**A Web-based Application to Coordinate Operations in a Clinic**

KIHUMBA KEVIN RENANA

168377

AIRA SAMSON OLOGI

166456

ICS 3 Group: B

Supervisor Name: PROF VINCENT OMWENGA

An Informatics Project Proposal Submitted to the School of Computing and Engineering

Sciences (SCES) in partial fulfilment of the requirements for the award of a Degree in Informatics and Computer Sciences

School of Computing and Engineering Science

Strathmore University

Nairobi, Kenya

22 May 2025

# Declaration and Approval

We declare that this work has not been previously submitted and approved for the award of a degree by this or any other University. To the best of our knowledge and belief, the research proposal contains no material previously published or written by another person except where due reference is made in the research proposal itself.

Student Name: KIHUMBA KEVIN RENANA

Admission Number: 168377

Student Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student Name: AIRA SAMSON OLOGI

Admission Number: 166456

Student Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Supervisor Name: PROF VINCENT OMWENGA

Supervisor Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Abstract**

This study proposes the design and development of a web-based application to coordinate operations in Kenyan outpatient clinics. By coordinating clinical workflow and integrating real-time doctor availability tracking, dynamic priority tagging for vulnerable and emergency cases and automated buffer slot insertion, the system aims to reduce overlapping appointments and long waiting times that currently plague level 2-5 facilities. The project follows an Object-oriented Analysis and Design paradigm implemented through an incremental development model resulting in a prototype featuring patient profile management, SMS based reminders and notifications and an administrative dashboard. The application’s backend and frontend integrate with an SMS gateway to ensure timely patient communication and seamless clinician schedule updates.

Evaluation assesses the system’s impact on average waiting times, appointments adherence and staff utilization through simulated clinic data sets. By addressing unpredictable practitioner availability, schedule disruptions caused by emergencies and the absence of automated triage and buffering in existing e-health platforms, this project aims to deliver a unified interface that enhances operational efficiency, improves patient satisfaction especially for the elderly and chronically ill and provides policymakers with actionable service-delivery metrics. The expected outcome is a solution that can be adapted across Kenya’s public health network to improve outpatient throughput, reduce financial and emotional burdens on patient and reduce staff workload.

Key Words: Web-based application, Outpatient clinics, Clinical workflow, Real-time doctor availability tracking, Dynamic priority tagging, Overlapping appointments, long waiting times, Object-Oriented Analysis and Design (OOAD), Prototype, Patient profile management, administrative dashboard, SMS gateway, Staff utilization, Simulated clinic data, Practitioner, Emergency disruptions, Automated triage, Operational efficient, Patient satisfaction, Policymakers.

# Table of Contents

Declaration and Approval .......................................................................................................... ii

Abstract .................................................................................................................................... iii

List of Abbreviations ............................................................................................................... vii

Chapter 1: Introduction ........................................................................................................ 1

1.1 Background Information ............................................................................................ 1

1.2 Problem Statement ..................................................................................................... 2

1.3 Aim/ Specific Objectives ........................................................................................... 3

1.3.1 Specific Objectives ................................................................................................... 3

1.3.2 Research Questions ................................................................................................... 4

1.4 Justification ................................................................................................................ 4

1.5 Scope .......................................................................................................................... 4

1.6 Limitations ................................................................................................................. 4

1.7 Delimitation ............................................................................................................... 4

Chapter 2: Literature Review............................................................................................... 5

2.1 Introduction ................................................................................................................ 5

2.2 Description of the Current Process of Coordinating Operations in a Clinic setup .......... 5

2.2.1 Appointment Assignment Process ............................................................................ 5

2.2.2 Tracking Real-time Clinician Availability ................................................................ 5

2.2.3 Queue and Triage Management ................................................................................ 5

2.2.4 Patient Communication and Notifications ................................................................ 6

2.3 Challenges in the Current Processes of Coordinating Operations in a Clinic Setup . 6 2.4 Systems used in Coordinating Operations in a Clinic Setup ......................................... 6

2.4.1 Premier Hospital EQMS ........................................................................................... 6

2.4.2 Luma Health ............................................................................................................. 7

2.4.3 Qmatic Orchestra ...................................................................................................... 8

2.5 Gaps in Related Works ............................................................................................... 8

2.6 Technology to be used ............................................................................................... 8

2.7 Conceptual Framework .............................................................................................. 9

Chapter 3: Methodology .................................................................................................... 10

3.1 Introduction .............................................................................................................. 10

3.2 Applied Development Approach.............................................................................. 10

3.2.1 Requirements &Planning (Increment 0) ................................................................. 10

3.2.2 Design & Development (Increment 1 & 2) ............................................................ 11

3.2.3 Integration & Validation (Increment 3) .................................................................. 11

3.2.4 Refinement (Increment 4) ....................................................................................... 11

3.3 System Analysis ....................................................................................................... 11

3.3.1 Use Case Diagram .................................................................................................. 11

3.3.2 Sequence Diagram .................................................................................................. 12

3.3.3 Entity-Relationship Diagram (ERD) ...................................................................... 12

3.3.4 Class Diagram ......................................................................................................... 12

3.3.5 Activity Diagram ............................................................................................. 12

3.3.6 State Diagram ......................................................................................................... 12

3.4 System Design ......................................................................................................... 12

3.4.1 Database Schema .................................................................................................... 12

3.4.2 Wireframes .............................................................................................................. 12

3.4.3 System Architecture ................................................................................................ 12

3.5 System Development Tools and Techniques ............................................................ 13

3.5.1 Integrated Development Environment .................................................................... 13

3.5.2 Backend Language & Framework ................................................................... 13

3.5.3 Frontend Technologies ............................................................................................ 13

3.5.4 Database Management System ............................................................................... 13

3.5.5 Collaboration & Version Control ............................................................................ 13

3.6 Deliverables ............................................................................................................. 13

3.6.1 Requirements & Design Documentation ................................................................ 13

3.6.2 Working System Modules ....................................................................................... 13

3.6.3 Test Artifacts ........................................................................................................... 14

3.6.4 System Demonstration and User Manual ........................................................ 14

3.6.5 Final Report ............................................................................................................ 14

References................................................................................................................................ 15 Appendix.................................................................................................................................. 18

Appendix 1: Gantt Chart ...................................................................................................... 18

**List of Abbreviations**

CIHEB – Clinton Health Access Initiative

CSS – Cascading Style Sheets

EQMS – Electronic Queue-Management System

ERD – Entity-Relationship Diagram

FCFS – First-Come, First-Serve

IOM – Institute of Medicine

MoH – Ministry of Health (Republic of Kenya)

OOAD – Object-Oriented Analysis and Design

QMS – Queue-Management System

UAT – User Acceptance Testing

# Chapter 1: Introduction

## 1.1 Background Information

Kenya’s health system is organized into six levels, from community health services (level 1) up to national referral hospitals (level 6), with most outpatient care delivered at dispensaries, clinics, health centres (levels 2–3), and sub-county or county hospitals (levels 4–5) (Republic of Kenya MoH, 2014). Despite this structured framework, clinics routinely struggle with inefficient scheduling, unpredictable practitioner availability, and long waiting times. The combination of these factors means planned appointments overlap and delays compound as the day progresses (Njoroge et al., 2021). Without an efficient system to manage overlaps and communicate changes, clinics find it difficult to maintain an orderly patient flow. By focusing on coordinating operations within outpatient clinics, this study addresses a core operational challenge in Kenya’s public health sector.

There are several factors that contribute to this situation, with the first one being unpredictable doctor availability. A doctor may fail to attend an appointment due to unforeseen circumstances or a doctor’s strike, leaving patients stranded and forced to return without any assistance (AP News, 2024). A study at the University of Nairobi staff clinic found that the average outpatient waiting time was around 55 minutes; 52% of respondents cited “improving staff availability” as the key way to reduce waiting time (Wafula & Ayah, 2021). Second, the digital systems lack the flexibility to prioritize patients such as the chronically ill and the elderly or manage overlaps through built-in buffers (Wanyee et al., 2019). Third, emergency cases disrupt the schedule of the day, forcing other appointments to start late. In Kenya, Emergency Departments often receive patients with life-threatening conditions. However, most Kenyan EDs operate without a standardized triage protocol, leading to inconsistent assessment and prioritization of patients (Wachira & Martin, 2011). Without a proper triage system, critically ill patients may be left waiting in the queue, increasing their risk of mortality.

These scheduling inefficiencies don’t exist in isolation — they have ripple effects across the entire healthcare ecosystem. Vulnerable patients such as the elderly and children are most at risk. In Homa Bay and Kisumu Counties, antenatal clinics reported missed appointment rates of 42% and 35% in 2019; of those, 78% in Homa Bay and 70% in Kisumu cited long waiting times as the primary reason for defaulting on scheduled appointments (Opon et al., 2020). Families must deal with the extra financial and emotional burdens as they travel to and from clinics. According to the Kenya Household Health Expenditure and Utilization Survey (2018), individuals incurred an average of Ksh 1,200 per outpatient visit, a significant expense for many low-income households (Ministry of Health, Kenya, 2019). Caregivers of children with tuberculosis in Kenya faced average household costs of USD 120 for non-medical needs, with half reporting catastrophic expenditures just to access free services (Barasa et al., 2017). Healthcare workers also face higher workloads as they attempt to manage overcrowded waiting rooms and make on-the-spot triage decisions. In Western Kenya, 61% of maternal care providers identified high workload as their top stressor (Namusonge et al., 2022). Failure to solve these problems leads to a chain of other issues that affect the entire health system. Patients who miss their scheduled appointments risk delayed detection of complications and interrupted treatment plans. Families end up spending more on travel and missing work just to keep appointments, pushing them further into poverty, and healthcare workers under constant stress face a higher risk of mistakes and fatigue.

Previous efforts such as the 2011–2017 Kenya National eHealth Strategy and the 2016–2030 National eHealth Policy have laid a foundation for electronic health records and mobile health platforms (Republic of Kenya MoH, 2011; Republic of Kenya MoH, 2016). Researchers in Western Kenya found that, while many level 3 and level 4 facilities now have basic e-health services in place, these systems rarely go beyond electronic record-keeping or simple apportionment calendars; they do not dynamically tag high-risk groups such as the elderly, perform real-time triage, or absorb schedule disruptions with buffer slots (Ondulo, 2020). Studies of electronic queue-management systems (EQMS) at Premier Hospital in Mombasa demonstrated modest improvements in patient satisfaction and reduced average wait times by 15%, but the systems did not integrate clinician availability or emergency prioritization, limiting their impact during sudden patient surges (Kuria, 2021). Despite these advances, no current platform in Kenya offers a unified, web-based interface that ensures predictable doctor availability, allows real-time tagging and re-prioritization of patients by emergency status, and manages dynamic queue buffers for inevitable delays. This project proposes a focused webbased application to coordinate operations in a clinic.

## 1.2 Problem Statement

In many Kenyan clinics, the patient appointment process typically follows a first-come, firstserve (FCFS) model. Patients arrive at the facility, get assigned an available time slot and are scheduled accordingly. This approach lacks the flexibility to account for varying patient needs and doctor availability. As a result, it often leads to inefficiencies such as overlapping appointments, extended waiting times and underutilization of medical staff. The absence of automated queueing based on the nature of consultation and doctor availability further complicates scheduling and service delivery (CIHEB, 2023).

A study conducted at the University of Nairobi Staff Clinic in late 2021 found that prolonged waiting times negatively affected patient behaviour. Patients often missed or arrived late for appointments, delayed starting treatment and struggled to follow care plans. The findings highlight that long wait times can significantly impact treatment adherence and the overall quality of healthcare delivery (Wafula & Ayah,2021).

A third delay analysis in Kenyan secondary-level public hospitals demonstrated that suboptimal patient flow management, inadequate continuity of care resources and scheduling inefficiencies directly contributed to delays starting essential treatments, compounding risks for time sensitive conditions (Onyango et al., 2024). If left unchecked, these pending appointments increase the burden on the administration and elevate patient mortality rates, especially among the elderly or chronically ill who miss critical treatment windows.

To address these challenges, we are developing a web-based application that is designed using an Object-Oriented Analysis and Design paradigm. It employs a three-tier architecture and a MySQL data layer for storage. This system integrates real-time doctor availability tracking, dynamic queue buffering, and patient priority tagging for the elderly, chronically ill and emergencies. Automated SMS reminders and notifications reduce missed appointments, while built-in buffers absorb overruns and prevent cascading delays. A centralized dashboard gives administrators end-to-end visibility of schedules and queue status, enabling proactive adjustments and smoother patient flow throughout the clinic.

## 1.3 Aim/ Specific Objectives

To develop a web-based application to coordinate operations in a clinic that optimizes patient scheduling, enhances queue management and improves service delivery in outpatient healthcare settings.

### 1.3.1 Specific Objectives

1. To assess factors influencing operational efficiency in outpatient clinic settings.
2. To review existing systems that coordinate operations in a clinic setup.
3. To design a web-based application that supports real-time doctor availability tracking, patient priority tagging and dynamic queue buffering.
4. To test the developed system in a simulated clinic setup.

### 1.3.2 Research Questions

1. What are the factors influencing operation coordination in outpatient clinic setups?
2. How effectively do existing systems coordinate operations in a clinic setup? iii. How can digital solutions be used to prioritize patients based on urgency and vulnerability? iv. How effective is a web-based application in coordinating operations in a simulated clinic setup?

## 1.4 Justification

This project addresses a key challenge in Kenya clinic: long delays caused by inflexed appointment system. It benefits patients by cutting waiting times, especially for the elderly and emergency cases, clinics and doctors by enabling smoother daily operations, fewer missed appointments and better use of their time and heath regulators for policymakers by providing accurate data on booking trends to inform staffing levels and infrastructure planning.

**1.5 Scope**

The system focuses on outpatient clinics managed by practitioners who have their own clinics. It provides modules for patient profile management, automated appointments booking with dynamic queue prioritization (including urgency tagging for elderly, chronically ill and emergency cases), SMS confirmation and update notifications, a real-time administrative dashboard displaying upcoming appointments and doctors’ availability. All driven by simulated yet realistic datasets reflecting typical Kenyan clinic decision support features (e-prescription), billing or insurance processing, or integration with national health information systems.

## 1.6 Limitations

Anticipated challenges include user resistance to adopt a new digital workflow and limited access to real-time data from hospitals for testing.

## 1.7 Delimitation

To address these limitations, the application is user friendly with an intuitive administrative dashboard and the project uses simulated datasets that reflect typical hospital appointment flows, patient categories and doctor availability scenarios.

# Chapter 2: Literature Review

## 2.1 Introduction

This chapter reviews the existing literature on outpatient scheduling and queue management in clinics. It begins by describing the current appointment processes and contextual factors, then examines the key challenges that affect efficient patient flow. Next, three representative systems are reviewed to understand their designs, strengths, and limitations. From the review of the existing systems, we identify the gaps present in current approaches. Finally, we look at the technologies commonly used in related solutions and present a conceptual framework that situates the proposed web-based scheduling and priority-management system within the broader health‐care context.

## 2.2 Description of the Current Process of Coordinating Operations in a Clinic setup

### 2.2.1 Appointment Assignment Process

In most Kenyan clinics, the patient appointment process typically follows a first-come, first serve (FCFS) model where clerks allocate 25-minute slots without buffers for delays or emergencies. Clinicians manually update their availability at the start of each day. If appointments run over their allocated time delays compound leading to extended wait times, overcrowded waiting areas and growing patient frustration. These inefficiencies reduce the daily throughput forcing clinics to either rush consultations or to leave patients waiting indefinitely, diminishing both the quality of care and resource utilization (Moh, 2016; Catherine, 2019).

### 2.2.2 Tracking Real-time Clinician Availability

Most Kenyan Clinics rely on basic digital calendars to record clinician schedules with updates made mainly at the start of each day. Unforeseen events such as a doctor’s strike, emergencies or transport issues go unreported until patients arrive. Without real-time visibility on practitioner status, patients can be left stranded all day or other clinicians must absorb extra workload, placing additional strain on staff and rising burnout (AP News, 2024).

### 2.2.3 Queue and Triage Management

Patient queues in Kenyan clinics commonly follow a FCFS line. There is no standardized triage protocol in most level 4-5 hospitals hence the clinic staff often make these decisions on the spot without a set process. Consequently, ill patients may wait alongside cases that are not urgent, increasing their risk of mortality. The lack of dynamic re-ordering and buffer insertion leads to both safety hazards and operational inefficiencies as clinics cannot adapt to sudden surges (Wachira & Martin, 2011).

### 2.2.4 Patient Communication and Notifications

Appointment reminders and updates are often limited to manual calls from clerks. Few clinics employ automated SMS systems and those that do usually inform you about your position in a queue. Lack of consistent, automated reminders leads to high rates of missed appointments and unpredictable schedule gaps. When patients miss appointments, valuable clinician time goes unused leading to wasted resources. (Wafula & Ayah, 2021).

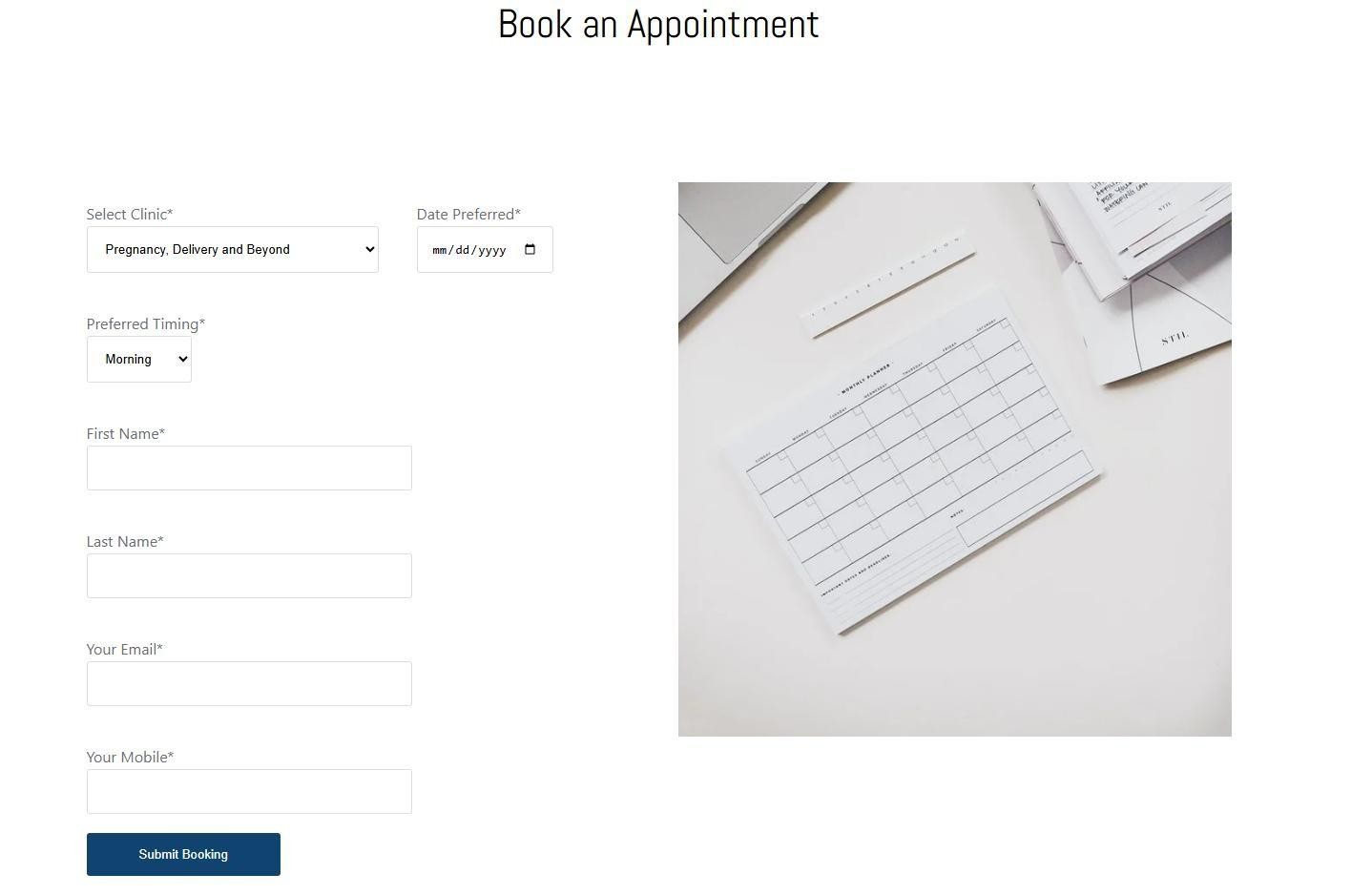
## 2.3 Challenges in the Current Processes of Coordinating Operations in a Clinic Setup

There are several deficiencies exist in the coordination of clinic operations in Kenya, particularly in appointment scheduling, clinician availability, queue management, and patient communication. Appointments are typically assigned on a first-come, first-serve basis, with fixed 25-minute slots and no buffers, leading to delays and overcrowded waiting areas. Clinicians manually update their availability daily, and sudden absences go unreported, causing patients to wait or overburdening other staff. Most clinics lack proper triage systems, forcing staff to prioritize patients on the spot, increasing risks for critical cases. Communication is often manual, with few clinics using automated SMS reminders, resulting in high no-show rates and wasted clinician time. While some digital tools have been introduced, such as calendars or queue SMS updates, they are limited and do not address realtime coordination or automated reordering. As a result, these inefficiencies persist, lowering service quality and increasing staff burnout. There is a critical need for a dynamic system to fill these gaps and improve care delivery.

## 2.4 Systems used in Coordinating Operations in a Clinic Setup

### 2.4.1 Premier Hospital EQMS

Premier Hospital in Mombasa deployed an Electronic Queue-Management System (EQMS) in 2019 to digitize check-in and notify patients via SMS when their turn approached (Abdulle, 2021). Patients register on arrival; the system calculates an estimated wait time and sends a text reminder before their consultation. It uses PHP backend, MySQL database, and a third-party SMS API. Though it addressed the long wait times and congested waiting areas, it lacks integration with clinician schedules and cannot reprioritize for emergencies or high-risk patients.



*Figure 2.1 Booking an Appointment (Premier Hospital, 2021)*

### 2.4.2 Luma Health

Luma Health is a cloud-based operations coordination platform for outpatient clinics. It addresses scheduling inefficiencies, unpredictable clinician availability and poor communication by offering online and SMS appointment booking and real-time publication of clinician calendars that automatically block off breaks or emergency absences. The system manages waitlists by automatically re-slotting missed appointments, applies basic rule-based patient prioritization and sends two-way SMS, email and in-app messages for confirmations. Built with: HTML5, CSS3 and JavaScript on the front end, a RESTful API-driven Node/PHP backend and a third-party SMS gateway, it also integrates bi-directional HL7/FHIR interfaces with major EMRs to share clinical data seamlessly (Luma Health, 2024). By combining scheduling, real-time availability, queue management, and patient notifications into one interface, Luma Health reduces missed appointments, smooths patient flow, and centralizes communication, though its prioritization remains rule-based and its buffer slots are statically defined rather than dynamically sized.

### 2.4.3 Qmatic Orchestra

Qmatic Orchestra is a mobile and desktop Customer Flow Management platform designed to streamline clinic operations. It addresses inefficient queue management and lack of real-time staff visibility by using a central engine that directs patients to the correct room or clinician as soon as they check in. The platform integrates with existing staff scheduling systems to display live on-duty status and room assignments. It dynamically rebalances queues by merging pre-booked appointments with walk-ins and sends SMS and email alerts. It uses HL7 interfaces to pull up electronic charts on demand and is built as a hybrid web and on-premises solution with APIs for EMR and SMS gateway connectivity (Qmatic, 2024). However, it lacks built-in mechanisms for feedback collection.

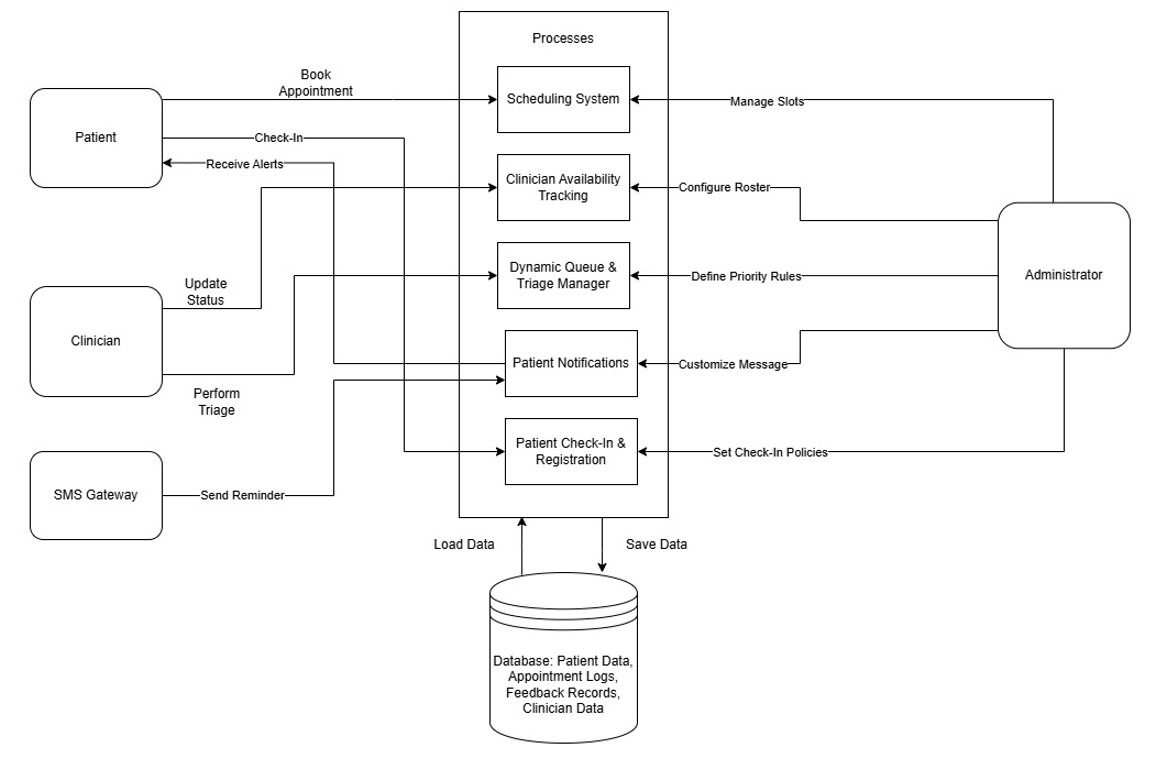
## 2.5 Gaps in Related Works

Although each system offers improvements, none simultaneously provides the following: real-time clinician availability updates to prevent booking conflicts; dynamic priority tagging for elderly, chronic, or emergency cases; automated buffer-slot insertion to absorb overruns without compounding delays; and an integrated administrator dashboard for end-to-end visibility of patient flow and service metrics.

## 2.6 Technology to be used

The proposed system uses several technologies to ensure effective coordination of clinic operations. An SMS gateway is used to send appointment confirmations and reminders to patients, to help reduce missed appointments. The web frontend is developed using HTML, CSS and JavaScript to ensure responsive design and compatibility with desktops and tablets commonly used in clinics. The backend is built using PHP and AJAX to enable real-time asynchronous updates of clinician availability and queue status. For data storage, MySQL is employed as a reliable relational database to manage patient records, appointments and priority levels efficiently.

## 2.7 Conceptual Framework



*Figure 2.2 Conceptual Framework of the Application*

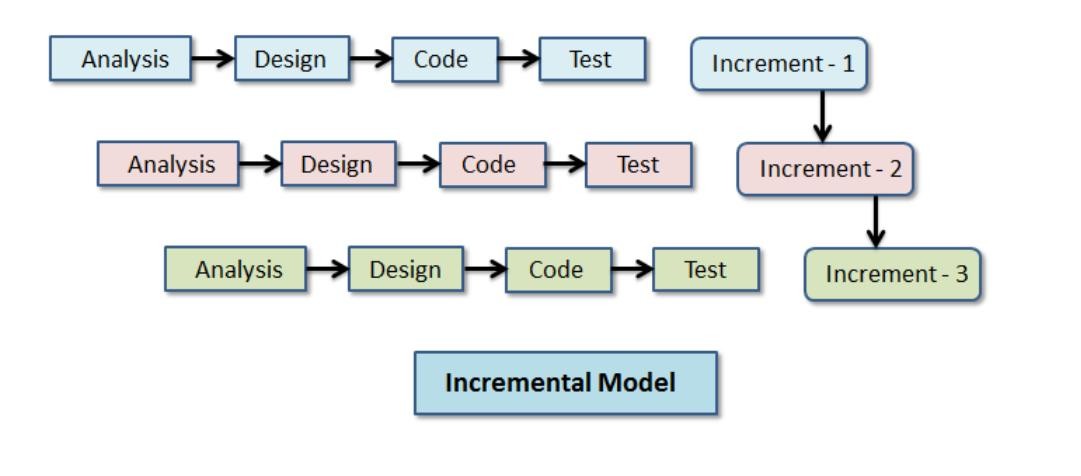
# Chapter 3: Methodology

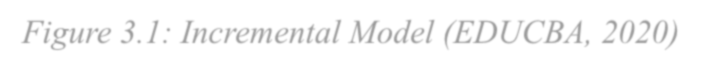
## 3.1 Introduction

This chapter outlines the overall research and development approach for the clinic operations coordination system. It explains the chosen paradigm and software methodology, describes each phase of the development process and details the analysis and design activities. It also presents the tools and techniques that are used and explains the concrete deliverables that result from this work.

## 3.2 Applied Development Approach

The project adopts an Object-Oriented Analysis and Design (OOAD) paradigm together with the Incremental Development Model. OOAD suits clinic workflows—such as patient bookings, doctor availability updates, queue prioritization, and feedback collection—because it maps each concept to a self-contained class. This structure is intuitive, easy to maintain, and supports clear traceability of requirements. The Incremental Model was chosen to deliver functionality in successive, deployable increments. Each increment adds or refines features, allowing clinic staff to validate and provide feedback on working software before the next increment begins.



*Figure 3.1: Incremental Model (EDUCBA, 2020)*

### 3.2.1 Requirements & Planning (Increment 0)

During Increment 0, clinic staff collaborate with the development team to gather and refine user stories, establish clear acceptance criteria, and prioritize the product backlog according to business value. By the end of this phase, a ranked feature list and a comprehensive project schedule for all subsequent increments are in place, and the development environment is fully configured.

### 3.2.2 Design & Development (Increment 1 & 2)

During Increment 1, the Appointment Scheduling module is designed and implemented, incorporating slot allocation and buffer-insertion logic to absorb overruns. Once development is complete, unit tests verify that each scheduling function meets its defined acceptance criteria, ensuring accurate slot assignments and robust handling of delays.

In Increment 2, the focus shifts to the development of Tracking Real-Time Clinician Availability and Dynamic Priority Tagging modules. These components enable live status updates for practitioners and automated reprioritization of patients based on urgency. Concurrently, the SMS Notification service is integrated, and all new modules undergo unit testing to confirm seamless interaction and adherence to requirements.

### 3.2.3 Integration & Validation (Increment 3)

During Increment 3, the Appointment, Availability, Prioritization, and Notification modules are combined to create a working prototype. System integration tests verify end-to-end workflows and ensure seamless interaction among all components. Demonstration sessions using simulated data. Observations and suggestions gathered during these sessions are documented and analysed to identify areas for refinement.

### 3.2.4 Refinement (Increment 4)

During Increment 4, user interfaces are fine-tuned, defects identified in validation are resolved, and system performance is optimized to meet operational requirements.

## 3.3 System Analysis

This section describes the analytical models that capture system requirements behaviour.

### 3.3.1 Use Case Diagram

Defines primary actors (Patient, Doctor, Administrator) and their interactions -such as “Book

Appointment”,” Update Availability”, and “View Dashboard”-to ensure all functional requirements are captured.

### 3.3.2 Sequence Diagram

Illustrates the keys scenarios. For example, the booking process:

Patient → Booking Controller → Scheduling Engine → SMS Gateway → Patient.

### 3.3.3 Entity-Relationship Diagram (ERD)

Specifies data entities (User, Appointment, Feedback, Priority Level), their attributes and relationships. This underpins the relational database design.

### 3.3.4 Class Diagram

Shows system classes (e.g. Patient, Doctor, Appointment Manager, Notification Service), their methods, attributes and associations-guiding object-oriented implementation.

### 3.3.5 Activity Diagram

Maps critical workflows such as “Process New Appointment,” including decision points (e.g., “Is slot available?”; “Is patient priority high?”).

### 3.3.6 State Diagram

Models the lifecycle of an Appointment object through states: Requested → Confirmed → Completed → Feedback Submitted.

## 3.4 System Design

This section outlines design artifacts that translate analysis models into concrete structures.

### 3.4.1 Database Schema

Derived from the ERD, the schema defines tables (users, appointments, feedback, priorities), columns (datatypes, keys), and indexing strategies to support efficient queries for real-time availability.

### 3.4.2 Wireframes

Low-fidelity sketches of key user interfaces—appointment form, priority settings, administrative dashboard, and feedback survey—illustrating layout, navigation, and data entry flows.

### 3.4.3 System Architecture

A three-tier architecture comprising:

Presentation Layer: HTML, CSS, and JavaScript for responsive web pages.

Application Layer: PHP controllers implementing business logic (OO-based), RESTful endpoints, and integration with the SMS gateway.

Data Layer: MySQL database for persistent storage of clinic data.

## 3.5 System Development Tools and Techniques

Below are the primary tools and justification for their selection.

### 3.5.1 Integrated Development Environment

Visual Studio Code is chosen for its simple nature, rich plugin ecosystem (PHP, MySQL, JavaScript), and integrated Git support, which streamlines coding and version control.

### 3.5.2 Backend Language & Framework

PHP with Composer - Offers broad hosting support, mature libraries (including SMS gateway SDKs), and easy integration with MySQL. Composer allows structured dependency management.

### 3.5.3 Frontend Technologies

HTML, CSS, JavaScript - Ensure fast load times and compatibility on clinic desktops and tablets.

AJAX is used for asynchronous updates such as live queue status.

### 3.5.4 Database Management System

MySQL - Provides robust transaction support and indexing, essential for reliable appointment booking under concurrent access. XAMPP is used for local development.

### 3.5.5 Collaboration & Version Control

Git on GitHub facilitates branching for feature development, pull-request reviews, and issue tracking to manage tasks, bugs, and enhancements.

## 3.6 Deliverables

The following deliverables align with the project milestones and examination requirements:

**3.6.1 Requirements & Design Documentation**

Detailed proposal, user stories, use case descriptions, and analysis or design diagrams.

### 3.6.2 Working System Modules

User Management Module: Registration, login, profile updates.

Appointment Module: SMS reminders and confirmations.

Dashboard Module: Real-time schedule overview, queue metrics and visualisation.

Feedback Module: Post-visit surveys and feedback storage.

**3.6.3 Test Artifacts**

Unit, integration and acceptance test cases with execution reports, performance or load testing results.

# 6.4System Demonstration and User Manual

A demonstration version of the system is prepared for presentation purposes.

## 3.6.5 Final Report

Comprehensive write-up of methods, implementation details, evaluation of results and recommendation for future work.

# Chapter 4: System Analysis and Design

## 4.1 Introduction

This chapter provides a detailed overview of the clinic operations coordination system. It explores both functional and non-functional requirements that are essential for the system’s development and operation. The chapter begins by outlining the key functionalities of the system, such as authentication, patient and clinician registration, appointment booking, dynamic queue management, feedback submission, and administrative management. These features ensure seamless user interactions and efficient clinic operations. Non-functional requirements like security, responsiveness, availability, scalability, maintainability, and usability are also discussed to guarantee the system’s performance and reliability. The chapter concludes with system analysis diagrams, including use case, sequence, entity relationship, and activity diagrams, followed by the system design diagrams such as database schema and wireframes. These diagrams serve to represent the interactions, workflows, and structure of the system, offering a visual understanding of its design and architecture.

## 4.2 System Requirements

These are the functionality needed in order to satisfy the user's requirements.

### 4.2.1 Functional Requirements

1. Authentication

The requirement applies to patients, clinicians and administrators. It enables them to log in securely and access the system. The user provides a valid email and password through input fields. The system validates these credentials against stored values in the database. Upon successful match, the user is redirected to their respective dashboard. Failed login attempts return error messages. Passwords must be strong, with a minimum length and a mix of characters. Users can also log out anytime.

1. Patient Registration Functionality

This functionality allows new patients to register on the platform. They provide their full name, email, mobile number, password, date of birth, gender and emergency contact details such as name and emergency contact number. The system checks if the email is already in use, ensures password complexity and stores the validated information in the patient database.

1. Clinician Registration Functionality

This functionality enables clinicians to register by providing their full name, email address, phone number, password, gender, date of birth and medical specialization. The system stores this data in the clinician database after verifying that the email is unique and all required fields are complete.

1. Appointment Booking Functionality

This functionality allows patients to book appointments with clinicians. The patient selects a date, time slot, reason for visit and preferred doctor. The system checks availability and inserts buffer slots. It then stores the appointment details, tags the priority level and sends confirmation via SMS. If a clinician is unavailable, the system suggests alternative slots.

1. Clinician Availability Update Functionality

This feature allows clinicians to update their availability on a daily basis. They specify time blocks during which they are available for appointments. The system captures this data and updates the real-time availability on the administrative dashboard. This prevents appointment overlaps ad supports dynamic rescheduling.

1. Dynamic Queue Management and Priority Tagging Functionality

This functionality ensures that patients are seen based on urgency rather than order of arrival. The system tags patients based on age, chronic condition or emergency indicators. During appointment booking, if high priority is detected, the patient is moved up the queue. Admins can also manually change the queue order in real time.

1. Patient Feedback Submission Functionality

This feature allows patients to give feedback after their appointments. The system presents a form where patients rate their experience, leave comments and answers simple satisfaction questions. Feedback is linked to specific appointments and stored for analysis by administrators.

1. Administrative Dashboard Management Functionality

This functionality is only available to administrators. It allows them to view live queues, clinician schedules, upcoming appointments and feedback reports. Admins can add or remove clinicians, adjust queue settings, view system analytics and broadcast notifications. The dashboard offers visual summaries such as charts and tables.

1. SMS Notification Functionality

This functionality sends automated SMS message to patients for appointments confirmations, reminders and updates. The system integrates with an SMS gateway and triggers notification when appointments are booked, rescheduled or cancelled. The message content is logged and linked to the appointment.

1. Report Generation Functionality

This requirement enables administrators to generate reports on appointments, wait times clinician utilization and patient satisfaction. These reports are compiled from appointments, feedback and availability records.

### 4.2.2 Non-Functional Requirements

1. Security

The system ensures that sensitive patient and clinician data is protected from unauthorized access. Authentication is enforced through secure login for all users using encrypted passwords. Access control is role-based, allowing patients, clinicians, and administrators to access only relevant features. All communication between the client and server, especially involving personal or medical data, is protected using HTTPS. The database implements proper permission settings and prevents SQL injection attacks.

1. Responsiveness

The system responds to user requests within acceptable time limits. AJAX is used to enable asynchronous updates, reducing load times when accessing real-time clinician availability or current queue status.

1. Availability

The system aims to be available most of the time during clinic operational hours. This includes the backend server, frontend interface, database and SMS notification service.

1. Scalability

The system is scalable to accommodate growth in the number of patients, clinicians, and appointments without compromising performance. The architecture supports future expansion to include more clinics or departments. Database indexing, modular backend components, and load balancing strategies are applied to handle increased demand.

1. Maintainability

The system is to be easy to update and maintain. The use of Object-Oriented Design and modular components ensures that future updates, bug fixes and feature additions can be done without affecting the rest of the system.

1. Usability

The user interface is intuitive and simple for all users. The forms include clear labels, placeholders, and real-time validation messages. The dashboard uses charts and tables to present data clearly. Accessibility features such as large fonts and contrast settings are be considered for elderly users.

1. Reliability

The system functions correctly under normal and peak conditions. Modules like appointment scheduling and SMS notifications operate without failure to ensure trust in the system. Data consistency is maintained even when multiple users are accessing or modifying records simultaneously.

1. Portability

The system is accessible from standard desktop browsers and mobile devices browsers without requiring installation.HTML, CSS and JavaScript ensure compatibility across devices, while responsive design ensures proper display on both tablets and PCs used in the clinics.

1. Data Integrity

All data stored in the system is accurate and consistent. For example, once an appointment is confirmed, it is overwritten without proper authorization. Input validation and referential integrity constraints are implemented at both the application and database levels.

1. Auditability

The system maintains logs of major actions such as login attempts, appointment changes, clinician availability updates, and admin interventions.

## 4.3 System Analysis Diagrams

### 4.3.1 Use Case Diagram

Figure 4.1 below illustrates how the main actors interact with the various use cases of the system. The system includes three main actors, namely the Patient, Clinician, and Administrator, as well as two external systems called the SMS Gateway and EMR System, which work with the system to support communication and integrate medical records.

Some use cases such as Register, Login and Logout are shared across all user types to access the system securely.

The Patient actor can register their profile, request appointments, view notifications, provide feedback after consultations, and check in.

The Clinician is responsible for updating their real-time availability, viewing scheduled appointments, and write notes during consultations. They also view submitted feedback and contribute to improving patient care delivery.

The Administrator manages the overall system including clinician accounts, patient records, and configuration of buffer times and queue rules. The administrator also accesses the real-time dashboard to monitor clinic flow, generate service delivery reports, and adjust settings based on trends or feedback.

The SMS Gateway is triggered by appointment bookings and updates, enabling the system to automatically send SMS notifications for confirmations, reminders, or rescheduling.

The EMR System records patient visit summaries and provides patient records.

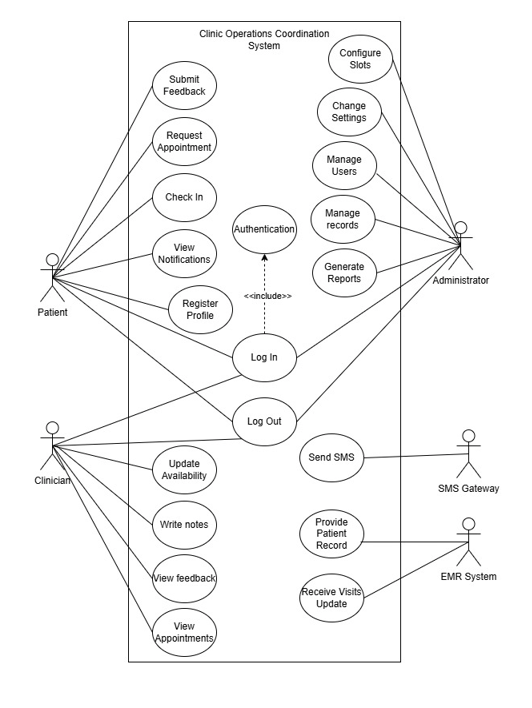


Figure 4.1

### 4.3.2 Sequence Diagram

When a Patient logs in, the System authenticates against the Database and returns success or failure; similarly, Clinicians and Administrators authenticate and log out through the System with corresponding Database checks. To book an appointment, the Patient sends a request to the System, which reserves an available slot in the Database, sends confirmation back to the Patient. Upon check-in, the Patient informs the System, which updates queue status in the Database, confirms check-in, sends an SMS alert. When a Clinician updates availability, the System saves the new schedule in the Database and then confirms the save. An Administrator configuring slots causes the System to save slot rules in the Database, confirm the configuration while managing user accounts similarly updates the Database and confirms the action.

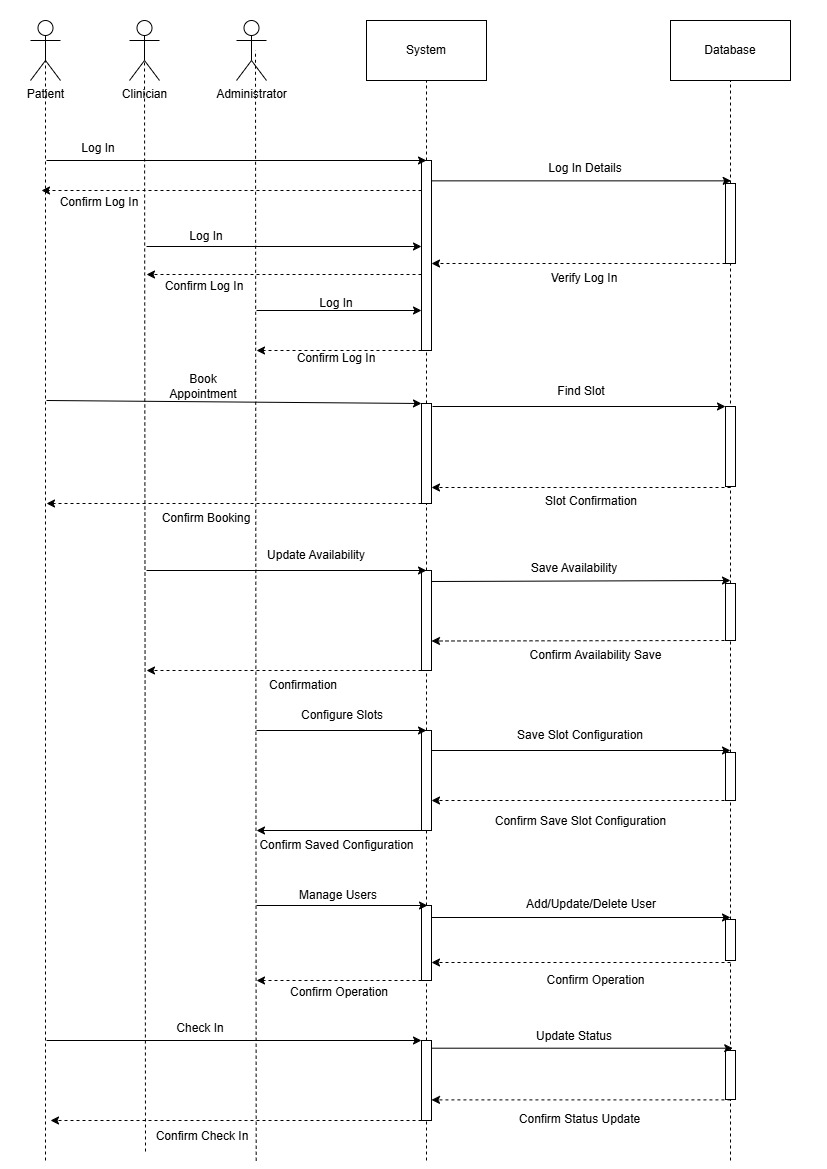


Figure 4.2

### 4.3.3 Entity Relationship Diagram

The system consists of several key entities including Patient, Clinician, Appointment, Admin, Feedback, ClinicianAvailability, PriorityManager, and external interfaces such as the SMS Gateway and EMR System. Figure 4.3 below illustrates how these entities are related within the system.

For example, the relationship between Patient and Appointment is one-to-many, meaning a patient can book multiple appointments, but each appointment is linked to only one patient. Similarly, a Clinician can be assigned multiple appointments, forming another one-to-many relationship.

The Admin entity manages system-level tasks and is associated with many patients and clinicians, resulting in a one-to-many relationship from Admin to those entities. The Feedback entity is linked to Appointment, indicating that each piece of feedback is submitted for a specific appointment.

The ClinicianAvailability entity is connected to Clinician, capturing their working hours or availability status on a daily basis. The PriorityManager entity is associated with Patient and is used to define whether a case is categorized as emergency, chronic, or normal.

The SMSNotification and SMSGatewayLogs tables track communication sent from the system to patients or clinicians. Each SMSNotification can have multiple SMSGatewayLogs, which audit the delivery of messages. Additionally, EMRRecords link Patients, Clinicians, and Appointments, representing the clinical documentation tied to each visit.

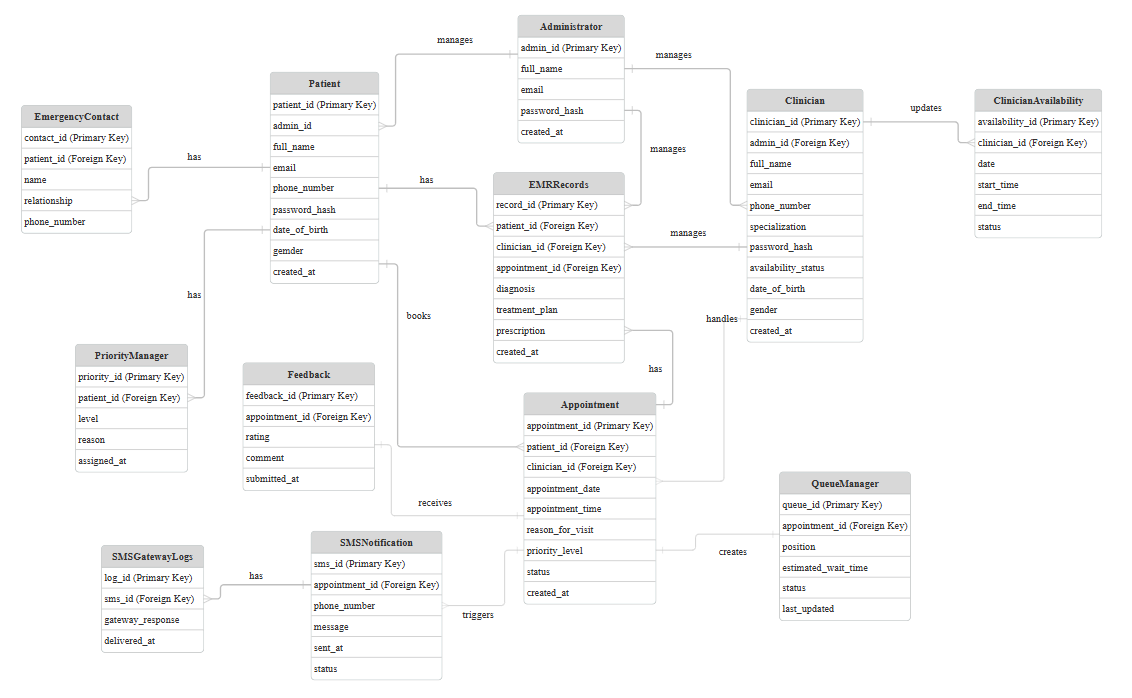


Figure 4.3

### 4.3.4 Activity Diagram

Figure 4.4 below illustrates the workflow of the system from start to finish. The system has three main users: Patient, Clinician, and Administrator, each interacting with the system through distinct processes, but beginning at a common start point—user registration. All users are first required to register and then log into the system.

Once logged in:

Patients can perform key operations such as updating their profiles, booking appointments and select their priority level. After attending the appointment, patients submit feedback which is stored in the system for evaluation.

Clinicians can update their availability and review upcoming appointments. After appointments, they can submit the consultation outcome.

Administrators monitor system-wide operations through a real-time dashboard. They can view clinician availability, manage patient records, configure buffer time rules, and generate operational reports.

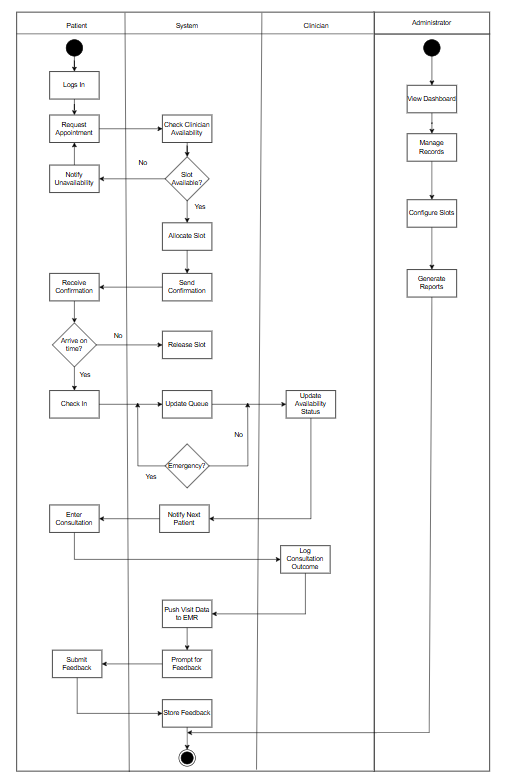


Figure 4.4

### 4.3.5 Class Diagram

The class diagram, as shown in Figure 4.5, describes the attributes and operations of the classes in the clinic operations coordination system. It also highlights the relationships between the classes and their responsibilities in managing clinic workflows. The system consists of multiple core classes, each representing a major actor or function in the clinic. The "Patient" and "Clinician" classes lie at the center of the system, representing the individuals who interact with the clinic either by seeking services or providing care. These classes contain attributes such as fullName, email, phone, dob, and gender, and offer methods to register, log in, book appointments, or update availability.

The "Admin" class is responsible for overseeing system-wide operations. It handles functions such as managing users, generating reports, and configuring queue buffers. The "Appointment" class handles all activities related to appointment creation, confirmation, and cancellation. It interacts directly with the patient and clinician classes to assign slots and insert buffer time where needed.

The "ClinicianAvailability" class keeps track of real-time practitioner status, allowing clinicians to update their availability throughout the day. The "PriorityManager" class tags patients based on urgency levels such as emergency, elderly, or chronic illness, helping prioritize them dynamically in queues.

The "SMSNotification" class enables communication between the system and the patient by sending SMS reminders and updates. It works closely with the "SMSGateway" class, which manages the actual delivery of messages through a third-party SMS platform.

Feedback from patients is managed through the "Feedback" class, which captures ratings and comments after an appointment is completed. This feedback is stored and can be retrieved by administrators for service improvement purposes.

Finally, the "EMRSystem" class integrates external electronic medical records into the platform, storing clinical notes and consultation history. This ensures continuity of care and allows clinicians to access patient records during appointments.

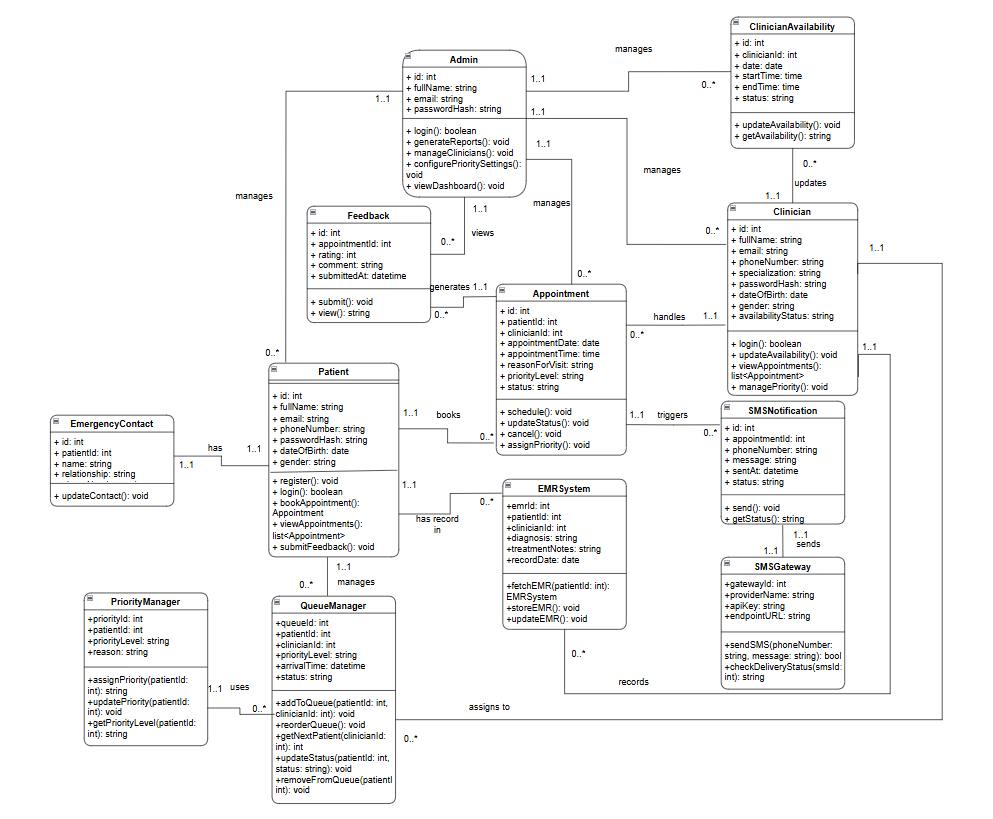


Figure 4.5

## 4.4 System Design Diagrams

### 4.4.1 Database Schema

The database schema, as shown in figure 4.6, provides a structured representation of the various entities, their attributes, and the relationships within the clinic operations coordination system. This schema reflects the real-world workflows of outpatient clinics, enabling structured storage and retrieval of operational and clinical data.

Key entities include Admins, Patients, and Clinicians, which represent the primary users of the system. Each of these entities contains core attributes such as full name, email, phone number, and authentication details. The admin entity manages the system, with each admin linked to multiple clinicians and patients.

The Appointments table records individual consultations, linking a patient and clinician with appointment-specific details like date, time, priority level, and status. Emergency contacts for patients are stored in the EmergencyContacts entity with a strict one-to-one relationship for each patient.

Feedback is linked to appointments and is used to capture patient ratings and comments. The ClinicianAvailability table logs daily schedules, aiding real-time tracking of clinician availability. Notifications are handled through the SMSNotifications table, which stores outgoing messages and links them to appointments.

Supporting modules include EMRRecords, which holds diagnosis and treatment data, associating patients, clinicians, and appointments. The PriorityManager table assigns priority levels to patients, especially the elderly, chronically ill, or emergencies, based on clinician input. The QueueManager table manages the dynamic queue by maintaining patient positions, estimated wait times, and statuses.

Lastly, SMSGatewayLogs ensures traceability and auditing of SMS delivery status, capturing gateway responses for each message sent.

To ensure data integrity, the schema incorporates primary keys for each table, foreign key constraints for relationships, and uniqueness constraints for emails and SMS records.

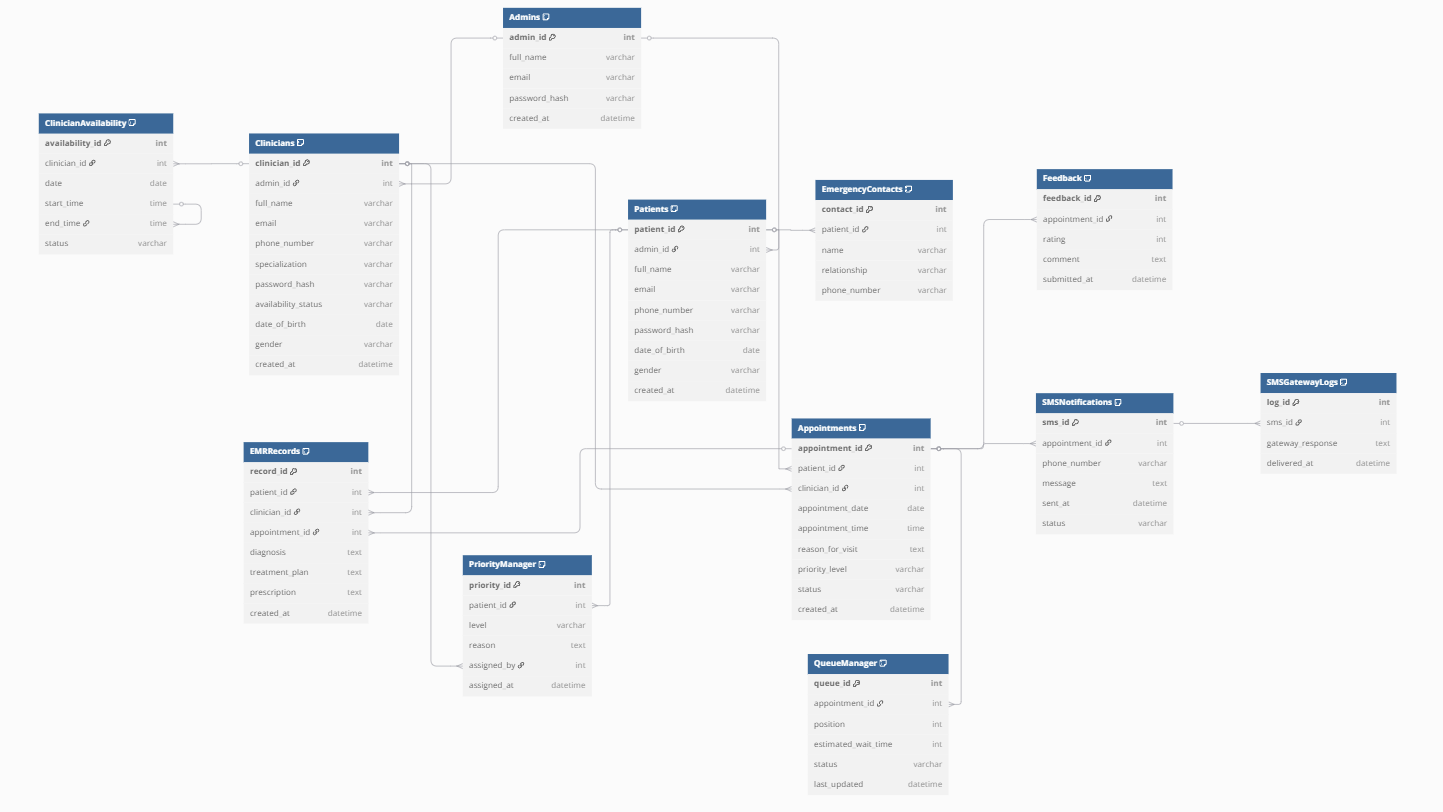


Figure 4.6

### 4.4.2 Wireframes

i) Login Page

The Login Page is designed to provide secure and easy access to the system for all users. It includes input fields for entering an email address and password, followed by a login button to initiate authentication. Additionally, the page offers a navigational link for users who have forgotten their password and another for new users to create an account.

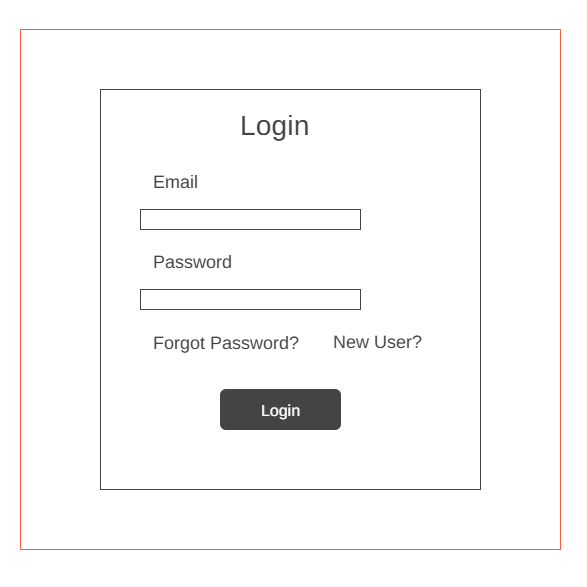


Figure 4.7

ii) Appointment Form

The Appointment Form is designed to allow patients to book outpatient visits efficiently. It includes text fields for entering the patient's full name, preferred date and time, department, symptoms, urgency level and phone number. A submit button finalizes the request. The form captures essential scheduling data and enables priority tagging based on urgency. It serves as the starting point for queue management and helps match patients with available clinicians while ensuring appropriate triage and real-time tracking.

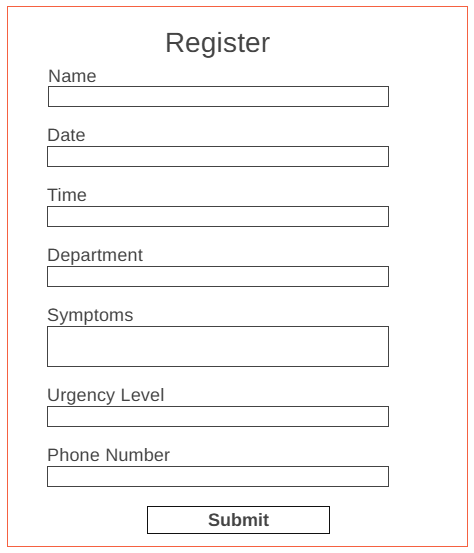


Figure 4.8

iii) Feedback Form

The Feedback Form enables patients to evaluate their experience after a consultation. It includes a field for entering the associated appointment ID, a rating scale from 1 to 5 to assess overall satisfaction, a yes or no question on whether the doctor was punctual, and an open-ended textbox for suggestions on how the service can be improved. This form supports service quality monitoring, gathers performance insights, and helps administrators identify recurring issues.

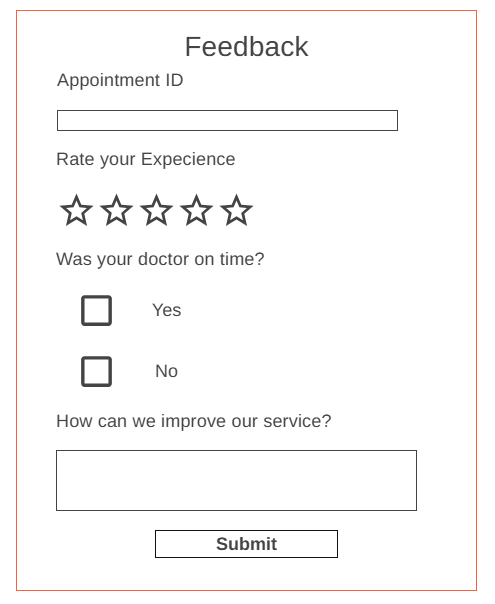


Figure 4.9

iv) Administrator Dashboard

The Administrator Dashboard centralizes key operational controls for clinic management. It features a real-time view of upcoming appointments and the live patient queue. A sidebar allows the administrator to manage users, view appointment history, access analytics, adjust system settings, and view notifications.

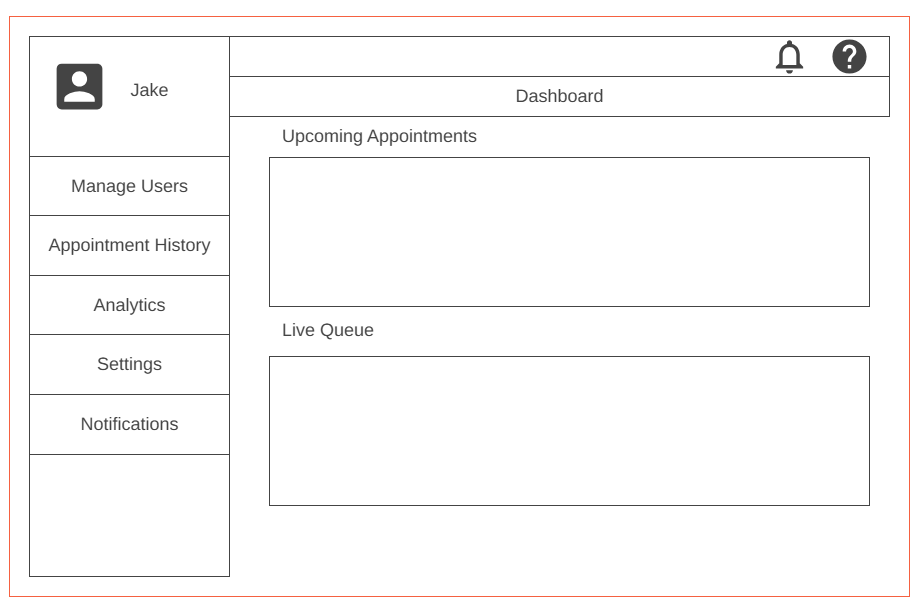


Figure 4.10

### 4.4.3 System Architecture

The system adopts a three-tier architecture to ensure modularity, scalability, and maintainability. The Presentation Layer uses HTML, CSS, and JavaScript to create responsive user interfaces accessible on desktop and mobile devices. The Application Layer consists of PHP-based controllers that handle core business logic using object-oriented programming. It also exposes RESTful endpoints and connects to the SMS gateway for communication. The Data Layer uses MySQL to manage and persist clinic data such as patient records, appointments, and clinician availability. This separation of concerns allows for efficient updates, secure data access, and smooth interaction between users and backend services.

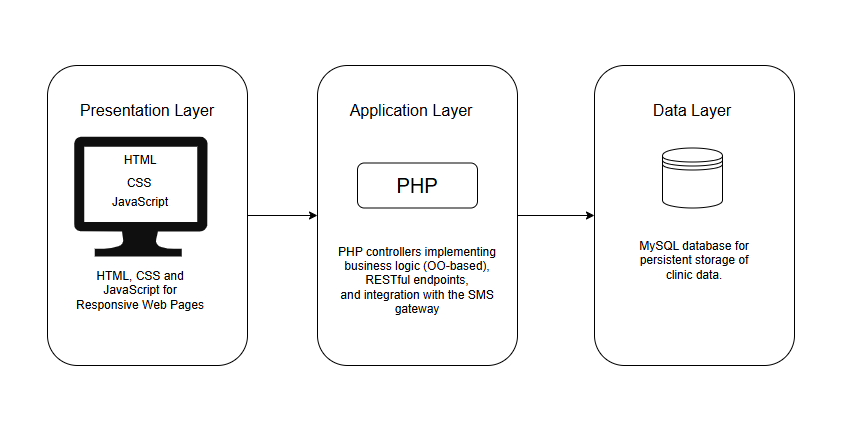


Figure 4.11

# References

Barasa, E. W., Mwaura, N., & Rogo, K. (2017). Caregiver costs for childhood tuberculosis treatment in Kenya. Journal of Global Health Reports. Retrieved from https://www.joghr.org/article/88168

Capital FM. (2024, September). M-TIBA cuts health-claim approval time by 96% with tech.

Capital FM Business. Retrieved from

https://www.capitalfm.co.ke/business/2024/09/m-tiba-cuts-health-claim-approvaltime-by-

96-with-tech/

CIHEB. (2023). Decreasing long waiting times through CQI. Clinton Health Access Initiative

– East, Central and Southern Africa. Retrieved from

https://www.ciheb.org/cqi/kenya/case-studies/decreasing-long-waiting-times-throughcqi/

Institute of Medicine. (2001). Crossing the Quality Chasm: A New Health System for the 21st

Century. National Academies Press. Retrieved from

https://nap.nationalacademies.org/catalog/10027/crossing-the-quality-chasm-a-newhealth-

system-for-the

Kuria, E. M. (2021). Evaluation of an electronic queue–management system at Premier

Hospital, Mombasa (Master’s thesis). United States International University – Africa.

Retrieved from

https://erepo.usiu.ac.ke/bitstream/handle/11732/7190/Kuria%2C%20Edwin%20Mutur a%20MIR%202021.pdf?sequence=1

Kuria, E. M., Otieno, D., & Wanjala, H. (2025). Impact of SMS reminders on outpatient attendance in Kenyan clinics: A quasi-experimental study. medRxiv. Retrieved from https://www.medrxiv.org/content/10.1101/2024.02.28.24303254v1

M-TIBA. (2022). M-TIBA Health Wallet Platform Overview. Retrieved from

https://www.capitalfm.co.ke/business/2024/09/m-tiba-cuts-health-claim-approvaltime-by-

96-with-tech/

Ministry of Health, Kenya. (2016). National eHealth Policy 2016–2030. Retrieved from https://kehia.org/resources/kenya-national-ehealth-strategy-2011-2017/

Ministry of Health, Kenya. (2019). Kenya Household Health Expenditure and Utilization

Survey 2018. Retrieved from https://www.treasury.go.ke/wpcontent/uploads/2021/05/REVISED-HEALTH-

SECTOR-REPORT.pdf

Namusonge, G. S., Wasunna, A., & Ayiero, P. (2022). Workload stressors among maternal care providers in Western Kenya. BMC Health Services Research. Retrieved from https://pmc.ncbi.nlm.nih.gov/articles/PMC10022275/

Njoroge, S., Wanjiku, P., & Kamau, T. (2021). Inefficiencies in outpatient scheduling and their impact on patient flow in Kenyan clinics. One Health Pan African Medical Journal,

5(17). Retrieved from https://www.one-health.panafricanmedjournal.com/content/article/5/17/full/?utm\_

Opon, R., Ochola, S. J., & Waweru, L. (2020). Factors influencing missed antenatal appointments in Homa Bay and Kisumu Counties, Kenya. One Health Pan African Medical Journal, 5(17). Retrieved from https://www.one-health.panafricanmedjournal.com/content/article/5/17/full/

Ondulo, C. I. (2020). Current status of e-health in Kenya and emerging global research trends.

Academia.edu. Retrieved from

https://www.academia.edu/71854659/Current\_Status\_of\_E\_Health\_in\_Kenya\_and\_E merging\_Global\_Research\_Trends?

OpenMRS Community. (2020). KenyaEMR+ Distribution and Appointment Module.

Retrieved from

https://openmrs.atlassian.net/wiki/spaces/docs/pages/25475417/KenyaEMR%2BDistri bution

Onyango, S., Mwangi, J., & Ouma, A. (2024). Third-delay analysis in Kenyan secondary-level public hospitals: Implications for patient flow. BMJ Open, 14(1), e072341. Retrieved from https://bmjopen.bmj.com/content/14/1/e072341.long

Republic of Kenya Ministry of Health. (2011). Kenya National eHealth Strategy 2011–2017. Retrieved from https://repository.kippra.or.ke/items/ac2e33c1-c1c3-4920b2ea63e91e49e2c3

Republic of Kenya Ministry of Health. (2014). The public health system in Kenya: Six levels of care. Retrieved from https://repository.kippra.or.ke/server/api/core/bitstreams/130e0906caa8-4a63-

8157ccf50d5c9e27/content#:~:text=The%20public%20health%20system%20consists,le vel %201)%20which%20includes%20all

Republic of Kenya Ministry of Health. (2016). Guidelines for timely service delivery.

Retrieved from https://repository.kippra.or.ke/server/api/core/bitstreams/c87714e853774a01-bf3d-7d3e18645a4b/content

Seif, R., Maina, L., & Otieno, M. (2025). Scheduling inefficiencies in outpatient clinics: A systematic review. International Journal of Research and Public Review, 6(4), 1385–

1402. Retrieved from https://ijrpr.com/uploads/V6ISSUE4/IJRPR41385.pdf

Wachira, B., & Martin, S. (2011). Triage without standardization: Emergency department practices in Kenyan hospitals. East Africa Family Medicine Journal. Retrieved from https://ecommons.aku.edu/eastafrica\_fhs\_mc\_fam\_med/18/?

Wanyee, K., Mwangi, T., & Odhiambo, P. (2019). Digital system limitations for patient prioritization in Kenyan outpatient clinics. PMC, 14(1). Retrieved from https://pmc.ncbi.nlm.nih.gov/articles/PMC6945428/

Wafula, G. O., & Ayah, R. (2021). Outpatient waiting times and patient satisfaction at the University of Nairobi Staff Clinic. International Journal of Innovative Research in

Medical Science, 8(2). Retrieved from https://www.ijirms.in/index.php/ijirms/article/view/1307

AP News. (2024, April). Kenya doctors’ strike leaves patients stranded at clinics. AP News. Retrieved from https://apnews.com/article/kenya-doctorsstrike2b7507464f5d77642eae9b771b6df54f

Muturi, S., & Kamau, E. (2021). Electronic queue-management system deployment at Premier Hospital, Mombasa. United States International University – Africa Repository. Retrieved from https://erepo.usiu.ac.ke/xmlui/handle/11732/7190?sequence=1

Catherine, A. (2019). Inefficiencies in outpatient scheduling and their impact on patient flow in Kenyan clinics. One Health Pan African Medical Journal, 5(17). Retrieved from https://www.one-health.panafrican-med-journal.com/content/article/5/17/full/

**Appendix**

# Appendix 1: Gantt Chart

