

Unit Test Description – Lab Space/Desktop Allocation and Server Management

Overview:

This document provides a concise summary of the unit test cases written for the project “**Lab Space/Desktop Allocation and Server Management**”. The unit tests aim to validate the correctness, reliability, and behavior of individual components within the application, especially focusing on the controller, service, and utility layers.

Structure of Unit Testing:

1. Controller Layer Tests

The controller tests validate the REST APIs for proper status codes, data responses, and error handling. Each test mocks the service layer to isolate controller behaviour.

Controllers tested:

- AuthControllerTest
- CheckStatusControllerTest
- LabAdminControllerTest
- LabControllerTest
- ProcessorControllerTest
- RequestSpaceAndServersControllerTest
- ServerControllerTest

Focus Areas:

- Correct mapping of endpoints (@GetMapping, @PostMapping)
 - Handling of successful and failed requests
 - Validations and error messages
-

2. Service Layer Tests

The service layer tests ensure that core business logic is properly executed. Mock repositories or dependencies are used to simulate interactions.

Services tested:

- CheckStatusServiceTest
- LabAdminServiceTest
- LabServiceTest
- ProcessorServiceTest
- RequestSpaceAndServersServiceTest

- ServerServiceTest
- UserServiceTest

Focus Areas:

- Correct delegation to repositories
 - Business logic correctness
 - Exception handling
-

3. Utility Class Tests

The JwtUtil class, responsible for token creation, validation, and claim extraction, has its own unit test.

Utility tested:

- JwtUtilTest

Focus Areas:

- Valid JWT generation
 - Token validation logic
 - Extraction of email and role from the token
-

Test Directory Structure:

Located in:

`src/test/java/com/example/demo/`

Sub-packages:

- controller
 - service
 - util
 - config
-

Tools & Frameworks Used:

- JUnit 5 (JUnit Jupiter)
 - Mockito for mocking dependencies
 - Spring Boot Test for loading application context
-

Note:

All unit tests are implemented and committed to the GitHub repository before the given deadline (Monday).

GITHUB REPO IS : <https://github.com/Elisala789/SE--LAB>