Mentorix: An Autonomous Agentic AI System for Personalized Learning

Capstone Project Report
Author: Himanshu Kumar Singh

---

1. Problem Statement

1.1 One-Line Problem Statement

There is no scalable and autonomous learning system that continuously adapts both the pace and pedagogy of mathematics education according to each student's evolving cognitive ability and learning behavior.

1.2 Problem Statement (Concise Version)

Students preparing for secondary-level mathematics examinations exhibit wide variations in learning speed, conceptual understanding, and problem-solving ability. However, most learning environments follow fixed instructional structures and uniform teaching styles that do not account for these individual differences. As a result, students often struggle with either cognitive overload or insufficient academic challenge. There is a need for an autonomous system that can continuously analyze learner performance, maintain long-term learning profiles, and dynamically adjust instructional tone, content complexity, assessment strategies, and revision plans to match each student's evolving capabilities.

1.3 Detailed Problem Statement

1.3.1 Variability in Student Learning Patterns

Secondary school students differ significantly in:

- Prior academic background
- Logical reasoning skills
- Conceptual clarity
- Learning speed
- Memory retention
- Response to instructional methods

In mathematics education, these differences strongly influence how effectively students grasp foundational concepts and apply them in problem-solving contexts.

Uniform instructional approaches are unable to accommodate this high degree of variability.

1.3.2 Limitations of Fixed Instructional Structures

Most structured learning programs follow predefined:

- Chapter sequences
- Teaching styles
- Explanation formats
- Assessment schedules
- Revision cycles

These structures are typically designed for an “average” learner and remain largely unchanged throughout the learning period.

Such rigidity prevents instruction from responding to:

- Sudden improvements
- Learning stagnation
- Conceptual misunderstandings
- Knowledge decay
- Shifts in motivation

1.3.3 Insufficient Adaptation of Pedagogical Strategy

Adaptive learning is often interpreted as adjusting study speed or scheduling frequency. However, effective personalization requires adaptation at a deeper pedagogical level, including:

- Explanation depth
- Language simplicity
- Example density
- Visualization strategies
- Assessment difficulty
- Revision selectivity

Without adapting these elements, personalization remains superficial and fails to address core learning needs.

1.3.4 Lack of Continuous Learner Modeling

Sustained personalization requires accurate and evolving learner representations derived from:

- Diagnostic assessments
- Performance trends
- Behavioral patterns
- Forgetting cycles
- Engagement metrics

Many systems lack mechanisms to maintain both short-term and long-term cognitive memory that can support fine-grained instructional decisions.

1.3.5 Absence of Autonomous Learning Orchestration

Most learning environments rely heavily on manual student interaction for:

- Requesting help
- Adjusting plans
- Initiating assessments
- Triggering revisions

This dependence limits learning continuity and places cognitive and motivational burdens on students.

There is a need for autonomous systems capable of:

- Proactively scheduling tasks
- Monitoring compliance
- Triggering interventions
- Re-planning trajectories
- Providing continuous guidance

without constant user prompting.

1.3.6 Challenges in Explainability and Trust

Personalized educational systems must be transparent and interpretable to ensure learner trust and engagement.

Students and parents require:

- Clear reasoning behind plan changes
- Justification for assessments
- Interpretability of feedback
- Visibility into progress metrics

The lack of explainable personalization reduces system credibility and long-term adoption.

1.3.7 Curriculum Grounding and Academic Alignment

For secondary-level education, learning systems must strictly align with:

- Prescribed curricula
- Standard textbooks
- Examination formats
- Evaluation rubrics

Without robust grounding mechanisms, generated instructional content risks conceptual inconsistency and academic misalignment.

1.3.8 Formal Research Problem Statement

The central research problem addressed in this work is the design and implementation of an autonomous, multi-agent learning system that continuously models learner cognition and dynamically adapts instructional strategies, content complexity, and assessment processes in alignment with curriculum standards and individual learning trajectories.

---

2. Project Details

2.1 Project Description

Mentorix is an autonomous, multi-agent adaptive learning system designed to deliver personalized mathematics education for secondary-level students. The system continuously models individual learner capabilities, behavioral patterns, and cognitive states to dynamically adjust instructional strategies, content complexity, assessment processes, and revision cycles.

Unlike static or semi-adaptive learning platforms, Mentorix emphasizes deep pedagogical personalization by adapting not only learning schedules but also the tone, depth, format, and instructional style of educational content according to each learner's evolving profile.

The system integrates curriculum-grounded retrieval mechanisms, long-term and short-term memory architectures, autonomous orchestration, and explainable feedback pipelines to ensure academic alignment, transparency, and sustained learner engagement.

Initially focused on CBSE Class 10 Mathematics, Mentorix is designed with extensibility to support additional subjects, curricula, and learning contexts.

Key Characteristics

- Autonomous learning orchestration
- Continuous learner profiling
- Pedagogical adaptation
- Curriculum-grounded generation
- Explainable personalization
- Multi-agent coordination
- Scalable system design

2.2 Solution Overview

Mentorix operates through a coordinated multi-agent framework where each agent contributes specialized intelligence toward learner modeling, planning, instruction, evaluation, and explainability. Together, these agents form a closed-loop adaptive learning system.

2.2.1 System Philosophy

Mentorix is built on the principle that effective personalization requires continuous co-evolution between:

- Learner cognition

- Instructional strategy
- Assessment mechanisms
- Learning trajectories

The system treats learning as a dynamic control process rather than a static content delivery pipeline.

2.2.2 Core Architectural Paradigm

Mentorix follows a Custom Agentic Micro-Architecture built on explicit separation of cognition, memory, orchestration, and execution.

Each agent performs domain-specific functions while sharing information through structured memory and knowledge representations.

Key Principles

- Agents are stateful, goal-driven, and autonomous
- Orchestration is code-controlled, not framework-controlled
- Adaptation is content-level + tone-level, not just scheduling
- Memory is long-term, explainable, and learner-centric
- Every decision is traceable (important for research)

2.3 System Architecture

The architecture consists of onboarding and profiling agents, planning and content agents, assessment and analytics agents, a memory management layer, a knowledge grounding layer, and an orchestration engine.

2.3.1 Onboarding and Profiling Agent

Responsibilities

- Conduct diagnostic assessments
- Analyze historical academic records
- Extract baseline cognitive indicators
- Initialize learner models

Outputs

- Initial proficiency vector
- Learning style indicators
- Risk assessment metrics

2.3.2 Learning Planner Agent

Responsibilities

- Generate long-term learning trajectories
- Produce short-term actionable plans
- Allocate chapter-wise timelines
- Balance coverage and mastery

Outputs

- Multi-week adaptive plans
- Weekly task schedules
- Revision strategies

2.3.3 Content Adaptation Agent

Responsibilities

- Retrieve curriculum-aligned material
- Generate personalized explanations
- Adjust linguistic complexity
- Control example density
- Produce multimodal assets

Outputs

- Adaptive lesson modules
- Personalized learning artifacts
- Explainable visualizations

Key Logic
If error_rate ↑ and response_time ↑ → simplify tone + add examples

Key USP
👉 Same chapter ≠ same breakdown for all students

2.3.4 Assessment Generation Agent

Responsibilities

- Create diagnostic tests
- Generate unit evaluations
- Produce mock examinations
- Balance difficulty levels

Outputs

- Personalized assessments
- Performance benchmarks

2.3.5 Evaluation and Analytics Agent

Responsibilities

- Analyze test results
- Detect misconception patterns
- Monitor forgetting curves
- Update learner models

Outputs

- Skill gap maps
- Progress dashboards
- Performance predictions

2.3.6 Compliance and Monitoring Agent

Responsibilities

- Track task completion

- Detect disengagement
- Trigger interventions
- Generate alerts

Outputs

- Engagement reports
- Intervention recommendations

2.3.7 Memory Management Layer

Short-Term Memory

- Weekly activities
- Recent errors
- Temporary learning states

Long-Term Memory

- Concept mastery history
- Cognitive profiles
- Behavioral trends
- Performance trajectories

2.3.8 Knowledge Grounding Layer (RAG)

Responsibilities

- Index NCERT textbooks
- Store curriculum embeddings
- Retrieve verified references
- Prevent hallucination

Technologies

- Embedding models
- FAISS vector store

- Hybrid retrieval

2.3.9 Orchestration and Scheduling Engine

Responsibilities

- Coordinate agent execution
- Schedule periodic tasks
- Manage workflows
- Resolve conflicts

Technologies

- Cron-based schedulers
- Event-driven triggers
- Workflow controllers

2.3.10 End-to-End Workflow Example: Teaching a Chapter

1. Learner Profiling Agent
    - Reads history
    - Updates learner state
2. Planner Agent
    - Breaks chapter adaptively
3. Content Agent
    - Generates base explanation
4. Adaptation Agent
    - Rewrites tone & depth
5. Tutor Interaction
    - Student engages
6. Assessment Agent
    - Tests understanding

7. Memory Agent
    - Stores outcomes
8. Reflection Agent
    - Explains decisions

2.4 Operational Workflow

1. Student onboarding
2. Baseline profiling
3. Plan generation
4. Content personalization
5. Assessment delivery
6. Performance analysis
7. Memory update
8. Plan re-optimization
9. Continuous iteration

This loop operates autonomously throughout the learning lifecycle.

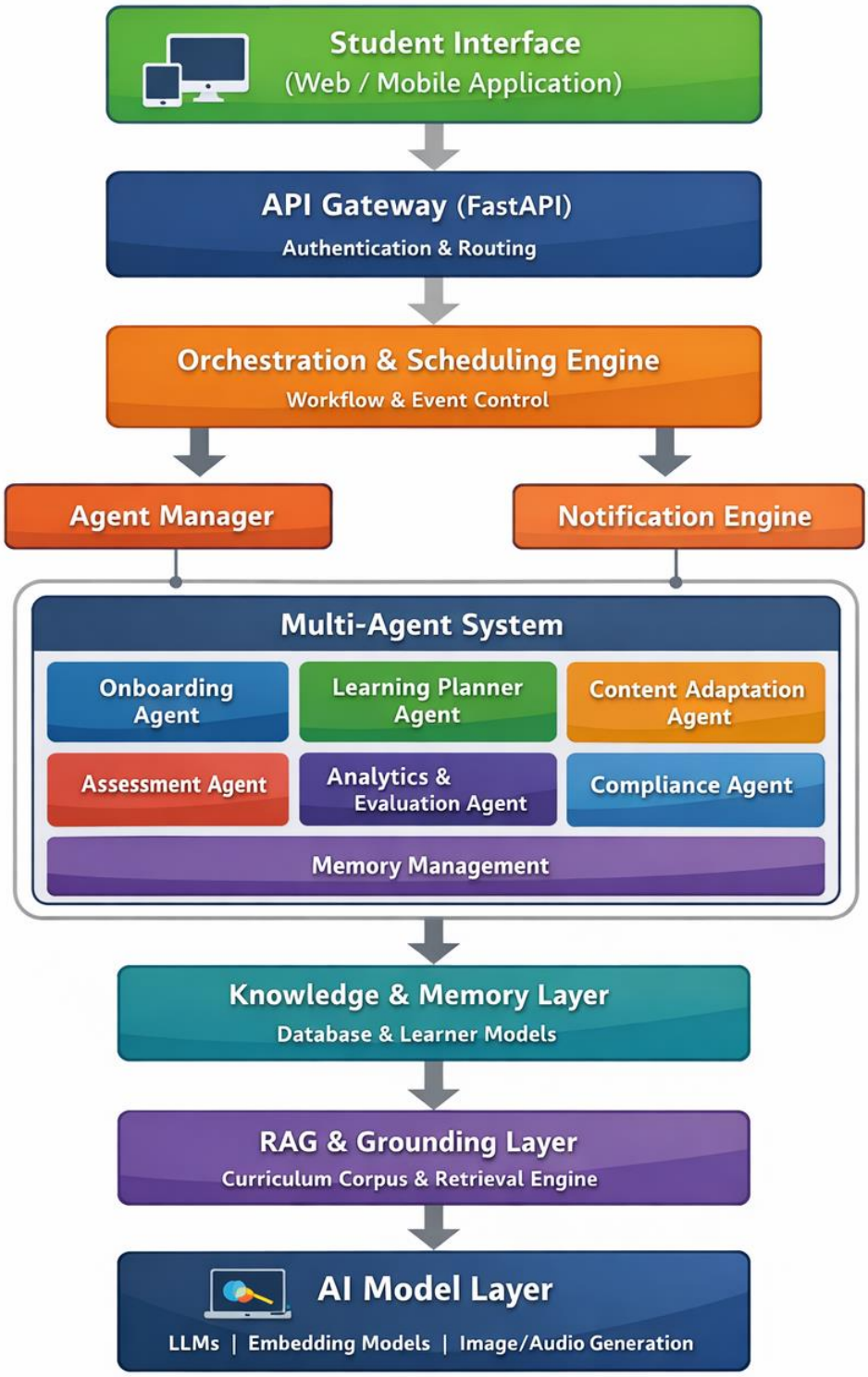
2.5 Explainability Layer

Mentorix incorporates an Explainable AI (XAI) module that provides:

- Plan change justifications
- Assessment reasoning
- Recommendation explanations
- Progress interpretations

This ensures transparency and trust.

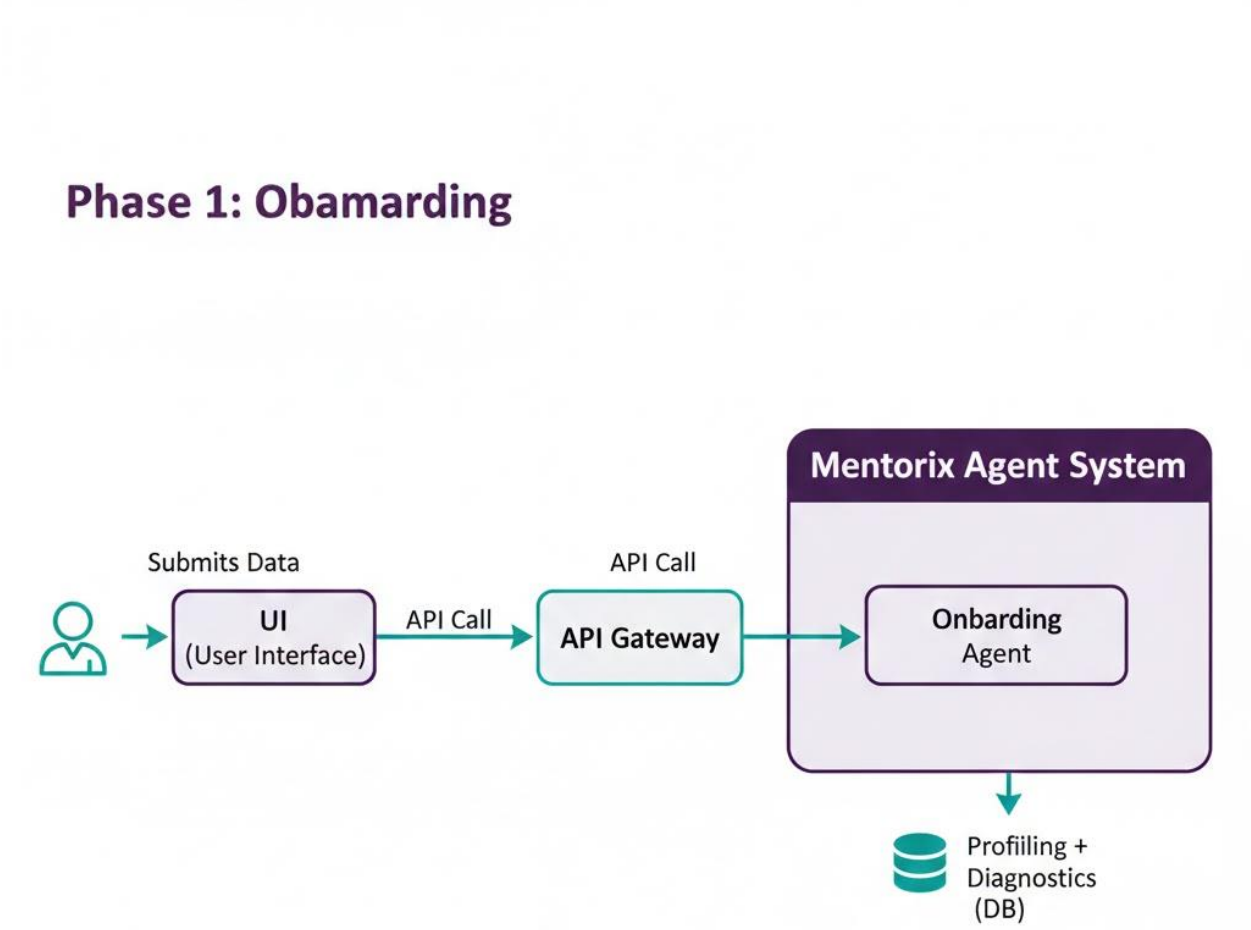
2.6 High-Level Architecture (Block View)


Student Interface
(Web / Mobile Application)
API Gateway (FastAPI)
Authentication & Routing
Orchestration & Scheduling Engine
Workflow & Event Control
Agent Manager
Notification Engine
Multi-Agent System
Onboarding Agent
Learning Planner Agent
Content Adaptation Agent
Assessment Agent
Analytics & Evaluation Agent
Compliance Agent
Memory Management
Knowledge & Memory Layer
Database & Learner Models
RAG & Grounding Layer
Curriculum Corpus & Retrieval Engine
AI Model Layer
LLMs | Embedding Models | Image/Audio Generation

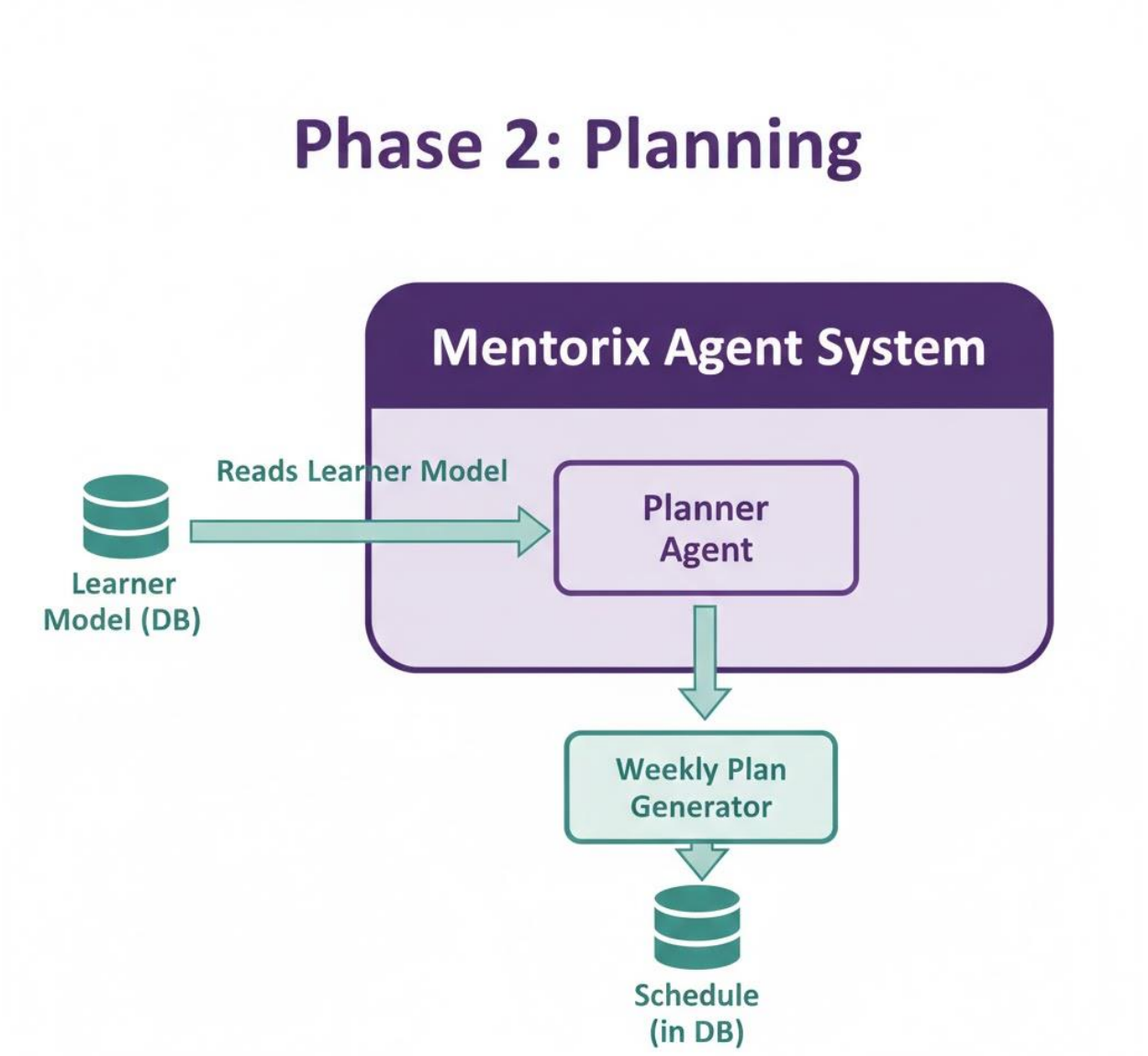
2.7 Detailed Data Flow

The learning process follows an event-driven agent pipeline, ensuring traceable, explainable, and adaptive instruction.

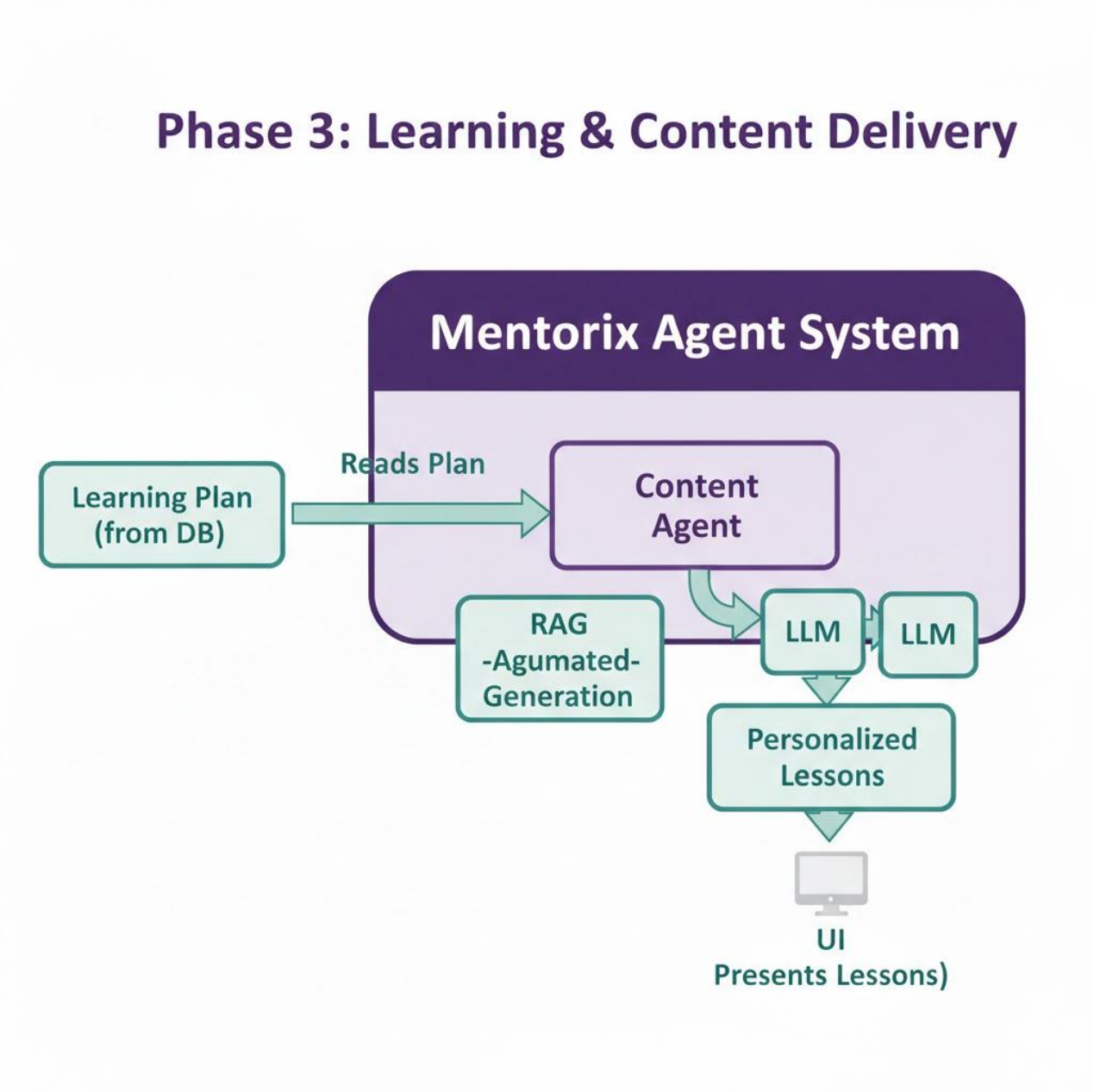
- Phase 1: Onboarding



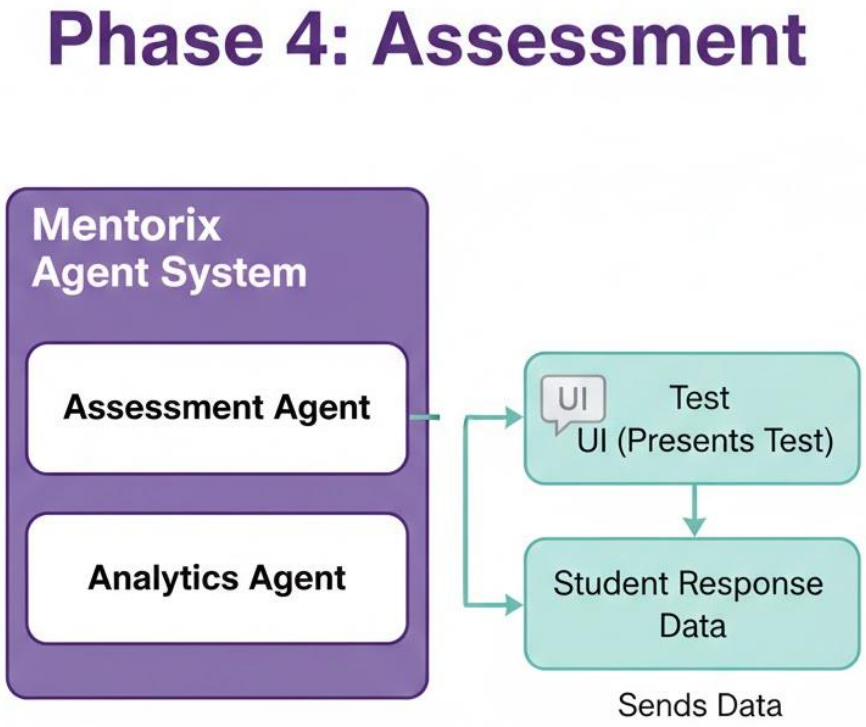
- Phase 2: Planning



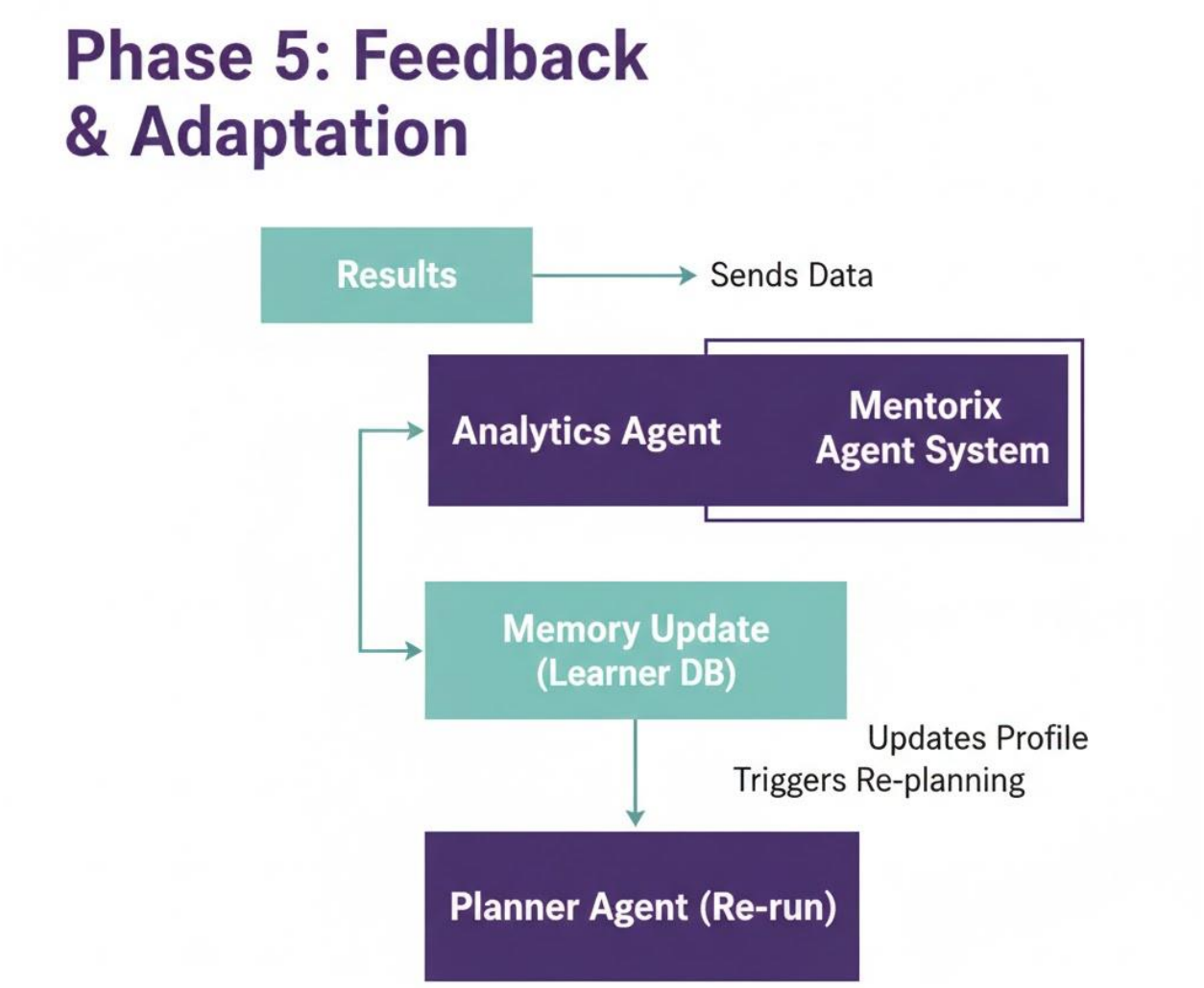
- Phase 3: Learning & Content Delivery



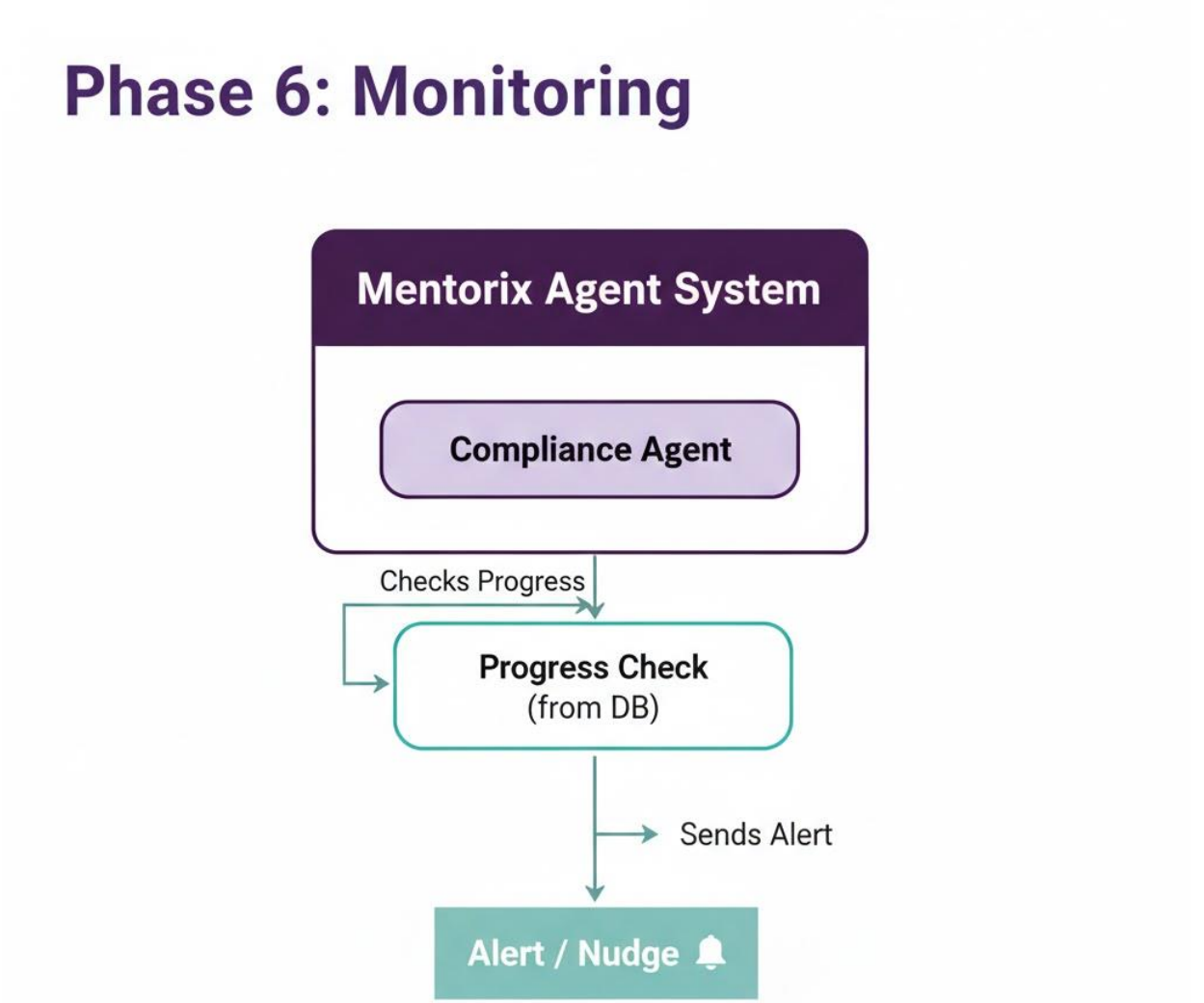
- Phase 4: Assessment



- Phase 5: Feedback & Adaptation



- Phase 6: Monitoring



2.8 Technology Stack

Core Backend

Component	Technology
Language	Python
API	FastAPI

Component	Technology
Async Tasks	Celery / BackgroundTasks
Orchestration	Custom Python State Machine
LLM	OpenAI / Open-source LLM
Embeddings	OpenAI / SentenceTransformers

Memory & Storage

Purpose	Technology
Vector Memory	FAISS / Chroma
Structured Data	PostgreSQL
Session Cache	Redis
Logs & Traces	JSON + Database

Frontend

Component	Technology
UI	React / Next.js
Charts	Recharts
Authentication	JWT

2.9 Business Scope and Societal Impact

Mentorix targets the large and growing student population preparing for secondary-level examinations. The system follows a direct-to-student model with scalable pricing. Socially, Mentorix aims to democratize education by enabling low-cost or offline deployments using local LLMs in underserved regions.

Business Scope & Societal Impact

2.9.1. Business Scope of Mentorix

Mentorix is designed as a **scalable agentic learning infrastructure**, not a single-use tutoring application. The core value lies in its ability to adapt content structure, explanation depth, and instructional tone dynamically for each learner.

Initial Scope

- Target users: Secondary school students (Class 9–10 Mathematics)
- Core offering:
    - Personalized learner profiling
    - Adaptive weekly learning plans
    - Concept-level explanation customization
    - Continuous assessment and feedback

This focused entry point allows controlled validation of learning outcomes while maintaining architectural generality.

Expansion Scope

Once validated, the same architecture can be extended to:

- Other STEM subjects (Physics, Chemistry, Biology)
- Non-STEM domains (History, Economics, Languages)
- Competitive exam preparation
- Skill-based and vocational education

The system's agent-based design ensures **domain independence**, with subject-specific knowledge introduced only at the content layer.

2.9.2. Business Pricing Model

Mentorix follows a **progressive value-based pricing model**, ensuring accessibility while enabling sustainability.

---

1. **Free Tier – Learner Onboarding & Awareness**

Objective: Mass onboarding and learner profiling

Features:

- Student onboarding and background intake

- Diagnostic assessment
- Cognitive and skill profiling
- High-level personalized learning roadmap (e.g., 25-week plan)
- No content delivery or scheduling

Value:

- Helps students understand their current standing
- Serves as an informed decision point for paid upgrades

---

2. **Core Subscription – Adaptive Learning Plan**

Objective: End-to-end course completion with personalization

Features:

- Weekly adaptive learning plans
- Concept-level explanation delivery
- Tone and difficulty adjustment per learner
- Continuous assessments and feedback loops
- Dynamic rescheduling based on performance

Value:

- Replaces rigid tuition schedules
- Offers individualized pacing without human dependency

---

3. **Premium Tier – Exam Readiness & Predictive Insights**

Objective: Performance optimization and outcome forecasting

Features:

- Chapter-wise and full-length mock tests
- Previous year question paper analysis
- Strength–weakness mapping by topic
- Score prediction and readiness estimation

- Targeted revision strategies

Value:

- Transforms preparation from intuition-driven to data-driven
- Provides confidence through measurable readiness indicators

---

2.9.3 Institutional / Deployment Model (Future)

- School-level licensing
- NGO and government partnerships
- Offline or on-premise deployments
- White-labeled learning systems

2.9.4 Long-Term Vision

The long-term vision of Mentorix is to evolve into a **universal autonomous learning agent framework** capable of serving learners across:

- Any subject
- Any academic level
- Any geographical or economic constraint

Key Expansion Dimensions

Subject Expansion

- Plug-and-play subject modules
- Domain-specific content agents
- Reusable learner profiling across subjects

Grade & Age Expansion

- Primary education (foundational learning)
- Higher secondary and competitive exams
- Adult learning and reskilling

Multi-Modal Learning

- Text + visual + audio explanations
- Animated and interactive content generation
- Personalized memory cards and mind maps

Mentorix is not a tutoring app; it is a **learning operating system**.

2.9.5 Social Purpose & Educational Equity

Mentorix is fundamentally aligned with **educational democratization**.

Addressing Cost Barriers

- Eliminates dependency on expensive private tutoring
- Scales personalized instruction without linear human cost
- Offers consistent quality regardless of location

Addressing Accessibility Barriers

- Can function in low-bandwidth environments
- Supports asynchronous learning
- Works without constant internet connectivity

2.9.6 Local Deployment & Decentralized Education Model

A core long-term goal is to enable **localized deployments** using **local or private LLMs**.

Offline / On-Premise Deployment

- Local LLMs hosted within schools or community centers
- On-device or edge-based inference
- No external data sharing
- Enhanced data privacy

Use Cases

- Rural schools with limited internet
- Government-funded education centers

- NGOs serving underserved populations
- Disaster-affected or remote regions

This model ensures:

- Data sovereignty
- Cost control
- Long-term sustainability
- Cultural and linguistic adaptability

2.9.7 Broader Societal Impact

Mentorix contributes to:

- Reducing learning inequality
- Enabling personalized education at scale
- Supporting self-paced, dignity-driven learning
- Complementing teachers rather than replacing them

By shifting personalization from human effort to autonomous agents, Mentorix enables **equitable education without compromising quality**.

2.9.8 Ethical & Responsible AI Perspective

- Transparent explainability via Reflection Agent
- No black-box grading decisions
- Learner-aware adaptation instead of ranking-based comparison
- Bias mitigation through continuous learner-specific grounding

---

2.10 Implementation Roadmap

(30-Day Capstone Plan)

Phase 1: Planning & Design (Days 1–5)

Week 1

Goals:

✅ Finalize architecture
✅ Define agents
✅ Setup GitHub
✅ Design DB schema
✅ Write proposal

Deliverables:

- System diagram
- Repo structure
- Tech stack
- Project board

Phase 2: Core Infrastructure (Days 6–12)

Week 2

Goals:

✅ FastAPI backend
✅ Auth + user management
✅ PostgreSQL setup
✅ FAISS pipeline
✅ PDF ingestion

Deliverables:

- Working backend
- RAG pipeline
- Data pipelines

Phase 3: Agent System (Days 13–20)

Week 3

Goals:

✅ Agent Manager
✅ Onboarding Agent
✅ Planner Agent
✅ Content Agent
✅ Memory system

Deliverables:

- Multi-agent framework
- Working workflows
- Initial personalization

Phase 4: Assessment & Analytics (Days 21–25)

Week 4 (Part 1)

Goals:

✅ Test generator
✅ Evaluation engine
✅ Skill mapping
✅ Prediction models

Deliverables:

- Adaptive tests
- Progress dashboard

Phase 5: UI, Orchestration & Deployment (Days 26–30)

Week 4 (Part 2)

Goals:

✅ React frontend
✅ Scheduler
✅ Notifications
✅ Demo hosting
✅ Documentation

Deliverables:

- Hosted demo
- Complete UI
- Final README
- Pitch deck

Risk Buffer (Optional Days 31–33)

- Performance tuning
- Bug fixes
- Scaling tests

2.11 References

1. EduPlanner: LLM-Based Multi-Agent Systems for Customized and Intelligent Instructional Design
   https://arxiv.org/pdf/2504.05370
2. Bringing Generative AI to Adaptive Learning in Education
   https://arxiv.org/pdf/2402.14601
3. Stanford AI Hub for Education
   https://scale.stanford.edu/ai
