

Insurance Policy & Management System


Salesforce Implementation Project Documentation

Project Overview

The **Insurance Management System** is a Salesforce-based application designed to streamline the management of **policies, claims, and approvals** for insurance companies. The solution provides separate access and dashboards for **Agents, Managers, and Administrators**, ensuring each role has clear responsibilities and visibility.

- **Agents** can create customer policies, upload claim requests, and track their work.
- **Managers** can review claims submitted by agents, approve/reject them, and monitor team performance.
- **Administrators** have complete control with full visibility into policies, claims, dashboards, users, and system configuration.

The project combines Salesforce's **declarative features (Objects, Flows, Reports, Dashboards, Profiles, Roles, Validation Rules)** with **programmatic features (Apex Classes, Triggers, SOQL, Test Classes)** to build a scalable and secure solution.

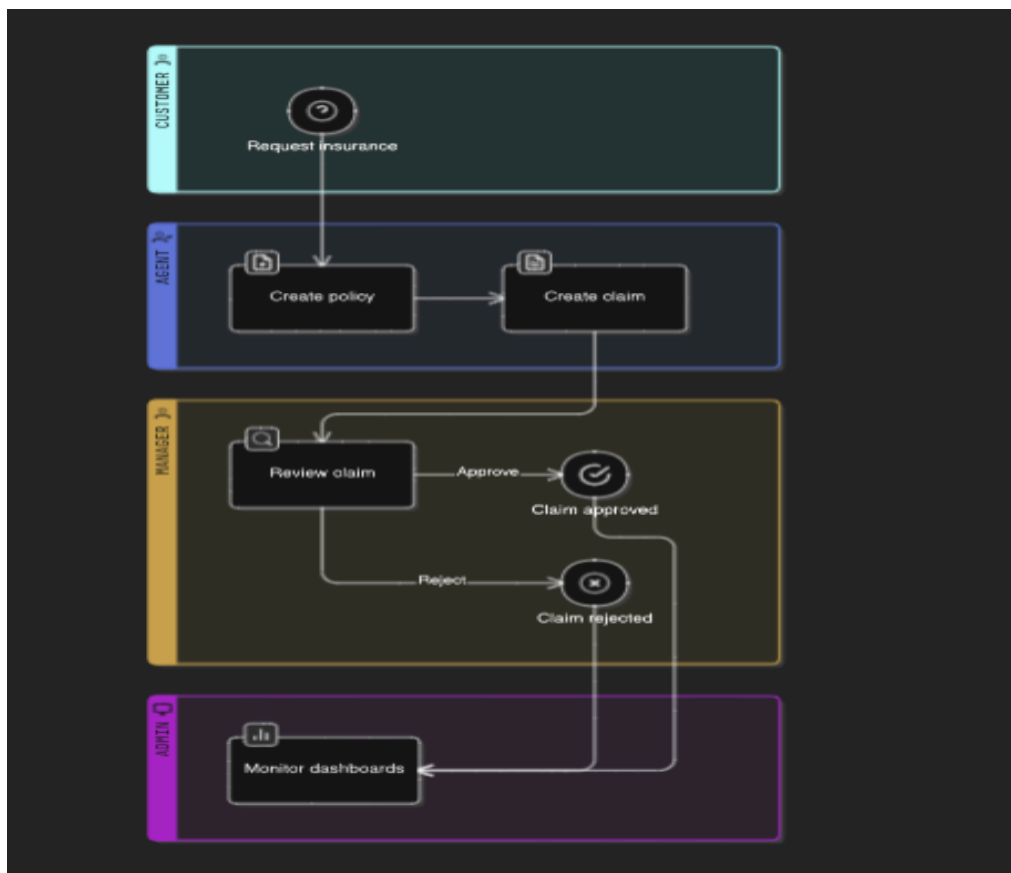
 *Screenshot Placeholder: Home Page of Insurance App with tabs (Policies, Claims, Dashboards).*

Project Objectives

1. Provide a **centralized system** for managing insurance policies and claims.
2. Automate **policy expiry** and **claim approval workflows** to reduce manual work.
3. Enable **role-based dashboards** for Agents, Managers, and Admins.
4. Ensure **data security and compliance** with Salesforce profiles, roles, and sharing rules.
5. Deliver **reporting insights** on claims, policies, and team performance.


Phase 1: Problem Understanding & Industry Analysis

- **Requirement Gathering:** Policies must store premium, type, dates; Claims must store amount, type, status.
- **Stakeholder Analysis:**
 - Agents → Create policies, submit claims.
 - Managers → Approve/reject claims.
 - Admins → Full control, dashboards.
 - Customers → Indirect users (data stored via agents).
- **Business Process Mapping:** Policy lifecycle → Creation → Claim filed → Manager approval → Dashboard reporting.
- **Industry-Specific Use Case:** Delays in manual approvals, poor tracking of expired policies.
- **AppExchange Exploration:** Explored **Vlocity Insurance** but chose custom build.



Phase 2: Org Setup & Configuration

- **Salesforce Edition:** Developer Org used.
- **Company Profile Setup:** Org name, locale, time zone updated.
- **Business Hours & Holidays:** Defined 9 AM–6 PM, excluding holidays.
- **User Setup & Licenses:** Created 3 profiles: Agent, Manager, Admin.
- **Profiles:** Object-level permissions defined.
- **Roles:** Hierarchy → Agent → Manager → Admin.
- **Permission Sets:** For special permissions (e.g., File Upload).
- **OWD:** Private → ensures Managers see only Agent records through hierarchy.
- **Sharing Rules:** Claims shared with Managers.
- **Login Access Policies:** Admin login enabled to impersonate users.
- **Sandbox Usage:** Development & testing.
- **Deployment Basics:** Metadata deployed using Change Sets.

 **SETUP**
Roles

Creating the Role Hierarchy

You can build on the existing role hierarchy shown on this page. To insert a new role, click **Add Role**.

Your Organization's Role Hierarchy

[Collapse All](#) [Expand All](#)

- Insurance Policy & Management System
 - [Add Role](#)
 - CEO [Edit](#) | [Del](#) | [Assign](#)
 - [Add Role](#)
 - Manager [Edit](#) | [Del](#) | [Assign](#)
 - [Add Role](#)
 - Agent [Edit](#) | [Del](#) | [Assign](#)
 - [Add Role](#)

Phase 3: Data Modeling & Relationships

- Custom Objects:
 - Policy__c: Policy Number, Policy Type, Premium, Start Date, End Date, Status.
 - Claim__c: Claim Type, Claim Amount, Claim Date, Status, Related Policy (Lookup).
- Relationships: One Policy → Many Claims.
- Record Types: Policy Types (Life, Health, Vehicle).
- Page Layouts: Separate layouts for Agents, Managers.
- Compact Layouts: Highlight key fields (Policy Type, Premium, Status).
- Schema Builder: Shows relationship between Policy & Claim.

SETUP
Object Manager
1 Items, Sorted by Label

policy

Schema Builder

Create

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Policy	Policy__c	Custom Object	Stores insurance policy details (type, dates, premium, status).	9/27/2025	✓

SETUP
Object Manager
1 Items, Sorted by Label

claim

Schema Builder

Create

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Claim	Claim__c	Custom Object	Stores claim requests linked to insurance policies.	9/28/2025	✓

Phase 4: Process Automation (Admin)

- Validation Rules:
 - Premium > 0.
 - End Date > Start Date.
- Approval Process: Manager approval on Claims.

- **Flow Builder:**
 - Record-Triggered Flow → Auto-set Policy to Expired if End Date < Today.
 - Scheduled Flow → Daily expiry check at midnight.
 - **Email Alerts:** Notify Managers on new claims.
 - **Tasks & Notifications:** Auto-create follow-up task when claim filed.
-

Phase 5: Apex Programming (Developer)

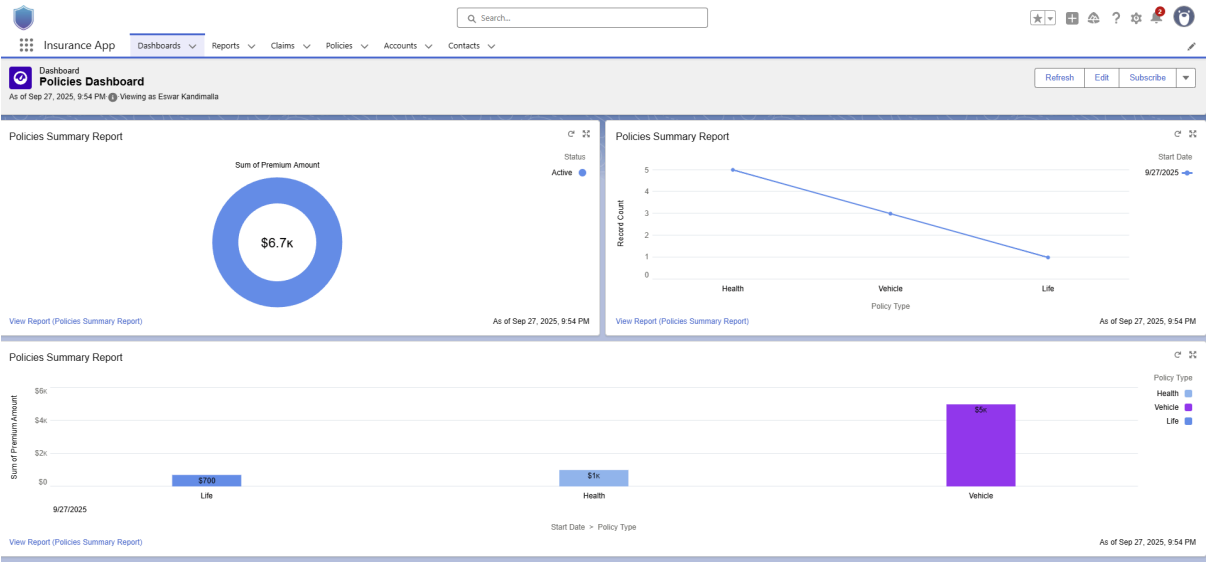
- **Apex Classes:** PolicyHandler.cls, ClaimHandler.cls.
 - **Triggers:**
 - Auto-generate Policy Number.
 - Auto-update Policy Claim Count.
 - **SOQL:** Fetch policies by type, claims by status.
 - **Batch Apex:** Update expired policies.
 - **Scheduled Apex:** Run expiry process daily.
 - **Future Methods:** Async notifications.
 - **Test Classes:** Achieved >75% code coverage.
-

Phase 6: User Interface Development

- **Lightning App Builder:** Created “Insurance App”.
- **Tabs:** Policies, Claims, Dashboards.
- **Home Page:** Summary of policies/claims.
- **Record Pages:** Custom layouts for Policy & Claim objects.

- **Dashboards:** Role-specific dashboards for Agents, Managers, Admins.

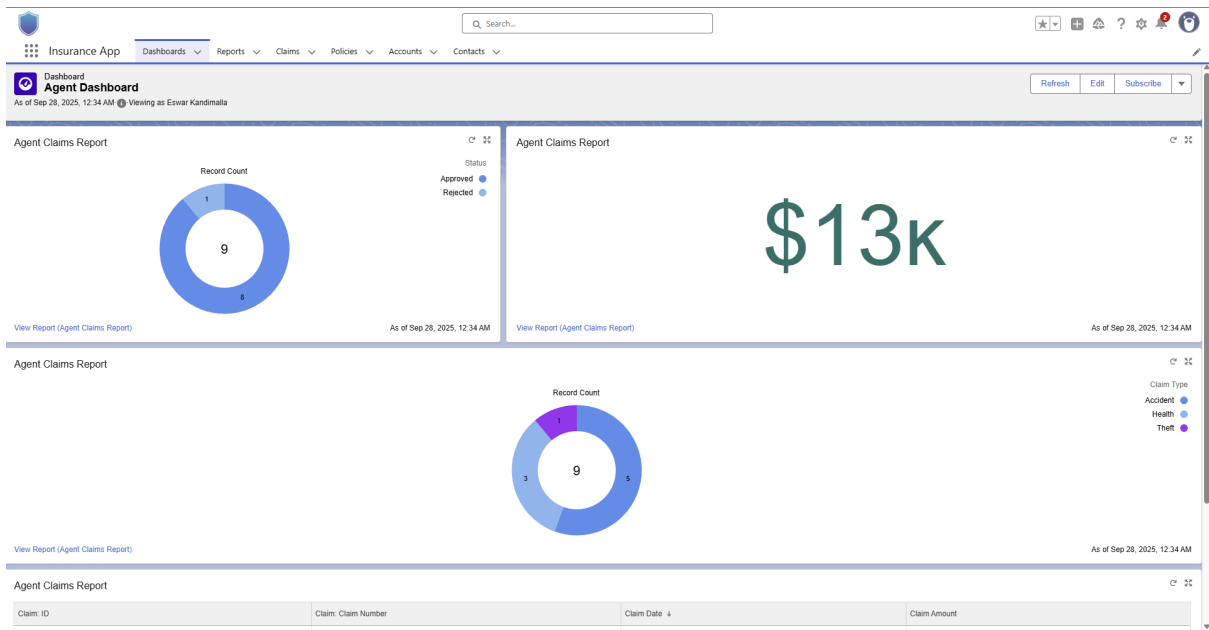
Policies Dashboard :



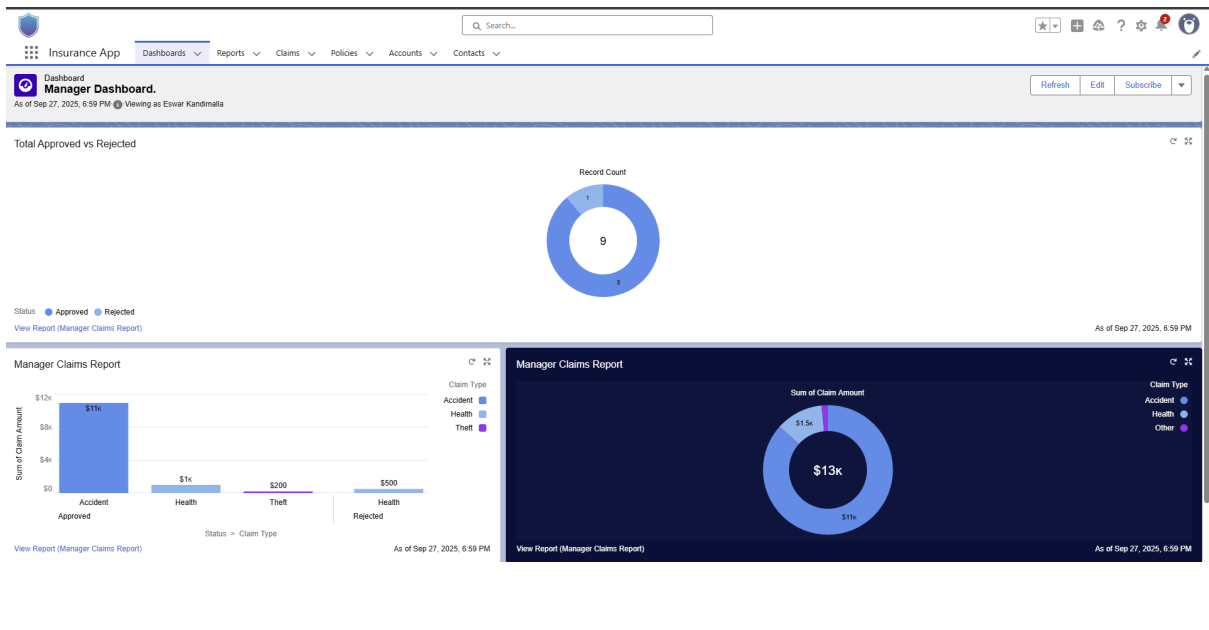
Claims Dashboard:



Agent Dashboard:



Manager Dashboard:



Phase 7: Integration & External Access

- **Named Credentials:** Future use for payment gateway integration.
- **Web Services (REST):** Possible future integration with external claim systems.
- **Platform Events:** Claim event notifications.

- **Change Data Capture:** Track policy updates in real time.
- **Remote Site Settings:** Configured for external callouts.
- _____

Phase 8: Data Management & Deployment

- **Data Import Wizard:** Imported sample policies & claims.
 - **Data Loader:** Bulk uploads for claims.
 - **Duplicate Rules:** Prevent duplicate Policy Numbers.
 - **Data Export:** Scheduled weekly backups.
 - **Change Sets:** Moved changes from sandbox to production.
 - **VS Code & SFDX:** Apex & metadata deployment.
-

Phase 9: Reporting, Dashboards & Security Review

- **Reports:**
 - Policies by Type.
 - Claims by Status.
 - Agent Performance.
- **Dashboards:**
 - Agent Dashboard → Policies, Claims handled.
 - Manager Dashboard → Approved vs Rejected, Claims by Type.
 - Admin Dashboard → Enterprise-wide view.
- **Dynamic Dashboards:** Manager sees only their team data.
- **Security Review:**

- Field-Level Security: Hide Premium from Agents.
- Session Settings: Timeout 30 min.
- Login IP Ranges: Restricted for Admin.
- Audit Trail: Setup changes tracked.

Phase 10: Final Presentation & Demo

- **Pitch Presentation:** Problem → Solution → Demo.
- **Demo Walkthrough:**
 - Agent → Create Policy & Claim → Agent Dashboard.
 - Manager → Review Claim → Approve/Reject → Manager Dashboard.
 - Admin → Organization Dashboard.
- **Handoff Documentation:** Project setup guide.

Conclusion

The Insurance Management System successfully demonstrates how Salesforce can be used to **automate insurance workflows**:

- Agents handle customer-facing tasks.
- Managers process claims efficiently.
- Admins oversee the business with dashboards.

The system integrates **data modeling, automation, Apex coding, reporting, and security**, making it both scalable and enterprise-ready.