# **CRM APPLICATION FOR JEWEL MANAGEMENT**

**College Name:Thavathiru Santhalinga Adigalar Arts Science & Tamil College**

**College Code:Bru20**

**TEAM ID: NM2025TMID48755**

**TEAM MEMBERS: 4**

**Team LeaderName: YAMUNA THULASI S**

**Email: yamunathulasi47@gmail.com**

**Team Member1: SUNMADHI K**

**Email: sunmasunmadhi@gmail.com**

**Team Member: PAVITHRA . E**

**Email: eshwaranpavithra@gmail.com**

**Team Member: LOGESWARAN V**

**Email: logeshwaranlogesh1502@gmail.com**

**1. INTRODUCTION**

**1.1 Project Overview**

The Jewel Management CRM is a Salesforce-based application designed to streamline the processes of managing jewelry business operations. It handles customer management, item tracking, order processing, and billing with automation features such as flows, triggers, validation rules, reports, and dashboards. The system provides jewelers with an efficient way to store data, track sales, and generate insights while ensuring secure role-based access.

A diagram of a jewel management

AI-generated content may be incorrect.

**1.2 Purpose**

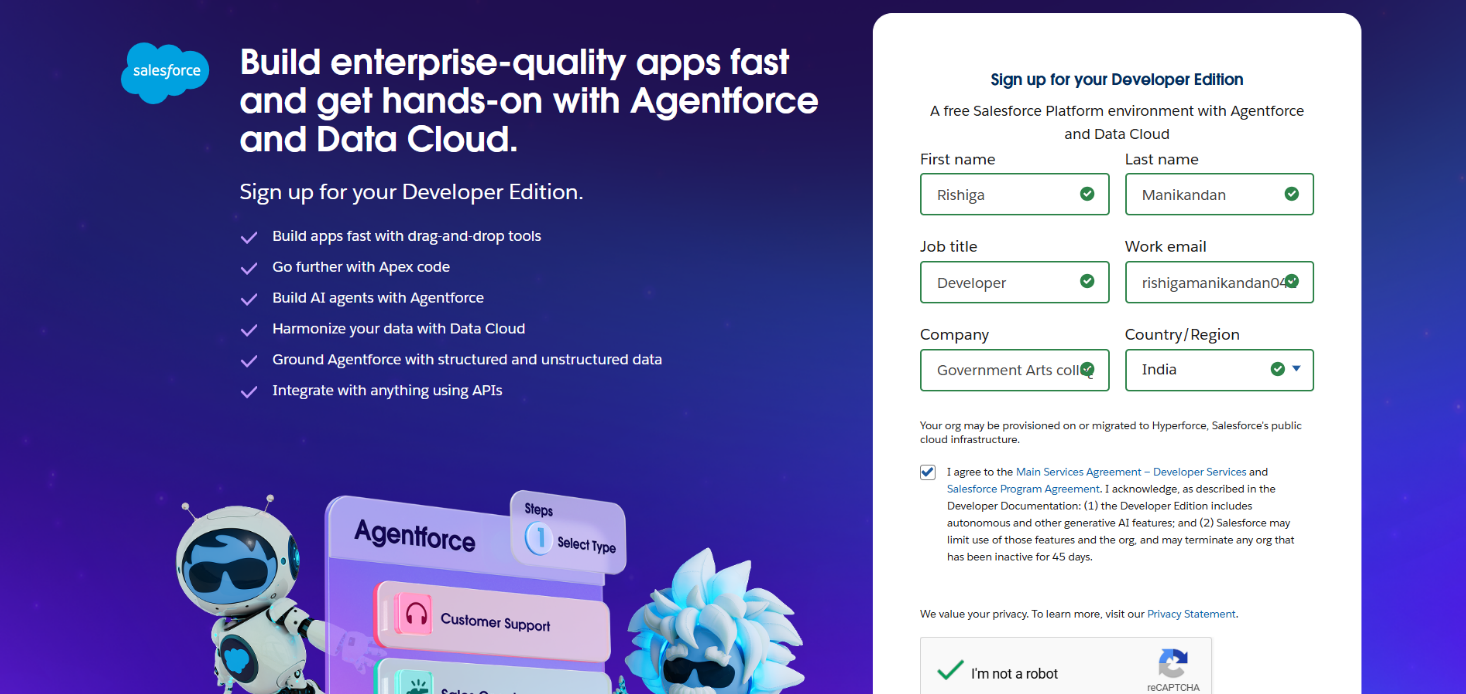
The main objective of the project is to enable jewelry businesses to efficiently manage customers, inventory, and transactions within Salesforce. It reduces manual errors, improves accuracy, and provides automation for day-to-day processes. With dashboards and reports, the CRM also helps in making data-driven business decisions and ensures better customer satisfaction.

**2. Development Phase**

**2.1 Salesforce Setup**

Created Developer Account and activated it.

By using this URL -  <https://developer.salesforce.com/signup>

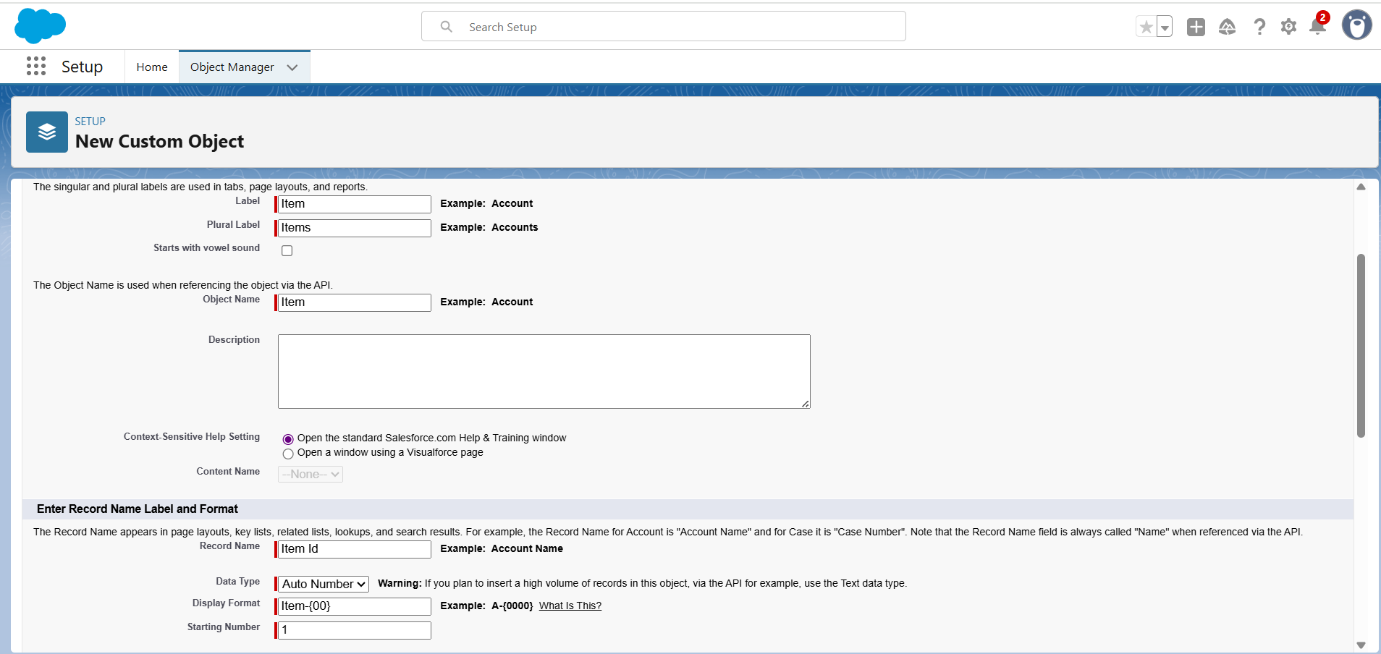


**2.2 Objects & Tabs**

Created custom objects (Jewel Customer, Item, Order, Billing).

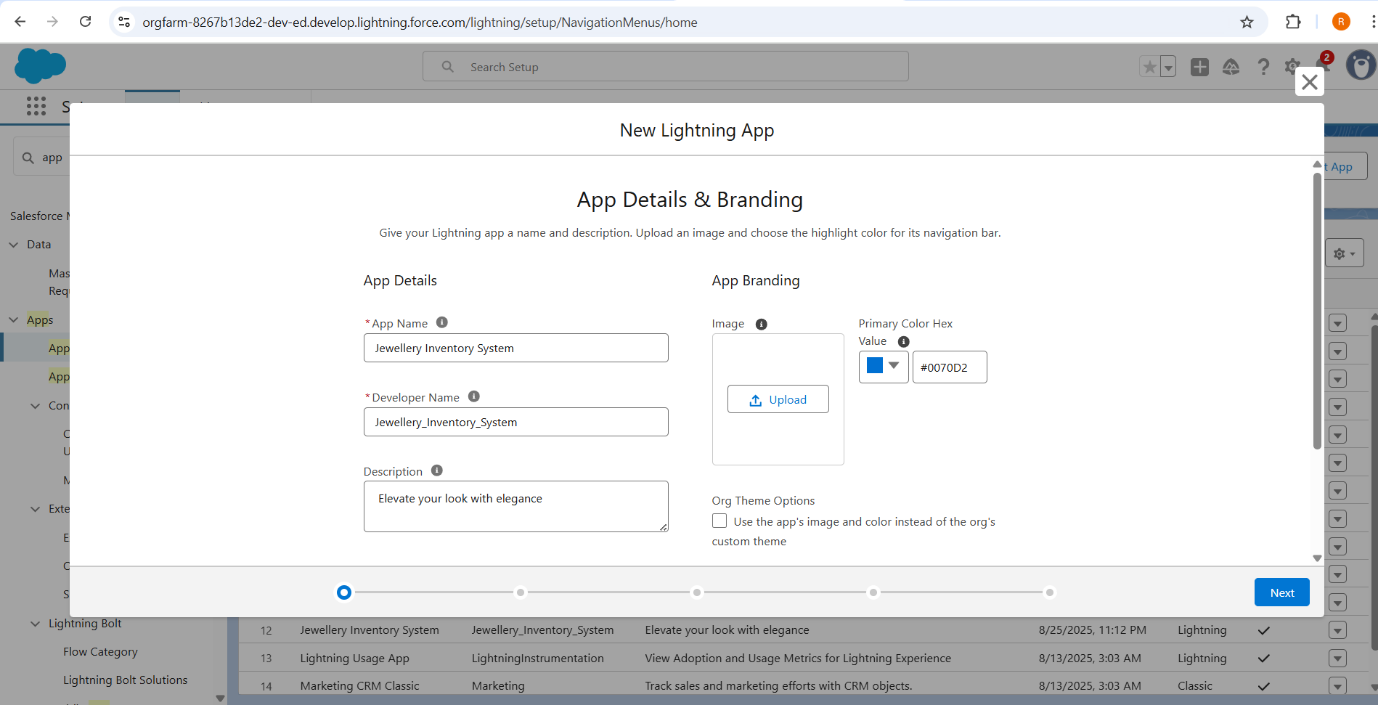
Created custom tabs for each object. A screenshot of a computer

AI-generated content may be incorrect.



**2.3 Lightning App**

Created Jewel Management Lightning App with all required tabs.

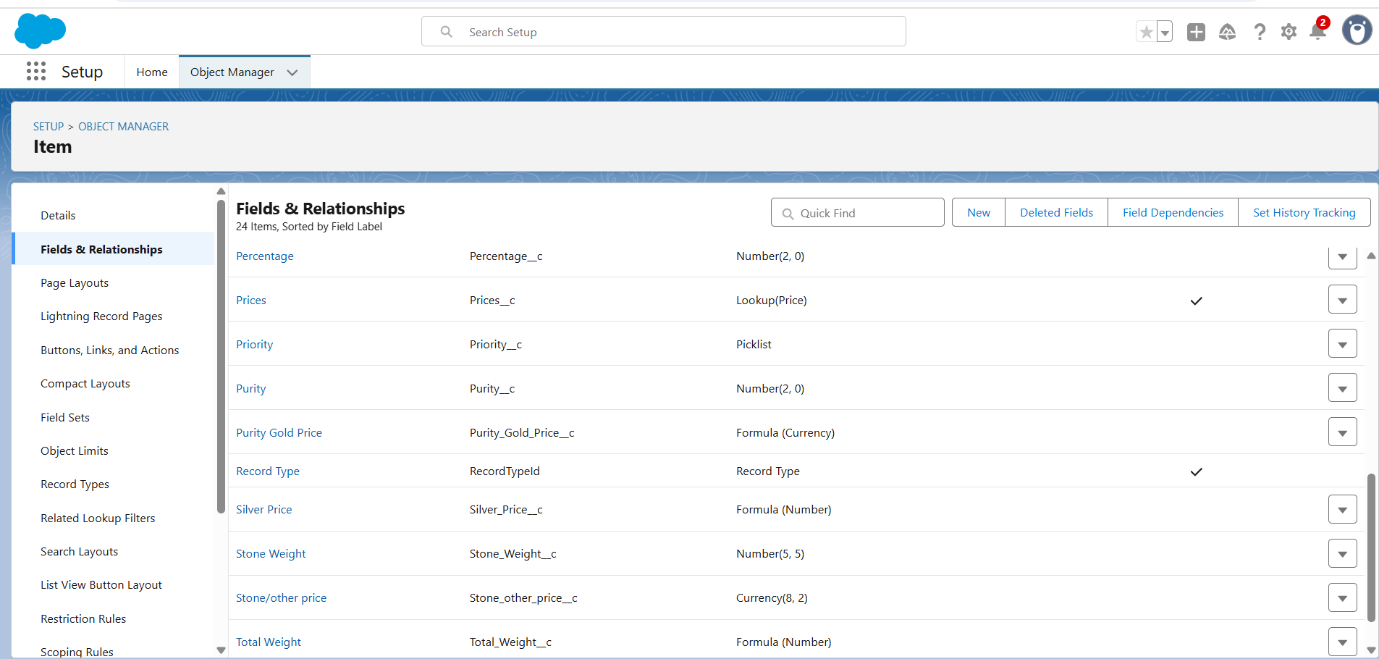


**2.4 Fields & Relationships**

Added fields (Phone, Email, Purity, Price, Picklist, Formula fields, etc.).

Created Lookup and Master-Detail relationships.

A screenshot of a computer

AI-generated content may be incorrect.

**2.5 Schema Builder**

Visualized object relationships and dependencies.

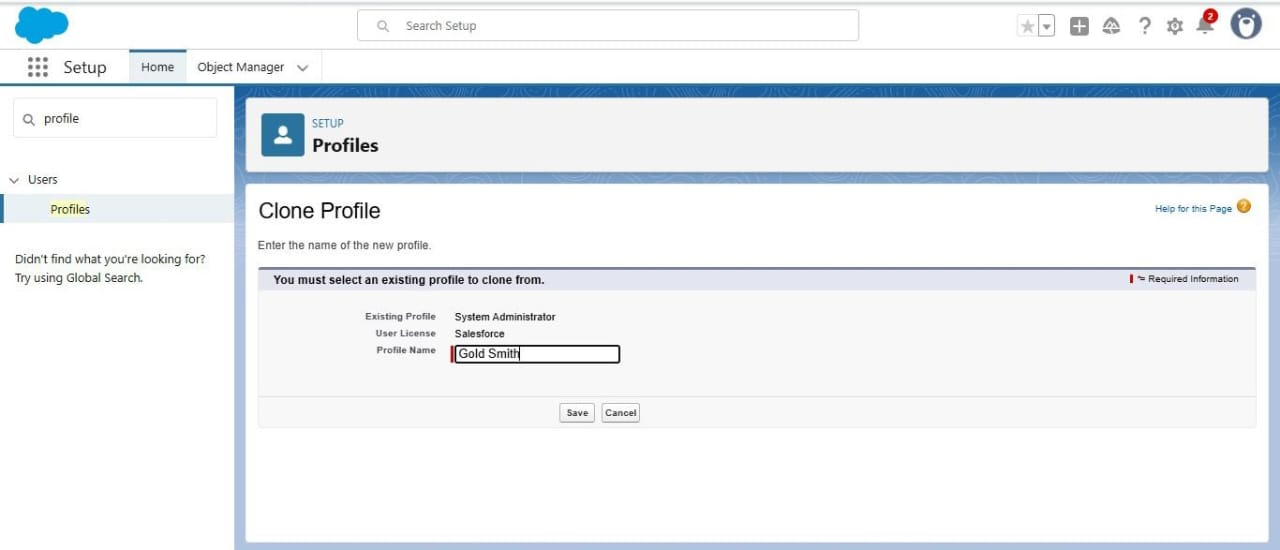
A screenshot of a computer

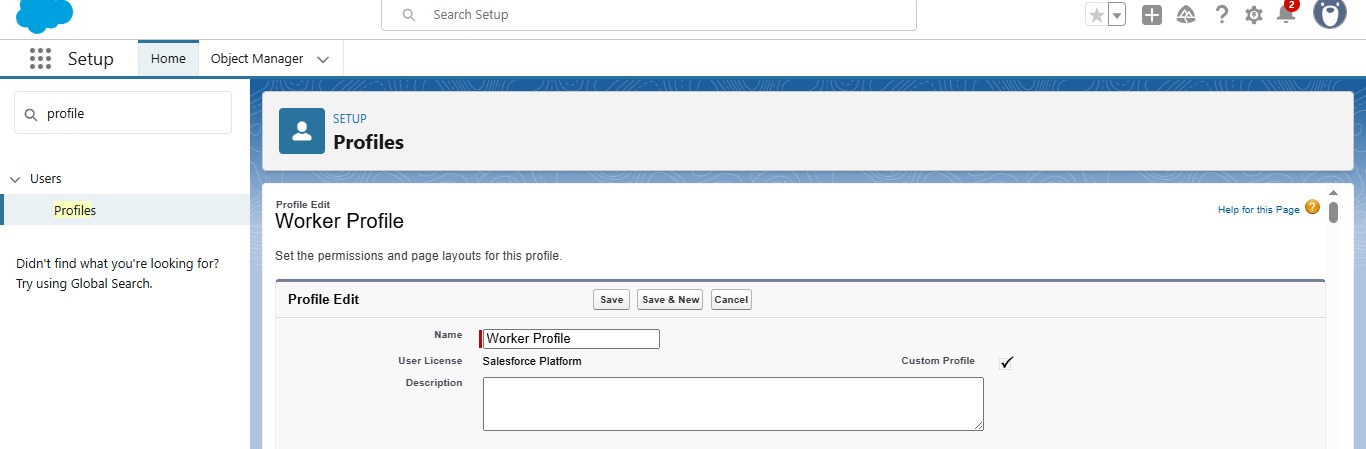
AI-generated content may be incorrect.

**2.6 Profiles & Roles**

Created Goldsmith and Worker profiles.

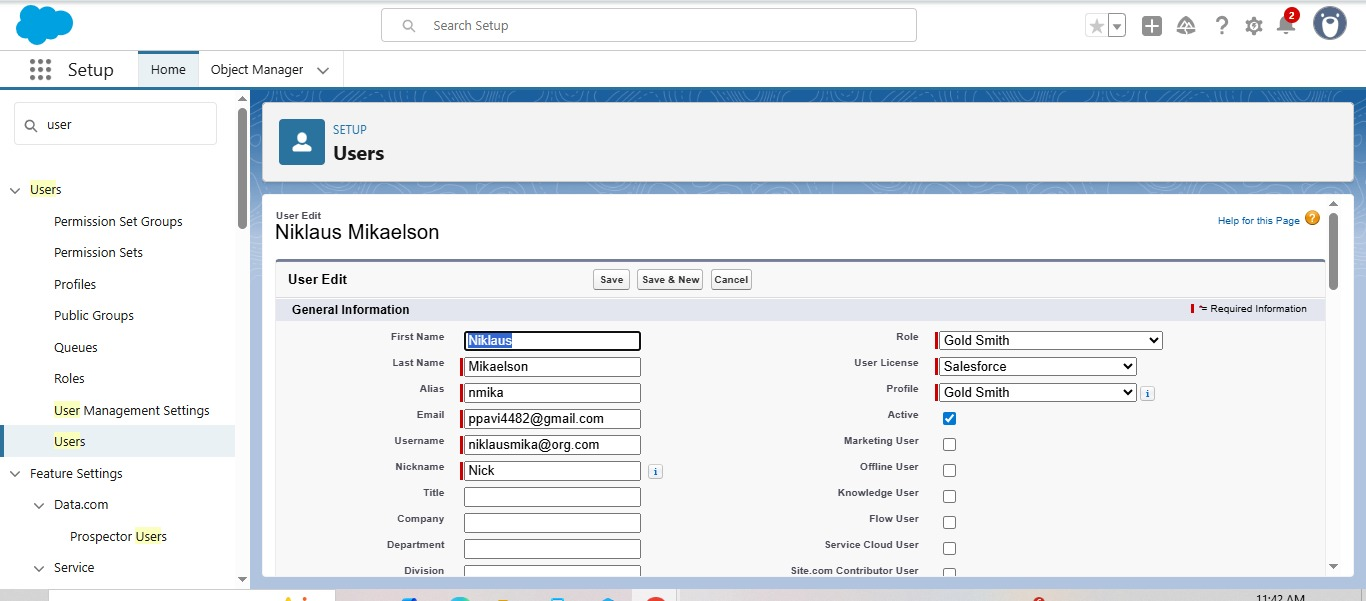
Defined role hierarchy (Worker reports to Goldsmith).





**2.7 Users**

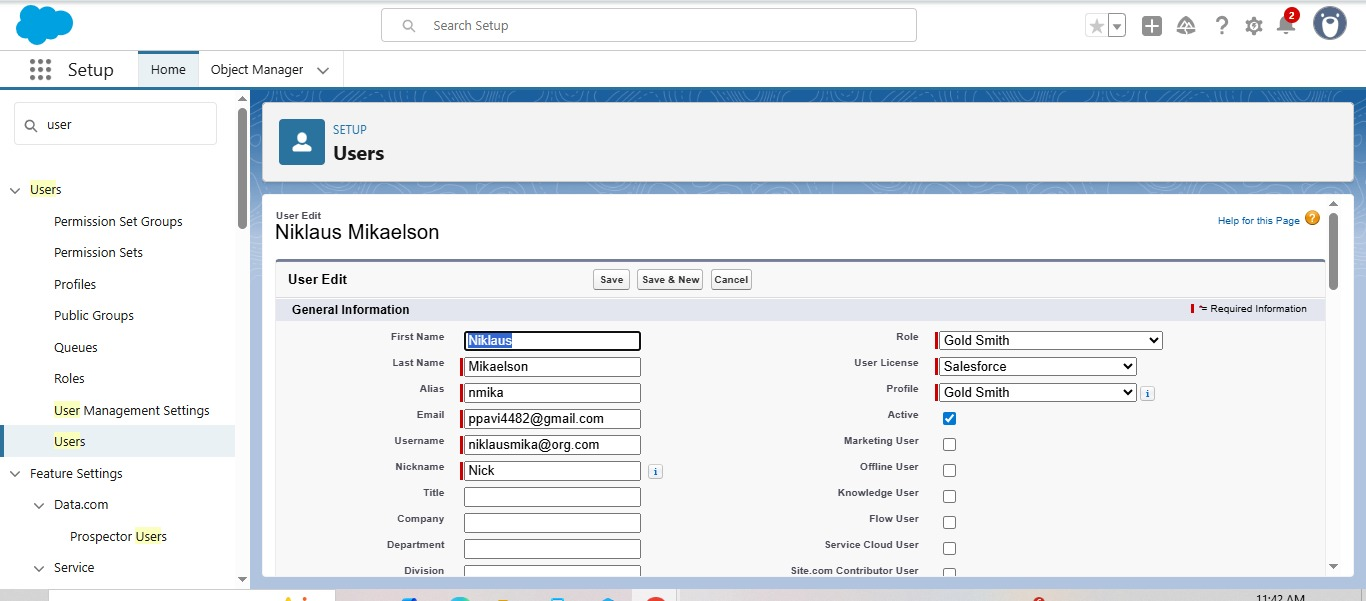
Created users and assigned profiles/roles.



**2.8 Page Layouts & Record Types**

Designed Gold & Silver layouts.

Created record types (Gold, Silver).

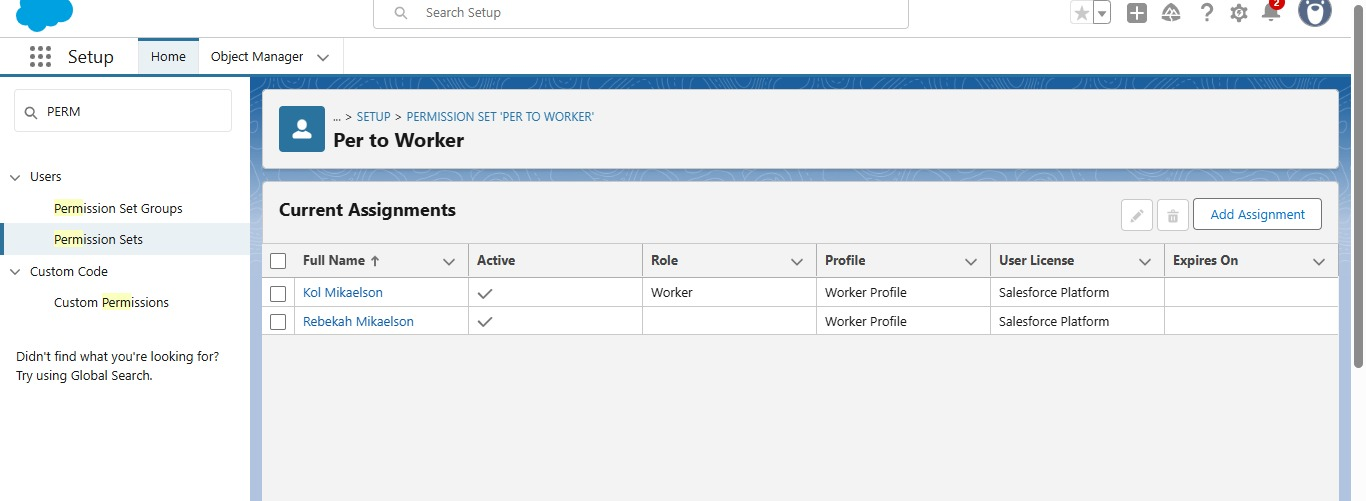


A screenshot of a computer

AI-generated content may be incorrect.

**2.9 Permission Sets**

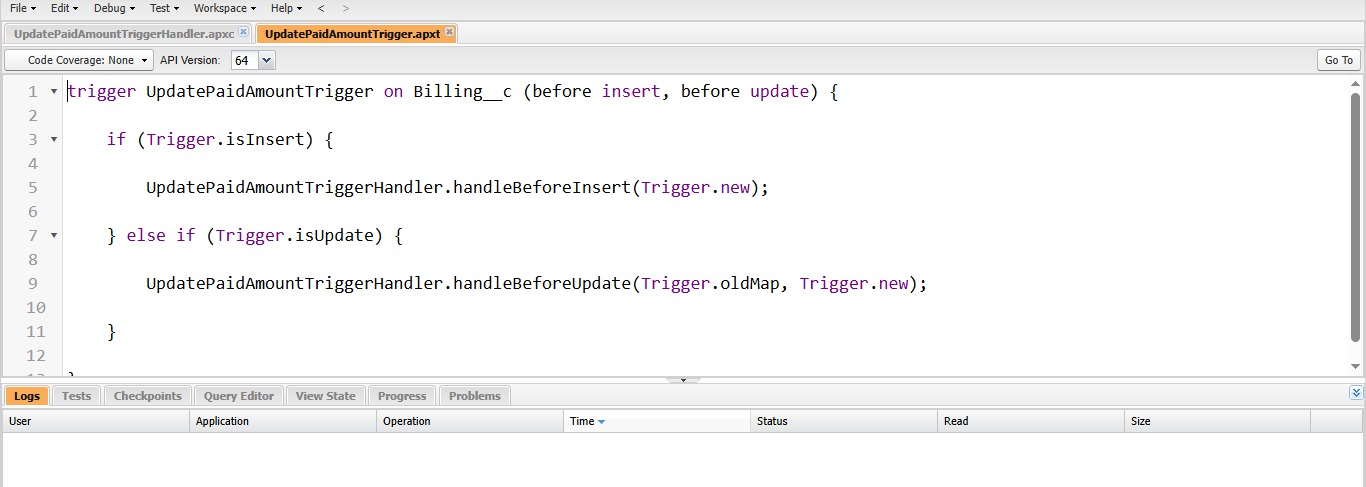
Created permission set for Billing Access.



**2.10 Automation (Triggers & Flows)**

Apex Trigger to auto-generate Billing.

Flow for order confirmation and approval process.



A screenshot of a computer

AI-generated content may be incorrect.

**2.11 Reports & Dashboards**

Reports for sales, customers, monthly sales trends.

Dashboard with KPIs.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**2.12 User Adoption**

Tested by creating, viewing, and deleting Jewel Customer records.

Monitored login history & usage metrics.

A screenshot of a computer

AI-generated content may be incorrect.

**4.Performance Testing**

The purpose of performance testing is to ensure that the CRM functions efficiently under different workloads and maintains smooth user experience.

**Tests Conducted**

**4.1 Record Load Test**

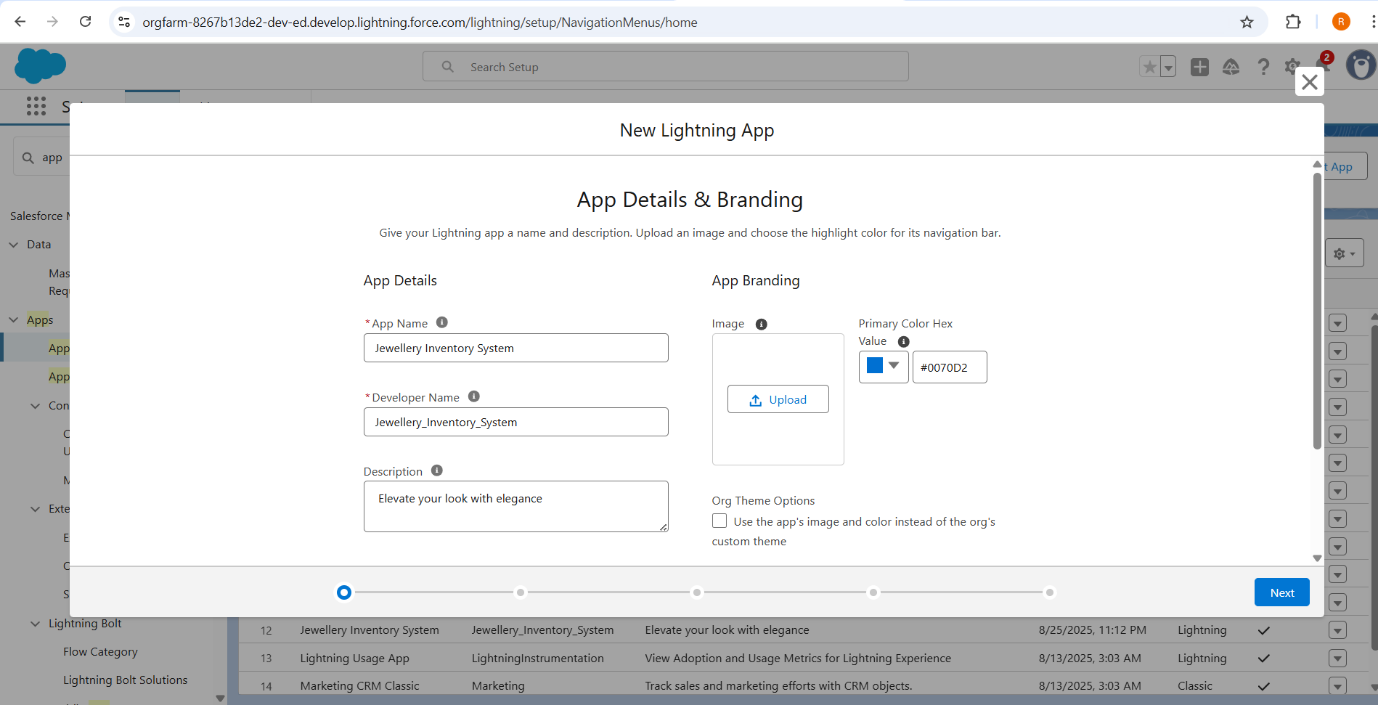
Created 100+ Jewel Customer records to validate system stability and responsiveness.

A screenshot of a computer

AI-generated content may be incorrect.

**4.2 Page Load Performance**

Measured Lightning record page load time for Item and Order objects.

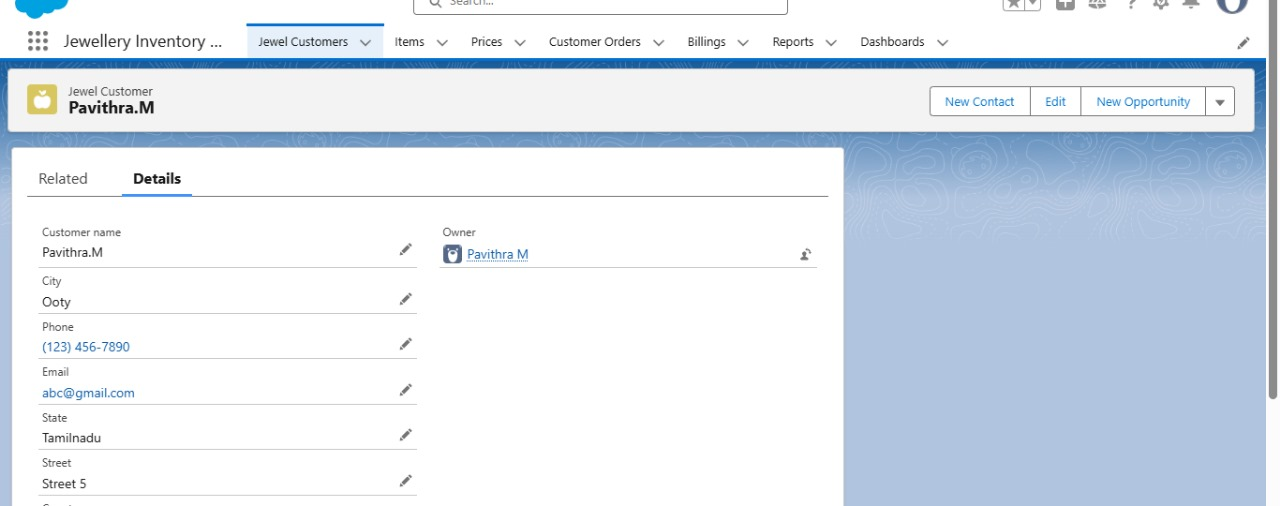


**5. Testing Results**

**5.1 Functional Testing Results**

**Object Creation (Pass)**

Successfully created Jewel Customer and Item records with mandatory fields.



**Validation Rules (Pass)**

Invalid Phone number/Email triggered error messages correctly.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Trigger Automation (Pass)**

Billing record was auto-created when Order status = Confirmed.

A screenshot of a computer

AI-generated content may be incorrect.

**Reports & Dashboards (Pass)**

Reports displayed accurate totals, dashboards generated charts successfully.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**5.2 Performance Testing Results**

Record Load Test (Pass)

Created and handled 100+ Jewel Customer records without errors.

A screenshot of a computer

AI-generated content may be incorrect.

**7. Conclusion**

The Jewel Management CRM Application in Salesforce successfully streamlines jewelry business operations by managing customers, items, orders, and billing in a single platform. With automation through triggers and flows, secure role-based access, and powerful reports and dashboards, the system improves efficiency and supports better decision-making.

The project also lays a strong foundation for future enhancements like AI recommendations, chatbot integration, and mobile app support, making it a scalable and reliable solution for jewel management.

**8. Appendix**

**8.1 Source Code: Apex Trigger & Handler**

**OrderTrigger.apxt**

trigger OrderTrigger on Order\_\_c (after insert, after update) {

if (Trigger.isAfter && (Trigger.isInsert || Trigger.isUpdate)) {

OrderTriggerHandler.createBilling(Trigger.new);

}

}

**OrderTriggerHandler.apxc**

public class OrderTriggerHandler {

public static void createBilling(List<Order\_\_c> newOrders) {

List<Billing\_c> billingList = new List<Billing\_c>();

for (Order\_\_c ord : newOrders) {

if (ord.Status\_\_c == 'Confirmed') {

Billing\_c bill = new Billing\_c();

bill.Order\_\_c = ord.Id;

bill.Invoice\_No\_\_c = 'INV-' + String.valueOf(System.currentTimeMillis());

bill.Amount\_Paid\_c = ord.Total\_Price\_c;

bill.Payment\_Status\_\_c = 'Pending';

billingList.add(bill);

}

}

if (!billingList.isEmpty()) {

insert billingList;

}

}

}

**8.2 Scheduled Job Example**

MonthlyReportScheduler.ap

global class MonthlyReportScheduler implements Schedulable {

global void execute(SchedulableContext sc) {

sendMonthlySalesReport();

}

public static void sendMonthlySalesReport() {

List<Order\_c> orders = [SELECT Id, Total\_Pricec, Customer\_r.Name

FROM Order\_\_c

WHERE CreatedDate = THIS\_MONTH];

// Example: Send email with monthly report (simplified)

String emailContent = 'This is the monthly sales report. Total Orders: ' + orders.size();

Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();

email.setToAddresses(new String[]{'manager@jewelcrm.com'});

email.setSubject('Monthly Sales Report');

email.setPlainTextBody(emailContent);

Messaging.sendEmail(new Messaging.SingleEmailMessage[]{email});

}

}