

Evaluating GrizSpace without users

Jon-Michael Deldin (Group 8)

CS 576

Project 3: Evaluating without users

2012-03-27

Table of Contents

- [1 Iterative design](#)
- [2 Audience](#)
- [3 Cognitive walkthrough](#)
 - [3.1 Add course to course list \(build a class schedule\)](#)
 - [3.1.1 Tap "Courses"](#)
 - [3.1.2 Tap "Add Course"](#)
 - [3.1.3 Scroll to find a subject or enter the subject name at the top](#)
 - [3.1.4 Tap the subject](#)
 - [3.1.5 Tap the "Select Course" button](#)
 - [3.1.6 Tap the course from the selection screen](#)
 - [3.1.7 Tap "Add To Course List"](#)
 - [3.2 Find an added class](#)
 - [3.2.1 Tap "Courses"](#)
 - [3.2.2 Tap "ECON 511"](#)
 - [3.2.3 Tap "Find It!"](#)
 - [3.2.4 The end screen](#)
 - [3.3 Delete a class](#)
 - [3.3.1 Tap "Courses"](#)
 - [3.3.2 Swipe from right to left over the course](#)
 - [3.3.3 Tap "Delete"](#)
 - [3.4 Find a building](#)
 - [3.4.1 Tap "Find Building"](#)
 - [3.4.2 Enter the name at the top](#)
 - [3.4.3 Tap the desired building](#)
 - [3.4.4 Final result](#)
 - [3.5 View schedule](#)
 - [3.5.1 Tap "Courses"](#)
 - [3.5.2 Tap "Schedule View"](#)
 - [3.5.3 Scroll to see the entire schedule](#)
 - [3.6 Summary](#)
 - [3.6.1 Re-sketch](#)
- [4 Heuristic evaluation](#)
 - [4.1 Home screen](#)
 - [4.1.1 Re-sketch](#)
 - [4.2 Add Course](#)
 - [4.3 Finding a course on a map](#)
 - [4.3.1 Re-sketch](#)
 - [4.4 Find building](#)
 - [4.4.1 Re-sketch](#)
 - [4.5 I'm Bored](#)
 - [4.5.1 Re-sketch](#)

1 Iterative design

Based on the feedback from Project 2, we

- combined functionality of find course and course list into one tab
- added a new schedule view that shows the user's added courses with times and locations in chronological order
- added a "view all courses on map" button on the courses page
- added building abbreviations to the "find a building" screen
- removed the title on the map page and navigation bar to increase screen real estate
- added course details to the map pin, instead of just showing the room number

We have not incorporated the following:

- enable custom courses

We want to cover the 80% of our user's needs. Freshman are very likely to have a stable course that is listed on the registrar's schedule.

- remove map, hybrid, and satellite view options

We feel these can be useful, especially the map view since the satellite view is outdated (new journalism building is still being constructed).

- remove the map tab

At the time of our medium fidelity prototype, we intended to use this for showing the upcoming class and the user's current location. However, results of the heuristic evaluation indicate we should drop this tab as well.

- integrate with Cyberbear or the Academic Planner

We will try to add this functionality and cut through IT red-tape once we have the initial version of the app done.

2 Audience

Freshman and transfer students to UM are the target audience. These students are technologically inclined, as they own iPhones. Since Verizon carries iPhones, it is much becoming more common for new smartphone users to pick iPhones.

These users have backgrounds in using technology to accomplish tasks, such as searching on Google for a phone number instead of using a phone book, or relying on GPS instead of a map to get to a destination. As new students, they will need help finding their classes, and the most logical source is an easy-to-use application. They are also used to responsive applications like Facebook and Google Search.

These students may be novice iPhone users (i.e., just set it up and downloaded one or two apps) and not know every shortcut or gesture, especially if they just started their own cell

phone plan after moving out of the house.

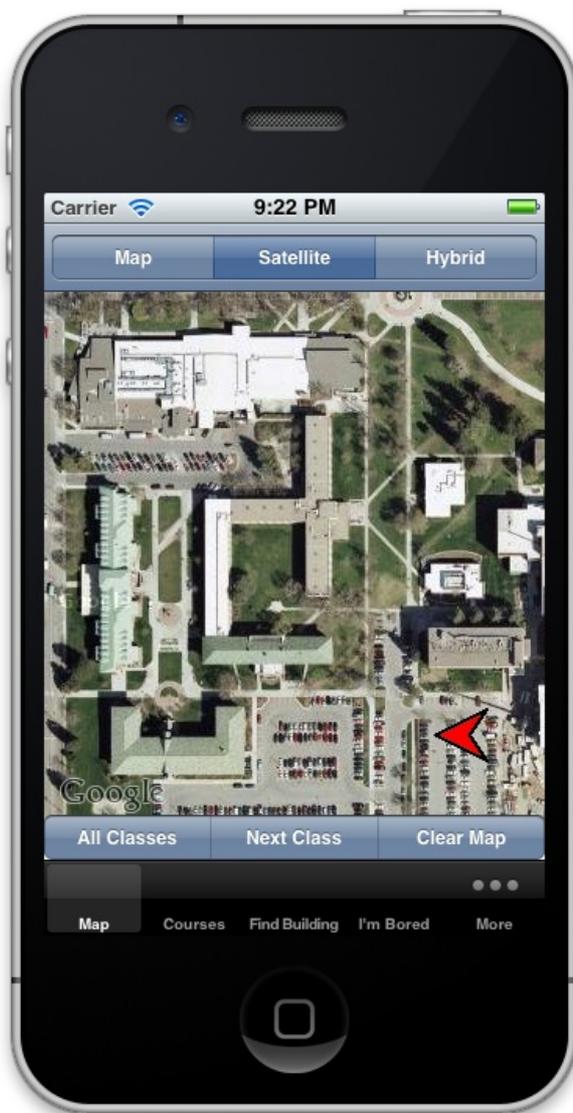
They will use this application heavily during the first weeks of each semester to plan their schedules and to attend their classes.

3 Cognitive walkthrough

Note: For the cognitive walkthrough, I took a binary approach to answering questions. If there is a "maybe" or a "but...", then the interface needs improvement.

3.1 Add course to course list (build a class schedule)

3.1.1 Tap "Courses"



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. "Courses" is one of the main navigation tabs.

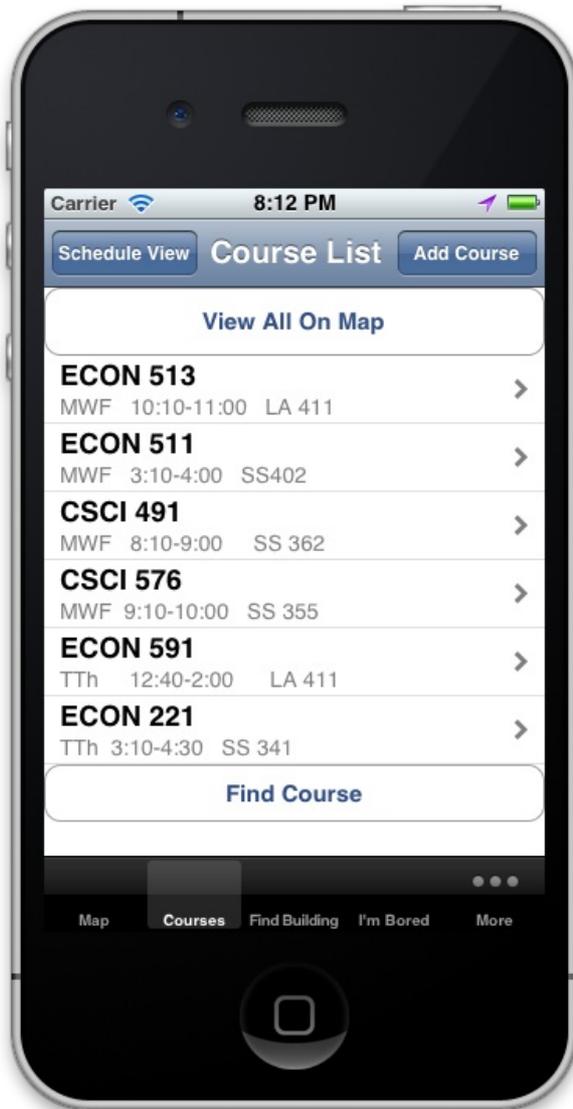
3. Once users find the control, will they recognize that it produces the effect they want?

No. It is not obvious that "Courses" is the user's course list. They may select "More" in search of "add class" or "add to schedule"

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. They will see the "Add Course" button on the resultant screen.

3.1.2 Tap "Add Course"



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. They will see the "Add Course" button.

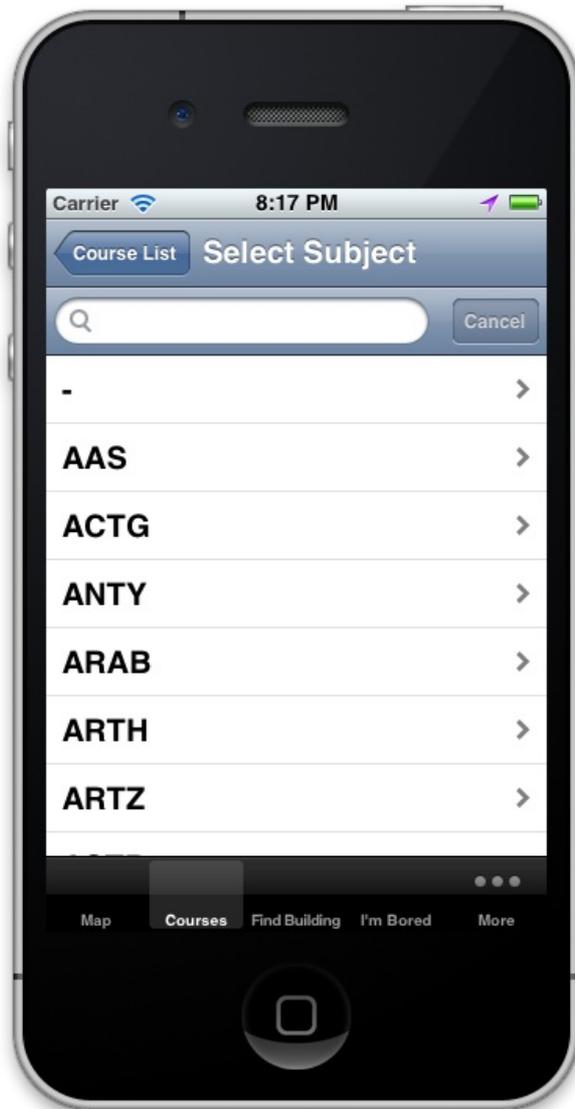
3. Once users find the control, will they recognize that it produces the effect they want?

Yes. It is clearly labeled to avoid any confusion.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. They will realize they can select a subject, as they can in Cyberbear, to get to a course list.

3.1.3 Scroll to find a subject or enter the subject name at the top



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. The text box has a magnifying glass icon used in other applications to indicate a search field.

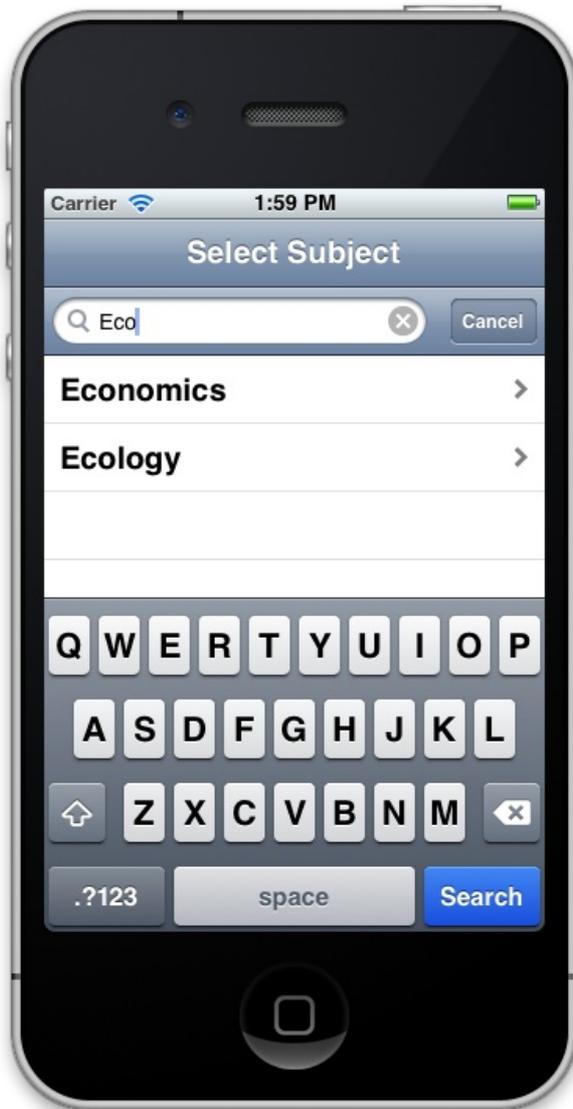
3. Once users find the control, will they recognize that it produces the effect they want?

Yes. It is obvious they can enter their subject name in the search box or scroll.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. They can see the results immediately.

3.1.4 Tap the subject



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. The subject list narrowed in the previous step to minimize their gulf of execution.

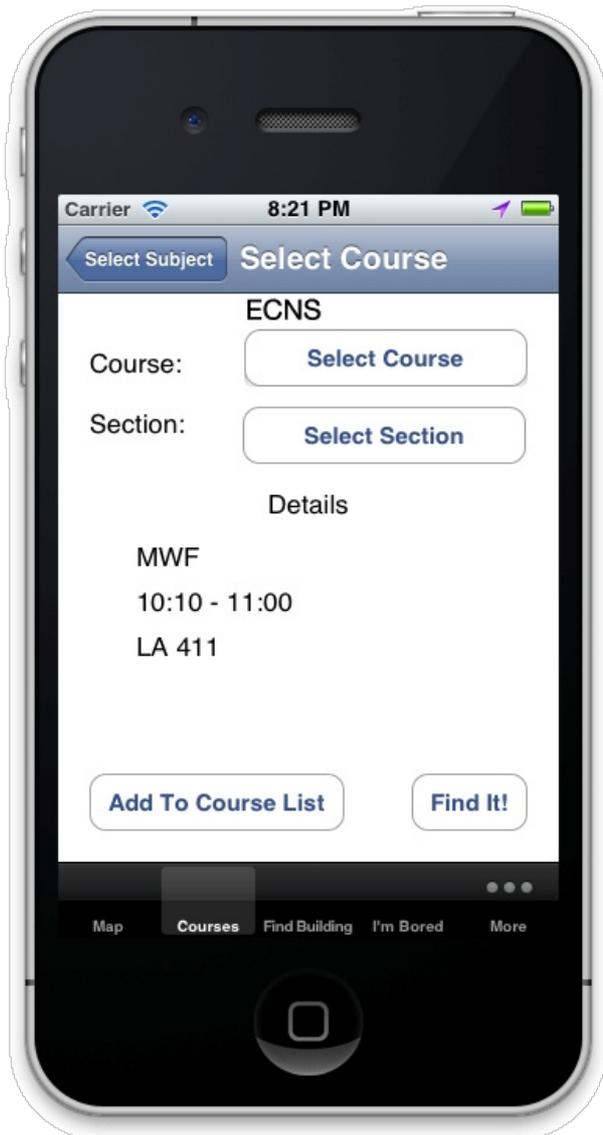
3. Once users find the control, will they recognize that it produces the effect they want?

Yes. There is an arrow on the right that is an iOS heuristic to indicate selecting the item will slide to a new screen.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. Tapping the subject will load a clearly labeled "Select Course" screen.

3.1.5 Tap the "Select Course" button



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

No. The "select course" button does not look like a button.

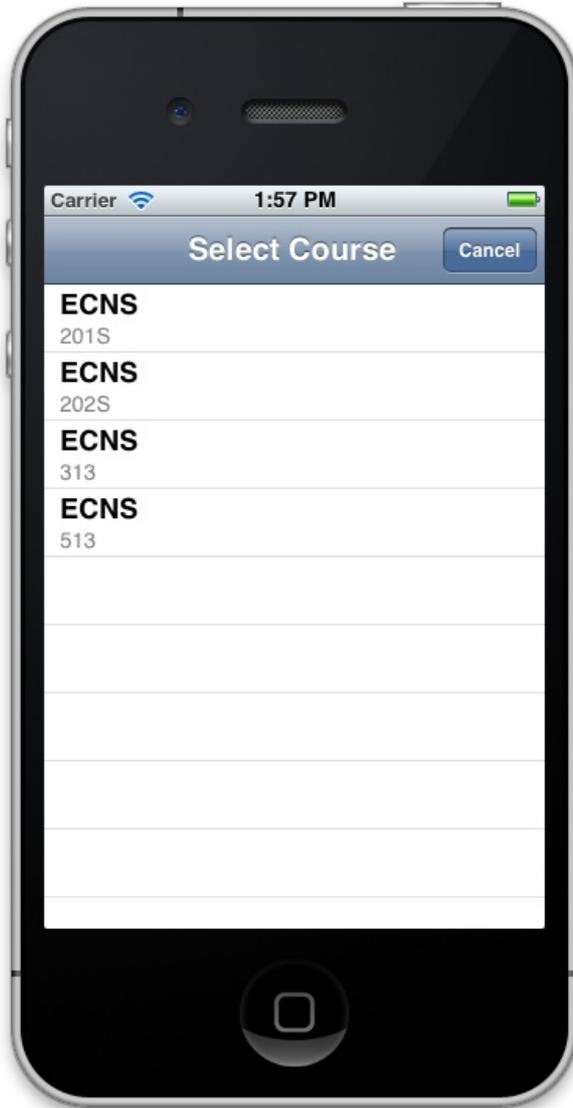
3. Once users find the control, will they recognize that it produces the effect they want?

No. While the wording instructs them to select a course, there is a button labeled "Find It" that could be confusing.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. A screen to select the course will appear.

3.1.6 Tap the course from the selection screen



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

No. It is not obvious the user should tap the courses.

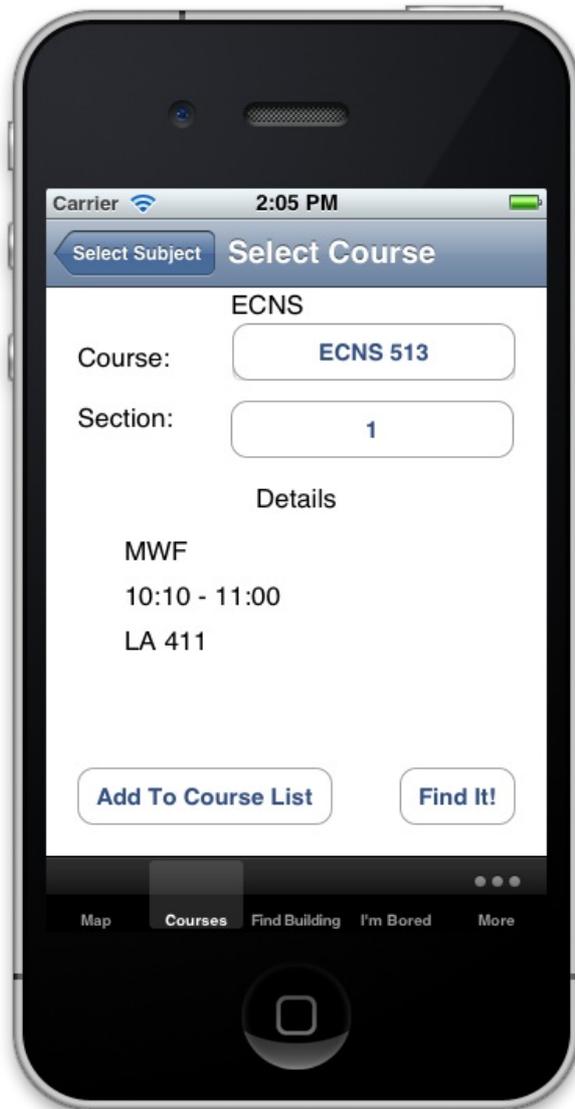
3. Once users find the control, will they recognize that it produces the effect they want?

Yes. There is nothing else on the screen that can be tapped that looks like it could add the course.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. The user is taken back to the course details screen, which now has their course info populated.

3.1.7 Tap "Add To Course List"



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

*No. There is a large gulf of execution, as there are **five** similar-looking buttons on the same screen, and the button is not on the right side where the user is more likely to look.*

3. Once users find the control, will they recognize that it produces the effect they want?

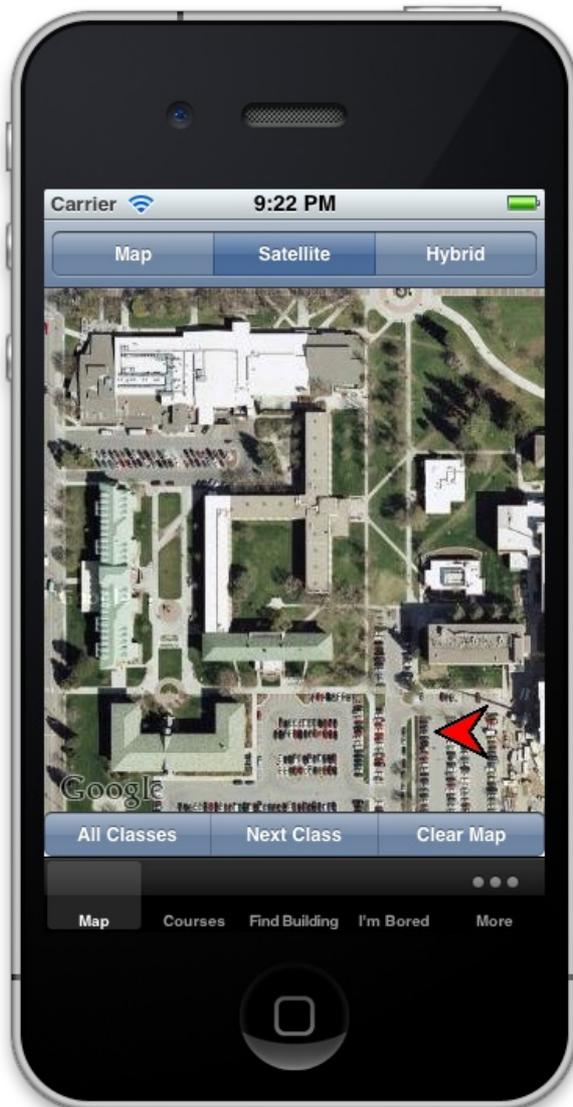
Yes. The wording instructs them it will add the course.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

No. There is no obvious feedback because the course is silently added to their course list.

3.2 Find an added class

3.2.1 Tap "Courses"



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. "Courses" is a main navigation tab.

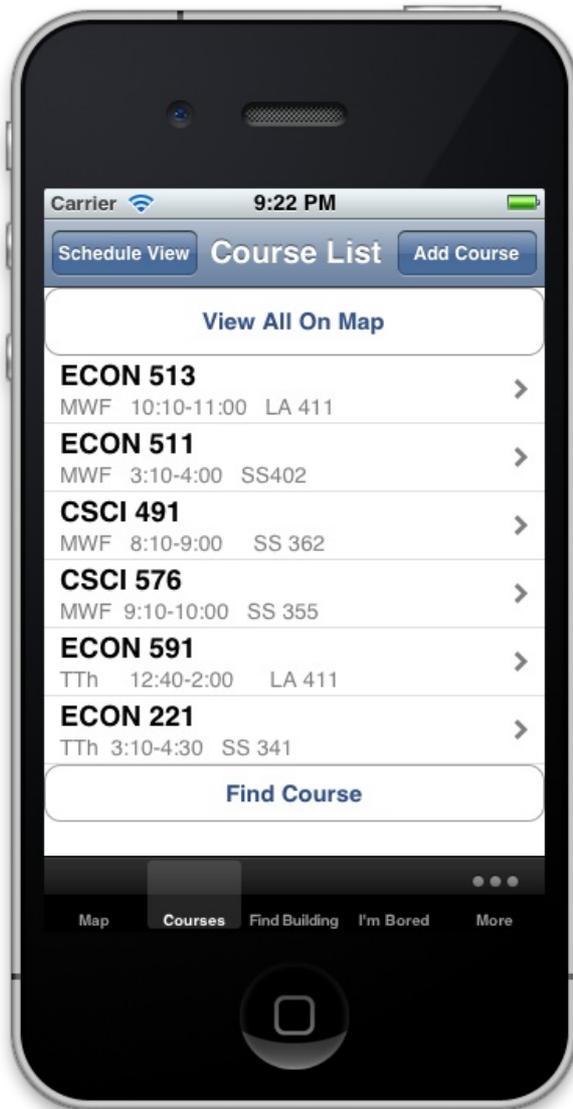
3. Once users find the control, will they recognize that it produces the effect they want?

No. The label doesn't indicate they'll be taken to their course list. While they may recognize "Courses" from adding a class, they may forget on second use.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

No. A wider gulf of evaluation exists because their course list is sandwiched between two buttons, and it is not clearly labeled.

3.2.2 Tap "ECON 511"



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. It uses a common iOS UI heuristic to indicate something is a button with the small arrow.

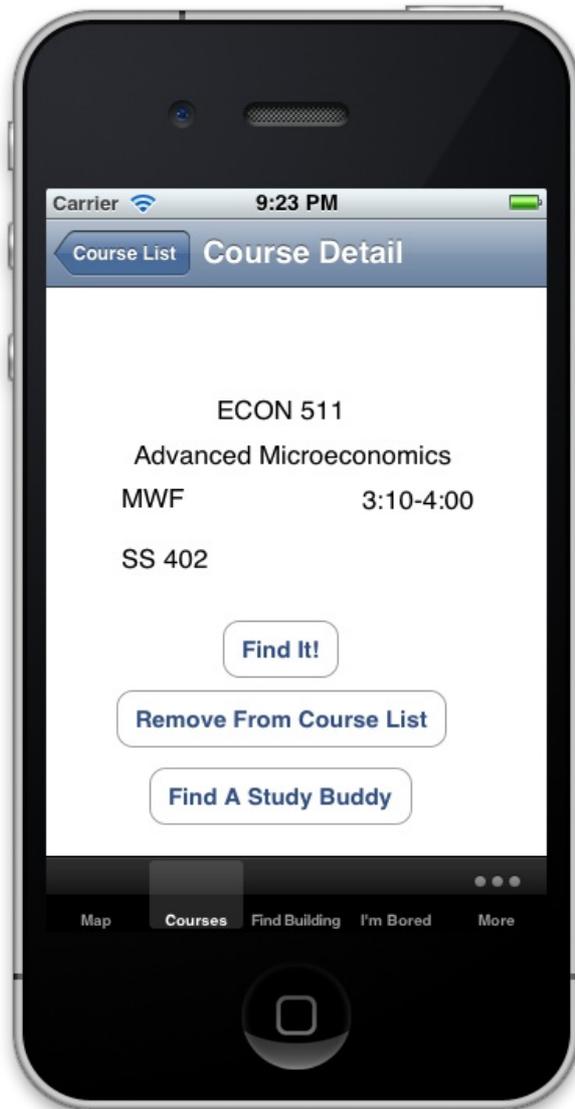
3. Once users find the control, will they recognize that it produces the effect they want?

Yes. They will understand the arrow and text label will link them to more information about the course.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. The next screen shows the class details and a button labeled "Find It!"

3.2.3 Tap "Find It!"



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. It is the first button in a column under their class name.

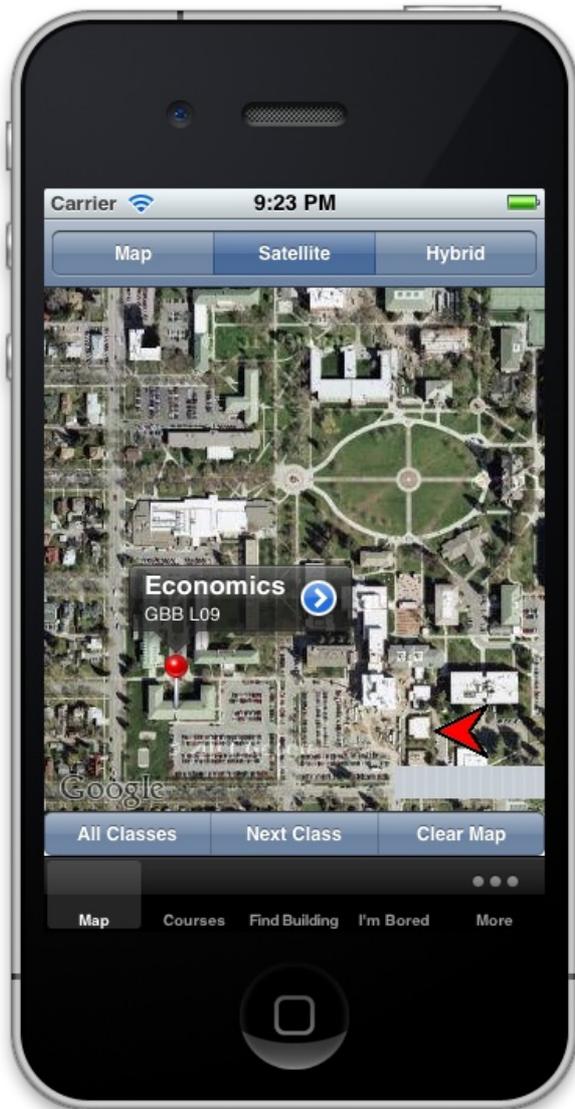
3. Once users find the control, will they recognize that it produces the effect they want?

Yes. The button is clearly labeled "Find It!".

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

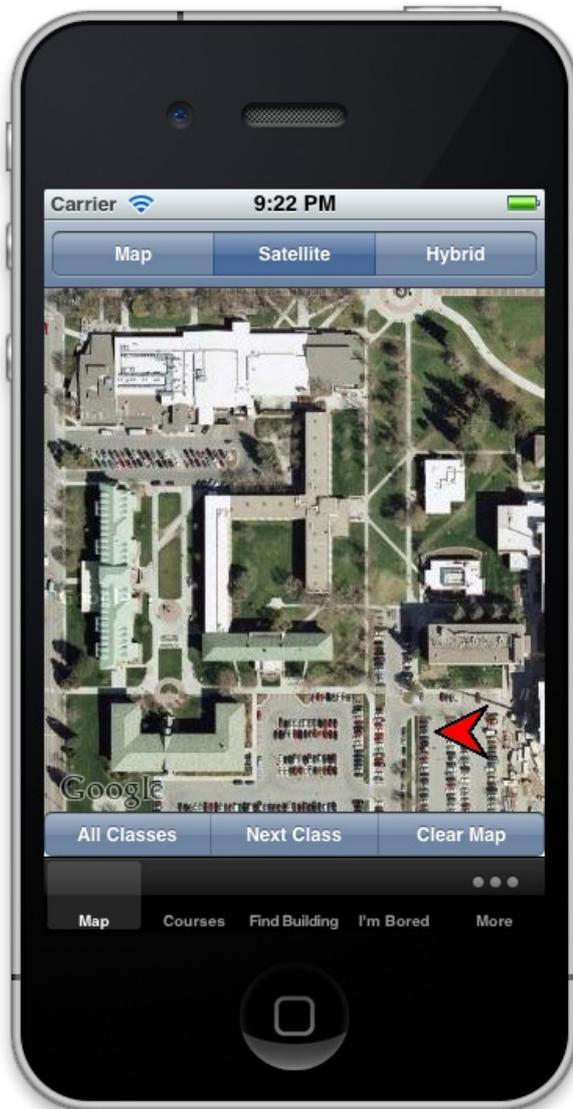
Yes. The next screen will show them a map with their current location and a labeled pin of their class.

3.2.4 The end screen



3.3 Delete a class

3.3.1 Tap "Courses"



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. "Courses" is one of the main navigation tabs.

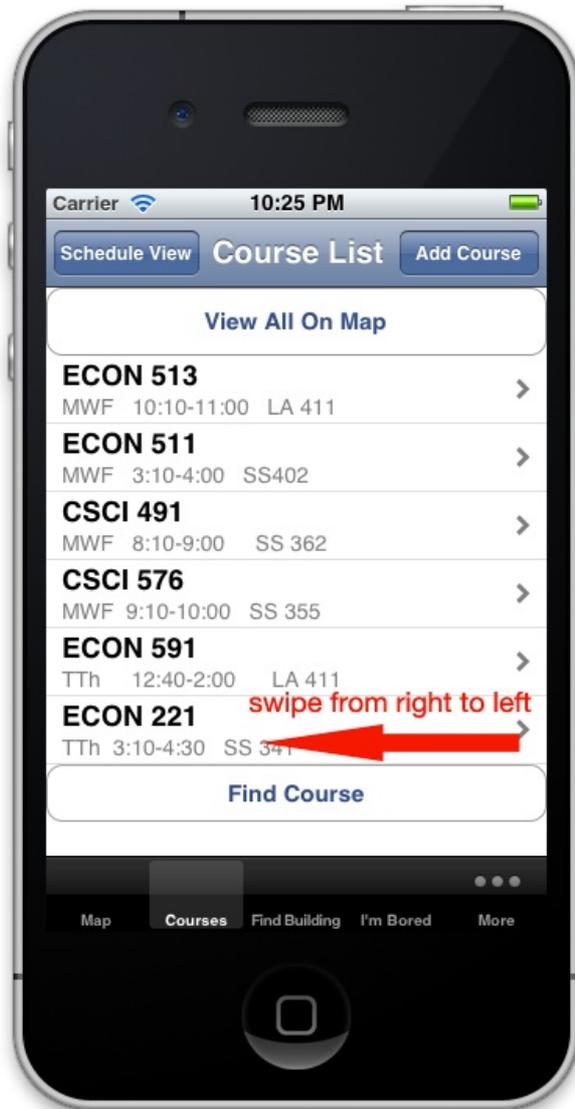
3. Once users find the control, will they recognize that it produces the effect they want?

No. "Courses" is too vague, so users may not know this is where they can edit their schedule.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

No. The resulting screen doesn't give any indication that the user can drop a course.

3.3.2 Swipe from right to left over the course



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

No. It is a gesture, and there is no indication the user should swipe.

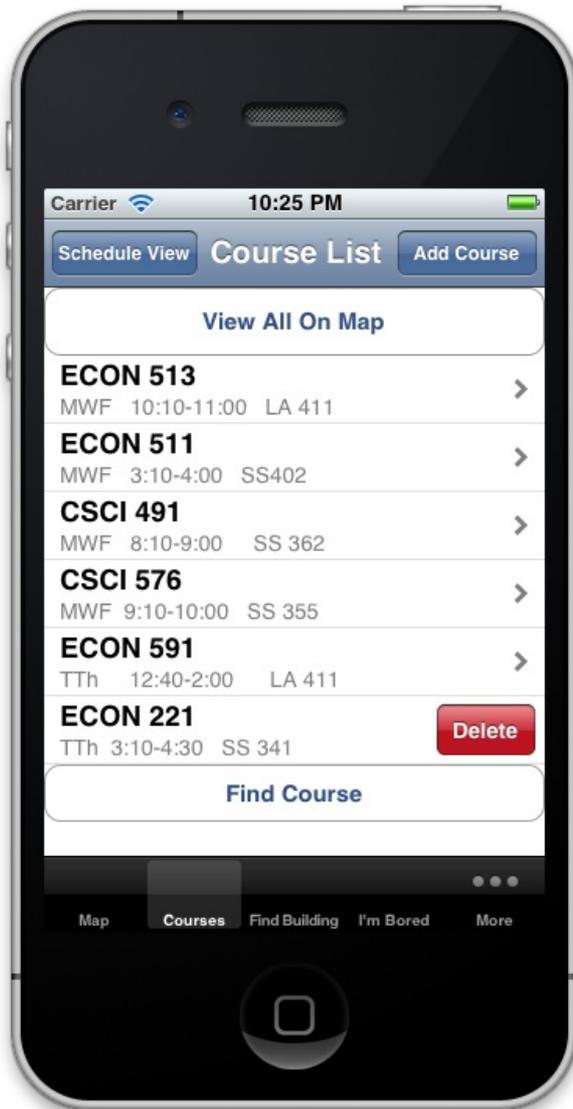
3. Once users find the control, will they recognize that it produces the effect they want?

No. Only if the user is an experienced iPhone user will he or she assume swiping from right to left is a delete action.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. A delete button is revealed to the right of the course.

3.3.3 Tap "Delete"



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. It is bright red and located to the right of the class they want to delete.

3. Once users find the control, will they recognize that it produces the effect they want?

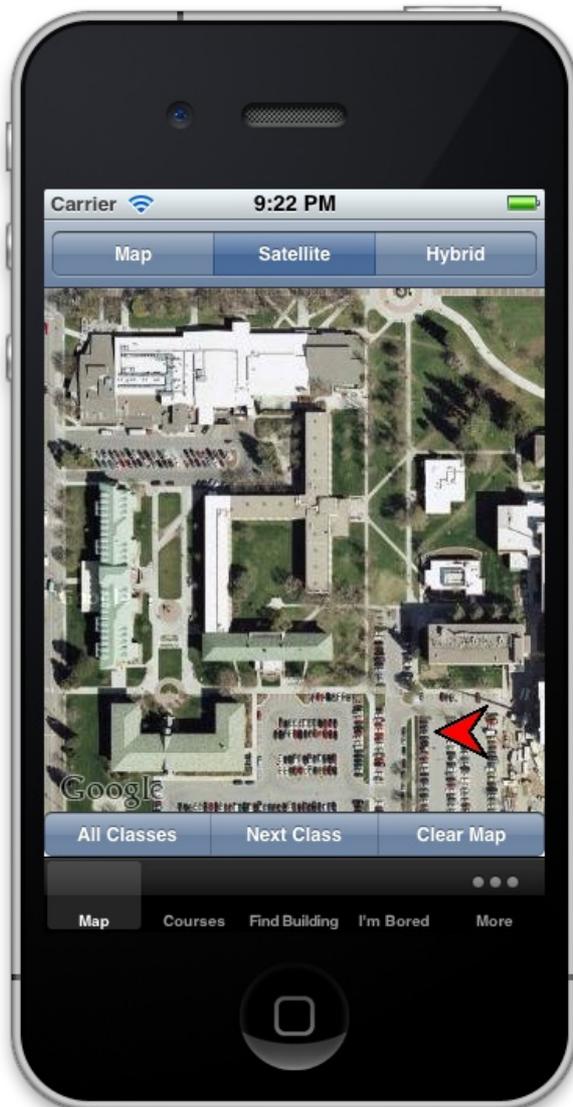
Yes. It says "Delete" and looks like other iOS delete buttons.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. The course will disappear from the list.

3.4 Find a building

3.4.1 Tap "Find Building"



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. The button is one of only a few main navigation tabs.

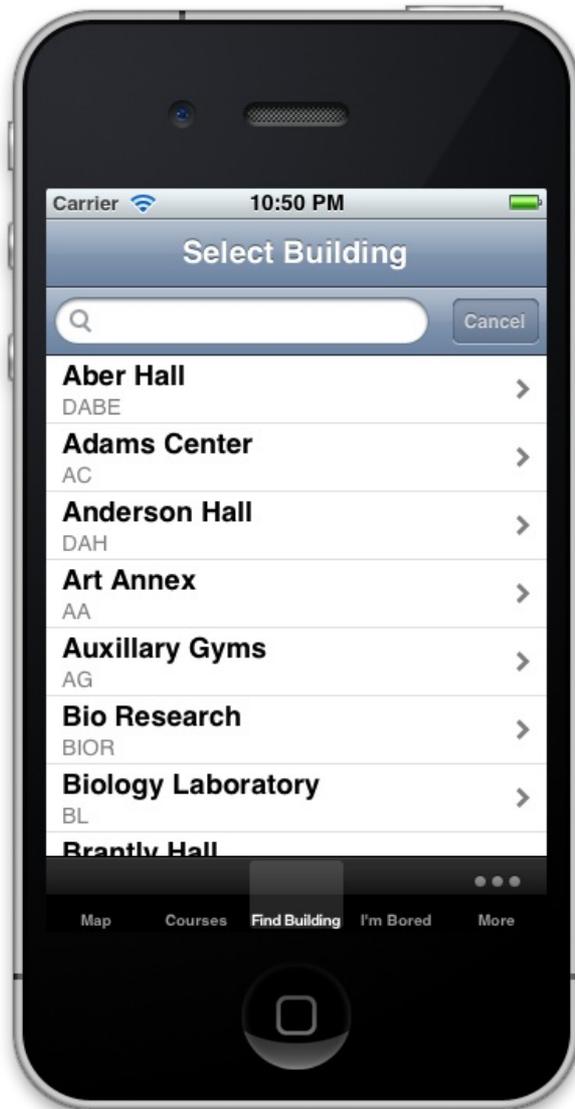
3. Once users find the control, will they recognize that it produces the effect they want?

Yes. It is clearly labeled "find building", so users know they will get to search for a building on the next page.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. The next screen is clearly labeled "Select Building" and features a magnifying glass to represent searching.

3.4.2 Enter the name at the top



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. They will notice the buildings are listed in alphabetical order and can scroll because some buildings will be peeking through. They will also see the large search box.

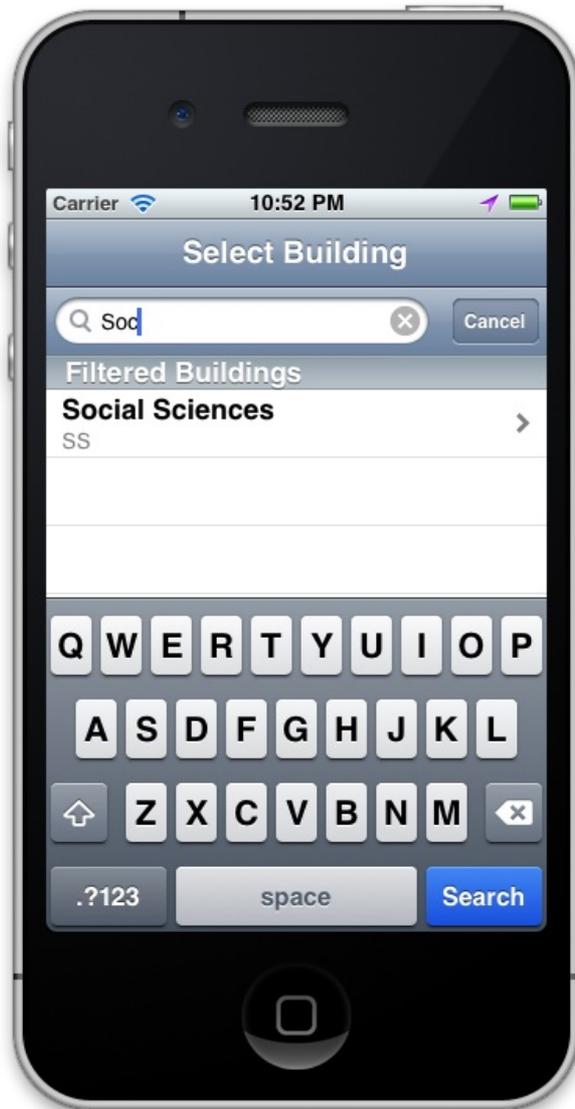
3. Once users find the control, will they recognize that it produces the effect they want?

Yes. If they want to search, the search box has a large magnifying glass. Tapping inside this launches the keyboard to type in a result.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. If they are filtering with the text box, then they will see the results update in real-time.

3.4.3 Tap the desired building



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. Only the building they want will appear if they have filtered with the text box.

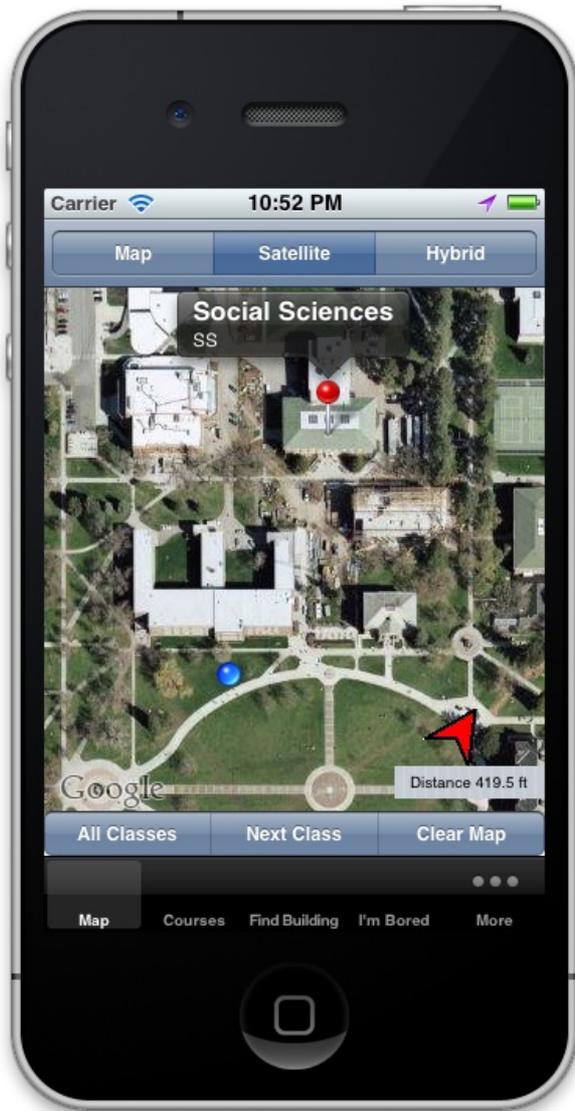
3. Once users find the control, will they recognize that it produces the effect they want?

Yes. The building has a right-indicator arrow that is a common iOS heuristic to indicate a button.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

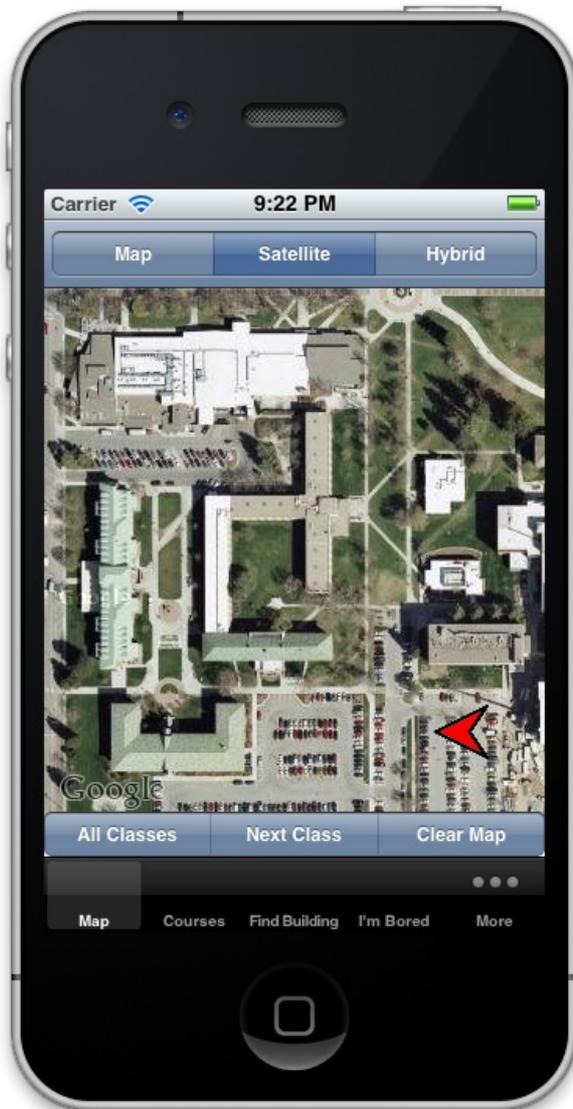
Yes. Tapping the building will load the map with pins for each point.

3.4.4 Final result



3.5 View schedule

3.5.1 Tap "Courses"



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. "Courses" is a main navigation tab.

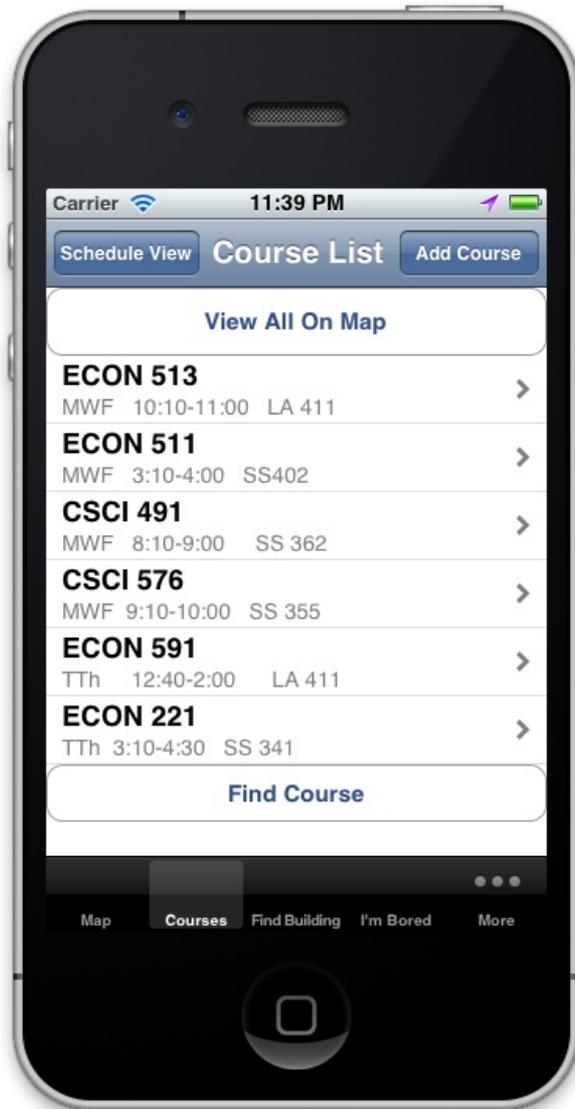
3. Once users find the control, will they recognize that it produces the effect they want?

No. It is not obvious that "Courses" contains the schedule. They will not see anything that would appear to contain their schedule.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. They will see a button for "Schedule" on the resulting screen.

3.5.2 Tap "Schedule View"



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. It is one of the major, dark blue buttons.

3. Once users find the control, will they recognize that it produces the effect they want?

Yes. It is clearly labeled "Schedule View".

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. The next screen will show their schedule.

3.5.3 Scroll to see the entire schedule



1. Will users be trying to produce whatever effect the action has?

Yes.

2. Will users see the control (button, menu, switch, etc.) for the action?

Yes. They will notice Monday is the first item, followed by Tuesday, and so on. This indicates more days are further down.

3. Once users find the control, will they recognize that it produces the effect they want?

Yes. Starting to scroll will move the content instantly.

4. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Yes. They will see the immediate effect of scrolling with content moving up or down depending on the direction they scrolled.

3.6 Summary

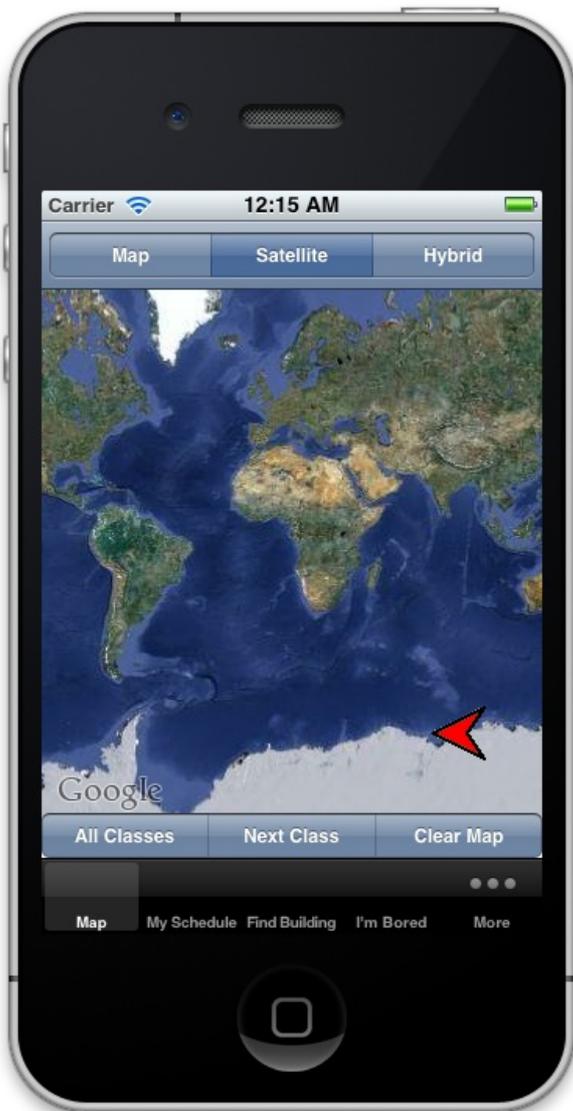
The cognitive walkthrough illuminated a number of problematic sections:

- *the "Courses" tab is too vague given how much functionality is hidden behind it*
- *the schedule feature is tucked away, even though it will be frequently accessed*
- *relying on gestures alone will make the gulf of execution limitless for some users*
- *when lists are on the resultant screen, the user should be alerted of what action he or she took in the form of a modal window or highlighted result to decrease the gulf of evaluation*
- *whenever possible, constraints should be provided to minimize user error and confusion*

These issues highlight the evolution of the app from the initial "find a class" stage to "manage a class schedule". After much evaluation, it is clear the latter will be the more common use case, and as such, eliminating some of the duplicated and confusing functionality is in order.

3.6.1 Re-sketch

- Rename "Courses" tab to "My Schedule"



- Make the new "My Schedule" tab show the user's schedule

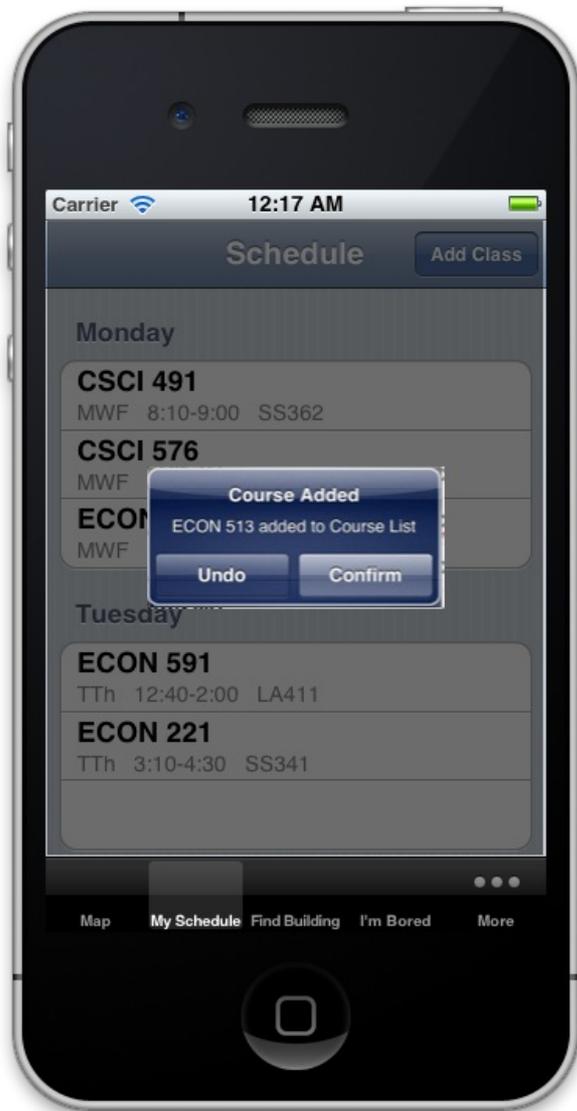


This removes the separate "Find Course" button, so a user can only add classes and not just search for them.

- Remove the swipe gesture

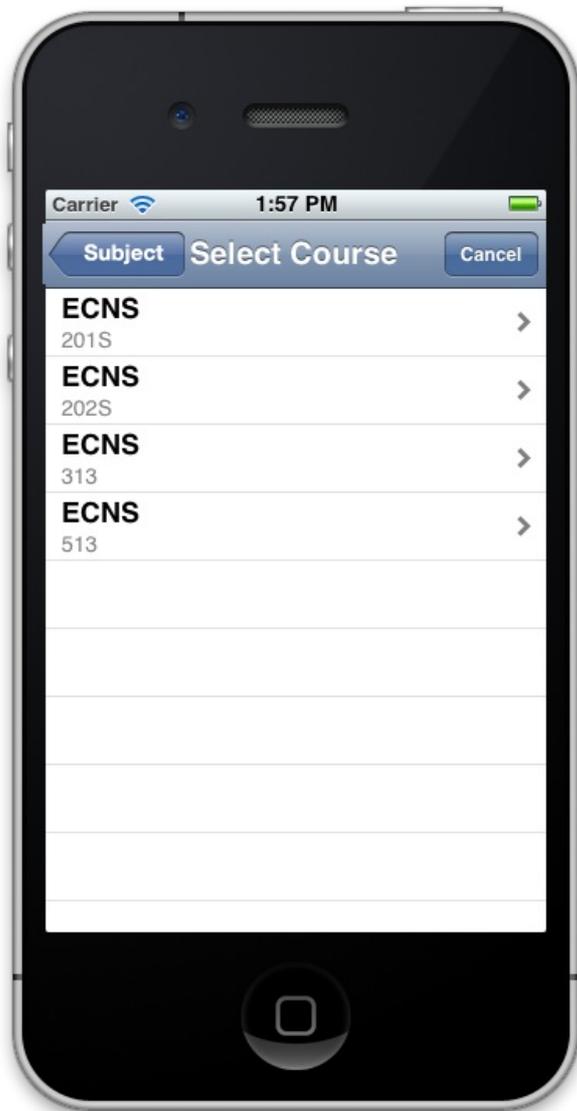
No re-sketch is necessary. When a user selects a class from the schedule view, he or she can press "delete" on that screen and be returned to the list.

- Highlight new actions with modal boxes



Along with removing the plain "find a class" (but not add) feature, presenting a modal window to confirm the add is more efficient and decreases the gulf of evaluation when adding a new course..

- Automatically guide user through adding a course and section



The app should be consistent with the subject selector, which only lets the user go forwards or backwards. The same change pictured for the course selector would be made for sections as well. However, if there is only one section, then the app will skip past the screen and add the class to the schedule.

4 Heuristic evaluation

The following interfaces were heuristically evaluated from the perspective of a new user who just installed the app:

- home screen
- add course
- finding a course on a map
- find building
- "I'm Bored"

4.1 Home screen

- *Simple and natural dialog*

While the dialog is simple and clear, why is the "Map" tab first? It doesn't seem to provide anything useful.

- *Speak the user's language*

Work has been done to make the tabs more user-centric (e.g., "My Schedule" instead of "Courses").

- *Minimize user memory load*

With the improvements from the cognitive walkthrough, the user no longer needs to remember all of the functions hidden behind the former "Courses" tab.

- *Be consistent*

The app is consistent with other iOS apps in terms of having main navigation as tabs at the bottom.

- *Provide feedback*

The "Map" tab is highlighted when the user is on it, as are the other tabs when the user navigates to them.

- *Provide clearly marked exits*

This is provided by the iPhone's only hardware button. In the app itself, selecting a tab will transition to another independent function.

- *Provide shortcuts*

No shortcuts are provided, since everything is within reach of the user's thumb.

- *Good error messages*

Nothing happens on the home screen, so no errors will occur on it.

- *Prevent errors*

Again, there is nothing that will error out.

4.1.1 Re-sketch



Removing the map tab will reduce confusion and make the dialog more natural.

4.2 Add Course

In this evaluation, the user wants to add a course to his or her schedule.

- *Simple and natural dialog*

The "My Schedule" tab is self-evident. The same issue from the home screen occurs again though; "Map" is confusing – can I get directions to anything? "Add Course" is obvious.

- *Speak the user's language*

On Cyberbear, schedules are presented as "Student Detail Schedule", so this "My Schedule" is obvious and short enough. "Subject", "Course", and "Section" are all familiar terms from adding classes on Cyberbear.

- *Minimize user memory load*

The app excels here. The user is funneled through selecting a subject, class, and section, without having to think about what they were doing beforehand.

- *Be consistent*

The interface is consistent with other iOS apps. The subject, course, and section screens now operate the same too. Additionally, the operation is similar to filling out a paper add form or searching on Cyberbear.

- *Provide feedback*

The user gets feedback along the way and afterwards. While adding a class, she can see where she is in the process with a "back" button labeled with the previous screen's name. Once they add a class, a modal window tells them the class was added.

- *Provide clearly marked exits*

Tapping any of the other tabs will "pause" adding a class. The user can use the iPhone hardware button to exit too. If the user adds the wrong class, he can "undo" the operation with a clearly marked "undo" button.

- *Provide shortcuts*

Shortcuts are provided in the form of a search box. The user can scroll or start typing part of a subject or course to save time.

- *Good error messages*

If a user is too specific in his or her query, perhaps a "no results" message should be displayed. Currently, the user gets whitespace as a result, and it happens in real-time, so it serves as a visual error message.

- *Prevent errors*

The only user error could be adding the wrong class. One option would be to confirm the user choice with a separate screen before adding the class, but I believe the current modal window works better.

4.3 Finding a course on a map

- *Simple and natural dialog*

What do the "All Classes" and "Next class" buttons do when I'm looking at directions to another class? Why are they there?

The compass arrow does a great job of indicating the user's direction without N/W/S/E clutter.

- *Speak the user's language*

The app does use some Google Maps terminology, but these terms are familiar to anyone who has used Google Maps or another online mapping program, as our core audience surely has.

- *Minimize user memory load*

The class the user is seeking is displayed with a pin and a tooltip, along with the building abbreviation and room number. If space wasn't an issue, it would be ideal to display the full building name, so the user didn't have to remember the abbreviation. However, the user can tap the pin to view more details about the class, and on this page, the full building name should be displayed.

- *Be consistent*

The interface is consistent in appearance and gestures to Google Maps.

- *Provide feedback*

The user will always know his or her location with the blue orb and its real-time positioning. Additionally, the compass arrow tells them what direction to head in.

- *Provide clearly marked exits*

Same as the "Add Class" evaluation.

- *Provide shortcuts*

The user can pinch and zoom in the map like other mapping programs.

- *Good error messages*

No error messages are displayed on this page.

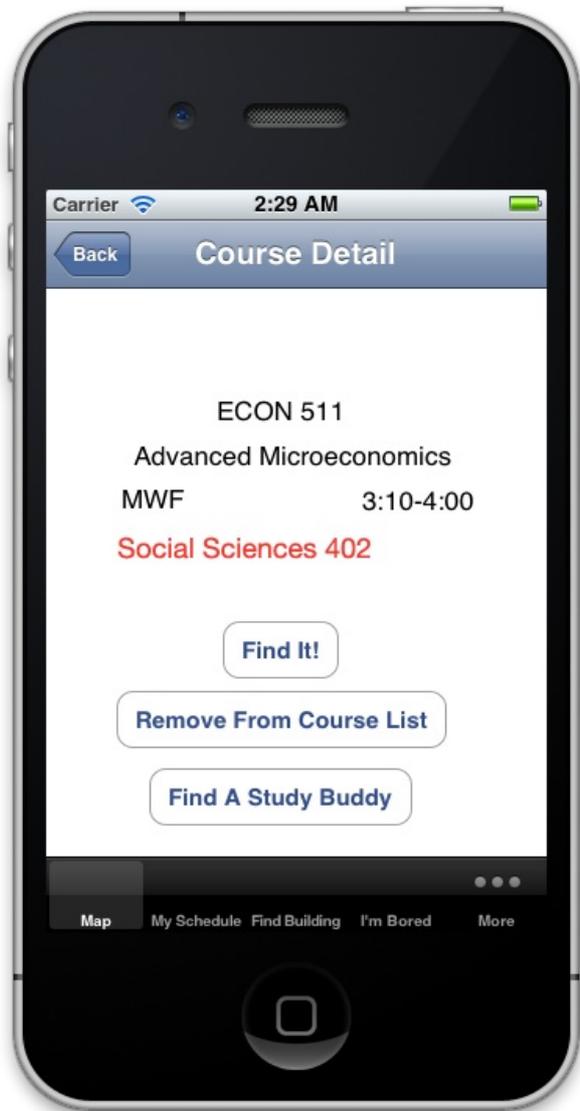
- *Prevent errors*

Disabling or removing the clear map button would prevent accidentally removing the directions.

4.3.1 Re-sketch



This eliminates unnecessary and confusing dialog.



This reduces memory load by not having to remember what "SS" stands for.

4.4 Find building

- *Simple and natural dialog*

"Find building" is very clear.

- *Speak the user's language*

"Find building" is consistent and common with a user's query.

- *Minimize user memory load*

The user doesn't have to remember anything about the app when wanting to find a building. They just need to know the name or abbreviation of the building. Being able to search by abbreviation enables the user to forget the abbreviation to name mapping pre-GrizSpace students know. Additionally, the user can search for "library" and have the Mansfield Library show up as a result, which is great for freshmen and transfer students.

- *Be consistent*

The search/ scroll interface is used in the add course segment, and it is also used in other iPhone apps.

- *Provide feedback*

When searching, users are provided feedback with immediate results. Once they find their building in a list, selecting it will load a map with their location.

- *Provide clearly marked exits*

Same as the "Add Class" evaluation.

- *Provide shortcuts*

Same as the "Add Class" evaluation.

- *Good error messages*

Same as the "Add Class" evaluation.

- *Prevent errors*

Same as the "Add Class" evaluation.

4.4.1 Re-sketch



Giving the user feedback that their search query produced no results may be useful.

4.5 I'm Bored

- *Simple and natural dialog*

"Previous I'm Bored Events" is confusing. How can I replay something I've already done in real life?

- *Speak the user's language*

All activities are labeled with their familiar names. The "I'm Bored" button is strangely worded. It's like "I'm Feeling Lucky" in Google Search, but it might be clearer to use "Find an activity" or something similar. Is there really an appeal to finding a random activity?

- *Minimize user memory load*

The user has to remember the "Settings" button will allow them to modify their interests.

- *Be consistent*

Same as the "Find building" evaluation.

- *Provide feedback*

Feedback is provided when the user taps an event or "I'm Bored" in the form of a map with their location and the destination.

- *Provide clearly marked exits*

Same as the "Find building" evaluation.

- *Provide shortcuts*

No shortcuts are needed here.

- *Good error messages*

If the user tries to attend an old event, he will get an alert saying "Sorry, the event is no longer available".

- *Prevent errors*

If it's 13:55 and the user wants to attend an event at 14:00, a notice should display "Warning: The event finishes in 5 min".

4.5.1 Re-sketch



This removes the random activity button and lets the user select an upcoming event. This also more clearly labels the past activities the user has experienced.

