Software Testing Plans

Maverick Applications Corporation:
Ryan Carlson, Stephen Pandorf, Kevin Rock

December 14, 2013

1.0 Introduction

This document lays out the testing plans to be brought out by our team when verifying that its implementation matches the requirements laid out in the specification document. The test plans will not include these requirements: R1.2, R2, R14. As we have discussed, these three requirements will not be implemented by our group. This leaves the agreed number of requirements to be tested at this stage. To test, we have laid out the testing requirements in section 2, and list the test cases in section 3.

2.0 Testing Requirements

To sufficiently test our program, we have produced testing requirements that will help us cover the requirements laid out in the requirements specification document. We will list the testing requirements in terms of the requirements that they satisfy in the requirements specification document. They will be written as TR#, where TR means testing requirement, and # indicates the requirement it satisfies from the specification document.

TR1: This will require that the user can successfully register with the system by using a UI to sign up. They should be able to enter data in all fields and have this data available in the model the next time they sign in.

TR3: This will require that the user is not allowed to sign up with the same username as the username of another registered account in the system. We will test this by attempting to sign up with the same username as another username registered in the system. At this point, we should be able to observe that the system did not allow us to register with that username and offer a suggestion.

TR4: This will require that, during user registration, the user is shown an indication of password strength, they should be required to enter the password twice, and returned to the registration page if the passwords in the two fields did not match. The password should be obscured by dots while being typed, and after 5 failed attempts, they will be locked out for 1 minute.

TR5: This will require that the user can log into the system using a username and password that correspond to the account that they signed up with. This should mean after they sign in, they are greeted with the main page with the posts from the users they follow and their own posts. Anything they do in the system until they sign out will be under the account they signed up with. This also requires that user can recover their password from the login page.

TR6: This will require that the user can change their password through the UI. They will be required to enter their old password, and after 5 failed attempts, they will be locked out for 1 minute. We will test this by attempting to change the password with an incorrect old password 5 times. We should observe that we are locked out for 1 minute. We will also enter the correct password and observe that the password has been changed for the current user.

TR7: This will require that the user can sign out of the system. This means after they sign out, they cannot interact with the system under their account until they sign back in.

TR8: This will require that the user can see a list of all users registered in the system. This means when they search for a user with an empty string, they are shown a page with a list of all users in the system.

TR9: This will require that the user can search for other users. This means they can enter either the first name, last name, full name, or username of the user they are looking for and after pressing search, are shown a page with a list of users that match their query.

TR10/11: This will require that it is impossible for the user to follow the same user twice or unfollow a stranger. This means when clicking on the follow/unfollow button, the appropriate action is taken depending on the current state of the system.

TR12: This will require that when typing a post, the user is not allowed to type more than 200 characters and is shown how many characters they can still type. This means the character count is correct and the user is completely blocked from typing over the 200 character limit.

TR13: This will require that when viewing the current user's main page, they are shown a list of posts that they have posted, and the users they are following have posted. Default view is chronological, they can also sort by username and hashtags. When looking at the posts, the user should be able to see the username and timestamp of each post. The system should refresh the page or the user every 30 seconds.

TR15: This will require that the user can include hashtags in their posts. This means that when a user makes a post and includes text following the character '#', each unique string will constitute a new hashtag in the system that is associated with that post.

TR16: This will require that the user has a way to see all the posts they have hashtags that they are following. We will test this with a setup that the current user has followed a few hashtags that have associated posts. We will navigate to the 'your hashtags' section in the UI and observe that every post that has hashtags that the current user is following appear.

TR17: This will require that the user can search for posts by the hashtags associated with them. This means when indicating search by hashtag, a list of posts that are associated with the given hashtag will be listed. The user should also be able to search for posts by the indicated username.

TR18: This will require that the user can like a post. This means, after liking a post, the system will contain a number indicating how many times the post has been liked.

3.0 Test Cases

Test 1

PURPOSE:

Test registering of a user.

SETUP:

Navigate to localhost/views/login.php

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
Click 'Create an account'	Create account page
The Email textbox: w	
The Username textbox: y	
First Name Textbox: a	
Last Name Textbox: b	

Password Textbox: password	
Confirm Password Textbox: password	
Pressing ‘Create account’	Login page

REQUIREMENTS COVERED:

1

Test 2

PURPOSE:

Test a user logging in to the system

SETUP:

Localhost/views/login.php, the user to test must have registered.

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
Username textbox: y	
Password textbox: password	
Press the Login Button	home page should be displayed

REQUIREMENTS COVERED:

5

Test 3

PURPOSE:

Test the password policy

SETUP:

User should be on create account page and filled out every field besides passwords.

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
Password Textbox: abc12345	Password text is changed to dots. Displays the strength of the password
Confirm Password Textbox: abc12478	
Click ‘Create Account’	Message displays “passwords don’t match”. User is redirected to sign up page

REQUIREMENTS COVERED:

4.2-4.3

Test 4

PURPOSE:

Test the lockout functionality after 5 attempts for 1 minute.

SETUP:

Localhost/views/login.php, Complete Test 1

INPUTS and EXPECTED OUTPUTS:

inputs	expected output

Username Textbox: y	
Password Textbox: q	
Click Login	
Repeat previous four steps 4 more times.	Redirected to lockout.php
Wait one minute and repeat Test 2	redirected to home page

REQUIREMENTS COVERED:

4.4

Test 5

PURPOSE:

Test the logout button

SETUP:

Test 2

INPUTS and EXPECTED OUTPUTS:

inputs	expected output

Click logout button	User is directed to login page and is logged out of the system.

REQUIREMENTS COVERED:

7

Test 6

PURPOSE:

Test the Users Page which shows a complete list of registered users in the system.

SETUP:

Test 2

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
--	
Click Search	Search Page displayed
Click Search	all users displayed

REQUIREMENTS COVERED:

8

Test 7

PURPOSE:

Test the posting functionality

SETUP:

Test 2

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
--	
Click Make a Post	Post page
Into Post TextArea enter “This post contains a #hashtag”	
Click Post	Home page, Post shows up

REQUIREMENTS COVERED:

12, 13, 15,

Test 8

PURPOSE:

Test the Hashtag page

SETUP:

Test 7, User on their home page

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
--	
Click Search	display search page

Enter ‘#’ into Search Textbox	
Select Hashtag from dropdown	
Click Search	list all hashtags in system

REQUIREMENTS COVERED:

17.1

Test 9

PURPOSE:

Test the search functionality

SETUP:

Test 2, User on their home page

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
Click Search	Search page displayed
Enter y into textbox	
Click Search	return a list of users with similar names

REQUIREMENTS COVERED:

9,R17,

Test 10

PURPOSE:

Test the follow functionality

SETUP:

Repeat Test 1 and Test 2 with a different username, etc, Test 9 with user y

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
Click the Follow/Unfollow Button	the zero is now a one when the search is repeated
F5	other User’s posts show up

REQUIREMENTS COVERED:

10, 11

Test 11

PURPOSE:

Test sorting of the post feed

SETUP:

Test 7, Repeat Test 7 with different input (including hashtags) in the textbox

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
View the home page	Posts are sorted by descending date by default.
Selects sort by Username from dropdown	
Click Sort	Posts are displayed by Username alphabetically
Selects sort by Hashtag from dropdown	
Click Sort	Posts are displayed by Hashtag alphabetically based on the first alphabetic hashtag appearing in the post. Posts with no hashtag appear last. Numbers are considered before letters.
Selects sort by Likes from dropdown	
Click Sort	Posts are displayed by descending like count.

REQUIREMENTS COVERED:

13

Test 12

PURPOSE:

Test the change password button

SETUP:

User should be on their homepage.

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
Press the change password button	User is directed to Change Password Page
User enters an incorrect old password 5 times	User is locked out for 60 seconds
Press the change password button	User is directed to Change Password Page
Enter the correct old password and new password	
Click ‘Change Password’	Password for the current user is changed.

REQUIREMENTS COVERED:

6

Test 13

PURPOSE:

Test the proper behavior of not allowing a user to sign up with a username that is already taken.

SETUP:

User should have completed Test 1. Navigate to views/login.php

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
Click ‘Create an account’	Create account page
The Email textbox: w	
The Username textbox: y	
First Name Textbox: a	
Last Name Textbox: b	
Password Textbox: password	
Confirm Password Textbox: password	
Pressing ‘Create account’	Message displays “Sorry, that username is taken. Try adding (more) numbers to the end”. No registration takes place.

REQUIREMENTS COVERED:

3

Test 14

PURPOSE:

Test the showing of posts with hashtags the user is following.

SETUP:

User should have signed up and followed at least one hashtag. Navigate to the homepage.

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
Click the ‘Your Hashtags’ button	All posts with hashtags that the current user is following appear.

REQUIREMENTS COVERED:

16

Test 15

PURPOSE:

Test the proper functioning of liking a post.

SETUP:

User should have signed up made one post. Navigate to home page.

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
Click the ‘Like’ button below the first post	Page reloads with like count+1.

REQUIREMENTS COVERED:

18

Test 16

PURPOSE:

Test password recovery

SETUP:

User should have signed up. User should be on login page.

INPUTS and EXPECTED OUTPUTS:

inputs	expected output
Click “forgot password” button	Opens recovery page
User enters email address into box	
User clicks “forgot password”	User is sent to login page, password is sent to email

REQUIREMENTS COVERED:

5