



# NightOwl

Get help now.

## Hi-Fi Report CS147

**Griffin Dietz**  
Developer

**Lachlan Green**  
Product Manager  
Documentation

**Zara Saraon**  
Design  
User Testing

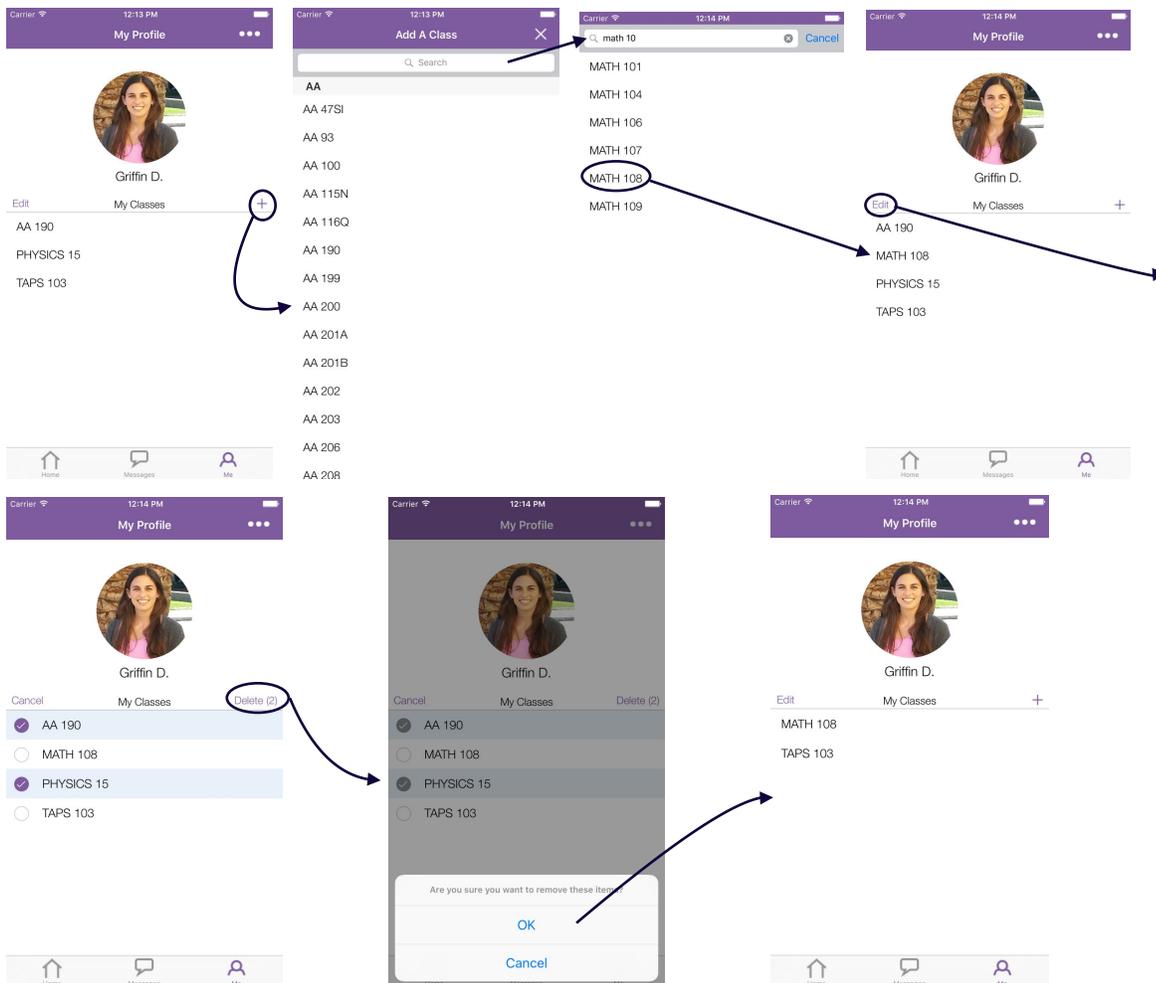
# Problem and Solution Overview

From our contextual inquiry and needfinding, we found that students working late at night often struggle to find help on assignments and problem sets. NightOwl solves this problem by connecting students with peers who are awake, nearby, and working on the same thing. NightOwl is a mobile application that enables a student to make himself available for any class that he is enrolled in, and then will present the user with a list of peers who are currently doing work for that class – ordered by proximity. Students can then message these peers through NightOwl, meet up with them, and collaborate on work.

## Tasks and Final Interface Scenarios

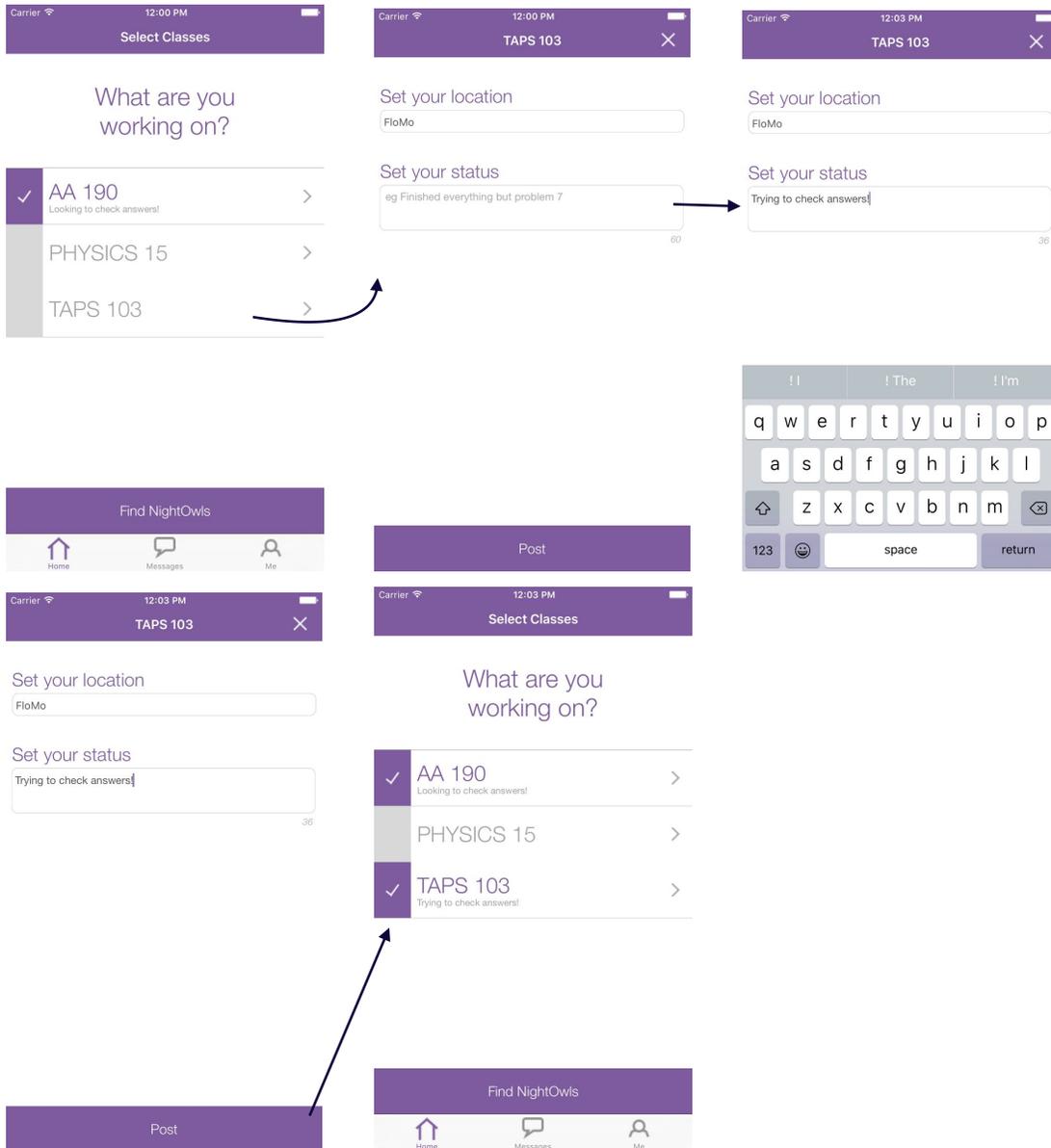
### Simple Task: Update list of current classes

We chose this task because in order to use NightOwl over many academic quarters, users need to be able to change the list of classes they are currently enrolled in.



## Medium Task: Become available for a class & write a status

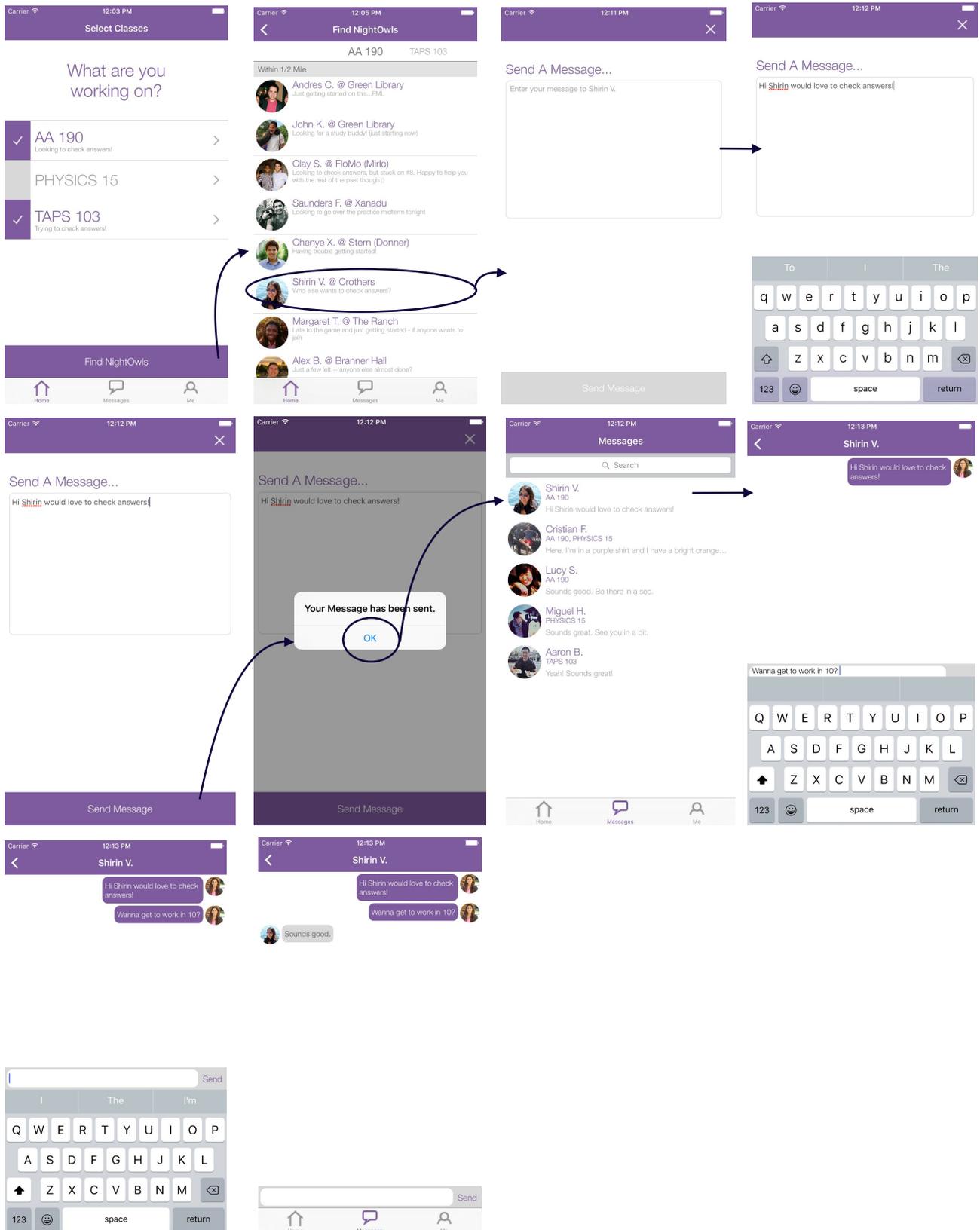
We chose this task because it is the first step in being able to find a peer to work with: before you can view fellow peers, you must indicate which class you are working on and write a status indicating your progress on the assignment thus far.



## Complex Task: Reach out to classmates

This task involves seeing a list of classmates near you (as well as their locations and statuses) for all of the classes you are available for, and being able to message them. We chose this task as our complex task because it is the meat of our app: being able to message classmates who are also awake, working on the same thing as you, and looking to collaborate is how NightOwl users

are ultimately able to get immediate help with work. This task also extends across both the Home tab and messages tab in the app, as conversations begin in the former but continue in the latter.



# Design Evolution

## Step 1: Needfinding

We began by interviewing a diverse sample of people who exercised extreme focus. Each had interesting things to say about how they related to focus, but our most significant insights came from looking at where the answers of each overlapped.

 <p>focus feels like being totally relaxed, totally loose</p> <p>phone is distracting separate work and non-work spaces</p> <p>practice makes perfect</p> <p>refocused by remembering larger goals</p> <p>you should be aware of your thoughts like they are clouds drifting in the sky</p> <p>distracted by iMessage messages you know you're focused when you don't care about anything else</p> <p>sometimes forgot what she was working on</p> <p>Instant results aren't realistic intense focus without breaks is unsustainable</p> <p>focus is all about rhythm</p> <p>knew she was focused when she didn't care about anything except</p>	 <p>how their body is feeling</p> <p>how quickly they are breathing their environment</p> <p>what things are distracting them at the moment</p> <p>what they want to achieve what is happening around them</p> <p>think about the need to focus</p> <p>thinking too far ahead goals</p> <p>what thoughts are top of mind</p>
 <p>perpetual glances at cell phone</p> <p>refocus with deepen breath constantly aware of inner thoughts</p> <p>set goals to focus on and work towards listen to calm (not pump-up) music before match</p> <p>a set routine to refocus constantly switching between tasks</p> <p>create a daily routine opened another tab rather than navigating in current window</p> <p>likes reading before working to get focused always trying to find his rhythm</p> <p>has someone hide cell phone before working</p> <p>play music to drown out distractions</p>	 <p>attached to work when focused less reactive</p> <p>worried about not being efficient enough guilt for procrastinating</p> <p>fear of missing out more calm</p> <p>anxiety about not producing enough strong and healthy</p> <p>the mind and soul are in it irritated by distractions</p> <p>stressed out before important events</p>

Empathy Map

Our biggest insights and their appropriate needs from this stage were the following:

1. environment affects focus → people need to be in the right environment to focus well
2. focus is (usually) a means to an end → people need to remember that end
3. focus is analogous to finding your rhythm → people need to find personal rhythm
4. people focus in different ways → people may need different techniques
5. physical comfort is important for good focus → people need to be comfortable physically
6. long-term focus requires breaks → people may need to be reminded to take breaks

## Step 2: POVs, HMWs and Experience Prototypes

Next, we interviewed more people that fit within our more focused audience of college STEM students, identified some point of views (POVs) for our potential users, formulated some “How Might We” (HMW) statements, and created several experience prototypes to get feedback on some of our initial ideas for solutions.

After interviewing three STEM students, we generated many POVs and chose our 3 favorite.

We then generated around 15 HMW statements for each POV, and selected our favorite one among them. Here are the HMWs we generated for one the POVs, with our favorite HMW bolded:

**We met...**  
Muhammad, a Junior studying CS at University of Texas who interned at Facebook last summer as a software engineer.

**We were amazed to realize...**  
he spent 30-45 min every other day trying to answer questions.

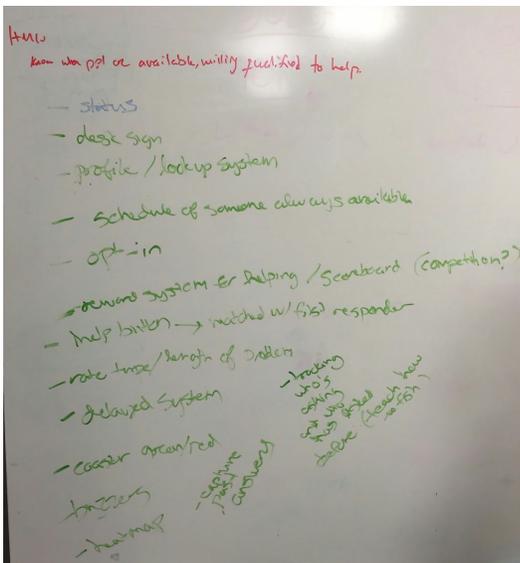
**It would be game-changing to....**  
help him find answers to his questions for efficiently.

**How might we...**

- create a process to more efficiently find the answer
- get better documentation
- connect him with others asking the same question
- allow him to ask questions anonymously
- help him find the right person to ask
- eliminate the need to ask questions
- make asking question to boss more like asking questions to friend
- make asking questions less embarrassing
- make asking questions fun
- anticipate questions
- provide unintimidating
- know when people are available/willing/qualified to help**
- give him something to do during while he's searching for an answer

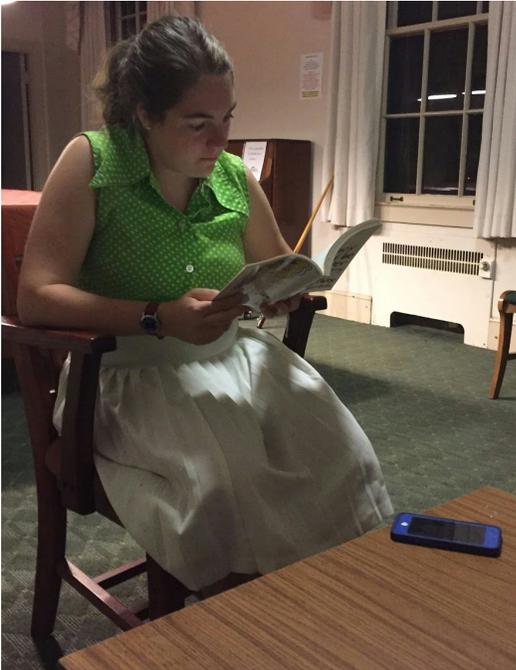
HMWs generated for one of our subjects

We then brainstormed around 15 solutions for the three HMWs we selected to respond to our three POVs and selected our favorite three solutions among all of those.

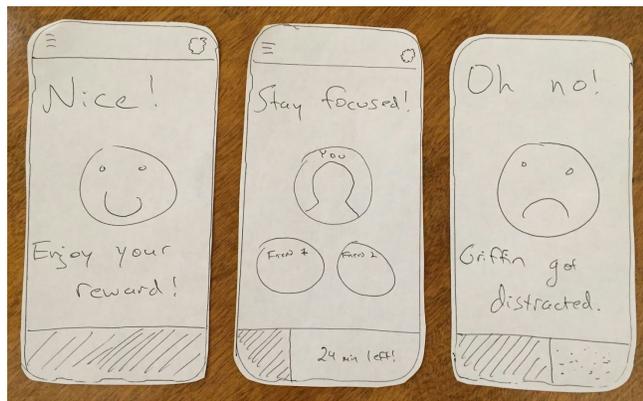


Brainstorm of solutions to one HMW

After selecting our favorite solutions among all those we generated for all HMWs, we created three experience prototypes that we could show users to test out the core experience for each solution. In the first, we tested how people reacted when we buzzed each time they touched their phone when they were supposed to be working. In the second, we tested the concept of leveraging social pressure to enhance focus by penalizing a group of friends if one lost focus. Finally, in the third, we tested how much people liked being able to find 1st, 2nd and 3rd degree connections in the classes they were taking. During this last experience prototype, we realized that students wanted to be able to find someone to work with, say, the night before a pset was due more than just find friends in general.



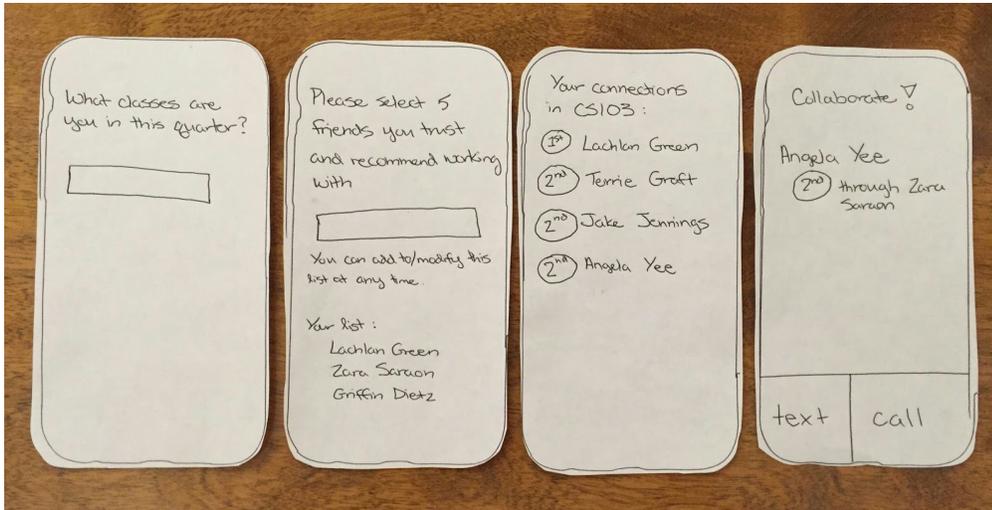
Experience Prototype 1 "Buzz"



Experience Prototype 2 "Social Pressure"



Experience Prototype 2 "Social Pressure"

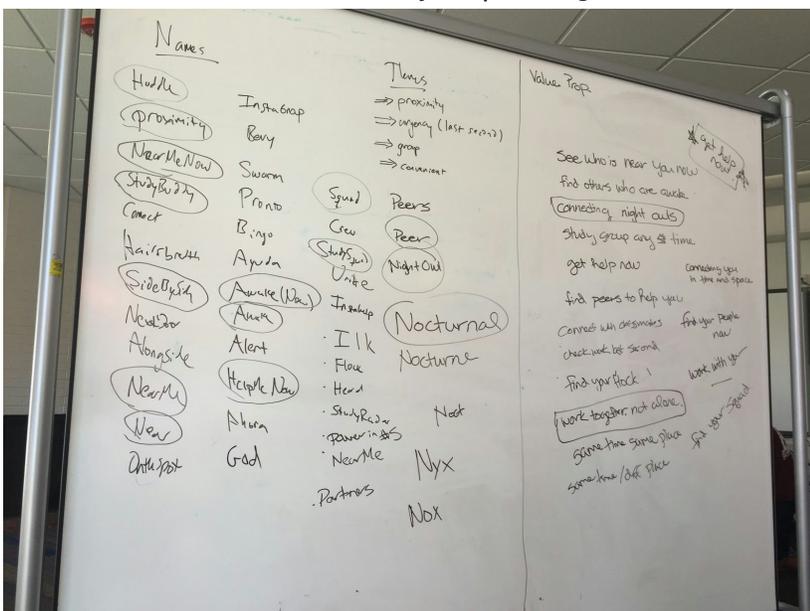


Experience Prototype 3 "Connecting Students"

### Step 3: Concept Video

Next, we defined our problem and solution, came up with a name and value proposition, came up with 3 tasks that our app could enable, storyboarded scenes for a video depicting those tasks, and then filmed and edited a video illustrating the tasks.

To pick a name and value prop, we generated a whiteboard full of options and then discussed each of our favorites. After some time, we decided on NightOwl for the name because we liked that it is tangible, has interesting connotations, and sounds catchy. For a value proposition at the time, we chose "Connect with classmates. Collaborate on work" because we felt like it captured the core value of our app, though we would later change that to "Get help now" because we felt it was catchier and more directly responding to a need of our audience.



Brainstorm of names and value propositions

Weighing brevity and completeness, we chose problem and solution statements that were as concise as possible.

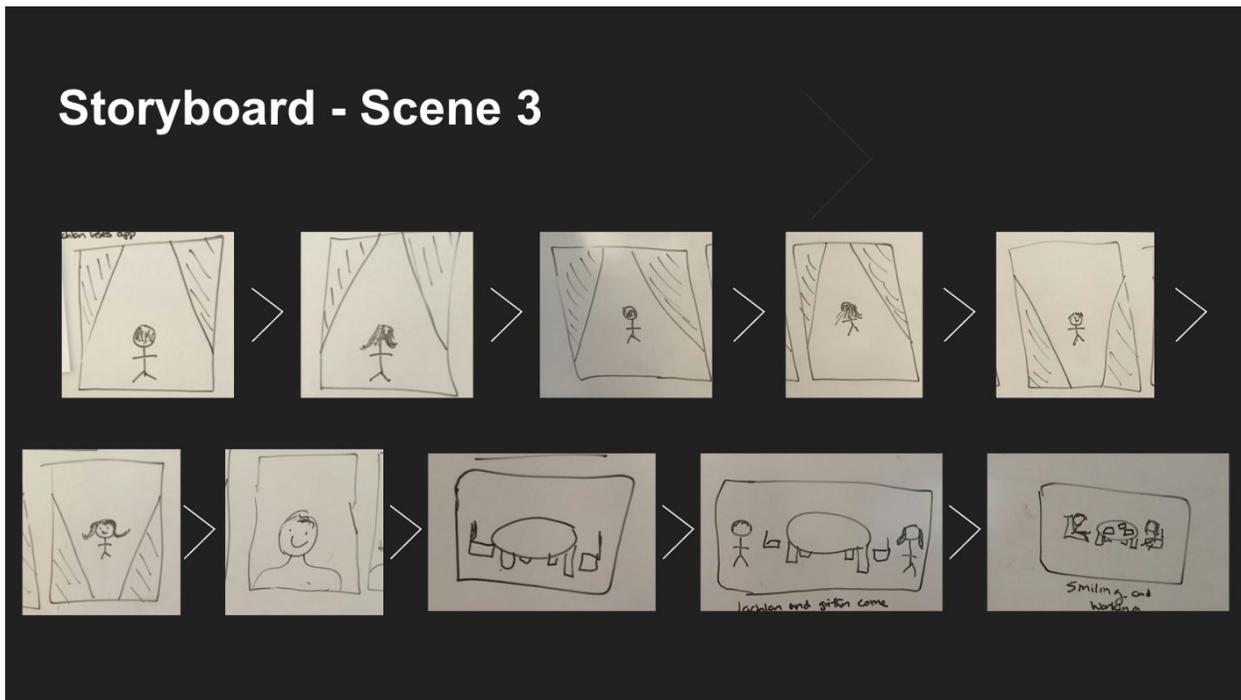
Our problem: "Students working at night often struggle to find help on assignments."

Our solution: "NightOwl connects students with peers who are awake, nearby, and working on the same thing."

Next, we set out to choose three core essential tasks, one simple, one medium and one complex, that our app enabled. After further discussion, we settled on the following:

1. Find people in your class who are working now
2. Respond to someone reaching out to you
3. Make yourself available and write a status

Then we storyboarded scenes that showcased the completion of those tasks and filmed a short concept video.



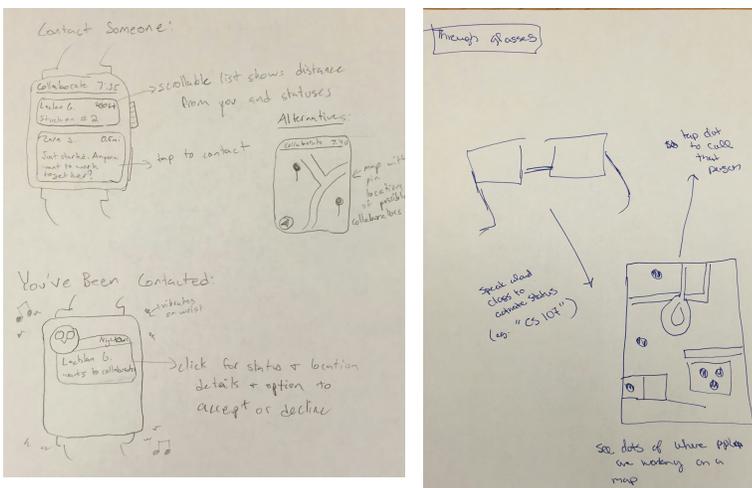
Storyboard of Scene 3



Shot from scene 3

#### Step 4: Low-Fi Prototypes

Next, we generated a lot of concept sketches, selected our favorite two to flesh out further, chose our favorite among those two finalists, and then developed and tested that final choice.

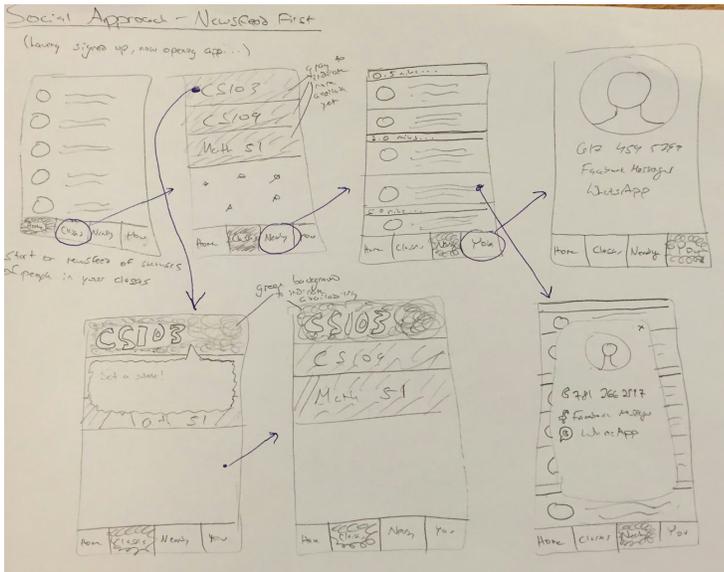


Concept sketches

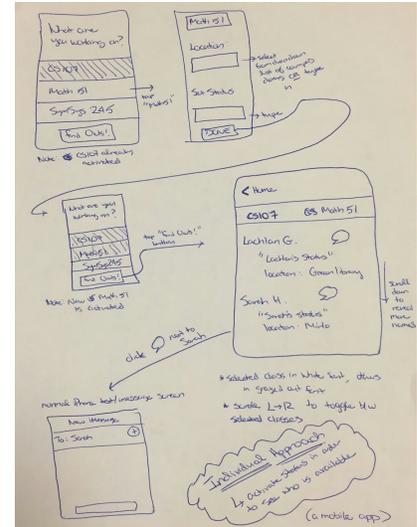
We ultimately decided to go with a mobile app because we felt that without a clear functional advantage to using a wearable or a desktop app, the ubiquity of phones dictated that they

should be our first choice. Because we had experience with iOS but not with Android, we chose to develop for the iPhone.

Our two finalists, after we fleshed them out, were most significantly different in what the user saw first and how much freedom the user was given to navigate throughout the app. Between these two designs (pictured below), we chose the “Individual First” approach that pushed the user to fill out what classes they were taking before being able to see who else was available so that we could service more relevant and useful content.



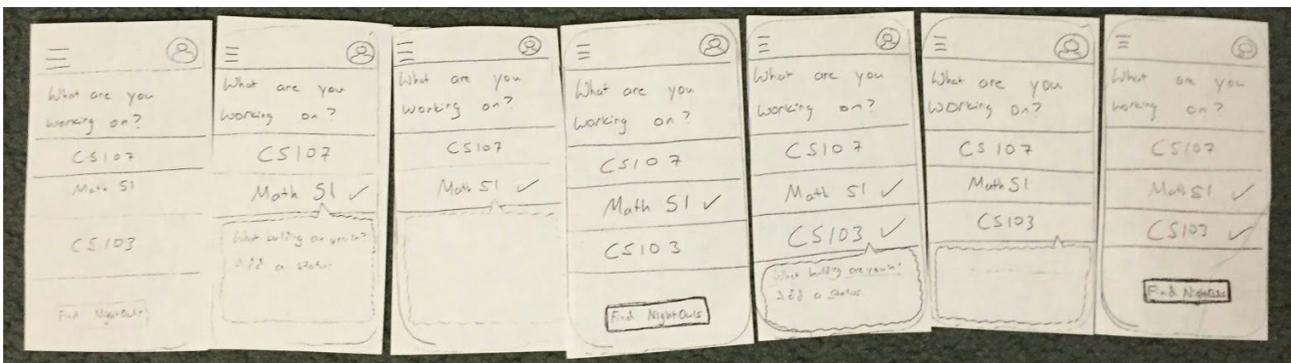
Individual First



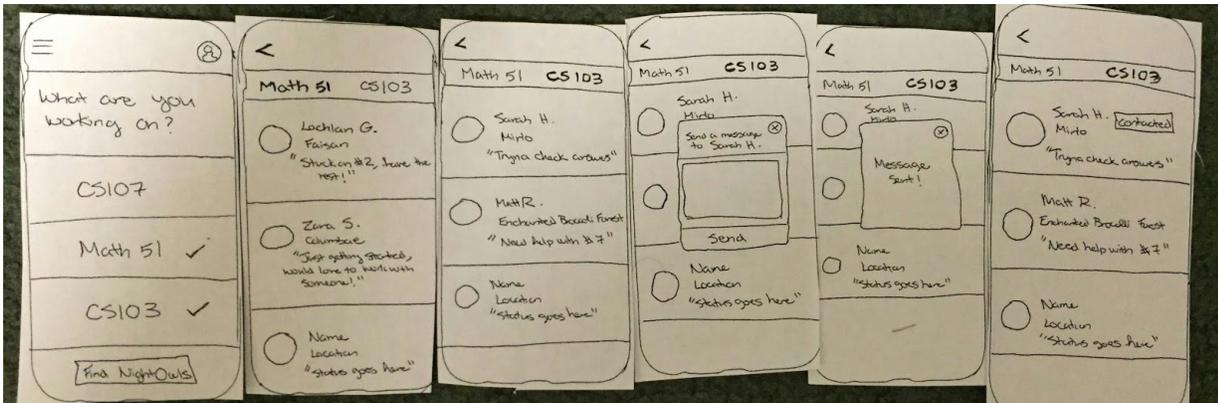
Social Approach

We then mocked up the low fi paper prototypes you see below based on the following three tasks:

- Simple Task: Respond to someone reaching out to you
- Moderate Task: Make yourself available and write a status
- Complex Task: Find people in your class who are working now and connect with them



Complex Task: Find people in your class who are working now and connect with them)



*Moderate Task: Make yourself available and write a status*



*Simple Task: Respond to someone reaching out to you*

Our most significant learnings were that people a) struggled with posting a status, particularly since there was no clear “Post” button, b) needed to be able to view their conversation threads, and c) wanted to know more about other people using the service before reaching out to them. We would directly address each of these in our next iteration.

### Step 5: Medium-Fi Prototype and Heuristic Evaluation

For our next iteration, we used Sketch and Marvel to create a Medium-Fi prototype that responded to the feedback we received during from our Low-Fi prototype. We made posting a status more clear with a large Post button that changed color when the appropriate information had been filled in, we added a Messaging tab where users could view their conversations with other users, and we added informative example text in the status and location fields that disappeared when the user began typing to push the user to add informative statuses.

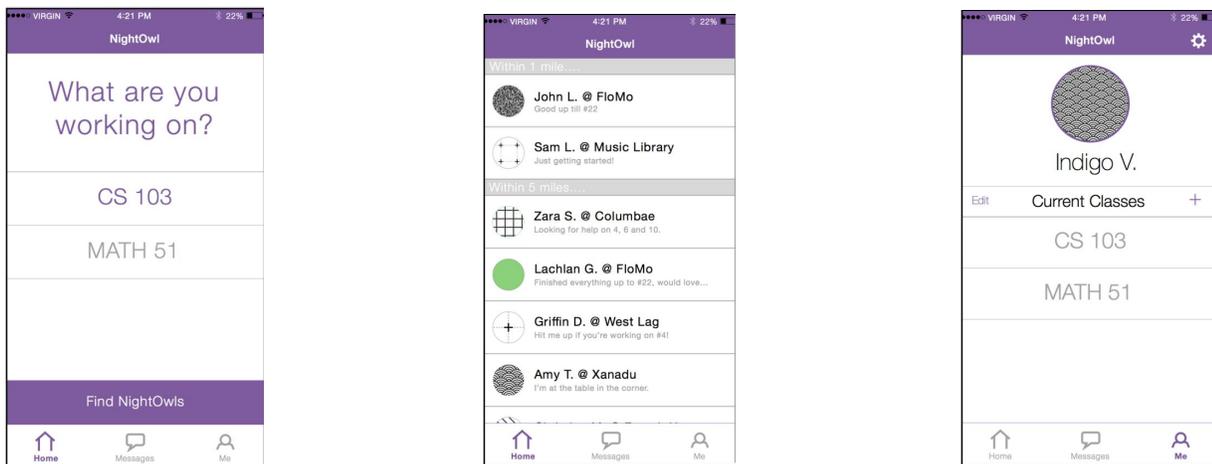
We also updated our value proposition to “Get help now” based on feedback we received from our TA and our peers, who felt “Get help now” was more immediate and compelling than “Connect with classmates. Collaborate on work.”

We also used Sketch to create higher fidelity UIs and Marvel to imitate the UX. These tools allowed us to create prototypes that felt significantly more real.

Some of the major design decisions we made during this process were the colors, the tab bar organization and the icons.

After playing around with a lot of different shades of blue and purple, we settled on **this color** (#805BA0). We decided to avoid blue for reasons discussed in lecture and because we felt it was overused, and chose this shade of purple mainly out of aesthetic preference. We chose to use just one non-grayscale color to ensure that the UI was clean and simple (we also used two shades of gray and black for text and borders.) Deciding what sections to include on the tab bar required the most amount of debate. For a while we weren't sure whether or not to separate out the action of making oneself available from the list of available students awake and available. Though separating them might make more sense from a high level, we felt that functionally, it made more sense to combine them since it's important that the students we show a user are relevant to the class that user is trying to work on at that time. For the icons, we wanted to maintain a minimalist design that still clearly conveyed each section, so sought to create icons in that style.

Here is the result from that step.



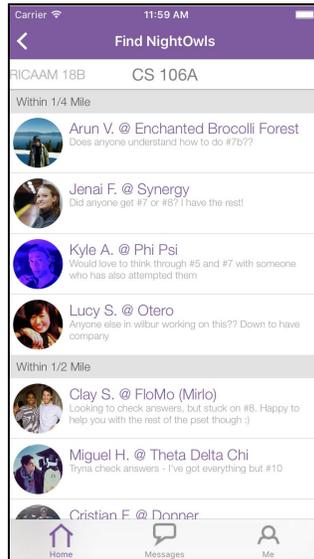
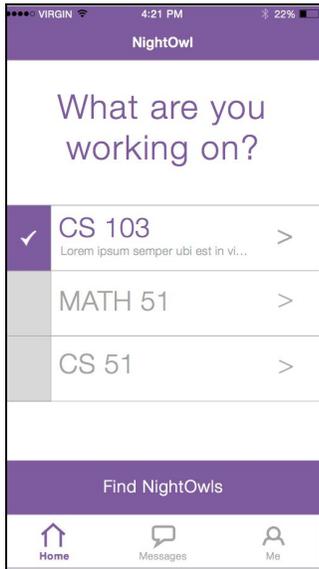
*Medium-fi prototype screens*

Once we had created the screens and connected them on Marvel, we handed off our prototype to a group of our classmates to conduct a Heuristic Evaluation. We would end up implementing many of the suggested changes in our Hi-Fi prototype. We discuss these changes in detail in the next section.

### Step 6: Hi-Fi Prototype

Based on the feedback we received from the Heuristic Evaluation conducted on our medium-fi prototype, we developed our hi-fi prototype. We used Sketch to update our UI and to create icons, and used XCode, Objective C and nib files to build the app.

Here are some screenshots of some of the new screens we made in response to the feedback we got on the medium-fi prototype.



Hi-fi iOS screens

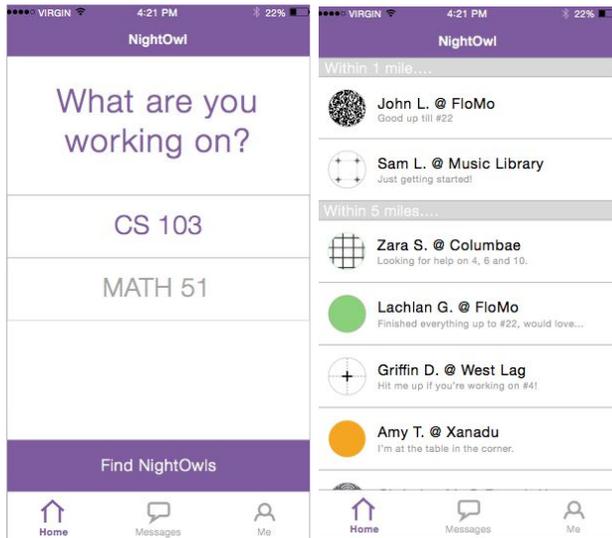
# Major Usability Problems Addressed

Our heuristic evaluation identified nineteen problems with our medium-fi prototype; nine of these were of severity 3 or 4. Listed below are each of the severity 3 and 4 problems that were identified, along with either a rationale for the design changes we made to fix that problem or an explanation for why that problem did not lead to a design change.

## 1. [H2-3 User Control & Freedom] [Severity: 3]

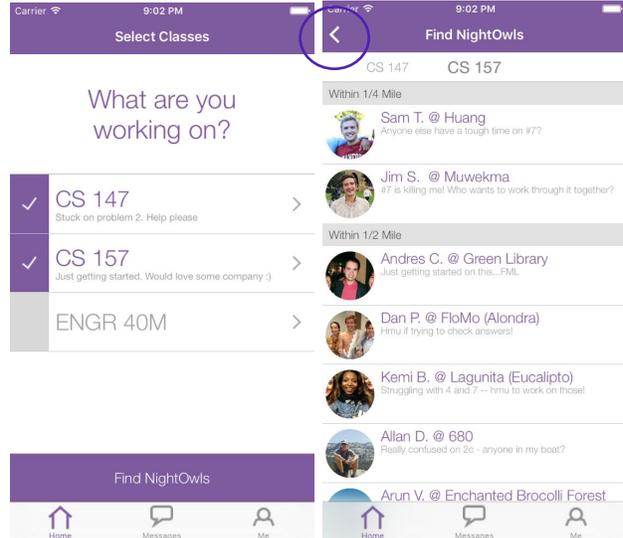
After clicking “Find NightOwls” in the home tab, there is no way to go back to the “select classes” screen. To fix this, we have added a back button on the newsfeed screen to enable users to easily navigate back to the “select classes” screen. We chose to add the back button because it was a simple and elegant way to give the user the freedom to navigate back to the “select classes” screen from the newsfeed, without changing much UI.

Old Design:



New Design:

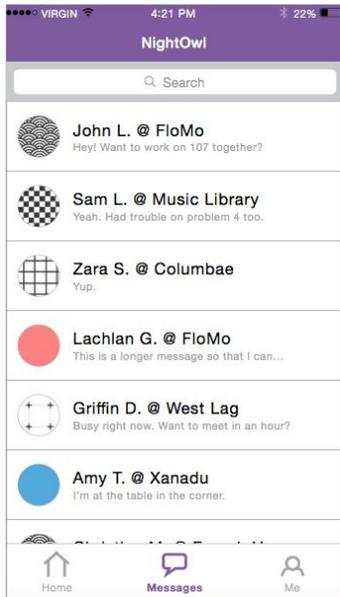
Back arrow added



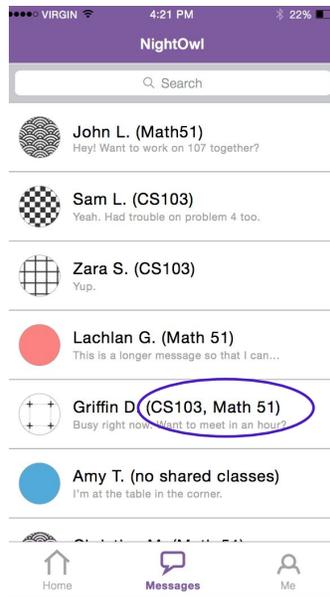
## 2. [H2-6 Recognition & Recall] [Severity: 4]

When viewing your message threads in the messages tab, there is no way to tell which classes you share with any given individual. To fix this, we revised our design to show a list of all shared classes next to a person’s name in the messaging tab. We chose to revise our design in this way because we felt it was important remind users of exactly which classes they share with the peers they are about to message, without forcing them to navigate to any other screen to get this info.

### Old Design:



### New Design:

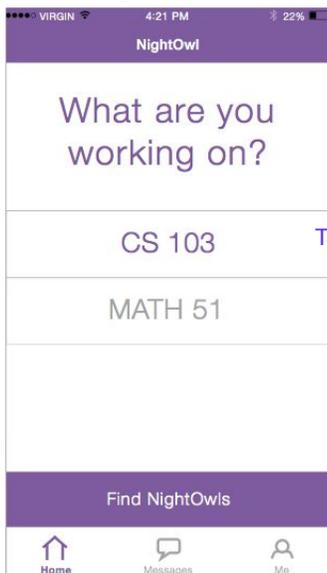


All shared classes listed next to each name

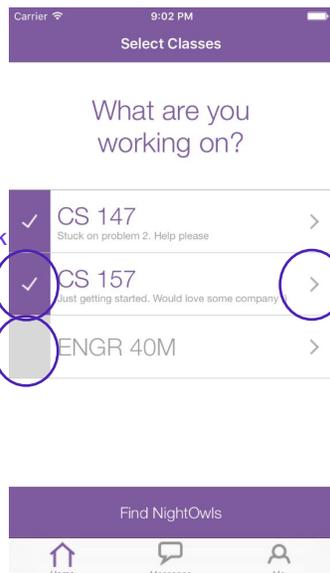
### 3. [H2-5 Error Prevention] [Severity: 3]

In our old design, the Heuristic Evaluators found it unclear how to unselect a class when you are done working on it. In our revised design, we have added a check mark next to selected classes that you can uncheck to deselect that class. You can also unselect a class by hitting the arrow that appears on the right side of a selected class (which takes you to a screen where you can update your status for the class or deselect the class). We chose to do this because it provides power users with an easy way to deselect classes (tapping the check on the main screen) while also making the option clear to novice users (who will likely navigate to the “>” to understand what options they have in regards to that particular class). The purple highlight makes it clear whether any given class is selected or not at any given moment.

### Old Design:



### New Design:



Tap check mark to deselect

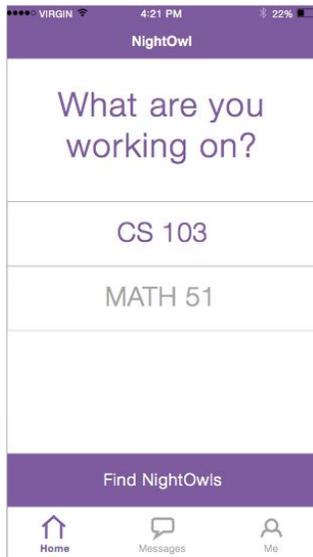
Tap gray box to select

Tap arrow to update status or deselect class

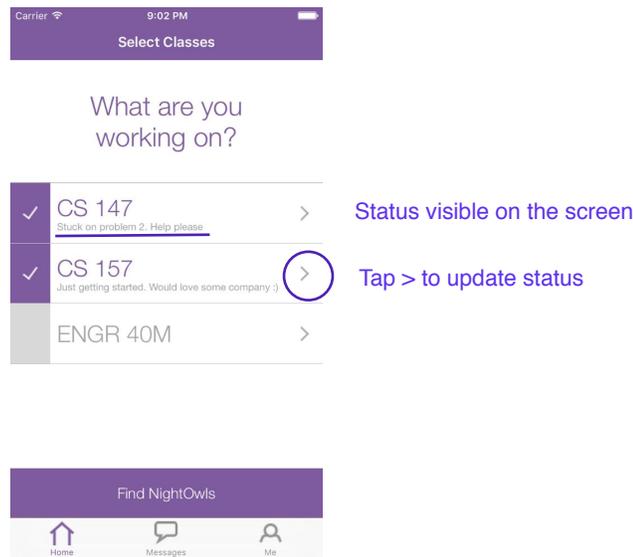
#### 4. [H2-3 User Control & Freedom] [Severity: 4]

In the old design, it was unclear how a user could view their current status for a class and update that status. In our revised design we have made the current status for a class visible on the home tab, and added a “more” arrow which will allow a user to update their status at any time. These design changes enable users to more easily view and modify statuses.

##### Old Design:



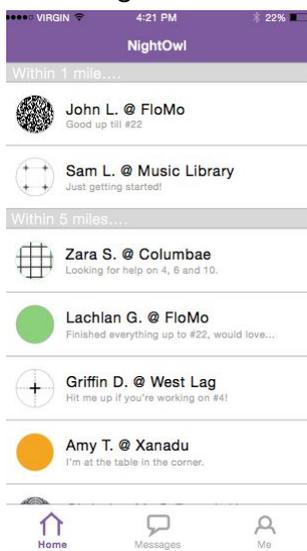
##### New Design:



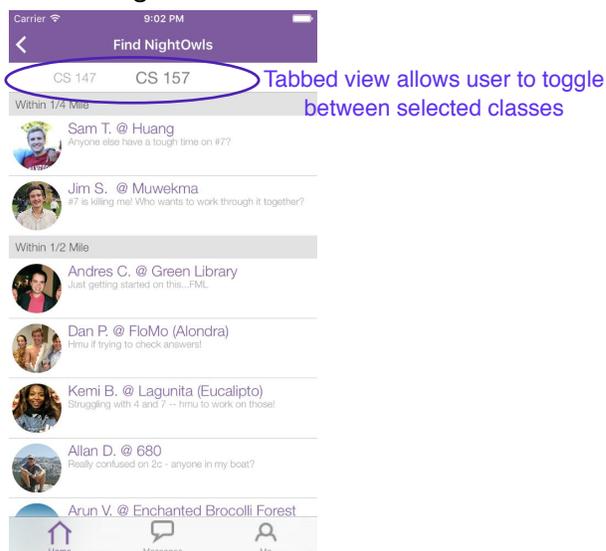
#### 5. [H2-1 Visibility of System Status] [Severity: 4]

In our old design, if a user is working on two classes at once (e.g. simultaneously working on 2 psets), he or she has no way to know which NightOwls in the newsfeed are for which class. To fix this, we revised our design to have a tabbed view (one tab per selected class) so that users can toggle between separate newsfeeds for each selected class.

##### Old Design:



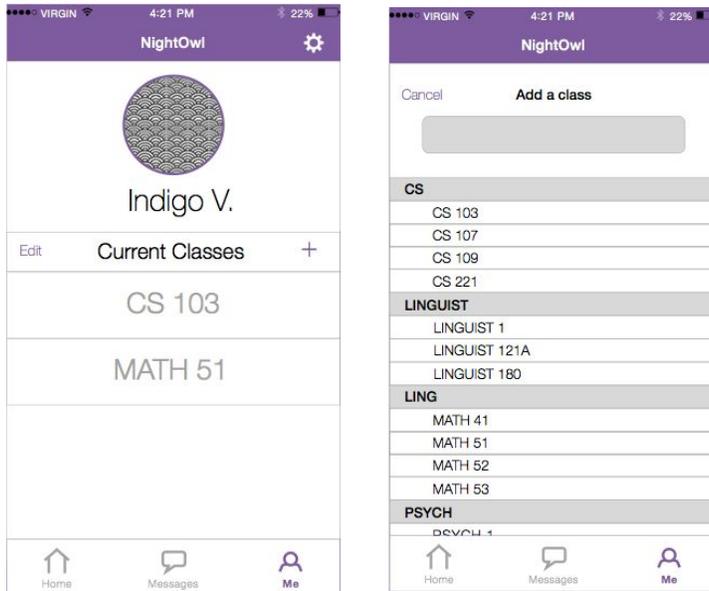
##### New Design:



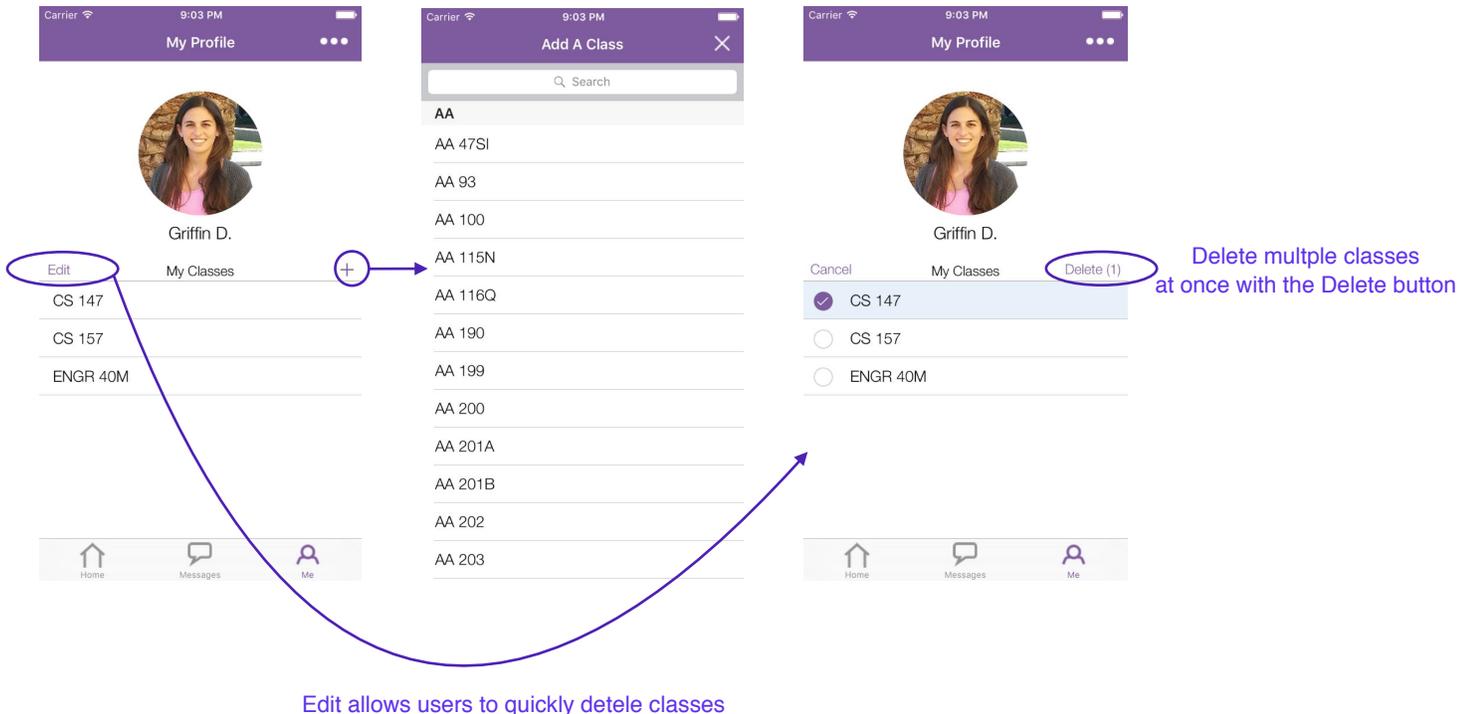
## 6. [H2-5 Error Prevention] [Severity: 3]

Our Heuristic Evaluators found that we did not prevent users from adding a new class by accident. This violation has not led to a design change; instead, we have made deleting classes such a quick process. If a user does make a mistake, they are only one tap away from rectifying that error and deleting the accidentally added class. This solution has the added benefit of keeping the “add a class” flow as simple as possible.

### Old Design:



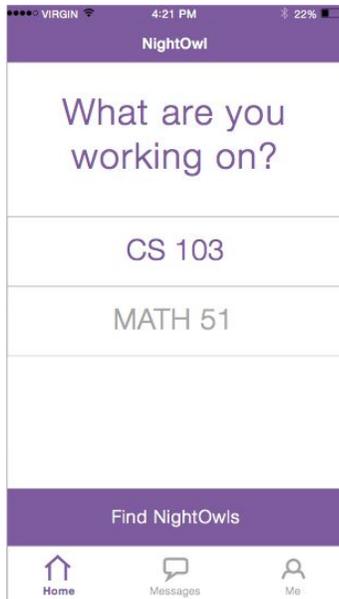
### New Design:



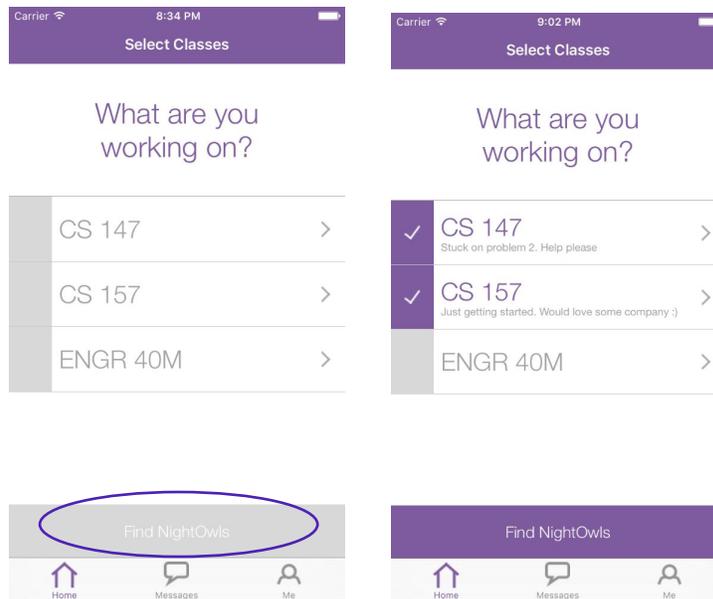
7. [Consistency and Standards] [Severity: 3]

The “Find NightOwls” button in the original design seemed confusing to one Heuristic Evaluator - who wasn’t sure how that differed from the home screen. We made the “Find NightOwls” button grayed out and unselectable when there are no classes selected - so that it is clear to users that they can only find NightOwls once they are active in a class. We feel that after a single use of the app, it will become very apparent that the initial home screen is where you select the classes you are working on, while the “Find NightOwls” call to action button leads you to the newsfeed (which is very different from the initial home screen).

Old Design:



New Design:



Button grayed out when no classes selected

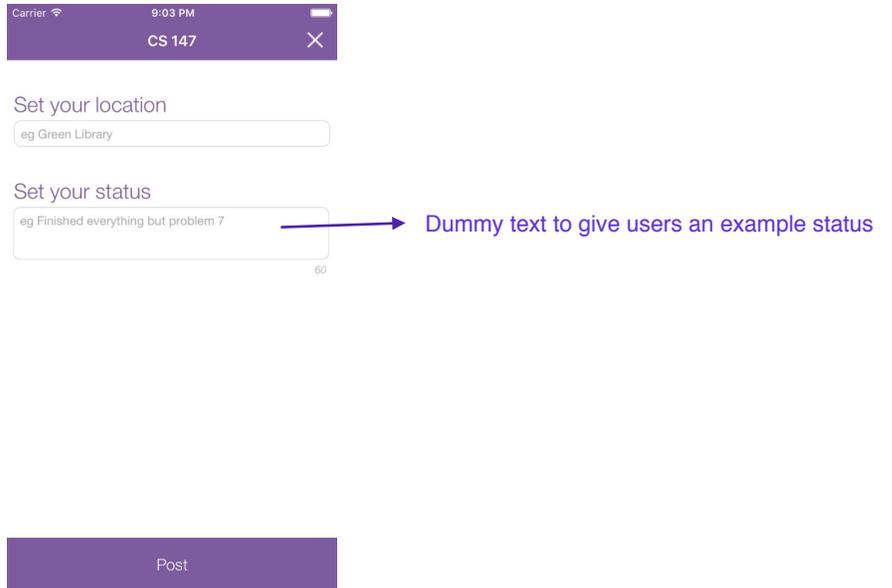
8. [H2-6 Recognition and Recall] [Severity: 3]

Our Heuristic Evaluation uncovered the fact that users were confused by the “set location” and “set status” boxes - finding them vague and unclear. In our design, we have continued to use example text to indicate to users what information they should include in the location and status boxes. We have chosen not to provide further instructions to users regarding statuses because there are many possibilities for what types of things a user could include in a status, and we don’t want to limit users in any way. Additionally, we believe that over time, the NightOwl community will create norms for what type of information should be included in a status.

### Old Design:



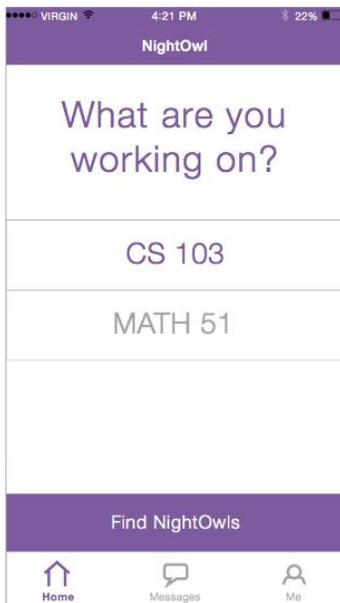
### New Design:



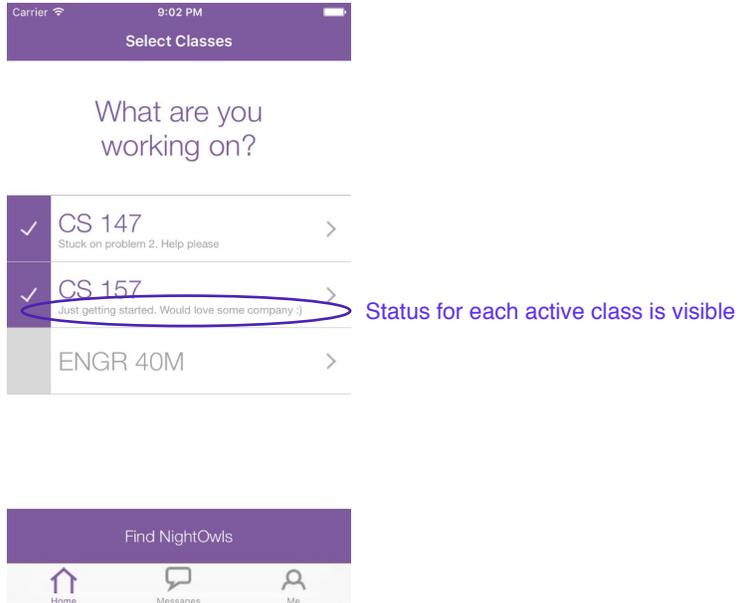
### 9. [Visibility of System Status] [Severity: 3]

In our old design, there was no way to view current status. In our revised design, we have solved this issue by making the current status for each class visible on the home screen.

### Old Design:



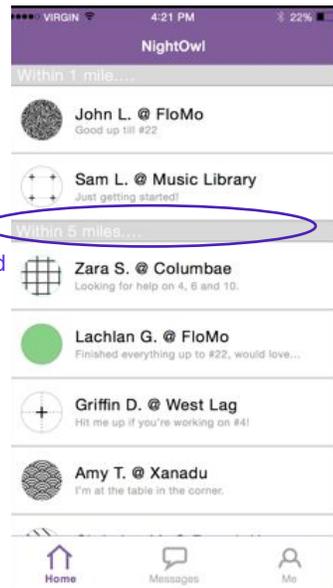
### New Design:



We also made a few additional design changes, explained below:

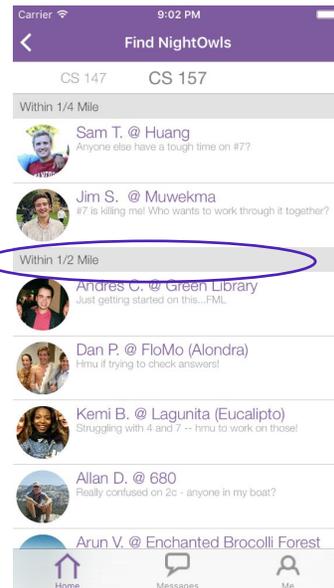
- In our old design we would sometimes use “X” and sometimes use “cancel.” In our revised design we are consistent about using “X” everywhere to close modal views.
- We have changed some colors to make the UI easier to read.

### Old Design



Hard to read

### New Design



New colors make text easier to read

## Prototype Implementation

### Tools

We implemented our hi-fi prototype using Sketch, Objective C, XCode and Nib files. We used Sketch to update our UI based on the feedback we received from the Heuristic Evaluation of our app and to create icons for the app. Meanwhile, we wrote the app in Objective C within the XCode environment. We used nib files to create base UIs graphically rather than programmatically, allowing us to experiment with different layouts more quickly.

### Wizard of Oz

We used Wizard of Oz techniques for messaging to convey the experience of messaging other users without building out the backend. Each time the user sent a message from the messaging tab, a few seconds later we designed the app to respond “Sounds good!” This allowed us to focus our energy on the interface and flow of the app rather than on its functional implementation.

## Hard-Coded Data

First, we hard-coded data for other users including their names, pictures, locations, statuses and messages. Second, we hard-coded the classes that users could add by using a curl request ExploreCourses. This returned an xml file, which we then parsed with a Python script we wrote and NSXMLParser in objective C to get all the names of classes offered. Thus it is not dynamic content. Finally, we hard coded message responses; all messages sent from the messages tab will receive the response, “Sounds good!”, 3 seconds after a message is sent. Finally, we have hardcoded the user’s location to be the d.school, since that is where the final project fair took place.

## What is Missing

Missing from our hi-fi prototype is an onboarding flow and the backend to register and keep track of new users, a working messaging feature, and a classes list that dynamically links to Stanford’s servers and displays classes appropriate to the quarter. We would also need to implement geotagging to appropriately order other nearby users.

## Summary

9 weeks ago we set out to create an app inspired by the theme of Focus. Many interviews, brainstorms, iterations, prototypes and discussions later, we are proud to present NightOwl, an app that allows college students to find and connect with classmates who are awake, nearby and working on the same assignment. Though we began skeptical of the d.school process, we conclude this project with newfound respect and admiration for the colored sticky notes. We empathized with our audience, and through open-minded exploration arrived at a topic we, and people in our audience, were excited about. We defined, taking time to think deeply about the core problem we were trying to solve, and with that in mind designed our solution to ensure it addressed that problem. We ideated, with endless whiteboard sessions and Post-It notes, and from all those brainstorms came up with far better ideas than we would have sitting around calling out a few thoughts. We prototyped, with experience prototypes, concept sketches, UI sketches, and low-fi, medium-fi and hi-fi prototypes, and ended up with a hi-fi version that has been designed thoughtfully and carefully. And at each stage, we tested, and benefitted immensely from the feedback we received from our classmates, our TA, our professor and our target users, resulting in a product that worked for people from an array of perspectives. We also grew in our ability to work as a team, supporting one another throughout the long nights. And so, we leave CS147 with a hi-fi prototype that we are proud of, a greater appreciation for and understanding of the design process, and better skills working on a team towards a common goal.