Software Prototype Evaluation

CS 345-346

Calendar Application

Group 13

Name	Roll Number
Tanay Maheshwari	190101092
Vignesh Ravichandra Rao	190101109
Tattukolla Lokesh	190101094
Shreyansh Meena	190101084

Contents

1 Introduction

1.1 Purpose

This Medical Assistance application is a telemedicine application which assists in the remote delivery of healthcare services like consultations. This Design DFD document is a part of the project under CS 346 - Software Engineering Lab. The intended audience is our course instructor, Dr Samit Bhattacharya, and the teaching assistants. This document provides description of the design, also shows how the functions described in the SRS can be implemented

1.2 Product Scope

The COVID-19 has shown the world the importance of telemedicine. More and more people have started to prefer short online consultations over physically visiting clinics. This application allows healthcare providers to evaluate, diagnose and treat patients without the need for an in-person visit. The application aims to connect doctors and patients in an easy-to-use and convenient manner.

1.3 Overview

The remaining part of the Design document contains:

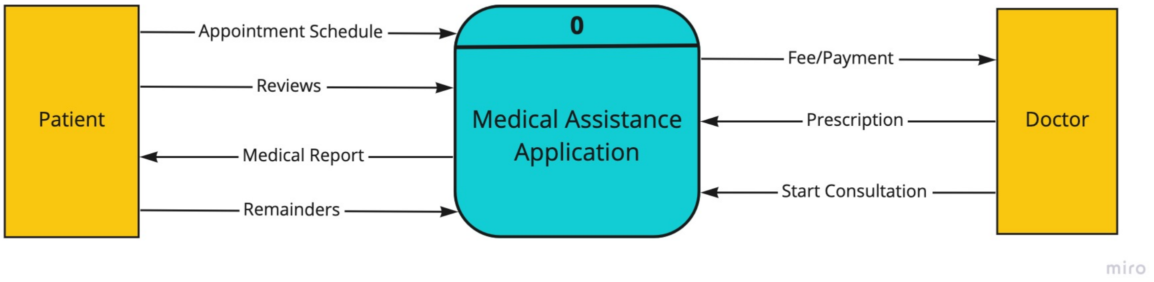
1. Data flow diagrams
2. Data Design and Description
3. Design Justification

1.4 References

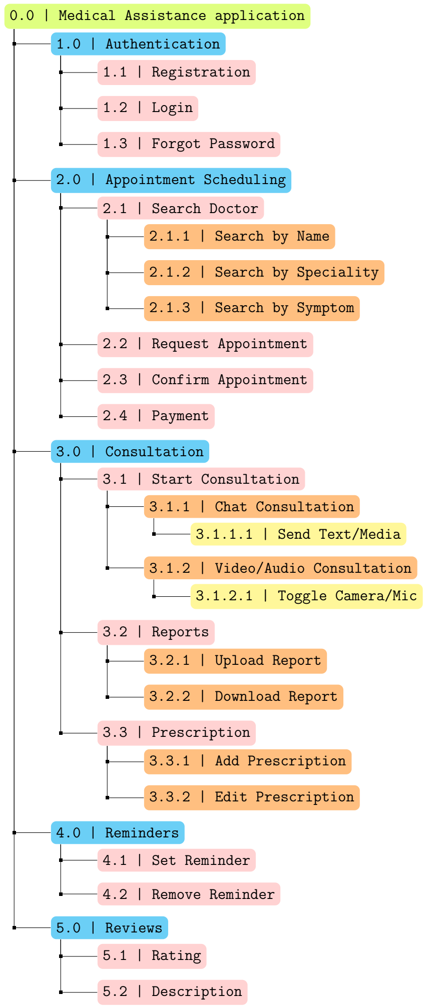
- CS345 Lecture Slides by Dr. Samit Bhattacharya.
- Fundamentals of Software Engineering, Rajib Mall

2 Data flow diagrams

2.1 Context Diagram: Level 0



2.2 Overview Diagram: Level 1



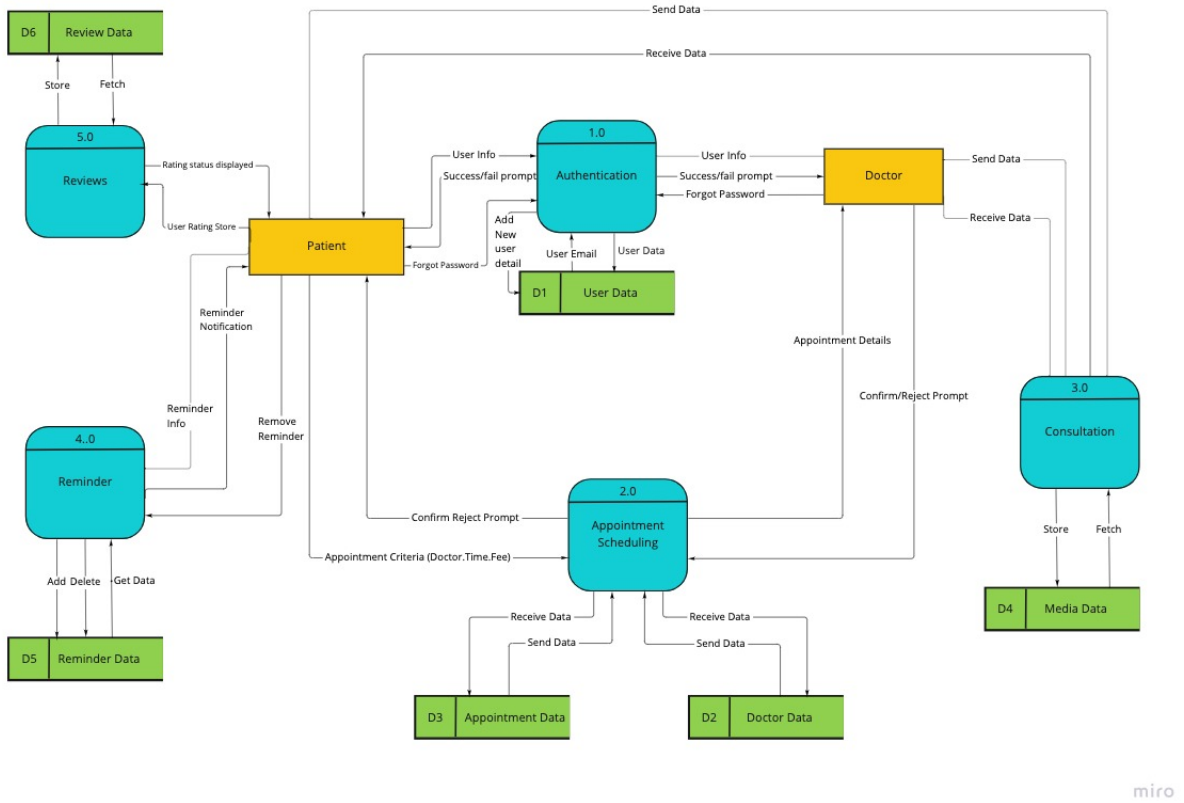
Our entire application can be divided into 5 modules:

1. Authentication (Managing Users): Handles the creation of user accounts and user authentication to access data and cases of forgetting password.
2. Appointment Scheduling: This module helps the patient to find the appropriate doctor and request for appointment and doctors can confirm the appointment based on feasibility and also handles payment of fees.

3. Consultation: This module handles the media (text, pictures, reports, prescription, video/audio call) communication between the doctor and the patient.

4. Reminders: This helps in handling the reminders for users s to take their medicine on time.

5. Reviews: This module handles the review section which is given by the patients after diagnosis by the doctor.



2.3 Level 2: Detailed Diagrams:

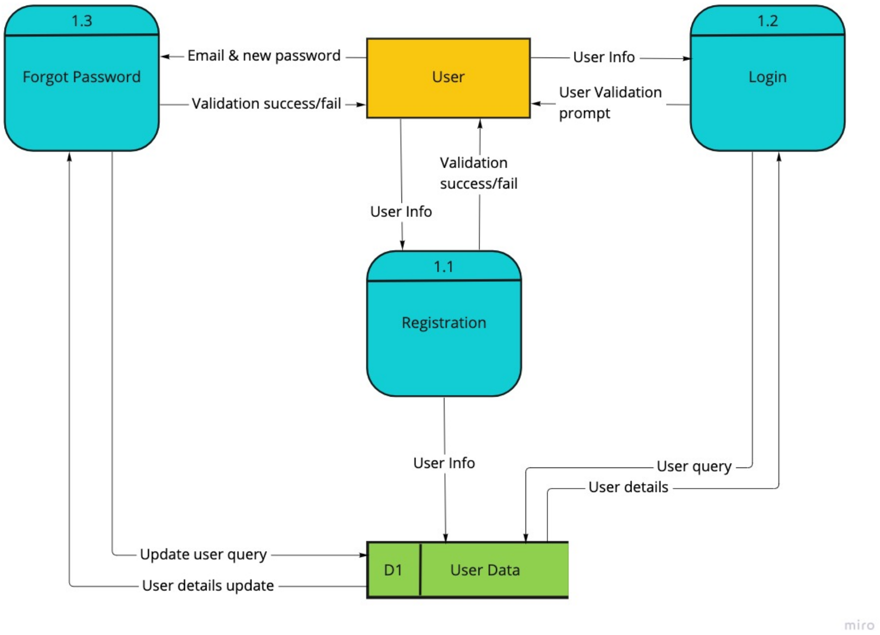
2.3.1 Level 2 Module 1: (Authentication)

In this module we have 3 processes:

1. Registration: User enters their details. Based on the validity of entries new account is created or an error message is displayed

2. Login: On successful authentication, the user is logged in.

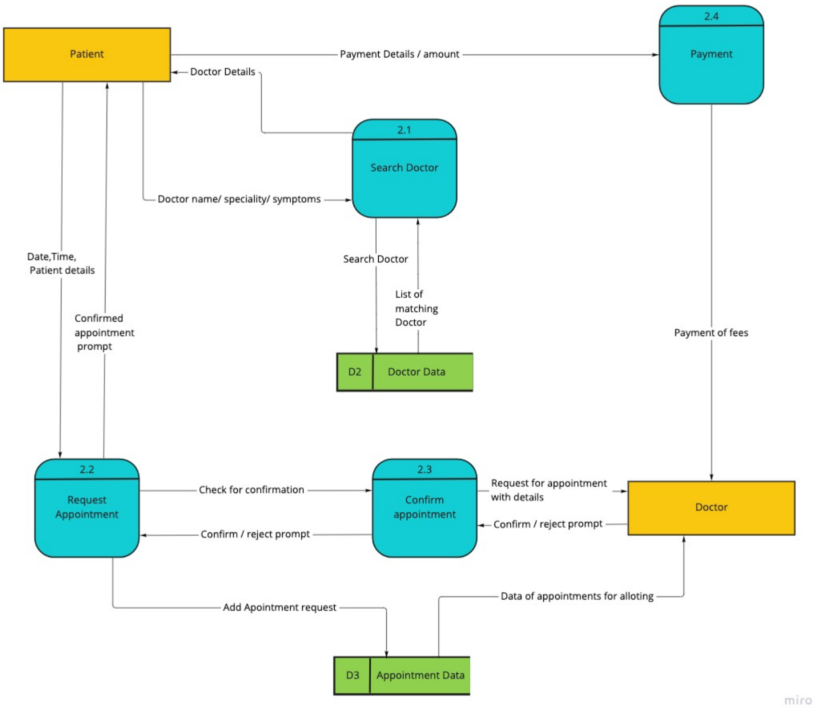
3. Forgot password: Update the user password on verifying email



2.3.2 Level 2 Module 2: (Appointment Shceduling)

In this module we have 4 processes:

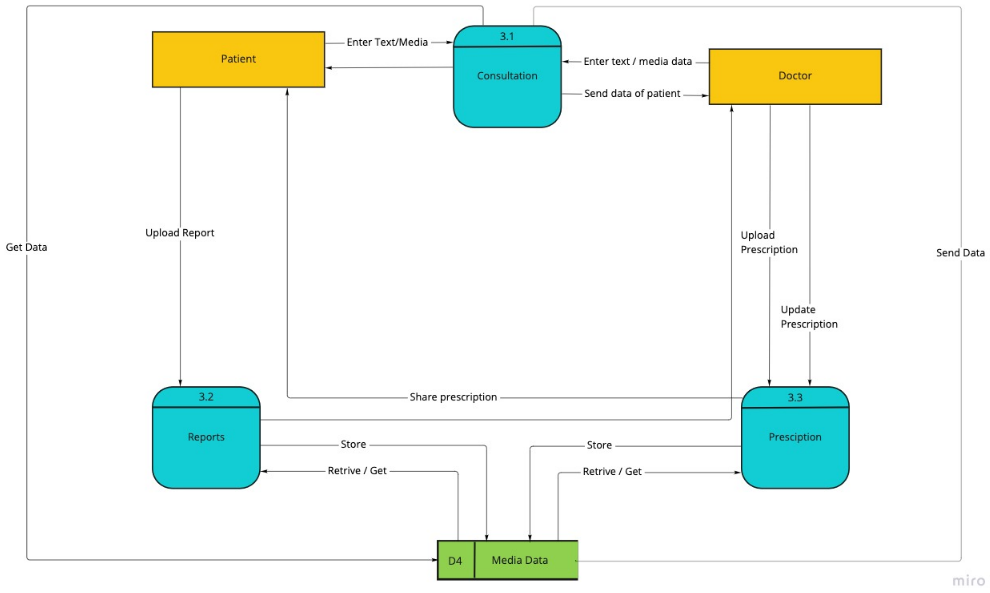
1. Search Doctor: Filters the list of doctors based on their speciality/name/common symptoms
2. Request Appointment: Users can request an appointment at a specified date and time with the available doctor.
3. Confirm Appointment: Doctors can confirm or reject appointment request from patients
4. Payment: Users pay their fee online to doctor



2.3.3 Level 2 Module 3 (Consultation)

In this module we have 3 processes:

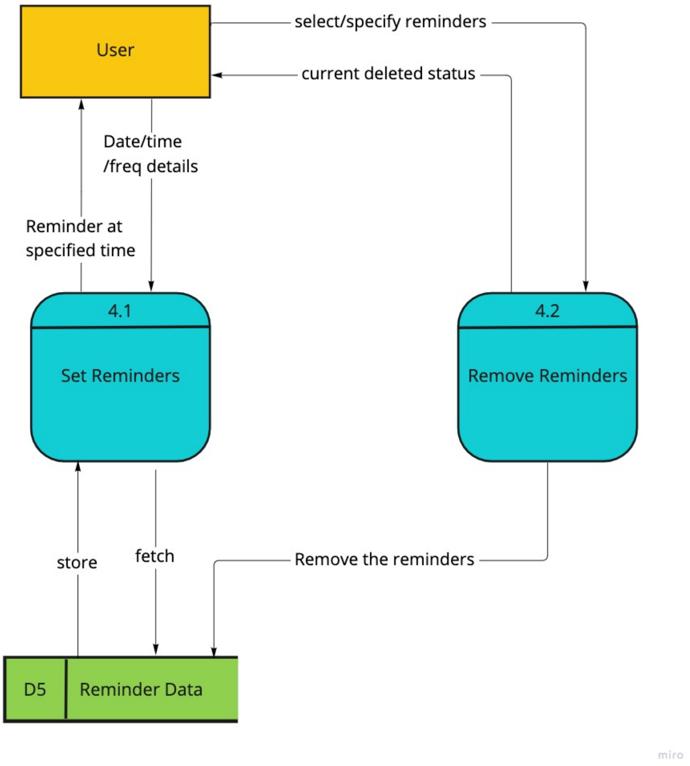
1. Consultation; facilitates two way chat communication between user and patient through text and media
2. Reports: Doctors can download reports of patients for medical examination
3. Prescription: Doctors can write and edit prescription and upload it for patients



2.3.4 Level 2 Module 4 (Reminders)

In this module we have 2 processes:

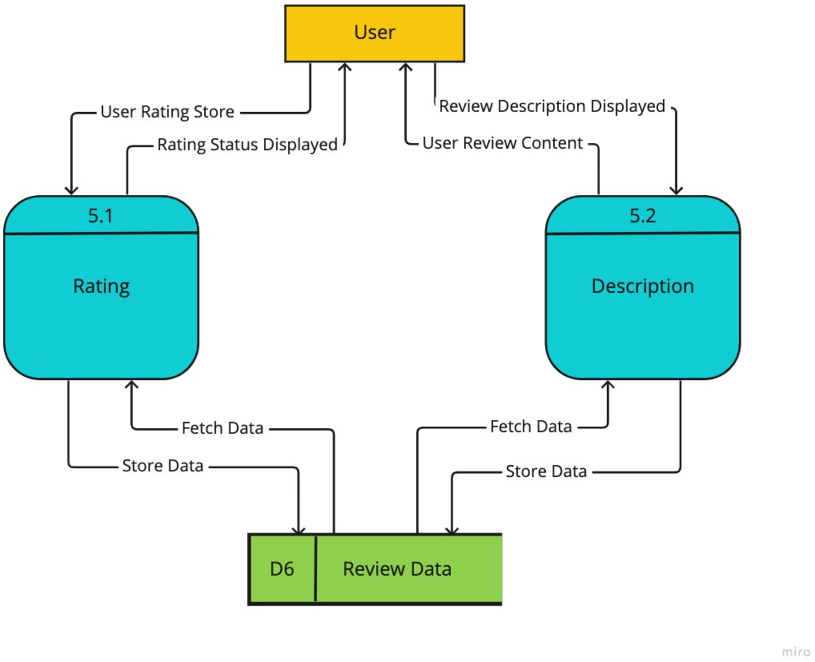
1. Set Reminders: Patients can set reminders to take their medicine on time.
2. Remove Reminders: Delete a reminder that is no longer required.



2.3.5 Level 2 Moudule 5: (Reviews)

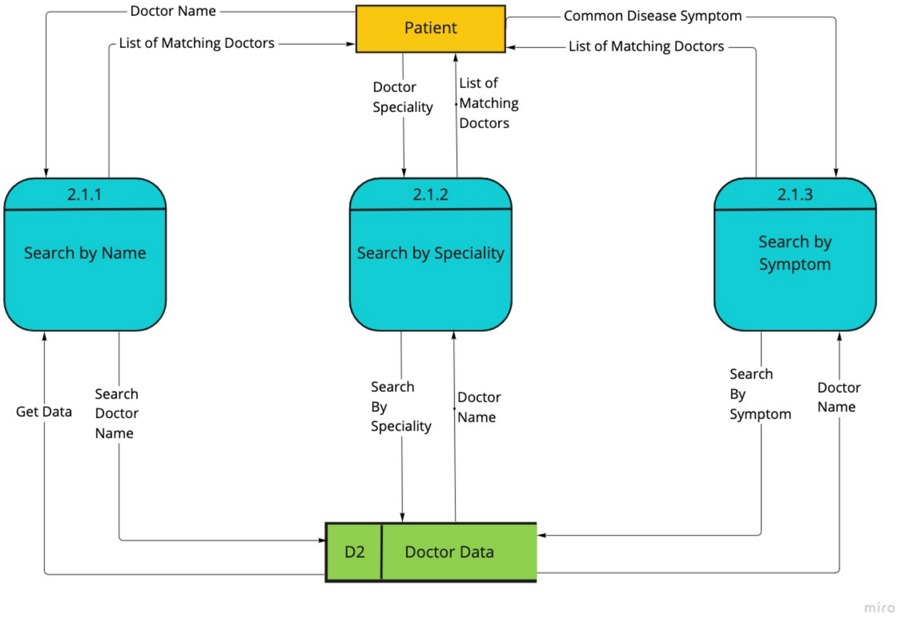
In this module we have 2 processes:

1. Rating: patients can give rating in stars/points to doctors after consultation
2. Description: Patients can write their consultation experience for other patients to see



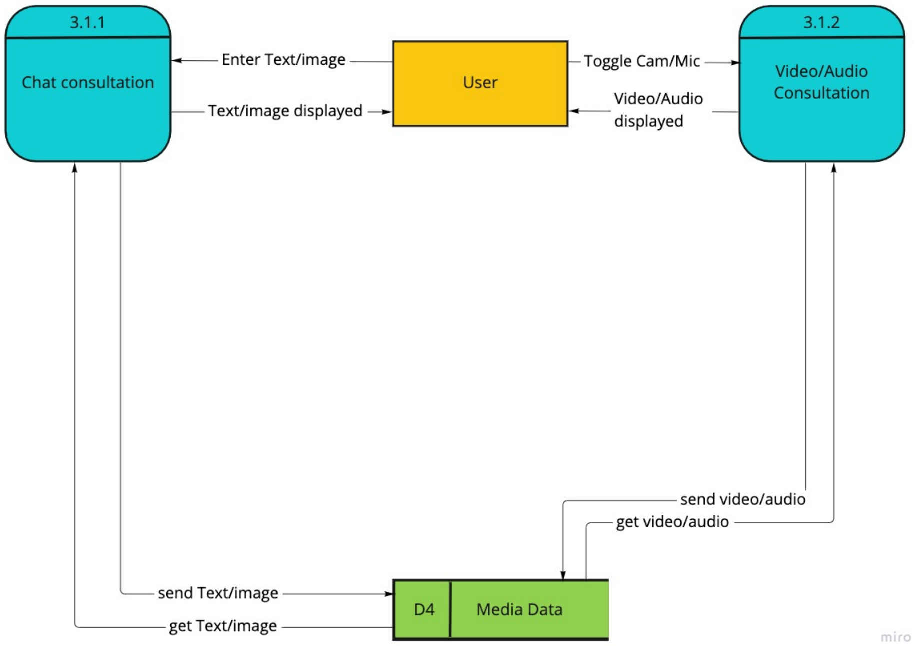
2.4 Level 3 Process (Search Doctor) of Module 2:

1. Search By Name:Filters the list of doctors based on their name
2. Search By Speciality:Filters the list of doctors based on their speciality (like cardiologist /dermatologist)
3. Search By Symptoms: Filters the list of doctors based on the symptoms entered by patients (like nausea/headache)



2.5 Level 3 Process (Consultation) of Module 3:

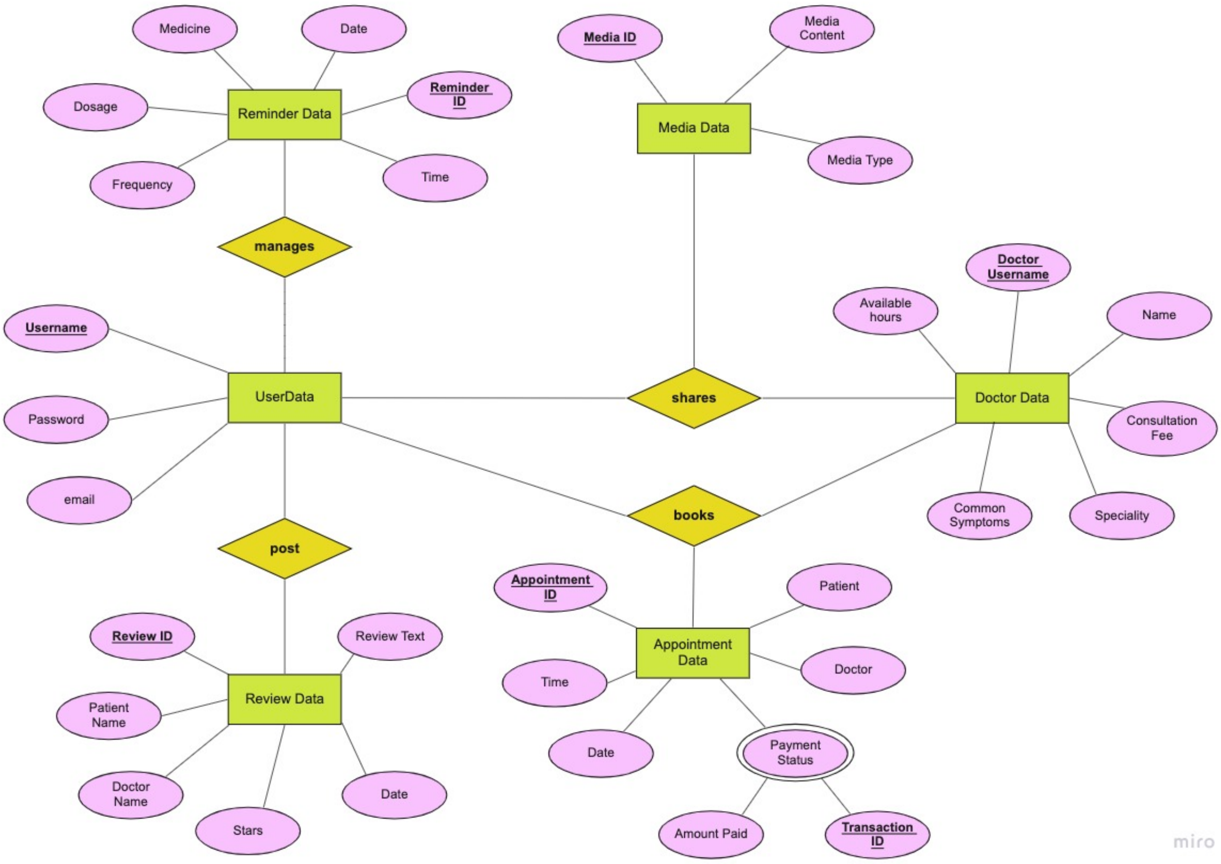
1. Chat Consultation: Facilitates two way chat communication between user and patient through text and images
2. Video/Audio Consultation: Turn camera/mic on if it is currently switched off and vice versa for video or consultation



3 Data Design and Description

3.1 ER Diagram

We have provided the Entity-Relationship Diagram, which helps in expressing the internal structure, organization and relationships in data stores. The basic components in ER Diagram are entity (rectangle), attribute (oval) and the relationships between these(diamond)



3.2 Data Dictionary

3.2.1 User Data

It stores the credentials of every user who uses this application and assists in managing access. Attributes:

- **Username**: Unique alphanumeric for every user of the app
- **Password**: Secret pass code to authenticate the user
- **Email**: Email address of the user

3.2.2 Doctor DATA

: It stores the information about the doctor like:

- **Name**: doctor name
- **Doctor username**: user name in the app for the doctor
- **Consultation fee**: Fee amount paid to the doctor for diagnosing
- **Available hours**: Timings during which doctor will be available
- **Speciality**: branch of medical practice that doctor specialises
- **Common symptoms**: Symptoms of the disease in which doctor specialises

3.2.3 Appointment data

: It stores the appointment details like which doctor, time, fee amount and patient details. The following are the attributes:

- **Patient**: Username of the patient
- **Doctor**: username of the doctor
- **Appointment id**: Identifier used in the app for managing the appointment
- **Time**: time of the appointment
- **Date**: Date of the appointment
- **Payment status**:
  - Amount paid: The fee paid by the patient
  - Transaction Id: Id of the transaction for the transfer of the amount from the patient to the doctor

3.2.4 Media data

It acts as buffer by storing the data to be of the user's and helps mediating between the doctor and the patient. The attributes are the follwoing:

- **Media Id**: Unique Identifier for the media content
- **Media Content**: Content of the media
- **Media Type**: Identifies the type of media which can be a report, prescription, video, audio, image or text

3.2.5 Reminder data

It stores the data related to the reminders created byt the user, the following are the attributes:

- **Date**: Date(s) for which reminder is set
- **Frequency**: No.of times the reminder should work
- **Time**: Time(s) of the day at which reminder is set
- **Reminder id**: Identifier for the reminder set
- **Medicine**: The medicine(s) to be taken
- **Dosage**: Dose of the medicine prescribed

3.2.6 Review data

It stores the information related to the reviews posted by the users, the following are it's attributes

- **Review Id**: Identifier for the review posted by the user
- **Patient name**: Name of the patient who posted review
- **Doctor name**: Name of the doctor on whom the review was posted
- **Stars**: No.of stars given to the doctor by the patient
- **Review Text**: Description of the review (comment) by the patient
- **Date**: The day on which review was posted

4 Design Justification

The Medical Assistance application is decomposed into 5 modules-Authentication, Appointment Scheduling, Consultation, Reminders, Reviews. Now let us see the cohesion-coupling analysis between these modules

4.1 Cohesion of Modules

4.1.1 Module Authentication

- This module has three processes: Registration, Login, Forgot Password. All the three processes deal with the same datastore - User Data.
- Registration creates a new entry in User Data, Login refers existing entries in User Data to provide access to the account and Forgot Password changes the username and/or the password in the entry using email, So we can see there is **Communication** between functions in the module.
- **Logically** all the three perform similar functionality of managing the users in the application and when a new user joins both Registration and Login occur at the same time so **Temporal**.

4.1.2 Module Appointment Scheduling

- This module has 4 processes: Search Doctor, Request Appointment, Confirm Appointment, Payment.
- We can **sequential** processing between Request Appointment and Confirming the Appointment process and both using the same data store showing **Communication** between them in the module.
- All the processes belong to the same **procedure** for scheduling the appointment.

4.1.3 Module Consultation

- This module has three processes: Consultation, Report and Prescription.
- We can see **communication** between the modules as all the process use the same Media Data store.
- All the procedures perform similar functions (communicating through various media) hence **Logical**.

4.1.4 Module Reminder

- This module has 2 processes: Set Reminders and Remove Reminders.
- We can see **communication** as both use the same data store, also it's **Logical** as they perform process related to managing reminders.

4.1.5 Module Review

- This module has 2 processes: Rating and Reviews.
- We can see both process belong to the same procedure so **Procedural**, and both the processes use the data store showing **communication**.
- Also both of them perform functions with similar operations so **logical**.

4.2 Coupling between Modules

1. Module Authentication: This module is independent of all other modules, but other modules like Review are dependent on it.
2. Module Appointment Scheduling: This is **data** dependent on the Authentication module, independent of others.
3. Module Consultation: This module is **control** dependent on Appointment Scheduling module and **data** dependent on both Authentication and also Appointment Scheduling module, it's independent of others.
4. Module Reminder: This module is **data** dependent on Consulataiton and Authentication modules.
5. Module Review: This module is **data** dependent on Authentication module and independent of others.

We tried to make the modules more cohesive and see that there is less coupling between them for ensuring better design.