# Problem Statement: Agentic AI Tutor for Adaptive Learning in EdTech

## Background

Traditional e-learning platforms often provide the same content and pace to all learners, regardless of their background knowledge, learning speed, or weak areas. This leads to low engagement and poor learning outcomes.

**Agentic AI systems** can overcome this limitation by acting as **autonomous teaching agents**: they can reason about a student's progress, plan the next learning step, and adaptively deliver content. Unlike static chatbots, an agentic tutor can:

- Assess a learner's knowledge gaps,
- Choose the right learning material,
- Generate tailored questions and hints,
- Monitor engagement, and
- Escalate when human intervention is needed.

## Research Problem

Design and prototype an **Agentic AI Tutor** that provides **personalized, adaptive learning experiences** for students preparing for competitive exams (e.g., JEE, SAT, GRE).

The system should:

1. Ingest learning material (notes, questions, solutions).
2. Interact with students through natural conversation.
3. Dynamically assess student performance (via quizzes, Q&A).
4. Adapt study plans based on progress, weak topics, and learning pace.
5. Operate with **agent-like autonomy**: reason, plan, and act using available tools (content retrieval, quiz generator, progress tracker).

# Objectives

- Implement a **multi-agent workflow** (e.g., Planner Agent, Quiz Generator Agent, Feedback Agent).
- Use **RAG (Retrieval-Augmented Generation)** to pull content from a knowledge base of study material.
- Introduce **active recall techniques** (flashcards, self-testing) into the learning loop.
- Provide **explainable recommendations** (why a topic or question is being assigned).
- Log all interactions for teachers/parents as progress reports.

---

# Example User Flow

- Student: *“I’m struggling with Probability in Math.”*
- Tutor Agent:
    - Retrieves relevant notes on Probability.
    - Generates 3 progressively harder practice questions.
    - Evaluates answers and gives hints when wrong.
    - Updates the student profile → marks “Probability” as weak area.
    - Plans next session to focus on conditional probability.

# Deliverables

1. **Flask/Streamlit application** with endpoints like:
    - POST /ask → student asks a question, agent responds.
    - POST /quiz → generate adaptive quiz for a topic.
    - GET /progress/{student_id} → fetch student progress report.
2. **Knowledge base integration** (PDFs/notes stored with embeddings).
3. **Agent orchestration** (planning + execution + feedback).
4. **Final research report**: design, experiments, evaluation (engagement, accuracy, adaptiveness).

# Evaluation Metrics

- **Adaptiveness**: % of recommendations aligned with student's weak areas.
- **Learning gain**: pre-test vs post-test improvement.
- **Engagement**: average session length, retention across sessions.
- **Explainability**: clarity of AI-generated justifications.

## Sample Input and Output for Endpoints

## POST /ask

- Student asks: *“I don’t understand conditional probability.”*
- **Planner Agent**: decides → retrieve explanation + generate an easy quiz.
- Calls **Retriever Agent** → returns explanation.
- Calls **Quiz Generator Agent** → returns 2 sample questions.


```
{
  "plan_executed": [
    "Identified weak topic: Probability",
    "Retrieved relevant notes",
    "Generated 2 practice questions"
  ],
  "retrieved_content": "Conditional probability is the likelihood of A given B...",
  "quiz": [
    {"id": 1, "question": "A coin is tossed twice..."},
    {"id": 2, "question": "If P(A)=0.3... what is P(A|B)?"}
  ],
  "natural_language_response": "I noticed you’re struggling with Probability. Let’s review the basics and try 2 quick practice questions."
}
```

## POST /quiz

- Student explicitly asks for a quiz.
- **Planner Agent**: checks progress → selects “medium” difficulty if student is weak, “hard” if strong.
- **Quiz Generator Agent** creates the quiz accordingly.

## GET /progress/{student_id}

- **Progress Tracker Agent** fetches profile.
- **Planner Agent** recommends next topic.


```
{
  "progress": {
    "Probability": {"accuracy": 50, "strength": "weak"},
    "Algebra": {"accuracy": 90, "strength": "strong"}
  },
  "agent_recommendation": "Focus next on Conditional Probability. I’ll prepare a mixed quiz for you tomorrow.",
  "natural_language_summary": "You’re strong in Algebra but need more practice in Probability. I’ll guide you through conditional probability next."
}
```