Mobile Student Lookup

Milestone 2 Hand-In Document

10/7/2011

Brandon Knight, Mark Vitale, Ann Say, Katie Greenwald

Contents

Executive Summary

The Mobile Student Lookup application is an application which will allow students to access available class schedule, room, and course ID information easily from their phone. There are many use cases, or steps which take place between the user and the software system to be defined within the Mobile Student lookup application. We will attempt to completely describe these actions and all of the steps required. Along with the use cases, we have updated the features listing, and we will map each feature to a related use case. Along with the use cases and features, we are creating storyboards which will make it easy for the client and the users to see exactly some example cases, or uses, the software could have. We spent time trying to make the storyboards intuitive and easy for the user or client to read through. We have provided constraints for our system, along with new references.

Introduction

The main focus of this document is detailing how the user will interact with the Mobile Student Lookup application. To this end, the new content in this second milestone includes use cases, or detailed interactions between the user and the system, as well as storyboards, or comic-like visualizations of how the user will interact with the application. At the same time, some of the information from Milestone 1 was updated and included for reference. The features have been updated, and a constraints section has been added. The product perspective and user needs have been improved to more fully provide a view of the software from the user perspective and to include survey data that was collected on potential users. All of this information will help the client and developers better understand the context in which the use cases and storyboards are presented.

Product Perspective

The mobile student lookup application will be a native application for iOS devices (devices which run the mobile operating system distributed by Apple). This application will allow anyone with Rose-Hulman Network access (students or faculty) to access all schedule lookup tools provided by the Registrar's student lookup website. These tools include search by username (partial or whole), search by class ID, search by room number, and the ability to search through all past terms.

The student schedule system we are building would be a native mobile application. This is meant to replace the current web based system. The current system's data will not be able to be directly accessed, however. If we want to utilize the current system in any way, we will have to scrape the data from the website. For our mobile application, we will take the data that we scrape, and have a User Interface (UI) that is a skinned version of the web based one. We want to really simplify the UI, and make it intuitive. We were thinking of having only one search bar for user names. The current system and the application will have no interaction beyond simple data scraping. The mobile application will heavily rely on the registrar's student lookup page, but the registrar's page does not care that the mobile app exists.

User Needs

A survey was completed by forty-seven members of the student and faculty, who were asked for features and functionality that they would like to see in a Schedule Lookup Page app. Over half of the people who responded noted that the layout of the schedule lookup page needs improvement, especially on the iPhone or iPod Touch screen. Twelve people complained that logging in every time was annoying and that an app for the schedule lookup page should store that information. Other common suggestions were the ability to compare or overlay schedules, and bookmarking your own schedules and those of your friends. A few people surveyed said they would like to see more information on a person's page, such as their major and campus mailbox number, and some asked for the ability to search by a person's name, rather than username, or by a department. One faculty member also noted that he would like the ability to see a list of all the classes a student has ever taken.

Most common suggestions, such as a layout redesign and saving logins, are core requirements for this project. A few, such as integrating the study buddy tool, Angel, and class registration; showing maps of campus; or showing professors office hours, are out of the scope of the project. Others, such as displaying extra information on a student's page and rotating the interface when the device is rotated, would be of low importance if they were included at all.

Feature Listing

Below is a listing of the features for the mobile student lookup app. The priorities have been assigned based on input from the client, while effort, risk, and stability have been estimated from previous experience developing apps for the iOS platform.

1. Users will be able to look up a schedule for the current quarter by username.

Status	Accepted
Priority/benefit	Critical
Effort	Medium
Risk	Medium
Stability	High
Target Release	Version 1.0
Assigned to	Team
Reason	Required to satisfy basic project requirements

2. Credentials will be stored on first entry and will only be requested again if credentials are incorrect.

Status	Accepted
Priority/benefit	Critical
Effort	Low
Risk	Low
Stability	High
Target Release	Version 1.0
Assigned to	Team
Reason	Required to satisfy basic project requirements

3. Users will be able to view the roster of a class based on a class section.

Status	Accepted
Priority/benefit	Critical
Effort	Medium
Risk	Low
Stability	High
Target Release	Version 1.0
Assigned to	Team
Reason	Required to satisfy basic project requirements

4. All users will have instant access to their own schedule with a click of a button.

Status	Accepted
Priority/benefit	Critical
Effort	Low
Risk	Low
Stability	High
Target Release	Version 1.0
Assigned to	Team
Reason	This will reduce the amount of time needed to use

	the app when a user is trying to find their next class

5. Users will be able to view schedule information for all quarters supported by schedule lookup page

Status	Accepted
Priority/benefit	High
Effort	Medium
Risk	Medium
Stability	High
Target Release	Version 1.0
Assigned to	Team
Reason	Allows users to view historical or future information without needing a computer. Could be useful when seeing if a friend has already taken a class.

6. Look up schedule information based on class section number.

Status	Accepted
Priority/benefit	High
Effort	Medium
Risk	Medium
Stability	High
Target Release	Version 1.0
Assigned to	Team
Reason	Useful if attempting to view how many sections are currently offered or various professors teaching a certain class

7. Lookup a user's contact information.

Status	Accepted
Priority/benefit	High
Effort	Medium
Risk	Medium
Stability	High
Target Release	Version 1.0
Assigned to	Team
Reason	Often when looking up the schedule of others, the user intends to contact that user in some fashion.

8. Open a new email in the native email client addressed to another student or faculty when clicking on any email address in the app.

Status	Accepted
Priority/benefit	High
Effort	Medium
Risk	Low
Stability	High

Target Release	Version 1.0
Assigned to	Team
Reason	The ability to quickly compose an email to a student or faculty member from within the app would streamline even communication with others

9. Lookup a schedule based on room number.

Status	Accepted
Priority/benefit	Low
Effort	Medium
Risk	Low
Stability	High
Target Release	Version 1.0
Assigned to	Team
Reason	Students and teachers alike often wonder whether a certain room has class during a specific period to determine whether or not that room would be a useful meeting location.

10. Sync current schedule with calendar app simplistically.

Status	Accepted
Priority/benefit	Low
Effort	High
Risk	High
Stability	Medium
Target Release	Version 1.0
Assigned to	Team
Reason	This would provide a simple way to get schedule information into a user's calendar which will sync automatically to any backend calendar already used on the device.

11. Provide a layover of various schedules to help determine common breaks for meeting times.

Status	Accepted
Priority/benefit	Low
Effort	High
Risk	High
Stability	High
Target Release	Version 1.0
Assigned to	Team
Reason	This functionality is not present in the current schedule lookup page, but it is frequently the goal of users looking up schedules to identify common breaks for meetings. The app should at minimum support 4 users in layover view.

12. Access and maintain list of favorite users to quickly get up-to-date information on commonly viewed schedules.

Status	Accepted
Priority/benefit	Low
Effort	Low
Risk	Low
Stability	High
Target Release	Version 1.0
Assigned to	Team
Reason	The majority of users don't access a wide variety of schedules on a regular basis, but want to visit a few schedules quickly and frequently.

13. Send any schedule information to another person via email.

Status	Accepted
Priority/benefit	Low
Effort	High
Risk	High
Stability	Medium
Target Release	Version 1.0
Assigned to	Team
Reason	The intention of the user may be to look up a specific schedule in order to share it with group members or other interested parties. This would simplify that process.

Constraints

Several constraints on the project were discovered through conversations with the client. They are listed below.

1. The mobile application developed will be an iOS native app on iPhone and iPod Touch devices.
2. The development will be done using Xcode 4.3 and targeting iOS 5.
3. The app will follow all of the constraints discussed in Apple's Human Interface Guidelines.
4. The language used in the development of this app will be Objective-C
5. The project is operating with no budget at all.
6. The project must be completed by May, 2012 at which point all development will cease.
7. The capabilities of this software will be limited to the functionality of the web system, except where otherwise specified.

Use Cases

Global Pre Conditions

We are assuming that all users of the application will have a Rose-Hulman network account, if they do not, they will not be able to access any of the application's features. We also assume that:

1. The application is already running.
2. Except where specifically stated, the user's credentials are stored in the application settings page.
3. The user has entered a valid username or password. But if they haven't:
    a. The user will be denied access to the server functionalities.
    b. The user will view a warning from the system.
    c. The user will have to enter a username and password before proceeding.

Global Alternate Flow

If the server is not available, or if the user has entered an incorrect username or password, we would need a separate flow of events to describe the actions necessary in the respective situation.

1. If the username or password is incorrect when attempting to access the server:
    a. Prompt the user to go to the settings page in order to enter the correct username and password
    b. If the user does not have a Rose account, or cannot remember their password, our application will not handle these situations, because IAIT currently has webpages devoted to these activities that are beyond our control.
2. If the server cannot be reached:
    a. Display a warning to the user, then cancel the current action. To avoid errors, or something malfunctioning, we will simply display blank schedules if necessary.
        i. Nothing will be displayed
3. If the user cancels a current action:
    a. If the action is loading server information, the user will not be able to see the information and will be taken back to the previous screen.

1. Mobile Student Lookup Use Case: Invalid or No Credentials

Revision History			
Date	Issue	Description	Author
9/30/11	1.0	Initial Creation of “First Time Use” use case.	Brandon Knight
10/02/11	1.1	Revisions and word editing	Ann Say
10/3/11	1.2	Revisions and editing.	Brandon Knight
10/5/11	1.3	Changing name to “Invalid or No Credentials,” generalizing the use case	Mark Vitale

Relevant Feature(s): 2

Brief Description: This is the use case for the sequence of actions a user would have to perform whenever there are no credentials registered with the app or the credentials that are registered are invalid. This includes the first use of the application before the user has entered valid credentials.

Basic Flow:

1. If the app does not have any credentials registered or, when trying to query the schedule lookup page, the stored credentials fail, the application defaults to the “Settings” page, where the user must enter his or her username and password for his or her Rose-Hulman account.
2. The server verifies the credentials, and the first step is repeated until the credentials are found to be valid.
3. If this is the first use of the application, the user’s schedule is immediately loaded and displayed. In the case that previous credentials failed, the user is taken to the page they were trying to access when the previous credentials were refused.

Pre-conditions: We are going to assume that the application is running

Post conditions: The user will be able to see the most updated version of their personal schedule in the “My Schedule” tab of the application. The user will also be able to access any other part of the application without re-entering their password.

2. Mobile Student Lookup Use Case: View Own Schedule

Revision History			
Date	Issue	Description	Author
9/30/11	1.0	Initial Creation of “View Own Schedule” use case	Brandon Knight
10/3/2011	1.1	Revisions and editing, clarifying.	Brandon Knight

Relevant Feature(s): 4

Brief Description: This use case describes the actions a user would take to view his/her own schedule. The actions are simple, in that there is a separate section of the application that will directly retrieve the user’s schedule based on his/her username.

Basic Flow: The basic flow begins when the user opens the application. The application will perform the following upon opening:

1. Retrieve the user’s stored username and Kerberos password from the device’s memory.
2. Automatically use the username and password to attempt to access the server
3. Use the username on server to find user’s schedule and retrieve it
4. Display the updated schedule in a tabular format that orders the classes by time, earliest to latest.
5. The user is then free to view his or her schedule in the “My Schedule” tab, and perform other application features.

If the user has the application open, he or she is able to click a ‘refresh’ button on the screen, which will perform all of the above actions, therefore updating the schedule in the “My Schedule” tab.

Post conditions: The user views his/her most recent schedule through the “My Schedule” tab.

3. Mobile Student Lookup Use Case: Search by Username

Revision History			
Date	Issue	Description	Author
10/02/11	1.0	Initial Creation of “Search by Username” use case	Ann Say
10/03/11	1.1	Editing.	Brandon Knight
10/5/11	1.1	Added an alternate flow handling exact and partial matches	Mark Vitale

Relevant Feature(s): 1

Brief Description: This use case describes the sequence of actions the user and the system takes when the user decides he or she would like to search by a person’s username in the mobile application.

Basic Flow: This use case will begin when the user selects the “Search Bar.”

1. The user will select the “username” option.
2. The user will type a username that he or she wants to search.
3. The app will automatically use the stored Kerberos username and password to navigate to server and pull off all available information available from a “username search” on the website.
4. If the user typed in an exact match for an individual’s username, that individual’s information is displayed directly. If the user typed in a partial username, follow the alternative flow of events under “If the user only typed in part of a username.”
5. The application will display this information to the user formatted for the small screen.

Alternative Flow of Events:

If the user only typed in part of a username:

- Return a list of usernames that match the current search entry. The user should be able to scroll through the list and by selecting a username, they are taken to that person’s schedule. Go to step 5 in the basic flow.

If there are no usernames that match the user’s query:

- Display a message saying no people with that username were discovered.

Post conditions: The user views the schedule for the username entered.

4. Mobile Student Lookup Use Case: Search by Classroom

Revision History			
Date	Issue	Description	Author
9/30/11	1.0	Initial Creation of search by classroom use case	Brandon Knight
10/5/11	1.1	Added an alternate flow handling exact and partial matches	Mark Vitale

Relevant Feature(s): 9

Brief Description: This use case describes the sequence of actions the user and the system take when the user decides he/she would like to search by classroom in the mobile application.

Basic Flow: We are going to assume that the user has the application open already, so the basic flow begins when the user goes to the search tab of the application.

1. The user will select the search bar
2. The user will select the 'class room' option
3. The user will type a class number that they want to search
4. The app will automatically use the stored Kerberos username and password to navigate to server and pull off all available information available from a 'classroom search' on the website.
5. If the user typed in an exact match for a classroom number, that classroom's information is displayed directly. If the user typed in a partial classroom number, follow the alternative flow of events under "If the user only typed in part of a classroom number."
6. The application will display this information to the user formatted for the small screen.

Alternative Flow of Events:

If the user only typed in part of a classroom number:

- Return a list of classrooms that match the current search entry. The user should be able to scroll through the list and by selecting a classroom, they are taken to that classroom's schedule. Go to step 6 in the basic flow.

Post conditions: The user views the information for the classroom entered.

5. Mobile Student Lookup Use Case: Search by Course ID

Revision History			
Date	Issue	Description	Author
9/30/11	1.0	Initial Creation of search by course ID use case	Brandon Knight
10/5/11	1.1	Added an alternate flow handling exact and partial matches	Mark Vitale

Relevant Feature(s): 6

Brief Description: This use case describes the sequence of actions the user and the system take when the user decides he/she would like to search by course ID in the mobile application.

Basic Flow: We are going to assume that the user has the application open already, so the basic flow begins when the user goes to the search tab of the application.

1. The user will select the search bar
2. The user will select the 'course ID' option
3. The user will type a course ID number that they want to search
4. The app will automatically use the stored Kerberos username and password to navigate to server and pull off all available information available from a 'course ID search' on the website.
5. If the user typed in an exact match for a course ID, that course's information is displayed directly. If the user typed in a partial course ID, follow the alternative flow of events under "If the user only typed in part of a course ID."
6. The application will display this information to the user formatted for the small screen.

Alternative Flow of Events:

If the user only typed in part of a Course ID:

- Return a list of classrooms that match the current search entry. The user should be able to scroll through the list and by selecting a course ID, they are taken to that course's schedule. Go to step 6 in the basic flow.

Post conditions: The user views the information for the course ID entered.

6. Mobile Student Lookup Use Case: Future or Past Schedule Search

Revision History			
Date	Issue	Description	Author
10/2/11	1.0	Initial Creation of “Future or Past Schedule Search” use case.	Mark Vitale
10/3/11	1.1	Updated this use case.	Brandon Knight

Relevant Feature(s): 5

Brief Description: This use case describes the sequence of actions the user and the system take when the user is trying to find previous or future schedules for an individual.

Basic Flow: We are going to assume that the user is looking at the individual’s current schedule.

1. The user will press a button that indicates more options.
2. The app will check the schedule lookup page to view the available quarters for query (whether past or future).
3. The user will select one of these quarters from a list.
4. The app will find the selected individual’s schedule from the selected quarter.
5. The app will display that schedule to the user in the same format as all other schedules.

Alternative Flow of Events:

If the individual was not enrolled at Rose-Hulman during the selected quarter:

- Display a warning.
- Go back to the current schedule for the individual.

Pre conditions: We are assuming that the user is currently viewing an individual’s schedule in the current quarter.

Post conditions: The user views an individual’s past or future schedule information or is brought back to the user’s current schedule information.

7. Mobile Student Lookup Use Case: Overlaying Schedules

Revision History			
Date	Issue	Description	Author
10/02/11	1.0	Initial Creation of "Overlaying Schedules" use case.	Ann Say
10/03/11	1.1	Updated and edited Alternate Flow.	Brandon Knight
10/5/11	1.2	Added more steps to clarify the basic flow	Mark Vitale

Relevant Feature(s): 11

Brief Description: This use case describes the actions a user would take to overlay two or more schedules in the mobile application.

Basic Flow: The basic flow begins when the user reaches the first screen of the application, which would show their schedule.

1. A user can search for a schedule.
2. Once the schedule is viewed, the user chooses an option to overlay the schedule.
3. The user then searches for other schedules and repeats as necessary until all desired schedules are selected for overlay.
4. The user is then taken to the overlay screen where periods in which all schedules are free are made available.
5. The user, while viewing the overlay, can click a button to add more schedules to the current overlay. If the user wants to add more schedules, he or she is taken back to basic flow step 3 and all proceeds from there as usual.

Alternative Flow of Events:

The user may remove any number of schedules from the overlay while viewing the overlay

The user has tried to overlay too many schedules:

- Display a warning saying that no more new schedules can be placed in overlay.
- The most recent schedule that was attempted to be placed in the overlay (and was one too many) will not be added to the overlay.

The user wants to clear the overlay:

- An option in the overlay viewing window is the ability to clear the overlay. All schedules would be removed.

The user attempts to overlay the exact same schedule multiple times:

- No checks are made if the schedules are identical. Both schedules are placed in overlay mode.

Post conditions: Schedules are placed in the overlay view.

8. Mobile Student Lookup Use Case: Find Contact Information

Revision History			
Date	Issue	Description	Author
10/2/11	1.0	Initial Creation of “Find Contact Information” use case.	Mark Vitale
10/3/11	1.1	Edited.	Brandon Knight

Relevant Feature(s): 7

Brief Description: This use case describes the sequence of actions the user and the system take when the user is trying to find information on an individual at Rose-Hulman, whether student or faculty.

Basic Flow: We are going to assume that the user has the application open already, so the basic flow begins when the user goes to the search tab of the application.

1. The user will select the search bar.
2. The user will select the username option.
3. The user will type the username of the individual whose contact information they want.
4. The app will automatically use the stored Kerberos username and password to navigate to server and pull off all available information available from a ‘username’ on the website.
5. The user will then see a list of contact information for the individual.
    a. If the individual is a student, the information available will be username, email, advisor name, campus mail, and a link to the individual’s schedule.
    b. If the individual is a professor, the information available will be username, email, campus mail, office room number, office phone number, and a link to the individual’s schedule.

Alternative Flow of Events:

If the username entered in step 3 does not directly map to a single username (such as searching by a portion of the username):

- the list of potential matches will be listed
    - If there are no potential matches, the user will see an empty list and need to go back to basic flow step 3.
- The user will select the individual they desire information about.
- Move to basic flow step 5.

Post conditions: The user views the information for a specific user.

9. Mobile Student Lookup Use Case: E-mail a User

Revision History			
Date	Issue	Description	Author
10/2/11	1.0	Initial Creation of “E-mail a User” use case	Mark Vitale
10/3/11	1.1	Updated and edited Pre-conditions, Alternate Flows.	Brandon Knight

Relevant Feature(s): 8

Brief Description: This use case describes the sequence of actions the user and the system take when the user is trying to email a user.

Basic Flow:

1. The user will scroll to the individual’s email address in the list of contact information.
2. The user will select this item in the list.
3. The app will then pull up the native iOS mail client and have a blank new email ready to be composed, with the selected individual’s email address in the “to:” field.

Alternate Flow of Events: It is possible that the user of the app has not set up any email accounts on their iOS device. In this case, selecting the individual’s email address will navigate to the iOS native screen for creating a mail client or importing accounts.

Pre-conditions: We are going to assume that the user has an individual’s contact information already pulled up. For more information on how to get to this state, see the “Find Contact Information” use case. We also are assuming that the user has navigated to an individual’s contact information

Post conditions: The user either has a blank email addressed to the desired individual ready to be composed or the user is still on the contact information page if no email accounts have been set up in the native mail app.

10. Mobile Student Lookup Use Case: E-mail a Schedule

Revision History			
Date	Issue	Description	Author
10/02/11	1.0	Initial Creation of E-mail Schedule Use Case	Katie Greenwald
10/02/11	1.1	Added the alternate flow information	Ann Say
10/03/11	1.2	Edited, removed preconditions and alternate flow updated.	Brandon Knight

Relevant Feature(s): 13

Brief Description: User creates an email including the schedule they selected in the native iPhone e-mail app.

Basic Flow:

1. Use case begins when user chooses to send a selected schedule as an e-mail.
2. The system takes the user to the e-mail app.
3. The system opens a partially-composed e-mail containing the selected schedule as a simple text list.

Alternate Flow: If the user has not set up a mail client, or does not have any e-mail accounts programmed onto their iOS device, then selecting the e-mail will navigate to a the iOS native screen for creating a mail client or importing accounts.

Pre-conditions:

1. User has selected to e-mail a schedule.

Post conditions:

User is taken to the e-mail app with an e-mail containing the schedule they selected.

11. Mobile Student Lookup Use Case: Add to Favorites

Revision History			
Date	Issue	Description	Author
10/02/11	1.0	Initial Creation of “Add to Favorites “ use case	Ann Say
10/03/11	1.1	Updated and Alternate Flow edited.	Brandon Knight

Relevant Feature(s): 12

Brief Description: The user adds a specific person, such as a student or a professor, to a list, allowing him or her to quickly view his or her schedules.

Basic Flow:

1. The user starts the application and is taken to his or her schedule.
2. The user selects the search bar and searches by username.
3. The user sees the desired person’s schedule.
4. On the viewing screen, there is an option called “Add to Favorites” in which the person can select to add the user to the list.

Alternative Flow of Events: Server is not up, or it crashes after the person had already logged into the system, or the person is already in the user’s favorites list

If the person is already on the user’s favorites list:

- Do not add the user to the favorites list again.

Pre-conditions: The user has already used the application before, so the user’s username and password are already stored.

Post conditions: When the user views his or her favorites list, the new person is in the list.

12. Mobile Student Lookup Use Case: View Favorites

Revision History			
Date	Issue	Description	Author
10/02/11	1.0	Initial Creation of "View Favorites" use case.	Ann Say
10/03/11	1.1	Updated and edited a few sections.	Brandon Knight

Relevant Feature(s): 12

Brief Description: This use case describes the sequence of actions the user and the system take when the user wants to view his or her favorites list.

Basic Flow: The basic flow begins when the user is at the main page of the application and currently views his or her schedule.

1. The user will select the 'view favorites' option.
2. The user goes to a list of names of users they have saved in their favorites list.
3. Clicking on their user names will show that individual's schedule.

Alternative Flow of Events: The server may not be live, or the user may have an empty favorites list.

If the server cannot be reached:

- Display the most recent favorites list of the user.

If the user has an empty favorites list:

- A message is displayed saying that the user has no favorites added yet.

Post conditions: The user is viewing his or her favorites list.

13. Mobile Student Lookup Use Case: Sync Schedule

Revision History			
Date	Issue	Description	Author
10/02/11	1.0	Initial creation of “Sync Schedule“ Use Case	Katie Greenwald
10/03/11	1.1	Edited several segments.	Brandon Knight

Relevant Feature(s): 10

Brief Description: User chooses the option to sync their schedule from one particular quarter with their iPhone Calendar. Their schedule is automatically added for all the days that class is in session for that quarter.

Basic Flow:

1. Use case begins when user selects to add their schedule to his or her calendar.
2. The system shows the quarters that the user is able to sync with his or her calendar.
3. The user selects a quarter to sync with their calendar. [Alternative Flow: The user does not want to sync a quarter with their calendar].
4. The system adds instances of the users classes from the quarter selected to all the days on which there is class in that quarter.

Pre-conditions:

1. User has working login information.
2. The Schedule Lookup Page is functional and can be accessed.
3. User has selected to add their schedule to Calendar.

Post conditions:

User’s schedule for quarter selected is added to iPhone Calendar. If the user chooses to cancel, the system takes him or her back to the page they were on previously.

14. Mobile Student Lookup Use Case: View Class Roster

Revision History			
Date	Issue	Description	Author
10/07/11	1.0	Initial creation of “View Class Roster“ Use Case	Mark Vitale

Relevant Feature(s): 3

Brief Description: After a user has navigated to a course’s information page the user can view that course’s roster. Each of the individuals listed in the courses roster can then be selected for further information.

Basic Flow:

1. Use case begins when user is on the information page for a course.
2. The user presses the button on this page named “view roster.”
3. The system queries the database for the class roster and presents the user with a list of names of members of the course. The list of names on this page is clickable and will take the user to the individual information page for the name he or she selected.

Pre-conditions:

User has already navigated to a course information page.

Post conditions:

User views the roster for the class of interest.

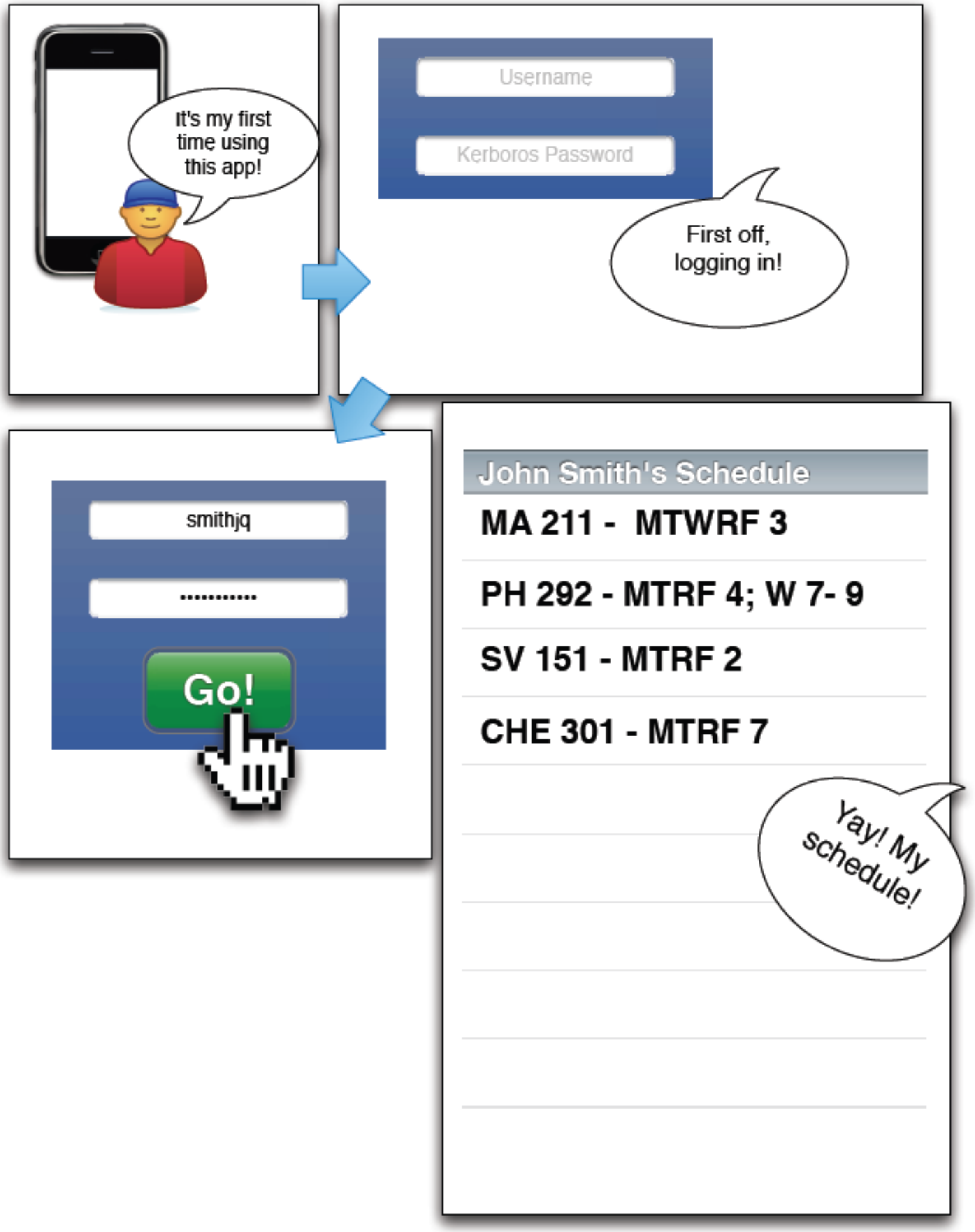
Use Case to Feature Mapping

This information is also provided in each use case under “Relevant Feature(s)” but is provided in table form here for convenience.

Use Case	Feature
1	2
2	4
3	1
4	9
5	6
6	5
7	11
8	7
9	8
10	13
11	12
12	12
13	10
14	3

Storyboards

First Time Use



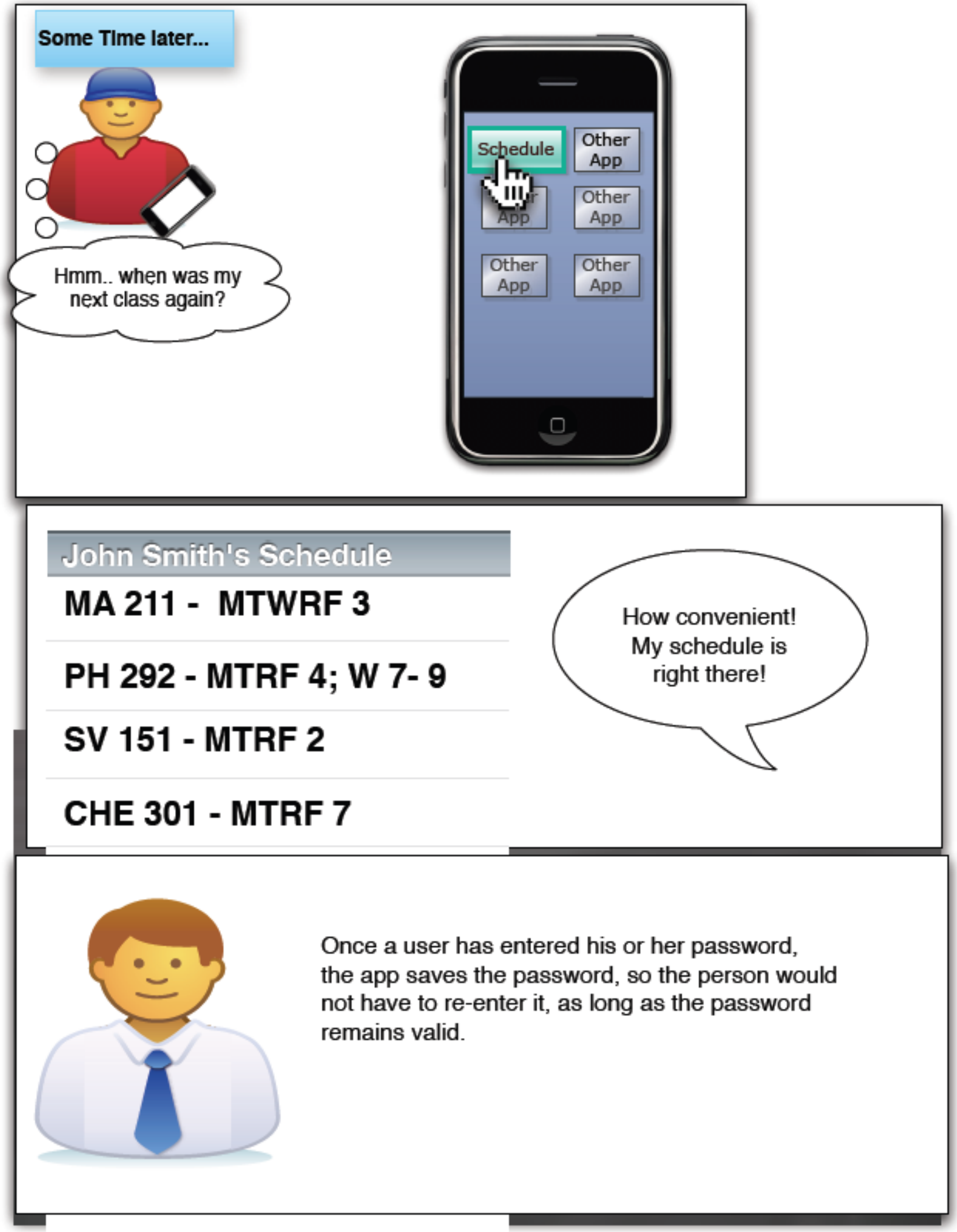
John Smith's Schedule

MA 211 - MTWRF 3

PH 292 - MTRF 4; W 7- 9

SV 151 - MTRF 2

CHE 301 - MTRF 7


Some Time later...
Hmm.. when was my next class again?
Schedule
Other App
Other App
Other App
Other App
Other App
How convenient! My schedule is right there!

John Smith's Schedule

MA 211 - MTWRF 3

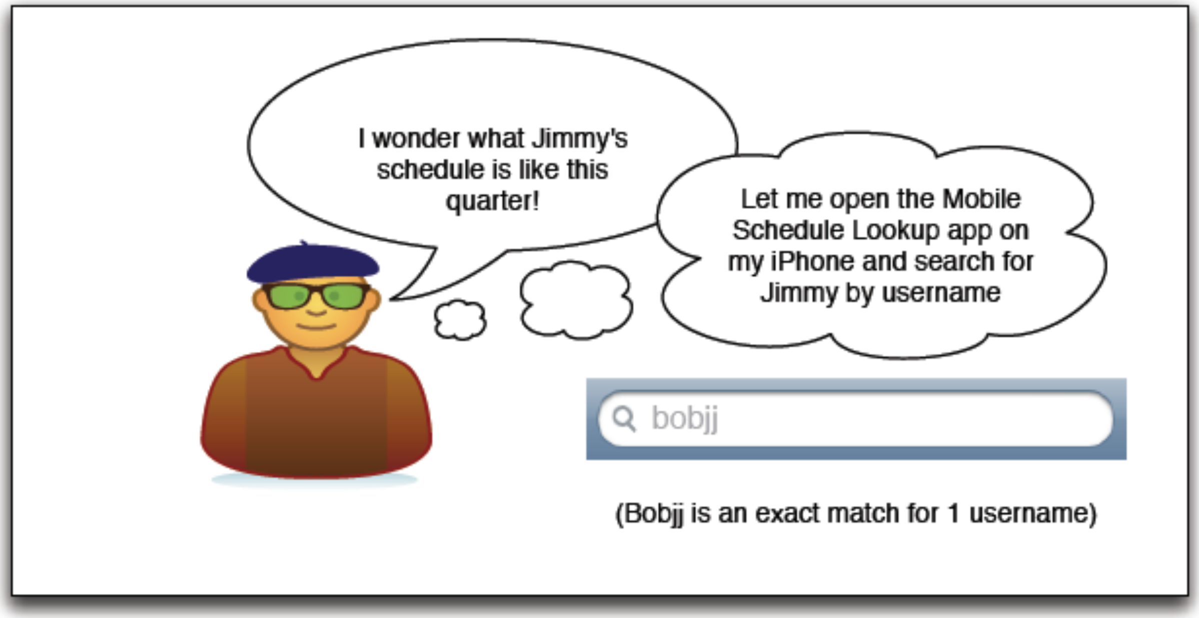
PH 292 - MTRF 4; W 7- 9

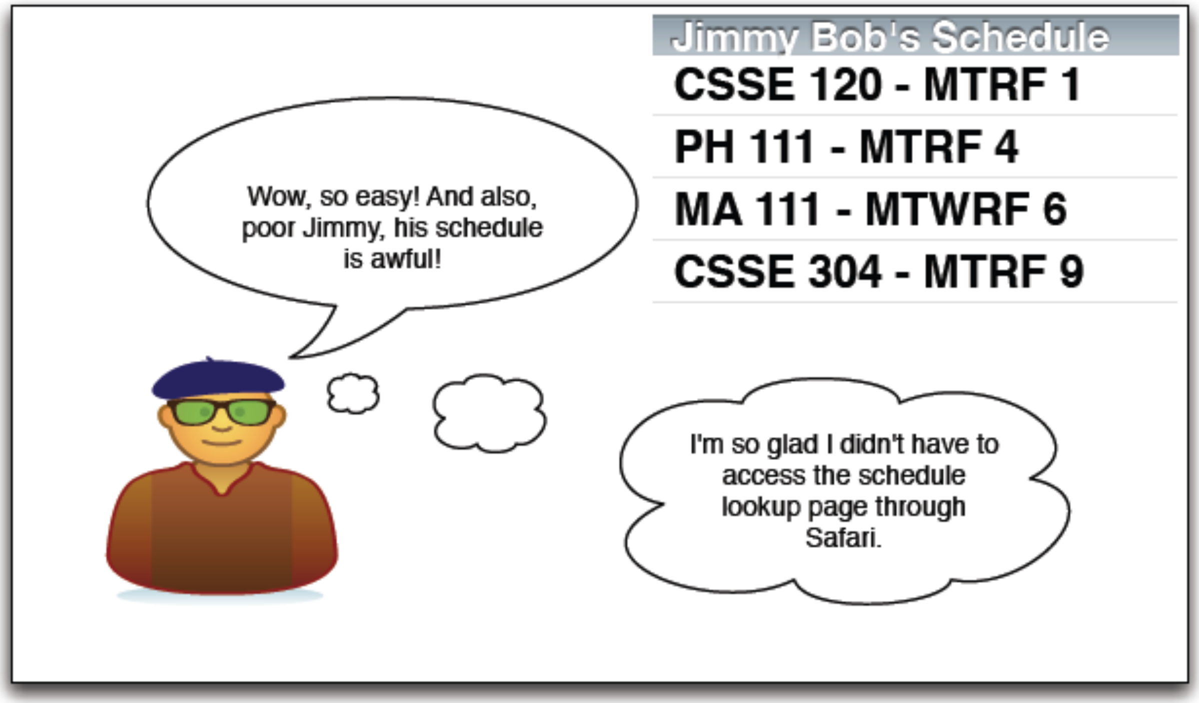
SV 151 - MTRF 2

CHE 301 - MTRF 7

Once a user has entered his or her password, the app saves the password, so the person would not have to re-enter it, as long as the password remains valid.

Looking Up a Friend


I wonder what Jimmy's schedule is like this quarter!
Let me open the Mobile Schedule Lookup app on my iPhone and search for Jimmy by username
bobjj
(Bobjj is an exact match for 1 username)


Jimmy Bob's Schedule

CSSE 120 - MTRF 1

PH 111 - MTRF 4

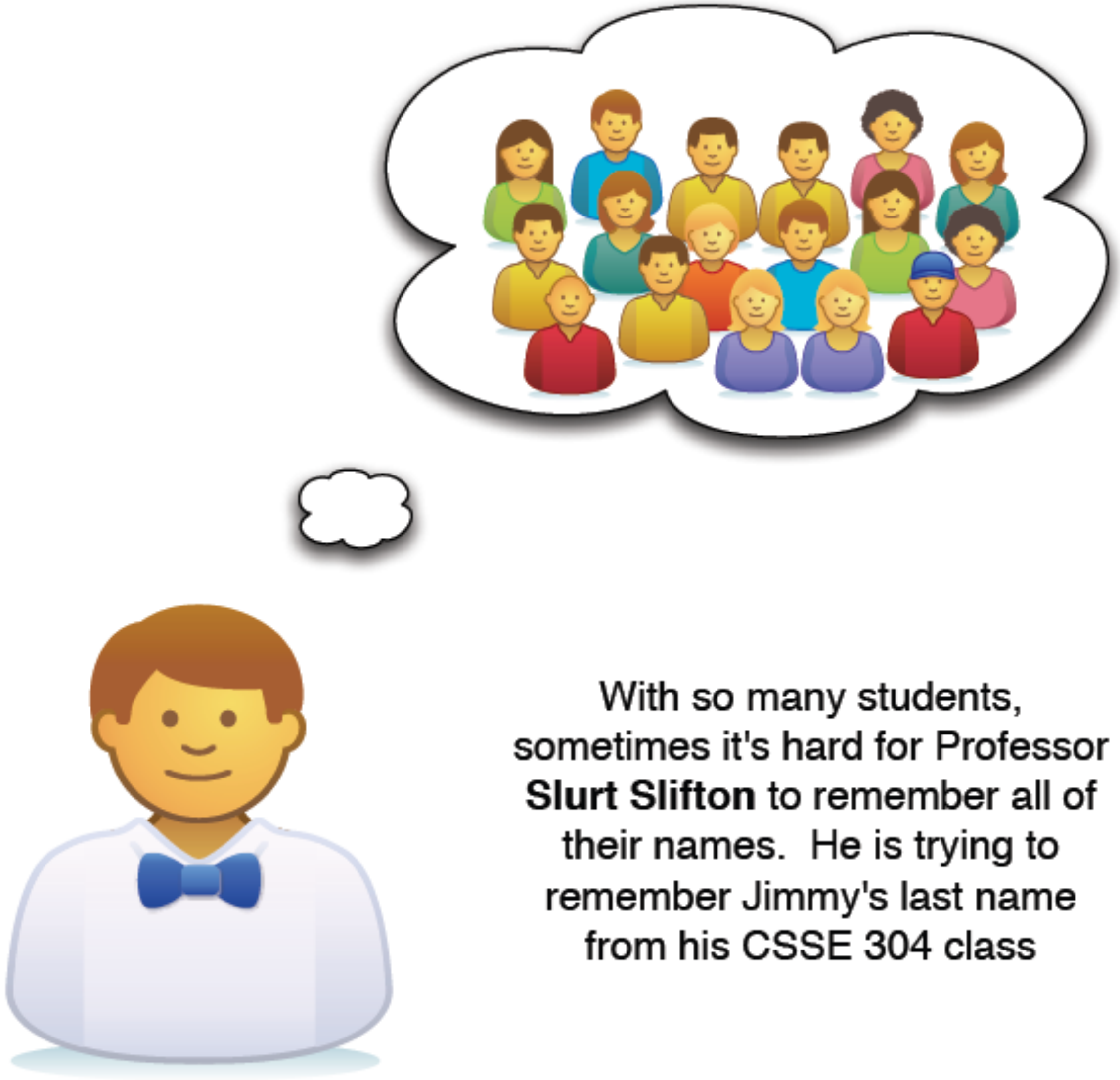
MA 111 - MTWRF 6

CSSE 304 - MTRF 9

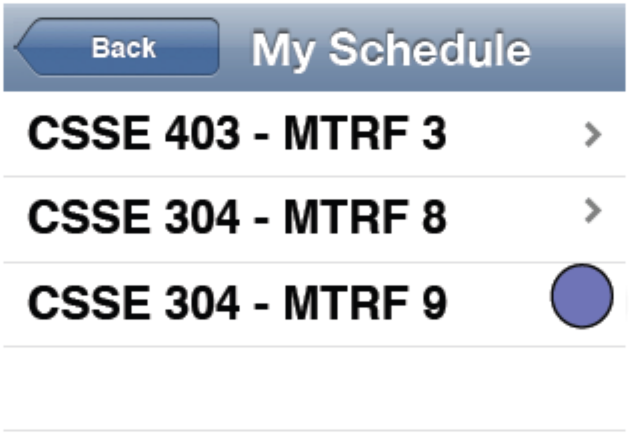
Wow, so easy! And also, poor Jimmy, his schedule is awful!
I'm so glad I didn't have to access the schedule lookup page through Safari.

Steve wants to look up his friend Jimmy's schedule while on the go. The mobile schedule lookup app makes it easy to access the information he wants in the timeframe he needs.

Class Roster and Sending Email



With so many students, sometimes it's hard for Professor **Slurt Slifton** to remember all of their names. He is trying to remember Jimmy's last name from his CSSE 304 class


Back

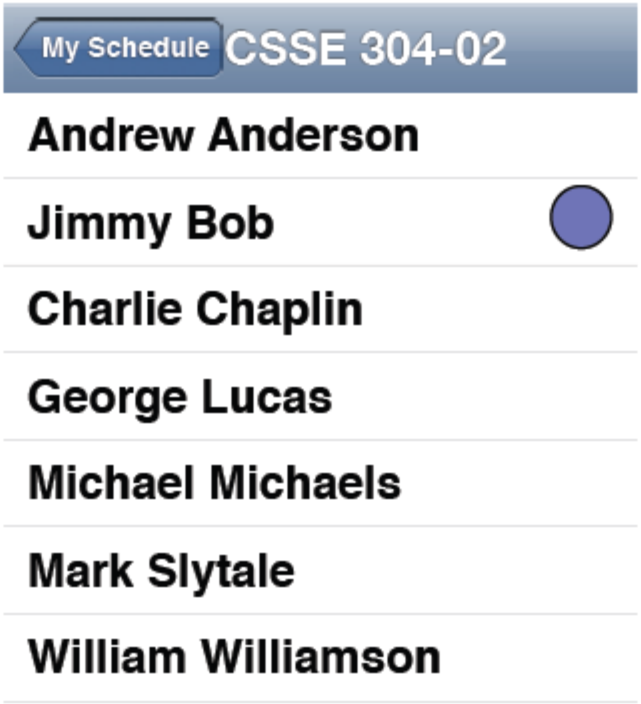
My Schedule

CSSE 403 - MTRF 3 ›

CSSE 304 - MTRF 8 ›

CSSE 304 - MTRF 9 ●

Oh right, his name is Jimmy Bob! How could I forget! I really need to send him an email.



CSSE 304-02

Andrew Anderson

Jimmy Bob ●

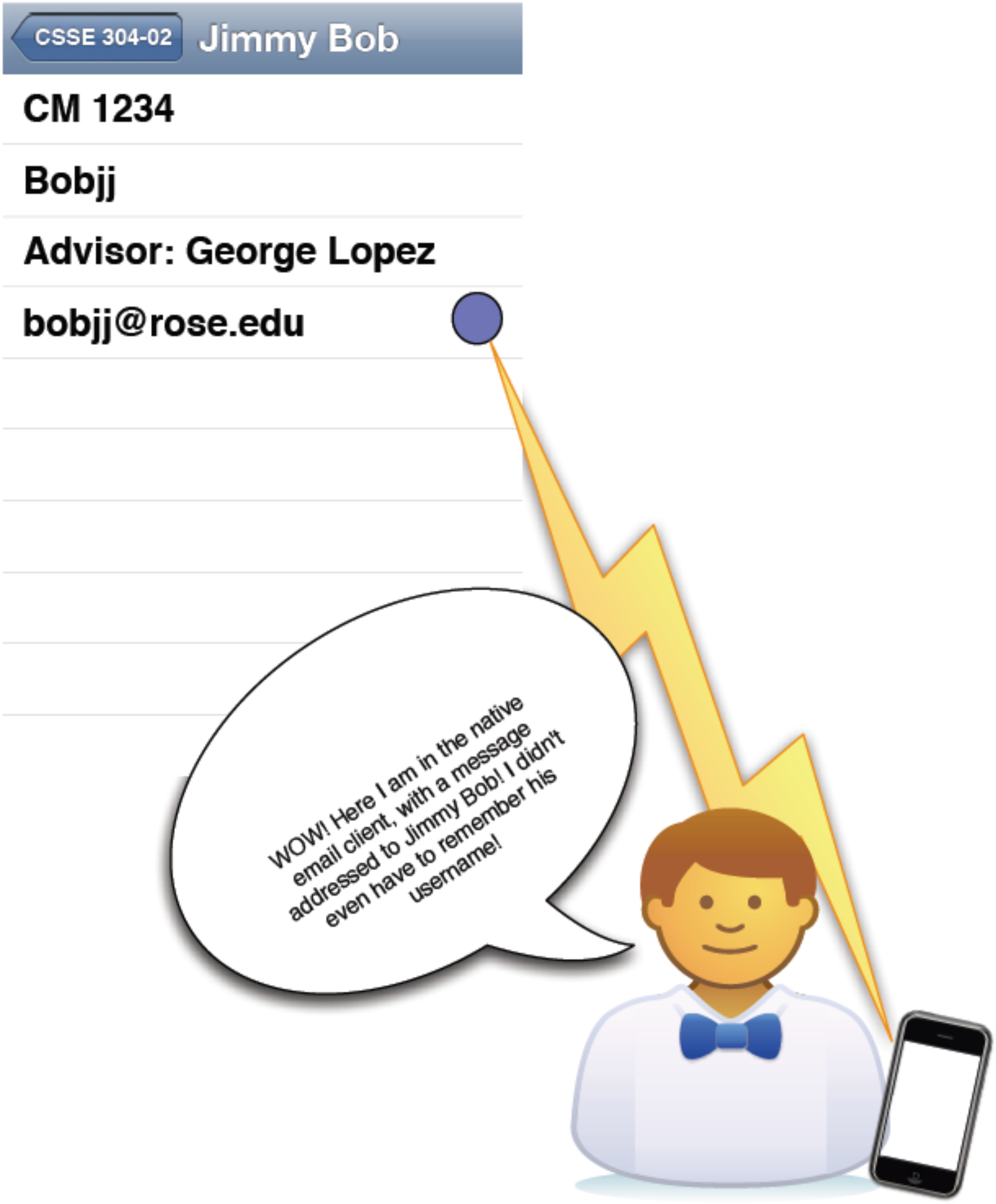
Charlie Chaplin

George Lucas

Michael Michaels

Mark Slytale

William Williamson



CSSE 304-02 Jimmy Bob

CM 1234

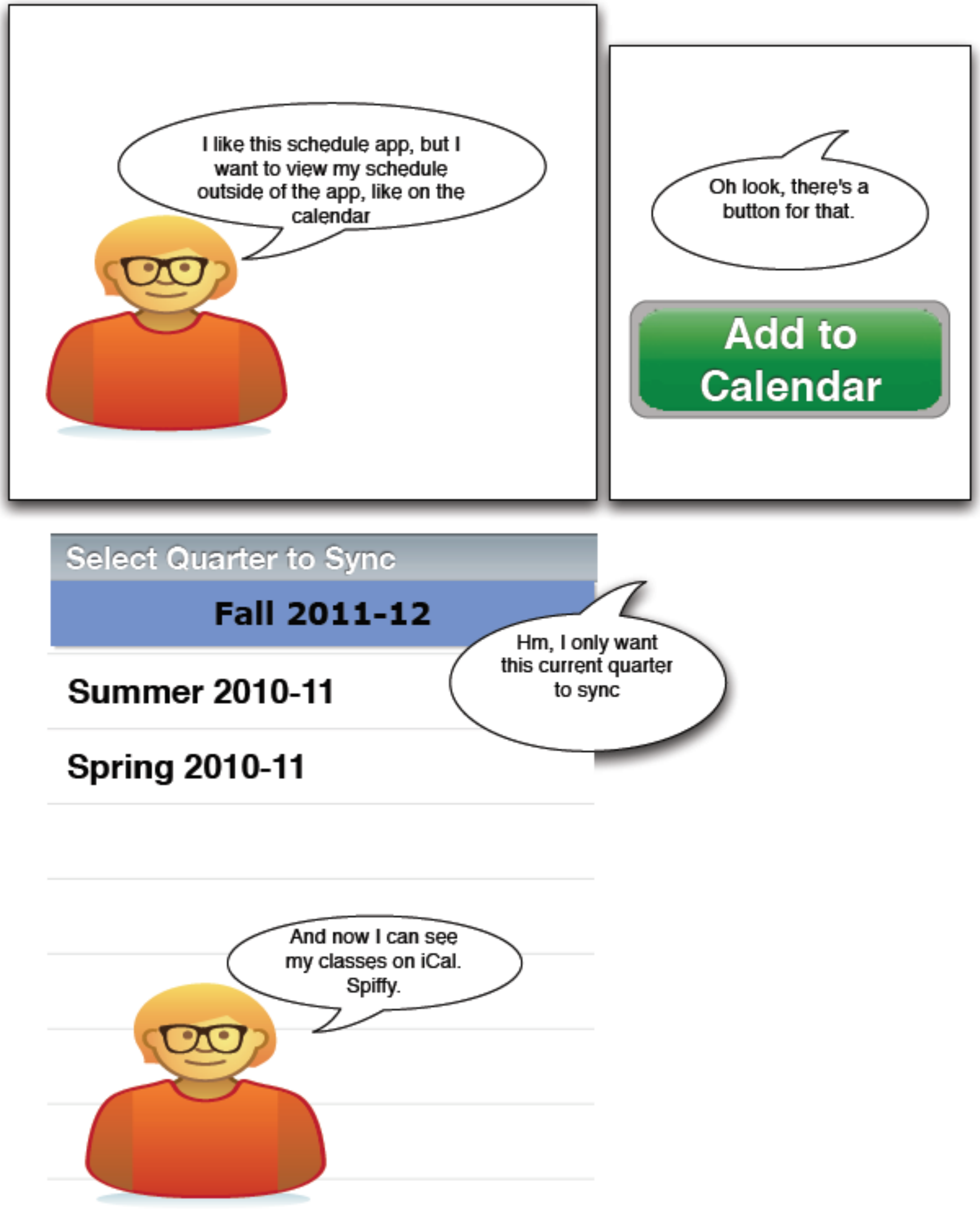
Bobjj

Advisor: George Lopez

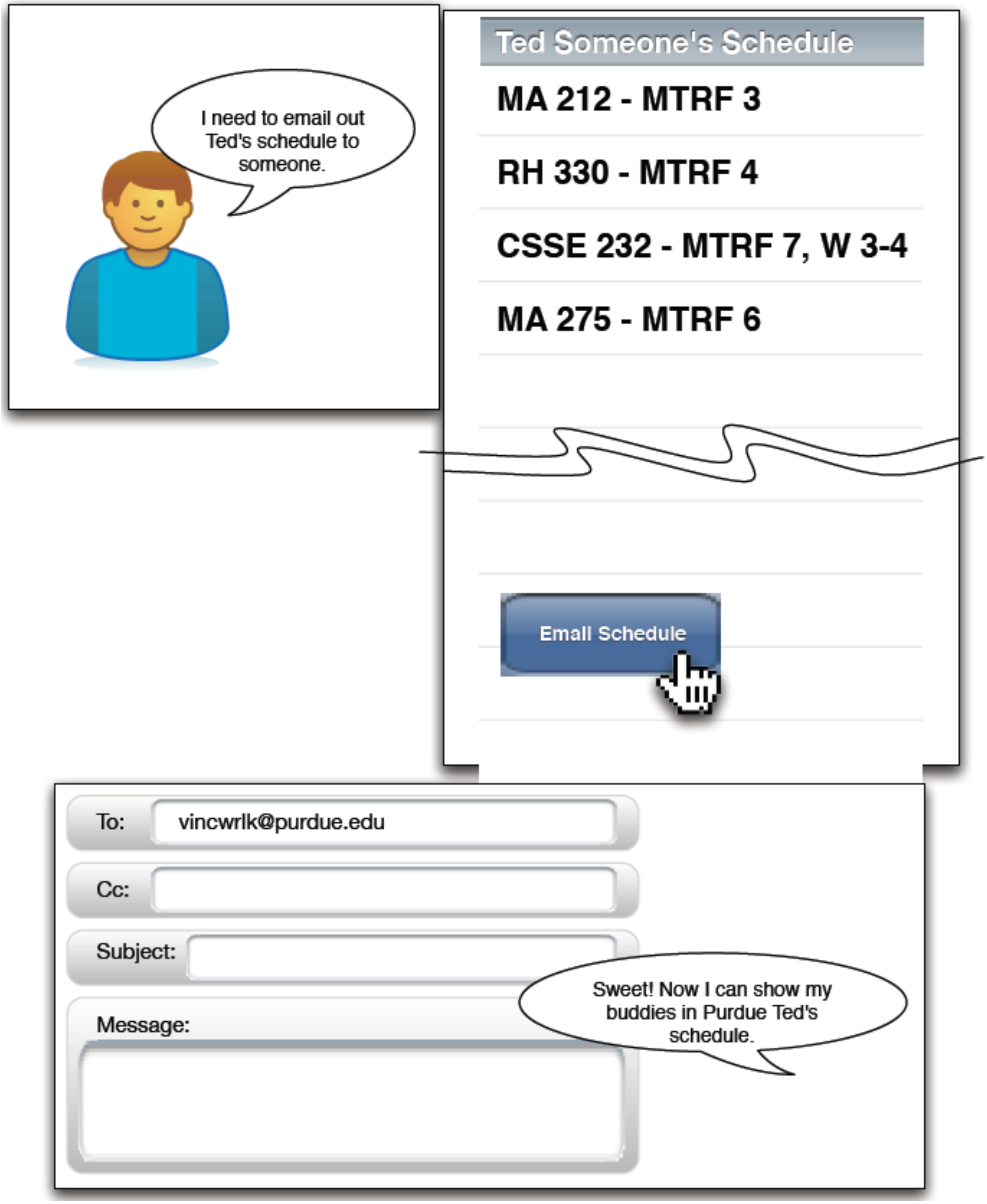
bobjj@rose.edu

WOW! Here I am in the native email client, with a message addressed to Jimmy Bob! I didn't even have to remember his username!

Calendar Sync


I like this schedule app, but I want to view my schedule outside of the app, like on the calendar
Oh look, there's a button for that.
Add to Calendar
Select Quarter to Sync
Fall 2011-12
Summer 2010-11
Spring 2010-11
Hm, I only want this current quarter to sync
And now I can see my classes on iCal. Spiffy.

Email Schedule



I need to email out Ted's schedule to someone.
Email Schedule
Sweet! Now I can show my buddies in Purdue Ted's schedule.

Ted Someone's Schedule

MA 212 - MTRF 3

RH 330 - MTRF 4

CSSE 232 - MTRF 7, W 3-4

MA 275 - MTRF 6

To: vincwrlk@purdue.edu

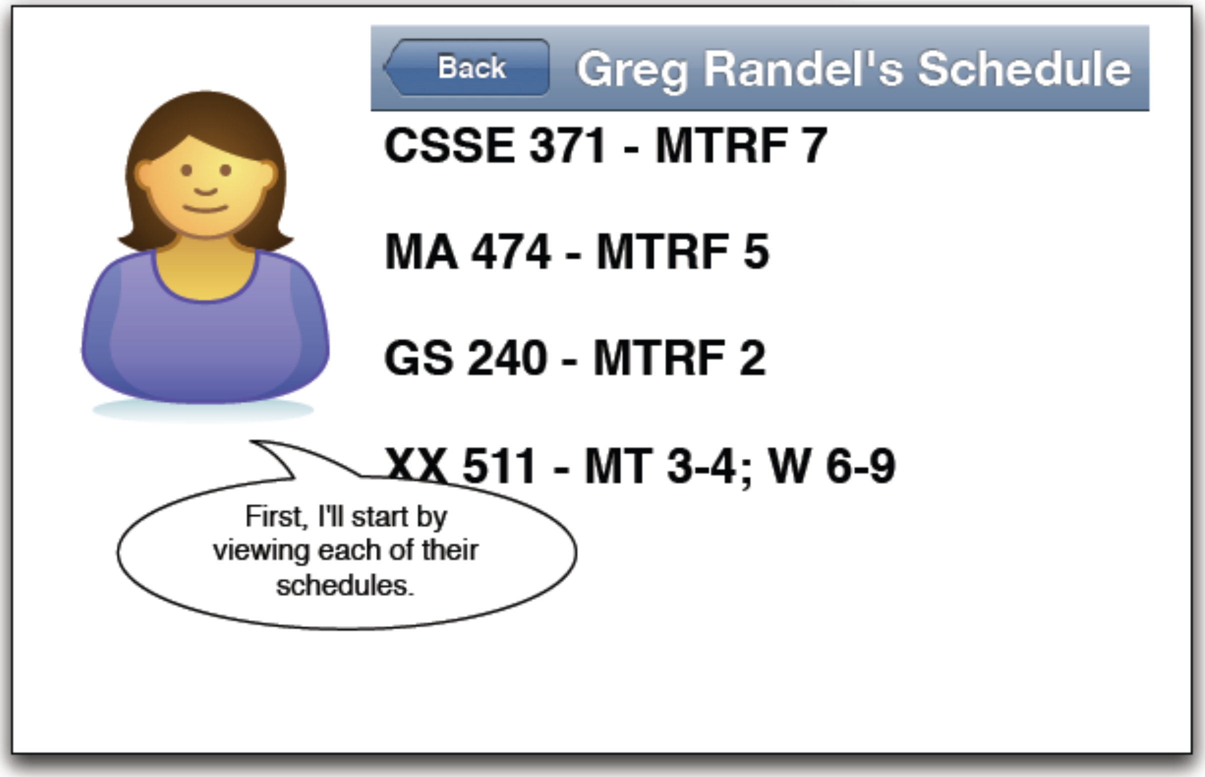
Cc:

Subject:

Message:

Overlay Schedules

We have a group project, but my team members and I have different schedules
When should we meet?



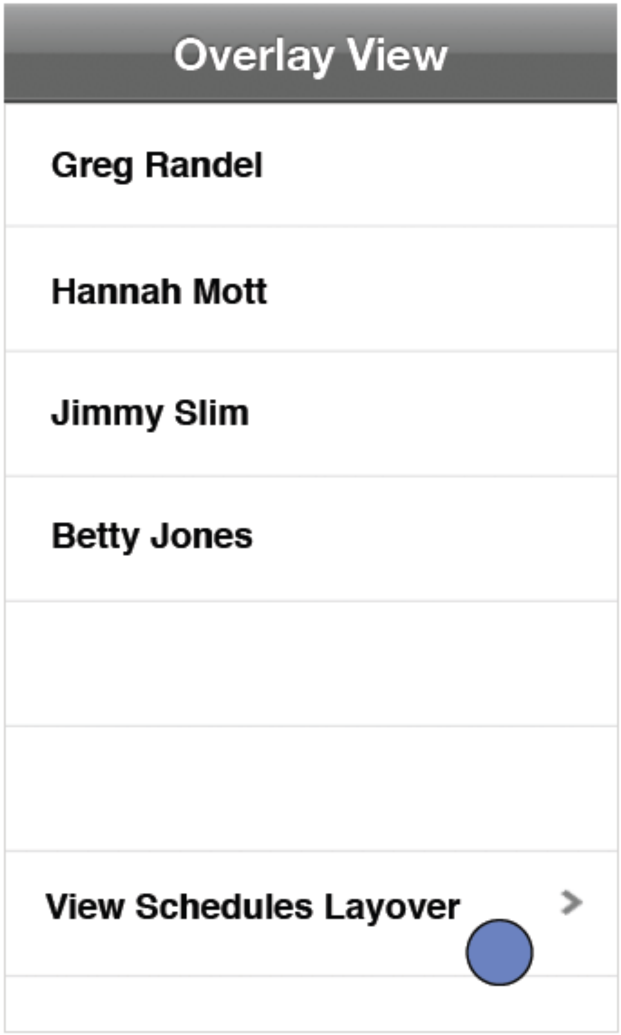
CSSE 371 - MTRF 7

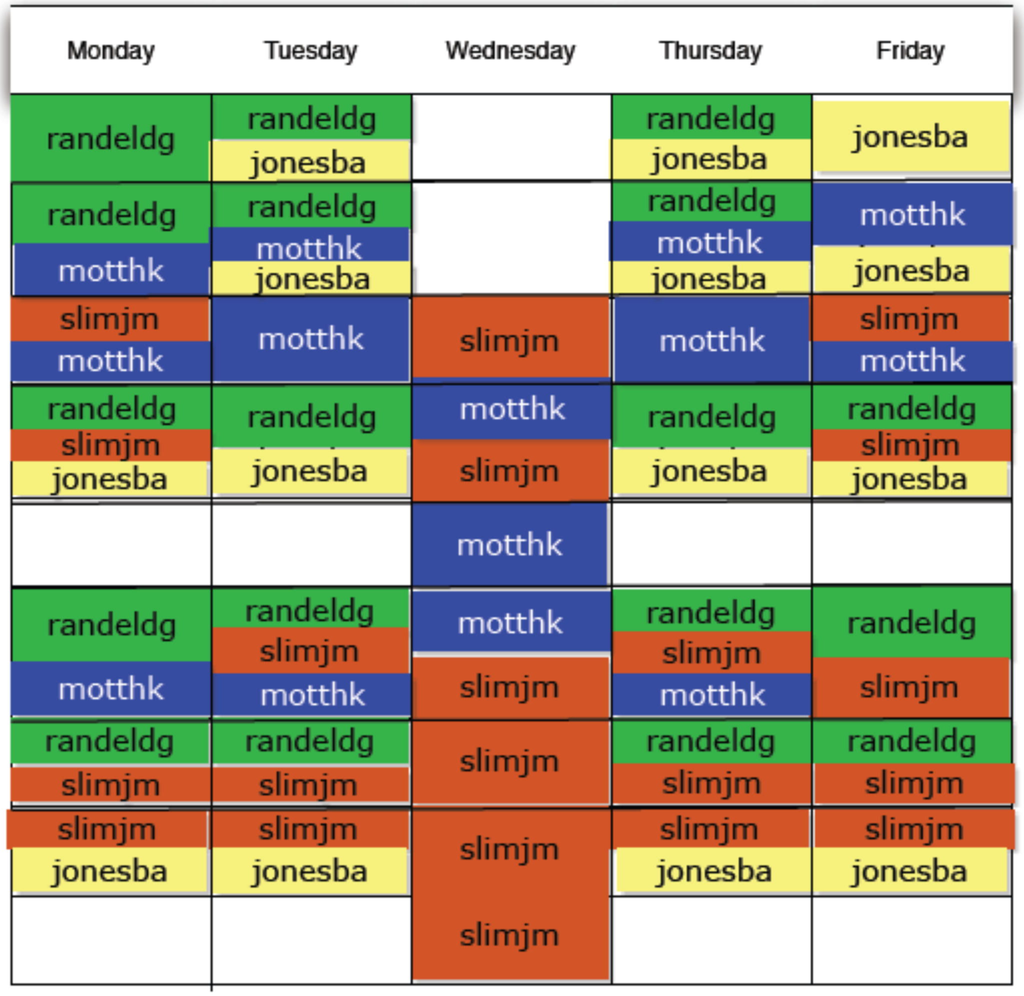
MA 474 - MTRF 5

GS 240 - MTRF 2

XX 511 - MT 3-4; W 6-9

Add to overlay
Then I'll each one to the overlay.

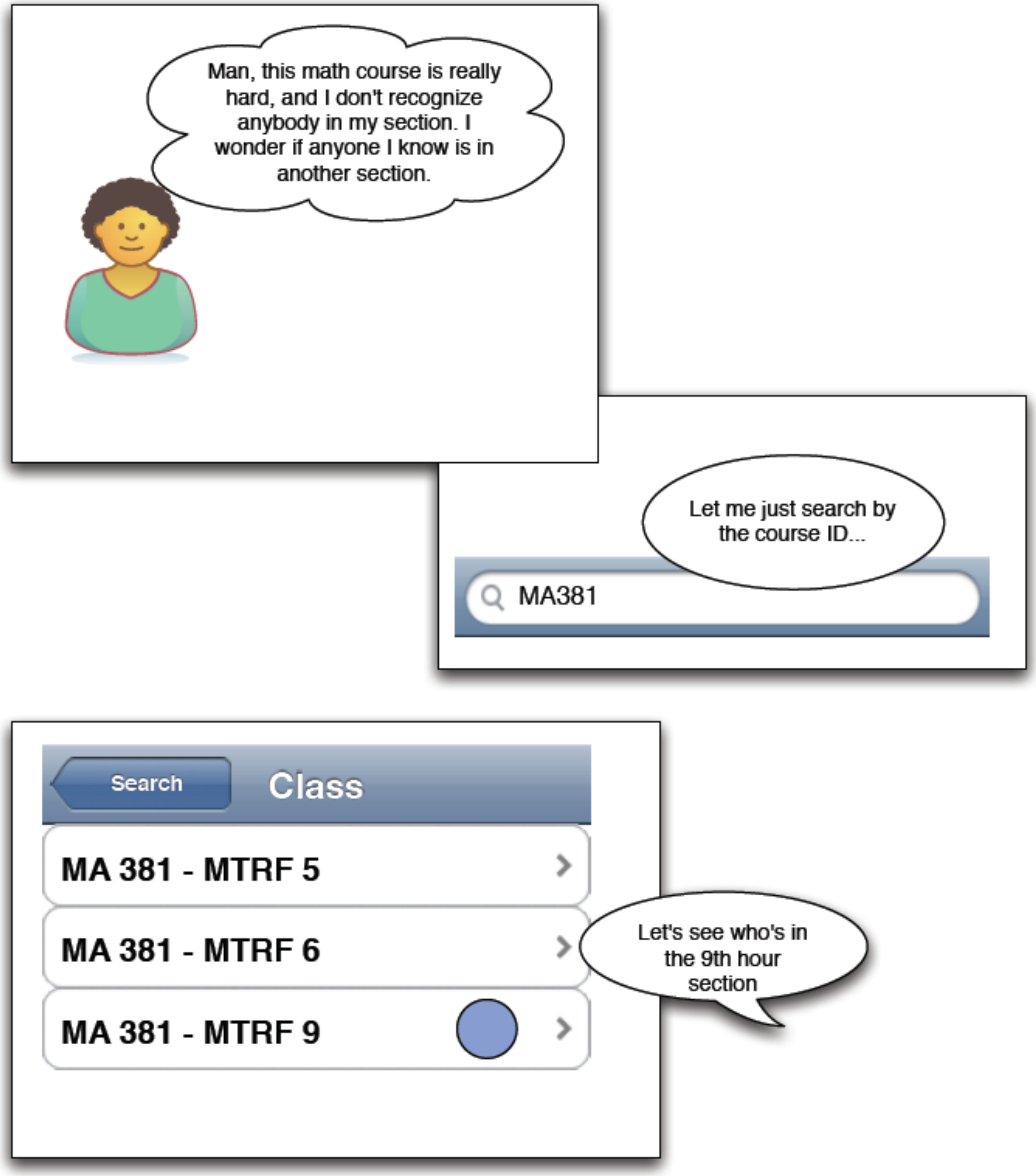

Overlay View
Greg Randel
Hannah Mott
Jimmy Slim
Betty Jones
View Schedules Layover

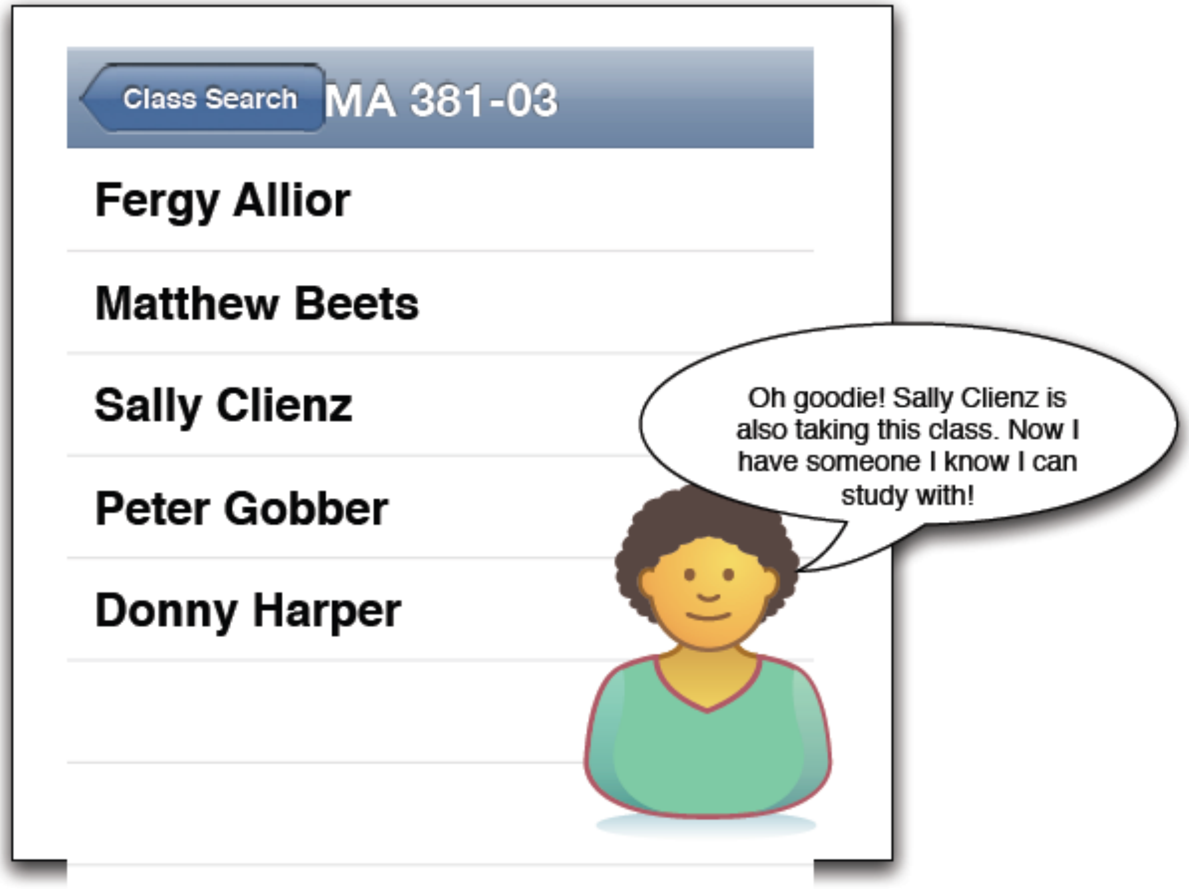


Monday	Tuesday	Wednesday	Thursday	Friday
randeldg	randeldg jonesba		randeldg jonesba	jonesba
randeldg motthk	randeldg motthk jonesba		randeldg motthk jonesba	motthk jonesba
slimjm motthk	motthk	slimjm	motthk	slimjm motthk
randeldg slimjm jonesba	randeldg jonesba	motthk slimjm	randeldg jonesba	randeldg slimjm jonesba
		motthk		
randeldg motthk	randeldg slimjm motthk	motthk slimjm	randeldg slimjm motthk	randeldg slimjm
randeldg slimjm	randeldg slimjm	slimjm	randeldg slimjm	randeldg slimjm
slimjm jonesba	slimjm jonesba	slimjm	slimjm jonesba	slimjm jonesba
		slimjm		

Look's like everyone is free 5th hour except for Wednesday. Why don't we meet then?
Sounds good

Search by course ID


Man, this math course is really hard, and I don't recognize anybody in my section. I wonder if anyone I know is in another section.
Let me just search by the course ID...
MA381
Search
Class
MA 381 - MTRF 5
MA 381 - MTRF 6
MA 381 - MTRF 9
Let's see who's in the 9th hour section



MA 381-03

Fergy Allior

Matthew Beets

Sally Clienz

Peter Gobber

Donny Harper

Oh goodie! Sally Clienz is also taking this class. Now I have someone I know I can study with!

Index

Glossary

iOS	The mobile operating system developed by Apple that runs on iPhone, iPod Touch, and iPad devices
Rose-Hulman	The number one undergraduate engineering school in the nation, and the school at which the schedule lookup app will be used
Use Case	A description of the interactions between a user and the system that leads to a useful outcome for the user

References:

iOS Developer Library Human Interface Guidelines. (2011). Retrieved from Apple: Apple Inc. (2011) Human Interface Guidelines https://developer.apple.com/library/ios/#documentation/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html

(2011). iPhone Mobile Lookup Survey. (K. Greenwald, Interviewer)

Apple Inc. (2011). *iOS 5 for Developers*. Retrieved October 11, 2011, from www.apple.com: http://developer.apple.com/technologies/ios5/

Ekl, T., & Stokes, E. (2011, 9 21). Client Meeting. (M. Vitale, B. Knight, K. Greenwald, & A. Say, Interviewers)

Yourdon, D. L. (2003). *Managing Software Requirements, 2nd edition, A Use Case Approach.* Addison Wesley: Pearson Education.