

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Penelitian sebelumnya, yang serupa pernah dilakukan oleh Zia Rizki Saputri, Anzani Nur Oktavia, Lis Saumi Ramdhani, dan Acep Suherman dengan judul “Rancang Bangun Sistem Informasi Pemesanan Makanan Berbasis *Web* Pada *Cafe* Surabiku”. Penelitian ini bertujuan untuk mengatasi masalah dalam pelayanan pemesanan sehingga dapat memberikan kepuasan terhadap konsumen. Pada penelitian ini, mereka menggunakan metode *waterfall*, untuk *database* menggunakan Mysql [3].

Penelitian berikutnya yang dilakukan oleh Tanjung, dan Sitepu, pada penelitian ini bertujuan untuk mendukung terwujudnya sebuah aplikasi yang mampu mengembangkan usaha pada Toko YT. *Wall Interior*. Peneliti membuat aplikasi berbasis *web* dan android yang nantinya akan berisi fitur-fitur semacam jenis wallpaper yang ditawarkan, berapa stok barang yang ditoko, dan cara pemesanannya. Sehingga dari tersedianya aplikasi tersebut mampu membantu customer dalam mengetahui informasi yang ditawarkan pada Toko YT. *Wall Interior*. Pada penelitian ini menggunakan metode *waterfall*, kemudian PHP sebagai Bahasa pemrograman nya untuk database menggunakan MySql [4].

Penelitian lain, oleh Hafizh Alfiansyah, Iguh Widipangestu, Dede Supriyadi, dengan judul “Perancangan Sistem Informasi Penjualan Makanan dan Minuman Berbasis *Web* dengan Metode *Extreme Programming* (Studi Kasus *Cafe Gematos Pamulang*)”. Penelitian yang dilakukan diakibatkan oleh pengunjung yang mengantri, oleh karena itu dibuatlah suatu *web* untuk mengatasinya, dengan menggunakan metode *extreme programming* [5]. Alur pada proses nya menggunakan UML yaitu dengan *use case*, *activity diagram*, *sequence diagram*, *class diagram* dan ERD. *Web Gematos* dibuat menggunakan *Sublime Text* dan *Php MyAdmin* sebagai *database*, dan diuji dengan menggunakan *Black Box* untuk mengetahui apakah applikasi sesuai dengan yang di inginkan atau tidak.

Penelitian berikutnya ada dari Masya, dan Lusti dengan judul “Analisa Perancangan Sistem Informasi Pemesanan pada *Wedding Organizer* Berbasis *Web*” [6]. *Website Wedding Planner* Berbasis *Web*

ini dapat memberikan kemudahan dalam pemesanan paket secara *online*, baik paket yang telah ditentukan maupun pilihan secara manual dan otomatis. Dengan adanya fitur-fitur serta konten yang membantu *sharing* informasi antara calon pengantin dengan pihak *Wedding Organizer*. Sistem yang dibuat ini menggunakan Bahasa pemrograman PHP, kemudian MySQL sebagai pengelola *database* nya.

Penelitian lain telah dilakukan oleh Hernawan, Septiansyah, Deddy, dan Prehanto yang meneliti pada *cafe KopiRide* yang dimana KopiRide masih menggunakan sistem lama dengan membuat buku daftar menu makanan dan minuman, dimana dibutuhkan seorang pegawai untuk mengantarkan atau menawarkan daftar menu yang ada. Dalam sistem pemesanan dengan daftar menu buku membuat pelanggan mengalami ketidak nyamanan dalam proses pemesanan menu karena takutnya pelanggan dalam penularan virus. Kemudian dibuatlah sistem pemesanan yang baru menggunakan *Simple Additive Weighting (SAW)* [7]. Sistem yang dibuat juga menggunakan *gateway payment* untuk pembayarannya untuk mengurangi kontak fisik yang terjadi antara pembeli dengan kasir, serta metode yang digunakan merupakan *Waterfall*, yang mana banyak juga digunakan untuk pengembangan software lain nya.

Berdasarkan dari semua kajian penelitian sebelumnya, persamaan penelitian yang akan dilakukan yakni sistem mencakup tentang pemesanan, kemudian dalam penggunaan bahasa pemrograman sama menggunakan PHP, dan mysql sebagai basis datanya. Sedangkan perbedaan dengan sistem yang akan dibuat, dimana di sistem yang akan di buat akan menggunakan *framework Codeigniter 4* dalam membangun sistem nya, serta penggunaan *Javascript* untuk membuat hasil *website* yang lebih dinamis,. Kemudian untuk fiturnya terdapat total harga dari pesanan yang dipesan oleh pelanggan, yang akan memudahkan pelanggan untuk melakukan kegiatan pemesanan. Pelanggan juga bisa melihat ketersediaan dari menu yang ada. Kemudian juga akan diberikan struk berupa pesanan yang dipesan, yang didalamnya terdapat jumlah pesanan dan total jumlah pembayaran yang harus dibayarkan. Serta juga dapat membuat laporan penjualan secara otomatis berdasarkan *filter* waktu tertentu.

Dari penelitian penelitian diatas dirancanglah tabel perbandingan agar menjadi sebuah titik banding atau perbedaan yang jelas. Berikut disajikan tabel perbandingan antara penelitian sebelumnya dengan penelitian yang sekarang dilakukan.

Tabel 2. 1 Tabel Perbandingan Penelitian Terdahulu

No	Nama Peneliti	Judul	Metode	Tahun	Hasil Penelitian	Perbedaan
1	Zia Rizki Sapuri, Anzani Nur Oktavia, Lis Saumi Ramdhani, Acep Suherman	Rancang Bangun Sistem Informasi Pemesanan Makanan Berbasis <i>Web</i> Pada <i>Cafe</i> Surabaya	<i>Waterfall</i>	2019	Hasil dari penelitian ini berupa <i>Website</i> sistem pemesanan berupa makanan dan minuman, dan mengelola ketersediaan dari menu, dan dapat menghasilkan laporan	1. Menggunakan php native. 2. Pada analisa kebutuhan penelitiannya, untuk aktor pelanggan terdapat perbedaan dimana pelanggan diharuskan mempunyai akun
2	Tanjung, Sitepu	Rancang Bangun Aplikasi Pemesanan dan Penjualan Berbasis <i>Web</i> dan Android Pada Toko YT. Wall Interior	<i>Waterfall</i>	2020	Hasil dari penelitian adalah sistem pemesanan dan penjualan berbasis <i>website</i> dan android.	1. Produk yang ditawarkan berbeda, pada penelitian sebelumnya, produknya berupa interior. 2. Pada penelitian ini, menghasilkan sistem yang berbasis android saja.

No	Nama Peneliti	Judul	Metode	Tahun	Hasil Penelitian	Perbedaan
3	Hatfzh Alfansyah, Iguh Widipangestu, Dede Supriyadi	Perancangan Sistem Informasi Penjualan Makanan dan Minuman Berbasis <i>Web</i> dengan Metode Extreme Programming (Studi Kasus <i>Cafe Gematos Pamulang</i>)	<i>Extreme Programming</i>	2022	Hasil dari penelitian adalah sistem penjualan berbasis <i>website</i> .	1. Metode yang digunakan pada penelitian sebelumnya menggunakan metode extreme programming.

No	Nama Peneliti	Judul	Metode	Tahun	Hasil Penelitian	Perbedaan
4	Masya, Lusiti	Analisa Perancangan Sistem Informasi Pemesanan pada Wedding Organizer Berbasis Web	Tidak tercantum didalam jurnal	2020	Hasil dari penelitian ini yaitu sebuah sistem pemesanan berbasis <i>website</i> yang hanya digunakan untuk kegiatan pemesanan paket <i>wedding</i> .	<ol style="list-style-type: none"> 1. Kegiatan yang dilakukan didalam sistem pada penelitian sebelumnya hanya sebatas untuk memesan saja. 2. Didalam penelitian sebelumnya pemesanan diwajibkan untuk mendaftar akun..

No	Nama Peneliti	Judul	Metode	Tahun	Hasil Penelitian	Perbedaan
5	Hernawan, Septiansyah, Dedy, Prehanto	Rancang Bangun Sistem Pemesanan Menu <i>Cafe</i> “KopiRide” menggunakan Metode SAW dan QR-Code berbasis Android	<i>Waterfall</i>	2021	Hasil dari penelitian ini adalah sistem pemesanan yang hanya bisa di akses dengan perangkat android, yang mana menggunakan QR-Code	1. Pada penelitian sebelumnya menggunakan SAW untuk perhitungannya. 2. Pada sistem KopiRide sistem dapat diakses hanya terbatas pada perangkat android.

2.2 Landasan Teori

2.2.1 Sistem Informasi

Secara sederhana sistem merupakan suatu kumpulan dari unsur, variabel, komponen yang terorganisir, saling berinteraksi, saling terhubung dan bergantung satu sama lain.

Sistem informasi adalah suatu sistem yang dibuat yang terdiri dari komponen-komponen di dalam organisasi untuk menyajikan informasi. Sistem informasi juga didefinisikan sebagai sekumpulan prosedur dalam organisasi yang dilaksanakan akan memberikan informasi untuk pengambilan keputusan. Sistem informasi menerima *input* data dan perintah, mengelola data sesuai dengan perintah, dan *output* yang dihasilkan. Komponen dari sistem informasi yaitu [8]:

- a. *Hardware* (Perangkat Keras), terdiri dari komputer, printer dan komponen perangkat keras lainnya.
- b. *Software* (Perangkat Lunak), merupakan kumpulan perintah-perintah yang ditulis sesuai dengan aturan tertentu untuk memberi perintah kepada komputer untuk melaksanakan tugas tertentu.
- c. Data, merupakan alat yang menjadi penghubung antara manusia dengan perangkat terjadinya proses pengolahan data.
- d. Prosedur, panduan dan kebijakan untuk mengoperasikan suatu sistem.

2.2.2 Pemesanan

Pemesanan merupakan perjanjian antara dua belah pihak atau lebih untuk memenuhi kebutuhannya dalam mengusahakan barang maupun jasa tersebut sehingga dapat dipakai [9].

2.2.3 Pemrograman Berorientasi Objek



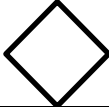

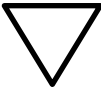



Pemrograman berorientasi objek adalah cara berpikir dan logika baru untuk menangani masalah yang diselesaikan oleh komputer, di mana setiap objek merupakan unit individu dengan kombinasi struktur data dan fungsi tertentu [10].

2.2.4 Flowchart

Flowchart (Diagram Alir) merupakan bagan yang mengarahkan alir di dalam prosedur sistem secara logika. *Flowchart* merupakan cara dalam menjelaskan tahap-tahap pemecahan masalah dengan merepresentasikan simbol-simbol tertentu yang mudah digunakan dan

dipahami [8]. Berikut Tabel 2.2 merupakan simbol-simbol yang ada didalam *flowchart*:

Tabel 2. 2 Simbol-Simbol *Flowchart*

No.	Simbol	Nama	Fungsi
1.		Proses	Digunakan untuk menunjukkan pengolahan yang dilakukan komputer
2.		<i>Manual Operation</i>	Digunakan untuk menunjukkan pengolahan yang tidak dilakukan komputer
3.		<i>Decision</i>	Digunakan untuk memilih proses berdasarkan kondisi
5.		Terminal	Digunakan untuk memulai atau mengakhiri proses.
6.		<i>Offline Storage</i>	Digunakan untuk memberitahukan bahwa data akan disimpan ke suatu media tertentu
7.		Manual Input Simbol	Digunakan untuk input data secara manual dengan keyboard
8.		Input/Output	Digunakan untuk Input dan Output.
9.		Dokumen	Digunakan untuk data masukan dan keluaran dari dokumen






2.2.5 UML


UML (*Unified Modeling Language*) merupakan bahasa berupa grafik atau gambar yang berguna untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak berbasis *Object-Oriented* [10].

1. *Use case diagram*

Use case diagram merupakan pemodelan yang mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case diagram* digunakan untuk mengetahui fungsi-fungsi pada sistem informasi dan siapa saja yang berhak menggunakan fungsi tersebut. Aktor dan *use case* merupakan dua hal utama dalam sebuah *use case diagram* [11]. Simbol-simbol dalam *use case* dapat dilihat pada Tabel 2.3 dibawah ini.

Tabel 2. 3 Simbol-Simbol *Use case diagram*







No.	Simbol	Nama	Fungsi
1.		<i>Use case</i>	Digunakan sebagai unit yang saling bertukar pesan antar unit dan aktor.
2.		Aktor	Orang yang berinteraksi dengan sistem yang dibuat.
3.		Asosiasi	Penghubung komunikasi antar aktor dan <i>Use case</i>
4.		<i>Extend</i>	Relasi <i>Use case</i> yang menjadi penghubung komunikasi actor dan <i>Use case</i> .
5.		<i>Include</i>	Menunjukkan bahwa suatu <i>Use case</i> seluruhnya merupakan fungsionalitas dari <i>Use</i>


			<i>case</i> lainnya.
6.		Generalisasi	Menunjukkan spesialisasi actor untuk dapat berpartisipasi dengan <i>Use case</i> .

2. *Activity diagram*

Activity diagram yang merupakan bagian dari *Unified Modelling Language* yang digunakan untuk memberikan pemahaman aktivitas apa saja yang terjadi pada proses pemilihan rute, pilihan yang ada, dan letak perulangan yang dapat terjadi didalamnya [12]. Simbol *Activity diagram* ditunjukkan pada Tabel 2.4 berikut ini.

Tabel 2. 4 Simbol-Simbol *Activity diagram*


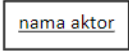

Notasi	Keterangan	Simbol
<i>Initial State</i>	Titik awal untuk memulai suatu aktivitas	
<i>Final State</i>	Titik akhir untuk mengakhiri suatu aktivitas	
<i>Activity</i>	Menandakan sebuah aktivitas	
<i>Decision Node</i>	Pilihan untuk pengambilan keputusan	
<i>Control Flow</i>	Arus Aktivitas	
<i>Fork Join</i>	Menunjukkan kegiatan menggabungkan dua panel activity	

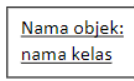

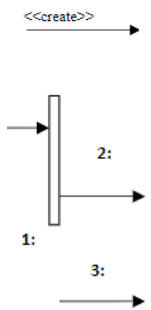
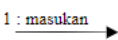
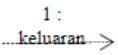
	menjadi satu atau satu panel activity menjadi dua	
<i>Note</i>	Catatan khusus untuk sebuah aktivitas	

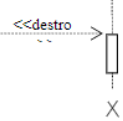



3. *Sequence diagram*

Sequence diagram berfokus terhadap perilaku didalam suatu sistem, yang mana mengilustrasikan bagaimana objek berinteraksi dengan objek lainnya. Didalam *sequence diagram* terdapat objek dan pesan yang dikirim antar objek. Biasanya *sequence diagram* berguna untuk menggambarkan interaksi objek yang terjadi dalam suatu *use case*. Dalam satu *use case* hanya diperlukan satu *sequence diagram*, jika terdapat beberapa skenario dalam *use case* maka bisa ilustasikan sebagai fragmen dalam *sequence diagram* [13]. Simbol-simbol dalam *sequence* dapat dilihat pada Tabel 2.5 berikut ini.

Tabel 2. 5 Simbol-Simbol *Sequence diagram*

No.	Simbol	Nama	Fungsi
1.	 	Aktor Tanpa Waktu aktif	Orang, pproses, atau sistem lain yang berinteraksi dengan sistem informas yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dari aktor adalah gambar orang, tapi actor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor
2.		Garis hidup / <i>Lifeline</i>	Menyatakan kehidupan suatu objek



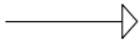
3.		Objek	Menyatakan objek yang berinteraksi pesan.
4.		Waktu Aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
5.		Pesan tipe <i>create</i>	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi maka operasi yang dipanggil harus ada pada <i>diagram</i> kelas sesuai dengan kelas objek yang berinteraksi.
6.		Pesan tipe <i>send</i>	Menyatakan bahwa suatu objek mengirimkan data ke objek lain, dan arah panah mengarah pada objek yang dikirim.
7.		Pesan tipe <i>return</i>	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah ke objek yang mengirim pesan.





8.		Pesan tipe <i>destroy</i>	Menyatakan objek mengakhiri hidup objek lain, arah panah mengarah ke objek yang diakhiri, jika ada create maka akan ada destroy.
9.		<i>Boundary</i>	Digunakan untuk menggambarkan sebuah <i>form</i> .
10.		<i>Control</i>	Digunakan untuk menghubungkan <i>Boundary</i> dengan table.
11.		<i>Entity Class</i>	Menggambarkan awal sistem yang menjadi sebuah landasan Menyusun basis data.

4. *Class Diagram*

Class diagram memperlihatkan struktur sistem dalam hal pendefinisian *class-class* yang akan digunakan untuk membangun sistem. Class memiliki atribut dan metode, yang juga disebut sebagai operasi. Dalam tabel 2.6 di bawah ini, terdapat simbol-simbol yang digunakan dalam *class diagram* simbol-simbol yang digunakan dalam *Class Diagram* [5].

Tabel 2. 6 Simbol-Simbol *Class Diagram*


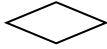

No.	Simbol	Nama	Fungsi
1.		<i>Class</i>	Kelas pada struktur sistem
2.		Antarmuka / <i>Interface</i>	Sama dengan konsep interface dalam pemrograman berbasis objek.
3.		Generalisasi	Relasi antar kelas dengan makna generalisasi (umum)


			khusus)
4.		Asosiasi Berarah	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas lain
5.		Asosiasi	Hubungan antar kelas dengan makna umum
6.		kebergantungan	Hubungan antar kelas dengan makna kebergantungan antar kelas.
7.		Agregasi	Hubungan antar kelas dengan makna semua bagian.

5. *Entity Relationship Diagram*

Entity Relationship Diagram atau ERD adalah alat perancangan untuk memodelkan suatu basis data. Tujuan dibuatnya *Entity Relationship Diagram* adalah untuk pemodelan yang menunjukkan hubungan antar data yang mempunyai relasi, juga mendokumentasikan data yang ada dengan cara menjelaskan tiap masing-masing data dan relasinya [5]. Simbol ERD ditunjukkan pada Tabel 2.7 berikut ini.

Tabel 2. 7 Simbol-Simbol *Entity Relationship Diagram*

No	Simbol	Nama	Keterangan
1		<i>Entitas</i>	Suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
2		Relasi	Menunjukkan adanya hubungan diantara sejumlah entitas yang berbeda.
3		<i>Association</i>	Penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.

4		Atribut	Mendeskripsikan karakter entitas (atribut yang berfungsi sebagai key yang diberi garis bawah).
---	---	---------	--

2.2.6 Rekayasa Perangkat Lunak

Rekayasa perangkat lunak adalah ilmu yang membahas mengenai aspek produksi perangkat lunak yang dimulai dari menganalisa kebutuhan untuk perangkat lunak [14], menganalisa dan menentukan spesifikasi dari kebutuhan pengguna, membuat desain, menulis kode atau *coding*, pengujian sampai melakukan pemeliharaan sistem setelah digunakan rekayasa perangkat lunak juga memiliki keterkaitan dengan proses produksi seperti anggaran produksi, SDM, penjadwalan dan lainnya.

A. Metode Pengembangan Sistem

Metode yang digunakan dalam pengembangan sistem ini adalah metode *waterfall* atau metode pendekatan klasik (*Classic Life Cycle*) atau "*Linear Sequential Model*" yang menggambarkan metode pengembangan perangkat lunak secara berurutan dan linier atau berurutan. Rangkaian aktivitas dalam metode ini disajikan dalam proses yang terpisah dengan 5 tahapan. Tahapan-tahapan pengembangan sistem yaitu [11]:

1. *Requirements Definition* (Analisis Kebutuhan)
Fase Analisa Kebutuhan adalah langkah awal yang dilakukan dengan tujuan untuk mengetahui kebutuhan yang berkaitan dengan sistem. Pada tahapan ini juga merupakan tahapan untuk merincikan baik kebutuhan data maupun kebutuhan pengguna yang diterapkan di dalam sistem.
2. *System and Software Design* (Perancangan)
Tahapan perancangan desain sistem yang didalamnya termasuk desain arsitektur desain secara keseluruhan. Perancangan desain bertujuan untuk membantu memberikan gambaran secara lengkap mengenai apa yang harus dikerjakan nantinya.
3. *Implementation and Unit Testing* (Coding)
Tahap implementasi perangkat lunak yang sesuai dengan rancangan yang telah dibuat fase sebelumnya. Pada tahap ini

rancangan sistem yang sebelumnya dibuat akan diubah menjadi bahasa yang mudah dipahami komputer.

4. *Integration and System Testing* (Pengujian)

Tahap ini merupakan tahap pengujian pada semua fungsi sistem perangkat lunak yang dihasilkan telah sesuai dengan keinginan pengguna dan perancangan sistem. Pengujian dilakukan juga dilakukan untuk memeriksa fungsi sistem berjalan dengan baik tanpa ditemukan kesalahan.

5. *Operation and Maintenance* (Perawatan)

Tahap ini merupakan tahap terakhir dalam proses pembuatan sistem, setelah tahap sebelumnya selesai dan kemudian sistem akan digunakan oleh pengguna. Perbaikan dan pengembangan sistem dilakukan untuk pemantauan kinerja kerja dari sistem yang dibuat.

Sistem ini menggunakan metode *Waterfall* untuk metode pengembangan sistem karena metode pengembangan ini dilakukan secara berurutan dan linier. Kualitas sistem yang dihasilkan akan lebih baik karena setiap tahap dilakukan secara bertahap. Karena tahap demi tahap yang dilalui harus menunggu tahap sebelumnya selesai dan saling berkaitan sehingga mengurangi terjadinya kesalahan

B. Metode *BlackBox Testing*

Pengujian *blackbox* merupakan metode pengujian perangkat lunak dimana penguji tidak mengetahui secara internal dari struktur sistem yang akan diuji. Pengujian dilakukan pada setiap bagian untuk mengetahui keberhasilan dari sistem yang dikembangkan. Metode pengujian ini difokuskan pada *input* dan *output* dari sistem apakah sistem tersebut layak untuk digunakan atau belum. Metode ini bertujuan untuk mengetahui apakah Penguji hanya menguji pada bagian *interface* sistem saja dan tidak perlu memiliki pengetahuan Bahasa Pemrograman yang digunakan [5].

2.2.7 PHP

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman yang digunakan secara luas untuk pembuatan dan pengembangan sebuah *web* dan biasa digunakan pada HTML. PHP adalah bahasa yang biasanya digunakan untuk komunikasi sisi server dan digunakan oleh

seorang programmer untuk mengembangkan sebuah *website* yang dinamis dan interaktif [14].

2.2.8 Basis Data (*Database*)

Basis data atau *database* adalah data yang dapat didesain dan berintegrasi sehingga dapat memenuhi kebutuhan dari pengguna [15]. *Database* mewakili entitas, atribut, dan hubungan logis antara entitas. Basis data terdiri dari kumpulan data yang terorganisir, relasi antar data, dan objektivinya.

DBMS (*Database Management System*) merupakan suatu perangkat lunak pengelola yang digunakan untuk menciptakan, mengakses, mendefinisikan dan merawat basis data. Tujuan DBMS adalah menyediakan lingkungan yang mudah dan aman [15].

SQL (*Structured Query Language*) merupakan salah satu produk RDBMS. RDBMS memiliki kepanjangan *Relational Database Management System*. Merupakan salah satu produk andalan yang dibuat oleh Microsoft yang berfungsi sebagai *realional database*. yaitu hubungan antar table yang berisi data data pada suatu *database* [15]

2.2.9 Framework

Framework merupakan sebuah *software* atau aplikasi yang bisa dibilang seperti kerangka kerja yang fungsinya untuk memudahkan *developer* dalam mengembangkan aplikasi *website* yang ada. Ada beragam jenis *framework* yang sering digunakan untuk membuat *website*. Setiap *framework* memiliki fungsinya masing-masing dan menggunakan bahasa pemrograman yang berbeda [2].

2.2.10 CodeIgniter

Codeigniter adalah sebuah *framework* bahasa pemrograman PHP yang bersifat *open source* dan menggunakan metode *model, view, controller* (MVC) yang berguna untuk memudahkan pengembang atau programmer dalam membangun sebuah sistem tanpa harus membuatnya dari awal[14].