

IMPLEMENTATION OF FINE-TUNED MODEL FOR AI BASED SUPPORT SYSTEM IN LEARNING PLATFORM

A PROJECT REPORT

Submitted by

SARAN S V	962821104082
VARSHA HEROLIN A P	962821104101
ABINANTH S	962821104314
BRITHIKSHA N V	962821104704

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING



UNIVERSITY COLLEGE OF ENGINEERING NAGERCOIL

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2025

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report “**IMPLEMENTATION OF FINE-TUNED MODEL FOR AI BASED SUPPORT SYSTEM IN LEARNING PLATFORM**” is the Bonafide work of “**SARAN S V (962821104082), VARSHA HEROLIN A P (962821104101), ABINANTH S (962821104314), BRITHIKSHA N V (962821104704)**” who carried out this project under my supervision.

SIGNATURE

Dr. K.L. NEELA, M.E., Ph.D.,
HEAD OF THE DEPARTMENT,
Assistant professor (SI.Gr),
Computer Science and Engineering,
University College of Engineering,
Nagercoil.

SIGNATURE

Dr. T. VIVEKA, M.E., Ph.D.,
SUPERVISOR,
Teaching Fellow,
Computer Science and Engineering,
University College of Engineering,
Nagercoil.

Submitted for viva-voce examination held on _____

Internal Examiner

External Examiner

ABSTRACT

The rise of virtual learning platforms has exposed a critical gap in delivering real-time, adaptive, and personalized support, as traditional systems rely heavily on static content, limited interactivity, and predefined responses. This often leads to learner disengagement, lack of timely feedback, and inefficient knowledge transfer, especially when students face unique or complex queries. To address these limitations, this project introduces an AI-powered support system designed to enhance the digital learning experience through real-time query resolution, intelligent recommendations, and contextual educational guidance. The system is built using React for the frontend and Python for the backend, integrating fine-tuned AI models with advanced Natural Language Processing (NLP) to better understand and respond to student queries. A GPT-based conversational agent serves as the core of the system, offering human-like interaction, concept explanations, and automated assistance. Key features include automated assessment, intelligent hint generation, personalized feedback, and adaptive learning pathways that evolve based on the learner's behavior and performance. Additionally, the platform supports contextual knowledge retrieval, enabling it to pull relevant data from course materials or preloaded databases to generate meaningful and accurate responses. This approach not only reduces the workload on educators but also increases accessibility for students who require immediate and reliable support. By integrating intelligent automation with educational content delivery, this AI-based support system aims to redefine online learning environments, making them more interactive, efficient, scalable, and personalized for every learner. It ultimately contributes to improved academic outcomes, greater learner satisfaction, and a more inclusive and responsive digital education ecosystem.

ACKNOWLEDGEMENT

First and foremost, we thank **GOD** for his grace to enable us this project work successfully.

We would like to express our gratitude to Anna University Chennai for providing a better infrastructure for carrying out this project work.

Words cannot be expressed to show our gratitude to our respected Dean **Dr. V. A. NAGARAJAN, Ph.D.**, for his support and freedom throughout our course.

We also express our deep gratitude to our respected Head of the Department **Dr. K.L. NEELA, M.E., Ph.D.**, Assistant Professor (SI.Gr), Department of Computer Science and Engineering for her valuable suggestions throughout the course of this project.

We thank our Project Coordinator **Mr. R. NARENDRAN, M.Tech.**, Teaching Fellow, Department of Computer Science and Engineering for providing us with valuable suggestions and constant encouragement towards our project.

We express our gratefulness and profound thanks to our Supervisor, **Dr. T. VIVEKA, M.E., Ph.D.**, Teaching Fellow, Department of Computer Science and Engineering, for her valuable guidance throughout the project.

And our sincere thanks to all **Department Teaching and Non – Teaching Staff** members for their constant support for making this project a grand success.

Finally, we wish to thank **our parents** for their blessings to complete the project successfully.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	iii
	ACKNOWLEDGEMENT	iv
	LIST OF FIGURES	ix
	LIST OF TABLES	xi
	LIST OF ABBREVIATIONS	xii
1.	INTRODUCTION	1
	1.1 Description	1
	1.2 Objective	2
	1.3 Problem Definition	2
	1.3.1 Purpose	3
	1.3.2 Scope	3
	1.3.3 Contributions	4
2.	LITERATURE REVIEW	5
3.	PROPOSED SYSTEM	15
	3.1 Existing System	15
	3.1.1 Disadvantages	15
	3.2 Proposed System	16

3.2.1 Advantages	16
3.3. Architecture Diagram	17
3.4 Modules	18
3.4.1 User Interface Module	18
3.4.2 Authentication Module	19
3.4.3 Backend API Module	21
3.4.4 AI Query Processing Module	21
3.4.5 Database Module	22
4. SYSTEM REQUIREMENTS	23
4.1 System Description	23
4.2 Components/Tools Required	23
4.3 Software Tools	23
4.3.1 Visual Studio Code	23
4.3.2 Version Control with GitHub	24
4.3.3 OpenAI API Key	24
4.3.5 Google OAuth Client Credentials	32
4.4 Software Packages	33
4.5 Comparison with Existing System vs Proposed System	27
5. SYSTEM DESIGN AND IMPLEMENTATION	28
5.1 Introduction	28
5.2 Admin Module	28

5.2.1 Login Page	28	
5.2.2 School Creation	29	
5.2.3 Admin Page	30	
5.2.4 Course Creation	31	
5.2.5 Cohort Page	32	
5.2.6 Team Management	34	
5.2.7 Course Management	35	
5.3 User Module	36	
5.4 Backend Module	39	
5.5 Sample Code	39	
6.	CONCLUSION AND FUTURE WORK	42
6.1 Conclusion		42
6.2 Future Enhancement		42
7.	REFERENCES	44

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
3.1	System Architecture	17
3.4.2	Authentication module	20
5.1	Login Page	29
5.2	School Creation	30
5.3	Successful Creation Page	30
5.4	Admin Page	31
5.5	Course Creation	32
5.6	Course Created	32
5.7	Cohort Management	33
5.8	Assigning Course to Cohort	33
5.9	Adding Learners	34
5.10	Adding Mentors	34
5.11	Course Page	35
5.12	User Enrolled Course	36
5.13	Learning Material with AI Assistance	37
5.14	AI Assistance in Tamil	37
5.15	Quiz	38
5.16	Chapter Completion	38

LIST OF TABLES

TABLE NO	TITLE	PAGE NO
4.1	Comparison: Existing System vs Proposed System	27

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
LMS	Learning Management System
NLP	Natural Language Processing
LLM	Large Language Model
API	Application Programming Inter- face
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheets
JS	JavaScript
DB	Database
CSV	Comma-Separated Values
IDE	Integrated Development Environ- ment
CPU	Central Processing Unit
RAM	Random Access Memory
SSD	Solid State Drive
HTTP	HyperText Transfer Protocol
URL	Uniform Resource Locator
JSON	JavaScript Object Notation
XML	eXtensible Markup Language
ML	Machine Learning
UI/UX	User Interface/User Experience
DBMS	Database Management System
CRUD	Create, Read, Update, Delete

CHAPTER 1

INTRODUCTION

In today's educational landscape, Artificial Intelligence (AI) is revolutionizing how students interact with learning content and receive academic support. Traditional platforms often fall short in providing real-time, personalized assistance, leading to gaps in student understanding and engagement. Fine-tuning these models on subject-specific data further enhances their accuracy and relevance for learners.

This project implements a fine-tuned AI model as an intelligent academic assistant in an online learning environment. The system offers instant, context-aware responses to student queries, improving comprehension and delivering tailored guidance. Built with a React.js frontend and FastAPI backend, it ensures a seamless user experience and efficient communication with the AI. By reducing dependency on human tutors, this AI-driven solution provides consistent, accurate support and encourages self-paced learning—marking a significant advancement in the future of online education.

1.1 DESCRIPTION

This This project develops an AI-powered academic support system integrated into a digital learning platform. It uses a fine-tuned OpenAI language model trained on domain-specific content to provide real-time, accurate responses and personalized guidance, simulating a virtual tutor.

Built with a React.js frontend and FastAPI backend, the system ensures efficient, scalable interaction between students and the AI. It addresses challenges like delayed support and limited resources by offering consistent, personalized assistance that boosts engagement, promotes independent learning, and improves educational outcomes—advancing digital

transformation in education.

1.2 OBJECTIVE

The aim of this project is to create an AI-powered support system for an online learning platform by fine-tuning a pre-trained OpenAI language model. The goal is to improve student learning by providing a smart virtual assistant that can answer academic questions, explain concepts, and guide learners in real time. By customizing the model with educational content specific to the subject, the system ensures accurate, relevant, and context-aware responses. The project uses a modern web architecture, with React.js for the frontend to interact with users and FastAPI for the backend to handle data processing and AI integration. This AI assistant is designed to offer personalized support around the clock, reduce students' reliance on human tutors, and encourage independent, self-paced learning in digital environments.

1.3 PROBLEM DEFINITION

In both traditional and online learning environments, students often encounter difficulties in accessing timely and personalized academic support. These challenges include delays in receiving assistance from tutors or instructors, limited availability of resources outside regular hours, and the inability to quickly clarify complex concepts.

Although many platforms offer forums or FAQs, these solutions typically do not provide real-time, context-aware assistance. Students may also find it difficult to sift through large amounts of information or may not know where to locate the resources they need. As a result, unanswered questions can impede their learning progress. By integrating a fine-tuned language model into the learning platform, the solution offers real-time answers, explanations of concepts, and guidance across a range of academic subjects, ensuring a smooth and efficient learning experience without relying solely on human tutors.

1.3.1 Purpose

The purpose of this project is to develop and implement a fine-tuned artificial intelligence (AI) model that serves as a support system within a digital learning platform. The goal is to enhance the overall learning experience for students by providing intelligent, real-time assistance based on their individual needs. This AI-based system is designed to perform tasks such as answering student queries, evaluating quizzes, recommending learning materials, and offering explanations for complex topics. By fine-tuning a pre-trained language model using educational data, the system becomes more accurate, relevant, and responsive to academic contexts. This project aims to reduce the burden on educators, support personalized learning, and improve student engagement and academic performance by delivering timely and context-aware help within the platform.

1.3.2 Scope

The objective of this project is to design and implement a fine-tuned AI-powered support system aimed at enhancing the learning experience within online educational platforms. The system will provide real-time, intelligent assistance to students by answering academic queries, explaining concepts, and guiding learners through various topics. OpenAI's language model will be fine-tuned with educational datasets to ensure that the AI delivers accurate and relevant responses within an academic context.

The frontend of the system will be developed using React.js, creating an intuitive and user-friendly interface for smooth interaction between students and the AI assistant. FastAPI will power the backend, enabling high-performance handling of API requests and seamless integration with the fine-tuned AI model. The system will cover a wide range of subjects, including mathematics, science, language, and more, providing comprehensive support

across multiple educational domains. However, the project will not involve developing a complete learning platform, as it focuses on integrating the AI support system into an existing or hypothetical platform. Additionally, the project will be limited to text-based interactions, without incorporating voice or multimedia capabilities.

1.3.3 Contribution

- **Real-Time Query Resolution:** Instantly answers academic questions, helping students learn more efficiently and without delay.
- **Enhanced Learning Experience:** Delivers precise, context-aware responses through a fine-tuned AI model trained with subject-specific educational data.
- **Supports Independent Learning:** Encourages students to study at their own pace by enabling them to find answers and guidance on their own.
- **Reduced Tutor Workload:** Handles repetitive student queries, allowing educators to concentrate on more complex or value-driven teaching tasks.
- **Seamless Technology Integration:** Combines React.js on the frontend and FastAPI on the backend to ensure a responsive and user-friendly experience.
- **Subject Flexibility:** The system can be tailored to support various academic subjects and levels, making it adaptable for diverse learning needs.
- **Scalable Design:** Built to expand easily across different educational institutions and online learning platforms.
- **Wider Accessibility:** Accessible from any device with a web browser, improving usability and reach for all learners.
- **Personalized Learning with Data:** Can be further fine-tuned using student interaction data to deliver increasingly customized support over time.

CHAPTER 2

LITERATURE REVIEW

[1] B. Galhotra and D. Lowe, "AI Based Examination System: A Paradigm Shift in Education Sector," 2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON), Faridabad, India, 2022,

The paper titled "AI Based Examination System: A Paradigm Shift in Education Sector" by B. Galhotra and D. Lowe, presented at the 2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON) in Faridabad, India, discusses the transformative role of artificial intelligence in modernizing assessment methods within the education sector. The authors propose a comprehensive AI-driven examination system designed to address the limitations of traditional assessment techniques, such as manual grading, time consumption, and potential bias. The system leverages AI technologies like natural language processing, image recognition, and machine learning to automate test generation, invigilation, and evaluation processes. One of the key features highlighted is the use of intelligent proctoring to detect and prevent cheating during online examinations using real-time video analysis and behavioral tracking. Additionally, the paper discusses adaptive testing mechanisms where question difficulty adjusts based on the student's performance, enhancing the personalization and fairness of the evaluation. The authors emphasize how such AI-based systems can significantly improve efficiency, reduce administrative burdens, and provide immediate feedback to learners. Challenges such as data privacy, infrastructure requirements, and algorithmic fairness are also acknowledged. Overall, the study presents AI as a powerful tool for revolutionizing examinations and promoting a more efficient and reliable education system.

[2] A. E. Ubah et al., "A Review of Artificial Intelligence in Education," 2022 International Conference on Artificial Intelligence of Things and Crowdsensing (AIoTCs), Nicosia, Cyprus, 2022

The paper titled "A Review of Artificial Intelligence in Education" by A. E. Ubah et al., presented at the 2022 International Conference on Artificial Intelligence of Things and Crowdsensing (AIoTCs) in Nicosia, Cyprus, provides a comprehensive overview of the role and impact of artificial intelligence (AI) in modern educational systems. The authors explore how AI technologies are transforming traditional teaching and learning processes by enabling personalized learning, intelligent tutoring systems, automated grading, and data-driven decision-making. The review highlights key AI techniques used in education, including machine learning, natural language processing, and learning analytics, and how these methods enhance both the teaching experience and student engagement. Additionally, the paper discusses the integration of AI with educational platforms to adapt content delivery based on learners' individual progress, preferences, and performance. It also addresses critical challenges such as ethical concerns, data privacy, algorithmic bias, and the digital divide. The study emphasizes the importance of balancing technological innovation with educational equity and transparency. Overall, the paper offers valuable insights into the current state, opportunities, and future directions of AI in education, making it a useful resource for educators, researchers, and policymakers aiming to leverage AI to improve learning outcomes and teaching efficiency.

[3] M. Murtaza, Y. Ahmed, J. A. Shamsi, F. Sherwani and M. Usman, "AI-Based Personalized E-Learning Systems: Issues, Challenges, and Solutions," 2022,

The paper titled "AI-Based Personalized E-Learning Systems: Issues, Challenges, and Solutions" by M. Murtaza, Y. Ahmed, J. A. Shamsi, F. Sherwani, and M. Usman (2022) explores the growing impact of artificial intelligence in shaping personalized e-learning environments. The authors examine how AI technologies are being utilized to tailor learning experiences according to individual learner profiles, preferences, and performance patterns. The paper provides an in-depth analysis of various AI approaches used to build adaptive learning systems, including recommendation engines, intelligent tutoring systems, and predictive analytics. While highlighting the benefits of these systems—such as enhanced engagement, improved learner outcomes, and scalable education delivery—the authors also identify several challenges. These include concerns related to data privacy, the accuracy of personalization algorithms, scalability, content adaptation, and the digital divide among learners. Furthermore, the paper presents possible solutions to these issues, such as implementing transparent AI models, using secure data-handling techniques, and incorporating learner feedback loops for continual system improvement. The authors emphasize the importance of human-centered AI design in education to ensure ethical, inclusive, and effective implementation. Overall, the paper contributes to the ongoing discourse on integrating AI into e-learning by offering a balanced view of both its transformative potential and the challenges that need to be addressed.

[4] Y. Chen, H. Chen and S. Su, "Fine-Tuning Large Language Models in Education," 2023 13th International Conference on Information Technology in Medicine and Education (ITME), Wuyishan, China, 2023,
[5]

The paper titled "Fine-Tuning Large Language Models in Education" by Y. Chen, H. Chen, and S. Su, presented at the 2023 13th International Conference on Information Technology in Medicine and Education (ITME), investigates the application of large language models (LLMs) in the educational domain, with a focus on fine-tuning these models to better serve teaching and learning objectives. The study explores how pre-trained language models, such as those based on transformer architectures, can be adapted using domain-specific datasets to improve their effectiveness in educational settings. The authors highlight the potential of LLMs to enhance student learning through intelligent tutoring, automated feedback generation, personalized content delivery, and natural language interactions. By fine-tuning models on academic content, the system can provide context-aware responses that align with curriculum standards and student needs. The research also examines technical aspects, including model training, dataset selection, and evaluation metrics. Additionally, ethical concerns and challenges such as bias, data privacy, and model interpretability are discussed. The paper concludes that fine-tuned LLMs have the potential to revolutionize digital education by making AI-driven learning support more accurate, personalized, and scalable. This work contributes to the growing field of AI in education by demonstrating practical implementations of LLM fine-tuning techniques.

[6] J. Divasón, F. J. Martínez-de-Pisón, A. Romero and E. Sáenz-de-Cabezón, "Artificial Intelligence Models for Assessing the Evaluation Process of Complex Student Projects," in *IEEE Transactions on Learning Technologies*, vol. 16, no. 5, pp. 694-707, Oct. 2023,

The paper by J. Divasón, F. J. Martínez-de-Pisón, A. Romero, and E. Sáenz-de-Cabezón, titled "Artificial Intelligence Models for Assessing the Evaluation Process of Complex Student Projects," published in the *IEEE Transactions on Learning Technologies* (Vol. 16, No. 5, Oct. 2023), explores the use of AI to improve the fairness and consistency of assessing complex student projects. Traditional evaluation methods can be subjective, especially for projects involving multiple stages, diverse topics, and interdisciplinary content. The authors propose the use of machine learning models to support the evaluation process by identifying patterns, predicting outcomes, and ensuring that grading criteria are applied consistently across evaluators.

The study uses real-world data from educational institutions and applies different AI techniques, such as decision trees, support vector machines, and neural networks, to analyse assessment results. The findings show that AI can provide valuable feedback to educators, help reduce human bias, and enhance the reliability of project assessments. The paper emphasizes the importance of transparency and explainability in AI models to ensure trust in automated evaluation systems. Overall, the research highlights the potential of AI in modernizing educational assessment practices, making them more objective, data-driven, and aligned with student learning outcomes.

[7] B. F. Mon, A. Wasfi, M. Hayajneh and A. Slim, "A Study on Role of Artificial Intelligence in Education," 2023 International Conference on Computing, Electronics & Communications Engineering (iCCECE), Swansea, United Kingdom, 2023.

The paper titled "A Study on Role of Artificial Intelligence in Education" by B. F. Mon, A. Wasfi, M. Hayajneh, and A. Slim, presented at the 2023 International Conference on Computing, Electronics & Communications Engineering (iCCECE) in Swansea, United Kingdom, investigates the multifaceted impact of artificial intelligence (AI) on the education sector. The study explores how AI technologies are transforming educational practices by enabling personalized learning, intelligent tutoring systems, automated grading, and enhanced student engagement. The authors analyze various AI techniques such as machine learning, natural language processing, and data analytics, demonstrating their applications in curriculum design, assessment, and adaptive content delivery. Emphasizing the benefits, the paper highlights AI's role in providing tailored learning experiences that accommodate individual student needs, improving learning efficiency and outcomes. It also discusses the challenges in implementing AI in education, including data privacy concerns, ethical considerations, and the digital divide affecting equitable access to technology. The paper suggests strategies to overcome these challenges, such as developing transparent AI models and ensuring inclusive infrastructure. Overall, this study provides a comprehensive overview of AI's potential to reshape education by making learning more interactive, accessible, and effective, offering valuable insights for educators, policymakers, and researchers aiming to integrate AI into educational frameworks.

[8] A. K. Rekha, K. Gopal, D. Satheeskumar, U. A. Anand, D. S. S. Doss and S. Elayaperumal, "AI-Powered Personalized Learning System Design: Student Engagement And Performance Tracking System," 2024 4th International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 2024,

The paper titled "AI-Powered Personalized Learning System Design: Student Engagement and Performance Tracking System" by A. K. Rekha et al., presented at the 2024 4th International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), explores the integration of artificial intelligence into modern education systems to enhance personalized learning experiences. The study focuses on designing a system that dynamically adapts to individual student needs, learning pace, and performance levels by utilizing AI-driven algorithms and real-time data analytics. The proposed model not only delivers customized learning paths but also actively tracks student engagement and academic performance. Key components of the system include intelligent content recommendation, automated performance evaluation, and continuous feedback mechanisms to support both students and educators. By analyzing learner behavior and progress, the system offers timely interventions and suggestions, thereby improving learning outcomes. The paper emphasizes the importance of adaptive learning environments in higher education and highlights how AI can bridge gaps in traditional teaching methods. The authors also discuss system architecture, implementation strategies, and future enhancements. This research contributes significantly to the field of educational technology by presenting a scalable and data-driven approach to personalized learning and performance tracking, aligned with current advancements in artificial intelligence.

[9] A. Sabar, S. Gamage, S. Abishek, C. Lakmal, K. A. D. T. Kulawansa and P. Perera, "AI-Powered Student Learning System," 2024 8th SLAAI International Conference on Artificial Intelligence (SLAAI-ICAI), Ratmalana, Sri Lanka, 2024.

The paper titled "AI-Powered Student Learning System" by A. Sabar, S. Gamage, S. Abishek, C. Lakmal, K. A. D. T. Kulawansa, and P. Perera, presented at the 2024 8th SLAAI International Conference on Artificial Intelligence (SLAAI-ICAI) in Ratmalana, Sri Lanka, introduces an intelligent learning platform designed to enhance student education using artificial intelligence. The system integrates various AI techniques to personalize learning experiences based on individual student performance, learning speed, and preferences.

The authors developed a framework that leverages natural language processing (NLP), machine learning algorithms, and real-time data analytics to create adaptive learning paths. It also features an AI-based recommendation engine that suggests study materials, quizzes, and feedback tailored to each student's needs. The platform aims to improve student engagement and academic outcomes by offering continuous assessment, timely interventions, and automated support.

Through practical implementation and testing in academic environments, the study demonstrates how the AI-powered system helps identify learning gaps and provides customized support. It also supports educators by offering insights into student progress and performance trends. This research highlights the growing importance of AI in modern education and showcases how intelligent systems can transform traditional learning into a more effective and student-centered process.

[10] J. Han, J. Lu, Y. Xu, J. You, and B. Wu, "Intelligent Practices of Large Language Models in Digital Government Services," *IEEE Access*, vol. 12, pp. 8633-8640, Jan. 2024, doi: 10.1109/ACCESS.2024.3349969.

The article titled "Intelligent Practices of Large Language Models in Digital Government Services" by J. Han, J. Lu, Y. Xu, J. You, and B. Wu, published in *IEEE Access* (Vol. 12, Jan. 2024), explores the integration and application of large language models (LLMs) in the development of intelligent digital government services. The paper highlights how LLMs, such as those based on transformer architectures, can enhance the efficiency, accuracy, and responsiveness of public service delivery through advanced natural language processing capabilities. By analyzing real-world use cases, the authors demonstrate how these models support tasks such as automated citizen interaction, document summarization, multilingual translation, and knowledge retrieval from large volumes of government data. The study emphasizes the potential of LLMs to enable more accessible and inclusive digital platforms, especially for diverse populations with varying language and literacy levels. Moreover, the paper addresses key implementation challenges, including data privacy, model bias, computational costs, and the need for transparency in AI decision-making. The authors suggest best practices for responsible deployment of LLMs, including human-in-the-loop strategies, secure data handling, and continuous performance monitoring. This research contributes valuable insights into how AI, particularly LLMs, can modernize public administration and foster smarter, citizen-centric governance.

[10] M. Z. Li, "Using Prompt Engineering to Enhance STEM Education," 2024 IEEE Integrated STEM Education Conference (ISEC), Princeton, NJ, USA, 2024

The paper titled **"Using Prompt Engineering to Enhance STEM Education"** by M. Z. Li, presented at the 2024 IEEE Integrated STEM Education Conference (ISEC) in Princeton, NJ, explores the innovative use of prompt engineering techniques to improve teaching and learning in STEM (Science, Technology, Engineering, and Mathematics) education. Prompt engineering involves designing and refining input prompts to guide AI models, such as large language models, to generate more relevant, accurate, and context-aware educational content. The study highlights how well-crafted prompts can enhance interactive learning experiences by providing personalized explanations, problem-solving guidance, and adaptive feedback tailored to individual students' needs. By leveraging AI's natural language understanding capabilities, prompt engineering can support educators in creating dynamic learning environments that foster critical thinking and conceptual understanding in STEM subjects. The paper discusses practical applications, including automated tutoring, real-time question answering, and personalized learning pathways. Additionally, it addresses challenges such as designing effective prompts, ensuring content accuracy, and aligning AI responses with curriculum standards. The research underscores the potential of prompt engineering as a cost-effective and scalable approach to augment traditional STEM education methods, promoting student engagement and improving learning outcomes through AI-enhanced instruction.

CHAPTER 3

PROPOSED SYSTEM

3.1 EXISTING SYSTEM

Current academic support systems in traditional and online settings rely heavily on human involvement and static content. In physical classrooms, help is provided mainly during lessons or scheduled sessions, with limited support outside class through appointments or group studies. Online platforms offer resources like PDFs, videos, and forums, sometimes supplemented by basic rule-based chatbots that answer only common questions via predefined scripts. These methods lack personalized, scalable, and immediate assistance. As a result, students often face delays in getting help outside regular hours or when instructors are unavailable. Such systems struggle to provide timely, individualized support, especially in large-scale or self-paced online courses where real-time human interaction is limited or absent.

3.1.1 Disadvantages

- **Lack of Real-Time Interaction:** Static resources like videos and notes offer no opportunity for follow-up questions or deeper exploration of concepts.
- **No Personalization:** Traditional systems fail to adapt to individual learning styles, paces, or needs, making it harder for some students to keep up.
- **Faculty Overload:** Instructors spend a lot of time addressing repetitive questions, which limits their ability to focus on innovation, mentoring, or curriculum improvement.
- **Low Student Engagement:** A lack of interactive, real-time support in digital platforms often leads to reduced motivation and participation.
- **Poor Scalability:** Human-based support cannot effectively scale to

accommodate large student populations while maintaining quality and personalization.

3.2 PROPOSED SYSTEM

The proposed system is an AI-powered academic support solution that offers real-time, personalized assistance using a fine-tuned GPT language model trained on subject-specific educational content. Integrated into a learning platform with a React.js frontend and FastAPI backend, it enables students to interact naturally with the AI tutor—asking questions and receiving instant, accurate, context-aware responses. The model understands academic terminology, handles diverse question types, and delivers concept-driven explanations tailored to each learner's level. By eliminating reliance on instructor availability, this system provides flexible, consistent, and scalable support, significantly enhancing the learning experience

3.2.1 Advantages

- **Instant Support:** Students get immediate answers to their questions, helping them stay on track and avoid disruptions in their learning process.
- **Contextual Understanding:** The fine-tuned model delivers subject-specific responses that closely resemble human-level comprehension.
- **Personalized Assistance:** Responses are tailored to each student's learning level and the complexity of their questions, promoting individualized learning.
- **Highly Scalable:** Capable of supporting a large number of users simultaneously, the system is ideal for growing institutions and online platforms.
- **Cost-Efficient Solution:** Minimizes the need for additional tutors or support staff, offering an affordable and sustainable academic support system.

3.3 ARCHITECTURE DIAGRAM

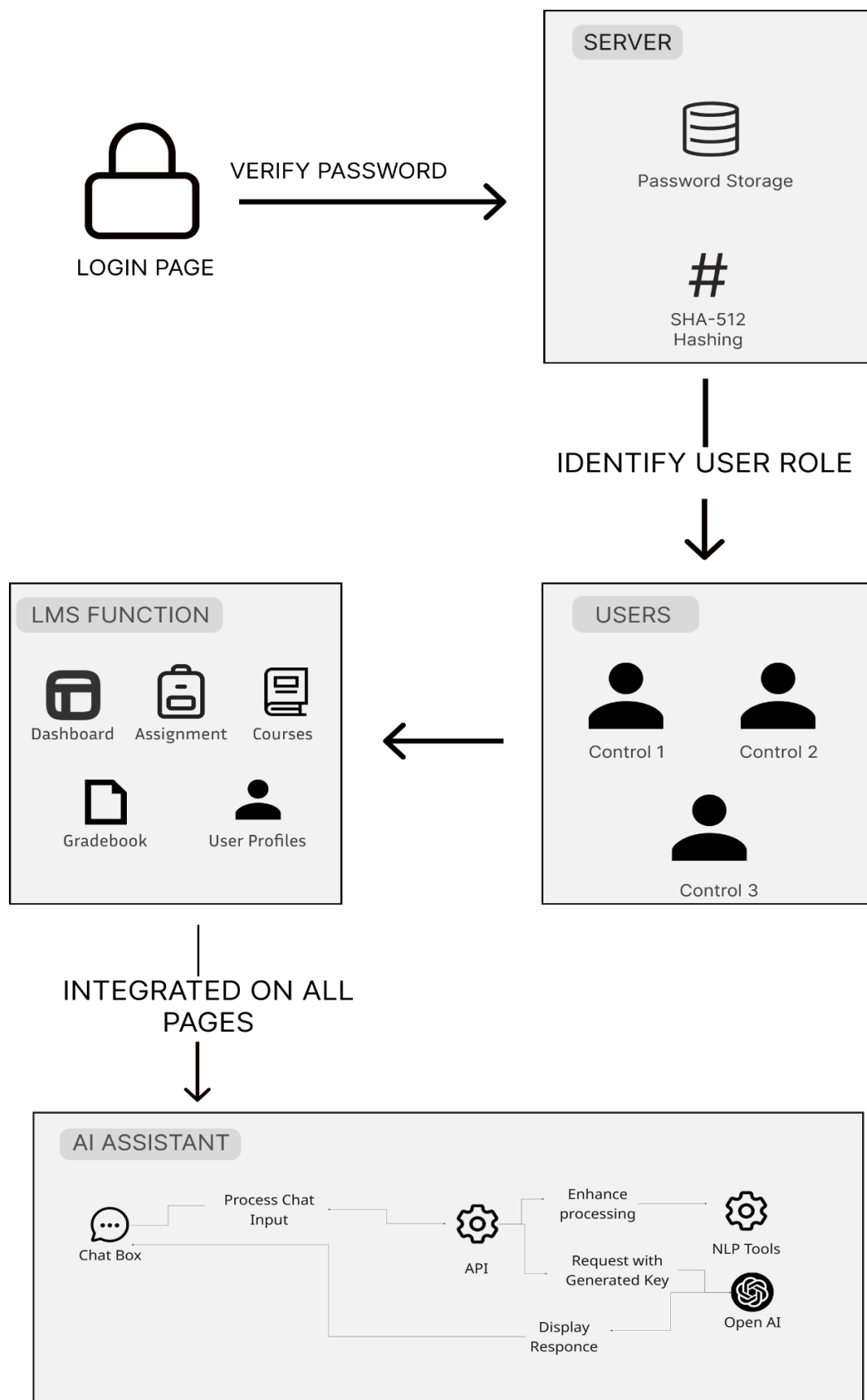


Figure 3.1 System Architecture

The architecture illustrates the complete workflow of an AI-enhanced academic support system integrated into a Learning Management System (LMS). It is designed to ensure secure user access, personalized experiences based on user roles, and intelligent real-time support through a fine-tuned AI model. The process begins at the login page, where users enter their credentials. These credentials are verified by an AWS-hosted server, with passwords securely stored and encrypted using SHA-512 hashing. Once authentication is successful, the system identifies the user's role—such as student, teacher, or admin (Control-1, Control-2, Control-3)—which determines their access level within the LMS.

After logging in, users are granted access to core LMS functionalities. These include a personalized dashboard, assignment management, access to course materials, gradebook views, and user profile settings. To provide seamless academic assistance, an AI-powered chat box is integrated across all pages of the platform. When a student submits a question, the input is first processed through an API and may be refined using Natural Language Processing (NLP) tools for improved understanding. The processed query, secured with a generated key, is then sent to a fine-tuned OpenAI model that has been trained on domain-specific educational data. This model interprets the query and generates a context-aware, accurate response. The AI-generated reply is then returned and displayed in the chat box, enabling students to receive instant support without waiting for human intervention. This system ensures around-the-clock, scalable academic assistance, improving learning efficiency and overall user engagement.

3.4 MODULES

3.4.1 User Interface Module (Frontend – React.js)

The User Interface (UI) Module is built using React.js, a powerful framework for developing fast, component-based web applications. It offers a responsive

and intuitive front-end for the AI-powered learning support system, focusing on simplicity, usability, and seamless interaction. When users visit the platform, they encounter a clean landing page with options to log in or register. Once logged in, they are directed to a personalized dashboard displaying recent learning activity, progress updates, and notifications, with a sidebar menu for easy access to various sections including chat, tasks, profile, and performance tracking.

A standout feature of the UI is the AI Chat Interface, which resembles modern messaging apps with features such as message bubbles, typing animations, timestamps, and a scrollable history. Students can select a subject, ask academic questions, and receive instant AI-generated answers. The platform also allows students to rate the responses to aid in model refinement. For guided learning, the Task Panel presents categorized tasks showing their current status—such as pending, in progress, or completed—and allows answer submissions directly through the interface.

Additional elements include a Progress Tracking Module that visualizes performance with charts, progress bars, and badges, helping students stay motivated. In the Profile Section, users can update their personal information, set learning preferences, and customize display settings such as light or dark mode. A Feedback Module is also integrated, enabling students to report bugs, suggest improvements, or share their experience via a simple form. The entire interface is optimized for responsiveness, ensuring accessibility across desktops, tablets, and mobile devices while maintaining a clean, distraction-free layout that enhances the overall learning experience.

3.4.2 Authentication Module

The Authentication Module is a core part of the system that ensures secure and controlled user access, allowing only authorized individuals to interact with the platform. Built using React.js for the frontend and FastAPI for the backend,

it manages key functionalities such as login, registration, and session management. When users first arrive, they encounter a Login or Sign-Up screen where they provide essential details like name, email, password, and role—whether as a student, instructor, or admin. These details are validated on the client side and then securely sent to the backend, where they are encrypted and stored in the database.

During login, the system authenticates the user's credentials on the server and, if verified, grants access by issuing a JSON Web Token (JWT) that maintains the user's session securely. Additional features like “Forgot Password” allow users to reset their credentials through a link or OTP verification, enhancing user convenience without compromising security. Role-based access control is enforced to ensure users can only interact with features relevant to their roles—students can access chat and task sections, while admins have permissions to manage users and view analytics. To further secure the platform, route protection is implemented to prevent unauthorized navigation to internal pages, and sessions are set to expire after periods of inactivity to avoid potential misuse. All data transferred between the frontend and backend is encrypted using HTTPS, ensuring secure communication throughout. Overall, the authentication process is crafted to be seamless, reliable, and secure, providing both students and educators with safe and straightforward access to the system.

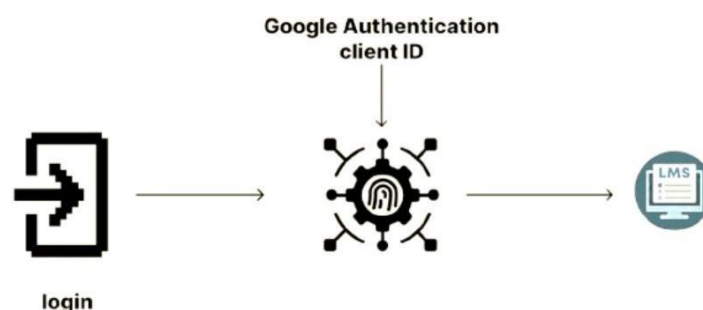


Figure 3.4.2 Authentication module

3.4.3 Backend API Module

The The Backend API Module, built with FastAPI, acts as the core service layer of the AI-based learning platform. It handles user requests such as logins, AI queries, and progress checks by validating inputs, querying a MySQL database, and communicating with the AI model to return responses efficiently. Secure authentication is managed via JWT, restricting access based on user roles (student, teacher, admin), and all data exchanges are encrypted.

The module also supports features like task tracking, feedback submission, and performance monitoring, with clear error handling to improve user experience. Its modular and scalable design ensures smooth interaction between the frontend, database, and AI services, while allowing for easy maintenance and future expansion

3.4.4 AI Query Processing Module (OpenAI Fine-Tuned Model)

The AI Query Processing Module is the system's intelligent core, handling student questions using a fine-tuned OpenAI model specialized in educational content. It processes queries by incorporating relevant context like subject and preferred formats, enabling the model to generate accurate, clear, and student-friendly answers based on textbooks, solved problems, and FAQs.

When a question is asked, it's routed through the backend to the fine-tuned model, which produces responses designed to promote understanding and suggest related topics for deeper learning. This real-time academic support empowers students to learn independently with personalized, context-aware assistance that goes beyond general language models.

The module continuously improves through user feedback and interaction data, allowing the AI to refine its responses over time. By leveraging domain-specific training and adaptive learning, it ensures that students receive up-to-date, relevant, and engaging support tailored to their individual learning needs.

3.4.5 Database Module

The Database Module is a critical component of the AI-based support system, responsible for securely storing, organizing, and managing all the data that powers the platform. Acting as the system's memory, it holds essential information such as user profiles, login credentials, chat history, AI-generated responses, learning tasks, feedback, and student progress records. This module uses a relational database like MySQL, which structures the data in defined tables with relationships (using foreign keys) to ensure consistency and integrity. For instance, separate tables are used for users, messages, tasks, progress, and feedback, all interconnected to maintain an organized and scalable data structure.

Whenever a user registers, logs in, submits a question, or receives an AI response, the relevant data is either stored in or retrieved from the appropriate table. For example, student queries and the AI's answers are saved in the chat history table, while performance scores and activity timestamps are recorded for tracking academic progress.

The backend communicates with the database using SQLITE queries or ORM (Object Relational Mapping) tools to perform operations like inserting, updating, retrieving, or deleting records. The Database Module also includes security measures to ensure that only authorized users can access their data, maintaining user privacy and data protection. Additionally, performance optimizations such as indexing, caching, and efficient queries help the system remain fast and responsive, even under heavy usage. Backup strategies are also in place to safeguard against data loss. In summary, the Database Module ensures accurate, secure, and efficient data management, serving as a foundation for the smooth and reliable operation of the learning platform

CHAPTER 4

SYSTEM REQUIREMENTS

4.1 SYSTEM DESCRIPTION

This chapter gives a brief Introduction about the hardware and software requirements, the tools and functions that is used in Implementation of Fine-Tuned Model for AI Based Support System in Learning Platform.

4.2 COMPONENTS REQUIREMENTS

➤ Hardware Specification

- A PC or Laptop with Windows OS (10 or above)
- 8 GB RAM
- 5 GB free Memory Storage

➤ Software Tools

- Code Editor
- Version Control with GitHub
- OpenAI API Key
- Judge0 API Key
- Google OAuth Client Credential

4.3 SOFTWARE TOOLS

4.3.1 Visual Studio Code

Visual Studio Code (VS Code) is a lightweight, open-source code editor by Microsoft used as the main development environment for this AI-powered system. It supports multiple languages and frameworks, making it ideal for managing both frontend (React.js), backend (FastAPI), and AI scripting tasks. Features like IntelliSense, integrated terminal, Git support, debugging, and extensions (e.g., Python, React, REST Client) enhance productivity and streamline development. Its performance and organized structure make it efficient for large-scale, full-stack projects.

4.3.2 Version Control with GitHub

GitHub is essential for modern development, providing distributed version control and collaborative features. It allows remote teams to manage code, track changes, and work simultaneously via branching and merging. Pull requests support code review and collaboration, while issue tracking, wikis, project boards, and discussions enhance team coordination. GitHub also integrates with CI/CD tools to automate testing and deployment, enabling efficient, high-quality software delivery.

4.3.3 OpenAI API Key

The OpenAI API Key is a secure token that allows the backend (FastAPI) of this system to access OpenAI's AI models, including fine-tuned versions, for real-time student support. It enables features like chat completions, embeddings, and other AI capabilities essential to the learning platform.

The key ensures that only authorized applications can access the API and helps track usage (requests and tokens), which is important for managing costs. For security, it must never be exposed in frontend code or public repositories.

Judge0 API Key

The Judge0 API Key connects your app to Judge0, an online code execution engine supporting multiple languages (Python, Java, C++, etc.). It allows students to write, run, and test code within the platform, with results returned in real-time—including output, errors, and execution time.

4.3.4 Google OAuth Client Credentials

Google OAuth Client enable secure user login via Google accounts using the OAuth 2.0 protocol. This simplifies authentication for students and teachers by leveraging existing Gmail credentials.

- **Client ID:** Public identifier for the app
- **Client Secret:** Private key for secure server communication with Google

4.4 SOFTWARE PACKAGES

This section outlines the major Python packages utilized in developing the AI-based learning support system, emphasizing their purpose and role in the platform's architecture.

Langchain-core: Langchain-core was used to create intelligent workflows through its Runnable interface, allowing prompt chaining and memory handling. It structured conversations for dynamic, multi-turn user interactions. This helped maintain context across sessions, making the learning experience more fluid and responsive.

Pydantic: Pydantic ensured that data exchanged between the frontend and backend was valid and reliable by using Python type hints for input validation. It was used for forms like login and chat input, catching data issues early. This improved security, reduced bugs, and enhanced user trust.

OpenAI: The OpenAI Python library was used to integrate GPT models for tasks like natural language understanding and intelligent Q&A. It enabled the backend to provide smart, context-aware replies in real-time. This added AI-powered interactivity to the learning platform, making it more engaging.

Python-dotenv: Python-dotenv helped securely load configuration values like API keys and database credentials from a .env file. It separated sensitive data from the source code, promoting safe and scalable deployment. This made the app easier to configure across environments while improving security.

FastAPI: FastAPI served as the backbone of the backend, offering fast, async handling of API requests. It was integrated with OpenAI and Pydantic for building secure and efficient RESTful endpoints. Its performance and simplicity allowed for rapid development and seamless API interaction.

Uvicorn: Uvicorn acted as the high-performance server to run the FastAPI app, using uvloop and httptools for optimized networking. It ensured that user requests

were handled quickly and concurrently. This boosted real-time responsiveness and made the system scalable under load.

Imgkit: Imgkit converted HTML templates into image formats like PNG and JPEG using `wkhtmltoimage`. It was used to create visual snapshots of reports, user progress, or dynamically generated content. This helped produce shareable and visually appealing output from structured HTML.

Aiosqlite: Aiosqlite provided asynchronous access to a lightweight SQLite database, ideal for logging user progress and storing interaction data. It allowed non-blocking read/write operations in a FastAPI environment. This ensured efficient data management without compromising real-time performance.

Google-OAuth: Google-OAuth enabled secure login via Google accounts, using OAuth 2.0 for authentication and access management. This reduced the need to store sensitive user credentials and facilitated safe access to Google services. It improved security and simplified the user login process.

HTTPtools: HTTPtools, used as a FastAPI dependency, handled low-level HTTP request parsing and response formatting. It optimized the handling of user traffic, supporting faster and concurrent data exchange.

APScheduler: APScheduler scheduled automated tasks like report generation, reminders, and database cleanup. It ran jobs at intervals without blocking the main application threads. This allowed smooth automation of background processes, improving operational efficiency.

Websockets: Websockets enabled real-time, two-way communication between users and the server, especially for chat features. Unlike traditional HTTP, it kept connections alive for instant updates. This created a responsive, interactive learning experience with live feedback and support.

4.5 Comparison: Existing System vs Proposed System

ASPECT	TRADITIONAL PLATFORMS	PROPOSED SYSTEM
Evaluation Mechanism	Manual or quiz-based evaluation, lacks personalization.	AI-powered task evaluation using NLP and ML models for instant, contextual feedback.
Learning Format	Pre-recorded video lectures, passive learning.	Task-first, active learning with real-time tasks, instant validation, and feedback.
Assistance & Support	Limited to forums or scheduled mentor sessions.	Instant AI assistance using inbuilt AI Coach that explains concepts, errors, and guides improvement.
Adaptability	Fixed curriculum, same for all learners.	Adaptive learning paths based on learner's performance and AI feedback loops.
Feedback & Improvement	Delayed, generalized feedback (e.g., end-of-course grades).	Granular, real-time feedback after every task using attention maps, code analysis, etc.
Time to Skill	Longer, as learners need to synthesize learning independently.	Shorter, AI-accelerated skill acquisition through hands-on, assisted learning.
Scalability & Cost	Heavy reliance on content creators, high bandwidth and licensing costs.	Open-source, lightweight, easy to deploy in classrooms, institutions, and enterprises.
Personalization	One-size-fits-all course structure.	Dynamic task recommendation based on learner strengths and weaknesses (AI-driven skill graph).
Project Readiness	Courses often end with guided projects, lacking autonomy.	Projects are autonomous, real-world aligned, AI-reviewed, and GitHub-exportable.
Deployment Focus	Academic or certification-oriented, not optimized for deployment.	Designed for skill-to-deployment transition, enabling learners to build deployable outputs quickly.
Interactivity	Mostly watch-and-learn model, low engagement.	AI-curated interactive tasks and simulations improve engagement and concept clarity.

CHAPTER 5

SYSTEM DESIGN AND IMPLEMENTATION

5.1 INTRODUCTION

The AI-based academic support system was successfully implemented and integrated into a custom Learning Management System (LMS). The system allows users—students, instructors, and administrators—to securely log in and access various educational functionalities based on their assigned roles. Students can interact with a fine-tuned AI model via a chat interface to receive instant, context-aware responses to academic queries across different subjects. The system was tested with multiple sample inputs and showed high accuracy in understanding and answering subject-specific questions.

5.2 ADMIN MODULE

5.2.1 Login Page

The login page of InternTribe offers a modern and minimalistic user interface with a dark-themed background. It provides users with a quick and easy sign-in option using Google authentication. On the left side, a bold tagline "Teach smarter. Reach further." communicates the platform's mission of empowering educators through AI.

Below the tagline, a short description explains InternTribe's purpose as an AI-powered Learning Management System (LMS). It highlights the platform's ability to support learners through smart questioning and grading, helping teachers maximize reach without compromising quality.

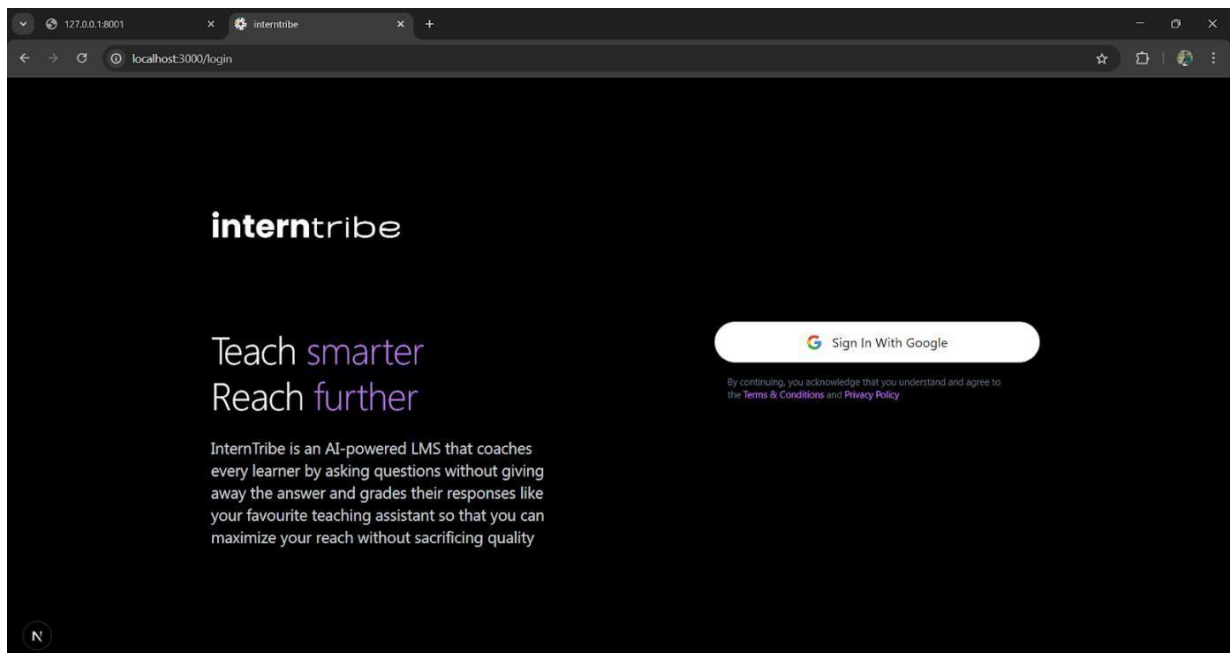


Figure 5.1 Login Page

The InternTribe login page uses Google Sign-In with OAuth 2.0. When the user clicks the "Sign in with Google" button, they are safely logged in using their Google account. This method is secure because the user doesn't have to enter a password on the site. Instead, Google confirms their identity and shares basic details like name and email. This makes the login process easy, fast, and trusted.

5.2.2 School Creation

This page allows users to create a new school on the InternTribe platform. It is designed with a clean and dark-themed layout.

There are two input fields:

1. School Name – where the user can enter the name of their school or organization (up to 40 characters).
2. School Link – where the user sets a unique link that students will use to access the school online.

After filling in both fields, the user can click the “Create School” button to complete the setup.

This page is part of the onboarding process for admins to start managing their own learning space on the InternTribe system.

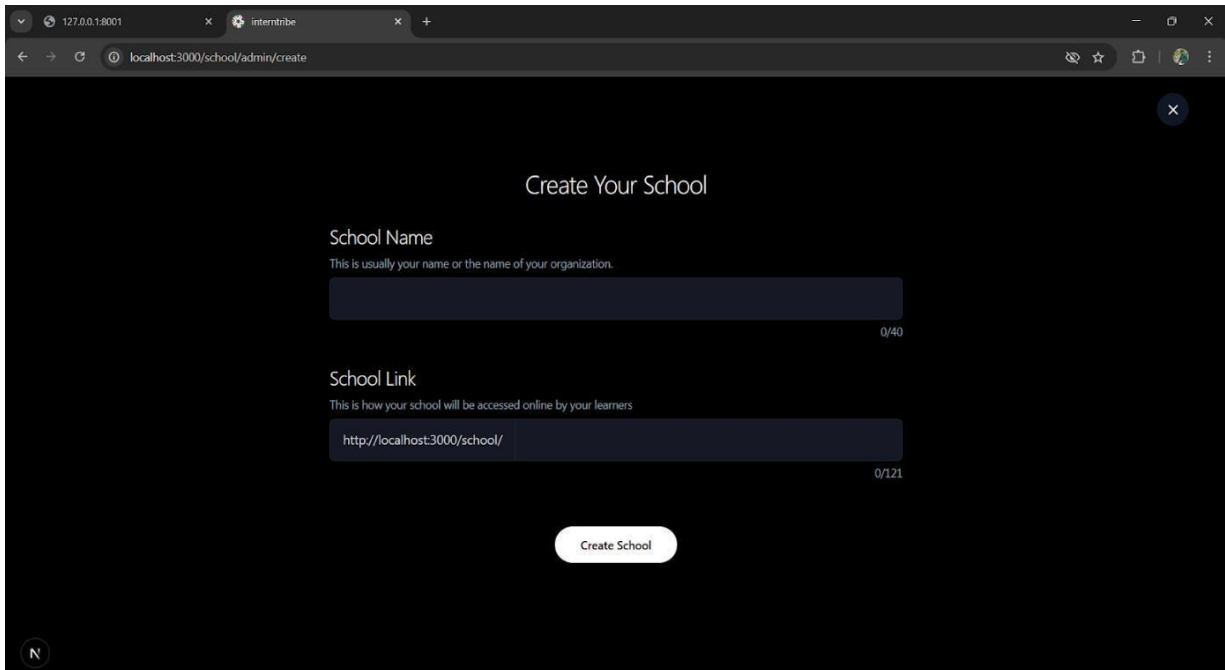
A screenshot of a web browser showing the 'Create Your School' page. The browser's address bar displays 'localhost:3000/school/admin/create'. The page has a dark background. At the top, it says 'Create Your School'. Below this, there are two input fields. The first is labeled 'School Name' with a hint 'This is usually your name or the name of your organization.' and a character count '0/40'. The second is labeled 'School Link' with a hint 'This is how your school will be accessed online by your learners' and a character count '0/121'. The link field already contains 'http://localhost:3000/school/'. At the bottom of the form is a 'Create School' button. A small 'N' icon is visible in the bottom left corner of the page.

Figure 5.2 School Creation

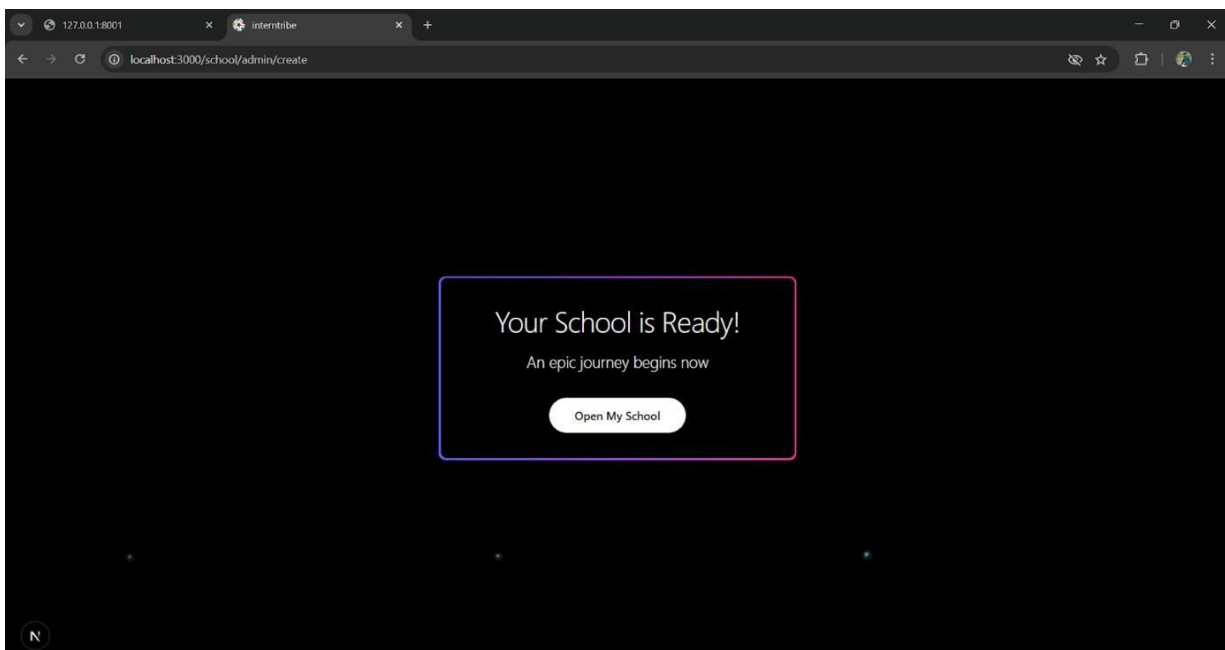


Figure 5.3 Successful Creation Page

5.2.3 Admin Page

This is the admin dashboard of the InternTribe platform, shown after a

school is created. The page displays the school's name, its unique link, and three main tabs: Courses, Cohorts, and Team.

Below that is a button labelled “Create Course”, which allows the admin to start building a new course for their learners. This interface makes it easy for school admins to manage their educational content.

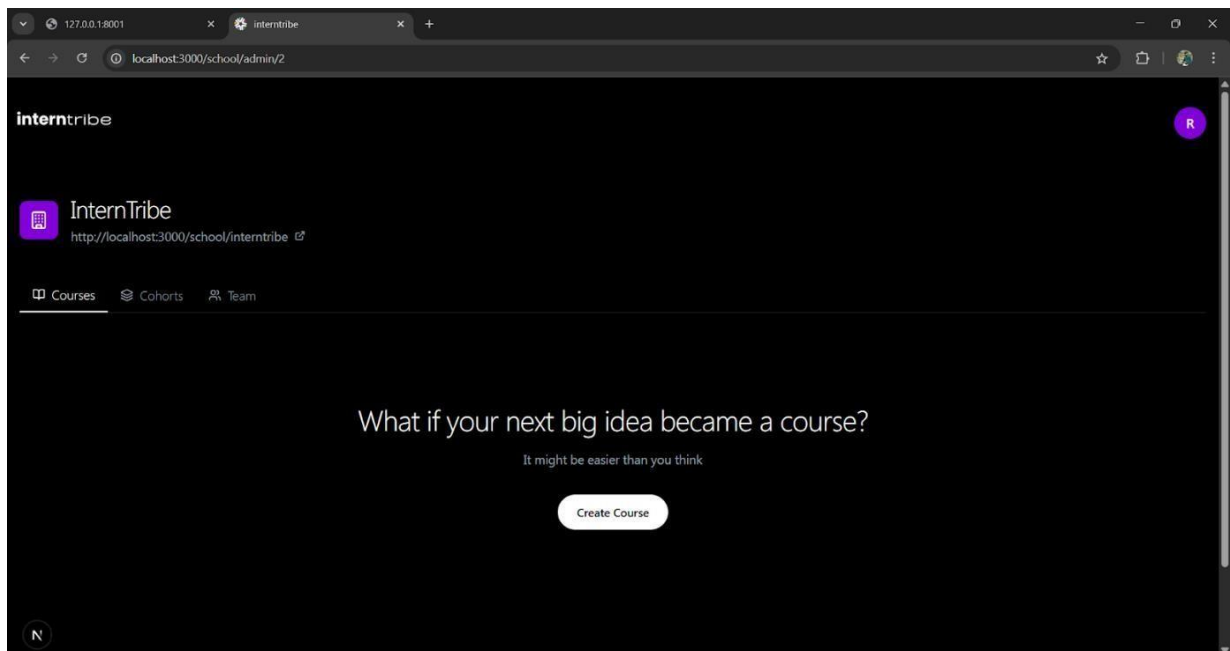


Figure 5.4 Admin Page

5.2.4 Course Creation

The course creation feature allows administrators or instructors to easily add new courses to the learning management system. When initiating the process, a user-friendly dialog box appears, prompting the user to enter the course name. The design emphasizes usability with clear options to either cancel the action or confirm course creation, enhancing the overall user experience.

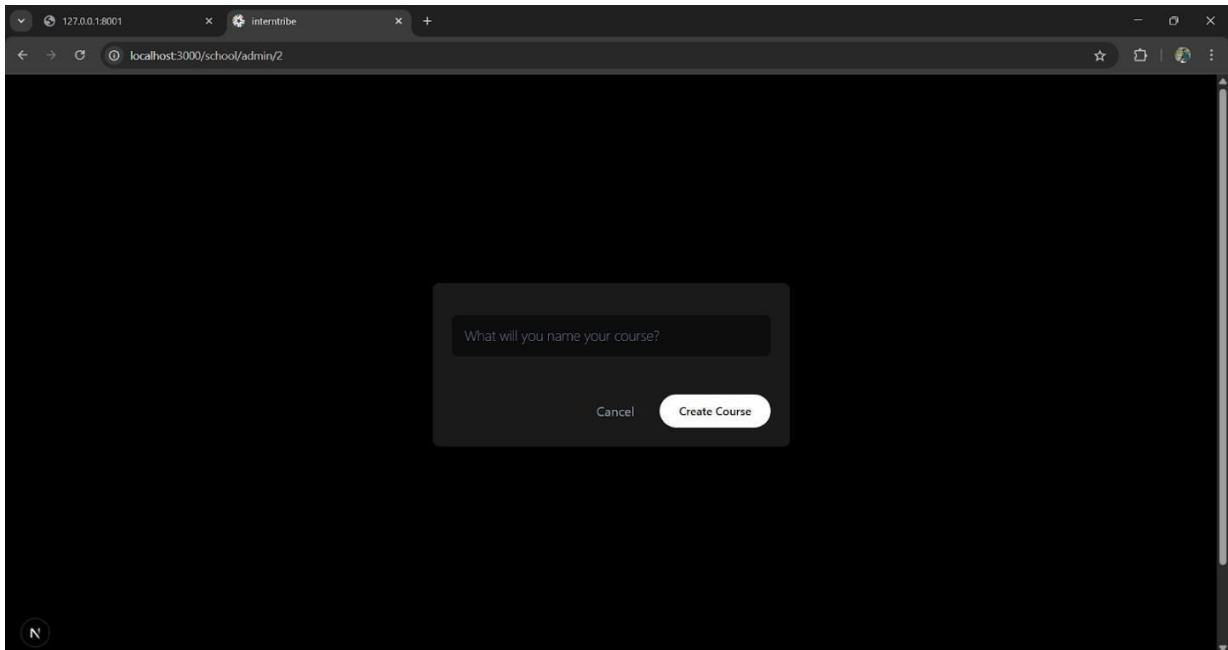


Figure 5.5 Course Creation

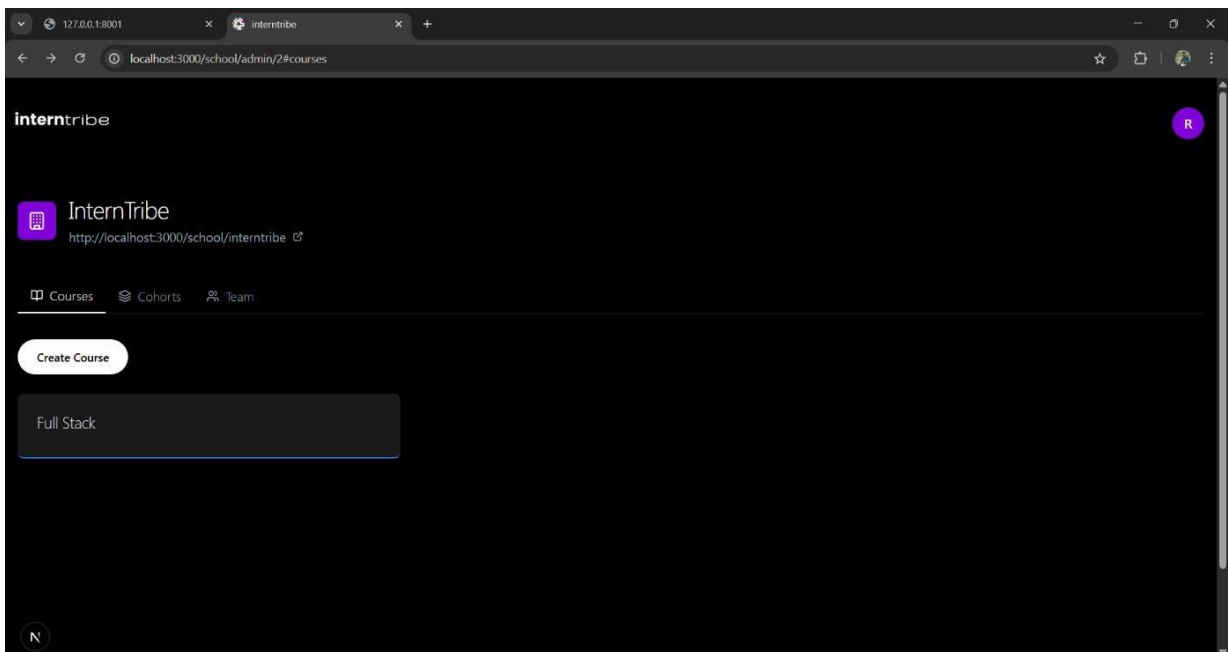


Figure 5.6 Course Created

5.2.5 Cohort Page

A cohort refers to a group of learners who begin and progress through a structured learning program together. This model is widely used in educational platforms to provide a collaborative and supportive environment for students

aiming to enhance specific skills over a defined time frame.

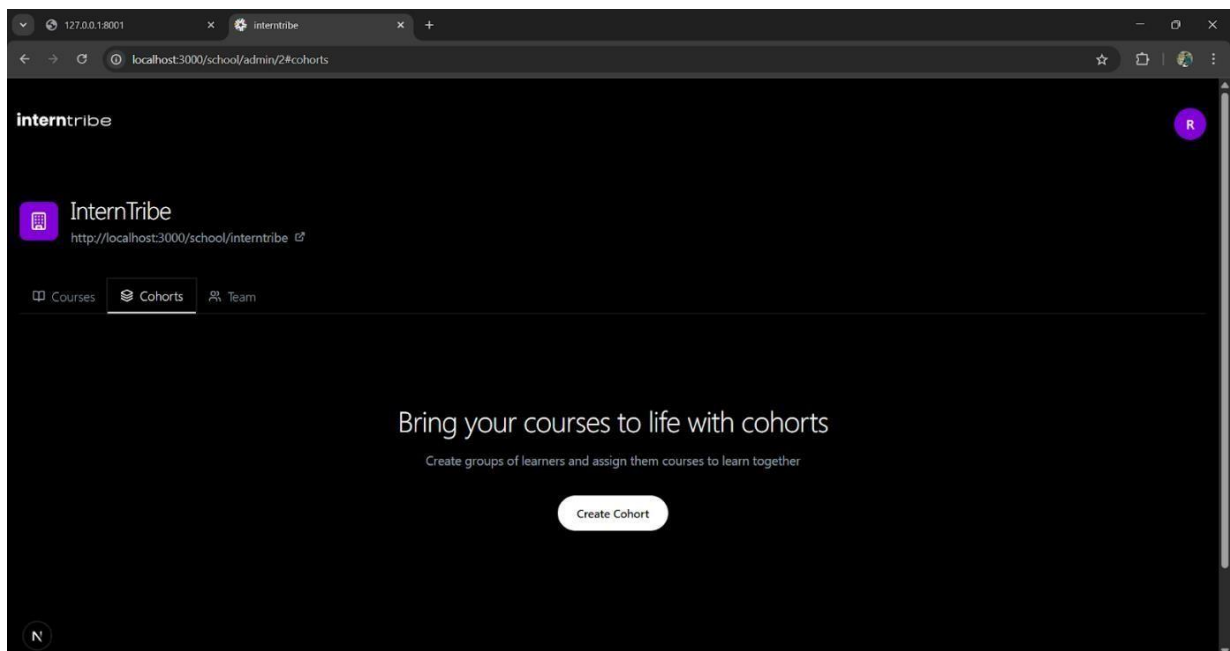


Figure 5.7 Cohort Management

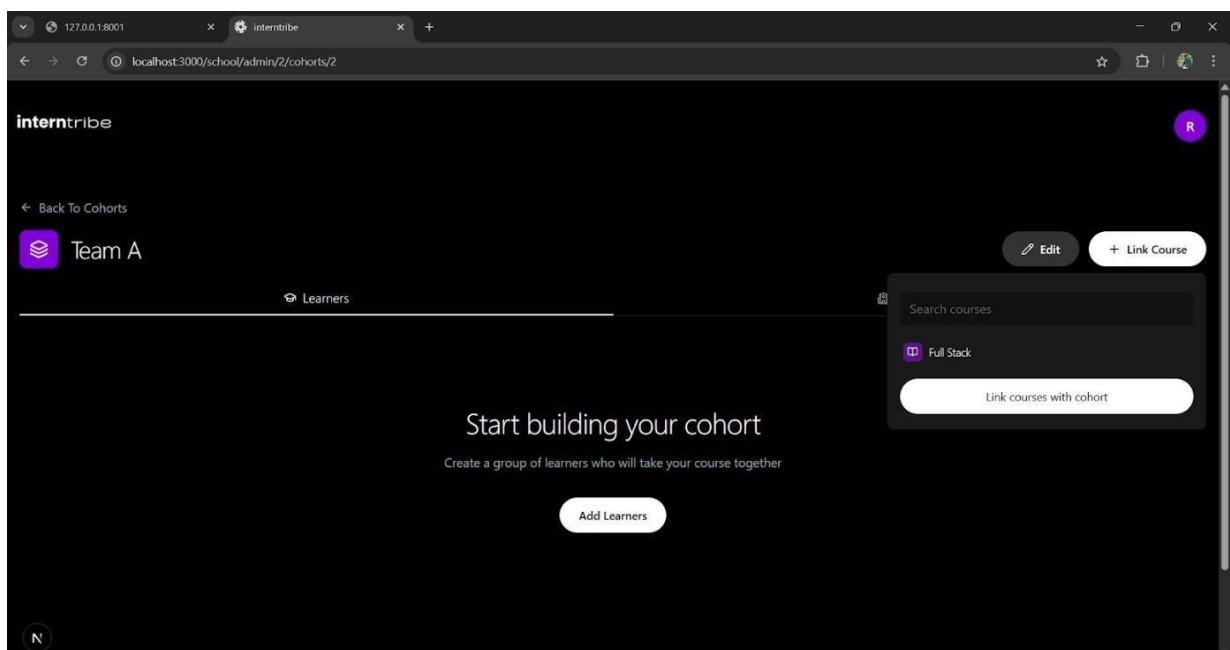


Figure 5.8 Assigning Course to Cohort

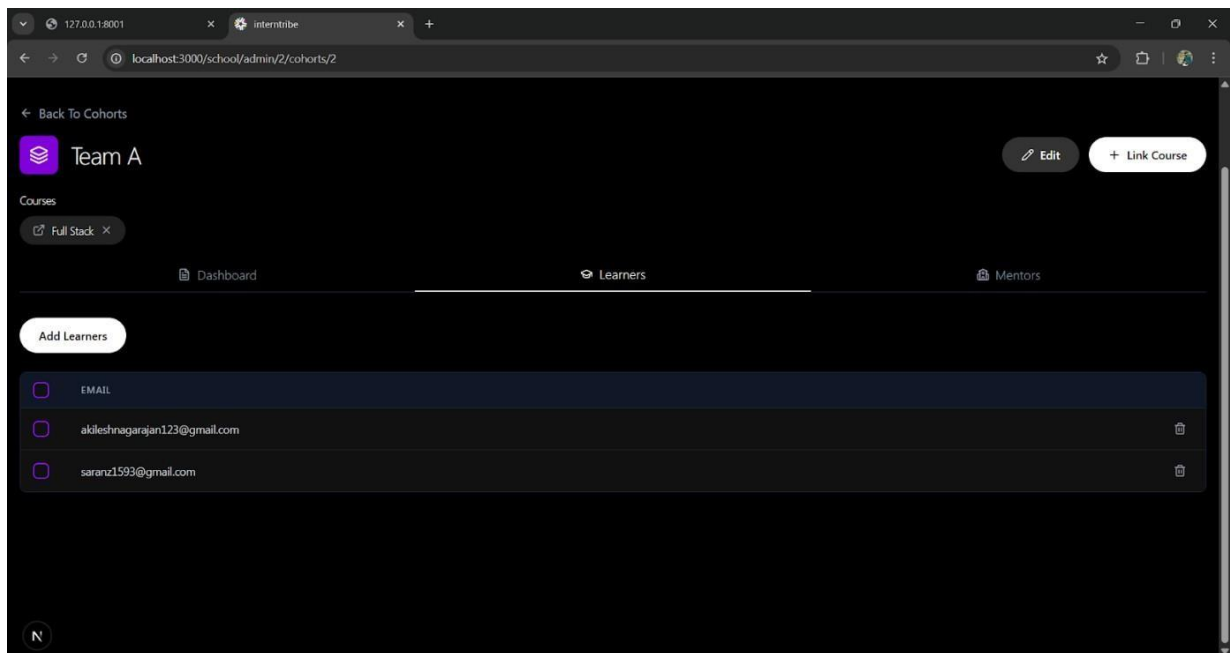


Figure 5.9 Adding Learners

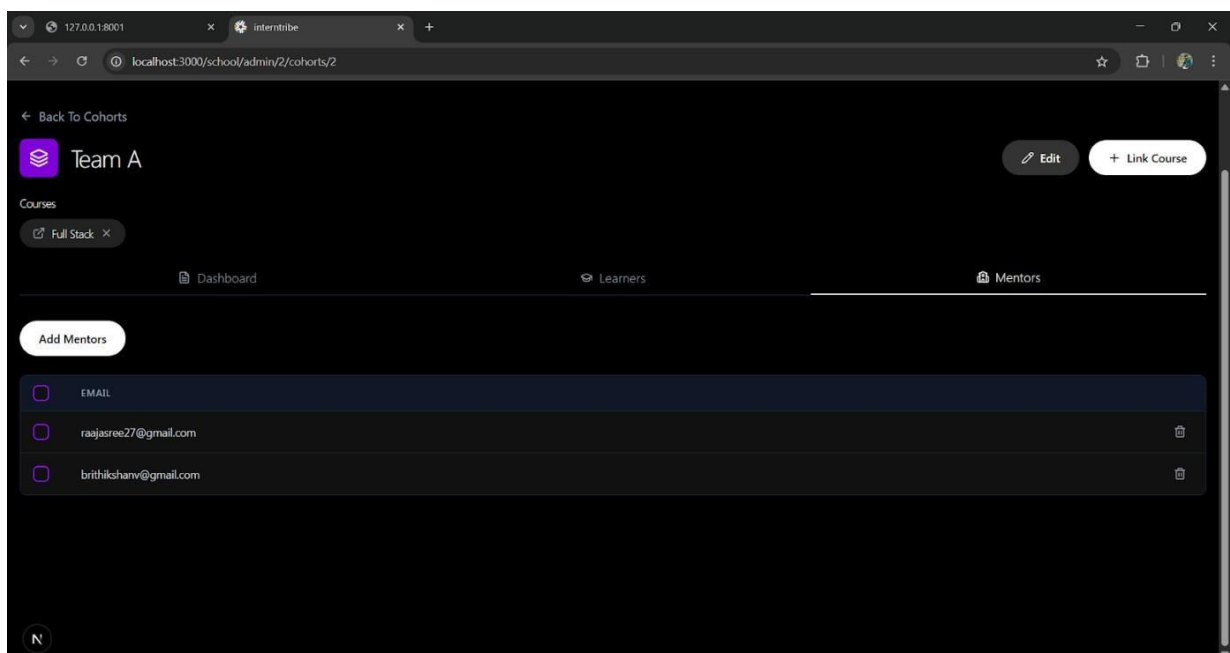


Figure 5.10 Adding Mentors

5.2.6 Team Management

Users added to an organization are automatically granted admin privileges. Admins have complete access to manage the organization's learning content, monitor user activities, and configure platform settings. While the system does

not explicitly categorize roles beyond admins, it supports a collaborative structure where multiple admins can work together. These team members may take on distinct responsibilities such as creating courses, monitoring cohorts, and evaluating learners, ensuring efficient and organized management of the learning platform.

5.2.7 Course Management

Course The Course Management system in the Admin Panel enables organizations to create, customize, and manage educational courses efficiently. Admins can add course details (title, description, duration, outcomes) and organize content modularly with videos, readings, quizzes, assignments, and projects. Courses are reusable across cohorts and categorized with tags for easy management. Scheduling options allow for weekly or custom timelines to guide learners. The system also offers performance insights to help mentors track engagement and enhance course effectiveness.

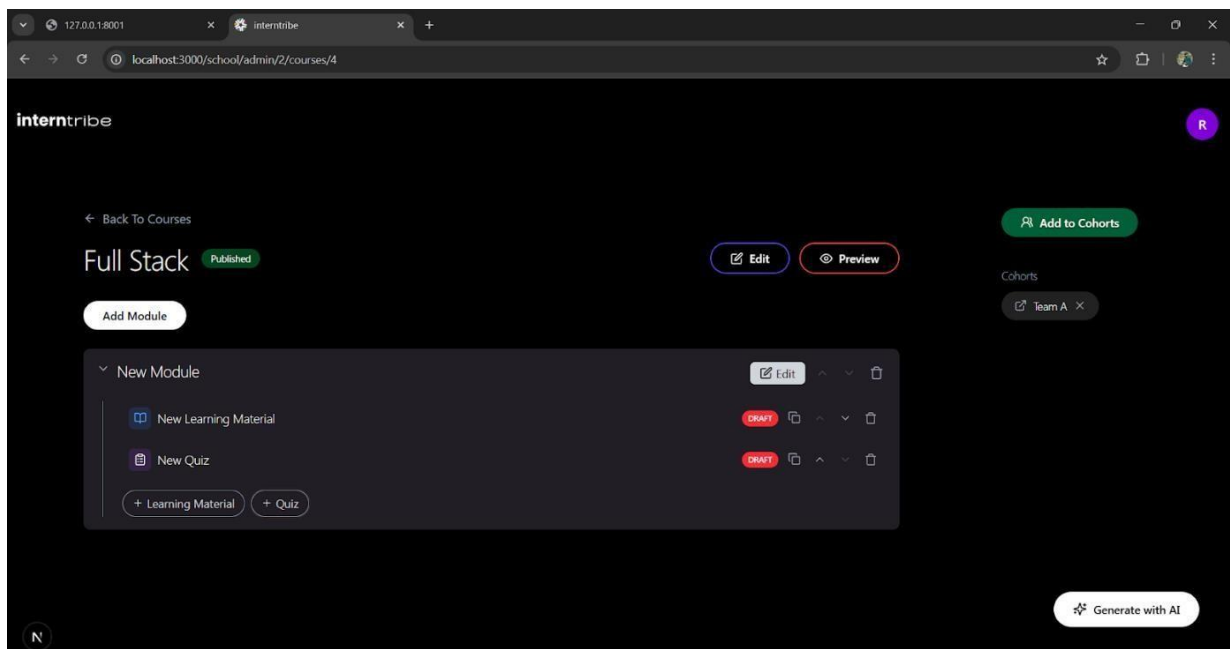


Figure 5.11 Course Page

Learning Material

Learning material refers to the structured educational content provided to

learners as part of their course or cohort experience. It plays a vital role in facilitating self-paced and guided learning. The platform offers a wide range of learning materials that cater to various learning styles and academic needs.

Quiz

Quizzes are an essential component of the learning system used to assess student understanding, reinforce key concepts, and identify learning gaps. The platform integrates quizzes directly into the learning experience, ensuring timely evaluation and feedback for both learners and mentors.

5.3 USER MODULE

The User Page serves as a personalized dashboard for each learner on the platform. It provides an overview of the learner's progress, activities, tasks, and performance across courses and cohorts. This centralized view helps students stay organized and engaged with their learning journey.

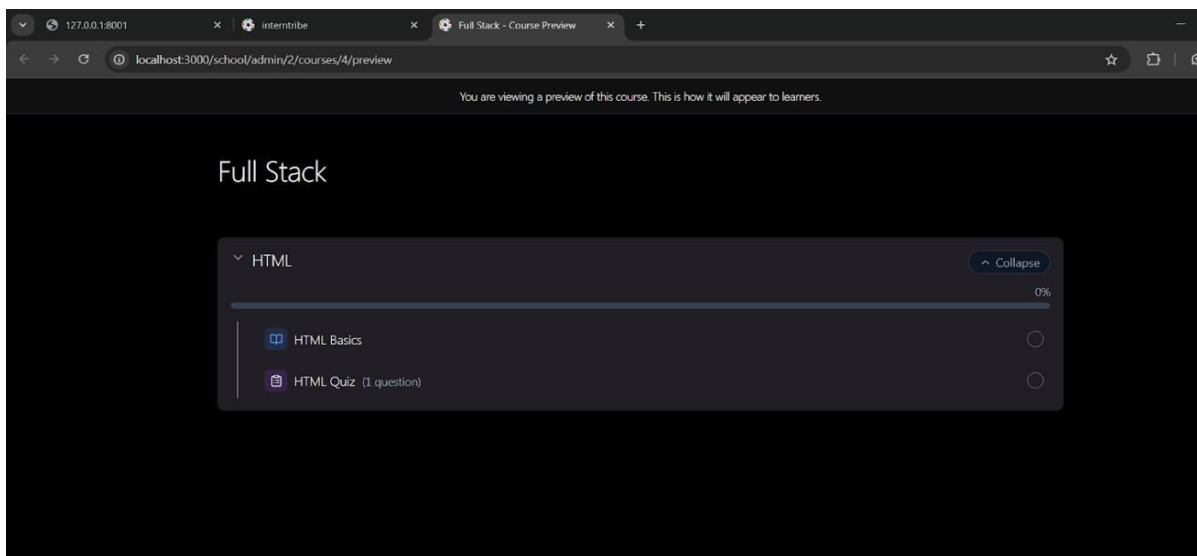


Figure 5.12 user Enrolled Course

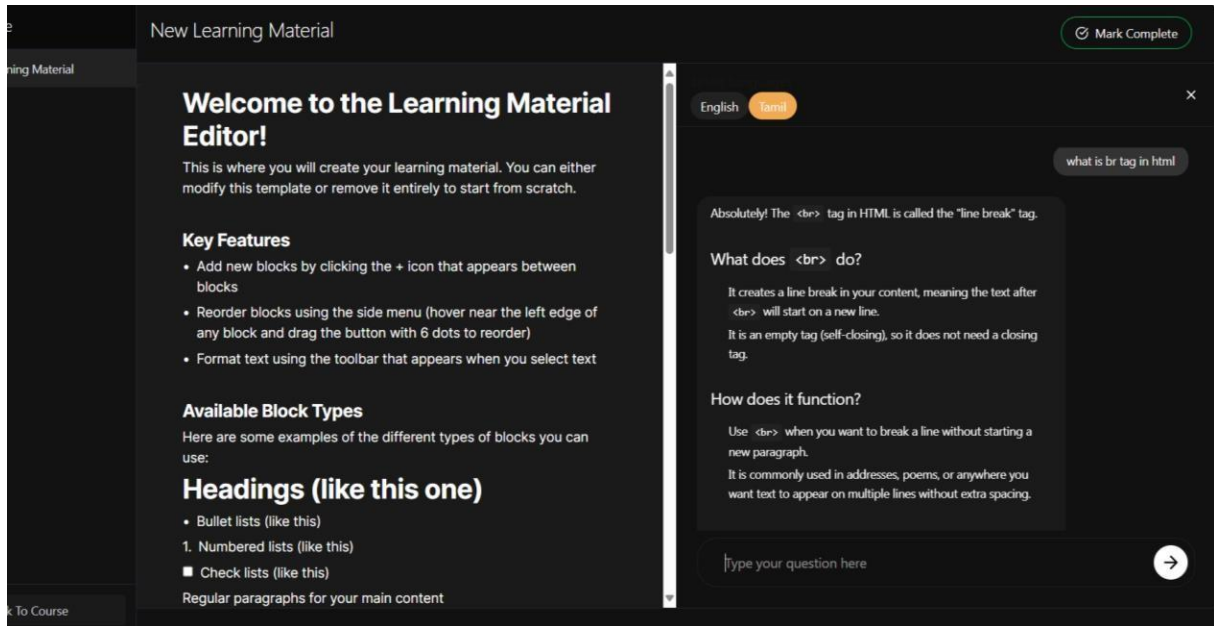


Figure 5.13 Learning Material with AI Assistance

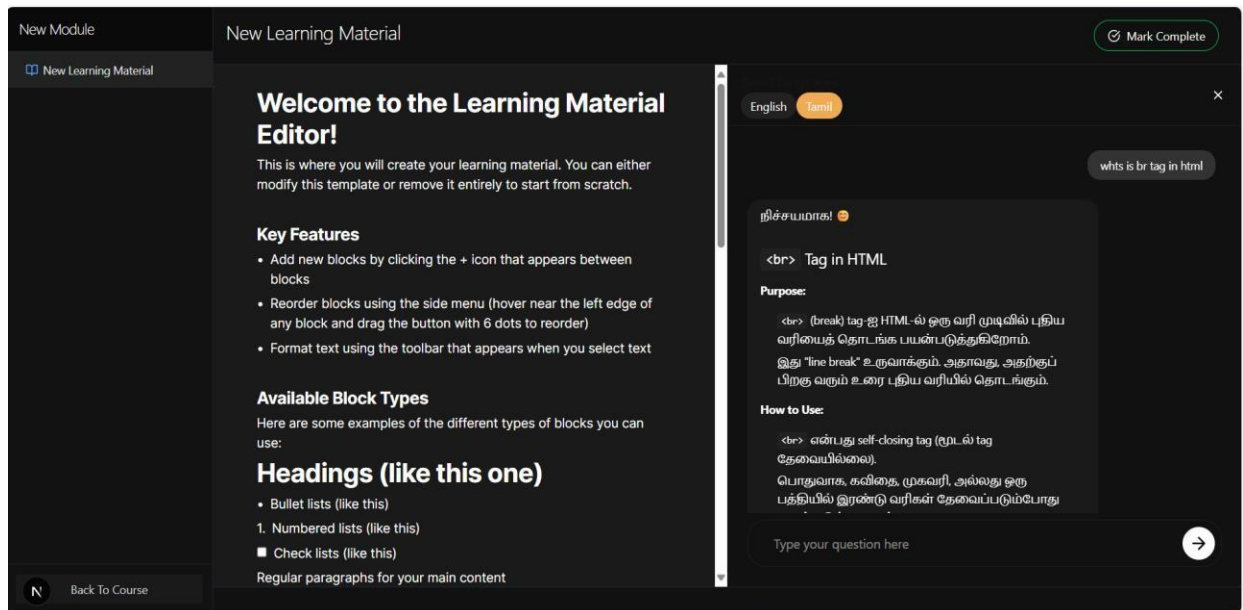


Figure 5.14 AI Assistance in Tamil

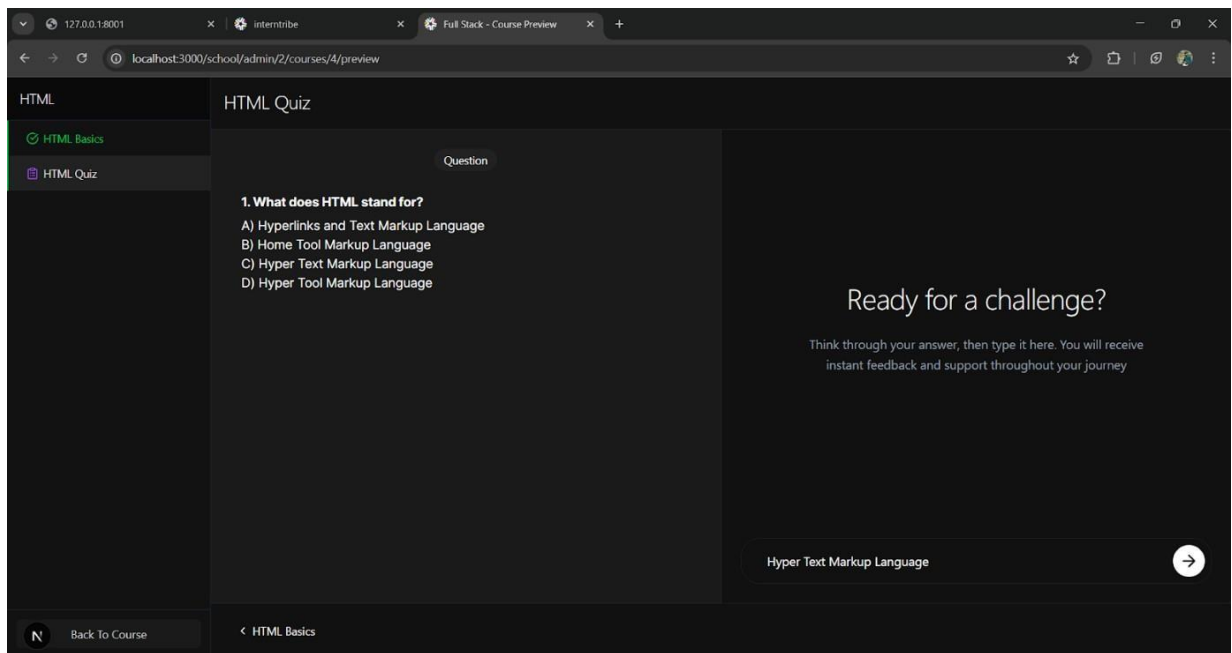


Figure 5.15 Quiz

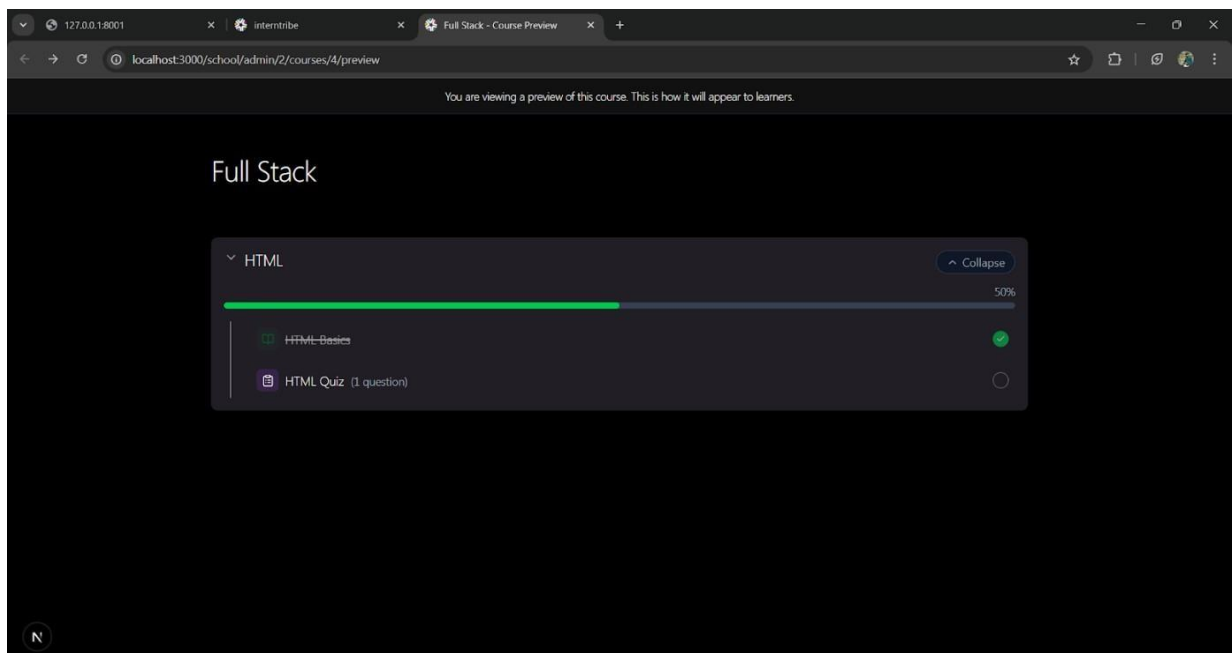


Figure 5.16 Chapter Completion

5.4 BACKEND MODULE

The Backend Module serves as the core of the platform's server-side operations, managing the flow of data and processing all business logic between the user interface and the database. It handles essential functions such as user authentication and authorization to ensure secure access based on roles. This module supports all data management tasks, including creating, reading, updating, and deleting information related to users, cohorts, tasks, quizzes, learning materials, and performance metrics. It provides a set of RESTful APIs that enable efficient and secure communication with the frontend, allowing seamless content delivery, task submissions, progress updates, and more. The backend also incorporates key logic for quiz evaluation, skill score calculation, and conditional content unlocking based on learner progress. Additionally, it collects and processes user activity data to generate progress reports and performance analytics.

5.5 SAMPLE CODE

```
@router.post("/chat")
async def ai_response_for_question(request: AIChatRequest):
    if request.task_type == TaskType.QUIZ:
        if request.question_id is None and request.question is None:
            raise HTTPException(
                status_code=400,
                detail=f"Question ID or question is required for {request.task_type}
tasks",
            )
        if request.question_id is not None and request.user_id is None:
            raise HTTPException(
                status_code=400,
```

```

        detail="User ID is required when question ID is provided",
    )

    if request.question and request.chat_history is None:
        raise HTTPException(
            status_code=400,
            detail="Chat history is required when question is provided",
        )
    else:
        if request.task_id is None:
            raise HTTPException(
                status_code=400,
                detail="Task ID is required for learning material tasks",
            )
        if request.chat_history is None:
            raise HTTPException(
                status_code=400,
                detail="Chat history is required for learning material tasks",
            )
        if request.task_type == TaskType.LEARNING_MATERIAL:
            task = await get_task(request.task_id)
            if not task:
                raise HTTPException(status_code=404, detail="Task not found")

            chat_history = request.chat_history

            reference_material = construct_description_from_blocks(task["blocks"])
            question_details = f"""\nReference
Material:\n```\n{reference_material}\n```"""
        else:

```

```

        if request.task_type == TaskType.LEARNING_MATERIAL:
            message["content"] = json.dumps({"feedback": message["content"]})

        message["content"] =
        get_ai_message_for_chat_history(message["content"])
    user_message = (
        get_user_audio_message_for_chat_history(request.user_response)
        if request.response_type == ChatResponseType.AUDIO
        else get_user_message_for_chat_history(request.user_response)
    )
    user_message = {"role": "user", "content": user_message}

    if request.task_type == TaskType QUIZ:
        if question["type"] == QuestionType.OBJECTIVE:
            answer_as_prompt =
            construct_description_from_blocks(question["answer"])
            question_details += f"""\n\nReference Solution (never to be shared with
the learner):\n```\n{answer_as_prompt}\n```\n"""
        else:
            scoring_criteria_as_prompt = ""
            for criterion in question["scorecard"]["criteria"]:
                scoring_criteria_as_prompt += f"""\n- **{criterion['name']}** [min:
{criterion['min_score']}, max: {criterion['max_score']}]:
{criterion['description']}\n"""
            question_details += (
                f"""\n\nScoring Criteria:\n```\n{scoring_criteria_as_prompt}\n```\n"""
            )

```

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

Our project introduced an AI-powered academic support system designed to overcome the limitations of traditional academic help, such as limited availability and lack of personalization. Built with React.js, FastAPI, and a fine-tuned OpenAI model, the system provides real-time, subject-specific assistance through a user-friendly interface. Its key features—AI chat, role-based access, and progress tracking—create a seamless learning experience. The modular architecture allows easy integration into existing platforms and reduces educator workload by automating common student queries. In summary, the system effectively delivers on-demand, personalized support, enhancing student learning, engagement, and independence.

6.2 FUTURE ENHANCEMENTS

While the current version of the system demonstrates promising results, several enhancements can be considered to further improve its functionality, adaptability, and impact:

- **Voice and Multilingual Support:** Integrating speech-to-text and multilingual capabilities will make the platform more inclusive and accessible, especially for students with reading difficulties or those who prefer native languages.
- **Adaptive Learning Engine:** Incorporating machine learning algorithms to analyze student interactions and tailor learning paths based on individual performance and behavior.
- **Gamification Features:** Adding elements like badges, leaderboards, and

daily learning streaks can increase student engagement and motivation.

- **Teacher and Peer Collaboration Tools:** Introducing collaborative features such as shared whiteboards, live group chats, or co-learning spaces to enable more interactive and community-driven learning.
- **Model Self-Improvement via Feedback Loop:** Automate the refinement of the AI model using highly rated responses and real user feedback to ensure continuous learning and accuracy improvements.
- **Mobile App Deployment:** Creating a native mobile application with offline mode support to expand accessibility for students with limited or intermittent internet access.
- **Data Analytics Dashboard for Educators:** Provide instructors and admins with a dashboard to monitor student progress, identify struggling learners, and make data-driven interventions.
- **Security Enhancements:** Implement advanced security measures such as two-factor authentication and anomaly detection to further protect user data and prevent misuse.

By implementing these future enhancements, the system can evolve into a comprehensive, intelligent learning companion that meets the diverse needs of students and institutions across different educational landscapes.

CHAPTER 7

REFERENCES

1. Chen, Y., H. Chen and S. Su, 2023, "Fine-Tuning Large Language Models in Education", 13th International Conference on Information Technology in Medicine and Education (ITME), Wuyishan, China.
2. Divasón, J., F. J. Martínez-de-Pisón, A. Romero and E. Sáenz-de-Cabezón, 2023, "Artificial Intelligence Models for Assessing the Evaluation Process of Complex Student Projects", IEEE Transactions on Learning Technologies, vol. 16, no. 5, pp. 694-707.
3. Galhotra, B. and D. Lowe, 2022, "AI Based Examination System: A Paradigm Shift in Education Sector", International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON), Faridabad, India.
4. Han, J., J. Lu, Y. Xu, J. You, and B. Wu, 2024, "Intelligent Practices of Large Language Models in Digital Government Services", IEEE Access, vol. 12, pp. 8633-8640.
5. Li, M. Z., 2024, "Using Prompt Engineering to Enhance STEM Education", IEEE Integrated STEM Education Conference (ISEC), Princeton, NJ, USA.
6. Mon, B. F., A. Wasfi, M. Hayajneh and A. Slim, 2023, "A Study on Role of Artificial Intelligence in Education", International Conference on Computing, Electronics & Communications Engineering (iCCECE),

Swansea, United Kingdom.

7. Murtaza, M., Y. Ahmed, J. A. Shamsi, F. Sherwani and M. Usman, 2022, "AI-Based Personalized E-Learning Systems: Issues, Challenges, and Solutions".
8. Rekha, A. K., K. Gopal, D. Satheeskumar, U. A. Anand, D. S. S. Doss and S. Elayaperumal, 2024, "Ai-Powered Personalized Learning System Design: Student Engagement and Performance Tracking System", 4th International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India.
9. Sabar, A., S. Gamage, S. Abishek, C. Lakmal, K. A. D. T. Kulawansa and P. Perera, 2024, "AI-Powered Student Learning System", 8th SLAAI International Conference on Artificial Intelligence (SLAAI-ICAI), Ratmalana, Sri Lanka.
10. Ubah, A. E., et al., 2022, "A Review of Artificial Intelligence in Education", International Conference on Artificial Intelligence of Things and Crowdsensing (AIoTCs), Nicosia, Cyprus.