TutorNOW Final Report

On-demand, student-to-student campus tutoring
Ryan Matsumoto, Josephine Chen, Asli Kimya, Louis Brion

Problem and Solution Overview

While college students currently use resources like office hours, Piazza, and dormitory resident tutors for homework help, these services can often be inadequate: office hours are crowded, resident tutors can be busy, and one-on-one tutoring can be expensive. There is a clear need for an on-demand tutoring service that is reliable, fast, and user-friendly. Because college campuses have thousands of students who take similar classes, our solution leverages mobile technology to connect these students for an on demand peer-to-peer tutoring application.

Tasks & Final Interface Scenarios (1/2 page)

1. **Tutor others in need** (*simple*): Sign up to be a tutor for classes you've taken before, and share your expertise with a fellow students at times that are convenient for you.

We chose this task in order to allow students to participate in a collaborative learning community, where they can not only be students but be the teachers who share their knowledge with fellow students. In addition, tutoring can provide a source of income for many cash-strapped college students and can be especially attractive for that reason.

2. **Get homework help now** (*medium*): Get homework help immediately by requesting a tutor now! Our platform will match users with an available tutor based on proximity, subject area expertise, and ratings. We chose this task because we felt that the need to get immediate help for homework assignments or exam studying was shared by many students, and colleges currently do not have the resources to address this need.

Screen 1

Carrier 📶 2:29 PM

tutorNOW

Tutor

Be Tutored

Appointments

Welcome, Ryan!

Screen 2

Carrier 📶 2:30 PM

Ryan M.

I need help with: BIO 42

When? Now | Later

2

Immunity

420-040

Stanford Hospital and Clinics
Stanford University
Stanford University Parking Structure 7

< Back   Continue >

Screen 3

Carrier 📶 2:30 PM

Ryan M.

Tutoring Offers for BIO 42

No Offers Yet :(

Tap the book to refresh

(Refreshes automatically every 30 sec)

Edit Request

Cancel Request

Screen 4

Carrier 📶 2:30 PM

Ryan M.

Tutoring Offers for BIO 42

Josephine C.   Accept Offer >
Rate: $25/hour   2 minutes
⭐⭐⭐⭐⭐

Barack O.   Accept Offer >
Rate: $52/hour   5 minutes
⭐⭐⭐⭐⭐

Edit Request

Cancel Request

Screen 5

Carrier 📶 2:30 PM

Ryan M.

Josephine C.

School: Stanford
Year: Class of 2015
Degree: BS in Biology
⭐⭐⭐⭐⭐ (24)

Classes: BIO 42, CS 106A, PSYC 135

Skills and Endorsements   **Market Rates**

5 Ruby on Rails   BIO: $25/hr
3 Java   CS: $35/hr
5 C++

Recent Reviews:

"Very knowledgeable about biology. She helped me pass BIO 42!" -James

"Wonderful tutor! Very personable." -Amy

< Back   Accept Offer >

Screen 6

Carrier 📶 2:30 PM

Ryan M.

Barack O.

School: Harvard
Year: Class of 1991
Degree: J.D. (Law Degree)
⭐⭐⭐⭐ (57)

Classes: POLISCI 2, POLISCI 114S, LAW 203

Skills and Endorsements   **Market Rates**

4 JavaScript   POLISCI: $56/hr
4 HTML   LAW: $70/hr
3 CSS

Recent Reviews:

"Quite an impressive young man! I enjoyed his lesson on constitutional law." -Asli

"Great at explaining checks + balances!" -Danny

< Back   Accept Offer >

Screen 7

Carrier 📶 2:30 PM

Ryan M.

Confirmed!

Meet your tutor, Josephine C., at Room 420-040:

The Oval Park
Stanford University

Call Josephine

< Cancel   Start >

Screen 8

Carrier 📶 2:30 PM

Ryan M.

BIO 42 Tutoring

with Josephine

in Progress

00:28

Tutoring Complete

Back

Screen 9

Carrier 📶 2:30 PM

Ryan M.

We hope your session with Josephine C. has been helpful for your studies!

Confirm Payment

$25

Josephine's Rate for BIO42: $25/hour
Duration of Tutoring: 1 hour

Yes! Confirm Payment

Nope! Resume Tutoring

Screen 10

Carrier 📶 2:30 PM

Ryan M.

We hope your session with Josephine C. has been helpful for your studies!

✅

$25

has been charged to your account

Review Josephine

Screen 11

Carrier 📶 2:30 PM

Ryan M.

Josephine C.
School: Stanford
Year: Class of 2015
Major: Biology

Rate your experience:
⭐⭐⭐⭐⭐

Any comments?

Endorse Josephine's skills:

2 Molecular Biology ➕
2 RT-PCR ➕

Submit Review

Screen 12

Carrier 📶 2:30 PM

Ryan M.

Review Completed!

✅

Thank you for using

tutorNOW

Return Home

3. **Find tutoring at a later time** (*complex*): Schedule tutoring appointments for later in the day or the week. Our platform allows users to easily find reliable tutors at future times based on their availability. We chose this task based on the heuristic evaluations, where the evaluators brought up the issue that many college students have tight schedules and deadlines, which means they may want to schedule appointments in the future. Our application matches tutors and tutees for future appointments based on their joint availabilities from the Google Calendar API. Tutors and tutees can additionally view upcoming appointments.
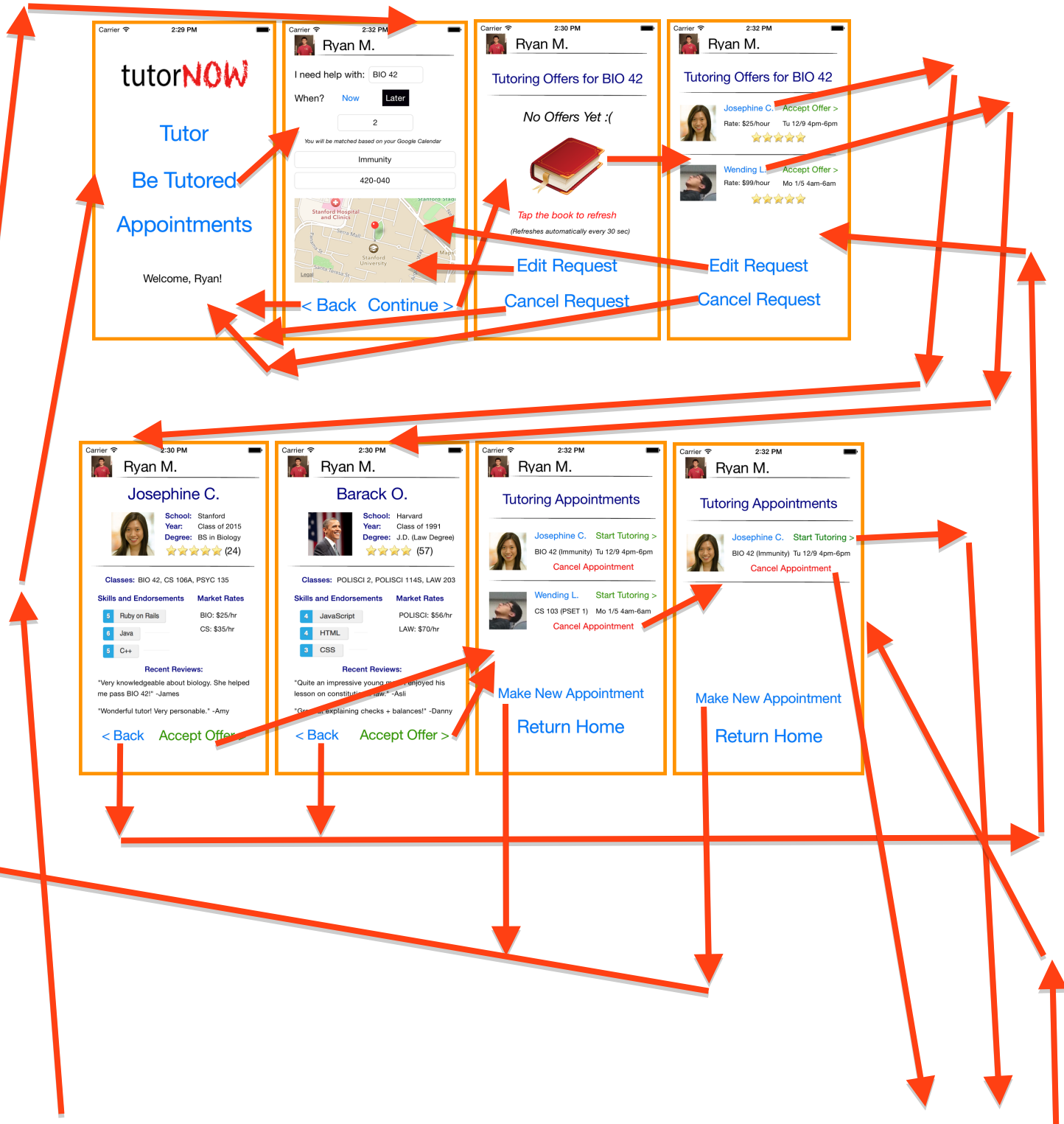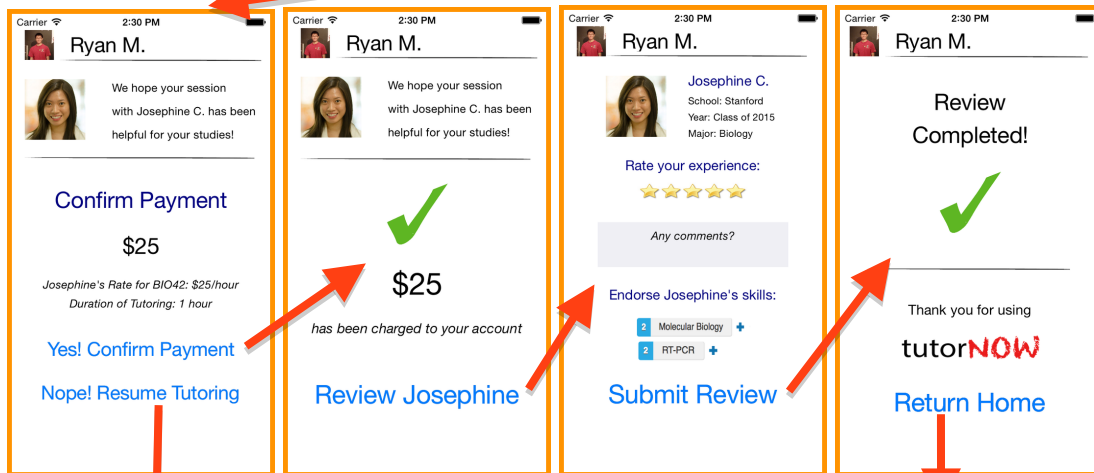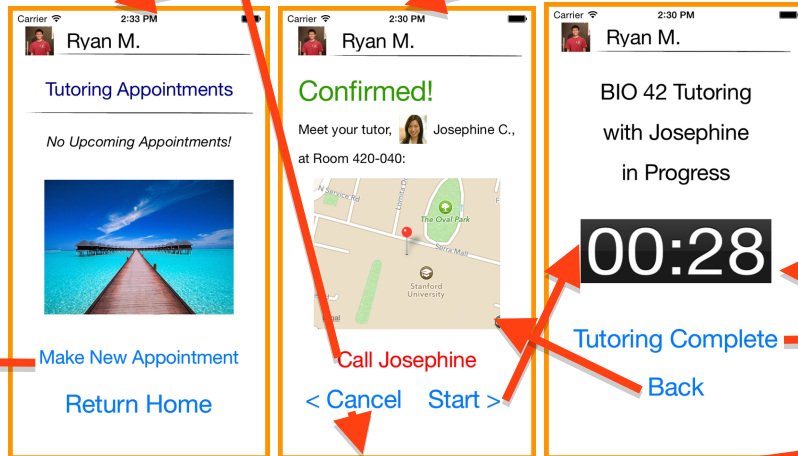
Major Usability Problems Addressed (2-3 pages)

After we received the Heuristic Evaluation results, we grouped the severity Level 3 and Level 4 Heuristic violations into three categories: The violations related to the (1) One directional flow, (2) Prioritization of information, and (3) Real need for additional services.

Violations Related to Previously Having a One Directional Flow

This category covers the heuristic violations that occurred because we previously had a very one-directional flow. The main fixes we had to make involved adding functionality that allowed for greater user flexibility, such as "Cancel" and "Back" buttons, to enable canceling operations, revising input, and quickly recovering from mistakes.

1) [H2-3. User control and freedom] [Severity 3] [Found by: D, C, B, E] On the screen where the user views a list of tutoring offers, the main button is to "Cancel Request." However, this takes the user all the way back to the home screen. In case the user just wants to go back one screen and edit something about the request, there is no way to

do that; they will need to fill out the entire request again. There should be an option to just go back one step and edit the request.

Fix: We added an "Edit Request" button, on the Tutoring Offers screen to enable the user to go back to the previous screen. We kept the "Cancel Request" button that goes back to the main page and adjusted the flow accordingly, so that the user has the most freedom by being able to edit and cancel or proceed with one of the offers.
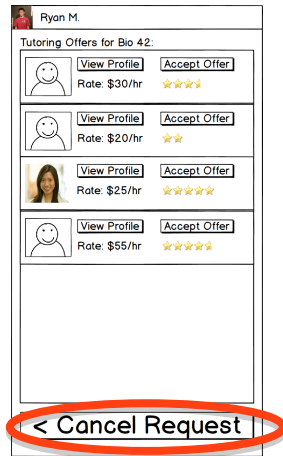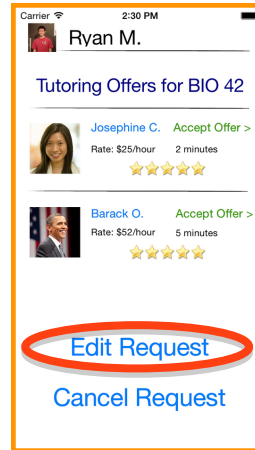

Left: Before   Right: After

2) [H2-1. Visibility of system status] [Severity 3] [Found by: C, D, E, B, A] Once I accept the offer, I am taken to a confirmation page for the tutoring job. However, if I click "Cancel" from this page, I am taken all the way back to the home screen. There is no confirmation that my tutoring job has been canceled, and I am not sure what the status of the tutoring job is, or where to check that. One suggestion would be to add a confirmation screen that indicates the tutoring job has been canceled, and perhaps a suggestion for what to do next.

Fix: In the getting help now flow, when the user clicks "Cancel" from the confirmation page, he/she is taken back to the tutor offers page, which indicates that the previous job was canceled and he/she can select a new job. In the flow of getting tutoring help later, the "Tutoring Appointments" page will indicate which appointments are still valid and give the users the ability to cancel appointments.
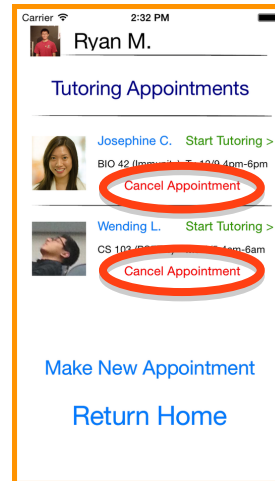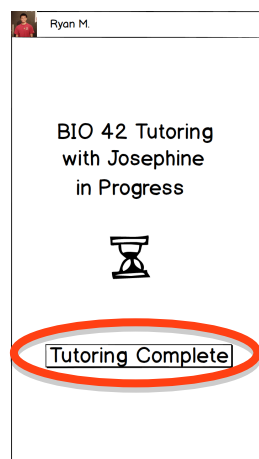
Left: Before   Right: After

3) [H2-3. User control and freedom] [Severity 4] [Found by: C, E, D, B]
   If the user accidentally clicks the "Start" button when they didn't mean to, there should
   be a quick "Back" button. Otherwise, they will be charged for the tutoring session when
   they didn't mean to. Currently, the only option on this page is "Tutoring Complete,"
   which does not give the user much control.

   Fix: We added a "Back" button on "Tutoring in Progress" screen that goes back to
   confirmation screen.
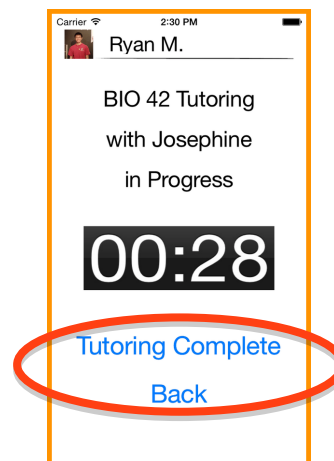
Left: Before   Right: After

Violations Related to Prioritization of Information

We also realized that our application prioritized certain information in ways that were not very intuitive or helpful for users. For instance, adding payment information is not at the same level of importance as tutoring. In addition, among the information listed on tutor profiles, ratings are paramount.

1) [H2-5. Error prevention] [Severity 3] [Found by: C, D, B]
   What happens if a user fills out a tutoring request, completes the session, but doesn't have a Venmo account set up? Do they set this up at that point within the flow? Or are they prevented from filling out any tutoring requests until they link their Venmo account from the Payments flow? One suggestion would be to disable the "Be Tutored" button until Payments has been filled out.

   Fix: We realized that the payments flow was slightly orthogonal to our three tasks, so we removed the specifics of that flow from our designs. Through the Wizard of Oz technique, we assume that each tutee is offered $20 in free tutoring credit, and a page prompting them to input credit card information will come up once they've exceeded the free credit amount.

   ***** No pictures because we are using a "Wizard of Oz" technique as a fix*****

2) [H2-8 Aesthetic and minimalist design] [Severity 3] [Found by: E]
   The "star" rating can be a misleading measure of tutor's quality. While Josephine could be a rockstar biology tutor and is highly rated, she could also be a poor psychology tutor at the same time. However, this fact is obscured by her high rating from tutoring biology. This would render the combined star rating unhelpful. Consider adding a breakdown of ratings by class/subject, as well as adding the number of raters in a bracket following the stars. Many tutee would want to see this information first in order to identify high quality tutor effectively, so it should be moved to be visible without further scrolling as well.

   Fix: We moved the star ratings to the top of the profile so that they are the first things that a tutee will see and added the number of raters next to it. However, we believe that the skills endorsement section provides more subject/expertise-specific information than class-specific ratings would provide.
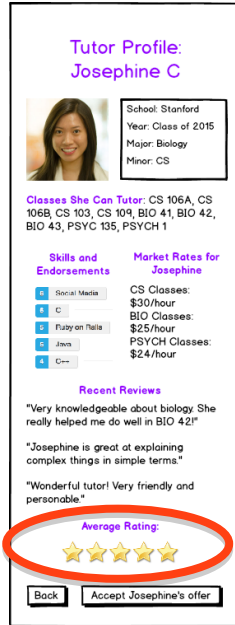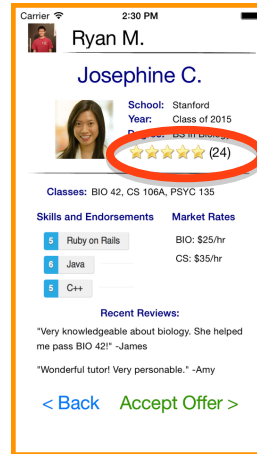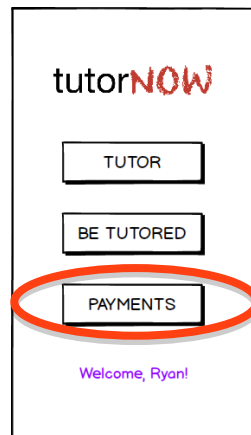
Left: Before   Right: After

3) [H2-8 Aesthetic and minimalist design] [Severity 3] [Found by: E]
In the home screen, it is unclear why the "Payment" button is on it. Most people would set up the payment (Venmo link) only once. As a result, making the payment button as large and visible as the other two key functions (Tutor and Be tutored) doesn't make much sense. The payment setting should be hidden in a setting function somewhere.

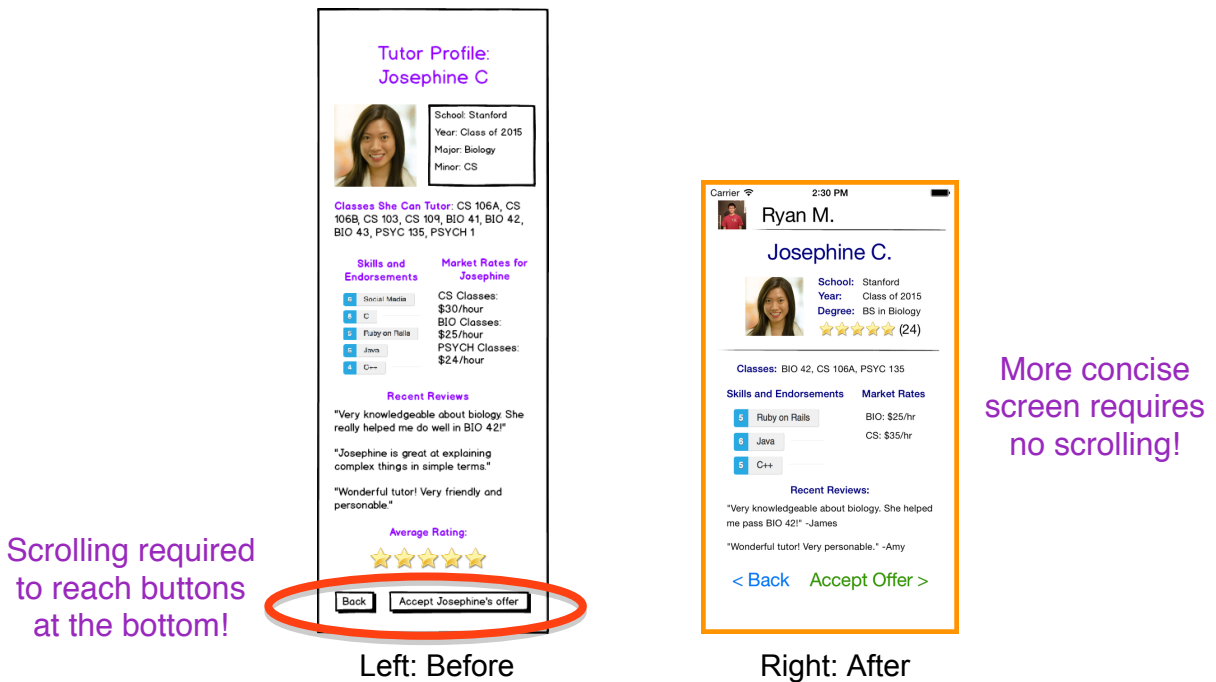Fix: Removed Payments from the home screen and revised our three tasks.
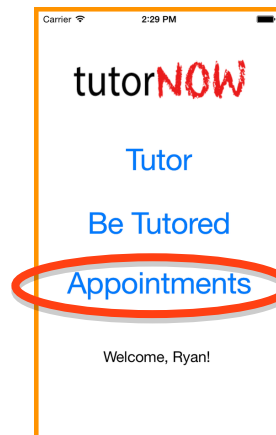
Left: Before   Right: After

4) [H2-7 Flexibility and efficiency of use] [Severity 3] [Found by: B, A]
On the tutor profile page, the user needs to scroll down through all the information in order to reach the "back" and "accept offer" buttons, which might be inefficient, particularly when a tutor has multiple reviews. Have the same buttons on the top of the screen, or have a sticky tab on the bottom with those buttons.

Fix: Added the same buttons to the top of screen in revised med-fi prototype. In hi-fi prototype, we removed some irrelevant information from the profile page, which allowed for the entire page to fit in one screen, so no scrolling is required.

Scrolling required to reach buttons at the bottom!

More concise screen requires no scrolling!

Left: Before   Right: After

5) [H2-8 Aesthetic and minimalist design] [Severity 3] [Found by: A]
The chunk of text on the payment screen is way too long for anyone to read, especially because the buttons are already quite self-explanatory. Put this information in a separate screen that's only seen if someone clicks on a "Help" type of button.

Fix: Using Wizard of Oz technique to assume payment system is already in place.

*** No pictures because we are using a "Wizard of Oz" technique as a fix***

Violations Related to Need for Additional Services
Based on the heuristic evaluation feedback, we also realized the need to include several services that we hadn't realized were necessary.

1) [H2-3. User control and freedom] [Severity 3] [Found by: C, D, B]
   When the user clicks "Tutoring Complete," the next screen shows that $25 has automatically been charged to his/her Venmo account. In situations where I am paying money, I would feel more comfortable if I had a chance to review first. Even though I am going to pay anyway, it gives a sense of control if I actively say yes, rather than having money automatically charged to my account (e.g. reviewing a bill in a restaurant before putting down your credit card, order confirmation pages for online shopping). Some applications such as Uber automatically charge to your account, but I would argue that the situation of taking a taxi is much more time constrained because you are usually hopping in and out in a rush. One suggestion would be to add a quick confirmation screen that says something like "Tutoring session completed! Confirm payment?" This would also prevent errors where the user accidentally presses a button and ends the tutoring session early.

   Fix: We added a payment confirmation page after the "Tutoring in progress" page with two buttons: 1. resume tutoring 2. confirm payment. This allows the user to either go back and resume tutoring if the button was pressed in error or to confirm payment if the session has concluded. The payment page was also changed to demonstrate that the payment had gone through properly.


Left: Before   Right: After
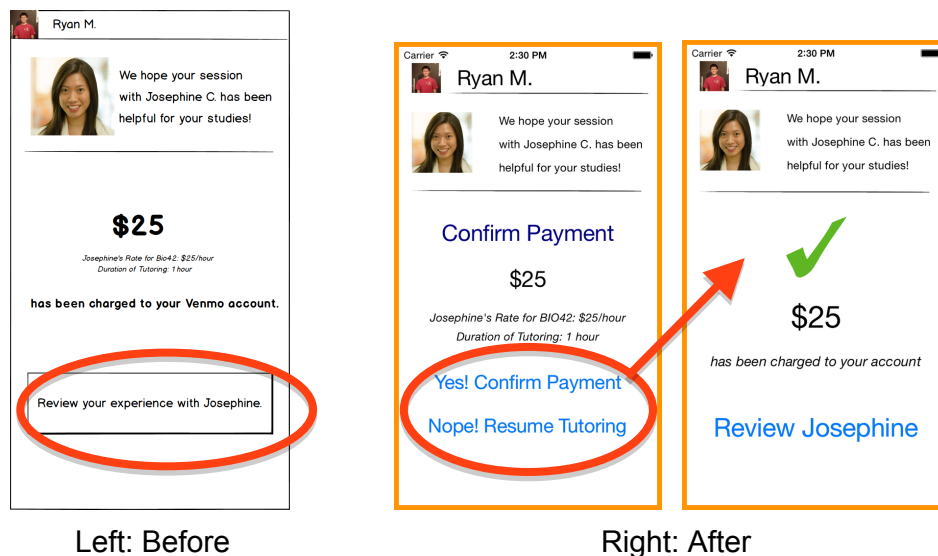
2) [H2-5 Error prevention] [Severity 4] [Found by: E, B]
    The "confirmed" screen, after accepting an offer from the tutor, lacks an option to contact the tutor. The need to contact the tutor may arise in many cases, e.g. a tutee is kicked out of a room in Old Union and now need to change her location

    Fix: We added the ability for the tutor to call the tutee and vice versa on the "Confirmation Screen."
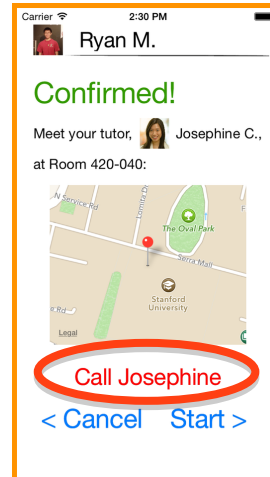
Left: Before   Right: After

3) [H2-5 Error prevention] [Severity 4] [Found by: E]
    The "tutoring in progress" screen doesn't have a pause button. This could cause several problems because the charge is directly calculated based on the length of the tutoring. If the start button is tapped by mistake or if the tutor needs to take a break (emergency phone call, etc.), the charge could be inaccurate, causing disagreements between the users. A pause button should be added to prevent the aforementioned issues.

    No fix: If the the start button is tapped by mistake, there is an option to go back and re-start the session. However, in real life normal tutoring situations, tutoring fees are also based on length and there's no ability to pause tutoring even if tutoring does not take place for the entirety of the time. Think about the last tutoring session you had--was the tutoring time adjusted because the tutor went to the bathroom?

    *** No pictures because we are not changing anything***

4) [H2-7 Flexibility and efficiency of use] [Severity 3] [Found by: B]
    The interface allows the user to choose whether he would like to be tutored "now" or

between a range of time. Even though there is a clear goal of immediacy in this task, many college students have tight schedules and deadlines, which means they are able to predict when the will need and have time for tutoring. Consequently, users might want to be able to schedule tutoring ahead, to make sure they will be helped. However, the interface only supports same-day tutoring. Add a "set date and time" to the second screen.

Fix: Created a new task and flow for scheduling tutoring help for later.

See the tasks/scenarios section for the NEW tutoring later flow

5) [H2-4 Consistency and standards] [Severity 4] [Found by: B]
On both the tutoring details and tutoring confirmation page, we see a map, related to the location where the tutoring session would happen. However, there are no pins on the map specifying the exact location, which makes the map not very useful. Add pin to location specified.

Fix: Added pin to map.
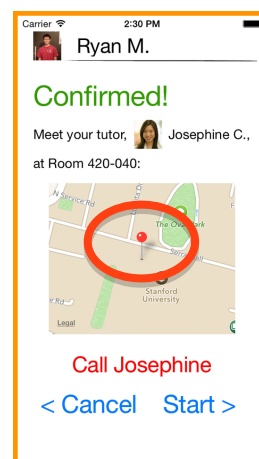
Left: Before   Right: After

6) [H2-5 Error prevention] [Severity 3] [Found by: A]
The user could easily pinpoint the wrong location by tapping the map because exact locations are difficult to find in such a small map. Add a search option for locations so the user can easily find them without looking through the map.

No fix: The pin is an approximation of the location; we have included a text field to specific the exact location description (building and room number). In addition, because our service is meant to be college campus-specific, all the tutors/tutees are on the same college campus, which means that the area is quite constrained.

***** No pictures because we are not changing anything*****

7) [H2-7 Flexibility and efficiency of use] [Severity 3] [Found by: A]
As far as I understand this app, it seems to only show tutors that match the time and location exactly. Chances are, tutors will not always be available in that manner. If there aren't enough tutors that meet the exact criteria, also include similar times and nearby locations.

No fix: This is already what is done; our search includes all tutors from the surrounding area who have relevant expertise (just ranked by proximity and relevance of skills/expertise).

*** No pictures because we are not changing anything***

Other changes that we made:

1) As we presented the high-fidelity prototype in class, there seemed to be questions about how the tutor offers would come in, so we included a page with no tutor offers and prompt the user of the prototype to tap a refresh button to see what offers could come in.
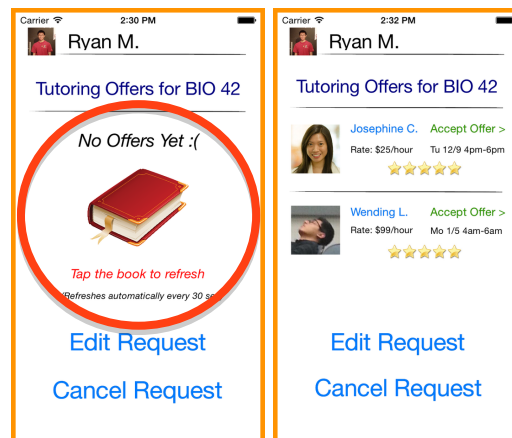

Making the tutor offering system more clear by showing a screen with no offers.

Design Evolution

The **low fidelity prototypes** are our initial designs after doing task inquiry and textual analysis. The **medium fidelity prototypes** are revisions of our low fidelity prototypes after getting feedback on user testing. The **high fidelity prototypes** are revisions of our medium

fidelity prototypes taking into account the heuristic evaluation that was performed on our medium fidelity prototype.

Start screen evolution:

Low Fidelity   Medium Fidelity   High Fidelity
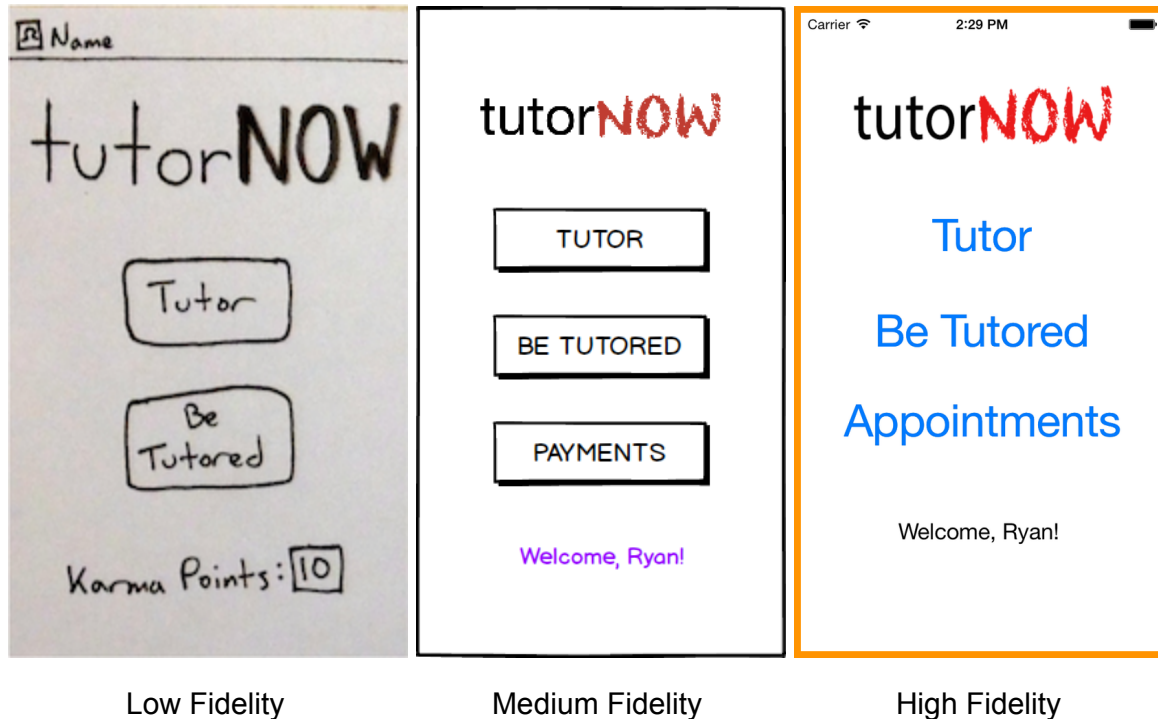
Low Fidelity:
Tutor / tutee matching start screen- involves two separate screen progressions depending on whether you wanted to be a tutor or tutee at the time. This is the initial screen in that progression after creating a profile.
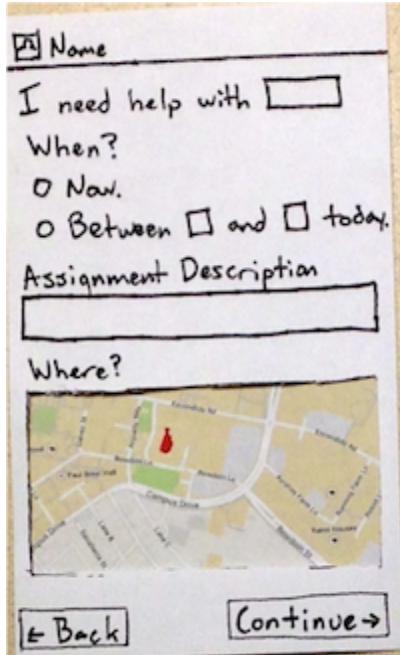
Medium Fidelity:
No initial profile required anymore- One of our application testers noted that setting up her profile took a very long time. We decided to no longer make it mandatory to set up a tutor/tutee joint profile before requesting help. This will allow first-time customers to more easily access our application without getting frustrated by a lengthy sign up process. We also removed the karma points system and instead began using
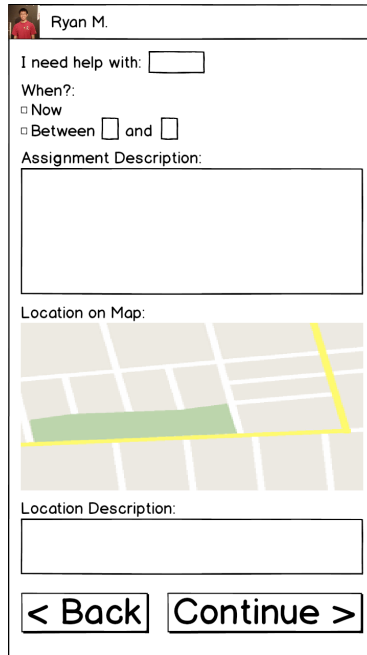
High Fidelity:
Assumed payment and profile authentication- through third party APIs such as Venmo, Paypal, and Facebook, using this applications became much easier. These design choices are reflected by the new initial screen. We also wanted to make it clear that future tutoring sessions are supported by our application as well.
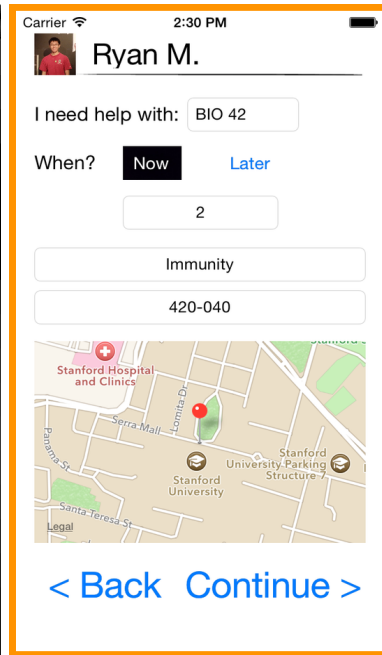
Tutoring request screen evolution:

| Low Fidelity | Medium Fidelity | High Fidelity |

Low Fidelity:

Initial tutor request screen- This screen allows the user to input more specific information about exactly what they want to learn at the time, including relevant class, assignment, timing, and location information through the use of text fields, checkboxes, and maps.
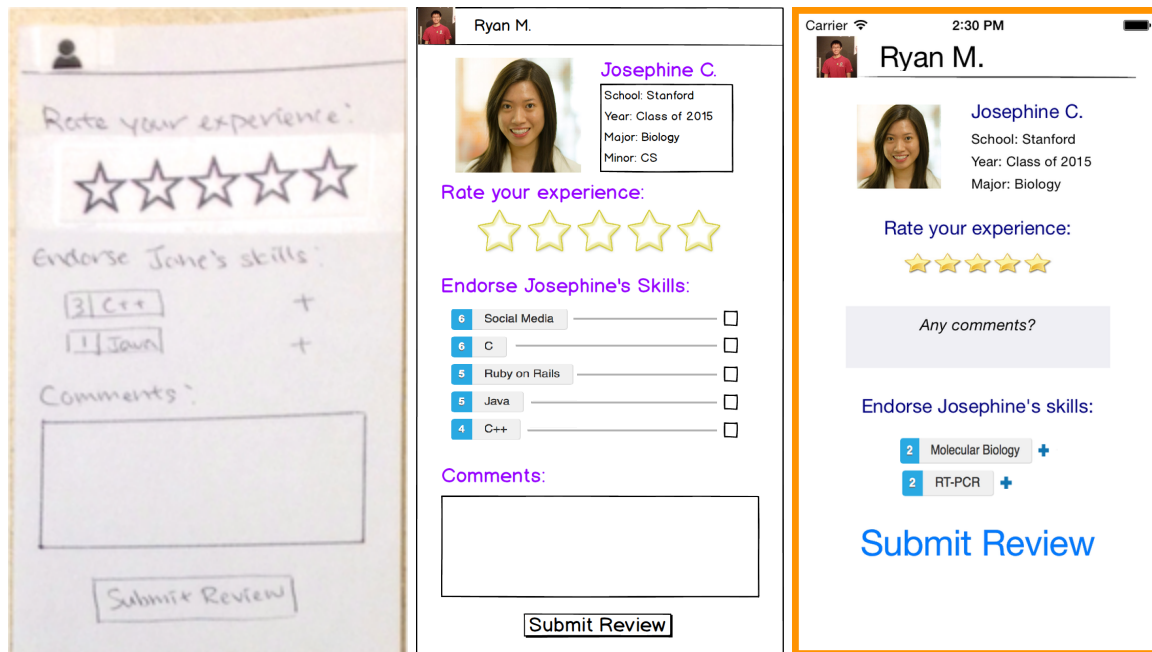
Medium Fidelity:

Simplified tutee-decided location- Two our application testers were confused about who decided the location where the tutoring took place. In our new design, we make it clear that tutees are the ones who choose the location, and tutors cannot "negotiate" these locations. Thus, students are matched with tutors more efficiently where the student wants to meet.

High Fidelity:

Streamlined design- We removed some of the awkward white space after the heuristic evaluation as well as moved the assignment description and room location to be together so like elements could be grouped.

Review screen evolution:

| Low Fidelity | Medium Fidelity | High Fidelity |

Low Fidelity:
Reviewing system- this has two similar but separate flows that allow tutors to review their tutees (not shown here) and for tutees to review their tutors through a star system, an endorsement system, and text boxes for comments.

Medium Fidelity:
Reviews and ratings for the tutor, not the tutee- We observed from our user testing that it wasn't really necessary for tutors to rate their tutees. This just takes up our customers' valuable time and is not really important to accomplishing our major tasks. We also added more relevant information to the review screen about the tutor.

High Fidelity:
Streamlined review screen- based off of our heuristic evaluation and trying to make it as easy as possible to get through each part of our application, we made the comments section smaller and before the endorsements. We also cleaned up how endorsements looked so they don't unnecessarily fill up the whole screen width.

Prototype Implementation

Our high-fidelity prototype was developed in iOS with Xcode 6.1, optimized for iPhone 5 and iPhone 5S. In Xcode, we dragged and dropped view controllers onto the storyboard to add new screens. Then we would add appropriate labels, buttons, images, and text fields using

drag-and-drop features before linking the screens together using modal cross-dissolve transitions.

How the tools helped:
- Xcode's storyboard helped us in rapidly developing major screens without having to actually code that much. Most of our screens only required "drag-and-drop" items.
- Xcode's transitions worked very well for our app, and were easy to implement.
- Xcode's "MapView" class allowed us to add well-positioned maps to appropriate screens with a little bit of coding.

How the tools did not help:
- Xcode 6.1 has a new positioning interface that requires you to use "constraints" to position items so that they fit across multiple devices (4S, 5, 5S, 6, 6 Plus). To avoid the complexity of this new positioning interface, we decided to optimize our application for the iPhone 5 and 5S so we could focus on design rather than sizing.
- Xcode's views do not automatically have scrolling as a feature. Thus, if you want to create a screen that is bigger than the iPhone 5/5S, you must implement "ScrollView," which is also very time-consuming and easy to mess up. Thus, we decided to vertically shrink our tutor profile screens, which were previously longer than the iPhone 5.
- Objective-C is a difficult programming language, so we were unable to implement much of the backend necessary for a fully functioning application.

Wizard of Oz
- We assume that a payments system is already implemented, so that users can add credit card information to get paid for lessons and pay tutors.
- From the tutor perspective, the "Waiting for tutee to accept offer" and "Waiting for tutee to confirm payment" screens instruct the user to tap an image to advance to the next page. We did this because we did not implement a backend that would handle interaction between tutors and tutees in real time.

Hard-Coded Data
- Whenever a tutee accepts an offer, he/she is automatically re-directed to a tutoring session with Josephine C. for BIO 42. We hard-coded a single flow for this tutoring session because we did not implement a backend with multiple tutors.
- The comprehensive tutor profiles were hard-coded to showcase the different features of a profile (classes tutored, endorsements, ratings, market rates, reviews).
- Several tutoring appointments are hard-coded into the tutor-later flow to give prototype users multiple options.
- Tutor matches are not dynamically generated. Tutees initially see a screen with no tutoring offers, but once they tap the book icon to "refresh" the page, they will see two hard-coded tutoring offers.

What is missing & What might you add in the future?

- To schedule tutoring appointments later, we assume that tutors and tutees will log on with their Google Accounts. This will allow us to use the Google Calendar API to match tutors and tutees for future appointments based on their schedules. This gives more flexibility in case a tutee makes a request but her calendar changes before a tutor offers help. Thus, the scheduling automatically keeps up with students' busy lives. Of course, all of this will happen in the actual application, but is not implemented yet in the prototype.
- We also want to add a flow for tutors to set up and edit their profiles.
- We also need to implement a legitimate backend database to manage tutor-tutee interactions in real-time. We can do this once we learn more iOS programming.