Phase 3
Salesforce Data Model Implementation

Step 1: Create Custom Objects

Navigate to Setup → Object Manager → Create → Custom Object.

Create the following custom objects one by one:

- **Student**
  - Record Name: Student ID (Auto-Number)
  - Enable: Reports & Activities
- **Faculty**
  - Record Name: Faculty ID (Auto-Number)
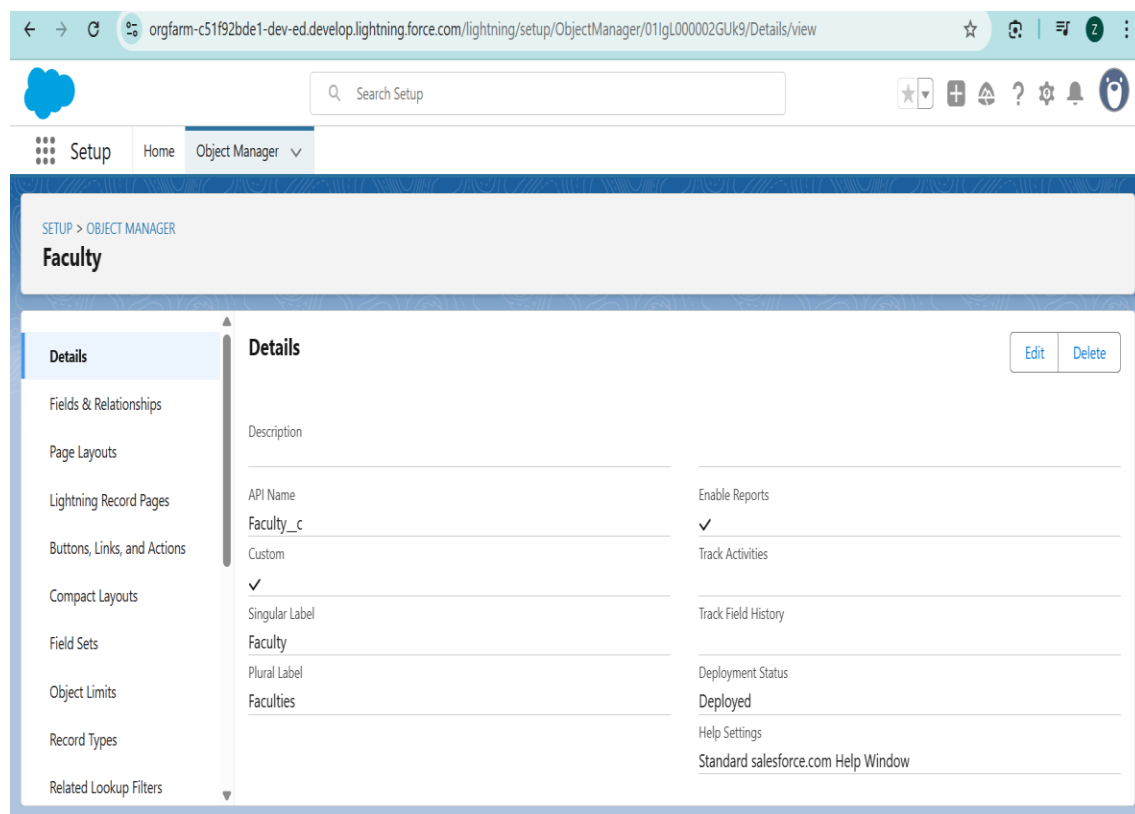 - Enable: Reports & Activities
-

Fig1: Faculty Custom Object

- **Course**
  - Record Name: Course ID (Auto-Number or Text)
  - Enable: Reports & Activities
- **Enrollment**
  - Record Name: Enrollment ID (Auto-Number)
  - Enable: Reports & Activities
- **Fee Payment**
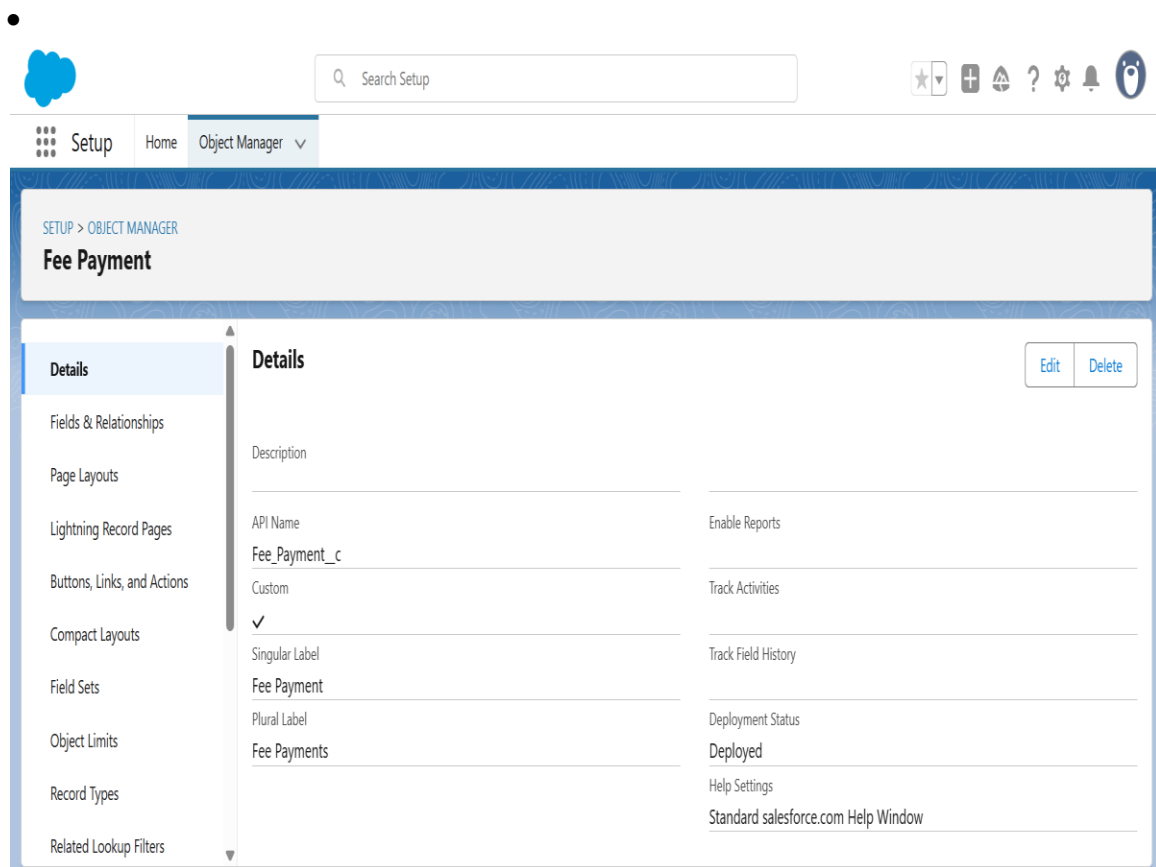 - Record Name: Payment ID (Auto-Number)
  - Enable: Reports & Activities
-

Fig 2:Fee payment custom Object

Step 2: Add Fields to Custom Objects

Student
- Email → Text
- Phone → Phone
- DOB → Date
- Enrollment Status → Picklist (Admitted, Active, Completed, Dropped)
- Course → Lookup (Course)

Fig 3:Fields of Student object

Faculty

- Email → Email
- Phone → Phone
- Department → Picklist (CSE,ECE,EEE,CE,MEC,AIML,AIDS)
- Faculty ID → Auto-Number

Course

- Course Name → Text
- Course Code → Text
- Duration → Number (Months)
- Credits → Number
- Fees → Currency
- Faculty → Lookup (Faculty)

Enrollment

- Student → Lookup (Student)
- Course → Lookup (Course)

- Enrollment Date → Date
- Status → Picklist (Active, Completed, Dropped)
- Grade → Picklist (A, B, C, D, F)

Fig 4: Picklist of Status Field

Fee Payment

- Student → Lookup (Student)
- Payment Date → Date
- Amount → Currency
- Mode of Payment → Picklist (Cash, Card, Online)
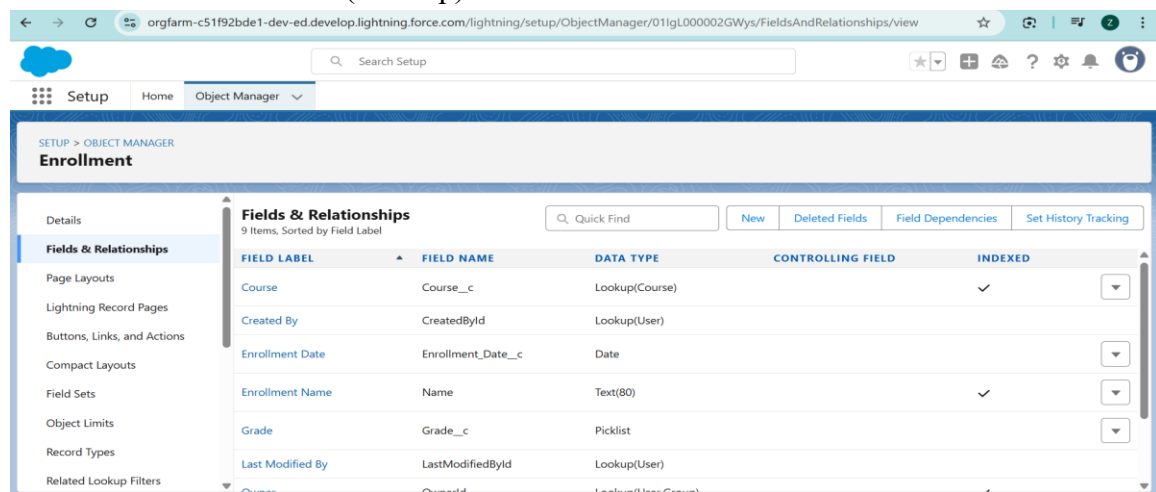- Status → Picklist (Paid, Pending)

Fig 5: Fields of Fee payment

Step 3: Setup Relationships:

Junction Object → Enrollment:
The Enrollment object enables a many-to-many relationship between Students and Courses.

- Enrollment → Student (Lookup)
- Enrollment → Course (Lookup)

Fig 6:Junction Object

- Course → Faculty (Lookup)
- Fee Payment → Student (Lookup)

This creates the following structure:
- One Student → many Courses (via Enrollment)
- One Course → many Students (via Enrollment)
- One Student → multiple Fee Payments
- One Faculty → many Courses

Step 4: Verify Schema with Schema Builder

1. Go to Setup → Schema Builder
   2. Drag & drop objects: Student, Faculty, Course, Enrollment, Fee Payment
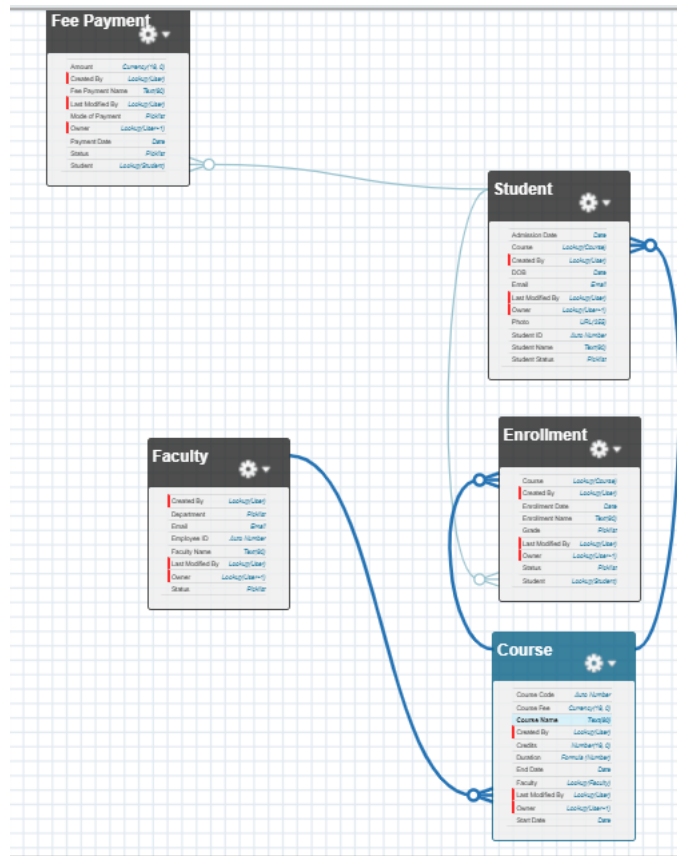   3. Confirm relationships visually.

Fig 7 Schema map

Step 5: Creating Tabs for Objects

In this step, I created **custom objects** in Salesforce to represent the key entities of the Nandini Institute Of Technology. For each object, I also created a **tab** so that users can access and manage records easily from the Salesforce app navigation bar.

1. Navigated to **Setup → Object Manager**.

2. Created a **new custom object** for each entity (Student, Faculty, Course, Fee Payment, Enrollment).
3. For every object, created a **tab** via:
   o Setup → Tabs → New Tab
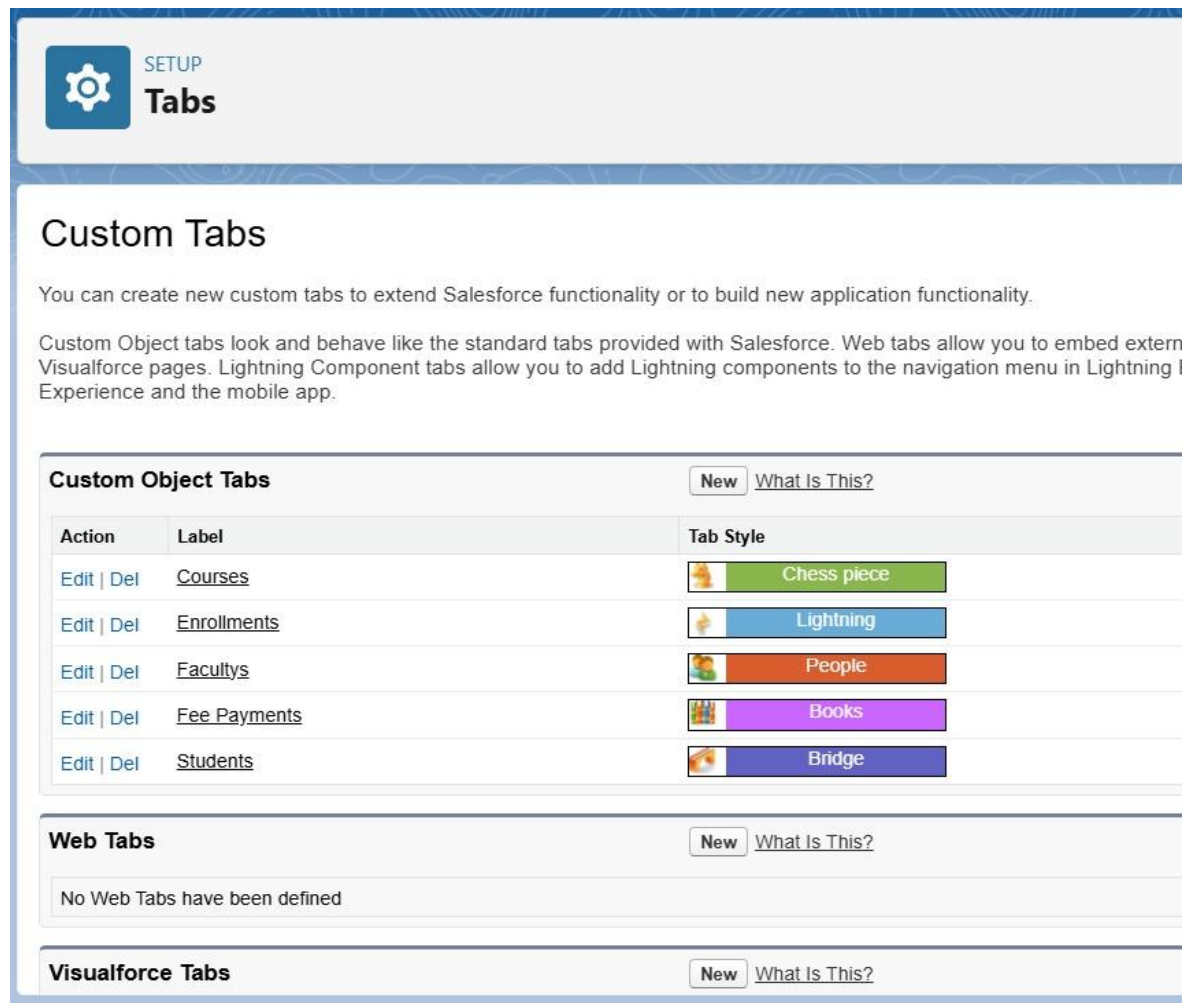   o Selected the object and an icon, then saved.

Fig:Creation of Tabs

Step 6: App Creation :

Step 1: Create a New App

1. Go to **Setup → App Manager**.
2. Click on **New Lightning App**.
3. Enter **App Name** (e.g., *Institute Management System*).
4. Upload an **App Logo** to personalize the app.

7

5. Click **Next** to continue.

Step 2: Configure Navigation

1. Select **Navigation Style → Standard Navigation**.
2. Choose whether the app should be available on **Desktop** or **Desktop & Mobile**.
3. Click **Next**.

Step 3: Assign User Profiles

1. Select the **profiles** that should access this app.
   o Example: System Administrator, Standard User.
2. Click **Save and Finish**.

Step 4: Verify the App

1. Open the **App Launcher** (grid icon in top-left).
2. Search for your app (Nandini Institute Of Technology*)*.
3. Open it and confirm that all required **tabs** (Students, Faculty, Courses, Fee Payment, Enrollments) are visible.
4. Test by creating a sample record in each tab to ensure everything works properly.
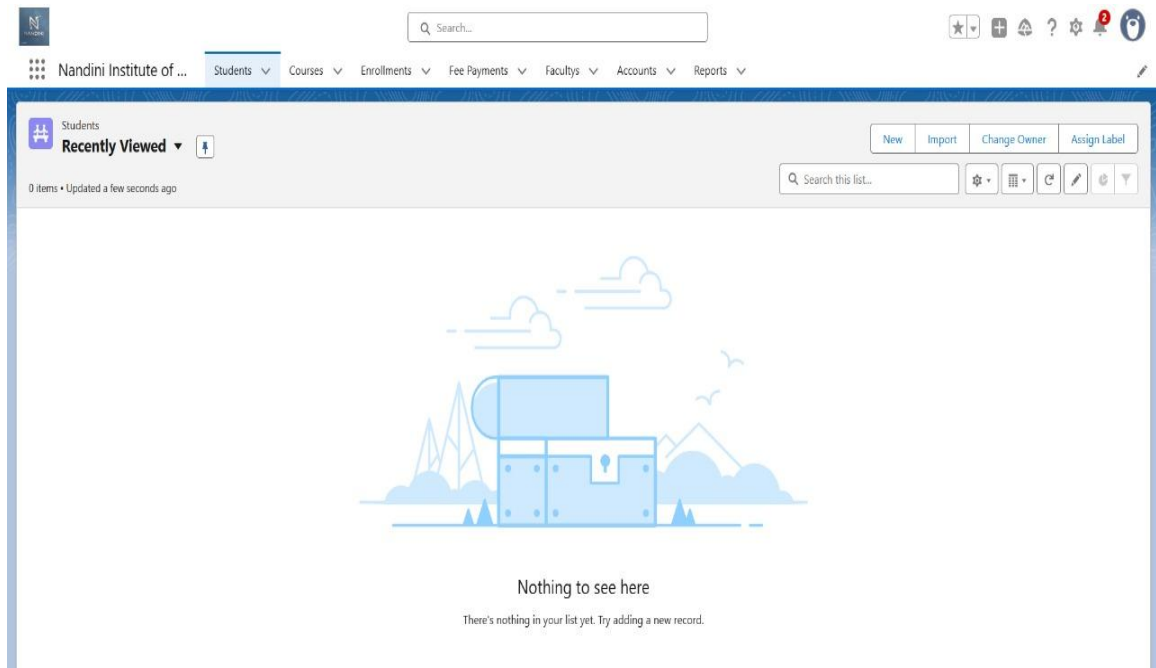
Fig: App creation-Nandini Institute of Technology

Fig: Verification