Smartphone Tracking App for Microsoft HoloLens

DESIGN DOCUMENT

Team Number

27

Client

Optical Operations

Advisers

Dr.Qiao

Team Members/Roles

Travis Harbaugh/HoloLens Development
Ben Holmes/ Android Development
Anthony House/Website Development/Security
Ryan Quigley/Database Admin
Joes Lopez/Website Development
Cory Johannes/Report Management

Team Email

sdmay19-27@iastate.edu

Team Website

http://sdmay19-27.sd.ece.iastate.edu/team.html

Revised: Date/Version

October 11th, Version 1

Table of Contents

List of Figures

List of Tables

List of Symbols

Not Applicable

List of Definitions

Not Applicable

1 Introduction

1.1 Acknowledgement

Sdmay17-29 would like to thank Andrew Guillemette and his engineers from Optical Operations for their contribution to our project. Andrew contribute significant assistant informing our team with technical advice and equipment contribution. Special thanks goes to Professor Qiao for taking the time to meet with our team and helping us learn about the different technique in embedded sensor. Lastly, our team wants to thank Andrew engineers for graciously helping our team with the HoloLens.

1.2 Problem and Project Statement

General problem

Work surveillance is difficult in large scale construction site. These sites have many different zones with no way to manage workers in 60 story buildings. These sites can be very dangerous and there is no real time tracking solution that can manage the logistic. GPS accuracy is within ± 3 meters and doesn't always work indoors because there is a lot of interference with steel, concrete, and large objects (skyscrapers). Many large tech corporations have tired to improve indoor navigation but unable to find a scalable solution. if a feasible method for tracking workers is realized, it would provide a way for managers to monitor their employees and update their client on process.

General solution approach

The purpose of our project is to make an application that can track an individual indoors within ± 1 meter as a proof of concept so our client can create a prototype that can be used as a construction site tracking solution.

To create our unique solution, we will be developing mobile/HoloLens software application that will work complement each other. The mobile application Primary responsible will track a user within 1-meter accuracy indoor/outdoor environment. To achieve this requirement, were going create a smartphone application which utilize the phone's Accelerometer, Magnetometer, and Gyroscope to update changes on the phone's movement, and relay the information to a server.

The accelerometer sensor is used to determine the change in acceleration which will be used to find the user new location. The accelerometer will also be used for step detection and will estimate the length of user's footsteps in real-time. The Magnetometer, and Gyroscope will be use to determine the changes in the device orientation which will give you the user's direction/heading. With these three sensors we will be able to calculate the user's location and relay the information back to the database that will be used by the HoloLens application.

The HoloLens application primary responsibly is to retrieve the user's location stored in the database recorded by the mobile device and then render the new position of the user's avatar in the augmented environment on the HoloLens. To achieve this requirement, we are going to create unity application that uses a mapping environment that will take 3D model of the jobsite and located it on the map. We will then get 3d avatar models on the unity store that will simulate workers on the website that will simulate the user's movement throughout the work day.

Project Motivation

What is driving this project is that there is no real solution that provides any surveillance in a onsite. There is a lot of construction that goes on create huge skyscrapers using steel, concrete, etc.

There are many different zones, machinery, and infrastructure that needs to be monitoring to provide safety for workers. According to Untied States Department of labor, 4,190 workers were killed in 2016. They stated that 1 out 5 workers deaths last year were in construction. We need a solution that can provide information on what goes on at these construction site to provide better safety at the work site.

Project Overview

The project is mobile indoor tracking solution on a construction job site that will allow you to track an individual's movement indoor/outdoor. The construction worker will be track by using the phone's Accelerometer, Magnetometer, and Gyroscope sensor that will update the user's position and rely it to the server. From the office we will use the HoloLens to display a 3D model of the jobsite. Within the model there will be 3D avatars that represent construction workers that allow the office employees to monitor their position at the job site in augmented reality.

Project Outcome

Our team would like to complete an Android application which unitize phone sensors to predict an individual's location to within 1 meter of accuracy inside a worksite and store that information to the database. Next we would like to develop the HoloLens application that will take the new position of the user and use augmented reality to display their position in a 3D environment in real time. The outputs to this product would be update the workers position has they move there out the worksite and display it in Realtime on the HoloLens.

1.3 Operational Environment

Our operating environment is going to be for a construction site. our end project will be using a mobile device application to gather a user's location with little to no assistance with a GPS sensor. We will be using secondary sensors on the phone and Bluetooth beacons, to track the use location inside buildings. Due to extreme weather and dangerous conditions our device must secured on the person's body. With very dusty conditions and extreme hot temperatures the device must be able to perform tracking operations without failure. It also must not impede the workers ability to perform on the job site. The mobile device will track accident reporting and identify if a collision has occurred.

The consumer must wear this device at all times while on the job site. This application must be able to store the location data where there is no access to a connection for a long period of time (4hr) and then be able to update that data to a server when we have access to an access point. Also, the device must not be too CPU intensive where the device can last for an 8-hr day.

1.4 Intended Users and Uses

Our intended user base is construction workers. Our smartphone application will track its user using a combination of the phone's sensors. Doing so will provide a level of supervision over construction projects. It is our hope that this addition level of supervision can be useful in solving logistic issues. It will also provide a record of daily work, which will detail the productivity and current progress of the construction project.

Branching out from the Construction industry, this method of smartphone tracking could also have applications in other fields, such as medicine. For example, Doctors could use the application for

monitoring their patients. The app could send alerts to the Doctor if a patient has left their room or gotten out of their bed.

1.5 Assumptions and Limitations

We assume that all equipped phones have the following sensors:

- Magnetometer
- Gyroscope
- Accelerometer

We also assume that the audience will be either walking or running when wearing the device. This algorithm is no meat to track people riding in vehicles. To ensure the accuracy of our tracking algorithm, we require that user's wear the smartphone device on their right arm. We are also limiting the environment to Durham. Aside from this, the application is expected to work regardless of a user's trajectory, or type of movement. They can be walking or running.

1.6 Expected End Product and Deliverables

The end product will include 3 parts:

- Android application
- Website
- Hololens application

Deliverables:

- An Android application which relays position data to a server without the need for GPS
- Robust tracking algorithm which utilizes smartphone sensors
- Server application which displays lat/lng coordinates on Google Maps for semi real-time position tracking UI
- Access to the git repository which contains all the documents from the semester and code written for above systems.

2. Specifications and Analysis

2.1 Proposed Design

2.1.1 Concept Diagram

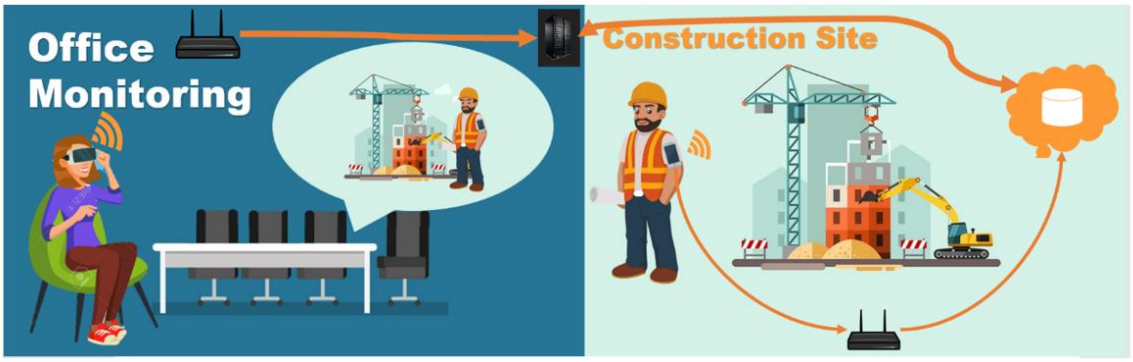

Figure 1: Concept Diagram

Office Monitoring

From the client's office an employee will be sitting in the conference room with the Microsoft HoloLens monitoring the jobsite. The HoloLens renders the construction site, building objects, vehicle objects, and employee avatars in augmented reality. Once the database has received an update of an employee or vehicle position change, then the server will transmit the location data to the HoloLens and render the employee or vehicle object in augmented reality with the HoloLens. The HoloLens will use an access point from the office to receive the transmission from the server.

Construction Site

When the construction workers enter the work site, the mobile application will deadlock their current position. The phone will be attached to the users right arm using an armband and must be worn at all times during construction hours. As the construction workers walk around the site, the mobile application will record their change in position using the cellphone hardware sensors. The data collected from the sensors will then calculate the construction workers new position in Realtime and then send data to the database to be recorded.

2.1.2 Architecture Diagram

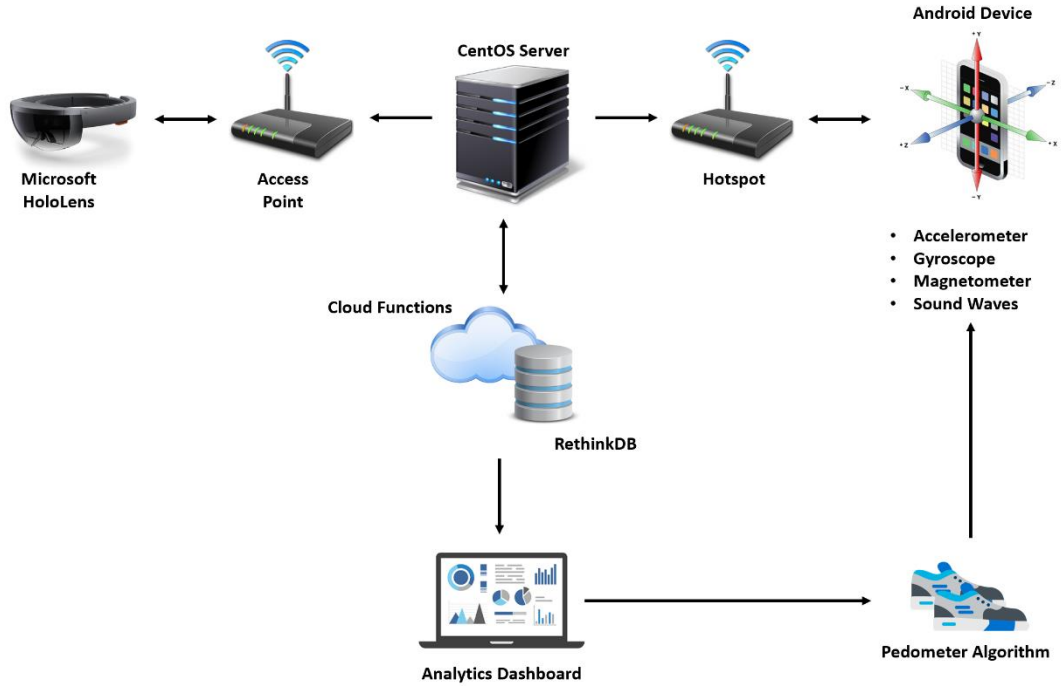

Figure 2: Architecture Diagram

Website

The website receives longitude/latitude coordinate data from the database and updates the construction workers marker on Mapbox. The website also receives accelerometer x,y,z data from the database measured from the change in acceleration due to user movement. The website creates a graph to display the accelerometer data with respect to time. Our team analyzes the peak and valleys of the graph to create an algorithm that can be used to detect user's movement and estimate distance traveled.

HoloLens

The HoloLens connects to the office access point so it can receive notations from the server. The server sends a notification when the database receives an update from the mobile application due to a change in user position(longitude/latitude coordinates). The HoloLens takes longitude/latitude coordinates and renders the new user position and displays it in augmented reality.

Android Application

The android application detects user movement from the hardware sensors. The application determines if the user has moved from his/her current position and calculates the new location. The application then connects to an access point within the construction site. After a connection has been established, a request to the server is made to store the updated user location.

Database

The database stores x, y, z accelerometer data and longitude/latitude coordinate data. When the database receives a request from the server to store new user position or accelerometer x, y, z data from the phone, the database updates its storage and broadcasts the new data to both the Microsoft HoloLens and website.

2.1.3 Mobile Application

Android OS

Android OS is the world's most popular mobile operating system. The operating system is Linux kernel and it offer 3 different programming languages you can develop with. There are many companies that design and employ android devices. Android has great documentation and support that makes it easy for new developer to create applications. They offer many different development tools that helps distinguish them from their competitors.

android studio

Android studio provides tools for building apps on every device (TV, Cars, Watch, Computer, tablet) that allow developers create new experiences to its users. The visual layout editor allows you to make complex layouts that give you many configurations that allows you create visual appearing views. The APK analyzer allows you to compare your application as you develop new features in the future. Android studio give you a fast emulator that allow you to develop application and install it on a virtual phone so you can simulate different configurations that can enhance your users experience. The intelligent code editor allow you better code and be more productive because you have faster access to the API. Lastly, they provide real-time profilers that provide statistics for your application that allows you to identify performance issues.

Java

Java is a programming object-oriented language that can be run on all platforms. Java is the top choice when it comes to programming android applications. This is because large parts of android was developed in java. Java allow you to use design patterns, inheritance, and other object oriented features that allow developers modular code.

Our design is an mobile application that is used by construction works to track their indoor/outdoor movement throughout the work day. Our solution will use the secondary phone sensors to track the individual by using sensors. Those sensors include: GPS, accelerometer, magnetometer, gyroscope, pedometer.

GPS

The GPS sensor will be used to get the users latitude and longitude coordinate which is a relative position to where the user is in the world. The idea is the following: given a GPS location you turn that into real coordinate and you use the longitude and latitude to translate it to X,Y,Z coordinate that will then be used with the other three sensor data to estimate the user location relative to the world. The GPS sensor is used as a substitute to get a landmark for an initial location to track a user as they move inside a building.

Accelerometer

The Accelerometer sensor is used to measure the acceleration of a moving body. It will detect if there is any acceleration relative to the mobile device. The data that our app will receive will be X,Y,Z coordinates. My team then can double integrate the acceleration with the respect to time to get the new user position. the data that our application will collect from the accelerometer sensor will estimate the user location from the previous location that was collected from an earlier reading.

Our team will be using the accelerometer to track a user's footsteps. This will allow you to calculate the distance the uses walks by using the accelerometer data. The reason we are using this approach is because if you double integrate the acceleration from the accelerometer there are some error involved getting position.

Gyroscope & Magnetometer

The gyroscope is used to determine the cellphones orientation. This allows you to rotate the phone to measure the angular rotation velocity. The magnetometer sensor uses magnetic field to determine heading. This will get heading by detecting the earth's magnetic field along each x, y, z axis. Using the gyroscope and magnetometer we can determine the direction the user is heading. The reason for using both of these is there is some error that is associated with this sensors.

Volley

Our team will use volley to push construction worker location data to the RethinkDB. Volley is an HTTP library that allows android to send and receive data easier to our application. Volley allow use to make easier and faster network calls to the database.

2.1.4 DATABASE

RethinkDB is open source database that is scalable for Real Time applications solution. This database scheme offers an unique solution from traditional base which instead of pulling changes from the database you can subscribe to the database and it will continuously send you updated information in Real Time. With this new architecture design, you can have an application that pushes new information to the database and then the database will take care of sending the new update information to its subscribers.

The benefit of using this database is it model in a way that works well with web applications. This is because it maps to the HTTP's request-response. But unlike the traditional application that requires you to send the data to the client this will allow you design the application that is not concern how the client uses this information which makes it more modular approach.

RethinkDB is different than then Google firebase because unlike the Real Time sync databases that are cloud base, RethinkDB is open source and you can deploy it in your applications because it overs open source. It gives you more freedom without restrictions. It doesn't limit you on your querying capabilities like amazon or other database options. Lastly, Realtime API requires you to have access to the bowser but RethinkDB can be accessed from an application server which can be very appealing for Realtime applications.

One of the most appealing benefits of RethinkDB over mongoDB is the fact that instead of polling for changes, you can tell the database to continuously push the update the quires to your application. one of major benefits to this design is that you can subscribe to changes on query results, and not just on raw replication data which reduces the development time on your application (ref 2). According to ref 2 RethinkDB advantage over MongoDB are:

1. An advanced query language that supports table joins, subqueries, and massively parallelized distributed computation.
2. An elegant and powerful operations and monitoring API that integrates with the query language and makes scaling RethinkDB dramatically easier.
3. A simple and beautiful administration UI that lets you shard and replicate in a few clicks.

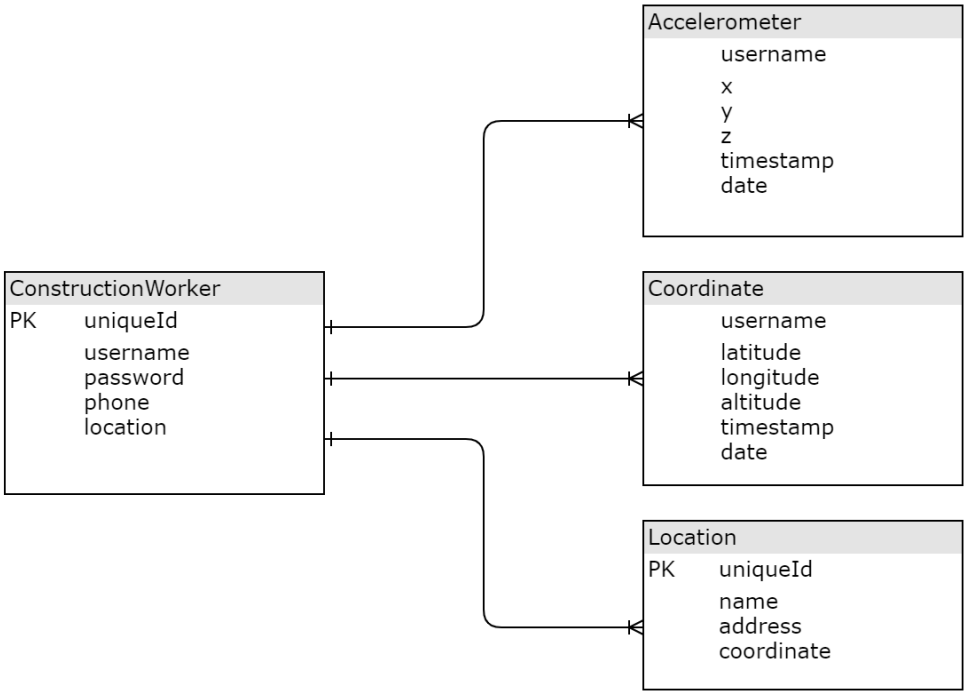

Figure 3: Database Schema Diagram

Construction Worker Table

The database will have 4 schemas (construction worker, Accelerometer, Coordinate, location). The construction table is used to store an single user that you want to track at the jobsite. The table will store their username, password, phone number, and their jobsite location. This table will be used to extinguish each construction site and their workers so that we can actually track the individual at the current location.

Accelerometer table

The accelerometer table is used to store the accelerometer x,y,z values along with the timestamp and date that will be used to analysis the data to determine an algorithm for step detection.

Coordinate Table

The coordinate table is used to store the longitude, latitude, altitude, timestamp, and data the coordinate was recorded. This will allow the client to track workers on the construction site. By storing the data and time stamp it gives the ability to retrace a worker's footsteps.

Location Table

The Location table is used to store the name, address, and coordinate (longitude/latitude) of the job site that is associated with the construction worker. This table will be used to extinguish each jobsite the client wants to monitor.

2.1.5 HOLOLENS

For the HoloLens development we will be utilizing the unity software framework. Unity is a game engine that allow you to architecture 3D model assets and render the textures into augmented reality. Unity has its own IDE that will allow you to use the game components that allow you to interact with the world that you have created with scripting, 3D modeling, animation, object colliding detection, UI menus, backend, Mapbox apk. The advantage of using unity over C++ is that it is very easy to take 2D/3D experiences and augment them into the HoloLens and use the state-of-the-art characteristic to provide new user experiences.

2.1.6 SCRIPTS

Camera scripts

The building zone camera script will allow the office personal to monitor different zones of the jobsite by using an xbox controller, hand gestures, or voice commands to cycle through the different zones.

The 1st person/3rd camera script will allow the office personal to monitor from the workers perspective by switch to the avatar they want to monitor and toggle between 1st or 3rd person camera angle. This can be done by using xbox controller, hand gestures, or voice commands inputs.

Worker/vehicle movement scripts

The player control script will allow the avatar to move from the current position to its new position received from the mobile application and then translate their movement in the HoloLens augmented relativity environment where the it can be monitor by the client.

The vehicle control script will allow the vehicle game object to be move from its current position to its new position when it receives an update from rethink database and then translate their movement in augmented reality to display there current activity in real-time.

Particle System script

The particle system script will render some special effects when an avatar hits an object (wall, Vehicle) to notify the monitor that the worker has an object.

Animation script

The animation script will animate (idle state, walking state, running state, etc.) on the HoloLens depending on the activity of the worker/vehicle.

Controller script

The controller input script will allow the monitor in the office to switch to different zones, workers or vehicles on the construction site. it will also allow to perform a playback facture that lets view previous movements.

Collider Detection script

The collider detection script will detect when a worker or vehicle has collided with an object on the job site. this script will inform the mobile application if the worker avatar has hit a wall indoors . this event will tell the mobile device that we are experiencing error in there tacking and they need to try to hone the users position back in.

UI menus scripts will receive finger gestures, voice commands from the HoloLens and perform the necessarily response from that command. These commands could be but not limited to switching to different zones on the construction site, switch to different worker or vehicle objects, or monitor an earlier event.

Database script will send or receive data from the database. if the database sends data to the HoloLens it will parse the data received and send it to the player or vehicle script to render the new movement of the game object.

2.1.7 MODELING

3d modeling

The HoloLens renders 3D models of the worksite in augmented reality using the Mapbox SDK, a mapping platform for Unity3D. Our prototype involves tracking user locations inside of Iowa state campus buildings as a proof of concept. To be able to monitor individuals on the HoloLens we need to have 3D model of the campus buildings so that it can be rendered in augmented reality.

To locate information about the campus building about the campus building our team searched the Iowa state website www.fpm.iastate.edu. the site helped us locate a link to 3d SketchUp models from Trimble Warehouse that had 3d models of the campus buildings. The model of each building is in a .skp file format along with a texture pack for each campus building.

Unity requires that the 3d model be in the format of .obj or .aed to import it into its IDE. SketechUp Pro offers a converter that can convert .skp files to .aed and .obj model. Once a model file is converted and loaded into Unity3d, it can take the textures files and apply them to the polygons of the 3D model. The model can then be geolocated to the exact location on the map platform to be rendered in augmented reality.

To be able to monitor individual workers and vehicles on the job site we need to create some vehicle and character models. Unity offers these models on the asset store and can be purchased and imported into our project.

Figure 4: 3D Model of Durham

2.1.8 WEB

We will be using a web platform showing a 2D rendering of real time events. This will allow an administrator or project manager to view actions in real time and better coordinate efforts in getting tasks done. The technologies we will be using include, but are not limited to, HTML, CSS, JavaScript, and potentially JQuery.

2.2 DESIGN ANALYSIS

Cross platform vs Native

One of the problems about development for android or iOS is that they use different native languages to develop on their platform or device. For example, if you wanted to program for the iOS platform you would have to have access to a MacBook computer. There are two platforms that allow you to create application for both platform (android, iOS) using wither Xamarin or React Native.

Xamarin is an open source software that uses the .Net framework. Xamarin uses the C# programming language within the .Net framework that allow you to make cross platform applications. React Native is another open source software that uses JavaScript. Since it only uses JavaScript

Pros of React Native:

1. developers that know how to develop in reactive native can create apps for the web, iOS, and android platforms.
2. Allow you team to create prototypes more quickly which can save development time
3. Allows to reuse components for different applications
4. You spend less time on regression tests
5. Bugs are solved more quickly

Cons of React Native:

1. it a fairly new software, so it has less support for new developers.
2. It requires the developer to have experience in the native platform because they might have to write code for that platform.

3. If you need some advance features for your application then it will hurt your runtime performance
4. It has a steep learning curve
5. It will affect your security to your application. You have to be very careful because of malicious attacks.
6. It is not suited for intensive computation

Pros of Xamarin:

1. Xamarin allow you to develop for mobile platforms. This will make your development more scalable.
2. Xamarin performance is very close to the native language and it allows for fast prototyping which can be very helpful in the early stages in development.
3. Xamarin is open source software which allow for better support communities.
4. Xamarin has there one IDE which will allow for faster development.

Cons of Xamarin"

1. It has delayed support for new iOS/Android software updates.
2. Mobile devices has more access to open source libraries then Xamarin
3. Developers have to have knowledge to the native languages to use this platform
4. The Xamarin application are larger than the native applications
5. It has compatibility issues with third party libraries.

The advantages for programing in android native are:

1. allows experience android developers to create complex UI design that will create better user experience.
2. Android applications are optimized for their particular system
3. You have access to libraries and frameworks written by google
4. The programmer has access to android sensors

The disadvantages for programming in android native are:

1. The android platform uses a different operation system which will require the developer to learn new software to import their application to other devices.
2. It requires your user to download new updates.
3. You only know how to develop on Android IDE environment.

Cross platform disadvantages are:

1. If your application is on multiple platforms you must maintain multiple different code bases.
2. It is more expensive to maintain for multiple platforms
3. Android and iOS app stores have different requirements that the developer has to meet in order to release there application on there store.
4. You don't have direct access to android frameworks, libraries, and hardware sensors.

iOS vs Android

The iOS mobile device has Accelerometer, Gyroscope, Pedometer, Magnetometer, GPS, barometric, proximity, facial recognition, radio, Bluetooth, and fingerprint sensor. Android device has Accelerometer, Gyroscope, Magnetometer, Heart rate, Light, proximity, Fingerprint, Heartrate, proximity, Thermometer and Bluetooth sensor.

The devices that are available for our project is android devices. Our client has given our team two android device (HTC one, add other) to our team. One of our team member has an iPhone X while the rest of the team members have android devices. Since the majority of devices our android it would be very hard to develop on iOS because we would be limited to one mobile device.

Our solution for our indoor tracking constraint requires the user to wear an arm band on their right arm. There are armband sleeve, sport band, and wristband. The prices for these range from 7.99 to 17.97 on amazon. Since most of these armbands are universal we are not limited to either android or iOS device. The armband that looks promising is the rotatable armbands because it allow us to test our tracking methods from different phone orientations.

The pros of using an android device are:

1. Majority of our team has more experience on android
2. Android is open source which reduces cost of our project (licensing cost or royalty). Open sources Allows for more creative ideas.
3. Screen views are highly customizable and easy to manage through fragmentation
4. Android is programmed in Java or Kotlin which is easy to learn through many different open source learning communities.
5. Android offers expandable storage without buying a new device.

Con of using Android device are:

1. Android does heavy operations in the background which can drain you battery quickly.
2. Android process for processing a new application to the google store is very easy which leads to low security.
3. Android has complex layout schemes which makes it harder to develop for startup companies.
4. Android has higher fragmentation due to different android devices run on different operating systems.

The pros of using an iOS device are:

1. Fingerprint /face recognition increase security
2. iOS using swift programming language is easy to read. It does not have any programming language dependencies like iOS predecessor Object C.
3. Object C has two file (header, source file) like C that they have to manage which maybe harder to maintain in a large project.
4. Swift is open-source

Cons of using iOS device are:

1. Old iOS applications are coded in objective C which causes issue for new developers because of lack of Objective-C experience.
2. Object C is more prone to have runtime errors
3. iOS is not open source which reduces creative ideas and causes a steep learning curve which increases development costs.
4. Internal storage thus you can upgrade the hard drive unless you buy a new device.
5. iOS framework documentation is harder to find then android
6. swift IDE has a steep learning curve

Our team chose android because our mobile development team has more experience working with android development. Android has more opensource resources that allow our team to find more creative ideas for our project. While iOS is opensource it is harder to locate the documentation for the phone sensors which makes it harder to develop on their platform. iOS IDE has a steep learning curve and must own a mac computer to program the application and we were provided with android devices by are advisor. Swift is a specialized programming language which means unless you have prior experience making application for apple device you will have to learn a new language, however, in android uses java and java is used in different applications which will allow an easier transition to development in android.

With all of these constraints our team chose android because the majority of our team are very knowledgeable in java/android which will allow us to focus more on development our project instead of learning how to use different APIs. This will allow us to meet our two-semester timeline.

Backend:

Our solution revolves around storing the construction workers location during normal business hours. Since our solution involves having an access point to transfer the user location data, it must be able to store the data internally if it cannot send it to the external database.

The advantages of our solution is that we are using a mobile device which has a internal SQLite database that is optimize for Android and IOS. The advantages and disadvantages of internal over traditional:

1. SQLite is designed to small light weight operations and subset of features of traditional DB classes.
2. SQLite is serverless and it is zero figuration. Serverless means it runs only when your application runs while traditional servers are on all the time waiting for connections. Zero config means you don't need an IT or assistant administer to set up the system.
3. If your app has permission SQLite can read and write files without no extra configuration.
4. SQLite can access files directly, as for traditional systems are accessible through the network or other types of sockets.
5. SQLite is design for small datasets, one of the reasons is when you do a write operation it locks the whole file so other operations can't happen at that time. Traditional DB will lock one table at a time or potentially subset of tables. Meaning tradition DB's handle a lot of different writers at the same time. Whereas SQLite better with readers than writers.

For external database that our available for our project are MySQL, RethinkDB, MongoDB, and MariaDB. The advantages and disadvantages of these are:

MYSQL advantages:

1. Host-based verification
2. Has a very Flexible privilege and password system
3. Has Security encryption fo all password traffic

MySQL disadvantages:

1. No built-in support for XML or OLAP
2. Speed deficiency on big tables (not good for agile)

MongoDB advantages:

1. The database engine supports JSON
2. It can store any structure of data
3. The data Schema can be written without any downtime

4. Supports JSON
5. Encrypted storage engine

MongoDB disadvantages:

1. it was not designed to handle relation data models
2. SQL is not used as a query language
3. Setting up MongoDB is a long process
4. The default settings is not secure

MariaDB advantages

1. Allows high scalability with easy integration
2. Give you real-Time Access
3. Increase query performance and processing
4. Has the core function of MySQL
5. It has available Encryption for network, server and application levels
6. The system is fast and stable
7. Varity of plug-ins support

MariaDB disadvantages:

1. MariaDB is poorly supported on some operation system (it works the best for linux).
2. Switch back from MariaDB to MYSQL is not easy
3. Hosting environment doesn't support MaraDB yet
4.

RethinkDB advantages

1. Changefeeds , you can listen for changes what are continuous pushed
2. Supports sharing, parallel queries and MVCC
3. Powerful query language, (Node driver for JavaScript developers)
4. Compatible with JavaScript (Node.js), Python, PHP, Ruby, C, C#, C++, Objective-C, Java
5. Rethink is designed to access from an application server which allows a lot of flexibility
6. Atomic Updates
7. Easy to setup and learn
8. Is open source software

RethinkDB c disadvantages ons:

1. It's not ACID-complaint, doesn't have schema, it stores the field name of each document individually which can impact compression.
2. Mongo is 3x faster then Rethink for querying
3. You need to setup your own auth and user accounts

We chose RethinkDB because of the change feeds that the architecture can provide. The reason why this is important is because the HoloLens and website have to update the construction workers current position in real time. If we used a different database, then both the HoloLens and website would have to constantly make pull requests to check if any of the construction workers current location has changed. This would require our software to do a lot pull requests to achieve real time solution due to number of construction workers at each worksite. If we use RethinkDB our website/HoloLens team can subscribe to the change feed and will receive an event when a construction workers location has been updated and we can update both platforms without ever making a pull request.

HoloLens

The AR technology that is available are Microsoft HoloLens, Goggle Glass, and Magic Leap. The Microsoft HoloLens is AR technology that allows you to see 3D images that are projected in front of you while seeing the real world around you.

Advantages of HoloLens:

1. Comfortable to wear the device
2. Gesture recognition to select items
3. Voice command to navigate and control your apps
4. gaze to move the cursor to select holograms
5. Hands free experience
6. Unity development

Disadvantages of HoloLens:

1. Gesture recognition is very tricky and only a few gesture are available.
2. Cost $3000 for developer edition
3. Field of view gets distorted once you start moving
4. Needs windows for development

Disadvantages of Google glass

1. In 2017, Google only launched the Google Glass Enterprise Edition to companies like Boeing.
2. Google Glass Developer Edition was discontinued in 2015

Advantages of Magic Leap:

1. Much wider filed of view than the HoloLens
2. As two different size glasses which depends on your interpupillary distance
3. Lighter hardware on your head
4. Supports Mac development

Disadvantages of Magic Leap:

1. Field of view
2. Cost $2,295 for developer edition
3. Light pack computing device has to be attached to your belt that has a USB-C cord attached to the headset. also if the computing device breaks it will cost an extra $495 to replace
4. The leap glasses level needs to be aligned almost perfectly
5. Leap technology tracking and sense recognition is more visual jittery than HoloLens. space mating doesn't work properly.
6. Problem when glasses have to scan a room. Menus can go though walls.
7. Excessive heat on your head from wearing the device.

Our team chose HoloLens because that is what our client has provided.

3 Testing and Implementation

3.1 Interface Specifications

Mobile interface:

- Use 388 about the hub, sensors

Once the app is running, the use shouldn't have to interface with it. Since all of the apps functions are automatic and will run in the background, the workers can put the phone in their arm stap and let it be.

HoloLens Interface:

The HoloLens will make use of augmented reality(AR) by displaying the map for the user to observed. Outside of viewing the amp however, the user will not be able to interact with the map.

3.2 Hardware and Software

Phone Requirements:

For the ap to function properly, the right phone model is necessary. At minimum, the phone needs to support the Android operating system and include all the sensors unitized by the app (accelerometer, gyroscope, magnetometer, ect). Additionally, a model with a longer battery life is preferred; however, it is yet to be seen if the average smartphone battery could support continuous running over a single workday (~ 8 hours). Battery pack extensions could be an option if the default battery life is inadequate.

3.3 Functional Testing

Functional Requirement:	**Test Plan:**	**Verification:**	**Validation:**
Indoor/outdoor user tracking	We will setup 3 android devices at certain starting locations for tracking. We will provide the user with a specific path to walk which includes a start point and an end point. We will record their path and display it on Mapbox and compare their path with the test path.	Integration Testing: The AR team will design unit tests: • 3 avatar game object instances get created when 3 individuals are using the mobile application. • Compare the avatar path with the test path and report the results The Web team will design Unit tests: • Will store the user's path from the start point to their end point and	AR validation: The tester shall verify that there are 3 avatars that are rendered in the 3D worksite. They will verify that their path matches the test path specified by the test plan and it is rendered in the correct location on the map. Web Validation: The tester will verify visually that the users

		compare their path with the test path and report their results. The backend team will create unit tests: • Verifies that the schema tables get the correct user path from the mobile application. The test from all 3 teams will have code walkthroughs and reviews to verify that the unit tests are tracking 3 individuals according to their path trajectories.	path taken has the same longitude and latitude coordinates as the test plan. Database Validation: It will verify that the table schemas recorded all the location data from the 3 users' path from the mobile device. After the tests are completed, the tester will do a report analysis of the errors that they found and will be reviewed by the development team.
Movement Sensitivity	We will turn on the mobile application and then have the user perform normal walking and running speeds.	Usability Mobile test: • The user will launch the application and perform a walking movement and running movement and report the results. The mobile team will have a walkthrough to verify how this test is going to meet moving sensitivity requirement.	Tester: The user will turn on the mobile application and will perform running and walking movement and look to see if the mobile device detects their walking and running speeds. After the tests are completed, the tester will do a report analysis of the errors that they found and will be reviewed by the development team.
Store tracking to Rethink DB	We will connect the mobile device to a Wi-Fi and will then perform a step and make a http post request to the database.	Mobile unit test: • Verifies that the mobile application does an HTTP request when the device is connected to a WI-FI access point. Backend unit Test: • Verifies that the data received from mobile	Mobile tester: Verifies that the application does an HTTP request to the server and the connection was successful. Backend Tester: The tester will verify that the data recorded on the

		application matches with the recorded data from the mobile application. The backend and mobile team will have code walkthroughs and reviews to verify that they the unit tests are fulfilling sending data through an access point according to the specifications of the storing tracking information requirement.	phone matches with the data stored on the database table scheme. After the tests are completed the tester will do a report analysis of the errors that they found and will be reviewed by the development team.
Distance Accuracy	We will give a user a new starting location to start the application at. We will instruct him/her to move 1 meter and see if the new current location is within ± 1 meter.	Mobile unit Test: Create a unit test that starts at Location A and moves to location B and then verifies the user new location is within our 1-meter accuracy requirement. The mobile test team will perform a code walkthrough that explains how their test meets the specification of the distance accuracy requirement. The review team will check documents and files to ensure that the code meets our coding standards.	Mobile tester: Verifies that the movement from Location A to Location B is within 1-meter accuracy. After the tests are completed the tester will do a report analysis of the errors that they found and will be reviewed by the development team
Delay Accuracy	We will use a timestamp when the mobile application has gathered all the sensor data and starts to perform the smoothing/prediction algorithm while the software creates a 5 second delay. After the delay is completed the software will create another time stamp and compute the time difference.	Mobile unit test: • Create a unit test that takes the time stamp after the data has been collected and after the x second delay has been completed. Then compute the time difference and compare it with delay requirement. The mobile test team will perform a code walkthrough that explains how their test meets the specification of the delay accuracy requirement. The review team will look over the documents and files and confirm it meets our coding standards.	Mobile tester: Verifies that the delay time of the application is within the 5 second thresh hold by running the test script and report the results. After the test is completed, the tester will complete a report analysis of the errors that were found. The development team will review these documents.

Drift Accuracy	We will start a user at a starting point by Durham. We will then have the user move 2 meters in any direction and record the position and then do a computation of the possible positions and see if the user is within the 1 meter of the computed location.	Mobile unit test: • Create a unit test that takes the starting position of the test plan and then computes all the possible new position paths the user could had taken and then compare those position with the user's current position and report the results. The mobile test team will perform a code walkthrough that explains how their new positions will meets the specification for the drift accuracy requirement. The reviewers will look at the code to see if it meets our coding standards.	Mobile testers: Verifies the users new position is within 1-meter of the possible computed positions that was computed. After the test is completed, the tester will complete a report analysis of the errors that were found. The development team will review these documents.
HoloLens Monitoring	The HoloLens will receive the new recorded location data from the Rethink database. It will then render the users new position in the augmented reality map. The user that is wearing the device should see the avatar move from the user's current position to the new position that was received.	AR unit test: • Create a unit test that takes the newly recorded data from RethinkDB and calculate the new position that the avatar transform should read after unity renders the new position. Compare the avatar's new position with the computed position and report the results. The AR team will perform a code walkthrough that explains how their new computed transform position meets the specification of the HoloLens monitoring requirement. The reviewers will look at the code and see if it meets the unity coding standards.	AR testers: Verifies the new transform position the avatar moves to in the AR map is the same as the newly computed position. After the test is completed, the tester will complete a report analysis of the errors that were found. The development team will review these documents and address the software errors that were reported.
Monitor Accuracy	We will create a timestamp and send a location packet by using a http request to the database. When the database	AR unit test: • Create a unit test that take the recorded timestamp from the mobile application and create a new timestamp.	AR testers: Verifies the that recorded location is being displayed on the map in AR within the 10 second threshold from the time it

	has received the packet, it will create another timestamp. The database will then send the location packet to the website and HoloLens software. The HoloLens and website will then create a new time stamp and calculate the difference and see if it is within 10 second threshold.	Then compute the difference and compare it with the 10 second requirement and report the results. Web unit test: • Create a unit test that takes the recoded timestamp from the mobile application and creates a new timestamp. Then compute the difference and compare it with the 10 second requirement and reports the results. Backend unit test: • Create a unit test that takes the recoded timestamp from the mobile application and create a new timestamp. Then compute the difference and compare it with the 10 second requirement and reports the results. The AR, Web, and Backend will perform a code walkthrough that explains how their time calculation meets the monitor accuracy requirement. The reviews will look at the code and see if it meets the coding standards.	was recorded on the mobile device. Web testers: Verifies the that recorded location is being displayed on the website map within the 10 second threshold from the time it was recorded on the mobile device. Backend testers: Verifies the that recorded location is being displayed on the website map within the 10 second threshold from the time it was recorded on the mobile device. After the test is completed, the tester will complete a report analysis of the errors that were found. The development team will review these documents and address the software errors that were reported.
Bluetooth Sensor	The android device will be paired with the beacon so that they our connected to each other. We will set the beacon at a location x inside the jobsite. When the android device receives a signal from the beacon, it will send a location update to the android	Mobile unit test: • Create a unit test that takes the updated the location from the beacon and then compares it with the current location that is stored on the device. It should compare these two and report the result. The Mobile team will perform a code walkthrough that explains how their comparisons meet the	Mobile tests: The user will stand within 1 meter of the beacon to trigger the location update. They will then verify that the mobile devices current location is getting updated. After the test is completed, the tester will complete a report

	device.	Bluetooth sensor requirement. The reviews will look at the code to make sure that the developers are meeting the coding standards.	analysis of the errors that were found. The development team will review these documents and address the software errors that were reported.
Android Low Battery Notification	The android device battery must be drained to 10% of its total capacity. Then we will run the application on the android device.	Usability mobile test: • Let the device get below 10 percent of battery and display a low battery notification. The Mobile team will perform a code walkthrough that explains how their notification meets the Android low battery notification requirement. The reviews will look at the code to make sure that the developers are meeting the coding standards.	Mobile tester: Tester should see if the low battery notification displays when the devices battery is below 10 percent. After the test is completed, the tester will complete a report analysis of the errors that were found. The development team will review these documents and address the software errors that were reported
HoloLens Low Battery Notification	The HoloLens battery must be drained to 10% of its total capacity and then run the application on the device.	Usability AR test: • Let the device get below 10 percent of battery and display a low battery notification. The AR team will perform a code walkthrough that explains how their notification meets the HoloLens low battery notification requirement. The reviews will look at the code to make sure that the developers are meeting the coding standards.	AR tester: Tester should see if the low battery notification displays when the devices battery is below 10 percent. After the test is completed, the tester will complete a report analysis of the errors that were found. The development team will review these documents and address the software errors that were reported

Table 1: Functional Requirements Test Plan

3.4 NON-FUNCTIONAL TESTING

Non-Functional Requirements	**Test Plan**	**Verification**	**Validation**
Battery Lifecycle	A user will turn on the mobile device and launch the application as a background process and monitor the battery and CPU usage throughout the 8-hour work day.	Usability Mobile test: • The user will launch the app and monitors the mobile application every hour and report the results. The Mobile team will explain to the team why the monitoring test meets the Battery Lifecyle requirement.	Mobile tester: The user will launch the mobile application and monitor the android profiler every hour for 8 hours. After the test is completed, the tester will complete a report analysis of the CPU usage and if the phone lasted for 8 hours. The development team will review these documents and address the software batter issues if there are any.
GPS Sensor	We will turn off the GPS location on the mobile device and then create a preplanned walking plan around Durham. We will record their path walked and compare it with the preplanned walking path and show the results.	Mobile unit test • Create a unit test that takes the preplanned route and compares it with the user's route and reports the results. The mobile team will perform a code walkthrough that explains how their unit test meets the GPS sensor requirement. The reviews will look at the code to make sure that the developers are meeting the coding standards.	Unit test where you create a walking path and then perform the walk and compare the expected path with the actual path and return result
Look and Feel	We will place the mobile device inside of the armband and wear the device for 8 hours.	Usability Test: • We will create a test that requires the tester to wear the mobile device for a normal work day and report the results.	Mobile Tester: The tester will wear the device around his arm for 8 hours and then report the comfortability to the development team.

Environment	We will launch the HoloLens application at the clients facility inside of their conference rooms.	Usability test: • We will create a test that requires the tester to wear the HoloLens inside of the clients facility.	AR Tester The tester will go to the client's facility and see if their facility has the proper environment for running the HoloLens application. The tester will complete a report analysis of their findings to the HoloLens development.

Table 2: Non-Functional Requirement Test Plan

3.5 Process

Our process for testing our product will be someone hold the phone with the screen facing out in front of them. The app will run in the background and no other apps will be running. The person will then walk in Durham along the pre-laid out course. The same course will be done multiple times so that we can compare the precision with our app.

3.6 Results

By the final results, we need to be able to use the phones internal sensors to approximate a users location. Every known user will have their information saved in a database where we will then be able to pull that information and display it back in real time using Unity. Unity will display a similar environment where the top-down use would be able to explore daily activities.

4 Closing Material

4.1 Conclusion

Summarize the work you have done so far. Briefly re-iterate your goals. Then, re-iterate the best plan of action (or solution) to achieving your goals and indicate why this surpasses all other possible solutions tested.

4.2 References

1. "UNITED STATES DEPARTMENT OF LABOR." *Occupational Safety and Health Administration*, www.osha.gov/oshstats/commonstats.html.
2. "Getting Started." *Frequently Asked Questions - RethinkDB*, rethinkdb.com/faq/.
3. "The Top 7 Free and Open Source Database Software Solutions." *Capterra Blog The Top 7 Free and Open Source Database Software Solutions Comments*, blog.capterra.com/free-database-software/.
4. "The Pros and Cons of 8 Popular Databases." *KeyCDN Blog*, 20 Apr. 2017, www.keycdn.com/blog/popular-databases.
5. "Getting Started." *RethinkDB 2.1.5 Performance & Scaling Report - RethinkDB*, rethinkdb.com/docs/2-1-5-performance-report/.

4.3 Appendices

Any additional information that would be helpful to the evaluation of your design document.

If you have any large graphs, tables, or similar that does not directly pertain to the problem but helps support it, include that here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc. PCB testing issues etc. Software bugs etc.