# Module Summary

This module is part of ERP project that enables customers to pay scheme\* instalments through a web portal.

\*Scheme: A monthly instalment program that can be redeemed for a purchase on a later date at a physical store.

# Scope of project

You are expected to create a serverless function using Lambda that communicates with the database located on AWS RDS and to create a webpage that has

* login and sign up functionality,
* Fetch data from given database and display due instalment details
* Receive payment

The users who are signing up will be those who are already registered for schemes at physical store.

# Provided Resources

1. Database structure with sample data
2. Access to an AWS account to test run your code.

# Deliverables

1. **Serverless Function**: You’ll be required to create serverless function using AWS’s Lambda which interfaces with a MYSQL DB on AWS RDS and call HTTP requests.
2. **Webpage:** The webpage facilitate to
   1. Create new user with OTP verification and save the username and password after encryption.
   2. Accept login data.
   3. Show user data and scheme dues & details.
   4. Link payment gateway to receive payment.
   5. Save the payment data into the RDS server and call the payment-received function on Lambda
   6. Facility to change password by user has to be provided.

# Notes

1. Create serverless functions using AWS Lambda, connect to RDS and ensure CRUD operations.
2. Modify serverless functions to access and write data as required.
3. Make a function for HTTP request to a given SMS provider for OTP generation and validation
4. Make a function for HTTP request to a given SMS provider for payment received confirmation message
5. Create API triggers with authentication for the serverless functions.
6. Create a webpage for the end user with login functionality. Use the API triggers to implement the function in the webpage.
7. Link to suggested payment gateway into the webpage to accept payments. Store the payment data into the RDS server.

# AWS Login Details

Username : swaroop\_ajit

Password : nsx4life!

# **Database structure**

**Branchmast - Details of Branch stored**

|  |  |  |  |
| --- | --- | --- | --- |
| Column\_name | Type | Length | Column Desc |
| BRNCH\_KEY | float | 8 | Unique Identifier for Branch |
| BRNCH\_NAME | varchar | 200 | Name of the Base Branch |

**Chitgroup - Stores details of Scheme**

|  |  |  |  |
| --- | --- | --- | --- |
| Column\_name | Type | Length | Column Desc |
| CGrp\_Key | float | 8 | Unique Identifier for Scheme Group |
| CGrp\_Code | varchar | 40 | Code of the Scheme Group |
| CGrp\_Name | varchar | 70 | Name of the Scheme Group |
| Scheme\_Key | float | 8 | Type of the Scheme Group \* |
| Brnch\_key | float | 8 |  |

\*- Types of Scheme Group – 4 types of Scheme Group is available, where 1 & 2 is Fixed and 3,4 is Flexi. In fixed type the instalment amount will be fixed, whereas in Flexi instalment amount will not be fixed and customer can pay any amount

**Onlineuser – Store details of users who registers online**

|  |  |  |  |
| --- | --- | --- | --- |
| Column\_name | Type | Length | Column Desc |
| User\_key | Float | 10 | Unique identifier for User,PrimaryKey |
| User\_name | varchar | 20 | Online username |
| User\_password | varchar | 20 | Online password |
| Phone\_No | varchar | 50 |  |
| Brnch\_Key | float | 8 | Base Branch key based on the branch which the customer has physically registered |

Phone no saved here in ONLINEUSER has to be linked with phone no in CHITMAST and retrieve the customer scheme details & dues

**Chitmast - Stores details of Scheme Customer with type of scheme group customer has enrolled and no of instalments paid till date**

|  |  |  |  |
| --- | --- | --- | --- |
| Column\_name | Type | Length | Column Desc |
| Chit\_Key | float | 8 | Unique Identifier for Chitmast |
| CGrp\_Key | float | 8 | Key of the Scheme Group |
| Chit\_No | varchar | 30 | Member no of the customer |
| Cust\_Name | varchar | 70 | Name of the Customer |
| Phone\_No | varchar | 50 |  |
| EMail | varchar | 50 |  |
| DOJ | varchar | 10 | Date of joining of the customer |
| Inst\_No | float | 8 | Total no. of installments based on Scheme |
| Inst\_Amt | float | 8 | Installment amount to be paid (Fixed type) |
| Open\_No | float | 8 |  |
| Paid\_No | float | 8 | Total No.of installment paid by the customer |
| CCust\_key | Float | 8 | Unique identifier for customer |
| Brnch\_Key | float | 8 | Base Branch key based on the branch which the customer has physically registered |
|  |  |  |  |

**Mchitrcptmast - Details of payments received are stored here**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column\_name** | **Type** | **Length** | **Column Desc** |
| Serial\_No | float | 8 | Primary Key , Unique |
| Ref\_no | float | 8 | Last No +1 |
| Ref\_Date | varchar | 10 | Current date |
| INST PAID | FLOAT | 8 | No. of installments paid |
| Amount | float | 8 | Amount paid by customer |
| Chit\_key | float | 8 |  |
| Brnch\_key | float | 8 |  |
| Trx\_Refno | varchar | 100 | Online transaction ref. received from payment gateway |

# Notes about database structure

- Data and their relevant tables

* Customer already registered at physical store has to create username and password which has to be stored in CHITMAST by validating the phone number saved in has to be verified in CHITMAST. Username and password has to be encrypted using ascii key + 100
* The scheme details and instalment details like instalment number, instalment amount, and due date has to be fetched from CHITMAST using Chit\_key
* Scheme name(s) have to be fetched from CHITGROUP using CGrp\_Key
* Branch name have to be fetched from BRANCHMAST using Brnch\_key
* The payment details has to be saved in MCHITRCPTMAST
* Serial\_no has to be incremented by 1 for each record
* Ref\_no has to be incremented by 1 for each new record
* Instpaid is no. of instalments which will be 1

- Payment dues calculation

* In Fixed scheme type, the customer is allowed to pay only once in a month and fixed amount which is stored in Inst\_amt in CHITMAST. For example, if a customer joins for a scheme on
* 10th of November, he/she can pay next installment on or after 10th December only
* Due month has to be calculated by taking Month difference of Date of Joining and date of receipt. The Paid\_no column in CHITMAST has to be updated along with saving payment details in MCHITRCPTMAST
* Paid\_no column in CHITMAST stores the no of instalments paid by the customer
* In Flexi scheme type customer is allowed to pay ‘n’ number of installments.

- Http request URL will be provided later