Software Design Specification

Hospital Token Management System

Prepared by

AADITYA C K

MUHAMMED SINAN

RINESH S

Table of Contents

Revision History

Version	Name	Reason for changes	Date

1.Introduction

1.1 Purpose of this document

The purpose of this document is to provide an overview of a mobile application designed to manage patient queue and appointment scheduling in hospitals. It explains the features and benefits of the application, as well as the technical aspects of its development and implementation. Moreover, this document provides a detailed description of the ability of the application to manage patient appointments, real-time tracking of patient flow, and the provision of alerts and notifications. It also outlines the technical requirements for implementing the application, including hardware and software requirements, as well as security and privacy considerations. It also enables quality assurance and testing, ensuring that the system's components and interfaces are tested thoroughly and that the system meets its design goals and objectives.

1.2 Scope of the Development Project

The scope of development of this project is quite broad and would require expertise in areas such as user interface design, database development, software engineering, and project management. Some stages involved are :

1. Requirements gathering: This stage involves identifying the specific needs of the hospital, including the number of patients, the types of appointments, and the existing patient flow management system.
2. Design and architecture: Once the requirements have been identified, the next step is to design the application's user interface, database structure, and system architecture.
3. Development: During this stage, the application is developed using appropriate programming languages and tools. The development team would also work on integrating the application with other systems used in the hospital, such as electronic health records (EHR).
4. Testing and quality assurance: Before deployment, the application would undergo rigorous testing to ensure that it functions as intended and meets the hospital's needs. This includes testing for functionality, performance, and security.
5. Deployment and implementation: Once the application has been tested and approved, it can be deployed and implemented in the hospital. This includes configuring the application for the hospital's specific needs, training staff on its use, and providing ongoing support and maintenance.

1.3 Definitions, acronyms and abbreviation

- Definition:

Hospital Token Management System : An Application which allows users to access their medical history, check doctors availability and book appointments or consultations.

- Acronyms:

HTM : Hospital Token Management

1.4 Overview of document

The hospital token management system application is a mobile application designed to manage patient queues and appointment scheduling in hospitals. The application is developed to streamline patient flow, reduce waiting times, and enhance the overall patient experience in healthcare facilities.

This application provides a solution for hospitals and healthcare facilities to manage patient queues effectively and s designed to provide real-time tracking of patient flow, manage patient appointments, and provide alerts and notifications to patients about their appointments.

The development of this project involves several stages, including requirements gathering, design and architecture, development, testing and quality assurance, and deployment and implementation and would also require expertise in areas such as user interface design, database development, software engineering, and project management. The application must be scalable, secure, and reliable and should integrate with other hospital systems to provide a seamless patient experience.

In summary, the development process of the project is extensive and requires expertise in various areas to develop a comprehensive solution that meets the hospital's specific needs as this application is a crucial solution for healthcare facilities to manage patient queues effectively, reduce waiting times, and enhance the overall patient experience.

2. System architecture description

2.1 Overview of modules / components

To provide a comprehensive solution for managing patient queues and appointment scheduling this application consists of several modules or components :

1. User interface (UI) design: The user interface design module includes the design of the application's graphical user interface (GUI). The GUI should be user-friendly and intuitive, allowing patients to easily book appointments and track their queue position.

2. Appointment booking module: This module allows patients to book appointments through the application. The module should allow patients to choose their preferred doctor, appointment time, and location.

3. Queue management module: This module is responsible for managing patient queues. The module should provide real-time tracking of patient flow, allowing healthcare professionals to manage patient queues effectively.

4. Notification module: This module provides notifications to patients about their appointment status, queue position, and any changes to their appointment schedule. The module should be designed to provide automated notifications to patients through various channels, such as SMS or push notifications.

5. Data management module: This module is responsible for managing the application's database. The database should store patient data, appointment schedules, and other relevant information securely.

6. Reporting and analytics module: This module provides reporting and analytics features to healthcare professionals, enabling them to track patient flow and identify areas for improvement.

7. Integration module: This module allows the application to integrate with other hospital systems, such as electronic health records (EHR) and billing systems.

8. Security module: This module ensures the application is secure and compliant with relevant data protection regulations. It should include features such as authentication and access control, data encryption, and regular security updates.

2.2 Structure and relationships

At the top of the structure is the user interface (UI) design module, which is responsible for designing the application's graphical user interface. The UI design module is dependent on the requirements gathering process, which identifies the specific needs of the hospital and its patients. Beneath the UI design module is the appointment booking module, which allows patients to book appointments through the application. The appointment booking module is dependent on the UI design module, as it needs to integrate with the application's graphical user interface.

Beneath the appointment booking module is the queue management module, which manages patient queues in real-time. The queue management module is dependent on the appointment booking module, as it needs to track patient appointments and manage patient flow accordingly. The notification module sits alongside the appointment booking and queue management modules, as it provides automated notifications to patients about their appointment status and queue position. The notification module is dependent on both the appointment booking and queue management modules, as it needs to access patient data and appointment schedules.

The data management module sits at the bottom of the structure, as it manages the application's database. The data management module is dependent on all the modules above it, as it needs to store patient data, appointment schedules, and other relevant information securely. The reporting and analytics module sits alongside the data management module, as it provides reporting and analytics features to healthcare professionals. The reporting and analytics module is dependent on the data management module, as it needs to access patient data and appointment schedules.

The integration module sits alongside the data management and reporting and analytics modules, as it allows the application to integrate with other hospital systems, such as electronic health records (EHR) and billing systems. The integration module is dependent on the data management module, as it needs to access patient data and appointment schedules, and on the reporting and analytics module, as it needs to provide reporting and analytics features to healthcare professionals .Finally, the security module sits alongside all the other modules, as it ensures the application is secure and compliant with relevant data protection regulations. The security module is dependent on all the other modules, as it needs to secure patient data and protect the application from external threats.

2.3 User Interface

2.3.1 Login / Register Screen

The first screen shown after opening application. Here user needs to enter details to enter into application either as patient,doctor or admin.

2.3.2 Dashboard

After logging in, user searches for nearby hospitals and select the doctor you want to get appointment for. Before fixing appointment, user can see current token number, waiting list and ongoing visits. Doctor’s have the option to enable for revisiting a patient after preliminary tests are conducted. Late tokens are given another time to visit the doctor on the same day.

2.3.3 Administration

If admin is logged in, he / she has authority to make changes in the application, like editing hospital info, generating key password for doctor’s...etc

2.3.4 Doctor

If doctor is logged in, he / she has authority to give token limit,token call and missed token to be recalled later

2.3.5 Patient

If doctor is logged in, he / she can see the current number of bookings, current token number visits doctor, profile details...etc. An alarm system is present in application for notifying the user when his time is up for appointment. It is based on customization by user when or how it is should inform him at his / her time is up to meet the doctor.

3. Data design

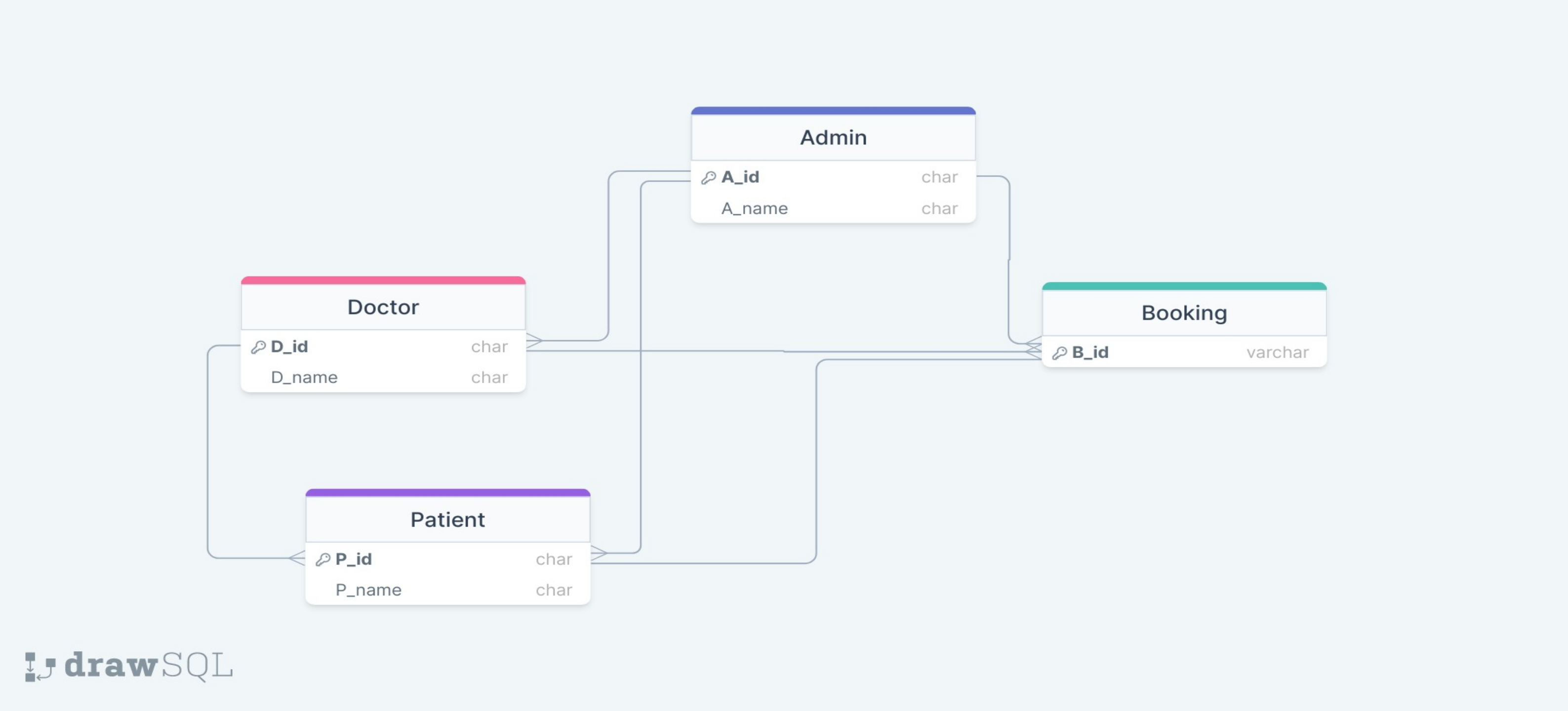
3.1 Data Description

This project stores different data related to a patients healthcare under which doctor they seek and the recorded data is stored in the hospital's database. The stored data can be retrieved both by patient and doctor whereas updation remains on doctor’s hands.

3.2 Data dictionary

Field name	Data Type	Description
P_id	Charfield	Unique id for Patient
P_name	Charfield	Name of Patient
D_id	Charfield	Unique id for Doctor
D_name	Charfield	Name of Doctor
B_id	Charfield	Unique Booking id

3.3 Data Storage and Access



- This application stores the data onto firebase.
- The database schema consists of five tables with columns for storing Admin's name, Admin's id, Patient's name, Patient's id, Doctor's id, Doctor's name,reason of visit and date of visit.

- This data stored on firebase application is queried and displayed on application's user interface.

4.Appendices

Appendix A: User Interface (UI) Design Mockups

This section contains mockups of the user interface design for the hospital token management system app. The UI design was created to ensure that the app is user-friendly, intuitive, and visually appealing.

[Insert UI design mockups]

Appendix B: Technical Specifications

This section contains technical specifications for the hospital token management system app. The specifications cover the programming languages, software tools, and hardware requirements needed to develop and deploy the app.

Programming Languages: Kotlin

Software Tools: Firebase

Hardware Requirements: Android devices (version 5.0 and above)

Appendix C: Data Management Schema

This section contains the data management schema for the hospital token management system app. The schema includes the database tables and their corresponding fields, data types, and constraints.

[Insert data management schema]

Appendix D: Testing Plan

This section contains the testing plan for the hospital token management system app. The plan outlines the testing scenarios, test cases, and expected outcomes.

Testing Scenarios:

1. User registration and login
2. Appointment booking and management
3. Queue management
4. Notification system
5. Data management and security

Test Cases: [Insert test cases and expected outcomes]

Appendix E: Glossary

This section contains a glossary of terms used in the hospital token management system app.

[Insert glossary]

5.Conclusion

In conclusion, the hospital token management system application is an essential tool for healthcare professionals and patients alike. This application streamlines patient flow, appointment scheduling, and queue management, which enhances hospital efficiency and patient satisfaction. The application's user-friendly interface and automated notification system make it easy for patients to schedule appointments and receive timely updates about their status. The application's data management and security features ensure that patient data is securely stored and compliant with relevant data protection regulations. Moreover, the application's reporting and analytics features enable healthcare professionals to track patient flow, identify areas for improvement, and optimize hospital management. In summary, the hospital token management system application is a valuable addition to any hospital's management system, as it offers convenience, efficiency, and improved patient outcomes.