# Sensory Awareness Device for Bars and Restaurants

Megan Heinhold, Evan Lindquist, Carl Wolff

meganjh3@illinois.edu, evanl3@illinois.edu, cwolff2@illinois.edu

Design Document for ECE 445, Senior Design, Spring 2022
TA: Amr Ghoname

20 February, 2022
Project Number 17

**1. Introduction** **2**
1.1 Problem 2
1.2 Solution 3
1.3 Visual Aid 3
1.4 High-Level Requirements List 5
**2. Design** **6**
2.1 Block Diagram 6
2.2 Requirements & Verification Tables 6
2.2.1 Power Supply 6
2.2.2 Control Unit 7
2.2.3 WiFi Module 8
2.2.4 Sensor Block 9
2.3 Circuit Schematic 10
2.4 Tolerance Analysis 14
3. Cost and Schedule 18
3.1 Cost Analysis 18
3.2 Schedule 20
**4. Ethics and Safety** **21**
4.1 Development 21
4.2 Misuse 22
4.3 General Safety Concerns 22
**References** **24**

# 1. Introduction

## 1.1 Problem

Many people suffer from conditions such as ADD, epilepsy, and sensory processing disorder that affect their ability to operate and be comfortable in certain environments[1],[2]. Those affected by these conditions need awareness of certain triggers such as loud noises, flashing lights, or crowded areas before they enter an establishment. Beyond common sense predictions, there is not a reliable way to assess how many and to what degree these triggers will be present at a bar or restaurant.


*Figure 1. Infographic showing common noises and their relative DBs.*

Along with the previously mentioned issues, many people want to gauge the activity of a public place before making the trek out to said establishment, whether it is for a lively night out, or a quiet sitdown get-together.

## 1.2 Solution

In this project, we propose the use of a static measurement device to measure various factors in a social area in order to provide a live report of the activities going on within. Data is taken from the scene, interpreted, and then uploaded to a website so that users can view it online. This would allow people to assess the various noise, light, and capacity levels before making a commitment to going out, to comfortably satisfy their individual needs.

## 1.3 Visual Aid

This visual aid (Figure 2) shows the flow of information from the initial measurements to interpreted data reaching a given user.

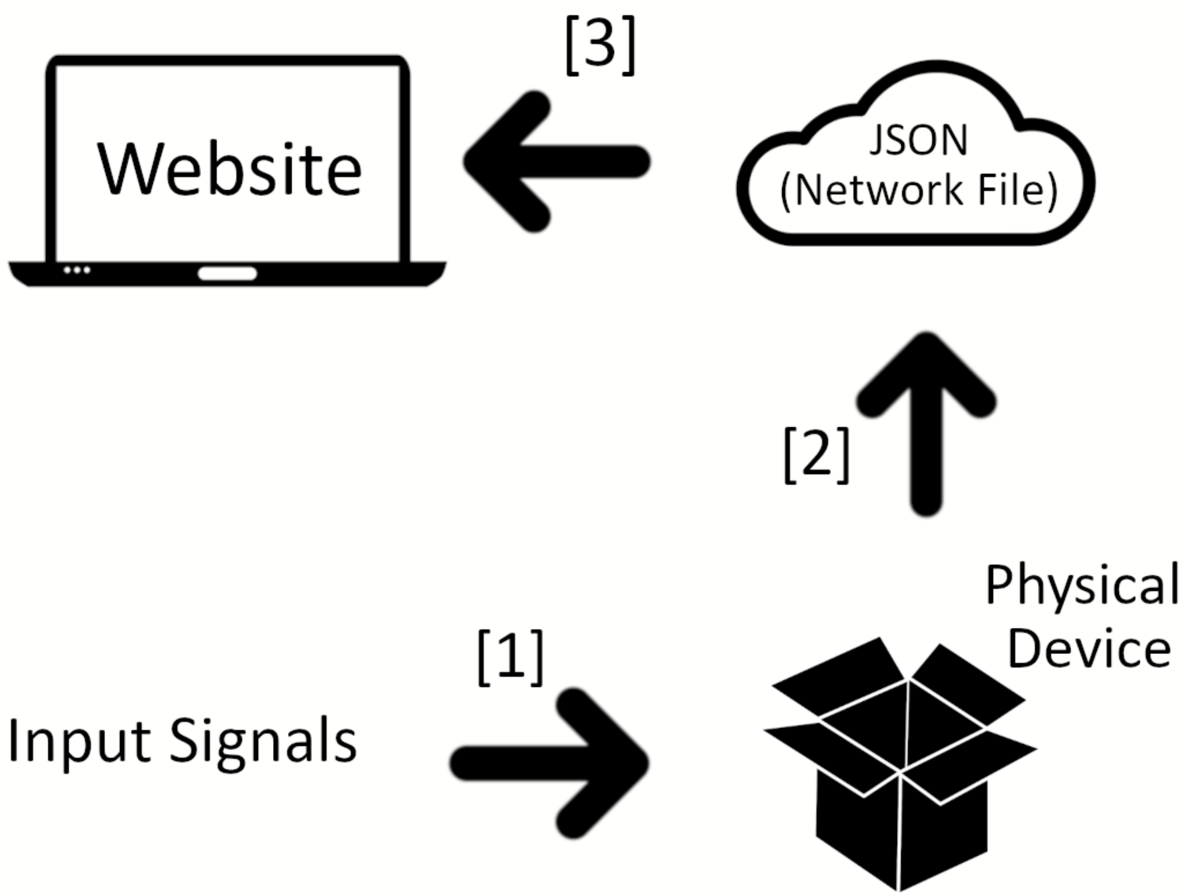


*Figure 2. Overview of data flow within the system.*

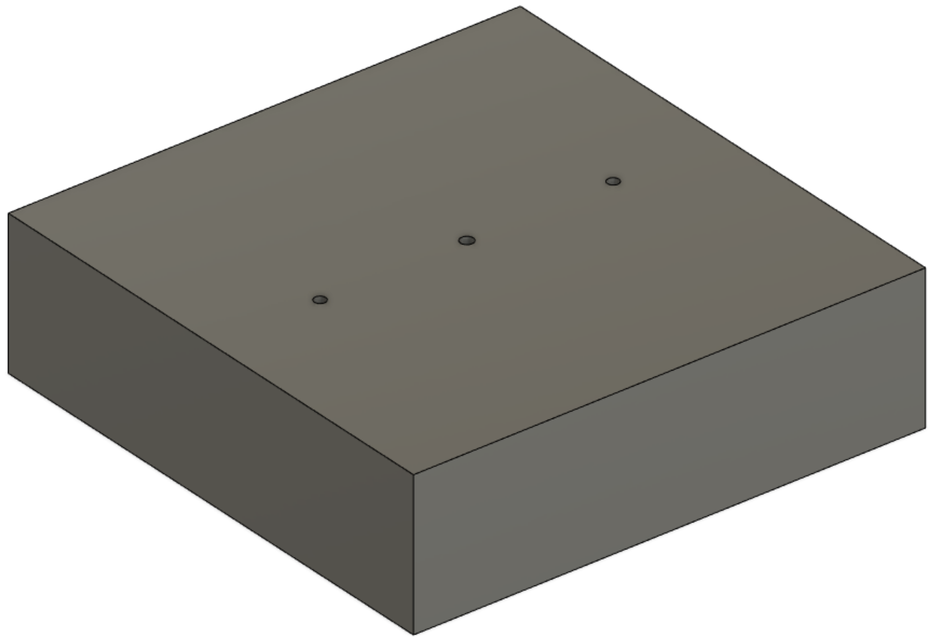

*Figure 3. Enclosure visualization, with holes for sensors.*

## 1.4 High-Level Requirements List

1. The device must be able to be plugged into a standard 110 V wall outlet and fit within a 10 inch by 10 inch by 2 inch footprint. It should not require maintenance more than once a month.
2. The device must accurately characterize both ambient levels and safety alerts. Temperature should be recorded to within +/- 1 degree Fahrenheit. Safety alerts should be identified for sounds above 110 dB and flashing light frequencies between 10-12 Hz.
3. The transmission of data to the user must be “real-time”, so collection, transmission, and interpretation of data cannot take more than 10 minutes. Safety alerts regarding flashing lights and dangerous sound levels should be communicated to the web app in no more than 2 minutes.

# 2. Design

## 2.1 Block Diagram

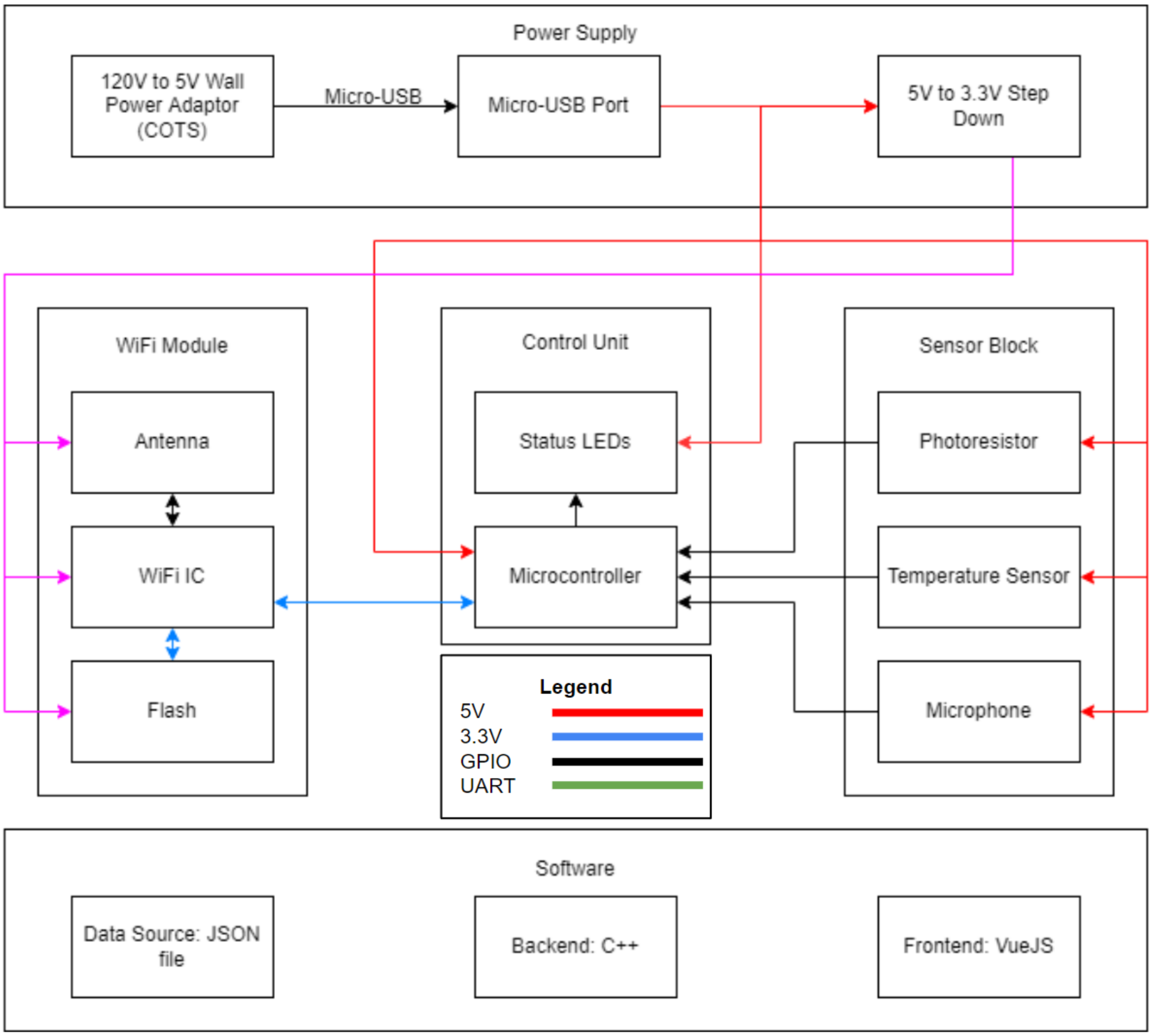


*Figure 4. Block diagram of connections within the system.*

## 2.2 Requirements & Verification Tables

### 2.2.1 Power Supply

The requirements for the power supply are listed below in Table 1.

*Table 1: Power Supply Requirements & Verification*

| Requirements | Verification |
|---|---|
| 1. The power supply provides 5 V +/- 0.5% from a wall power adaptor | 1A. Measure the output voltage from the wall power adaptor using an oscilloscope, ensuring that the output voltage stays within 0.5% of 5 V. |
| 2. The power supply provides a 3.3 V +/- 0.5% from a low dropout regulator driven by the 5V mentioned above | 2A. Measure the output voltage from the regulator using an oscilloscope, ensuring that the output voltage stays within 0.5% of 3.3 V. |
| 3. The power supply is able to operate within 0-1 A from the 5 V source and able to operate within 0-0.1 A from the 3.3 V source | 3A. For the 5 V source, connect the 5 V line to multiple resistors together such that the total resistance of the network is 5 Ω +/- 0.5% but no resistors are dissipating more power than they can handle. Measure the voltage network across the resistive network using an oscilloscope to confirm it is 5 V+/- 0.5%.<br>3B. For the 3.3 V source, connect the 3.3 V line to multiple resistors together such that the total resistance of the network is 3 Ω +/- 0.5% but no resistors are dissipating more power than they can handle. Measure the voltage network across the resistive network using an oscilloscope to confirm it is 3.3 V +/- 0.5%. |

### 2.2.2 Control Unit

The requirements for the microcontroller and network LED are listed below in Table 2.

*Table 2: Control Unit Requirements & Verification*

| Requirements | Verification |
|---|---|
| 1. There must be a total of six GPIO pins that can appropriately handle signals between 0 V and 5 V +/- 0.5% while sinking/sourcing at least 20 mA +/- 0.5% of current per pin. | 1A. Connect an I/O pin of the MCU to a series combination of an LED with a forward voltage of 3.2 V +/- 0.1 V and a resistor of 90 Ω +/- 0.5%.<br>1B. Using an ISP programmer, program the board with a test program that blinks the LED on and off. |
| 2. The RX pin must be able to interpret 3.3 V +/- 0.5% signals (from WiFi chip) as logical HIGHs. | 2A. Use the same LED setup as 1A while also connecting the WiFi chip to the MCU.<br>2B. Using an ISP programmer, program the board with a test program that pings the WiFi chip and illuminates an LED upon a successful response. |
| 3. There must be an A/D converter with at least ten bits of resolution and three available channels which can be read sequentially. | 3A. Use the same LED setup as 1A while also connecting the TMP36 to a different I/O pin while the sensor is in a temperature-controlled environment.<br>3B. Using an ISP programmer, program the board with a program that reads the temperature and illuminates the LED if the temperature is different from the initial reading.<br>3C. After the initial reading is complete, adjust the temperature of the environment by a small amount (~1°F) to ensure that the LED illuminates. |

### 2.2.3 WiFi Module

The requirements for the WiFi module are listed below in Table 3.

*Table 3: WiFi Module Requirements & Verification*

| Requirements | Verification |
| --- | --- |
| 1. The chip must be able to take commands via UART from an external microcontroller. | 1A. Connect the chip to an Arduino (or similar microcontroller) using a breadboard.<br>1B. Upload a test program to the Arduino that attempts to ping the WiFi chip and indicates using the serial monitor if a response was received. |
| 2. The RX pin must be able to handle a 5 V +/- 0.5% signal (from the microcontroller) without breaking the chip. | 2A. Connect the RX pin to the microcontroller, using the same program as in Table 2 1A. Verify that the LED is able to illuminate, indicating a successful ping. |
| 3. The chip must be able to connect to a network and send 4 kB (max size) of data supplied by the microcontroller once every ten minutes +/- thirty seconds. | 3A. After completing 2A, use the same hardware setup but change the software on the MCU using an ISP programmer to have the WiFi chip connect to a network and upload a dummy data packet to a JSON file once every ten minutes.<br>3B. Verify on a separate computer that the JSON file is modified once every ten minutes. |

### 2.2.4 Sensor Block

The requirements for the power supply are listed below in Table 4.

*Table 4: Sensor Block Requirements & Verification*

| Requirements | Verification |
|---|---|
| 1. The output of the photoresistor must be within 0 V and 5 V +/- 0.5% with a current of no more than 20 mA +/- 0.5% in various light levels. | 1A. On a breadboard, connect the photoresistor and a 1 M Ω +/- 0.5% resistor in series. Connect one lead of the photoresistor to 5 V using an Arduino (or similar microcontroller) and ground the circuit.<br>1B. Connect the second lead of the photoresistor to an analog I/O pin on the Arduino.<br>1C. Run a test program on the Arduino that records the voltage level of this pin.<br>1D. Use an oscilloscope to measure the current running through the resistors.<br>1E. Verify that the voltage stays between 0 - 5 V +/- 0.5% for different sound levels. |
| 2. The microphone output must be within 0 V and 5 V +/- 0.5% with a current of no more than 20 mA +/- 0.5% depending on sound levels. | 2A. Power and ground the microphone chip using an Arduino (or similar microcontroller) 5 V output. Connect the analog output of the microphone to an analog I/O pin on the Arduino.<br>2B. Run a test program on the Arduino that records the voltage level of this pin.<br>2C. Use an oscilloscope to measure the current running through the microphone.<br>2D. Verify that the voltage stays between 0 - 5 V +/- 0.5% for different sound levels. |
| 3. The temperature sensor output must be within 0 V and 5 V +/- 0.5% with a current of no more than 20 mA +/- 0.5% depending on temperature. | 3A. Using the same setup as in 2A - 2C, connect the output of the temperature sensor to the Arduino and run the same program.<br>3B. Verify that the voltage stays between 0 - 5 V +/- 0.5% and current less than 20 mA +/- 0.5% for different temperature levels. |

### 2.2.5 Software

The requirements for the software are listed below in Table 5.

*Table 5. Web Application Requirements & Verification*

| Requirements | Verification |
|---|---|
| 1. The C++ program must take raw input data from 3 sensors over the span of 10 minutes and output averaged levels in terms of temperature, light, and sound. | 1A. Using an ISP programmer, program the microcontroller with the C++ program.<br>1B. Connect the outputs of the 3 sensors to the input pins of the microcontroller.<br>1C. After 10 minutes, verify that the output values are reasonable for the environment in terms of temperature, light, and sound. |
| 2. Data from the microcontroller must be able to be transmitted via the WiFi module to the JSON file. | 2A. Write a test program that can update a single value in a JSON file.<br>2B. Using an ISP programmer, program the microcontroller with the test program.<br>2C. Run the program, and verify that the JSON file is updated. |
| 3. The web application must display updated data within 1 minute +/- 30 seconds. | 3A. Edit the JSON file from which the web application reads its data.<br>3B. Refresh the web application. Continue to refresh the page every 2 seconds. Verify that the updated changes appear within 1 minute +/- 30 seconds. |

## 2.3 Circuit Schematic

The schematic of the circuit's power supply can be seen in Figure 5. It contains a micro USB-B port that will introduce a 5 V power supply to the system. That port is connected to a low dropout voltage regulator that steps the voltage down from 5 V to 3.3 V; there are two decoupling capacitors connected to the regulator to reduce any noise from the power supply or rest of the circuit. There is also a status LED which is illuminated when the power supply is connected and is off otherwise.

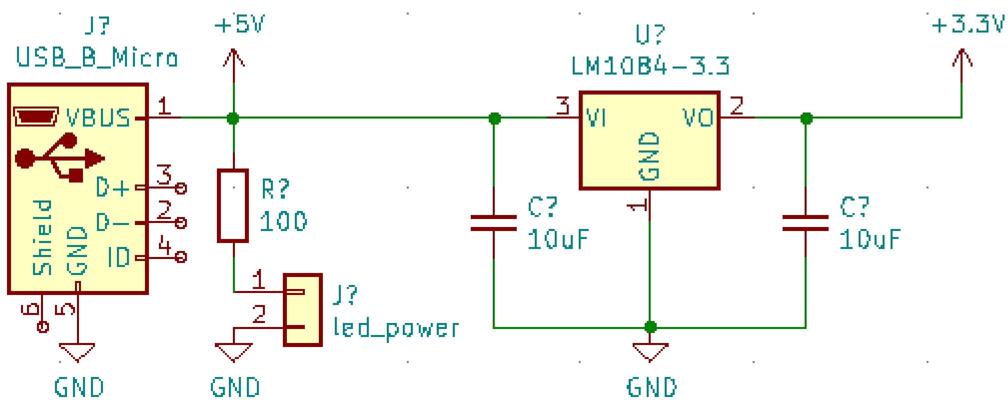


*Figure 5. Circuit Power Supply Schematic.*

The schematic of the circuit's microcontroller can be seen in Figure 6. It is programmed using an external programmer over SPI. It communicates with the WiFi module using UART. All of the sensors are connected to the microcontroller and will make use of its built-in analog to digital converter.

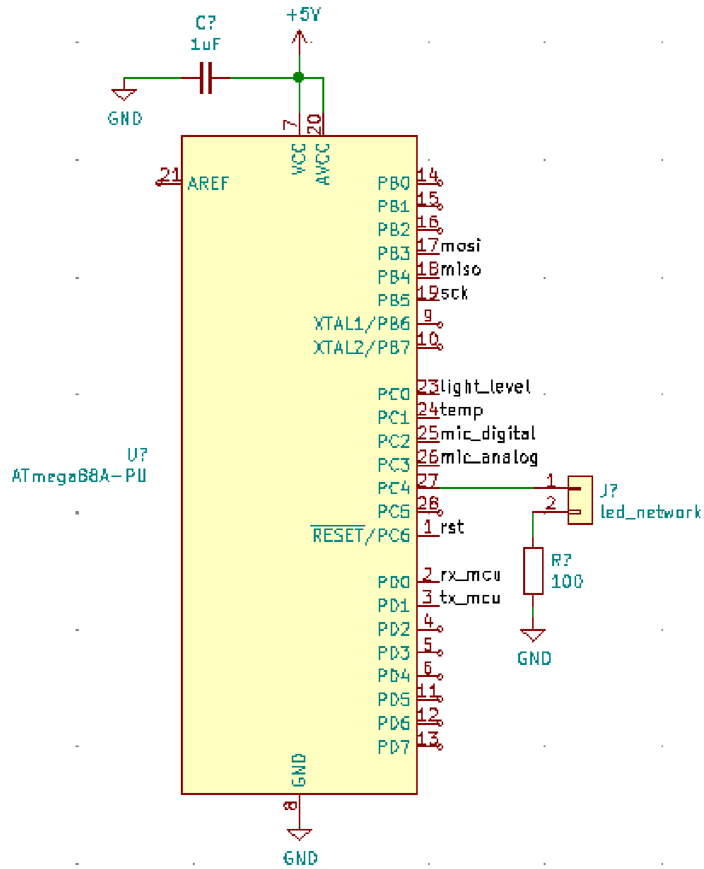


*Figure 6. Microcontroller Schematic.*

The schematic of the circuit's sensor block can be seen in Figure 7. The photoresistor is connected as part of a resistor divider and the light level is read from the middle of that divider. The temperature sensor and microphone are connected directly to the microcontroller.

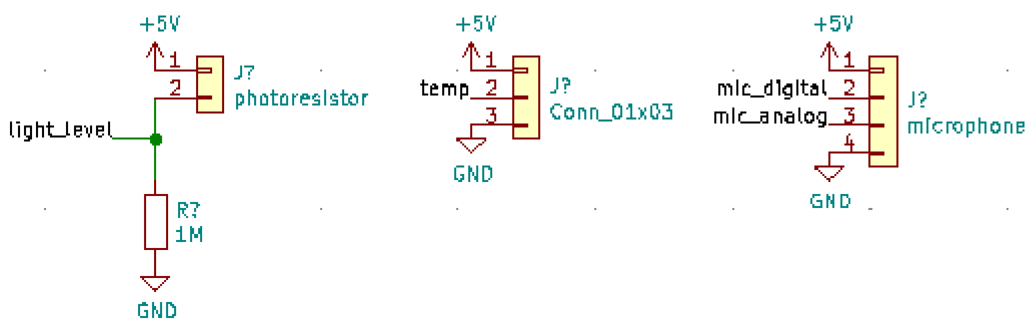


*Figure 7. Sensor Block Schematic.*

The schematic of the circuit's WiFi can be seen in Figure 8. The chip is connected to the microcontroller via UART. The pins are connected such that the chip is being driven by UART commands as opposed to having a program loaded onto it. We will use an integrated WiFi chip that has a WiFi IC, antenna, and flash all on board. Using an integrated chip is preferable to three independent components because it is cheaper, easier to use, and takes up less space on the PCB.

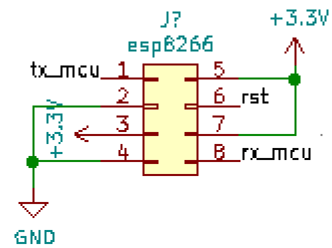


*Figure 8. WiFi Module Schematic.*

The schematic of the circuit's programming header can be seen in Figure 9. This is used to program the microcontroller via an external programmer.

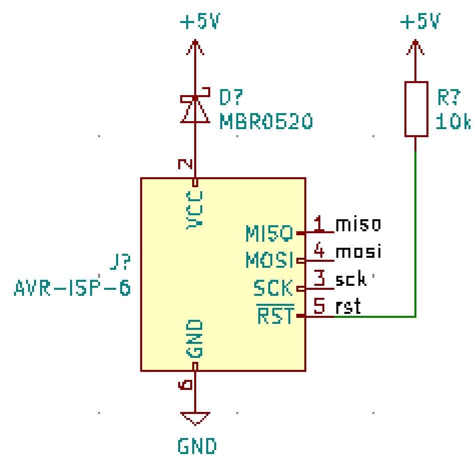


*Figure 9. Programming Header.*

## 2.4 Tolerance Analysis

Photoresistors come in many different varieties that each have slightly different response times, resistances, etc. A direct comparison of these attributes and important characteristics is necessary to make an optimal photoresistor choice for the device. Table 6 showcases this comparison using data from the photoresistor datasheet [3]. Note that resistor R2 is sized such that the output voltage in a well lit room is approximately 5 V while the output voltage in a dark room is 0 V.

*Table 6. Table of relevant photoresistor data.*

| Part Number | lit R1 (kΩ) | dark R1 (kΩ) | R2 (kΩ) | lit V_o (V) | dark V_o (V) | lit Power (mW) | dark Power (mW) |
|---|---|---|---|---|---|---|---|
| 5506 | 4 | 150 | 50 | 4.630 | 1.250 | 0.463 | 0.125 |
| 5516 | 7.5 | 200 | 50 | 4.348 | 1.000 | 0.435 | 0.100 |
| 5526 | 14 | 1000 | 100 | 4.386 | 0.455 | 0.219 | 0.023 |
| 5528 | 15 | 1000 | 100 | 4.348 | 0.455 | 0.217 | 0.023 |
| 5537 | 37 | 2000 | 470 | 4.635 | 0.951 | 0.049 | 0.010 |
| 5539 | 60 | 5000 | 470 | 4.434 | 0.430 | 0.047 | 0.005 |
| 5549 | 90 | 10000 | 1000 | 4.587 | 0.455 | 0.023 | 0.002 |

One characteristic of importance is the response time (Figure 10). Since the photoresistor needs to be able to detect flashing lights, its response time needs to be less than the half period of flashing lights because the circuit needs to enter steady state during light flashes and the dark moments between them. Commercially-available strobe lights have a max frequency of 12 Hz (or 12 flashes per second) which means that each flash of light lasts for just over 40 milliseconds. Therefore, it is essential that our photoresistors have a maximum response time of 40 milliseconds, although ideally it should have a response time even lower than that. This criteria narrows down the options to part numbers 5526 through 5549.

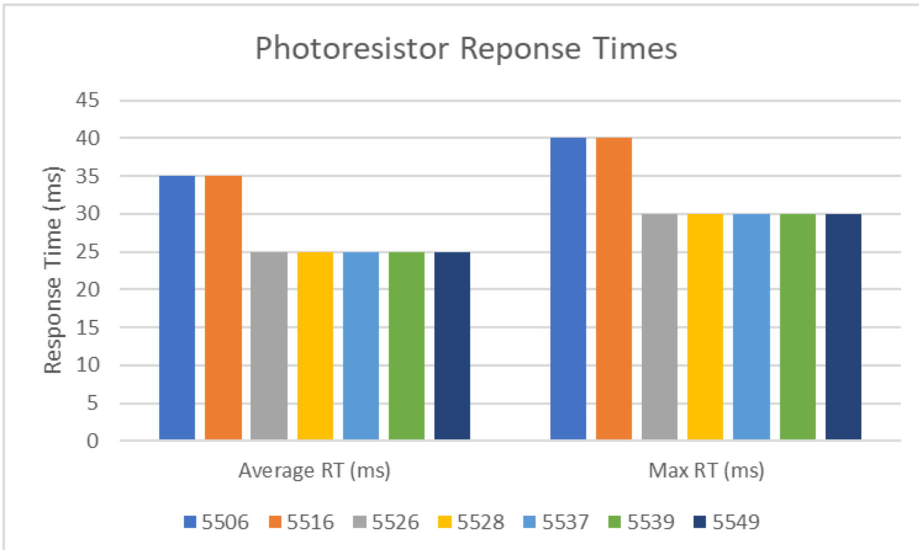


*Figure 10. Photoresistor Response Time*

Another important criteria for the photoresistor is the overall power consumption of the module (Figure 11). The smaller resistance values allow more current to run through the photoresistor which ultimately dissipates much more power as heat. Because the device will be running indefinitely (no planned off-periods), minimizing power consumption is critical to avoid large negative impacts on the electrical bills for the establishments using it. The 5549 photoresistor has the lowest power consumption with an effective response time which makes it the best choice for our device.

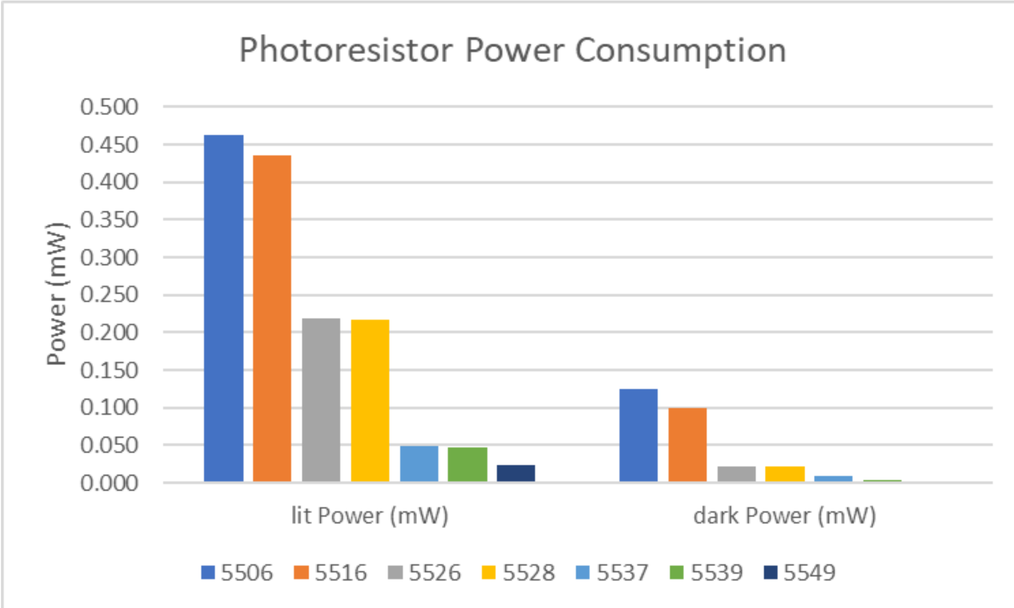


*Figure 11. Chart of Photoresistor Power Consumption*

Now that the 5549 photoresistor has been chosen, we will include a brief analysis of the 5549 photoresistor in action within the circuit. The photoresistor will be connected in series to a standard resistor; these two resistors will drop a total of 5 V. The datasheet for the photoresistor provides a few values of note which are shown in Table 7.

*Table 7. Selected Photoresistor Datasheet Values.*

| Part Number | Minimum Light (10 lux) Resistance (kΩ) | Maximum Light (10 lux) Resistance (kΩ) | Minimum Dark Resistance (kΩ) |
|---|---|---|---|
| 5549 | 45 | 140 | 10000 |

The most important metric to verify is that the current never exceeds the limit of the microcontroller’s I/O pins. This is fulfilled because even at the minimum resistance with no other resistors (i.e. the whole 5V drop occurs on the photoresistor), the current would still be less than 1mA which is below our microcontroller’s GPIO current limits.

Another important metric to verify is that the difference in output voltage between light and dark is quite large. This is important because we want to maximize the precision of the reading of the light level which is limited by the analog to digital converter. If the difference in the output voltages was small, most of the ADC's range would be completely wasted and the precision of the reading would be quite small. Table 8 shows the percentage of the ADC that is utilized. Because we will quantify the reading from our photoresistor into one of several buckets (such as "bright", "well lit", "soft lighting", "mood lighting", "dark", etc.), the precision does not need to be incredibly high, so the minimum utilization (78.63%) is more than adequate for our device.

*Table 8. Photoresistor ADC Utilization.*

| Part Number | lit R1 (kΩ) | dark R1 (kΩ) | lit V_o (V) | dark V_o (V) | ADC levels used | % of ADC levels used |
|---|---|---|---|---|---|---|
| 5549 (min) | 45 | 10000 | 4.785 | 0.455 | 887 | 86.60% |
| 5549 (avg) | 90 | 10000 | 4.587 | 0.455 | 846 | 82.65% |
| 5549 (max) | 140 | 10000 | 4.386 | 0.455 | 805 | 78.63% |

# 3. Cost and Schedule

## 3.1 Cost Analysis

Our overall cost is broken down into 2 main parts: worker labor, and parts cost. Worker labor can be approximated at around 40/hr to approximate a realistic starting salary for a recently graduated ECE student with a bachelor's degree. We have also approximated this project to take around 4 to 10 hours of our time each week, leaving us with a total worker's compensation within a range of 400 $ to 1000 $. This range is quite large, owing to the uncertainty in the amount of work required to design and build our final product.

Our parts list is listed on our team's repository on Github, and will be laid out in Table 9.

*Table 9. Cost Analysis for circuitry parts.*

| Part ID / Function / Manufacturer | Part Pricing ($) (All quantities are 1 unless otherwise stated) |
|---|---|
| LM1084-3.3 / LDO / TI | 0.90 |
| ATMEGA328p / Microcontroller / Microchip Technology | 2.47 |
| LEDs / Indicator Lights / Generic | ~ |
| 5549s / Photoresistor / XINGYHENG | 11.99 for a box of photoresistors |
| TMP36s / Temperature Sensor / KOOKYE | 11.69 |
| KY-038 / Microphone / DEVMO | 13.99 |
| ESP8226 / WiFi Module / Esspressif Systems | 12.99 |
| Micro-USB Port / Initial Setup / Generic | ~ |
| Total | 54.03 |

This puts our total cost at approximately $ 754.03.

## 3.2 Schedule

*Table 10. Proposed Schedule of work for each group member.*

| Week | Megan Heinhold | Evan Lindquist | Carl Wolff |
|---|---|---|---|
| Feb. 21 | Design Doc<br>Soldering Assignment | Design Doc<br>Soldering Assignment<br>Order parts | Design Doc<br>Soldering Assignment |
| Feb. 28 | Write script to read data from JSON file to web app | Order remaining parts<br>Locate connectors<br>Schematic Review | Unit test ICs |
| Mar. 7 | Teamwork Eval<br>Flash ATmega<br>Confirm resistor divider values | Teamwork Eval<br>Routing and Labeling<br>PCBway order | Teamwork Eval<br>Test power supply<br>Characterize Sensors |
| Mar.14 | Spring Break | Spring Break | Spring Break |
| Mar. 21 | Solder PCB<br>Ensure WiFi module can connect to internet through Arduino | Code MCU<br>PCBway order | Review first PCB<br>Code MCU (data from sensors) |
| Mar. 28 | Progress Report<br>Tweak data representation on web app<br>Implement logins(opt.) | Progress Report<br>Ensure WiFi module can connect to networks | Progress Report |
| Apr. 4 | Solder Second PCB | Determine testing | Review Second PCB |
| Apr. 11 | Full device testing for room conditions | Full device testing for room conditions | Full device testing for room conditions |
| Apr. 18 | Mock Demo | Mock Demo | Mock Demo |
| Apr. 25 | Demo<br>Mock Presentation | Demo<br>Mock Presentation | Demo<br>Mock Presentation |
| May 2 | Presentation<br>Final Paper | Presentation<br>Final Paper | Presentation<br>Final Paper |

# 4. Ethics and Safety

## 4.1 Development

Regarding the development of our project, we plan to adhere to and respect both IEEE and ACM Codes of Ethics. Since the underlying purpose of our project seeks to improve accessibility of social gatherings in restaurants and bars, we will also ensure that the development of our project respects all persons, and does not discriminate against anyone especially against those with disabilities that our project may be useful for as outlined in [4, Principle 1.4] and [5, Sec. II]. Use of our device by establishments will be completely optional, and furthermore continued use of our device and corresponding app will be optional as well. At any time, users of our device and app can discontinue use, especially if undue harm to their customers or business occurs from the use of our project [4, Principle 1.2]. To respect privacy and honor confidentiality, our project will not collect any data that can be traced back to individuals. Our sensors will not transmit audio recordings, only signals corresponding to audio levels, and our web app will not collect private information  [4, Principle 1.6]. As a part of the course for which this project will be carried out, we will seek out and listen to review from peers and advisors regarding technical aspects, social context, and any ethical concerns of our project [5, Sec. I.5], [3, Principle 2.4]. To keep ourselves and others safe, we will follow the safety guidelines of the laboratory space we work in. The individual components of our project should pose no serious safety concerns because we intend to use them all within the manufacturer’s guidelines.

## 4.2 Misuse

Improper use of our product may result in violation of both IEEE and ACM Codes of Ethics. Because our device must be installed near a power outlet which may require mounting to a wall or ceiling or placement in an open area, improper placement of the device can result in physical injury as well as general malfunction of the device. In addition, since there is a software component of our project that involves both individuals and establishments interacting with the system, there is a possible opportunity for misuse by either party. It is possible that establishments may try to falsify readings or data shown on the app to appeal to their target audience, which could cause safety concerns or severe discomfort for a user who makes a decision to enter a business based on this data.

## 4.3 General Safety Concerns

As mentioned above regarding misuse of this device, a physical object could pose a tripping hazard if it is installed near a floor, or a general hazard if placed in a space that people might bump into it. To mitigate this, we will research required techniques to prevent accidental kicking, bumping, or spillage onto or around our product. In addition, since our product is intended to identify safety concerns regarding sensory input, misclassification of these inputs by our device could lead to safety concerns or severe discomfort of a user who is expecting there to be none. Because of this, our app will specifically need to inform users of the risks associated with relying only on information displayed in our app coming from our devices.

# References


[1] “Epilepsy data and statistics.” *Centers for Disease Control and Prevention*. 30-Sep-2020. [Online]. Available: https://www.cdc.gov/epilepsy/data/index.html. [Accessed: Feb. 8, 2022].

[2] “Data and statistics about ADHD.” *Centers for Disease Control and Prevention*. 23-Sep-2021. [Online]. Available: https://www.cdc.gov/ncbddd/adhd/data.html. [Accessed: Feb. 8, 2022].

[3] “GL55 Series Photoresistor,” *Digilent,* [Online]. Available: https://digilent.com/reference/_media/ni/photoresistor_ds.pdf [Accessed: Feb. 21, 2022].

[4] “ACM Code of Ethics and Professional Conduct”. Association for Computing Machinery Website. [Online]. Available: https://www.acm.org/code-of-ethics [Accessed Feb. 7, 2022].

[5] “IEEE Code of Ethics”. IEEE Website. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html [Accessed Feb. 7, 2022]