**AI-Powered Developer Performance Analytics Dashboard**

**Methodology**

**1. Data Collection:** The system retrieves data from GitHub by using a **personal access token** and a specified **repository URL**. This ensures secure access to repository information. The main data collected includes **fork statistics**, **commit frequency**, and **issue tracking (open/closed issues)**. These metrics are essential for assessing the activity and health of a repository. The system is flexible enough to handle both private and public repositories.

**2. Data Processing & Report Generation:** After data is gathered, it undergoes processing to generate meaningful insights. The processed data is organized into a structured format, with reports automatically generated in **.csv** and **.json** formats for easy sharing and further analysis. These reports contain crucial information such as the number of forks, the rate of commits, and the status of issues (open, closed, in-progress). This step ensures that key repository metrics are both captured and presented in a user-friendly format.

**3. Data Visualization:** To facilitate understanding, the system presents data through a **visualization dashboard**. This dashboard displays key metrics like **overall fork information**, **commit frequency trends**, and the **status of issues**. The visualizations offer users an at-a-glance overview of repository health and activity, allowing for quick assessments without needing to parse through raw data.

**4. Natural Language Processing (NLP) & Model Integration:** Users interact with the system through natural language queries. An **NLPProcessor** linked to a **LLaMA3.18b** model processes these queries. By using **retrieval-augmented generation (RAG)**, the system searches for relevant information based on the user’s question and generates a context-aware response. For example, users can ask about the current number of open issues or the most active period for commits, and the system will respond with accurate data.

**5. User Interface:** The system provides a **user-friendly interface** where data visualizations and responses to queries are displayed in a clean and interactive format. The UI is designed to allow both technical and non-technical users to easily navigate, request data, and interpret results.

