Thutong LMS
Testing Document

Edited by:

Fiwa Lekhulani
Daniel Rocha
lebogang Ntatleng
Lesego Mabe
Tlou Lebelo
Oluwatosin Botti

Contents

1 Definition of Testing

1.1 Purpose

Testing serves as the basic measure of progress within the softaware development process. Therefore without testing software throughout the develpment cycle there is no form of progress which prove that developers have made an attempt to solving the problems specified by the customer or user.

Also, in situations where products need to ensure compliance with regulatory requirements, software testing can safeguard the organization from legal liabilities by verifying compliance. Such as protecting user data.

1.2 Goal

The main goal to testing is ensuring that Thutong LMS system software fulfills its requirements as stipulated in the systems requirement documentation. The requiements to be meet include both functional and nonfunctional requirements which might way down the performance, security or any other system qualities.

2 Description of the test process

Test-Driven Developement (TDD) would serve of as our testing machanism which would be applied throughout the development cycle to allow a continous testing. This in turn allows a better measure to the development progress.

2.1 Prepare for test driven development.

Team members are to fully understand the system functionality they are expected to implement before they start implementation. This should be demonstrated through UML diagrams.

- Sequence Diagram.
- Activity diagram
- Class diagram.

2.2 Write tests.

Testing code is implemented from the existing UML structures as a guide. Therefore no actual implementation would be implemented before testing is established.

2.3 Implement and test the features.

After implementation of any artifact such as classes a test should be conducted from the test that was implemented prior.

2.4 Repeat until all features are correctly implemented.

Developers repeat the process for all the artifacts and subsystem modules.

2.5 Accomplish test coverage.

Developers check and review the testing scope of the subsystem or artifact, to be able to merge it with other parts of the system.

3 Test Evaluation

Use Case-Based Testing serve as the testing objective throughout the testing phases of the systems life cycle. This allows the team to measure progress.

Table 1: UC1: Register a New User

Actor: New User	System: Web Application
User link	0. System displays homepage with a Register
1. User fills in the login ID (email or Facebook), password, retyped password, and Click Submit button	2. System displays a new User Registration Form with options. 2.1. Register via email 2.2. Register via Facebook
	3. System verifies the login ID and passwords, 3.1. Display the Registration Success page, or 3.2. Display an error massage and ask the user to try again.
4. User sees the Registration Successful page.	

Table 2: UC2: Login User

Actor: User	System: Web Application
	0. System displays homepage with a login User 0.1. Login via email 0.2. Login via Facebook
1. User fills in the login ID, password and Click Login button	2. System verifies the login ID and passwords 2.1. Display the Login Success page, or 2.2. Display an error massage and ask user to try again
3. The user sees the Login Successful page	

Table 3: UC3: Write Quiz

Actor: User	System: Web Application
0. User browse a coarse content topic	1. System displays a subtopic page with a Write Quiz button
1. User click the Quiz button	2. System Displays the quiz (and allow user interaction) with a submit or exit button.
3. The user write the quiz and submit	4. System marks the quiz and return, accumulated score.
5. User receives accumulated results	

4 Quality Level to be achieved

4.1 Performance

- Able to load under weak network signal i.e. 2g network.
- Less data bandwidth required to load and browse.

4.2 Availability

- System functionalities must be up and available at all times 24 hours a day and 7 days a week. this includes the server, database and browser web interface.
- Aways accessable through web browser dispite the type of device be used. Whether desktop or mobile devices.

4.3 Usability

- The system must be efficient to use: takes less time to accomplish a particular task. This sould take less than a minute
- Easier to learn: operation can be learned by observing the object i.e. the names of different functionalities is intuitive.
- More satisfying to use: student users comfort in using the system for educational purposes i.e. the user friendly

5 Approach to Test Process Improvement

Travis CI is well enough tool to aid with the TDD. It Why travis CI? Travis ci provides a syncing functionality to github. This allows the Team (Digital BlackSmiths) to use whats already in place without having to learn new optimizing tools which might make software testing a baden.