

Insurance Claim Tracking System

PHASE 2–5 IMPLEMENTATION (DETAILED)

2. Salesforce Org Setup

Purpose:

The purpose of this phase was to establish a secure and structured Salesforce environment where all insurance claim-related operations could be configured. This setup ensures that data security, access control, and company-wide standards are properly implemented before moving on to data modeling and automation.

Implementation:

1. Developer Org/Sandbox Creation:

- A Salesforce Developer Org was created for initial configuration and testing.
- Alternatively, a Sandbox can be used to mirror production settings.

2. Company Settings:

- Configured Company Profile including organization name, address, primary currency, and locale.
- Defined Business Hours to align with customer service availability.
- Configured Fiscal Year settings to match insurance industry standards.

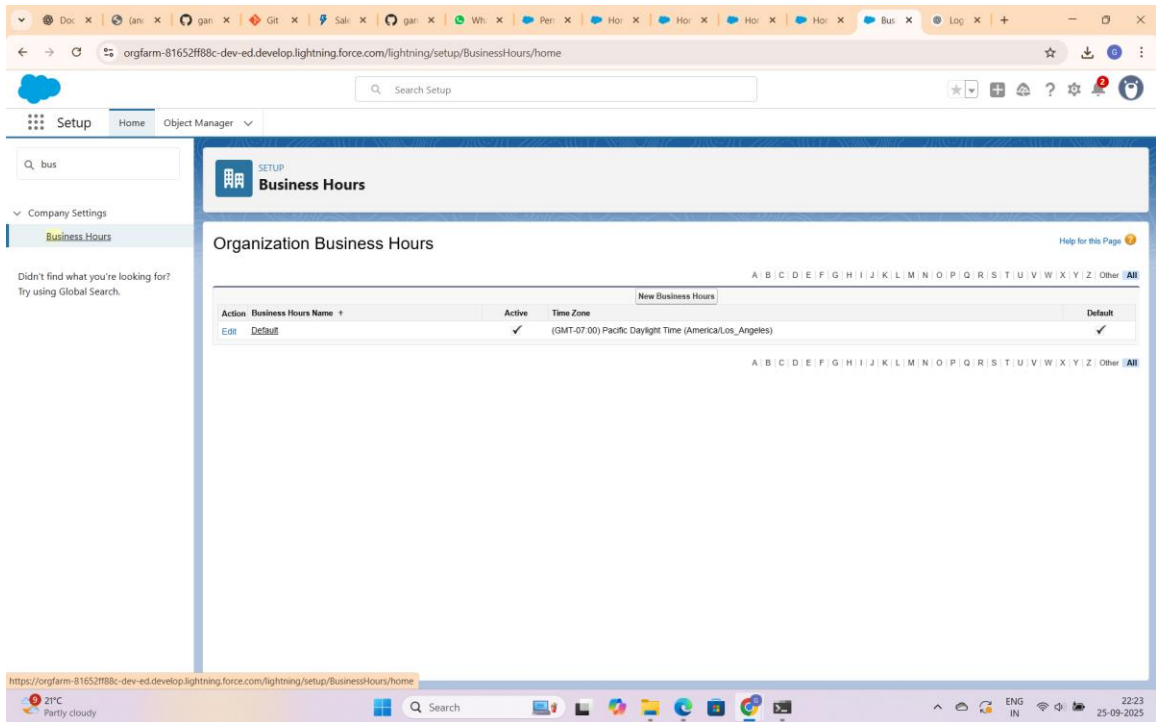
The screenshot shows the Salesforce Setup interface for a company named 'SJBIT'. The left sidebar contains a navigation menu with options like 'Company Settings', 'Calendar Settings', 'Company Information', 'Data Protection and Privacy', 'Fiscal Year', 'Holidays', 'Language Settings', and 'My Domain'. The main content area is titled 'Company Information' and displays the organization's profile details. Below this, there is a 'User Licenses' section with a table showing license status and expiration dates.

Organization Detail

Field	Value
Organization Name	SJBIT
Primary Contact	OrgFarm EPIC
Division	
Address	United States
Fiscal Year Starts In	April
Activate Multiple Currencies	<input type="checkbox"/>
Enable Data Translation	<input type="checkbox"/>
Newsletter	<input checked="" type="checkbox"/>
Admin Newsletter	<input checked="" type="checkbox"/>
Hide Notices About System Maintenance	<input type="checkbox"/>
Hide Notices About System Downtime	<input type="checkbox"/>
Locale Formats	ICU
Phone	
Fax	
Default Locale	English (India)
Default Language	English
Default Time Zone	(GMT-07:00) Pacific Daylight Time (America/Los_Angeles)
Currency Locale	English (United States) - USD
Used Data Space	352 KB (7%) View
Used File Space	18 KB (0%) View
API Requests, Last 24 Hours	18 (15,000 max)
Streaming API Events, Last 24 Hours	0 (10,000 max)
Restricted Logins, Current Month	0 (0 max)
Salesforce.com Organization ID	00DgK00000byOx8
Organization Edition	Developer Edition
Instance	CAN96
Created By	OrgFarm EPIC, 9/18/2025, 4:24 PM
Modified By	Ganashree B S, 9/24/2025, 6:52 AM

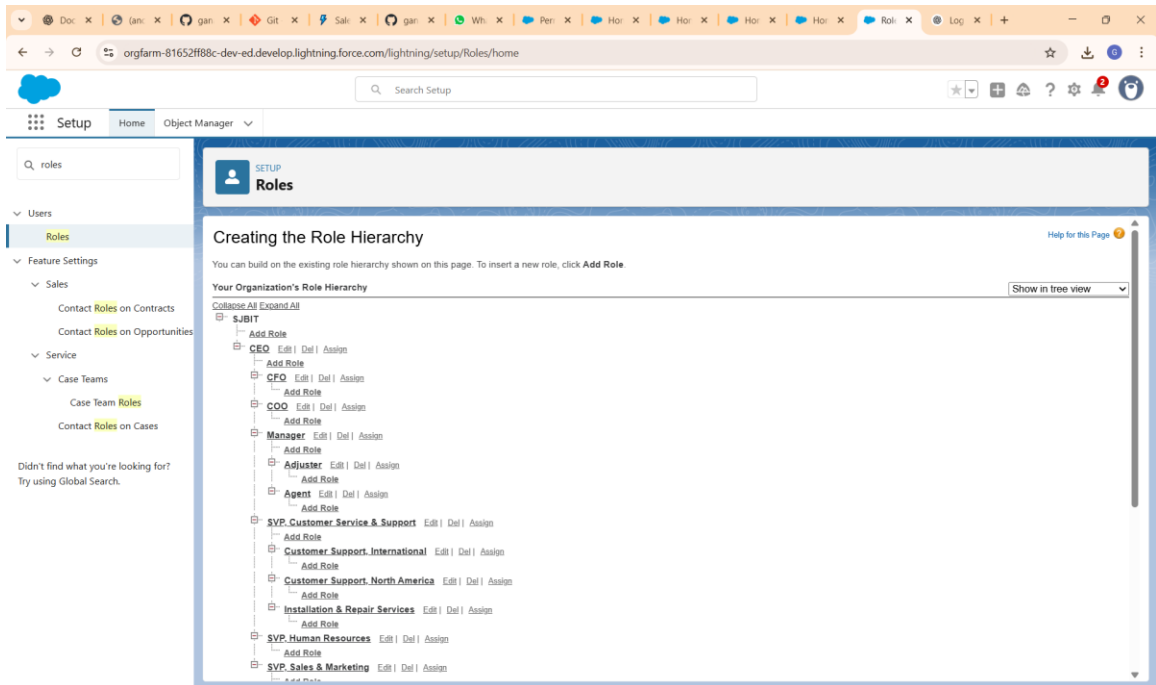
User Licenses

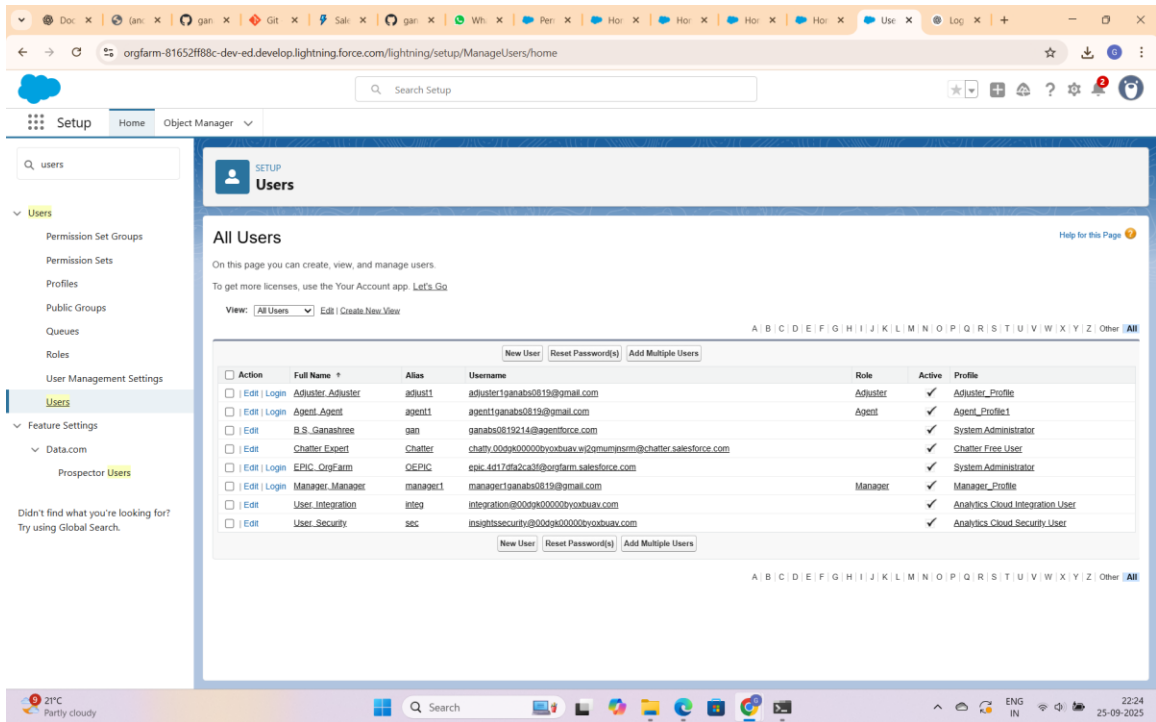
Name	Status	Total Licenses	Used Licenses	Remaining Licenses	Expiration Date
...



3. Role Hierarchy Setup:

- Created roles for Admin, Adjuster, Agent, and Manager.
- Ensured proper hierarchy where Managers oversee Adjusters and Agents.



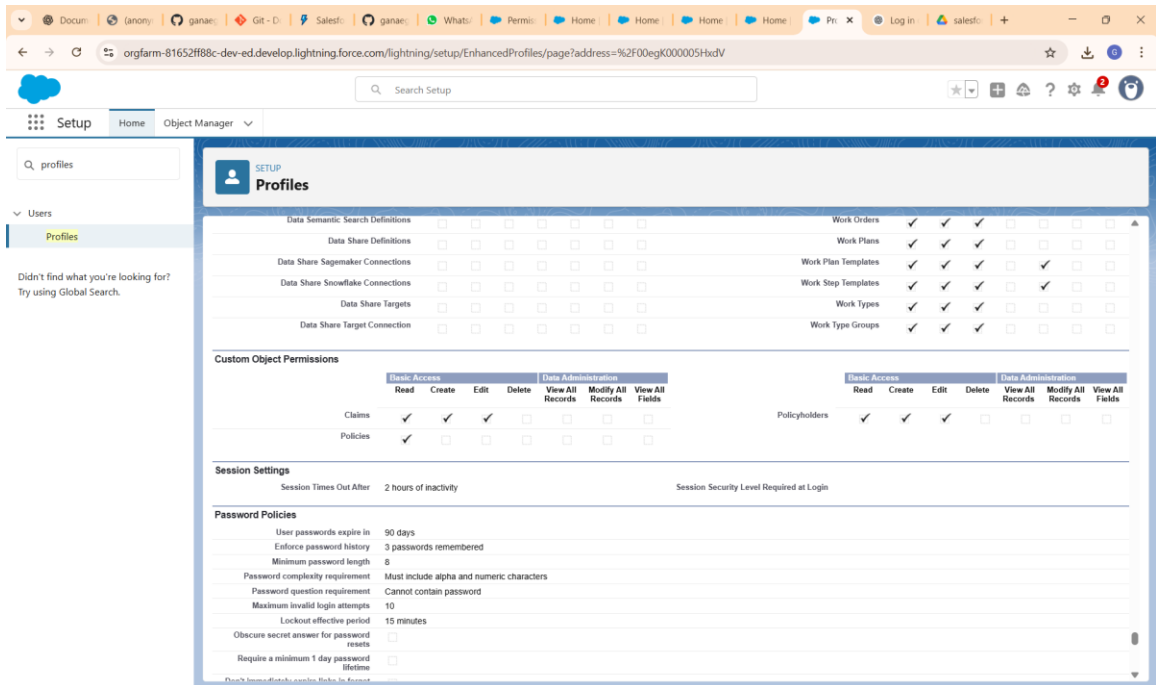


The screenshot shows the Salesforce Setup interface for managing users. The left sidebar contains navigation options: Setup, Home, Object Manager, and a search bar. The main content area is titled "All Users" and includes a description: "On this page you can create, view, and manage users. To get more licenses, use the Your Account app. Let's Go". Below this is a table of users with columns for Action, Full Name, Alias, Username, Role, Active, and Profile. The table lists several users, including Admin, Agent, and Manager roles. At the bottom, there are buttons for "New User", "Reset Password(s)", and "Add Multiple Users".

Action	Full Name	Alias	Username	Role	Active	Profile
[Edit] [Login]	Admin	admin	admin@org.com	Admin	✓	Admin_Profile
[Edit] [Login]	Agent	agent	agent@org.com	Agent	✓	Agent_Profile
[Edit]	B.S. Ganeshree	gan	gan@org.com	System Administrator	✓	System Administrator
[Edit]	Chatter Expert	chatter	chatter@org.com	Chatter Free User	✓	Chatter Free User
[Edit] [Login]	EPIC_OrgEam	OEPIE	epic_4d17f42a36@org.com	System Administrator	✓	System Administrator
[Edit] [Login]	Manager Manager	manager	manager@org.com	Manager	✓	Manager_Profile
[Edit]	User Integration	integ	integration@org.com	Analytics Cloud Integration User	✓	Analytics Cloud Integration User
[Edit]	User Security	sec	insightssecurity@org.com	Analytics Cloud Security User	✓	Analytics Cloud Security User

4. Profiles and Permission Sets:

- Customized Profiles to restrict or grant access based on responsibilities.
- Added Permission Sets for advanced tasks like claim approvals.



The screenshot shows the Salesforce Setup interface for managing profiles. The left sidebar contains navigation options: Setup, Home, Object Manager, and a search bar. The main content area is titled "Profiles" and includes a table of profiles with columns for Name, Role, and Active. Below this is a section for "Custom Object Permissions" with a table of permissions for various objects. At the bottom, there are sections for "Session Settings" and "Password Policies".

Name	Role	Active
Admin	Admin	✓
Agent	Agent	✓
Manager	Manager	✓

Object	Read	Create	Edit	Delete	View All Records	Modify All Records	View All Fields
Claims	✓	✓	✓	✓	✓	✓	✓
Policies	✓	✓	✓	✓	✓	✓	✓

Session Settings

Session Times Out After: 2 hours of inactivity

Session Security Level Required at Login: [Select]

Password Policies

User passwords expire in: 90 days

Enforce password history: 3 passwords remembered

Minimum password length: 8

Password complexity requirement: Must include alpha and numeric characters

Password question requirement: Cannot contain password

Maximum invalid login attempts: 10

Lockout effective period: 15 minutes

Obscure secret answer for password resets: [Select]

Require a minimum 1 day password lifetime: [Select]

The top screenshot displays the 'Login Access Policies' page in Salesforce Setup. The left sidebar shows the navigation menu with 'Security' expanded and 'Login Access Policies' selected. The main content area is titled 'Login Access Policies' and includes a sub-header 'Manage Support Options'. Below this, there is a table with columns for 'Setting', 'Packages', 'Available to Users', and 'Available to Administrators Only'. The 'Setting' column lists 'Administrators Can Log in as Any User' and 'Support Organization'. The 'Available to Users' column has a radio button selected for 'Support Organization'. The 'Available to Administrators Only' column has a radio button selected for 'Support Organization'. The bottom screenshot displays the 'Sharing Settings' page in Salesforce Setup. The left sidebar shows the navigation menu with 'Security' expanded and 'Sharing Settings' selected. The main content area is titled 'Sharing Settings' and includes a table with columns for 'Object', 'Sharing Model', and 'Sharing Method'. The table lists various objects such as 'Streaming Channel', 'Tableau Host Mapping', 'User Presence', 'User Provisioning Request', 'Waitlist', 'Web Cart Document', 'Work Order', 'Work Plan', 'Work Plan Template', 'Work Step Template', 'Work Type', 'Work Type Group', 'Claim', 'Policy', and 'Policyholder'. The 'Sharing Model' column shows the current sharing model for each object, and the 'Sharing Method' column shows the sharing method. Below the table, there is a section for 'Other Settings' with checkboxes for 'Manager Groups', 'Secure guest user record access', and 'Require permission to view record names in lookup fields'. At the bottom, there is a section for 'Sharing Rules' with a 'New' button and a 'Recalculate' button.

5. Organization-Wide Defaults (OWD):

- Set Claims object to Private to ensure only authorized users can access claim details.
- Set Policies object to Read-Only to balance transparency with data integrity.

sharin

Security

- Guest User Sharing Rule Access Report
- Sharing Settings

Didn't find what you're looking for?
Try using Global Search.

Sharing Settings

Tableau Host Mapping	Public Read Only	Private	✓
User Presence	Public Read Only	Private	✓
User Provisioning Request	Private	Private	✓
Waitlist	Private	Private	✓
Web Cart Document	Private	Private	✓
Work Order	Private	Private	✓
Work Plan	Private	Private	✓
Work Plan Template	Private	Private	✓
Work Step Template	Private	Private	✓
Work Type	Private	Private	✓
Work Type Group	Public Read/Write	Private	✓
Claim	Private	Private	✓
Policy	Public Read Only	Private	✓
Policyholder	Public Read/Write	Private	✓

Other Settings

Other Settings Help ?

Manager Groups ☐ ⓘ

Secure guest user record access ☒ ⓘ

Require permission to view record names in lookup fields ☐ ⓘ

Sharing Rules

Lead Sharing Rules

Lead Sharing Rules Help ?

No sharing rules specified.

6. Verification:

- Created test users under different roles to validate access and security rules.

Business Impact:

By configuring the Salesforce Org with roles, profiles, and OWD, the project ensures data security and role-based access control. This prevents unauthorized access, supports compliance with industry regulations, and lays a strong foundation for upcoming phases.

Testing/Verification:

Test users were assigned to roles such as Agent, Adjuster, and Manager. Each user's access was verified to confirm they could only see and perform the actions allowed for their profile. For example, Agents could view their assigned claims but not access claims owned by other agents.

Completion Status: Salesforce Org setup successfully completed with secure foundation for claim tracking.

3. Data Modeling

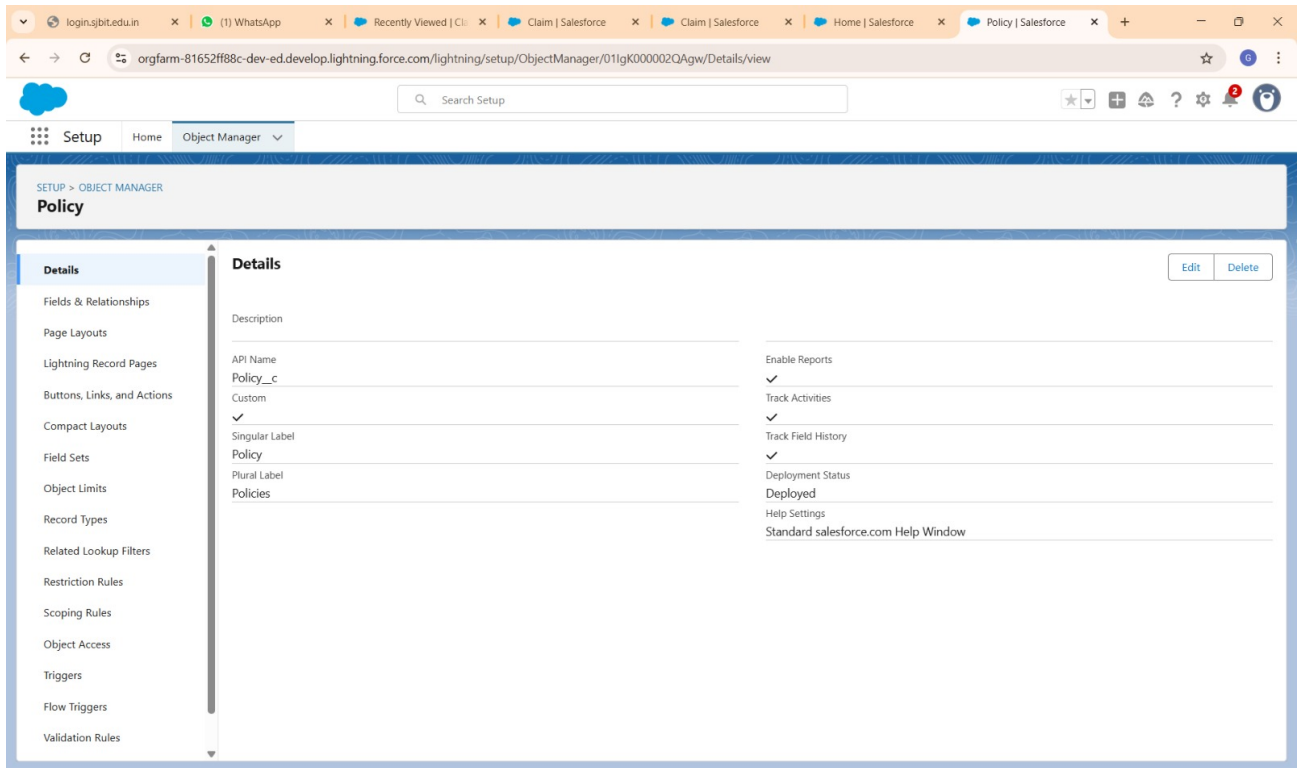
Purpose:

This phase focused on designing the underlying data model to store and manage information about policyholders, policies, and claims. A scalable and logical data structure was essential for enabling claim processing, reporting, and automation.

Implementation:

1. Custom Objects:

- Policyholder__c with fields like Name, Date of Birth, and Contact Details.
- Policy__c with fields such as Policy Number, Coverage Amount, Policy Type.
- Claim__c with fields like Claim Number, Claim Amount, Status, Claim Date.



2. Relationships:

- Policyholder__c → Policy__c: One policyholder may have multiple policies.
- Policy__c → Claim__c: One policy may have multiple claims.

login.sbit.edu.in (1) WhatsApp Recently Viewed | Claim | Salesforce Claim | Salesforce Home | Salesforce Policy | Salesforce

orgfarm-81652ff88c-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01gK000002QA9w/FieldsAndRelationships/view

Search Setup

Setup Home Object Manager

SETUP > OBJECT MANAGER

Policy

Details

Fields & Relationships

9 Items, Sorted by Field Label

Quick Find New Deleted Fields Field Dependencies Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Coverage Amount	Coverage_Amount__c	Currency(18, 0)		
Coverage End Date	Coverage_End_Date__c	Date		
Coverage Start Date	Coverage_Start_Date__c	Date		
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Policy Name	Name	Text(80)		✓
Policyholder	Policyholder__c	Lookup(Policyholder)		✓
Record Type	RecordTypeId	Record Type		✓

25°C Mostly cloudy Search ENG IN 19:12 26-09-2025

3. Record Types:

- Configured separate record types for Health, Auto, and Property claims.
- This allowed tailored page layouts and business rules per claim type.

Document xanonym xganaeg xGit - Do xSalesfor xganaeg xWhatsApp xObject xPolicy | xLog in xsalesfor x

orgfarm-81652ff88c-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgK000002QAgw/RecordTypes/new

Search Setup

SetupHomeObject Manager

SETUP > OBJECT MANAGER

Policy

DetailsFields & RelationshipsPage LayoutsLightning Record PagesButtons, Links, and ActionsCompact LayoutsField SetsObject LimitsRecord TypesRelated Lookup FiltersRestriction RulesScoping RulesObject AccessTriggersFlow TriggersValidation Rules

New Record Type

Policy

Step 1 of 2

Step 1. Enter the details

Enter a name and description for the new record type. The new record type will include all the picklist values from the existing record type selected below. After saving the new record type, you will be able to customize the picklist values.

Record Type

Existing Record Type--Master--Record Type LabelHealthRecord Type NameHealthDescriptionHealth insurance policyActive☒

Select Make Available to give users assigned to this profile the ability to create and clone records of this record type, or assign this record type to existing records. To make the new record type the default for a profile, select Make Default. Users assigned to this record type can still view and edit records associated with record types not available for their profiles.

Profile Name	Record Types Currently Available	<input checked="" type="checkbox"/> Make Available	<input type="checkbox"/> Make Default
Adjuster_Profile		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Agent_Profile		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Agent_Profile1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Analytics Cloud Integration User		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Analytics Cloud Security User		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Anonymous Integration		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Document xanonym xganaeg xGit - Do xSalesfor xganaeg xWhatsApp xObject xPolicy | xLog in xsalesfor x

orgfarm-81652ff88c-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgK000002QAgw/RecordTypes/012gK000002Z5N5/view

Search Setup

SetupHomeObject Manager

SETUP > OBJECT MANAGER

Policy

DetailsFields & RelationshipsPage LayoutsLightning Record PagesButtons, Links, and ActionsCompact LayoutsField SetsObject LimitsRecord TypesRelated Lookup FiltersRestriction RulesScoping RulesObject AccessTriggersFlow TriggersValidation Rules

Record Type

Auto

Back to Custom Object: Policy

Use the Edit button to change the properties of this record type. Use the Edit links in the Picklist Values related list to choose the picklist values available for records with this record type.

Edit

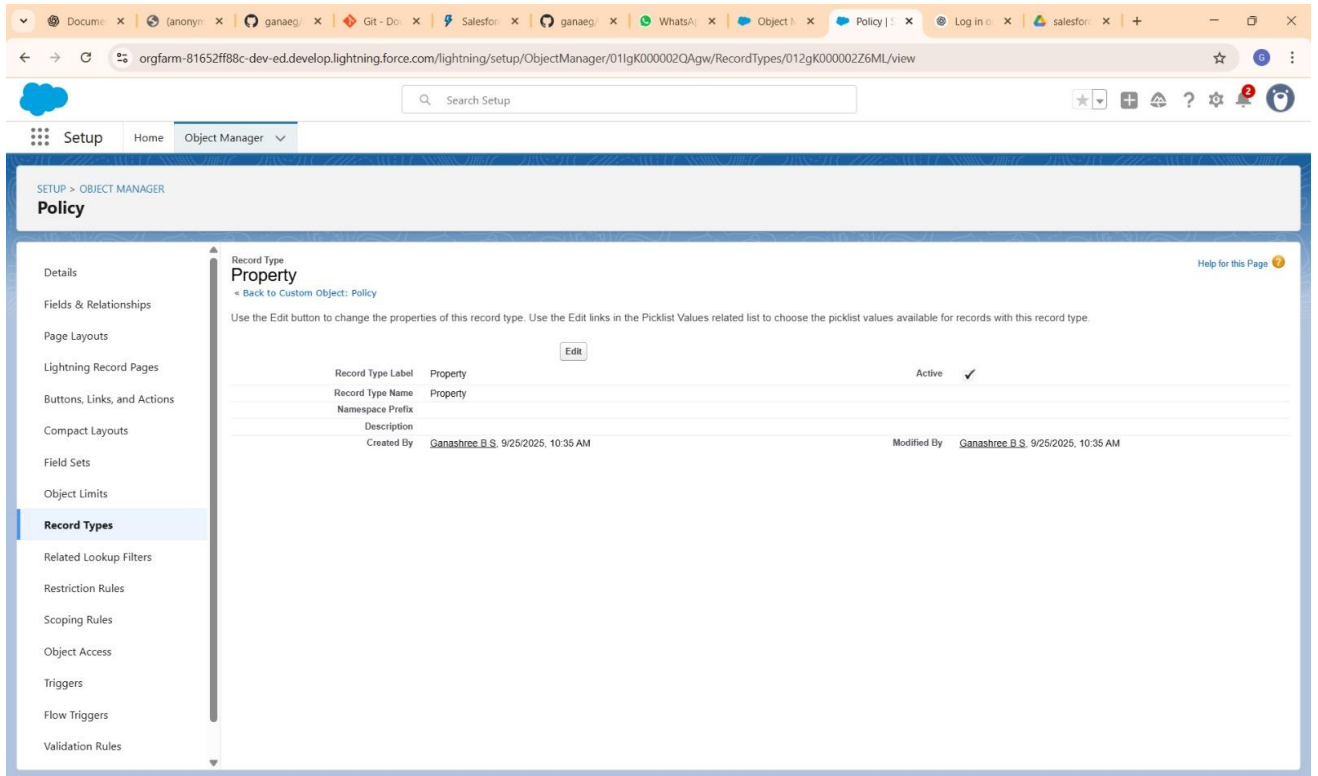
Record Type Label	Auto	Active	<input checked="" type="checkbox"/>
Record Type Name	Auto		
Namespace Prefix			
Description			
Created By	Ganashree D.S.	9/25/2025, 10:34 AM	Modified By Ganashree D.S. 9/25/2025, 10:34 AM

21°C Mostly clear

Search

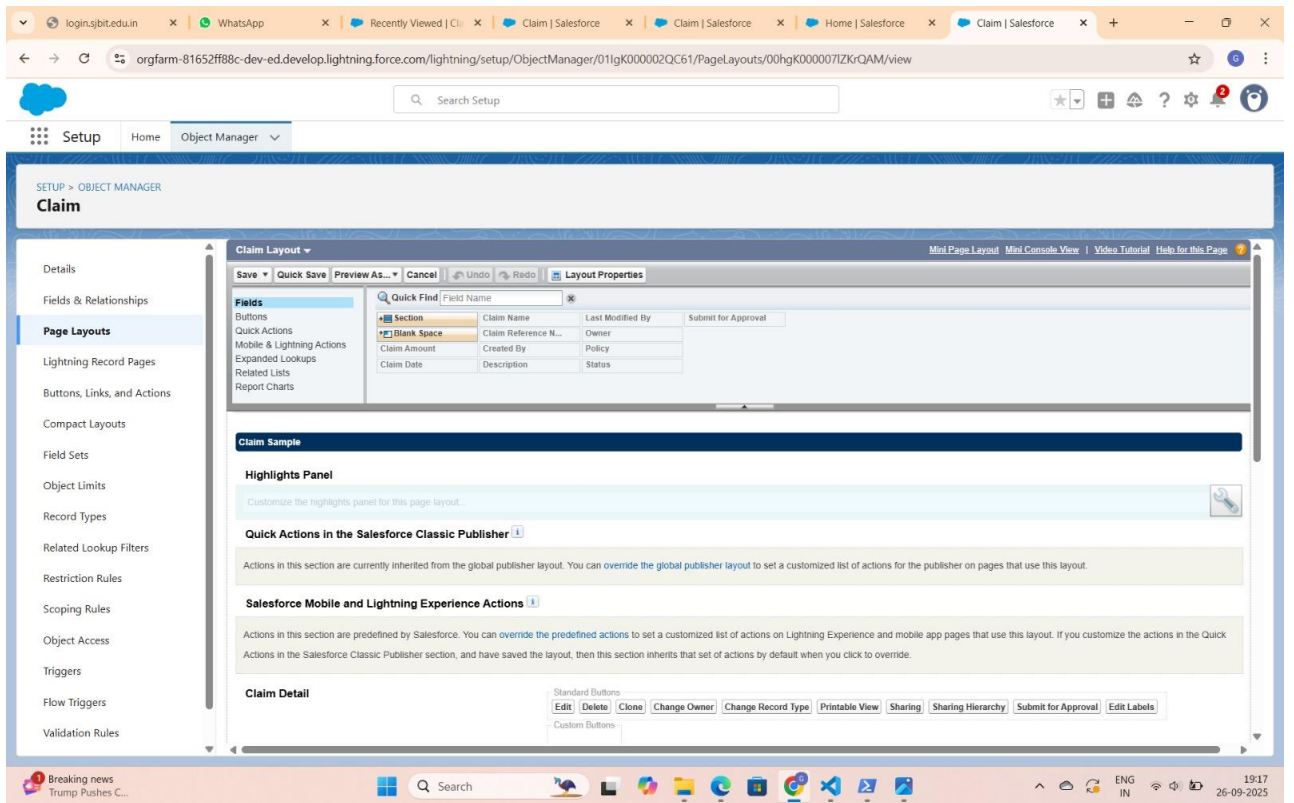
ENG IN

23:04 25-09-2025



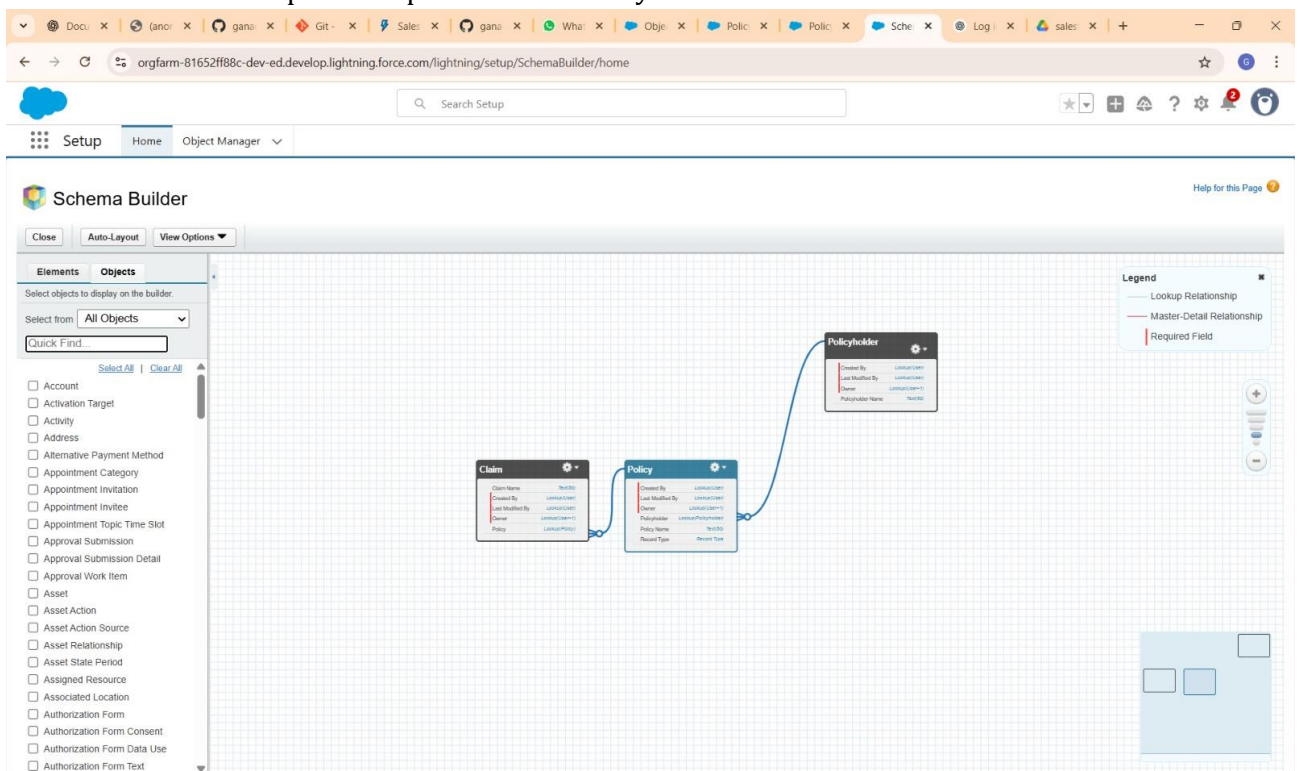
4. Page Layouts & Compact Layouts:

- Designed layouts specific to each role (Agent, Adjuster, Manager).
- Compact layouts optimized for mobile and list views.



5. Schema Builder:

- Verified relationships and dependencies visually.



Business Impact:

A well-defined data model ensures accurate storage and retrieval of insurance data. It enables consistency in claim processing, reduces redundancy, and supports analytics. Future automation and integrations rely heavily on this foundation.

Testing/Verification:

Sample policyholders, policies, and claims were created to validate object relationships. Tests confirmed that policies linked correctly to policyholders, and claims linked correctly to policies.

Completion Status: Data model finalized with record types, layouts, and schema validation.

4. Process Automation

Purpose:

The goal of this phase was to automate business processes to improve efficiency, ensure compliance with rules, and streamline claim submission and approval workflows.

Implementation:

1. Validation Rules:

- Configured rules to ensure claim amount does not exceed policy coverage.
- Formula Example: `Claim_Amount__c <= Policy__r.Coverage_Amount__c`.

The screenshot shows the Salesforce Lightning Setup interface. The browser address bar displays the URL: `orgfarm-81652ff88c-dev-ed.develop.lightning.force.com/lightning/setup/ApprovalProcesses`. The top navigation bar includes the Salesforce logo, a search bar labeled "Search Setup", and tabs for "Setup", "Home", and "Object Manager". The left sidebar shows the "Approval Processes" section under "Process Automation". The main content area is titled "Approval Processes" and contains two sections: "Specify Entry Criteria" and "Select Approver".

Specify Entry Criteria

Use this approval process if the following **criteria are met** ▼:

Field	Operator
--None--	--None--
--None--	--None--
--None--	--None--
--None--	--None--
--None--	--None--

[Add Filter Logic...](#)

Select Approver

Using the options below, specify the user to whom the approval request should be assigned.

☐ Let the submitter choose the approver manually.

☐ Automatically assign an approver using a standard or custom hierarchy field:

--None-- ▼

☒ Automatically assign to queue. 🔍

☐ Automatically assign to approver(s).

Because this is the Jump Start Wizard, Salesforce automatically chooses some settings for you. [Show More](#)

Save Cancel

2. Approval Process:

- Created approval workflow requiring Manager approval for high-value claims.
- Configured rejection paths and notifications.

HotelsWhatsAppAppLightning

AccountLightning

orgfarm-81652ff88c-dev-ed.develop.lightning.force.com/lightning/setup/ApprovalProcesses

Search Setup

Setup

HomeObject Manager

Approval Processes

Process Automation

Approval Processes

Didn't find what you're looking for?
Try using Global Search.

SETUP

Approval Processes

Specify Entry Criteria

Use this approval process if the following criteria are met

Field

--None--

--None--

--None--

--None--

--None--

Add Filter Logic...

Select Approver

Using the options below, specify the user to whom the approval request should be assigned.

Let the submitter choose the approver manually.

Automatically assign an approver using a standard or custom hierarchy field:

--None--

Automatically assign to queue.

Automatically assign to approver(s).

Because this is the Jump Start Wizard, Salesforce automatically chooses some settings for you. [Show More](#)

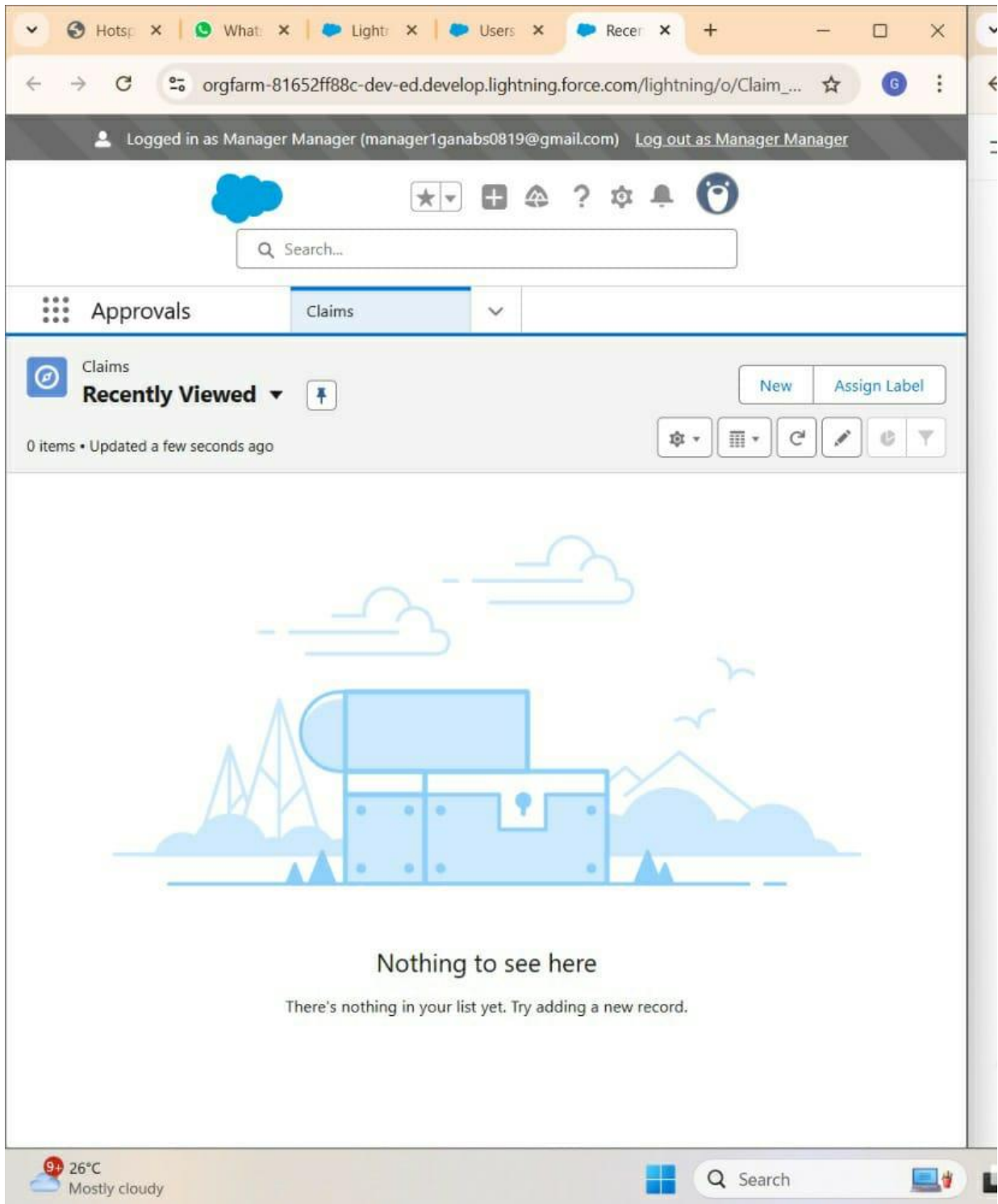
Save

Cancel

9+ Rainy days ahead
27°C

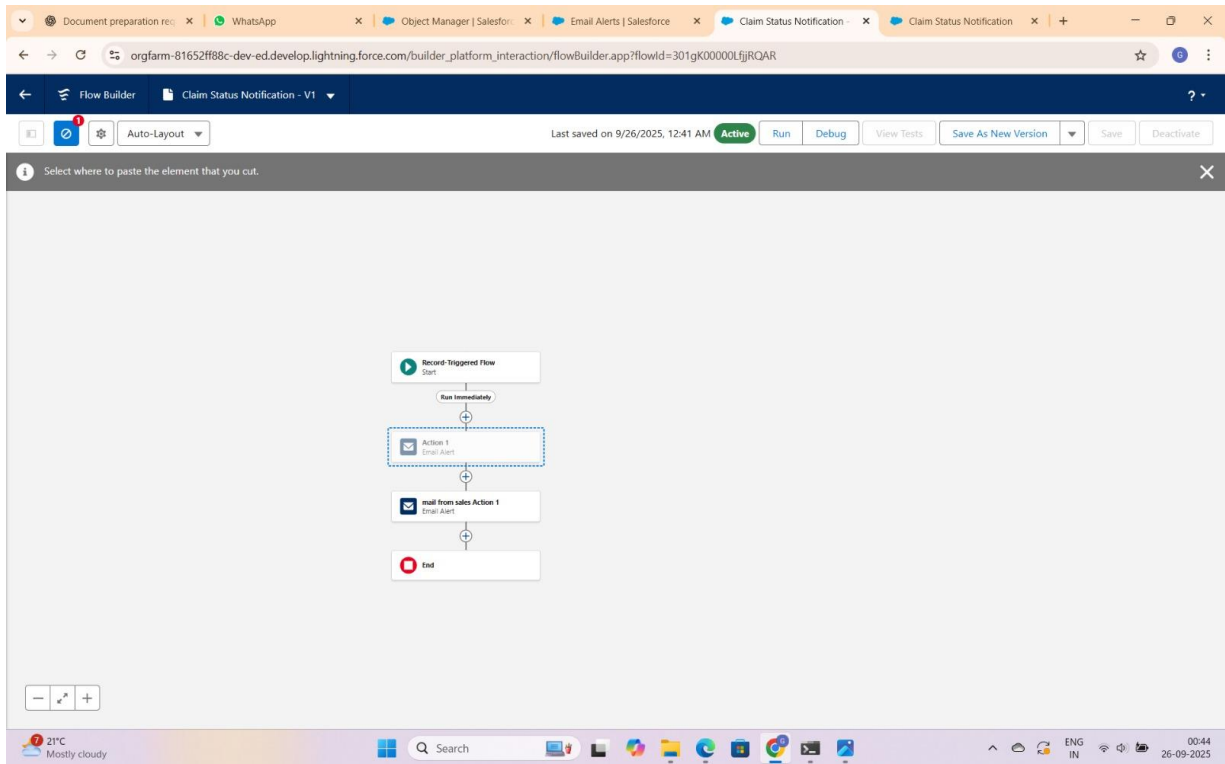
Windows

Search



3. Flows:

- Record-triggered flow to send notifications when claims are created.
- Screen flow to guide agents through claim submission.



4. Notifications:

- Email alerts set up for policyholders upon claim submission.
- SMS alerts integrated with third-party service.

The screenshot shows the Salesforce Lightning interface for a 'Claim Approval' process instance step. The status is 'Approved'. The details show the claim name 'win' and the owner 'Agent Agent'. The approver is 'Manager Manager'.

Submitter	Date Submitted	Actual Approver	Assigned To
Agent Agent	25-Sept-2025	Manager Manager	Manager Manager

Approval Details

Claim Name	Owner
win	Agent Agent

Approver Comments

Manager Manager
ya its good
25-Sept-2025, 5:54:46 am

Business Impact:

Automating claim submission and approval reduced manual work, ensured policy rules were enforced consistently, and improved customer experience with real-time notifications.

Testing/Verification:

Created multiple test claims to validate that low-value claims bypassed manager approval, while high-value claims routed correctly to managers. Notifications were verified via test emails.

Completion Status: Process automation implemented and successfully verified.

5. Apex Programming

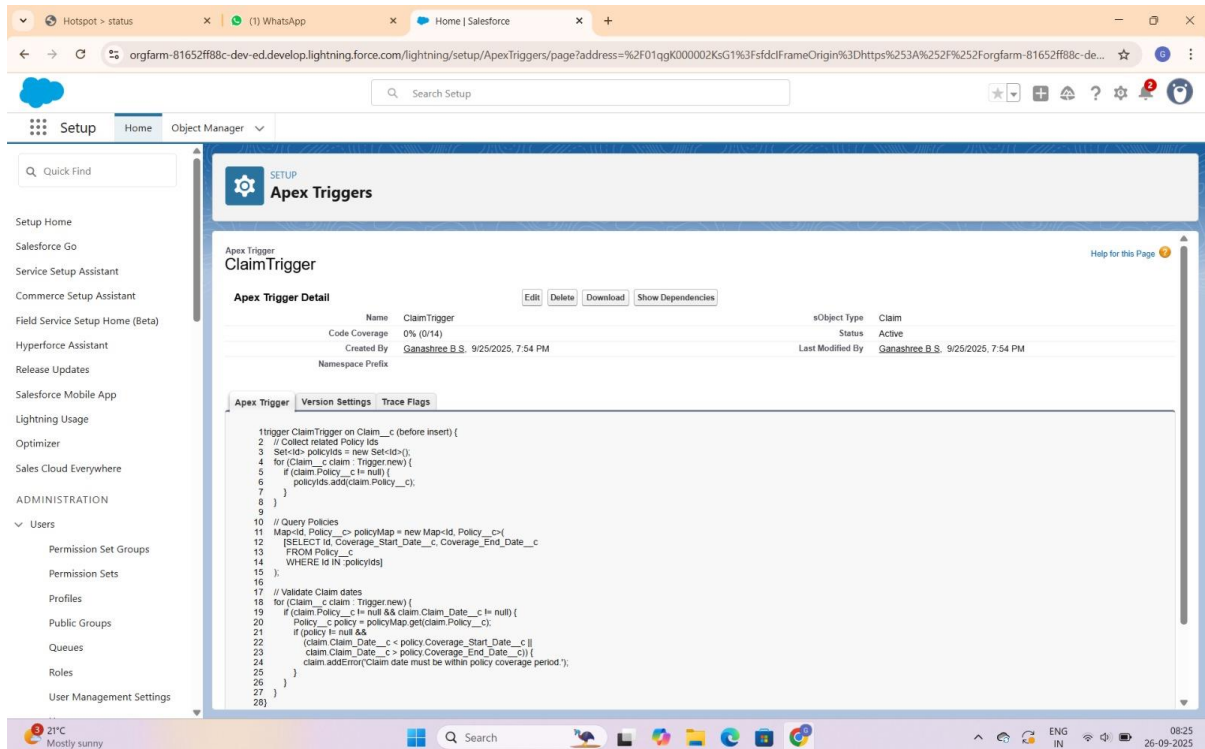
Purpose:

Custom logic was required to extend Salesforce's standard functionality. Apex programming allowed automation of tasks not possible with declarative tools.

Implementation:

1. Triggers:

- Before Insert: Ensured claim date cannot be in the future.
- After Insert: Generated unique claim reference numbers.



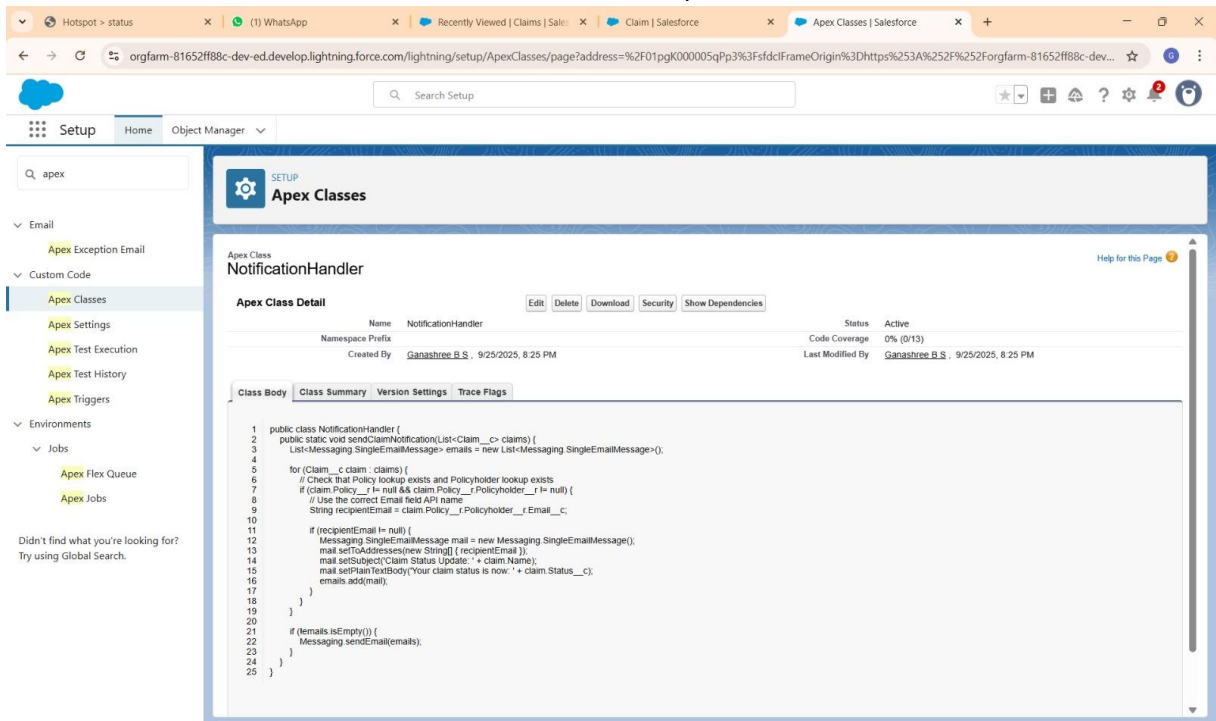
The screenshot displays the Salesforce Setup interface for Apex Triggers. The left sidebar shows the Setup menu with options like Setup Home, Salesforce Go, Service Setup Assistant, Commerce Setup Assistant, Field Service Setup Home (Beta), Hyperforce Assistant, Release Updates, Salesforce Mobile App, Lightning Usage, Optimizer, Sales Cloud Everywhere, and ADMINISTRATION. The main content area is titled 'Apex Triggers' and shows the 'Apex Trigger Detail' for 'ClaimTrigger'. The trigger is active, created by Ganashree B.S. on 9/25/2025 at 7:54 PM, and last modified by Ganashree B.S. on 9/25/2025 at 7:54 PM. The trigger is associated with the 'Claim' object type. The Apex Trigger code is displayed in the 'Apex Trigger' tab, showing logic for validating claim dates and generating unique claim reference numbers.

```
1 trigger ClaimTrigger on Claim__c (before insert) {
2   // Collect related Policy Ids
3   Set<Id> policyIds = new Set<Id>();
4   for (Claim__c claim : Trigger.new) {
5     if (claim.Policy__c != null) {
6       policyIds.add(claim.Policy__c);
7     }
8   }
9   // Query Policies
10  Map<Id, Policy__c> policyMap = new Map<Id, Policy__c>();
11  (SELECT Id, Coverage_Start_Date__c, Coverage_End_Date__c
12   FROM Policy__c
13   WHERE Id IN :policyIds)
14  }
15  // Validate Claim dates
16  for (Claim__c claim : Trigger.new) {
17    if (claim.Policy__c != null && claim.Claim_Date__c != null) {
18      Policy__c policy = policyMap.get(claim.Policy__c);
19      if (policy != null &&
20          (claim.Claim_Date__c < policy.Coverage_Start_Date__c ||
21           claim.Claim_Date__c > policy.Coverage_End_Date__c)) {
22        claim.addError('Claim date must be within policy coverage period');
23      }
24    }
25  }
26 }
27 }
28 }
```

2. Apex Classes:

- ClaimService.cls: Managed claim validation and calculations.

- NotificationHandler.cls: Sent notifications via email/SMS.



3. Batch Apex:

- Scheduled batch class to generate monthly pending claims report.

4. Test Classes:

- Created unit tests with 75%+ coverage.
- Validated logic for triggers, classes, and batch jobs.

Business Impact:

Apex extended Salesforce's power by providing advanced validations and reporting features. Managers gained automated monthly insights, and policyholders benefited from instant notifications.

Testing/Verification:

Ran unit tests in Developer Console with successful code coverage. Deployed triggers and classes to Sandbox for further validation.

Completion Status: Apex components successfully developed, tested, and deployed.