Ver.	Date		Author	Comments
1.0	**13-03-2023**		George Maina Ndiritu	First Draft

SOLUTION BACKEND LOAN LENDING MINI PROJECT

Prepared by:	George Maina Ndiritu
Prepared for:	
Date submitted:	13th March 2023

Table of Contents

1 PURPOSE

The purpose of this document is to define a design for backend loan lending platform.

2 SOLUTION OVERVIEW

2.1 Business Rules Created

1. A customer can acquire more than loan at a time.
2. No Loan offer should exceed customers loan limit
3. Notifications on all loans given/ repaid are via SMS and an email
4. The Customer can only pay the loan in full(Partial loan repayments not enabled)

2.2 Solution Domain

For the services Creation, The following services will be availed

1. Loan Offer service -> This Displays offers to the customer depending on the Loan limit the customer has
2. Apply Loan -> This Allows a customer to apply one of the loans given from the loan offers presented.
3. View Active Loan -> This Allows the customer to view his/her current outstanding loan.
4. Get Customer details -> This allows to view customer details (Personal details, wallet details)
5. View Customers -> Lists details of all customers in the system
6. Loan Repayment -> Allows a customer to repay an active loan

2.3 Scope

2.3.1 In-Scope

The scope of the solution is limited to:

1. Loan Offers presentation.
2. Loan Application
3. Loan Repayment
4. Customer detail's view
5. Auto loan repayment

2.3.2 Out-Scope

The following requirements are out of scope for this delivery:

1. Customer creation
2. Customer details update
3. SMS/Email notification to customer
4. Wallet Top ups

2.4. SYSTEM REQUIREMENTS

2.4.1 Functional Requirements

a. Customer should be able to apply for loan
b. Customer should be able to pay their active loan

2.4.2 Non-Functional Requirements

The Fiber for Business needs to meet the following non-functional requirements:

1. High availability preferably a layered architecture
2. System monitoring
   This Diagnostics page will help you to perform a Ping or the Trace route to troubleshoot the network connection.
3. Performance
4. Scalability
5. Security

2.5 Assumptions, Constraints and Open Issues

2.5.1 Assumptions

- The system already has active customers.
- There is enough money to be lend out.

3 SOLUTION ARCHITECTURE & DESIGN ATTRIBUTES

Customers details

The Application will be preloaded with few active Customers with a predefined attributes such as wallet balance and loan limit. This are predefined in a way to allow exhaustive testing of the application

Loan offers

The loan Offers will be generated based on two products

1. Product A – (Max allowable limit 1000, interest percentage 10%, tenure 15 days),

2. Product B – (Max allowable limit 2500, interest percentage 12.5%, tenure 30 days)

Scenarios

a. For Customers with a loan limit below or equal 1000 they will be given an offer with amount equivalent to their loan limit and other attributes will be of Product A ie(rate 10% tenure 15 days);
b. For Customers with loan limit greater 1000 and below 2500, they will get 2 offers one equivalent to Product A and the other with amount equal to their loan Limit . eg Customer with a loan limit of 1500 will get the following offers
    - Loan offer-> Amount : 1000, Interest rate 10%: tenure 15 Days
    - Loan Offer -> Amount: 1500, Interest rate 12.5% tenure 30 days
c. For Customers with loan Limit equivalent to 2500 or above they will get offers of both product A and B

ERD Diagrams/ Database Design

For the loan lending application, the choice of the Data storage is a relational database.

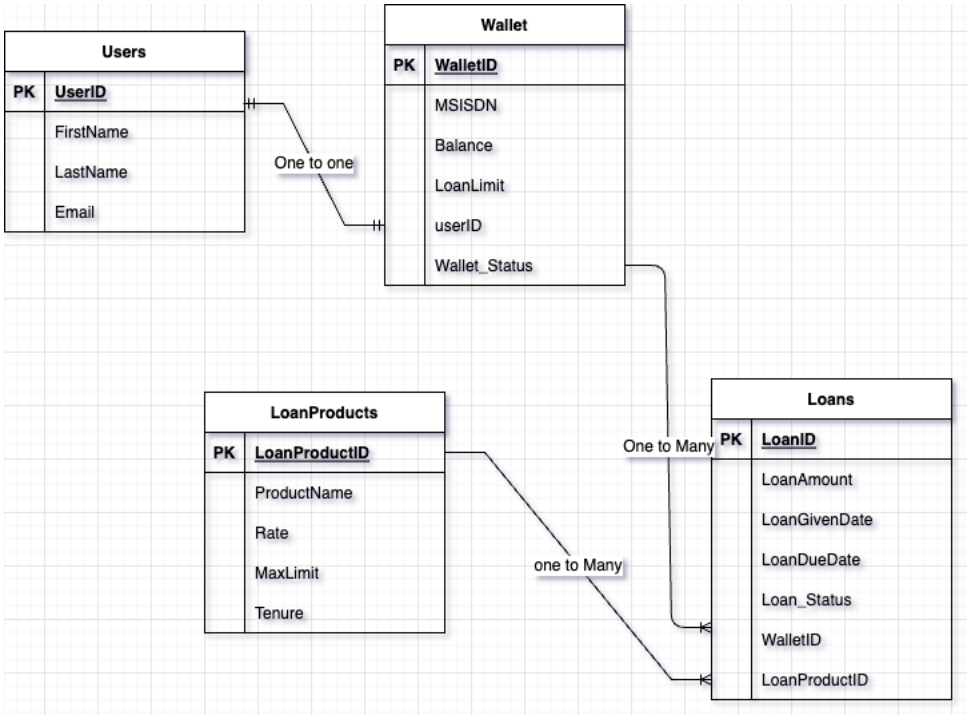
This choice is because relational database offers.

- ➔ Consistency
- ➔ Durability
- ➔ Atomicity
- ➔ Isolation

This gives a better management to the lending process though the downside of it is that it may be slow.

For this project, The Database used is a H2 database which is an in-Memory database allowing the packaging of the project in a single package with no external dependencies.

The Following is the ERD diagram for the project.



Process flows

Loan Application

- A request is made via API call to the service to get All the offers for the customer

  The Request is as Follows:

  - Method : GET
  - Parameters: UserID
  - Ressponse: A list of loan offers the customer qualifies for
- To Apply for Loan a request is sent to the service for loan application with the following
  - Method: Post
  - Header Parameter: UserID
  - Body: Loan offer
- On submission the service validates the details, if the details are correct the customer wallet is credited with the amount, a record with the amount payable after applying interest is created and due date calculated based on the loan product
- For customer notification, A log is created for send SMS and Email which is supposed to go out to the customer
- The service responds with the record of the loan details to customer showing the customer new wallet balance and the amount due for payment and payment due date

Loan Repayment process flow

- A request is made to view all customers active loans on the application
    - Method: Get
    - Parameter: UserID
- The service queries all the active Loans under the userID and presents them
- The loan to be repaid is submitted for repayment
    - Method: Post
    - Body: Loan
- On submission The service validates the request then proceeds to fulfill the request.
- If the customer has sufficient balance in his/her wallet a deduction is made and the loan is settled
- If the customer has insufficient balance a notification is sent to customer.
- The service responds with the status of the loan

Loan Repayment Auto repayment process flow

- A cron job is Scheduled to run at 12 midnight every day fetching all the loans due on that date
- The service then tries to settle each loan and notifies the customer either to top up their wallet if the amount is insufficient or the loan has been successfully settled