

Occupancy Monkey



A Seating Solution

The Use Case:

Currently there is no efficient way to find an open seat in the CMU University Center 2nd floor, especially during peak hours.

Our Solution:

Track table occupancy in real time and display current occupancy on a web application.

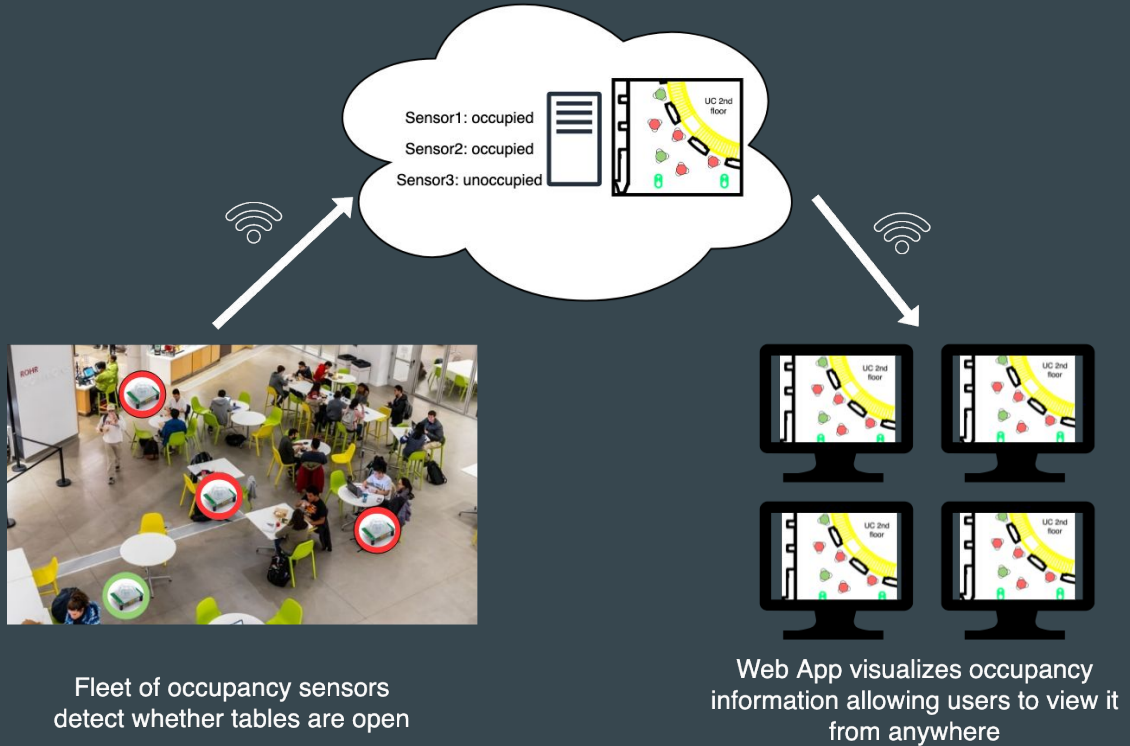


Use Case Implementation

ECE Areas:

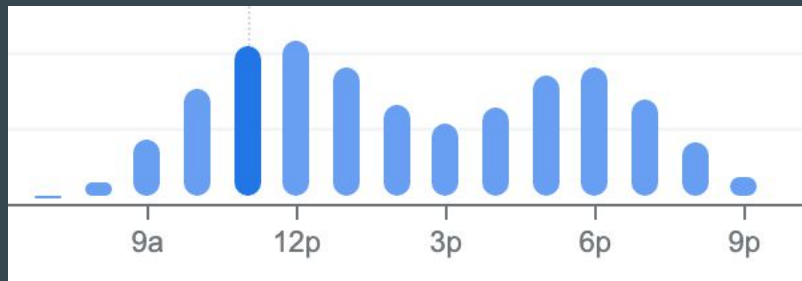
- Software
 - Web Applications
 - Embedded Systems Programming
- Hardware
 - Embedded Systems Architecture

Each sensor's occupancy data is uploaded to the cloud and made accessible to the web app



Use-Case Requirements: Hardware

- 55+ hour battery life for hardware system
 - Product will function for UC peak weekday hours of 9am to 8pm
 - Device will run in ultra low power mode during off-peak hours
 - Batteries will be changed and web app will not operate over the weekend when UC occupancy decreases significantly



Use-Case Requirements: HW/SW Communication

- Occupancy status is accurately updated on web app every minute
 - Trade-off between power usage and web app accuracy
 - Usually takes about a minute for someone to pack up and leave a table

- Hardware will never go more than 30 minutes without sending information
 - Battery saving will likely mean device transmissions are minimized.
 - No L.O.S. will cause incorrect data to be displayed for more than 30 minutes
 - CMU classes exit every 30 minutes. A 30 minute check-in ensures a malfunction could only damage the experience of one wave of students looking for seats

Use-Case Requirements: Web Application

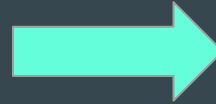
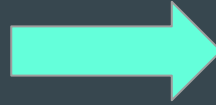
- Web App Front end will support 25 concurrent users
 - We envision the web app experience lasting ~5 minutes
 - Application is designed around crowd mitigation -> we expect lots of concurrent users
 - During peak hours this is the number of people who enter the UC every 5 minutes
- Cloud Deployment can support 80 hardware devices at once
 - Ubiquitous integration would require a device at every table we want to support
 - There are 80 tables in the UC

Technical Challenges: Hardware + HW/SW Communication

Requirements

Device Battery must last for 55+ hours of use

Occupancy status is updated on web app every minute; hardware will never go more than 30 minutes without sending information



Technical Challenges

Sensors and wifi module must draw low power loads during operation without compromising performance

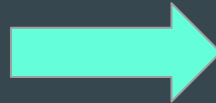
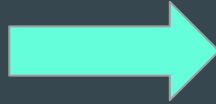
Hardware must be able to quickly and accurately detect occupancy while filtering out noise

Technical Challenges: Web Application

Requirements

Cloud deployment can support 80 hardware devices at once

Web App deployment can support 25 users at once



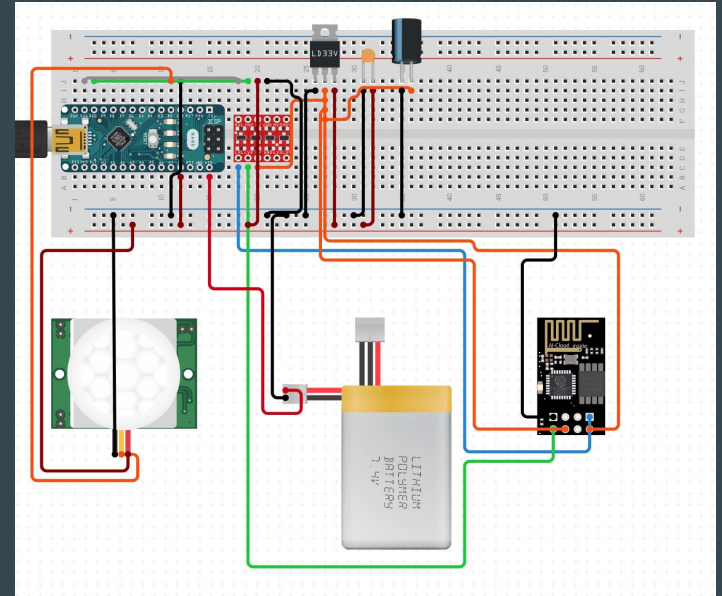
Technical Challenges

Need to ensure proper and accurate communication between device and server for high number of hardware devices

Server will need to cope with high influx of concurrent requests

Solution Approach - Hardware

- IoT device
 - Arduino + esp8266 vs esp8266
 - Battery
 - Sensor
- Sensor options:
 - **PIR infrared motion sensor** (displayed in the diagram on the right)
 - Thermal sensor
 - thermometer
 - Thermal imager
 - Wifi/bluetooth RSSI based distance detection



Solution Approach - Web Application

- AWS hosted
- Web application that displays availability of tables in UC second floor
 - Django web framework
- MySQL database to store occupation status



Testing Verification and Metrics

Requirement	Test	Metric
Device can run for 55 hours without battery replacement	Run hardware device continually with sensors and status updates	Record time until battery died
Device updates status on web app within a minute	Run a single hardware device linked to web app and spoof a status update	Record time from change in status to website indicating change in status
80 hardware device can be supported simultaneously	Create a test program to send several devices worth of information	Record number of simulated nodes at which data is first lost
25 users can be supported on web app concurrently	Create a test program to simulate web traffic	Record EC2 CPU credit balance loss

Tasks + Division of Labor

Jake Cerwin

- Create HW/SW interface for sending data from devices to web
- Minimize power consumption of sending data

Ryan Huang

- Order and assemble all hardware components
- Set up and read data from all sensors to determine occupancy

Angela Zhang

- Make UI for map of UC - Angela
- Front end design and implementation

Schedule

	Assigned	Progress	FEBRUARY 2022																												MARCH 2022																															APRIL 2022																															
			1	2	3	4	7	8	9	10	11	14	15	16	17	18	21	22	23	24	25	28	1	2	3	4	7	8	9	10	11	14	15	16	17	18	21	22	23	24	25	28	29	30	31	1	4	5	6	7	8	11	12	13	14	15	18	19	20	21	22	25	26	27	28	29	2	3	4	5	6	9	10																				
Occupancy Monkey		0%	[Progress bar]																																																																																										
▼ Presentations		0%	[Progress bar]																																																																																										
Project proposal	Angela, Jake Cerwin, Ryan	0%	[Task bar]																																																																																										
Design presentation	Angela, Jake Cerwin, Ryan	0%	[Task bar]																																																																																										
Final presentation	Angela, Jake Cerwin, Ryan	0%	[Task bar]																																																																																										
▼ Hardware		0%	[Progress bar]																																																																																										
Create a hardware BOM and order parts	Ryan	0%	[Task bar]																																																																																										
Determine esp8266 vs esp8266 + Arduino	Jake Cerwin, Ryan	0%	[Task bar]																																																																																										
Create modular power supply with battery	Ryan	0%	[Task bar]																																																																																										
Create switch driven hardware unit for early...	Ryan	0%	[Task bar]																																																																																										
Create sleep and awake states for hardware	Ryan	0%	[Task bar]																																																																																										
Ensure Arduino read properly from all senso...	Ryan	0%	[Task bar]																																																																																										
Create Occupancy detection algorithm with...	Ryan	0%	[Task bar]																																																																																										
▼ HW/SW Communication		0%	[Progress bar]																																																																																										
Determine AWS IoT Core vs HTTP Post	Jake Cerwin	0%	[Task bar]																																																																																										
Send and receive some data from esp8266 ...	Jake Cerwin	0%	[Task bar]																																																																																										
Create artificial stress test to simulate multip...	Jake Cerwin	0%	[Task bar]																																																																																										
Create check in protocol to ensure no devic...	Jake Cerwin	0%	[Task bar]																																																																																										
design low power wifi updates	Jake Cerwin	0%	[Task bar]																																																																																										
Integrate HW data into webapp	Jake Cerwin	0%	[Task bar]																																																																																										
▼ Software		0%	[Progress bar]																																																																																										
Purchase EC2 instance	Angela	0%	[Task bar]																																																																																										
Front end Django app	Angela	0%	[Task bar]																																																																																										
Set up Github repo	Angela	0%	[Task bar]																																																																																										
Design UI for indicating occupancy	Angela	0%	[Task bar]																																																																																										
Determine if single EC2 can support use ca...	Jake Cerwin	0%	[Task bar]																																																																																										
Create artificial Stress test to simulate traffi...	Jake Cerwin	0%	[Task bar]																																																																																										