**Objective**

Conduct white box testing to identify logic errors, security flaws, and improve code reliability in the authentication module. Focus is on **login**, **registration**, and **password reset** functionalities.

**Component Tested:**

* loginUser(email, password)
* registerUser(name, email, password)
* resetPassword(email, token, newPassword)

**Identified Vulnerability:**

| **Vulnerability Type** | **Description** | |
| --- | --- | --- |
| Input Validation Lapses | Email/password not consistently validated using regex or length checks. | |
| Weak Hashing | Found use of MD5 instead of a stronger hashing algorithm like BCrypt. | |
| Hardcoded Secrets | Static JWT secret key in code (should be stored in env vars). | |
| Token Expiry Logic | Password reset tokens not checked for expiration. | |
| Missing Null Checks | | Some null/undefined conditions not handled (can lead to exceptions). |

**Test Cases:**

**Login Function: (loginUser())**

| **Test Case** | **Description** | | | | | **Expected Result** | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| TC\_L1 | Valid email & password | | | | | Success login, return session token | | |
| TC\_L2 | | | Invalid email format | | | | Validation error | |
| TC\_L3 | | | | Wrong password | Error: "Incorrect credentials" | | | |
| TC\_L4 | | | | SQL Injection in input | Reject input, log attempt | | | |
| TC\_L5 | | Null input | | | | | | Handle gracefully, return error |

**Registration Function: (registration())**

| **Test Case** | **Description** | **Expected Result** |
| --- | --- | --- |
| TC\_R1 | New user valid data | Register and hash password securely |
| TC\_R2 | Existing email | Return error: "Email already in use" |
| TC\_R3 | Weak password (e.g., "123") | Enforce password policy |
| TC\_R4 | Script injection | Sanitize input and reject |
| TC\_R5 | No email provided | Return validation error |

**Password Reset Function: (resetFunction())**

| **Test Case** | **Description** | **Expected Result** |
| --- | --- | --- |
| TC\_P1 | Valid email and token | Allow password reset |
| TC\_P2 | Expired token | Reject and prompt new request |
| TC\_P3 | Tampered token | Log error and deny access |
| TC\_P4 | Empty fields | Return error: "All fields required" |

**Tool:**

* JUnit.
* JaCoCo.
* SonarQube

**Security Enhancements Suggested:**

| **Issue** | **Recommendation** |
| --- | --- |
| Weak hash (MD5) | Replace with BCrypt or Argon2 |
| Hardcoded secrets | Move to environment/config file |
| Input sanitization | Use validation libraries (e.g., Hibernate Validator) |
| Token checks | Implement expiry validation & refresh logic |