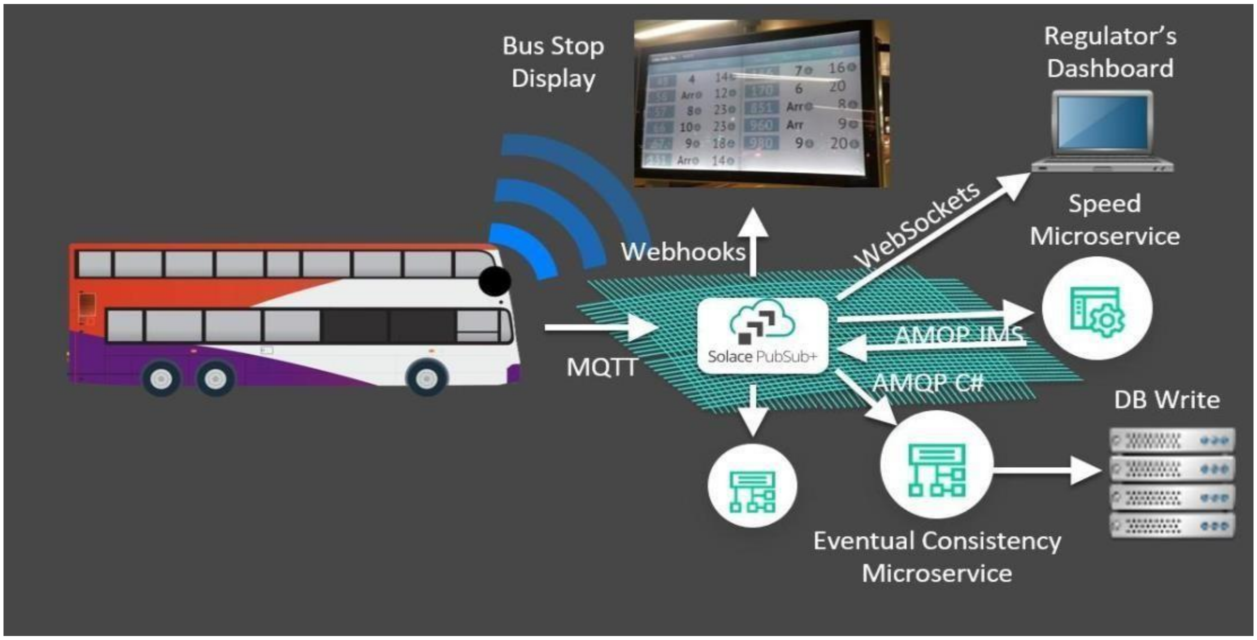
PUBLIC TRANSPORTATION OPTIMIZATION USING IOT

TEAM MEMBER

Sundharavel M - (912221106020)

Phase 4 Submission Document

PROJECT: Public Transportation Optimization

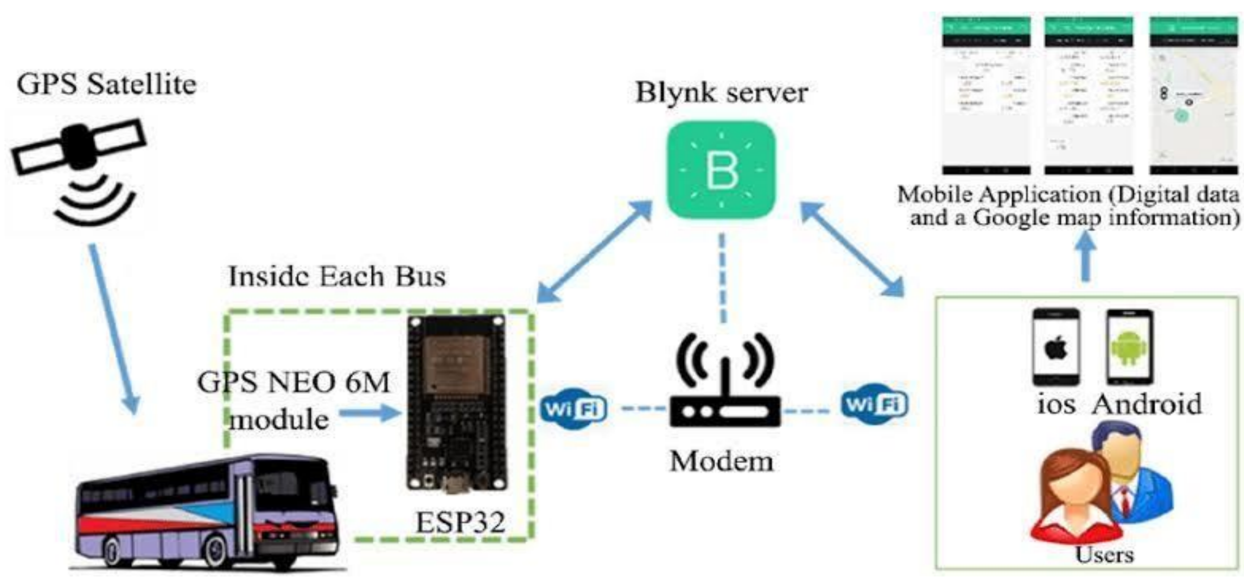


INTRODUCTION:

In modern transportation systems, the greatest challenge is to minimize, in general terms, energy consumption, and maximize economic, technological and social goals. The problem of optimizing and finding the best timetable for public transportation (PT) vehicles has been known for years [1–3]. Recent research has provided more efficient algorithms which have achieved better results by modifying known mathematical models and modifying or combining various known algorithms [4–6]. Planning of PT is a highly complex task which is usually analyzed via two different aspects: minimization of the passenger waiting time (PWT) at the station and optimization of the number and/or sizes of vehicles. The train timetabling problem (TTP) has recently been studied, and the main problem in this field is to determine a periodic or non periodic timetable, which satisfies the capacities of vehicles and limits of operations [7–10]. Some of the greatest challenges in waiting time (WT) minimization models are to find the optimal number of vehicles and to find the optimal route and minimize travel time [11].

The goal of every PT service should be to attract more people to use it by reducing the use of private cars, which is directly related to reducing traffic congestion, decreasing the number of car accidents and reducing pollution. The use of PT services by passengers depends on three elements; namely, travel time (walking, waiting and riding times), fare (ticket and other related services' costs) and convenience (a comfortable walk, waiting under a shelter, having a seat on the vehicle, air-conditioning on the vehicle, etc.). However, for the PT operator to maximize profits, operational costs need to be minimized. The PT operator's requirements can be achieved by designing an efficient network (the more transfers the network has the more efficient it is), adopting a quality timetable, efficiently schedule the vehicles and maximize the vehicle occupancy ratio (VOR). This paper proposes a model of the PT timetabling problem whose solution improves PT operations planning in terms of timetabling and vehicle scheduling; that is, changes to the departure times and assignment of the PT vehicles, so as to reduce total PWT and increase VOR (thereby minimizing operational costs of the vehicles for the PT operator). Due to the complexity of the problem, a multiobjective particle swarm optimization (MOPSO) algorithm is used in order to minimize PWT while maximizing VOR. The paper is structured as follows: Section 2 extensively describes the state-of-the-art in the timetabling problem. Section 3 elaborates on the problem statement. Section 4 presents the proposed method, and Section 5 contains two numerical examples. Section 6 presents the analysis and discussion of the results. Section 7 contains the concluding remarks.

BLOCK DIAGRAM:



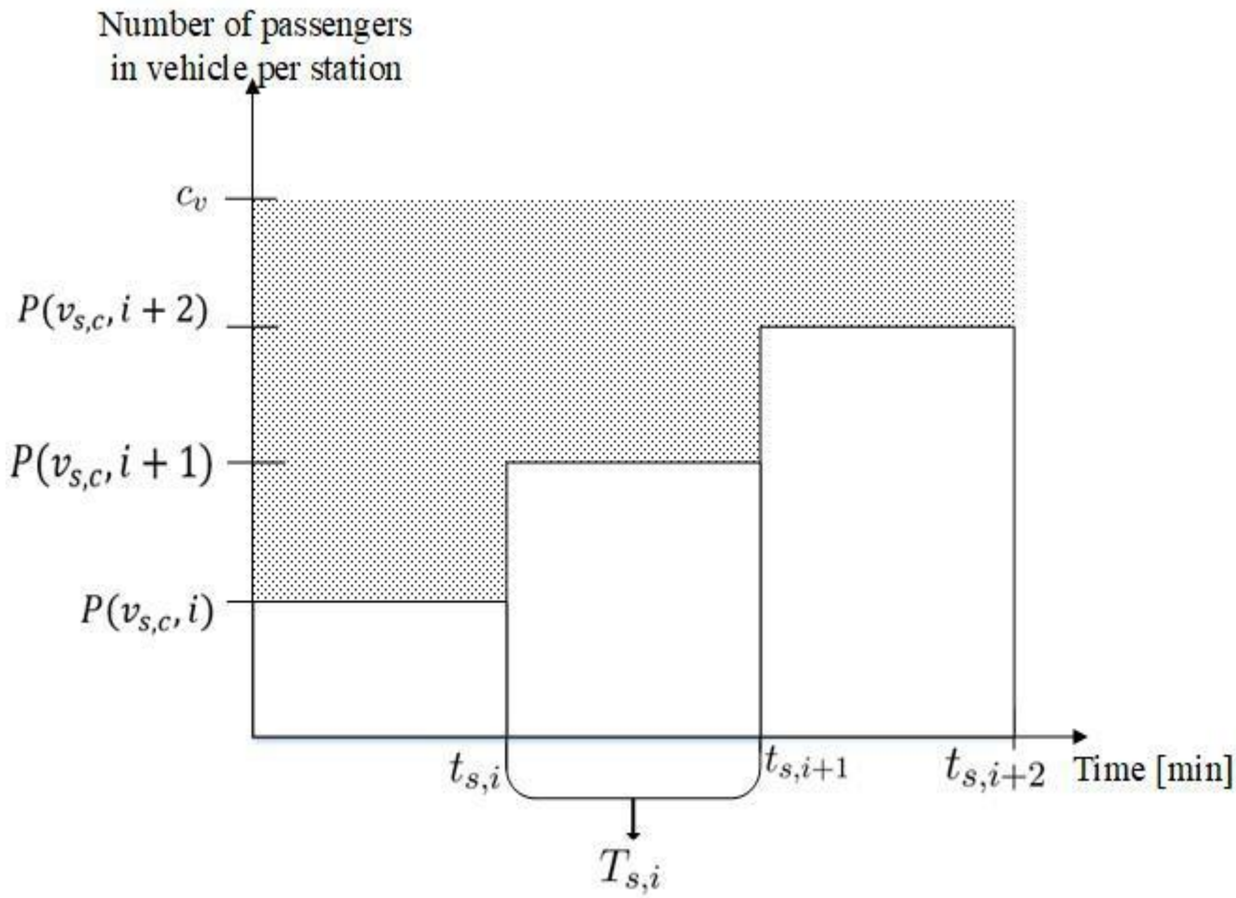


Figure: Vehicle occupancy depending on time, between consecutive stations for a given service

Public transport optimization including sensors and components:

In. Public transport optimization using Various sensors and components are used in public transport systems to enhance efficiency, safety, and passenger experience. Some of these include:

- GPS (Global Positioning System) Sensors
- Passenger Counting Sensors
- CCTV Cameras
- Ticketing and Fare Collection Systems
- Vehicle Health Monitoring Systems
- Automated Announcements and Information Displays
- Wi-Fi and Connectivity Components

- GPS (Global Positioning System) Sensors: Used for real-time tracking and monitoring of vehicles to provide accurate location information.

- Passenger Counting Sensors: Employed to measure the number of passengers boarding and alighting at different stops or stations, aiding in demand analysis and resource allocation.

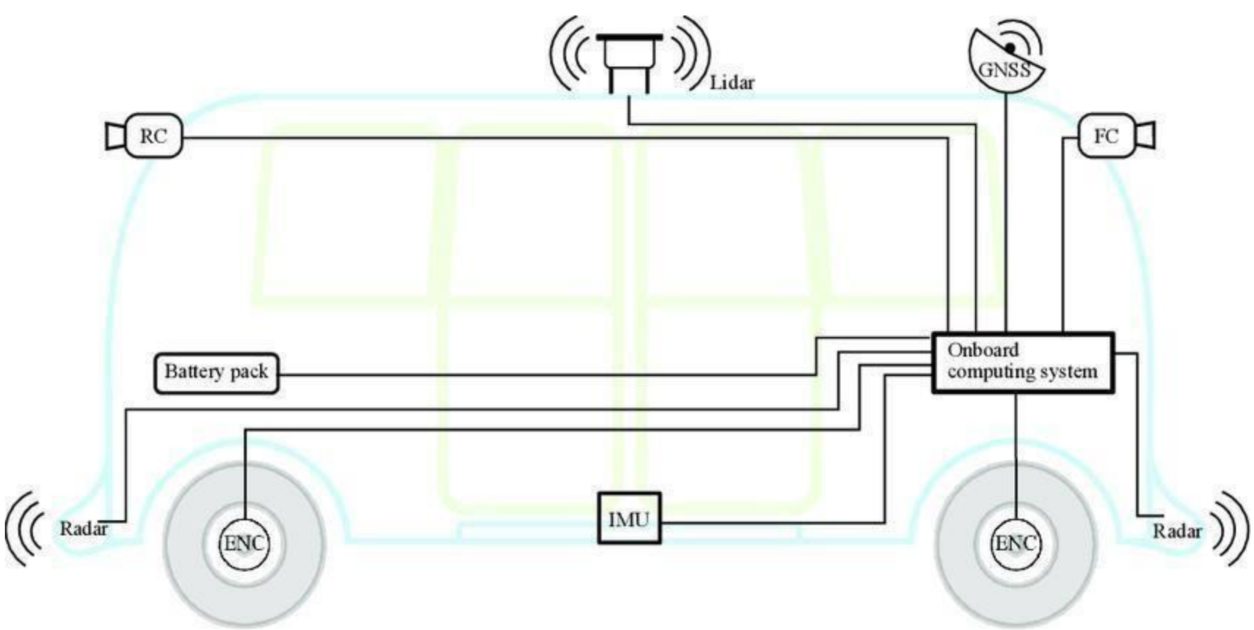
- CCTV Cameras: Installed for security purposes to ensure passenger safety and monitor any incidents that may occur on the vehicles or at the stations.

- Ticketing and Fare Collection Systems: Components such as ticket machines, contactless smart cards, or mobile payment systems are utilized for convenient and efficient fare collection.

- Vehicle Health Monitoring Systems: Sensors are employed to monitor the health and performance of the vehicles, including engine health, fuel efficiency, and maintenance requirements.

- Automated Announcements and Information Displays: Components such as audio systems and digital displays are used to provide passengers with real-time information about routes, schedules, and any important updates.

- Wi-Fi and Connectivity Components: Equipped to provide passengers with internet access and connectivity during their commute, enhancing the overall passenger experience.



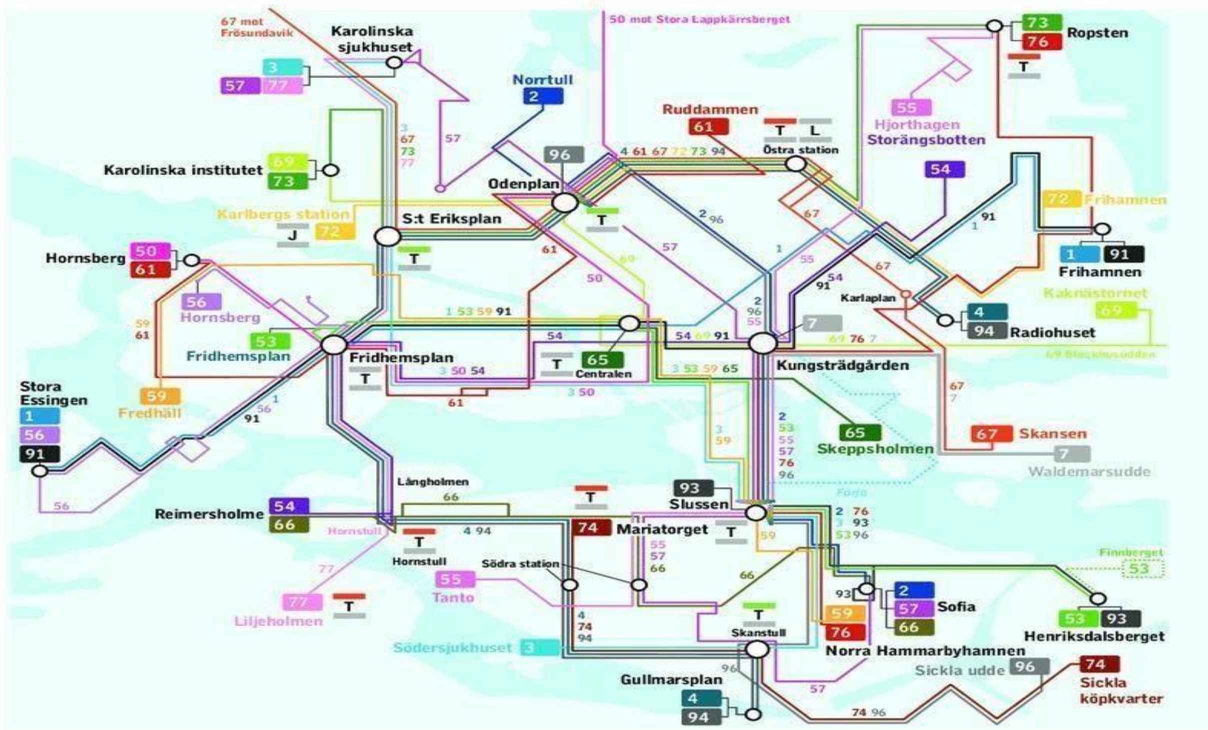
Program:

```
Import random

Import time

# Simulated IoT data for buses

Bus_data = [

{“bus_id”: 1, “location”: (37.7749, -122.4194)}, # San Francisco

{“bus_id”: 2, “location”: (34.0522, -118.2437)}, # Los Angeles

# Add more buses and their locations

]

# Simulated passenger demand

Passenger_demand = [

{“location”: (37.7749, -122.4194), “destination”: (34.0522, - 118.2437)}, # SF to LA

# Add more passenger demand routes

]

Def optimize_routes(buses, demand):

# Add your optimization logic here

# For demonstration, we’ll assign buses to the nearest passenger demand
```

```
Optimized_routes = []

For passenger in demand:

Min_distance = float("inf")

Assigned_bus = None

For bus in buses:

Distance = haversine(bus["location"], passenger["location"])

If distance < min_distance:

Min_distance = distance

Assigned_bus = bus["bus_id"]

    Optimized_routes.append({"passenger": passenger, "bus_id": assigned_bus})

Return optimized_routes

Def haversine(coord1, coord2):

# Haversine formula to calculate distance between two coordinates

    Lat1, lon1 = coord1
    Lat2, lon2 = coord2

# Calculate distance (for simplicity, this is not accurate for long distances)

 Return random.uniform(10, 200) # Simulated distance

 While True:

    Optimized_routes = optimize_routes(bus_data, passenger_demand)
```

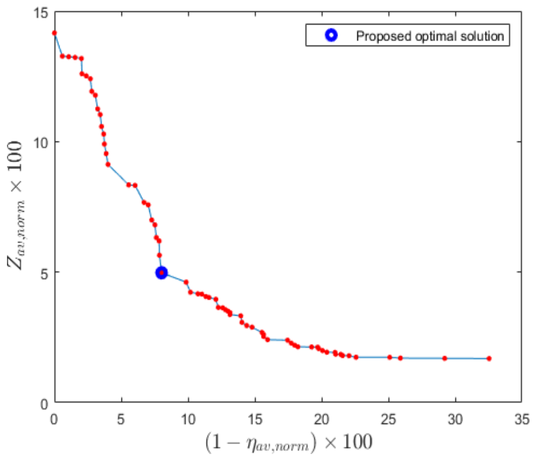
```
For route in optimized_routes:

 Print(f"Bus {route['bus_id']} assigned to passenger route from

        {route['passenger']['location']} to {route['passenger']['destination']}")

        Time.sleep(300) # Simulated time delay, e.g., 5 mminutes
```

Output:

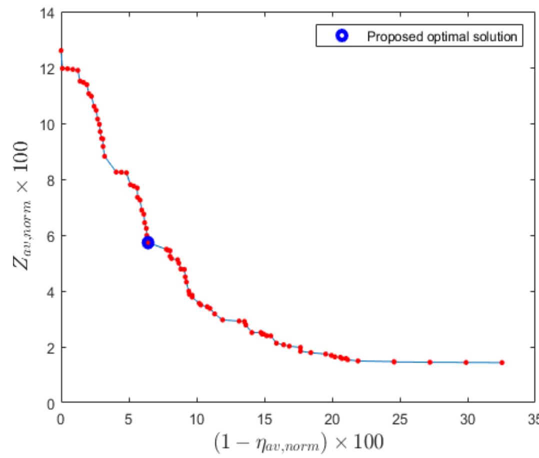


Node to Node Route Planning

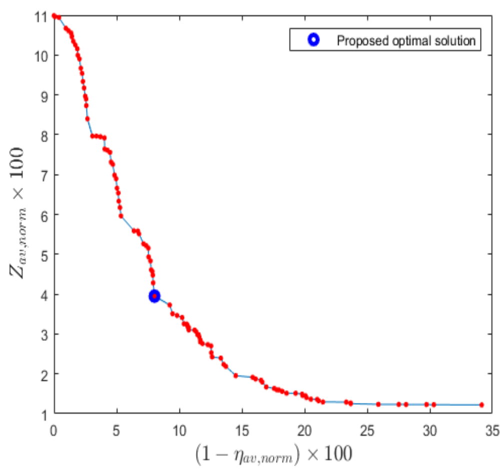
		To Node					
From Node		Entrance	Exit	Bus Stop	Fare Gate	Bus	MRT
	Entrance			Start Trip	Start Trip		
	Exit						
	Bus Stop		End Trip	**Walking**	**Walking**	**Boarding**	
	Fare Gate		End Trip	**Walking**			**Boarding**
	Bus			Alighting		**Travelling**	
	MRT				**Alighting**		**Travelling**



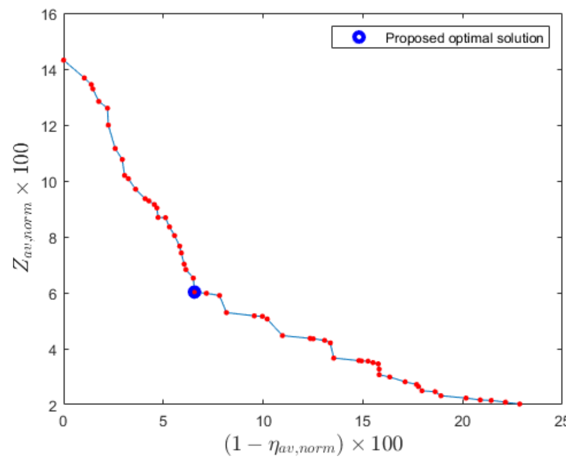
(a)



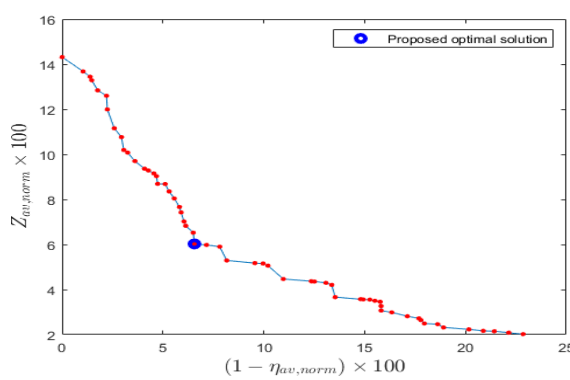
(b)



(c)



(d)



(e)

Figure: Obtained Pareto-optimal set and the proposed optimal solutions.

Conclusions:

This paper introduced a novel IoT-based bus stop that provided smart monitoring and maintenance solutions to reduce energy consumption and increase the satisfaction of commuters. The system consisted of layers corresponding to data acquisition, processing, storage, and presentation. The system monitored the bus stop's occupancy and sensors so that based on the sensor readings, the lights and air-conditioning could operate more efficiently. The air pollution in and around the bus stop vicinity was also monitored. The bus stop operation status and air pollution levels were then sent to a cloud-based server. A mobile app was developed to enable operation engineers to monitor the air conditioning and lights remotely. Utilizing Google Maps, a bus stop operation status was displayed using different colored attributes. In order to meet the system objectives, test cases were conducted to ensure that the system performance was aligned with the goals. The results were conclusive to determine that this project can be scaled to multiple bus stops. The proposed system cuts down on power consumption by controlling the bus stop's utilities based on its occupancy. By keeping constant track of the bus stop's occupancy, the system can control the lights and the air conditioning. This helps save energy, as compared to traditional bus stops where the utilities remain running even during hours of no occupancy, leading to a waste of resources. The exact amount of saved energy will be reported in future work as we plan to add a solar energy system to supply power to the bus stop. Another aspect than can be addressed in the future is the communication between the vehicle and the infrastructure (Bus Stop). Additionally, the versatility of the system can also expand its application to various settings such as schools, weather stations, hospitals, and the like.