Hackathon Report
Queue Management System for DISCOM Offices

Team ID: TM000082

September 29, 2025

Abstract

Our smart queue management system improves fairness, efficiency, and customer satisfaction by using a strong mathematical model that accounts for walk-ins, appointments, and priority customers. It calculates and updates an Estimated Time to Response (ETR) using real-time service data. The system uses modern technology, offering digital tokens, live wait time countdowns, and advance appointments through a responsive web portal and on-site kiosks. Smart reshuffling handles delays fairly, with clear rules for late arrivals or priority cases. The customer experience is central, allowing users to track their status via website, kiosk, SMS, or email. For staff and administrators, the system provides efficient dashboards to manage queues and balance workloads, while managers can monitor the entire system, adjust configurations, and respond to emergencies. A feedback loop ensures continuous improvement, with future plans including AI for predicting busy times and making the system more adaptive.

Contents

1. Hackathon and Team Details

Project Overview	
Team ID	TM000082
Industry Name	Gujarat Urja Vikas Nigam (GUVNL)
Problem ID	PS000029
Problem Statement	Queue management system for DISCOM Offices
Team Leader	Kavathiya Tirth Hareshbhai (tirthkavathiya@gmail.com, 9558569072)
Faculty Advisor	Dr. Shankar Parmar (shankar.parmar@gtu.edu.in, 9824916667)

2. Team Members

Sr.	**Name**	**Contact**	**Role**
1	Kavathiya Tirth Hareshbhai	tirthkavathiya@gmail.com	Team Leader
2	Satyajeetsinh Mahida	satyajeetsinhmahida44@gmail.com	Team Member
3	Bhatiya Kavya Bhavin	kavyabhatiya3019@gmail.com	Team Member
4	Swapnil Patil	swapnilpatil2075@gmail.com	Team Member
5	PRAJAPATI ADITYA NANJIBHAI	adityaprajapati3047@gmail.com	Team Member
6	Purohit Rutva Akshaybhai	rutvapurohit14046@gmail.com	Team Member
7	Ahir Hevinkumar Keshavbhai	hevinahir179@gmail.com	Team Member

All members are 3rd Semester Computer Engineering students at Government Engineering College, Bharuch.

3. Problem Statement Analysis

3.1. Challenge Description

It is a regular scene at government and even private offices to find a large number of people standing in long queues waiting for their turns. This experience is often frustrating for customers and leads to inefficiencies, such as inquiries directed at non-relevant staff. This environment increases consumer dissatisfaction while simultaneously reducing staff efficiency. An effective queue management system must not only minimize customer waiting times but also empower staff to provide service more efficiently. Kiosk and mobile-based systems offer a modern solution to these limitations.

3.2. Target Users

The solution is designed for Government/Private Offices, hospitals, and any organization dealing with a high volume of customer interactions.

3.3. Expected Outcomes

- A proper system for appointment, allocation, and direction.
- Faster processing of customer requests.
- Reduced customer wait times and relief from long queues.
- Real-time performance monitoring for management.
- Time-saving for both companies and customers.

3.4. Potential Impact

- Improved workforce efficiency.
- Better management of consumers, leading to increased satisfaction.
- Improved productivity and operational efficiencies.

4. Proposed Solution: A Smart Queue Management System (QMS)

Our proposed Queue Management System (QMS) is a comprehensive solution that integrates a responsive web portal, mobile accessibility, and on-site kiosks to streamline customer interactions. The system is designed to efficiently handle walk-in, scheduled, priority, and overdue customers by leveraging digital token issuance, a dynamic Estimated Time to Response (ETR), AI-based query handling, and automated load balancing. Dedicated interfaces for managers, service providers, and customers ensure transparency and efficiency throughout the service process.

5. Technology Stack

Component	Technology
Frontend	HTML5, CSS3, JavaScript, Tailwind CSS (Potential for React.js)
Mobile App	Flutter / React Native
Backend	Flask (Python) with RESTful APIs
Database	MySQL (Structured Data), Redis (Cache & Session Management)
AI Module	Natural Language Processing (e.g., Dialogflow, Rasa) for chatbots
Authentication	JWT / OAuth 2.0 for secure, role-based access control
Real-time Notifications	Twilio SendGrid (Email/SMS), Firebase (Push Notifications)
Version Control & Deployment	Git, GitHub, GitHub Actions
Hosting & DevOps	AWS / Azure / GCP, Docker & Kubernetes
Data Analysis	Python data analysis libraries (e.g., Pandas, Matplotlib)

6. System Architecture and Flow

6.1. System Architecture

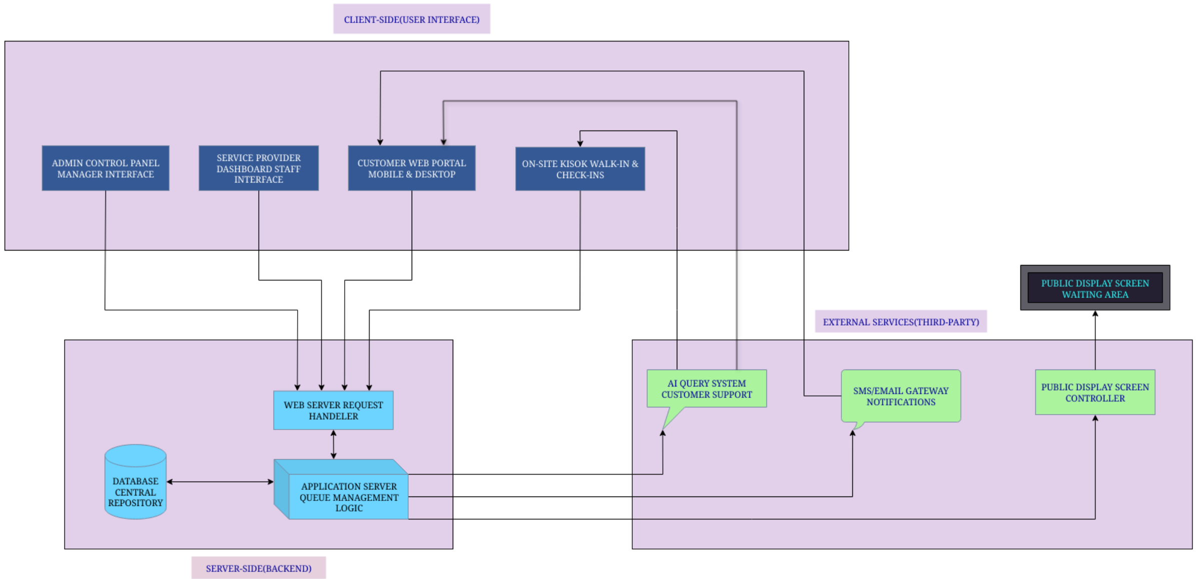

Figure 1: High-Level System Architecture.

6.2. User Flow

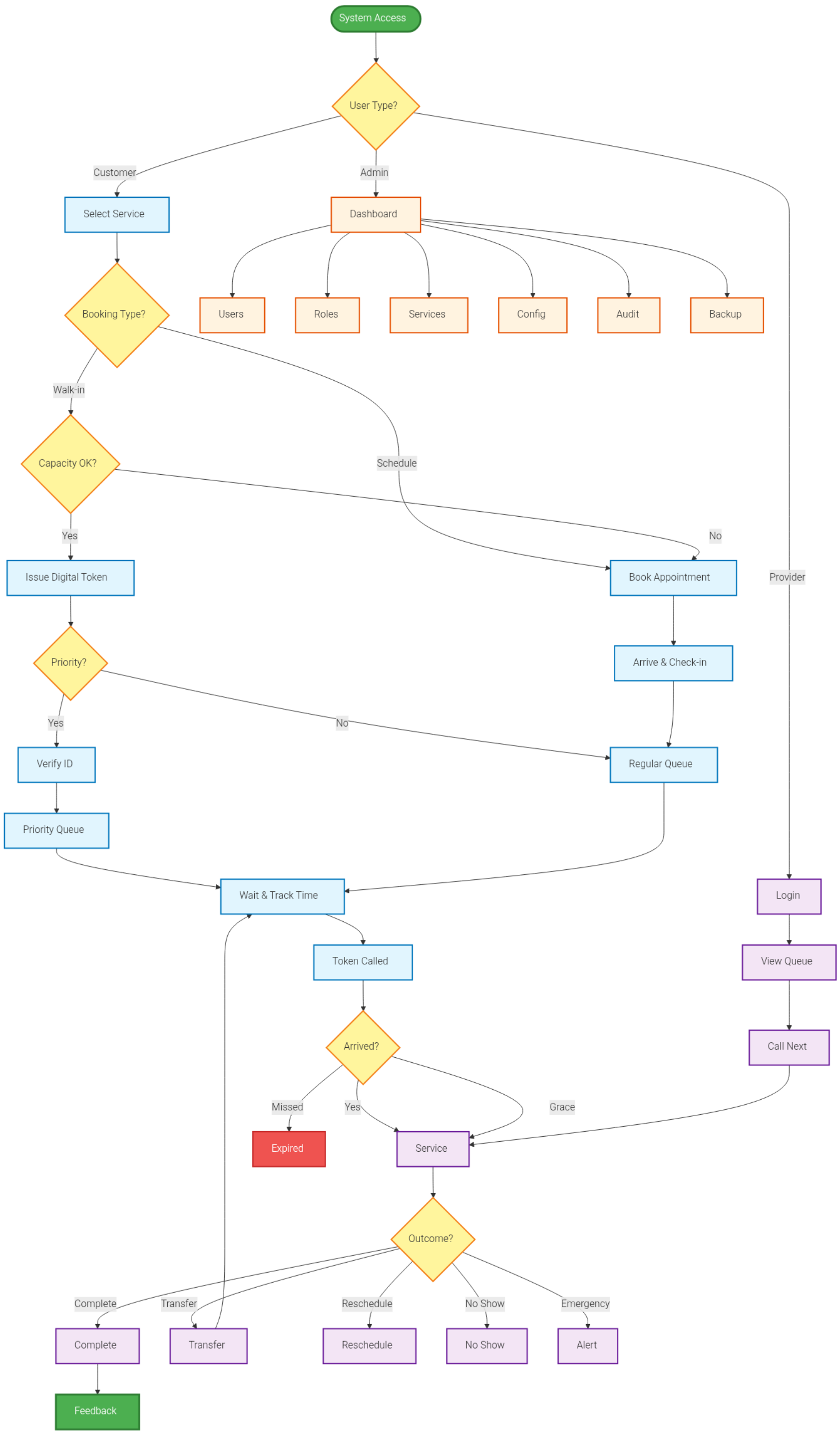

Figure 2: Comprehensive User Flow Diagram.

7. User Interface and User Experience (UI/UX)

7.1. Mobile and Kiosk Interface

The user interface is designed to be clean, intuitive, and responsive, ensuring a seamless experience across all devices.

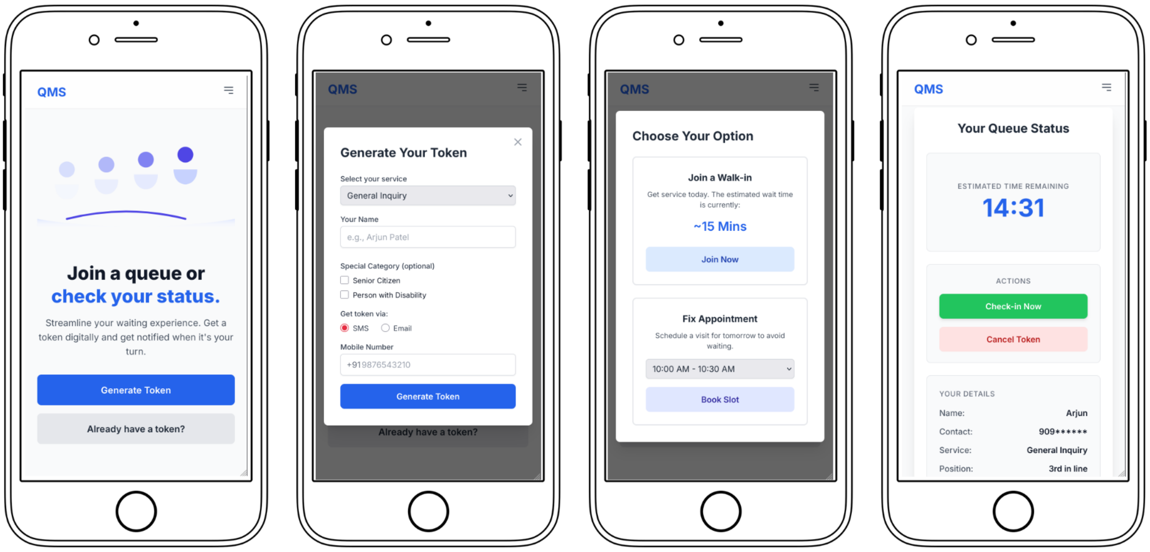

Figure 3: Customer-facing mobile interface.

7.2. Officer Dashboard

The officer dashboard provides service staff with all the necessary tools to manage the queue efficiently.

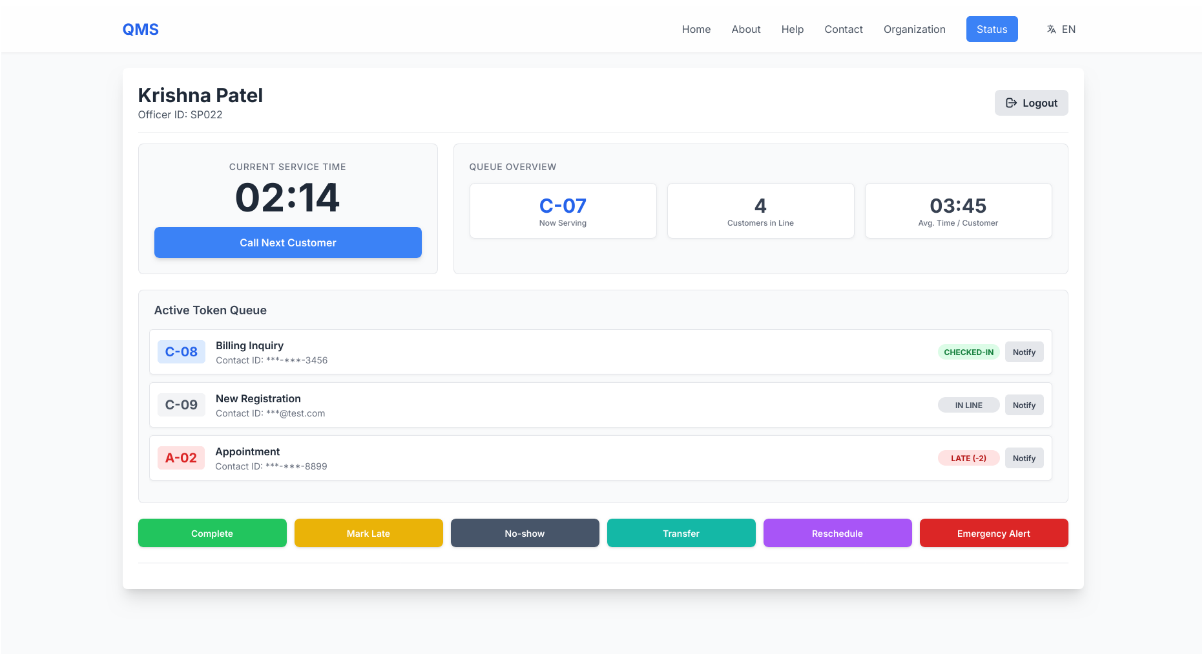

Figure 4: Dashboard for service providers/officers.

8. Core Features and Functionality

8.1. Introduction

Advanced queue management leverages technological innovation and mathematical models to handle customer queues more effectively. This document outlines the user interface for such a system, focusing specifically on the different user roles and the core features they interact with.

8.2. Accessing the Interface

Users can access the interface through a dedicated website, compatible with both mobile devices and personal computers, which connects to the company's central queue management server. An authentication system is utilized to differentiate between user types. Additionally, on-site kiosks will be provided for direct interaction within the office premises.

8.2.1. User Roles

The system defines several distinct user roles, each with specific permissions:

Manager/Admin Manages the system and has a complete overview of all operations.

Service Provider The staff member at the service desk who serves customers.

Waiting Customer A standard walk-in customer waiting for service.

Scheduled Customer A customer with a pre-booked appointment.

Priority Customer Individuals such as seniors or persons with disabilities (PWD) who are given precedence.

Overdue Customer A customer who has missed their scheduled appointment time.

8.3. Customer Interface Features

The customer-facing portion of the system is comprised of several key components designed to create a seamless and transparent experience.

Responsive Web Portal & Kiosk The interface is a responsive web portal accessible on any device (mobile, desktop) and via a self-service kiosk in the office. This provides high accessibility, supporting both remote and walk-in customers.

Mandatory Service Triage Customers must select the exact purpose of their visit (e.g., Billing, New Connection, Complaint). This ensures they are routed to the correct, specialized queue, preventing misdirection.

Contact-Based Identification The customer provides a mobile number or email for communication and identification. This generates a traceable transaction ID and establishes a channel for real-time alerts.

Digital Token Issuance A unique token number is instantly provided on-screen and delivered via SMS/Email, offering immediate confirmation and replaces the ambiguity of a physical line.

Flexible Booking Options Customers have the choice between joining the immediate queue or booking a specific future appointment slot. This empowers the customer with control over their time and helps avoid unnecessary travel.

Optimized Scheduling The system highlights the next available low-wait window for the current day. This encourages customers to visit during non-peak hours, reducing overall wait times and crowding.

Proactive 'Today is Impossible' Alert If wait times become excessive, the system advises against joining the immediate queue and prompts the user to book a future appointment. This prevents customer frustration and wasted trips to the office.

Dynamic Estimated Time Remaining (ETR) A visible, continuously updated countdown timer is displayed, calculated based on real-time officer service speeds. This reduces waiting anxiety by making the experience transparent and predictable.

Dedicated Token Status Page A personalized page shows the customer's current queue position, the assigned officer/desk, and the ETR. This provides full transparency and allows for productive waiting away from the physical queue.

Grace Period for Late Arrivals The system reserves a customer's slot for a short grace period after their token is called before marking them as a no-show. This policy is fair and accommodates minor, unavoidable delays.

Automatic Token Expiry Tokens are automatically expired if the customer misses their turn after the grace period has passed. This prevents "ghost customers" from holding up the line's progress.

Office/Desk Mapping Upon a token being called, the system provides visual guidance to the correct officer's desk. This streamlines the final step of the process and eliminates the need for customers to ask for directions.

Priority Customer Verification For Priority Customers (Seniors/PWD), the system will include a verification step using official IDs like Aadhaar or a PWD ID card to ensure proper allocation of priority status.

AI-Based Query System An integrated question-and-answer system, likely powered by AI, will be available to assist customers with common queries and provide information without needing human intervention.

8.4. On-Site Check-in Process

Upon arrival at the office, especially for scheduled appointments, customers must check in to activate their token in the service queue. This confirms their presence and makes them eligible to be called by a service provider. Several check-in methods are available:

QR Code Scan Customers can scan a QR code displayed at the office entrance or reception using their mobile device to instantly check in.

Kiosk Check-in The on-site self-service kiosk will feature a prominent "Check-in" button, allowing customers to find and confirm their appointment or token.

Manual Token Entry Using the web portal on their phone or a kiosk, customers can manually enter their token number and confirm their location within the office to check in.

Receptionist Assistance Customers can approach the receptionist, who can look up their details and manually check them into the system.

8.5. Public Display Screen

A large public-facing screen in the office waiting area provides real-time updates to all customers. The display typically shows a list of token numbers currently being served and the corresponding service desk or officer number they should proceed to. In the event of a significant queue restructuring by a manager, the screen will display a prominent message advising customers to check their mobile notifications for updated queue information.

8.6. Communication and Notifications

The system actively communicates with customers through SMS and/or email to keep them informed about their queue status. Proactive notifications are sent at key moments:

Significant ETR Changes An alert is sent if the Estimated Time Remaining (ETR) changes significantly, allowing customers to adjust their plans accordingly.

Grace Period Alert When a customer's turn is called, they are notified that their grace period has begun, informing them how long they have to report to the service desk.

'You Are Next' Notification An alert is sent when the customer's position in the queue becomes second, giving them advance notice to prepare.

Queue Transfer Notification If a customer is transferred to a new service queue, they receive an alert with their new queue details and token number.

Queue Restructuring Alert If a major queue change is made by management, a notification is sent to affected customers, prompting them to check for updated details.

Reschedule Confirmation When an appointment is rescheduled or a new token is issued, a confirmation message is sent with the updated details.

8.7. Service Provider Interface

The service provider interface is designed for efficiency, focus, and effective queue management from the officer's perspective.

Secure & Personalized Login Officers log in with a password and are presented with a personalized view showing only the queues and appointments assigned to their specific skill set. This eliminates distractions and ensures focus on relevant service requests.

Call Next Button & Service Timer A prominent button allows the officer to summon the next customer, which simultaneously triggers the customer's alert and starts a service timer. This ensures a smooth flow and captures real-time service duration data essential for calculating the dynamic ETR.

Active Token View A dedicated area of the screen displays all relevant details of the customer currently being served (service type, token number, contact ID), ensuring all necessary information is immediately available.

Automatic Load Balancing The system automatically distributes tokens for a specific service (e.g., "Billing") among all available, authorized officers. This prevents any single officer from being overwhelmed and speeds up the entire service line.

Check-In Integration Scheduled appointments appear in an officer's queue only after the customer has physically "checked in" at the office (via kiosk or mobile app). This prevents officers from wasting time waiting for no-show appointments.

Seamless Service Transfer A button allows the officer to instantly transfer a customer to a different queue or officer for a follow-up service (e.g., transfer to a cashier after an initial consultation). This manages complex customer journeys smoothly without forcing the customer to restart the process.

In-Service Rescheduling Officers have the ability to book a future follow-up appointment for the current customer directly from their interface, efficiently closing the current session while securing time for more complex issues.

Transaction Closure Options Clear buttons are provided to log the outcome of each interaction: Completed Successfully, Cancelled (with a mandatory reason), or Escalated/Pending Documents. This provides accurate data to the system and stops the service timer correctly.

Emergency Alert Button A dedicated button allows the officer to send a discrete alert or message to a manager for assistance with complex, disruptive, or emergency situations, enabling quick support without leaving the desk.

No-Show Management Officers can mark a token as a "No Show" if the customer does not appear after the grace period. This removes the token from the active queue, maintains flow, and allows the next person to be called.

8.8. Manager and Admin Interface

The admin interface provides high-level control over system configuration, user management, and data integrity.

User Creation and Management Admins have the ability to create, manage, and define access levels for all user accounts (Managers and Service Providers). This ensures only authorized personnel can access their respective system functions.

User Deactivation Admins can immediately suspend or delete any user account, blocking login access instantly. This is critical for security, especially in cases of staff changes or potential breaches.

Role and Permission Definition Admins can define and modify the specific actions each role can perform (e.g., only Managers can view reports). This provides granular control and prevents unauthorized actions.

Officer Skill Set Definition The interface allows for the definition and modification of skill sets linked to each officer role (e.g., authorizing a "Billing Officer" for "Billing Dispute" services). This allows the system to adapt to changes in service offerings and staff expertise.

Service Master Data Management Admins can add, edit, or archive the list of services available to customers. This ensures the customer-facing interface is always up-to-date with current offerings.

System Hour Configuration The ability to set and adjust official operating hours, defining when tokens can be issued and appointments scheduled. This sets operational boundaries and controls system availability.

System Audit Log A comprehensive, time-stamped record of all critical events, including logins, configuration changes, manual queue interventions, and user assignments. This is essential for accountability, security audits, and troubleshooting.

Data Backup and Recovery Admins can manage scheduled data backups and execute recovery procedures in the event of a system failure, ensuring business continuity and data protection.

8.9. Service Finalization and Feedback

Once a service is concluded, the system initiates final steps to close the loop with the customer and gather valuable feedback.

Service Completion Message Upon the service provider marking the transaction as complete, the customer receives a final SMS/Email confirming that their service has been successfully concluded.

Feedback System Shortly after the completion message, a link to a feedback survey is sent to the customer. This allows them to rate their experience and provide optional comments, contributing to service quality analysis.

8.10. Conclusion

The proposed queue management system provides a comprehensive, multi-faceted solution designed to streamline operations and enhance the user experience for all stakeholders. By integrating customer-centric features, efficient service provider tools, and powerful administrative controls, the system aims to reduce wait times, increase transparency, and provide management with actionable, This holistic approach transforms a traditional queuing process into a modern, efficient, and customer-friendly service interaction.

9. Mathematical Model

9.1. Introduction

A mathematical model of a queue management system is needed to implement a technological solution to solve day-to-day challenges in queue handling at offices. This document serves as a foundational mathematical framework, designed to simplify the implementation of a technological solution. By breaking down complex queueing dynamics into a set of defined rules, principles, and equations, this model provides a clear blueprint for developing an effective and intelligent queue management system.

9.2. The Complexity of Modern Queueing Systems

At its core, an office provides certain services to its customers. The key participants are the **Service Provider**, who delivers the service, and the **Customer**, who receives it. However, the complexity of managing the interaction between them grows significantly when we consider the different types of customers that a modern system must handle:

- **Waiting Customer:** A standard walk-in customer.
- **Scheduled Customer:** A customer with a pre-booked appointment.
- **Priority Customer:** Individuals such as seniors or persons with disabilities (PWD) who are given precedence.
- **Overdue Customer:** A customer who has missed their scheduled appointment time.

The simplest queueing model involves a single service provider with a single line of walk-in customers. Complexity can be layered on by adding multiple queues for the same service. A truly complex system, however, is one that must juggle appointments, priority customers, and latecomers simultaneously.

While this document provides rules to manage these interactions, these rules can fail because real-world scenarios are not perfect. In such cases, a purely mathematical model is not the right answer; we must use computational technologies to manage the system. For instance, consider a single desk handling multiple services with a mix of appointments, latecomers, and priority customers. This mayhem of variables creates chaos that can lead to system failure. While there should be an engineering fail-safe in the technological implementation, there cannot be a perfect mathematical fail-safe for every real-world contingency.

9.3. Single Queue Management

This section outlines the model for a system with a single queue and a single service provider. This is the foundational model upon which more complex scenarios will be built.

9.3.1. Service Time Metrics

To effectively manage the queue, it's crucial to track the time taken to serve each customer. This allows the system to build predictive metrics.

Average Service Time The average time per customer is calculated as a rolling average. This provides an up-to-date measure of the service provider's performance. It is defined as the sum of all previous service durations divided by the total number of customers served.

Let T_{avg} be the average service time and T_i be the service time for the i-th customer. For n customers served, the average service time is:

$$T_{avg} = \frac{\sum_{i=1}^{n} T_i}{n} \tag{1}$$

For the very first customer ($n = 1$), a predefined base service time, T_{base}, can be used as the initial average to start the calculation.

Service Time Deviation We can measure how much the service time for a particular customer deviates from the current average. This metric, which we'll denote as Δt, helps in identifying anomalies or trends in service duration. The deviation is calculated as:

$$\Delta t = T_{actual} - T_{avg} \tag{2}$$

where T_{actual} is the time taken for the current customer. This value is continuously tracked and can be used for effective resource management.

Root Mean Square (RMS) of Service Time The RMS is a statistical measure that gives a higher weight to larger values. In this context, it will be more sensitive to customers who take an unusually long time to serve. The RMS of the service times for n customers is defined as:

$$\text{RMS}_T = \sqrt{\frac{\sum_{i=1}^{n} T_i^2}{n}} \tag{3}$$

Where T_i is the service time for the i-th customer.

9.3.2. Queue Closure Condition

To avoid creating a queue that cannot be fully served within business hours, the system needs a rule to stop admitting new customers. The queue for walk-in customers will remain open only as long as the following condition holds true:

$$\frac{T_{rem}}{T_{avg}} > N_{queue} \tag{4}$$

Where:

- T_{rem} is the operational time remaining for the service provider.
- T_{avg} is the current average service time per customer.
- N_{queue} is the number of customers currently waiting in the queue.

This inequality ensures that the projected time to serve everyone currently in the queue is less than the time remaining in the workday. Once the number of customers in the queue is equal to or greater than the number of customers that can be served in the remaining time, no new walk-in customers are admitted.

9.3.3. Estimated Time to Response (ETR)

Providing an estimated wait time is a critical feature of a queue management system. The Estimated Time to Response (ETR) for the i-th customer in the queue can be calculated by considering the average service time and the actual performance of the service provider on preceding customers.

The ETR for the i-th customer is given by:

$$\text{ETR}_i = (T_{avg} \cdot (i-1)) + \sum_{j=1}^{i-1} \Delta t_j \tag{5}$$

Where:

- ETR_i is the estimated time until the i-th customer is served.
- T_{avg} is the average service time.
- i is the position of the customer in the queue.
- Δt_j is the Service Time Deviation for the j-th customer who has already been served.

This formula provides a dynamic estimate. The first term, $(T_{avg} \cdot (i-1))$, calculates a baseline ETR based purely on the average service time for the customers ahead. The second term, $\sum_{j=1}^{i-1} \Delta t_j$, adjusts this estimate by summing the total extra or less time taken for all customers who have already completed their service. This ensures the ETR adapts to the service provider's real-time performance.

9.3.4. System Stability and Dynamic ETR

The standard ETR calculation is effective under normal conditions. However, on days with high variability in service times (e.g., a busy Monday), the average service time, T_{avg}, may not fully capture the potential for large delays. To account for this, we can introduce a more stable, dynamic ETR that incorporates the Root Mean Square (RMS) of service times.

Weighted Dynamic ETR We can now define a dynamic ETR that blends the average service time with the RMS value using a weighting factor, w. This allows the system to switch between a standard estimate and a more conservative one.

$$\mathrm{ETR}_i^{\mathrm{dynamic}} = [w \cdot T_{avg} + (1-w) \cdot \mathrm{RMS}_T] \cdot (i-1) + \sum_{j=1}^{i-1} \Delta t_j \tag{6}$$

The weighting factor w is a value between 0 and 1.

- When w is close to 1, the ETR relies heavily on the average, suitable for stable, predictable days.
- When w is close to 0, the ETR gives more weight to the RMS value. This makes the estimate more conservative and accounts for greater volatility in service times.

The power of this model lies in its adaptability. By using data analytics, the system can learn to adjust the weight w based on historical data. For instance, it could automatically lower w on days that are historically busier (like Mondays) and raise it during typically quieter periods, leading to more accurate and reliable wait time predictions for customers.

9.3.5. Integrating Scheduled Appointments

Implementing a system for scheduled appointments is a strategic decision for the operating company. Should they choose to offer this feature, the model must be adapted to manage a hybrid queue of both walk-in customers and those with appointments.

Appointment Availability Condition The system must determine if new appointments can be offered. This is based on the projected service completion time for a hypothetical new walk-in customer relative to the closing time. The number of available appointment slots, S_{avail}, can be calculated as follows:

$$S_{avail} = \left\lfloor \frac{T_{close} - (T_{now} + ETR_{N_{queue}+1})}{t_{slot}} \right\rfloor \tag{7}$$

Where:

- T_{close} is the closing time of the business (e.g., 5:00 PM).
- T_{now} is the current time.
- $ETR_{N_{queue}+1}$ is the Estimated Time to Response for a hypothetical *new* walk-in customer who would join the end of the current queue.
- t_{slot} is the duration of a single appointment slot, determined by the company.
- $\lfloor \cdot \rfloor$ is the floor function, as only full slots can be offered.

The term $(T_{now} + ETR_{N_{queue}+1})$ represents the projected time of day when the last potential walk-in would begin service. The formula calculates the remaining time in the workday after this point and determines how many appointment slots can fit. New appointments can be offered only if $S_{avail} > 0$.

Scope of Availability It is important to note that this formula is intended for booking available slots on the **present day only**. It ensures that appointments are not booked beyond the closing time of the current business day. The process for scheduling appointments for future days is considered a separate function and is not governed by this real-time availability calculation.

Calculating Walk-in ETR with Appointments When the system must handle both walk-ins and appointments, a walk-in's Estimated Time of Response (ETR) must be calculated **iteratively** to be accurate. The wait time is repeatedly adjusted to account for the "chain reaction" of delays caused by scheduled appointments until the final ETR becomes stable. The core logic is to find a "stable" ETR where adding the delay from all preceding appointments no longer pushes the ETR past any new appointments.

Definitions:

$ETR_{\textbf{base}}$ The initial ETR for the walk-in, based only on the existing walk-in queue.

$t_{\textbf{slot}}$ The fixed duration of one appointment slot (e.g., 20 minutes).

$T_{\textbf{now}}$ The current time when the walk-in requests service.

$m_{\textbf{final}}$ The **final, stable count** of appointments between T_{now} and the final, projected service time ($T_{\text{now}} + ETR_{\text{final}}$).

Final State Formula: The process is iterative, but the final, correct ETR will aatisfy this equation:

$$ETR_{\text{final}} = ETR_{\text{base}} + m_{\text{final}} \cdot t_{\text{slot}} \tag{8}$$

Finding m_{final} requires an iterative algorithm where the ETR is calculated, the number of appointments within that new timeframe is counted, and the ETR is recalculated with the new appointment count until the number of appointments no longer increases.

9.3.6. Managing Overdue Customers (Latecomers)

A fair and flexible system must have a clear policy for customers who arrive after their expected service time. The management of a late customer is handled with a dynamic approach triggered by their check-in status.

Handling Walk-in Latecomers The process is divided into two phases.

Phase 1: Customer is Late (Not Checked In) When a customer's turn arrives but they have not checked in, they are marked as 'Late' and temporarily skipped. This immediately shortens the ETR for all subsequent customers, as the queue is effectively one person shorter.

Phase 2: Late Customer Arrives (Checks In) When the late customer physically checks in, their new position in the queue is determined by a configurable company policy, represented by a **Lateness Placement Factor** (W_L). This factor, a value between 0 and 1, allows a business to choose how strictly to handle latecomers.

- $W_L = 1$ **(Strict Policy):** The customer is sent to the very end of the line.
- $W_L = 0$ **(Lenient Policy):** The customer is placed at the front of the waiting queue (immediately after the person currently being served).
- $0 < W_L < 1$ **(Hybrid Policy):** The customer is placed proportionally along the length of the queue. A value of $W_L = 0.5$ would place them in the middle of the current line.

The new position, P_{new}, is calculated as follows:

$$P_{new} = \text{round}(1 + W_L \cdot N_{queue}) \tag{9}$$

Where N_{queue} is the number of customers currently in the queue. Once the late customer is re-inserted at this new position, the ETR for all subsequent customers is recalculated.

Grace Period for Walk-ins To prevent penalties for minor delays, a grace period is implemented. If a customer checks in within this window, they retain their original spot. The grace period is calculated dynamically:

$$T_{grace} = (T_{avg} \cdot w_{grace}) + T_{base\_grace} \tag{10}$$

Where T_{avg} is the average service time, w_{grace} is a weighting factor (< 1) adjusted via data analytics, and T_{base_grace} is a fixed base grace time.

Handling Appointment Latecomers Customers who miss their scheduled appointment are also granted a grace period, calculated with the same formula as for walk-ins. If they arrive after their grace period has expired, they are given two options to ensure fairness to punctual customers:

1. **Join the Walk-in Queue:** Forfeit their appointment and enter the queue as a new walk-in customer.
2. **Reschedule:** Book the next available future appointment slot.

9.3.7. Managing Priority Customers

This model handles priority customers by physically inserting them into the queue at a position determined by a pre-defined ratio, ensuring a fair and predictable wait for everyone.

Logic: Priority Insertion System A company can define multiple levels of priority (e.g., for seniors, persons with disabilities, expectant mothers), each with its own configurable ratio, R. This ratio dictates the guaranteed spacing of that priority type within the queue. For example, a ratio of $R = 4$ means there will be at least 3 regular customers between each customer of that priority type.

Insertion Algorithm When a priority customer arrives, their position in the queue is determined algorithmically. This ensures the correct spacing is maintained and the queue is reordered transparently.

Algorithm Steps:

1. **Identify Type and Ratio:** When a priority customer of type X with ratio R_X arrives, the system identifies their specific rules.
2. **Find Last Position:** The system scans the current queue from front to back to find the position of the last customer who is also of type X. Let this position be P_{last_X}.
3. **Calculate New Position:** The insertion position for the new customer, P_{new}, is calculated.
   - If a customer of type X was found, the new position is $P_{new} = P_{last\_X} + R_X$.
   - If no customer of type X was found in the queue, the new position is $P_{new} = R_X$.
4. **Insert and Reorder:** The new priority customer is inserted at position P_{new}. All subsequent customers are shifted back one spot. If the calculated position P_{new} is greater than the current length of the queue, the customer is simply placed at the end.

9.4. Multi-Queue Management

The model is extended to a system with multiple officers, each managing a separate queue but providing the identical service. This introduces the challenge of distributing customers and managing potential imbalances between the queues.

9.4.1. Customer Distribution

A multi-queue system requires a sophisticated method for assigning customers to a queue.

Initial Assignment: Round-Robin When customers first arrive, they are assigned to the available queues in a round-robin fashion to ensure an initially balanced distribution. However, this can lead to imbalances if one officer is significantly slower than others.

Dynamic Assignment: Lowest ETR Principle To overcome this limitation, a dynamic principle is introduced: **A new customer is always assigned to the queue with the lowest ETR for them.** The system calculates their potential ETR for each queue and directs them to the one providing the shortest estimated wait time. This ensures customers are routed to the most efficient service point at that moment.

9.4.2. Managing Different Customer Types

The principles for handling scheduled, priority, and overdue customers can be extended to the multi-queue environment through effective technological implementation. For example, a business can leverage technology to dedicate specific queues for specific functions (e.g., one for appointments, one for walk-ins) or apply priority logic globally. The core ETR and service time metrics for each individual queue remain the same. The key is the intelligent, system-level application of these rules.

9.4.3. Intelligent Reshuffling for Queue Balancing

Even with intelligent initial assignment, variations in service times can cause queues to become imbalanced. To counteract this, the system employs a continuous optimization algorithm to dynamically rebalance the queues. The goal is not simply to equalize queue lengths, but to minimize the **Total System ETR** (the sum of ETRs for all customers).

Algorithm Logic

1. Calculate the current Total System ETR.
2. For each customer in the system, simulate moving them to every other available empty slot (in their current queue or another queue).
3. For each simulated move, recalculate what the new Total System ETR would be.
4. If a move is found that results in a lower Total System ETR, make that move permanent.
5. Repeat this process until no single customer move can be found that further reduces the Total System ETR. The system is now in an optimized state.

This "smart shuffling" method ensures that customers are arranged in a way that is demonstrably more efficient for the group as a whole, providing a gentle and continuous rebalancing mechanism.

9.5. Conclusion

This document has detailed a comprehensive mathematical model for a modern queue management system, progressing from a foundational single queue to a complex, multi-queue environment. The model incorporates a dynamic ETR, configurable policies for diverse customer types (including appointments, priority, and latecomers), and intelligent algorithms for customer assignment and queue balancing. The principles and equations outlined provide a robust and adaptable framework that balances operational efficiency with customer fairness, serving as a solid foundation upon which a powerful and practical queue management technology can be built.

10. Literature Review and Existing Solutions

Existing queue management solutions range from enterprise-scale platforms with advanced AI and robotics (e.g., **Q'SOFT EQMS**, **QLess**) to simpler, kiosk-based systems (e.g., **Qminder**). Q'SOFT integrates AI-powered management with robotics, while QLess is a cloud-based pioneer with strong business intelligence tools. Other solutions like **Qmatic** focus on physical infrastructure for walk-in-heavy industries, and **Greetly** combines visitor and queue management. Academic research, such as the Massimo project (2024), explores using machine learning and computer vision to improve public service flow. Our solution builds upon these concepts by integrating a strong mathematical model for dynamic ETR calculation and intelligent queue reshuffling to optimize for the Total System ETR.

11. Project Status

11.1. Work Done to Date

The core functionality of the Queue Management System (QMS) has been successfully implemented. Users can obtain a digital token by providing personal and service details, with options for both walk-in and appointment queues. They can track their position, estimated wait time, and receive token details via email. Built with Flask, the system effectively utilizes Twilio SendGrid for email-based tokens and OTPs. Service staff can access a live list of waiting customers, including their details and required services. The web-responsive frontend ensures a consistent user experience across mobile phones, desktops, and kiosks.

11.2. Challenges Encountered

Key challenges included designing a consistent database schema for customers, tokens, and service providers, and integrating the Flask backend with MySQL. Ensuring secure login authentication and session management was critical to avoid security risks. Integrating email notifications via SMTP also presented formatting issues. Difficulties with Git version control, particularly with managing unstaged files, risked code loss. Finally, the complex integration of all modules meant that any single component failure could disrupt the entire system.

11.3. Expected Outcomes

- **Reduced Waiting Times:** 40-60% reduction in customer wait times.
- **Increased Staff Productivity:** 25-35% increase in efficiency.
- **Higher Customer Satisfaction:** 80-90% customer satisfaction rate.
- **Reduced Crowding:** 60-70% less crowding in waiting areas.

12. Future Scope

- Integration with biometric or Aadhaar-based authentication for priority customers.
- Advanced AI chatbots for handling customer queries.
- Predictive analytics for peak-hour management.
- Multi-language support for inclusivity.
- Integration with mobile wallets and UPI for seamless payments.
- Expansion to support a wider range of government offices, banks, and hospitals.