

LAPORAN PRAKTIKUM 4
PEMROGRAMAN BERBASIS WEB - A



Mahasiswa:

Ihsan Adi Putra (4522210085)

Dosen:

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

TEKNIK INFORMATIKA
UNIVERSITAS PANCASILA

2024

Penjelasan

1. Model Guest (models/Guest.php)

```
<?php
class Guest {
    private $conn;
    private $table_name = "guests";

    public $id;
    public $name;
    public $comment;
    public $created_at;

    public function __construct($db) {
        $this->conn = $db;
    }

    // Menambahkan data tamu baru
    public function create() {
        $query = "INSERT INTO " . $this->table_name . " SET name=:name,
comment=:comment";
        $stmt = $this->conn->prepare($query);

        // Bersihkan data
        $this->name = htmlspecialchars(strip_tags($this->name));
        $this->comment = htmlspecialchars(strip_tags($this->comment));

        // Bind data
        $stmt->bindParam(":name", $this->name);
        $stmt->bindParam(":comment", $this->comment);

        if($stmt->execute()) {
            return true;
        }
        return false;
    }

    // Mengambil semua data tamu
    public function readAll() {
        $query = "SELECT * FROM " . $this->table_name . " ORDER BY created_at
DESC";
        $stmt = $this->conn->prepare($query);
        $stmt->execute();
        return $stmt;
    }

    // Menghapus data tamu
    public function delete() {
```

```

        $query = "DELETE FROM " . $this->table_name. " WHERE id = ?";
        $stmt = $this->conn->prepare($query);
        $stmt->bindParam(1, $this->id);

        if($stmt->execute()) {
            return true;
        }
        return false;
    }

    // Mengupdate data tamu
    public function update() {
        $query = "UPDATE " . $this->table_name. " SET name=:name,
comment=:comment WHERE id=:id";
        $stmt = $this->conn->prepare($query);

        // Bersihkan data
        $this->name = htmlspecialchars(strip_tags($this->name));
        $this->comment = htmlspecialchars(strip_tags($this->comment));
        $this->id = htmlspecialchars(strip_tags($this->id));

        // Bind data
        $stmt->bindParam(":name", $this->name);
        $stmt->bindParam(":comment", $this->comment);
        $stmt->bindParam(":id", $this->id);

        if($stmt->execute()) {
            return true;
        }
        return false;
    }
}
?>

```

Deskripsi: Class untuk memanipulasi data tamu dalam database, seperti menambah, mengambil, memperbarui, dan menghapus data tamu.

Fungsi Utama:

create(): Menambahkan tamu baru.

readAll(): Mengambil semua data tamu, diurutkan berdasarkan waktu.

delete(): Menghapus tamu berdasarkan ID.

update(): Memperbarui data tamu.

2. Model Admin (models/Admin.php)

```
<?php
class Admin {
    private $conn;
    private $table_name = "admins";

    public $id;
    public $username;
    public $password;

    public function __construct($db) {
        $this->conn = $db;
    }

    // Fungsi login admin
    public function login() {
        $query = "SELECT * FROM " . $this->table_name . " WHERE username = :username LIMIT 0,1";
        $stmt = $this->conn->prepare($query);

        // Bersihkan data
        $this->username = htmlspecialchars(strip_tags($this->username));

        // Bind data
        $stmt->bindParam(":username", $this->username);
        $stmt->execute();

        $row = $stmt->fetch(PDO::FETCH_ASSOC);

        if($row) {
            if(password_verify($this->password, $row['password'])) {
                $this->id = $row['id'];
                return true;
            }
        }
        return false;
    }
}
```

Deskripsi: Class untuk menangani login admin.

Fungsi Utama:

login(): Melakukan login admin dengan memverifikasi username dan password (password diverifikasi dengan password_verify).

3. Controller Guest (controllers/GuestController.php)

```
<?php
include_once './config/Database.php';
include_once './models/Guest.php';

class GuestController {
    private $guest;

    public function __construct() {
        $database = new Database();
        $db = $database->getConnection();
        $this->guest = new Guest($db);
    }

    // Fungsi untuk menambahkan tamu baru
    public function create($name, $comment) {
        $this->guest->name = $name;
        $this->guest->comment = $comment;
        return $this->guest->create();
    }

    // Fungsi untuk mendapatkan semua data tamu
    public function readAll() {
        return $this->guest->readAll();
    }

    // Fungsi untuk menghapus tamu
    public function delete($id) {
        $this->guest->id = $id;
        return $this->guest->delete();
    }

    // Fungsi untuk mengupdate tamu
    public function update($id, $name, $comment) {
        $this->guest->id = $id;
        $this->guest->name = $name;
        $this->guest->comment = $comment;
        return $this->guest->update();
    }
}
?>
```

Deskripsi: Controller untuk menghubungkan model Guest dengan view, mengatur alur logika aplikasi.

Fungsi Utama:

create(): Mengatur proses penambahan tamu baru.

readAll(): Mengambil semua data tamu untuk ditampilkan.

delete(): Menghapus data tamu.

update(): Memperbarui data tamu.

4. Controller Admin (controllers/AdminController.php)

```
<?php
include_once './config/Database.php';
include_once './models/Admin.php';

class AdminController {
    private $admin;

    public function __construct() {
        $database = new Database();
        $db = $database->getConnection();
        $this->admin = new Admin($db);
    }

    // Fungsi untuk login
    public function login($username, $password) {
        $this->admin->username = $username;
        $this->admin->password = $password;
        return $this->admin->login();
    }
}
?>
```

Deskripsi: Controller untuk proses login admin.

Fungsi Utama:

login(): Memverifikasi username dan password admin, mengarahkan ke dashboard jika berhasil.

5. Form Buku Tamu (views/guest_form.php)

```
<h2>Selamat Datang di Buku Tamu</h2>
<form action="index.php" method="post">
    <label for="name">Nama:</label><br>
    <input type="text" name="name" required><br><br>
    <label for="comment">Komentar:</label><br>
    <textarea name="comment" rows="5" required></textarea><br><br>
```

```
<input type="submit" value="Kirim">
```

Deskripsi: Form HTML yang memungkinkan pengunjung untuk memasukkan nama dan komentar.

Elemen Utama:

Input untuk nama dan komentar.

Tombol submit untuk mengirim data ke server.

6. Daftar Buku Tamu (views/guest_list.php)

```
<h3>Daftar Komentar</h3>
<table border="1" cellpadding="5" cellspacing="0">
  <tr>
    <th>Nama</th>
    <th>Komentar</th>
    <th>Waktu</th>
  </tr>
  <?php while ($row = $guests->fetch(PDO::FETCH_ASSOC)) { ?>
    <tr>
      <td><?php echo htmlspecialchars($row['name']); ?></td>
      <td><?php echo htmlspecialchars($row['comment']); ?></td>
      <td><?php echo $row['created_at']; ?></td>
    </tr>
  <?php } ?>
</table>
```

Deskripsi: Menampilkan daftar komentar dari tamu dalam tabel.

Elemen Utama:

Tabel yang menampilkan nama, komentar, dan waktu komentar dibuat.

7. Form Login Admin (views/admin_login.php)

```
<h2>Login Admin</h2>
<form action="login.php" method="post">
  <label for="username">Username:</label><br>
  <input type="text" name="username" required><br><br>
  <label for="password">Password:</label><br>
  <input type="password" name="password" required><br><br>
  <input type="submit" value="Login">
</form>
```

Deskripsi: Form login admin, dengan input untuk username dan password.

Elemen Utama:

Input untuk username dan password.

Tombol submit untuk login.

8. Dashboard Admin (views/admin_dashboard.php)

```
<h2>Dashboard Admin</h2>
<p>Selamat datang, Admin!</p>
<a href="logout.php">Logout</a>
<table border="1" cellpadding="5" cellspacing="0">
  <tr>
    <th>Nama</th>
    <th>Komentar</th>
    <th>Waktu</th>
    <th>Aksi</th>
  </tr>
  <?php while ($row = $guests->fetch(PDO::FETCH_ASSOC)) { ?>
    <tr>
      <td><?php echo htmlspecialchars($row['name']); ?></td>
      <td><?php echo htmlspecialchars($row['comment']); ?></td>
      <td><?php echo $row['created_at']; ?></td>
      <td>
        <a href="edit.php?id=<?php echo $row['id']; ?>">Edit</a> |
        <a href="admin.php?delete_id=<?php echo $row['id']; ?>"
onclick="return confirm('Yakin ingin menghapus?')">Hapus</a>
      </td>
    </tr>
  <?php } ?>
</table>
```

Deskripsi: Halaman dashboard admin setelah berhasil login.

Fungsi Utama:

Menampilkan daftar komentar tamu, dengan opsi untuk mengedit dan menghapus komentar.

9. Halaman Utama (Index.php)

```
<style>
/* Styling dasar untuk body */
body {

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    background-color: #f4f4f9;
    color: #333;
    margin: 0;
    padding: 20px;
    line-height: 1.6;
}

/* Heading utama */
h2, h3 {
    text-align: center;
    color: #34495e;
    font-size: 2.2rem;
    margin-bottom: 20px;
}

/* Styling untuk form */
form {
    background-color: #ffffff;
    border-radius: 10px;
    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
    padding: 20px;
    max-width: 600px;
    margin: 0 auto;
    transition: transform 0.3s ease;
}

form:hover {
    transform: scale(1.02);
}

/* Styling untuk input teks dan textarea */
input[type="text"], input[type="password"], textarea {
    width: 100%;
    padding: 12px;
    border: 2px solid #bdc3c7;
    border-radius: 5px;
    margin-bottom: 20px;
    font-size: 1rem;
    transition: border-color 0.3s ease;
}

input[type="text"]:focus, input[type="password"]:focus, textarea:focus {
```

```
border-color: #3498db;
outline: none;
}

/* Styling untuk tombol submit */
input[type="submit"] {
  background-color: #3498db;
  color: #fff;
  padding: 12px 20px;
  border: none;
  border-radius: 5px;
  cursor: pointer;
  font-size: 1.1rem;
  transition: background-color 0.3s ease, transform 0.3s ease;
}

input[type="submit"]:hover {
  background-color: #2980b9;
  transform: scale(1.05);
}

/* Styling tabel buku tamu */
table {
  width: 100%;
  border-collapse: collapse;
  margin-top: 20px;
}

table th, table td {
  padding: 12px;
  text-align: left;
}

table th {
  background-color: #34495e;
  color: #ffffff;
  font-size: 1.2rem;
}

table td {
  background-color: #ffffff;
  color: #333;
}

table tr:nth-child(even) {
  background-color: #ecf0f1;
}
```

```

/* Hover effect for table rows */
table tr:hover {
    background-color: #bdc3c7;
    transition: background-color 0.3s ease;
}

/* Link styling for actions (Edit, Hapus) */
a {
    color: #3498db;
    text-decoration: none;
    font-weight: bold;
    transition: color 0.3s ease;
}

a:hover {
    color: #2980b9;
}

/* Styling untuk pesan error */
p {
    color: red;
    font-weight: bold;
    text-align: center;
    font-size: 1.1rem;
}

/* Responsif */
@media (max-width: 768px) {
    form {
        padding: 15px;
        margin: 0 10px;
    }

    input[type="submit"], input[type="text"], textarea {
        font-size: 0.9rem;
    }

    h2, h3 {
        font-size: 1.8rem;
    }
}
</style>

<?php
require_once 'controllers/GuestController.php';

$guestController = new GuestController();

```

```

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $name = $_POST['name'];
    $comment = $_POST['comment'];
    $guestController->create($name, $comment);
    header("Location: index.php");
}

$guests = $guestController->readAll();

include 'views/guest_form.php';
include 'views/guest_list.php';
?>

```

Deskripsi: Halaman utama yang memuat form buku tamu dan daftar komentar.

Alur: Mengirim data tamu baru jika form diisi, menampilkan daftar tamu menggunakan GuestController.

10. Halaman Login Admin (login.php)

```

<?php
session_start();
require_once 'controllers/AdminController.php';

$adminController = new AdminController();

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $username = $_POST['username'];
    $password = $_POST['password'];

    if ($adminController->login($username, $password)) {
        $_SESSION['admin'] = $username;
        header("Location: admin.php");
    } else {
        $error = "Username atau password salah!";
    }
}

include 'views/admin_login.php';

if (isset($error)) {
    echo "<p style='color:red;'>$error</p>";
}
?>

```

Deskripsi: Memproses login admin berdasarkan input dari form login.

Alur: Jika login berhasil, admin diarahkan ke dashboard, jika gagal, muncul pesan error.

11. Halaman Dashboard Admin (admin.php)

```
<?php
session_start();
if (!isset($_SESSION['admin'])) {
    header("Location: login.php");
    exit();
}

require_once 'controllers/GuestController.php';

$guestController = new GuestController();

if (isset($_GET['delete_id'])) {
    $id = $_GET['delete_id'];
    $guestController->delete($id);
    header("Location: admin.php");
}

$guests = $guestController->readAll();

include 'views/admin_dashboard.php';
?>
```

Deskripsi: Mengelola komentar tamu melalui dashboard, memungkinkan admin untuk menghapus komentar.

Alur: Jika admin menghapus komentar, halaman akan memproses penghapusan dan memuat ulang halaman.

12. Logout Admin (logout.php)

```
<?php
session_start();
session_unset();
session_destroy();
header("Location: login.php");
?>
```

Deskripsi: Menghapus session admin dan mengarahkan kembali ke halaman login.

13. Halaman Edit Komentar (edit.php)

```
<?php
session_start();
if (!isset($_SESSION['admin'])) {
    header("Location: login.php");
    exit();
}

require_once 'controllers/GuestController.php';

$guestController = new GuestController();

if (isset($_GET['id'])) {
    $id = $_GET['id'];
    $stmt = $guestController->readAll();
    while ($row = $stmt->fetch(PDO::FETCH_ASSOC)) {
        if ($row['id'] == $id) {
            $name = $row['name'];
            $comment = $row['comment'];
            break;
        }
    }
}

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $id = $_POST['id'];
    $name = $_POST['name'];
    $comment = $_POST['comment'];
    $guestController->update($id, $name, $comment);
    header("Location: admin.php");
}

?>

<h2>Edit Komentar</h2>
<form action="edit.php" method="post">
    <input type="hidden" name="id" value="<?php echo $id; ?>">
    <label for="name">Nama:</label><br>
    <input type="text" name="name" value="<?php echo htmlspecialchars($name);
?>" required><br><br>
    <label for="comment">Komentar:</label><br>
    <textarea name="comment" rows="5" required><?php echo
htmlspecialchars($comment); ?></textarea><br><br>
    <input type="submit" value="Update">
</form>
```

Deskripsi: Menyediakan form untuk mengedit nama dan komentar tamu berdasarkan ID.

Alur: Mengambil data tamu berdasarkan ID dan menampilkan form untuk memperbarui komentar.

14. Menambah CSS

```
<style>
/* Styling dasar untuk body */
body {

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    background-color: #f4f4f9;
    color: #333;
    margin: 0;
    padding: 20px;
    line-height: 1.6;
}

/* Heading utama */
h2, h3 {
    text-align: center;
    color: #34495e;
    font-size: 2.2rem;
    margin-bottom: 20px;
}

/* Styling untuk form */
form {
    background-color: #ffffff;
    border-radius: 10px;
    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
    padding: 20px;
    max-width: 600px;
    margin: 0 auto;
    transition: transform 0.3s ease;
}

form:hover {
    transform: scale(1.02);
}

/* Styling untuk input teks dan textarea */
input[type="text"], input[type="password"], textarea {
    width: 100%;
    padding: 12px;
    border: 2px solid #bdc3c7;
    border-radius: 5px;
    margin-bottom: 20px;
}
```

```

    font-size: 1rem;
    transition: border-color 0.3s ease;
}

input[type="text"]:focus, input[type="password"]:focus, textarea:focus {
    border-color: #3498db;
    outline: none;
}

/* Styling untuk tombol submit */
input[type="submit"] {
    background-color: #3498db;
    color: #fff;
    padding: 12px 20px;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    font-size: 1.1rem;
    transition: background-color 0.3s ease, transform 0.3s ease;
}

input[type="submit"]:hover {
    background-color: #2980b9;
    transform: scale(1.05);
}

/* Styling tabel buku tamu */
table {
    width: 100%;
    border-collapse: collapse;
    margin-top: 20px;
}

table th, table td {
    padding: 12px;
    text-align: left;
}

table th {
    background-color: #34495e;
    color: #ffffff;
    font-size: 1.2rem;
}

table td {
    background-color: #ffffff;
    color: #333;
}

```



```

table tr:nth-child(even) {
    background-color: #ecf0f1;
}

/* Hover effect for table rows */
table tr:hover {
    background-color: #bdc3c7;
    transition: background-color 0.3s ease;
}

/* Link styling for actions (Edit, Hapus) */
a {
    color: #3498db;
    text-decoration: none;
    font-weight: bold;
    transition: color 0.3s ease;
}

a:hover {
    color: #2980b9;
}

/* Styling untuk pesan error */
p {
    color: red;
    font-weight: bold;
    text-align: center;
    font-size: 1.1rem;
}

/* Responsif */
@media (max-width: 768px) {
    form {
        padding: 15px;
        margin: 0 10px;
    }

    input[type="submit"], input[type="text"], textarea {
        font-size: 0.9rem;
    }

    h2, h3 {
        font-size: 1.8rem;
    }
}
</style>

```

Deskripsi: Gaya modern dan responsif untuk tampilan form, tabel, dan elemen lain pada halaman buku tamu dan dashboard admin.

Fitur Utama:

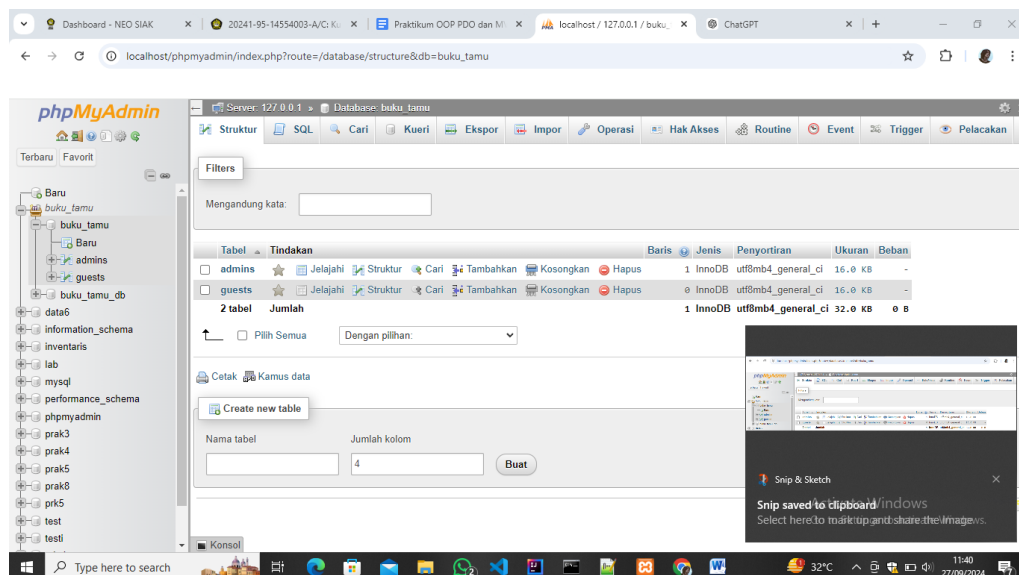
Desain minimalis dengan transisi halus.

Efek hover pada elemen form, tabel, dan tombol.

Gambar latar belakang yang responsif dengan ukuran penuh.

Hasil SS

Database dan Table



Hasil Tampilan setelah menginput css

