## Acceptance Testing

## Group G

CITS3200 Professional Computing

Semester 2 2013

University of Western Australia

Crawley, WA 6009

## Table of Contents

Objectives  
Document References  
Test Summary  
Testing Strategy

User Authentication Testing

Text and Task Item Test  
Email Notification Test  
Navigation and Content Pane Test

Testing Materials

## Objectives

Tests are written so that a developer or client looking at them will know what output should be returned given some inputs. This will help speed up development as it gives a clear plan for the developers and will also be beneficial during the debugging phase if something goes wrong. A client will also be able to read this document and see what the expected outputs should be when the site is used. These tests will also be performed by the client during project handover to ensure that all of the required functionality is present and accounted for. There will be ongoing changes within the document as the project advances and changes are made to the features set.

## Document References

We will use section 3.2 of the Requirements Analysis Document as a guideline for making tests that cover all the requirements. Requirements which have high value and are of greater importance, namely the text item, email notifications, user authentication and user interface, will have more extensive tests while tests for low value requirements will be less detailed.

## Test Summary

To meet the requirements the client has listed there will be tests for:

* User authentication to ensure that only authorised users can access the site to view and edit the contents.
* Adding/editing/deleting of task items which is the main purpose of this site.
* Email notification for events that are occurring and tasks that are due.
* Navigation to ensure that there is a consistent look to the site.

Tests for the copy/move items and log user activity requirements are left out for now as the group has deemed them to be too difficult and time consuming at the moment. If there is additional time to complete these requirements, tests will also need to be written for these.

## Testing Strategy

Using Ruby on Rails allows us to use tools like Rspec which can automate testing for us. Using the ideas of Test Driven Development, tests should be written first then the code which implements the functionality that is being tested.

Having a set of well written automated tests ensures tests are consistently carried out each time. Emphasis is on the tests being well written as tests can only reveal that there are bugs and not that the application is bug free.

Throughout development these tests will be run when functionality is implemented to check that it is working and also to see if adding that functionality does not break the rest of the system. Tests also speed up development by allowing someone new to the project to easily see what parts have been tested and what parts still needs work.

The following test will be written in a way that assumes the reader is the user.

## User Authentication Test

A functional user authentication system should prevent unauthorised users from accessing the site. We want this so that only the club committee members are allowed to view and edit the site's content.

### Test Specification

This test will verify the requirement that the user authentication system works. The parts that will be tested are the login and logout actions. Tests will simulate what happens if the user is authenticated and also what happens if the user fails authentication. We will assume that the user has stable internet access and that the server hosting the web application will be functional when the user authentication occurs.

### Test Description

The first test will ensure that the login functionality is working.

* At the login page, you enter a username and password and click on the login button.
* If you have provided proper authentication, you are logged into the site and redirected to the main page.
* Otherwise if you submit a blank form or enter incorrect details, you are presented with a message saying “Please enter a correct username and password” and remain on the login page.

The second test will demonstrate what happens if the user attempts to access the internal site via direct URL without logging in, in an attempt to bypass the authentication system.

* After logging in, the user copies the URL of a Task item and logs out.
* You then later try to access the URL directly, without logging in.
* You are presented with a message saying “Please enter a correct username and password” and remain on the login page.

The third test will assume that the user is logged in and will test the logging out functionality.

* Assume you are on a page where a logout button is present.
* Upon clicking the logout button you should be logged out and redirected to the main login page with a message saying “Successfully logged out.”

### Test Analysis Report

To be determined.

## Task Item Test

This test will evaluate the functionality of our task item type, ensuring that everything functions as described in our RAD document. The functionality of the task item is to allow the user to store, edit and delete information in an easy to read format, mark tasks as “inactive”, “incomplete”, or “incomplete”, as well as appending related task items to each other. If an incomplete tasks children are all set to complete should automatically set itself to complete. One example of a task item that will be repeatedly referred to throughout this document is to store information about an important upcoming event.

**Test Specification**

This test will look for the following functionality:

* The ability to append a new task item to any existing item, or the root.
* The ability to designate a task item as inactive, incomplete or complete
* The ability to edit the contents of any task item
* The ability to remove task items

### Test Description

The contents of the data is not particularly important, however it is important to ensure that the data is stored and displayed exactly as it is entered.

Presume that the user begins logged into the web interface as described in the user authentication test:

* You click on the create task item button and fields for the name of the item and its contents appears on the contents pane
* You click on a field and can enter text, and select from inactive or incomplete.
* When the field is no longer in focus the text you entered should still be in the field.
* If you click the delete button, the item and its contents should no longer be there.

These tests ensure that the functionality of the task item type is working as performed. If any of the above tests fail then it is critical to determine and correct the cause of failure as soon as possible. When determining what is an acceptable delay for updating the back-end database, the user should not be left waiting for any extended amounts of time. Certainly any longer than 10 seconds would be considered a failure under normal conditions.

### Test Analysis Report

To be determined.

**Email Notification Test**

A functional email alert system will send out emails when upcoming events are near and when incomplete tasks are due.

**Test Specification**

The requirement that there are date triggers associated with tasks will be tested with this set of tests. This will be handled automatically by the web application after the user sets the dates for the events and tasks. We will assume that both the user and the server the web application is hosted on are set to the same time zone. Another assumption is that the user is already logged in.

**Test Description**

This test assumes that you are starting at the main page after logging in.

* You select or create the task that you want an email notification for.
* You click on a field that lets you enter when an task occurs.
* Clicking on another field lets you fill out when an email notification should be sent out.
* On the notification date, you should receive an email with the task name in the subject heading and its contents in the body.

With the tasks there is an additional feature to mark a task as complete:

* Assume you have set a date for a task.
* Before the task due date, you complete the task and mark the task as complete.
* The task should be marked as complete and you should not receive any notification.

If you do not set a due date for an task then you should not receive any emails for that task.

### Test Analysis Report

To be determined.

## Navigation and Content Pane Test

A functioning Navigation Pane is crucial for guiding the user around the site and allowing them to access existing events page. This, the navigation pane should be constantly updating to reflect the latest changes the user has made to any events. The contents pane will display all the text that goes with an event and the associated tasks.

### Test Specification

This test will ensure that the Navigation Pane is reliably and accurately updating itself to reflect any newly inserted events, changes to existing events or to accommodate the removal of events. This should be handled by the web server whenever necessary.

This test will test all functionality of the Navigation Pane, including:

* Ensuring that the navigation pane automatically updates itself whenever an event is created, edited or deleted.
* Clicking on any event on the navigation pane will transfer the user to the corresponding event page.

The tests for the Contents Pane are already covered for in the tests for the Text and Task Item tests.

### Test Description

This test assumes that you are on the main page after logging in. The first test will test that the navigation pane updates after creating a new event.

* You create a new task as described in the task item testing.
* The navigation pane will now contain an entry for the newly created event.

The second test will test both the ability to access event pages from the navigation pane and the ability to edit itself whenever an event is edited.

* You click on a task in the navigation pane which takes you to the description of that event.
* Editing the events name as outlined in the task item test automatically updates the navigation pane
* You decide instead to delete the task by clicking on a Delete button, which removes the event and its corresponding entry in the navigation pane.

### Test Analysis Report

To be determined.

## Test Materials

The test will be performed on both our private servers while we are working on the project and again on the live servers during handover to ensure that any changes to the hardware and software of both systems does not affect the functionality of the web application. Testing will be automated through the use of Rspec in addition to the manual testing of the site that will naturally happen during development. The use of Capybara which allows simulation of a user using a web browser and Cucumber which allows tests to be written in plain English will also be beneficial during development.