

# EduPlannerAI: Agentic Academic Scheduler Powered by Custom Trained LLMs

CH. Lakshmi Kumari<sup>1</sup>,

<sup>1</sup>Assistant Professor, Department of IT, Mahatma Gandhi Institute of Technology, Hyderabad, Telangana, India.

Korivi Venkata Sai Shamitha<sup>2</sup>, Ethamukkala Narendra<sup>3</sup>

<sup>2,3</sup>UG Student, Department of CSBS, Mahatma Gandhi Institute of Technology, Hyderabad, Telangana, India.

**Abstract** - EduPlannerAI is an intelligent, multi-agent academic planning system designed to help students manage their studies through adaptive scheduling, AI-powered note understanding, and natural conversational interaction. The system integrates data from diverse sources—including email, WhatsApp, LMS platforms, and uploaded documents—to automatically extract tasks, deadlines, and relevant study materials. Using a combination of machine learning models, optimization algorithms, and AI summarization, EduPlannerAI creates personalized study plans that adjust dynamically based on a student's pace, progress, and learning behavior.

The architecture is built around a LangGraph-driven orchestration layer that coordinates specialized agents for scheduling, progress tracking, tutoring, note summarization, and conversational planning. These agents work together with a centralized backend (Supabase), a vector retrieval layer, and an automated ingestion pipeline (n8n) to deliver a unified, real-time learning environment. The frontend, built with Next.js, provides an interactive interface that synchronizes seamlessly with backend updates and supports chat-based learning, adaptive to-do lists, and AI-assisted note insights. By combining advanced scheduling algorithms, adaptive feedback loops, and natural language interfaces, EduPlannerAI transforms traditional study planning into a dynamic, personalized experience.

**Keywords:** Agentic AI, Multi-Agent Systems, LangGraph, Large Language Models (LLMs), Retrieval-Augmented Generation (RAG), Adaptive Scheduling, OR-Tools, Vector Database, FastAPI, Supabase, Next.js, Personalized Learning, Conversational Planner.

## 1. INTRODUCTION

In recent years, intelligent learning technologies have gained significant attention for their potential to transform academic planning and student productivity. AI-driven systems, adaptive schedulers, and automated note-processing tools have introduced new ways for learners to organize, understand, and manage their studies without relying solely on traditional manual methods. While these advancements have improved accessibility and personalization, they have also highlighted challenges related to planning accuracy, information overload, and effective time management.

Within modern learning ecosystems, students are often expected to navigate large volumes of content, manage multiple deadlines, and make independent decisions about how to allocate their study time. This flexibility encourages autonomy but also exposes gaps in self-regulation, task prioritization, and timely revision habits. EduPlannerAI addresses these gaps by combining multi-agent orchestration, retrieval-augmented generation, and constraint-based optimization.

- [1] Adaptive scheduling that automatically generates and updates study plans based on deadlines, student performance, and real-time progress.
- [2] Task prioritization and workload management driven by analysis of academic requirements, task complexity, and learning behavior.
- [3] Intelligent note processing capable of summarizing, structuring, and categorizing study materials for easier revision and better retention.
- [4] Predictive insights that help students identify learning gaps and recommend optimal times for revision or additional study support.
- [5] Conversational interaction through a Study Copilot that interprets natural-language commands such as "Move chemistry revision to Friday."
- [6] Unified integration of assignments, exams, notes, and reminders into a single platform for improved accessibility and

organization.

By merging planning intelligence with content processing and natural communication, EduPlannerAI reduces the cognitive load of managing studies and allows students to focus more on actual learning while preserving full autonomy over their study decisions.

### **Problem Statement**

Students today are required to manage a growing number of academic tasks—assignments, exams, readings, projects, and personal notes—often spread across multiple platforms and formats. Although digital tools exist to store information or set reminders, they rarely help learners understand how to structure their study routine or when to focus on specific topics. Static planners cannot adjust when a student falls behind, works faster than expected, or needs additional revision.

Alongside scheduling challenges, students frequently struggle to transform large volumes of notes into clear study material. The key problem is therefore the absence of an integrated system that can interpret learning materials, analyze student behavior, and produce an adaptive plan that evolves automatically. Students need a tool that not only tracks their academic responsibilities but also guides their study process with meaningful, context-aware assistance.

## **2. EXISTING SYSTEM**

Conventional academic planning tools rely on static calendars, manual to-do lists, and rule-based reminder systems that do not adapt to a student's evolving pace or context. While platforms such as Google Calendar, Notion, and traditional Learning Management Systems offer organization features, they place the burden of structuring study routines, prioritizing tasks, and condensing study material entirely on the learner.

More recent AI-based e-learning systems introduce content recommendation and adaptive learning paths, but most are built around isolated capabilities—either personalized content delivery, basic scheduling, or chatbot-style tutoring—without unifying them into a cohesive, agent-driven workflow. Multi-agent scheduling research has demonstrated the value of distributed reasoning for dynamic plan adjustment, yet such approaches have rarely been combined with Retrieval-Augmented Generation (RAG) for grounded tutoring or with constraint-based optimization for realistic timetable generation.

Existing tutoring assistants powered by LLMs frequently suffer from hallucinations because they do not retrieve from the student's own learning material. As a result, students still must switch between separate tools for note management, scheduling, tutoring, and progress tracking, leading to fragmented workflows and increased cognitive load.

## **3. LITERATURE SURVEY**

AI-Based Personalized E-Learning Systems: Issues, Challenges, and Solutions by Mir Murtaza et al. examines the current landscape of AI-driven personalization in e-learning and discusses how learner modeling, adaptive pathways, and intelligent recommendations can enhance academic experiences. The authors outline major challenges such as data sparsity, privacy concerns, and the need for accurate feedback mechanisms while highlighting promising solutions involving machine-learning-based profiling, dynamic content sequencing, and responsive learning interventions.[1]

Learning in Multi-Agent Systems to Solve Scheduling Problems by Gabriel Icarte-Ahumada provides a detailed analysis of how multi-agent learning techniques have been applied to complex scheduling tasks.[2]

Retrieval-Augmented Generation for Educational Applications by Z. Li et al. presents an exploration of how RAG frameworks can support educational systems by combining retrieval mechanisms with large language models to generate grounded, context-aware explanations and responses. The authors demonstrate that integrating external knowledge sources improves factual reliability and helps align AI output with curricular material.[3]

Explainable Artificial Intelligence (XAI) 2.0: A Manifesto of Open Challenges and Research Directions by Longo et al. surveys emerging challenges in explainability and proposes forward-looking research directions aimed at making AI systems more transparent, interpretable, and human-centered. The authors argue for enhanced causal reasoning, interactive explanations, and user-adaptive interpretability frameworks that support effective decision-making.[4]

AI-Driven Adaptive Learning for Sustainable Educational Development by W. Strielkowski et al. discusses how adaptive learning systems powered by artificial intelligence can contribute to equitable, efficient, and scalable education. The paper outlines how personalized instruction, automated content delivery, and data-driven learning pathways can improve engagement and long-term

educational outcomes, highlighting the role of AI in fostering sustainable educational models that extend access and support diverse learners.[5]

Retrieval-Augmented Generation to Improve Math Question-Answering by Levonian, Li et al. investigates how RAG influences the quality of mathematical question-answering by evaluating the balance between factual grounding and user-preferred response styles.[6]

How to Build an Adaptive AI Tutor for Any Course Using Knowledge Graph-Enhanced Retrieval-Augmented Generation (KG-RAG) by Chenxi Dong et al. introduces a framework that integrates knowledge graphs with retrieval-augmented generation to produce course-aligned tutoring responses. The authors demonstrate how structured knowledge representations improve context relevance, support prerequisite mapping, and enable flexible adaptation across different subjects, with empirical improvements of approximately 35% over baseline approaches in controlled experiments.[7]

Retrieval-Augmented Generation Chatbots for Education by Swacha and Gracel surveys RAG-based chatbot use in educational settings, covering architectures, indexing strategies, response evaluation methods, and implementation challenges. The survey highlights the growing adoption of retrieval-enhanced dialogue systems due to their improved accuracy and contextual grounding, noting their applicability across diverse educational tasks ranging from explanation generation to content review.[8]

#### 4. PROPOSED SYSTEM

The proposed methodology for EduPlannerAI presents a robust agentic system that combines a LangGraph-based multi-agent orchestration layer, Retrieval-Augmented Generation (RAG), and constraint-based optimization to deliver adaptive academic planning. The architecture is organized into three major layers—an Interface Layer (Next.js + React), an Application Layer (FastAPI + Supabase), and an Intelligence Layer (specialized AI agents).

At the heart of the system, specialized agents collaborate through LangGraph: the SchedulerAgent applies OR-Tools optimization and pacing data to generate conflict-free study plans; the NotesAgent extracts text, summarizes content, and creates embeddings for fast semantic retrieval.

The TutorAgent uses RAG over a vector database (Milvus / Weaviate) to deliver explanations grounded in the student's own materials, while the ProgressAgent monitors task completion to dynamically adjust upcoming workload distribution. Custom-trained machine learning models estimate task difficulty and predicted study durations, enabling realistic schedule generation that adapts to each student's pace. A conversational planner powered by LLMs interprets natural-language commands such as "Move chemistry revision to Friday," allowing students to modify schedules without navigating complex menus.

In contrast to conventional planners that rely on static rules or isolated AI features, this system minimizes cognitive load by combining automation with personalization while preserving student autonomy. Its modular and scalable design supports a wide range of academic use cases, from daily revision planning to long-horizon exam preparation, making EduPlannerAI a comprehensive digital companion for modern learners.

The proposed system overcomes the fragmentation of existing tools by delivering a unified, conversational, and continuously adaptive learning assistant—one that interprets a student's materials, monitors progress, and produces personalized study plans without manual intervention.

#### 5. SYSTEM ARCHITECTURE

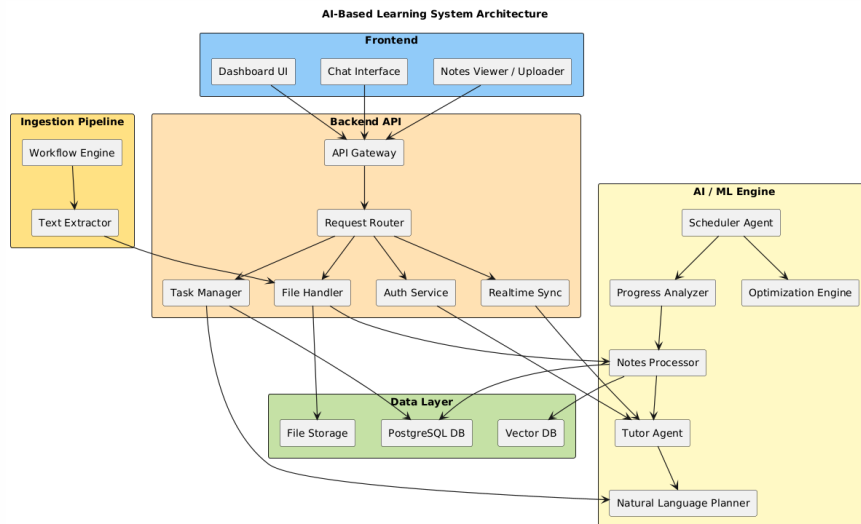


Fig. 1: System Architecture

EduPlannerAI follows a layered, agent-oriented architecture composed of an Interface Layer, an Application Layer, and an Intelligence Layer. The Interface Layer—built using Next.js, React, and Tailwind CSS—provides dashboards, calendars, the chat-based Study Copilot, and a notes uploader. User actions flow through the API gateway in the Application Layer, where FastAPI services communicate with Supabase (PostgreSQL + Storage + Realtime) for persistent data.

The Intelligence Layer is orchestrated by LangGraph, which coordinates a graph of specialized agents: the SchedulerAgent (powered by OR-Tools), NotesAgent (RAG indexing pipeline), TutorAgent (RAG retrieval + LLM generation), and ProgressAgent (pace tracking). A vector database (Milvus / Weaviate) stores embeddings of notes, summaries, and learning content, while transformer-based embedding models support semantic search for tutoring and note retrieval. An ingestion workflow engine (n8n) automates the intake of uploaded documents and triggers downstream processing.

## 6. RESULTS

The EduPlannerAI system was evaluated across multiple workflows including authentication, note upload and processing, conversational scheduling, exam preparation, and real-time interaction. End-to-end acceptance testing covering user signup, dashboard navigation, study material upload, AI-driven scheduling queries, RAG-based question answering, and dynamic progress tracking confirmed that all major flows perform reliably under normal usage conditions.

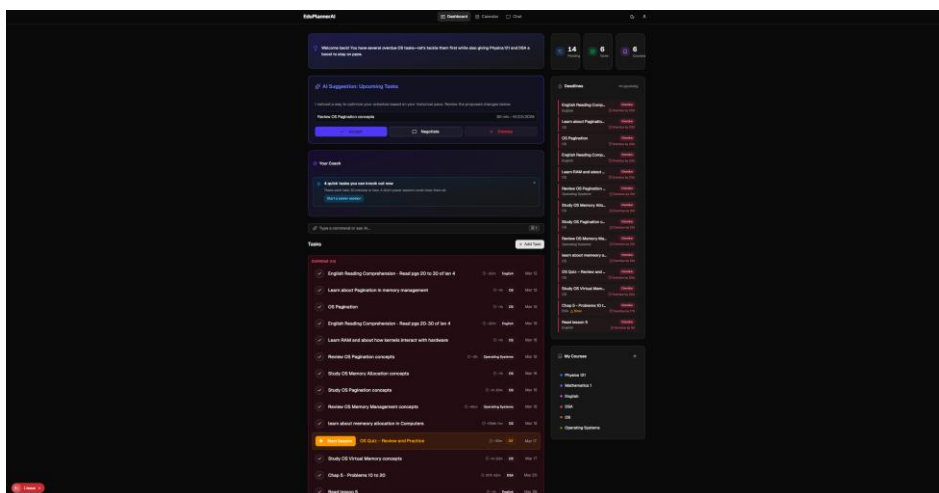


Fig. 2: Dashboard – Task Management & AI Insights

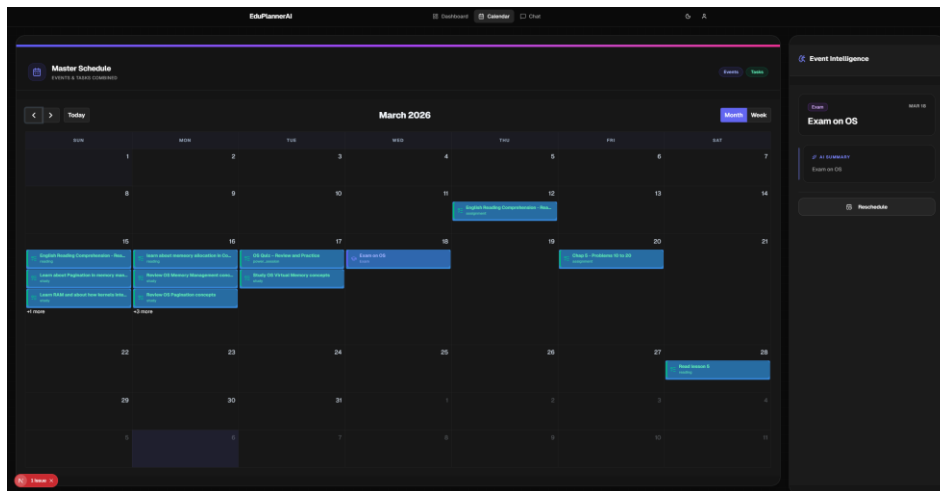


Fig. 3: Calendar – Adaptive Study Schedule

Adaptive scheduling produced conflict-free, personalized study plans that respond dynamically to deadlines, task difficulty, and user pace. The Study Copilot conversational interface successfully interpreted natural-language commands for plan modification, while the RAG-based tutor returned grounded, context-aware explanations sourced from the student's uploaded materials. Real-time synchronization between the dashboard, calendar, and chat interface ensured that updates to tasks, schedules, and notes were immediately reflected across the platform.

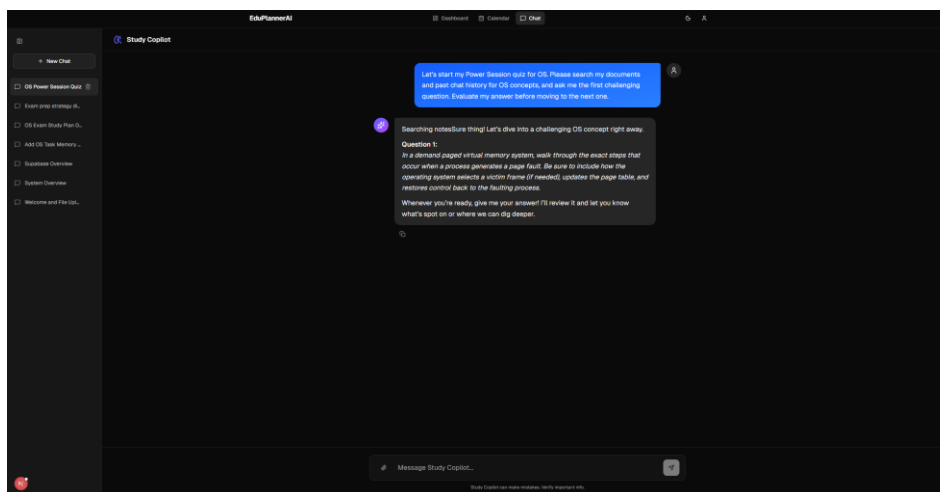


Fig. 4: AI Chat Interface – Intelligent Tutoring & Interaction

The dashboard view exposes urgency-based task categorization, AI-generated suggestions, and a coaching panel that summarizes recent academic performance. The calendar view displays the adaptive study schedule with quick rescheduling options. The chat interface supports retrieval-augmented tutoring and step-by-step concept walkthroughs. Together these interfaces demonstrate the system's ability to consolidate planning, content understanding, and tutoring into a single user-centric experience.

Observed limitations include occasional non-determinism in LLM responses, dependence on external services (Supabase, LLM providers, RAG microservice) for uptime, OCR inaccuracies on scanned PDFs, and heuristic-based fallbacks in scheduling under unusual constraints.

The system's performance was assessed using key qualitative indicators:

- [7] Reliability — measured through end-to-end acceptance tests across authentication, scheduling, note upload, and chat workflows.
- [8] Adaptability — measured by the system's ability to reschedule, rebalance, and reprioritize tasks dynamically in response to changes.
- [9] Groundedness — measured by the AI tutor's reliance on retrieval over the student's own materials rather than open

generation.

[10] Usability — measured by the smoothness of dashboard, calendar, and Study Copilot interactions during real-time use.

## 7. CONCLUSION

EduPlannerAI demonstrates the design and implementation of an intelligent, adaptive academic planning system that addresses the growing complexity of modern learning environments. By integrating a LangGraph-orchestrated multi-agent architecture, Retrieval-Augmented Generation, OR-Tools-based optimization, and a real-time Next.js / Supabase stack, the system unifies adaptive scheduling, AI-driven note processing, and conversational interaction into a single user-centric platform.

The multi-agent design enables specialized reasoning across scheduling, progress tracking, note summarization, and tutoring, while RAG ensures that AI responses remain grounded in the student's own materials. Custom-trained models estimate task difficulty and pacing, allowing schedules to adapt continuously based on real-time student behavior. The conversational Study Copilot further reduces friction by enabling natural-language plan modification.

Future work will focus on a cross-platform mobile application with push notifications, voice-based interaction, deeper personalization through reinforcement learning, integration with external LMS platforms (Moodle, Google Classroom, Microsoft Teams), multimodal tutoring with diagrams and quizzes, collaborative study features, cloud-native scaling, and stronger privacy and ethical-AI safeguards.

## REFERENCES

- [1] M. Murtaza, M. A. Khan, S. A. A. Shah, and M. Z. Asghar, "AI-Based Personalized E-Learning Systems: Issues, Challenges, and Solutions," *Journal of Computer and Education Research*, vol. 11, no. 1, 2024, pp. 145–160.
- [2] G. Icarte-Ahumada, "Learning in Multi-Agent Systems to Solve Scheduling Problems: A Systematic Literature Review," *Applied Sciences*, vol. 14, no. 5, 2024. DOI: 10.3390/app14052881.
- [3] Z. Li et al., "Retrieval-Augmented Generation for Educational Applications," *IEEE Transactions on Learning Technologies*, 2024. DOI: 10.1109/TLT.2024.3381298.
- [4] L. Longo et al., "Explainable Artificial Intelligence (XAI) 2.0," arXiv:2310.19775, 2024.
- [5] W. Strielkowski, K. Dvořák, and T. Chlebana, "AI-Driven Adaptive Learning for Sustainable Educational Development," *Sustainability*, vol. 16, no. 3, 2024. DOI: 10.3390/su16031252.
- [6] Z. Levonian, C. Li et al., "Retrieval-Augmented Generation to Improve Math Question-Answering," *Proc. EDM 2024*, pp. 210–221.
- [7] C. Dong, Y. Yuan, K. Chen, S. Cheng, and C. Wen, "How to Build an Adaptive AI Tutor for Any Course Using Knowledge Graph-Enhanced Retrieval-Augmented Generation (KG-RAG)," arXiv:2312.06789, 2024.