**WATECHPARK**

SMART Parking Lot System

**Vikas Sharma, George Alexandris, Elias Sabbagh**

|  |
| --- |
|  |

January 15, 2020

Computer Engineering Technology

Status

/1 Hardware present?

/1 Title Page

/1 Declaration of Joint Authorship

/1 Proposal (500 words)

/1 Executive Summary

# Declaration of Joint Authorship

We, Student A (Vikas Sharma), Student B (George Alexandris), and Student C (Elias Sabbagh), confirm that this work submitted is the joint work of our group and is expressed in our own words. Any uses made within it of the works of any other author, in any form (ideas, equations, figures, texts, tables, programs), are properly acknowledged at the point of use. A list of the references used is included. The work breakdown is as follows: Each of us provided functioning, documented hardware for a sensor or effector. Student Aprovided VCNL4010 Proximity Sensor. Student B provided IR Beam Sensor. Daniel O Donnell provided LSR Camera Sensor/2 Stepper motors. Due to unexpected circumstances, Daniel’s involvement with the project was passed to Elias Sabbagh joining the group during the Winter 2020 semester, to ensure successful progression of the project. In the integration effort, Student A is the lead for further development of our mobile application, Student B is the lead for the Hardware, and Student C is the lead for connecting the two via the Database.

# Proposal

**WatechPark**- SMART Parking Lot System Proposal

**From:** Vikas Sharma, N01160135

**Discipline:** Computer Engineering Technology

**Date:** January 15, 2020

# Background

This document will outline the software portion of the project in CENG 319 that will be coupled with hardware in CENG 317 for the final integration in CENG 355. Our project is going to be on a SMART parking lot system.Many busy parking lots are often plagued with congestion, drivers competing to find a spot by cruising around and visually finding spots. This is inefficient, time consuming where productivity is lost for consumers and businesses. The system we will be developing will address payment for parking, capacity management and location finding following an IoT approach using hardware and software.

# Problem Statement

The problem being addressedincludes, time spent searching for a parking spot, increased capacity levels during peak hours. This project is focused on solving these issues by connecting consumers to parking lot owners and providing parking services by using a more convenient, simpler method to retrieve parking lot data seamlessly.

# Methodology

Phase 1: Hardware Design/Build

The small physical prototypes that we build are to be small and safe enough to be brought to class every week as well as be worked on at home. In alignment with the space below the tray in the Humber North Campus Electronics Parts kit the overall project maximum dimensions are 12 13/16" x 6" x 2 7/8" = 32.5cm x 15.25cm x 7.25cm.Keeping safety and Z462 in mind, the highest AC voltage that will be used is 16Vrms from a wall adapter from which +/- 15V or as high as 45 VDC can be obtained. Maximum power consumption will not exceed 20 Watts.

Phase 2: System Integration/Connection

This phase will be completed during the final semester of the Computer Engineering Program. The work gathered from both software/hardware courses will be combined and integrated for the final capstone project.The development platforms we will be working with is Android Studio 3.5.2, Raspberry Pi 4 ModelB, and Google Firebase database.The mobile application provides key functionality to allow consumers to access parking lot data, view sensor/effector information specific to a location and choose the best parking space during different peak hours of the day.The VCNL4010 Proximity sensor will be used to detect the status of a given parking space at a specific time of access. The IR Beam Sensor will control the gate opening/closing and detect the presence of a vehicle near/far away. The LSR Camera sensor will be used for valid license plate recognition. The 2 stepper motors will control the gate and allow entry/exit based on the sensor data, and status of the lot.

Phase 3: Final Demonstration to Potential Employers

At this stage, we will demonstrate our 2 semester’s worth of work to be assessed.Our project description/specifications will be reviewed by, Mike Wrona, ideally an employer in a position to potentially hire once we graduate.

# Hypothesis

This project is focused on providing a solution for managingparking lot data, providing a less time-consuming experience with a simple, intuitive interface. This is an opportunity to showcase our knowledge and understanding to build a collaborative effort for an industry sampled IoT project. I request approval of this project.

# Executive Summary

In retrospect, this document outlines both the hardware and software aspects of the project. This project intends to build an IoTdesign that would help support industry related issues such as capacity management, location-finding by finding ways to reduce the time spent manually searching for parking spots. This document aims to provide insight into the design, development, testing phase of our SMART parking lot system project. In collaboration with our partner at ParkingBoxx, we have gathered our ideas to create a simple, intuitive and user-friendly platform for consumers within the market.

Our product aims to provide the essential needs for both consumers/businesses to view and manage parking lot data. In terms of market use, we believe through the project we will build a product that can be offered from an industry standpoint as well as be marketable to other fields of interest. Through the development of this product, we wanted to reach as many demographics and be able to provide aninexpensive and reliable platform where parking lot information can be retrieved at a glance.We offer users with the ability to be able to add/manage cars, view parking lot data, make on-the go reservations for parking passes, accessible via an online database to send/receive information in real-time, all built-in with a simple, effective interface. Due to these reasons, we believe it will be ideal to be considered to be hired by an investor for employment. This will be an extraordinary opportunity for us to be able present our work, knowledge and skills to promote our product from a marketing perspective.

# 1.0 Introduction

This report will outline the development and integration ofour final capstone project as part of the Computer Engineering Technology program at Humber. The individual team contribution for this project goes towards George Alexandris, Vikas Sharma, and Elias Sabbaghall 3 of whom were extensively involved with bringing the project to life. The focus was to implement an IoT (Internet of Things) design,where software and hardware interaction would be vital to address an industry related issue and help solve real-world problems. This project consists of a SMART parking lot management platform, which allows consumers/businesses the ability to manage and monitor parking lot data through real-time progression. The goal, being to address problems arising in the parking industry specifically in terms of capacity management, increased manual interference, and the lack of location-finding near/far from an area. The product looks towards determining the challenges in the parking industry today, and provide a platform to navigate to a parking space quicker, and in a more efficient manner. This includes, managing parked users, or monitoring the status of a lot at a given time.The main objective of this undertaking is to provide a more efficient and reliable platform to aid with parking scenarios. In particular, for the purpose of the consumer demographic who may be searching for an alternative parking lot management system.Our focus was to develop a platform, that would be the gateway to support consumers with finding the best parking space during any time, any place or anywhere in the world.

# Project Schedule

The following is an overall breakdown of our work schedule for the duration of the entire project, and the two consecutive semesters:

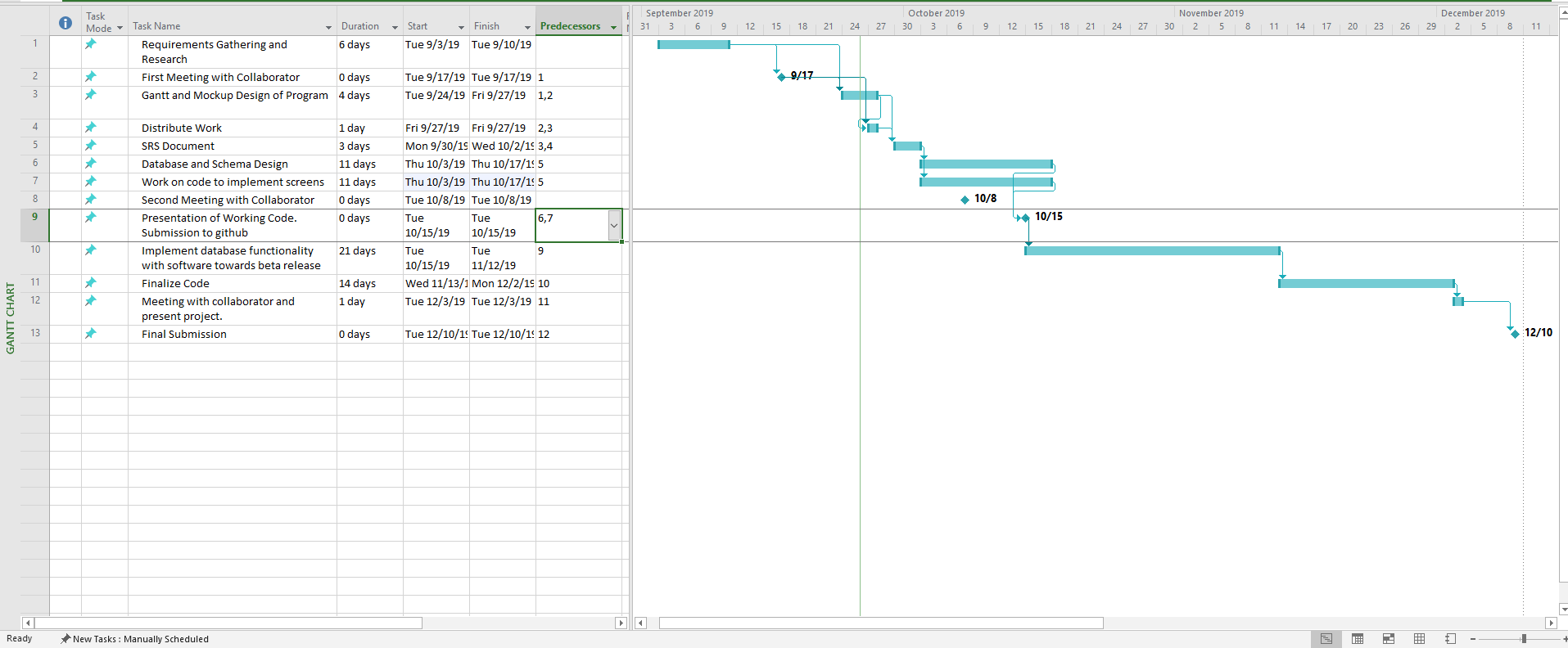
****

Figure 1 - Gantt Chart (Software)

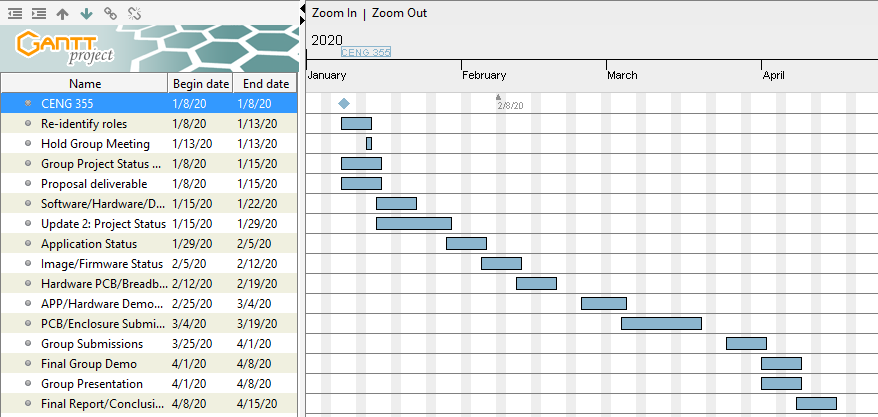


Figure 2 - Gantt Chart (Final Breakdown)

In terms of overall scheduling purposes, the project requires approximately fifteen weeks to complete during the Winter 2020 semester of the Computer Engineering Technology program. During the start of the final semester, the work done from the previous fall 2019 semester from CENG 319 and CENG 317 was carried over to complete the final integration. Most of the software development portion of the mobile application, and the connection to the database was done during the fall semester. Our group also ensured we were on the right track, by ensuring at least one of our team members can connect their sensor data to the database to send/display data to the platform during the fall semester. The core amount of work was accomplished towards the end of the Software Programming course, as well as in the Hardware Production Technology course. Our team followed our planned schedule efficiently throughout the fall semester, and carried the workload into the Winter 2020 semester. The final semester needed the group to finish implementing the sensor data to the Firebase database to be able to read/display data from the actual hardware. At this time, our team will extensively focus on completing the mobile application, and adding its intended features. As well as, re-designing our PCB boards to accumulate all three sensors from each team member into a final PCB design. The final semester will be used appropriately to ensure successful completion, and a polished end product.

## 1.1 Scope and Requirements

This report will address the key fundamentals of the project, including the software/hardware aspects of the design/specification and the final integration between these applications. It will touch on how the hardware (sensors/effectors) communicate and interact with the software application to allow consumers to track parking lot data. Further details will be examined in this document, in regards to the hardware sensors used, the main objectives behind choosing the type of hardware and how it correlates to the application being developed. Along with, describing the database structure used to push and retrieve data from the hardware to the mobile application, and vice-versa. In terms of the scope of the project, there were instances where the planned outcomes from the initial stages of the development period had to be modified or re-considered. This includes, having some minor design/features changes and removals mainly due to time constraints or software limitations, to further increase productivity and meet the overall requirements. Some of these changes included, a major overhaul of the main Home screen to provide a smoother, sleeker interface for user interaction, removing Google API login integration, removing the search function from the mobile application after much consideration. A major change, occurred during the Winter 2020 semester, where our development team decided it was in our best interest to not move forward with the development of the business aspect of the application. This feature would have allowed users to track their daily progress and set goals to reach in the mobile application from a marketing standpoint. The team decided to improve on the existing consumer functionality instead, with implementing the proposed camera sensor functionality, and stepper motor gate control. Due to this reason, these features had to be omitted as a result of not meeting the plan, and the overall extent of the project had to be re-evaluated and worked on. Our team followed an agile methodology/procedure where work would be committed and evaluated on a daily basis to ensure the team stays on track. Development was split into distinct sprints of software coding/hardware testing to ensure both applications are met within the due time. During this time, coding was separated into blocks of development periods where the team focused on completing a single feature, and aiming towards further progression from both sides. The limitations provided our team with an outlet, to focus on the main goal/priority and cut down on excess amount of work that can be evaluated later in terms of a marketing perspective. For example, one of the ideas the team came up with was introducing the business aspect through an online website where users would be able to view financial, user information along with providing identical software/hardware interaction. This SMART parking assist platformprovidesreal-time proximity measurements, lot detection methods with use of the hardware elements. Such as, the entry/exits of vehicles, detecting parking space movement, gate control.

## Development Platform Specification

The following are the specifications of the software side of development. In terms of application use, consumers only need to know how to use a smartphone device. No technical expertise is needed as the platform we develop will be simple to use and gain a grasp of. The following are the list of software requirements vital for the platform to operate as intended:

* Android Studio 3.5.2 development platform
* Java (coding language) used for mobile application
* Software must have bilingual capabilities for English/French language integration
* Internet connection (Wi-fi) is needed to access the mobile application, and its main functions.

## Hardware Specification

The following is a list of hardware requirements needed by the user to operate the application, and its functions:

* Raspberry Pi 4 Model B (CPU platform to process sensor data)
* Must support at least 2GB of storage, RAM
* Embedded CPU (Raspberry Pi) device will always need a connection to the server for the purposes of authenticating users, and receiving data.

## Android Device Specification

* Must be running Android OS on mobile system
* Mobile APP will run only on Android devices
* API 21(Android Version 5.0 Lollipop) and above (supported roughly over 80% of the Android population)

## Database Specification/Protocols

* Google Firebase database for storage purposes, push/retrieve real-time data from sensors to mobile application, and vice -versa.
* User Authentication

**Protocols:**

* HTTPS/SSL Encryption end to end communication
* TCP/UDP Connection
* Wifi/Cellular connection

**Server-Side:**

* Email: SMTP (Simple Mail Transfer Protocol)
* Data transfer rates must be capped to not utilize an excess amount of data, depending on connection type (size must be compressed).

Report

/1 Hardware present?

/1 Introduction (500 words)

/1 Scope and Requirements

/1 Background (500 words)

/1 Referenc

# 2.0 Background

In the industry today, there have many occurrences where parking in general has become a hassle for city residents and parking lot owners. This includes, not possessing the right tools to manage capacity when a parking lot is full, where drivers are struggling to find the best spot to park their vehicles. This can lead to dis-satisfying scenarios, where drivers are unaware of their surroundings, before even entering into the space. Due to this reason, it can lead to congestion in major traffic centric cities, with drivers competing to find a spot. This can be time-consuming, inefficient where productivity is lost for consumers and businesses. This project is focused on helping reduce the impact of this cause, by developing a system that will address payment for parking by taking an advanced and modern approach towards capacity management, and real-time information gathering to keep consumers up to date with their daily occurrences.

The group would like to thank Mike Wrona, installation manager, of Parking Boxx who provided support for this project. The project is a SMART parking lot system that incorporates a phone app to manage a user’s tickets, account, and where to park in the parking lot. The idea of this project came up when the group realized that we can develop an easier way to find parking spots, by connecting all the spots to a parking app. We thought about creating an IoT parking lot that can connect to a database and update the database with information about its open/closed parking spots. It will be able to send and retrieve information about the parking lot. The mobile application will allow users to connect to the database and manage user accounts and payments for their tickets. Examples of some existing platforms are Indigo, BestParking, EasyPark and ParkWhiz. These platforms have reservation capabilities, on the go parking with mobile or web application. What we are going to do differently from these companies is to integrate sensors to help users navigate to a parking spot.

One of the parking companies we looked at is EasyPark(EasyPark, 2016). EasyPark offers monthly payments for its customers to park in the EasyPark parking facilities. The goal for EasyPark is “provide safe, clean, friendly, convenient and affordable parking to the Greater Vancouver community” (EasyPark, 2016). EasyPark offers a phone app (EasyPark, 2016) called EasyPark Parking as well that allows it users to view parking available at its facilities. You can pay for parking on the phone app (Google Play, 2020). The parking app allows its users to register, find EasyPark parking lots close to you, keeptrack of how long and where your car is parked in the parking facility. Payment is done through the mobile platform with a much simpler method of operation.

We looked at the other parking companies that were mentioned such as Indigo, BestParking, and ParkWhiz. In terms of overall design and interface almost each application had full resemblance to the previous. Indigo offers a map that allows users to choose which parking lot they want to park at. Users can book their parking ticket in advance, reserve the spot for thedesignated parking facility owned by the company they want to park at, and they can see the rates of the parking lots (Park Indigo Canada Inc, 2019). Apps such asParkWhiz, offers the options to add cars for verification, pay for monthly parking on the app and their ticket (ParkWhiz, 2019).

# 6.0 References

*EasyPark*. (2016). Retrieved from EasyPark Mobile Parking App: https://www.easypark.ca/products-services/mobile-parking-app

*EasyPark*. (2016). Retrieved from City Parking Vancouver, Public Parking, EasyPark Mission: https://www.easypark.ca/about-easypark/mission

*EasyPark*. (2016). Retrieved from History: https://www.easypark.ca/about-easypark/history

*Google Play*. (2020). Retrieved from EasyPark Parking - Apps on Google Play: https://play.google.com/store/apps/details?id=ca.easypark2.app&hl=en

Park Indigo Canada Inc. (2019). Indigo. Retrieved from https://ca.parkindigo.com/en

ParkWhiz. (2019). Find and Book Parking Anywhere. Retrieved from https://www.bestparking.com/

# 

# 3.0 Methodology

The following section will outline and extensively explain our method of operation for the duration of this project. This includes the steps taken to ensure we meet the correct criteria in reaching the end goal. Following an agile methodology, we will keep working diligently to get each part of the application up and running. In the event of any outstanding circumstance, we will review the design and re-assess which parts are causing the issue and work around from there. Development and testing phases of the project will be split into distinct sprints of development. The next few branches of this section, will go into detail in regards to the required resources. Initially, focusing on the parts/components/materials used to complete the project, the manufacturing process for PCB and enclosure, tools/facilities utilized, shipping/duty/taxes and lastly touch on the working time versus lead time needed to complete the final hardware integration for the project.

## 3.1 Required Resources

Report

/1 Parts/components/materials (500 words)

/1 PCB, case (500 words)

/1 Tools, facilities (500 words)

/1 Shipping, duty, taxes (250 words)

/1 Working time versus lead time (250 words)

### 3.1.1 Parts, Components, Materials

## Parts and Components

This project consists of a variety of parts, components and materials which were utilized during the development, testing and final integration phase of our parking application. These components were essential to different features/functionalities of the parking platform and vital in terms of performing each functionality of our application. The project made use of three primary hardware sensors/effectors. These included, the VCNL4010 Proximity Sensor, IR Break Beam Sensor (Receiver/Transmitter), USB Camera Sensor (YoLuke HD Webcam). The CPU Broadcom Development platform chosen for the project is the Raspberry Pi 4 Model B. The Raspberry Pi 4 is the fastest, latest piece of technology that is capable in providing multi-support interfaces, at much higher speeds. For large data processing, and data heavy projects such as this extensive project it is the suggested microcontroller to do the job. Firstly, for the VCNL4010 Proximity sensor, we made use of the following hardware components: 3 (220) ohm variable resistors to limit the current flow through the circuit, Common Anode RGB(red,green,blue) LED to detect and react to changing states in proximity readings/parking lot status, 20x2 stackable header, to mount the PCB board to the Raspberry Pi, 6 pin stackable header, to mount the sensor to the soldered PCB board, and finally a set of jumper wires(female to male) for the breadboard testing portion. For the IR Break Beam sensor, the components used included the following: 4.7K ohm resistor, 1.0K ohm resistor, and a red LED to detect movement of an object, in this case for our application it would detect if a vehicle is in the way of the gate to allow entry/exit as a result, breaking the beam. The Raspberry Pi 4 Model B microcontroller will provide the key functionalities for the VCNL4010 Proximity sensor, IR Bream Beam sensor, and the USB Camera sensor alongside the 2 stepper motors used for parking lot gate control.

Furthermore, other components used include a power supply alongside an Ethernet to USB (Universal Serial Bus) adapter cable. These components will provide the interface to connect to the Broadcom Development platform through either remote desktop connection or via VNC Viewer connection to further develop, integrate the sensor/effector functionalities. An SD card will be used to provide the imaging/firmware configuration settings from an Ethernet connection, using the universal serial bus to Ethernet adapter.

Along with these essential parts, multiple jumper wires were used to connect to the GPIO (General Purpose Input/Output) pins of the Raspberry Pi device. The VCNL4010 Proximity sensor, and the IR Beam sensor will be used under a I2C interface. Using a 40-pin stackable header for the GPIO pins of the Raspberry Pi, the VCNL4010 Proximity sensor and the IR Beam sensor will be connected to the Raspberry Pi using its general-purpose input/output pins corresponding to each circuitry design for each sensor. These sensors will then communicate based on the surrounding environment, and its conditions. For example, detect whether a vehicle is in the way of the gate or a space is being occupied/ statuses of the lot. These components will then interface through the real-time database setup through Firebase database to push/retrieve parking lot data/statuses actively. The USB Camera sensor will be used for license plate recognition and capture an image of a valid license plate at entry. The data captured will be sent to the database for further examination. More details in terms of the mobile application, and how the data is presented will be explained in the Development Platform section under the Mobile Application branch in this report.

Additionally, three stackable headers were needed to mount all three sensors to the PCB (printed circuit board) for the final PCB design process, further detailed in the Manufacturing part of this section. Our team focused on retrieving parts and components that were feasible and inexpensive to our planned budget and would provide the best quality of performance. The parts made use of for the project included the presented ones above, including the added components needed to successfully test and implement each sensor.

## Materials

The materials to be used as part of the project, includes the following: laser-cut acrylic for the final enclosure design, to protect all three sensors from potential harm/damage prepared in a suitable housing environment. Other materials included making use of a breadboard to test the hardware sensors and the corresponding circuits that were built. Some other materials used included, the printed circuit board constructed and etched through sheets/layers of copper foil, and glass fiber material (fiberglass epoxy resin). These dielectric materials, such as glass, fiber was used as an insulating layer to create the PCB from the ground up through both the top/bottom layers of the printed board. Through the laser-cutting work achieved for the enclosure, acrylic material will be used alongside plastic housing to support the hardware and provide panels for the safety of the Raspberry Pi platform, printed circuit board, and its sensors. Acrylic material is lightweight, superior in quality and provides the best resistance for each sensor housed in the case. Similar to fiberglass material, the enclosure will be able to offer a clearer insight into the different components and sensors/effectors and provide high protection for all hardware components assembled. All-inclusively, these parts, components and materials gathered will be important in the integration effort to ensure successful completion of the project, following our planned methodology and strategy.

### 3.1.2 Manufacturing

### PCB/Case

In terms of the manufacturing process, most of the components gathered were ordered through the Amazon, BuyAPi, and Adafruit industrial companies. The printed circuit board designed was also manufactured and produced by the prototype lab, free of cost.

The final PCB and enclosure designs were all discussed in this semester including completed PCB and case designs from last semester. Last semester in the Hardware Production Technology course we had to design PCB’s and enclosures for the sensors we required for the final Parking Lot project in this semester. In this part of the report, we will take a look at the previous designs of the PCBs and cases being used in the project, and how they will be integrated into one final design.

The IR Break Beam Sensor project was designed by George Alexandris. The IR Break Beam sensor had a few design changes over the Hardware course last semester. The first version of the PCB design for the IR Break Beam Sensor did not fit onto the Raspberry Pi 4 Model B. The size of the first version was too large to be put onto the Raspberry Pi. The second size of the PCB was smaller and was able to be fit onto the board successfully.

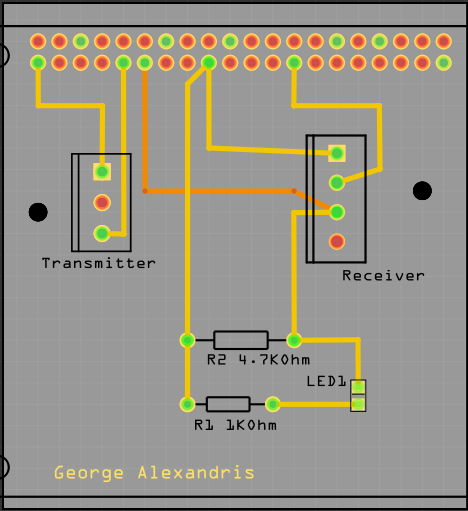
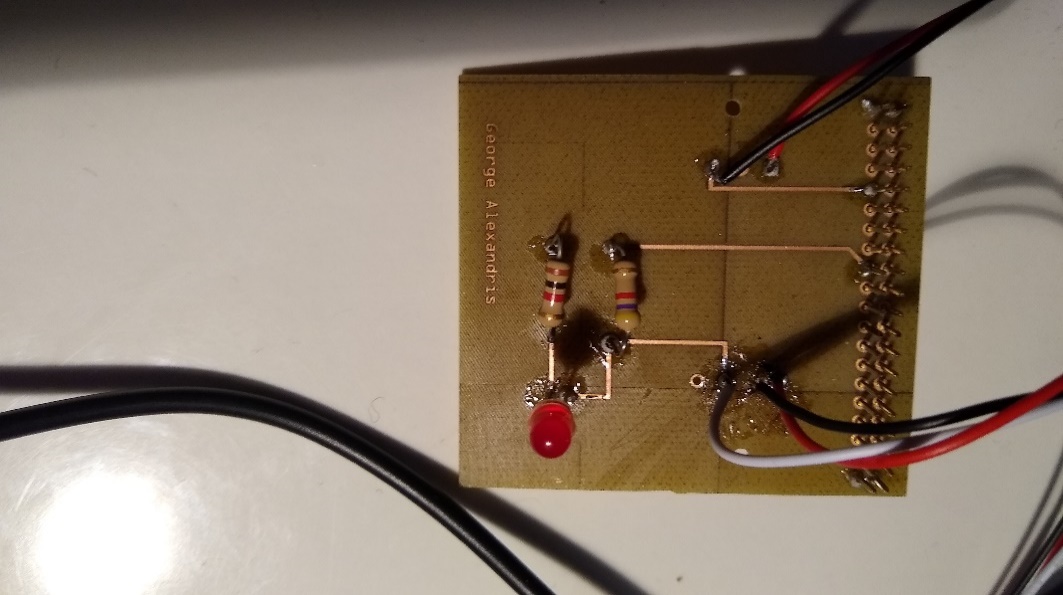
Below are images of the second version of the PCB in Fritzing and the printed PCB: 

Figure 3: Printed PCB from Fritzing file Figure 4: Image of PCB Design in Fritzing

PCB’s were all designed in a free program called Fritzing and were printed in the Prototype Lab at Humber College. IR Break Beam sensor is in two parts one is the transmitter which as two connections directly to 3.3V power and ground on the board. The other part is the receiver which has three connections one for 3.3V and ground directly to the Pi while the third connection is to one of the GPIO pins. The GPIO pins can be setup for either input or output but the third wire from the receiver is a digital output so the GPIO pin it is connected to is set up as an input to read the value from the output wire.

The enclosure design for the IR Break Beam sensor project went through many design changes as well. The design for the case had to consider the size of the Raspberry Pi and the PCB for the IR Break beam sensor. The cases were designed in CorelDRAW and laser cut in the prototype lab. Early designs for the case of the IR Break Beam Sensor Project did not fit the sensor and the Raspberry Pi together in one case. The PCB of the IR sensor was long and the early design for the enclosure was too small in length for the case to fit. The measurements were taken to adjust the size of the case so the PCB and Raspberry Pi can fit together. The new design was longer for the PCB and Raspberry Pi to fit and the final design is shown below.

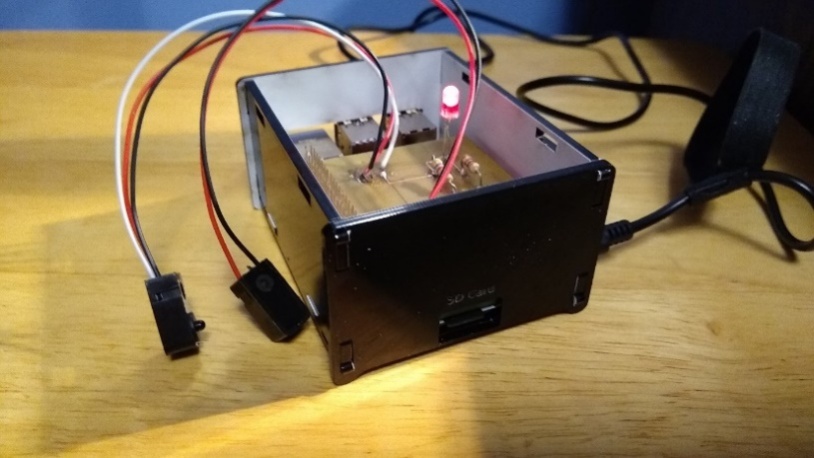


Figure 5: Final Case Design for IR Break Beam Sensor

The second sensor being used for the project is the VCNL4010 Proximity/Light Sensor. The PCB and case for this sensor was designed by Vikas Sharma. This step required use of the Fritzing software, which is an open-source application that allows users to visually create schematic, breadboard designs and finally incorporate their designs into a printed circuit board to be sent for etching and laser-cutting services. Over the course of the semester, Vikas had many different design flaws for the PCB. This was due to connections to the top layer of his PCB. In previous designs there were overlapping connections for the top/bottom layers, so Vikas had to redesign his PCB many times to not overlap his connections. The connection for the GPIO pins has to be on the top layer because of soldering the PCB to a 40-pin connector for the Raspberry Pi, it is not possible to solder the bottom layer.

To solder the PCB board, you would require a lot of focus, and attention, because any accidental procedures/connections going wrong can cause you to redesign the whole board. Vikas was able to luckily not have to go through that process, while soldering. For his design, there were 4 vias that needed to be soldered. You must thread a single thin wire through the holes, solder it and then cut off the remaining excess wires. The same process, was taken for the resistors where Vikas set up the resistors in their place and had to solder the 2 sides. Then, cut off the excess wire. While soldering the LED, Vikas had to be very cautious as the connections were designed really close to each other, so a lot of focus was needed here. Initially, there is a 6-pin header that comes with the sensor that needed to be soldered. Vikas also soldered the 40 pin GPIO pins on the PI, to keep the header stable while soldering and to create a sturdy connection. After soldering, the board should be ready to be tested, and should look like the screenshot shown on the left below.

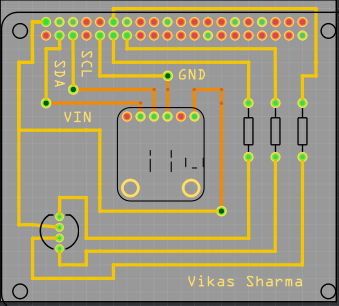
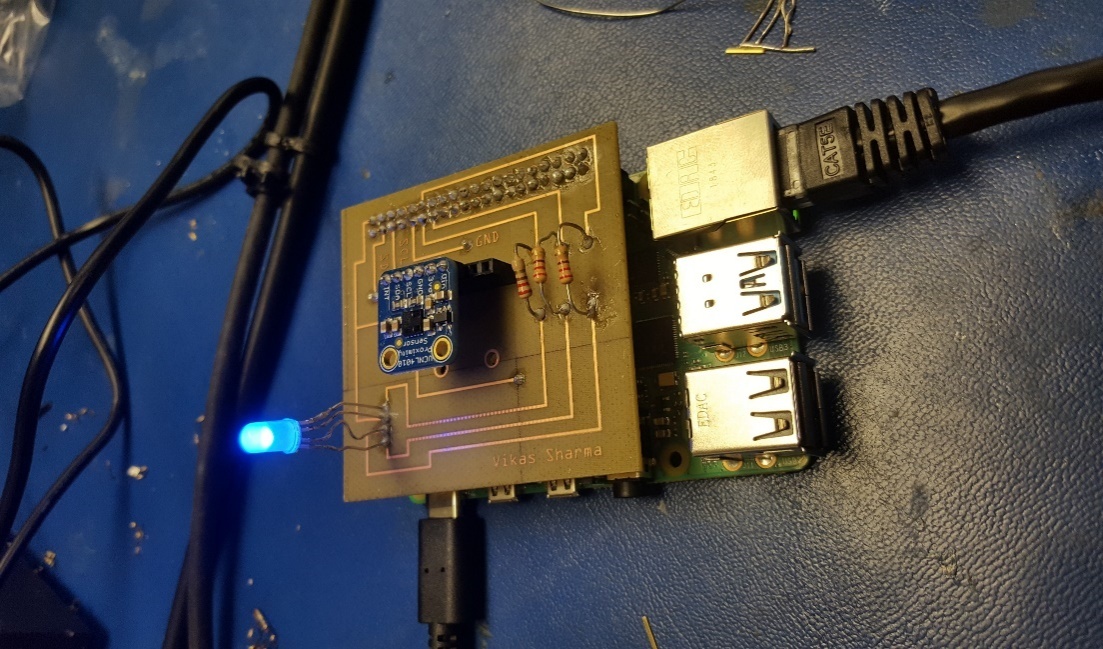


Figure 6: PCB final Design for VCNL4010 Figure 7: PCB Design for VCNL4010 in Fritzing

Once the PCB board testing phase was successfully complete and Vikas was retrieving/sending data to Firebase and the mobile application, he moved on to the enclosure design . For this, he decided to use the CorelDraw software as it seemed like the best choice to design the case the way you want. Vikas went with a acrylic case design, as the case is to prevent/protect your Raspberry Pi and sensor from potentially being damaged. While making the case , you have to be aware of the dimensions of the Raspberry Pi(height,width from all sides). For the design, he had to make some changes due to using the Raspberry Pi 4, some ports are arranged differently. Such as the LAN(ethernet port), and the 2 micro HDMI ports. For this, Vikas decided to make a wider port/hole to fit both mini HDMI ports into one. Vikas ran into a issue where the hole for the power port was cut too short, for this he had to shave off some excess acrylic to get the hole to be wider, to fit the power cable into the port. Vikas was able to fit it perfectly in the outcome. For the design, Vikas kept the top of the case open with a medium-sized hole, as there needed to be a object in the way of the sensor or far away for it to display the readings.Once all the requirements for the dimensions of the openings were met the final case design was complete.

14

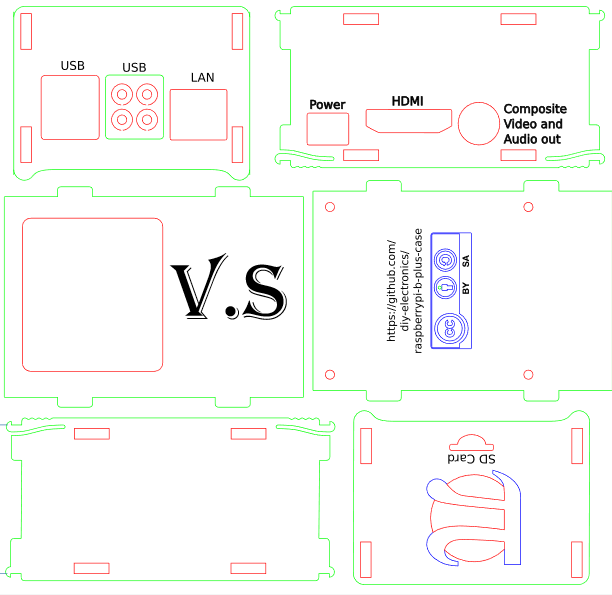


Figure 8: Final Case CorelDRAW design for VCNL4010 Figure 9: Printed Case for VCNL4010

The last sensor, Yoluke HD Camera Sensor, did not have a design for an enclosure and its PCB since it is just a USB camera. The servo motor however, does have a PCB design and will be shown below.

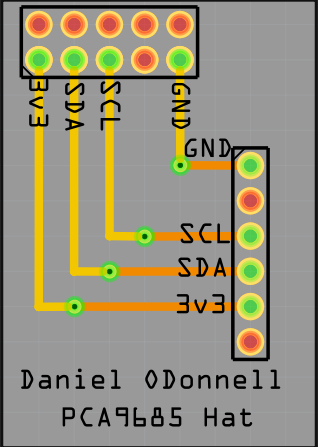


Figure : Fritzing PCB design for Servo motor

Taking into account the previous semesters work, the initial plan which will be immediately targeted is to combine all three sensors into a single printed circuit board entity. This will be done to ensure progression of the project, and meet the requirements for this semester in completing the capstone project. After testing each sensor, and connecting it to the database and the mobile application we will design a new PCB using the Fritzing software. This will enable us to work with all three sensors using only one Raspberry Pi 4 Model B Broadcom Development Platform, connected to a single PCB in the final stage. Elias will be responsible for helping with the hardware implementation and design the final PCB through the open-source software. Along with gaining further support from Vikas in working with the VCNL4010 Proximity sensor, IR Break Beam sensor, and the USB Camera sensor to read/display the values according to the parking lot data/changes in status. Using the I2C interface we will design the final PCB through integrating each sensor into a single design for multiple sensor support. The IR Beam sensor will be extensively tested for the gate control mechanics and George will be accountable to ensure the hardware functions as designed.

Towards the final stages of the project, a re-design of the enclosure will be needed for a cleaner, sleeker and compact design. It will be important to house all three sensors which will be part of the final PCB design, and create a suitable and appropriate design to meet the requirements for each sensor. A collaborative effort will be made from all three members of the WatechPark developer team here, to redesign and refine the casing using the CorelDraw software. The prototype lab will be used for further inspection, and laser-cutting services during this phase of the project. Generally, this section helped explain the design process/experience with the hardware PCB and case. This included, providing an individual look at each sensor, and its printed circuit board design process and the work that was put into each hardware project initially. Finally, this section addresses the final PCB, casing design and integration that will be the focus for this semester with the criteria mentioned above.

### 3.1.3 Tools and Facilities

Throughout the duration of the project, for accommodating both the hardware and software side, it was necessary to work with a set of tools and environments to ensure successful completion, and the planned outcome for each hardware component. The tools used to complete the project ranged from the initial design, development, testing and presentation phase of the overall end product. To begin, in terms of the PCB (printed circuit board) process, was the requirement of using copper wire, and the pin headers that came with each sensor, wire strippers to cut off excess wire during soldering the PCB together. The copper wire would be used to solder the components on the printed circuit board together, using a soldering iron provided by the Prototype Lab in Humber College itself. The soldering iron was available from the parts crib and most of the soldering was done in the prototype lab itself. Wire cutters were used to cut off excess wires from soldering the vias, or cutting down on extra wires from the components used for the PCB.

The computer lab facility provides the most necessary components at your disposal. This includes, the PC system itself where most of the configuration was done for each sensor, a soldering iron including the solder itself. Major work was accomplished in the prototype lab, where the solder was received through the parts crib and was used in conjunction with the soldering iron available at the station. The following steps were taken to solder, firstly it is recommended to heat the soldering iron before applying the solder at 360 - 470 degrees Fahrenheit. The solder can then be applied to the tip of the soldering iron to produce the best results before starting the process. Other tools used in the production of the PCB (printed circuit board) included a helping hand. This tool allows you to reposition the PCB at different angles to solder from different perspectives, to avoid any overlapping connections or accidently joining a connection together. This tool is used to angle the printed circuit board and hold the board while you solder the components as necessary, without having the need to manually re-adjust the board at different times of soldering.

The facilities used in the project, included the prototype lab in Humber College. This facility is the main source of providing the services to etch the PCB board during its final stages of production, as well as provide the laser-cutting services for the final enclosure design. This facility was used to solder the components, sensors together on the PCB and test the final design. Through this environment, we were able to learn the process of soldering, and using the human resources in the facility we were able to make the best decisions, or re-think our strategy to solder the PCB. The facility provided us with more viable options, recommendations and the best way to overcome any issue we were experiencing. This included, designing the PCB board, using the CorelDraw software for housing the sensors in the final enclosure developed. Adjustments were made depending on the different scenarios, and based on the advice of Vlad and Kelly whom were present to help. Along with their support, we were able to come to re-assess, learn and fix our design mistakes to create a more polished end product for both cases. These services were provided by the prototype lab, for both etching and cutting services for the printed circuit board and the acrylic casing, with a laser-cutting machine being available to us without any cost. Overall, these tools and facilities supported our team and our project in achieving its goals and provided us with the outlet of both human and machine resources. This was accomplished through the help of others in the facilities, or through etching and cutting services provided to meet the project requirements in the end.

### 3.1.4 Shipping, Duty, Taxes

Budget for the project will be discussed in this section in terms for where we ordered our parts, how much it was shipping, taxes and totals will be shown. Most of the parts were ordered from last semester. George ordered all the parts he needed for the IR Break Beam sensor project, Vikas ordered his components necessary for the VCNL4010 Proximity/Light sensor and the final parts were the 2 Servo Motors and YoLuke HD Camera Sensor.

The components utilized for this project were ordered ahead of time to ensure work can be progressed or to test any faulty hardware piece for a quicker solution. Shipping time for most of the components was divided between 2-5 days, depending on the type of service chosen at the time of the transaction. The shipping options ranged from DHL Express shipping which transits worldwide and internationally. For example, for Vikas’s case while ordering the VCNL4010 Proximity sensor and its components at the time of the order, he chose he option of DHL Express shipping. This method of shipping provided, a much faster, reliable form of service with the components arriving roughly 2 days later from when the order was placed. In this case, the shipping cost was divided between two totals as shown in the screenshot below in the red outline. The shipping method and time required was lower than expected with other methods of shipping requiring 5 business days to ship the product from the source to the destination.

Duty was not placed, and avoided at all costs during this time to prevent any further costs from materials being shipped internationally or from other countries. For George’s case, most of the products shipped were either from Amazon, or through BuyAPi, with a single purchase for the IR Beam sensor being through Adafruit. The method of shipping was also different here, as with Amazon it provides the option of Amazon prime shipping. This depends upon the location the component is being shipped from of course, and based on that location, the time is calculated to ship the product out. Although, with Amazon Prime shipping most orders placed during a significant time of the day can be processed, and shipped within 2 days and delivered at a much quicker rate. Other hardware parts required a shipping time of 2-5 days with UPS or different forms of shipping requiring higher costs, if the item is needed within a set amount of time. For the most part, the components used in this project were shipped out within 2-5 days, and duty expenses were avoided to ensure an inexpensive project or to avoid any further delays.

Taxes were applied accordingly as well, with US (United States) tax rate being applied to the parts that were purchased through online shopping. Since these rates were in US dollars, we had to convert this value into CAD(Canadian) currency with the rate of 13% being applied instead as displayed below.

Below is the total project cost for the project, in terms for all the parts we all have currently in a BOM (Bill of Materials) format:

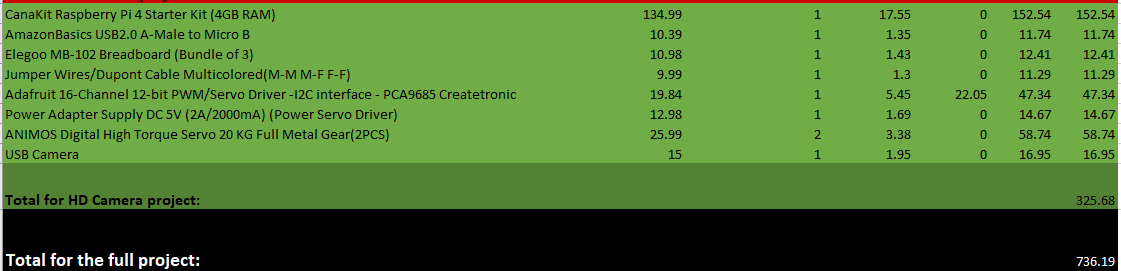


Figure 11: Final Project Budget

Most parts were ordered last semester and were used to complete their individual project. The total budget for the project is $736.19 CAD. The project total was calculated by adding up all the project budgets from last semester from each individual project. The projects involved are the IR Break Beam Sensor, VCNL4010 project and the HD Camera Project. They were all calculated from last semesters budget. There will be extra parts ordered for the final project and will be added to the total budget. These extra parts are not taken into consideration, the group is deciding on what extra parts we will use for this project. Such as, potentially ordering 3 more VCNL4010 Proximity sensor to accommodate our final parking lot prototype model with 4 parking spots. We plan on using all the parts ordered from the previous semester in the new semester. Some parts not considered in the budget are the parts being used from the parts kit such as some resistors and LEDs. The parts kit was purchased from the first semester and some of those parts are being used for the final project. This section helped provide a better insight into the shipping, duty and tax rates that went into ordering the components needed to complete the project and the financial state of the project before/after further development.

### 3.1.5 Time expenditure

Throughout the development of The WatechPark project, our overall time was split between hands on tasks, such as discussions, development, designing, building, debugging and testing, while the rest of our time was spent waiting for external tasks to be completed of a similar kind as parts delivery, laser cutting and PCB printing.

Please note that the time calculated in the following list, represents the total time spent by the three members of the project. For example, when discussing the time spent on designing the first PCB and the finalized PCB, in reality it reflects the collective time spent designing three PCBs for three different sensors and an additional three finalized PCBs for those sensors. This section of the report will touch on the total working time allotted to complete each requirement for the project, as well as address the amount of time needed to manufacturer the component from the total lead time. The following is the total breakdown of each section of both cases.  
Total of 105 hours spent as Working Time:

* 12 hours on project discussions and researching the required sensors and controllers
* 6 hours on finding the best vendor in terms or pricing and shipping time, and ordering the required parts, components and materials
* 6 hours on building and assembling breadboard prototypes
* 3 hours on voltage and safety testing for the breadboard connections before connecting our development platforms
* 6 hours on researching and installing the necessary third-party libraries for the sensors
* 6 hours on developing testing programs to be used on bread board circuits
* 6 hours on testing and debugging any anomalies in both the software code and the hardware connections
* 12 hours on designing and submitting both first and the finalized PCBs to be printed
* 6 hours on PCB assembly and soldering the needed components
* 3 hours circuit testing and safety verifications before connecting to the development platform
* 8 hours on designing, laser cutter submission and assembly of the PCB housing cases
* 15 hours on designing and developing the WatechPark android application
* 6 hours on Firebase initialization and filling up with data to be used with our android application
* 10 hours on testing and debugging our android application to ensure expected functionality with simulated sensor data stored in Firebase

Total of 22 days spent as Lead Time:

* 5 days on parts to be shipped and delivered
* 10 days on PCBs to be printed
* 7 days on housing case to be laser cut