Deborah Newberry
27 September 2016

Senior Project Proposal - Iota Pi Meeting Organizational App

Project Overview

For my senior project, I'll be creating a meeting organization app for my school's chapter of Kappa Kappa Psi, Iota Pi. It will keep track of attendance, have a calendar of events, serve as an efficient way to anonymously vote on chapter actions, allow users to update chapter inventory, be able to send out announcements, and have more features as requested by the brothers.

Background

There are a few specific meeting-type apps that exist in the app store:

- Meeting Pad: a calendar that schedules meetings and has the additional functionality of addings notes to people's names and taking minutes. There are a few more features, but all of them are very specific to a short business meeting situation.
- TimeStation: keeps track of attendance. It uses employee badges or credentials to allow them to "clock in" to and "clock out" of the meeting.
- Google Sheets (and other spreadsheet apps): a spreadsheet is what we've been using to take inventory. I would like to integrate Google Sheets somehow so I can link inventory changes to the spreadsheet to update it.
- Poll Everywhere: lets people vote anonymously. I've seen it used in a classroom, and it would be nice to also integrate this into the app.

In short, there are many generic apps for aspects of a meeting, but I believe Iota Pi would benefit from having a single app that has all of these features.

I hope to have a login system that an administrator (the president or anyone in the chapter's executive council) could accept members into. A @calpoly or @iotapi email would be required to login, and the executive council (as determined by their @iotapi email) would be able to have more power over features like calendar editing, announcements, and attendance. I would use Firebase as my database management system in order to keep essential information for the app, and I would attempt to integrate Google Calendar (and possibly Sheets and Poll Everywhere) for different sections of the app.

System and User Requirements

This system will make the normal process of Iota Pi meeting simpler and quicker by cutting down the hassle and time necessary to vote, take attendance, and other monotonous things.

The app will be developed for the iPhone. David Smith (https://goo.gl/xxVDIR) claims that over 80% of iPhone users use version 9 or higher; the Apple App Store will enforce this. It'll be developed using Swift in XCode (current version 7.2) on my personal MacBook Pro (current OS 10.11.4). I'll be using a private repository on GitHub in order to successfully back-up my code while also being able to share it with my advisor, and I'll be using InVision to create a Horizontal Prototype (see below).

Experiments

In order to confidently build an app that is both functional and easy to use, it will be pertinent to have users test it. I intend to create a survey (using Google Forms) at four different points in the project: at the beginning of the quarter to receive input on app features, at the end of my Horizontal Prototype, halfway through the final product, and at the end of the process. Questions for the latter three will be centered around how effective the user thinks the app is at its main function, how instinctive it is to use, and if there are any suggestions they have to improve either functionality or visual appeal. During the incremental phases, I can use this information to shift direction and at the end I will compile

all the surveys into my final report. The final questionnaire results can be used when I do a retrospective look at the process I took and implementation decisions I made.

Schedule

Vertical Prototype

The Vertical Prototype should be focused on testing and implementing basic working versions of APIs needed to construct the final application. This includes database libraries, any login system the client decides to use, image-to-text libraries (Optical character recognition), and methods of extracting useful information from text.

Horizontal Prototype

The Horizontal Prototype should be focused purely on the UI necessary to build the app. Every single screen the user can see should be mocked up in as much detail as possible. I'll probably use InVision (https://www.invisionapp.com/) with GIMP (https://www.gimp.org/) to create this so that the client can see the process and comment along the way.

CPE 484: User-Centered Interface Design and Development

This fall, I'm taking CPE 484 as a technical elective. I'm hoping to use knowledge and techniques gained from this class to create and improve upon the Horizontal Prototype for the application (and thus, the UI for the final app).

Fall Quarter 2016

Dates	***Milestone***
Week 1 (Sept 22 - Sept 23) - Week 6 (Oct 24 - Oct 28)	Write app proposal. Brainstorm features, ask for ideas. **Finish Vertical Prototype by October 31**.
Week 7 (Oct 31 - Nov 6) - Week 11 (Nov 5 - Dec 11)	Start work on Horizontal Prototype. Receive feedback from vertical prototype. **Finish Horizontal Prototype by December 12.**
Finals Week (Dec 12 - Dec 18)	Start work on main application.

Winter Break 2016 (Dec 19 - Jan 9)

- Work on main application.

Winter Quarter 2017 (January 10 - March 24)

- **January** - Finish ~40% functionality with bare-bones UI application. If possible, being simple user testing. Collect data on ease of use, desired functionality, and pleasantness of UI.
- **February** - Finish ~80 % functionality with mostly completed UI application. If possible, being intermediate user testing. Collect data on ease of use, desired functionality, and pleasantness of UI.
- **March** - Finish 100% functionality with complete UI application. If possible, finish complete user testing. Collect data on ease of use, desired functionality, and pleasantness of UI. Publish into the App Store. Write final paper.