Swaad: Food Delivery Web Application with Dual OTP Verification using MERN stack

M Dolie Ciri
M. Tech Student, Andhra University College of Engineering(A), Visakhapatnam, Andhra Pradesh, India

***Abstract*—This paper explains the design and implementation of a secure web-based food delivery app specifically for single-restaurant operations, built with the MERN stack (MongoDB, Express.js, React.js, Node.js). This app features separate modules for admin, customers, and delivery partners. A main feature is the dual-stage One-Time Password (OTP) verification, which involves a pickup OTP that is generated and visible only to the restaurant and delivery partner during order collection, while a different delivery OTP is generated after pickup and sent to the customer, which the delivery partner must verify at handover. This layered OTP system enhances the order process, ensuring the authenticity and precision of all status updates and reducing the risk of accidental or fraudulent changes. The project demonstrates that using modern full-stack technologies together with security measures like OTPs allows for the creation of highly reliable, scalable, and tamper-proof food delivery solutions.**

***Index Terms*—Food Delivery Application, MERN stack, OTP Verification, Admin, Delivery Partner, Customer.**

1. INTRODUCTION

The food ordering and delivery processes have undergone a substantial shift in the past few decades. In the initial stages, customers would place orders over the phone, and restaurants would manage deliveries manually. With the surge in internet usage and smartphone adoption, food delivery has evolved into an all-digital experience. Customers can now see menus, place orders, make online payments, and get real-time delivery updates.

The platforms, namely Swiggy and Zomato, emerged to capitalize on this trend, which connects restaurants, delivery agents, and customers through web and mobile applications. Despite this, they still have significant security and accountability gaps. A higher number of platforms depend on status updates managed by delivery partners or backend automations, which often lack direct verification from either the restaurant or the customer.

As a result, these systems remain weak in the aspects of false confirmation of pickup or delivery by agents, order theft or misplacement during transit, customer disputes, and reputational damage because of fraudulent delivery claims. Research addresses the above challenges by presenting a secure food delivery web application built using the MERN stack - MongoDB for data persistence, Express.js and Node.js for server-side, and React.js for front-end interface. System defines three different roles — customer, delivery partner, and admin—each with dedicated modules, interfaces, and permission scopes. To ensure a reliable and tamper-free order process development, this application consolidates a dual One-Time Password verification mechanism at two critical handover points: One at the time of order pickup and another OTP at the time of order delivery.

1.1 Problem Statement

Current food delivery platforms lack a reliable verification mechanism to confirm that the order has been physically picked up from the restaurant or handed over to the correct customer. This problem presents the organization with false status updates, order loss, or theft, which ultimately hinders trust in service.

1.2 Objectives

1. Design and implement a MERN stack web application that simplifies interactions among the restaurant, its customers, and delivery partners.
2. Adding OTP-based verification at both the pickup and delivery stages to secure all critical order handovers.
3. Provide administrative error for manual assignment of delivery partner and complete visibility into OTP verification events.

4. To allow customers to track their orders and securely access delivery OTPs in a read-only view.

2. LITERATURE REVIEW

1. Netfood: A Software System for Food Ordering and Delivery

Netfood is a food ordering and delivery management system that allows customers to place both individual and group orders from multiple restaurants via a web interface. Built on a multi-layer Java Spring Boot architecture with Hibernate ORM and a MySQL database, which ensures efficiency and scalability. Security is imposed through JWT-based authentication. Administrators can manage restaurants, menus, and user roles, while customers have features like order history and friend management. Delivery staff utilize a proper Android mobile client, powered by Retrofit, for REST communication. The platform uses Angular for the frontend and sticks to the Scrum methodology [1].

2. Food Ordering Website "Cooked with Care" Developed Using MERN Stack

"Cooked with Care" is a web application built using the MERN stack—MongoDB, Express.js, React, and Node.js - that aims to help homemakers, mess owners, and small business owners to sell and manage food orders online. This platform offers features such as order management, menu updates, customer feedback, dynamic multi-role dashboards (admin, manager, user), secure payments, and AWS integration for image storage. Aspects consist of supporting one-time and subscription meal orders, as well as enabling ordinary individuals to run online food businesses. This paper compares the solution to others, noting a better user experience and broader inclusion of non-restaurant sellers [2].

3. Digital Ordering System for a Restaurant Using Android

This system introduces a digital ordering platform that uses Android tablets installed on restaurant tables, allowing customers to browse menus, place orders, provide feedback, and handle billing. The backend is built on a MySQL database and the solution with kitchen order and billing systems to streamline operations. This paper highlights the elimination of paper-based disorganization, a better customer ordering experience, and improved management controls, while acknowledging the upfront costs and maintenance needs associated with deploying tablets and their infrastructure [3].

4. Android Application Food Delivery Services

The application provides a user-friendly platform that connects customers, restaurants, and delivery personnel. Customers can browse menus, customize orders, and track deliveries in real-time. Restaurants manage menus and receive orders efficiently, reducing manual work. Delivery agents receive optimized updates, enabling timely delivery. This app provides multiple payment methods, ensuring secure transactions. Features are order history, ratings, and feedback to improve the user experience. Using geolocation, the app estimates delivery times and optimizes logistics. Executed using the Android SDK and RESTful APIs, which ensures cross-device compatibility and scalability, thereby improving operational efficiency and customer satisfaction [4].

3. METHODOLOGY

This web application is a development of a secure, role-based food delivery web application using the MERN stack, with a strong focus on making sure every food order handoff is verified through an OTP system. A web application is built in three components: frontend, backend, and database. The frontend of the application is built through React.js, which provides different interfaces for customers, admins, and delivery partners. The backend is designed employing Node.js and Express.js, which handles about logic, user authentication, payment confirmation, and OTP processing. For storing information, MongoDB Atlas is used as a secure cloud database. These separate parts communicate with each other through RESTful APIs, making the system easier to test, manage, and scale in the future.

The frontend is designed with React 19 using functional components and hooks like useState, useEffect, and useContext to keep data and overall application behavior smooth and responsive. React Router handles the app flow between pages, and data such of user sessions and orders are managed through the Context API for a balance between simplicity and flexibility. All communication with the backend occurs via Axios to ensure quick and reliable data exchange. For instant feedback, whether it's confirming an OTP, updating an order status, or showing an error, React-toastify is used to give timely notifications. Application adapts to the role of the user: customers can sign up, log in, order food, track deliveries, and view their OTPs; admins can manage menus, assign orders to delivery partners, and update the state of orders; delivery partners can only see the orders assigned to them and can validate OTPs at both the pickup stage and final delivery.

The backend ensures security via JWT-based authentication, holding user sessions private and protected. Whenever a customer places an order, two OTPs are generated: one for the restaurant pickup and one for customer delivery, which are saved securely in MongoDB. Delivery partners must enter the pickup OTP before the system allows them to mark the order as “Out for Delivery,” and then enter the delivery OTP when handing over the food to the customer, confirming successful delivery "Delivered". Payments are made through various payment methods, including Stripe and the Cash on Delivery method. The backend tracks for payment confirmation via webhooks before saving the order and creating the related OTPs. The backend itself is divided into organized route files and controllers, keeping the code simple to maintain and expand.

MongoDB Atlas acts as the database to store information. The customer users collection holds login details and profiles, while the order details save food items, customer details, and payment status, OTPs, timestamps, and order progress stages that are “Food Processing,” “Waiting for Pickup,” “Out for Delivery,” and “Delivered.” Data is validated and indexed with Mongoose, ensuring that information stays accurate and easy to query when needed.

The OTP system is the most important layer of safety and trust in the delivery process of this application. The pickup OTP is shown only to the admin after the order is placed and the status of the order changes from "Food Processing" to "Waiting for Pickup", which the delivery partner must enter at the time of collecting the order. The delivery OTP is visible to the customer only when the order status is "Out for Delivery" and must be given to the delivery partner at the time of handoff to confirm the status as "Delivered". The pickup and delivery OTPs are six-digit random unique codes stored as hash values rather than plain numbers. Wrong attempts are recorded, and rate limits are applied to prevent brute-force guessing.

The APIs that connect to the system are carefully customized according to the roles of users. Customers access login, registration, profiles, browsing the menu, managing orders, cancelling orders, and retrieving their OTPs. Admins have control over menus, order assignments, and OTP verification. Delivery partners can only get access to their assigned orders and OTP checks. Every confidential route is protected by the JWT-based authentication middleware, ensuring that only authorized users can access.

Development was carried out in Visual Studio Code with ESLint to enforce clean code and Prettier to maintain consistent formatting. During development, Nodemon used in the backend for automatic server restarts, while MongoDB Atlas provided secure cloud data management. All testing and deployment started locally, but the setup has been designed to make future cloud deployment straightforward.

Overall, this approach ensures that the application is modular, secure, and reliable. By merging role-based access, OTP verification for both pickup and delivery, and the Stripe payment system, the food ordering and delivery platform provides users with a safe and trustworthy experience. Alongside, the modular architecture leaves room for scalability and future upgrades without compromising security or performance**.** Figure 1 shows the interaction flow between components about how order progresses from placing an order to delivery through OTP generation and verification at both pickup and delivery stages.

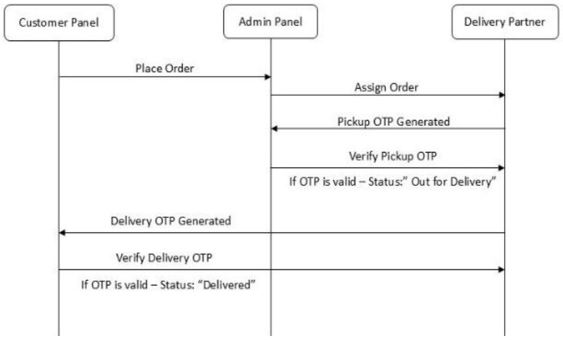


Figure 1: Sequence diagram of food delivery application

4. RESULTS

The MERN stack-based food delivery web application was developed to streamline order management, payment, and delivery verification. Testing with user roles and workflows to confirm the system fulfills all key functional and reliability criteria. The outcomes for each module and core feature are outlined below.

4.1 Functional Module Outcomes

4.1.1 Customer Module

The customer module of the application made ordering simple, secure, and stress-free. Customers can easily sign up and log in using a safe JWT-based authentication; user information is protected at every step. The clean and easy-to-use interface meant that placing orders and payments is seamless using Stripe Checkout, which added an extra layer of trust and security to transactions. The functioning of the customer module is shown in Figure 2. Each customer receives a unique OTP along with their order details, which is verified during delivery to ensure safety for both sides. Real-time order tracking gave customers visibility into every stage of their order, letting them follow progress and even cancel if required. All these features help customers to feel confident and in control, enjoying a transparent and secure experience from start to finish. Figure 3 shows the customer register/login page, need to enter their name, email, and password to continue ordering.

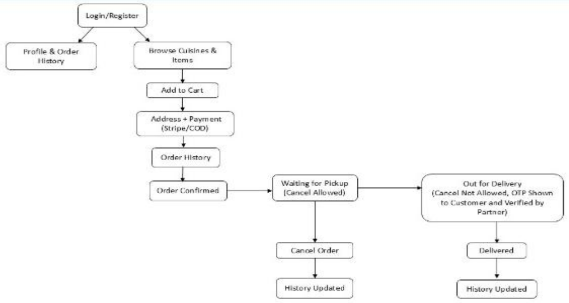
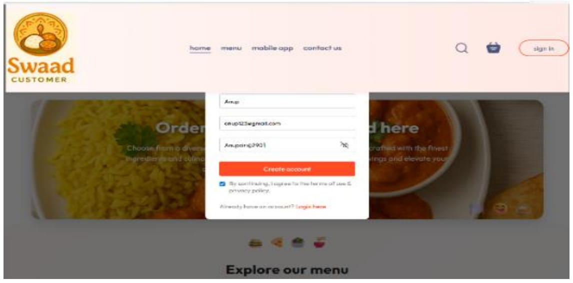


Figure 2: Workflow of Customer



Anup

anup123@gmail.com

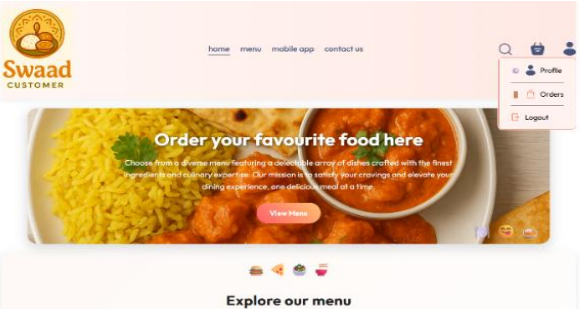
Anupang2901

☑ By continuing, I agree to the terms of use & privacy policy.

Already have an account? Login here

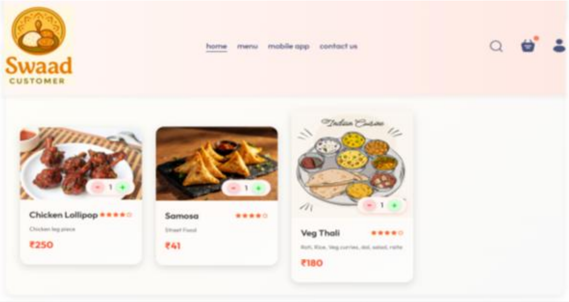
Figure 3: Register/Login Page

After logging in, the customer is directed to the Home page, as shown in Figure 4, where the dishes are displayed after the categories are selected, as shown in Figure 5. After adding items to the cart, it shows the total amount and item details, as shown in Figure 6. Order history page, where the orders are detailed into three different sections, namely active orders, delivered orders, and cancelled orders, as shown in Figure 7.



Choose from a diverse menu featuring a delectable array of dishes crafted with the finest ingredients and culinary expertise. Our mission is to satisfy your cravings and elevate your dining experience, one delicious meal at a time.

Figure 4: Home Page

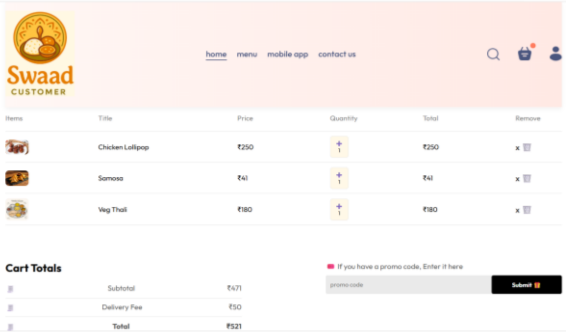


Chicken Lollipop
Chicken leg piece
₹250

Samosa
Street Food
₹41

Veg Thali
Roti, Rice, Veg curries, dal, salad, raita
₹180

Figure 5: Menu



Items	Title	Price	Quantity	Total	Remove
	Chicken Lollipop	₹250	1	₹250	x
	Samosa	₹41	1	₹41	x
	Veg Thali	₹180	1	₹180	x

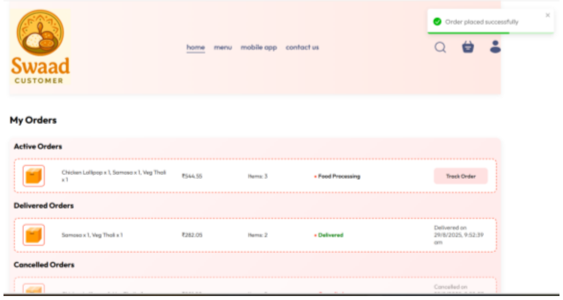
Cart Totals

Subtotal	₹471
Delivery Fee	₹50
Total	₹521

If you have a promo code, Enter it here

promo code

Figure 6: Cart



Order placed successfully

My Orders

Active Orders

Chicken Lollipop x 1, Samosa x 1, Veg Thali x 1 ₹544.55 Items: 3 • Food Processing

Delivered Orders

Samosa x 1, Veg Thali x 1 ₹282.05 Items: 2 • Delivered Delivered on 29/8/2025, 9:52:39 am

Cancelled Orders

Cancelled on

Figure 7: Order History

4.1.2 Admin Module

The admin module of the application serves as the system's central command center. Adding new menu items by specifying the name, price, and category, and these changes are instantly visible to customers. A list item view allowed for efficient oversight, with the flexibility to remove items when necessary. The functioning of the Admin module component, with adding items, a list of items, and orders, is presented in Figure 8. The orders received from the customers are mentioned in different sections as active orders, delivered orders, and cancelled orders, as shown in Figure 9.

Admins can easily track order details and their assigned pickup OTPs. All these features make the pickup more secure and efficient for order dispatch.

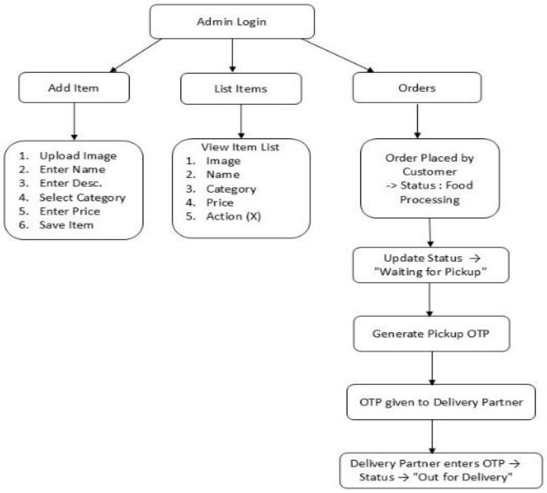
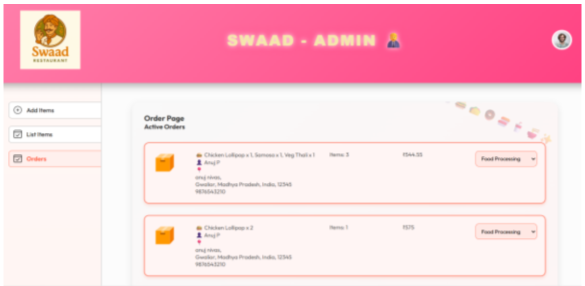


Figure 8: Workflow of Admin



SWAAD - ADMIN

Add Items
List Items
Orders

Order Page
Active Orders

Chicken Lollipop x 1, Samosa x 1, Veg Thali x 1 | Items: 3 | ₹544.55 | Food Processing
Anuj P
anuj nivas,
Gwalior, Madhya Pradesh, India, 12345
9876543210

Chicken Lollipop x 2 | Items: 1 | ₹575 | Food Processing
Anuj P
anuj nivas,
Gwalior, Madhya Pradesh, India, 12345
9876543210

Figure 9: Order History

4.1.3 Delivery Partner Module

The dual-OTP mechanism ensures authenticity at both pickup and drop-off. The component follows a strict role-based linear flow, preventing progress without successful OTP validation at each stage, thereby guaranteeing systematic tracking and secure order fulfilment, as shown in the workflow of the delivery partner in Figure 10. The delivery partner module ensures reliable and accountable last-mile delivery. These show a filtered list of only their assigned orders with the columns for verifying OTPs, as shown in Figure 11.

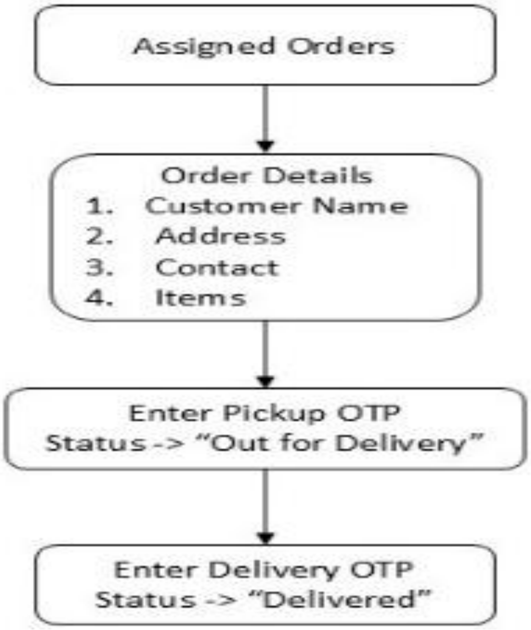
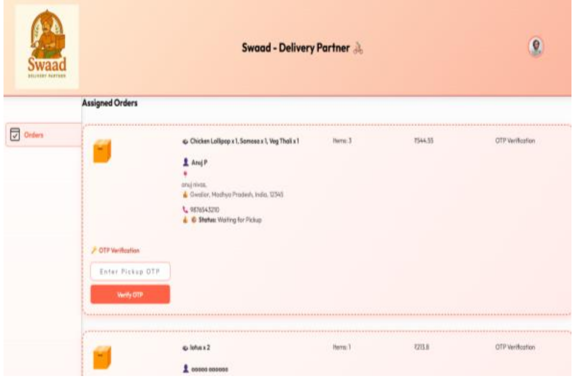


Figure 10: Workflow of Delivery Partner



Swaad - Delivery Partner

Orders

Assigned Orders

Chicken Lollipop x 1, Samosa x 1, Veg Thali x 1 | Items: 3 | ₹544.55 | OTP Verification
Anuj P
anuj nivas,
Gwalior, Madhya Pradesh, India, 12345
9876543210
Status: Waiting for Pickup

OTP Verification
Enter Pickup OTP
Verify OTP

lotus x 2 | Items: 1 | ₹213.8 | OTP Verification
aaaaa aaaaaa

Figure 11: Assigned Orders

5. CONCLUSION

The MERN-based food ordering and delivery web application successfully demonstrates a secure, role-driven approach to order management, payment, and delivery verification. By using OTP validation at the two most important stages, which ensures transparency, prevents unauthorized pickups and deliveries, and also enhances reliability in last-mile logistics. Role-based access controls protect customer and order data, and on the other hand, Stripe Checkout and MongoDB Atlas provide secure payments and scalable storage. It is designed as a prototype for a single restaurant and delivery partner. This web application has scope for future growth by making it easier to add more features like multi-restaurant support, secure delivery accounts, real-time location tracking, and reviews by customer. This application is designed with future growth in mind, making it ready to support new features and expansions easily as they arise.

6. ACKNOWLEDGMENT

I am thankful to Dr. A. Mary Sowjanya, Department of CS&SE, Andhra University, for guiding me throughout the project work.

REFERENCES

[1] C. E. Domokos, B. Séra, K. Simon, L. Kovács and T. B. Szakács, "Netfood: A Software System for Food Ordering and Delivery," 2018 IEEE 16th International Symposium on Intelligent Systems and Informatics (SISY), Subotica, Serbia, 2018, pp. 000143-000148.

[2] A. Shersingh Chauhan, S. Bhardwaj, R. Shaikh, A. Mishra and S. Nandgave, "Food Ordering website 'Cooked with care' developed using MERN stack", 2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2022.

[3] Ashutosh Bhargave, Niranjan Jadhav, Apurva Joshi, Prachi Oke, Prof. Mr. S. R Lahane (2018); Digital Ordering System for Restaurant Using Android; Int J Sci Res Publ 3(4) (ISSN: 2250-3153). http://www.ijsrp.org/research-paper-0413.php?rp=P16979

[4] L. C. Laoh, T. Adithia Pongantung and C. Mulalinda, "Android Application Food Delivery Services," 2020 2nd International Conference on Cybernetics and Intelligent System (ICORIS), Manado, Indonesia, 2020.

[5] Juned Pathan | Pratham Khobragade | Sohel Khan | Tayyab Sheikh | Prof. Rina Shirpurkar "Food Ordering Web Application" Published in International Journal of Trend in Scientific Research and Development (ijtsrd)

[6] A Multi-Stage OTP Authentication Protocol for Secure Logistics and Delivery Services, International Journal of Security and Networks, 2022

[7] One-Time Password (OTP) Solution for Two Factor Authentication: A Practical Case Study, Scar Efrén Acosta Mayorga and Sang Guun Yoo

[8] A. Tiwari, S. Kumar, P.N. Singh, D.Kapila, A. Shukla, and A. Pachori, "Building an E-commerce Platform with the MERN Stack" 2024 International Conference on Augmented Reality, Intelligent Systems, and Industrial Automation (ARIIA), Manipal, India, 2024.

[9] V. K. Singh, P. Awasthi, L. Tripathi, S. Thakur, and P. Singh, "Mern (Mongodb, Express-Js, React-js, Node-js) Stack Web-Based Streaming and Broadcasting Application," 2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT), Greater Noida, India, 2024.

[10] M. M. Rahman, M. R. Foysal, N. N. Moon and F. N. Nur, "TraFood: An Android Application for Food Delivery in Train," 2021 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON), Pune, India, 2021.