International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 11 | Nov - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

ServeIQ: Smart Restaurant Management Platform

Ashutosh Watharkar[1], Sakshi Kumari[2], Vardhan Deokate[3], Atharv Nikam[4]

Department of Computer Science and Engineering, MIT ADT University, Pune, India

---

Abstract— The management of restaurant operations involves several activities, including menu management, customer ordering, table coordination, and billing. When these activities are handled through manual processes or separate systems, restaurants may face difficulties in maintaining organized operations and managing customer orders efficiently. To address these challenges, this paper presents ServeIQ, a web-based Smart Restaurant Management Platform designed to simplify restaurant operations through a centralized digital environment.

The proposed platform is developed using HTML, CSS, Bootstrap, and Python Flask, following a client-server architecture. It provides a customer-facing interface through which users can browse food items, explore menu categories, search for dishes, add items to a shopping cart, and place orders. The platform also incorporates QR-based table management to associate customer orders with specific restaurant tables. On the administrative side, ServeIQ provides a dashboard for managing menu items, monitoring orders, accessing table information, and handling bills and payment-related activities.

The system also incorporates Razorpay payment gateway integration to support online payment workflows. The application is designed with attention to data handling, user access, and security considerations, with further security improvements identified as part of ongoing development.

This paper presents the proposed system architecture, major functional modules, implementation approach, and operational workflow of ServeIQ. It also discusses the platform's capabilities, testing requirements, current limitations, and possible future enhancements. The proposed system aims to provide restaurants with a structured digital solution for managing customer ordering and administrative activities through a common web-based platform.

Index Terms— Smart Restaurant Management, ServeIQ, QR-Based Ordering, Digital Menu, Restaurant Management System, Python Flask, Web Application, Order Management, Table Management, Razorpay Integration, Online Payment.

I. INTRODUCTION

The restaurant industry is gradually adopting digital technologies to improve daily operations and provide better services to customers. Traditional restaurant management often involves manual activities such as taking food orders, maintaining menus, managing tables, preparing bills, and tracking customer requests. When these activities are handled separately or through paper-based records, they can become difficult to coordinate, particularly during busy operating hours. Maintaining accurate order information and ensuring smooth communication between customers and restaurant staff are also important challenges.

Web-based applications provide an opportunity to simplify these activities by bringing different restaurant operations into a common digital platform. Through an online interface, customers can view available food items, explore menu categories, select dishes, and place orders. Similarly, restaurant administrators can use a centralized dashboard to manage menu information, monitor orders, access table details, and handle billing activities. Such an approach can reduce dependence on manual processes and make restaurant information easier to manage.

One useful technology in this area is Quick Response (QR) code-based ordering. QR codes can be associated with individual restaurant tables, allowing customers to access the digital menu by scanning the code using a smartphone. This approach can simplify the process of identifying table-specific orders and provide a convenient way for customers to interact with restaurant services. The integration of digital menus and centralized order management can further support communication between customers and restaurant administrators.

To address these requirements, this paper presents ServeIQ, a web-based Smart Restaurant Management Platform developed using HTML, CSS, Bootstrap, and Python Flask. The application follows a client-server architecture in which the frontend provides the customer and administrative interfaces, while the Flask backend handles application requests and business logic.

The customer-facing interface allows users to browse food items, explore different menu categories, search for dishes, add selected items to a shopping cart, and proceed with the ordering process. The platform also includes QR-based table management to associate orders with specific restaurant tables. On the administrative side, the system provides a dashboard through which restaurant staff can manage menu items, monitor orders, view table information, and handle billing and payment-related activities.

ServeIQ also incorporates Razorpay payment gateway integration to support online payment workflows. The application considers security requirements related to user access, data handling, and transaction processing. Further improvements to the security mechanisms are considered part of the ongoing development of the platform.

The primary objective of ServeIQ is to provide an integrated digital environment that connects customer ordering activities with restaurant administration. By combining menu browsing, QR-based table interaction, order management, billing, and payment gateway integration, the platform aims to simplify restaurant-related workflows through a common web application.

The main contributions of this work are summarized as follows:

1. Development of a web-based Smart Restaurant Management Platform using HTML, CSS, Bootstrap, and Python Flask.
2. Implementation of a customer-facing digital menu with food categories, search functionality, and shopping cart support.
3. Integration of QR-based table management to support table-associated ordering.

4. Development of an administrative dashboard for managing menu items, restaurant tables, orders, and billing activities.
5. Integration of the Razorpay payment gateway to support online payment workflows.
6. Identification of functional requirements and security considerations for further system improvement and evaluation.

The remainder of this paper is organized as follows. Section II presents the literature survey and discusses existing restaurant management approaches. Section III explains the proposed ServeIQ system, its architecture, and major functional modules. Section IV discusses the implementation outcomes, system functionality, and proposed evaluation scenarios. Finally, Section V presents the conclusion and possible future enhancements.

II. LITERATURE SURVEY

The use of digital technologies in the restaurant industry has increased as businesses look for better ways to manage customer orders, menus, and daily operations. Traditional ordering methods often involve printed menus and manual order recording, which can create delays and make order management difficult during busy hours. Web-based restaurant management systems and QR-based ordering platforms provide an alternative by allowing customers and restaurant staff to access information through a common digital platform.

A. QR-Based Menu and Ordering Systems

Ozturkcan and Kitapci (2025) discussed QR code menus as a digital solution for the hospitality industry. Their study provides an understanding of how QR-based menus can support restaurant service operations and the adoption of digital ordering interfaces [3].

Ula (2025) examined restaurant owners' and managers' perspectives on QR code menu adoption in restaurant service operations. The study highlights the importance of considering operational requirements when introducing QR-based menu systems. These findings are relevant to the development of digital restaurant platforms that aim to simplify menu access and ordering activities [2].

B. Customer Experience with Digital Menus

Hewage, Boman, and Lefebvre (2024) investigated the relationship between QR code menus and customer loyalty. Their research highlights the importance of considering customer experience when introducing digital menus into restaurant environments [1]. This indicates that introducing technology alone does not guarantee a better customer experience. The interface should be easy to understand, and customers should still have access to staff assistance when required.

Ashrafi, Iskender, and Shahid (2026) examined QR menu use behavior and customer-related factors in restaurant environments. Their research provides further context for understanding customer adoption of digital menu systems and the role of technology in restaurant services [4].

C. Integrated Restaurant Management Platforms

Existing restaurant management solutions demonstrate the importance of connecting different operational activities through digital systems. Instead of handling menu updates, customer orders, table information, and billing separately, an integrated platform can provide a centralized interface for managing these activities. This approach can help restaurant administrators monitor orders and maintain more organized operations.

ServeIQ follows this approach by combining a customer-facing digital menu and ordering interface with an administrative dashboard. Customers can browse food categories, search for dishes, add items to the cart, and place orders. The administrative interface provides facilities for managing menu items, viewing orders, handling table information, and accessing billing and payment-related activities.

D. Research Gap

The reviewed studies discuss QR-based ordering, digital menu adoption, and customer experience in restaurant environments. They also identify practical considerations related to usability, accessibility, customer preferences, and restaurant operations. These findings highlight the importance of considering customer ordering and restaurant-side administration together rather than treating them as completely separate activities.

ServeIQ addresses this project requirement by providing a common web-based platform that connects customer ordering with restaurant administration. It includes QR-based table management and Razorpay payment gateway integration to support online payment workflows. The platform is developed using HTML, CSS, Bootstrap, and Python Flask. Its objective is to provide a structured approach to managing customer orders and restaurant activities through a centralized application.

III. PROPOSED SYSTEM

A. System Overview

ServeIQ is a web-based Smart Restaurant Management Platform developed to simplify restaurant operations by connecting customer ordering activities with administrative functions. The system provides a centralized environment where customers can explore the restaurant menu, select food items, and place orders, while administrators can manage menu details, monitor orders, access table information, and handle billing activities.

The application follows a client-server architecture. The frontend is developed using HTML, CSS, and Bootstrap to provide an interactive and responsive user interface, while Python Flask is used for backend development to process application requests and manage system operations. The platform also incorporates QR-based table management and Razorpay payment gateway integration, which is currently configured in test mode to support the testing of payment-related workflows.

B. System Architecture

The proposed system consists of the following major components:

1. Customer Interface Module

The customer interface allows users to access the digital restaurant menu through a web browser. Customers can explore different food categories, search for specific dishes, view item details and prices, and add selected items to their shopping cart. The cart displays the selected items along with

an order summary before the customer proceeds with the ordering process.

2. QR-Based Table Management Module

The platform includes QR-based table management to support digital interaction between restaurant tables and the ordering system. QR codes are associated with table information, providing a way to connect the digital ordering process with restaurant seating arrangements. This module aims to make table-related order handling more organized.

3. Order Management Module

The order management module handles customer order information and provides administrators with access to order details. Restaurant administrators can monitor incoming orders and review the selected food items. This helps maintain a centralized record of customer ordering activities.

4. Admin Dashboard Module

The admin dashboard provides a common interface for restaurant administrators to manage important platform activities. It includes options for managing menu items, viewing table information, monitoring orders, accessing bills, and reviewing payment-related details. The dashboard also displays operational information, including revenue summaries and popular menu items.

5. Payment Module

ServeIQ incorporates Razorpay payment gateway integration, currently configured in test mode. This enables the payment workflow to be tested without processing actual financial transactions. In addition, the platform provides a pay-at-reception option, allowing customers to complete their payments directly at the restaurant.

6. Backend Processing Module

The backend is developed using Python Flask and acts as the central processing component of the application. It manages communication between the frontend and the system's functional modules by handling incoming requests, processing application data, and returning appropriate responses to the user interface.

C. Working Procedure

The general working procedure of ServeIQ can be explained through the following steps:

1. The customer opens the restaurant's web application through a web browser.
2. The customer accesses the digital menu and explores the available food categories.
3. The customer selects the required food items and adds them to the shopping cart.
4. The customer reviews the selected items and the order summary.
5. The ordering process uses the platform's table-related information where applicable.
6. The customer proceeds with an available payment option, either through the Razorpay test-mode workflow or the pay-at-reception option.
7. The order information becomes available through the administrative interface for restaurant-side monitoring.
8. The administrator can manage menu information, review orders, access table details, and handle billing-related activities.

D. Expected Benefits

The proposed system aims to provide the following benefits:

- Centralized Management: Brings customer ordering and restaurant administration together within a single platform.
- Digital Menu Access: Allows customers to browse food items and categories without depending entirely on printed menus.
- Organized Order Handling: Provides an administrative interface for monitoring and reviewing customer orders.
- Table Management: Supports the association of QR-based table information with restaurant operations.
- Payment Workflow Testing: Provides Razorpay test-mode integration for evaluating online payment-related activities.
- Accessible Interface: Uses web technologies to provide access through commonly available web browsers.

Overall, ServeIQ presents a centralized approach to restaurant management by combining digital menu browsing, customer ordering, table-related information, administrative operations, and payment workflow support within a single web-based platform.

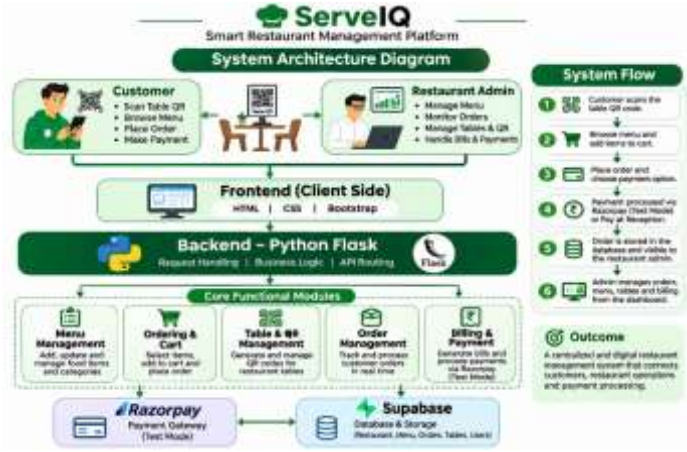


Figure 1 – System Architecture of ServeIQ

IV. RESULTS AND DISCUSSION

A. Implementation Overview

The ServeIQ platform was developed as a web-based application using HTML, CSS, Bootstrap, and Python Flask. The implementation provides separate interfaces for customers and restaurant administrators. The customer interface focuses on digital menu browsing and order placement, while the administrative dashboard provides access to various restaurant management activities.

The platform brings different restaurant-related functions together through a common application. Its interface is designed to simplify food selection, order review, and administrative operations by providing users with an organized and accessible environment.

B. Customer Interface

The customer interface provides a digital menu through which users can explore available food items. It includes food categories, search and filtering options, item prices, and a shopping cart. Customers can select dishes, add them to the

cart, and review their order summary before proceeding with the ordering process.
The interface also provides payment choices, including pay-at-reception and a Razorpay test-mode workflow. This allows the payment process to be tested without conducting actual financial transactions.



Figure 2. Customer Interface of ServeIQ

C. Administrative Dashboard

The administrative dashboard provides a centralized view of restaurant operations. It includes options for managing menu information, viewing orders, accessing table and QR-related information, and handling bills and payment-related activities.
The dashboard also presents operational information, such as revenue summaries and popular menu items. These features help administrators access important restaurant information through a single interface rather than relying on separate management tools.
tools.



Figure 3. Administrative Dashboard of ServeIQ

D. Payment Workflow

The platform incorporates Razorpay payment gateway integration, currently configured in test mode, to support the online payment workflow. This integration allows payment-related interactions to be tested without processing actual customer payments. The pay-at-reception option is also available for customers who prefer to complete their payment directly at the restaurant.

E. Discussion

The implementation demonstrates how customer-facing ordering functions and restaurant-side administrative activities can be brought together in a single web application. The digital menu and shopping cart provide a structured ordering experience, while the administrative dashboard gives restaurant staff access to menu, order, table, and billing information.
The use of Python Flask provides the backend framework for processing application requests, while HTML, CSS, and Bootstrap support the presentation and interaction layer. The Razorpay test-mode configuration also makes it possible to evaluate the payment workflow during project testing and demonstration.
Although the platform provides several important restaurant management functions, further testing is required to evaluate its reliability, usability, security, and performance under different operating conditions. Live payment processing and additional production-level improvements can also be considered in future development.
Overall, ServeIQ demonstrates the practical application of web technologies in connecting digital customer ordering with restaurant administration through a centralized platform.

V. CONCLUSION AND FUTURE SCOPE

A. Conclusion

This paper presented ServeIQ, a web-based Smart Restaurant Management Platform developed to simplify restaurant operations through a centralized digital system. The platform combines customer-facing menu browsing and ordering facilities with administrative functions for managing menus, monitoring orders, accessing table information, and handling billing activities.
The application is developed using HTML, CSS, Bootstrap, and Python Flask. Its customer interface allows users to explore food categories, search for dishes, add items to the shopping cart, and proceed with the ordering process. The administrative dashboard provides restaurant staff with a common interface to access important operational information. The platform also incorporates QR-based table management to support digital restaurant operations.
Furthermore, ServeIQ includes Razorpay payment gateway integration to support online payment workflows, currently configured for testing. Security considerations have also been included in the development process; however, further implementation and testing are required to strengthen the platform's security mechanisms.
Overall, ServeIQ demonstrates how web technologies can be used to connect customer ordering and restaurant administration within a single application. It provides a foundation for further development towards a more organized and efficient digital restaurant management solution.

B. Future Scope

Although ServeIQ provides the basic functionality required for digital restaurant management, several improvements can be considered for future development:

1. Enhanced Payment Processing: Further improve the payment integration to support reliable and secure online transactions in a production environment.
2. Enhanced Security: Strengthen authentication, authorization, session management, and data protection mechanisms.
3. Advanced Order Tracking: Introduce improved order status updates so that customers and restaurant staff can monitor order progress more effectively.
4. Inventory Management: Add stock tracking and automatic availability updates for food items based on ingredient availability.
5. Reports and Analytics: Develop detailed reports for sales, popular dishes, customer orders, and overall restaurant performance.
6. Mobile Compatibility: Further optimize the interface for smartphones and tablets to provide a smoother experience across different devices.

7. Performance and Usability Testing: Conduct testing with actual restaurant users to evaluate application reliability, usability, and performance under different operating conditions.

These enhancements can help extend ServeIQ into a more comprehensive restaurant management platform capable of supporting a wider range of restaurant operations.

REFERENCES

[1] G. S. U. Hewage, L. Boman, and S. Lefebvre, "Deciphering the dilemma: The surprising impact of QR code menus on diminishing customer loyalty," *Journal of Hospitality and Tourism Management*, vol. 61, pp. 187–190, 2024. doi: 10.1016/j.jhtm.2024.10.006.

[2] P. B. Ula, "The perceptions of the restaurants towards QR code menu adaptation in the restaurant service operation," *International Journal of Gastronomy and Food Science*, vol. 41, Art. no. 101250, 2025. doi: 10.1016/j.ijgfs.2025.101250.

[3] S. Ozturkcan and O. Kitapci, "A sustainable solution for the hospitality industry: The QR code menus," *Journal of Information Technology Teaching Cases*, vol. 15, no. 1, pp. 2–7, 2025. doi: 10.1177/20438869231181599.

[4] D. M. Ashrafi, A. Iskender, and T. Shahid, "Bytes to bites: Investigating QR code menu use behavior and green satisfaction in the restaurantscapes through a hybrid PLS-SEM and machine learning approach," *Journal of Foodservice Business Research*, vol. 29, no. 6, pp. 1599–1645, 2026. doi: 10.1080/15378020.2025.2466873.