UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Testing Report

Broadsword Access

Surname, First Name	**Student Number**
Pennels Keaton	12345678
Mpofu Munyaradzi	12345678
Cilliers Joshua	12345678
Walsh Brent	15300201
Groutsch Stephanie	12345678
Schouwstra Rikard	12345678

Contents

1 Introduction

2 Test Model

3 List of Service Contracts Tested

3.1 Users

3.1.1 User Registration

Test Case 1
Action: Fill out all of the user registration details with correct data
Expected Result: The registration is a success
Actual Result: The registration was a success. A message is displayed to the user to indicate it was a success and the user is navigated back to the sign in page

Test Case 2
Action: All registration input fields are left blank
Expected Result: The registration is unsuccessful
Actual Result: The registration was not successful; the input boxes had a red outline to indicate that the data needed to be filled in
Possible improvements: Feedback in the form of text could have been given to indicate that the data needs to be filled in before a successful registration can be achieved

Test Case 3
Action: An incorrect email address is entered
Expected Result: The registration is unsuccessful
Actual Result: The registration was a success
Possible improvements: The application should check if the users email address actually exists before allowing users to register with it

Test Case 4
Action: Register using two different passwords
Expected Result: The registration is unsuccessful
Actual Result: The registration was not successful, the password input boxes have a red border to indicate there was a problem
Possible improvements: More feedback could be given. For example, a message saying that the passwords are not the same could be displayed. This would help the user as there could be other issues with the password such as it not being long enough or the password is not successful since it may need capital letters or characters.

Test Case 5
Action: Register using two different email addresses
Expected Result: The registration is unsuccessful

Actual Result: The registration was not a success, the email input boxes have a red border around them.
Possible improvements: More feedback could be given as to what the problem with the email addresses was.

Test Case 6
Action: Register more than one user with the same username
Expected Result: The registration is unsuccessful
Actual Result: The registration was a success
Possible improvements: More should be done in order to ensure that only users with unique usernames are allowed to register.

Test Case 7
Action: Register more than one user with the same email address
Expected Result: The registration is unsuccessful
Actual Result: The registration was a success
Possible improvements: The system should ensure that users are not registered to the application if their given email address is already been used by another user. This is important since email addresses are used to sign into the application.

Total Mark: 4.5/10
Additional Comments: Since almost half of the registration tests failed, the awarded mark out of ten was reduced to five and a half. This mark was then reduced by one since other improvements such as more feedback could be added in order to improve the registration process.

3.1.2 User Login

Test Case 1
Action: Fill out all of the user login details with correct data
Expected Result: The user will be logged in successfully
Actual Result: The login was a success; a message is displayed to the user that they have logged in successfully
Possible Improvements: Instead of the application taking the user to the navigation page once they have logged in, they could instead be taken to a homepage.

Test Case 2
Action: All login input fields are left blank
Expected Result: The login is successful
Actual Result: The login was not a success, the input boxes glow red to indicate that they should be filled in in order to login
Possible Improvements: The user could also be notified by means of an alert

which indicates why the login was unsuccessful

Test Case 3
Action: Fill out the user login details with a correct username and an incorrect password
Expected Result: The login is not successful
Actual Result: The login was not a success, a message is displayed which states that either the email address or password were incorrect

Total Mark: 9/10
Additional Comments: Since the actual results of all there test cases were the same as the expected results, the sign in should be awarded full marks. However, since the sign in can still be improved by giving the user more feedback when data is left blank, only 9/10 was awarded as there is still room for improvement.

3.1.3 Edit Profile

Test Case 1
Action: Update the user details and fill out all required input fields
Expected Result: The profile update is successful
Actual Result: The profile update is successful and the user is notified of the successful profile update.
Possible Improvements: Once the profile has been updated, the application remains on the update profile screen. As a possible improvement, the application could navigate back to the menu instead.

Test Case 2
Action: Update the user details and omit the details in one of the required input fields
Expected Result: The profile update process is not completed.
Actual Result: Upon clicking the "Update profile" button, a red border appears around the empty input field, and the profile update fails.
Possible Improvements: A possible improvement would be to notify the current user by means of an alert indicating to them why the profile update failed.

Test Case 3
Action: Update the user details with an invalid email address
Expected Result: The profile update process is not completed.
Actual Result: The update is a success even though the password is not valid
Possible Improvements: The email address of the user should be validated

Total Mark: 5/10
Additional Comments: The mark awarded for this module is 5/10 since the process of updating oneâĂŹs profile was not a completely uniform process. This

is mainly because the user has to navigate to a different page if they wish to update their password. Marks were also lost since one of the test cases failed. A possible improvement could be to add an input field to the âĂIJEdit ProfileâĂİ page for the user to change their password.

3.1.4 Delete Profile

N/A

3.2 GIS (GIS Management)

3.2.1 View GIS Data

Test Case 1
Action: The admin user presses the "Manage GIS" button
Expected Result: The application displays an interface for the admin user to manage locations.
Actual Result: The application displays a list of location names. It is assumed that this is a representation of GIS objects.
Possible Improvements: The list of GIS locations displayed could also include the GPS coordinates of the location to the admin user.

Total Mark: 2/10
Additional Comments: The application is intended to display an interface which an admin user can use to manage GIS locations, but instead it only provides a list of locations and no options to make any changes to these objects.

3.2.2 Add GIS Data

N/A

3.2.3 Edit GIS Data

N/A

3.2.4 Remove GIS Data

N/A

3.3 Points of Interest (Location Access)

3.3.1 Search Location

Test Case 1
Action: The search icon is pressed with the intent of searching for a location on the map

Expected Result: The application displays an input field in which the user can enter a location name and search for it
Actual Result: The application displays a list of location names.
Possible Improvements: The application could offer a search bar which the user can use to search for a specific location.

Total Mark: 2/10
Additional Comments: The absence of a search bar when searching for a location could lead to a very inconvenient experience for the user, especially if the user has a large list of saved locations.

3.3.2 Save Location

N/A

3.3.3 Get Current Location

Test Case 1
Action: The "Get Current Location" icon is pressed by the user.
Expected Result: The application will mark the user's current location on the map
Actual Result: The application marks a location on the map which is not the current user's location
Possible Improvements: A possible improvement would be to mark the user's correct location on the map

Total Mark: 2/10
Additional Comments: The application marks a location on the map, although this locations is not the correct location. This would lead to incorrect locations being saved, and could also affect the process of navigating from the users current location to another desired location.

3.4 Points of Interest (Location Management)

3.4.1 View Locations

N/A

3.4.2 Add location

N/A

3.4.3 Modify Location

N/A

3.4.4 Remove Location

N/A

3.5 Navigation

3.5.1 Navigate to Location

N/A

3.5.2 Save Preferences

N/A

4 List of Functional Requirements Tested

- Create route to valid location
- Save routes
- Heat maps
- Current user location
- Save locations
- Search for locations
- Report protest action or emergency
- Create public event
- View all locations
- Request addition, removal, or modification of locations
- Register as student, staff, admin or guest
- Login
- Manage user accounts
- Add profile information

5 List of Non-Functional Requirements Tested

5.1 Performance requirements:

5.1.1 Performance:

- Offline activities should have a response time of +/- 2 seconds (instantaneous) when responding to an activity, while online activities such as calculating routes should have a response time of +/- 2-4 seconds so that the users have an uninterrupted experience.
- It should also allow the integration of a variety of services.

5.1.2 Reliability:

- The application should be reliable, in that it will provide the fastest route every time without fail and complete all other computations successfully.
- All activities should be completed with a 10
- The application should provide accurate locations in a constantly changing environment.

5.1.3 Security:

- Data transmission should be securely transmitted without unauthorized access, or loss of information.

5.2 Design Constraints:

- The system should be accessible on smart devices, such as Android and iOS devices.
- The system should not use GPS, but only the WiFi network.
- The proposed system should be able to be integrated into the Computer Science Department's Web site.
- The system should be a modular system, to reduce the dependencies in the system.
- Software Fault Tolerance: If a malfunction cannot be avoided, then the software design should be constrained so that the system can recover without causing damage to the system.
- The system should have an aesthetically pleasing and easy to use interface.
- The system must be able to run on smart devices which has limited processing power, battery life and storage space. The system must thus use resources efficiently.
- The system needs to use open source technologies.

5.3 Software System Attributes:

- Users should have the option to withdraw all information gathered by the system.
- The system should be available online as well as offline.
- The system should stay updated, to ensure reliable information. For instance the maps of campuses should be updated regularly.
- The system should easily be updated, without complications.
- The system should be managed efficiently, checking for problems regularly.
- The system should be secure to prevent unauthorized modification or access of information.
- The system should be user-friendly, the application should meet the requirements of the user by providing good access for disabled users, and resulting in a good overall user experience.

6 Evaluation of Test Cases for Non-Functional Requirements