**HEALTHCARE BASED PROJECT**

**DOCTOR APPOINTMENT WEB BASED PROJECT**

SOFTWARE REQUIREMENT SPECIFICATION

PREPARED BY:

VISHAL PAIKRA

HUMAN LAL

TABLE OF CONTENT

|  |  |  |
| --- | --- | --- |
| Sr. No. | Title | Page No. |
| 1 | Abstract |  |
| 2 | Introduction |  |
|  | 2.1 purpose of the system |  |
|  | 2.2 target audience |  |
|  | 2.3 scope of the project |  |
|  | 2.4 operating environment |  |
|  |  |  |
| 3 | Existing system |  |
| 4 | Need for the new system |  |
| 5 | Functional specification |  |
| 6 | Hardware and software requirement |  |
| 7 | System flow chart |  |
| 8 | E -R diagram |  |
| 9 | UML diagram |  |
|  | 9.1 use case diagram |  |
|  | 9.2 activity diagram |  |
|  | 9.3 class diagram |  |
|  | 9.4 sequence diagram |  |
|  | 9.5 collaboration diagram |  |
| 10 | Data dictionary |  |
| 11 | Input design |  |
| 12 | Output design |  |
| 13 | Testing |  |
| 14 | Post implement review |  |
| 15 | Future enhancement |  |
| 16 | Bibliography |  |

PROJECT PROFILE

|  |  |
| --- | --- |
| Project name | Online doctor appointment system. |
| Objective | The main objective is to develop the software that covers all therapists of management and operations of clinics. It enables healthcare providers to improve operational effectiveness, reduce costs, reduce medical errors, reduce time consumption and enhance delivery of quality of care. |
| Platform | PHP |
| Front end /data base | Java Script/PHP MySQL |
| Using programming  language | PHP, java Script, CSS |
| Operating system | Window11 |

ABSTRACT

Life is becoming too busy to get medical

appointments in person and to maintain a proper health care.

The main idea of this work is to provide ease and comfort to

patients while taking appointment from doctors and it also

resolves the problems that the patients has to face while making

an appointment. The android application Mr.Doc acts as a client

whereas the database containing the doctor’s details, patient’s

details and appointment details is maintained by a website that

acts as a server.

Life is becoming too busy to get medical

appointments in person and to maintain a proper health care.

The main idea of this work is to provide ease and comfort to

patients while taking appointment from doctors and it also

resolves the problems that the patients has to face while making

an appointment. The android application Mr.Doc acts as a client

whereas the database containing the doctor’s details, patient’s

details and appointment details is maintained by a website that

acts as a server.

Life is becoming too busy to get medical

appointments in person and to maintain a proper health care.

The main idea of this work is to provide ease and comfort to

patients while taking appointment from doctors and it also

resolves the problems that the patients has to face while making

an appointment. The android application Mr.Doc acts as a client

whereas the database containing the doctor’s details, patient’s

details and appointment details is maintained by a website that

acts as a server.

Life is becoming too busy to get medical appointments in person and to maintain a proper health care. The main idea of this work is to provide ease and comfort to patients while taking appointment from doctors and it also resolves the problems that the patients have to face while making an appointment. Though India is a developing country, a number of internet users are in here. So, through internet if people want to get connected to their desired doctors there is a nexus will be needed. For that purpose, we have planned to build a website to get an appointment. This will help common people to get instant support without wasting time and effort even they will get this service from home and abroad. By using this system people can easily get to know about the timing of doctor’s counselling period and make their meeting whenever they want. Proper categorized list will make people more comfort to browse their expected doctors.

INTRODUCTION

* 1. PURPOSE OF THE SYSTEM

The purpose of the doctor appointment system is to automate the existing manual system by the help of computerized equipment and full- fledged computer software fulfilling their requirement, so that their valuable data information can be stored for a longer period with easy accessing and manipulation of the same. the required software and hardware are easily available and easy to work with.

* 1. TARGET AUDIENCE

The main objective of the project is to manage the details of the doctor appointment, patient, booking doctor schedule .it manage all the information about doctor, doctor fee. The project is totally built at administrative end and thus only the administrator is guaranteed the access. Purpose is to build an application program to reduce the manual work for managing the doctor etc.it track all the details.

* 1. SCOPEOF THE PROJECT
* Our project aims at business process automation we have tried to computerized various process of the doctor appointment system.
* it satisfies the user requirement.
* be easy to understand by the user and operator.
* have a good user interface.
* in the computer system, it is not necessary to create the manifest but we are directly print it, which save our time.
* in computer system the person has to fill the various forms & number of copies of the form can be easily generated at a time.
* it generates the report on doctor, appointments, doctor fee.
* We can easily export a pdf for the doctor.
  1. OPERATING INVIRONMENT

|  |  |
| --- | --- |
| Programming language | PHP, java Script, CSS |
| IDE | Visual studio code |
| Data base | MYSQL |

EXISTING SYSTEM

* Managing the many different appointment types
* Gathering all documentation and authorizations necessary before an appointment
* Adjusting schedules based on provider preferences for appointment types
* Handling referrals and the associated authorizations

NEED FOR THE NEW SYSTEM

* Time saving
* Monetary saving
* 24 hours convenience
* Online payment
* Sustain tranquility

FUNCTIONAL SPECIFICATION

* USER SPECIFICATION

ADMIN-

* 1. Admin can add edit delete and view doctor data
  2. Admin can view all patient data
  3. Admin can schedule any doctor availability data and manage that data like edit or delete it.
  4. Admin can view all doctor appointment data and he or she can also process appointment.
  5. Admin can see analytics data.

DOCTOR USERS-

* + 1. doctor can add edit delete their availability data under the system.
    2. Doctor can view only his or her appointment data, they cannot see the other doctor appointment data.
    3. Doctor can see the appointment patient details.
    4. Doctor can write comment in patient details.
    5. Doctor can edit profile details
    6. Doctor can login into this system.

PATIENT USER-

1. Patient can view doctor availability data on web page.
2. Patient can register into the system.
3. Patient can login into the system.
4. Patient can book appointment and they can also cancel appointment.
5. Patient can download appointment in pdf format also.
6. Patient can change their profile details also.

* MODULE SPECIFICATION-

1. Doctor management module
2. Doctor schedule module
3. Doctor fee module
4. Appointment management module
5. Patient module
6. Booking module
7. Registration module
8. Login module
9. User module

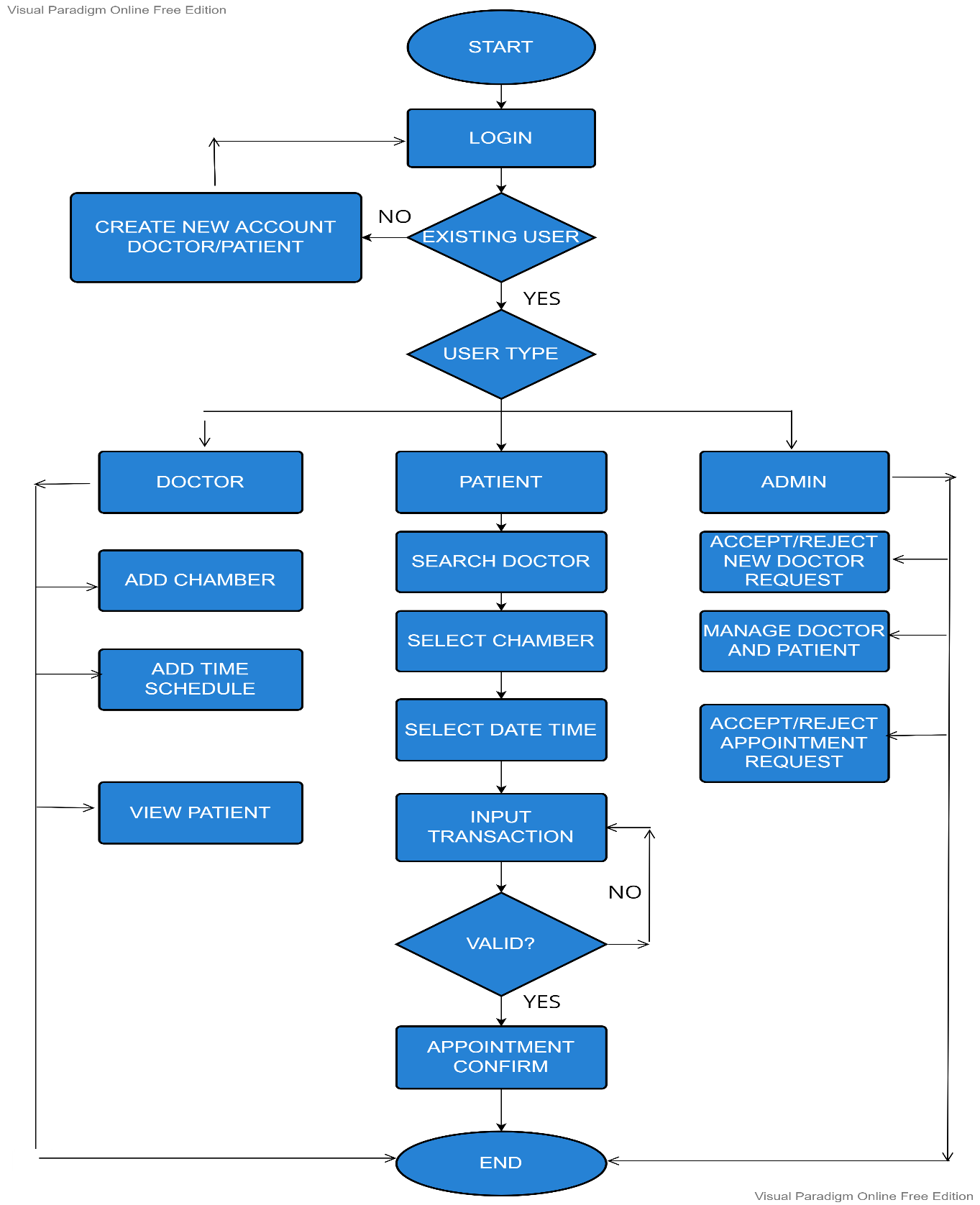
HARDWARE REQUIREMENT

|  |  |
| --- | --- |
| NAME OF COMPONENT | SPECIFICATION |
| Processor | 2.4 GHz processor speed, Pentium 630MHZ |
| RAM | 128MB |
| Hard disk | 500 GB |
| Keyboard | 122 keys |

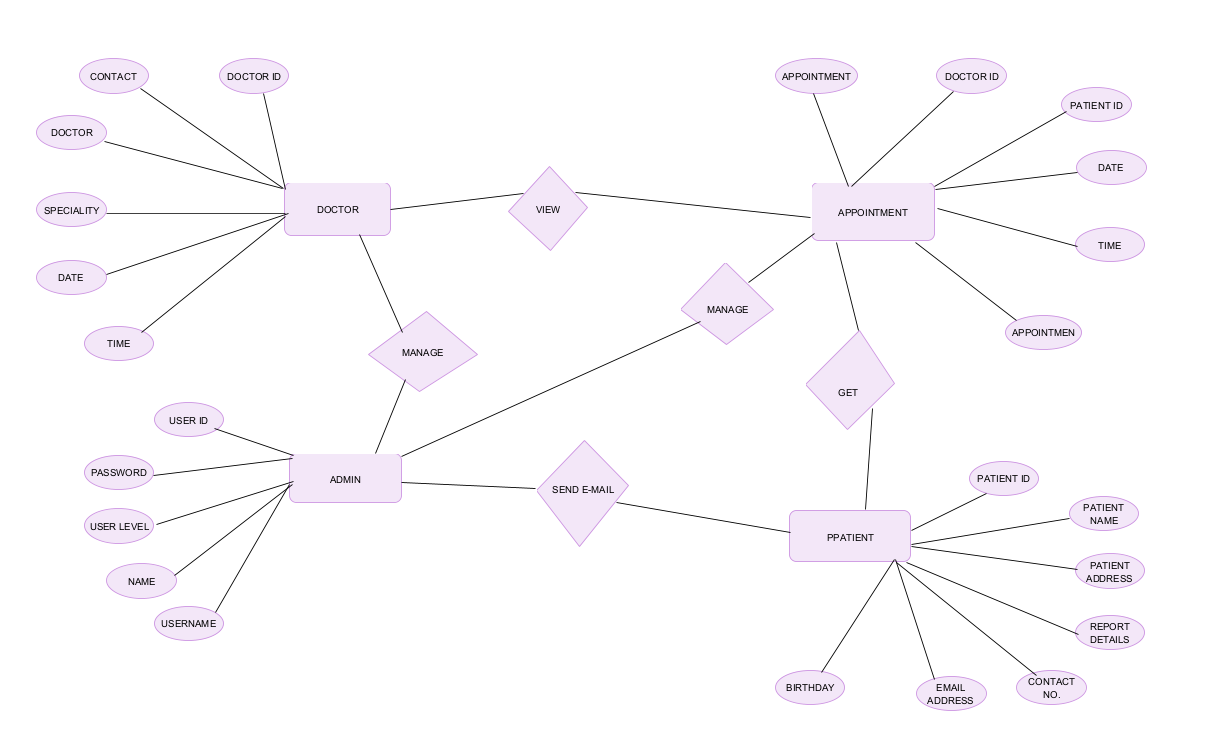
SOFTWARE REQUIREMENT

|  |  |
| --- | --- |
| NAME OF COMPONENT | SPECIFICATION |
| Operating system | Window, Linux |
| Language, browser | PHP, java Script, CSS |
| Database | MYSQL server |
| Web server | Chrome |
| Database JDBC driver | MYSQL Jconnector. |

SYSTEM FLOW CHART

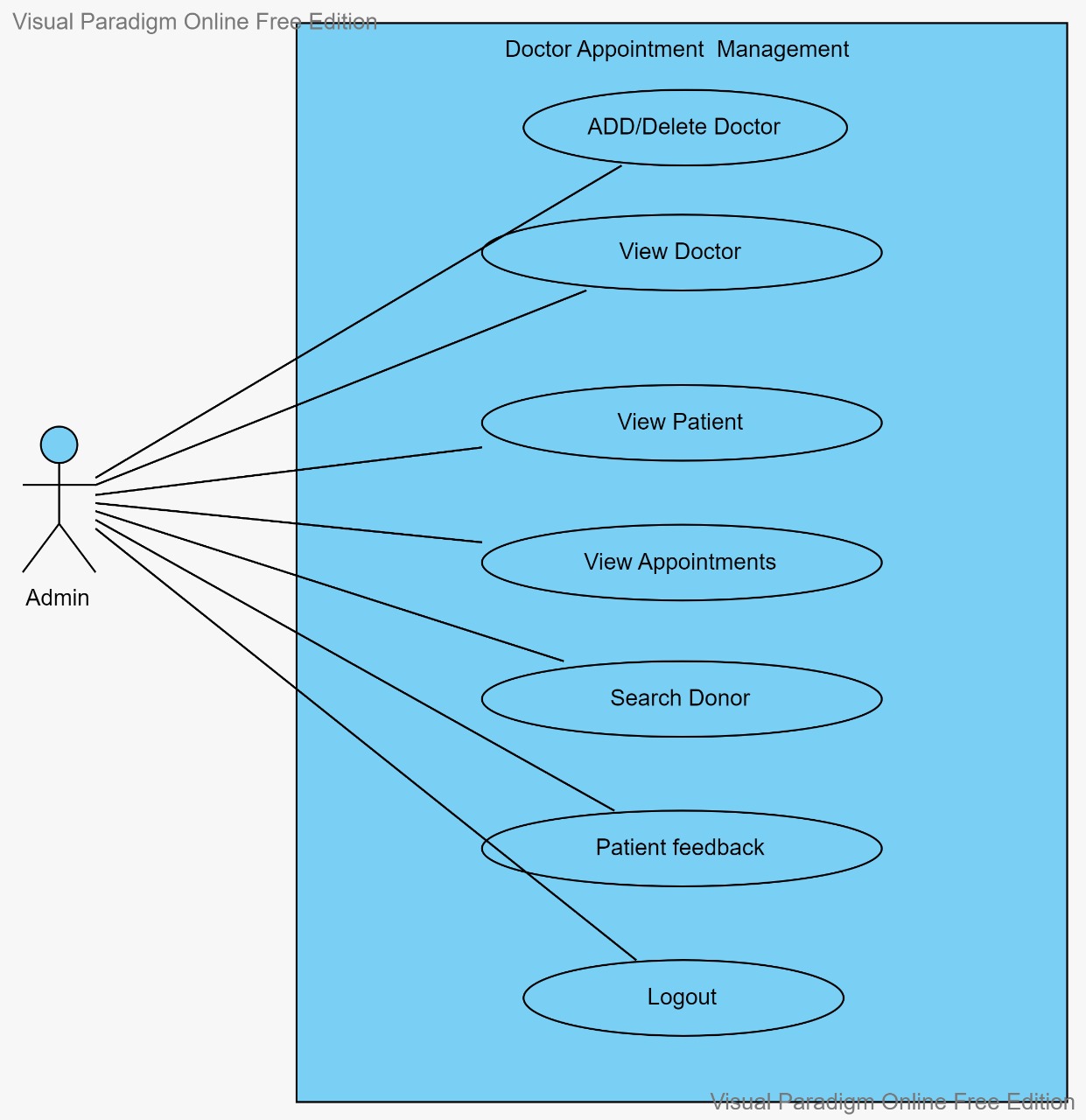


E-R DIAGRAM

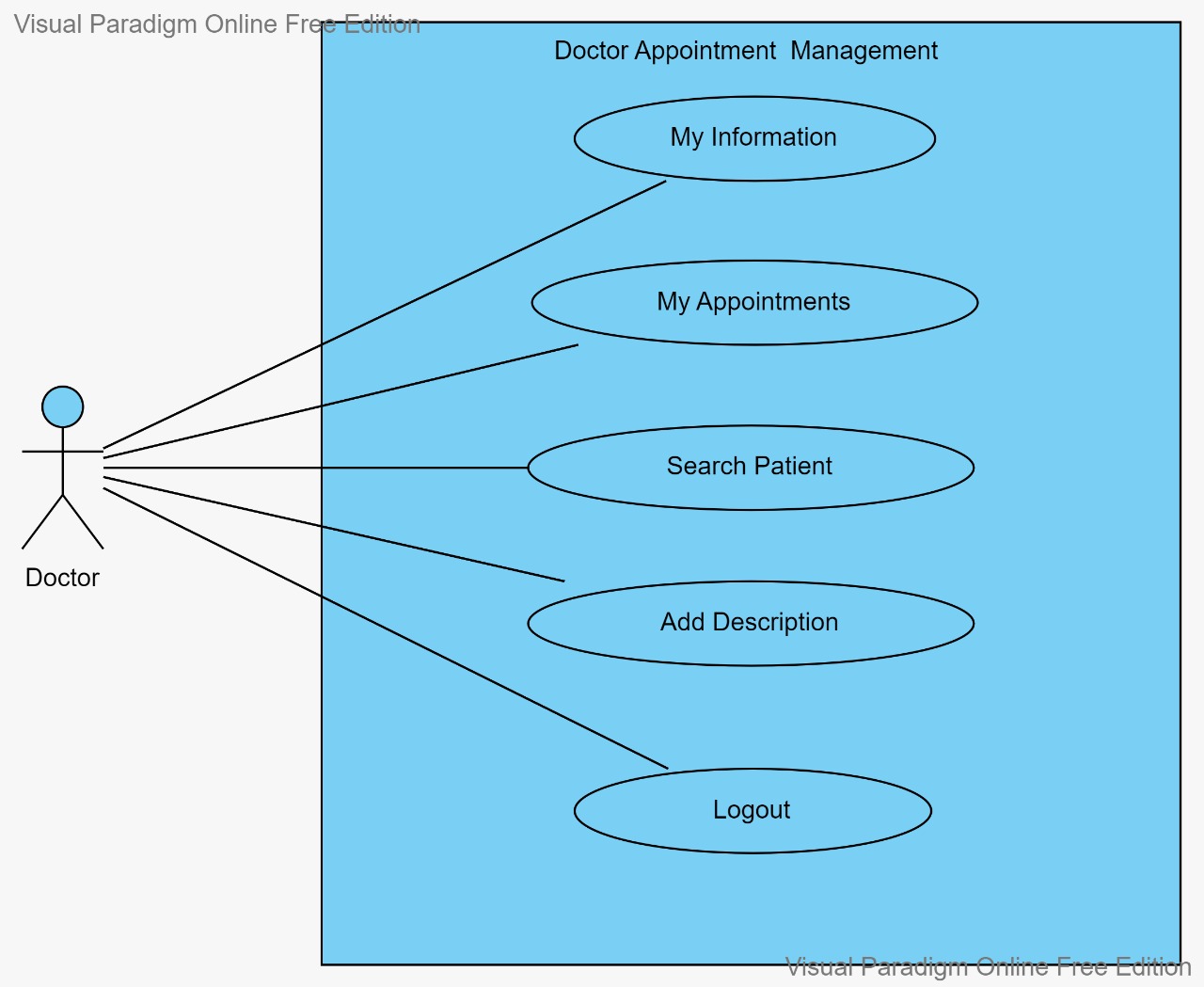


USE CASE DIAGRAM

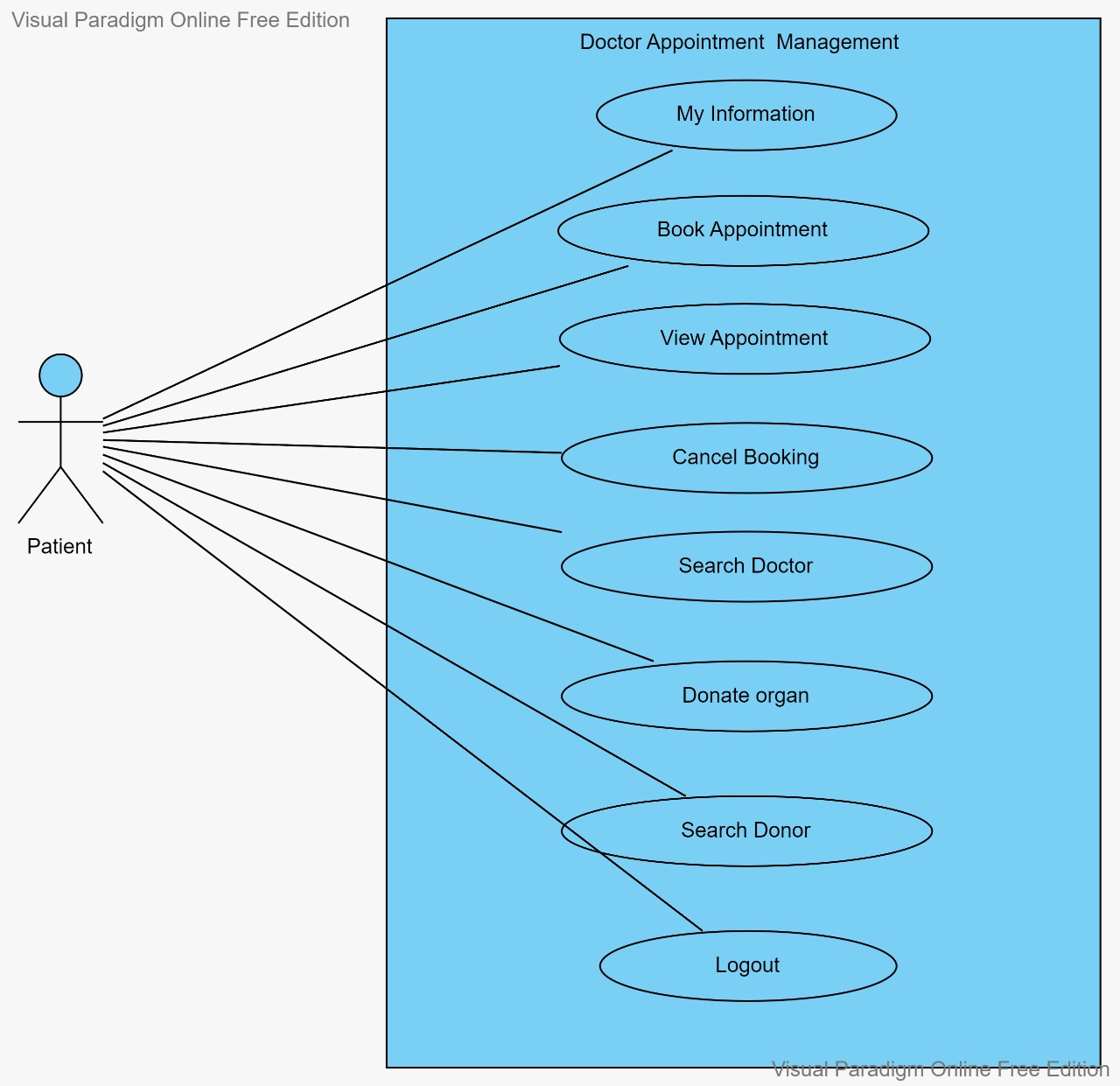
ADMIN



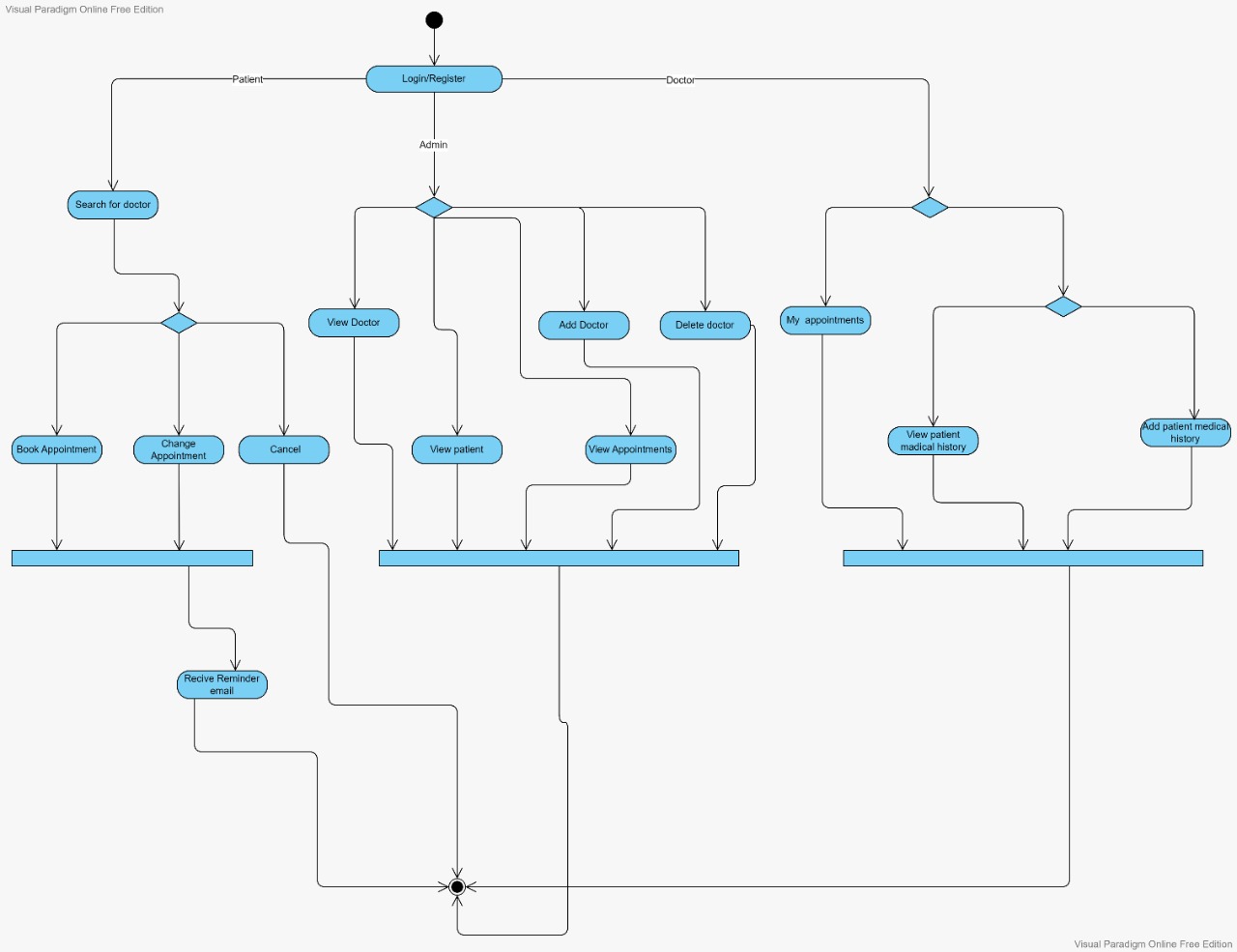
DOCTOR



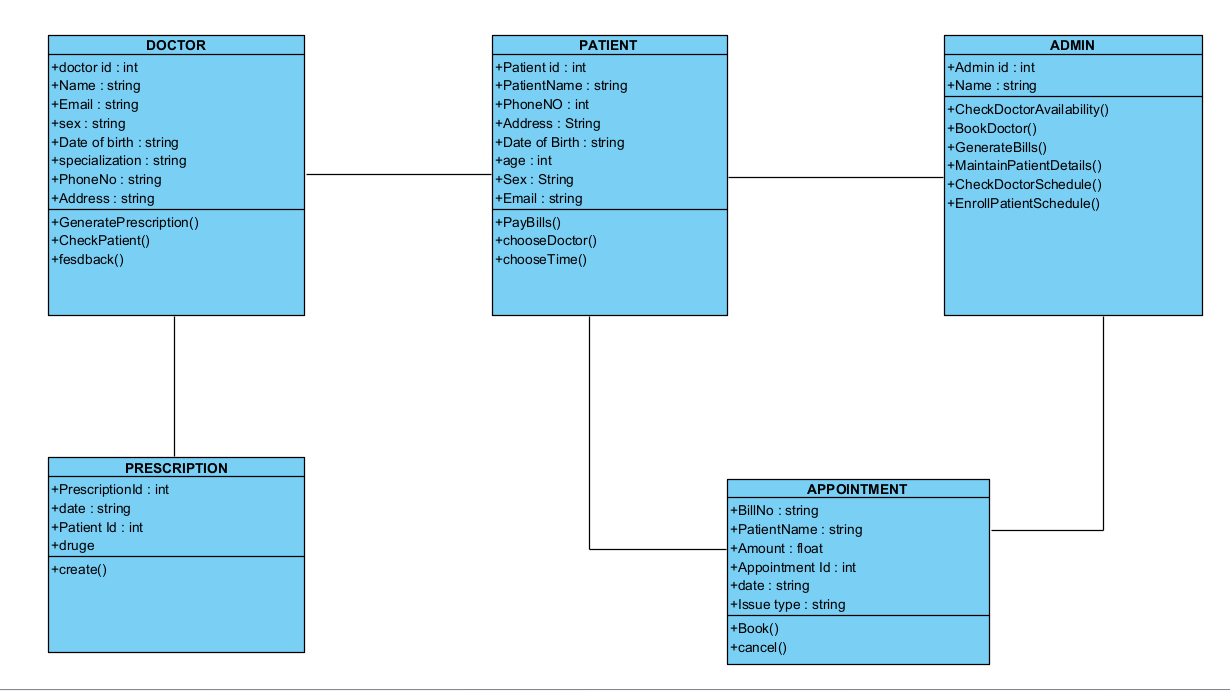
PATIENT



ACTIVITY DIAGRAM

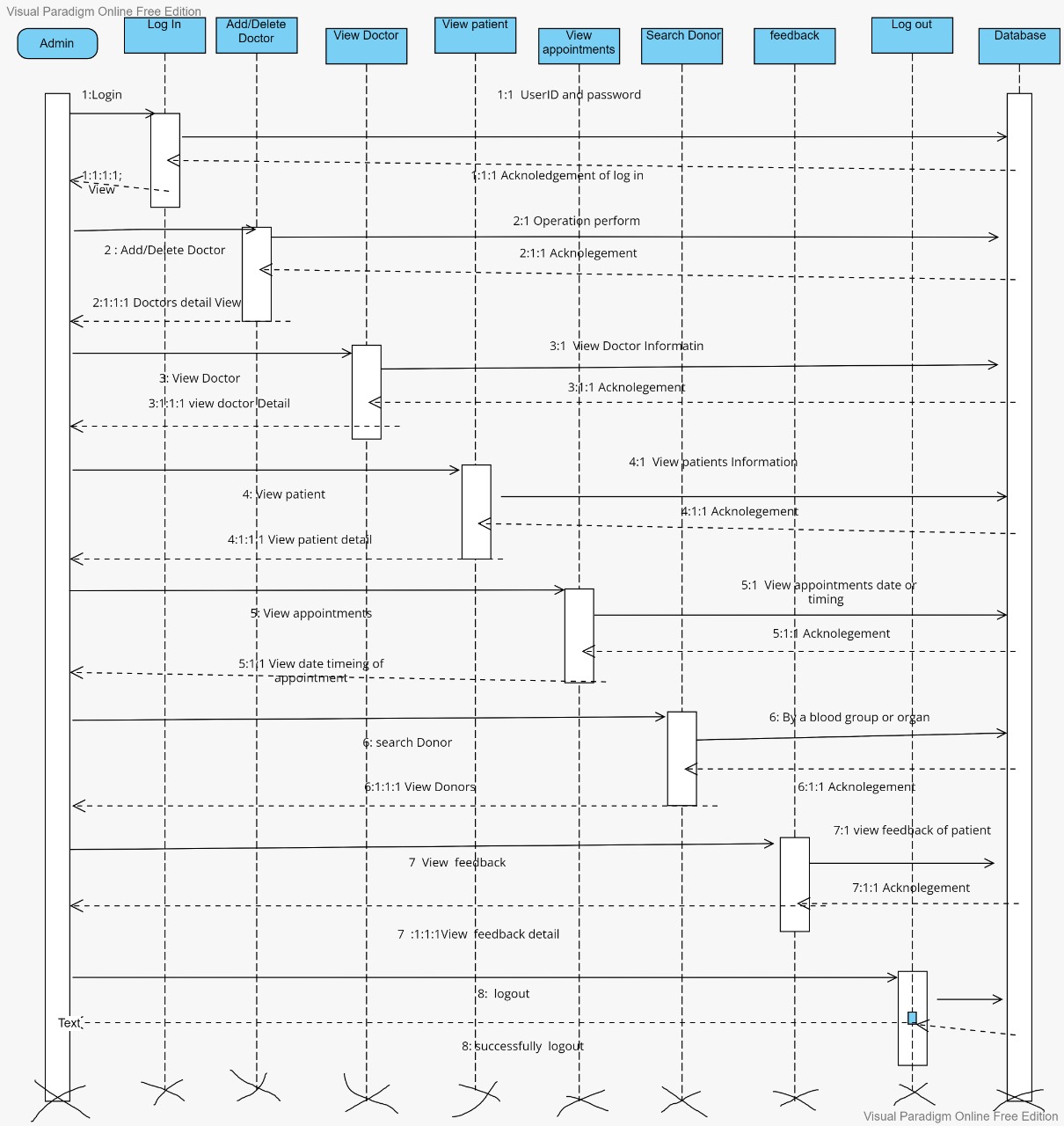


CLASS DIAGRAM

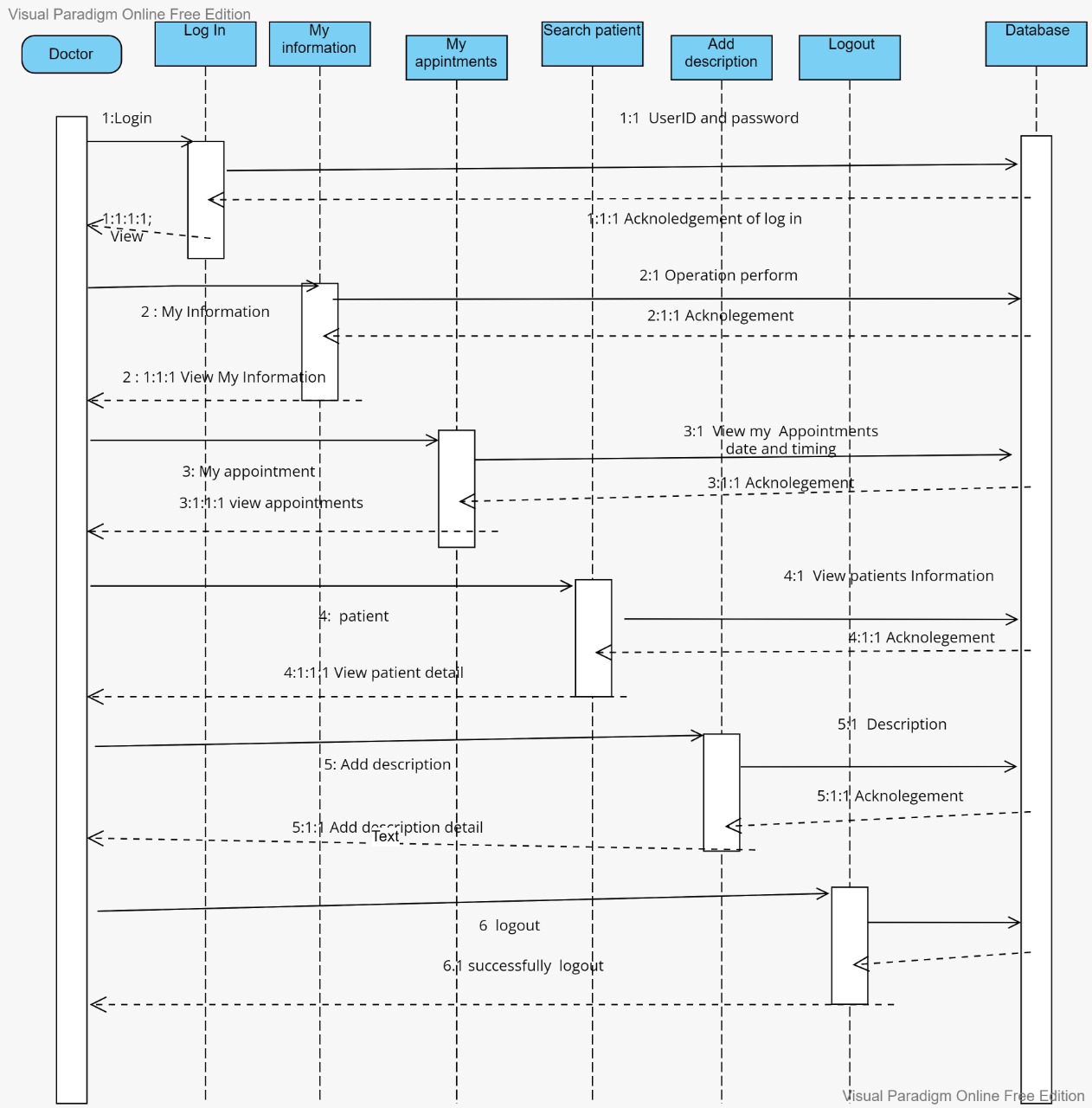


SEQUENCE DIAGRAM

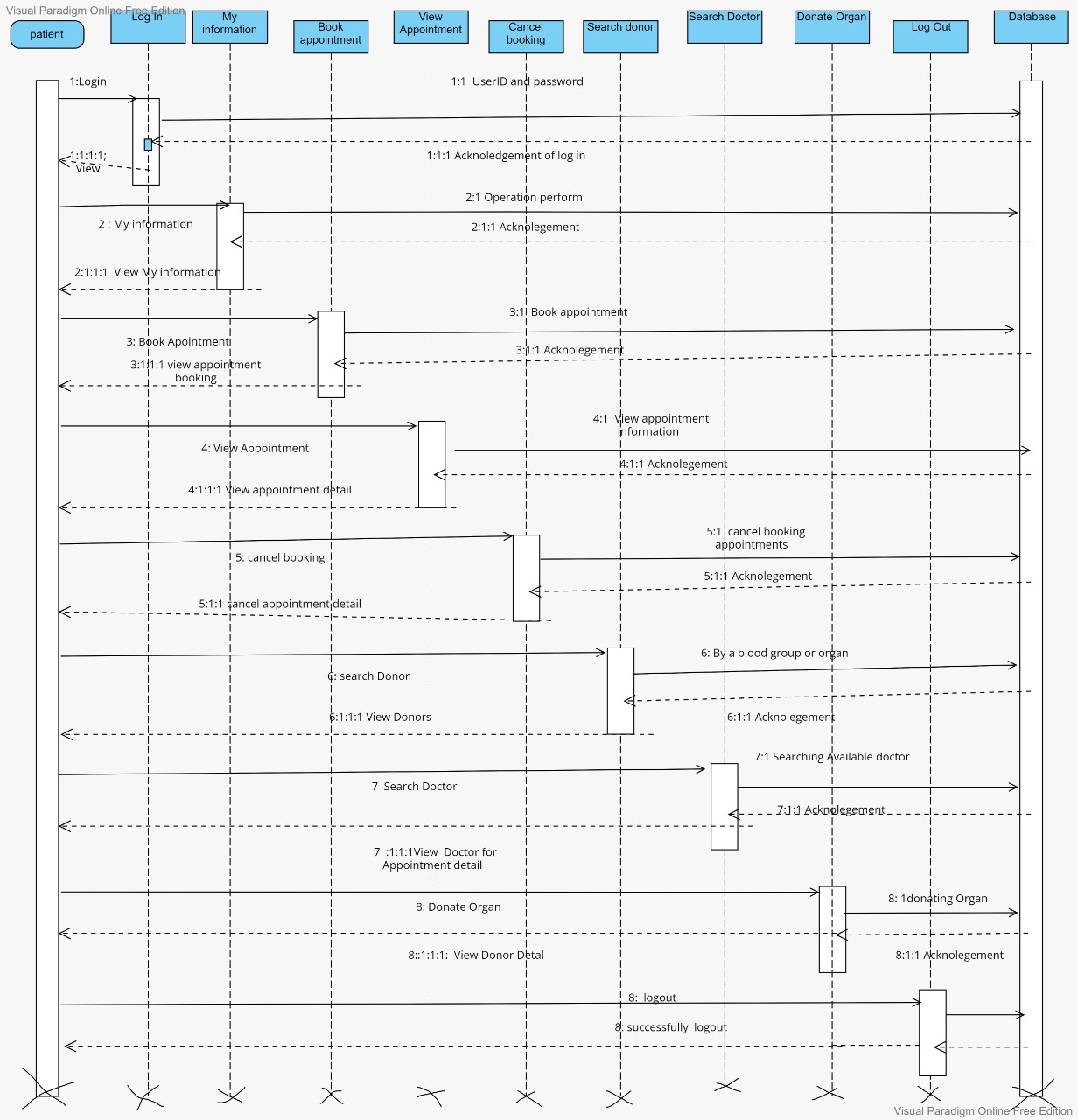
Sequence diagram for admin



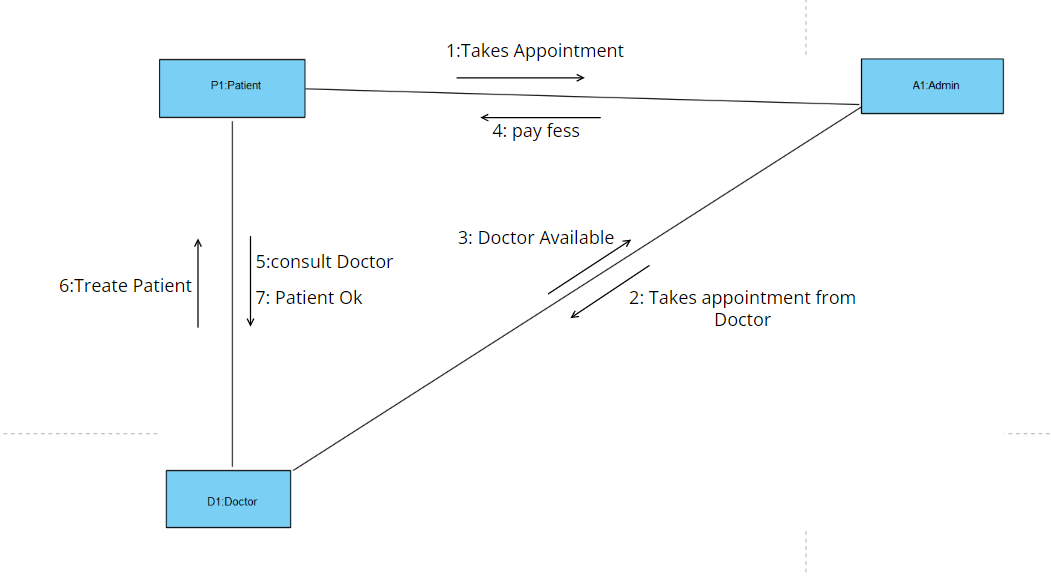
Sequence diagram for doctor



Sequence diagram for patient



COLLABORATION DIAGRAM



DATA DICTIONARY

ADMIN

|  |  |
| --- | --- |
| Table name | Admin |
| description | This table is store information about Admin |
| Primary key | Admin\_id |
| Foreign key | - |

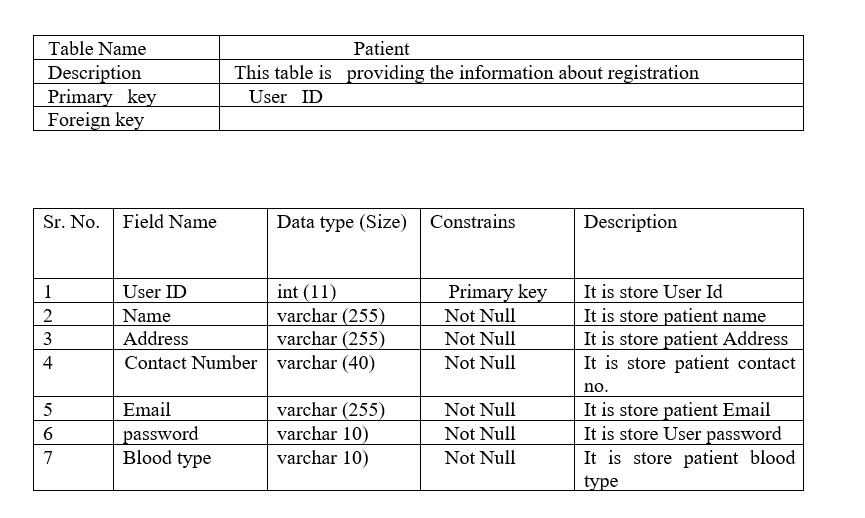
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.no. | Field name | Data type | constraints | Description |
| 1 | Admin\_id | int | Primary key | It is store in admin id |
| 2 | Admin\_Name | Varchar (11) | NOT NULL | It is store in admin name |
| **3** | password | Varchar (30) | NOT NULL | It is store in admin password |
| 4 | Contact\_No | Varchar (12) | NOT NULL | It is store in contact no. of admin |

DOCTOR

|  |  |
| --- | --- |
| Table name | Doctor |
| Description | This table is store information about doctor |
| Primary key | Doctor\_id |
| Foreign key | - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.no. | Field name | Data type | constraints | description |
| 1 | Doctor\_id | Int (11) | Primary key | It is store in doctor id |
| 2 | Doctor name | Varchar (255) | NOT NULL | It is store in doctor name |
| 3 | e- mail | Varchar (50) | NOT NULL | It is store in doctor email |
| 4 | address | Varchar (255) | NOT NULL | It is store in doctor address |
| 5 | Contact no. | Varchar (255) | NOT NULL | It is store in doctor contact no. |
| 6 | password | Varchar (255) | NOT NULL | It is store in doctor password |
| 7 | category | Varchar (255) | NOT NULL | It is store in doctor category. |

PATIENT

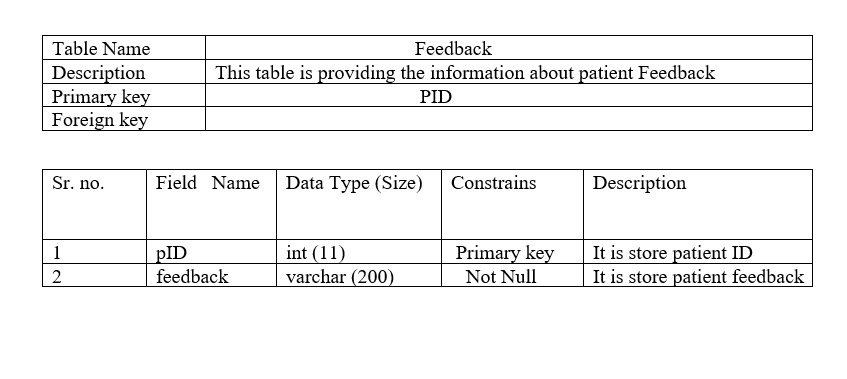


DONOR

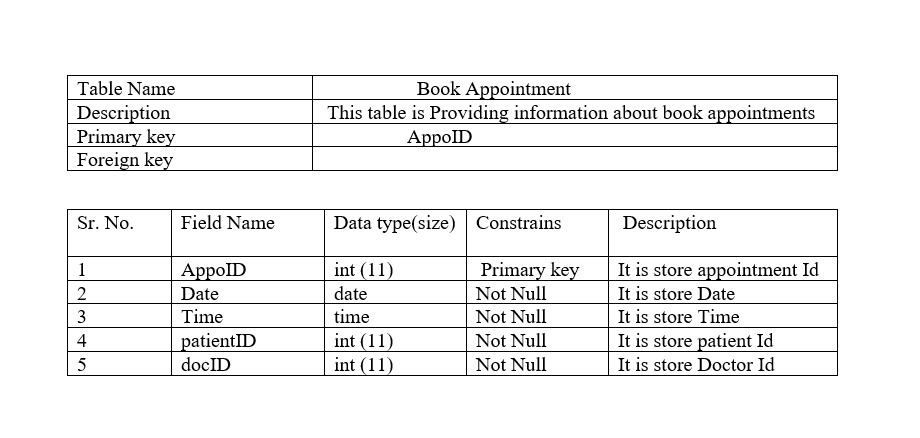
|  |  |
| --- | --- |
| Table name | Donor |
| description | This table is store information about donor |
| Primary key | Donor\_id |
| Foreign key | - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.no. | Field name | Data type | constraints | description |
| 1 | Donor\_id | Int (11) | Primary key | It is store in donor id |
| 2 | Donor name | Varchar (50) | NOT NULL | It is store in donor name |
| 3 | Donor address | Varchar (50) | NOT NULL | It is store in donor address |
| 4 | Contact no. | Varchar (40) | NOT NULL | It is store in donor contact no. |
| 5 | e-mail | Varchar (50) | NOT NULL | It is store in donor e-mail |
| 6 | blood | Varchar (11) | NOT NULL | It is store in donor blood type |
| 7 | organ | Varchar (20) | NOT NULL | It is store in donor organ type |

FEEDBACK

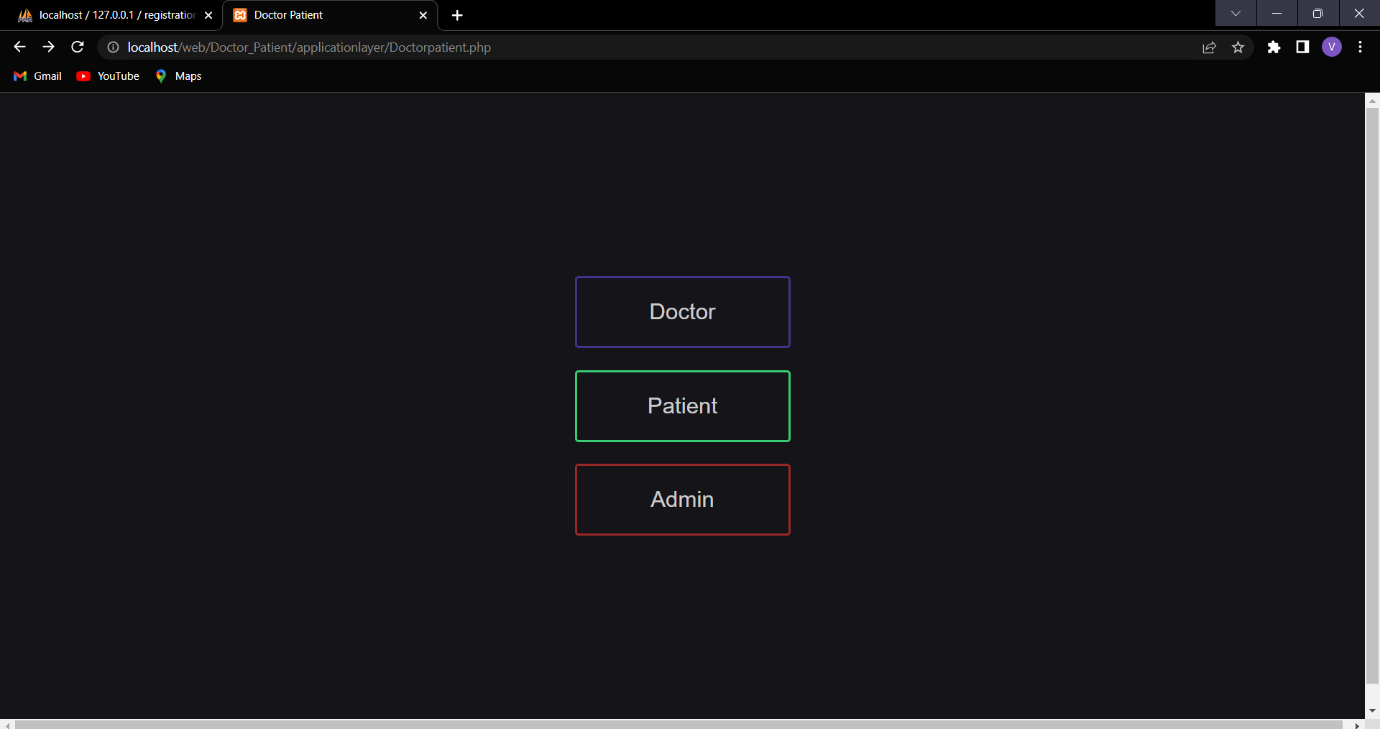


BOOK APPOINTMENT

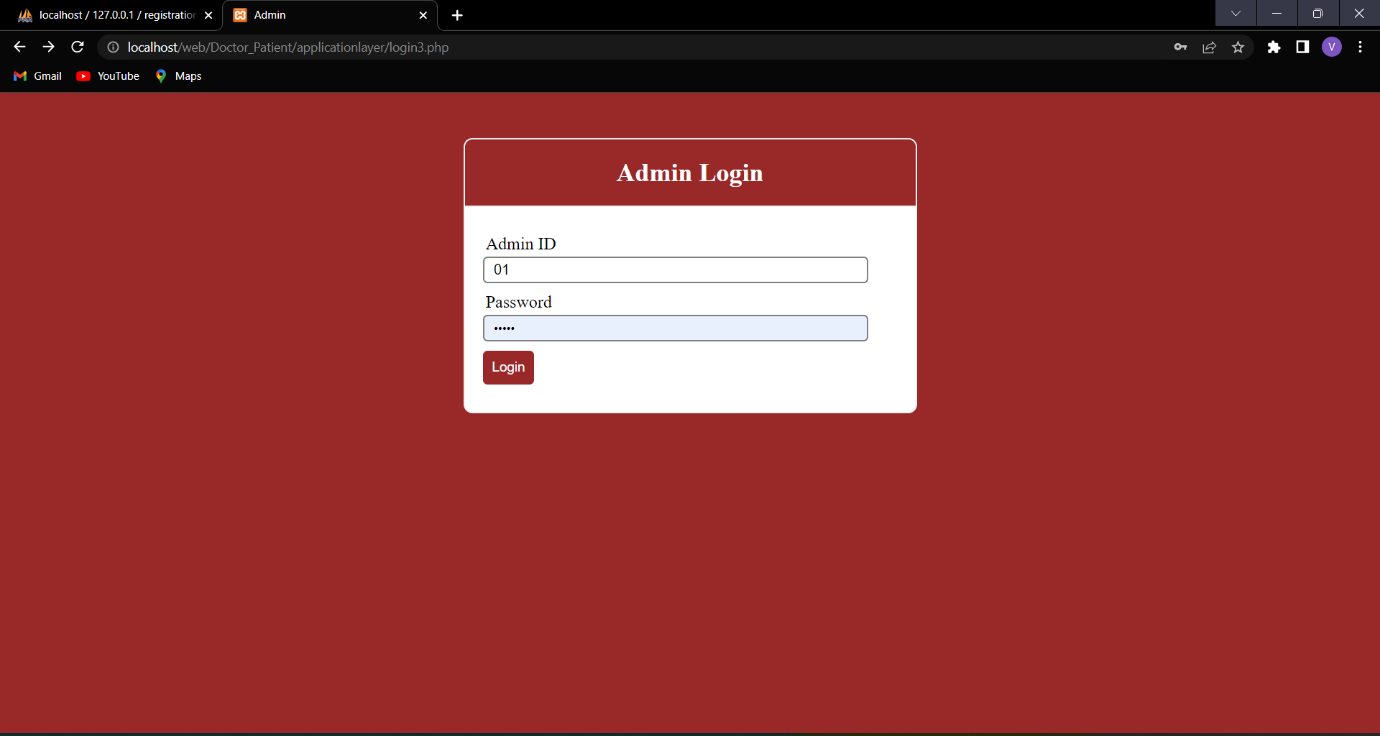


INPUT DESIGN

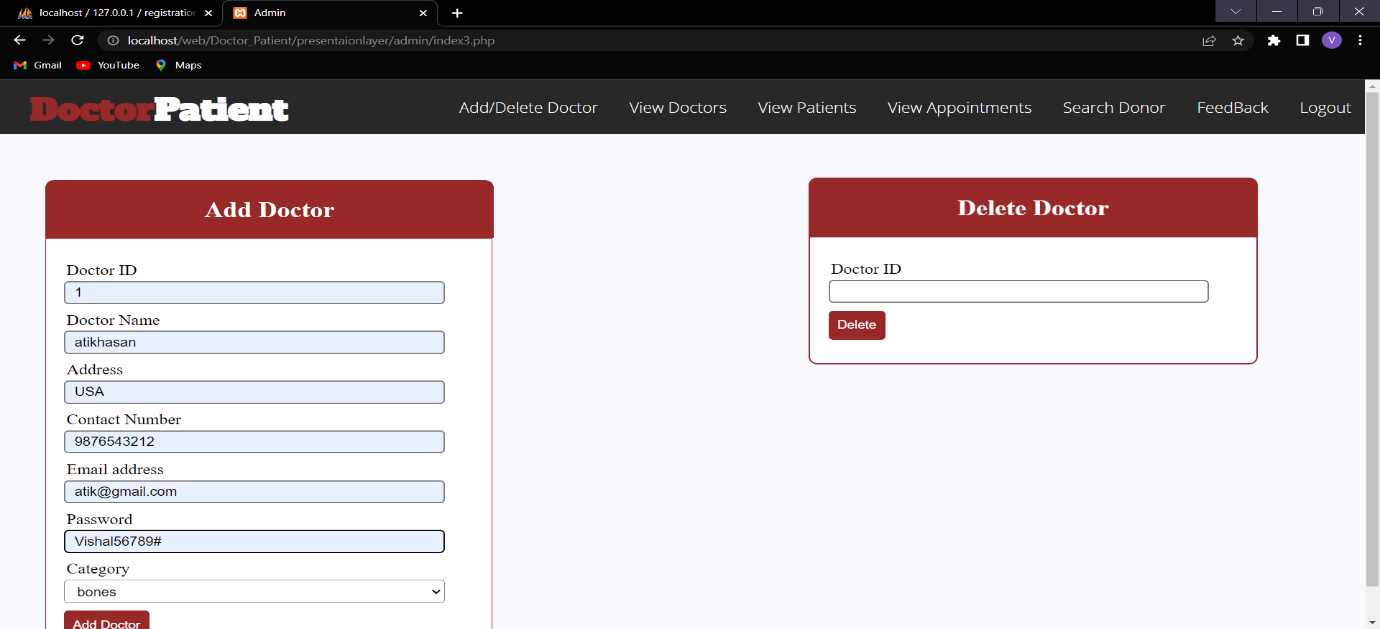
Home page



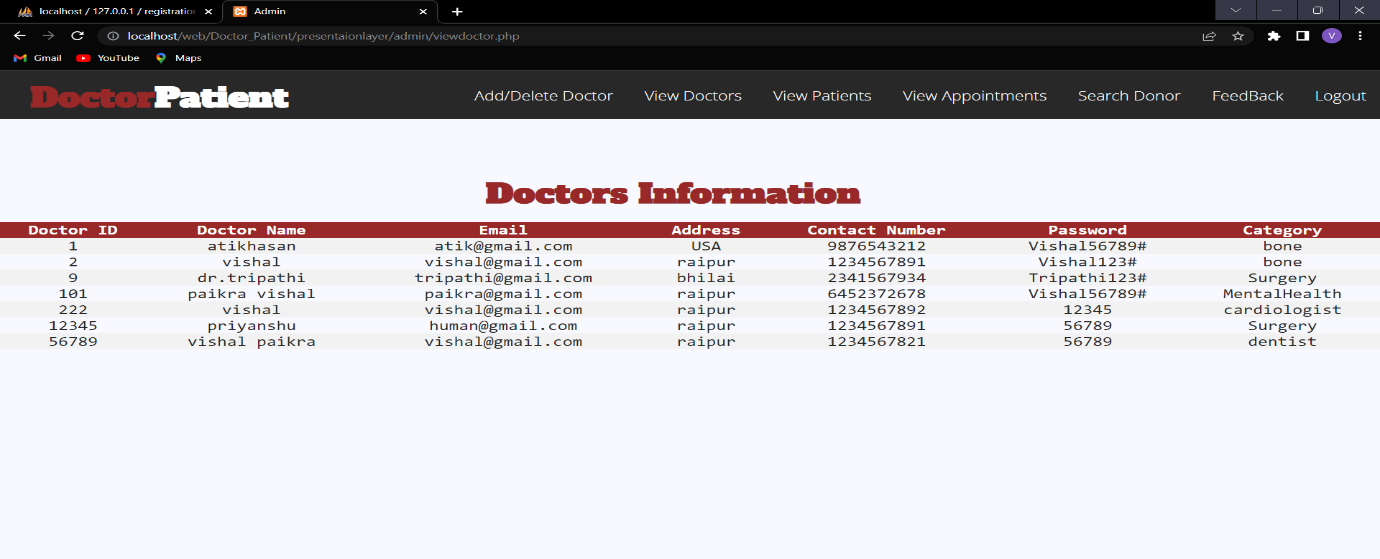
ADMIN LOGIN PAGE



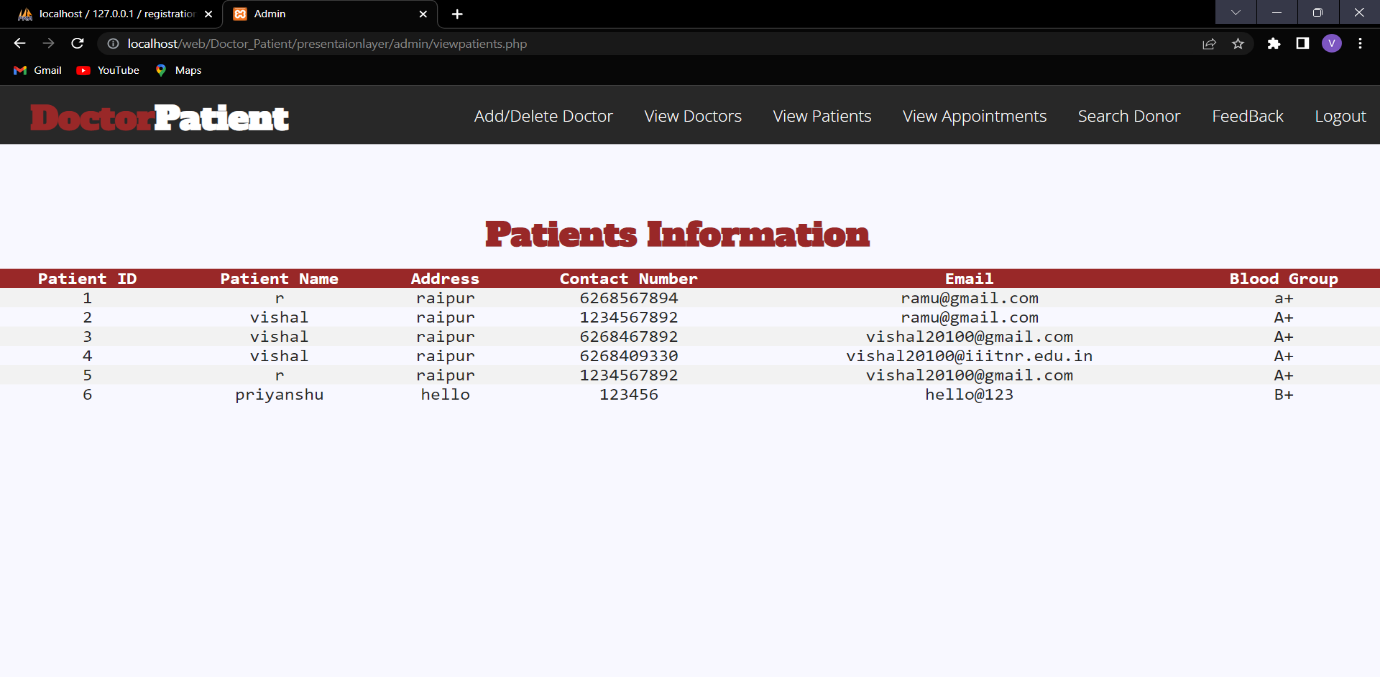
ADMIN HOME PAGE



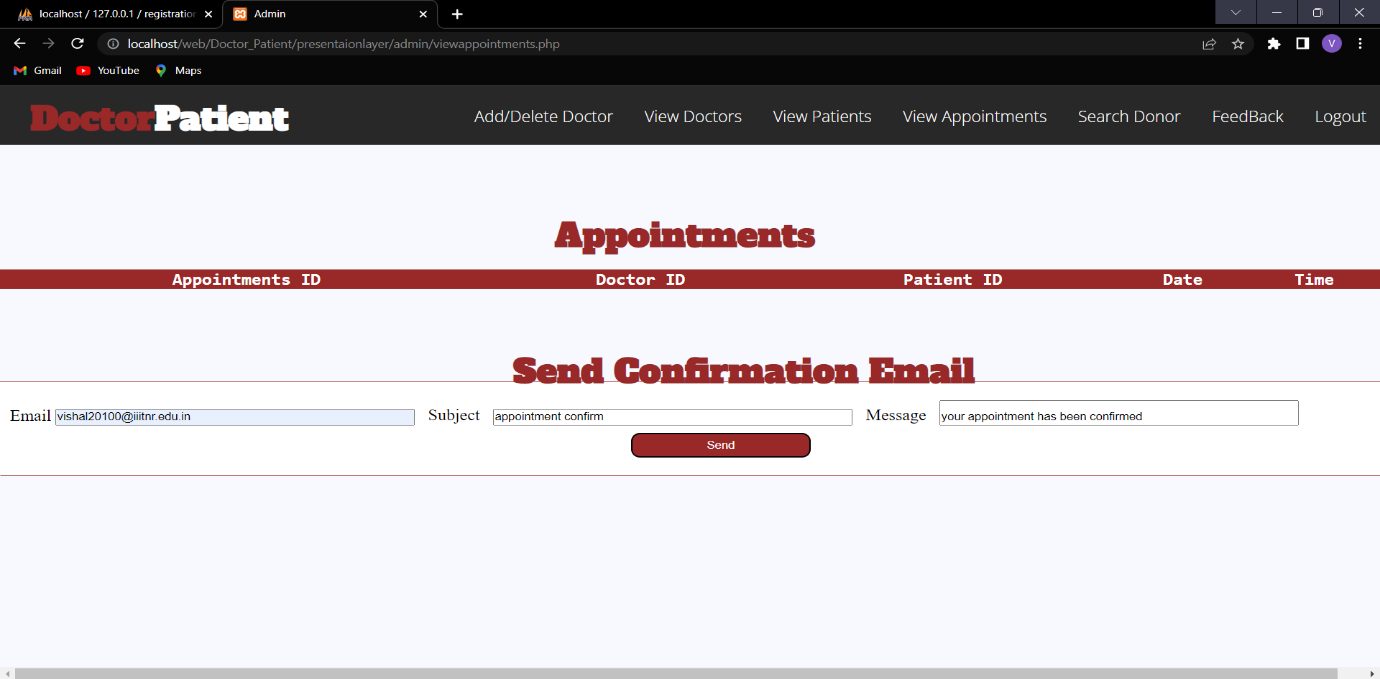
DOCTOR INFORMATION PAGE



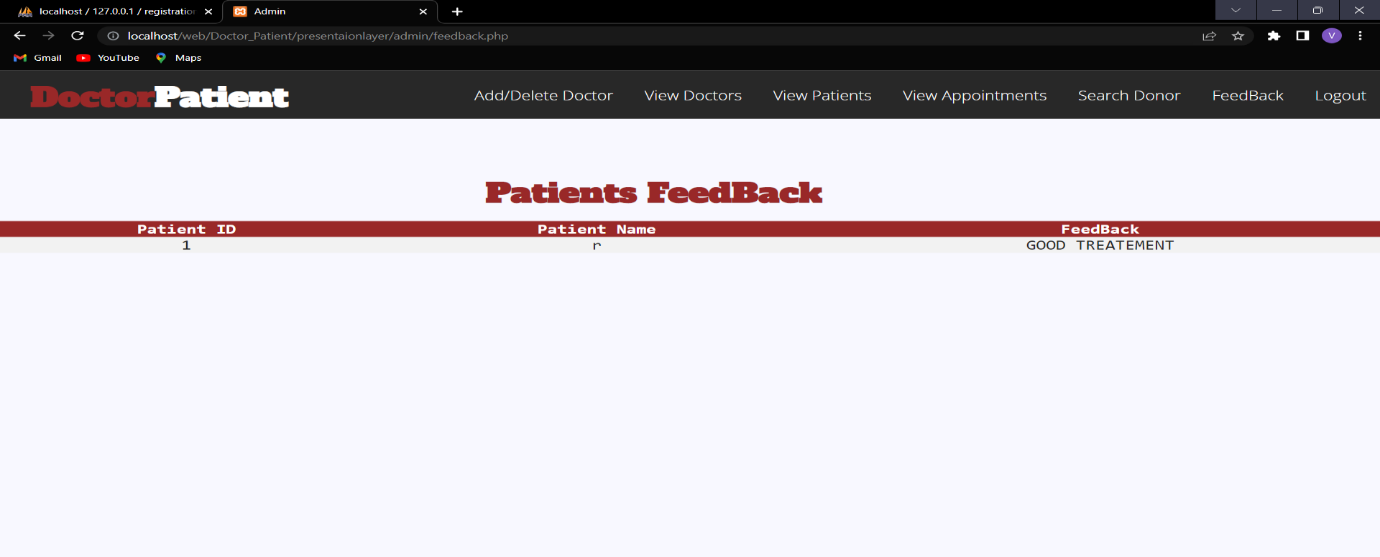
PATIENT INFORMATION PAGE



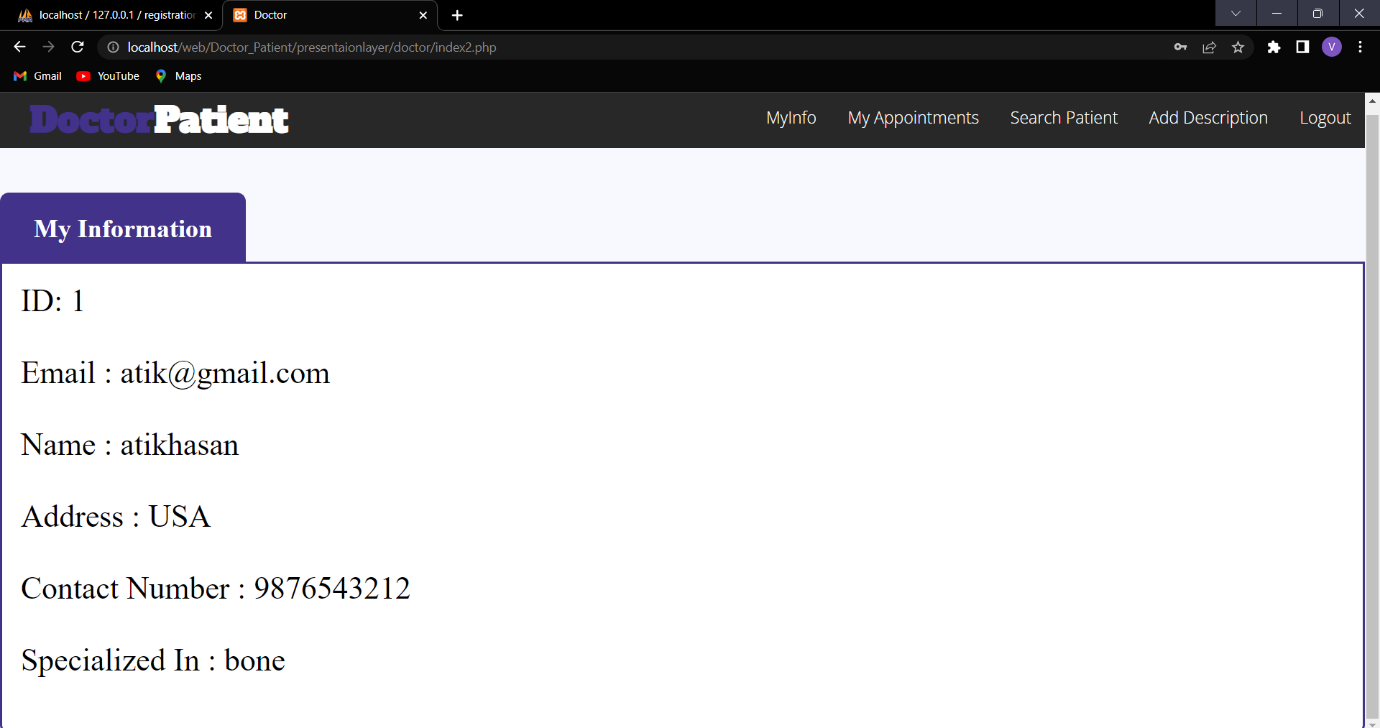
CONFIRMATION EMAIL PAGE



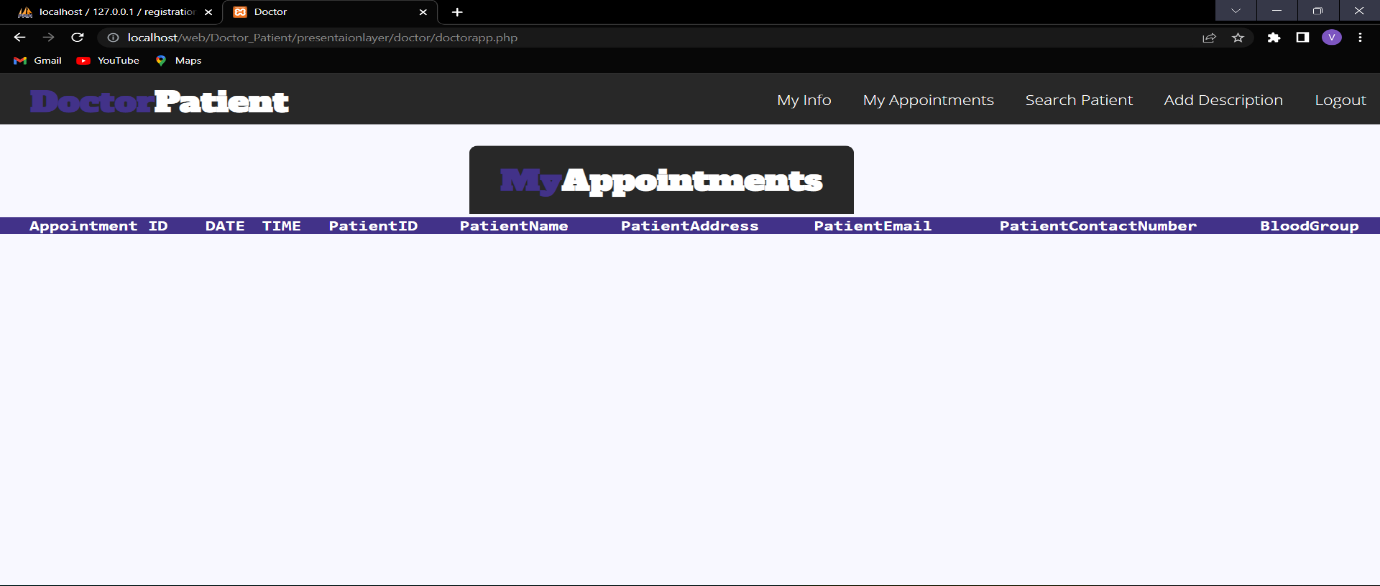
FEEDBACK BY PATIENT



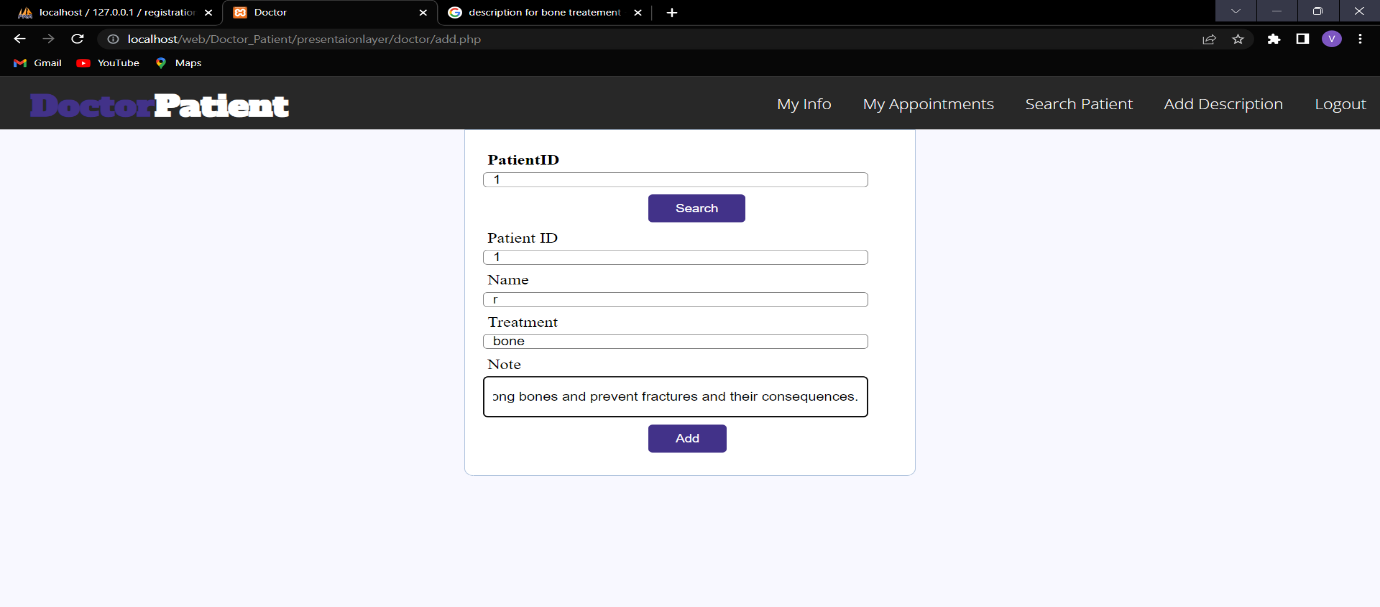
DOCTOR HOME PAGE



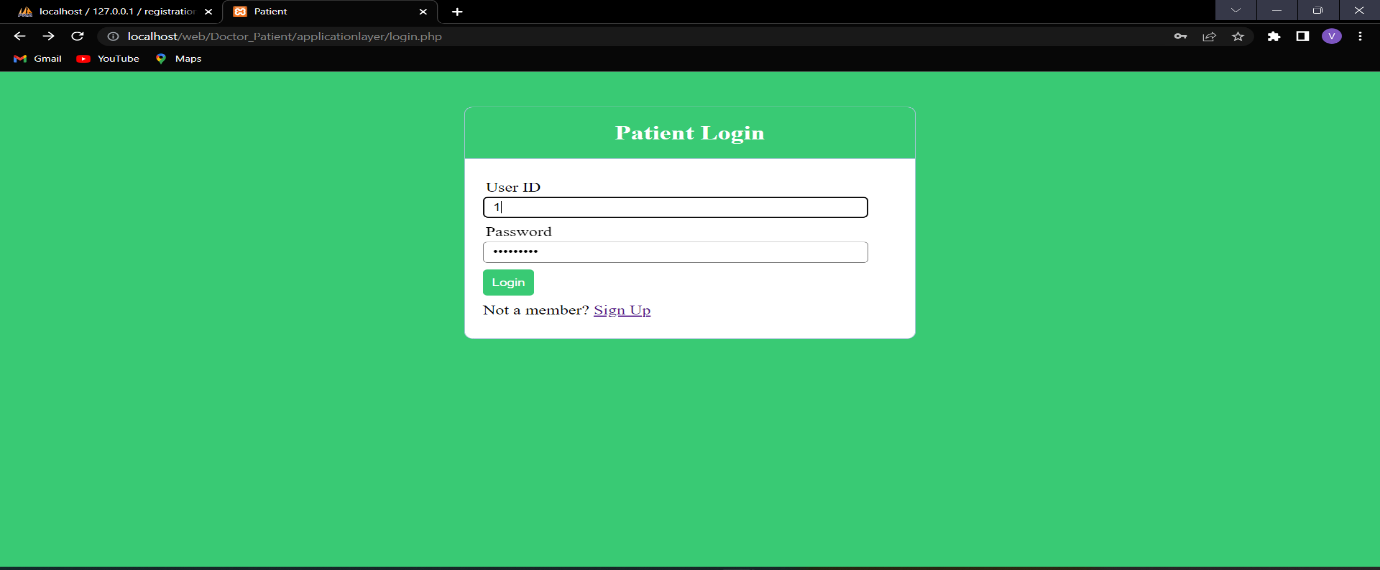
Doctor appointment page



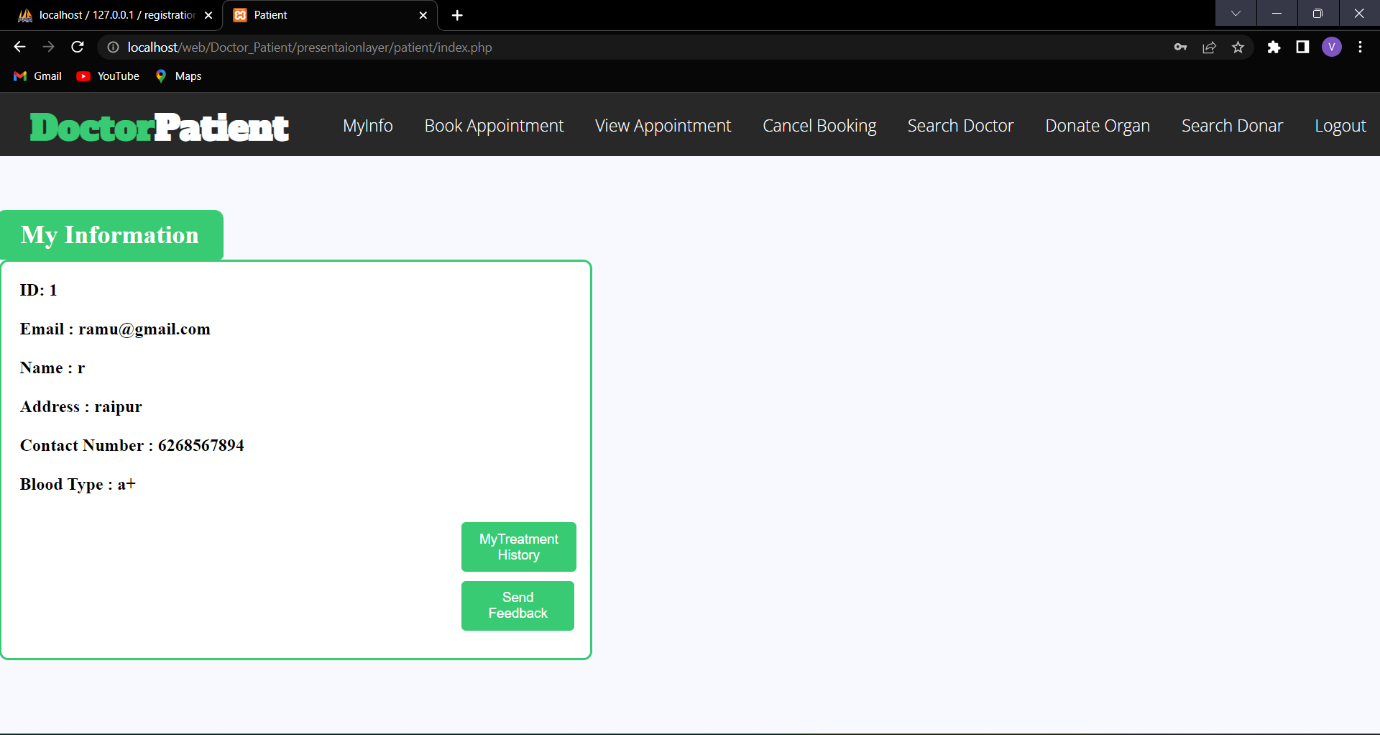
DOCTOR DESCRIPTION



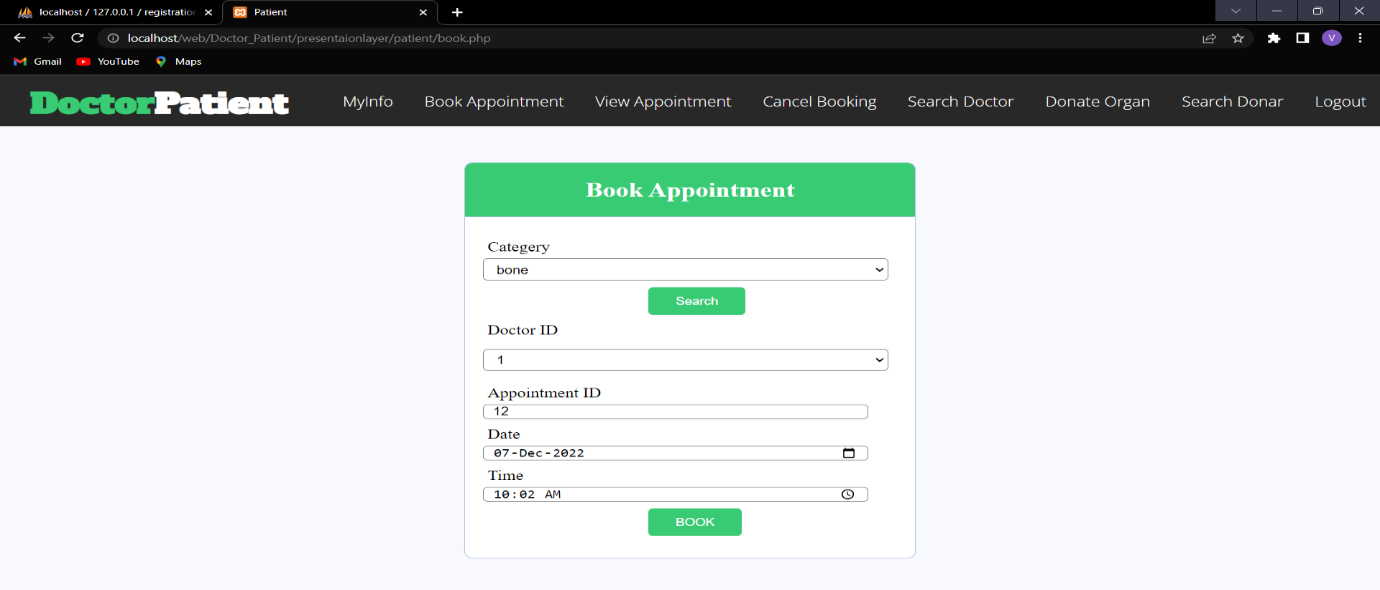
PATIENT LOGIN PAGE



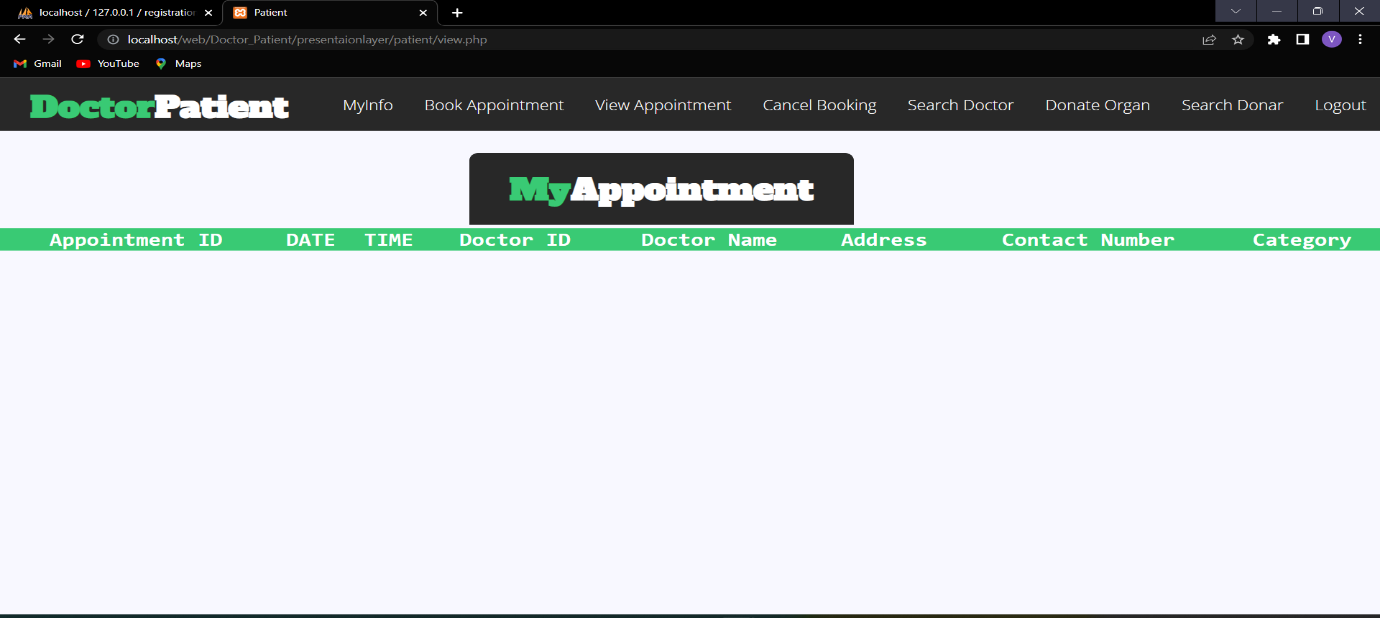
PATIENT HOME PAGE



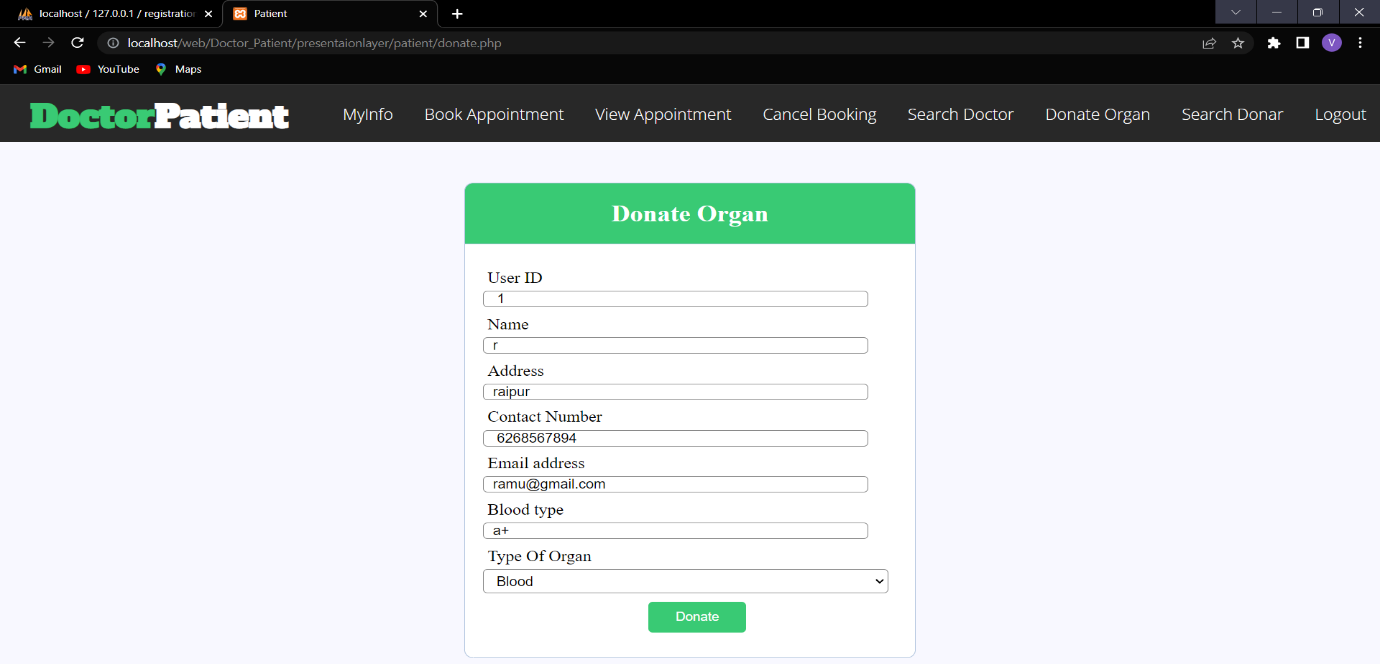
BOOK APPOINTMENT



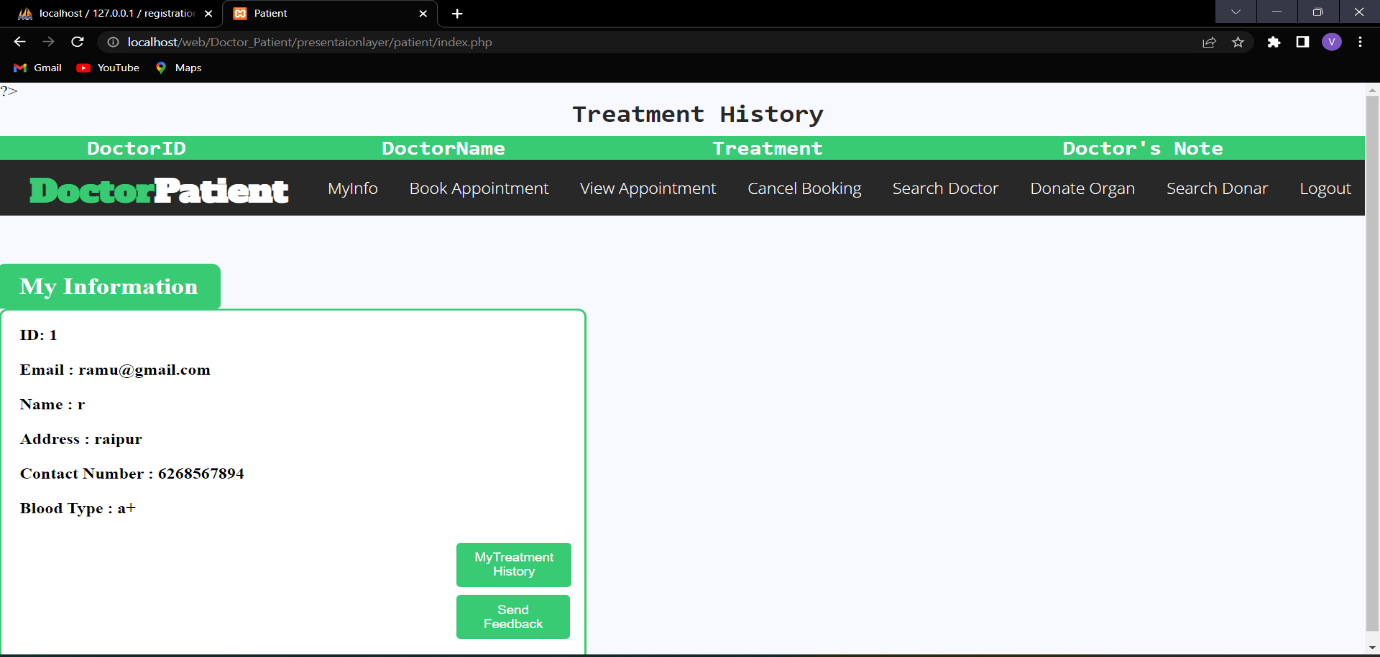
MY APPOINTMENT PAGE



DONATE ORGAN

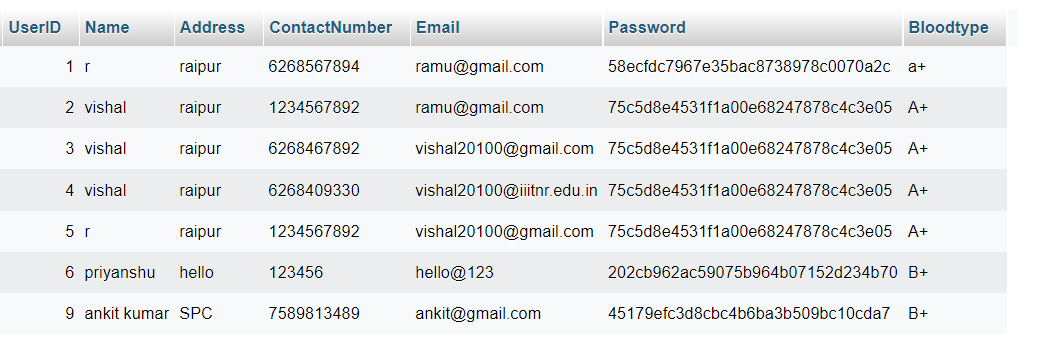


TREATEMENT HISTORY PAGE

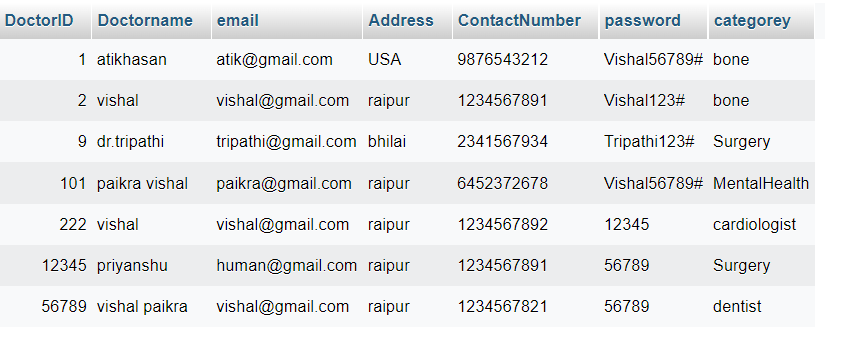


OUTPUT

PATIENT REGISTRATION



DOCTOR



ADMIN



APPOINTMENT



TESTING

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | DATA INPUT | EXPECTED OUTPUT | ACTUAL OUTPUT | PASS/FAIL |
| 1 | All files are empty | Error message | Error message | pass |
| 2 | User id | Error message: invalid user id | Error message: Invalid user id | pass |
| 3 | Admin id | Error message: invalid admin id | Error message: invalid admin id | pass |
| 4 | Doctor id | Error message: invalid doctor id | Error message: invalid doctor id | pass |
| 5 | password | Error message:  password does not match | Error message:  password does not match | pass |
| 6 | Login | Login to the system should be try with the login assigned by the admin and the correct password | Login should be successful and the user should enter into the system | fail |
|  | Login | The system gives an error and denied from the login | Login should fail with an error invalid detail. | pass |
| 7 | User | Login should be allowed and admin get admin home page. | Login successfully and admin get its admin home page | pass |
|  |  | Login should be allowed and patient get patient home page. | Login successfully and patient get its patient page. | pass |
|  |  | Login should be allowed and doctor get doctor home page. | Login successfully and doctor get its doctor home page | pass |
| 8 | Validation test  classes | Pre-defined format must be required in control. | System gives error to enter valid input | pass |
|  |  | Enter data in a compulsory field with required field validation. | Data must be field in compulsory field otherwise its message are displayed. | pass |

POST IMPLEMENTATION

* The website application is self-maintained.
* This website support multiple doctors, admin and user.
* Further we can add nurse.

BIBLOGRAPHY

* BOOK REFERENCE

My Doctor Appointment Log Book and Journal

AUTHOR: OLIVIA DAVENPORT

* WEB REFERENCE

1. <https://nevonprojects.com/doctor-appointment-booking-system/>
2. <https://projectworlds.in/free-projects/php-projects/online-doctor-appointment-booking-system-php-and-mysql/>
3. <https://www.webslesson.info/2021/02/online-doctor-appointment-system-project-in-php-mysql.html>