

# PHP Frameworks (2)

## Pertemuan - 13

Matakuliah Pemrograman Web (TIW213)  
Semester Genap 2015/2016

Yuan Lukito & Antonius Rachmat C



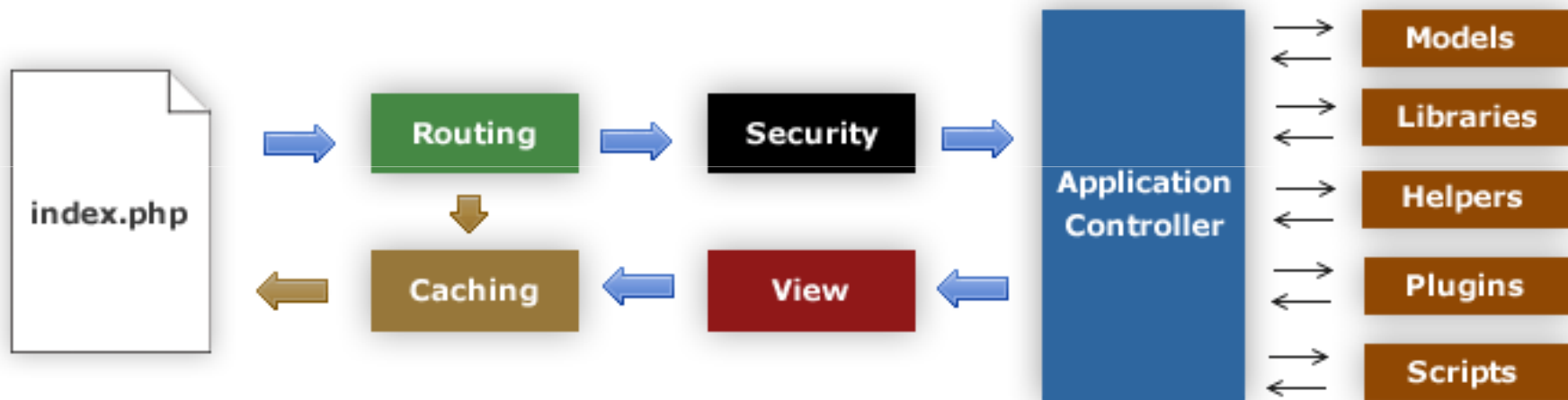
Prodi Teknik Informatika  
Fakultas Teknologi Informasi  
Universitas Kristen Duta Wacana



# Topik Bahasan

- Komponen Frameworks CodeIgniter
  - Routing, Caching, Security, Controllers, Models, Libraries, Helpers, Plugins, Scripts, View
- Konfigurasi CodeIgniter
  - Base URL, Encryption Key
  - Database
  - Auto load
  - Routes
- Penggunaan Input, Session dan Database pada CodeIgniter
- Review MVC dari studi kasus
  - Koleksi Buku

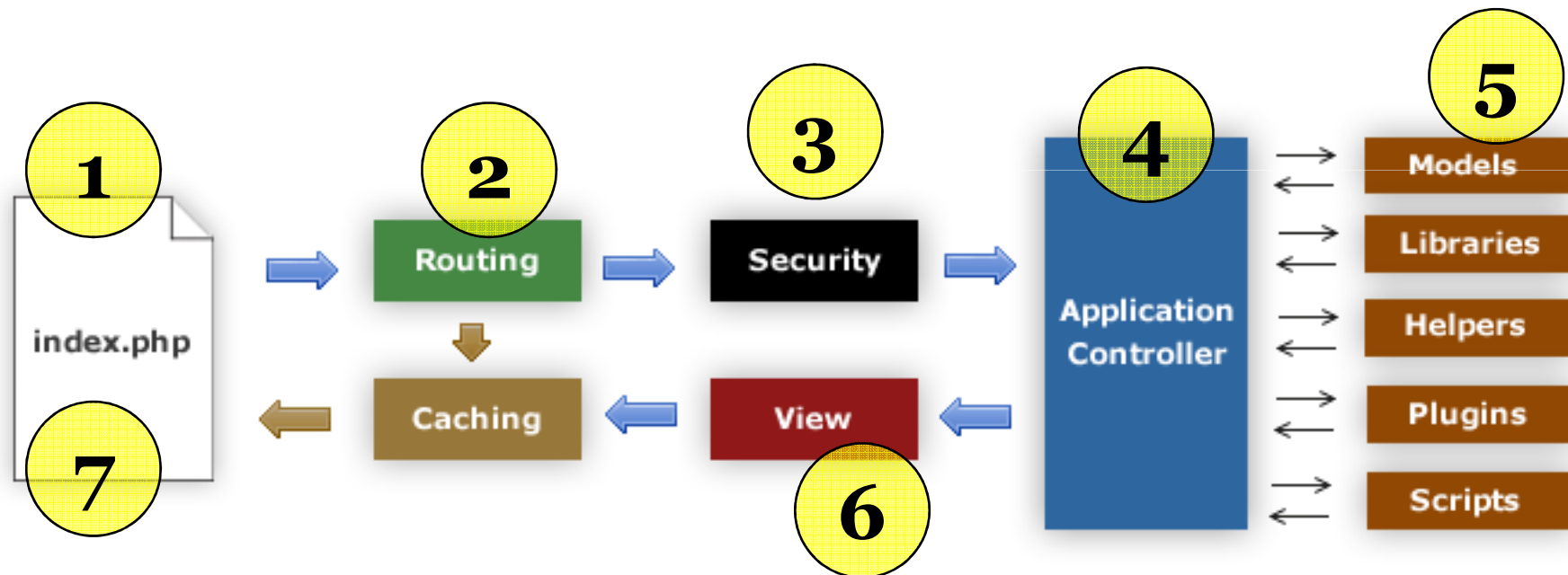
# Frameworks CodeIgniter



# Frameworks CodeIgniter

Request: **Tampilkan semua buku**

URL: **<http://localhost/index.php/buku/index>**



# CodeIgniter URL

- Menggunakan Segment-based URL
  - Controller / Method/ Parameter / ....<parameters>
- <http://perpustakaan.com/index.php/book/delete/10>
  - Controller: book
  - Method/Function: delete
  - Parameter: 10
- <http://perpustakaan.com/index.php/book/>
  - Controller: book
  - Method/Function: index
  - Parameter: - (tidak ada)
  - Sama seperti  
<http://perpustakaan.com/index.php/book/index>

# CodeIgniter URL: index.php

- “Menghilangkan” index.php di URL
  - Menggunakan .htaccess
- Apache, mod\_rewrite = enable (.htaccess)  
RewriteEngine On  
RewriteCond %{REQUEST\_FILENAME} !-f  
RewriteCond %{REQUEST\_FILENAME} !-d  
RewriteRule ^(.\*)\$ index.php/\$1 [L]
- <http://perpustakaan.com/book/>
  - Tanpa perlu ada index.php

# Routing & Caching

- Routing: memetakan request yang masuk
  - Controller, Method dan Parameter
- Caching: Menyimpan request yang sudah pernah dilakukan, sehingga request yang sama berikutnya bisa langsung diambil dari cache
  - Biasanya untuk static content atau dynamic content, dalam batas waktu tertentu
  - Dukungan: APC, File-based caching, Memcached, Wincache, Redis dan Dummy cache

# Caching

```
$this->load->driver('cache', array('adapter' =>
'apc', 'backup' => 'file'));

if ( ! $foo = $this->cache->get('foo'))
{
    echo 'Saving to the cache!<br />';
    $foo = 'foobarbaz!';

    // Save into the cache for 5 minutes
    $this->cache->save('foo', $foo, 300);
}

echo $foo;
```

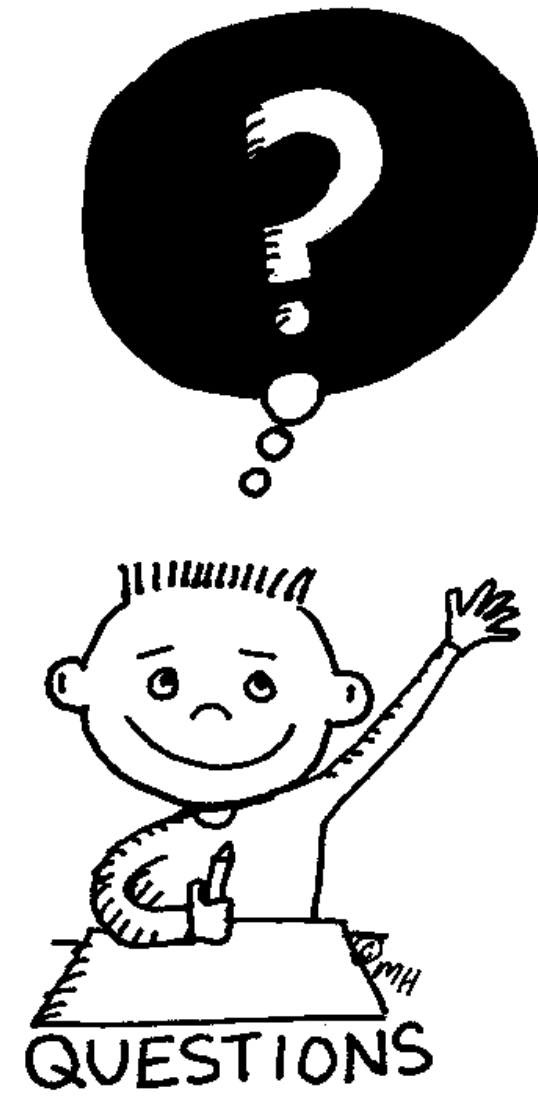


# Security

- URI Security
  - Membatasi jenis karakter yang dapat digunakan di URL
- Register\_globals
  - Saat inisialisasi, seluruh Global variable akan di-unset
  - Sama seperti register\_globals = off (default PHP)
- CSRF Protection
  - Proteksi dari Cross Site Request Forgery secara default
- XSS Filtering
  - Filtering Cross Site Scripting secara default
- Validasi input dan escape
  - Form validation library dan Input Class

# Library vs Helper

- Library berbentuk Class, sedangkan Helper hanya berisi kumpulan fungsi
- Libraries: Config, Email, Form Validation, Database, Pagination, Session, Security, File Uploading, ...
- Helpers: Array, Captcha, URL, Form, Date, XML, ...
- Helper
  - `$this->load->helper('url');`
  - `$this->load->helper('email');`
- Library
  - `$this->load->library('session');`
  - `$this->load->library('pagination');`



# Struktur CodeIgniter



application



system



user\_guide



.htaccess



index.php

- Model, View, Controller
- Cache, config, ...

# Struktur CodeIgniter

 application

 system

- Core, helpers, libraries

 user\_guide

 .htaccess

 index.php

# Struktur CodeIgniter

 application

 system

 user\_guide

 .htaccess

 index.php

- Documentation (tidak perlu diupload ke hosting)

# Struktur CodeIgniter



application



system



user\_guide



.htaccess



index.php

- URL tidak perlu menggunakan index.php

```
RewriteEngine On
```

```
RewriteCond
```

```
%{REQUEST_FILENAME} !-f
```

```
RewriteCond
```

```
%{REQUEST_FILENAME} !-d
```

```
RewriteRule ^(.*)$
```

```
index.php/$1 [L]
```

# Struktur CodeIgniter

 application

 system

 user\_guide

 .htaccess

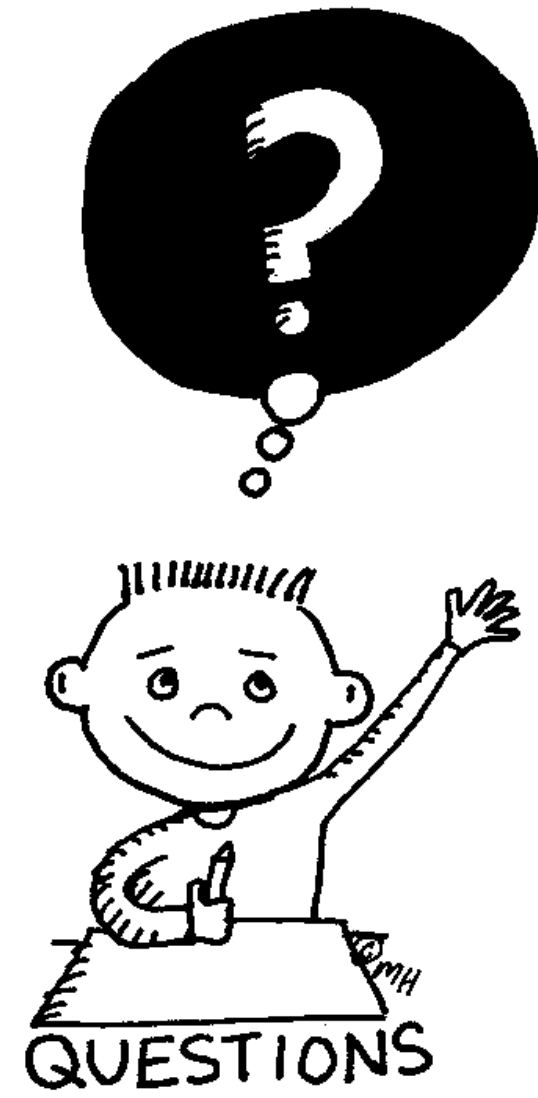
 index.php

- Semua request akan melalui index.php
- Inisialisasi system (core)



# Konfigurasi CodeIgniter

- application/config/config.php
  - Base URL
  - Encryption Key (Very long random String)
  - Session
  - XSS Filtering (TRUE)
  - CSRF Protection (TRUE)
- application/config/database.php
  - Host, username, password, database name
- application/config/autoload.php
  - Autoload model, libraries, helpers, drivers, config, languages
- application/config/routes.php
  - Menentukan controller default, 404 error page, ...



# Controller

- Berisi application logic, mengatur jalannya aplikasi
  - Menerima request, meminta data dari model, mengirimkan data ke view, meneruskan request ke controller lain
- Controller berupa Class, harus extends CI\_Controller

```
class Book extends CI_Controller {  
    .....  
}
```
- Satu Controller bisa memiliki lebih dari 1 methods
  - Public function dan private function

# Controller

- Public function dapat diakses (request), sedangkan private function hanya dapat digunakan di dalam Controller tersebut

```
class Book extends CI_Controller {  
    public function index(){  
        .....  
    }  
  
    private function _check_login(){  
        .....  
    }  
}
```

`http://<BASE URL>/book/index`

`http://<BASE URL>/book/_check_login`

# Model

- Mengelola dan melakukan semua operasi yang berkaitan dengan database atau sumber data lainnya
- Model seharusnya hanya dapat diakses oleh Controller
  - View tidak boleh mengakses Model secara langsung
- Model berupa Class, extends CI\_Model

```
class Book_model extends CI_Model {  
    .....  
}
```
- Model harus di-load dulu sebelum dapat digunakan
  - `$this->load->model('Book_model');`
  - Dapat dimasukkan ke config/autoload.php sehingga langsung di-load saat inisialisasi

# Model

- Model harus di-load dulu (atau masukkan autoload), kemudian baru dapat dipanggil semua public method yang ada di Model tersebut

```
class Book_model extends CI_Model {  
    public function get_all_books(){  
        ...  
    }  
}
```

```
.....  
$this->load->model('Book_model');  
$books = $this->Book_model->get_all_books();
```

# View

- Secara umum View berisi HTML, CSS dan JS
- View akan ditampilkan setelah dipanggil oleh Controller
  - `$this->load->view('all_book_view');`
- View bertugas menampilkan data yang diberikan oleh Controller
  - Data tersebut dikirimkan saat view dipanggil
  - `$this->load->view('all_book_view', $data);`
- Variabel `$data` berupa array. Misalkan ingin mengirimkan variable `$a = 10` dan `$b = 15` ke suatu View

```
$data[ 'a' ] = $a;
$data[ 'b' ] = $b;
$this->load->view( 'all_book_view', $data);
```

# View

- Pada View, variable yang dikirimkan tersebut dapat langsung diakses sesuai namanya

```
//Controller.....  
$data['a'] = 10; $data['b'] = 20;  
$this->load->view('some_view', $data);
```

```
//View (some_view.php)  
<p><?php echo $a; ?></p>  
<p><?php echo $b; ?></p>
```



# Membaca Input Form

- Untuk membaca `$_POST`, `$_GET` atau `$_REQUEST` dapat menggunakan Input Class
  - Diinisialisasi secara otomatis, tidak perlu di-load

//PHP umumnya

```
$username = $_POST[ 'username' ];
```

//CodeIgniter (POST)

```
$this->input->post( 'username' );
```

//GET

```
$this->input->get( 'title' );
```

# Menangani Session

- Untuk menangani Session, pertama kali harus load library Session terlebih dahulu
  - `$this->load->library('session');`
  - Atau bisa juga dimasukkan ke `config/autoload.php`
- Untuk mengakses suatu variable dalam Session

//PHP umumnya

```
$email = $_SESSION['email'];
```

//CodeIgniter (old method)

```
$email = $this->session->userdata('email');
```

//CodeIgniter (magic item)

```
$email = $this->session->email;
```

# Menangani Session

- Untuk mengecek apakah suatu variable berada di dalam Session

```
if($this->session->has_userdata('profilepic')){
```

```
.....
```

```
}
```

- Untuk menyimpan suatu variable ke dalam Session

```
//PHP umumnya
```

```
$checked = true;
```

```
$_SESSION['checked'] = $checked;
```

```
//CodeIgniter
```

```
$checked = true;
```

```
$this->session->set_userdata('checked', $checked);
```

# Menangani Session

- Menghapus suatu variable dari dalam Session

//PHP umumnya

```
unset($_SESSION['email']);
```

//CodeIgniter

```
$this->session->unset_userdata('email');
```

- Menghapus seluruh variable dalam Session

//PHP umumnya

```
session_destroy();
```

//CodeIgniter

```
$this->session->sess_destroy();
```

# Menangani Database

- Membuka koneksi ke database
  - `$this->load->database();`
  - Bisa dimasukkan ke config/autoload.php
- Menjalankan suatu query
  - `$result = $this->db->query('SELECT * FROM buku');`
- Query Binding
  - `$title = 'PHP';`
  - `$author = 'Lerdorf';`
  - `$sql = 'SELECT * FROM buku WHERE pengarang = ?`  
`AND judul = ?';`
  - `$this->db->query($sql, array($author, $title));`

# Menangani Database

- Membaca hasil query
  - Result arrays (multiple records)
  - Result rows (single records)
- Contoh (Result arrays)

```
$books = $this->db->query('SELECT * FROM buku');  
foreach($books->result() as $row){  
    echo $row->judul_buku;  
    echo $row->pengarang;  
    echo $row->tahun;  
}
```

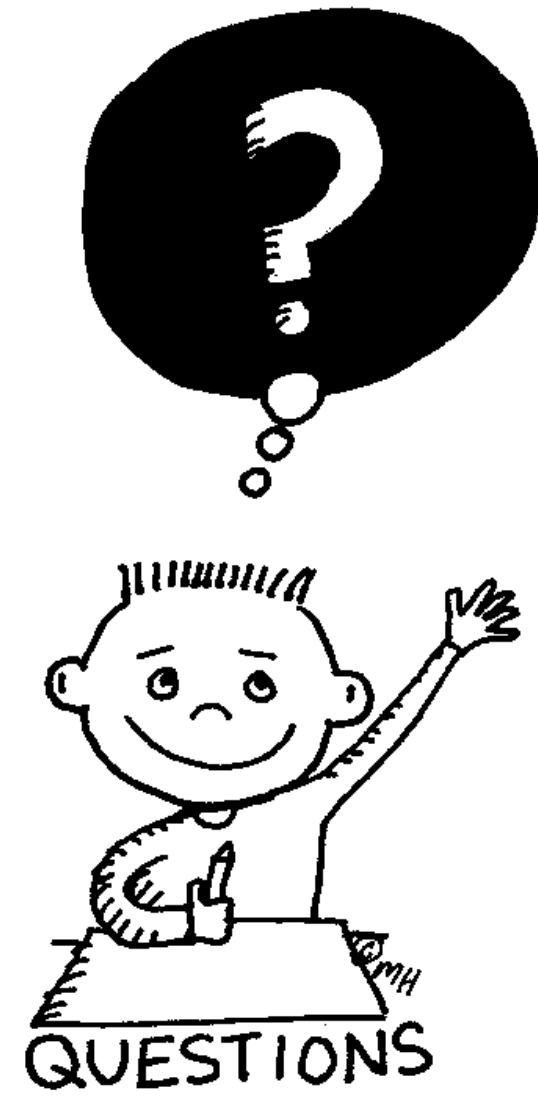
# Menangani Database

- Contoh (Result arrays)

```
$books = $this->db->query('SELECT * FROM buku');  
foreach($books->result_array() as $row){  
    echo $row['judul_buku'];  
    echo $row['pengarang'];  
    echo $row['tahun'];  
}
```

- Result row jika ingin mengambil satu record saja

```
$books = $this->db->query('SELECT * FROM buku');  
$book = $books->row(10);  
echo $book->judul_buku;  
echo $book->pengarang;
```





Misal Base URL = http://perpustakaan.com

```
class User extends CI_Controller {  
    public function index(){  
        http://perpustakaan.com/user/index  
    }  
  
    public function login(){  
        http://perpustakaan.com/user/login  
    }  
  
    public function logout(){  
        http://perpustakaan.com/user/logout  
    }  
}
```

Misal Base URL = http://perpustakaan.com

```
class Book extends CI_Controller {  
    public function index()  
    }
```

<http://perpustakaan.com/book/index>

```
    public function add()  
    }
```

<http://perpustakaan.com/book/add>

```
    public function edit($book_id)  
    }
```

<http://perpustakaan.com/book/edit/19>

# Controller

```
public function index(){  
    //cek apakah sudah login?  
    $this->_check_login();  
  
    //ambil semua data buku (Book_model)  
    $this->load->model('Book_model');  
    $books = $this->Book_model->get_all_books();  
  
    //kirirkan data buku tersebut ke view  
    $data['books'] = $books;  
    $this->load->view('book_index_view', $data);  
}
```

# Model

```
public function get_all_books(){  
    //ambil data dari tabel  
    $books = $this->db->query('SELECT  
        id_buku, judul_buku, pengarang,  
        penerbit, tahun, isbn FROM buku');  
    return $books;  
}
```

# View

```
<?php foreach($books->result() as $book): ?>
    <tr>
        <td><?php echo $book->judul_buku; ?></td>
        <td><?php echo $book->pengarang; ?></td>
        <td><?php echo $book->penerbit; ?></td>
        <td><?php echo $book->tahun; ?></td>
        <td><?php echo $book->isbn; ?></td>
        <td>
            <?php echo anchor('/book/edit/' . $book->id_buku, 'Ubah'); ?>
            <?php echo anchor('/book/delete/' . $book->id_buku, 'Hapus'); ?>
        </td>
    </tr>
<?php endforeach; ?>
```

---

# DEMO URL Shortener

# Database Configuration

```
$db['default']['hostname'] = 'localhost';  
$db['default']['username'] = 'root';  
$db['default']['password'] = '';  
$db['default']['database'] = 'shorty';  
$db['default']['dbdriver'] = 'mysql';  
$db['default']['dbprefix'] = '';  
$db['default']['pconnect'] = TRUE;  
$db['default']['db_debug'] = TRUE;  
$db['default']['cache_on'] = FALSE;  
$db['default']['cachedir'] = '';  
$db['default']['char_set'] = 'utf8';  
$db['default']['dbcollat'] = 'utf8_general_ci';  
$db['default']['swap_pre'] = '';  
$db['default']['autoinit'] = TRUE;  
$db['default']['stricton'] = FALSE;
```

## Table : urls

```
CREATE TABLE `urls` (  
  `id` INT(10) NOT NULL AUTO_INCREMENT,  
  `url_origin` VARCHAR(500) NULL DEFAULT NULL,  
  `url_short` VARCHAR(500) NULL DEFAULT NULL,  
  `created` TIMESTAMP NULL DEFAULT  
    CURRENT_TIMESTAMP,  
  `clicks` INT(10) NULL DEFAULT '1',  
  PRIMARY KEY (`id`) ) COLLATE='latin1_swedish_ci'  
ENGINE=InnoDB;
```



# Step 1 : Creating Views

- Displaying data from Controller
- Collect user input
- Create wireframe for designing

# Wireframes (index\_view)



A wireframe diagram of a web page layout for 'ShortMe'. The layout is contained within a large rectangular frame. At the top center is the title 'ShortMe' in a bold, black, sans-serif font. Below the title is a horizontal rectangular input field with the placeholder text 'Long URL' in a smaller, black, sans-serif font. Centered below the input field is a blue, underlined link labeled 'List All'. At the bottom center of the frame is the footer text 'ShortMe (C) 2012' in a small, black, sans-serif font.

**ShortMe**

Long URL

[List All](#)

ShortMe (C) 2012

# Wireframes (Result)

The wireframe shows a central rectangular box representing the application interface. Inside this box, the title "ShortMe" is centered at the top. Below the title is a text input field with the placeholder text "Long URL". Underneath the input field, the text "Short URL:" is followed by a blue, underlined link "http://localhost/shortme/abc". Below the link is another blue, underlined link "List All". At the bottom of the box, the text "ShortMe (C) 2012" is centered.

**ShortMe**

Long URL

Short URL: <http://localhost/shortme/abc>

[List All](#)

ShortMe (C) 2012

# Wireframes (List URL)

## ShortMe

Long URL

[List All](#)

URL	Short URL	Click	Delete
http://www.ukdw.ac.id	http://localhost/shortme/abc	3	delete
http://www.ukdw.ac.id	http://localhost/shortme/abc	3	delete
http://www.ukdw.ac.id	http://localhost/shortme/abc	3	delete

ShortMe (C) 2012

## Step 2 : Creating Models

- URLModel
  - Kumpulan dari data-data URL yang sudah dipendekkan

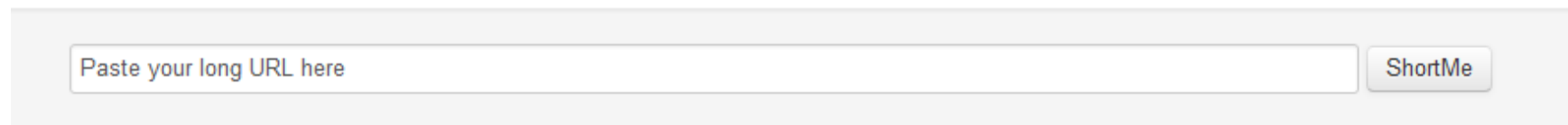
## Step 3 : Creating Controllers

- URLController
  - Handling URL operation

# Creating View (ShortMe)

Tampilan awal

**ShortMe!**



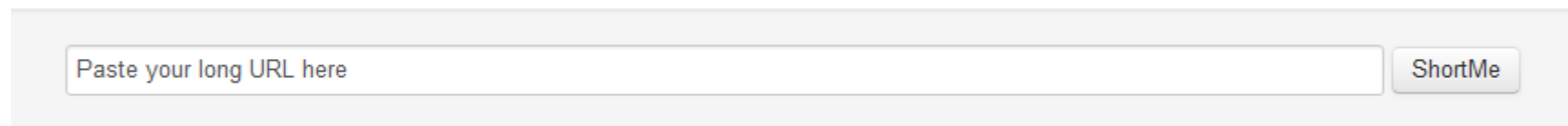
Paste your long URL here

ShortMe

[List All](#)

Tampilan setelah ada url yang dipendekkan

**ShortMe!**



Paste your long URL here

ShortMe

<http://localhost/shortme/vyps>

[List All](#)

# View List URL

## ShortMe!

Paste your long URL here

ShortMe

Hide

URL	Short	Created	Click	Delete
<a href="http://mediafiremoviez.com/brrip/gangs-of-wasseypur-2012-dvdrip-720p-800mb/?utm_source=rss&amp;utm_medium=rss&amp;utm_campaign=gangs-of-wasseypur-2012-dvdrip-720p-800mb">http://mediafiremoviez.com/brrip/gangs-of-wasseypur-2012-dvdrip-720p-800mb/?utm_source=rss&amp;utm_medium=rss&amp;utm_campaign=gangs-of-wasseypur-2012-dvdrip-720p-800mb</a>	<a href="http://localhost/shortme/pg5r">http://localhost/shortme/pg5r</a>	2012-08-12 01:07:29	1	<a href="#">Delete</a>
<a href="http://ganool.com/kyou-no-go-no-ni-ova">http://ganool.com/kyou-no-go-no-ni-ova</a>	<a href="http://localhost/shortme/hpg1">http://localhost/shortme/hpg1</a>	2012-08-12 00:51:10	1	<a href="#">Delete</a>
<a href="http://ganool.com/war-of-the-dead-2011-bluray-720p-500mb-ganool">http://ganool.com/war-of-the-dead-2011-bluray-720p-500mb-ganool</a>	<a href="http://localhost/shortme/h7t0">http://localhost/shortme/h7t0</a>	2012-08-12 00:47:10	1	<a href="#">Delete</a>
<a href="http://ganool.com/maji-de-watashi-ni-koi-shinasai">http://ganool.com/maji-de-watashi-ni-koi-shinasai</a>	<a href="http://localhost/shortme/51jl">http://localhost/shortme/51jl</a>	2012-08-12 00:45:43	1	<a href="#">Delete</a>
<a href="http://codeigniter.com/user_guide/database/results.html">http://codeigniter.com/user_guide/database/results.html</a>	<a href="http://localhost/shortme/oj3s">http://localhost/shortme/oj3s</a>	2012-08-12 00:40:05	2	<a href="#">Delete</a>
<a href="http://lecturer.ukdw.ac.id/anton/v2/pbk.php">http://lecturer.ukdw.ac.id/anton/v2/pbk.php</a>	<a href="http://localhost/shortme/o8vz">http://localhost/shortme/o8vz</a>	2012-08-11 23:53:28	4	<a href="#">Delete</a>
<a href="http://lecturer.ukdw.ac.id/anton/v2/progweb.php">http://lecturer.ukdw.ac.id/anton/v2/progweb.php</a>	<a href="http://localhost/shortme/77cb">http://localhost/shortme/77cb</a>	2012-08-11 23:39:07	2	<a href="#">Delete</a>



# Routes and Config

Jika url yang diinputkan oleh user tidak ada nama controllernya maka lempar ke url/decode dan kemudian redirect ke halaman aslinya dan update clicks +1

```
$route['default_controller'] = "url";  
$route['url/add'] = "url/add";  
$route['url/del'] = "url/del";  
$route['404_override'] = 'url/decode';
```

```
*/  
$config['base_url'] = 'http://localhost/shortme/';  
/*  
|-----  
-----
```

Misalnya: <http://localhost/shortme/xyz>

# Kode Controller

```
class URL extends CI_Controller {

    public function __construct()
    {
        parent::__construct();
        $this->load->model('url_model');
    }

    public function index($x='')
    {
        $data['title'] = "ShortMe! - A URL Shortener";
        $this->load->view('index_view', $data);
    }
}
```

```
public function getUrl () {
    $data['title'] = 'ShortMe! - View All URLs';
    $data['all'] = $this->url_model->getUrl();
    $this->load->view('index_view', $data);
}

public function decode() {
    $shorty=$this->uri->segment(1);
    redirect($this->url_model->getUrlOri($shorty));
}

public function del($id) {
    if((int)$id > 0){
        $this->url_model->delete_url($id);
        redirect("url/getURL");
    }
}
```

# Controller add url baru

```
public function add() {
    $this->load->helper('form');
    $this->load->library('form_validation');
    $this->form_validation->set_rules('url_origin', 'URL cannot
be empty', 'required');
    |
    if ($this->form_validation->run() === FALSE)
    {
        $this->load->view('index_view');
    }
    else
    {
        $urlinput=mysql_real_escape_string($_POST['url_origin'
]);

        $id=rand(10000,99999);
        $shorturl=base_convert($id,20,36);
        $this->url_model->add_url($urlinput,$shorturl);
        $data['title'] = "ShortMe! - Add URL";
        $data['urlhasil'] = $shorturl;
        $this->load->view('index_view',$data);
        //redirect("url/getURL");
    }
}
```

# Model

```
class Url_model extends CI_Model {

    public function __construct()
    {
        $this->load->database();
    }

    public function add_url($url1,$url2)
    {
        $sql = "insert into urls(url_origin,url_short) values
(?,?)";
        $query = $this->db->query($sql,array($url1,$url2));
    }

    public function getURL(){
        $hasil = $this->db->query("select * from urls order by
created desc");
        return $hasil;
    }
}
```



“Everything is hard before it is easy”  
Johan Wolfgang von Goethe

# Pertemuan 14

## Presentasi Akhir Project Web