Problem Statement:

AI-Powered Interactive Learning Assistant for Classrooms

Unique Idea Brief (Solution)

AI Learning Assistant is an AI-powered interactive learning assistant designed to create a more dynamic and responsive classroom environment. It addresses the challenge of providing personalized support to students by acting as an on- demand tutor that can understand and respond to queries in multiple formats. By integrating real-time engagement monitoring, it also offers educators subtle, actionable insights into student comprehension and focus, allowing for timely adjustments to teaching strategies.

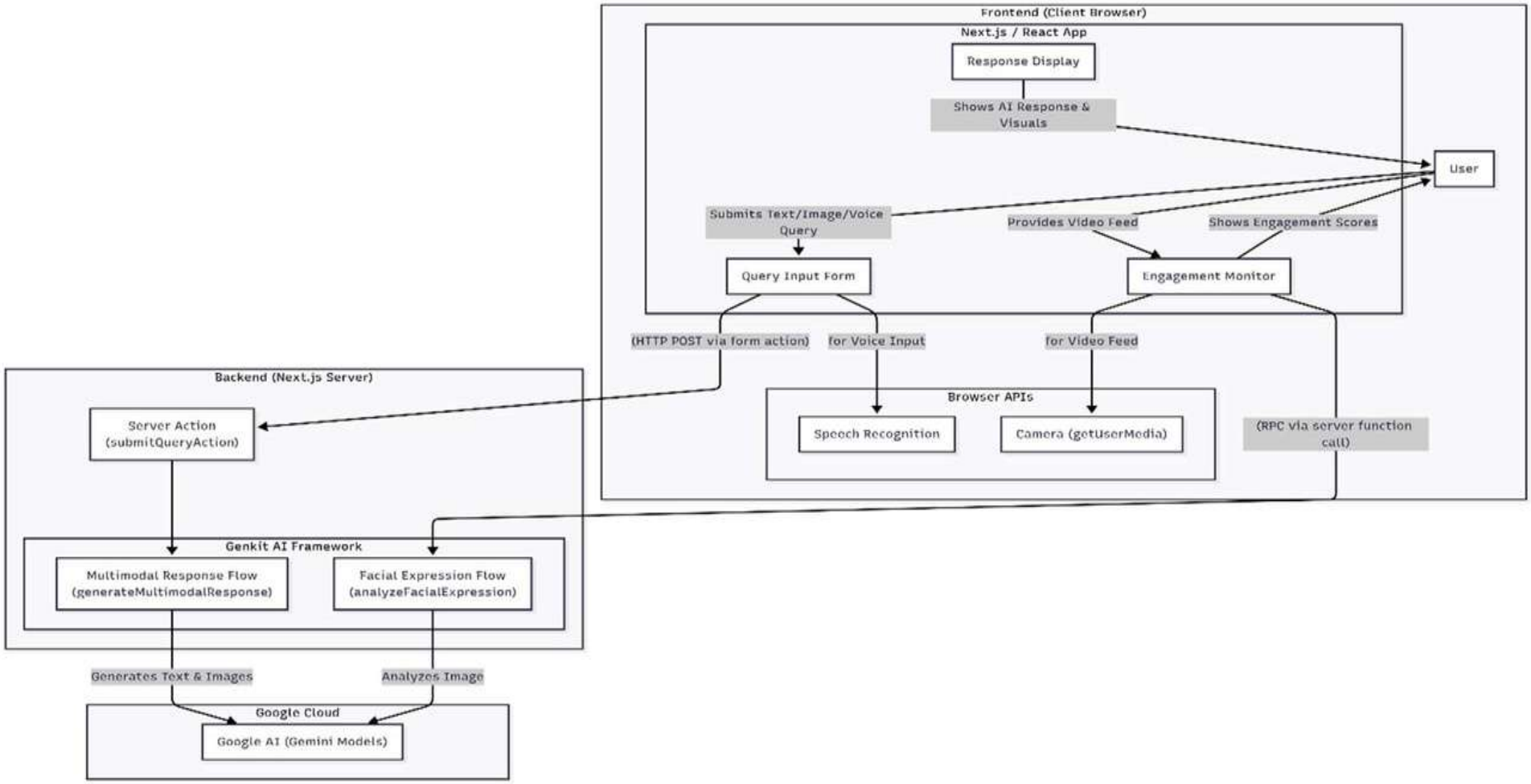
Features Offered

- **Multimodal Query Input**: Students can ask questions using text, voice (speech-to-text), or by uploading images for visual context.
- **Multimodal AI Response**: The application generates comprehensive answers that combine clear textual explanations with relevant, AI-generated visual aids like charts and diagrams.
- **Live Engagement Monitoring**: Using the device's camera, the application analyzes a student's facial expressions to provide real-time, non-intrusive feedback on their estimated levels of engagement, attention, and confusion.
- **Actionable Recommendations**: Based on the engagement analysis, the system provides simple, actionable recommendations to the educator to help improve student understanding (e.g., "Consider re-explaining the last concept in a different way").
- **Modern & Focused UI**: The user interface is clean, intuitive, and designed with a specific color palette (deep teal, light grey, olive green) to promote focus and reduce eye strain.

Process flow

- **User Interaction** : A student initiates a query through the "Ask a Question" form. They can type a question, use the microphone to speak their question, or upload a relevant image.
- **Backend Request**: The query is submitted to a Next.js Server Action. This action securely packages the input data.
- **AI Processing**: The Server Action invokes a Genkit AI flow (generateMultimodalResponse). This flow processes the text, voice, or image data and uses the Gemini model to generate a textual explanation. If the query suggests a need for a visual, it also instructs the model to create a chart or diagram.
- **Displaying Results**: The generated text and visual aid (as a data URI) are sent back to the frontend and displayed in the "Response" panel. The text can also be read aloud using the browser's text-to-speech capabilities.
- **Engagement Analysis (Parallel Flow):** Simultaneously, the "Engagement Monitor" panel, if enabled, accesses the user's webcam.

Architecture Diagram



Technologies used

- **Framework**: Next.js(App Router)
- **Language**: TypeScript
- **AI Integration**: Google AI via Genkit for defining and managing AI flows.
- **UI Library**: ShadCN UI for pre-built, accessible React components.
- **Styling**: Tailwind CSS for utility-first styling.
- **Form Handling**: React Hooks (useActionState) for managing form submissions and state.
- **Icons**: Lucide React

Team members and contribution:

- **Dyapa Nikitha** : Integrated Genkit and Gemini AI, Handled Engagement Monitor, text-to-speech integration.
- **Dumpala Lokesh** : Build frontend components, connected input modules, Image analysis integration.
- **Bheemanapally Abhaya Sri** : Designed UI, Documentation, speech-to-text integration.

Conclusion

AI Learning Assistant serves as a powerful proof-of-concept for the future of AI in education. By combining multimodal interaction with real-time analytics, it provides a scalable solution for personalized learning and offers educators a unique tool to better understand and respond to their students' needs in the moment. It is built on a modern, robust, and server-centric tech stack that is both performant and scalable.