

# Synth AI- Language Learning Model

MOHIT SHARMA<sup>1</sup>, ADITYA SHARMA<sup>2</sup>, ANSH KHAJURIA<sup>3</sup>, DR. BHAWNA SHARMA<sup>4</sup>, ER. SHEETAL GANDOTRA<sup>5</sup>

<sup>1, 2, 3</sup> Students, Dept. of Computer Engineering Government College of Engineering and technology, Jammu, J&K, India

<sup>4</sup> Professor, Dept. of Computer Engineering Government College of Engineering and Technology, Jammu, J&K, India

<sup>5</sup> Associate Professor, Dept. of Computer Engineering Government College of Engineering and Technology, Jammu, J&K, India.

*Abstract— This project report details the development of SynthAI, a smart chat tool made using the MERN stack (MongoDB, Express.js, React.js, and Node.js), designed to offer specialized features that address specific user needs and overcome the limitations of existing AI models like OpenAI's ChatGPT. The SynthAI web application integrates advanced natural language processing (NLP) and machine learning techniques to deliver a versatile platform with five key features: an advanced text summarizer, a context-aware paragraph generator, an enhanced AI chatbot, a robust JavaScript converter, and a high-quality sci-fi image creator. The advanced text summarizer condenses lengthy documents into concise, information-rich summaries, making it invaluable for research, legal, and academic users who need to quickly grasp essential content. The context-aware paragraph generator produces coherent and relevant paragraphs from use prompts, supporting creative writing, content creation, and educational purposes. The enhanced AI chatbot offers improved context management and sustained conversational coherence, suitable for customer support, tutoring, and interactive storytelling applications. The robust JavaScript converter accurately translates code snippets between various programming languages and JavaScript, as well as refactoring existing JavaScript code, providing developers and programmers with precise, optimized code solutions. The high-quality sci-fi image creator generates visually appealing and contextually accurate images from textual descriptions, catering to the creative needs of industries such as marketing, entertainment, and game development. The MERN stack was chosen for its flexibility and efficiency in developing full-stack applications. MongoDB offers a scalable and flexible database solution, Express.js and Node.js facilitate the creation of a robust backend infrastructure, and React.js ensures a dynamic and responsive user interface, enhancing the overall user experience. The report provides a comprehensive overview of the system's architecture, detailing how each feature is implemented and integrated within the MERN framework. It also addresses the*

*challenges encountered during development, such as maintaining conversational context, ensuring the accuracy of code translations, and generating high-quality images.*

## I. INTRODUCTION

SynthAI is a MERN Stack web application based on ChatGPT. It is a simple web application and yet very powerful and effective.

We have used React in the Frontend and MongoDB in the backend .

ChatGPT is a sibling model to InstructGPT, which is trained to follow an instruction in a prompt and provide a detailed response.

MERN stands for MongoDB, Express, React, Node, after the four key technologies that make up the stack.

- MongoDB — document database.
- Express(.js) — Node.js web framework .
- React(.js) — a client-side JavaScript framework
- Node(.js) — the premier JavaScript web server

Express and Node make up the middle (application) tier. Express.js is a server-side web Framework, and Node.js is the popular and powerful JavaScript server platform. Regardless of which variant you choose, MERN is the ideal approach to working with JavaScript and JSON all through the way.

Synth-AI consists of five segregated features which are-

- Text summarizer.
- Paragraph generator.
- ChatBot..

- Javascript Converter.
- Scifi Images.

All these features enhance the user experience by providing the benefit of using each feature separately and efficiently.

*A. Objectives*

1. **SEGREGATION OF FEATURES:** - The main aim of the Synth-ai app is to segregate the features of the CHAT-GPT. This will help the users touse the feature which they want to use separately.
2. **USER FRIENDLY INTERFACE:** - This app has a user friendly and intuitive interface that makes it easy for the users to register and login themselves on this app and perform suitable searches. This app is accessible to users of all skill levels, from beginners to advance level users.
3. **DATA SECURITY AND PRIVACY:** - This app prioritizes data security and privacy in order to protect the sensitive information of the users like their passwords, usernames, etc

**II. PROPOSED METHODOLOGY**

**TECH STACK(MERN STACK)**

The MERN stack is a popular full-stack JavaScript solution used for building dynamic web applications. It comprises four key technologies:

1. **MongoDB:** A NoSQL database that uses a flexible, JSON-like document structure, allowing for easy storage and retrieval of data. It provides scalability and performance, making it suitable for handling large amounts of unstructured data.
2. **Express.js:** A lightweight and flexible Node.js web application framework that provides a robust set of features for web and mobile applications. Express.js facilitates the development of server-side logic, handles HTTP requests and responses, and provides middleware to handle different aspects of the web application.
3. **React.js:** A front-end JavaScript library developed by Facebook for building user interfaces, particularly single-page applications. React allows developers to create reusable UI components, manage the view layer of the application, and efficiently update and render the UI based on data changes using the virtual DOM.

4. **Node.js:** A JavaScript runtime built on Chrome's V8 engine that allows the execution of JavaScript code on the server side. Node.js is known for its event-driven, non-blocking I/O model, which makes it lightweight and efficient, ideal for building scalable network applications.

*A. REGISTRATION OF THE USER*

Here the user will register himself by creating his own usrename and password and an entry of these credentials of the user will be made in the Mongo DB database. After the registration of the user, the credentials of the user(username , email and password) will be saved in the database. So, the user will not have to register again and again after logging in once he have registered.

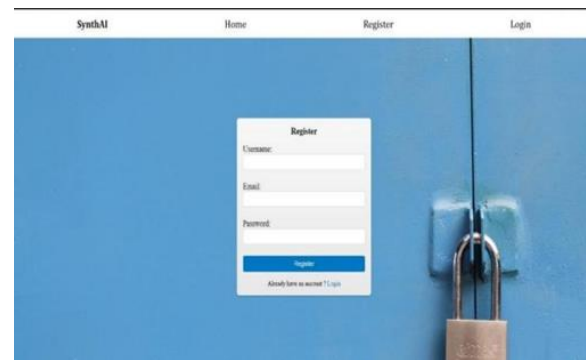


Fig. 1.1 Registration

*B. LOGIN OF THE USER*

Once the user had registered. the user will now have to login by entering his login credentials (username and password). If the email or password entered by the user are wrong, an error message will be generated.



Fig. 1.2 Login

**C. HOME PAGE**

The homepage of our web application will consist of five features :- text summarizer, chatbot , sci-fi images paragraph generator and js converter. The user will click on the feature which he wants to use.



**D. TEXT SUMMARIZER**

This feature will enable the user to summarize the text. The user will have to enter the text that the user wants to summarize in the text area and then the summarized text will be generated.



Fig. 1.3 Text Summarizer

**E. PARAGRAPH GENERATOR**

This feature will generate a paragraph on a given title to the model . The user will enter the tile for the paragraph to be generated in the text area and on clicking generate, model will generate the paragraph.



Fig. 1.4 Paragraph Generator

**F. CHATBOT**

This feature provides the user with a chatting bot with whom the user can interact.

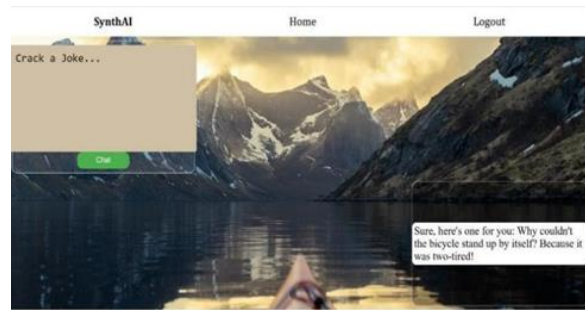


Fig. 1.5 ChatBot

**G. JAVASCRIPT CONVERTYER**

Using this feature the user can get javascript code for any problem statement. This javascript generator will generate javascript code of any complexity.



Fig. 1.6 JS Converter

**H. SCIFI IMAGES**

Sci-fi stands for scientific fictional . Using this feature the user can generate fiction type images that are not real images.

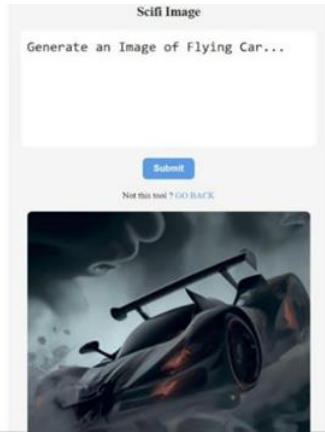
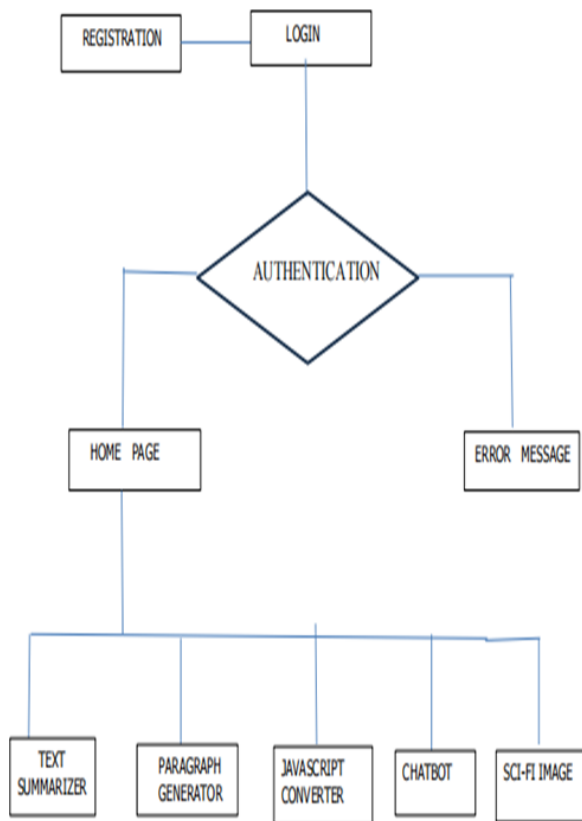


Fig. 1.7 Scifi Images



### III. CONCLUSION AND FUTURE SCOPE

In conclusion, ChatGPT is a powerful language model that can be used to create chatbots and conversational AI applications. Building a ChatGPT clone using HTML, CSS, and JavaScript can be a challenging but rewarding project for developers who are familiar with these front-end technologies. By following the step-by-step guide outlined in this report, you can create a simple chatbot interface that emulates the

functionality of ChatGPT, allowing users to interact with the chatbot and receive human-like responses.

ChatGPT's capabilities, based on the Transformer architecture and its large-scale training on diverse text data, make it a promising technology for enhancing user interactions with automated systems. However, it is crucial to use ChatGPT responsibly, ensuring that it adheres to ethical and legal standards. It is also important to consider user privacy, data security, and transparency in the design and implementation of ChatGPT-powered chatbots. As technology continues to advance, the field of conversational AI and chatbots is expected to grow rapidly. With the right implementation, a ChatGPT clone can offer a seamless and engaging conversational experience for users, opening up new possibilities for customer service, information retrieval, and other domains.

### FUTURE SCOPE

The future of AI is a topic of much speculation and excitement. There are many potential applications for AI, and it is likely to have a major impact on our lives in the years to come. Some of the potential applications of AI include:

- Automating tasks: AI can be used to automate many tasks that are currently done by humans, such as driving cars, diagnosing diseases, and writing legal documents. This could lead to significant productivity gains and cost savings.
- Personalizing experiences: AI can be used to personalize experiences for individuals, such as recommending products, suggesting content, and tailoring advertising. This could lead to a more engaging and relevant user experience.
- Making decisions: AI can be used to make decisions that are currently made by humans, such as allocating resources, setting prices, and hiring employees. This could lead to more efficient and effective decision making.
- Creating new products and services: AI can be used to create new products and services that are not possible with human intelligence alone. This could lead to new markets and opportunities.

## REFERENCES

- [1] MongoDB Official documentation: <https://www.mongodb.com/docs/>
- [2] NodeJS official documentation: <https://nodejs.org/docs/latest/api/>
- [3] ExpressJS official documentation: <https://expressjs.com/en/guide/routing.html>
- [4] ReactJS official documentation: <https://react.dev/learn>
- [5] OpenAI Official Documentation: <https://platform.openai.com/docs/overview>
- [6] VS Code Official documentation: <https://code.visualstudio.com/Docs>
- [7] Postman RESTful APIs Testing: <https://www.postman.com/api-documentation-tool/>
- [8] ChatGPT documentation : <https://openai.com/index/chatgpt/>
- [9] Typescript documentation: <https://www.typescriptlang.org/docs/>
- [10] ViteJS starter guide: <https://vitejs.dev/guide/>
- [11] JWT documentation: <https://jwt.io/introduction/>  
Environment Variables documentation: <https://www.dotenv.org/docs/>
- [12] Getting started with Git Version control: <https://git-scm.com/doc>
- [13] Mongoose Official documentation: <https://mongoosejs.com/docs/>
- [14] Understanding HTML, CSS and JavaScript: <https://www.w3schools.com/whatis/>
- [15] Understanding Node Package Manager : <https://docs.npmjs.com/>
- [16] “Beginning MERN Stack: Build and Deploy a Full Stack MongoDB, Express, React, Node.js App” by Greg Lim
- [17] Nodemon documentation: <https://www.npmjs.com/package/nodemon>
- [18] Body Parser documentation: <https://www.npmjs.com/package/body-parser>  
Bcrypt package documentation: <https://www.npmjs.com/package/bcrypt>