UTD- Fall 2022

CS 6360 Database Design - Database Project

Phase 1

Team members

Ujwal Shah (USS220000)

Prabhleen Kaur Saini (PXS210087)

Deekshitha Juvvadi (DXJ220012)

Phase 1: Requirements Analysis (20%)

a) System Description

Introduction

This project outlines the proposed DBMS System for a Telemedicine company. Telemedicine is the delivery of health-related services and information using electronic data and telecommunications technologies. It permits remote patient and doctor communication, care, guidance, reminders, education, intervention, monitoring, and admissions. Telemedicine represents the future of medical consulting. The United States Department of Health and Human Services states that the term telemedicine means ‘remote clinical services’. The World Health Organization uses telemedicine to describe all aspects of health care including preventive care. Telemedicine may fill the gap when access to care is restricted by rural locations, a lack of transportation, a lack of mobility, conditions brought on by outbreaks, pandemic decreased funding, or a lack of staff in addition to offering distance learning, consultations, guidance, and meetings between practitioners and patients, internet resources and health data management, and healthcare system inclusion. There has also been a debate whether telemedicine will replace the original in-person medical consultation. Though that is a ridiculously huge topic which can be debated, we would assume in this report that the use of telemedicine will only improve the standards of healthcare system in a region. It supports in-person health consultation.

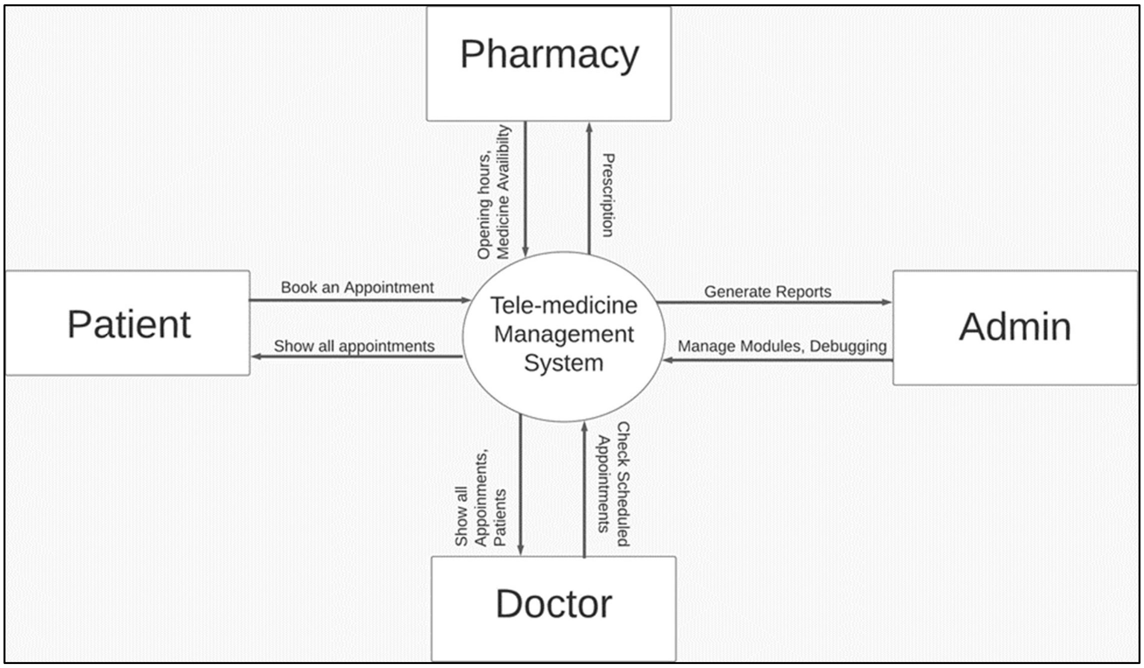
Purpose

The purpose of this project is to create a real life, efficiently working telemedicine consultation software. This telemedicine database management system project will help us to understand the whole architecture of Database Management System. This project will also provide hands-on experience with real world software systems. This project facilitates to design and implement both front end and back end of a database management system right from scratch for the database admin side to the naive user side.

Motivation

The COVID-19 pandemic has taught the entire world at least one thing. That is the topmost priority for any individual in their life is only and only their health and not anything else. People have now started to look up to their health in a quite unique way than they were looking earlier. Similarly, the way people now look at healthcare startups have also changed. According to Fierce healthcare, Digital health startups banked record-breaking $29.1B in 2021. Around $30 Billion was invested by investors to invest in Digital health startups. These numbers speak for themselves. This puts faith in the fact that people are now using, believing, investing, and giving telemedicine the importance of which it is due. Hence, such huge funding numbers, hundreds and thousands of startups emerging in this domain was the main motivation for us doing a telemedicine database project.

b) Context Diagram (System Architecture)


Pharmacy
Opening hours, Medicine Availibilty
Prescription
Patient
Book an Appointment
Show all appointments
Tele-medicine Management System
Generate Reports
Manage Modules, Debugging
Admin
Show all Appoinments, Patients
Check Scheduled Appointments
Doctor

c) Functional Requirements (User's Operational Concepts)

1. Login Functional Requirements:

   1.1. The system will allow the user to log in.
   1.2. The system will verify the username and password.
   1.3. The system will not allow the user to log in with an invalid username/password.
   1.4. The system will be able to remember usernames and passwords.
   1.5. The system will allow users to create accounts.
   1.6. The system will enable users to log out of their account.

2. Browsing Functional Requirements:

   2.1. The user (patient) can see the different doctors and their reviews.
   2.2. The user (patient) can book appointments with doctor on their available timings.
   2.3. The doctor can see their scheduled appointments and patient list.
   2.4. The doctor can process the appointments. i.e., can cancel, accept, and reschedule the appointments based on their availability.
   2.5. The user (patient) can search for a pharmacy where medicine is available by entering the medicine name.
   2.6. The user (patient) can store their medical records in their profile.
   2.7. The user (patient) can view their past actions through history.
   2.8. The doctor can send suggestions and prescriptions to the user against the scheduled appointment.

2.9. The user (patient) can rate the doctor based on their satisfaction with the appointment.
2.10. The user (patient) can search for nearby available pharmacies.
2.11. The system provides a help and support feature where users (patients, doctors, pharmacists) can raise any concerns they have with the website such as issue booking appointments, unable to open the website etc.
2.12. The doctor can view the list of payments in the payment records.
2.13. The user (patient) can process the payments through the payment module of the system
2.14. The patient can select the services they want as a part of the appointment reservation.
2.15. The doctor can view the list of payments in the payment records.
2.16. List of services are included in the billing statement with their corresponding amount or service fee.
2.17. The system users can process and verify the payment transactions.
2.18. The user (patient) can only process a reservation one at a time.

3. Administrator functional requirements

3.1. The new users (patients) and doctors can be added or removed based on their availability.
3.2. The new pharmacies can be onboarded and any updates to existing pharmacies can be managed.
3.3. The issues raised by users can be viewed and addressed.
3.4. The user information can be maintained and updated in their profiles.
3.5. The doctor license information can be maintained and updated.
3.6. The reports related to user(patient) appointments, doctor appointments and schedules can be generated.
3.7. The user medical records should be maintained.
3.8. The pharmacy information should be maintained and managed according to pharmacist updates.
3.9. The ratings and reviews for doctors should be maintained.
3.10. The prescriptions and comments for users(patients) from the doctor need to be maintained for users to access.

d) Non-functional requirements (e.g., response time, maintainability)

1. Security:
    - A user can do concurrent login in only 1 device.
    - The password should have at least 1 Uppercase, 1 Lowercase, 1 Number, & 1 Special Character.
    - The password length should be from 4 characters to 15 characters.

2. Usability:

- The maximum number of tablets for one medicine which can be placed at the pharmacy is less than equal to 5.
- A patient cannot book more than one appointment at the same time.
- There is no overlapping of appointments at the doctor’s dashboard.

3. Maintainability:

- The system requires less frequency of revisions as the main modules of the system are consistent and would require only enhancements in future.
- The system is based on modules therefore it would require less resources to maintain the system.

4. Scalability:

- The system can be scaled up in future to allow large volumes of operations.
- In case of increase in traffic, the system could be deployed in the cloud.

5. Speed:

- The speed of the system would be high as it uses an optimized database system.
- The system will have low response time as it processes information from structured data in quick time.

6. Reliability:

- The system data will be highly dependable as updates/modifications made in the system will be real time.
- The system will have session-based login and logout features and will have a smaller number of critical failures.