

PEMROGRAMAN BERBASIS WEB – OOP, PDO, dan MVC

Untuk Memenuhi Tugas Praktikum Pemrograman Berbasis Web



Oleh :

Nama : Afandi Ikhsyan Al Karim

NPM : 4522210032

Dosen :

Adi Wahyu Pribadi , S.Si., M.Kom

S1- Teknik Informatika

Universitas Pancasila

2023/2024

Tujuan Praktikum

- Implementasi prinsip OOP dalam PHP
- Menggunakan PDO untuk koneksi dan manipulasi data di MySQL
- Menerapkan konsep MVC
- Membuat fitur login admin untuk mengedit dan menghapus komentar dari pengguna

Langkah-Langkah

1. Buat database baru dengan nama buku_tamu

```
CREATE DATABASE buku_tamu;
```

2. Pilih db buku_tamu

```
USE buku_tamu;
```

3. Buat table guest untuk menyimpan data buku tamu

```
CREATE TABLE guests (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(100) NOT NULL,  
    comment TEXT NOT NULL,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

4. Buat table admins utk data login admin

```
CREATE TABLE admins (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    username VARCHAR(50) NOT NULL,  
    password VARCHAR(255) NOT NULL  
);
```

5. Insert user admin dan passwordnya yaitu 'admin123' namun kita masukkan dahulu ke fungsi hash sehingga hasilnya adalah

```
INSERT INTO admins (username, password) VALUES ('admin',  
'$2y$10$e0NRz6iVb6q5VhB8GzJ4JeQc2YJwZgQ8hGgFXc7V3K6i5J9H1  
K8uK');
```

Server: 127.0.0.1 » Basis data: buku_tamu

Struktur SQL Cari Kueri Ekspor Impor Operasi Hak Akses Routine Event

Filters

Mengandung kata:

Tabel	Tindakan	Baris	Tipe	Penyortiran	Ukuran	Beban
<input type="checkbox"/> admins	Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	1	InnoDB	latin1_swedish_ci	16 KB	-
<input type="checkbox"/> guests	Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	latin1_swedish_ci	16 KB	-
2 tabel	Jumlah	1	InnoDB	latin1_swedish_ci	32 KB	0 B

Struktur Direktori Proyek

/buku-tamu/

```

|
|— index.php
|— admin.php
|— login.php
|
|— /config/
|   |— Database.php
|
|— /controllers/
|   |— GuestController.php
|   |— AdminController.php
|
|— /models/
|   |— Guest.php
|   |— Admin.php
|
|— /views/
|   |— guest_form.php
|   |— guest_list.php
|   |— admin_login.php
|   |— admin_dashboard.php

```

Konfigurasi Koneksi Database

1. /config/Database.php

```
Database.php X
buku-tamu > config > Database.php
1  <?php
2  class Database {
3      private $host = "localhost";
4      private $db_name = "buku_tamu";
5      private $username = "root";
6      private $password = "";
7      public $conn;
8
9      public function getConnection() {
10         $this->conn = null;
11         try {
12             $this->conn = new PDO("mysql:host=" . $this->host . ";dbname=" . $this->db_name,
13                 $this->username, $this->password);
14             $this->conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
15         } catch(PDOException $exception) {
16             echo "Koneksi error: " . $exception->getMessage();
17         }
18         return $this->conn;
19     }
20 }
21
```

2. /controllers/AdminController.php

```
AdminController.php X
buku-tamu > controllers > AdminController.php
1  <?php
2  include_once './config/Database.php';
3  include_once './models/Admin.php';
4
5  class AdminController {
6      private $admin;
7
8      public function __construct() {
9          $database = new Database();
10         $db = $database->getConnection();
11         $this->admin = new Admin($db);
12     }
13
14     // Fungsi untuk login
15     public function login($username, $password) {
16         $this->admin->username = $username;
17         $this->admin->password = $password;
18         return $this->admin->login();
19     }
20 }
21
```

3. /controllers/GuestController.php

```
GuestController.php X
buku-tamu > controllers > GuestController.php
1  <?php
2  include_once './config/Database.php';
3  include_once './models/Guest.php';
4
5  class GuestController {
6      private $guest;
7
8      public function __construct() {
9          $database = new Database();
10         $db = $database->getConnection();
11         $this->guest = new Guest($db);
12     }
13
14     // Fungsi untuk menambahkan tamu baru
15     public function create($name, $comment) {
16         $this->guest->name = $name;
17         $this->guest->comment = $comment;
18         return $this->guest->create();
19     }
20
21     // Fungsi untuk mendapatkan semua data tamu
22     public function readAll() {
23         return $this->guest->readAll();
24     }
25
26     // Fungsi untuk menghapus tamu
27     public function delete($id) {
28         $this->guest->id = $id;
29         return $this->guest->delete();
30     }
31
32     // Fungsi untuk mengupdate tamu
33     public function update($id, $name, $comment) {
34         $this->guest->id = $id;
35         $this->guest->name = $name;
36         $this->guest->comment = $comment;
37         return $this->guest->update();
38     }
39 }
40 <?>
```

4. /models/Admin.php

```
Admin.php X
buku-tamu > models > Admin.php
1  <?php
2  session_start();
3  if (!isset($_SESSION['admin'])) {
4      header("Location: login.php");
5      exit();
6  }
7
8  require_once 'controllers/GuestController.php';
9
10 $guestController = new GuestController();
11
12 if (isset($_GET['delete_id'])) {
13     $id = $_GET['delete_id'];
14     $guestController->delete($id);
15     header("Location: admin.php");
16 }
17
18 $guests = $guestController->readAll();
19
20 include 'views/admin_dashboard.php';
21 <?>
```

5. /models/Guest.php

```
Guest.php X
buku-tamu > models > Guest.php
1  <?php
2  class Guest {
3      private $conn;
4      private $table_name = "guests";
5
6      public $id;
7      public $name;
8      public $comment;
9      public $created_at;
10
11     public function __construct($db) {
12         $this->conn = $db;
13     }
14
15     // Menambahkan data tamu baru
16     public function create() {
17         $query = "INSERT INTO " . $this->table_name . " SET name=:name, comment=:comment";
18         $stmt = $this->conn->prepare($query);
19
20         // Bersihkan data
21         $this->name = htmlspecialchars(strip_tags($this->name));
22         $this->comment = htmlspecialchars(strip_tags($this->comment));
23
24         // Bind data
25         $stmt->bindParam(":name", $this->name);
26         $stmt->bindParam(":comment", $this->comment);
27
28         if($stmt->execute()) {
29             return true;
30         }
31         return false;
32     }
33
34     // Mengambil semua data tamu
35     public function readAll() {
36
37         $query = "SELECT * FROM " . $this->table_name . " ORDER BY created_at DESC";
38         $stmt = $this->conn->prepare($query);
39         $stmt->execute();
40         return $stmt;
41     }
42
43     // Menghapus data tamu
44     public function delete() {
45         $query = "DELETE FROM " . $this->table_name . " WHERE id = ?";
46         $stmt = $this->conn->prepare($query);
47         $stmt->bindParam(1, $this->id);
48
49         if($stmt->execute()) {
50             return true;
51         }
52         return false;
53     }
54
55     // Mengupdate data tamu
56     public function update() {
57         $query = "UPDATE " . $this->table_name . " SET name=:name, comment=:comment WHERE id=:id";
58         $stmt = $this->conn->prepare($query);
59
60         // Bersihkan data
61         $this->name = htmlspecialchars(strip_tags($this->name));
62         $this->comment = htmlspecialchars(strip_tags($this->comment));
63         $this->id = htmlspecialchars(strip_tags($this->id));
64
65         // Bind data
66         $stmt->bindParam(":name", $this->name);
67         $stmt->bindParam(":comment", $this->comment);
68         $stmt->bindParam(":id", $this->id);
69
70         if($stmt->execute()) {
71             return true;
72         }
73         return false;
74     }
75 }
```

6. /views /admin_dashboard.php

```
admin_dashboard.php X
buku-tamu > views > admin_dashboard.php
1  <h2>Dashboard Admin</h2>
2  <p>Selamat datang, Admin!</p>
3  <a href="logout.php">Logout</a>
4  <table border="1" cellpadding="5" cellspacing="0">
5      <tr>
6          <th>Nama</th>
7          <th>Komentar</th>
8          <th>Waktu</th>
9          <th>Aksi</th>
10     </tr>
11     <?php while ($row = $guests->fetch(PDO::FETCH_ASSOC)) { ?>
12     <tr>
13         <td><?php echo htmlspecialchars($row['name']); ?></td>
14         <td><?php echo htmlspecialchars($row['comment']); ?></td>
15         <td><?php echo $row['created_at']; ?></td>
16         <td>
17             <a href="edit.php?id=<?php echo $row['id']; ?>">Edit</a> |
18             <a href="admin.php?delete_id=<?php echo $row['id']; ?>" onclick="return confirm('Yakin ingin menghapus?')">Hapus</a>
19         </td>
20     </tr>
21     <?php } ?>
22 </table>
```

7. /views /admin_login.php

```
admin_login.php X
buku-tamu > views > admin_login.php
1  <h2>Login Admin</h2>
2  <form action="login.php" method="post">
3      <label for="username">Username:</label><br>
4      <input type="text" name="username" required><br><br>
5      <label for="password">Password:</label><br>
6      <input type="password" name="password" required><br><br>
7      <input type="submit" value="Login">
8  </form>
```

8. /views /guest_form.php

```
guest_form.php X
buku-tamu > views > guest_form.php
1  <h2>Selamat Datang di Buku Tamu</h2>
2  <form action="index.php" method="post">
3      <label for="name">Nama:</label><br>
4      <input type="text" name="name" required><br><br>
5      <label for="comment">Komentar:</label><br>
6      <textarea name="comment" rows="5" required></textarea><br><br>
7      <input type="submit" value="Kirim">
8  </form>
9  <div id="style">
10     body {
11         font-family: Arial, sans-serif;
12         background-color: #f4f4f4;
13         margin: 0;
14         padding: 0;
15     }
16
17     h2, h3 {
18         text-align: center;
19         color: #333;
20     }
21
22     form {
23         width: 50%;
24         margin: 0 auto;
25         background-color: #fff;
26         padding: 20px;
27         border-radius: 10px;
28         box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
29     }
30
31     label {
32         font-weight: bold;
33         color: #555;
34     }
35
36     input[type="text"], textarea {
37         width: 100%;
38         padding: 10px;
39         margin: 8px 0;
40         border: 1px solid #ddd;
41         border-radius: 5px;
42
43         font-size: 16px;
44     }
45
46     input[type="submit"] {
47         width: 100%;
48         padding: 12px;
49         background-color: #4CAF50;
50         color: white;
51         border: none;
52         border-radius: 5px;
53         font-size: 18px;
54         cursor: pointer;
55     }
56
57     input[type="submit"]:hover {
58         background-color: #45a049;
59     }
60
61     table {
62         width: 80%;
63         margin: 20px auto;
64         border-collapse: collapse;
65         background-color: #fff;
66         box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
67     }
68
69     th, td {
70         padding: 12px;
71         text-align: left;
72         border-bottom: 1px solid #ddd;
73     }
74
75     th {
76         background-color: #4CAF50;
77         color: white;
78     }
79
80     tr:hover {
81         background-color: #f2f2f2;
82     }
83 </style>
```

9. /views /guest_list.php

```
guest_list.php X
buku-tamu > views > guest_list.php
1  <h3>Daftar Komentar</h3>
2  <table border="1" cellpadding="5" cellspacing="0">
3      <tr>
4          <th>Nama</th>
5          <th>Komentar</th>
6          <th>Waktu</th>
7      </tr>
8      <?php while ($row = $guests->fetch(PDO::FETCH_ASSOC)) { ?>
9          <tr>
10             <td><?php echo htmlspecialchars($row['name']); ?></td>
11             <td><?php echo htmlspecialchars($row['comment']); ?></td>
12             <td><?php echo $row['created_at']; ?></td>
13          </tr>
14      <?php } ?>
15 </table>
16
17 <style>
18 body {
19     font-family: Arial, sans-serif;
20     background-color: #f4f4f4;
21     margin: 0;
22     padding: 0;
23 }
24
25 h2, h3 {
26     text-align: center;
27     color: #333;
28 }
29
30 form {
31     width: 50%;
32     margin: 0 auto;
33     background-color: #fff;
34     padding: 20px;
35     border-radius: 10px;
36     box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
37 }
38
39 label {
40     font-weight: bold;
41     color: #555;
42 }
43
44 input[type="text"], textarea {
45     width: 100%;
46     padding: 10px;
47     margin: 8px 0;
48     border: 1px solid #ddd;
49     border-radius: 5px;
50     font-size: 16px;
51 }
52
53 input[type="submit"] {
54     width: 100%;
55     padding: 12px;
56     background-color: #4CAF50;
57     color: white;
58     border: none;
59     border-radius: 5px;
60     font-size: 18px;
61     cursor: pointer;
62 }
63
64 input[type="submit"]:hover {
65     background-color: #45a049;
66 }
67
68 table {
69     width: 80%;
70     margin: 20px auto;
71     border-collapse: collapse;
72     background-color: #fff;
73     box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
74 }
75
76 th, td {
77     padding: 12px;
78     text-align: left;
79     border-bottom: 1px solid #ddd;
80 }
81
82 th {
83     background-color: #4CAF50;
84     color: white;
85 }
86
87 tr:hover {
88     background-color: #f2f2f2;
89 }
90
91 </style>
```


10. /admin.php

```
admin.php X
buku-tamu > admin.php
1  <?php
2  class Admin {
3      private $conn;
4      private $table_name = "admins";
5
6      public $id;
7      public $username;
8      public $password;
9
10     public function __construct($db) {
11         $this->conn = $db;
12     }
13
14     // Fungsi login admin
15     public function login() {
16         $query = "SELECT * FROM " . $this->table_name . " WHERE username = :username LIMIT 0,1";
17         $stmt = $this->conn->prepare($query);
18
19         // Bersihkan data
20         $this->username = htmlspecialchars(strip_tags($this->username));
21
22         // Bind data
23         $stmt->bindParam(":username", $this->username);
24         $stmt->execute();
25
26         $row = $stmt->fetch(PDO::FETCH_ASSOC);
27
28         if($row) {
29             if(password_verify($this->password, $row['password'])) {
30                 $this->id = $row['id'];
31                 return true;
32             }
33         }
34         return false;
35     }
36 }
37 >
```

11. /edit.php

```
edit.php X
buku-tamu > edit.php
1  <?php
2  session_start();
3  if (!isset($_SESSION['admin'])) {
4      header("Location: login.php");
5      exit();
6  }
7
8  require_once 'controllers/GuestController.php';
9
10 $guestController = new GuestController();
11
12 if (isset($_GET['id'])) {
13     $id = $_GET['id'];
14     $stmt = $guestController->readAll();
15     while ($row = $stmt->fetch(PDO::FETCH_ASSOC)) {
16         if ($row['id'] == $id) {
17             $name = $row['name'];
18             $comment = $row['comment'];
19             break;
20         }
21     }
22 }
23
24 if ($_SERVER['REQUEST_METHOD'] == 'POST') {
25     $id = $_POST['id'];
26     $name = $_POST['name'];
27     $comment = $_POST['comment'];
28     $guestController->update($id, $name, $comment);
29     header("Location: admin.php");
30 }
31
32 >
33
34 <h2>Edit Komentar</h2>
35 <form action="edit.php" method="post">
36     <input type="hidden" name="id" value="php echo $id; ?">
37     <label for="name">Nama:</label><br>
38     <input type="text" name="name" value="php echo htmlspecialchars($name); ?" required><br>
39     <label for="comment">Komentar:</label><br>
40     <textarea name="comment" rows="5" required><?php echo htmlspecialchars($comment); ?></textarea><br>
41     <input type="submit" value="Update">
42 </form>
```

12. ./index.php

```
index.php x
buku-tamu > index.php
1  <?php
2  require_once 'controllers/GuestController.php';
3
4  $guestController = new GuestController();
5
6  if ($_SERVER['REQUEST_METHOD'] == 'POST') {
7      $name = $_POST['name'];
8      $comment = $_POST['comment'];
9      $guestController->create($name, $comment);
10     header("Location: index.php");
11 }
12
13 $guests = $guestController->readAll();
14
15 include 'views/guest_form.php';
16 include 'views/guest_list.php';
17 ?>
```

13. ./login.php

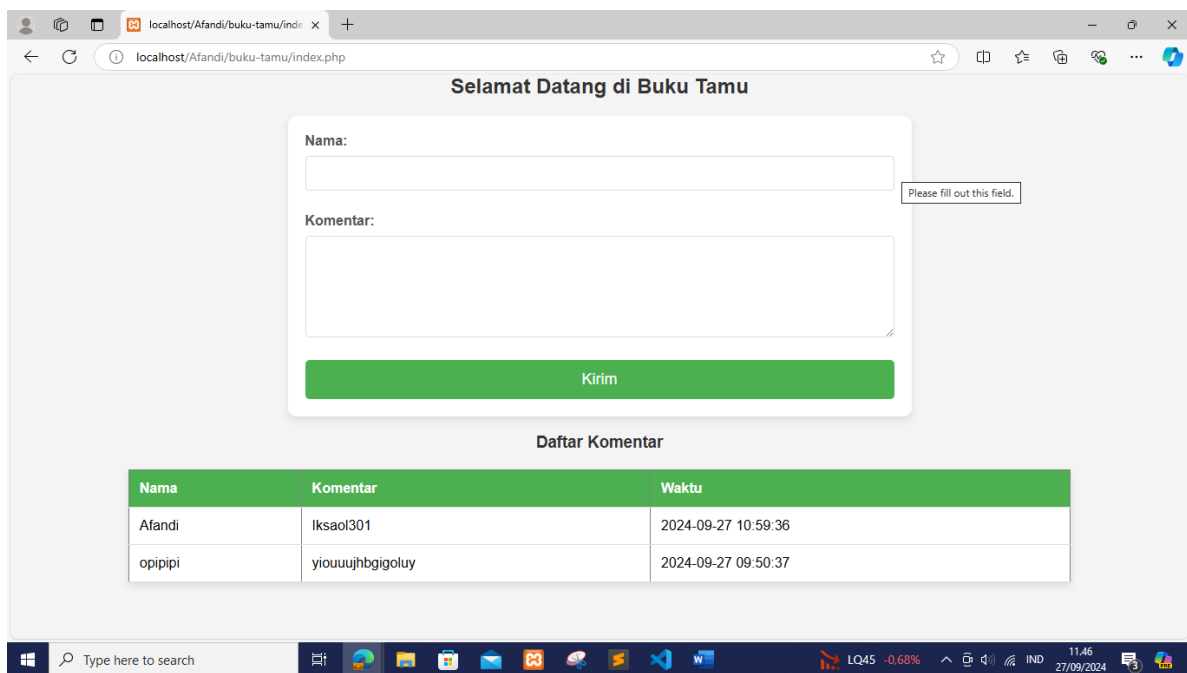
```
login.php x
buku-tamu > login.php
1  <?php
2  session_start();
3  require_once 'controllers/AdminController.php';
4
5  $adminController = new AdminController();
6
7  if ($_SERVER['REQUEST_METHOD'] == 'POST') {
8      $username = $_POST['username'];
9      $password = $_POST['password'];
10
11     if ($adminController->login($username, $password)) {
12         $_SESSION['admin'] = $username;
13         header("Location: admin.php");
14     } else {
15         $error = "Username atau password salah!";
16     }
17 }
18
19 include 'views/admin_login.php';
20
21 if (isset($error)) {
22     echo "<p style='color:red;'>$error</p>";
23 }
24 ?>
```

14. /logout.php

```
logout.php X
buku-tamu > logout.php
1  <?php
2  session_start();
3  session_unset();
4  session_destroy();
5  header("Location: login.php");
6  ?>
```

Capture Halaman Web

/index.php



Selamat Datang di Buku Tamu

Nama:

Please fill out this field.

Komentar:

Kirim

Daftar Komentar

Nama	Komentar	Waktu
Afandi	lksaol301	2024-09-27 10:59:36
opipipi	yiouuujhbgigoluy	2024-09-27 09:50:37