**LAPORAN PRAKTIKUM**

**PEMROGRAMAN WEB II**

**MODUL 7**

****

**CRUD DAN LOGIN PADA CODEIGNITER**

**Oleh:**

**Naufal Elyzar NIM. 2310817210019**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS LAMBUNG MANGKURAT**

**Mei 2025**

# LEMBAR PENGESAHAN

**LAPORAN PRAKTIKUM PEMROGRAMAN WEB II**

**MODUL 7**

Laporan Praktikum Pemrograman Web II Modul 7: CRUD dan Login pada Codeigniter ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman Web II. Laporan Prakitkum ini dikerjakan oleh:

Nama Praktikan : Naufal Elyzar

NIM : 2310817210019

|  |  |
| --- | --- |
| Menyetujui,  Asisten Praktikum  Rohid Akmal  NIM. 2210817310017 | Mengetahui,  Dosen Penanggung Jawab Praktikum  Andreyan Rizky Baskara, S.Kom., M.Kom.  NIP. 19930703 201903 1 011 |

# DAFTAR ISI

[LEMBAR PENGESAHAN 2](#_Toc199078680)

[DAFTAR ISI 3](#_Toc199078681)

[DAFTAR GAMBAR 4](#_Toc199078682)

[DAFTAR TABEL 5](#_Toc199078683)

[SOAL 1 6](#_Toc199078684)

[A. Source Code 6](#_Toc199078685)

[B. Output Program 36](#_Toc199078686)

[C. Pembahasan 37](#_Toc199078687)

[D. Tautan Git 39](#_Toc199078688)

# DAFTAR GAMBAR

[Gambar 1. Screenshot Hasil Jawaban Soal 1 Halaman Register 36](#_Toc199078698)

[Gambar 2 Screenshot Hasil Jawaban Soal 1 Halaman Login 36](#_Toc199078699)

[Gambar 3 Screenshot Hasil Jawaban Soal 1 Halaman Utama 36](#_Toc199078700)

[Gambar 4 Screenshot Hasil Jawaban Soal 1 Halaman create 37](#_Toc199078701)

[Gambar 5 Screenshot Hasil Jawaban Soal 1 Jika memaksa ke halaman lain sebelum login 37](#_Toc199078702)

# DAFTAR TABEL

# SOAL 1

Buatlah sebuah website yang memiliki fitur login dan logout. Setelah login pengguna website dapat melakukan CRUD yang terhubung langsung dengan database. Beberapa syarat yang harus ada pada website:

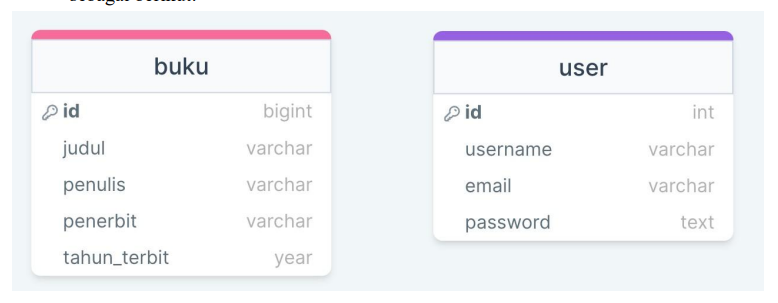
● Jika seorang pengguna belum login dan mengakses halaman yang mengharuskan login, maka tampilkan sebuah pesan peringatan di halaman login, peringatan berisi “Login terlebih dahulu!”.

● Berikan validasi pada bagian ‘Create’ atau ‘tambah data’ di form, sebagai berikut:

* Judul : Harus diisi dan berupa string
* penulis : Harus diisi dan berupa string
* penerbit : Harus diisi dan berupa string
* tahun\_terbit : Harus diisi dan berupa angka, angka harus lebih besar dari 1800 dan lebih kecil dari 2024

● (opsional) buatlah custom pesan validasi menggunakan bahasa indonesia

● Buat file migration pada codeigniter dengan Desain konseptual database



Simpan dalam folder github PRAK 701

## Source Code

2024-03-19-create\_users\_table.php(Database)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49 | <?php  namespace App\Database\Migrations;  use CodeIgniter\Database\Migration;  class CreateUsersTable extends Migration  {  public function up()  {  $this->forge->addField([  'id' => [  'type' => 'BIGINT',  'unsigned' => true,  'auto\_increment' => true,  ],  'username' => [  'type' => 'VARCHAR',  'constraint' => '100',  'unique' => true,  ],  'email' => [  'type' => 'VARCHAR',  'constraint' => '255',  'unique' => true,  ],  'password' => [  'type' => 'VARCHAR',  'constraint' => '255',  ],  'created\_at' => [  'type' => 'DATETIME',  'null' => true,  ],  'updated\_at' => [  'type' => 'DATETIME',  'null' => true,  ],  ]);    $this->forge->addKey('id', true);  $this->forge->createTable('users');  }  public function down()  {  $this->forge->dropTable('users');  }  } |

2024-03-19-create\_books\_table.php (Database)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50 | <?php  namespace App\Database\Migrations;  use CodeIgniter\Database\Migration;  class CreateBooksTable extends Migration  {  public function up()  {  $this->forge->addField([  'id' => [  'type' => 'BIGINT',  'unsigned' => true,  'auto\_increment' => true,  ],  'judul' => [  'type' => 'VARCHAR',  'constraint' => '255',  ],  'penulis' => [  'type' => 'VARCHAR',  'constraint' => '255',  ],  'penerbit' => [  'type' => 'VARCHAR',  'constraint' => '255',  ],  'tahun\_terbit' => [  'type' => 'YEAR',  ],  'created\_at' => [  'type' => 'DATETIME',  'null' => true,  ],  'updated\_at' => [  'type' => 'DATETIME',  'null' => true,  ],  ]);    $this->forge->addKey('id', true);  $this->forge->createTable('books');  }  public function down()  {  $this->forge->dropTable('books');  }  } |

UserModel.php (Model)

|  |  |
| --- | --- |
| 1 | <?php |
| 2 | namespace App\Models; |
| 3 | use CodeIgniter\Model; |
| 4 |  |
| 5 | class UserModel extends Model |
| 6 | { |
| 7 | protected $table = 'users'; |
| 8 | protected $primaryKey = 'id'; |
| 9 | protected $useAutoIncrement = true; |
| 10 | protected $returnType = 'array'; |
| 11 | protected $useSoftDeletes = false; |
| 12 | protected $allowedFields = ['username', 'email', 'password']; |
| 13 | protected $useTimestamps = true; |
| 14 | protected $createdField = 'created\_at'; |
| 15 | protected $updatedField = 'updated\_at'; |
| 16 |  |
| 17 | protected $validationRules = [ |
| 18 | 'username' => 'required |
| 19 | 'email' => 'required |
| 20 | 'password' => 'required |
| 21 | ]; |
| 22 |  |
| 23 | protected $validationMessages = [ |
| 24 | 'username' => [ |
| 25 | 'required' => 'Username harus diisi', |
| 26 | 'min\_length' => 'Username minimal 4 karakter', |
| 27 | 'is\_unique' => 'Username sudah digunakan' |
| 28 | ], |
| 29 | 'email' => [ |
| 30 | 'required' => 'Email harus diisi', |
| 31 | 'valid\_email' => 'Email tidak valid', |
| 32 | 'is\_unique' => 'Email sudah digunakan' |
| 33 | ], |
| 34 | 'password' => [ |
| 35 | 'required' => 'Password harus diisi', |
| 36 | 'min\_length' => 'Password minimal 6 karakter' |
| 37 | ] |
| 38 | ]; |
| 39 |  |
| 40 | protected $skipValidation = false; |
| 41 | protected $cleanValidationRules = true; |
| 42 | } |

BookModel.php (Model)

|  |  |
| --- | --- |
| 1 | <?php |
| 2 | namespace App\Models; |
| 3 | use CodeIgniter\Model; |
| 4 |  |
| 5 | class BookModel extends Model |
| 6 | { |
| 7 | protected $table = 'books'; |
| 8 | protected $primaryKey = 'id'; |
| 9 | protected $useAutoIncrement = true; |
| 10 | protected $returnType = 'array'; |
| 11 | protected $useSoftDeletes = false; |
| 12 | protected $allowedFields = ['judul', 'penulis', 'penerbit', 'tahun\_terbit']; |
| 13 | protected $useTimestamps = true; |
| 14 | protected $createdField = 'created\_at'; |
| 15 | protected $updatedField = 'updated\_at'; |
| 16 |  |
| 17 |  |
| 18 | protected $validationRules = [ |
| 19 | 'judul' => [ |
| 20 | ` 'rules' => 'required |
| 21 | 'errors' => [ |
| 22 | 'required' => 'Judul harus diisi', |
| 23 | 'string' => 'Judul harus berupa teks' |
| 24 | ] |
| 25 | ], |
| 26 | 'penulis' => [ |
| 27 | ` 'rules' => 'required |
| 28 | 'errors' => [ |
| 29 | 'required' => 'Penulis harus diisi', |
| 30 | 'string' => 'Penulis harus berupa teks' |
| 31 | ] |
| 32 | ], |
| 33 | 'penerbit' => [ |
| 34 | ` 'rules' => 'required |
| 35 | 'errors' => [ |
| 36 | 'required' => 'Penerbit harus diisi', |
| 37 | 'string' => 'Penerbit harus berupa teks' |
| 38 | ] |
| 39 | ], |
| 40 | 'tahun\_terbit' => [ |
| 41 | ` 'rules' => 'required |
| 42 | 'errors' => [ |
| 43 | 'required' => 'Tahun terbit harus diisi', |
| 44 | 'numeric' => 'Tahun terbit harus berupa angka', |
| 45 | 'greater\_than' => 'Tahun terbit harus lebih besar dari 1800', |
| 46 | 'less\_than' => 'Tahun terbit harus lebih kecil dari 2024' |
| 47 | ] |
| 48 | ] |
| 49 | ]; |
| 50 | } |

Auth.php(Controller)

|  |  |
| --- | --- |
| 1 | <?php |
| 2 | namespace App\Controllers; |
| 3 | use App\Models\UserModel; |
| 4 |  |
| 5 | class Auth extends BaseController |
| 6 | { |
| 7 | protected $userModel; |
| 8 |  |
| 9 | public function \_\_construct() |
| 10 | { |
| 11 | $this->userModel = new UserModel(); |
| 12 | } |
| 13 |  |
| 14 | public function login() |
| 15 | { |
| 16 | if (session()->get('isLoggedIn')) { |
| 17 | return redirect()->to('/books'); |
| 18 | } |
| 19 | return view('auth/login'); |
| 20 | } |
| 21 |  |
| 22 | public function attemptLogin() |
| 23 | { |
| 24 | $data = $this->request->getPost(['username', 'password']); |
| 25 |  |
| 26 | if (!$this->validateData($data, [ |
| 27 | 'username' => 'required', |
| 28 | 'password' => 'required' |
| 29 | ])) { |
| 30 | return redirect()->back()->withInput(); |
| 31 | } |
| 32 |  |
| 33 | $username = $this->request->getPost('username'); |
| 34 | $password = $this->request->getPost('password'); |
| 35 |  |
| 36 | $user = $this->userModel->where('username', $username)->first(); |
| 37 |  |
| 38 | if (!$user) { |
| 39 | session()->setFlashdata('error', 'Username atau password salah'); |
| 40 | return redirect()->back()->withInput(); |
| 41 | } |
| 42 |  |
| 43 | $passwordCheck = password\_verify($password, $user['password']); |
| 44 |  |
| 45 | if (!$passwordCheck) { |
| 46 | session()->setFlashdata('error', 'Username atau password salah'); |
| 47 | return redirect()->back()->withInput(); |
| 48 | } |
| 49 |  |
| 50 | $userData = [ |
| 51 | 'id' => $user['id'], |
| 52 | 'username' => $user['username'], |
| 53 | 'email' => $user['email'], |
| 54 | 'isLoggedIn' => true |
| 55 | ]; |
| 56 |  |
| 57 | session()->set($userData); |
| 58 | return redirect()->to('/books'); |
| 59 | } |
| 60 |  |
| 61 | public function register() |
| 62 | { |
| 63 | if (session()->get('isLoggedIn')) { |
| 64 | return redirect()->to('/books'); |
| 65 | } |
| 66 | return view('auth/register'); |
| 67 | } |
| 68 |  |
| 69 | public function attemptRegister() |
| 70 | { |
| 71 | $rules = [ |
| 72 | ` 'username' => 'required |
| 73 | ` 'email' => 'required |
| 74 | ` 'password' => 'required |
| 75 | ` 'confirm\_password' => 'required |
| 76 | ]; |
| 77 |  |
| 78 | $errors = [ |
| 79 | 'username' => [ |
| 80 | 'required' => 'Username harus diisi', |
| 81 | 'min\_length' => 'Username minimal 4 karakter', |
| 82 | 'is\_unique' => 'Username sudah digunakan' |
| 83 | ], |
| 84 | 'email' => [ |
| 85 | 'required' => 'Email harus diisi', |
| 86 | 'valid\_email' => 'Email tidak valid', |
| 87 | 'is\_unique' => 'Email sudah digunakan' |
| 88 | ], |
| 89 | 'password' => [ |
| 90 | 'required' => 'Password harus diisi', |
| 91 | 'min\_length' => 'Password minimal 6 karakter' |
| 92 | ], |
| 93 | 'confirm\_password' => [ |
| 94 | 'required' => 'Konfirmasi password harus diisi', |
| 95 | 'matches' => 'Konfirmasi password tidak cocok' |
| 96 | ] |
| 97 | ]; |
| 98 |  |
| 99 | if (!$this->validate($rules, $errors)) { |
| 100 | return redirect()->back()->withInput()->with('validation', $this->validator); |
| 101 | } |
| 102 |  |
| 103 | $hashedPassword = password\_hash($this->request->getPost('password'), PASSWORD\_DEFAULT); |
| 104 |  |
| 105 | $this->userModel->save([ |
| 106 | 'username' => $this->request->getPost('username'), |
| 107 | 'email' => $this->request->getPost('email'), |
| 108 | 'password' => $hashedPassword |
| 109 | ]); |
| 110 |  |
| 111 | session()->setFlashdata('success', 'Registrasi berhasil, silakan login'); |
| 112 | return redirect()->to('/login'); |
| 113 | } |
| 114 |  |
| 115 | public function logout() |
| 116 | { |
| 117 | session()->destroy(); |
| 118 | return redirect()->to('/login'); |
| 119 | } |
| 120 | } |

Home.php(Controller)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | <?php  namespace App\Controllers;  class Home extends BaseController  {  public function index()  {  if (session()->get('isLoggedIn')) {  return redirect()->to('/books');  }  return redirect()->to('/login');  }  } |

Books.php(Controller)

|  |  |
| --- | --- |
| 1 | <?php |
| 2 | namespace App\Controllers; |
| 3 |  |
| 4 | use App\Models\BookModel; |
| 5 | use CodeIgniter\Controller; |
| 6 |  |
| 7 | class Books extends Controller |
| 8 | { |
| 9 | protected $bookModel; |
| 10 |  |
| 11 | public function \_\_construct() |
| 12 | { |
| 13 | $this->bookModel = new BookModel(); |
| 14 | } |
| 15 |  |
| 16 | public function index() |
| 17 | { |
| 18 | $data = [ |
| 19 | 'title' => 'Daftar Buku', |
| 20 | 'books' => $this->bookModel->findAll() |
| 21 | ]; |
| 22 |  |
| 23 | return view('books/index', $data); |
| 24 | } |
| 25 |  |
| 26 | public function create() |
| 27 | { |
| 28 | $data = [ |
| 29 | 'title' => 'Tambah Buku', |
| 30 | 'validation' => \Config\Services::validation() |
| 31 | ]; |
| 32 |  |
| 33 | return view('books/create', $data); |
| 34 | } |
| 35 |  |
| 36 | public function store() |
| 37 | { |
| 38 | if (!$this->validate($this->bookModel->validationRules)) { |
| 39 | return redirect()->back()->withInput()->with('validation', $this->validator); |
| 40 | } |
| 41 |  |
| 42 | $this->bookModel->save([ |
| 43 | 'judul' => $this->request->getPost('judul'), |
| 44 | 'penulis' => $this->request->getPost('penulis'), |
| 45 | 'penerbit' => $this->request->getPost('penerbit'), |
| 46 | 'tahun\_terbit' => $this->request->getPost('tahun\_terbit') |
| 47 | ]); |
| 48 |  |
| 49 | session()->setFlashdata('success', 'Buku berhasil ditambahkan'); |
| 50 | return redirect()->to('/books'); |
| 51 | } |
| 52 |  |
| 53 | public function edit($id) |
| 54 | { |
| 55 | $data = [ |
| 56 | 'title' => 'Edit Buku', |
| 57 | 'validation' => \Config\Services::validation(), |
| 58 | 'book' => $this->bookModel->find($id) |
| 59 | ]; |
| 60 |  |
| 61 | if (empty($data['book'])) { |
| 62 | throw new \CodeIgniter\Exceptions\PageNotFoundException('Buku tidak ditemukan'); |
| 63 | } |
| 64 |  |
| 65 | return view('books/edit', $data); |
| 66 | } |
| 67 |  |
| 68 | public function update($id) |
| 69 | { |
| 70 | if (!$this->validate($this->bookModel->validationRules)) { |
| 71 | return redirect()->back()->withInput()->with('validation', $this->validator); |
| 72 | } |
| 73 |  |
| 74 | $this->bookModel->update($id, [ |
| 75 | 'judul' => $this->request->getPost('judul'), |
| 76 | 'penulis' => $this->request->getPost('penulis'), |
| 77 | 'penerbit' => $this->request->getPost('penerbit'), |
| 78 | 'tahun\_terbit' => $this->request->getPost('tahun\_terbit') |
| 79 | ]); |
| 80 |  |
| 81 | session()->setFlashdata('success', 'Buku berhasil diperbarui'); |
| 82 | return redirect()->to('/books'); |
| 83 | } |
| 84 |  |
| 85 | public function delete($id) |
| 86 | { |
| 87 | $this->bookModel->delete($id); |
| 88 | session()->setFlashdata('success', 'Buku berhasil dihapus'); |
| 89 | return redirect()->to('/books'); |
| 90 | } |
| 91 | } |

Create.php(Views)

|  |  |
| --- | --- |
| 1 | <?= $this->extend('layout/template') ?> |
| 2 |  |
| 3 | <?= $this->section('content') ?> |
| 4 | <div class="container mt-5"> |
| 5 | <div class="row"> |
| 6 | <div class="col-12 col-md-8 offset-md-2"> |
| 7 | <div class="card"> |
| 8 | <div class="card-header"> |
| 9 | <h5 class="mb-0"><?= $title ?></h5> |
| 10 | </div> |
| 11 | <div class="card-body"> |
| 12 | <form action="<?= base\_url('books/store') ?>" method="post"> |
| 13 | <?= csrf\_field() ?> |
| 14 |  |
| 15 | <div class="mb-3"> |
| 16 | <label for="judul" class="form-label">Judul</label> |
| 17 | <input type="text" class="form-control <?= (session('validation') && session('validation')->hasError('judul')) ? 'is-invalid' : '' ?>" |
| 18 | id="judul" name="judul" value="<?= old('judul') ?>"> |
| 19 | <?php if (session('validation') && session('validation')->hasError('judul')) : ?> |
| 20 | <div class="invalid-feedback"> |
| 21 | <?= session('validation')->getError('judul') ?> |
| 22 | </div> |
| 23 | <?php endif; ?> |
| 24 | </div> |
| 25 |  |
| 26 | <div class="mb-3"> |
| 27 | <label for="penulis" class="form-label">Penulis</label> |
| 28 | <input type="text" class="form-control <?= (session('validation') && session('validation')->hasError('penulis')) ? 'is-invalid' : '' ?> " |
| 29 | id="penulis" name="penulis" value="<?= old('penulis') ?>"> |
| 30 | <?php if (session('validation') && session('validation')->hasError('penulis')) : ?> |
| 31 | <div class="invalid-feedback"> |
| 32 | <?= session('validation')->getError('penulis') ?> |
| 33 | </div> |
| 34 | <?php endif; ?> |
| 35 | </div> |
| 36 |  |
| 37 | <div class="mb-3"> |
| 38 | <label for="penerbit" class="form-label">Penerbit</label> |
| 39 | <input type="text" class="form-control <?= (session('validation') && session('validation')->hasError('penerbit')) ? 'is-invalid' : '' ?> " |
| 40 | id="penerbit" name="penerbit" value="<?= old('penerbit') ?>"> |
| 41 | <?php if (session('validation') && session('validation')->hasError('penerbit')) : ?> |
| 42 | <div class="invalid-feedback"> |
| 43 | <?= session('validation')->getError('penerbit') ?> |
| 44 | </div> |
| 45 | <?php endif; ?> |
| 46 | </div> |
| 47 |  |
| 48 | <div class="mb-3"> |
| 49 | <label for="tahun\_terbit" class="form-label">Tahun Terbit</label> |
| 50 | <input type="number" class="form-control <?= (session('validation') && session('validation')->hasError('tahun\_terbit')) ? 'is-invalid' : '' ?> " |
| 51 | id="tahun\_terbit" name="tahun\_terbit" value="<?= old('tahun\_terbit') ?>" min="1800" max="2023"> |
| 52 | <?php if (session('validation') && session('validation')->hasError('tahun\_terbit')) : ?> |
| 53 | <div class="invalid-feedback"> |
| 54 | <?= session('validation')->getError('tahun\_terbit') ?> |
| 55 | </div> |
| 56 | <?php endif; ?> |
| 57 | </div> |
| 58 |  |
| 59 | <div class="d-flex justify-content-between"> |
| 60 | <a href="<?= base\_url('books') ?>" class="btn btn-secondary">Kembali</a> |
| 61 | <button type="submit" class="btn btn-primary">Simpan</button> |
| 62 | </div> |
| 63 | </form> |
| 64 | </div> |
| 65 | </div> |
| 66 | </div> |
| 67 | </div> |
| 68 | </div> |
| 69 | <?= $this->endSection() ?> |

edit.php(Views)

|  |  |
| --- | --- |
| 1 | <?= $this->extend('layout/template') ?> |
| 2 |  |
| 3 | <?= $this->section('content') ?> |
| 4 | <div class="container mt-5"> |
| 5 | <div class="row"> |
| 6 | <div class="col-12 col-md-8 offset-md-2"> |
| 7 | <div class="card"> |
| 8 | <div class="card-header"> |
| 9 | <h5 class="mb-0"><?= $title ?></h5> |
| 10 | </div> |
| 11 | <div class="card-body"> |
| 12 | <form action="<?= base\_url('books/update/' . $book['id']) ?>" method="post"> |
| 13 | <?= csrf\_field() ?> |
| 14 |  |
| 15 | <div class="mb-3"> |
| 16 | <label for="judul" class="form-label">Judul</label> |
| 17 | <input type="text" class="form-control <?= (session('validation') && session('validation')->hasError('judul')) ? 'is-invalid' : '' ?> " |
| 18 | id="judul" name="judul" value="<?= old('judul', $book['judul']) ?>"> |
| 19 | <?php if (session('validation') && session('validation')->hasError('judul')) : ?> |
| 20 | <div class="invalid-feedback"> |
| 21 | <?= session('validation')->getError('judul') ?> |
| 22 | </div> |
| 23 | <?php endif; ?> |
| 24 | </div> |
| 25 |  |
| 26 | <div class="mb-3"> |
| 27 | <label for="penulis" class="form-label">Penulis</label> |
| 28 | <input type="text" class="form-control <?= (session('validation') && session('validation')->hasError('penulis')) ? 'is-invalid' : '' ?> " |
| 29 | id="penulis" name="penulis" value="<?= old('penulis', $book['penulis']) ?>"> |
| 30 | <?php if (session('validation') && session('validation')->hasError('penulis')) : ?> |
| 31 | <div class="invalid-feedback"> |
| 32 | <?= session('validation')->getError('penulis') ?> |
| 33 | </div> |
| 34 | <?php endif; ?> |
| 35 | </div> |
| 36 |  |
| 37 | <div class="mb-3"> |
| 38 | <label for="penerbit" class="form-label">Penerbit</label> |
| 39 | <input type="text" class="form-control <?= (session('validation') && session('validation')->hasError('penerbit')) ? 'is-invalid' : '' ?> " |
| 40 | id="penerbit" name="penerbit" value="<?= old('penerbit', $book['penerbit']) ?>"> |
| 41 | <?php if (session('validation') && session('validation')->hasError('penerbit')) : ?> |
| 42 | <div class="invalid-feedback"> |
| 43 | <?= session('validation')->getError('penerbit') ?> |
| 44 | </div> |
| 45 | <?php endif; ?> |
| 46 | </div> |
| 47 |  |
| 48 | <div class="mb-3"> |
| 49 | <label for="tahun\_terbit" class="form-label">Tahun Terbit</label> |
| 50 | <input type="number" class="form-control <?= (session('validation') && session('validation')->hasError('tahun\_terbit')) ? 'is-invalid' : '' ?> " |
| 51 | id="tahun\_terbit" name="tahun\_terbit" value="<?= old('tahun\_terbit', $book['tahun\_terbit']) ?>" min="1800" max="2023"> |
| 52 | <?php if (session('validation') && session('validation')->hasError('tahun\_terbit')) : ?> |
| 53 | <div class="invalid-feedback"> |
| 54 | <?= session('validation')->getError('tahun\_terbit') ?> |
| 55 | </div> |
| 56 | <?php endif; ?> |
| 57 | </div> |
| 58 |  |
| 59 | <div class="d-flex justify-content-between"> |
| 60 | <a href="<?= base\_url('books') ?>" class="btn btn-secondary">Kembali</a> |
| 61 | <button type="submit" class="btn btn-primary">Update</button> |
| 62 | </div> |
| 63 | </form> |
| 64 | </div> |
| 65 | </div> |
| 66 | </div> |
| 67 | </div> |
| 68 | </div> |
| 69 | <?= $this->endSection() ?> |

Index.php(views)

|  |  |
| --- | --- |
| 1 | <?= $this->extend('layout/template') ?> |
| 2 |  |
| 3 | <?= $this->section('content') ?> |
| 4 | <div class="container mt-5"> |
| 5 | <div class="row"> |
| 6 | <div class="col-12"> |
| 7 | <div class="card"> |
| 8 | <div class="card-header d-flex justify-content-between align-items-center"> |
| 9 | <h5 class="mb-0"><?= $title ?></h5> |
| 10 | <a href="<?= base\_url('books/create') ?>" class="btn btn-primary">Tambah Buku</a> |
| 11 | </div> |
| 12 | <div class="card-body"> |
| 13 | <?php if (session()->getFlashdata('success')) : ?> |
| 14 | <div class="alert alert-success"> |
| 15 | <?= session()->getFlashdata('success') ?> |
| 16 | </div> |
| 17 | <?php endif; ?> |
| 18 |  |
| 19 | <div class="table-responsive"> |
| 20 | <table class="table table-hover"> |
| 21 | <thead> |
| 22 | <tr> |
| 23 | <th>No</th> |
| 24 | <th>Judul</th> |
| 25 | <th>Penulis</th> |
| 26 | <th>Penerbit</th> |
| 27 | <th>Tahun Terbit</th> |
| 28 | <th>Aksi</th> |
| 29 | </tr> |
| 30 | </thead> |
| 31 | <tbody> |
| 32 | <?php $i = 1; ?> |
| 33 | <?php foreach ($books as $book) : ?> |
| 34 | <tr> |
| 35 | <td><?= $i++ ?></td> |
| 36 | <td><?= esc($book['judul']) ?></td> |
| 37 | <td><?= esc($book['penulis']) ?></td> |
| 38 | <td><?= esc($book['penerbit']) ?></td> |
| 39 | <td><?= esc($book['tahun\_terbit']) ?></td> |
| 40 | <td> |
| 41 | <a href="<?= base\_url('books/edit/' . $book['id']) ?>" class="btn btn-sm btn-warning">Edit</a> |
| 42 | <form action="<?= base\_url('books/delete/' . $book['id']) ?>" method="post" class="d-inline" onsubmit="return confirm('Apakah Anda yakin ingin menghapus buku ini?')"> |
| 43 | <?= csrf\_field() ?> |
| 44 | <button type="submit" class="btn btn-sm btn-danger">Hapus</button> |
| 45 | </form> |
| 46 | </td> |
| 47 | </tr> |
| 48 | <?php endforeach; ?> |
| 49 | <?php if (empty($books)) : ?> |
| 50 | <tr> |
| 51 | <td colspan="6" class="text-center">Tidak ada data buku</td> |
| 52 | </tr> |
| 53 | <?php endif; ?> |
| 54 | </tbody> |
| 55 | </table> |
| 56 | </div> |
| 57 | </div> |
| 58 | </div> |
| 59 | </div> |
| 60 | </div> |
| 61 | </div> |
| 62 | <?= $this->endSection() ?> |

Register.php(Views)

|  |  |
| --- | --- |
| 1 | <?= $this->extend('layout/template') ?> |
| 2 |  |
| 3 | <?= $this->section('content') ?> |
| 4 | <div class="container mt-5"> |
| 5 | <div class="row justify-content-center"> |
| 6 | <div class="col-md-6"> |
| 7 | <div class="card"> |
| 8 | <div class="card-header"> |
| 9 | <h4 class="mb-0">Register</h4> |
| 10 | </div> |
| 11 | <div class="card-body"> |
| 12 | <form action="<?= base\_url('register') ?>" method="post"> |
| 13 | <?= csrf\_field() ?> |
| 14 |  |
| 15 | <div class="mb-3"> |
| 16 | <label for="username" class="form-label">Username</label> |
| 17 | <input type="text" class="form-control <?= (session('validation') && session('validation')->hasError('username')) ? 'is-invalid' : '' ?>" |
| 18 | id="username" name="username" value="<?= old('username') ?>"> |
| 19 | <?php if (session('validation') && session('validation')->hasError('username')) : ?> |
| 20 | <div class="invalid-feedback"> |
| 21 | <?= session('validation')->getError('username') ?> |
| 22 | </div> |
| 23 | <?php endif; ?> |
| 24 | </div> |
| 25 |  |
| 26 | <div class="mb-3"> |
| 27 | <label for="email" class="form-label">Email</label> |
| 28 | <input type="email" class="form-control <?= (session('validation') && session('validation')->hasError('email')) ? 'is-invalid' : '' ?>" |
| 29 | id="email" name="email" value="<?= old('email') ?>"> |
| 30 | <?php if (session('validation') && session('validation')->hasError('email')) : ?> |
| 31 | <div class="invalid-feedback"> |
| 32 | <?= session('validation')->getError('email') ?> |
| 33 | </div> |
| 34 | <?php endif; ?> |
| 35 | </div> |
| 36 |  |
| 37 | <div class="mb-3"> |
| 38 | <label for="password" class="form-label">Password</label> |
| 39 | <input type="password" class="form-control <?= (session('validation') && session('validation')->hasError('password')) ? 'is-invalid' : '' ?>" |
| 40 | id="password" name="password"> |
| 41 | <?php if (session('validation') && session('validation')->hasError('password')) : ?> |
| 42 | <div class="invalid-feedback"> |
| 43 | <?= session('validation')->getError('password') ?> |
| 44 | </div> |
| 45 | <?php endif; ?> |
| 46 | </div> |
| 47 |  |
| 48 | <div class="mb-3"> |
| 49 | <label for="confirm\_password" class="form-label">Konfirmasi Password</label> |
| 50 | <input type="password" class="form-control <?= (session('validation') && session('validation')->hasError('confirm\_password')) ? 'is-invalid' : '' ?>" |
| 51 | id="confirm\_password" name="confirm\_password"> |
| 52 | <?php if (session('validation') && session('validation')->hasError('confirm\_password')) : ?> |
| 53 | <div class="invalid-feedback"> |
| 54 | <?= session('validation')->getError('confirm\_password') ?> |
| 55 | </div> |
| 56 | <?php endif; ?> |
| 57 | </div> |
| 58 |  |
| 59 | <div class="d-grid"> |
| 60 | <button type="submit" class="btn btn-primary">Register</button> |
| 61 | </div> |
| 62 | </form> |
| 63 |  |
| 64 | <div class="mt-3 text-center"> |
| 65 | <p>Sudah punya akun? <a href="<?= base\_url('login') ?>">Login disini</a></p> |
| 66 | </div> |
| 67 | </div> |
| 68 | </div> |
| 69 | </div> |
| 70 | </div> |
| 71 | </div> |
| 72 | <?= $this->endSection() ?> |

Login.php(Views)

|  |  |
| --- | --- |
| 1 | <?= $this->extend('layout/template') ?> |
| 2 |  |
| 3 | <?= $this->section('content') ?> |
| 4 | <div class="container mt-5"> |
| 5 | <div class="row justify-content-center"> |
| 6 | <div class="col-md-6"> |
| 7 | <div class="card"> |
| 8 | <div class="card-header"> |
| 9 | <h4 class="mb-0">Login</h4> |
| 10 | </div> |
| 11 | <div class="card-body"> |
| 12 | <?php if (session()->getFlashdata('error')) : ?> |
| 13 | <div class="alert alert-danger"> |
| 14 | <?= session()->getFlashdata('error') ?> |
| 15 | </div> |
| 16 | <?php endif; ?> |
| 17 |  |
| 18 | <?php if (session()->getFlashdata('success')) : ?> |
| 19 | <div class="alert alert-success"> |
| 20 | <?= session()->getFlashdata('success') ?> |
| 21 | </div> |
| 22 | <?php endif; ?> |
| 23 |  |
| 24 | <form action="<?= base\_url('login') ?>" method="post"> |
| 25 | <?= csrf\_field() ?> |
| 26 |  |
| 27 | <div class="mb-3"> |
| 28 | <label for="username" class="form-label">Username</label> |
| 29 | <input type="text" class="form-control" id="username" name="username" value="<?= old('username') ?>" required> |
| 30 | </div> |
| 31 |  |
| 32 | <div class="mb-3"> |
| 33 | <label for="password" class="form-label">Password</label> |
| 34 | <input type="password" class="form-control" id="password" name="password" required> |
| 35 | </div> |
| 36 |  |
| 37 | <div class="d-grid"> |
| 38 | <button type="submit" class="btn btn-primary">Login</button> |
| 39 | </div> |
| 40 | </form> |
| 41 |  |
| 42 | <div class="mt-3 text-center"> |
| 43 | <p>Belum punya akun? <a href="<?= base\_url('register') ?>">Daftar disini</a></p> |
| 44 | </div> |
| 45 | </div> |
| 46 | </div> |
| 47 | </div> |
| 48 | </div> |
| 49 | </div> |
| 50 | <?= $this->endSection() ?> |

Template.php(Views)

|  |  |
| --- | --- |
| 1 | <!DOCTYPE html> |
| 2 | <html lang="en"> |
| 3 | <head> |
| 4 | <meta charset="UTF-8"> |
| 5 | <meta name="viewport" content="width=device-width, initial-scale=1.0"> |
| 6 | <link href="https://cdn.jsdelivr.net/npm  /bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet"> |
| 7 | </head> |
| 8 | <body> |
| 9 | <nav class="navbar navbar-expand-lg navbar-dark bg-dark"> |
| 10 | <div class="container"> |
| 11 | <a class="navbar-brand" href="<?= base\_url() ?>">Sistem Manajemen Buku</a> |
| 12 | <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav"> |
| 13 | <span class="navbar-toggler-icon"></span> |
| 14 | </button> |
| 15 | <div class="collapse navbar-collapse" id="navbarNav"> |
| 16 | <ul class="navbar-nav ms-auto"> |
| 17 | <?php if (session()->get('isLoggedIn')) : ?> |
| 18 | <li class="nav-item"> |
| 19 | <a class="nav-link" href="<?= base\_url('books') ?>">Daftar Buku</a> |
| 20 | </li> |
| 21 | <li class="nav-item"> |
| 22 | <a class="nav-link" href="<?= base\_url('logout') ?>">Logout</a> |
| 23 | </li> |
| 24 | <?php else : ?> |
| 25 | <li class="nav-item"> |
| 26 | <a class="nav-link" href="<?= base\_url('login') ?>">Login</a> |
| 27 | </li> |
| 28 | <li class="nav-item"> |
| 29 | <a class="nav-link" href="<?= base\_url('register') ?>">Register</a> |
| 30 | </li> |
| 31 | <?php endif; ?> |
| 32 | </ul> |
| 33 | </div> |
| 34 | </div> |
| 35 | </nav> |
| 36 |  |
| 37 | <?= $this->renderSection('content') ?> |
| 38 |  |
| 39 | <script src="https://cdn.jsdelivr.net/npm/bootstrap  @5.3.0/dist/js/bootstrap.bundle.min.js"></script> |
| 40 | </body> |
| 41 | </html> |

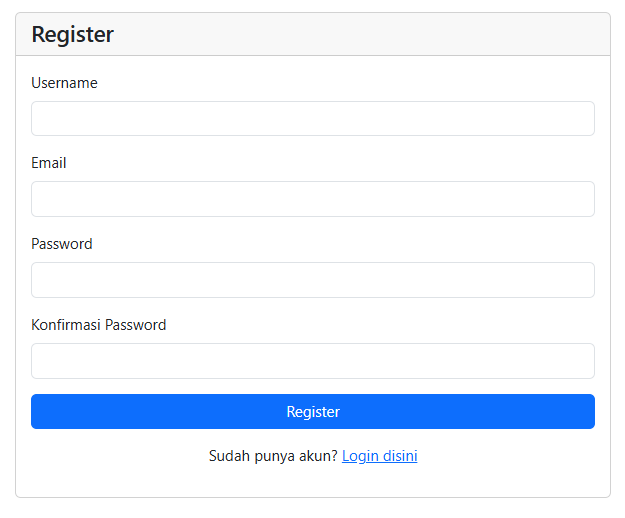
Routes.php(config)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23 | <?php  use CodeIgniter\Router\RouteCollection;  /\*\*  \* @var RouteCollection $routes  \*/  $routes->get('/', 'Home::index');  $routes->get('/login', 'Auth::login');  $routes->post('/login', 'Auth::attemptLogin');  $routes->get('/register', 'Auth::register');  $routes->post('/register', 'Auth::attemptRegister');  $routes->get('/logout', 'Auth::logout');  $routes->group('books', ['filter' => 'auth'], function($routes) {  $routes->get('/', 'Books::index');  $routes->get('create', 'Books::create');  $routes->post('store', 'Books::store');  $routes->get('edit/(:num)', 'Books::edit/$1');  $routes->post('update/(:num)', 'Books::update/$1');  $routes->post('delete/(:num)', 'Books::delete/$1');  }); |

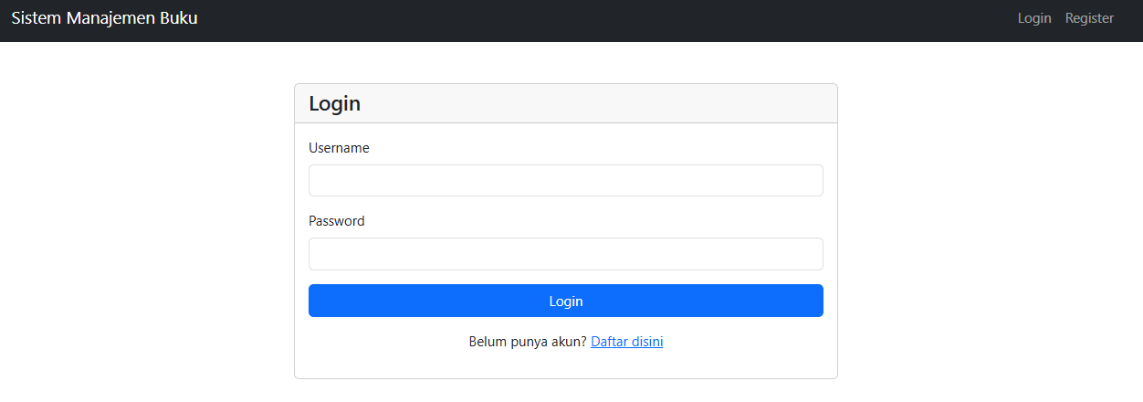
AuthFillter.php(Filters)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23 | <?php  namespace App\Filters;  use CodeIgniter\Filters\FilterInterface;  use CodeIgniter\HTTP\RequestInterface;  use CodeIgniter\HTTP\ResponseInterface;  class AuthFilter implements FilterInterface  {  public function before(RequestInterface $request, $arguments = null)  {  if (!session()->get('isLoggedIn')) {  session()->setFlashdata('error', 'Login terlebih dahulu!');  return redirect()->to('/login');  }  }  public function after(RequestInterface $request, ResponseInterface $response, $arguments = null)  {    }  } |

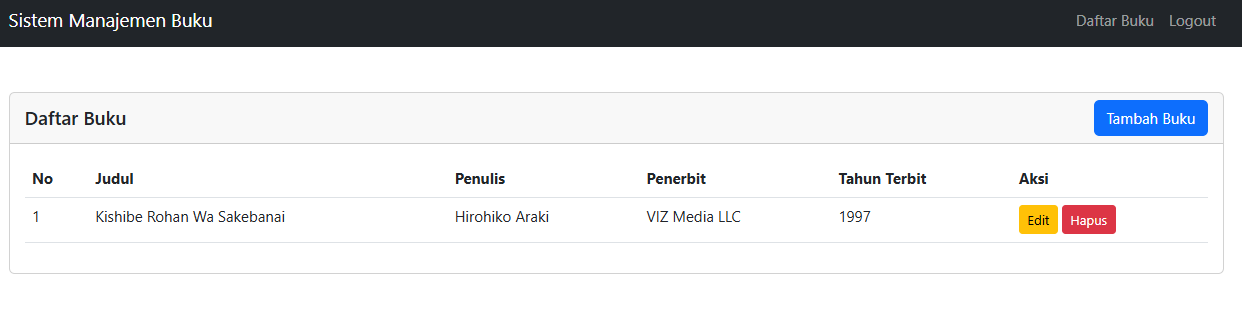
## Output Program



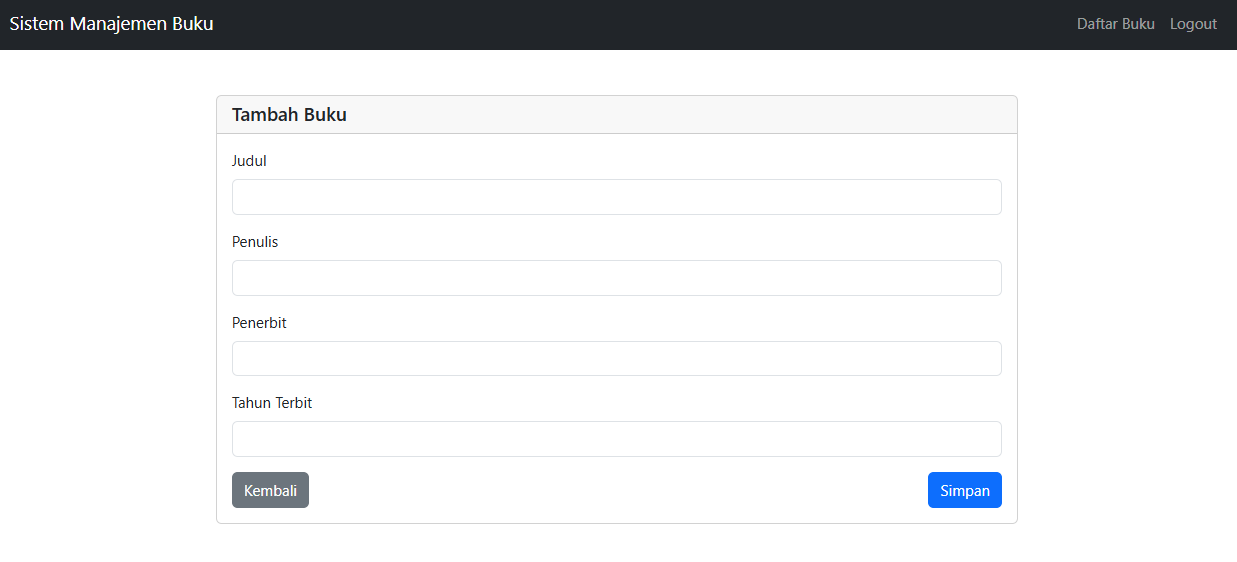
Gambar 1. Screenshot Hasil Jawaban Soal 1 Halaman Register



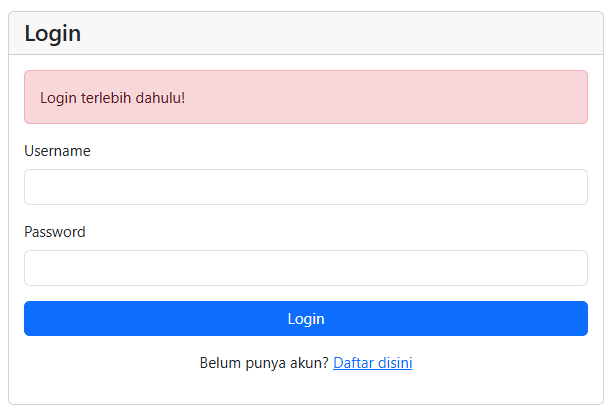
Gambar 2 Screenshot Hasil Jawaban Soal 1 Halaman Login



Gambar 3 Screenshot Hasil Jawaban Soal 1 Halaman Utama



Gambar 4 Screenshot Hasil Jawaban Soal 1 Halaman create



Gambar 5 Screenshot Hasil Jawaban Soal 1 Jika memaksa ke halaman lain sebelum login

## Pembahasan

template.php merupakan layout utama untuk seluruh tampilan. File ini memuat struktur HTML dasar, link Bootstrap, dan navbar dinamis sesuai status login user.

Baris [6] memuat CSS Bootstrap dari CDN agar tampilan responsif dan modern.

Baris [11] menampilkan teks “Sistem Manajemen Buku” sebagai brand dan link ke halaman utama.

Baris [17-31] adalah logika PHP untuk menampilkan menu navigasi yang berbeda tergantung apakah user sudah login atau belum.

Baris [37] <?= $this->renderSection('content') ?> digunakan untuk menampilkan isi halaman dinamis.

Baris [39] memuat script JavaScript Bootstrap agar navbar bisa collapse di mobile.

login.php adalah form login untuk user. Baris [11-22] menampilkan alert jika ada error login atau pesan sukses, seperti setelah registrasi.

Baris [24] membuka form login dengan metode POST ke route /login.

Baris [29] dan [34] adalah input untuk username dan password.

Baris [38] tombol submit.

Baris [43] berisi link ke halaman register jika belum punya akun.

Register.php merupakan form registrasi akun baru. Baris [12] form POST ke /register.

Baris [17-23] input username, dengan validasi jika error akan muncul.

Baris [27-33] input email. Baris [37-45] input password.

Baris [49-56] input konfirmasi password.

Baris [60] tombol submit. Baris [65] link ke login jika sudah punya akun.

Sedangkan index.php untuk menampilkan daftar buku dalam bentuk tabel.

Baris [10] tombol “Tambah Buku” mengarah ke /books/create.

Baris [13-17] alert jika ada flashdata sukses (misalnya setelah tambah/edit buku). Baris [20-28] header tabel.

Baris [32-47] perulangan untuk menampilkan setiap buku di tabel. Ada tombol Edit dan Hapus.

Baris [49-52] jika tidak ada data buku, tampilkan “Tidak ada data buku”.

Create.php dan edit.php keduanya hampir sama, berisi form input data buku. create.php untuk tambah buku, edit.php untuk ubah data.

Baris [12] form mengarah ke /books/store atau /books/update/{id}.

Baris [15-57] masing-masing adalah input untuk judul, penulis, penerbit, tahun terbit, lengkap dengan validasi error.

Baris [60-61] tombol Simpan (create) atau Update (edit), dan tombol Kembali.

Auth.php merupakan Controller untuk login, register, logout.

Baris [5] mendefinisikan kelas Auth yang extends BaseController.

Baris [11-13] membuat instance UserModel.

Baris [15-20] jika user sudah login, redirect ke /books, jika belum tampilkan form login. Baris [22-58] proses login:

Baris [24-31] validasi input kosong.

Baris [33-36] ambil user berdasarkan username.

Baris [38-48] verifikasi password dan jika cocok, simpan data ke session.

Baris [60-67] tampilkan form register jika belum login.

Baris [69-112] proses register:

Baris [71-76] validasi input form.

Baris [78-97] pesan error khusus jika validasi gagal. Baris [103-109] simpan user baru ke database dengan password yang di-hash.

Baris [115-118] method logout: hapus session dan redirect ke login.

Books.php untuk Controller untuk CRUD buku.

Baris [7] mendefinisikan class Books.

Baris [13] membuat instance BookModel.

Baris [16-24] method index(), ambil semua buku dan tampilkan ke view.

Baris [26-34] method create(), tampilkan form tambah buku.

Baris [36-51] method store(), validasi lalu simpan buku ke database.

Baris [53-66] method edit(), cari buku berdasarkan id dan tampilkan ke form edit.

Baris [68-83] method update(), simpan perubahan ke database.

Baris [85-89] method delete(), hapus buku dari database.

UserModel.php adalah model untuk user. Mengatur nama tabel, field yang bisa diisi, validasi, dan pesan error.

Baris [5] class UserModel extends Model.

Baris [7-15] mengatur properti seperti nama tabel, primary key, return type.

Baris [17-21] validasi untuk username, email, password.

Baris [23-38] pesan error untuk masing-masing field jika validasi gagal.

BookModel.php merupakan model untuk buku.

Baris [5] class BookModel extends Model.

Baris [7-15] seperti UserModel, mengatur tabel dan field.

Baris [18-49] validasi dan pesan error untuk judul, penulis, penerbit, dan tahun\_terbit.

## Tautan Git

Berikut adalah tautan untuk source code yang telah dibuat.

https://github.com/NaufalElyzar11/Pemrogramman-Web-ii/tree/main/Praktikum/Modul%206