Project Idea

Title: Virtual Queue Management System

|  |  |  |
| --- | --- | --- |
| **#** | **Member Name** | **Role** |
| 1 | Sridhara Sai Kaushik | Front End Developer |
| 2 | Naga | Project Analyst |
| 3 | Harika | Back End Developer |
| 4 | Akshitha | Project Manager |
| 5 | Shivalee | Quality Assurance & Additional Support |

Table of Contents

[1. Introduction 3](#_Toc193009305)

[2. Problem Statement: 3](#_Toc193009306)

[3. Objective 4](#_Toc193009307)

[4. Technologies to be used 5](#_Toc193009308)

[Task Distribution Table 6](#_Toc193009309)

[Gantt Chart (3 Weeks) 8](#_Toc193009310)

[References 9](#_Toc193009311)

# Introduction

Waiting customers encounter inefficient queues frequently in service-based industries including healthcare facilities along with banking establishments and retail locations and public administration centers. Digital queues cause clients to become unhappy along with increased wait periods that result in reduced operational efficiency. A Virtual Queue Management System should be developed according to this project design to digitize and optimize queue operations.

A remote virtual queue-through the mobile application or web portal interface enables users to add themselves to virtual lines. When users join the virtual queue through the mobile application or web portal the system provides them with real-time information about the queue status and estimates waiting durations as well as alerts customers when they are ready to be served.

Through their admin dashboards both service providers (such as banks and hospitals) gain control over queue management combined with priority case handling capabilities and resource capacity optimization tools. This system serves to upgrade customer satisfaction while shortening waiting periods and optimizing procedure efficiency at service facilities.

# Problem Statement:

Waiting periods that extend for extended periods in hospitals and other public institutions such as banks and governmental offices create dissatisfied users and decrease operational effectiveness. The queue management systems used traditionally work through physical tokens and manual operations although these methods consistently produce erroneous delays.

A virtual queue management system must develop because it will enhance the queue process while decreasing waiting periods combined with offering users contemporary queue status information.

# Objective

**Create a web-based service for customers to reserve spots**

Construct a user-friendly online system using Django programming and React.js technology for users to join remote lines accessing healthcare, bank services or retail stores. Our system will make it possible for users to check in digitally without waiting in lines.

**Real-Time Queue Updates**

Connect users to real-time queue status through WebSocket or Django Channels to show how long they line up and when their position becomes available. Users will get real-time updates so they do not need to manually update their screen.

**Admin Dashboard for Queue Management**

Employ an admin dashboard built with React.js and the Django admin user interface so service providers can handle queues by placing crucial cases first plus inspecting queue metrics actively.

**Performance Analysis and Reporting**

Extract queue performance data from Django’s ORM while rendering it as charts through React.js using chart libraries like Chart.js or Recharts to help administrators track queue figures.

**Scalable and Customizable Design**

Construct the system with interchangeable modules that adjust seamlessly when switching between different industries including hospitals banks and government buildings.

**Enhance User Experience**

Build a smooth user experience system with React.js to provide desktop and mobile access to users. Set up the system to make ticket queue QR codes and send emails or SMS alerts through Twilio's basic plan or Django's built-in email service for free.

**Optimize Resource Allocation**

Give administrators ways to use their queue data to staff and run services more effectively which saves time and reaches better results.

**Promote Contactless Service Delivery**

Our system lets users join remote queues and get updates through electronic means to match how services are delivered now in pandemic recovery.

# Technologies to be used

**Backend:**

Django: A high-level Python web framework for building the backend.

Django REST Framework (DRF): For creating RESTful APIs.

Django Channels: For real-time communication using WebSocket.

PostgreSQL: A relational database for storing user, queue, and booking data.

**Frontend:**

React: A JavaScript library for building the user interface.

React Router: For navigation between pages.

Axios: For making HTTP requests to the backend.

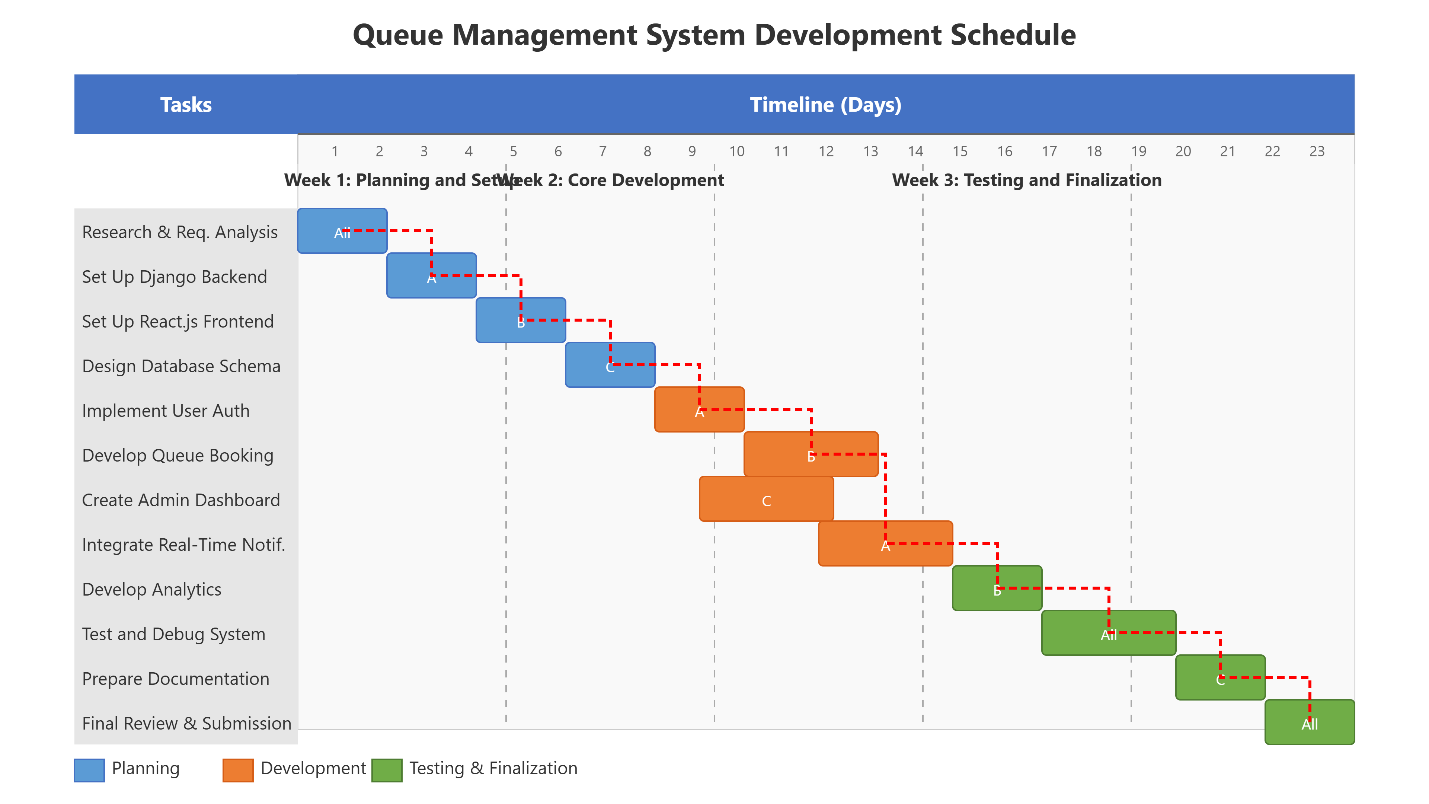
Socket.IO: For real-time updates and notifications.

Chart.js: For visualizing analytics data.

# Task Distribution Table

| **Task** | **Assigned To** | **Description** |
| --- | --- | --- |
| Project Idea | Sridhara Sai Kaushik | Defines the overall concept and purpose of the Virtual Queue Management System. |
| Introduction | Naga | Describes the need for the system and its impact on service-based industries. |
| Problem Statement | Harika | Identifies the inefficiencies in current queue management and the need for a virtual system. |
| Objective | Naga | Outlines the goals of the system, including real-time updates, admin dashboards, and performance tracking. |
| Project Management | Akshitha | Oversees project planning, scheduling, and communication across the team. |
| Frontend Development | Sridhara Sai Kaushik | Develops the React UI, integrates React Router, Axios, and WebSockets for real-time updates. |
| Backend Development | Harika | Implements Django, Django REST Framework, Django Channels, and PostgreSQL. |
| Admin Dashboard (Coding) | Sridhara Sai Kaushik | Builds the React.js-based dashboard for queue management. |
| Real-Time Queue Updates | Sridhara Sai Kaushik | Implements WebSockets (Django Channels) to provide live queue updates. |
| Performance Analysis (Coding) | Sridhara Sai Kaushik | Develops analytics visualization using Chart.js/Recharts. |
| Scalability & Customization | Sridhara Sai Kaushik | Ensures the system is adaptable for different industries (hospitals, banks, etc.). |
| Enhancing User Experience (Coding) | Sridhara Sai Kaushik | Implements UI/UX optimizations, QR code generation, and notification alerts. |
| Resource Optimization (Coding) | Sridhara Sai Kaushik | Develops tools for better queue and staff management based on real-time data. |
| Contactless Service Delivery (Coding) | Sridhara Sai Kaushik | Integrates electronic updates and notifications for users. |
| Technology Stack Documentation | Harika | Documents backend (Django, DRF, PostgreSQL), frontend (React, React Router, etc.), and real-time technologies (WebSockets, Django Channels). |
| Performance Analysis (Documentation) | Naga | Documents how queue efficiency can be tracked with analytics. |
| Enhancing User Experience (Documentation) | Harika | Describes UX/UI considerations and accessibility features. |
| Quality Assurance & Testing | Shivalee | Oversees testing phases, ensuring the system’s functionality, performance, and user experience are up to standard. |
| Gantt Chart (3 Weeks) | Sridhara Sai Kaushik | Plans project timeline and tasks in a structured format. |
| References & Citations | Harika | Compiles and formats references according to required standards. |

# Gantt Chart (3 Weeks)



# References

Saini,R. (2024, December 9). What is virtual queuing and how does it work? Vizitor. https://www.vizitorapp.com/blog/what-is-virtual-queuing-and-how-does-it-work/

WaveTec. (n.d.). What is virtual queuing in customer service? WaveTec Blog. https://www.wavetec.com/blog/what-is-virtual-queuing-in-customer-service/