Test Plan Identifier

ID: [HH1 – Team Aguacate] Rev.1.0
Type: Unit Test
Plan Performer: Daniel Santiago

References

- Project Proposal Document
- System Design Document

Introduction

This test plan document will outline the procedure to perform unit testing and validation of the functionality of the individual software modules of Team Aguacate's Panda Code Reviews. This document contains information about testing procedure for the Backend Modules.

Test Items

- Users
  - Create User
  - Login as user
  - Get created user
  - Get multiple users
  - Get user with courses, assignments and submissions
  - Delete user
- Courses
  - Create Course
  - Get Course
  - Get Courses
  - Get Course with Users, assignments and submissions
  - Add student to course
  - Remove user from course
  - Delete course
- Assignments

  - Create Assignment
  - Get Assignment
  - Get Assignments
  - Get assignment with submissions
  - Add test case
  - Delete test case
  - Delete Assignment
- Submissions
- Testing Framework
  - Run Single File Source Tester
  - Run Multiple Files Source Tester
  - Run Multiple Files Source Tester against multiple test cases and provide verdict based on expected result.

Risks Issues

Compiling and running unsecure code requires sandboxing, the machine performing the testing procedure must be properly configured with the 'jail' environment for the tests designated to the Testing Framework to be successfully correct. Solution involves writing an upstart script that configures the machine on boot.

Features to be tested

- Users
  - Create User
  - Fail creating user with missing payload
  - Fail creating user with email taken
  - Fail logging in with bad username
  - Fail logging in with pad password
  - Login as user
  - Get created user

    - Get multiple users
    - Get user with courses, assignments and submissions
    - Delete created user
    - Fail deleting already deleted user
    - Fail getting deleted user
- Courses
    - Create Course
    - Fail creating course with missing payload
    - Get Created Course
    - Get Multiple Courses
    - Get a course with its users, assignments and submissions
    - Add student to course
    - Get course and check student was added
    - Fail adding student to unknown course
    - Fail adding unknown student to course
    - Remove user from course
    - Fail removing user from unknown course
    - Get course and check student was removed
    - Delete course
    - Fail deleting already deleted course
    - Fail getting deleted course
- Assignments
    - Create Assignment
    - Fail creating assignment with missing payload
    - Get created assignment
    - Get multiple assignments
    - Get an assignment with its submissions

  - Add test case to assignment
  - Check test case was added to assignment
  - Delete test case
  - Check test case was deleted in assignment
  - Fail creating test case without payload
  - Create test case with files
  - Delete Assignment
  - Fail deleting already deleted assignment
  - Fail getting deleted assignment

- Testing Framework
  - Single file correct verdict
  - Single file wrong answer verdict
  - Compile Error verdict
  - I/O piping test
  - Runtime Exception verdict
  - Time limit exceeded verdict
  - Memory limit exceeded verdict
  - Multiple files correct verdict
  - Multiple files wrong answer verdict
  - Queue test

Features not to be tested

- Submissions – this module depends on integration with repository management system and frontend.

Approach

General Description

The tests will consist of performing various HTTP requests to the backend and analyzing their responses against expected behavior responses. The procedure is included in the test procedure plan.

Tools

- **Terminal:** Needed to run the test script
- **Node.js:** The tests are written using node
- **Chai:** For the assertions
- **Mocha:** Testing framework
- **Text Editor (Sublime Text):** To write the tests.
- **MongoDB:** Inspect Database content
- **Postman:** Performs HTTP requests.

Item Pass/Fail Criteria

For the tests to be performed in the following plan, a feature will ***pass*** the test ***only if the test results match the expected behavior***. If the test result differs from the expected result the test will be considered as ***failed***.

Test Deliverables

This test plan will deliver a test procedure plan which will describe the test routine needed to test the individual modules which were described on this document.

Remaining Test Tasks

For the remaining of the project, the remaining tasks will consist of system integration and integration testing of all modules.