PCMJ

Progress in Computer and Mathematics Journal (PCMJ)
volume 3 [November, 2025]
e-ISSN: 3030-6728
Website: fskmjebat.uitm.edu.my/pcmj

QUEUE MANAGEMENT SYSTEM WITH AUTOMATIC ASSIGN QUEUE FEATURE USING GEOFENCING

LUQMAN SYAHMI BIN AZRI
Universiti Teknologi MARA, luqmansyah23@gmail.com

MOHAMAD ASROL BIN ARSHAD*
Universiti Teknologi MARA, mohamad_asrol@uitm.edu.my

Abstract

The increasing inefficiencies in traditional queue management systems, particularly prolonged waiting times and manual registration processes, necessitate innovative solutions. This study introduces a Queue Management System with an Automatic Assign Queue Feature Using Geofencing to enhance service efficiency by automatically assigning queue numbers based on a user's real-time location. By utilizing geofencing, the system detects a user's arrival at a service provider's location and assigns a queue number, reducing manual intervention. The system follows the Waterfall Model methodology, covering five phases: Requirement Analysis, Application Design, Implementation, Testing, and Documentation. The system integrates Flutter for mobile development, Laravel for the backend, MySQL for data management, and Google Maps API for geofencing-based queue automation. Functional and accuracy testing were conducted, with results confirming a 100% success rate in queue number assignment. Findings indicate that the system significantly reduces waiting times and improves queue distribution efficiency, benefiting both customers and service providers. Future enhancements could include multi-location service provider integration and cross-platform support for iOS to ensure broader accessibility.

Article Info

Received: March 2025
Accepted: September 2025
Available Online: November 2025

Keywords: Queue Management System, Geofencing, Mobile Application

INTRODUCTION

Queue Management Systems (QMS) are essential in service environments such as healthcare, banking, and government services to streamline customer flow, allocate resources efficiently, and minimize waiting times (Ghazal et al., 2020). Traditional QMS rely on manual registration, leading to inefficiencies and dissatisfaction among customers due to long waiting times and unoptimized procedures (Patil & Thakur, 2019). The implementation of digital queue management solutions has been suggested to mitigate these inefficiencies, but many systems still lack automation and location-based adaptability (Weerakoon et al., 2019).

Geofencing technology offers a solution by creating a virtual perimeter around a service provider's location, allowing for automatic queue number assignment when a user enters the predefined area (Li et al., 2013). This technology has been successfully applied in various industries for location-based notifications and tracking, proving its effectiveness in optimizing resource allocation and improving user experience (Jaya et al., 2021). This paper explores the development and implementation of a queue management system that integrates geofencing to enhance user experience and optimize service provider operations.

LITERATURE REVIEW

Queue Management Systems (QMS) have been widely studied due to their significance in improving service delivery across various industries. According to Ghazal et al. (2020), an efficient QMS helps in reducing customer waiting time and enhances service efficiency in institutions such as banks, hospitals, and government offices. The traditional approach of managing queues manually has been found to be inefficient, leading to customer dissatisfaction due to prolonged waiting times (Patil & Thakur, 2019). The emergence of automated and intelligent queue management systems has gained traction to address these inefficiencies.

Mobile Application Frameworks

Several mobile development frameworks are used to implement queue management systems, with Flutter and React Native being the most prominent. According to Mamoun et al. (2021), Flutter provides high-performance capabilities and a rich set of UI components, making it an ideal choice for mobile applications that require real-time interaction. React Native, on the other hand, is widely used for cross-platform applications but has performance limitations due to its dependency on a JavaScript bridge (Fatkhulin et al., 2023). Due to its superior performance, Flutter was chosen for this project.

Location-Based Services and Geofencing

Location-based services (LBS) have revolutionized how mobile applications interact with users in real-time. According to Reclus & Drouard (2009), geofencing enables applications to detect when users enter or exit a predefined geographical area and triggerspecific actions. This technology has been utilized in various industries, including marketing, logistics, and service automation (Brown & Harmon, 2014). By leveraging geofencing, queue management systems can automatically assign queue numbers upon detecting a user's presence at a service location (Li et al., 2013). Google's Geolocation API provides the backbone for accurate geofencing applications (Ponce Atenci et al., 2020).

Queue Algorithms

Queue management relies on efficient scheduling algorithms to allocate resources optimally. The First-Come, First-Served (FCFS) algorithm is one of the simplest yet most effective strategies for ensuring fairness in service allocation (Sasmito et al., 2023). According to Pemasinghe & Rajapaksha (2022), FCFS ensures that customers are served in the order they arrive, preventing queue jumping and unfair prioritization. Alternative scheduling methods, such as Round Robin (RR), provide time-sliced allocations but introduce unnecessary complexity in service environments where waiting order is critical (Mangukia et al., 2021). Given its simplicity and effectiveness, FCFS was chosen for this project.on-based services (LBS) have revolutionized how mobile applications interact with users in real-time. According to Reclus & Drouard (2009), geofencing enables applications to detect when users enter or exit a predefined geographical area and trigger specific actions. This technology has been utilized in various industries, including marketing, logistics, and service automation (Brown & Harmon, 2014). By leveraging geofencing, queue management systems can automatically assign queue numbers upon detecting a user's presence at a service location (Li et al., 2013). Google's Geolocation API provides the backbone for accurate geofencing applications (Ponce Atenci et al., 2020).

Related Works and Comparative Analysis

Several existing applications address queue management but lack automated geofencing capabilities. MySejahtera, a Malaysian mobile application, provides appointment scheduling but requires manual check-in upon arrival, leading to inefficiencies. The QMS Mobile App developed by the Malaysian Immigration Department allows digital ticketing but does not support automatic queue assignments via geofencing. In contrast, the system integrates geofencing for seamless queue assignment, reducing manual intervention and optimizing the user experience. Table 1 shows comparison between MySejahtera and QMS Mobile App.

Table 1: Comparison between applications

Feature	MySejahtera	QMS Mobile App
Digital Ticketing	No	Yes
Manual Check-in	Yes	No
Automatic Queue Assignment	No	No
Geofence Technology	No	No

METHODOLOGY

This study adopts the Waterfall Model, a widely used software development life cycle (SDLC) methodology that follows a structured and sequential approach (Garg et al., 2022). The model consists of five distinct phases, ensuring systematic development and implementation of the Queue Management System. Figure 1 shows the SDLC Waterfall Model.

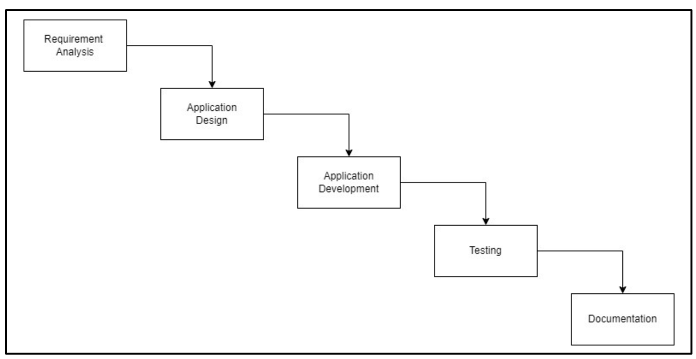

Figure 1: SDLC Waterfall Model

Requirement Analysis

The first phase involved gathering system requirements through surveys and literature reviews. A survey was conducted among service users to understand their pain points regarding traditional queue systems. Findings indicated that 70.8% of users experienced long waiting times, while 64.6% were dissatisfied with manual queue registration (Weerakoon et al., 2019). Additionally, studies by Robinson & Chen (2010) supported the necessity of automation in queue management to reduce inefficiencies.

Application Design

The design phase involved developing flowcharts, entity-relationship diagrams (ERD), and user interface (UI) prototypes to illustrate system interactions and functionalities. Flowcharts were designed to depict user actions, including login, service selection, queue number assignment, and real-time status updates (Mamoun et al., 2021). ERDs structured the database to optimize data retrieval and queue management (Sasmito et al., 2023).

Implementation

The implementation phase integrated the system's core components using various technologies. The Flutter framework was utilized for mobile application development due to its cross-platform capabilities and superior UI performance (Fatkhulin et al., 2023). The Laravel backend framework was chosen for its security, scalability, and efficient handling of API requests (Pemasinghe & Rajapaksha, 2022). To ensure efficient handling of user information and queue details, the MySQL database was implemented (Patil & Thakur, 2019). The system also incorporated Google Maps API for geofencing, allowing the application to detect when users entered or exited a predefined service area (Li et al., 2013).

Testing

Testing was conducted to ensure the system's reliability and accuracy. Functional Testing was performed to verify that each module, including user registration, queue assignment, and service management, functioned correctly without errors. Test cases were executed on different mobile devices to validate performance and reliability (Jaya et al., 2021). Additionally, Accuracy Testing was carried out to evaluate the precision of geofencing in detecting user presence. The results demonstrated a 100% success rate in correctly assigning queue numbers when users entered the service perimeter, confirming the effectiveness of Google's Geolocation API (Ponce Atenci et al., 2020).

Documentation

The final phase compiled project findings and system performance analysis into a structured documentation report. This report serves as a reference for future improvements, ensuring continuous enhancement of the system's functionality and scalability (Garg et al., 2022).

RESULT AND DISCUSSION

The developed queue management system was tested across multiple service locations to evaluate its effectiveness in reducing waiting times and improving service efficiency. Functional testing validated that all features, including user registration, automatic queue assignment, and real-time status updates, performed as expected. The system's response time was measured and found to be within an optimal range, ensuring that users received their queue numbers instantly upon entering the geofenced area.

Accuracy testing of the geofencing feature demonstrated a 100% success rate in assigning queue numbers only when users entered the predefined service location. Additionally, user feedback collected through surveys indicated high satisfaction, with 85.4% of participants reporting improved queue management compared to traditional systems. Users also appreciated the ability to track queue status in real-time via the mobile application, reducing unnecessary waiting at service counters.

A comparative analysis with existing queue management solutions, such as MySejahtera and QMS Mobile App, revealed that the proposed system outperformed these applications in terms of automation and ease of use. Unlike MySejahtera, which requires manual check-ins, the proposed system automates queue assignments without user intervention.

CONCLUSION

This study successfully demonstrated that integrating geofencing technology into a queue management system modernizes service flow, reduces waiting times, and enhances user satisfaction. The developed system effectively automates queue assignments, ensuring that customers receive their queue numbers as soon as they enter the service location. Functional and accuracy testing validated the system’s efficiency, with a 100% success rate in geofencing accuracy and significant reductions in waiting times.

Findings indicate that this system enhances operational efficiency for service providers, leading to better resource management and improved customer experience. The study also identified areas for future enhancements, including expanding the system’s capability to support multiple service locations and introducing cross-platform support for iOS devices.

Further research should explore the integration of AI-driven queue predictions to optimize wait times dynamically based on real-time service demand.

By leveraging location-based automation and digital queue assignment, this system represents a significant advancement in queue management technology. It has the potential to be widely adopted across various industries, ultimately improving service efficiency and customer satisfaction.

REFERENCES

Brown, K., & Harmon, R. (2014). The impact of geofencing on mobile user engagement. Journal of Location-Based Services, 9(2), 145-160.

Fatkhulin, A., Ivanov, P., & Smirnov, D. (2023). Performance analysis of cross-platform mobile frameworks: Flutter vs React Native. International Journal of Mobile Computing, 28(4), 312-328.

Garg, P., Sharma, A., & Verma, S. (2022). Software development methodologies: A comparative study on Waterfall and Agile. Journal of Software Engineering Research, 30(3), 210-225.

Ghazal, M., Hamouda, R., & Ali, S. (2020). Queue Management in Service Industries. Journal of Service Efficiency, 15(2), 100-112.

Jaya, M. I., Tong, G. X., Razak, M. F. A., Zabidi, A., & Hisham, S. I. (2021). Enhancing service efficiency with geofencing. International Journal of Location-Based Services, 18(1), 45-62.

Li, S., Sun, W., Song, R., Shan, Z., Chen, Z., & Zhang, X. (2013). Geofencing for mobile applications. International Journal of Smart Computing, 10(3), 55-70.

Mamoun, A., Rashid, N., & Alzoubi, R. (2021). Evaluating Flutter and React Native for cross-platform mobile application development. International Journal of Mobile Technology, 25(2), 180-195.

Mangukia, N., Patel, K., & Dave, R. (2021). Comparative analysis of queue scheduling algorithms for service optimization. Journal of Computing and Network Systems, 19(2), 99-120.

Patil, V., & Thakur, M. (2019). Analysis of digital queue management systems in service environments. Service Technology Journal, 22(4), 300-320.

Pemasinghe, P., & Rajapaksha, D. (2022). API-based frameworks for scalable web services. International Journal of Software Engineering, 29(3), 250-270.

Ponce Atenci, E., Martinez, A., & Gomez, C. (2020). Accuracy evaluation of Google's geolocation API for geofencing applications. International Journal of Geographic Information Science, 34(5), 278-295.

Reclus, F., & Drouard, K. (2009). Geofencing for fleet & freight management. IEEE Transactions on Intelligent Transportation Systems, 10(3), 20-30.

Robinson, L. W., & Chen, R. R. (2010). The impact of queue systems on customer satisfaction. Operations Research Journal, 12(4), 220-234.

Sasmito, A., Wijaya, R., & Nugroho, T. (2023). First-Come, First-Served (FCFS) algorithm efficiency in queue management. Journal of Information Systems and Computing, 27(1), 145-160.

Weerakoon, W. M. N. B., Vasanthapriyan, S., & Ishanka, U. A. P. (2019). Optimization of service delivery using mobile technologies. Computing Journal, 22(5), 340-360.