Can U Cardio?

B2: Ian Brito, Ian Falcon, Nataniel Arocho-Nieves
18-500 Capstone Design, Spring 2023
Electrical and Computer Engineering Department
Carnegie Mellon University

Product Pitch

Can U Cardio? is a system capable of providing occupancy and availability of gym cardio machines in real-time. This system is specifically designed for stationary bikes and treadmills. It aims to reduce the time wasted in the gym as well as help students plan cardio workout sessions in a time efficient manner. Incorporating sensors and wireless communications, this system reflects this data via a web application available to students and staff.

With our users and their requirements in mind we designed our system centered around detection accuracy, detection delay, usage time, and commodity. Our system can detect and display occupancy with an average accuracy of 98.8% and an average delay of 3.19 seconds through 4 days of continuous use (always on) or 6 days of interrupted use (17-hour intervals per day, typical of the UC gym hours) all while not interfering with gym-goers' workouts and minimally occupying space on treadmill and bike dashboards.

System Architecture

Our system incorporates two main modules – a sensor module and a software stack. The sensor module consists of an IR sensor, a NodeMcu microcontroller, and a 5V power supply. The software stack is a Django web application running on an EC2 instance and utilizes Python, ReactJS, and MySQL.

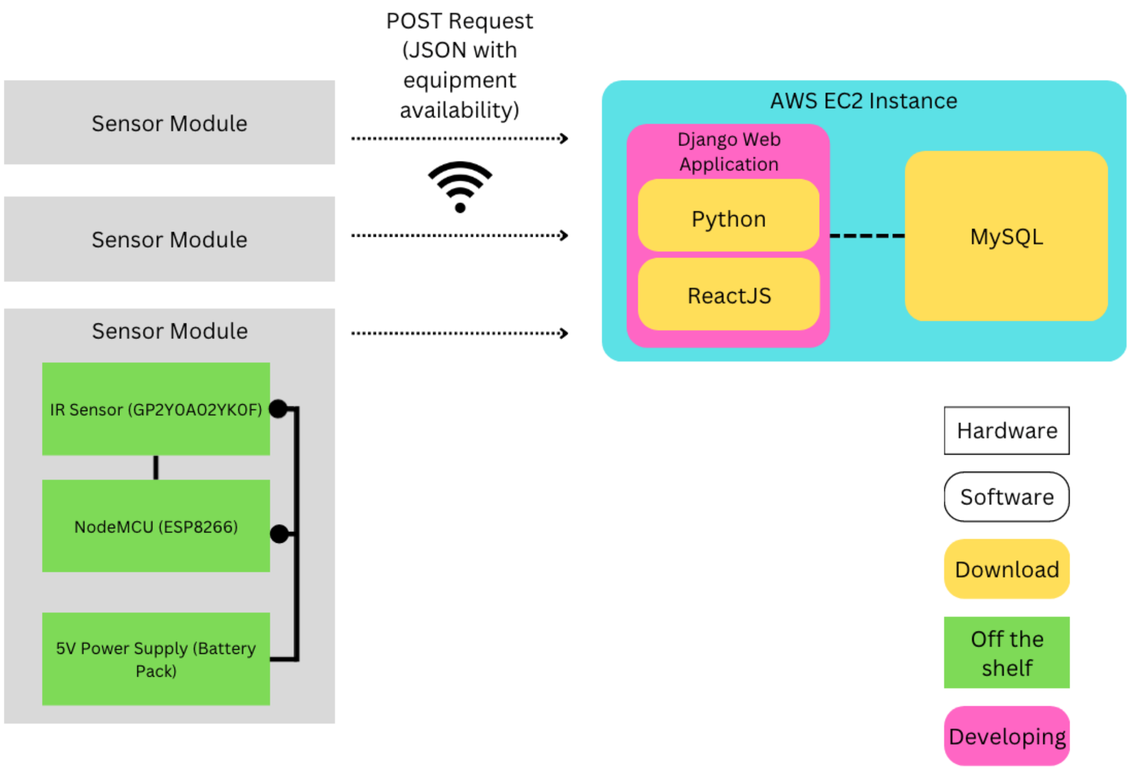


Fig 1. System Block Diagram

Conclusions & Additional Information

Overall, our system met our expectations based on the minimum viable product we wished to deliver of 4 sensor modules and a web app, and the use-case requirements we determined. Detection accuracy and delay requirements were met with over 90% accuracy and less than 30 seconds of delay. When it comes to usage and battery life our requirements were conservative yet met. In terms of the invasiveness of the system it is fairly noninvasive, yet our mounting spots and encasings could be a bit cumbersome for some users since they occupy some dashboard space that is used by gym goers for other purposes like holding their phones or other belongings.

http://course.ece.cmu.edu/~ece500/projects/s23-teamb2/

So, a future project might develop an implementation that improves battery life to a full week of usage without recharge regardless of method of utilization (always on or interrupted use). Said project could also incorporate a better and more secure mounting mechanism that reduces dashboard space used and increases module stability With this in mind, further expansion could also be done to include other machines in the UC gym, such as other cardio machines and weight machines on the first floor.

System Description

A single sensor module is comprised of a Sharp IR distance sensor (GP2Y0A02YK0F), a 100µF capacitor to stabilize the power supply line, a NodeMcu (ESP8266) microcontroller, and a 5V power supply comprised of 4 rechargeable Ni-MH AA batteries with a total 11,200 mAh. The module is housed in a 3D-printed encasing specific to a treadmill or bike installation.

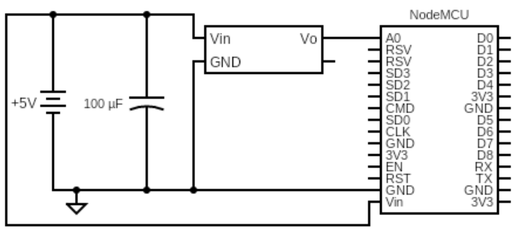


Fig 2. Sensor Module circuit diagram

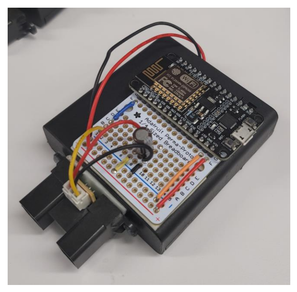


Figure 3. Sensor Module

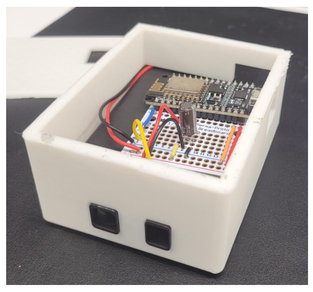


Figure 4. Treadmill Module Encasing

The IR sensor feeds an output voltage to the NodeMcu's A0 analog port which utilizes an analog to digital converter ADC to process the voltage. Then, our algorithm determines occupancy and communicates this to the web app. The web app utilizes the Django framework (Python) and is hosted on an Apache web server on an EC2 instance. Data is received from the sensor module in the form of a POST request, which the web application uses to update its models stored in a MySQL database. AJAX is used for real-time updating of equipment status.

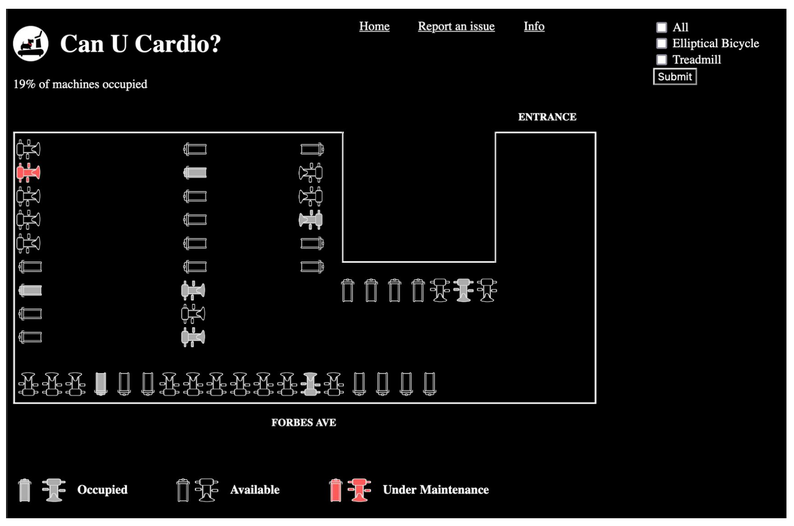
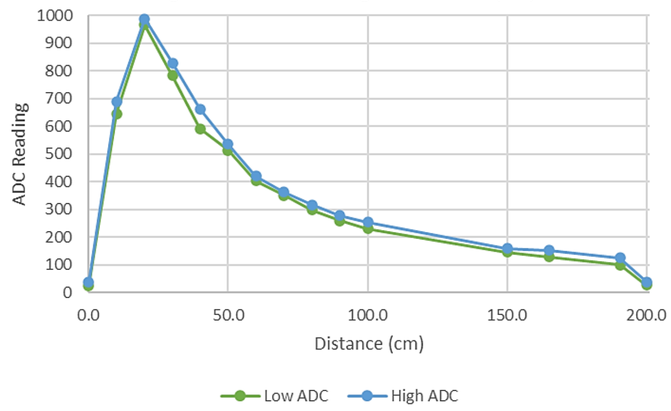


Figure 5. Web-app that displays an occupancy map based on the UC gym cardio room layout as well as the percentage of machines in use. It has filters for treadmill and bikes. In addition, there is an option for a user to report an issue with a machine. The info tab displays occupancy history based on weekly data.

System Evaluation

Our first phase of testing revolved around determining distance thresholds for sensor calibration using the NodeMcu. The analog to digital converter (ADC) reading was the basis for our detection scheme. Figure 5 illustrates the correlation between distance and ADC reading which itself correlates to the voltage outputted by the sensor. With the data from this figure, we calibrated the sensor by setting specific ADC readings that corresponded to our distance thresholds.

Figure 5. ADC Reading vs Distance (cm)



With our thresholds set, we measured our detection accuracy using four different sets of trials highlighted in Figure 6. First, we stood in our determined detection range to ensure that a person using the machine is always detected. Then, we tested standing outside the detection range (but in its line of sight) to ensure that it would not have false detections. Next, we performed "pass-bys" mimicking a user walking through a corridor out of range to guarantee no false detections. Finally, we tested the system's surroundings by standing in range of the sensor, but at a wide angle to ensure adjacent machines would not be detected. We observed a 99.4% average accuracy with our tests.

	# of successful trials out of 10			
Sensor Module #	**On-range stand-in**	**Out-of-range stand-in**	**Pass-by**	**Surroundings (wide angle)**
1 (treadmill)	10	10	10	10
2 (treadmill)	10	10	10	10
3 (bike)	10	10	10	10
4 (bike)	10	9	10	10

Figure 6. Detection Accuracy Testing Results

For detection delay, we tested the amount of time it took for the web app to reflect that a machine had gone from free to busy and busy to free by activating the sensor module and manually timing the web app's response time. Overall, our average detection delay was 3.19 seconds.

	Avg detection delays out of 10 trials (s)		
Sensor Module #	**Free to Busy**	**Busy to Free**	**Overall (avg of BF and FB)**
1	2.37	4.77	3.57
2	4.30	2.93	3.62
3	1.61	3.46	2.55
4	3.92	2.12	3.02

Figure 7. Detection Delay Testing Results

Electrical & Computer ENGINEERING
Carnegie Mellon