
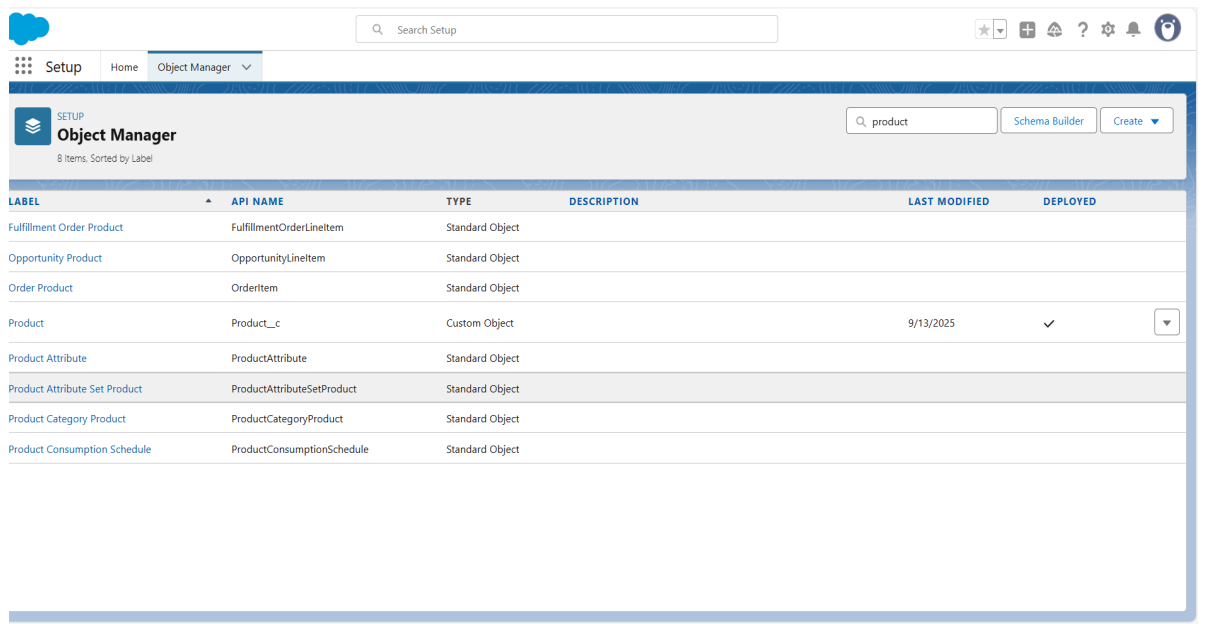


# Phase 3: Data Modeling & Relationships

## Step 1: Standard & Custom Objects

- Confirm in **Object Manager** that you have:
  - Product\_\_c
  - Warranty\_\_c
  - Service\_Request\_\_c
- (You already created these .



Search Setup

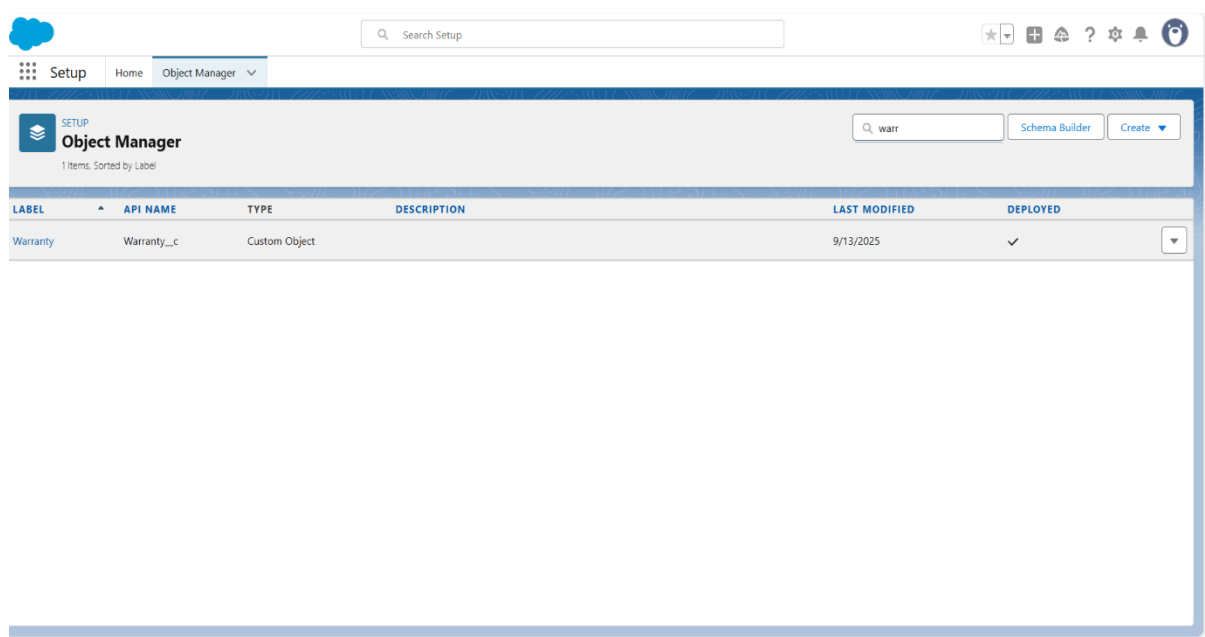
Setup Home Object Manager

Object Manager

8 Items, Sorted by Label

product Schema Builder Create

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Fulfillment Order Product	FulfillmentOrderLineItem	Standard Object			
Opportunity Product	OpportunityLineItem	Standard Object			
Order Product	OrderItem	Standard Object			
Product	Product__c	Custom Object		9/13/2025	✓
Product Attribute	ProductAttribute	Standard Object			
Product Attribute Set Product	ProductAttributeSetProduct	Standard Object			
Product Category Product	ProductCategoryProduct	Standard Object			
Product Consumption Schedule	ProductConsumptionSchedule	Standard Object			



Search Setup

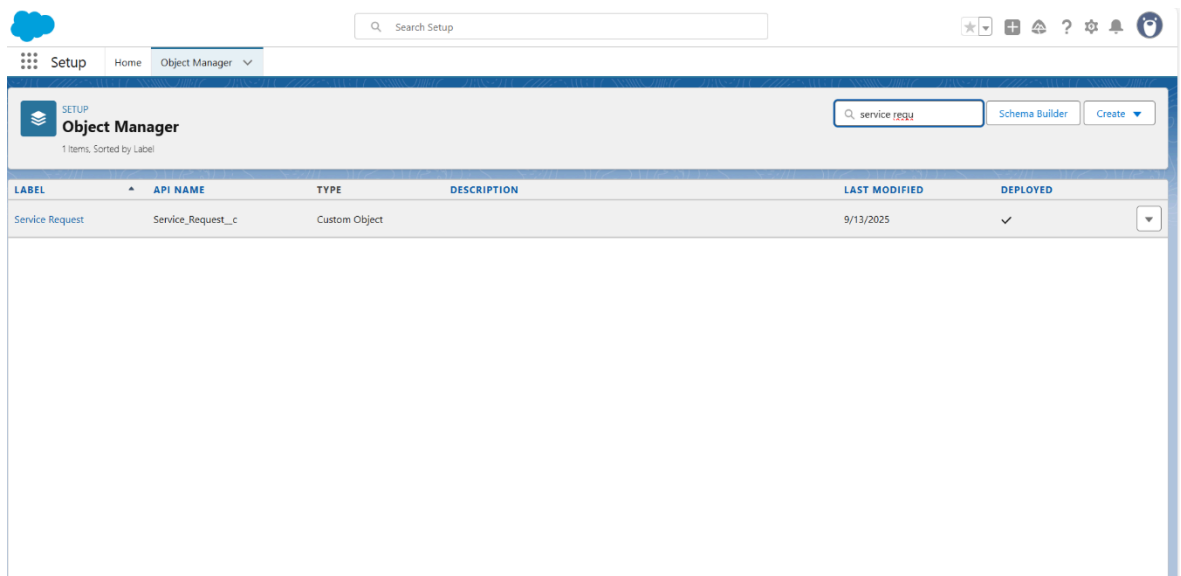
Setup Home Object Manager

Object Manager

1 Items, Sorted by Label

warr Schema Builder Create

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Warranty	Warranty__c	Custom Object		9/13/2025	✓

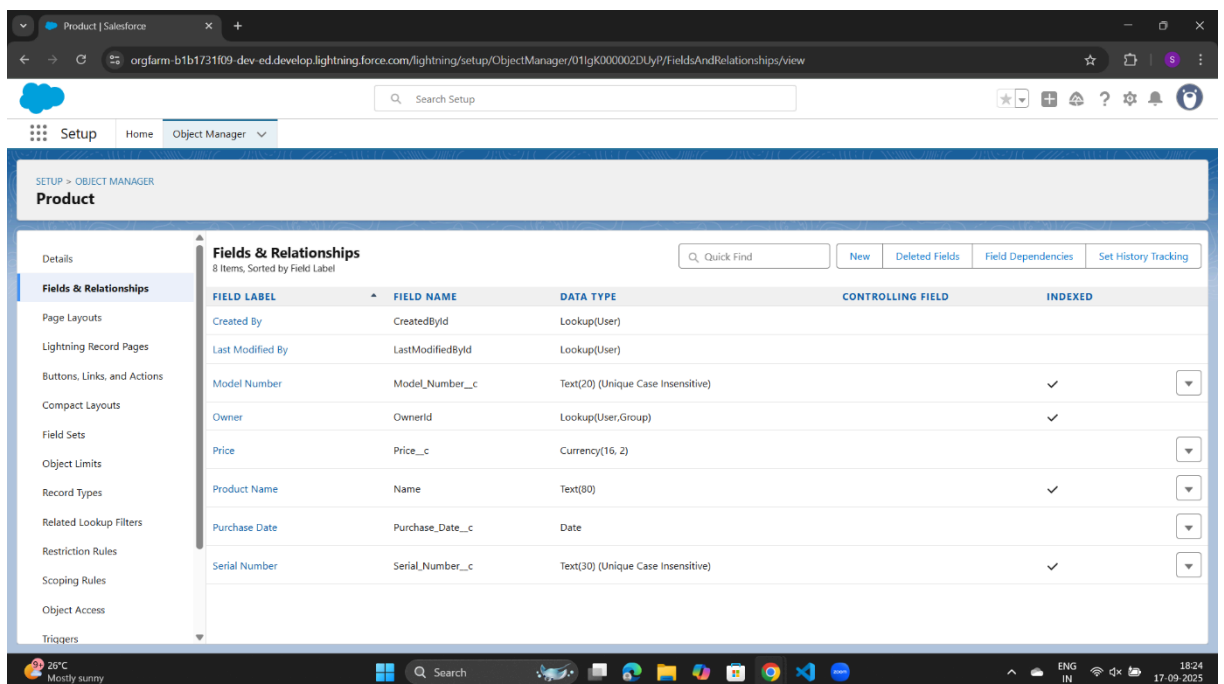


## ❖ Step 2: Fields

Create fields for each object.

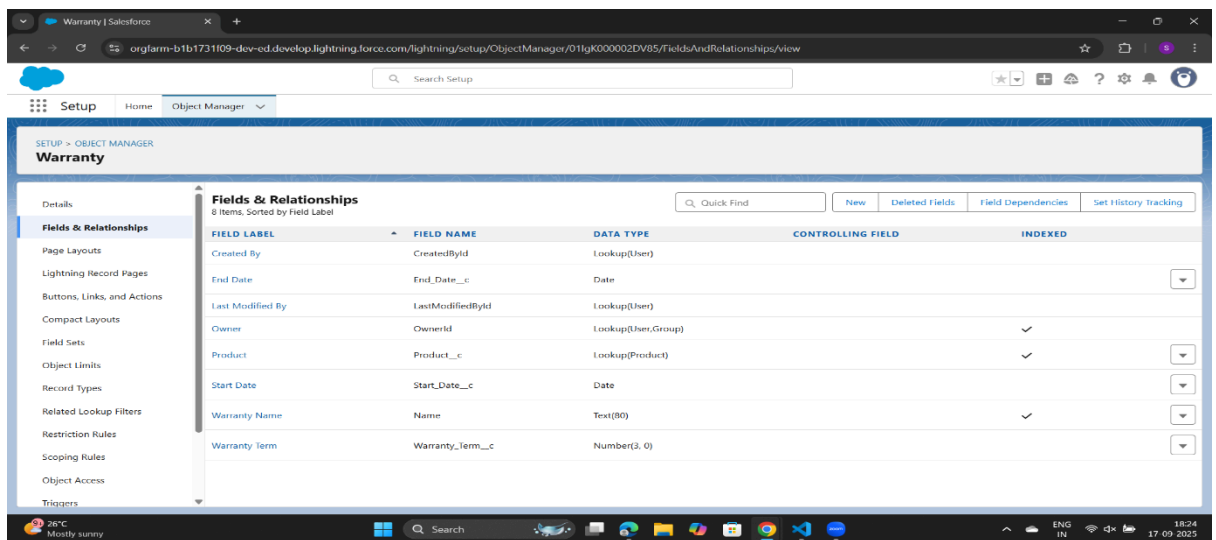
### Product\_\_c

- Serial Number → Text(30), Unique ✓
- Model Number → Text(20), Required ✓
- Price → Currency(16,2), Required ✓
- Purchase Date → Date, Required ✓



## Warranty\_\_c

- Start Date → Date, Required ✓
- End Date → Date, Required ✓
- Warranty Term (Months) → Number(3,0), Required ✓
- Related Product → Lookup(Product\_\_c), Required ✓

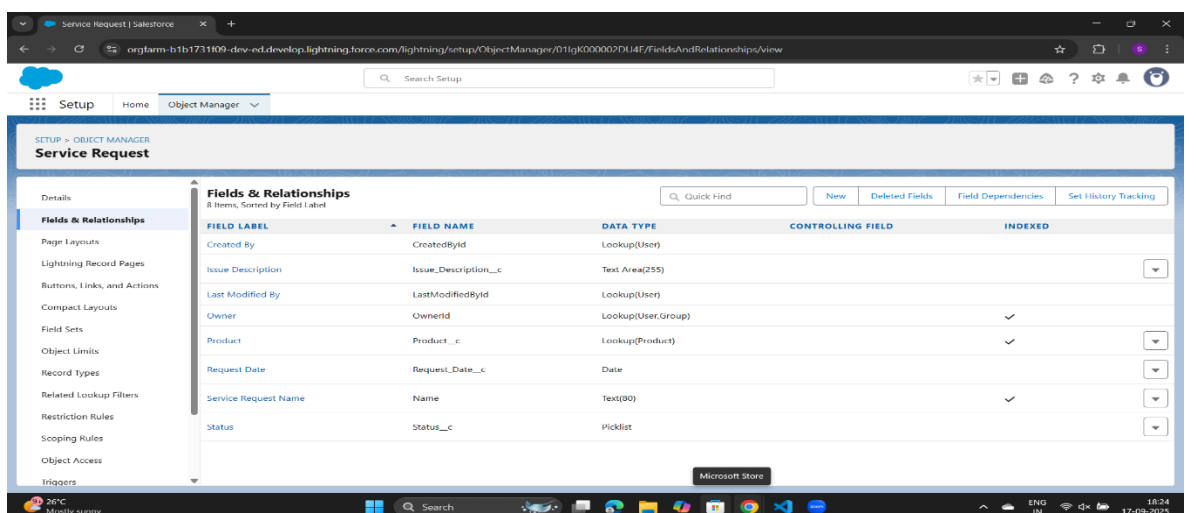


The screenshot shows the Salesforce Setup interface for the 'Warranty' object. The 'Fields & Relationships' section is active, displaying a table of 8 fields. The table has columns for Field Label, Field Name, Data Type, Controlling Field, and Indexed. The fields listed are: Created By (Lookup(User)), End Date (Date), Last Modified By (Lookup(User)), Owner (Lookup(User, Group)), Product (Lookup(Product)), Start Date (Date), Warranty Name (Text(80)), and Warranty Term (Number(3, 0)).

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
End Date	End_Date__c	Date		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User, Group)		✓
Product	Product__c	Lookup(Product)		✓
Start Date	Start_Date__c	Date		
Warranty Name	Name	Text(80)		✓
Warranty Term	Warranty_Term__c	Number(3, 0)		

## Service\_Request\_\_c

- Request Date → Date, Default = TODAY(), Required ✓
- Status → Picklist (New, In Progress, Completed, Closed), Default = New
- Issue Description → Long Text Area (500)
- Related Product → Lookup(Product\_\_c), Required ✓
- Related Warranty → Lookup(Warranty\_\_c), Required ✗



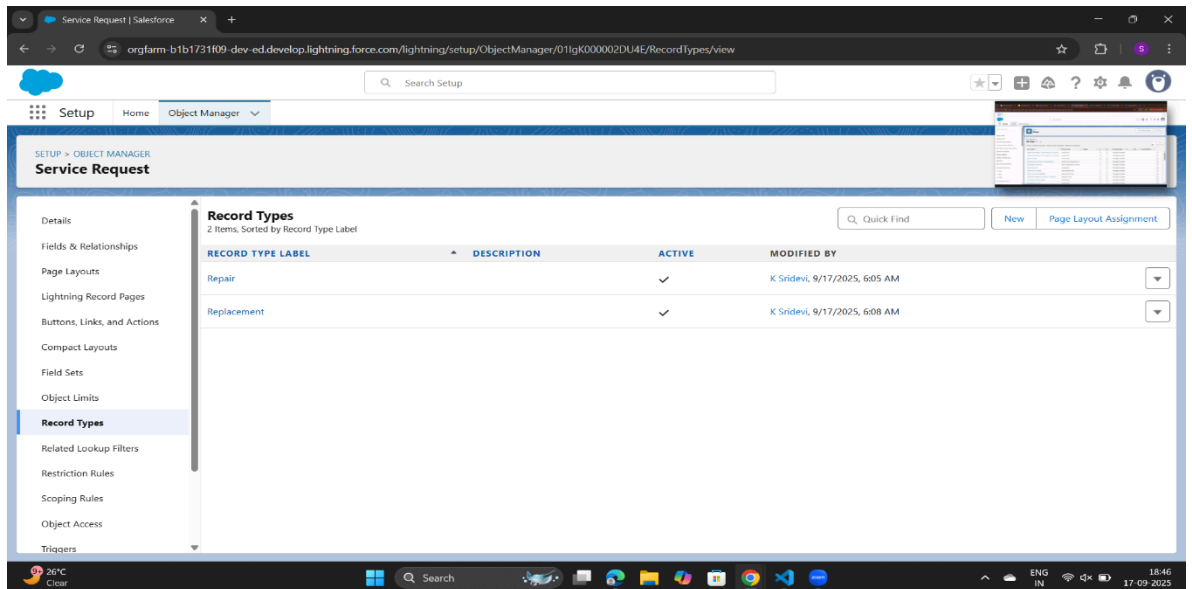
The screenshot shows the Salesforce Setup interface for the 'Service Request' object. The 'Fields & Relationships' section is active, displaying a table of 8 fields. The table has columns for Field Label, Field Name, Data Type, Controlling Field, and Indexed. The fields listed are: Created By (Lookup(User)), Issue Description (Text Area(255)), Last Modified By (Lookup(User)), Owner (Lookup(User, Group)), Product (Lookup(Product)), Request Date (Date), Service Request Name (Text(90)), and Status (Picklist).

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Issue Description	Issue_Description__c	Text Area(255)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User, Group)		✓
Product	Product__c	Lookup(Product)		✓
Request Date	Request_Date__c	Date		
Service Request Name	Name	Text(90)		✓
Status	Status__c	Picklist		

## ❖ Step 3: Record Types

(Do this only for Service\_Request\_\_c).

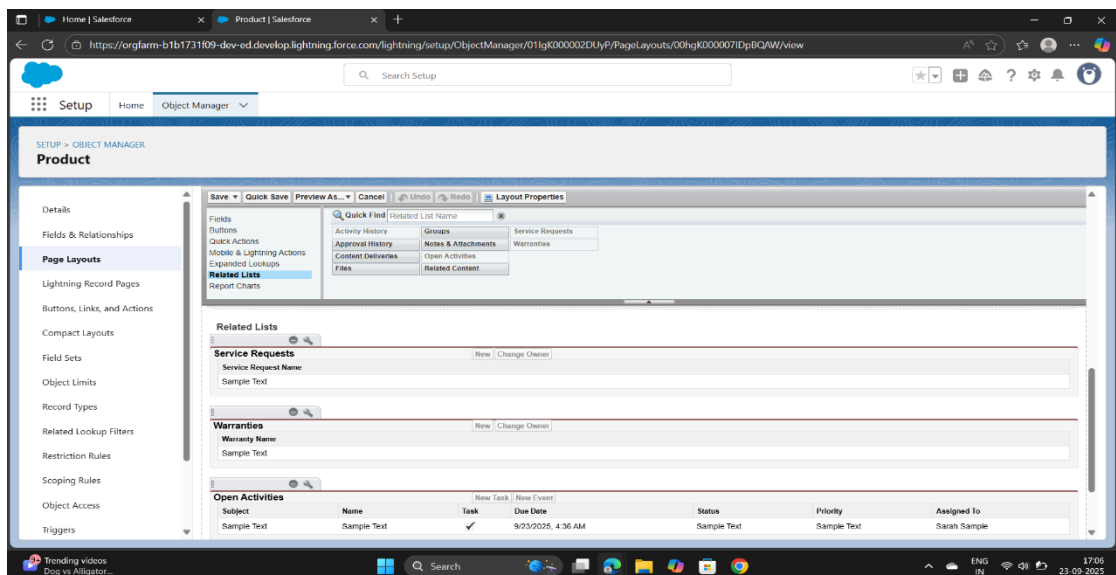
- Create **Repair** record type.
- Create **Replacement** record type.
- Assign to all profiles.
- For both, use the *Service Request Layout* (for now).



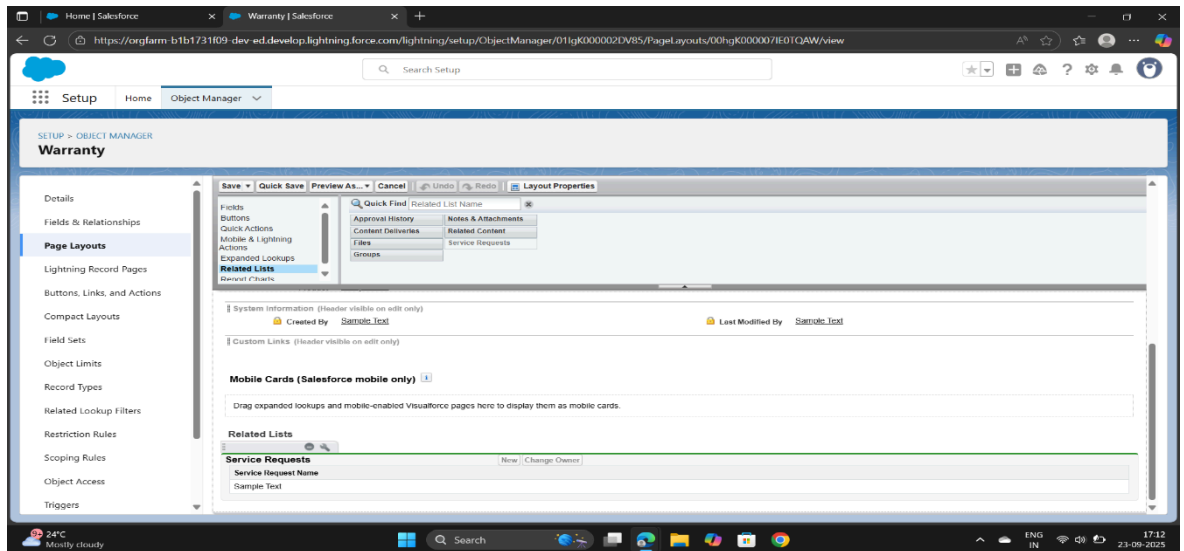
#### ❖ Step 4: Page Layouts

Customize layouts so relationships are visible.

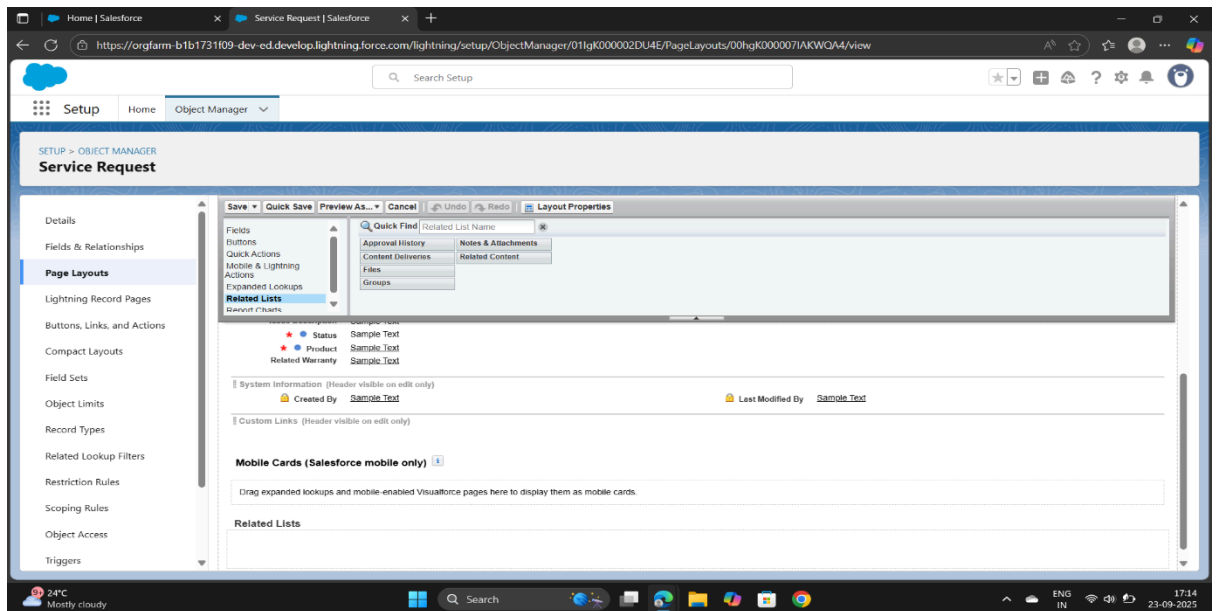
- **Product\_\_c Layout** → add Related Lists: Warranties, Service Requests.



- **Warranty\_\_c Layout** → add Related Product + Service Requests.

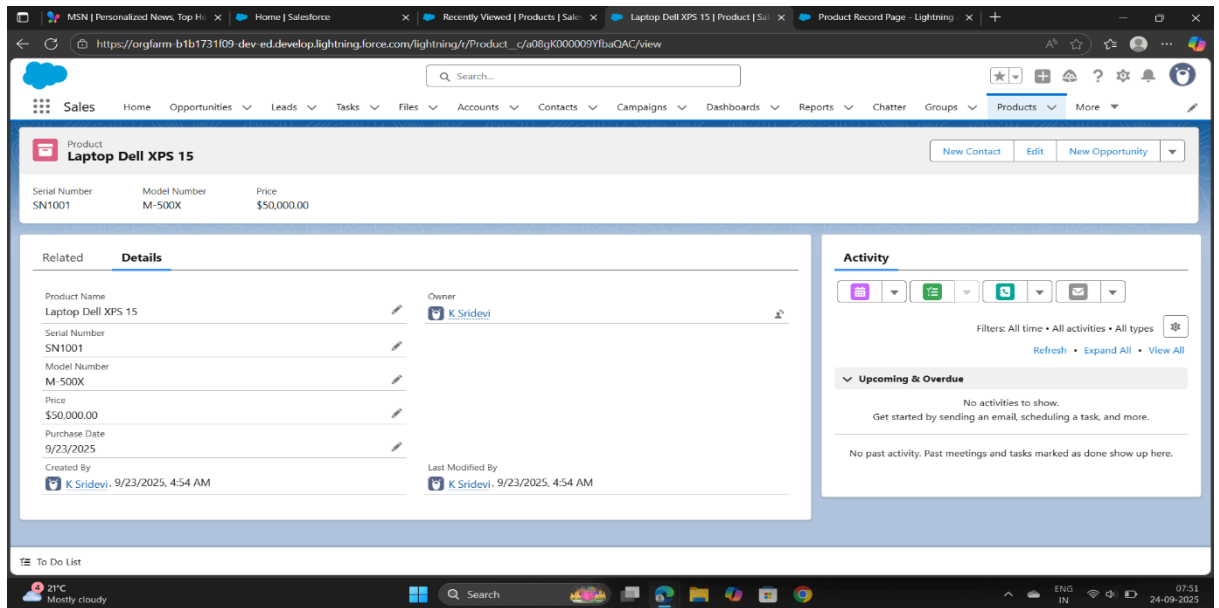


- **Service\_Request\_\_c Layout** → add Related Product + Related Warranty.



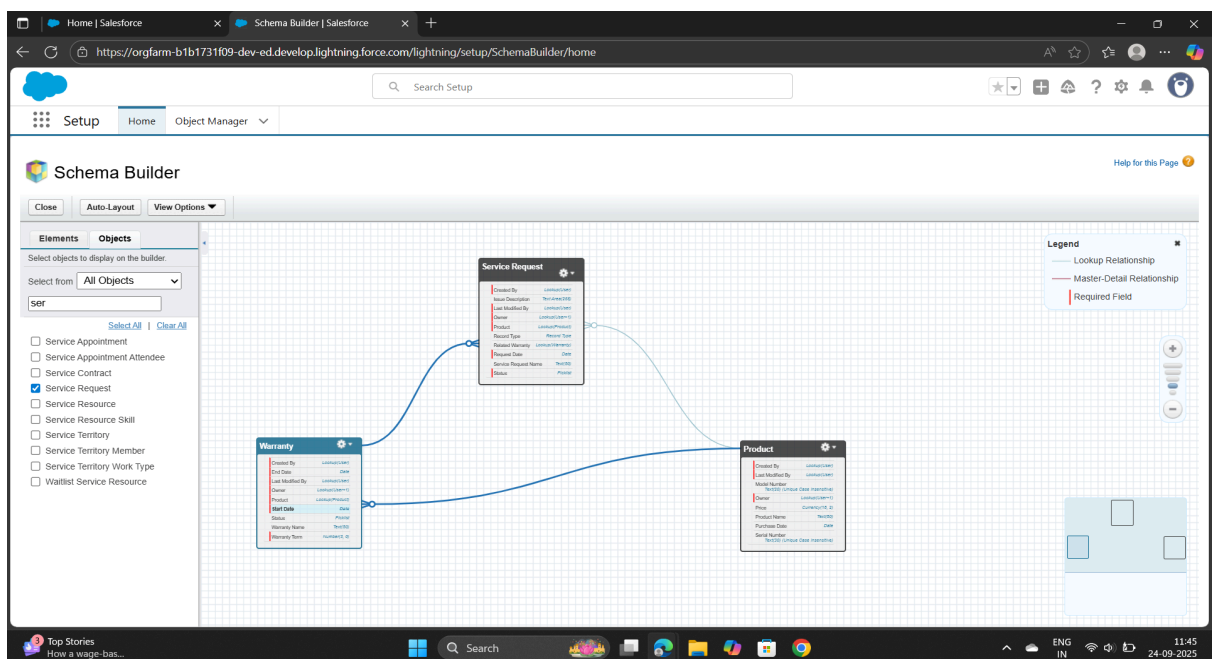
### ❖ Step 5: Compact Layouts

- For each object, set compact layout fields (these show in Highlights Panel).
- Example:
  - o Product: Serial Number, Model Number, Price
  - o Warranty: Start Date, End Date, Product
  - o Service Request: Status, Request Date, Product



## ❖ Step 6: Schema Builder

- Go to **Schema Builder**.
- Select your 3 objects (Product, Warranty, Service Request).
- See the relationship lines.



## ❖ Step 7: Relationships

- Ensure relationships are correct:
  - Product → Warranty (Lookup)
  - Product → Service Request (Lookup)

- o Warranty → Service Request (Lookup)

#### ❖ Step 8 — Junction Objects

A Junction Object in Salesforce is a custom object with two Master-Detail relationships used to model a many-to-many relationship between two objects. For example, if a Service Request could apply to multiple Products, and each Product could be linked to multiple Service Requests, then we would create a junction object called `Product_Service_Request__c` with:

- Master-Detail to `Product__c`
- Master-Detail to `Service_Request__c`
- In our Warranty & Service Tracker project, we do not require junction objects because our relationships are **one-to-many**.
- A Product can have many Warranties and Service Requests.
- But each Warranty and Service Request belongs to only one Product.

Hence, junction objects were not created but are explained here for completeness.

#### ❖ Step 9 — Hierarchical Relationships

Salesforce also supports a special type of relationship called **Hierarchical Relationship**, which is only available on the User object.

It allows users to be related in a parent-child way, such as showing reporting structures (e.g., Manager → Agent).

In our Warranty & Service Tracker project, hierarchical relationships are **not required** because we are focusing on Products, Warranties, and Service Requests. However, hierarchical relationships are often used in real-life scenarios for approval flows or reporting managers.