

## **BAB II**

### **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

Pada saat perkembangan informasi sudah sangat luas seperti sekarang ini definisi penjualan memiliki arti yang berbeda-beda menurut para ahli adapun definisi umum penjualan adalah sebuah usaha atau langkah konkrit yang dilakukan untuk memindahkan suatu produk, baik itu berupa barang ataupun jasa, dari produsen kepada konsumen sebagai sasarannya. Adapun beberapa penelitian – penelitian yang sudah dilakukan terdahulu bisa dijadikan bahan acuan serta memperoleh perbandingan-perbandingan yang sesuai dengan topik yang di teliti. Beberapa penelitian yang sudah dilakukan yang berkaitan dengan penjualan *online* diantaranya adalah seperti di bawah ini.

Khalifar Avi Shena (2017) melakukan penelitian yang berjudul “Perancangan dan Implementasi *E-Commerce* Pada Toko Mgear Jember” penelitian ini bertujuan bisa memenuhi kebutuhan dan bisa memudahkan penjual dalam menjalankan kegiatan bisnis seperti penjualan, pembelian, promosi barang baru, pembayaran dan transaksi jarak jauh pada toko mgear jember. Dari sistem yang telah dibuat tersebut masih memiliki kelemahan, karena belum terdapat fitur laporan penjualan agar toko memiliki gambaran tentang kondisi penjualan perusahaan secara nyata.

Rizal, Bayu Setiaji (2017) melakukan penelitian dan membuat sebuah jurnal yang berjudul “Perancangan Website *E-Commerce* Pada Toko Sepatu Angga Sport Menggunakan Framework Codeigniter”. Tujuan penelitian ini sendiri adalah dapat menginformasikan dan menawarkan produk secara global. Sehingga toko lebih cepat maju dan berkembang.

Dari sistem yang telah dibuat tersebut masih memiliki kelemahan, karena belum terdapat fitur laporan penjualan agar toko memiliki gambaran untuk tolak ukur untuk melihat usaha yang dijalankan berjalan dengan baik.

Pratama, Bayu Kaesar (2012) melakukan penelitian yang berjudul “Membangun Aplikasi E-Commerce Dengan Menggunakan Framework Codeigniter” (Studi Kasus Caesar Distro)” Tujuan penelitian ini sendiri adalah memberikan kemudahan untuk pelanggan dan admin pada saat order barang di Caesar Distro. Dari sistem yang telah dibuat tersebut masih memiliki kelemahan, karena belum terdapat fitur laporan penjualan agar toko memiliki gambaran tentang kondisi penjualan perusahaan secara nyata.

**Tabel 2. 1** Perbandingan Literatur

<b>Penulis, Judul</b>	<b>Tujuan dan Metode</b>	<b>Hasil dan kesimpulan</b>
Khalifar Avi Shena (2017), Perancangan dan Implementasi <i>E-Commerce</i> Pada Toko Mgear Jember	Memenuhi kebutuhan dan bisa memudahkan penjual dalam menjalankan kegiatan bisnis seperti penjualan, pembelian, promosi barang baru, pembayaran dan transaksi jarak jauh pada toko mgear jember. Bahasa pemrograman menggunakan php dan database mysql.	Sistem penjualan ini menyajikan informasi terkait dengan toko Mgear Outdoor yang ditujukan pada konsumen, perlunya penambahan laporan penjualan pada toko.

<b>Penulis, Judul</b>	<b>Tujuan dan Metode</b>	<b>Hasil dan kesimpulan</b>
Rizal, Bayu Setiaji (2017), Perancangan Website E-Commerce Pada Toko Sepatu Angga Sport Menggunakan Framework Codeigniter	Dapat menginformasikan dan menawarkan produk secara global. Bahasa pemrograman menggunakan php dan database mysql dan menggunakan framework codeigniter	Menghasilkan website pada toko sepatu angga sport, stok di bagian keranjang belanja belum bisa tampil, belum adanya laporan penjualan
Pratama, Bayu Kaesar (2012), Membangun Aplikasi E-Commerce Dengan Menggunakan Framework Codeigniter (Studi Kasus Caesar Distro)	Memberikan kemudahan untuk pelanggan dan admin pada saat order barang di Caesar Distro. Bahasa pemrograman menggunakan php dan database mysql dan menggunakan framework codeigniter	Sistem ini dapat memberitahukan kepada member atau pengunjung mengenai detail barang, belum adanya laporan penjualan

Adapun perbedaan pada penelitian ini yaitu pada penelitian sebelumnya dalam sistem yang dibuat peneliti masih belum adanya fitur laporan bulanan. Sedangkan dalam sistem yang baru dalam skripsi ini sudah memiliki fitur laporan bulanan sehingga Toko Gemilang memiliki arsip laporan bulanan.

## **2.2 Landasan Teori**

### **2.2.1 Pengertian Website**

*Website* adalah sebuah halaman yang menyajikan informasi baik dalam bentuk tulisan, gambar, suara, atau video yang dihubungkan dengan jaringan halaman atau *hyperlink*. *Website* sering juga disebut sebagai *web*, *site*, situs, atau situs *web*. *Website* adalah metode untuk menampilkan informasi di *internet*, baik

itu berupa teks, gambar, video & suara maupun interaktif memiliki keuntungan yang menghubungkan (*link*) dari dokumen dengan dokumen lainnya (*hypertext*) yang dapat diakses melalui *browser* (Yuhefizar, 2008).

### **2.2.2 Pengertian Sistem Informasi**

Sistem informasi adalah komponen yang saling berkaitan dan bekerja sama yang bertujuan untuk mengumpulkan, mengolah, menyimpan, dan menampilkan informasi untuk mendukung dalam pengambilan keputusan, koordinasi, pengaturan, analisa, dan visualisasi pada sebuah organisasi (Laudon, 2012, p. 16). Sistem informasi adalah data yang dikumpulkan, dikelompokkan dan diolah yang bertujuan untuk menjadikan suatu informasi yang berharga bagi yang menerimanya (Muhyuzir, 2001, p. 8).

### **2.2.3 Sistem Informasi Berbasis Website**

Sistem informasi berbasis *web* adalah suatu sistem didalam sebuah organisasi yang mendukung operasi, bersifat managerial dan beroperasi pada sebuah *browser* aplikasi dan teknologi internet (Whitten, 2004).

### **2.2.4 Pengertian E-Commerce**

*E-Commerce* didefinisikan sebagai kegiatan menjual barang dagangan atau jasa melalui *internet*. Seluruh komponen yang terlibat dalam bisnis praktis diaplikasikan disini, seperti *customer service*, produk yang tersedia, cara pembayaran, jaminan atas produk yang dijual, cara promosi dan sebagainya (Net). *E-Commerce* adalah penggunaan jaringan komputer untuk melakukan komunikasi bisnis dan transaksi komersial (Gao, 2005).

### **2.2.5 Codeigniter**

(Upton, 2009, p. 7) *CodeIgniter* adalah *powerful open source PHP framework* yang mudah dikuasai, dibangun untuk *PHP programmers* yang membutuhkan *toolkit* sederhana dan baik untuk membuat *full-featured web applications*. *CodeIgniter* adalah *MVC framework* yang di *design* untuk mempermudah penggunaannya. Saat ini, *CodeIgniter* dikembangkan dan dimaintain

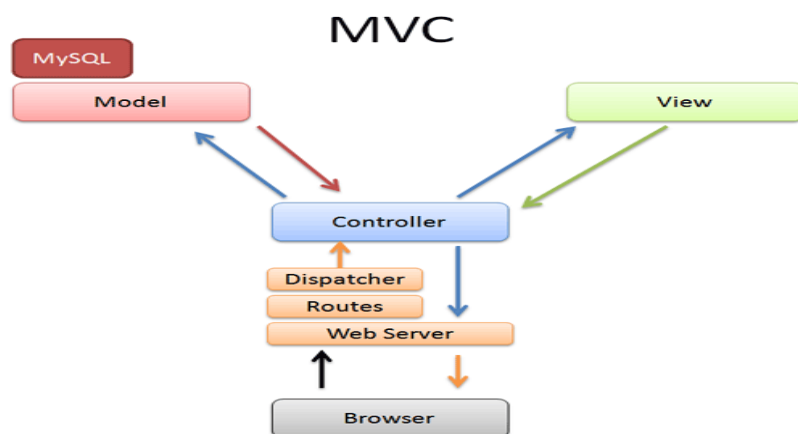
oleh *Expression Engine Development Team*. Adapun beberapa keuntungan menggunakan *CodeIgniter* antara lain :

- Berukuran sangat kecil, *File download* nya hanya sekitar 2MB dan sudah *include* dokumentasi yang sangat lengkap.
- CodeIgniter* menggunakan konsep *MVC* yang memungkinkan pemisahan *layer application-logic* dan *presentation*.
- CodeIgniter* mempunyai *library* yang lengkap untuk mengerjakan operasi-operasi yang umum dibutuhkan oleh sebuah aplikasi berbasis *web*, misalnya mengakses *database*, mengirim *email*, memvalidasi *form*, menangani *session* dan sebagainya.

### 2.2.6 Desain MVC Codeigniter

*CodeIgniter* adalah *framework PHP* yang dibuat berdasarkan kaidah *Model-View-Controller (MVC)*. Secara sederhana *CodeIgniter* akan memisahkan komponen-komponen didalam pengkodean aplikasi berbasis *web*, sehingga diharapkan nantinya lebih mudah untuk dikelola (Hakim, 2010, p. 4).

Adapun alur berjalannya aplikasi berbasis *framework CodeIgniter* dapat dilihat pada gambar berikut:



**Gambar 2. 1** Alur Berjalannya Aplikasi Berbasis *Framework CodeIgniter*

Pengertian gambar diatas adalah ketika ada sebuah *user request* datang maka *user request* ditangani oleh *controller*, setelah itu *controller* memanggil

*model* jika diperlukan operasi *database*. Hasil eksekusi *query* yang terdapat pada *model* kemudian dikembalikan ke *controller*. *Controller* memanggil *view* yang tepat dan mengkombinasikannya dengan hasil eksekusi *query* yang terdapat pada *model*. Hasil dari operasi kemudian akan ditampilkan pada *browser*.

### **2.2.7 Web Browser**

Menurut (Irawan, 2011, p. 3) “*Web Browser* adalah aplikasi yang digunakan untuk menampilkan halaman *web* beserta kontennya”. Beberapa aplikasi *browser* yang dapat digunakan antara lain *Internet Explorer*, *Firefox*, *Chrome*, dan *Opera*. *Web browser* yang paling terkenal penggunaannya adalah *Internet Explorer* dan *Mozilla Firefox*. *Web browser* adalah sebuah perangkat lunak yang memungkinkan pengguna untuk menampilkan serta berinteraksi dengan *text*, gambar, dan informasi yang ada pada halaman *web* di *World Wide Web (WWW)* atau *local area network (LAN)*, (Sufandi, 2007).

### **2.2.8 XAMPP**

*XAMPP* adalah perangkat lunak *open source* dan salah satu *web server* lokal yang paling banyak digunakan oleh para pengembang dan desainer situs web dalam membuat situs *web offline*. Program ini sudah mendukung banyak sistem operasi dari beberapa program. *XAMPP* adalah paket program *web* lengkap yang dapat anda pakai untuk belajar pemrograman *web*, khususnya *PHP* dan *MySQL*, (Nugroho, 2013, p. 1).

### **2.2.9 PHP (Personal Home Page Hypertext Processor)**

Menurut (Valade, 2004, p. 9) *PHP* akronim dari *Hypertext Preprocessor* adalah *open source* yang banyak digunakan sebagai tujuan utama *scripting language*. Di desain untuk digunakan pada pengembangan *website*. *PHP* berawal dari *personal home page tools*, yang di kembangkan oleh Rasmus Lerdorf untuk membantu user dengan *web page tasks*. *PHP* dibuktikan sangat berguna dan populer serta secara bertahap berkembang untuk menjadi *full-featured language*.

### 2.2.10 MySQL

*MySQL* (Junaedi, 2005, p. 151) adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelolaan datanya.

*MySQL* merupakan *Relational Database System (RDBMS)*, *RDBMS* adalah program yang memungkinkan pengguna *database* untuk membuat, mengelola dan menggunakan data pada *model relational* sehingga tabel-tabel yang terdapat pada *database* akan mempunyai relasi antar satu tabel dengan tabel lainnya.

Di bawah ini merupakan fitur-fitur yang terdapat dalam *MySQL*:

1. *Relational Database System(RDBMS)*
2. Arsitektur *Client-Server*. *MySQL* memiliki arsitektur *client-server* dimana *server database MySQL* terinstal di *server*. *Client MySQL* dapat berada di komputer yang sama dengan *server*, dan dapat juga di komputer lain yang berkomunikasi dengan *server* melalui jaringan bahkan *internet*.
3. Mengenal perintah standar *SQL*
4. Dapat menangani basis data dalam skala besar. Basis data dalam *MySQL server* dapat berisi 50 juta *record*.
5. *Client* dapat terkoneksi ke *MySQL server* menggunakan socket *TCP/IP* pada *platform* manapun.

## 2.3 Perancangan Sistem

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

*UML* adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, (Shalahuddin, 2016, p. 133). Menurut (Booch, 2005), *UML* merupakan suatu bahasa. Suatu bahasa terdiri dari kata-kata, dan memiliki aturan untuk menggabungkan kata-kata tersebut, sehingga tercipta komunikasi. Sebuah permodelan bahasa seperti *UML* telah menjadi bahasa standar untuk merencanakan suatu aplikasi.

UML yang digunakan untuk *website* toko gemilang ini sendiri yaitu *use case diagram*, *activity diagram*, *sequence diagram* dan *class diagram*.

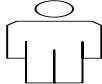
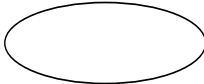


### 2.3.2 Use Case Diagram

*Use Case* adalah diagram yang digunakan untuk menggambarkan interaksi beberapa *actor* serta komponen komponennya untuk memperkenalkan suatu sistem yang akan dibuat.

*Use Case* (Rosa. A.S. d. S., 2014, p. 155) 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. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

komponen-komponen yang ada dalam *use case* diagram dapat dilihat pada tabel 2.2.

**Tabel 2. 2** Komponen-Komponen Use Case Diagram

Simbol	Keterangan
	<i>Actor</i> : Menggambarkan yang berinteraksi dengan sistem yang akan dibuat
	<i>Use Case</i> : Digunakan untuk berinteraksi atau bertukar pesan antar unit maupun aktor
	<i>Assosiation</i> : Relasi yang digunakan untuk menggambarkan interaksi antara <i>use case</i> dan aktor.
	<i>Generalization</i> : Relasi yang menggambarkan pewarisan sifat baik aktor maupun <i>use case</i> .




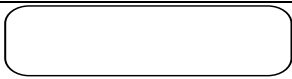


Simbol	Keterangan
<<include>>	<i>Include</i> : Menggambarkan jalannya suatu <i>use case</i> memicu jalannya use case lain
-- <<extend>> ->	<i>Extend</i> : Menggambarkan bahwa suatu <i>use case</i> dijalankan karena ada persyaratan tertentu dari use case lain jika suatu kondisi sudah terpenuhi


### 2.3.3 Activity Diagram

*Activity diagram* adalah diagram yang digunakan untuk menggambarkan aktivitas yang dilakukan pada sistem yang akan di buat.

*Activity diagram* menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak (Rosa. A.S. d. S., Rekayasa Perangkat Lunak, 2014, p. 161) komponen-komponen yang ada dalam *activity diagram* diagram dapat dilihat pada tabel 2.3.

**Tabel 2. 3** Komponen-Komponen *Activity Diagram*

Simbol	Keterangan
	<i>Activity</i> : Menunjukkan masing-masing kelas saling berinteraksi satu sama lain
	<i>Action</i> : State dari sistem yang mencerminkan eksekusi dari suatu aksi
	<i>Initial Node</i> : Bagaimana proses suatu objek diawali
	<i>Final Node</i> : Bagaimana proses suatu objek diakhiri

Simbol	Keterangan
	<i>Join Node</i> : Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.




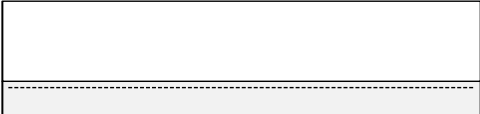
### 2.3.4 Class Diagram

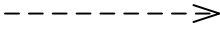

*Class diagram* adalah diagram yang digunakan untuk menggambarkan *class-class* yang ada dalam sistem dan relas-relasi yang ada didalamnya yang akan dibuat untuk membangun sistem.

Menurut (Rosa. A.S. d. S., 2014, p. 141), *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

komponen-komponen yang ada dalam *class diagram* dapat dilihat pada tabel 2.4.

**Tabel 2. 4** Komponen-Komponen Class Diagram.

Simbol	Keterangan
	<i>Generalization</i> : Hubungan dimana anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
	<i>Association</i> : Sesuatu yang menghubungkan antara satu objek dengan objek lainnya.
	<i>Realization</i> : Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Class</i> : Himpunan dari berbagai objek-objek yang berbagi atribut serta operasi yang sama.

Simbol	Keterangan
	<p><i>Depedency</i> : Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung pada elemen yang tidak mandiri.</p>
	<p><i>Collaboration</i> : Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.</p>

### 2.3.5 ERD (Entity Relationship Diagram)

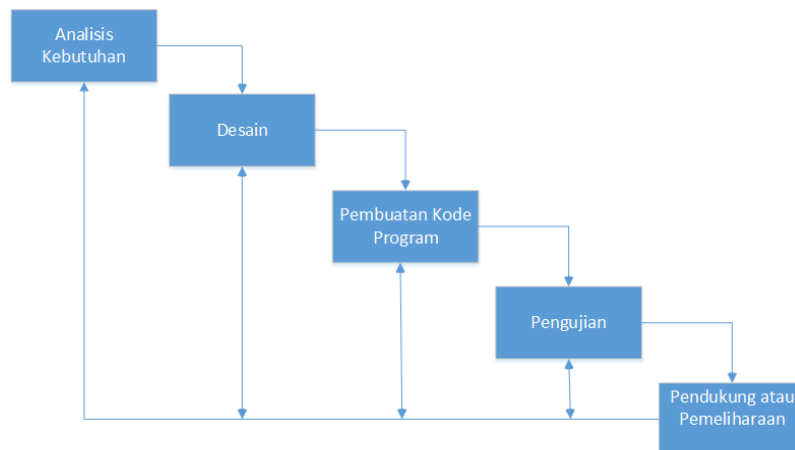
*ERD* (*Entity Relationship Diagram*) merupakan sebuah permodelan *basis data* yang di maksudkan untuk menjelaskan hubungan antar *data* yang terdapat dalam sebuah *basis data* berdasarkan objek-objek dasar yang mempunyai hubungan antar relasi.

Menurut (Begg, 2010, pp. 371-395), *Entity Relationship Diagram* digunakan untuk menggambarkan struktur *logical database* dalam bentuk *diagram* ERD, serta menyediakan cara yang sederhana dan mudah untuk memahami bagian berbagai komponen dalam desain *database*. ERD memiliki 3 komponen yaitu *entity*, *relationship* dan *atribut*.

## 2.4 Metode Pengembangan Sistem

### 2.4.1 Metode Waterfall

Metode pengembangan sistem *waterfall* sering juga disebut model sekuensial linier (*sequential linier*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau urut dimulai dari analisis, desain, pengkodean, pengujian dan tahap support (M, 2011).



**Gambar 2. 2** Metode *Waterfall*

Dibawah ini merupakan penjelasan urutan metode *waterfall*

a) Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk memesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

b) Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c) Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d) Pengujian

Pengujian fokus pada perangkat lunak secara segi logik dan

fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

e) Pendukung (support) atau pemeliharaan (maintenance)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari tahap analisis spesifikasi untuk perubahan perangkat lunak baru.