

Journal Mandiri Modul 15

Nama: Satria Ramadhan

Nim: 2311104026

Kelas: SE-07-01

1. Source Code

a. Index.js

```
15_ReviewTugasBesar > JM_Modul_15 > index.js > ...
1  const readline = require("readline");
2  const { register, login } = require("./auth");
3
4  const rl = readline.createInterface({
5    input: process.stdin,
6    output: process.stdout,
7  });
8
9  console.log("Pilih menu: \n1. Registrasi\n2. Login");
10 rl.question("Pilihan: ", (menu) => {
11   if (menu === "1" || menu === "2") {
12     rl.question("Masukkan username: ", (username) => {
13       rl.question("Masukkan password: ", (password) => {
14         if (menu === "1") {
15           register(username, password);
16         } else {
17           login(username, password);
18         }
19         rl.close();
20       });
21     });
22   } else {
23     console.log("Menu tidak valid");
24     rl.close();
25   }
26 });
```

b. auth.js

```
15_ReviewTugasBesar > JM_Modul_15 > auth.js > ...
1  const fs = require("fs");
2  const crypto = require("crypto");
3  const usersFile = "users.json";
4
5  // Load data pengguna
6  function loadUsers() {
7    if (!fs.existsSync(usersFile)) {
8      return [];
9    }
10   return JSON.parse(fs.readFileSync(usersFile));
11 }
12
13 // Simpan data pengguna
14 function saveUsers(users) {
15   fs.writeFileSync(usersFile, JSON.stringify(users, null, 2));
16 }
17
18 // Validasi username dan password
19 function validateInput(username, password) {
20   const asciiRegex = /^[a-zA-Z0-9-014+]/;
21   const specialCharRegex = /^[!@%&'<*>+]/;
22
23   if (
24     username.length < 5 ||
25     username.length > 15 ||
26     !asciiRegex.test(username)
27   ) {
28     return "Username harus 5-15 karakter dan hanya boleh huruf/angka ASCII";
29   }
30
31   if (
32     password.length < 8 ||
33     password.length > 20 ||
34     !/d/.test(password) ||
35     !specialCharRegex.test(password) ||
36     password.toLowerCase().includes(username.toLowerCase())
37   ) {
38     return "Password minimal 8 karakter, mengandung angka & karakter spesial, dan tidak boleh mengandung username";
39   }
40   return null; // valid
41 }
42
43 // Hash password
44 function hashPassword(password) {
45   return crypto.createHash("sha256").update(password).digest("hex");
46 }
47
48 }
```

```

48 // Registrasi user baru
49 function register(username, password) {
50   const validation = validateInput(username, password);
51   if (validation) {
52     console.log("Registrasi gagal: ${validation}");
53     return;
54   }
55
56   const users = loadUsers();
57   if (users.find((user) => user.username === username)) {
58     console.log("Username sudah terdaftar.");
59     return;
60   }
61
62   users.push({ username, password: hashPassword(password) });
63   saveUsers(users);
64   console.log("Registrasi berhasil!");
65 }
66
67 // Login user
68 function login(username, password) {
69   const users = loadUsers();
70   const hashed = hashPassword(password);
71   const user = users.find(
72     (user) => user.username === username && user.password === hashed
73   );
74
75   if (user) {
76     console.log("Login berhasil! Selamat datang,", username);
77   } else {
78     console.log("Login gagal. Username atau password salah.");
79   }
80 }
81
82 module.exports = { register, login };
83
84

```

2. Output

```

• >> node index.js
Pilih menu:
1. Registrasi
2. Login
Pilihan: 1
Masukkan username: Viona
Masukkan password: MieAyam18@
Registrasi berhasil!
> pwsh JM_Modul_15 main ?14 ~1 -28 12s 941ms 22.14.0
• >> node index.js
Pilih menu:
1. Registrasi
2. Login
Pilihan: 2
Masukkan username: Viona
Masukkan password: MieAyam18@
Login berhasil! Selamat datang, Viona

```

3. Penjelasan

Tugas Jurnal Modul 15 ini mengimplementasikan sistem registrasi dan login user berbasis CLI menggunakan Node.js dengan praktik keamanan kode (secure coding practices) seperti validasi input dan password hashing. Program ini memisahkan logika ke dalam dua file utama: `auth.js` untuk proses login dan registrasi, serta `index.js` sebagai antarmuka utama. Validasi dilakukan untuk memastikan username hanya menggunakan karakter ASCII dan memiliki panjang 5–15 karakter, sementara password wajib memiliki panjang 8–20 karakter, mengandung angka dan karakter spesial, serta tidak boleh mengandung bagian dari username. Password disimpan dalam bentuk hash SHA256 dan seluruh data disimpan ke file JSON. Dengan struktur ini, program tidak hanya memenuhi kebutuhan fungsional tetapi juga menjaga keamanan dasar data pengguna.