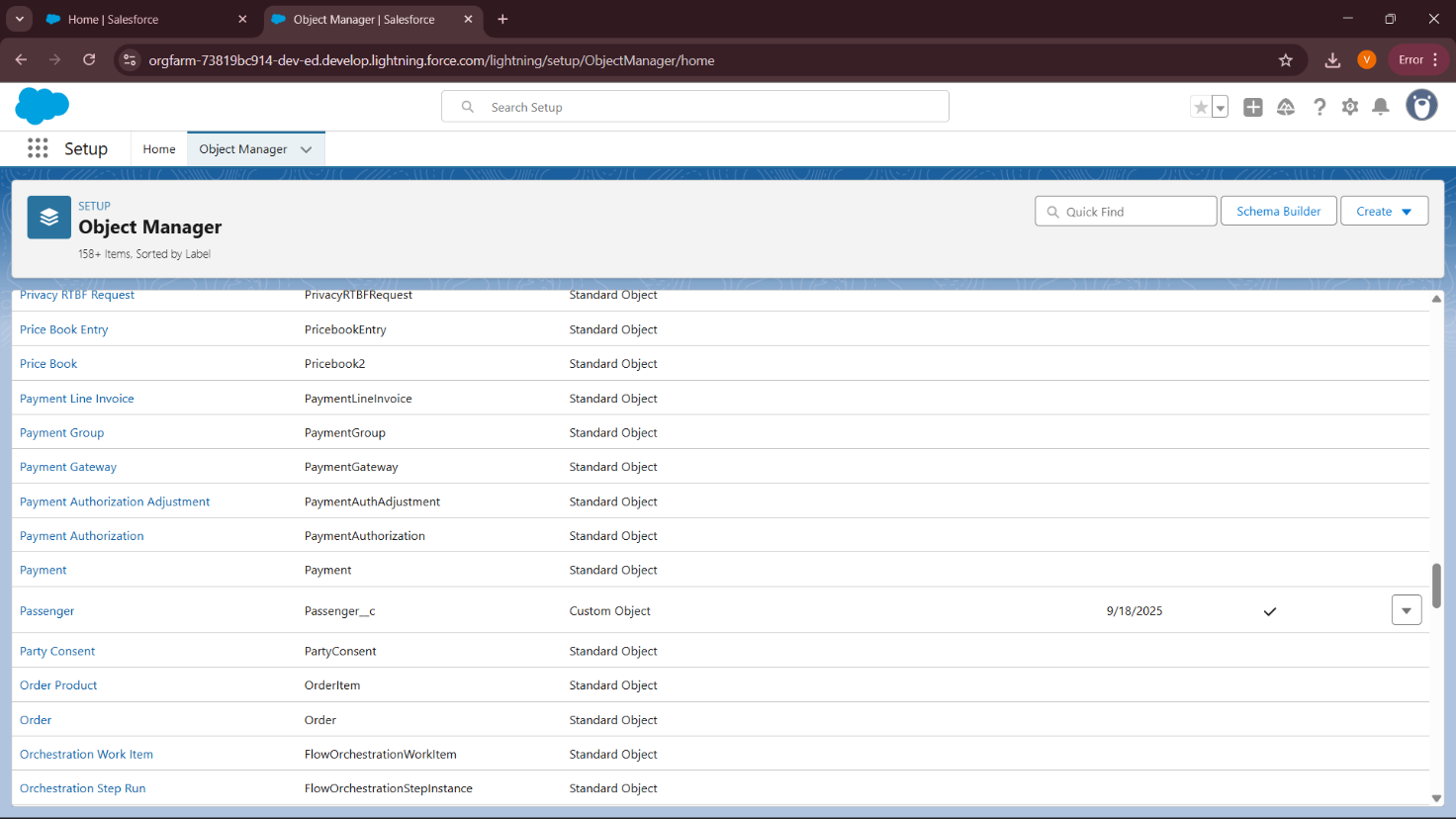
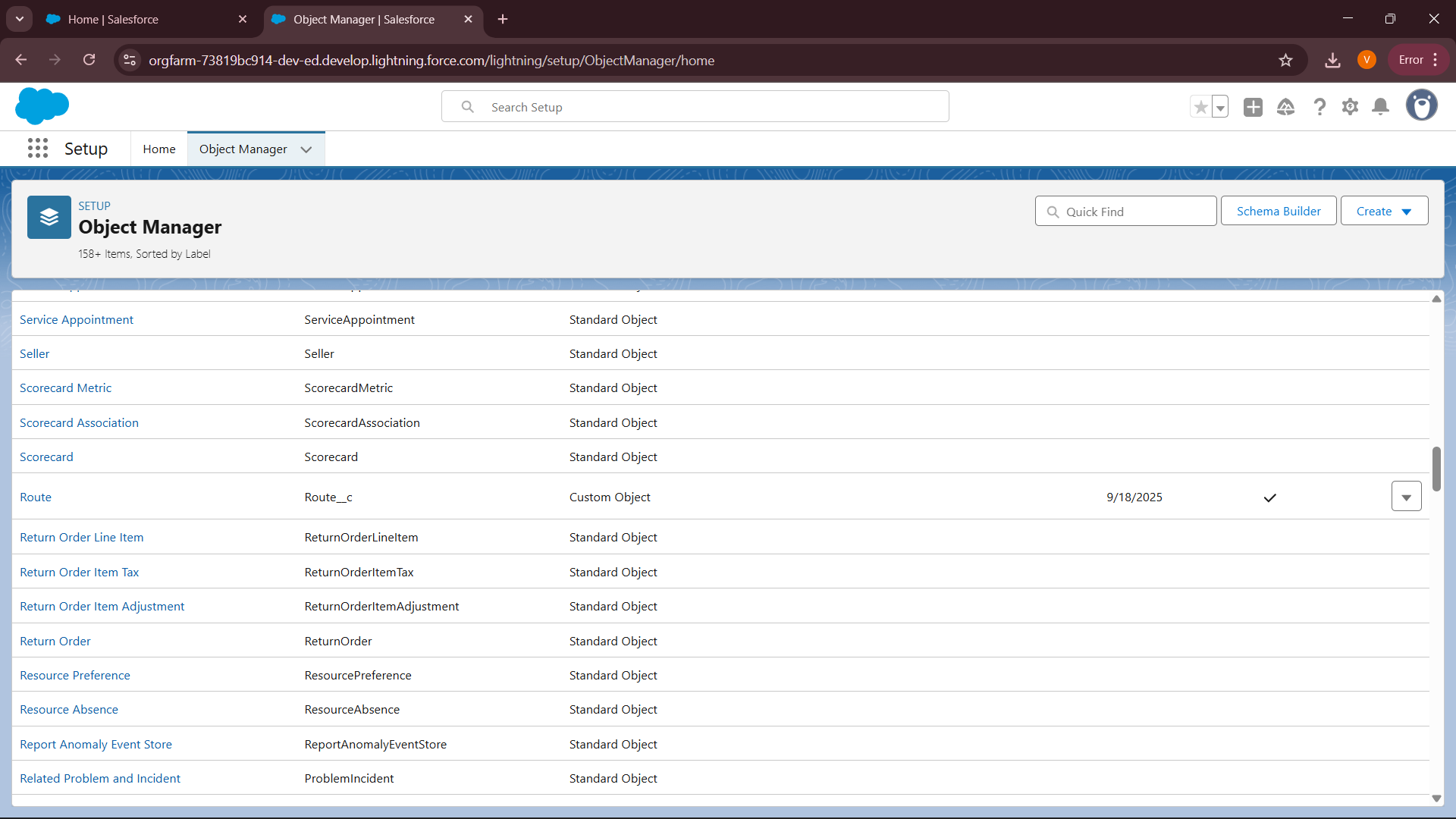
Train Ticket Booking Management System

In this phase, we design and configure the data model for the **Train Ticket Booking System**. This involves creating custom objects, fields, relationships, record types, layouts, and schema visualization.

**🔹 1. Standard & Custom Objects**

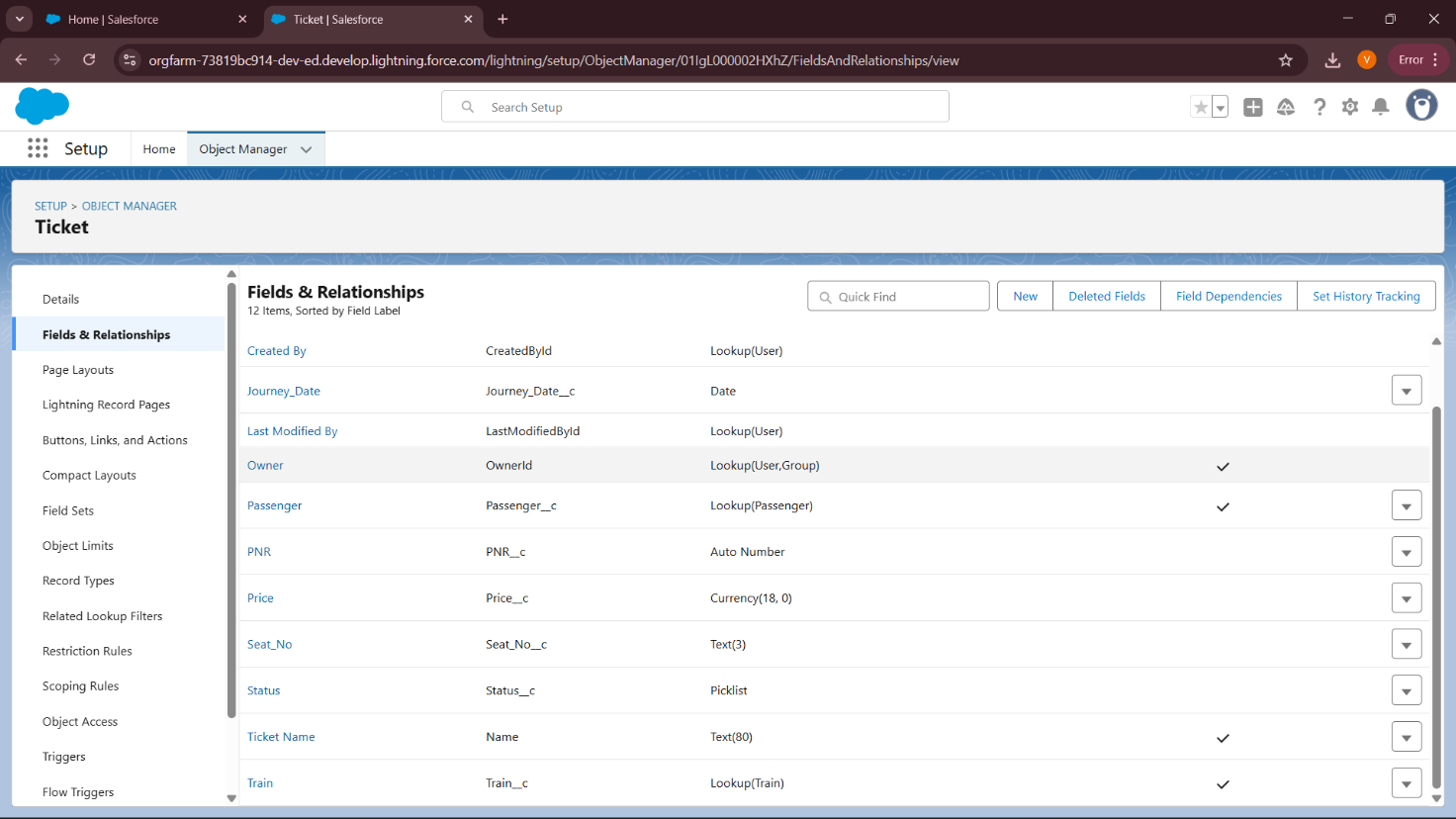
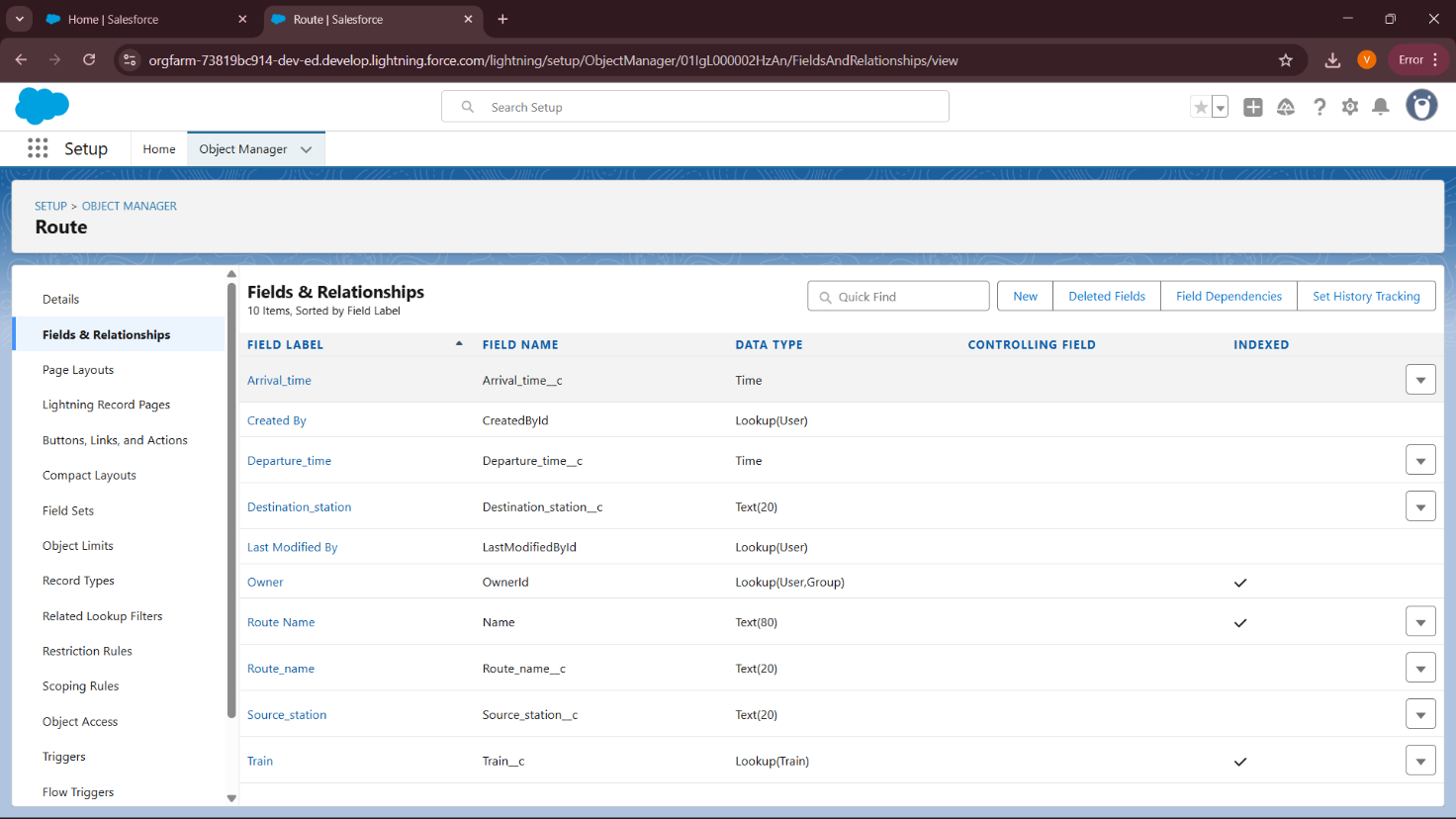
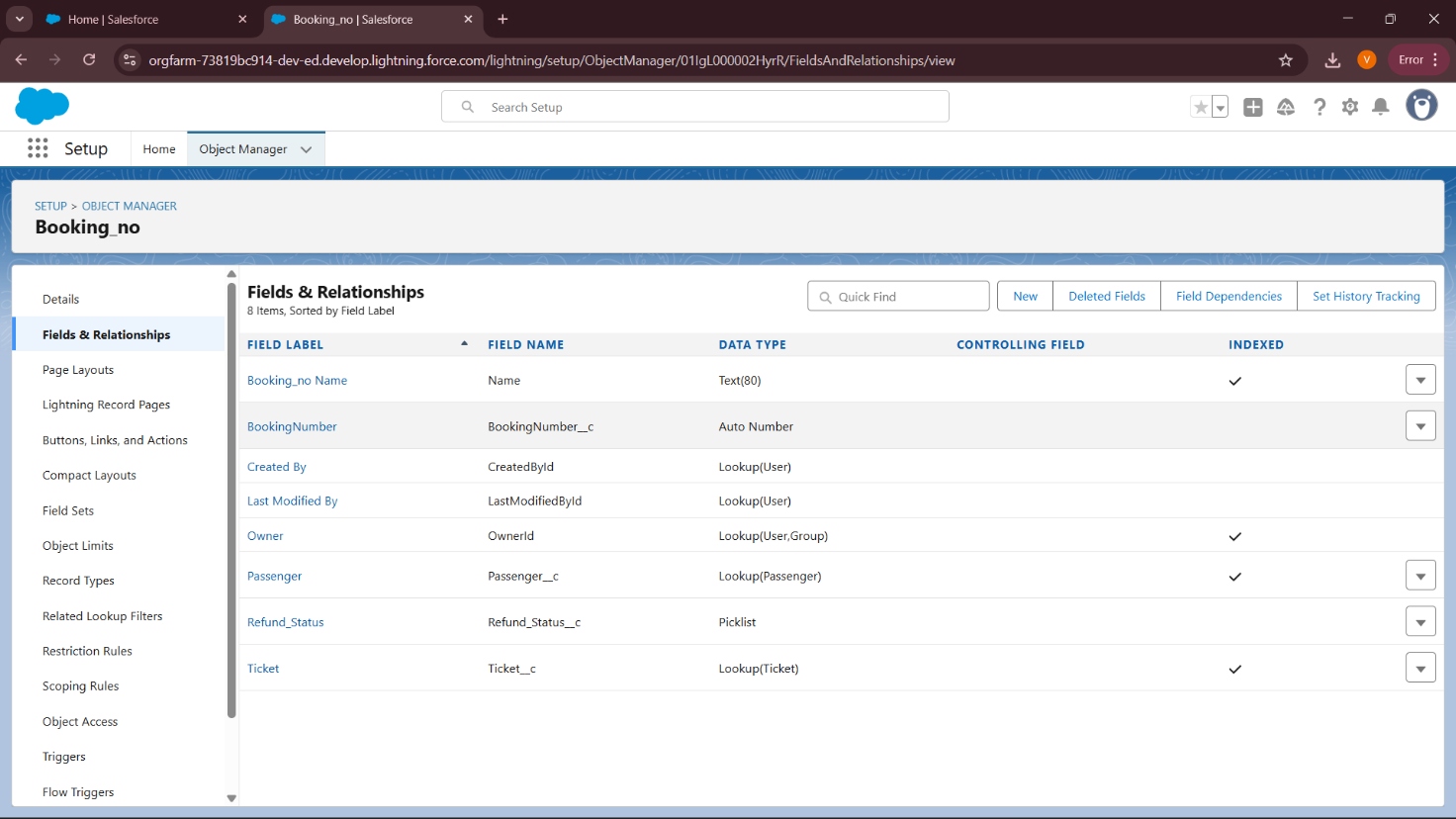
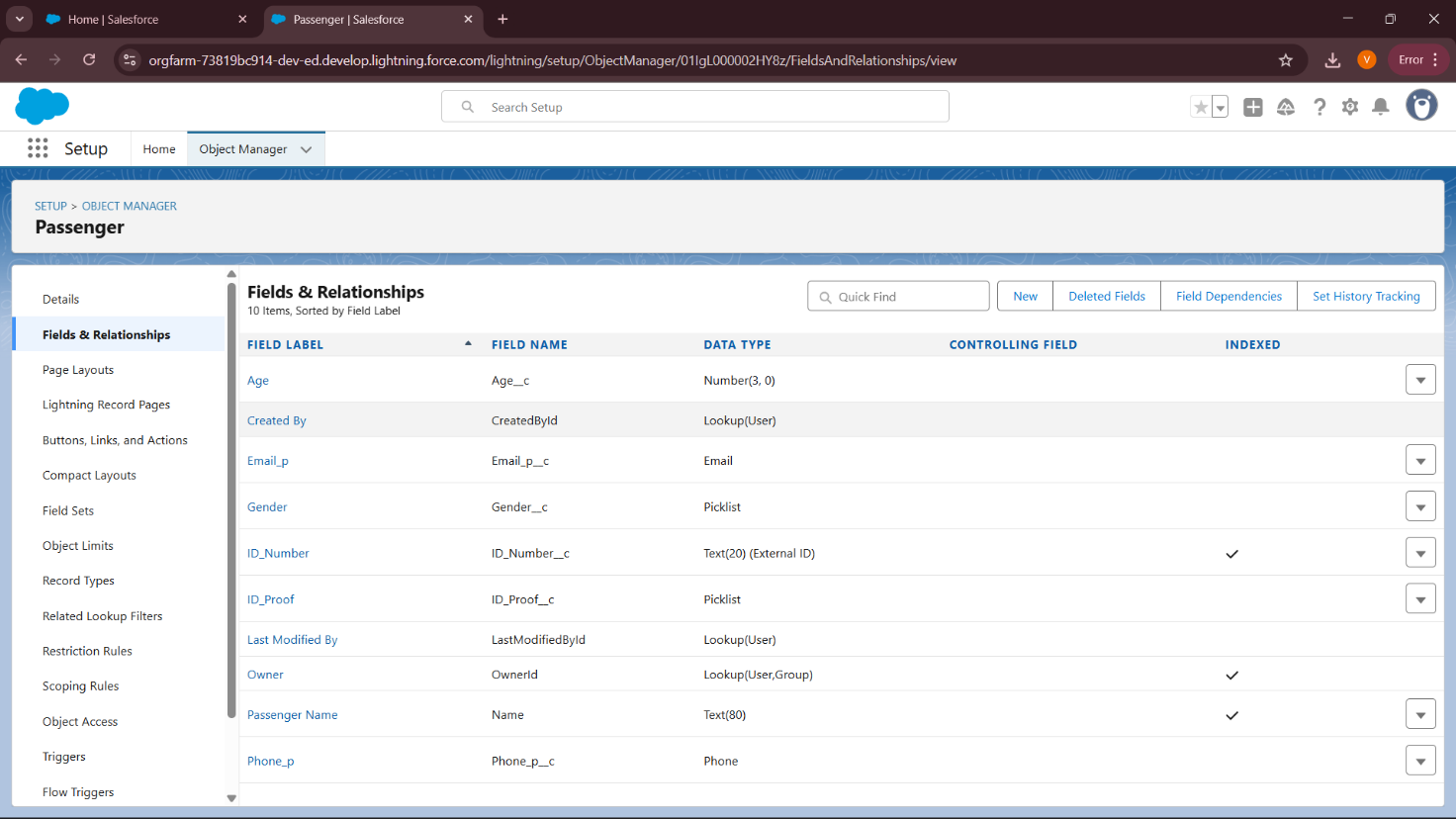
* **Standard Objects Used**: User (for Agent assignment), Contact (optional alternative to Passenger).
* **Custom Objects Created**:
  + **Train\_\_c** → Stores train details.
  + **Passenger\_\_c** → Stores passenger/customer details.
  + **Ticket\_\_c** → Stores ticket information (PNR, journey, status).
  + **Booking\_\_c** (optional junction) → Groups passenger & ticket if many-to-many is required.
  + **Route\_\_c** → Stores source, destination, and station details.



**🔹 2. Fields**

**Example Fields Created:**

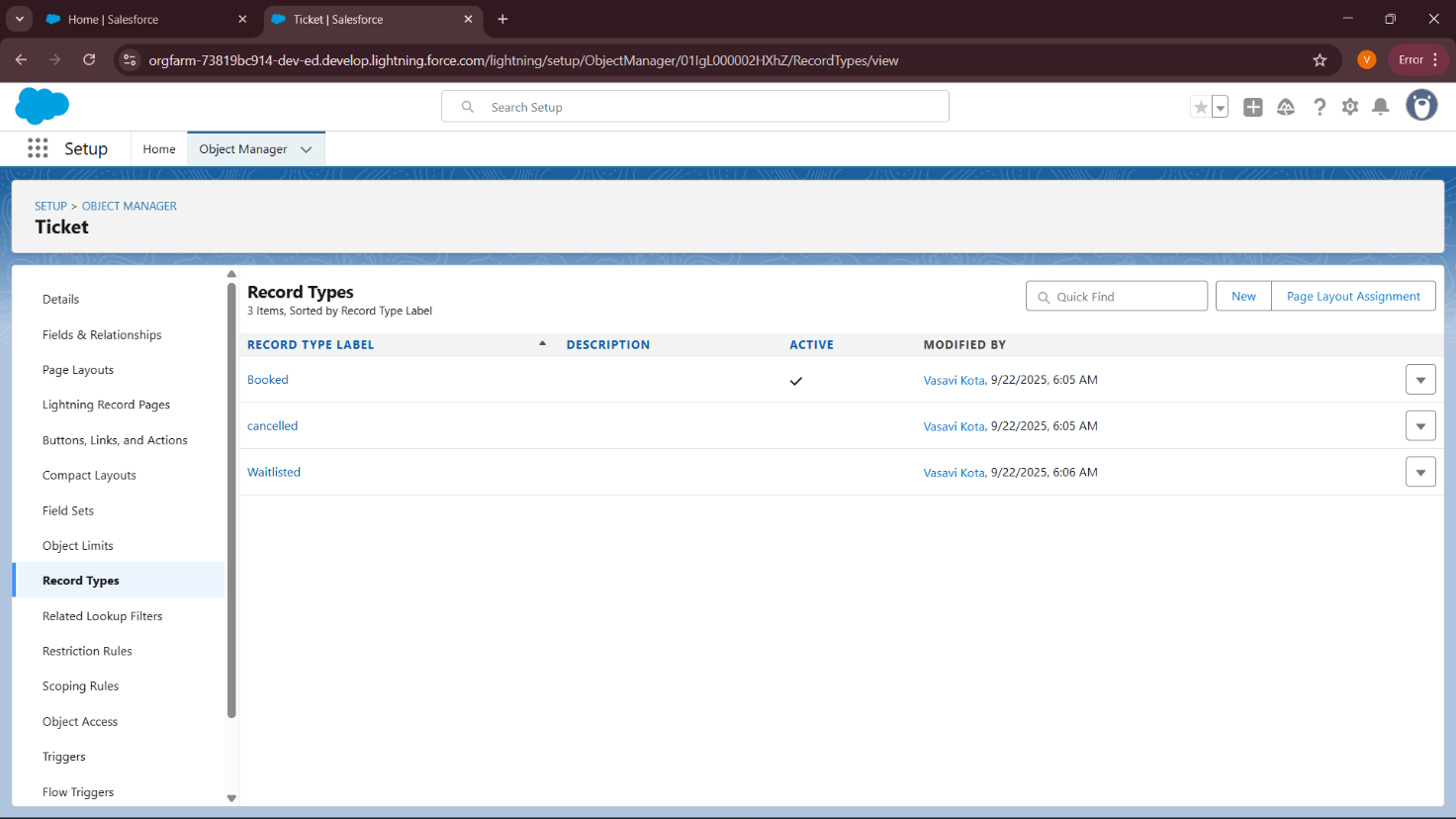
* **Train\_\_c** → Train\_No, Train\_Name, Source, Destination, Class\_Type, Total\_Seats, Available\_Seats.
* **Passenger\_\_c** → Name, Age, Gender, Email, Phone, Wallet\_Balance, ID\_Proof, ID\_Number.
* **Ticket\_\_c** → PNR (Auto Number), Journey\_Date, Status, Seat\_No, Price, Booking\_Date, Passenger (Lookup), Train (Lookup), Assigned Agent (Lookup to User), Payment\_Status, Refund\_Status.
* **Route\_\_c** → Route\_Name, Source\_Station, Destination\_Station, Station\_List, Departure\_Time, Arrival\_Time.



**🔹 3. Record Types**

Created record types on **Ticket\_\_c** to manage business processes:

* Booked
* Cancelled
* Waitlisted
* Pending Payment



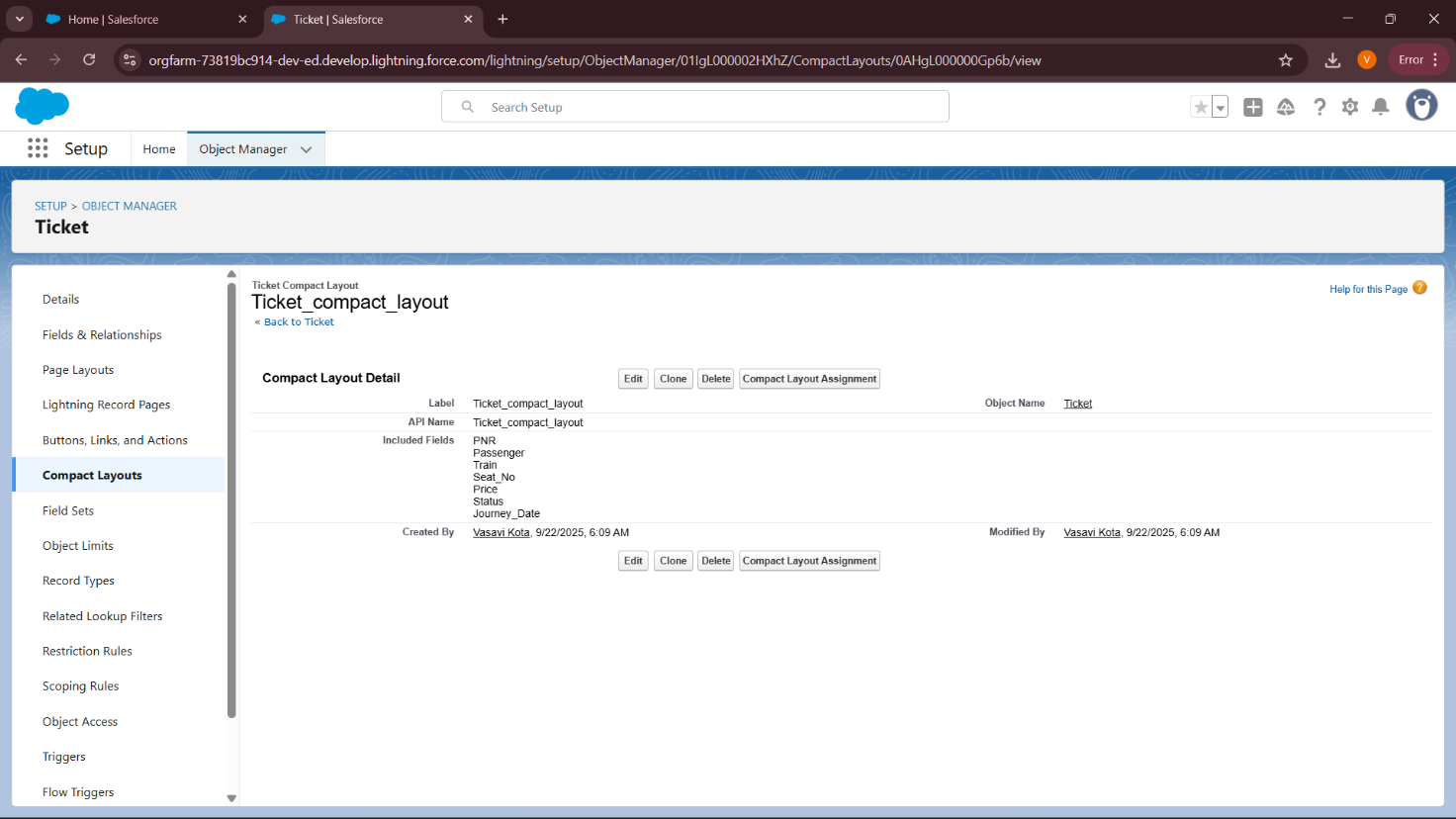
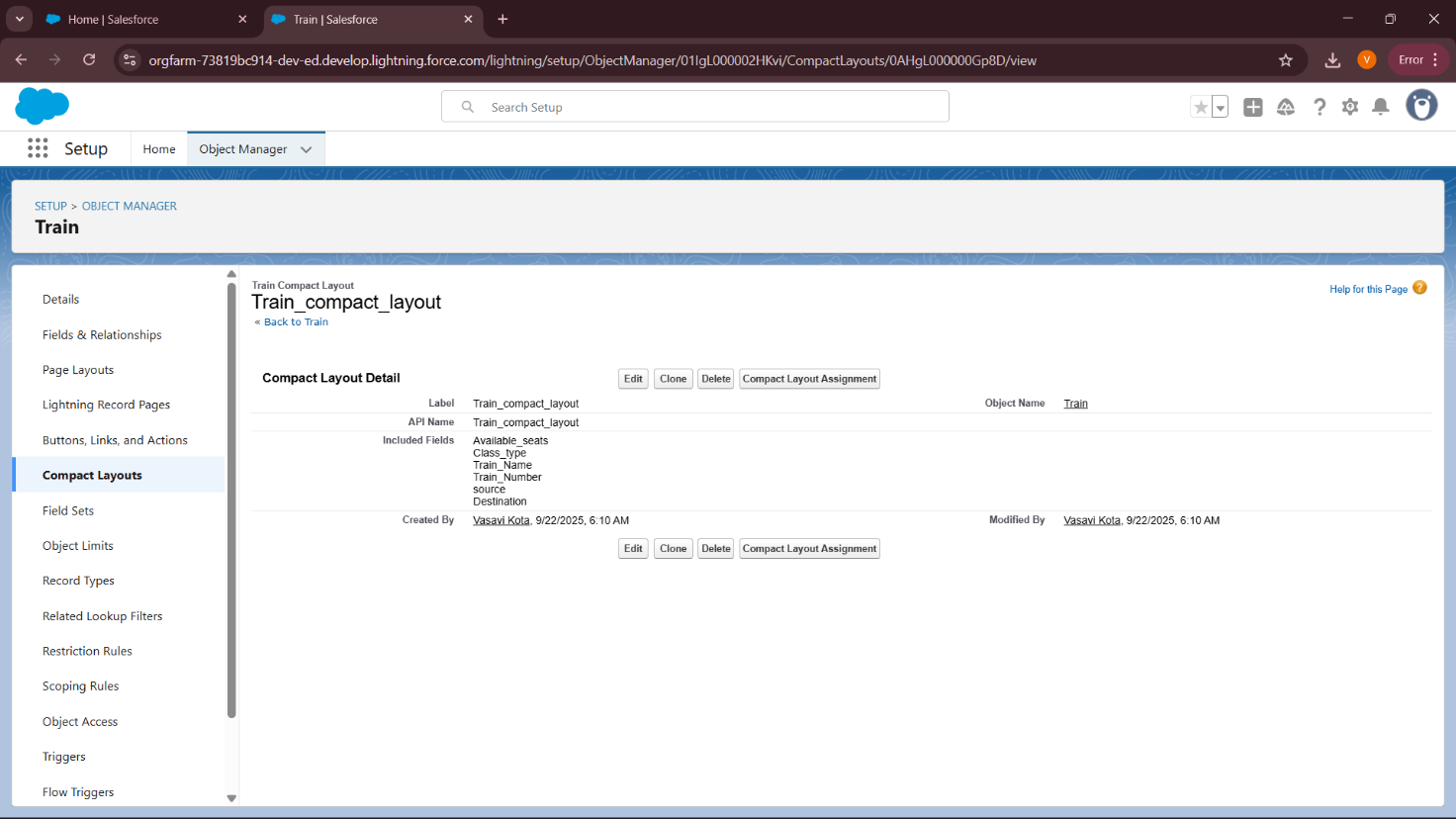
**🔹 4. Page Layouts**

* **Ticket — Passenger Layout**: Only booking details, no refund or agent fields.
* **Ticket — Agent Layout**: Includes Assigned Agent, Refund Status, Payment Status.
* **Ticket — Admin Layout**: Full access to all fields.

Layouts are assigned to profiles based on record type.

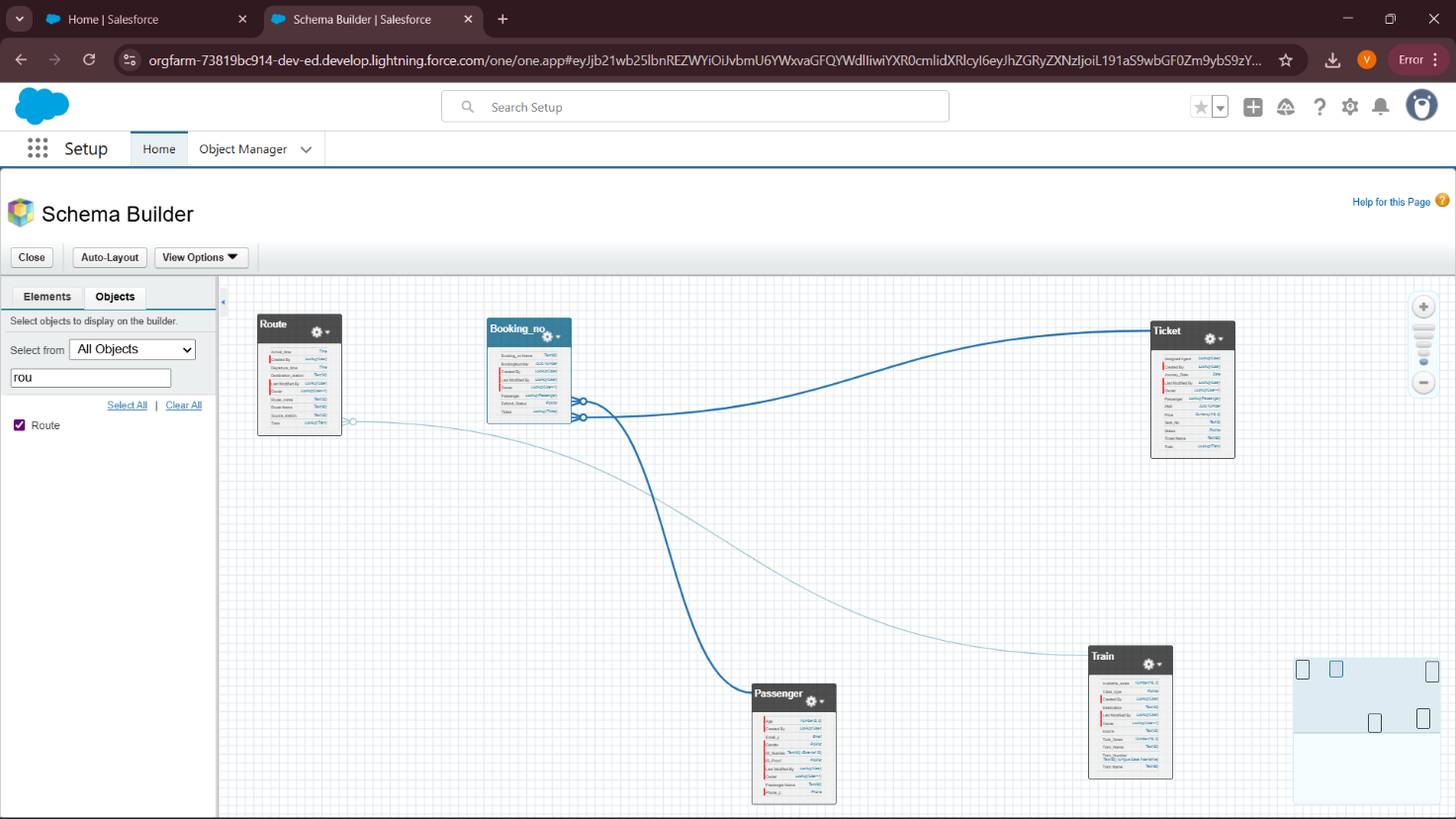
**🔹 5. Compact Layouts**

Compact layouts were configured for quick view in Lightning header & mobile:

* **Train\_\_c** → Train Name, Train No, Available Seats.
* **Ticket\_\_c** → PNR, Status, Journey Date, Train.
* 

**🔹 6. Schema Builder**

* Verified relationships using **Schema Builder**.
* Relationships configured:
  + **Ticket → Train** = Lookup (1 Train : Many Tickets).
  + **Ticket → Passenger** = Lookup (1 Passenger : Many Tickets).
  + **Ticket → Assigned Agent** = Lookup(User).
  + **Booking → Ticket & Passenger** = optional Master-Detail (junction).
  + **Route → Train** = Lookup.
* Schema Builder used to visualize and confirm links between objects.



**🔹 7. Relationship Choices**

* **Lookup** used where we want flexibility (Train ↔ Ticket, Passenger ↔ Ticket).
* **Master-Detail** used in junction (Booking ↔ Ticket, Booking ↔ Passenger) if grouping is required.
* **Hierarchical Relationship** is only for User object (not applied here).
* This design ensures data safety (Tickets not auto-deleted if Train is deleted)