

Virtual Classroom

Test Plan

COP 4331C, Fall, 2015

Modification History

Version	Date	Who	Comment
V 0.0	10/01/2015	J. Casserino	Created Initial Draft
V 0.1	10/06/2015	J. Casserino	Updated Rough Draft - Added the Introduction - Added Description of Test Environment
V 0.2	10/08/2015	J. Casserino J. Bender	Updated Rough Draft - Added Overall Stopping Criteria - Description of Individual Test Cases
V 1.0	10/08/2015	J. Casserino	Check for errors and updated Rough Draft to Final Draft

Team Name: Group 26

Team Members:

Joseph Bender jbender94@knights.ucf.edu

Joshua Casserino Joshua.casserino@knights.ucf.edu

Chad Armstrong chad.armstro@knights.ucf.edu

Miles Friedman milesfriedmanfl@gmail.com

Contents of this Document

Introduction.....	3
Overall Objective for Software Test Activity.....	3
Reference Documents.....	3
Description of Test Environment.....	3
Overall Stopping Criteria.....	4
Description of Individual Test Cases.....	4

Introduction

Overall Objective for Software Test Activity

The initial software tests are to make sure that we have a working and bug-free application before issuing the software to customers. We hope to identify any flaws or possible improvements in the program during these tests in an effort to make the product release ready.

Reference Documents

Concept of Operations;

github.com/bender-joe/VirtualClassroom/tree/master/Documetation/Deliverable%201

Project Plan;

github.com/bender-joe/VirtualClassroom/tree/master/Documetation/Deliverable%201

SRS;

github.com/bender-joe/VirtualClassroom/tree/master/Documetation/Deliverable%202

Description of Test Environment

An iPhone with a current operating system and internet access will be used for testing. The testers will be team and non-team members, who are non-developers, to use the software objectively without knowing all the ins-and-outs of the implementation. The developers will perform tests in developed functionalities and performance, to test individual features, before beta testing. The environment for development will be the same as the live environment.

Stopping Criteria

During development of the application, the developers will be running their own unit tests as they develop to increase the quality of the code produced. During the testing phase, the team will test the software logging all defects and giving them a rank of severity in relation to the usability of the application. As issues and bugs are reported, the developer(s) will immediately begin addressing the highest severity issues and form a resolution plan to be approved with the remainder of the team before implementation. This will ensure that a comprehensive solution to the issue is to be implemented.

Multiple test cases will be run, feeding in 90% of all possible permutations of variable input for each functionality and component of the system. Once these test cases have all passed without error and the system is observed to behave correctly in all scenarios, then the software may be deemed ready for delivery. This will require that there are no known errors within the system. The test cases will involve completing all tasks for each functionality in the application and passing all permutations of possible input to the applications.

Description of Individual Test Cases

Test Case 1 – Login

- **Test Objective:** This will test the functionality of logging in, and reject or grant access to the system based on input credentials.
- **Test Description:** The login function will have inputs that contain both false and valid username and password combinations. A test user name of TestUser@email.com and a password of Passw0rd in order to test login

acceptance. Invalid combinations involving all permutations will be used to test login rejection.

- Test Condition: This is not applicable as this functionality is under a general mode for the application.
- Expected Results: For valid test username and password combinations that are in the database the system will log the user in and present the home page to them. Invalid usernames and passwords will be rejected further access to the system.

Test Case 2 – Create User Account

- Test Objective: Testing the functionality of allowing the user to input information and create an account.
- Test Description: Create account function will receive input with all ranges of data. The only character string of information that will require validation are the password and email fields. The test will provide both valid and invalid passwords and an email addresses to test the applications data verification functionalities. Once the Teacher or Student user request a new account the Admin user will either approve or deny the request.
- Test Condition: The application will be in a neutral mode as this functionality is available to both Student, Teacher and Admin users
- Expected Results: When the system receives valid information an account will be created and the user will receive verification that the account has been created. If the information provided is invalid against the constraints the user will receive notification that the account could not be created.

Test Case 3 – Privacy of personal data

- Test Objective: This test will check the application's security.
- Test Description: A Student user will attempt to access other Student users personal info (Email address, Grades. Etc.).
- Test Condition: The application will be in a Student mode as this functionality only requires Student users.
- Expected Results: The system will not allow the personal data to be accessed.

Test Case 4 – Create Course and Content

- Test Objective: Testing the functionality for Teacher and Admin users to create a course module and populate it with relevant content.
- Test Description: The test case will provide input into the content fields when creating a course. The data validation functionalities will be tested with invalid and valid data strings in fields such as Course Name and Course ID. This test case will also provide testing for publishing content into the files and assignments sections of the course.
- Test Condition: This test case will require the mode of Admin and Teacher users.
- Expected Results: The application should create a course based on the input from the 'create course' function and if valid character strings are received for the course code, the application will create a new course with all provided data.

Test Case 5 – Student Select and Add Courses

- Test Objective: Testing the functionality for Students to select an available course and add it to their inventory of courses.
- Test Description: The test case will simulate a Student user searching and adding a course to their available course list. The system will then inform the Teacher

user of that Student is trying to access the course. Teachers can accept or deny access to that course.

- Test Condition: This test case will require the mode of Students and Teacher users.
- Expected Results: If the Teacher user allows the application should add the course to the Students available/current courses list otherwise the application will reject the request.

Test Case 6 – Students Submit Assignments

- Test Objective: Testing the functionality for Students to view and upload assignments available for that course.
- Test Description: This test case will simulate a Student user selecting to view available assignment. It will also test if the Student will be able to upload completed assignments to the course.
- Test Condition: This test case will require the mode of Students users.
- Expected Results: The application should allow the user access to view and upload assignments.

Test Case 7 – Student have access to course data

- Test Objective: Testing the functionality for Students to have access to an available course.
- Test Description: The test case will simulate a Student user selecting to access the course and the course's material of a previously approved course.
- Test Condition: This test case will require the mode of Students users.

- Expected Results: The application should allow access to the available course and its available content.

Test Case 8 – Teacher Create/Grade Assignments

- Test Objective: Testing the functionality for the Teacher user to create an assignment for a course.
- Test Description: The test will simulate a Teacher user uploading an assignment into a course and inputting requirements (i.e. Due Date, Date Available, File type, etc.).
- Test Condition: This test case will require the mode of Teacher users.
- Expected Results: The application should upload the assignment and give Students currently in the course access to the relevant content.

Test Case 9 – Creating an Interactive Poll

- Test Objective: Testing the functionality for Teacher users to create an interactive poll for that course.
- Test Description: The test case will simulate a Teacher user selecting to view create an interactive poll. Once the poll is created a Student user will interact with poll.
- Test Condition: This test case will require the mode of Students and Teacher users.
- Expected Results: The application should create a poll and allow the Student user access to view content.

Test Case 10 – Student Post/Create Discussion

- Test Objective: Testing the functionality for Student users to create a discussion and/or post to a current discussion for the courses.
- Test Description: The test will simulate Student users creating a new discussion and/or post to a current discussion attached to that course.
- Test Condition: This test case will require the mode of Student users.
- Expected Results: The application should upload the discussion and/or discussion post and give Students currently in the course access to the relevant content.