

FLAVR

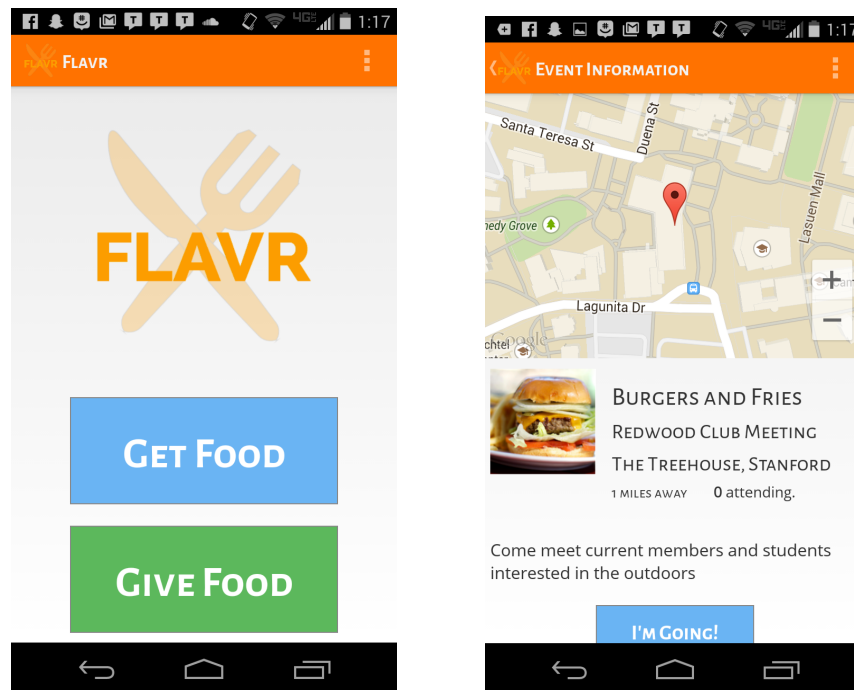
Final Project Report

Andrew Luo: design/development
Ben Ulmer: documentation/development
Alex Engel: user testing/development
Onkur Sen: manager/development

Value Proposition: Making it easy to find, share, and enjoy food instantly

Problem and Solution Overview

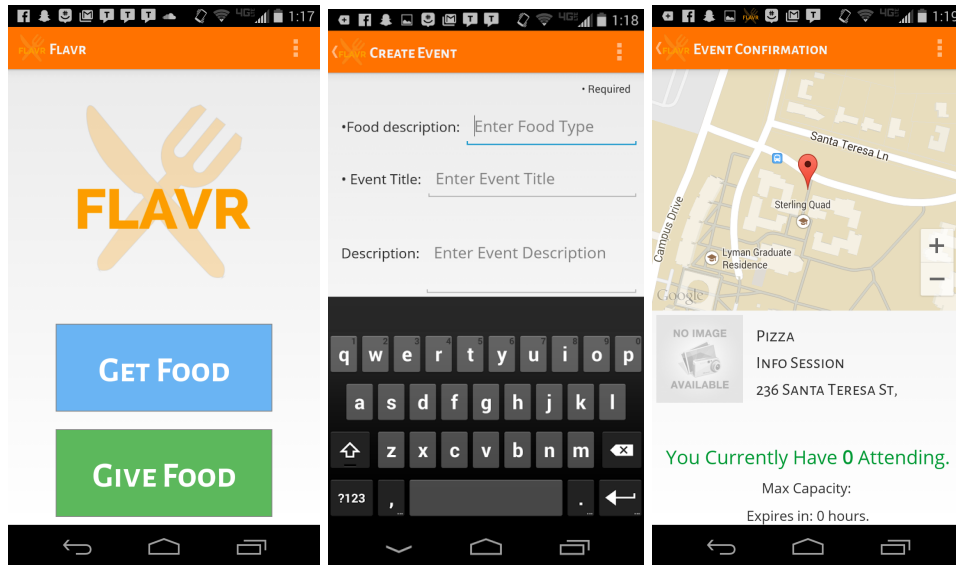
The words "free food" have been the bait for students attending countless events and gatherings across college campuses for good reason. Food is often high on a college student's agenda. However, events advertising free food are often met with an unenthusiastic turnout, spawning a host of problems ranging from food waste to unnecessarily hungry students. Although FreeFood@Stanford attempts to address these issues, it both lacks real time updates and is geared towards announced events, thus lacking the functionality to broadcast information on the impromptu leftovers that so often occur. Flavr remedies all of these problems by connecting students with food in real time through a more extensive food distribution network.



Figures: High Fidelity Example Interfaces

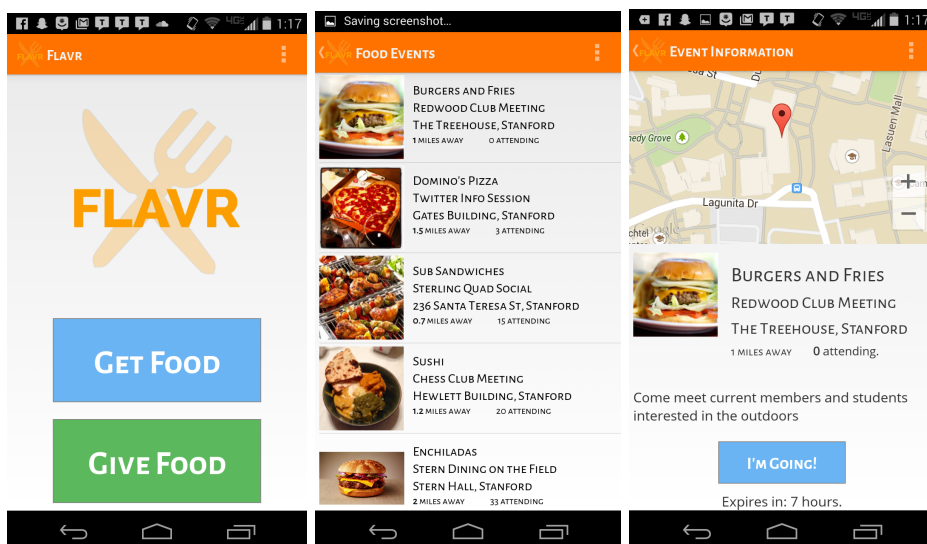
Tasks & Final Interface Scenarios

1. Our simple task is when providers post an event with information about what, when, and where food will be available. Once the event is posted, providers can get real-time information about the number of people who will go to the event.



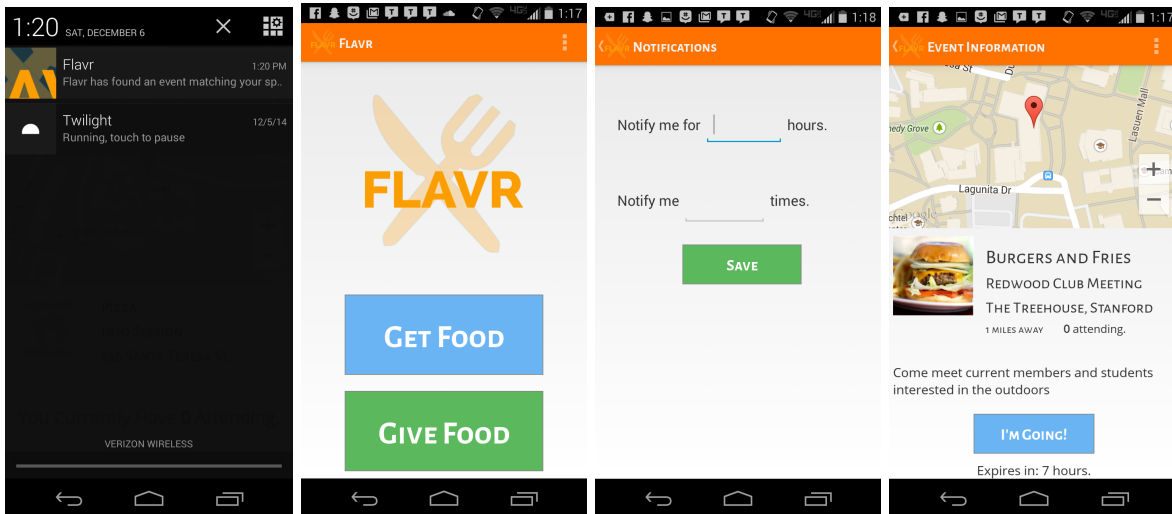
Figures: High Fidelity Simple Task Storyboard

2. Our moderate task is when students find free food on campus. The user must specify which type of food he or she wants, understand the directions to the food, and verify that they're coming. This task is of medium complexity as it requires the student to both find and respond to an event.



Figures: High Fidelity Moderate Task Storyboard

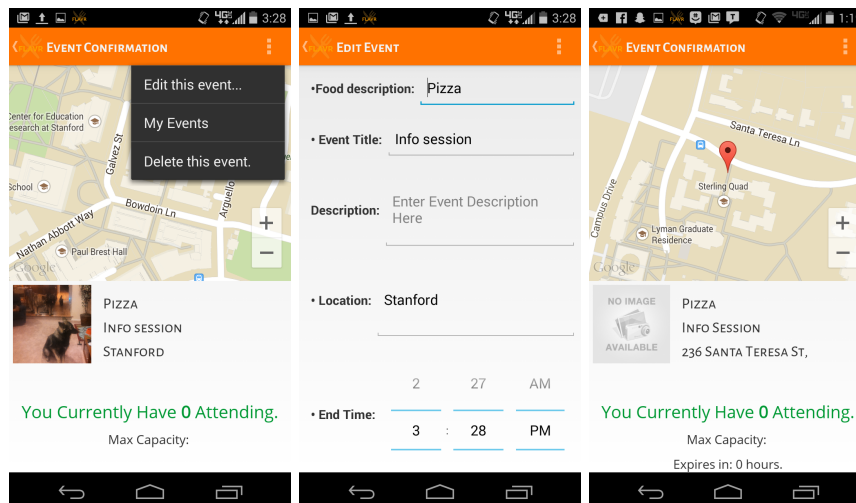
- Our complex task allows consumers to customize their experience, ranging from dietary preferences to restricting the radius within which results should be displayed. Furthermore, we allow users to set up a notification system, which will let them know if food becomes available in the near future.



Figures: High Fidelity Complex Task Storyboard

Major Usability Problems Addressed

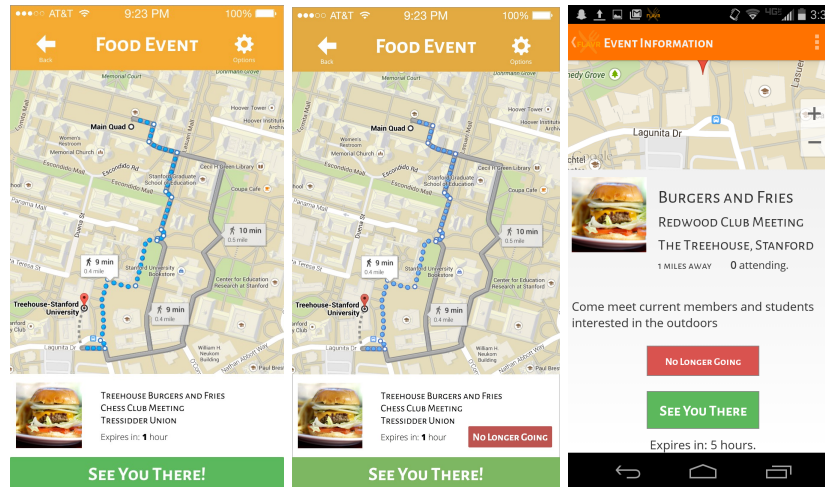
- [H2-3][4]: Event hosts could not edit events after creation.
 - Event hosts can now click an “Edit Event” button after creation to modify fields in case the host made a mistake or circumstances had changed.



Figures: High-Fi Edit Event User Flow

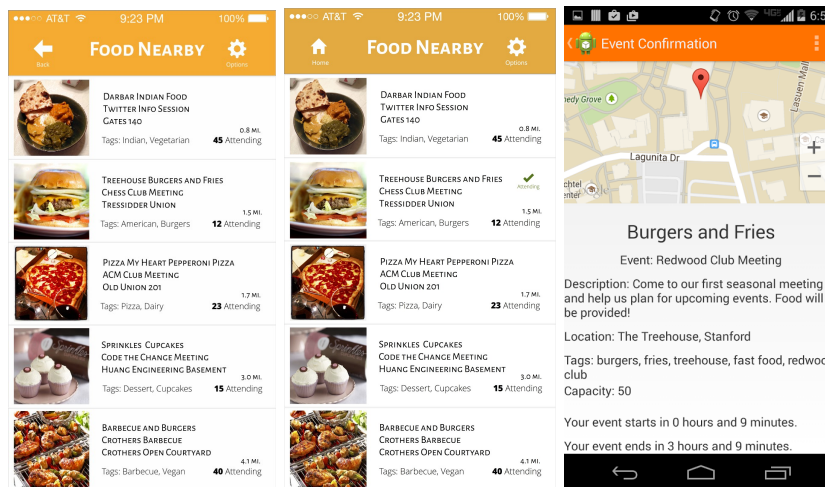
- [H2-3][3]: Consumers cannot undo attendance of events.

- We have now included a “No Longer Going” button after a consumer clicks “I’m Going” that allows undoing attendance of events.



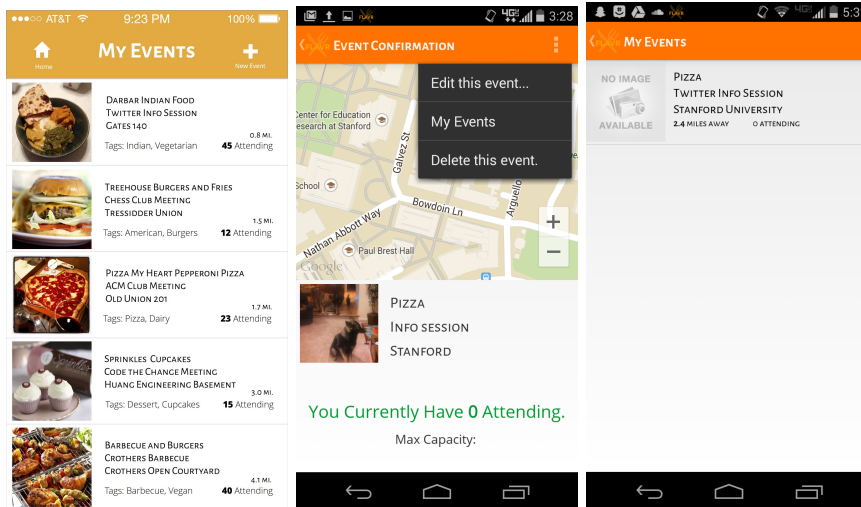
Figures: Medium Fi (Before Evaluation), Updated Redesign (After Evaluation), High-Fi

- [H2-1][3]: Listings of food events do not have an indicator for events that students are already attending.
 - Added a count of how many students are attending



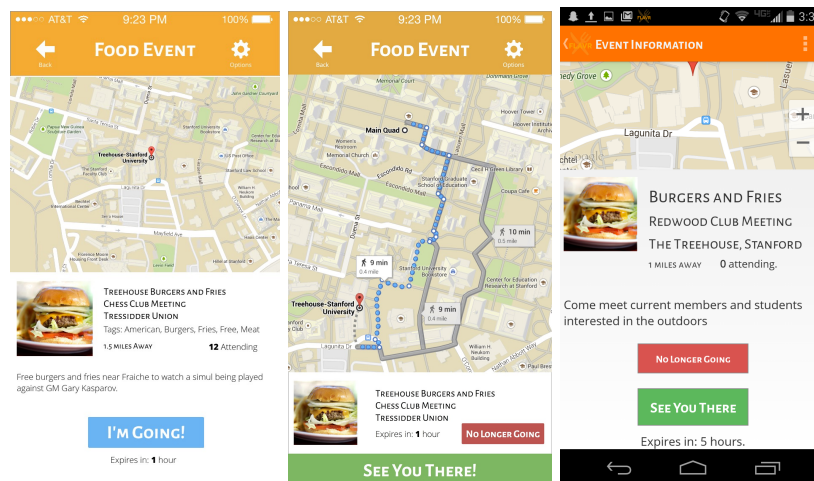
Figures: Medium Fi (Before Evaluation), Updated Redesign (After Evaluation), High-Fi

- [H2-6][4]: Event organizers cannot view previously created events.
 - Added a screen so event organizers can now see previously created events through a “My Events” menu.



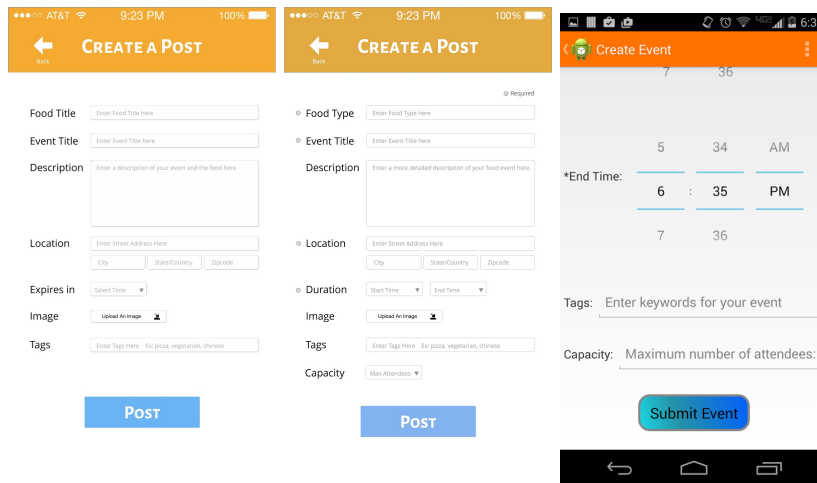
Figures: Updated Redesign (After Evaluation), High-Fi Flow

- [H2-1][3]: Consumers do not receive confirmation upon deciding to attend an event.
 - After clicking on the “I’m Going” button, consumers now receive a message saying “See You There!”, indicating that their attendance has been confirmed. In addition, the count for the number of people attending is visibly incremented to the user.



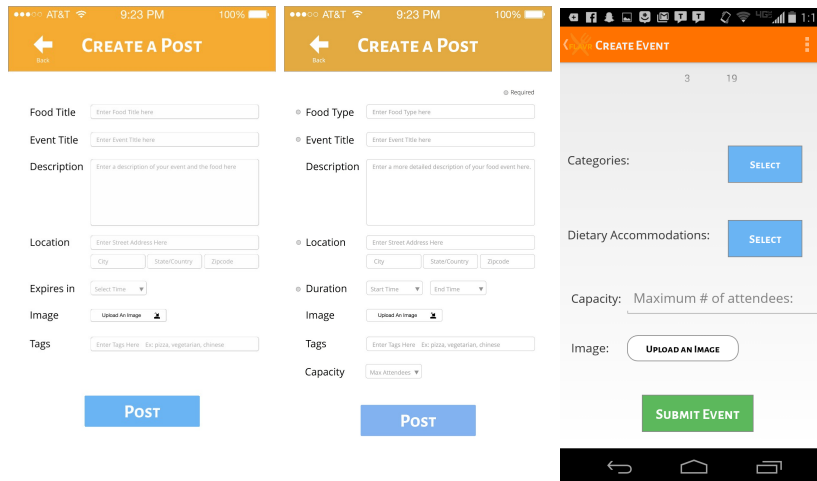
Figures: Confirmation Screen, Added Screen (After Evaluation)

- [H2-5][3]: Event hosts can only post expiration time rather than start and end times.
 - We decided to change expiration time to end time because we agreed that it might be more intuitive. We did not add start time, however, because the purpose of the application is for event hosts to post food when it is ready to be distributed as opposed to posting food ahead of time. This is a key differentiating factor from existing approaches such as FreeFoodStanford.



Figures: Med-Fi (Before Evaluation), Updated Redesign (After Evaluation), High-Fi

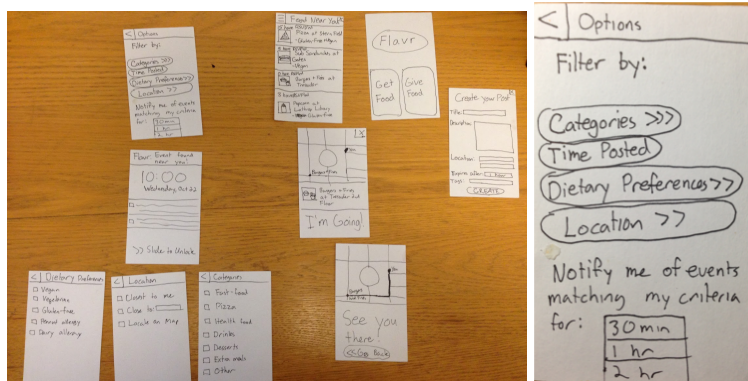
- [H2-5][3]: Event hosts cannot limit the number of people who can attend an event.
 - During event creation, event hosts can now set a maximum attendance. Furthermore, upon confirmation of event creation, hosts will receive live updates on the number of people attending.



Figures: Med-Fi (Before Evaluation), Updated Redesign (After Evaluation), High-Fi

Design Evolution

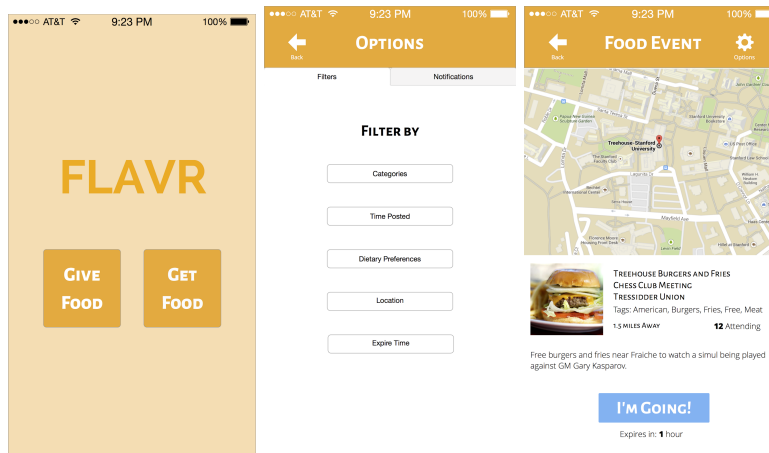
Low-Fi Prototype:



Figures: Low-Fi Prototype Images

Above you can see our initial design. We found that users were confused with options page and icons like the back button and the options menu. We therefore split the options menu into several separate pages for notifications and filtering, and used Android patterns for the icons.

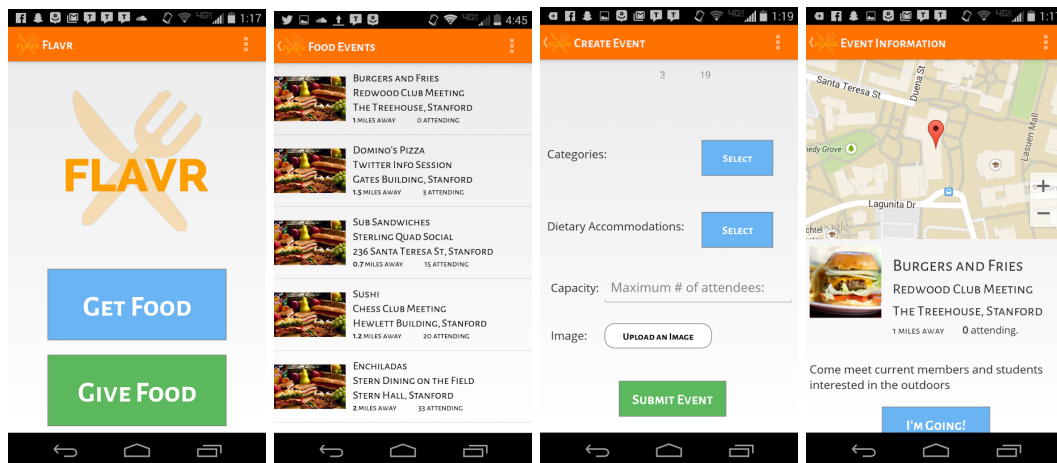
Medium-Fi Prototype:



Figures: Medium-Fi Prototype Screenshots

We responded to heuristic evaluation by adding a my events page and changing the buttons on the home screen to better distinguish between give and get food (see figure 6 below). We also added editing functionality for events and updated the event confirmation process for students. In terms of aesthetics, we adjusted our home screen by adding a logo and removing the background color.

Hi-Fi Prototype:



Figures: Hi-Fi Prototype Screenshots

Prototype Implementation

Tools

For our concept video, we used:

- Whiteboard markers to draw demo screens
- Canon Rebel T2i DSLR to record footage
- iMovie to edit and compile the footage into a complete video

For our lo-fi prototype, we used:

- Pencil and paper to draw out the various screens
- A text editor to take notes on user interactions with the prototype

For the medium-fi prototype, we used:

- Sketch to create the various screens for the application
- InVision to create a functioning prototype with hotspots connecting various screens.
- Bootstrap CSS icons for styling

For the hi-fi prototype, we used:

- Sketch for prototyping and logo design
- Android Studio as an IDE for developing an Android application
- The Google Maps API for showing the location of food events
- Github for collaboration among team members

Over the course of the project, we found several tools particularly useful. Sketch was a quick and painless way to create images for our medium-fi prototype, and InVision allowed us to easily and rapidly iterate through the flow of our app. Moreover, there's a

reason paper and pencil have been a mainstay in the design process through the years; we were able to quickly create and change designs without the pain or hassle of writing code or working on a computer.

On the other hand, some tools weren't as useful as we would have liked. We were new to Android Studio, and the learning curve slowed down our design process. The Google Maps API ran into issues with the demo functionality of Android Studio, which delayed development for those of us without Android devices.

Wizard of Oz

We do not use any Wizard of Oz techniques in our application.

Hard-Coded Data

Right now, we have hard-coded several events with relevant information (food, location, images, etc.), and all events are run on the client as opposed to being on a database.

Missing Features and Future Work

- We would like to add navigation instructions using the Google Maps API for a consumer when he/she decides to attend a food event.
- Currently, we assume a default user and have not created the user account flow. We would like to implement this by using either an email/password requirement or OAuth in combination with Facebook, Twitter, etc.
- Currently, an event host must specify a location where his/her event is located. We would like to default to the event host's current location since most event hosts will be posting from where their food is located.