

Contoh Penerapan MVC pada Codegniter

Studi Kasus: Pengelolaan data Buku

Database: Ci

Tabel: Tabel_buku

#	Column	Type	Collation	Attributes	Null	Default
1	<u>kode_buku</u>	char(10)	latin1_swedish_ci		No	None
2	judul_buku	char(30)	latin1_swedish_ci		No	None
3	penerbit	char(30)	latin1_swedish_ci		Yes	NULL
4	isbn	char(30)	latin1_swedish_ci		Yes	NULL
5	stok_buku	int(11)			Yes	NULL

Konfigurasi CodeIgniter

Setelah persiapan awal selesai, lakukan persiapan berikutnya yaitu konfigurasi CodeIgniter untuk aplikasi perpustakaan yang akan kita buat. Beberapa file konfigurasi yang perlu kita rubah adalah:

- File 'autoload.php', untuk mengatur packages, libraries, file helper, file custom config, file language dan models untuk di-load otomatis secara default.
Lokasi: htdocs\workshop_ci\application\config\autoload.php

Konfigurasi untuk library 'database'

```
$autoload['libraries'] = array('database');
```

Konfigurasi untuk helper 'form', 'url'

```
$autoload['helper'] = array('form','url');
```

- File 'config.php', untuk mengatur beberapa konfigurasi utama dalam aplikasi yang kita buat.
Lokasi: htdocs\workshop_ci\application\config\config.php

Konfigurasi 'Base Site URL' = url web yang kita gunakan

```
$config['base_url'] = 'http://localhost/workshop_ci/';
```

- File 'database.php', untuk mengatur konfigurasi MySQL Server dan database yang akan kita gunakan.
Lokasi: htdocs\workshop_ci\application\config\database.php

Konfigurasi server MySQL & database

```
$db['default']['hostname'] = 'localhost';  
$db['default']['username'] = 'root';  
$db['default']['password'] = '';  
$db['default']['database'] = 'ci';
```

```
$db['default']['dbdriver'] = 'mysql';
```

- File 'routes.php', untuk mengatur routing dari CodeIgniter.
Lokasi: htdocs\workshop_ci\application\config\routes.php

Konfigurasi default_controller: 'perpustakaan ' agar dipanggil ketika 'Base Site URL' di load

```
$route['default_controller'] = "perpustakaan";
```

Contoh penerapan MVC

Membuat Controller (C)

Dalam konsep pemrograman MVC, karakter 'C' itu merupakan 'Controller' yang menjembatani antara model dan view. Controller berisi perintah-perintah yang berfungsi untuk memproses suatu data (dengan me-load model) dan mengirimkan hasilnya ke halaman web (dengan me-load view). Dalam aplikasi perpustakaan ini kita membuat satu file controller yaitu controller.php sebagai controller utama. Berikut langkahnya:

- Buat file controller dengan nama file 'perpustakaan.php'
Lokasi: htdocs\workshop_ci\application\controllers\perpustakaan.php

```
<?php
class Perpustakaan extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
    }
    function index()
    {
        $this->load->view('perpustakaan');
    }
    function tampil_buku()
    {
        $kode='all';
        $this->load->model('buku');
        $data_buku['databuku']=$this->buku->tampil_data_buku($kode);
        $this->load->view('tampilbuku',$data_buku);
    }
    function tambah_buku()
    {
        $this->load->view('tambahbuku');
    }
    function simpan_buku()
    {
        if (isset($_POST['mysubmit'])){
            $data = array(
                'kode_buku' => $this->input->post('kode_buku'),
                'judul_buku' => $this->input->post('judul_buku'),
                'penerbit' => $this->input->post('penerbit'),
                'isbn' => $this->input->post('isbn'),
                'stok_buku' => $this->input->post('stok_buku'),
                'mode' => $this->input->post('mode')
            );
            $this->load->model('buku');
            $hasil=$this->buku->simpan_data_buku($data);
            if ($hasil){
                echo "Simpan data berhasil";
            }else{
                echo "Simpan data gagal";
            }
            echo "<br/>";
            echo anchor('perpustakaan', 'Kembali');
        }
    }
    function koreksi_buku($kd)
    {
        $this->load->model('buku');
        $data_buku['databuku']=$this->buku->tampil_data_buku($kd);
        $this->load->view('koreksibuku',$data_buku);
    }
}
```

```

}

function konfirmasi_hapus_buku($kd)
{
    $this->load->model('buku');
    $data_buku['databuku']=$this->buku->tampil_data_buku($kd);
    $this->load->view('konfirmasihapus',$data_buku);
}
function hapus_buku($kd)
{
    $this->load->model('buku');
    $hasil=$this->buku->hapus_data_buku($kd);
    if ($hasil){
        echo "Hapus data berhasil";
    }else{
        echo "Hapus data gagal";
    }
    echo "<br/>";
    echo anchor('perpustakaan', 'Kembali');
}
}
?>

```

Penjelasan:

- **function __construct()**, controller untuk memanggil constructor pada class parent. Fungsi constructor untuk melakukan set nilai default atau menjalankan proses default ketika class di load.
- **function index()**, controller untuk memanggil halaman utama. Fungsi ini akan me-load view 'perpustakaan.php' dengan sintak: `$this->load->view('perpustakaan')` (lihat pada bagian bagan alur).
- **function tampil_buku()**, controller untuk melayani proses tampil buku. Fungsi ini akan me-load view model 'buku' dengan sintak: `$this->load->model('buku')` kemudian memanggil fungsi `tampil_data_buku($kode)` yang terdapat pada model 'buku' lalu mengirimkan hasilnya ke view dengan me-load view 'tampilbuku.php' dan ditampilkan ke browser dengan sintak: `$this->load->view('tampilbuku',$data_buku)`.
- **function tambah_buku()**, controller untuk melayani proses tambah buku. Fungsi ini akan memanggil view 'tambahbuku.php'.
- **function simpan_buku()**, controller untuk melayani proses simpan buku. Fungsi ini akan menangkap data yang dikirim user melalui form, baik tambah buku atau koreksi buku melalui variabel POST kemudian memanggil model 'buku.php' dan menjalankan fungsi `simpan_data_buku($data)` dengan nilai return status simpan buku berhasil atau tidak.
- **function koreksi_buku(\$kd)**, controller untuk melayani proses koreksi buku. Fungsi ini akan me-load view model 'buku' dan menjalankan fungsi `tampil_data_buku($kode)` lalu mengirimkan hasilnya ke view dengan me-load view 'koreksibuku.php' dan ditampilkan ke browser.
- **function konfirmasi_hapus_buku(\$kd)**, controller untuk melayani proses konfirmasi hapus buku. Fungsi ini akan me-load view model 'buku' dan menjalankan fungsi `tampil_data_buku($kode)` lalu mengirimkan hasilnya ke view dengan me-load view 'konfirmasihapus.php' dan ditampilkan ke browser.
- **function hapus_buku(\$kd)**, controller untuk melayani proses hapus buku. Fungsi ini akan memanggil model 'buku.php' dan menjalankan fungsi

hapus_data_buku(\$data) dengan nilai return status simpan buku berhasil atau tidak.

Membuat Model (M)

Model mewakili struktur data. Model berisi fungsi-fungsi untuk pengelolaan basis data seperti ambil tambah, koreksi dan hapus data. Karena tulisan ini membahas tentang data buku, maka fungsi-fungsi untuk memanipulasi tabel buku kita buat pada model 'buku.php'. Berikut langkahnya:

- Buat file model dengan nama file 'buku.php'
Lokasi: htdocs\workshop_ci\application\models\buku.php

```
<?php
class Buku extends CI_Model
{
    public function tampil_data_buku($kode)
    {
        if ($kode=='all'){
            $hasil=$this->db->get('tabel_buku');
        }else{
            $this->db->where('kode_buku',$kode);
            $hasil=$this->db->get('tabel_buku');
        }
        return $hasil->result();
    }
    public function simpan_data_buku($data)
    {
        if ($data['mode']=='baru'){
            unset($data['mode']);
            $hasil=$this->db->insert('tabel_buku', $data);
        }else{
            unset($data['mode']);
            $this->db->where('kode_buku',$data['kode_buku']);
            $hasil=$this->db->update('tabel_buku', $data);
        }
        return $hasil;
    }
    public function hapus_data_buku($kode)
    {
        $this->db->where('kode_buku', $kode);
        $hasil=$this->db->delete('tabel_buku');
        return $hasil;
    }
}
?>
```

Penjelasan:

- **public function tampil_data_buku(\$kode)**, model untuk proses tampil data. Fungsi ini akan memanggil \$this->db->get('tabel_buku') yang terdapat dalam Database class sama seperti perintah query "Select * from tabel_buku". Dan apabila menampilkan data dengan kriteria kode buku tertentu, maka fungsi database class yang digunakan adalah \$this->db->where('kode_buku',\$kode) lalu \$this->db->get('tabel_buku'). Perintah tersebut akan sama dengan perintah query "Select * from tabel_buku where kode_buku=\$kode".
- **public function simpan_data_buku(\$data)**, model untuk proses simpan data baik data baru ataupun data hasil koreksi. Sintak untuk simpan data

baru adalah `$this->db->insert('tabel_buku', $data)`. Sintak tersebut sama dengan perintah query `"insert into tabel_buku values()"`. Sedang sintak untuk simpan data hasil koreksi adalah `$this->db->where('kode_buku', $data['kode_buku'])` dan `$hasil=$this->db->update('tabel_buku', $data)`. Sintak tersebut sama dengan perintah query `"Update tabel_buku set field=value where kode_buku=$data['kode_buku']"`.

- **public function hapus_data_buku(\$kode)**, model untuk proses hapus buku. Fungsi ini melakukan hapus data menggunakan sintak `$this->db->where('kode_buku', $kode)` dan `$hasil=$this->db->delete('tabel_buku')`. Sintak tersebut adalah sama dengan perintah query `"Delete from tabel_buku where kode_buku=$kode"`.

Membuat View (V)

View adalah bagian yang mengatur tampilan untuk dilihat oleh pengguna melalui web browser, umumnya view ini adalah file HTML biasa. Untuk menangani beberapa transaksi dalam aplikasi perpustakaan- data buku ini, terdapat beberapa file view yang harus kita siapkan, yaitu:

- File view 'perpustakaan.php', sebagai halaman awal aplikasi perpustakaan.
Lokasi: `htdocs\workshop_ci\application\views\perpustakaan.php`

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title>Perpustakaan Online</title>
<meta name="" content="">
</head>
<body>
<h1>Perpustakaan</h1>
<ul>
<li><?php echo anchor('perpustakaan/tampil_buku', 'Tampil Buku');?></li>
<li><?php echo anchor('perpustakaan/tambah_buku', 'Tambah Buku');?></li>
</ul>
</body>
</html>
```

- File view 'tampilbuku.php', sebagai halaman untuk menampilkan daftar buku.
Lokasi: `htdocs\workshop_ci\application\views\ tampilbuku.php`

```
<?php
echo "<h2>Tampil data buku</h2>";
echo "<table border=1>";
echo "<tr><th>kode buku</th> <th>judul buku</th> <th> penerbit </th> <th> isbn
</th> <th> stok buku </th><th>&nbsp;</th></tr>";
foreach ($databuku as $isi)
{
    echo "<tr>";
    echo "<td>$isi->kode_buku</td>";
    echo "<td>$isi->judul_buku</td>";
    echo "<td>$isi->penerbit</td>";
    echo "<td>$isi->isbn</td>";
    echo "<td>$isi->stok_buku</td>";
    echo "<td>";
    echo anchor('perpustakaan/koreksi_buku/'.$isi->kode_buku, 'Edit')." | ";
    echo anchor('perpustakaan/konfirm_hapus_buku/'.$isi->kode_buku, 'Hapus');
    echo "</td>";
}
```

```

        echo "</tr>";
    }
    echo "</table>";
    echo "<p>".anchor('perpustakaan', 'Kembali')."</p>";
?>

```

- File view 'tambahbuku.php', sebagai halaman form untuk menambah buku.
Lokasi: htdocs\workshop_ci\application\views\ tambahbuku.php

```

<?php
echo "<h2>Tambah data buku</h2>";
echo form_open('perpustakaan/simpan_buku');
echo "<table align='left' width='100%'>";

$field1=array('name' => 'kode_buku','size'=>'15');
echo "<tr><td width='10%'>Kode Buku</td><td width='1%'>
:</td><td>".form_input($field1)."</td></tr>";

$field2=array('name' => 'judul_buku','size'=>'30');
echo "<tr><td>Judul</td><td> :</td><td>".form_input($field2)."</td></tr>";

$field3=array('name' => 'penerbit','size'=>'30');
echo "<tr><td>Penerbit</td><td> :</td><td>".form_input($field3)."</td></tr>";

$field4=array('name' => 'isbn','size'=>'30');
echo "<tr><td>ISBN</td><td> :</td><td>".form_input($field4)."</td></tr>";

$field5=array('name' => 'stok_buku','size'=>'11');
echo "<tr><td>Stock</td><td> :</td><td>".form_input($field5)."</td></tr>";

echo form_hidden('mode', 'baru');
echo "<tr><td>".form_submit('mysubmit','Simpan')."</tr></td>";

echo "</table>";
echo form_close();
echo "<p>".anchor('perpustakaan', 'Kembali')."</p>";
?>

```

- File view 'koreksibuku.php', sebagai halaman form untuk koreksi buku.
Lokasi: htdocs\workshop_ci\application\views\ koreksibuku.php

```

<?php
echo "<h2>Koreksi data buku</h2>";
echo form_open('perpustakaan/simpan_buku');
echo "<table align='left' width='100%'>";
$field1=array('name' => 'kode_buku','size'=>'15','value'=>$databuku[0]->
kode_buku);
echo "<tr><td width='10%'>Kode Buku</td><td width='1%'>
:</td><td>".form_input($field1)."</td></tr>";

$field2=array('name' => 'judul_buku','size'=>'30','value'=>$databuku[0]->
judul_buku);
echo "<tr><td>Judul</td><td> :</td><td>".form_input($field2)."</td></tr>";

$field3=array('name' => 'penerbit','size'=>'30','value'=>$databuku[0]->penerbit);
echo "<tr><td>Penerbit</td><td> :</td><td>".form_input($field3)."</td></tr>";

$field4=array('name' => 'isbn','size'=>'30','value'=>$databuku[0]->isbn);
echo "<tr><td>ISBN</td><td> :</td><td>".form_input($field4)."</td></tr>";

$field5=array('name' => 'stok_buku','size'=>'11','value'=>$databuku[0]->
stok_buku);
echo "<tr><td>Stock</td><td> :</td><td>".form input($field5)."</td></tr>";

```

```

echo form_hidden('mode', 'koreksi');
echo "<tr><td>".form_submit('mysubmit','Simpan')."</tr></td>" ;

echo "</table>";
echo form_close();
echo "<p>".anchor('perpustakaan', 'Kembali')."</p>";
?>

```

- File view 'konfirmasihapus.php', sebagai halaman konfirmasi untuk hapus buku.
Lokasi: htdocs\workshop_ci\application\views\ konfirmasihapus.php

```

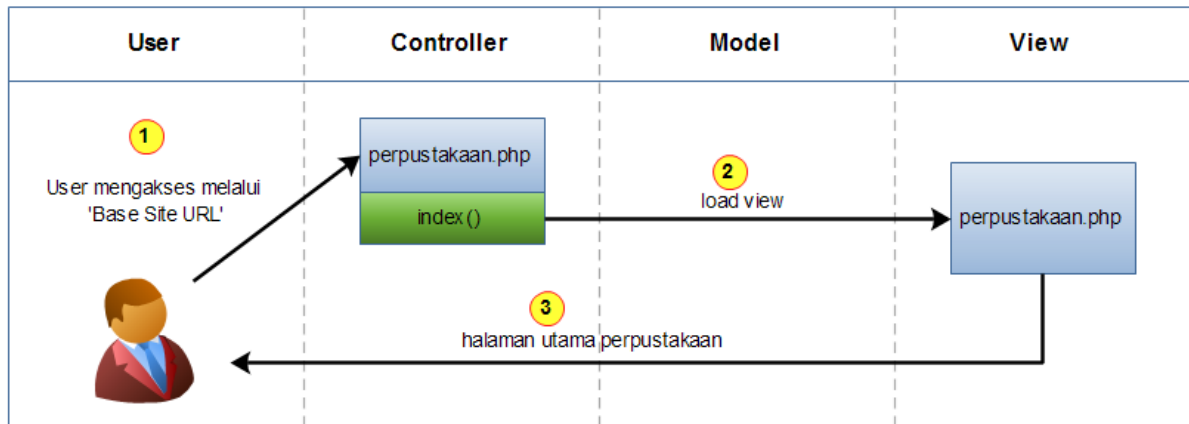
<?php
echo "<h2>Konfirmasi hapus data buku</h2>";
echo "<p>Apakah data buku berikut ini akan dihapus?</p>";
echo "<table border=1>";
echo "<tr><th>kode buku</th> <th>judul buku</th> <th> penerbit </th> <th> isbn
</th> <th> stok buku </th></tr>";
foreach ($databuku as $isi)
{
    echo "<tr>";
        echo "<td>$isi->kode_buku</td>";
        echo "<td>$isi->judul_buku</td>";
        echo "<td>$isi->penerbit</td>";
        echo "<td>$isi->isbn</td>";
        echo "<td>$isi->stok_buku</td>";
    echo "</tr>";
}
echo "</table>";
echo "<p>".anchor('perpustakaan/hapus_buku/'.$isi->kode_buku, 'Ya')." | ";
echo anchor('perpustakaan', 'Tidak')."</p>";
?>

```

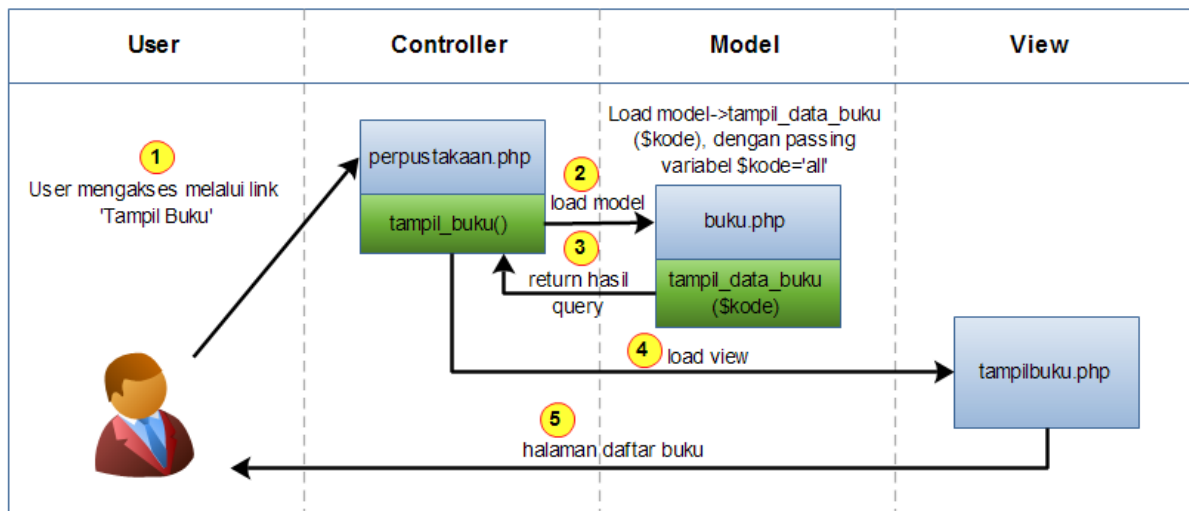

Bagan Alur setiap proses dalam MVC

Bagan alur dibawah ini menjelaskan alur data dari setiap proses dalam aplikasi perpustakaan.

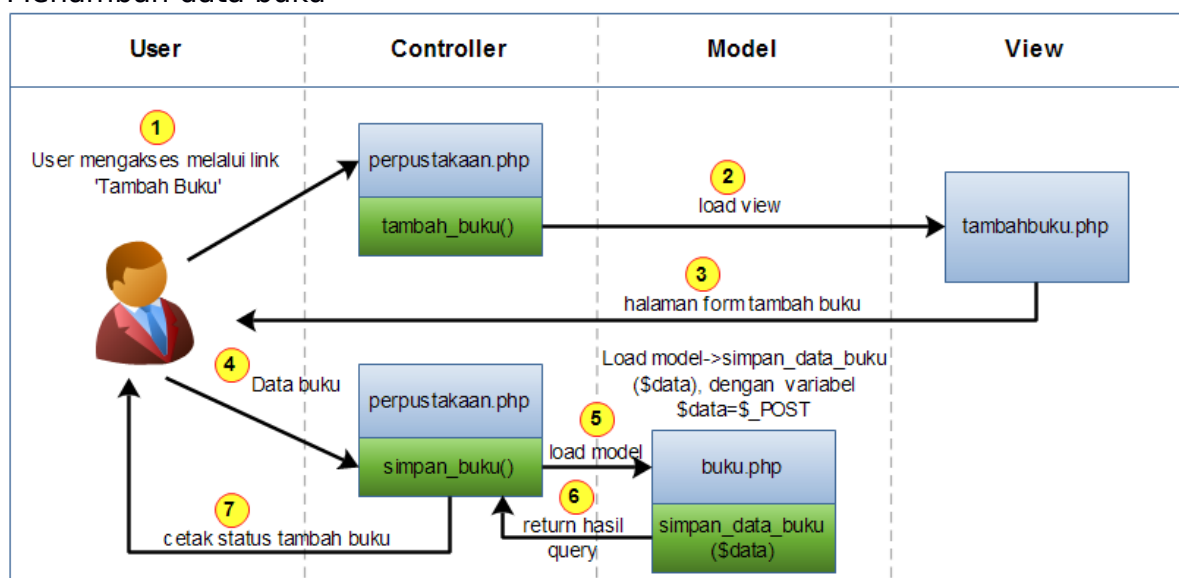
- Mengakses halaman utama



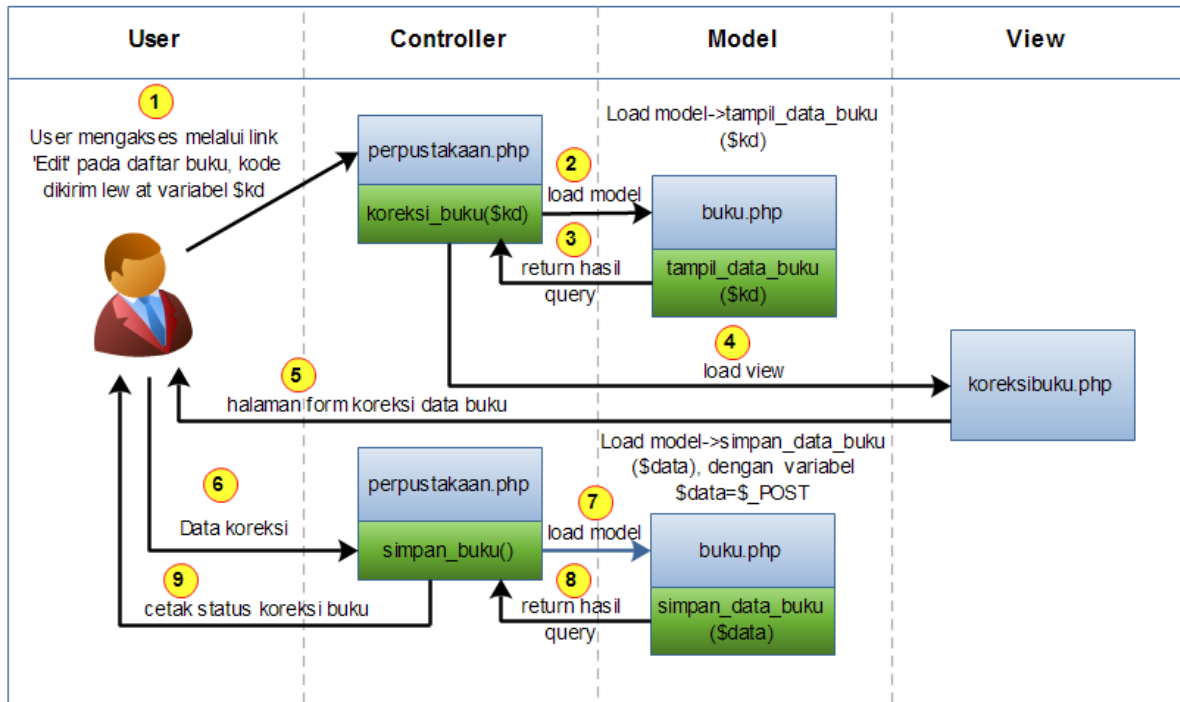
- Melihat daftar buku



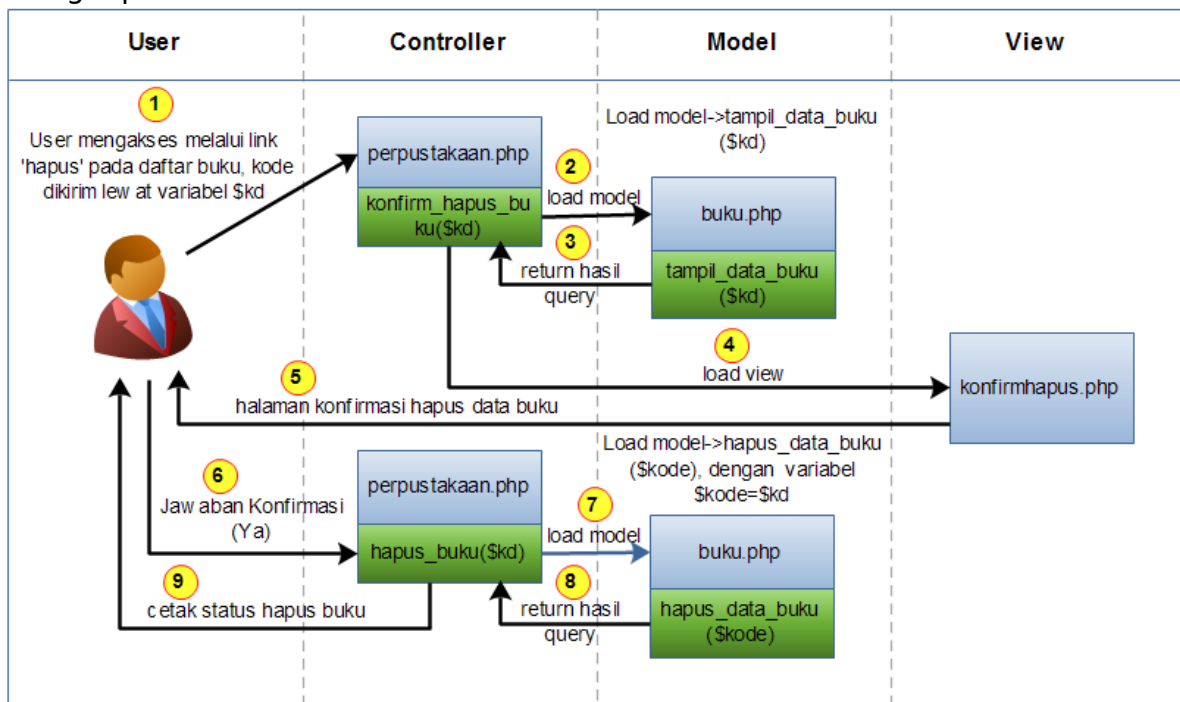
- Menambah data buku



- Mengkoreksi data buku



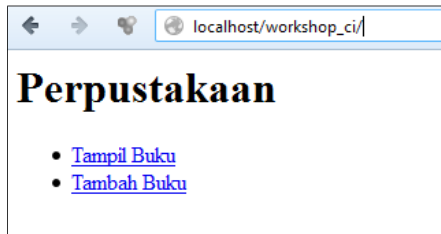
- Menghapus data buku



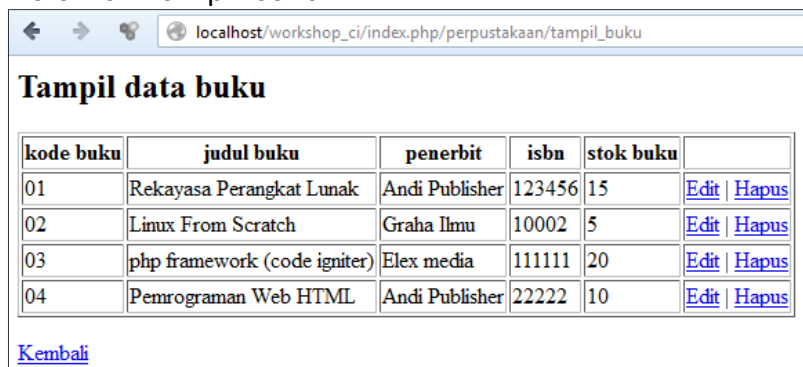
Tampilan Aplikasi

Buka aplikasi perpustakaan dengan menggunakan browser dengan alamat:
http://localhost/workshop_ci/

- Halaman utama



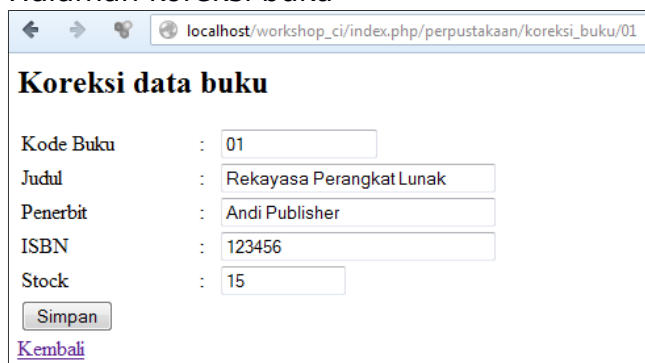
- Halaman tampil buku



- Halaman tambah buku



- Halaman koreksi buku



- Halaman hapus buku

localhost/workshop_ci/index.php/perpustakaan/konfirm_hapus_buku/01

Konfirmasi hapus data buku

Apakah data buku berikut ini akan dihapus?

kode buku	judul buku	penerbit	isbn	stok buku
01	Rekayasa Perangkat Lunak	Andi Publisher	123456	15

[Ya](#) | [Tidak](#)