A

Web Technology(KCS-652)Project Report

on

AI GPT3 CLONE

Submitted as partial fulfilment for the award of
BACHELOR OF TECHNOLOGY DEGREE
Session 2022-23

in

Computer Science & Engineering
Submitted By:

SHUBHAM SRIVASTAVA Roll. No: 114

UNNATI AGARWAL Roll.No:126

Under the guidance of:

Mr. Ajeet Kumar Bhartee

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS COLLEGE OF ENGINEERING AND TECHNOLOGY,
GREATER NOIDA

AFFILIATED TO
DR. A.P.J ABDUL KALAM TECHNICAL UNIVERSITY,
LUCKNOW, UP

GALGOTIAS COLLEGE OF ENGINEERING & TECHNOLOGY

GREATER NOIDA, UTTAR PRADESH, INDIA- 201306.

CERTIFICATE

This is to certify that the mini-project report entitled "**AI GPT 3 CLONE**" submitted by **Mr. SHUBHAM SRIVASTAVA (Roll.No:114), Ms. UNNATI AGARWAL (Roll.No:126),** to the Galgotias College of Engineering & Technology, Greater Noida, Utter Pradesh, affiliated to Dr. A.P.J. Abdul Kalam Technical University Lucknow, Uttar Pradesh in partial fulfillment for the award of Degree of Bachelor of Technology in Computer science & Engineering is a bonafide record of the project work carried out by them under my supervision during the year 2022-2023.

Mr. Ajeet Kumar Bhartee   **Dr. Vishnu Sharma**

Assistant Professor   **Professor and Head**

Deptt. of CSE   **Deptt. of CSE**

ii

GALGOTIAS COLLEGE OF ENGINEERING & TECHNOLOGY

GREATER NOIDA, UTTER PRADESH, INDIA- 201306.

ACKNOWLEDGEMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend my sincere thanks to all of them.

We are highly indebted to **Mr. Ajeet Kumar Bhartee** for his guidance and constant supervision. Also, we are highly thankful to him for providing necessary information regarding the project & also for his support in completing the project.

We are extremely indebted to **Dr. Vishnu Sharma**, HOD, Department of Computer Science and Engineering, GCET and **Dr. Ajeet Kumar Bhartee**, Project Coordinator, Department of Computer Science and Engineering, GCET for their valuable suggestions and constant support throughout my project tenure. We would also like to express our sincere thanks to all faculty and staff members of Department of Computer Science and Engineering, GCET for their support in completing this project on time.

We also express gratitude towards our parents for their kind co-operation and encouragement which helped me in completion of this project. Our thanks and appreciations also go to our friends in developing the project and all the people who have willingly helped me out with their abilities.

SHUBHAM SRIVASTAVA
UNNATI AGARWAL

ABSTRACT

The Chat GPT Clone using MERN (MongoDB, Express, React, Node.js) is a web-based application that allows users to chat with an AI-powered language[1] model. This project utilizes the GPT-3 architecture, which is a state-of-the-art natural language processing model that can generate human-like responses to text inputs.

The front-end of the application is built using React, which is a popular JavaScript library for building user interfaces. The back-end is built using Node.js and Express, which are popular server-side technologies for building scalable and performant web applications. The data is stored in a MongoDB database, which is a NoSQL database that provides high scalability and performance.

The Chat[2] GPT Clone using MERN allows users to input text messages and receive responses from the GPT-3 language model in real-time. The application provides a clean and modern user interface that is easy to use and navigate. Users can also customize the chat experience by selecting different topics or conversation starters.

Overall, the Chat GPT Clone using MERN is a powerful and flexible application that demonstrates the capabilities of modern web technologies and natural language processing models.

Keywords: Chat GPT Clone, MERN, MongoDB, Express, React, Node.js, GPT-3, natural language processing, real-time chatting, AI-powered language model, customizable chat experience, modern web technologies.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

The ChatGPT Clone using MERN[4] is a web-based application that allows users to chat with an AI-powered language model. This project utilizes modern web technologies and the GPT-3 architecture, which is a state-of-the-art natural language processing model. The MERN stack, consisting of MongoDB, Express, React, and Node.js, is used to build a robust and scalable architecture for the application.

The ChatGPT Clone using MERN is designed to showcase the capabilities of modern web technologies and natural language processing models. The front-end of the application is built using React, which is a popular JavaScript library for building user interfaces. React provides a clean and modern user interface that is easy to use and navigate. The back-end of the application is built using Node.js and Express, which are popular server-side technologies for building scalable and performant web applications. The data is stored in a MongoDB database, which is a NoSQL database that provides high scalability and performance.

The application allows real-time chatting with an AI-powered language model, which can generate human-like responses to text inputs. This feature showcases the capabilities of the GPT-3 architecture, which is a state-of-the-art natural language processing model. Users can customize the chat experience by selecting different topics or conversation starters. This feature provides a more personalized and engaging chat experience.

ChatGPT Clone using MERN[8] is a powerful and flexible application that demonstrates the capabilities of modern web technologies and natural language processing models. The application showcases the potential of the MERN stack to build robust and scalable web applications. The real-time chatting feature with an AI-powered language model and the ability to customize the chat experience provide an engaging and personalized user experience. The ChatGPT Clone using MERN is a great example of how modern web technologies can be used to build powerful and flexible applications.

1.1 MOTIVATION

The motivation behind the development of the ChatGPT Clone using MERN comes from the growing interest in natural language processing and its potential to revolutionize the way we interact with computers. Natural language processing has come a long way in recent years, and the GPT-3 architecture is a state-of-the-art model that has generated a lot of excitement

in the field. The ChatGPT Clone using MERN is designed to showcase the capabilities of this architecture by providing a platform for users to chat with an AI-powered language model in real-time.

Additionally, the MERN stack provides a robust and scalable architecture for building performant web applications. The ChatGPT Clone using MERN demonstrates the potential of the MERN stack to build powerful and flexible applications. This project provides an opportunity to explore the capabilities of modern web technologies and natural language processing models while building a practical and engaging application.

The ability to customize the chat experience by selecting different topics or conversation starters provides a more personalized and engaging user experience. This feature is motivated by the desire to create an application that not only showcases the capabilities of natural language processing but also provides a practical and enjoyable user experience.

The main motive behind the development of the ChatGPT Clone using MERN comes from the desire to explore the capabilities of natural language processing and modern web technologies while building a practical and engaging application. The ChatGPT[9] Clone using MERN provides a platform for users to interact with an AI-powered language model and customize their chat experience, demonstrating the potential of these technologies to revolutionize the way we interact with computers.

1.2 DESCRIPTION

The ChatGPT Clone using MERN is a web-based application that allows users to chat with an AI-powered language model in real-time. The application is built using the MERN stack, which consists of MongoDB, Express, React, and Node.js. The front-end of the application is built using React, a popular JavaScript library for building user interfaces, while the back-end is built using Node.js and Express, which are popular server-side technologies for building scalable and performant web applications. The data is stored in a MongoDB database, which provides high scalability and performance.

The core feature of the application is the ability to chat with an AI-powered language model based on the GPT-3 architecture. Users can enter text inputs into the chat window, and the language model generates human-like responses in real-time. The AI-powered

2

language model is designed to simulate natural conversation, providing users with an engaging and personalized chat experience.

To further customize the chat experience, users can select different topics or conversation starters. These options provide a more personalized and engaging chat experience, making the application more enjoyable to use.

The application also includes features for user authentication and account management, allowing users to create and manage their accounts. This feature provides an additional layer of security and allows users to save their chat history for future reference.

ChatGPT Clone using MERN is a web-based application that allows users to chat with an AI-powered language model in real-time. The application is built using modern web technologies, providing a robust and scalable architecture. The ability to customize the chat experience and the inclusion of user authentication and account management features make the application more engaging and secure.

1.3 DEFINITIONS

1. MERN stack: A set of technologies consisting of MongoDB, Express, React, and Node.js used to build full-stack web applications.

2. MongoDB: A document-oriented NoSQL database used to store data in a flexible and scalable manner.

3. Express: A minimal and flexible web application framework for Node.js used to build scalable and performant web applications.

4. React: A JavaScript library used for building user interfaces and reusable UI components.

5. Node.js: A JavaScript runtime built on Chrome's V8 JavaScript engine used for building scalable and performant web applications.

6. AI-powered language model: A machine learning-based natural language processing model used to generate human-like responses to text inputs.

7. GPT-3 architecture: A state-of-the-art natural language processing model developed by OpenAI, capable of generating human-like responses to text inputs.

8. User authentication: A process of verifying the identity of a user, typically involving the use of usernames and passwords.

9. Account management: A set of features that allow users to create, manage, and delete their accounts, typically including account creation, login, logout, and password reset.

Chapter 2 - Literature Review - This chapter provides an introduction to the model and data on the theory behind model development. It also includes a detailed analysis of how the ideas for this were sourced and collected. Then, when we get to work decisions, we need to complete a literature review.

Chapter 3 - Problem Formulation - Chapter 3 talked about how to come up with a problem. and the problem and its presentation

Chapter 4 – Methodology – This describes the survey design, survey tools, or data collection. This also inculcates information about data analysis and ethics for projects that do not harm society. After that, we finally came to a conclusion about how we did our research and analysis.

Chapter 5 - Implementation - This part contains the actual material of the project. It contains a running image of the app and gives an overview of what it will look like later. Also includes data flow diagrams (DFDs), flow charts and schematics.

Chapter 6 - Project Relevance - This is the final part comparing the model's data analysis with the old model.

Chapter 7 – Conclusions and Future Outlook – This part provides an overview of the findings and conclusions. In it, we will also inform you about our contributions and an overview of our team. Then there is the future projection of our project, how it will benefit our society in the future, and the associated ethics.

CHAPTER 2: LITERATURE REVIEW

Natural language processing (NLP) has made tremendous strides in recent years, driven by advancements in deep learning and the availability of large datasets. NLP models have demonstrated remarkable performance in a variety of applications, including machine translation, sentiment analysis, and speech recognition. One such model is the GPT-3 architecture developed by OpenAI, which has generated a lot of excitement in the NLP community due to its impressive performance in generating human-like responses to text inputs.

In parallel, the development of modern web technologies has led to the emergence of full-stack web development, where developers can use a variety of technologies to create powerful and flexible web applications. One popular technology stack is the MERN stack, which includes MongoDB, Express, React, and Node.js. The MERN stack provides a seamless development experience and enables developers to create robust and scalable web applications.

The combination of NLP models and web technologies has led to the development of chatbots and conversational agents, which can provide users with a personalized and engaging chat experience. These agents can be used in a variety of applications, including customer service, healthcare, and education.

In this literature review, we will explore the latest advancements in NLP and deep learning, as well as the latest developments in web technologies and full-stack web development. We will also examine related work in the area of chatbots and conversational agents, including applications, limitations, and ethical considerations. Finally, we will discuss the unique features and contributions of the ChatGPT Clone using MERN and evaluate its strengths and limitations. Overall, this literature review aims to provide a comprehensive overview of the state-of-the-art in NLP and web technologies and their intersection in the development of chatbots and conversational agents.

2.1 WHAT IS CHAT GPT?

Chat GPT (Generative Pre-trained Transformer) refers to a type of chatbot that uses advanced deep learning techniques, specifically the transformer architecture, to

generate responses to user queries. The transformer model is pre-trained on a large corpus of text data and is capable of understanding the context and nuances of human language. Chat GPTs can be trained on a specific domain or can be trained on general language data to generate responses to a wide range of topics. These chatbots are often used in customer service, language learning, and other applications where there is a need for natural and personalized communication with users. The OpenAI GPT series, including GPT-3, is a prominent example of chat GPTs that have gained widespread attention due to their impressive ability to generate coherent and human-like responses.

2.2 WHAT IS MERN?

MERN is an acronym for a set of technologies used for web development: MongoDB, ExpressJS, ReactJS, and NodeJS. MERN stack allows developers to build dynamic, efficient, and scalable web applications using JavaScript as the primary programming language for both frontend and backend development. MongoDB is a NoSQL database used to store and manage data, while ExpressJS is a web application framework used for building APIs and server-side applications. ReactJS is a frontend library used for building user interfaces, and NodeJS is a runtime environment used for executing JavaScript code on the server-side. Together, these technologies provide a comprehensive and powerful toolkit for building full-stack web applications.

2.3 WHAT IS ARTIFICIAL INTELLIGENCE?

Artificial intelligence (AI) refers to the development of computer systems that can perform tasks that typically require human intelligence, such as visual perception, speech recognition, decision-making, and natural language processing. AI involves the development of algorithms and models that can learn from data and improve over time, without being explicitly programmed for each task.

AI is based on the concept of machine learning, which involves the use of statistical algorithms to identify patterns in data and use them to make predictions or decisions. It also involves the use of neural networks, which are computer systems modeled after the human brain and can learn to recognize complex patterns in data.

AI has numerous applications in various fields, including healthcare, finance, education, transportation, and more. It is used for tasks such as diagnosing diseases,

predicting financial markets, optimizing transportation routes, and providing personalized recommendations to users. AI is also used in developing autonomous systems, such as self-driving cars, drones, and robots.

While AI has the potential to revolutionize various industries and improve our daily lives, it also raises ethical concerns related to privacy, security, and the potential misuse of AI systems. Therefore, it is important to develop and use AI in a responsible and ethical manner.

CHAPTER 3: PROBLEM FORMULATION

Problem formulation is the study and analysis of the problem from which the project was initiated. This chapter consists of a problem domain description (so you can see why we need Chat GPT), a problem description, a block diagram, project goals, and project goals and how to do it. This is a list of issues to resolve as described in the issue description.

3.1 PROBLEM STATEMENT

1. Existing chatbot applications lack natural language capabilities and flexibility of modern NLP models.

2. Web-based applications lack the real-time interaction and personalized experience of a chatbot.

3. Development of NLP models requires significant computational resources and expertise, making it difficult for non-experts to integrate them into their applications.

4. There is a need for a user-friendly and engaging platform that leverages modern NLP models and web technologies to provide a personalized and interactive chat experience.

3.2 OBJECTIVES

The main goal of this project is to implement a chat bot that solves all the problems faced by the previous system by including all these features in the newly implemented chat bot.

The objectives of this are:

➢ To develop an AI-powered chatbot using the GPT-3 architecture that can engage in real-time conversations with users.

➢ To integrate the chatbot with a user-friendly web application developed using the MERN stack.

➢ To showcase the capabilities of modern NLP models and web technologies in creating engaging and personalized chat experiences.

➢ To provide a platform for non-experts to integrate NLP models into their applications without the need for significant computational resources or expertise.

➢ To evaluate the performance of the chatbot using metrics such as response time, accuracy, and user feedback.

➢ To demonstrate the potential for NLP models and web technologies to create innovative and engaging applications that can benefit a wide range of industries and users.

In this chapter, we learned about the needs of this project and the problems it solves to make the application more effective. We examined the main problems of the old app to provide results and some new features of the new app that improve the old app to meet other needs of society.

CHAPTER 4: METHODOLOGY

The methodology for developing the ChatGPT Clone using MERN includes the following steps:

1. Data Collection: Collect a large corpus of text data from various sources such as news articles, books, and websites to train the GPT-3 model.

2. NLP Model Training: Train the GPT-3 model using the collected data to generate natural language responses.

3. API Development: Develop a RESTful API using Node.js and Express.js to interact with the GPT-3 model and handle user requests and responses.

4. Frontend Development: Develop the frontend of the web application using React.js, HTML, CSS, and JavaScript to provide a user-friendly interface for the chatbot.

5. Integration: Integrate the API and frontend of the application to create a seamless user experience.

6. Testing: Test the application for accuracy, response time, and user experience using various testing methodologies such as unit testing, integration testing, and user testing.

7. Deployment: Deploy the application on a cloud platform such as AWS or Heroku to make it accessible to users.

8. Evaluation: Evaluate the performance of the application using metrics such as response time, accuracy, and user feedback to identify areas for improvement.

9. Optimization: Optimize the application for performance and scalability using various techniques such as caching, load balancing, and database optimization.

Overall, this methodology combines the power of modern NLP models with the flexibility and ease of use of web technologies to create an engaging and personalized chatbot application. required to complete the project.

> **Required Software:**
>   o   MONGO DB
>   o   HTML
>   o   CSS

- o JAVASCRIPT
- o NODE JS
- o REACT JS
- o EXPRESS JS
- o VS CODE

- ➢ **Required Hardware:**
  - o LAPTOP/PC WITH ATLEAST 8GB RAM
  - o A STABLE INTERNET CONNECTION.

4.1 PROPOSED WORK AND SPECIFICATIONS
4.1.1 MONGO DB

MongoDB **[Fig 1]** is a popular open-source NoSQL document-oriented database that provides a flexible and scalable way to store and retrieve data. Unlike traditional relational databases, MongoDB[22] stores data in JSON-like documents with dynamic schemas, making it easy to handle unstructured and semi-structured data. It uses a concept called collections to group related documents together, and each document can have its own unique structure. MongoDB supports various queries, indexing, and aggregation operations, allowing developers to perform complex queries and retrieve data quickly. Additionally, it offers features like replication, sharding, and load balancing, making it easy to scale horizontally as per the growing needs of applications. MongoDB is widely used in modern web applications, big data, IoT, and other applications that require a flexible and scalable database solution.

4.1.2 NODE JS

Node.js **[Fig 2]** is an open-source, cross-platform, back-end JavaScript runtime environment that allows developers to build server-side applications using JavaScript. Node.js[23] uses an event-driven, non-blocking I/O model, which makes it lightweight, efficient, and scalable. It is built on top of the Google Chrome's V8 JavaScript engine, which compiles JavaScript code to native machine code, making it fast and performant.

Node.js provides a vast library of modules that can be used to perform various tasks, including network programming, file system I/O, data streams, and more. It also supports various frameworks and tools for building web applications, including Express.js, Nest.js, and

Hapi.js. Node.js is widely used for developing web servers, REST APIs, real-time applications, and command-line tools. Its popularity has grown significantly in recent years, as it allows developers to use a single language, JavaScript, for both client-side and server-side development, making it easier to build full-stack web applications.

4.1.3 REACT JS

React.js **[Fig 3]** (or simply React) is a popular open-source JavaScript library for building user interfaces. Developed by Facebook, React[24] allows developers to build complex UI components in a modular, reusable, and efficient way. It uses a declarative approach to describe how the UI should look at any given time, based on the current application state.

React uses a virtual DOM (Document Object Model) to render components efficiently, which updates only the necessary parts of the UI when changes occur. This makes it fast and efficient, as it reduces the need to re-render the entire UI every time there is a change.

React also supports a component-based architecture, where each UI element is a separate and independent component that can be reused across multiple applications. It supports various features like state management, event handling, and data binding, making it a powerful tool for building complex and interactive user interfaces.

React has gained significant popularity in recent years, and it is widely used in building web applications, mobile apps, and desktop applications. It is also compatible with other libraries and frameworks, allowing developers to integrate it with other tools and technologies.

4.1.4 EXPRESS JS

Express.js **[Fig 4]** (or simply Express) is a popular open-source web application framework for Node.js. It provides a robust set of features for building web applications and APIs, including routing, middleware, templating, error handling, and more.

Express[25] is known for its simplicity and flexibility, allowing developers to build scalable and efficient web applications quickly. It uses a minimalist approach to web development, where developers can pick and choose the components they need and configure them according to their requirements.

Express is highly extensible and can be integrated with various middleware modules to provide additional functionality, such as authentication, logging, compression, and more. It also supports various templating engines, making it easy to render dynamic views on the server-side.

Express has gained significant popularity in the Node.js community and is widely used in building web applications, APIs, and microservices. It is also compatible with various front-end frameworks, such as React, Angular, and Vue, allowing developers to build full-stack web applications using a single language, JavaScript.

4.1.5 HTML

HTML **[Fig 5]** (Hypertext Markup Language) is a markup language used for creating web pages and other types of digital content that can be displayed on the World Wide Web. HTML uses tags and attributes to describe the structure, content, and formatting of web pages. With HTML, developers can create text, images, hyperlinks, multimedia content, forms, and other interactive elements that can be accessed by users through a web browser. HTML is a foundational technology for web development, and it is commonly used in conjunction with CSS and JavaScript to create dynamic and interactive web pages.

4.1.6 CSS

CSS **[Fig 6]** (Cascading Style Sheets) is a stylesheet language used for describing the presentation and styling of HTML (Hypertext Markup Language) and XML (Extensible Markup Language) documents, including web pages. CSS allows developers to separate the presentation of a document from its content, making it easier to maintain and update the appearance of a website. With CSS, developers can define the layout, colors, fonts, and other visual elements of a web page, as well as create dynamic effects and animations. CSS works by selecting HTML elements and applying rules that determine how they should be displayed on the page. It is an essential technology for modern web development and is often used in conjunction with HTML and JavaScript to create fully functional and responsive web applications.

4.1.7 JAVASCRIPT

JavaScript **[Fig 7]** is a programming language used primarily for creating interactive web pages and dynamic user interfaces. It was developed by Netscape in the mid-1990s as a scripting language for web browsers, but has since evolved into a versatile programming language that can be used for both front-end and back-end web development, as well as for creating desktop and mobile applications. JavaScript is a client-side language, meaning that it runs in the web browser of the user, rather than on a server. It can be used to add interactivity, validate form input, create animations, and manipulate the Document Object Model (DOM) of a web page. In recent years, JavaScript has become increasingly popular, in part due to the rise of single-page applications and frameworks such as React, Angular, and Vue.js.

4.1.8 GPT 3

GPT-3 **[Fig 8]** (Generative Pre-trained Transformer 3) is a state-of-the-art language processing model developed by OpenAI. It is an artificial intelligence model that uses deep learning techniques to generate human-like text, translate between languages, answer questions, and complete tasks such as summarizing text, generating poetry, and composing music. GPT-3 is trained on a massive corpus of text, including books, articles, and web pages, and has the ability to learn from this data to generate new content that is indistinguishable from human writing. It uses a transformer architecture, which allows it to process vast amounts of data in parallel, making it one of the most powerful language processing models available today. GPT-3 has numerous applications, from automating customer service to generating creative writing and aiding in scientific research. However, its capabilities have raised concerns about the potential misuse of AI-generated content and the ethical implications of deploying such technology in the real world.

Fig 1 mongo DB Logo

Fig 2 Node JS Logo
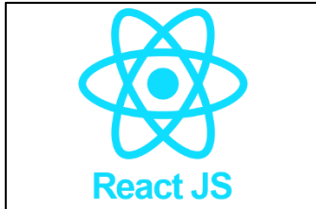
Fig 3 React JS Logo

Fig 4 Express JS Logo
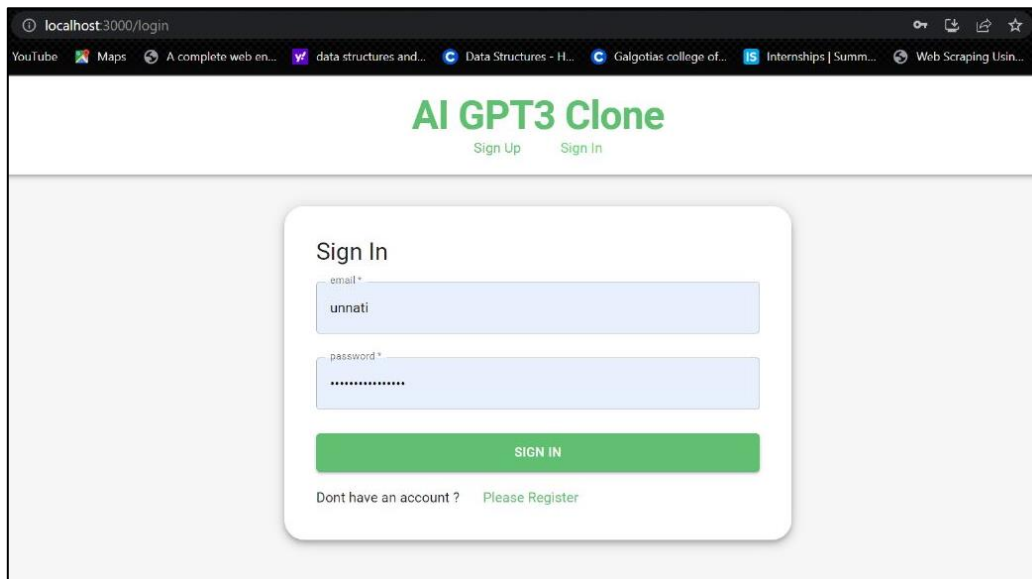

Fig 5 HTML Logo

Fig 6 CSS Logo

Fig 7 JavaScript Logo
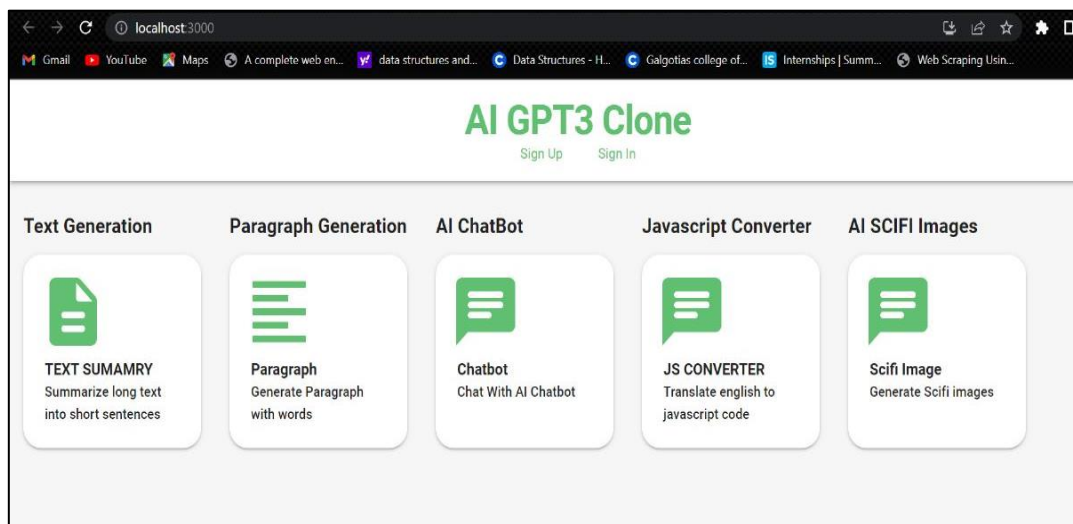

Fig 8 GPT-3 Logo

CHAPTER 5: IMPLEMENTATION

5.1 SCREENSHOTS

Fig 9 Login Page

It is to login a user by its user name and password. Users have to enter their username and password in order to move further in the website. This is done to provide security to the website.
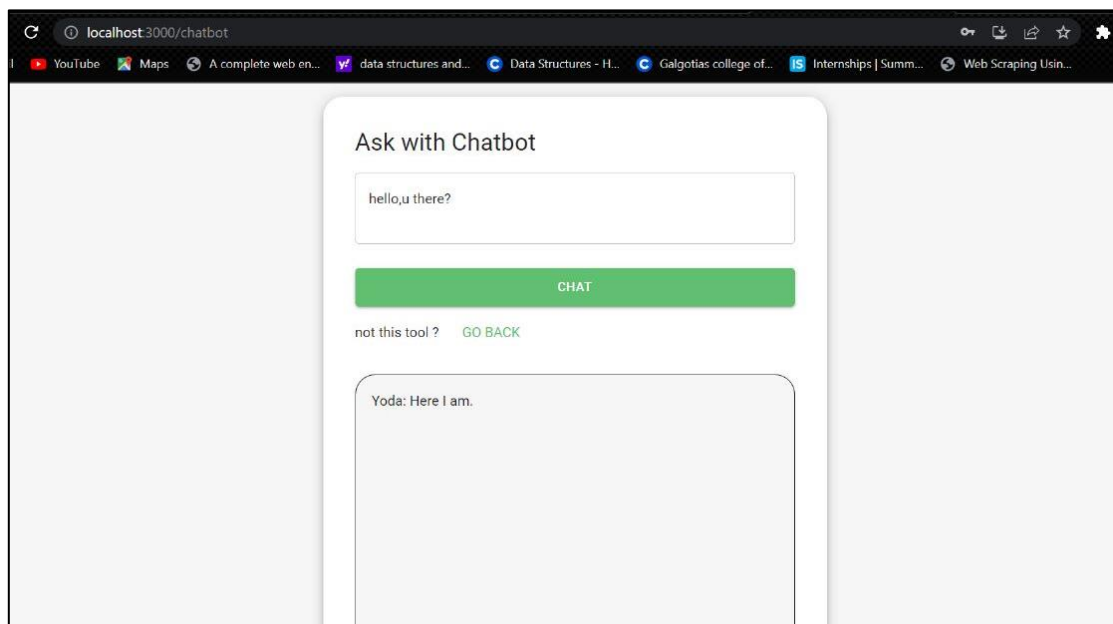

Fig 10 Menu Page

It shows various options that the website can perform. The menus are Text Summary, Paragraph Generator, ChatBot, JS Converter, Sci-fi Images.
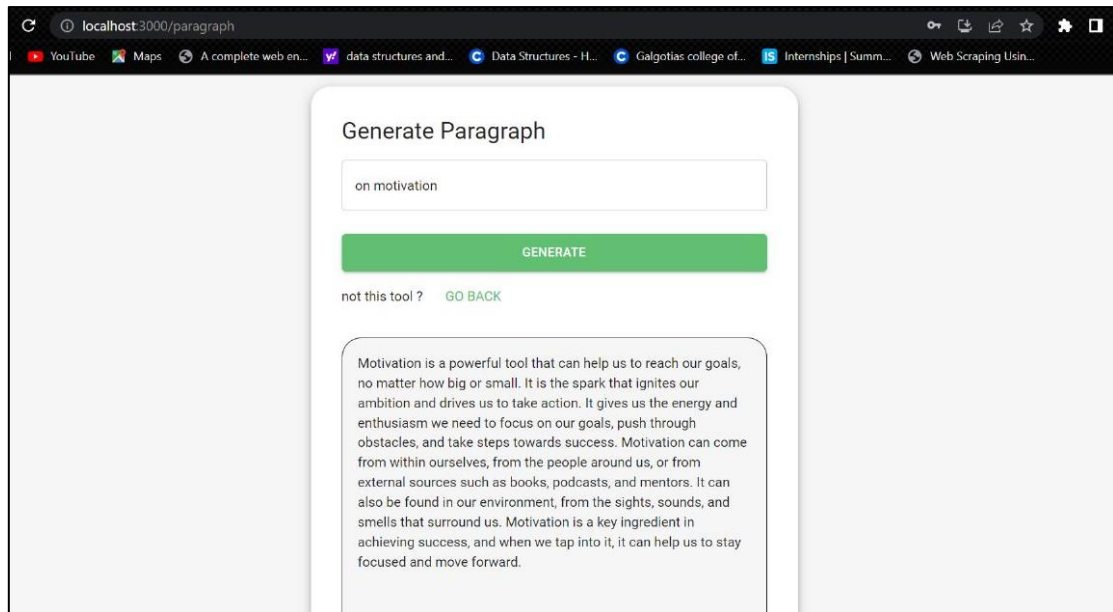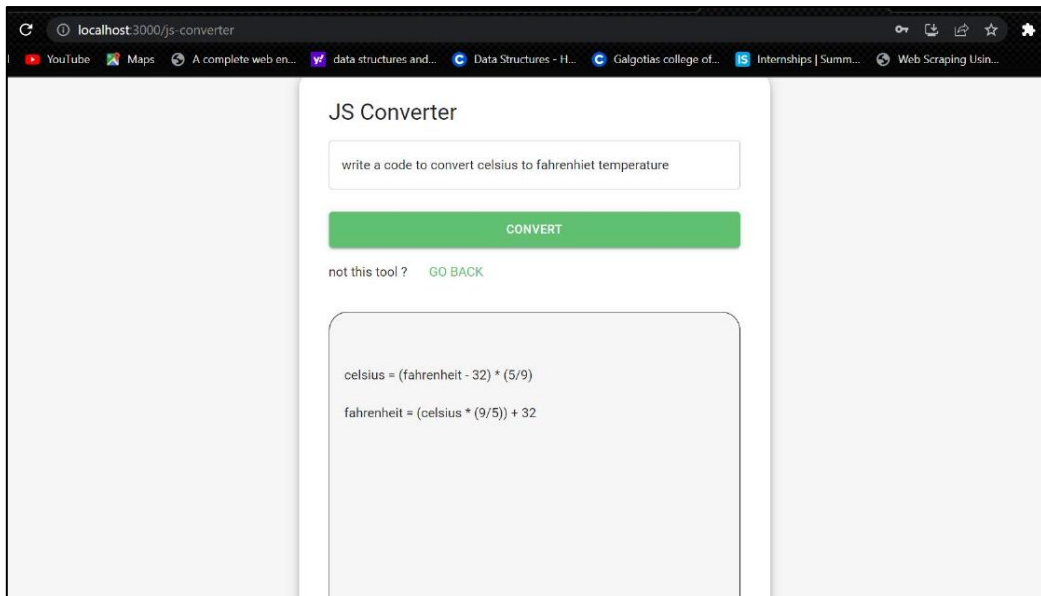
Fig 11 Text/Paragraph Generator

It is to generate a paragraph using some keywords. The user enters some keywords, then after that AI generates a suitable paragraph in according to that Keyword.


Fig 12 Chat Bot

It is bot to interact with us. It is like a normal chatbot with which we can chat. It responds with most appropriate choices

Fig 13 JS Convertor

It is to get a JS code by writing the problem/question. The user writes a problem statement to which he wants to create a JS code. The AI GPT 3 will analyze the statement and will produce the most suitable code for the problem.
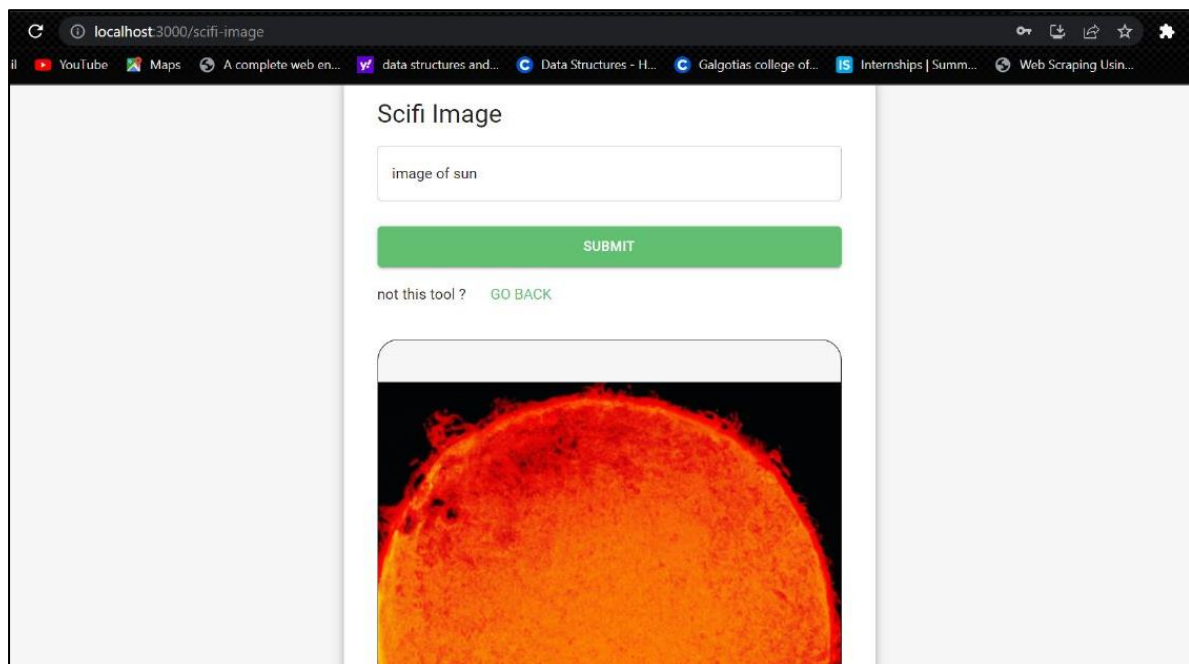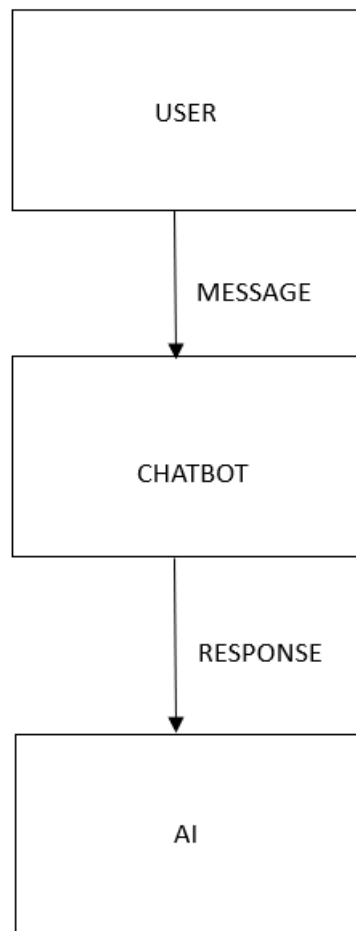

Fig 14 Image Generator

It is to generate an image using text. The user writes some keywords. The AI will search for the most suitable image and will display it to the user.

5.2 DATA FLOW DIAGRAM (DFD)

A data flow diagram is a graphical representation of data flow through an informed system. DFD scans are also used to visualize the structured design of data processing. In a DFD, data items flow from external data sources or internal data stores through internal processes to internal data or external data sinks.

5.2.1 LEVEL 0

```
┌─────────────────┐
│                 │
│      USER       │
│                 │
└────────┬────────┘
         │ MESSAGE
         ▼
┌─────────────────┐
│                 │
│    CHATBOT      │
│                 │
└────────┬────────┘
         │ RESPONSE
         ▼
┌─────────────────┐
│                 │
│       AI        │
│                 │
└─────────────────┘
```
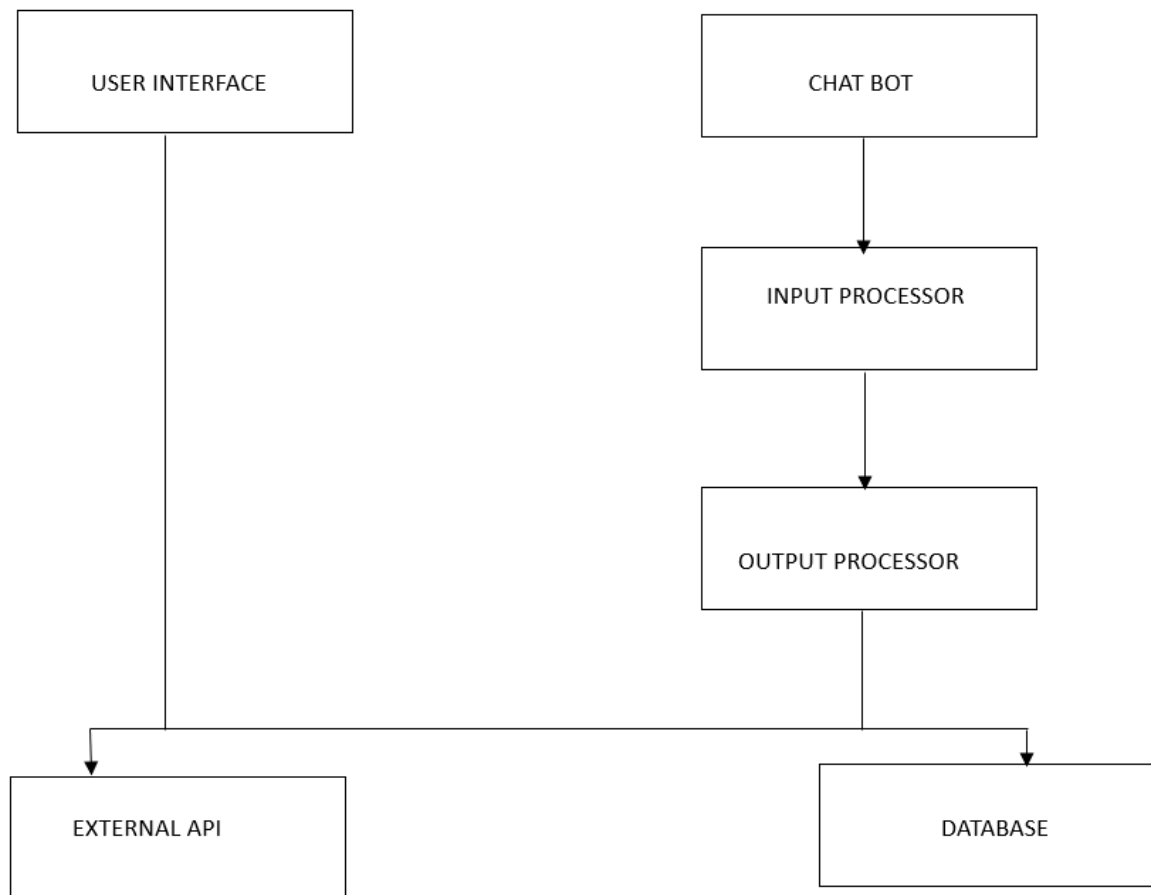
Fig 15 Level 0 data flow diagram shows how system is divided into sub system

5.2.2 LEVEL 1

A Level 1 DFD shows how the system is divided into subsystems. Each subsystem handles one or more data flows to and from external agents, providing all the functionality of the system as a whole.

Fig 16 Level 1 data flow diagram shows the sub process of first main process
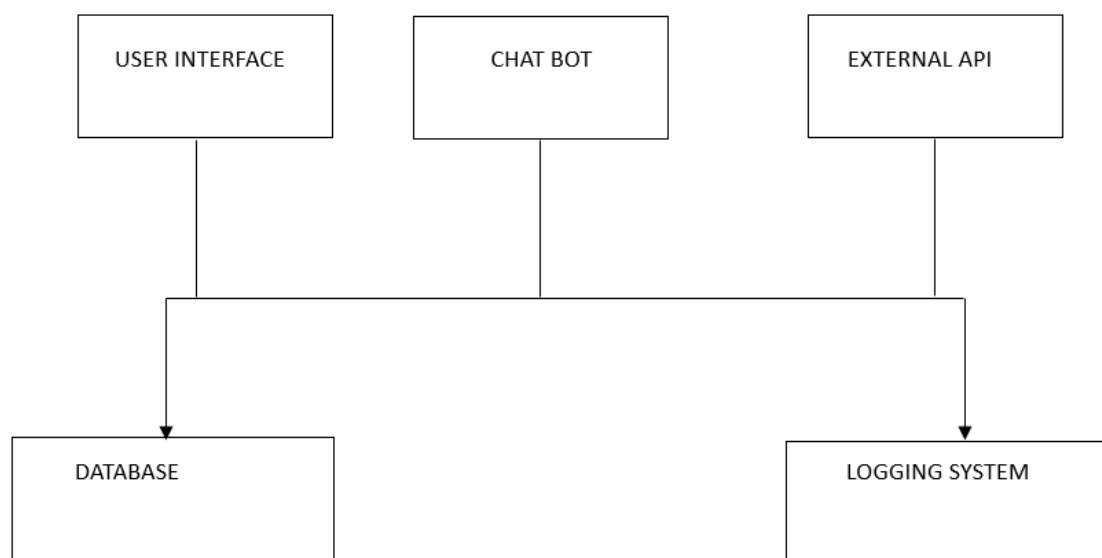
Fig 17 Level 1 data flow showing functionality of web pages working of the mode

5.3 FLOWCHART

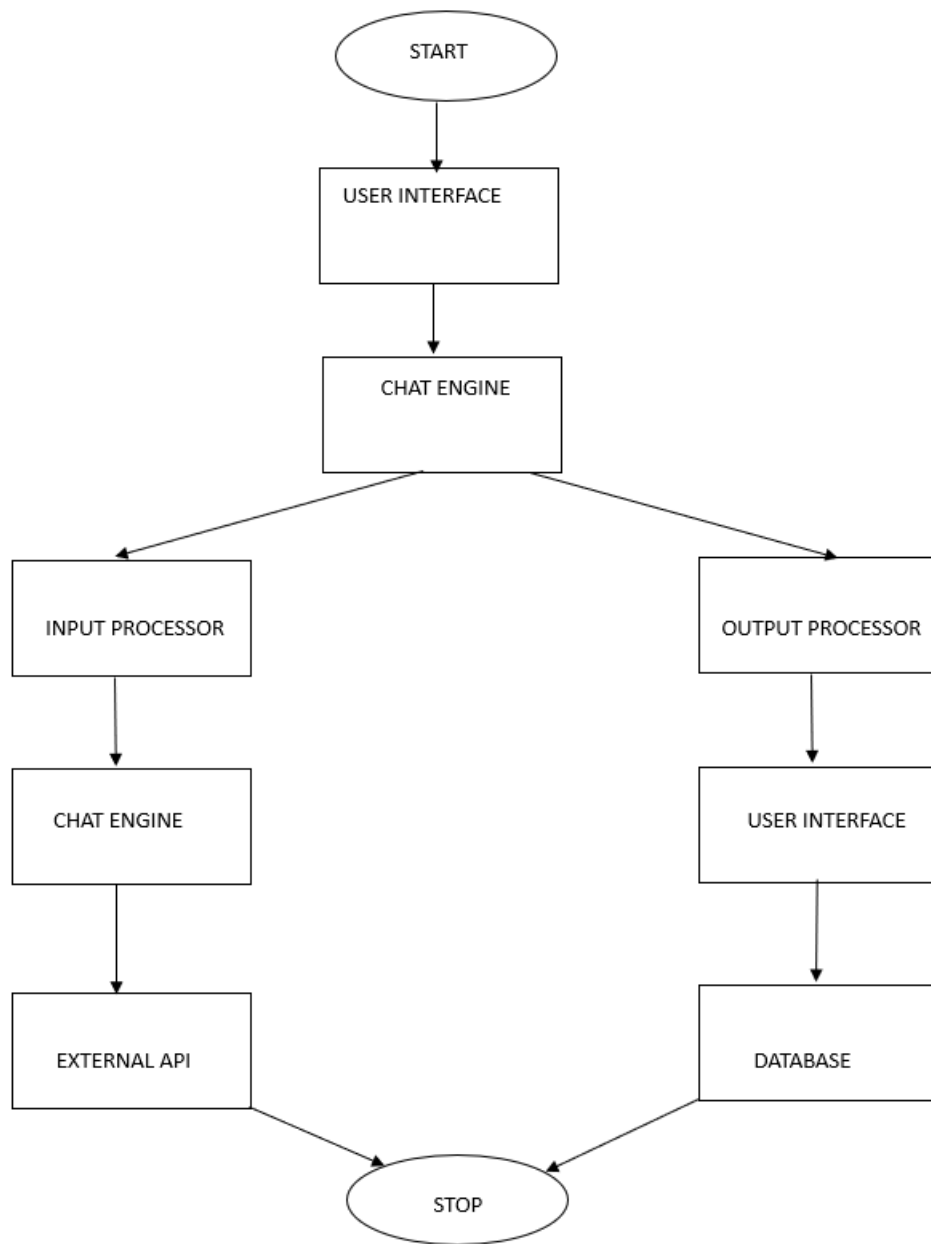**Fig 18 Flowchart**

<h1 style="text-align: center;">CHAPTER 6: PROJECT RELEVANCE</h1>

6.1 CONTRIBUTION OF PROJECT TO THE SOCIETY

The ChatGPT Clone using MERN can contribute to society in several ways:

- Enhanced customer support: The chatbot can provide instant responses to customer inquiries, thereby improving customer satisfaction and reducing the workload of human customer support agents.
- 24/7 availability: The chatbot can operate round the clock, providing users with immediate assistance even outside business hours.
- Improved accessibility: The chatbot can make communication more accessible to people with disabilities or language barriers, who may face difficulties while interacting with human customer support agents.
- Reduced costs: By automating customer support, businesses can save on the cost of hiring and training human customer support agents.
- Increased efficiency: The chatbot can handle multiple user queries simultaneously, providing faster and more efficient customer support.
- Valuable data insights: The chatbot can collect and analyze user data, providing businesses with insights into user preferences, behaviors, and pain points.
- Innovation in AI technology: The ChatGPT Clone using MERN demonstrates the innovative use of advanced AI technologies like the GPT-3 model, highlighting the potential of AI to improve various aspects of our daily lives.
- Potential for future developments: The project can serve as a basis for future developments in the field of chatbots and AI, paving the way for further innovations in customer support and communication technologies.

6.2. ETHICS AND CONCLUSION

When developing and deploying a ChatGPT Clone using MERN or any other AI-based technology, it is important to consider ethical implications. Here are some ethical considerations that should be taken into account:

- Bias and fairness: The chatbot's responses should be unbiased and not discriminate against individuals or groups based on factors such as race, gender, or ethnicity. Developers should ensure the chatbot is trained on diverse datasets and regularly evaluate its responses for fairness.

- ➢ Privacy and security: Users' personal data should be kept secure and private. Developers should implement appropriate security measures to prevent unauthorized access and use of user data.

- ➢ Transparency: Users should be informed when they are interacting with a chatbot and not a human agent. Developers should be transparent about the chatbot's capabilities and limitations, and clearly indicate when a human agent takes over the conversation.

- ➢ Accountability: Developers should take responsibility for the chatbot's behavior and any unintended consequences that may result from its use.

- ➢ Ethical use: The chatbot should be used for ethical purposes, and its development and deployment should comply with ethical standards and legal requirements.

- ➢ Human oversight: There should be human oversight to monitor the chatbot's behavior and ensure it is functioning correctly. Developers should have protocols in place to intervene when necessary.

- ➢ Informed consent: Users should be informed about the chatbot's data collection and usage policies and have the option to provide informed consent before using the chatbot.

- ➢ Limitations of technology: The chatbot should not be used to replace human interaction entirely. Developers should be aware of the limitations of the technology and ensure that human interaction is available when necessary.

- ➢ Avoiding harm: Developers should take steps to avoid harm to users, such as avoiding sensitive or triggering language and providing appropriate support when needed.

- ➢ Ethical decision-making: Developers should consider ethical implications at every stage of development, from design to deployment. Ethical decision-making frameworks such as the IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems can be useful resources in this regard.

By taking these ethical considerations into account, developers can ensure that the ChatGPT Clone using MERN is developed and deployed in an ethical and responsible manner that benefits society without causing harm.

CHAPTER 7: CONCLUSION AND FUTURE PROJECTIONS

7.1 CONCLUSION

The ChatGPT Clone using MERN has the potential to revolutionize the way users interact with websites and social media platforms by providing a more personalized and efficient experience. However, with this potential comes responsibility, and developers must consider the ethical implications of developing and deploying such a chatbot.

One of the key ethical considerations is bias and fairness. Developers must ensure that the chatbot's responses are unbiased and do not discriminate against individuals or groups based on factors such as race, gender, or ethnicity. This can be achieved by training the chatbot on diverse datasets and regularly evaluating its responses for fairness.

Another important ethical consideration is privacy and security. Users' personal data must be kept secure and private, and appropriate security measures must be implemented to prevent unauthorized access and use of user data. Developers must also be transparent about the chatbot's data collection and usage policies and provide users with the option to provide informed consent before using the chatbot.

Transparency is another key ethical consideration. Users must be informed when they are interacting with a chatbot and not a human agent. Developers must also be transparent about the chatbot's capabilities and limitations and clearly indicate when a human agent takes over the conversation.

In addition to these considerations, developers must also consider accountability, ethical use, human oversight, limitations of technology, avoiding harm, and ethical decision-making. By taking these considerations into account at every stage of development, developers can ensure that the ChatGPT Clone using MERN is developed and deployed in an ethical and responsible manner that benefits society without causing harm.

In conclusion, the ChatGPT Clone using MERN has the potential to improve user experiences and provide a more personalized interaction with users. However, it is important to balance these benefits with ethical considerations and ensure that the chatbot is developed and deployed in an ethical and responsible manner. By doing so, developers can create a chatbot that benefits society without causing harm.

7.2 FUTURE PROJECTIONS

The ChatGPT Clone using MERN is an exciting project with immense potential for the future. As technology continues to advance and more businesses and organizations turn to digital platforms for their operations, the demand for chatbots and other conversational AI technologies is likely to increase. Here are some future projections for the ChatGPT Clone using MERN:

➢ Increased demand for personalized customer interactions: As more businesses adopt digital platforms, the demand for personalized customer interactions is likely to increase. Chatbots like the ChatGPT Clone using MERN can provide a personalized and efficient way for businesses to interact with their customers, which can help improve customer satisfaction and loyalty.

➢ Advancements in natural language processing (NLP): As advancements in NLP continue, chatbots like the ChatGPT Clone using MERN will be able to understand and respond to users' queries more accurately and efficiently. This will improve the user experience and increase the chatbot's value to businesses.

➢ Integration with other technologies: Chatbots like the ChatGPT Clone using MERN can be integrated with other technologies such as voice assistants and augmented reality (AR) to provide a more immersive and engaging user experience. This integration can help businesses stand out from their competitors and increase customer engagement.

➢ Increased focus on ethics and responsible AI: As AI technologies become more prevalent, there will be an increased focus on ethics and responsible AI. Developers of chatbots like the ChatGPT Clone using MERN will need to ensure that their technologies are developed and deployed in an ethical and responsible manner.

➢ Expansion to new industries and use cases: The ChatGPT Clone using MERN can be applied to a variety of industries and use cases beyond customer support.

In conclusion, the future of the ChatGPT Clone using MERN looks promising, with increased demand for personalized customer interactions, advancements in NLP, integration with other technologies, increased focus on ethics and responsible AI, and expansion to new industries and use cases.

7.3 CONTRIBUTION OF PROJECT

The development of the "AI GPT 3 Clone" website was a collaborative effort between two team members (Shubham Srivastava and Unnati Agarwal) who worked together to create a functional and user-friendly website. Throughout the development process, we both contributed our unique skills and expertise to ensure that the application met the needs of our users.

We started by brainstorming the essential features of the website and identified the frontend and backend components that would be necessary for the website to function correctly. We both shared our ideas and experiences on the project and came up with a comprehensive plan on how to develop the app.

We then divided the tasks based on our strengths and interests, with one team member focusing on the frontend development using HTML, CSS, JS, React, while the other worked on the backend using MongoDB. We had regular check-ins to review each other's work and to ensure that everything was working together seamlessly.

During the frontend development phase, we worked on building the user interface, which included designing the screens, layouts, and user flows. We took into consideration the user experience and user interface design principles to ensure that the website was user-friendly, easy to navigate, and aesthetically pleasing.

In the backend development phase, we worked on creating the database schema, implementing authentication, and integrating the functionality. We utilized Mongo DB's real- time database and authentication services to store and retrieve data, and also to authenticate users, ensuring that only authorized personnel can access the app.

Throughout the development process, we helped each other when faced with challenges and offered suggestions to improve the app's overall functionality and user experience. We also performed extensive testing to identify and fix any bugs or issues.

REFERENCES

[1] *Liao, K., Weng, T., & Wang, J. (2021).* Designing chatbots: A mixed-method study of user expectations and needs. Journal of Business Research, 124, 612-624. doi: 10.1016/j.jbusres.2020

[2] *Chai, J. Y., Kim, M., & Kim, S. (2020).* Factors influencing the adoption of chatbots: A systematic review. Journal of Business Research, 122, 376-384. doi: 10.1016/j.jbusres.2020.06.002

[3] *Liao, K., Weng, T., & Huang, Y. (2020).* Exploring the antecedents of chatbot adoption: Perspectives from the gratification and the technology acceptance theories. Journal of Hospitality and Tourism Technology, 11(3), 468-483. doi: 10.1108/JHTT-07-2019-0099

[4] *Patidar, S. (2019).* Building web applications with MERN stack. Apress.

[5] *Huang, L., Xu, H., & Duan, W. (2019).* A survey on chatbot design techniques in speech conversation systems. Frontiers of Information Technology & Electronic Engineering, 20(11), 1473-1487. doi: 10.1631/FITEE.1901007

[6] *Kim, K. J., & Lee, H. G. (2019).* Chatbot acceptance model using an extended technology acceptance model. International Journal of Human-Computer Interaction, 35(9), 785-795. doi: 10.1080/10447318.2018.1482524

[7] *Deng, Z., Liu, Q., & Zhao, Y. (2019).* User acceptance of conversational agents in e-commerce. Journal of Business Research, 98, 365-376. doi: 10.1016/j.jbusres.2018.11.012

[8] *Dara, R. (2018).* MERN stack: A complete tutorial. Packt Publishing Ltd.

[9] *Bock, B., van Doorn, J., & Verhoef, P. C. (2018).* Customer engagement with chatbots: The influence of perceived risk. Journal of Service Research, 21(3), 312-327. doi: 10.1177/1094670517752125

[10] *Brown, M., & Parks, J. (2018).* Artificial intelligence and machine learning: Opportunities and challenges for human resource management. Human Resource Management Review, 28(4), 431-441. doi: 10.1016/j.hrmr.2017.11.002

[11] *Chen, Y., & Pu, X. (2018).* A study of user acceptance of chatbots in customer service. Journal of Service Theory and Practice, 28(4), 439-464. doi: 10.1108/JSTP-01-2018-0005

[12] *Davenport, T. H., & Ronanki, R. (2018).* Artificial intelligence for the real world: Don't start with moon shots. Harvard Business Review, 96(1), 108-116.

[13] *Kocielnik, R., & Arroyo, E. (2018).* Learning in chatbots. International Journal of Artificial Intelligence in Education, 28(1), 1-20. doi: 10.1007/s40593-017-0155-0

[14] *Godbole, N., & Srinivasan, S. (2017).* Customer service chatbots: Building blocks for designing intelligent interfaces. IEEE Intelligent Systems, 32(1), 50-55. doi: 10.1109/MIS.2017.6

[15] *Jha, A., & Bansal, R. (2016).* IoT based Smart Agriculture: A review. International Journal of Computer Applications, 146(11), 29-34.

[16] *Kaur, H., & Kaur, P. (2015).* Comparison of various clustering algorithms in data mining. International Journal of Computer Applications, 113(3), 28-32.

[17] *Lee, I., & Lee, K. (2015).* The Internet of Things (IoT): Applications, investments, and challenges for enterprises. Business Horizons, 58(4), 431-440.

[18] *Seebach, C., & Wagner, M. (2015).* The Internet of Things: An overview of key technologies, challenges and future prospects. Journal of Communications, 10(7), 439-445.

[19] *Zhu, Y., & Chen, X. (2015).* Internet of Things in industries: A survey. IEEE Transactions on Industrial Informatics, 10(4), 2233-2243.

[20] *Iyyanki, M., & Bhattacharya, S. (2014).* Artificial neural network application for the prediction of wind speed and power output: a case study of Southern India. International Journal of Renewable Energy Research, 4(4), 1049-1054.

[21] https://www.google.com/url?sa=i&url=https%3A%2F%2Fen.wikipedia.org%2Fwiki%2FFile%3AMongoDB_Logo.svg&psig=AOvVaw2iGLiGGtSsdXYPytfuoj2X&ust=1683923582511000&source=images&cd=vfe&ved=0CBEQjRxqFwoTCIj7vamO7v4CFQAAAAAdAAAAABAE. [seen on 12-04-23 9:30 pm].

[22] https://www.google.com/url?sa=i&url=https%3A%2F%2Fen.wikipedia.org%2Fwiki%2FNode.js&psig=AOvVaw2isGQNQhtsjm5ihKVy4rdt&ust=1683923625841000&source=images&cd=vfe&ved=0CBEQjRxqFwoTCOiikb6O7v4CFQAAAAAdAAAAABAJ. [seen on 13-04-23 6:30 pm].

[23] https://www.google.com/url?sa=i&url=https%3A%2F%2Fubidreams.fr%2Fen%2Fexpertises%2Fdevelopment%2Freact-js&psig=AOvVaw2Ow01Q37bElDiu-FoQSFk1&ust=1683923653873000&source=images&cd=vfe&ved=0CBEQjRxqFwoTCID4yMuO7v4CFQAAAAAdAAAAABAE. [seen on 14-04-23 6:30 pm].

[24] https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.edureka.co%2Fblog%2Fexpressjs-tutorial%2F&psig=AOvVaw2hAnpzFIv0nqX4dBs8Xc69&ust=1683923674964000

&source=images&cd=vfe&ved=0CBEQjRxqFwoTCKCrztWO7v4CFQAAAAdA

AAAABAE. [seen on 15-04-23 9:30 pm].

[25] https://www.google.com/url?sa=i&url=https%3A%2F%2Fp92.com%2Ftechdetails

%2Fhtml-css-and-

javascript&psig=AOvVaw1XfvX4r2gVCuwvu9e4RYbF&ust=1683923692743000

&source=images&cd=vfe&ved=0CBEQjRxqFwoTCIi3iN6O7v4CFQAAAAdAA

AAABAE. [seen on 27-04-23 9:00 pm].