Team Details

a. Team name: ஆசிரியர் (Aasiriyar AI Team)
b. Team leader name: Soundararajan Arunachalam
c. Problem statement: Empowering Teachers in Multi-Grade Classrooms

Sahayak: AI Teaching Assistant

Being a teacher or working in education can be super rewarding, but let's face it, the workload, especially all the **prep work**, can be challenging! Plus, there's often not enough **staff and tutoring** can be **expensive**. That's why we're proposing an AI-powered teaching assistant Sahayak.

What Sahayak Does

✓ Simplifies Prep Work – Auto-generates lesson plans, quizzes, worksheets
✓ Supports Local Languages – Handles regional input via Gemini
✓ Creates Interactive Content – Visuals, games, and learning materials
✓ Adds Audio Power – Audio materials and speech-driven responses

Built With

Google technologies: ADK, Vertex AI, Gemini API, Firebase Studio.

Impact

Empowers teachers, enables personalized learning, and brings AI into classrooms that need it most.

Opportunities

How different is it from existing solutions?

- ✓ **Hyper-localization**: Unlike generic EdTech platforms, Sahayak generates content in regional languages with culturally relevant examples and contexts specific to rural and semi-urban India.
- ✓ Most edtech platforms focus on student-facing tools or generic content delivery, this is **teacher-based tool**.
- ✓ **Low-tech compatibility**: Works with minimal technology (can export to printable formats, create blackboard-friendly visuals) unlike solutions requiring 1:1 devices.
- ✓ **Multimodal input/output**: Accepts voice, text, or images as input and generates diverse output formats suitable for varied teaching contexts.

How it solves the problem

- ✓ **Reduces preparation time**: Automates the creation of lesson plans, worksheets, and visual aids that would normally take hours.
- ✓ **Enables differentiation**: Instantly creates variations of the same material at different complexity levels for different grades.
- ✓ **Bridges resource gaps**: Provides access to high-quality teaching materials even in schools with limited resources.
- ✓ **Supports local language**: Functions in local languages, enabling teachers to provide instruction in students' mother tongues.

Opportunities

USP of the proposed solution

- ✓ **Designed for the multi-grade classroom reality** of rural India rather than imposing Western single-grade classroom models.
- ✓ **Works within existing constraints** (limited connectivity, minimal technology, vernacular medium) rather than requiring significant infrastructure upgrades.
- ✓ **Culturally contextual content generation** that reflects local communities, traditions, and references.
- ✓ **Adaptable to varying levels of teacher tech proficiency** with simple, intuitive interfaces.

List of features offered by the solution

Course material Generation

- Lesson Planner
- Assessment Builder
- Worksheet Creator

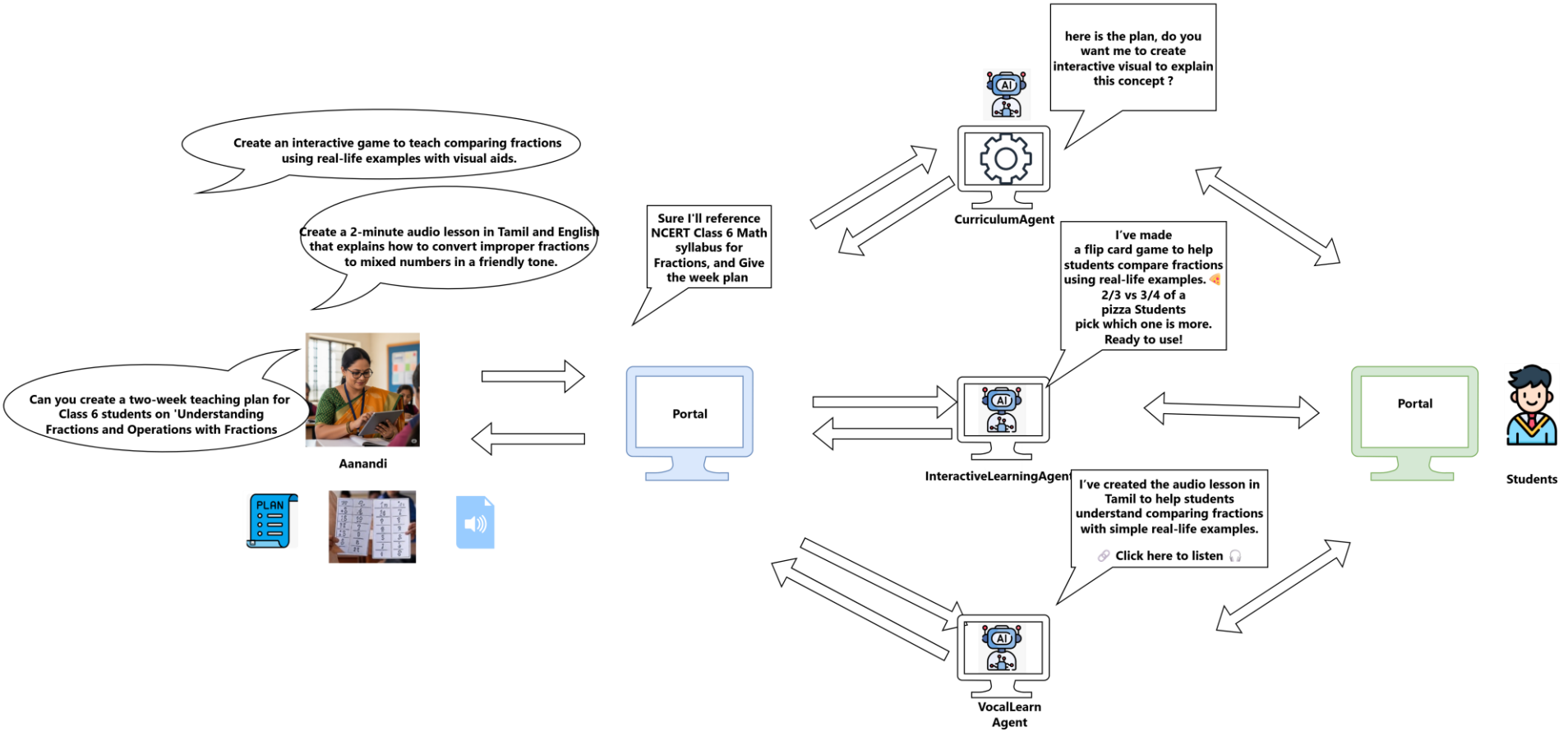
Multilingual interactive learning

- Blackboard-Friendly Diagrams
- Chart Generator
- Concept Visualization
- Game Builder (Activity ,Quiz)

Teacher & Students Support Tools

- GUI
- Speech-to-Text Q&A
- Resource Library
- Export Options (print/share-friendly)

Process flow diagram or use-case diagram



Architecture diagram of the proposed solution

Google Cloud

Agents

Cloud Run
Portal Agent
Python Flask

Cloud Run
Curriculum Agent
Python Flask

Lesson Planner Agent
Assessment Builder Agent
Worksheet Creator Agent

Cloud Run
InteractiveLearning Agent (quiz, game)

GameBuilderAgent
ConceptExplainer Agent
DiagrmAgent

Cloud Run
VocalLearn Agent

NarrationAgent
Localization Agent
TTSAgent
AudioPackage and DeliveryAgent

Data

Cloud Storage
Textbooks

Cloud Storage
Lessons plan

Cloud Storage
Assessments

Firestore
Dynamic contents (Quiz/infographics)
HTML

AI

Vertex AI
Vector Search

VertexAI
Gemini2.5 flash

Events

Pub/Sub
Topi: content-generated

Eventarc
InteractiveLearning Agent topic trigger

Eventarc
VocalLearn Agent topic trigger

Internet

Miscellaneous

Artifact Registry
Agent containers

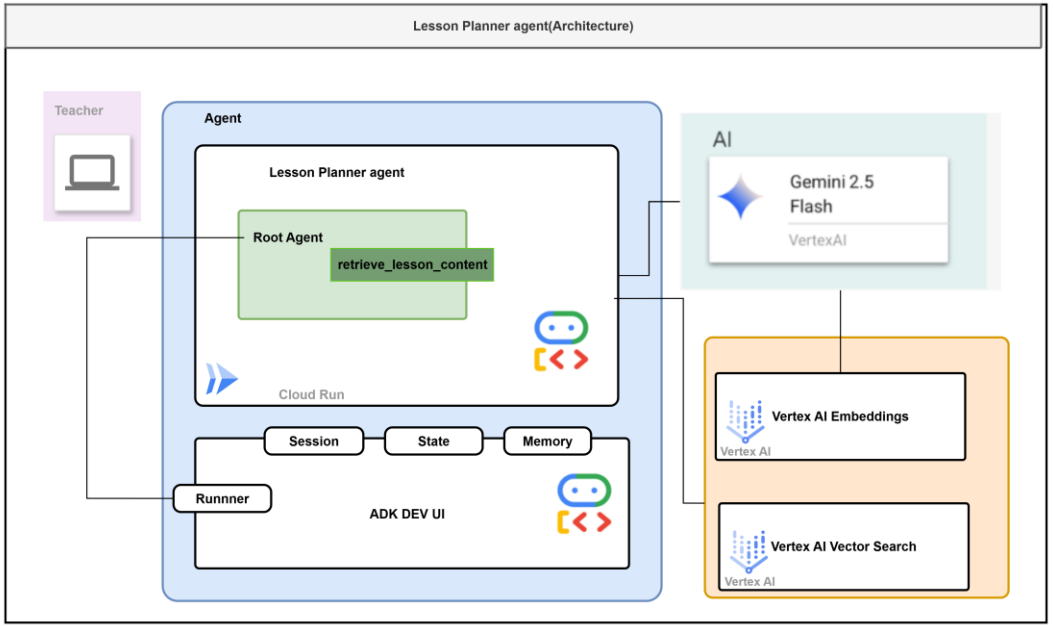
Safe and Responsible AI Agents / Guardrails

Teacher

Teacher

Students

Basic Agent detailed view with ADK WEB



1. Teacher submits a request via Runner UI.
2. Runner sets up session, state, and memory.
3. Root Agent receives and handles the request.
4. Agent calls retrieve_lesson_content.
5. Query is embedded using Vertex AI.
6. Vector Search fetches relevant content.
7. Gemini Flash generates lesson plan.
8. Plan is sent back to the teacher.

Technologies to be used in the solution

Google Cloud Platform (GCP): Central to the entire system:

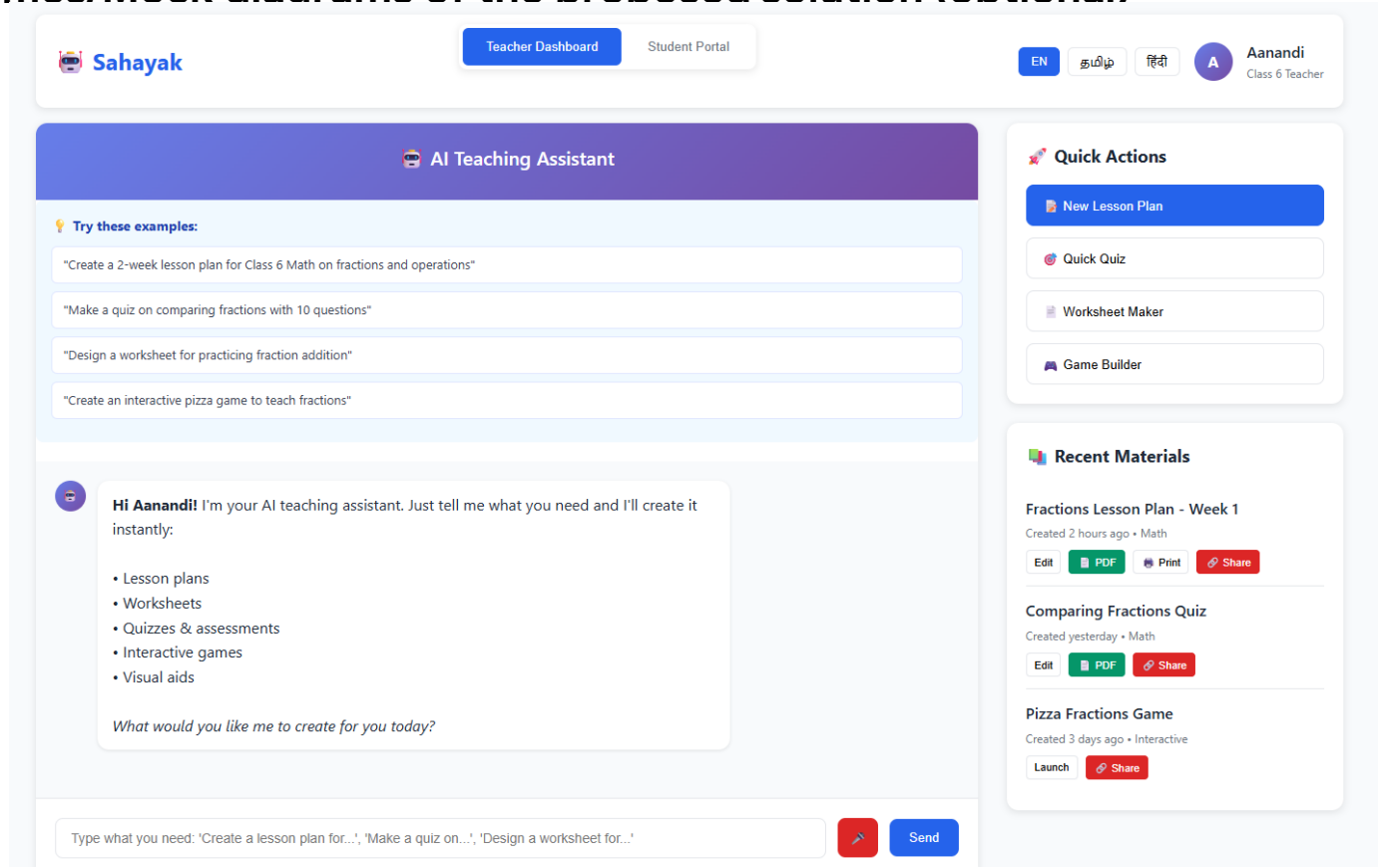
- **Vertex AI**: Google's powerful Gemini 2.5 flash(multi model) LLM for various AI tasks alongside vector search capabilities.
- **Cloud Run**: Serverless platform for deploying containerized agents and functions.
- **Pub/Sub & Eventarc**: to enable event-driven architecture, enabling asynchronous communication between components.
- **Cloud Storage**: Stores PDF , audio recaps and assignment files.
- **Firestore** – to handle dynamic(HTML / Json) data hosting
- **Artifact Registry**: Stores Docker images for the agents.

Google's Gemini models (Gemini 2.5 flash): The core large language model responsible for intelligent tasks like lesson planning, content creation, dynamic HTML generation, quiz explanations, and assignment consolidation.

Google ADK(agent development kit) : Frameworks for Agent LLM Application Development
Facilitates the creation of complex multi-agent workflows. Enables the intelligent orchestration of tools (API calls, external tools, web searches).
Implements event-driven architecture for system scalability and flexibility.

In essence, our architecture combines the power of LLMs with structured data and event-driven communication, all running on Google Cloud. This lets us build a scalable, reliable, and effective teaching assistant.

Wireframes/Mock diagrams of the proposed solution (optional)



Google Cloud

PRESENTS

Agentic AI Day

Build the next generation of intelligent agents

Powered by H2S

Thank you!