

DOCBOOK

(A Full Stack Doctor Appointment Management System)

A Project Report Submitted to Cotton University in Partial Fulfilment of the
Requirements for the Degree of
Master of Computer Application (MCA)

In the Department of Computer Science and Information Technology
By
Akshayata Das
MCA2265004
MCA IV Semester



Under the Guidance of **Dr. Kangkana Bora**
Assistant Professor,
Dept. of Computer Science & IT,
Cotton University, Guwahati-01

DEPARTMENT OF COMPUTER SCIENCE AND IT
COTTON UNIVERSITY, GUWAHATI
ASSAM-781001, INDIA
APRIL 2024

DECLARATION

I, **Akshayata Das**, bearing MCA Enrolment No.: MCA2265004 hereby declare that the subject matter of the project entitled "***DOCBOOK: A Full stack Doctor Appointment Management System***" is the record of work done by me under the guidance of Name of the Guide, Department of Computer Science and Information Technology, Cotton University, Guwahati-01, Assam. I further declare that the contents of this project report did not form the basis for the award of any degree to me or to anybody else to the best of my knowledge. The report has not been submitted to any other University or Institution. This report is being submitted to Cotton University, Guwahati-01 for the degree of MCA in the Department of Computer Science and Information Technology.

Akshayata Das
MCA IV Semester
Enrolment No. : MCA2265004
Dept. of Computer Science & IT,
Cotton University, Guwahati-01

Place:

Date:

ACKNOWLEDGEMENT

I am extremely grateful to my supervisor respected **Dr. Kangkana Bora**, Assistant Professor, Department of Computer Science and Information Technology, Cotton University, for her valuable advice, encouragement, endless support and guidance with patience during the course of my project work and write up this project report.

I would like to express my sincere gratitude to the Head of the Department, all the faculty members, students, non-teaching staff and office staff members of the Department of Computer Science and Information Technology, Cotton University, Guwahati-01 for their help and moral support in various forms during the course of my project work.

I remember forever my father for all his motivation and help to reach a greater position in my life. I am also very much thankful to my mother and other family members for their infinite love, motivation, suggestions and supports.

Akshayata Das

MCA2265004
MCA IV Semester

ABSTRACT

The Doctor Appointment System is a web application built on the MERN (MongoDB, Express.js, React.js, Node.js) stack, incorporating technologies such as Axios, Redux, Reducer, and JWT (JSON Web Tokens). This system aims to streamline the process of scheduling appointments for both patients and healthcare providers, offering a user-friendly interface and robust security features. Patients can easily schedule appointments with doctors based on their availability and specialization, while doctors can manage their profiles, including personal information and clinic details. Patient profile management allows users to update their information and view appointment history

CONTENT

Chapter 1 Introduction

1.1 Project Title	1
1.2 Objectives of the Project	1
1.3 Existing System	1

Chapter 2 Feasibility Study

2.1 Technical Feasibility	2
2.1.1 Hardware Requirements (Developer's Environment)	2
2.1.2 Software Requirements (Developer's Environment)	2
2.1.3 Software & Hardware Requirements (User's Environment)	3
2.2 Technologies Used	3
2.3 Work Breakdown Structure	4
2.4 Gantt Chart	5

Chapter 3 Design Diagrams

3.1 DFD Diagram (For Patient and Doctor)	6
3.2 DFD Diagram (For Admin)	7

Chapter 4 Implementation

4.1 WELCOME PAGE	8
4.2 LOGIN PAGE	8
4.3 SIGNUP PAGE	9
4.4 HOME PAGE	9

Chapter 5 Conclusion

Reference	11
-----------	----

CHAPTER 1

INTRODUCTION

1.1 TITLE

DocBook: A Full Stack Doctor Appointment Management Application

1.2 OBJECTIVE OF THE PROJECT

The major objectives for the development of the recipe application using the MERN stack are as follows:

- Develop an intuitive and responsive Doctor Appointment Web App using the MERN stack.
- Implement secure user authentication and authorization for patients and doctors, including registration, login, logout, and password management.
- Enable doctors to manage their profiles, including personal information, specialization, availability, and clinic/hospital details.
- Provide patients with profile management capabilities, including personal information, medical history, and preferred doctors.
- Allow patients to update their profiles and view appointment history.
- Create an admin dashboard for managing user accounts, doctor profiles, appointment schedules, and system configurations, with monitoring and moderation capabilities.

1.3 EXISTING SYSTEM

The current Doctor Appointment Management system relies on manual processes, leading to inefficiencies and delays in scheduling appointments. To address these challenges, DocBook proposes a digital platform built on the MERN stack. This platform streamlines appointment scheduling by offering a user-friendly interface for patients to book appointments conveniently.

CHAPTER 2

FEASIBILITY STUDY

2.1 TECHNICAL FEASIBILITY

2.1.1 Software Requirements (Development environment)

Table 2.1.1 Software requirement

SL NO.	TYPE	REQUIREMENT
1	FRONTEND	HTML5, CSS, ReactJS
2	BACKEND	NodeJS, ExpressJS, Mongoose
3	PROGRAMMING LANGUAGE	Javascript
4	BROWSER	Google Chrome
5	Operating System	Mac OS 64 bit
6	Application Software/platform	VS Code
7	Database	MongoDB

2.1.2 Hardware Requirements (Development environment)

Table 2.1.2 Hardware requirement

SL NO.	TYPE	REQUIREMENT
1	Processor	Intel i3/M1/Ryzen 3(recommended)
2	RAM	4GB (recommended)
3	HDD	10GB (recommended)

2.1.3 Software & Hardware Requirements (Users environment)

Table 2.1.3 Software and Hardware requirement

SL NO.	TYPE	REQUIREMENT
1	DEVICE (hardware)	Smartphone, Tablet, Computer
2	OPERATING SYSTEM	Android/IOS(for Smartphone or Tablet) Windows/Linux/MacOS(for Computer)
3	RAM	Min 2GB

2.2 TECHNOLOGIES USED

The Doctor Appointment Project utilizes a variety of technologies and tools to create a comprehensive and efficient system for managing doctor appointments. Here's a short note highlighting the key technologies used:

- **MERN Stack:** The project is built using the MERN stack, which consists of MongoDB (database), Express.js (backend framework), React (frontend library), and Node.js (runtime environment). This stack provides a robust foundation for developing scalable and dynamic web applications.
- **Axios:** Axios is used for making HTTP requests from the frontend to the backend server. It simplifies the process of sending asynchronous requests and handling responses, facilitating smooth communication between the client and server.
- **Redux:** Redux is employed for state management in the application. It helps manage the application's state in a centralized store, making it easier to access and modify data across different components. Redux enhances the predictability and maintainability of the application's state management.
- **Redux Reducer:** Redux Reducer is utilized to specify how the application's state changes in response to actions dispatched to the Redux store. Reducers define pure functions that specify how the state transitions occur, ensuring a predictable and consistent state management flow.
- **Material-UI:** Material-UI is a popular React UI framework that provides pre-designed components following the Material Design guidelines. It offers a wide range of customizable UI components, including buttons, forms, and navigation elements, allowing for the creation of aesthetically pleasing and user-friendly interfaces.

- **JSON WEBTOKEN:** JWT is employed for secure authentication and authorization mechanisms in the application. It allows for the generation of tokens containing encoded information about the user's identity and permissions. JWT enables secure transmission of user credentials and helps prevent unauthorized access to sensitive resources.

By leveraging these technologies and tools, the Doctor Project ensures the development of a robust, scalable, and user-friendly application for managing doctor appointments effectively.

2.3 WORK BREAKDOWN STRUCTURE

A WBS provides the necessary framework for detailed cost estimating and control along with guiding schedule development and control.

The total project development time (in hours) for our web application is 840 hrs. To further explain the calculation of hours. Total number of weeks = 21 weeks

So now,

Total number of weeks × Number of days/week × Number of hours of work per day
 $21 \text{ weeks} \times 5 \text{ days/week} \times 8 \text{ hours/day} = 840 \text{ hours}$.

Then, we divided the total hour (840 hrs.) according to the various phases mentioned in the diagram below.

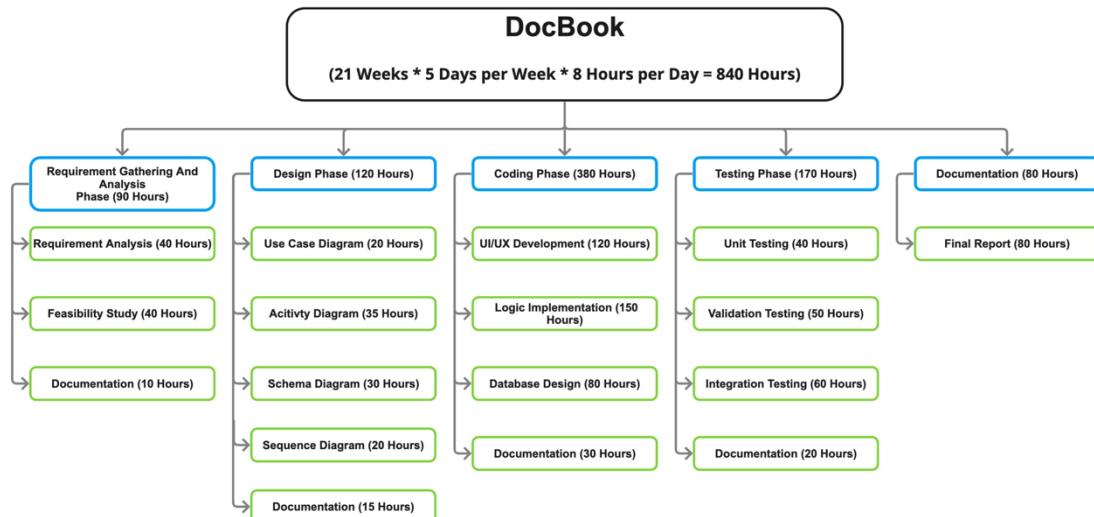


FIG 2.3 WORK BREAKDOWN STRUCTURE

2.4 GANTT CHART

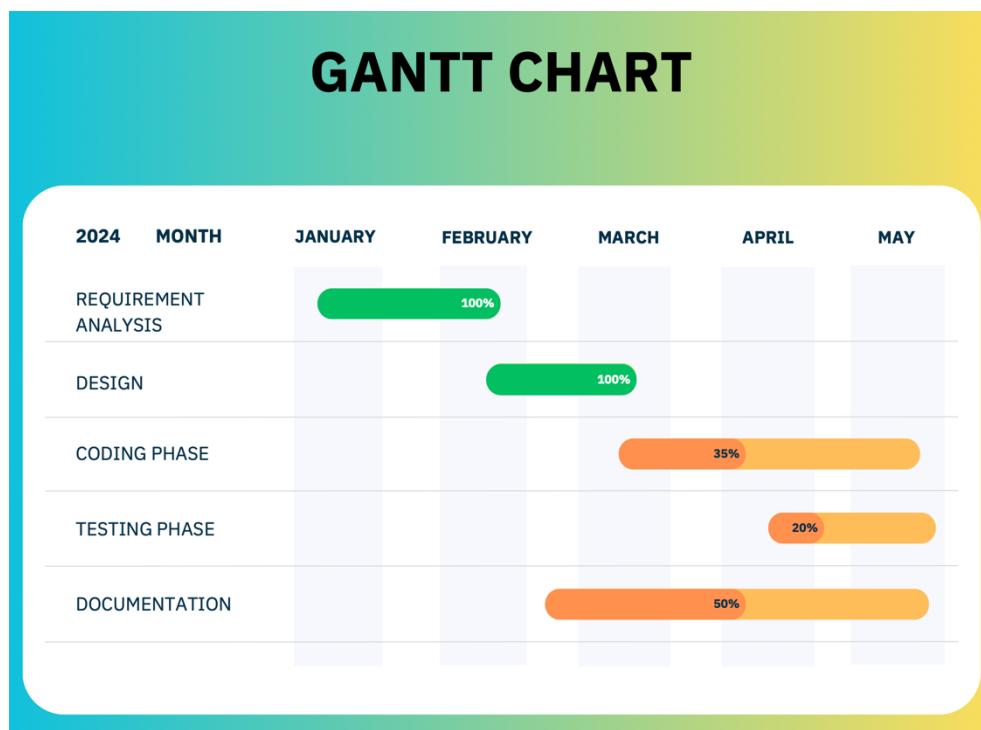


FIG 2.4 GANTT CHART

CHAPTER 3 DESIGN DIAGRAM

3.1 DFD DIAGRAM (FOR PATIENT AND DOCTOR)

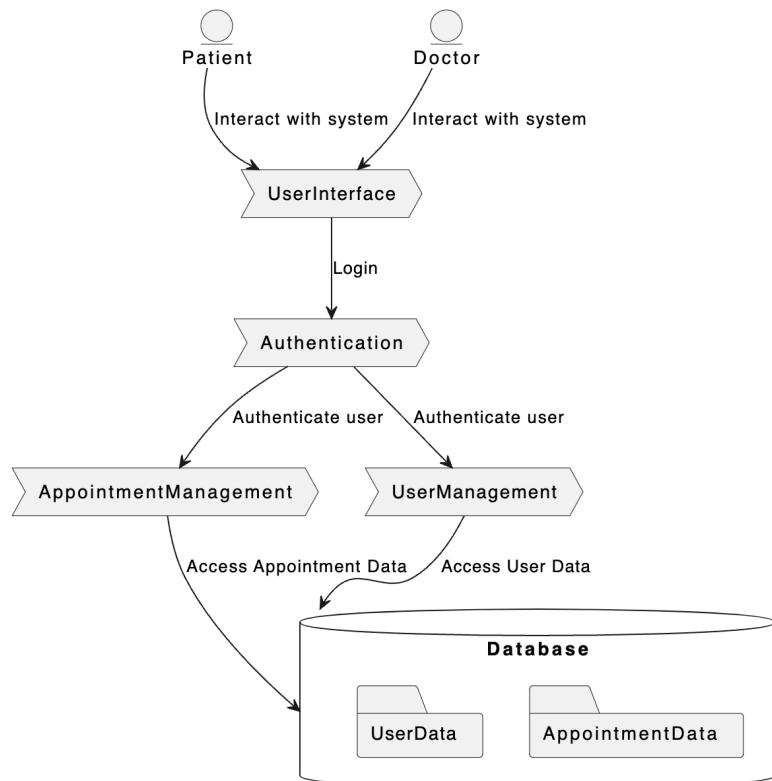


FIG 3.1 DFD (PATIENT AND DOCTOR)

3.2 DFD DIAGRAM (FOR ADMIN)

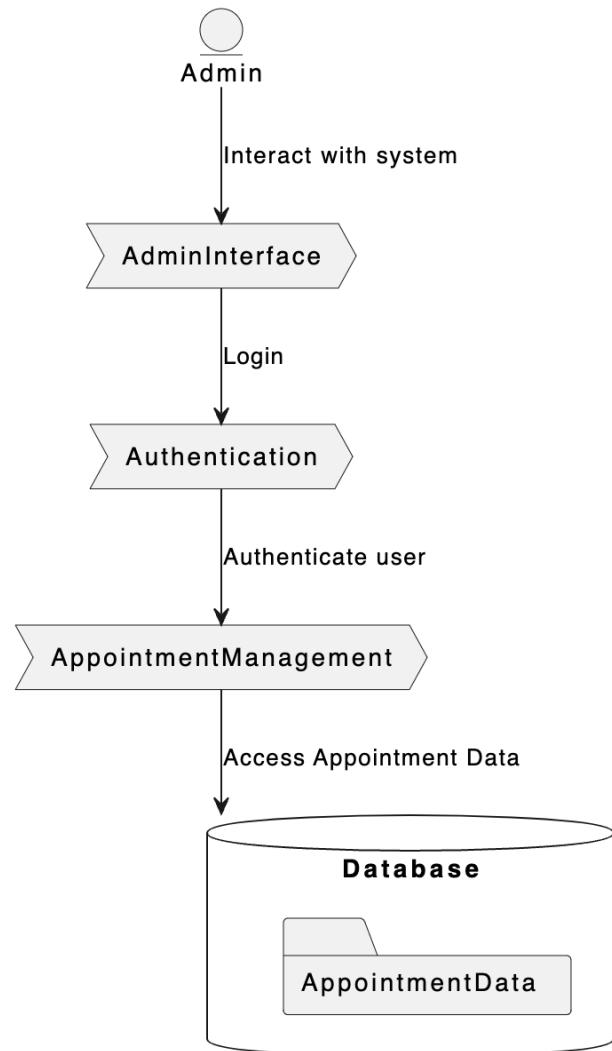
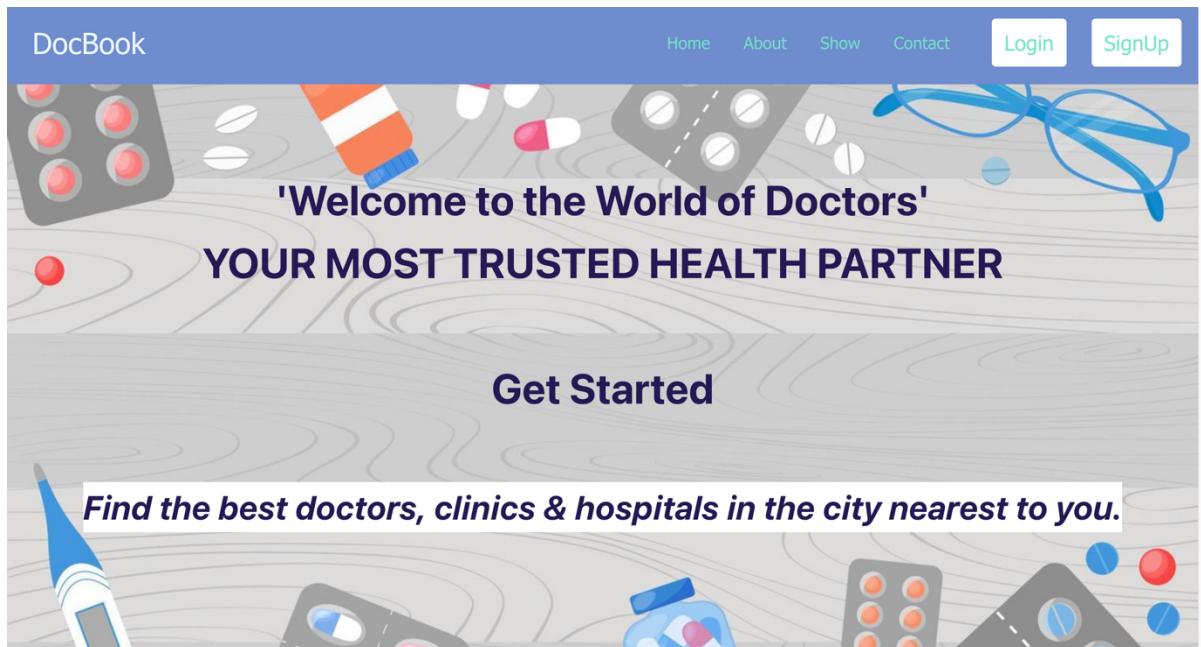


FIG 3.2 DFD (ADMIN)

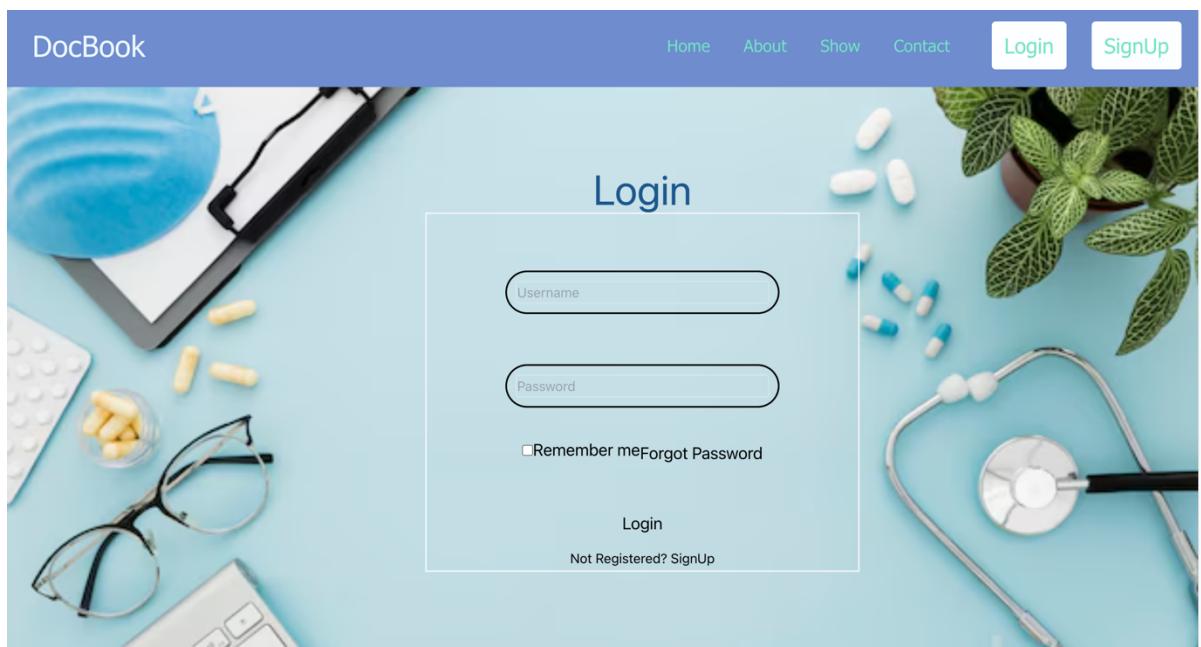
CHAPTER 4

IMPLEMENTATION

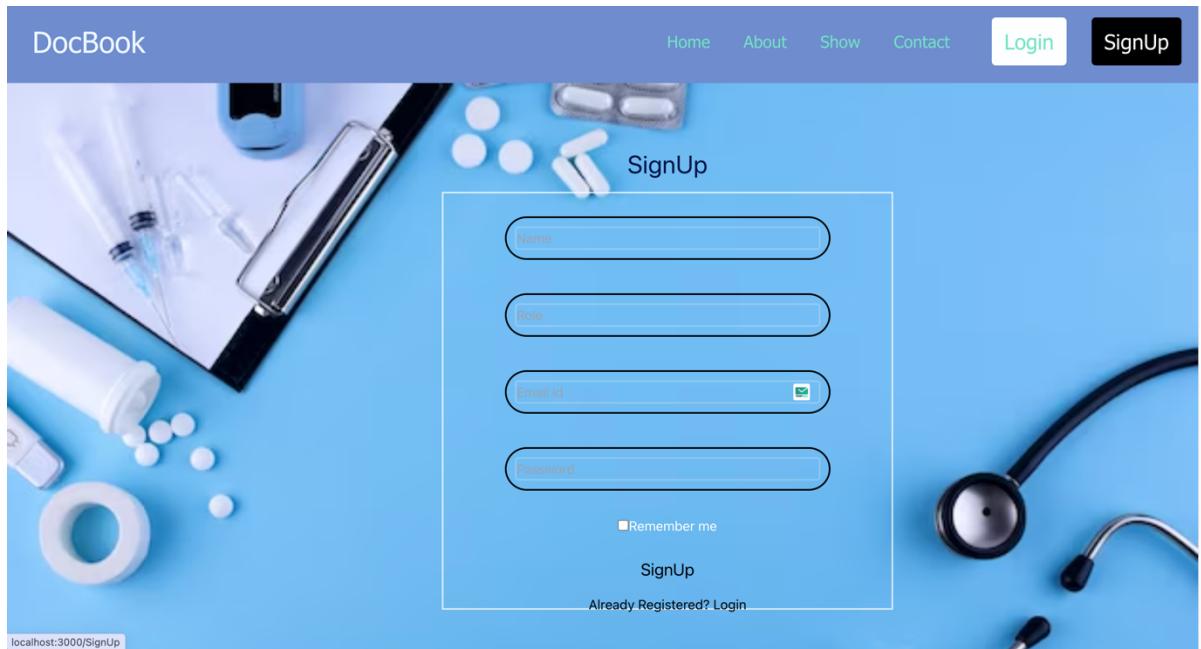
4.1 WELCOME PAGE



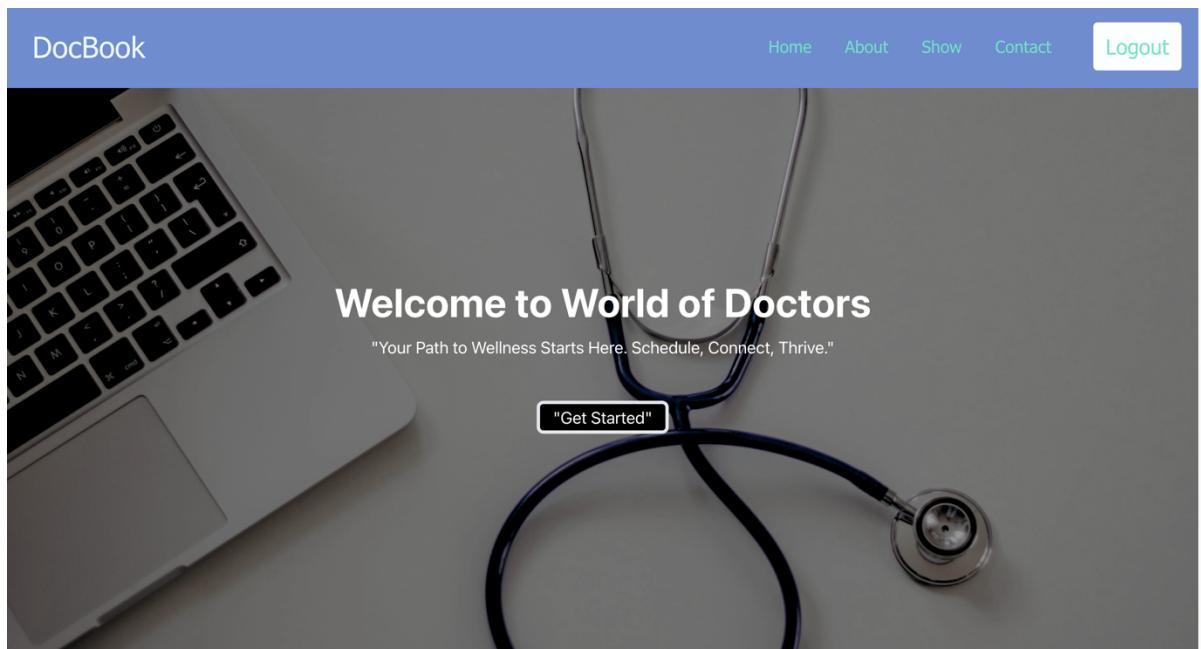
4.2 LOGIN PAGE



4.3 SIGNUP PAGE



4.4 HOME PAGE



CHAPTER 5

CONCLUSION

In conclusion,, the DocBook project, developed using the MERN stack, represents an advancement in modernizing the doctor appointment management process. By leveraging innovative technology, DocBook offers patients and healthcare providers a seamless and efficient platform for booking appointments, managing healthcare needs, and improving overall patient care experiences. With user-friendly features. As a result, DocBook stands poised to significantly improve operational efficiency within healthcare facilities, optimize patient-provider interactions, and ultimately enhance the quality of healthcare delivery.

REFERENCE:

- <https://react-redux.js.org/tutorials/quick-start>
- <https://www.geeksforgeeks.org/mern-stack/>
- <https://www.mongodb.com/docs/>
- <https://axios-http.com/docs/intro>
- <https://mui.com/material-ui/>
- <https://jwt.io/introduction>
- <https://expressjs.com/en/5x/api.html>