Smart Farming CRM

Phase 1: Problem Understanding & Industry Analysis

1. Requirement Gathering

- Collect key requirements from stakeholders (farmers, buyers like distributors and retailers, logistics partners, admin/management).
- Example requirements:
    - Farmers must list crops with details (type, quantity, quality, price).
    - Buyers should browse crops, specify requirements, place orders.
    - Orders should go through farmer approval and digital contracts.
    - Payment tracking and automated reminders needed.
    - Logistics coordination for crop delivery and shipment tracking.
    - Ratings and feedback system for crop quality and transactions.

2. Stakeholder Analysis

- **Farmers:** Upload/manage crop listings, confirm orders, review buyer requirements.
- **Buyers (Distributors/Retailers):** Search crops, place requests/orders, provide feedback.
- **Logistics Partners:** Manage shipments, update delivery status.
- **Admins/Managers:** Oversee workflows, approvals, reporting, security.
- **Support Team:** Handle disputes, assist users.

3. Business Process Mapping

- Draw a flow diagram:
  Farmer lists crops → Buyer browses and submits requirements → Farmer reviews and confirms → Buyer places order → Contract signing → Payment processing → Logistics shipping → Delivery confirmation → Feedback & ratings.

4. Industry-specific Use Case Analysis

- Farmers often lack direct access to markets; need transparent platforms for fair pricing.
- Crop quality and payment delays are key issues.

- Logistics complexities require coordination with third parties.
- Digital contracts and automated payments streamline trust and transactions.
- CRM must support multiple user profiles and secure data sharing.

5. AppExchange Exploration

- Research existing agriculture or farm management apps for ideas.
- Identify gaps to customize solution per project goals.
- Consider apps supporting digital contracts, payment automation, and supply chain tracking.

Phase 2: Org Setup & Configuration

- Setup Salesforce Developer Org.
- Configure company info, business hours according to harvest and delivery seasons.
- Define roles for Farmers, Buyers, Logistics, Admins.
- Setup sharing rules and permissions.

Phase 3: Data Modeling & Relationships

- Custom objects: Farmer, Crop, Buyer Requirement, Order, Payment, Shipment, Feedback.
- Define fields: Crop type, quantity, price, order status, payment status.
- Use record types for crop categories or buyer types.

Phase 4: Process Automation (Admin)

- Validation rules (e.g., order quantity ≤ crop quantity).
- Flows for order approval, payment reminders.
- Email alerts for new crop listings, order status.
- Approval processes for contracts and large orders.

Phase 5: Apex Programming (Developer)

- Triggers for updating crop availability post-order.

- Batch Apex for payment reconciliation.
- Queueable Apex for async notifications.
- Test classes for coverage.

Phase 6: User Interface Development

- Lightning App with tabs for Crops, Orders, Payments.
- LWCs for crop catalog, order forms, feedback submission.
- Utility bar for alerts.

Phase 7: Integration & External Access

- Named credentials and callouts for logistics and payment gateway APIs.
- Platform events for shipment status updates.

Phase 8: Data Management & Deployment

- Data import for demo farmers and crops.
- Duplicate rules to prevent redundant records.
- Change sets and SFDX for deployment.

Phase 9: Reporting, Dashboards & Security Review

- Sales dashboards by crop type and region.
- Payment aging reports.
- Role-based sharing and field security.

Phase 10: Final Presentation & Demo Day

- Prepare pitch linking farmer empowerment and supply chain efficiency.
- Demo user journeys: crop listing, order creation, shipment tracking, payment.
- Compile technical and user manuals.
- Showcase on portfolio or LinkedIn.