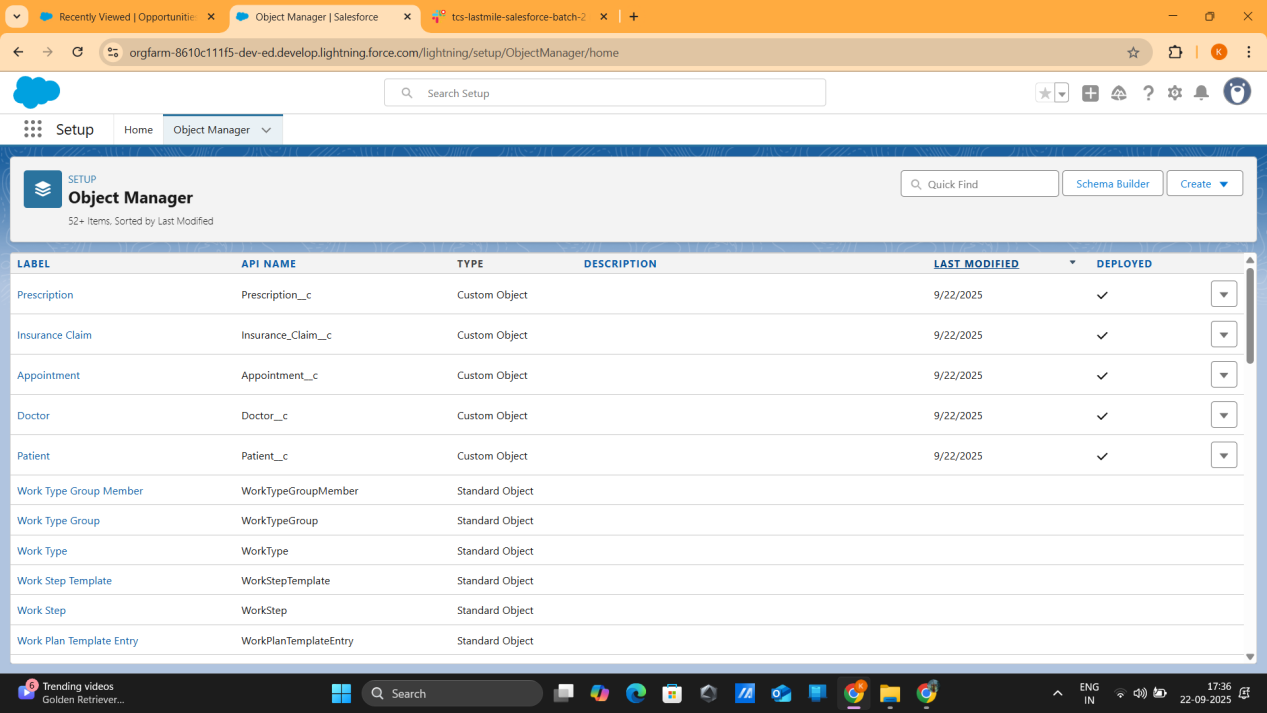
# ****Phase 3: Data Modeling & Relationships****

### ****Objective****

The objective of Phase 3 is to design the core data structure for the hospital management system in Salesforce. This includes creating custom objects, defining key fields, establishing relationships, configuring record types, and visualizing the schema. The goal is to ensure that all patient, appointment, doctor, prescription, and insurance claim data is organized, linked, and ready for efficient operations, automation, and reporting.

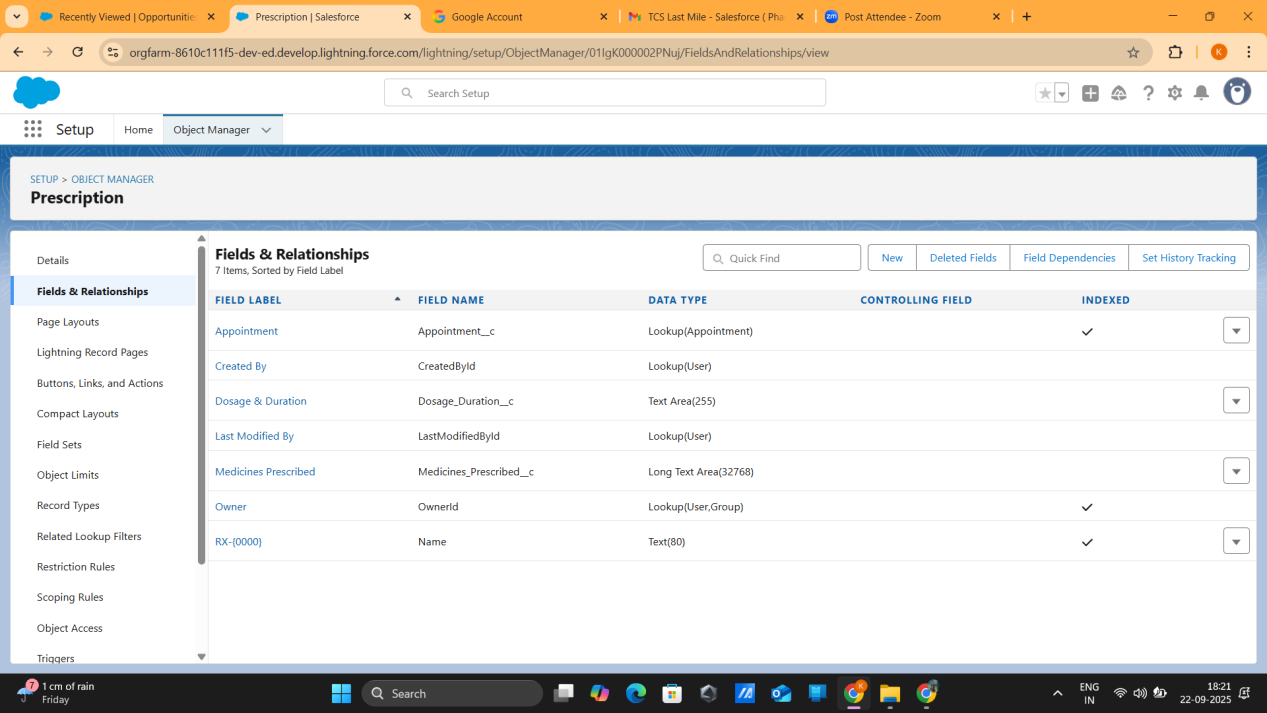
### ****1. Objects****

Five custom objects were created to represent the key entities in the hospital management system. The **Patient** object stores essential patient details along with their medical history. The **Doctor** object captures doctor-specific details, including their specialization such as General Physician, Cardiologist, Neurologist, Orthopedic, Dermatologist, or Other. The **Appointment** object tracks scheduled visits, linking patients to doctors, and includes fields such as Appointment Time and Status. The **Insurance Claim** object manages claims submitted for payment, capturing the claim type and its status. Finally, the **Prescription** object records medicines prescribed, including dosage and duration, and links each prescription to the corresponding appointment.

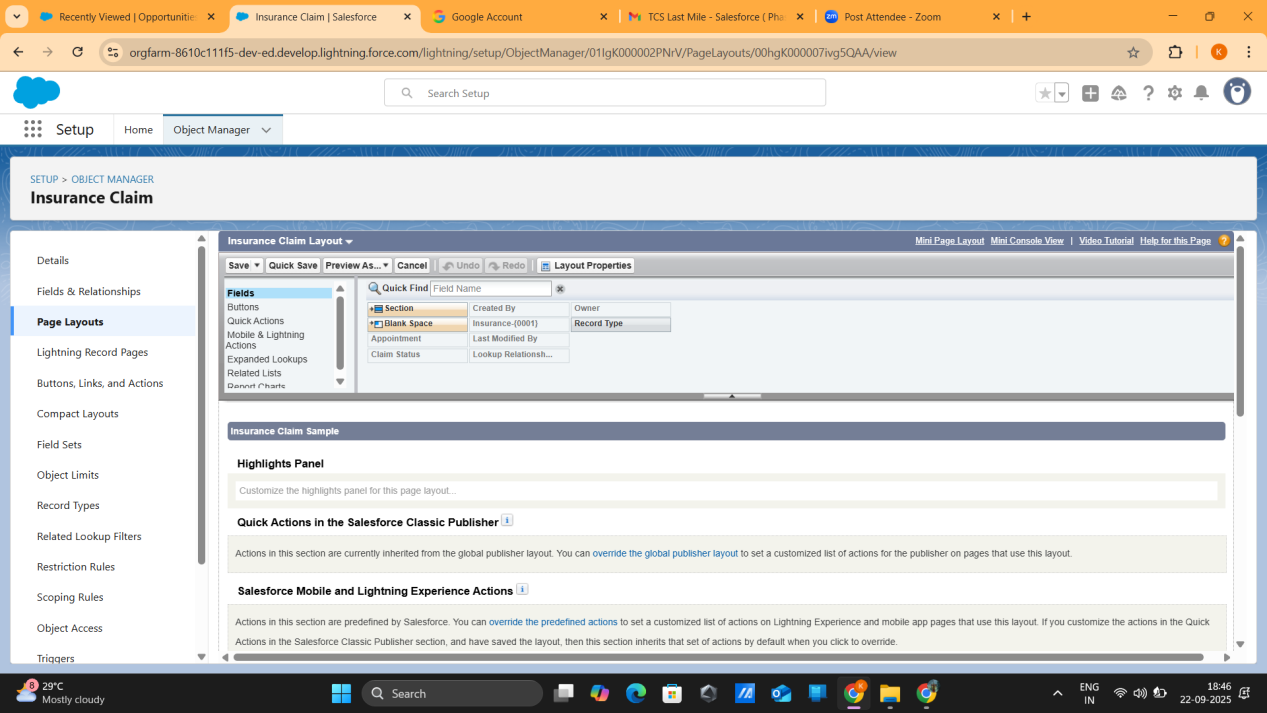
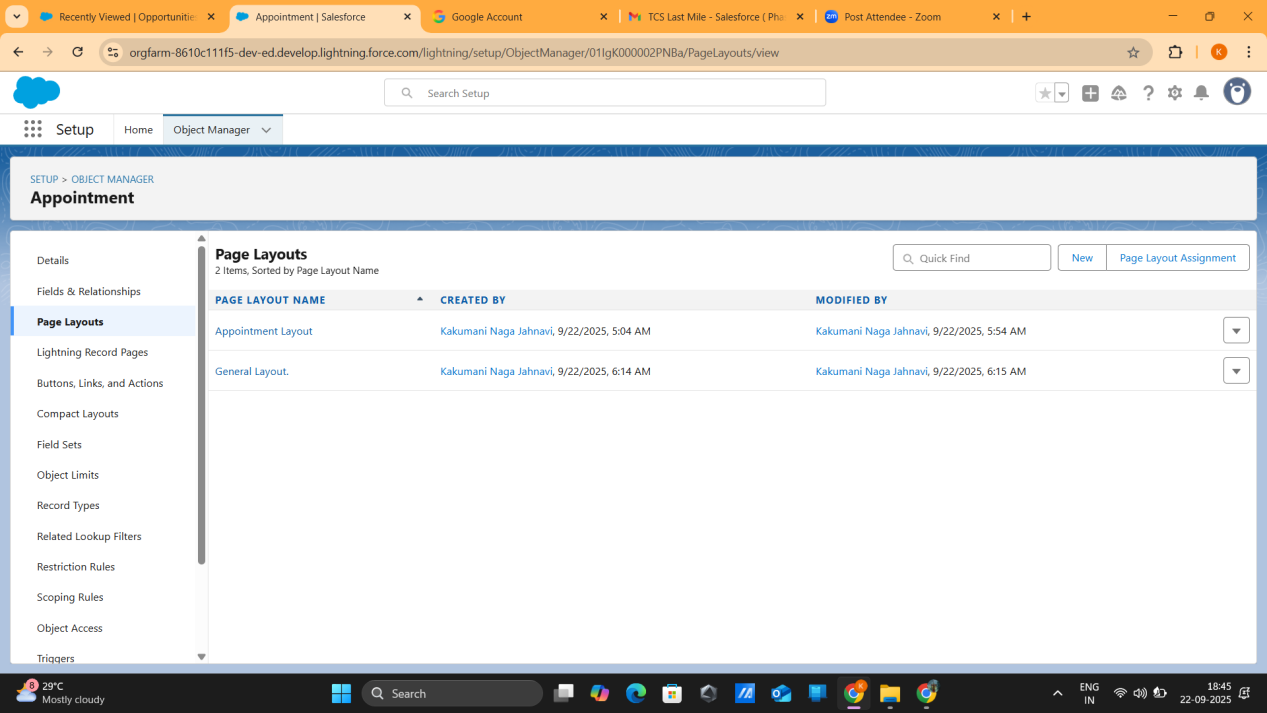


### ****2. Key Fields****

Each object includes key fields necessary for operational efficiency. The Appointment object contains **Appointment Time** to record the scheduled date and time. The Patient object includes **Patient History** as a long text area to store medical notes. The Insurance Claim object features a **Claim Status** picklist with values such as Submitted, Approved, Rejected, and Pending. The Doctor object includes a **Doctor Specialization** picklist to classify doctors by their area of expertise. The Prescription object includes **Medicines Prescribed** as a long text field and **Dosage & Duration** as a text field to provide detailed instructions for each prescription.

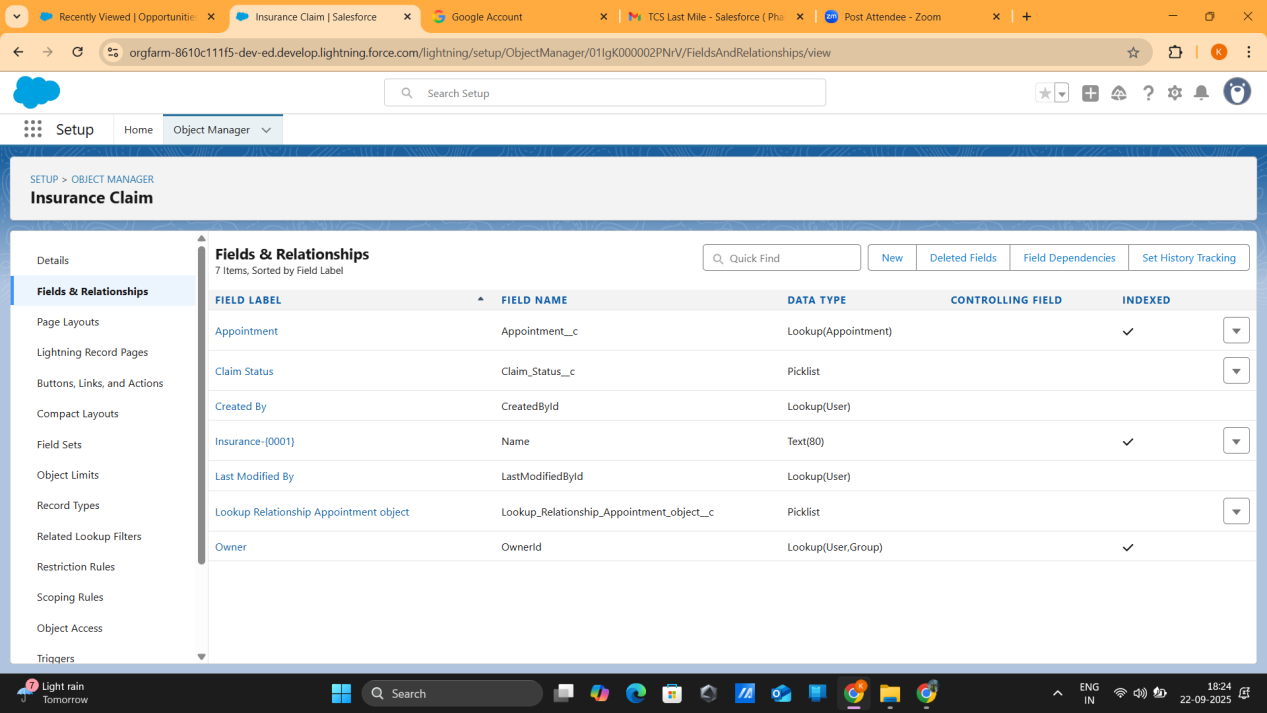


### ****3. Relationships****

Proper relationships were defined to maintain data integrity and reflect real-world hospital operations. Each **Patient** can have multiple **Appointments** (1:M), and each **Doctor** can be associated with multiple **Appointments** (1:M). Each **Appointment** may generate one **Insurance Claim** (1:1) and can produce multiple **Prescriptions** (1:M). Lookup relationships were primarily used for flexibility, while master-detail relationships were considered in scenarios requiring strict ownership. These relationships ensure that records are properly linked and can be easily queried for reporting and automation.

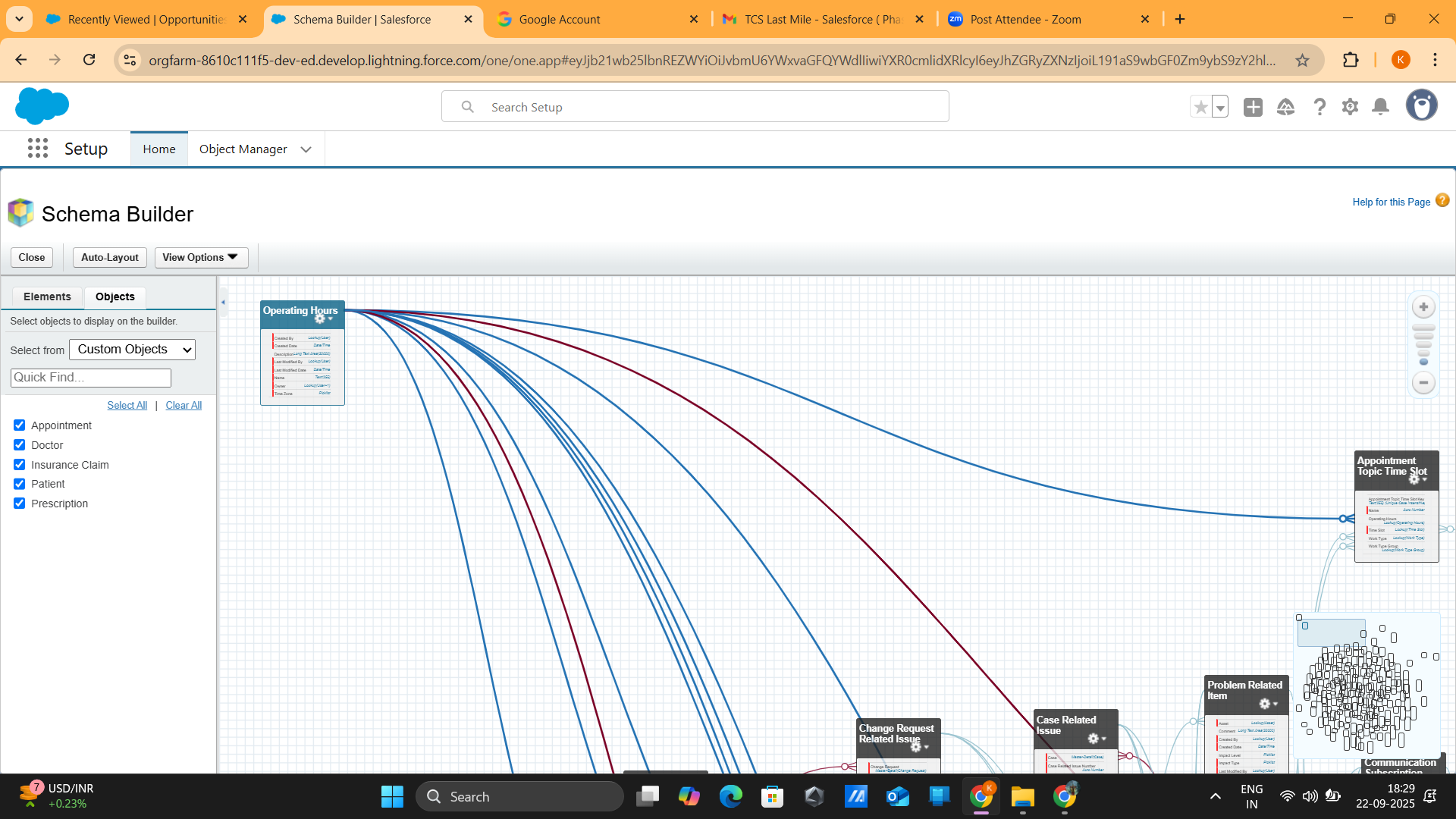
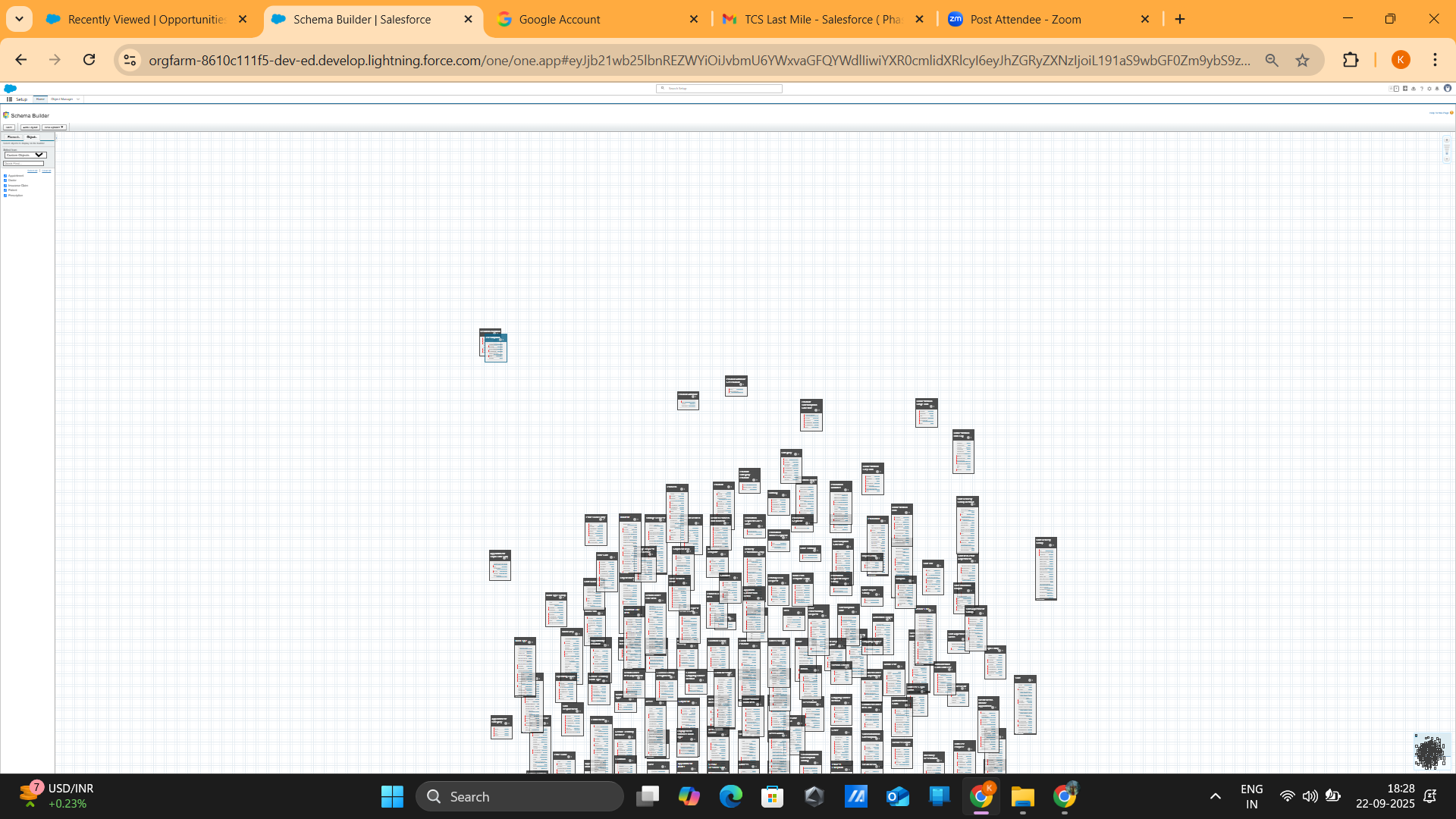
### ****4. Record Types****

Record types were implemented to manage different business scenarios efficiently. For **Appointments**, two record types were created: **General** and **Emergency**, allowing for distinct page layouts and fields such as priority for urgent visits. For **Insurance Claims**, record types include **Insurance** and **Self-Pay**, enabling users to capture the type of payment and apply tailored page layouts. Record types ensure that users only see relevant fields based on the context, improving data accuracy and user experience.



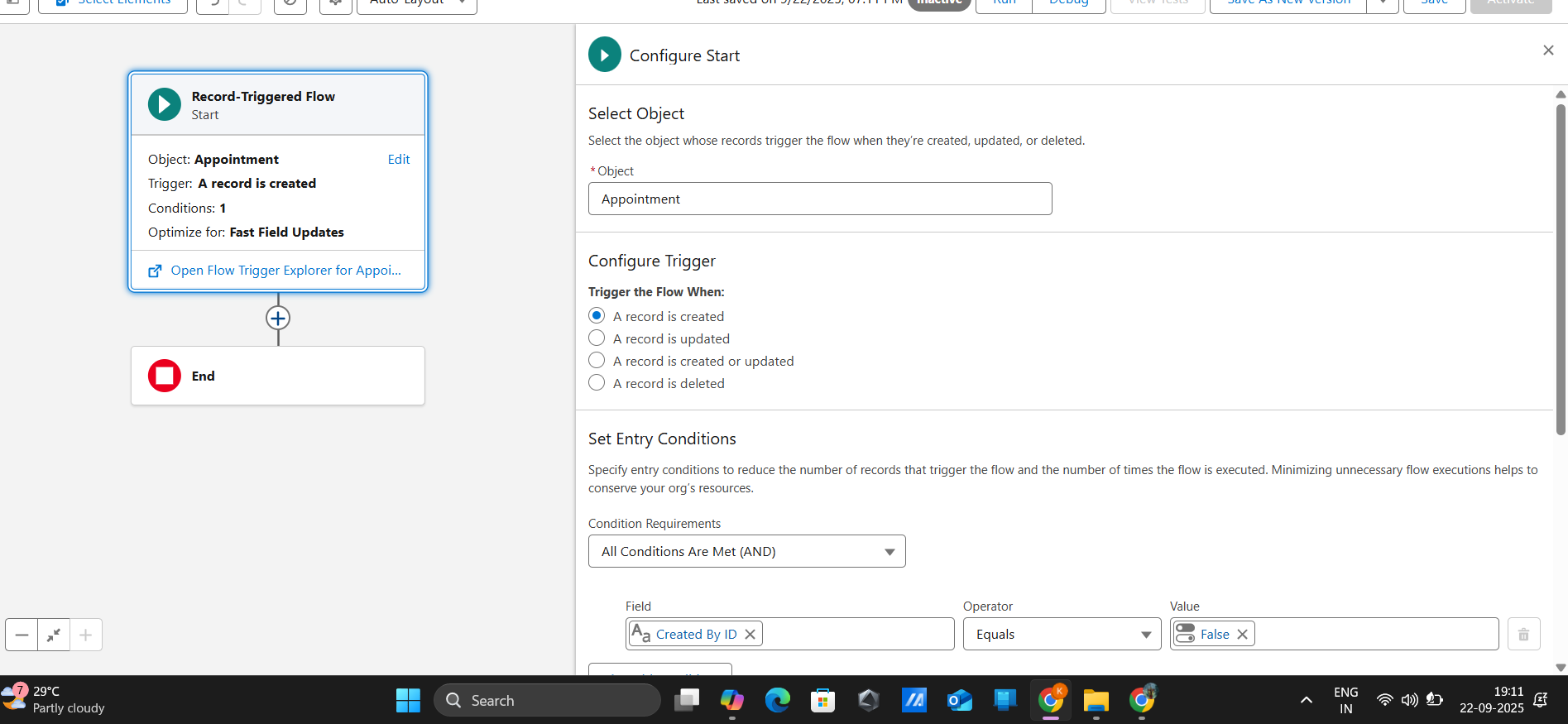
### ****5. Schema Visualization****

The Salesforce **Schema Builder** was used to visualize all objects and relationships. This tool provides a clear graphical representation of how objects such as Patient, Appointment, Doctor, Insurance Claim, and Prescription are interconnected. Proper lookup fields were confirmed to maintain data integrity, ensuring smooth operations across the system. Key schema notes include: Patient → Appointment (1:M), Appointment → Doctor (M:1), Appointment → Claim (1:1), and Appointment → Prescription (1:M).



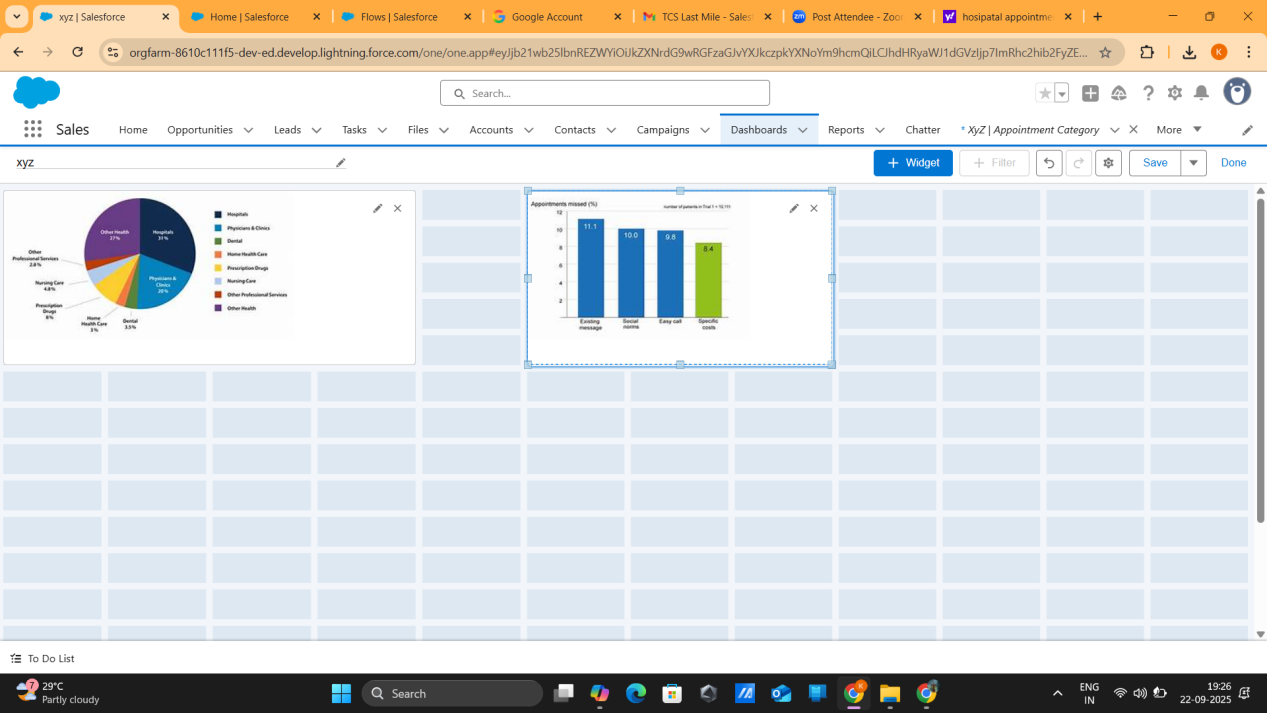
### ****6. Automation & Rules****

Automation was implemented to improve efficiency and reduce manual errors. **Validation rules** were created to prevent duplicate insurance claims for the same appointment. Salesforce **Flows and Workflows** were used to automatically send email or SMS reminders to patients and doctors before appointments. Additionally, a flow was implemented to auto-update the status of insurance claims when payment is received. These automations ensure timely communication, accurate claim processing, and consistent data management.



### ****7. Reporting & Dashboards****

Comprehensive reporting and dashboards were developed for hospital administrators. Reports include **Appointments per Doctor**, **Claim Status Summary**, and **Emergency vs General Appointments**, providing detailed insights into operations. Dashboards visualize these reports using bar charts for doctor workloads, donut charts for claim status distribution, pie charts for appointment types, and tables displaying the latest prescriptions. These dashboards allow administrators to monitor key metrics at a glance, enabling data-driven decision-making.



### ****Outcome****

Phase 3 results in a fully structured Salesforce data model that supports efficient hospital management. The system enables accurate tracking of patients, appointments, prescriptions, and insurance claims. Automated notifications and flows reduce manual effort, while reports and dashboards provide comprehensive analytics for administrators. Overall, this phase establishes a robust foundation for operational