**Software Requirements Specification (SRS)**

**Smart Queue Management System (SQMS)**

**1. Introduction**

* **Purpose:**  
  To streamline patient check-ins, doctor assignment, service tracking, and reporting in a clinical or hospital environment.
* **Scope:**  
  This system handles:
  + Patient registration/check-in
  + Automatic or manual doctor assignment
  + Real-time dashboard for admin
  + Logs of actions (check-ins, serves, etc.)
  + Export functionality (PDF/Excel/CSV)
  + Admin and doctor authentication
  + Optional simulation module for testing

**2. Functional Requirements**

* Patient can:
  + Check in using a form (name, phone, category)
  + View assigned doctor and receipt
* Admin can:
  + View queue dashboard (served, waiting, by doctor)
  + Filter/search patients
  + Serve patients
  + View system logs
  + Start/Stop simulation
  + Export logs (PDF, Excel, CSV)
  + Logout securely
* Doctor can:
  + Log in to a dedicated dashboard
  + View and serve only their assigned patients
  + See real-time queue updates
* Simulation:
  + Simulates patient check-ins and service using realistic wait times
  + Auto-assigns doctors if available
  + Automatically serves patients when session ends

**3. Non-Functional Requirements**

* Web-based using Flask (Python)
* Backend: MySQL
* UI: HTML/CSS + Jinja + Font Awesome
* Responsive design with dark mode
* Authentication for admin and doctors
* Real-time updates via auto-refresh
* Modular codebase for maintainability

**4. Assumptions and Constraints**

* Assumes admin knows patient categories
* Assumes doctors are pre-registered
* Simulation is optional and controllable via UI
* Data stored securely in MySQL
* Basic input validation applied on forms

**5. System Models**

* **Actors:** Admin, Doctor, Patient, System (auto simulation)
* **Entities:** Patient, Queue, Doctor, Receipt, Log
* **Relationships:**
  + Patients → Receipts, Queues
  + Doctors → Queues
  + Logs track actions involving Patients and Queues