Ghost Grab



Dylan Dover, Selina Chen, Jason Mayer, Lawrence Waller

ENGM 3700.02

04/21/2016

# Executive Summary

Our company will design an innovative Android game called GhostGrab that will utilize GPS coordinates to create an immersive environment. We will utilize a number of tools and frameworks to accomplish this goal. Among them are Unity on the client-side and node.js on the server-side.

GhostGrab will be a location-based game that will allow users to use their phones to view and capture “ghosts” across a chosen location radius and release them or set them on others for points. Users will have to play mini-games or solve riddles to capture ghosts and there may be random team competitions, allowing users to collect bonus points. There will be a leaderboard, which will update in real-time to show who is leading in points and allow users to track their rank and the ranks of their friends. Created in Unity, it will be easily portable to a variety of platforms, though for a minimum viable product, we will be releasing an Android-only version.

To keep the game interesting over time, updates will be available in the form of new ghosts and new mini-games. This model makes it possible to easily update the game with minimal time investment, meaning that our company will have time to pursue other projects simultaneously.

Contents

[Executive Summary 2](#_Toc448850915)

[Scheduling 4](#_Toc448850916)

[Responsibility Matrix 5](#_Toc448850917)

[Risk Management 6](#_Toc448850918)

[Problems & Solutions 7](#_Toc448850919)

[Quality Assurance 8](#_Toc448850920)

[Future Plans 9](#_Toc448850921)

# The Project

## Background and Rationale

GPS has become widely embedded in mobile devices, and games that interact with users in the real world have become more and more popular. With the recent announcement of the VR (Virtual Reality) game, Pokémon GO™, and already existing augmented reality mobile apps such as Ingress, there is a lot of widespread interest in games that users can bring into the real world.

## About the App/Functionality?

This app is available to Android phones. A basic walkthrough of how the app is used is demonstrated through the pictures below:

First login by creating a username and password? [INSERT PIC]

Next

Happy ghost hunting!

Some other features include a leaderboard, where you can see who is leading in points.

[INSERT PIC]

## The Code

The code for this app as well as other documentation is publicly available and can be found at <https://github.com/wallerl2/ghostgrab>. A basic layout of the way the code is set up is demonstrated in the UML below:

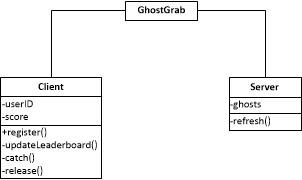


Figure : UML Diagram for App

The client, which is the code written in Unity, holds the graphics for the game as well as the most of the game logistics. The server, which is being hosted on an AWS instance holds information such as ghost types, location of all the ghosts in the game, and the scores of the players. The client requests information from the server whenever it needs to (ex: when a user wishes to see the leaderboard, it sends a request) and it also posts information to server as well (ex: updateLeaderboard() function is called when a user’s score changes, and the updated score is posted to the server).

# Scheduling

This project took place over a period of about 3 weeks, starting on Tuesday, March 29th, 2016 and ending on April 21, 2016.



Figure : Gantt Chart Documenting the Project Schedule

# Responsibility Matrix

# Risk Management

# Problems & Solutions

Much of the initial problems that arose came from client-server interaction. Unity is a rather new.

Solved by using a library called UnityHTTP provided by <https://github.com/andyburke/UnityHTTP>, which is itself based on Simon Wittber’s UnityWeb code, and therefore is licensed under GPL, which is GNU General Public License, which allows end users, whether they be companies or individuals, to run, study, modify, and share the software. This eliminated much of the work and provided a neat way how Unity interacts with JSON web requests.

# Quality Assurance

Our company will design an innovative Android game called GhostGrab that will utilize GPS coordinates to create an immersive environment. We will utilize a number of tools and frameworks to accomplish this goal. Among them are Unity on the client-side and node.js on the server-side.

GhostGrab will be a location-based game that will allow users to use their phones to view and capture “ghosts” across a chosen location radius and release them or set them on others for points. Users will have to play mini-games or solve riddles to capture ghosts and there may be random team competitions, allowing users to collect bonus points. There will be a leaderboard, which will update in real-time to show who is leading in points and allow users to track their rank and the ranks of their friends. Created in Unity, it will be easily portable to a variety of platforms, though for a minimum viable product, we will be releasing an Android-only version.

To keep the game interesting over time, updates will be available in the form of new ghosts and new mini-games. This model makes it possible to easily update the game with minimal time investment, meaning that our company will have time to pursue other projects simultaneously.

# Future Plans

The current model of our product allows users to capture ghosts for points, which while acceptable for a minimum viable product is rather boring.

As mentioned in the executive summary, we intend to release updates, which will contain more features for our product to make the game more interesting. One such update we plan to release is one that will allow users to set their captured ghosts on other users, which hit points.

Currently, ghosts are differentiated by their appearance and the points they carry. In the future, we would differentiate more between the ghosts by adding mini-games and movements specific to each ghost type.