INTRODUCTION OF HEALTHCARE AND TELEMEDICINE

1. INTRODUCTION

"Health care" involves the organized provision of medical services and interventions to maintain or improve individual and community well-being. It encompasses a spectrum of practices, from preventive measures to therapeutic interventions, delivered by various healthcare professionals in diverse settings.

Telemedicine is a subset of healthcare leveraging digital communication technologies for remote medical consultations, diagnostics, and treatment. It transcends geographical constraints, offering virtual healthcare services that enhance accessibility and efficiency. Telemedicine includes virtual consultations, remote monitoring, and digital health records, providing cost-effective and convenient healthcare solutions, especially in situations where in-person visits are challenging.

1.1.ADVANTAGES:

- Telemedicine can help to improve access to care, especially for patients in rural and underserved areas.
- Telemedicine can help to improve continuity of care by providing more frequent and regular contact between patients and their care providers.
- Telemedicine can help to save on costs by reducing the need for travel and accommodation for patients and care providers.
- Telemedicine can offer more flexible appointment times and locations.
- Telemedicine can be more convenient for patients as they can receive care from the comfort of their own homes.
- Telemedicine can help to improve patient satisfaction by providing a more convenient and personalized care experience.
- Telemedicine can help to increase access to specialists, who may not be available in a patient's local area.

1.2.OBJECTIVES:

- Enable access to healthcare services beyond geoghraphical constraints.
- Streamline healthcare processes through digital communication technologies.
- Reduce healthcare costs by minimizing the need for physical infrastructure and In –person visits.
- Facilities easy access to medical consultations, diagnostics, and treatment.
- Promote equal access to services , overcoming barriers to traditional in –person care

1.3. GOALS:

- Improving accessibility to medical services
- Enhancing patient's outcomes.
- Reducing healthcare cost.
- Facilitating remote consultations for timely interventions.
- Ensuring data security and privacy of patient's records.

2. PHP:

PHP is a server-side scripting language designed for web development but also used as a general purpose programming language. **PHP** is now installed on more than 244 million website and 2.1 million web servers. Originally created by **Rasmus Lerdorf** in 1995, the reference implementation of **PHP** is now produced by The **PHP** Group. While **PHP** originally stood for **Personal Home Page**, it now stands for **PHP; Hypertext Preprocessor,** a recursive acronym. **PHP** code is interpreted by a web server with a **PHP** processor module, which generates the resulting web page: **PHP** commands can be embedded directly into an **HTML** source document. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications**. PHP** is free software released under the **PHP** License. **PHP** can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.

2.1. Php Syntax:

```
< ? Php
Echo 'Hello Words';
?>
```

2.2. Why we use PHP?

You have obviously head of a number of programming language out there you may be wondering why we should want to use PHP as our poison for the web programming. Below are some of the compelling reasons.

- PHP is open source and free
- Large community document.
- It is regular updated to keep abreast with the latest technology trends.

3. HTML

HTML stands for **Hypertext Markup Language** for creating web pages.

- HTML stands for hypertext markup language.
- It describes structure of web pages.
- HTML elements are represented by tags.
- It consist series of elements.

3.1. HTML Example:

```
<html>
   <head>
        <title> page title </title>

   </head>
        <body>
             <h1> my first heading </h1>
             <p> my first paragraph </p>
        </body>
</html>
```

3.2. Advantages of HTML:

- The first advantage it is widely used.
- Every browser support HTML language.
- Easy to learn and use.
- It is by default in every window so you don't need to purchase extra software.
- We can integrate HTML with CSS, JavaScript, and Php etc.

3.3. Disadvantages of HTML:

- It can create only static and plain pages so if we need dynamic pages then **HTML.**
- Need to write a lot of codes for making simple webpage.
- Security features are not good in **HTML.**
- If we need to write long code for making a webpage then it produces some complexity.

3.4. Why HTML is used in web pages:

Web developing includes two main sections.

Backend: codes that are written by **Python, PHP, ASP.Net**, and **Go** language to name but a few by the developer.

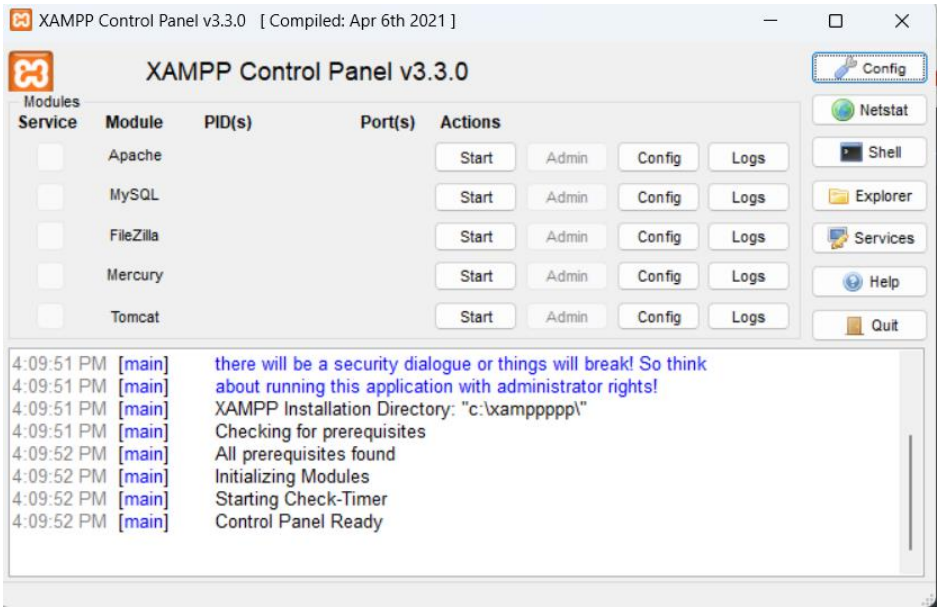
Frontend: which is makeup showed by clients or users browsers and for doing this we should use **HTML (Hypertext Markup Language),** it just shows some elements for users and doesn't run any functions.

When we go to a specific **URL**, your request is sent to your desired server and it'll render for your **HTML** of the site in fact the server runs any server-side functions.

4. Xampp

XAMPP is a free and open-source cross-platform web server solution stack package developed by **Apache Friends**, consisting mainly of the Apache **HTTPServer, MariaDB** database, and interpreters for scripts written in the **PHP** and **Perl programming languages.** Since most actual web server deployments use the same components as **XAMPP**, it makes transitioning from a local test server to a live server possible.

XAMPP's ease of deployment means a **WAMP** or **LAMP** stack can be installed quickly and simply on an **operating system** by a developer, with the advantage that common add-in applications such as **Word Press** and **Joomla!**Can also be installed with similar ease using **Bitnami.**



I. Xampp

5. SYSTEM ANALYSIS:

5.1. Main Problem With Existing System:

- **Time Consuming:**

  In our current system, all the processes are carried out by human so naturally it. Requires more time and, in that case, it will require more time to solve problem.

- **Difficult in Implementing:**

  It is difficult to implement the solution because sometime the problem is big and cannot solve by any people. So we must have to prepare about this always to do what in this.

5.2. REQUIREMENT SPECIFICATION:

a) Software Requirements:

- Operating System: - Window 8, Window 10, Window 11, Linux etc.
- Framework:- Laravel 5.8*
- Server: - Xampp, Wamp, etc.
- Database:- MySQL(phpmyadmin)
- Text Editor: - Vs. Code, Notepad++ etc.
- Other Software: - Git, Composer etc.

b) Hardware Requirements:

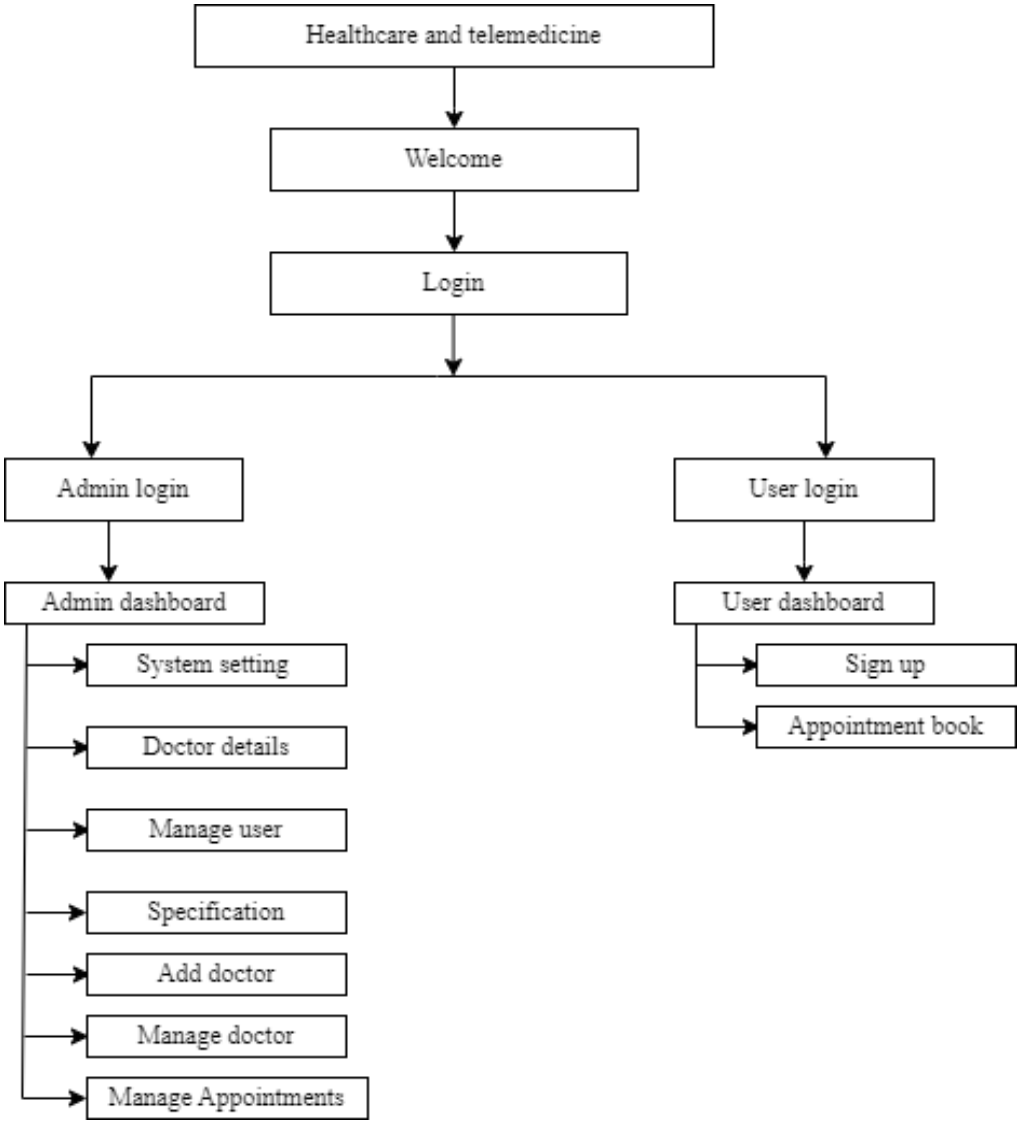
- Processor:- Intel i3 or more
- Processor Speed:- 2.30GHz or more
- Hard Disk:- 100 GB or more
- RAM:- 4GB or more
- Other: - Keyboard, Mouse, etc.

6. SYSTEM DESIGN:

The system design for healthcare and telemedicine involves the integration of various technologies to ensure efficient and accessible healthcare services. The architecture includes secure electronic health record (EHR) systems for storing and managing patient data, interoperability standards for seamless information exchange between healthcare providers, and robust telemedicine platforms for remote consultations. The system prioritizes data security and privacy, employing encryption and authentication measures to protect sensitive health information.

6.1.MODULE:

Module is a diagrammatic representation of functionality within a project. It may have one or many module in software. In other words, module is an approach that subdivides a system into smaller parts called Module that independently created and then in Different Systems. Each of a set of standardized parts or independent unit can be used to construct a more complex structure.



II. Module of Healthcare and telemedicine

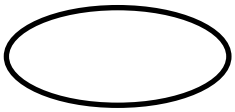
6.2. ER DIAGRAM:

ER diagram stands for Entity Relationship Diagram. It is a High Level Data Diagram. This Diagram is used to define the data elements and relationship for a specified system. The components of ER diagram are explained below:-

a) Entity: An entity may be any object, class, person or place. In the ER diagram, an entity can be represented as Rectangles.

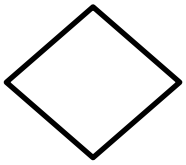
a. Fig: Rectangle

b) Attribute: The attribute is used to describe the property of an entity. Eclipse is used to represent an attribute.

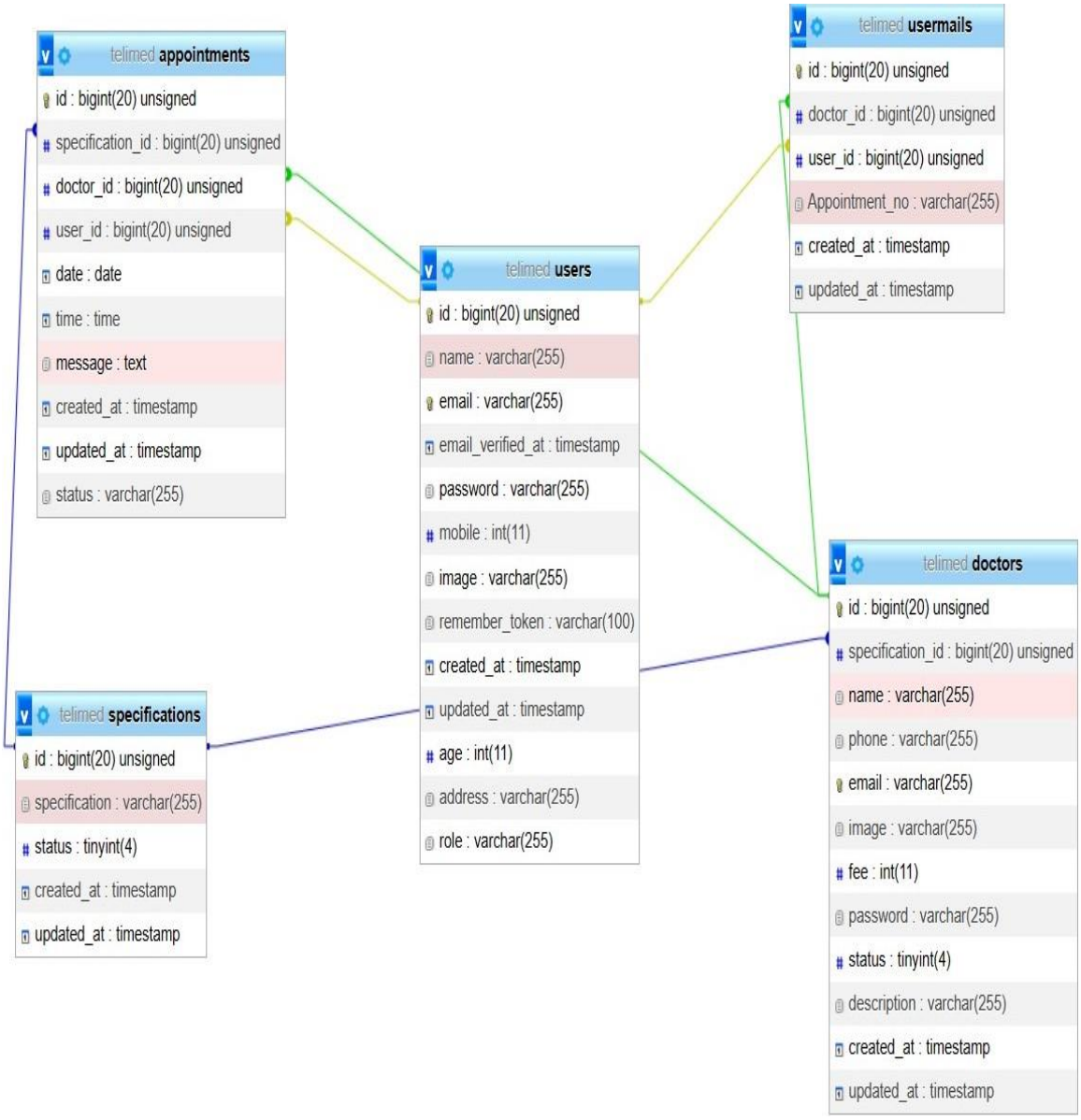


b. Fig: Eclipse

c) Relationship: A Relationship is used to describe the relation between entities. Diamond or Rhombus is used to represent the relationship.



c. Fig: Diamond

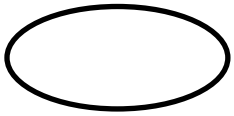


III. ER – Diagram

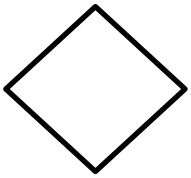
6.3. Flowchart Diagram:

A Flowchart is a visual representation of a process that makes it easy to understand the process at a glance. Flow Charts depict the nature and flow of steps in a process. Steps and decision points of a process are linked by connecting lines and directional arrows showing process flow direction. This makes it easy for anyone to rationally follow the process from beginning to end. It is important to note that each process step is represented by a different symbol showing different types of actions in a process. There are mainly four types of flowchart symbol which are given below:-

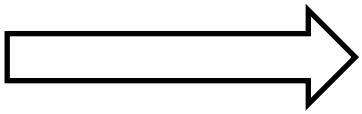
a) **Terminator:** The Terminator symbol represents the starting or ending point of the system.



b) Decision: A diamond represents a decision or branching point. Lines coming out from the diamond indicate different possible situations, leading to different sub processes.



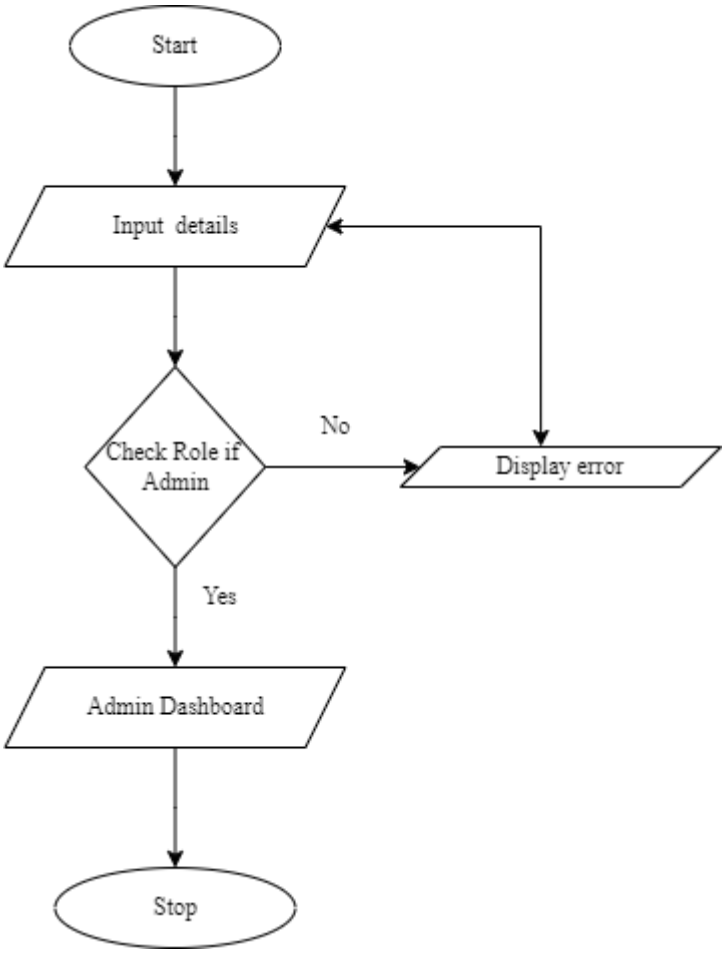
c) Flow: Lines represent the flow of the sequence and direction of a process.



d) Data: It represents information entering or leaving the system. An input might be an order form the customer. Output can be a product to be delivered

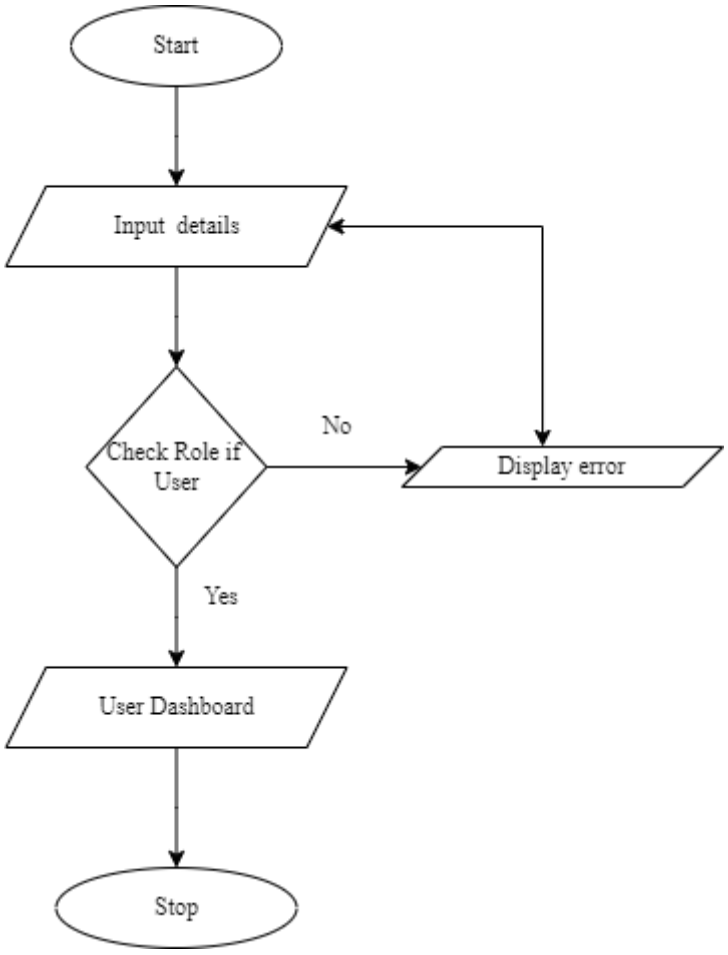
.

6.4. ADMIN LOGIN FLOW CHART:



IV. Fig: Admin Login Flow chart Diagram

6.5. USER LOGIN FLOWCHART :



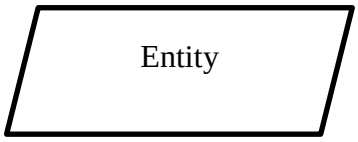
V. Fig: User Login Flowchart Diagram

7. DATAFLOW DIAGRAM:

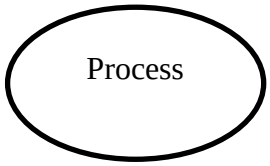
Dataflow Diagram is a graphical representation of flow of data in an information system. It is capable of depicting incoming data flow, outgoing data flow and stored data.

The DFD does not mention anything about how data flows through the system. There is a prominent difference between the DFD and Flowchart. The Flowchart depicts flow of control in program modules. DFD depict flow of data in the system at various levels. DFD does not contain any control or branch elements. The different component of Dataflow Diagram is explained below:-

a) Entities: Entities are source and destination of information data. Entities are represented by a rectangle with their respective names.



b) Process: Activities action taken on and the data are represented by Circle or oval Shapes.



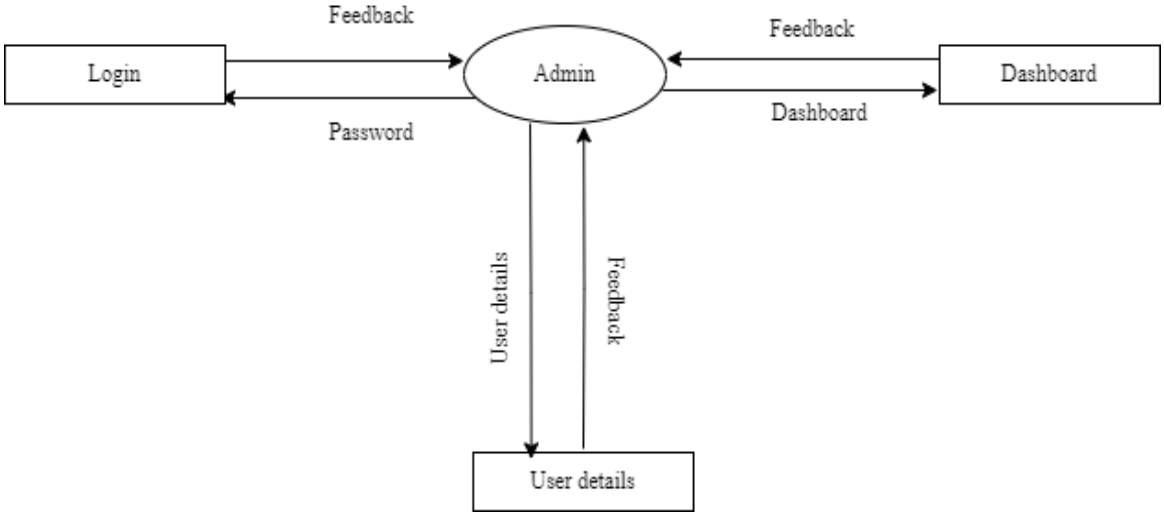
c) Data Storage : There are two variants of data storage- It can either be represented as a rectangle with absence of both smaller sides or as an open sided rectangle with only one side missing.

d) Data Flow: Movement of data is shown by pointed arrows. Data movement is shown from the base of arrow as its source towards head of the arrow as destination.

8. CONTEXT DIAGRAM:

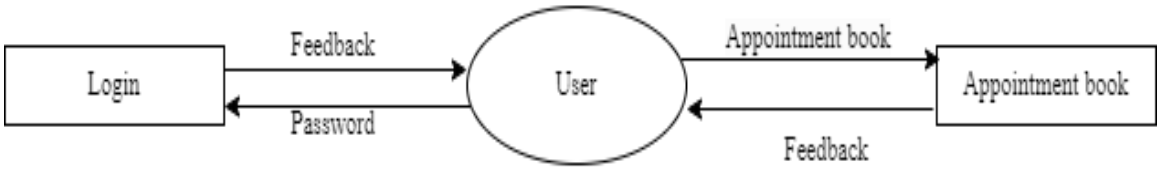
VI. DFD Level 0: Context Diagram

9. ADMIN LOGIN SYSTEM:



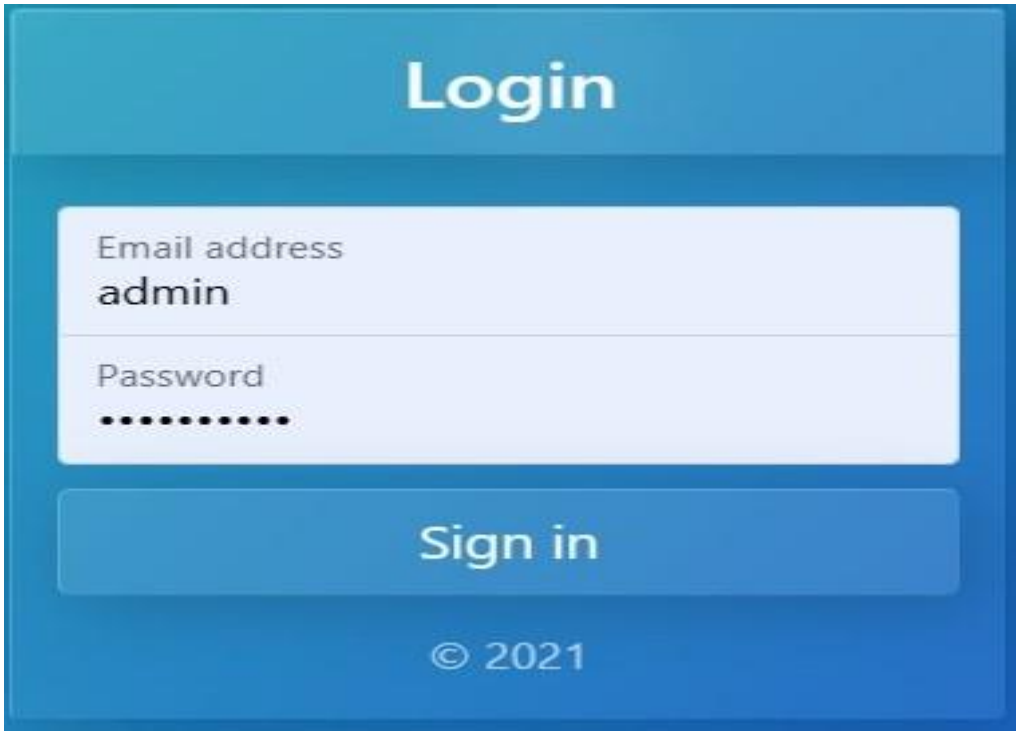
VII. DFD Level 1.0: Admin Login

9.1. User login System:



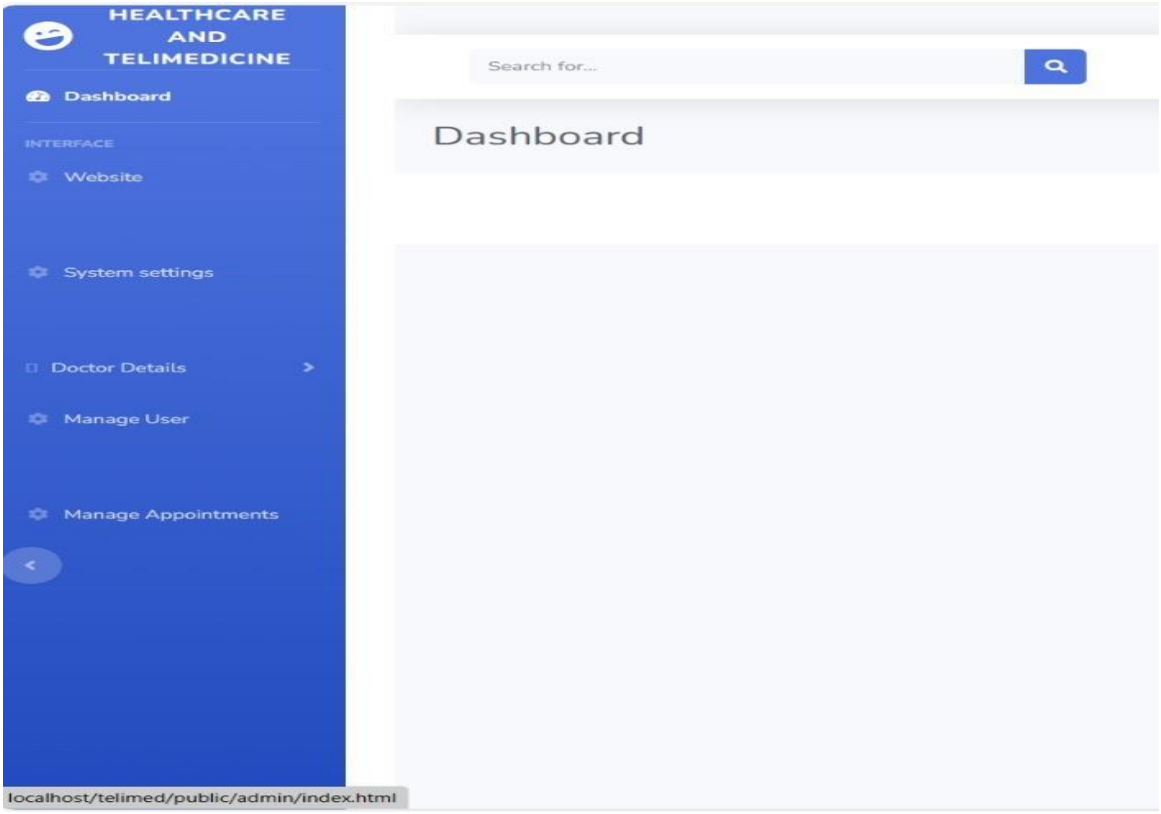
VIII. DFD Level 1.1: User Login System

10. LOGIN PAGE:



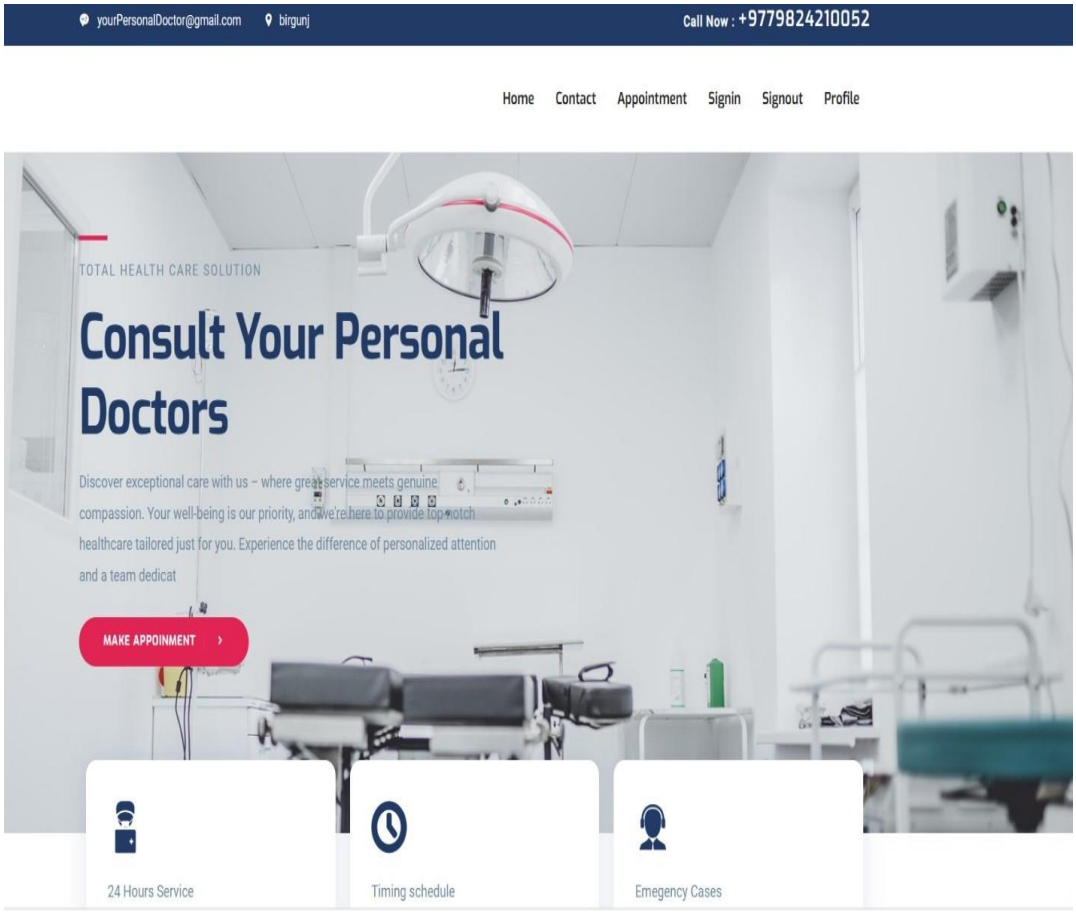
IX. Login Page

10.1. ADMIN HOME PAGE:



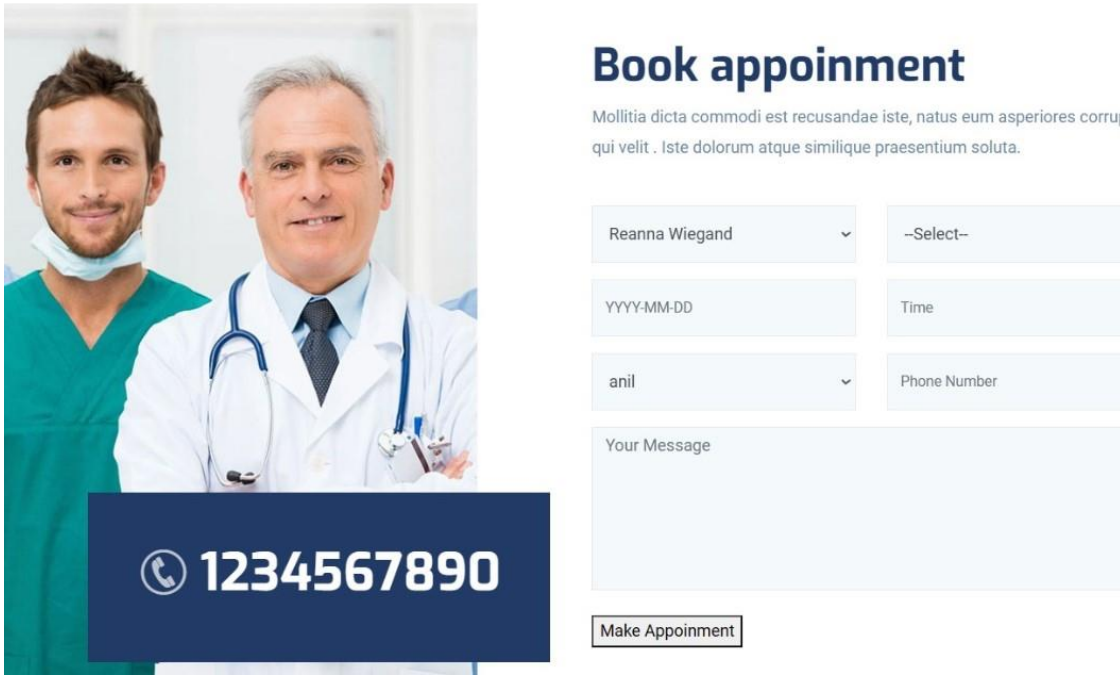
X. Fig : Admin Home Page

10.2. USER HOME PAGE:



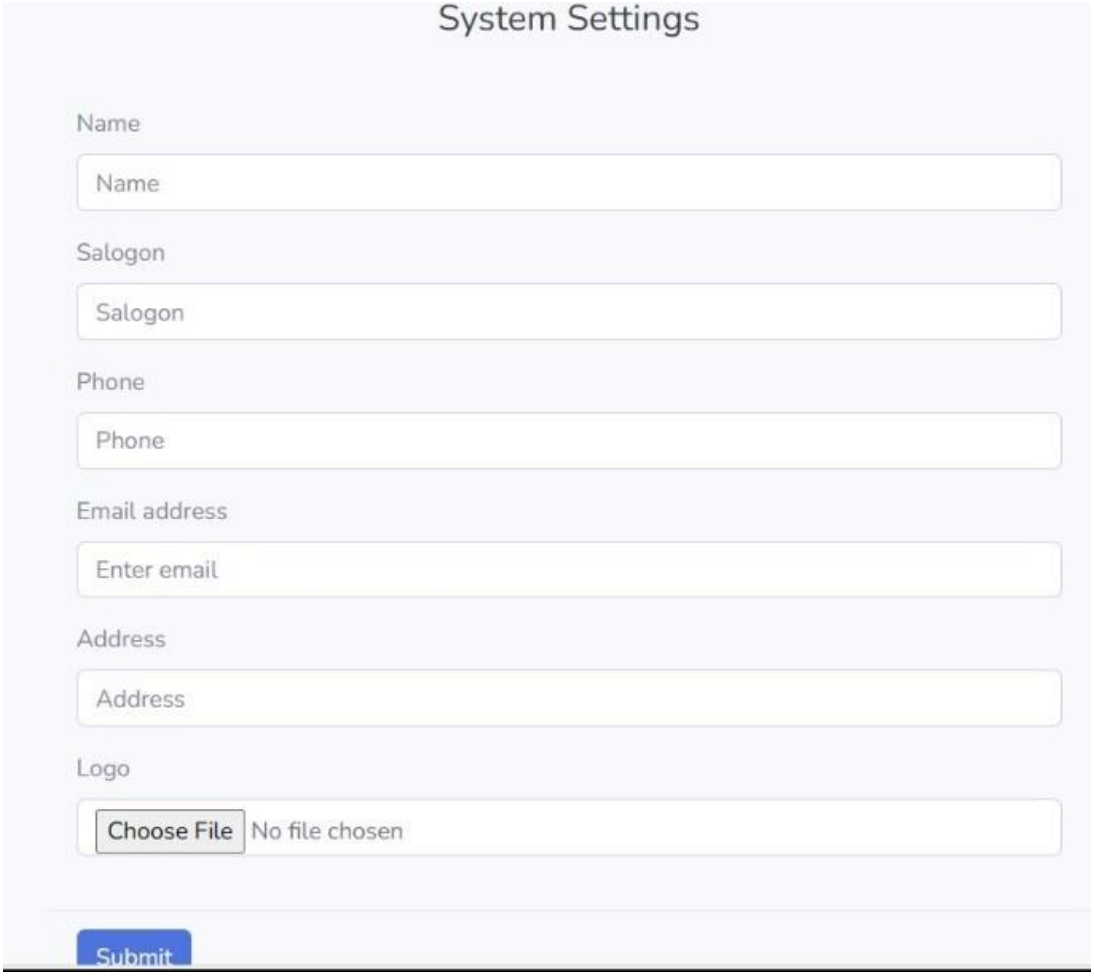
XI. Fig: User Home Page

10.3. USER BOOK APPOINMENT:



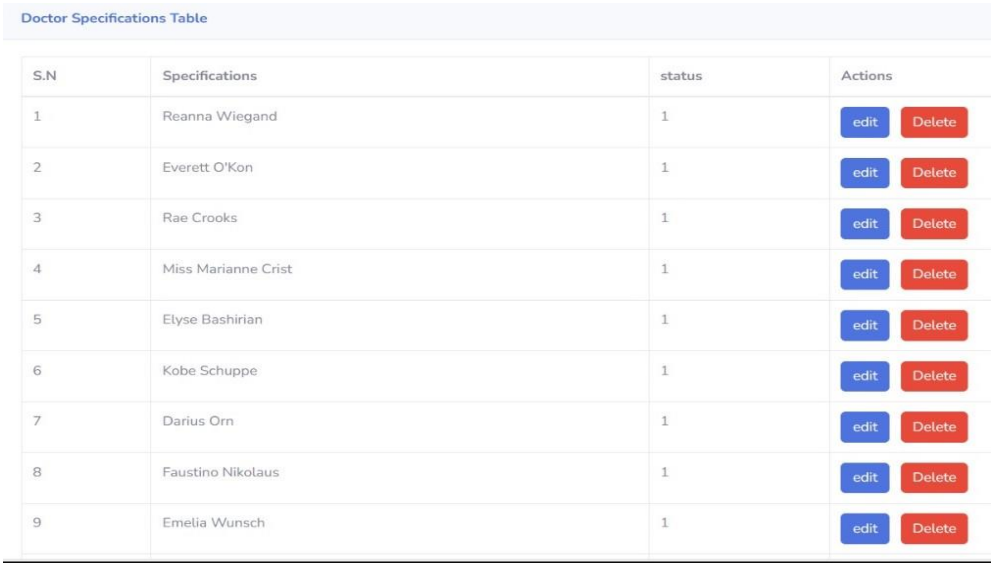
XII. Fig: User Book Appoinment

10.4. SYSTEM SETTING:



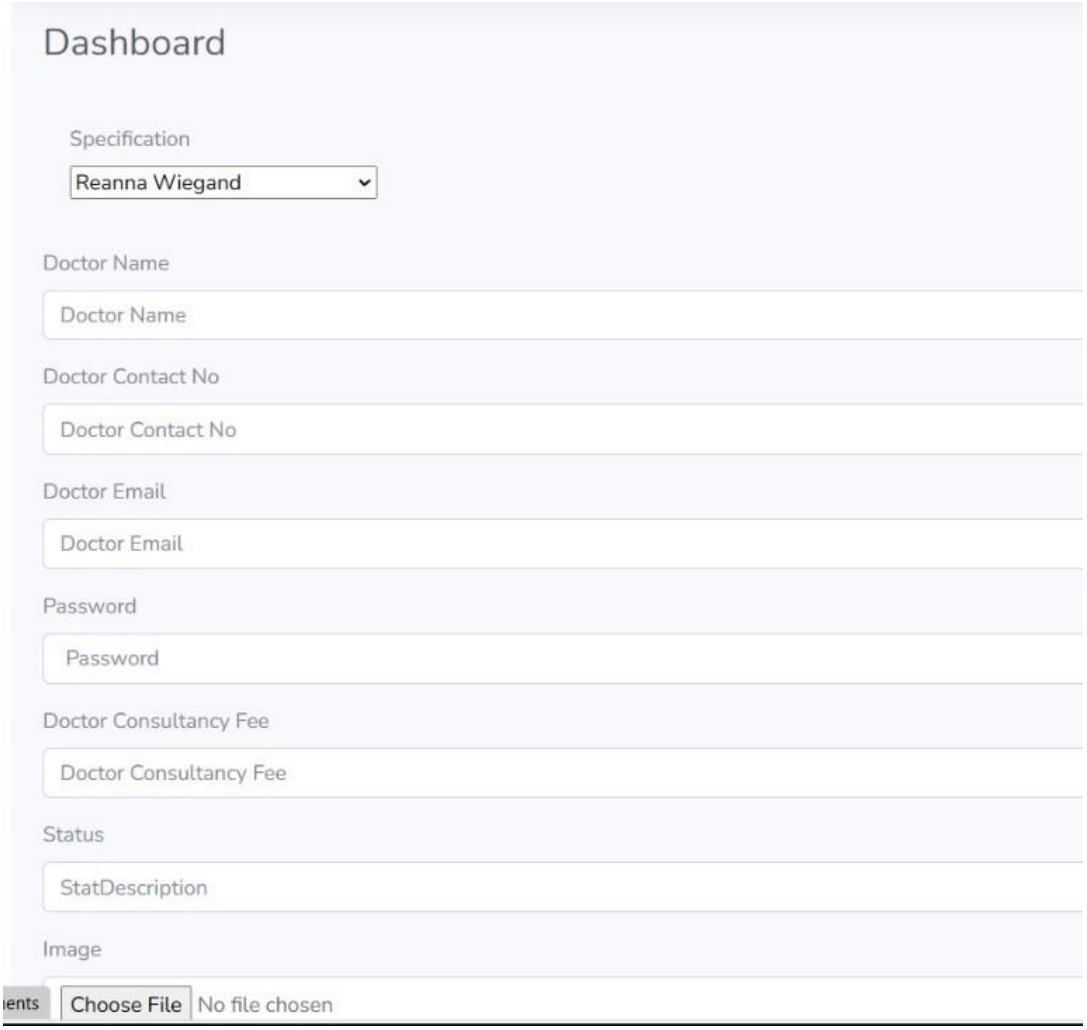
XIII. Fig: System Setting

10.5. USER SPECIFICATION TABLES:



XIV. Fig: User Specification Tables

10.6. DOCTOR SPECIFICATION INDEX:



XV. Fig: Doctor Specification Index

11. DATABASE SQL TABLES:

11.1. USER LOGIN TABLES:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop More
2	name	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
3	email	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
4	email_verified_at	timestamp			Yes	NULL			Change Drop More
5	password	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
6	mobile	int(11)			Yes	NULL			Change Drop More
7	image	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
8	remember_token	varchar(100)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
9	created_at	timestamp			Yes	NULL			Change Drop More
10	updated_at	timestamp			Yes	NULL			Change Drop More
11	age	int(11)			Yes	NULL			Change Drop More
12	address	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
13	role	varchar(255)	utf8mb4_unicode_ci		No	user			Change Drop More

XVI. User Login Tables

11.2. SYSTEM SETTING TABLES:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop More
2	specification_id	bigint(20)		UNSIGNED	No	None			Change Drop More
3	name	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
4	phone	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
5	email	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
6	image	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
7	fee	int(11)			No	None			Change Drop More
8	password	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
9	status	tinyint(4)			No	None			Change Drop More
10	description	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
11	created_at	timestamp			Yes	NULL			Change Drop More
12	updated_at	timestamp			Yes	NULL			Change Drop More

XVII. System Setting Tables.

11.3. USER MAIL TABLES:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop More
2	doctor_id	bigint(20)		UNSIGNED	No	None			Change Drop More
3	user_id	bigint(20)		UNSIGNED	No	None			Change Drop More
4	Appointment_no	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
5	created_at	timestamp			Yes	NULL			Change Drop More
6	updated_at	timestamp			Yes	NULL			Change Drop More

XVIII. User Mail Tables

11.4. DOCTOR TABLES:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop More
2	specification_id	bigint(20)		UNSIGNED	No	None			Change Drop More
3	name	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
4	phone	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
5	email	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
6	image	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
7	fee	int(11)			No	None			Change Drop More
8	password	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
9	status	tinyint(4)			No	None			Change Drop More
10	description	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
11	created_at	timestamp			Yes	NULL			Change Drop More
12	updated_at	timestamp			Yes	NULL			Change Drop More

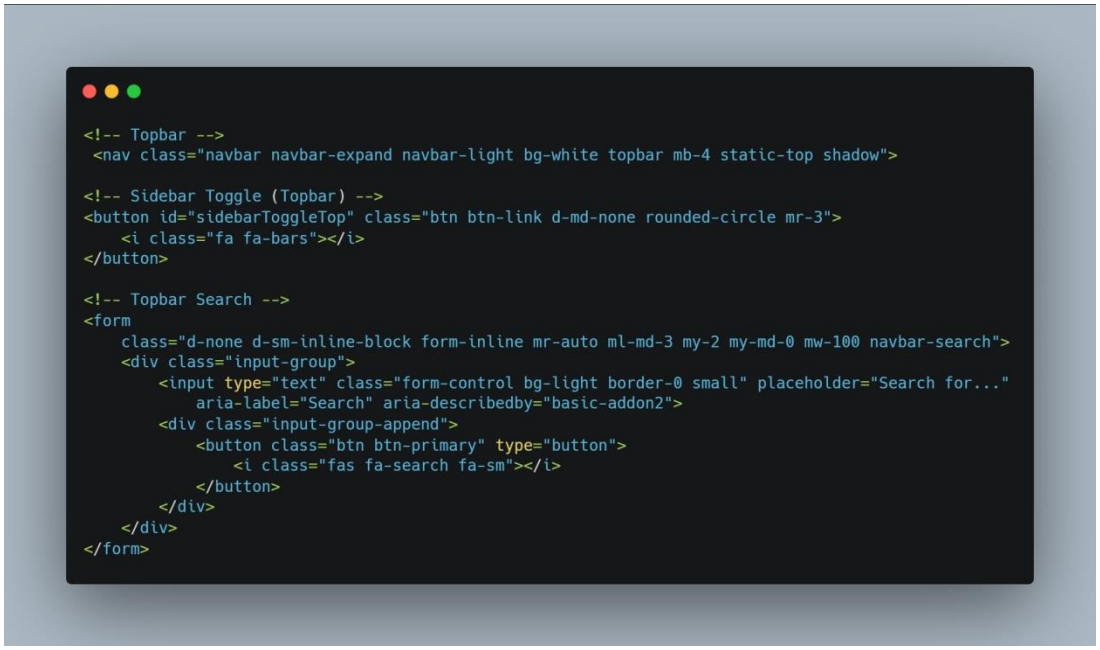
XIX. Doctor Tables.

12. CODE:

12.1. Footer:

```
<!-- Footer -->
  <footer class="sticky-footer bg-white">
              <div class="container my-auto">
                  <div class="copyright text-center my-auto">
                      <span>Copyright &copy; Your Website 2021</span>
                  </div>
              </div>
          </footer>
          <!-- End of Footer -->
```

12.2. Header:

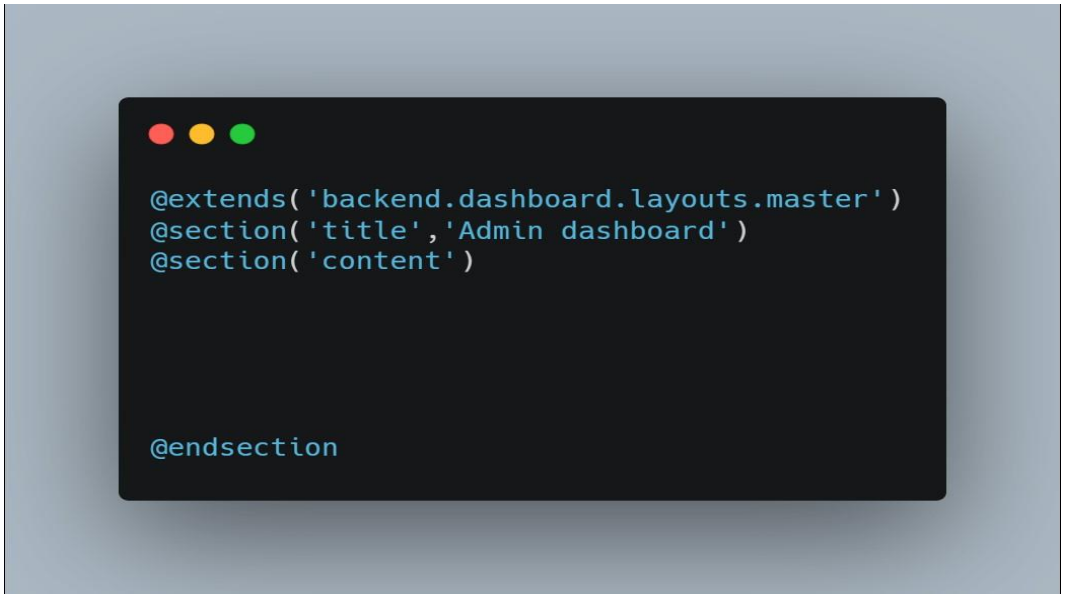


```
<!-- Topbar -->
 <nav class="navbar navbar-expand navbar-light bg-white topbar mb-4 static-top shadow">

<!-- Sidebar Toggle (Topbar) -->
<button id="sidebarToggleTop" class="btn btn-link d-md-none rounded-circle mr-3">
    <i class="fa fa-bars"></i>
</button>

<!-- Topbar Search -->
<form
    class="d-none d-sm-inline-block form-inline mr-auto ml-md-3 my-2 my-md-0 mw-100 navbar-search">
    <div class="input-group">
        <input type="text" class="form-control bg-light border-0 small" placeholder="Search for..."
            aria-label="Search" aria-describedby="basic-addon2">
        <div class="input-group-append">
            <button class="btn btn-primary" type="button">
                <i class="fas fa-search fa-sm"></i>
            </button>
        </div>
    </div>
</form>
```

12.3. Admin Dashboard:



```
@extends('backend.dashboard.layouts.master')
@section('title','Admin dashboard')
@section('content')




@endsection
```

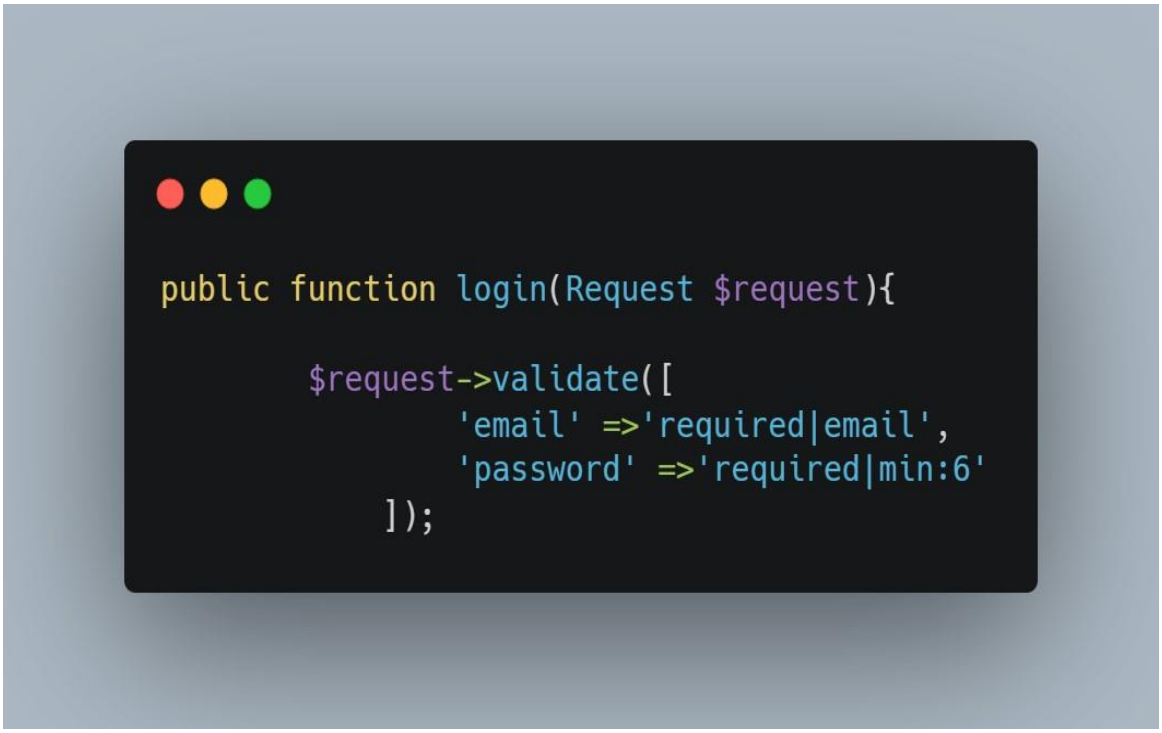
12.4. BOOKAPPOINMENT:

```
public  function storeAppointments( Request  $request)
    {
        $request->validate([
            'date' => 'required',
            'time' =>'required',
            'message' =>'required',

        ]);
        //add validations
        $data = [
           'specification_id' => $request->specification,
            'user_id' => $request->userid,
            'doctor_id' => $request->doctor_id,
            'date' => $request->date,
            'time' => $request->time,
            'message' => $request->message
        ];

        Appointment::insert($data);
       Mail::to(auth()->user()->email)->send(new AppointmentMail('Pending'));
        //redirect from here
    }
```

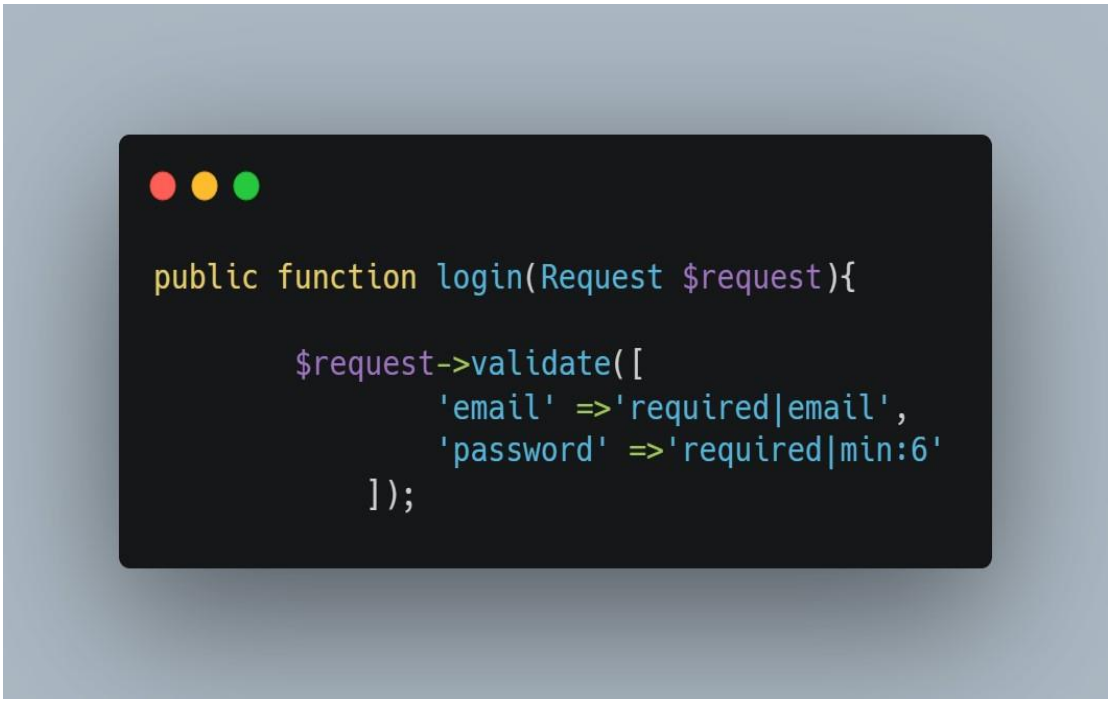
12.5. LOGIN CODE



```
public function login(Request $request){

        $request->validate([
                'email' =>'required|email',
                'password' =>'required|min:6'
            ]);
```

12.6. LOGOUT CODE:



```
public function login(Request $request){

        $request->validate([
                'email' =>'required|email',
                'password' =>'required|min:6'
            ]);
```

12.7. REGISTER CODE:

```
public function usersignup(Request $request){
        $request->validate([
            'name' => 'required',
            'email' =>'required|email',
            'password' =>'required|min:6',
            'mobile' => 'required',
            'address' => 'required',

        ]);

        $image_url = '';
        if($request->has('image') && $request->file('image')){
            $file  = $request->file('image');
            $name  = time().'-'. rand(10,9999999999999).'-'.$file->getClientOriginalName();
            $path = public_path().'/users'.'/';
            $file->move($path,$name);
            $image_url = asset('/users').'/'.$name;
        }

        $data=[
            'name'=>$request->name,
           'email'=>$request->email,
           'password'=>bcrypt($request->password),
           'mobile'=>$request->mobile,
           'image'=>$image_url,
           'age'=>$request->age,
           'address'=>$request->address,


        ];

        User::insert($data);
        return redirect()->back();
    }
```

13. CONCLUSION:

The healthcare and telemedicine project successfully leveraged technology to improve accessibility and delivery of medical services. Through virtual consultations, it addressed geographical barriers, ensuring timely healthcare access. User feedback highlighted increased convenience and reduced healthcare disparities. Ongoing advancements in telemedicine demonstrate its potential for shaping the future of healthcare delivery.

While telemedicine has demonstrated benefits, its widespread adoption requires addressing regulatory, technological, and privacy challenges. Striking a balance between in-person care and virtual solutions is crucial for optimizing healthcare delivery and improving overall patient outcomes.

14. REFERENCE:

- www.google.com
- www.javapoint.com
- www.w3school.com
- www.wikipedia.com
- www.youtube.com