Sluggit

A Computer Science blog and resource

Team Members: Jasmine Dupree,, Maximilian Brain, Tarum Fraz, Diego Garcia (SM), Hope Ashcraft, Chris Gradwohl(PO)

System Test Scenarios Report

Revision Number: v1.1

Revision Date: 06/05/17

Test 1: Test the posts interface

This test relates to the following user stories:

    A. User Story 2 from Sprint 1: As a Sluggit user, I want to view the Sluggit public feed so that I can discover new content from the Sluggit community.

    B. User Story 2 from Sprint 2: As a Sluggit user, I want to be able to post a blog to share my ideas to the Sluggit community.

    C. User Story 1 from Sprint 3: As a user I would like a feed to display the latests posts and the most popular.

    D. User Story 2 from Sprint 3: As a Sluggit user I want to be able to be able to 'like' blogs so that I can shape the landscape of the sluggit community.

How this was tested:

1. Navigate to Sluggit page
   a. login/signup

      i. User name <tester>

      ii. Password <123>
   b. Hit create post tab

      i. Title <sample title>

      ii. Body <here is a sample body….>
2. Directs you to the public feed page
3. Click "upvote"
   a. See the votes in upper right hand corner increase by 1
4. Click "downvote"
   a. See the total votes in the upper right hand corner decrease by 1
5. Click "Popular on Sluggit"
   a. View the posts with highest number of votes

Expected Result: The Popular on Sluggit page should have same features as Feed page, however the posts should be ordered by the total votes in the right hand corner.

Actual Result: The system passed this test. Every member of the group tested this and enjoyed using the interface.

Test 2: Test the ability for a user to login.

This test relates to the following user stories:
A. User story 3 from sprint 1: As a Sluggit user, I want to signup and login, so that I can post content to Sluggit.
B. User story 2 from Sprint 2: As a Sluggit user, I want to be able to post a blog to share my ideas to the Sluggit community.

How this was tested:
1. Start Sluggit app; Select "Sign up"
   a. Enter name, username, email, and password
   b. Click Submit
2. This takes you to the login page
   a. Enter credentials and click login
3. This takes you to the public feed
   a. Click "Create a Post"
   b. Enter a title and description
   c. Click "Submit"

Expected Result: The public feed page should include your post with you as the author.

Actual Result: The system passed this test. We ensured the feature works by logging into another account and going through the steps again then monitoring the output.

Test 3: Test functionality of User Profile

This test relates to the following user stories:
A. User story 2 from sprint 3: As a Sluggit user, I want to customize my Sluggit user profile so that my online identity can reflect my ideals and beliefs.

How this was tested:
1. Start Sluggit app; Select "sign in"
   a. Log into or create an account
2. Select User Profile
   a. User should be able to see basic profile information, as well as blank fields not filled out yet if profile has never been edited
3. Select 'Edit Profile'
   a. User should be able to fill out all fields, in order to update information
4. Select Save

a. User should be directed to profile, and can now view basic profile information

Expected Result: The user can view and edit basic profile information successfully

Actual Result: The user can view and edit basic profile information successfully

Test 4: Test functionality of DELETE POST

This test relates to the following user stories:
   A. User story 2 from sprint 1: As a Sluggit user, I want to view the Sluggit public feed so that I can discover new content from the Sluggit community.
   B. User story 3 from sprint 1: As a Sluggit user, I want to post text to the Sluggit public feed, so that I can share my ideas with the Sluggit community.
   C. User story 2 from sprint 2: As a Sluggit user, I want to view a clean, simple public feed, so that I can enjoy reading and discovering new content.

How this was tested:
   1. Start Sluggit app; Select "sign in"
   2. Select Feed
        a. Re-directs user to feed that contains all posts
   3. Select "Delete Post"
        a. This should only show up for posts made by logged in user
   4. Post should be deleted successfully

Expected Result: The user is able to delete a post successfully.

Actual Result: The user can delete a post successfully. However, if a user creates a post, and then decides to delete it, the "delete" post button will show up for all posts, even if they are not made by that user. If the user chooses this path, they are able to delete posts that belong to other users.

Test 5: Test functionality of EDIT POST

This test relates to the following user stories:
   A. User story 2 from sprint 1: As a Sluggit user, I want to view the Sluggit public feed so that I can discover new content from the Sluggit community.
   B. User story 3 from sprint 1: As a Sluggit user, I want to post text to the Sluggit public feed, so that I can share my ideas with the Sluggit community.

C. User story 2 from sprint 2: As a Sluggit user, I want to view a clean, simple public feed, so that I can enjoy reading and discovering new content.
D. User story 3 from sprint 3: As a Sluggit user, I want to write blog posts using an intuitive text editor so that posting to Sluggit is easy and fun.

How this was tested:
1. Start Sluggit app; Select "sign in"
2. Select Feed
    a. Re-directs user to feed that contains all posts
3. The user is able to view and select the edit post button
    a. Should only be visible on posts made by the user logged in
4. The user is able to edit and submit the post
    a. Post should be edited and reflective on the feed.

Expected Result: The user is able to edit a post successfully

Actual Result: The user is able to edit a post successfully. We do not have the same issue here as we did with deleting the post. A user is only able to edit a post that belongs to them.

User Stories that were not implemented:

User Stories:
A. User story 1 from sprint 2: As a Sluggit user, I want to signup and login quickly and easily(Google, Github, Facebook authentication) with a clean intuitive login UI, so that I enjoy using Sluggit.

Reasons:
● This was a low priority user story, so we focused on the more essential features.
● We did not implement Google, Facebook, and Github authentication, but we were able to build a clean and intuitive user interface

Bugs and other stuff that doesn't work:

1. The delete post functionality works, but has some bugs
    a. Initially, a user is only able to delete a post that they have created
    b. If they create a post, and then delete it, the delete button shows for all posts.

c. If a user chooses this path, they are able to delete posts made by other users.