**Prak. Pemrograman Berbasis Web**

Untuk Memenuhi Tugas Prak. Pemrograman Berbasis Web

****

**Disusun Oleh :**

**Muhamad Suhuddin Jaballul Karim**

**4522210119**

**Dosen Pengampu :**

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

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS PANCASILA**

**JAKARTA**

**2023/2024**

**Source Code**

**buku\_tamu/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';

?>

**buku\_tamu/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>

**buku\_tamu/index.php**

<?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';

?>

**buku\_tamu/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>";

}

?>

**buku\_tamu/logut.php**

<?php

session\_start();

session\_unset();

session\_destroy();

header("Location: login.php");

?>

**buku\_tamu/config/Database.php**

<?php

class Database {

    private $host = "localhost";

    private $db\_name = "buku\_tamu";

    private $username = "root";

    private $password = "";

    public $conn;

    public function getConnection() {

        $this->conn = null;

        try {

            $this->conn = new PDO("mysql:host=" .$this->host. ";dbname=" .$this->db\_name, $this->username, $this->password);

            $this->conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

        } catch(PDOException $exception) {

            echo "Koneksi error: " . $exception->getMessage();

        }

        return $this->conn;

    }

}

?>

**buku\_tamu/controllers/adminControllers.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();

    }

}

?>

**buku\_tamu/controllers/GuestControllers.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();

    }

}

?>

**buku\_tamu/model/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;

    }

}

?>

**buku\_tamu/model/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;

    }

}

?>

**buku\_tamu/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>

**buku\_tamu/views/admin\_login.php**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Login Admin</title>

    <style>

        body {

            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

            background-color: #f7f7f7;

            margin: 0;

            padding: 0;

            display: flex;

            justify-content: center;

            align-items: center;

            height: 100vh;

        }

        form {

            background-color: white;

            padding: 30px;

            border-radius: 12px;

            box-shadow: 0px 6px 15px rgba(0, 0, 0, 0.1);

            width: 320px;

        }

        h2 {

            font-size: 24px;

            color: #333;

            text-align: center;

            margin-bottom: 20px;

        }

        label {

            display: block;

            font-size: 14px;

            color: #666;

            margin-bottom: 8px;

            text-align: left;

        }

        input[type="text"], input[type="password"] {

            width: 100%;

            padding: 12px;

            margin: 8px 0 16px;

            border: 1px solid #ccc;

            border-radius: 6px;

            box-sizing: border-box;

            transition: border-color 0.3s ease;

        }

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

            border-color: #5d9cec;

            outline: none;

        }

        input[type="submit"] {

            width: 100%;

            background-color: #5d9cec;

            color: white;

            padding: 12px;

            border: none;

            border-radius: 6px;

            cursor: pointer;

            font-size: 16px;

            transition: background-color 0.3s ease;

        }

        input[type="submit"]:hover {

            background-color: #4a8fd4;

        }

        .form-container {

            text-align: center;

            padding: 15px;

        }

    </style>

</head>

<body>

    <div class="form-container">

        <h2>Login Admin</h2>

        <form action="login.php" method="post">

            <label for="username">Username:</label>

            <input type="text" name="username" required><br>

            <label for="password">Password:</label>

            <input type="password" name="password" required><br>

            <input type="submit" value="Login">

        </form>

    </div>

</body>

</html>

**buku\_tamu/views/guest\_form.php**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Buku Tamu</title>

    <style>

        body {

            font-family: 'Arial', sans-serif;

            background-color: #f3f3f3;

            display: flex;

            justify-content: center;

            align-items: center;

            height: 100vh;

            margin: 0;

        }

        form {

            background-color: white;

            padding: 25px;

            border-radius: 8px;

            box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);

            width: 350px;

        }

        h2 {

            text-align: center;

            font-size: 24px;

            color: #333;

            margin-bottom: 20px;

        }

        label {

            display: block;

            font-size: 14px;

            color: #555;

            margin-bottom: 8px;

        }

        input[type="text"], textarea {

            width: 100%;

            padding: 10px;

            margin-bottom: 15px;

            border: 1px solid #ccc;

            border-radius: 5px;

            font-size: 14px;

            box-sizing: border-box;

        }

        textarea {

            resize: none;

        }

        input[type="submit"] {

            width: 100%;

            background-color: #4CAF50;

            color: white;

            padding: 12px;

            border: none;

            border-radius: 5px;

            font-size: 16px;

            cursor: pointer;

            transition: background-color 0.3s ease;

        }

        input[type="submit"]:hover {

            background-color: #45a049;

        }

        input:focus, textarea:focus {

            border-color: #4CAF50;

            outline: none;

        }

    </style>

</head>

<body>

    <form action="index.php" method="post">

        <h2>Selamat Datang di Buku Tamu</h2>

        <label for="name">Nama:</label>

        <input type="text" name="name" required><br>

        <label for="comment">Komentar:</label>

        <textarea name="comment" rows="5" required></textarea><br>

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

    </form>

</body>

</html>

**buku\_tamu/views/** **guest\_list.php**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Daftar Komentar</title>

    <style>

        body {

            font-family: 'Arial', sans-serif;

            background-color: #f4f4f4;

            margin: 0;

            padding: 20px;

            display: flex;

            justify-content: center;

            align-items: center;

            min-height: 100vh;

        }

        table {

            width: 80%;

            border-collapse: collapse;

            background-color: white;

            box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);

        }

        h3 {

            text-align: center;

            color: #333;

            font-size: 24px;

            margin-bottom: 20px;

        }

        th, td {

            padding: 12px 15px;

            text-align: left;

        }

        th {

            background-color: #4CAF50;

            color: white;

            font-weight: bold;

        }

        tr:nth-child(even) {

            background-color: #f2f2f2;

        }

        tr:hover {

            background-color: #eaeaea;

        }

        td {

            border-bottom: 1px solid #ddd;

        }

        td:first-child {

            width: 25%;

        }

        td:last-child {

            width: 20%;

        }

        table th, table td {

            border: 1px solid #ddd;

        }

    </style>

</head>

<body>

    <h3>Daftar Komentar</h3>

    <table>

        <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>

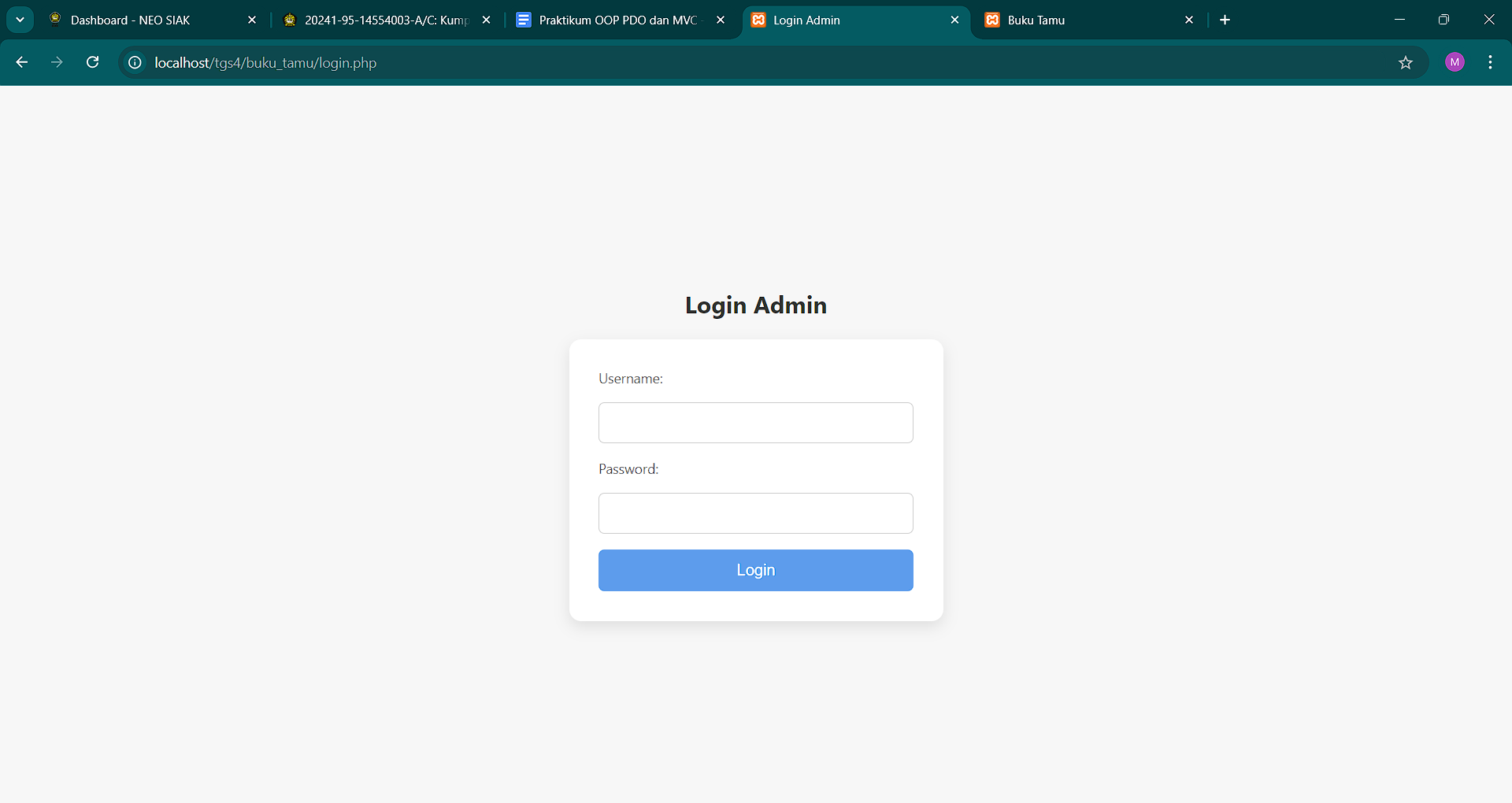
        <?php } ?>

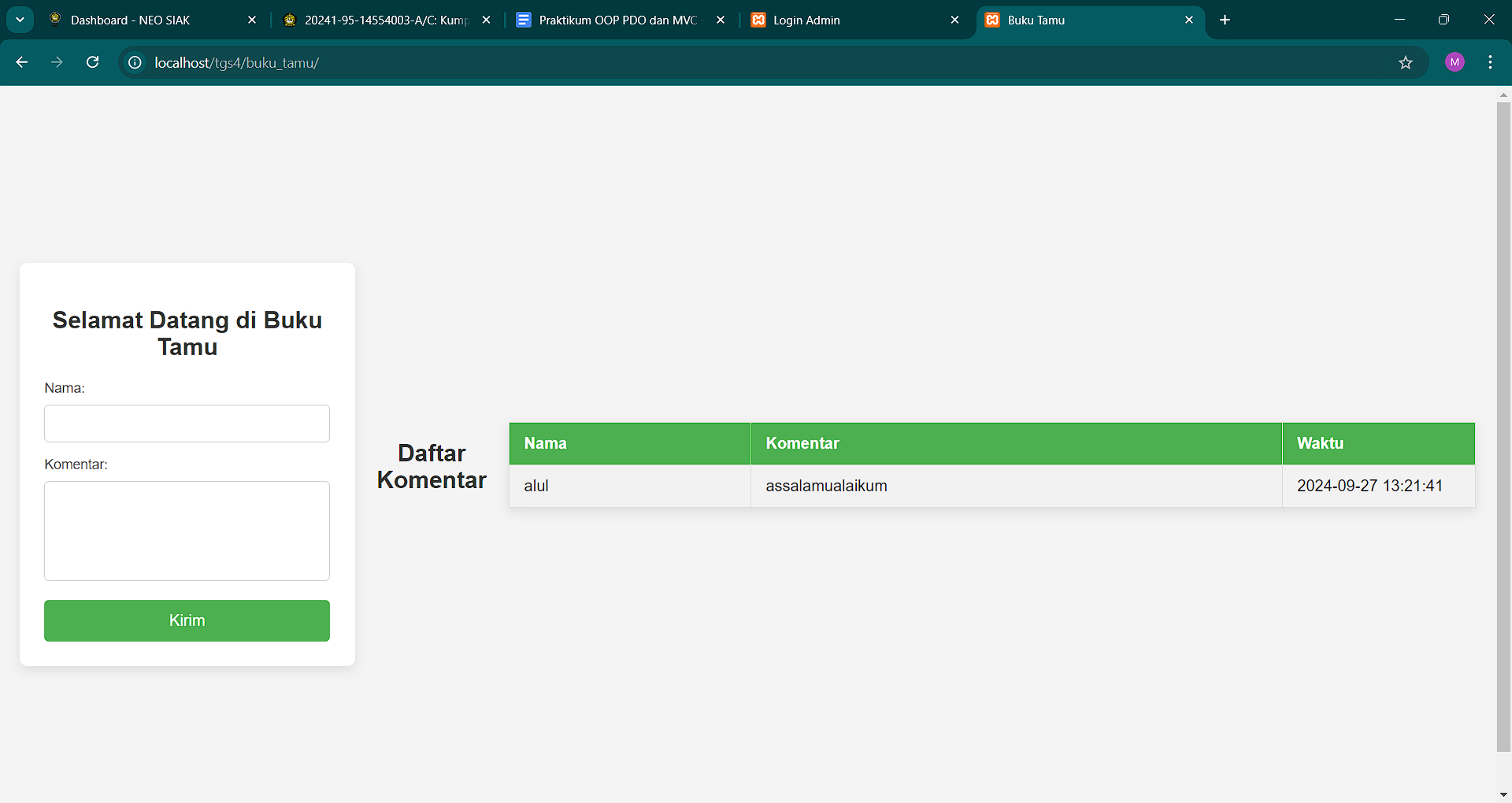
    </table>

</body>

</html>

**Hasil Running Program**

****

****

Link github:

<https://github.com/suhuddin/PBW>