STaRS Judging APP

Final Report

Team Polaris

Nancy Sardar, Jacob Weekley, Juan Linares, Andrew Plourde

Application Summary

The STaRS Judging App is a hybrid mobile and web application for faculty judges to utilize during the STaRS poster competition held at Georgia Gwinnett College. It is used to facilitate and manage the ratings judges provide and keeps track of them whether the application is online or offline. Also provides results for student presenters to check their scores. The main goal of this application is to provide faculty judges with an easier, user-friendly interface to submit their ratings and critiques for student posters during the STaRS event.

Features Implemented

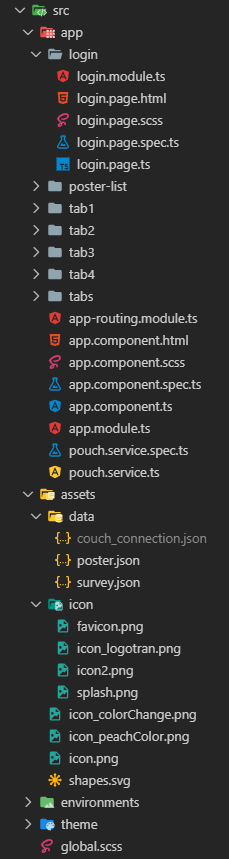
* Updated to a modern UI/UX design
* Updated to Angular version 8.1.3 and Ionic version 5.4.4
* User can log into the application if their credentials already in the database
* User has the ability to:
  + view list of posters for STaRS event and posters already surveyed by judge
  + search for the poster based on the poster’s title
  + filter the posters based on subject
  + rate the posters based on many different criteria
  + add personalized comments to their poster rating
  + use help icons to help them better understand question specifics
  + submit their vote to the database
  + refresh the homepage in order to see their updated surveys
  + submit application feedback
  + log-out of the application
  + check to make sure the user has filled out all of the survey questions

Features Remaining To Do

* Implementing:
  + ability for new users to register/a way for admins to register new users
  + view judges that have already voted on a specific poster
  + count the number of surveys on a specific poster
  + view the ratings that the user has previously submitted
  + change that ratings that the user has previously submitted
  + Offline usability that updates once back online
* Implementing
  + a student-side view to view results of their posters
  + an admin-side view to view results and send results to students
    - Angular application available but needs to be refactored

Code Architecture and Organization

* New Angular folder structure
  + Every page has its own html, controller, scss
  + Overall Folder Structure:



* All routing is configured in its own file (*app-routing-module.ts*)
* Implemented an Angular service:
  + is used to connect to CouchDB and PouchDB
  + Can inject methods into other component controllers to get the database data
* Fetched local JSON files
  + The *couch\_connection.json* file has the password to the database.
    - Must store this password in a file that is not on repo in order **to maintain security.**

Project Flyer



STaRS Judging APP

Software Development II | Fall 2019

Team Polaris



Team Polaris is a dedicated team of Software Development students who were part of the Software Development II - Fall 2019 course offered at Georgia Gwinnett College (GGC). These students worked to the best of their ability on the ‘STaRS Judging APP’ to create an updated, modern UI/UX design while updating the application to the latest Angular and Ionic versions. While working with brand new frameworks, these students also learned about Agile Software Development as well as Scrum-based methodologies.

They succeeded in making a viable application and were a success at the CREATE Symposium at GGC - Fall 2019. They were recognized as the top team presenting for Software Development II.

Nancy Sardar

* + Project Manager | Client Liaison | Programming Team

Jacob Weekley

* + Lead UI/UX | Lead Programmer

Juan Linares

* + Lead Programmer | Testing Team

Andrew Plourde

* + Lead Documentation | Lead Tester

Clients

Team Polaris worked with two clients in Fall 2019, Dr. Anca Doloc Mihu and Dr. Cengiz Gunay. As based in Agile Development, the clients worked hard with Team Polaris as part of the development team to ensure the best possible product can be achieved. Client meetings were held frequently with communication via emails. This process was to include them through each processes and iteration of the project. Team Polaris is grateful for their dedication and teamwork this semester and were honored to have worked with them.



**Dr. Anca Doloc Mihu**

* Part of the Information Technology staff and Professor at Georgia Gwinnett College



**Dr. Cengiz Gunay**

* + - Part of the Information Technology staff and Professor at Georgia Gwinnett College

Testing

**Methods:**

Team Polaris’ testing methodology for this project has been Test After Development (TAD). As opposite to the popular Test Driven Development (TDD), TAD is a method that has the team write code and then test it.

The testing software used were Karma, Protractor, Jasmine, and Istanbul.

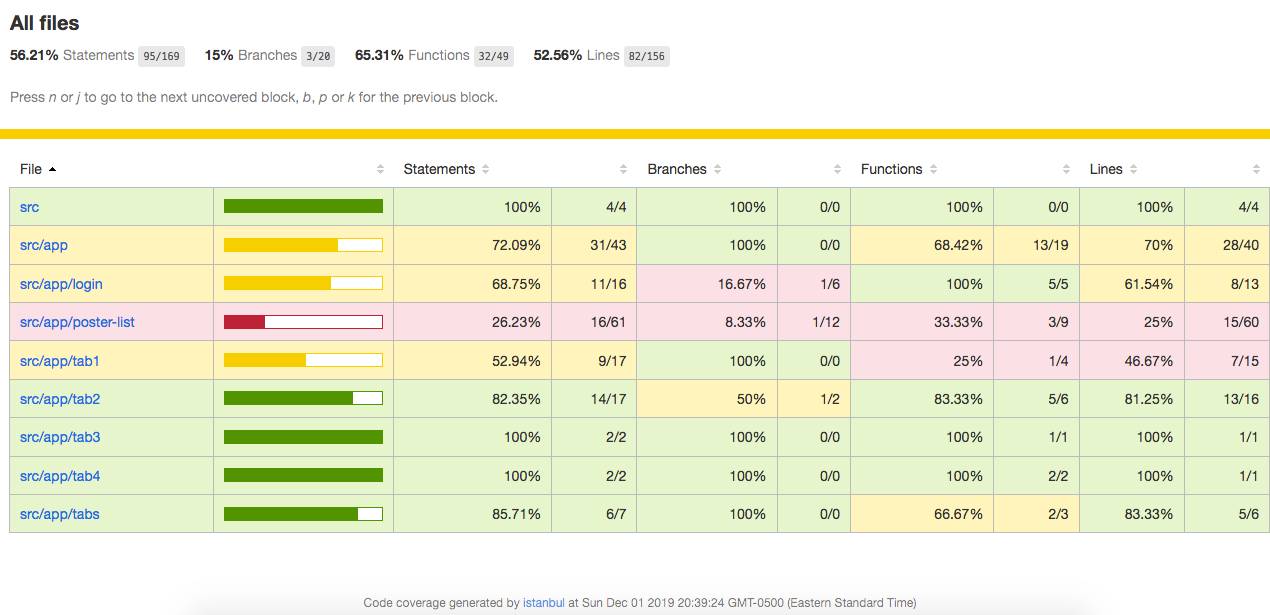
* Karma is the testing environment. It provides the setup for launching the tests in a browser for easy to understand test results.
* Protractor is the testing framework that comes with Angular. This is present in the project but karma is used as the test environment.
* Jasmine is a library that allows the generation of spies and mock objects as testing tools.
* Istanbul is a library that displays code coverage. It can be presented in many ways, key example being an HTML page as shown in the Results section.

**Coverage:**

* The testing done on this project were primarily unit tests.
* Minor integration testing was done.
* End to End testing was accomplished explicitly through the view portion of the application, i.e., hands-on use of the application.
* User Acceptance testing was performed at the CREATE Symposium at Georgia Gwinnett College - Fall 2019.

**Results:**

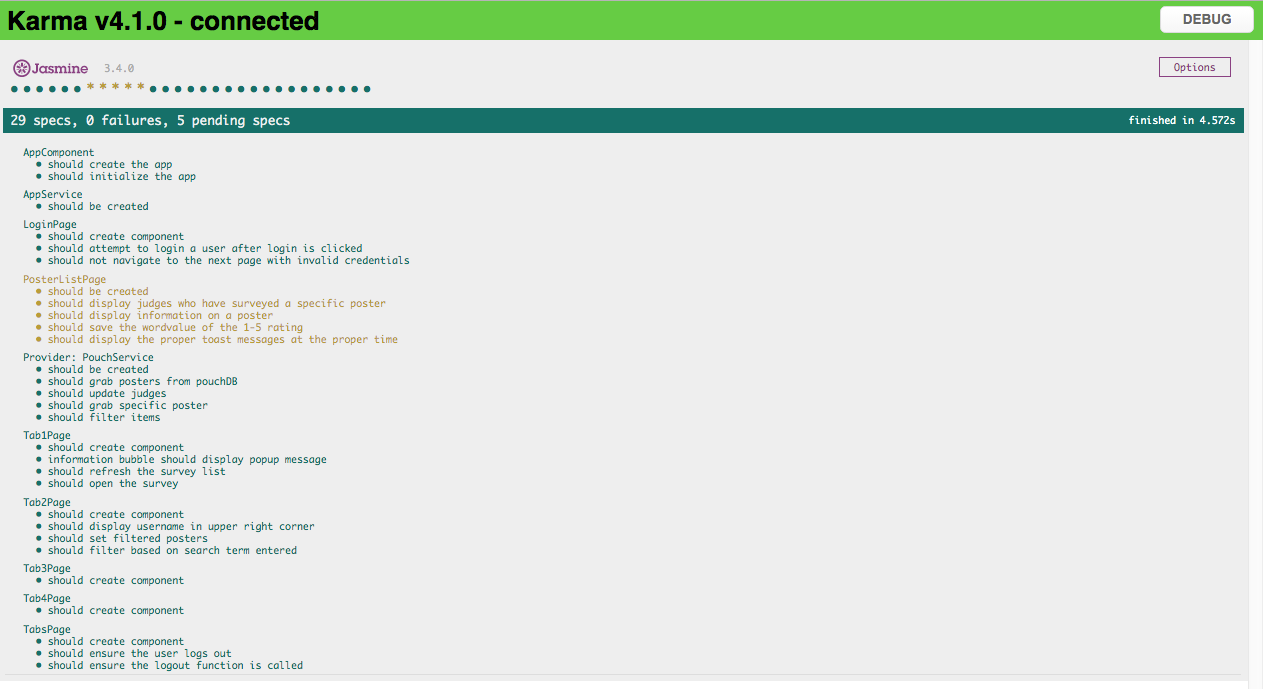
**Code Coverage Report using Istanbul**



Ideally, 90% or greater Testing Coverage is the desirable result. The *src/app/poster-list* file contains the bulk of the code for the project. This file contains all the logic for grabbing, sorting, filtering, and displaying the posters for judging. For reasons unknown, the file does not want to compile. The issue lies in its dependence on CouchDB and PouchDB. All the information from the database goes through this file, therefore making a very difficult system to mock and test. That is the reason as to why the code coverage is so low.

Unit test stubs were written with TODO in the comments.

**Jasmine/Karma Test Spec Results**



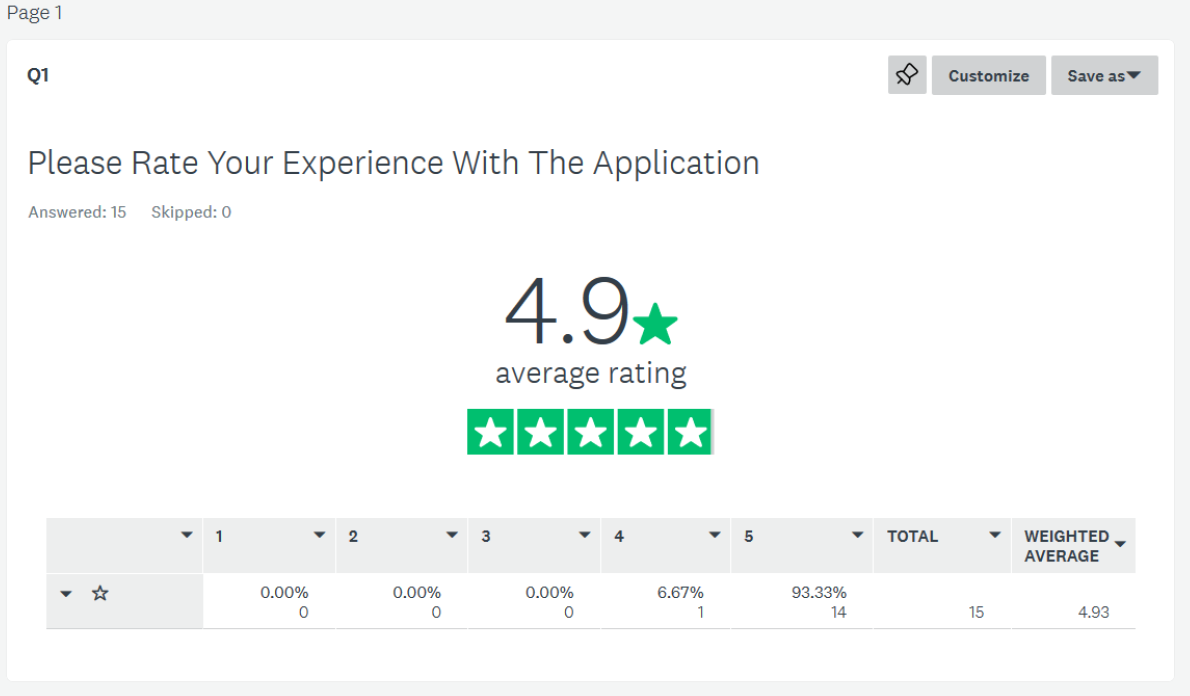
This is the results page of the *Spec.ts* files that were tested inside of the Karma testing environment. This directly correlates to the Code Coverage report above. Tests are broken up by component. Each component has its tests listed beneath it. Tests are written with the phrasing of “*They should do (action)*”.

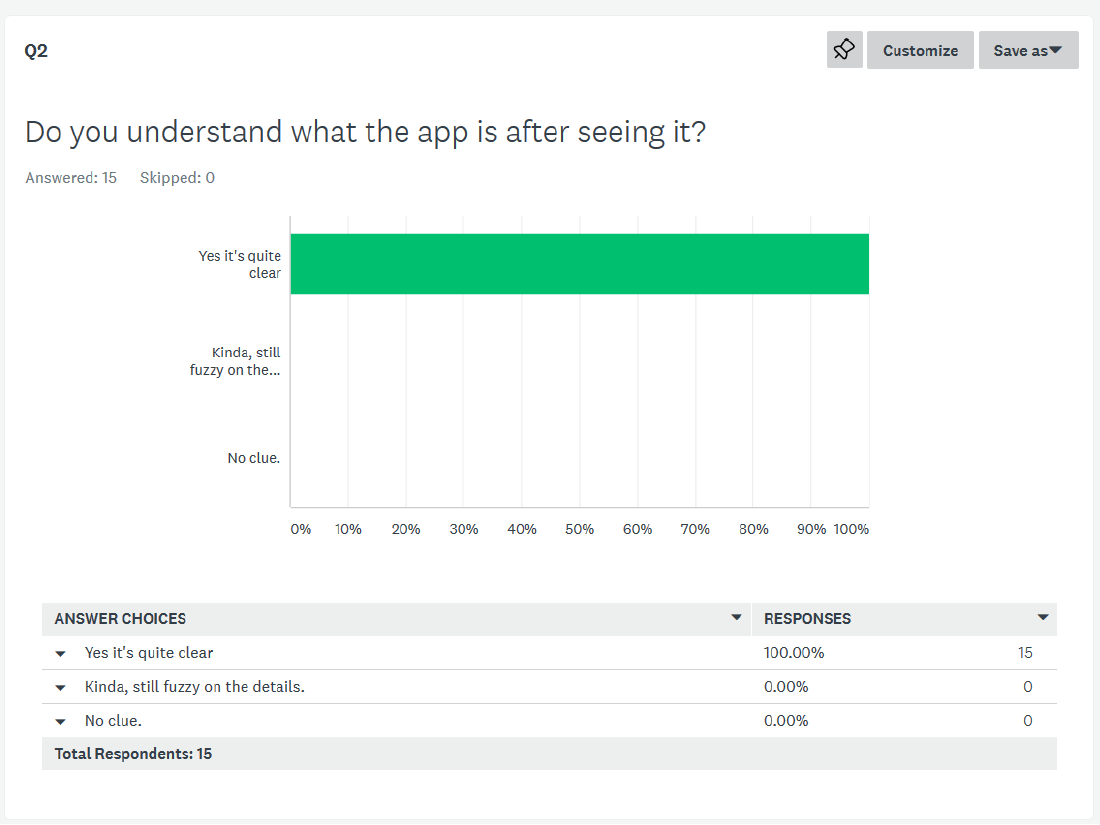
The tests consisted of ensuring the component was generated, and the methods return their expected values. Green tests are tests that have passed, red tests are tests that have failed, and yellow tests are tests that are pending.

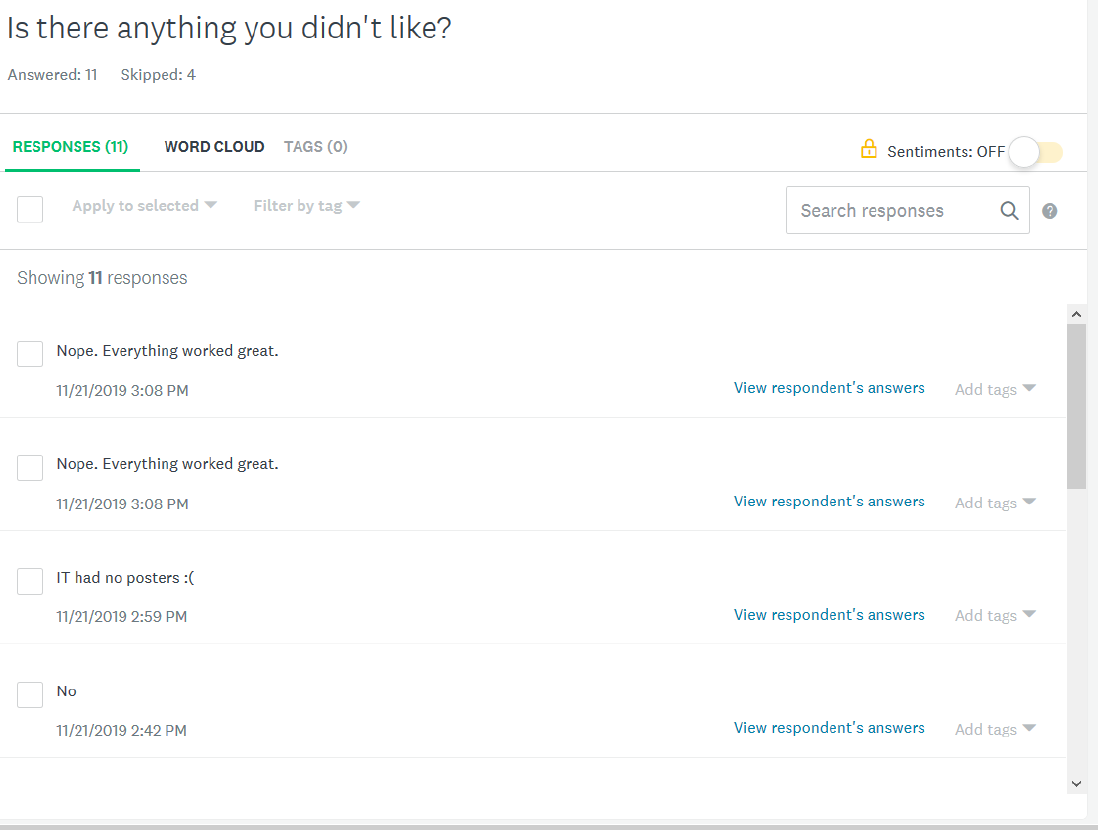
As stated previously, the *PosterListPage* component would not compile in a test environment. Therefore, those tests are skipped.

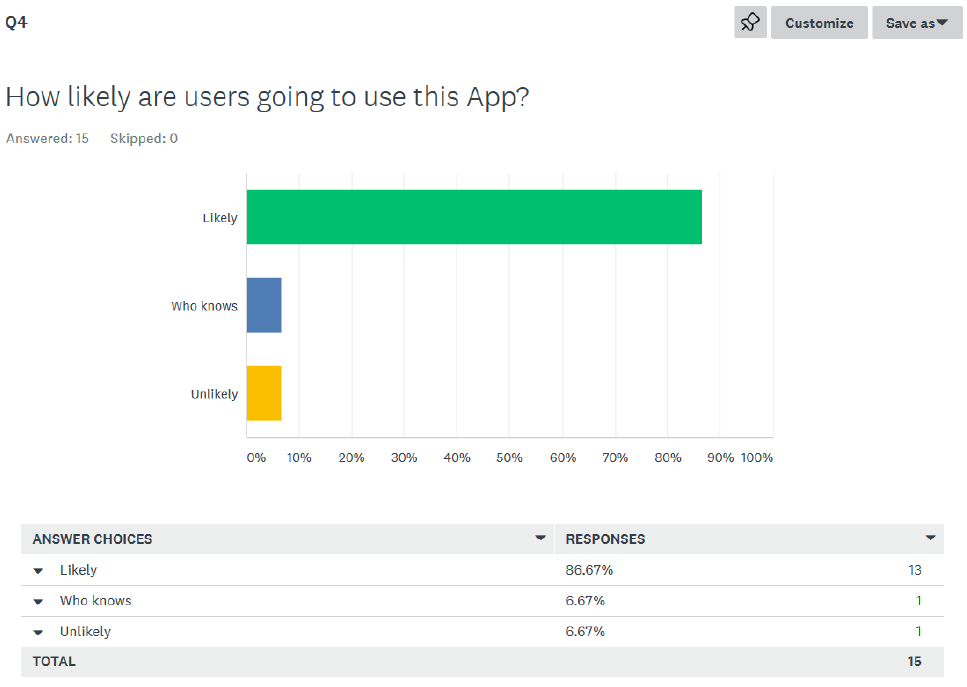
Usability Survey Results

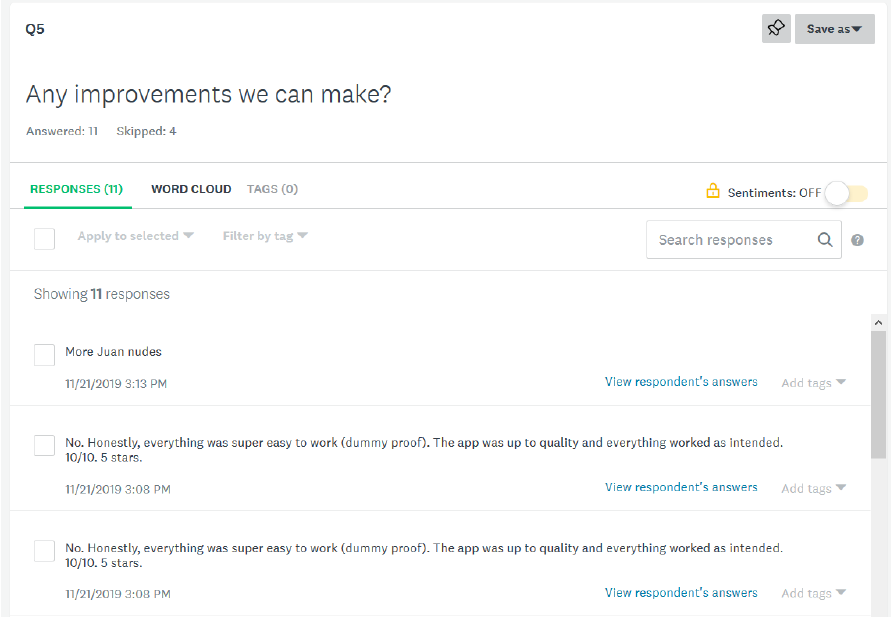
*via Survey Monkey*

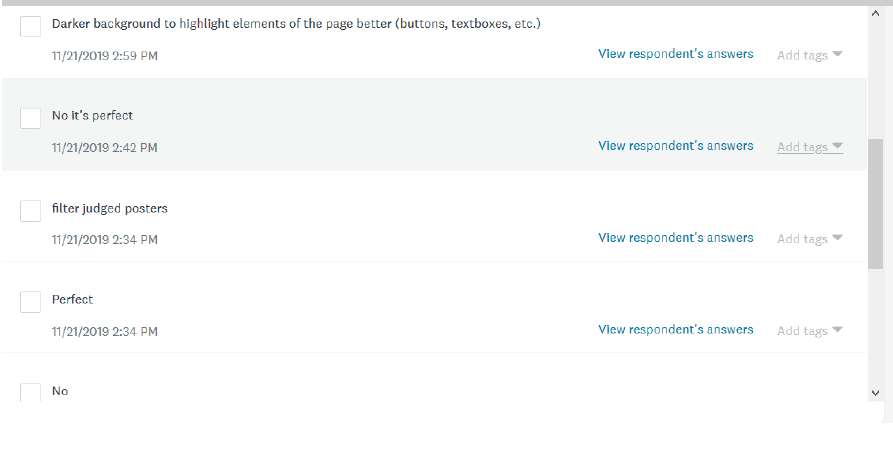


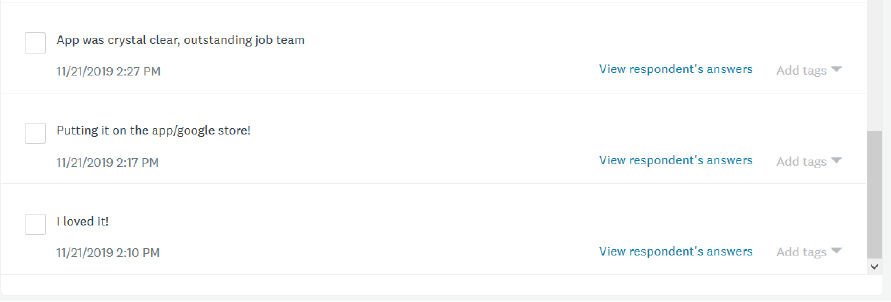






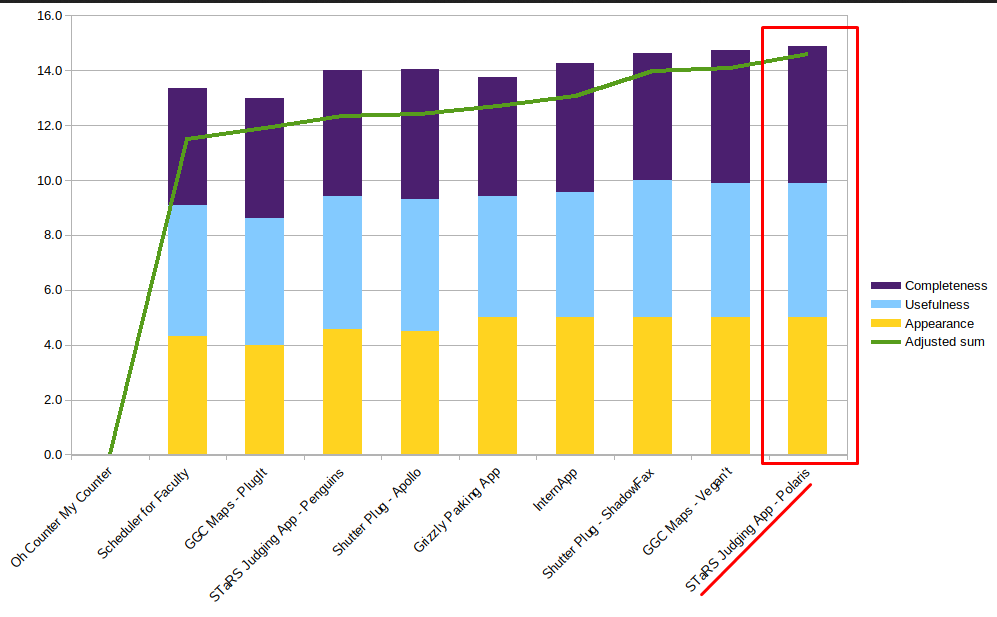






Survey Results

*of all Software Development projects at CREATE - 2019*



Installation Instructions

To Install:

1. npm is required to install packages and run the app
2. Install Ionic and Cordova
   1. npm install -g ionic
   2. npm install -g cordova
3. Clone repository
4. Navigate to the folder location in a terminal and run npm i to install dependencies

To Run:

1. After installation navigate to folder in a terminal
2. Can be served to the browser with ionic serve

### Android

1. Can be run with an Android Studio emulator with ionic cordova emulate android
2. Can be run on connected device as well
   1. If only one Android device attached use ionic cordova run android
   2. If multiple Android devices attached must specify device
      1. To get device list run ionic cordova run android --list
      2. Run to targeted device with ionic cordova run android --target=e78ab88d

### iOS

* **Can only be run/emulated on Apple computer**
* Create an Xcode project using the STaRS directory
  + Ionic will now launch the app using Xcode
* Emulation for iOS
  + ionic cordova emulate ios
* Run on Apple device
  + ionic cordova run ios

User Documentation

*via Github’s wiki pages on the project’s repository.*

<https://github.com/soft-eng-practicum/STaRS/wiki>

Screencast/Video Demo of Application

<https://youtu.be/5cVWPM-nqDI>

Software Usage License

<https://creativecommons.org/licenses/by/4.0/>

Signed Intellectual Property Terms

<https://drive.google.com/file/d/1t1LgBZt1pIM-z_tPM7ahCBO4Ces56_kP/view>