MTD BUS TRACKING DISPLAY PROJECT PROPOSAL

By

Amber Wilt

Daniel Vlassov

Ziad Dohaim

Project Proposal for ECE 445, Senior Design, Fall 2025

10 September 2025

Project No. 4

Contents

1. Introduction

This project proposal will outline the motivation behind the MTD Bus Tracking Display Project for ECE 445 senior design. It will highlight the importance of the problem statement, how the project aims to address this problem, and the proposed design solution. Furthermore, it will address the detailed design of the final project, the requirements for a working and successful implementation, as well as any ethics or safety considerations that will need to be addressed.

1.1 Problem

Champaign has an extensive and complex bus system that is administered by the Champaign-Urbana Mass Transit District (MTD), which serves both the UIUC campus and the surrounding city. With so many buses, routes, schedules, and transfer points scattered throughout the city/campus, it can be extremely challenging for students to easily plan and navigate their daily commutes. This challenge can become significantly difficult for students who have to travel large distances between classes, dormitories, or off-campus housing in a limited timeframe between classes.

Although the MTD does provide real-time tracking information, this data is typically only displayed on electronic screens located at major bus stops. These screens show the estimated arrival times for upcoming buses, but are only accessible once a student has already reached that stop. In many cases, a student may have to travel 5-10 minutes from instructional buildings to reach a specific stop, meaning they may only discover they've missed the bus by the time they've reached the stop.

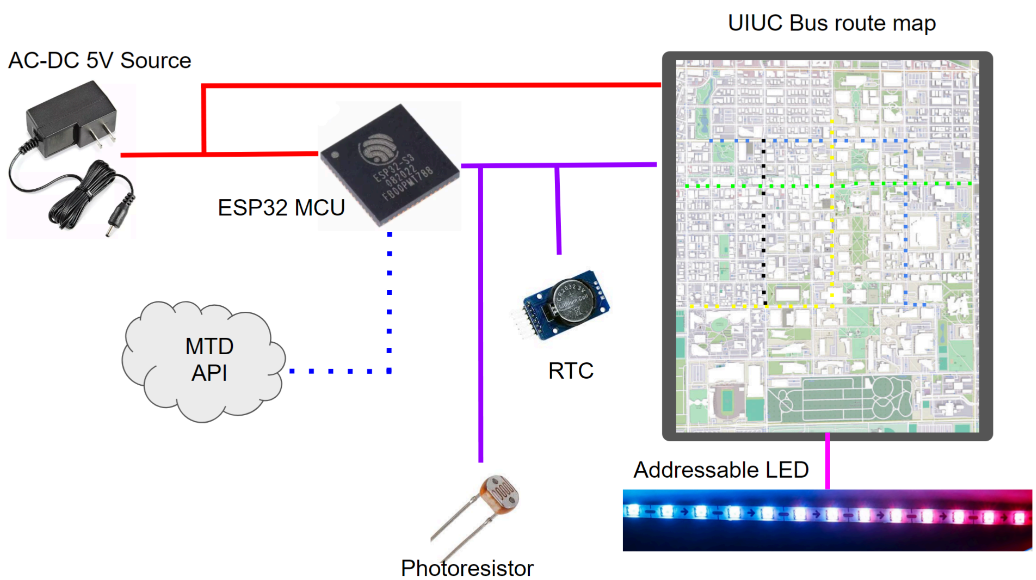
While some apps exist to assist in planning, students often run into issues that limit their reliability and effectiveness. These may include connectivity problems, device power, and hard-to-follow user interfaces that can make these apps more frustrating than helpful, sometimes causing students to miss their desired bus. For example, when opening an app to check arrival times, the display may only show the nearest stop without clearly indicating which direction the bus is heading. In other cases, the lack of route comparison tools forces students to manually sort through options.

1.2 Solution

To address the limitations of the current bus-tracking methods, we propose the design of a large 3D-printed display model that shows the real-time locations of all buses in the surrounding campus area. The model would feature a physical map of Champaign-Urbana and the campus area, with addressable RGB LEDs to represent different bus lines. Each bus would be color-coded, and its position would be updated approximately every 30 seconds via the publicly available MTD API. To further improve usability, the system would periodically light up the entire path of each bus, making it easier for students to identify which route they need to take. Additional customization features, such as adjustable themes, brightness levels, and controllable route display (user chooses to light up entire route), would allow the display to be used effectively in a variety of lighting environments.

This 3D-modeled implementation offers several advantages over simply showing bus information on a digital screen. Traditional screens often present data in lists or small maps that can feel abstract and overwhelming, requiring users to interpret bus numbers, directions, and estimated arrival times within a short period. In contrast, the 3D-printed display provides an intuitive and easy-to-follow visualization of bus movement, making it immediately clear where a bus is relative to the student's building or destination. The 3D model also makes it more engaging and accessible, particularly in busy, communal spaces where many students may take a glance without needing to navigate through an app. By combining real-time data with a physical, visual representation, this approach reduces confusion, reduces the time needed for students to plan trips, and diminishes the risk of students missing important events in their tight schedules.

1.3 Visual Aid



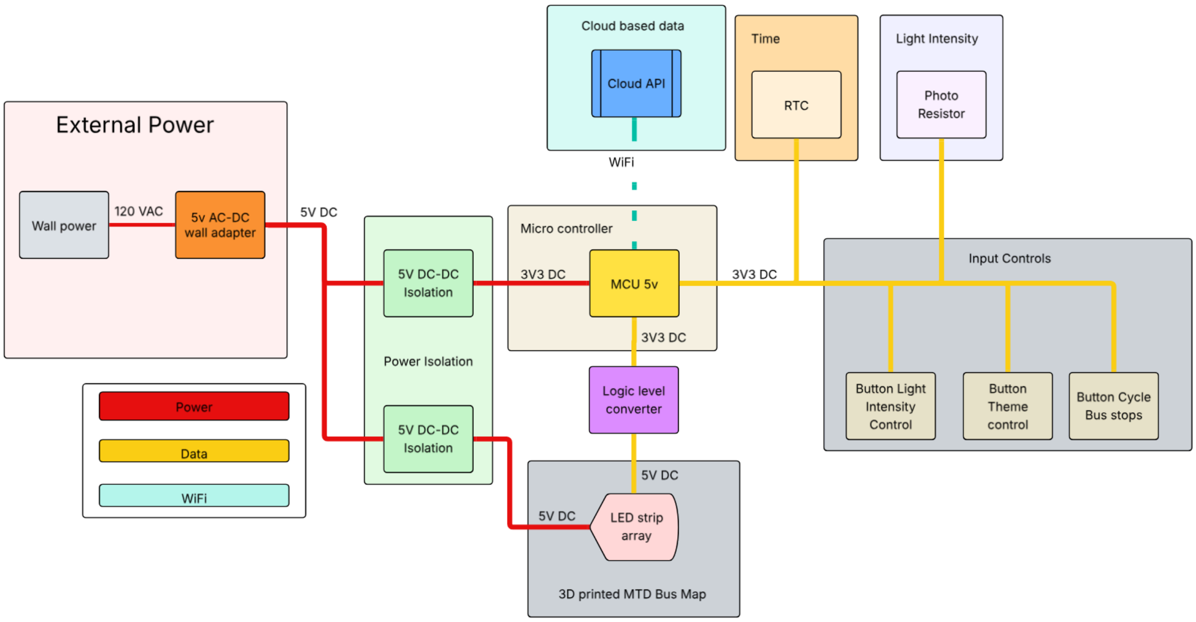
1.4 High-Level Requirements List

Three high-level requirements that can demonstrate the success of our project in solving the described problem include:

1. Real-Time Accuracy - the system must be able to display the current bus location with a positional accuracy within 1 minute compared to the actual location
2. Visibility and Accessibility - the display will need to be clearly visible and understandable within a distance of 15 feet
3. System Reliability - the system must be able to function for 7 days nonstop with no error or timing issues, with no human intervention.

2 Design

2.1 Block Diagram



2.2 Subsystem Overview

External Power:

A 120V AC wall plug will power our project. We will step down the 120V and get DC from a 5V barrel jack wall adapter. This power will then connect to our power isolation module to get clean and separate power for our microcontroller and control for the MTD LED map.

Power Isolation:

We will have a 5V input voltage, which then feeds into two DC-DC power converter isolators to separate the MCU from the power required for the LED, and in case anything shorts or fails, the MCU, wall plug, and LED will be safe

Cloud API:

The MCU connects to the Cloud API provided by the MTD Service. We will request different data for each bus every 30 seconds to obtain the longitude and latitude of each bus, and through our C code, be able to map its location to the LED Map.

MicroController:

The microcontroller we will use will control and handle our data. The MCU connects to our user input of 3 buttons, will have an RTC and a photo resistor to detect the brightness in the room, and will be able to dynamically change the brightness of the map's LEDs. The MCU will also use its built-in WiFi to connect to the MTD API.

Input controls/time/light intensity:

The buttons for control, RTC, and the photo resistor all connect to digital or analog pins on our MCU to give data from the outside world to our MCU.

MTD LED Map:

Power isolation will feed power to the LED, while the MCU itself gives the data signal to control the different indexed LEDs to control the bus routes.

2.3 Subsystem Requirements

External Power:

- Must provide **regulated 5V DC ±0.3V** with a **minimum current capacity of 2A** to power both the MCU and the LED map.
- Source: A 5V barrel jack wall adapter stepping down from 120V AC wall power.
- Output ripple voltage must be <100mV peak-to-peak to minimize electrical noise.

Power Isolation:

- Must provide isolation between MCU power and LED power using two isolated DC-DC converters, each rated for at least:
    - **5V output**
    - **1A current** for the MCU
    - **3A current** for the LED map (depending on LED density)
- Isolation voltage rating: ≥500V between the two output sides.
- Must protect against overcurrent and short circuit conditions.

Cloud API:

- MCU must poll the MTD API **every 30 seconds** to get real-time GPS data for buses.
- If Wi-Fi is disconnected, the MCU must detect failure and attempt reconnection within 5 seconds.
- Successful API response time must be <2 seconds on average for a good user experience.
- System must parse and update LED states within <2 seconds of data retrieval.

Microcontroller:

The microcontroller must contain:

- Wi-Fi capability (e.g., ESP32, ESP8266)
- At least 6 GPIOs for buttons, photoresistor, and RTC
- 1 I^2C bus for RTC
- 1 ADC input for light sensing
- Firmware should operate using a finite state machine (FSM) with stable states and recovery from faults (including Wi-Fi loss).
- Should update the LED map within 2 seconds of receiving new bus data.
- Maintain real-time clock with ±1 minute/day accuracy.

System Inputs

3 buttons:

- Must be debounced in software or hardware to prevent false triggers
- Response time: <50ms

RTC (e.g., DS3231):

- Connected over I²C, must provide accurate time within ±2 minutes per month

Photoresistor:

- Must measure ambient light in the range of 0–1023 ADC values
- The brightness of LEDs should adjust based on a simple mapping curve:
    - <300 ADC ⟶ 100% LED brightness
    - 300–700 ⟶ scale linearly
    - >700 ⟶ dim to 30% LED brightness

MTD LED Map

Must be capable of displaying a map of the UIUC campus with:

- Minimum of 100 individually addressable RGB LEDs (e.g., WS2812)
- LED spacing is maximum two inches apart on the map
- LED brightness - must be clearly visible in both ambient indoor lighting and dim environments
- Adjustable brightness from 30% to 100% based on photoresistor input
- Each bus route must be accurately mapped to specific LED indices
- MCU must be able to control LEDs at ≥30 FPS update rate to avoid flicker

2.4 Tolerance Analysis

Power Requirement Calculation for WS2812B LEDs

LED Strip Specs:

- Type: WS2812B (5V, 3-channel RGB)
- Density: 160 LEDs/m
  Voltage: 5V DC
- Max current per LED: 60mA (full white – R, G, B all at 100%)
- LED Strip Length Assumed: Let's use 1 meter (~160 LEDs)

Worst Case:

I_{max}= 160 leds X 60 mA = 9.6 A
$P_{LED\ Max}$ = 5v x 9.6A = 48W

A 50W power supply can come directly from a 5V 10A wall brick. Scalable once the exact amount of LED needed to represent bus routes is found.

ESP32 WiFi Transmission Rate and Timing

The ESP32 supports:

- 802.11 b/g/n WiFi
- Typical throughput: ~2-6 Mbps (conservative estimate: 2 Mbps for IoT use)
- JSON API responses are typically <2 kB

API Call Duration Estimate

Assuming:

- Ping + handshake + response takes <2 seconds
  Data payload per bus: ~1–2 kB JSON
- Assume 2 seconds for total fetch + parse

We will have ~ 120 requests per hour, which is well below the 1,000 requests per hour API limit.

1 request every 30s → 120 requests/hour

We can track 8 buses individually by rotating through 1 request per bus every few seconds

WS2812B LED Data Transmission Timing

Each WS2812 LED takes 24 bits of data (8-bit RGB).

Total Bits to Send:

160Led x 24 bits = 3840 bits
Each bit is ~1.25 µs long

Therefore, the total time to update the strip:
3840 x 1.25us = 4.8 mS

For 160 LEDs, our frame rate is 1 / 4.8 mS = 206 FPS.

3 Ethical Considerations

3.1 Safety Issues

Electrical Safety

The LED matrix and ESP32 microcontroller system require a power supply capable of handling significant current. Risks include overheating, short circuits, or electric shock. Compliance with Underwriters Laboratories and IEC electrical safety standards is required. Proper fuses, insulated wiring, and thermal management must be integrated properly.

Fire Hazard

High current draw in LED matrices can generate heat. Overloaded circuits could present a fire hazard. Following NFPA (National Fire Protection Association) guidelines on electronic equipment and ensuring circuit breakers and buck converters are properly rated will mitigate this.

3. 2 Ethical Issues

Accuracy and Reliability of Information

Displaying real-time bus data should be accurate. If the system displays incorrect bus positions or delays, students may miss classes. According to the IEEE Code of Ethics, engineers must hold paramount the safety and welfare of the public and avoid misleading claims. To ensure accuracy, we must test data accuracy against the official MTD API and include error handling for delayed and/or incomplete data, and display so accordingly.

Fairness and Accessibility

The ACM Code of Ethics emphasizes avoiding harm and ensuring equal access to technology. A display located only in certain buildings risks creating an accessibility divide. To address this, the design should support replication in multiple locations and adhere to ADA-compliant visibility and brightness standards, ensuring that visually impaired users are not excluded. In addition, brightness and light patterns must be managed to reduce the risk of triggering photosensitive epilepsy.

Data Privacy and Security

We will be using public API data; however, there are some constraints on its usage and how often the API should be called. Additionally, intentional misuse could occur if the system were modified to track or log users' travel habits, which could aid in stalking people. The ACM Code highlights respecting privacy and ensuring systems are not used for unjust surveillance. To avoid misuse, we will never collect or store user-specific data and will abide by the rules set by the MTD Bus API.

3.3 Regulatory and Standards Compliance

<u>IEEE and ACM Codes of Ethics:</u>
Require prioritization of public safety, accuracy, and avoidance of harm.

<u>FCC Part 15 Compliance:</u>
Since the ESP32 uses Wi-Fi, it must not interfere with other wireless devices.

<u>UL and IEC Standards:</u>
Apply to electronic component safety and power supply certification.

<u>ADA Standards:</u>

Ensure the display is accessible to individuals with disabilities (contrast, brightness, and placement).

Campus Facilities and Safety Codes:
Enforced through UIUC's DRS and building management.

3.4 Mitigation of Ethical and Safety Risks

Implement redundant checks for API accuracy and display error messages if data is unreliable.

Use brightness control and energy-efficient components to reduce environmental impact.

Follow UL/IEC standards for wiring, insulation, and overcurrent protection.

Prevent data misuse by never storing or transmitting personal location information.

Conduct a safety inspection with campus officials before installation.

References

[1] IEEE Policy 7.8. *IEEE Code of Ethics*. IEEE. https://www.ieee.org/about/corporate/governance/p7-8.

[2] ACM. *ACM Code of Ethics and Professional Conduct*. Association for Computing Machinery, 2018. https://www.acm.org/binaries/content/assets/about/acm-code-of-ethics-and-professional-conduct.pdf.

[3] Simcona. "Understanding UL Wire Standards: Key Specs & Compliance Explained." *Simcona Blog*, 2 May 2025, https://simcona.com/blog/ul-wire-specifications.

[4] Clark Testing. "IEC 61000-4-11: Electromagnetic Compatibility (EMC) – Part 4-11: Testing and Measurement Techniques – Voltage Dips, Short Interruptions and Voltage Variations Immunity Tests." *Clark Testing*, 2025

https://clarktesting.com/testing-standard/iec-61000-4-11/?msclkid=7e3ac1736d2c191acf85a9b149
1526f4.