📘 Customer Complaint CRM Project – Full Explanation

Phase 1: Problem Understanding & Industry Analysis

👉 **Goal: Understand what we're building and why.**

1. **Requirement Gathering**
    - Capture customer complaints from multiple channels (calls, email, web form).
    - Track complaint type (billing issue, service issue, product defect, etc.).
    - Assign complaints to agents automatically.
    - Monitor resolution SLA and escalations.
    - Generate performance reports (e.g., average resolution time).
2. **Stakeholder Analysis**
    - **Admin** – configures system, manages users.
    - **Customer Support Agents** – handle and resolve complaints.
    - **Manager** – monitors performance, escalates issues.
    - **Customer** – submits complaint, views status.
3. **Business Process Mapping**
   Flow:
    - Customer submits complaint → Complaint logged → Auto assignment to Agent → Resolution attempt → Closure OR Escalation → Notification sent to customer.
4. **Industry-specific Use Case Analysis**
    - High complaint volume → need automation to assign and track.

- SLA compliance is key (e.g., respond within 24 hours).
- Customer satisfaction improves if they can track complaint status.

5. **AppExchange Exploration**
   - There are “Case Management” apps, but we’ll build our own lightweight version for learning.

Phase 2: Org Setup & Configuration

👉 **Goal: Prepare Salesforce environment.**

1. Use **Developer Edition Org**.
2. **Company Settings** → set local time zone, currency (INR/USD).
3. **Business Hours & Holidays** → define support hours (9 AM – 6 PM).
4. **Fiscal Year** → standard (Jan–Dec).
5. **User Setup & Licenses** → create users: Agent, Manager, Admin.
6. **Profiles**
   - Agent: Can create/update complaints.
   - Manager: Full access.
7. **Roles**
   - Manager → Agent hierarchy.
8. **Permission Sets**
   - Extra permissions (e.g., Reports) via permission sets.
9. **Org-Wide Defaults (OWD)**
   - Complaint: Private.

    ○ Customer (Contact): Controlled by Parent.

10. **Sharing Rules**

- Share escalated complaints with Manager.

11. **Login Access Policies** → restrict login hours for agents.

Phase 3: Data Modeling & Relationships

👉 **Goal: Build data structure.**

1. **Objects**
    - **Contact (Standard)** → Customers.
    - **Complaint (Custom)** → Core object.
2. **Complaint Fields**
    - Complaint Number (Auto Number).
    - Subject (Text).
    - Description (Long Text).
    - Complaint Type (Picklist: Billing, Service, Product).
    - Status (Picklist: New, In Progress, Resolved, Escalated, Closed).
    - Priority (High, Medium, Low).
    - Assigned Agent (Lookup → User).
    - Resolution Notes (Long Text).
    - SLA Due Date (Date/Time).
3. **Relationships**

   - Complaint → Contact (Lookup).
   - Complaint → User (Agent).
4. **Layouts & Compact Layouts**
   - Complaint Layout: Show Subject, Type, Status, Assigned Agent.
   - Contact Layout: Show related Complaints.

Phase 4: Process Automation (Admin)

👉 **Goal: Automate tasks.**

1. **Validation Rules**
   - Ensure Complaint must have Type before saving.
   - Prevent closing complaint without Resolution Notes.
2. **Flow Builder**
   - Auto-assign complaints to available Agent (round-robin).
   - Calculate SLA Due Date = CreatedDate + 24 hours.
   - Auto-update Status = “Escalated” if not resolved within SLA.
3. **Approval Process**
   - Escalated complaints → Manager approval before closure.
4. **Email Alerts**
   - Notify Customer when complaint is created, updated, or resolved.

Phase 5: Apex Programming (Developer)

👉 **Goal: Add advanced logic.**

1. **Trigger** on Complaint
   - Prevent duplicate open complaints for same customer & same type.
2. **SOQL Query**
   - Find unresolved complaints for reporting.
3. **Batch Apex**
   - Night job: auto-close complaints older than 90 days.
4. **Scheduled Apex**
   - Send daily SLA breach report to Manager.
5. **Exception Handling**
   - Handle assignment failures.
6. **Test Classes**
   - Verify triggers, batch jobs, and flows.

Phase 6: User Interface Development

👉 **Goal: Build user-friendly screens.**

1. **Lightning App** → “Customer Complaint CRM.”
2. **Tabs** → Complaints, Contacts, Reports.
3. **Page Layouts** → Complaint page shows key info + related Contact.
4. **Home Page** → Dashboard of open complaints by status.
5. **Utility Bar** → Quick “New Complaint” button.

6. **LWC (optional)** → Complaint submission form with file upload.

Phase 7: Integration & External Access

👉 **Goal: Connect with other systems.**

1. **Web-to-Case (or Web-to-Complaint)** → Allow complaint submission from website.
2. **Email-to-Case** → Auto-create complaint when customer emails support.
3. **REST API** → Expose complaint creation for mobile app.
4. **Platform Events** → Notify external system if complaint is escalated.

Phase 8: Data Management & Deployment

👉 **Goal: Manage data properly.**

1. **Data Import Wizard** → Import demo complaints.
2. **Data Loader** → Bulk complaint migration.
3. **Duplicate Rules** → Prevent duplicate customer entries.
4. **Data Export** → Weekly backups.
5. **Change Sets** → Deploy to Production.

Phase 9: Reporting, Dashboards & Security

👉 **Goal: Provide insights & protect data.**

1. **Reports**

- Average resolution time.
- Complaints by type & priority.
- Agent performance report.

2. **Dashboards**
   - Manager Dashboard: SLA compliance, escalations.
   - Agent Dashboard: My complaints.
3. **Security**
   - Complaints: Private OWD.
   - Sharing Rules: Manager access.
   - Field-level security: hide SLA Due Date from customers.

Phase 10: Final Presentation & Demo

👉 **Goal: Wrap project for delivery.**

1. **Pitch Presentation** → Problem → Solution → Benefits.
2. **Demo** → Submit complaint → Auto assignment → SLA escalation → Resolution.
3. **Documentation** → System design + user guide.
4. **Showcase** → Publish project on LinkedIn/Portfolio.

   -