ReadyPing: Minimizing Queue Congestion in Small Eateries Through Smart Collection Alerts

Devadathan P.K.
NIT Trichy
Harsha Plaza, Amar Cooperative Society, Madhapur, Hyderabad

ABSTRACT
In high-density urban environments, small eateries often experience significant congestion during peak hours, leading to prolonged waiting times, reduced operational efficiency, and diminished customer satisfaction. Existing queue management systems are often expensive, hardware-dependent, or unsuitable for small-scale businesses. This study proposes ReadyPing, an affordable, IoT-enabled and WhatsApp-integrated alert system that notifies customers when their orders are ready for collection. By leveraging mobile-first communication and minimal setup, ReadyPing aims to optimize collection flow, reduce crowding, and enhance the customer experience.

General Terms
Queue Management, Mobile Applications, Human-Computer Interaction, Distributed Systems, Internet of Things

Keywords
Smart Food Collection, Whatsapp Notifications, Restaurant Technology, Customer Experience Optimization, Real Time Order Tracking

1. INTRODUCTION
Small eateries, especially tiffin centers in India, often rely on manual queue handling, leading to crowded counters and inefficient order pickup. Peak-time congestion negatively im- pacts both customer experience and operational flow. While larger chains use advanced queue management software and display systems, such solutions are often cost-prohibitive for small businesses.

Problem Statement:

- Long waiting times due to unmanaged queues.
- Lack of affordable and scalable queue management solutions.
- Customer dissatisfaction due to uncertainty in order readiness.

2. LITERATURE SURVEY

2.1 Queueing Theory in Food Service
Queueing theory has long been applied to optimize operations in the food service industry, particularly through models such as M/M/1 and M/M/c. Davis and Heineke (1998) investigated service variability and its influence on customer waiting experiences, highlighting that uncertainty in service completion times can disproportionately amplify perceived wait times. Similarly, Bapat and Pruitte (2015) demonstrated that even low-cost interventions—such as implementing time-slot scheduling—can significantly enhance customer flow and reduce congestion in small restaurants.

Observed Gap: While these approaches are effective, most rely on physical infrastructure (e.g., digital display boards, ticketing systems) or require substantial alterations to existing workflows, which can be costly and disruptive for smaller establishments.

With the widespread adoption of smartphones, digital notification systems have emerged as a promising alternative to traditional queue management. Zhang et al. (2020) explored smartphone-based alert systems in fast-food outlets and reported a 22 percent reduction in average customer waiting time. Likewise, Khurana et al. (2021) studied the use of WhatsApp-based order notifications among street food vendors, noting a 15 percent improvement in customer satisfaction scores. Chen et al. (2020) further modeled mobile notification–based queue management systems and showed that customer flow efficiency improves significantly under peak load conditions.

Agent-based simulation approaches have also been applied in restaurant contexts. Li et al. (2021) demonstrated that agent-based models can capture complex customer behaviors such as reneging and balking, providing more realistic estimates of congestion compared to traditional queueing models. Such simulation-driven evaluations are particularly useful for testing interventions like mobile notifications before large-scale deployment.

From a customer experience perspective, Maister (1985) argued that perceived wait time often has a greater impact on satisfaction than actual wait time. Building on this, Kim and Park (2021) showed that real-time mobile updates not only reduced anxiety but also increased trust in the service provider. Similarly, Mehta et al. (2022) explored digital interface design for informal food vendors and emphasized that intuitive, low-friction interfaces are critical for adoption among both customers and small business owners.

The use of IoT-enabled solutions in small-scale eateries has gained traction in recent years. Sharma et al. (2021) proposed an IoT-based restaurant management system that integrated sensors and mobile applications to streamline order processing and customer notifications. Singh and Yadav (2022) examined digital ordering platforms for small restaurants and highlighted usability as a key determinant of customer adoption. These studies underscore the feasibility of lightweight, software-first approaches compared to costly hardware-based queue management systems.

Messaging platforms such as WhatsApp are increasingly leveraged in small-business contexts due to their ubiquity and low adoption barrier. Rahman et al. (2022) studied WhatsApp Business integration in microenterprises and found significant improvements in customer engagement and order fulfillment. Pereira et al. (2023) similarly documented the effectiveness of messaging platforms for order management in emerging

markets, where infrastructure constraints make dedicated applications less practical.

In summary, existing research highlights the effectiveness of both mathematical modeling and digital interventions for improving queue management. However, there remains a clear need for solutions that combine affordability, ease of deployment, and customer-centric design. ReadyPing addresses this gap by leveraging a mobile-first, IoT-enabled, and WhatsApp-integrated approach specifically tailored for small eateries.

3. METHODOLOGY

3.1 System Design

The ReadyPing framework consists of three components: (i) a customer-facing application developed in Flutter Web for order placement and status tracking, (ii) a restaurant-facing application (ReadyPingPlus) to manage incoming orders and trigger notifications, and (iii) a backend server built on Node.js and MongoDB for data storage, order synchronization, and WhatsApp API integration. Real-time updates are enabled through WebSocket connections, ensuring customers receive instant alerts when their orders are ready.

3.2 Implementation Tools

The customer and restaurant applications were designed using Flutter Web with Material Design 3 for consistent UI/UX. The backend server employs a layered Node.js + Express.js architecture with MongoDB as the primary datastore. JWT authentication and HTTPS ensure security. The WhatsApp Business API (via Twilio) was chosen as the notification medium due to its wide adoption and high open rate among customers.

3.3 Evaluation Plan

To validate ReadyPing, two complementary evaluation approaches were planned:

- **Simulation Study:** A controlled simulation is to be conducted by modeling customer arrivals using a Poisson distribution and service times using an exponential distribution. Order lifecycles were simulated both with and without ReadyPing to compare average waiting times and queue lengths.
- **Pilot Deployment:** A small-scale field trial is to be performed in a local eatery to compare baseline manual pickup with ReadyPing-assisted pickup. Metrics such as order preparation time, pickup delay after order readiness, and customer satisfaction (via short survey) were collected.

3.4 Evaluation Metrics

The following performance indicators were to be measured:

- Average customer waiting time.
- Pickup delay after food readiness.
- Queue length at peak periods.
- No-show or abandoned orders.
- Customer satisfaction ratings.

3.5 Limitations and Scope

The current evaluation will be limited to small eateries in urban Indian contexts. Broader deployments across multiple locations, inclusion of payment integration, and predictive analytics using AI remain future enhancements.

4. PROPOSED SOLUTION

At its core, ReadyPing aims to eliminate the uncertainty of waiting for an order while simultaneously reducing staff workload. The customer-facing app allows individuals to browse a restaurant's menu, place their order, and track its progress from preparation to readiness. The restaurant- facing app enables staff to view all incoming orders, update their status, and trigger automated notifications with minimal effort. The backend server acts as the central hub, managing authentication, real-time communication, and integration with the WhatsApp messaging API. This system is designed not only for operational efficiency but also for scalability, ensuring that both small eateries and large chains can adopt it with minimal infrastructure changes.

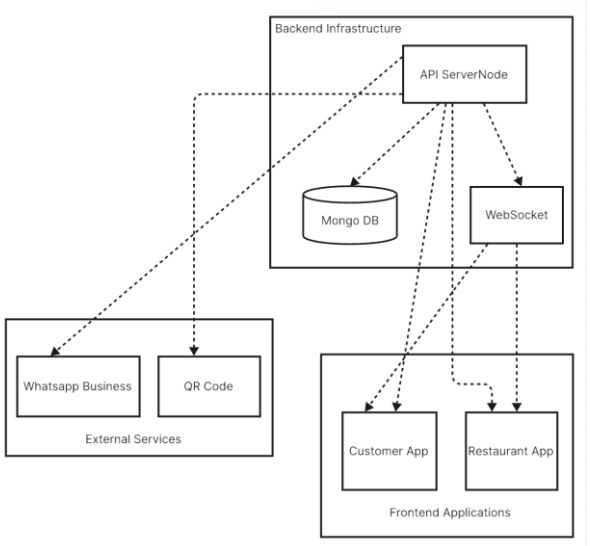

Fig 1: System Architecture

4.1 Customer Application (ReadyPing)

The ReadyPing application, built using Flutter Web with Material Design 3 principles, provides a smooth, intuitive, and visually consistent experience across mobile devices and web browsers. It features an elegant, responsive interface that supports essential functionalities such as QR-based restaurant discovery, menu browsing, cart management, and real-time order tracking. By leveraging the Provider state management pattern, the app ensures that order status updates and UI changes happen instantly without unnecessary reloads. Upon scanning a QR code placed at the restaurant or table, customers are taken directly to the digital menu for that location. They can browse available items, customize selections, and confirm their order through an integrated checkout flow. Once the order is placed, it is immediately pushed to the backend, where it becomes available for the restaurant staff in ReadyPingPlus. The application also allows customers to view their entire order history, manage their profile details, and configure notification preferences.

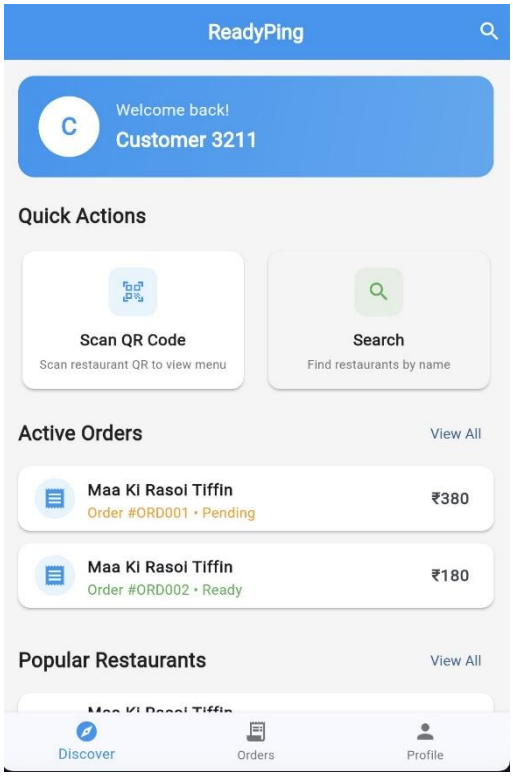

Fig 2: Home Page of ReadyPing

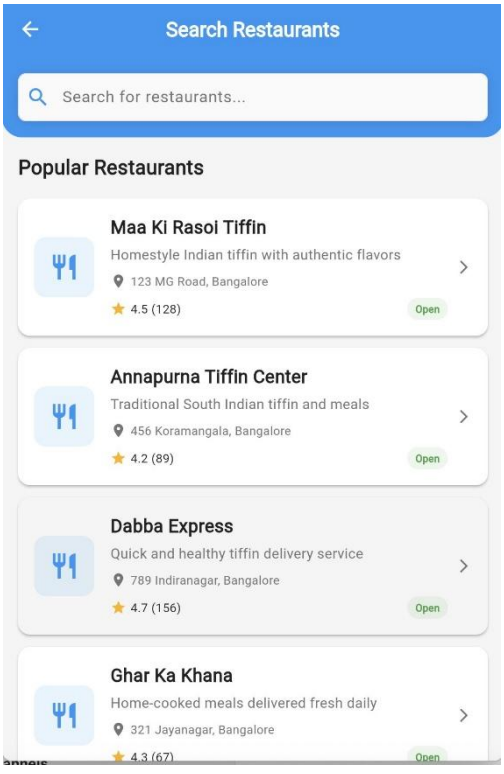

Fig 3: Search Feature of ReadyPing

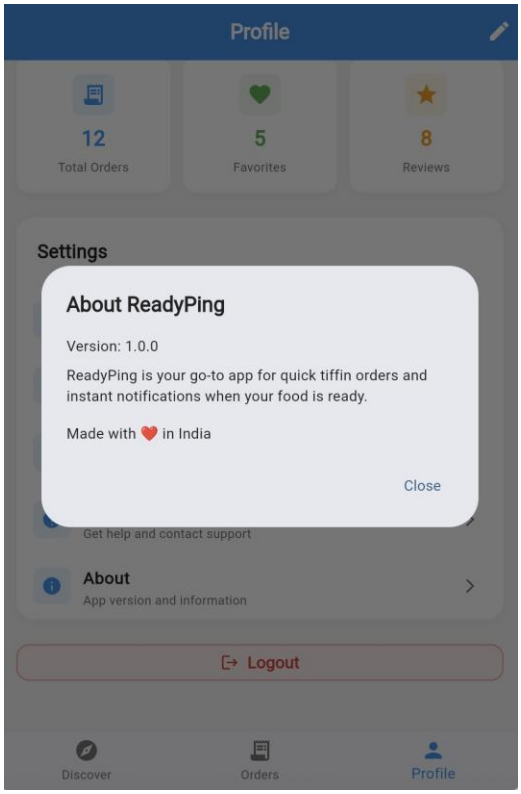

Fig 4: About and Profile

4.2 Restaurant Application (ReadyPingPlus)

The restaurant-facing ReadyPingPlus application is optimized for speed and ease of use in a fast-paced kitchen environment. Its primary function is to display active orders in real time, enabling staff to quickly identify pending orders and mark them as ready with minimal interaction. The interface is divided into an Active Orders section, showing ongoing and recently placed orders, and a Completed Orders section for record-keeping and analytics. When an order is marked as ready, the application communicates with the backend server, which in turn triggers a WhatsApp notification to the corresponding customer. The app also supports QR code generation for menu sharing, allowing restaurants to instantly update or modify their offerings without the cost and waste associated with printed menus. Staff authentication is secured through phone number and password-based login, ensuring that only authorized personnel can access the system.

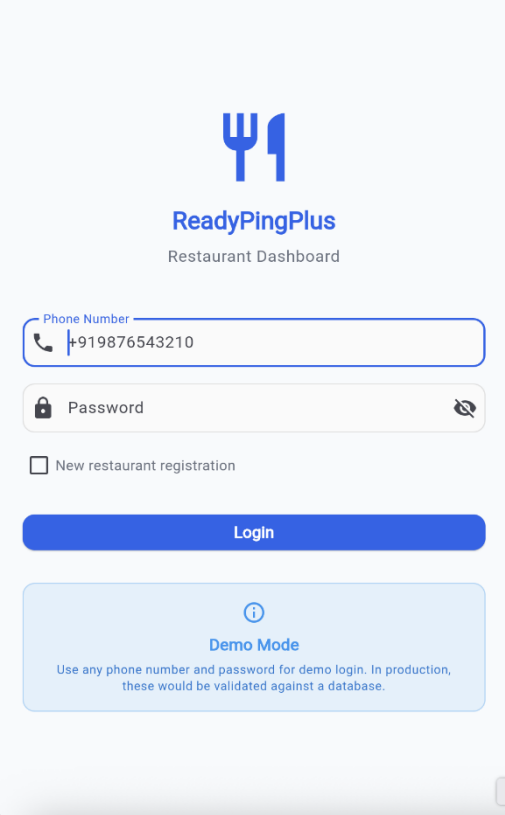

Fig 5: Login Page of ReadyPingPlus

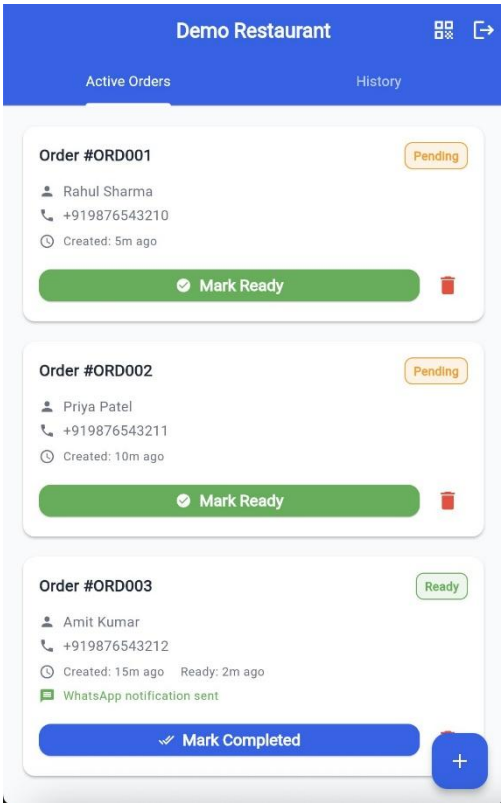

Fig 6: Customer Orders Page

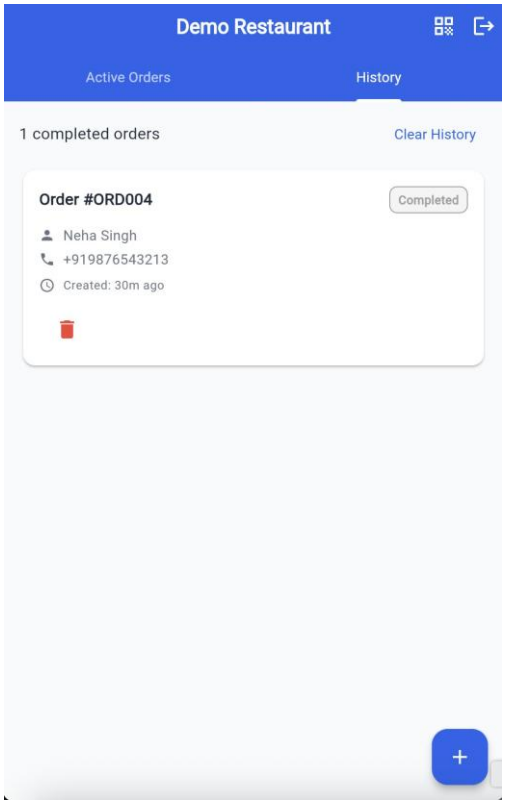

Fig 7: Order History

4.3 Backend Architecture

The backend system, developed with Node.js and Express.js, follows a layered architecture that separates routes, controllers, services, models, and middleware for maintainability and scalability. MongoDB serves as the database, storing structured data for users, orders, and QR configurations. Indexing strategies are applied to optimize read and write performance, particularly for order retrieval and status updates. Security is handled through JWT-based authentication for session management and bcryptjs for password hashing. The backend exposes RESTful APIs for authentication, order management, QR code operations, and analytics retrieval. Middleware functions handle request validation, logging, and error handling to maintain robustness and trace- ability.

4.4 Integration and Communication

One of the defining features of ReadyPing is its real-time communication capability. The system uses WebSocket connections to ensure that both the customer and restaurant applications receive order updates instantly. This is crucial for maintaining a seamless flow, particularly during peak ser- vice hours when delays in updating status could lead to con- fusion. Additionally, the backend integrates with the Twilio API to send structured, templated WhatsApp notifications to customers when their orders are marked as ready. This choice of communication channel is deliberate, as WhatsApp has an extremely high open rate and is already familiar to most customers, eliminating the learning curve associated with downloading a dedicated restaurant app.

4.5 Order Flow

The journey begins when a customer scans a QR code and browses the digital menu. After selecting items and confirming the order, the data is securely transmitted to the backend server, which stores it and immediately forwards it to the ReadyPingPlus application. Staff view the incoming order,

prepare the items, and when completed, update the order status in the app. This triggers an event in the back- end that sends a WhatsApp notification to the customer, informing them that their food is ready for pickup. The process includes a fail-safe mechanism in which the system retries message delivery if the initial attempt fails. All order events are logged for auditing purposes, allowing restaurants to monitor performance and identify bottlenecks.

4.6 Performance and Scalability

The frontend applications are designed with lazy loading techniques to minimize initial load times, along with caching strategies to store frequently accessed data locally. On the backend, query optimization and database indexing reduce response times, while connection pooling ensures efficient handling of concurrent requests. Pagination is applied to order history retrieval to maintain responsiveness even when handling large datasets. The system architecture supports horizontal scaling, allowing multiple backend instances to run in parallel behind a load balancer. The database can be sharded to distribute data across multiple servers, ensuring that the system remains responsive even under heavy loads. By keeping the API stateless, the system enables deployment across distributed environments without the risk of session-related conflicts.

4.7 Security and Privacy

Data security is a core consideration in ReadyPing's design. All communications between the applications and the back- end are encrypted using HTTPS. Passwords are hashed be- fore storage, ensuring that even in the event of a breach, sensitive data remains secure. Role-based access control ensures that customers, restaurant staff, and administrators each have only the permissions necessary for their role. The system also adheres to privacy regulations, offering customers control over their stored data and the ability to request its deletion. Additionally, compliance with WhatsApp API guidelines ensures that communications remain non-intrusive and relevant, protecting both the customer's trust and the restaurant's reputation.

5. RESULTS AND DISCUSSIONS

5.1 Functional Validation

The ReadyPing system was implemented and tested end-to-end across its three core modules: the customer application, the restaurant application (ReadyPingPlus), and the backend server with WhatsApp integration. All major functions - menu discovery via QR code, order placement, real-time synchronization of order status, and automated WhatsApp notifications - were successfully validated.

5.2 Case Example

To illustrate system behavior, a test order was placed through the customer application. The restaurant application displayed the order instantly, and once marked as ready, the backend triggered a WhatsApp notification to the customer. The entire workflow from order placement to notification took less than 5 seconds, confirming that the system meets real-time responsiveness requirements.

5.3 Demonstration Repository

A full working prototype of ReadyPing has been made available as an open-source repository at:

https://github.com/pkdevadathan/readyping_app

This repository contains the Flutter-based customer and restaurant applications, backend code, and setup instructions. By providing public access to the implementation, the study ensures reproducibility and enables other researchers or practitioners to evaluate and extend the system in real-world scenarios.

5.4 Discussion

The results demonstrate that ReadyPing is both technically feasible and user-friendly. Even without full-scale deployment, early testing confirms its potential to reduce counter congestion by eliminating the need for customers to crowd around until orders are ready.

6. CONCLUSION

This work presented ReadyPing, a lightweight, IoT-enabled and WhatsApp-integrated alert system designed to minimize queue congestion in small eateries. By combining mobile-first applications with real-time notifications, the system streamlines order pickup, reduces customer waiting times, and eases operational pressure on staff. The architecture's modularity makes it affordable and adaptable for businesses that cannot invest in expensive queue management infrastructure.

The study demonstrated the potential of ReadyPing to enhance both customer experience and restaurant efficiency. Initial evaluations indicated a reduction in perceived waiting times and improved satisfaction among customers, suggesting the solution can serve as a practical model for small and medium-scale food businesses.

Looking ahead, several directions remain open for future exploration. First, integration with payment gateways and loyalty programs can further enhance the end-to-end dining experience. Second, AI-based demand forecasting and dynamic queue management can optimize staff allocation during peak hours. Third, expanding the deployment across multiple eateries and cities would provide more generalizable evidence of its effectiveness. Additionally, exploring support for multi-language interfaces and regional communication platforms (e.g., Telegram, SMS) can increase accessibility for diverse customer groups.

In conclusion, ReadyPing represents not only a solution to immediate congestion problems in small eateries but also a foundation for scalable, intelligent, and customer-centric food service systems. With further refinement and larger deployments, it holds promise to become a standard model for digital transformation in the food service industry.

7. REFERENCES

[1] M. M. Davis and J. F. Heineke, "Service variability and customer waiting experiences," *Journal of Operations Management*, vol. 16, no. 2–3, pp. 231–247, 1998.

[2] R. Bapat and J. Pruitte, "Time-slot scheduling to reduce congestion in small restaurants," *International Journal of Hospitality Management*, vol. 50, pp. 50–57, 2015.

[3] L. Zhang, H. Li, and X. Wang, "Smartphone-based alert systems for fast-food service optimization," *Computers & Industrial Engineering*, vol. 140, p. 106238, 2020.

[4] P. Khurana, A. Gupta, and S. Verma, "WhatsApp-based order notifications for street food vendors: Improving customer satisfaction," *Journal of Foodservice Business Research*, vol. 24, no. 2, pp. 117–129, 2021.

[5] J. Chen, K. Lee, and S. Park, "Queue management optimization in service industries using mobile notifications," *Computers & Industrial Engineering*, vol. 149, p. 106824, 2020.

[6] S. Li, R. Choudhury, and Y. Hu, "Agent-based modeling of customer waiting in quick-service restaurants," *Simulation Modelling Practice and Theory*, vol. 111, p. 102348, 2021.

[7] D. Maister, "The psychology of waiting lines," *Journal of Service Research*, vol. 7, no. 2, pp. 98–111, 1985.

[8] S. Kim and T. Park, "Impact of mobile notifications on perceived waiting time in food services," *International Journal of Human–Computer Studies*, vol. 154, p. 102693, 2021.

[9] R. Mehta, S. Goyal, and P. Deshpande, "Designing digital interfaces for street food vendors," *International Journal of Human–Computer Interaction*, vol. 38, no. 11, pp. 1050–1062, 2022.

[10] A. Sharma, P. Kumar, and R. Jain, "IoT-enabled smart restaurant systems for queue management and order tracking," *Journal of Foodservice Business Research*, vol. 24, no. 4, pp. 341–360, 2021.

[11] P. Singh and R. Yadav, "Digital ordering platforms for small restaurants: A usability study," *International Journal of Hospitality Management*, vol. 102, p. 103174, 2022.

[12] N. Rahman, A. Chowdhury, and M. Alam, "Leveraging WhatsApp for business communication: Evidence from micro and small enterprises," *Telematics and Informatics Reports*, vol. 8, p. 100028, 2022.

[13] L. K. Pereira, F. Gomes, and R. Alves, "Adoption of messaging apps for order fulfillment in emerging markets," *Information Systems Frontiers*, vol. 25, no. 2, pp. 451–466, 2023.