

**CS147: Introduction to Human-Computer Interaction**  
**HouseMates Final Report and Hi-Fi Prototype**

[Website link](#)



# HouseMates

Ben Barnett | Software Engineer  
Sean Chang | Software Engineer  
Jeffrey Chen | Software Engineer  
Ali Sarilgan | Product Manager

## Value Proposition

Connecting **home-seekers**  
&  
**Help-seekers**



## Problem & Solution Statement

Throughout our user interviews we saw that elderly people and those with physical disabilities had an overwhelming need for helpers: friends, family and even strangers who would take care of their physical needs including laundry, shopping, cleaning, cooking and daily chores. On the other end of the spectrum, we met with several young professionals and students who struggled to find affordable housing in areas they lived in.

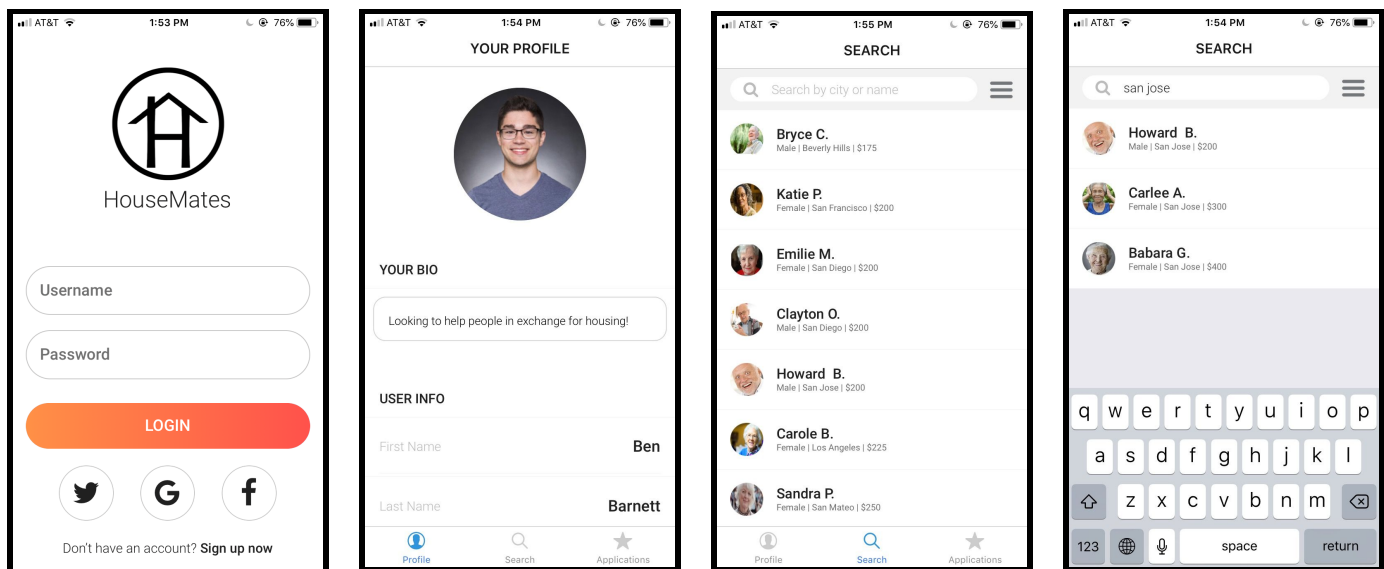
To address these seemingly disparate problems, we created HouseMates: a mobile platform that aims to connect home-seekers with help-seekers. A young, able-bodied person can utilize our platform to browse houses and connect with help-seekers in their city. The elderly home-owner can invite such young people to live in their house rent-free or heavily discounted in exchange for them performing basic chores and tasks.

## Tasks

Throughout the quarter we focused on both the home-seeker and the help-seeker and designed solutions to fit their unique needs and ultimately we decided that two different apps would be required to cater to their sensibilities. When the young and tech-savvy home-seeker values professionalism and aesthetic design, the old help-seeker wants simplicity. We will elaborate on our design process and sketches throughout this discussion but our final prototype and three tasks represent the home-seeker side of the two-sided platform.

### Simple: Log in and search for a housing opportunity

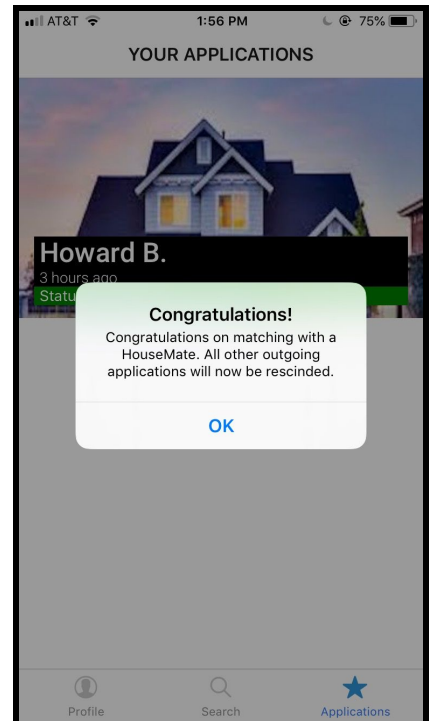
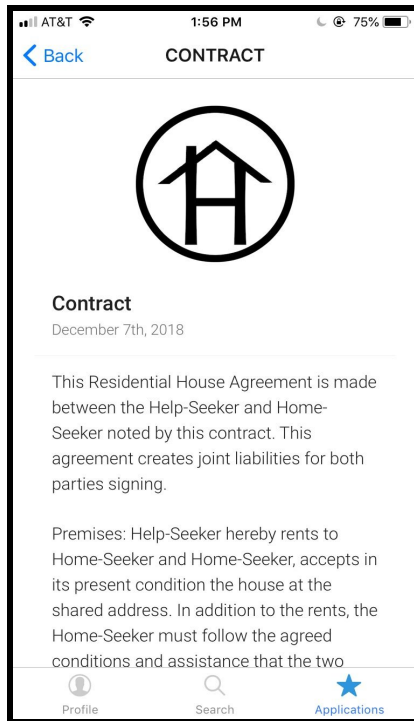
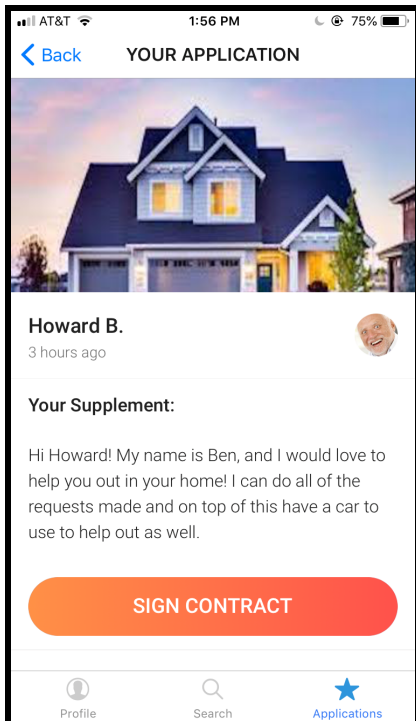
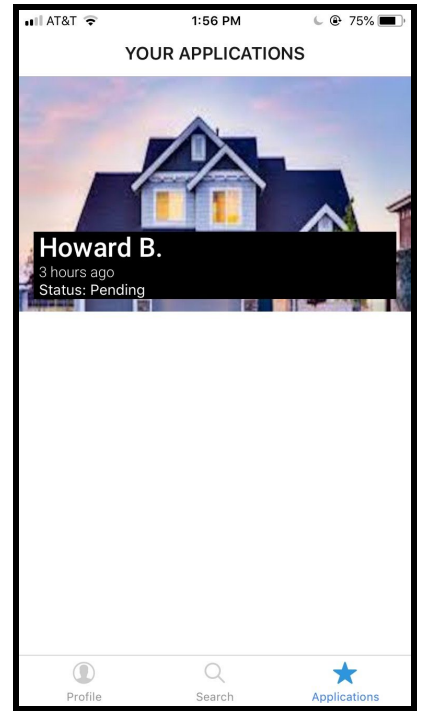
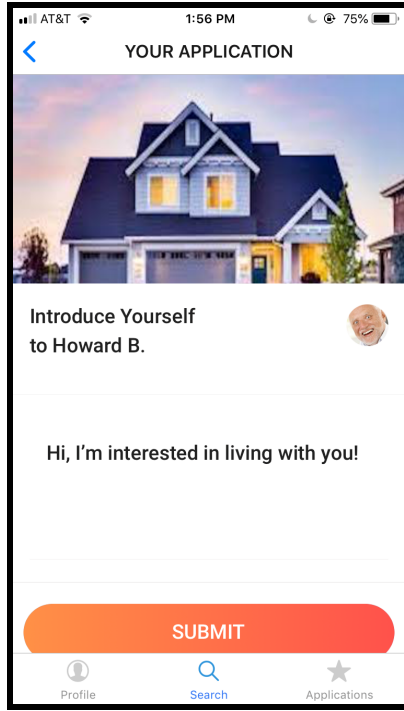
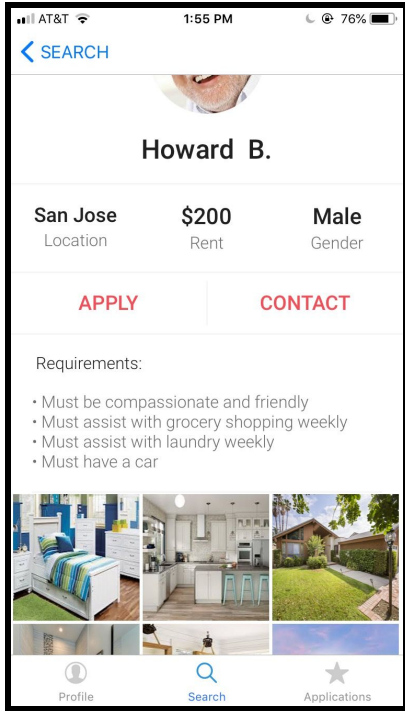
After logging in the user will be taken to the first tab, Profile where they can edit any aspect of their personal information. The simplest task will be clicking the “Search” tab on the bottom menu and searching for different housing opportunities. They can further utilize our search function by searching for city’s or names of hosts in the textbox or by clicking the filter icon to the right of the textbox and filtering y different characteristics, such as gender. This task is at the core of our platform, since users cannot find housing if they don’t know how to search for it, so we chose to include it to test our main assumption of ease-of-use and simplicity.



### Moderate: Apply to a house and sign a contract

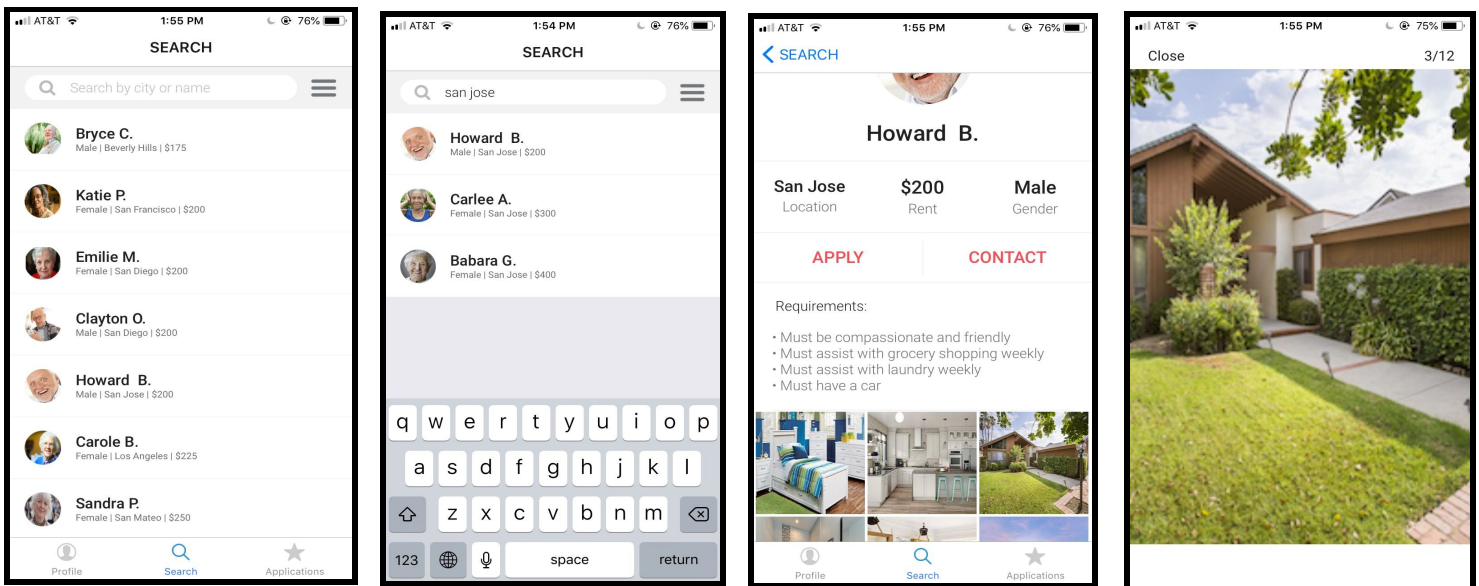
To apply to a house the home-seeker goes on to the profile of the help-seeker that they would like to live with and click on the apply button. Once clicked, the app will take them to their application where they can write the help-seeker a personalized message and apply. Once applied, the home-seeker will be taken to the applications tab on the right-most tab of the bottom menu. If their application is approved, they will be taken to their original application where they can click “Sign Contract,” and review and validate the contact. Once they have signed the

agreement they will see a congratulatory message. During our interviews, we saw that formalizing the agreement was a concern many people had and it represent the end of our user-flow so we chose to include it as our moderate task.



## Complex: Compare three different housing opportunities and weigh pro/cons

As our simple task tackled the beginning of the process and our moderate task tackled the ending, our complex task fills the gap by asking the user to compare different housing opportunities and weigh their options. From the search bar the user can click on each different person and examine their profile where they'll see the requirements of the help-seeker (such as must assist with laundry weekly,) the rent that they're charging and pictures of their homes. Once they are on the profile they can go back to the search tab and simply click on another profile. Similarly, when they're examining pictures they can simply click "close" (located at the same place as the back function to Search) and go back to the user profile. We chose this as our complex task since it requires the most initiative and mental concentration from the user as they will be expected to weigh the pros and cons of different living options.

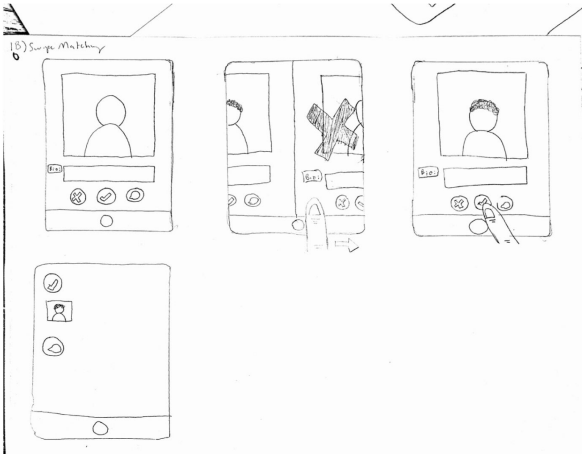


## Desing Evolution

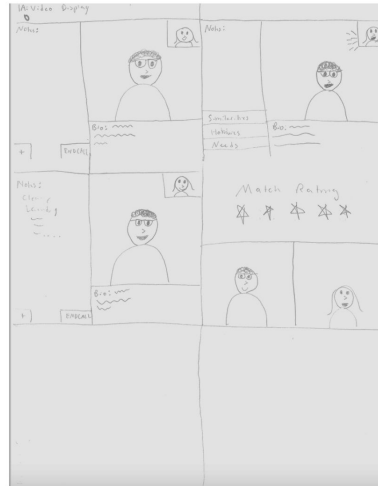
### Initial Platform Explorations

At the end of our needfinding interviews and POV/HMW explorations we decided on the idea of HouseMates but before we even began our low-fi prototyping we visualized what the best medium to create this service would be. A VR service where users could tour houses, a video chat where users could meet and greet one another, a Tinder-like matching service between home and help seekers, a wearable device, or a search and apply based application? We drew all of these ideas out and evaluated the merits of these ideas.

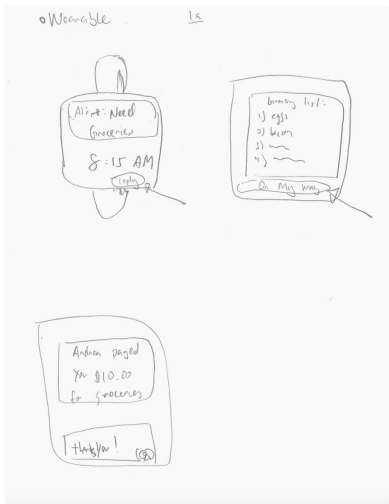
# Swipe



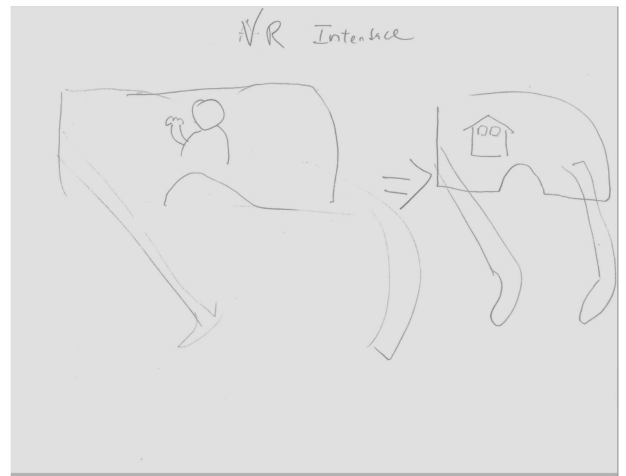
# Video Meet and Greet



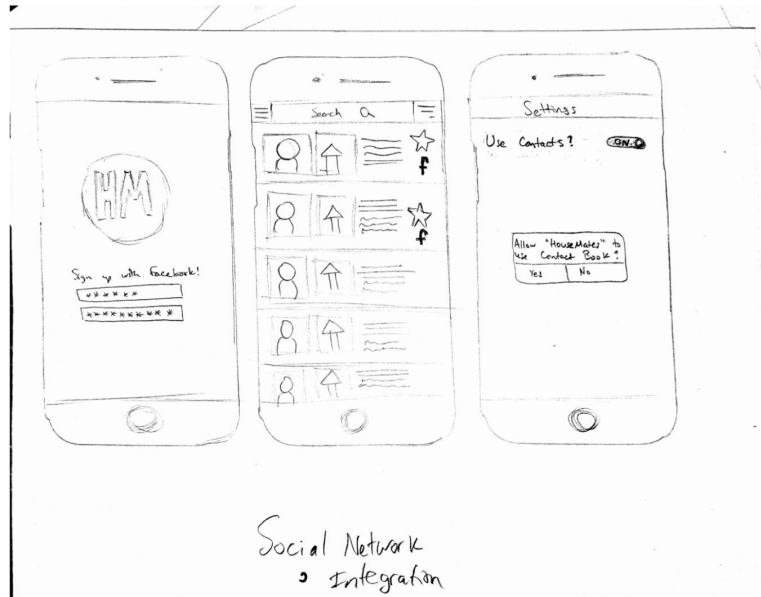
# Wearable Devices



# VR



# Search and Apply



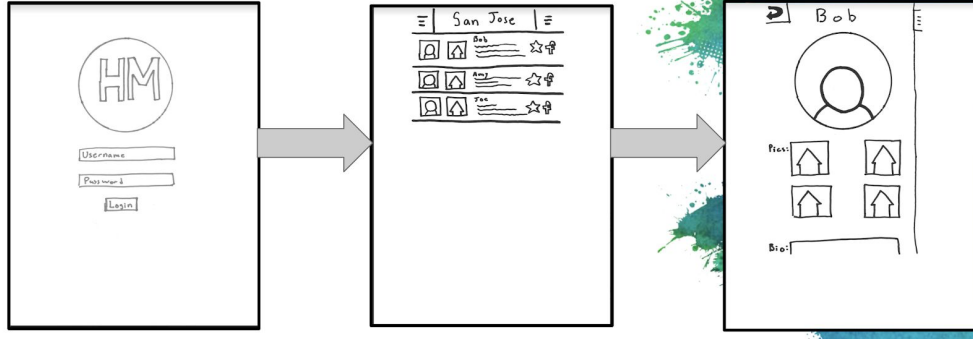
## Platform Selection and Low-Fi Prototype with 3 Tasks

We took these 5 designs into user testing and asked users to interact with screens we had created. These users reacted the best to the idea of search and apply based selection mechanism that was based on a mobile app so we re-hashed the idea and created a low-fi prototype of it along with our three tasks that we could further test with strangers at a coffee shop.



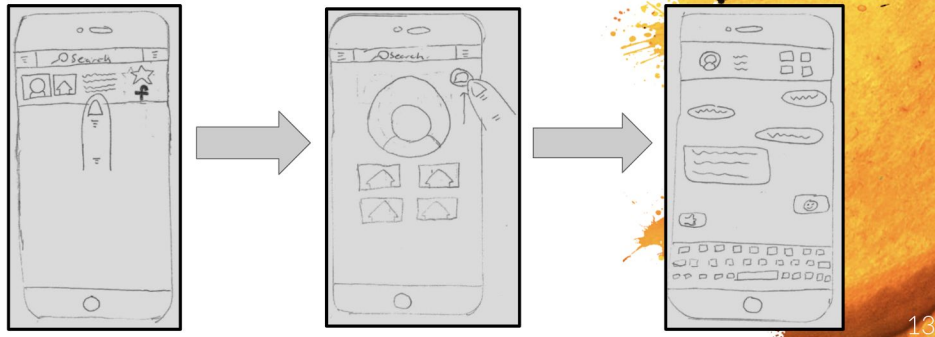
# Simple Task

Tom searches for housing opportunity



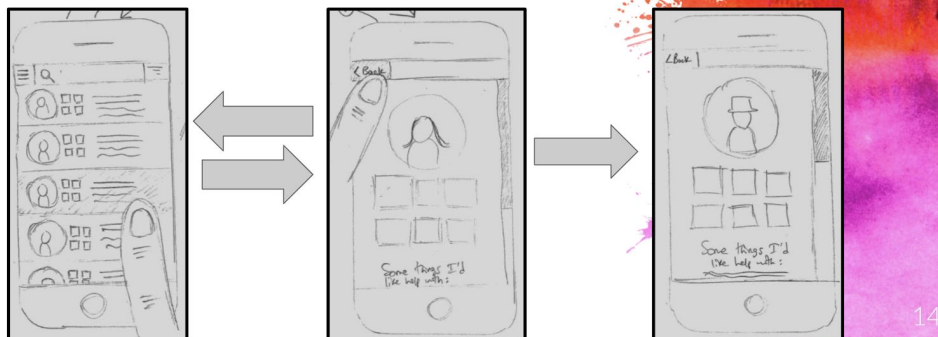
# Moderate Task

Tom and Ernest go over expectations of each other and confirm a contract to formalize their agreement.



# Complex Task

Tom compares 3 different housing opportunities and weights the pros and cons

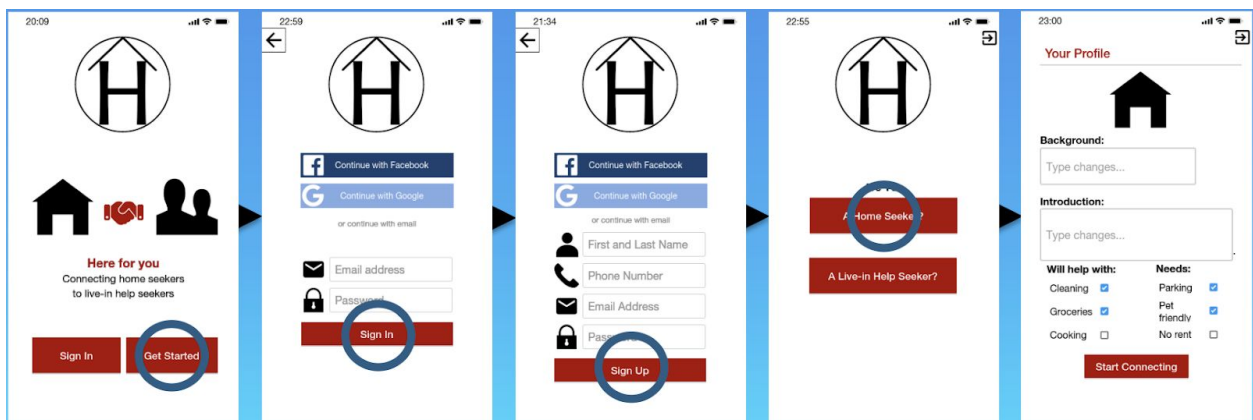


## Medium-Fi Prototyping

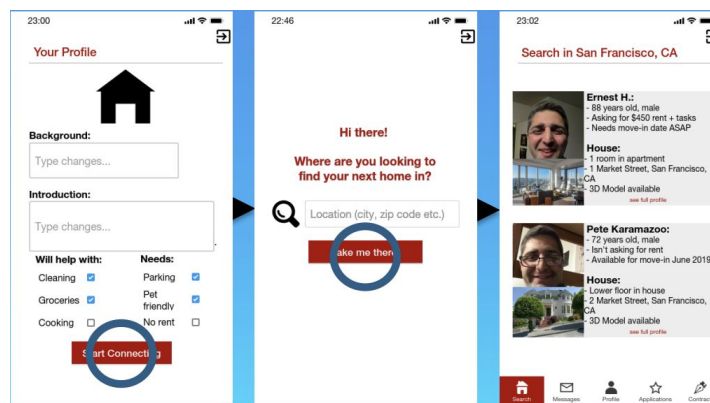
Throughout this user testing we learned that users really valued simplicity and flexibility. They wanted to be able to edit their own wishes and expectations from a potential housemate, and wanted a simple UI that was non-distracting.

Furthermore, our users suggested that we add a bottom menu bar that would constantly remind them where they were within the flow and give them the option to navigate seamlessly.

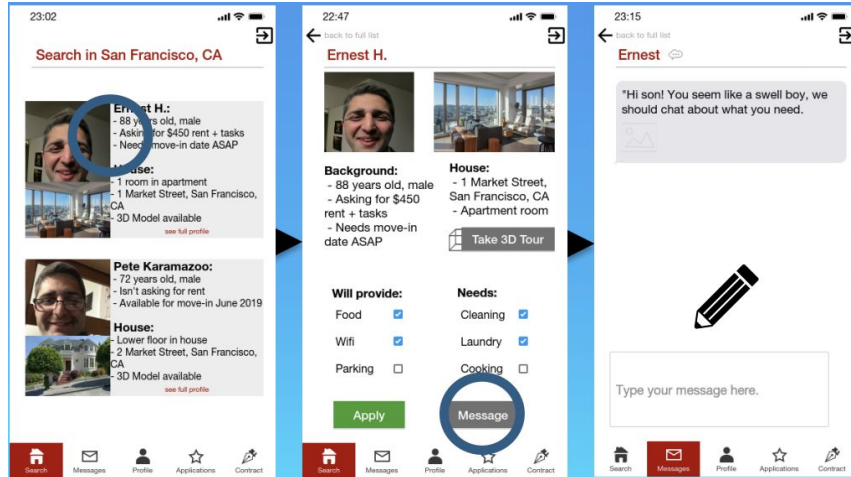
And finally, our users told us that they wanted detailed information on both the home-seeker and the help-seeker that was immediately visible. Taking all of their suggestions to heart we went to Proto.io and created a single app that represented the home-seeker and help-seeker flow together.



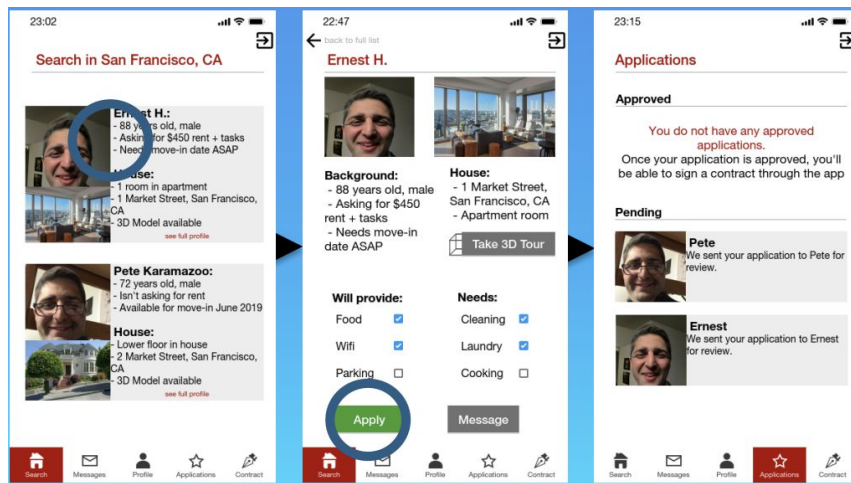
Home-seeker login



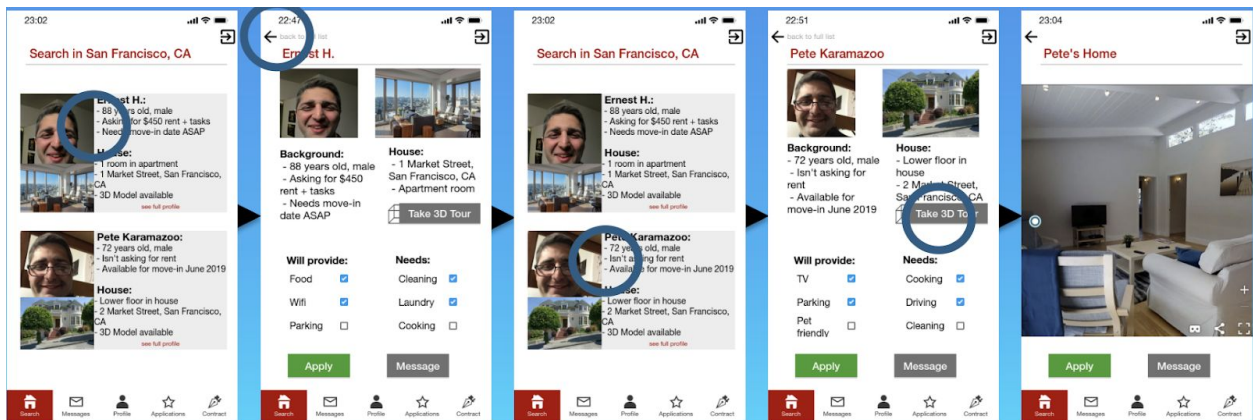
Home-seeker search initialization



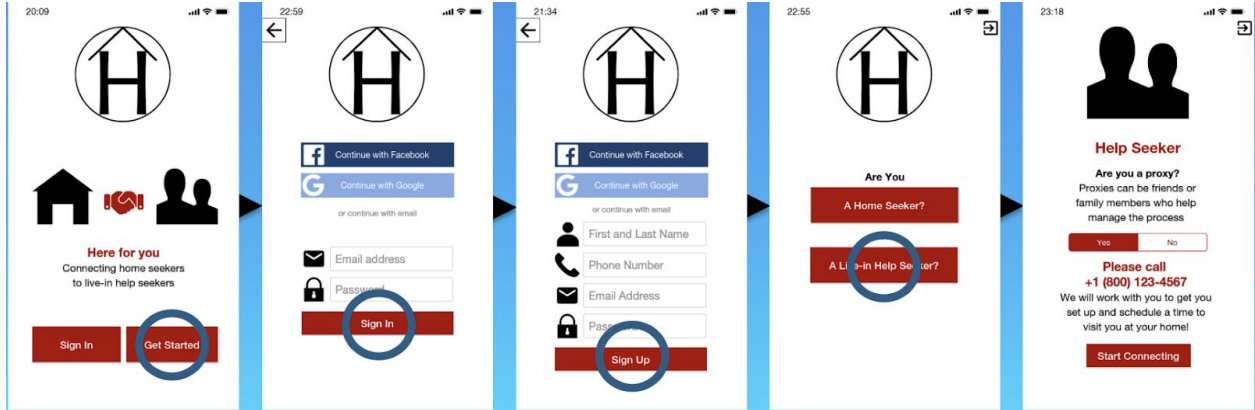
Home-seeker search and message interesting property owners



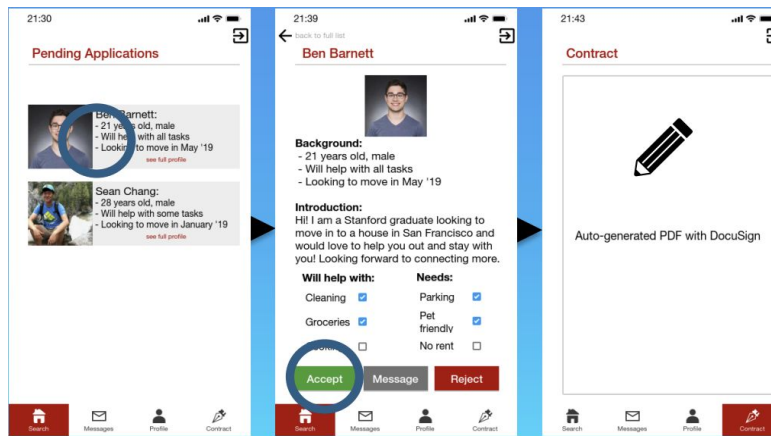
Home-seeker apply



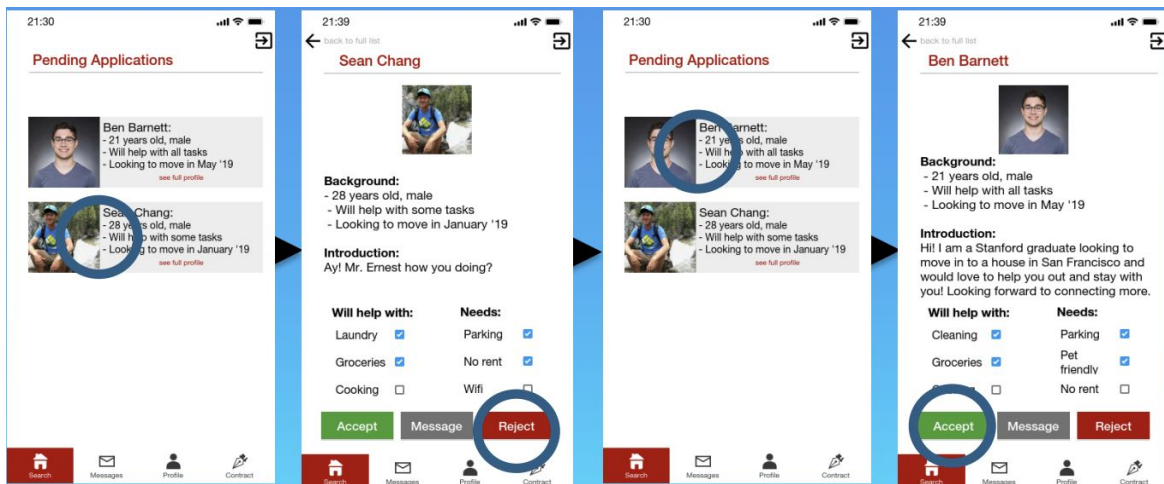
Home-seeker examine apartment



Help-seeker login



Help-seeker view application



Help-seeker accept application

## **Major Usability Problem Addressed**

We received a lot of great quality feedback from our peers during the heuristic evaluation and decided to fix all of their Severity 3 & 4 error suggestions attached below:

### **[H3. User control and freedom] / Severity 3**

When signing up, if the user accidentally clicks “home seeker,” the user cannot return to the previous screen (and vice versa for clicking “home seeker”).

### **[H5. Error prevention] / Severity 4**

When a home-seeker clicks “Apply”, the user’s application is immediately submitted.

### **[H9. Help users recognize, diagnose, and recover from errors] / Severity 3**

Currently, it seems that the list of home-seeker’s applications are not clickable. As a continuation of the previous point, if a home-seeker realizes that their application had a typo, there is no way for them to edit or withdraw their application.

### **[H5. Error prevention] / Severity 3**

When creating a profile, it is unclear how much/what type of information is expected in the Background, Introduction, and House sections.

### **[H8. Aesthetic and minimalist design] / Severity 3**

There is too much information on the Help Seeker - Proxy screen, which makes it content confusing. It seems like “Start Connecting” will immediately auto dial the phone number above, rather than take you to the interface of the app.

### **[H7. Flexibility and efficiency of user] / Severity 4**

When editing the profile of a help-seeker, there is no expectation or way to attach photos of your house to your profile.

### **[H5. Error prevention] / Severity 3**

User isn’t required to set proxy setting before pressing start connecting, could make a mistake with no way to go back.

### **[H3. User control and freedom] / Severity 4**

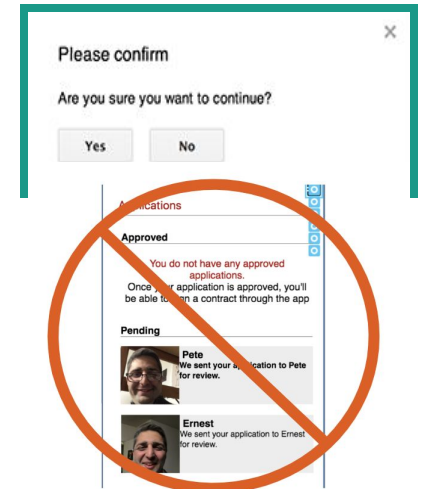
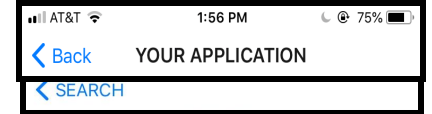
On Home Seeker screens, user unable to change locations after initial setup Fix: Allow user to go back and change search location.

After discussing this feedback amongst ourselves and our evaluators we saw three major themes and implemented changes to adress all of these issues:

## 1) Error Prevention

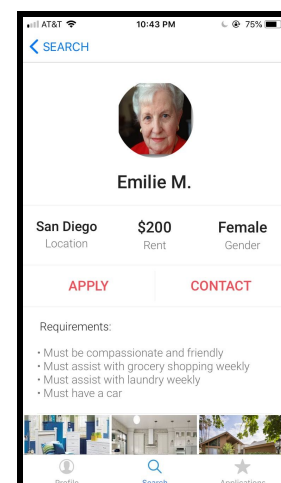
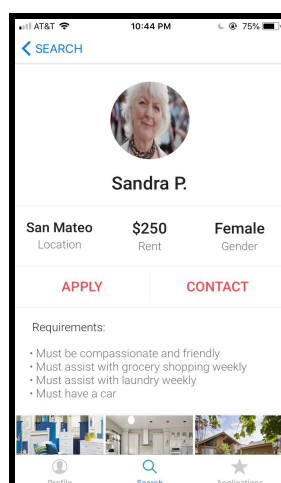
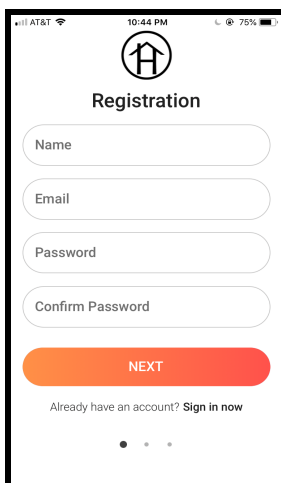
We learned from our evaluators and user testing that they were confused by the lack of clear back buttons. They felt like they had no way of going back once they've entered the wrong part of the flow by mistake. As a result, we added back buttons and back icons to all our pages with consistent design.

Furthermore they did not have the ability to edit their information, or submitted applications, and did not like that once they click apply their application was immediately submitted and they were taken to the pending applications page. We fixed these issues by implementing a confirmation segment where an application would not be submitted without a second ask and users always have the power to go and edit back. We also reduced the tab size from 5 to 3 for added convinience and simplicity.



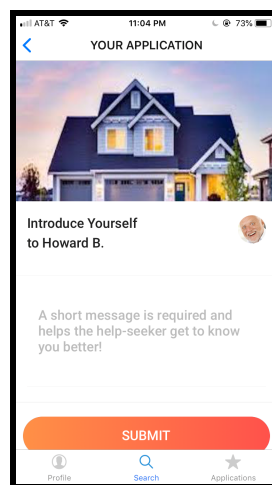
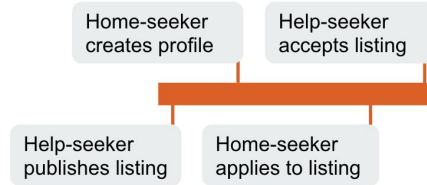
## 2) Design Consistency

Our evaluators helped us realize that we used inconsistent tones throughout our med-fi prototype, for instance our two ways of viewing profiles were radically different. We had tabs and titles misaligned, and we had a mix of formal and semi-formal language throughout the app. We resolved these issues by using a standart package from UI Kittens and made sure all of our profile pages appeared the same way and our buttons had the same colors. We further revised our language for professionalism.



### 3) Taskflow Concerns

We noticed that our Heuristic Evaluators were concerned about the nature of the taskflow. Specifically, they were confused that both the home-seeker and help-seeker went through the same interactions even though their needs were drastically different. To resolve this issue we decided to think of the two sides of the platform as separate apps and focus on the home-seeker side for our hi-fi prototype. In addition our users were confused about the exact content of the application and about the messaging etiquette. They did not know what to write. So we added placeholder texts that would gently guide them.



### Prototype Implementation

In the development of our high fidelity prototype we used React Native to code our features and Expo to easily share and demonstrate it. In addition, we used GitHub to cooperate on our development efforts and UI Kittens for easily customizable templates that added our app consistency.

These tools helped us by providing us:

- A short iteration loop that made for quick development
- Synchronization abilities through Git that made it easy to collaborate
- Easy documentation of installation steps across different platforms
- Control over our screens and varying applications
- Style sheets made possible by UI kittens that made standards and consistency easily enforceable
- Easy ability to develop for iOS and Android simultaneously

These tools came with the following challenges:

- The screen layouts had to be designed by hand or modified from an open-sourced page as our platform did not allow for it
- It was very time consuming to navigate across different platforms that we had not used before
- They required significant environment setup to get started.
- Creating the prototype was nearly not as accessible as lower fidelity ones since they required significant coding and React Native experience.

Wizard of Oz Techniques:

- We did not publish or store any profiles or information on a database and they were not stored after the experience demo
- The sign-up screen allowed users to login no matter what credentials they provided
- The search filter used local data and only worked with the three cities we had inputted

Hard-coded features:

- The locations and help-seeker profiles were hard coded

Missing features and future implementation:

- We did not develop messaging capabilities and instead when a home-seeker clicked “Contact” on a help-seekers page it went automatically to the call screen
- We did not create a high fidelity prototype of the help-seeker experience and that would definitely be our next step.

## **Conclusion**

Overall, we loved working on this project and truly found our work very impactful. In our needfinding exercises we got to hear fascinating stories by awesome people. With our low fidelity prototyping we saw how low barriers there were for design and felt even more encouraged. After receiving feedback on our medium-fi prototype we learned how to incorporate user feedback into the design process and constantly iterate on our designs, solutions and ideas. And at the end of the project we created something that was deeply meaningful and valuable. We hope to carry the skills we honed in on this class to more academic and professional settings, and even to our own lives by realizing that every single aspect of our experience can use a little design thinking within a constant feedback loop.