Salesforce CRM Enhancement for Sales Pipeline Management

Project Overview

This project aims to improve sales pipeline visibility and automate key sales processes using Salesforce. It combines Salesforce Admin and Development features to enhance CRM efficiency.

---

1. Project Objective

The objective is to streamline the sales process by:

- Automating follow-up task creation.
- Ensuring data accuracy through validation rules.
- Providing insights through dashboards and reports.
- Utilizing Apex triggers for data consistency.

---

2. Features and Implementation

A. Custom Objects & Fields

- **Custom Object:** `Deal`
  - Fields:
    - `Deal Name` (Text)
    - `Deal Value` (Currency)
    - `Close Date` (Date)
    - `Stage` (Picklist: Prospecting, Negotiation, Closed Won, Closed Lost)
- **Custom Object:** `Sales Meeting`
  - Fields:
    - `Meeting Date` (Date)
    - `Meeting Notes` (Long Text Area)
    - `Follow-up Required` (Checkbox)
- Lookup relationship established between `Deal` and `Account` for data linkage.

B. Flow Automation

Flow Name: `Auto Follow-up Task Creation`

- **Trigger:** When `Deal` Stage changes to "Negotiation."
- **Actions:**
  - Create a task with:
    - Subject: `Follow-up on Negotiation Stage`
    - Assigned To: `Account Owner`
    - Due Date: 3 days from trigger date.
- **Outcome:** Ensures no deal in the negotiation phase is left unattended.

C. Validation Rules

Rule Name: `Ensure Critical Deal Data`

- **Condition:**

```
AND(
  ISBLANK( Deal_Value__c ),
  ISPICKVAL(Stage__c, "Closed Won")
)
```

- **Error Message:** "Deal Value is required before marking the deal as Closed Won."
- **Outcome:** Ensures critical data is entered for accurate reporting.

D. Reports & Dashboards

Dashboard Name: `Sales Pipeline Overview`

- **Components:**
  - Bar Chart: `Deals by Stage`
  - Pie Chart: `Upcoming Follow-ups by Owner`
  - Metric: `Total Deal Value`
- **Outcome:** Provides managers with a comprehensive view of sales performance.

E. Apex Trigger (Development Part)

Trigger Name: `UpdateAccountTotalValue`

```
trigger UpdateAccountTotalValue on Deal__c (after insert, after update) {
    Map<Id, Account> accountMap = new Map<Id, Account>();

    for (Deal__c deal : Trigger.new) {
        if (deal.Account__c != null) {
            accountMap.put(deal.Account__c, new Account(
                Id = deal.Account__c,
                Total_Deal_Value__c = deal.Deal_Value__c // Auto-update logic
            ));
        }
    }

    if (!accountMap.isEmpty()) {
        update accountMap.values();
    }
}
```

- **Outcome:** Ensures `Total Deal Value` on the Account reflects the sum of associated Deals.

F. Profiles & Permission Sets

- Configured **Profiles** for `Sales Manager`, `Sales Executive`, and `Support Staff` with appropriate object-level permissions.
- Created a **Permission Set** called `Deal Access` to grant selective CRUD access to `Deal` records.

---

3. Technologies Used

- Salesforce Admin Tools (Object Manager, Validation Rules, Flows)
- Salesforce Lightning for UI customization
- Apex for custom logic
- SOQL for optimized data retrieval

---

4. Outcome and Benefits

- Improved sales efficiency by reducing manual follow-ups.
- Enhanced data integrity through validation rules.
- Provided actionable insights via dashboards for better decision-making.
- Ensured data consistency between `Deal` and `Account` using Apex triggers.

---

5. Future Enhancements

- Add email alerts for follow-up reminders.
- Implement scheduled batch jobs for weekly sales performance summaries.
- Integrate third-party email services for improved customer communication.

---

6. Conclusion

This project demonstrates Salesforce Admin and Development skills by combining automation, security, and data visibility. It effectively improves sales pipeline tracking and enhances CRM functionality, making it a suitable project for beginner-level Salesforce developers.