INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT AKURDI, PUNE

Documentation On

“HELLO DOCTOR”

e-DAC AUG 2019

Submitted By:

Group No: 28
J K BIRAJDAR (PRN: 210541281029)
ANURAG NANDKISHOR KANOJE (PRN: 210541281081)

Prashant Karhale	**Mr. Milind Arjun**
Centre Coordinator	**Project Guide**
IACSD, AKURDI	**e-DAC**

Table of Contents

1. Introduction 1

Document Purpose 4

Problem Statement 4

Product Scope 5

Aim & Objectives 5

2. Overall Description 5

Product Perspective... 5

Benefits of Hello Doctor project ……….. 6

User and Characteristics 7

Operating Environment 8

Design and Implementation Constraints 8

3. Requirements Specification 9

External Interface Requirements 9

Non-Functional Requirements.. 9

4. System Diagram 11

Activity Diagram 11

Data Flow Diagram 13

Use Case Diagram 17

ER Diagram 18

5. Table Structure 19

Admin table.. 19

Appointment table 19

Blood donor table 20

Doctor table 20

Doctor time table available slo.. 21

Doctor time table holidays.. 21

Doctor timetable table... 21

Patient table.. 22

6. Conclusion 23

Future Scope 23

7. References 24

List of Figures

Activity Diagram 11

Data Flow Diagram 13

Use Case Diagram 17

ER Diagram 18

1. Introduction

Hospitals are very case sensitive areas specially in this pandemic situation. People should not make crowd at hospital places, because of this pandemic hit very bad on the society. To make the process entirely computerized so as to help people accordingly for the convenience of people in real time. The purpose of this project to make the entire hospital system systematic and using web application so that people can get benefit of it and stop their wastage of time in hospitals and no need to suffer during the emergency.

Document Purpose

The advancement in Information Technology and internet penetration has greatly enhanced various business processes and communication between patients and doctors. This Hello Doctor is developed to provide the following services:

Enhance Hospital Processes:

To be able to use internet technology to project to the global world instead of limiting their services to their local domain alone, many hospitals and doctors can use this so as to make the processes easy simple and convenient.

Online Appointment of doctor:

People can take online appointment from any location by searching and by checking the doctors available time slots.

Search for blood donor :

Blood donors registered by admin in this project so as to avoid any false information and facility to search area and city wise available blood donors of registered blood donors to help the needy during the emergency.

Problem Statement

Existing System are not that matured to provide various facilities. Existing systems are specifically designed for particular hospitals and clinics for their own use. Existing system doesn’t provide blood donor related information. Systems are not so convenient and easy to use. To take the appointment and consultation from another one can need to visit the that doctor’s own platform which is sort of irritating thing.

Product Scope

This project traverses a lot of areas ranging from city, state and country wide. One single place where patients can see multiple doctors with their specialization and can contact or can directly take an appointment with that doctor

- This project will be useful to any small clinics, medium and big hospitals which gives convenience to the people.
- People can search blood donors from their area or from their city by choosing the blood group.

Aims & Objectives

Specific goals are: -

- To produce a web-based system that allow the admin to add doctors, blood donors and provide functionalities to its role.
- To ease for booking appointment with doctor.
- To make entire process simple and easy to use over wider range.

Overall Description

Product Perspective:

2.1.1 Existing system function:

Existing System are not that matured to provide various facilities. Existing systems are specifically designed for particular hospitals and clinics for their own use. Existing system doesn’t provide blood donor related information. Systems are not so convenient and easy to use. To take the appointment and consultation from another one can need to visit the that doctor’s own platform which is sort of irritating thing.

III. Proposed System

Product functionality:

Hello Doctor provides the features for admin, doctors and patients. It includes several functionalities describes as below:

Book appointment with doctor:

It provides facility to take an appointment with doctor with doctor’s available time slots. Doctor can see the details of the patients and cancel the appointment any time. Patients can check their current active appointment and history of appointment.

Search for blood donor:

Project provides facility to search the blood donor of their choice of blood group from and any city of they want and also see the contact details of the blood donor. That will help during emergency situation.

Verified personal details data security:

Administrator can only add the blood donor so as to avoid any false information on the platform. Admin can see the donors and doctors list and can delete any time.

Benefits of Hello Doctor

- This project is fully functional and flexible.
- It is very easy to use.
- This project gives full control to the administrator to avoid any mis functionality
- It saves a lot of time of patients avoids staying the queues at the hospitals and also give blood donors information.
- This project is not specific to any hospital. Doctors can contact to admin to get registered on the platform to give services to people.
- The application acts as an office that is open 24/7.
- It is easy to use and very convenient and real time further improvement and addition of features easily possible.

Users and Characteristics:

Admin:

- Admin can login to the system.
- View the list of all doctors.
- Add new doctor.
- Delete Flat doctor.
- Add new blood donor.
- View list of patients.
- View list of donors.
- Delete patients.
- Logout the admin.

Doctor:

- Doctor can login to the system.
- Can view and update his/her details.
- View active appointments.
- View appointment history.
- Create time slots for appointments.
- View todays time slots.
- Logout from the system.

Patient:

- Patient can login to system.
- Patient can update his/her profile.
- Can book appointment with doctor.
- Show current appointment.
- View appointment history.
- Can search and get info blood donor.
- Logout from the system.

Operating Environment:

Server Side:

Processor: Intel® Xeon® processor 3500 series
HDD: Minimum 500GB Disk Space
RAM: Minimum 2GB
OS: Windows 10, Linux 6
Database: Oracle 11g

Client Side (minimum requirement):

Processor: Intel Dual Core
HDD: Minimum 80GB Disk Space
RAM: Minimum 1GB
OS: Windows 7 or above/ Linux

Software Requirements:

Operating System:	Windows 10
Browser	**:** Google Chrome
Front End	**:** VS Code
Backend	**:** Spring tool suite and Mysql

Design and Implementation Constraints:

- The application will use React js, axios, Rest API , Bootstrap and css as main web technologies.
- HTTP and FTP protocols are used as communication protocols. FTP is used to upload the web application in live domain and the client can access it via HTTP protocol.
- Several types of validations make this web application a secured one and SQL Injections can also be prevented.

- Since Hello Doctor is a web-based application, internet connection must be established.
- The Hello Doctor will be used on PCs and will function via internet or intranet in any web browser.

Requirement specification

External Interface Requirements:

User Interfaces:

- All the users will see the same page when they enter in this website. This page gives the option to signup for patients and common login option to all admin, doctor and patients.
- After being authenticated by correct username and password, user will be redirect to their corresponding profile where they can do various activities.
- The user interface will be simple and consistence, using terminology commonly understood by intended users of the system. The system will have simple interface, consistence with standard interface, to eliminate need for user training of infrequent users.

Hardware Interfaces:

- No extra hardware interfaces are needed.
- The system will use the standard hardware and data communication resources.
- This includes, but not limited to, general network connection at the server/hosting site, network server and network management tools.

Application Interfaces:

OS: Windows 10, Linux

Web Browser:

The system is a web-based application; clients need a modern web browser such as Mozilla Firebox, Internet Explorer, Opera, and Chrome. The computer must have an Internet connection in order to be able to access the system.

Communications Interfaces:

- This system uses communication resources which includes but not limited to, HTTP protocol for communication with the web browser and web server and TCP/IP network protocol with HTTP protocol.
- This application will communicate with the database that holds all the booking information. Users can contact with server side through HTTP protocol by means of a function that is called HTTP Service. This function allows the application to use the data retrieved by server to fulfil the request fired by the user.

System Diagrams

Activity Diagram

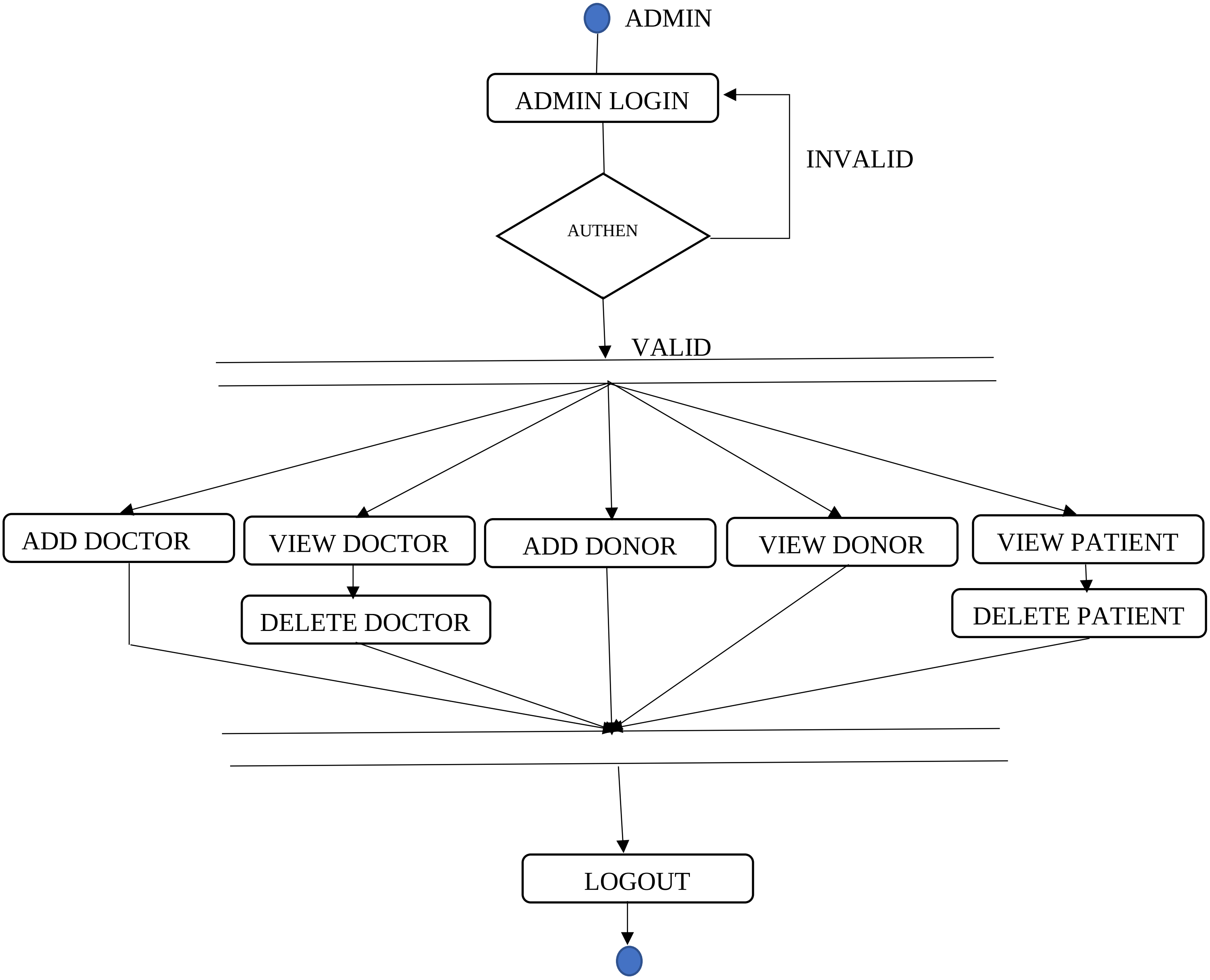


Figure 1: Admin Activity Diagram

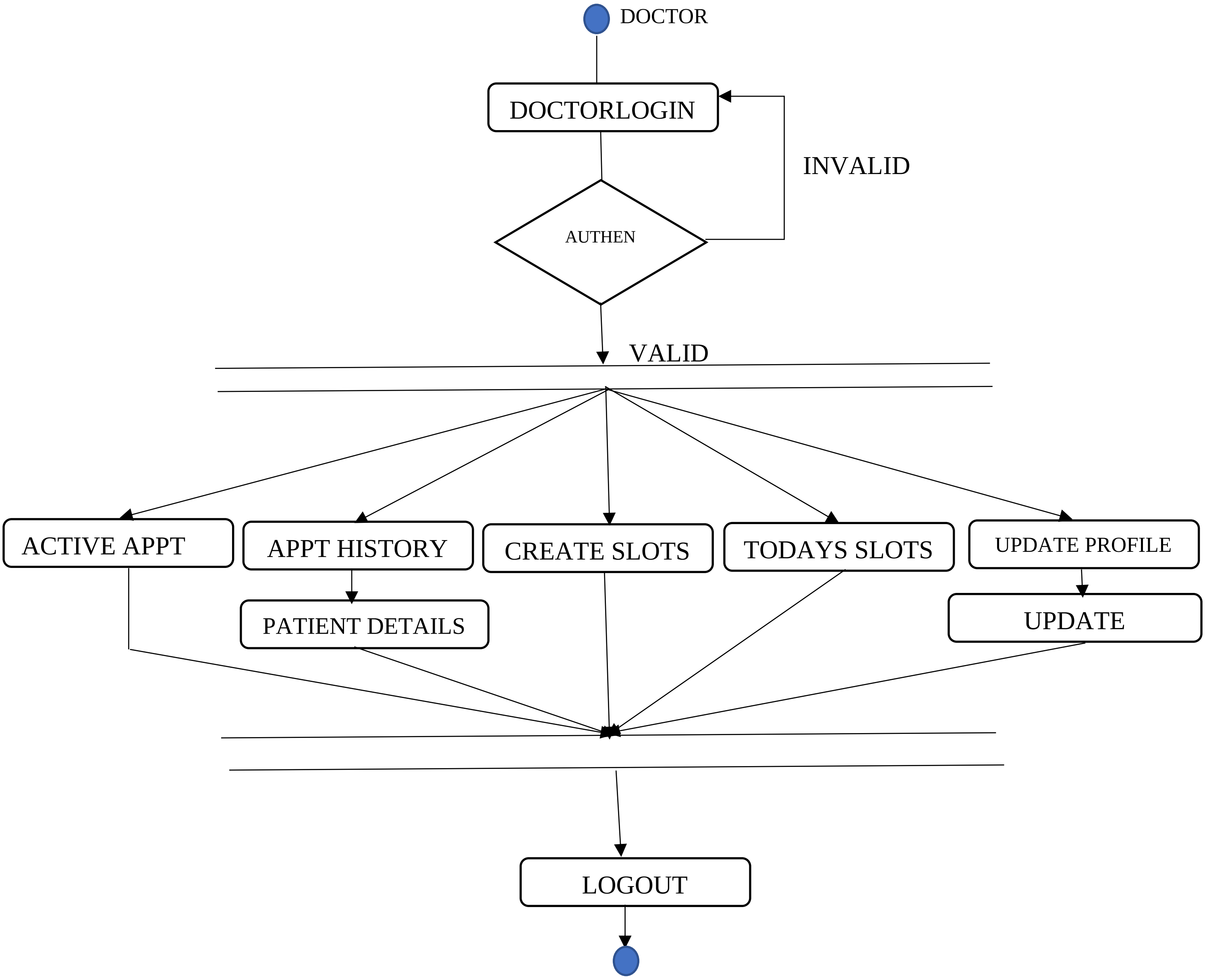


Figure 2: Doctor Activity Diagram

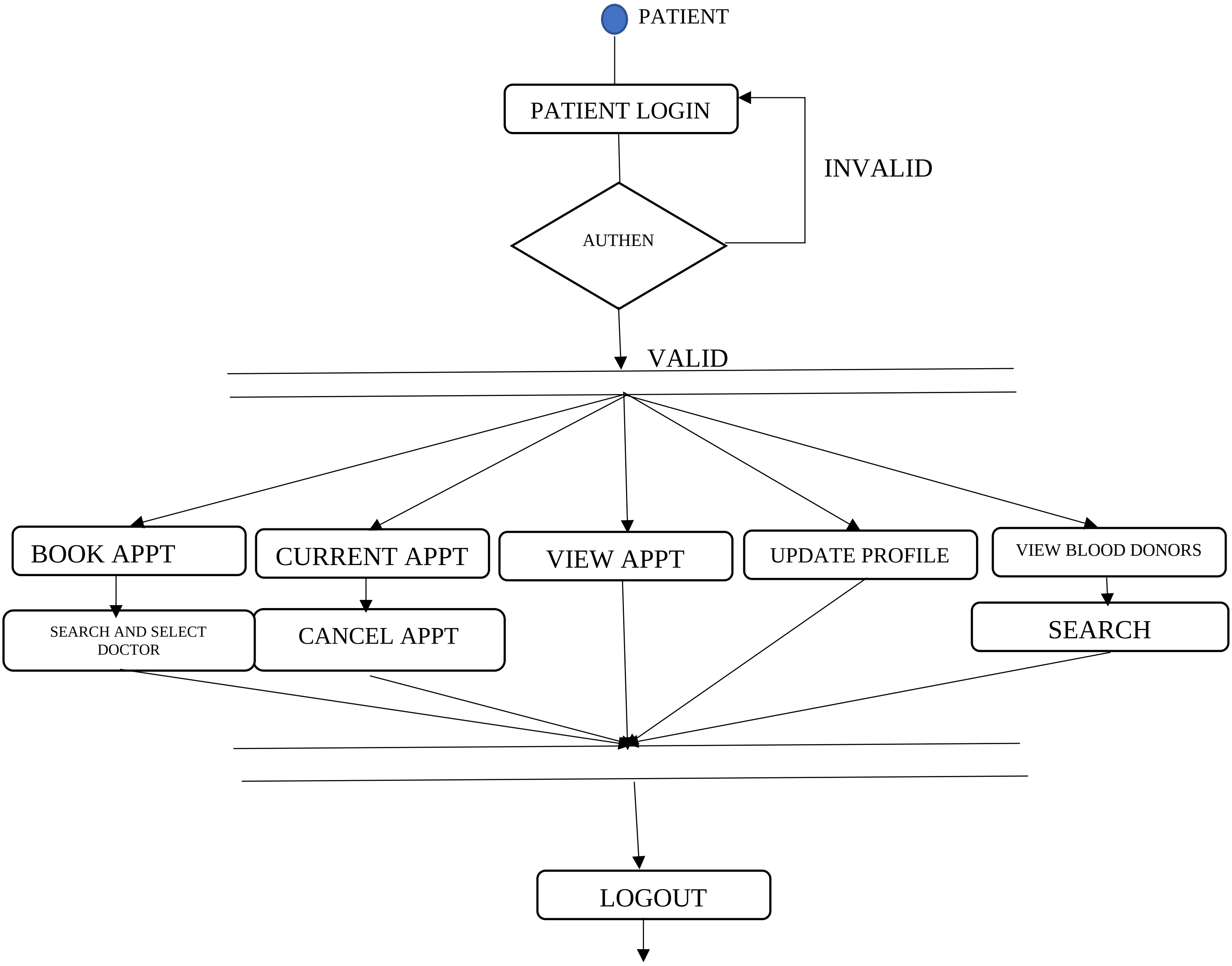


Figure 3: Watchman Activity Diagram

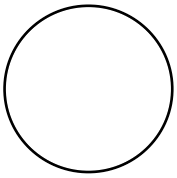
Data Flow Diagram (DFD)

Data Flow Diagram (DFD) is a pictorial representation, which shows the data passes from various stages one by one during the processing. DFD has some in defined symbols using, which we can denote input, dataflow and storing databases files.

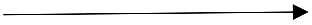
Symbols used in DFD:-

Input & Output

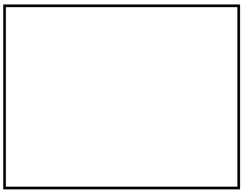
Process



Flow of data



Data storage



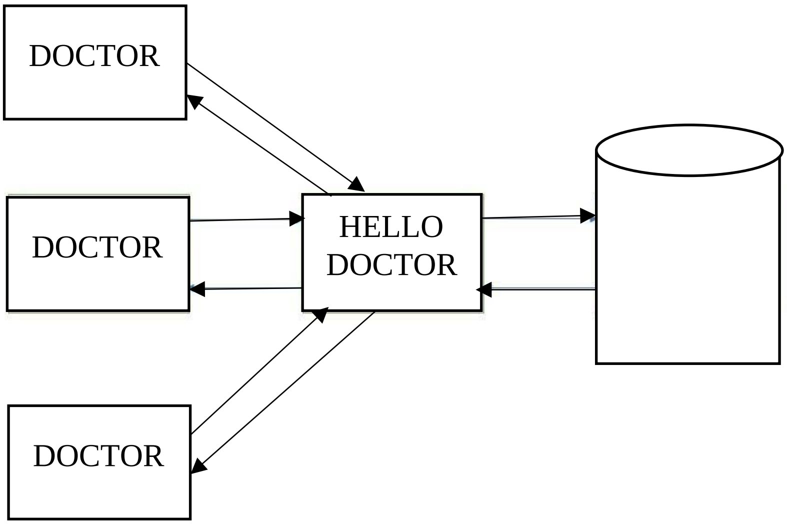


Figure 4: Level 0 Data Flow Diagram

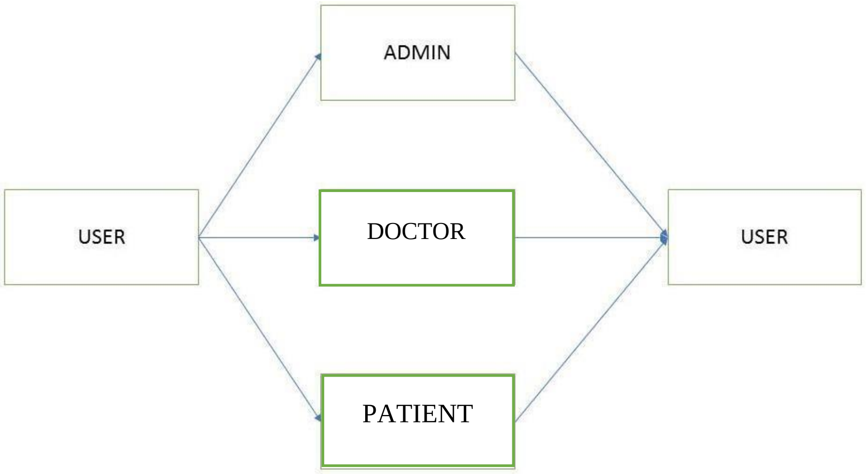


Figure 5: Level 1 Data Flow Diagram

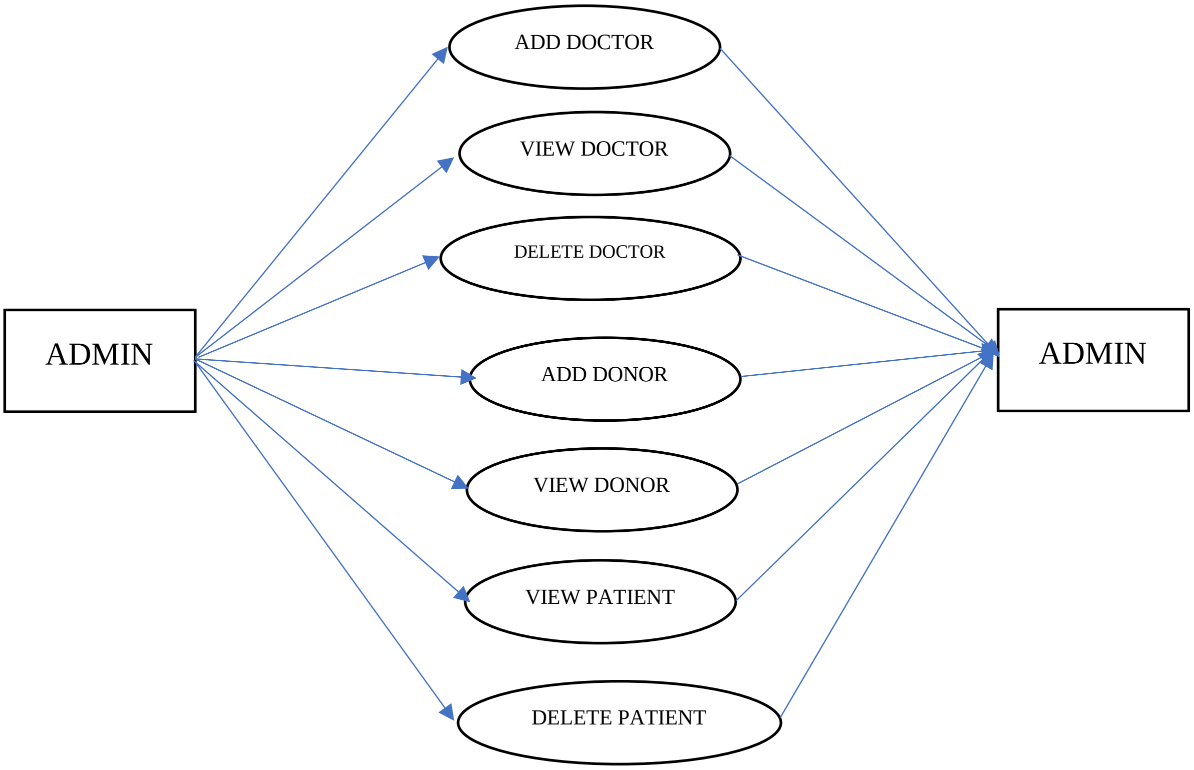


Figure 6: Level 2 Data Flow Diagram for Admin

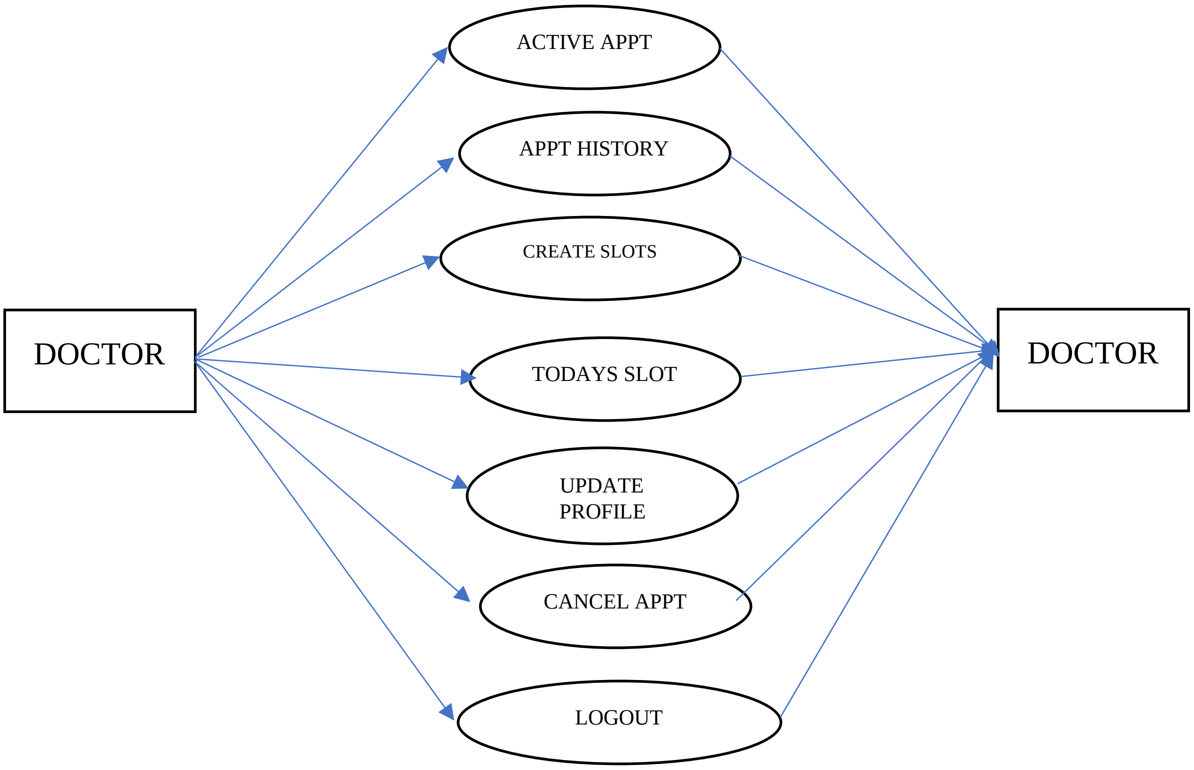


Figure 7: Level 2 Data Flow Diagram for Doctor

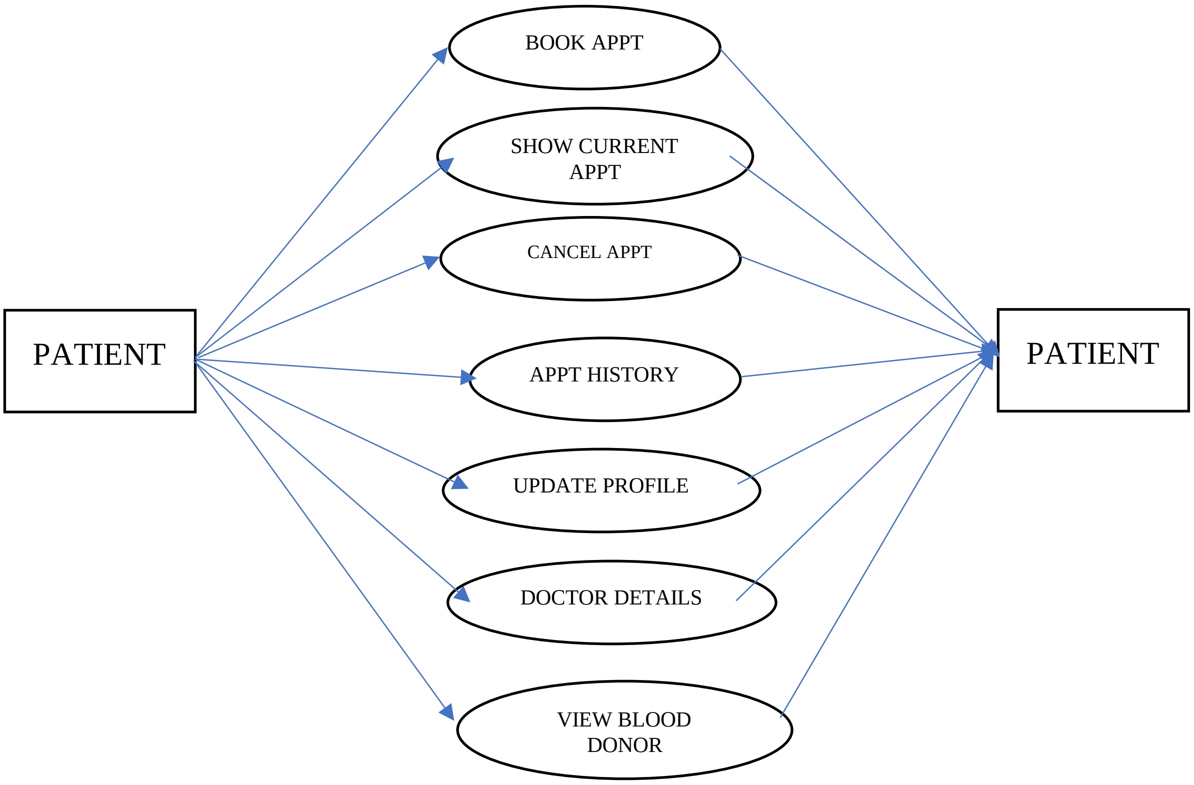


Figure 8: Level 2 Data Flow Diagram for Patient

Use Case Diagram

A Use case is an explanation of set of sequence of events graphically. It is rendered as an ellipse with rock-solid line up as well as lone its name. Use case diagram is a behavioural diagram that shows a set of use cases and actors and their relationship. It is a relationship among the use cases and actors. An actor represents a real-world object.

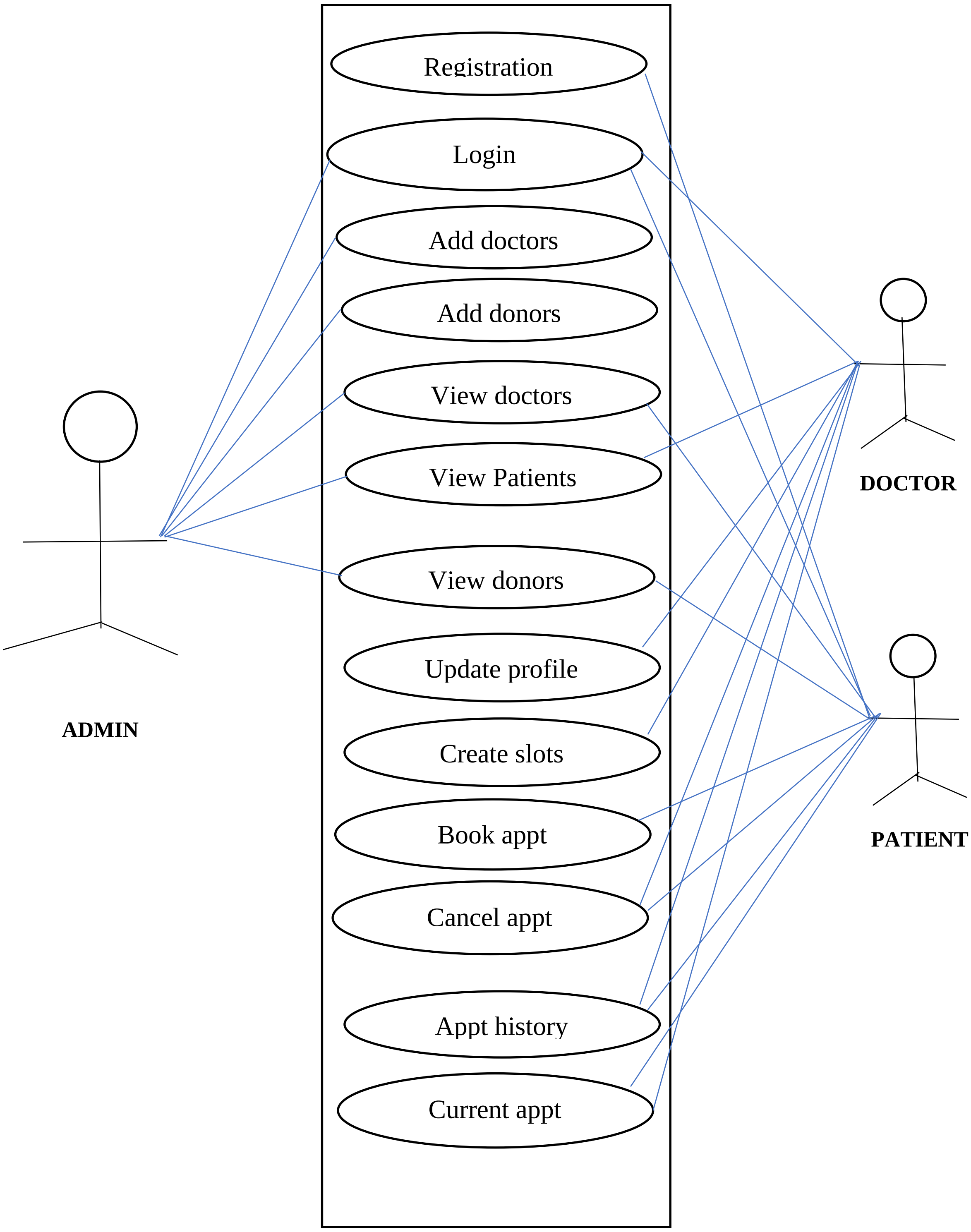


Figure 10: Use Case Diagram

ER Diagram

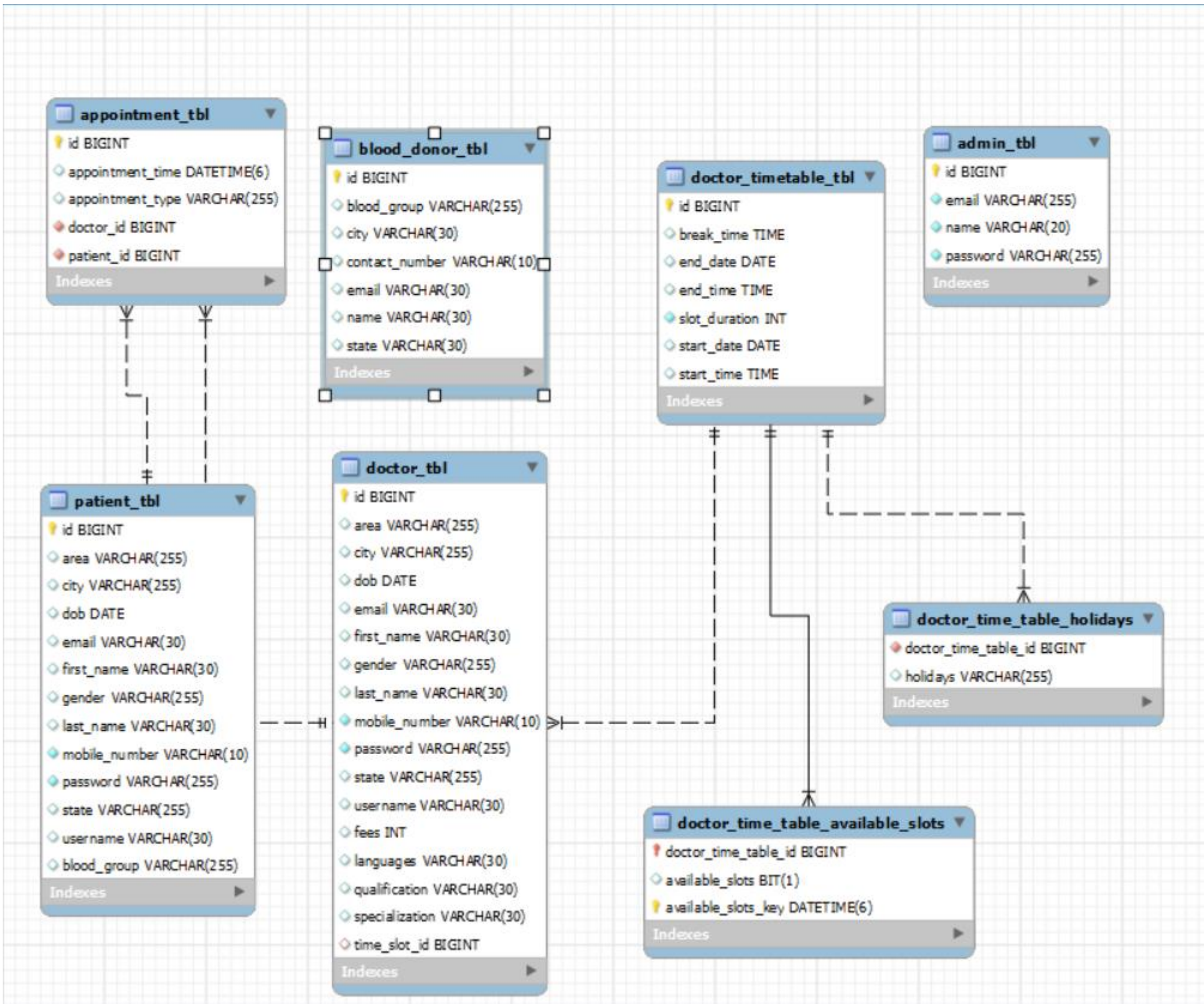
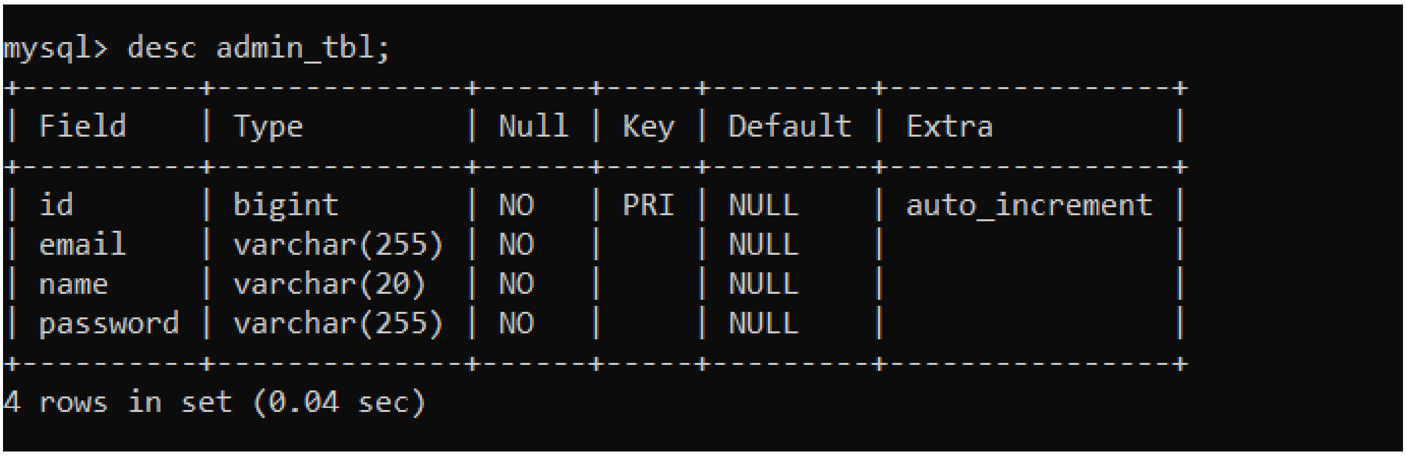


Figure 11: ER Diagram

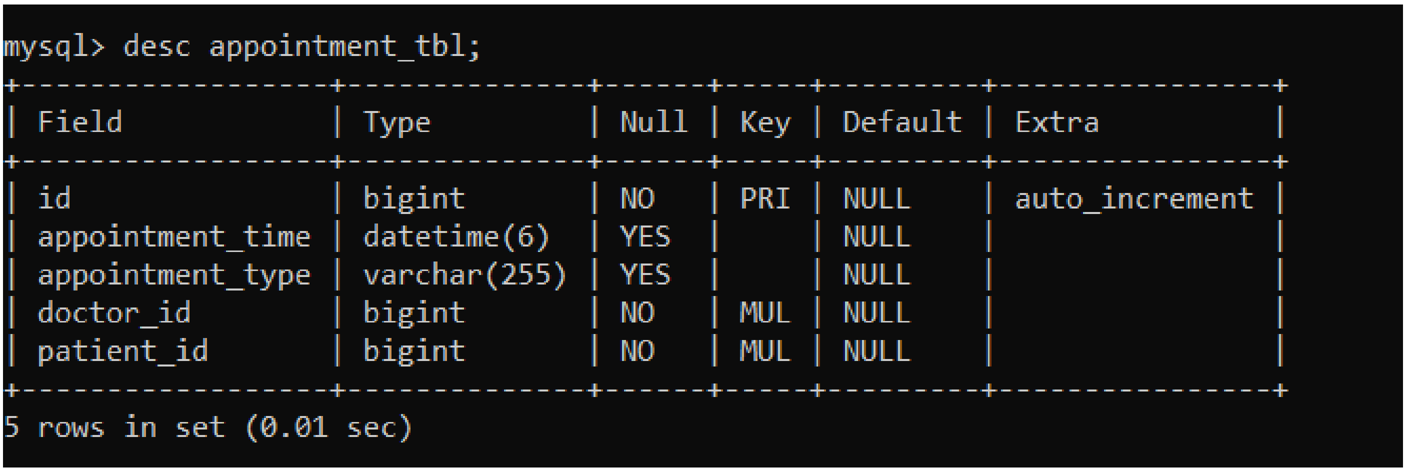
Table Structure

Admin table:



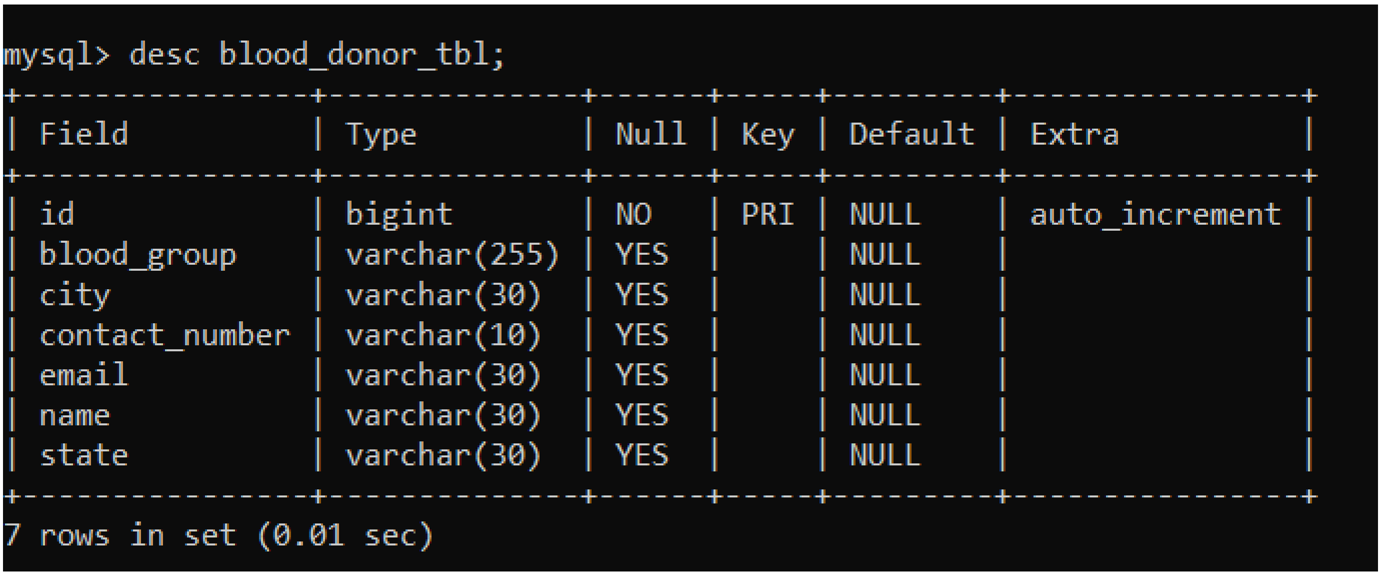
```
mysql> desc admin_tbl;
+----------+--------------+------+-----+---------+----------------+
| Field    | Type         | Null | Key | Default | Extra          |
+----------+--------------+------+-----+---------+----------------+
| id       | bigint       | NO   | PRI | NULL    | auto_increment |
| email    | varchar(255) | NO   |     | NULL    |                |
| name     | varchar(20)  | NO   |     | NULL    |                |
| password | varchar(255) | NO   |     | NULL    |                |
+----------+--------------+------+-----+---------+----------------+
4 rows in set (0.04 sec)
```

Appointment table:



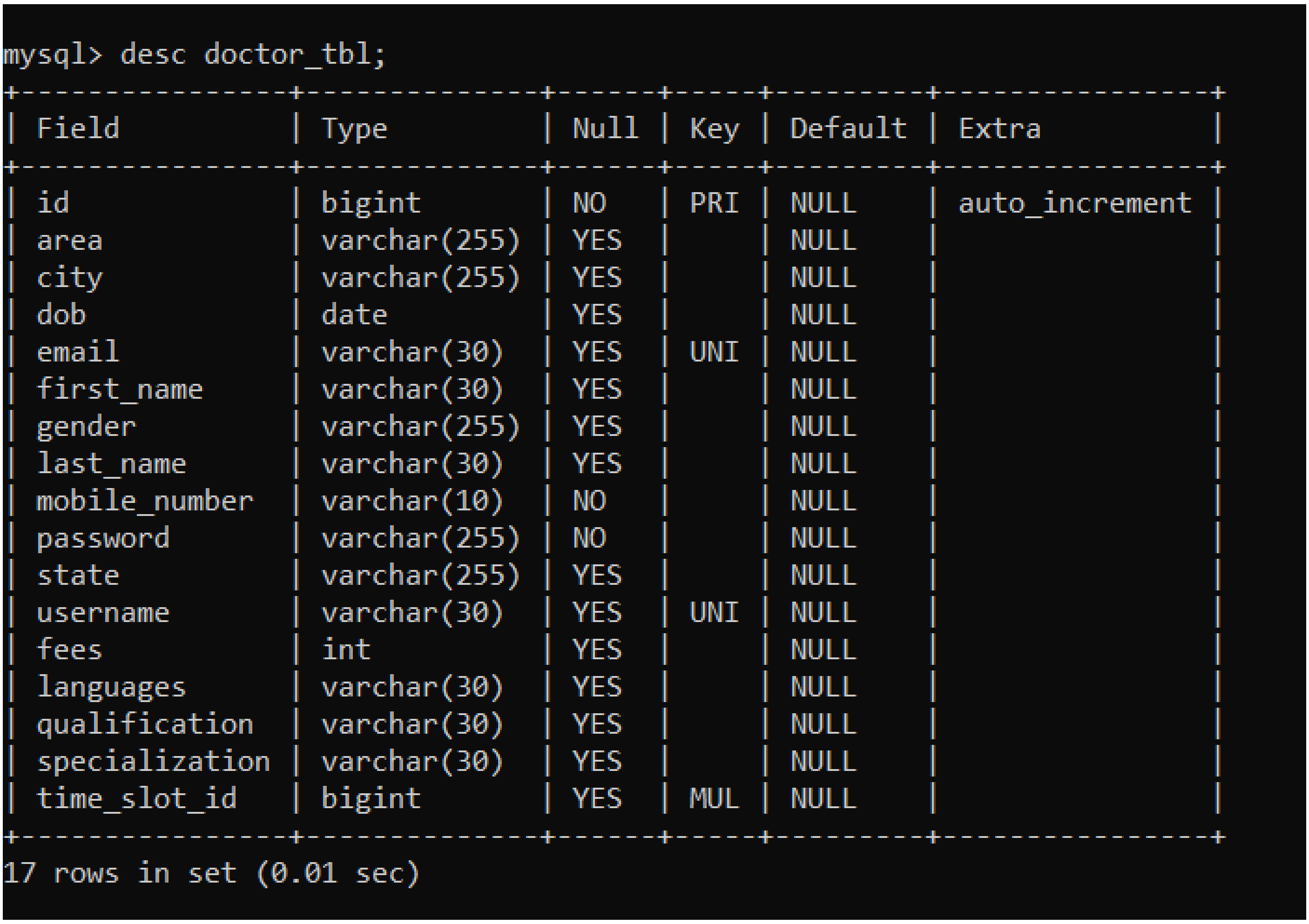
```
mysql> desc appointment_tbl;
+------------------+--------------+------+-----+---------+----------------+
| Field            | Type         | Null | Key | Default | Extra          |
+------------------+--------------+------+-----+---------+----------------+
| id               | bigint       | NO   | PRI | NULL    | auto_increment |
| appointment_time | datetime(6)  | YES  |     | NULL    |                |
| appointment_type | varchar(255) | YES  |     | NULL    |                |
| doctor_id        | bigint       | NO   | MUL | NULL    |                |
| patient_id       | bigint       | NO   | MUL | NULL    |                |
+------------------+--------------+------+-----+---------+----------------+
5 rows in set (0.01 sec)
```

Blood donor table:



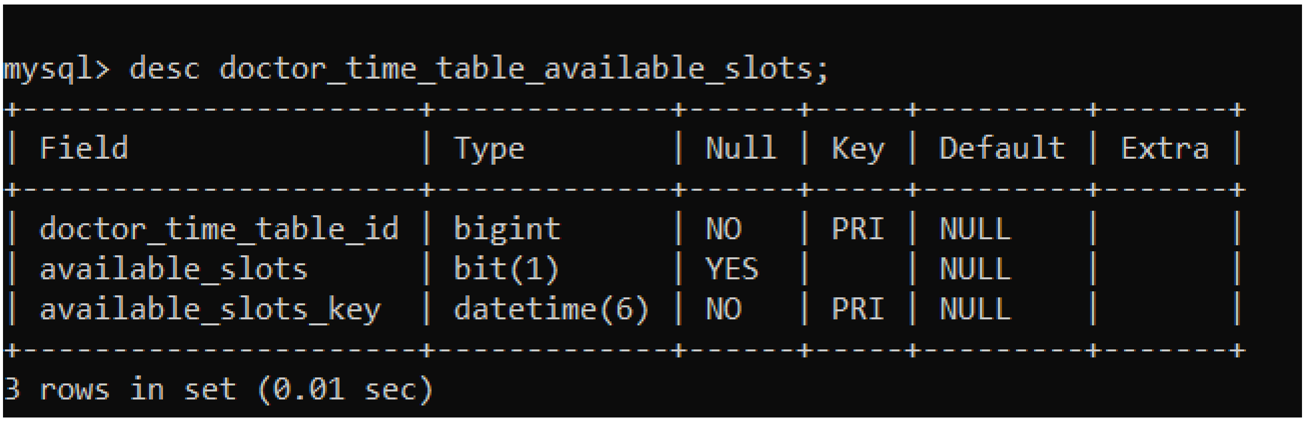
```
mysql> desc blood_donor_tbl;
+----------------+--------------+------+-----+---------+----------------+
| Field          | Type         | Null | Key | Default | Extra          |
+----------------+--------------+------+-----+---------+----------------+
| id             | bigint       | NO   | PRI | NULL    | auto_increment |
| blood_group    | varchar(255) | YES  |     | NULL    |                |
| city           | varchar(30)  | YES  |     | NULL    |                |
| contact_number | varchar(10)  | YES  |     | NULL    |                |
| email          | varchar(30)  | YES  |     | NULL    |                |
| name           | varchar(30)  | YES  |     | NULL    |                |
| state          | varchar(30)  | YES  |     | NULL    |                |
+----------------+--------------+------+-----+---------+----------------+
7 rows in set (0.01 sec)
```

Doctor table:



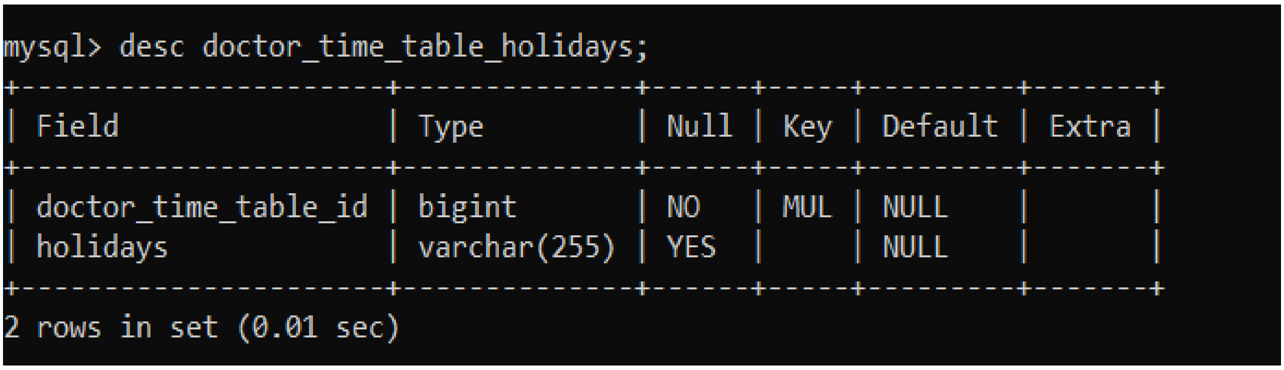
```
mysql> desc doctor_tbl;
+----------------+--------------+------+-----+---------+----------------+
| Field          | Type         | Null | Key | Default | Extra          |
+----------------+--------------+------+-----+---------+----------------+
| id             | bigint       | NO   | PRI | NULL    | auto_increment |
| area           | varchar(255) | YES  |     | NULL    |                |
| city           | varchar(255) | YES  |     | NULL    |                |
| dob            | date         | YES  |     | NULL    |                |
| email          | varchar(30)  | YES  | UNI | NULL    |                |
| first_name     | varchar(30)  | YES  |     | NULL    |                |
| gender         | varchar(255) | YES  |     | NULL    |                |
| last_name      | varchar(30)  | YES  |     | NULL    |                |
| mobile_number  | varchar(10)  | NO   |     | NULL    |                |
| password       | varchar(255) | NO   |     | NULL    |                |
| state          | varchar(255) | YES  |     | NULL    |                |
| username       | varchar(30)  | YES  | UNI | NULL    |                |
| fees           | int          | YES  |     | NULL    |                |
| languages      | varchar(30)  | YES  |     | NULL    |                |
| qualification  | varchar(30)  | YES  |     | NULL    |                |
| specialization | varchar(30)  | YES  |     | NULL    |                |
| time_slot_id   | bigint       | YES  | MUL | NULL    |                |
+----------------+--------------+------+-----+---------+----------------+
17 rows in set (0.01 sec)
```

Doctor time table available slots:



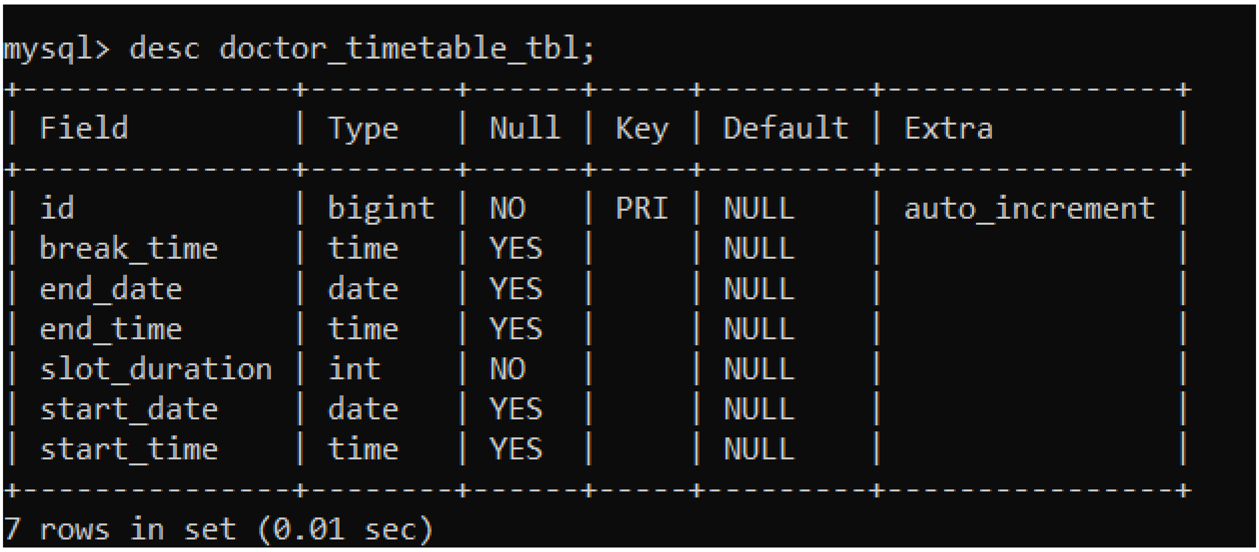
```
mysql> desc doctor_time_table_available_slots;
+----------------------+-------------+------+-----+---------+-------+
| Field                | Type        | Null | Key | Default | Extra |
+----------------------+-------------+------+-----+---------+-------+
| doctor_time_table_id | bigint      | NO   | PRI | NULL    |       |
| available_slots      | bit(1)      | YES  |     | NULL    |       |
| available_slots_key  | datetime(6) | NO   | PRI | NULL    |       |
+----------------------+-------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

Doctor time table holidays:



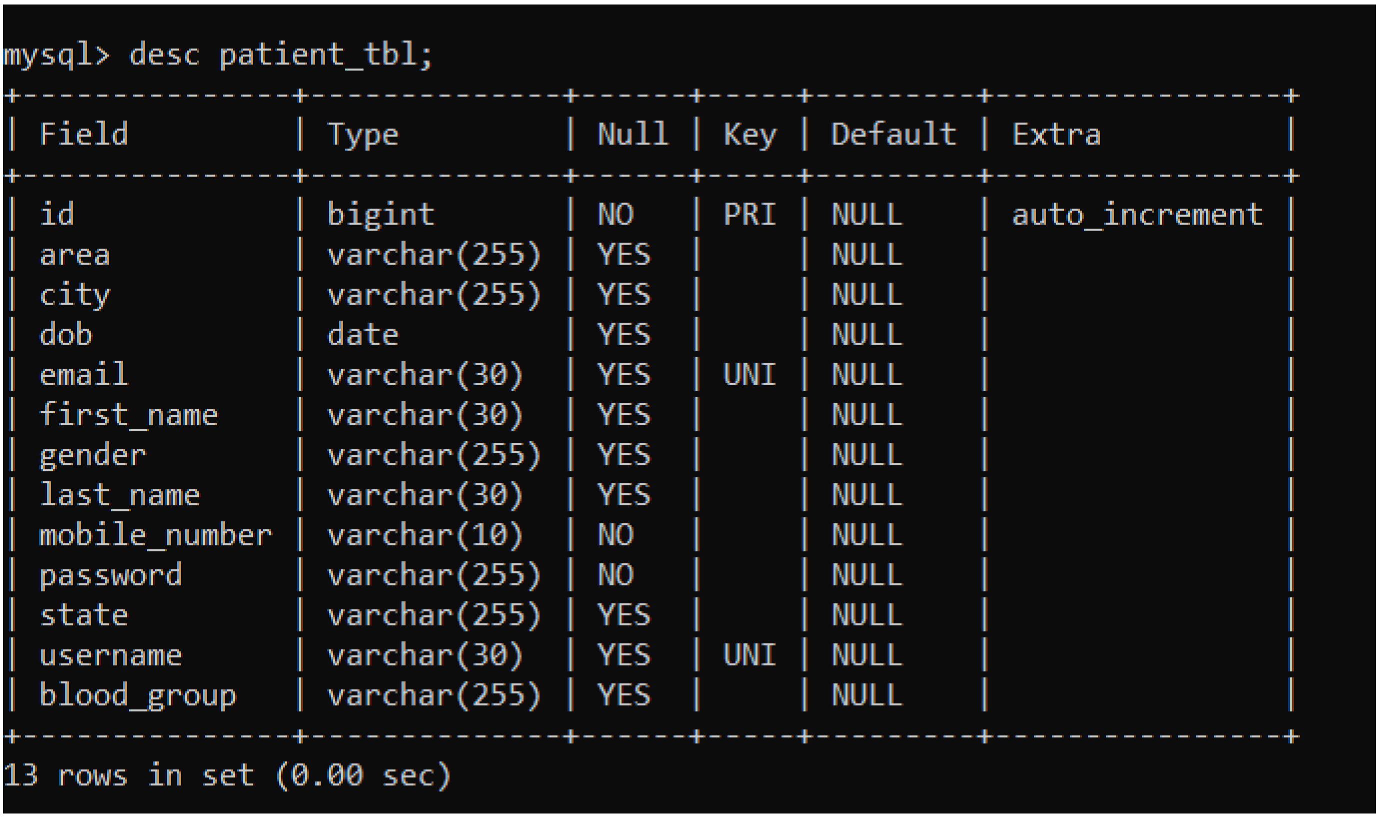
```
mysql> desc doctor_time_table_holidays;
+----------------------+--------------+------+-----+---------+-------+
| Field                | Type         | Null | Key | Default | Extra |
+----------------------+--------------+------+-----+---------+-------+
| doctor_time_table_id | bigint       | NO   | MUL | NULL    |       |
| holidays             | varchar(255) | YES  |     | NULL    |       |
+----------------------+--------------+------+-----+---------+-------+
2 rows in set (0.01 sec)
```

Doctor timetable table:



```
mysql> desc doctor_timetable_tbl;
+---------------+--------+------+-----+---------+----------------+
| Field         | Type   | Null | Key | Default | Extra          |
+---------------+--------+------+-----+---------+----------------+
| id            | bigint | NO   | PRI | NULL    | auto_increment |
| break_time    | time   | YES  |     | NULL    |                |
| end_date      | date   | YES  |     | NULL    |                |
| end_time      | time   | YES  |     | NULL    |                |
| slot_duration | int    | NO   |     | NULL    |                |
| start_date    | date   | YES  |     | NULL    |                |
| start_time    | time   | YES  |     | NULL    |                |
+---------------+--------+------+-----+---------+----------------+
7 rows in set (0.01 sec)
```

Patient table:



```
mysql> desc patient_tbl;
+----------------+--------------+------+-----+---------+----------------+
| Field          | Type         | Null | Key | Default | Extra          |
+----------------+--------------+------+-----+---------+----------------+
| id             | bigint       | NO   | PRI | NULL    | auto_increment |
| area           | varchar(255) | YES  |     | NULL    |                |
| city           | varchar(255) | YES  |     | NULL    |                |
| dob            | date         | YES  |     | NULL    |                |
| email          | varchar(30)  | YES  | UNI | NULL    |                |
| first_name     | varchar(30)  | YES  |     | NULL    |                |
| gender         | varchar(255) | YES  |     | NULL    |                |
| last_name      | varchar(30)  | YES  |     | NULL    |                |
| mobile_number  | varchar(10)  | NO   |     | NULL    |                |
| password       | varchar(255) | NO   |     | NULL    |                |
| state          | varchar(255) | YES  |     | NULL    |                |
| username       | varchar(30)  | YES  | UNI | NULL    |                |
| blood_group    | varchar(255) | YES  |     | NULL    |                |
+----------------+--------------+------+-----+---------+----------------+
13 rows in set (0.00 sec)
```

Conclusion

This project helps in making paperless activities. It reduces the workload from Doctor and Receptionist. It provides more ease and flexibility to Doctor, Administrator and Receptionist. This digitalization has reduced costs of Hospital. This work has created a little awareness and promotes the idea that the concept of paperless office is reality. Online appointment helps to reduce people’s wastage of time and also reduce the crowd at hospitals specially in this pandemic.

Future Scope

The Project can be further extended to provide it with a reliable and genuine Payment System. Also, the project can be extended in to allow the tracking of the locations of hospitals. Many other functionalities such as subscribing to future events, online consultation make the system more efficient and more reliable.

7.0 References

[1] Google for problem solving
[2] Head First Java 2nd Edition
[3] http://www.javatpoint.com/java-Latorial
[4] http://www tutorialspoint.com/mysql/
[5] https://www.w3schools.com/
[6] https://www.geeksforgeeks.org/