

Pemrograman Web Berbasis Framework



Pertemuan 4 : Konsep MVC : Controller

Hasanuddin, S.T., M.Cs.

Prodi Teknik Informatika UAD

hasan@uad.ac.id



Pokok Bahasan

- Pendahuluan
- Struktur direktori dari web framework
- Instalasi dan konfigurasi
- Controller Dasar
- Pemanggilan View
- Pemanggilan Model
- Pemanggilan Fungsi

TIK :

Setelah mengikuti kuliah ini mahasiswa dapat mengetahui dan memahami konfigurasi dan instalasi web framework serta konsep Controller dalam konsep MVC



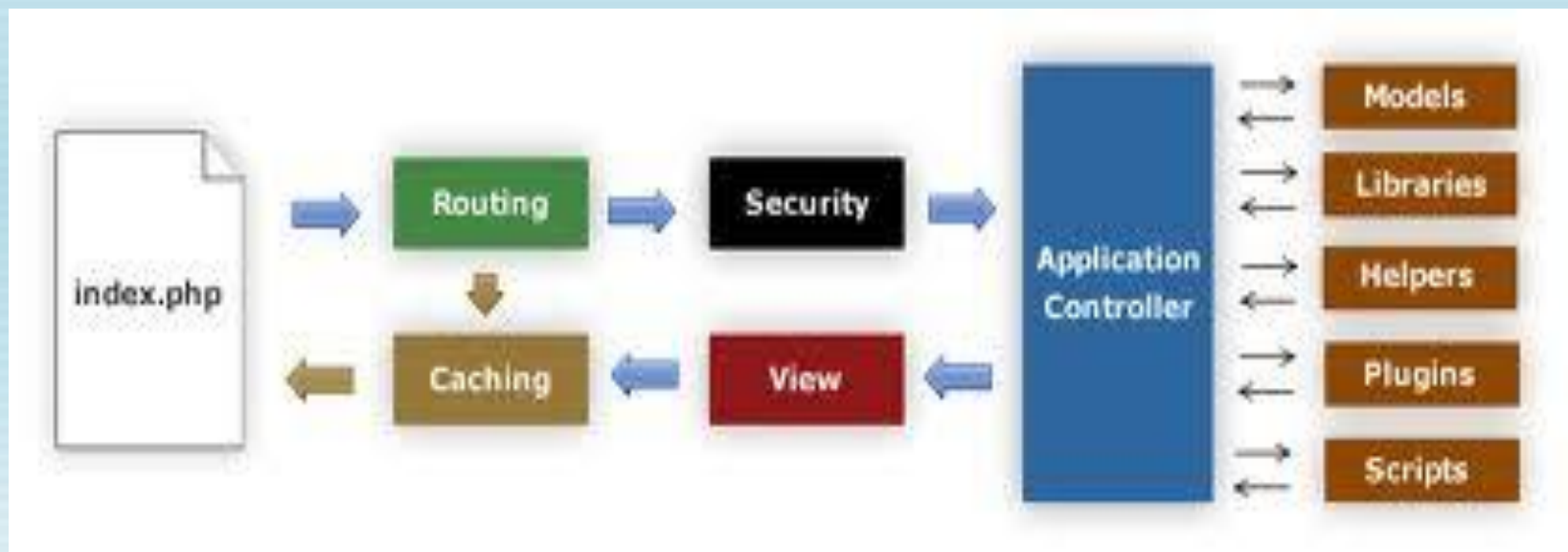
Pendahuluan

- Untuk memahami konsep MVC suatu framework harus memahami :
 - Instalasi dan konfigurasi web framework tsb
 - Struktur direktori dalam framework
 - Penerapan MVC (sintaks)
- Kebutuhan tools:
 - Web server (XAMPP, WAMP, dll)
 - Editor Web (Mac. Dreamweaver, dll)
 - Framework CodeIgniter 1.7.2.



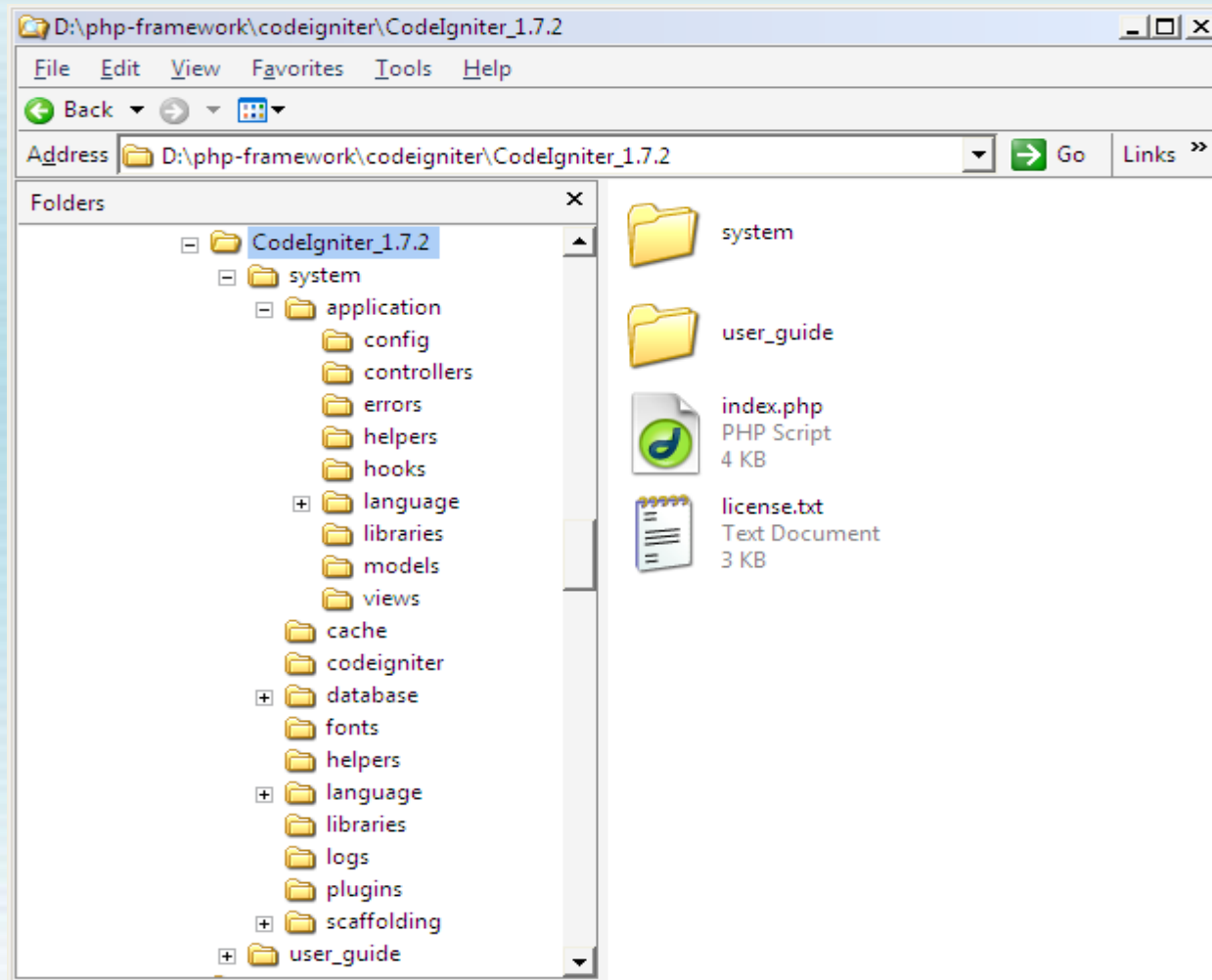
Pendahuluan (2)

Flowchart Aplikasi pada Web Framework





Struktur Direktori CodeIgniter





Struktur Direktori CodeIgniter (2)

- Folder system → folder sistem aplikasi yang dibangun
- Folder user_guide → manual penggunaan CodeIgniter
- File index.php → file yang meng-handle semua request yang dilakukan oleh client (pengunjung)



Struktur Direktori CodeIgniter (3)

- Folder system terdiri dari :
 - Folder application → tempat menyimpan file aplikasi yang dibangun
 - Folder cache → menyimpan semua cache yang dibuat cache library
 - Folder codeigniter → menyimpan file internal codeigniter
 - Folder database → menyimpan driver database
 - Folder fonts → menyimpan font yg digunakan pada sistem
 - Folder helpers → menyimpan helper core dari codeigniter
 - Folder language → menyimpan file bahasa core codeigniter
 - Folder logs → menyimpan semua log generated oleh codeigniter
 - Folder plugin → menyimpan plugin core codeigniter
 - Folder scaffolding → menyimpan semua file yg berfungsi sebagai scaffolding



Struktur Direktori CodeIgniter (4)

- Folder application terdiri dari :
 - Folder config → menyimpan konfigurasi aplikasi
 - Folder controllers → menyimpan semua file controller dari aplikasi
 - Folder errors → menyimpan semua file template error aplikasi
 - Folder helpers → menyimpan helper aplikasi (bukan core codeigniter)
 - Folder hook → menyimpan hook yg digunakan untuk mengubah alur fungsi core codeigniter
 - Folder language → tempat menyimpan bahasa yg digunakan aplikasi
 - Folder libraries → menyimpan semua library buatan sendiri (bukan core)
 - Folder models → menyimpan semua file model dari aplikasi
 - Folder views → menyimpan semua file view dari aplikasi



Instalasi/Konfigurasi

- Instalasi maupun konfigurasi framework CodeIgniter terletak pada direktori `system/application/config`
- Secara umum ada 4 file yang perlu dilakukan perubahan untuk konfigurasi/instalasi, yaitu :
 - `Config.php`
 - `Autoload.php`
 - `Database.php`
 - `Routes.php`



Instalasi/Konfigurasi (2)

- Pada file config.php
 - Ubahlah baris program berikut :
`$config['base_url'] = "http://example.com/";`
 - Menjadi URL project, misalnya:
`$config['base_url'] = "http://localhost/latihan/";`
- Base URL merupakan alamat default dari website/aplikasi yg dibuat.
- Secara default URL tersebut akan digunakan sebagai awal link dalam halaman website/aplikasi.



Instalasi/Konfigurasi (3)

- Pada file database.php
 - Ubahlah baris program berikut :

```
$db['default']['hostname'] = "localhost";  
$db['default']['username'] = "";  
$db['default']['password'] = "";  
$db['default']['database'] = "";
```
 - Menjadi sesuai konfigurasi koneksi database server, misalnya:

```
$db['default']['hostname'] = "localhost";  
$db['default']['username'] = "root";  
$db['default']['password'] = "";  
$db['default']['database'] = "latihandb";
```
- Untuk kebutuhan konfigurasi di atas, nama database harus dibuat terlebih dahulu



Instalasi/Konfigurasi (4)

- Pada file autoload.php
 - Ubahlah baris program berikut :
`$autoload['libraries'] = array();`
 - Menjadi :
`$autoload['libraries'] =
array('database', 'session', 'validation');`
- Kumpulan library di atas merupakan library yang akan diaktifkan secara otomatis diawal program.
- Beberapa library yang umum digunakan adalah database, session, validation, form_validation, dll.



Instalasi/Konfigurasi (5)

- Pada file autoload.php
 - Ubahlah baris program berikut :
`$autoload['helper'] = array();`
 - Menjadi :
`$autoload['helper'] = array('url', 'file', 'form', 'table');`
- Helper merupakan kumpulan fungsi-fungsi dalam berbagai kategori untuk memudahkan pengembangan aplikasi.
- Kumpulan helper di atas merupakan helper yang akan diaktifkan secara otomatis diawal program.
- Beberapa helper yang umum digunakan adalah url, file, form, table, dll.



Instalasi/Konfigurasi (6)

- Pada file routes.php
 - Ubahlah baris program berikut :
`$route['default_controller'] = "welcome";`
 - Menjadi default controller kita, misalnya:
`$route['default_controller'] = "utama";`
- Default controller ini dimaksudkan sebagai controller yang akan dijalankan pertama kali oleh user.
- Contoh :
<http://localhost/latihan/index.php/utama/index>
- Maka hanya perlu ditulis:
<http://localhost/latihan>



Controller Dasar

- Controller dasar merupakan controller utama dan sederhana, misalnya menampilkan tulisan “Selamat Belajar CodeIgniter”
- Misalkan terdapat controller **latihan** (file **latihan.php** di direktori `system/application/controllers/`, maka akan terdapat kode program sebagai berikut :

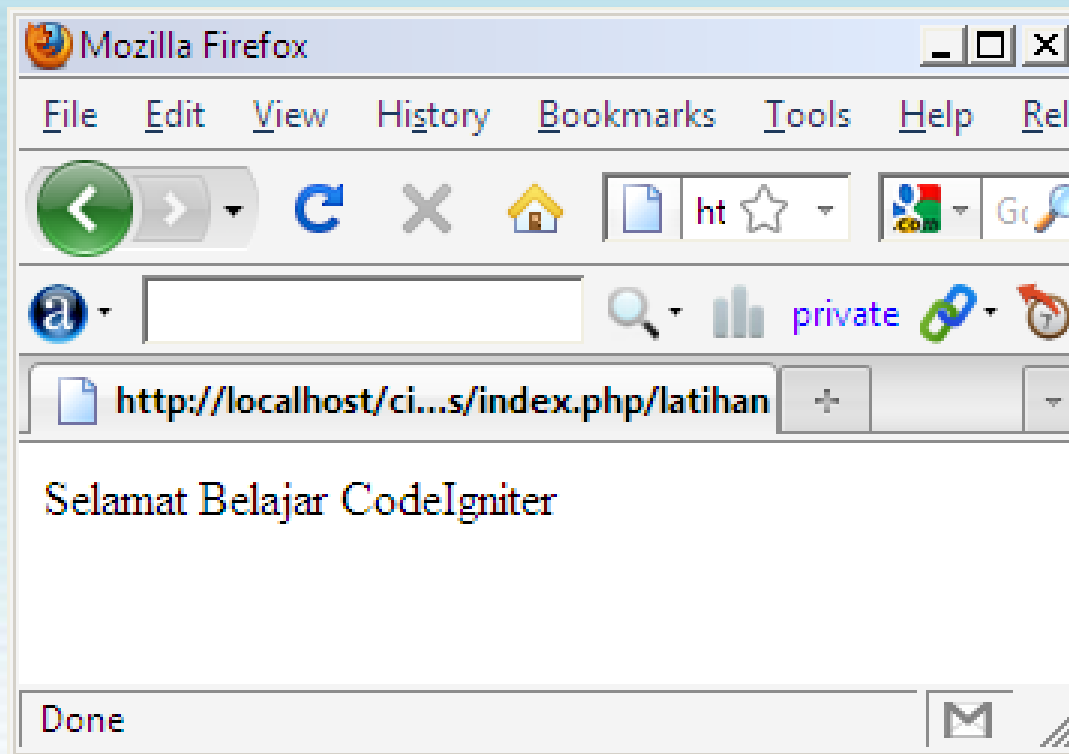
```
<?php
class Latihan extends Controller {
    function Latihan()
    {
        parent::Controller();
    }
    function index()
    {
        echo “Selamat Belajar CodeIgniter”;
    }
}
```



Controller Dasar (2)

- Untuk melakukan pemanggilan controller latihan, maka digunakan alamat URL berikut :

<http://localhost/utama/index.php/latihan>





Pemanggilan View

- View dalam konsep MVC merupakan bagian untuk menangani tampilan.
- Misalkan terdapat view vlatihan (vlatihan.php di direktori system/application/views/) dengan kode sbb:

```
<?php  
echo "Selamat Belajar CodeIgniter";  
echo "<br>ini merupakan tulisan yang ditampilkan melalui view";  
?>
```

- Maka pemanggilan view menjadi :
`$this->load->view('vlatihan');`



Pemanggilan View (2)

- Jika melalui view dilewatkan suatu variabel, maka variabel tersebut harus berupa variabel array.

- Contoh :

```
$data['teks'] = "Selamat Belajar CodeIgniter";
```

```
$this->load->view('vlatihan',$data);
```

- Maka pada view **vlatihan** cukup berisi kode :

```
<?php
```

```
echo $teks;
```

```
?>
```



Pemanggilan Model

- Model dalam konsep MVC merupakan bagian yang menangani pengolahan data ke dan dari database
- Model berisi class dan fungsi seperti Controller, tidak seperti pada View
- Pemanggilan Model hampir sama dengan pemanggilan View, namun perbedaannya dalam pemanggilan model dilakukan dengan pemanggilan fungsi



Pemanggilan Model (2)

- Contoh : terdapat model mlatihan (file mlatihan.php di direktori system/application/models/) berisi kode program :

```
<?php  
Class mlatihan extends Model {  
    function mlatihan() {  
        parent::Model();  
    }  
    function getData() {  
        $data = "Selamat Belajar CodeIgniter";  
        return $data;  
    }  
}  
?>
```



Pemanggilan Model (3)

- Maka pemanggilan model mlatihan sebagai berikut :

```
$this->load->model('mlatihan');
```

```
echo $this->mlatihan->getData();
```



Pemanggilan Fungsi

- Dalam suatu controller dimungkinkan adanya pemanggilan fungsi
- Contoh :

```
function index() {  
    echo form_open('latihan/terima');  
    echo form_input('nim','07018111');  
    echo form_input('nama','budiman');  
    echo form_submit('submit','Kirim');  
    echo form_close();  
}  
  
function terima() {  
    echo "NIM = ".$_POST['nim'];  
    echo "<br>Nama= ".$_POST['nama'];  
}
```



Pemanggilan Fungsi (2)

- Pemanggilan/penggunaan fungsi dapat melibatkan variabel input
- Misalnya pada fungsi baca akan dilewatkan dua variabel integer dengan alamat :
- <http://localhost/utama/index.php/latihan/baca/5/10>
- Maka fungsi baca berisi kode program :

```
function baca($bil1, $bil2) {  
    echo "Bilangan 1 = ".$bil1;  
    echo "<br>Bilangan 2 = ".$bil2;  
}
```



Referensi :

- Awan Pribadi Basuki, ***Membangun Web Berbasis PHP dengan Framework CodeIgniter***, Lokomedia, Yogyakarta, 2010.
- Wardana, ***Menjadi Master PHP dengan Framework CodeIgniter***, Elexmedia Komputindo, Jakarta, 2010.
- David Upton, ***CodeIgniter for Rapid PHP Application Development: Improve your PHP coding productivity with the free compact open-source MVC CodeIgniter framework***, Packt Publishing, 2007.