Software Engineering Group Project
Test Plan Including Test Log Form

Authors:	Tom Reed, Sam Jackson
Config Ref:	SE_N66_xxx_xx
Date:	2012-11-07
Version:	1.4
Status:	Draft

Department of Computer Science
Aberystwyth University
Aberystwyth
Ceredigion
SY23 3DB

CONTENTS

1. INTRODUCTION

The purpose of this document is to set out test that will help us to show that our program does what it is intended to do and to discover any defects before it is put into use.

2. PURPOSE OF THIS DOCUMENT

2.1 Scope

This document specifies the tests that will be run to discover various bugs and put general strain on the system. It will in detail outline the tests that will be run and the required outcome as well as what will constitute a failed test. This document will also outline what will be done with the test data and how it will be reported.

2.2 Objectives

The objective of this document is to outline the tests that will be run on the system, how they will be structured and how they will be documented and reported.

3. TEST PLAN

All coders will be responsible for testing their code, code will be tested using a set of JUnit tests. As the code has not been written yet it is impossible to stipulate at this time what the tests will need to be. HTML and CSS will be tested using the W3C validators as well as tested in various browsers that support HTML5. For the system testing, we need to test it through the client side as there is no real way access the server side of the system without using JUnit testing which has already been stipulated.. So the only way to approach it is to test if the server side does as we expect from the client side.

3.2 Test Specification

Test Ref	Req. Being Tested	Test Content	Input	Output	Pass Criteria
SE-N17-001	FR1, FR7	Check that a valid user can login to the system.	Input on login page for existing user. User-name: slj11@aber.ac.uk Valid password for this user: mypassword submit button pressed	User authenticated onto the system and redirected to their profile page. New session for this user started on server.	User is authenticated onto the system.
SE-N17-002	FR1, FR7	Check user names that are not emails are not admitted to the system.	Input on login page for existing user. User-name: slj11 Valid password: mypassword	Error message warns user that email isn't valid.	User is not authenticated. Error message shown.

SE-N17-003	FR1, FR7	Check passwords of length less than 8 are too short.	Input on login page for existing user. User-name: slj11@aber.ac.uk Password: mypassw	Error message warns that password is invalid.	User is not authenticated. Error message shown.
SE-N17-004	FR1, FR7	Check that passwords of length greater than 20 are too long.	Input on login page for existing user. User-name: slj11@aber.ac.uk Password: mypasswordqwertylopwq	Error message warns that password is invalid.	User is not authenticated. Error message shown.
SE-N17-005	FR1, FR7	Check a user with a valid email, but not on the system is not authenticated.	Input on login page for existing user. User-name: xxx@aber.ac.uk password: mypassword	Error message warns that the user is not registered on the system.	User is not authenticated. Error message shown.
SE-N17-006	FR1, FR7	Check a user with a valid email but incorrect password is not authenticated.	Input on login page for existing user. User name: slj11@aber.ac.uk password: mywrongpass	Error message warns that the user entered the wrong password.	User is not authenticated. Error message shown.
SE-N17-007	FR1, FR7, FR6	Check that a new user can be registered on the system.	Input on login page for new user form. Username: newuser@aber.ac.uk password: newpassword password check: newpassword	User added to system. Given starting monster and redirected to there profile page.	User is created in the system, has a new monster created and is authenticated and directed to their profile page.
SE-N17-008	FR1, FR7, FR6	Check a new user name that is not an email is not registered	Input on login page for new user. User-name: newuser Valid password: mypassword password check: mypassword	Error message warns user that email isn't valid.	User is not created. Error message shown.
SE-N17-009	FR1, FR7, FR6	Check passwords of length less than 8 are too short.	Input on login page for new user. User-name: newuser@aber.ac.uk Password: mypassw Password check: mypassw	Error message warns that password is invalid.	User is not created. Error message shown.

SE-N17-010	FR1, FR7, FR6	Check that passwords of length greater than 20 are too long.	Input on login page for new user. User-name: newuser@aber.ac.uk Password: mypasswordqwertylopwq Password check: mypasswordqwertylopwq	Error message warns that password is invalid.	User is not created. Error message shown.
SE-N17-011	FR1, FR7, FR6	Check a user with a valid email that is already on the system is not created	Input on login page for new user. User-name: slj11@aber.ac.uk password: mypassword password check: mypassword	Error message warns that the user is already registered with the system.	User is not created. Error message shown.
SE-N17-012	FR1, FR7, FR6	Check that if the password that has been typed matches	Input on login page for new user. User-name: newuser@aber.ac.uk Password: mypassword Confirm Password: mypassss	Error message warns that the passwords entered do not match.	User is not created. Error message shown.
SE-N17-013	FR1, FR7	Check that a user can only access parts of the site if they are registered and authenticated.	Attempt to access all other pages of the site other than the login/register page.	Should be redirected back to the login/register page.	Unregistered/ Unauthenticated users are redirected back to the login/ register page.
SE-N17-014	FR7	Check a user can logout from the system.	Login to the system using a valid registered users details. Then logout using the logout button.	User is directed to the login/register page.	The users session is terminated and they can no longer access secure parts of the site.
SE-N17-015	FR2, FR8	User can access their friends page from their profile.	From the profile click the friends button.	User is taken to a list of their friends.	User is shown a list of their current friends.
SE-N17-016	FR3, FR8	User is shown a list of there current monsters and their stats on the profile page.	On the profile page a list of monsters is shown. Click/ highlight a monster to view it's stats.	The monsters stats are shown next to the monster.	Monsters stats are successfully shown the user.

SE-N17-017	FR4, FR8	User can view a list of notifications.	On the profile page click the notifications button.	User should be shown a list of current notifications for battle/breeding/ friend requests and fight/breeding results.	List of battle/ breeding/friend requests and fight/breeding results shown.
SE-N17-018	FR4, FR8	User can view a list of notifications.	On the profile page click the notifications button when the user has no pending notifications.	User should be shown a "no notifications" message	No notifications message shown.
SE-N17-019	FR4, FR8	For each type of notification (battle, breeding, friend requests and fight/ breeding results) check that the list is displayed correctly when the user has only one of these requests and no others.	On the profile page click the notifications button.	User should be shown a list of current notifications for battle/breeding requests and fight/breeding results.	List of battle/ breeding requests and fight results shown.
SE-N17-020	FR2, FR6, FR8, FR9	Check that accepting a friend request adds the friend to the users friend list.	Accept a friend notification by clicking accept.	User should be shown a message telling them that the request has been accepted.	Message shown and friend added to the friends list.
SE-N17-021	FR2, FR6, FR8, FR9	Check that the user can send a friend request.	On the friends page, click add friend. Email: test@aber.ac.uk Click send request.	User should be notified with a message that a request has been sent.	Message shown and request should be pending on server.
SE-N17-022	FR2, FR6, FR8, FR9	Check that the user can not add a friend that doesn't exist.	On the friends page click add a friend. Email: noone@aber.ac.uk Click send a request.	User should be notified that the user doesn't exist	Message should be shown. List of friend not changed.

SE-N17-023	FR2, FR6, FR8, FR9	Check that the user can not add a friend with an email address that is not well formed.	On the friends page click add a friend. Email: noone Click send a request.	User should be notified that the email address is invalid.	Message should be shown. List of friend not changed.
SE-N17-024	FR4, FR8	Check that a user can accept a battle request.	Accept a battle notification by clicking accept.	User should be taken to battle page and a battle should start.	User directed to battle page.
SE-N17-025	FR4, FR8	Check that a user can send a battle request.	On the friends page, click on the button to send a battle request to a friends monster.	User should be notified with a message that a request has been sent.	Message shown and request should be pending on server.
SE-N17-026	FR4, FR8	Check that a user can accept a breeding request.	Accept a breeding notification by clicking accept.	User should be taken to breeding page and breeding should start.	User directed to breeding page.
SE-N17-027	FR6, FR8	Check that user can send a breeding request.	On the friends page, click on the button to send a breeding request to a friends monster.	User should be notified with a message that a request has been sent.	Message shown and request should be pending on server.
SE-N17-028	FR3, FR4, FR10	Check that battles are executed correctly.	Accept a battle request. Users should be redirected to battle page.	Battle should run. The user can then view the results of the battle. Notification should also be sent to opponent to view the outcome.	Battle shown correctly. Both users are able to view the stats of the battle.
SE-N17-029	FR3, FR4, FR6, FR8	Check that breeding monsters is executed correctly.	Accept a breeding request. User should be redirected to breeding results page	Results of breeding should be shown. A notification of the breeding should be sent to friend linking them to the results of the breeding.	Breeding stats shown correctly. Both player are able to access the outcomes of breeding. Request sender gets the children and request acceptor gets money prize.
SE-N17-030	FR3, FR8, FR10	Check that the outcome of a battle can be viewed by the request sender.	After a battle, check that the request sender gets a notification and can view the outcome of the battle.	Notification should be viewable and clicking it will show the outcome of the battle.	Outcome of the battle is viewable by the user.

SE-N17-031	FR3, FR6, FR8	Check that the outcome of breeding can be viewed by the request sender.	After a breeding event, check that the request sender gets a notification and can view the outcome of the breeding.	Notification should be viewable and clicking it will show the outcome of the breeding.	Outcome of the breeding is viewable by the user.
SE-N17-032	FR6, FR8, FR9	Check that a friend request that has been sent can be cancelled.	Click cancel on a pending request.	Message shown that the request has been removed.	Request removed from the system and no longer displayed to either of the users.
SE-N17-033	FR6, FR8	Check that a battle request that has been sent can be cancelled.	Click cancel on a pending request.	Message shown that the request has been removed.	Request removed from the system and no longer displayed to either of the users.
SE-N17-034	FR6, FR8	Check that a breeding request that has been sent can be cancelled.	Click cancel on a pending request.	Message shown that the request has been removed.	Request removed from the system and no longer displayed to either of the users.
SE-N17-035	FR8, FR9	Check that a friend request can be rejected.	Click reject on a request	Message shows that the request has been rejected.	The request should be removed from the system, the other user should be informed that their request was rejected.
SE-N17-036	FR8	Check that a battle request can be rejected.	Click reject on a request	Message shows that the request has been rejected.	The request should be removed from the system, the other user should be informed that their request was rejected.
SE-N17-037	FR8	Check that a breeding request can be rejected.	Click reject on a request	Message shows that the request has been rejected.	The request should be removed from the system, the other user should be informed that their request was rejected.

SE-N17-037	FR11	Check that a rich list of the players friends can be viewed.	Go to the friends page.	A list of the richest friends should be showed	Rich list is shown and is accurate.
SE-N17-038	FR5	Check that tests SE-N17-[020-036] can be performed with a user who is not on this server.	Perform all test from 20-36 using a friend/friends that are not on our server.	All test should pass seamlessly as they do for local users.	All tests pass the same as for local users.
SE-N17-039	N/a	Clicking on the help button.	Clicking on the help button takes you the informative help page.	User shown the help page.	Help page shown.

3.3 Test Result Reporting

Test results will be recorded in a test log form (included on the next page). These documents will be stored in the groups Github repository under the folder “Test Data”, there will be two sub folders with in this directory called “Module Tests” and “System Tests”. The corresponding test documents will be stored into those two directories.

3.4 Configurations

Using Github we can branch off from the master copy of the project for testing and working on, doing this will insure that everyone is using the same version of the project. This is to make sure that everyone runs their tests on the same copy of the project insuring consistency. Branching off from the master copy will also allow for version roll back. Meaning that if a mistake is made and the system no longer works. We can return to a stable version where we know it does.

TEST LOG FORM	Test Log No:
Test ID:	Test Date:
Tester:	Group:
Version When Tested:	
Test Passed? (Y/N): Passed Testing 1st[ ],2nd[ ],3rd[ ],4th[ ] time. Other....................	
Description of test:	
Comments:	
The Following Section only applies if the first test failed and changes needed to be made. If changes need to be made a change control form needs to be completed documenting this. The details of that form will also need to be highlighted below.	
Change Control Number: Description of Changes Made:	

DOCUMENT HISTORY

Version	*CCF No.*	*Date*	*Changes made to document*	*Changed by*
1.0	N/A	07/11/2012	N/A - original version	thr5
1.2	1	07/11/2012	Remove un-used reference	thr5
1.3	2	08/11/2012	HTML Validations.etc	thr5
1.4	3	11/11/2012	Server Testing	thr5