

Laporan
Mata Kuliah Pemrograman Web dan Pemrograman Berorientasi
Objek



UTS

Dosen Pengampu : Willdan Aprizal Aripin, S.Pd., M.Kom.

Disusun Oleh :

Nuril Khairiyah

2300231

PROGRAM STUDI SISTEM INFORMASI KELAUTAN
UNIVERSITAS PENDIDIKAN INDONESIA
2024

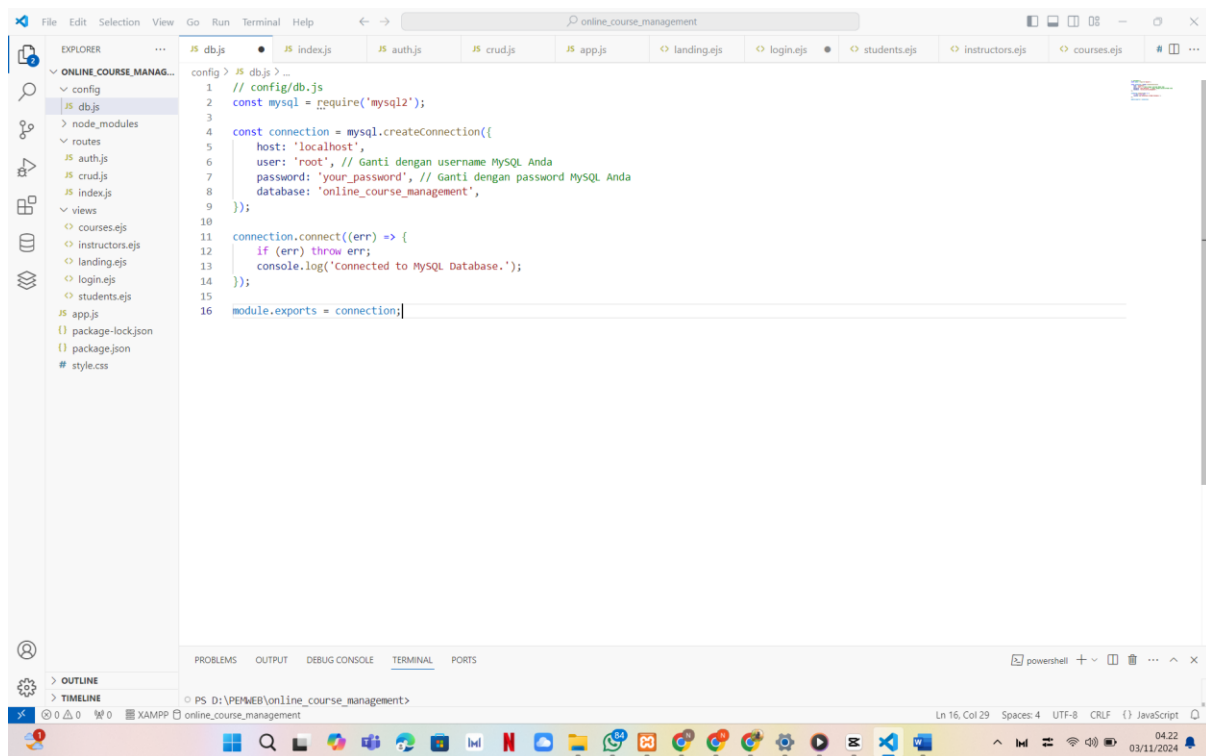
I. PENDAHULUAN

Session adalah mekanisme yang digunakan dalam aplikasi web untuk menyimpan data pengguna secara sementara selama mereka berinteraksi dengan aplikasi. Fungsinya adalah untuk mempertahankan informasi pengguna di antara permintaan HTTP yang berbeda, memungkinkan aplikasi untuk "mengingat" pengguna selama sesi berlangsung. Dalam Node.js, session digunakan untuk melacak dan menyimpan status pengguna agar data seperti informasi login, preferensi atau pengaturan tertentu tetap tersedia dan konsisten sepanjang penggunaan aplikasi tersebut meskipun terjadi beberapa permintaan atau navigasi halaman.

II. ALAT DAN BAHAN

- Visual Studio Code
- Chrome
- Laptop
- Node.js
- Xampp
- MySQL

III. DOKUMENTASI KODE



```
1 // config/db.js
2 const mysql = require('mysql2');
3
4 const connection = mysql.createConnection({
5   host: 'localhost',
6   user: 'root', // Ganti dengan username MySQL Anda
7   password: 'your_password', // Ganti dengan password MySQL Anda
8   database: 'online_course_management',
9 });
10
11 connection.connect((err) => {
12   if (err) throw err;
13   console.log('connected to MySQL Database.');
```

```
1 // routes/auth.js
2 const express = require('express');
3 const router = express.Router();
4 const db = require('../config/db');
5 const session = require('express-session');
6
7 // Middleware untuk mengecek session
8 function checkAuth(req, res, next) {
9   if (req.session.user) {
10     return next();
11   } else {
12     res.redirect('/login');
13   }
14 }
15
16 // Halaman login
17 router.get('/login', (req, res) => {
18   res.render('login');
19 });
20
21 router.post('/login', (req, res) => {
22   const { username, password } = req.body;
23   db.query('SELECT * FROM users WHERE username = ? AND password = ?', [username, password], (err, results) => {
24     if (err) throw err;
25     if (results.length > 0) {
26       req.session.user = results[0];
27       res.redirect('/');
28     } else {
29       res.redirect('/login');
30     }
31   });
32 });
33
34 // Tambah user baru
35 router.post('/register', (req, res) => {
36   const { username, password, role } = req.body;
37   db.query('INSERT INTO users (username, password, role) VALUES (?, ?, ?)', [username, password, role], (err) => {
38     if (err) throw err;
39   });
40 });
```

```
1 // routes/crud.js
2 const express = require('express');
3 const router = express.Router();
4 const db = require('../config/db');
5
6 // CRUD untuk Students
7 router.get('/students', (req, res) => {
8   db.query('SELECT * FROM students', (err, results) => {
9     if (err) throw err;
10     res.render('students', { students: results });
11   });
12 });
13
14 // Tambah siswa
15 router.post('/students', (req, res) => {
16   const { name, email } = req.body;
17   db.query('INSERT INTO students (name, email) VALUES (?, ?)', [name, email], (err) => {
18     if (err) throw err;
19     res.redirect('/students');
20   });
21 });
22
23 // Edit siswa
24 router.get('/students/edit/:id', (req, res) => {
25   const { id } = req.params;
26   db.query('SELECT * FROM students WHERE id = ?', [id], (err, results) => {
27     if (err) throw err;
28     res.render('student', { student: results[0] });
29   });
30 });
31
32 router.post('/students/edit/:id', (req, res) => {
33   const { id } = req.params;
34   const { name, email } = req.body;
35   db.query('UPDATE students SET name = ?, email = ? WHERE id = ?', [name, email, id], (err) => {
36     if (err) throw err;
37     res.redirect('/students');
38   });
39 });
```

The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying the project structure. The file explorer shows a folder named 'ONLINE...' containing 'config', 'db.js', 'node_modules', 'routes', 'auth.js', 'crud.js', 'index.js', 'views', 'courses.ejs', 'instructors.ejs', 'landing.ejs', 'login.ejs', 'students.ejs', 'app.js', 'package-lock.json', 'package.json', and 'style.css'. The 'routes/index.js' file is selected and open in the editor. The code in the file is as follows:

```
1 // routes/index.js
2 const express = require('express');
3 const router = express.Router();
4
5 router.get('/', (req, res) => {
6   res.render('landing');
7 });
8
9 module.exports = router;
```

The bottom status bar shows the file is at line 9, column 18, with 4 spaces, UTF-8 encoding, and CRLF line endings. The system tray at the bottom shows the date and time as 03/11/2024, 03:46.

The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying the project structure. The file explorer shows a folder named 'ONLINE...' containing 'config', 'db.js', 'node_modules', 'routes', 'auth.js', 'crud.js', 'index.js', 'views', 'courses.ejs', 'instructors.ejs', 'landing.ejs', 'login.ejs', 'students.ejs', 'app.js', 'package-lock.json', 'package.json', and 'style.css'. The 'views/courses.ejs' file is selected and open in the editor. The code in the file is as follows:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Courses</title>
5 </head>
6 <body>
7   <h1>Courses List</h1>
8   <table>
9     <thead>
10      <tr>
11        <th>Title</th>
12        <th>Description</th>
13        <th>Instructor</th>
14        <th>Actions</th>
15      </tr>
16    </thead>
17    <tbody>
18      <% courses.forEach(course => { %>
19        <tr>
20          <td><%= course.title %></td>
21          <td><%= course.description %></td>
22          <td><%= course.instructor_name %></td>
23          <td>
24            <a href="/courses/edit/<%= course.id %>">Edit</a>
25            <a href="/courses/delete/<%= course.id %>">Delete</a>
26          </td>
27        </tr>
28      <% }) %>
29    </tbody>
30  </table>
31  <form action="/courses" method="POST">
32    <input type="text" name="title" placeholder="Course Title" required>
33    <textarea name="description" placeholder="Description"></textarea>
34    <select name="instructor_id" required>
35      <% instructors.forEach(instructor => { %>
36        <option value="<%= instructor.id %>"><%= instructor.name %></option>
37      <% }) %>
38    </select>
39  </form>
```

The bottom status bar shows the file is at line 15, column 18, with 4 spaces, UTF-8 encoding, and CRLF line endings. The system tray at the bottom shows the date and time as 03/11/2024, 03:47.

The screenshot shows the Visual Studio Code editor with the file `instructors.ejs` open. The Explorer sidebar on the left shows the project structure, including `views`, `routes`, `controllers`, and `models`. The `instructors.ejs` file is selected in the Explorer. The main editor area displays the following HTML code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <title>Instructors</title>
5 </head>
6 <body>
7 <h1>Instructors List</h1>
8 <table>
9 <thead>
10 <tr>
11 <th>Name</th>
12 <th>Bio</th>
13 <th>Actions</th>
14 </tr>
15 </thead>
16 <tbody>
17 <%= instructors.forEach(instructor => { %>
18 <tr>
19 <td>%= instructor.name %</td>
20 <td>%= instructor.bio %</td>
21 <td>
22 <a href="/instructors/edit/%= instructor.id %">Edit</a>
23 <a href="/instructors/delete/%= instructor.id %">Delete</a>
24 </td>
25 </tr>
26 <%= }) %>
27 </tbody>
28 </table>
29 <form action="/instructors" method="POST">
30 <input type="text" name="name" placeholder="Name" required>
31 <textarea name="bio" placeholder="Bio"></textarea>
32 <button type="submit">Add Instructor</button>
33 </form>
34 </body>
35 </html>
```

The bottom of the editor shows the TERMINAL panel with the command prompt at `PS D:\PEP\MEB\online_course_management>`. The status bar at the bottom indicates the file is at line 35, column 8, with 4 spaces, UTF-8 encoding, and CRLF line endings.

The screenshot shows the Visual Studio Code editor with the file `landing.ejs` open. The Explorer sidebar on the left shows the project structure, including `views`, `routes`, `controllers`, and `models`. The `landing.ejs` file is selected in the Explorer. The main editor area displays the following HTML code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <title>Landing Page</title>
5 </head>
6 <body>
7 <h1>Welcome to Online Course Management System</h1>
8 <a href="/login">Login</a>
9 </body>
10 </html>
```

The bottom of the editor shows the TERMINAL panel with the command prompt at `PS D:\PEP\MEB\online_course_management>`. The status bar at the bottom indicates the file is at line 10, column 8, with 4 spaces, UTF-8 encoding, and CRLF line endings.

online_course_management

login.ejs

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>login</title>
5 </head>
6 <body>
7   <h1>login</h1>
8   <form action="/login" method="POST">
9     <input type="text" name="name" placeholder="name" required>
10    <input type="password" name="password" placeholder="Password" required>
11    <button type="submit">login</button>
12  </form>
13 </body>
14 </html>
```

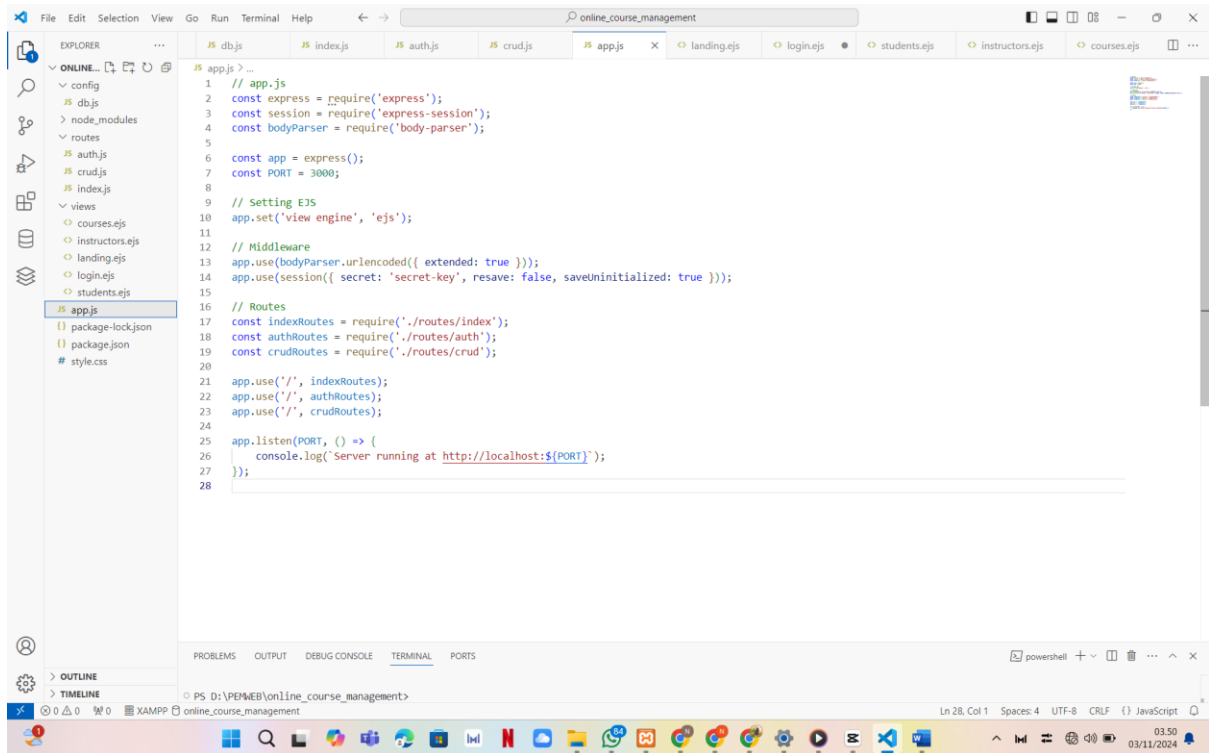
PS D:\PEM\EB\online_course_management>

online_course_management

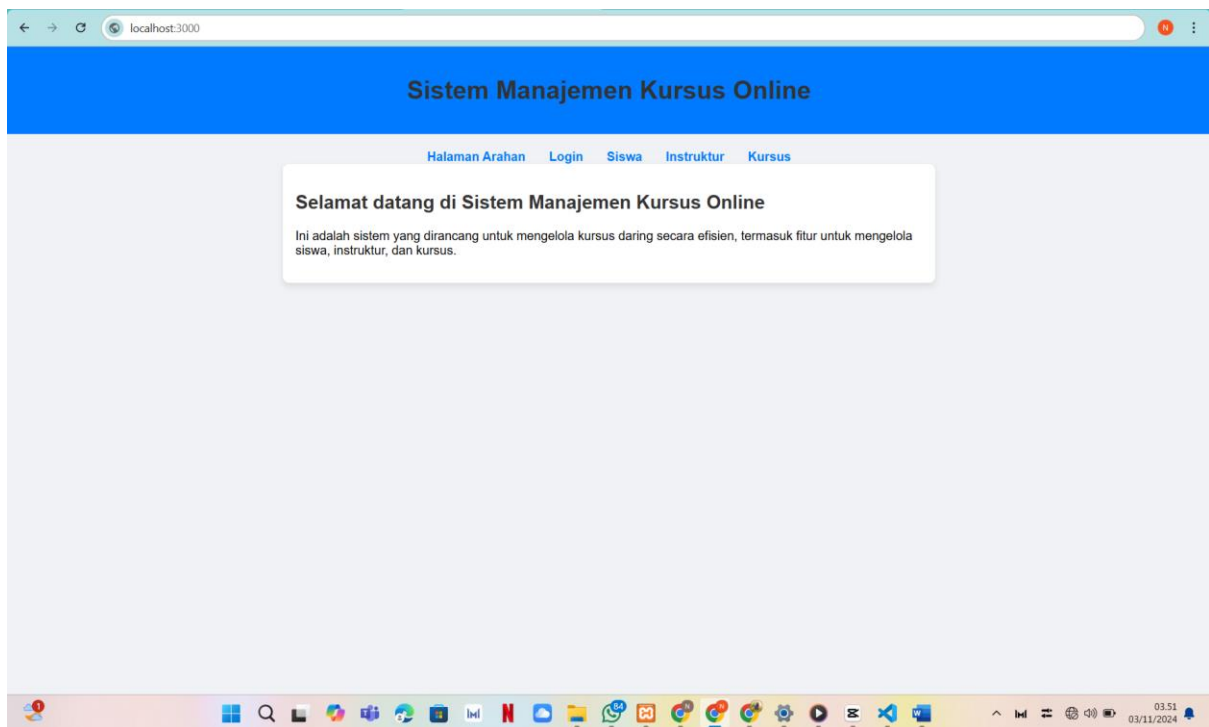
students.ejs

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Students</title>
5 </head>
6 <body>
7   <h1>Students List</h1>
8   <table>
9     <thead>
10      <tr>
11        <th>Name</th>
12        <th>Email</th>
13        <th>Actions</th>
14      </tr>
15    </thead>
16    <tbody>
17      <% students.forEach(student => { %>
18        <tr>
19          <td><%= student.name %></td>
20          <td><%= student.email %></td>
21          <td>
22            <a href="/students/edit/<%= student.id %>">Edit</a>
23            <a href="/students/delete/<%= student.id %>">Delete</a>
24          </td>
25        </tr>
26      <% }) %>
27    </tbody>
28  </table>
29  <form action="/students" method="POST">
30    <input type="text" name="name" placeholder="Name" required>
31    <input type="email" name="email" placeholder="Email" required>
32    <button type="submit">Add Student</button>
33  </form>
34 </body>
35 </html>
```

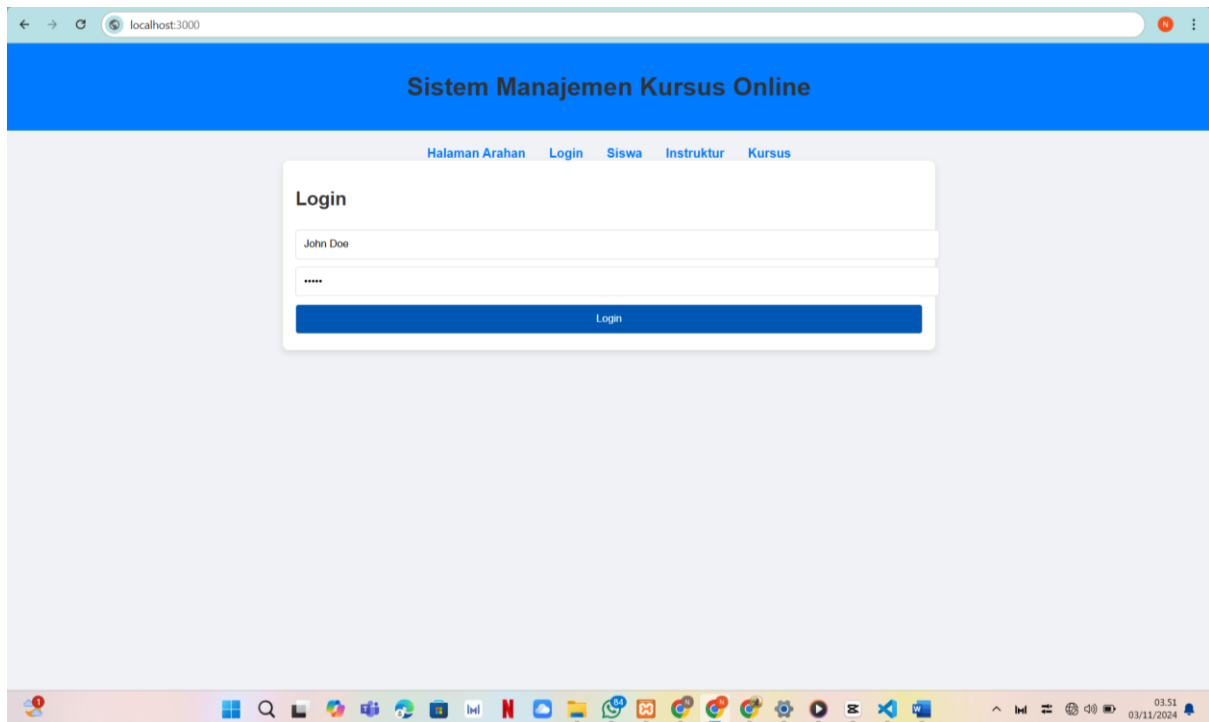
PS D:\PEM\EB\online_course_management>



IV. PANDUAN



Halaman pertama

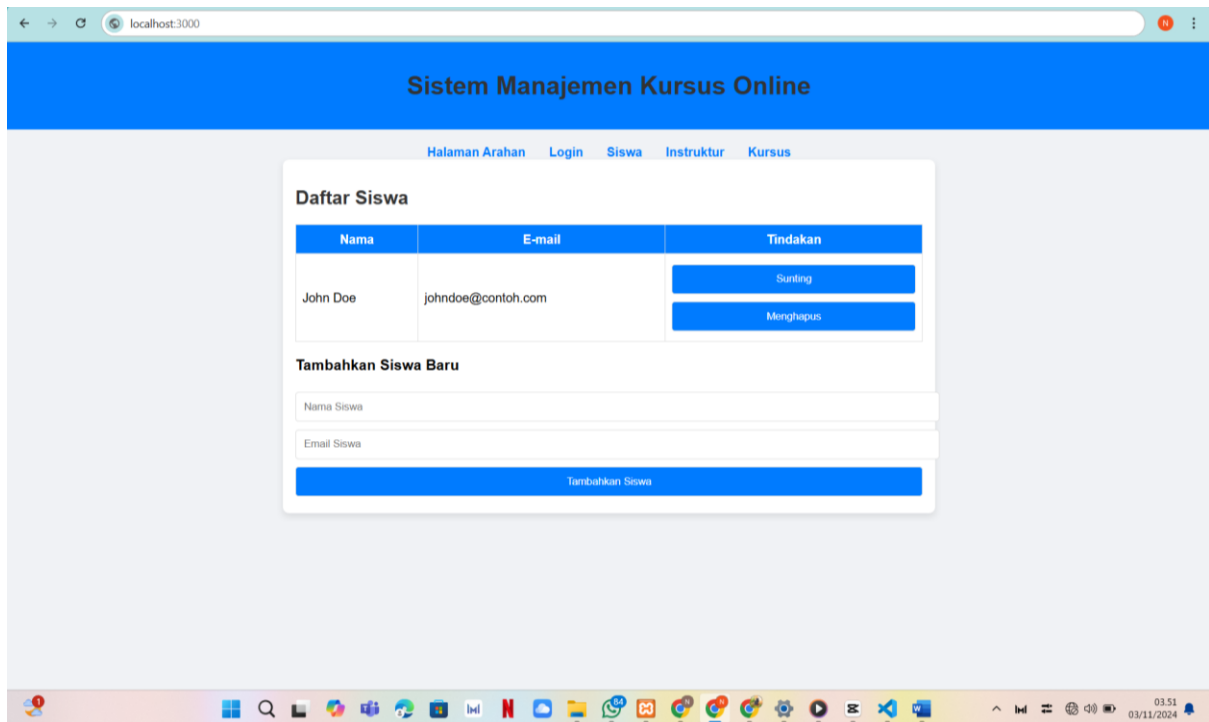


Halaman Login

- Saat pertama kali mengunjungi web Anda akan diarahkan ke halaman login.
- Masukkan nama pengguna (username) dan kata sandi (password) di kolom yang tersedia.
- Tekan tombol “Login” untuk masuk ke sistem.

Login ke Sistem

- Ketik nama pengguna (misalnya, “John Doe”) dan kata sandi.
- Klik “Login” untuk melanjutkan.
- Jika login berhasil Anda akan diarahkan ke halaman berikutnya



Tampilan Daftar Siswa

- Tabel Daftar Siswa: Menampilkan daftar siswa yang terdaftar di sistem dengan kolom Nama, E-mail, dan Tindakan.
- Nama dan E-mail: Menunjukkan nama lengkap siswa dan alamat e-mail yang terkait.
- Tindakan: Terdapat dua tombol untuk setiap siswa:
 - Sunting: Klik tombol ini untuk mengedit informasi siswa yang sudah ada.
 - Menghapus: Klik tombol ini untuk menghapus siswa dari sistem.

Menambahkan Siswa Baru

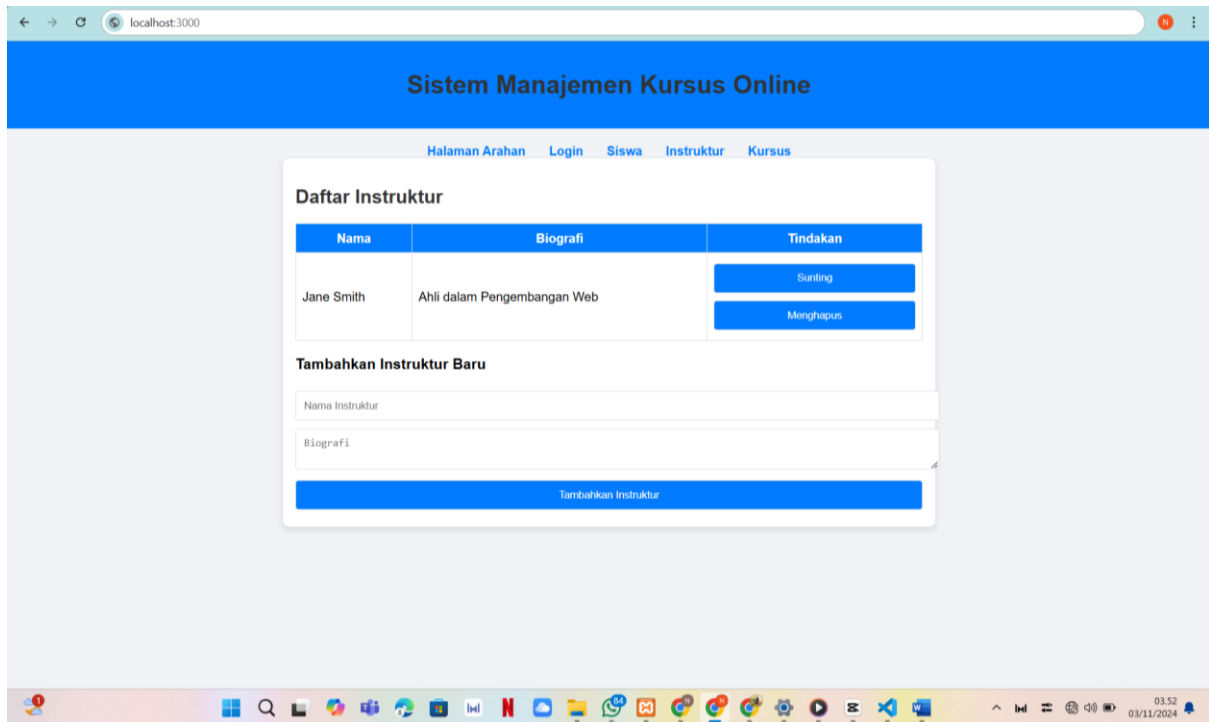
- Formulir Tambah Siswa: Terletak di bawah tabel daftar siswa.
- Input Nama Siswa: Masukkan nama lengkap siswa di kolom “Nama Siswa”.
- Input Email Siswa: Masukkan alamat e-mail siswa di kolom “Email Siswa”.
- Tombol “Tambahkan Siswa”: Tekan tombol ini setelah mengisi kedua kolom untuk menambahkan siswa baru ke dalam sistem.

Mengedit Data Siswa

- Tekan tombol “Sunting” pada baris siswa yang ingin diedit.
- Formulir edit akan muncul untuk mengubah data siswa.
- Simpan perubahan dengan menekan tombol “Simpan” setelah selesai mengedit.

Menghapus Data Siswa

- Tekan tombol “Menghapus” pada baris siswa yang ingin dihapus.
- Akan muncul kotak konfirmasi untuk memastikan apakah Anda yakin ingin menghapus data siswa tersebut.
- Tekan “Ya” atau “Hapus” untuk melanjutkan, atau “Batal” untuk membatalkan.



Tampilan Daftar Instruktur

- Tabel Daftar Instruktur: Menampilkan informasi lengkap tentang instruktur dengan kolom Nama, Biografi, dan Tindakan.
- Nama dan Biografi: Menampilkan nama instruktur dan deskripsi singkat tentang keahlian atau latar belakang mereka.
- Tindakan: Setiap baris instruktur memiliki dua tombol tindakan:
 - Sunting: Digunakan untuk mengedit informasi instruktur.
 - Menghapus: Digunakan untuk menghapus instruktur dari sistem.

Menambahkan Instruktur Baru

- Formulir Tambah Instruktur: Terletak di bawah tabel daftar instruktur.
- Input Nama Instruktur: Masukkan nama lengkap instruktur di kolom “Nama Instruktur”.
- Input Biografi: Masukkan biografi atau deskripsi singkat mengenai latar belakang instruktur di kolom “Biografi”.

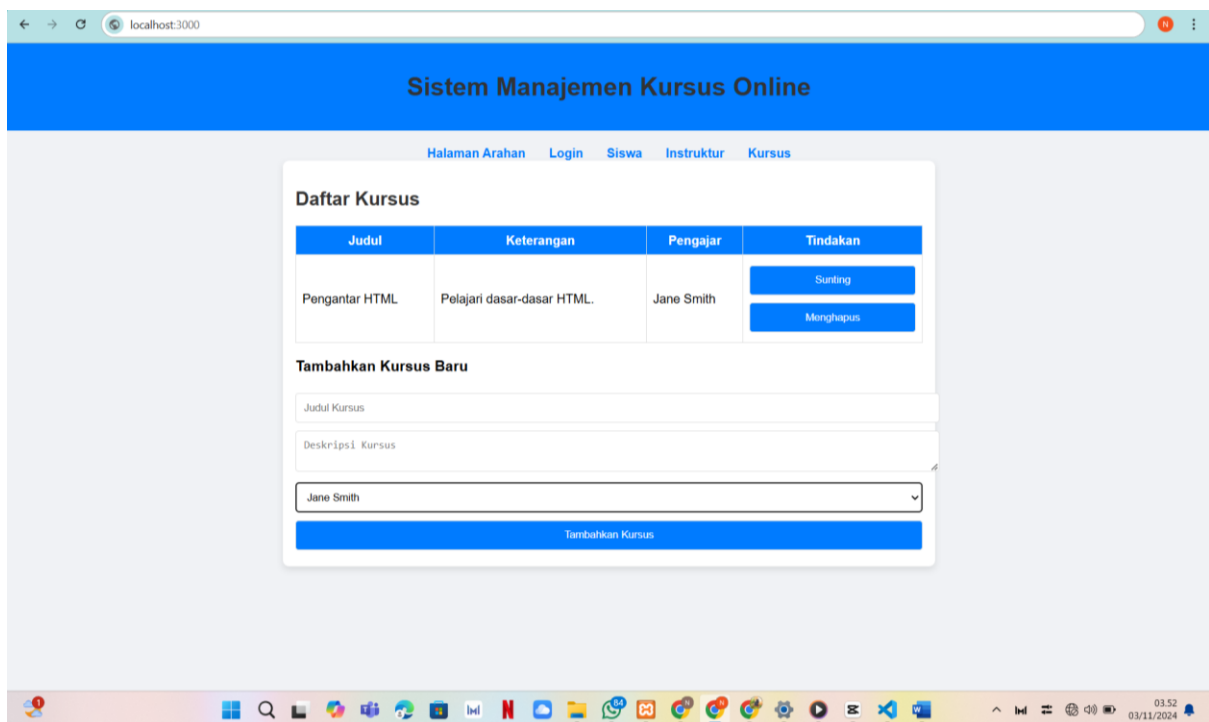
- Tombol “Tambahkan Instruktur”: Tekan tombol ini setelah mengisi kedua kolom untuk menambahkan instruktur baru ke dalam sistem.

3. Mengedit Data Instruktur

- Tekan tombol “Sunting” pada baris instruktur yang ingin diedit.
- Formulir edit akan muncul memungkinkan Anda untuk memperbarui informasi seperti nama atau biografi.
- Simpan perubahan dengan menekan tombol “Simpan” setelah selesai mengedit.

4. Menghapus Data Instruktur

- Tekan tombol “Menghapus” pada baris instruktur yang ingin dihapus.
- Akan muncul kotak konfirmasi untuk memastikan bahwa Anda ingin menghapus data instruktur tersebut.
- Tekan “Ya” atau “Hapus” untuk mengonfirmasi penghapusan, atau “Batal” jika ingin membatalkannya.



Daftar Kursus

Bagian ini menampilkan semua kursus :

- Judul: Nama kursus yang tersedia.
- Keterangan: Deskripsi singkat mengenai kursus.
- Pengajar: Nama instruktur yang mengajar kursus tersebut.

- Tindakan: Tombol Sunting untuk mengedit data kursus dan Menghapus untuk menghapus kursus dari daftar.

Menambahkan Kursus Baru

Di bawah daftar kursus, terdapat formulir untuk menambahkan kursus baru:

- Judul Kursus: Masukkan nama kursus yang ingin ditambahkan.
- Deskripsi Kursus: Tulis deskripsi singkat mengenai kursus.
- Pengajar: Pilih instruktur dari daftar yang tersedia.
- Setelah semua data diisi, klik Tambahkan Kursus untuk menambahkan kursus baru ke daftar.

Tips Penggunaan

- Pastikan semua data sudah diisi dengan benar sebelum menambahkan atau menyimpan perubahan.
- Gunakan tombol Sunting untuk memperbarui data kursus, dan tombol Menghapus jika kursus sudah tidak tersedia.

LINK YOUTUBE PRESENTASI

[https://youtu.be/ bSgeDW9fTU](https://youtu.be/bSgeDW9fTU)