

Project Title: Smart Customer Retention & Loyalty Management System

Phase 6: User Interface Development

1. Lightning App Builder (Record Page)

Steps

1. Setup → **Lightning App Builder** → New → Record Page.
2. Choose **Customer_Retention_Data__c** object.
3. Add **Record Detail** + custom LWC (created below).
4. Save → Activate.

2. Create LWC

Create a New Salesforce DX Project

1. Open **VS Code** → View → **Command Palette** (or Ctrl+Shift+P).
2. Type **SFDX: Create Project** → Enter.
3. Choose **Standard**.
4. Enter a project name, e.g., CustomerRetentionProject.
5. Choose a location on your computer → Click **Create**.

After this, VS Code will create a project folder with this structure:

Connect to Your Salesforce Org

1. Open **Command Palette** → **SFDX: Authorize an Org**.
2. Choose **Production or Sandbox**.
3. Log in to your Salesforce org via the browser.
4. After successful login, the CLI will link the org to your project.



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS HISTORY Filter Salesforce CLI
Starting SFDX: Authorize an Org
23:30:24.577 sf org:login:web --alias CustomerRetentionProject --instance-url https://login.salesforce.com --set-default
Successfully authorized harikanayudu@03515@gentForce.com with org ID 000gK000007IsUTAU
23:31:09.540 sf org:login:web --alias CustomerRetentionProject --instance-url https://login.salesforce.com --set-default
ended with exit code 0
```

1 Create the Apex Class First

1. In **VS Code** → **Explorer**, go to:
2. force-app/main/default/classes/
3. **Create new file:** RetentionController.cls

```

force-app > main > default > classes > RetentionController.cls ...
1  public with sharing class RetentionController {
2      @AuraEnabled
3      public static Boolean redeemPoints(Id customerId, Integer points){
4          if(customerId == null || points == null) throw new AuraHandledException('Invalid parameters');
5
6          Customer_Retention_Data__c c = [
7              SELECT Id, Points_Balance__c
8              FROM Customer_Retention_Data__c
9              WHERE Id = :customerId
10             FOR UPDATE
11         ];
12
13         if(c.Points_Balance__c == null || c.Points_Balance__c < points)
14             throw new AuraHandledException('Insufficient points');
15
16         c.Points_Balance__c -= points;
17         update c;
18         return true;
19     }
20 }
21

```

4. Create the **metadata file** next to it: RetentionController.cls-meta.xml:

```

force-app > main > default > classes > RetentionController.cls-meta.xml
1  <?xml version="1.0" encoding="UTF-8"?>
2  <ApexClass xmlns="http://soap.sforce.com/2006/04/metadata">
3      <apiVersion>60.0</apiVersion>
4      <status>Active</status>
5  </ApexClass>

```

2 Deploy the Apex Class

1. Right-click the RetentionController.cls file → **SFDX: Deploy Source to Org**.
2. Wait for “Deployed Source to Org” confirmation in the Output panel.

```

23:37:15.515 Starting SFDX: Deploy This Source to Org
== Deployed Source
STATE    FULL NAME        TYPE      PROJECT PATH
Created  RetentionController  ApexClass  force-app\main\default\classes\RetentionController.cls
Created  RetentionController  ApexClass  force-app\main\default\classes\RetentionController.cls-meta.xml
23:37:21.80 Ended SFDX: Deploy This Source to Org

```

3 Deploy the LWC

1. Now go back to your LWC folder:
2. force-app/main/default/lwc/customerRetentionCard/

```

<template>
    <lightning-card title="Customer Summary" icon-name="standard:contact">
        <div class="slds-p-around_medium">
            <p><strong>Name:</strong> {name}</p>
            <p><strong>Risk Score:</strong> {risk}</p>
            <p><strong>Loyalty Tier:</strong> {tier}</p>
            <div>
                <lightning-button label="Redeem 100 pts" onclick={handleRedeem}></lightning-button>
            </div>
        </lightning-card>
    </template>

```

The screenshot shows the VS Code interface with the following details:

- Explorer View:** Shows the project structure under "CUSTOMERRETENTIONPROJECT".
- Code Editor:** Displays two files:
 - customerRetentionCard.js:** A JavaScript file containing LWC component logic. It imports LightningElement, api, and wire from 'lwc'. It uses @api recordId; and @wire(getRecord, { recordId: '\$recordId', fields: [NAME_FIELD, RISK_FIELD, TIER_FIELD] }) to fetch record fields. It has getter methods for name, risk, and tier. It includes a handleRedeem() method that calls redeemPoints({ customerId: this.recordId, points: 100 }) and dispatches a refresh event.
 - RetentionController.cls:** A Apex class extending LightningElement. It has a constructor that takes @api recordId; and a handleRedeem() method that calls redeemPoints({ customerId: recordId, points: 100 }) and dispatches a refresh event.
- Terminal View:** Shows the command "CustomerRetentionProject" and the XML metadata file "customerRetentionCardjs-meta.xml".

3. Right-click the folder → SFDX: Deploy Source to Org

```
23:37:37.923 Starting SFDX: Deploy This Source to Org
==== Deployed Source
STATE FULL NAME TYPE PROJECT PATH
Created customerRetentionCard LightningComponentBundle force-app\main\default\lwc\customerRetentionCard\customerRetentionCard.html
Created customerRetentionCard LightningComponentBundle force-app\main\default\lwc\customerRetentionCard\customerRetentionCard.js
Created customerRetentionCard LightningComponentBundle force-app\main\default\lwc\customerRetentionCard\customerRetentionCard.js-meta.xml
23:37:41.726 Ended SFDX: Deploy This Source to Org
```

1. **Open Lightning App Builder** → Edit your Customer_Retention_Data__c record page.
2. **Drag your LWC (customerRetentionCard) from the Custom section onto the page.**
3. **Save & Activate** → choose who can see the page.

Lightning App Builder | Pages | Customer Retention Record Page

Components Fields

Customer Name: Bunney Owner: Nayudu Harika

Risk Score: 89 Loyalty Tier: Silver

Last Interaction Date: 9/9/2025 Churn Flag: ✓

Home Balance: \$1000 Customer Email: abc@gmail.com

Last Purchase Date: 8/3/2025 Total Purchases: \$7,000

Loyalty Status: Enrolled

Created By: Nayudu Harika Last Modified By: Nayudu Harika

Customer Summary

Name: Bunney Risk Score: 89 Loyalty Tier: Silver

Recomm. 100 pts

Analyze Activation... Save

Page

*Label: Customer Retention Record Page

*API Name: Customer_Retention_Record_Page

*Page Type: Record Page

Object: Customer Retention Data

Template: Grouped Header and One Re... Change

Description:

Enable page-level dynamic actions for the Salesforce mobile app

Get more on the AppExchange

The screenshot shows a custom record page for 'Customer Retention Record Page' in the Lightning App Builder. The page displays various customer details such as name, risk score, loyalty tier, and interaction history. A summary section at the bottom recommends 100 points. The page is configured with a grouped header and one record template. The API name is set to 'Customer_Retention_Record_Page'.