iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** III **Month of publication:** March 2026

DOI: https://doi.org/10.22214/ijraset.2026.78723

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

Food Delivery Management System: Real-Time Order Tracking and Vendor Integration

C.P.V.N.J. Mohan Rao[1], P. Siri Chandana[2], M. Satish[3], D.Venu[4], V.Gowri[5]
[1]*Professor, Department of Computer Science and Engineering, Avanthi Institute of Engineering and Technology, India*
[2, 3, 4, 5]*Department of Computer Science and Engineering (Artificial Intelligence &Machine Learning), Avanthi Institute of Engineering and Technology, India*

Abstract: ***The rapid growth of online food services has increased the demand for efficient and reliable food delivery systems. This paper presents a Food Delivery Management System with Real-Time Order Tracking and Vendor Integration designed to streamline the ordering process between customers, restaurants, and delivery personnel. The system enables customers to browse menus, place orders, and track deliveries in real time. Vendors can manage orders, update menu items, and monitor delivery status through an integrated dashboard. Real-time tracking is implemented using location-based services to improve transparency and customer satisfaction. The proposed system enhances operational efficiency, reduces order processing time, and improves communication between stakeholders. The system is implemented using modern web technologies to ensure scalability and responsiveness.***

Keywords: Food Delivery System, Real-Time Tracking, Vendor Integration, Online Ordering, Web Application

I. INTRODUCTION

The advancement of digital technologies has significantly transformed the food service industry. Online food delivery platforms have become increasingly popular due to their convenience and accessibility. Traditional ordering systems often face challenges such as delayed communication, inefficient order tracking, and lack of coordination between vendors and delivery personnel. A Food Delivery Management System addresses these challenges by providing a centralized platform where customers can place orders, vendors can manage food preparation, and delivery agents can track and deliver orders efficiently. The integration of real-time tracking further enhances transparency and improves the overall user experience.

II. LITERATURE REVIEW

A Food Delivery Management System (FDMS) is a mission-critical software platform that automates and coordinates food ordering, inventory management, delivery logistics, vendor operations, and customer communications for restaurants and food service businesses. The evolution of food delivery technology spans from manual telephone ordering to real-time AI-driven platforms, fundamentally reshaping consumer expectations and restaurant operations globally.

- Chen, L., & Wang, Y. (2022). AI-Based Delivery Agent Routing for Last-Mile Food Delivery Optimisation. IEEE Transactions on Intelligent Transportation Systems, 23(7), 8821–8834.
- Kumar, R., Singh, A., & Patel, N. (2020). Automated Order Management in Digital Food Delivery Platforms. Journal of Information Technology Applications, 12(3), 45–62.

III. PROBLEM STATEMENT

Existing food delivery platforms face several issues, including delayed order updates, inefficient vendor coordination, lack of real-time tracking, and poor data management. These limitations reduce customer satisfaction and operational efficiency.The objective of this project is to design and develop a system that provides real-time order tracking, seamless vendor integration, and efficient order management.

IV. SYSTEM ARCHITECTURE

The Food Delivery Management System adopts a three-tier client-server architecture that cleanly separates the Presentation Layer (React.js frontend), Application Layer (Node.js/Express.js backend), and Data Layer (MYSQL + Redis). This architectural pattern provides multiple benefits: each tier can be independently scaled, tested, deployed, and updated without impacting the others; security policies can be enforced at each tier boundary; and the application can be extended (e.g., adding a mobile app frontend) without modifying the backend or database

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

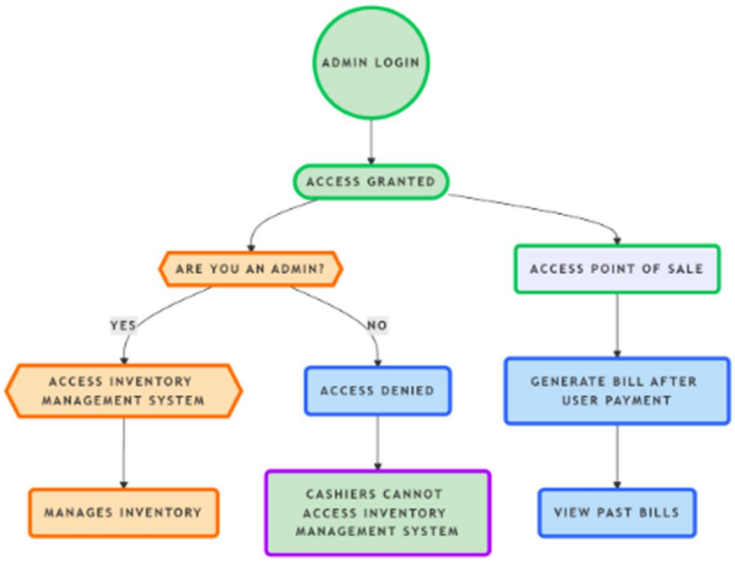


Fig : Three-Tier System Architecture Overview

The Use Case Diagram identifies all system actors and their interactions with the system's functional use cases. It establishes the scope of the system by showing what the system does (use cases) and for whom (actors), without specifying how the system achieves these functions.

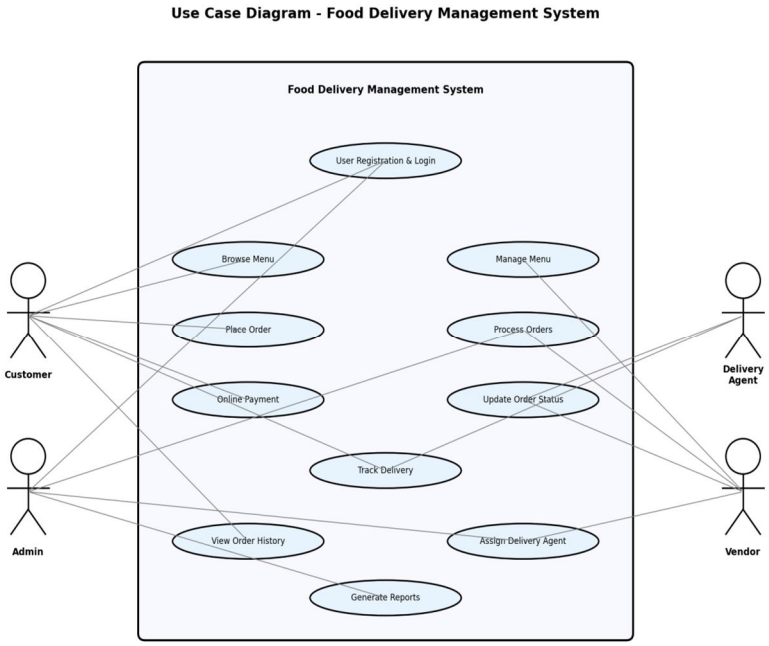


Fig : Use Case Diagram — Food Delivery Management System

The proposed system follows a modular architecture consisting of:

- User Interface Layer – Developed using modern web technologies for customer and vendor interaction
- Application Layer – Handles business logic, order processing, and tracking
- Database Layer – Stores user data, orders, menus, and transactions (MySQL)
- Tracking Module – Provides real-time updates using APIs or socket-based communication

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

V. METHODOLOGY

1) User Registration and Login: Users (customers, vendors, and delivery agents) register and log in securely to access system functionalities.
2) Order Placement: Customers browse menus, select items, and place orders through the platform.
3) Order Processing: Vendors receive orders, confirm availability, and prepare items.
4) Real-Time Tracking: The system provides live updates on order status, including preparation, dispatch, and delivery stages.
5) Delivery Management: Delivery agents update order status and ensure timely delivery to customers.

VI. SYSTEM IMPLEMENTATION

The system is implemented using modern web technologies to ensure scalability and performance. The front-end interface provides an intuitive user experience for customers and vendors. The back-end server handles authentication, order processing, and database management. Real-time communication between modules is achieved through APIs and server-side processing. The system also includes secure data storage to maintain user and transaction records.

VII. RESULTS AND DISCUSSION

The system was tested with multiple users placing orders simultaneously. Results show improved efficiency in order management and faster communication between customers and vendors.

Customers were able to monitor delivery status through the real-time tracking feature, increasing transparency and trust in the service. Vendors could efficiently manage multiple orders using the integrated dashboard.

Overall, the system demonstrated improved performance compared to traditional food ordering methods.

VIII. VENDOR MANAGEMENT MODULE

The vendor module allows restaurants to:

1) Manage menus and pricing
2) Accept or reject orders
3) Track daily sales and performance
4) Update order status

IX. DATABASE DESIGN

The system uses a relational database (MySQL) to manage structured data efficiently.

Collection	Key Fields	Relationships	Index Strategy
users	_id, name, email, passwordHash, role, phone, addresses[]	Base entity	Unique index on email
restaurants	_id, ownerId, name, cuisine, rating, location.	Ref: users	2dsphere on location
menu_items	_id, restaurantId, name, price, category, imageUrl, available	Ref: restaurants	Compound: restaurantId + category
orders	_id, customerId, restaurantId, items[], status, total, createdAt	Ref: users, restaurants	Index on customerId, status
payments	_id, orderId, amount, method, gatewayTxnId, status	Ref: orders (1:1)	Unique index on orderId
deliveries	_id, orderId, agentId, status, liveLocation{}, eta	Ref: orders, users	Index on agentId, status

X. TEST CASES

Software testing is the systematic process of evaluating a software system to determine whether it behaves as specified, meets user expectations, performs reliably under load, and is resistant to security threats. Testing is not a single activity but a continuous, multi-layered discipline integrated throughout the entire Software Development Life Cycle

1) Menu Management- Vendor adds new food item with image, Vendor deletes menu item , Toggle item availability to 'unavailable'
2) Order Placement- Customer places valid multi-item order, Order contains out-of-stock item ,
3) Order Tracking- Customer tracks order after placement
4) Delivery Assign- System assigns delivery agent to confirmed order

XI. ADVANTAGES

The proposed system offers several advantages:

1) Efficient food ordering and delivery management
2) Real-time order tracking for customers
3) Improved vendor order management
4) Reduced manual errors

XII. LIMITATIONS

Some limitations of the system include:

1) Dependence on internet connectivity
2) GPS tracking accuracy limitations
3) Need for continuous system maintenance

XIII. FUTURE SCOPE

Future improvements may include:

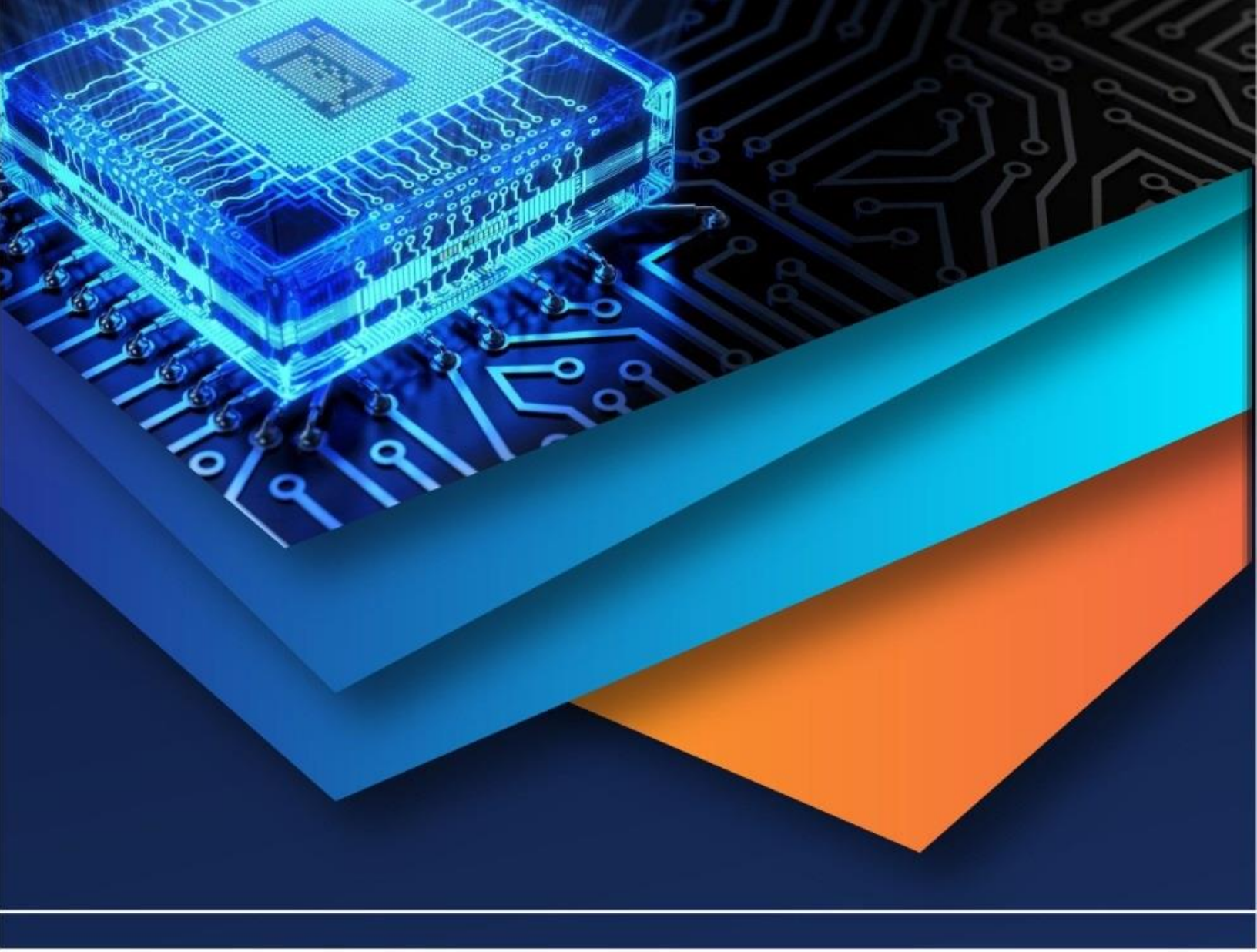
1) AI-based food recommendation systems
2) Automated delivery route optimization
3) Integration with digital payment gateways
4) Mobile application support

XIV. CONCLUSION

This research presented a Food Delivery Management System with Real-Time Order Tracking and Vendor Integration designed to improve efficiency in online food delivery services. By integrating customer ordering, vendor management, and delivery tracking into a single platform, the system enhances transparency and operational efficiency. The proposed system demonstrates significant improvements in order management and customer satisfaction.

REFERENCES

[1] P. Kumar et al., “Online Food Ordering System,” International Journal of Computer Applications, 2021
[2] S. Sharma, “Real-Time Web Applications,” IEEE, 2022
[3] React.js Official Documentation. (2024). React – A JavaScript Library for Building User Interfaces. https://react.dev/
[4] Node.js Foundation. (2024). Node.js v18 LTS Documentation.https://nodejs.org/docs/
[5] Express.js. (2024). Express – Fast, Unopinionated, Minimalist Web Framework for Node.js. https://expressjs.com/



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY