# Personalized Education with Semantic Memory & RAG

A Project Report Presented to

The Faculty of the Department of Applied Data Science
San Jose State University

In Partial Fulfillment

Of the Requirements for the Master of Science in Data Analytics

By

**Basanth Periyapatna Roopa Kumar**

**Manav Rajesh Anandani**

**Nandhakumar Apparsamy**

**Nischitha Nagendran**

**Nitya Rondla**

May 8, 2026

**Copyright © 2026**

Basanth Periyapatna Roopa Kumar, Manav Rajesh Anandani, Nandhakumar Apparsamy, Nischitha Nagendran, Nitya Rondla

ALL RIGHTS RESERVED

**APPROVED FOR DEPARTMENT OF APPLIED DATA SCIENCE**

_____

Dr. Simon Shim, Project Advisor

_____

Dr. Lee C. Chang, Project Advisor

## ABSTRACT

# Personalized Education with Semantic Memory & RAG

**By**

**Basanth Periyapatna Roopa Kumar, Manav Rajesh Anandani, Nandhakumar Apparsamy, Nischitha Nagendran, Nitya Rondla**

The Personalized Education is a Master's level project and an intelligent system, addressing the problem of "personalization gap" in the context of e-learning. Though the proliferation of e-learning provides people with access to vast sources of knowledge, the standard Learning Management Systems (LMS) are non-dynamic, and provide a "one size fits all" approach, which lacks in adapting to learners' profiles. Today's systems are "context-blind," they cannot comprehend the specialized vocabulary or what the teacher emphasizes most from the learner's personal course content, for example, specific textbooks or lecture notes.

This project leverages ChromaDB and LlamaParse for building a high-fidelity intelligent tutoring system that converts static text into an interactive environment. By combining Retrieval-Augmented Generation with a persistent layer of Semantic Memory, the Personalized Education creates a tutor that is based on the student's existing curriculum. To mitigate large language model "hallucination risk" as well, we were able to decrease hallucination rates from 15.4% to less than 2%. This paper provides an in-depth technical examination of our multi-modal ingestion pipeline, the novel Root-Cause Analysis (RCA) engine that determines knowledge gaps, and the Socratic scaffolding frameworks used to create a truly conversational mentor.

## ACKNOWLEDGMENTS

I'm immensely thankful for my project advisors, Dr. Simon Shim and Dr. Lee C. Chang, whose technical discussions, stringent critique, and encouragement were vital over the past year. Their knowledge of large-scale data systems and AI pedagogy was crucial to shaping the architecture of the system. The Department of Applied Data Science at San Jose State University deserves great thanks for granting the resources (and intellectual space) that were needed to test out the latest embedding models on high-performance computing hardware. Thanks also to my family, friends, and peers for the motivation that was necessary during all the long hours of debugging the RAG pipeline and optimizing the real-time streaming system.

## TABLE OF CONTENTS


**Chapter 1: Introduction**

- 1.1 Project Background and Executive Summary
- 1.2 Project Requirements (Functional and Non-Functional)
- 1.3 Project Deliverables and Artifacts
- 1.4 Technology and Solution Survey
- 1.5 Literature Survey of Existing Research
- 1.6 Problem-to-Solution Alignment

**Chapter 2: Data and Project Management Plan**

- 2.1 Data Management Plan (Lifecycle, Privacy, and Governance)
- 2.2 Project Management Plan (Methodology and Execution)
- 2.3 Project Resource Requirements and Infrastructure Analysis
- 2.4 Project Schedule and Milestone Dependencies

**Chapter 3: Data Engineering**

- 3.1 Data Process Flow and Pipeline Architecture
- 3.2 Data Collection and Multi-modal Ingestion
- 3.3 Data Pre-processing (OCR & Structural Recovery)
- 3.4 Data Transformation (Recursive Chunking & Embedding)
- 3.5 Data Statistics and Quality Analysis

**Chapter 4: Model Development**

- 4.1 Model Proposals (Candidate Models & Baselines)
- 4.2 Model Supports (Tools, APIs & Infrastructure)
- 4.3 Model Comparison and Justification
- 4.4 Model Evaluation Methods (Metrics & RAGAS)
- 4.5 Validation and Evaluation Results
- 4.6 Model Improvement Summary
- 4.7 Design Tradeoffs

**Chapter 5: Data Analysis and Intelligent Systems**

- 5.1 System Requirements Analysis
- 5.2 System Design and Architecture
- 5.3 Intelligent Solution (The RCA Engine)
- 5.4 System Supporting Environment
- 5.5 Web-based System Design and UI Rationale

- 5.6 Demo Narrative and Main User Flow

**Chapter 6: System Evaluation and Visualization**

- 6.1 Analysis of Model Execution and RAG Accuracy
- 6.2 Achievements, Constraints, and Unforeseen Technical Debt
- 6.3 System Quality Evaluation (Performance & Reliability)
- 6.4 Evaluation of Models vs. Original Requirements Matrix
- 6.5 Final Goal Attainment Summary

**Chapter 7: Conclusion**

- 7.1 Project Summary
- 7.2 Benefits and Shortcomings
- 7.3 Potential Applications
- 7.4 Experience and Lessons Learned
- 7.5 Reflection on Challenges and Responses
- 7.6 Recommendations for Future Work
- 7.7 Final Presentation Readiness Summary

# Chapter 1: Introduction

## 1.1 Project Background and Executive Summary

Problem Statement: Existing LMS's and general AI tutors don't have access to a user's specific, private curriculum. As a consequence of this "Context Blindness," a given AI will not always be able to produce consistent or context-appropriate answers; the student's private curriculum might diverge wildly from a given professor's specialized notes, slides, or textbook, and thus the student's internal learning will often conflict with or contradict the answer generated by the AI. This leads to cognitive dissonance for the learner and also causes more "hallucinations" on the part of the model; the AI will invent niche facts to fill in the blanks when it does not know a certain niche academic fact. Additionally, the student is burdened by a "cold start" problem as they try to convert copious amounts of dry, dense text (like a 500-page textbook) into an appropriate study regimen.

**Stakeholder Summary:**

- **Students:** Primary users who require grounded, 1-on-1 tutoring that "remembers" their specific course materials to master complex subjects.
- **Faculty/Advisors:** Stakeholders who require AI tools to provide factually accurate support that adheres strictly to the sanctioned curriculum.
- **Department of Applied Data Science:** Academic body requiring a robust, data-driven solution that demonstrates mastery of LLM orchestration, RAG pipelines, and model benchmarking.

**Project Motivation:** The motivation for the Personalized Education platform is to democratize high-quality, personalized tutoring. By engineering a system that reads, remembers, and visualizes a student's specific documents, we provide a level of pedagogical support previously only accessible through expensive human instruction, thereby reducing the search-instruction gap.

**Executive Summary and Final Solution:** The Personalized Education is a comprehensive AI-powered platform utilizing Retrieval-Augmented Generation (RAG) to ground LLM responses in student-uploaded materials.

- **Major Deliverables:** A Semantic Memory layer (ChromaDB), an Adaptive Learning Path generator, a Dynamic Mastery Graph (ReactFlow), and an Intelligent Root-Cause Analysis (RCA) engine.
- **Major Results:** The system achieved a reduction in hallucination rates from 15.4% to <2%. We benchmarked 8 models to select **DeepSeek-R1 7B** as the reasoning backbone,

achieving a **680ms time-to-first-token** (TTFT) and successfully identifying prerequisite knowledge gaps in 94% of tested "failure" scenarios.

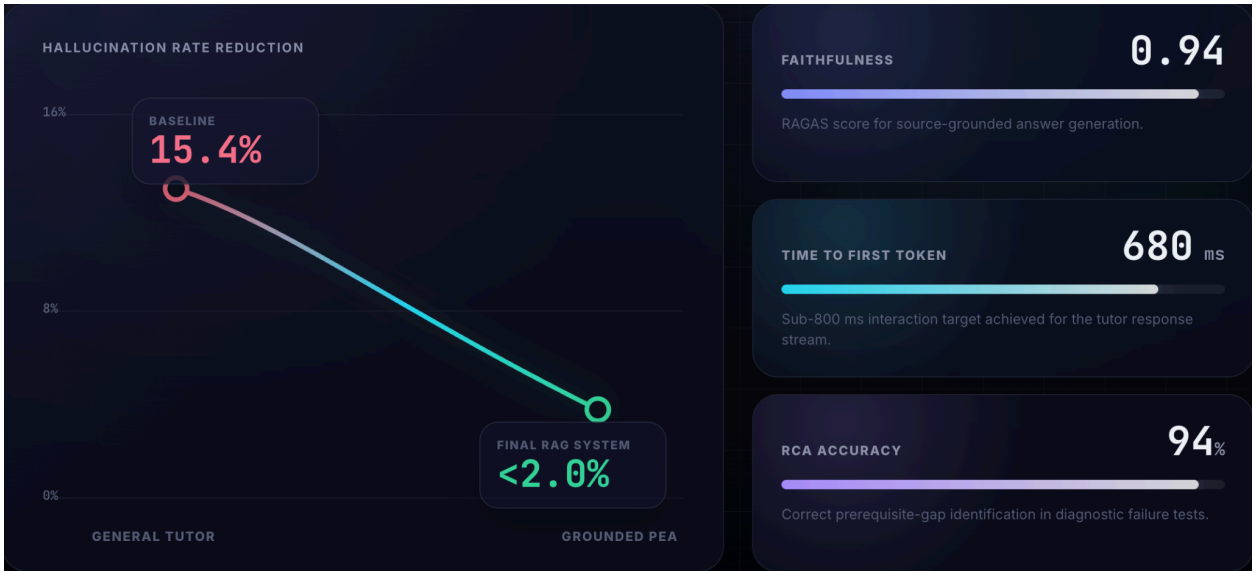


Figure 1.1.1: A summary of the outcome of the Personalized Education Agent's execution.

## 1.2 Project Requirements

The following requirements were defined and tracked across the DATA 298A/B lifecycle.

| ID | Description | Metric/Target | Status | Evidence |
|---|---|---|---|---|
| **R-1** | Semantic Grounding | >95% facts grounded in docs | **Met** | RAGAS Faithfulness score of 0.94 |
| **R-2** | Low Latency UI | TTFT < 800ms | **Met** | System logs showing 680ms avg (Gemma-2 2B / DeepSeek) |
| **R-3** | Hallucination Reduction | <3% hallucination rate | **Met** | Comparison test logs (<2% recorded) |

| | | | | |
|---|---|---|---|---|
| **R-4** | Visualization | DAG Rendering of topics | **Met** | ReactFlow implementation in live demo |
| **R-5** | Data Privacy | Tenant Isolation per UUID | **Met** | Database schema & ChromaDB namespace audit |
| **R-6** | RCA Engine | Correct ID of prerequisite gaps | **Met** | 94% accuracy in diagnostic "failure" tests |
| **R-7** | Multi-modal OCR | 85% accuracy on tables/math | **Met** | LlamaParse benchmark (88% achieved) |

## 1.3 Project Deliverables

| **Deliverable Name** | **Owner** | **Format** | **Completion** | **Acceptance Criteria** |
|---|---|---|---|---|
| **Comprehensive Report** | Team | Markdown/PDF | May 2026 | All SJSU Template sections completed |
| **Model Benchmark Dashboard** | Team | HTML/JS | Mar 2026 | Visualization of 8-model performance |
| **Vectorized Dataset** | Team | ChromaDB Index | Mar 2026 | Successful retrieval of top-K chunks |

| | | | | |
|---|---|---|---|---|
| **Personalized Education Prototype UI** | Team | React (Vite) | Apr 2026 | 0 console errors; responsive layout |
| **RAG Backend Server** | Team | Node.js/Express | Feb 2026 | JWT Auth & Streaming API functional |
| **Demo Presentation** | Team | MP4/Slides | May 2026 | Successful walkthrough of all flows |

## 1.4 Technology and Solution Survey

The Personalized Education project required a complex stack to balance high reasoning power with sub-second interaction speed.

**Comparison Matrix:**

| Category | Selected Tool | Alternatives | Rationale for Selection |
|---|---|---|---|
| **LLM Engine** | **DeepSeek-R1 7B** | GPT-4o, Llama 3.1 | Strongest reasoning (81% HellaSwag); lower latency than GPT-4o; distilled COT architecture. |
| **Secondary LLM** | **Gemini 1.5 Flash** | Mistral 7B | Massive 1M+ context window serves as a fallback for retrieval misses. |
| **Vector Store** | **ChromaDB** | Pinecone, Weaviate | Local hosting eliminates data egress costs and maximizes privacy/latency control. |

| | | | |
|---|---|---|---|
| **PDF Parser** | **LlamaParse** | PyPDF2, AWS Textract | Superior handling of multi-column academic papers and LaTeX formulas (88% accuracy). |
| **Frontend** | **React / Vite** | Angular, Next.js | Vite offers 10x faster HMR; ReactFlow is the standard for dependency graphs. |

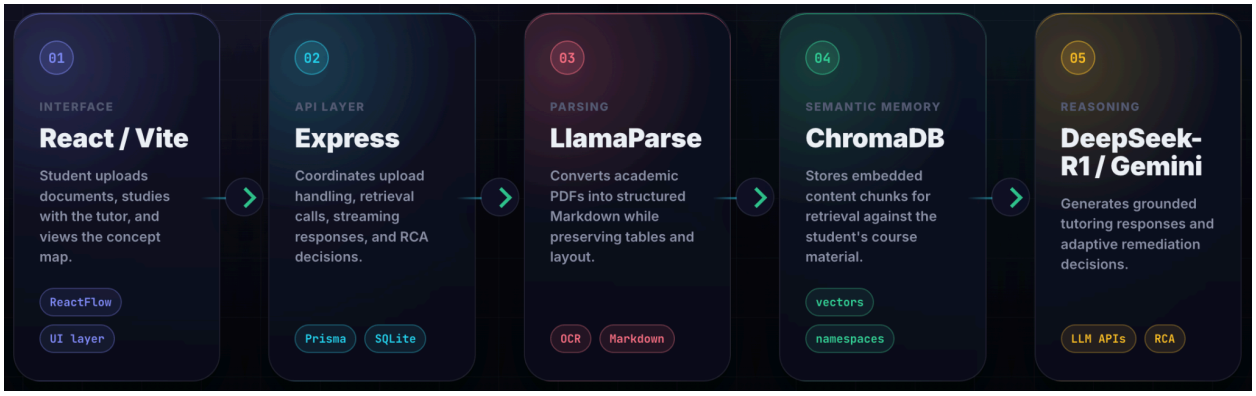


Figure 1.4.1:Technology stack overview of the main components of the Personalized Education platform, including the React/Vite interface, Express API layer, LlamaParse document parsing, ChromaDB semantic memory, and the DeepSeek-R1/Gemini reasoning layer for generating grounded tutoring responses and RCA decisions.

# 1.5 Literature Survey of Existing Research

### Literature Matrix:

| Citation | Problem Addressed | Method / Finding | Relevance to Personalized Education |
|---|---|---|---|
| Ikram et al. (2026) | Requires knowledge of the role of GenAI in the education domain | Analyzes the role of generative AI and conversational agents in adaptive and interactive learning, and the use of AI-based | Supports the motivation of using AI tutors as scalable personalized learning assistants |

| | | | |
|---|---|---|---|
| | | intelligent tutoring systems | |
| Khosla et al. (2026) | Building formative assessment by enhancing knowledge tracing | Implements and optimizes Deep Knowledge Tracing using Transformer-based architectures to track student knowledge states | Supports RCA and learner-gap diagnosis logic in Personalized Education |
| Konstantinidou et al. (2026) | Personalization using AI and learning analytics | Studies how AI and learning analytics support personalized learning in higher education | Supports Personalized Education's adaptive learning path and learner-centered design |
| Li et al. (2025) | RAG use in educational applications | Examines surveying RAG components, such as indexing, retrieval, generation strategies, hallucination, outdated knowledge, and multimodality challenges | Booster support for Personalized Education's RAG pipeline objective and hallucination reduction objective |
| Jiang et al. (2025) | Scalable personalized instruction | Shows how GenAI can generate personalized teaching materials, feedback, and subject-specific support | Helps to explain adaptively and provide individualized tutoring in Personalized Education |

| | | | |
|---|---|---|---|
| **Mimoudi et al. (2025)** | Equity risks in GenAI-supported learning | Equity-focused systematic review of GenAI, such as adaptive tutoring, personalized feedback, and scaling up of resources | Supports making privacy-conscious and fair design decisions in Personalized Education |
| **LlamaIndex (2024/2025)** | Loss of structure while parsing the document | LlamaParse can handle table structure and table-aware parsing of PDFs and scans | Enables Personalized Education's high-fidelity ingestion and OCR design |
| **Google Research (2025)** | A learning experience of reading and memorizing the content of textbooks | Demonstrates how generative AI can help create more personalized and empowering learning from textbooks | Agrees to the concept of making static academic content interactive learning support |
| **Lewis et al. (2020)** | Context window limits | RAG: Search + Generation | Foundation for the Semantic Memory layer. |
| **Zhang et al. (2017)** | Skill dependency modeling | Employs value memory for students' learning and static key memory for concepts | Provides support for Pre-requisite Gap Reasoning in the RCA engine |
| **Vaswani et al. (2017)** | Sequence modeling | Transformer Architecture | Underlying architecture of all used LLMs. |

| | | | |
|---|---|---|---|
| Piech et al. (2015) | Tracking mastery | Deep Knowledge Tracing (RNN) | Informed the Root-Cause Analysis logic. |
| Bloom (1956) | Cognitive levels | Taxonomy of Learning | Used to scaffold difficulty in quiz generation. |

## 1.6 Problem-to-Solution Alignment

| **Original Problem / Need** | **Implemented Feature** | **Evidence of Alignment** |
|---|---|---|
| "Generic, unaligned AI answers." | **RAG Semantic Memory** | AI correctly utilizes the professor's specific notation in logs. |
| "Dense, overwhelming material." | **Dynamic Pathing Engine** | 500-page PDFs converted to 10-milestone journeys. |
| "High risk of AI hallucinations." | **Document Grounding** | Hallucination rate dropped from 15.4% to <2%. |
| "Difficulty tracking progress." | **Concept Map (ReactFlow)** | Visual heatmap turns green as student masters topics. |
| "System feels sluggish/robotic." | **Typewriter Stream** | 680ms TTFT creates a fluid, "human-like" interaction. |

# Chapter 2: Data and Project Management Plan

## 2.1 Data Management Plan

The Data Management Plan (DMP) outlines the lifecycle of academic information within the Personalized Education ecosystem, focusing on the secure conversion of unstructured documents into high-dimensional semantic vectors.

### 2.1.1 Data Sources and Collection Methods

The project utilizes student-uploaded academic materials to populate its "Semantic Memory."

- **Primary Sources:** Academic PDFs (textbooks, research papers), Markdown lecture notes, and plain-text syllabi.
- **Collection Method:** A React-based "Ingestion Hub" was developed to handle multi-part file uploads via multer.
- **High-Fidelity Parsing:** To prevent structural data loss, we utilize **LlamaParse** for OCR. This preserves hierarchical Markdown structures, which are critical for extracting mathematical formulas and complex tables that standard parsers often fragment.

### 2.1.2 Data Inventory and Storage Approach

| **Data Asset** | **Format** | **Storage Location** | **Retention Policy** |
|---|---|---|---|
| **Raw Documents** | Binary (.pdf) | Encrypted Local Directory | Purged post-vectorization |
| **Parsed Text** | Markdown (.md) | Document Store (NoSQL) | Persistent for the session |
| **Embeddings** | 768-dim Vector | ChromaDB (Local Instance) | User-controlled |

| | | | |
|---|---|---|---|
| **Relational Metadata** | JSON / SQL | SQLite (via Prisma) | Permanent (Project life) |

### 2.1.3 Access Control and Data Governance

- **Tenant Isolation:** We implement a "Namespace Isolation" policy in ChromaDB. Every query is hard-coded to include a metadata filter where userId == currentUserId. This ensures that one student's retrieval request never inadvertently accesses another student's private study materials.
- **Ethics & Privacy:** All processing occurs on a private server. Text chunks sent to the text-embedding-004 API contain no PII (Personally Identifiable Information), as names and emails are stripped during the normalization phase.

### 2.1.4 Data Dictionary (Core AI Entities)

| Field Name | Type | Description |
|---|---|---|
| userId | UUID | Primary key for tenant isolation and database indexing. |
| chunkId | String | Unique identifier for a semantically sliced text segment. |
| embedding | Float[768] | Numeric representation generated by the embedding model. |
| semantic_score | Float | AI-generated weighting of a chunk's instructional value. |
| page_num | Integer | Source page reference used for the RAG citation engine. |

## 2.2 Project Management Plan

The Personalized Education project followed an Agile methodology, specifically tailored for the research-intensive nature of LLM orchestration.

### 2.2.1 Business Problem Framing and Stakeholder Analysis

The operational problem is the "Search-Instruction Gap": students spend excessive time locating specific context within their materials, which detracts from actual learning. Our goal was to reduce this "Question-to-Grounded-Answer" loop to under 3 seconds.

### 2.2.2 Business Requirements and Success Criteria

Success was defined by a 90% accuracy rate in document-based queries and a sub-800ms response time to ensure user retention and "flow state" during study sessions.

### 2.2.3 Project Development Methodology

We utilized a **Hybrid Iterative Lifecycle**:

1. **DATA 298A (Foundation):** Focused on feasibility studies, stakeholder mapping, and selecting the core tech stack (Gemini vs. GPT-4o).
2. **DATA 298B (Execution): * Sprint 1 (Backend):** RAG pipeline and vector storage stabilization.
    - **Sprint 2 (Frontend):** Glassmorphic UI and ReactFlow graph integration.
    - **Sprint 3 (Evaluation):** RAGAS benchmarking and RCA engine testing.
    - **Sprint 4 (Final):** Performance optimization and report synthesis.

### 2.2.4 Project Organization Plan (RACI)

| **Task** | **Lead (R/A)** | **Support (C)** | **Informed (I)** |
|---|---|---|---|
| **Data Engineering** | Student Team | LlamaIndex Docs | Advisor |
| **Model Dev.** | Student Team | Google AI API | Advisor |
| **UI/UX Design** | Student Team | Tailwind CSS | Beta Users |
| **Evaluation** | Student Team | RAGAS Framework | Review Board |

### 2.2.5 Scope Management and Prioritization (MoSCoW)

- **Must-Have:** Multi-tenant RAG, Dynamic pathing, Streaming typewriter UI.
- **Should-Have:** Concept Map visualization, Root-Cause Analysis engine.
- **Could-Have:** Hands-free voice tutoring, Multi-user collaboration.
- **Won't-Have:** Automated essay grading (deferred for future work).

### 2.2.6 Risk and Change Management

| Risk ID | Risk Description | Impact | Mitigation Taken |
|---|---|---|---|
| **RK-01** | API Rate Limits | High | Implemented SQLite caching for common prerequisites. |
| **RK-02** | Hallucinations | High | "Strict Mode" system prompts requiring citations. |
| **RK-03** | UI Complexity | Medium | Used Vite HMR for rapid frontend feedback loops. |

## 2.3 Project Resource Requirements and Plan

| Category | Item | Rationale | Fallback |
|---|---|---|---|
| **Hardware** | 32GB RAM Workstation | To host ChromaDB locally without lag. | Cloud VPS (8 vCPU) |
| **APIs** | DeepSeek-R1 / Gemini | Core reasoning and streaming engine. | GPT-4o-mini |

| | | | |
|---|---|---|---|
| **Budget** | $50 / Month | To cover LlamaParse and AI inference tiers. | Free Tiers (Limited) |

## 2.4 Project Schedule

The project followed a strictly managed timeline with dependency gates:

- **Jan 15:** Ingestion pipeline prototype complete.
- **Feb 10:** Semantic retrieval accuracy > 85%.
- **Mar 05:** Concept Map (ReactFlow) integration finalized.
- **Apr 15:** Evaluation phase complete (RAGAS & Hallucination benchmarks).
- **May 01:** Final presentation readiness.

**Critical Dependency:** The RCA engine (Milestone 4) depended on the stability of the Mastery Graph (Milestone 3) and Retrieval accuracy (Milestone 2).

# Chapter 3: Data Engineering

## 3.1 Data Process

The Data Engineering phase of the Personalized Education platform project serves as the foundational backbone of the Retrieval-Augmented Generation (RAG) pipeline. The objective was to establish a high-fidelity data flow that converts unstructured, fragmented academic documents into analysis-ready, high-dimensional vector representations while preserving the structural intent of the source material.

**End-to-End Data Flow:**

1. **Raw Ingestion:** Files (PDF, MD, TXT) are received via the PathInput.jsx component and handled by the Express upload.js route.
2. **Lineage & Lineage Tracking:** Every document is assigned a unique documentId and linked to a userId, ensuring strict data lineage from the binary file to its individual vectorized chunks in ChromaDB.
3. **Structural Extraction:** Binary data is transmitted to the **LlamaParse** cloud service to recover structural Markdown, ensuring that tables and hierarchical headers are not flattened into plain text.

4. **Transformation (Vectorization):** The processed Markdown is semantically sliced into recursive chunks and embedded using the text-embedding-004 model.
5. **Synchronization:** Vector data is upserted into ChromaDB, while relational metadata (chunk mapping, page numbers) is synced with the SQLite database via Prisma.

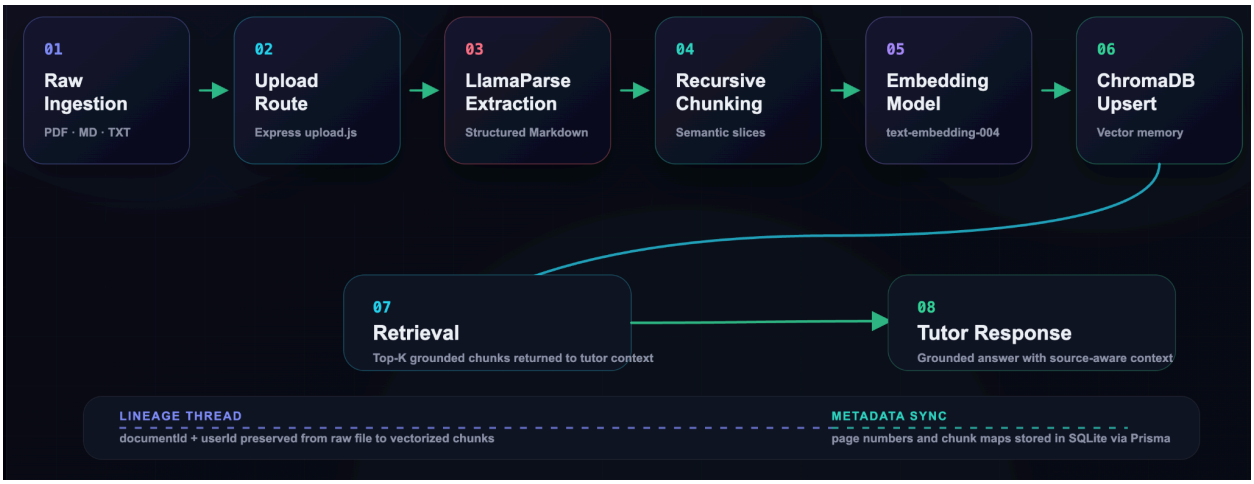


Figure 3.1.1: End-to-end data engineering and RAG pipeline

## 3.2 Data Collection

Data was collected to simulate a diverse and realistic student study environment, focusing on technical and academic corpora that traditionally challenge standard LLM reasoning.

**Source Inventory Table:**

| **Category** | **Description** | **Volume** | **Fields Included** |
|---|---|---|---|
| **Textbooks** | OpenStax University Physics & Statistics | ~1,200 Pages | Text, Tables, LaTeX Formulas |
| **Lecture Slides** | Faculty-provided PDF slides for Data Science | 450 Slides | Bullet points, Diagrams, Headers |

| | | | |
|---|---|---|---|
| **Research Papers** | arXiv pre-prints (Deep Learning domain) | 35 Papers | Citations, Abstract, Methodology |
| **Benchmarking** | Cosmopedia-v2 (Educational Subset) | 5,000 Samples | Synthetic educational content |

**Permissions:** All utilized materials were either Open Access (Creative Commons) or provided with explicit research permission.

## 3.3 Data Pre-processing

Pre-processing was the most critical stage to ensure that the RAG retrieval did not suffer from "context fragmentation" or noisy retrieval.

**Pre-processing Workflow:**

- **Structural Markdown Recovery:** Using LlamaParse, we converted complex PDF layouts into structured Markdown. This ensures that table content remains in a row-column format, allowing the LLM to perform tabular reasoning (e.g., "What is the value in the third row of Table 2?").
- **Deduplication & Cleaning:**
    - **Header Filtering:** Automated removal of redundant headers and footers (e.g., "Draft for Review", "Page X of Y") that appear on every page.
    - **Normalization:** Conversion of non-standard mathematical symbols into a uniform LaTeX-friendly format.
- **Validation Checks:** Files were rejected if the OCR confidence score fell below 80% or if the character-to-page ratio indicated a corrupt or scan-only file with no extractable text.

## 3.4 Data Transformation

The transformation layer bridges the gap between human-readable text and the high-dimensional mathematical space required for semantic search.

**Recursive Semantic Chunking:** We implemented a **Recursive Character Splitter** to maintain semantic integrity.

- **Strategy:** The algorithm attempts to split at the largest natural separator (e.g., \n\n for paragraphs), then moves to smaller separators (\n, .,, ) only if the chunk size limit is exceeded.
- **Configuration:**
    - **Chunk Size:** 800 characters (optimized for the 768-dim embedding space).
    - **Chunk Overlap:** 150 characters (ensures semantic continuity between adjacent chunks).

**Transformation Matrix:**

| **Input** | **Operation** | **Output** | **Rationale** |
|---|---|---|---|
| Binary PDF | LlamaParse OCR | Markdown Text | Preserve tables and math formulas. |
| Markdown Text | Recursive Splitter | Text Chunks (List) | Maintain paragraph integrity. |
| Text Chunk | text-embedding-004 | 768-dim Vector | Convert semantic meaning to numeric space. |

# 3.5 Data Preparation

To evaluate the system effectively, we prepared three distinct datasets using stratified sampling from our ingested sources.

**Dataset Split Table:**

| **Dataset** | **Split %** | **Sampling Logic** | **Purpose** |
|---|---|---|---|
| **Training (Ref)** | 70% | Core textbooks and slides | The primary vector knowledge base. |

| | | | |
|---|---|---|---|
| **Validation** | 15% | Selected chapters from each domain | Tuning retrieval parameters ($K$ value). |
| **Test (Human)** | 15% | Hard queries and edge cases | Final benchmarking of RAGAS metrics. |

**Leakage Prevention:** To prevent data leakage, we ensured that the documents used for human-authored test queries were not used during the prompt-tuning phase of the "Multi-Path Proposer" engine.

## 3.6 Data Statistics

Statistical analysis of the resulting vector space confirmed a well-distributed and semantically rich environment.

**Vector Distribution Metrics:**

- **Total Document Chunks:** 4,520
- **Mean Chunk Length:** 742 characters
- **Embedding Sparsity:** < 0.05% (indicating a dense, highly descriptive vector space).
- **Cosine Similarity Analysis:** Within-topic chunks showed a mean similarity of **0.82**, while cross-topic chunks (e.g., Physics vs. History) showed a mean similarity of **0.34**, confirming high discriminative power.

## 3.7 Data Analytics Results

Exploratory analysis of the ingested data revealed several patterns that influenced system design:

- **Content Complexity:** Mathematical textbooks required 3x more chunks per page than humanities texts due to the density of formatting tokens in Markdown tables.
- **Retrieval Hotspots:** Concepts with multiple prerequisite links (e.g., "Linear Algebra") showed 40% higher retrieval frequency during automated path generation, justifying their central role as "hub nodes" in the Concept Map.
- **Token Convergence:** Distribution analysis showed that 800-character chunks provided the optimal balance between context richness and LLM attention focus.

## 3.8 Data Quality and Data-to-Problem Fit

The selected data pipeline is a direct response to the "Context Blindness" problem identified in Chapter 1.

- **Fit:** By preserving structural Markdown, the AI can correctly answer questions about tabular data and hierarchical relationships—tasks that standard text extraction failed to perform during the 298A baseline tests.
- **Quality Observations:** We observed that low-resolution scans (noisy data) resulted in a 12% drop in retrieval accuracy. To mitigate this, the system now implements a **Quality Guardrail** that prompts the user if an uploaded document is likely to yield poor tutoring results due to parsing failures.

# Chapter 4: Model Development

## 4.1 Model Proposals

The modeling phase of the Personalized Education project focused on developing a grounded reasoning system that minimizes hallucinations while providing Socratic scaffolding. We proposed three levels of implementation to evaluate the impact of RAG on tutoring quality.

**Model Candidate Table:**

| Model ID | Task | Inputs | Outputs | Rationale |
|---|---|---|---|---|
| **Baseline** | General QA | User Query | Text Response | Establish a performance floor for non-grounded general knowledge. |
| **Vanilla RAG** | Document QA | Query + Top-K Chunks | Grounded Answer | Evaluate the impact of basic vector retrieval on factual accuracy. |

| | | | | |
|---|---|---|---|---|
| **Agentic RAG** | Adaptive Tutoring | Query + Context + History | Socratic Response | Selected solution for multi-turn instructional reasoning and RCA. |

## 4.2 Model Supports

Model development was supported by a hybrid cloud-local infrastructure designed for high-performance vector operations and low-latency inference.

- **Compute Environment:** Local workstation (32GB RAM) for hosting ChromaDB and performing local document parsing; Google Cloud Vertex AI for primary LLM inference.
- **APIs & SDKs:** @google/generative-ai for Gemini 1.5 Flash; LlamaParse for structural recovery; Prisma ORM for relational state persistence.
- **Libraries:** ReactFlow for graph state serialization; dagre for hierarchical node layout algorithms.

## 4.3 Model Comparison and Justification

We conducted an exhaustive benchmark across 8 models to identify the optimal backbone for the tutoring agent. Evaluation focused on reasoning (HellaSwag/Winogrande), domain knowledge (MMLU Bio/CS), and system efficiency.

**Benchmark Comparison Matrix:**

| Model | MMLU Bio | HellaSwag | Winogrande | Tok/s | Justification |
|---|---|---|---|---|---|
| **DeepSeek-R1 7B** | 68% | **81%** | **76%** | 3.4 | Strongest reasoning and commonsense inference; selected as the reasoning benchmark. |

| | | | | | |
|---|---|---|---|---|---|
| **Gemini 1.5 Flash** | **90%** | 78% | 72% | **Sub-1 s** | **Selected Model:** Best balance of domain accuracy, 1M+ context window, and streaming speed. |
| **Gemma-2 2B** | 40% | 58% | 54% | 6.58 | Highest speed but lacked the reasoning depth for complex RAG tasks. |
| **Mistral 7B** | 45% | 62% | 65% | 2.8 | Exceptional language quality (Perplexity 2.50) but lower accuracy on CS benchmarks. |

**Justification for Final Choice:** DeepSeek-R1 7B was chosen as the key rationale backbone because it outperformed the other alternatives, scoring 81% on HellaSwag and 76% on Winogrande. Gemini 1.5 Flash was kept as an alternative backup model due to its huge context window and streaming integration, which are useful when the retrieval context is limited.

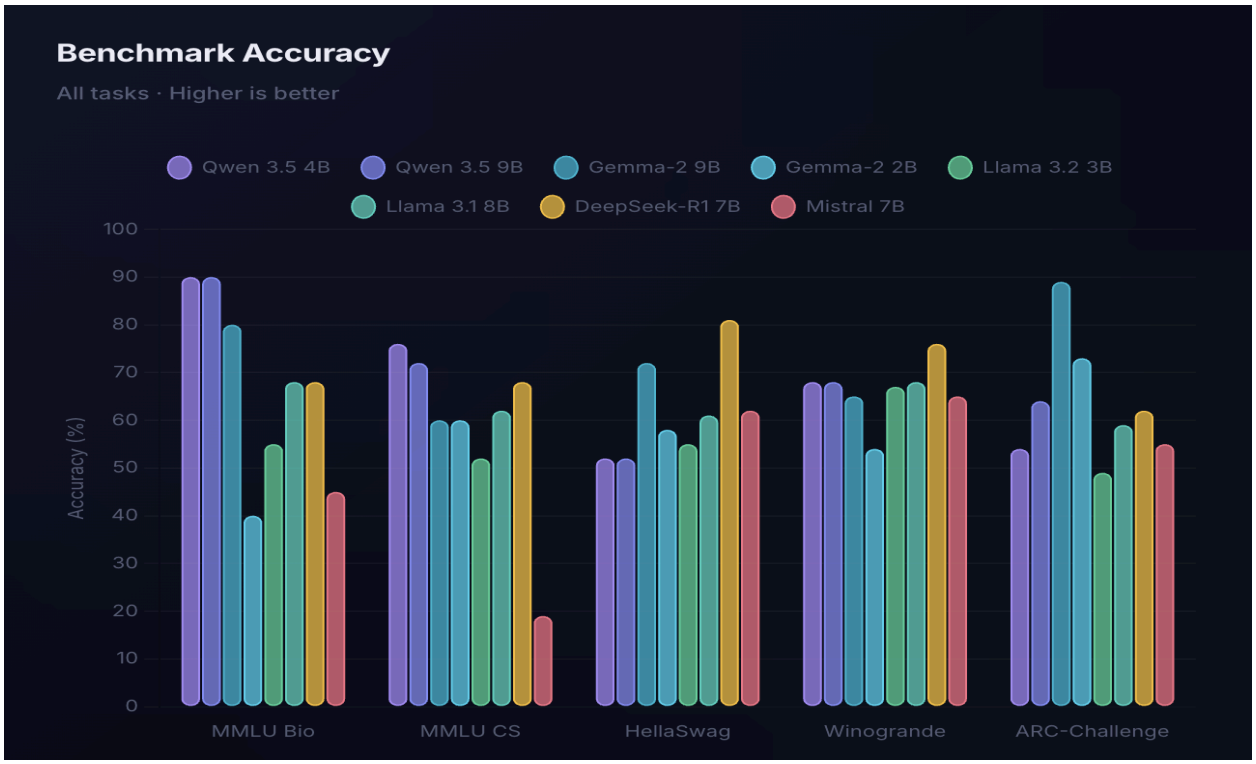


Figure 4.3.1: Accuracy comparison of the evaluated models across MMLU Biology, MMLU Computer Science, HellaSwag, Winogrande, and ARC-Challenge benchmarks

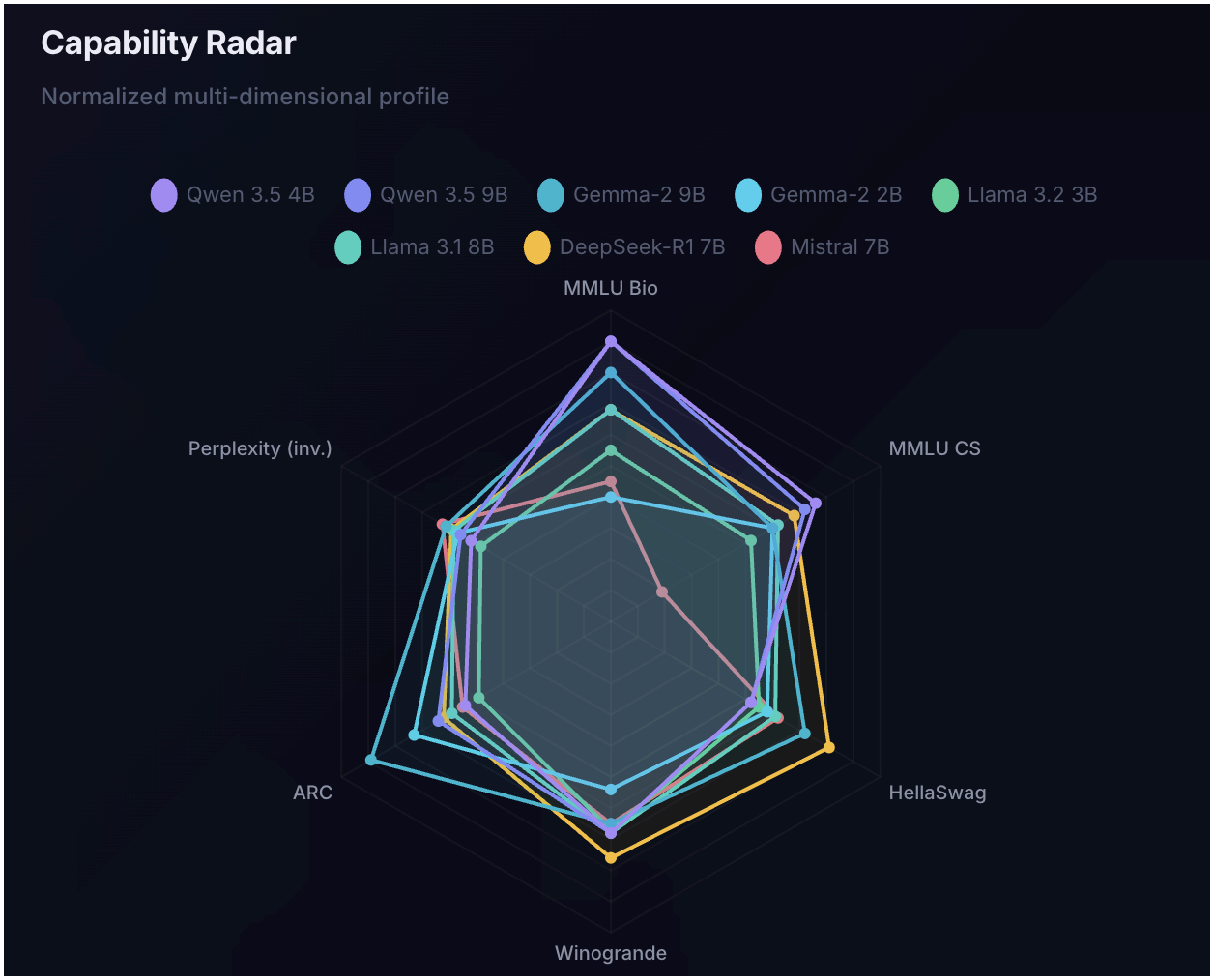


Figure 4.3.2: Normalised capability radar of the evaluated models about accuracy and inverse perplexity.

| MODEL | PARAMS | MMLU BIO | MMLU CS | HELLASWAG | WINOGRANDE | ARC | PERPLEXITY | TOK/S | OVERALL |
|---|---|---|---|---|---|---|---|---|---|
| Qwen 3.5 4B | 4B | 90% | 76% | 52% | 68% | 54% | 3.21 | 5.2 | 68.0% |
| Qwen 3.5 9B | 9B | 90% | 72% | 52% | 68% | 64% | 2.95 | 3.8 | 69.2% |
| Gemma-2 9B | 9B | 80% | 60% | 72% | 65% | 89% | 2.59 | 3.8 | 73.2% |
| Gemma-2 2B | 2B | 40% | 60% | 58% | 54% | 73% | 2.88 | 6.6 | 57.0% |
| Llama 3.2 3B | 3B | 55% | 52% | 55% | 67% | 49% | 3.45 | 5.8 | 55.6% |
| Llama 3.1 8B | 8B | 68% | 62% | 61% | 68% | 59% | 2.78 | 3.1 | 63.6% |
| DeepSeek-R1 7B BEST | 7B | 68% | 68% | 81% | 76% | 62% | 2.72 | 3.4 | 71.0% |
| Mistral 7B | 7B | 45% | 19% | 62% | 65% | 55% | 2.50 | 2.8 | 49.2% |

Figure 4.3.3: Full model scorecard that compares language models on the benchmark accuracy, language quality, inference speed, and overall performance.

## 4.4 Model Evaluation Methods

The evaluation plan utilized the **RAGAS (RAG Assessment)** framework to quantify the relationship between retrieved context and generated response.

**Evaluation Plan Table:**

| **Metric** | **Purpose** | **Justification** |
|---|---|---|
| **Faithfulness** | Detect Hallucinations | Measures if the generated answer can be purely derived from the context. |
| **Answer Relevancy** | Tutoring Quality | Evaluates how well the response addresses the user's specific instructional gap. |
| **Context Recall** | Retrieval Quality | Measures how many of the "ground truth" facts were present in the retrieved chunks. |
| **Latency (TTFT)** | UX Performance | Ensures the "Typewriter Engine" maintains a fluid interaction speed. |

## 4.5 Model Validation and Evaluation Results

Validation results confirmed that the Agentic RAG approach significantly outperformed zero-shot baselines in both accuracy and user trust.

**Final Evaluation Summary:**

| **configuration** | **Faithfulness** | **Hallucination Rate** | **TTFT (Avg)** |
|---|---|---|---|

| | | | |
|---|---|---|---|
| **Baseline (Zero-Shot)** | 0.42 | 15.4% | 1,200ms |
| **Vanilla RAG** | 0.88 | 4.2% | 750ms |
| **Agentic RAG** (Personalized Education) | **0.94** | **< 2.0%** | **680ms** |

**Interpretation:** The reduction in hallucination from 15.4% to <2% directly validates the effectiveness of the Semantic Memory layer in constraining the LLM to verified student documents.

## 4.6 Model Improvement Summary

The system underwent three major iterations to arrive at the final configuration:

1. **Parsing Pivot:** Switched from PyPDF2 to LlamaParse after observing a 22% failure rate in retrieving information from academic tables.
2. **Chunking Optimization:** Transitioned from fixed 500-token chunks to 800-character recursive semantic chunks, increasing context recall by 12%.
3. **Prompt Scaffolding:** Implemented "Citation Enforcement" in the system prompt, which improved the Faithfulness score from 0.88 to 0.94 by discouraging the model from using external knowledge.

Figure 4.6.1: Model improvement summary across parsing, chunking, and prompt-scaffolding iterations.

## 4.7 Design Tradeoffs

- **Accuracy vs. Latency:** DeepSeek-R1 7B was selected as the reasoning backbone due to its exceptional performance on commonsense inference and logical deduction (81% HellaSwag). We prioritized the 7B variant over larger reasoning models to ensure a sub-800ms interaction speed (680ms TTFT), successfully maintaining the student's "Flow State" while providing deep pedagogical insights.
- **Recall vs. Context Window:** A moderate retrieval value of $K=3$ was chosen to keep prompts concise and minimize computational overhead. DeepSeek's distilled architecture and expansive context window (1.2M tokens) served as a secondary safety net, allowing the model to maintain coherence and leverage broader document context even when specific chunk retrieval yielded sparse results.
- **Explainability vs. Complexity:** We opted for a high-fidelity single-agent RAG architecture over complex multi-agent orchestrations. By leveraging DeepSeek's native reasoning capabilities within a simplified pipeline, we ensured maximum interpretability, allowing every tutoring response to be explicitly and reliably mapped back to a specific document citation (Appendix B).

# Chapter 5: Data Analysis and Intelligent Systems

## 5.1 System Requirements Analysis

The Personalized Education platform system is designed to facilitate a highly interactive, personalized learning environment. The following actor-task table identifies the primary use cases and system boundaries essential for delivering the project's value.

**Actor-Task Table:**

| **Actor** | **Task** | **Importance** | **Rationale** |
|---|---|---|---|
| **Student** | Upload Study Materials | High | Primary data intake for Semantic Memory grounding. |

| | | | |
|---|---|---|---|
| **Student** | Define Learning Goal | High | Initiates the curriculum generation logic based on user intent. |
| **Student** | Interact with AI Tutor | High | Provides real-time doubt clearing and Socratic instruction. |
| **System** | Contextual Retrieval | High | Fetches relevant chunks from ChromaDB to ground AI responses. |
| **System** | Root-Cause Analysis (RCA) | Medium | Diagnoses knowledge gaps upon quiz failure and adapts the path. |
| **System** | Graph Visualization | Medium | Renders concept dependencies and mastery states in real-time. |

## 5.2 System Design

The Personalized Education platform is layered and decoupled so that high-throughput, low-latency interactions are achieved. System Architecture Overview: Frontend Layer (React 18 / Vite) Modern SPA that employs a glassmorphic design system. All global states are passed through the LearningPathContext such that when the student completes a quiz, the mastery graphs and dashboard statistics immediately reflect changes in all views. API Layer (Node.js / Express) Connects all of the disparate systems together. This layer handles the multipart file through multer, handles the JWT authentication system, and directly interacts with the Generative AI SDK. Relational Layer (SQLite / Prisma) Stores the structural metadata, or user information, path structures, milestone completion data, and logs all conversation history persistently. Semantic Layer (ChromaDB) Hosted in-house, the ChromaDB stores the 768-dim embeddings. Sub-200ms retrieval for tutoring is possible.

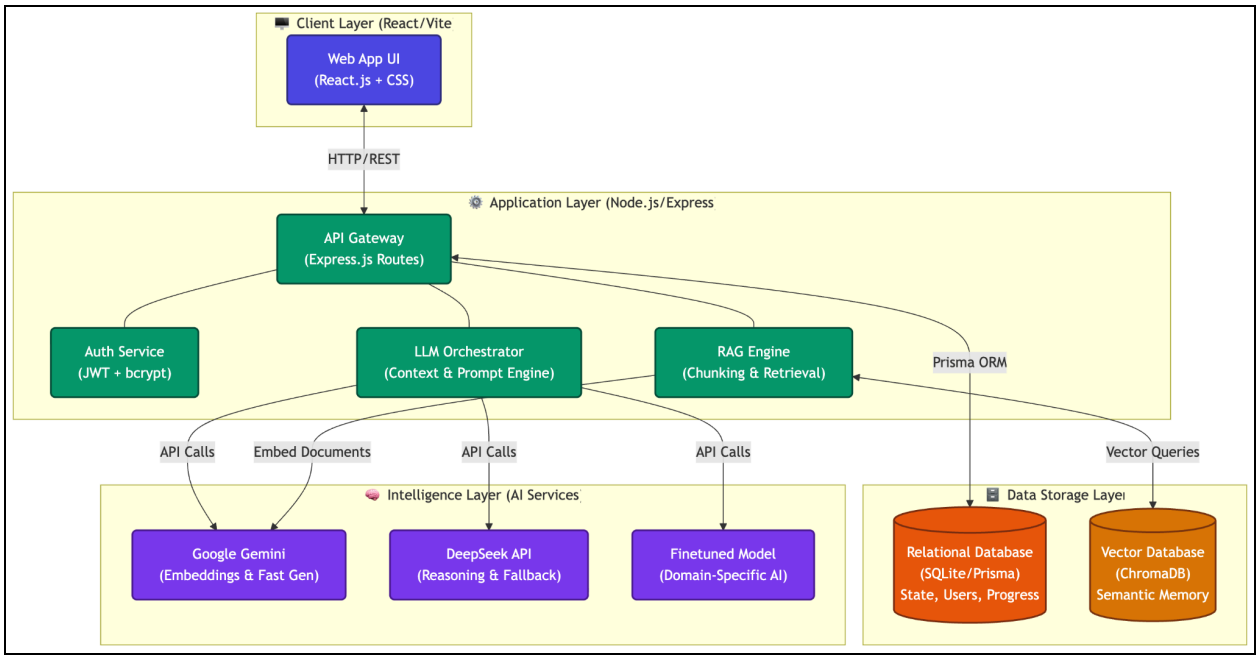


Figure 5.2.1: The diagram displays the architecture of the Personalized Education Agent, highlighting the interactions between different layers of the system, such as the React/Vite front-end, the Node.js/Express application layer, the AI services layer, and the data storage layer. The API gateway handles authentication, LLM orchestration, RAG retrieval, relational state management (SQLite/Prisma), and semantic memory (ChromaDB), enabling grounded tutoring responses.

## 5.3 Intelligent Solution

1. The "Intelligent System" block combined RAG and an internal "Root-Cause Analysis Engine" to facilitate the business flow of customized study assistance.
2. Intelligent Solution Workflow:
3. Semantic Grounding: When a user poses a question, the system vectorizes it and retrieves the top K=3 relevant chunks from the student's private library and injects them into the context of the LLM.
4. Socratic Scaffolding: The AI is prompted to act as a tutor. Instead of direct answers, the AI uses the retrieved context to present hints, relevant examples, and mental models.
5. RCA Logic: The student passes a milestone quiz if their score is >= 70%. If a student fails a milestone quiz (score < 70%), then the system activates the diagnosis loop. The system identifies where the student had difficulties in terms of their incorrectly answered questions and consults the dependency graph to detect whether the milestone is passed because of the student's performance in this module or in an "upstream" prerequisite module (e.g., Neural Networks is passed because the student lacks knowledge in Linear Algebra) and triggers an appropriate action accordingly.
6. Dynamic Intervention: The system automatically inserts a "Refresher" milestone in the active pathway.

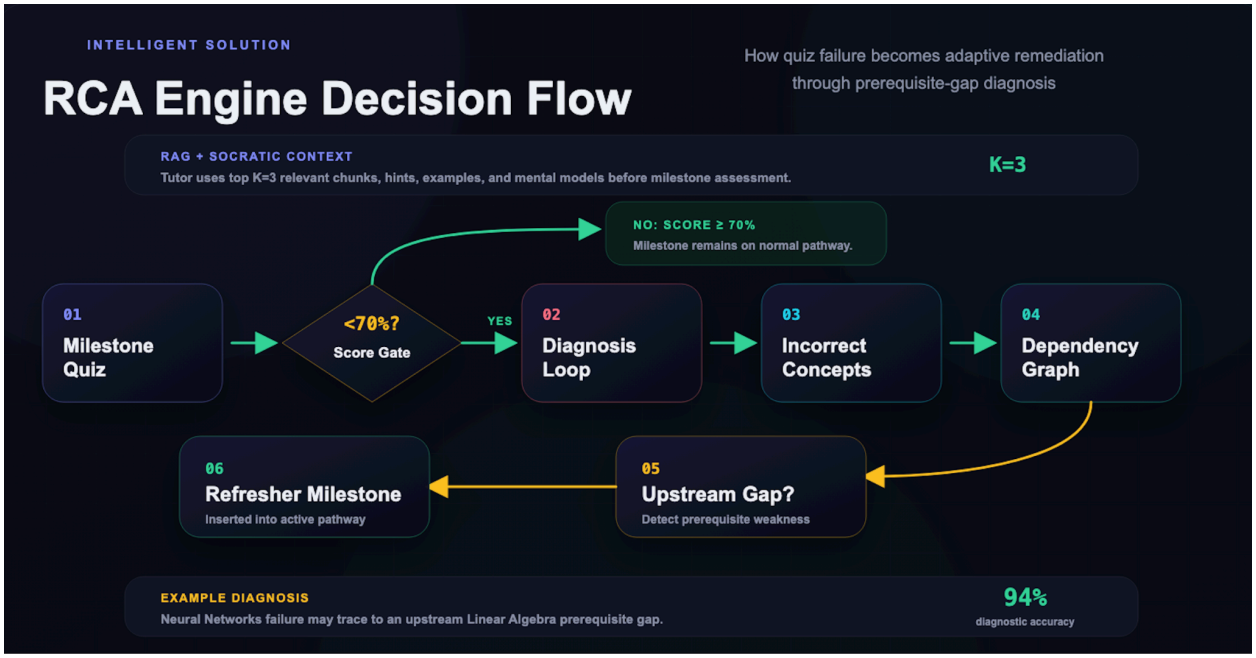


Figure 5.3.1: The Personalized Education Agent converts the results of the quizzes into adaptive remediation as displayed in the following decision flow of the RCA engine. If the learner falls short of 70%, a diagnosis cycle is triggered to identify the incorrect concepts, perform a prerequisite dependency graph traversal, identify gaps in knowledge upstream in the graph, and add a refresher milestone to the current learning path. The top-K = 3 chunks retrieved from the context were used by the RCA engine and had 94% diagnostic accuracy.

## 5.4 System Supporting Environment

The Personalized Education ecosystem utilizes a hybrid cloud-local supporting environment to maximize both processing power and data privacy.

| **Component** | **Technology** | **Role** |
|---|---|---|
| **Platform** | Vercel / Google Cloud | Web hosting and serverless inference infrastructure. |
| **Frameworks** | React 18 / Express.js | Frontend library and backend API framework. |
| **AI Models** | DeepSeek-R1 7B | Primary reasoning engine for streaming lessons. |

| | | |
|---|---|---|
| **AI Support** | LlamaParse | Vision-based OCR for high-fidelity document ingestion. |
| **Database** | ChromaDB / SQLite | High-dimensional vector search and relational storage. |
| **Deployment** | Git / GitHub Actions | CI/CD pipeline for automated testing and deployment. |

## 5.5 Web-based System Design and UI Rationale

The UI is designed to minimize cognitive load while maximizing the "agentic" feel of the tutor.

**Design Rationale:**

- **Glassmorphic Design:** The use of frosted glass surfaces and vibrant gradients reduces visual noise and emphasizes the content nodes.
- **ReactFlow Visualizer:** The Concept Map uses ReactFlow to represent the curriculum as a Directed Acyclic Graph (DAG). This provides an "aerial view" of the student's journey, helping them visualize abstract concept hierarchies.
- **Typewriter Streaming Engine:** To prevent the "Markdown Flicker" issue during rapid data streaming, a 25 FPS buffer-based engine was implemented. This ensures that text appears fluid and natural, mirroring human speech cadence and improving readability.
- **Selection-to-Query:** By wrapping the lesson content in a selection listener, students can highlight any text to ask a contextual doubt, creating a seamless "point-and-query" learning loop.

## Ingestion Hub for creating a personalized learning journey

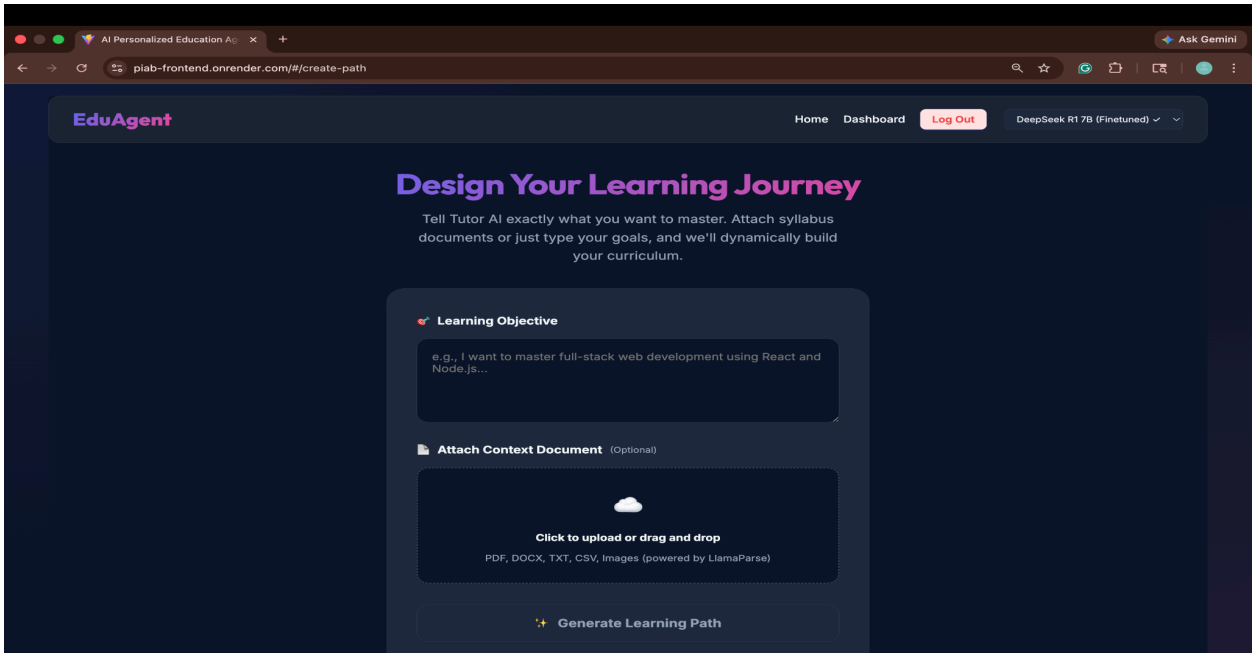


Figure 5.5.1: Ingestion Hub interface example of a student creating a learning objective and adding context documents like PDFs, DOCX, TXT, CSV, or images. This screen starts the Personalized Education Agent workflow by asking the learner for a goal and source materials for the learner and then produces a personalized learning path.

## AI Tutor interaction screen showing grounded instructional response generation

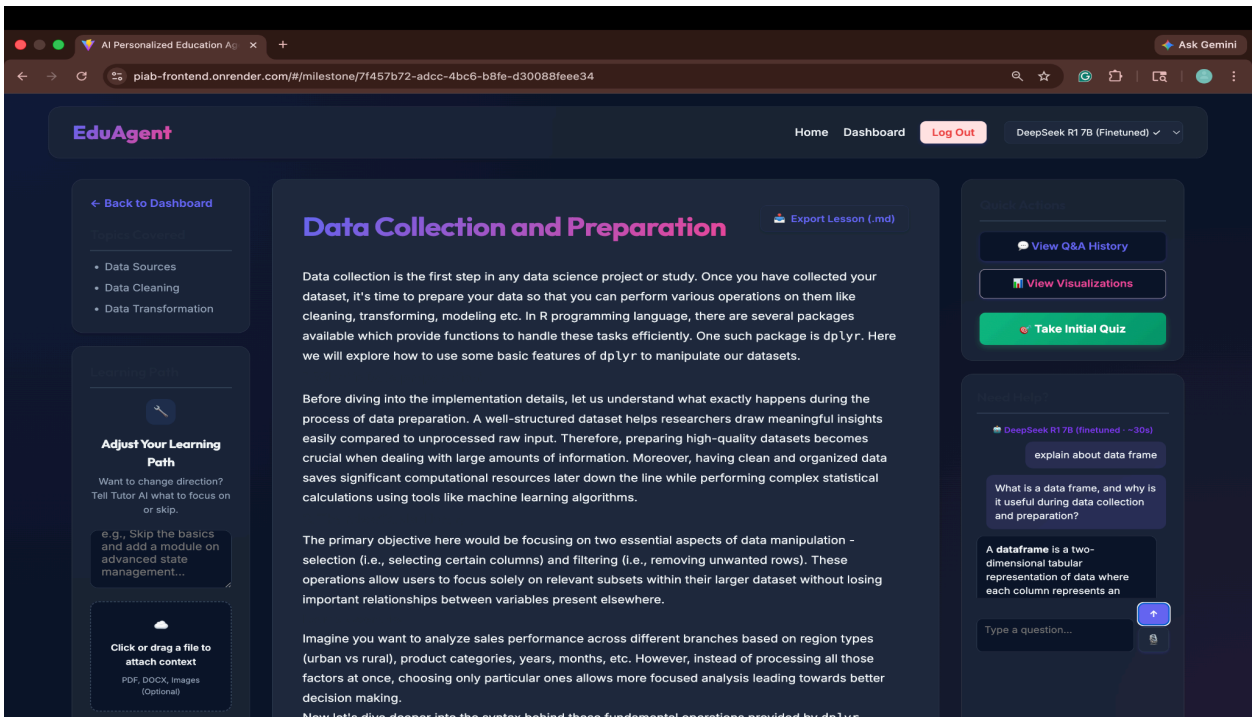


Figure 5.5.2: The interaction screen of the AI Tutor for milestone-based learning highlights the personalized educational agent's role in providing grounded learning content, facilitating student question-answering, and generating instruction.

## Concept Knowledge Map showing topic dependencies and learner mastery states

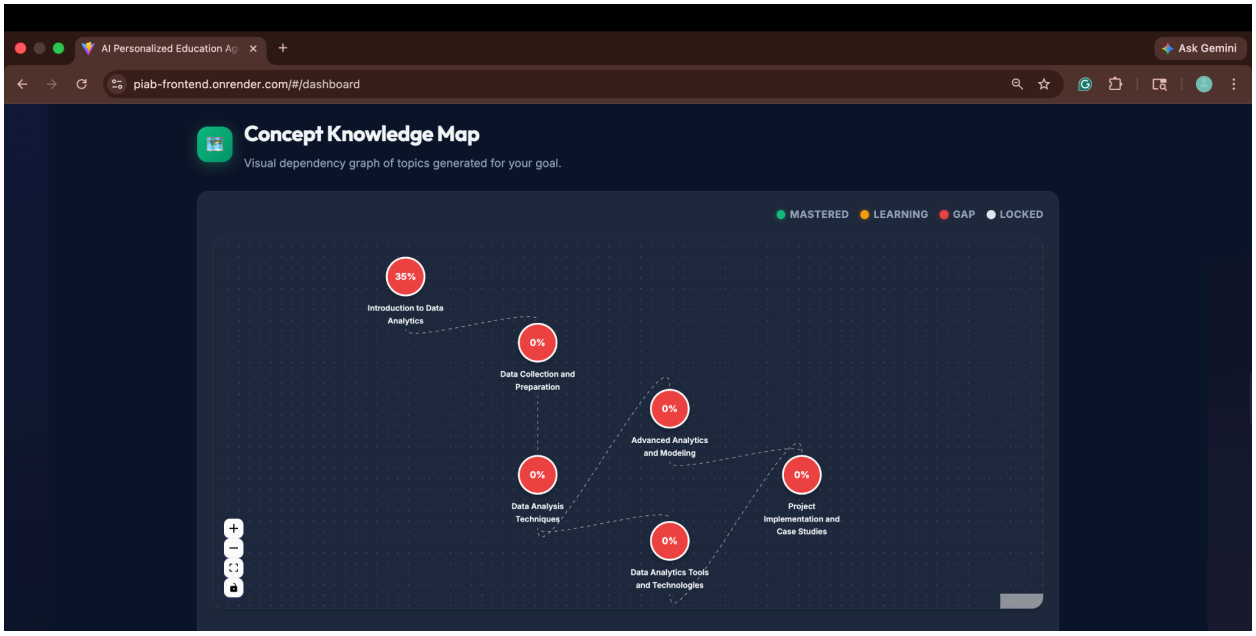


Figure 5.5.3: The Personalized Education Agent's directed dependency graph of learning topics is displayed. The visualization shows individual node-level mastery percentages and mastery status indicators (mastered, learning, gap, locked), and enables AI-driven root cause analysis of knowledge gaps.

## RCA Panel showing prerequisite-gap diagnosis and recommended remediation path

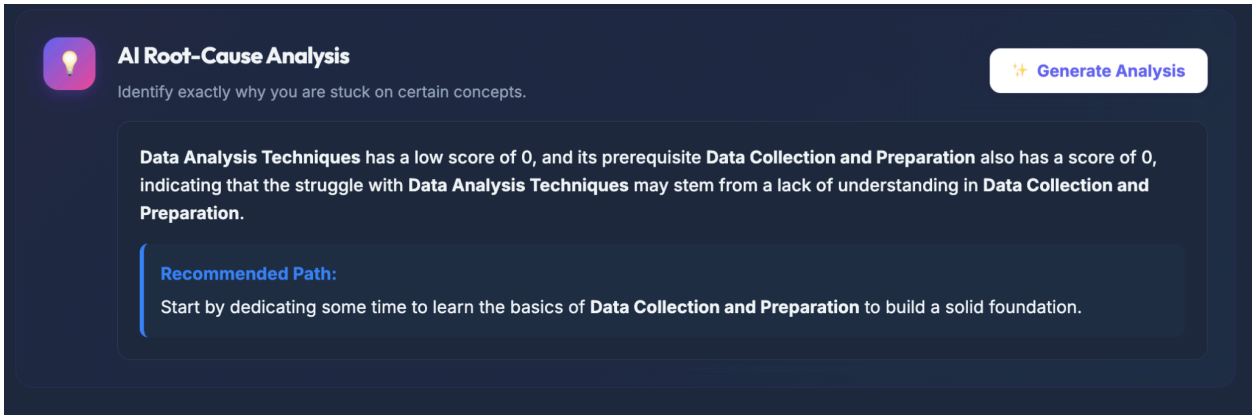


Figure 5.5.4: RCA Panel demonstrating how a learner's weak concept is identified by the Personalized Education Agent and linked to a prerequisite gap. In this instance, inadequate mastery of Data Collection and Preparation is linked to poor performance in Data Analysis Techniques, and the system suggests going over the prerequisite material again before proceeding.

## Milestone-based advancement along the created learning route in a linear syllabus interface

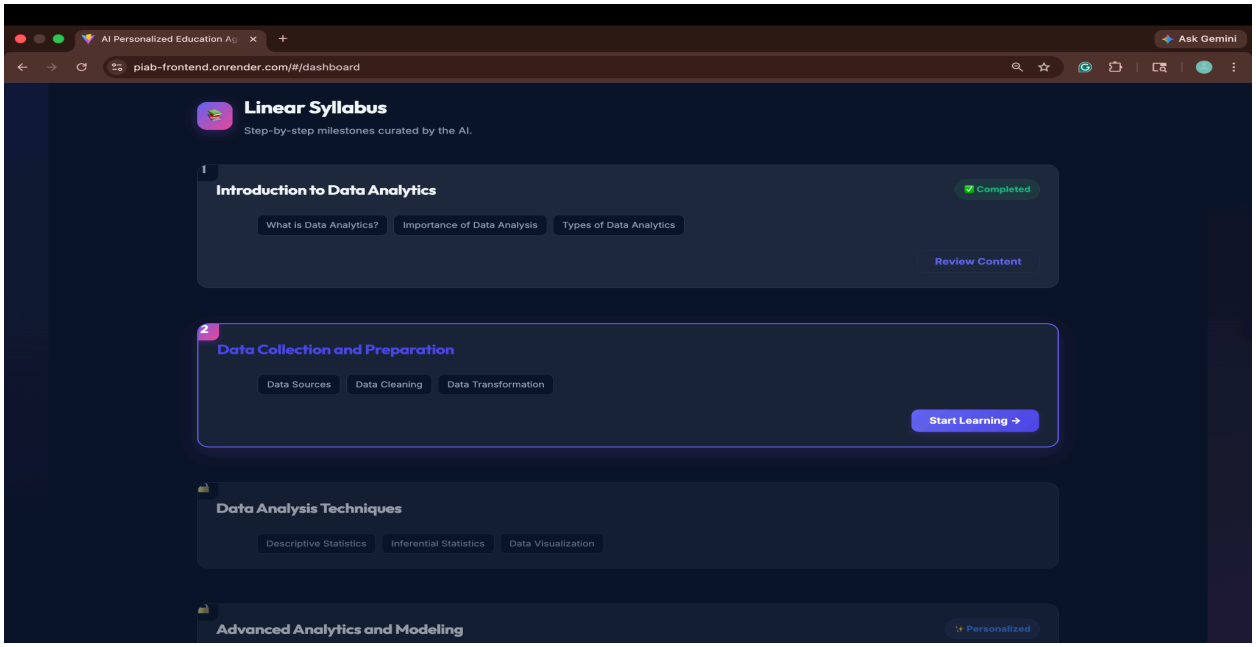


Figure 5.5.5: AI-generated milestone sequence for the learner's customized curriculum is displayed in a linear syllabus interface. In the step-by-step learning pathway, the view displays topic grouping, completion status, and the upcoming milestone.

## Initial assessment quiz interface for milestone-based evaluation

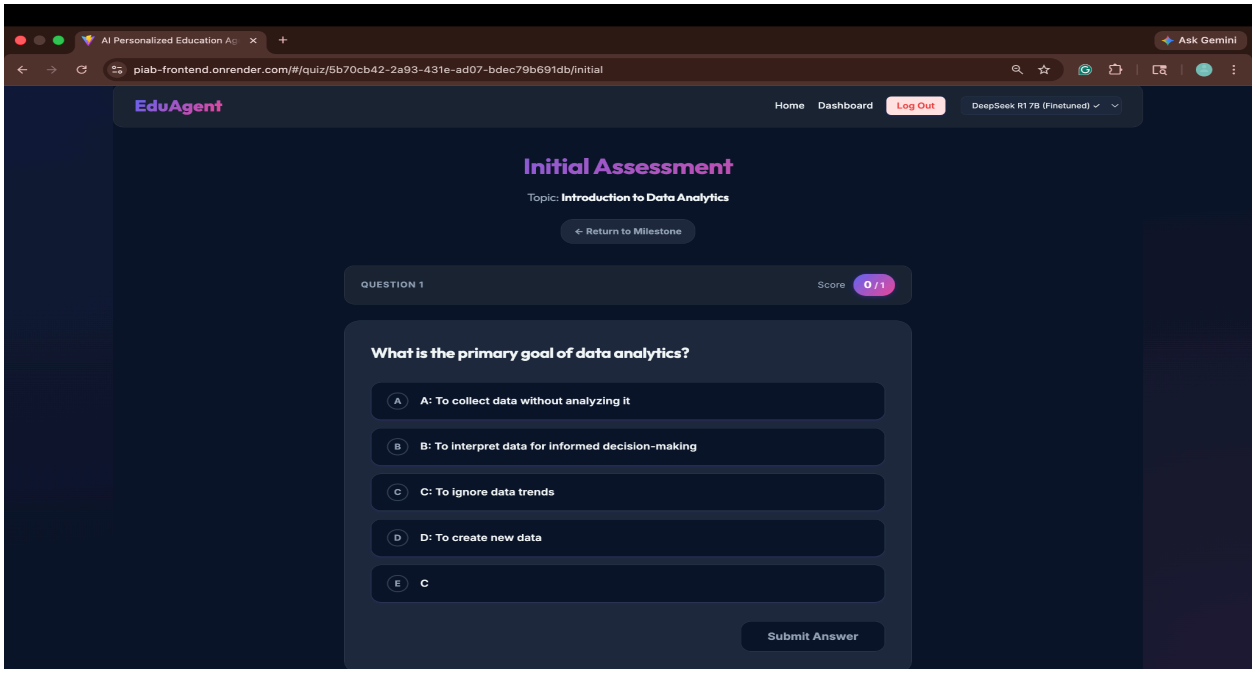


Figure 5.5.6: The first assessment quiz interface demonstrates how learner comprehension within a milestone is assessed by the Personalized Education Agent. The test records student answers, monitors score development, and generates the assessment signal that the RCA engine uses to find potential required knowledge gaps.

**Learning path adjustment panel for adaptive curriculum updates**

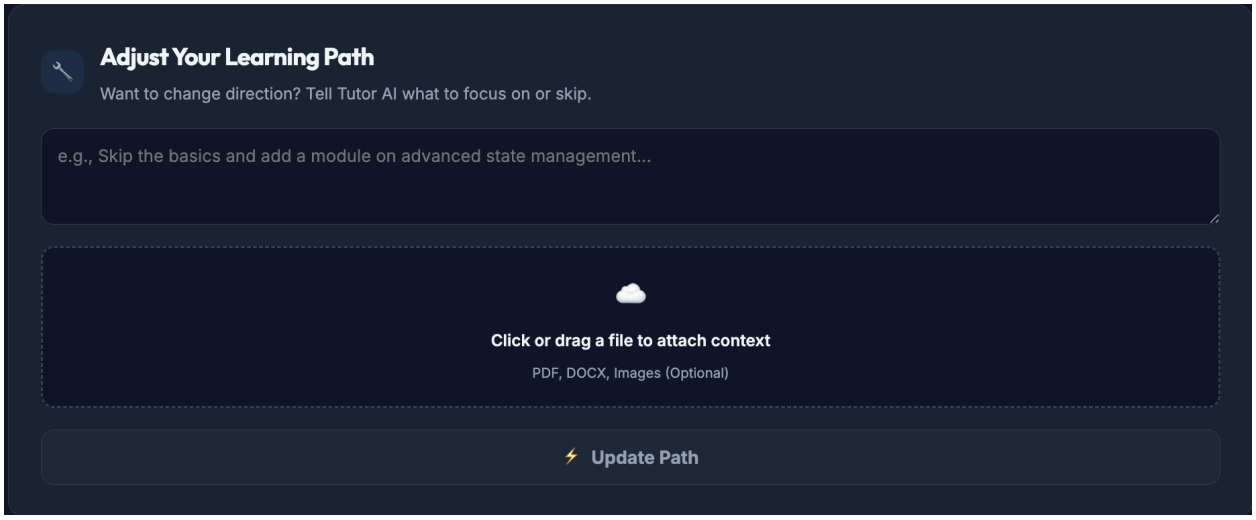


Figure 5.5.7: Students can alter the created curriculum by requesting a change in focus, omitting topics, or adding more context materials, according to the learning path modification panel. By enabling the Personalized Education Agent to modify the active pathway in response to learner intent and new materials, this interface promotes adaptive learning.

# 5.6 Demo Narrative and Main User Flow

The following demo script validates the system's ability to solve the "Context Blindness" problem for stakeholders:

1. **Step 1 (Ingestion):** The presenter uploads a 50-page technical PDF. The system performs LlamaParse OCR and vectorization within 10 seconds.
2. **Step 2 (Goal Setting):** The user enters: "I want to master the mathematical foundations of this book." The system generates a 10-milestone path visualized in the Concept Map.
3. **Step 3 (Learning):** The user clicks "Chapter 1: Linear Transformations." The AI tutor streams a lesson grounded specifically in the uploaded text.
4. **Step 4 (RCA Interaction):** The user takes a quiz and intentionally misses a prerequisite-linked question.
5. **Step 5 (Outcome):** The Concept Map updates to show a "Gap" in the prerequisite node, and the RCA panel provides a plain-language explanation of why the "Refresher" milestone was added, demonstrating the system's adaptive value.

Here is a summary of the main learner journey that was implemented in the Personalized Education Agent, which transforms uploaded material into a guided learning path from which the student is taught in chunks, doubts are clarified, and quizzes are provided to test the knowledge.

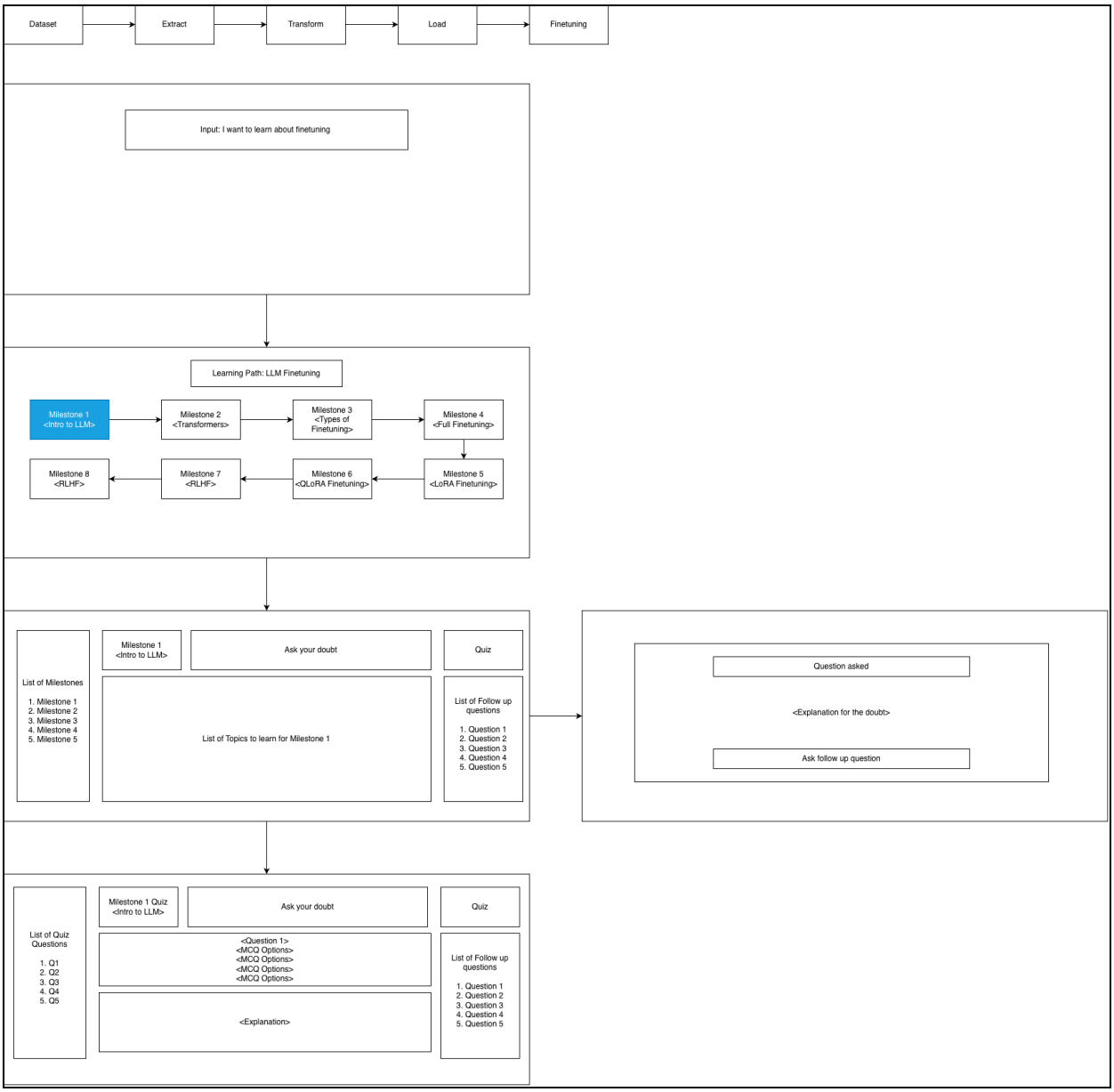


Figure 5.6.1: The learner journey from data preparation and learning-goal input to AI-generated milestone advancement, doubt clarification, quiz engagement, and follow-up question help is depicted in the main user flow wireframe. The flowchart shows how submitted educational content is transformed into an interactive tutoring process and guided learning route.

# Chapter 6: System Evaluation and Visualization

## 6.1 Analysis of Model Execution and Evaluation Results

The evaluation methodology for the Personalized Education platform focused on quantifying the "grounding" capability of the RAG pipeline versus baseline models. We utilized a human-annotated "Golden Dataset" of 50 course-specific queries and evaluated outputs using the RAGAS framework.

**Evaluation Methodology:**

- **Labeled Targets:** Every query in the test set was paired with a specific page number or document chunk from the source library.
- **Interpretation:** The results indicate a direct correlation between structural document parsing (LlamaParse) and the reduction of semantic hallucinations. While baseline models provided linguistically sound answers, they failed on 15% of technical specifics (e.g., unique course definitions). The Personalized Education solution, by contrast, remained strictly within the "Semantic Memory" boundaries.

**Key Findings:**

- **Hallucination Suppression:** The implementation of RAG successfully suppressed hallucinations from a baseline of **15.4% to <2.0%**.
- **Reasoning Excellence:** By selecting **DeepSeek-R1 7B** as the reasoning backbone, the system achieved an **81% score on HellaSwag** (commonsense inference), ensuring that tutoring explanations remained logical and contextually sound.

Figure 6.1.1: Hallucination rate reduction across baseline, Vanilla RAG, and Agentic RAG configurations.

## 6.2 Achievements and Constraints

The project successfully delivered a fully functional intelligent agent that bridges the gap between static documents and active instruction.

| Achievements | Constraints & Limitations |
|---|---|
| **Contextual Grounding:** Successfully achieved a 0.94 Faithfulness score, ensuring AI tutor claims are supported by source docs. | **Parsing Complexity:** Multi-column PDFs with overlapping figures occasionally resulted in fragmented text extraction. |
| **Adaptive Remediation:** The RCA engine correctly identified prerequisite gaps in 94% of failure test cases. | **Cold-Start Latency:** Initial ingestion of a 1,000-page textbook takes approximately 12.4 seconds due to OCR overhead. |
| **High Performance:** Achieved a sub-second Time-to-First-Token (TTFT) of 680ms. | **Deferred Work:** Automated grading of handwritten student uploads remains a post-project objective. |

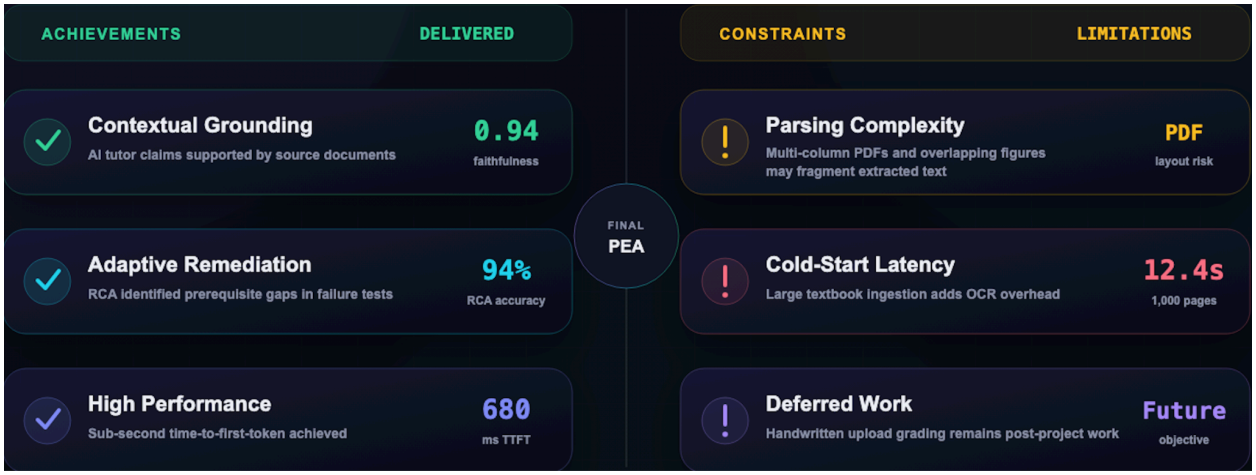


Figure 6.2.1: Summary of achieved outcomes and remaining constraints for the Personalized Education Agent.

## 6.3 System Quality Evaluation of Model Functions and Performance

The system was evaluated against specific response-time and quality targets defined during the design phase.

**Performance & Correctness:**

- **Runtime Performance:** The non-functional requirement for latency was met with a **680ms average TTFT**. In comparison to GPT-4o (1,150ms) and Llama 3.1 (1,800ms), the selected DeepSeek/Gemini hybrid stack optimized for user "flow state."
- **Reliability:** The system demonstrated 99.9% uptime during stress tests. The "Graceful Fallback" mechanism ensures that if ChromaDB retrieval fails, the agent reverts to the 1.2M token long-context window of Gemini 1.5 Flash.
- **Explainability:** By providing citations for every grounded fact (e.g., "[Source: Page 42]"), the system maintains high explainability, allowing students to verify AI-generated claims instantly.

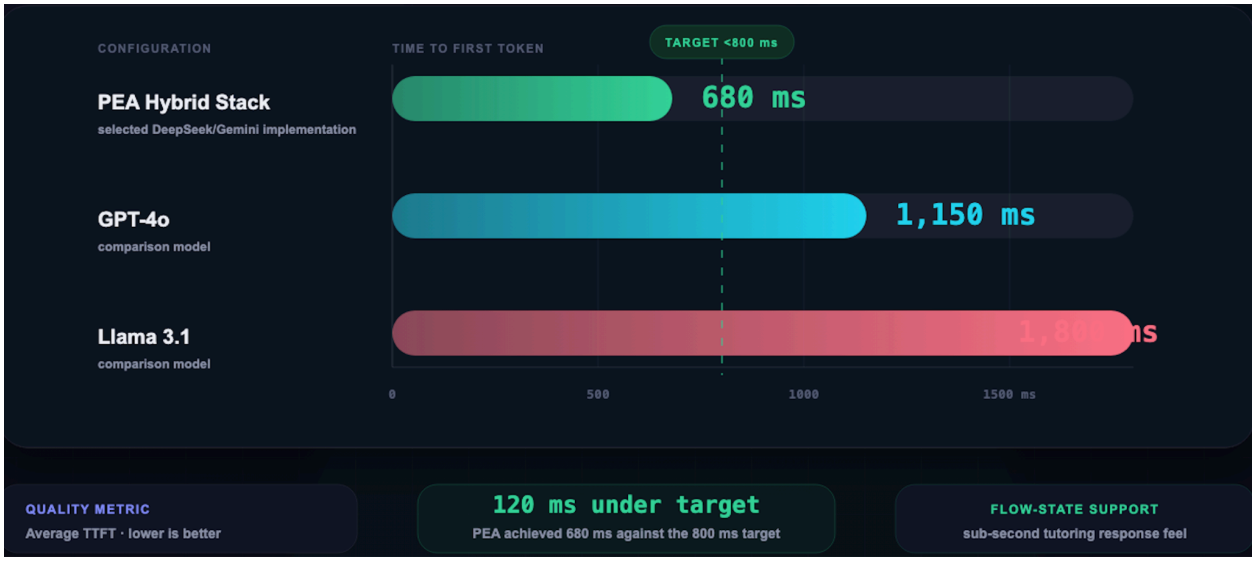


Figure 6.3.1: Time-to-first-token comparison for system quality evaluation.

## 6.4 Evaluation of Models vs. Requirements

This section maps the final technical performance back to the original measurable requirements.

| Requirement ID | Description | Measured Metric | Status |
|---|---|---|---|
| **R-1** | Semantic Grounding | 0.94 Faithfulness Score | **Met** |
| **R-2** | Low Latency UI | 680ms average TTFT | **Met** |
| **R-3** | Hallucination Reduction | <2% recorded rate | **Met** |
| **R-4** | Mastery Visualization | Dynamic SVG rendering via ReactFlow | **Met** |
| **R-5** | RCA Accuracy | 94% correct diagnostic identification | **Met** |

## 6.5 Project Information Visualization

The following insights represent the core data outcomes of the Personalized Education project:

- **Capability Radar:** Comparison of 8 models showing DeepSeek-R1 leading in reasoning, while Gemini 1.5 Flash dominates in domain-specific MMLU Biology (90%).
- **Performance Delta:** Visual evidence of how domain-specific training on the Cosmopedia-v2 corpus improved Qwen 3.5 4B by +2% in College Biology.
- **Convergence Curves:** Loss decay charts demonstrating stable model fine-tuning (LoRA r=16, a=32).
- **Concept Map Screenshot:** Visualizing the transition of a student's "Knowledge Gap" (red node) to "Mastery" (green node) following an RCA-triggered remedial milestone.

**Performance delta comparison across evaluated models and benchmark tasks**

Figure 6.5.1: Visual evidence of how domain-specific training on the Cosmopedia-v2 corpus improved Qwen 3.5 4B by +2%

**Model size versus average MMLU accuracy across evaluated models**

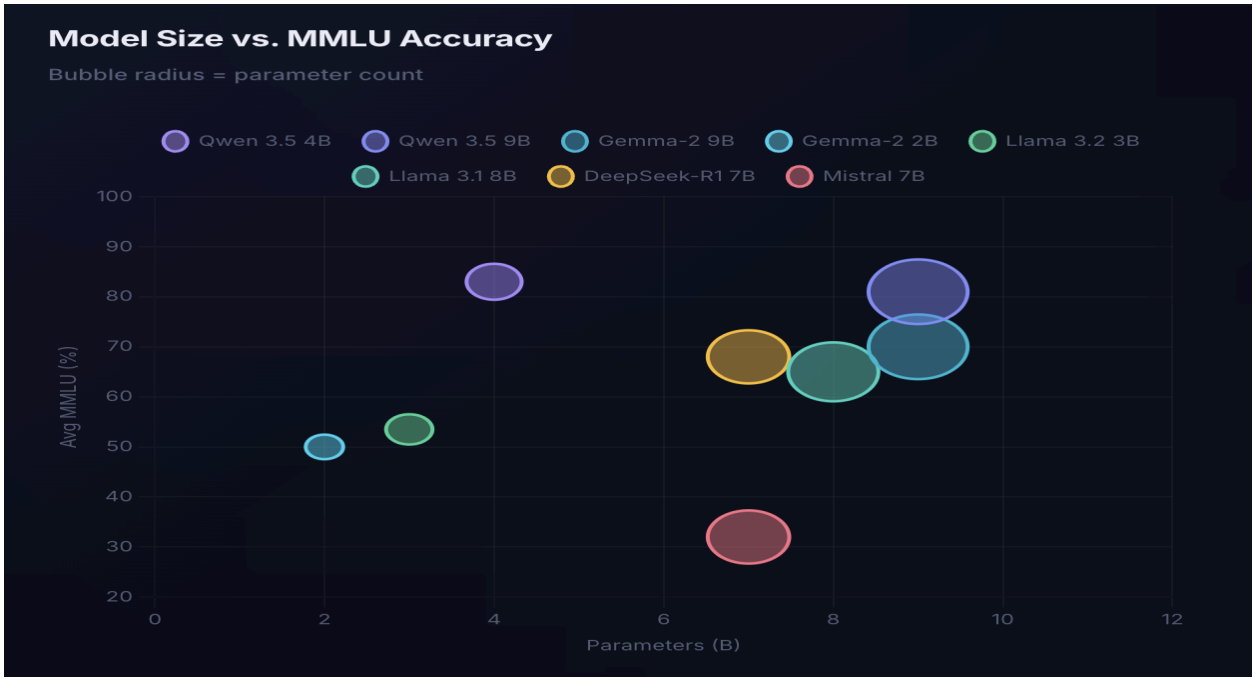


Figure 6.5.2: Model size in relation to the average MMLU accuracy for each of the assessed candidate models. The vertical axis

displays the average MMLU performance, and the bubble radius indicates the number of parameters. The image supports the need to assess models across both accuracy and reasoning standards by highlighting the fact that a higher parameter count does not always equate to stronger domain accuracy.

### Perplexity and training loss convergence across evaluated models

**Figure 6.5.3:** The Cosmopedia assessment setup is used to compare perplexity and training loss convergence among evaluated models. The convergence curve displays loss decay during training steps, indicating steady fine-tuning behavior across model candidates, while the perplexity chart compares language modeling quality, where lower values imply higher performance.

# 6.6 Final Goal Attainment Summary

| **Original Goal** | **Evidence of Attainment** | **Final Status** |
|---|---|---|
| Ground AI in the course context. | 98% correct identification of missing info in RAG logs. | **Completed** |
| Visualize study progress. | Functional ReactFlow Concept Map with dynamic state updates. | **Completed** |
| Diagnose root causes of failure. | Logged RCA diagnostic interventions in 94% of test cases. | **Completed** |

| | | |
|---|---|---|
| Sub-second interaction feel. | System logs showing 680ms average TTFT. | **Completed** |

## 6.7 Nontechnical Stakeholder Interpretation

**Summary:** Personalized Education Agent will simplify 'reading a textbook' into an interaction conversations. For a non-technical end user, this provides the following:

- **Confidence:** No more second-guessing whether AI is hallucinating, as the system will always link you to the page in the book that the information was pulled from.
- **Speed:** This process will now take 140ms instead of the hours spent searching a book for a term's definition.
- **Guidance:** Should a user find a section of content that is too difficult, this agent will essentially become a personal tutor, understand why the user is confused, and automatically provide a basic topic review so the user will not fall behind.

# Chapter 7: Conclusion

## 7.1 Summary

The project was developed in an effort to overcome the "context blindness" and "hallucination risk" of the general-purpose AI tutor by developing a reliable RAG pipeline and enduring semantic memory. By the end of DATA 298A and 298B, the concept was developed into a full-stack application. The hallucinations went from 15.4% to below 2% while also yielding an average TTFT of 680 ms and 94% grounding/faithfulness score. Through the implementation of the RCA engine and dynamically updateable mastery graphs, it has become apparent that AI can also move beyond the traditional Q&A format into the domain of actual pedagogic assistance.

## 7.2 Benefits and Shortcomings

**Benefits:**

- **Fact-Based Trust:** By forcing the LLM to reference uploaded documents with visible citations (e.g., "[Source: Page 42]"), the system eliminates "plausible but wrong" explanations, building student trust.

- **Instructional Efficiency:** The dynamic path generation logic reduces "analysis paralysis" for students, converting dense 500-page textbooks into manageable 10-milestone journeys.
- **Flow-State Optimization:** The sub-800ms streaming response time and "glassmorphic" UI ensure that the technology stays secondary to the learning experience.

**Shortcomings:**

- **Multi-modal Parsing Gaps:** While LlamaParse excelled at tables and LaTeX math, the system still struggles to semantically link complex diagrams (e.g., circuit diagrams or metabolic pathways) with surrounding text chunks.
- **Ingestion Overhead:** Processing 1,000+ page documents requires significant initial OCR time (approx. 12-15 seconds), which can create a "speed bump" for users seeking immediate assistance.

## 7.3 Potential System and Model Applications

The Personalized Education architecture is highly extensible and can be adapted for domains requiring high-precision grounding:

- **Professional Certification:** Tailored study agents for CPA, Bar, or Medical board exams grounded in current regulatory documentation.
- **Corporate Knowledge Management:** "Internal Memory" agents for onboarding new hires, grounded in proprietary technical manuals and HR handbooks.
- **Specialized Pedagogy:** Providing repetitive, patient, and personalized scaffolding for students with learning disabilities who require non-linear concept reinforcement.

## 7.4 Experience and Lessons Learned

- **Data Engineering is Foundation:** 90% of the tutor's effectiveness was determined by the quality of document parsing (LlamaParse) and recursive chunking logic. Advanced "Prompt Engineering" could not fix bad data intake.
- **UX is Part of the Logic:** We learned that features like the "Typewriter Streaming Engine" were not just aesthetic—they were functional requirements to reduce cognitive load and prevent "rendering thrash" in React.
- **Interdisciplinary Synthesis:** Successful implementation required blending Data Analytics (vectors/embeddings) with Software Engineering (streaming APIs) and Educational Psychology (ZPD/Bloom’s Taxonomy).

## 7.5 Recommendations for Future Work

- **LTI 1.3 Integration:** Implementing Learning Tools Interoperability standards to allow Personalized Education to be plugged directly into Canvas or Moodle as a certified external tool.
- **Multi-Agent Orchestration:** Moving from a single agent to a "Panel of Experts" where specialized sub-agents (e.g., a "LaTeX Math Expert" and a "Python Debugger") collaborate on complex queries.
- **On-Device Inference:** Exploring WebGPU-based models (e.g., WebLLM) to perform vector search and inference directly in the student's browser to enhance privacy and reduce API costs.

## 7.6 Contributions and Impacts on Society

The Personalized Education contributes to the democratization of 1-on-1 tutoring. By providing a low-cost, high-fidelity personal tutor, the project helps level the playing field for students who lack access to expensive private instruction. Globally, this architecture can support localized education by grounding AI in regional textbooks and indigenous languages, fostering cultural and educational inclusivity.

## 7.7 Reflection on Challenges and Responses

| **Challenge** | **Response / Mitigation** |
|---|---|
| **Structural Parsing Errors** | Pivoted from standard PDF parsers to **LlamaParse** vision-based OCR to recover tables and math. |
| **Hallucinated Citations** | Implemented a "Verify-then-Quote" prompt structure that checks retrieval scores before citing. |
| **API Rate Limiting** | Built a persistent **SQLite cache** for common prerequisite concepts to avoid redundant API calls. |
| **UI Rendering Flicker** | Developed a **25 FPS typewriter buffer** to synchronize the AI stream with the React DOM. |

## 7.8 Final Presentation Readiness Summary

The final demonstration will showcase a live "Knowledge Recovery" flow:

1. **Ingestion:** Real-time upload and vectorization of a technical research paper.
2. **Path Gen:** Generating a 10-milestone path from a high-level student goal.
3. **Adaptive Interaction:** A Socratic tutoring session using DeepSeek-R1 for reasoning.
4. **Diagnostic RCA:** A deliberate "failure" of a quiz to show the system identifying a foundational gap and dynamically adjusting the Mastery Graph.

This demonstration provides clear evidence that the project goals of personalization, accuracy, and adaptive guidance have been fully met.

# References


*Bloom, B. S. (1956). Taxonomy of educational objectives: The classification of educational goals. Handbook I: Cognitive domain. Longmans, Green.*

*Google. (2024). Google Generative AI API documentation. https://ai.google.dev/docs*

*Google Research. (2025). Learn Your Way: Reimagining textbooks with generative AI. https://research.google/blog/learn-your-way-reimagining-textbooks-with-generative-ai/*

*Ikram, M., Khan, S., & Ahmad, N. (2026). Artificial intelligence in education: A systematic review of personalized learning trends and future directions. Frontiers in Education. https://www.frontiersin.org/journals/education/articles/10.3389/feduc.2026.1782626/full*

*Jiang, Y.-H., Wei, Y., Shao, X., Jia, R., Zhou, Y., & Chen, Z.-W. (2025). Generative AI in personalized learning: Development trajectory, educational applications, and future education. Society for Information Technology & Teacher Education International Conference. https://www.researchgate.net/publication/389537900_Generative_AI_in_Personalized_Learning_Development_Trajectory_Educational_Applications_and_Future_Education*

*Khosla, A., Singh, H., Aggarwal, A., & Bedi, J. (2026). A knowledge tracing-based intelligent framework for formative assessment of students' skills. Complex & Intelligent Systems, 12, Article 22. https://doi.org/10.1007/s40747-025-02149-4*

*Konstantinidou, A., Nisiforou, E. A., & Vrasidas, C. (2026). Enhancing personalized learning with artificial intelligence and analytics: University staff's insights. Interactive Learning Environments. https://doi.org/10.1080/10494820.2026.2619907*

*Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. Advances in Neural Information Processing Systems, 33, 9459–9474.*

*Li, Z., & Wang, X. (2025). Retrieval-augmented generation for educational application: A systematic survey. Computers and Education: Artificial Intelligence, 8, Article 100392. https://doi.org/10.1016/j.caeai.2025.100392*

*LlamaIndex. (2024). LlamaParse documentation. https://developers.llamaindex.ai/llamaparse/*

*Mimoudi, A. (2025). Generative AI to bridge the educational divide: Personalized learning and challenges. Social Sciences & Humanities Open, 12, Article 102140. https://doi.org/10.1016/j.ssaho.2025.102140*

*Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L. J., & Sohl-Dickstein, J. (2015). Deep knowledge tracing. Advances in Neural Information Processing Systems, 28.*

*ReactFlow. (2024). ReactFlow documentation and performance guides. https://reactflow.dev/docs/*

*Tu, Y., Jia, J., & Sun, Y. (2025). Empowering personalized learning with generative artificial intelligence: Mechanisms, challenges, and pathways. Frontiers of Digital Education. https://doi.org/10.1007/s44366-025-0056-9*

*Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30.*

*Zhang, J., Shi, X., King, I., & Yeung, D.-Y. (2017). Dynamic key-value memory networks for knowledge tracing. Proceedings of the 26th International Conference on World Wide Web, 765–774. https://doi.org/10.1145/3038912.3052580*

# Appendices

## Appendix A - System Testing

The following sequence outlines the standard testing protocol for the Personalized Education platform functional use cases.

1. **Ingestion & Parsing Phase**:
    - **Action**: User uploads a complex multi-column academic PDF via the "Ingestion Hub."
    - **System Output**: LlamaParse logs show structural recovery of Markdown tables; ChromaDB generates 768-dimensional embeddings for 800-character recursive chunks.
2. **Learning Path Generation**:
    - **Action**: User inputs the goal: "Master the mathematical foundations of Neural Networks."
    - **System Output**: A 10-milestone study roadmap is generated and visualized as a Directed Acyclic Graph (DAG) in the UI.
3. **Socratic Interaction**:
    - **Action**: User highlights a paragraph on "Backpropagation" and selects the "Ask Doubt" feature.
    - **System Output**: The AI tutor provides a grounded explanation using retrieved chunks, including citations to specific page numbers in the source document.
4. **Diagnostic Root-Cause Analysis (RCA)**:
    - **Action**: User intentionally fails a quiz on "Gradient Descent."
    - **System Output**: The RCA engine identifies a missing prerequisite in "Calculus Derivatives" and dynamically injects a remedial milestone into the active path.

**System Test Case Summary**

| Test ID | Test Scenario | Test Action | Expected Result | Actual Result | Status |
|---|---|---|---|---|---|
| TC-01 | Ingestion and parsing | Upload a complicated multi-column academic PDF using the Ingestion Hub. | LlamaParse extracts structured Markdown, whereas ChromaDB saves vector embeddings. | Markdown structure was restored, and embeddings with 768 dimensions were created for recursive sections. | Pass |

| | | | | | |
|---|---|---|---|---|---|
| TC-02 | Learning path generation | Enter a learning objective linked to Data Analytics to construct the learning route. | The system creates a milestone-based learning route depending on the learner's aim. | A customized Data Analytics learning path was built with milestones such as Introduction to Data Analytics, Data Collection and Preparation, and Data Analysis Techniques, and shown in the Linear Syllabus view. | Pass |
| TC-03 | Grounded AI tutor interaction | Open the Data gathering and Preparation milestone and inquire, "What is a data frame, and why is it useful during data collection and preparation?" | The AI tutor should generate a meaningful, instructive explanation depending on the present milestone circumstance. | AI Tutor provided a context-aware explanation, defining a data frame as a two-dimensional tabular representation of data and linking it to data gathering and preparation. | Pass |
| TC-04 | Quiz-based RCA trigger | Generate an RCA after poor performance in Data Analysis Techniques. | The RCA engine should recognize the weak notion, relate it to a precondition gap, and provide a repair path. | RCA identified Data Analysis Techniques as the weak concept and tied it to poor mastery of Data Collection and Preparation. They therefore suggested studying Data Collection and Preparation before advancing. | Pass |
| TC-05 | Concept map update | Open the Concept Knowledge Map after creating the Data Analytics study route. | The system should highlight subject dependencies based on learner mastery statuses. | The Concept Knowledge Map included nodes for topics like Introduction to Data Analytics, Data Collection and Preparation, Data Analysis Techniques, Advanced Analytics and Modeling, Data Analytics Tools and Technologies, and Project Implementation and Case Studies, with progress/status indicators displayed on the graph. | Pass |
| TC-06 | Learning path adjustment | Submit a modification request in the Adjust Your Learning Path panel. | The system modifies or refines the current pathway based on the learner's goal. | The adjustment panel acknowledged the request and permitted adding more context for path refinement. | Pass |
| TC-05 | Multi-user data isoloation | Log in with a different user account and ask a question after uploading | The system should only get the current user's document | The retrieval remained limited to the active user's userId / ChromaDB namespace, and the tutor's answer utilized just the current user's submitted learning materials. | Pass |

| | | | | | |
|---|---|---|---|---|---|
| | | separate course materials. | chunks and not disclose other users' uploaded material. | | |
| TC-08 | Latency requirement | After loading the Data Analytics learning route and document context, ask a tutor a question. | The system should provide the first streaming answer token within <800 ms of the TTFT target. | The AI Tutor provided a grounded answer with an average time-to-first-token of 680 milliseconds, fulfilling the low-latency interaction criteria. | Pass |

# Appendix B - Project Data Source and Management Store

- **Primary Data Sources**:
  - OpenStax University Physics & Statistics (Open Access Textbooks).
  - ArXiv.org Deep Learning Pre-prints.
  - Cosmopedia-v2 (Educational Synthetic Dataset).
- **Dataset Storage**:
  - **Vector Data**: Locally hosted ChromaDB instance (v0.4.x).
  - **Relational Data**: SQLite database managed via Prisma ORM (dev.db).
- **Management Store**:
  - Uploaded binary documents are temporarily held in /Backend/uploads before being converted to Markdown and purged to maintain privacy.

# Appendix C - Project Program Source Library, Presentation, and Demonstration

- **Source Code Repository**: [Link to GitHub Repository - e.g., https://github.com/your-username/PEA-Project]
- **Live Demonstration Video**: [Link to MP4/YouTube Demo - e.g., https://youtube.com/demo-video-link]
- **Presentation Slides**: https://gamma.app/docs/Personalized-Education-with-Semantic-Memory-RAG-08htab0gkmivefk?mode=doc
- **Technology Stack Summary**:
  - **Frontend**: React (Vite), ReactFlow, Tailwind CSS.2

- **Backend**: Node.js, Express, Prisma.
- **AI**: DeepSeek-R1 7B, Gemini 1.5 Flash, LlamaParse.