Event Attendance Tracker

Team 13:
Anand Sunderrajan, Eric Layne,
Mason Edwards

What is it?

- A system to track attendance at booths during an event such as CES or a career fair
- Utilizes Bluetooth Low Energy (BLE) to handle booth detection and communication
- Records what booths a user attends at event, provides additional information about those booths
- Provides attendance statistics to the booth presenters

Objective

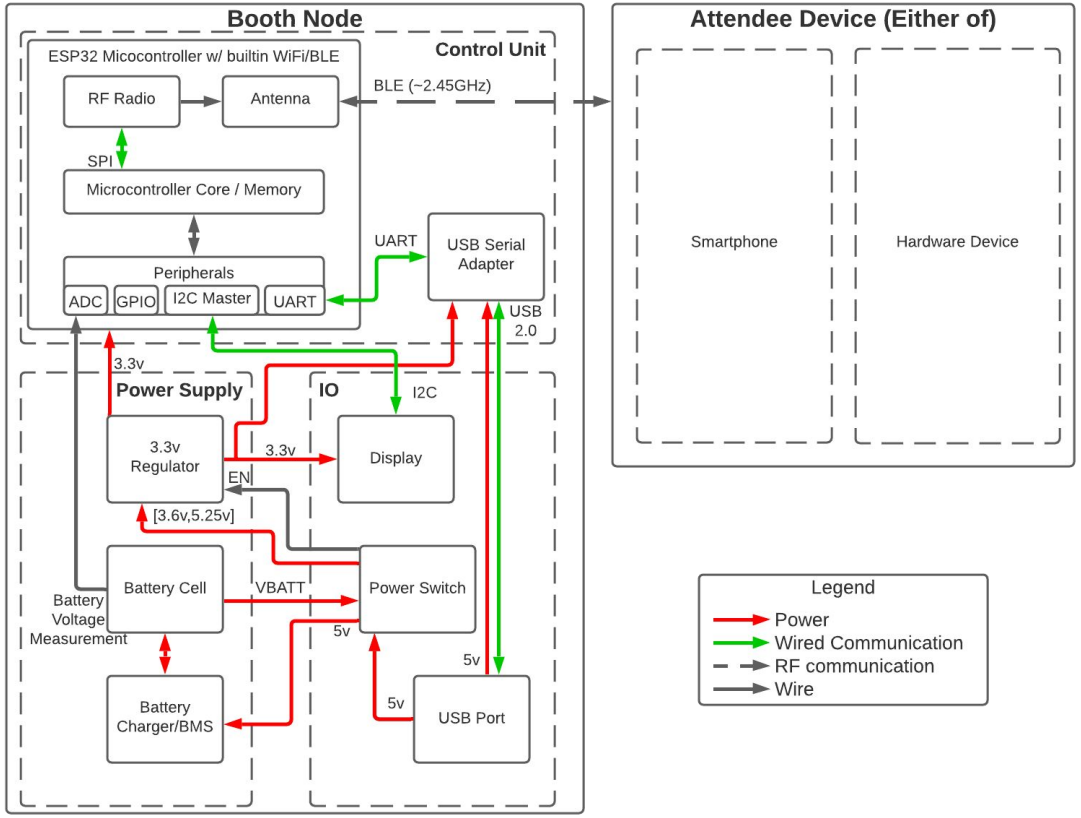
- Event attendees often see a large number of booths
  - Recalling all the booths after the event can be difficult if not impossible
- Requirements
  - Needs to run on battery power for at least 4 hours
  - Log attendance at correct booth when multiple are available
  - Send up to 1kB of data from the booth to a user's device in under 250ms

Design Choices

- BLE over WiFi or Cellular
- Tilted OLED display rather than LCD
- Lithium polymer battery cell rather than NiMH, Lithium Ion, etc

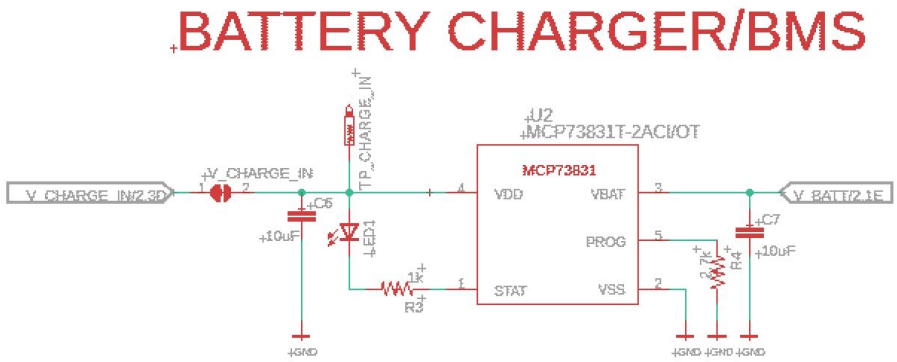
System Overview

- Power Supply
- Control Unit
- I/O Subsystem
- Event Attendee
  - Attendee Device
  - Smartphone App



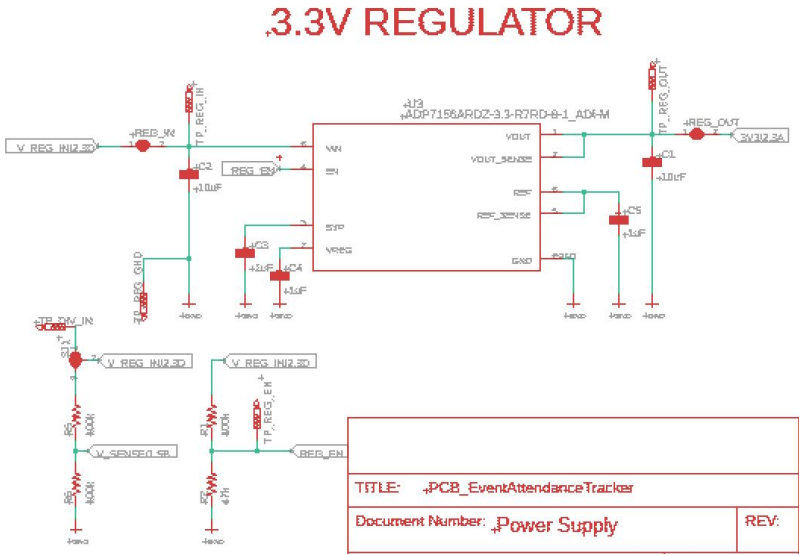
Power Supply - BMS

- Safely charges internal lithium polymer battery cell
- Uses constant current charging until reaching a preset voltage, then uses constant voltage to finish and maintain charge



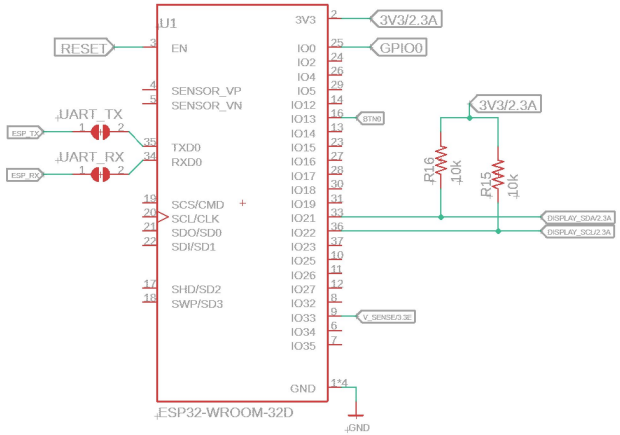
Power Supply - Regulator

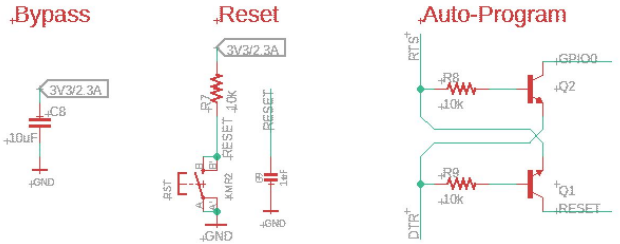
- Regulates battery or USB voltage to 3.3V
- Configurable hardware voltage cutoff (set at 3.5V)



Control Unit

- ESP32 microcontroller (built in RF radio)
- Handles BLE communication
- Manages device hardware
  - OLED display
  - Battery voltage monitoring
  - USB port serial communication

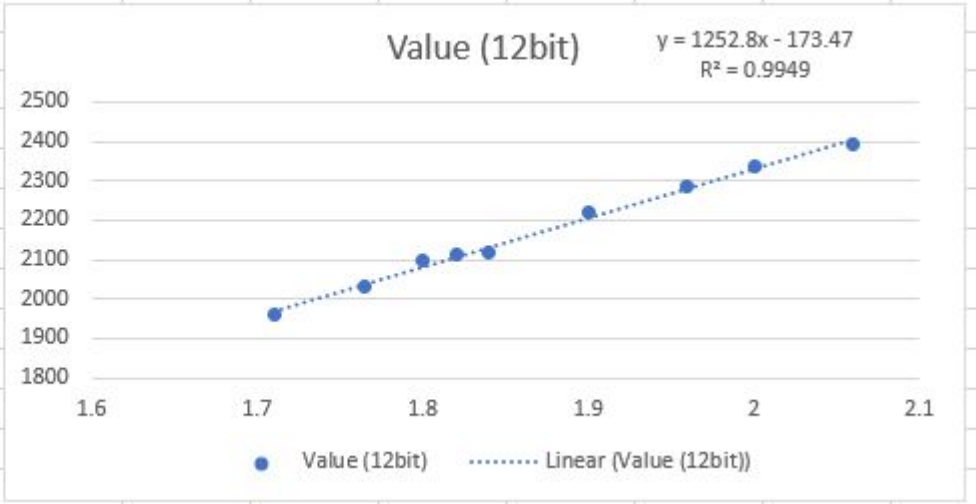




Control Unit

- Battery voltage ADC calibration

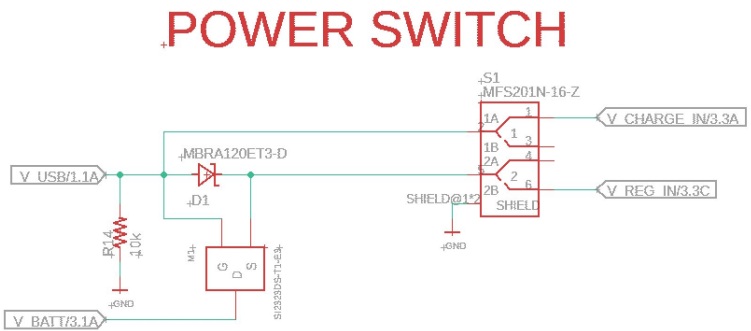
	A	B	C
1	Voltage[v]	Value (12bit)	
2	1.71	1959	
3	1.765	2034	
4	1.8	2096	
5	1.82	2112	
6	1.84	2118	
7	1.9	2218	
8	1.96	2288	
9	2	2335	
10	2.06	2395	
11			
12			
13			



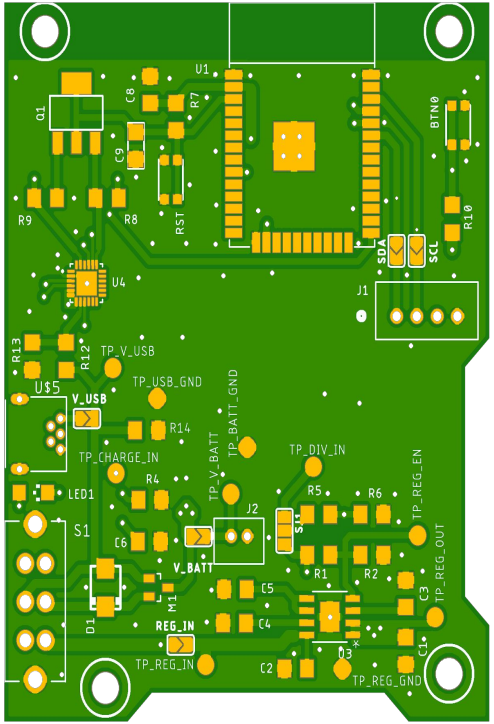
I/O Subsystem

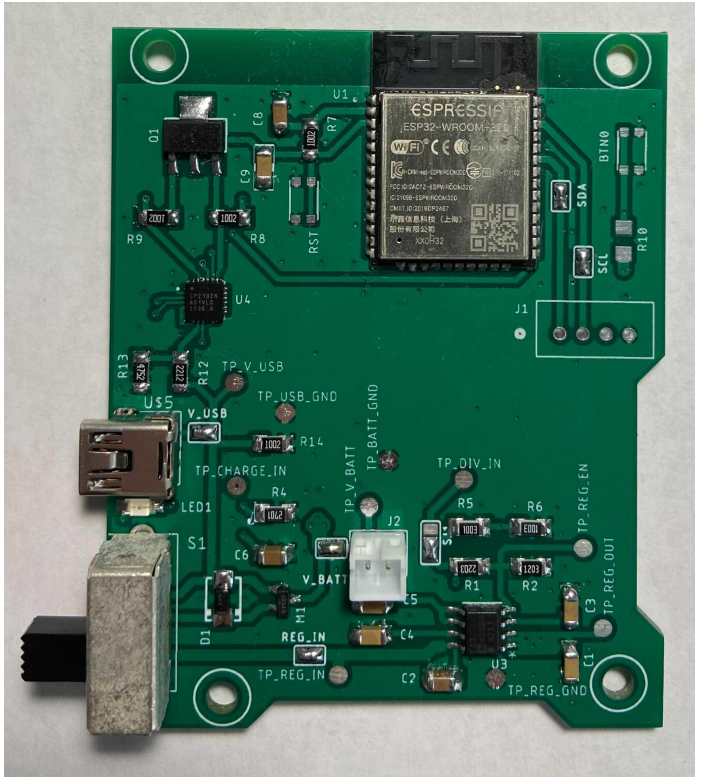
- OLED display shows the booth name, battery percentage, and attendee count, and a low battery screen
- Switch circuit uses USB power when available and ensures battery safety
- USB port allows communication

  with the device for extracting data



PCB


U1
Q1
C8
C9
R7
R9
R8
RST
BTN0
R10
SDA
SCL
U4
J1
R13
R12
TP_V_USB
U$5
V_USB
TP_USB_GND
TP_BATT_GND
R14
TP_CHARGE_IN
TP_V_BATT
TP_DIV_IN
R4
R5
R6
TP_REG_EN
LED1
S1
J2
C6
V_BATT
M1
R1
R2
C5
C3
TP_REG_OUT
D1
REG_IN
C4
D3
TP_REG_IN
C2
C1
TP_REG_GND


ESPRESSIF
ESP32-WROOM-32E
U1
Q1
C8
C9
R7
R9
R8
RST
BTN0
R10
SDA
SCL
U4
J1
R13
R12
TP_V_USB
U$5
V_USB
TP_USB_GND
TP_BATT_GND
R14
TP_CHARGE_IN
TP_V_BATT
TP_DIV_IN
R4
R5
R6
TP_REG_EN
LED1
S1
J2
C6
V_BATT
M1
R1
R2
C5
C3
TP_REG_OUT
D1
REG_IN
C4
TP_REG_IN
C2
U3
TP_REG_GND

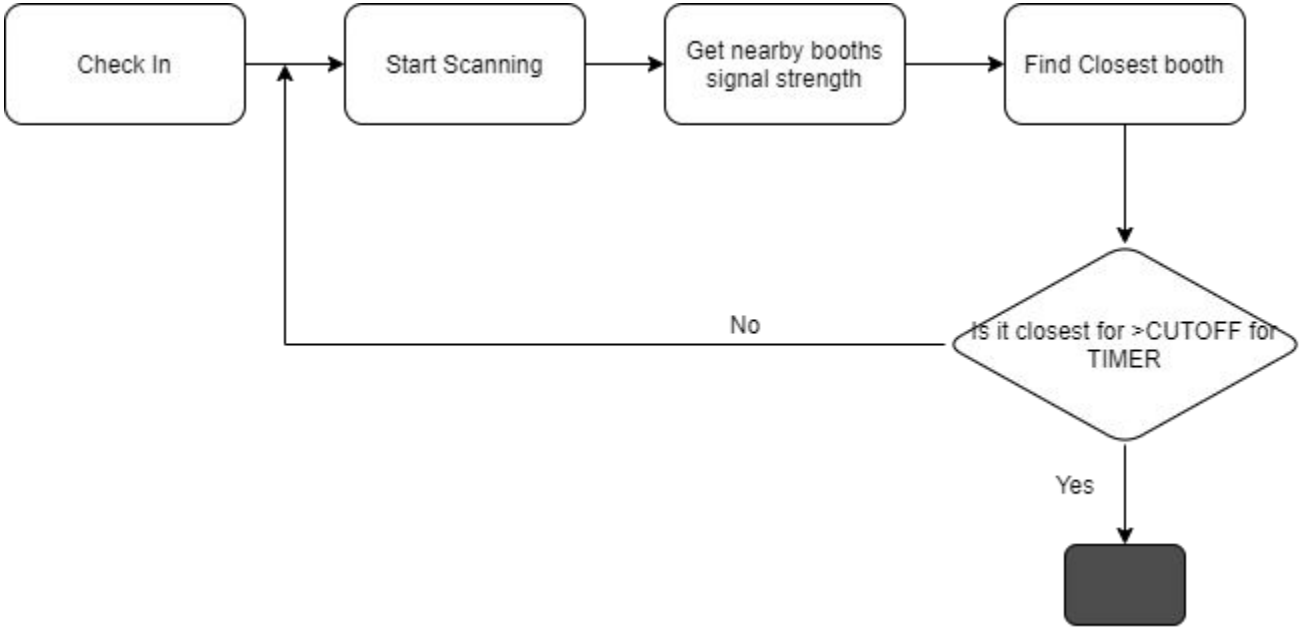
Software Design

- Attendee Device
  - Utilizes the same hardware as a booth node
  - Battery/USB powered
  - Decision making is self contained
- Smartphone App
  - Currently only for android devices
  - Utilizes devices owned by attendees
  - Lightweight with low battery and memory consumption

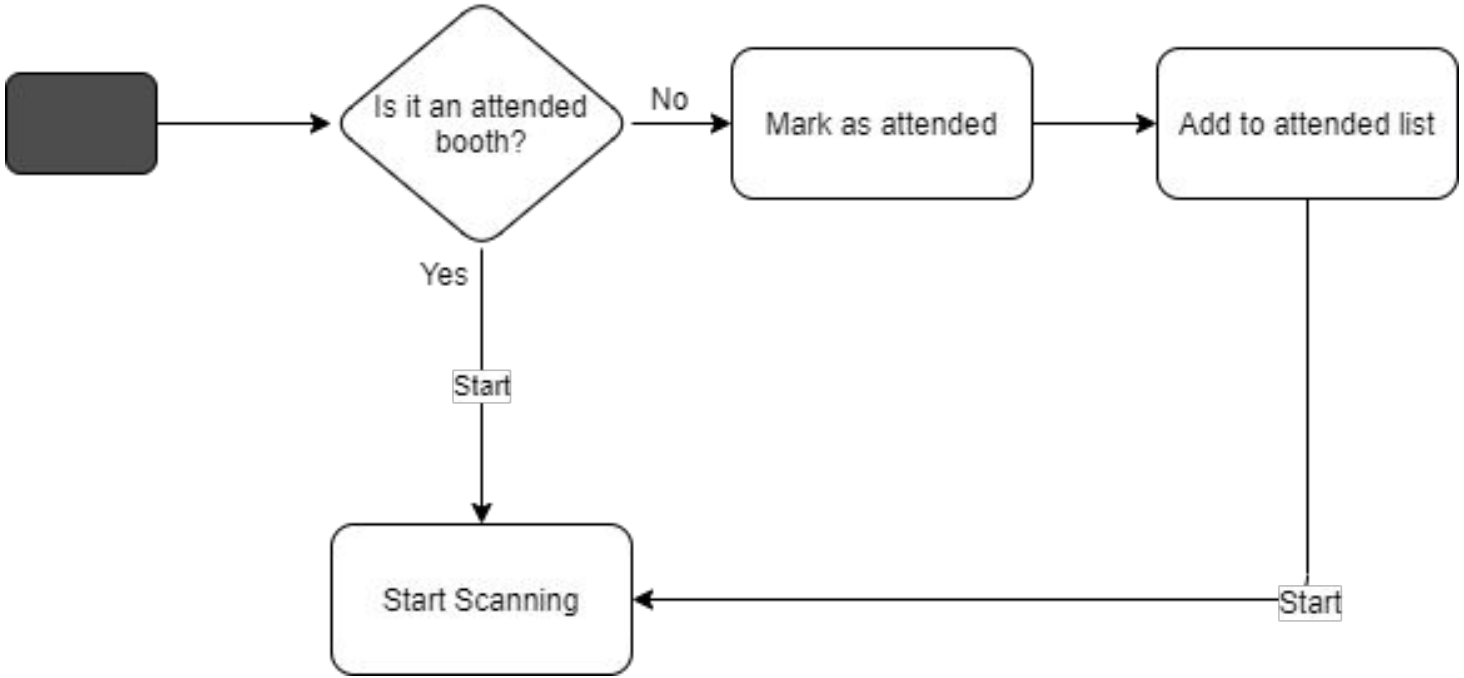
Requirements

- Detect nearby booths and display them with their respective signal strengths
- Distinguish between nearby booths and determine the closest
- Flag the closest booth as attended after the user configurable parameters are met

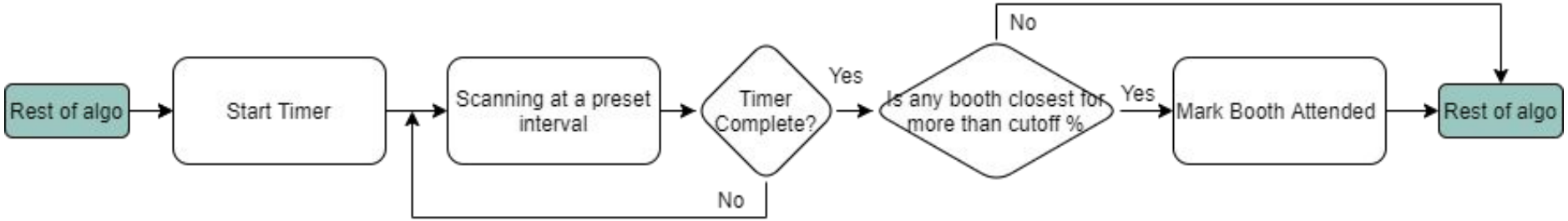
Decision Making Algorithm



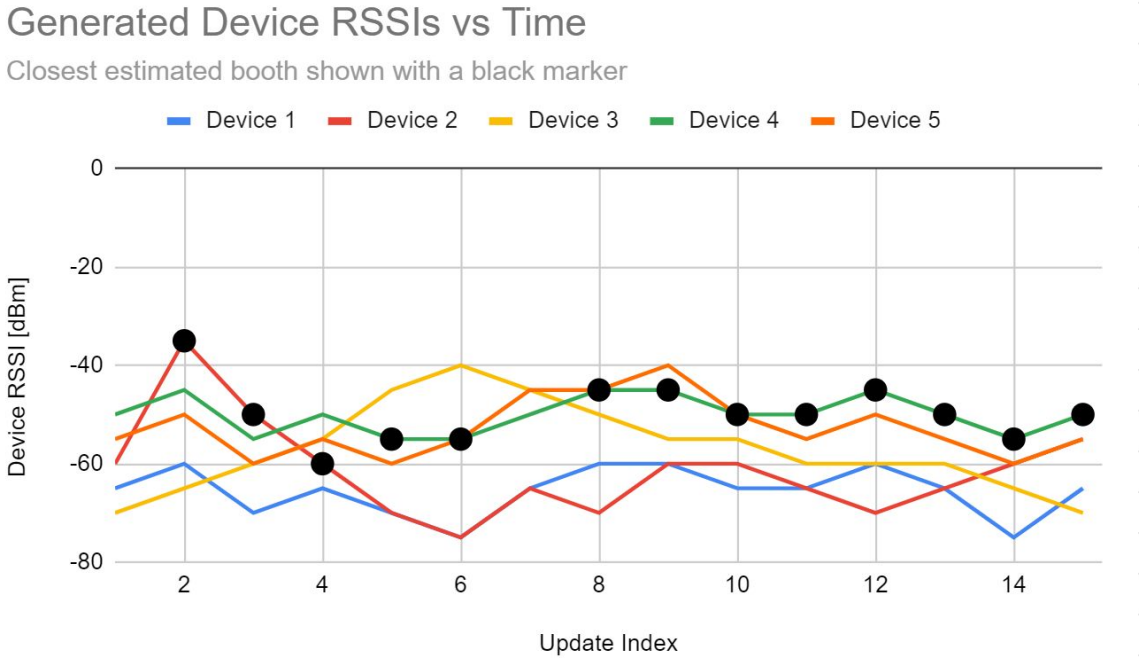
Decision Making Algorithm



Decision Making Analysis Algorithm



Decision Making Analysis

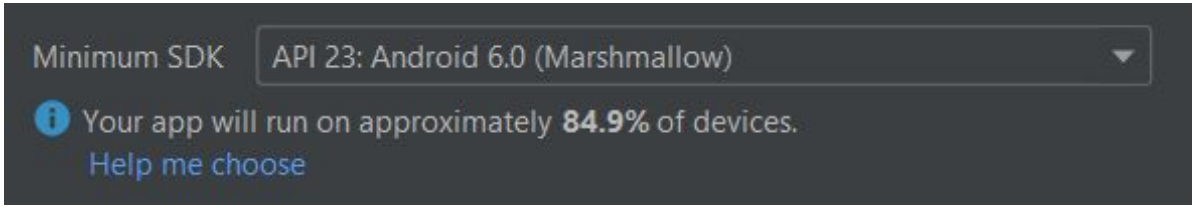


Attendee Device

- Utilizes same hardware as booth node
- Software built using C++
- Displays nearby booths and attended booths on console
- Modular programming allows for user configurable variables

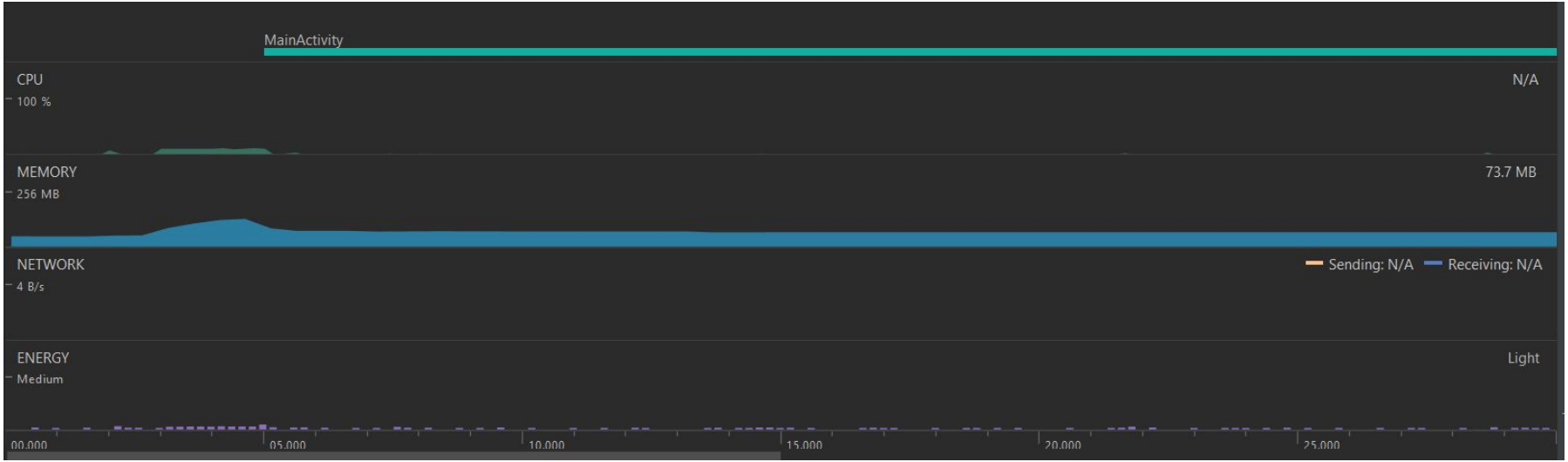
Smartphone App

- Built using Java
- Bluetooth, background, and location permissions dependent on API level
- Compatible with Android 6.0+ (API 23+) ~84.9% of devices



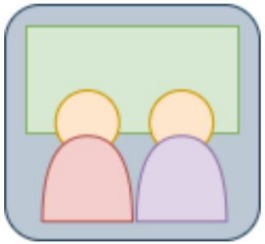
Smartphone App

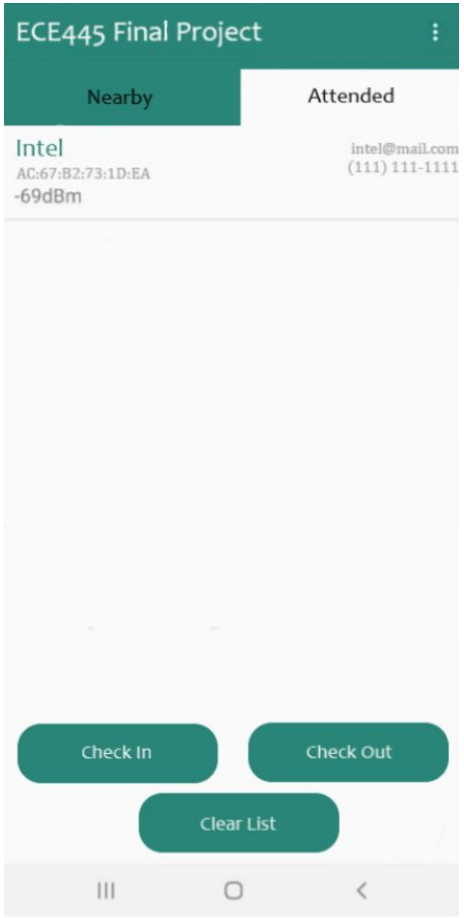
- Lightweight with low CPU, energy, and memory utilization



Smartphone App

- Material design compatible with different screen sizes
- Displays nearby and attended booths separately under different tabs
- Buttons for check-in, check-out, and clearing the list
- Custom icon





Successes

- Booth hardware
- Block level design
- Decision making algorithm

Challenges

- Decision making algorithm
- Using BLE with RSSI values to denote signal strength
- The smartphone app

$$DevScore = (1 - \alpha) * DevScore + \alpha * \frac{(RSSI - avg)}{std.\ dev.}$$

$$distance = 10^{\frac{A-RSSI}{10n}}$$

Future Hardware Development

- Create a smaller hardware device available to event attendees
- Add SD card/USB for offloading data or configuring the device
- Reduce the cost by exchanging overrated components
  - Drive decisions based on actual device power requirements

Future Software Development

- Improving usability, and resource consumption of the app
- Porting to iOS devices
- Porting for use on wearables
- Creating database containing booth information for each event
- Web interface for accessing data for presenters and attendees

SWOT Analysis

Strengths
Low cost Small form factor and extremely portable Provides both software and hardware solutions

Weaknesses
Uses RSSI values for signal strength and decision making Chance for booth detection false positives

Opportunities
Event tracking Contact tracing

Threats
Privacy concerns

Thank you!

Questions?