BookInTime

Software Test Plan

CSCI-P465/565 (Software Engineering I)

Project Team
Brendan Mcshane
Shanthan Reddy M
Aditya Shahapure
Prathmesh Deshmukh

1. Overview

As of this sprint we plan on implementing and testing a user login interface. We'll need to be able to store usernames, passwords, and other information about the user. We'll need to check user input against these records to see what access to give the user, given they input correct information when they attempt to log in.

1.1 Test Objectives

- ensure that usernames, passwords, emails, and other user information makes sense and fits constraints when creating a new user

- ensure that the data is stored and maintained correctly

- ensure that the login process is functioning, and that only users entering correct information that is stored in our database are allowed access to the site

1.2 Test Environment

Describe the environment that will be used to test the software product. Include a description of the hardware and operating system of all machines involved in the test. Describe the networking between those machines (if your product requires network support). Describe any support software packages that need to be installed/available/running on the machine for the software to operate. Comment on whether the test environment is the same as the operational environment, and descibe any differences.

We plan on utilizing the Ubuntu OS to test our software. The hardware specifications of the computers being used to test are not important as of now. A network is also annecessary as of this sprint. React, Docker, Django, MySQL all need to be installed for this software to run.

1.3 Test Personnel

Describe the personnel involved in the testing effort, and the roles that they will fill. Describe the level of user involvement in the testing process (if any).

The whole team will be involved in the testing process, each member coming up with individual manual test cases to ensure the system works.

1.4 Acceptance Criteria

Under what conditions will the software be considered acceptable to deliver. For example, how many errors will you tolerate before the testing is abandoned to correct the errors and try again? Or, how many of the requirements need to be verified before the software is considered deliverable. Identify in this section an acceptance criteria that is feasible for your development team and acceptable for your customer/user.

If all aforementioned testing objectives in section 1.1 are met and guaranteed, the sprint will be acceptable to deliver. Any errors or mishaps in the login process will need to be addressed.

1.5 Noted Omissions

Identify any software specification statements that are not intended to be verified by this test, and justify the exclusion of these specifications.

The login page will be the sole feature of our product at time of testing, so nothing we have created will be omitted from testing.

2. Test Cases

The test cases are the partitioning of the verification of the software into manageable sections. Often these sections correspond to the set of active use case scenarios, but can be organized as the test developer sees fit. Test cases should be in place to cover all of the software verification methods. Even non-execution based testing methods (i.e., inspection/analysis) may be detailed here. The intent is for the test case procedures to provide a repeatable verification of the software specification.

For each test case describe the following:

Number: Provide a unique number for the Test Case
Name: Provide a unique, descriptive name for the Test Case.
Description: Provide a summary description (1-3 sentences) of the Test Case, and what is intended to be verified by the test case.
Initial Conditions: Describe any unique conditions in the test environment that must be in place to run the test.
Also, note whether another Test Case is intended to be run prior to this Test Case.
Input Data: Describe any specific data inputs that are to be used in the Test Case
(e.g., a mock data file, a test user ID, erroroneous inputs to stimulate error conditions,

etc).
Specifications: List the Specificaiton statements (by number) that are intended to be verified by this Test Case.
Procedure: Provide a link to the Test Case Procedure for this Test Case.

You may represent this information as a table, or simply list it.
NOTE: If you are using the rigorous process, the Specifications described above are already captured in the RVM and may be omitted from this document

Number: 1

Name: create user test

Description: tests to make sure we can store user data and create new users

Initial Conditions: N/A

Input Data: we'll create information for a new user, the specifics of which are not important as of now

Specifications: need an entry for each column in the user data table

Procedure: enter information into provided text boxes, check to make sure the user was created in our data

Number: 2

Name: login user test

Description: tests to make sure we can login to an existing user

Initial Conditions: N/A

Input Data: the email and password of the user created in test 1

Specifications: N/A

Procedure: enter information into provided text boxes, check for some print or return statement to confirm the account was found in our data

Number: 3

Name: incorrect user test

Description: tests to make sure users entering incorrect/non-existing information cannot login

Initial Conditions: N/A

Input Data: email and password that don't exist in our data

Specifications: N/A

Procedure: enter information into provided text boxes, check for some print or return statement to confirm the account wasn’t found in our data

Number: 4

Name: login with google

Description: tests to make sure users is successfully logging in with google account

Initial Conditions: N/A

Input Data: email and password that exist in our data

Specifications: N/A

Procedure: enter information into provided text boxes

Number: 5

Name: login with facebook

Description: tests to make sure users is successfully logging in with facebook account

Initial Conditions: N/A

Input Data: email and password that exist in our data

Specifications: N/A

Procedure: enter information into provided text boxes

Number: 6

Name: login with phonenumber

Description: tests to make sure users is successfully logging in with phonenumber

Initial Conditions: N/A

Input Data: phonenumber and password that exist in our data

Specifications: N/A

Procedure: enter information into provided text boxes

Number: 7

Name: login with google failed

Description: tests to make sure users log in failed with google account

Initial Conditions: N/A

Input Data: email and password that don’t exist in our data

Specifications: N/A

Procedure: enter information into provided text boxes, suggest that the user does not exist please signup

Number: 8

Name: login with facebook failed

Description: tests to make sure users log in failed with facebook account

Initial Conditions: N/A

Input Data: email and password that don’t exist in our data

Specifications: N/A

Procedure: enter information into provided text boxes, suggest that the user does not exist please signup

Number: 9

Name: login with phone number failed

Description: tests to make sure users log in failed with phone number

Initial Conditions: N/A

Input Data: phone number and password that don’t exist in our data

Specifications: N/A

Procedure: enter information into provided text boxes, suggest that the user does not exist please signup

Number: 10

Name: signup failed due to existing user mailid

Description: tests to make sure users signup failed due to existing account

Initial Conditions: N/A

Input Data: email id already exists In record and user is trying to sign up

Specifications: N/A

Procedure: enter information into provided text boxes, suggest that the user already exists please reset password

Revision History

Revision	Date	Change Description