Government Schemes Management – Salesforce CRM Project

Phase 1: Problem Understanding & Industry Analysis

Goal: Understand why this project is needed and how it will help.

1. **Requirement Gathering**
    - Talk to stakeholders (Government Officers, Scheme Auditors, Citizens who apply for schemes).
    - Example requirements:
        - Citizens should be able to apply for schemes online.
        - Officers should approve/reject applications.
        - Auditors should verify reports and track misuse.
        - Generate scheme utilization reports.
2. **Stakeholder Analysis**
    - **Admin** → You, managing Salesforce setup and data security.
    - **Officer** → Reviews and approves citizen applications.
    - **Auditor** → Monitors schemes, checks reports, prevents fraud.
    - **Citizen** → Portal user, applies for schemes.
3. **Business Process Mapping**
    - **Flow:**
      Citizen submits scheme application → Officer reviews → Approval/Rejection → Auditor audits records → Reports generated.
4. **Industry-specific Use Case Analysis**
    - In government schemes, **transparency and accountability** are important.
    - Need to:
        - Avoid duplicate applications.
        - Track fund allocation and usage.
        - Provide audit trails.
5. **AppExchange Exploration**

- Look for “Government Solutions” or “Public Sector CRM” apps.
- Decide to build custom **Government Schemes Management CRM** for learning.

Phase 2: Org Setup & Configuration

Goal: Prepare Salesforce environment for the project.

1. **Salesforce Edition**
    - Use **Developer Edition Org** (free org).
2. **Company Profile Setup**
    - Company Name: “Govt Schemes Department.”
    - Time zone, fiscal year = April–March (Indian government cycle).
    - Currency = INR.
3. **Business Hours & Holidays**
    - Working Hours: 10am–5pm.
    - Add national holidays → no approvals on these days.
4. **User Setup & Licenses**
    - Create users: Officer, Auditor, Citizen. Assign **Salesforce License** for Officer/Auditor, **Portal/Community License** for Citizen.
5. **Profiles**
    - **Officer Profile**: Can review and approve applications.
    - **Auditor Profile**: Can view all records but cannot modify citizen data.
    - **Citizen Profile**: Can submit and track their own applications.
6. **Roles**
    - Role Hierarchy:
        - **Admin (top)**
            - Officers (under Admin)
            - Citizens (external portal users)
7. **Permission Sets**
    - Example: Give Auditors extra access to “Reports” via Permission Sets.

8. **OWD (Org-Wide Defaults)**
    - Scheme Applications: **Private** (only owner and officer can see).
    - Schemes: **Public Read Only** (everyone can see available schemes).
9. **Sharing Rules**
    - Allow Citizens to see their own records.
    - Allow Auditors to see all scheme applications.
10. **Login Access Policies**

- Restrict Officers' login hours (10am–5pm).
- Restrict Citizens to India IP ranges.

Phase 3: Data Modeling & Relationships

Goal: Design the database structure.

1. **Standard & Custom Objects**
    - Standard: **Contact (Citizen)**.
    - Custom:
        - **Scheme** → Master Data (Scheme Name, Eligibility, Budget).
        - **Scheme Application** → Submitted by Citizens.
2. **Fields**
    - **Scheme**: Name, Budget, Start Date, End Date, Eligibility Criteria.
    - **Scheme Application**: Applicant, Scheme, Application Date, Status, Amount Approved.
3. **Record Types**
    - Application → "Student Scheme," "Farmer Scheme," "Healthcare Scheme."
4. **Page Layouts**
    - Officer sees all application details with approval buttons.
    - Citizen sees only their submission details.
5. **Compact Layouts**
    - On mobile: Application shows Applicant Name, Scheme Name, Status.
6. **Schema Builder**

- Visualize **Scheme ↔ Application ↔ Citizen** relationships.

7. **Lookup vs Master-Detail**
    - Scheme ↔ Application → Master-Detail (application depends on scheme).
    - Application ↔ Citizen → Lookup (citizen can apply for multiple schemes).
8. **Junction Objects**
    - If one application had multiple schemes → need a junction object.

Phase 4: Process Automation (Admin)

Goal: Automate scheme processes.

1. **Validation Rules**
    - Example: Application Date cannot be before Scheme Start Date.
2. **Approval Process**
    - Officer approval required for any scheme application.
    - Amount > ₹50,000 → send to higher-level officer.
3. **Flow Builder**
    - Auto-calculate approved amount.
    - Auto-change status after approval.
4. **Email Alerts**
    - Citizen gets email when application is Approved/Rejected.
5. **Tasks**
    - Officer assigned task to review pending applications.
6. **Custom Notifications**
    - In-app notification to Citizen: “Your scheme has been approved.”

Phase 5: Apex Programming (Developer)

Goal: Add custom business logic.

1. **Apex Triggers**
    - Prevent duplicate applications from the same citizen for the same scheme.

2. **SOQL Queries**
    - Example: Find all pending applications for a specific scheme.
3. **Batch Apex**
    - Monthly job: Close expired schemes automatically.
4. **Scheduled Apex**
    - Weekly: Email Auditor with "Top 10 funded schemes."
5. **Exception Handling**
    - Prevent errors when budgets exceed limits.
6. **Test Classes**
    - Ensure triggers, flows, and approvals work correctly.

Phase 6: User Interface Development

Goal: Build a user-friendly app.

1. **Lightning App Builder**
    - Create "Govt Schemes Management App."
2. **Record Pages**
    - Scheme page shows applications list.
    - Application page shows Citizen details.
3. **Tabs**
    - Schemes, Applications, Citizens.
4. **Dashboards on Home Page**
    - Show "Applications Pending," "Funds Utilized."
5. **LWC (Lightning Web Components)**
    - Build Citizen Portal form: Apply for scheme online.
    - Officer Dashboard: Approve/Reject applications.

Phase 7: Integration & External Access

Goal: Connect with other systems.

1. **Named Credentials** → Store API keys for Aadhaar/Identity verification.
2. **REST API Callouts** → Validate citizen details with government DB.
3. **Platform Events** → Notify when scheme budget is exhausted.
4. **Salesforce Connect** → Link with Finance/ERP system to track funds.

Phase 8: Data Management & Deployment

Goal: Manage and migrate data.

1. **Data Import Wizard** → Upload demo Schemes.
2. **Data Loader** → Import citizen applications.
3. **Duplicate Rules** → Prevent duplicate scheme records.
4. **Data Export & Backup** → Weekly backups.
5. **Change Sets** → Move from Sandbox → Production.

Phase 9: Reporting, Dashboards & Security Review

Goal: Monitor schemes and ensure security.

1. **Reports**
    - Applications by Scheme.
    - Funds Utilized per Scheme.
    - Applications Approved vs Rejected.
2. **Dashboards**
    - Officer Dashboard → Pending Approvals.
    - Auditor Dashboard → Scheme-wise Fund Utilization.
3. **Security**
    - Field-level security → Hide citizen's Aadhaar number from Officers.

- IP Restrictions → Officers login only from government office.
- Audit Trail → Track changes to approvals.

Phase 10: Final Presentation & Demo Day

Goal: Deliver project like a real CRM solution.

1. **Pitch Presentation**
    - Problem → Solution → Transparency in government schemes.
2. **Demo Walkthrough**
    - Citizen applies → Officer approves → Auditor reviews → Report generated.
3. **Handoff Documentation**
    - User Guide for Citizens, Officers, Auditors.
    - Technical Documentation for Admins.
4. **Portfolio Showcase**
    - Share LinkedIn post: *"Built Govt Schemes Management CRM on Salesforce with full automation, approvals, and dashboards."*