

# Software Requirements Specification (SRS)

**Project:** Real-Time Queue & Appointment Optimizer System

## 1. Introduction

This document describes the functional and non-functional requirements of the Real-Time Queue & Appointment Optimizer System. It explains what the system is expected to do and how it should perform under different conditions. The purpose of this document is to provide a clear understanding of system behavior for developers, testers, and evaluators. It also serves as a reference for future improvements and maintenance of the system.

## 2. Overall Description

The Real-Time Queue & Appointment Optimizer System is designed to automate and manage queues in service-based environments such as banks, hospitals, and government offices. The system helps in organizing customer flow by generating queue tokens, assigning customers to available service counters, and providing real-time updates on queue status.

The system aims to reduce manual effort, improve service efficiency, and enhance customer satisfaction. It operates as a web-based application that allows customers and staff to view live queue information. The backend system controls queue movement and counter availability automatically, ensuring smooth and coordinated service operations.

## 3. Functional Requirements

The functional requirements describe what the system should do:

- The system shall allow customers to join a queue digitally.
- The system shall generate a unique token for each customer.
- The system shall display real-time queue status to users.
- The system shall assign customers automatically to available service counters.
- The system shall allow staff to update counter status (free or busy).
- The system shall store customer and service details in a database.
- The system shall provide basic administrative functions such as monitoring queue performance.

These functions ensure that the queue is managed automatically and efficiently.



## 4. Non-Functional Requirements

The non-functional requirements define how the system should perform:

### **Performance:**

The system should update queue status in real time with minimal delay.

### **Security:**

The system should restrict unauthorized access to administrative features and protect stored data.

### **Usability:**

The system interface should be simple and easy to understand for both customers and staff.

### **Reliability:**

The system should continue to function correctly even during peak usage and recover from minor failures.

These requirements ensure that the system is efficient, safe, and user-friendly.

## 5. System Interfaces

### **UserInterface:**

The system provides a web-based interface for customers, staff, and administrators. Customers can view queue status, staff can manage counters, and administrators can monitor system activity.

### **HardwareInterface:**

The system runs on standard computing devices such as desktop computers, laptops, and tablets. No special hardware is required.

### **SoftwareInterface:**

The system interacts with a database to store and retrieve queue information. It also uses web technologies and real-time communication protocols to exchange data between clients and the server.