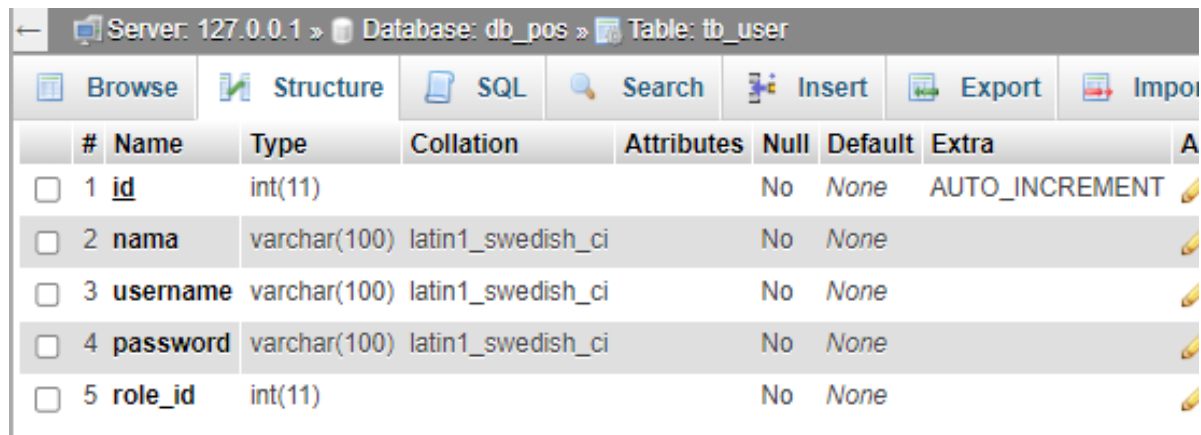


## MEMBUAT FORM LOGIN DAN LOGOUT

### 1. Membuat Database

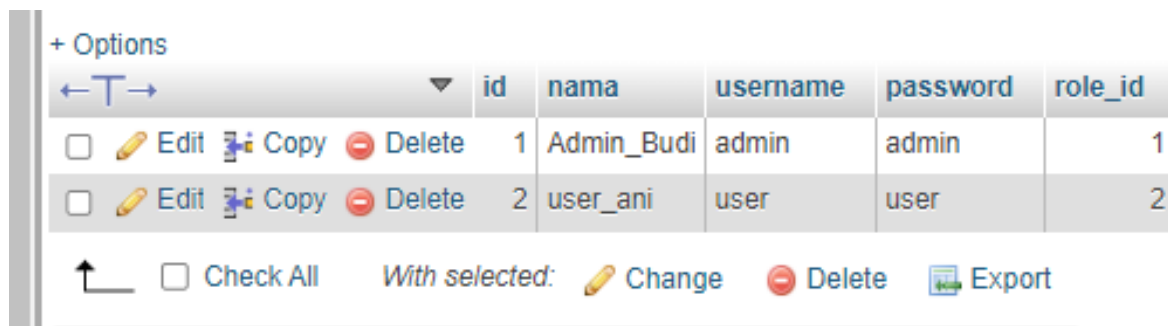
Persiapkan database “db\_pos” anda bisa menggunakan database pada pertemuan sebelumnya kemudian beri nama tabel “tb\_user” dengan struktur data sbb :



The screenshot shows the MySQL Table Structure window for the 'tb\_user' table in the 'db\_pos' database. The table has five columns: 'id' (int(11), AUTO\_INCREMENT), 'nama' (varchar(100)), 'username' (varchar(100)), 'password' (varchar(100)), and 'role\_id' (int(11)). All columns are set to 'No' for Null and 'None' for Default.

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	id	int(11)			No	None	AUTO_INCREMENT
2	nama	varchar(100)	latin1_swedish_ci		No	None	
3	username	varchar(100)	latin1_swedish_ci		No	None	
4	password	varchar(100)	latin1_swedish_ci		No	None	
5	role_id	int(11)			No	None	

Insert data record pada tabel “tb\_user” sebagai berikut :



The screenshot shows the MySQL Data Table window for the 'tb\_user' table. It displays two records. The first record has 'id' 1, 'nama' 'Admin\_Budi', 'username' 'admin', 'password' 'admin', and 'role\_id' 1. The second record has 'id' 2, 'nama' 'user\_ani', 'username' 'user', 'password' 'user', and 'role\_id' 2.

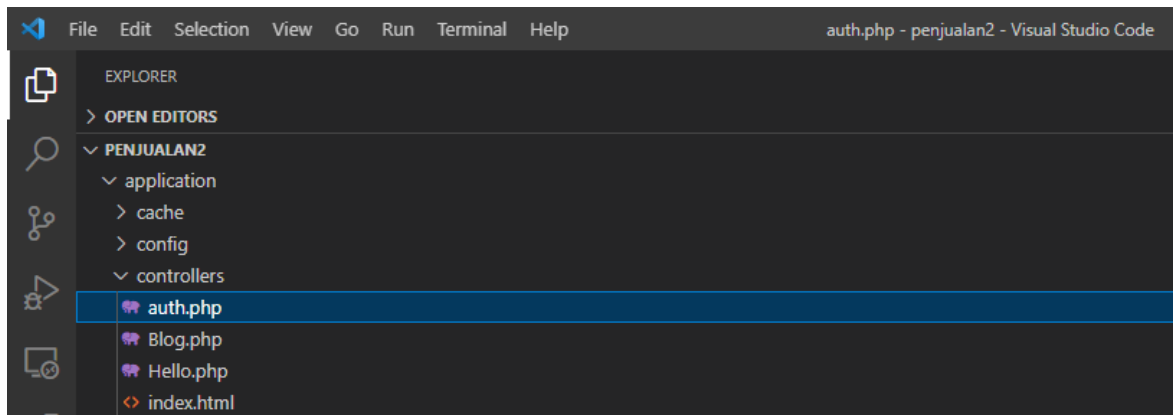
	id	nama	username	password	role_id
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	Admin_Budi	admin	admin	1
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	user_ani	user	user	2

Jika dilihat pada gambar diatas role\_id 1 digunakan untuk level admin dan role\_id 2 digunakan untuk level user

### 2. Membuat Form Login

Langkah-langkah membuat form login sebagai berikut :

- Buatlah sebuah file Controller didalam application/controller dengan nama “auth.php” seperti gambar berikut :



Kemudian buat function login() dengan ketikan kode berikut :

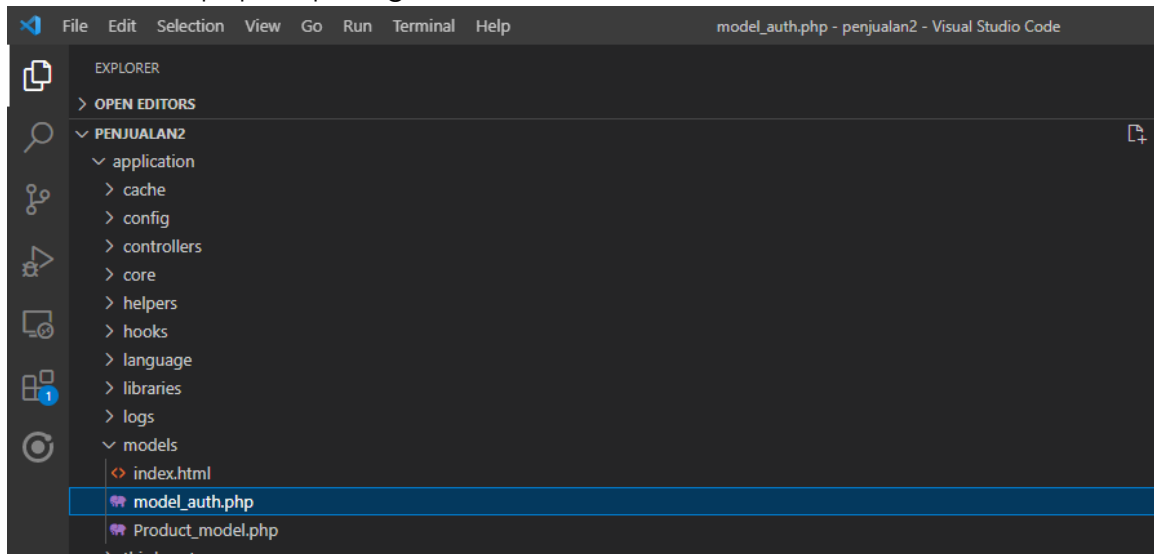
```
<?php
class Auth extends CI_Controller{
public function login()
{
    $this->form_validation->set_rules('username','Username','required',[
        'required' => 'Username wajib diisi'
    ]);
    $this->form_validation->set_rules('password','Password','required',[
        'required' => 'passsword wajib diisi'
    ]);
    if ($this->form_validation->run()== FALSE)
    {
        $this->load->view('form_login');
    }else{
        $auth=$this->model_auth->cek_login();
        if ($auth == FALSE)
        {
            $this->session->set_flashdata('pesan','<div class="alert alert-warning
            alert-dismissible fade show" role="alert">
                Username atau Password Anda Salah
            <button type="button" class="close" data-dismiss="alert" aria-
            label="Close">
                <span aria-hidden="true">&times;</span>
            </button>
            </div>');
            redirect('auth/login');
        }else{
            $this->session->set_userdata('username', $auth->username);
            $this->session->set_userdata('nama', $auth->nama);
            $this->session->set_userdata('role_id', $auth->role_id);
            switch($auth->role_id){
                case 1 : redirect('crud');
```

```

        break;
        case 2 : redirect('crud/tambah');
        default: break;
    }
}
}
}
?>

```

- b. Buatlah sebuah file Model didalam application/model dengan nama "model\_auth.php" seperti gambar berikut :



Kemudian buatlah function cek\_login() atau ketikkan kode sebagai berikut :

```

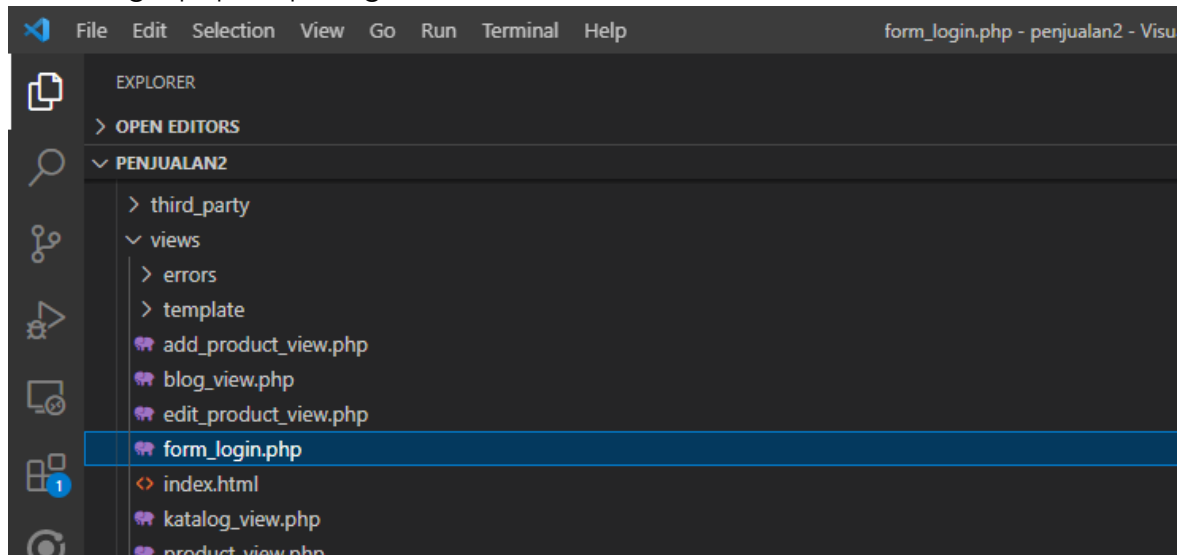
<?php
class Model_auth extends CI_Model{
public Function cek_login()
{
    $username = set_value('username');
    $password = set_value('password');

    $result = $this->db->where('username',$username)
        ->where('password',$password)
        ->limit(1)
        ->get('tb_user');
    if($result->num_rows() >0){
        return $result-> row();
    }else{
        return array();
    }
}
}

```

```
}  
}  
?>
```

- c. Buatlah sebuah file View didalam application/view dengan nama "form\_login.php" seperti gambar berikut :



```
<!DOCTYPE html>  
<html lang="en">  
  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Login Form</title>  
  <style>  
    body {  
      font-family: Arial, sans-serif;  
      background-color: #f4f4f4;  
      margin: 0;  
      display: flex;  
      justify-content: center;  
      align-items: center;  
      height: 100vh;  
    }  
  
    .login-container {  
      background-color: #fff;  
      border-radius: 8px;  
      padding: 20px;  
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
    }  
  </style>  
</head>  
<body>  
  <div class="login-container">  
    <h2>Login</h2>  
    <form>  
      <input type="text" value="Username" />  
      <input type="password" value="Password" />  
      <input type="submit" value="Login" />  
    </form>  
  </div>  
</body>  
</html>
```

```

        width: 300px;
        text-align: center;
    }

    .login-container h2 {
        margin-bottom: 20px;
    }

    .form-group {
        margin-bottom: 20px;
    }

    .form-group label {
        display: block;
        margin-bottom: 8px;
    }

    .form-group input {
        width: 100%;
        padding: 8px;
        box-sizing: border-box;
    }

    .form-group button {
        background-color: #4caf50;
        color: #fff;
        padding: 10px;
        border: none;
        border-radius: 4px;
        cursor: pointer;
    }
</style>
</head>

<body>

    <div class="login-container">
        <h2>Login</h2>
        <form method="post" action="<?php echo base_url('auth/login') ?>">
            <div class="form-group">
                <label for="username">Username:</label>
                <input type="text" id="username" name="username" required>
            </div>

            <div class="form-group">

```

```

        <label for="password">Password:</label>
        <input type="password" id="password" name="password" required>
    </div>

    <div class="form-group">
        <button type="submit">Login</button>
    </div>
</form>
</div>

</body>

</html>

```

- d. Konfigurasi autoload libraries dan autoload model pada file "autoload.php" yang terdapat pada application/config konfigurasi nya sebagai berikut :

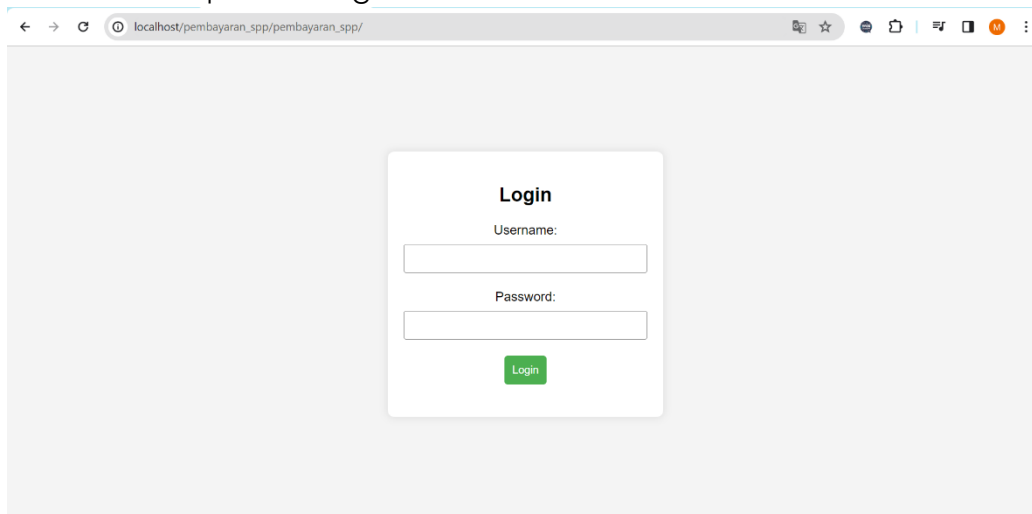
#### Konfigurasi autoload libraries

```
$autoload['libraries'] = array('database', 'form_validation', 'session', 'upload');
```

#### Konfigurasi autoload model

```
$autoload['model'] = array('model_auth');
```

- e. Sekarang jalankan project anda dengan memasukkan link [http://localhost/pembayaran\\_spp/auth/login](http://localhost/pembayaran_spp/auth/login) pada browser anda, jika sukses maka tampilan sebagai berikut :



- f. Silahkan Login dengan username dan password yang sudah anda daftarkan pada database.

### 3. Membuat Logout

Tambahkan function logout pada file auth.php kode sebagai berikut :

```
public function logout(){
    $this->session->sess_destroy();
    redirect('auth/login');
}
```

### 4. Membatasi hak Akses

Untuk menghindari akses pengguna yang belum login aplikasi maka dibutuhkan security pada website yang kita bangun, pada kesempatan ini kita akan memberikan keamanan sederhana pada website yang kita sedang bangun dengan cara memanfaatkan session\_data cara mengupdate file controller crud.php yang terdapat pada application/controller. Perubahan bisa kita lakukan pada function \_\_construct() sebagai berikut :

```
function __construct() {
    parent::__construct();
    $this->load->helper('url', 'form');
    $this->load->model('m_data');

    if($this->session->userdata('role_id')!='1'){
        $this->session->set_flashdata('pesan','<div class="alert alert-warning alert-dismissible fade show" role="alert">Anda Belum login<button type="button" class="close" data-dismiss="alert" aria-label="Close">
            <span aria-hidden="true">&times;</span>
        </button>
        </div>');
        redirect('auth/login');
    }
}
```

Dengan cara ini jika terdapat pengguna mengakses link tanpa login maka akan dikembalikan ke form\_login dan akan tampil pesan Anda belum login.