Mutma Adebayo
Elisa Carrillo
Aashi Agrawal
Kanchan Krishna

Project Proposal

Pitch

Imagine you are craving that delicious, mouth watering meal from that one restaurant that your boyfriend's aunt's best friend's neighbor's yoga instructor told you about. Only one problem - you can't remember the restaurant name, the exact order you purchased, or where in the world it was. No worries! You can use our app to create a food diary for yourself by tracking your meals at each restaurant you visit. Feel free to share your delectable meals history with your friends and family, and don't hesitate to take advantage of the multitude of accessibility features found within the app.

Functionality (one tab for each of these)

- Users can create an account and a profile, log in, and log out using authentication services provided within the app
- Users can log their meals with specific information, such as the restaurant name, location, menu, what they ordered, and the transaction amount
- Users can access and view their previous entries
- Users can find restaurants near their current location, specifically those that are disability friendly, black owned, etc.
- Users can add/follow friends on their food diaries
- Users can share their restaurant history with their friends
- Users can provide feedback on each restaurant's service and quality of food
- Users will be able to take advantage of accessibility features such as text-to-speech
- (Potential) Users will be able to acquire points via the application's reward system and use those points to obtaining a free meal at a specific restaurant

Tables:

1. Username / login

Components

Frontend:

- React.JS/Javascript: a couple of our team members are already familiar with the language and framework so we thought it would be more efficient to proceed with a tech stack that most of us are familiar with
- HTML: we want to use HTML to assist us in the creation of forms, buttons, etc to help us create the foundation of the web pages
- CSS: we want to use CSS to help us add some basic styling to different aspects of our webpage
- Firebase for hosting: we want to use Firebase to host the website in order to deploy the web application

*To test the frontend components, we are going to compare our Figma designs to what our website actually portrays and fix any discrepancies.

Backend:

- Google Maps API: we will need to use the Google Maps API to help find restaurants that are near someone's location
- Geolocation API: we will need this to access the user's current location

- Javascript: we were thinking of using Javascript to write backend essential functions in our web app, such as loading the database of all the users entries
- Firebase for authentication: we will need this for the user to create their account and log in and out of the system

Database:

- Firebase (to store food and restaurant entries under each username)

*To test the backend components including the database, we are going to act as if we are the user and input information and see if the correct information shows up in the corresponding entry in the database for the test user we create. Then, we will see if the Google maps API is working by seeing if we truly get nearby restaurants. We will test the geolocation API and see if the location generated is accurate. Lastly, we will test the accessibility features by implementing the text-to-speech feature and seeing if the spoken words are correct and match the screen text.

Continuous Integration

- We are going to store our code on a Github repository and conduct pull requests
- We will use the Javascript style guide found on Google (https://google.github.io/styleguide/jsguide.html) and will continuously refer to this as we are not all well-versed in Javascript
- We will be utilizing the React testing library to write automated tests to check the functionality of our front end and back end components. We will also be conducting manual tests on both components as well. The React testing library also provides a functionality to retrieve code code coverage.
- Our pull request workflow will consist of us reviewing each other's pull requests after we have made changes to the code. We are going to go in a circle; for example Elisa can review Kanchan's, who will review Mutma's, who will review Aashi's, who will review Elisa's.

Schedule

*Our schedule is based on the different features listened in the Functionality section; each week, we will strive to work on a different feature and assign different features to each person based on their interests

Week 1: Draw out website design, Setup project, create (2?) GitHub repo(s) - make one for now, create Figma mockups, decide if going to use virtual environment
Week 2: Authentication functionality (firebase), create and test form for diary entries that should be saved to the database under each user
Week 3: Ensure users can access and view their previous entries; Users can share their restaurant history with their friends
Week 4: Users can find restaurants near their current location, specifically those that are disability friendly, black owned, etc.
Week 5: Users will be able to add/follow friends on their food diaries
Week 6: Add in accessibility features like text to speech
Week 7: link to food apps, feeling lucky feature (give users a random restaurant near them to try based on their past interests?)
Week 8: Work on deploying the app to the cloud

Risks

- The authentication part of our project may not work the first time; not all of us know how to use Firebase for creating user accounts and logging them in and out. Our backup plan is to do some research and use Stack Overflow to help us debug our programming issues.

- This is a general case, but we may fall short on a task during an abnormally busy week. Our backup plan is to dedicate extra time to the project over the weekend, so by the time Friday arrives, we are either on schedule or a little bit ahead.
- The application may crash if we do not integrate the APIs correctly. Our first backup plan is to work together to debug the issues, but we also plan to ask our mentor for supplementary guidance.

Teamwork

- In order to divide the assignments, we have decided to focus on the aspects of the project in which each of us are most interested in. For example, Aashi will be working on the front end. We are also going to work according to the weekly plan under the Schedule heading and divide the tasks up as each week approaches. For the coming week, we are all going to work together to draw the web page design and create the Figma mockups. If someone finishes a task they chose early, they could either help a team member with their current task or move on to the next feature to be completed in the next week in the schedule.
- We will be working on our own branches within GitHub to avoid conflicting code and increase efficiency. We will also be requiring merge requests before any code can be merged to the master branch to ensure that the code we write in our own individual branches are approved by other members of the team.
- In order to maintain a consistent developer environment, we are looking into using Docker containers as we did in CS128.

Diagram