ambi

High-Fidelity Prototype README

melissa lee | joseph ngo | kaitlin peng | faith zhang

High-Fi Prototype

We used React Native and Expo to develop and build our high-fi prototype, as well as our personal devices and the Apple Xcode simulator for testing.

How To Use

ambi can be accessed through an iOS device, Android, or web browser. We recommend running it on an iOS if possible.

To run ambi on your device:

1. Install the ExpoGo app on your phone
2. Log into Expo Go with
   a. **Username:** ambics147
   b. **Password:** ambics147
3. Scan the QR code below with your mobile device, or go to exp://u.expo.dev/update/bd712483-2256-475f-8508-020efe57e3a8 on your phone browser

To run from ambi's code repository:

1. Clone our [git repository](#)
2. cd into the "frontend" folder
3. Run "npm install"
4. Run "npx expo start"
5. Download the ExpoGo app if you haven't yet.
6. Scan the QR code

Limitations

Although we are proud and impressed by how far we have come with our hi-fi prototype, some parts of the design were not able to be implemented due to our time constraints. For these features, we have chosen to either hardcode or simulate them for the purpose of demonstration.

Wizard of Oz Techniques

The following components of the app have been simulated using the Wizard of Oz technique:

1. **Discovery page sorting algorithm**

In the final product, a sophisticated location-based algorithm would be used to organize a user's discovery page based on their proximity to restaurants. In this high-fi prototype, that functionality is simulated by displaying content in chronological order.

2. **Endless Discovery page**

   In our high-fi prototype, the Discovery page continuously offers an endless stream of content, ensuring the user is always entertained without interruption. This would not always be the case in our final product.

3. **Flawless execution of the QR code scanner**

   To ensure a smooth prototyping experience, our QR code scanner succeeds with every attempt. This allows our prototype to deliver the results users seek without the hassle of needing to decode QR codes.

Hard-Coded Elements

The following components of the app have been hard-coded for demonstration purposes:

1. Restaurant landing pages
   a. Video content, word clouds, busiest times, fit checks
2. Example posts on the Discovery and Following tabs
3. Profile pages, including all content displayed