### **Risk Analysis System Documentation**

#### **User Authentication and Risk Analysis**

Upon logging into the portal, the user undergoes authentication. After successful verification, the system initiates the risk analysis process.

#### **Data Analysis**

Utilizing data from the **Account Aggregator**, the system further analyzes user spending and savings. This analysis considers both the mode and category of payment, which are extracted from transaction descriptions.

#### **Risk Scoring**

The risk is evaluated on a scale of **1 to 10**, where:

* **1** indicates the lowest risk,
* **10** signifies the highest risk.

The risk score assessment is based on three key parameters:

* **Income Resilience**: This measures the stability and reliability of a user’s income over time. A higher resilience indicates a steady income source, reducing the likelihood of financial distress.
* **Risk Vigilance**: This parameter assesses the user’s awareness and proactive measures regarding potential financial risks. Users who actively monitor their finances and are cautious about their spending habits are considered more vigilant.
* **Spending Propensity**: This reflects the user’s tendency to spend money based on their financial behavior. A higher propensity indicates a greater likelihood of spending, which can increase financial risk if not managed properly.

The mean of these three parameters is calculated, and the risk is categorized into the following brackets:

* **1 - 3** : Low Risk
* **4 - 7** : Medium Risk
* **8 - 10** : High Risk

#### **Recommendations**

After the final risk calculation, the user receives personalized recommendations on ways to improve their risk score.

#### 

#### 

#### **Tech Stack and Frameworks Used**

* **AWS Cognito**: It is used for secure user login and authentication, providing a scalable and reliable method for managing user sessions and credentials.
* **Python:** Utilized for writing scripts that calculate the risk score using defined formulas. Its versatility allows for effective implementation of the risk assessment logic.
* **Streamlit:** Employed for deploying the application, providing an intuitive web interface for users to interact with the risk analysis system. It facilitates easy visualization of data and results.
* **Flask:** For handling data fetching from the Account Aggregator. It serves as the back-end framework, enabling the application to retrieve and process user financial data efficiently.
* **Large Language Model (LLM)**: LLM is utilized for generating tailored recommendations based on the user's risk profile. It leverages natural language processing to provide actionable insights and guidance on improving the risk score.