

NAAN MUDHALVAN

SALESFORCE PROJECT REPORT

Project Title: Creation of an Application for School Management

NAME	NMID
MEENA KUMAARI M M	F74D267361D1FF2BE974F056620EEA5B
REMINA R	FC8350AF87EB2C5827F4F1BCEF36F061
JENISHA J	E66479E1576E98585B4414A80C555ACF
BEETRISHYA M	499CE409D834944BDF544E661C434ACE
MERIN STEPHY E H	FECD6194A0F1ADCAC5A407E21CD7B467

1.1 Introduction:

Implementing School Management Application represents an innovative project tailored to streamline educational institution operations. This comprehensive system is designed to improve administrative efficiency, and provide real-time access to critical school-related information.

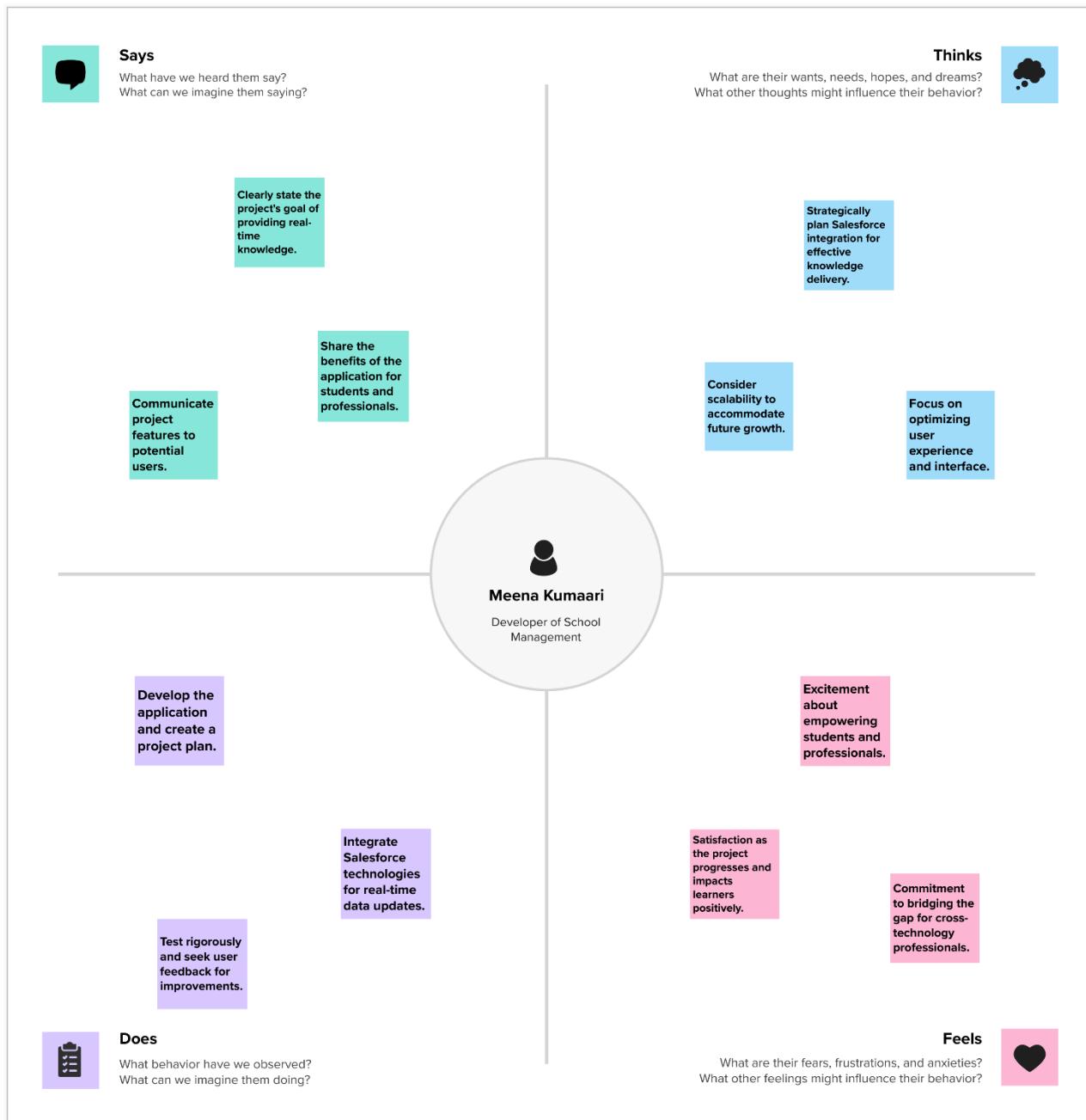
1.2 Project Goals:

1. Modernize school management.
2. Improve efficiency and data accuracy.
3. Enhance the educational experience.
4. Provide real-time access to data.

1.3 Problem Definitions and Design Thinking:

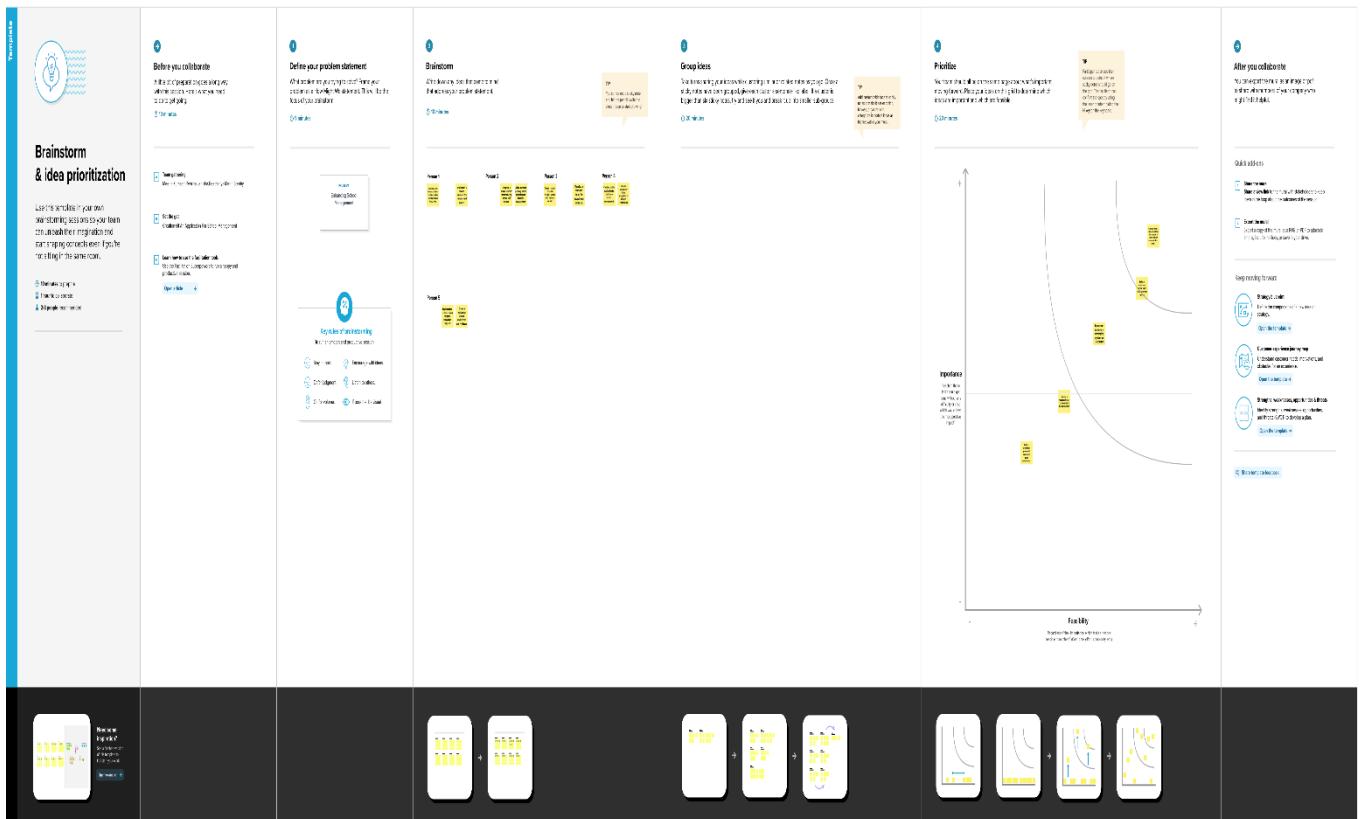
Empathy Map:

Use the empathy map to outline user emotions, thoughts, and pain points when navigating the school management application, helping design a more user-centered system.



Ideation & Brainstorming:

Generating creative solutions to address the identified challenges, using brainstorming and collaboration.



1.4 Scope:

- 1.Creation of a comprehensive CRM system in Salesforce.
- 2.Automation of data entry, processing, and control
- 3.User-friendly dashboard for real-time data access.

1.5 Technical Requirements:

- 1.Technology Stack: Salesforce CRM, database management systems.
- 2.Hardware: Servers and storage for Salesforce instance.
- 3.Software: Salesforce licenses, custom development tools.

1.6 Functional Requirements:

Role-Based Access: Role-based access control for administrators, teachers, students, parents

Data Entry: Implement user-friendly data entry forms for candidate

Reporting: Create customizable reports and dashboards for data analysis.

Automation: Implement workflows and triggers for automated processes.

Customization: Allow for customization of fields and layouts to accommodate various needs.

Milestone 1: [Creation of developer account]

Creating developer org:

Go to developer.salesforce.com

Click on sign up.

On the sign-up form, enter the following details

- a. First name & Last name
- b. Email
- c. Role - Developer
- d. Company - College Name
- e. Country - India
- f. Postal Code - pin code
- g. Username- should be a combination of your name and company

This need not be an actual email id, can give anything in the format:

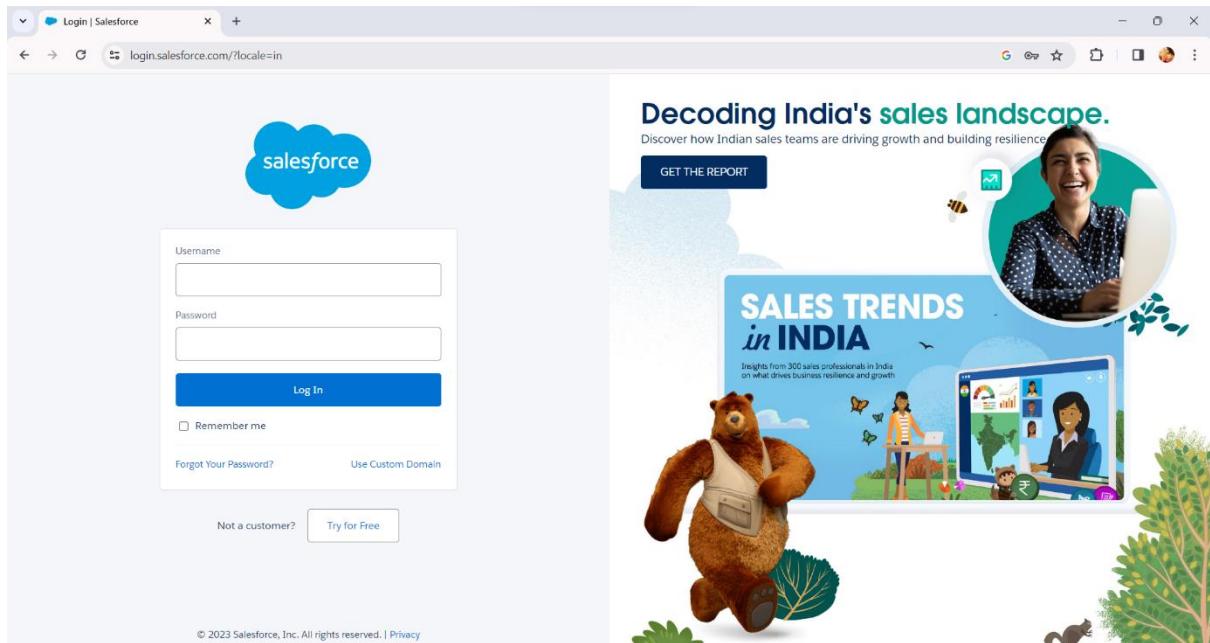
username@organization.com

Login To Your Salesforce Account

Go to salesforce.com and click on login.

Enter the username and password

After login this is the home page



The screenshot shows the Salesforce Setup Home page.

Left Sidebar:

- Setup Home
- Service Setup Assistant
- Commerce Setup Center
- Multi-Factor Authentication Assistant
- Hyperforce Assistant
- Release Updates
- Lightning Experience Transition Assistant
- Salesforce Mobile App
- Lightning Usage
- Optimizer
- ADMINISTRATION

 - Users
 - Data
 - Email

- PLATFORM TOOLS

 - Subscription Management
 - Apps
 - Feature Settings
 - Slack

Top Bar:

- Search bar: Search Setup
- Create button

Home Section:

- Get Started with Einstein Bots**: Launch an AI-powered bot to automate your digital connections. Get Started button.
- Mobile Publisher**: Use the Mobile Publisher to create your own branded mobile app. Learn More button.
- Real-time Collaborative Docs**: Transform productivity with collaborative docs, spreadsheets, and slides inside Salesforce. Get Started button.

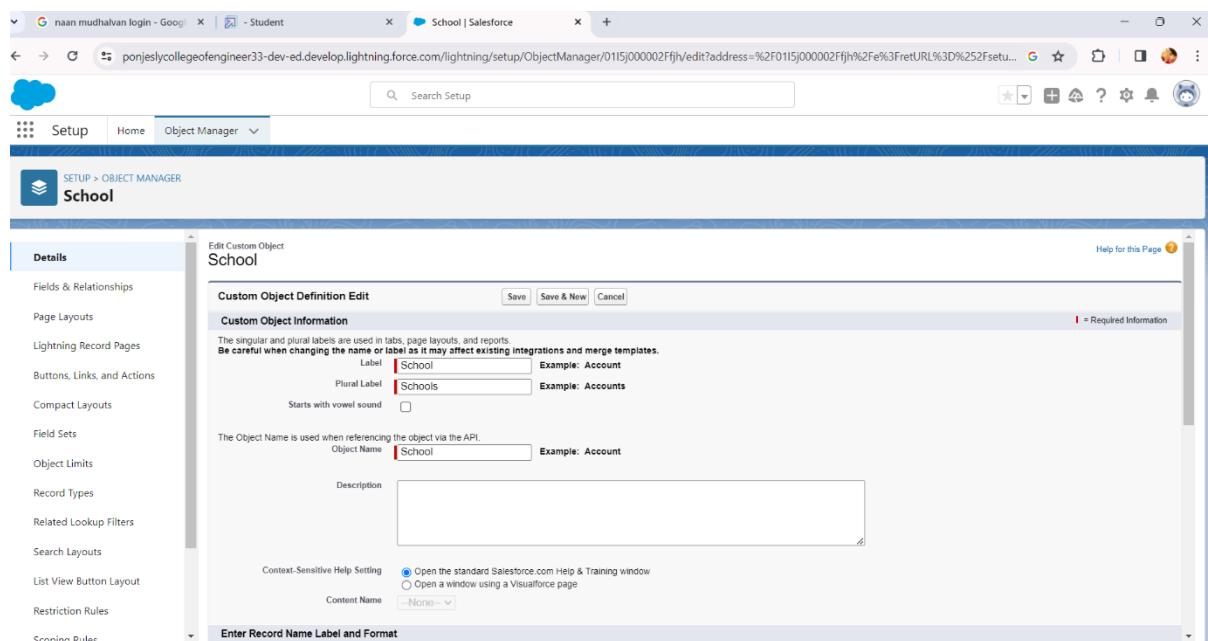
Most Recently Used:

NAME	TYPE	OBJECT
teacher_permission	Permission Set	
principal_permission	Permission Set	
Internalmarks	Apex Trigger	

Milestone 2: [Object Creation]

Creation Of School Object:

1. Click on the gear icon and then select Setup.
2. Click on the object manager tab just beside the home tab.
3. After the above steps, have a look on the extreme right you will find a Create Dropdown click on that and select Custom Object.
4. On the Custom Object Definition page, create the object as follows:
 5. Label: School
 6. Plural Label: Schools
 7. Record Name: School Name
 8. Check the Allow Reports checkbox
 9. Check the Allow Search checkbox
10. Click Save. (Use same procedure for creating student, parent object)



Create Student Object:

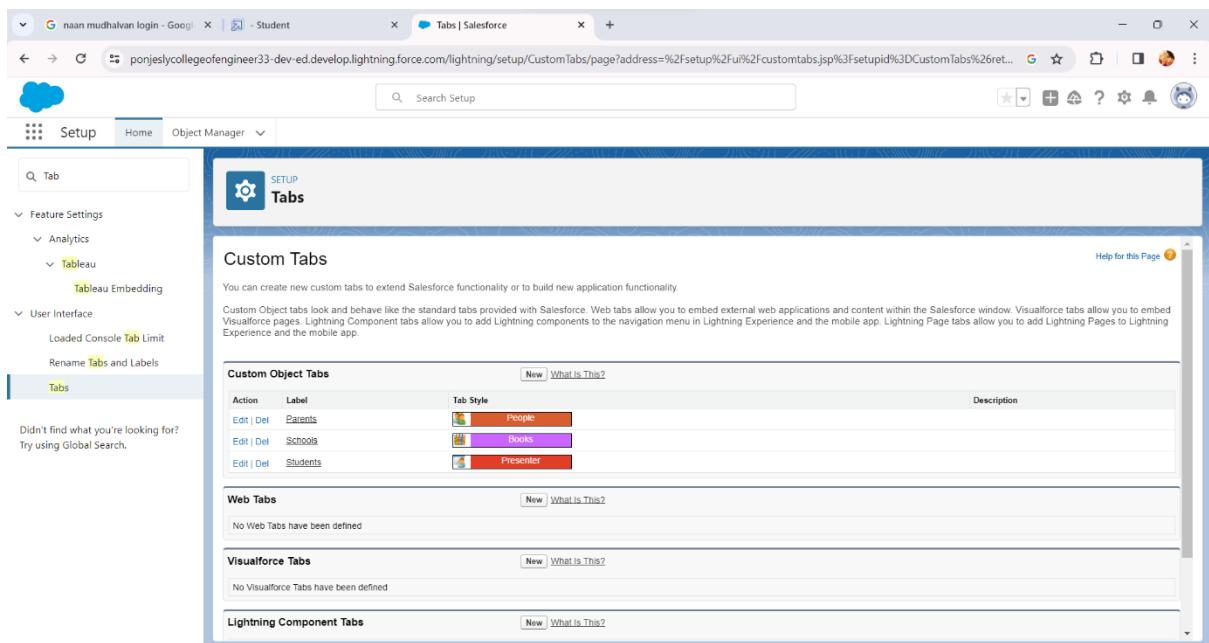
The screenshot shows the Salesforce Setup interface with the title 'Student | Salesforce'. The main content is the 'Edit Custom Object Student' page. On the left, there's a sidebar with various settings like Fields & Relationships, Page Layouts, and Record Types. The main area has tabs for 'Custom Object Definition Edit' and 'Custom Object Information'. Under 'Custom Object Information', the 'Label' is set to 'Student' and the 'Plural Label' is set to 'Students'. There's also a note about changing the name or label affecting existing integrations and merge templates. Below that, the 'Object Name' is set to 'Student'. A 'Description' field is present, though empty. At the bottom, there are 'Context-Sensitive Help Setting' options (radio buttons for standard help or Visualforce page) and a 'Content Name' dropdown set to 'None'. A 'Help for this Page' link is at the top right.

Create Parent Object:

This screenshot shows the 'Edit Custom Object Parent' page in the Salesforce Setup interface. The layout is identical to the 'Student' object page, with a sidebar on the left and configuration fields on the right. The 'Label' is set to 'Parent' and the 'Plural Label' is set to 'Parents'. The 'Object Name' is set to 'Parent'. The rest of the configuration fields (Description, Context-Sensitive Help Setting, Content Name) are identical to the 'Student' object page.

Milestone 3: [Tab Creation]

1. Now create a custom tab. Click the Home tab, enter Tabs in Quick Find and select Tabs. Under Custom Object Tabs, click New.
2. For Object, select Students.
3. For Tab Style, select any icon.
4. Leave all defaults as is. Click Next, Next, and Save.

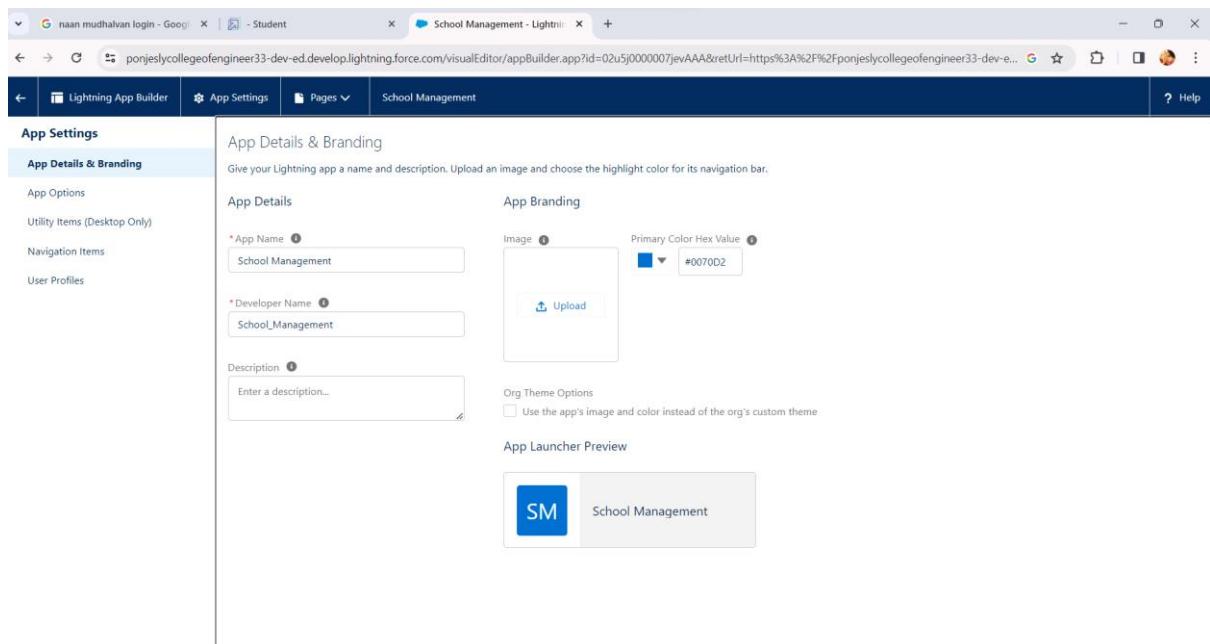


Milestone 4: [Create App]

Create The School Management App

1. From Setup, enter App Manager in the Quick Find and select App Manager.
2. Click New Lightning App.
3. Enter School Management as the App Name, then click Next
4. Under App Options, leave the default selections and click Next.

5. Under Utility Items, leave as is and click Next.
6. From Available Items, select Schools, Students, Parents, Reports, and Dashboards and move them to Selected Items. Click Next.
7. From Available Profiles, select System Administrator and move it to Selected Profiles. Click Save & Finish.



Milestone 5:[Fields & Relationships]

Creation of fields for the school objects:

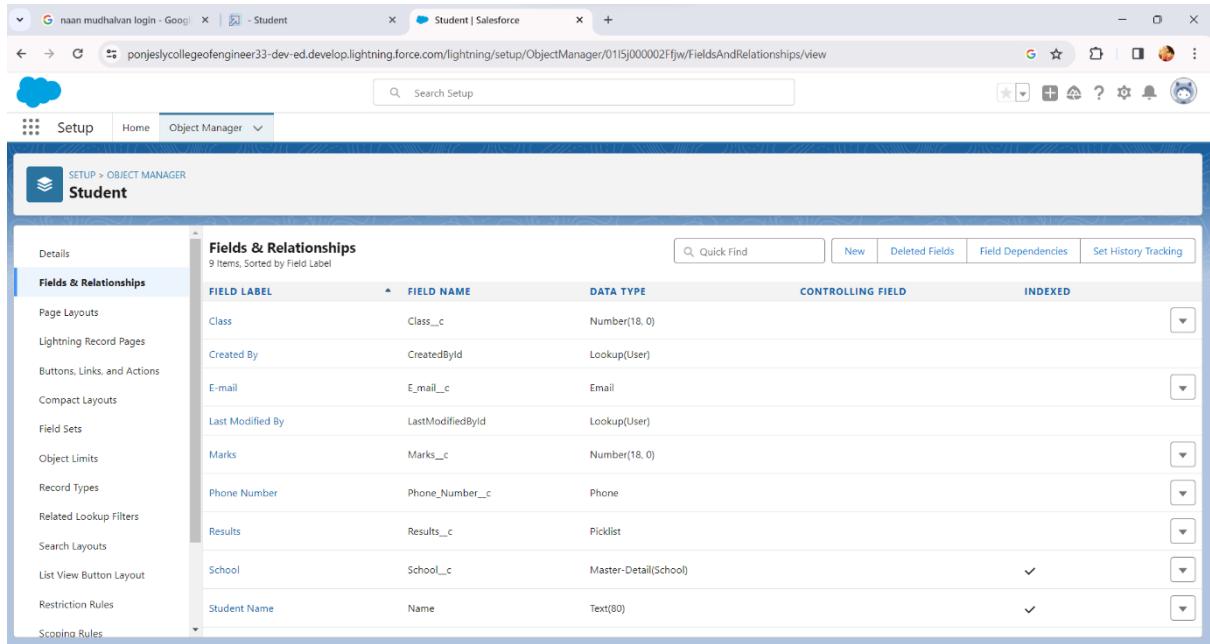
1. Click the gear icon and select Setup. This launches Setup in a new tab.
2. Click the Object Manager tab next to Home.
3. Select School.

4. Select Fields & Relationships from the left navigation
5. Click New
6. Select the Text Area as the Data Type, click Next.
7. For Field Label, enter Address.
8. Click Next, Next, then Save & New.
9. Follow steps 1 to 3 and create two more Text type field - District & State.
10. Create URL type field & give Schoolwebsite as the field label.

The screenshot shows the Salesforce Object Manager interface for the 'School' object. The left sidebar shows navigation links for Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, and Restriction Rules. The main content area is titled 'Fields & Relationships' and displays a table of 10 items, sorted by Field Label. The table columns are FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The listed fields are:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address__c	Text Area(255)		
Created By	CreatedById	Lookup(User)		
District	District__c	Text Area(255)		
Highest Marks	Highest_Marks__c	Roll-Up Summary (MAX Student)		
Last Modified By	LastModifiedById	Lookup(User)		
Number of students	Number_of_students__c	Roll-Up Summary (COUNT Student)		
Owner	OwnerId	Lookup(User/Group)	✓	
Phone Number	Phone_Number__c	Phone		
School Name	Name	Text(80)	✓	

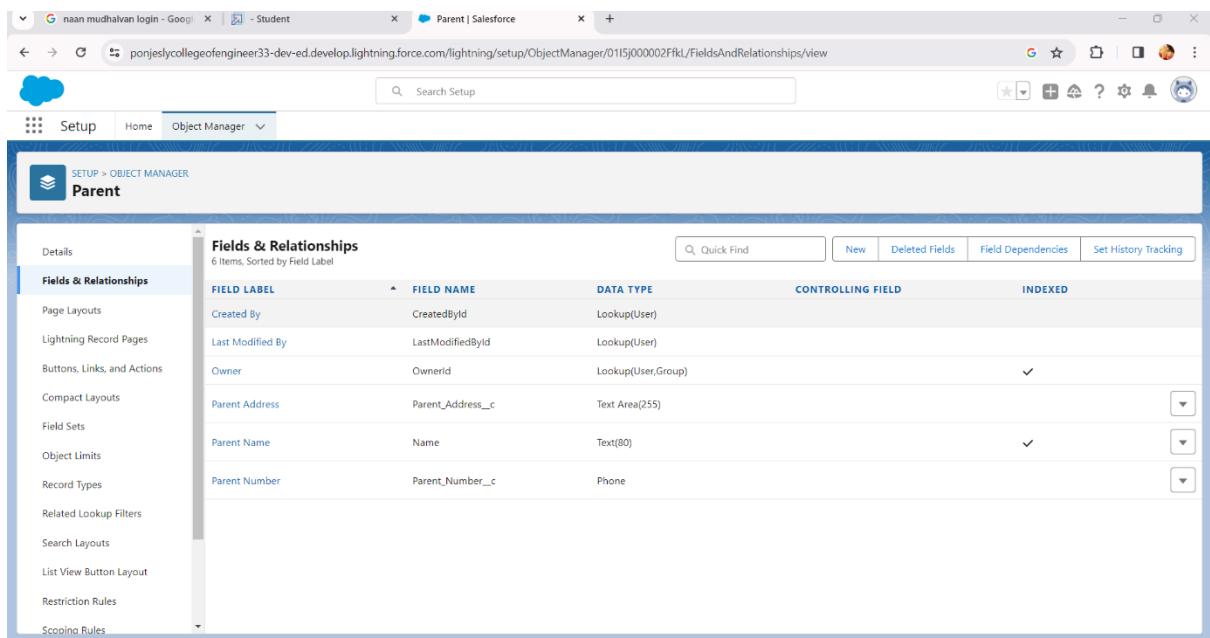
Creation of fields for the student objects:



The screenshot shows the Salesforce Object Manager interface for the 'Student' object. The left sidebar lists various setup options like Page Layouts, Lightning Record Pages, and Field Sets. The main area is titled 'Fields & Relationships' and displays a table of 9 items, sorted by Field Label. The table columns are FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Class	Class__c	Number(18, 0)		
Created By	CreatedById	Lookup(User)		
E-mail	E_mail__c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Marks	Marks__c	Number(18, 0)		
Phone Number	Phone_Number__c	Phone		
Results	Results__c	Picklist		
School	School__c	Master-Detail(School)		
Student Name	Name	Text(80)		

Creation of fields for the parent objects:



The screenshot shows the Salesforce Object Manager interface for the 'Parent' object. The left sidebar lists various setup options. The main area is titled 'Fields & Relationships' and displays a table of 6 items, sorted by Field Label. The table columns are FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		
Parent Address	Parent_Address__c	Text Area(255)		
Parent Name	Name	Text(80)		
Parent Number	Parent_Number__c	Phone		

Milestone 6:[Profile]

Creation On Profile:

1. From Setup enter Profiles in the Quick Find box
2. Select Profiles.
3. Click new, From the list of profiles, find Standard User (profile to clone). For Profile Name, enter School Profile
4. Click Save.
5. While still on the school profile page, then click Edit.
6. Scroll down to Custom Object Permissions and Give view all access permissions.

The screenshot shows the Salesforce Profiles setup interface. The top navigation bar includes tabs for Setup, Home, and Object Manager. A search bar is at the top right. The main content area has a sidebar with a search field and sections for Users and Profiles. The 'Profiles' section is selected. A modal window titled 'Profile Edit' is open for the 'School Profile'. The modal contains fields for Name (School Profile), User License (Salesforce), and Description. Below these are sections for 'Custom App Settings' and 'Custom Object Permissions'. Under 'Custom Object Permissions', there are two tables: one for 'Sales (standard__Sales)' and another for 'Service (standard__Service)'. In the 'Sales' table, the 'Sales Console' permission is set to 'Visible' and 'Default'. In the 'Service' table, the 'School Management' permission is set to 'Visible' and 'Default'. Other standard permissions like 'Salesforce Chatter' and 'Salesforce Scheduler Setup' are also listed. At the bottom of the modal are 'Save', 'Save & New', and 'Cancel' buttons.

Milestone 7:[Role and role hierarchy]

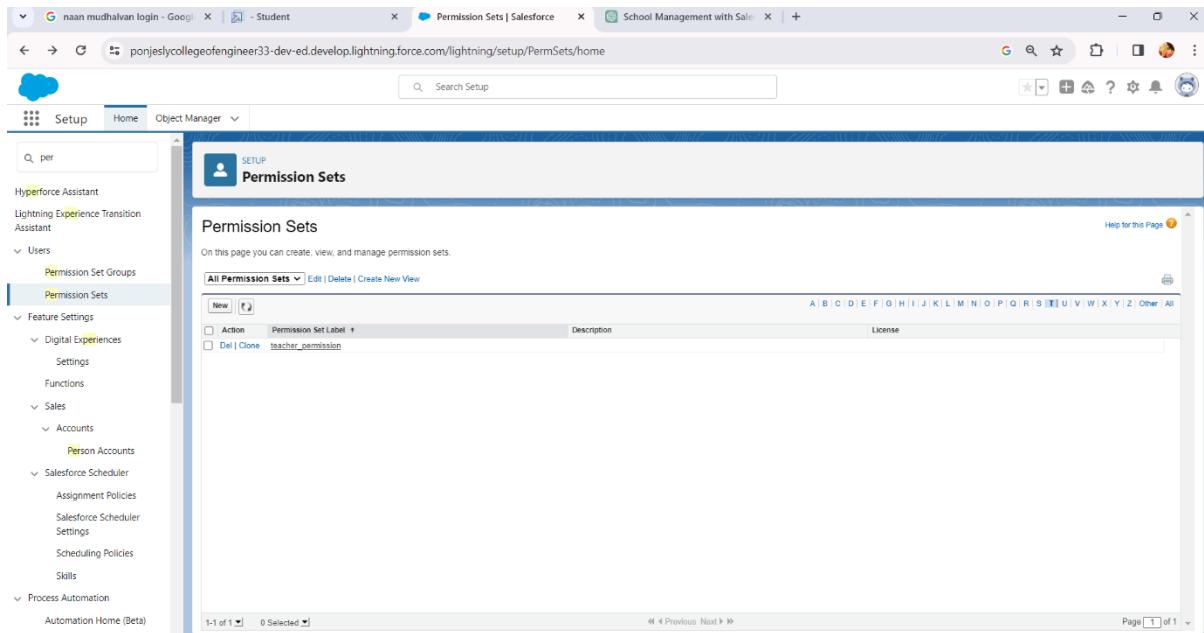
Roles and role hierarchy are foundational components in Salesforce that help you organize and manage users, their access levels, and data visibility. These concepts are essential in a school management application to ensure that the right individuals have appropriate access to the data and features relevant to their roles, and they are vital for creating a secure and effective system.

Object	Default Sharing	Status
Seller	Private	✓
Service Appointment	Public Read/Write	✓
Service Contract	Private	✓
Service Resource	Public Read/Write	✓
Service Territory	Public Read/Write	✓
Shift	Private	✓
Streaming Channel	Public Read/Write	✓
Tableau Host Mapping	Private	✓
User Provisioning Request	Private	✓
Waitlist	Private	✓
Web Cart Document	Private	✓
Work Order	Private	✓
Work Plan	Private	✓
Work Plan Template	Private	✓
Work Step Template	Private	✓
Work Type	Private	✓
Work Type Group	Public Read/Write	✓
Parent	Public Read/Write	✓
School	Public Read/Write	✓
Student	Controlled by Parent	Controlled by Parent

Create A Permission Set:

Permission set 1

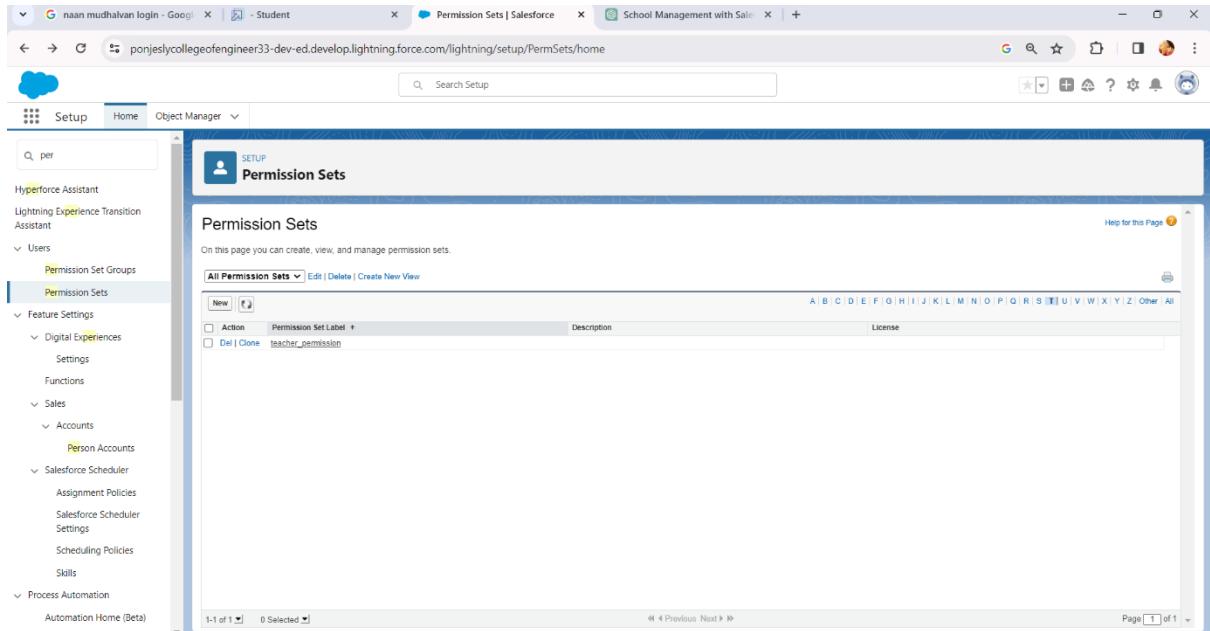
1. From Setup, enter Permission Sets in the Quick Find box, then select Permission Sets.
2. Click New.
3. Give the name of the Permission set name as teacher permission.
4. Under the object settings give the view create and edit permissions to all 3 custom objects (By click open the object)
5. Click on manage assignment
6. Click on add assignment.
7. Click on Teacher (user), Next, Assign.



Permission set 1

1. From Setup, enter Permission Sets in the Quick Find box, then select Permission Sets.
2. Click New.
3. Give the name of the Permission set name as Principal permission and then under the object settings give all

permissions for the all 3 custom objects and assign them to the Principal user.



The screenshot shows the Salesforce Setup interface with the 'Permission Sets' page open. The left sidebar shows various setup categories like 'Users', 'Feature Settings', 'Sales', and 'Process Automation'. The main area displays a table for 'Permission Sets' with one entry:

Action	Permission Set Label	Description	License
<input type="checkbox"/>	teacher_permission		

Milestone 8: [Users]

Creating A User:

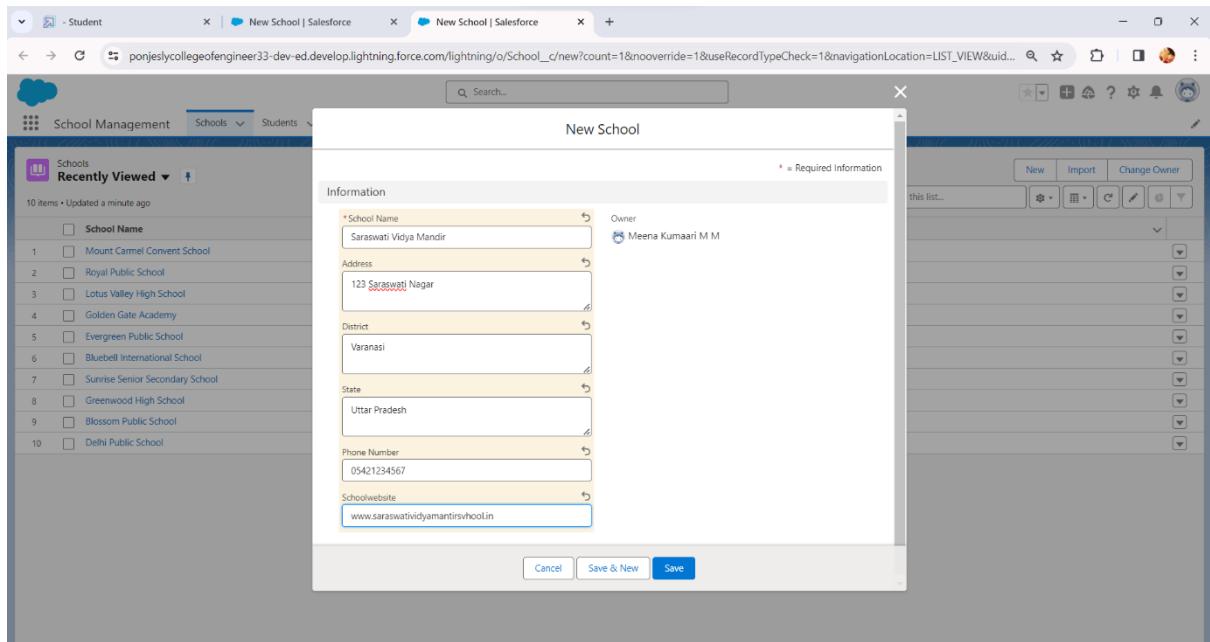
1. From Setup, in the Quick Find box, enter Users.
2. Select Users.
3. Click New User.
4. Enter the user's name Parents and (Your) email address and a unique username in the form of an email address. By default, the username is the same as the email address.
5. Select a User License as salesforce.
6. Select a profile as a school profile.
7. Check Generate new password and notify the user immediately to have the user's login name and a temporary password emailed to your email.
8. Similarly follow the above steps and create 2 users as Teachers and principals

Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/>	Chatter Expert	Chatter	chatty.00d500000chmbkeav.lmnchbckah@chatter.salesforce.com		<input checked="" type="checkbox"/>	Chatter Free User
<input type="checkbox"/>	M.M. Meena Kumar	MM.M	meena@0792t.com		<input checked="" type="checkbox"/>	System Administrator
<input type="checkbox"/>	Parents	pare	schoolparents@gmail.com		<input type="checkbox"/>	School Profile
<input type="checkbox"/>	Principals	prin	schoolsprincipals@gmail.com		<input checked="" type="checkbox"/>	School Profile
<input type="checkbox"/>	Teachers	teac	schoolteachers@gmail.com		<input type="checkbox"/>	School Profile
<input type="checkbox"/>	User_Integration	integ	integration@00d500000chmbkeav.com		<input checked="" type="checkbox"/>	Analytics Cloud Integration User
<input type="checkbox"/>	User_Security	sec	inightssecurity@00d500000chmbkeav.com		<input checked="" type="checkbox"/>	Analytics Cloud Security User

Milestone 9: [User Adoption]

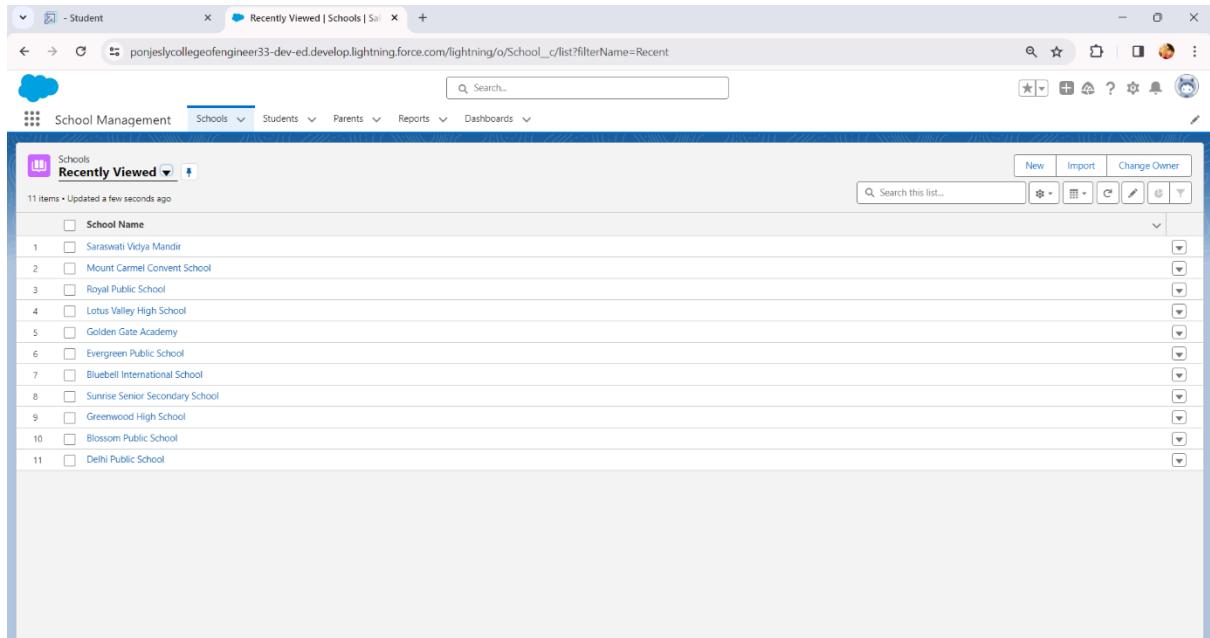
Create Record (School):

1. Click on App Launcher on left side of screen.
2. Search School Management App & click on it.
3. Click on Schools tab.
4. Click new button
5. Fill all School record details.
6. Click on Save Button



View Record (School):

1. Click on App Launcher on left side of screen.
2. Search School Management App & click on it.
3. Click on Schools Tab.
4. Click on any record name. you can see the details of the school.



Delete Record (School):

1. Click on App Launcher on left side of screen.
2. Search School Management App & click on it.
3. Click on Schools Tab.
4. Click on Arrow at right hand side on that Particular record.
5. Click delete and delete again.

School Name

1	Saraswati Vidya Mandir
2	Mount Carmel Convent School
3	Royal Public School
4	Lotus Valley High School
5	Golden Gate Academy
6	Evergreen Public School
7	Bluetell International School
8	Sunrise Senior Secondary School
9	Greenwood High School
10	Blossom Public School
11	Delhi Public School

Milestone 10: [Reports]

Create Report:

In School Management App click Reports tab.

1. Click New Report.
2. Select the report type as School with students and parents for the report.
3. Click start report.
4. Customize your report, then save and run
5. Give report name – Schools with Students Report, Click Save

Schools with Students

Address	Student: Student Name	School: School Name	Phone Number	District
222 Gateway Street (2)	Remina Beetrishya	Golden Gate Academy Golden Gate Academy	0807654321 0807654321	222 Gateway Street
Subtotal				
333 Carmel Street (3)	Priya dharshini Varsha Shruthi	Mount Carmel Convent School Mount Carmel Convent School Mount Carmel Convent School	01728765432 01728765432 01728765432	Chandigarh
Subtotal				
456 Pine Road (2)	Meena Kumaari	Evergreen Public School	0334567890	Kolkata

View Report:

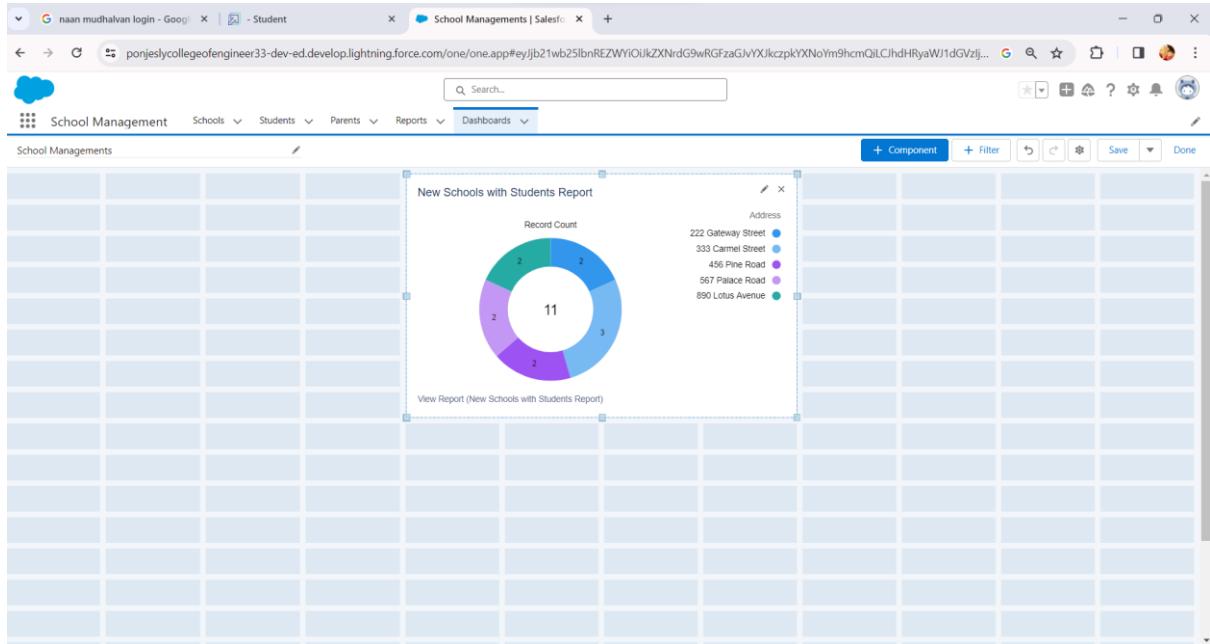
1. Click on App Launcher on left side of screen.
2. Search School Management App & click on it.
3. Click on Reports Tab.
4. Click on School with Students report and see records

Schools with Students Reports

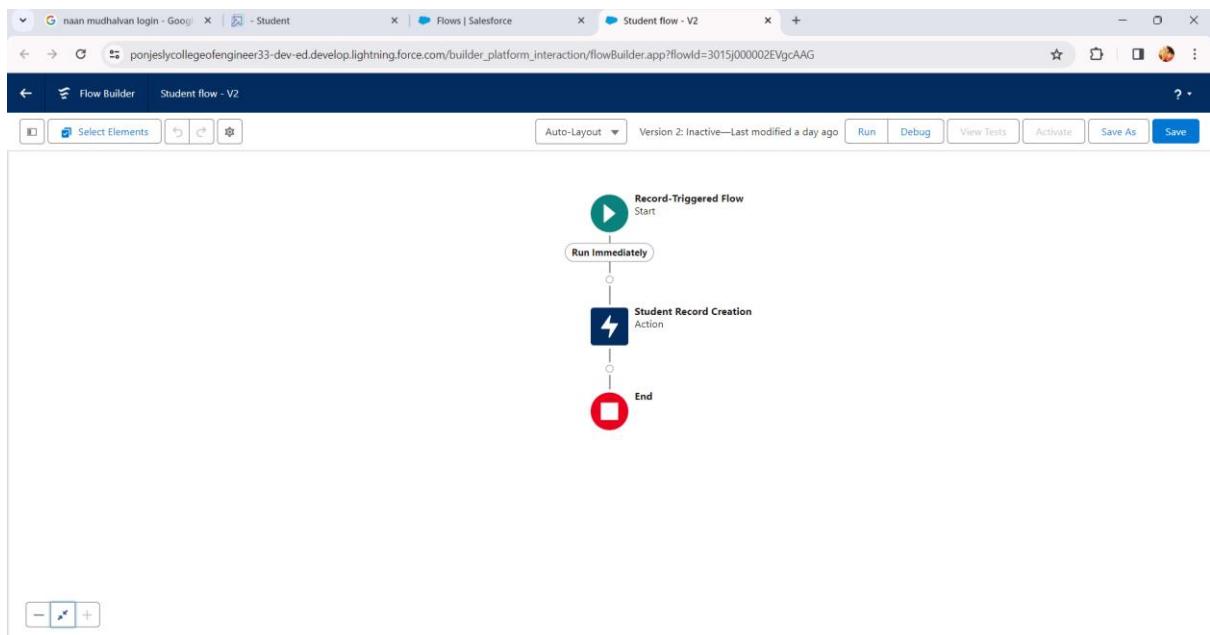
Total Records: 11

Address	Student: Student Name	School: School Name	Phone Number	District
222 Gateway Street (2)	Remina Beetrishya	Golden Gate Academy Golden Gate Academy	0807654321 0807654321	222 Gateway Street
Subtotal				
333 Carmel Street (3)	Priya dharshini Varsha Shruthi	Mount Carmel Convent School Mount Carmel Convent School Mount Carmel Convent School	01728765432 01728765432 01728765432	Chandigarh
Subtotal				
456 Pine Road (2)	Meena Kumaari Jenisha	Evergreen Public School Evergreen Public School	0334567890 0334567890	Kolkata
Subtotal				
567 Palace Road (2)	Aenorius Jebisha	Royal Public School Royal Public School	0402345678 0402345678	Hyderabad
Subtotal				
890 Lotus Avenue (2)	Merin Stephly Neha	Lotus Valley High School Lotus Valley High School	01209876543 01209876543	Noida
Subtotal				

Milestone 11: [Dashboards]



Milestone 12: [Flows]



Milestone 13: [Apex Classes and Triggers]

The screenshot shows the Salesforce Setup interface for Apex Classes. The left sidebar includes sections for Email, Custom Code (with Apex Classes selected), Development, Environments, and Jobs. The main content area is titled "Apex Classes" and displays a table of classes. A green box at the top right indicates "Percent of Apex Used: 0.01%" and "You are currently using 461 characters of Apex Code (excluding comments and @isTest annotated classes) in your organization, out of an allowed limit of 6,000,000 characters. Note that the amount in use includes both Apex Classes and Triggers defined in your organization." Below this, there's a "Dynamic Apex Classes" section with a note about extending programming reach. A search bar at the bottom right says "Show hidden icons".

This screenshot shows the "Apex Class Detail" page for the class "schoolHandler". The top navigation bar has "Apex Class schoolHandler" and tabs for "Edit", "Delete", "Download", "Security", and "Show Dependencies". The "Status" is Active and "Code Coverage" is 0% (0/4). The "Last Modified By" field shows Meena Kumari M.M. The "Class Body" tab is selected, displaying the following Apex code:

```
1 public class schoolHandler {  
2     public static void beforeDelete(List<School__c> oldList){  
3         for(School__c s : oldList){  
4             if(s.SchoolWebsite__c == null){  
5                 saddError('you cannot delete the record');  
6             }  
7         }  
8     }  
9 }  
10 }  
11 }  
12 }  
13 }  
14 }  
15 }  
16 }
```

Below the code, there are tabs for "Class Summary", "Version Settings", and "Trace Flags". At the bottom are "Edit", "Delete", "Download", "Security", and "Show Dependencies" buttons.

The screenshot shows the Salesforce Setup interface with the URL <https://ponjeslycollegeofengineer33-dev-ed.develop.lightning.force.com/lightning/setup/ApexTriggers/home>. The left sidebar is expanded, showing sections like Email, Custom Code, Development, Environments, and Jobs. Under Custom Code, 'Apex Triggers' is selected. The main content area is titled 'Apex Triggers' and contains a message stating 'This page allows you to view and modify all the triggers in your organization. To create a new trigger, navigate to the appropriate sObject triggers page.' Below this is a green box with a checkmark and the text 'Percent of Apex Used: 0.01% You are currently using 461 characters of Apex Code (excluding comments and @isTest annotated classes) in your organization, out of an allowed limit of 6,000,000 characters. Note that the amount in use includes both Apex Classes and Triggers defined in your organization.' A button 'Compile all triggers' is available. The main table lists one trigger: Internalmarks, which is a School object type trigger created by Meena Kumari M.M. on 28/10/2023 at 11:01 am. The table includes columns for Action, Name, Namespace Prefix, sObject Type, API Version, Status, Size Without Comments, Last Modified By, and Has Trace Flags.

The screenshot shows the Apex Trigger detail page for 'Internalmarks'. The URL is <https://ponjeslycollegeofengineer33-dev-ed.develop.lightning.force.com/lightning/setup/ApexTriggers/page?address=%2F01q5j000001tdmz>. The left sidebar is identical to the previous screenshot. The main content area is titled 'Apex Trigger Detail' for 'Internalmarks'. It shows basic details: Name (Internalmarks), sObject Type (School), Status (Active), and Last Modified By (Meena Kumari M.M. on 28/10/2023 at 11:01 am). Below this is a code editor window containing the Apex trigger code:

```
1 trigger Internalmarks on School__c (before insert, after delete) {  
2     if(trigger.isDelete){  
3         if(trigger.isBefore){  
4             schoolHandler.beforeDelete(trigger.old);  
5         }  
6     }  
7 }  
8  
9 }  
10  
11 }  
12  
13}
```