# **Loan Management Data Pipeline**

## **Overview:**

In this project, **Apache Spark** and **MySQL** are used to process, transform, and manage loan-related data. Apache Spark is used for **Processing** and MySQL is used for **Data Storage** (Foreign key constraints are managed to maintain relational integrity). The system ensures data integrity, consistency, and scalability by leveraging distributed data processing and a relational database.

## **Technologies Used:**

1. **Apache Spark (PySpark):** For distributed data processing and transformation.
2. **MySQL:** For storing structured data in a relational database.
3. **Python:** For scripting and pipeline logic.
4. **MySQL Connector:** For connecting Spark to the MySQL database.

### **Assumptions:**

Since data will be coming from Loan Applications, let us assume we get data in the form of JSON files.

**Files:**

1. **Users** – Users who signed up in the application
2. **User KYC** – Details of KYC

* Approved
* Pending
* Rejected

1. **Loan Offers** – All the Loan Offers with tenure and Interest Rate (This includes total money to be paid back adding interest

* Active – Whether the loan offer still exists
* Expired – Loan offer is no more available

1. **User Loans** – Loan sanctioned to a user.
2. **Transactions**

* **Borrow** for Loan Disbursement – When a loan is sanctioned
* **Repay** for Loan Repayment

## **Implementation:**

Data Processing Pipelines:

**1. Users Pipeline:**

* Processes user data from the incoming file
* Upsert (update and insert) logic is implemented
* Existing user details will be updated (in case of any updates)
* New Users will be Inserted

**2. User KYC Pipeline:**

* Processes user kyc data from the incoming file
* Upsert (update and insert) logic is implemented
* Existing user kyc details will be updated (in case of any updates like kyc is successful or rejected)
* New Kyc requests will be Inserted

**3. Loan Offers Pipeline:**

* Processes user kyc data from the incoming file
* Upsert (update and insert) logic is implemented
* Existing loan offers details will be updated (in case of any updates like interest rate, tenure, and total amount and whether the offer is active or expired)
* New loan offers will be Inserted

**4. User Loans Pipeline:**

* Processes user loans data from the incoming file
* Tracks loans taken by users, including loan status, application dates, and approval dates.
* Upsert (update and insert) logic is implemented
* Existing user loans details will be updated (whether applied loan is approved or rejected)
* New requests will be Inserted

**5. Transactions Pipeline:**

* Processes financial transactions related to loans, such as repayments and borrowings.
* Insert logic is implemented as every transaction is a new record with a unique transaction id

**6. Loan Balances Pipeline: (Derived Table)**

* Calculates remaining loan balances, due dates, and days past due for each loan.
* Uses Transactions, User Loans, Loan Offers tables for Calculation
* Over-write logic is implemented as it calculates the loan balances of all the users every time for effectively tracking the balances

**7. Defaulters Pipeline: (Derived Table)**

* Identifies and tracks loan defaulters based on overdue payments and remaining balances.
* Over-write logic is implemented as it finds out the users who didn’t clear loans on or before due date every time for effectively tracking the defaulters

**8. Resolutions Pipeline:**

* Manages resolution data for resolved defaulters and updates loan balances accordingly.
* Based on the resolution by an executive, loan balanced will be updated so that we do not get the details of resolved defaulters again

## **Features:**

**1. Error Handling and Logging:**

* Error handling is implemented to capture and log errors during pipeline execution.
* Detailed logging ensures traceability and debugging capabilities.

**2. Scalability:**

* The pipeline is built using \*\*Apache Spark\*\*, enabling distributed data processing for large datasets.
* It can be scaled to run on a cluster for production use.