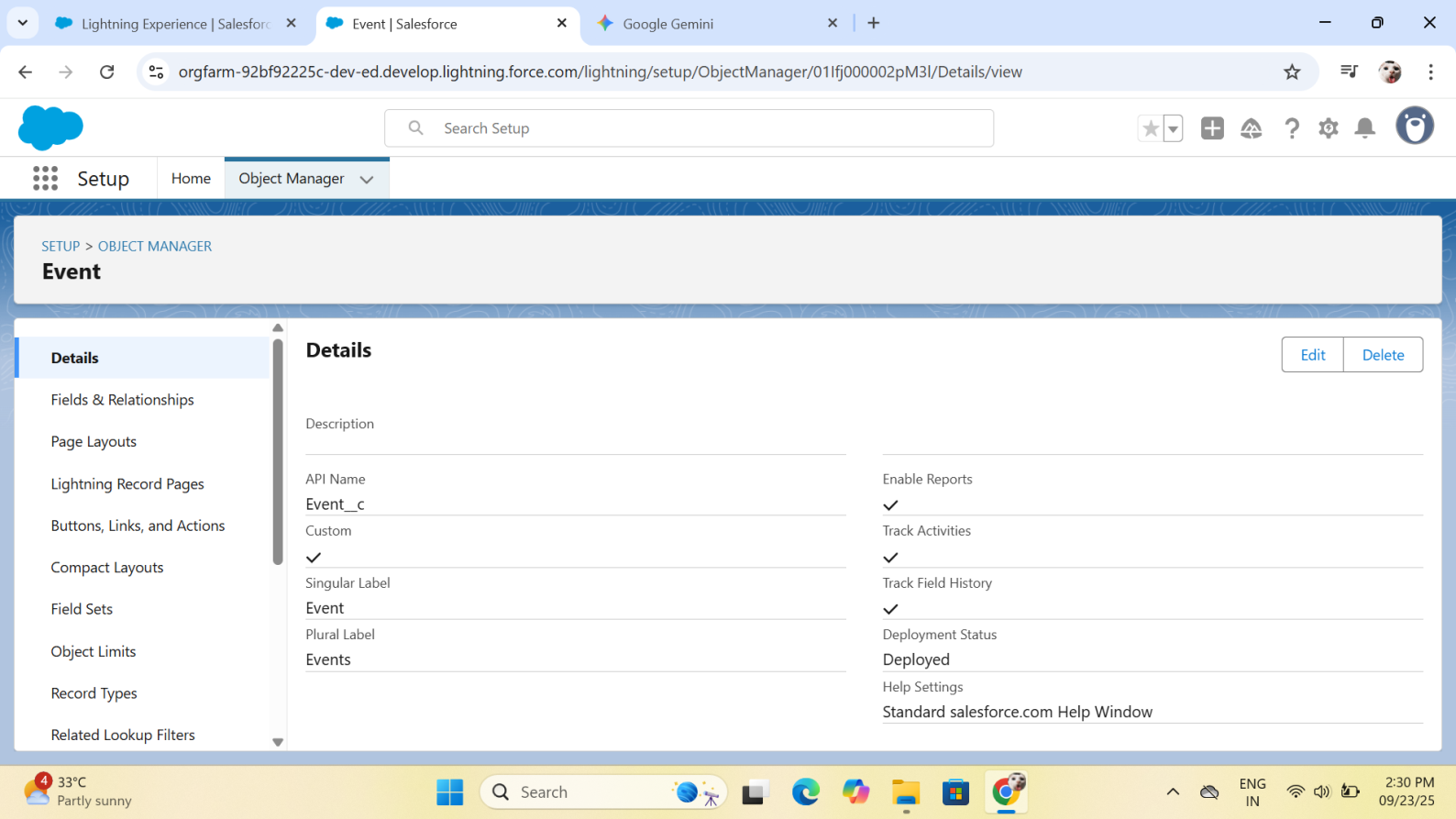
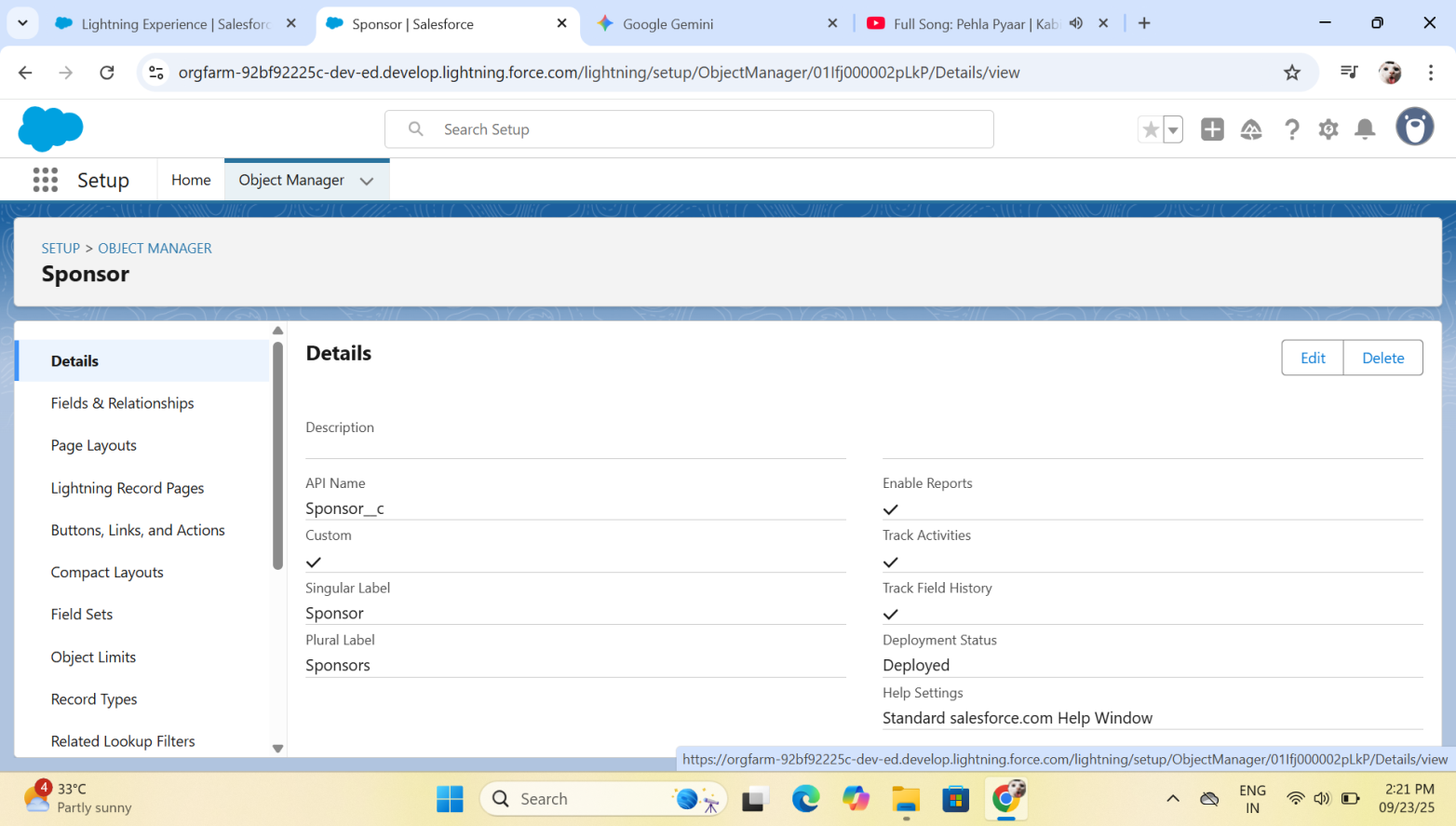
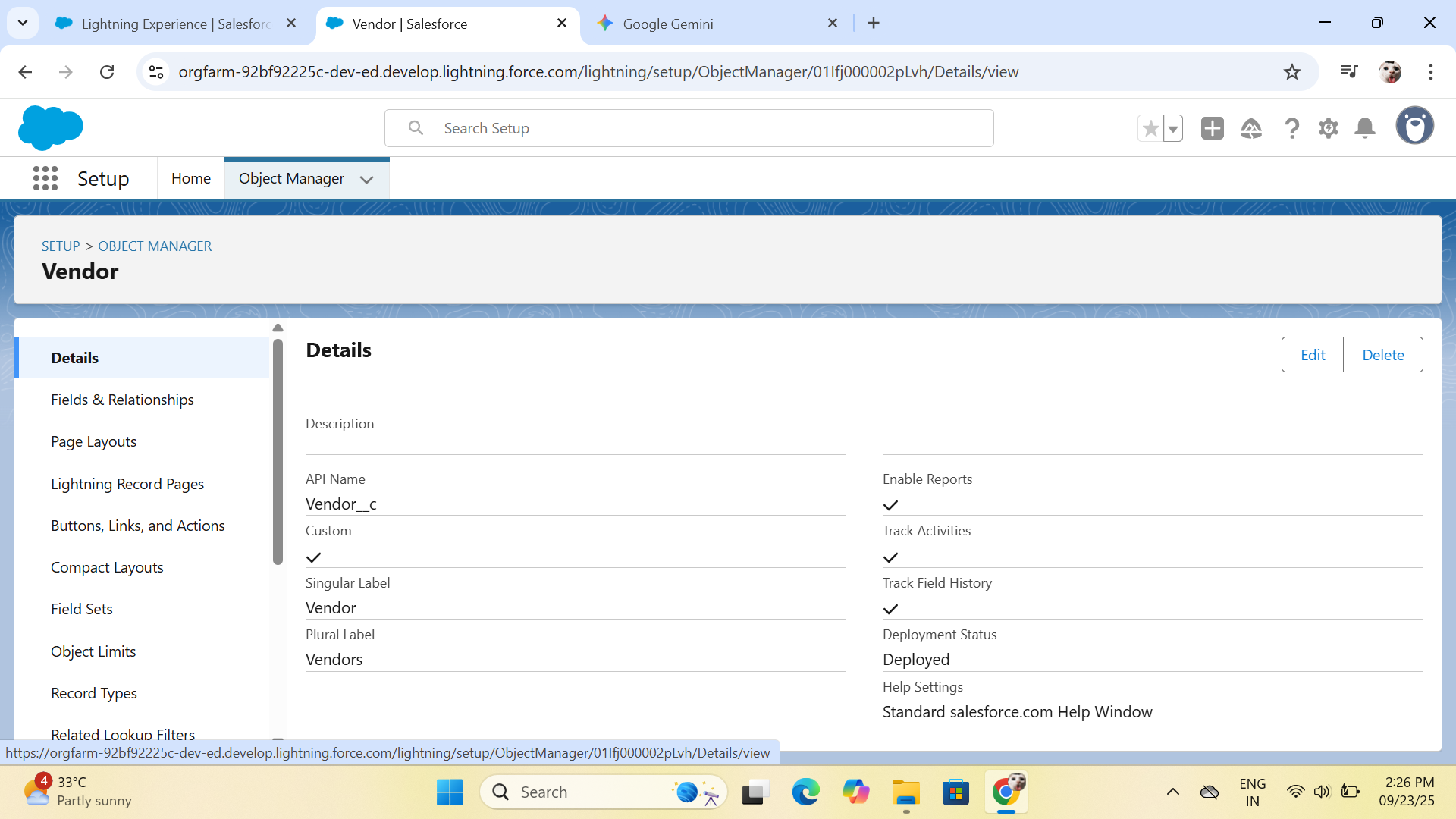
Phase III: Data Modeling and Relationships

**AURA EVENTS & SPONSOR**:  
College Events & Sponsor Management

**Standard & Custom Objects**

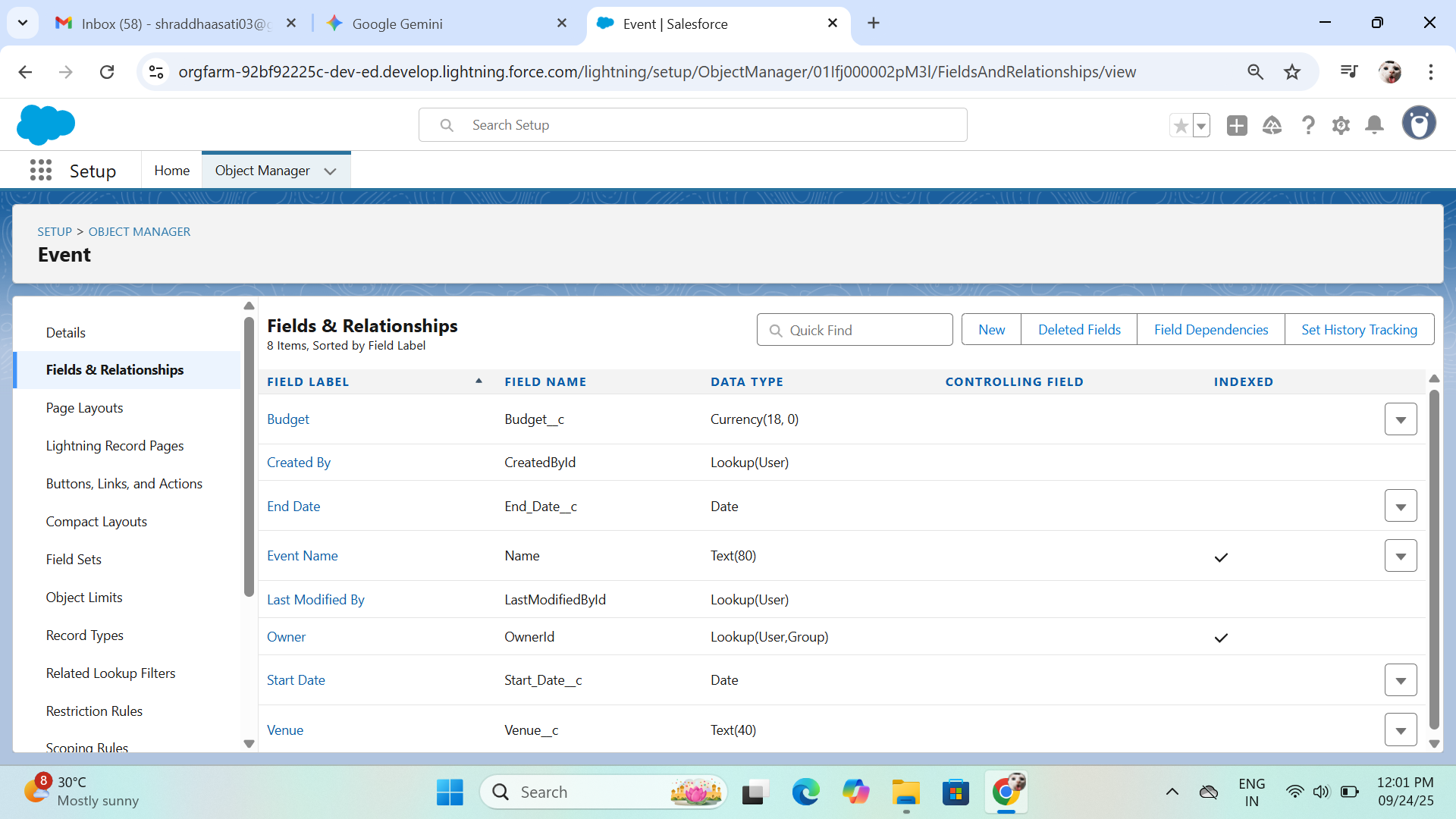
* **Explanation:** The project leverages a hybrid model using both existing Salesforce objects and new custom objects to fit specific business needs.
* **Salesforce Path:** Setup → Object Manager
* **Data Model:**
  + **Standard Objects Used:** Campaign (for marketing promotions), Account (for sponsors), and Contact (for sponsor contacts).
  + **Custom Objects Created:**
    - Event: The primary object for tracking all event details.
    - Sponsor: A custom object to manage sponsorship commitments, separate from standard Accounts for project-specific data.
    - Vendor: A custom object to manage all vendor contracts.
    - Sponsor Invoice: A custom object for the finance team to track invoices related to sponsorships.
    - Promotion Request: A custom object for the marketing team to manage their event-related tasks.
    - Sponsorship Contribution: A **junction object** to create a many-to-many relationship between Sponsor and Event records.

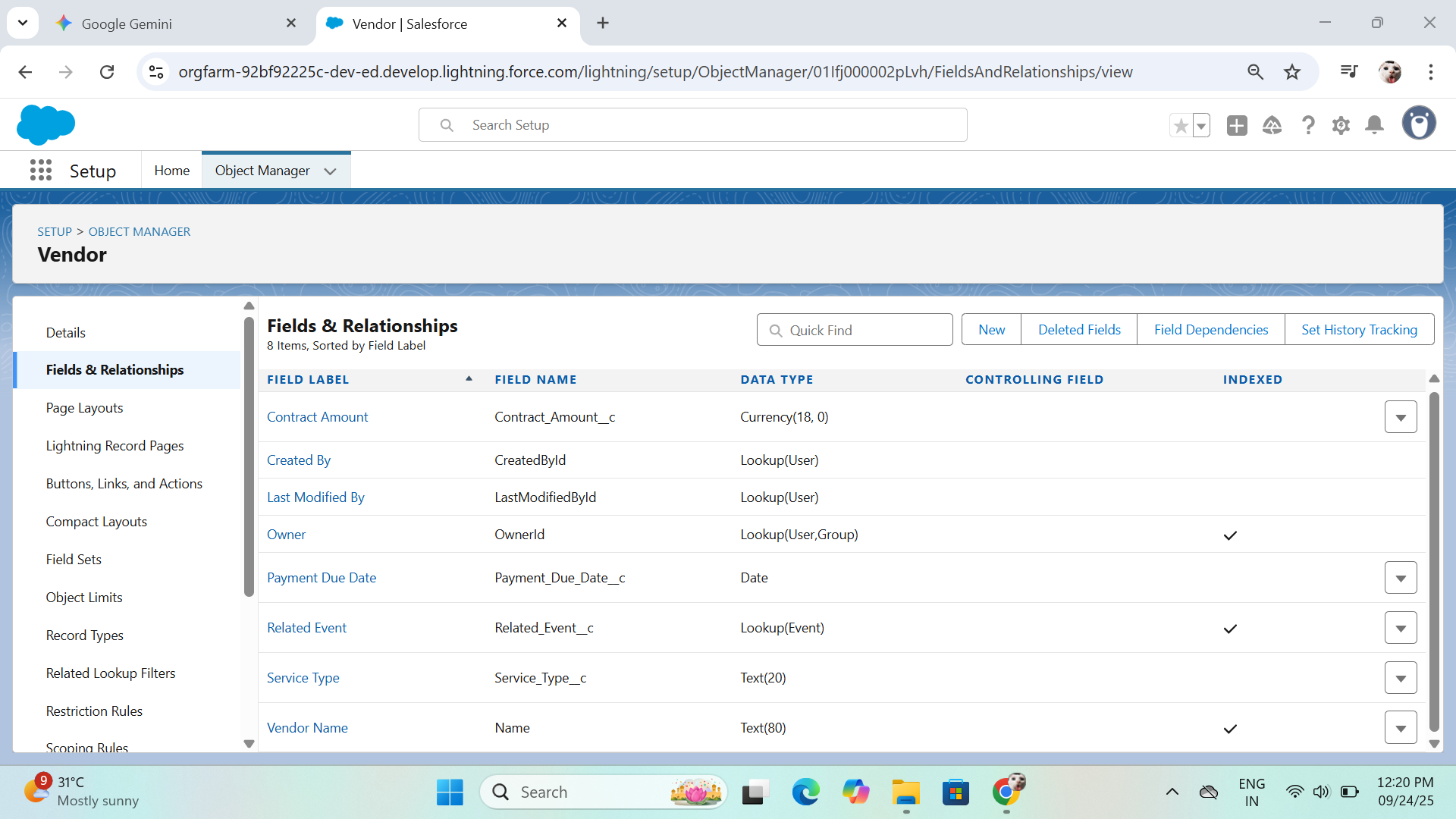
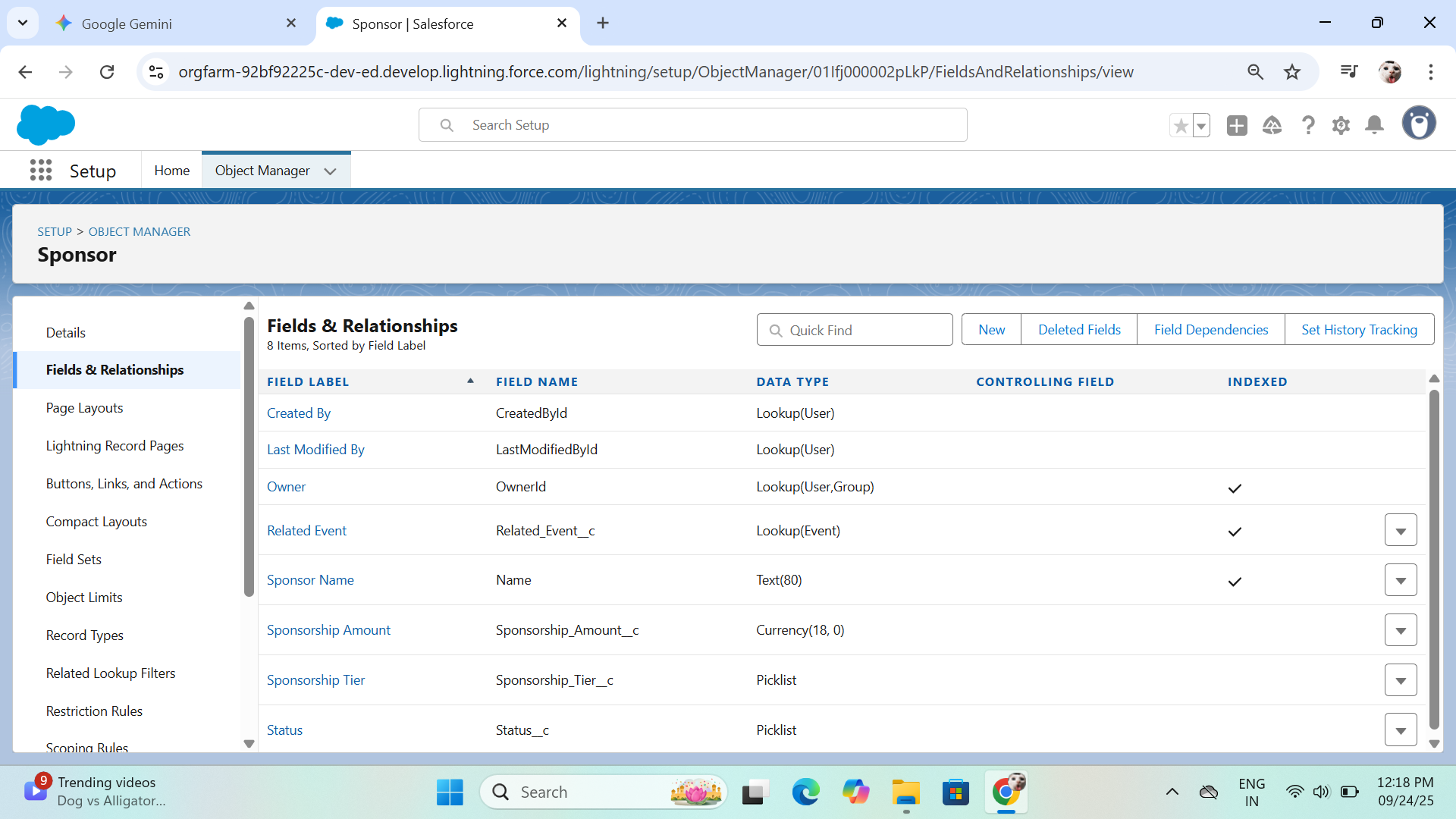




#### Fields & Relationships

* **Explanation:** Custom fields were created for each object to store specific data points, and relationships were built to connect them.
* **Salesforce Path:** Setup → Object Manager → Select Object → Fields & Relationships
* **Procedure & Configuration:**
* **Custom Fields:**
  + - **Event Object:** Venue (Text), Budget (Currency), Start Date (Date), End Date (Date), Featured Artists (Text Area, for cultural events), Performance Type (Multi-Select Picklist, for cultural events), Competition Type (Picklist, for tech fests), Keynote Speaker (Text, for academic events), Primary Topic (Text, for academic events).
    - **Sponsor Object:** Sponsorship Amount (Currency), Sponsorship Tier (Picklist), Status (Picklist).
    - **Vendor Object:** Service Type (Text), Contract Amount (Currency), Payment Due Date (Date).
    - **Sponsor Invoice Object:** Invoice Amount (Currency), Due Date (Date), Payment Status (Picklist), Invoice Number (Auto Number).
    - **Promotion Request Object:** Promotion Type (Picklist), Status (Picklist)
* **Relationships (Lookup):**
* Sponsor connected to Event.
* Vendor connected to Event.
* Sponsor Invoice connected to Sponsor.
* Promotion Request connected to Event.

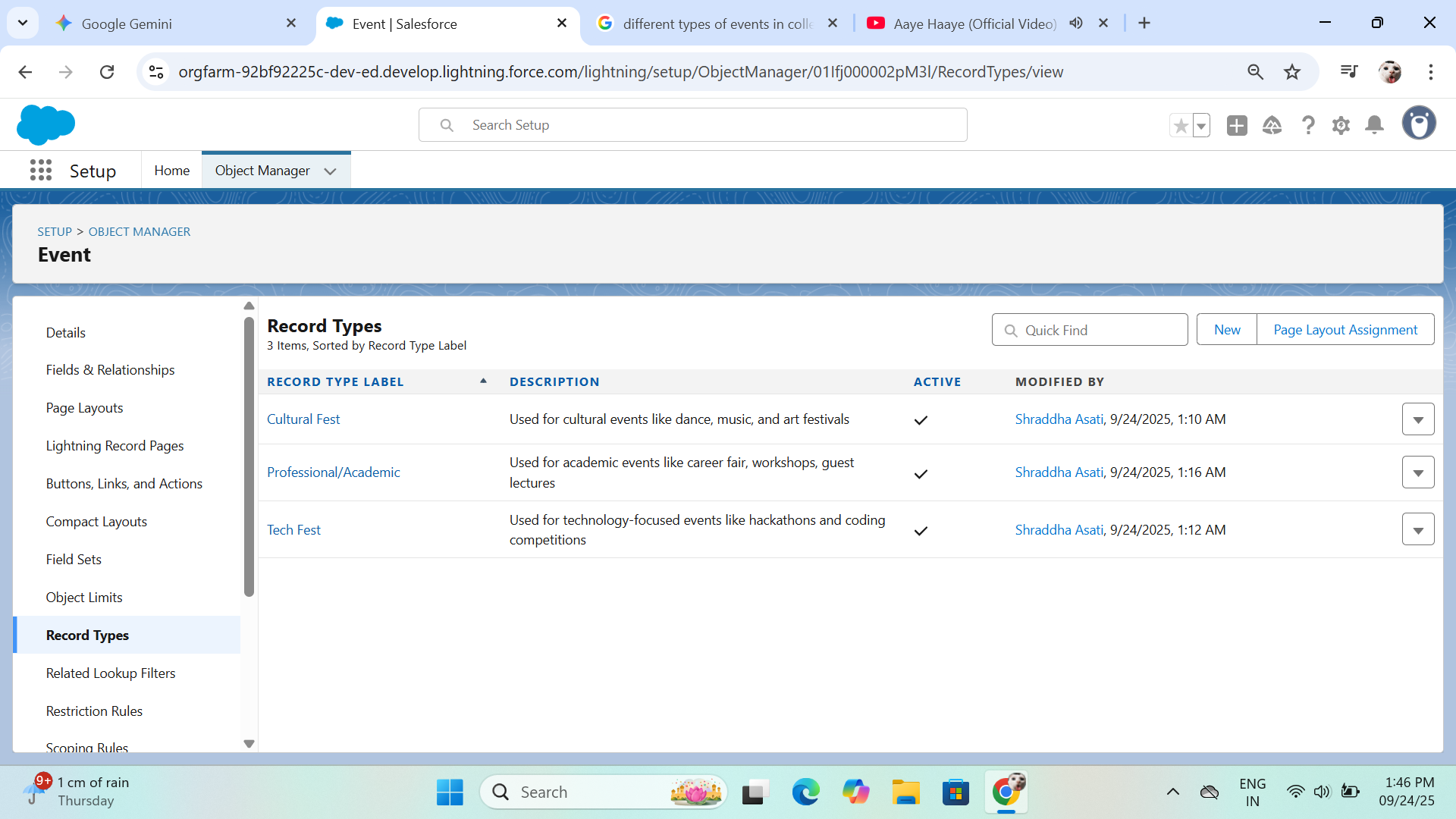


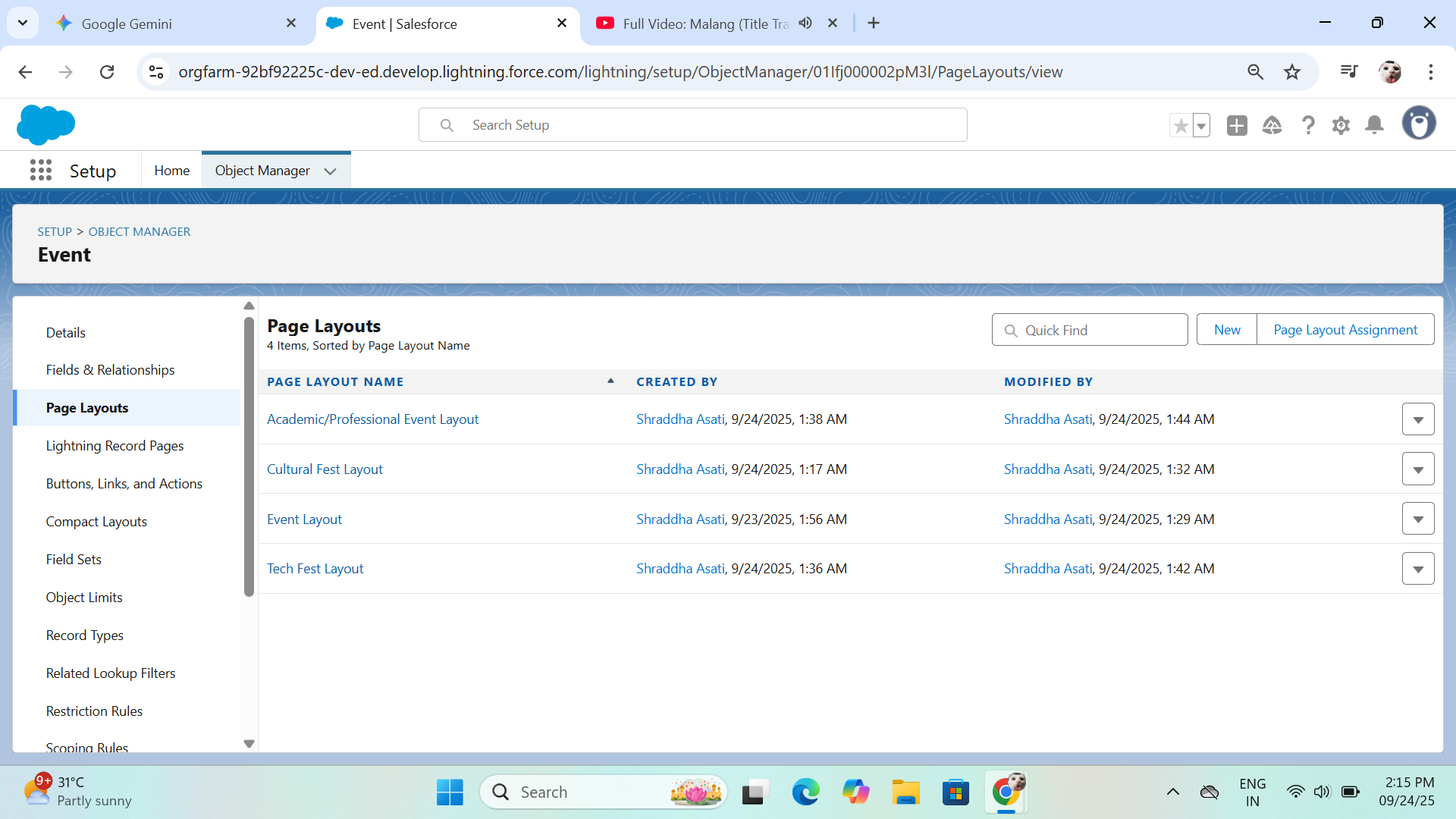


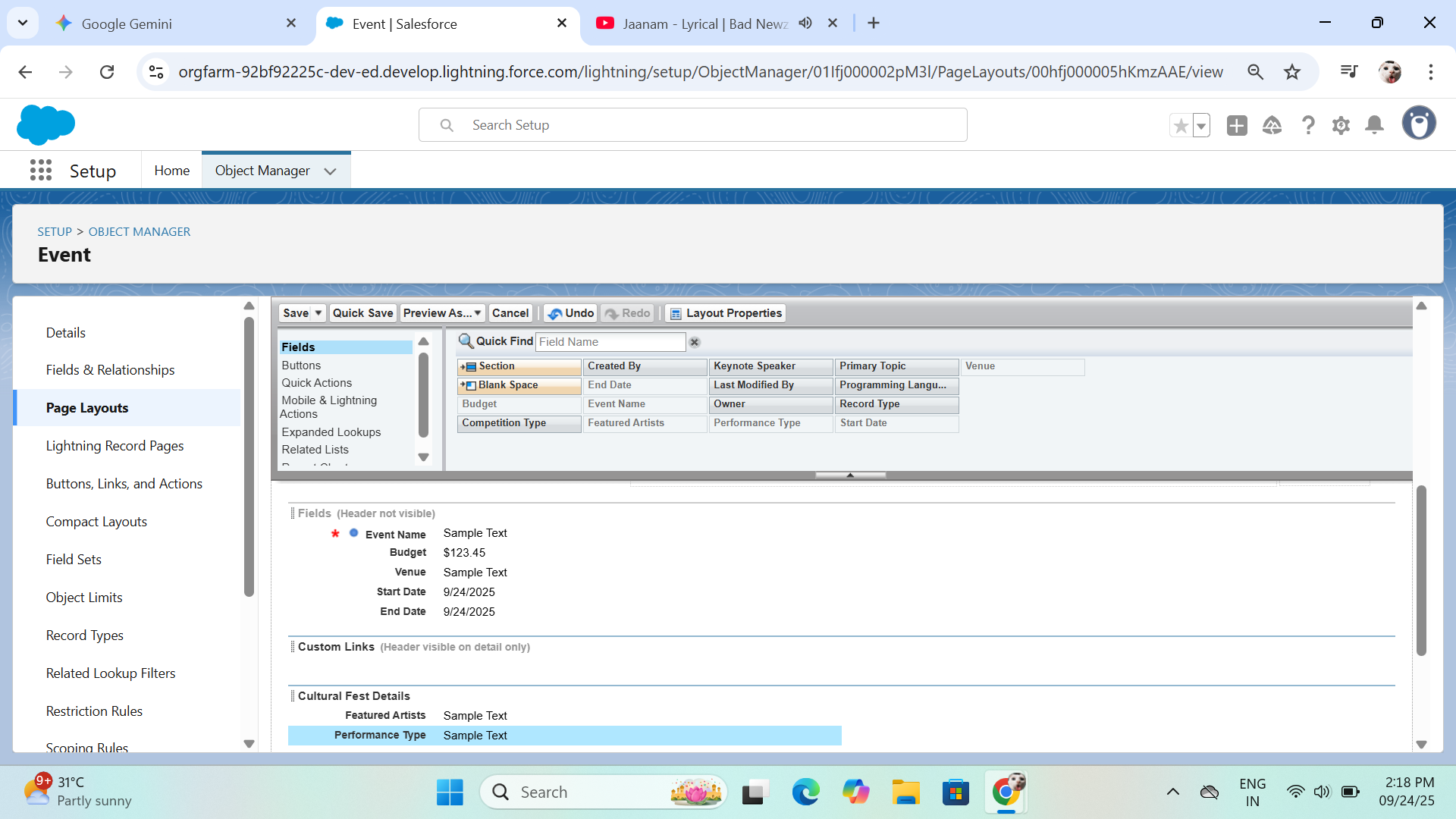
#### Record Types & Page Layouts

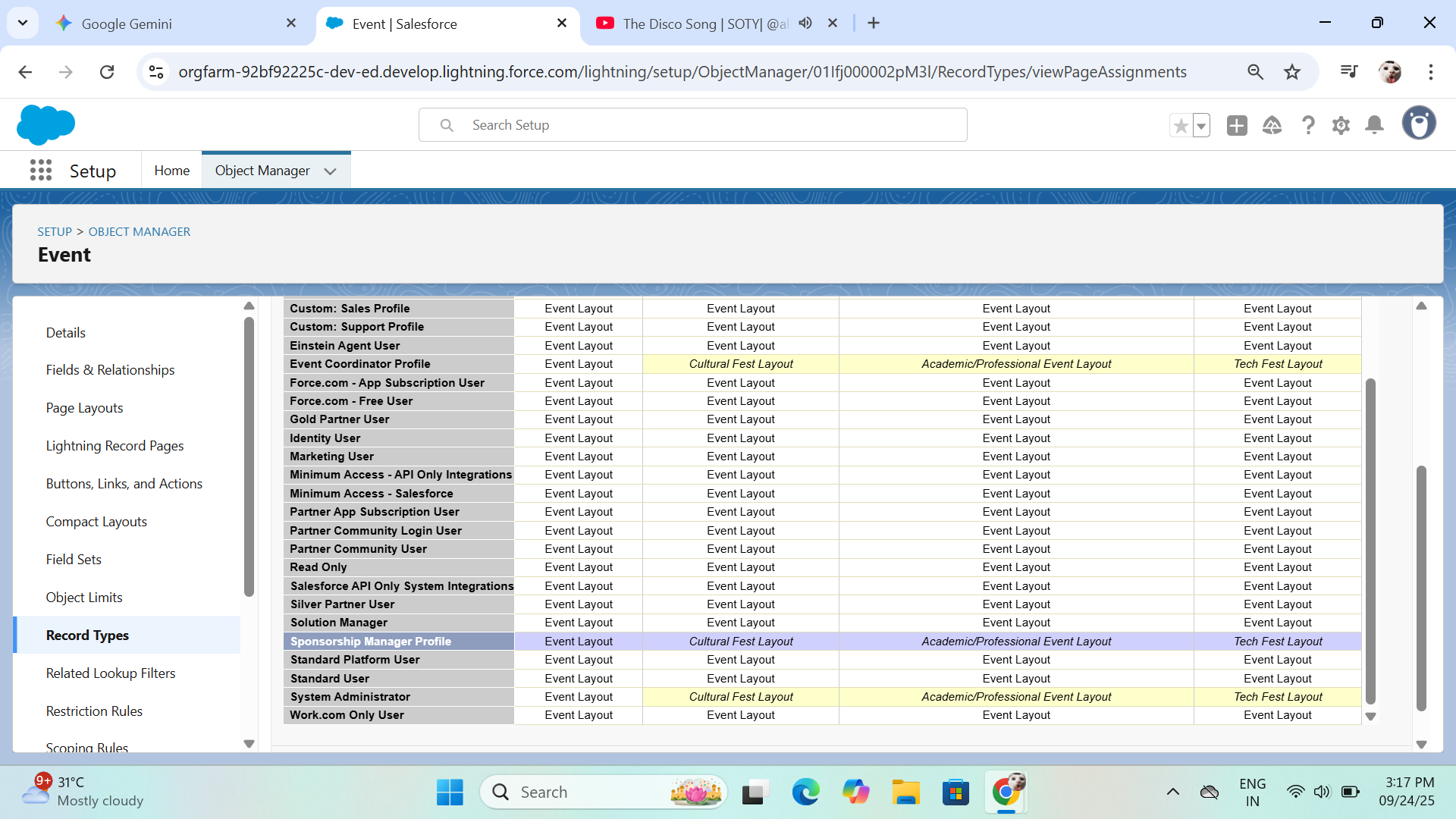
#### Explanation: Record types and page layouts were configured to provide a tailored user experience based on the type of event. Compact layouts were set up to provide a quick summary of key information.

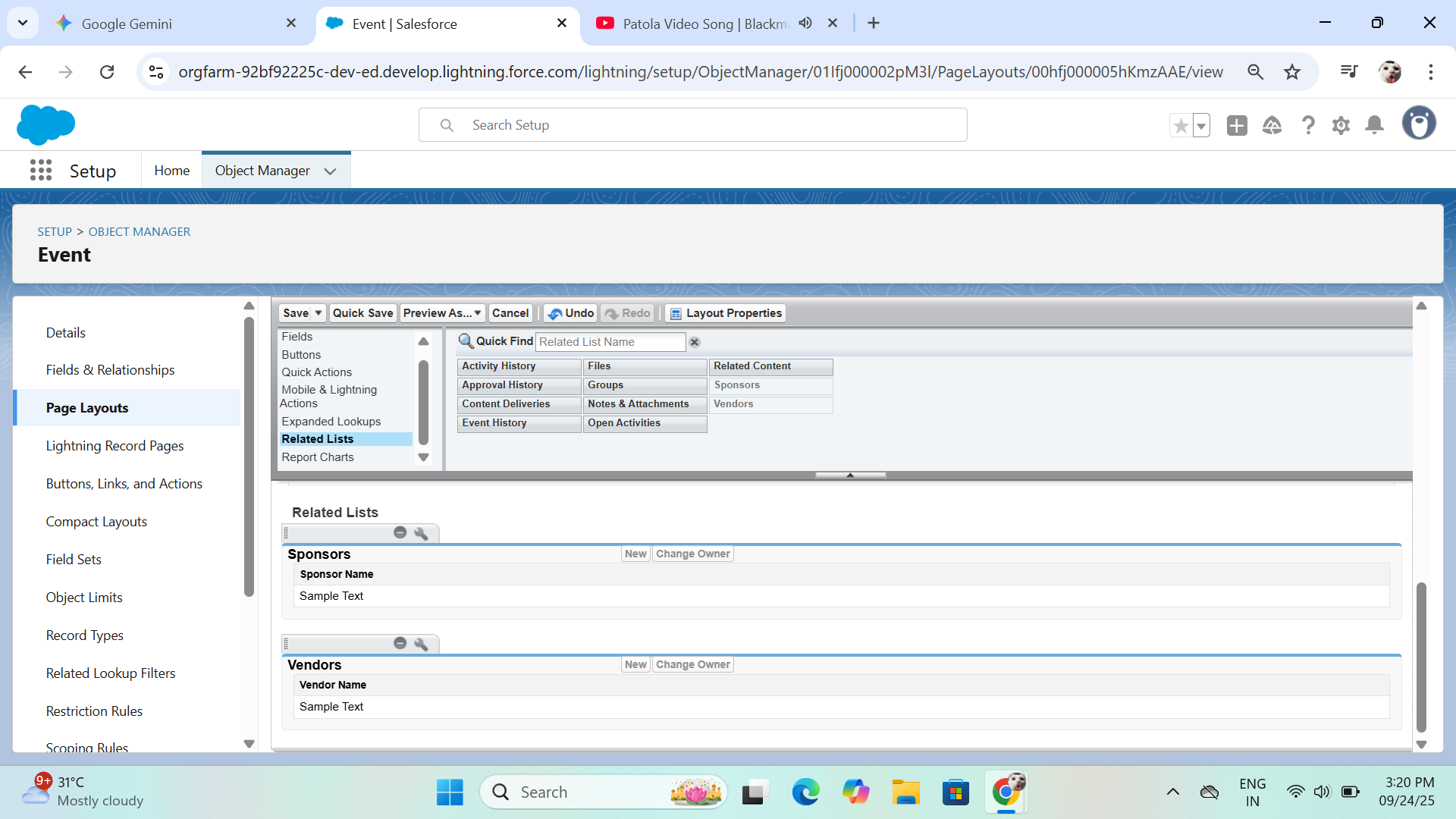
* **Salesforce Path:** Setup → Object Manager → Event → Record Types and Page Layouts
* **Procedure & Configuration:**
  + **Record Types:** Three record types were created for the Event object: Cultural Fest, Tech Fest, and Academic Professional Event.
  + **Page Layouts:**
    - Three corresponding page layouts were created, each displaying fields relevant to its record type (e.g., Featured Artists on the Cultural Fest Layout).
    - Related lists for Sponsors and Vendors were added to all event layouts.







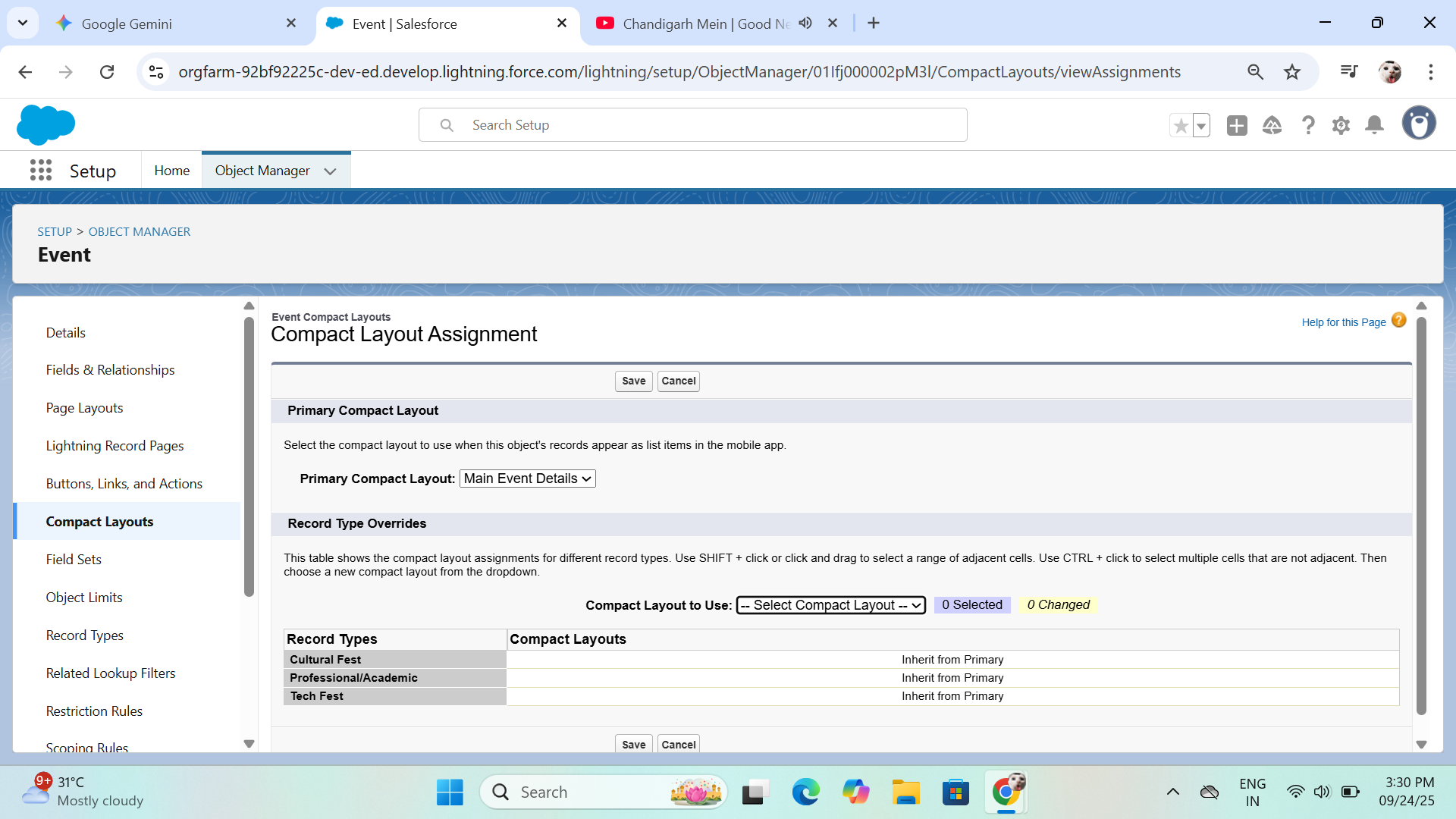


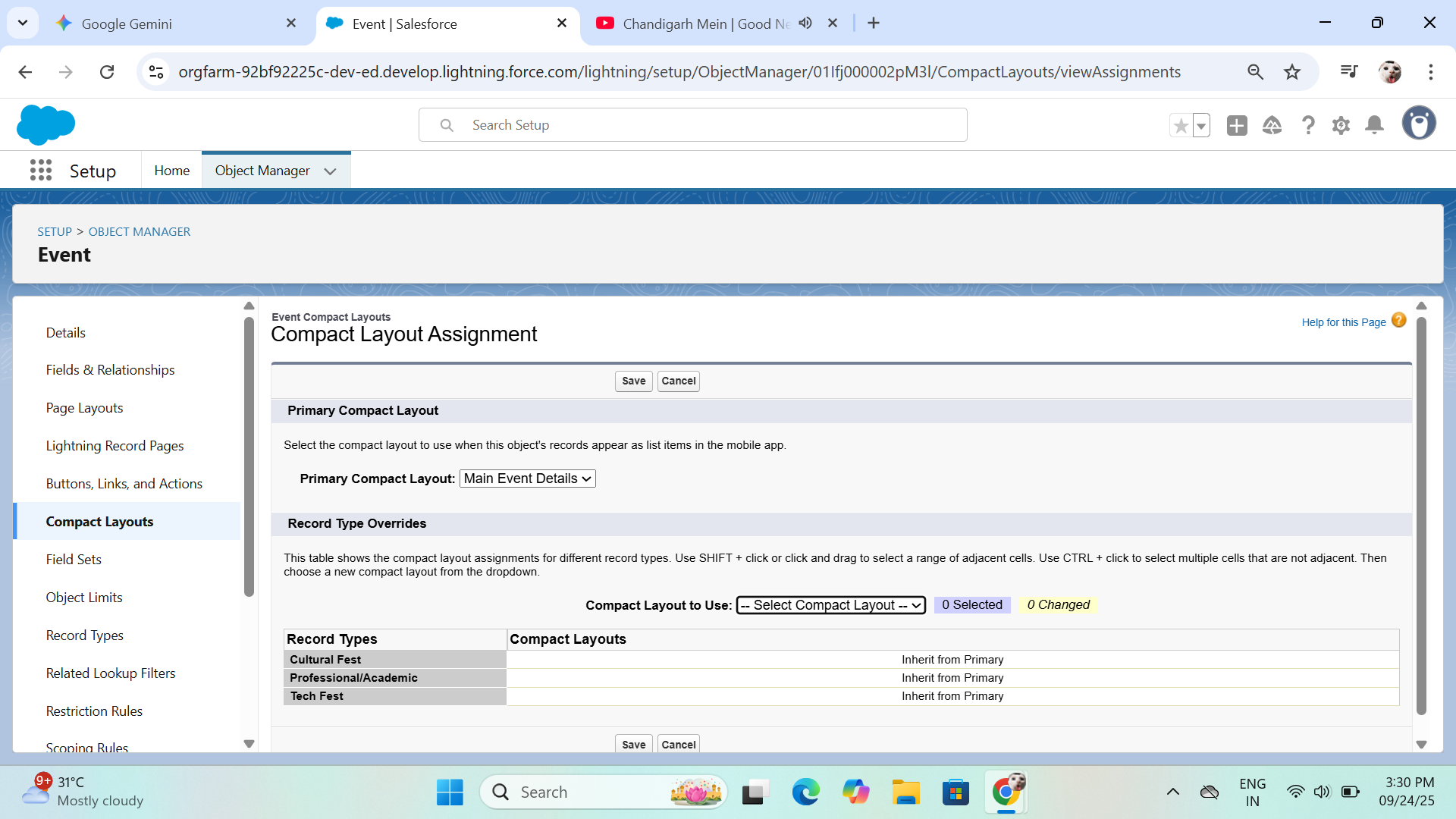


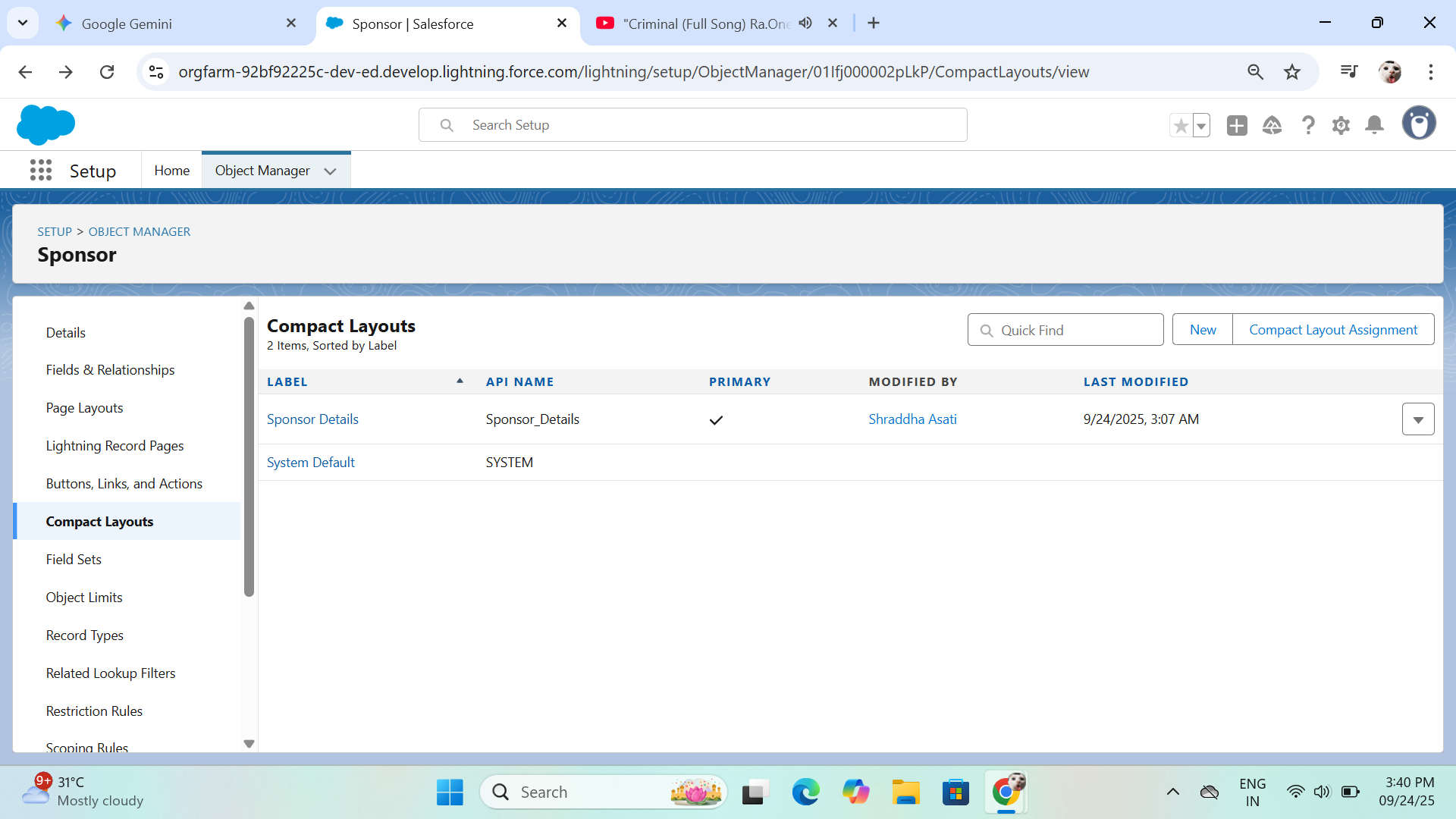
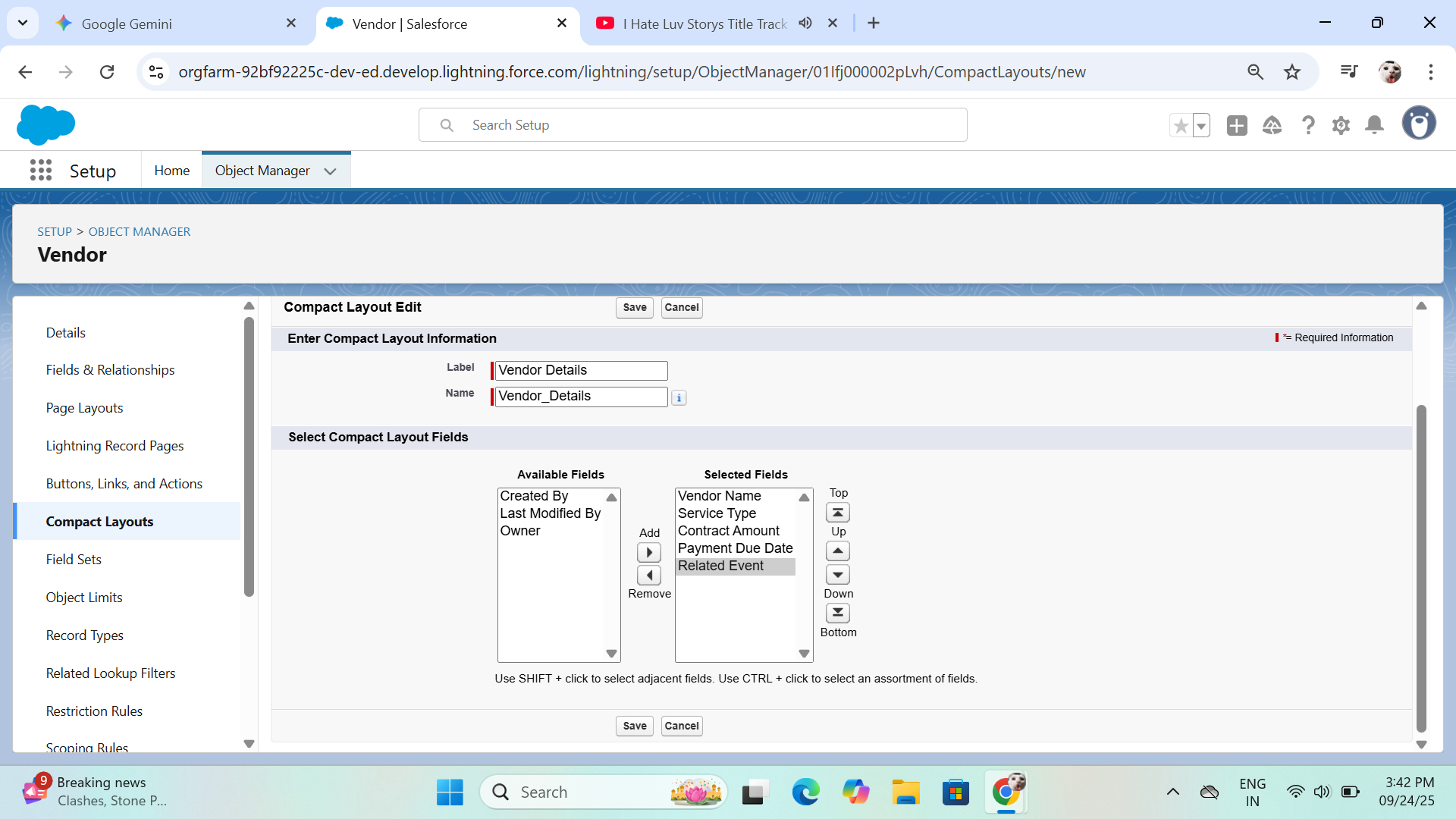
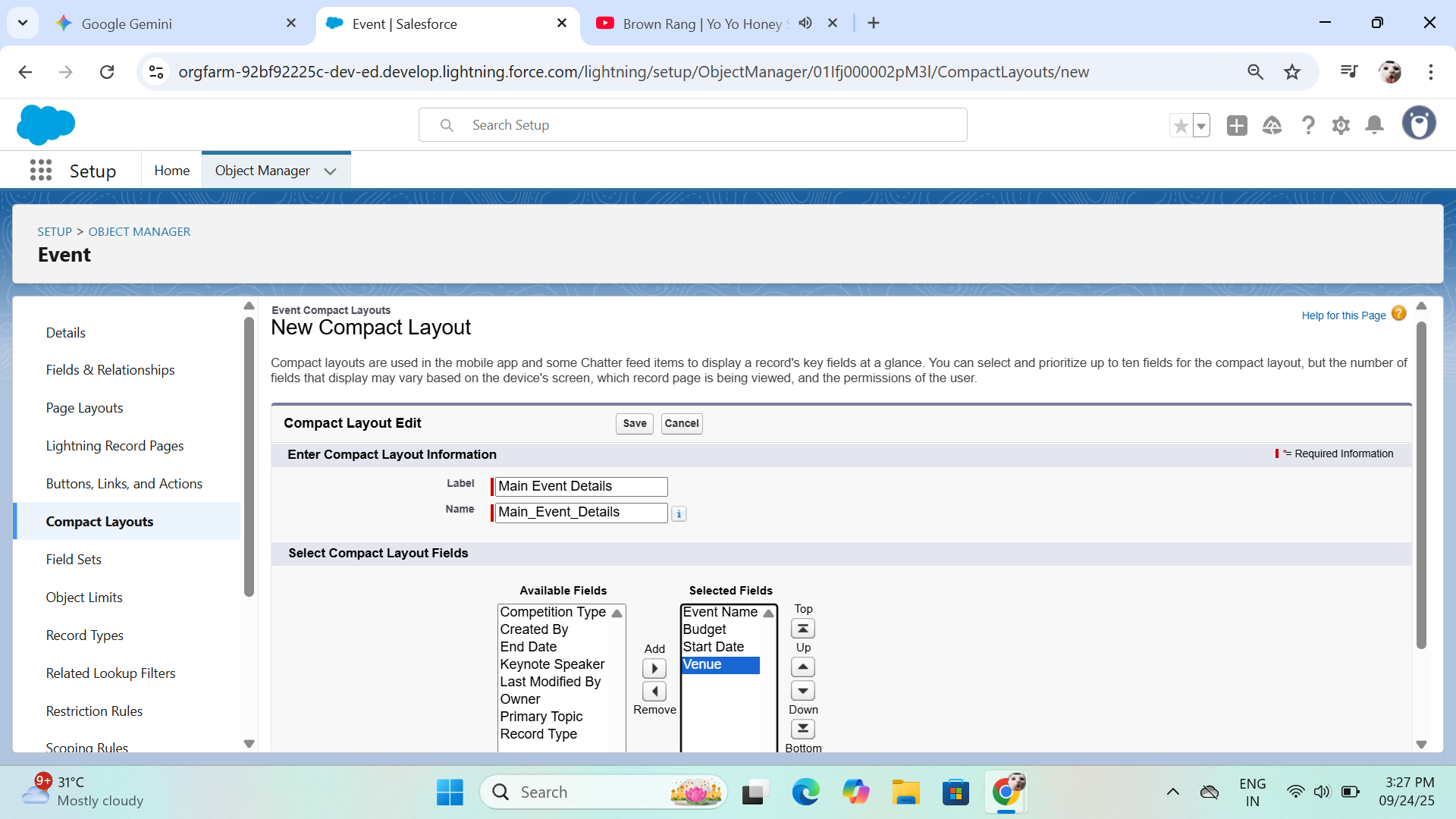
**Compact Layouts**

The Main Event Details compact layout was created and assigned as the primary layout for the Event object, ensuring key fields are visible in the highlight panel. The Sponsor and Vendor objects also had their compact layouts configured.

* **Final Configuration:** When a user creates a new event, they select a record type, which automatically displays the correct page layout and its associated fields.

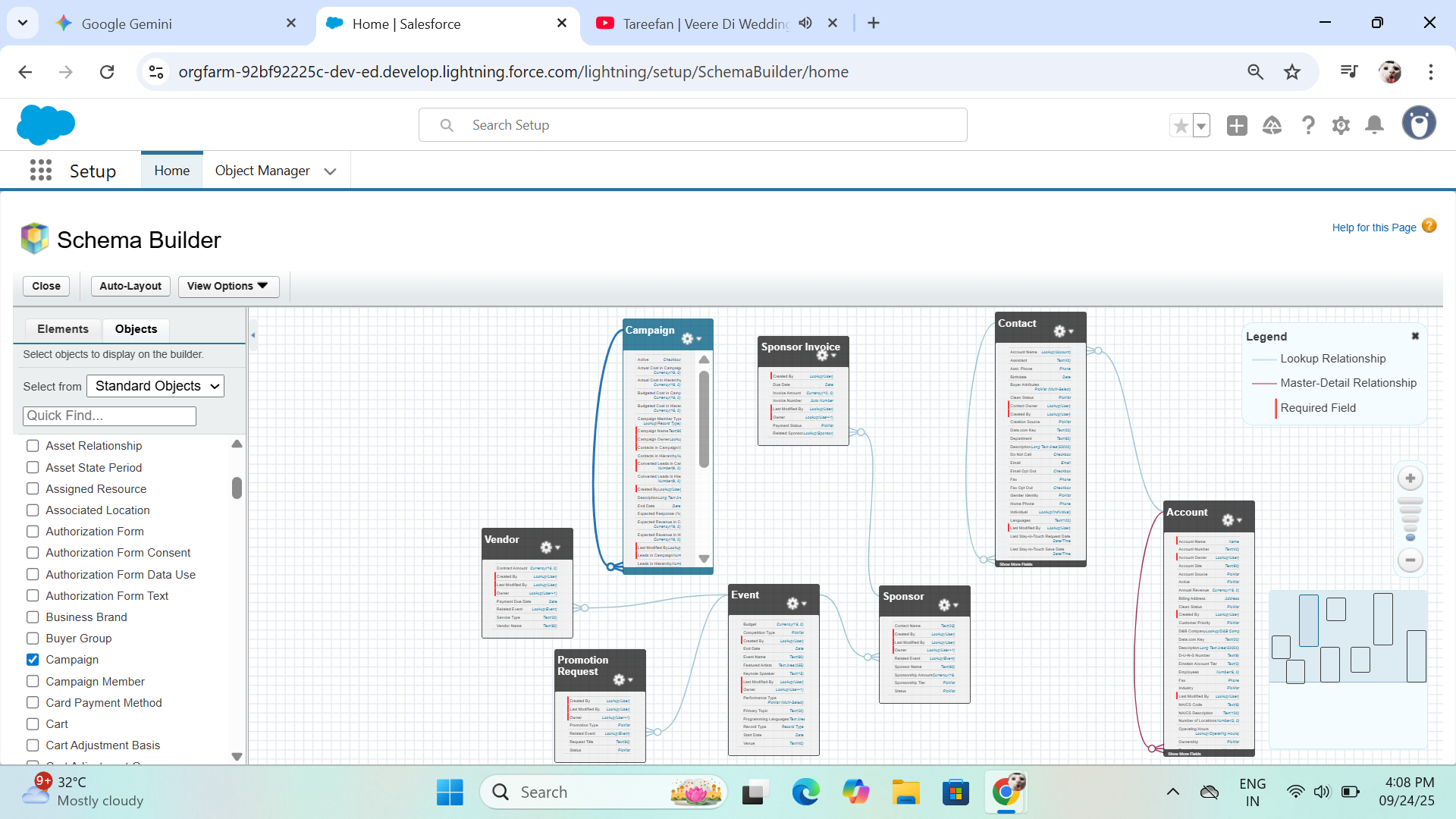






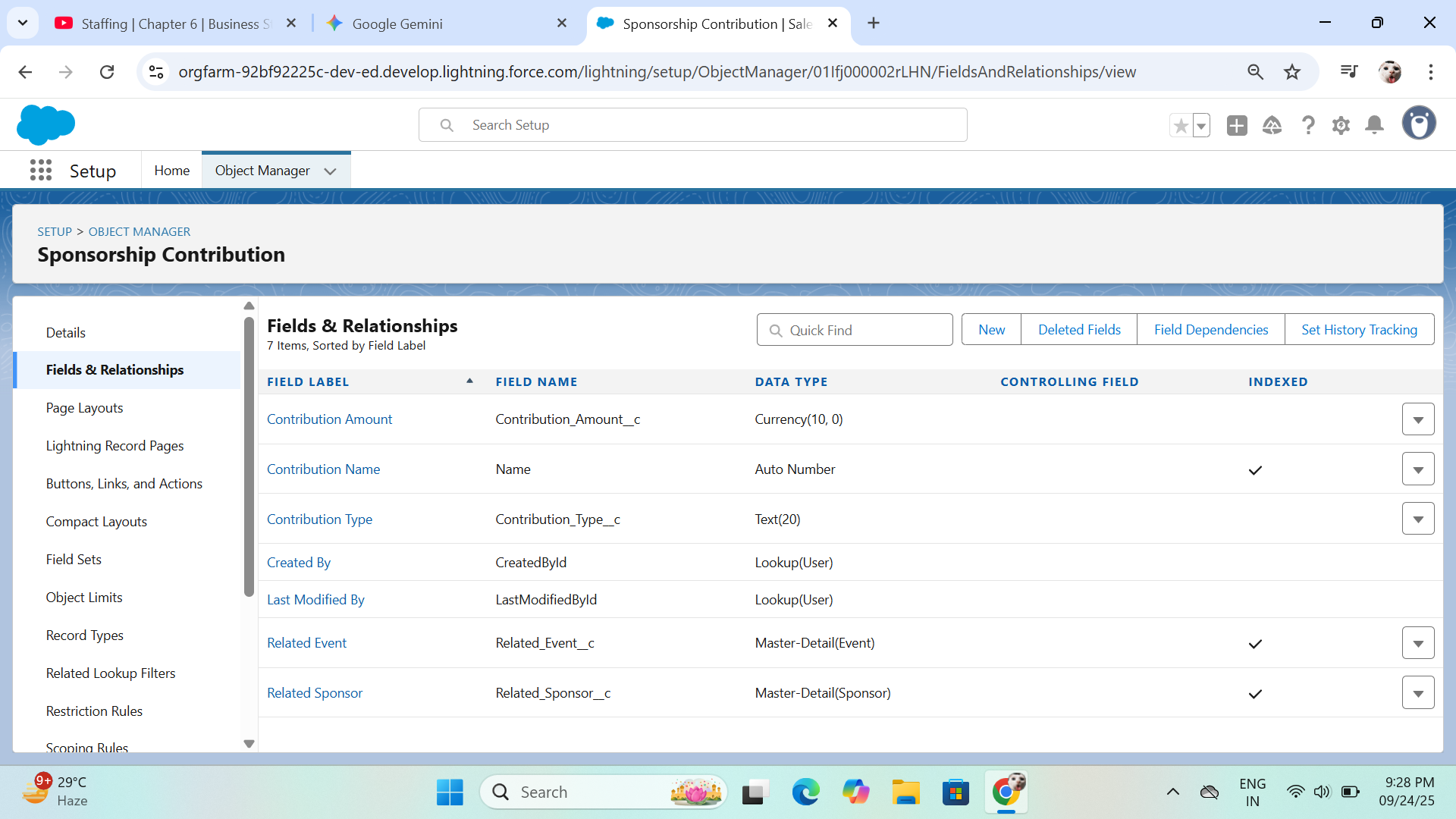
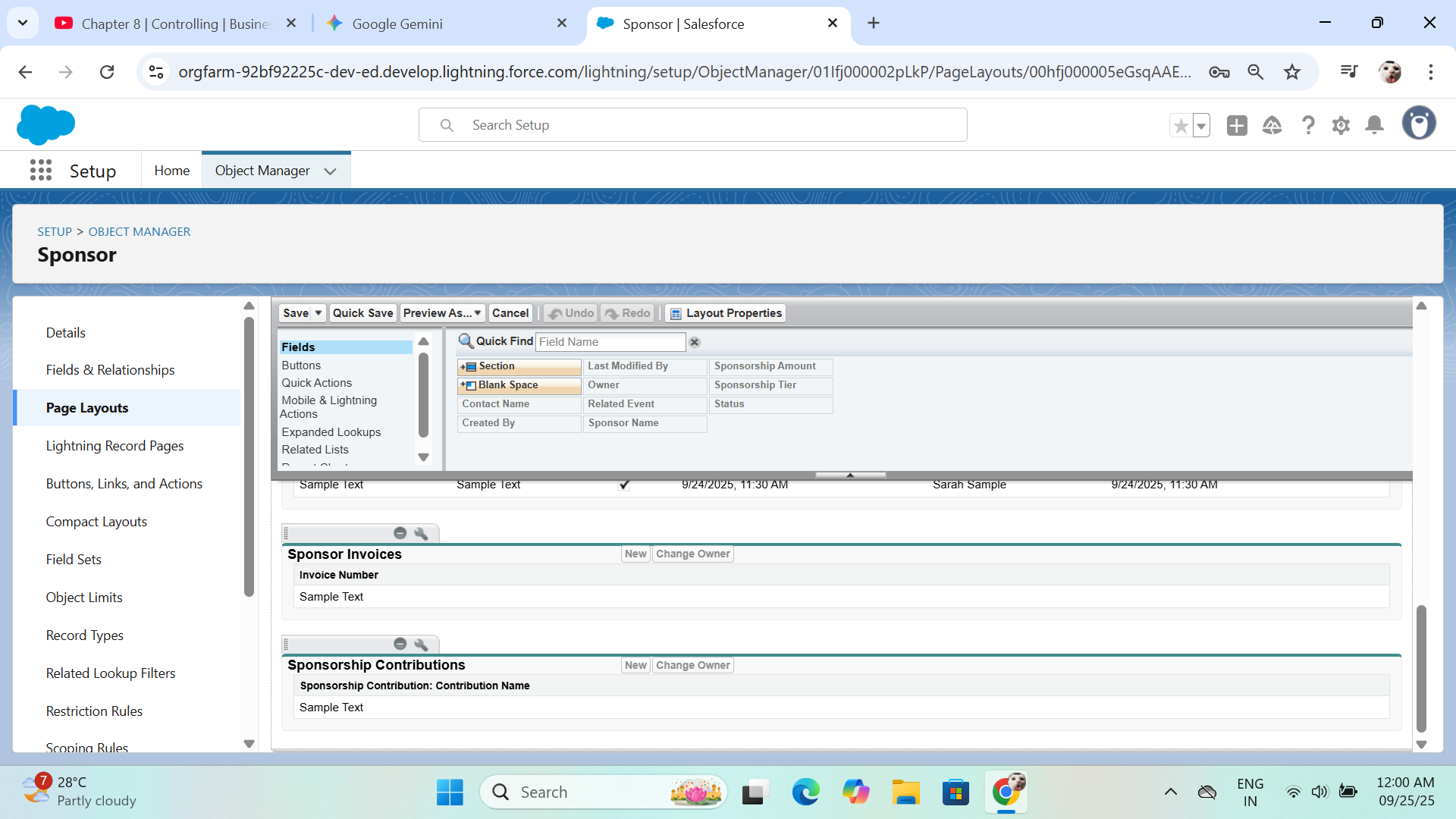
#### Schema Builder

* **Explanation:** The Schema Builder was used as a visual tool to confirm that all objects and their relationships were correctly created and connected.
* **Salesforce Path:** Setup → Quick Find → Schema Builder
* **Procedure & Configuration:**
  + The canvas was loaded with all relevant custom and standard objects.
  + The diagram confirmed that all lookup and master-detail relationships were correctly established, with lines connecting the appropriate objects, confirming the integrity of the data model.



### Junction Objects

* **Explanation:** A junction object is a custom object with two master-detail relationships that is used to create a **many-to-many relationship** between two other objects. In this project, the Sponsorship Contribution object solves the problem of a single sponsor being able to contribute to many events, and a single event having many sponsors. It acts as a bridge to track the specific details of each contribution.
* **Salesforce Path:** Setup → Quick Find → Object Manager → Create → Custom Object
* **Procedure & Configuration:**
  1. **Object Creation:** The Sponsorship Contribution custom object was created. The **Record Name** was configured as an **Auto-Number** with a display format like SC-{0000} to ensure a unique identifier for each contribution.
  2. **Master-Detail Relationships:**
     + A **Master-Detail Relationship** field was created on the Sponsorship Contribution object, linking it to the Event object.
     + A second **Master-Detail Relationship** field was created on the Sponsorship Contribution object, linking it to the Sponsor object.
  3. **Custom Fields:** Fields were created on the Sponsorship Contribution object to store specific data for each contribution:
     + Contribution Amount (Currency)
     + Contribution Type (Text)
  4. **Page Layouts & Related Lists:** The **Sponsorship Contributions** related list was added to both the Event and Sponsor object page layouts.
* **Final Configuration:** The Sponsorship Contribution object now sits between the Event and Sponsor objects, creating a robust and accurate data model. Users can now view all contributions from a single event record or from a single sponsor record.



**External Objects:**

* **Explanation:** These are used to access data from an external system without storing it in Salesforce.
* **Procedure:** As all data will be managed within the Salesforce org, it is not required.