

# INTEGRATED PROJECT REPORT

On

**HEALTHPLUS**

Submitted in partial fulfilment of the requirement for the  
Course Integrated Project (CS 203) of

**COMPUTER SCIENCE AND ENGINEERING**  
**B.E. Batch-2020**



**JUNE-2023**

**Under the Guidance of:**

Dr. Shilpi Harnal  
Assistant Professor

**Submitted By:**

Name	Roll Number
Liza	2010991608
Luckshay	2010991610
Puneet Jain	2010991674
Reeya Bansal	2010991682

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**  
**CHITKARA UNIVERSITY**  
**PUNJAB**

## **CERTIFICATE**

This is to be certified that the project entitled “HealthPlus” has been submitted for the Bachelor of Computer Science Engineering at Chitkara University, Punjab during the academic semester January, 2023 – May, 2023 is a Bonafede piece of project work carried out by “Luckshay (2010991610), Liza (2010991608), Puneet Jain (2010991674), Reeya Bansal (2010991682)” towards the partial fulfilment for the award of the course Integrated Project (CS 203) under the guidance of “and supervision.

**Sign. of Project Guide:**

**Dr. Shilpi Harnal**

Assistant Professor,  
Department of Computer Science and Engineering

**CANDIDATE'S DECLARATION**

We, “**Luckshay (2010991610), Liza (2010991608), Puneet Jain (2010991674), Reeya Bansal (2010991682)**”, B.E.-2020 of the Chitkara University, Punjab hereby declare that the Integrated Project Report entitled “**HealthPlus**” is an original work and data provided in the study is authentic to the best of our knowledge. This report has not been submitted to any other Institute for the award of any other course.

**Sign. of Student 1**

Liza

ID No.: 2010991608

**Sign. of Student 2**

Luckshay

ID No.: 2010991610

**Sign. of Student 3**

Puneet Jain

ID No.: 2010991674

**Sign. of Student 4**

Reeya Bansal

ID No.: 2010991682

**Place:** Chitkara University, Rajpura

**Date:** 8<sup>th</sup> June, 2023

## **ACKNOWLEDGEMENT**

It is our pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced my thinking, behavior and acts during the course of study.

We express our sincere gratitude to all for providing me an opportunity to undergo Integrated Project as the part of the curriculum.

We are thankful to “**Dr. Shilpi Harnal**” for her support, cooperation, and motivation provided to us during the training for constant inspiration, presence and blessings.

We also extend our sincere appreciation to “**Dr. Shilpi Harnal**” who provided her valuable suggestions and precious time in accomplishing our Integrated Project Report.

Lastly, we would like to thank the almighty and our parents for their moral support and friends with whom we shared our day-to day experience and received lots of suggestions that improve our quality of work.

<b>Liza</b>	<b>Luckshay</b>	<b>Puneet Jain</b>	<b>Reeya Bansal</b>
<b>ID No.: 2010991608</b>	<b>ID No.: 2010991610</b>	<b>ID No.: 2010991674</b>	<b>ID No.: 2010991682</b>

## INDEX

<b>1. ABSTRACT/KEYWORDS .....</b>	<b>6</b>
<b>2. INTRODUCTION TO THE PROJECT .....</b>	<b>7</b>
2.1 Background .....	7
2.2 Problem Statement .....	7
<b>3. SOFTWARE AND HARDWARE REQUIREMENTS .....</b>	<b>9</b>
3.1 Methods .....	9
3.2 Programming/Working Environment .....	11
3.3 Requirements to run the application .....	11
<b>4. GUI ANALYSIS .....</b>	<b>12</b>
<b>5. CODE IMPLEMENTATION .....</b>	<b>27</b>
<b>6. SYSTEM TESTING.....</b>	<b>33</b>
<b>7. LIMITATIONS .....</b>	<b>34</b>
<b>8. CONCLUSION.....</b>	<b>35</b>
<b>9. FUTURE SCOPE .....</b>	<b>36</b>
<b>10. BIBLIOGRAPHY / REFERENCES .....</b>	<b>37</b>

## **ABSTRACT/KEYWORDS**

### **1.1 Abstract**

The HealthPlus application is a MERN stack project developed to streamline healthcare management.

It offers features such as appointment scheduling, digital medical record management, and efficient communication between patients and healthcare providers. It has a number of features that make it stand out from other healthcare management websites and apps including:

- **Seamless Appointment Scheduling:** Patients can easily book and manage appointments with healthcare providers, ensuring a smooth and convenient healthcare experience.
- **Digital Medical Record Management:** The app allows users to maintain their medical records digitally, eliminating the need for physical paperwork and enabling secure access to health information.
- **Effective Communication:** The application facilitates seamless communication between patients and healthcare providers through messaging and notification systems, improving the overall healthcare experience.

### **1.2 Keywords**

- HealthCare Management System
- Patient-Doctor interaction
- Medical record management
- Appointment scheduling
- Quality healthcare services.

## **INTRODUCTION TO THE PROJECT**

### **2.1 Background**

The HealthPlus project was initiated to address the growing need for a streamlined healthcare management solution. With the increasing demand for efficient appointment scheduling, digital medical record management, and improved patient-provider communication, there was a need for a comprehensive application that could integrate these functionalities.

### **2.2 Problem Statement**

The lack of an efficient and integrated healthcare management system leads to challenges in appointment scheduling, medical record management, and communication between healthcare providers and patients.

### **2.3 Project**

The objective of HealthPlus project is to create an advanced healthcare management application leveraging the MERN stack. By integrating modern technologies, the application aims to enhance healthcare services, streamline processes, and facilitate effective communication between patients and healthcare providers, ultimately improving the overall healthcare experience.

HealthPlus app provides a user-friendly interface for patients to easily book and manage their appointments with healthcare providers, streamlining the scheduling process and reducing administrative burdens.

With HealthPlus, users can maintain their medical records digitally, eliminating the need for physical paperwork and enabling secure access to their health information anytime, anywhere.

HealthPlus prioritizes the privacy and security of sensitive healthcare data. Robust measures are implemented to ensure data encryption, access control, and compliance with industry regulations, providing users with peace of mind.



# SOFTWARE AND HARDWARE REQUIREMENTS

## 3.1 Methods

### 3.1.1 Modern UI/UX Design:

- The project incorporates modern UI/UX design principles to create an intuitive and visually appealing user interface.
- The use of React.js enables efficient state management and dynamic rendering of data, enhancing the user experience.
- CSS frameworks like Bootstrap or Material-UI may be utilized to enhance the design aesthetics and ensure responsiveness across different devices.

### 3.1.2 Data Storage and Retrieval:

- MongoDB, a NoSQL database, is utilized to store and manage healthcare-related data, including user inputs, medical records, and other relevant information.
- The Mongoose library may be used to facilitate object modeling and interaction with the MongoDB database.
- Data retrieval techniques such as querying and aggregation are implemented to fetch and process the required healthcare information.

### 3.1.3 Appointment Scheduling and Management:

- The project includes a robust appointment scheduling system that allows patients to easily book and manage their appointments with healthcare providers.
- The system handles appointment availability, conflict resolution, and reminders to ensure efficient scheduling and management.

### 3.1.4 Digital Medical Record Management:

- HealthPlus offers a secure and centralized platform for users to maintain and access their digital medical records.
- Users can securely store and manage their health information, eliminating the need for physical paperwork and enabling seamless access to medical records.

### **3.1.5 Communication and Messaging:**

- The application facilitates seamless communication between patients and healthcare providers through messaging and notification systems.
- Patients can reach out to their doctors, ask questions, and receive timely updates regarding their healthcare.
- Healthcare providers can efficiently manage patient inquiries, respond promptly, and provide necessary guidance.

### **3.1.6 User Authentication and Authorization:**

- The project implements user authentication mechanisms to secure access to the application and protect user data.
- Techniques such as password hashing and salting are employed to ensure the confidentiality and integrity of user credentials.
- Authentication libraries or frameworks such as Passport.js or JWT may be utilized for streamlined user authentication and authorization processes.

### **3.1.7 Integration of Third-Party APIs:**

- The project may integrate third-party APIs to enhance functionality, such as integrating maps to show location of the facilities, payment gateways for seamless online transactions and SMS gateways for appointment reminders and notifications.

### **3.1.8 Scalability and Performance Optimization:**

- The application is designed to handle a large volume of users and data, with optimizations in place to ensure scalability and performance.
- Techniques like caching, load balancing, and database indexing may be implemented to optimize the application's performance and responsiveness.

### **3.1.9 Continuous Testing and Improvement:**

- The project follows a rigorous testing process to identify and resolve bugs, security vulnerabilities, and usability issues.
- Continuous improvements are made based on user feedback and emerging technologies to enhance the overall functionality and user experience of the application.

### **3.1.10 Deployment and Maintenance:**

- The application is deployed to a reliable hosting environment namely, ensuring availability and stability.
- Regular maintenance activities, including security updates and bug fixes, are performed to keep the application running smoothly and securely.

## **3.2 Programming/Working Environment**

HealthPlus is programmed on MERN Stack based on HTML/CSS, JS, React, Express, MongoDB, NodeJS integrated with Python.

## **3.3 Requirements to run the application**

To run HealthPlus, you will need a web browser and an internet connection.

**The software requirements for HealthPlus are as follows:**

- A web browser
- An internet connection

**The hardware requirements for HealthPlus are as follows:**

- A computer with a minimum of 1GB of RAM
- A hard drive with a minimum of 10GB of free space

## GUI ANALYSIS

Observe the following screenshots to understand the User Experience Flow

### 4.1 Home Page

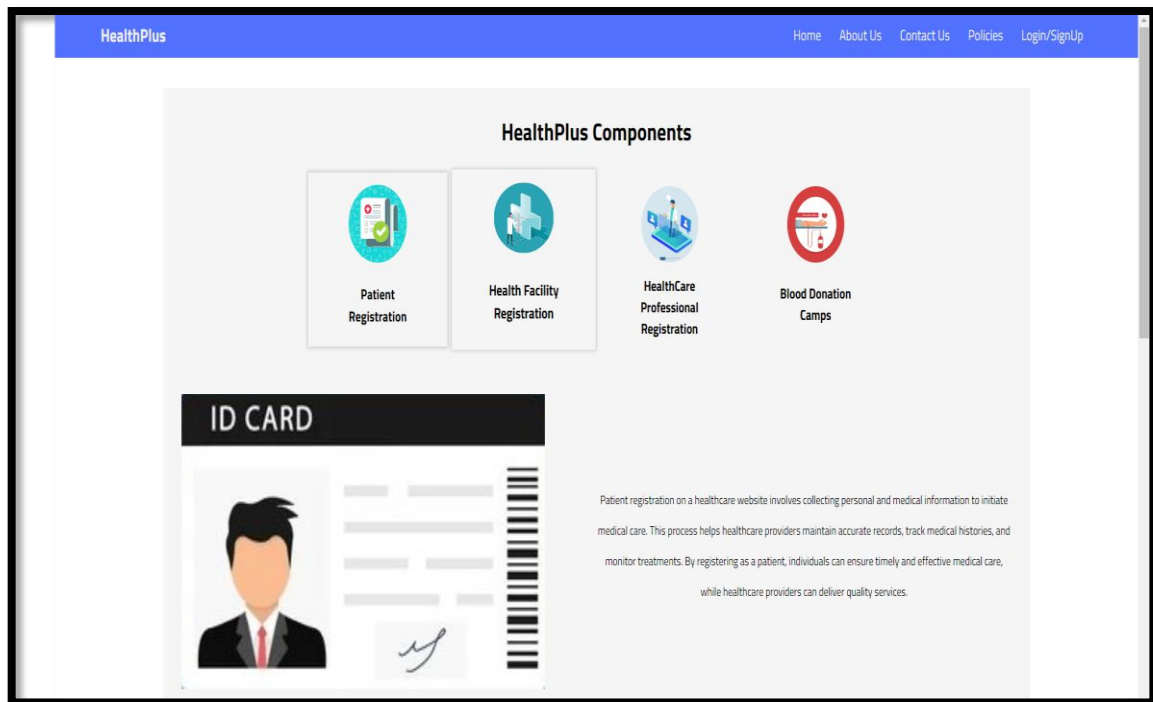


Figure 4.1.1

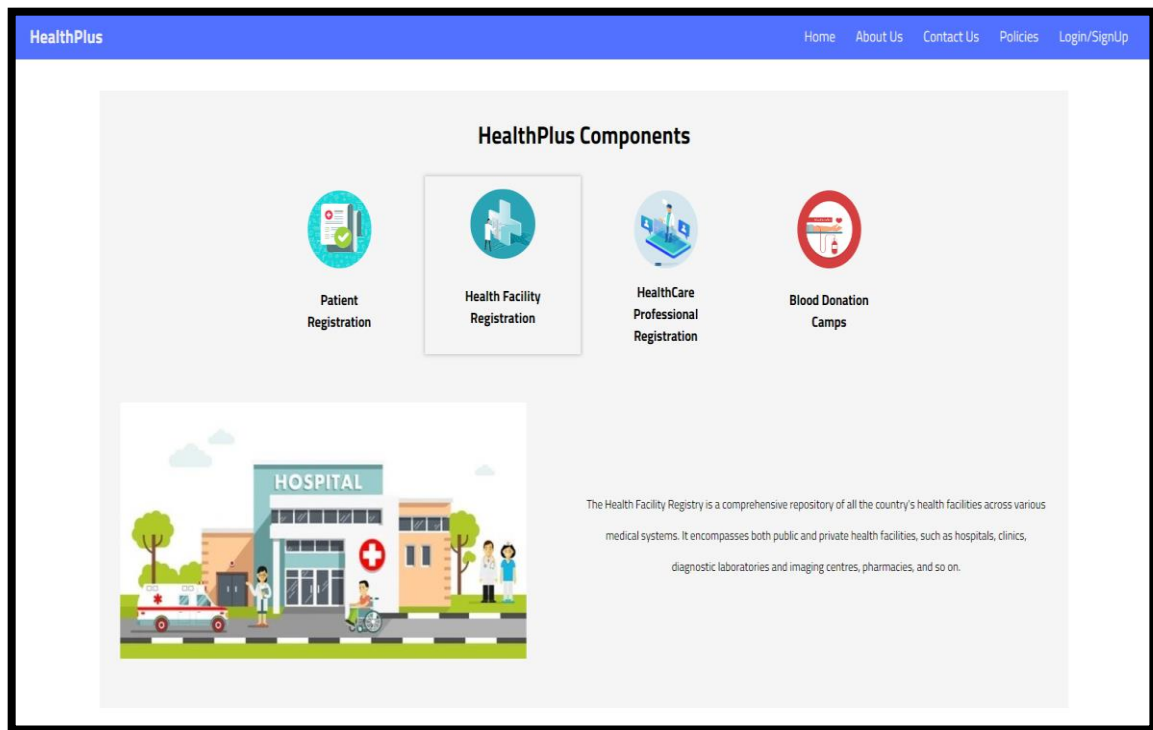


Figure 4.1.2

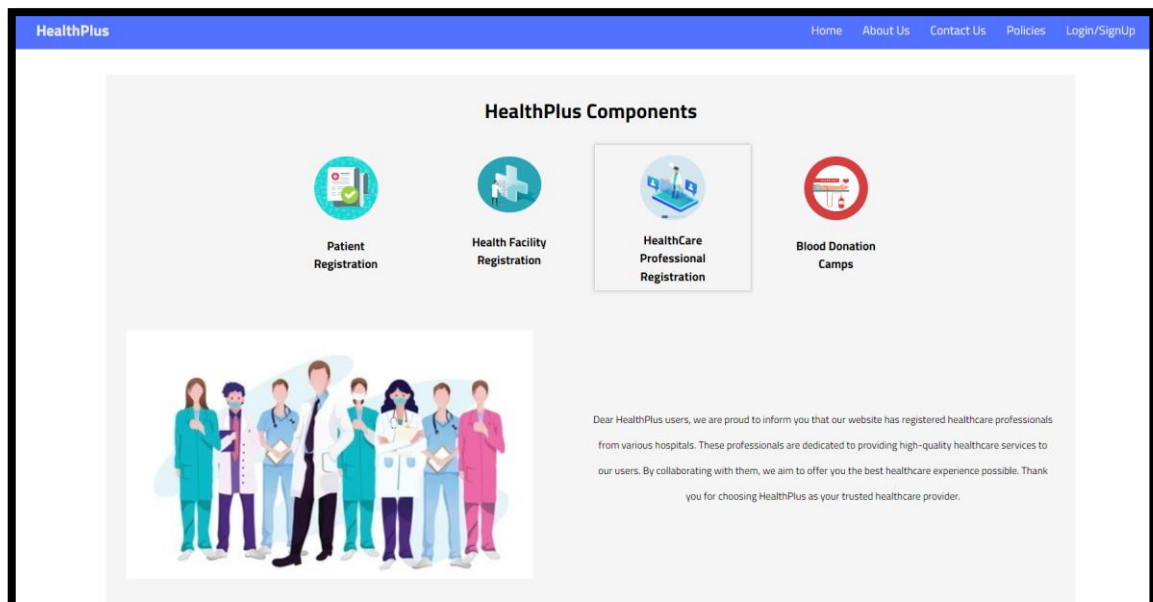


Figure 4.1.3

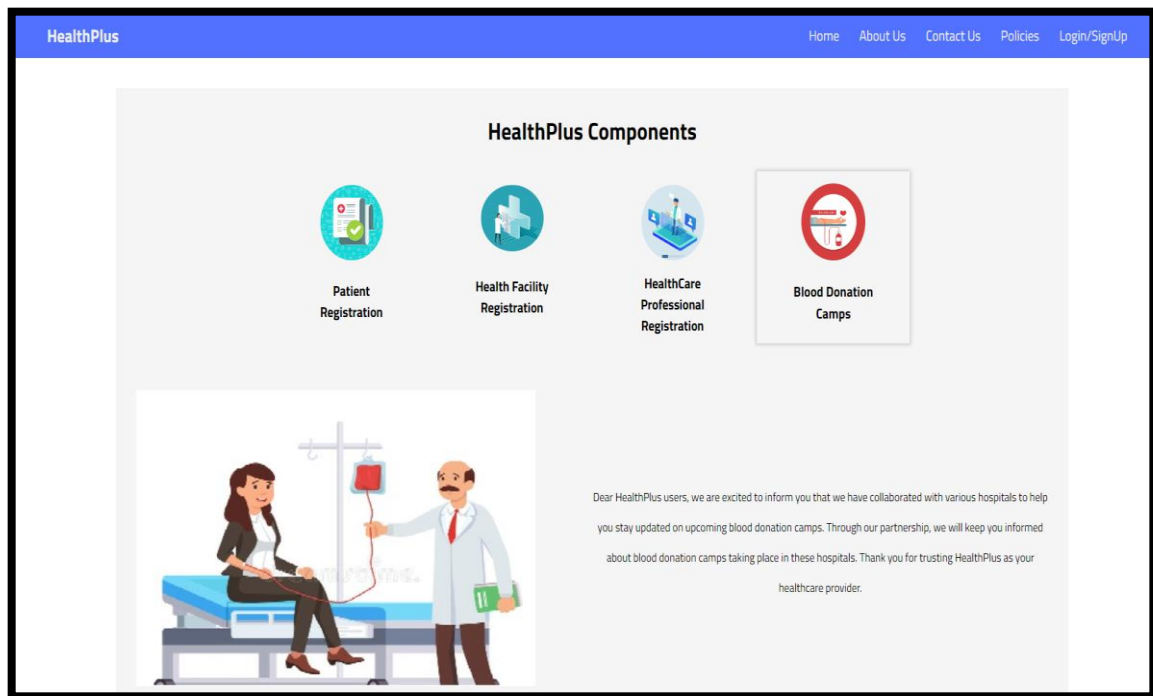


Figure 4.1.4

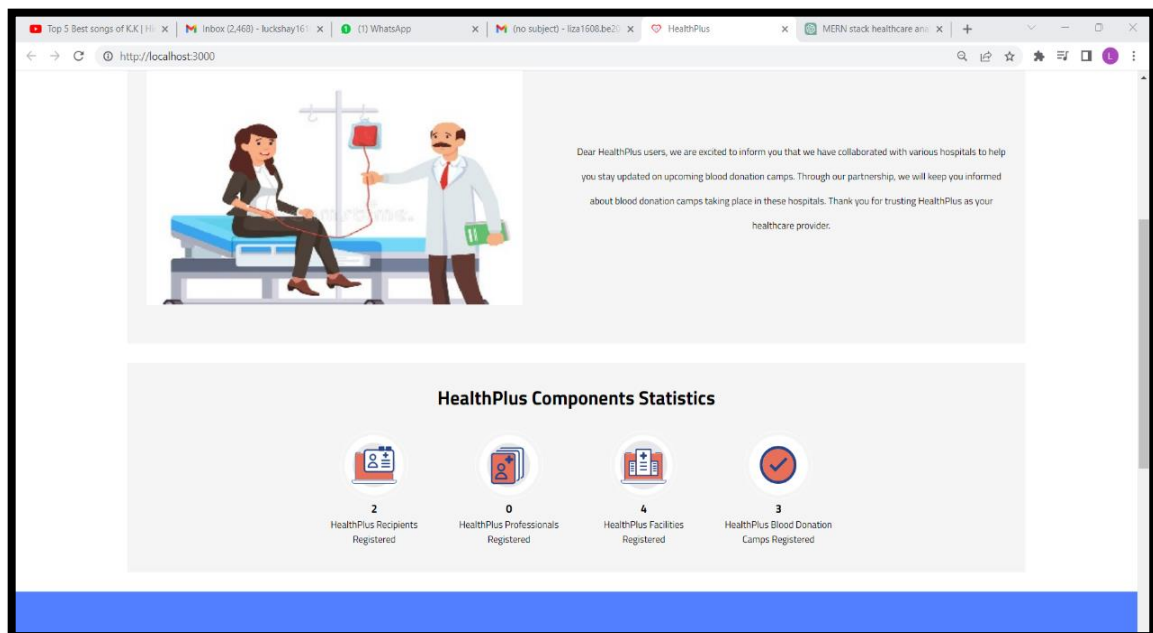


Figure 4.1.5

## 4.2 About Us Component

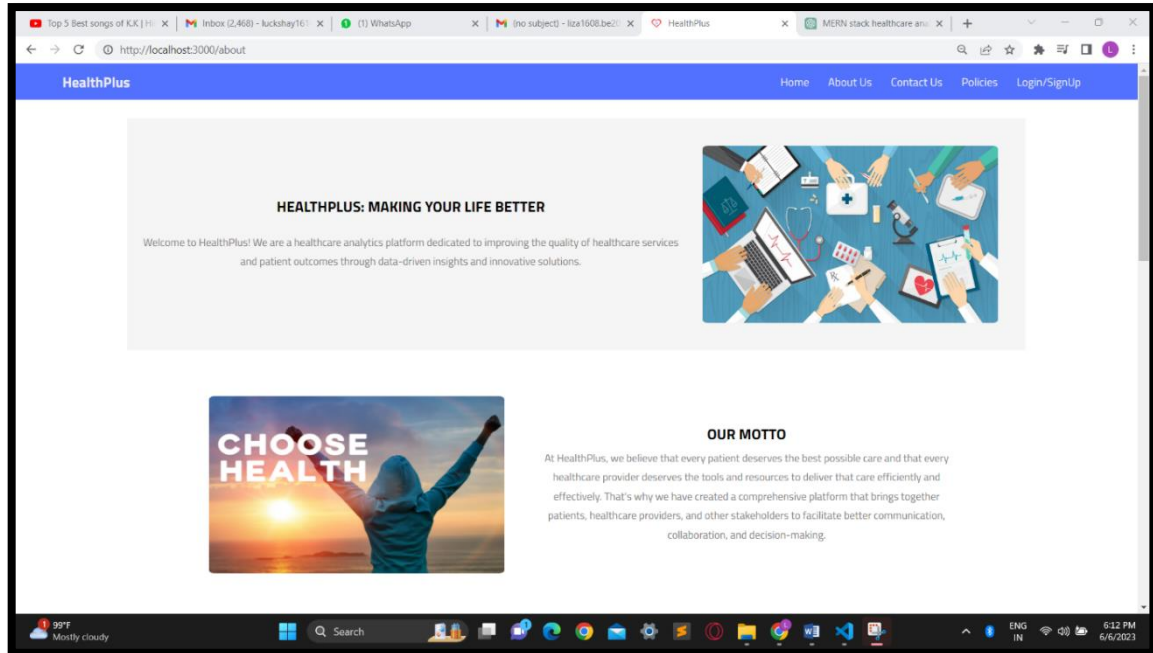


Figure 4.2.1

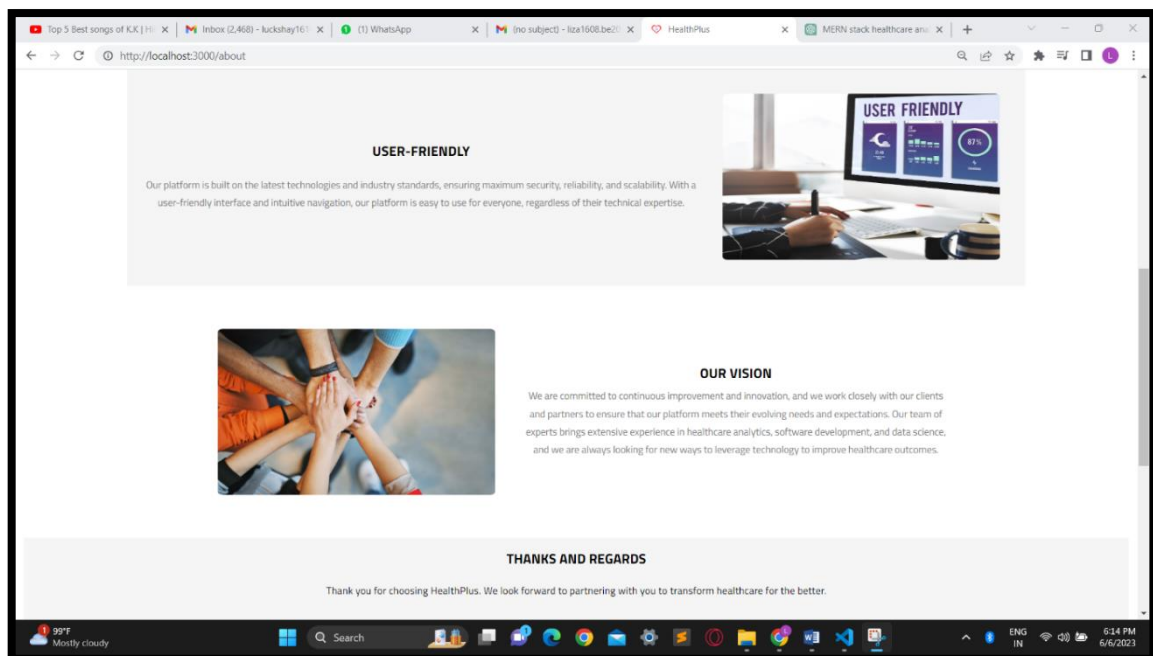


Figure 4.2.2



## 4.3 Contact Us

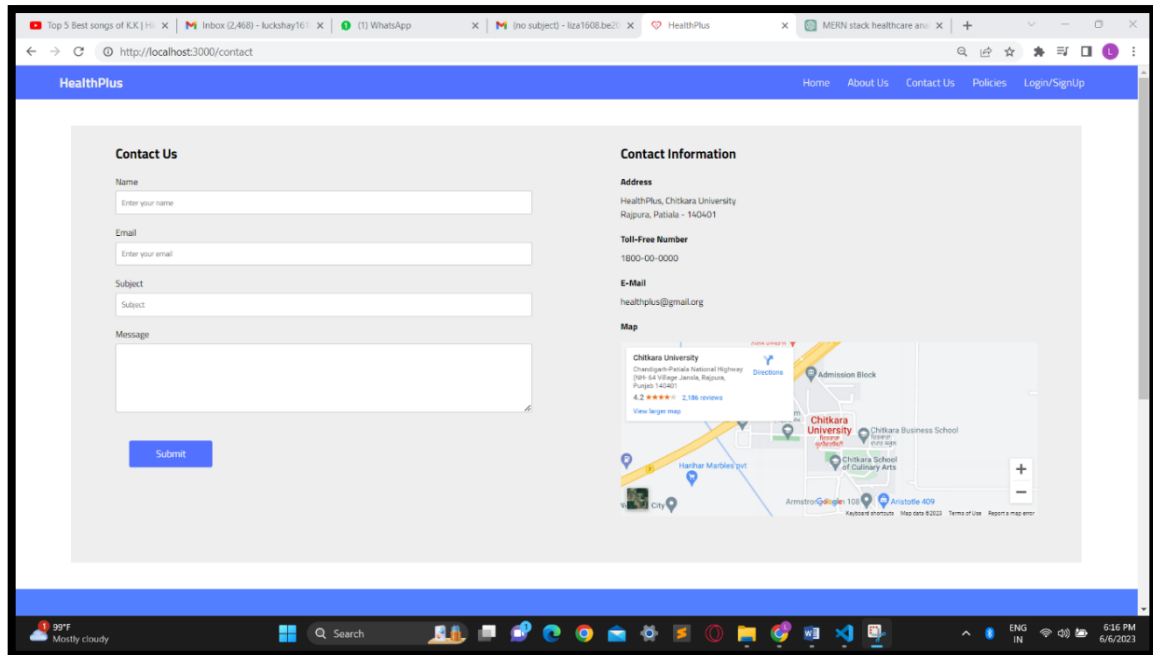


Figure 4.3.1





## 4.4 Policies

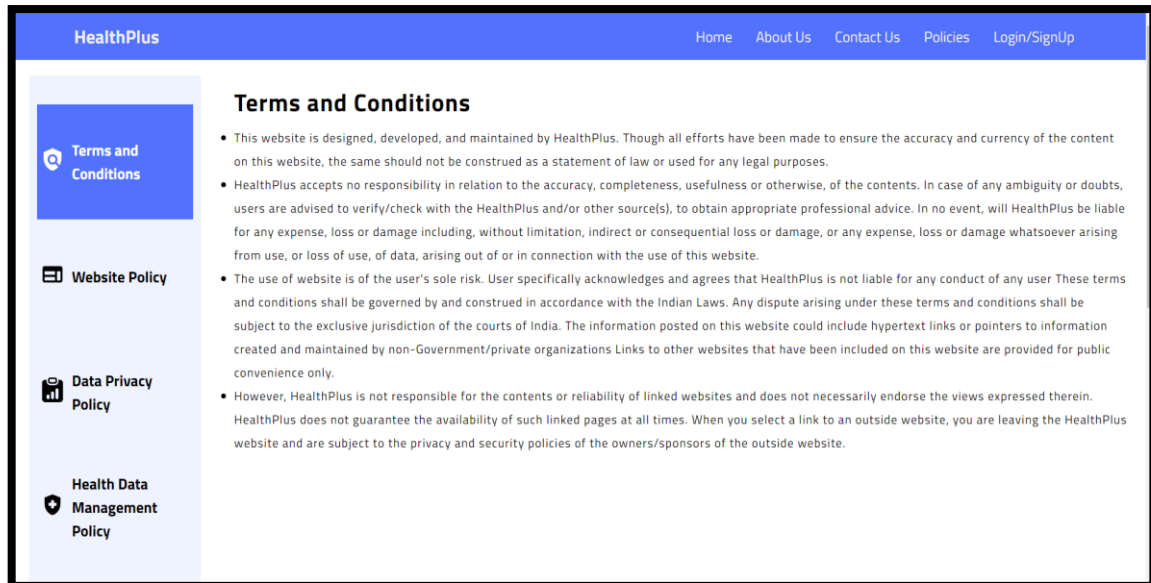


Figure 4.4.1

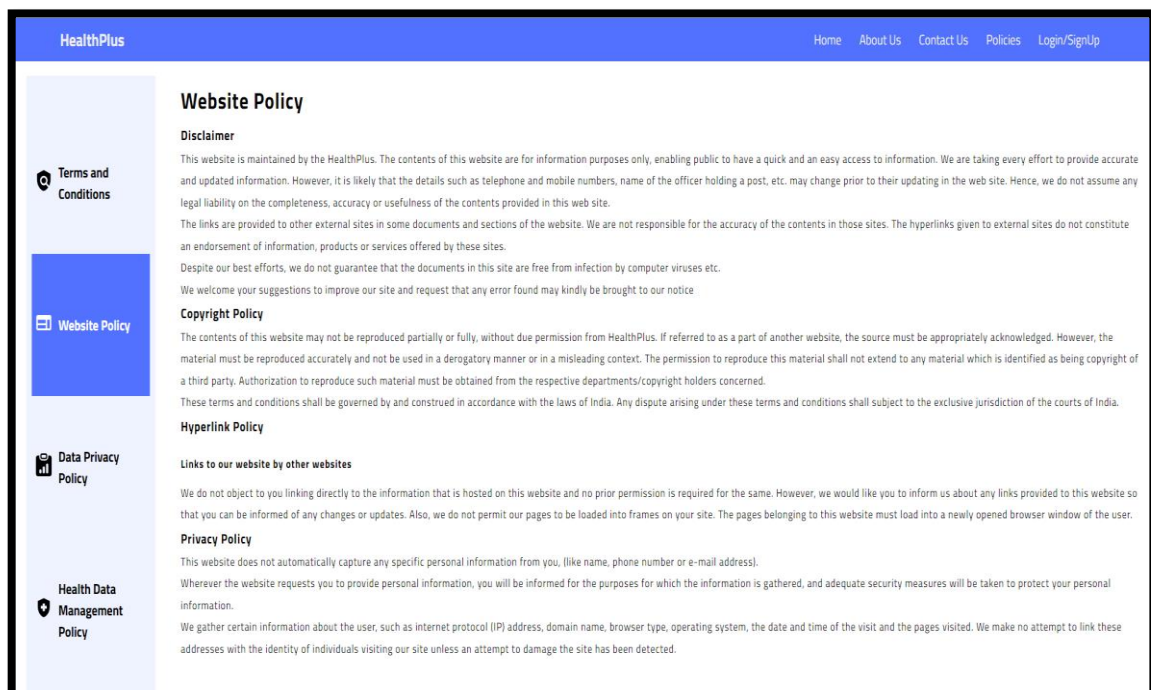


Figure 4.4.2





## 4.5 Footer



Figure 4.5.1



## 4.6 Signup

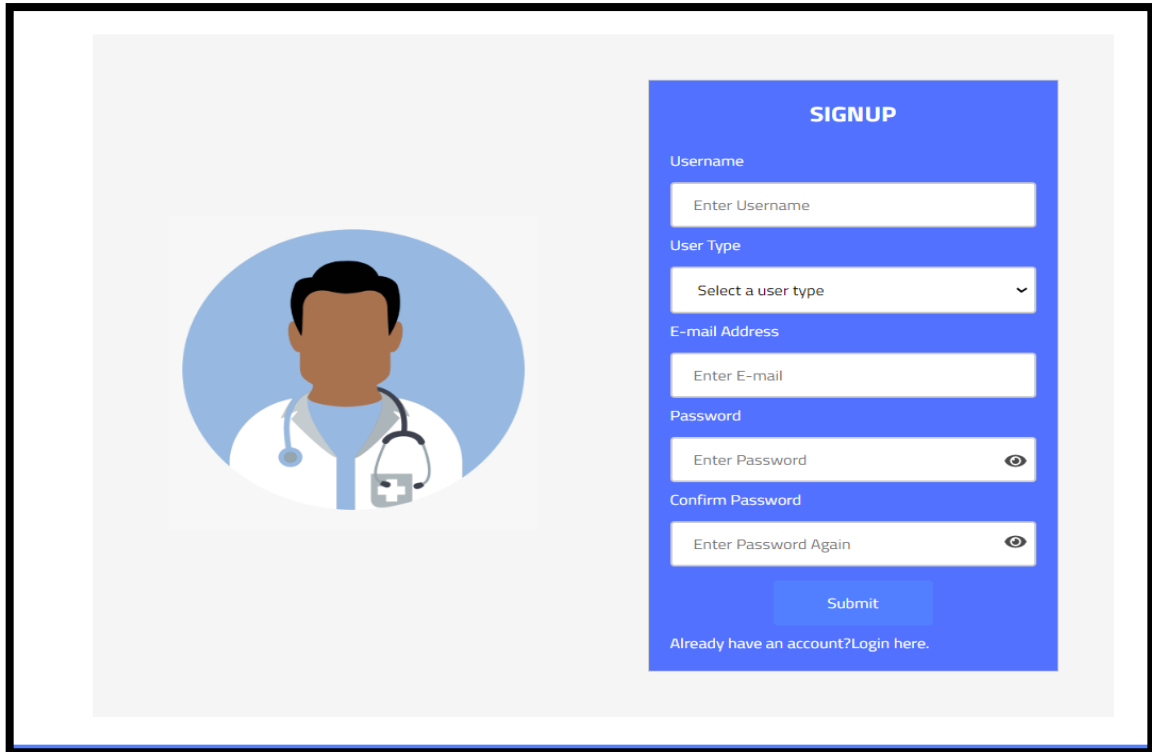
A screenshot of a web application's signup page. On the left, there is a circular profile picture placeholder showing a doctor in a white coat with a stethoscope. On the right, there is a blue rectangular form titled 'SIGNUP'. The form contains the following fields: 'Username' with a text input field labeled 'Enter Username'; 'User Type' with a dropdown menu labeled 'Select a user type'; 'E-mail Address' with a text input field labeled 'Enter E-mail'; 'Password' with a text input field labeled 'Enter Password' and an eye icon for toggling visibility; and 'Confirm Password' with a text input field labeled 'Enter Password Again' and an eye icon. Below these fields is a blue 'Submit' button. At the bottom of the form, there is a link that says 'Already have an account? Login here.'

Figure 4.6.1



## 4.7 Login

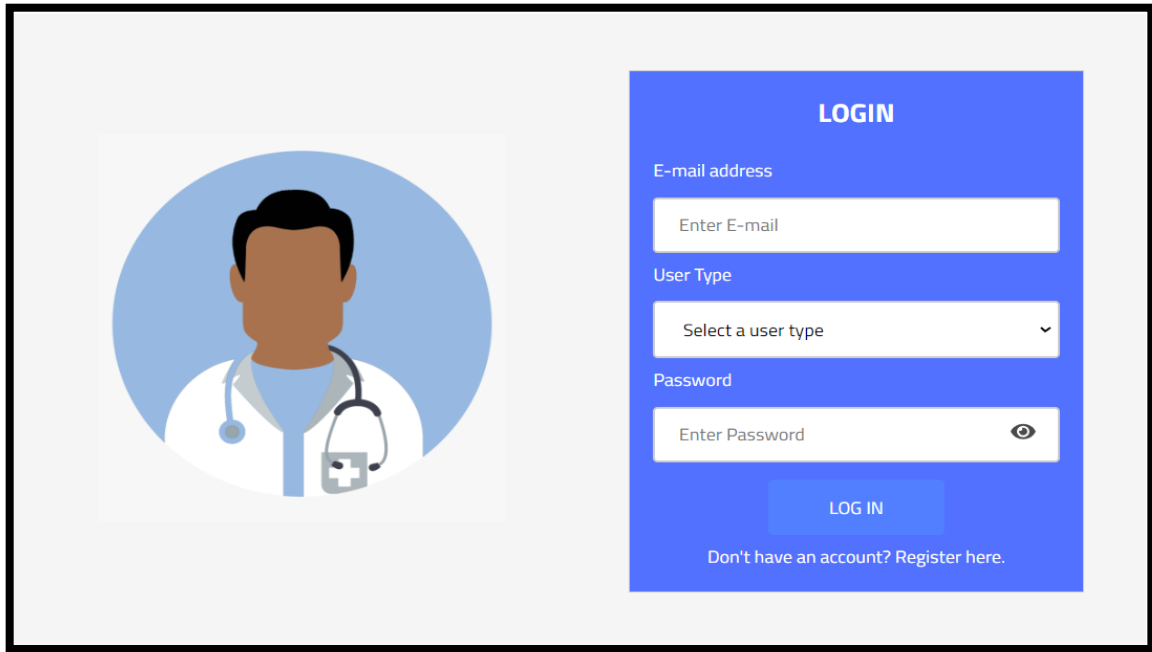
A login form interface with a blue background. On the left, there is a circular profile picture of a doctor. On the right, the form is titled 'LOGIN' and contains three input fields: 'E-mail address' with a placeholder 'Enter E-mail', 'User Type' with a dropdown menu showing 'Select a user type', and 'Password' with a placeholder 'Enter Password' and a toggle icon. Below the fields is a blue 'LOG IN' button. At the bottom, there is a link that says 'Don't have an account? Register here.'

Figure 4.7.1



## 4.8 Recipient Dashboard

### 4.8.1 Home

**Your Appointments**

Hospital	Doctor	Date	Time	Specialty
Apollo Hospital	Dr. Ram Gupta	June 10, 2023	10:00 AM	Cardiology
Neelam Hospital	Dr. Shyam Bhatia	June 11, 2023	11:00 AM	Pediatrics
Max Hospital	Dr. Harneet Narula	June 14, 2023	1:00 PM	Orthopedics

**Active Blood Donation Camps**

Hospital	Location	Date	Start Time	End Time
City Hospital	Rajpura	June 10, 2023	10:00 AM	5:00 PM
WellSpring Hospital	Zirakpur	June 11, 2023	10:00 AM	5:00 PM
Apex Hospital	Rajpura	June 11, 2023	10:00 AM	5:00 PM

Figure 4.8.1

### 4.8.2 Profile Details

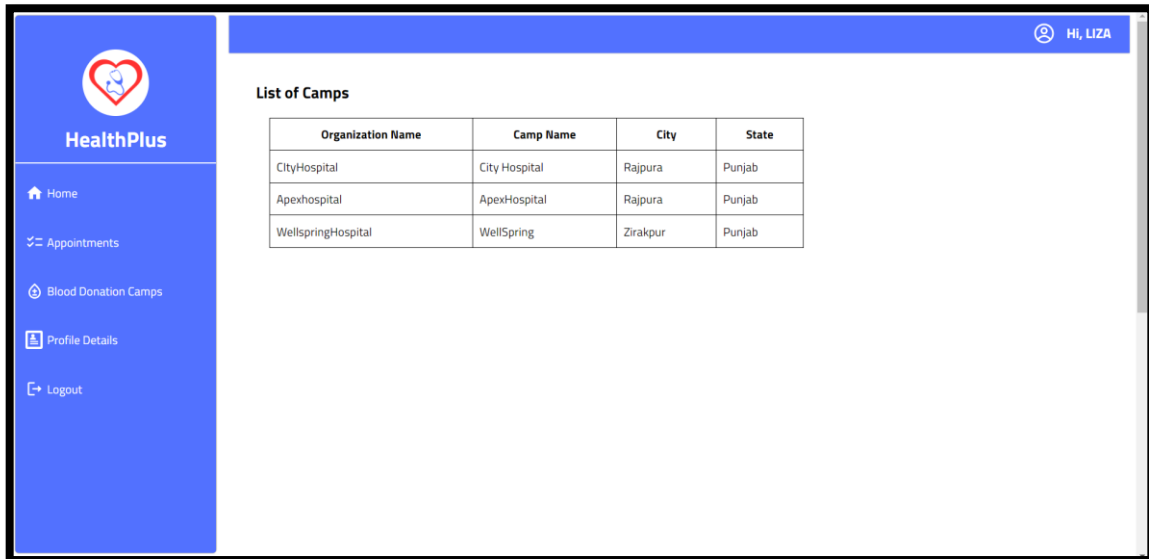
**User Profile**

Name:	LIZA
Email:	liza1608.be20@chitkara.edu.in
Date of Birth:	14-04-2002
Age:	21 years 1 months
Gender:	Female
Contact No:	9728757072
Address:	.. Sirsa, Haryana, 125055, India
Blood Group:	B+

[Update Profile](#)

Figure 4.8.2

### 4.8.3 Active Blood Donation Camps



**HealthPlus**

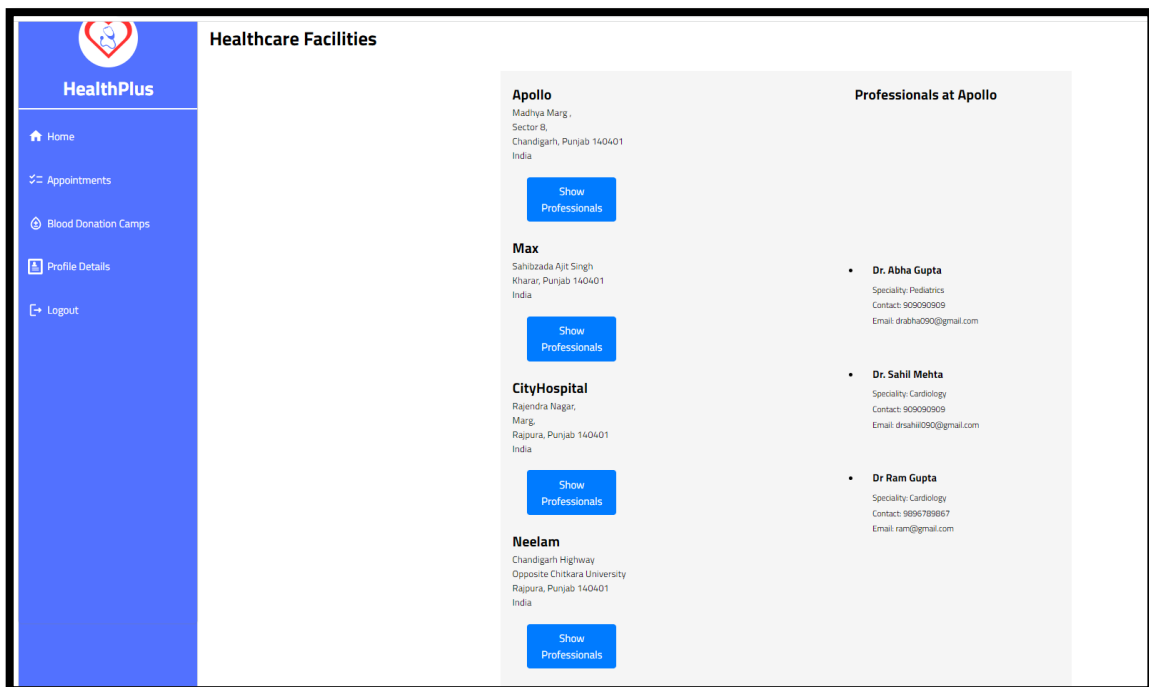
- Home
- Appointments
- Blood Donation Camps
- Profile Details
- Logout

**List of Camps**

Organization Name	Camp Name	City	State
CityHospital	City Hospital	Rajpura	Punjab
Apexhospital	ApexHospital	Rajpura	Punjab
WellspringHospital	WellSpring	Zirakpur	Punjab

Figure 4.8.3

### 4.8.4 Appointments



**HealthPlus**

- Home
- Appointments
- Blood Donation Camps
- Profile Details
- Logout

**Healthcare Facilities**

**Apollo**  
Madhya Marg,  
Sector 8,  
Chandigarh, Punjab 140401  
India

Show Professionals

**Max**  
Sahibzada Ayt Singh  
Kharar, Punjab 140401  
India

Show Professionals

**CityHospital**  
Rajendra Nagar,  
Marg,  
Rajpura, Punjab 140401  
India

Show Professionals

**Neelam**  
Chandigarh Highway  
Opposite Chitkara University  
Rajpura, Punjab 140401  
India

Show Professionals

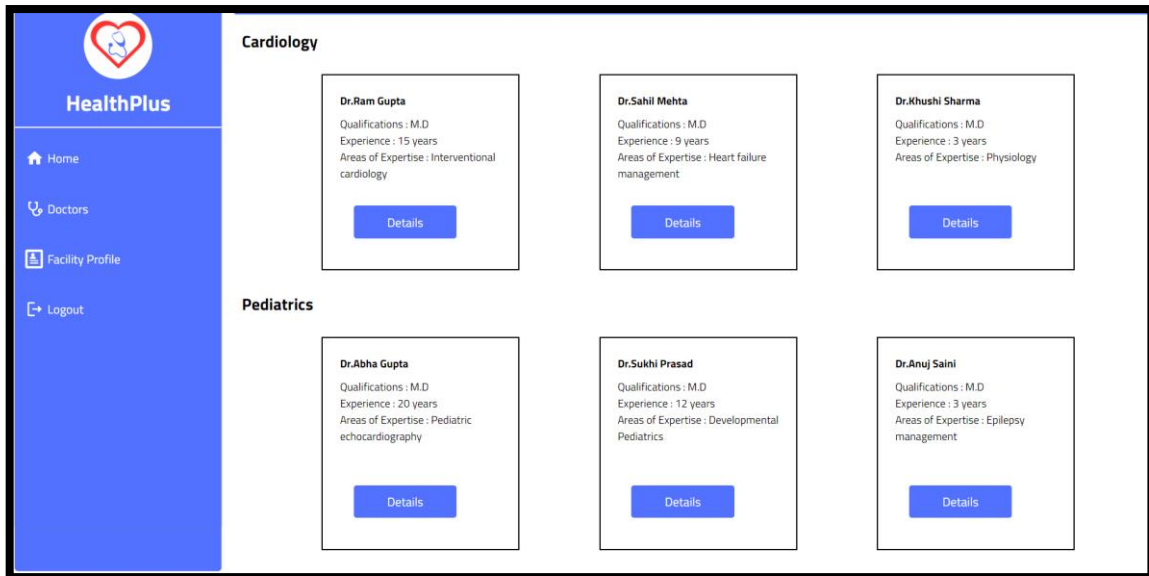
**Professionals at Apollo**

- Dr. Abha Gupta**  
Speciality: Pediatrics  
Contact: 909090909  
Email: drabha90@gmail.com
- Dr. Sahil Mehta**  
Speciality: Cardiology  
Contact: 909090909  
Email: drsahil90@gmail.com
- Dr Ram Gupta**  
Speciality: Cardiology  
Contact: 9896789867  
Email: ram@gmail.com

Figure 4.8.4

## 4.9 HealthCare Facility Dashboard

### 4.9.1 Home



**HealthPlus**

- Home
- Doctors
- Facility Profile
- Logout

**Cardiology**

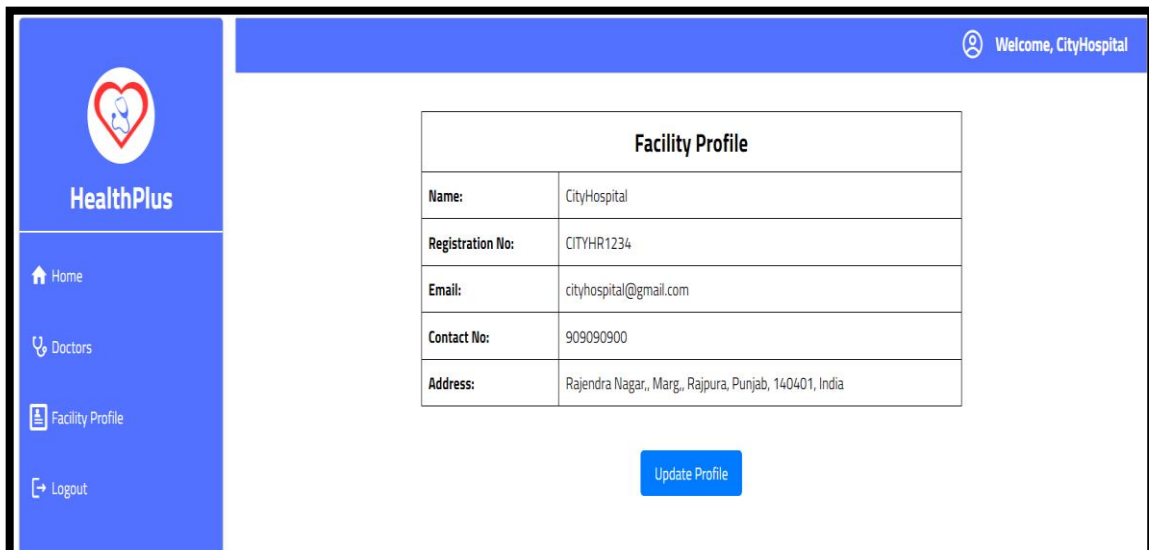
- Dr. Ram Gupta**  
Qualifications : M.D  
Experience : 15 years  
Areas of Expertise : Interventional cardiology  
[Details](#)
- Dr. Sahil Mehta**  
Qualifications : M.D  
Experience : 9 years  
Areas of Expertise : Heart failure management  
[Details](#)
- Dr. Khushi Sharma**  
Qualifications : M.D  
Experience : 3 years  
Areas of Expertise : Physiology  
[Details](#)

**Pediatrics**

- Dr. Abha Gupta**  
Qualifications : M.D  
Experience : 20 years  
Areas of Expertise : Pediatric echocardiography  
[Details](#)
- Dr. Sukhi Prasad**  
Qualifications : M.D  
Experience : 12 years  
Areas of Expertise : Developmental Pediatrics  
[Details](#)
- Dr. Anuj Saini**  
Qualifications : M.D  
Experience : 3 years  
Areas of Expertise : Epilepsy management  
[Details](#)

Figure 4.9.1

### 4.9.2 Facility Profile



**HealthPlus**

- Home
- Doctors
- Facility Profile
- Logout

Welcome, CityHospital

Facility Profile	
Name:	CityHospital
Registration No:	CITYHR1234
Email:	cityhospital@gmail.com
Contact No:	909090900
Address:	Rajendra Nagar,, Marg,, Rajpura, Punjab, 140401, India

[Update Profile](#)

Figure 4.9.2





### 4.9.3 Doctors Component

**HealthPlus**

- Home
- Doctors
- Facility Profile
- Logout

**HealthCare Professionals List:**

Professional Name	Speciality	Contact Number	Email	Joining Date	Leaving Date	Actions
Dr. Abha Gupta	Pediatrics	909090909	drabha090@gmail.com	2001-06-07T00:00:00.000Z		<div>Update</div> <div>Cancel</div>
Dr. Sahil Mehta	Cardiology	909090909	drsahil090@gmail.com	2001-04-07T00:00:00.000Z		<div>Update</div> <div>Cancel</div>
Dr Ram Gupta	Cardiology	9896789867	ram@gmail.com	2022-01-06T00:00:00.000Z		<div>Update</div> <div>Cancel</div>

Add New HealthCare Professional

Figure 4.9.3



## 4.10 Blood Donation Camp Dashboard

### 4.10.1 Blood Donation Camps

**HealthPlus**

- Home
- Blood Donation Camps
- Appointments
- Profile Details
- Logout

Welcome, CityHospital

**Camps List**

Camp Name	Date	Start Time	End Time	Address	Units Collected	Active	Camp Actions
City Hospital	6/10/2023	10:00 AM	5:00 PM	, , Rajpura, Punjab, undefined	A+: 0 A-: 0 B+: 0 B-: 0 AB+: 0 AB-: 0 O+: 0 O-: 0	No	<div>Update</div> <div>Cancel</div>

Add a new Camp

Figure 4.10.1



## CODE IMPLEMENTATION

The code can be accessed at: <https://github.com/luckshay/HealthPlus-Project.git>

### 5.1 Home Page

```

1  import React, { useState, useEffect } from "react";
2  import axios from "../../config/axios";
3  import "../../styles/Home.css";
4  import Navbar from "../Navbar";
5  import image1 from "../../assets/logo/patient-registration-logo-homepage.svg";
6  import image2 from "../../assets/logo/health-facility-registration-logo-homepage.svg";
7  import image3 from "../../assets/logo/health-care-professional-registration-homepage.avif";
8  import image4 from "../../assets/logo/blood-donation-homepage.avif";
9  import image5 from "../../assets/images/patient-registration-no-homepage.jpeg";
10 import image6 from "../../assets/images/Health-facility-registry-homepage.jpeg";
11 import image7 from "../../assets/images/Health-care-professionals-registry-homepage.jpeg";
12 import image8 from "../../assets/images/blooddonation-camp-homepage.jpg";
13
14 function Home() {
15   const [activeTab, setActiveTab] = useState(1);
16   const [hoverTab, setHoverTab] = useState(null);
17
18   const handleTabClick = (tabNumber) => {
19     setActiveTab(tabNumber);
20     setHoverTab(null);
21   };
22
23   const handleTabHover = (tabNumber) => {
24     setHoverTab(tabNumber);
25   };
26
27   const handleTabLeave = () => {
28     setHoverTab(null);
29   };
30
31   const tabContent = [
32     {
33       id: 1,
34       title: "Patient Registration",
35       image: image1,
36       content: (
37         <div className="tab-content-wrapper">
38           <div className="tab-content-image">
39             <img src={image5} alt="Patient" />
40           </div>
41           <div className="tab-content-text">
42             <p>
43               Patient registration on a healthcare website involves collecting
44               personal and medical information to initiate medical care. This
45               process helps healthcare providers maintain accurate records,
46               track medical histories, and monitor treatments. By registering as
47               a patient, individuals can ensure timely and effective medical

```

Figure 5 1

## 5.2 About Page

```

1 import React from 'react'
2 import Navbar from './Navbar'
3 import './styles/About.css'
4 const Aboutpage = () => {
5   return (
6     <>
7       <Navbar />
8       <div className="about-container">
9         <div className="section-a align="center">
10           <div className="section-a-content">
11             <h1>HEALTHPLUS: MAKING YOUR LIFE BETTER</h1>
12             <p>Welcome to HealthPlus! We are a healthcare analytics platform dedicated to improving the quality of healthcare services and patient outcomes through data-driven insights and personalized care.</p>
13           </div>
14           <div className="section-a-img">
15             <img src={require("../assets/images/about-us-main-image.jpg")} alt="about-us-main" width="800"/>
16           </div>
17         </div>
18         <div className="section-b align="center">
19           <div className="section-b-content">
20             <h2>OUR MOTTO</h2>
21             <p>At HealthPlus, we believe that every patient deserves the best possible care and that every healthcare provider deserves the tools and resources to deliver it.</p>
22           </div>
23           <div className="section-b-img">
24             <img src={require("../assets/images/about-us-motto-image.jpg")} alt="about-us-motto" width="800"/>
25           </div>
26         </div>
27         <div className="section-c align="center">
28           <div className="section-c-content">
29             <h2>USER-FRIENDLY</h2>
30             <p>Our platform is built on the latest technologies and industry standards, ensuring maximum security, reliability, and scalability. With a user-friendly interface, it's easy for anyone to get started.</p>
31           </div>
32           <div className="section-c-img">
33             <img src={require("../assets/images/about-us-user-friendly-image.jpg")} alt="about-us-user-friendly" width="800"/>
34           </div>
35         </div>
36         <div className="section-d align="center">
37           <div className="section-d-content">
38             <h2>VISION</h2>
39             <p>Our vision is to revolutionize healthcare by leveraging data and technology to improve patient outcomes and reduce costs. We are committed to continuous innovation and growth.</p>
40           </div>
41           <div className="section-d-img">
42             <img src={require("../assets/images/about-us-vision-image.jpg")} alt="about-us-vision" width="800"/>
43           </div>
44         </div>
45       </div>
46     </>
47   )
48 }
49 export default Aboutpage

```

Figure 5.2



## 5.3 Contact Page

```

1 import React, { useState } from "react";
2 import Navbar from "../Navbar";
3 import axios from "../config/axios";
4
5 import "../styles/Contact.css";
6
7 const ContactPage = (props) => {
8   const [formData, setFormData] = useState({
9     name: "",
10    email: "",
11    subject: "",
12    message: "",
13  });
14
15  const handleChange = (event) => {
16    const { name, value } = event.target;
17    setFormData( ...formData, {name: value } );
18  };
19
20  const handleSubmit = async (event) => {
21    event.preventDefault();
22
23    try {
24      const res = await axios.post('/api/userquery', formData);
25      alert(res.data.message);
26    } catch (error) {
27      const res = error.response;
28      alert(res.data.message);
29    }
30  };
31
32  return (
33    <Navbar />
34    <div className="contact-container">
35      <div className="form-container">
36        <h2>Contact Us />
37        <form onSubmit={handleSubmit}>
38          <div className="form-group">
39            <input
40              type="text"
41              name={name}
42              className="form-control"
43              id={name}
44            />
45          </div>

```

Figure 5.3

## 5.4 Policy Page

```

1 import React, { useState, useEffect } from "react";
2 import Navbar from "../Navbar";
3 import { MdPolicy, MdOutlineWeb, MdHealthAndSafety } from "react-icons/md";
4 import { BsFillClipboardDataFill } from "react-icons/bs";
5 import "../styles/Policies.css";
6 import { useLocation } from "react-router-dom";
7
8 const Policypage = () => {
9   const [selectedPolicy, setSelectedPolicy] = useState("");
10   const location = useLocation();
11
12   useEffect(() => {
13     const searchParams = new URLSearchParams(location.search);
14     const policyName = searchParams.get("name");
15     setSelectedPolicy(policyName || "Terms and Conditions");
16   }, [location]);
17
18   const handlePolicyClick = (policy) => {
19     setSelectedPolicy(policy);
20   };
21
22   return (
23     <div>
24       <Navbar />
25       <div className="policy-page-container">
26         <div className="policy-sidebar">
27           <div
28             className={`policy-sidebar-item ${selectedPolicy === "Terms and Conditions" ? "selected" : ""}
29             >
30             <div>
31               <div>
32                 <p className="privacy-icons">
33                   <MdPolicy size="2rem" />
34                 </p>
35                 <h3>Terms and Conditions</h3>
36               </div>
37             </div>
38             <div
39               className={`policy-sidebar-item ${selectedPolicy === "Website Policy" ? "selected" : ""}
40             >
41             <div>
42               <p className="privacy-icons">
43                 <MdOutlineWeb size="2rem" />
44               </p>
45               <h3>Website Policy</h3>
46             </div>
47           </div>
48         </div>
49       </div>
50     </div>
51   );
52 }

```

Figure 5.4



## 5.5 Login Page

```

1  client > src > components > pages > Login.js
2  import React, { useState } from 'react';
3  import Navbar from '../Navbar';
4  import { Link, useNavigate } from 'react-router-dom';
5  import axios from '../../config/axios';
6  import './styles/Login_Signup.css';
7  import { FaEye, FaEyeSlash } from 'react-icons/fa';
8
9  const Login = ({ handleLogin }) => {
10   const [formData, setFormData] = useState({
11     email: '',
12     userType: '',
13     password: '',
14   });
15   const [showPassword, setShowPassword] = useState(false);
16
17   const handleChange = (e) => {
18     let val = e.target.value;
19     let name = e.target.name;
20     setFormData({ ...formData, [name]: val });
21   };
22
23   const navigate = useNavigate();
24
25   const handle = async (e) => {
26     e.preventDefault();
27
28     try {
29       const res = await axios.post('/api/auth/login', formData);
30       alert(res.data.message);
31       sessionStorage.setItem('id', res.data.userId);
32       navigate('/dashboard');
33     } catch (error) {
34       const res = error.response;
35       alert(res.data.message);
36     }
37   };
38
39   const togglePasswordVisibility = () => {
40     setShowPassword(!showPassword);
41   };
42
43   return (
44     <div>

```

Figure 5.5

## 5.6 Signup Page

```

client > src > components > pages > # Signup.js > ...
1 import React, { useState } from "react";
2 import Navbar from "../Navbar";
3 import { Link, useNavigate } from "react-router-dom";
4 import axios from "../../config/axios";
5 import "../../styles/Login_Signup.css";
6 import { FaEye, FaEyeSlash } from "react-icons/fa";
7
8 const Signup = () => {
9   const Navigate = useNavigate();
10
11   const [formData, setFormData] = useState({
12     name: "",
13     userType: "",
14     email: "",
15     password: "",
16     confirmPassword: "",
17   });
18
19   const [showPassword1, setShowPassword1] = useState(false);
20   const [showPassword2, setShowPassword2] = useState(false);
21
22   const handleOnChange = (e) => {
23     let val = e.target.value;
24     let name = e.target.name;
25     setFormData({ ...formData, [name]: val });
26   };
27
28   const togglePasswordVisibility1 = () => {
29     setShowPassword1(!showPassword1);
30   };
31
32   const togglePasswordVisibility2 = () => {
33     setShowPassword2(!showPassword2);
34   };
35
36   const handle = async (e) => {
37     e.preventDefault();
38
39     try {
40       const res = await axios.post("/api/auth/register", formData);
41       alert(res.data.message);
42       if (res.status === 201) {
43         Navigate("/login");
44       }
45     } catch (error) {
46       // console.log(error);
47     }
48   };
49
50   return (
51     <div>
52       <Navbar />
53       <div>
54         <h2>Sign Up</h2>
55         <form>
56           <input type="text" value={formData.name} />
57           <input type="text" value={formData.userType} />
58           <input type="text" value={formData.email} />
59           <input type={showPassword1 ? "text" : "password"} value={formData.password} />
60           <input type={showPassword2 ? "text" : "password"} value={formData.confirmPassword} />
61           <button type="button" value="Sign Up" />
62         </form>
63       </div>
64     </div>
59

```

Figure 5.6



## **SYSTEM TESTING**

At the development level, each function written was tested in isolation for its functionality and then added to the development environment. Then, after developing the single functionality, it was unit tested to check if it behaves as expected. We included each and every test case we can think of. Afterwards, integration testing was performed to check for any integration issues. At the end, when the application was deployed, we performed Performance testing and Regression testing to check how the application performed in different scenarios. Finally, the team reviewed the metrics and found it at par with the standards.

## LIMITATIONS

### Limitations of HealthPlus Project:

- **Connectivity Dependence:** The functionality of HealthPlus relies heavily on internet connectivity. Users may face limitations in accessing the app or performing certain functions in areas with poor or no internet connectivity.
- **User Adoption:** The success of HealthPlus depends on user adoption and willingness to embrace digital healthcare solutions. Some users, particularly elderly or less tech-savvy individuals, may face challenges in adapting to the app, limiting its reach.
- **HealthPlus focuses on specific healthcare management aspects,** such as appointment scheduling and medical record management. It may not encompass all aspects of healthcare delivery, and users may need to rely on additional resources or platforms for comprehensive healthcare needs.

Despite these limitations, HealthPlus offers valuable healthcare management functionalities and has the potential to enhance patient care, improve efficiency, promote digital transformation, increase accessibility to healthcare services, streamline medical record management, facilitate effective communication between patients and healthcare providers, and provide convenient appointment scheduling. The application strives to overcome the limitations through continuous improvements, addressing user feedback, and incorporating emerging technologies to ensure a comprehensive and user-centric healthcare solution.

Overall, HealthPlus is a modern healthcare management solution that streamlines services, improves patient care, and enhances accessibility. Despite limitations, it offers valuable functionalities for appointment scheduling, medical records, and secure communication. HealthPlus aims to revolutionize healthcare for better outcomes.

## CONCLUSION

In conclusion, The HealthPlus project is a comprehensive healthcare management solution that leverages modern technologies to streamline various aspects of healthcare services. It offers valuable functionalities such as appointment scheduling, digital medical record management, and secure communication between patients and healthcare providers.

The project demonstrates the potential of the MERN stack in developing robust and user-centric healthcare applications. Despite limitations, the HealthPlus project holds great promise in improving healthcare accessibility, efficiency, and patient care.

The successful development of the HealthPlus project highlights the significance of technology in revolutionizing the healthcare industry and addressing key challenges in healthcare management.

I hope that you have found HealthPlus to be a useful and enjoyable resource. Thank you for using HealthPlus!

## **FUTURE SCOPE**

The future scope of HealthPlus is vast. There are many ways that the project can be improved and expanded. Here are a few ideas:

- Collaborations with healthcare institutions, clinics, and hospitals can lead to partnerships and integrations, making HealthPlus a preferred platform for healthcare providers and patients.
- Continuous updates and improvements in security measures can ensure the protection of sensitive health information and maintain data privacy.
- The project can consider incorporating electronic health records (EHR) interoperability standards to facilitate seamless data exchange between different healthcare systems and improve continuity of care.
- Exploring possibilities for AI-driven chatbots or virtual assistants can enhance user experience by providing instant support and guidance.

I believe that with its strong foundation and continuous development, the HealthPlus project has the potential to revolutionize the healthcare industry and improve access to quality healthcare services.

## **BIBLIOGRAPHY / REFERENCES**

1. MERN Documentation:

- a. MongoDB. (n.d.). Retrieved from <https://www.mongodb.com/>
- b. Express.js. (n.d.). Retrieved from <https://expressjs.com/>
- c. React. (n.d.). Retrieved from <https://reactjs.org/>
- d. Node.js. (n.d.). Retrieved from <https://nodejs.org/>

2. Chat-GPT:

- a. Open AI. (n.d.). Chat-GPT. Retrieved from <https://chat.openai.com/>

3. NDHM App:

- a. National Digital Health Mission. (n.d.). Retrieved from <https://ndhm.gov.in/>