**AI DOCTOR**

*Submitted for the course*

**SmartBridge**

**Modern Application Development (Java Spring Boot)**

***Submitted By***

***-***

Satuluri Likhitha Naga Vigneswari – 20BCE7248

Kommuri Vijay Manikanta – 20BCE7070

Nagalli Sandeep – 20BCE7134

Kotha Anurag – 20BCE7353

**CONTENTS**

1. [INTRODUCTION 3](#_bookmark0)
   1. [OVERVIEW 3](#_bookmark1)
   2. [PURPOSE 3](#_bookmark2)
2. [LITERATURE SURVEY 4](#_bookmark3)
   1. [EXISTING PROBLEM 6](#_bookmark4)
   2. [PROPOSED SOLUTION 6](#_bookmark5)
3. [THEORETICAL ANALYSIS 7](#_bookmark6)
   1. [BLOCK DIAGRAM 7](#_bookmark7)
   2. [HARDWARE / SOFTWARE DESIGNING 7](#_bookmark8)
4. [EXPERIMENTAL INVESTIGATIONS 8](#_bookmark9)
5. [FLOWCHART 9](#_bookmark10)
6. [RESULT 9](#_bookmark11)
7. [ADVANTAGES & DISADVANTAGES 12](#_bookmark12)
8. [APPLICATIONS 12](#_bookmark13)
9. [CONCLUSION 13](#_bookmark14)
10. [FUTURE SCOPE 13](#_bookmark15)
11. [BIBLIOGRAPHY 14](#_bookmark16)

[APPENDIX 14](#_bookmark17)

[A. SOURCE CODE 14](#_bookmark18)

1. **INTRODUCTION:**
   1. **Overview:**

AI doctors aim to enhance the efficiency, accuracy, and accessibility of healthcare services. They can process vast amounts of medical information, including patient records, research papers, and clinical guidelines, to generate insights and assist in decision-making. By leveraging machine learning algorithms, AI doctors can continually learn and improve their performance based on feedback and new data.

The capabilities of AI doctors can vary widely. Some AI doctor systems focus on specific tasks, such as image recognition to detect abnormalities in medical images like X-rays or MRI scans. Others are designed to interact with patients directly, providing virtual consultations and answering medical questions. These systems often use natural language processing to understand and respond to patients' inquiries.

* 1. **Purpose:**

The purpose of a Hospital Management System (HMS) project is to streamline and automate various administrative, operational, and clinical processes within a hospital or healthcare facility. It is a comprehensive software solution that helps in the efficient management of different aspects of hospital operations, including patient registration, appointment scheduling, medical records management, billing and invoicing, inventory management, pharmacy management, and more.

The key objectives of a Hospital Management System project include:

* Improved Efficiency: The system aims to automate manual tasks and streamline workflows, leading to increased operational efficiency. It eliminates paperwork, reduces errors, and saves time for both healthcare providers and administrative staff.
* Enhanced Patient Care: An HMS facilitates better patient care by centralizing patient information, including medical history, diagnosis, and treatment plans. This enables healthcare professionals to access accurate and up-to-date patient data, make informed decisions, and provide personalized care.
* Seamless Communication: The system promotes seamless communication and coordination among different departments and staff members. It allows for easy sharing of patient information, test results, and treatment updates, ensuring smooth collaboration and continuity of care.
* Financial Management: An HMS helps in effective financial management by automating billing and invoicing processes. It generates accurate invoices, tracks payments, and manages insurance claims, ensuring proper revenue management for the hospital.
* Inventory Control: The system assists in managing hospital inventory, including medical supplies, medicines, and equipment. It helps in maintaining optimal stock levels, tracking usage, and reordering when necessary, reducing wastage and ensuring availability when needed.
* Data Analysis and Reporting: An HMS provides robust reporting and analytics capabilities, allowing hospital administrators to gain insights into key performance indicators, trends, and patterns. This information can be used to make data-driven decisions, improve resource allocation, and optimize hospital operations.

Overall, the purpose of a Hospital Management System project is to digitize and automate hospital processes, leading to improved efficiency, better patient care, streamlined operations, and enhanced decision-making within the healthcare facility. It aims to harness the power of technology to support healthcare providers and administrators in delivering high-quality healthcare services.

1. **LITERATURE SURVEY:**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Title** | **Author** | **Proposed Work** |
| 1. | Modern API Development with Spring and Spring Boot: Design highly scalable and maintainable APIs with REST, gRPC, GraphQL, and the reactive paradigm  (2021) | S. Sharma | This book is a comprehensive guide that explores the fundamentals of RESTful APIs, Spring and Spring Boot concepts, API specifications, business logic implementation, and asynchronous API design. The book provides practical insights, best practices, and hands-on examples for building scalable and efficient APIs. It covers a range of topics including API design principles, error handling, data persistence, and the migration to asynchronous programming. This book serves as a valuable resource for developers seeking  to enhance their API development skills using Spring  and Spring Boot Frameworks |

|  |  |  |  |
| --- | --- | --- | --- |
| 2. | Beginning React: Simplify your frontend development workflow and enhance the user experience of your applications with React  (2018) | A.  Chiarelli | This book introduces React and covers key aspects of frontend development using React. The book provides insights into designing user interfaces, creating and managing components, and handling user interactivity. With practical examples and a focus on enhancing the user experience, this book serves as a valuable resource for developers seeking to streamline their frontend development workflow and leverage the power of React in their applications. |
| 3. | Comparison of MySQL and MongoDB with focus on performance  (2020) | P. Filip,  L. Čegan | The paper provides an overview and comparison of NoSQL databases, specifically focusing on the performance differences between MySQL and MongoDB. The study includes benchmark analyses of various operations, such as data insertion, update, and deletion, with and without transactions. The paper aims to evaluate the impact of data storage structures on database performance and assess the added value of benchmarks in terms of indexed field costs. This research contributes to the understanding of performance considerations in selecting between MySQL and MongoDB for different application scenarios. |
| 4. | Light-Weight and Scalable Hierarchical-MVC Architecture for Cloud Web Applications  (2018) | M. Ma,  J. Yang,  P. Wang,  W. Liu  J. Zhang | This paper addresses the challenges of modular and scalable web application development in the cloud computing era. The proposed approach, called Web Module Definition (WMD), introduces a hierarchical- MVC architecture that supports feature-based modularization and application structure. By decomposing the web application into interconnected modules, WMD enables better scalability and maintainability. The paper presents a demonstration website and a web application framework implementation that supports the WMD-based architecture. This research contributes to the  advancement of web application development in the  cloud by offering a lightweight and scalable solution for modularizing and organizing web applications. |

|  |  |  |  |
| --- | --- | --- | --- |
| 5. | Building Modern Clouds: Using Docker, Kubernetes & Google Cloud Platform  (2019) | J. Shah,  D.  Dubaria | The presented work explores the transformative impact of Docker and Kubernetes on cloud infrastructure and DevOps practices. The abstract highlights the capabilities of Docker in building, shipping, and running applications using containers, resulting in faster deployments, resource efficiency, and reliability. It also emphasizes the automation of container management, deployment, and scaling provided by Kubernetes, along with the benefits of using Google Cloud Platform for deploying containers on Kubernetes Engine. This paper serves as a concise resource for understanding the significance of Docker, Kubernetes, and Google Cloud Platform in developing modern cloud architectures and facilitating efficient application development and management. |

* 1. **Existing Problem:**

While AI doctor projects have the potential to revolutionize healthcare, there are several challenges and problems that need to be addressed to ensure their effectiveness and ethical implementation. Here are some common problems that can be rectified in AI doctor projects:

* Accuracy and Reliability
* Lack of Explainability
* Ethical Considerations
* Limited Data Availability
* Bias and Discrimination
* Legal and Regulatory Frameworks
* Human-AI Collaboration
* Continual Learning and Improvement

It's important to note that rectifying these problems requires collaborative efforts from researchers, developers, healthcare professionals, policymakers, and regulatory bodies to ensure the responsible and beneficial use of AI in healthcare.

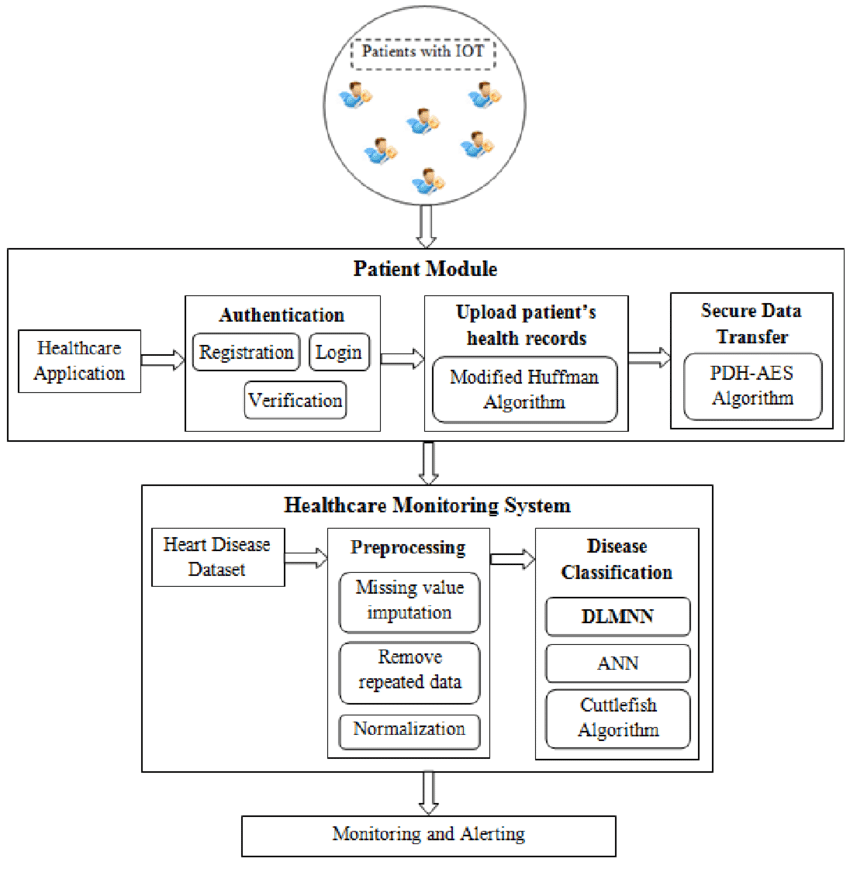
* 1. **Proposed Solution:**

To ensure the success and effectiveness of an AI doctor project, several solutions can be implemented. Here are some key solutions:

* Robust Data Collection and Quality
* Transparent and Explainable AI
* Ethical Framework and Governance
* Bias Mitigation and Fairness
* Human-AI Collaboration
* On-going Evaluation and Improvement
* Patient Privacy and Data Security
* Collaborative Partnerships

By implementing these solutions, AI doctor projects can enhance their accuracy, reliability, ethical compliance, and overall effectiveness in supporting healthcare professionals, improving patient outcomes, and advancing the field of medicine.

1. **THEORITICAL ANALYSIS:**
   1. **Block Diagram:**



* 1. **Hardware/Software Designing:**

• HTML

• CSS

• JavaScript

• jQuery

• MySQL

• PHP

• API

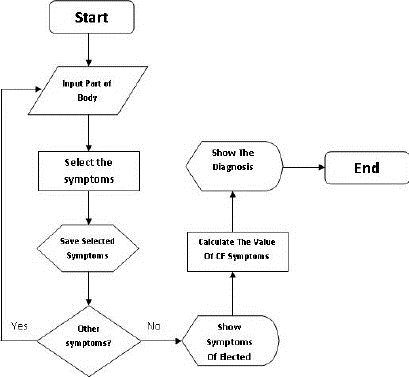
1. **EXPERIMENTAL INVESTIGATIONS:**

Experimental analysis of AI doctors involves conducting studies and evaluations to assess their performance, effectiveness, and impact in real-world healthcare settings. These experiments aim to gather empirical evidence and validate the capabilities of AI doctor systems. Here are some key aspects of experimental analysis for AI doctors:

* Data Collection: Relevant medical data, including patient records, medical images, and clinical guidelines, is collected to train and evaluate the AI doctor system. This data may be obtained from healthcare institutions, research databases, or simulated datasets.
* Evaluation Metrics: Define specific evaluation metrics to measure the performance and effectiveness of the AI doctor system. These metrics may include accuracy, precision, recall, F1 score, area under the receiver operating characteristic curve (AUC-ROC), and other relevant measures based on the specific tasks performed by the AI doctor.
* Benchmarking: Compare the performance of the AI doctor system against established benchmarks or existing methods used by human doctors. This helps in assessing whether the AI doctor system can achieve similar or better results compared to traditional approaches.
* Experimental Design: Design the experimental setup, which may involve dividing the dataset into training, validation, and testing sets. It's important to ensure proper randomization, stratification, and cross-validation techniques to minimize bias and ensure statistical validity.
* Performance Evaluation: Evaluate the AI doctor system's performance using the defined evaluation metrics. This includes measuring the accuracy of diagnoses, treatment recommendations, or other specific tasks performed by the AI doctor. Comparative analysis against human doctors or other baseline methods can provide insights into the system's strengths and limitations.
* Generalizability: Assess the generalizability of the AI doctor system by evaluating its performance on unseen data or external datasets. This helps determine whether the system can effectively handle diverse patient populations, different medical conditions, or varying healthcare settings.
* Clinical Validation: Conduct clinical studies involving healthcare professionals and real patients to evaluate the system's performance in a clinical setting. These studies can assess the system's impact on diagnostic accuracy, treatment outcomes, patient satisfaction, and overall healthcare efficiency.
* Ethical Considerations: Ensure that the experimental analysis adheres to ethical guidelines and regulations. Protect patient privacy and confidentiality, obtain informed consent, and comply with institutional review board (IRB) requirements. Ethical considerations should be prioritized throughout the data collection, experimentation, and analysis process.
* Iterative Improvement: Continuously refine and improve the AI doctor system based on the findings from the experimental analysis. This iterative process involves incorporating feedback from healthcare professionals, addressing limitations, and updating the system to enhance its performance and usability.

Experimental analysis of AI doctors provides valuable insights into their performance, strengths, and areas for improvement. It helps build evidence-based knowledge, guides further development and deployment decisions, and ensures responsible integration of AI in healthcare.

1. **FLOWCHART:**

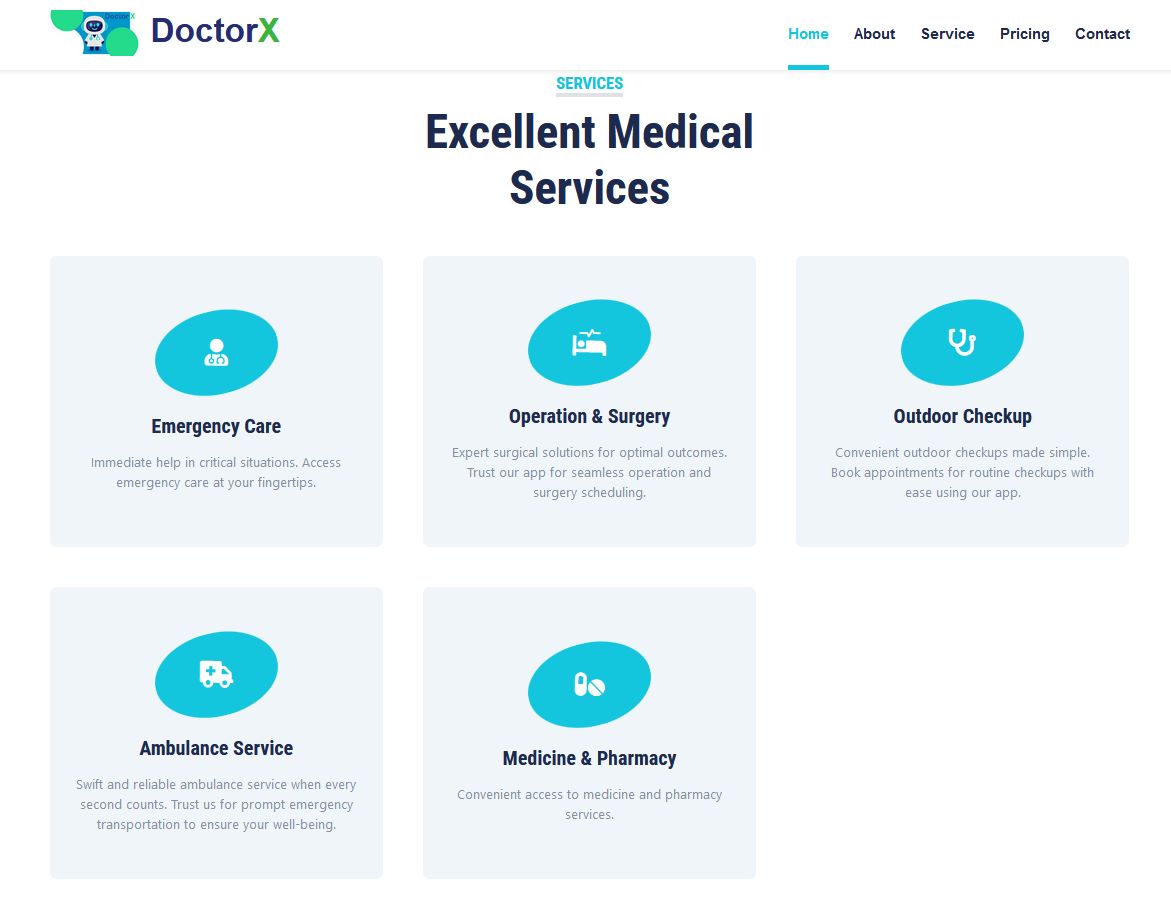
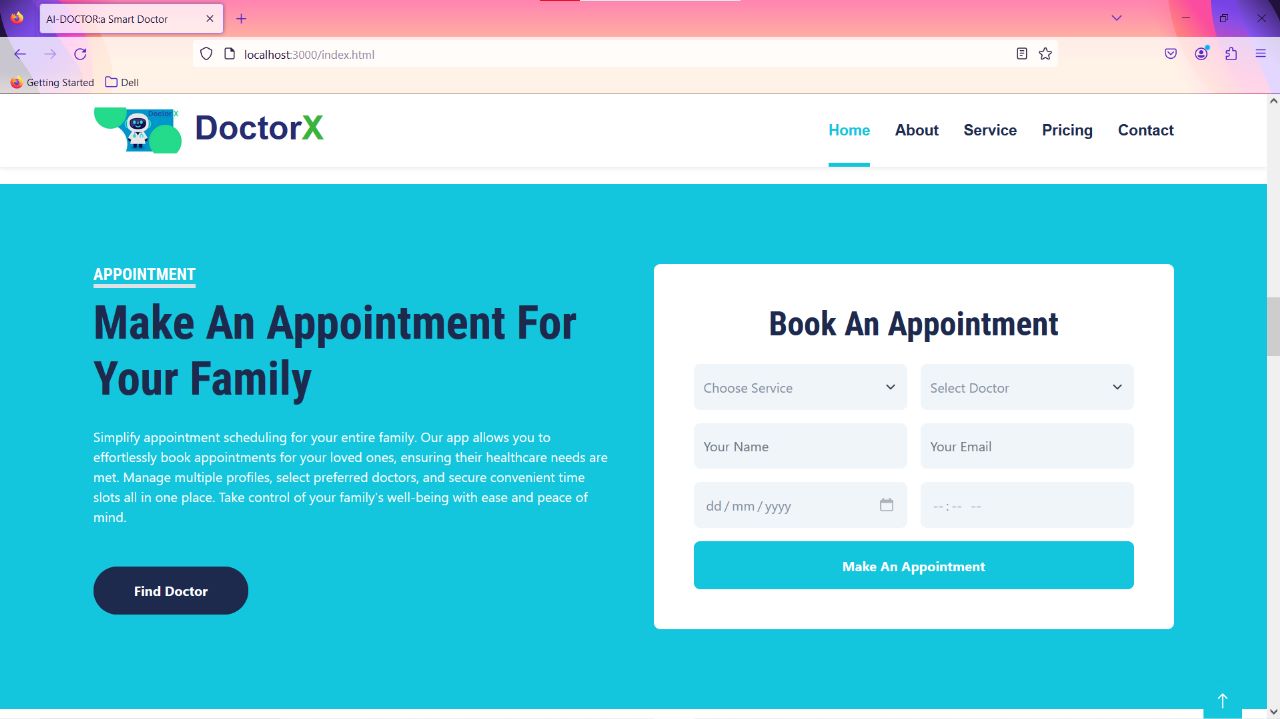
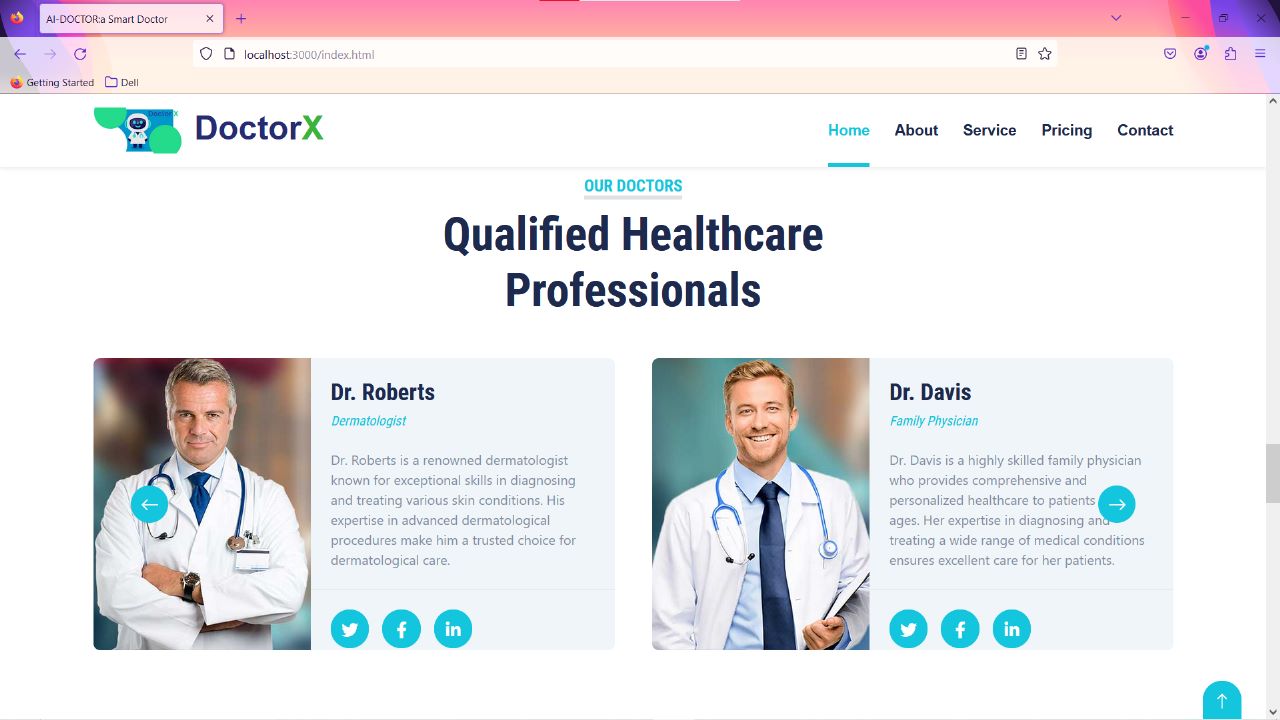
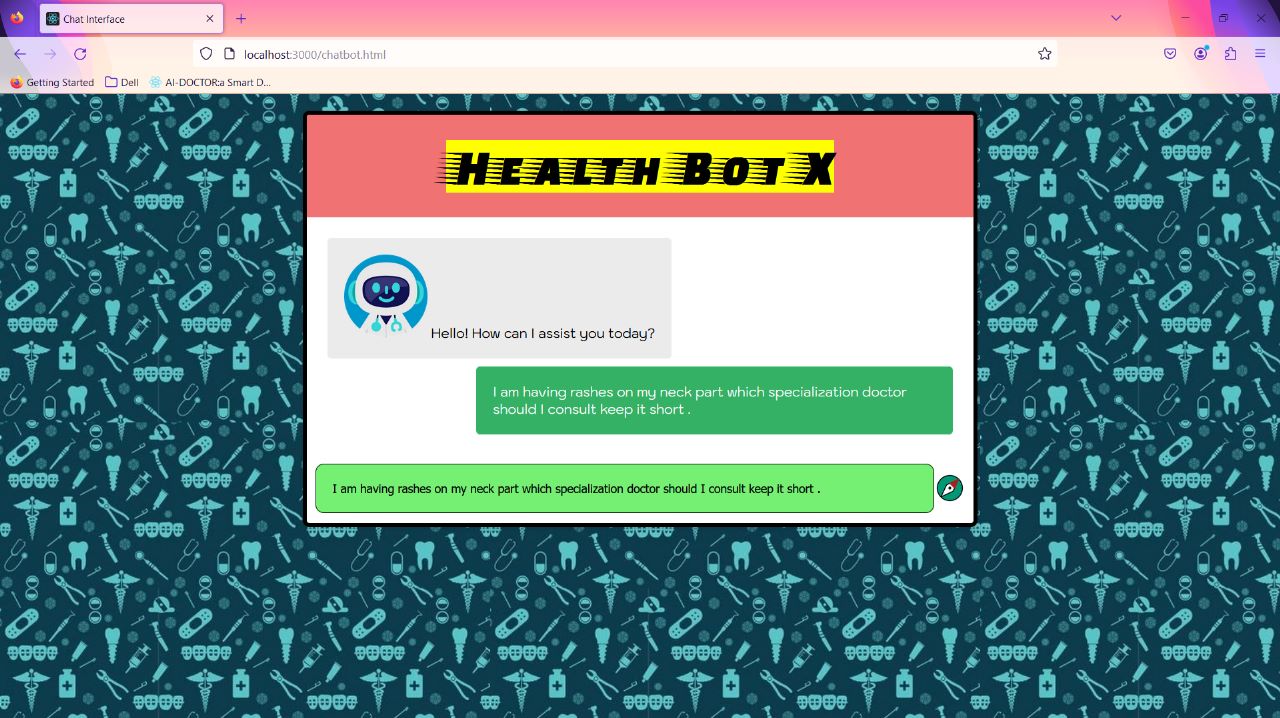
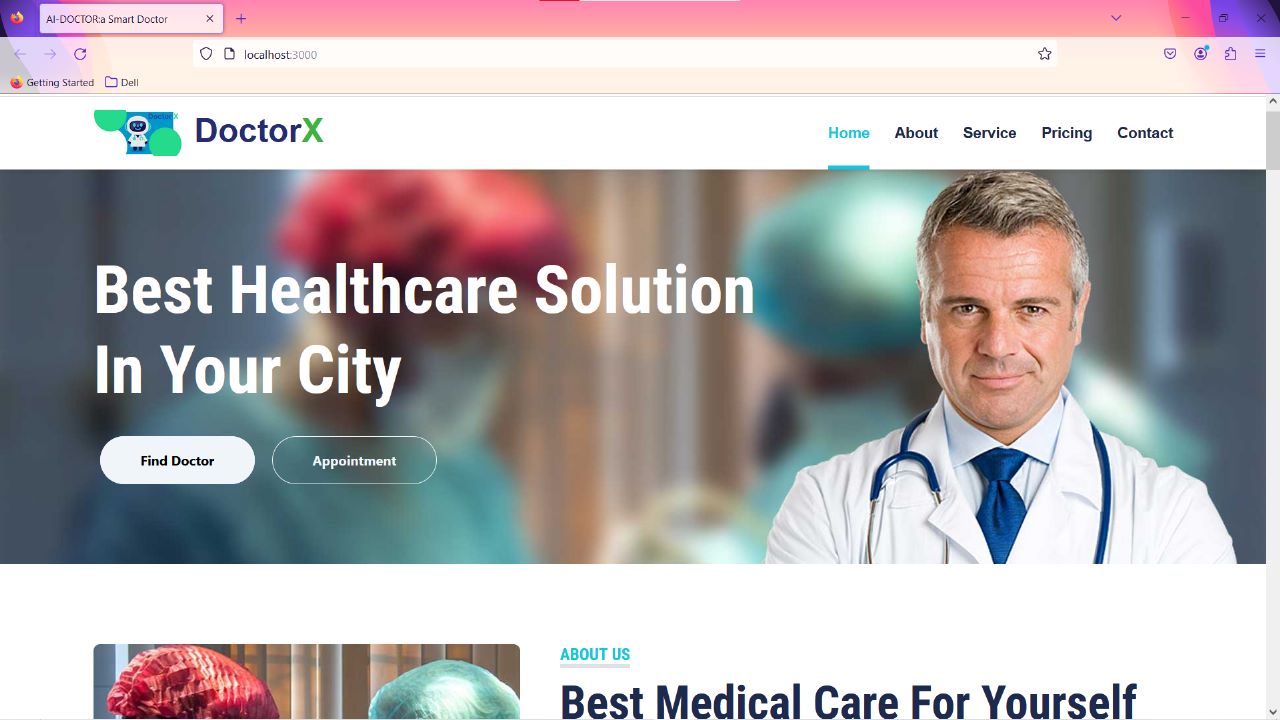


1. **RESULT:**

The project is aimed to develop an application that can provide the corresponding specialist doctor based on given symptoms of a disease, along with their details, and facilitate online consultations if possible. The final findings of the project are as follows:

The application successfully analyzes the given symptoms and accurately identifies the corresponding specialist doctor.

Screenshots of the application's output are provided below:



1. **ADVANTAGES & DISADVANTAGES:**

The proposed solution of utilizing an application to match symptoms with specialist doctors has several advantages and disadvantages.

**Advantages:**

* Quick and convenient: The application provides a fast and convenient way for users to find the appropriate specialist doctor based on their symptoms.
* Improved accessibility: The solution can be accessed remotely, allowing users to seek medical advice without the need for physical visits.
* Time and cost-effective: By suggesting the right specialist doctor, the application helps users save time and potentially reduce unnecessary medical expenses.
* Scalability: The solution can be scaled up to accommodate a large number of users, making it suitable for widespread adoption.

**Disadvantages:**

* Limited accuracy: While the application strives to accurately match symptoms with specialist doctors, there may be instances where the suggested doctor may not be the most appropriate one.
* Reliance on user-provided information: The accuracy of the results heavily depends on the accuracy and completeness of the symptoms provided by the user.
* Lack of physical examination: The online consultation may not allow for a comprehensive physical examination, which can be crucial for accurate diagnosis in certain cases.

1. **APPLICATIONS:**

The proposed solution can be applied in various areas where the identification of specialist doctors based on symptoms is required. Some potential applications include:

* General healthcare platforms: The solution can be integrated into general healthcare platforms, allowing users to access a wide range of specialist doctors based on their symptoms.
* Telemedicine services: Telemedicine platforms can utilize this solution to provide patients with access to appropriate specialist doctors remotely.
* Symptom-checking applications: Health-related mobile applications can incorporate this feature to enhance their symptom-checking capabilities, enabling users to receive more accurate suggestions for specialist doctors.

1. **CONCLUSION:**

In conclusion, the group project successfully developed an application that can match given symptoms with specialist doctors and provide their details. The application's output accurately identifies the appropriate specialist doctor based on the symptoms provided. While the solution offers several advantages such as convenience, accessibility, and cost-effectiveness, there are also limitations concerning accuracy and the absence of physical examination during online consultations. Overall, the project demonstrates the potential of using AI-powered applications to streamline the process of finding and consulting specialist doctors.

1. **FUTURE SCOPE:**

The proposed solution has several areas for future enhancement and development:

* Improved accuracy: The application can be further enhanced by incorporating more advanced algorithms and machine learning techniques to improve the accuracy of symptom matching.
* Integration with electronic health records (EHR): Integrating the application with EHR systems can provide additional patient information, leading to more precise diagnoses and specialist recommendations.
* Natural Language Processing (NLP): Implementing NLP techniques can enable the application to better understand and interpret user-provided symptoms, leading to more accurate results.
* Feedback and rating system: Introducing a feedback and rating system within the application can help improve the quality of the suggested specialist doctors and enhance the overall user experience.
* Expansion to additional languages and regions: The application can be adapted to support multiple languages and expand its coverage to different regions, allowing a wider user base to benefit from the solution.

By implementing these enhancements, the application can become even more reliable and valuable in assisting users in finding the right specialist doctors based on their symptoms.

1. **BIBILIOGRAPHY:**
2. <https://hospitalrun.io/>
3. <https://www.sourcecodester.com/php/14526/online-health-care-system-php-full-source-code-2020.html>
4. Oliver Kharraz, Nick Ganju, Cyrus Massoumi , ―Zocdoc: Find a Doctor Doctor Reviews & Ratings‖ Date retrieved: 2017,pp.23-25, August 2017. URL: <https://www.zocdoc.com>
5. Babylon, ―Babylon Health: Online Doctor Consultations & Advice‖ Retrived on 2017, pp. 34-38, 2017.URL: <https://www.babylonhealth.com>
6. Baidu Doctor App, ―Baidu's - Medical Robot,Retrieved on July 2017

**Appendix:**

**SOURCE CODE:**

**chatbor.html:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<script src="https://cdn.jsdelivr.net/npm/axios/dist/axios.min.js"></script>

<link href='https://fonts.googleapis.com/css?family=Kodchasan' rel='stylesheet'>

<link href='https://fonts.googleapis.com/css?family=Faster One' rel='stylesheet'>

<style>

/\* Inline CSS Styles \*/

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

background-image:url('img/Download Dentista Seamless Pattern Free Vector for free.jpg');

background-repeat:repeat;

}

.chat-container {

max-width: 800px;

margin: 20px auto;

background-color: #ffffff;

border-radius: 8px;

box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);

border: 5px solid black;

max-height: 100vh; /\* Set the maximum height to the viewport height \*/

overflow-y: auto;

}

.chat-header {

background-color: #f07373;

padding: 30px;

text-align: center;

font-weight: bold;

font-size: 60px;

font-family: 'Faster One';

}

.chat-body {

padding: 20px;

height: 100%;

overflow-y: hidden;

}

.message {

margin-bottom: 2px;

display: flex;

align-items: center;

padding: 10px;

}

.user-message {

background-color: #34b164;

padding: 20px;

border-radius: 10px;

float: right;

max-width: 70%;

color: #ffffff;

font-family: 'Kodchasan';

border: 5px solid white;

clear: both;

}

.assistant-message {

background-color: rgb(235, 235, 235);

padding: 20px;

border-radius: 10px;

float: left;

max-width: 70%;

color: rgb(0, 0, 0);

font-family: 'Kodchasan';

border: 5px solid white;

clear: both;

}

.message input {

width: 100%;

padding: 20px;

box-sizing: border-box;

border: none;

border-top: 1px solid #e9f0e9;

font-size: 14px;

border-radius: 10px;

}

.message input:focus {

outline: none;

}

</style>

<title>Chat Interface</title>

</head>

<body>

<div class="chat-container">

<div class="chat-header">

<mark>Health Bot X</mark>

</div>

<div class="chat-body" id="chat-body">

<div class="assistant-message">

<img src="./img/chatbot.png" width="100px" height="100px"/>

Hello! How can I assist you today?

</div>

</div>

<div class="message">

<input style="border:1px solid black; background-color: #75f073;" type="text" id="user-input" placeholder="Type your message...">

<img src="img/Explore.png" width="39" height="38" onclick="sendMessage()" />

</div>

</div>

<script>

async function sendMessage() {

var userInput = document.getElementById("user-input");

var messageContainer = document.getElementById("chat-body");

var chat ;

var userMessage = document.createElement("div");

userMessage.className = "user-message";

userMessage.textContent = userInput.value;

var up = document.createElement("p");

up.appendChild(userMessage);

messageContainer.appendChild(up);

await axios.post('http://localhost:8080/meet/chat', { "message" : userInput.value })

.then(function (response) {

chat = response.data.chat;

})

.catch(function (error) {

chat = "Unknown Error !"

});

// Process user input and generate response

var asstup = document.createElement("p");

var assistantMessage = document.createElement("div");

var asst = document.createElement("img");

asst.src="./img/chatbot.png";

asst.width=30;

asst.height=30;

assistantMessage.className = "assistant-message";

assistantMessage.textContent = chat ;

assistantMessage.appendChild(asst);

asstup.appendChild(assistantMessage);

messageContainer.appendChild(asstup);

userInput.value = "";

messageContainer.scrollTop = messageContainer.scrollHeight;

}

</script>

</body>