



EV Fleet Management System with Smart Charging and Real-Time Monitoring

Vedanti Kawale¹, Sejal Khode², Vansh Patil³, Aboli Agashe⁴, Manish Funde⁵, Dr. Muneeb Ahmad⁶

⁶Under the guidance

Department Of Electrical Engineering

Priyadarshini College Of Engineering , Nagpur

ABSTRACT :

The use of electric vehicles (EVs) is increasing because they are more energy-efficient and help reduce dependence on conventional fuels. As the number of electric vehicles increases, managing a group of EVs becomes difficult, especially when it comes to monitoring battery condition, vehicle location, charging status and vehicle availability. This paper presents an IoT-based EV Fleet Management System that helps the fleet operator monitor and manage multiple electric vehicles from a single platform. The proposed system uses an ESP32 controller, voltage and current sensors, a GPS module, MQTT communication and an IoT-based application. It collects important information such as battery State of Charge (SOC), voltage, current, temperature, location and charging status, and sends it to the server through the internet for viewing by the fleet operator. The system can help in deciding which vehicle needs charging, which vehicle is available and which vehicle may require maintenance. The proposed system is designed as a low-cost prototype and can be further improved by adding smart charging, artificial intelligence and predictive maintenance.

Index Terms—Electric vehicle, fleet management, IoT, ESP32, MQTT, battery monitoring, GPS, smart charging.

I. Introduction

Electric vehicles are becoming popular because of the increasing need for cleaner and more efficient transportation. However, when several EVs are operated together, managing them manually can become difficult. A fleet operator needs to know the location of each vehicle, its battery level, charging condition and whether the vehicle is available for operation.

Unlike conventional vehicles, an EV cannot be refuelled in a few minutes. Charging takes time and depends on the battery level, charger availability and the planned duty of the vehicle. A wrong charging decision can leave a vehicle unavailable when it is needed, or can keep it idle at a charger for longer than required. For this reason, accurate and timely battery information is very important for fleet operation.

To solve this problem, we propose an IoT-based EV Fleet Management System. The main idea of this work is to collect information from different EVs and display it on one platform. The system uses sensors to measure battery parameters and GPS to track the vehicle location. An ESP32 controller processes the collected data and sends it to the server using MQTT communication.

The fleet operator can then check the information through an IoT-based mobile or web application. This makes fleet monitoring easier and provides useful information for charging and maintenance decisions. The remainder of this paper describes the related work, problem statement, objectives, system design, hardware and software implementation, charging logic, evaluation plan and future scope.

II. Related Work

The Internet of Things (IoT) connects physical devices to the internet so that their data can be collected, shared and analysed remotely [1]. In transportation, IoT is widely used for vehicle tracking, remote diagnostics and fleet monitoring, where each vehicle acts as a data-generating node.

For communication between devices and servers, lightweight messaging protocols are preferred. MQTT is a publish/subscribe protocol designed for constrained devices and unreliable networks, which makes it suitable for vehicle-mounted nodes [2]. Compared with request/response protocols, it needs less bandwidth and allows many vehicles to publish data to a common broker.

Battery State of Charge (SOC) is one of the most important parameters of an EV. Common estimation approaches include open-circuit-voltage lookup and coulomb counting; enhanced coulomb counting methods improve accuracy by accounting for efficiency and capacity changes [3]. Low-cost

microcontrollers such as the ESP32, which has built-in Wi-Fi, are widely used to build such monitoring nodes [4]. The present work combines these elements into a single low-cost fleet-level platform.

III. Problem Statement

Managing multiple electric vehicles manually can take a lot of time and may not provide real-time information. The fleet operator may not know the exact battery level or location of every vehicle at a particular time. The main problems are:

- Difficulty in monitoring multiple EVs at the same time.
- Lack of real-time battery information.
- Difficulty in tracking vehicle location.
- Improper or delayed charging decisions.
- Difficulty in identifying vehicles that need maintenance.
- Manual recording of fleet information.

Therefore, an automated IoT-based system is proposed to monitor the important parameters of an EV fleet.

IV. Objectives

The main objectives of this work are:

- 1) To develop an IoT-based EV fleet monitoring system.
- 2) To monitor the battery SOC, voltage and current.
- 3) To track the location of EVs using GPS.
- 4) To transmit vehicle data using MQTT.
- 5) To display vehicle information through an IoT application.
- 6) To help the operator plan vehicle charging.
- 7) To monitor the condition and availability of different EVs.
- 8) To provide a base for future predictive maintenance.

V. Proposed System

The proposed system is designed to monitor multiple electric vehicles from a central platform. Sensors installed in the vehicle collect battery-related information such as voltage, current and temperature. The GPS module provides the current location of the vehicle.

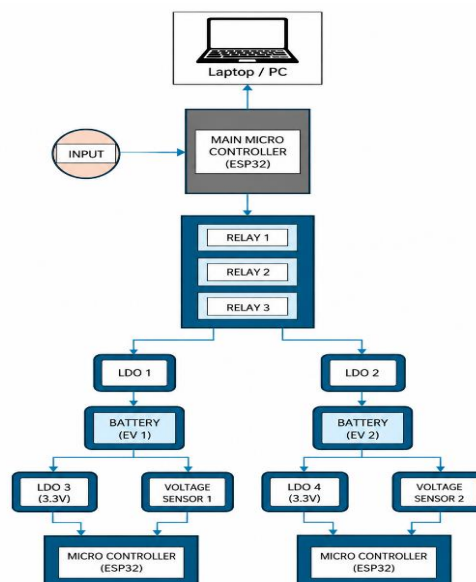
The ESP32 acts as the main controller. It receives the sensor information, processes it and sends the data to the server using Wi-Fi and MQTT communication. The collected information is stored on the cloud or database and can be viewed through the IoT application.

For example, if the battery SOC of one vehicle falls below a set threshold, the system marks that vehicle as needing charging and informs the operator, while vehicles with sufficient charge remain available for assignment. In the same way, an abnormal temperature or current reading can be used to flag a vehicle for maintenance inspection.

VI. System Architecture

The system follows a layered IoT architecture, shown in Fig. 1. The sensing layer inside each vehicle consists of the voltage, current and temperature sensors and the GPS module. The processing layer is the ESP32, which reads the sensors, converts raw values into meaningful quantities and forms a data packet. The network layer carries the packet over Wi-Fi and the internet to an MQTT broker. The application layer stores the data and presents it to the fleet operator.

Fig. 1. Block diagram of the proposed IoT-based EV fleet management system.



Because every vehicle has its own node and publishes to a common broker, more vehicles can be added without changing the server-side design. Each node is identified by a unique vehicle ID.

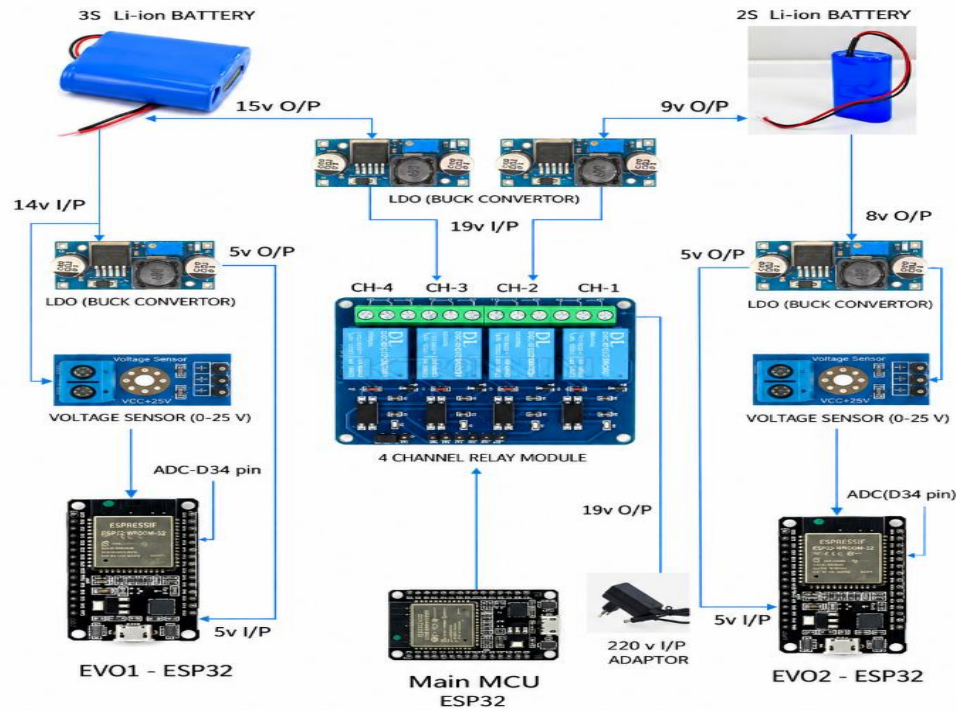


Fig. 1. Circuit diagram of the proposed IoT-based EV fleet management system.

VII. Hardware Implementation

The hardware is selected to keep the prototype low-cost and easy to reproduce. The main components and their functions are listed in Table I.

TABLE I
MAIN HARDWARE COMPONENTS

Component	Function
ESP32	Main controller; reads sensors, processes data and connects to Wi-Fi
Voltage sensor	Measures battery pack voltage
Current sensor	Measures charging and discharging current
Temperature sensor	Measures battery temperature
GPS module	Provides vehicle latitude and longitude

The ESP32 reads the analog outputs of the voltage and current sensors through its analog-to-digital converter, the temperature sensor through its digital or analog input, and the GPS module through a serial (UART) interface. The vehicle data is sampled periodically and sent as one combined packet to reduce network traffic.

VIII. Communication and Software

Data is exchanged using MQTT. Each vehicle publishes its readings to a topic that contains its vehicle ID, and the application subscribes to these topics to receive live updates. The proposed topic structure is given in Table II.

TABLE II
Proposed MQTT Topic Structure

Topic	Data carried
fleet/<id>/battery	SOC, voltage, current
fleet/<id>/temp	Battery temperature
fleet/<id>/location	Latitude, longitude
fleet/<id>/status	Charging / available / maintenance

The payload is sent in a simple JSON format, which is easy to generate on the ESP32 and easy to parse in the application. The broker forwards the messages to the database, where each reading is stored with a timestamp so that the history of a vehicle can be reviewed later. Reconnection logic is added on the ESP32 so that the node automatically reconnects to Wi-Fi and the broker if the connection is lost.

IX. Battery Monitoring and SOC Estimation

The ESP32 calculates the battery power from the measured voltage and current as

$$P = V \times I \quad (1)$$

The SOC of the battery can be estimated by coulomb counting, in which the charge flowing into or out of the battery is integrated over time:

$$SOC(t) = SOC(t_0) - (1/C_n) \int \eta \cdot I(\tau) d\tau \quad (2)$$

where C_n is the rated battery capacity, I is the battery current (positive during discharge), and η is the coulombic efficiency. As coulomb counting drifts over time, the estimate can be corrected using the battery voltage when the vehicle is at rest [3]. The estimated SOC, together with voltage, current and temperature, is sent to the server with every update.

X. Smart Charging and Availability Logic

The server applies simple rules to every incoming reading to decide the status of each vehicle, as shown in Fig. 2. The thresholds are configurable by the operator; the values in Table III are example settings.

TABLE III
EXAMPLE VEHICLE STATUS RULES

Condition	Vehicle status
$SOC \geq 60\%$	Available
$30\% \leq SOC < 60\%$	Available; plan charging
$SOC < 30\%$	Needs charging
Temperature or current abnormal	Maintenance check

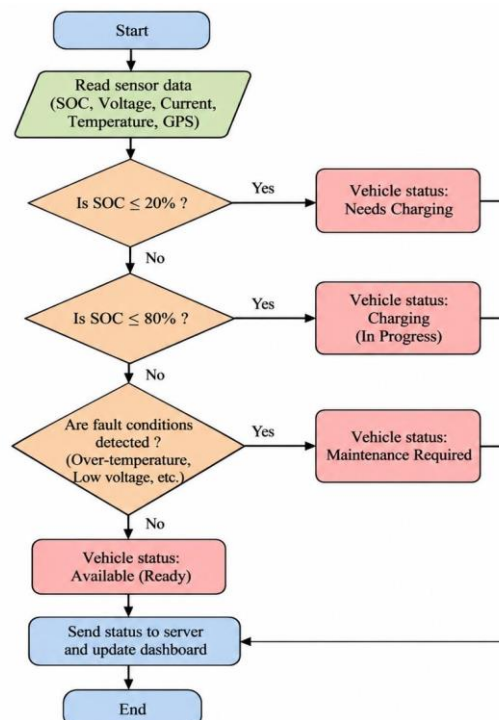


Fig. 2. Flowchart of the vehicle status decision.

Using these statuses, the operator can quickly decide which vehicle to send for charging, which can be assigned to a trip and which should be inspected. This is the first step toward fully smart charging, where charging could later be scheduled automatically according to vehicle demand, charger availability and electricity tariff.

XI. Application Dashboard

The IoT application gives the fleet operator a single view of the whole fleet. The main features are:

- Live map showing the location of every vehicle.
- Battery SOC, voltage, current and temperature of each vehicle.
- Colour-coded status: available, charging needed or maintenance.
- Alerts for low battery and abnormal temperature or current.
- Stored history of each vehicle for later analysis.

XII. Testing and Performance Evaluation

The prototype will be evaluated on the accuracy of its measurements and the reliability of its data transfer. The parameters considered are the data update interval, the delay between publishing and displaying a reading, the GPS position error and the SOC estimation error. Table IV is provided for recording the measured values.

XIII. Advantages and Limitations

The main advantages of the system are its low cost, its use of a lightweight protocol, its ability to add more vehicles easily and the single-platform view it gives the operator. Its limitations are that it depends on Wi-Fi or internet coverage, the GPS accuracy can reduce in covered areas, and the SOC estimate from a simple sensor set is approximate compared with that from a full battery management system. These points can be improved in later versions.

XIV. Future Scope

The system can be extended in several ways. A cellular (4G/5G) module can replace Wi-Fi so that vehicles stay connected while on the road. Smart charging can be added to schedule charging automatically based on vehicle demand, charger availability and electricity price. Artificial intelligence can be used to predict battery degradation and the remaining driving range, and predictive maintenance can use the stored history to warn about faults before they occur. Integration with the vehicle's own battery management system and with charging stations is also possible.

XV. Conclusion

This paper presented an IoT-based EV fleet management system that allows an operator to monitor battery SOC, voltage, current, temperature, location and charging status of multiple vehicles from a single platform. By using an ESP32, simple sensors, GPS and MQTT, the system provides real-time information that helps in charging, availability and maintenance decisions. The low-cost prototype also provides a base for future work, including smart charging, artificial intelligence and predictive maintenance.

REFERENCES

- [1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015.
- [2] A. Banks and R. Gupta, Eds., "MQTT Version 3.1.1," OASIS Standard, Oct. 2014.
- [3] K. S. Ng, C.-S. Moo, Y.-P. Chen, and Y.-C. Hsieh, "Enhanced coulomb counting method for estimating state-of-charge and state-of-health of lithium-ion batteries," *Applied Energy*, vol. 86, no. 9, pp. 1506–1511, Sep. 2009.
- [4] Espressif Systems, "ESP32 Series Datasheet," Espressif Systems, Shanghai, China.
- [5] B. Devi, V. Suresh Kumar, T. Karthick, and C. Balasundar, "Deep learning based IoT and cloud-integrated state of charge estimation for battery powered electric vehicles," *Journal of Energy Storage*, vol. 100, Art. no. 113622, 2024. doi: 10.1016/j.est.2024.113622.
- [6] O. P. D. Obuli et al., "Enhanced SOC estimation of lithium ion batteries with RealTime data using machine learning algorithms," *Scientific Reports*, vol. 14, Art. no. 16036, 2024. doi: 10.1038/s41598-024-66997-9.
- [7] "Battery Monitoring and Control System for EV using IOT Technology," *IEEE Conference Publication*, 2024. doi: 10.1109/TEECCON64024.2024.10939820.
- [8] "Series Connected Cell Monitoring System of Li-ion Battery with I2C and Cloud Interfacing," in *Proceedings of the IEEE 50th Annual Conference of the Industrial Electronics Society (IECON)*, 2024. doi: 10.1109/IECON55916.2024.10905926.
- [9] "A Review of Electric Vehicle Smart Charging and Its Impact on Power Grid," in *Proceedings of IEEE SoutheastCon 2025*, 2025. doi: 10.1109/SoutheastCon56624.2025.10971723.

-
- [10] S. Ghanbari Motlagh, J. Oladigbolu, and L. Li, "A review on electric vehicle charging station operation considering market dynamics and grid interaction," *Applied Energy*, vol. 392, Art. no. 126058, 2025. doi: 10.1016/j.apenergy.2025.126058.