

**LAPORAN PRAKTIKUM
PEMROGRAMAN WEB II
MODUL 6**



CRUD dan Login pada Codeigniter

Oleh:

Aditya Oktaviari

NIM. 2210817110008

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS LAMBUNG MANGKURAT
MEI 2024**

LEMBAR PENGESAHAN
LAPORAN PRAKTIKUM PEMROGRAMAN WEB II
MODUL 6

Laporan Praktikum Pemrograman Web II Modul 6: CRUD dan Login pada Codeigniter ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman Web II. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Aditya Oktaviari
NIM : 2210817110008

Menyetujui,
Asisten Praktikum

Mengetahui,
Dosen Penanggung Jawab Praktikum

Muhammad Ammarin Ihsan
NIM. 2010817210002

Ir. Eka Setya Wijaya, S.T., M.Kom.
NIP. 198205082008011010

DAFTAR ISI

LEMBAR PENGESAHAN	2
DAFTAR ISI	3
DAFTAR GAMBAR.....	4
DAFTAR TABEL	5
SOAL.....	6
A. Source Code	6
B. Output Program.....	43
C. Pembahasan.....	45
D. Tautan Git	52

DAFTAR GAMBAR

Gambar 1. Halaman Login	43
Gambar 2. Halaman Register.....	43
Gambar 3. Halaman Home	44
Gambar 4. Halaman Untuk Menambahkan Data.....	44
Gambar 5. Halaman Untuk Edit Data.....	45

DAFTAR TABEL

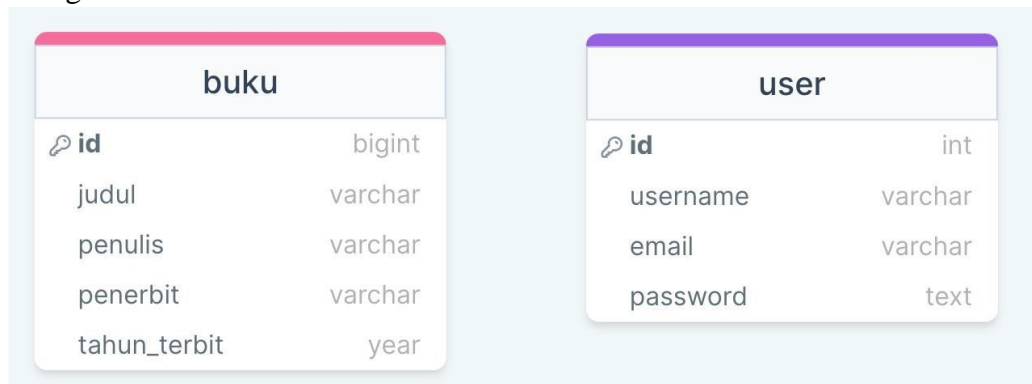
Tabel 1. Source Code Routes.php	7
Tabel 2. Source Code Buku.php.....	10
Tabel 3. Source Code Home.php.....	10
Tabel 5. Source Code Login.php.....	12
Tabel 6. Source Code Register.php	13
Tabel 7. Source Code AuthFilter.php.....	14
Tabel 8. Source Code vw_add.php.....	17
Tabel 9. Source Code vw_buku.php.....	23
Tabel 10. Source Code vw_edit.php.....	26
Tabel 11. Source Code vw_login.php	30
Tabel 12. Source Code vw_register.php.....	34
Tabel 13. Source Code BukuModel.php	35
Tabel 14. Source Code UserModel.php.....	36
Tabel 15. Source Code 2024-06-03-103310_Users.php	37
Tabel 16. Source Code 2024-06-03-122804_Buku.php.....	38
Tabel 17. Source Code Seeds Buku.php.....	38
Tabel 18. Source Code UserSeeder.php	39
Tabel 19. Source Code perpustakaan.sql.....	42

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 tampilan 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 sebagai berikut:



Simpan dalam folder Github PRAK601

A. Source Code

```
1 <?php
2
3 use CodeIgniter\Router\RouteCollection;
4
5 /**
6  * @var RouteCollection $routes
7  */
8 $routes->setAutoRoute(true);
9 $routes->get('/', 'Buku::login');
10 $routes->get('home', 'Buku::index');
11 $routes->get('/register', 'Register::index');
12 $routes->post('/register/process', 'Register::process');
```

13	\$routes->get('/login', 'Login::index');
14	\$routes->post('/login/process', 'Login::process');
15	\$routes->get('/logout', 'Login::logout');
16	\$routes->get('/buku/create', 'Buku::create');
17	\$routes->post('buku/store', 'Buku::store');
18	\$routes->post('buku/update/', 'Buku::update', ['filter' => 'auth']);
19	\$routes->get('buku/edit/', 'Buku::edit', ['filter' => 'auth']);
20	\$routes->get('buku/delete/', 'Buku::delete', ['filter' => 'auth']);

Tabel 1. Source Code Routes.php

1	<?php namespace App\Controllers;
2	
3	use App\Models\BukuModel;
4	use CodeIgniter\Controller;
5	use App\Models\PostModel;
6	
7	class Buku extends Controller
8	{
9	/**
10	*
11	* @var Model
12	*/
13	protected \$model;
14	
15	public function __construct()
16	{
17	\$this->model = new BukuModel();
18	\$this->helpers = ['form', 'url'];
19	
20	}
21	
22	public function login()
23	{
24	\$data = [
25	'posts' => \$this->model->paginate(10),
26	'pager' => \$this->model->pager,
27	'title' => 'POST LIST'
28];
29	
30	return view('vw_login', \$data);
31	}
32	
33	public function index()
34	{
35	if (!session()->get('logged_in')) {

```

36         return redirect()->to(base_url('/login'));
37     }
38     $data = [
39         'posts' => $this->model->paginate(10),
40         'pager' => $this->model->pager,
41         'title' => 'POST LIST'
42     ];
43
44     return view('vw_buku', $data);
45 }
46
47 public function create()
48 {
49     // Check if user is authenticated
50     if (!session()->get('logged_in')) {
51         return redirect()->to(base_url('/login'));
52     }
53     $data = ['title' => 'Create new post'];
54
55     return view('vw_add', $data);
56 }
57
58 public function store()
59 {
60     $data = $this->request->getPost(['judul',
61     'penulis', 'penerbit', 'tahun_terbit']);
62
63     if (! $this->validateData($data, $this->model-
64     >validationRules)) {
65         return $this->create();
66     }
67
68     $post = $this->validator->getValidated();
69
70     $save = $this->model->save([
71         'judul' => $post['judul'],
72         'penulis' => $post['penulis'],
73         'penerbit' => $post['penerbit'],
74         'tahun_terbit' => $post['tahun_terbit'],
75     ]);
76
77     session()->setFlashdata('success', 'Post has
78     been added successfully.');
```

```

79     return redirect()->to(base_url('home'));
80 }
81
82 public function edit($id)

```



```

80     {
81         // Check if user is authenticated
82         if (!session()->get('logged_in')) {
83             return redirect()->to(base_url('/login'));
84         }
85         $post = $this->model->find($id);
86         if (empty($post)) {
87             session()->setFlashdata('error', 'Post      not
found');
88             return redirect()->back();
89         }
90         $data = [
91             'title' => 'Edit Post',
92             'post' => $post
93         ];
94
95         return view('vw_edit', $data);
96     }
97
98     public function update($id)
99     {
100         // Check if user is authenticated
101         if (!session()->get('logged_in')) {
102             return redirect()->to(base_url('/login'));
103         }
104         $post = $this->model->find($id);
105
106         if (empty($post)) {
107             session()->setFlashdata('error', 'Post      not
found');
108             return redirect()->back();
109         }
110
111         $data      =      $this->request->getPost(['judul',
'penulis', 'penerbit', 'tahun_terbit']);
112
113         if (! $this->validateData($data, $this->model-
>validationRules)) {
114             return $this->create();
115         }
116
117         $updatedPost = $this->validator->getValidated();
118
119         $update      =      $this->model->update($post['id'],
$updatedPost);
120
121         if ($update) {

```

122	session()->setFlashdata('success', 'Post has been updated successfully');
123	return redirect()->to(base_url('home'));
124	} else {
125	session()->setFlashdata('error', 'Some problems occurred, please try again.');
126	return redirect()->back();
127	}
128	}
129	
130	public function delete(\$id)
131	{
132	// Check if user is authenticated
133	if (!session()->get('logged_in')) {
134	return redirect()->to(base_url('home'));
135	}
136	
137	if (empty(\$id)) {
138	return redirect()->to(base_url('home'));
139	}
140	
141	\$delete = \$this->model->delete(\$id);
142	
143	if (\$delete) {
144	session()->setFlashdata('success', 'Post has been removed successfully.');
145	return redirect()->to(base_url('home'));
146	} else {
147	session()->setFlashdata('error', 'Some problems occurred, please try again.');
148	return redirect()->to(base_url('home'));
149	}
150	}
151	}

Tabel 2. Source Code Buku.php

1	<?php
2	
3	namespace App\Controllers;
4	
5	class Home extends BaseController
6	{
7	public function index(): string
8	{
9	return view('vw_buku');
10	}
11	}

Tabel 3. Source Code Home.php

```

1  <?php
2
3  namespace App\Controllers;
4
5  use App\Models\UsersModel;
6
7  class Login extends BaseController
8  {
9      public function index()
10     {
11         return view('vw_login');
12     }
13
14     public function process()
15     {
16         $users = new UsersModel();
17         $username = $this->request->getVar('username');
18         $password = $this->request->getVar('password');
19         $dataUser = $users->where('username', $username)-
20 >first();
21
22         if ($dataUser) {
23             // Check the password using object property
24             syntax
25             if (password_verify($password, $dataUser-
26 >password)) {
27                 session()->set([
28                     'username' => $dataUser->username,
29                     'logged_in' => true
30                 ]);
31                 return redirect()->to(base_url('home'));
32             } else {
33                 session()->setFlashdata('error',
34 'Username & Password Salah');
35                 return redirect()->to(base_url('login'));
36             }
37         } else {
38             session()->setFlashdata('error', 'Username &
39 Password Salah');
40             return redirect()->to(base_url('login'));
41         }
42     }
43
44     public function logout()
45     {
46         session()->destroy();
47         return redirect()->to(base_url('login'));
48     }
49 }

```

43	}
44	}

Tabel 4. Source Code Login.php

1	<?php
2	
3	namespace App\Controllers;
4	
5	use App\Models\UsersModel;
6	
7	class Register extends BaseController
8	{
9	public function index()
10	{
11	return view('vw_register');
12	}
13	
14	public function process()
15	{
16	// Validate input
17	if (!\$this->validate([
18	'username' => [
19	'rules' =>
20	'required min_length[4] max_length[20] is_unique[users.us
21	ername]',
22	'errors' => [
23	'required' => '{field} Harus diisi',
24	'min_length' => '{field} Minimal 4
25	Karakter',
26	'max_length' => '{field} Maksimal 20
27	Karakter',
28	'is_unique' => 'Username sudah
29	digunakan sebelumnya'
30]
31],
32	'password' => [
33	'rules' =>
34	'required min_length[0] max_length[50]',
35	'errors' => [
36	'required' => '{field} Harus diisi',
	'min_length' => '{field} Minimal 0
	Karakter',
	'max_length' => '{field} Maksimal 50
	Karakter',
]
],
	'password_conf' => [
	'rules' => 'matches[password]',

```

37         'errors' => [
38             'matches' => 'Konfirmasi Password
tidak sesuai dengan password',
39         ],
40     ],
41     'email' => [
42         'rules' =>
'required|min_length[4]|max_length[100]|valid_email',
43         'errors' => [
44             'required' => '{field} Harus diisi',
45             'min_length' => '{field} Minimal 4
Karakter',
46             'max_length' => '{field} Maksimal 100
Karakter',
47             'valid_email' => 'Email tidak valid'
48         ]
49     ],
50 )) {
51     session()->setFlashdata('error', $this-
>validator->listErrors());
52     return redirect()->back()->withInput();
53 }
54
55 $users = new UsersModel();
56
57 // Hash the password before storing it
58 $hashedPassword = password_hash($this->request-
>getVar('password'), PASSWORD_DEFAULT);
59
60 $users->insert([
61     'username' => $this->request-
>getVar('username'),
62     'password' => $hashedPassword,
63     'email' => $this->request->getVar('email')
64 ]);
65
66 return redirect()->to('/login');
67 }
68 }

```

Tabel 5. Source Code Register.php

```

1 <?php
2 // app/Filters/AuthFilter.php
3
4 namespace App\Filters;
5
6 use CodeIgniter\HTTP\RequestInterface;
7 use CodeIgniter\HTTP\ResponseInterface;

```

```

8 use CodeIgniter\Filters\FilterInterface;
9
10 class AuthFilter implements FilterInterface
11 {
12     public function before(RequestInterface $request,
13 $arguments = null)
14     {
15         // Check if the user is logged in
16         if (!session()->has('user_id')) {
17             // If not logged in, redirect to the login
18             page
19             return redirect()->to('/login');
20         }
21     }
22     public function after(RequestInterface $request,
23 ResponseInterface $response, $arguments = null)
24     {
25         // Do nothing after the request
26     }
27 }

```

Tabel 6. Source Code AuthFilter.php

```

1 <?= $this->extend('layouts/main') ?>
2
3 <?= $this->section('content') ?>
4 <style>
5     body {
6         background-image:
7         url('https://images.unsplash.com/photo-1661097410573-
8         16d926ec1ec7?q=80&w=2070&auto=format&fit=crop&ixlib=rb-
9         4.0.3&ixid=M3wxMjA3fDB8MHxwaG90bylwyWdlfHx8fGVufDB8fHx8f
10         A%3D%3D');
11         background-repeat: no-repeat;
12         background-attachment: fixed;
13         background-position: center;
14         background-size: cover;
15         color: #ececec;
16     }
17     .card {
18         background: rgba(31, 31, 31); /* Opasitas
19         background 20% */
20         color: #ececec;
21         box-shadow: 0 6px 10px rgba(0, 0, 0, 0.15);
22         border-radius: 10px;
23     }
24 }

```

```

21     .card-header {
22         background-color: #00bfa5;
23         border-bottom: 1px solid #37474f;
24     }
25
26     .btn-success {
27         background-color: #00bfa5;
28         border: none;
29     }
30
31     .btn-success:hover {
32         background-color: #008e76;
33     }
34
35     .btn-secondary {
36         background-color: #000000;
37         border: none;
38     }
39
40     .btn-secondary:hover {
41         background-color: #333333;
42     }
43
44     .alert {
45         background-color: rgba(255, 138, 101, 0.1);
46         color: #ffab40;
47         border: 1px solid #ffab40;
48     }
49
50     .form-control {
51         background-color: #2e2e2e;
52         color: #ececec;
53         border: 1px solid #37474f;
54     }
55
56     .form-control::placeholder {
57         color: #bdbdbd;
58     }
59
60     label {
61         color: #ececec;
62     }
63 </style>
64
65 <div class="container mt-5">
66     <div class="row justify-content-center">
67         <div class="col-md-8">

```


	<code>?>" class="btn btn-secondary">Kembali</code>
101	<code> </div></code>
102	<code> <?= form_close(); ?></code>
103	<code> </div></code>
104	<code></div></code>
105	<code></div></code>
106	<code></div></code>
107	<code></div></code>
108	
109	<code><?= \$this->endSection() ?></code>
110	
111	<code><?= \$this->section('extra-js') ?></code>
112	<code><!-- include summernote css/js --></code>
113	<code><link</code>
	<code>href="https://cdn.jsdelivrivr.net/npm/summernote@0.8.15/dist/summernote-bs4.min.css" rel="stylesheet"></code>
114	<code><script</code>
	<code>src="https://cdn.jsdelivrivr.net/npm/summernote@0.8.15/dist/summernote-bs4.min.js"></script></code>
115	<code><script></code>
116	<code> \$(document).ready(function() {</code>
117	<code> \$('#post_content').summernote({</code>
118	<code> tabsize: 2,</code>
119	<code> height: 500</code>
120	<code> });</code>
121	<code> });</code>
122	<code></script></code>
123	<code><?= \$this->endSection() ?></code>

Tabel 7. Source Code vw_add.php

1	<code><?= \$this->extend('layouts/main') ?></code>
2	
3	<code><?= \$this->section('content') ?></code>
4	<code><style></code>
5	<code> body,</code>
6	<code> html {</code>
7	<code> margin: 0;</code>
8	<code> padding: 0;</code>
9	<code> font-family: Arial, sans-serif;</code>
10	<code> height: 100%;</code>
11	<code> display: flex;</code>
12	<code> flex-direction: column;</code>
13	<code> background-color: #121212;</code>
14	<code> color: #ececec;</code>
15	<code> position: relative;</code>
16	<code> overflow: hidden;</code>
17	<code> }</code>
18	

```

19     body::before {
20         content: "";
21         position: absolute;
22         top: 0;
23         left: 0;
24         width: 100%;
25         height: 100%;
26         background-image:
url('https://images.unsplash.com/photo-1661097410573-
16d926ec1ec7?q=80&w=2070&auto=format&fit=crop&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8f
A%3D%3D');
27         background-size: cover;
28         background-position: center;
29         opacity: 0.1;
30         z-index: 1;
31     }
32
33     .navbar {
34         z-index: 2;
35         position: relative;
36     }
37
38     .container,
39     .card {
40         z-index: 2;
41         position: relative;
42     }
43
44     .navbar {
45         background-color: #1f1f1f !important;
46         border-bottom: 1px solid #37474f;
47     }
48
49     .navbar-brand,
50     .nav-link {
51         color: #00bfa5 !important;
52     }
53
54     .card {
55         background: #1f1f1f;
56         color: #ececec;
57         box-shadow: 0 6px 10px rgba(0, 0, 0, 0.15);
58         border-radius: 10px;
59     }
60
61     .card-header {

```

```

62         background-color: #00bfa5;
63         border-bottom: 1px solid #37474f;
64     }
65
66     .btn {
67         background-color: #00bfa5;
68         color: white;
69         border: none;
70     }
71
72     .btn:hover {
73         background-color: #008e76;
74     }
75
76     .table thead {
77         background-color: #37474f;
78     }
79
80     .table th,
81     .table td {
82         border: 1px solid #37474f;
83         color: #fff; /* Change table headers and content
text color to white */
84     }
85
86     .pagination li a {
87         background-color: #1f1f1f !important;
88         color: #00bfa5 !important;
89         border: 1px solid #00bfa5 !important;
90     }
91
92     .pagination li a:hover {
93         background-color: #00bfa5 !important;
94         color: #1f1f1f !important;
95     }
96
97     .alert {
98         background-color: rgba(255, 138, 101, 0.1);
99         color: #ffab40;
100        border: 1px solid #ffab40;
101    }
102 </style>
103
104 <!-- Navbar -->
105 <nav class="navbar navbar-expand-lg navbar-light shadow-
sm">
106     <div class="container">

```

```

107         <a class="navbar-brand" href="<?= base_url('/')
?>">My Library</a>
108         <button class="navbar-toggler" type="button"
data-toggle="collapse" data-target="#navbarNav" aria-
controls="navbarNav" aria-expanded="false" aria-
label="Toggle navigation">
109             <span class="navbar-toggler-icon"></span>
110         </button>
111         <div class="collapse navbar-collapse"
id="navbarNav">
112             <ul class="navbar-nav ml-auto">
113                 <?php if (session()->get('logged_in')) :
?>
114                     <li class="nav-item">
115                         <a class="nav-link"
href="#">Welcome, <?= session()->get('username'); ?></a>
116                     </li>
117                     <li class="nav-item">
118                         <a class="nav-link" href="<?=
base_url('login/logout'); ?>">Logout</a>
119                     </li>
120                     <?php else : ?>
121                     <li class="nav-item">
122                         <a class="nav-link" href="<?=
base_url('login'); ?>">Login</a>
123                     </li>
124                     <li class="nav-item">
125                         <a class="nav-link" href="<?=
base_url('register'); ?>">Register</a>
126                     </li>
127                     <?php endif; ?>
128                 </ul>
129             </div>
130         </div>
131     </nav>
132
133     <div class="container mt-5">
134         <div class="row justify-content-center">
135             <div class="col-md-10">
136                 <div class="card shadow-sm">
137                     <div class="card-header text-white">
138                         <h5 class="mb-0">List Buku</h5>
139                         <a href="<?=
base_url('buku/create'); ?>" class="btn btn-sm float-
right">Tambah Data</a>
140                     </div>
141                     <div class="card-body">

```

```

142         <?php           if           (session() -
>getFlashdata('success')) : ?>
143             <div         class="alert     alert-
success">
144                 <?=           session() -
>getFlashdata('success'); ?>
145             </div>
146         <?php endif; ?>
147
148         <?php           if           (session() -
>getFlashdata('error')) : ?>
149             <div class="alert alert-danger">
150                 <?=           session() -
>getFlashdata('error'); ?>
151             </div>
152         <?php endif; ?>
153
154         <table class="table table-bordered
table-striped table-hover">
155             <thead class="text-center">
156                 <tr>
157                     <th
scope="col">Judul</th>
158                     <th
scope="col">Penulis</th>
159                     <th
scope="col">Penerbit</th>
160                     <th         scope="col">Tahun
Terbit</th>
161                     <th
scope="col">Action</th>
162                 </tr>
163             </thead>
164             <tbody>
165                 <?php if (!empty($posts) &&
is_array($posts)) : ?>
166                     <?php foreach ($posts as
$row) : ?>
167                         <tr>
168                             <td><?=
$row['judul']; ?></td>
169                             <td><?=
$row['penulis']; ?></td>
170                             <td><?=
$row['penerbit']; ?></td>
171                             <td><?=
$row['tahun_terbit']; ?></td>

```

```

172                                     <td class="text-
center">
173                                     <form
onsubmit="return confirm('Apakah Anda Yakin ?');"
action="<?= base_url('buku/delete/' . $row['id']); ?>"
method="POST">
174                                     <input
type="hidden" name="{csrf_token}" value="{csrf_hash}">
175                                     <input
type="hidden" name="_method" value="DELETE">
176                                     <a
href="<?= base_url('buku/edit/' . $row['id']); ?>"
class="btn btn-warning btn-sm">Edit</a>
177                                     <button
type="submit" class="btn btn-danger btn-
sm">Delete</button>
178                                     </form>
179                                     </td>
180                                     </tr>
181                                     <?php endforeach; ?>
182                                     <?php else : ?>
183                                     <tr>
184                                     <td colspan="5"
class="text-center">No post found.</td>
185                                     </tr>
186                                     <?php endif; ?>
187                                     </tbody>
188                                     </table>
189
190                                     <div class="d-flex justify-content-
center">
191                                     <?= $pager->links(); ?>
192                                     </div>
193                                     </div>
194                                     </div>
195                                     </div>
196                                     </div>
197 </div>
198
199 <?= $this->endSection() ?>
200
201 <?= $this->section('extra-js') ?>
202 <script>
203     $(document).ready(function() {
204         $('.pagination li').addClass('page-item');
205         $('.pagination li a').addClass('page-link');
206     });

```

207	</script>
208	<?= \$this->endSection() ?>

Tabel 8. Source Code vw_buku.php

1	<?= \$this->extend('layouts/main') ?>
2	
3	<?= \$this->section('content') ?>
4	<style>
5	body {
6	background-image:
	url('https://images.unsplash.com/photo-1661097410573-16d926ec1ec7?q=80&w=2070&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MHxwaG90bylwYWdlfHx8fGVufDB8fHx8fA%3D%3D');
7	background-repeat: no-repeat;
8	background-attachment: fixed;
9	background-position: center;
10	background-size: cover;
11	color: #ececec;
12	}
13	
14	.card {
15	background: #1f1f1f;
16	color: #ececec;
17	box-shadow: 0 6px 10px rgba(0, 0, 0, 0.15);
18	border-radius: 10px;
19	}
20	
21	.card-header {
22	background-color: #00bfa5;
23	border-bottom: 1px solid #37474f;
24	}
25	
26	.btn-success {
27	background-color: #00bfa5;
28	border: none;
29	}
30	
31	.btn-success:hover {
32	background-color: #008e76;
33	}
34	
35	.btn-secondary {
36	background-color: #000000;
37	border: none;
38	}
39	
40	.btn-secondary:hover {

```

41         background-color: #333333;
42     }
43
44     .alert {
45         background-color: rgba(255, 138, 101, 0.1);
46         color: #ffab40;
47         border: 1px solid #ffab40;
48     }
49
50     .form-control {
51         background-color: #2e2e2e;
52         color: #ecec;
53         border: 1px solid #37474f;
54     }
55
56     .form-control::placeholder {
57         color: #bdbdbd;
58     }
59
60     label {
61         color: #ecec;
62     }
63 </style>
64
65 <div class="container mt-5">
66     <div class="row justify-content-center">
67         <div class="col-md-8">
68             <div class="card shadow-sm">
69                 <div class="card-header text-white">
70                     <h4>Edit</h4>
71                 </div>
72                 <div class="card-body">
73
74                     <?php          if          (session()-
75 >getFlashdata('error')) : ?>
76                         <div class="alert alert-danger">
77                             <?=
78                             session()-
79 >getFlashdata('error'); ?>
80                             </div>
81                         <?php endif; ?>
82
83                         <?= validation_list_errors() ?>
84
85                         <?=      form_open('buku/update/'      .
86 $post['id']); ?>
87                         <input type="hidden" name="_method"
88 value="PUT">

```



```

84
85         <div class="form-group">
86             <label for="judul">Judul</label>
87             <input type="text" class="form-
control" id="judul" name="judul" placeholder="Masukkan
Judul" value="<?= $post['judul'] ?>">
88             </div>
89             <div class="form-group">
90                 <label
for="penulis">Penulis</label>
91                 <input type="text" class="form-
control" id="penulis" name="penulis"
placeholder="Masukkan Penulis" value="<?=
$post['penulis'] ?>">
92                 </div>
93                 <div class="form-group">
94                     <label
for="penerbit">Penerbit</label>
95                     <input type="text" class="form-
control" id="penerbit" name="penerbit"
placeholder="Masukkan Penerbit" value="<?=
$post['penerbit'] ?>">
96                     </div>
97                     <div class="form-group">
98                         <label for="tahun_terbit">Tahun
Terbit</label>
99                         <input type="number"
class="form-control" id="tahun_terbit"
name="tahun_terbit" placeholder="Masukkan Tahun Terbit"
value="<?= $post['tahun_terbit'] ?>">
100                         </div>
101                         <div class="form-group text-right">
102                             <button class="btn btn-
success">Update</button>
103                             <a href="<?= base_url('home')
?>" class="btn btn-secondary">Kembali</a>
104                             </div>
105                             <?= form_close(); ?>
106
107                     </div>
108                 </div>
109             </div>
110         </div>
111     </div>
112
113     <?= $this->endSection() ?>
114

```

115	<?= \$this->section('extra-js') ?>
116	<!-- include summernote css/js -->
117	<link href="https://cdn.jsdelivrivr.net/npm/summernote@0.8.15/dist/summernote-bs4.min.css" rel="stylesheet">
118	<script src="https://cdn.jsdelivrivr.net/npm/summernote@0.8.15/dist/summernote-bs4.min.js"></script>
119	<script>
120	\$(document).ready(function() {
121	\$('#post_content').summernote({
122	tabsize: 2,
123	height: 500
124	});
125	});
126	</script>
127	<?= \$this->endSection() ?>

Tabel 9. Source Code vw_edit.php

1	<!DOCTYPE html>
2	<html lang="en">
3	
4	<head>
5	<meta charset="UTF-8">
6	<meta name="viewport" content="width=device-width, initial-scale=1.0">
7	<title>Login</title>
8	<style>
9	body {
10	display: flex;
11	justify-content: center;
12	align-items: center;
13	flex-direction: column;
14	min-height: 100vh;
15	padding: 0;
16	margin: 0;
17	font-family: Arial, sans-serif;
18	background-color: #121212;
19	color: #ececec;
20	position: relative;
21	overflow: hidden;
22	}
23	
24	body::before {
25	content: "";
26	position: absolute;
27	top: 0;
28	left: 0;

```

29         width: 100%;
30         height: 100%;
31         background-image:
url('https://iispsm.sch.id/new/wp-
content/uploads/2021/06/library.jpg');
32         background-size: cover;
33         background-position: center;
34         opacity: 0.1; /* Set the opacity to 20% */
35         z-index: 1;
36     }
37
38     .welcome-message,
39     .form-signin {
40         z-index: 2;
41         position: relative;
42     }
43
44     .welcome-message {
45         width: 100%;
46         max-width: 600px;
47         padding: 20px;
48         text-align: center;
49         margin-bottom: 20px;
50         background: #1f1f1f;
51         border-radius: 10px;
52         box-shadow: 0 6px 10px rgba(0, 0, 0, 0.15);
53         color: #00bfa5;
54         font-size: 24px;
55     }
56
57     .form-signin {
58         width: 100%;
59         max-width: 330px;
60         padding: 20px;
61         margin: auto;
62         background: #1f1f1f;
63         border-radius: 10px;
64         box-shadow: 0 6px 10px rgba(0, 0, 0, 0.15);
65         position: relative;
66     }
67
68     .form-signin img {
69         margin-bottom: 20px;
70     }
71
72     .form-signin h1 {
73         margin-bottom: 10px;

```

```

74         font-size: 24px;
75         font-weight: normal;
76         color: #00bfa5;
77     }
78
79     .form-signin p {
80         margin-bottom: 20px;
81         font-size: 14px;
82         color: #b0bec5;
83     }
84
85     .form-signin input[type="text"],
86     .form-signin input[type="password"] {
87         width: 100%;
88         padding: 10px;
89         margin-bottom: 10px;
90         border: 1px solid #37474f;
91         border-radius: 4px;
92         background: #263238;
93         color: #ececec;
94         box-sizing: border-box;
95     }
96
97     .form-signin input[type="text"]::placeholder,
98     .form-signin input[type="password"]::placeholder
99 {
100     color: #90a4ae;
101 }
102
103     .form-signin button {
104         width: 100%;
105         padding: 10px;
106         background-color: #00bfa5;
107         color: white;
108         border: none;
109         border-radius: 4px;
110         cursor: pointer;
111         font-size: 16px;
112     }
113
114     .form-signin button:hover {
115         background-color: #008e76;
116     }
117
118     .form-signin .alert {
119         padding: 10px;
120         margin-bottom: 10px;

```

```

120         border-radius: 4px;
121         color: #ffab40;
122         background-color: rgba(255, 138, 101, 0.1);
123         border: 1px solid #ffab40;
124         display: none;
125         position: relative;
126     }
127
128     .form-signin .text-muted {
129         color: #b0bec5;
130         margin-top: 20px;
131     }
132
133     .btn-register {
134         display: block;
135         width: 93%;
136         text-align: center;
137         padding: 10px;
138         margin-top: 10px;
139         background-color: #37474f;
140         color: #00bfa5;
141         border: 1px solid #00bfa5;
142         border-radius: 4px;
143         cursor: pointer;
144         font-size: 14px;
145         text-decoration: none;
146     }
147
148     .btn-register:hover {
149         background-color: #00bfa5;
150         color: #121212;
151     }
152 </style>
153 </head>
154
155 <body>
156
157     <div class="welcome-message">
158         Selamat Datang di Perpustakaan Bersama
159     </div>
160
161     <main class="form-signin">
162         <!-- Alert untuk informasi saat halaman dimuat -
163         ->
164         <div class="alert" id="initial-alert"
165             style="display: block;">
166             Login terlebih dahulu!

```

165	</div>
166	
167	<!-- Alert untuk kesalahan login -->
168	<div class="alert" style="display: <?php echo (session()->getFlashdata('error')) ? 'block' : 'none'; ?>;">
169	<?= session()->getFlashdata('error'); ?>
170	</div>
171	
172	<form method="post" action="<?= base_url(); ?>/login/process">
173	<?= csrf_field(); ?>
174	<h1 class="h3 mb-3 fw-normal">Login</h1>
175	<p>Register dulu apabila belum memiliki akun</p>
176	
177	<input type="text" name="username" id="username" placeholder="Username" required autofocus>
178	<input type="password" name="password" id="password" placeholder="Password" required>
179	
180	<button type="submit">Login</button>
181	</form>
182	<a class="btn-register" href="<?= base_url('register'); ?>">Register
183	</main>
184	
185	</body>
186	
187	</html>

Tabel 10. Source Code vw_login.php

1	<!DOCTYPE html>
2	<html lang="en" class="h-100">
3	
4	<head>
5	<meta charset="utf-8">
6	<meta name="viewport" content="width=device-width, initial-scale=1.0">
7	<title>Register</title>
8	<style>
9	body,
10	html {
11	margin: 0;
12	padding: 0;
13	font-family: Arial, sans-serif;
14	height: 100%;
15	display: flex;

```
16         justify-content: center;
17         align-items: center;
18         flex-direction: column;
19         background-color: #121212;
20         color: #ececec;
21     }
22
23     body::before {
24         content: "";
25         position: absolute;
26         top: 0;
27         left: 0;
28         width: 100%;
29         height: 100%;
30         background-image:
31         url('https://iispsm.sch.id/new/wp-
32         content/uploads/2021/06/library.jpg');
33         background-size: cover;
34         background-position: center;
35         opacity: 0.1;
36         z-index: 1;
37     }
38
39     .welcome-message,
40     .form-register {
41         z-index: 2;
42         position: relative;
43     }
44
45     .welcome-message {
46         width: 100%;
47         max-width: 600px;
48         padding: 20px;
49         text-align: center;
50         margin-bottom: 20px;
51         background: #1f1f1f;
52         border-radius: 10px;
53         box-shadow: 0 6px 10px rgba(0, 0, 0, 0.15);
54         color: #00bfa5;
55         font-size: 24px;
56     }
57
58     .form-register {
59         width: 100%;
60         max-width: 330px;
61         padding: 20px;
62         margin: auto;
```

```

61         background: #1f1f1f;
62         border-radius: 10px;
63         box-shadow: 0 6px 10px rgba(0, 0, 0, 0.15);
64         position: relative;
65     }
66
67     .form-register h1 {
68         margin-bottom: 10px;
69         font-size: 24px;
70         font-weight: normal;
71         color: #00bfa5;
72     }
73
74     .form-register p {
75         margin-bottom: 20px;
76         font-size: 14px;
77         color: #b0bec5;
78     }
79
80     .form-register input[type="text"],
81     .form-register input[type="password"],
82     .form-register input[type="email"] {
83         width: 100%;
84         padding: 10px;
85         margin-bottom: 10px;
86         border: 1px solid #37474f;
87         border-radius: 4px;
88         background: #263238;
89         color: #ececec;
90         box-sizing: border-box;
91     }
92
93     .form-register input[type="text"]::placeholder,
94     .form-register
input[type="password"]::placeholder,
95     .form-register input[type="email"]::placeholder
96     {
97         color: #90a4ae;
98     }
99
100    .form-register button {
101        width: 100%;
102        padding: 10px;
103        background-color: #00bfa5;
104        color: white;
105        border: none;
106        border-radius: 4px;

```



```

106         cursor: pointer;
107         font-size: 16px;
108     }
109
110     .form-register button:hover {
111         background-color: #008e76;
112     }
113
114     .form-register .alert {
115         padding: 10px;
116         margin-bottom: 10px;
117         border-radius: 4px;
118         color: #ffab40;
119         background-color: rgba(255, 138, 101, 0.1);
120         border: 1px solid #ffab40;
121         display: none;
122         position: relative;
123     }
124
125     .form-register .text-muted {
126         color: #b0bec5;
127         margin-top: 20px;
128     }
129 </style>
130 </head>
131
132 <body>
133
134     <div class="welcome-message">
135         Form Pendaftaran
136     </div>
137
138     <main class="form-register">
139         <!-- Alert untuk informasi saat halaman dimuat -
->
140         <div class="alert" id="initial-alert"
style="display: block;">
141             Isi form untuk mendaftar!
142         </div>
143
144         <!-- Alert untuk kesalahan register -->
145         <div class="alert" style="display: <?php echo
(session()->getFlashdata('error')) ? 'block' : 'none';
?> ">
146             <?= session()->getFlashdata('error'); ?>
147         </div>
148

```

```

149         <form method="post" action="<?= base_url();
?>/register/process">
150             <?= csrf_field(); ?>
151             <h1 class="h3 mb-3 fw-normal">Register</h1>
152             <p>Isi data di bawah untuk membuat akun
baru</p>
153
154             <input type="text" name="username"
id="username" placeholder="Username" required autofocus>
155             <input type="password" name="password"
id="password" placeholder="Password" required>
156             <input type="password" name="password_conf"
id="password_conf" placeholder="Confirm Password"
required>
157             <input type="email" name="email" id="email"
placeholder="Email" required>
158
159             <button type="submit">Register</button>
160         </form>
161     </main>
162
163 </body>
164
165 </html>

```

Tabel 11. Source Code vw_register.php

```

1  <?php
2
3  namespace App\Models;
4
5  use CodeIgniter\Model;
6
7  class BukuModel extends Model
8  {
9      protected $table = 'buku';
10     protected $allowedFields = ['judul', 'penulis' ,
'penerbit', 'tahun_terbit'];
11     protected $createdField = 'created_at';
12     protected $updatedField = 'updated_at';
13     protected $deletedField = 'deleted_at';
14
15     protected $validationRules = ([
16         'judul' => [
17             'rules' => 'required|string',
18             'errors' => [
19                 'required' => 'Judul harus diisi.',
20                 'string' => 'Judul harus berupa
string.'

```

```

21         ]
22     ],
23     'penulis' => [
24         'rules' => 'required|string',
25         'errors' => [
26             'required' => 'Penulis harus diisi.',
27             'string' => 'Penulis harus berupa
string.'
28         ]
29     ],
30     'penerbit' => [
31         'rules' => 'required|string',
32         'errors' => [
33             'required' => 'Penerbit harus diisi.',
34             'string' => 'Penerbit harus berupa
string.'
35         ]
36     ],
37     'tahun_terbit' => [
38         'rules' => 'required|integer|greater_than[1800]|less_than[2024]',
39         'errors' => [
40             'required' => 'Tahun terbit harus
diisi.',
41             'integer' => 'Tahun terbit harus
berupa angka.',
42             'greater_than' => 'Tahun terbit harus
lebih besar dari 1800.',
43             'less_than' => 'Tahun terbit harus
lebih kecil dari 2024.'
44         ]
45     ],
46 ];
47
48 protected $skipValidation = false;
49
50 }

```

Tabel 12. Source Code BukuModel.php

```

1  <?php
2
3  namespace App\Models;
4
5  use CodeIgniter\Model;
6
7  class UsersModel extends Model
8  {
9      protected $table = "users";

```

10	protected \$returnType = "object";
11	protected \$useTimestamps = true;
12	protected \$allowedFields = ['username', 'password', 'email'];
13	}

Tabel 13. Source Code UserModel.php

1	<?php
2	
3	namespace App\Database\Migrations;
4	
5	use CodeIgniter\Database\Migration;
6	
7	class Users extends Migration
8	{
9	public function up()
10	{
11	\$this->forge->addField([
12	'id' => [
13	'type' => 'INT',
14	'unsigned' => TRUE,
15	'auto_increment' => TRUE
16],
17	'username' => [
18	'type' => 'VARCHAR',
19	'constraint' => 255,
20	'null' => FALSE,
21],
22	'email' => [
23	'type' => 'VARCHAR',
24	'constraint' => 255,
25	'null' => FALSE,
26],
27	'password' => [
28	'type' => 'TEXT',
29	'null' => FALSE,
30],
31	'created_at' => [
32	'type' => 'datetime',
33	'null' => TRUE
34],
35	'updated_at' => [
36	'type' => 'datetime',
37	'null' => TRUE
38]
39]);
40	
41	\$this->forge->addKey('id', TRUE);

```

42         $this->forge->createTable('users');
43     }
44
45     //-----
46
47     public function down()
48     {
49         $this->forge->dropTable('users');
50     }
51 }

```

Tabel 14. Source Code 2024-06-03-103310_Users.php

```

1  <?php
2
3  namespace App\Database\Migrations;
4
5  use CodeIgniter\Database\Migration;
6
7  class Buku extends Migration
8  {
9      public function up()
10     {
11         $this->forge->addField([
12             'id' => [
13                 'type' => 'BIGINT',
14                 'unsigned' => TRUE,
15                 'auto_increment' => TRUE
16             ],
17             'judul' => [
18                 'type' => 'VARCHAR',
19                 'constraint' => 255,
20                 'null' => FALSE,
21             ],
22             'penulis' => [
23                 'type' => 'VARCHAR',
24                 'constraint' => 255,
25                 'null' => FALSE,
26             ],
27             'penerbit' => [
28                 'type' => 'VARCHAR',
29                 'constraint' => 255,
30                 'null' => FALSE,
31             ],
32             'tahun_terbit' => [
33                 'type' => 'YEAR',
34                 'null' => TRUE
35             ]

```

36]);
37	
38	\$this->forge->addKey('id', TRUE);
39	\$this->forge->createTable('buku');
40	}
41	
42	public function down()
43	{
44	\$this->forge->dropTable('buku');
45	}
46	}

Tabel 15. Source Code 2024-06-03-122804_Buku.php

1	<?php
2	
3	namespace App\Database\Seeds;
4	
5	use CodeIgniter\Database\Seeder;
6	
7	class Buku extends Seeder
8	{
9	public function run()
10	{
11	\$this->db->table('buku')->insert([
12	'judul' => 'Laut Bercerita',
13	'penulis' => 'Leila S. Chudori',
14	'penerbit' => 'Gramedia',
15	'tahun_terbit' => '2017'
16]);
17	
18	\$this->db->table('buku')->insert([
19	'judul' => 'Tentang Kamu',
20	'penulis' => 'Tere Liye',
21	'penerbit' => 'Republika',
22	'tahun_terbit' => '2016'
23]);
24	}
25	}

Tabel 16. Source Code Seeds Buku.php

1	<?php
2	
3	namespace App\Database\Seeds;
4	
5	use CodeIgniter\Database\Seeder;
6	
7	class UserSeeder extends Seeder
8	{

9	public function run()
10	{
11	\$this->db->table('users')->insert([
12	'username' => 'Farly',
13	'email' => 'farlyhaydyl23@gmail.com',
14	'password' => password_hash('123',
15	PASSWORD_DEFAULT),
16]);
17	}

Tabel 17. Source Code UserSeeder.php

1	-- phpMyAdmin SQL Dump
2	-- version 5.2.1
3	-- https://www.phpmyadmin.net/
4	--
5	-- Host: 127.0.0.1
6	-- Waktu pembuatan: 11 Jun 2024 pada 19.57
7	-- Versi server: 10.4.32-MariaDB
8	-- Versi PHP: 8.2.12
9	
10	SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
11	START TRANSACTION;
12	SET time_zone = "+00:00";
13	
14	
15	/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */; SET
16	/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */; SET
17	/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */; SET
18	/*!40101 SET NAMES utf8mb4 */;
19	
20	--
21	-- Database: `perpustakaan`
22	--
23	
24	-- -----
25	---
26	--
27	-- Struktur dari tabel `buku`
28	--
29	
30	CREATE TABLE `buku` (
31	`id` bigint(20) UNSIGNED NOT NULL,
32	`judul` varchar(255) NOT NULL,

```

33 `penulis` varchar(255) NOT NULL,
34 `penerbit` varchar(255) NOT NULL,
35 `tahun_terbit` year(4) DEFAULT NULL
36 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
  COLLATE=utf8mb4_general_ci;
37
38 --
39 -- Dumping data untuk tabel `buku`
40 --
41
42 INSERT INTO `buku` (`id`, `judul`, `penulis`,
43 `penerbit`, `tahun_terbit`) VALUES
44 (15, 'Laut Bercerita', 'Leila S. Chudori', 'Gramedia',
45 '2017'),
46 (16, 'Tentang Kamu', 'Tere Liye', 'Republika', '2016');
47
48 --
49 -- Struktur dari tabel `migrations`
50 --
51
52 CREATE TABLE `migrations` (
53 `id` bigint(20) UNSIGNED NOT NULL,
54 `version` varchar(255) NOT NULL,
55 `class` varchar(255) NOT NULL,
56 `group` varchar(255) NOT NULL,
57 `namespace` varchar(255) NOT NULL,
58 `time` int(11) NOT NULL,
59 `batch` int(11) UNSIGNED NOT NULL
60 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
  COLLATE=utf8mb4_general_ci;
61
62 --
63 -- Dumping data untuk tabel `migrations`
64 --
65
66 INSERT INTO `migrations` (`id`, `version`, `class`,
67 `group`, `namespace`, `time`, `batch`) VALUES
68 (1, '2024-06-03-103310',
69 'App\\Database\\Migrations\\Users', 'default', 'App',
1718081782, 1),
70 (2, '2024-06-03-122804',
71 'App\\Database\\Migrations\\Buku', 'default', 'App',
1718081782, 1),
72 (4, '2024-06-11-173406',

```



```

'App\\Database\\Migrations\\Users', 'default', 'App',
1718127395, 3);
70
71  -- -----
72  ---
73  --
74  -- Struktur dari tabel `users`
75  --
76
77  CREATE TABLE `users` (
78    `id` int(10) UNSIGNED NOT NULL,
79    `username` varchar(255) NOT NULL,
80    `email` varchar(255) NOT NULL,
81    `password` text NOT NULL,
82    `created_at` datetime DEFAULT NULL,
83    `updated_at` datetime DEFAULT NULL
84  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
  COLLATE=utf8mb4_general_ci;
85
86  --
87  -- Dumping data untuk tabel `users`
88  --
89
90  INSERT INTO `users` (`id`, `username`, `email`,
  `password`, `created_at`, `updated_at`) VALUES
91  (4, 'Farly', 'farlyhaydy123@gmail.com',
  '$2y$10$xYhsftGqi132/1Y8Qrypge2VGyR2bS0TEBFpXTa4sXUKI4Gw
  nEoh2', NULL, NULL);
92
93  --
94  -- Indexes for dumped tables
95  --
96
97  --
98  -- Indeks untuk tabel `buku`
99  --
100  ALTER TABLE `buku`
101    ADD PRIMARY KEY (`id`);
102
103  --
104  -- Indeks untuk tabel `migrations`
105  --
106  ALTER TABLE `migrations`
107    ADD PRIMARY KEY (`id`);
108
109  --

```

```

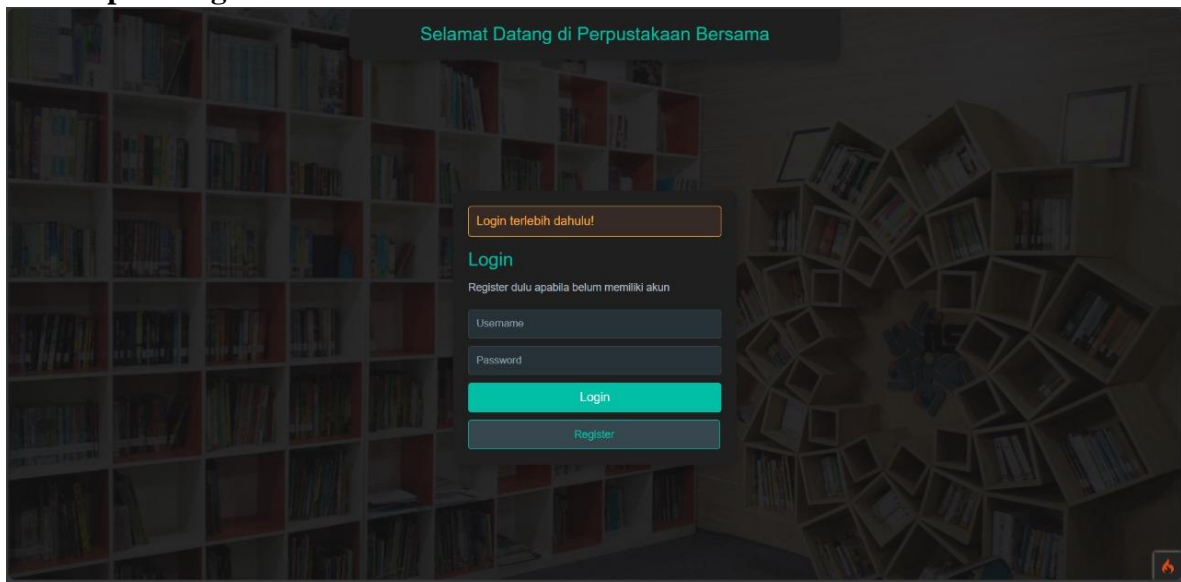
110 -- Indeks untuk tabel `users`
111 --
112 ALTER TABLE `users`
113     ADD PRIMARY KEY (`id`),
114     ADD UNIQUE KEY `email` (`email`);
115
116 --
117 -- AUTO_INCREMENT untuk tabel yang dibuang
118 --
119
120 --
121 -- AUTO_INCREMENT untuk tabel `buku`
122 --
123 ALTER TABLE `buku`
124     MODIFY `id` bigint(20) UNSIGNED NOT NULL
125     AUTO_INCREMENT, AUTO_INCREMENT=19;
126 --
127 -- AUTO_INCREMENT untuk tabel `migrations`
128 --
129 ALTER TABLE `migrations`
130     MODIFY `id` bigint(20) UNSIGNED NOT NULL
131     AUTO_INCREMENT, AUTO_INCREMENT=5;
132 --
133 -- AUTO_INCREMENT untuk tabel `users`
134 --
135 ALTER TABLE `users`
136     MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT,
137     AUTO_INCREMENT=5;
138 COMMIT;
139
140
141

```

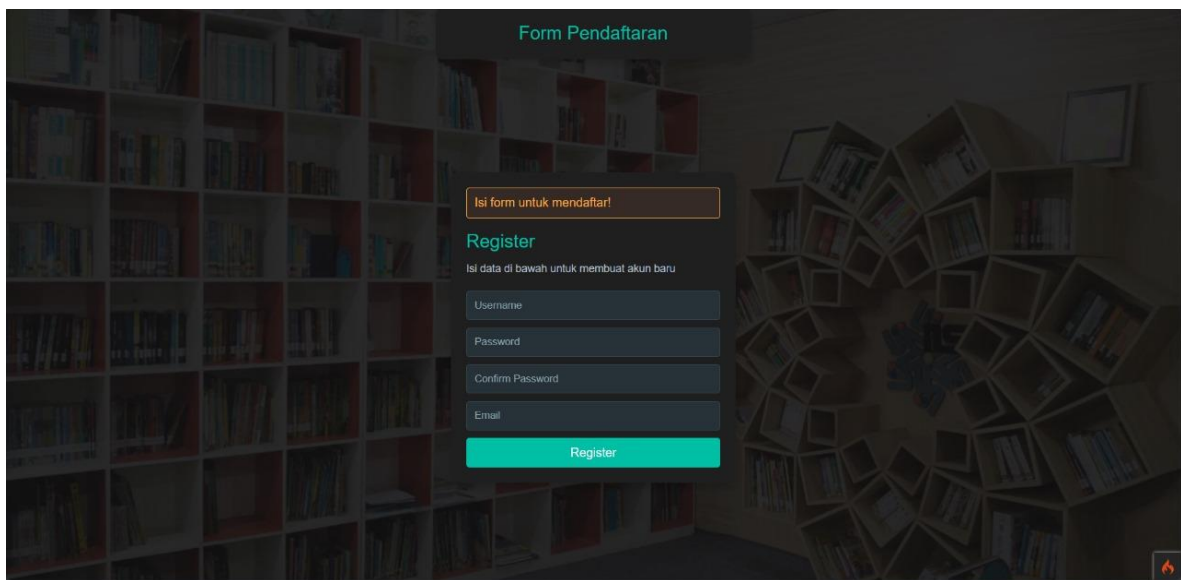
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
 /*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
 /*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

Tabel 18. Source Code perpustakaan.sql

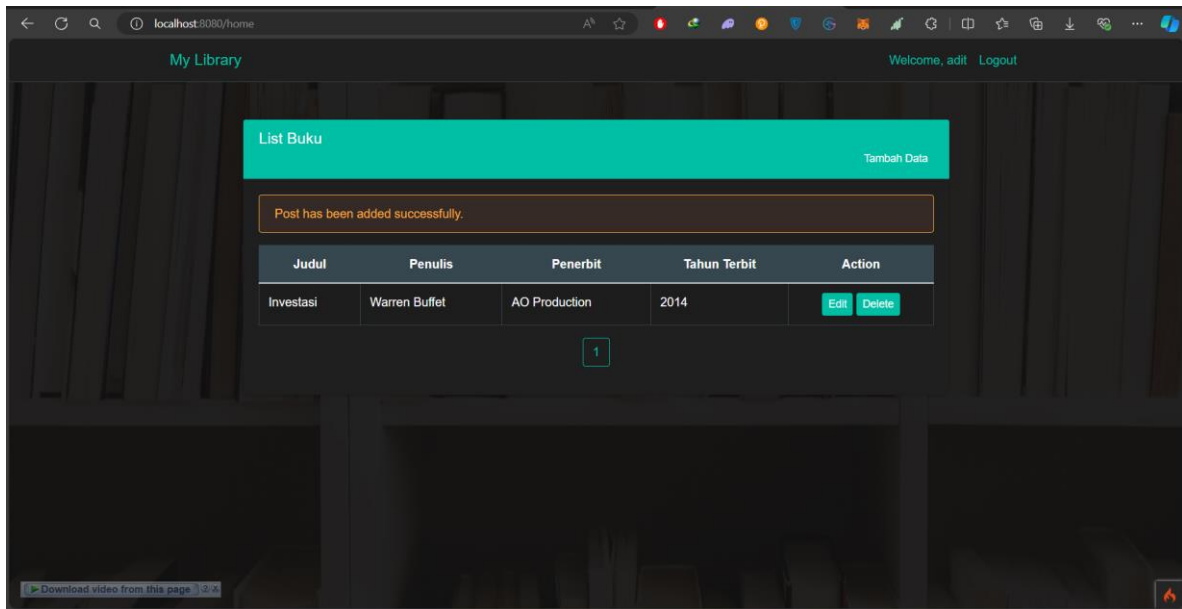
B. Output Program



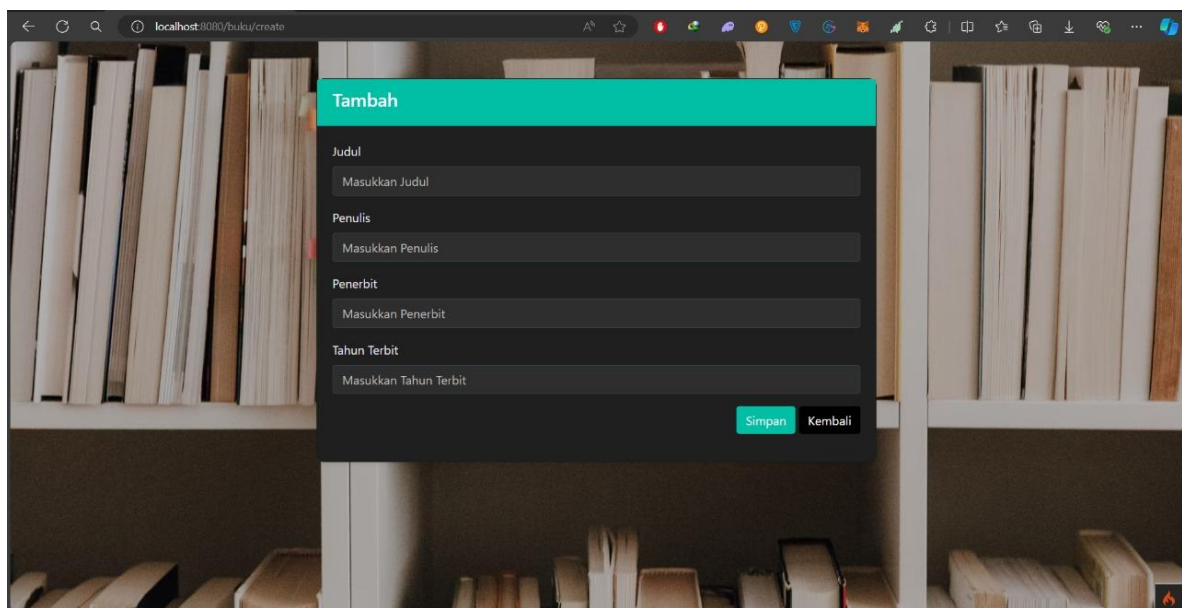
Gambar 1. Halaman Login



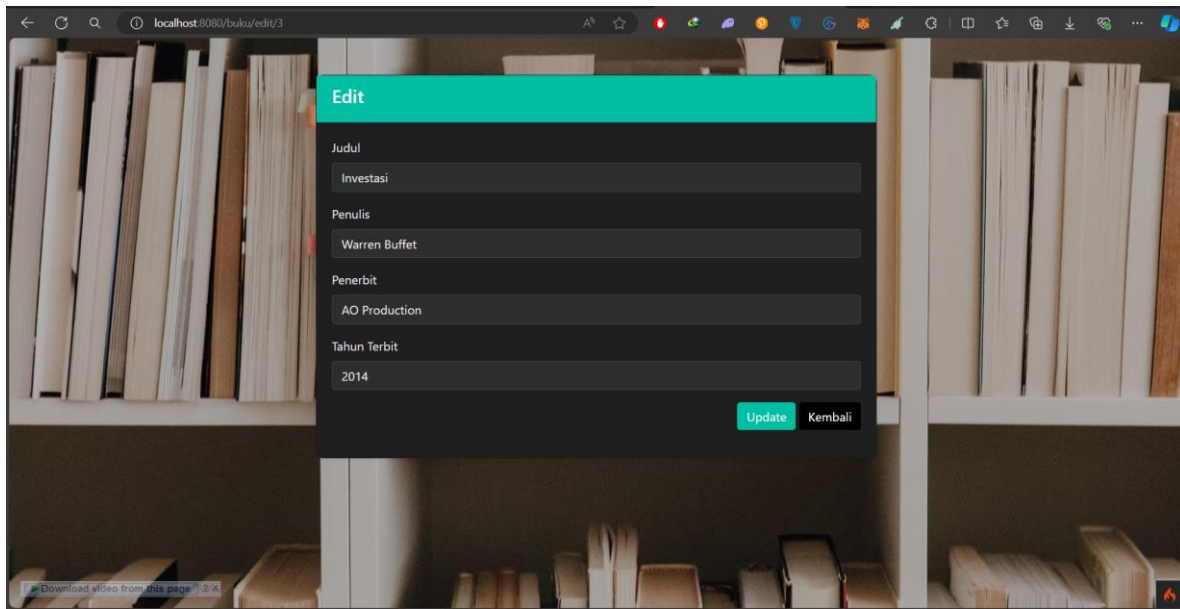
Gambar 2. Halaman Register



Gambar 3. Halaman Home



Gambar 4. Halaman Untuk Menambahkan Data



Gambar 5. Halaman Untuk Edit Data

C. Pembahasan

Routes.php

Kode di atas merupakan konfigurasi routing menggunakan framework CodeIgniter. Pertama, menggunakan namespace CodeIgniter\Router\RouteCollection, di mana \$routes merupakan objek RouteCollection yang digunakan untuk mendefinisikan rute aplikasi. setAutoRoute(true) mengaktifkan pengaturan otomatis untuk mencocokkan rute berdasarkan kontroler dan metode yang disebutkan. Rute-rute yang didefinisikan kemudian adalah sebagai berikut: Rute '/' mengarah ke metode login dari kontroler Buku. Rute 'home' mengarah ke metode index dari kontroler Buku. Rute '/register' mengarah ke metode index dari kontroler Register, dan rute '/register/process' mengarah ke metode process dari kontroler Register, dengan metode HTTP POST. Rute '/login' mengarah ke metode index dari kontroler Login, dan rute '/login/process' mengarah ke metode process dari kontroler Login, dengan metode HTTP POST. Rute '/logout' mengarah ke metode logout dari kontroler Login. Rute '/buku/create' mengarah ke metode create dari kontroler Buku, dan rute '/buku/store' mengarah ke metode store dari kontroler Buku, dengan metode HTTP POST. Terakhir, rute '/buku/update/', '/buku/edit/', dan '/buku/delete/' masing-masing mengarah ke metode update, edit, dan delete dari kontroler Buku, dengan pengecualian filter auth yang menunjukkan bahwa akses ke rute-rute tersebut memerlukan autentikasi pengguna.

Buku.php

Kode kontroler Buku dalam aplikasi CodeIgniter mengelola operasi CRUD untuk entitas buku. Dalam konstruktor, BukuModel diinisialisasi untuk interaksi dengan database, dan helper form serta url dimuat untuk keperluan pembuatan formulir dan pengelolaan URL.

Metode seperti login(), index(), create(), store(), edit(\$id), update(\$id), dan delete(\$id) memfasilitasi tampilan dan manipulasi data buku, dengan memvalidasi input sebelum menyimpan atau memperbarui data, serta memastikan akses terbatas pada pengguna yang sudah login. Pesan sukses atau gagal disertakan dalam setiap operasi untuk memberikan umpan balik kepada pengguna terkait hasil operasi CRUD yang dilakukan.

Home.php

Kode di atas adalah kontroler Home dalam aplikasi CodeIgniter. Kontroler ini merupakan turunan dari kelas BaseController, yang memungkinkannya untuk mengakses fitur dan metode dasar dari framework CodeIgniter. Dalam metode index(), kontroler ini mengembalikan tampilan dengan nama vw_buku, yang kemungkinan besar merupakan halaman utama atau halaman yang menampilkan daftar buku dalam aplikasi tersebut. Dengan mengembalikan tampilan ini, pengguna dapat melihat konten yang sesuai dengan fungsi dari aplikasi buku yang sedang dibangun.

Login.php

Kode kontroler Login dalam aplikasi CodeIgniter mengelola proses autentikasi pengguna. Metode index() memungkinkan pengguna untuk melihat halaman login (vw_login). Metode process() memvalidasi username dan password yang diterima dari formulir login, menggunakan UsersModel untuk mencari pengguna berdasarkan username, dan memverifikasi password dengan fungsi password_verify(). Jika autentikasi berhasil, session pengguna diset dengan informasi username dan status logged_in, dan pengguna diarahkan ke halaman utama (home). Jika gagal, pesan kesalahan ditampilkan dan pengguna diarahkan kembali ke halaman login. Metode logout() menghapus semua session dan mengarahkan pengguna kembali ke halaman login, memastikan proses logout berjalan dengan baik. Ini memastikan bahwa aplikasi hanya memberikan akses ke konten yang sesuai dengan pengguna yang telah terautentikasi dengan benar.

Register.php

Kode kontroler Register dalam aplikasi CodeIgniter bertanggung jawab untuk proses pendaftaran pengguna baru. Metode index() menampilkan halaman registrasi (vw_register), sementara metode process() melakukan validasi terhadap input yang diterima dari formulir registrasi. Validasi mencakup aturan untuk username yang harus unik dan memiliki panjang minimal 4 karakter serta maksimal 20 karakter, password dengan panjang minimal 8 karakter dan maksimal 50 karakter, konfirmasi password yang harus sesuai dengan password, dan email yang harus valid. Jika validasi gagal, pesan kesalahan ditampilkan dan pengguna diarahkan kembali ke halaman registrasi dengan input sebelumnya dipertahankan. Jika berhasil, password di-hash menggunakan password_hash() sebelum disimpan ke database melalui model UsersModel, dan pengguna diarahkan ke halaman login. Kontroler ini memastikan bahwa proses registrasi aman dan sesuai dengan aturan

validasi yang ditetapkan, serta mengamankan password dengan penggunaan hash sebelum penyimpanan ke database.

AuthFilter.php

Kode filter AuthFilter yang terletak di `app/Filters/AuthFilter.php` dalam aplikasi CodeIgniter bertujuan untuk memastikan keamanan akses dengan memverifikasi apakah pengguna telah login sebelum mengizinkannya mengakses halaman tertentu. Dalam metode `before()`, filter ini mengecek keberadaan kunci `'user_id'` dalam sesi. Jika tidak ada, pengguna diarahkan kembali ke halaman login menggunakan `redirect()->to('/login')`. Hal ini memastikan bahwa hanya pengguna yang sudah terautentikasi yang dapat mengakses halaman yang dilindungi oleh filter ini, memberikan lapisan keamanan tambahan untuk aplikasi dengan memastikan akses yang sesuai dengan izin pengguna.

vw_add.php

Kode di atas merupakan sebuah template halaman untuk menambah data buku dalam aplikasi CodeIgniter, yang diperluas (`extend`) dari layout utama yang disebut `'layouts/main'`. Halaman ini memiliki gaya yang ditentukan secara internal dengan latar belakang gambar dari Unsplash, serta pengaturan gaya untuk elemen-elemen seperti kartu (`card`), tombol (`button`), dan input form. Pada bagian konten (`content`), terdapat sebuah form untuk menambahkan buku baru, dengan validasi pesan kesalahan yang ditampilkan jika ada, menggunakan `session()->getFlashdata('error')`. Form ini mengirimkan data ke rute `buku/store` melalui `form_open()` dari CodeIgniter. Tombol "Simpan" akan mengirimkan data form, sementara tombol "Kembali" akan mengarahkan pengguna kembali ke halaman utama (`home`). Bagian JavaScript ekstra (`extra-js`) memuat library Summernote untuk menyediakan editor teks yang memungkinkan pengguna untuk memasukkan konten dengan lebih interaktif. Ini termasuk pengaturan untuk menginisialisasi Summernote pada elemen dengan `id post_content`. Dengan ini, halaman ini menyediakan antarmuka yang nyaman dan estetis untuk menambahkan informasi buku dalam aplikasi.

vw_buku.php

Kode di atas adalah sebuah halaman dalam aplikasi web CodeIgniter yang menampilkan daftar buku. Halaman ini diperluas (`extend`) dari layout utama yang disebut `'layouts/main'`. Desain halaman menggunakan CSS untuk menetapkan gaya tata letak dan penampilan elemen-elemen seperti navbar, kartu (`card`), tombol (`button`), dan tabel. Latar belakang halaman menggunakan gambar dari Unsplash dengan efek opasitas yang ditambahkan untuk mempertahankan fokus pada konten utama. Navbar di bagian atas halaman menampilkan menu navigasi yang berubah berdasarkan status login pengguna, menampilkan tautan untuk login, registrasi, atau tautan selamat datang dan logout jika pengguna sudah login. Bagian konten halaman berisi daftar buku dalam tabel yang menampilkan judul, penulis, penerbit, dan tahun terbit dari setiap buku. Setiap baris tabel

memiliki tombol untuk mengedit dan menghapus buku yang memicu konfirmasi sebelum aksi penghapusan dilakukan. Di bagian bawah tabel, terdapat navigasi halaman (pagination) untuk memudahkan navigasi antara halaman-halaman data. JavaScript tambahan digunakan untuk menambahkan kelas CSS ke elemen-elemen navigasi halaman untuk memperbaiki tampilan paginasi. Dengan ini, halaman menyediakan antarmuka yang responsif dan estetik untuk manajemen data buku dalam aplikasi.

vw_edit.php

Kode di atas merupakan halaman untuk mengedit entri buku dalam aplikasi web menggunakan CodeIgniter. Pertama, halaman ini diperluas (extend) dari layout 'layouts/main' yang menentukan struktur umum dan gaya halaman. Di dalam <style> tag, pengaturan tampilan background, warna teks, dan gaya kartu (card) diatur untuk menciptakan tampilan yang estetik. Dalam bagian konten, sebuah form edit dibuat menggunakan <?= form_open('buku/update/' . \$post['id']); ?> untuk mengirimkan data update ke fungsi update dalam controller. Setiap input form seperti judul, penulis, penerbit, dan tahun terbit sudah diisi dengan nilai yang sudah ada (\$post['judul'], \$post['penulis'], \$post['penerbit'], dan \$post['tahun_terbit']) untuk memastikan informasi yang ada tidak hilang saat pengeditan. Tombol "Update" akan mengirimkan form untuk memperbarui entri buku, sementara tombol "Kembali" akan mengarahkan pengguna kembali ke halaman utama. Di bagian extra-js, digunakan plugin Summernote untuk memperluas kemampuan input teks dengan menambahkan editor WYSIWYG, memungkinkan pengguna untuk mengedit konten dengan lebih mudah. Dengan demikian, halaman ini tidak hanya berfungsi untuk mengedit data buku tetapi juga memperkaya pengalaman pengguna dengan tambahan fitur editor teks yang responsif.

vw_login.php

Kode HTML di atas adalah halaman login untuk aplikasi perpustakaan, dengan pengaturan tampilan dan interaksi yang responsif. Di dalam <style> tag, elemen <body> ditata agar konten berada di tengah layar dengan latar belakang gambar yang diberikan efek opacity untuk menciptakan nuansa visual yang menarik. Dua div dengan class .welcome-message dan .form-signin masing-masing menampilkan pesan selamat datang dan form login. Pesan alert awal di #initial-alert memberitahukan pengguna untuk login terlebih dahulu dan diatur untuk tampil secara default. Alert kedua menampilkan pesan error jika login gagal dengan menggunakan session Flashdata dari framework CodeIgniter. Form login meminta username dan password dengan desain input yang konsisten dan tombol "Login" yang mengeksekusi proses login. Terdapat juga tombol "Register" yang mengarahkan pengguna ke halaman registrasi untuk membuat akun baru jika belum terdaftar. Halaman ini dirancang agar mudah diakses, informatif, dan responsif terhadap perangkat yang digunakan pengguna.

vw_register.php

Kode HTML di atas adalah halaman registrasi untuk aplikasi perpustakaan, dengan tampilan yang dirancang agar responsif dan menarik. Pada bagian `<style>`, elemen `<body>` dan `<html>` diatur agar konten berada di tengah layar dengan latar belakang gambar yang memiliki efek opacity untuk menambahkan nuansa visual. Dua div dengan class `.welcome-message` dan `.form-register` masing-masing menampilkan pesan selamat datang dan form registrasi. Pesan alert awal di `#initial-alert` memberitahukan pengguna untuk mengisi form pendaftaran, dan diatur untuk tampil secara default. Alert kedua menampilkan pesan error jika proses registrasi gagal dengan menggunakan session Flashdata dari framework CodeIgniter. Form registrasi meminta username, password (dengan konfirmasi), dan alamat email dengan desain input yang konsisten dan tombol "Register" yang mengeksekusi proses registrasi. Halaman ini dirancang agar mudah digunakan, informatif, serta dapat menangani interaksi pengguna dengan baik.

BukuModel.php

Kelas `BukuModel` merupakan model yang digunakan dalam aplikasi berbasis CodeIgniter untuk berinteraksi dengan tabel 'buku' di database. Pada baris pertama, namespace `App\Models` menunjukkan lokasi file kelas ini di dalam struktur direktori aplikasi. Kelas ini meng-extend dari `CodeIgniter\Model`, yang menyediakan berbagai fitur untuk pengelolaan data. Properti `$table` menentukan nama tabel yang akan diakses, yaitu 'buku'. `$allowedFields` mengizinkan kolom mana saja yang dapat diisi dengan data menggunakan model ini, dalam hal ini adalah 'judul', 'penulis', 'penerbit', dan 'tahun_terbit'. Properti `$createdField`, `$updatedField`, dan `$deletedField` menentukan nama kolom untuk penanda waktu pembuatan, pembaruan, dan penghapusan data. Properti `$validationRules` mendefinisikan aturan validasi untuk setiap kolom, seperti 'judul' harus diisi dan berupa string, 'tahun_terbit' harus berupa angka antara 1800 dan 2024, dan seterusnya. `$skipValidation` diatur false, sehingga validasi data akan diterapkan secara default. Dengan struktur ini, `BukuModel` memfasilitasi operasi CRUD (Create, Read, Update, Delete) dengan pengelolaan validasi data yang ketat sesuai dengan aturan yang telah ditentukan.

UserModel.php

Kelas `UsersModel` merupakan model yang digunakan dalam aplikasi berbasis CodeIgniter untuk berinteraksi dengan tabel 'users' di database. Pada baris pertama, namespace `App\Models` menunjukkan lokasi file kelas ini di dalam struktur direktori aplikasi. Kelas ini meng-extend dari `CodeIgniter\Model`, yang menyediakan berbagai fitur untuk pengelolaan data. Properti `$table` menentukan nama tabel yang akan diakses, yaitu 'users'. Properti `$returnType` diatur sebagai "object", yang menandakan bahwa data hasil query akan dikembalikan dalam bentuk objek. `$useTimestamps` diatur true, sehingga secara otomatis akan menggunakan kolom timestamps untuk menyimpan waktu pembuatan dan pembaruan data. Properti `$allowedFields` mengizinkan kolom mana saja yang dapat diisi dengan data menggunakan model ini, dalam hal ini adalah 'username', 'password', dan 'email'. Dengan

struktur ini, UsersModel memfasilitasi operasi CRUD (Create, Read, Update, Delete) pada tabel 'users' dengan pengelolaan waktu dan pengisian data yang terkontrol sesuai dengan aturan yang telah ditentukan.

2024-06-03-103310_Users.php

Kelas Users adalah migration untuk mengelola struktur tabel 'users' dalam database menggunakan CodeIgniter. Metode up() digunakan untuk mendefinisikan struktur tabel dengan menggunakan \$this->forge->addField(...) untuk menentukan kolom-kolom seperti 'id' sebagai INT dengan auto increment, 'username' sebagai VARCHAR dengan panjang maksimum 255 karakter, 'email' sebagai VARCHAR dengan panjang maksimum 255 karakter, 'password' sebagai TEXT, serta 'created_at' dan 'updated_at' sebagai datetime yang dapat bernilai NULL. Setelah menentukan struktur, addKey('id', TRUE) digunakan untuk menambahkan kunci utama pada kolom 'id', dan createTable('users') untuk membuat tabel 'users' berdasarkan definisi yang telah disiapkan. Metode down() akan menjatuhkan (drop) tabel 'users' jika migration tersebut di-rollback atau dihapus. Migration ini memfasilitasi pengelolaan struktur database secara konsisten dan mudah diubah melalui penggunaan CodeIgniter's Migration class.

2024-06-03-122804_Buku.php

Kelas Buku adalah migration untuk mengelola struktur tabel 'buku' dalam database menggunakan CodeIgniter. Pada metode up(), addField(...) digunakan untuk menentukan kolom-kolom seperti 'id' dengan tipe BIGINT yang auto increment, 'judul', 'penulis', dan 'penerbit' sebagai VARCHAR dengan panjang maksimum 255 karakter, serta 'tahun_terbit' sebagai YEAR yang dapat bernilai NULL. Setelah menentukan struktur kolom, addKey('id', TRUE) digunakan untuk menambahkan kunci utama pada kolom 'id', dan createTable('buku') untuk membuat tabel 'buku' berdasarkan definisi tersebut. Metode down() akan menjatuhkan (drop) tabel 'buku' jika migration ini di-rollback atau dihapus. Migration ini memudahkan pengelolaan struktur database dalam aplikasi CodeIgniter, memastikan konsistensi dan fleksibilitas dalam pengaturan skema database.

Seeds Buku.php

Kelas Buku adalah sebuah seeder dalam framework CodeIgniter yang digunakan untuk mengisi data awal ke tabel 'buku' dalam database. Pada metode run(), digunakan \$this->db->table('buku')->insert(...) untuk menyisipkan dua baris data ke tabel 'buku'. Baris pertama memasukkan judul 'Laut Bercerita' oleh Leila S. Chudori, diterbitkan oleh Gramedia pada tahun 2017. Baris kedua memasukkan judul 'Tentang Kamu' oleh Tere Liye, diterbitkan oleh Republika pada tahun 2016. Seeder ini membantu dalam menginisialisasi data penting secara otomatis untuk pengujian atau pengembangan aplikasi, mempercepat proses pengisian data pada tahap awal pengembangan.

UserSeeder.php

Kelas UserSeeder merupakan sebuah seeder dalam framework CodeIgniter yang bertanggung jawab untuk menambahkan satu baris data ke tabel 'users' dalam database. Pada metode run(), terdapat satu perintah `$this->db->table('users')->insert([...])` yang memasukkan data pengguna dengan username 'Adit', email 'aditya.oktaviari@gmail.com', dan password yang telah di-hash menggunakan fungsi password_hash() dengan algoritma default PASSWORD_DEFAULT. Seeder ini digunakan untuk menyediakan data awal pengguna pada aplikasi, memungkinkan pengembang untuk melakukan pengujian dan konfigurasi awal dengan lebih mudah.

perpustakaan.sql

Ini adalah dump SQL untuk database perpustakaan yang mencakup struktur tabel dan data yang terkait dengan buku dan pengguna. Dump dimulai dengan pengaturan SQL_MODE, timezone, dan struktur database yang disesuaikan dengan pengaturan InnoDB dan charset utf8mb4.

Tabel buku memiliki kolom id sebagai primary key dengan tipe data BIGINT UNSIGNED yang diatur untuk auto increment. Tabel ini juga memuat kolom seperti judul, penulis, penerbit, dan tahun_terbit yang digunakan untuk menyimpan informasi terkait buku, seperti judul buku, nama penulis, penerbit, dan tahun terbit.

Tabel migrations digunakan untuk melacak versi migrasi yang telah dijalankan dengan detail seperti id, version, class, group, namespace, time, dan batch.

Tabel users berisi informasi pengguna dengan kolom seperti id sebagai primary key dengan tipe data INT UNSIGNED, username, email, password (yang di-hash untuk keamanan), serta created_at dan updated_at untuk mencatat waktu pembuatan dan pembaruan data pengguna.

Setiap tabel dilengkapi dengan indeks yang sesuai untuk mempercepat pencarian data dan konfigurasi AUTO_INCREMENT untuk mengelola penambahan otomatis pada kolom primary key. Data awal yang dimasukkan ke dalam tabel buku mencakup informasi tentang buku dengan judul "Investasi" yang ditulis oleh Warren Buffet dan diterbitkan oleh AO Production pada tahun 2014. Tabel users juga memiliki data awal dengan informasi pengguna dengan username 'adit', email 'aditya.oktaviari@gmail.com', dan password yang telah di-hash.

Dump ini menyediakan struktur dasar dan data awal yang siap digunakan untuk aplikasi perpustakaan, menggunakan framework atau aplikasi yang mendukung PHP dan database MariaDB/MySQL.

D. Tautan Git

Berikut adalah tautan untuk source code yang telah dibuat.

[Pemrograman-Web-2/Modul6 at main · aditokta/Pemrograman-Web-2 \(github.com\)](https://github.com/aditokta/Pemrograman-Web-2)