

Vicino

By

Alois Barreras

Submitted to
the Faculty of the School of Information Technology
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Technology

© Copyright 2015 Alois Barreras

The author grants to the School of Information Technology permission
to reproduce and distribute copies of this document in whole or in part.



Alois Barreras

4-30-2015

Date



Xuetao Wei, Faculty Advisor

4-30-2015

Date

University of Cincinnati
College of
Education, Criminal Justice, and Human Services

April 2015

Table of Contents

Acknowledgements	1
Abstract	2
Introduction.....	3
Problem Statement.....	3
Solution.....	4
Interface Experience and User Profile	5
Use Case	6
Use Case Figure 1.0	6
Technologies.....	6
Budget	8
Proposed Budget Table Figure 2.0	8
Timeline	8
Project Timeline Figure 3.0	8
Strategic Plan.....	9
Works Cited.....	10

Acknowledgements

I would like to thank all the employees at the University of Cincinnati App Lab for providing me with assistance and knowledge of building iOS applications. Furthermore, I deeply thank all team members of the raywenderlich.com website for providing amazing knowledge and in depth tutorial about iOS development. I would like also to thank Professors Kumpf and Scott for their valuable input and guidance throughout the course of this project.

Vicino

Abstract

Vicino is a mobile application for iOS devices. Developed using the Swift programming language and featuring a sleek, intuitive interface, a student can use the mobile application to broadcast for what class, assignment, and professor he is studying. The application uses GPS to determine the user's location and displays a view on a map in that contains information about the study group. Other students can then view the map and search for anyone who might be working on the same thing and join the study group.

Introduction

A problem facing many students today is the need to find other students with which to study. Aside from sending an email through Blackboard, there is no convenient way for a student to communicate with people in his class to get help on homework and assignments. Vicino is a mobile application for iOS devices designed to fill this need.

Problem Statement

Many students coming to a university can be overwhelmed by the large classes and new, challenging concepts. For example, a student may be struggling in a Calculus class with 200 other students but not know anyone else in the class. Currently, the only way to reach out to other students is to send a mass email to everyone in the class through Blackboard. Emails have a very low response rate, and even if another student does respond, they need to take extra steps to find an agreed upon meeting place and time. Given the reluctance to give out phone numbers to strangers, these exchanges occur solely through email, making the process slow and inconvenient.

Another option students have to get help is to meet with the professor outside of class individually. However, in classes with a large number of students, professors simply do not have enough time to meet with each student and address their issues. Furthermore, a limited window of office hours and adjunct professors not having offices on campus cause professors to resort to solving problems over emails. However, any student will admit professors are slow to respond to emails. More so, when professors do respond, emails alone are not nearly as effective as having an in-person meeting.

Solution

I solved these problems by creating a mobile application that streamlines the process of students finding others with which to study. A majority of mobile devices in the marketplace run the iOS operating system, so I am creating the Vicino application for iOS devices. Using the application, a student will be able to broadcast his GPS location, along with other details including the subject and professor of the class for which he is studying. If another student is also going to be studying for a particular class, he can use the application to search for someone else working on the same subject material. Once a study group is found, the student can request to join it. If accepted, the student will receive directions and be able to chat with the other members in the study group.

Any student can use the application and find other students for free. However, there will be a paid option for users to meet with a certified member who is guaranteed to know the subject material as well. Any student can register to be a certified member, and once certified, they can use the application like a standard user. However, their study group will be displayed with a special icon to notify other students that he is a certified member. Then, any student can join the study group with the certified member for a fee.

My solution will make it very easy for students to reach out to one another and will encourage them to collaborate with each other studying related subject material. This increased collaboration will improve grades, understanding of the material, and innovation within a university, resulting in a better overall student program and a higher quality of students. The application will also relieve some burden on professors. Many professors will meet with several students a week, often to address the same problem for each. This consumes a lot of their time, which could be better spent elsewhere. With Vicino, students will be able to get help from each other as well, allowing professors to spend more time grading and preparing for lectures.

Interface Experience and User Profile

Interface:

Vicino was designed following the guidelines laid out in the *iOS Human Interface Guidelines* from Apple. This will create an interface to which iOS device owners are accustomed, so they can very quickly learn how to navigate the application.

The main component of the application features a Tab Bar Controller that will feature three main View Controllers: an MKMapView displaying MKMapAnnotationViews that represent the study groups within a certain distance of the user's location, a Table View in which the user can search for specific study groups and narrow down the list of results, and a Table View with several other options including a profile settings page, log out button, and other miscellaneous options the user can customize.

Users:

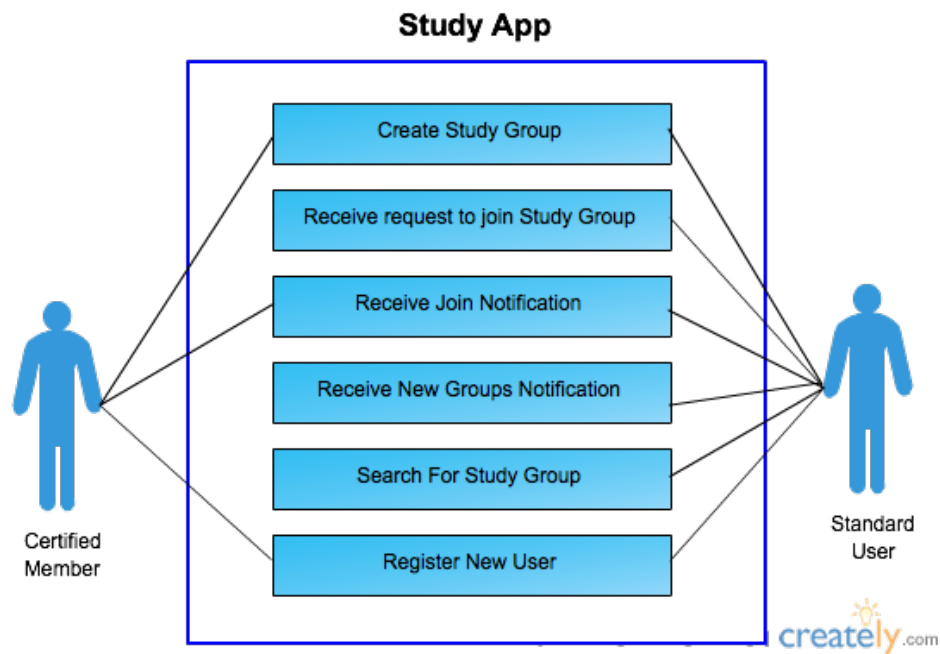
The main users of this application include all students attending a university with an iOS device.

The user will not need any formal training to use the application, but will require a familiarity with iOS interfaces. In-app tutorials will be also provided to ensure users will understand the application.

The other user for the application is the certified member, and is a person who is guaranteed to know the subject material. The interface for these users is the same as a standard user; the only difference is that a study group hosted by a registered member will be displayed with a different icon on the map.

Use Case

The use case for this application is designed to be as simple and easy to use as possible. A user registers his account, then can either create a study group or search for one to join. The user can receive various notifications as a result of other events as well. The use case for this application is illustrated in the following Figure 1.0.



Use Case Figure 1.0

Technologies

I used the following technologies during the development of Vicino:

Swift:

Swift is an exciting new programming language from Apple. It improves upon Apple's traditional programming language, Objective-C, and offers improved speed and safety. Swift uses an easy to use syntax and allows developers to use the existing Cocoa frameworks to create iOS applications.

Xcode 6:

Xcode is an IDE provided by Apple. It comes packaged with the Swift programming language and is required to develop iOS applications.

Parse:

Parse is a cloud-based database and user management system. It is owned by Facebook, and offers easy to use tools and APIs that allow developers to interact with the system. Parse provides an iOS framework, which I integrated into my application and allows me to communicate with the server. Parse also provides Push Notification functionality so I can engage my users and increase retention.

JavaScript:

The Parse service has a cloud-code functionality, which allows me to run custom code on the server. Using this feature, I can perform custom actions to objects on save, delete, update, as well as create background jobs on the server to clean up objects. All cloud code is written in JavaScript

JSON:

All data on the Parse server is stored as JavaScript Object Notation. I used this technology while sending data to Parse and interacting with Push Notifications.

Budget

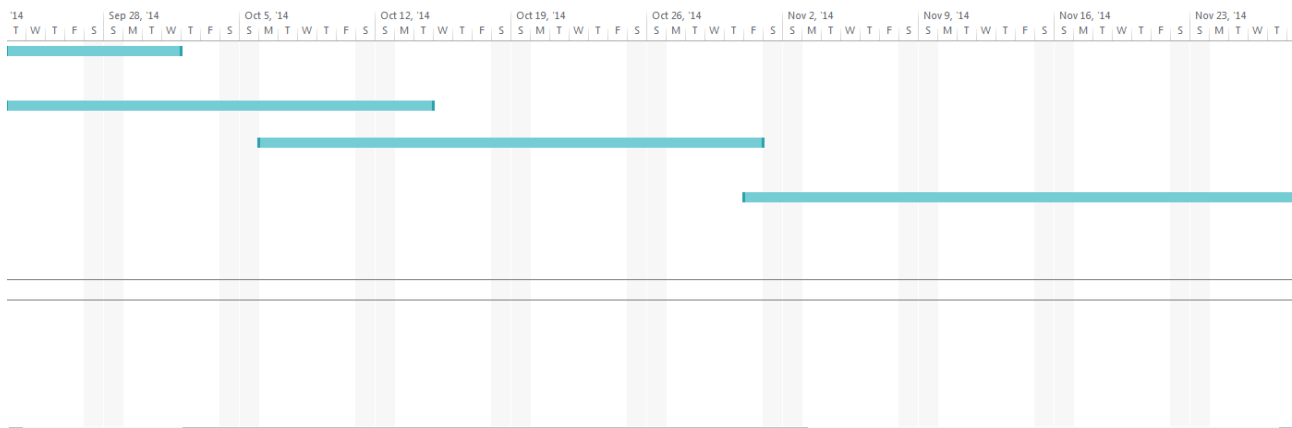
My budget for this project is approximately \$800. The main cost is from purchasing the devices on which to test the application. Various legal fees will be determined at a later date.

Item	Price
Apple iPad Mini	\$399
Apple iPhone 6 Plus	\$299
Apple Developer Program Membership	\$99/yr
Incorporation/Legal Fees	TBD
Total	\$799

Proposed Budget Table Figure 2.0

Timeline

A Gant Chart shown in Figure 2.0 represents the development timeline for Vicino. The timeline incorporates the deadlines for submitting to the Apple App Store and the University of Cincinnati IT Expo. The expected release date is March 2015.



Project Timeline Figure 3.0

The project will consist of 4 main development phases.

1. Functionality for users to login

2. Functionality for users to post a study session
3. Functionality for users to search for study sessions
4. Functionality for users to join study sessions.

Conclusion / Strategic Plan

The Parse service is initially free, but once an application exceeds a certain number of server requests per second, there is a monthly fee. In order to avoid this, I will only initially make the application available to University of Cincinnati students. I will achieve this by making sure users have a valid UC email address in order to register. The application is built with the ability to easily expand in mind. After market interest is gauged, I will consider opening the application to other universities pending there is enough income generated for me to pay for the upgraded Parse service to handle the increased number of server requests.

Works Cited

"IOS Human Interface Guidelines: Designing for IOS." *IOS Human Interface Guidelines: Designing for IOS*. N.p., n.d. Web. 10 Oct. 2014.

"Ray Wenderlich." *Ray Wenderlich*. N.p., n.d. Web. 10 Nov. 2014.

"The Swift Programming Language: The Basics." *The Swift Programming Language: The Basics*. N.p., n.d. Web. 10 Nov. 2014.