

Phase 1. Project Title

SN SmartLearn - Student & Course Management System

2. Problem Statement

An online education platform is currently managing student applications, course enrollments, and communications through a fragmented system of spreadsheets and emails. This manual process is inefficient, prone to error, and lacks a centralized view of student data. As the platform grows, this approach is unsustainable, making it difficult to provide a quality student experience, track enrollment trends, and scale operations effectively.

The company requires a robust Salesforce CRM solution to overcome these challenges.

3. Objectives

The primary goals of this Salesforce implementation are to:

- **Automate** the student application and enrollment process to minimize manual errors.
- **Centralize** all student, course, and progress data into a single source of truth.
- **Track** student progress, course history, and assessment results effectively.
- **Streamline** communications with students, instructors, and the admissions team.
- **Enable** real-time dashboards and reports for management to monitor key metrics like enrollment and retention.

4. Stakeholder Analysis

The key stakeholders and their primary needs are identified as follows:

- **Admissions Team:** Needs an efficient system for tracking applications and reducing manual data entry.
- **Course Instructors:** Require easy access to student enrollment lists and progress data.
- **Students:** Expect a smooth, transparent enrollment process and timely, relevant communication.
- **Management:** Wants clear visibility into the admissions funnel, course popularity, and student retention rates for strategic decision-making.
- **IT/Admin:** Responsible for ensuring system stability, data integrity, and security.

2. Business Process Mapping

A comparison of the current and proposed business processes highlights the intended improvements.

Current Process (Before Salesforce)

1. A prospective student submits an application via a web form.
2. An administrator manually enters the application data into a spreadsheet.
3. The admissions team reviews applications from the shared spreadsheet.
4. All communication (updates, requests) is handled via individual emails, which are difficult to track.
5. Course enrollment and progress are logged in separate, disconnected documents.

Proposed Process (After Salesforce Implementation)

1. A student's application from the web form is **automatically captured** as a Lead

record in Salesforce.

2. An automated workflow assigns the application, creates follow-up tasks, and updates its status.
3. Once approved, the Lead is converted into Contact (Student), Account (if applicable), and custom Enrollment records.
4. Automated welcome emails and deadline reminders are sent to students via email alerts.
5. All student data, course history, and progress are tracked in a unified, 360degree view.

3. Industry-Specific Use Case Analysis

The EdTech industry has unique requirements that this project will address:

- **Student Enrollment:** Automatically capture applications from web forms and track the status from submission to enrollment.
- **Course Management:** Maintain a centralized inventory of all courses, including details on modules and assigned instructors.
- **Student Progress Tracking:** Utilize custom objects to log student progress, assignment completion, and grades.
- **Cohort Management:** Group students by program or start date for targeted communication and specialized reporting.
- **Alumni Relations:** Build a foundation to manage relationships with graduates for future engagement and networking opportunities.

4. AppExchange Exploration

To enhance functionality, we will explore solutions on the Salesforce AppExchange:

- **Form Integration Apps (e.g., FormAssembly, Formstack):** To build complex web forms that map directly to Salesforce objects for seamless data capture.
- **Document Generation (e.g., Conga, DocuSign):** For automatically generating and sending enrollment agreements or completion certificates.
- **Enhanced Notification Apps (e.g., Twilio):** To implement SMS/WhatsApp notifications for critical reminders and updates.

5. Conclusion

This initial analysis confirms that a Salesforce CRM implementation is the ideal solution to address SN-SmartLearn's challenges. The project will automate manual processes, create a centralized data system, and provide the analytical tools needed to scale operations and enhance the overall student experience.

Phase 2 - Org Setup & Configuration

1. Salesforce Editions

We used a **Salesforce Developer Edition Org** for this implementation. This edition was selected because it provides all the core CRM features required for our project, such as custom objects, roles, profiles, automation tools, and APIs. It also supports AppExchange integration, which we plan to explore in later phases.

2. Company Profile Setup

- Setup → **Company Information**
- Updated **Organization Name** to SN SmartLearn.
- Set **Default Currency** as INR (Indian Rupees).

- Configured **Locale** as English (India) to ensure formatting of numbers, currency, and dates as per Indian standards.
- Set **Time Zone** to (GMT+5:30) Asia/Kolkata.

This ensures consistency across all student and instructor records, communications, and reports.

The screenshot shows the Salesforce Setup interface. On the left is a navigation menu with options like 'Setup', 'Home', 'Object Manager', 'Company Settings', 'Business Hours', 'Calendar Settings', 'Public Calendars and Resources', 'Company Information', 'Data Protection and Privacy', 'Fiscal Year', 'Holidays', 'Language Settings', 'Manage Currencies', and 'My Domain'. The 'Company Information' section is selected. The main content area displays the 'Company Information' for 'SN SmartLearn'. It includes fields for Organization Name, Primary Contact, Division, Address, Fiscal Year Starts In, Activate Multiple Currencies, Enable Data Translation, Newsletter, Admin Newsletter, Hide Notices About System Maintenance, Hide Notices About System Downtime, and Locale Formats. The 'Default Time Zone' is set to '(GMT+05:30) India Standard Time (Asia/Kolkata)'. The 'Default Locale' is set to 'English (India)'. The 'Default Language' is set to 'English'. The 'Corporate Currency' is set to 'Indian Rupee'. The 'Used Data Space' is 380 KB (7%) and the 'Used File Space' is 19 KB (0%). The 'API Requests, Last 24 Hours' is 0 (15,000 max) and the 'Streaming API Events, Last 24 Hours' is 0 (10,000 max). The 'Restricted Logins, Current Month' is 0 (0 max). The 'Salesforce.com Organization ID' is 000GL000007Vrte and the 'Organization Edition' is Developer Edition. The 'Instance' is CANS6. The page also shows 'User Licenses (10)', 'Permission Set Licenses (10)', 'Feature Licenses (11)', and 'Usage-based Entitlements (10)'. The 'Created By' is OrgFarm EPIC and the 'Modified By' is Nishant Dubey.

3. Business Hours & Holidays

- Setup → **Business Hours** → Created SN SmartLearn Hours as **9:00 AM to 6:00 PM (Mon–Fri)**.
- Setup → **Holidays** → Added major holidays like **Diwali, Republic Day, Independence Day, and New Year**.

These settings ensure that automation processes like case escalations, reminders, and email alerts respect the organization's working schedule.

The screenshot shows the Salesforce Setup interface for configuring Business Hours. The left sidebar includes a search bar with 'business' entered, and a navigation menu with 'Company Settings' and 'Business Hours' selected. The main content area is titled 'Organization Business Hours' and includes a 'Business Hours Edit' section with 'Save' and 'Cancel' buttons. The configuration steps are as follows:

- Step 1. Business Hours Name:** The name is 'SN SmartLearn Hours'. There is a checkbox for 'Active' which is checked, and a checkbox for 'Use these business hours as the default' which is also checked. A red icon indicates 'Required Information'.
- Step 2. Time Zone:** The time zone is set to '(GMT+05:30) India Standard Time (Asia/Kolkata)'.
- Step 3. Business Hours:** A table shows the business hours for each day of the week. Each day has a start time of 9:00 AM and an end time of 6:00 PM, with a checkbox for '24 hours' which is unchecked for all days.

4. Fiscal Year Settings

- Setup → Fiscal Year
- Configured Standard Fiscal Year (April–March) to align with the Indian academic and financial cycle.

This setup ensures reporting and dashboards for admissions, enrollments, and revenue match the organization’s fiscal planning.

The screenshot shows the Salesforce Setup interface for configuring the Fiscal Year. The left sidebar includes a search bar with 'fiscal year' entered, and a navigation menu with 'Company Settings' and 'Fiscal Year' selected. The main content area is titled 'Organization Fiscal Year Edit: SN SmartLearn' and includes a 'Fiscal Year Information' section with a warning message about changing the fiscal year. The configuration steps are as follows:

- Fiscal Year Information:** A warning message states: 'Changing the fiscal year shifts fiscal periods and impacts opportunities and forecasts across your organization. If your forecast periods are set to quarterly, adjusting the fiscal year start month will erase existing forecast adjustments and quotas. Consider exporting a data backup before implementing this change.'
- Change Fiscal Year Period:** The 'Standard Fiscal Year' option is selected. The 'Name' is 'SN SmartLearn'. The 'Fiscal Year Start Month' is set to 'April'. The 'Fiscal Year is Based On' is set to 'The ending month'.

<https://orgfarm-5e644081b6-dev-ed.develop.lightning.force.com/lightning/setup/ForecastFiscalYear/home>

5. User Setup & Licenses

We created different users to represent key stakeholders of the system:

- **Admissions Officer** – Responsible for managing student applications and enrollment.
- **Course Instructor** – Access to course records, enrolled student lists, and progress data.
- **Student (Test User)** – Limited access to check the student experience.

Each user was assigned appropriate licenses (Salesforce / Salesforce Platform) depending on their responsibilities.

6. Profiles

We created custom profiles by cloning the **Standard User Profile** and tailoring objectlevel permissions:

- **Admissions Profile** – Full access to Leads, Contacts, and Enrollment objects.
- **Instructor Profile** – Access to Course and Student Progress objects.
- **Student Profile** – Read-only access to their own course and progress records.

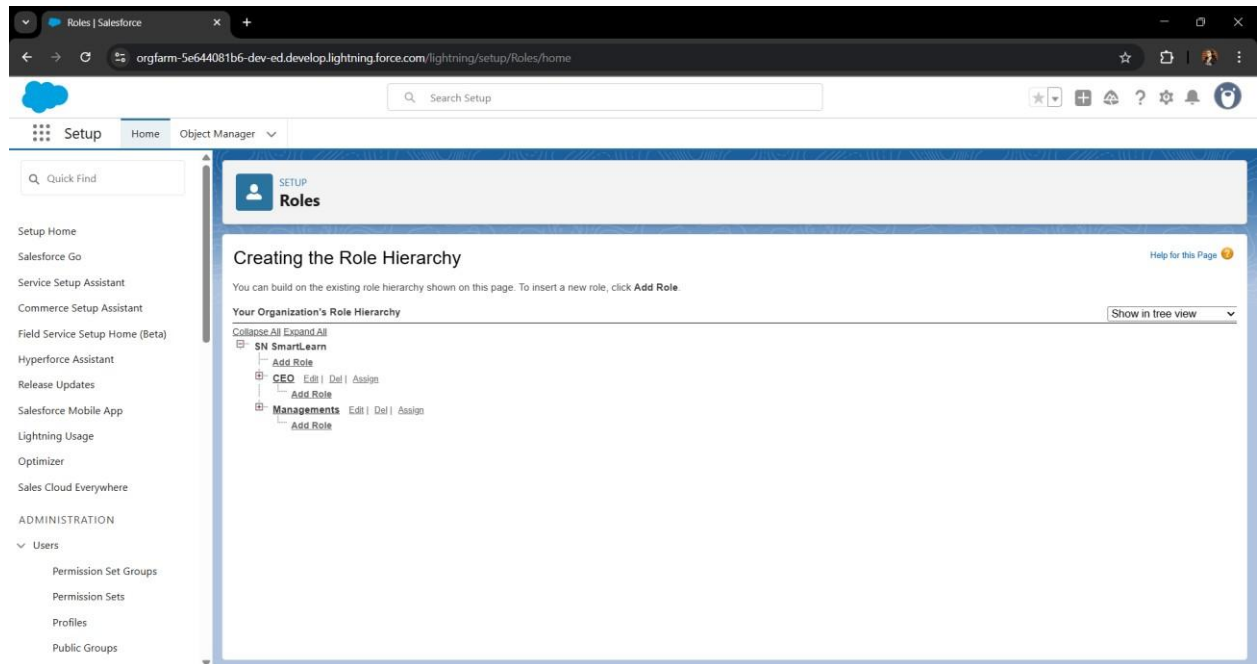
Profiles ensure each role has the exact level of access needed to perform their duties, reducing risks of unauthorized data exposure.

7. Roles

Setup → **Roles** → Created a hierarchy to control data visibility:

- **Management (Top)** ○ **Admissions Head** ○ **Course Instructor** ○ **Students**

This hierarchy ensures managers and admissions heads can view all related data, while instructors and students see only what is relevant to them.



8. Permission Sets

To provide additional, flexible access without altering profiles, we created:

- **Progress Tracking Access** → For instructors to log and monitor student progress.
- **Report Viewer** → For management to access analytical dashboards.

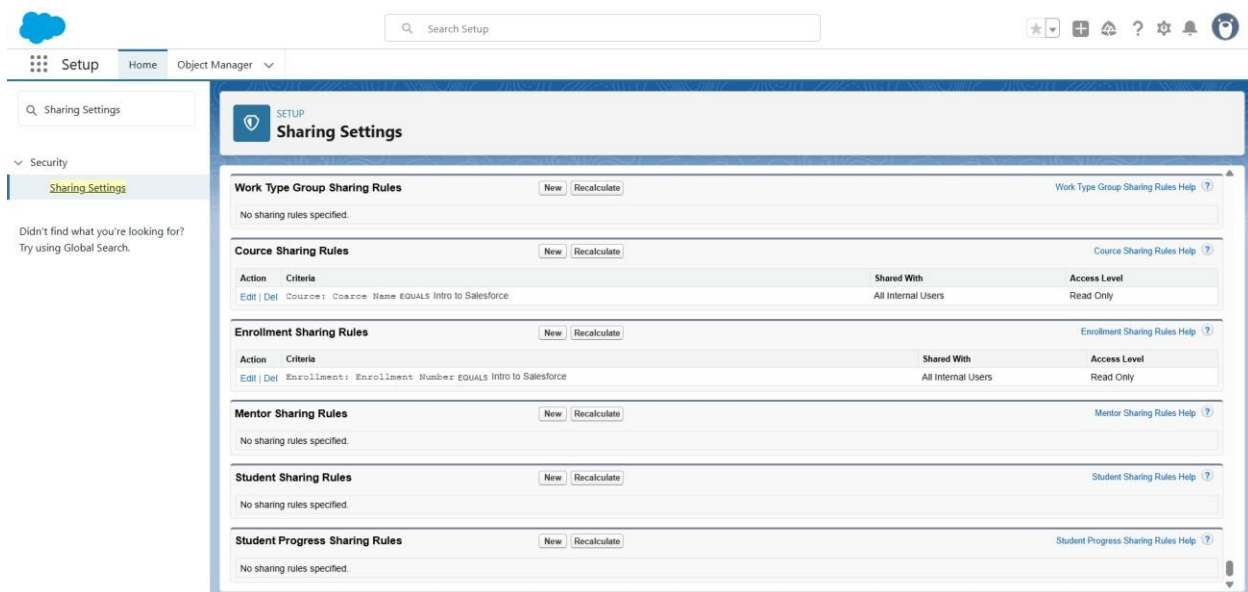
Permission Sets give fine-grained control and can be assigned on a need basis.

9. Org-Wide Defaults (OWD)

Setup → **Sharing Settings** → Configured the following:

- **Students** → Private (students can only view their own records).
- **Courses** → Public Read/Write (so instructors and admins can update them).
- **Enrollments** → Controlled by Parent (data visibility depends on related student/course record).

This enforces data security and ensures confidentiality of student records.



10. Sharing Rules

We implemented sharing rules for controlled data access:

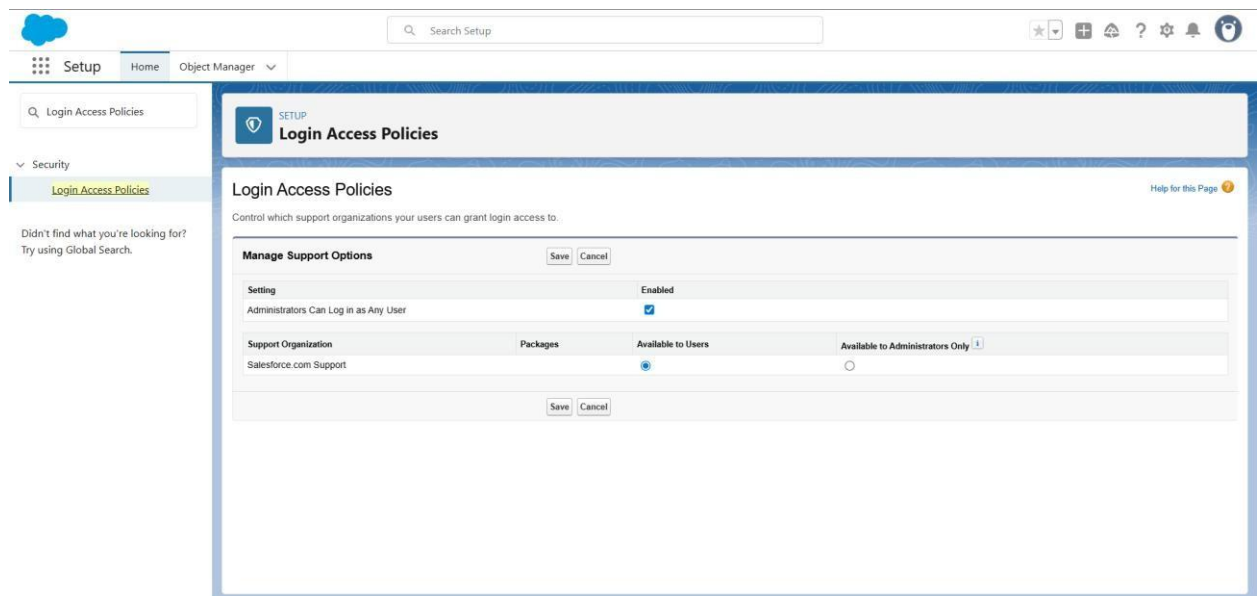
- Admissions users can access all student records to process applications.
- Instructors only see records of students enrolled in their assigned courses.

This prevents unnecessary exposure of sensitive data while enabling collaboration.

11. Login Access Policies

- Setup → **Login Access Policies**
- Enabled **Administrators Can Log in as Any User** to simplify troubleshooting and support. For example, the admin can log in as a student to check if course enrollment processes are working correctly.
- Enabled **Salesforce.com Support Login Access** to allow Salesforce support teams to securely access the org in case of technical issues.

This ensures quick issue resolution and strong governance during system operations.



12. Developer Org Setup

- Create Salesforce Developer Edition account.
- Configure Company Profile, Users, Roles, Profiles, Business Hours, and Security settings.
- Enable required features: custom objects, automation, reports.
- Integrate with GitHub/Salesforce CLI for version control.

13.Sandbox Usage

- Use Developer Sandbox for building and testing changes safely.
- Optionally, use Full Sandbox for testing production-level scenarios.
- Always test major changes in a sandbox before deploying to production.

14.Deployment Basics

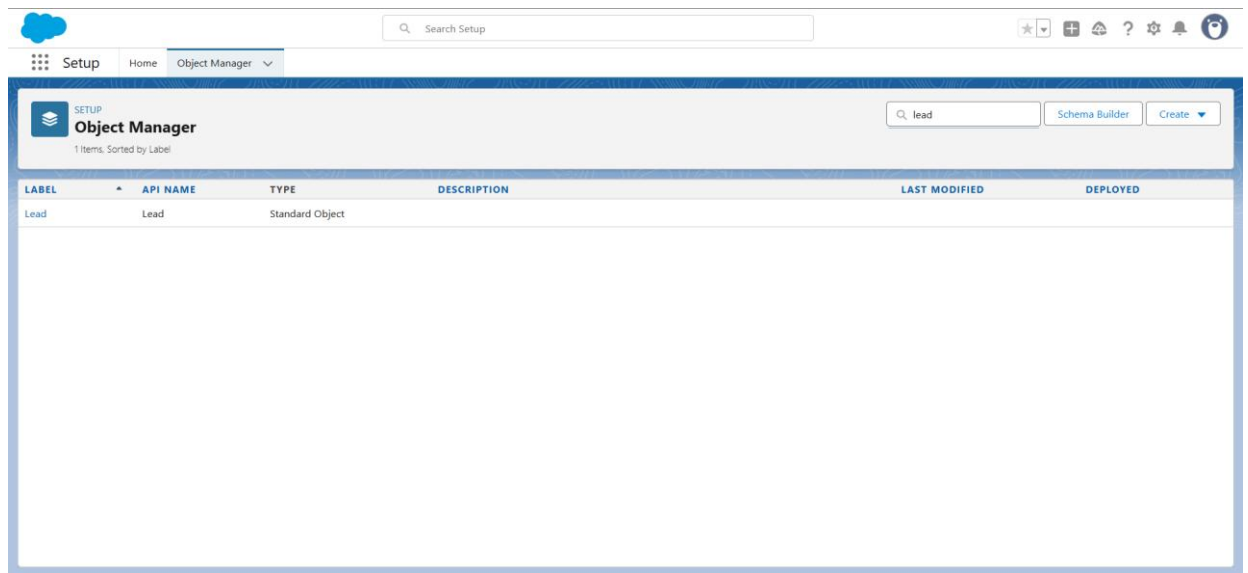
- Change Sets: Simple point-and-click deployment between orgs.
 - Salesforce CLI (SFDX): Advanced deployment with version control and automation.
 - GitHub Integration: Track changes, collaborate, and maintain version control.
- Always document deployment steps and maintain backups

Phase 3 : Data Modeling & Relationships

1. Standard & Custom Objects

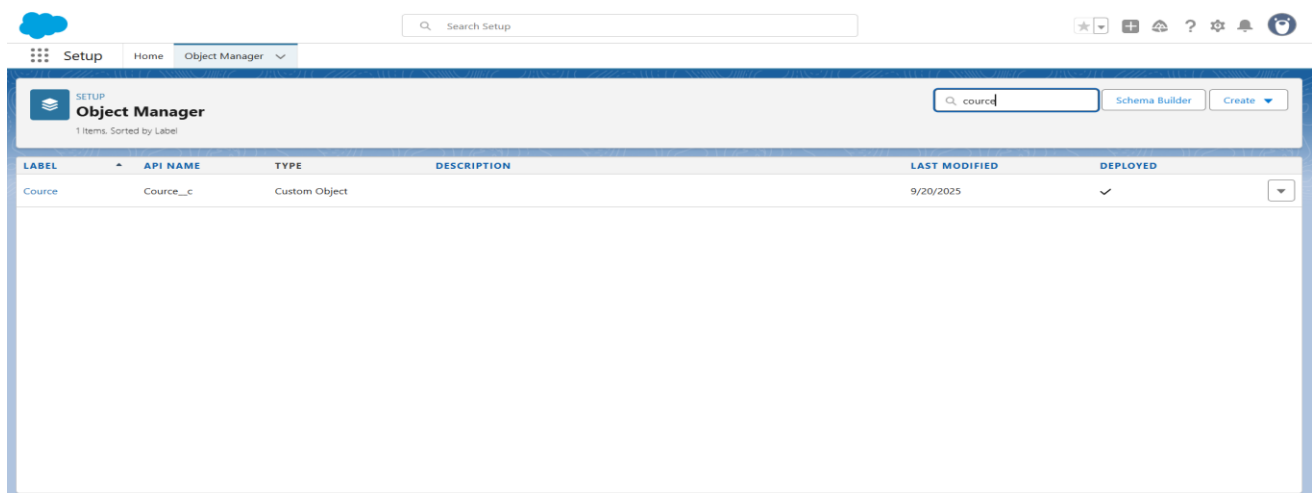
This is the foundation of your system. You will use a combination of standard and custom objects.

- **Standard Objects to Use:**
 - **Lead:** This will be used to automatically capture a student's application from your web form.
 - **Contact:** This will represent the student record after their application is approved and the Lead is converted.
 - **Account:** This can be used to represent the student's household or a sponsoring organization, created during Lead conversion.



Custom Objects to Create:

- **Course:** This is required to maintain a centralized inventory of all courses offered.
- **Enrollment:** This custom object will be created upon Lead conversion to link a student to a specific course.
- **Student Progress:** This object is necessary to log assignment completion and grades, enabling effective progress tracking.



2. Fields

These are the specific data points you will track on each object.

Action Steps:

1. Navigate to **Setup > Object Manager**.
2. Select each custom object (Course, Enrollment, Student Progress) and use the **Fields & Relationships** section to add the following fields:
 - **On the Course object:**
 - Course Code (Data Type: Text, **Unique**)
 - Instructor (Data Type: Lookup to **User**)
 - Status (Data Type: Picklist; Values: Active, Planned, Archived)
 - **On the Enrollment object:**
 - Enrollment Date (Data Type: Date)
 - Status (Data Type: Picklist; Values: Applied, Enrolled, In Progress, Completed, Dropped)
 - **On the Student Progress object:**
 - Assessment Type (Data Type: Picklist; Values: Quiz, Assignment, Final Exam)
 - Grade (Data Type: Percent)
 - Submission Date (Data Type: Date)

Student Progress | Salesforce

orgfarm-5e644081b6-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgL000002lrPh/FieldsAndRelationships/view

Search Setup

Setup Home Object Manager

SETUP > OBJECT MANAGER

Student Progress

Details

Fields & Relationships

6 Items, Sorted by Field Label

Q Quick Find New Deleted Fields Field Dependencies Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Currency	CurrencyIsoCode	Picklist		
Enrollment	Enrollment__c	Lookup(Enrollment)		✓
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Progress ID	Name	Auto Number		✓

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Restriction Rules

Scoping Rules

Object Access

Triggers

Enrollment | Salesforce

orgfarm-5e644081b6-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgL000002lrMT/FieldsAndRelationships/view

Search Setup

Setup Home Object Manager

SETUP > OBJECT MANAGER

Enrollment

Details

Fields & Relationships

6 Items, Sorted by Field Label

Q Quick Find New Deleted Fields Field Dependencies Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Course	Course__c	Master-Detail(Course)		✓
Created By	CreatedById	Lookup(User)		
Currency	CurrencyIsoCode	Picklist		
Enrollment Number	Name	Auto Number		✓
Last Modified By	LastModifiedById	Lookup(User)		
Student	Student__c	Master-Detail(Contact)		✓

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Restriction Rules

Scoping Rules

Object Access

Triggers

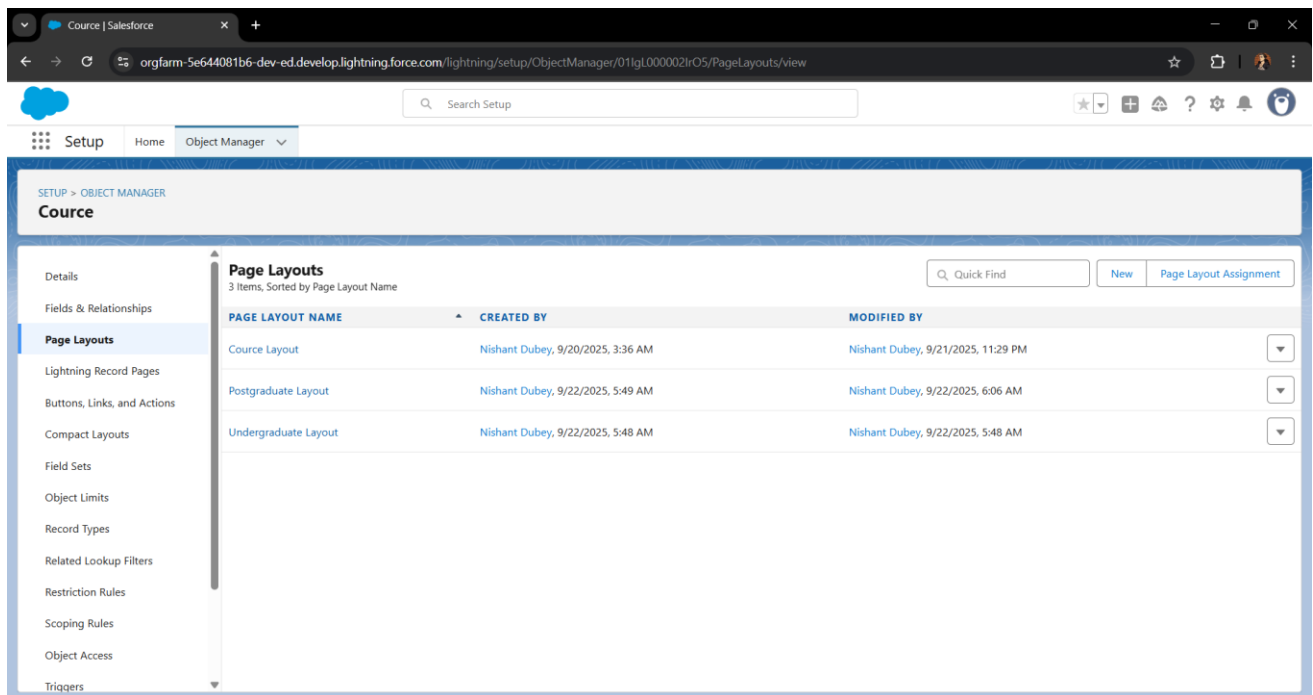
3. Record Types & Page Layouts

These allow you to customize the user experience for different processes. For example, you can create different page layouts on the

Contact object for a prospective student vs. an enrolled student to support the needs of the Admissions Team and Instructors.

Action Steps:

1. Navigate to the **Contact** object and create two **Page Layouts**: one named "Undergraduate Layout" and another named "Postgraduate Layout".
2. On the **Contact** object, go to **Record Types** and created new record type: "Thesis Required", assigning the corresponding layout.



The screenshot shows the Salesforce Setup interface for the 'Course' object. The left sidebar contains a navigation menu with options: Details, Fields & Relationships, Page Layouts (selected), Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Restriction Rules, Scoping Rules, Object Access, and Triggers. The main content area is titled 'Page Layouts' and shows a table with 3 items, sorted by Page Layout Name. The table has columns for PAGE LAYOUT NAME, CREATED BY, and MODIFIED BY. The items listed are 'Course Layout', 'Postgraduate Layout', and 'Undergraduate Layout', all created by Nishant Dubey. There are 'New' and 'Page Layout Assignment' buttons at the top right of the table.

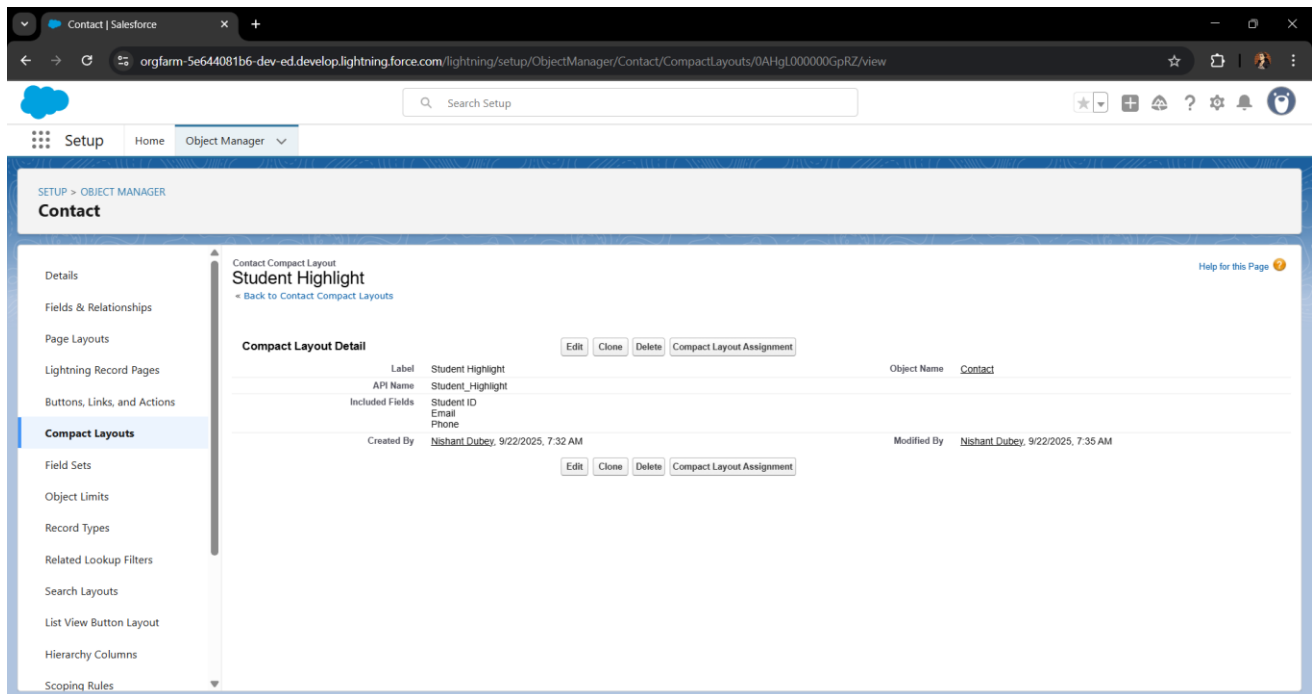
PAGE LAYOUT NAME	CREATED BY	MODIFIED BY
Course Layout	Nishant Dubey, 9/20/2025, 3:36 AM	Nishant Dubey, 9/21/2025, 11:29 PM
Postgraduate Layout	Nishant Dubey, 9/22/2025, 5:49 AM	Nishant Dubey, 9/22/2025, 6:06 AM
Undergraduate Layout	Nishant Dubey, 9/22/2025, 5:48 AM	Nishant Dubey, 9/22/2025, 5:48 AM

4. Compact Layouts

This controls the highlights panel at the top of a record.

Action Steps:

1. Navigate to the **Contact** object and go to **Compact Layouts**.
2. Create a new layout named "Student View".
3. Add key fields like **Name**, **Email**, and **Phone**.
4. Use **Compact Layout Assignment** to make this the primary layout.

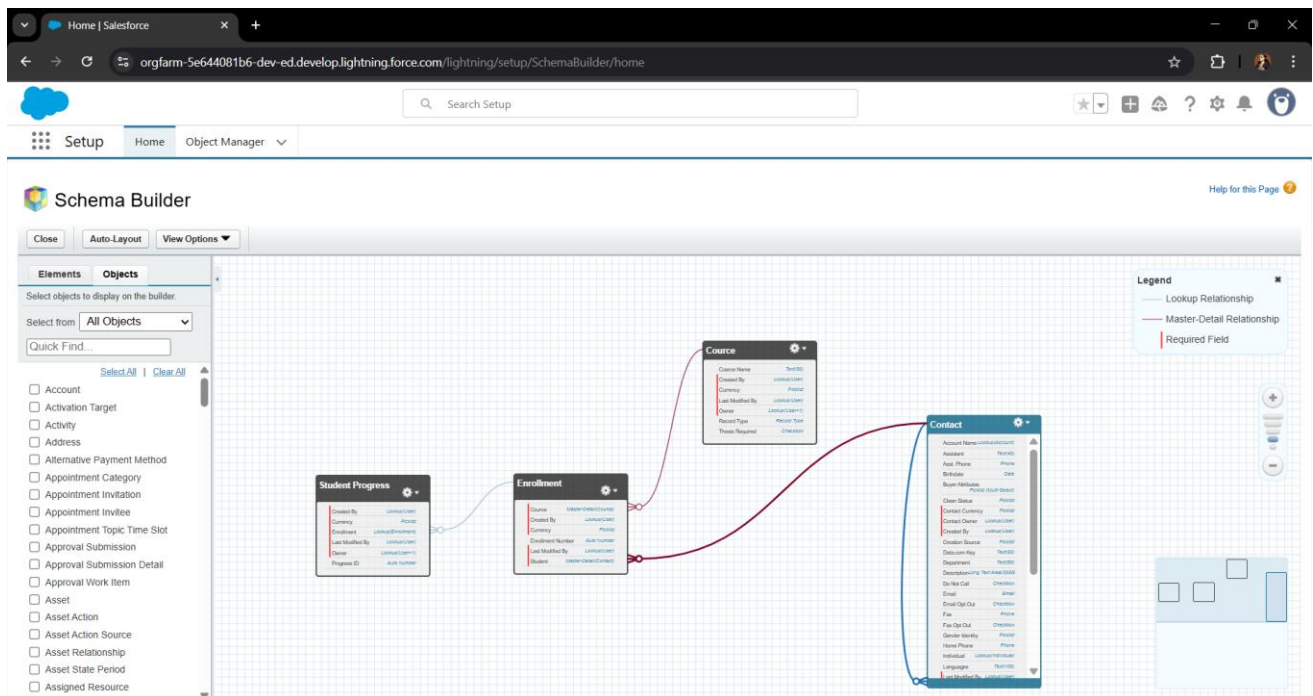


5. Schema Builder

Use this tool to visualize your completed data model.

Action Steps:

1. Go to **Setup > Schema Builder**.
2. Select your objects: **Lead**, **Contact**, **Account**, **Course**, **Enrollment**, and **Student Progress**.
3. Review the diagram to visually confirm the relationships you built.



6.Relationships & Junction Objects

These relationships will connect your objects to create the 360-degree view of the student mentioned in your project plan.

- **Junction Object:** Your **Enrollment** object is the junction object. It connects Students (Contacts) and Courses, creating a many-to-many relationship.

Action Steps:

1. On the **Enrollment** object, create two **Master-Detail Relationship** fields:
 - One that links to the **Contact** object (label it Student).
 - A second one that links to the **Course** object (label it Course).
2. On the **Student Progress** object, create a required **Lookup Relationship** that links to the **Enrollment** object.

7. External Objects

This is a conceptual topic for this project. External Objects allow you to view data from other systems. For example, if your platform used an external library management system, you could create an External Object to display a student's checked-out books within Salesforce without actually storing that data.