International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 13 Issue: 04 | Apr 2026 www.irjet.net p-ISSN: 2395-0072

A Service for Delivering Food Using MERN Stack: A Web-Based Solution for Restaurant Management

Rekha S Patil[1], Shakuntala[2], Sneha[2], Samiksha[2]

[1]Professor, Department of Computer Science and Engineering, PDACEK, Kalaburagi, India
[2]B.E Student, Department of Computer Science and Engineering, PDACEK, Kalaburagi, India

---

Abstract - *This paper presents the development and implementation of a web-based food delivery service tailored for a specific restaurant using the MERN (MongoDB, Express.js, React.js, Node.js) stack. The system aims to streamline restaurant operations by providing an integrated platform for customers to browse menus, place orders, and make secure payments. Restaurant administrators can manage menu items, process orders, and track their progress in real-time. The proposed solution emphasizes scalability, responsiveness, and security, providing small and medium-scale restaurants with a modern digital presence without relying on third-party platforms. The system ensures an enhanced customer experience, cost efficiency, and maintainability, representing a practical example of full-stack web application development.*

Keywords - MERN Stack, Food Delivery, Web Application, Full Stack Development, Online Ordering, Restaurant Management

I. INTRODUCTION

The increasing use of digital technologies and widespread internet access has transformed how customers interact with food service providers. Online food delivery platforms have become essential due to their convenience, speed, and ability to offer a seamless ordering experience. Users now expect modern features such as digital menu browsing, secure payments, and real-time order updates. As demand continues to rise, the need for efficient, scalable, and user-friendly food delivery systems has grown significantly.

However, many existing platforms still face challenges, including slow performance during peak hours, outdated interfaces, unreliable tracking mechanisms, and limited support for small vendors. Traditional systems also struggle to adapt to changing customer expectations, leading to decreased satisfaction and operational inefficiencies. Security concerns such as weak authentication and unsafe data handling further highlight the need for better technological solutions.

To address these issues, the MERN stack—MongoDB, Express.js, React.js, and Node.js—provides a robust foundation for building modern web applications. Its unified JavaScript environment enables seamless communication between the frontend and backend, ensuring flexibility, speed, and scalability. This paper presents the development of a MERN-based food delivery application featuring secure authentication, dynamic menu management, real-time order tracking, and an admin dashboard. The proposed system aims to enhance performance, usability, and reliability compared to traditional platforms.

II. LITERATURE REVIEW

Several studies have explored digital food-ordering systems and full-stack web technologies.

[1] Chauhan, Bhardwaj, Shaikh, and Mishra (2022) developed *Cooked with Care*, a food ordering website built using the MERN stack to support homemakers, mess owners, and small food businesses. The system enables sellers to digitally manage menus, interact with customers, and operate in a wireless environment. Their work highlights the importance of digital transformation for small-scale food providers but lacks advanced features such as real-time tracking and scalability, which motivates further enhancements in modern food delivery systems.

[2] Brindashree C. B. (2021) proposed a canteen automation system integrating a MERN-based web

interface with an online payment gateway. The system aims to eliminate traditional queueing and manual order mistakes by enabling students to place orders digitally. Although the platform improves operational efficiency and accuracy, it primarily focuses on institutional canteens and does not address commercial delivery requirements, leaving room for broader applications.

[3] Kulkarni, Khuba, and Shaikh (2021) presented an online food delivery system designed to streamline the ordering process for customers and restaurant staff. Their platform features digital menus, online ordering, and administrative tools for inventory and order management. While the system reduces human errors and enhances convenience, it does not fully explore modern interface design or scalable architectures needed for larger user bases.

[4] Mohammad (2023) analyzed the suitability of the MERN stack for building food delivery applications. The study discusses how MongoDB's flexible schema, Express.js and Node.js server efficiency, and React's dynamic UI capabilities collectively support responsive and scalable systems. The findings confirm MERN as a strong option for modern applications, though the study remains theoretical without presenting a complete implementation.

[5] Kumar, Jain, and Bajwa (2021) developed *Foodie*, an online food delivery app offering menu browsing, online payments, and admin-side control. The application incorporates machine learning algorithms, such as decision trees, to analyze user preferences and improve service quality. While the inclusion of machine learning adds value, the system has limited focus on real-time order tracking and advanced administrative functions.

[6] Gujar, Rao, Panpatil, Badhe, and Patil (2024) created a food delivery web application using the MERN stack with features such as real-time order tracking, secure payment integration, and dynamic menu listing. The system emphasizes user experience, providing intuitive UI elements and mobile compatibility. Although comprehensive, the platform can be enhanced with more advanced scalability features for large-scale deployment.

[7] Mponela, Shereef, and Tawarish (2024) proposed an online food ordering system primarily designed for restaurants and hotels. The application allows users to browse menus, customize orders, and place them via a mobile interface. Their work highlights the potential of mobile technology in replacing traditional ordering practices; however, it lacks an admin dashboard and does not incorporate web-based scalability.

[8] Marshettiwar (2025) introduced *Hunger-Bite*, a food ordering and donation system built to support both commercial and social needs. The system connects customers, restaurant owners, food donors, and NGOs through a unified platform. While the inclusion of donation features makes the system socially impactful, it diverges from traditional food delivery workflows and does not focus heavily on enhanced UI/UX or advanced backend scaling.

[9] Aulia, Zakir, and Dafitri (2017) developed an Android-based restaurant ordering mechanism to replace manual ordering with tablet or smartphone interfaces. Orders are sent wirelessly to the kitchen, helping reduce processing errors and improving service speed. However, the system remains limited to restaurant interiors and does not support online food delivery operations.

[10] Nikose, Hatwar, and Nikose (2023) proposed a cafeteria food ordering system that uses QR code technology to simplify ordering. Customers scan the QR code placed on tables to access menus, place orders, and track updates. Although efficient for cafeteria environments, the system lacks external delivery capabilities and real-time routing features found in modern food delivery apps.

[11] Dubey, Dhanve, Nikalje, and Bodhale (2023) designed an online food ordering system that includes order tracking, digital menus, rating features, and personalized recommendations. The system allows users to rate food items, helping restaurants improve service quality. While it enhances user engagement, the application does not

incorporate advanced security features or scalable backend structures.

[12] Kushwah, Sharma, Sharma, and Negi (2024) developed *Cafe Ease*, an improved online food ordering platform based on user-centered design principles. The study incorporates user surveys and usability testing to identify weaknesses in existing applications and proposes solutions for better UI/UX, accuracy, and delivery efficiency. However, the system remains conceptual and lacks a fully implemented backend.

[13] Raj Kishor (2025) explored the architecture and design principles behind full-stack food delivery platforms, covering backend development, user interface design, and scalability considerations. The study discusses modern tools and best practices for building robust systems but does not include a functional prototype or real-world testing.

Although these works provide strong foundations, gaps remain in scalability, real-time updates, and admin-side monitoring. This paper addresses these gaps with an integrated, optimized MERN architecture.

III. PROBLEM STATEMENT

The rapid growth of online food delivery services has significantly changed how customers interact with restaurants, yet many existing platforms continue to face persistent shortcomings that affect both user experience and operational efficiency. Traditional food ordering methods—such as phone calls or physical visits—are time-consuming, prone to miscommunication, and unable to meet the expectations of today's digitally driven consumers. Current online food delivery systems also encounter several challenges, particularly during peak hours, where high traffic leads to server overload, slow response times, and occasional order failures. These issues negatively impact user satisfaction and reduce the reliability of the platform.

Additionally, many platforms lack accurate real-time order tracking, leaving customers without clear information on the status of their order. This creates uncertainty and frustration, especially in time-sensitive situations. Small food vendors and home-based businesses also struggle to gain visibility on large commercial platforms due to high commissions and technical limitations, restricting their ability to reach a wider customer base.

Security remains another major concern, as improper authentication and weak data management expose users to risks such as data leaks, unauthorized access, and insecure transactions. Several existing applications are built on outdated or monolithic architectures, making them difficult to scale, maintain, or enhance with new features.

Therefore, there is a strong need for a modern, secure, efficient, and scalable food delivery web application. A MERN-based solution can address these issues by offering a unified JavaScript environment, real-time communication, dynamic interfaces, and robust backend management. This project aims to develop such a system to overcome the limitations of existing platforms and enhance both user satisfaction and administrative efficiency.

IV. OBJECTIVES

The primary objective of this project is to design and develop a modern, scalable, and user-friendly online food delivery web application using the MERN stack. The system aims to provide a seamless ordering experience through an intuitive and responsive user interface. Specific Specific objectives include:

1. Implement a fully functional frontend using React.js to enable dynamic food browsing, search, filtering, and cart management.
2. Develop a secure and modular backend using Node.js and Express.js to handle authentication, order processing, and system logic.
3. Integrate JSON Web Token (JWT)–based authentication to ensure secure user access and data protection.
4. Provide real-time order status updates to enhance customer engagement and transparency.

5. Build an admin dashboard for managing food items, orders, and platform operations.

Ensure application scalability, responsiveness, and maintainability for future enhancements.

V. PROPOSED SYSTEM

The proposed system is a MERN-based online food delivery web application designed to provide a fast, secure, and user-friendly ordering experience. The system uses React.js for an interactive frontend, Node.js and Express.js for backend processing, and MongoDB for storing user, product, and order data. It includes two main modules: a **User Module** for browsing menus, adding items to the cart, placing orders, making payments, and tracking order status, and an **Admin Module** for managing food items and processing orders. Secure JWT authentication, responsive design, and real-time status updates ensure a reliable and efficient platform.

A. System Architecture

The application follows a client-server architecture where: - React.js handles frontend UI rendering. - Express.js and Node.js handle backend logic and routing.
- MongoDB stores user, product, and order data.

B. Functional Components

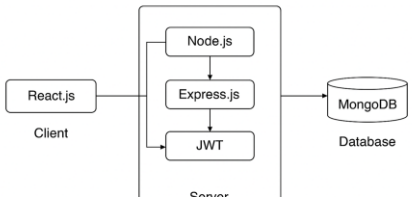

Fig 5a. Architecture of MERN based Food

1) User Module

- Browse food items
  Add items to cart
- Place orders and complete payments
- Track orders in real time

2) Admin Module

- Add/update/delete food products
- Manage orders and update status
- Monitor system activity

3) Authentication System

JWT-based login ensures secure access control.

4) Payment Integration

Supports Razorpay / Stripe / UPI.

5) Real-Time Order Tracking

Order stages include: *Order Placed → Confirmed → Preparing → Out for Delivery → Delivered*

VI. METHODOLOGY

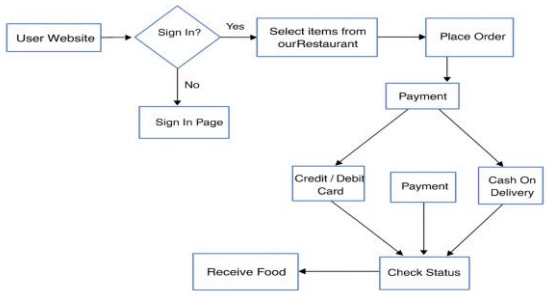

Fig 6a. Flowchart for MERN Based Food Delivery Application

We will follow the **Agile Methodology** to ensure efficient, iterative, and flexible development. The project will be divided into multiple **sprints**, each focusing on a key module. Instead of planning an entire project upfront, Agile breaks down large projects into small, manageable cycles called **sprints** or iterations.

Agile Methodology Life Cycle

Fig 6b. Agile Methodology

Agile Process in Our Project:

- **Planning & Requirement Analysis:** Identify system needs, research existing solutions, and define the MERN stack architecture.
- **System Design:** Create UI wireframes, database schema, and API endpoints.

1. **Development (Sprint-Based):**
   - **Sprint 1-2:** Setup environment, authentication system.
   - **Sprint 3-4:** Restaurant management, order placement.
   - **Sprint 5-6:** Payment gateway, real-time tracking, notifications.
   - **Sprint 7:** Review & rating system.

2. **Testing & Debugging:** Perform unit and system testing to ensure security and efficiency.

3. **Deployment & Maintenance:** Host on Vercel, Render, and MongoDB Atlas, with continuous monitoring and updates.

VII. Results and Discussion

The developed application, titled 'BiteSpot,' offers seamless user experience with modules for customer and admin roles. Customers can register, browse menus, manage carts, and track their orders, while admins manage menus, orders, and payment records. The platform supports responsive design across mobile and desktop screens, with optimized backend performance ensuring low latency. The system was tested for usability and responsiveness, confirming its stability and efficiency for small restaurant use cases.

Screenshots Placeholder:1. Home Page:

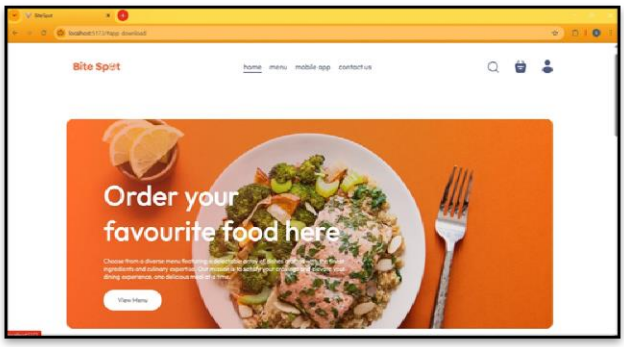

Fig 7.1: Home Page of BiteSpot Food Deliverv Website

2. Menu Page:

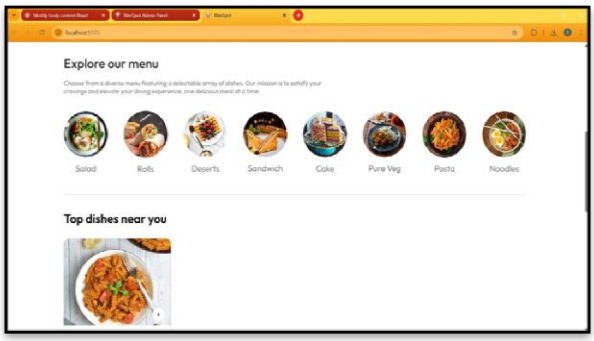

Fig 7.2: Menu Page of BiteSpot Food Delivery Website

3. Order Page:

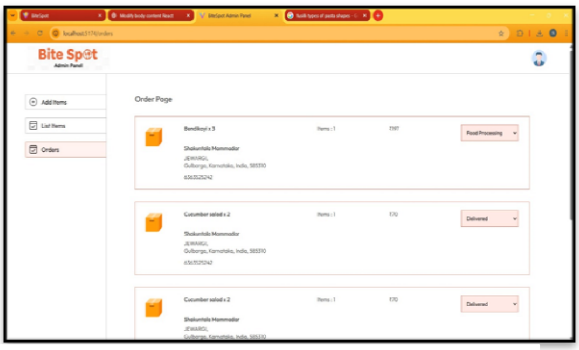

Fig 7.3: Order Page of BiteSpot Food Delivery Website

4. Admin Page:

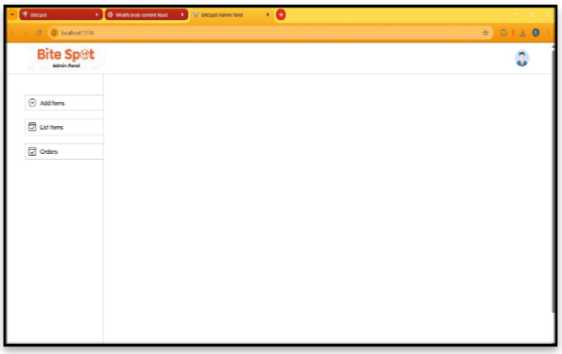

Fig 7.4: Admin Page of BiteSpot Food Delivery Website

VIII. CONCLUSION

This paper presented 'A Service for Delivering Food with MERN Stack,' a modern web-based food delivery application that demonstrates the effectiveness of full-stack JavaScript technologies. The system enhances operational efficiency for restaurants and provides customers with a reliable, secure, and user-friendly digital ordering experience. Future enhancements may include implementing live order tracking via WebSockets, integrating AI-driven food recommendations, and expanding to multi-restaurant support. The project contributes to practical understanding of MERN stack application development and its potential for small business digitization.

References

[1] A. S. Chauhan et al., 'Food Ordering Website "Cooked with Care" Developed Using MERN Stack', 2022.

[2] B. C. B., 'Implementation of Canteen Automation System with Payment Gateway Using MERN Stack', 2021.

[3] S. V. Kulkarni et al., 'Online Food Delivery System', 2021.

[4] D. Mohammad, 'Evaluating the Suitability of the MERN Stack in the Development of Food Delivery Applications', 2023.

[5] A. Gujar et al., 'Food Delivery Web Application with MERN', 2024.

[6]MongoDBDocumentation:https://www.mongodb.com/docs/

[7] Express.js Guide: https://expressjs.com/

[8]React.jsOfficialDocumentation:https://reactjs.org/

[9]Node.jsAPIDocumentation:https://nodejs.org/en/docs/