TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PURWANCHAL CAMPUS

A MINOR PROJECT PROPOSAL ON

LEARNGENIE: AN AI-DRIVEN PERSONALIZED LEARNING PLANNER

BY

ANUJA KATUWAL(PUR079BCT009)

INTAF MALIK(PUR079BCT029)

KESHAB NEUPANE(PUR079BCT031)

SAMRIDHHI LAMSAL(PUR079BCT072)

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

PURWANCHAL CAMPUS

DHARAN, NEPAL

JANUARY 13, 2026

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude to the Head of the Department (HOD), Mr. Manoj Kumar Guragain, and the Deputy Head of the Department, Mr. Dharti Raj Shah and Mr.Bikram Shah, for their constant guidance, valuable suggestions, and constructive feedback throughout the preparation of this proposal. Their continuous support and encouragement have played a significant role in shaping this work.

We are also sincerely thankful to all the respected teachers and faculty members for their helpful advice, motivation, and academic insights, which have greatly enhanced the quality of this proposal. Their dedication to academic excellence and passion for sharing knowledge have been a true source of inspiration.

The collective support and guidance from the department have greatly contributed to the refinement and overall improvement of this proposal, and we are deeply grateful for their mentorship and encouragement throughout the process.

LIST OF ABBREVIATIONS

AI : Artificial Intelligence

LLM : Large Language Model

API : Application Programming Interface

PWA : Progressive Web App

LMS : Learning Management Systems

ITS : Intelligent Tutoring Systems

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Background Study

The rapid growth of digital learning platforms such as YouTube, Udemy, and Coursera has made education more accessible than ever. However, this abundance of resources has also created significant information overload, leaving learners confused about where to begin and how to progress in a structured manner. Learners often rely on multiple disconnected tools for content consumption, scheduling, and note-taking, which results in inefficiency, poor time management, and low course completion rates.

Recent advancements in Artificial Intelligence (AI) and recommendation systems present an opportunity to address these challenges. While adaptive learning platforms highlight the benefits of personalized education, there is still no unified solution that integrates structured learning paths, high-quality resource recommendations, personalized timetables, progress tracking, and built-in learning tools. LearnGenie aims to bridge this gap by leveraging AI-driven technologies to deliver a guided, efficient, and personalized learning journey from beginner to advanced levels.

1.2 Problem Definition

Learners often struggle with unclear topics, poor time management, suboptimal resource selection, and lack of progress tracking, leading to high dropout rates in online courses. Existing apps lack comprehensive AI-driven timetables, high-quality video recommendations from YouTube/Udemy based on ratings/reviews, integrated note-taking with summaries, and adaptive tracking.

1.3 Objectives

- Generate topic sequences for any skill using Large Language Models (LLMs).
- Recommend top-rated videos and resources from YouTube and Udemy and so on.
- Enable note-taking, completion status tracking.

1.4 Scope of Project

The project develops an AI-powered web application for structured technical skill learning. It generates personalized learning paths using LLMs, creates resources from YouTube, Udemy, Coursera and so on via APIs, implements adaptive scheduling, and provides integrated note-taking and progress tracking. Delivered as a responsive web/PWA, it initially focuses on English technical content and excludes native mobile apps.

2 LITERATURE REVIEW

2.1 Historical Evolution

Early intelligent educational systems focused on rule-based instruction and static content delivery. Woolf [1] highlighted the role of artificial intelligence in intelligent tutoring systems, showing how learner progress can be monitored and instruction adapted accordingly.

As e-learning platforms expanded, limitations of monolithic architectures became evident. Dagger et al. [2] proposed service-oriented learning platforms that improved flexibility, reusability, and personalization. With the growth of learner data, learning analytics gained importance. Verbert et al. [3] demonstrated that analytics dashboards help visualize learner behavior and support data-driven decisions. Recent studies have explored artificial intelligence in higher education. Zawacki-Richter et al. [4] identified key AI applications and challenges related to scalability and system integration.

2.2 Algorithms and Models

2.2.1 Personalized Learning Models

Personalized learning models adapt content and learning paths based on learner profiles. Brusilovsky [5] showed that adaptive hypermedia techniques effectively generate customized learning experiences.

2.2.2 AI-Driven Learning Support Models

Holmes et al. [6] discussed how artificial intelligence can support learning through intelligent feedback systems, automated assessment, and adaptive content recommendation. Rather than relying solely on traditional analytics dashboards, AI-driven models focus on continuous learner monitoring and real-time instructional adjustment.

2.2.3 Agent-Based and Modular Models

Agent-based and modular architectures are increasingly used in modern educational systems. LangGraph [7] enables multi-agent workflows using large language models, supporting dynamic and adaptive learning roadmap generation.

2.3 Data Processing and Learning Analytics

Effective data processing is essential for accurate personalization in learning systems. Learning analytics techniques analyze learner behavior to improve engagement and system effectiveness. Panigrahi et al. [8] reported that proper data analysis significantly enhances learning outcomes in online education platforms.

2.4 Gaps and Challenges

Although personalized learning systems have advanced, many platforms still rely on static learning paths with limited adaptability. Such systems often fail to update learning plans based on continuous learner progress and changing goals.

Another challenge is the lack of integration among intelligent components, resulting in fragmented learning experiences and reduced effectiveness. Therefore, there is a need for an AI-driven learning planner capable of dynamically generating and managing personalized learning roadmaps.

LEARNGENIE aims to address these challenges by providing an adaptive, intelligent, and personalized learning planning system.

3 METHODOLOGY

3.1 System Design

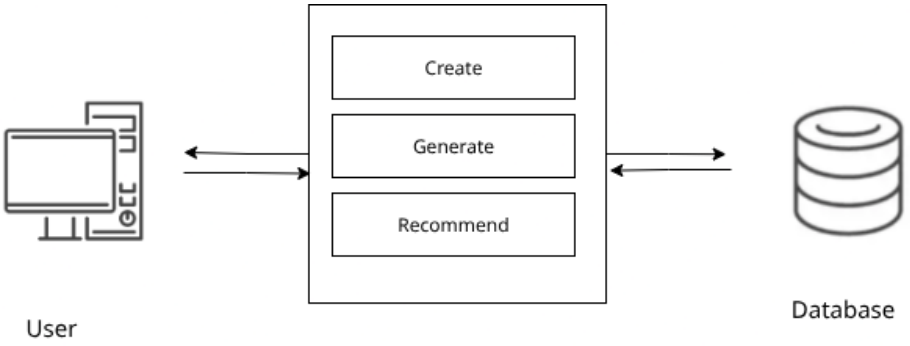

Figure 1: System Design

3.2 Proposed Block Diagram

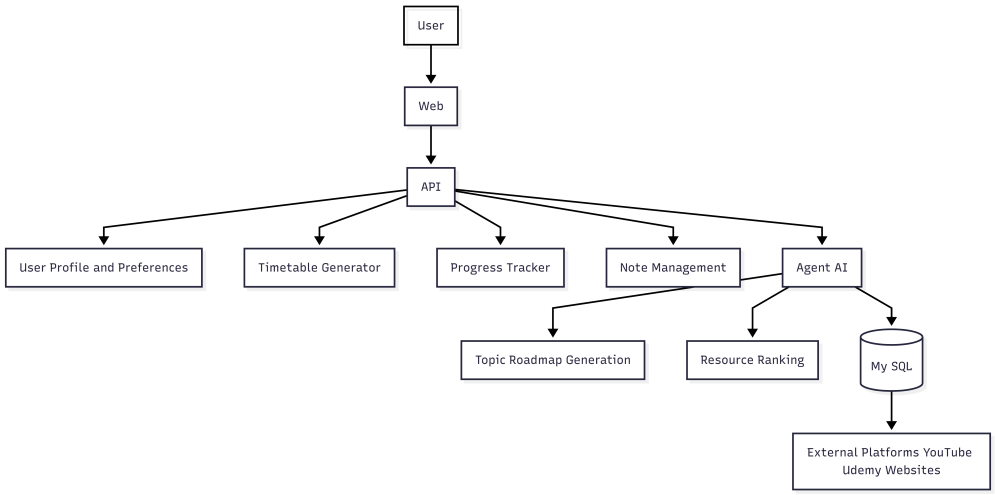

Figure 2: Proposed Block Diagram

3.3 Tools and Techniques

3.3.1 Tools

i. **Python**

Python is used as the main backend programming language for the LearnGenie system. It supports easy integration with Large Language Models (LLMs) and

helps in processing data efficiently.

ii. **Python Libraries**

- **Django** – Django is a free and open-source web framework built using Python. It provides a collection of integrated tools and components that enable developers to create web applications or websites.
- **LangGraph** - LangGraph is a framework built on top of LangChain that is designed to develop stateful and multi-step Large Language Model (LLM) workflows using graph-based execution.

iii. **React.js**

React.js is a JavaScript library used to develop the frontend of the application. It uses a component-based architecture and JSX, which combines HTML-like syntax with JavaScript logic to create fast, interactive, and responsive user interfaces.

iv. **Node.js**

Node.js is used for backend development to handle API requests, logic, and communication with the database and external services.

v. **MySQL**

MySQL is used as the database to store user information, learning paths, notes, and completion status.

3.3.2 Techniques

The following techniques will be adopted during the development of the proposed system, LearnGenie, to ensure systematic planning, efficient implementation, and reliable performance.

i. **Requirement Analysis**

In this phase, user requirements and system objectives will be identified through observation and analysis of common learning challenges faced by students. This step helps in defining the functional and non-functional requirements of the learning assistant platform.

ii. **Data Collection and Filtering**

Learning resources will be collected from online platforms such as YouTube and educational documentation websites. The collected resources will be filtered based on relevance to the learning topic, popularity indicators, and difficulty level to ensure quality recommendations.

iii. **System Design**

The overall system architecture will be designed using appropriate modeling tools such as Data Flow Diagrams (DFD), Entity Relationship (ER) diagrams, and UML diagrams. These diagrams will clearly represent data flow, system components, and interactions between different modules.

iv. **Model Design and Integration**

A rule-based and Agentic AI-driven approach will be designed to generate structured learning paths and recommend resources. The model logic will be integrated with the backend system to ensure seamless interaction between learning goals, recommendations, and progress tracking.

v. **Website Development**

The frontend of the system will be developed using modern web technologies with a focus on simplicity, responsiveness, and ease of use. The interface will allow users to select learning goals, view learning plans, and track progress efficiently.

vi. **Deployment**

After development and testing, the proposed system will be deployed on a local or cloud environment for demonstration purposes. Deployment will ensure that all system components work together in a real-world scenario.

4 IMPLEMENTATION

LearnGenie is implemented as a modular web-based application that integrates a frontend interface, API layer, AI services, database, and external learning platforms. The system focuses on personalized learning, efficient planning, and seamless user interaction.

4.1 Frontend Implementation

The frontend is developed as a responsive web application that allows users to select skills, view personalized timetables, track learning progress, and take notes. Users can mark tasks as completed, manage deadlines, and access learning resources easily. Progressive Web Application (PWA) features ensure usability across different devices.

4.2 API Layer Implementation

The API layer manages communication between the frontend, AI engine, database, and external platforms. It provides secure RESTful endpoints for user management, timetable generation, progress tracking, and note handling, ensuring smooth data flow and system reliability.

4.3 AI Recommendation Engine Implementation

The AI recommendation engine uses Large Language Models (LLMs) to generate structured learning roadmaps and topic sequences based on user preferences, skill level, and available study time. Learning plans are updated dynamically as users progress.

4.4 Resource Recommendation Implementation

Learning resources are retrieved from platforms such as YouTube and Udemy using their APIs. Resources are ranked using popularity metrics such as views, ratings, and relevance to ensure high-quality content recommendations.

4.5 Progress Tracking and Note Management

The system tracks topic completion status and due dates to monitor learner progress. Users can store and manage notes for each topic, supporting better understanding and revision. All updates are synchronized in real time.

4.6 Database and System Integration

MySQL is used as the primary database to store user profiles, learning paths, progress records, and notes. The relational schema ensures efficient storage and retrieval of structured data. All components are integrated through the API layer, enabling a unified, scalable, and efficient learning platform.

5 EXPECTED OUTPUT

LearnGenie delivers a unified learning platform with personalized timetables, top-rated YouTube/Udemy video recommendations, real-time progress dashboards, and integrated note-taking for every topic—transforming self-study into structured skill mastery.

6 EXPECTED SCHEDULE

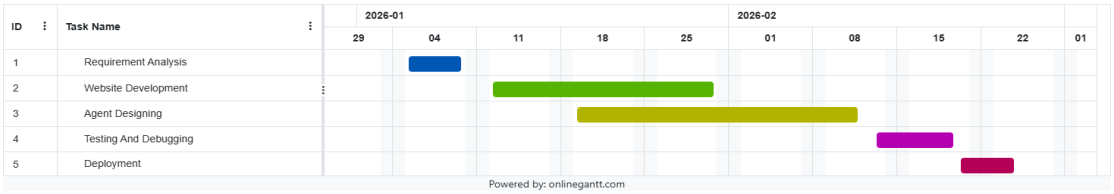

Figure 3: Gantt Chart of Expected Schedule

REFERENCES

[1] B. P. Woolf, *Building Intelligent Interactive Tutors*. Morgan Kaufmann, 2010.

[2] D. Dagger, A. O'Connor, S. Lawless, E. Walsh, and V. P. Wade, "Service-Oriented E-Learning Platforms: From Monolithic Systems to Flexible Services," *IEEE Internet Computing*, vol. 11, no. 3, pp. 28–35, 2007.

[3] K. Verbert, E. Duval, J. Klerkx, S. Govaerts, and J. L. Santos, "Learning Analytics Dashboard Applications," *American Behavioral Scientist*, vol. 57, no. 10, pp. 1500–1519, 2013.

[4] O. Zawacki-Richter, M. Bond, V. I. Marin, and F. Gouverneur, "Systematic Review of Research on Artificial Intelligence Applications in Higher Education," *International Journal of Educational Technology in Higher Education*, 2019.

[5] P. Brusilovsky, "Adaptive Hypermedia for Education and Training," in *Adaptive Technologies for Training and Education*. Cambridge University Press, 2001.

[6] W. Holmes, M. Bialik, and C. Fadel, *Artificial Intelligence in Education: Promises and Implications for Teaching and Learning*. Center for Curriculum Redesign, 2019.

[7] LangChain Documentation Team. (2023) Langgraph: Multi-agent workflows for language models. [Online]. Available: https://docs.langchain.com

[8] R. Panigrahi, P. R. Srivastava, and D. Sharma, "Online Learning: Adoption, Continuance, and Learning Outcomes—A Review," *International Journal of Information Management*, 2018.