

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, the 'Setup' menu is open, and 'Company Information' is selected under 'Company Settings'. The main content area displays the 'Company Information' page for 'SN SmartLearn'. The page includes a search bar at the top, a navigation menu on the left, and a main content area with tabs for 'Edit' and 'Currency Setup'. The 'Edit' tab is active, showing various organization details. A tooltip is visible over the 'Company Information: SN SmartLearn - Salesforce - Developer Edition' link.

Organization Detail		Phone	
Organization Name	SN SmartLearn	Phone	
Primary Contact	Nishant Dubey	Fax	
Division		Default Locale	English (India)
Address	Madhya Pradesh, India	Default Language	English
Fiscal Year Starts In	April	Default Time Zone	(GMT+05:30) India Standard Time (Asia/Kolkata)
Activate Multiple Currencies	<input checked="" type="checkbox"/>	Corporate Currency	Indian Rupee
Enable Data Translation	<input type="checkbox"/>	Used Data Space	380 KB (7%) View
Newsletter	<input checked="" type="checkbox"/>	Used File Space	19 KB (0%) View
Admin Newsletter	<input checked="" type="checkbox"/>	API Requests, Last 24 Hours	0 (15,000 max)
Hide Notices About System Maintenance	<input type="checkbox"/>	Streaming API Events, Last 24 Hours	0 (10,000 max)
Hide Notices About System Downtime	<input type="checkbox"/>	Restricted Logins, Current Month	0 (0 max)
Locale Formats	ICU	Salesforce.com Organization ID	00DGL000007Vite
		Organization Edition	Developer Edition
		Instance	CAN36

Created By: OrgFarm EPIC, 7/18/2025, 4:16 AM
Modified By: Nishant Dubey, 9/20/2025, 4:21 AM

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 'Business Hours Name' field is set to 'SN SmartLearn Hours'. The 'Active' checkbox is checked. There is a 'Required Information' error indicator.
- Step 2. Time Zone:** The 'Time Zone' dropdown 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 (9:00 AM) and an end time (6:00 PM) field, followed by a checkbox for '24 hours'.

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. 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:** A modal window shows the 'Change Fiscal Year Period' section with 'Save' and 'Cancel' buttons. The 'Name' field is set to 'SN SmartLearn'. The 'Fiscal Year Start Month' dropdown is set to 'April'. The 'Fiscal Year is Based On' radio buttons are set to 'The ending month'.

<https://orgfarm-56544081b6-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.

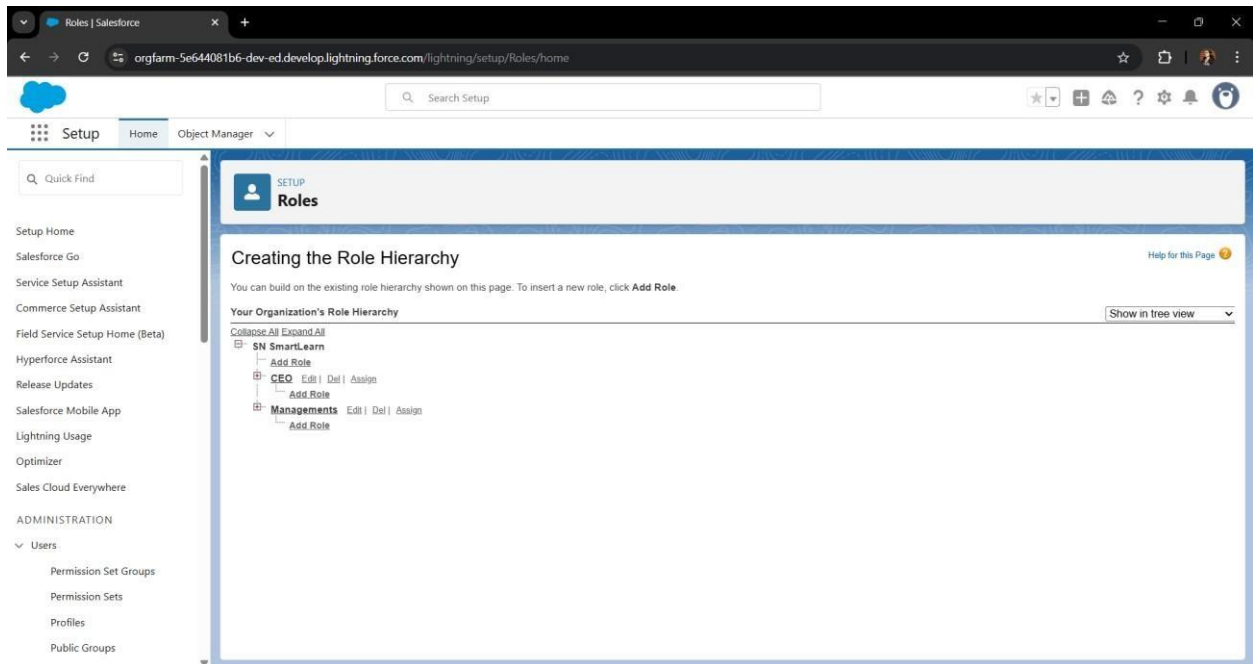
Profiles ensure each role has the exact level of access needed to perform their duties, reducing risks of unauthorized data exposure.

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.

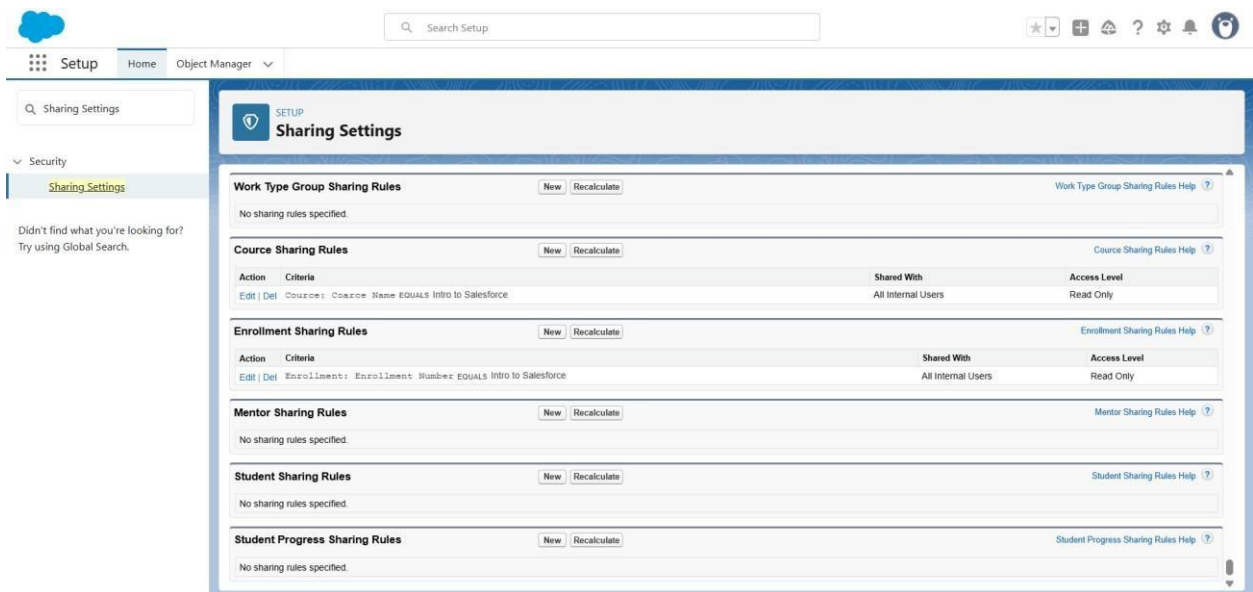
Permission Sets give fine-grained control and can be assigned on a need basis.

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.

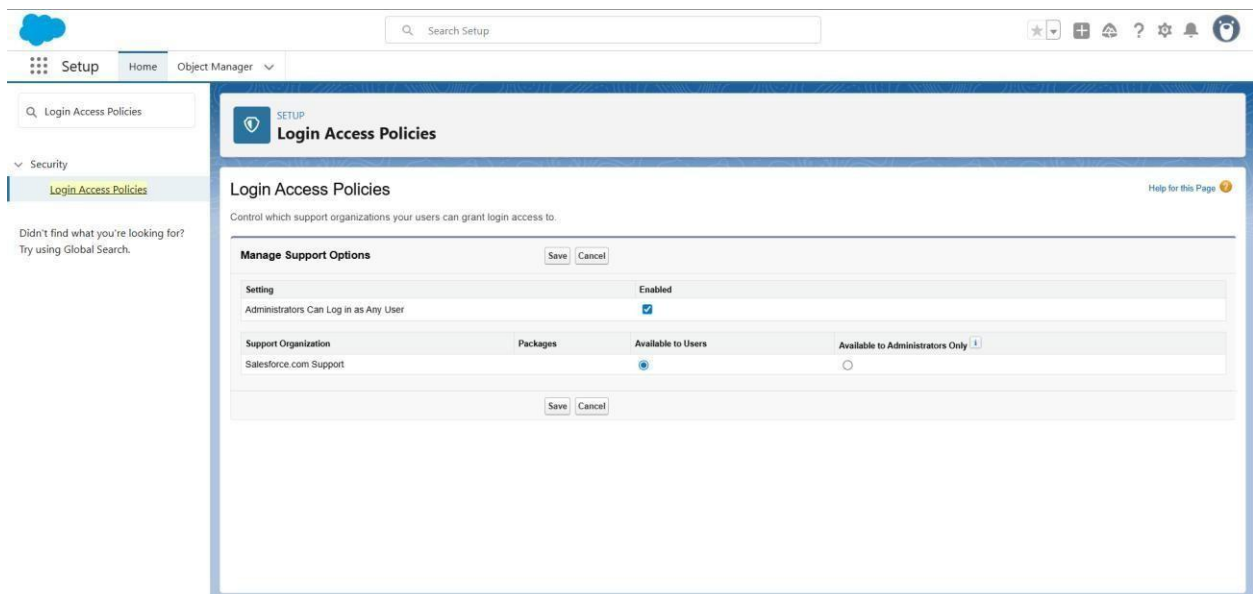
This prevents unnecessary exposure of sensitive data while enabling collaboration.

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

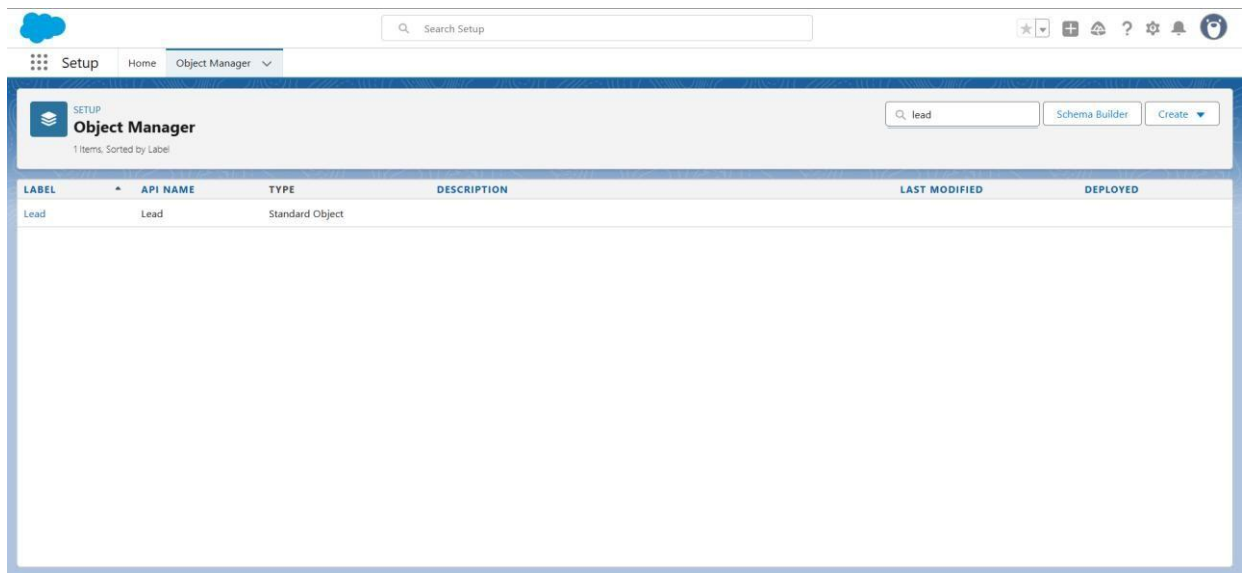
- Change Sets: Simple point-and-click deployment between orgs.
 - Salesforce CLI (SFDX): Advanced deployment with version control and automation.
 - GitHub Integration: Track changes, collaborate, and maintain version control.
- Always document deployment steps and maintain backups

Phase 3 : Data Modeling & Relationships

1. Standard & Custom Objects

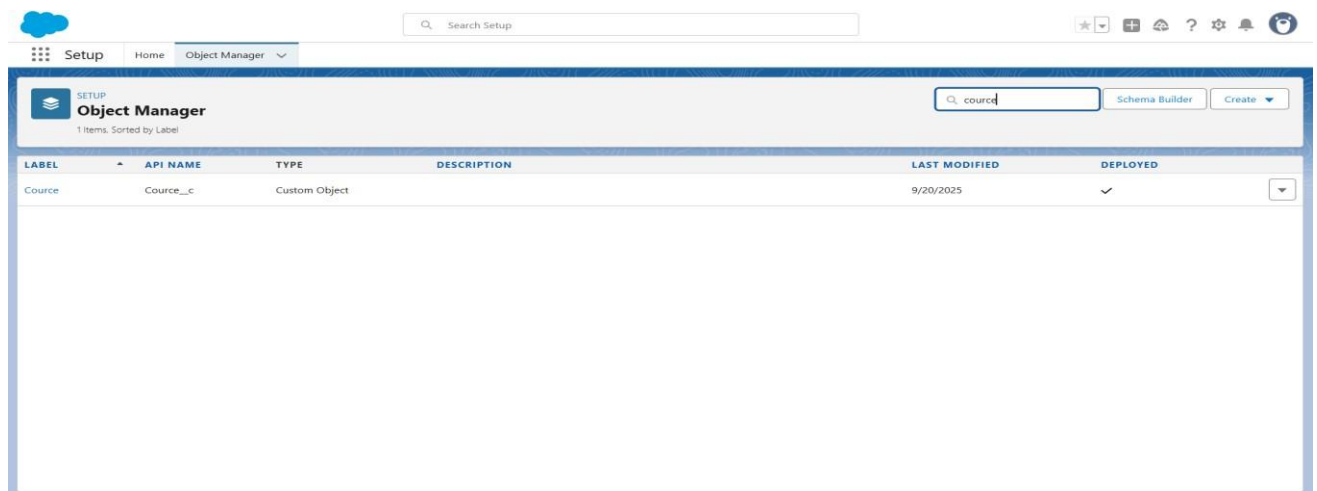
This is the foundation of your system. You will use a combination of standard and custom objects.

- **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

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

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		✓

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

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

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)		✓

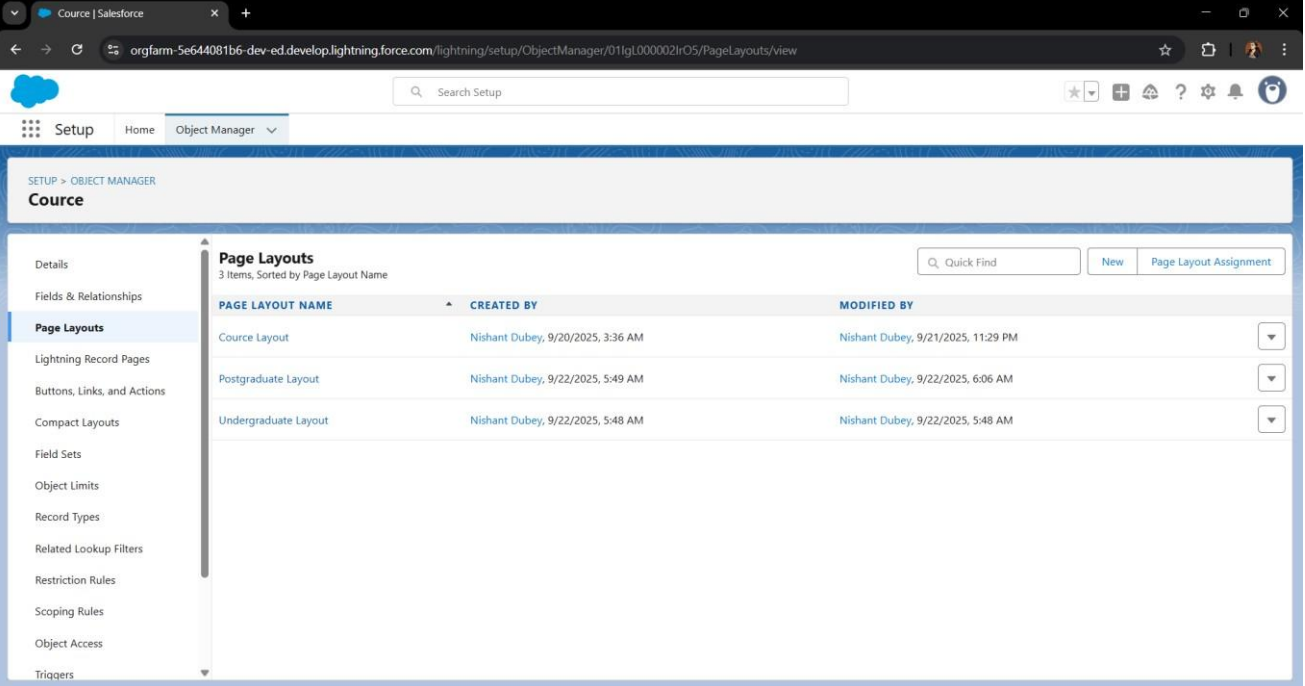
3. Record Types & Page Layouts

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 list of 3 items, sorted by Page Layout Name. The list includes 'Course Layout', 'Postgraduate Layout', and 'Undergraduate Layout'. Each row displays the layout name, the user who created it (Nishant Dubey), the creation date and time, and the last modified date and time. There are also 'New' and 'Page Layout Assignment' buttons at the top right of the list.

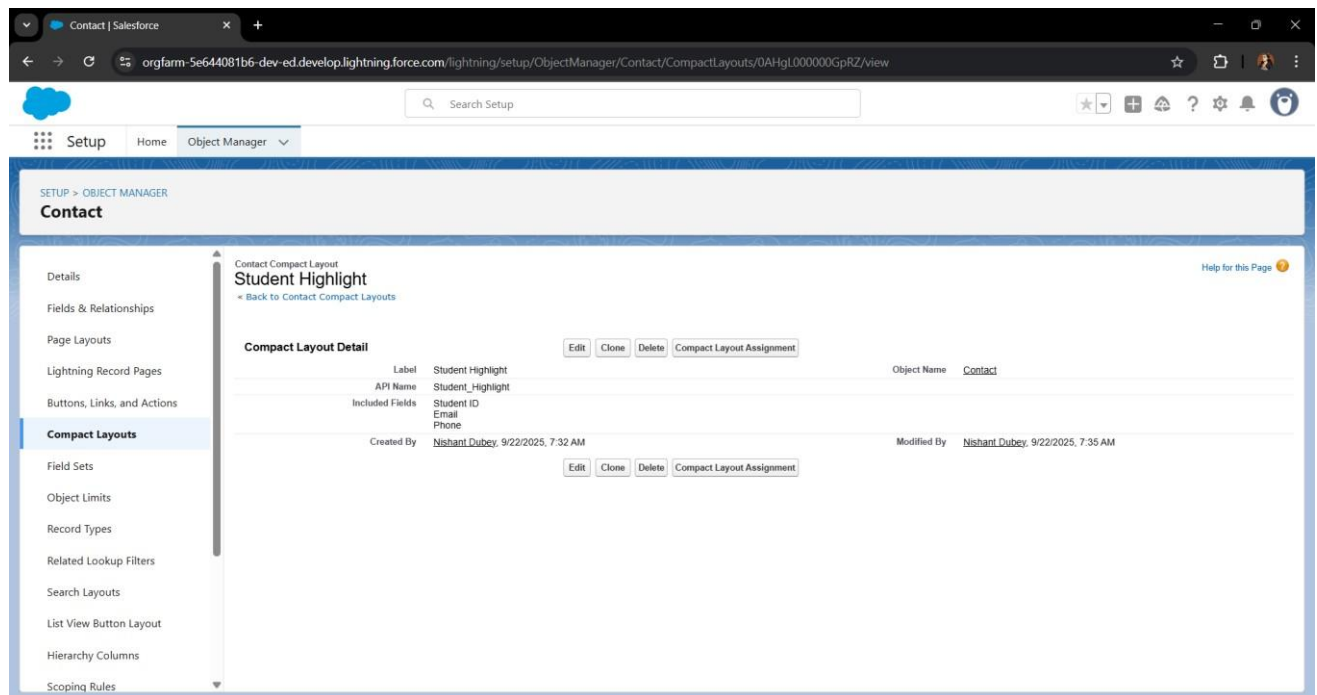
PAGE LAYOUT NAME	CREATED BY	MODIFIED BY
Course Layout	Nishant Dubey, 9/20/2025, 3:36 AM	Nishant Dubey, 9/21/2025, 11:29 PM
Postgraduate Layout	Nishant Dubey, 9/22/2025, 5:49 AM	Nishant Dubey, 9/22/2025, 6:06 AM
Undergraduate Layout	Nishant Dubey, 9/22/2025, 5:48 AM	Nishant Dubey, 9/22/2025, 5:48 AM

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.

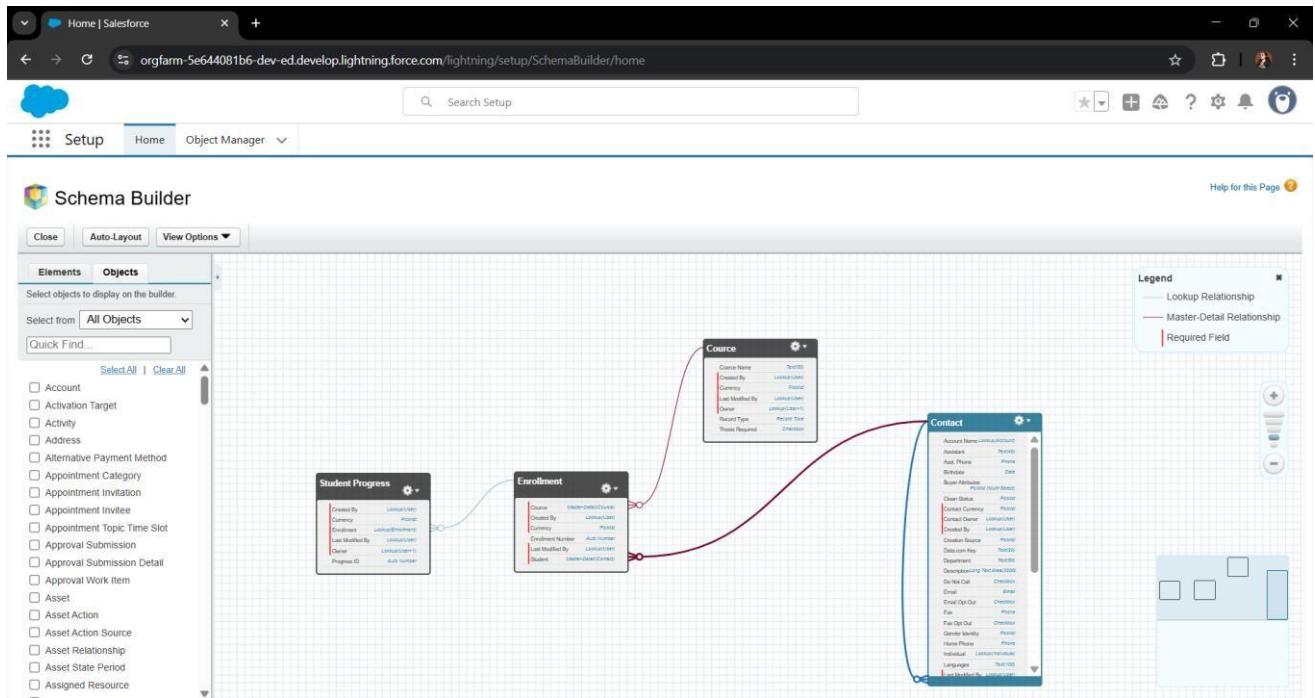


5. Schema Builder

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.

Phase 4 : Process Automation (Admin)

In this phase, I implemented Salesforce automation tools to streamline business processes for SN SmartLearn. Each tool was configured with clear use cases to reduce manual work, ensure data accuracy, and improve the student/admissions experience.

1. Validation Rules

Validation Rules enforce data integrity by preventing users from saving records with invalid values.

Use Case: Prevent a Grade on a *Student Progress* record from being greater than 100%.

Steps Implemented:

1. Setup → Object Manager → Student Progress.
2. Open **Validation Rules** → **New Rule**.

3. Rule Name: Grade_Cannot_Exceed_100.
4. Error Condition Formula:
5. Grade__c > 100
6. Error Message: *"A grade cannot be greater than 100%."*
7. Save and Activate.

This ensures grading remains within the correct range and prevents data entry errors.

The screenshot displays the Salesforce Setup interface. The top navigation bar includes the Salesforce logo, a search bar labeled "Search Setup", and various utility icons. Below the navigation bar, the "Setup" menu is expanded, showing options like Home, Object Manager, and others. The "Object Manager" section is selected, and the "Student Progress" object is chosen. The main content area shows the "Student Progress Validation Rule" configuration page. On the left, a sidebar lists various setup categories, with "Validation Rules" selected. The main panel displays the "Validation Rule Detail" for the "Grade_Cannot_Exceed_100" rule. The rule is active, as indicated by a checkmark in the "Active" column. The "Error Condition Formula" is "Grade__c > 100", and the "Error Message" is "A grade cannot be more than 100%". The "Error Location" is set to "Top of Page". The "Created By" and "Modified By" fields both show "Nishant Dubey, 9/22/2025, 10:09 AM".

Validation Rule Detail		Active
Rule Name	Grade_Cannot_Exceed_100	<input checked="" type="checkbox"/>
Error Condition Formula	Grade__c > 100	
Error Message	A grade cannot be more than 100%.	Error Location: Top of Page
Description		
Created By	Nishant Dubey, 9/22/2025, 10:09 AM	Modified By: Nishant Dubey, 9/22/2025, 10:09 AM

2. Workflow Rules (Legacy Tool)

Workflow Rules are an older automation tool, now replaced by Flow Builder, but documented here for completeness.

Use Case: Automatically create a follow-up Task for Admissions when a new *Student Application (Lead)* is created.

Steps Implemented:

1. Setup → Workflow Rules → New Rule.
2. Object: **Lead**.
3. Rule Name: *Create Task for New Application*.
4. Evaluation Criteria: *Created*.
5. Rule Criteria: None (runs for every new Lead).
6. Immediate Workflow Action → New Task:
 - Assigned To: Admissions Team User.
 - Subject: *Follow up on new application*.
 - Due Date: Lead Created Date + 7 days.
 -
7. Save → Done → Activate Rule.

The screenshot shows the Salesforce Setup interface for configuring a Workflow Rule. The left sidebar contains navigation links: Setup, Home, Object Manager, Process Automation, Workflow Actions (with sub-links for Email Alerts, Field Updates, Outbound Messages, Send Actions, and Tasks), and Workflow Rules (highlighted). The main content area is titled 'Workflow Rules' and shows a specific rule named 'Create Task for New Application'. A yellow banner at the top provides information about the Flow Builder. The 'Workflow Rule Detail' section includes fields for Rule Name, Active status, Object, Evaluation Criteria, Description, Rule Criteria, Created By, and Modified By. The 'Workflow Actions' section shows an immediate action of type 'Task' with the description 'Follow up on new application'. A warning message at the bottom states: 'You cannot add new time triggers to an active rule. Deactivate This Rule'.

Setup

Search Setup

workflow

Process Automation

Workflow Actions

Email Alerts

Field Updates

Outbound Messages

Send Actions

Tasks

Workflow Rules

Didn't find what you're looking for? Try using Global Search.

Workflow Rules

Workflow Rule

Create Task for New Application

Go with the flow! With Flow Builder, the future of low-code automation, you can do everything you do with workflow rules — and more! Salesforce plans to retire workflow rules and recommends building automation in Flow Builder. [Tail Me More](#) | [Migrate your workflow rules to Flow](#)

Workflow Rule Detail [Edit](#) [Clone](#) [Deactivate](#)

Rule Name	Create Task for New Application	Object	Lead
Active	✓	Evaluation Criteria	Evaluate the rule when a record is created
Description			
Rule Criteria	Lead: Created Date NOT EQUAL TO NULL		
Created By	Nishant Dubey, 9/22/2025, 10:17 AM	Modified By	Nishant Dubey, 9/22/2025, 10:21 AM

Workflow Actions [Edit](#)

Immediate Workflow Actions

Type	Description
Task	Follow up on new application

Time-Dependent Workflow Actions [See an example](#)

Warning: You cannot add new time triggers to an active rule. [Deactivate This Rule](#)

3. Process Builder (Legacy Tool)

Process Builder was used to automate record updates. It is now deprecated, but included here for legacy documentation.

Use Case: When *Enrollment* status changes to “Completed,” update the related *Contact (Student)* record’s status.

Steps Implemented:

1. Setup → Process Builder → New.
2. Process Name: *Update Student Status on Course Completion.*
3. Start Process: *When a record changes.*
4. Object: **Enrollment**. Trigger: Created or Edited.
5. Add Criteria:
 - Criteria Name: *Course Completed.*
 - Conditions:
 - Status__c Is Changed = True.
 - Status__c Equals = “Completed”.
6. Immediate Action → Update Records:
 - Record Type: Student (related Contact).
 - New Value: Enrollment_Status__c = "Completed".
7. Save → Activate.

Setup

Home

Object Manager

Search Setup

Go with the flow! With Flow Builder, the future of low-code automation, you can do everything you do with Process Builder—and more! Salesforce plans to retire Process Builder and recommends building automation in Flow Builder.

Try Flow Builder | Use Migrate to Flow Tool

Process Builder - Update Student Status on Course Completion

Back To Setup

Help

Expand All

Collapse All

View All Processes

Clone

View Properties

Deactivate

Read Only

START

Enrollment

Course Completed

TRUE

IMMEDIATE ACTIONS

Update Contact to A...

STOP

FALSE

+ Add Criteria

TRUE

IMMEDIATE ACTIONS

+ Add Action

FALSE

STOP

Update Records

Action Name*

Update Contact to Alumnus

Record*

[Enrollment_c].Student

Criteria for Updating Records*

Updated records meet all conditions

No criteria—just update the records!

Set new field values for the records you update

Field*	Type*	Value*
Enrollment Status	Picklist	Completed

Save

Cancel

Setup

Home

Object Manager

Search Setup

Go with the flow! With Flow Builder, the future of low-code automation, you can do everything you do with Process Builder—and more! Salesforce plans to retire Process Builder and recommends building automation in Flow Builder.

Try Flow Builder | Use Migrate to Flow Tool

Process Builder - Update Student Status on Course Completion

Back To Setup

Help

Expand All

Collapse All

View All Processes

Clone

View Properties

Deactivate

Read Only

START

Enrollment

Course Completed

TRUE

IMMEDIATE ACTIONS

Update Contact to A...

STOP

FALSE

+ Add Criteria

TRUE

IMMEDIATE ACTIONS

+ Add Action

FALSE

STOP

Define Criteria for this Action Group

Criteria Name*

Course Completed

Criteria for Executing Actions*

Conditions are met

Formula evaluates to true

No criteria—just execute the actions!

Set Conditions

	Field*	Operator*	Type*	Value*
1	[Enrollment_c]....	Is changed	Boolean	True
2	[Enrollment_c]....	Equals	Picklist	Completed

Conditions*

All of the conditions are met (AND)

Any of the conditions are met (OR)

Customize the logic

Save

Cancel

4. Approval Process

Approval Processes enable formal sign-off flows for records.

Use Case: Require management approval for enrollments with discounts above 20%.

Steps Implemented:

1. Setup → Approval Processes → New Approval Process.
2. Object: **Enrollment**.
3. Process Name: *Discount Approval*.
4. Entry Criteria: `Discount_Percentage__c > 20`.
5. Next Approver: User in *Management Role*.
6. Final Approval Action: Field Update → Enrollment Status = “Approved”.
7. Activate Process.

The screenshot displays the Salesforce Setup interface for configuring an Approval Process. The left sidebar shows the navigation menu with 'Setup' selected. The main content area is titled 'Approval Processes' and shows the configuration for 'Enrollment: Discount Approval'.

Process Definition Detail

Field	Value	Field	Value
Process Name	Discount Approval	Active	✓
Unique Name	Discount_Approval	Next Automated Approver Determined By	Manager of Record Submitter
Description			
Entry Criteria	Enrollment: Discount Percentage GREATER THAN 0.20		
Record Editability	Administrator ONLY	Allow Submitters to Recall Approval Requests	<input type="checkbox"/>
Approval Assignment Email Template			
Initial Submitters	Contact Owner, Role: Admissions Team		
Created By	Nishant Dubey, 9/22/2025, 11:22 AM	Modified By	Nishant Dubey, 9/22/2025, 11:25 AM

Initial Submission Actions

Action Type	Description
Record Lock	Lock the record from being edited

Approval Steps

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions Edit	1	Manager Approval	Manager reviews discount requests over 20%.		Manager	Final Rejection

Final Approval Actions

Action Type	Description
Edit Record Lock	Lock the record from being edited

Final Rejection Actions

Action Type	Description

5. Flow Builder

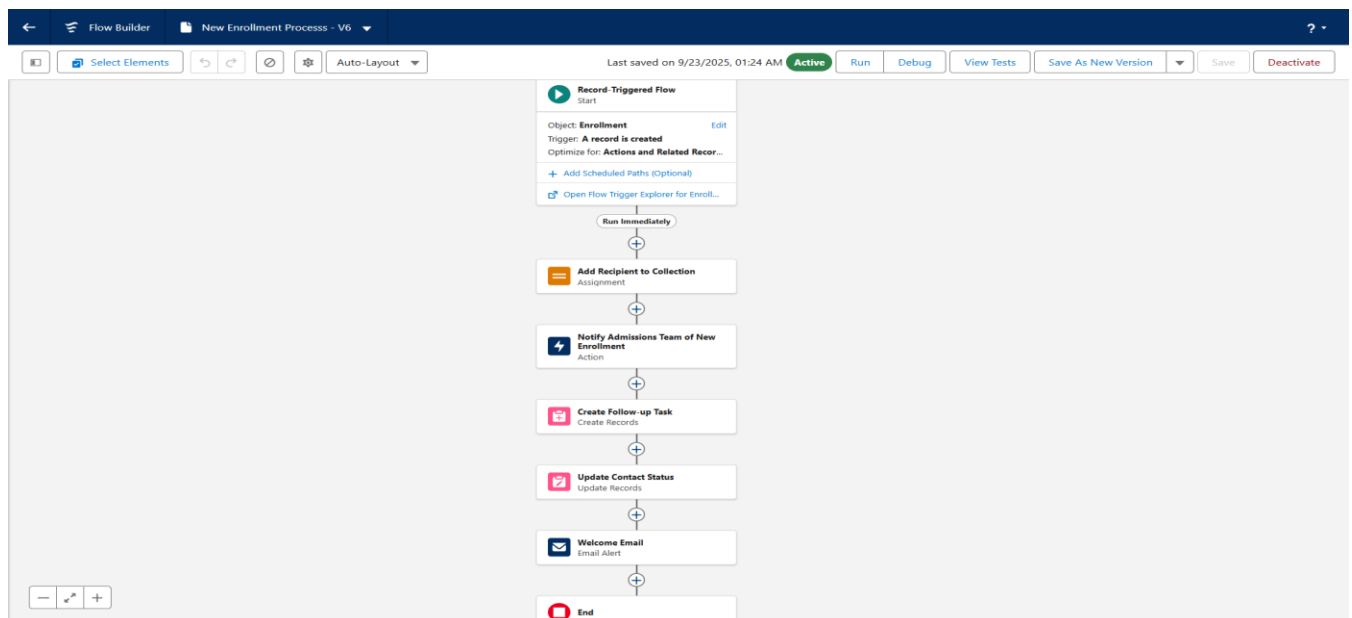
Flow Builder is the **primary automation tool** in Salesforce, replacing Workflow and Process Builder.

Use Case: Automate the student welcome process after enrollment.

Steps Implemented:

1. Setup → Flows → New Flow.
2. Flow Type: Record-Triggered Flow.
3. Object: **Enrollment**. Trigger: *Record Created*.
4. Optimize For: Actions & Related Records.
5. Added actions:
 - Send Welcome Email (Email Alert).
 - Create Admissions Follow-up Task.
 - Update Student Record fields.

6. Save → Activate



6. Email Alerts

Email Alerts send predefined email templates to specific recipients.

Use Case: Send a welcome email to newly enrolled students.

Steps Implemented:

1. Setup → Classic Email Templates → Create Template.
2. Setup → Email Alerts → New Alert.
 - Description: *Welcome Email to Student.*
 - Object: Enrollment.
 - Email Template: Select Welcome Template.
 - Recipient: Related Contact → Student.
3. Save.

The screenshot shows a web application interface for setting up email templates. On the left is a navigation sidebar with a search bar containing 'classic' and a menu with 'Email' expanded, showing 'Classic Email Templates' and 'Classic Letterheads'. The main content area is titled 'Classic Email Templates' and 'New Student Welcome Email'. It includes a 'View Sample Template' link and a note about the Description field. Below this is an 'Available Merge Fields' section with a table for selecting fields. The 'Email Template Edit' section contains fields for Folder, Availability, Name, Unique Name, Encoding, Description, Subject, and Email Body. The Email Body field contains a sample welcome message.

Search Setup

Setup Home Object Manager

classic

Email

Classic Email Templates

Classic Letterheads

Didn't find what you're looking for? Try using Global Search.

SETUP Classic Email Templates

New Student Welcome Email

Use merge fields to personalize your email content. You can add substitute text to any merge field. Substitute text displays only if the merge record does not contain data for that field. Enter substitute text after a comma in the merge field, for example, {{Contact.FirstName,Sir or Madam}}. When you save the template, the merge field will appear in the email body of the template with the following syntax: {{NullValue(Contact.FirstName,'Sir or Madam')}}. Click on the link below to see a sample email template.

[View Sample Template](#)

Note that the Description field is for internal use only. It will be listed as the title of any email activities you log when sending mass email.

Select Field Type	Select Field	Copy Merge Field Value
Contact Fields		

Copy and paste the merge field value into your template below.

Email Template Edit

Save Save & New Cancel

Email Template Information

Folder Unfiled Public Classic Email Templates

Available For Use ☒

Email Template Name New Student Welcome Email

Template Unique Name New_Student_Welcome_Email

Encoding Unicode (UTF-8)

Description

Subject Welcome to SN SmartLearn!

Email Body

Hello {{Enrollment.Student_r.FirstName}},

Welcome to SN SmartLearn! You are now enrolled in {{Enrollment.Course_r.Name}}. We're excited to have you.

The SN SmartLearn Team

7. Field Updates

Field Updates automatically change field values when triggered by automation.

Use Case: Update Student's Contact record when Enrollment is created.

Steps Implemented (via Flow):

1. Flow Builder → Add *Update Records* element.
2. Object: **Contact**.
3. Condition: Id = {!\$Record.Student__c}.
4. Field Value: Enrollment_Status__c = "Enrolled".
5. Save → Connect → Activate.

The screenshot displays the Salesforce Flow Builder interface. The top navigation bar shows 'Flow Builder' and 'New Enrollment Process - V6'. The flow canvas on the left contains a sequence of steps: 'Run Immediately', 'Add Recipient to Collection' (Assignment), 'Notify Admissions Team of New Enrollment' (Action), 'Create Follow-up Task' (Create Records), 'Update Contact Status' (Update Records, highlighted with a blue border), 'Welcome Email' (Email Alert), and 'End'. The right-hand pane is open to the configuration for the 'Update Records' element. It includes fields for 'Label' (Update Contact Status) and 'API Name' (Update_Contact_Status). The 'How to Find Records to Update and Set Their Values' section has four radio buttons, with 'Specify conditions to identify records, and set fields individually' selected. The 'Object' is set to 'Contact'. The 'Filter Contact Records' section shows a condition: 'Contact ID' equals 'Enrollment__c > Student > Email'. The 'Set Field Values for the Contact Records' section is currently empty.

8. Tasks

Tasks create actionable to-dos for users inside Salesforce.

Use Case: Assign follow-up task for Admissions Officer when a new application is submitted.

Steps Implemented (via Flow):

1. Flow Builder → Add *Create Records* element.
2. Object: **Task**.
3. Field Values:
 - Subject: *Follow up with new student*.
 - Whold = Student (\$Record.Student__c).
 - OwnerId = Record Owner (\$Record.OwnerId).
4. Save → Activate.

The screenshot displays the Salesforce Flow Builder interface for a flow named "New Enrollment Process - V6". The flow is currently in the "Run Immediately" state. The flow steps are as follows:

- Run Immediately
- Add Recipient to Collection (Assignment)
- Notify Admissions Team of New Enrollment (Action)
- Create Follow-up Task (Create Records) - This step is highlighted with a blue border.
- Update Contact Status (Update Records)
- Welcome Email (Email Alert)
- End

The right-hand pane shows the configuration for the "Create Records" step:

- Label:** Create Follow-up Task
- API Name:** Create_Follow_up_Task
- Description:** (Empty text area)
- How to set record field values:** Manually
- Create a Record of This Object:** Task
- Set Field Values for the Task:**
 - Field: Assigned To ID → Value: Running User > Id
 - Field: Subject → Value: Follow up with new student
 - Field: Name ID → Value: Triggering Enrollment_c > Student
- Manually assign variables (advanced):** (Unchecked)
- Check for Matching Records:** (Toggle switch)

9. Custom Notifications

Custom Notifications send alerts to users in Salesforce (bell icon & mobile).

Use Case: Notify Admissions Team Lead when a new student enrolls.

Steps Implemented:

1. Setup → Custom Notifications → Create Notification Type.
2. Flow Builder → Add Action → *Send Custom Notification*.
3. Configure:
 - Notification Type: *Enrollment Notification*.
 - Title: *New Student Enrollment*.
 - Body: *A new student has enrolled. Please review.*
 - Recipient: Admissions Team Lead.

Save → Activate.

The screenshot displays the Salesforce Flow Builder interface for a flow named 'New Enrollment Process - V6'. The flow is currently in 'Active' status and was last saved on 9/23/2025 at 01:24 AM. The flow diagram on the left shows a sequence of steps: 'Run Immediately' (Start), 'Add Recipient to Collection' (Assignment), 'Notify Admissions Team of New Enrollment' (Action, highlighted), 'Create Follow-up Task' (Create Records), 'Update Contact Status' (Update Records), 'Welcome Email' (Email Alert), and 'End'.

The right-hand pane shows the configuration for the 'Send Custom Notification' action. The configuration includes the following fields:

- Label:** Notify Admissions Team of New Enrollment
- API Name:** Notify_Admissions_Team_of_New_Enrollment
- Description:** (Empty text area)

Below these fields, a note states: 'Use values from earlier in the flow to set the inputs for the "Send Custom Notification" core action. To use its outputs later in the flow, store them in variables.'

The 'Set Input Values' section contains the following configurations:

- Custom Notification Type ID:** New Enrollment Notification
- Notification Body:** {!\$Record.Student__r.FirstName} {!\$Record.Stude Q
- Notification Title:** New Student Enrolled
- Recipient IDs:** A3 recipientIdCollection X

Custom Notifications | Salesforce

orgfarm-5e644081b6-dev-ed.develop.lightning.force.com/lightning/setup/CustomNotifications/home

Search Setup

Setup

Home

Object Manager

Custom Notifications

Notification Builder

Custom Notifications

Didn't find what you're looking for?
Try using Global Search.

SETUP

Custom Notifications

When you create and use custom notifications, the title and body of the custom push notification may be saved to and processed by Google, Microsoft and/or Apple. Salesforce is not responsible for the privacy and security practices of third-party systems or applications like Google Cloud Messaging or Apple Push Notification Service.

Custom Notification Types

New

Send custom notifications using [Flows](#) or [Process Builder](#)

NOTIFICATION NAME	API NAME	NAMESPACE	DESKTOP	MOBILE	
enablement_coaching_feedback_ready	enablement_coaching_feedback_ready		✓		▼
New Enrollment Notification	New_Enrollment_Notification		✓	✓	▼
SN Notification	SN_Notification		✓	✓	▼