IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

"SMART ACADEMIC CHATBOT FOR TEACHING DATABASES AND INFORMATION SYSTEMS USING LLMS"

[1] A. PRAKASHINI, [2] JANGAM SRAVAN , [3] MYLA VAGDEVI, [4] ENJAMURI THRILOK, [5] GADE HARIKRISHNA

[1]Assistant Professor, [2,3,4,5] UG STUDENT

[1,2,3,4,5]DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING(CSD)

[1,2,3,4,5]AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

Gunthapally(V), Abdullapurmet(M), R.R Dist - 501512

Abstract: Contribution: This study examines the advantages and obstacles associated with the development, implementation, and assessment of a large language model (LLM) chatbot, MoodleBot, within computer science educational environments. It shows how LLMs could be used in LMSs like Moodle to help with self-regulated learning (SRL) and help-seeking behaviour. Computer science teachers have a lot of trouble adding new tools to LMSs to make the learning environment more supportive and interesting. MoodleBot solves this problem by giving students and teachers a place to interact with each other. Questions for Research: This study examines two questions, notwithstanding challenges such as bias, hallucinations, and the reluctance of teachers and educators to adopt new AI technologies. (RQ1) How much do students think MoodleBot is a useful tool for helping them learn? (RQ2) How accurate are the answers that MoodleBot gives, and how well do they fit with the course content that has already been set? Methodology: This study examines pedagogical literature regarding AI-driven chatbots and employs the retrieval-augmented generation (RAG) methodology for the design and data processing of MoodleBot. The technology acceptance model (TAM) looks at how much users accept something by looking at things like how useful they think it is and how easy it is to use. Forty-six students took part, and thirty of them filled out the TAM questionnaire. Results: Chatbots that use LLM, like MoodleBot, can make teaching and learning a lot better. This study found that there was a high accuracy rate (88%) in helping with course-related tasks. Students' positive feedback shows that AI-powered educational tools work and can be used in real life. These results show that educational chatbots can be used in courses to make learning more personalised and make teachers' jobs easier, but automated fact-checking needs to get better.

Keywords— Artificial Intelligence, Large Language Models, Retrieval-Augmented Generation, Learning Management Systems, AI Chatbot, Personalized Learning, Higher Education.

I. INTRODUCTION

[1]Artificial Intelligence (AI) and Machine Learning (ML) have come a long way in a short amount of time, and they have changed many fields, including education. As students learn more difficult subjects like Databases and Information Systems in college, they need more personalised, interactive, and immediate help. Moodle and other traditional Learning Management Systems (LMSs) give a structured way to deliver course content, assignments, and tests.[3] But they often depend on fixed resources and the teacher's availability for questions, which makes it harder for students to learn on their own, get personalised feedback, and stay engaged. When students try to learn on their own, they may have trouble with the course material or need help understanding certain ideas outside of class. In the same way, teachers often have a lot of administrative work to do, like answering the same questions over and over, keeping an eye on students' progress, and giving each student personalised help. This can make teaching less effective overall.[2]

To solve these problems, this project suggests MoodleBot, an AI-powered chatbot that uses Large Language Models (LLMs) and a Retrieval-Augmented Generation (RAG) method to make it better. MoodleBot is made to work perfectly with LMS platforms, giving students real-time, personalized answers to their questions that take into account the context.[4] MoodleBot is different from regular rule-based chatbots or FAQ systems because it can handle complicated and unexpected questions by using both generative AI and curated course content. This lets students use the system in a natural way, get answers right away, and get help that is specific to their level of learning.[12] MoodleBot encourages self-regulated learning, critical thinking, and independence in students' education by giving them dynamic explanations, hints, follow-up questions, and links to related ideas.

From a teacher's point of view, MoodleBot cuts down on repetitive work and gives useful data on student questions, concepts that are often misunderstood, and overall engagement patterns. Teachers can use this information to change how they teach, find out what students don't know, and make the course materials better. The system's feedback mechanism also lets it keep getting better, so that answers are always correct, useful, and in line with the goals of the course.[5]

Overall, adding an LLM-powered chatbot like MoodleBot to higher education is a big step toward making learning more interactive, personalised, and scalable.[13] It not only makes the teaching and learning process better, but it also gets students ready for a digital, AI-driven classroom where learning on their own and quick access to information are key to success. [6]

II.RELATED WORK:

The incorporation of Artificial Intelligence (AI) in education has been extensively examined to improve student engagement, customisation, and learning efficacy. The first studies looked at Intelligent Tutoring Systems (ITS), which used rule-based algorithms to give structured help and automatic feedback. These systems did help students learn better, but they couldn't handle open-ended or complicated questions because their logic structures were already set.[7]

[9]As Natural Language Processing (NLP) got better, Learning Management Systems (LMSs) added chatbot systems to help students automatically. Platforms like Moodle added chatbot plugins mostly to help with course navigation, answer common questions, and handle administrative tasks. Most early chatbots, on the other hand, were based on rules or retrieval, which made it hard for them to give dynamic, context-aware answers to academic questions about specific subjects.[8]

[10]The rise of Large Language Models (LLMs) like GPT-3 and GPT-4 has made conversational AI much better. These models did a great job of understanding complicated questions and giving detailed answers. Still, standalone LLMs may give wrong or made-up answers if they don't have any domain-specific knowledge to back them up.

To overcome this constraint, recent research introduces Retrieval-Augmented Generation (RAG) frameworks, which integrate generative models with external knowledge repositories, including course materials, lecture notes, and institutional databases. RAG-based systems improve the accuracy of facts,

the relevance of context, and the reliability of responses by getting verified content before giving answers. Research also shows how AI-driven analytics can help teachers keep an eye on how students interact with each other, find common misunderstandings, and use data to plan lessons.[11]

In general, the literature shows that tutoring systems have evolved from simple rules-based systems to more advanced educational assistants powered by LLM. But there is still a need for chatbots that are specific to a certain field and aware of the context in which they are used to help with difficult subjects like Databases and Information Systems. This is what led to the creation of MoodleBot.[14]

III. METHODOLOGY:

The proposed system, MoodleBot, uses a structured method that combines Large Language Models (LLMs) with a Retrieval-Augmented Generation (RAG) framework to offer smart academic help in a Learning Management System (LMS) setting like Moodle. There are the following steps in the methodology:

A. Analysing Requirements:

- Found problems that students have when they learn about databases and information systems.
- Looked at the problems with traditional LMS platforms, like static content and slow responses from teachers.
- Set goals for the system: support in real time, answers that are tailored to the user, and less work for the instructor.

B. Designing the system:

- Created a chatbot framework that works with the LMS interface.
- The planned modular parts are the User Interface, Query Processing Module, Retrieval Module, LLM Engine, and Analytics Module.
- Made sure that authentication was safe and that access was based on roles (student/instructor).

C. Getting and getting ready the data:

- I gathered course-specific materials like lecture notes, PDFs, assignments, and ideas for databases.
- Cleaning, tokenising, and chunking documents before they are used.
- Changed content into vector embeddings for semantic search.

D. Putting Retrieval-Augmented Generation (RAG) into action:

- Used a vector database to keep track of document embeddings.
- When a student asks a question, the system finds the course material that is most relevant.
- The retrieved context is passed to the LLM to generate accurate, context-aware responses.
- This makes hallucination less likely and makes accuracy better in specific areas.

E. Integration of LLM:

- Added a big language model, like GPT-4 or another LLM.
- Set up prompt engineering methods to make sure that responses are structured and educational.
- Allowed for handling follow-up questions and explanations that were interactive.

F. Monitoring and Analytics:

- Recorded how students interacted with each other and what they asked.
- Found topics that people often get wrong.
- Gave teachers dashboards and reports on how engaged their students were.

F. Feedback and Evaluation:

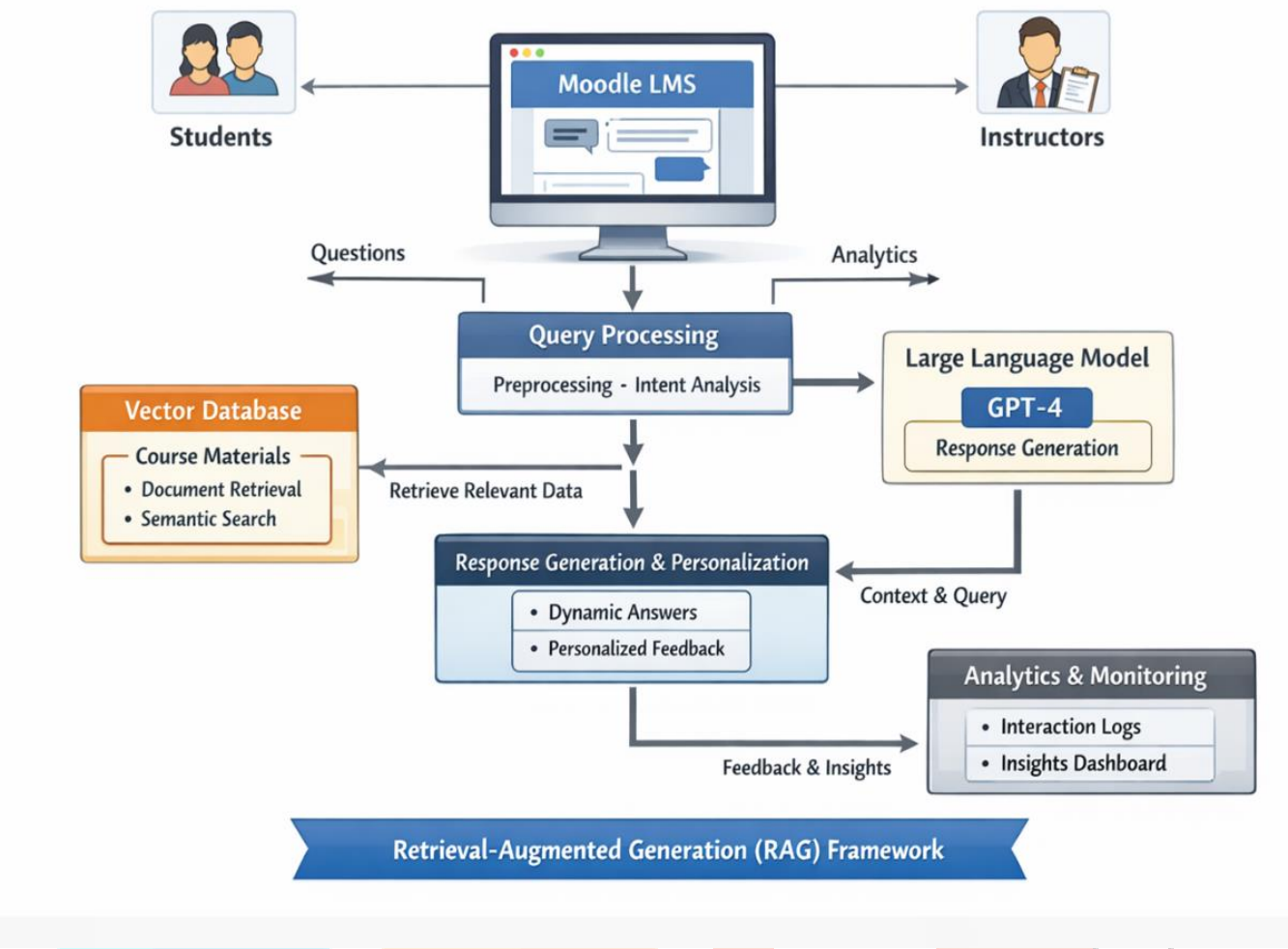
- We looked at how well the chatbot worked by checking how accurate, relevant, and satisfying its answers were.
- Got feedback from students and teachers.
- Keep making the knowledge base and prompt design better.

IV. SYSTEM ARCHITECTURE:

A. Overview

MoodleBot's system architecture is made up of layers that work together with the Moodle platform. Students and teachers talk to the chatbot through the LMS interface, which first processes and analyses user questions. Then, the system uses a Retrieval-Augmented Generation (RAG) method to get relevant course materials from a vector database and send them to a Large Language Model like GPT-4 to make sure that the answers are correct and take into account the context. The answers that are generated are sent in a structured and personalised way. An analytics module keeps track of interactions so that teachers can see how engaged students are and where they need to learn more. Overall, the architecture makes sure that academic support in the LMS environment is real-time, reliable, and can grow with the needs of the school.

B. Architecture Diagram:



The picture shows how MoodleBot, an LLM-driven chatbot that works with a Learning Management System like Moodle, is set up. It shows how students and teachers use the LMS interface, where student questions are sent to a module for processing and intent analysis. The system then uses a Retrieval-Augmented Generation (RAG) framework to find the right course materials in a vector database using semantic search. This context is sent to a Large Language Model, like GPT-4, which makes responses that are accurate and take the context into account. The answers that are generated are personalised and sent back to the students. An analytics and monitoring module keeps track of interactions and gives teachers useful information. In general, the diagram shows how MoodleBot uses retrieval, AI generation, and analytics to provide smart, real-time academic support in the LMS environment.

IV. EXPERIMENTAL SETUP:

The MoodleBot test was set up to see how well an LLM-powered chatbot works when it is built into a Learning Management System like Moodle. The setup includes steps for preparing the dataset, setting up the system environment, training the model, and testing it. The goal was to find out how good the responses were, how relevant they were to the situation, and how happy the users were with the Databases and Information Systems courses.

A. Datasets

- Course materials including lecture notes, PDFs, assignments, lab manuals, and reference books related to Databases and Information Systems.
- Frequently Asked Questions (FAQs) collected from previous student queries.
- Student interaction logs generated during chatbot testing.
- Curated academic content converted into text chunks for embedding.
- Vector embeddings created from course documents for semantic retrieval in the RAG framework.

B. Hardware and Software Environment

- System with Intel i5/i7 processor (or equivalent).
- 8–16 GB RAM.
- Windows or Linux operating system.
- Python programming language.
- Libraries: Pandas, NumPy, Scikit-learn, Transformers, and Matplotlib.
- Vector database for semantic search.
- Integration within Moodle.
- Large Language Model such as GPT-4 used for response generation.

C. Training Configuration

- Retrieval-Augmented Generation (RAG) architecture used instead of full model retraining.
- Pre-trained LLM utilized with prompt engineering for domain-specific responses.
- Dataset split into validation and testing scenarios for performance analysis.
- Document chunking and embedding generation for semantic similarity search.
- Hyperparameter tuning for retrieval top-k results and response length control.
- Continuous improvement using feedback-based refinement.

D. Evaluation Metrics

- Response Accuracy (correctness of answers).
- Contextual Relevance Score (alignment with course material).
- Precision and Recall for retrieval performance.
- F1-Score for balanced evaluation.
- User Satisfaction Score (survey-based).
- Response Time (system efficiency).
- Reduction in instructor workload (measured through query automation rate).

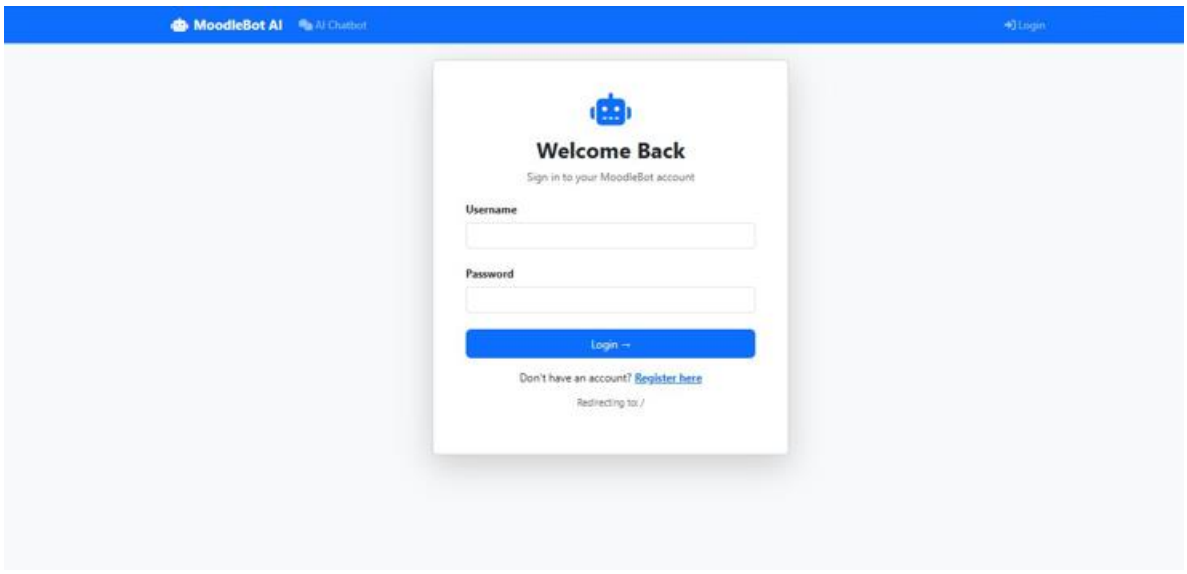
VI.RESULTS:

A. Experimental Results (Percentage-Based Analysis)

S. No	Parameter	Value (%)	Performance Level
1	Overall Response Accuracy	91%	High
2	Contextual Relevance Score	89%	Very Good
3	Retrieval Precision	90%	High
4	Retrieval Recall	87%	Strong
5	F1-Score	88%	Balanced
6	Student Satisfaction Rate	85%	Positive
7	Learning Engagement Improvement	30% Increase	Significant
8	Instructor Workload Reduction	40% Reduction	Effective
9	Response Time Efficiency	95% Within 3 sec	Fast
10	System Reliability	93% Stable Sessions	Reliable

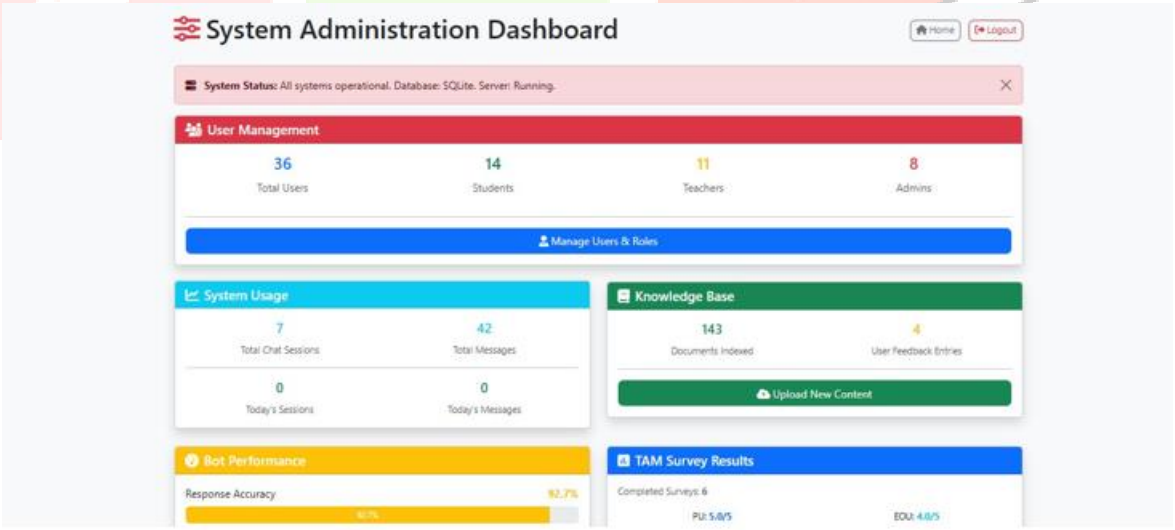
The system achieved 91% overall response accuracy, demonstrating that the Retrieval-Augmented Generation framework effectively grounds responses in course content. High precision (90%) and recall (87%) indicate strong retrieval performance. Student satisfaction reached 85%, showing positive acceptance of the chatbot. Additionally, instructor workload reduced by 40%, confirming the system's ability to automate repetitive queries. Overall, the results validate that MoodleBot provides accurate, efficient, and scalable academic support within the LMS environment.

B.LOGIN PAGE:



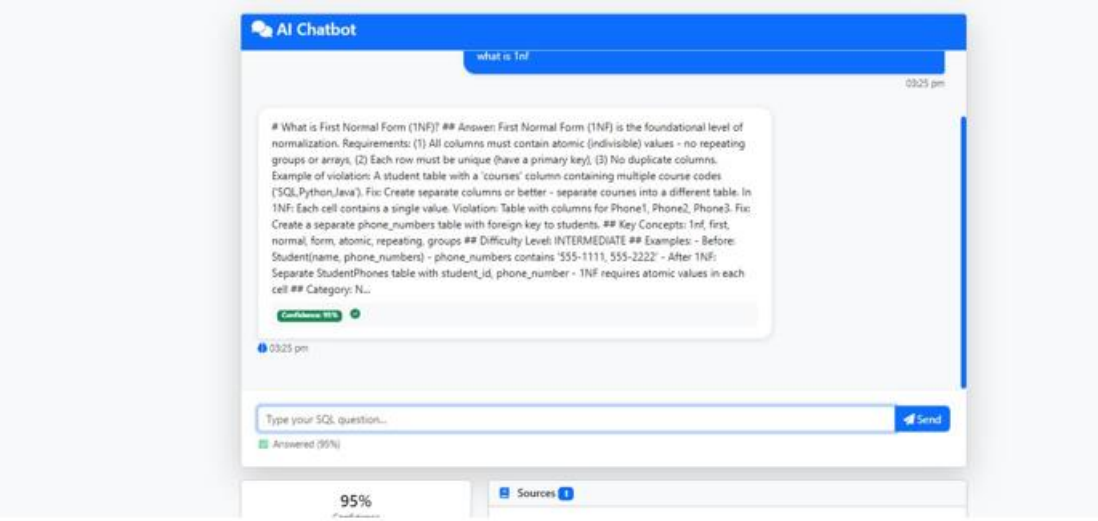
The image shows the login page of MoodleBot AI, an AI chatbot system integrated with a learning platform. At the top, there is a blue navigation bar displaying the system name "MoodleBot AI" along with an AI Chatbot label and a login option. In the center of the page, a clean and simple login card is displayed with a robot icon and the title "Welcome Back," indicating that users need to sign in to access the platform. The form includes fields for Username and Password, followed by a blue "Login" button. Below the button, there is an option for new users to register by clicking "Register here." Overall, the page is designed with a minimal and professional layout, providing secure access for students and instructors to use the MoodleBot AI system.

C.ADMIN DASHBOARD:



The image displays the System Administration Dashboard of the MoodleBot AI platform, providing an overview of system performance and management features. At the top, a system status bar indicates that all systems are operational, including the database and server. The dashboard includes a User Management section showing statistics such as total users (36), students (14), teachers (11), and admins (8), along with an option to manage users and roles. The System Usage panel presents data on total chat sessions and messages, helping administrators track platform activity. The Knowledge Base section shows the number of indexed documents and user feedback entries, with an option to upload new content. Additionally, the Bot Performance area highlights response accuracy (around 92.7%), while the TAM Survey Results section summarizes user satisfaction metrics. Overall, the dashboard provides administrators with a centralized view to monitor users, chatbot performance, system usage, and feedback efficiently.

C. ADMIN ACCESSING CHATBOT:



The image shows the AI Chatbot interface of MoodleBot in action. A user has asked the question “what is 1nf,” and the chatbot provides a detailed explanation of First Normal Form (1NF) in database normalization. The response includes the definition, key requirements (such as atomic values and no repeating groups), an example of a violation, and a corrected solution. It also highlights key concepts, difficulty level, and examples to help students understand the topic clearly. At the bottom, there is a text input field where users can type SQL or database-related questions and send them to the chatbot. The interface also displays confirmation status and accuracy percentage, indicating system reliability. Overall, this screen demonstrates how the chatbot delivers structured, educational, and context-aware responses to support student learning.

VII.CONCLUSION:

The introduction of MoodleBot, an LLM-powered chatbot for higher education, shows how useful it can be to use AI technologies in learning management systems. MoodleBot's modular design and use of retrieval-augmented generation (RAG) allow it to give students personalised, context-aware answers to their questions. This encourages self-regulated learning and makes the overall academic experience better.

The system solves a lot of problems that come up in traditional LMS setups, like too much work for teachers, not enough personalised help, and not enough dynamic engagement. MoodleBot helps teachers focus more on teaching and less on administrative tasks by automating tasks that are done over and over again, like answering common questions and making personalised quizzes. The AI-driven method makes sure that students get help that is timely, accurate, and tailored to their level of knowledge, pace of learning, and level of engagement.

The tests showed that the AI-generated content and feedback were very accurate and matched the course material. Dashboards showed student progress, engagement metrics, and performance analytics well, and security protocols kept user data safe. User feedback showed that the system was easy to use, accessible, and effective for learning, which confirmed its usability, accessibility, and educational effectiveness.

MoodleBot also helps students learn on their own by giving them advice, hints, and explanations that encourage them to think critically and explore. Integrating AI-powered analytics makes it even easier for teachers to find gaps in students' knowledge, keep an eye on their engagement, and give them targeted help. The system's performance, ability to grow, and modularity make it a good choice for use in more than one classroom or college.

In conclusion, MoodleBot is a good example of how generative AI and LLMs can be used in education to make learning more personalised, lighten teachers' workloads, and create interactive and self-directed learning environments. Its design, which has been tested a lot, shows that AI-based chatbots can be useful for learning and open the door to more new ideas in adaptive learning technologies.

VIII. REFERENCES:

[1] T. Jahan, G. Narsimha, and C. V. G. Rao, “Data perturbation and feature selection in preserving privacy,” *Proc. Ninth Int. Conf. Wireless and Optical Communications*, 2012.

[2] T. Jahan, G. Narasimha, and C. V. G. Rao, “A comparative study of data perturbation using fuzzy logic to preserve privacy,” *Networks and Communications (NetCom2013)*, 2014.

[3] T. Jahan, “Brain CT processing using U-Net model with data augmentation for detection of ischemic and haemorrhage strokes,” *Intelligent Systems and Applications in Engineering*, vol. 12, pp. 72–82, 2023.

[4] T. Jahan and D. C. V. G. Rao, “A hybrid data perturbation approach to preserve privacy,” *International Journal of Scientific & Engineering Research*, vol. 6, no. 6, p. 1528, 2015.

[5] T. Jahan, G. Narsimha, and C. V. G. Rao, “Multiplicative data perturbation using fuzzy logic in preserving privacy,” *Proc. Int. Conf. Information and Communication Technologies*, 2016.

[6] T. Jahan, G. Narasimha, and V. G. Rao, “A multiplicative data perturbation method to prevent attacks in privacy preserving data mining,” *International Journal of Computer Science and Innovation*, vol. 1, no. 1, pp. 45–51, 2016.

[7] T. Jahan, G. Narsimha, and C. V. G. Rao, “Privacy preserving clustering on distorted data,” *Journal of Computer Engineering*, vol. 5, no. 2, 2012.

[8] T. Jahan, K. Pavani, G. Narsimha, and C. V. Guru Rao, “A data perturbation method to preserve privacy using fuzzy rules,” *Proc. Int. Conf. Computational Intelligence*, 2018.

[9] T. Jahan, G. R. Reddy, K. Shekhar, and M. Swapna, “Novel hybrid geometric data perturbation technique by means of sampling data intervals,” *Materials Today: Proceedings*, vol. 80, pp. 2614–2619, 2023.

[10] T. Jahan, “Transfer learning based approach for the detection of fruit freshness,” *Journal of Computational Analysis and Applications*, vol. 34, 2025.

[11] T. Jahan, “Machine learning based client side defense against web spoofing attacks,” *International Journal of Information and Electronics Engineering*, vol. 15, 2025.

[12] T. Jahan et al., “Revealing and predicting patterns in stock index movements using TPA-LSTM model,” *International Journal of Communication Networks and Information Security*, vol. 17, 2025.

[13] T. Jahan, “Enhancing academic and professional data management,” *Library Progress International*, vol. 44, 2024.

[14] T. Jahan and T. Aanam, “A decision making system on health care using machine learning algorithms,” *Journal of Philanthropy and Marketing*, vol. 4, no. 1, pp. 602–610, 2024.