1) Synopsis Abstract

The answer to old queuing systems, excessive customer wait times, and jammed service halls is the Advanced Queue Management System (QMS). The latest, multi-dimensional platform is meant to take the place of lines and offer a smooth, clear service experience available through mobile, desktop, and kiosks. The primary goal of the QMS is to significantly reduce customer wait times, do away with congestion, increase operational transparency, and improve employee productivity.

The QMS's intelligence lies in its robust mathematical model that goes beyond basic first-in, first-out reasoning. It actively handles a rich blend of customer types such as Walk-ins, Scheduled appointments, and Priority guests and regularly computes an up-to-date Estimated Time of Resolution (ETR). The system employs a proprietary "smart shuffling" algorithm to proactively rebalance the entire service process across multiple queues. This guarantees the overall waiting time for all individuals in the system is kept at a minimum, resulting in clearly more efficient delivery of service.

The QMS offers clear, intuitive digital technology for all interested parties. Customers are provided with digital tokens and real-time queue updates by email or SMS (using Twilio SendGrid), so they can follow their position anywhere. Service Providers use an efficient, live dashboard for streamlined task management and customer finalization, eliminating manual paperwork. Administrators maintain full control over policy setting, staff assignment, and auditing all system activity. Built using Flask, the QMS currently features fully operational digital token issuance and live status updates. This technological approach is projected to yield high returns, including 80-90% customer satisfaction and up to 35% increased staff productivity.

2) Literature Review/Existing Innovation-technology to solve related to your problem

Present-day queue management systems (QMS) greatly improve customer experience and operational effectiveness by harmoniously coupling Queuing Theory with advanced technology. Some of the key innovations include Virtual Queuing, which enables customers to remotely join queues and be updated in real-time through their mobile phones, freeing them from the space constraints of in-person lines. Sophisticated QMS also supports appointment scheduling for pre-booked visits, significantly cutting wait times. In addition, these systems use AI and data analytics to forecast busy times with accuracy, allowing for intelligent staffing levels and offering precious information on customer flow. Cloud-based solutions allow for easy rollout across numerous sites, central management, and strong integration abilities with key platforms such as CRM, POS, and IoT. These improvements as a whole enable a quicker, smoother, and more customer-focused wait time experience, ultimately driving higher satisfaction, elevating employee productivity, and achieving increased customer loyalty.

3) What would be your solution to the problem

Our intelligent queue management system enhances equity, effectiveness, and customer satisfaction through a robust mathematical model that considers walk-ins, appointments, and

priority customers. It estimates and updates an Estimated Time to Response (ETR) based on data such as service time.

The system leverages cutting-edge technology by providing digital tokens, real-time wait time countdowns, and advance booking. Smart reshuffling manages delays equitably with transparent rules for tardiness or priority situations.

Customer experience is paramount. Members monitor their status online, kiosk, via SMS, or through email, and reschedule in case of too long waits, minimizing stress.

Administrators and staff gain from streamlined tools. Staff receive dashboards to manage queues, while the system redistributes workloads. Managers oversee the configuration, fine-tune hours, manage roles, and handle emergencies.

A feedback loop guarantees ongoing improvement. Rule adjustments are informed by customer feedback and system data. Future development involves AI for anticipating peak hours, making the system responsive and turning queues into seamless experiences.

6) Tools and technologies to be used to solve the problem

In constructing a sophisticated queue management system, we intend to employ a combination of contemporary tools and technologies both for the front end and back end. On the frontend, we will employ HTML, CSS, and Tailwind CSS to develop a clean, responsive design that is perfect for all devices. We also intend to employ an appropriate frontend framework that suits our project requirements. On the backend, we will utilize Python with Flask to manage the server-side tasks. We will build RESTful APIs to control features such as real-time notification and sending emails or SMS using Twilio services like SendGrid. For data storage, we will use MySQL as our primary database. To assist users more conveniently, we intend to include a chatbot with a Large Language Model (LLM) API. We will also incorporate multi-factor authentication to ensure the system is secure. For version control and deployment, we will utilize Git, GitHub, and GitHub Actions. We can also utilize Amazon Web Services (AWS) to deploy and scale the system. Finally, we will utilize Python data analysis packages to analyze the queue data and refine the system based on the insights gained.

7) Challenges/Risk in implementing your Final prototype

The development of the Queue Management System was faced with numerous challenges. Major challenges involved making a uniform database schema for customers, tokens, and service providers and connecting the Flask backend with MySQL, which required managing connection and authentication problems. Front-end integration of CSS and JavaScript with Flask templates proved challenging because of improper file paths. Secure login authentication and session management were essential to prevent security vulnerabilities. Email notification integration encountered SMTP and formatting problems. Git version control issues, specifically with resetting changes and handling unstaged files, threatened the loss of code. Lastly, the intricate deployment and integration of all modules (backend, database, frontend) led to any single component failure compromising the overall system. These issues highlighted the necessity of careful planning, testing, and error handling.

8) Potential output of your work

This queue management system has short-term and long-term advantages. It cuts waiting times by 40–60% and removes the anxiety of standing in line. Work is distributed evenly for staff to avoid burnout, and customers are given live feedback about their wait times.

Long term, the system generates insight for better decision-making, staffing optimized, and minimizing waiting spaces. It's adaptable and makes it easier for every customer to access. Success measures are 80–90% customer satisfaction, 25–35% higher staff productivity, and 60–70% fewer crowds. There should also be fewer complaints.

However, there are risks involved like difficulty for the non-tech-savvy users, system dependency issues, and complicated installation in need of good planning. Overall, this system computerizes queuing, enhancing customer satisfaction and business efficiency.

9) Work done till date

The Queue Management System (QMS) has been successfully deployed to manage customer queues electronically, which greatly enhances efficiency. Customers are now able to get a token by entering personal and service details, both for walk-in and appointment queues. They are able to monitor their position, estimated waiting time, and obtain token details through their email.

Developed with Flask, the system successfully leverages Twilio SendGrid for email tokens and OTPs. Service personnel have access to a live waiting customers list, including details and services needed, which has improved service delivery and task fulfillment.

The QMS has improved customer experience by minimizing waiting times and removing queues. It has also made staff management easier by eliminating manual calls and paperwork. Main features such as digital tokens, real-time updates of the queue, and email notifications are all in full working order.

The web-responsive frontend makes the user experience device-agnostic and accessible on mobile phones, desktops, and kiosks.