iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.81902

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

AI Content Writer & Summarizer (Mirage AI)

Sahil Patel, Ayan Baig, Soham Jadhav, Dr. P. Sivaram
Department of Computer Science and Engineering, Sandip University, Nashik, 422213, India

Abstract: ***The rapid evolution of large language models (LLMs) has enabled the development of intelligent applications capable of natural language understanding and generation. MirageAI is a modern AI-powered web application that provides multiple intelligent tools including chat, summarization, translation, and text improvement. The system is built using ReactJS, FastAPI, and PostgreSQL, and integrates Groq API for high-speed AI responses. The application ensures secure authentication, scalable architecture, and real-time interaction, offering a unified platform for AI-driven productivity.***
MirageAi is a Ai powered Chatbot with some additional features like Summary, Translation, Prompt improvement all in one platform. The main objective of MirageAi is to provide these features in a single platform for access which makes it more productive and efficient. Unlike conventional tools that offer isolated features, MirageAI focuses on unifying these capabilities to improve accessibility and workflow efficiency.

I. INTRODUCTION

MirageAI is a web-based platform designed to integrate multiple AI functionalities into a single interface. It enables users to interact with AI, summarize content, improve text, and translate languages efficiently. It has a interactive and clean UI/UX experience and follows modern design patterns with fast and efficient processing.Its architecture is optimized for speed and reliability, allowing for smooth real-time processing of user requests. By combining performance-focused backend design with a streamlined frontend interface, MirageAI delivers a balanced system that is both functionally rich and easy to use.

II. LITERATURE REVIEW

Modern AI systems such as GPT and LLaMA have enabled intelligent applications. Web technologies like ReactJS and FastAPI improve performance and scalability.

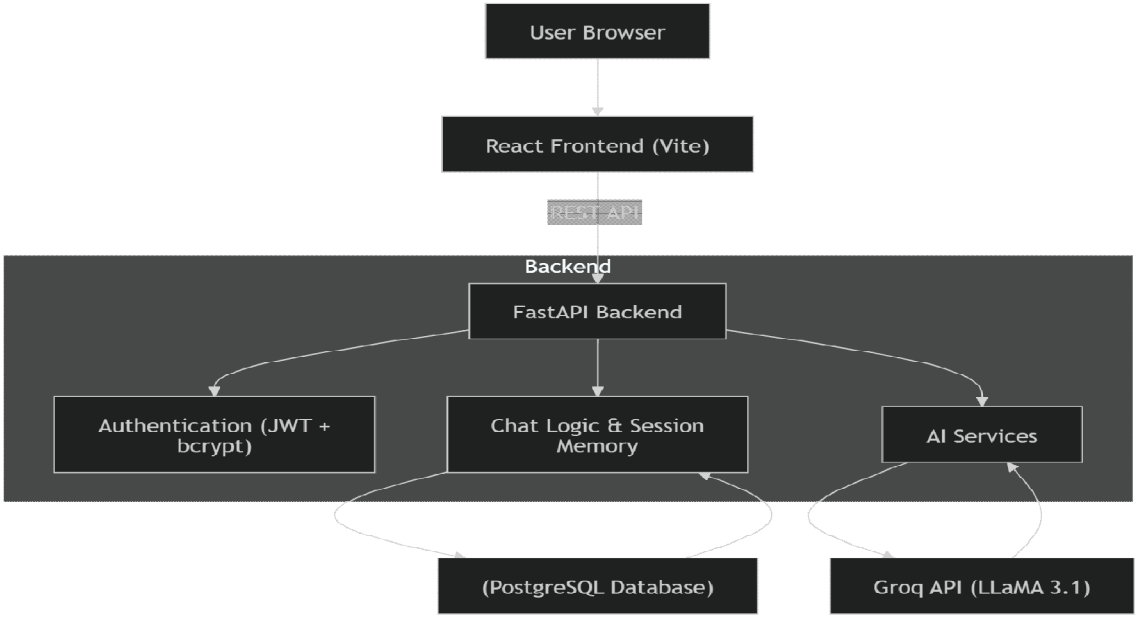
Recent advancements in AI-driven applications have focused on improving natural language understanding and system scalability. LLM-based systems have demonstrated strong performance in conversational tasks and text generation. Frameworks such as FastAPI and React have become widely adopted for building scalable and responsive applications.

Studies show that persistent chat memory improves user engagement by maintaining conversational context. Additionally, secure authentication mechanisms such as JWT and bcrypt hashing are essential for protecting user data in modern web applications.

Based on these advancements, MirageAI combines AI capabilities with a robust full-stack architecture to deliver a secure and efficient system.

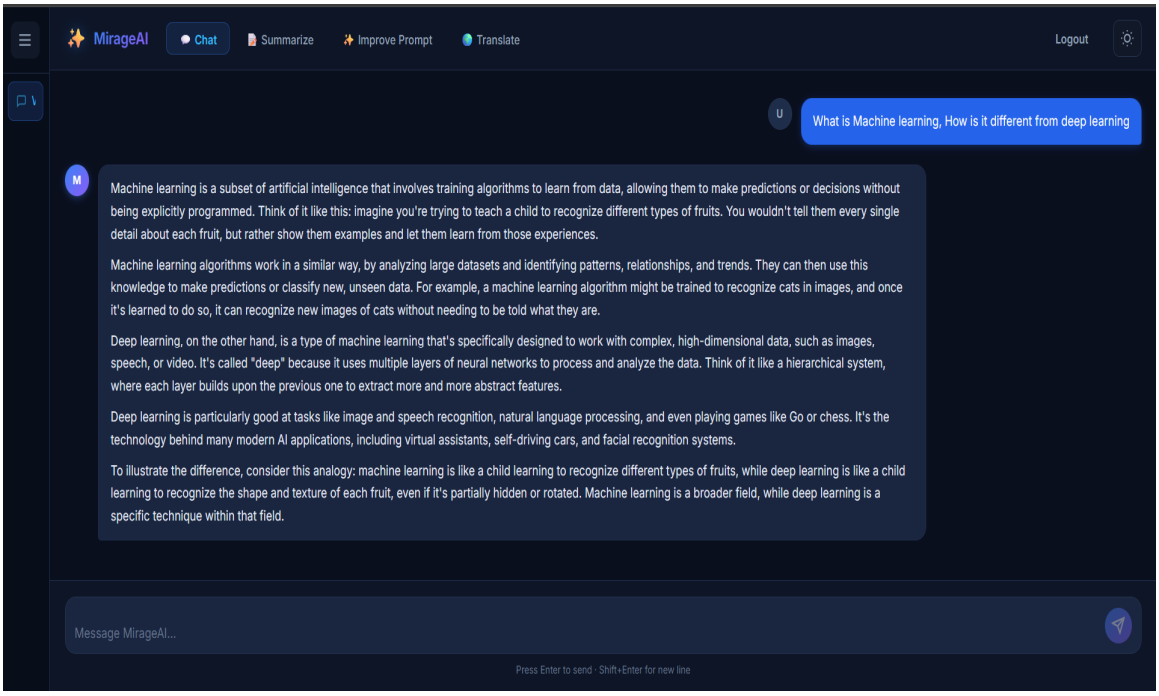
III. METHODOLOGY

MirageAI follows a layered architecture consisting of frontend, backend, and data layers. The system is built using a frontend (React), backend (FastAPI), and database (PostgreSQL). AI processing is handled via Groq API. Frontend Layer: Developed using ReactJS with Vite, the frontend provides a dynamic and responsive user interface. It manages routing, state handling, and API communication.Backend Layer: Built using FastAPI, the backend handles API requests, authentication, and AI processing. It uses asynchronous operations to ensure high performance.Data Layer: PostgreSQL is used for storing user data, chat sessions, and messages. SQLAlchemy ORM ensures efficient database operations.AI Integration: The system uses Groq API with LLaMA 3.1 to generate responses. Different temperature settings are applied based on task type.Authentication: JWT-based authentication ensures secure access, while bcrypt hashing protects user credentials.System Workflow: The user interacts with the frontend, which sends requests to the backend. The backend validates authentication, processes the request using AI services, stores relevant data, and returns the response.

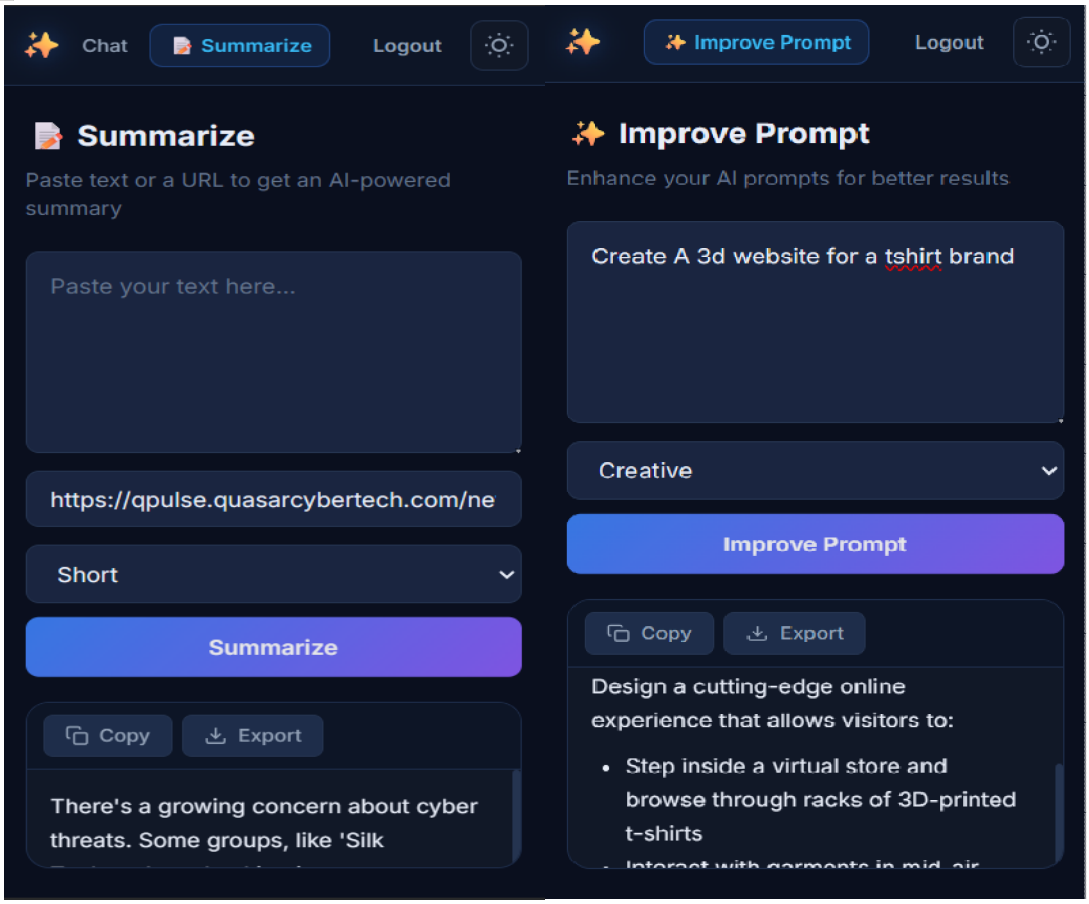


IV. EXPERIMENTAL RESULTS AND ANALYSIS

The MirageAI system was evaluated across key parameters including performance, responsiveness, usability, and response accuracy. The results indicate that the application delivers consistently fast response times, primarily due to FastAPI's asynchronous processing capabilities combined with optimized AI inference through the Groq API. This ensures that user interactions occur with minimal delay, maintaining real-time responsiveness. The React-based frontend contributes to a smooth and seamless user experience, enabling intuitive navigation and consistent behavior across different devices and screen sizes. Additionally, the system demonstrates strong contextual accuracy in AI responses, achieved through session-based memory storage that preserves conversation history and enhances the relevance of outputs. From a scalability perspective, the use of PostgreSQL along with a modular backend architecture allows the system to efficiently handle multiple users and concurrent requests without significant performance degradation.



ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com



Hardware Acceleration

Hardware acceleration refers to the use of a device's GPU (Graphics Processing Unit) instead of only the CPU to improve performance and rendering speed. Hardware acceleration improves performance by utilizing GPU rendering in browsers and optimized backend processing.In MirageAi, hardware acceleration helps enhance the smoothness of animations, transitions, and scrolling effects.Since the application is built using ReactJS, modern browsers automatically utilize hardware acceleration to render graphical elements efficiently. CSS properties such as transforms, transitions, and animations benefit from GPU acceleration, resulting in faster page rendering and improved user experience.By leveraging browser-based hardware acceleration, the application ensures smooth performance, reduced lag, and better responsiveness across different devices, especially on mobile platforms.

V. DISCUSSION

The development of MirageAI demonstrates how modern web technologies and large language models can be effectively combined to build a scalable and user-centric AI application. By integrating a ReactJS frontend with a FastAPI backend and leveraging the Groq API for AI processing, the system delivers real-time, context-aware responses within a clean and responsive interface.

One of the key strengths of MirageAI lies in its unified approach, where multiple AI functionalities—such as conversational interaction, summarization, translation, and text improvement—are consolidated into a single platform. This integration enhances usability and reduces the need for switching between different tools, thereby improving overall productivity. The use of session-based memory further strengthens the conversational experience by maintaining context across interactions, resulting in more relevant and coherent responses.

Additionally, the system emphasizes simplicity and intuitive design, ensuring that users can easily navigate through features without unnecessary complexity. The responsive UI, combined with efficient backend processing, contributes to a smooth and consistent user experience across various devices.

Overall, MirageAI highlights the importance of integrating scalable architecture, secure authentication, and efficient AI services to develop modern intelligent applications. It serves as a strong example of how full-stack development and AI integration can be combined to create practical, high-performance solutions for real-world use cases.

VI. CONCLUSION

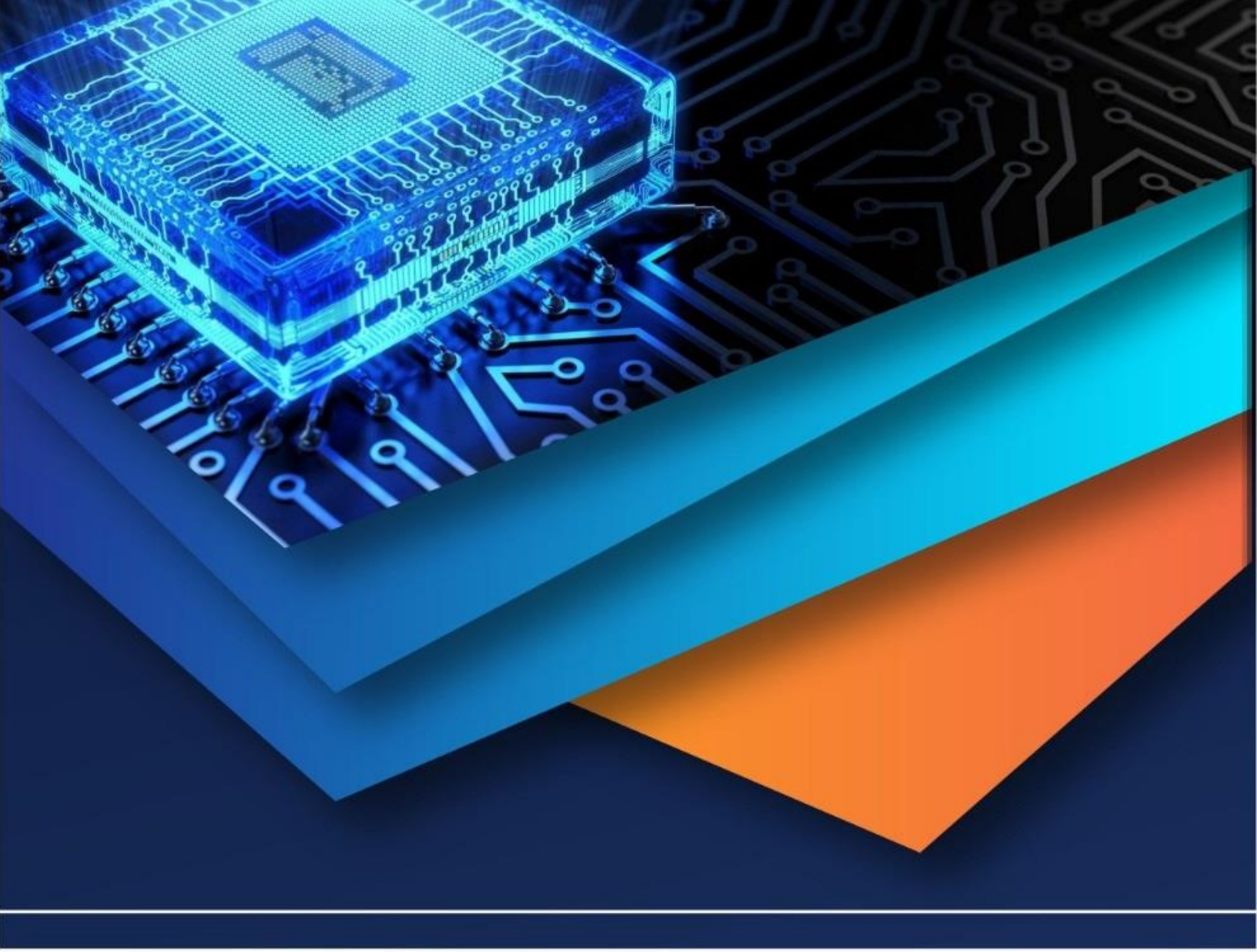
MirageAI integrates AI tools into a scalable platform.MirageAI application successfully provides a fast, reliable, and user-friendly platform for accessing multiple AI-powered functionalities within a single interface. By utilizing ReactJS, FastAPI, and PostgreSQL, the system ensures responsive design, efficient data handling, and smooth real-time performance. The integration of the Groq API with LLaMA 3.1 enables accurate and context-aware responses for tasks such as conversational interaction, summarization, translation, and text improvement.

The project achieves its primary objective of consolidating essential AI tools into a unified and accessible platform, reducing the need for multiple applications and enhancing overall productivity. The inclusion of session-based memory and secure authentication further strengthens the system's usability and reliability.

Overall, MirageAI demonstrates how modern web technologies combined with advanced AI models can be effectively used to develop scalable, efficient, and intelligent applications tailored to the needs of today's digital users.

REFERENCES

[1] React Official Documentation. Available at: https://react.dev
[2] FastAPI Official Documentation. Available at:https://fastapi.tiangolo.com/
[3] PostgreSQL Documentation. Available at:https://www.postgresql.org/docs/
[4] Groq API Documentation. Available at:https://console.groq.com/docs
[5] LLaMA Model Overview (Meta AI). Available at:https://ai.meta.com/llama/
[6] SQLAlchemy ORM Documentation. Available at:https://docs.sqlalchemy.org/
[7] PyJWT Documentation (JWT Authentication). Available at:https://pyjwt.readthedocs.io/
[8] Mozilla Developer Network (MDN Web Docs). HTML, CSS and JavaScript Documentation. Available at:https://developer.mozilla.org
[9] Vite Official Documentation. Available at:https://vitejs.dev/guide/
[10] Modern Web Application Architecture – REST APIs Overview. Available at:https://restfulapi.net



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY