# SeatDetect


By

Owen Brown

Yue Li

Huey Nguyen



Project Proposal for ECE 445, Senior Design, Spring 2021

TA: Haoqing Zhu

18 February 2021

Project #18

# Contents

# 1 Introduction

## 1.1 Objective

Before COVID-19, there was always a shortage of tables and seats at the Grainger Engineering Library, especially during weeknights. The cubicles were almost always full and if they were not, they were occupied by miscellaneous items. During those times, it would be extremely helpful to know which table in the library is open instead of walking in circles and waiting for a table to open up. Also, when there are absolutely no seats available, you just wasted your time coming to Grainger just to realize there is no room.

To solve this problem, we will design and implement a motion-sensing device for each table. The occupancy status should be transmitted through Wi-Fi to the end-user; the status should be updated in a timely manner after each change. The user should be able to see the change and walk to the desired location and occupy the free seat. This will not only solve the problem of not being able to find a seat, but also additional problems such as knowing when the library is completely full, and this makes it easier for the employees to enforce policies where people lose their spot if they are gone for too long.

## 1.2 Background

While there are no known existing solutions to this specific problem, solutions to similar problems are already present today. One of these similar problems is finding a parking spot in a parking garage. This problem is solved using infrared technology [2], AI [3], and using various other sensor types.

From here, is easy to draw the parallel between finding a parking spot and finding a seat in the library. Since these technologies are utilized in real-world applications, it proves that this problem can be solved in an affordable way, especially since the technology needed to detect a person in a booth is arguably cheaper than detecting a car in a parking spot.

## 1.3 Physical Design

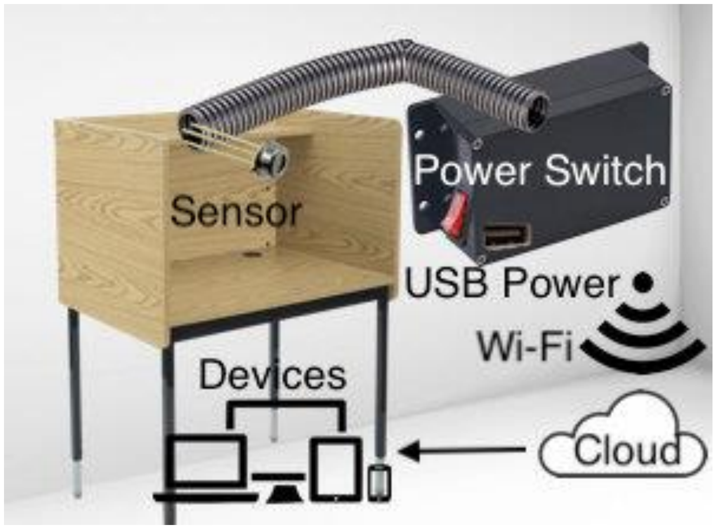


Figure 1. Physical Design Diagram

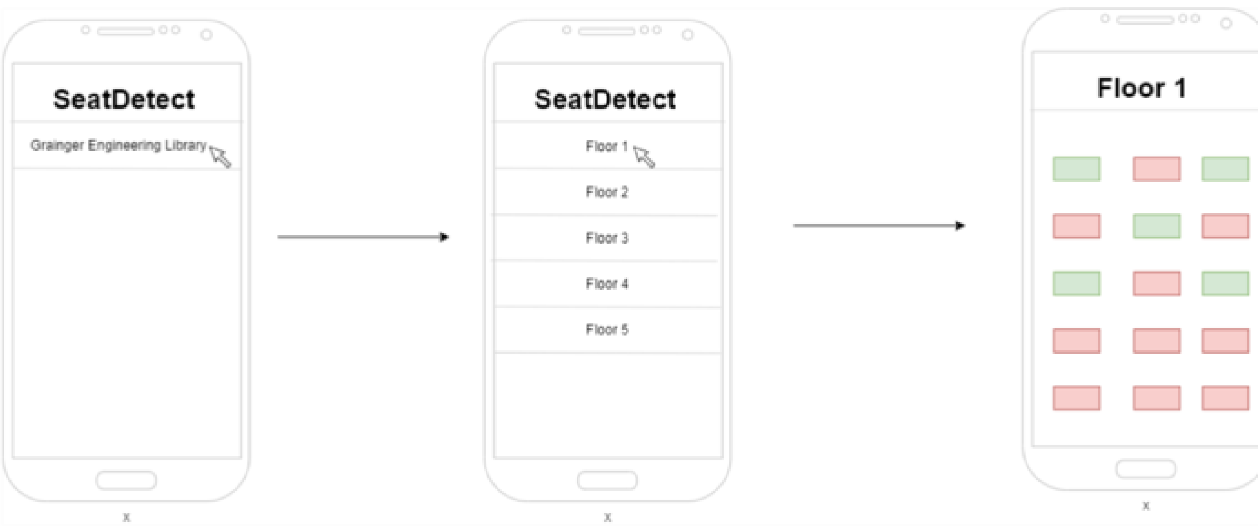


Figure 2. User Interface

## 1.4 High-level Requirements List

- Accuracy of occupancy status over 95% on repeated tests of the same booth. This will be measured by running multiple occupancy sensing tests on the same sensor unit and counting how many successful readings there are out of total attempts.
- Occupancy status changes from unavailable to available 15 mins after the seat is no longer occupied. Status changes from available to unavailable immediately after the seat becomes occupied.
- Occupancy status updates on the mobile/web app within 30 seconds of a status change.

# 2 Design

## 2.1 Block Diagram

SeatDetect consists of four main modules: a power supply, a software module, a control unit, and a sensing module. The power system ensures that the system can be powered continuously all day and night with the proper 3.3V. The control unit contains a microcontroller with integrated Wi-Fi, which will handle the data received from the sensing module. Lastly, the software module displays the status data to the end user, the students.

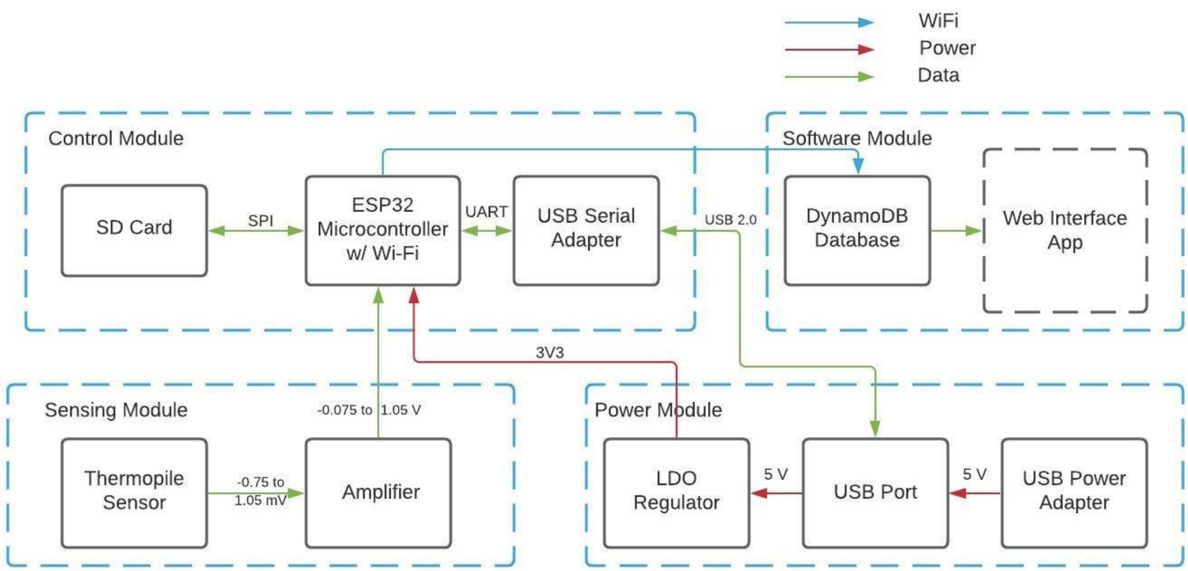


Figure 3. Block Diagram

## 2.2 Functional Overview

### 2.2.1 Control Module

The control module interfaces with every single module. It is the module that needs power. It also relays preprocessed data from the sensing module to the software module and includes dynamic memory storage in the form of an SD card.

#### 2.2.1.1 ESP32 Microcontroller w/ Wi-Fi

The ESP32 microcontroller receives the sensor signals, processes the data into whether the seat is occupied or not, and sends the data to the database over Wi-Fi.

#### 2.2.1.2 USB Serial Adapter

The USB serial adapter is necessary to enable the programming of the ESP32. It takes in the upload of code from a computer by converting the USB data to serial data that the ESP32 can then interpret.

*Requirement:* The USB Serial Adapter must use UART communication to communicate with the ESP32.

#### 2.2.1.3 SD Card

The microSD card is used to store data from the thermopiles and is connected to the microcontroller through SPI.

### 2.2.2 Sensing Module

The sensing module senses if the user is in the booth using a thermopile sensor and some residual circuitry. It relays passive signals from the thermopile to the control module where it is then processed to sensible and understandable data.

#### 2.2.2.1 Thermopile Sensor

ZTP-135SR is a thermopile sensor that detects human presence by thermal energy correspondence.

*Requirement:* The thermopile sensor must output voltages from a range of -0.75mV to 10.5mV corresponding to its thermal energy ranging from temperatures of 20°C to 80°C.

#### 2.2.2.2 Amplifier

The amplifier needs to amplify the signal from the sensor as the signal voltages are much too small to be read by the ESP32.

*Requirement:* The amplifier must amplify the voltage from the sensor by 100x (gain of 100).

### 2.2.3 Power Module

The power module is a wired USB power supply for increased scalability and consistency, which is required to keep the communication network up continually. In a network with multiple sensing modules, this power module will be used to power the whole system. Because the addition of sensing modules does not add much additional power requirements from the control module, the addition of sensing modules will not require significantly more power from the power module. In the scenario where enough sensing modules are adding to warrant additional control modules due to the I/O limitations of the ESP32, there would be also be a need for more power modules, and therefore the whole system would be duplicated except the software module. The following requirement does not cover the ladder case.

*Requirement:* The power module must be able to supply at least 250mA and power the control module with a continuous 3.3V ± 0.1V.

#### 2.2.3.1 USB Power Adapter

The power adapter converts the grid electricity into DC power that is used for the whole system.

*Requirement:* The USB power adapter must convert 120V AC from the wall outlet to 5V DC.

#### 2.2.3.2 USB Port

The mini–B female USB port receives power from the power adapter and also allows for programming the ESP32 from a computer (at separate times). The port connects to the voltage regulator for power supply and a serial adapter for communication to the ESP32. When a computer is hooked up to the port, the power adapter will not be, and vice versa.

#### 2.2.3.3 LDO Regulator

The LDO regulator steps down the voltage from the power adapter for the ESP32.

*Requirement:* The LDO regulator must step down 5V DC to 3.3V ± 0.1V DC.

### 2.2.4 Software Module

The software module consists of a web-based application that allows users to check availability and location of seats as well as a database that stores the information pertaining to each seat.

*Requirement:* The software module must reflect the status changes detected by the control module and display such a change to the end user.

#### 2.2.4.1 DynamoDB Database

Stores the data and the ability to view the status of a seat (occupied by a person, personal items, inactivity, etc.).

The data transmitted by the sensors must be persisted into the DynamoDB database. This database should have the Seat ID as a primary key and it will store the perspective occupancy status and the time which it was last updated. As a table it will look like **Error! Reference source not found.**.

| **Seat #(Integer)** | Status(Boolean) | lastUpdatedTime(DateTime) |
|---|---|---|
| 1 | FALSE | 2021-02-13T17:09:42.411 |
| 2 | TRUE | 2021-02-13T17:10:42.411 |

**Figure 4. Database Table**

*Requirement 1:* The DynamoDB database must have read and write capabilities.

*Requirement 2:* The DynamoDB database must have SeatID as the primary key.

#### 2.2.4.2 Web Interface App

The Web Interface App allows the user to view the map of Grainger Library (for now) and its corresponding tables on each floor.

*Requirement 1:* The web interface app must be able to establish a connection with the DynamoDB database.

*Requirement 2:* The web interface app must be able to display the change in status of a specific SeatID by the DynamoDB database.

## 2.3 Risk Analysis

An inherent risk of this design is that this project will have to depend on the functionality of the school Wi-Fi. If the school Wi-Fi at the Grainger Engineering Library malfunctions, a timely update will not be possible and as it would result in the failure to detect the occupancy status.

Another difficulty of this project deals with the accuracy of the thermopile sensor. In order to achieve a 95% accuracy of detection, we need to make sure that the thermophile sensor is as accurate as possible. We need to find out exactly how accurate it is and if the sensor can indeed differentiate between body heat and room temperature in the case when room temperature is high. If the sensor is unable to do that, we need to find an additional sensor to pair with the thermopile sensor to ensure the accuracy.

Another risk that is specific to this semester is in an event which the course goes all online. Our contingency plan includes that we will place more emphasis on developing the software as well as making sure that the interface is as user-friendly as possible. Moreover, there is more analysis we could do regarding the data that we collect such as what day of the week or time Grainger Library tends to be the busiest, which floor has the most congregation of students, etc.

# 3 Ethics and Safety

Ethics and Safety will be imperative to successfully carry out our project. During this difficult time of COVID-19, it is especially important that we follow closely IEEE Code of Ethics #1 [1], that we do not put other people's health in harm's way while conducting this project. That means to closely follow the CDC guidelines as well as to build and test our project with as little face to face interactions as possible. When going to the lab is necessary, precautions such as wearing gloves, using hand sanitizer regularly needs to be taken extremely seriously.

In addition to paying attention to safety, it is also important that we follow the guidelines of IEEE Code of Ethics #5, which suggests that we need "to seek, accept and offer honest criticism of technical work, to acknowledge and correct errors, to be honest and realistic" [1]. We as a team need to take advantage and make the most out of the weekly TA meetings and be proactive when possible. Furthermore, we plan on following this guideline and conducting surveys of our prototype once it starts working to receive feedback to further improve our product. Otherwise, there are very few ethical considerations specific to our project.

# 4 References


[1] ieee.org, "IEEE Code of Ethics", 2016. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 08-Feb-2021].

[2] D. Roos, “How Parking Garages Track Open Spaces, and Why They Often Get It Wrong,” *HowStuffWorks*. [Online]. Available: https://electronics.howstuffworks.com/everyday-tech/how-parking-garages-track-open-spaces-why-they-often-get-it-wrong.htm

[3] *ParkingDetection*. [Online]. Available: https://www.parkingdetection.com/