

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Aplikasi dengan tema *Framework CodeIgniter* pernah disusun oleh Ahmad Muntoha (2016) Mahasiswa STMIK Akakom, dengan judul *Framework CodeIgniter untuk Membangun Web Komisariat STMIK Akakom Yogyakarta*, Ance Teresia Manseni (2016) Mahasiswa STMIK Akakom dengan judul *Implementasi Framework CodeIgniter pada E-crommerce Studi Kasus Toko Kain Batik Arfak Mandiri Papua Monokuari*, Eko Yuswanto (2018) Mahasiswa STMIK Akakom dengan judul *Implementasi Framework CodeIgniter Untuk Pengelolaan Nilai Di SMK ASH-Shiddiqiyah Balingasal*, Frentisno Libranto Tennis (2018) Mahasiswa STMIK Akakom dengan judul *Aplikasi Ujian Online Berbasis Web Menggunakan Teknologi Framework CodeIgniter* dan Epid Erdanosa Branata (2018) Mahasiswa STMIK Akakom dengan judul *Implementasi Framework Codeigniter Untuk Pemesanan Makanan Dan Minuman Berbasis Web*. Aplikasi-aplikasi tersebut dibuat dari hasil implementasi CodeIgniter untuk membantu pengembangan dalam pembuatan website, sedangkan pada penelitian kali ini akan membuat sebuah web dimana web tersebut dibuat dari hasil implementasi *framework* CodeIgniter dan jQuery Mobile.

Tabel 2.1 Data Penelitian Sebelumnya

Sumber	Topik	Teknologi	Bahasa	
			Pemrograman	Interface
Ahmad Muntoha (2016)	<i>Framework CodeIgniter untuk Membangun Web Komisariat STMIK Akakom Yogyakarta</i>	<i>Framework CodeIgniter</i>	<i>PHP (Hypertext Preprocessor)</i>	Web
Ance Teresia Manseni (2016)	<i>Implementasi Framework CodeIgniter pada E-crommerce Studi Kasus Toko Kain Batik Arfak Mandiri Papua Monokuari</i>	<i>Framework CodeIgnite r</i>	<i>PHP (Hypertext Preprocessor)</i>	Web
Eko Yuswanto (2018)	<i>Implementasi Framework CodeIgniter Untuk</i>	<i>Framework CodeIgniter</i>	<i>PHP (Hypertext Preprocessor)</i>	Web

	<i>Pengelolaan Nilai Di SMK ASH- Shiddiqiyah Balingasal</i>			
Frentisno Libranto Tenis (2018)	<i>Ujian Online Berbasis Web Menggunakan Teknologi Framework CodeIgniter</i>	<i>Framework Codeigniter</i>	<i>PHP (Hypertext Preprocessor)</i>	Web
Epid Erdanosa Branata (2018)	<i>Implementasi Framework Codeigniter Untuk Pemesanan Makanan Dan Minuman Berbasis Web</i>	<i>Framework CodeIgniter</i>	<i>PHP (Hypertext Preprocessor)</i>	Web
Usulan (2018)	<i>Impelmentasi Framework Codeigniter Dan JQuery Mobile dalam</i>	<i>Framework CodeIgniter Dan JQuery Mobile</i>	<i>PHP (Hypertext Preprocessor)</i>	Web

	<p><i>pengembangan</i></p> <p><i>website studi kasus</i></p> <p><i>website toko online</i></p>			
--	--	--	--	--

2.2 Dasar Teori

2.2.1 CodeIgniter

CodeIgniter (CI) adalah *framework* pengembangan aplikasi (*application development framework*) dengan menggunakan PHP, suatu kerangka untuk bekerja atau membuat program dengan menggunakan PHP yang lebih sistematis. Pemrograman tidak perlu membuat program dari awal lagi (*from scratch*), karena CI menyediakan sekumpulan pustaka yang diperlukan untuk menyelesaikan pekerjaan yang umum, dengan menggunakan antarmuka dan struktur logika yang sederhana untuk mengakses librarynya. Pemrograman dapat memfokuskan pada kode yang harus dibuat untuk menyelesaikan suatu pekerjaan.

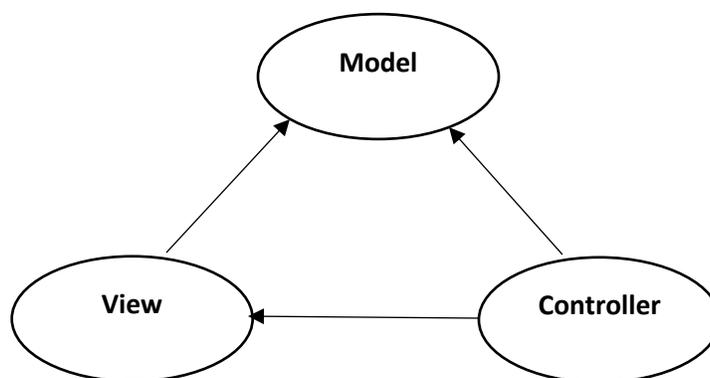
CI dapat digunakan dengan menggunakan *server web Apache* dan *MS IIS*. Hanya saja, tidak semua fitur dari CI dapat diimplementasi dengan sempurna di *MS IIS*, karena CI dirancang dan dikembangkan dengan acuan *apache*; contoh yang pasti adalah penggunaan file *.htaccess*, yang memang milik *apache*.

Secara umum, maka CI memberikan dan menyediakan:

1. Suatu kerangka kerja dengan jejak yang sedikit (*small footprint*)
2. Kinerja yang bagus
3. Kompatibilitas dengan ukuran hosting standar yang terpasang dengan perbedaan versi dan konfigurasi

4. Suatu kerangka kerja yang hampir tidak memerlukan konfigurasi
5. Suatu kerangka kerja yang tidak memerlukan kita menggunakan baris perintah (*command line*)
6. Suatu kerangka kerja yang tidak mengharuskan bekerja dengan aturan pengkodean yang mengikat/membatasi
7. Suatu aturan yang tidak mengharuskan kita menggunakan suatu bahasa templating (walaupun CI menyediakan suatu parser untuk template)
8. Cara untuk menghindarkan kompleksitas dalam membuat program, dengan menyediakan solusi sederhana
9. Suatu dokumentasi kerangka kerja yang menyeluruh dan jelas.

CI didasarkan pada pola pengembangan *Model-View-Controller*. MVC adalah sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. Dalam praktiknya, hal itu memungkinkan halaman web untuk memuat script kecil karena presentasi terpisah dari scripting PHP. Teknik pemrograman MVC (*Model View Controller*) merupakan teknik pemrograman yang populer saat ini, yang mengharapkan pemrograman secara disiplin untuk membagi program menjadi 3 bagian : *model*, *view*, dan *controller*, seperti gambar berikut :



Gambar 2.1 Komponen Dasar Framework

1. Model

Model berfungsi sebagai komponen dari program yang menggunakan teknik MVC, serta digunakan untuk menyediakan pemrosesan data. Sehingga objek *model* adalah bagian dari aplikasi yang mengimplementasikan logika untuk domain data aplikasi. Umumnya, objek *model* digunakan untuk mengambil data dari database atau menyimpan data ke dalam database.

Nama *model* dideklarasikan di dalam file *model*, diawali dengan pernyataan *class*, kemudian nama *model*, dan pernyataan *extends CI_Model*. Nama *model* harus sama dengan nama filenya atau sebaliknya, yang pasti adalah bahwa nama *model* dan nama file harus sama.

Contoh :

```
<?php
class namamodel extends CI_Model{
    [ deklarasi variable ]
    Function namafungsi_method ([ listparams]) {
    ..
    [ return nilai;)
    } .. }?>
```

Deklarasi variabel adalah daftar variabel yang diperlukan oleh *model*, bisa konstanta atau variabel yang diperlukan oleh semua fungsi atau method yang ada dalam *model*. Nama fungsi yang disediakan oleh *model* untuk pemrosesan atau menyediakan data untuk *controller* yang memanggil atau menggunakannya. *Listparams* adalah daftar parameter yang diperlukan oleh fungsi atau method untuk menghasilkan data yang diinginkan.

2. View

View adalah komponen yang menampilkan antarmuka untuk pengguna (*user interface, UI*) aplikasi. Antarmuka ini dibuat berdasarkan data dari *model*. *View* secara prinsip merupakan komponen dari program yang menggunakan teknik MVC, yang digunakan untuk menampilkan

hasil dari proses. *View* digunakan untuk menampilkan data hasil pemrosesan ke dalam format dokument HTML.

1) Aturan pembuatan view

Tidak seperti pada pembuatan *controller* dan *model*, maka pembuatan *view* tidak perlu menggunakan pernyataan *class* terlebih dahulu, karena *view* sebenarnya adalah dokument HTML. Yang digunakan sebagai template informasi luaran dari suatu program yang dibuat dengan menggunakan CI

2) Sintak Pembuatan *View* Dalam *Controller*

Sama seperti halnya *model*, maka *view* harus dilakukan pemuatan terlebih dahulu oleh *controller*, dengan perintah `$this->load->view(namaview)`.

3) Pemuatan *View* Dengan Data

Data yang akan ditampilkan di dalam suatu *view*, di dalam *controller* harus dimasukan sebuah variabel *array* asosiatif terlebih dahulu, variabel *array* asosiatif ini kemudian akan menjadi parameter yang harus dilewatkan ke dalam *view*, pada perintah `$this->load->view()`. Setiap key dari *array* asosiatif akan menjadi sebuah variable di dalam *view*.

3. Controller

Controller merupakan komponen yang digunakan untuk menangani interaksi pengguna, bekerja dengan *model*, dan memilih *view* mana yang digunakan untuk merender data. *Controller* adalah komponen utama dari suatu program yang menggunakan teknik program MVC. *Controller* secara sederhana adalah sebuah file class yang memiliki nama yang dapat diasosiasikan dengan sebuah URI atau URL.

Perhatikan URL berikut ini : <http://server/index.php/hello/>

Pada contoh ini url di atas, CodeIgniter akan berusaha untuk menemukan sebuah *controller* dengan nama `hello.php` dan memuatnya untuk dieksekusi.

1) Aturan pembuatan controller

Setiap *controller* yang dibuat secara default, harus disimpan atau diletakan di dalam direktori *controllers* yang ada di bawah direktori *application*, di bawah direktori instalasi CI. Dalam penjelasan ini, kita menggunakan XAMPP, maka *controller* harus dipasang di :

```
c:\xampp\htdocs\ci20\application\controllers.
```

2) Deklarasi Controller

Controller dideklarasikan sebagai class yang di turunkan dari *controller* CodeIgniter, yaitu *CI_Controller*.

```
<?php
```

```
Class C_namakontroller extends CI_Controller(  
..  
)
```

C_namakontroller adalah nama *controller* baru yang akan dibuat.

CI_controller adalah objek *controller* awal atau parent CI yang digunakan untuk membuat *controller*.

Extends adalah kata kunci untuk membuat class baru yang diturunkan dari class yang sudah ada, dalam hal ini dari class *CI_Controller*

3) Nama controller

Nama *controller* harus sama dengan nama file *controller*. Aturan penamaan *controller* diawali dengan pernyataan class, kemudian nama *controller*, dengan nama sebaiknya diawali dengan huruf besar, selanjutnya dengan huruf kecil, dan tanda baca yang boleh sebagai bagian dari nama

class sudah underscore(_). Sebelum versi 2, nama *controller* harus diawali dengan huruf besar; tetapi kita masih bisa tetap menggunakan aturan tersebut. Nama *controller* tidak perlu menggunakan .php. hanya nama file *controller* saja yang memiliki ekstensi .php

2.2.2 JQuery Mobile

JQuery Mobile adalah web *framework* yang dioptimalkan untuk perangkat-perangkat layar sentuh (*touch-optimized*). Sebagai pustaka, jQuery Mobile juga dikenal sebagai JavaScript maupun *framework mobile* yang juga dikenal sebagai library JavaScript maupun *Framework mobile* ini dikembangkan oleh tim jQuery Project. Tujuan pengembangannya adalah berfokus pada pembuatan *framework* yang kompatibel dengan beraneka ragam perangkat smartphone dan tablet. JQuery Mobile dikembangkan sebagai alternatif yang menjawab kebutuhan akan pengembangan aplikasi web *mobile* yang mudah dan memiliki banyak potensi. jQuery Mobile sangat mungkin untuk membuat aplikasi web *mobile* yang penampilan dan interaktivitasnya konsisten di semua perangkat yang mendukung. Serta memiliki kemampuan *user interface (UI)* yang canggih.

JQuery Mobile merupakan sistem user interface berbasis HTML5 dan pengembangan dari library jQuery dan jQuery UI yang sedang populer saat ini. Kelebihan jQuery adalah kodenya ringan, menerapkan desain yang dapat beradaptasi dengan berbagai jenis perangkat *mobile*, dan menonjolkan markup *semantic*. Oleh karena itu, desainnya sangat fleksibel dan mempunyai *Theme* yang bisa disesuaikan. Hal ini membuat jQuery Mobile sangat sekali digunakan.

JQuery memiliki sekumpulan *plug-in* dan widget yang dimaksudkan untuk menyediakan API agar dapat dikenali di berbagai *platform* untuk mengimplementasi kode, jQuery Mobile sangat mirip jQuery UI. Perbedaan antara jQuery UI dan *Mobile* berfokus pada aplikasi desktop. jQuery

Mobile dibangun dengan pertimbangan perangkat *mobile*. banyak yang sudah menerapkan jQuery Mobile untuk produksi. Hal ini dapat menjadi bukti tidak hanya kualitas dan kestabilan library-nya, tapi juga mudahnya penggunaan jQuery Mobile.

jQuery Mobile dapat dimanfaatkan untuk memenuhi kebutuhan web *mobile*. Penerapannya dapat membantu dalam pengembangan web *mobile* tanpa banyak melibatkan praktik pemrograman yang rumit sehingga keinginan pengguna perangkat *mobile* untuk dapat mengakses web *mobile* secara lebih mudah dan cepat dapat dipenuhi dengan menerapkan jQuery Mobile.

jQuery Mobile dibuat sebagai *framework* yang modern, memanfaatkan banyak pola dan best practice untuk pengembangan aplikasi *multiplatform*. Fitur-fitur utama *framework* antara lain:

1. Dikembangkan dari inti jQuery yang sudah dikenal dan banyak diadaptasi. Selain itu juga juga memanfaatkan pola-pola dan kode jQuery UI. Bagi yang sudah terbiasa dengan jQuery ataupun jQuery UI akan merasa lebih familiar dengan sintaksnya sehingga muda mempelajarinya.
2. Lintas *platform*, lintas perangkat, dan lintas *browser*. jQuery Mobile kompetibel dengan mayoritas perangkat *mobile*, tablet, e-reader, dan *platform* desktop seperti iOS, Android, Blackberry, Palm, WebOS, Nokia/Symbian, Windows Phone 7, Opera Mobile atau Opera Mini, firefox Mobile, Kindle, Nook, MeeGo, dan semua *browser* modern dengan tingkat-tingkat kompatibilitas tertentu.
3. Ukurannya ringan, kompresannya hanya sekitar 12kb, dan sangat sedikit bergantung pada gambar grafis sehingga memungkinkan peningkatan kepada kecepatan.
4. Arsitektur modular, bisa dioptimalkan dengan menyertakan fitur-fitur yang diperlukan untuk aplikasi tertentu saja dan tidak memilih yang di butuhkan.
5. *Mark-up driven*, jQuery Mobile hanya menggunakan kode HTML5 semantik yang tidak mengganggu (*nonintrusif*), tanpa memerlukan pengetahuan tentang javascript, CSS, atau API

untuk mempelajarinya. Memungkinkan pengembangan secara cepat dan hanya perlu menulis sedikit skrip

6. Pendekatan *progressive enhancement* yang membawa konten dan fungsionalitas intinya pada semua *platform* (*mobile*, tablet, desktop), dan membawa pengalaman yang mirip dengan aplikasi sesungguhnya pada *platform-platform mobile* terbaru.
7. Teknik desain responsif yang memungkinkan aplikasi untuk mengadaptasi berbagai ukuran layar, mulai dari ukuran layar smartphone hingga desktop hanya dengan satu basis kode yang sama.
8. AJAX call untuk memuat konten-konten dinamis dengan otomatis. Sistem navigasi juga memanfaatkan AJAX untuk menciptakan efek transisi halaman, sementara menjaga tombol back, bookmark, dan URL tetap bersih.
9. *User Interface* yang dioptimalkan bagi perangkat layar sentuh yang mendukung touch event dan mouse event. Dukungan API yang disediakan cukup menyederhanakan proses dalam mendukung sentuhan dan inputan pengguna berdasarkan *focus mouse* dan *cursor*.
10. *Framework* yang power full untuk memuat theme. Ada juga aplikasi ThemeRoller untuk mempermudah pembuatan theme yang dapat disesuaikan dengan brand.

JQuery Mobile terdengar seperti solusi yang menjanjikan untuk memenuhi tantangan di dunia *mobile* yang semakin berkembang dengan pesat. jQuery Mobile terdiri atas program dan file-file pendukung yang saling terintegrasi dengan baik membentuk *framework* powerfull untuk pengembang aplikasi *mobile*. *Framework* jQuery Mobile bekerja dengan sistem navigasi dan manajemen page yang canggih untuk menciptakan pengalaman *mobile* yang menarik dan fungsional bagi pengguna aplikasi.

Berikut ini kelebihan jQuery Mobile yaitu :

1. Mendukung Mobile Browser

- 1) Mendukung web *mobile browser* persis sama dengan desktop web *browser*
- 2) Tata Letak Responsif

2. **Class Media Query Helper**

1. JQuery Mobile menambahkan kelas untuk elemen HTML
2. Mengikuti bentuk rotasi *browser* dan min /maksimal CSS pada umumnya Class ini mengikuti proses, perubahan size dan perubahan orientasi
3. Memungkinkan untuk mengunci kelas ini di CSS
4. Membentuk layout responsif walaupun *browser* tidak mendukung Elemen HTML akan selalu memiliki pilihan "potret" atau "landscape"
5. Tergantung pada orientasi *browser* atau perangkat
6. Contoh penggunaan kode CSS sebagai berikut :

```
portrait {  
  
    /* portrait orientation changes go here! */  
  
}  
  
.landscape {  
  
    /* landscape orientation changes go here! */  
  
}
```

Breakpoint Class Min/Max Width

Secara *default*, *min* dan *max* kelas breakpoint dibuat pada ukuran berikut: 320, 480, 768, 1024 Kelas yang terlihat seperti ini "*min-width-320px*", "*max-width-480px*". Dapat digunakan sebagai pengganti atau tambahan setara permintaan media yang mereka meniru

```
.myelement {  
float: none;  
}  
  
.min-width-480px .myelement {  
float: left;  
}
```

Elemen form mengapung disamping label mereka ketika *browser* lebih lebar dari 480 piksel. CSS untuk mendukung perilaku ini untuk input teks bentuk terlihat seperti ini:

```
label.ui-input-text {  
    display: block;  
}  
  
.min-width-480px label.ui-input-text {  
display: inline-block;  
}
```

Menjalankan Media Query

Fungsi pengujian memungkinkan apakah CSS Media Query berjalan
panggil *\$.mobile.media()* dan query

Hasil akan muncul jika query yang benar dan mendukung, seperti di bawah ini

```
//test for screen media type  
  
$.mobile.media("screen");  
//test a min-width media query  
$.mobile.media("screen and(min-width: 480px)")
```