Taps Near Me

Zach George, Tim Leonard, Alex Saltstein, Connor Smith

September 13th, 2021

**INTRODUCTION**

Our project is to create a mobile application that utilizes the Untappd API in order to allow users to search nearby bars and breweries for different craft beers. The purpose is to allow the user to search for bars and beers by searching and filtering by beer type, styles, contents, etc. Currently applications on the market that provide craft beer information do not provide the ability to search for beers at breweries and bars within a given radius. Instead, it requires a manual search, which includes searching for breweries and bars within an area in order to find what is offered.

Our project will also utilize location services that can provide recommendations for craft beer locations in a given users area, as well as potentially employ data-mining techniques to pull from bar craft beer menus in order to provide a complete dataset of available craft beers. This can also be updated for breweries based on changes in seasonal brewing, new releases, and overall if beers are in-service or not. For example, if there is a user that likes a Sour style of beer, and they are willing to travel a maximum distance of 30 miles from their location, they can search for beers of that style within 30 miles from their location. This application can also be beneficial for the craft beer enthusiast that is travelling. If an enthusiast wants a beer lighter in alcohol content while they are visiting Orlando, they can find beers of a certain alcohol percentage in that area. This application is successful if it accurately outputs the breweries and their respective beers based on the requested search criteria in a sophisticated manner via a user-friendly interface. Additionally, the application is successful if it correctly outputs when there are not any beers available based on the provided search criteria.

# ROLES AND RESPONSIBILITIES

Project Lead **Zach George**

Project Manager **Zach George**

Development Lead **Tim Leonard**

Architect **Connor Smith**

Developers **Alex Saltstein, Tim Leonard, Connor Smith, Zach George**

Buildmeister **Connor Smith**

Infrastructure lead **Connor Smith**

Test Lead **Zach George**

Testers **Zach George**

Documentation **Connor Smith**

Designer **Alex Saltstein**

User advocate **Zach George**

Risk Management **Tim Leonard**

System Administrator **Connor Smith**

Requirements Resource **Zach George**

Customer Representative **Zach George**

# METHOD

* Software:
  + Language: JavaScript
  + Framework:
    - React Native (front end visual framework)
    - Recoil (front end state management)
    - Express (back end)
  + Platform: Expo (front end tool suite)
  + Design software: AdobeXD
* Infrastructure:
  + Cloud Provider: Google Cloud (back end api hosting)
  + Front end: Expo build service
* Development Process:
  + Our development process will be Agile and revolve around our weekly meeting schedule
    - Sprints will be 2 weeks each
  + **Usability Reviews** - UAT Sessions to test usability on both mobile operating systems
    - May employ people outside of the project team for testing
    - Brooke’s Usability Scale can be employed to measure the usability of the system
  + **Design Reviews** - Meetings between design, user advocate, and engineering to discuss Adobe XD specs for new features / layouts
  + **Code Reviews** - mandatory for any new pull request, at least one other engineer should do a review of the code itself, the visual usability will be tested in weekly meetings with usability reviews.
  + **Testing** - Github actions CI will run testing on any opened PR and log code coverage and require 100% test cases passing.
* Build Plan:
  + Version Control - GitHub
  + API deployments - GitHub Actions / Google Cloud App Engine
  + App building: Expo
    - Builds are then submitted to Apple App Store and Google Play Store.

Relevant documentation:  
<https://expo.dev/>

<https://recoiljs.org/>

<https://expressjs.com/>

<https://cloud.google.com/community/tutorials/run-expressjs-on-google-app-engine>

# COMMUNICATION PLAN

## Working team meetings

We have agreed to meet on Tuesday at 5:30 pm EST. The team plans to meet until the necessary material for the meeting is completed. Ad-hoc meetings can be scheduled as needed. Meetings will be done over Google Meet found here: <https://meet.google.com/ari-wedm-poq>

During these meetings we will review any upcoming design changes, review current requirements to be developed, review upcoming requirements to be developed, and do usability testing on any new features that are going to be added to the application soon.

## Status meetings

These will be done during class time. The team will present the application in its current state, struggles that the team has had during the current sprint, and changes that are expected to occur in the next iteration of the application.

## Issues meetings

We have a GroupMe in which every member is a part of. If any issue arises one can reach out to other members there. Issues can also be brought up in weekly team meetings, but if there is a need to have an additional meeting to discuss an issue, that can be conducted.

# TIMELINE AND MILESTONES

Week 1: Tool Setup

Week 2: Initial App design finished

Week 3: Most of Sprint 1 completed

Week 4: Sprint 1 completed/Technical Architecture

Week 5: Working App on IOS and Google play store (MVP), Most of Sprint 2 completed, Technical Presentation with Demo

Week 6: Sprint 2 completed

Week 7: Most of Sprint 3 completed

Week 8:Sprint 3 completed

Week 9: Most of Sprint 4 completed, Demo available

Week 10:Sprint 4 completed

Week 11: Most of Sprint 5 completed

Week 12:Sprint 5 completed

Week 13: Final Report and Presentation finished

# RISKS

* Anything that is alcohol related has some inherent risks involved. Apple and Google are slightly strict about alcohol related apps being approved for sale on the app stores, so this could cause us to not be approved. These can include anything from alcohol abuse, minors using the application, or any liabilities that can be associated with alcohol. To counteract this there are going to be waivers or warnings posted throughout the application that asks for birthday verification or something along those lines.
* Location and other data collection methods need to have thorough security so that they do not fall into the hands of bad actors. The application will request the user to provide their location and will only use their location while the application is open. When it is open, the location will be available only to the application.
* It is unknown if the source of data that will be used for this project is updated regularly. If it is not, the most accurate information will not be available for the users. The update frequency of the data source will be reviewed and outputted on the application, if available. If it is not, then the application will share a disclaimer that it is not the source of the information.
* If the data source is not available, the application cannot operate. The application will monitor the status of accessing the data source. If it is not available, the application will state that the application cannot be used.
* If the location of the user cannot be accessed, it cannot be used in the application. Instead, the user will be prompted to enter a location.

# ASSUMPTIONS

* We are assuming that there will be some way to collect data about craft beers. Either through web scraping or by using the Untapped API found here: <https://untappd.com/api/docs>
* We are assuming that users are willing to turn on their location services for the app to use and track while the application is open.
* We are assuming that if the data source is not web scrapped, that it is updated regularly.
* We are assuming that most, if not all, of the information that could be available for each beer is available for all beers.