IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

AI - BASED LEARNING MANAGEMENT SYSTEM

[1] Prof. Vijay Shankar , [2]Vaibhav Yewale, [3]Shivam Mourya, [4]Prathamesh Gate, [5]Khushi Parakh

Department of Artificial Intelligence and Data Science,
Rizvi College of Engineering, Mumbai, India.

Abstract: The rapid evolution of technology has led to a growing demand for personalized, efficient learning solutions. In response, this project proposes an AI-Based Learning Management System (AI-LMS) that caters to three distinct user profiles: students or learners, teachers, and hiring managers. The system's primary feature is its ability to predict and generate a personalized learning roadmap for a student based on the subject they enter and the time they are willing to dedicate to learning. The AI-LMS optimizes learning by analyzing the input subject and time constraints to produce a three-step learning path, helping students achieve their goals within the given time frame.

The AI-LMS stands out by focusing on customized learning paths rather than a one-size-fits-all curriculum. This feature provides learners with specific learning plans tailored to their individual needs, making it more engaging and effective. AI uses data from various sources, such as the student's past learning behavior, subject difficulty, and learning time constraints, to predict a roadmap that students can follow. The three-step learning roadmap breaks down the learning process into manageable phases: beginner, intermediate, and advanced levels, ensuring that learners move through the material at an optimal pace.

In addition to learners, the system also provides significant value for teachers. Teachers can supervise students, monitor their progress, and provide real-time feedback based on the roadmap. This supervisory role empowers teachers to identify struggling students, intervene when necessary, and offer customized guidance to ensure that each student stays on track. Teachers are also able to modify learning plans, if needed, to better suit the learner's capabilities, providing a dynamic and flexible learning environment.

Keywords: AI-Based Learning, (LMS) Learning Management System, Personalized Learning Roadmap, Roadmap Prediction, Roadmap Generator, Time-Specific Learning, Student Profile, Teacher Supervision, Recruiter Access, User Authentication, Dynamic Learning Updates, Roadmap Adjustment, Learning Progress Monitoring, Student Portfolio, Adaptive Learning, Data Security, Role-Based Access Control, Student-Led Learning, Teacher-Led Supervision, Employment Integration. Genmap AI Learning tool, Elapsed Time.

I. INTRODUCTION

In today's rapidly evolving educational landscape, traditional learning models often fall short in meeting the unique needs of individual learners. Students have different paces, interests, and levels of understanding, which highlights the need for personalized and adaptive learning systems. An AI-Based Learning Management System (AI-LMS) offers a modern solution by using artificial intelligence to create customized learning paths that align with each student's goals and time availability.

The AI-LMS is designed for three main user groups: students, teachers, and hiring managers. Students receive personalized roadmaps broken into beginner, intermediate, and advanced stages, helping them progress efficiently through a subject. The system dynamically adjusts the learning path based on a student's

performance, providing extra support or accelerating progress as needed. Teachers monitor and guide students, while hiring managers can review authenticated student portfolios to identify skilled candidates.

Security and data privacy are central to the AI-LMS. All users must register and authenticate before accessing the system, ensuring that each user only sees the data relevant to their role. Advanced security features such as data encryption and secure access controls protect user information, allowing students, teachers, and hiring managers to interact with the platform safely and confidently.

II. LITERATURE SURVEY

2.1 Survey of Existing Systems:

The integration of Artificial Intelligence (AI) in education has gained significant traction, transforming how learners interact with content and how educators deliver instruction. Traditional Learning Management Systems (LMS), such as Moodle and Blackboard, are widely used but offer limited personalization. These systems often follow a static, one-size-fits-all learning path, which fails to accommodate different learning styles, speeds, and preferences (Graesser et al., 2018).

Modern AI-based educational platforms have emerged to overcome these limitations. For instance, Intelligent Tutoring Systems (ITS) and adaptive learning platforms like Knewton and DreamBox use AI to customize learning content. Research by Kardan and Conati (2013) and Aleven et al. (2016) supports the effectiveness of these systems in improving student outcomes through adaptive pathways.

Recent studies (Spector et al., 2023; Zhang et al., 2024) demonstrate the use of deep learning and multimodal data to further enhance personalization, such as adjusting content based on facial expressions or real-time engagement. These systems allow for better learner modelling, real-time feedback, and predictive learning trajectories, making education more responsive and individualized.

Digital portfolios are also being widely adopted for showcasing student competencies. Research by Barrett (2010) and Clarke and Boud (2018) shows that student portfolios are becoming key tools in connecting education to employment, particularly when supported by AI analytics and secure access control mechanisms.

2.2 Limitations of the Existing Systems:

Despite progress, several limitations persist in current AI-education platforms:

- Lack of Real-Time Roadmap Generation: Most existing systems provide adaptive learning content but do not generate personalized roadmaps based on user-defined constraints such as available time or goals.
- Low Integration Between Education and Employment: Many LMS platforms are still not integrated with features that support hiring decisions through authenticated access to student progress or digital portfolios.
- Limited Flexibility in Traditional LMS: Conventional LMS platforms do not adapt learning paths dynamically in response to a student's performance in real-time.
- Minimal Teacher Involvement in AI Recommendations: AI suggestions often function independently of teacher feedback, which limits holistic learning adjustments**.**

2.3 Problem Statement and Objectives:

2.3.1 Problem Statement:

Traditional LMS platforms lack personalization and adaptability, failing to meet diverse learner needs and provide a guided, goal-oriented learning experience. There is also a disconnect between educational outcomes and employment visibility.

2.3.2 Objectives:

- To design and develop an AI-Based Learning Management System (AI-LMS) that generates personalized learning roadmaps based on subject, learner performance, and available study time.
- To implement dynamic adaptability in the learning path using real-time performance feedback.

- To involve teachers in the supervision and adjustment of AI-generated learning plans.
- To facilitate authenticated access for hiring managers to view student portfolios, bridging the gap between education and employment.

2.4 Scope of the Project:

The AI-LMS aims to:

- Support three user profiles: students, teachers, and hiring managers.
- Generate personalized learning plans segmented into beginner, intermediate, and advanced levels.
- Dynamically adjust learning content based on learner performance.
- Allow teachers to monitor and modify learning paths and assess student progress.
- Provide authenticated hiring managers access to student portfolios showcasing achievements and skill levels.
- Ensure user authentication and data privacy through secure access control and encryption.

III. PROPOSED SYSTEM

3.1 System Overview:

The proposed AI-Based Learning Management System (AI-LMS) is designed to overcome the limitations of traditional LMS platforms by offering personalized, adaptive, and time-bound learning roadmaps. The system architecture follows a modular, AI-driven framework with three core components:

- User Management Module (Students, Teachers, Hiring Managers)
- AI Learning Roadmap Generator
- Performance Monitoring and Adaptive Engine
- Portfolio Management and Access Control System.

3.3 Hardware and Software Requirements:

3.3.1 Hardware Requirements:

Below are the Hardware Requirements for the Model to run Locally

Table 1: Hardware Requirements

Component	**Specification (For Laptop/Desktop)**	**Specification (For Mobile/Tablet)**
Processor	Multi-core CPU (Intel i3 / AMD Ryzen 3) or higher	Supports all processors
RAM	8 GB Minimum	4GB Minimum
System	Windows 10/Linux Ubuntu/MacOS (64-bit)	Android/IOS
Storage	256 GB Minimum SSD or HDD	64GB Minimum

3.3.2 Software Requirements:

Below are the Software Requirements for the Model to run Locally

Table 2: Software Requirements

Category	**Description**
IDE	Visual studio code and AI studio by Google
Language	Python, Node.js, Typescript
Web Framework	Node.js, Typescript, React.js, Tailwind CSS
LLM Integration	Gemini-1.5-Flash fine-tuned for Roadmap generation
Database Supported	MongoDB
Package Manager	Node Package Manager and PIP for installing dependencies
Version Control	GitHub – for Collaborative development process and version tracking

3.4 Methodology:

The project follows an **Agile Development Approach**, broken down into the following phases:

1. **Requirement Analysis** – Identify user roles and learning objectives.
2. **System Design** – Define architecture, database, and modules.
3. **AI Model Integration** – Train models to predict roadmaps based on learning data.
4. **Implementation** – Develop and integrate frontend, backend, and AI.
5. **Testing** – Validate roadmap accuracy, user access levels, and adaptability.
6. **Deployment** – Launch platform with secure login and monitoring tools.
7. **Evaluation & Feedback** – Collect user input for iterative improvement.

IV. SYSTEM DESIGN AND IMPLEMENTATION DETAILS

4.1 SYSTEM ARCHITECTURE:

The AI-Based Learning Management System (AI-LMS) follows a three-tier architecture model consisting of:

1. Presentation Layer (Frontend):
   - Built using React.js and Tailwind CSS
   - Responsible for UI/UX interactions for students, teachers, and hiring managers
2. Application Layer (Backend):
   - Developed using Node.js and Express.js
   - Handles business logic, user authentication, API requests, and AI integration
3. Data Layer (Database):
   - Utilizes MongoDB for storing user data, learning progress, roadmaps, and portfolios
   - Includes integration with AI models for dynamic updates.

4.2 Architectural Components:

4.2.1 User Interface (UI)

The User Interface (UI) layer serves as the primary interaction point between the end-users and the system. Built using modern frontend technologies such as React.js and Tailwind CSS, this layer is responsible for rendering intuitive and responsive dashboards tailored to different user roles—students,

teachers, and hiring managers. Students use the interface to input their desired subject and available study time, view their personalized learning roadmaps, complete tasks and assessments, and monitor their progress. Teachers interact with this layer to oversee student performance, provide feedback, and make manual adjustments to learning plans.

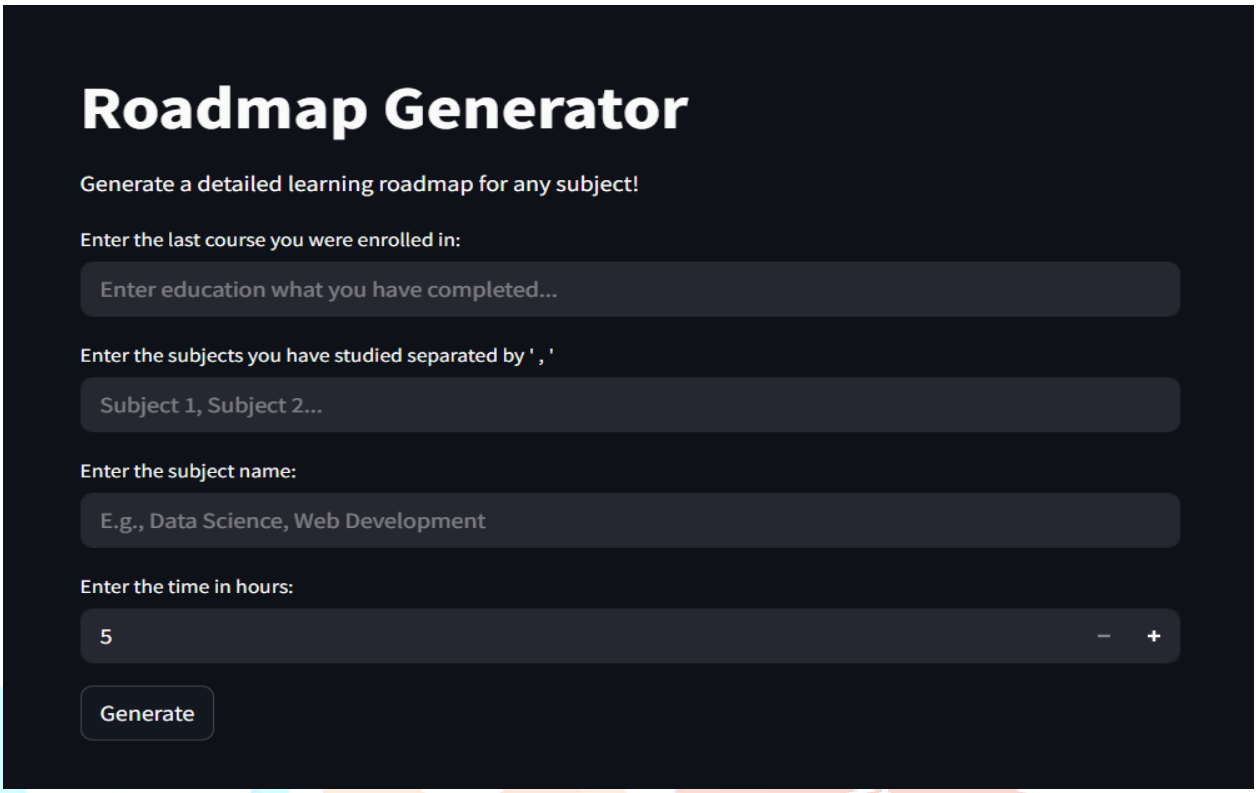

Fig. 1: Shows Streamlit Web Interface UI

4.2.2 Application Logic layer

The Application Logic layer, developed using Node.js and Express.js, constitutes the system's backbone. It handles routing, data validation, role-based access control, and manages communication between the frontend, database, and the AI engine. This layer ensures that user inputs are processed efficiently, learning roadmaps are dynamically generated, and that proper business rules are enforced. For instance, only authenticated teachers can access student performance data, and hiring managers can only view verified student portfolios after passing the system's security checks.

4.2.3 AI Engine or Personalization Module

The AI Engine or Personalization Module forms the heart of the adaptive learning system. Developed using Python and machine learning libraries such as TensorFlow or Scikit-learn, this component is responsible for generating personalized learning roadmaps for students. When a student inputs the subject they wish to study and the amount of time they have, the AI engine evaluates this data, along with historical learning performance (if available), to create a three-phase roadmap comprising beginner, intermediate, and advanced levels.

4.2.4 Database layer

The Database layer, implemented using MongoDB, provides the system's data persistence and management infrastructure. It houses several collections corresponding to user profiles, personalized roadmaps, learning progress, assessments, and portfolios. The non-relational structure of MongoDB supports the dynamic and evolving nature of the educational data, enabling efficient querying and storage of structured and semi-structured data.

4.2.5 Authentication and Security layer

Security and data privacy are paramount in any system that handles sensitive educational data. The Authentication and Security layer utilizes robust authentication frameworks such as Firebase Auth or Auth0, combined with JSON Web Tokens (JWT), to secure access to the platform. This component ensures that users are properly verified and that they can only access information relevant to their roles.

4.2.6 Monitoring and Feedback Component

It supports continuous system improvement and user engagement. This layer tracks user interactions, system performance, and behavioral metrics to provide feedback for both users and administrators. It collects student responses, completion rates, and system usage statistics, which are then used to refine the AI algorithms and enhance roadmap prediction accuracy. Administrators can access detailed reports through an admin dashboard, allowing them to monitor system health, address performance issues, and receive suggestions for future development based on user feedback.

V. IMPLEMENTATION DETAILS AND FINAL COST

5.1 Implementation Methodology:

The development of the AI-Based Learning Management System (LMS) was approached using an iterative and modular methodology, ensuring adaptability and integration of intelligent features with traditional e-learning functionalities. The implementation was divided into sequential stages, each addressing a specific aspect of the system design and deployment.

Initially, a comprehensive requirement analysis was conducted through stakeholder interviews, surveys, and market studies to identify user expectations, pain points, and desired outcomes. Based on the insights gathered, a functional specification document was created, guiding the subsequent design phase.

The system architecture was then conceptualized, emphasizing scalability, modularity, and support for artificial intelligence components. The technology stack was selected accordingly—comprising a React-based frontend for responsive interaction, a Node.js and Express.js backend for handling logic and APIs, MongoDB for efficient data storage, and Python-based AI models (leveraging libraries such as TensorFlow and scikit-learn). AI integration was achieved through microservices, allowing language models and recommendation systems to function independently and communicate through REST APIs.

5.2 Technical and operational challenges:

Throughout the development process, multiple technical and operational challenges were encountered. A significant obstacle was the availability and quality of training data for machine learning algorithms, which directly affected the performance of the recommendation system and chatbot. This issue was addressed by preprocessing existing datasets, simulating user interaction data, and adopting semi-supervised learning techniques where applicable.

Another major challenge was integrating AI models—primarily built in Python—with the Node.js-based LMS infrastructure. To resolve compatibility issues, the models were encapsulated as independent services with clearly defined API interfaces.

VI. TESTING AND RESULTS

6.1 Testing

The testing of an AI-based Learning Management System (LMS) involves a comprehensive strategy that addresses both conventional software functionality and the unique challenges introduced by integrated artificial intelligence components. The objective is to ensure that the system is reliable, efficient, user-friendly, and capable of delivering accurate, adaptive learning experiences.

6.1.1 Functional Testing

Functional testing is carried out to verify that the core features of the LMS operate in accordance with the defined requirements.

This includes modules such as user registration and authentication, course creation, content delivery, assessments, grading, and

progress tracking. Manual and automated testing approaches are employed to simulate user interactions and confirm the correct

execution of workflows. Tools such as Selenium or Cypress may be used to facilitate automated testing of the user interface

and backend logic.

6.1.2 AI Component Testing

AI functionalities within the LMS—such as personalized course recommendations, intelligent tutoring systems, automated

grading, and chatbots—are subjected to specialized testing methodologies. The primary focus is on evaluating the performance,

accuracy, fairness, and explainability of machine learning models embedded in the system.

6.2 Results:

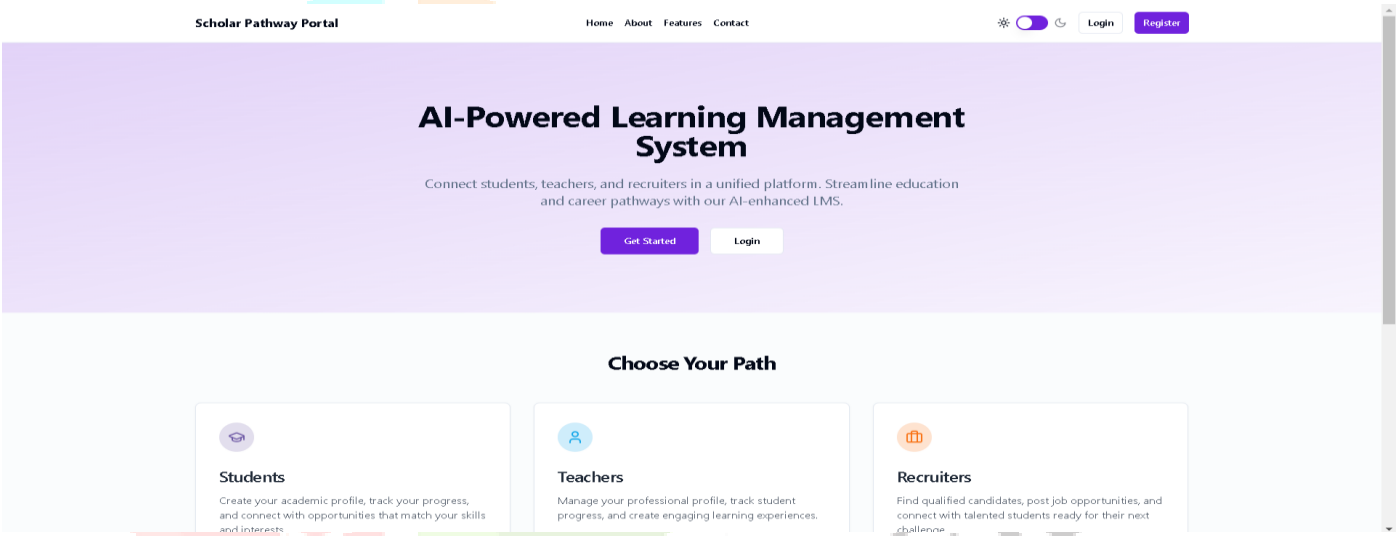

Fig. 2: Home Page

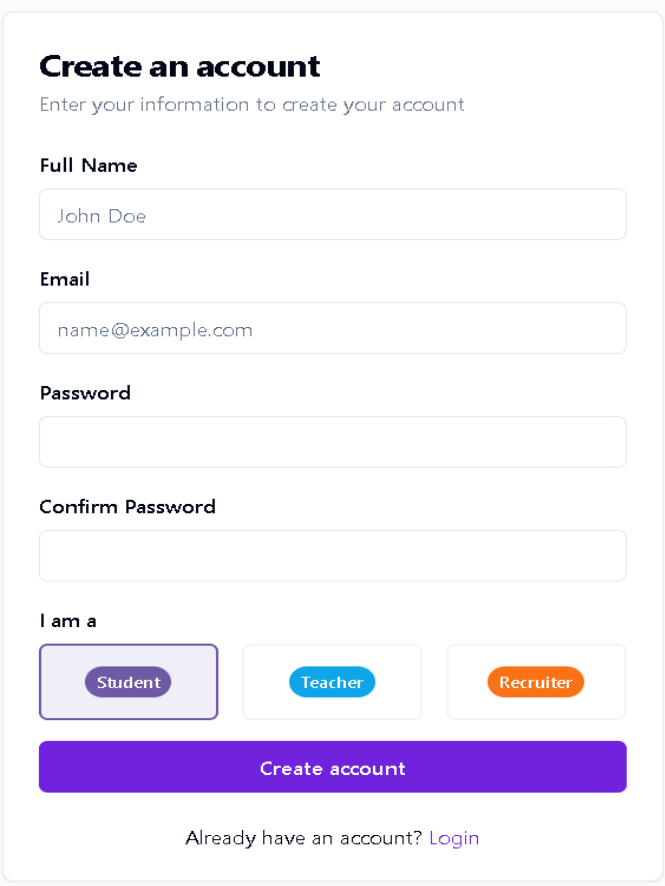

Fig. 3: Registration page

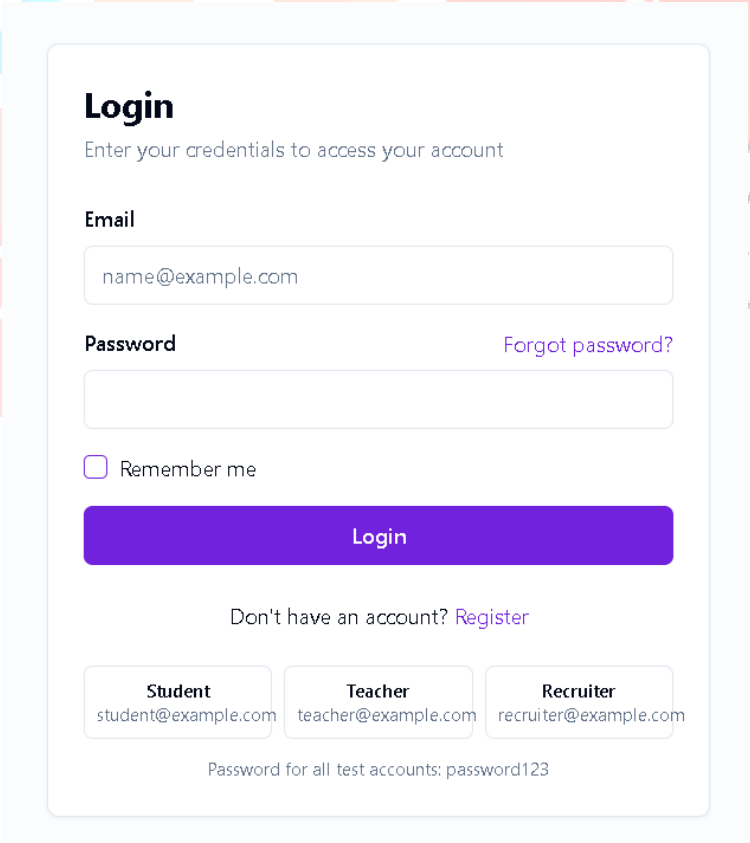

Fig. 4: login page

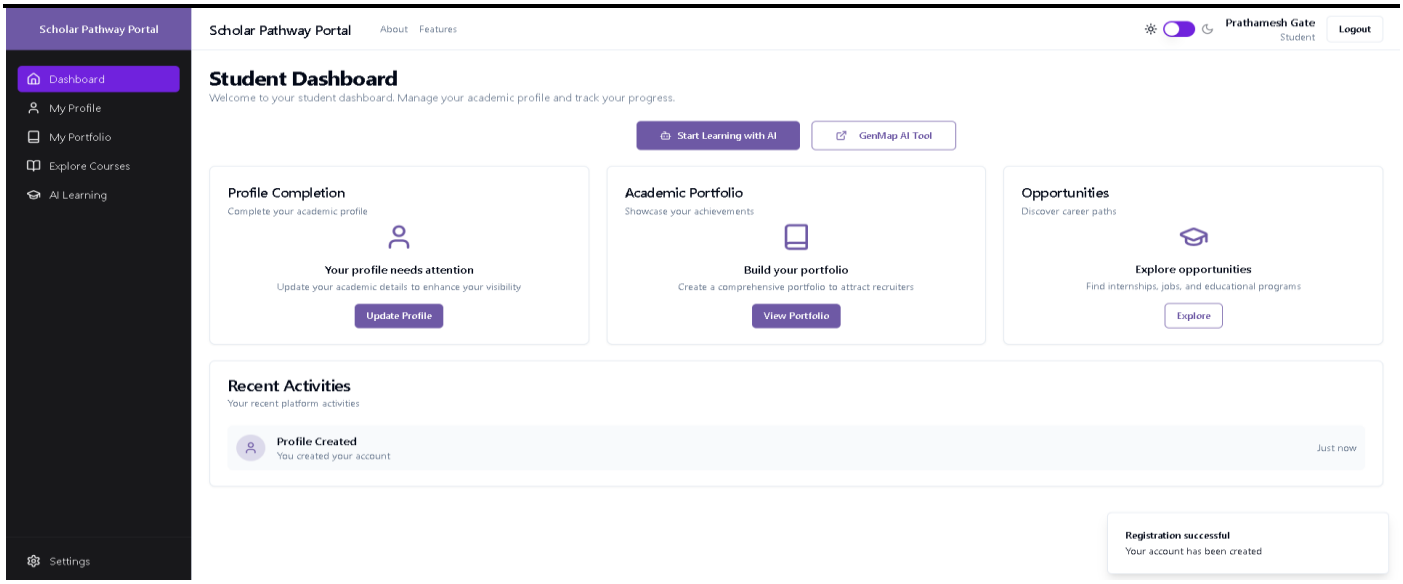

Fig. 5: Student dashboard page

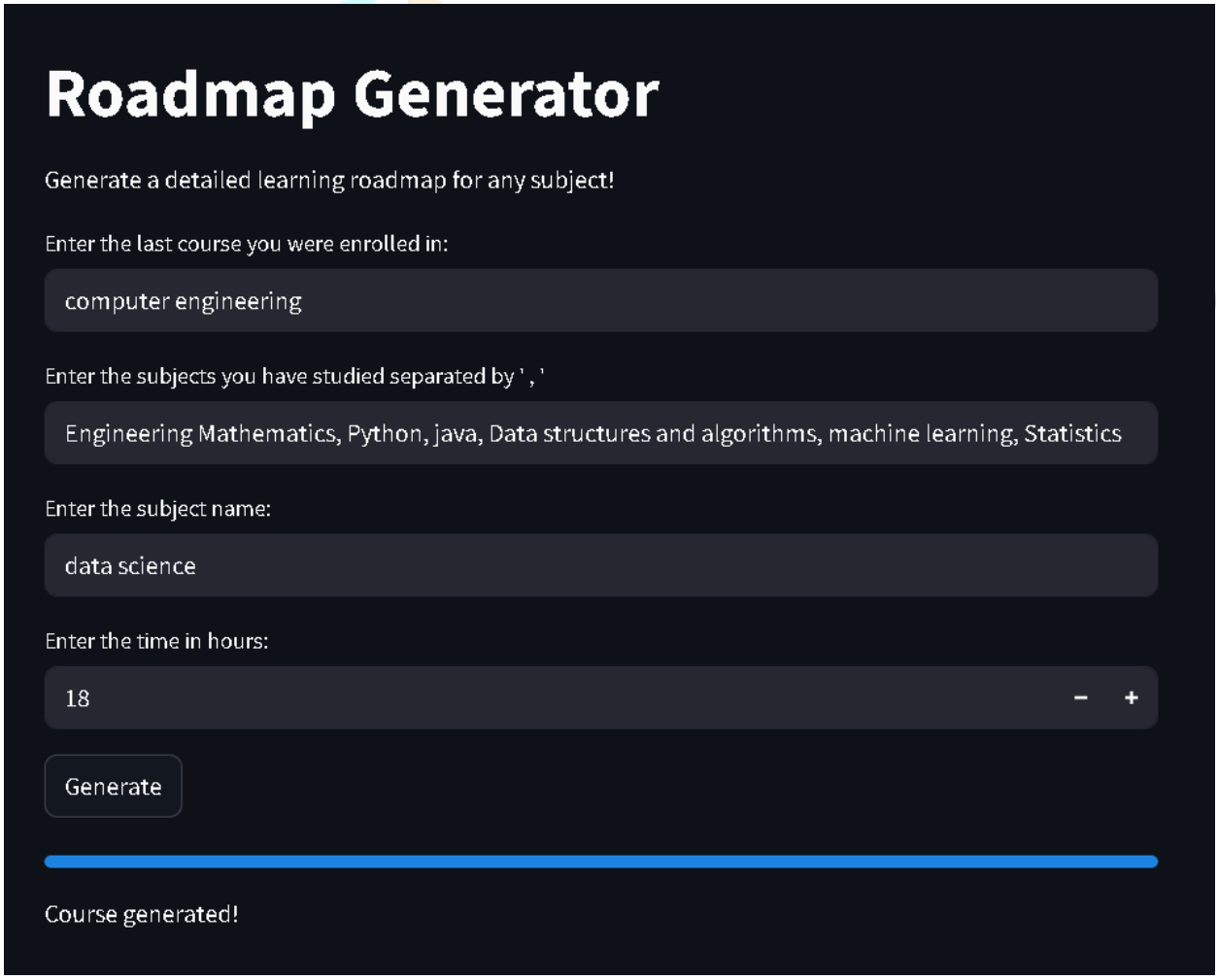

Fig. 6: Student's Roadmap Generator page (Genmap AI tool)

Fig. 7: Student's Roadmap Generator page's response

Module Titles	Elapsed time	Topics	YouTube Link
Data Science Foundations	4 hours	Introduction to Data Science and its applications The Data Science Lifecycle (CRISP-DM) Types of Data (Structured, Unstructured, Semi-structured) Data Wrangling and Cleaning techniques (handling missing values, outliers) Exploratory Data Analysis (EDA) with Python (visualization libraries like Matplotlib, Seaborn)	Watch here
Data Manipulation and Preprocessing with Python	3 hours	Pandas for data manipulation (DataFrame operations, data filtering, aggregation) Data cleaning and transformation using Pandas Feature Engineering techniques (scaling, encoding, dimensionality reduction) Handling imbalanced datasets	Watch here

Fig. 8: Student's Generated Roadmap

Statistical Modeling and Inference	4 hours	Review of descriptive and inferential statistics Hypothesis testing (t-tests, ANOVA) Regression analysis (linear, logistic) Model evaluation metrics (R-squared, RMSE, AUC)	Watch here
Machine Learning Fundamentals	5 hours	Supervised Learning (Regression, Classification) Unsupervised Learning (Clustering, Dimensionality Reduction) Model Selection and Hyperparameter Tuning (Cross-validation) Introduction to scikit-learn library	Watch here
Advanced Machine Learning Techniques	2 hours	Ensemble methods (Random Forest, Gradient Boosting) Support Vector Machines (SVM) Deep Learning Introduction (Neural Networks, TensorFlow/Keras - basic concepts) Model deployment considerations	Watch here

Fig. 9: Student's Generated Roadmap

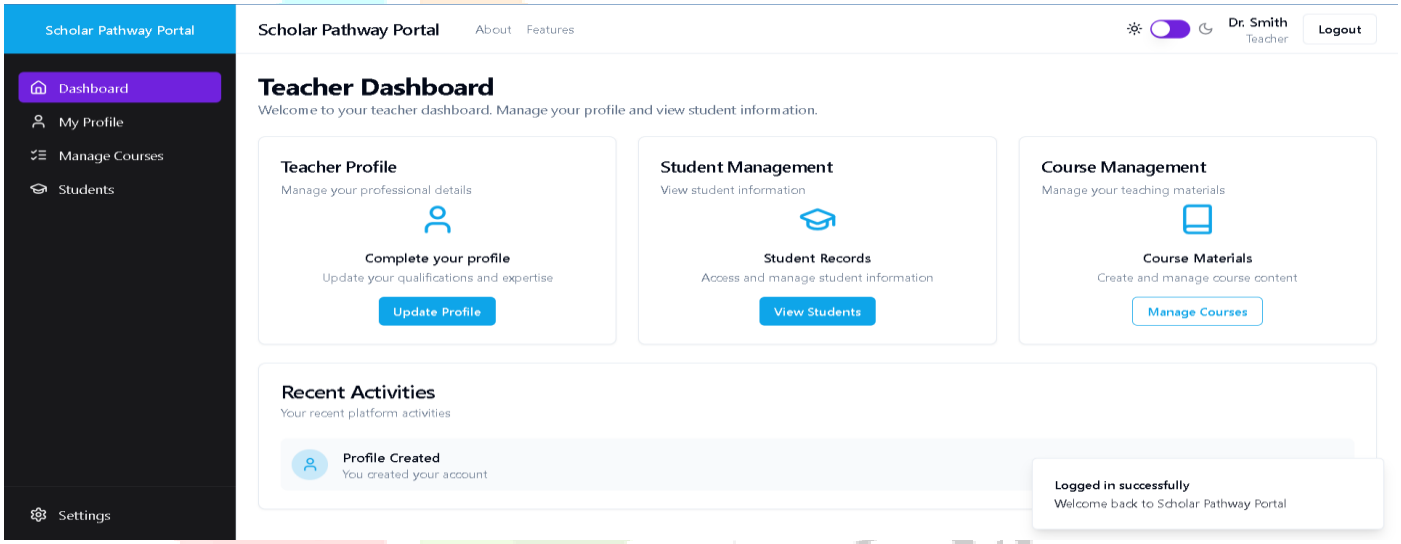

Fig. 10: Teacher's Dashboard page

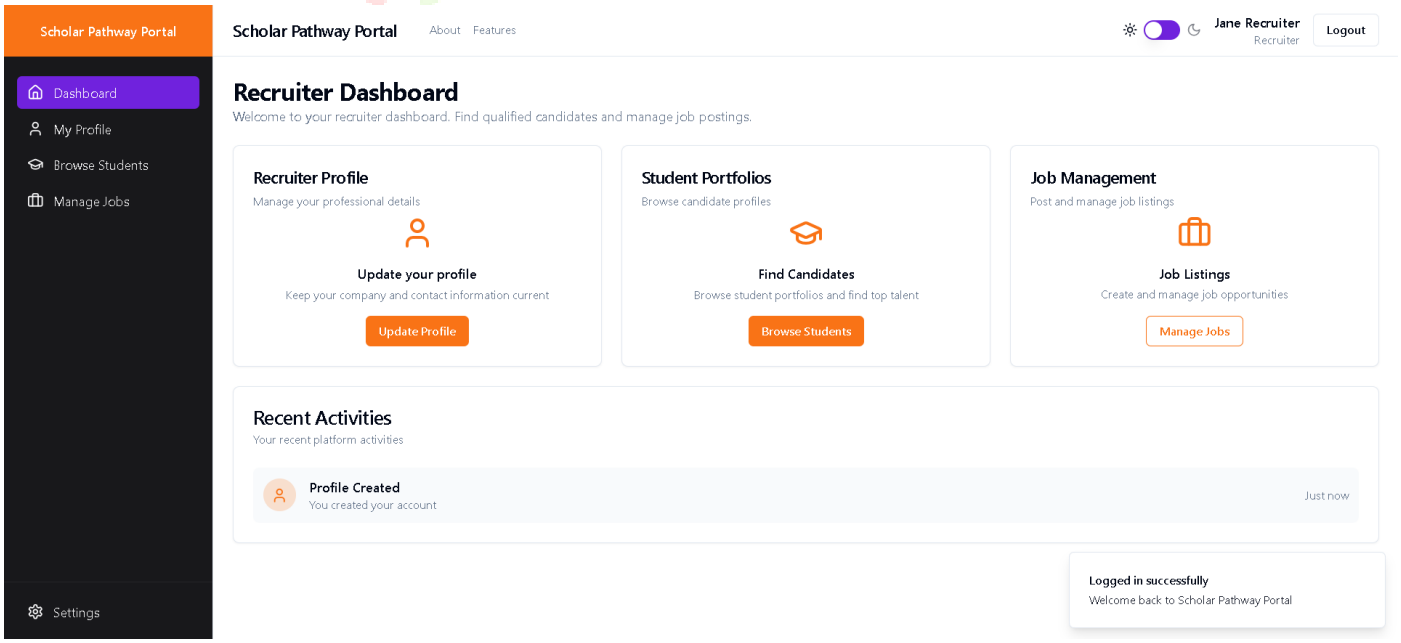

Fig. 11: Recruiter's Dashboard page

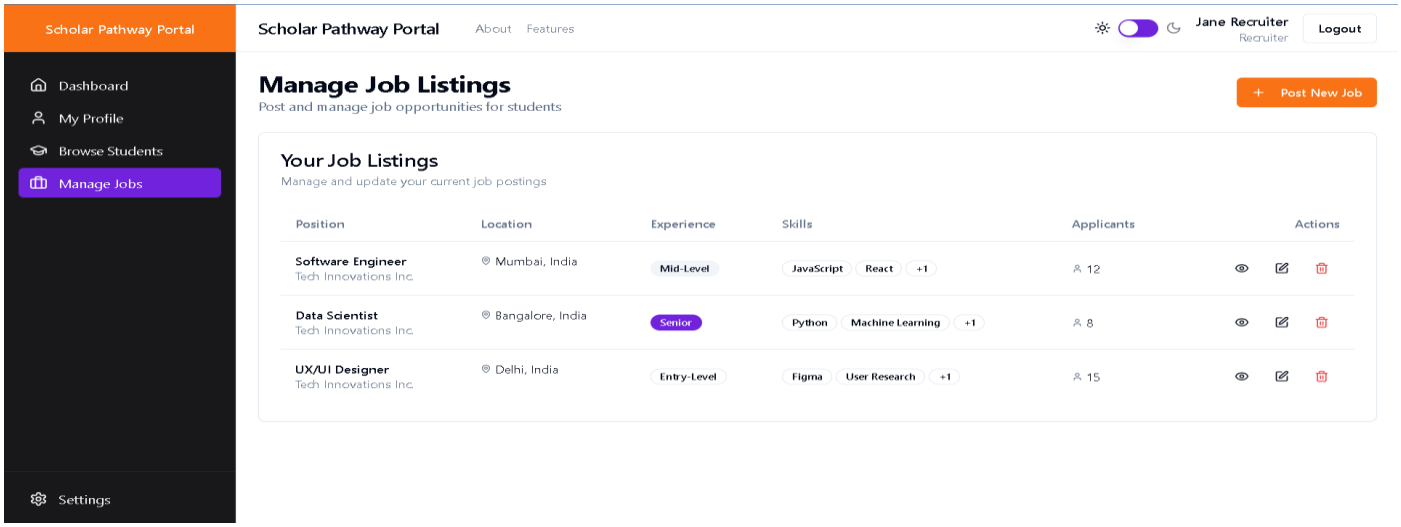

Fig. 12: Recruiter's Manage jobs page

VII. FUTURE SCOPE & RECOMMENDATIONS

7.1 Future Scope:

The AI-Based Learning Management System holds significant potential for further development and enhancement. Future advancements could include:

1. Voice-Enabled Assistance: Integrating voice-based AI assistants to help users navigate the platform and access learning content hands-free.
2. Predictive Analytics: Implementing machine learning models to predict student performance, dropout risks, and learning gaps for proactive intervention.
3. Gamification Features: Adding elements like badges, leaderboards, and rewards to increase student motivation and engagement.
4. Integration with AR/VR: Enhancing immersive learning experiences through augmented and virtual reality modules for subjects that benefit from interactive simulations.
5. Multilingual Support: Expanding the system's reach by incorporating multilingual NLP models to cater to a diverse user base.
6. Mobile App Development: Creating dedicated mobile applications for Android and iOS to ensure seamless learning on the go.

VIII. CONCLUSION

The AI-Based Learning Management System (LMS) project successfully integrated intelligent features such as adaptive learning recommendations, automated grading, and a responsive AI-powered chatbot into a traditional LMS framework, significantly enhancing user engagement, learning outcomes, and academic efficiency. Developed using a combination of modern web technologies, machine learning algorithms, and natural language processing tools, the system offered a personalized and interactive experience for students, teachers, and administrators alike. It was designed to be scalable and cost-effective, ensuring accessibility and ease of use across different devices and platforms.

Throughout the development process, the project encountered challenges such as limited datasets for training AI models, compatibility issues during integration with existing LMS modules, and maintaining real-time responsiveness. These obstacles were addressed through a modular architecture, continuous feedback loops, and iterative testing methodologies that allowed for quick identification and resolution of issues.

In addition to its core features, the system supports role-based dashboards, learning progress tracking, and dynamic scheduling, further adding to its versatility. It also ensures secure authentication for different user profiles, including students, teachers, and hiring managers, each with tailored access to relevant features and data.

Overall, the project not only showcases the transformative potential of AI in digital education but also lays a solid groundwork for future advancements. It opens up avenues for more sophisticated capabilities such as

predictive analytics, emotion-aware tutoring systems, and deeper integration with career guidance tools. As such, this AI-based LMS stands as a promising step toward the next generation of intelligent, personalized, and inclusive learning environments.

IX. REFERENCES

[1] Alharbi, S., & Alshammari, M. (2021). The effectiveness of using a learning management system on students' performance: A meta-analysis. *Journal of Educational Technology & Society*, 24(2), 1-15.

[2] Chen, L., & Liu, M. (2022). Personalized learning: A new paradigm for teaching and learning. *Educational Technology Research and Development*, 70(1), 153-175. https://doi.org/10.1007/s11423-021-10048-7

[3]Dabbagh, N., & Kitsantas, A. (2012). Personal Learning Environments, Social Media, and Self-Regulated Learning: A Natural Formula for Connecting Formal and Informal Learning. *The Internet and Higher Education*, 15(1), 3-8. https://doi.org/10.1016/j.iheduc.2011.06.002

[4]Liu, Y., & Wang, S. (2020). Developing a Learning Management System for Personalized Learning Based on Learning Styles. *International Journal of Emerging Technologies in Learning*, 15(10), 180-190. https://doi.org/10.3991/ijet.v15i10.12673

[5]McCarthy, J., & Wright, P. (2015). Technology as experience. *Interactions*, 22(6), 24-31. https://doi.org/10.1145/2804022

[6]Wang, F., & Wu, H. (2021). The Impact of Learning Management Systems on Student Performance: A Systematic Review. *Education and Information Technologies*, 26(3), 2413-2436. https://doi.org/10.1007/s10639-020-10300-0

[7] Zheng, Y., & Yu, Y. (2023). The Role of Artificial Intelligence in Education: Current Trends and Future Directions. *Computers & Education*, 180, 104483. https://doi.org/10.1016/j.compedu.2021.104483

[8] OpenAI. (2023). *GPT-4 Technical Report*. Retrieved from https://openai.com/research/gpt-4

[9] OpenAI. (2023). *ChatGPT* (Mar 14 version) [Large language model]. https://chat.openai.com/chat

[10]Tschofen, C., & Mackness, J. (2012). Connectivism and Learning Communities: A New Learning Paradigm. *International Review of Research in Open and Distributed Learning*, 13(3), 20-35. https://doi.org/10.19173/irrodl.v13i3.1383