goFIT: Medium-Fi Prototype README

Bryce Tham | Christina Ramsey | Olivia Gregory | Denis Russu

The goFIT prototype itself can be found here.

Tools

The goFIT prototype was designed in Sketch and runs via InVision. It was written in Sketch after Bryce attended the workshop; Olivia had had prior experience with InVision and added the majority of the prototype transitions. The icons used in the app came from Flaticon.

Operating Instructions

General Notes:

InVision is a platform that emulates a mobile app; clicking with the mouse represents a user pressing buttons with their finger.

Full functionality has not yet been implemented; not every button has a transition screen associated with it yet. Enough buttons are active to show the screen flow (which will be very similar throughout actions) for select representative tasks. You can read more about this in the *Limitations* section below.

Task Completion

We focused on having three key tasks that users can implement. We made them more specific for testing purposes, but the general tasks are as follows:

1. Add a personal challenge.
2. Invite friends to complete challenges.
3. Log your progress on current challenges.

There are multiple ways to complete some of the tasks. Sample instructions on how to complete these tasks are outlined at the bottom of this file.

Limitations

One major limitation is that InVision doesn't allow the full range of interactive motions that a user would have with their phone. For example, we had the idea that users could swipe left and right to cycle through the challenges on the main page; InVision can't do this, so we opted for buttons in the prototype.

Another limitation is that InVision doesn't have the capacity to store information in memory. All screen transitions are hard coded; that's why updating something on the screen (i.e. incrementing the walking challenge) will only update that immediate screen, not successive ones, within the app.

We have not yet implemented the full functionality of our app in this prototype. Many of the screens we did not build would be very similar to the ones we did build (i.e. highlighting particular icons when choosing an activity or friend, opening friend profiles that would display almost identical information, and opening challenges that would display almost identical information), so because of time constraints we decided to only implement what would mimic basic functionality, especially in such a way that demonstrates how this would pertain to our app.

We also wanted to focus primarily on tasks and uses for the app, and less so on static screens. For this reason, we did not yet build a user profile or personal history/progress screen, which we plan on including in future iterations.

Wizard of Oz Techniques

We currently don't have any users, so all of our challenges, challenge progress, friends, and friend profiles are hard coded. As stated above, only select, representative screen flows have been implemented.

Specific screen transitions have taken the place of storing actual memory (if a progress bar changes, that is because we have taken the tester to a new screen), which is why updates do not stay constant throughout the app experience.

How to Complete the Tasks

The following tasks were made more specific for testing purposes.

1. Add a personal walking challenge that ends this week.
    a. Home screen -> Challenge button (top left) -> Add new challenge (bottom plus button) -> Walking icon -> Next -> Me -> Next -> This week -> Go
2. Invite Christina to a swimming challenge that ends next week.
    a. Home screen -> Challenge button (top left) -> Add new challenge (bottom plus button) -> Swimming icon -> Next -> Christina -> Next -> Next week -> Go
    b. Home screen -> Social button (top right) -> Christina -> Add new challenge (plus button) -> Swimming Icon -> Next -> Next week -> Go
3. Log that you went for a walk today.
    a. Home Screen (Should be showing your current walking challenge) -> Plus
    b. Home Screen -> Challenge button (top left) -> Walking Challenge (top left) -> Plus