Medium-Fidelity Prototype

Link

https://www.figma.com/proto/JEnVtlaCrdu9X5zuOdsBKz/Prototype

How to Use

We built our medium-fidelity prototype in Figma, and target the iPhone 11's screen size. Our interface is not platform specific; although we included iOS graphical elements in our prototype, we do not rely on any particular phone features in our design (e.g., notch, back button).

To use our prototype, click on the Reclaim item from the home screen. From there, you will be brought into the application's main home screen. From there, you should easily be able to complete the four main tasks. If you click on a non-interactive element, Figma will highlight the interactive elements. Some of our task flows require filling out interactive forms; when appropriate, simply tap on an interactive element to fill it out (and, if a keyboard or select element pops up, tap on it as well). Note that in our actual prototype, you would be able to fill out the fields in any order, but this is not possible in our medium-fi prototype.

You can jump to different major parts of the interface using the bottom menu. The only requests that 'you' are involved in are 'CS107 Textbook' (as creator) and 'Desk Lamp' (as seller). Your name is 'Sarah'.

Tasks

Note that because of limitations of medium-fi prototyping an app heavily reliant on user-generated content, all of our tasks have some degree of hard-coding. (See *Limitations* and *Hard-coded Features* for more information.) These tasks—as well as their hard-coded elements—are summarized below.

1. Post a request for an item. You will create a post for 'CS107 Textbook.'
2. Post an offer for someone else's request. You will post an offer on the 'Desk Lamp' request.

3. Compare and respond to offers on your request. You will do this on the 'CS107 Textbook' request (from task 1); you can either do this immediately after creating the request, or by navigating to the offers from the `Your Posts` page or the notifications view.
4. Add an item to your watchlist. You will add 'String lights' to your watchlist.

Limitations

- The notifications view is static; it does not update in response to actions elsewhere in the app.
- The final task—adding an item to your watchlist—has no 'follow-up.' You will not later see a notification that someone requested the item you added.
- Edit buttons are not functional.
- All task-critical interface elements are implemented, however some 'tangential' functions—such as withdrawing an offer—are not.
- Our account view is static; although you can *start* editing your information, there is no way to input or save any changes.
- Searching and sorting in the home view is not implemented.
- In certain flows, the 'back' button takes you to the previous point in *time*, not the previous view. It could also take you further back than you think. This is a consequence of Figma's 'back' implementation, which does not always align with the structure of our prototype. Our high-fidelity prototype will not operate this way.
- In interactive views, the fields can only be filled out in a single order. In our actual prototype, you will be able to fill out the fields in any order.
- There are some interactive views where a keyboard pops up to indicate typing. In longer interactive flows, we do not include this keyboard because it would be annoying/cumbersome for the prototype user. Regardless of whether a keyboard is shown, context should make it clear whether typing is expected. See the instructions in 'How to Use' for more information about filling out interactive fields.

We justify these limitations and decisions in our presentation slides.

'Wizard of Oz' Techniques

- At steps in tasks where other users' actions are required (e.g., posting an offer to the main user's request), we 'fake' the waiting period by transitioning to the next screen automatically after a delay.
- We 'magically' update the relative times (e.g., '5 mins ago') in the application on views that require the passing of time.
- In input views, we autofill all content automatically after the user taps on the interactive element; Figma does not support dynamic input.

Hard-Coded Features

All user-generated content is hard-coded, including:

- Requests
- Offers
- Offer decisions
- Account settings
- Notifications
- Watchlist items