

---

# HashBit PSGCT

INCIDENT RESOLVER

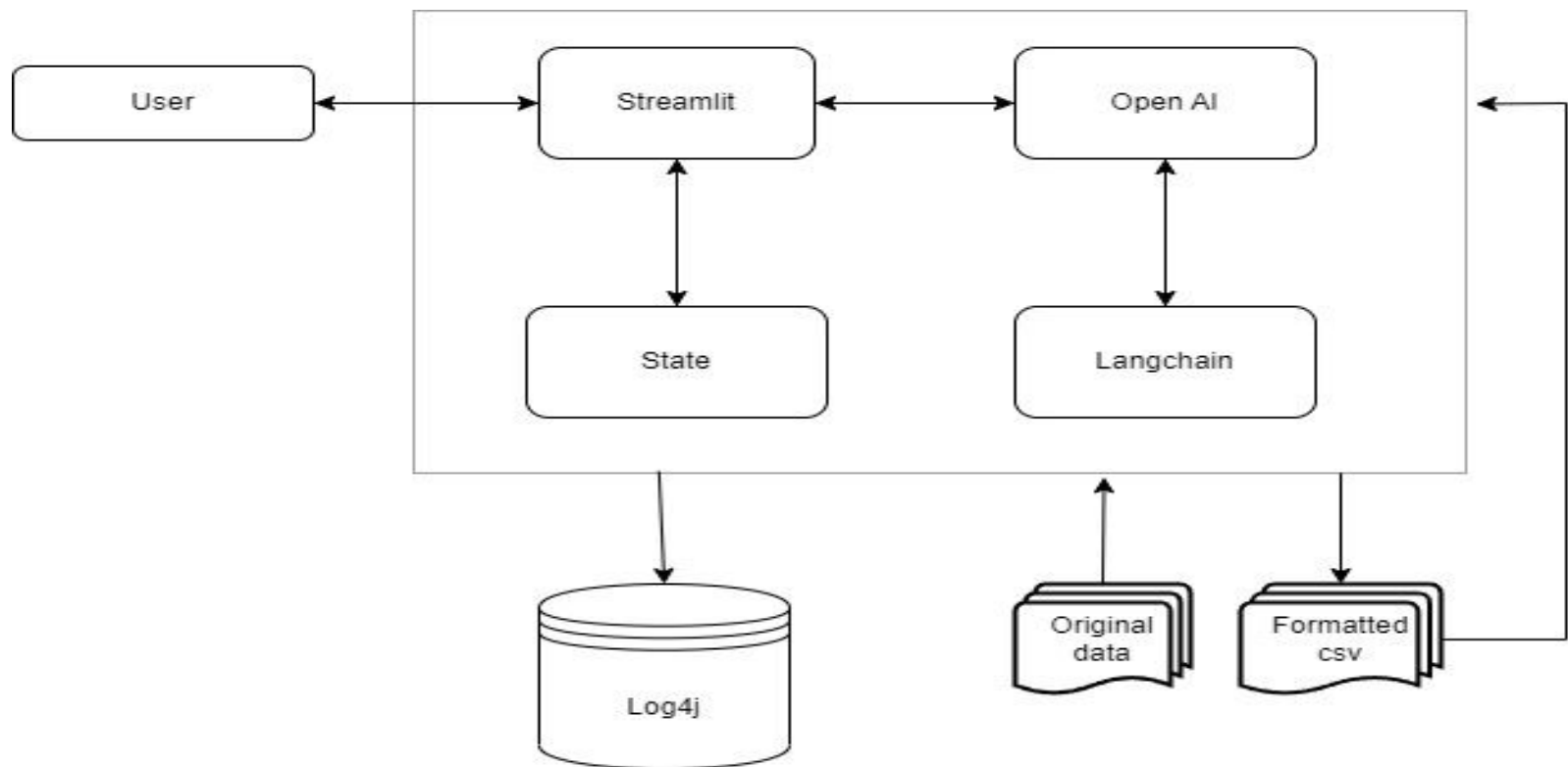
---

20i306 - Akash S P  
20i321 - Harshan R

# About

The main goal of this project is to create a strong and flexible platform that can automatically handle and solve incidents using AI methods. By looking at past data and patterns, participants will build a system that can find solutions for specific incidents. Additionally, the system will also automatically put these solutions into action, considering how often incidents happen and how successful the solutions have been historically.

# Architecture



# What is needed?

- To create an extensive incident-resolution database that covers a diverse range of incidents and their respective resolutions.
- To develop a sophisticated platform that integrates the capabilities of artificial intelligence.
- Implement a solution to incorporate self-healing mechanism that leverage historical data.
- A UI which facilitate user interaction, allowing users to report incidents and observe resolution outcomes.

# Work done

- Expanded the Dataset with additional data points to encompass a broader spectrum of incidents
- Preprocessing of the Dataset to establish a high-quality dataset that serves as the bedrock for subsequent analyses.
- GUI created with Streamlit which not only acts as the entry point for user interactions but also serves as the conduit through which users receive resolutions to their specific incidents.
- ChatGPT, a LLM AI framework has been integrated to search among the context
- Log4j for logging events

# Internal Working

## **User Query Integration via Streamlit GUI:**

- The Streamlit graphical user interface (GUI) serves as the ingress point for users to input their queries or prompts concerning specific incidents or issues necessitating resolutions.
- User interaction is channeled through this interface by entering queries into the designated input field, thereby initiating the systematic resolution-seeking process.

## **Query Processing and Langchain Incorporation:**

- The user-provided query undergoes processing via integration with the Langchain module.
- Langchain serves as a sophisticated language-processing component responsible for deciphering, parsing, and extracting essential contextual elements from the user's query. This entails converting natural language inputs into computationally accessible data structures for further analysis.

## Historical Data Analytics via Langchain Integration:

- Langchain interfaces with the repository housing historical data, employing its advanced natural language processing (NLP) capabilities for in-depth analysis.
- The historical data repository houses a corpus of past incidents, accompanied by their respective queries and associated resolutions.
- Langchain's NLP prowess enables the identification of intricate patterns, correlations, and potential resolutions entwined within the historical dataset.

## Resolution Rendering:

- Relying on the outcomes of historical data analysis, Langchain ascertains the most suitable resolution aligned with the user's query.
- The system proceeds to manifest this identified resolution within the Streamlit GUI interface.
- Users are then presented with a curated resolution, furnishing them with actionable directives to effectively address the specific incident at hand.

- Following the presentation of the resolution, the system captures both the original user query and the ensuing resolution.
- This newly minted set of data points is seamlessly incorporated into the existing historical data repository, bolstering its repository with each successive interaction.
- This perpetual expansion fortifies the system's adeptness in rendering precise and efficacious resolutions over time, nurturing an evolving repository tailored to user interactions.



# Limitation

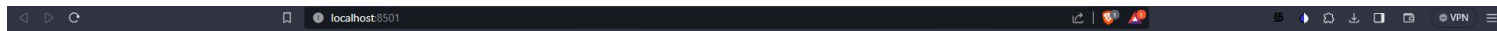
- Open AI provides a free API which has cap limit of 3 queries per minute (20 seconds between successive queries)

# Tools used

- Python
- Streamlit
- Log4j
- Open AI
- Langchain



# Outcome



RUNNING... [Stop](#) [Deploy](#)

## HashBit PSGCT Incident Resolver

Waiting for Response

Hey, my GUI is down. What to do ?

If the GUI is down, you can try restarting the App pool for the application to fix it.

Essbase config file is corrupted

To fix the issue of a corrupted Essbase config file, you can restore the config file from the backup.

failed to file transfer

I'm sorry to hear that you failed to transfer a file. Can you provide more information about the specific issue you encountered?

Any Issues ?




# Outcome




Deploy

## HashBit PSGCT Incident Resolver


 Hey, my GUI is down. What to do ?


 If the GUI is down, you can try restarting the App pool for the application to fix it.

 Essbase config file is corrupted

 To fix the issue of a corrupted Essbase config file, you can restore the config file from the backup.

 failed to file transfer

 I'm sorry to hear that you failed to transfer a file. Can you provide more information about the specific issue you encountered?

 Failed file transfer due to FTP server misconfiguration

 Corrected FTP server settings and resent the failed files.

Any Issues ?

