

# CAPSTONE PROJECT

## PROBLEM STATEMENT NO.12 -AGENTIC AI FOR PERSONALIZED COURSE PATHWAYS.

Presented By:

1. Krishna Koushik Padigala-Rajiv Gandhi University of  
Knowledge Technologies, RK Valley- CSE

# OUTLINE







- Problem Statement
- Proposed System/Solution
- System Development Approach
- Algorithm & Deployment
- Result (Output Image)
- Conclusion
- Future Scope
- References

---

# PROBLEM STATEMENT

- Students often face difficulty in identifying a clear and effective learning path that aligns with their personal interests and long-term career goals. The overwhelming abundance of online courses, lack of structured guidance, and absence of personalized learning plans lead to confusion, wasted time, and demotivation. Without adaptive support, learners struggle to progress efficiently or stay aligned with their aspirations.

# PROPOSED SOLUTION

-  **Data Collection**
  - Collect user inputs: interests, goals, current skill level.
  - Optionally enhance with preferences like content format and time availability.
-  **Data Preprocessing**
  - Clean and structure inputs.
  - Extract key features like learning gaps and user preferences.
-  **AI Logic**
  - Use AI agents to generate personalized learning paths.
  - Adjust roadmaps dynamically based on progress and feedback.
  - Recommend relevant resources, projects, and quizzes.
-  **Deployment**
  - Build a simple, interactive interface for students to receive and update their roadmaps.
  - Deploy on a scalable platform with support for real-time updates.
-  **Evaluation**
  - Measure accuracy and relevance of recommendations.
  - Use metrics like user satisfaction, progress tracking, and feedback to refine the system.
-  **Result**
  - LearnMate guides students with evolving, personalized learning paths—helping them stay focused, motivated, and goal-oriented.

# SYSTEM APPROACH

- ✓ System Requirements
- ◆ Frontend
  - Chat UI built with HTML/CSS/JS or React
  - Form inputs for capturing goals, skills, and interests
  - Live chat or embedded iframe (for watsonx Assistant Preview)
- ◆ Backend & AI
- IBM watsonx.ai Preview Environment
  - Use prompt playground to experiment with different instructions and outputs
  - Leverage **Mistral 7B** or **Granite Code/Chat** models for generation
  - Use **prompt templates** and **custom agent instructions** for dynamic roadmap building
- Vector Retrieval (Optional)
  - Upload vectorized documents (PDFs, Markdown) to **IBM Knowledge Catalog** or local FAISS
  - Use semantic search tools to enhance retrieval-based generation
- ◆ User Data Handling
  - Store basic student data (interests, progress, preferences) in a lightweight DB (e.g., MongoDB or SQLite)
  - Implement session memory to align with watsonx agent memory (if supported)
- □ Libraries & Tools (Client Side / Backend)
  - UseTool / LibraryAI Model Accesswatsonx.ai Playground / SDKLocal Testingtransformers, mistral (if self-hosted)Backend Serverflask, fastapi, requestsDocument Embeddingsentence-transformers, faiss, or IBM watsonx Knowledge StudioData Handlingpandas, numpyUIReact.js, Bootstrap, plain HTML/JSDeploymentIBM Cloud / Render / Vercel (frontend)

# ALGORITHM & DEPLOYMENT

- □ **Algorithm Section Algorithm Selection**
- For LearnMate, an **Agentic AI-based recommendation system** is used, leveraging **LLMs (e.g., Mistral 7B Instruct or watsonx.ai Granite models)** for dynamic, personalized roadmap generation. This approach is chosen because it can understand natural language inputs, adapt based on learner goals and skill levels, and retrieve or generate appropriate learning content. Traditional machine learning algorithms like classification or regression are insufficient due to the conversational and evolving nature of the problem.
- ♦ **Data Input**
- The algorithm takes the following input features:
- User interests (e.g., Frontend Development, Cybersecurity, UI/UX)
- Skill level (beginner, intermediate, advanced)
- Learning goals (e.g., get job-ready in 6 months)
- Content preferences (video, text, project-based)
- Optional: time availability, preferred learning style, and past progress (if available)
- ♦ **Training Process**
- The system uses **pre-trained LLMs (like Mistral or watsonx models)**, optionally fine-tuned or guided through **prompt engineering** and **RAG (Retrieval-Augmented Generation)**.
- Vectorized documents (curricula, course roadmaps, tutorials) are embedded using tools like sentence-transformers and stored in a vector database (e.g., FAISS, Pinecone, or IBM Knowledge Studio).
- Prompts are optimized through iterative testing in watsonx.ai's Prompt Lab, rather than traditional ML training with labels.
- ♦ **Prediction Process**
- During interaction:
- The model interprets the user's inputs through natural language.
- It retrieves relevant course modules or learning content based on semantic similarity.
- Using the current context (interests, skill level, progress), the LLM generates a **custom learning roadmap** or **suggestions**.
- The system updates over time based on feedback and engagement, adapting future suggestions accordingly.

# RESULT

K

You 12:37 AM

hey



LearnMate 12:37 AM

Hi, I am LearnMate – your AI learning coach. How can I help you today?

K

You 12:37 AM

give me roadmap on frontend

Type something...



K

You 12:37 AM

give me roadmap on frontend



LearnMate 12:37 AM

Here is a roadmap for learning frontend development:

## Internet

- How does the internet work?
- What is HTTP?

Type something...




# CONCLUSION

- LearnMate presents an innovative solution to the challenge of navigating the overwhelming world of online learning by offering personalized, adaptive learning paths through Agentic AI. By leveraging powerful language models like Mistral and watsonx.ai, combined with vectorized document retrieval and user-centric design, the system can understand individual learner profiles and guide them toward their goals efficiently. With continuous feedback and contextual adaptation, LearnMate has the potential to transform how students explore, engage with, and excel in skill-based learning journeys.



# FUTURE SCOPE

-  **Future Scope**
- **Adaptive Assessments and Skill Tracking**  
Integrate real-time quizzes and project evaluations to assess learners' progress and dynamically update their learning paths with greater precision.
- **Integration with MOOCs and LMS Platforms**  
Connect LearnMate with platforms like Coursera, edX, Udemy, and institutional LMSs to recommend and track real course enrollments directly within the system.
- **Multi-Agent Collaboration**  
Introduce multiple specialized AI agents (e.g., Career Coach, Skill Assessor, Project Mentor) that collaboratively guide students in a more holistic learning experience.
- **Mobile App Deployment**  
Expand accessibility through a cross-platform mobile application, enabling learning-on-the-go with personalized notifications and progress reminders.
- **Gamification Features**  
Add achievements, leaderboards, and progress badges to improve learner engagement and motivation.
- **Support for Non-Technical Domains**  
Extend the system to support other disciplines like arts, humanities, and commerce by incorporating domain-specific learning paths and content.
- **Multilingual and Voice Interaction Support**  
Implement multilingual support and voice-based interaction to enhance accessibility for diverse learners across regions.
- **Data-Driven Personalization with Learning Analytics**  
Use advanced analytics to provide deeper insights into learning patterns and recommend strategies for improvement and time management.

# REFERENCES

- Fanaee-T, Hadi, and Gama, João.  
“Event labeling combining ensemble detectors and background knowledge.” *Progress in Artificial Intelligence*, vol. 2, no. 2–3, 2014, pp. 113–127.
- Hyndman, Rob J., and Athanasopoulos, George.  
“Forecasting: Principles and Practice.” *OTexts*, 2nd edition, 2018.  
(Comprehensive resource for time-series forecasting methods such as ARIMA, Exponential Smoothing)  
<https://otexts.com/fpp2>
- Brownlee, Jason.  
“Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python.” *Machine Learning Mastery*, 2018.  
(Practical implementation of LSTM and deep learning models for time-series forecasting)
- IBM Watsonx Documentation.  
“watsonx.ai: Next-generation AI and Data Platform for Business”, IBM, 2023.  
<https://www.ibm.com/products/watsonx-ai>  
(Used for AI model experimentation and evaluation)

# IBM CERTIFICATIONS

In recognition of the commitment to achieve  
professional excellence



Krishna Koushik P

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence



Issued on: Jul 16, 2025  
Issued by: IBM SkillsBuild

Verify: <https://www.credly.com/badges/3a784441-fd2d-459a-8538-d7e26e839b37>



# IBM CERTIFICATIONS

In recognition of the commitment to achieve  
professional excellence



Krishna Koushik P

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution



Issued on: Jul 21, 2025  
Issued by: IBM SkillsBuild

Verify: <https://www.credly.com/badges/0c5da82d-901c-40fc-93fc-fb2fa567dd18>



# IBM CERTIFICATIONS

IBM **SkillsBuild**

Completion Certificate



This certificate is presented to  
**Krishna Koushik Padigala**

for the completion of

**Lab: Retrieval Augmented Generation with  
LangChain**

(ALM-COURSE\_3824998)

According to the Adobe Learning Manager system of record



**THANK YOU**