



University at Albany

Project Name	ICSI418-Group-Project
Release Version	1.0

Testing Plan and Results

Date: 05/02/2019

Team Members: Luke Prescott, Sean Loucks, Jack Holden, Max Moore, Chin Wa Cheung, Will Dahl, and Gary Passarelli.

Document Outline

Unit Testing	2
Introduction	2
Details Section	2
Integration Testing	9
Introduction	9
Details Section	9
Automated Testing	11
Introduction	11
Details Section(Planning Only)	11
System Testing	12
Introduction	12
Details Section(Planning Only)	12
Acceptance Testing	12
Introduction	12
Details Section	13

Unit Testing

Introduction

All tests in this section are focus on the implementation and detection of one single function. Each parts will show what the main functions of the test is, what the test methods are, and what the test results are.

Details Section

Test	Create Admin Account
Method	Use SQL try to insert an admin account in database with account name admin@gmail.com and password csi2019.
Result	Select from admin table in MySQL then can see the admin account exist.

Test	Login by Admin Account
Method	In login page, try to login with admin@gmail.com by password csi2019.
Result	Successful login, access to the administrator interface.

Test	Logout from Admin Page
Method	click Log-out links in the upper right corner to check logout function
Result	Successful logout, Back to Login page.

Test	Access Admin Management Page.
Method	In admin page, click Admin Magement Button.
Result	Successful access admin management page. BUG: Admin list table display incorrectly, email shows in ID column,

	password shows in Email column, edit link shows in password column.(FIXED)
--	--

Test	Create Admin Account - 2
Method	In Admin Management page, try to create two new admin accounts, one with new email address, one with an existing email addresses in the database.
Result	New address one create in database and show in web page successfully, existing one return warning and reject create operation.

Test	Admin edit function
Method	In Admin Management page, choose one admin account and use it's edit link to change it's password.Then try to delete this account.
Result	New password insert to database successfully. Then successfully remove from database.

Test	Go Back and Logout link in Admin Management page
Method	Try to use go back link to return Admin Page from Admin Management Page, and try to use Logout link to Logout from Admin Management Page
Result	Both functions works well.

Test	Access User Management Page.
Method	In admin page, click User Magement Button.
Result	Successful access user management page.

Test	Create User Account
Method	In User Management page, try to create two new user accounts, one with new email address, one with an existing email addresses in the database.
Result	New address one create in database and show in web page successfully, existing one return warning and reject create operation.

Test	User edit function
------	--------------------

Method	In User Management page, choose one user account and use it's edit link to change it's password. Then try to disable this account. Finally delete this Account.
Result	New password insert to database successfully. Then successfully disabled this account's useability. Then successfully remove from database.

Test	Go Back and Logout link in User Management page
Method	Try to use go back link to return Admin Page from User Management Page, and try to use Logout link to Logout from User Management Page
Result	Both functions works well.

Test	Login by User Account
Method	In login page, try to login with active @gmail.com by password csi2019.
Result	Successful login, access to the User interface.

Test	Logout from User Page
Method	click Log-out links in the upper right corner to check logout function
Result	Successful logout, Back to Login page.

Test	Access User Update Page.
Method	In User page, click Update Your File button.
Result	Successful access User Update page.

Test	User Information Update function.
Method	In User Update page, try to change confiren email address, password, and finally delete this account.
Result	All New change insert to database and can use to login, finally user's account remove from database successfully.

Test	Access Question Management Page.
Method	In admin page, click Question Magement Button.
Result	Successful access Question management page. Al questions show in web page.

Test	Create New Multiple Choice Question
Method	In Question Management page, try to create a new Multiple Choice Question.
Result	New Question insert to database and display on web page successfully.

Test	Create New Multiple Choice Question with Image
Method	In Question Management page, try to create a new Multiple Choice Question which has a image.
Result	New Question insert to database and display on web page successfully.

Test	Multiple Choice Question edit function
Method	In Question Management page, choose one Multiple Choice Question and use it's edit link to change it's description, answer, correct answer, and image.
Result	All New change insert to database and display on web page successfully.

Test	Create New True-False Question
Method	In Question Management page, try to create a new True-False Question.
Result	New Question insert to database and display on web page successfully.

Test	Create New True-False Question with Image
Method	In Question Management page, try to create a new True-False Question which has a image.
Result	New Question insert to database and display on web page successfully.

Test	True-False Question edit function
Method	In Question Management page, choose one True-False Question and use it's edit link to change it's description, answer, correct answer, and image.
Result	All New change insert to database and display on web page successfully.

Test	Delete Questions function
Method	In Question Management page, try to delete one Multiple Choice Question

	and one True-False Question.
Result	Selection questions successfully remove from database

Test	Go Back and Logout link in Question Management page
Method	Try to use go back link to return Admin Page from Question Management Page, and try to use Logout link to Logout from question Management Page
Result	Both functions works well.

Test	Access Test Management Page.
Method	In admin page, click Test Magement Button.
Result	Successful access Test management page. All exsit Tests show in web page.

Test	Create New Test
Method	In Test Management page, try to create a new Test.
Result	New Test insert to database and display on web page successfully.

Test	Create New Test with Image
Method	In Test Management page, try to create a new Test which has a image.
Result	New Test insert to database and display on web page successfully.

Test	Test edit function
Method	In Test Management page, choose one Test and use it's edit link to change its Title, Header, Footer, Available questions, Due Date and Image.
Result	All New change insert to database and display on web page successfully.

Test	Delete Test function
Method	In Test Management page, try to delete one Test.
Result	Selection Test successfully remove from database

Test	Test Invite_user(s) function
Method	In Test Management page, try to use invite_user(s) link of one Test to invite user to take test..
Result	Selection Test successfully shows in target users web page and ready to be taken.

Test	Go Back and Logout link in Test Management page
Method	Try to use go back link to return Admin Page from Test Management Page, and try to use Logout link to Logout from Test Management Page
Result	Both functions works well.

Test	User Take_test function
Method	In User page, try to use take_test link take a Test which signed by admin.
Result	Successfully access Exam Page.

Test	Exam page function
Method	Try to answer each question in Exam Page, then submit them.
Result	Successfully submit Exam and get Test result page as response..

Test	User view_result function
Method	In User page, try to use view_result link to check one specific taken test's result.
Result	Successfully access Result Page and details are display in web page.

Test	Go Back and Logout link in User Test Result page
Method	Try to use go back link to return User Page from Test Result Page, and try to use Logout link to Logout from Test Management Page
Result	Both functions works well.

Test	admin view_result function
Method	In admin page, try to use view_result link to check one specific taken test's result.

Result	Successfully access Result Page and details are display in web page.
--------	--

Test	Go Back and Logout link in User Test Result page
Method	Try to use go back link to return Admin Page from Test Result Page, and try to use Logout link to Logout from Test Management Page
Result	Both functions works well.

Test	Access Upload File Page.
Method	In admin page, clickFile Upload Button.
Result	Successful access Upload File page.

Test	Upload File function
Method	Try to update questions by upload CVS file
Result	Questions insert in database successfully, and display in web page.

Test	Go Back and Logout link in Upload File page
Method	Try to use go back link to return Admin Page from Upload File Page, and try to use Logout link to Logout from Upload File Page
Result	Both functions works well.

Integration Testing

Introduction

After the unit testing was performed the next level of testing was integration testing. Integration testing is an important aspect of the testing cycle as it not only confirms that each unit has been tested but also verifies that all the units work in unity.

While making our testing plan we found that using an incremental approach would be the best strategy. This is because we would have instances throughout the project where stubs and drivers would be used to implement our integration testing. Although big bang testing seemed like a viable option if we had a well documented plan, it was just too risky and we didn't want to wait until all of our classes/modules to completed to test. Most of the details section is confirming that these classes and links are working properly.

Details Section

Test	Check the link between the login and Admin page.
Method	Enter login credentials and click on the login button.
Expected Result	To be directed to the Admin page.

Test	Check the link between the Admin page and use management page.
Method	Click the the user management button.
Expected Result	To be directed to the user management page.

Test	Check the interface link between the create user and email.
Method	Enter new user credentials and create user.
Expected	User sent login credentials via email.

Result	
--------	--

Test	Check the link between the admin create test page and both admin and user page.
Method	Create a new test.
Expected Result	The newly created test should appear on both the admin and user page.

Test	Check the user testing and the change in the user homepage
Method	Start and submit a test.
Expected Result	The newly submitted test should be graded and appear on the user and admin homepage.

Test	Check the link between create test page and image upload
Method	Click the image upload link
Expected Result	To be directed to the image upload page and be able to upload an image.

Test	Check the link between create question page and image upload
Method	Click the image upload link
Expected Result	To be directed to the image upload page and be able to upload an image.

Test	Check the interface link between the login and User page.
Method	Enter login credentials and click on the login button.
Result	To be directed to the User page.

Automated Testing

Introduction

To make a project's testing extremely efficient, a common use is automated testing. It not only saves a lot of time but also eliminates any human error when testing. Automated testing tools-while plentiful these days-were not used in our project. In saying that, we'd like to provide some retrospective information about what types of automated testing we would have liked to use as well as how we would have implemented them in our project.

Details Section(Planning Only)

The first step in starting our automated testing plan would be to choose what testing tools to use. There is an extensive amount of testing tools out there and with the help of open source, there are a lot for free. We would have used some of the top testing tools like Ranorex, TestComplete, and Leapwork but we probably would have settled with QMetry because of it's built on Eclipse which is the IDE we used for development. After deciding on which testing tool we would then decide on what tests to automate. This basically comes down to how many times the test is repeated. Therefore, we would save the tests that are only performed a few times to manual testing and have the rest for automated testing. Once we establish these automated tests most of the job is done. By using automated testing it would free up some time for our testers which would allow them to conduct some more in depth manual testing or help out in another aspect of the project.

System Testing

Introduction

For cloud service programs, it should have the function of running normally on different type of system environments. So we plan to test our Cloud Applications on a variety of environments to ensure they functioning normal, including basic functionality, page UI display, and uploading or requesting data from the database.

Details Section(Planning Only)

We plan to visit our remote servers on a variety of major operation system to check that are basic functions functioning properly. These operation systems include Linux, Windows, macOS, Android and iOS.

We will login to administrator account and user account on these operating systems, and then perform their respective operations, such as setting up accounts, changing information, setting up problems, setting up examinations, accepting examinations, checking the results of examinations, check all UI display and so on.

Acceptance Testing

Introduction

Acceptance testing was the last level of testing completed before delivering the final product. Acceptance testing is usually completed either/both internally (by members of the organization that developed the software but who are not involved in development or testing) and/or externally (by customers of the organization that developed the software or the end-users of the software). The project was presented to the stakeholder weekly to make sure the developed functionalities were up to par with the stakeholder's requirements. Given that everyone in our group was involved in both development and testing, the traditional method of acceptance testing was not possible. Our group did, however, use ad-hoc black box testing, meaning a strict procedure was not followed. Although all of the members of our group had knowledge of the internal working of the system, the testing method was still black box because everyone tested parts of the system which they had not specifically developed themselves. Numerous tests were performed upon completion of the project based on the required functionalities section of the project PDF.

Details Section

Test	Administrators' creation, editing, and deletion of both a multiple choice and a true false questions with corresponding answers.
Method	Create a sample question of both a multiple and a true false question through the project. Attempt to edit it. Eventually, delete it.
Expected Result	The sample questions and answers entered should be displayed in the the question bank as well as stored in the database. Editing the questions should change the information edited when displayed in the test bank and viewed in the database. Deletion of the question should delete the question from both the displayed test bank and the database.

Test	Administrators' creation of both new users and new admins.
Method	Create a new user and admin with both consisting of an email and a password.
Expected Result	The sample user and admin created should receive an email with their login information, be able to use that login information to access the website as well as receive their corresponding permissions, and the login information should be stored in the database.

Test	Administrators' creation of tests through the selection of questions from the test bank.
Method	Attempt to create a test while logged in as an administrator, select the questions desired, and click create test.
Expected Result	The test should be displayed and available for view through the project as well as stored in the database.

Test	Administrators' creation, editing, and deletion of a question with an attached image.
Method	Create a sample question of both a multiple choice and a true false question and attach an image. Attempt to first edit the question and eventually delete it.
Expected Result	The question should be displayed in the test bank and stored in the database. Upon editing and viewing the question, the question should be

	displayed with the desired changes. The database should also be updated with the desired changes. Upon deletion of the question, the question should not be displayed in the test bank and should be removed from the database.
--	---

Test	Users' selecting and taking of a test.
Method	Log in as a user after inviting a user as admin to take a certain test. Select the test and take it, while verifying that progress is saved throughout.
Expected Result	The user should be able to select the test that he is invited to. Upon selecting the test, questions and options for answers should be displayed. Upon answering questions, selected answers should be saved, but not submitted, immediately. After submitting the test, the user should be able to see their score and the questions they got correct and incorrect.

Test	Users' viewing of tests taken
Method	Log in as a user after taking multiple tests. Select the tests from the Tests Taken display box and view each result.
Expected Result	The user should be able to view the results of a test he has previously taken. Upon viewing, the user should be shown a new page with their score as well as the questions they got correct and incorrect.

Test	Administrators' viewing of tests taken
Method	Log in as an administrator after multiple users have taken multiple tests. Select the tests from the Tests Taken display box and view each result.
Expected Result	The administrator should be able to view the results of a test a user has previously taken. Upon viewing, the administrator should be shown a new page with the user's score as well as the questions they got correct and incorrect.

Test	Administrators' viewing of tests created
Method	Log in as an administrator after creating multiple tests. Select the tests from the Tests Created display box and view each one.
Expected Result	The administrator should be able to see a listing of all the tests they have created. Upon selecting each one, the administrator should be shown a new page listing the questions in the test and their corresponding answers.

Test	Customizable header and footer for each administrator
Method	Log in as an administrator. Select the Update Style button and customize your header, footer, and header image. Repeat this with multiple administrator accounts.
Expected Result	Upon updating your header, footer, and header image, they should be displayed at the top and bottom of the page. When repeating this with a new administrator account with a different header, footer, and header image, the new style should be displayed. If an administrator chooses not to update their style, the default header, footer, and header image (UAlbany logo) should be displayed.

Test	Administrators' creation of a test through the upload of a .csv file
Method	Log in as an administrator. Select the upload file button and enter a title, header, and footer for the test. Upload a properly formatted .csv file.
Expected Result	Upon uploading the file, a functional test should be displayed in the Tests Created bank with the title, header, and footer you entered, as well as stored in the database. Upon viewing the test, the questions and answers from the .csv file should be displayed.