
LUMIRAIL

Project Description

- Create captivating LEGO art piece depicting Massachusetts Bay Transportation Authority (MBTA) map
- Provide real-time information of the subway system through LEDs
  - Precise locations of trains within the MBTA network
  - Status of each train station
- Offer commuters and enthusiasts an interactive and informative way to experience public transit

Development Team

Jake Greenbaum
(Team Lead)

Android App Development, Bluetooth Control

Chris Fisher

PCB Design, LED Display Integration

Zachary Richards

Map Design & Construction

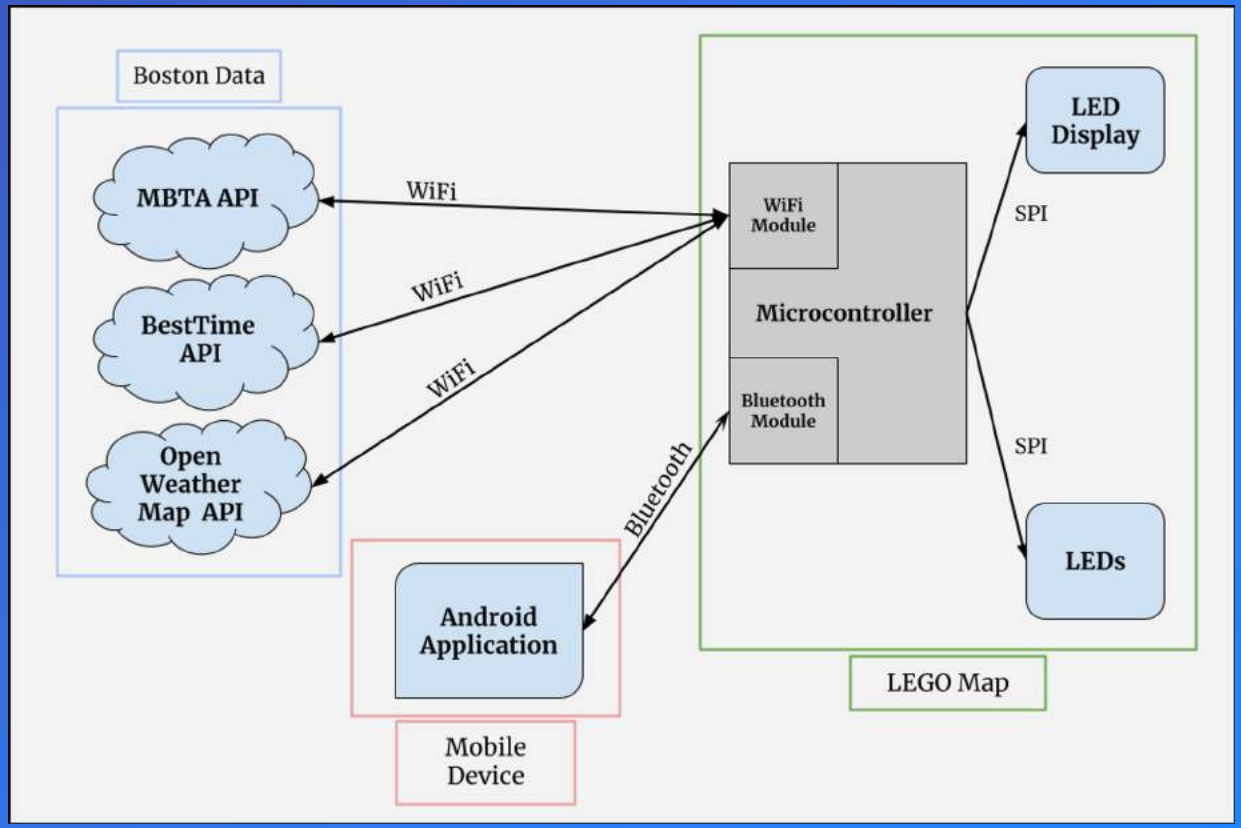
Jack Shoemaker

WiFi Control, API Control and Data Parsing

Sam Ng

LED Programming

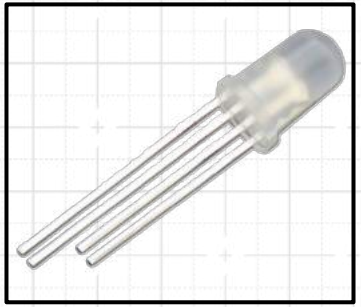
Block Diagram



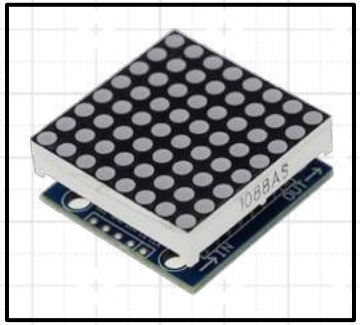
Components

- ESP32-WROOM-32-N4, Microcontroller

- PL9823 Addressable LEDs, Train Station Markers



- Max7219 Dot Matrix, LED Display



PCB

Headers

Microprocessor

Headers

PCB Dimensions:
2.5” x 2.5”

Micro USB Port

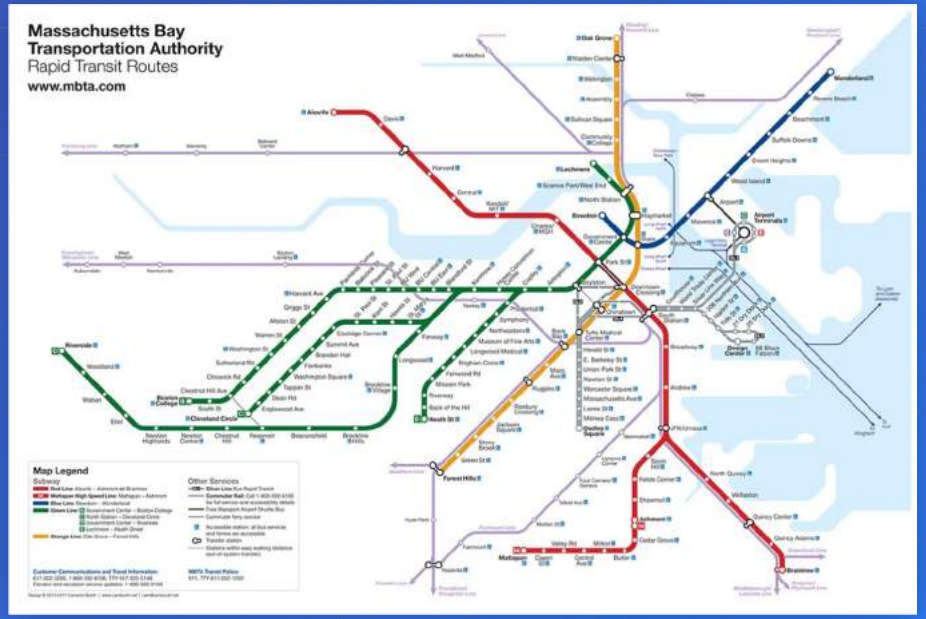
On/Off Switch

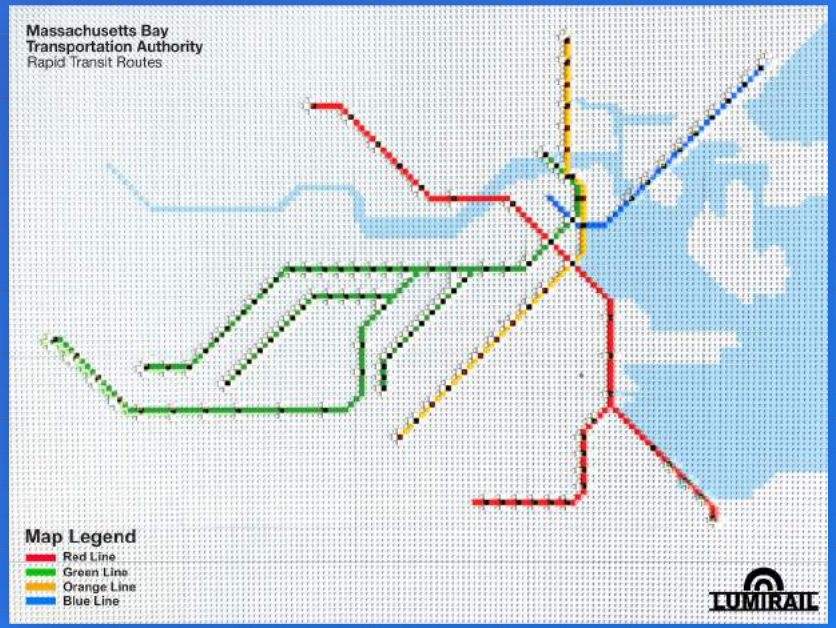
Buttons

DC Power Jack

Physical Map Design and Construction

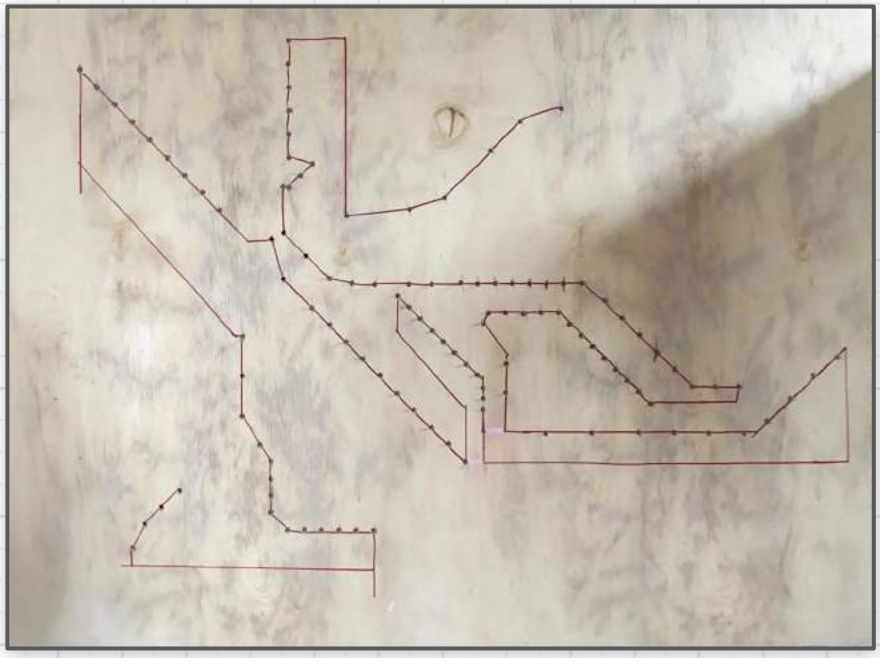
MBTA Map Layout

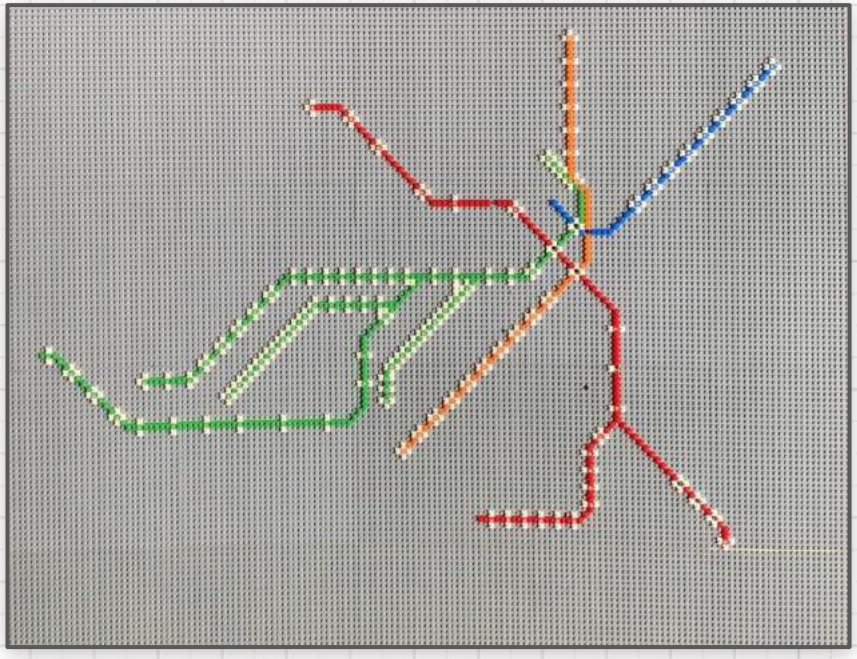




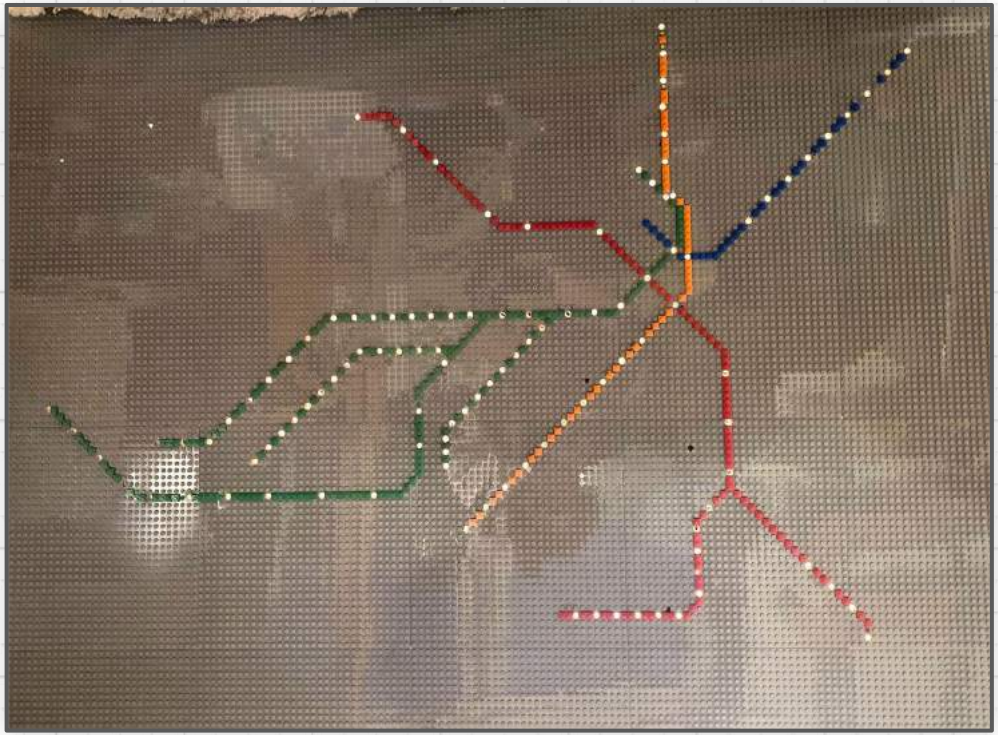
(48in x 36in)

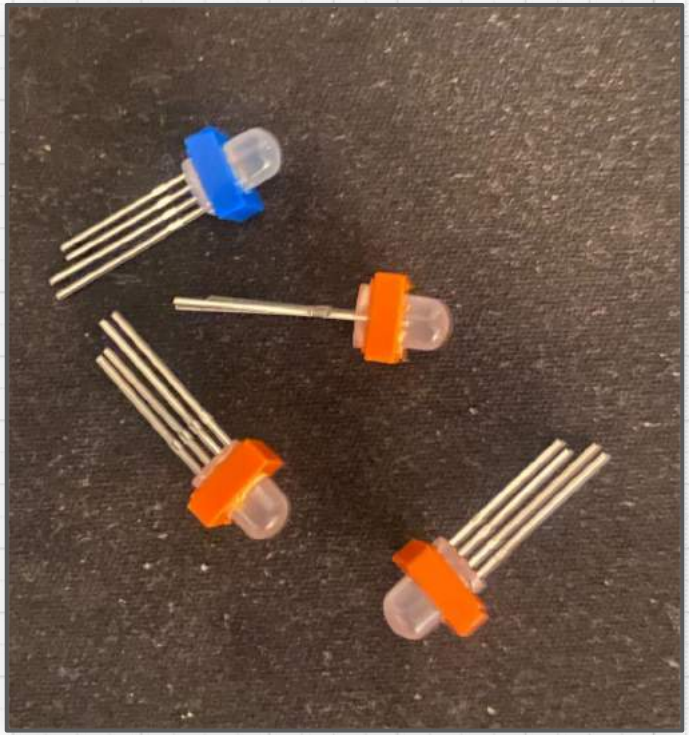
Map Construction



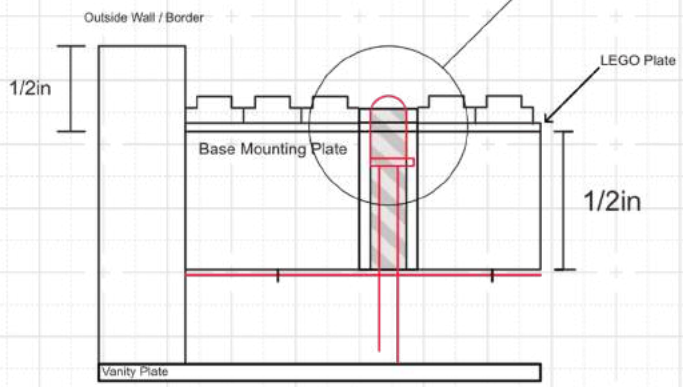


LED Mounting

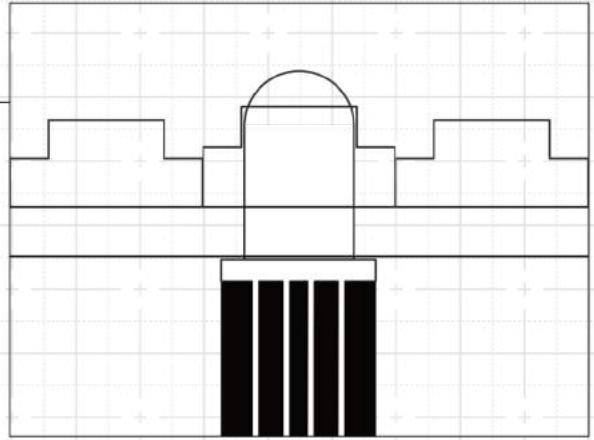




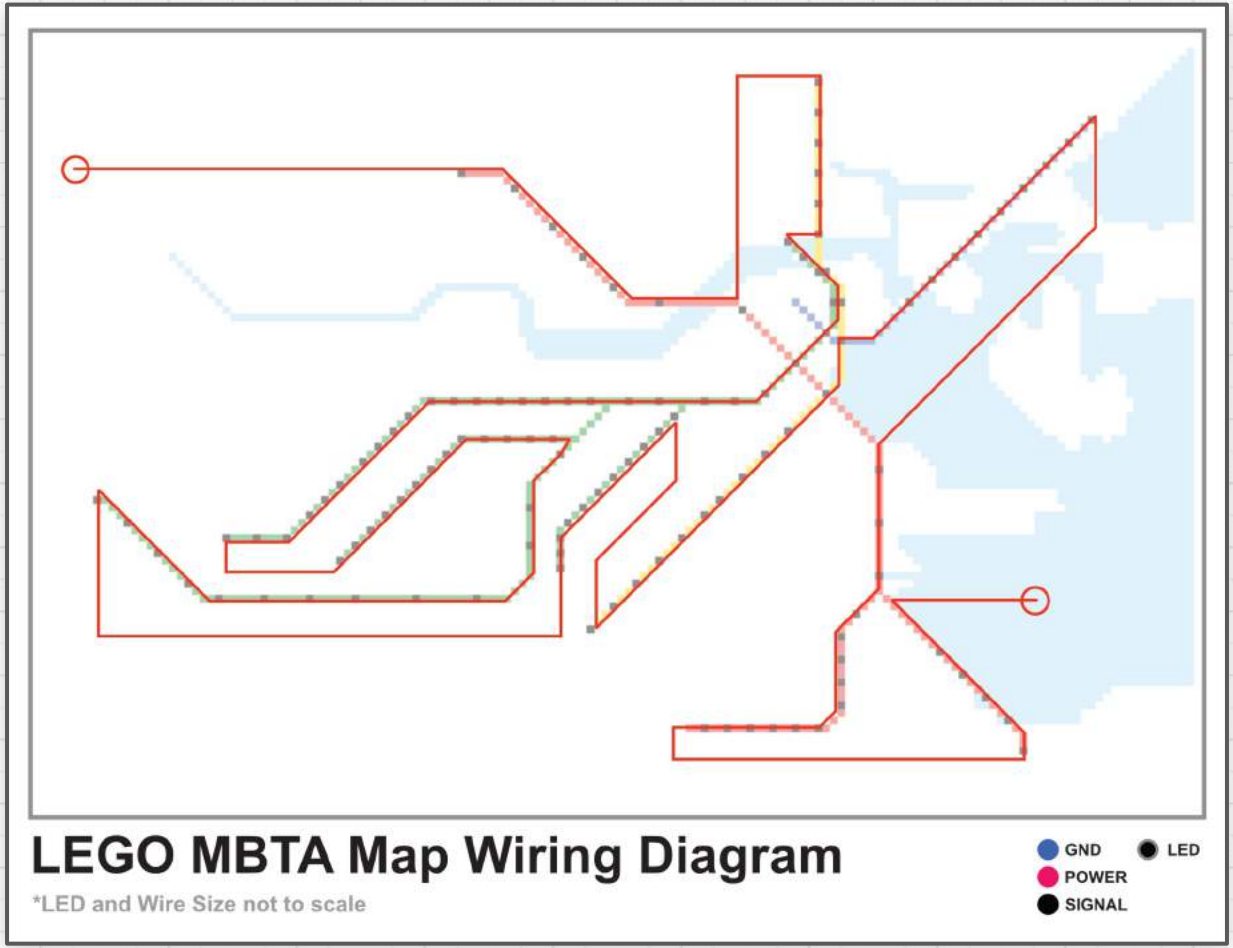
LED Mounting Schematics



Blown up LED Mounting Side View



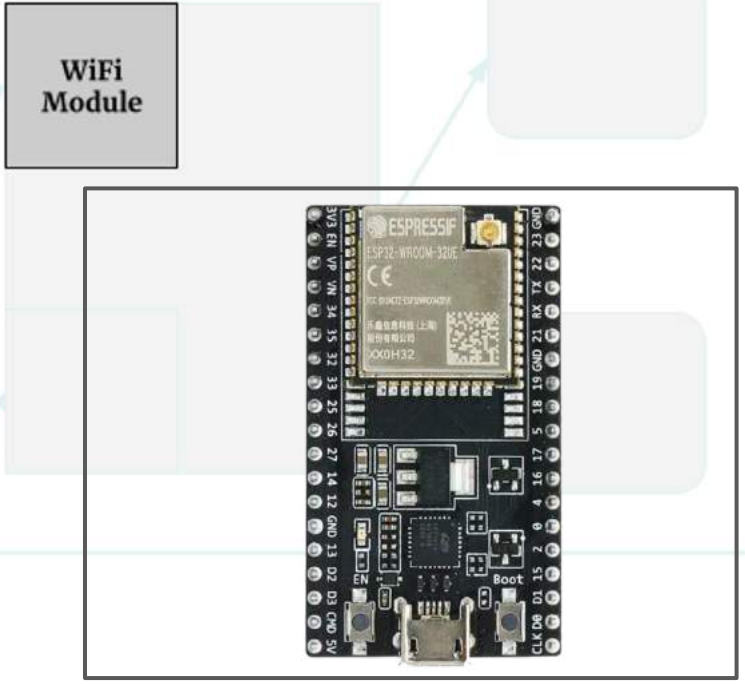
Bottom View of LED Series


LEGO MBTA Map Wiring Diagram
*LED and Wire Size not to scale
GND
LED
POWER
SIGNAL

WiFi Connection and Data Sources

ESP32 Integrated WiFi Module

- 802.11b/g/n capable WiFi Module
- ESP32 connects to local WiFi network using login information provided via user input, then sends HTTPS requests to corresponding APIs receiving transmitted data



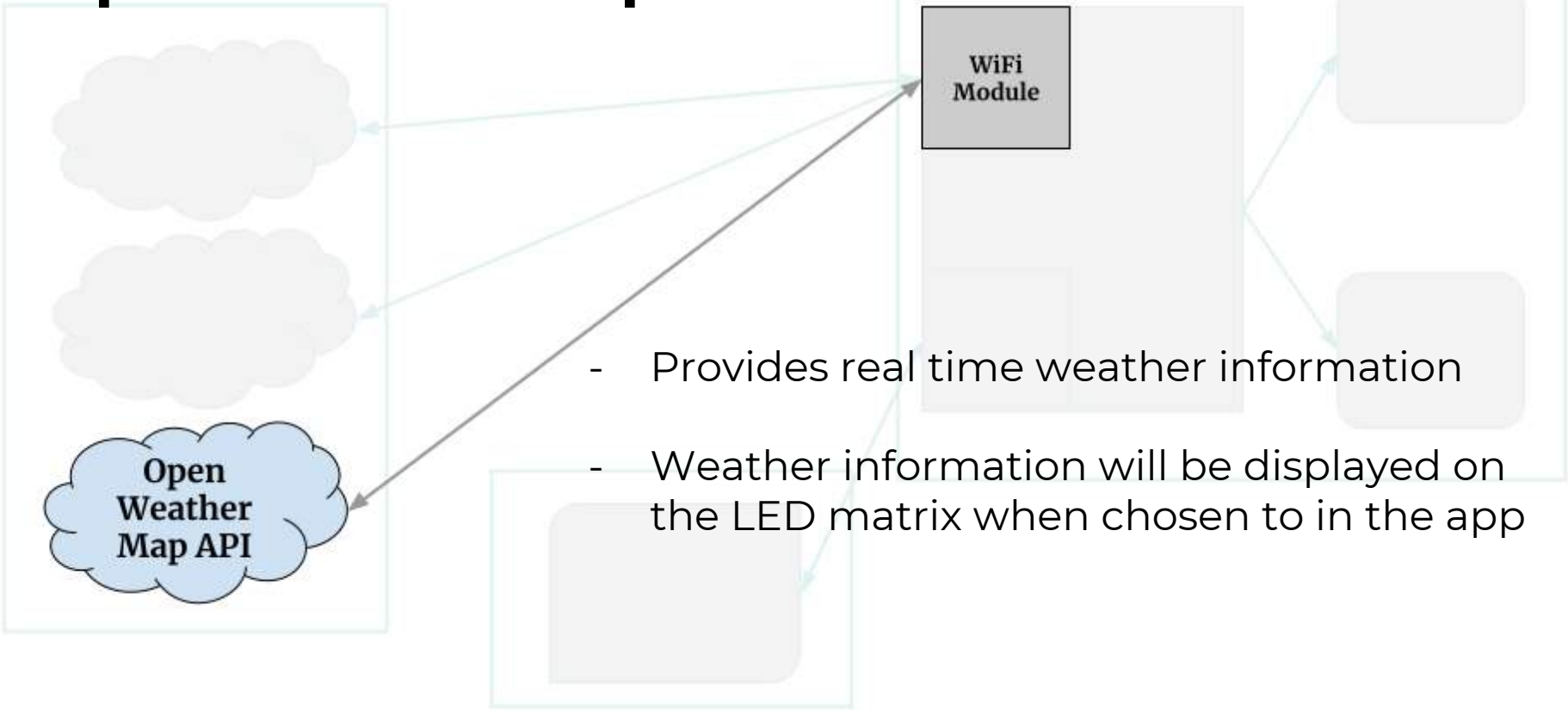
MBTA API

- Provides live vehicle information in Boston
- Filtered to only contain positions and directions of the train and subway system (displayed on LEDs in train mode)
- Information about next train arrival at chosen stop extracted (displayed on LED matrix display)

BestTime API

- Provides information about the current busyness levels at businesses
- Busyness info displayed when selected on Android application

OpenWeatherMap API

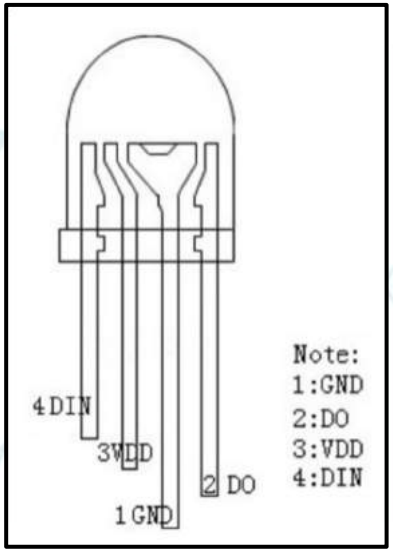


- Provides real time weather information
- Weather information will be displayed on the LED matrix when chosen to in the app

Train Station LEDs

LEDs

- PL9823
  - Addressable RGB LED
- Use similar protocol to WS2818
  - Using datasheet as reference
- Communicate to using SPI
  - Each bit of data represented by 3 bits in SPI
    - High = 110
    - Low = 100
  - Each color takes one byte of data, leading to each color being represented by 3 bytes in SPI



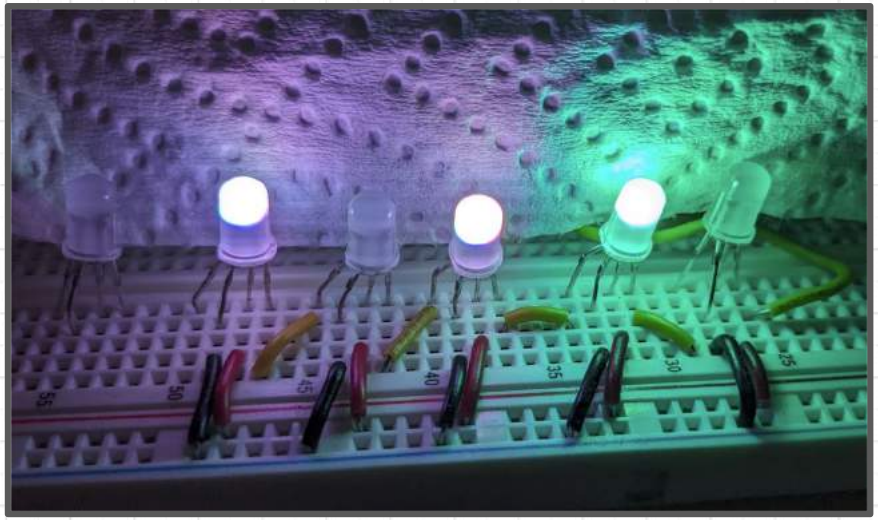
LEDs

Map Modes: Connecting

Loading action while waiting for internet connection:

- Not connected → orange pulse
- Unable to connect → red
- Connected, getting WiFi data → green pulse

Map Modes: Train Mode



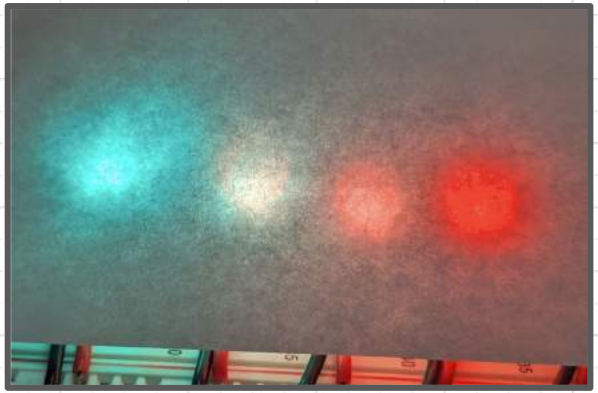
LEDs are lit up according to where trains currently are and where they are going:

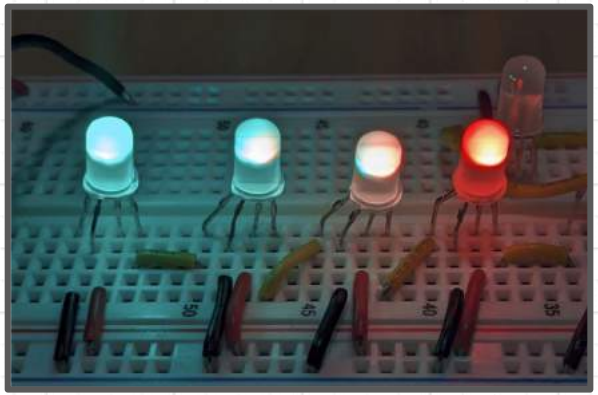
- White light → train at station
- Green light → train arriving at station
- No light → no train currently at station, or no train immediately arriving at/departing from station

Map Modes: Busyness Mode

LEDs are lit up according to traffic around station

- Green = light traffic
- Yellow = light-medium traffic
- Orange = medium-heavy traffic
- Red = severe traffic

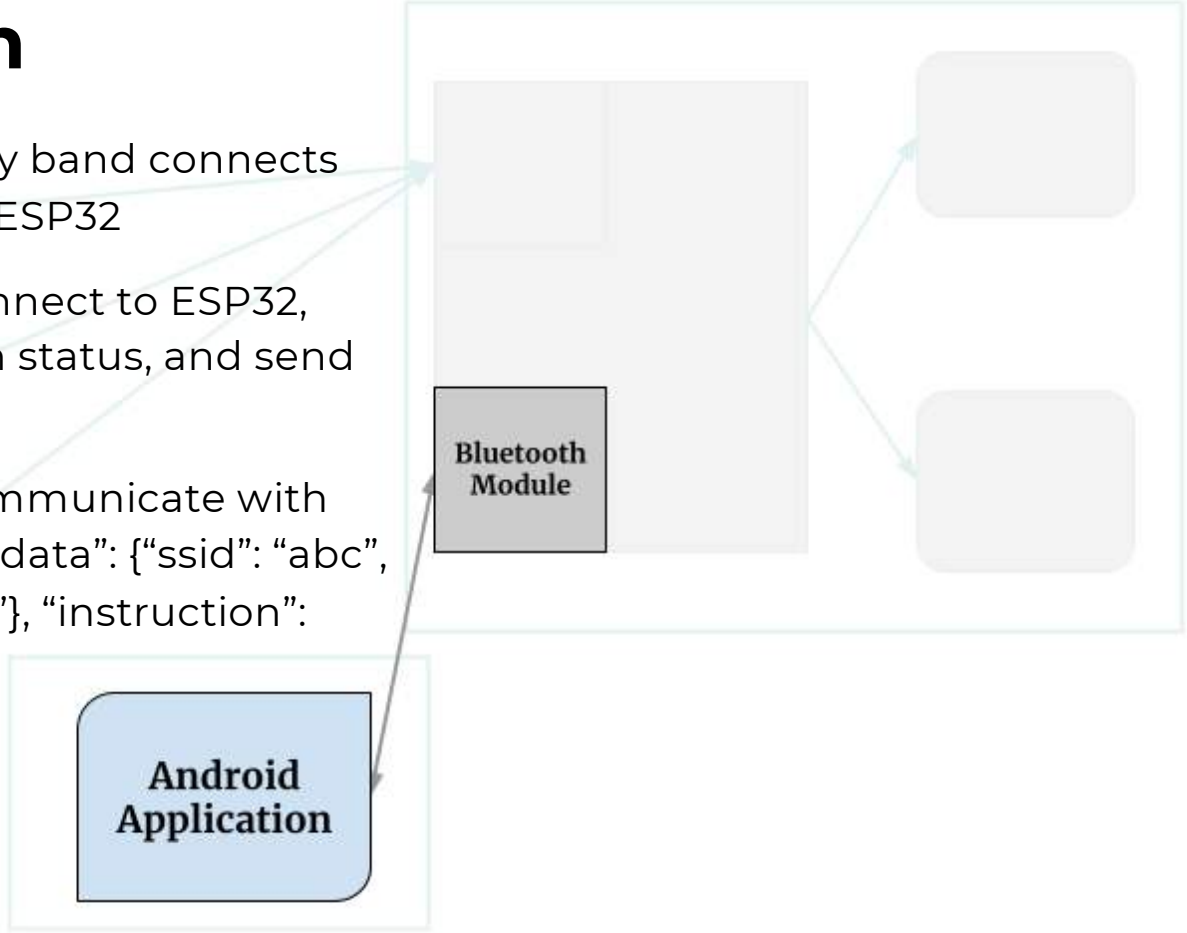




Android Application

The Application

- BLE on 2.4GHz frequency band connects Android smartphone to ESP32
- Application is able to connect to ESP32, maintain the connection status, and send commands to the board
- Uses JSON strings to communicate with and control the board ({“data”: {“ssid”: “abc”, “user”: “bob”, “pass”: “def”}, “instruction”: “WiFi”})

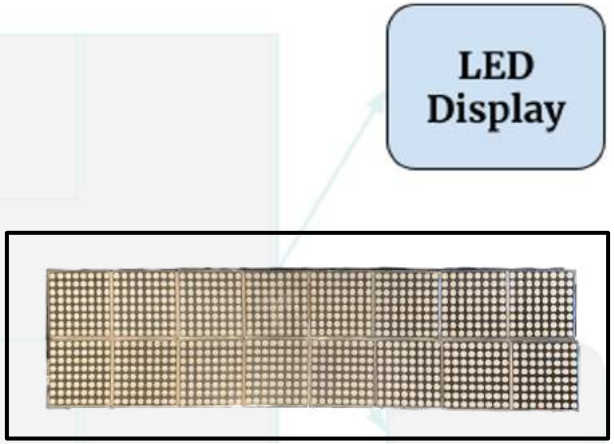


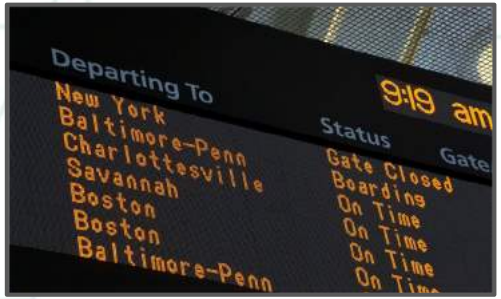
Application Demo

LED Display

MAX7219 LED Matrix Display

- LED Dot Matrices daisy-chained into a longer and wider display
- Serially interfaced via SPI
- Used to display information related to the Boston transit system:
    - Arrival/Departure times for specific trains
    - Traffic intensity at specific stations
    - Weather

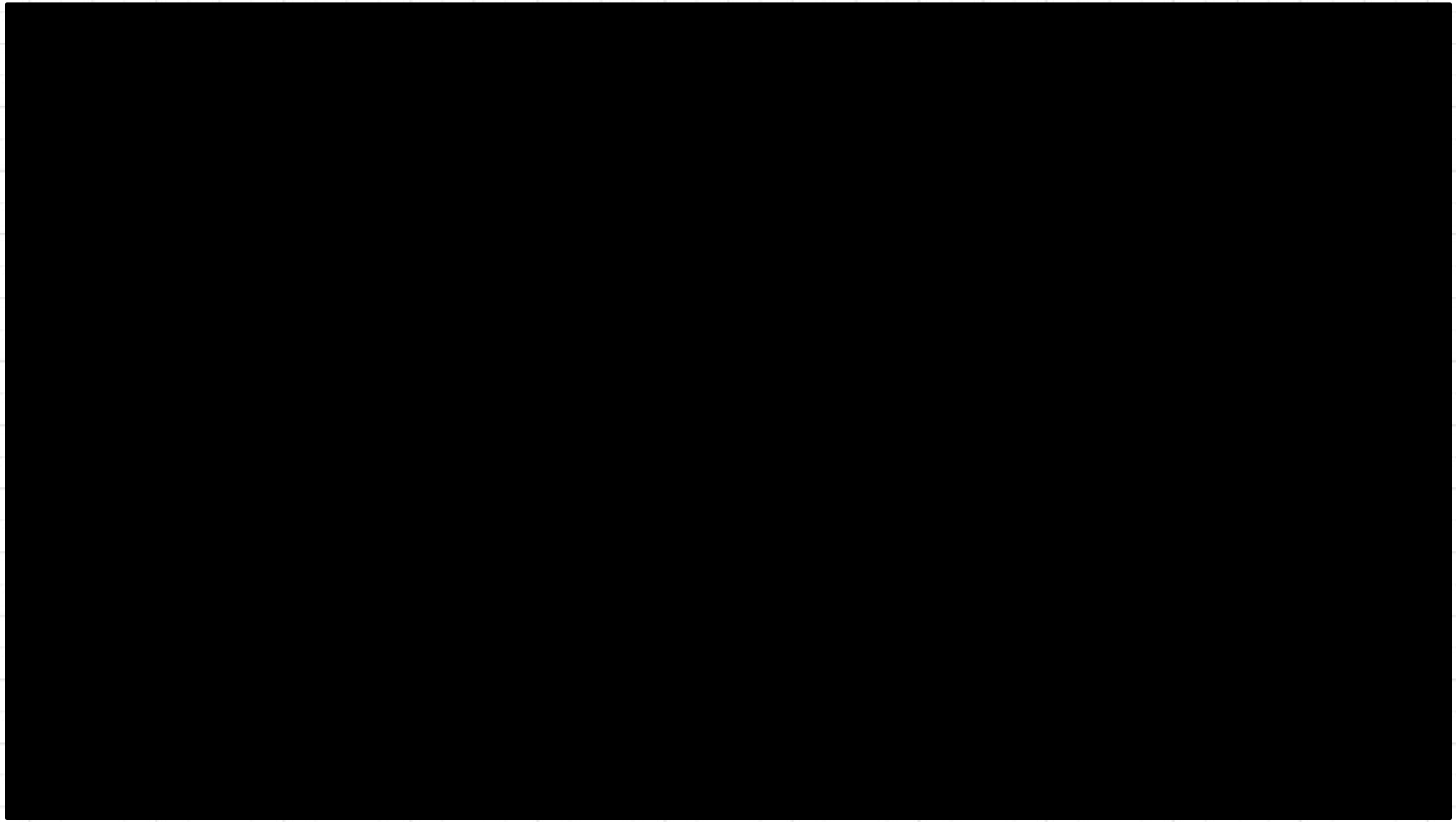




Challenges Faced

- FreeRTOS task scheduling library
- Power and data wire layout
- Board design and construction
- Custom app design and development
- API edge cases
- Incoming data management

Final Product



Acknowledgements

Dr. Yogananda Isukapalli

Dr. Haewon Jeong

Eric Hsieh

Questions?