



SIBER MU
UNIVERSITAS SIBER MUHAMMADIYAH

PENGEMBANGAN DAN PENYELENGGARAAN
PEMBELAJARAN DIGITAL (P3D)



Modul Pembelajaran **SISTEM INFORMASI**

Pemrograman Web Framework



Disusun oleh : Wicaksono Yuli Sulistyio



Modul Ajar: Sistem Manajemen Menu dengan Hak Akses menggunakan CodeIgniter 3

1. Pengantar

Dalam aplikasi berbasis web, sistem manajemen menu adalah fitur penting yang memungkinkan pengguna dengan peran (role) berbeda untuk mengakses menu yang sesuai dengan hak akses mereka. Di CodeIgniter 3, manajemen menu dapat dipadukan dengan sistem **role-based access control** (RBAC), di mana setiap peran pengguna memiliki hak akses yang berbeda-beda terhadap menu tertentu.

Pada modul ini, kita akan membahas cara membuat sistem manajemen menu menggunakan **CodeIgniter 3**, mulai dari teori tentang hak akses, implementasi sistem manajemen menu berbasis role, hingga penerapan logika di database dan controller.

2. Teori Hak Akses (Role-Based Access Control - RBAC)

RBAC adalah sebuah metode pengelolaan akses di mana hak akses diberikan kepada pengguna berdasarkan **role** atau perannya. Setiap peran diatur untuk memiliki hak terhadap beberapa resource, seperti menu atau fitur dalam aplikasi. Konsep RBAC biasanya terdiri dari komponen berikut:

1. **User (Pengguna)**: Individu yang menggunakan aplikasi. Setiap pengguna memiliki satu atau lebih peran.
2. **Role (Peran)**: Kategori akses yang mengelompokkan pengguna berdasarkan tanggung jawab atau hak akses mereka. Contohnya, role **admin**, **user**, **guest**, atau **manager**.
3. **Permissions (Hak Akses)**: Hak yang diberikan kepada peran untuk melakukan tindakan tertentu. Sebagai contoh, role **admin** bisa mengakses semua menu, sedangkan **user** hanya bisa mengakses menu tertentu.
4. **Resources (Sumber Daya)**: Hal-hal yang dapat diakses atau diubah oleh pengguna, seperti menu, halaman, atau fitur tertentu dalam aplikasi.

3. Struktur Database untuk Manajemen Menu

Untuk sistem manajemen menu berbasis hak akses, kita perlu mendesain beberapa tabel dalam database untuk menyimpan data menu, role, dan hak akses.

Tabel yang diperlukan:

1. **Tabel user**: Menyimpan informasi tentang pengguna.



sql

```
CREATE TABLE user (  
    id_user INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(100),  
    email VARCHAR(100),  
    password VARCHAR(255),  
    id_role INT  
);
```

2. **Tabel role:** Menyimpan peran (role) yang ada di dalam sistem.

sql

```
CREATE TABLE role (  
    id_role INT PRIMARY KEY AUTO_INCREMENT,  
    role_name VARCHAR(50)  
);
```

3. **Tabel menu:** Menyimpan daftar menu yang ada di aplikasi.

sql

```
CREATE TABLE menu (  
    id_menu INT PRIMARY KEY AUTO_INCREMENT,  
    menu_name VARCHAR(50),  
    url VARCHAR(100),  
    icon VARCHAR(50)  
);
```

4. **Tabel role_access:** Menyimpan informasi hak akses untuk setiap peran terhadap menu.

sql

```
CREATE TABLE role_access (  
    id_role_access INT PRIMARY KEY AUTO_INCREMENT,  
    id_role INT,  
    id_menu INT,  
    can_access BOOLEAN DEFAULT 0  
);
```

Relasi antar Tabel:

- **user -> role:** Setiap pengguna memiliki satu role.
- **role -> menu (melalui role_access):** Setiap role memiliki akses ke beberapa menu.

4. Controller: Menu.php



Controller ini berfungsi untuk mengatur logika di balik sistem manajemen menu. Sebagai contoh, controller `Menu.php` bertanggung jawab untuk:

- Menampilkan daftar menu yang bisa diakses oleh pengguna berdasarkan hak akses mereka.
- Mengelola akses menu untuk setiap peran.

php

```
defined('BASEPATH') or exit('No direct script access allowed');

class Menu extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        // Memastikan user sudah login dan memuat model yang digunakan
        if (!$this->session->userdata('email')) {
            redirect('auth/login');
        }
        $this->load->model('Menu_model');
    }

    // Fungsi untuk menampilkan halaman manajemen menu
    public function index()
    {
        $data['title'] = 'Menu Management';
        $data['user'] = $this->db->get_where('user', ['email' => $this->session->userdata('email')])->row_array();

        $data['menu'] = $this->db->get('menu')->result_array();

        $this->load->view('templates/header', $data);
        $this->load->view('templates/sidebar', $data);
        $this->load->view('templates/topbar', $data);
        $this->load->view('menu/index', $data);
        $this->load->view('templates/footer');
    }

    // Fungsi untuk menambah menu
    public function addMenu()
    {
        $this->form_validation->set_rules('menu', 'Menu', 'required');

        if ($this->form_validation->run() == false) {
            $this->index();
        } else {
            $this->db->insert('menu', ['menu_name' => $this->input->post('menu')]);
            $this->session->set_flashdata('message', '<div class="alert alert-success" role="alert">New menu added!</div>');
            redirect('menu');
        }
    }
}
```

```

// Fungsi untuk mengatur akses menu untuk role tertentu
public function roleAccess($id_role)
{
    $data['title'] = 'Role Access Management';
    $data['user'] = $this->db->get_where('user', ['email' => $this->session->userdata('email')])->row_array();
    $data['role'] = $this->db->get_where('role', ['id_role' => $id_role])->row_array();

    $this->db->where('id_role', $id_role);
    $data['menu'] = $this->Menu_model->getMenuWithAccess($id_role);

    $this->load->view('templates/header', $data);
    $this->load->view('templates/sidebar', $data);
    $this->load->view('templates/topbar', $data);
    $this->load->view('menu/role-access', $data);
    $this->load->view('templates/footer');
}

// Fungsi untuk mengubah hak akses menu
public function changeAccess()
{
    $id_role = $this->input->post('id_role');
    $id_menu = $this->input->post('id_menu');

    $data = [
        'id_role' => $id_role,
        'id_menu' => $id_menu
    ];

    $result = $this->db->get_where('role_access', $data);

    if ($result->num_rows() < 1) {
        $this->db->insert('role_access', $data);
    } else {
        $this->db->delete('role_access', $data);
    }

    $this->session->set_flashdata('message', '<div class="alert alert-success" role="alert">Access Changed!</div>');
}
}

```

Penjelasan:

1. **index()**:
 - Menampilkan daftar menu yang ada dalam sistem.
 - Mengambil data user yang login dan juga menampilkan semua menu dari tabel menu.
2. **addMenu()**:
 - Fungsi untuk menambahkan menu baru ke dalam tabel menu. Terdapat validasi untuk memastikan nama menu harus diisi.

3. `roleAccess($id_role):`

- o Menampilkan halaman untuk mengatur hak akses (role access) dari role tertentu terhadap menu yang ada.
- o Data hak akses diambil dari tabel `role_access` dan tabel `menu`.

4. `changeAccess ():`

- o Fungsi untuk mengubah akses menu dari role tertentu. Jika role sudah memiliki akses ke menu tersebut, akses akan dihapus. Jika belum, akses akan ditambahkan.

5. View: Menampilkan Menu Berdasarkan Hak Akses

Berikut adalah cuplikan kode untuk menampilkan menu di view `sidebar.php` yang hanya akan menampilkan menu yang sesuai dengan hak akses pengguna.

```

php
<?php
$role_id = $this->session->userdata('id_role');
$queryMenu = "SELECT `menu`.`id_menu`, `menu_name`
              FROM `menu` JOIN `role_access`
              ON `menu`.`id_menu` = `role_access`.`id_menu`
              WHERE `role_access`.`id_role` = $role_id
              ORDER BY `menu`.`id_menu` ASC";
$menu = $this->db->query($queryMenu)->result_array();
?>

<!-- Looping Menu -->
<?php foreach ($menu as $m) : ?>
    <div class="sidebar-heading">
        <?=$m['menu_name']; ?>
    </div>
<?php endforeach; ?>

```

Pada bagian ini, query digunakan untuk mengambil menu yang sesuai dengan `id_role` pengguna yang login. Hasil query kemudian ditampilkan di sidebar.

6. Hak Akses Berdasarkan Role

Sebagai contoh, pada aplikasi ini terdapat tiga role:

- **Admin:** Memiliki akses penuh ke semua menu.
- **Siswa:** Hanya memiliki akses ke menu yang berkaitan dengan aktivitas siswa.
- **Guru:** Memiliki akses ke menu yang berkaitan dengan guru.

Dengan menggunakan tabel `role_access`, setiap role bisa diberi atau dihapus akses terhadap menu tertentu.



7. Kesimpulan

Sistem manajemen menu dengan hak akses adalah komponen yang sangat penting dalam pengembangan aplikasi multi-user. Dengan menggunakan CodeIgniter 3 dan penerapan role-based access control, kita dapat mengelola hak akses terhadap menu dengan mudah dan terstruktur. Sistem ini juga dapat dikembangkan lebih lanjut, misalnya dengan menambahkan fitur sub-menu, pengelolaan lebih lanjut untuk sub-role, dan fitur log aktivitas pengguna.

Dengan demikian, modul ini memberikan landasan teoritis dan praktis untuk mengimplementasikan manajemen menu berbasis hak akses di aplikasi berbasis CodeIgniter 3.