

DOKUMENTASI APLIKASI CODEIGNITER 3

Daftar Isi

1. Pendahuluan
 2. Struktur Aplikasi
 3. Instalasi dan Konfigurasi
 4. Fitur Aplikasi
 5. Penjelasan Kode
 6. Cara Penggunaan
 7. Pengembangan Lebih Lanjut
-

1. Pendahuluan

Dokumentasi ini menjelaskan tentang aplikasi yang dibangun menggunakan framework CodeIgniter 3. Aplikasi ini merupakan sistem sederhana yang menerapkan konsep CRUD (Create, Read, Update, Delete), autentikasi pengguna, Pemrograman Berorientasi Objek (OOP), routing, dan templating.

1.1 Tujuan Aplikasi

Aplikasi ini dirancang untuk:

- Menunjukkan implementasi dasar dari framework CodeIgniter 3
- Memperlihatkan penerapan CRUD pada data
- Mendemonstrasikan sistem autentikasi sederhana
- Menerapkan konsep OOP dalam pengembangan aplikasi web
- Menggunakan sistem routing dan templating untuk menghasilkan aplikasi yang terstruktur

1.2 Teknologi yang Digunakan

- **CodeIgniter 3:** Framework PHP yang digunakan sebagai dasar aplikasi
 - **MySQL:** Sistem database yang digunakan untuk menyimpan data
 - **Bootstrap 5:** Framework CSS untuk tampilan antarmuka
 - **Font Awesome:** Library ikon untuk memperkaya tampilan
 - **HTML/CSS/JavaScript:** Bahasa dasar pengembangan web
-

2. Struktur Aplikasi

Struktur folder dan file pada aplikasi mengikuti standar dari framework CodeIgniter 3 dengan beberapa penyesuaian untuk kebutuhan aplikasi ini.

2.1 Struktur Folder Utama

```
/application/  
  /config/          - File konfigurasi aplikasi
```

/controllers/	- Controller untuk logika aplikasi
/models/	- Model untuk interaksi database
/views/	- View untuk tampilan
/auth/	- Tampilan autentikasi
/dashboard/	- Tampilan dashboard
/items/	- Tampilan CRUD item
/templates/	- Template layout dasar

2.2 File-File Penting

2.2.1 File Konfigurasi

- **config.php**: Konfigurasi dasar aplikasi
- **autoload.php**: Konfigurasi autoload library, helper, dan model
- **database.php**: Konfigurasi koneksi database
- **routes.php**: Konfigurasi routing URL

2.2.2 File Controller

- **Auth.php**: Controller untuk proses autentikasi
- **Dashboard.php**: Controller untuk halaman dashboard
- **Items.php**: Controller untuk manajemen data item

2.2.3 File Model

- **User_model.php**: Model untuk manajemen data pengguna
- **Item_model.php**: Model untuk manajemen data item

2.2.4 File View

- Views untuk autentikasi: login.php, register.php
- Views untuk dashboard: index.php
- Views untuk item: index.php, create.php, edit.php, view.php
- Views untuk template: header.php, footer.php, sidebar.php

3. Instalasi dan Konfigurasi

3.1 Persyaratan Sistem

- PHP versi 5.6 atau lebih tinggi
- MySQL versi 5.1 atau lebih tinggi
- Web server (Apache/Nginx)
- Ekstensi PHP yang aktif: session, mysqli, mbstring

3.2 Langkah-Langkah Instalasi

1. Persiapan Database:

- Buat database baru dengan nama `ci3_app`
- Aplikasi akan membuat tabel secara otomatis saat pertama kali dijalankan

2. Konfigurasi Aplikasi:

- Ekstrak/salin seluruh file aplikasi ke direktori web server Anda
- Buka file `application/config/database.php`
- Sesuaikan konfigurasi database dengan pengaturan lokal:

```
$db['default'] = array(  
    'hostname' => 'localhost',      // Sesuaikan dengan host database Anda  
    'username' => 'root',          // Sesuaikan dengan username database Anda  
    'password' => '',              // Sesuaikan dengan password database Anda  
    'database' => 'ci3_app',       // Nama database yang telah dibuat  
    ...  
);
```

- Buka file `application/config/config.php`
- Sesuaikan base URL dengan lokasi instalasi aplikasi Anda:

```
$config['base_url'] = 'http://localhost/ci3_app/'; // Ganti dengan URL aplikasi Anda
```

3. Konfigurasi `.htaccess`:

- Pastikan file `.htaccess` berada di root direktori aplikasi (untuk Apache)
- Jika menggunakan Nginx, sesuaikan konfigurasi server block

4. Akses Aplikasi:

- Buka browser dan akses URL aplikasi (sesuai dengan `base_url` yang dikonfigurasi)
- Aplikasi akan otomatis membuat struktur tabel yang dibutuhkan

3.3 Struktur Database

Aplikasi menggunakan dua tabel utama:

1. Tabel `users`:

- `id`: Primary key, auto increment
- `username`: Nama pengguna
- `email`: Alamat email pengguna
- `password`: Password terenkripsi
- `created_at`: Waktu pembuatan akun
- `updated_at`: Waktu terakhir update akun

2. Tabel `items`:

- `id`: Primary key, auto increment
- `name`: Nama item
- `description`: Deskripsi item
- `price`: Harga item
- `user_id`: Foreign key yang merujuk ke tabel `users`
- `created_at`: Waktu pembuatan item

- `updated_at`: Waktu terakhir update item
-

4. Fitur Aplikasi

4.1 CRUD untuk Data

Aplikasi menyediakan operasi CRUD lengkap untuk data item:

- **Create**: Menambahkan item baru ke dalam database
- **Read**: Menampilkan daftar item dan detail item
- **Update**: Mengubah informasi item yang sudah ada
- **Delete**: Menghapus item dari database

4.2 Autentikasi Pengguna

Aplikasi dilengkapi dengan sistem autentikasi dasar:

- **Registrasi**: Pengguna dapat membuat akun baru
- **Login**: Pengguna dapat masuk ke dalam sistem
- **Logout**: Pengguna dapat keluar dari sistem
- **Session Management**: Pengaturan sesi pengguna
- **Pengamanan Akses**: Fitur yang memerlukan login akan dilindungi

4.3 Penerapan OOP

Aplikasi mengimplementasikan konsep Pemrograman Berorientasi Objek:

- **Inheritance**: Model dan Controller mewarisi kelas inti CodeIgniter
- **Encapsulation**: Data dan fungsi dienkapsulasi dalam kelas
- **Abstraction**: Interface yang jelas untuk tiap komponen

4.4 Routing

Aplikasi menggunakan sistem routing untuk URL yang lebih bersih dan terstruktur:

- **Default Controller**: Menentukan halaman yang pertama kali dibuka
- **Custom Routes**: Mendefinisikan pola URL yang mudah diingat
- **Route Parameters**: Mendukung parameter dalam URL

4.5 Templating

Aplikasi menerapkan sistem templating untuk konsistensi tampilan:

- **Header/Footer**: Template untuk bagian atas dan bawah halaman
 - **Sidebar**: Menu navigasi samping untuk pengguna yang sudah login
 - **Content**: Area konten dinamis sesuai halaman yang dibuka
-

5. Penjelasan Kode

5.1 Model

5.1.1 User_model.php Model ini menangani operasi database yang berkaitan dengan pengguna:

```
class User_model extends CI_Model {

    private $table = 'users';

    public function __construct() {
        parent::__construct();

        // Buat tabel users jika belum ada
        $this->create_users_table();
    }

    // Fungsi untuk membuat tabel users
    public function create_users_table() {
        // ... kode untuk membuat tabel ...
    }

    // Fungsi untuk registrasi pengguna baru
    public function register($data) {
        // ... kode untuk registrasi ...
    }

    // Fungsi untuk login
    public function login($username, $password) {
        // ... kode untuk login ...
    }

    // Fungsi lainnya...
}
```

5.1.2 Item_model.php Model ini menangani operasi database yang berkaitan dengan item:

```
class Item_model extends CI_Model {

    private $table = 'items';

    // ... constructor dan fungsi pembuatan tabel ...

    // Mendapatkan semua item
    public function get_all() {
```

```

        // ... kode untuk mendapatkan semua item ...
    }

    // Mendapatkan item berdasarkan user_id
    public function get_by_user($user_id) {
        // ... kode untuk mendapatkan item per pengguna ...
    }

    // Membuat item baru
    public function create($data) {
        // ... kode untuk membuat item ...
    }

    // Mengupdate item
    public function update($id, $data) {
        // ... kode untuk update item ...
    }

    // Menghapus item
    public function delete($id) {
        // ... kode untuk menghapus item ...
    }
}

```

5.2 Controller

5.2.1 Auth.php Controller untuk menangani autentikasi:

```

class Auth extends CI_Controller {

    // ... constructor ...

    // Halaman registrasi
    public function register() {
        // ... kode untuk registrasi ...
    }

    // Halaman login
    public function login() {
        // ... kode untuk login ...
    }

    // Proses logout
    public function logout() {
        // ... kode untuk logout ...
    }
}

```

```
}
```

5.2.2 Dashboard.php Controller untuk menangani halaman dashboard:

```
class Dashboard extends CI_Controller {  
  
    // ... constructor dengan pengecekan login ...  
  
    // Halaman utama dashboard  
    public function index() {  
        // ... kode untuk menampilkan dashboard ...  
    }  
}
```

5.2.3 Items.php Controller untuk manajemen item:

```
class Items extends CI_Controller {  
  
    // ... constructor dengan pengecekan login ...  
  
    // Daftar semua item  
    public function index() {  
        // ... kode untuk menampilkan daftar item ...  
    }  
  
    // Detail item  
    public function view($id) {  
        // ... kode untuk menampilkan detail item ...  
    }  
  
    // Tambah item baru  
    public function create() {  
        // ... kode untuk menambah item ...  
    }  
  
    // Edit item  
    public function edit($id) {  
        // ... kode untuk mengedit item ...  
    }  
  
    // Hapus item  
    public function delete($id) {  
        // ... kode untuk menghapus item ...  
    }  
}
```

5.3 View

View dibagi menjadi beberapa kategori:

- **Templates:** Untuk layout dasar (header, footer, sidebar)
- **Auth:** Untuk halaman autentikasi (login, register)
- **Dashboard:** Untuk halaman dashboard
- **Items:** Untuk manajemen item (index, create, edit, view)

5.4 Routing

Konfigurasi routing didefinisikan di `application/config/routes.php`:

```
$route['default_controller'] = 'auth/login';
$route['register'] = 'auth/register';
$route['login'] = 'auth/login';
$route['logout'] = 'auth/logout';
$route['dashboard'] = 'dashboard';
$route['items'] = 'items/index';
$route['items/create'] = 'items/create';
$route['items/edit/(:num)'] = 'items/edit/$1';
$route['items/delete/(:num)'] = 'items/delete/$1';
$route['items/view/(:num)'] = 'items/view/$1';
```

6. Cara Penggunaan

6.1 Registrasi dan Login

1. Buka halaman utama aplikasi
2. Klik “Register” untuk membuat akun baru
3. Isi formulir registrasi dan klik tombol “Register”
4. Setelah berhasil, login dengan username dan password yang telah dibuat

6.2 Menggunakan Dashboard

Dashboard menampilkan: - Informasi profil pengguna - Statistik jumlah item yang dimiliki - Daftar item terbaru

6.3 Manajemen Item

6.3.1 Melihat Daftar Item

1. Klik menu “Items” di sidebar
2. Halaman akan menampilkan semua item yang dimiliki pengguna

6.3.2 Menambah Item Baru

1. Klik tombol “Add New Item” di halaman Items

2. Isi formulir dengan informasi item
3. Klik tombol “Save Item”

6.3.3 Melihat Detail Item

1. Klik icon mata pada baris item yang ingin dilihat
2. Halaman akan menampilkan detail lengkap item tersebut

6.3.4 Mengubah Item

1. Klik icon pensil pada baris item yang ingin diubah
2. Ubah informasi pada formulir
3. Klik tombol “Update Item”

6.3.5 Menghapus Item

1. Klik icon sampah pada baris item yang ingin dihapus
2. Konfirmasi penghapusan dengan klik “OK”

6.4 Logout

Untuk keluar dari sistem, klik menu “Logout” di sidebar atau navbar

7. Pengembangan Lebih Lanjut

Berikut adalah beberapa saran untuk pengembangan aplikasi lebih lanjut:

7.1 Penambahan Fitur

- **Manajemen Profil:** Memungkinkan pengguna mengedit informasi profil
- **Reset Password:** Fitur untuk mereset password
- **Kategori Item:** Menambahkan kategori untuk pengelompokan item
- **Upload Gambar:** Memungkinkan pengguna mengunggah gambar untuk item
- **Pencarian dan Filter:** Fitur pencarian dan filter untuk data item
- **Pagination:** Pembagian halaman untuk daftar item yang banyak

7.2 Peningkatan Keamanan

- **CSRF Protection:** Mengaktifkan proteksi CSRF di konfigurasi
- **XSS Filtering:** Mengaktifkan filter XSS
- **Rate Limiting:** Pembatasan percobaan login yang gagal
- **Verifikasi Email:** Menambahkan verifikasi email saat registrasi

7.3 Optimasi Performa

- **Caching:** Menerapkan sistem caching untuk query database
 - **Minify Assets:** Kompresi file CSS dan JavaScript
 - **Database Indexing:** Menambahkan indeks pada kolom yang sering dicari
-

Penutup

Aplikasi CodeIgniter 3 ini menunjukkan implementasi dasar dari operasi CRUD, autentikasi pengguna, OOP, routing, dan templating. Diharapkan aplikasi ini dapat menjadi dasar pengembangan aplikasi yang lebih kompleks sesuai kebutuhan.

Untuk pertanyaan atau bantuan lebih lanjut, silakan hubungi pengembang.

Dokumentasi ini dibuat pada Mei 2025