



STARS GPT (AI-Based Tutoring Chatbot)

Students: Vincent Carrancho, Robin Obregon, Stephanie Torres, Nicholas Belschner, Garvee Valcourt

Mentor: Dr. Patricia McDermott-Wells

Instructor/Faculty: Prof Seyedmasoud Sadjadi, Florida International University

Summary

This app aims to support the STARS Tutoring initiative by providing tutors with a powerful tool that can assist their students. The chatbot answers questions, clarifies concepts, and draws from years of data to provide students with the strongest support possible. In combination with the guidance of a STARS Tutor, this tool can further help students by being available 24/7 and making sure FIU students have a well-rounded learning environment.

Problem Statement

As LLMs and AI models such as ChatGPT and Google Gemini become more popular, they provide a risk to student learning as an overreliance on these technologies becomes a challenge to students and professors. Using the general-purpose model for educational purposes is also difficult and leads to inconsistent results. In order to address this concern, we developed a fine-tuned model that leverages data from past assignments in courses taught by Dr. McDermott-Wells to be trained in a way that it can positively impact the learning of students.

Technologies Used

Supabase: A comprehensive Backend-as-a-Service (BAAS) that includes authentication, database support (PostgreSQL), and more.

OpenAI API: Used to send messages to and receive responses from the AI Model.

Render: Deployment service used for frontend deployment

AWS: Deployment service used for deployment of backend

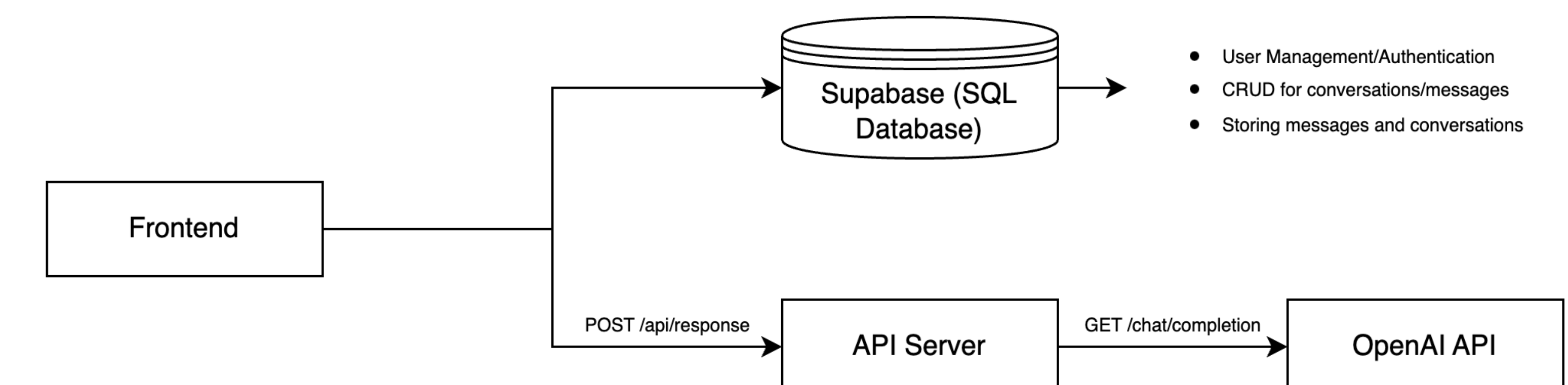
API Server Design

Deploying a FastAPI application on AWS EC2 involves setting up an EC2 instance running on a selected AMI, managing traffic through security groups, and utilizing networking components like VPCs and Internet Gateways. The FastAPI application, powered by Uvicorn, efficiently handles web requests. Optional components like Nginx as a reverse proxy enhance security and performance. AWS's scalability features like Auto Scaling and Load Balancing ensure optimal resource utilization. Additionally, implementing Continuous Integration and Continuous Deployment (CI/CD) pipelines automates the deployment process, ensuring rapid and reliable updates to the application. Overall, this architecture provides a secure, scalable, and efficient environment for hosting FastAPI applications on AWS EC2.

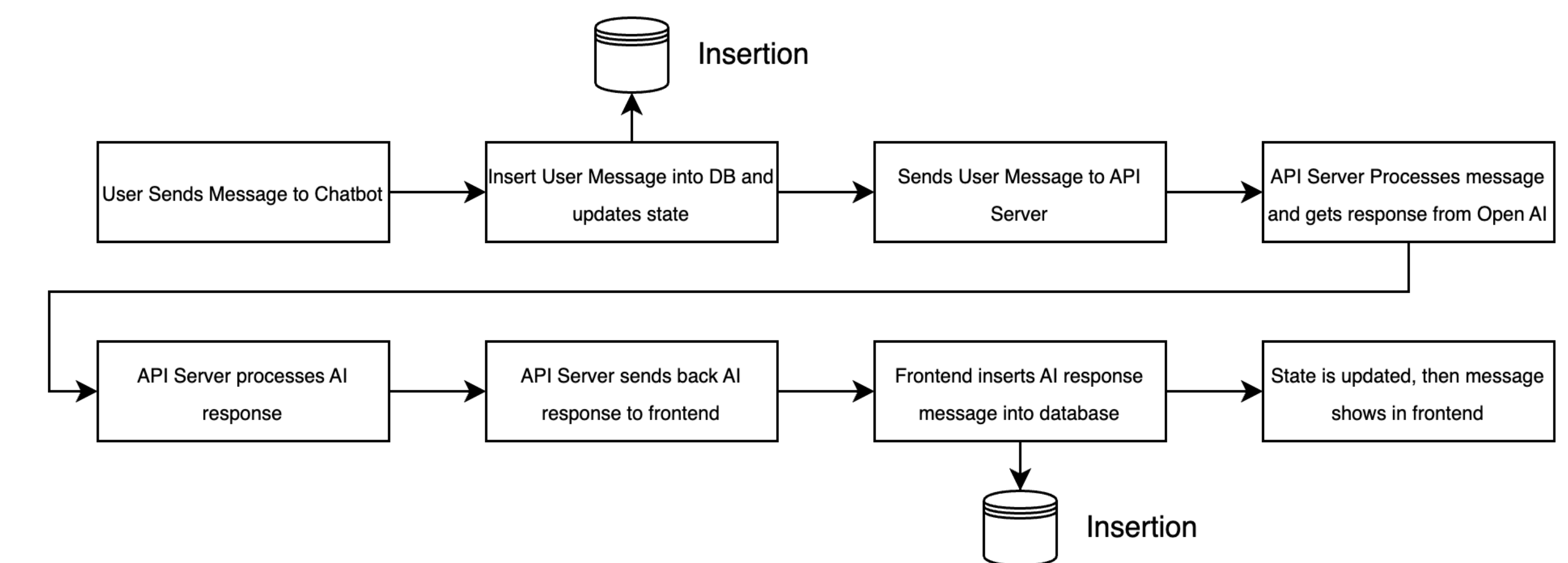
Deployment

In the deployment process for our application, we utilize two distinct platforms: Render for the frontend and AWS EC2 for the backend. On Render, the frontend components of our application, such as the user interface and client-side logic, are deployed. Render provides a seamless environment for hosting static assets and frontend code, ensuring fast and reliable delivery to users. Meanwhile, for the backend infrastructure, AWS EC2 serves as the backbone. Here, we deploy the backend components, including the FastAPI application and associated services. AWS EC2 offers scalability, security, and flexibility, allowing us to manage the server architecture efficiently. By leveraging these platforms in tandem, we ensure a comprehensive deployment strategy that addresses both frontend and backend requirements effectively.

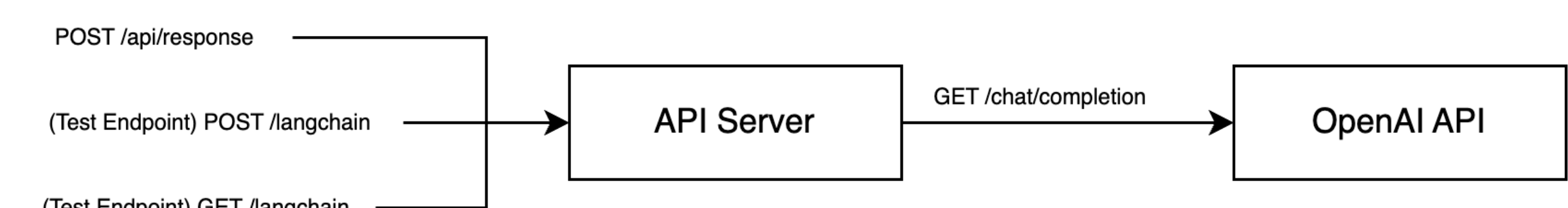
System Architecture:



AI Messaging System:



API Server Design



Screenshots

```
welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1014-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Mon Apr 15 16:31:04 UTC 2024

System load:  0.0          Processes:      100
Usage of /:   39.2% of 7.57GB Users logged in:    0
Memory usage: 27%          IPv4 address for eth0: 172.31.3.216
Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.
   https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.
21 updates can be applied immediately.
To see these additional updates run: apt list --upgradable
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

*** System restart required ***
Last login: Mon Apr 15 00:58:44 2024 from 174.48.20.201
ubuntu@ip-172-31-3-216:~$ ls
backend
ubuntu@ip-172-31-3-216:~$ cd backend
ubuntu@ip-172-31-3-216:~/backend$ ls
package-list.txt  readme.md  requirements.txt  src  tests
ubuntu@ip-172-31-3-216:~/backend$ cd src
ubuntu@ip-172-31-3-216:~/backend/src$ python3 -m uvicorn main:app
INFO: Started server process [12421]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
```

The AWS EC2 instance operates on an Ubuntu image and is accessible via SSH. This configuration ensures that we can securely manage and interact with the instance remotely, enabling efficient deployment and administration of the application hosted on AWS EC2.



ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by STARS Tutoring and Dr. Patricia McDermott-Wells. We thank Dr. McDermott-Wells for their assistance and mentorship that we received throughout the senior design project.