**BAB IV**

**UJI COBA APLIKASI**

**4.1. Tahap Pembuatan *Website***

Tahap ini adalah bagian untuk pengaturan aplikasi yang dilakukan pada file konfigurasi. Pengaturan - pengaturan ini termasuk :

* Base URL
* Koneksi Database
* Auto Load
* Session
* Alias Controller dan Alias Method
* Query filter
* Cache

**4.1.1. Konfigurasi Dasar**

File konfigurasi terdapat pada direktori :

apps/config.php

Hal dasar yang harus dilengkapi pada bagian konfigurasi ini adalah URL dasar (*base URL*).

Isikan lokasi url aplikasi pada parameter **base\_url**, misalnya:

$CONFIG['base\_url'] = 'http://localhost/cempakaputih/';

**4.1.2. *Webserver***

Secara default, file **index.php** telah di sertakan pada setiap *url*.

Contoh :

http://localhost/framework/**index.php**/home

Untuk menghilangkan **index.php** pada *url*, pertama - tama beri nilai kosong

('') pada parameter **index\_file** yang ada di dalam *file* apps/config.php. Ubah *array* berikut:

$CONFIG['index\_file'] = 'index.php/';

Menjadi seperti berikut:

$CONFIG['index\_file'] = '';

**4.2. *Controller***

**4.2.1. Membuat *Class Home***

1. *Edit file* home.php yang berada pada folder cempakaputih\apps\controller
2. Pastikan class *controller*-nya adalah *home*

*class Controller\_home*

1. Tambahkan fungsi *index (prefix)*

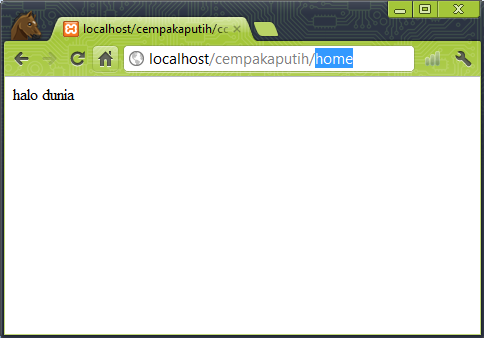
*public function index(){*

*echo ‘halo dunia’;*

*}*

1. Melihat *Output*

Masukan *url* http://localhost/cempakaputih/home pada *browser*.



**Gambar 4.1. *Output Class home***

*Controller* adalah sebuah *class* yang dalam mengaksesnya menggunakan *path url* tertentu. Nama *path url* mewakili *nama controller*.

Contoh :

nama *controller* = home

*path url* = localhost/cempakaputih/home

**4.3. *VIEW***

**4.3.1. Membuat *View dan me-Load Component***

* + 1. Untuk membuat *file view*, tambahkan *file* baru di dalam *folder apps\view\*
    2. Buat *file baru*, misalnya **home.php**

**Masukan kode program ke dalam home.php**

***<html>***

***<head>***

***<title><?php echo $title; ?></title>***

***</head>***

***<body>***

***<?php echo $content; ?>***

***<br />***

***data array : <br />***

***<i>component array[title]</i> : di load di <? echo $title;?><br />***

***<i>component array[content]</i> : di load di <? echo $content;?>***

***</body>***

***</html>***

* + 1. **Edit *class home* yang berada di file *home.php* pada *folder controller apps\controller\home.php***
    2. **Buat *array* pada fungsi atau *prefix index***

***public function index(){***

***$data = array(***

***'title' => 'Hello World Page!',***

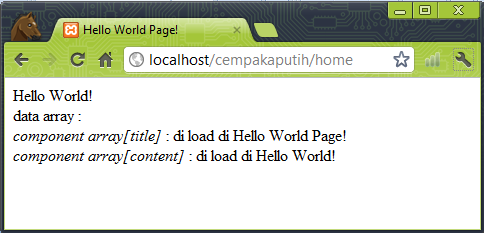
***'content' => 'Hello World!'***

***);***

***$this->view\_home($data);***

* + 1. Melihat *Output*

Masukan *url* http://localhost/cempakaputih/home pada *browser*.



**Gambar 4.2. *Load Component* pada *class home***

**4.4. Model**

**4.4.1. Membuat Model**

Untuk membuat sebuah *class* model, tambahkan *file* baru dan letakan di dalam *folder* **apps/model/**, misalnya **users.php**. Buat *class* *users* di dalam *file* ini dengan diawali *prefix* **"Model\_"**.

Contoh :

*<?php*

*class Model\_users {*

*public function \_\_construct(){*

*}*

*}*

**4.4.2. *Load Library* di dalam *Model***

Jika akan me-load library atau class lain di dalam model ini, ada dua cara yang bisa dilakukan :

1. Jika *resource* dari *class* atau *library* tersebut ingin digunakan oleh semua *method* yang ada di dalam *class* model, maka pengambilan dan penginisialisasian bisa dilakukan di *method* **\_\_construct**.
2. Jika *resource* hanya ingin digunakan pada salah satu *method* saja, maka inisialisasi dan pemanggilan bisa dilakukan di dalam *method* yang bersangkutan.

Salah satu hal yang paling umum dilakukan di dalam sebuah class model adalah me-*load* *library* *database*.

Contoh :

*<?php*

*class Model\_users {*

*public function \_\_construct(){*

*$this->db = new Library\_db();*

*}*

*}*

## 4.5. *Global Method* dan *Properties*

Berikut adalah *tools* bantuan yang bisa di manfaatkan dalam aplikasi.

**4.5.1. Base\_url**

*Base Url* adalah informasi *url root* dari aplikasi yang informasinya dapat di isikan pada bagian [Konfigurasi Dasar](http://localhost/cms/panada/documentation/id/config.html#general). Untuk mendapatkan *base\_url* aplikasi, bisa menggunakan properties:

$this->base\_url;

Outputnya adalah :

http://localhost/framework/

*Method* ini dapat di gunakan untuk memanggil *file statis* seperti *css*, *js*, *image* dan lain – lain.

Contoh :

<link rel="stylesheet" href=" <?php echo **$this->base\_url**;?> style.css" type="text/css" media="screen" />

**4.5.2. *Location***

Untuk menentukan lokasi *url*, dapat menggunakan *method location* :

$this->location('home');

Outputnya adalah http://localhost/cempakaputih/home

**4.5.3. *Redirection***

Untuk melakukan *redirect* ke alamat dalam aplikasi yang sama, formatnya cukup dengan **"nama\_controller/nama\_method"**. Bila ingin *redirect* ke halaman aplikasi lain, tambahkan ***http* atau *https*** pada alamat yang dituju.

Contoh :

Untuk melakukan redirect ke halaman utama :

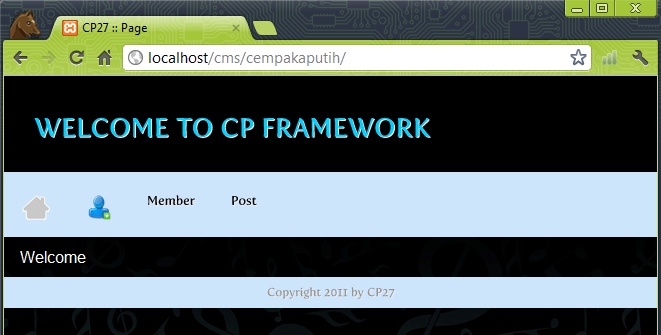
$this->redirect();

**4.6. *Module***

*Module* adalah sebuah sub-aplikasi yang memiliki komponen lengkap seperti layaknya aplikasi utama yang terdiri dari Model, *View*, *Controller* dan *Library*. Yang membedakannya adalah sebuah *module* harus digabungkan ke dalam aplikasi utama dan tidak dapat berdiri sendiri.

*Resource* sebuah *module* bisa langsung diakses dari dalam *controller* utama. *Module* ini akan memberikan *output resource* melalui *interface controller module*. *Resource* sebuah *module* juga dimungkinkan untuk diakses melalui *URL routing*. Sebuah *module* juga dapat mengakses *resource* dari *module* lain.

**4.7. *Output* pada *Browser***



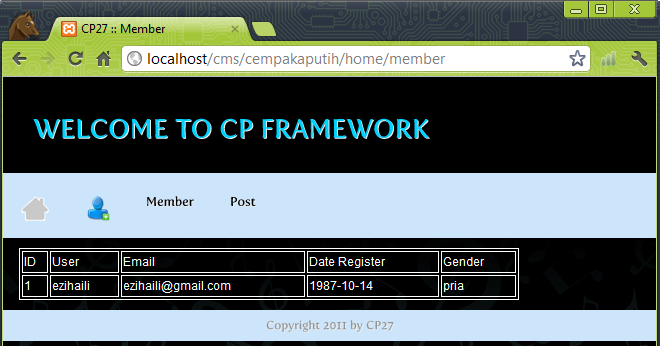
**Gambar 4.3. *Output class index controller***

Gambar tersebut menampilkan halaman *index*. Menunjukan bahwa *class index controller* yang berada pada *file index.php* di dalam *folder controller* mendapat *request* dari *url* untuk membuka tampilan (*view*) *index*.

****

**Gambar 4.4. *Output class home profile***

Gambar tersebut menampilkan halaman *profile*. Menunjukan bahwa *class profile controller* yang berada pada *file index.php* di dalam *folder controller* mendapat *request* dari *url* untuk membuka tampilan (*view*) *profile*.

****

**Gambar 4.5. Output class home member**

Gambar tersebut menampilkan halaman *member*. Menunjukan bahwa *class member controller* yang berada pada *file index.php* di dalam *folder controller* mendapat *request* dari *url* untuk membuka tampilan (*view*) *member*.