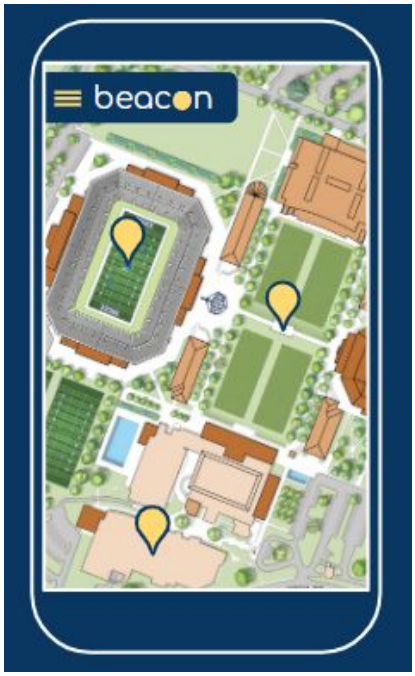
Project Plan for
No Bugs Allowed:
Beacon



Gerald Beinhauer, Jonathan Ingram, Charlie Redding

Capstone Project Spring 2020

The University of West Florida

January 31st, 2020

CIS 4592 Capstone Project

Dr. Bernd Owsnicki-Klewe

Abstract

This document is made for the purpose of explaining the project plan for developing the android app, Beacon and gives a brief overview of the purpose of the app. No Bugs Allowed will be working on the project over the semester to meet the requirements of a capstone project. In order to stay on path with the goal of this capstone, this document can be referenced to view decisions that have been made and what has changed from the beginning of the semester.

Table of Contents

1 Executive Summary

Beacon is a social network of places and events. Users will find interesting locales, pin them to a map, and share them with friends and others. These pins are dubbed beacons. Like the real-world counterparts they are named over, these beacons will draw people to their location

As more and more users discover and enjoy these beacons, they'll be able to contribute to their luminosity . Contribution grows the beacons and signals to others where many users are congregating. This feedback loop of finding and growing beacons is at the heart of the app.

Overtime, beacons' luminosity will slowly dim. Once completely dimmed, the beacon will disappear. This transient nature plays into the central feedback loop in two major ways. Users are incentivized to revisit locations they've previously been or to contribute to the locations they visit regularly. Additionally, if a beacon is tied to some temporary event, this process ensures that the beacons does not remain active long after the event's conclusion. Overall, this means the total number of beacons never grows impractically large.

2 Expected Project Requirements

To organize the project requirements, the group has opted to use Trello. Trello is an Agile tool that allows designers to create various lists to facilitate better, incremental, and more streamlined development. One such lists that we have created is that of Use Cases.

2.1 Use Cases

Below are some of the use cases we've devised for the project. All were pulled from the shared Trello board.

- Users would like to create their own beacons
- Users would like to view nearby beacons to their location
- Users would like to grow the beacons they enjoy
- Users would like to maintain a persistent account
- Users would like to save beacons for later or offline viewing
- Users would like to add images, descriptions, or tags to beacons

2.2 Main Components

The project will consist of three major components: a client, servers, and databases. The client will be the Android application, Beacon. This will be the main user interface of the project. Servers will be responsible for facilitating communication between the client and databases.

Databases will serve two main roles. First, databases will be responsible for maintaining information on existing beacons including geographical information, images, tags, etc. Furthermore, databases will need to store encrypted user account information. For purposes of security, these two databases will be logically separate.

3 Anticipated Development and Deployment Process Model

We will be using an Agile development process in order to complete our project. One advantage of this model is the ability to change minor aspects without redeploying the entire system. This model will also allow us to dynamically adjust the project schedule. We chose this model both for its familiarity and because its structure compliments the relatively short development window. Moreover, we will utilize the Scrum approach that utilizes sprints to achieve rapid and regular development.

4 Assignment of Team Member Roles

Capstone requires that teams be comprised of members with a broad range of technical backgrounds. Our group includes two Computer Science majors and one Software Engineering major. Charlie Redding is designated as the Team Leader, and, along with Jonathan Ingram, will be responsible for most of the programming and testing. Gerald Beinhauer will be the team's Software Engineering lead. At this time, the group does not include a Cybersecurity lead, but we will, to the best of our abilities, try to incorporate these principles.

Additionally, the development model we are employing advocates two unique roles. Because of our relatively smaller team size, there will need to be some overlap between these roles and the development team. The first role that we will need to fill is that of Scrum Master, who ensure we are adhering to the principles of our model. Gerald Beinhauer will fill this role. The second role is the Product Owner who informs the development course from the perspective of an outside client. Jonathan Ingram will fill this role.

5 Anticipated Testing Strategies and Tools

In order to test our software, we will be using integrated testing methods inside Android Studio. One such example of integrated testing capabilities is Espresso. Many, if not most, of our testing will revolve around the client application. The Android Studio environment also includes unit testing; this method will be the main thrust of our testing.

After passing the unit testing, we can move on to integrating testing between the classes and outside sources, i.e client-server communication and validations. Following this, the final phase of testing will consist of UI/UX tests to ensure that the software will have a positive human-computer interaction. Throughout the development process, we will occasionally emulate the software onto an Android device.

At this time, we are not aware of what testing will be necessary to or employed on the database and server components. While we know that some testing will be required, we have not designed how these components will ultimately look. Because of this, it is difficult to outline what testing may resemble.

6 Preliminary Project Timeline and Tracking

We are using the Scrum method to track the project. This means that we will have scrum meetings every two weeks. We will discuss what has been accomplished since the last meeting as well as what work will be done before the next. Furthermore, we will discuss what hurdles or complications may arise or have arisen related to our continued development.

Provided on the next page is a working schedule we have compiled to direct our efforts. It includes any deadlines that the class demands as well as self-imposed deadlines and meetings.

6.1 Schedule

1/7/2020	Form Group
1/14/2020	Project Idea
1/28/2020	Presentation 1 Project Plan
2/3/2020	Scrum Meeting 1 Sprint 1 Start
2/11/2020	Group Meeting Technical Documentation
2/17/2020	Scrum Meeting 2
2/25/2020	Group Report 1
3/2/2020	Scrum Meeting 3 Sprint 2 Start
3/3/2020	Presentation 2 Group report 2
3/16/2020	Scrum Meeting 4
3/17/2020	Group report 3
3/24/2020	Individual Presentations Group Report 4
3/30/2020	Scrum Meeting 5 Sprint 3 Start
3/31/2020	Individual Presentations Group Report 5
4/13/2020	Scrum Meeting 6
4/21/2020	Final Presentation Final Deliverables

7 Project Risk and Success

We define project success as a functional Android app. This includes being able to create and view beacons, grow others', and modify them with images, descriptions, and tags.

We anticipate many problems that may crop up. One example is the inability to scale the software to large geographical areas. In the event this happens, we will restrict the usable are to only the UWF Pensacola campus. Another potential issue is the inability to implement beacon-attached images, descriptions, and tags. In this event, he plan to prioritize tags first, followed, respectively, by descriptions and images. Our last and most important concern is an inability to produce a cyber secure app because of the composition of our team. Should we not be able to produce such a product, our plan is to outline whatever vulnerabilities exist within our final report.