**Test plan for**

**<<project name>>**

***Note that you can refine your testing plan as the project development goes. Keep the change log as follow:***

*ChangeLog*

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Change Date** | **By** | **Description** |
| version number | Date of Change | Name of person who made changes | Description of the changes made |
| 1 | 11/3/2022 | Aaron Nguyen |  |
| 2 | 20/4/2022 | Aaron Nguyen | Overhaul of test plan: update unit tests, integration test, acceptance test, load testing |

# Introduction

## Scope

Scope defines the features, functional or non-functional requirements of the software that **will be** tested.

Functional requirements:

1. Sign up/Log in
2. Update/change user profile
3. Adding courses
4. Modify/Add grade
5. Add sticker tasks
6. Modify/add flashcards

Non-functional requirement:

1. Load testing
2. Errors should be reported to users

## Roles and Responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Net ID** | **GitHub username** | **Role** |
| Abu Sayeed Khan |  | sayeedkhannabil | Developer |
| Aaron Nguyen |  | prosaox | Developer |
| Thomas Livingstone |  | thomaslivingstone | Developer |
| Hasin Ishrak |  | ishrakHP | Developer |

# Test Methodology

## Test Levels

Unit/Integration test

Core feature 1: Sign up/ Log in

* User with unique email, long enough password and a name can sign up
* All other cases: wrong format or available email, not long enough password, no name cannot sign up
* User with correct log in information can sign in
* User with wrong email and password cannot sign in

Core feature 2: Update/change user profile

* User enters format can change user profile
* User enters wrong format cannot change user profile

Core feature 3: Add/remove/modify course

* User enters correct format can add course successfully
* User enters wrong format cannot add new course
* User can remove specific course
* User can modfiy course information

Core feature 4: Add/Modify grades

* Users can add new test/exam
* Users can remove new test/exam

Core feature 5: Add sticky tasks

* Users can add new tasks
* Users cannot add new tasks with wrong format

Core feature 6: Add/Modify flashcards

* Users can add new flashcards with corect format
* Users can remove flashcards

Acceptance test:

Core feature 1:

* On login page click register
* Enter valid email, a name and a password larger than 6 character, click register
* Log out
* Enter the same email and password, click log in

Core feature 2:

* Log in using feature 1
* Click profile on navbar
* Edit your information
* Click Home to see information has changed

Core feature 3:

* Log in using feature 1
* Navigate to course using navigation bar
* Add a new course with random name and description
* Check the course page update

Core feature 4:

* Log in using feature 1
* Navigate to course using navigation bar
* Create new course using feature 3
* Click grade tracking button in course card
* Add a new test with required information
* Check the exam page update

Core feature 5:

* Log in using feature 1
* Navigate to course using navigation bar
* Create new course using feature 3
* Click View detail button in course card
* Add a new task with required information
* Check the page update

Core feature 6:

* Log in using feature 1
* Navigate to course using navigation bar
* Create new course using feature 3
* Click flashcard button in course card
* Add a new flashcard with required information
* Check the page update

## Test Completeness

Criterias for test completeness:

* Automated testings should be executed
* All the bugs and failed test should be checked and fixed regularly
* The test should cover 80% of the codebase
* Bugs should be fixed and reported

# Resource & Environment Needs

## Testing Tools

The following tools are being used in this project for testing:

* Automation: Jenkins
* Backend test: Jest, Supertest
* Frontend test: Jest
* Load test: Jmeter
* Other: Postman

## Test Environment

1. Windows 7 and above
2. Jenkins
3. Docker

# Terms/Acronyms

Make a mention of any terms or acronyms used in the project

| TERM/ACRONYM | DEFINITION |
| --- | --- |
| API | Application Program Interface |
| AUT | Application Under Test |