



# Pengembangan Aplikasi Menggunakan Framework Code Igniter 3 Level Basic

Helmy Faisal Muttaqin, S.Kom, M.T

## Apa itu Framework?

- Framework merupakan kumpulan library siap pakai yang dapat dimanfaatkan dalam membangun aplikasi tertentu.
- Untuk menggunakan library-library yang disediakan oleh suatu framework, developer harus menyesuaikan diri dengan aturan-aturan pada framework tersebut, karena setiap framework mempunyai aturan yang berbeda.

# Framework PHP yang umum digunakan:

- Codeigniter
- Yii (Yes It Is)
- Prado
- CakePHP
- Zend
- Laravel
- Kohana
- Symfoni
- dll



# Peringkat Framework 2015 di Dunia (menurut SitePoint, 2015)

1. Laravel
2. Symfoni2
3. Nette
4. **Codeigniter**
5. Yii 2
6. PHPixie
7. Yii 1
8. Zend Framework



# Framework Codeigniter (CI)

- Merupakan framework peringkat 1 yang paling banyak digunakan di Indonesia (menurut SitePoint, 2015).
- Digunakan oleh sebagian besar web developer di Indonesia.

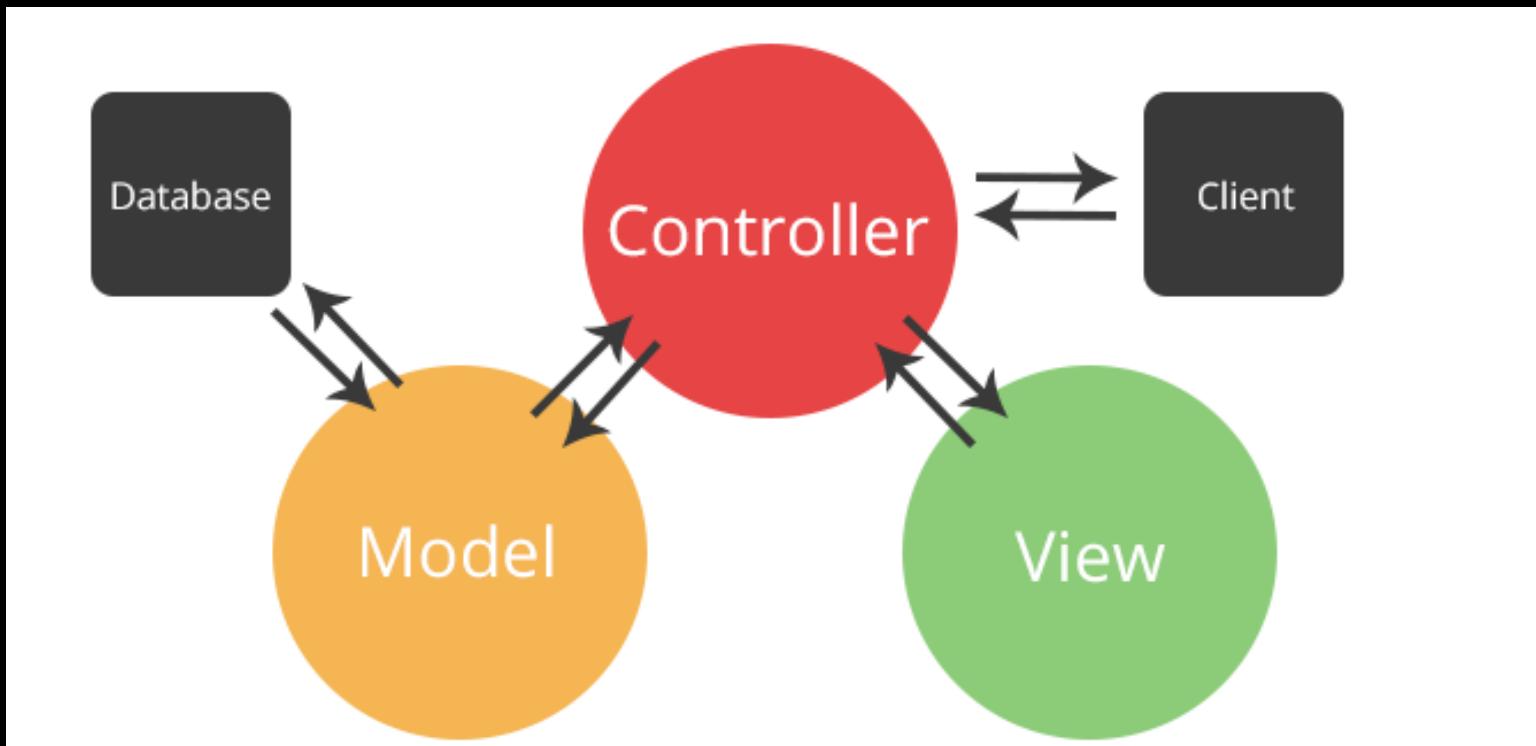


# Keunggulan Codeigniter

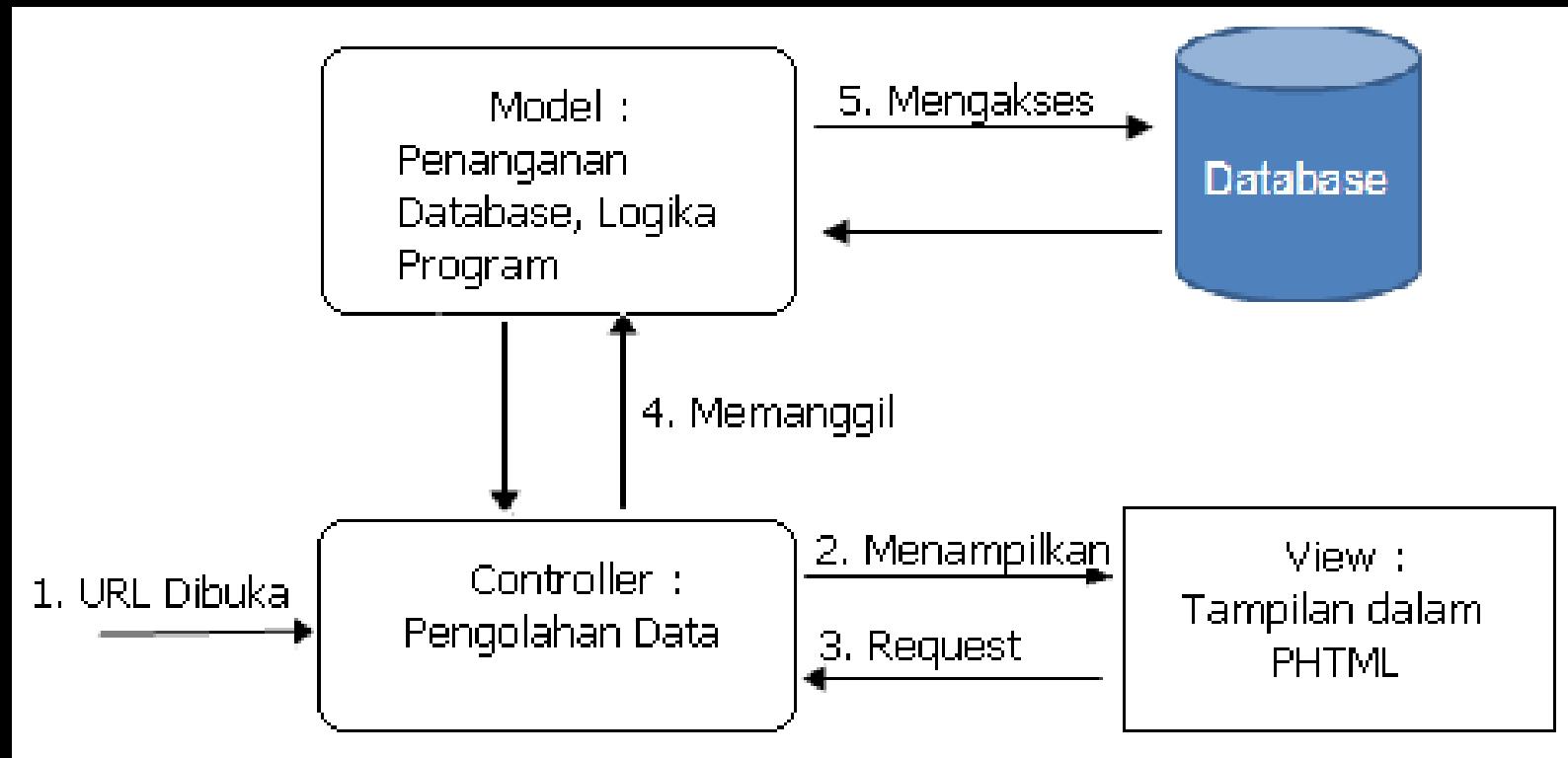


- Framework Open Source (Free)
- Sangat ringan di semua platform sistem operasi
- Berbasis MVC (Model View Controller)
- Library lengkap
- Dokumentasi lengkap
- Mempunyai komunitas pengguna yang besar
- Mudah dipelajari
- Berbasis PHP5

# MVC



# MVC pada CI



# Memulai Belajar Codeigniter. Tahap Persiapan:

- Text Editor (Notepad, Notepad++, dll)
- Server Localhost (Wamp, Xampp, dll)
- Framework Codeigniter → download di  
[www.codeigniter.com](http://www.codeigniter.com)
- Versi terbaru: Codeigniter 3.0.6

# Memulai Codeigniter

- Download Codeigniter
- Ekstrak dan letakkan pada directory www atau htdocs pada localhost
- Hidupkan service HTTP dan MySQL
- Jalankan pada web browser (firefox, chrome, dsb)

# Tampilan Default Code Igniter

Welcome to CodeIgniter!

The page you are looking at is being generated dynamically by CodeIgniter.

If you would like to edit this page you'll find it located at:

`application/views/welcome_message.php`

The corresponding controller for this page is found at:

`application/controllers/Welcome.php`

If you are exploring CodeIgniter for the very first time, you should start by reading the [User Guide](#).

# Konfigurasi #1

- Buka file routes.php yang terletak di folder application/config
- Temukan perintah `$route['default_controller'] = 'welcome';`
- Hal ini berfungsi untuk menentukan controller mana yang akan dipanggil pertama kali saat website dijalankan pada browser.

## Konfigurasi #2

- Buka file Welcome.php yang terletak di folder application/controllers

```
1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class Welcome extends CI_Controller {
5
6     public function index()
7     {
8         $this->load->view('welcome_message');
9     }
10 }
```

- Temukan fungsi **index()**. Fungsi ini akan dijalankan ketika controller Welcome dipanggil.

## Konfigurasi #3

- Pada baris ke 8 terdapat perintah untuk menampilkan halaman view dengan nama **welcome\_message**.
- Buka file **welcome\_message.php** yang ada di folder **application/views**
- Inilah halaman web yang akan ditampilkan di browser.

# Latihan

- Ketikkan kode program berikut pada fungsi index pada controller Welcome:

```
$pesan["pesan1"] = "Ini adalah pesan pertama";  
$pesan["pesan2"] = "Ini adalah pesan kedua";  
$this->load->view('myview',$pesan);
```

- Buat sebuah view baru dengan nama myview.php dan simpan pada folder application/views.

• • •

- Pada file myview.php tambahkan kode program:

```
<html>
<head>
<title>File View Pertama</title>
</head>

<body>
File view berhasil ditampilkan
<p>Nilai array pesan1 adalah: <?php echo $pesan1; ?></p>
<p>Nilai array pesan1 adalah: <?php echo $pesan2; ?></p>
</body>

</html>
```

- Kemudian jalankan pada browser.

## Mengakses File CSS, JS dan Gambar di CI

- aktifkan helper URL. Buka file `autoload.php` dan tambahkan: `$autoload['helper'] = array('url');`
- Buat folder dengan nama assets.
- Simpan file CSS, JS atau gambar pada folder ini.
- Untuk mengakses file css, tambahkan:

```
<link rel="stylesheet" type="text/css" href="php<br/echo base_url().'assets/css/bootstrap.min.css' ?>" />
```

# Mengakses File JS di CI

- Simpan file JS di folder assets/js
- Untuk mengakses file js, tambahkan:

```
<script src="base_url().'assets/js/javascript.js"  
      type="text/javascript" />  
</script>
```

# Mengakses File Gambar di CI

- Simpan file JS di folder **assets/pictures**
- Untuk mengakses file Gambar, tambahkan:

```

```

# Mengakses File Lainnya

- Simpan file JS di folder `assets/doc`
- Untuk mengakses file Gambar, tambahkan:  
`<a href="base_url().".'assets/doc/Document.docx  
" type="download">Link Download</a>`

# Config Database Connection (Default Config) -1

- Lokasi file config:  
**application\config\database.php**
- Setting variabel sebagai berikut:
  - **active\_group = 'nama\_group1';**
  - **\$active\_record = TRUE;**

# Config Database Connection (Default Config) -2

- \$db['nama\_group1']['hostname'] = 'localhost';
- \$db['nama\_group1']['username'] = 'root';
- \$db['nama\_group1']['password'] = '112233';
- \$db['nama\_group1']['database'] = 'kampus';
- \$db['nama\_group1']['dbdriver'] = 'mysql';
- \$db['nama\_group1']['dbprefix'] = '';

# Config Database Connection (Default Config) - 3

- \$db['nama\_group1']['pconnect'] = TRUE;
- \$db['nama\_group1']['db\_debug'] = TRUE;
- \$db['nama\_group1']['cache\_on'] = FALSE;
- \$db['nama\_group1']['cachedir'] = '';
- \$db['nama\_group1']['char\_set'] = 'utf8';
- \$db['nama\_group1']['dbcollat'] =  
'utf8\_general\_ci';
- \$db['nama\_group1']['swap\_pre'] = '';

# Config Database Connection (Default Config) - 4

- \$db['nama\_group1']['autoinit'] = TRUE;
- \$db['iwan9506']['stricton'] = FALSE;
- \$db['iwan9506']['port'] = 80;

# Config Database Connection (Active Record Config)

- \$active\_group variable bisa memilih group connection mana yang mau diaktifkan.
- Buat method di class model, dengan detail method berikut:

```
function active_dbs($dbs)
{
    //sama seperti default config
    $config['database']=$dbs;
    return $this->load->database($config,true);}
```

# Menghilangkan index.php

- Setting file: application/config/config.php  
temukan: \$config['index\_page'] = "";  
//kosongkan nilainya
- Setting file htaccess:  

```
RewriteEngine on  
RewriteCond %{REQUEST_FILENAME} !-f  
RewriteCond %{REQUEST_FILENAME} !-d  
RewriteRule .* index.php/$1 [PT,L]
```

# Membuat Library

- Library berekstensi .php
- Library berbasis oop.
- Library bawaan CI terletak di: **system/ libraries**
- Library buatan sendiri (custom) simpan di:  
**application/ libraries**

# Membuat Library

```
class new_lib
{
    function add($config)
    {
        //aksi
    }
}
```

# Memanggil Library di controller

- \$this->load->library('new\_lib', \$config\_file);
- \$lib = new new\_lib();
- /\*buat objek dari class library\*/
- \$nama = \$lib.execute();  
/\*eksekusi method dalam objek new\_lib\*/

# Membuat Helper (Yang perlu diperhatikan)

- Helper berekstensi .php
- Helper tidak berbasis oop.
- Helper biasanya berisi code HTML, CSS, JS.
- Beri akhiran “\_helper” pada nama file.  
Contoh: helper captcha disimpan dengan  
nama file **captcha\_helper.php**

# Memanggil Helper

- `$captcha = $this->load->helper('captcha');`  
`/* pada file aslinya helper captcha diberi nama`  
`captcha_helper */`

# Live Demo Coding Code Igniter

## Dasar





The End