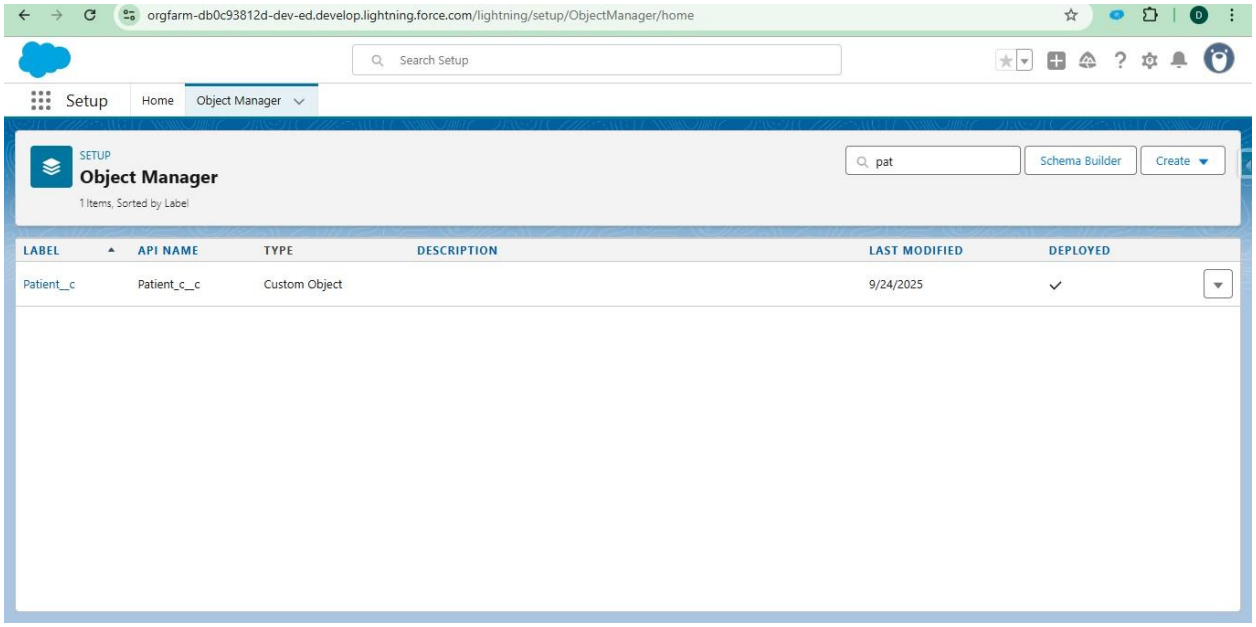
Phase 3: Data Modeling & Relationships

Standard & Custom Objects

For effective patient management, Salesforce standard objects are complemented with custom objects tailored to hospital workflows. The following custom objects were created: **Patient__c, Appointment__c, Treatment__c, Lab_Result__c, and Care_Plan__c**. These objects ensure that patient information, appointments, treatments, and care plans are stored in a structured, healthcare-specific format.



Fields

Each custom object contains fields designed to capture essential hospital data.

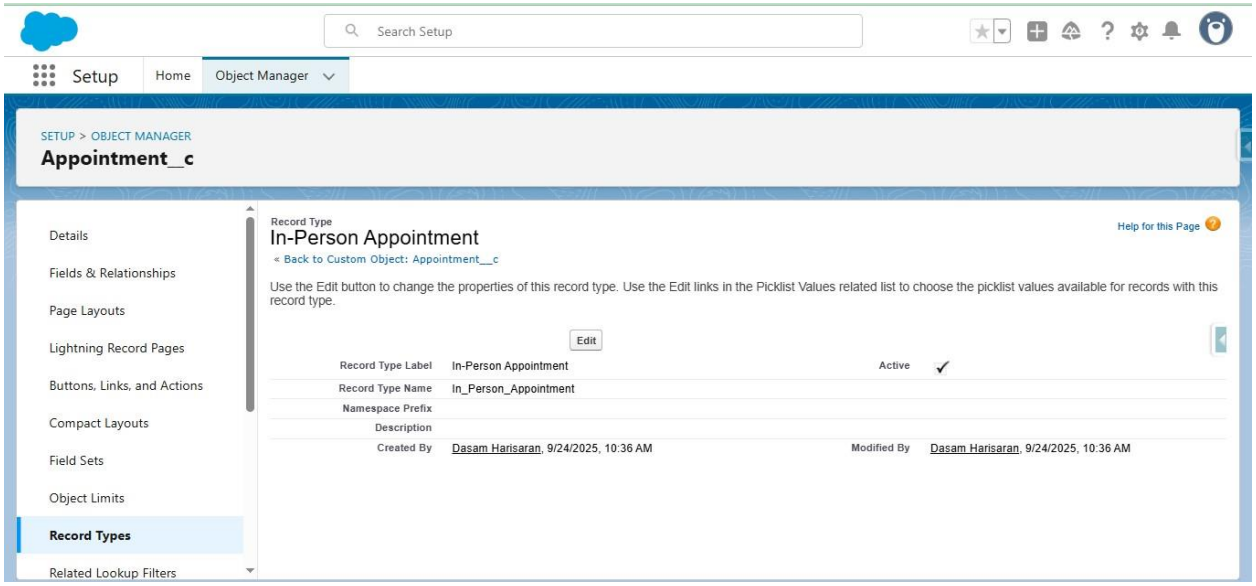
- **Patient__c** → Patient ID, Name, Age, Department, Email, Phone.
- **Appointment__c** → Date, Time, Doctor, linked Patient.
- **Treatment__c** → Diagnosis, Medication, Therapy, linked Patient.
- **Lab_Result__c** → Test Name, Date, Result, linked Patient.
- **Care_Plan__c** → Plan Name, Follow-up Date, Assigned Doctor, linked Patient.

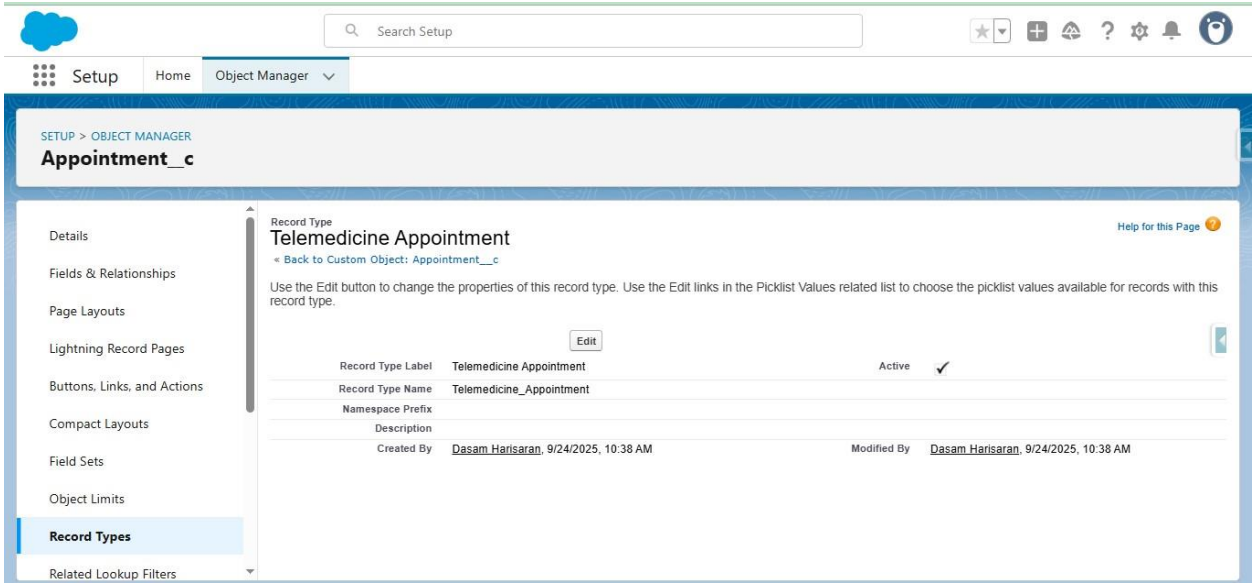
Record Types

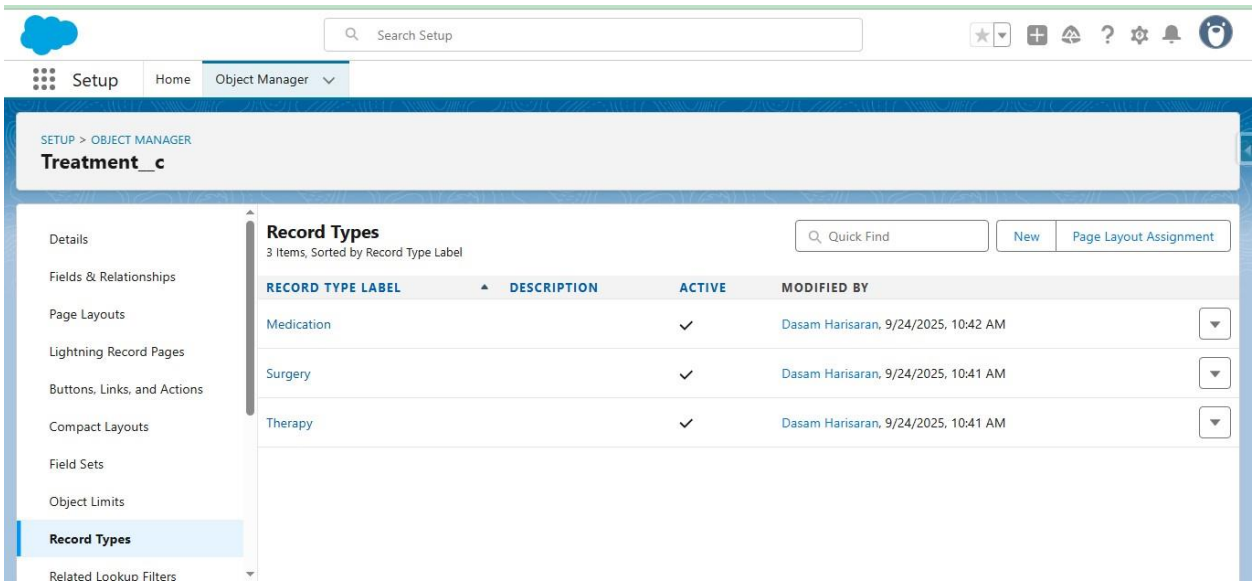
Record types allow differentiation between healthcare services.

- **Appointment__c** → In-Person, Telemedicine.
- **Treatment__c** → Surgery, Medication, Therapy.

This ensures flexibility in categorizing healthcare encounters.





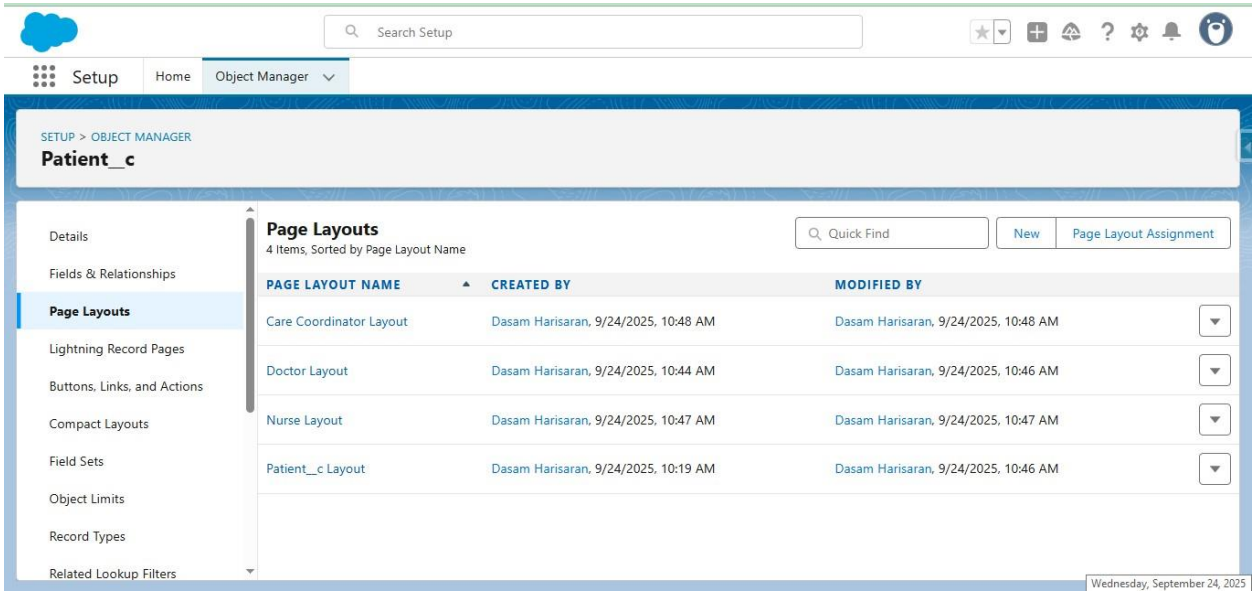


Page Layouts

Custom page layouts were developed for different roles.

- **Doctors:** Treatments and Appointments.
- **Nurses:** Patient details and Medication schedules.
- **Care Coordinators:** Care Plans and Follow-ups.
- **Patients:** Their own Appointments and Lab Results.

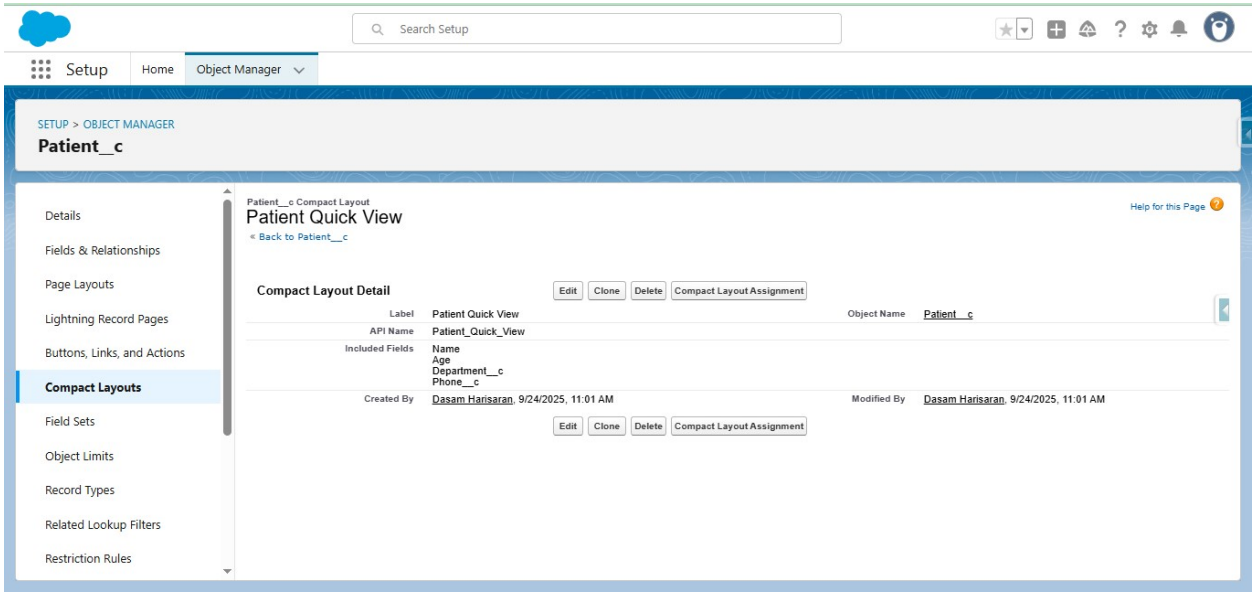
This role-based design ensures relevant data visibility.



Compact Layouts

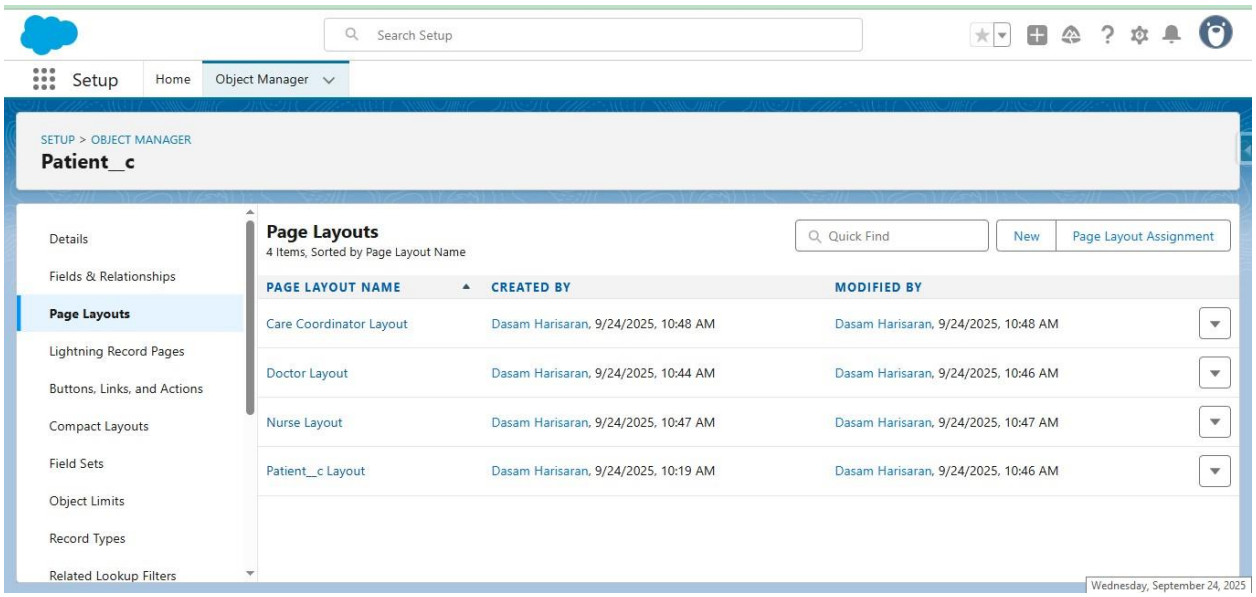
Compact layouts provide quick previews of essential fields.

- **Patient__c** → Name, Age, Department, Phone.
- **Appointment__c** → Date, Doctor, Status.



Schema Builder

The Schema Builder was used to visualize the entire healthcare data model. It highlights how **Patient__c** is the central object connected with Appointments, Treatments, Lab Results, and Care Plans, providing a clear understanding of data flow.



Lookup vs Master-Detail vs Hierarchical Relationships

Relationships were designed to reflect hospital workflows:

- Patient ↔ Appointment → Master-Detail
- Patient ↔ Treatment → Master-Detail
- Patient ↔ Lab_Result → Lookup
- Patient ↔ Care_Plan → Lookup

Additionally, a junction object **Doctor_Patient__c** was introduced to support many-to-many relationships between doctors and patients.

Junction Objects

Doctor_Patient__c serves as a junction object, enabling many-to-many relationships between patients and doctors. This ensures that multiple doctors can treat the same patient, and a doctor can handle multiple patients.

External Objects

To accommodate data from hospital systems such as Laboratory Information Systems (LIS), external objects were implemented. For instance, **External_Lab_Report__x** allows real-time integration with external lab data while linking results to **Patient__c**.

✅**Final Deliverables (Phase 3):**

1. Custom Objects & Fields created.
2. Record Types for healthcare scenarios.
3. Page Layouts & Compact Layouts by role.
4. Schema Builder diagram showing data model.
5. Relationships (Master-Detail, Lookup, Junction).
6. External Object for lab system integration.