COMP 4651 Group Project

MinimaLLM
Your AI-Powered Study & Research Assistant

Group 14
PARK Sangmin (20725436) | PARK Youkwang(20712623) | KIM Seoyoung (20799669)

1. Executive Summary

MinimaLLM is a study assistant project, designed to support students and researchers for learning, researching, and project-building purposes. Built upon personal experience of using AI for academic purposes, it aims to simplify and streamline the responsible use of AI in academic studies. With its intuitive interface and AI integrations, MinimaLLM helps users study smarter and more efficiently. The assistant offers a range of features, including a text summarizer that automatically summarizes texts, articles, or notes into concise summaries. It also includes a flashcard generator that converts raw text or uploaded materials into flashcards for active recall learning. Additionally, the AI-powered web search enhances queries with contextual search results and real-time information from the web, while the research and project planning toolkit assists in structuring ideas, generating outlines, and providing AI-supported project design. Users can upload PDFs, DOCX, or TXT files to extract content, summarize key points, and generate study tools.

MinimaLLM is particularly useful for students preparing for exams or assignments, researchers synthesizing sources and building literature reviews, teams organizing project documentation and research workflows, and self-learners building personalized learning plans. The assistant is built using a robust tech stack, including React.js and Vite for the frontend, FastAPI (Python) or Node.js for the backend, and Supabase for the database, authentication, and file storage. It leverages AI services such as OpenAI GPT-4o, web search APIs, and summarization agents, with storage options including Supabase Storage.

What sets MinimaLLM apart is its all-in-one functionality as an AI companion for studying, researching, and building projects. It features a simple interface with powerful behind-the-scenes logic and a modular architecture that allows for easy extension with modern AI tools. By saving time and enhancing retention through AI-enhanced learning, MinimaLLM provides a comprehensive solution for users looking to optimize their study and research processes.

However, the usage of network communication for a chat-based system requires careful standards and communication calls, which presents challenges for this project. Ensuring robust and efficient communication protocols is crucial for the operation of MinimaLLM, facilitating real-time interactions and enhancing the overall user experience.

2. Methodologies

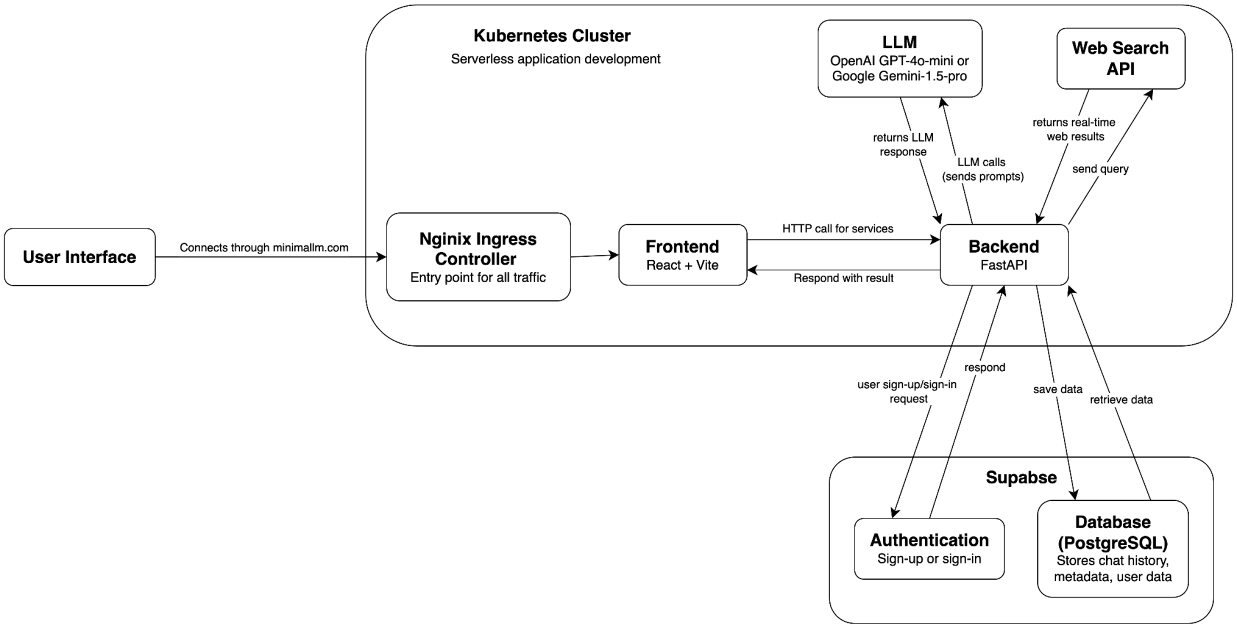

Kubernetes Cluster
Serverless application development
LLM
OpenAI GPT-4o-mini or Google Gemini-1.5-pro
Web Search API
returns LLM response
LLM calls (sends prompts)
returns real-time web results
send query
User Interface
Connects through minimallm.com
Nginix Ingress Controller
Entry point for all traffic
Frontend
React + Vite
HTTP call for services
Respond with result
Backend
FastAPI
user sign-up/sign-in request
respond
save data
retrieve data
Supabse
Authentication
Sign-up or sign-in
Database (PostgreSQL)
Stores chat history, metadata, user data

Figure 1. The system architecture of MinimaLLM

2.1 Backend

The backend of MinimaLLM is built using a combination of Vite, Supabase, and FastAPI to manage API communication effectively. This approach allows us to leverage various open-source tools for development, providing a modular and flexible architecture distinct from all-in-one systems like Amazon Web Services (AWS).

As a deployable, general practice is to utilize AWS EC2 as it offers a robust cloud computing environment that can simplify deployment and scalability. However, it requires meticulous planning and coordination when integrating various tools for database management, API fetching, runtime communication, and deployment.

There are a few advantages of Using AWS, listed below:

- **Scalability**: AWS EC2 provides the ability to scale resources up or down based on demand, allowing for efficient handling of varying workloads.
- **Reliability**: AWS offers high availability and redundancy, ensuring that applications remain operational and data is protected against failures.
- **Comprehensive Services**: AWS includes a wide range of services, from database management (RDS, DynamoDB) to machine learning (SageMaker) and more, enabling developers to build complex applications without needing to integrate multiple tools.
- **Security**: AWS provides robust security features, including identity and access management, encryption, and compliance certifications, which can help safeguard sensitive data.

Nevertheless, there are also some drawbacks of AWS:

- **Cost**: AWS can become expensive, especially for small projects or startups. Costs can escalate quickly with increased usage, and budgeting can be challenging due to the pay-as-you-go model.
- **Complexity**: While AWS offers many services, navigating the AWS ecosystem can be overwhelming for new developers. The complexity of managing multiple services may lead to configuration errors or mismanagement.
- **Vendor Lock-In**: Relying heavily on AWS services can lead to vendor lock-in, making it difficult to migrate to other platforms or solutions in the future.

Due to such drawbacks and the nature of our project architecture, we have decided to utilize the Kubernetes (K8s) cluster along with a Dockerized application.

Docker and K8s provide an alternative approach to application deployment and management. It allows developers to easily dockerize their applications and packages using Dockerfiles and also using the robust support of docker for K8s, deploy the docker images to a K8s server with minimal effort. This simplifies the dependencies, environment, and OS-specific issues during production and maintenance of the project.

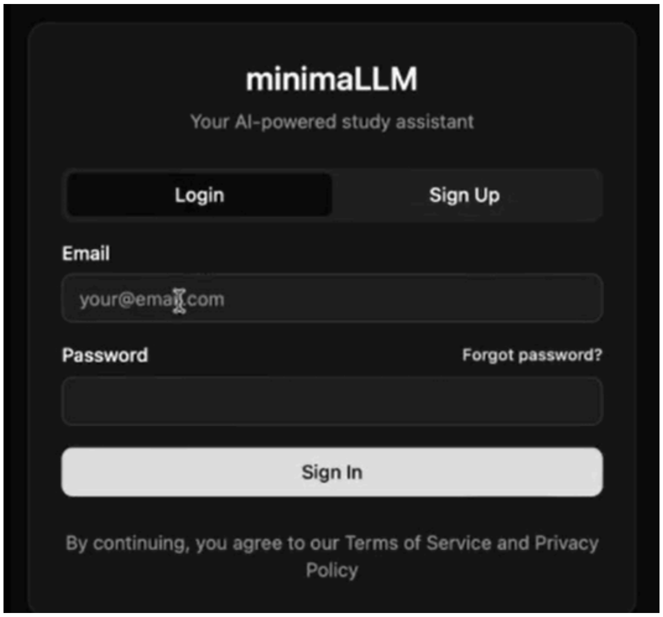
The usage of Docker and Kubernetes provides some advantages:

- **Portability**: Docker containers can run on any system that supports Docker, making it easy to move applications between development, testing, and production environments.
- **Efficiency**: Docker allows for rapid deployment and scaling of applications. Containers can be started and stopped quickly, making it easier to manage resources and respond to changes in demand.
- **Integration with CI/CD**: Docker integrates well with continuous integration and continuous deployment (CI/CD) pipelines, facilitating automated testing and deployment.
- **Customizability**: As users are able to fully customize the K8s system parameters, it could have a much more robust and task-specific system.
- **Scalability**: Users can decide on the scalability of the system, as adding extra nodes and storage is all that is needed.

However, there still exists some drawbacks:

- **Complexity in Orchestration**: While Docker simplifies application deployment, managing multiple containers can become complex without orchestration tools like Kubernetes.
- **Resource Overhead**: Although containers are lightweight, they still require resources to run. In resource-constrained environments, this can be a consideration.
- **Requires Expertise**: Initially setting up a K8s server environment can be challenging without knowledge of networks and database systems.
- **Initially Costly**: If a user decides to manage their own K8s cluster, it may have a higher initial cost than utilizing EC2 servers. But after the initial cost, it would be more cost-effective in the long run.

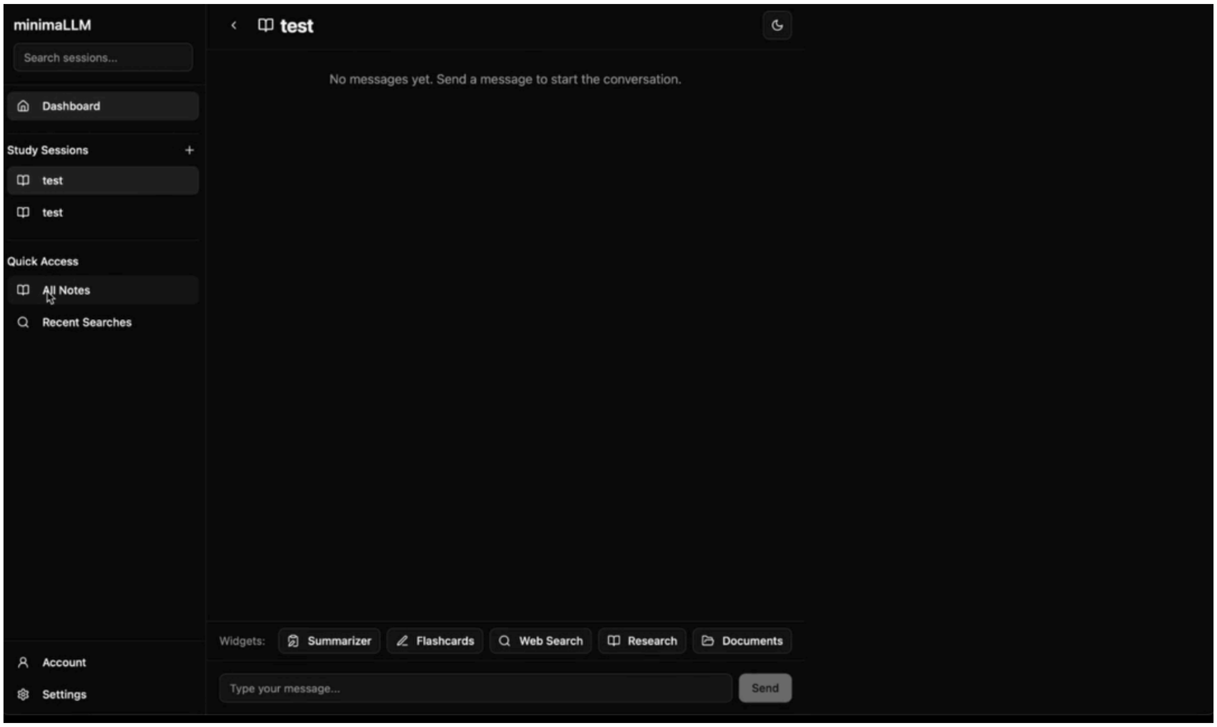
2.2 Frontend



Your AI-powered study assistant

By continuing, you agree to our Terms of Service and Privacy Policy

Figure 2. Login and sign-up user interface



No messages yet. Send a message to start the conversation.

Figure 3. MinimaLLM main page user interface

When the user successfully logs in, as illustrated in Figure 2, a new session is created and it directs to the main page shown in Figure 3. Users can type messages to interact with the model (figure 3). Below, there are summarizer, Flashcards, Web search, Research, and Documents widgets to choose from. Summarizer can summarize the user input texts, Flashcards can generate flashcards for study based on user-attached materials, Web search can search the user input topic on the internet, Research will plan out the research plan based on the research topic and details provided by the user, and Documents allow users to upload PDF, DOCX, and TXT files.

The list of features mentioned above can be experienced through the presentation video.

3. Conclusion

In conclusion, MinimaLLM offers users a tool for easier studying, researching, and project-building. By providing a comprehensive set of functions such as the text summarizer, flashcard generator, web search, research planner, and document upload features, MinimaLLM empowers students, researchers, and creators to enhance their learning and productivity effectively.

Moreover, the modular and extendable system of MinimaLLM ensures flexibility and adaptability to cater to diverse user needs and evolving technological advancements. This architecture allows for the seamless integration of new AI tools and functionalities, ensuring that the assistant remains relevant and cutting-edge in the rapidly changing landscape of educational technology.

Furthermore, the decision to deploy MinimaLLM serverless with a Kubernetes cluster showcases a commitment to efficiency, scalability, and cost-effectiveness. By leveraging Kubernetes for container orchestration and serverless computing for resource optimization, MinimaLLM can deliver a seamless and responsive user experience while effectively managing computational resources.

In conclusion, MinimaLLM represents a transformative solution for individuals to optimize their study and research processes through AI technology. Its user-centric design, robust tech stack, and emphasis on enhancing learning outcomes make it a valuable asset for students, researchers, and self-learners. As the project continues to evolve and expand its capabilities, MinimaLLM is poised to make a lasting impact in the educational technology space, revolutionizing the way users engage with information and knowledge.