Software Project Plan

for

CSU Campus Assistant

prepared by

Group 2

Kevin Dowling

Taylor Ehrhart

Corey Greer

Drew Polhamus

04/30/15

# **Abstract**

The CSU Campus Assistant is an app, available for Android, that will provide users with valuable information about the campus through their smartphone. The different types of information can be accessed in a variety of ways. Users will be able to view a list of all of the facilities on campus, and see what the facilities are used for, and how their fellow students rank the facility. Directions to a location on campus can be provided to the user through Get Me There feature. Get Me There will provide dynamic instructions, in the form of a compass and distance meter, to route the user to their destination. If the user simply wants to view the campus, then they can open the campus map feature. Through the use of these tools, no student should be lost on campus.

# **Introduction**

If you look up Columbus State University on Google maps, the only information provided is about the Campus as a whole. There is no information about individual buildings or other landmarks. This app will provide students with the ability to get to know the campus, and become comfortable in the learning environment that surrounds them.

A typical problem that incoming freshmen face, is getting to know the ins and outs of their new campus. Orientation will showcase some of the more important locations on campus, but what about comfortable places to study, or quiet places to sleep? That is where the Campus Assistant comes in. Through an amenity rating feature, users will be able to see what the student body thinks about the different facilities’ amenities. These combined features will provide students with everything they need to quickly assimilate into the college life.

# **Specifications and Features**

The CSU Campus assistant will be comprised of several features, each of which contributes to the overall goal of familiarizing the user with Main Campus. This section is a list of those features, along with detailed descriptions of each one.

## Features

* **The Facility List,** is simply a list of all the facilities on main campus that the user can learn more about. Minor information may be displayed about the facility, depending on the space allowance, and visibility of text.
* **The Facility Information** pages, are where the majority of the information will be displayed. It is here, that the user will find helpful ratings to give them an idea of which building will suit their needs. Users can also find out about the unique things that each building has to offer, such as; computer labs, recreation areas, and eating options.
* In the event that someone is vaguely familiar with campus and just needs their mind refreshed, a helpful **Map** will be provided by the campus assistant. This map can be zoomed in on for better visibility, and may be interactive. This feature will be an alternative to the Get Me There, for users that do not wish to drain their battery.
* The **Main Menu** will be short, and sweet. No need for excessive clutter to confuse the user, just quick access to the features provided by the Campus Assistant. This access will be in the form of buttons that will take the user to the feature of their choice.
* **Get Me There** will be the user’s camera feed, with a compass that points toward their destination, and a distance counter displayed over it. Accessed from an individual facility information page, the user can navigate their way across campus, and find their destination. Using GPS location, the compass needle, and distance counter will be updated every so many frames per second, keeping the user up to date on their distance and direction.

# **Design Overview**

The Campus Assistant will rely on the help of a web service, and a database in order to function properly. Information from the database will be delivered to the Campus Assistant via a web service. This information includes; user ratings, facility information (i.e. location, description, etc.) These deliveries will be seamless to the user, who will be busy finding their class or a place to sleep.

The design of the user interface will be very easy on the eyes, and not cluttered. Remembering that the goal is to be intuitive, as well as user-friendly, will be key in designing the layout of the U.I. The colors will be soft, minimal, and will flow well with one another. Each color will be paired with a particular feature, that way a familiar user will not even have to read the buttons.

Test Plan

# 

The Test Plan for this project will cover all applicable phases of testing during the development cycle. It will encompass applicable software risks to both the client and development team, as well as outline the strategy/approach to testing at each stage. Unit Testing will be performed by the developers and will lead to System Testing once each individual module is free of defects. Finally, Acceptance Testing will be carried out by select users to be monitored by the developers. Acceptance testing will only occur once all major defects found in System testing are removed.

Unit Testing will utilize a white box methodology to make sure internal structures function properly, with classes and their associated methods being the test cases. Expected results versus actual results will determine if the test passes or fails(positive v. negative test cases).   
 Metrics will be categorized based on; module, severity, and origin of the defect. All test cases will utilize a traceability matrix, and will include a description of the functionality tested, as well as the results of the test. Final testing deliverables will include test cases on all levels, interface prototypes and concepts, defect reports, and debug logs. Unit testing deliverables will include source code and a defect report. System testing will include any referenced documents as well as a defect report.

# Cost/Efficiency/Effort Algorithm

In order to keep development moving forward, we have to make sure that we are using our time efficiently. Since our development is feature driven, we must break down these features into measurable units, so that we can put our timeline into a measurable form. First, each feature will be given a certain percentage of our timeline, which is measured in hours. After that, each individual part that makes up a feature, will receive the appropriate percentage of the feature’s hours. Using this method, the development team can determine how many man-hours each individual part of each feature should take to develop. The amount of man hours put into a part, or a feature overall, shouldn’t be more than the percentage of hours that the particular part requires. In the event that more man hours are put into a part than the percentage of allotted man-hours the part requires, then there will be negative man hours counted against that part. Any negative man hours will incur inquiry upon the actual activities of the man hours used. This is where the final metric we will be using comes in, which is timesheets. These timesheets will contain a log, that is, what is worked on by each team member, how long they worked on their objective, and what day the worked took place. By checking these timesheets, we can look at how much actual effort is put into a part. We can correct development methods, and ensure efficiency through this particular “Algorithm”.

# 

**Project Schedule**

# 

1. Set Android Studio up, Web Service, and Database
2. Begin construction of features:
   1. Menu
   2. Map
   3. Database Connection
   4. Get Me There
3. Begin construction on next group of features
   1. Facility info page
   2. Link G.M.T. w/ Facility info page
   3. Facility list
4. Final system testing & Documentation

## Gantt.jpg

**Ethical Issues**

# 

The Campus Assistant is expected to be used predominantly by college students, however, it may be used by visitors to CSU as well. The information will be accessed, and displayed in such a way that no demographic will be offended, or feel that their use is not desired. The ratings apply only to the quality of the amenities located in a particular building, and not the demographic that uses these amenities. There are no foreseeable scenarios in which a user will feel that they are not appreciated for using the Campus Assistant.

# **Impact on Society**

The Campus Assistant is designed to have a positive impact on society, by helping users find their way, thus reducing stress on campus at the beginning of each semester. This stress will be reduced by minimizing the confusion over class locations during the first few days of the semester. Students are sure to be pleased with the Campus Assistant, as it is catered to their needs, and wants. Few students will actually ask where the best place to sleep is, but it is a question that many students ask themselves every day, and the Campus Assistant will be their source for the answer to this question. An app designed on campus that provides this much help to students will give CSU something to be proud of.

# **Risk and Liability**

There is no monetary risk involved with this project, as no money has been spent toward it’s development (except for pizza at the meetings). There was originally going to be a cost of RFID tags, however since then there have been some changes made that have eliminated the need for those. These changes appear in the change-log.

Liability for this app will be minimal to non-existent, seeing as how it is only an informative app that provides navigation service across a college campus. There is little room for any damage to be done, and certainly not at the fault of the Campus Assistant.

# **Conclusion and References**

Through the use of it’s many features, the Campus Assistant will be able to service the needs of incoming freshmen, transfer students, visitors, and even students who have been at CSU for a while. We look forward to getting to the construction phase of the project, so that we can see our ideas come to life. We have high hopes for the Campus Assistant, and look forward to sharing it with the student body.

# 

# 

# **Change Log**

|  |  |  |
| --- | --- | --- |
| **Team Member** | **Changes Made** | **Date** |
| Group | Revised the features of the app. Less A.R. (ooh and ahh), more functionality with use of ratings, Get Me There, etc. | 2/12/15 |
| Kevin Dowling | Renamed “Halp” to “Info/ Settings | 3/18/15 |
| Kevin Dowling | Gray buttons chosen in favor of colorful, in order to make the application easier on the user’s eyes, as well as look more professional. | 4/6/15 |
| Drew Polhamus | Use of PIN ID’s over RFID tags. This will eliminate the cost of RFID tags. | 4/15/15 |