

# JURNAL MODUL 15

Nama : Ghaza Zidane Nurraihan

Nim : 2311104038

Kelas : S1-SE07-01

Index.html

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <meta charset="UTF-8">
5    <title>Login & Register</title>
6    <style>
7      body { font-family: sans-serif; margin: 20px; }
8      input { display: block; margin-bottom: 10px; width: 100%; padding: 8px; }
9      button { padding: 10px; margin-top: 10px; width: 100%; }
10   </style>
11 </head>
12 <body>
13   <h2>Registrasi</h2>
14   <input type="text" id="regUsername" placeholder="Username">
15   <input type="password" id="regPassword" placeholder="Password">
16   <button onclick="register()">Register</button>
17
18   <h2>Login</h2>
19   <input type="text" id="loginUsername" placeholder="Username">
20   <input type="password" id="loginPassword" placeholder="Password">
21   <button onclick="login()">Login</button>
22
23   <p id="result"></p>
24
25   <script src="renderer.js"></script>
26 </body>
27 </html>
28
```

## Main.js

```
1  const { app, BrowserWindow } = require('electron');
2  const path = require('path');
3
4  function createWindow() {
5    const win = new BrowserWindow({
6      width: 500,
7      height: 600,
8      webPreferences: {
9        preload: path.join(__dirname, 'preload.js'),
10       nodeIntegration: true, // penting agar bisa akses fs di renderer
11       contextIsolation: false
12     }
13   });
14
15   win.loadFile('index.html');
16 }
17
18 app.whenReady().then(() => {
19   createWindow();
20 });
21
```

## Renderer.js

```
1  const UserManager = require('./userManager');
2
3  window.register = () => {
4    const username = document.getElementById('regUsername').value;
5    const password = document.getElementById('regPassword').value;
6    const result = UserManager.register(username, password);
7    document.getElementById('result').innerText = result;
8  }
9
10 window.login = () => {
11   const username = document.getElementById('loginUsername').value;
12   const password = document.getElementById('loginPassword').value;
13   const result = UserManager.login(username, password);
14   document.getElementById('result').innerText = result;
15 }
16
```

## UserManajer.js

```
1  const fs = require('fs');
2  const crypto = require('crypto');
3
4  class UserManager {
5    static readUsers() {
6      try {
7        const data = fs.readFileSync('./users.json', 'utf8');
8        return JSON.parse(data);
9      } catch {
10       return [];
11     }
12   }
13
14   static saveUsers(users) {
15     fs.writeFileSync('./users.json', JSON.stringify(users, null, 2));
16   }
17
18   static hashPassword(password) {
19     return crypto.createHash('sha256').update(password).digest('hex');
20   }
21
22   static isValidUsername(username) {
23     return /^[A-Za-z]+$/.test(username) && username.length >= 4 && username.length <= 16;
24   }
25
26   static isValidPassword(password, username) {
27     return /[!@#%&^*]/.test(password) &&
28       /[0-9]/.test(password) &&
29       !password.toLowerCase().includes(username.toLowerCase()) &&
30       password.length >= 8 && password.length <= 20;
31   }
32
33   static register(username, password) {
34     if (!this.isValidUsername(username)) {
35       return "Username hanya huruf A-Z, 4-16 karakter.";
36     }
37
38     if (!this.isValidPassword(password, username)) {
39       return "Password harus 8-20 karakter, mengandung simbol dan angka, dan tidak boleh mengandung username.";
40     }
41
42     const users = this.readUsers();
43     if (users.find(user => user.username === username)) {
44       return "Username sudah terdaftar.";
45     }
46
47     const hashed = this.hashPassword(password);
48     users.push({ username, password: hashed });
49     this.saveUsers(users);
50     return "Registrasi berhasil.";
51   }
52
53   static login(username, password) {
54     const users = this.readUsers();
55     const hashed = this.hashPassword(password);
56     return users.find(u => u.username === username && u.password === hashed)
57       ? "Login berhasil."
58       : "Username atau password salah.";
59   }
60 }
61
62 module.exports = UserManager;
63
```

Login & Register

File Edit View Window Help

### Registrasi

ghazali

\*\*\*\*\*

Register

### Login

Username

Password

Login

Registrasi berhasil.

Login & Register

File Edit View Window Help

### Registrasi

ghazali

\*\*\*\*\*

Register

### Login

ghazali

\*\*\*\*\*

Login

Login berhasil.

Aplikasi ini adalah aplikasi desktop berbasis Electron yang menyediakan fitur registrasi dan login user dengan tampilan UI sederhana menggunakan HTML, CSS, dan JavaScript. Saat user melakukan registrasi, input username dan password akan divalidasi sesuai secure coding practices: username harus huruf A-Z dengan panjang 4–16 karakter, dan password harus 8–20 karakter serta mengandung angka, simbol, dan tidak boleh mengandung username. Jika valid, password akan di-hash menggunakan SHA-256 dan disimpan ke file users.json. Pada proses login, sistem mencocokkan username dan password yang sudah di-hash.