Addis Ababa University

College of Natural and Computational Sciences

Department of Computer Science

Digital Queue Management System for Tikur Anbessa Hospital

Software Requirements Specification

Team Members

Name	ID NO.
1. Adonias Abiyot	UGR/0796/15
2. Eyob Assayie	UGR/1219/16
3. Lawgaw sefineh	UGR/8107/17
4. Samuel kinfe	UGR/2027/16
5. Sawel yohannes	UGR/2969/16
6. Tsegaye shewamare	UGR/2048/16

ADVISOR: Aderaw Semma

Acknowledgement

We would like to express our sincere gratitude to our project advisor, **Aderaw Semma**, for his continuous guidance, constructive feedback, and encouragement throughout the development of this Software Requirements Specification (SRS) document.

Our heartfelt appreciation also goes to the management and staff of **Tikur Anbessa Specialized Hospital**, located at coordinates **8.9817° N, 38.7578° E**, Addis Ababa, Ethiopia, for allowing us to visit their premises and providing valuable insights into the hospital's queuing challenges and daily operations.

We are particularly thankful to **Mr. Shibre Kassa** (Phone: **+251 912 013 497**) for facilitating our initial meeting at the hospital and sharing practical information that helped us better understand the workflow of patient registration and service delivery.

We would also like to extend our gratitude to our instructors and the **Department of Computer Science** for providing us with the knowledge, technical foundation, and resources essential for the completion of this work.

Finally, our deepest appreciation goes to our **families and friends** for their patience, motivation, and unwavering support during the entire project preparation and documentation process.

Revision History

Date	Description	Author	Comments
Oct 25, 2025	Version 1	Adonias Abiyot, Eyob Assayie, Lawgaw sefineh, Samuel kinfe, Sawel Yohannes, Tsegaye shewamare	Initial version of the Software Requirements Specification

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
	Mr, Haile	ICT office Manager	10/20/2025

Contents

List of Tables

List of figures

Definitions, Acronyms, and Abbreviations

Term / Acronym	Definition / Description
SRS	Software Requirements Specification — a document that describes the functions, performance, and constraints of a software system.
QMS	Queue Management System — the software designed to manage patient queues within Tikur Anbessa Hospital.
EMR	Electronic Medical Record — a digital version of patients' paper charts, containing their medical and treatment histories.
HIS	Hospital Information System — an integrated system that manages hospital administrative, financial, and clinical data.
HL7	Health Level Seven — a set of international standards for the exchange of clinical and administrative data between healthcare systems.
FHIR	Fast Healthcare Interoperability Resources — a standard developed by HL7 for exchanging healthcare information electronically.
WAN	Wide Area Network — a telecommunications network that extends over a large geographical area for computer networking.
API	Application Programming Interface — a set of definitions and protocols that allows different software systems to communicate with each other.
WCAG	Web Content Accessibility Guidelines — a set of guidelines developed to make web content more accessible to people with disabilities.
ICT	Information and Communication Technology — technologies that provide access to information through telecommunications.
RBAC	Role-Based Access Control — a security approach that restricts system access to authorized users based on their roles.
HTTPS	Hypertext Transfer Protocol Secure — a secure version of HTTP for secure communication over a computer network.
TLS	Transport Layer Security — a cryptographic protocol designed to provide secure communication over a computer network.
JSON	JavaScript Object Notation — a lightweight data-interchange format used for data exchange between systems.
CSV	Comma-Separated Values — a file format used to store tabular data, such as spreadsheets or databases.
MTBF	Mean Time Between Failures — a reliability metric that measures the average time between system failures.
EMRS	Electronic Medical Records Standard — a standard that defines interoperability and data structure for EMR systems.
UI/GUI	User Interface / Graphical User Interface — the part of the system through which users interact with the software.
CLI	Command Line Interface — a text-based interface used to interact with software or operating systems.

Table 1 -Acronyms

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included. We have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Group 1 Date**:** 10-25-2025

1. Introduction

This Software Requirements Specification (SRS) document provides a comprehensive and detailed description of the **Digital Queue Management System (QMS)** to be implemented at **Black Lion Hospital, Addis Ababa**. It serves as the foundational reference for defining the functional and non-functional requirements of the system, ensuring that all technical and managerial stakeholders possess a shared understanding of the system's objectives, scope, and operational behavior.

The document is intended to guide software engineers, system analysts, and project managers throughout the software development life cycle. It articulates the system's requirements in a manner that facilitates accurate design, implementation, verification, and maintenance. Moreover, it ensures that the developed system aligns with the strategic goals of Black Lion Hospital and adheres to established digital health interoperability standards.

1.1 Purpose

The primary purpose of this document is to define and specify all necessary requirements for the **Digital Queue Management System (QMS)**. The SRS establishes a structured foundation upon which the software will be designed, developed, tested, and maintained. It delineates the functional capabilities, performance expectations, and interface specifications necessary for the effective implementation of the system.

The intended audience for this document includes:

- **ICT Staff:** Responsible for the technical development, integration, and maintenance of the QMS within the hospital's existing digital infrastructure.
- **Hospital Administrators:** Accountable for understanding system capabilities, overseeing implementation, and ensuring the system supports hospital operational efficiency and patient service delivery.
- **Project Stakeholders:** Involved in evaluating system performance, approving requirements, and ensuring alignment with institutional objectives and compliance standards.

This document aims to facilitate effective communication between technical personnel and organizational stakeholders, ensuring that all requirements are clearly understood, traceable, and implementable.

1.2 Scope

The software product to be developed is the **Digital Queue Management System (QMS)**, a **web-based application** designed to optimize patient flow management and reduce waiting times within major service departments, including outpatient consultation units and hospital laboratories.

The QMS will perform the following key functions:

- Enable patients to securely request and manage queue entries for doctor consultations and laboratory services through a web interface.
- Allow medical professionals, including doctors and laboratory technicians, to manage patient queues, mark their availability, and update service progress in real time.
- Provide patients with real-time visibility into queue positions, estimated waiting times, and SMS notifications when their turn approaches or when laboratory results are available.
- Incorporate accessibility features such as bilingual (Amharic and English) interfaces and voice-based announcements to support visually impaired patients.
- Integrate seamlessly with the **Ewket Electronic Medical Record (EMR)** and **Hospital Information System (HIS)** using **HL7 FHIR–compliant APIs**, ensuring data consistency, interoperability, and adherence to both national and international digital health standards.
- Allow administrative users to manage accounts, configure service settings, and generate operational and performance reports.

The system will not:

- Manage billing, prescription generation, or clinical diagnosis functions.
- Replace the Ewket EMR as the hospital's central data repository; instead, it will complement the existing system by focusing solely on queue and workflow management.

Goals and Expected Benefits:

- To enhance patient experience by providing transparent and efficient service delivery.
- To reduce service delays and congestion in hospital departments through automated queue handling.
- To improve coordination among staff members and departments by providing real-time operational insights.
- To ensure compliance with the **Ethiopian National Digital Health Blueprint** and the **WHO Digital Health Interoperability Framework**.

1.3 Overview

The remainder of this Software Requirements Specification (SRS) document is organized as follows:

- **Section 2: General Description** Provides an overview of the system's background, operational environment, major functions, user characteristics, and design constraints.
- **Section 3: Specific Requirements** — Enumerates the detailed functional and non-functional requirements that define system behavior and performance.
- **Section 4: Change Management Process** — Outlines the formal procedures for updating the SRS document in response to changes in project scope or system requirements.
- **Appendices** — Contain references, supporting documentation, and supplementary materials relevant to the QMS.

This structure ensures logical coherence, traceability, and ease of reference for all readers involved in the design, development, and evaluation of the Digital Queue Management System.

2. General Description

This section provides a comprehensive overview of the Digital Queue Management System (QMS) for Black Lion Hospital, Addis Ababa. It explains the system's purpose, operational context, major functions, user types, constraints, and assumptions. This overview establishes the foundation for the specific functional and non-functional requirements defined in later sections.

2.1 Product Perspective

The Digital Queue Management System (QMS) is a web-based software solution integrated with Ewket, the hospital's Electronic Medical Record (EMR) and Hospital Information System (HIS).

The system is designed to enhance service coordination and reduce patient waiting times by managing queues across key hospital departments such as laboratories and doctor consultations.

Unlike automated queue systems, this QMS allows patients to manually request queue entries through a secure web interface. Patients can view their queue status, cancel requests, or receive SMS notifications about their turn or laboratory results.

Doctors, lab technicians, and administrators use the system to manage patient flow, mark availability, and ensure efficient service delivery.

The system integrates with Ewket EMR through HL7 FHIR–compliant APIs, ensuring interoperability, data consistency, and alignment with both the Ethiopian National Digital Health Blueprint and the WHO Digital Health Interoperability Framework.

It operates on the hospital's Wide Area Network (WAN), providing reliable access across departments and service points.

The system also includes voice-based announcements and screen display features to assist visually impaired patients and improve inclusiveness in patient communication.

2.2 Product Functions

The QMS provides the following key functions, derived from the system's use-case model:

1. User Authentication and Role Management

All users (patients, doctors, lab technicians, and administrators) must log in or sign up securely.

The system uses role-based access control to assign permissions and manage data visibility.

2. Queue Request Management (Patient)

Patients can manually request to join a queue for services such as doctor consultation or laboratory processing.

Patients may also cancel their queue request if they decide to withdraw or reschedule.

3. Queue Status and Notification (Patient)

Patients can view the live status of their queue, including position and estimated waiting time.

The system sends SMS notifications when their turn is near or when laboratory results are ready.

A voice mode feature is available for visually impaired patients.

4. Doctor Interaction

Doctors can mark their availability (“In Session” or “Not Available”).

Doctors can call the next patient from their department’s queue, triggering visual and audio announcements.

5. Laboratory Management

Lab technicians can view and update laboratory test results.

When results are updated, the system automatically sends an SMS notification to the patient indicating that results are ready for review.

6. Administration and System Control

Administrators can manage system users (create, modify, or deactivate accounts).

They can monitor queue activity, configure service types, and generate operational reports.

7. Accessibility and Inclusiveness

Voice-based announcements guide patients, particularly those with sight impairments.

The interface supports Amharic and English, with compliance to accessibility standards (WCAG).

8. Interoperability

The system exchanges data with Ewket EMR using HL7 FHIR resources such as Patient, Appointment, Observation, and Encounter for structured interoperability.

2.3 User Characteristics

The QMS will serve four primary user groups:

Patients

Can log in to request or cancel a queue, view queue status, and receive SMS or voice notifications.

Some users may rely on voice output due to visual impairments.

Skill level: Minimal; requires simple, intuitive web interface available in Amharic and English.

Doctors

Manage consultation queues by marking availability and calling the next patient.

Skill level: High; accustomed to using EMR systems and digital tools.

Laboratory Technicians

Manage laboratory queues and update test results that trigger patient notifications.

Skill level: Moderate to high; familiar with Ewket's laboratory module.

Administrators

Oversee system operation, manage users, configure settings, and generate performance reports.

Skill level: Moderate to high; responsible for maintaining data quality and security.

The system design emphasizes ease of use, multilingual support, and universal accessibility to accommodate users with varying technical proficiency and abilities.

2.4 General Constraints

1. Regulatory and Policy Compliance

The system must conform to the Ethiopian National Digital Health Blueprint, WHO Digital Health Interoperability Standards, and HL7 FHIR requirements.

It must comply with national data protection and patient confidentiality policies.

2. Infrastructure Constraints

The system operates on the hospital's existing Wide Area Network (WAN) and hardware.

It must be optimized for reliable performance across multiple connected buildings and departments.

3. Integration Constraints

Requires secure and stable API connections to Ewket EMR for data exchange.

Synchronization between QMS and Ewket must be accurate and bi-directional.

4. Accessibility Requirements

Must include voice announcements and be compatible with screen readers.

The interface must be accessible in both Amharic and English.

5. Performance and Reliability

Must handle concurrent access from multiple departments with minimal delay.

SMS and voice notifications must be delivered promptly and reliably.

6. Security and Privacy

All communications and stored data must be encrypted.

Role-based authentication and authorization must be enforced.

Sensitive medical data may not be transmitted via SMS.

7. Scalability and Maintainability

The system must support expansion to additional departments or hospital branches connected via WAN.

Maintenance and updates should not disrupt daily hospital operations.

8. Operational Constraints

The system must tolerate short-term network interruptions (temporary caching and resynchronization).

Integration should not interfere with existing Ewket workflows.

2.5 Assumptions and Dependencies

- The Ewket EMR/HIS remains the primary patient data repository.
- The hospital's Wide Area Network (WAN) provides reliable connectivity between departments.
- Existing servers, computers, and network equipment meet system requirements.
- An SMS gateway and audio system are available and integrated.
- Hospital staff are trained in using Ewket and related systems.
- Updates to HL7 FHIR specifications or national interoperability standards may require software adjustments.
- Effective operation depends on collaboration between departments and adherence to queue management protocols.

3. Specific Requirements

3.0.1 Inputs

The system shall receive input of the patient's card number and check its validity. There shall be a feature that displays errors in clear and precise terminology.

3.0.1 Ordering

The system shall give a proper queue number or ID that is based on the order at which the patients registered for the queue. There should be a feature that clearly shows the patient or their caretakers the queue number or ID they will get the service in order.

3.0.3 Alerts

The system shall alert the patients (caretakers) when there are a few other patients in front and continue doing so every couple times until their turn arrives.

3.0.4 Miss cases

The system automatically removes the patient shall they fail to avail themselves for the service they registered in queue for after a wait time of one minute or any time the specific doctor or clerk chooses.

3.0.5 Displaying

In addition to displaying the system in the users' devices it shall also be displayed on TV screens found on the lobbies of each floors of the hospital.

3.1 External Interface Requirements

3.1.1 User Interfaces

A Graphic User Interface will be used to interact with the users and administrators. The GUI will draw inspiration from applications and software that are largely in use by Ethiopians. It will be flexible enough to work on smart phones and computers and TV screens.

3.1.2 Hardware Interfaces

For the administrators, this system shall run on computers independent of their operating system using their internet browsers. The users will use smart phones and again the operating system doesn't matter. TV screen will be used to display queues. These TV screen should have HDMI ports where they can connect to a computer.

3.1.3 Software Interface

The system will run on a browser. The specific browser in use is Google Chrome (usually called chrome). The version is Chrome 129.x.

The system will use a postges database provider called Supabase. The version number for the software in use is Supabase CLI: 1.195.7.

3.1.4 Communication Interface

For our web interactions we will use the standard https protocol.

3.2 Functional requirements

3.2.1 Registering in queue

3.2.1.1 Introduction

This is when the patients come to register for queue. There will several processes ranging from validating patient card numbers to assigning queue numbers.

3.2.1.2 Inputs

The patient (caretaker) will need the patient's card number to register.

3. 2.1.3 Processing

The system shall check the validity of the patient's card number and if it is valid should assign a queue number based on the entry time of the patient.

3.2.1.4 Output

The system shall send alerts to patient's or caretaker's devices in addition to displaying them on the central screen available in the lobbies. The alerts shall be sent when the patients turn comes and also when there are few people left in front of a patient so the patients will be more likely to avail themselves.

3.2.1.5 Error Handling

A clear error message shall be displayed if the user enters an invalid patient card number.

3.2.2 Automatic queue member removal

3.2.2.1 Introduction

Sometimes patients fail to come in time for the service they registered in queue for. There is a adjustable wait time of one minute for such cases. This feature is required since there are cases where the patient doesn't arrive even after the wait time. Therefore, they will be removed from the queue.

3.2.2.2 Inputs

The patients' card number used to process the removal.

3.2.2.3 Processing

The entity for the patient queue will be deleted.

3.2.2.4 Outputs

A message will be displayed informing the patient that they missed their turn and urge them to register again if they still want the service.

3.3 Use Cases

3.3.1 Use Case #1: Doctor Availability and Patient Calling

Primary Actor: Doctor

Goal: Notify the system when the doctor is available and call the next patient.

Preconditions:

- The doctor is logged into the system.
- The queue for the respective department is initialized.

Trigger:

The doctor marks "Available" in the system interface.

Main Success Scenario:

1. The doctor logs into the system.

2. The doctor marks their status as “Available.”

3. The system updates the queue and identifies the next patient.

4. The system sends an SMS notification to the next patient and updates the public display screen.

5. The patient proceeds to the consultation room.

6. After finishing with one patient, the doctor selects “Next” to call the following patient.

Postconditions:

• The current patient’s status is updated to “Served.”

• The next patient’s status is changed to “In Progress.”

Alternative Flows:

• If no patients are available in the queue, the system displays “No patient in queue.”

• If the doctor becomes unavailable, they set the status to “Unavailable,” and the queue halts automatically.

3.3.2 Use Case #2: Laboratory Test Completion Notification

Primary Actor: Laboratory Staff

Goal: Notify the system that a patient’s laboratory tests are complete.

Preconditions:

• The patient’s test record exists in the system.

• The laboratory staff is authenticated.

Main Success Scenario:

1. The laboratory staff completes the patient’s test.

2. The staff updates the patient’s test status to “Completed.”

3. The system verifies whether all required tests for that patient are completed.

4. Once all tests are completed, the system automatically sends an SMS to the patient indicating readiness to see the doctor.

5. The patient is reinserted into the appropriate doctor’s queue.

Postconditions:

- The patient's test status is updated to "All Tests Completed."
- The patient can queue for doctor consultation.

3.3.3 Use Case #3: Queue Display and Notification System

Primary Actor: Display System (Automated Process)

Goal: Display real-time patient queue and doctor availability on hospital screens.

Preconditions:

- Queue and doctor availability data are updated in the system.

Main Success Scenario:

1. The system retrieves the latest queue data from the database.
2. The system displays department names, doctors' names, and the current and next patients.
3. The display updates in real-time when a doctor calls the next patient.
4. Visual and/or audio alerts notify waiting patients.

Postconditions:

- The public display shows accurate queue and doctor information in real-time.

3.3.4 Use Case #4: Patient Mobile Interface

Primary Actor: Patient

Goal: View real-time queue position, doctor information, and notifications on a mobile device.

Preconditions:

- The patient is registered and logged into the system.

Main Success Scenario:

1. The patient logs into their mobile account.
2. The system displays the current queue number, department, and expected waiting time.
3. The system updates the patient's status in real-time.
4. When called, the system displays a "Please proceed to doctor's room" message.
5. If the users changes mind he/she can discard the status.

Postconditions:

- The patient is notified promptly when their turn arrives.

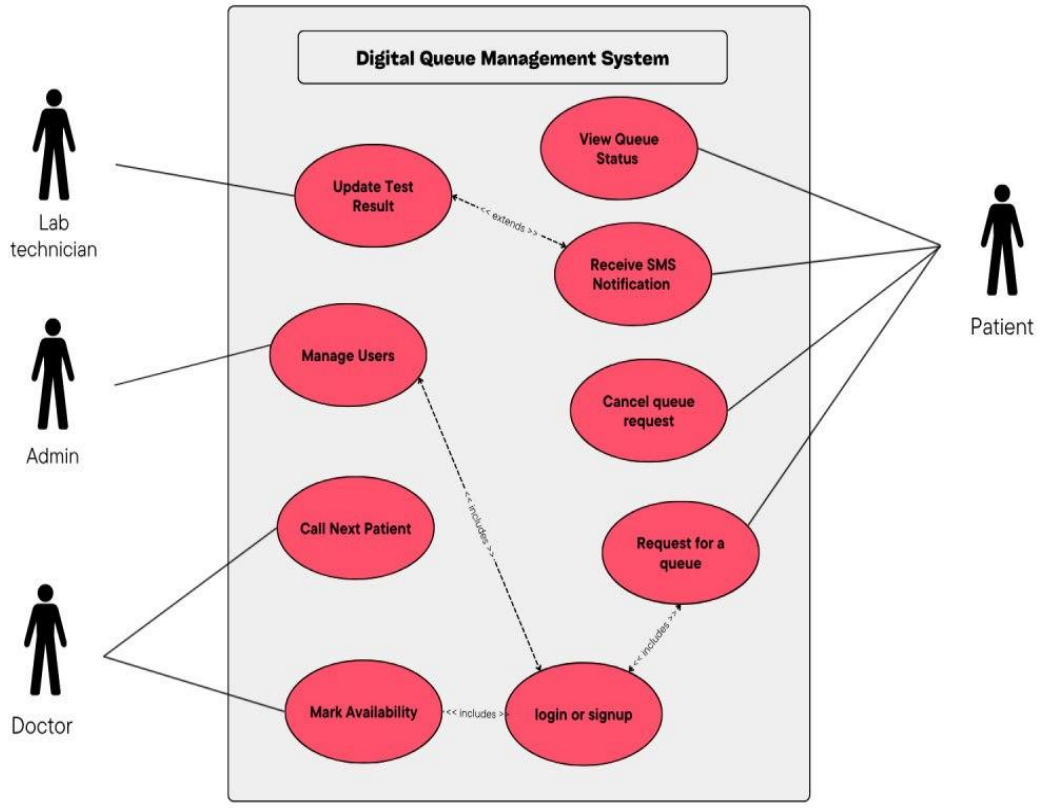

Figure 1 – use case

3.4 Non-Functional Requirements

This section defines the quality attributes that the Digital Queue Management System must meet. These requirements are defined in measurable terms and conform to the Ethiopian Electronic Medical Record (EMR) Standards and national digital health guidelines, ensuring reliability, security, and interoperability for hospital environments.

3.4.1 Performance Requirements

- The system shall process queue allocation, cancellation, and updates within 3 seconds under normal operating conditions.
- The system shall support at least 200 concurrent users without performance degradation.
- For patient verification and queue number generation, the system shall complete the process within 2 seconds.
- Notifications (SMS or on-screen) shall be sent or displayed within 5 seconds of a status change.
- Database queries and dashboard updates shall execute with 95% of transactions completed in less than 1 second.

3.4.2 Reliability Requirements

- The system shall maintain an operational uptime of at least 99% during hospital working hours.
- The system shall automatically recover from transient failures within 5 minutes.
- Daily data backups shall be performed automatically, ensuring no more than 24 hours of data loss in case of system failure.
- The Mean Time between Failures (MTBF) shall be at least 30 days.
- The system shall log all transactions and critical operations for traceability and post-incident analysis.

3.4.3 Availability Requirements

- The system shall be available 24 hours a day, 7 days a week for hospital operations.
- Scheduled maintenance periods shall not exceed 2 hours per month, performed during off-peak hours.
- Unplanned downtime shall not exceed 1 minute per day on average.
- The system shall include redundancy mechanisms to ensure service continuity in case of server failure.

3.4.4 Security Requirements

- All data transmissions shall be encrypted using HTTPS (TLS 1.2 or higher).
- User access shall be protected by role-based access control (RBAC) to ensure that only authorized roles (Admin, Doctor, Lab Technician, Patient) can perform specific functions.
- User passwords shall be encrypted using secure hashing algorithms (e.g., bcrypt or SHA-256) before storage.
- Automatic logout shall occur after 10 minutes of user inactivity.
- The system shall maintain comprehensive audit logs capturing login attempts, updates, and deletions.
- The system shall comply with health data confidentiality, privacy, and integrity standards outlined in national EMR policies.
- The system shall be interoperable with other EMR systems using standard data exchange protocols (e.g., HL7 or FHIR) to enable future integration.

3.4.5 Maintainability Requirements

- The system architecture shall follow a modular and layered design to support easy maintenance and component updates.
- Source code shall be documented with inline comments and developer guides.
- Critical issues (security or performance-related) shall be resolved within 48 hours of identification.
- The system shall support version control (e.g., Git) to manage code updates and deployment safely.

- Future enhancements, such as integrating appointment scheduling or EMR linkage, shall not require major redesign.

3.4.6 Portability Requirements

- The system shall operate on major server environments, including Windows Server and Linux (Ubuntu).
- The user interface shall be responsive and compatible with all major browsers (Google Chrome, Mozilla Firefox, and Microsoft Edge) and mobile devices.
- Data migration to another hosting platform (e.g., cloud) shall be completed within two business days.
- The system shall use standard data formats (JSON, CSV) and API endpoints to facilitate integration with national EMR or hospital systems.
- Installation and configuration shall not require more than one hour by a trained IT administrator.

3.5 Inverse Requirements

• The system shall not provide medical diagnoses or treatment recommendations.

• The system shall not prioritize patients based on medical urgency (this is handled by hospital staff).

• The system shall not manage billing or insurance operations.

• The system shall not store patient medical history beyond queue and test completion data.

3.6 Design Constraints

• The system shall comply with Tikur Anbessa Hospital's IT and security policies.

• The system shall comply with Electro medical records standard (EMRS).

• The system shall function effectively under low-bandwidth conditions (<2 Mbps).

• Development must utilize open-source technologies such as Node.js, MySQL, and React to minimize licensing costs.

3.7 Logical Database Requirements

• The system shall use a relational database (MySQL or PostgreSQL).
• The database shall include tables for:
  ◦ Patients: Patient ID, name, contact, department, queue number, status.
  ◦ Doctors: Doctor ID, name, department, availability status.
  ◦ Laboratories: Lab ID, test type, completion status.
  ◦ Queue Records: Queue number, timestamps, doctor assignment, current status.
• Each patient shall have a unique queue ID for tracking.
• Data retention shall be maintained for a minimum of 5 years.
• Automatic database backups shall be performed daily.
• Referential integrity shall be enforced through foreign key constraints.

3.8 Other Requirements

3.8.1 Training-Related Requirements
- All medical staff and system administrators shall undergo a 2-hour training session on system usage.
- A user manual and quick-start guide shall be provided in both English and Amharic.

3.8.2 Packaging Requirements
- The system shall be distributed as:
  - a web-based application accessible through the hospital's intranet.
  - an installation package containing all dependencies, README, and configuration files.

3.8.3 Legal Requirements
- The system shall comply with Ethiopian data protection and privacy laws.
- All open-source components used shall conform to their respective license agreements.
- Copyright © 2025 Tikur Anbessa Hospital Digital Queue Management System.

4. Change Management Process

The process for updating the Software Requirements Specification (SRS) ensures that all modifications to project scope, requirements, or system behavior are systematically reviewed, approved, and documented. This maintains alignment with project objectives, technical feasibility, and stakeholder expectations.

Change Submission:
Any project team member (developer, tester, or system analyst) can propose a change to the SRS. Change requests are submitted through formal documentation updates, clearly describing the reason for the change, its potential impact, and the affected components or requirements.

Review and Approval:
All submitted changes are reviewed by the **ICT Staff Manager**, who is responsible for evaluating the technical and operational impact of the proposed modification.

- **Major changes** (e.g., new system features, integration adjustments, or interoperability updates) require formal approval from the ICT Staff Manager before implementation.
- **Minor updates** (e.g., documentation corrections or small UI adjustments) may be approved informally after brief internal review by the technical team.

Version Control and Documentation:
Manual document versioning will be maintained to track all SRS updates. Each approved change will result in an incremented version number, with a summary of modifications recorded in the document's revision history section. Superseded versions will be archived for reference.

Notification and Implementation:
Upon approval, the technical/project team will be notified of the change for implementation and verification. In cases of **major updates**, all key stakeholders — including system administrators

and hospital management will also be informed to ensure proper coordination and communication across departments.

References

1] HL7 International, FHIR Release 5 (Fast Healthcare Interoperability Resources), 2024. [Online]. Available: https://hl7.org/fhir/

[2] Ministry of Health, Ethiopia, National Health Care Data Standard of Ethiopia, Addis Ababa, 2023.

[3] EMR Standards Committee, Electronic Medical Record (EMR) Standard for Health Facilities in Ethiopia, Federal Ministry of Health, Addis Ababa, 2022.

A. Appendices

A. 1 Appendix1 - Sample Questionnaire and Interview Questions

A. For Patients

1. How long do you usually wait before getting medical service?
2. What problems do you face while waiting (e.g., overcrowding, confusion)?
3. Would you use a digital queue system if available?
4. What would make such a system easy to use for you?
5. Do you prefer receiving updates via SMS or screen displays?

B. For Hospital Staff (Doctors, Technicians, and Admins)

1. How do you currently manage and organize queues in your department?
2. What difficulties do you face in managing patient flow?
3. How could a digital queue system help your daily work?
4. What type of information should the system show (e.g., patient name, queue order)?
5. What security or privacy measures should be considered?

A. 2 Appendix 2 - Sample System Flow Diagram and Description

System Flow Description:

1. **Patient Registration:** Patients register using their card number through an online portal or on-site.
2. **Queue Number Generation:** The system validates the entry and issues a queue number stored in the SQL database.
3. **Queue Monitoring:** Real-time status is displayed on screens and accessible via the web interface.
4. **Admin Control:** Administrators monitor queues, assign patients to doctors, and manage missed cases.
5. **Notifications:** The system sends SMS and on-screen alerts when a patient's turn approaches.

6. **Service Completion:** Once served, the queue record updates automatically for reporting.

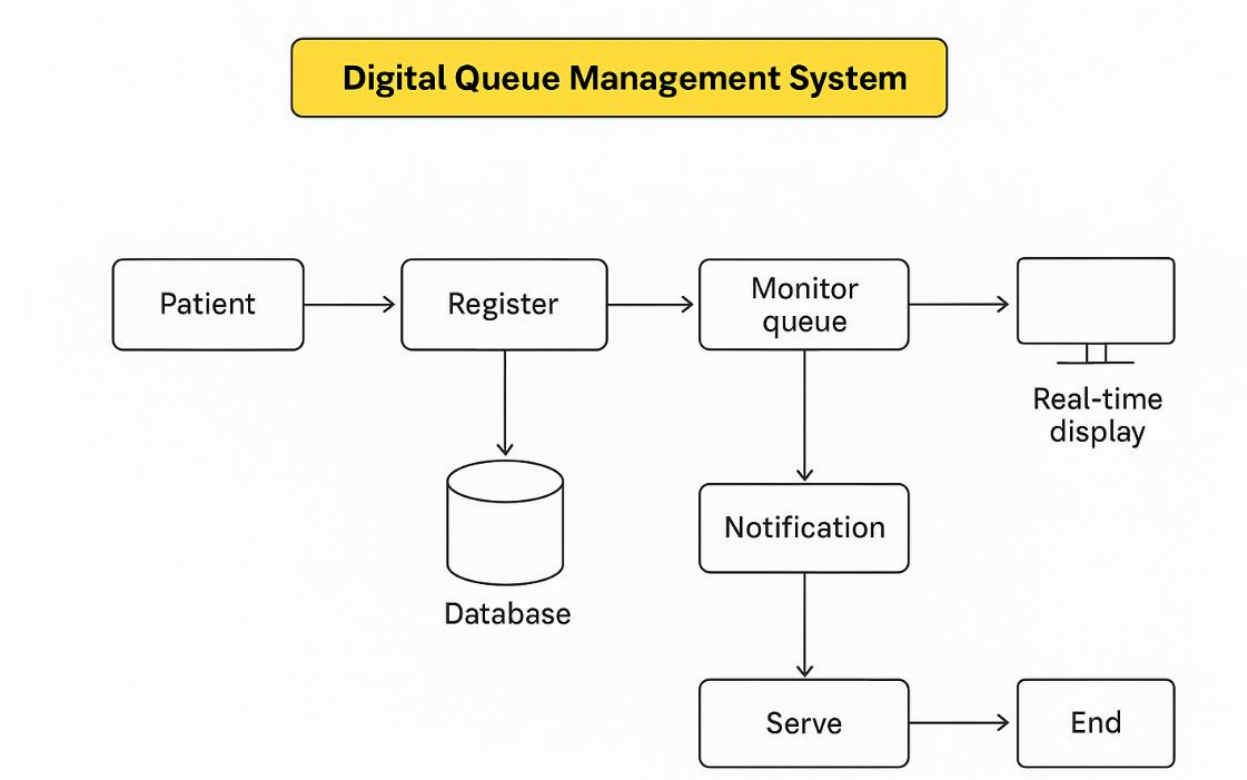

Figure 2 – system flow