CRM Application for Jewel Management – Developer Documentation

Project Overview

The Jewel Inventory System is a real-time CRM application built using Salesforce, designed to efficiently manage the inventory, sales, and customer relationships of a jewelry store or manufacturer.
This comprehensive solution provides end-to-end support for item tracking, sales transactions, reporting, and customer engagement — all within the Salesforce ecosystem.

Features and Functionalities

• Inventory Management
- Track jewellery items by type, weight, purity, and availability.
- Manage stock levels across different categories (Gold, Diamond, Silver, etc.).
- Enable automatic stock updates on sales or restocking.

• Sales and Customer Management
- Create, manage, and view customer records.
- Record and monitor sales transactions.
- Enable automated communication with customers via email templates and alerts.

• CRM Capabilities
- Customer engagement tracking.
- Integrated dashboards for sales and service reps.
- Scheduled follow-ups via Salesforce Flows and Alerts.

What You'll Learn

• Real-Time Salesforce Project Development
• Data Modelling: Custom objects like Jewel_Item, Customer, Purchase, and Supplier.
• Application Setup: Creating and configuring a custom Salesforce app.
• User Interface Customization: Lightning App Builder, record pages.
• Object & Relationship: Lookup and Master-Detail relationships.
• Formula Fields and Validation Rules
• Field Dependencies: Dynamic picklists.
• Record Types: For retail vs wholesale, and different jewellery types.
• Cross Object Formula Fields: Customer name in purchase, auto-calculated stock.

• Conditional Formatting: Highlight high-value orders, color-coded statuses.
• Automation with Flows: Auto-create follow-ups, deduct stock.
• Email Alerts and Templates
• Reports & Dashboards: Sales trends, top-selling items, customer history.

Security and Access Control

• Profile-based permissions for Admin, Sales Rep, Inventory Manager.
• Record-level security using Sharing Rules and Role Hierarchy.

Deployment and Testing

• Tested in Salesforce Developer Org.
• Validated all automation and formula logic through sample test records.
• UAT performed using different record types and user roles.

Future Enhancements

• Integration with POS systems.
• SMS alerts for order confirmation.
• Mobile-responsive Lightning components for on-the-go access.

Developer Role & Contribution

As a developer, your responsibilities include:
• Designing the object schema and data model.
• Building Flows and automation logic.
• Creating formula fields and validation rules.
• Developing dashboards and reports.
• Customizing user interface and record layouts.
• Testing and debugging throughout the lifecycle.