IITJ Health Care

Software Engineering & Web Development Project

THE SOLUTION

- Online Healthcare System for IIT Jodhpur Health Centre connecting Students and Residents with Doctors.
- Online prescriptions availability information.
- Notifying users by suggestions from time to time.

Project Features

- User Profile and Information Portal
- Login and Registration Portal
- Prescription History And Portal
- Health Centre Contact Info
- Prescription and Doctor Specifics
- Emergency Portal
- Small Scale Medicine Store

The technology :

- Django Framework - Python
- SQL
- HTML | CSS | Bootstrap
- Javascript

IDEATION
+
REPORT SUBMISSION

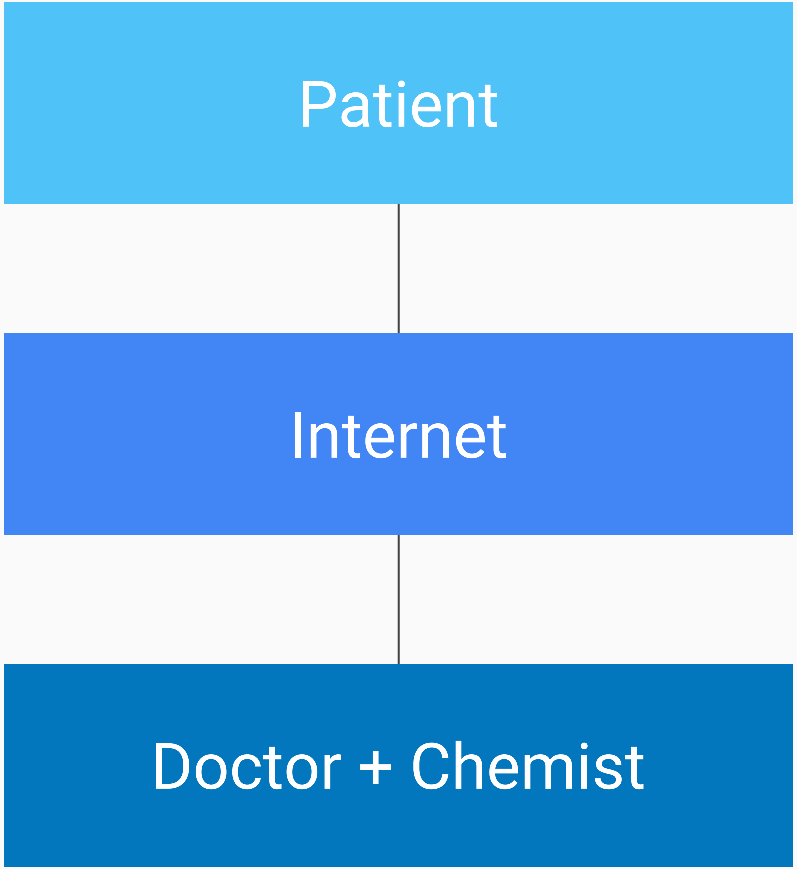
PROBLEM IDENTIFIED

- Need of assistance and monitoring of Healthcare in Hostel life as number of students in IIT Jodhpur will rapidly increase in coming years so management will become difficult.

- Making the process automated and online can increase the efficiency of the Health Centre as well as reduce the need of paperwork.

USER CASES

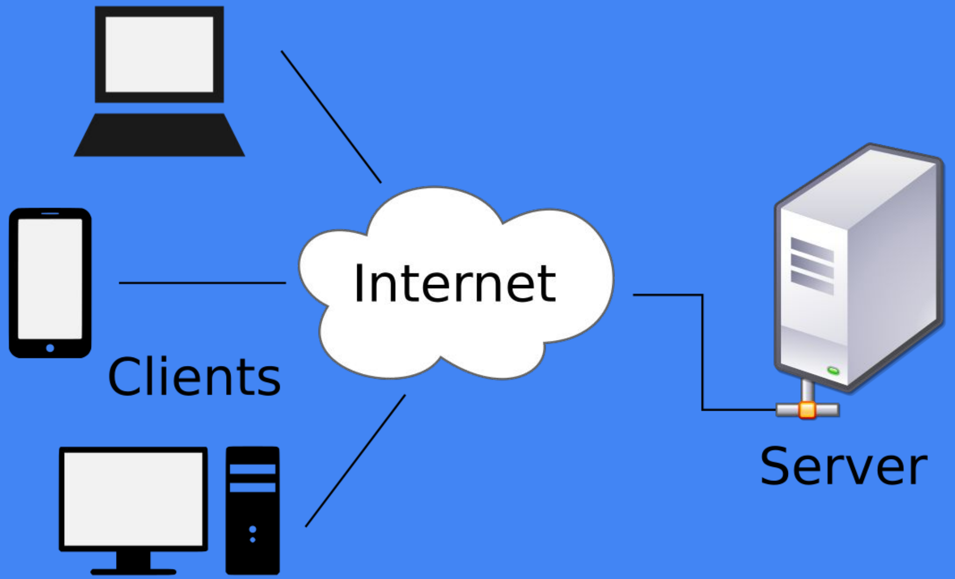
- Interface for interaction of doctor and patient through internet on appointment.
- Access to prescriptions and patient history.
- Online medicine availability portal.
- Reminders for expiry and consumption of medicines at regular intervals.



ARCHITECTURAL DESIGN

CLIENT - SERVER ARCHITECTURE

PATIENTS
ACTING AS
INDIVIDUAL
CLIENTS



HEALTH CENTER
HAVING ALL
RECORDS OF
MEDICAL DETAILS
OF PATIENTS

4+1 VIEWS

- LOGICAL
- PROCESS
- DEVELOPMENT
- PHYSICAL

+

USE CASE SCENARIO

LOGICAL VIEW

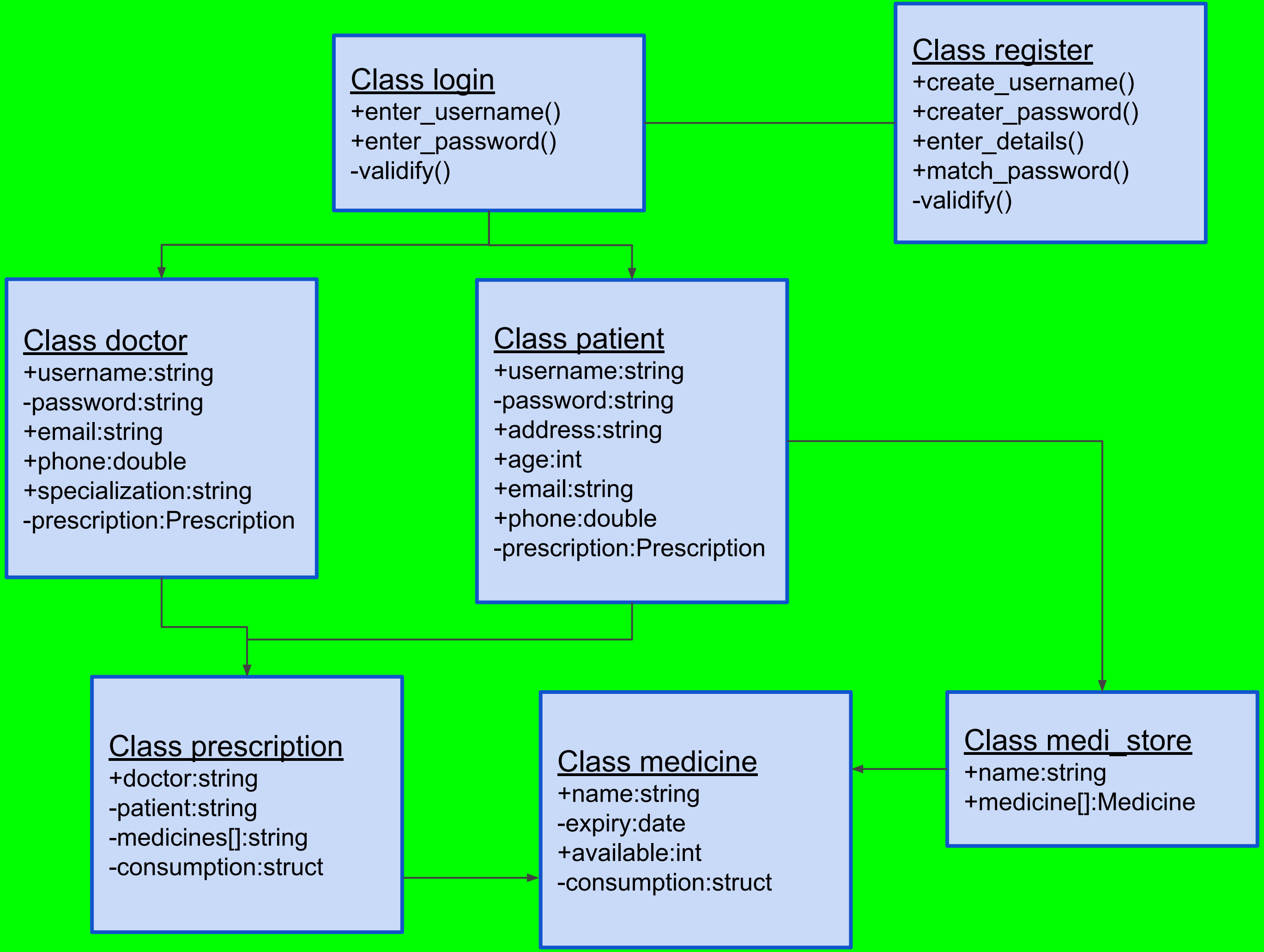
Class diagram
Describes the object model of the design.
Shows -

- Main functionality
- End user point of view

Our Model -

- Login
- Register
- Doctor
- Patient
- Prescription
- Medicine
- Medi_Store

LOGICAL VIEW



Class login
+enter_username()
+enter_password()
-validify()

Class register
+create_username()
+creater_password()
+enter_details()
+match_password()
-validify()

Class doctor
+username:string
-password:string
+email:string
+phone:double
+specialization:string
-prescription:Prescription

Class patient
+username:string
-password:string
+address:string
+age:int
+email:string
+phone:double
-prescription:Prescription

Class prescription
+doctor:string
-patient:string
-medicines[]:string
-consumption:struct

Class medicine
+name:string
-expiry:date
+available:int
-consumption:struct

Class medi_store
+name:string
+medicine[]:Medicine

PROCESS VIEW

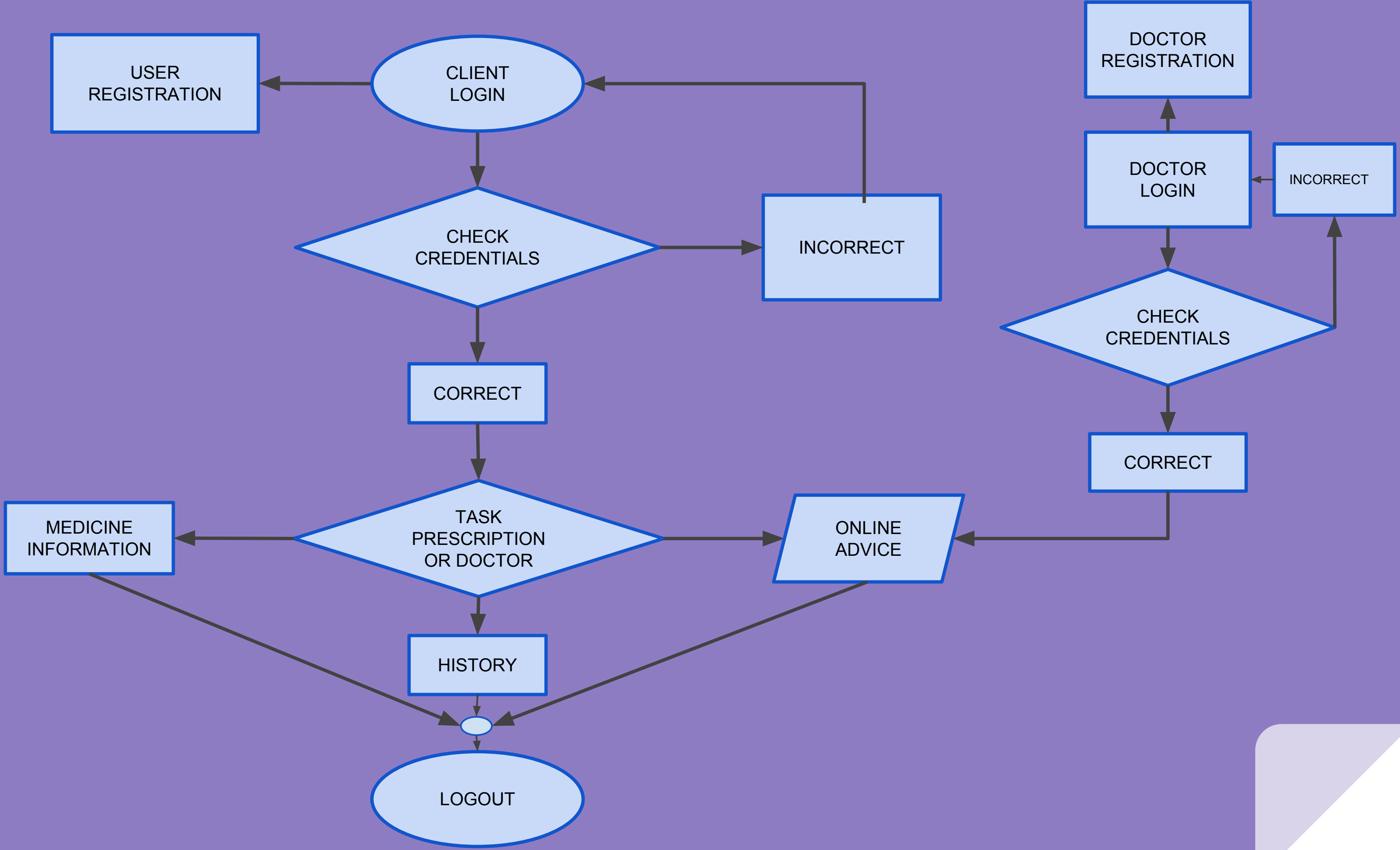
Activity diagram
Describes the activities of the system, captures the concurrency and synchronization aspects of the design.
Shows -

- **System functionality**
- **Performance**
- **Scalability**

Our Model -

- **Login**
- **Task - Advise/Medication**
- **History**
- **Logout**

PROCESS VIEW



DEVELOPMENT VIEW

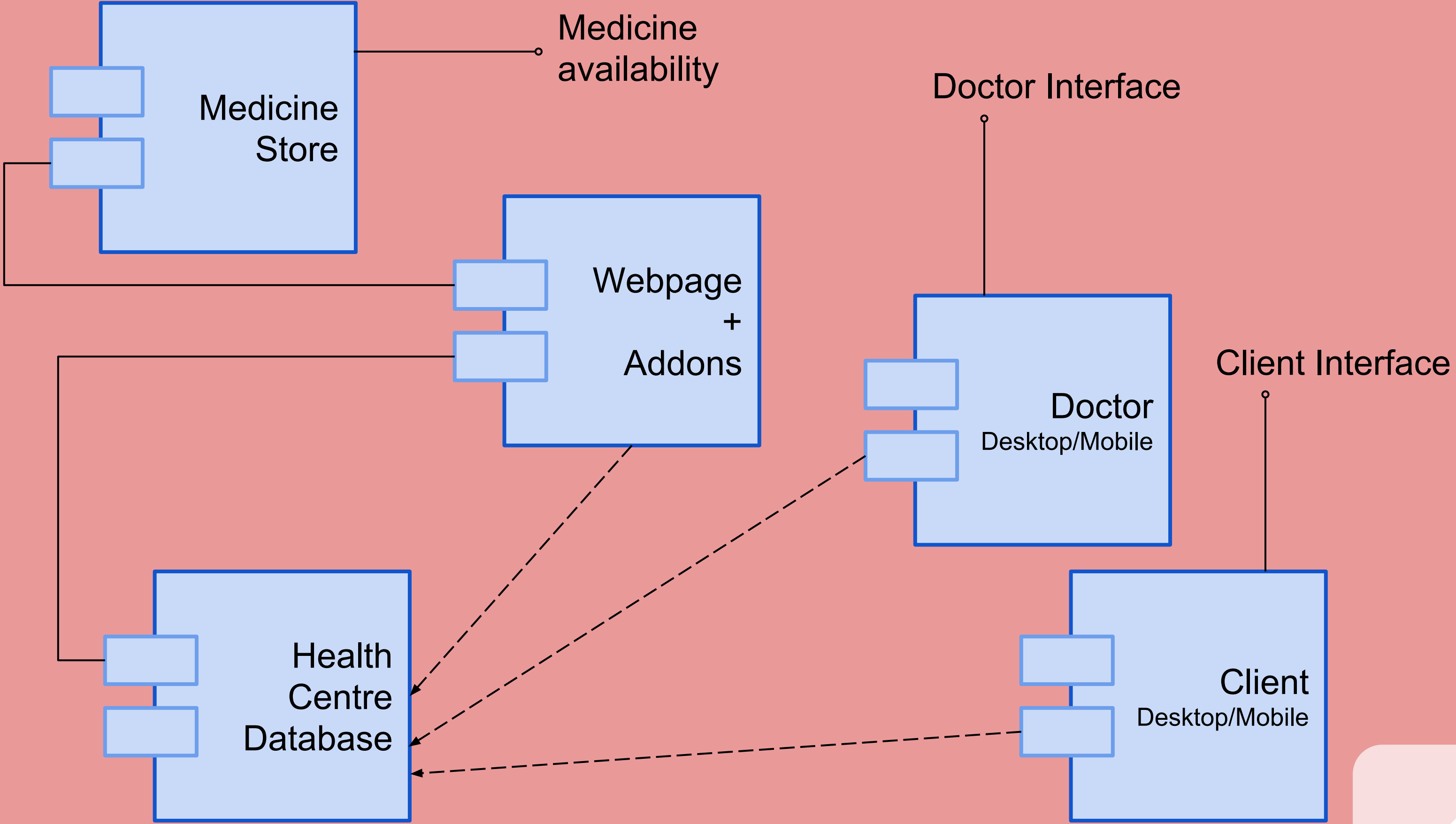
Component diagram
Describes the static organization or structure of the software in its development of environment.
Shows -

- **System Specification**
- **Decomposition**

Our Model -

- **Medicine Store**
- **Database**
- **Webpage**
- **Addons**
- **Client**
- **Doctor**

DEVELOPMENT VIEW



PHYSICAL VIEW

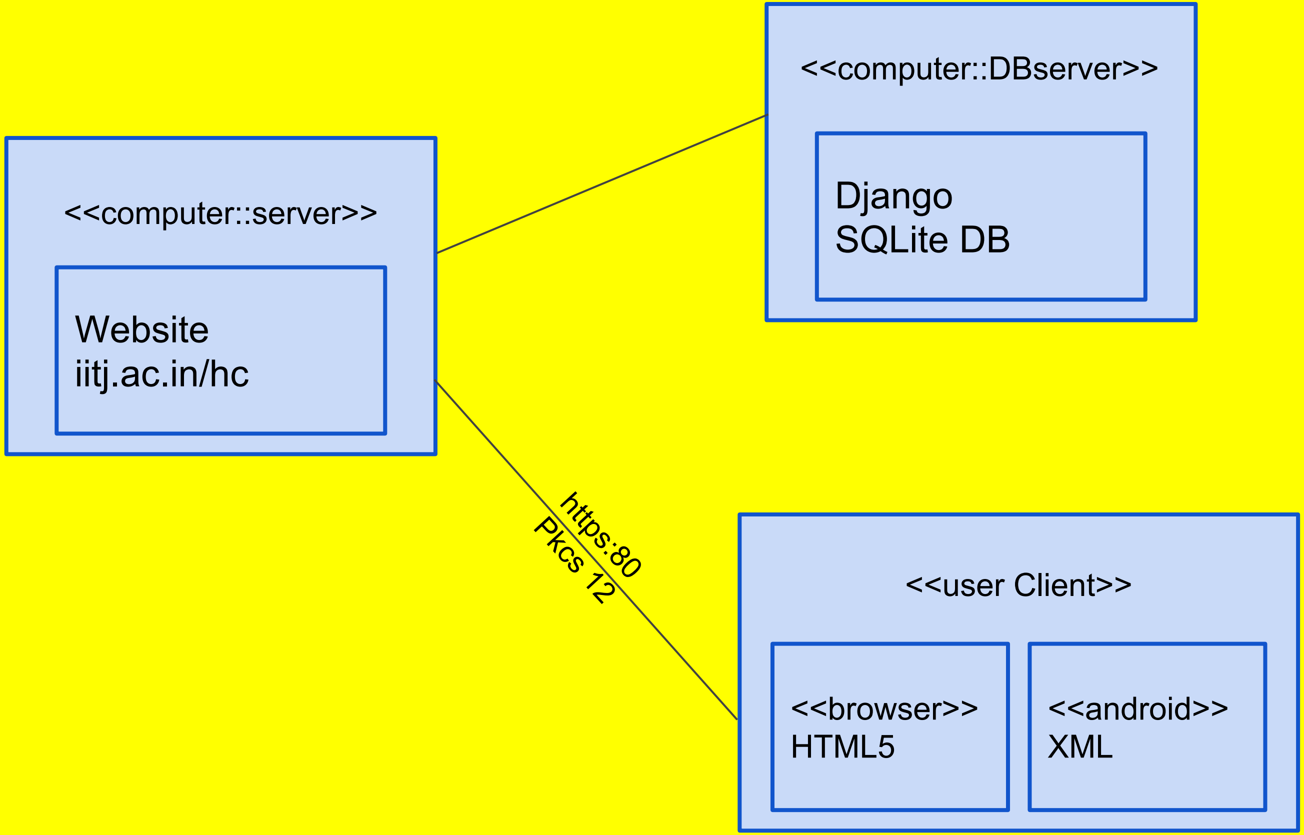
Deployment diagram
Describes the mapping of software onto hardware and reflects its distributed aspect.
Shows -

- **Product Topology**

Our Model -

- **User**
- **Interface**
- **Database**

PHYSICAL VIEW

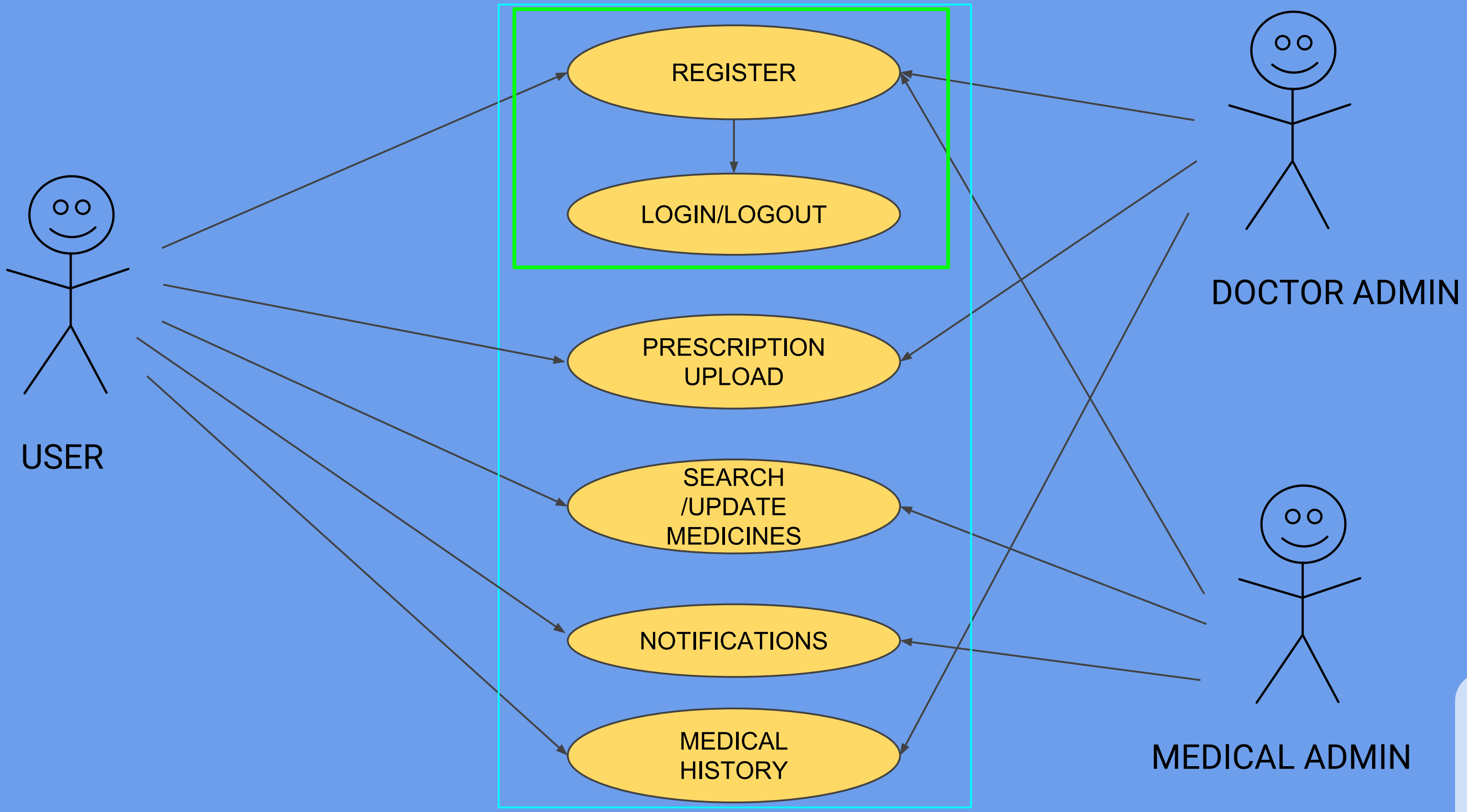


USE CASE SCENARIO

Describes the relationships among the functionalities and their internal/external controllers

-- <u>Actors</u> --
User
Doctor Admin
Medical Admin

USE CASE SCENARIO



Dependability properties

1
Availability

The Ability of a system to deliver services when requested

- RUNNING ON TWO DATABASE AND MULTIPLE SERVERS SO IF ONE COLLAPSES OTHER CAN WORK

2
Reliability

The Ability of a system to deliver services as specified

- GIVES THE INFORMATION OF MEDICINES AND PRESCRIPTIONS AND OTHER DATA.

3
Safety

The Ability of a system to operate without catastrophic failure

- USERS HAVE ACCESS TO THEIR DATA ONLY.

4
Security

The Ability of a system to protect itself against intrusion

- USING STANDARD ENCRYPTION TECHNIQUES FOR ENCRYPTION SO THAT USER DATA IS KEPT SAFE UNDER HIS/HER ACCESS ONLY

5
Resilience

The Ability of a system to resist and recover

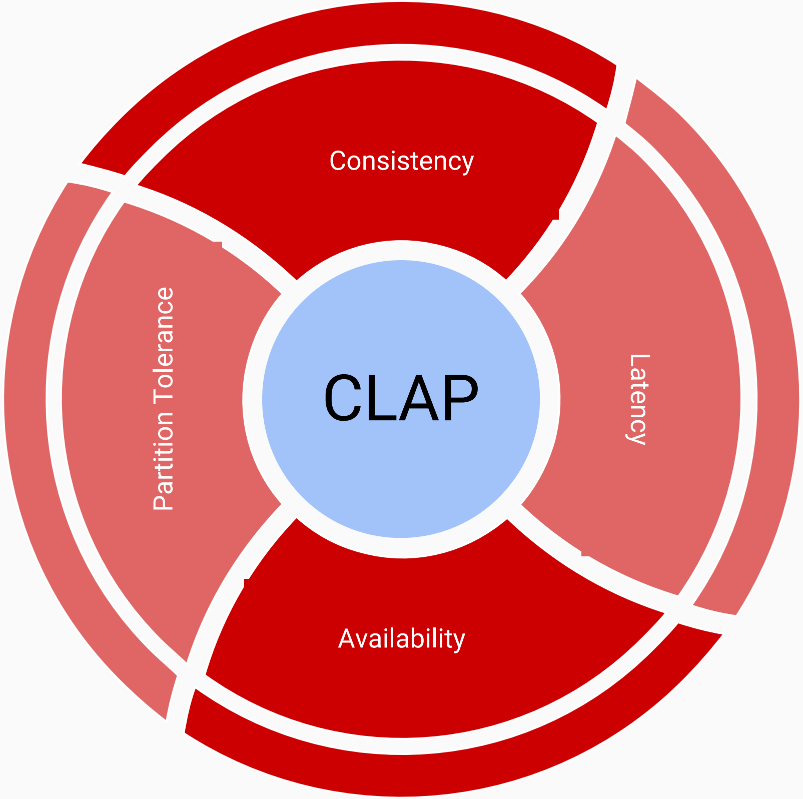
- Backup data to cloud

How we achieve dependability?

- Redundancy: Keep more than 1 version of a critical component available so that if one fails then a backup is available. Can be implemented with our 2 databases.
- Diversity: Provide the same functionality in different ways so that they will not fail in the same way. Can be implemented where we provide access and notifications to clients for prescription.

How we achieve dependability?

- Avoid accidental errors when developing the system.
- Design systems to be fault tolerant so that they can continue in operation when faults occur.
- Protection mechanisms that guard against external attacks.
- Configure the system correctly for its operating environment.
- Recovery mechanisms to help restore normal system service after a failure.



CONSISTENCY
Our consistency is somewhat what is a combination of what bank system and ecommerce systems have.

LATENCY
Our project has minimum latency as all the requests are within the college network itself which can ensure fast processing of requests.

AVAILABILITY
Multiple servers in computer centre and health centre.

PARTITION TOLERANCE
Our software is somewhat less partition tolerance as the medicine store and user related data can still be read in case a server disconnects from the main system but emergency situations can create a hurdle.

Why healthcare?

- Healthcare cannot be neglected and should be given utmost importance by students.
- This project might be helpful in case of emergency as well as daily medical needs of the patient.