunakan sebagai transaksi bisnis antara perusahaan yang satu dengan perusahaan yang lain, antara perusahaan dengan pelanggan (*customer*), atau antara perusahaan dengan institusi yang bergerak dalam pelayanan public.

### **2.1.4.2 Tipe-Tipe Aplikasi E-Commerce**

#### 1. *Electronic Markets (EMs)*

Ems adalah sebuah sarana yang menggunakan teknologi informasi dan komunikasi untuk melakukan/menyajikan penawaran dalam sebuah segmen pasar, sehingga pembeli dapat membandingkan berbagai macam harga yang ditawarkan.

#### 2. *Electronic Data Interchange (EDI)*

EDI adalah sarana untuk mengefisienkan pertukaran data transaksi-transaksi regular yang berulang dalam jumlah besar antara organisasi-organisasi komersial.

#### 3. *Internet Commerce*

Internet Commerce adalah penggunaan internet yang berbasis teknologi informasi dan komunikasi untuk perdagangan. Kegiatan komersial ini seperti iklan dalam penjualan produk dan jasa.

### **2.1.5 Metode Prototype**

Metode penelitian sistem yang digunakan dalam perancangan sistem ini menggunakan model prototype dikarenakan pada metode ini pengembang dan user dapat saling berinteraksi selama proses pembuatan system. Menurut Yurindra (2017:47) “prototype adalah suatu proses yang memungkinkan developer membuat sebuah model software, metode ini baik digunakan apabila client tidak bisa

memberikan informasi yang maksimal mengenai kebutuhan yang diinginkannya.” Adapun gambaran metode penelitian yang digunakan dapat dilihat pada gambar dibawah ini:



**Gambar 2.6** Tahapan Metode Prototype

## 2.1.6 Alat Bantu Perancangan Sistem

### 2.1.6.1 Data Flow Diagram (DFD)

DFD merupakan suatu cara atau metode untuk membuat rancangan sebuah sistem yang mana berorientasi pada alur data yang bergerak pada sebuah sistem nantinya. Data Flow Diagram (DFD) adalah sebuah alat yang menggambarkan aliran data sampai sebuah sistem selesai, dan kerja atau proses dilakukan dalam sistem tersebut (Indrajani, 2011: 11).

Ada beberapa simbol yang digunakan pada DFD : (Jogiyanto, 2001).

#### 1. Kesatuan Luar (*External Entity*)

Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lain yang berada pada lingkungan luarnya yang memberikan input atau menerima output dari sistem.

#### 2. Arus Data (*Data Flow*)

Arus Data (*data flow*) di DFD diberi simbol suatu panah. Arus data ini mengalir di antara proses, simpan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

#### 3. Proses (*Process*)


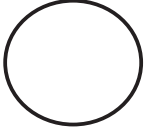
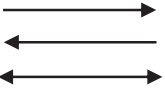

Proses (*process*) menunjukkan pada bagian yang mengubah input menjadi output, yaitu menunjukkan bagaimana satu atau lebih input diubah menjadi beberapa output. Setiap proses mempunyai nama, nama dari proses ini menunjukkan apa yang

dikerjakan proses.

#### 4. Simpanan Data (*Data Store*)

Data Store merupakan simpanan dari data yang dapat berupa suatu file atau database pada sistem komputer

**Tabel 2.3** Simbol DFD



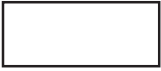
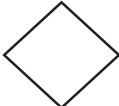



Simbol DFD	Nama
	Entitas / Kesatuan luar
	Proses
	Arus data / Aliran data
	Data store / Simpanan data

#### 2.1.6.2 *Flowchart*

*Flowchart* adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi.

*Flowchart* merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempermudah penyelesaian masalah yang khususnya perlu dipelajari dan dievaluasi lebih lanjut (Indrajani, 2011: 22).

Tabel 2.4 Simbol Flowchart

Simbol <i>Flowchart</i>	Keterangan
Terminal	Memulai dan Mengakhiri suatu proses
	
Input / Output	Digunakan untuk proses data maupun input data
	
Proses	Menampilkan suatu proses
	
Keputusan	Untuk menyeleksi kondisi didalam program. ( percabangan )
	
Persiapan	Digunakan untuk pemberian nilai awal suatu besaran
	
Proses pendefinisian	Menunjukkan suatu operasi yang rinciannya ditunjukkan ditempat lain
	
Garis alir	Garis suatu proses atau menunjukkan arus proses yang berjalan
	

### 2.1.6.3 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* (ERD) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. ERD menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data dengan menggunakan kardinalitas relasi. Kardinalitas relasi menunjukkan jumlah maksimum tupel yang dapat berelasi dngan entitas pada entitas yang lain (Bin Ladjamudin, 2013).

Terdapat 3 macam kardinalitas relasi yaitu sebagai berikut:

1. *One to One* (1:1)

Setiap anggota entitas A hanya boleh berhubungan dengan satu anggota entitas B, begitu pula sebaliknya.


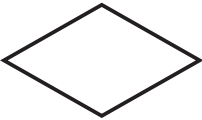


2. *One to many* (1:M / *Many*)

Setiap anggota entitas A dapat berhubungan dengan lebih dari satu anggota entitas B tetapi tidak sebaliknya.

3. *Many to Many* (M:M)

Setiap entitas A dapat berhubungan dengan banyak entitas himpunan entitas B dan demikian pula sebaliknya.

**Tabel 2.5** Simbol ERD

Simbol ERD	Keterangan
	Entitas : Memberikan identitas yang mewakili suatu objek dan dapat dibedakan dengan objek yang lain.
	Relasi : Menunjukkan hubungan diantara sejumlah entitas yang berbeda.
	Atribut : Properti yang dimiliki oleh suatu entitas, dimana dapat mendeskripsikan karakteristik dari entitas tersebut.
	Alur : Garis yang memiliki fungsi untuk menghubungkan atribut dengan entitas dan entitas dengan relasi.

#### 2.1.6.4 Kamus Data (*Data Dictionary*)

Kamus data merupakan tempat penyimpanan dari aliran-aliran data, *file* dan proses-proses dalam sebuah sistem. Kamus data digunakan untuk menjelaskan semua data yang mengalir atau digunakan dalam sistem, yaitu mengenai arus data yang masuk ke dalam sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.

Kamus Data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. *Data dictionary* tidak menggunakan notasi grafik sebagaimana halnya *data flow diagram*. Kamus data berfungsi untuk membantu user agar dapat mengerti aplikasi secara rinci (Sutabri, 2004).

### **2.1.7 Tools Pemograman**

#### **2.1.7.1 Hypertext Markup Language (HTML)**

HTML atau *Hypertext Markup Language* merupakan bahasa yang paling umum digunakan dalam pembangunan web. Sesuai dengan namanya, markup language yang berarti bahasa penandaan, HTML digunakan untuk menandai secara khusus bagian – bagian yang terdapat dalam sebuah halaman web. Penandaan pada setiap bagian menggunakan *tag* (Duckett, 2011).

Pada penggunaan *tag* HTML tanda kurung siku “< >” untuk memulai sebuah *code* serta adanya kata atau huruf di dalamnya. Dalam sebuah struktur file HTML pada setiap bagian menandakan yang mana *header, title, body, paragraph, tabel*, dan lainnya. Untuk memulai menulis sebuah kode HTML, bisa menggunakan program aplikasi *text editor* seperti *Notepad, Sublime Text, Adobe Dreamweaver*, dan sebagainya.

#### **2.1.7.2 Cascading Style Sheet (CSS)**

Kode pemrograman yang bertujuan untuk menghias dan mengatur gaya tampilan/layout halaman web supaya lebih menarik dan terlihat elegan. *Cascading Style Sheet* atau lebih dikenal dengan CSS dalam pemrograman web berfungsi untuk mengatur *style* dari halaman *web* (Duckett, 2011).

Dalam penulisan kode CSS memerlukan sebuah *rule* yang dapat diatur sesuai keinginan *programmer*. *Rule* dibagi menjadi dua bagian :

1. *Selector*

Menunjuk atau memilih elemen mana yang akan diberikan *rule* CSS.

2. *Declaration*

Menetapkan bagaimana sebuah elemen yang dipilih akan ditata. *Declaration* dibagi menjadi dua bagian yang dipisahkan dengan tanda titik dua “;” yaitu :

a. *Property*

Digunakan untuk memilih properti apa yang akan digunakan untuk memberikan efek kepada elemen yang telah dipilih.

b. *Value*

Memberikan nilai secara spesifik bagi *property* yang telah dipilih.

Oleh karena itu jika file HTML tidak memiliki *rule* CSS, maka tampilan yang diberikan HTML akan tampak tidak rapi atau kurang beraturan. Jika diterapkan *rule* CSS pada *elemen* atau *tag* pada file HTML maka tampilan yang diberikan menjadi lebih baik.

### 2.1.7.3 *Javascript*

*JavaScript* merupakan bahasa pemrograman yang digunakan dalam pemrograman web (Duckett, 2011).

Penulisan *code JavaScript* ditandai dengan *tag* ‘<script type=”JavaScript”></script>’ pada halaman *HTML*. *Code JavaScript* dapat dimasukkan secara internal ataupun eksternal.

Penulisan secara internal cukup dengan memasukan *code* di antara *tag script*. Sedangkan secara eksternal, *code* berupa file yang disimpan berlainan folder ataupun dari *link* internet dituliskan seperti berikut: <script type = ”JavaScript” src = ”scripts/jquery.js”> </script>.

Hal yang didapat jika menggunakan *JavaScript*:

1. Membaca dan menuliskan (*read & write*) elemen dan teks;
2. Memanipulasi atau memindahkan teks;
3. Menjalankan fungsi perhitungan pada data;
4. Bereaksi terhadap *events*, seperti saat pengguna menekan sebuah *button*;
5. Membaca waktu dan tanggal di komputer pengguna;
6. Menetapkan ukuran tampilan beserta resolusi layar dan web browser *version* pada komputer pengguna;
7. Melakukan pengecekan pada *input* yang diberikan oleh pengguna, seperti pada pengecekan *form validation*.



#### 2.1.7.4 Hypertext Processor (PHP)

PHP (*Hypertext Processor*) adalah bahasa pemrograman yang memungkinkan para *web developer* untuk membuat aplikasi *web* yang dinamis dengan cepat dan mudah. PHP dirintis dan diperkenalkan pertama kali sekitar tahun 1994 oleh Rasmus Lerdorf melalui situsnya untuk mengetahui siapa saja yang telah mengakses ringkasan *online*-nya (Gunawan, 2010).

PHP disebut bahasa pemrograman *server-side* karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman *client-side* seperti JavaScript yang diproses pada *web browser* (client). PHP merupakan bahasa pemrograman yang dapat digunakan dalam teknik pemrograman terstruktur maupun teknik pemrograman berorientasi objek (OOP).

PHP dirancang untuk membentuk *web* dinamis, gunanya membentuk suatu tampilan berdasarkan permintaan terkini. Untuk sebuah *web* jika menggunakan bahasa pemrograman PHP maka akan menjadi lebih dinamis dan rapi. kode PHP biasanya di sisipkan kedalam file HTML, karena itu PHP disebut juga sebagai *Scripting Language* atau bahasa pemrograman *script*.

PHP memiliki kelebihan dibanding dengan bahasa *scripting* lainnya. Berikut ini kelebihannya (Luke, Welling & Laura, 2009):

1. *Performance*: PHP dapat bekerja cepat dengan performa yang sangat baik;
2. *Scalability*: Skala komoditas yang besar;
3. Integrasi ke *database*: PHP memiliki *native connection* yang tersedia untuk berbagai macam *database system*;
4. *Built-in libraries*: PHP memiliki banyak fungsi *built-in* untuk melakukan tugas yang berhubungan dengan *web task*;
5. *Low cost*: PHP merupakan bahasa *scripting open source*, sehingga siapapun dapat menggunakannya secara gratis;
6. *Ease of learning and use*: PHP mudah untuk dipelajari. Bahasa pemrograman PHP didasari pada bahasa pemrograman pada umumnya, seperti C, C++, Perl, dan Java;
7. *Strong object-oriented (OOP) support*: PHP versi 5 sudah didukung dengan fitur OOP;

8. *Flexibility of development approach:* PHP memberikan kemudahan bagi *developer* dalam implementasi. PHP sangat mudah untuk beradaptasi dengan *framework* seperti CodeIgniter;
9. *Availability of source code:* *Developer* memiliki akses langsung dengan *source code* PHP;
10. *Availability of support and documentation:* PHP memberikan dokumentasi yang jelas dan tersedia komunitas yang siap membantu jika *developer* mengalami kesulitan dalam menggunakan PHP.

#### **2.1.7.5 Bootstrap**

*Bootstrap* merupakan *framework* untuk membangun desain web secara responsif. Artinya, tampilan web yang dibuat oleh *bootstrap* akan menyesuaikan ukuran layar dan browser yang kita gunakan baik di desktop, tablet ataupun mobile device. Dengan *bootstrap* kita juga bisa membangun web dinamis ataupun statis (Alatas, 2013).

#### **2.1.7.6 XAMPP**

XAMPP adalah paket program *web* lengkap yang dapat dipakai untuk belajar pemrograman *web*, khususnya PHP dan MySQL (Nugroho, 2013). XAMPP adalah perangkat lunak bebas yang mendukung banyak sistem operasi. XAMPP merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*) yang terdiri dari program Apache HTTP Server, MySQL *database* dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X(empat sistem operasi apapun), Apache, MySQL, PHP, dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas. XAMPP merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.

Bagian penting XAMPP yang biasa digunakan pada umumnya:

1. XAMPP *Control Panel Application* berfungsi mengelola layanan (*services*) XAMPP, seperti mengaktifkan layanan (*start*) dan menghentikan layanan (*stop*);
2. Htdocs adalah *folder* tempat meletakkan berkas-berkas yang akan dijalankan;
3. PHPMyAdmin merupakan bagian untuk mengelola *database*.

### 2.1.7.7 MySQL

MySQL adalah sebuah *relational database server* yang sangat cepat dan dapat menyesuaikan dengan kebutuhan pengguna. MySQL merupakan sebuah basis data yang dapat menyimpan, mencari, melakukan pengurutan, dan mengambil data (Luke, Welling & Laura, 2009).

Server MySQL mengontrol akses data sehingga dapat diakses oleh banyak pengguna secara bersamaan, menyediakan akses yang cepat, dan memastikan hanya pengguna yang terotorisasi yang dapat mengakses basis data. MySQL menggunakan SQL (*Standard Library Language*).

MySQL dan PHP merupakan sistem yang saling terintegrasi. Maksudnya adalah pembuatan database dengan menggunakan sintak PHP dapat di buat. Sedangkan input yang di masukkan melalui aplikasi web yang menggunakan script serverside seperti PHP dapat langsung dimasukkan ke database MySQL yang ada di server dan tentunya web tersebut berada di sebuah web server.

### 2.1.7.8 PhpMyAdmin

PhpMyAdmin adalah sebuah aplikasi open source yang berfungsi untuk memudahkan manajemen MySQL. Dengan menggunakan phpMyAdmin, user dapat membuat database, membuat tabel, memasukkan, menghapus dan mengupdate data dengan GUI dan terasa lebih mudah tanpa perlu mengetikkan perintah SQL secara manual.

Beberapa fitur dalam phpMyAdmin :

- 1) Antarmuka berbasis web;
- 2) Impor data dari CSV dan SQL;
- 3) Ekspor data dari berbagai format;
- 4) Membuat grafik PDF dari tampilan dan basis data user;
- 5) Membuat kompleks query menggunakan *Query-by-example* (QBE);
- 6) Transformasi data disimpan ke dalam format yang menggunakan satu set yang telah ditetapkan seperti menampilkan data blob-data atau download-link.

### 2.1.7.9 Basis Data

Basis Data adalah kumpulan data (elementer) yang secara logika berkaitan dalam merepresentasikan fenomena/fakta secara terstruktur dalam domain tertentu untuk mendukung aplikasi pada sistem tertentu. Basis data adalah kumpulan data

yang saling berhubungan yang merefleksikan fakta-fakta yang terdapat di organisasi (Hariyanto, 2004).

*Database* atau basis data adalah kumpulan data yang disimpan secara sistematis di dalam komputer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur dan batasan-batasan data yang akan disimpan. Basis data merupakan aspek penting dalam sistem informasi dimana basis data merupakan gudang penyimpanan data yang akan diolah lebih lanjut. Basis data menjadi penting karena dapat menghindari duplikasi data, hubungan antar data yang tidak jelas, organisasi data dan juga update yang rumit.

Proses memasukkan dan mengambil data ke dan dari media penyimpanan data memerlukan perangkat lunak yang disebut dengan sistem manajemen basis data (*Database Management System / DBMS*). Tujuan utama dari DBMS adalah untuk memberikan tinjauan abstrak data kepada user (pengguna). Jadi sistem menyembunyikan informasi tentang bagaimana data disimpan, dipelihara dan tetap dapat diambil (akses) secara efisien.

#### **2.1.7.10 SQL Injection**

*SQL injection* merupakan salah satu teknik hacking dengan cara memanipulasi sintaks SQL. Pada sql injection dilakukan dengan memasukan perintah – perintah yang digunakan dalam *database* melalui *url*. Pada dasarnya *SQL Injection* merupakan cara mengeksploitasi celah keamanan yang muncul pada level atau “layer” *database* dan aplikasinya. Celah keamanan tersebut ditunjukkan pada saat penyerang memasukkan nilai “string” dan karakter-karakter contoh lainnya yang ada dalam instruksi SQL. Dikatakan sebagai sebuah “injeksi” karena aktivitas penyerangan dilakukan dengan cara “memasukkan” string (kumpulan karakter) khusus untuk melewati filter logika hak akses pada website atau sistem komputer yang dimaksud (Zam, 2015).

### **2.1.8 Metode Pengujian Sistem**

#### **2.1.8.1 Black Box Testing**

*Black box Testing* atau pengujian *black box* merupakan metode uji coba yang memfokuskan pada keperluan fungsional dari software. Karna itu uji coba *blackbox* memungkinkan pengembang software untuk membuat himpunan kondisi

input yang akan melatih seluruh syarat-syarat fungsional suatu program.

*Black box testing* berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineers* untuk memperoleh set kondisi *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program (Pressman, 2010).

*Black box testing* berfokus pada spesifikasi fungsional dari perangkat lunak, *tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program.

*Black box testing* berusaha untuk menemukan kesalahan dalam kategori berikut:

1. Fungsi yang tidak benar atau fungsi yang hilang;
2. Kesalahan antarmuka;
3. Kesalahan dalam struktur data atau akses *database* eksternal;
4. Kesalahan perilaku (*behavior*) atau kesalahan kinerja;
5. Inisialisasi dan pemutusan kesalahan.

Kategori error yang akan diketahui melalui *black box testing* :

1. Fungsi yang hilang atau tak benar;
2. Error dari antar-muka;
3. Error dari struktur data atau akses eksternal database;
4. Error dari kinerja atau tingkah laku;
5. Error dari inisialisasi dan terminasi.

#### **2.1.8.2 User Acceptance Test (UAT)**

Pengujian UAT atau Uji Penerimaan Pengguna adalah suatu proses pengujian oleh pengguna yang dimaksudkan untuk menghasilkan dokumen yang dijadikan bukti bahwa *software* yang telah dikembangkan telah dapat diterima oleh pengguna, apabila hasil pengujian (*testing*) sudah bisa dianggap memenuhi kebutuhan dari pengguna.

*User Acceptance Testing* merupakan pengujian yang dilakukan oleh *end-user* dimana *user* tersebut adalah *staff*/karyawan perusahaan yang langsung berinteraksi dengan sistem dan dilakukan verifikasi apakah fungsi yang ada telah berjalan sesuai dengan kebutuhan/fungsinya (Perry, 2006).