

# **A CRM Application to Manage the Services offered by an Institution**

By

**Vanapalli Suryatejaswini**

[21131a05r4@gvpce.ac.in](mailto:21131a05r4@gvpce.ac.in)

# Abstract

The **EduConsultPro** Institute is one of these leading educational institutes that annually presents a greater quantity of various courses and programs for an increasing quantity of prospective students. The service standards need effective management in admissions management, student inquiries, and expert consulting to keep them in a high position. This project will entail the design and implementation of a comprehensive CRM application using Salesforce to support this process better and improve the experience of students and admissions staff alike. The solution will consist of features in admission application management, consulting services management, and immigration case management, among others. ***Admission Application Management*** will enable prospects to make applications online and for the admission staff to trace and scrutinize the applications through automated notifications. ***Consulting Services Management*** will help students create their request for expert advice, while consultants can schedule and manage their appointments with much ease. The Immigration Case Management will provide information on immigration- related questions and cases. Some features of this system include automated workflows, email notifications, comprehensive dashboards, and collaboration tools. **EduConsultPro** Institute has now revamped pages for their prospective students, which are smooth, transparent, and easily navigated, all made possible through the use of Salesforce CRM. The sole concept, however, is developing operational efficiencies with an increased throughput in the admission process through consulting services. Now, this project aligns not just with things up front but provides a nice platform for scalable and future-fit growth.

# INDEX PAGE

Topic	Page no
1. Title	1
2. Abstract	2
3. Introduction	4
4. Create Objects from Spreadsheet	5
5. Create a ScreenFlow for Student Admission Application process	8
6. Create Users	10
7. Create an Approval Process for Property Object	11
8. Create a Record Triggered Flow	13
9. Create a ScreenFlow for Existing Student to Book an Appointment	14
10. Create a ScreenFlow to Combine all the flows at one place	16
11. Create a lightning app page	18
12. Execution Screenshots	19

# INTRODUCTION

EduConsultPro Institute, a prestigious educational establishment, is committed to improving the management of its admission processes, consulting services, and immigration case handling. As the institute experiences a rise in student applications and service requests, it recognizes the need for a more efficient and integrated solution to address these challenges.

To enhance operational efficiency and provide a seamless experience for both prospective students and admissions staff, EduConsultPro Institute is turning to Salesforce CRM. This initiative will streamline various processes, including the handling of admission applications, consulting requests, and immigration cases, all within a single, cohesive CRM platform.

## **The project will involve:**

- **Developing a user-friendly admission application system** that allows prospective students to apply online, with automated notifications and detailed reporting for admissions staff.
- **Creating an efficient approval workflow** for consulting requests, including automated email alerts and a streamlined submission process.
- **Managing consulting services** by enabling students to request consultations, schedule appointments, and track service statuses within Salesforce.
- **Handling immigration cases** by allowing students to initiate cases through multiple channels, manage documents, and track case progress with integrated tools.

Include the likes of setup of Salesforce objects, development of custom flows, designing for unified app pages, and more in lightning. These changes should be targeted at bringing a clearer and more smooth experience for the concerned users who are going through the processes of admission and consultation.

# Task1

## Create Objects from Spreadsheet

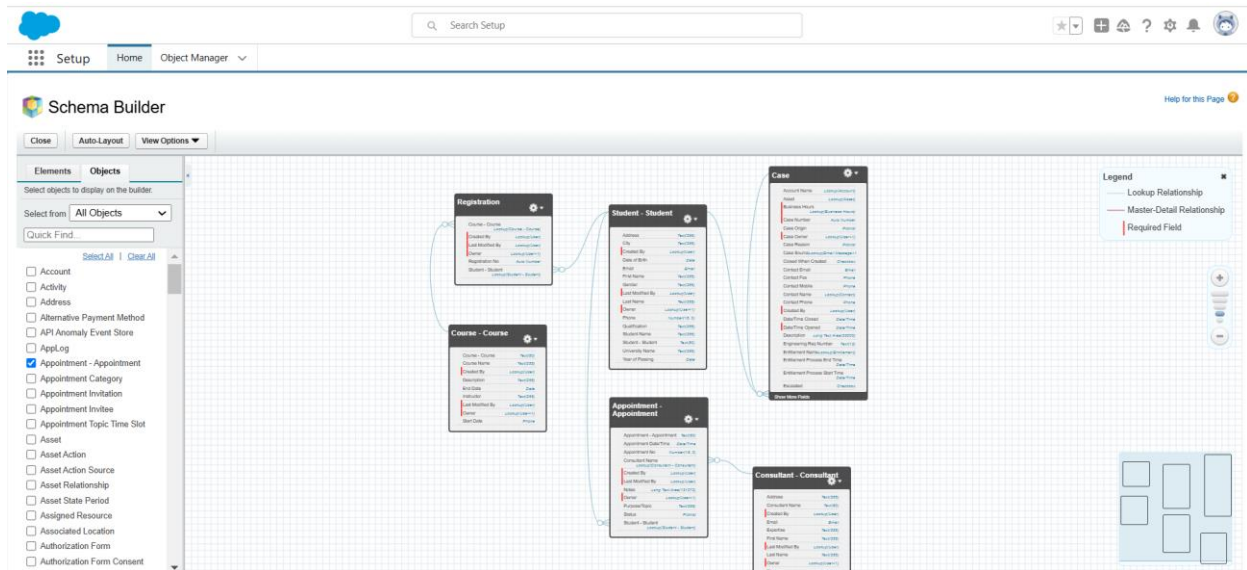
The outcome is enhanced services management at EduConsultPro Institute for which Salesforce functionality is leveraged to enable creation of objects directly from the spreadsheets. This will make it easier to integrate existing data in Salesforce and effectively manage the same for several institutional requirements.

Many objects were done using the spreadsheets given by the institution to solve the assignment. Course, Consultant, and Student, and Appointment are the objects designed, one representing a critical working of an institution. The lookup relationships between the said objects will ensure the design integrity and smooth navigation of data. At this point, lookup relationships were created between Appointment and Student and Appointment and Consultant.

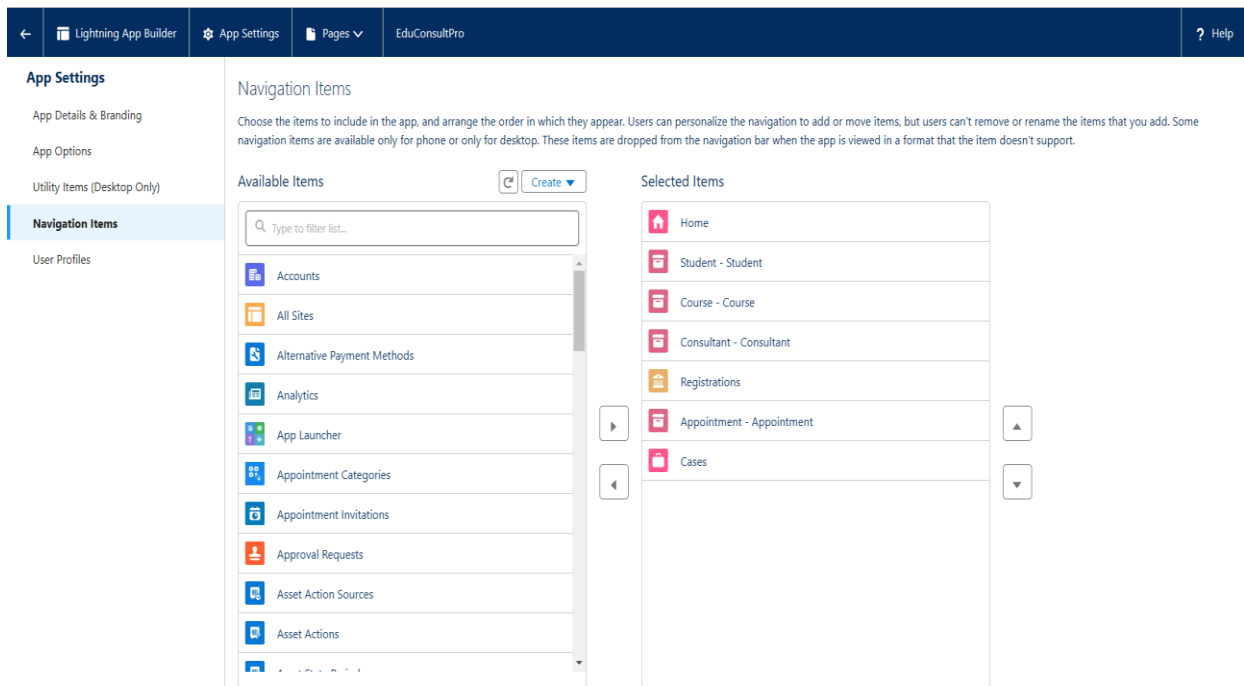
A new Registration object was created to hold student and course information and set up the lookup relationship on either an immigration or visa application regarding student queries between Student and Case.

It was set up to model the data regarding the requirement of the institution. For every new object, after that, tabs were created that would act as easy accesses. The Case object further customized Specified values entered for the "Type" field were Immigration and Visa Application those for "Status" were Open and In- progress. A new Lightning application, EduConsultPro, was developed that aggregates all these elements: home, students, courses, advisers, appointments, registrations, and cases. The interface was largely optimized for user experience between the system administrator and staff members, so it guaranteed proficient, effective, and transparent administration of all aspects of service within the Salesforce instance at the EduConsultPro Institute.

# Screenshots:



Lookup relationship among Objects



Adding items to EduConsultPro

Setup

Home

Object Manager

lightning app

User Interface

Lightning App Builder

Didn't find what you're looking for?

Try using Global Search.

SETUP

Lightning App Builder

The Lightning App Builder provides an easy-to-use graphical interface for creating custom Lightning pages for Salesforce Lightning Experience and mobile app. Lightning pages are built using Lightning components—compact, configurable, and reusable elements that you can drag and drop into regions of the page in the Lightning App Builder.

View: All

Create New View

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Other All

Lightning Pages

New

Action	Label	Name	Namespace Prefix	Description	Type	Created By	Last Modified By
<a href="#">Edit</a>   <a href="#">Clone</a>   <a href="#">Del</a>	EduConsultPro Home Page	EduConsultPro_Home_Page			Home Page	YSury 29/10/2024, 5:36 pm	YSury 29/10/2024, 5:36 pm
<a href="#">Edit</a>   <a href="#">Clone</a>   <a href="#">Del</a>	Enhancement Request Record Page	Enhancement_Request_Record_Page			Record Page	YSury 24/08/2024, 4:53 pm	YSury 24/08/2024, 4:53 pm
<a href="#">Edit</a>   <a href="#">Clone</a>   <a href="#">Del</a>	Home Page Default	Home_Page_Default			Home Page	YSury 24/08/2024, 4:53 pm	YSury 25/08/2024, 11:47 am
<a href="#">Edit</a>   <a href="#">Clone</a>   <a href="#">Del</a>	Research Proposal Record Page	Research_Proposal_Record_Page			Record Page	YSury 24/08/2024, 4:53 pm	YSury 25/08/2024, 12:12 pm

https://gayatrividyparishadcoll-1e-dev-ed.develop.lightning.force.com/lightning/setup/FlexiPageList/home

## EduConsultPro Lightning App

Setup

Home

Object Manager

Search Setup

Star

Plus

Help

Settings

Notifications

Profile

SETUP > OBJECT MANAGER

Case

Details

Fields & Relationships

Case Page Layouts

Case Close 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

Hierarchy Columns

Scoping Rules

Picklist Values Used

Active picklist values 7 (1,000 max)

Inactive picklist values 0

Field Dependencies

New

Field Dependencies Help

No dependencies defined.

Validation Rules

New

Validation Rules Help

No validation rules defined.

Case Type Picklist Values

New

Reorder

Replace

Printable View

Chart Colors

Case Type Picklist Values Help

Action	Values	API Name	Default	Chart Colors	Modified By
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	Mechanical	Mechanical	<input type="checkbox"/>	Assigned dynamically	Vanapelli SuryaTejaswini 24/08/2024, 4:53 pm
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	Electrical	Electrical	<input type="checkbox"/>	Assigned dynamically	Vanapelli SuryaTejaswini 24/08/2024, 4:53 pm
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	Electronic	Electronic	<input type="checkbox"/>	Assigned dynamically	Vanapelli SuryaTejaswini 24/08/2024, 4:53 pm
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	Structural	Structural	<input type="checkbox"/>	Assigned dynamically	Vanapelli SuryaTejaswini 24/08/2024, 4:53 pm
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	Other	Other	<input type="checkbox"/>	Assigned dynamically	Vanapelli SuryaTejaswini 24/08/2024, 4:53 pm
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	Immigration	Immigration	<input type="checkbox"/>	Assigned dynamically	Vanapelli SuryaTejaswini 27/10/2024, 12:52 pm
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	Visa Application	Visa Application	<input type="checkbox"/>	Assigned dynamically	Vanapelli SuryaTejaswini 27/10/2024, 12:52 pm

Inactive Values

No Inactive Values values defined.

## Configuring the Case Object

## Task2

# Create a ScreenFlow for Student Admission Application process

The EduConsultPro Student Flow simplifies the registration process for students by creating a more user-friendly application. I added the Screen element, represented as a "**Student Info**," which was there to carry all the information that pertained to the student. Therefore, I created a record variable known as **StudentRecordRes** to view the fields from the student object and present a very light experience when getting data.

Next, I developed a Create element named "**Create Student Record**" to enable saving the student record input to the Salesforce database. Then I included a "**Course Screen**" by including a second Screen element. The primary choice found in the screen is a picklist named "**Select Course**". The course options that may be selected are *IELTS, GRE, GMAT, Duolingo, and TOEFL*.

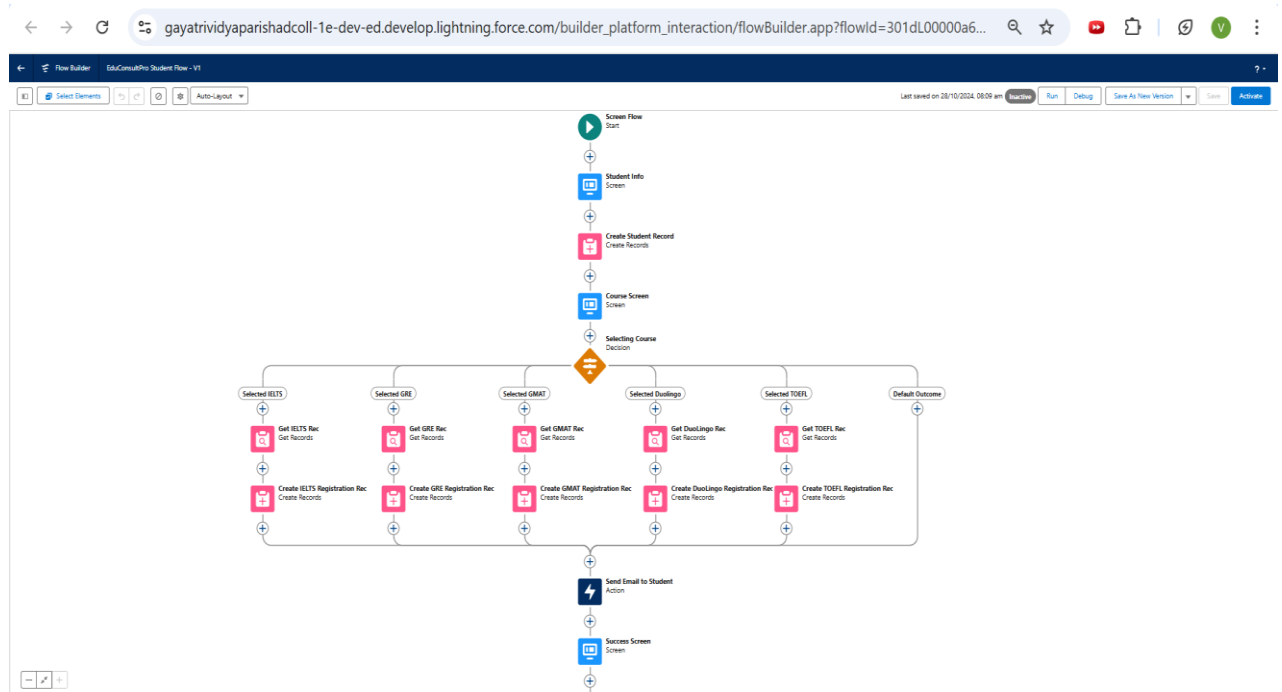
Added a Decision element known as "**Selecting Course**" to manage course selection. This checks the course selected and then directs the flow of action regarding the selection. Based on the selection made, under the condition of the choice, a Get Record element gets the related course record from the "**Course**" object so that registration is done correctly.

After collecting the course data that was needed, a Create Records element was leveraged to make a registration record in the "**Registration**" object, associating the selected course with all the information that student gave to me. I then made Text Template Resources for the body and subject of the email to add the message that would be sent to the student ensuring that they had properly registered and sought to join the platform.

I then added an Action element titled "**Send Email to Student**" for sending this email—including the capability to fill in the email with provided student details any time the process is run. To exit the process system, I added an Screen element titled "**Success Screen**" and then added a message, "Registration has been completed successfully," with advice to students to check their emails for details. This flow will be named hence: "**EduConsultPro Student Flow**."



# Screenshot:



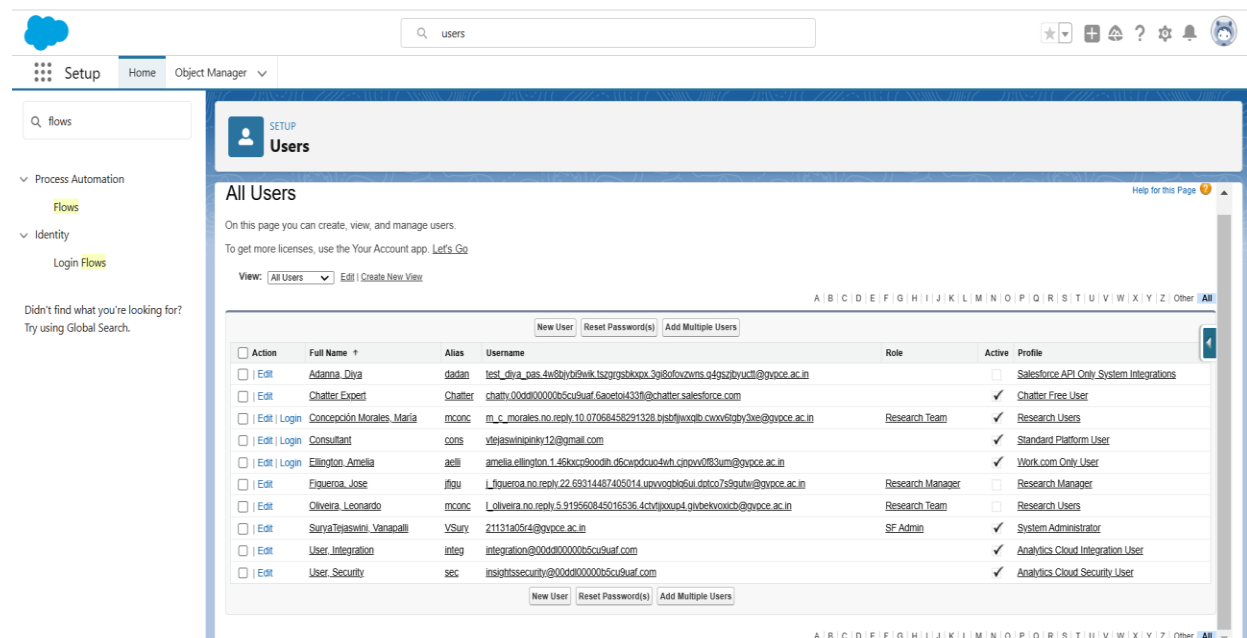
EduConsultPro Student Flow

# Task3

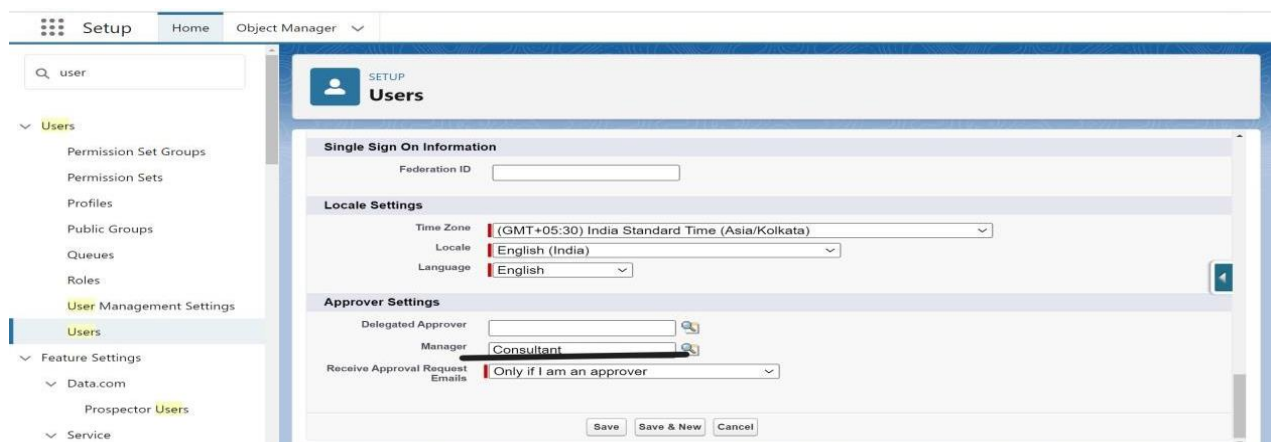
## Create Users

Created new user in Salesforce by navigating to Setup→Administrative→Users→New User. Entered "Consultant" for the second name and entered "Salesforce Platform" as the license type. I applied the profile "Standard Platform User," checked all required fields for data, and saved the record of the new user.

Configured user settings: Browse to Setup → Administration → Users → Edit the user profile Scrolled down to the Approver Settings section, Selected "Consultant" in the Manager Field, At the bottom of the screen, click Save.



User



## Configure the User Settings

# Task4

## Create an Approval Process for Property Object

Created email template in Salesforce. From Setup, I input "Templates" in the Quick Find box and clicked on "Lightning Email Templates", and clicked the toggle to on. Opened the App Launcher, searched for "Email Templates" from the search box, and saved. Created a new folder with the desired name. Created an new email Template and selected the new created folder. The specified text got pasted into the HTML Value to create this, in my case, "Submission Template"Designed two more email templates for approving and rejecting requests, ditto the first one.

Earlier, I did set up the approval process. I went to Setup and typed "Approval" in the Quick Find box, and clicked on "Approval Processes." I selected to manage approval processes for "Appointment" and clicked "Create New Approval Process" using the Jump Start Wizard. I named the process "Appointment Approval" and under the approver settings, I used the setting "Option Automatically assign an approver using a standard or custom hierarchy field" of "Manager." I set the "Next Automated Approver Determined By" field to "Manager."

Made the following changes to the Record Editability Properties: Prependded "Administrator OR the currently assigned approver" to the list Saved the process and checked out the form Made a new Field Update under the Initial Submission Actions with Name "Submitted" and configured it to make the update to the field "Appointment: Status" with "Pending"

Also created a new Email Alert type "Submission Email Alert," auto-populate the Unique Name, selected the "Submission Template" Email Template, selected the recipient type to the user's Name. Repeated that for the Final Approval and Final Rejection actions and set those two up the same way.

# Screenshots:

## Email Template

The screenshot shows the 'Email Templates' section of a CRM system. At the top, there's a search bar and navigation tabs for Sales, Home, Dashboards, Registrations, Opportunities, Leads, and Tasks. The 'Email Templates' section has a 'Recent' tab selected, showing a list of 4 items. The list has columns for 'Email Template Name', 'Description', and 'Folder'. The templates listed are 'Email template', 'Rejection of Request', 'Approval of Request', and 'Submission Template', all in the 'DesiredFolfer' folder. A sidebar on the left shows navigation options like 'Created by Me', 'Private Email Templates', 'Public Email Templates', 'All Email Templates', 'FOLDERS', 'All Folders', 'Created by Me', and 'Shared with Me'.

EMAIL TEMPLATES	Email Template Name	Description	Folder
Recent	Email template		DesiredFolfer
Created by Me	Rejection of Request		DesiredFolfer
Private Email Templates	Approval of Request		DesiredFolfer
Public Email Templates	Submission Template		DesiredFolfer
All Email Templates			

The screenshot shows the 'Approval Processes' section of a CRM system. The 'Appointment - Appointment: Appointment Approval' process is selected. The 'Process Definition Detail' section shows the process name, unique name, description, entry criteria, record editability, and approval assignment email template. The 'Initial Submission Actions' section shows a list of actions: 'Record Lock', 'Email Alert', and 'Field Update'. The 'Approval Steps' section is also visible.

Process Name	Unique Name	Description	Entry Criteria	Record Editability	Approval Assignment Email Template
Appointment Approval	Appointment_Approval			Administrator OR Current Approver	Appointment - Appointment Owner

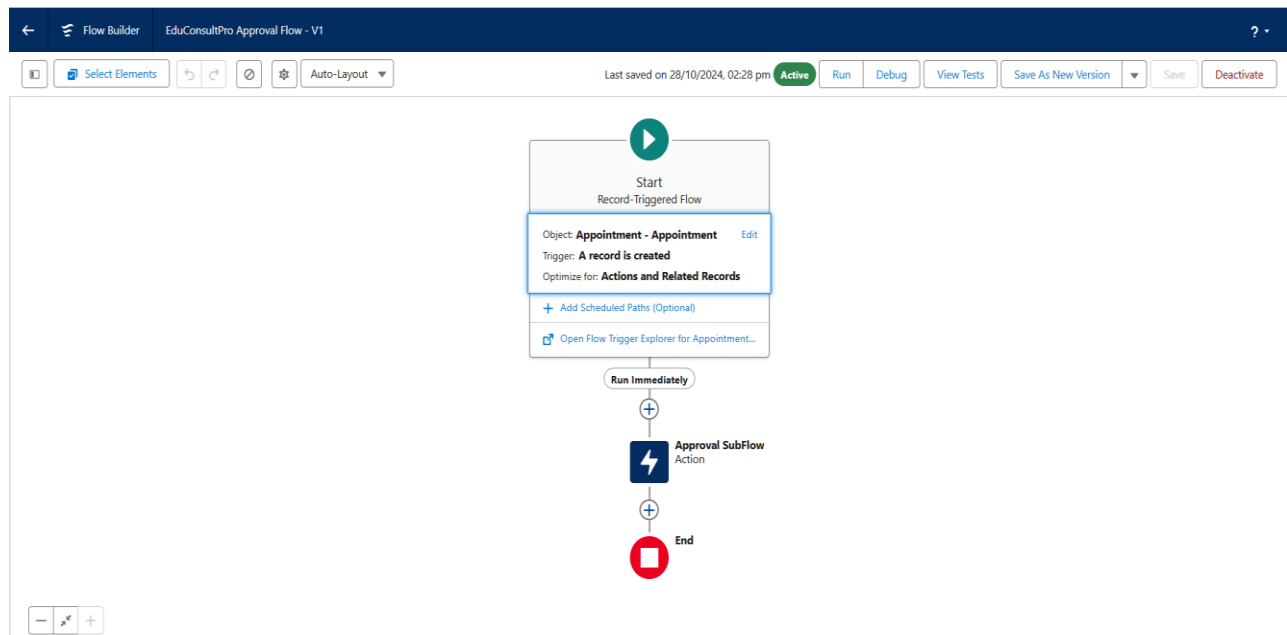
## Approval Process

# Task5

## Create a Record Triggered Flow

Created a record-triggered flow in Salesforce for appointment approvals. Set the flow trigger to 'When a record is created' on an appointment record and added an action element that would submit the record for approval. Label this Action element as "Approval SubFlow" and set RecordId to "{!\$Record.Id}". At this point, the flow was saved with the label "EduConsultPro Approval Flow" and activated.

### Screenshot:



## **Task6**

### **Create a ScreenFlow for Existing Student to Book an Appointment**

Configured an end-to-end flow in Salesforce for student appointments and cases management.

Added a Screen element called "Get Student Info" to collect the student's name and email, and added a GET Record element to retrieve student details depending on the name and email collected.

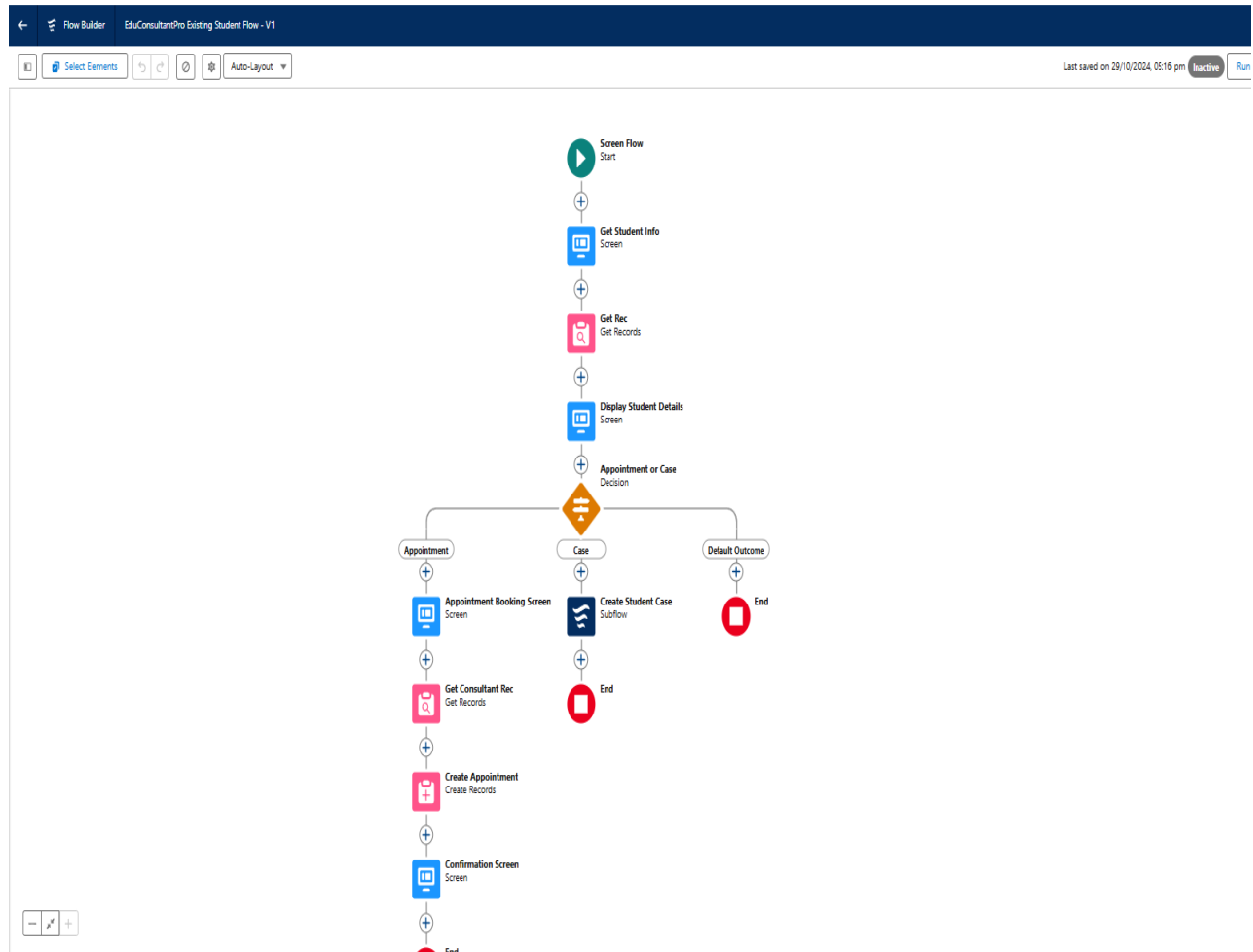
Added a Decision element to route between "Appointment" and "Case" depending upon the student's wishes. Added a Screen element labeled "Appointment Booking Screen" to gather details for appointments in case an appointment has been booked.

Inserted a GET Record element to obtain consultant details based on the selected name of the consultant. Applied a Create Records element to create a new appointment record with appropriate details.

Added a Screen element and named it "Confirmation Screen" to let the student see details of the appointment confirmation.

Added a Subflow element for case management and named it "Create Student Case" for creation of a case. Finally, save the flow as "EduConsultantPro Existing Student Flow" and prepare for deployment.

## Screenshot :



EduConsultantPro Existing Student Flow

# Create a ScreenFlow to Combine all the flows at one place

A new Screen Element in the flow was added labelled as "Welcome Screen". Inside this screen, a Display Text component was added and was labelled as "SuccessMessage."

Entered the given text into the Resource Picker box so that the welcome message by EduConsultantPro describes the services and support it provided. Added another Screen Element following the Welcome Screen, labeled "Existing or New Student Confirmation Screen."

Added a Radio Button component with the label "Are you an Existing Student?". Created two choices: "Yes" and "No." Then added an Action Element of type Decision and named it "Decision 1," placing it next to the Existing or New Student Confirmation Screen. Set up the decision criteria with the outcome labeled "If Existing Student." The condition will be when the resource `{!Are_you_a_Existing_Student}` is equal to the value `{!Yes}`. Added the second outcome for "No."

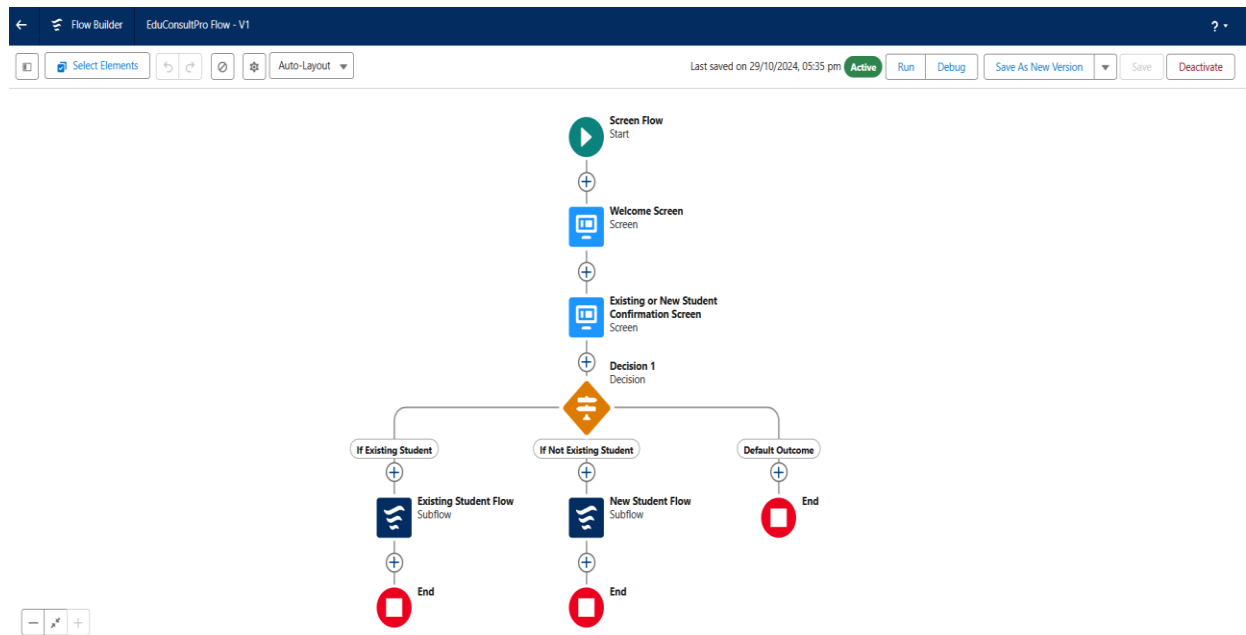
Inserted a Subflow Element in the "If Existing Student" path and selected "EduConsultantPro Existing Student Flow" and labelled it as "Existing Student Flow." Saved the flow under the name "EduConsultantPro Existing Student Flow."

Inserted another Subflow Element in the "If Not an Existing Student" path, selecting the "EduConsultantPro Student Flow" and labeling it "New Student Flow." Saved this flow with the name "EduConsultPro Flow."

Finalized by saving the entire flow, naming it "EduConsultPro Flow."



# Screenshot:

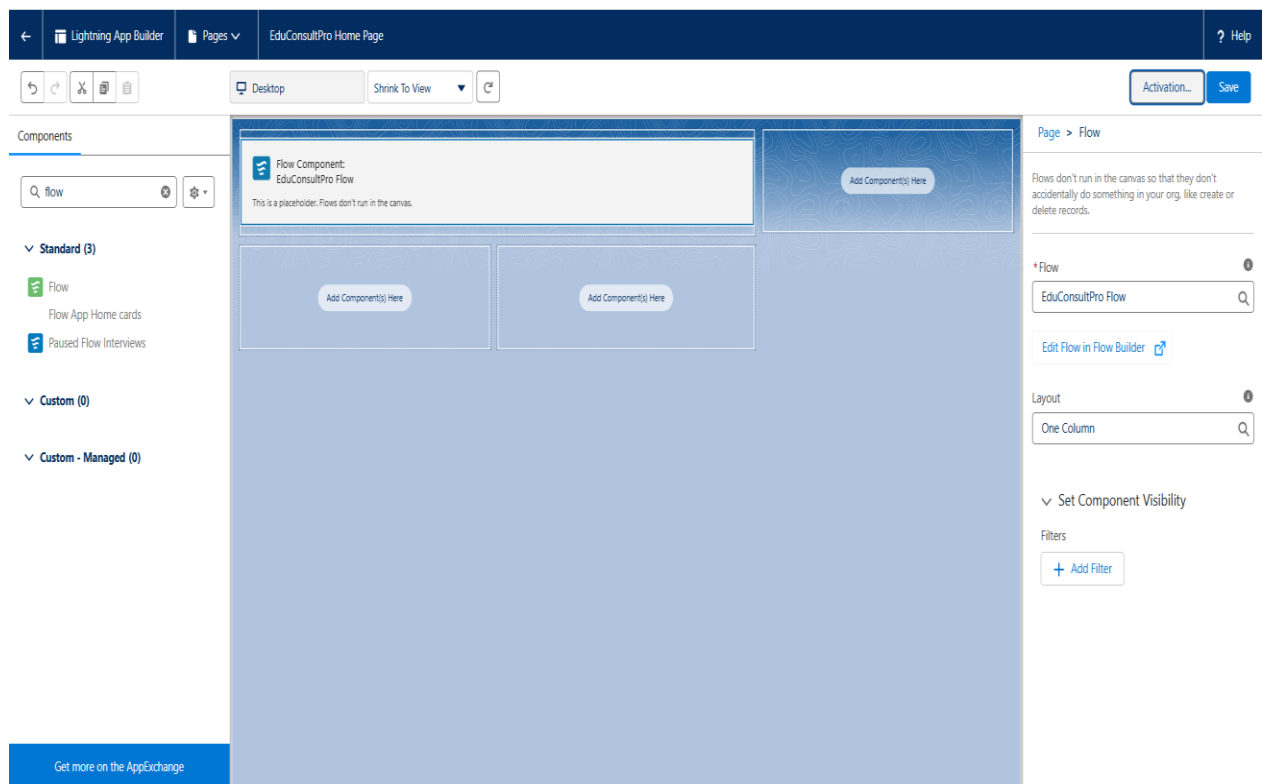


EduConsultPro Flow

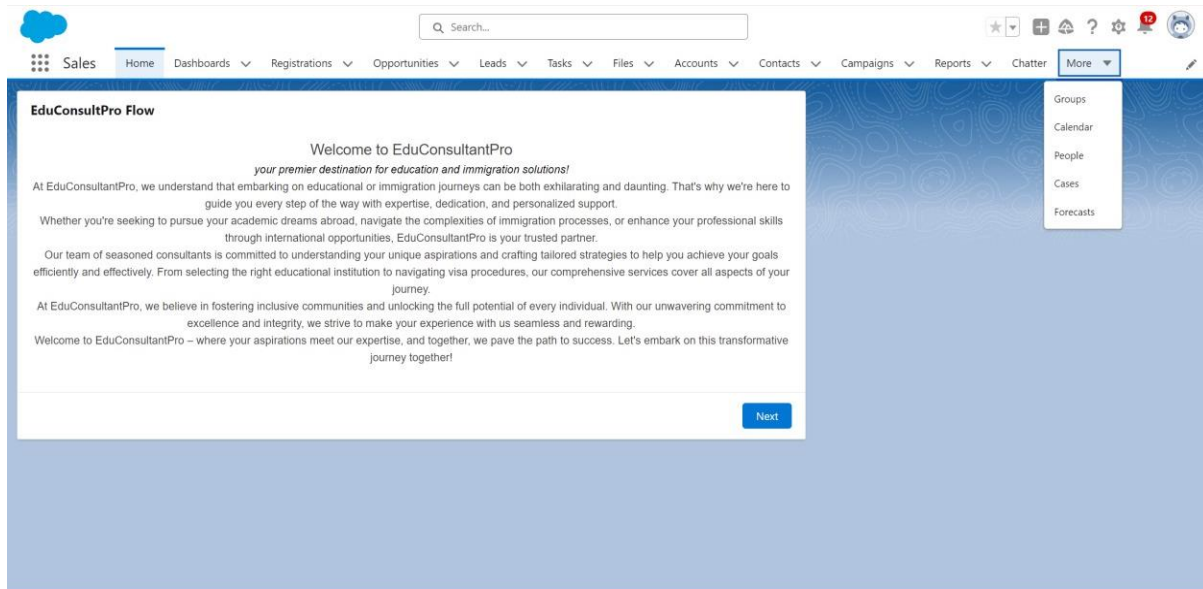
# Create a lightning app page

Created and configured a Lightning App Page named "EduConsultPro Home Page" using the Standard Home Page template. Placed the Flow component in the top-right region and incorporated the "EduConsultantPro Flow." Activated the page and assigned it to the Sales app and the System Administrator profile, ensuring it was available for the designated users.

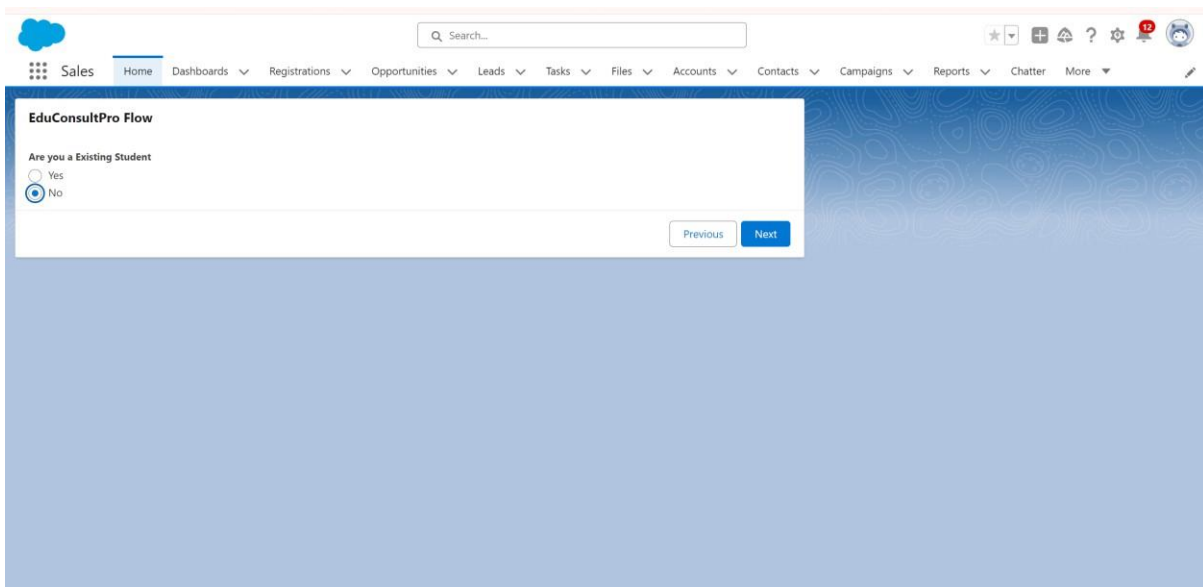
## Screenshots:



# Execution Screenshots:



To Do List







## Status Update Trigger:

```
trigger ApplicationStatusUpdateTrigger on Application__c (after update) {  
  
    // List to hold email notifications for applicants  
    List<Messaging.SingleEmailMessage> emails = new  
    List<Messaging.SingleEmailMessage>();  
  
    for (Application__c app : Trigger.new) {  
  
        // Check if the status has changed by comparing the old and new values  
        Application__c oldApp = Trigger.oldMap.get(app.Id);  
        if (app.Status__c != oldApp.Status__c) {  
  
            // Prepare email notification for the applicant  
            Messaging.SingleEmailMessage email = new  
            Messaging.SingleEmailMessage();  
            email.setToAddresses(new List<String>{app.Applicant_Email__c});  
            //Replace with applicant's email field  
            email.setSubject('Update on Your Application Status: ' + app.Name);  
            email.setPlainTextBody('Dear ' + app.Applicant_Name__c + ',\n\n' +  
            'Your application status has been updated to: ' + app.Status__c + '.\n\n' +  
            'Please contact us if you have any questions or need further assistance.\n\n' +  
            'Thank you,\nEduConsultPro Admissions Team');  
            emails.add(email);  
        }  
    }  
}
```

```
// Send the emails to applicants
if (!emails.isEmpty()) {
    Messaging.sendEmail(emails);
}
}
```

**THANKYOU**