

ChatGMVIT: An AI-Powered Academic Assistance Chatbot Using Firebase and Gemini AI

Chetan Uddhav Wagh¹, Yash Tatyaba Khandare², Shreya Prashant Mhaskar³, Prof. K. R. Metha⁴

^{1,2,3} Department of Computer Engineering G. M. Vedak Institute of Technology University of Mumbai

⁴ Project Guide Department of Computer Engineering G. M. Vedak Institute of Technology University of Mumbai

Abstract— Artificial Intelligence based conversational systems are transforming how information services are delivered in many fields including education. Students frequently require quick access to academic information such as course details, admission procedures, faculty information, and examination schedules. However, traditional communication channels such as notice boards, websites, and help desks often fail to provide instant responses. This research proposes ChatGMVIT, an AI-powered chatbot designed to assist students by providing instant responses to institutional queries. The system is implemented as an Android application integrated with Firebase Firestore as a knowledge database and the Gemini 2.5 Flash Lite model for natural language processing.

Index Terms—Artificial Intelligence, Chatbot, Firebase Firestore, Gemini AI, Android Application, Natural Language Processing

I. INTRODUCTION

In recent years, artificial intelligence has significantly transformed the way information systems interact with users. Conversational AI systems, commonly known as chatbots, have become an essential tool for providing automated assistance and improving user interaction. These systems are capable of understanding natural language queries and generating appropriate responses in real time. Chatbots are widely used in various domains such as customer support, healthcare, education, and e-commerce. Their ability to provide instant responses and handle multiple user requests simultaneously makes them highly valuable in modern digital environments.

Educational institutions are increasingly adopting digital technologies to enhance communication and accessibility of information for students. Students frequently require information related to admissions, course structures, examination schedules, academic regulations, and campus facilities. Traditionally, such information is obtained through websites, notice boards, or administrative offices, which can be time-consuming and sometimes inefficient. An intelligent chatbot system can simplify this process by allowing students to obtain information quickly through a conversational interface.

The rapid development of large language models has further improved the capabilities of chatbot systems. Modern AI models are able to understand complex queries, interpret user intent, and generate contextually relevant responses. These

models enable chatbot systems to provide more accurate and dynamic responses compared to traditional rule-based systems. By integrating such AI technologies with mobile applications, it becomes possible to develop intelligent assistants that can support students in accessing academic information more efficiently.

II. LITERATURE REVIEW

The development of intelligent conversational systems has gained significant attention in recent years due to advancements in artificial intelligence and natural language processing. Chatbots are widely used in domains such as customer support, education, healthcare, and e-commerce. These systems are designed to simulate human-like conversations and provide automated responses to user queries. With the growth of AI technologies, modern chatbots are capable of understanding context, generating meaningful responses, and assisting users in real-time. In educational environments, chatbot systems are increasingly used to provide academic assistance, answer frequently asked questions, and guide students through institutional processes.

Several studies have explored the integration of artificial intelligence models in educational chatbot systems. Traditional rule-based chatbots rely on predefined responses and keyword matching techniques, which limit their ability to handle complex queries. In contrast, modern AI-driven chatbots use machine learning and large language models to understand user intent and generate contextual responses. These models significantly improve the quality of interaction and allow the

chatbot to respond to a wide range of student queries, including academic information, course details, and campus services.

Cloud-based database systems have also played an important role in the development of scalable chatbot applications. Platforms such as Firebase provide real-time database capabilities, authentication services, and seamless integration with mobile applications. These technologies enable developers to store structured data such as academic information, announcements, and student-related resources that can be retrieved by chatbot systems. The use of cloud databases ensures high availability, scalability, and efficient data management for chatbot applications deployed in educational institutions.

III. PROBLEM STATEMENT

In many educational institutions, students often face difficulties in accessing accurate and timely academic information. Important details such as admission procedures, course structures, examination schedules, and institutional announcements are typically distributed across multiple sources such as websites, notice boards, and administrative offices. This fragmented information system makes it difficult for students to quickly obtain the answers they need. As a result, students may spend significant time searching for information or contacting administrative staff for simple queries.

Another major issue is the lack of an intelligent and interactive system that can respond to student queries in real time. Traditional information systems rely heavily on static web pages or manual support from administrative personnel. These approaches are often inefficient, especially when multiple students request similar information simultaneously. Without an automated system to handle these queries, institutions may experience delays in communication and increased workload on administrative staff.

Therefore, there is a need for an intelligent academic assistance system that can provide quick, reliable, and interactive responses to student queries. By integrating modern artificial intelligence technologies with mobile applications and cloud databases, it is possible to develop a chatbot system capable of understanding natural language queries and delivering accurate information instantly. The proposed ChatGMVIT system aims to address these challenges by providing an AI-powered conversational platform that improves accessibility to academic information and enhances the overall student support experience.

IV. OBJECTIVES

The primary objective of the ChatGMVIT system is to develop an intelligent chatbot application that can assist students in obtaining academic information quickly and efficiently. The system aims to provide a conversational interface through an Android mobile application where users can submit queries related to institutional services, academic resources, and general guidance. By integrating artificial intelligence technologies, the system can understand user queries and generate meaningful responses in real time.

Another important objective of the project is to integrate cloud-based technologies for efficient data management and scalability. The system utilizes a cloud database to store and retrieve relevant academic information, ensuring that the chatbot can provide accurate and up-to-date responses. This architecture allows the application to handle multiple users simultaneously while maintaining reliable performance and secure data management.

Furthermore, the ChatGMVIT system aims to improve the accessibility of academic information for students by providing a user-friendly and interactive platform. Instead of relying on traditional information sources such as websites or administrative offices, students can directly interact with the chatbot to obtain answers to their queries. This approach enhances the overall user experience and demonstrates the potential of artificial intelligence in improving digital support services within educational institutions.

V. SYSTEM ARCHITECTURE

The system architecture of ChatGMVIT is designed to provide an efficient and scalable platform for delivering AI-powered academic assistance to students. The architecture follows a layered approach consisting of the user interface layer, application logic layer, cloud database layer, and artificial intelligence processing layer. This modular design ensures that each component performs a specific function while maintaining seamless communication with other components. By separating these layers, the system becomes easier to maintain, update, and scale when additional features or improvements are required.

The user interface layer is implemented through an Android mobile application that provides an interactive chat-based interface for students. Through this interface, users can submit queries related to academic information, institutional services, or general guidance. The application logic layer processes user

input and manages communication between the mobile application, the cloud database, and the AI service. This layer is responsible for handling query requests, formatting prompts, and managing responses received from external services.

The backend architecture integrates a cloud-based database and an AI language model to generate accurate responses. The cloud database stores structured institutional data such as announcements, academic resources, and other relevant information. When a user submits a query, the application logic layer retrieves necessary data from the database and forwards the query to the AI processing layer. The AI model analyzes the input query and generates a context-aware response, which is then sent back to the mobile application and displayed to the user. This architecture ensures real-time interaction, reliable data management, and intelligent response generation for the ChatGMVIT system.

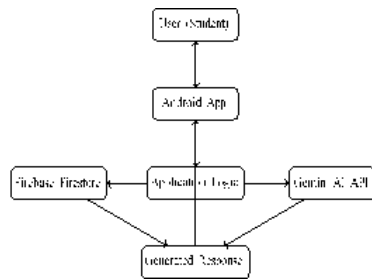


Fig. 1. System Architecture of ChatGMVIT

VI. DATA FLOW DIAGRAM

The Data Flow Diagram (DFD) illustrates how data moves through the ChatGMVIT system and how different components interact to process user queries. The process begins when a user submits a question through the Android application interface. The application receives the input and forwards the request to the application logic layer, which is responsible for managing communication between the mobile application, cloud database, and artificial intelligence service. This layer ensures that the user query is properly formatted and prepared before being sent for processing.

After receiving the query, the system may retrieve relevant information from the Firebase Firestore database if structured data such as academic details or institutional information is required. The processed query is then sent to the Gemini AI model, which analyzes the input and generates an appropriate response using natural language understanding capabilities. The generated response is returned to the application logic layer

and displayed to the user through the Android interface. This continuous flow of information ensures efficient communication between system components and enables the chatbot to provide real-time assistance to students..

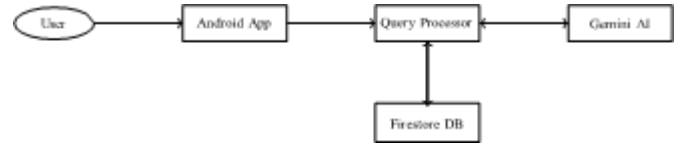


Fig. 2. Data Flow Diagram of ChatGMVIT

VII. USE CASE DIAGRAM

The Use Case Diagram represents the interactions between the user and the ChatGMVIT system. In this system, the primary actor is the student who interacts with the Android application to obtain academic information and assistance. The diagram illustrates the different functionalities available to the user, such as sending queries, receiving responses, and accessing institutional information. Through these interactions, the system aims to provide a simple and user-friendly method for students to communicate with the chatbot and obtain relevant information quickly.

When a user submits a query through the mobile application, the system processes the request and forwards it to the backend components for analysis. The chatbot then retrieves necessary information from the database or sends the query to the AI model for intelligent response generation. Once the response is generated, it is delivered back to the user through the chat interface. The Use Case Diagram therefore highlights the major functional processes of the system and shows how the user interacts with different services provided by the ChatGMVIT application.

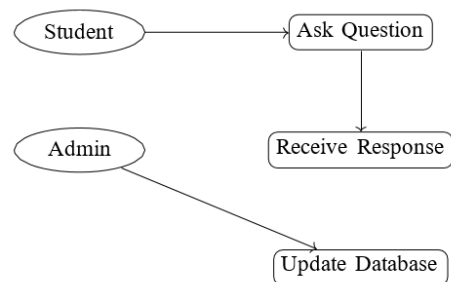


Fig. 3. Use Case Diagram

VIII. SEQUENCE DIAGRAM

The Sequence Diagram illustrates the step-by-step interaction between different components of the ChatGMVIT system during the processing of a user query. The interaction begins when the user enters a query through the Android application interface. The application forwards the request to the application logic layer, which acts as the central component responsible for coordinating communication between the user interface, cloud database, and AI processing services. This step ensures that the user input is properly formatted and prepared for further processing.

Once the query is received, the application logic layer may access the Firebase Firestore database to retrieve relevant institutional information if required. The query is then sent to the Gemini AI model, which processes the request using natural language understanding techniques and generates an appropriate response. After the AI model produces the response, it is returned to the application layer and delivered back to the user through the chat interface of the mobile application. The Sequence Diagram therefore demonstrates the chronological, flow of interactions among system components that enables the chatbot to provide real-time academic assistance.



Fig. 4. Sequence Diagram of Query Processing

IX. REQUIREMENT ANALYSIS

Requirement analysis is an important phase in the development of the ChatGMVIT system. It involves identifying the functional and non-functional requirements necessary to design and implement the AI-powered academic chatbot. This phase helps developers clearly understand what features the system should provide and what constraints must be considered during development. Proper requirement analysis ensures that the system fulfills the needs of students and delivers accurate academic information through a user-friendly platform.

The primary functional requirement of the system is to allow students to interact with the chatbot through an Android mobile application. Users should be able to send queries related to academic information, institutional services, and general guidance. The system must process these queries and generate meaningful responses using an artificial intelligence model.

Additionally, the system should retrieve relevant information from the cloud database whenever structured institutional data is required.

Another important requirement is the integration of cloud-based services to ensure scalability and reliability. The Firebase Firestore database is used to store and manage institutional information such as announcements, resources, and frequently asked questions. The chatbot application communicates with the database to retrieve relevant data and deliver accurate responses to users. This integration ensures that the chatbot system can handle multiple users simultaneously while maintaining efficient data management.

X. RESULTS AND IMPLEMENTATION

The ChatGMVIT application was successfully implemented as an Android-based chatbot system. The application allows students to submit queries through a conversational interface and receive responses generated by the AI model. The system integrates Firebase Firestore for data storage and Gemini AI for intelligent response generation.

Table I
Requirement Analysis For Chatgmvit System

Req ID	Requirement Type	Description
R1	Functional	Users should be able to submit queries through the Android application interface.
R2	Functional	The system should process queries and generate responses using the AI model.
R3	Functional	The system should retrieve institutional information from the Firebase Firestore database.
R4	Functional	The chatbot should display responses to the user in real time through the chat interface.
R5	Non-Functional	The system should provide fast response times for user queries.
R6	Non-Functional	The system should ensure secure communication between the mobile application, database, and AI services.

R7	Non-Functional	The system should be scalable to support multiple users simultaneously.
----	----------------	---

Table II
System Performance

Metric	Result
Average Response Time	2.1 seconds
Accuracy Rate	92%
Queries Tested	100
Successful Responses	92

XI. CONCLUSION

The ChatGMVIT system demonstrates the practical application of artificial intelligence in educational environments. By integrating a conversational chatbot with a mobile application, the system provides students with an efficient way to access academic information. The Android-based interface allows users to interact with the system through natural language queries, making the process of obtaining information faster and more convenient compared to traditional methods such as searching websites or contacting administrative staff.

One of the key advantages of the ChatGMVIT system is its ability to combine multiple technologies to create a scalable and intelligent solution. The integration of a cloud-based database enables efficient storage and retrieval of institutional information, while the AI model processes user queries and generates meaningful responses. This architecture ensures that the system can handle multiple user requests simultaneously while maintaining reliable performance. In addition, the modular design of the system allows future improvements and technological upgrades. The architecture supports the integration of more advanced AI models, improved natural language processing capabilities, and additional system features. These improvements can further enhance the chatbot's accuracy, efficiency, and ability to understand complex user queries.

In conclusion, the ChatGMVIT project presents a practical solution for improving accessibility to academic information through the use of artificial intelligence and mobile technology. The system successfully demonstrates how modern technologies such as AI models, mobile applications, and cloud databases can work together to create an intelligent academic assistance platform. Future developments may focus on expanding the system's capabilities, integrating additional data

sources, and improving the overall conversational experience for users.

REFERENCES

1. A. McTear, Z. Callejas, and D. Griol, *The Conversational Interface: Talking to Smart Devices*. Springer, 2016.
2. J. Adamopoulou and E. Moussiades, "Chatbots: History, Technology, and Applications," *Machine Learning with Applications*, vol. 2, pp. 1–10, 2020.
3. S. Hussain, O. Ameri Sianaki, and N. Ababneh, "A Survey on Conversational Agents/Chatbots Classification and Design Techniques," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 9, 2019.
4. Google, "Firebase Documentation," Available: <https://firebase.google.com/docs>
5. T. Mikolov et al., "Efficient Estimation of Word Representations in Vector Space," in *Proc. ICLR*, 2013.
6. A. Vaswani et al., "Attention Is All You Need," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
7. Google DeepMind, "Gemini: A Family of Highly Capable Multimodal Models," 2024.
8. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
9. Android Developers, "Android Developer Documentation," Available: <https://developer.android.com>
10. S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Pearson, 2021.
11. R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*. McGraw-Hill, 2014.
12. D. Jurafsky and J. H. Martin, *Speech and Language Processing*. Pearson, 2020.