Traffic-Management-System-for-Emergency-Vehicles

The traffic management system model aims to manage real-time traffic to provide the shortest and most-efficient path for the emergency vehicles to pass as quickly as possible.

Our model connects to the preexisting infrastructure in the Smart and planned Cities, that is LED Boards in Traffic Lights. Our code pushes the direction of incoming emergency vehicles and the direction in which they need to go to the LED Boards, so that the vehicles at traffic stop, would empty the lane for the coming of emergency vehicles. It will also track the real-time location of the emergency vehicles so that it can give the best and efficient route possible. Our model will push the directions on the LED screen two minutes ( i.e one twenty seconds prior) and will only push directions to the next two stops to avoid chaos and manage traffic efficiently.

For this we are using :

IOT devices:- ESP32(It retrieves data from webserver and pushes the data to the LCD.)

LCD(It shows the directions that our model provides to it.)

Folders

* **Original Code**

This folder Contains a copy of the Original Code that was produced during the development

1. find\_route.py: - The code finds the shortest path from the current location to the final location on a grid-like map while avoiding obstacles. It uses a breadth-first search algorithm and prints the directions taken to reach the destination.
2. format-changer.py: - This code takes a string of directions and splits them into individual directions, enclosing each direction in single quotes and separating them with commas. This will be provided to ESP32 that will further push it to the LCD.
3. microPython: - This code is using a LCD to display a sequence of predefined arrow directions with a blinking effect. It initializes the display, defines a list of arrow directions, and then shows each arrow direction on the screen for 3 seconds, followed by turning off the display for 1 second before displaying the next arrow direction. This creates a visual effect of arrows blinking on and off on the LCD screen.

* **Front-End**

This folder contains the node files, find\_route.py, format-changer.py, html, css and js files.

In this folder the python files are the updated and improvised versions of what there is in the original code.