Challenge

Design and implement ProfAI — an AI-powered personalized learning platform — capable of adapting to each learner's needs via multi-stage assessments, intelligent lesson generation, and real-time sentiment analysis. The system needed to remain portable (JSON data layer, no database), maintain educational quality, and integrate multiple external AI services seamlessly — all in a one-day hackathon.

Tools Used

Frontend: React 18.2.0 + TypeScript, Vite 4.5.0, React Router DOM, CSS variables. Backend: Flask 3.0.0, Flask-CORS, OpenAI API (GPT-4 for complex reasoning, GPT-3.5-turbo for lighter tasks), Matplotlib, NumPy, Rich for CLI formatting. Data Layer: JSON files with atomic writes. External APIs: OpenAI (content & assessment), ElevenLabs (TTS), analytics modules.

Successes

Implemented a Retrieval-Augmented Generation (RAG) pipeline for lesson creation, adaptive assessment stages targeting knowledge gaps, and sentiment analysis to monitor engagement. Integrated animated analytics visualizations and established a clean separation between frontend and backend.

Challenges

Preserving performance and scalability with file-based JSON storage, ensuring RAG-generated lessons matched educational standards, balancing API usage costs with feature richness, and handling sentiment analysis in real time without introducing latency.

Timeline

Morning: Defined scope, set up repository, outlined architecture. Late Morning: Implemented backend Flask API with core AI engine and RAG pipeline. Afternoon: Built React + TypeScript frontend with state management and API integration. Evening: Added sentiment analysis, analytics visualization, testing, and deployment prep.