

## System and Unit Test Report

Product Name: Purrsight

Team Name: J-Team

Completion Date: Tuesday July 21th, 2020

**Team Members:** Jon Zivku, Jake Forrester, Jake Liao, Silvana Moiceanu, Jimmy Tang

**Installation instructions:** <https://github.com/jonzivku/Purrsight> in README.md

Frontend and Backend servers need to be run at same time

- A. **User story 1 from sprint 2:** As a user, I want to sign up using credentials of my choosing so that I can create an account on the platform and distinguish myself from other users
- B. **User story 1 from sprint 3(Partial):** As a user I want to have my own profile so that I can see a photo of me, my contact information, and other pertinent info.
- C. **User story 2 from sprint 3(Partial):** As a user I want to be able to share my cat photos so that other users can see pictures of my cats.

### Scenario:

#### Scenarios for User Story 1 from sprint 2 (Login and Registration)

1. Navigate to Purrsight in your browser; select 'Sign Up'; type
  - a. email = <[testemail4210@foo.com](mailto:testemail4210@foo.com)>
  - b. password = <bestpassword123>
  - c. password confirmation = <bestpassword123>
  - d. click 'Signup'
  - e. User will be redirected to 'Photo Feed' page
2. After Registered Navigate to Login Page; type
  - a. email = <[testemail4210@foo.com](mailto:testemail4210@foo.com)>
  - b. password = <bestpassword123>
  - c. User will be redirected to 'Photo Feed' page
3. Navigate to Purrsight in your browser; select 'Sign Up'; type
  - a. email = <[testemail4210@foo.com](mailto:testemail4210@foo.com)>
  - b. password = <bestpassw>
  - c. password confirmation = <bestpassword123>
  - d. User will be met with an error message indicating issue with signup
4. After Registered Navigate to Login Page; type
  - a. email = <[testemail4210@foo.com](mailto:testemail4210@foo.com)>
  - b. password = <bestpa123>
  - c. User will be met with error message indicating invalid login credentials

### **Scenarios for User Story 1 from sprint 3 (Profile Creation)**

1. Navigate to Purrsight in your browser; select 'Login'; type
  - a. email = <[testemail4210@foo.com](mailto:testemail4210@foo.com)>
  - b. password = <bestpassword123>
  - c. User will be redirected to 'Photo Feed' pag
2. Once Logged in navigate to profile page by clicking on Purrsite in the top bar
  - a. This will navigate to home page
  - b. Profile Card is viewable
  - c. Post Card is viewable
  - d. Posting in middle of application accepts image file to upload

### **Scenarios for User Story 2 from sprint 3 (Posting images to profile)**

1. Navigate to Purrsight in your browser; select 'Login'; type
  - a. email = <[testemail4210@foo.com](mailto:testemail4210@foo.com)>
  - b. password = <bestpassword123>
  - c. User will be redirected to 'Photo Feed' pag
2. Once logged in enter information into posting text box
  - a. Click upload photo and select a photo to upload
  - b. Click submit and photo will be sent to backend
  - c. Information from that post will be gone
  - d. Backend server will respond with bad request because api doesn't exist

The majority of testing was a bang on it with a hammer approach. As our whole team was learning React and other development tools, a lot of the testing we did was learn as you go. Some logging of information to the browser console was used to test values. For the backend API, Postman was used to test certain endpoints for retrieval of information.