

# Multiplayer Activity Tracking Game for Phones

Pedro Henrique de Castro

2397775d

## Proposal

### Motivation

Activity tracking games are an amazing way to keep phone users and people who like to play games casually active, if done right, they combine the fun with playing games with the benefits of living an active lifestyle. In addition, making these games multiplayer encourages users to go and play with friends outside, and even meet new people doing so.

### Aims

This project aims to develop an android app that uses the user's current location to as the player's movement in the game. To have an effect in the game (make the player move), the user has to walk around in real life. Multiple users will enter the same party and will be able to see each other's movement in real time, that way they can work together towards a common goal.

### Progress

- Programming Software chosen. Coding is done with C# and the game logic is being implemented using Unity Engine.
- Multiplayer framework chosen. Multiplayer logic is handled by a Unity Library called "Mirror".
- Activity Tracking framework chosen. "Mapbox" is being used to reproduce real life maps and track user's current location.
- Research on activity tracking games and apps done to make sure they feel as smooth as possible.
- Very robust GUI has been implemented to replace Mirror's GUI template.
- Game objects synchronisation between clients is nearly done, with just a few bugs left.

## Problems and Risks

### Problems

- Mapbox framework only worked for an outdated version of Unity, this isn't mentioned in their page so it was tricky to figure out.
- Firebase authentication worked for the PC build but not for the android one, thus the idea of login and progress saving linked to a login had to be scrapped.
- Syncing player and other objects movement across all clients was a challenge due to the fact that there aren't many tutorials on Mirror online so I had to rely solely on their documentation which is very superficial.
- Updating text individually for each client is still a problem that needs to be solved, each client has their own score that updates individually but the UI text representing this score doesn't.
- The game was supposed to run on a server, that way players could have their progress saved and loaded, but dedicated servers cost money so the solution for this was to have temporary servers created every time someone starts a game.

## Risks

- Game will feel very repetitive and have little re-playability. **Mitigation:** Introduce different classes or difficulty levels that the player could explore.
- Game might not work on some devices on 4G because of firewall blocks. **Mitigation:** Ask for permission from the device to use data.
- Users might get distracted with the game and end up involved with accidents in real life. **Mitigation:** Not have serious in-game consequences if the player is not looking at their screen very often.

## Plan

### Semester 2

- Week 1-2: Fix problems syncing variables.
  - **Deliverable:** Fix all bugs regarding the syncing (or un-syncing) of variables and game objects movement across all clients.
- Week 3-4: Improve playability.
  - **Deliverable:** Have the player be able to choose between more classes and also make the enemies harder to beat.
- Week 5-6: Improve graphics.
  - **Deliverable:** Improved player and monster models, change the map look so it suits the premise of the game, improved interface graphics.
- Week 7-9: Evaluation.
  - **Deliverable:** Have users test the alpha version of the app and report on any bugs they find. Have the beta version of the app available on Play Store and get the feedback from other users.
- Week 8-10: Write up
  - **Deliverable:** Have a rough draft of the dissertation done and send it to supervisor for feedback two weeks before the official deadline.