Creating a use case diagram for your Log Analyzer project helps visualize the interactions between users and the system. Here’s a general outline of what your use case diagram might include, along with a description.

# Use Case Diagram Components

## Actors:

**User**: The person who interacts with the Log Analyzer (e.g., a system administrator or a developer).

* + **OpenAI API**: Represents the external API being used for log analysis.

## Use Cases:

* + **Upload Log Files**: User uploads log files to the system.
  + **Analyze Logs**: User initiates log analysis using the OpenAI API.
  + **View Analysis Report**: User views the results of the log analysis.
  + **Handle Errors**: System handles any errors that occur during log analysis or API calls.

## Use Case Diagram Example

Here's a textual representation of how the use case diagram might look:

User

Upload Log Files

Analyze Logs -> OpenAI API

View Analysis Report

Handle Errors

## Description of Use Cases

* **Upload Log Files**: The user uploads log files into the system for analysis. This can be done through a user interface where the user selects files from their file system.
* **Analyze Logs**: Once the logs are uploaded, the user initiates the analysis. This involves sending the log data to the OpenAI API for processing.
* **View Analysis Report**: After the analysis is complete, the user can view a report that summarizes the results, such as detected errors, patterns, or anomalies in the logs.
* **Handle Errors**: The system must gracefully handle any errors that occur during log analysis, such as issues with the uploaded files or problems with the OpenAI API (e.g., rate limits, connection errors).