WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal Homepage: www.ijrpr.com ISSN 2582-7421

EV Charging Station Finder and Slot Booking: A Role-Based Web Platform Using Java, JSP and MySQL

Mrs. Y. Sudha Madhuri[1], Noonela Tarun Kumar[2]

[1]Assistant Professor, [2]M. Tech Final Semester,
Department of Computer Science and Engineering, Sanketika Vidya Parishad Engineering College, Visakhapatnam, Andhra Pradesh, India

DOI : https://doi.org/10.55248/gengpi.07.1026.34024

A B S T R A C T

The rapid adoption of electric vehicles (EVs) has increased the need for charging infrastructure that is easy to locate, transparent in availability and simple to reserve. In practice, station information, slot allocation, payments and customer feedback are often spread across independent operator platforms or handled manually, which makes charging sessions harder to plan. This paper presents the design and implementation of an EV Charging Station Finder and Slot Booking system, a web-based application that brings EV users, charging station operators and an administrator onto a single platform. Users can register, search for stations by city or by latitude and longitude, and compare FAST and NORMAL charging services with their capacity and price, reserve a slot for a chosen date and time, pay, download an invoice and rate the station. Operators register and manage their stations and bookings, and each new station is verified by the administrator before it is listed. The system is built with Java Servlets, JSP, HTML, CSS and JavaScript, uses MySQL through JDBC for storage, and checks remaining slot capacity before a booking is confirmed. Functional, integration, database, validation and role-based access testing confirmed that the main workflows operate as intended. The platform provides a structured, low-cost foundation that can be extended with interactive maps, real-time charger status, online payment gateways and demand prediction.

Keywords: Electric vehicle; Charging station finder; Slot booking; Java Servlets; JSP; MySQL; Web application

1. Introduction

The steady growth of electric vehicle (EV) use for personal and commercial transport has made reliable and easily accessible charging infrastructure a practical necessity. Unlike conventional refueling, an EV charging session can take a considerable time, so the availability of a suitable charging point at the intended time directly affects how conveniently a journey can be planned. A well-organized charging management system should therefore let drivers find a nearby station, understand what it offers, and secure a slot in advance.

In current practice, however, the information needed for this decision is fragmented. Drivers may rely on maps, individual operator websites or applications, or direct contact with station staff, while operators keep station details, bookings, payments and customer feedback in separate systems or manual records. There is rarely a single environment in which the EV user, the station operator and the administrator who supervises the network all work on the same data.

This paper presents an EV Charging Station Finder and Slot Booking system developed as a web application to close that gap. The main contributions are as follows:

A centralized, role-based platform that serves EV users, charging station operators and an administrator through a common database.

Location-aware station discovery based on the city, latitude and longitude stored for every station.

Capacity-aware slot reservation for FAST and NORMAL charging services, with remaining capacity verified before a booking is confirmed.

An integrated booking lifecycle covering payment status, invoice generation and post-charging ratings and comments.

An approval workflow in which the administrator verifies newly registered stations before they become visible to users.

The remainder of the paper is organized as follows. Section 2 reviews related work. Section 3 analyses the existing approach and introduces the proposed system. Section 4 describes the architecture and data design, Section 5 the implementation, and Section 6 the testing and the resulting application. Section 7 concludes the paper and outlines future work.

2. Related work

Research on EV charging management has addressed monitoring, scheduling, booking and payment, usually one aspect at a time. Kumar, Gupta, and Sharma (2025) proposed a charging station management system that combines Internet of Things (IoT) and cloud computing so that station data can be collected and monitored from a central environment. Related IoT-oriented work on charging and monitoring (Al-Ali, Hassan, & Abdelsalam, 2025) and on smart charging (Patel & Shah, 2024) shows how connected stations can feed software-based management platforms.

A second group of studies concentrates on allocation and scheduling. Chen, Li, and Wang (2025) described a cloud-based approach to real-time station allocation and scheduling, in which current system information is used to assign vehicles to suitable charging facilities. Lee, Kim, and Park (2024) examined slot management using real-time data analytics, highlighting the value of data-driven decisions when several users compete for limited charging capacity.

Closer to the present work are studies on finding and booking. Karthik, Rajesh, and Prasad (2025) designed a charging slot booking system delivered through a mobile application, and Ahmed, Rahman, and Hossain (2024) reported a mobile application for locating charging stations and booking them. Reddy, Suresh, and Kumar (2024) proposed a cloud-based finder and reservation system that combines station discovery with reservation in one platform, which is the nearest counterpart to the system described here. Payment has also been considered in integrated charging solutions (Singh, Verma, & Tiwari, 2025; Mehta & Kulkarni, 2024).

The surveyed studies emphasize IoT and cloud connectivity, real-time allocation or mobile access. The present work instead concentrates on a complete, browser-based reservation workflow built from widely available open technologies: station search, slot booking against remaining capacity, payment status, invoicing and feedback, together with operator management and administrator approval. This makes it suitable as a practical base that can later be connected to the IoT, analytics and mobile approaches discussed above.

3. Problem analysis and proposed system

3.1 Existing approach and its limitations

At present an EV user typically finds a station through local knowledge, online maps, operator websites or dedicated applications, then contacts the station or uses that provider's own application to ask about availability. Reservations, where offered, may be taken by telephone, by staff or through separate booking platforms. Operators manage station details, capacity, pricing and bookings with their own records, and payments, invoices and customer reviews are handled elsewhere. The main limitations are:

No single source of station information, so users must search several applications or websites.

Inconsistent location information, which makes it difficult to identify nearby stations quickly.

Limited, unstructured slot reservation and little visibility of remaining capacity for a given charging type and time.

Manual booking management, which makes it hard to track pending, confirmed, completed and cancelled bookings.

Payment, invoice and feedback records kept apart from the booking they relate to.

No unified administrative control or consolidated reporting across users, stations, bookings, payments and feedback.

3.2 Proposed system

The proposed system is a web application in which three kinds of users work on one shared database. EV users register and log in, search for stations by city or by coordinates, view station details, and book a slot by selecting the station, the charging type (FAST or NORMAL), the date and the time. Operators register stations, configure capacity and price for each charging type, and manage incoming bookings. The administrator reviews each registration and approves or rejects it before the station appears in user searches, and also manages users, bookings, payments and feedback. Table 1 summarizes how the proposed system differs from the existing approach.

Table 1 - Existing approach compared with the proposed system.

Aspect	Existing approach	Proposed system
Station information	Scattered across operator applications, websites and maps	Centralized database with address, city, coordinates, charging type, capacity and price
Station search	Depends on individual platforms; location data is inconsistent	Search by city or latitude and longitude; nearby stations listed
Slot reservation	Telephone, visit or separate booking platforms	Online booking by station, charging type, date and time

Aspect	Existing approach	Proposed system
Availability	Limited visibility of remaining capacity	Remaining capacity checked before confirmation
Payment and invoice	Handled separately from the reservation	Payment status linked to the booking; invoice generated
Feedback	Spread over external review platforms	Ratings and comments stored for each station
Administration	Separate systems for each operator	Single administrator module for users, stations, bookings, payments and feedback

4. System architecture and design

4.1 Architecture

The application follows a three-layer architecture, shown in Fig. 1. The web interface layer consists of JSP pages styled with HTML, CSS and JavaScript and is accessed through a standard web browser by all three user roles. The application layer is written in Java: Servlets receive requests, perform validation and business logic, and call service components for booking, payment, invoice and report handling, while JDBC provides database connectivity. The database layer is a MySQL database that stores user details, station information, charging types and capacity, bookings and slot usage, payments and invoices, feedback and ratings, and station approval information.

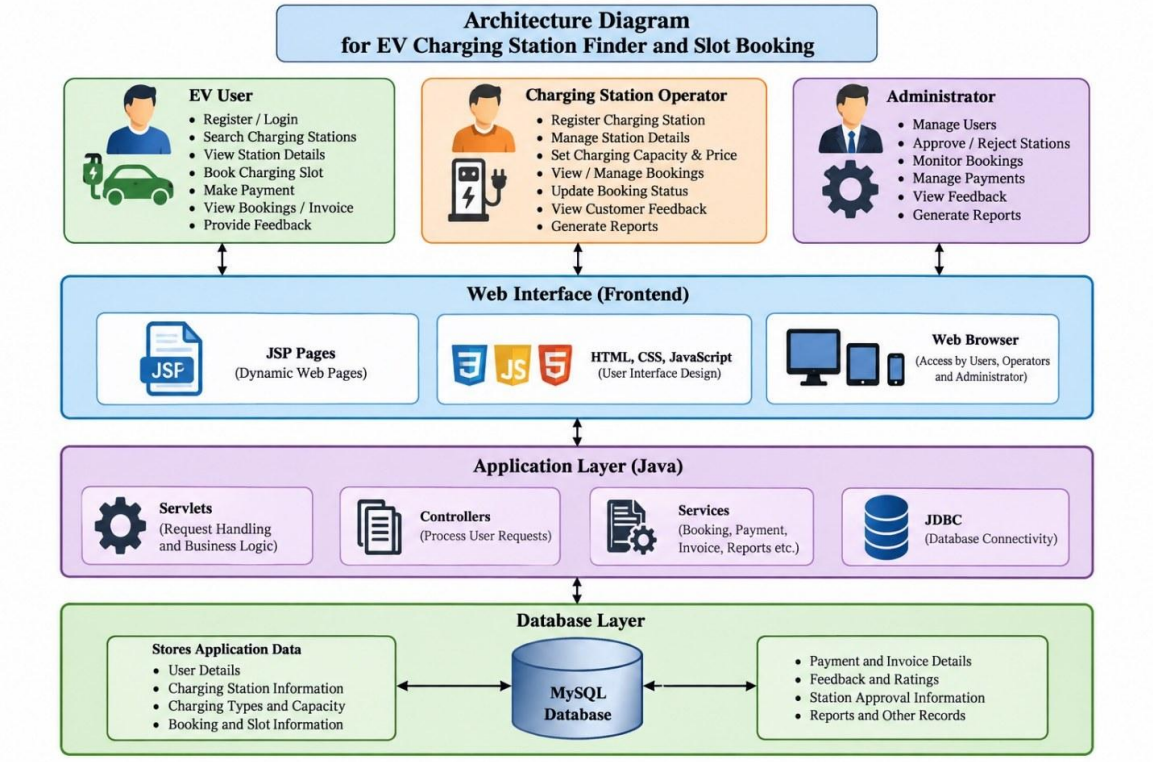


Fig. 1 - Three-layer architecture of the EV Charging Station Finder and Slot Booking system.

4.2 User roles

Access is organized by role. An **EV user** can register and log in, search stations, view station details, book a slot, make a payment, view bookings and invoices, and submit feedback. A **charging station operator** can register a station, maintain its details, set charging capacity and price, view and update bookings, read customer feedback and view reports. The **administrator** manages users, approves or rejects station registrations, and monitors bookings, payments, feedback and reports. Role-specific dashboards ensure that each user reaches only the functions assigned to that role.

4.3 Data design

The database holds the following groups of information. User records contain registration and login details together with the role. Station records contain the station name, owner, contact details, address, city, latitude, longitude and approval status; a newly registered station is kept as pending until the administrator acts on it. Charging service records describe the FAST and NORMAL services of a station with their slot capacity and cost. A slot-

usage record counts the slots already reserved for a station, charging type and time. Booking records link a user to a station and store the charging type, scheduled slot time, booking status and payment status. Payment, invoice and feedback records are linked to the booking, user and station to which they belong. Because every record references the entities it depends on, a booking can be traced to its user, station, payment, invoice and rating.

4.4 Input and output design

Input screens are designed to limit the amount of data typed, to reduce errors and to keep the process simple. Forms for registration, login, station registration, booking, payment and feedback validate required fields and invalid values and return clear messages when an entry cannot be accepted. Outputs include station lists with details, slot availability, booking history with status, a downloadable invoice, feedback listings, dashboards and reports for operators and the administrator.

5. Implementation

5.1 Development environment

The system was developed and tested on a standard desktop configuration, listed in Table 2. The application is written in Java, uses JSP for dynamic pages, Servlets for request handling and JDBC for communication with MySQL, and is deployed on Apache Tomcat.

Table 2 - Hardware and software environment.

Component	Specification
Processor	Pentium i3
Memory and storage	8 GB RAM; 20 GB hard disk
Display and input	15-inch LED monitor; keyboard and mouse
Operating system	Windows 10/11
Programming language	Java (JDK 23.0.1)
Front end	JSP, HTML, CSS, JavaScript
IDE	Apache NetBeans 24
Application server	Apache Tomcat 9.0.84
Database	MySQL (MySQL Workbench 8.0)

5.2 Modules

The application is divided into five modules.

User management and station finder. This module handles registration, login and user activities. After authentication, users view available stations with their location, city, charging types, capacity and cost. The latitude and longitude stored for each station support location-based identification of nearby stations.

Charging station management. Operators enter station details, owner and contact information, address, city and coordinates, and configure FAST and NORMAL services with their capacities and costs. Operators can add, edit and manage slot information. New stations remain pending until the administrator reviews them.

Slot booking and reservation. Users choose a station, a charging type, a date and a time. The system checks availability, records the booking with its status, and maintains slot-usage information by station, charging type and time. Operators can update booking status and complete or reject applicable bookings.

Payment, invoice and feedback. Payment is recorded against the booking together with its amount and status. Invoices for charging transactions can be generated and downloaded, and after a session users can submit a rating and comment that is stored with the user, station and booking.

Administrator and reporting. The administrator views registered users, monitors stations, approves or rejects registrations, and accesses booking and feedback information. Reporting functions let the administrator and operators review station and booking records.

5.3 Slot availability and booking workflow

Slot control is the core of the booking logic. For a station s, charging type t and time slot τ, the remaining capacity is

$$A(s, t, \tau) = C(s, t) - U(s, t, \tau) \qquad (1)$$

where $C(s, t)$ is the configured capacity of type t at station s and $U(s, t, \tau)$ is the number of slots already used in that time slot. A request is accepted only when $A(s, t, \tau) > 0$, after which the slot-usage record is updated. The complete workflow is: (i) the user logs in and searches for a station; (ii) the user selects a station, charging type, date and time; (iii) the system evaluates Eq. (1) and rejects the request if no capacity remains; (iv) an accepted booking is stored with a pending status; (v) the operator updates the booking status; and (vi) the user pays, downloads the invoice and submits feedback. Bookings pass through pending, booked, completed and cancelled states, so that users and operators can track a reservation throughout the charging process.

6. Testing, results and discussion

6.1 Testing approach

The complete application was tested as an integrated system in a web environment consisting of the Java application, JSP interface, MySQL database and a modern browser (Google Chrome, Microsoft Edge or Mozilla Firefox). The database contained sample records for users, stations, charging types, capacity, bookings, payments and feedback, and each of the three roles was tested separately. The testing covered five areas: functional testing of every major function; integration testing of module interactions (for example, station search to booking, and booking to payment, invoice and feedback); database testing of insert, retrieve, update and delete operations; validation testing with empty, incomplete and invalid input; and security and access testing of authentication, role restrictions and session handling after logout.

A suite of 20 test cases was defined; representative cases are listed in Table 3. System testing confirmed that data passes correctly between modules, that unavailable capacity does not produce a booking, that payment and invoice details correspond to the correct booking, and that restricted pages require authentication.

Table 3 - Representative system test cases.

ID	Module	Test scenario	Expected result
TC01	User management	Register with valid user details	User account is created successfully
TC03	User management	Log in with invalid credentials	Login is rejected and a message is displayed
TC04	Station finder	Search using a valid city	Matching charging stations are displayed
TC06	Station management	Register a new charging station	Registration is stored with pending status
TC08	Station management	Administrator approves a station	Station becomes available to users
TC09	Slot booking	Select an available charging slot	Booking request is processed
TC10	Slot booking	Book when capacity is unavailable	Booking is not processed
TC12	Payment	Submit payment for a booking	Payment information is recorded
TC13	Invoice	Generate invoice for a transaction	Invoice contains booking and payment details
TC14	Feedback	Submit rating and comment	Feedback is stored successfully
TC19	Access control	Open a restricted function without authorization	Access is restricted

6.2 Application walkthrough

Fig. 2 shows the user's search page. A user can search by city or by latitude and longitude, and the page lists nearby charging hubs with their location, a map link, a booking button and the station rating. Selecting a station opens the slot page in Fig. 3, where the user picks a date and sees the FAST and NORMAL services side by side with their hourly cost and with the remaining capacity shown under each time slot; choosing a slot displays the selected slot and the estimated total before the booking is confirmed. The new booking appears in the user's booking history with its charging type, scheduled slot, status and payment status, from where payment, invoice download and feedback are available.

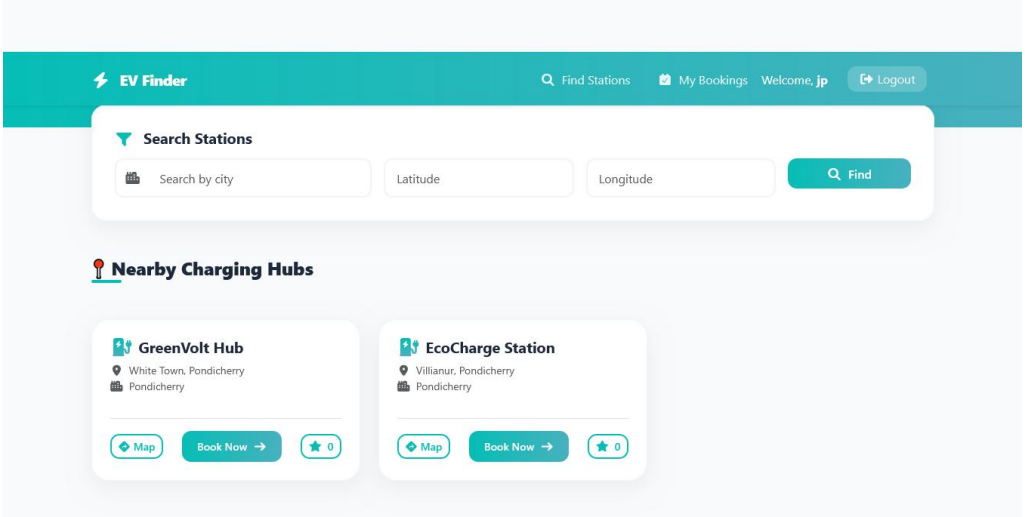


Fig. 2 - Station search page with nearby charging hubs.

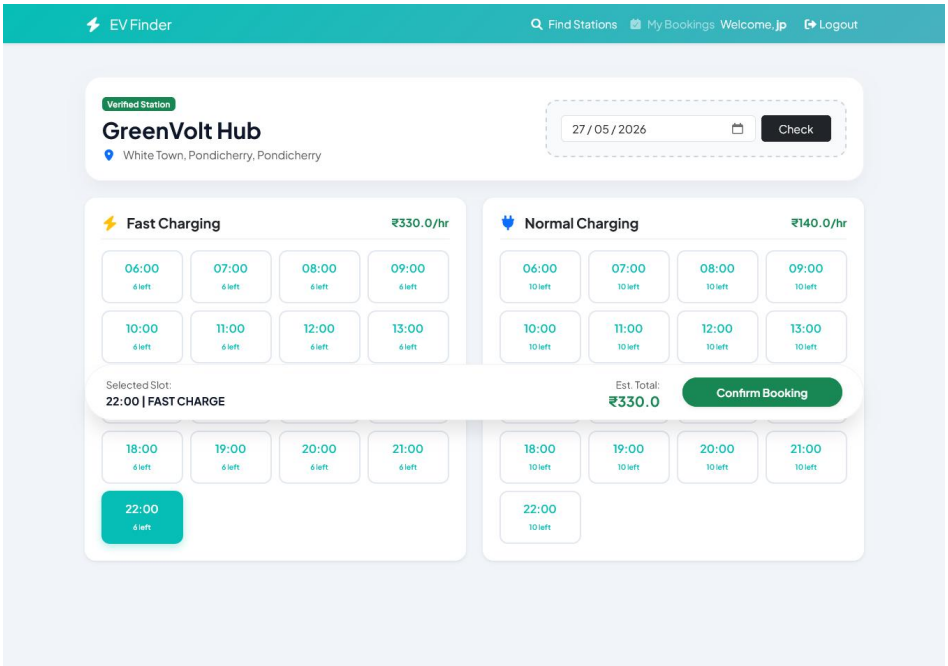


Fig. 3 - Slot selection for FAST and NORMAL charging with remaining capacity.

On the administrator side, the dashboard in Fig. 4 summarizes the number of users, stations, pending approvals and bookings, together with a growth overview and a station status chart that separates approved and pending stations. Further administrator screens list registered users, charging stations awaiting approval, bookings, customer feedback and revenue and booking reports, while operators have their own dashboard for services and incoming bookings.

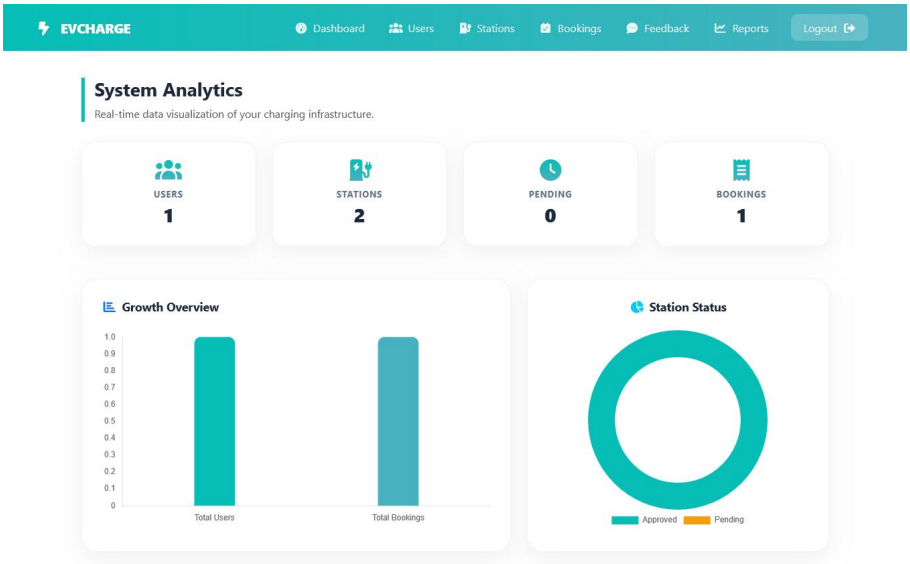


Fig. 4 - Administrator analytics dashboard.

6.3 Discussion

The proposed system is feasible on three counts. Economically, it uses freely available technologies (Java, JSP, Tomcat and MySQL) and ordinary desktop hardware, so no specialized equipment is needed for development or testing. Technically, the Servlet–JSP–JDBC stack is mature and well suited to the structured data of users, stations, bookings and payments. Socially, a browser-based workflow with role-specific screens is easy to learn and does not require installing separate software.

Several limitations remain. Availability is derived from recorded bookings and configured capacity rather than from live charger status. Payment is recorded as an amount and a status and is not yet connected to an online payment gateway. Map support is limited to station location rather than interactive route guidance, and security measures such as password encryption and stronger session management are listed as future enhancements. These limitations define the extension path described next.

7. Conclusion and future work

This paper presented a web-based EV Charging Station Finder and Slot Booking system that integrates station discovery, location-based search, FAST and NORMAL charging selection, capacity-checked slot booking, payment status, invoice generation, feedback and administrative control in a single Java application backed by MySQL. By connecting EV users, operators and the administrator through one workflow, the system supports the whole path from selecting a station to completing a charging transaction. Testing across functional, integration, database, validation and access scenarios confirmed that the modules work together as intended. Planned extensions include:

Android and iOS applications with push notifications for bookings, payments and reminders.

Real-time charger status, with IoT-enabled stations reporting availability, energy use and connector status.

Integration with online maps for interactive display, route navigation and travel distance.

An online payment gateway supporting UPI, cards, net banking and wallets, with automated confirmation.

Smart slot recommendation, dynamic pricing and charging duration and cost estimation based on vehicle profile and battery capacity.

Advanced analytics, machine-learning-based demand prediction and EV route and charging planning.

Stronger security, including password encryption, role-based access control and protection against common web vulnerabilities.

Scalable, cloud-based deployment for larger networks of stations across multiple cities.

Acknowledgements

The author thanks the faculty and project guide for their guidance and support, , for the project environment in which this system was developed.

References

[1] Ahmed, S., Rahman, M., & Hossain, T. (2024). Mobile application for electric vehicle charging station locator and booking. *Journal of Smart*

Transportation Systems, 8(2), 45–58.

[2] Al-Ali, A. R., Hassan, M. H., & Abdelsalam, M. (2025). IoT-based electric vehicle charging and monitoring system. In *Proceedings of the International Conference on Smart Grid Technologies* (pp. 85–90).

[3] Chen, J., Li, Y., & Wang, X. (2025). Real-time EV charging station allocation and scheduling using cloud-based systems. *IEEE Access, 13*, 11234–11248.

[4] Karthik, M., Rajesh, S., & Prasad, V. (2025). Design and implementation of EV charging slot booking system using mobile applications. *International Journal of Engineering Research & Technology, 14*(3), 210–215.

[5] Kumar, S., Gupta, R., & Sharma, P. (2025). Smart EV charging station management system using IoT and cloud computing. In *Proceedings of the IEEE International Conference on Smart Energy Systems* (pp. 120–125).

[6] Lee, K., Kim, H., & Park, J. (2024). Efficient EV charging slot management using real-time data analytics. *IEEE Transactions on Intelligent Transportation Systems, 25*(4), 3000–3010.

[7] Mehta, V., & Kulkarni, S. (2024). Integrated EV charging and payment system using cloud technology. *International Journal of Advanced Research in Engineering and Technology, 15*(1), 10–18.

[8] Patel, R., & Shah, N. (2024). IoT-enabled smart charging system for electric vehicles. In *Proceedings of the IEEE International Conference on Green Energy and Smart Systems* (pp. 150–155).

[9] Reddy, P. N., Suresh, K., & Kumar, L. (2024). Cloud-based EV charging station finder and reservation system. *International Journal of Computer Applications, 185*(12), 25–30.

[10] Singh, D., Verma, A., & Tiwari, R. (2025). Smart EV charging infrastructure with payment integration using IoT. In *Proceedings of the International Conference on Sustainable Energy Systems* (pp. 60–65).

[11] Project source: Noonela Tarun Kumar, "EV Charging Station Finder and Slot Booking: A Role-Based Web Platform Using Java, JSP and MySQL " M.Tech Project Report, Sanketika Vidya Parishad Engineering College, 2024–2026.