

Feast ReadMe

Dan Shiferaw, Peter Lowe, and Megan Wilson

InVision Prototype Link: <https://invis.io/TJ96VOPU9>

Using the Medium-Fi *Feast* Prototype:

- Our medium-fi prototype is designed to simulate user interaction with *Feast* to obtain nutrition information about their food, keep track of this food on their nutrition log, and share this food with others.
- From the home screen, the user is directed to rotate their phone to begin the scanning process. The user first gets instructions for scanning food through a series of instruction screens and scans a burger into the app. The prototype then confirms its guess for what the burger is (indicating restaurant and name of the burger from the restaurant menu) and informs the tester of its nutrition information.
- After obtaining nutrition information about the burger, the user may elect to add the food to their log and/or share their food through their social media accounts. The prototype then supports the user viewing their log as well as selecting which social media accounts to share to and viewing a mockup of their published result on the social media account of their choosing.
- Throughout the scanning, logging, and sharing process, the prototype supports the user deciding to return to the home screen by clicking the “x” button and accessing the settings by clicking the gear icon in the bottom right corner. The prototype also allows the user to click “View Log” at any time from the home screen to access their log without adding an item to it.
- Finally, the app supports the user adding a food to their log without scanning it using the “custom add” feature. Here, the user indicates the type of food, confirms the serving size, obtains nutrition information about the food, and adds the food to the log.

Notes:

- Due to the fact that InVision does not react to orientation changes, we were unable to make our medium-fi prototype respond to the user’s rotation of their phone to the landscape position. Instead, the user must tap the screen in our medium-fi prototype to trigger scanning to begin. In our final app, the movement of the phone from portrait to landscape position will cause the app to open the scan screen immediately.
- In order to make sure that our prototype flows from task to task, we had to hardcode a lot of the user inputs into our prototype. For example, we put images of a burger in our prototype rather than taking a real-time scan, specified the location of the user despite not knowing where this prototype will be tested, and reported nutrition information that was generated ahead of time rather than at time of use of the app.
- The prototype does not yet support linking with the user’s social media accounts. As a result, when the user shares their food on the prototype and elects to view their post,

they view it through the lens of one of the app creator's social media profile rather than their own. In the final version, electing to view the post will allow the user to view their post through the lens of their own social media account.

- The social media share screen also has a keyboard to personalize the post, but this does not work in this implementation.
- Although we have screens to configure options and settings, most changes made are not persistent (or the screens aren't fully interactive) due to the number of duplicate screens and paths needed to generate and track this behavior.
- The manually-added spaghetti doesn't persist if the user returns to the home screen (we used this as a simply way to clear the prototype so it can be easily replayed)