A CRM Application to Manage the Services offered by an Institution

Scope of the Project:

The project aims to implement Salesforce CRM to revolutionize the admission and enquiry processes at EduConsultPro Institute. It will involve automating the entire admission workflow, from managing prospective student inquiries to finalizing enrollments. The CRM will serve as a centralized system, enabling admissions staff to efficiently handle applications, track enquiries, and manage case interactions. Students will benefit from a seamless application experience, real-time updates, and personalized support. The project scope also includes integrating Salesforce's analytics features to provide insights into admission trends and decision-making. Additionally, the system will facilitate smooth communication between students and the admissions team, ensuring transparency throughout the process.

Objective of the Project

The primary objective is to streamline EduConsultPro Institute’s admission and enquiry management by leveraging Salesforce CRM. It aims to enhance the student experience by simplifying application processes, ensuring prompt query resolution, and delivering transparent updates. The project seeks to empower admissions staff with tools for efficient application review, data tracking, and case management, reducing manual tasks and errors. Furthermore, the objective is to increase the institute's operational efficiency and student satisfaction through improved process clarity and faster turnaround times. By implementing this solution, EduConsultPro will establish itself as a technologically advanced institution, meeting the growing demands of prospective students and stakeholders.

Admission Application Management

Students can submit detailed admission applications via the institute's portal, with all data stored in Salesforce CRM. Automated email notifications confirm submissions, and staff gain insights through dashboards analyzing application metrics and trends.

Approval Process Requirements

An automated approval process reviews consulting requests, with email alerts for approvals or rejections. Requests are automatically submitted upon creation for seamless processing.

Consulting Services Management

Students can request consulting services through the portal, with details stored in Salesforce CRM. Consultants receive notifications, manage requests, and schedule appointments directly within Salesforce, with status tracking for all interactions.

Immigration Case Management

Students can submit immigration cases through various channels, with details stored in Salesforce CRM. Agents track and process cases with integrated tools for document management and collaboration, ensuring streamlined updates on case status.

What is Salesforce?

Salesforce is a leading cloud-based customer relationship management (CRM) platform that helps businesses manage and optimize interactions with customers, clients, and stakeholders. It provides a suite of tools and services to streamline sales, marketing, customer service, and business operations. Salesforce enables organizations to collect, store, and analyze customer data, automate workflows, and enhance collaboration across teams.

With customizable applications, Salesforce caters to a wide range of industries, offering solutions for everything from lead management and marketing automation to analytics and AI-driven insights. It is known for its flexibility, scalability, and integration capabilities, making it a preferred choice for businesses of all sizes seeking to improve efficiency and customer engagement.

Creating Developer Account:

Before the start of the project development,we need to create an account in salesforce developer edition. This involves the following steps:

Creating a developer org in salesforce.

1. Go to https://developer.salesforce.com/signup

2. On the sign up form, enter the following details :

1. First name & Last name

2. Email

3. Role : Developer

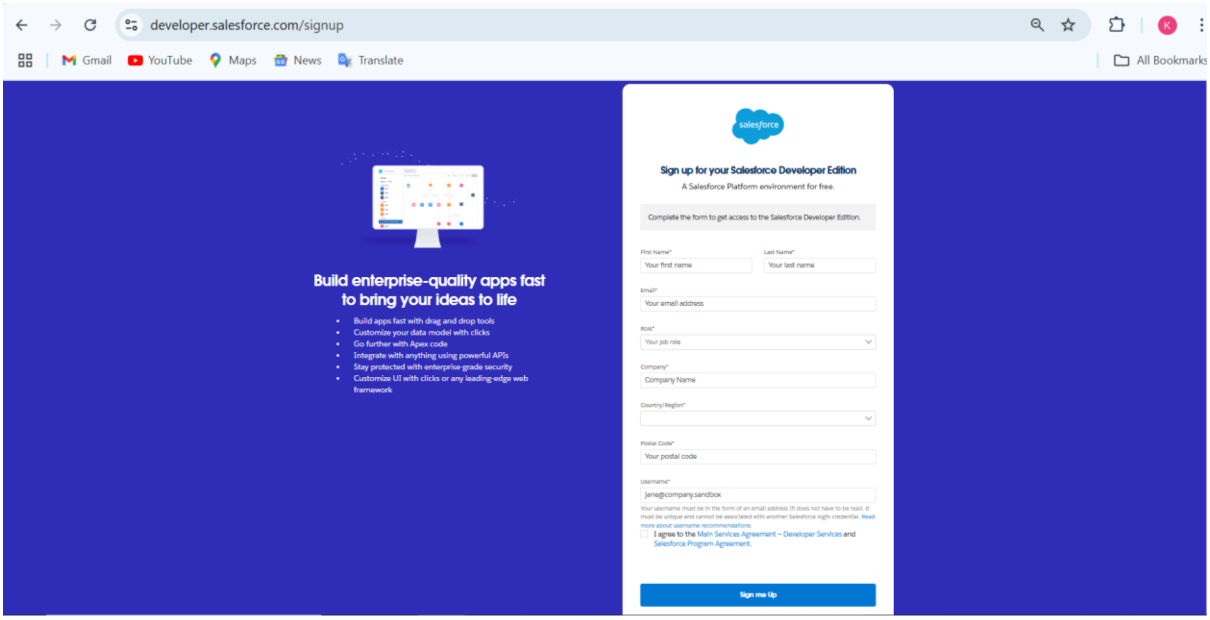
4. Company : College Name

5. County : India

6. Postal Code : pin code

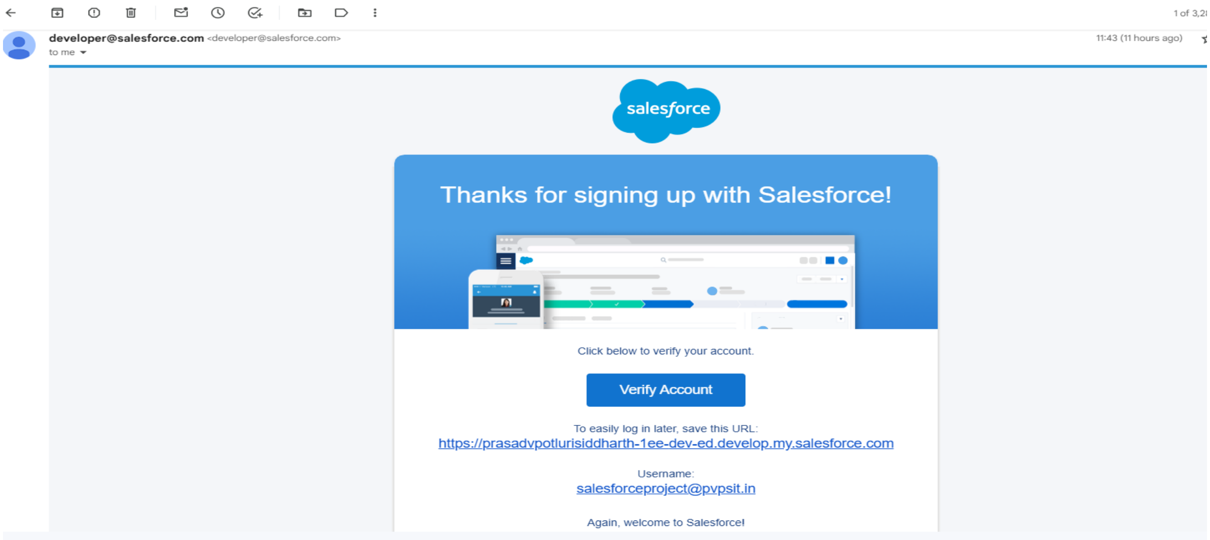
7. Username : should be a combination of your name and company
This need not be an actual email id, you can give anything in the format :
username@organization.com

Click on sign me up after filling these.



Account Activation

1. Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account.



MileStone 1: Create Objects from Spreadsheet

What Is an Object?

Salesforce objects are database tables that permit you to store data that is specific to organiza tion. What are the types of Salesforce objects Salesforce objects are of two types:

1. **Standard Objects:** Standard objects are the kind of objects that are provided by salesforce.com such as users, contracts, reports, dashboards, etc.

2. **Custom Objects:** Custom objects are those objects that are created by users. They supply information that is unique and essential to their organization. They are the heart of any application and provide a structure for sharing data.

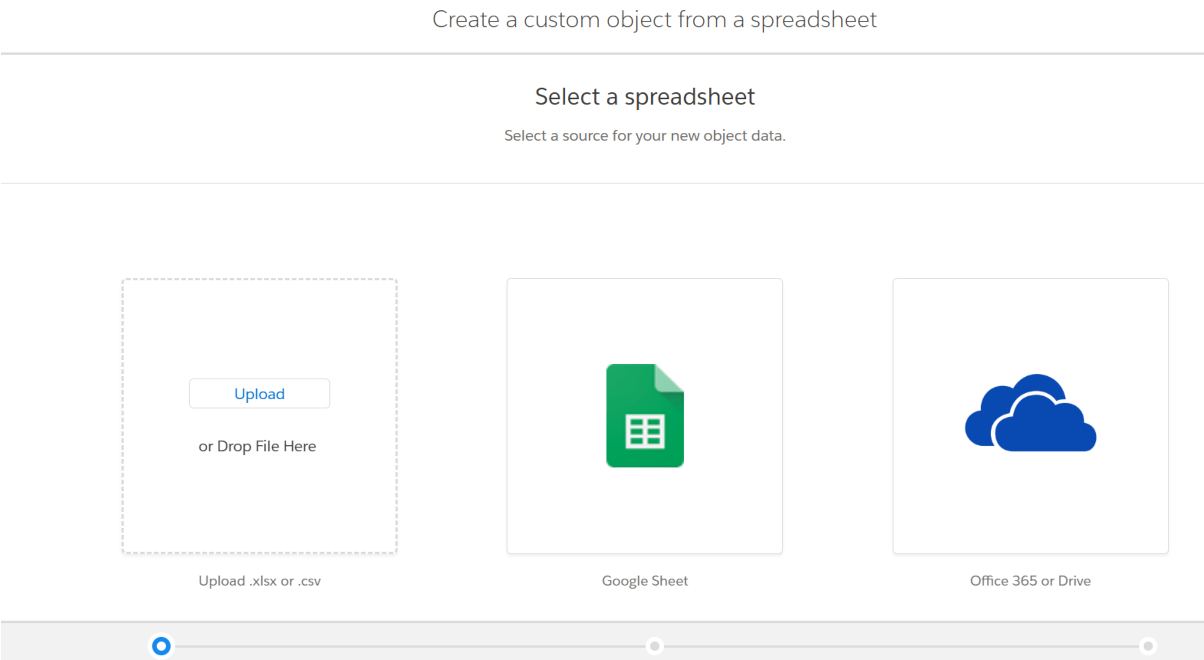
Create Course object

1. Go to your object manager and and click on create object from spreadsheet
2. Click on the link to get the spreadsheet, Course.

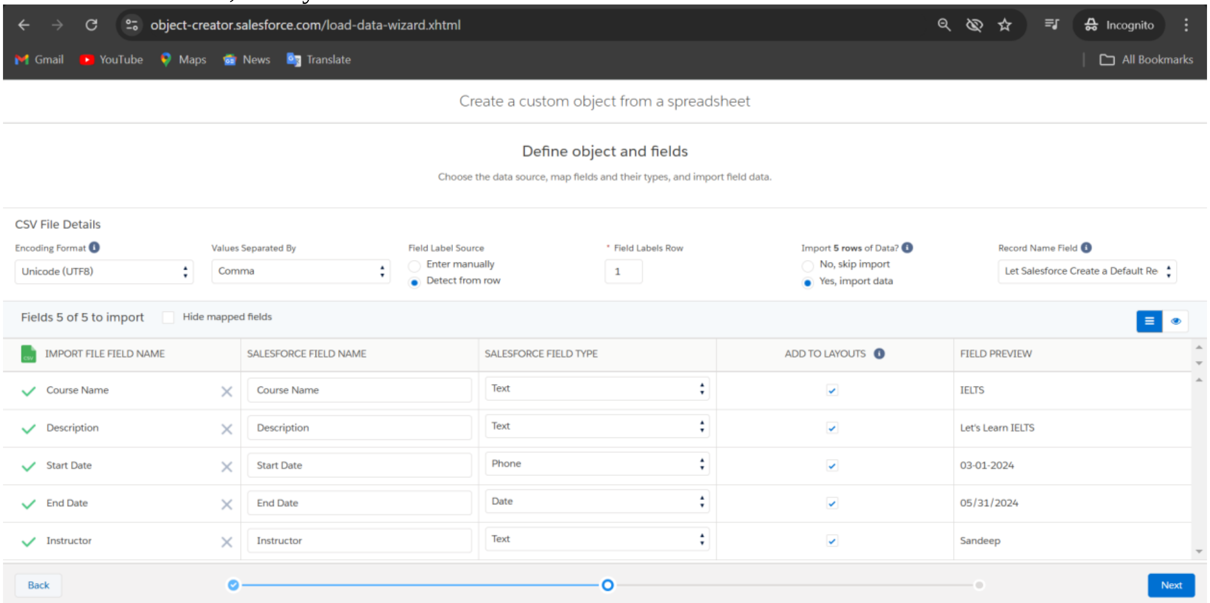
A	B	C	D	E
Course Name	Description	Start Date	End Date	Instructor
IELTS	Let's Learn IELTS	03-01-2024	05/31/2024	Sandeep
GRE	Let's Learn GRE	03-01-2024	07-11-2024	Shivam
TOEFL	Let's Learn TOEFL	04-01-2024	07/27/2024	Prajwal
DuoLingo	Let's Learn DuoLingo	04-01-2024	09/14/2024	Sameer
GMAT	Let's Learn GMAT	04-01-2024	01-09-2024	Sanjay

Download this spreadsheet from the link in .csv format.

After downloading, upload the file, map the fields and upload to create an object.



Click next and next , finally save the data.

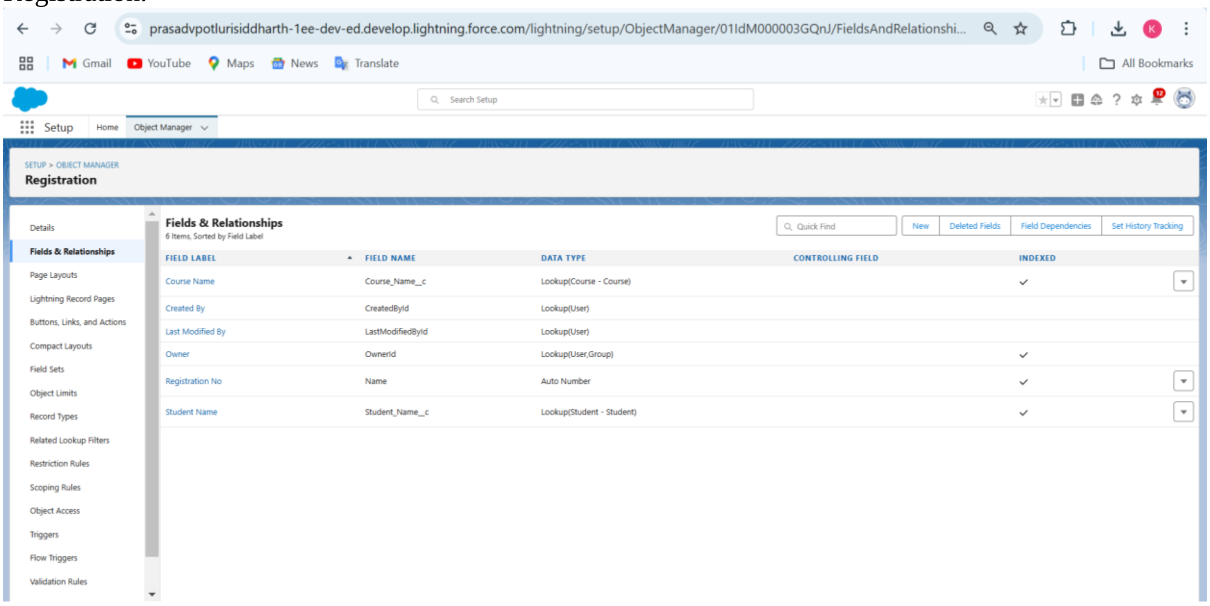


Follow the steps which we have followed for course object creation to create the remaining objects .Use the excel sheet data given in the module for following objects.

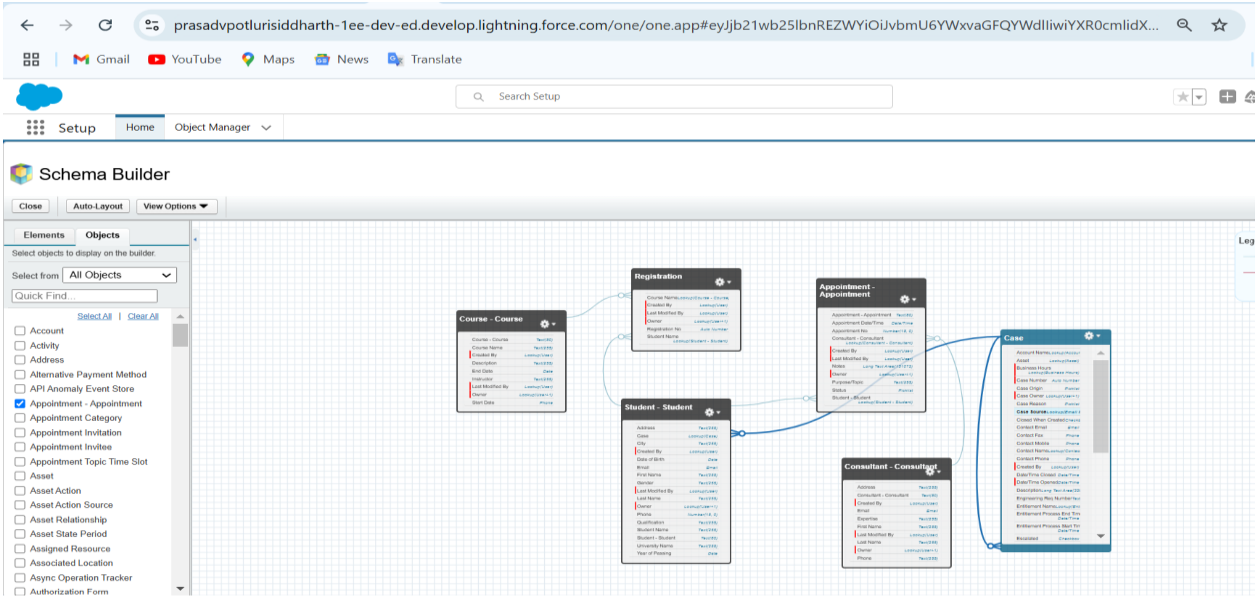
a. Consultant
b. Student
c. Appointment

Create Relationship among the objects

1. Create lookup between appointment and student, appointment and consultant.For that open object manager and search for appointment object and select it.
   Open fields and relationships and select New and then create the loop up relationship.
2. Create an object to store the information student and course details with the name Registration.Refer the data model for more information about the fields and relationships of object Registration.

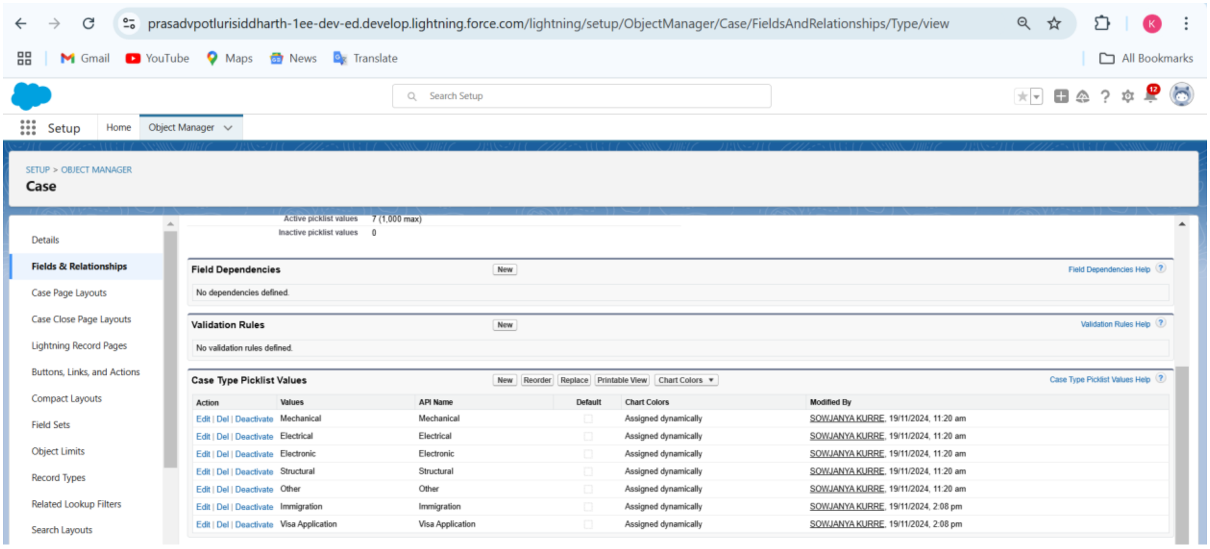


3. Also create a lookup between student and case to store the student queries for immigration or visa application.
4. The data model should be similar to the below Data Model with fields & relationships:

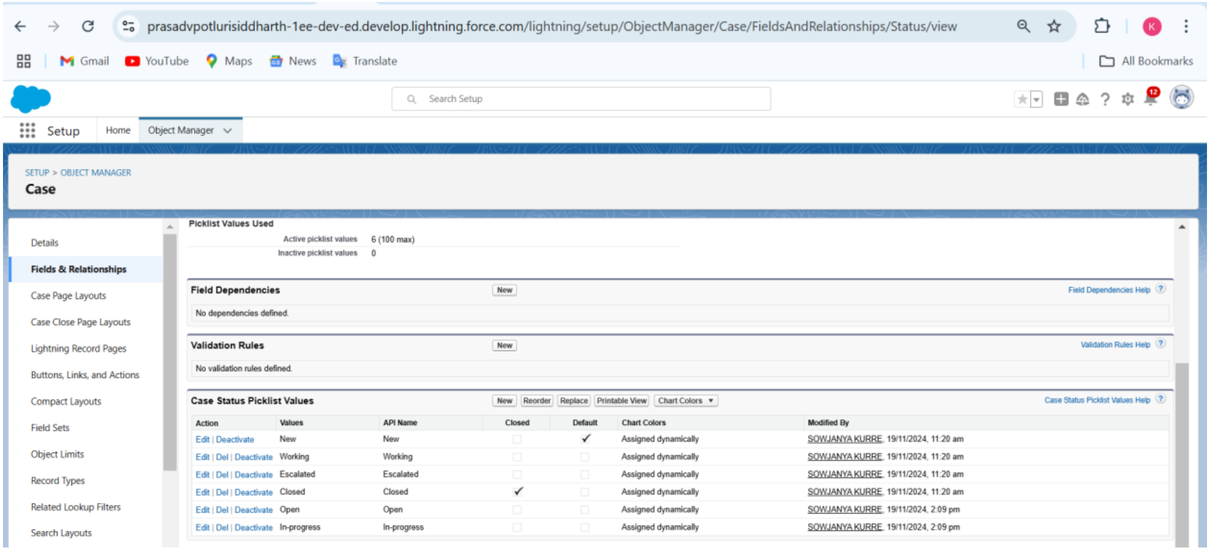


Configure the Case Object

1. Go to object manager, edit case object.
2. Select the "Type" field and add the values in it.
   Immigration
   Visa Application



1. Now Select the "Status" field and add the values in it.
   Open
   In-progress.



Before steping into creation of a lightning app it is important to create the tabs for the objects that we created earlier.

Tabs

What is Tab : A tab is like a user interface that is used to build records for objects and to view the records in the objects.

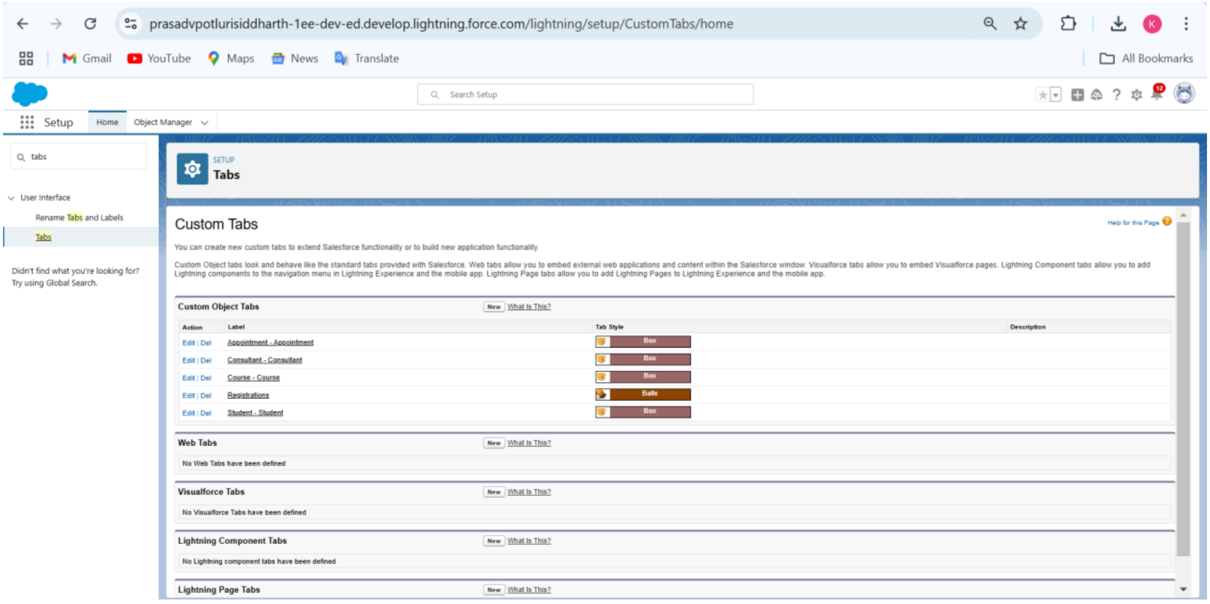
Creating a Custom Tab

1. Go to setup page >> type Tabs in Quick Find bar >> click on tabs >> New (under custom object tab)

2. Select Object>> Select the tab style >> Next (Add to profil page) keep it as default >> Next (Add to Custom App) uncheck the include tab .

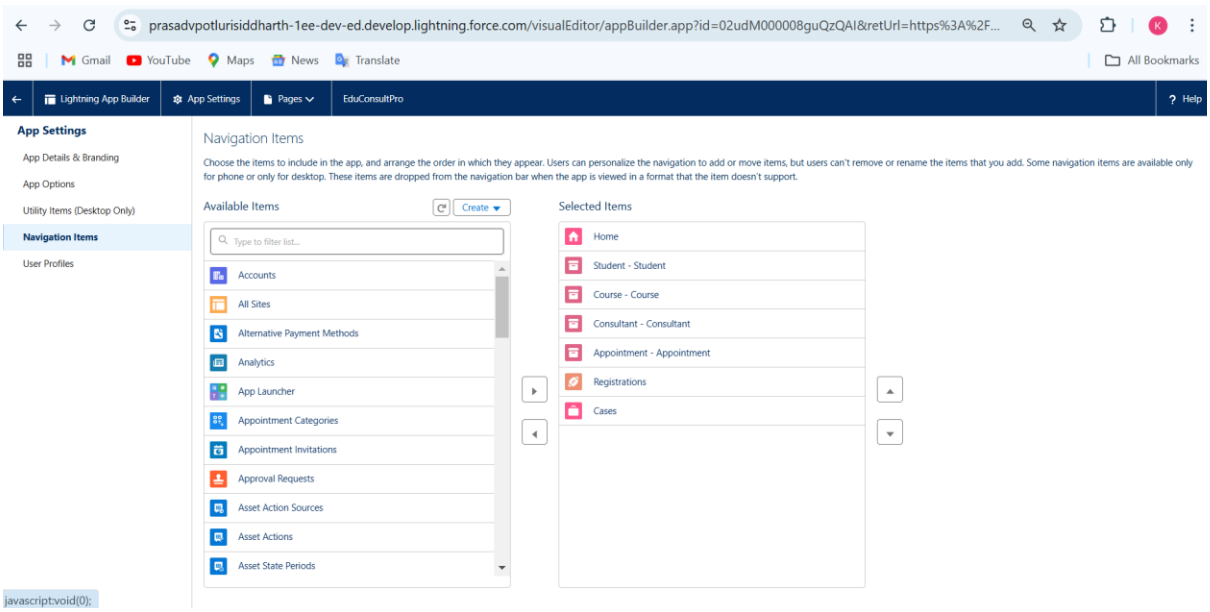
3. Make sure that the Append tab to users' existing personal customizations is checked.

4. Click save.



Create a Lightning App

1. Go to Setup, search for the App Manager in quick find
2. Click on New Lightning App
3. Give app name as “EduConsultPro”, click Next, Next, Next
4. Add Home, Students, Courses, Consultants, Appointments, Registrations, and Cases from the Available Items to Selected Items.



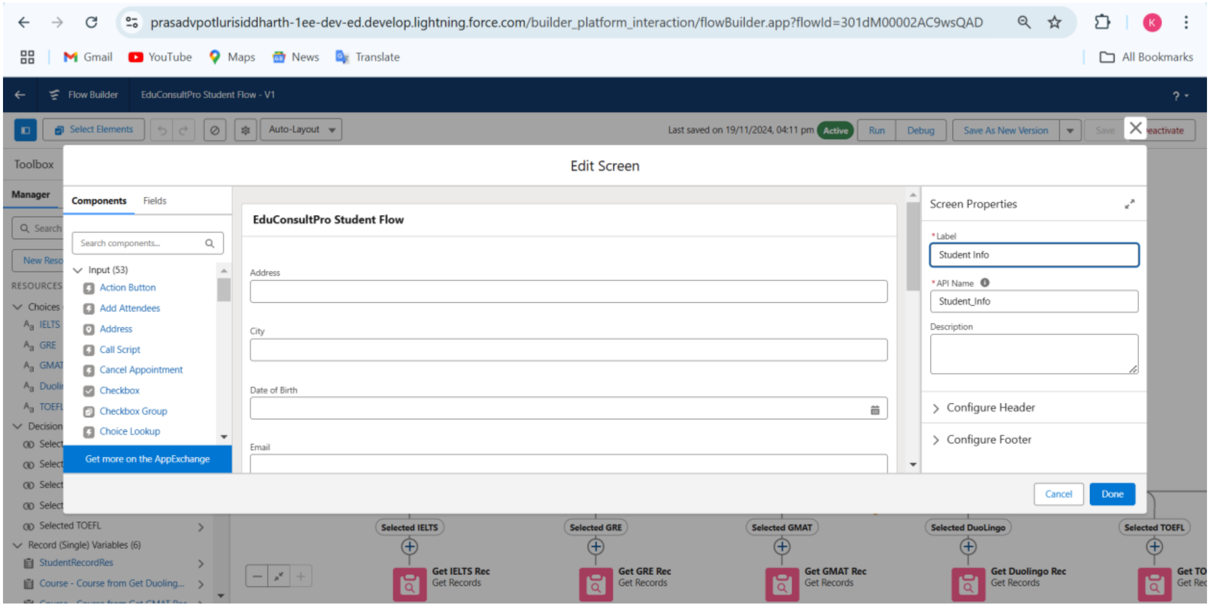
5. Add “System Administrator” profile from Available Profiles to Selected Profiles, click Save & Finish.

   A Lightning app with name EduConsultPro is created successfully.

MileStone 2 :Create a ScreenFlow for Student Admission Application process.

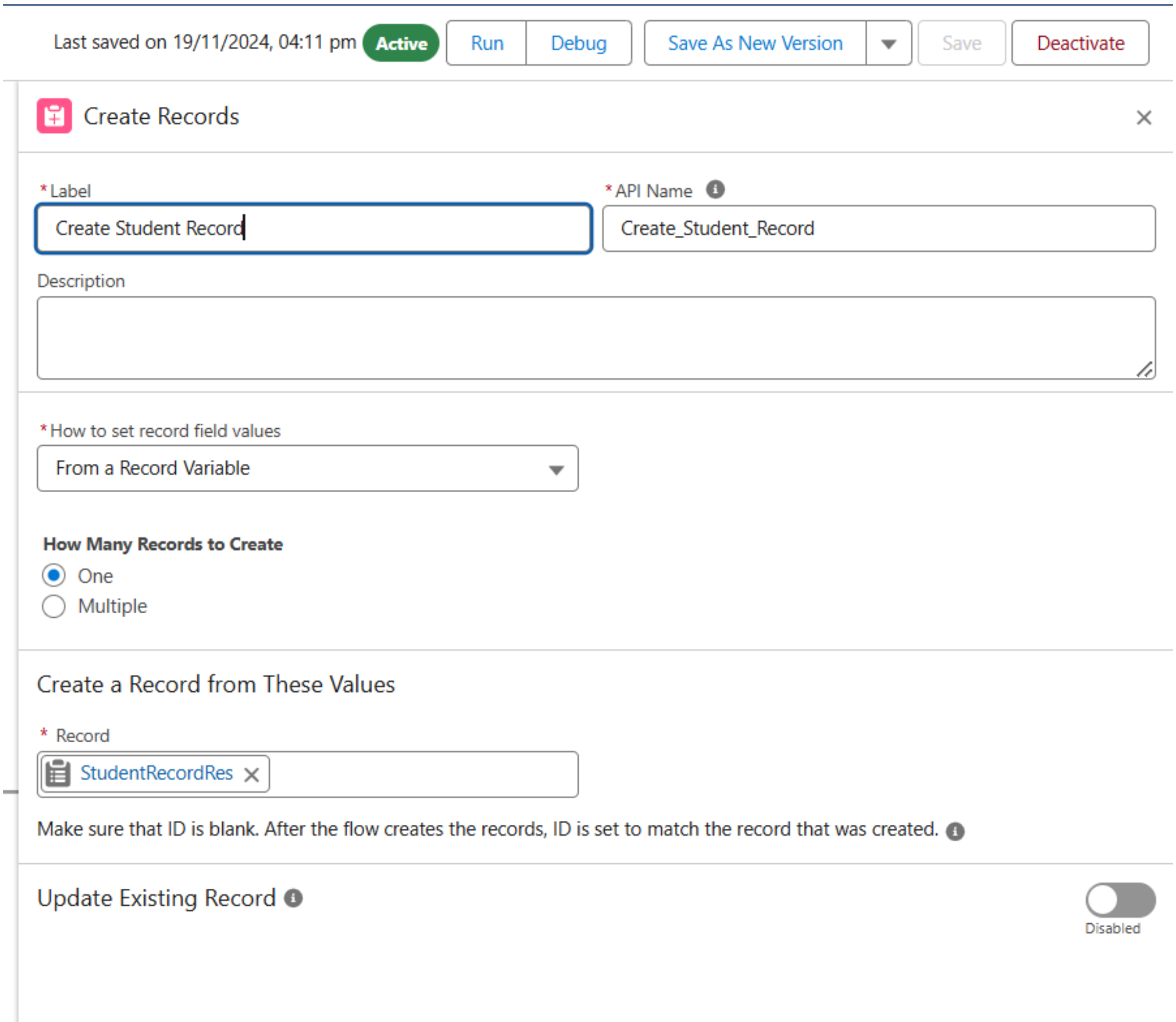
Add Screen Element

1. From Setup, enter Flow Builder in quick find, select new flow --> ScreenFlow.
2. Add a Screen element.
3. In the Screen Properties pane, for Label, enter “Student Info”.
4. Click on Fields, click on the record variable input and create a new Resource(StudentRecordRes) to display all the fields which are in the student object. Drag all the fields which are needed to add on the screen inorder to collect the student information.



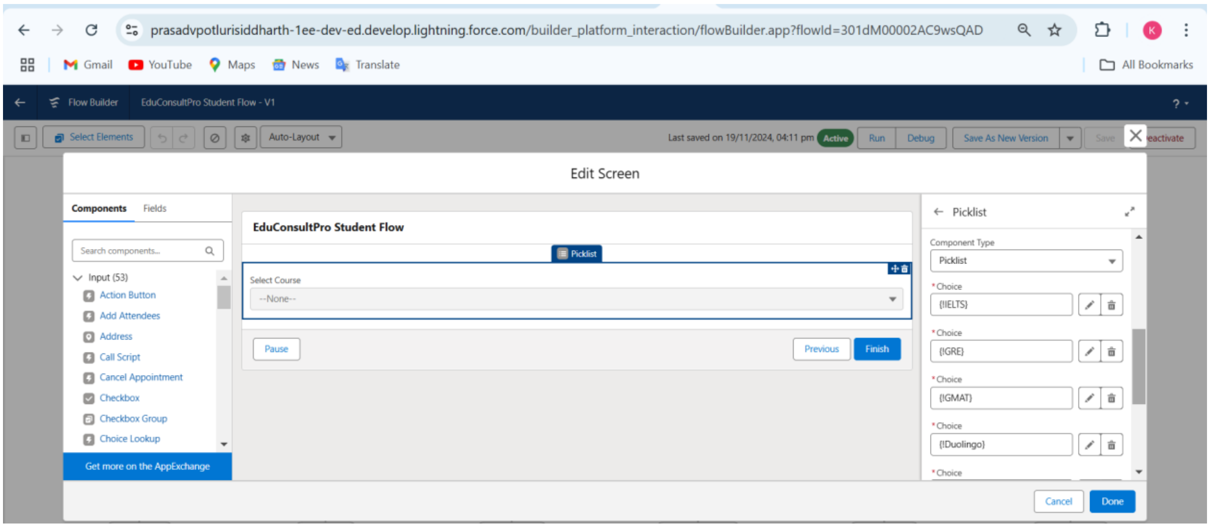
Create Student Record using Create Element

1. Add a Create element after Student Info Screen Element, Label it as “Create Student Record.”
2. Select “one” under How many records to Create, and select “use all values from a record” under How to Set the record fields.
3. Select the record variable resource(StudentRecordRes) which we have created in the Student Info screen element, under Create a record from these values.



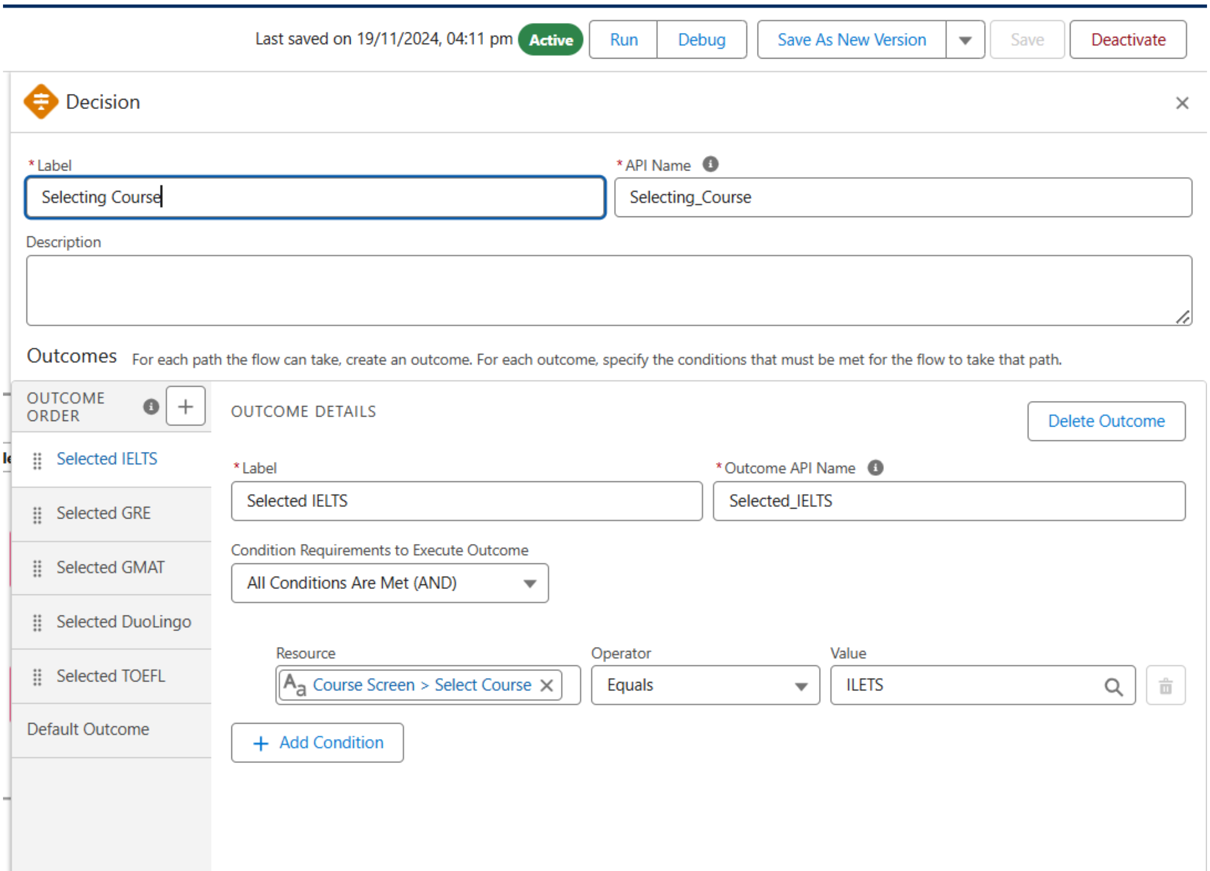
Add Screen Element

1. Add a Screen Element after Create Student Record Element and label it as Course Screen.
2. Add a picklist component from the left side panel label it as “Select Course”, under choices type “IELTS” and enter. This creates a variable with the name IELTS.
3. Repeat the same for GRE, GMAT, Duolingo, TOEFL.



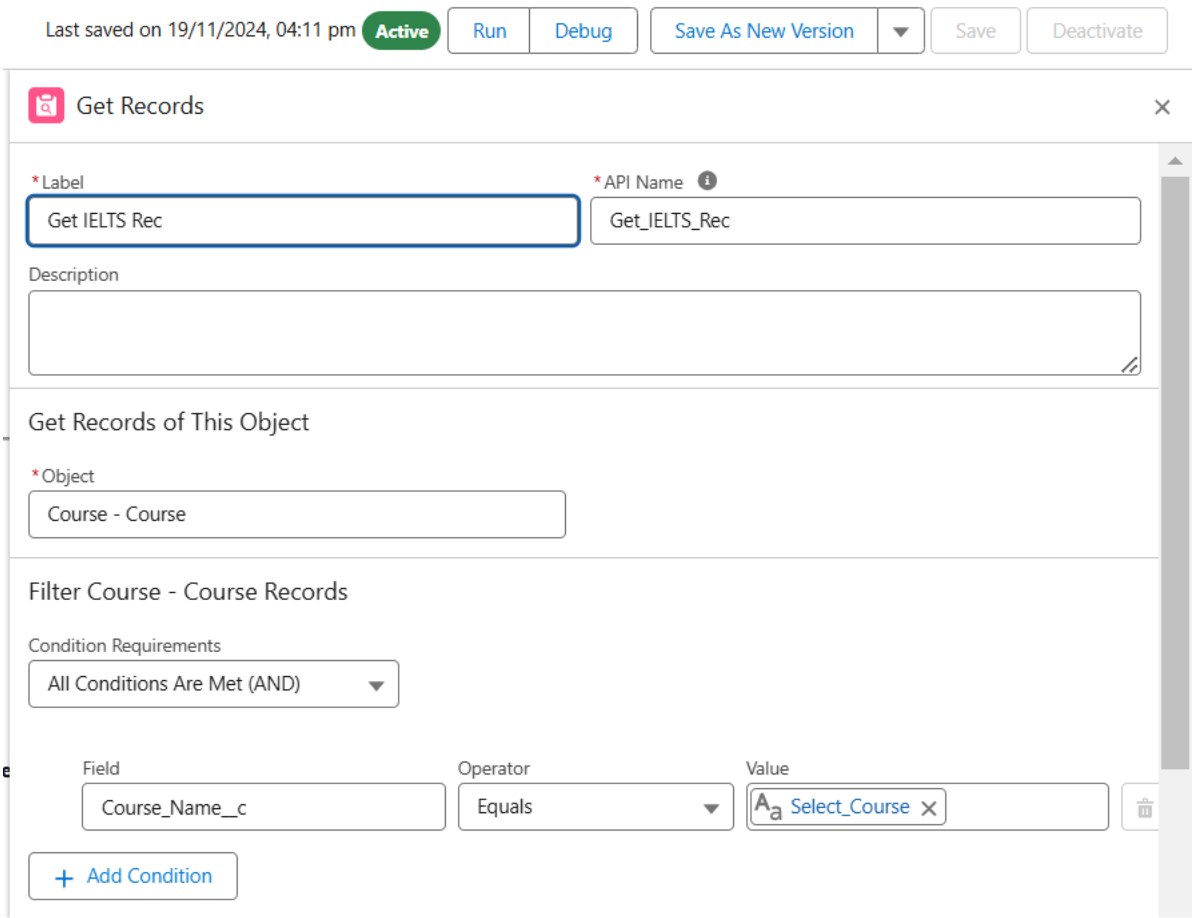
Add Decision Element

1. Add a Decision Element after Select Course Screen Element, label it as Selecting Course.
2. Under outcome label it as “Selected IELTS” and write the condition such as below:
   Resource : Select_Course (Screen Component from Select Course Screen Element)
   Operator : Equals
   Value : IELTS (Choice Variable from Select Course Screen Element)
3. Click on the “+” icon and Repeat step 2 for other options mentioned as below:
   a) GRE
   b) GMAT
   c) DuoLingo
   d) TOEFL
4. Click Done.



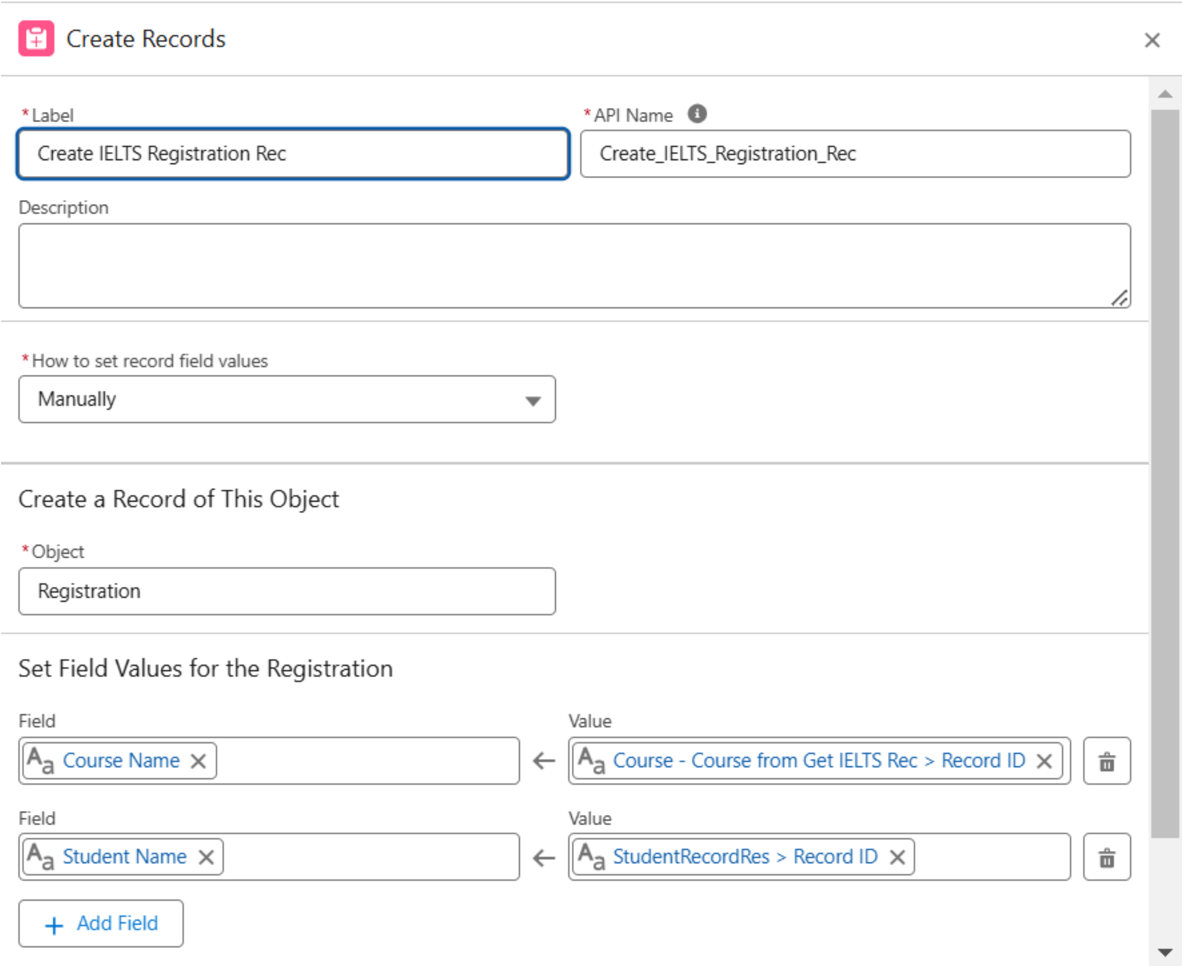
Add GET Record Element

1. Add a GET Record Element after Decision Element, under the IELTS path and label it as “Get IELTS Rec”.
2. Select Object : Course
   Condition Requirement : All Conditions are Met(AND)
      Field : Course Name
      Operator : Equals
      Value : {!Select_Course}



Create Registration Record using Create Records Element

1. Add a Create element after the Get IELTS Rec element and label it as "Create IELTS Registration Rec".
2. Select "one" under How many records to Create, and select "Use separate resources, and literal values" under How to Set the record fields.
3. Select Object : Registration
   Field : Course_Name__c
   Value : {!Get_IELTS_Rec.Id}
   Field : Student_Name__c
   Value : {!StudentRecordRes.Id}

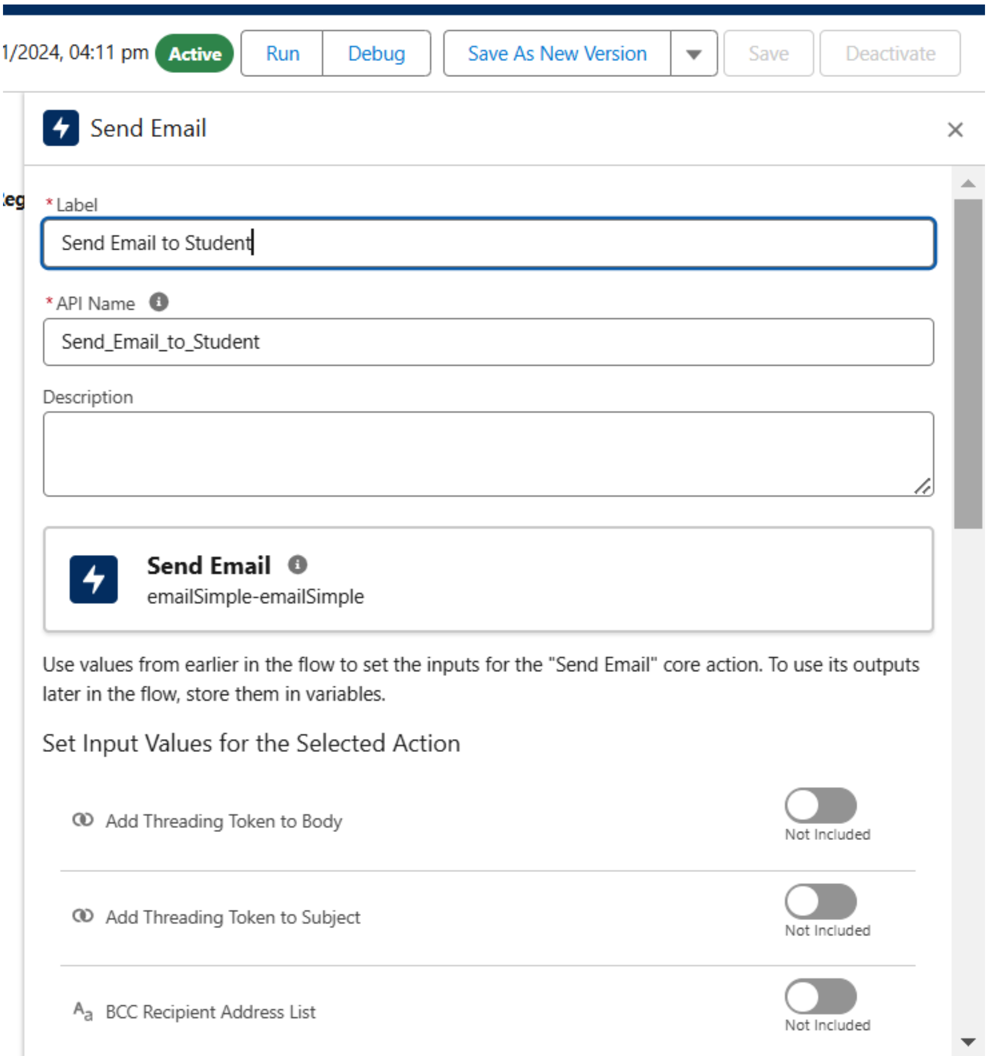


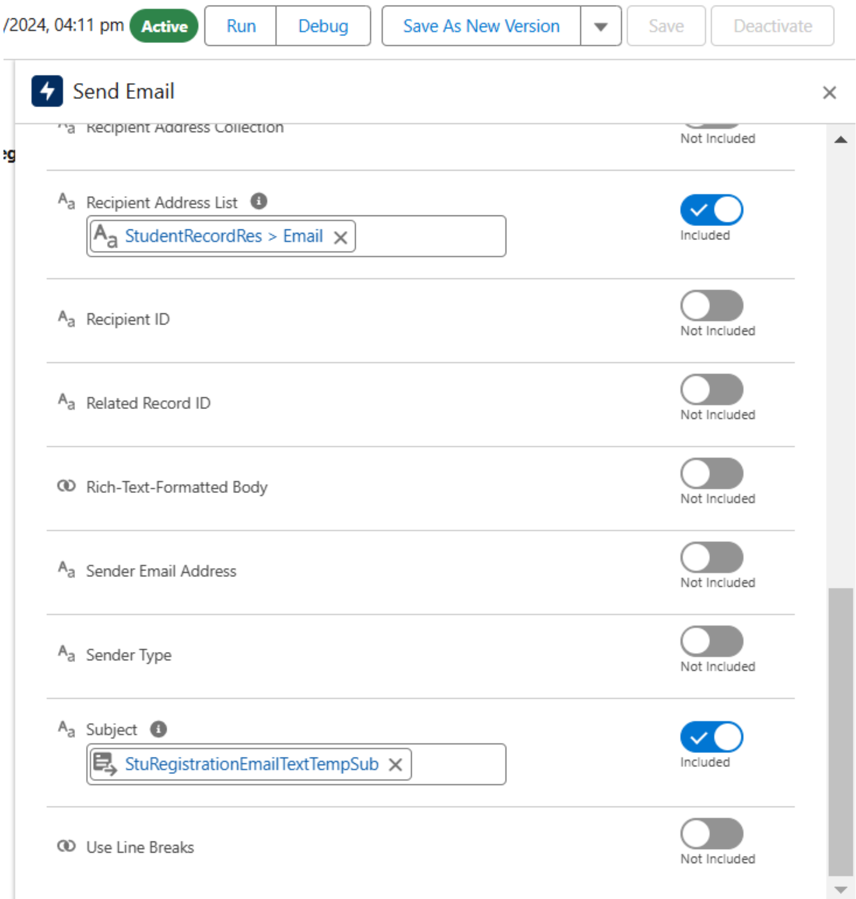
Create Email Text Template Variables for email body and subject

1. Click on the toggle toolbox on the left corner, click “New Resource”, then select “Text Template” as Resource Type.
2. Give the API name as “StuRegistrationEmailTextTempBody”, select “view as plain text” and paste the message text in body as below and follow same to create for subject.

Add an Action Element

1. Add an Action Element after all the Decision paths, label it as “Send Email to Student”.
2. Under “Set input values for selected action”, include body, Recipient Address List and Subject.
3. For input Body : {!StuRegistrationEmailTextTempBody},
   Recipient Address List : {!StudentRecordRes.Email__c},
   Subject : {!StuRegistrationEmailTextTempSub}.


1/2024, 04:11 pm
Active
Run
Debug
Save As New Version
Save
Deactivate
Send Email
*Label
Send Email to Student
*API Name
Send_Email_to_Student
Description
Send Email
emailSimple-emailSimple
Use values from earlier in the flow to set the inputs for the "Send Email" core action. To use its outputs later in the flow, store them in variables.
Set Input Values for the Selected Action
Add Threading Token to Body
Not Included
Add Threading Token to Subject
Not Included
BCC Recipient Address List
Not Included



Add Screen Element

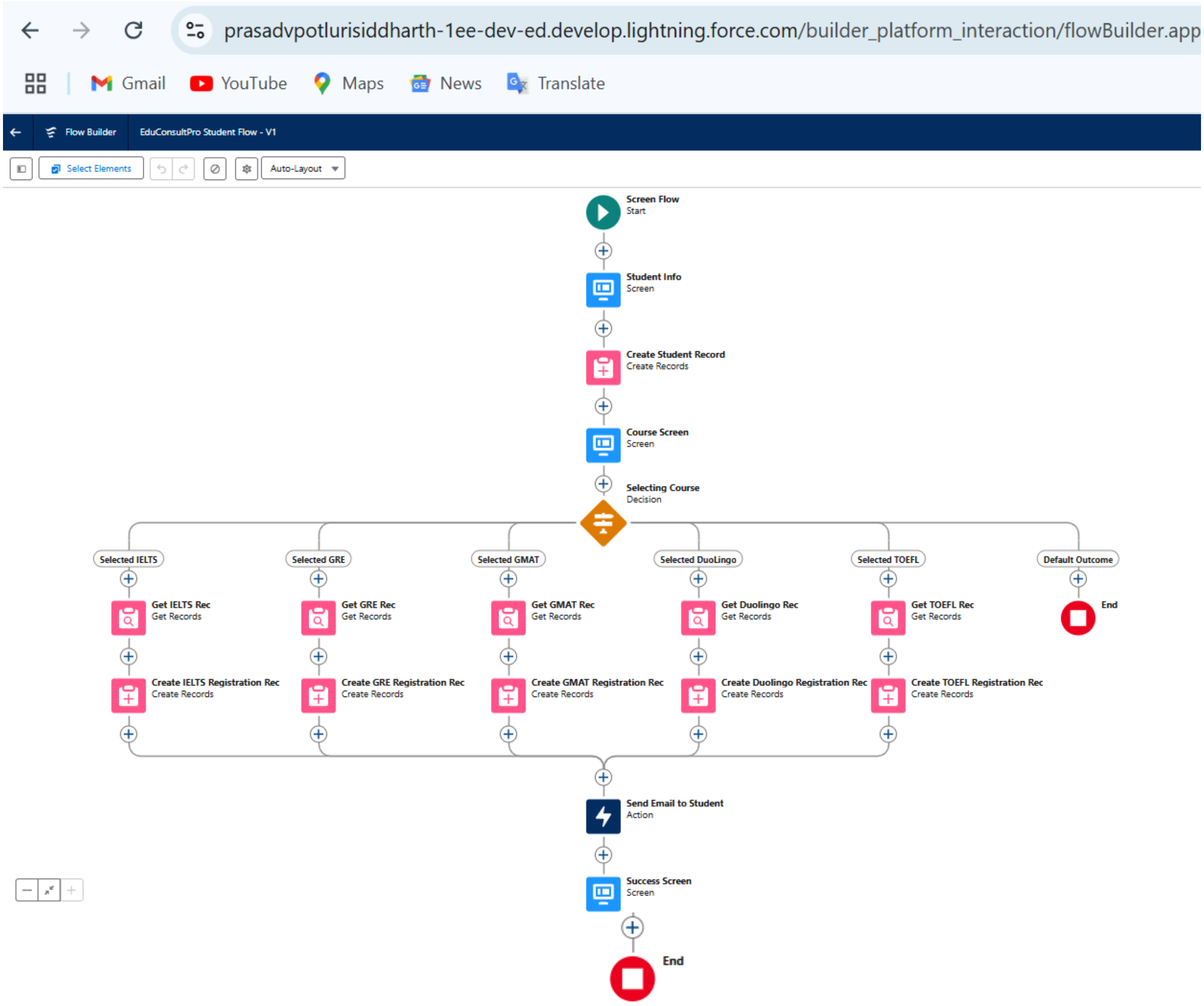
1. Add a Screen Element after the Send Email to Student Action Element, label it as Success Screen.
2. From the left side panel search for the Display text component and drag it to the main panel, label it as "SuccessMessage".
3. Paste the below in the Resource picker box.
   "Dear {!StudentRecordRes.Name},

   Congratulations and welcome to EduConsultantPro!
   We are delighted to inform you that your registration on our platform has been successfully completed. You are now part of our esteemed community dedicated to empowering students like you to achieve their educational and immigration aspirations.

   Your Registration details have been sent through mail kindly check it once.

   Thank you."

4. Click Done.
5. Save the flow and name it as "EduConsultPro Student Flow". Your flow will look as shown below:

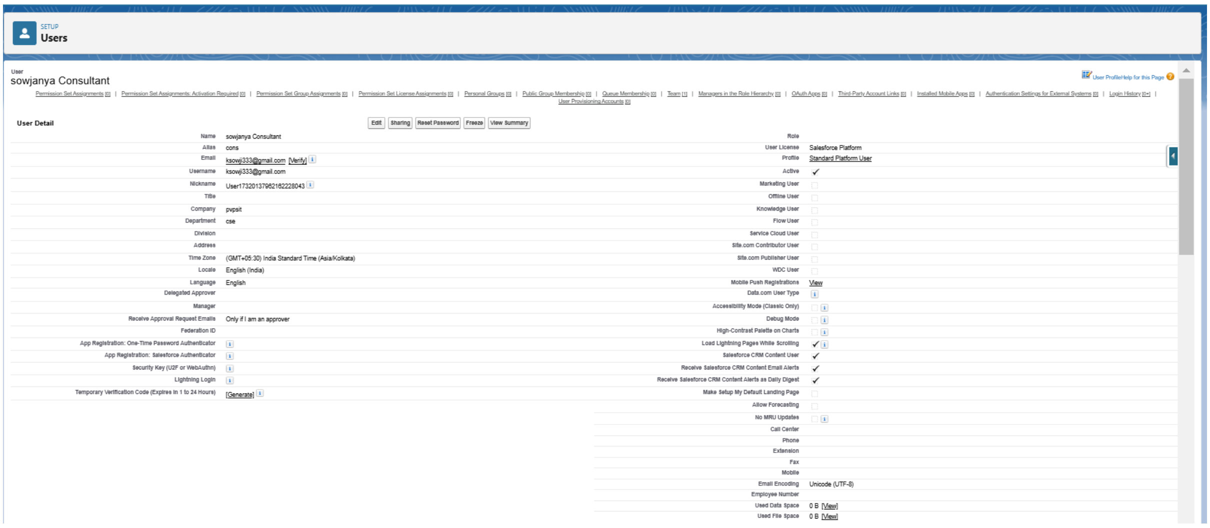


prasadvpotlurisiddharth-1ee-dev-ed.develop.lightning.force.com/builder_platform_interaction/flowBuilder.app
Gmail
YouTube
Maps
News
Translate
Flow Builder
EduConsultPro Student Flow - V1
Select Elements
Auto-Layout
Screen Flow
Start
Student Info
Screen
Create Student Record
Create Records
Course Screen
Screen
Selecting Course
Decision
Selected IELTS
Selected GRE
Selected GMAT
Selected DuoLingo
Selected TOEFL
Default Outcome
Get IELTS Rec
Get Records
Get GRE Rec
Get Records
Get GMAT Rec
Get Records
Get Duolingo Rec
Get Records
Get TOEFL Rec
Get Records
End
Create IELTS Registration Rec
Create Records
Create GRE Registration Rec
Create Records
Create GMAT Registration Rec
Create Records
Create Duolingo Registration Rec
Create Records
Create TOEFL Registration Rec
Create Records
Send Email to Student
Action
Success Screen
Screen
End

MileStone 3:Create Users

User

1. Go to Setup --> Administration --> Users --> New User
2. LastName : Consultant
3. License : Salesforce Platform
4. Profile : Standard Platform User
5. Fill all the mandatory fields & Save.

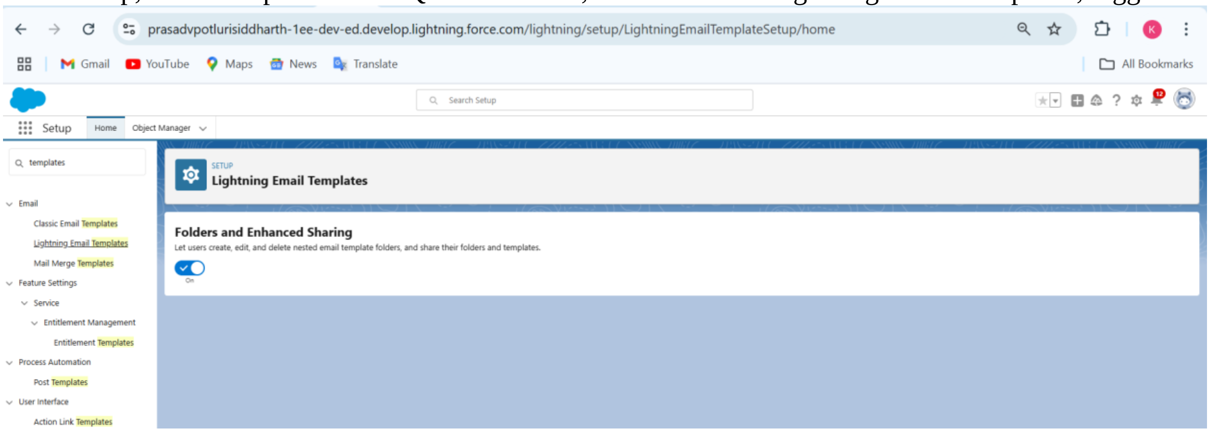


Configure the User Settings

1. Go to Setup --> Administration --> Users --> click Edit next to your name
2. Scroll down to bottom, under Approver Settings, Select “Consultant” the Manager Field.
3. Click Save.

MileStone 4: Create an Approval Process for Property Object

From Setup, enter Templates in the Quick Find box, and then select Lightning Email Templates, toggle on.



1. go to app launcher, search for “Email Templates”, Create a new folder with the desired name.Here I named it as emails.
2. Then create a new email template, select the folder which we have created in the previous steps, enter the below text in the HTML Value and Save it as "Submission Template".

"Dear {{{Appointment__c.Student_Name__c}}},
I hope this email finds you well. I am writing to confirm the details of our upcoming appointment scheduled
for {{{Appointment__c.Appointment_DateTime__c}}} regarding
{{{Appointment__c.PurposeTopic__c}}}.

Appointment Details:
Appointment No : {{{Appointment__c.Name}}},
Student Name : {{{Appointment__c.Student_Name__c}}},
Consultant Name : {{{Appointment__c.Consultant__c}}},
Date & Time : {{{Appointment__c.Appointment_DateTime__c}}},
Purpose : {{{Appointment__c.PurposeTopic__c}}}

I want to assure you that I am looking forward to our meeting and am fully prepared to address any questions or concerns you may have regarding {{{Appointment__c.PurposeTopic__c}}}. Your success and satisfaction are my top priorities, and I am committed to providing you with the guidance and support you need.

If you have any specific topics or questions you would like to discuss during our appointment, please feel free to share them with me in advance. This will help ensure that our time together is as productive and beneficial as possible.

If for any reason you need to reschedule or cancel our appointment, please notify me at your earliest convenience so that we can make alternative arrangements.

Once again, thank you for choosing to work with me on this matter. I am confident that our collaboration will lead to positive outcomes and progress toward your goals.

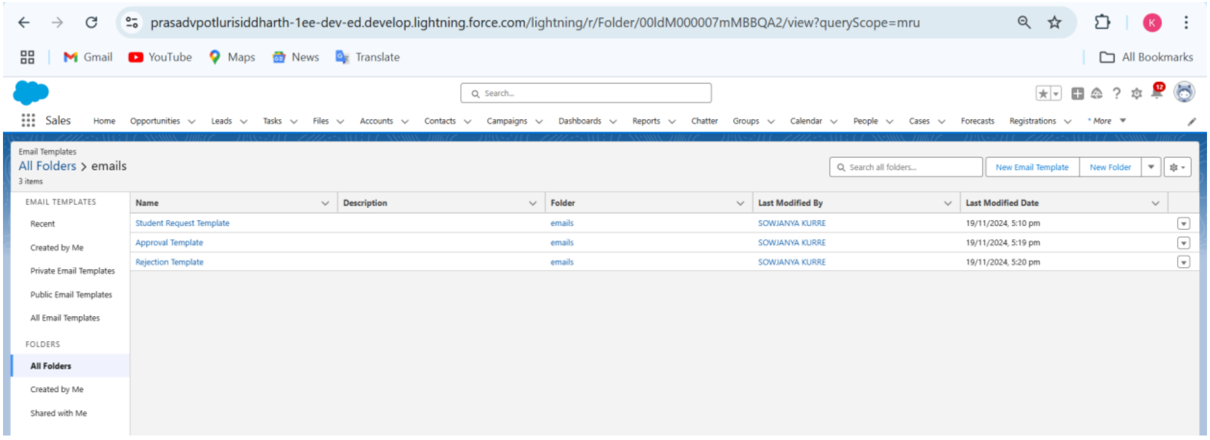
If you have any questions or require further information before our scheduled appointment, please don't hesitate to reach out to me.

Looking forward to our meeting.

Best regards,

{{{Recipient.Name}}},

EduConsultantPro"



Create an Approval Process

1. From Setup, enter Approval in the Quick Find box, and then select Approval Processes.
2. In Manage Approval Processes For, select Appointment.
3. Click Create New Approval Process --> Use Jump Start Wizard.
4. Configure the approval process.
5. Process Name - Appointment Approval, Under Select Approver, Select Manager for the option : “Automatically assign an approver using a standard or custom hierarchy field.”
6. Click next and “Next Automated Approver Determined By” --> Select Manager.
7. From Record Editability Properties --> Click on Administrators OR the currently assigned approver can edit records during the approval process.
8. Save the approval process.
9. Click View Approval Process Detail Page.
10. Under Initial Submission Actions, click Add New --> Field Update, and configure it with these values.

Field	Value
Name	Submitted
Field to Update	Appointment: Status

11. click Add New --> Email Alert, and configure it with these values.
    Description : Submission Email Alert
    Unique Name : Auto Populates
    Email Template : Submission Template
    Recipient Type : Select your Name
12. Repeat the Steps 10 - 11 for Final Approval and Final Rejection actions.

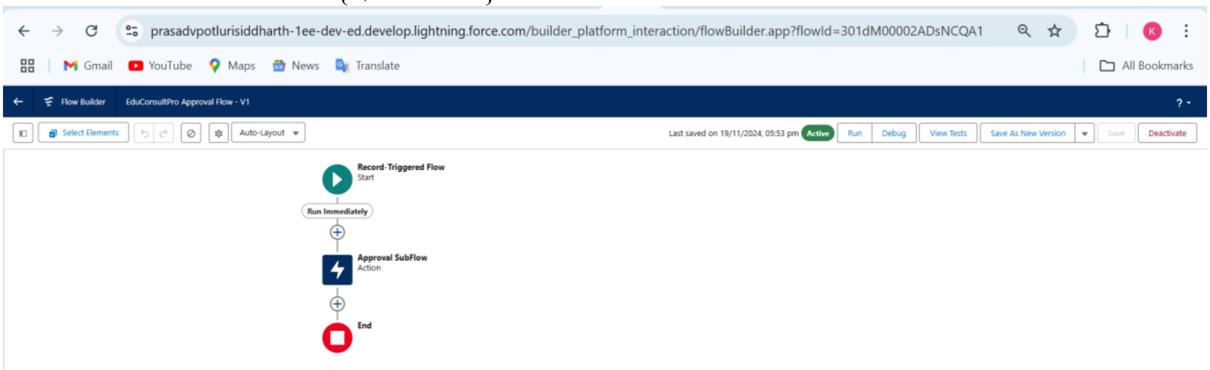
MileStone 5: Create a Record Triggered Flow

Configure the Start Element

1. From Setup, enter Flows in the Quick Find box, then select Flows.
2. Click New Flow.
3. Select Record-Triggered Flow.
4. Click Create. The Configure Start window opens.
5. For Object, select Appointment.
6. For Trigger the Flow When, select A record is created. The flow will look like this:

Add an Action Element

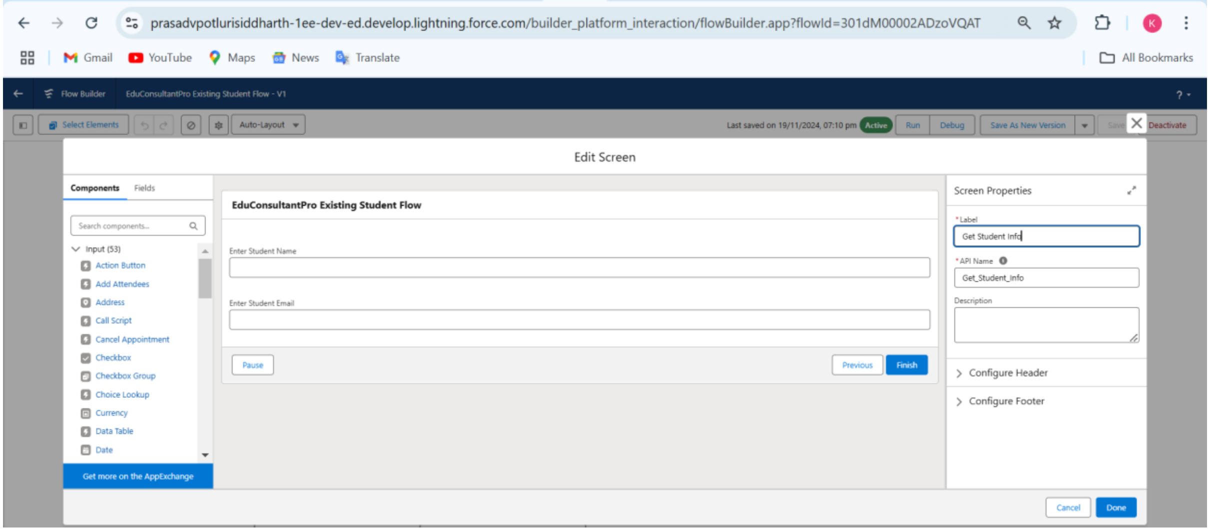
1. Add an Action element after the Start Element and Select the Submit for approval action, label it as “Approval SubFlow”.
2. Set the RecordId to “{!$Record.Id}”.



MileStone 6: Create a ScreenFlow for Existing Student to Book an Appointment

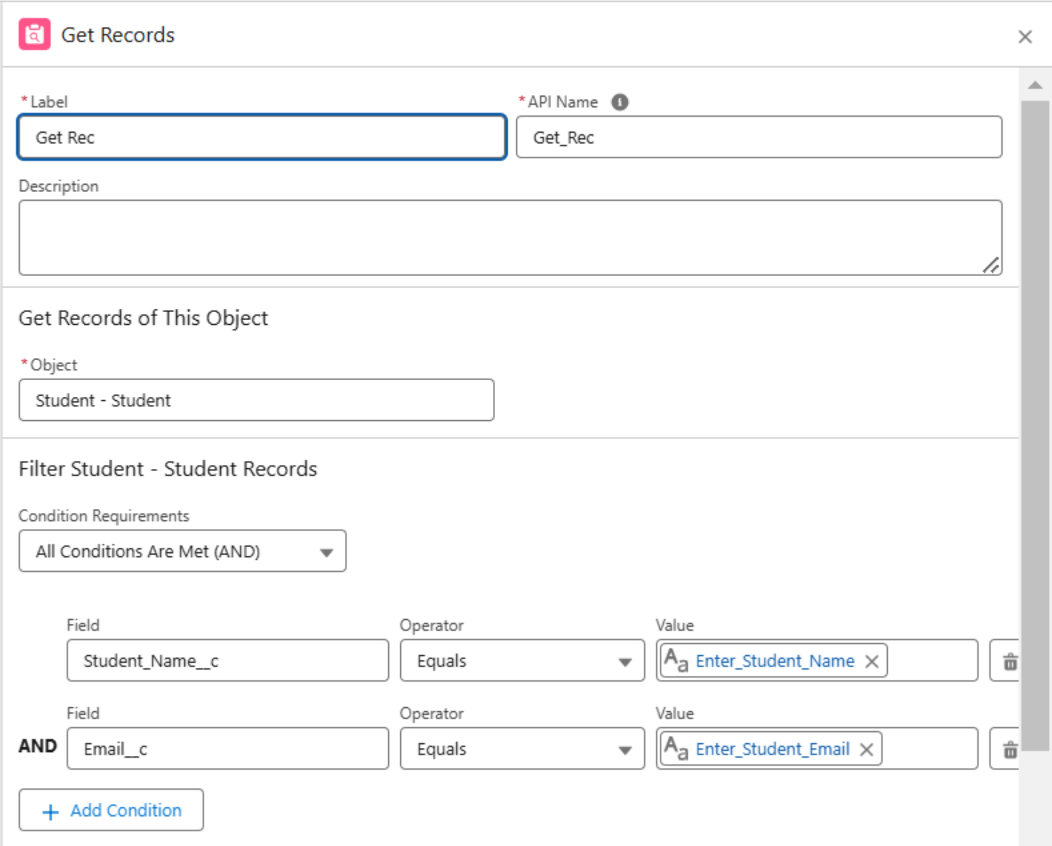
Add Screen Element

1. From Setup, enter Flow Builder in quick find, select new flow -> ScreenFlow.
2. Add a Screen element.
3. In the Screen Properties pane, for Label, enter “Get Student Info”.
4. Add two Text components from the left side panel. Give the Label’s as follows:
   1st Text Component Label : Enter Student Name
   2nd Text Component Label : Enter Student Email



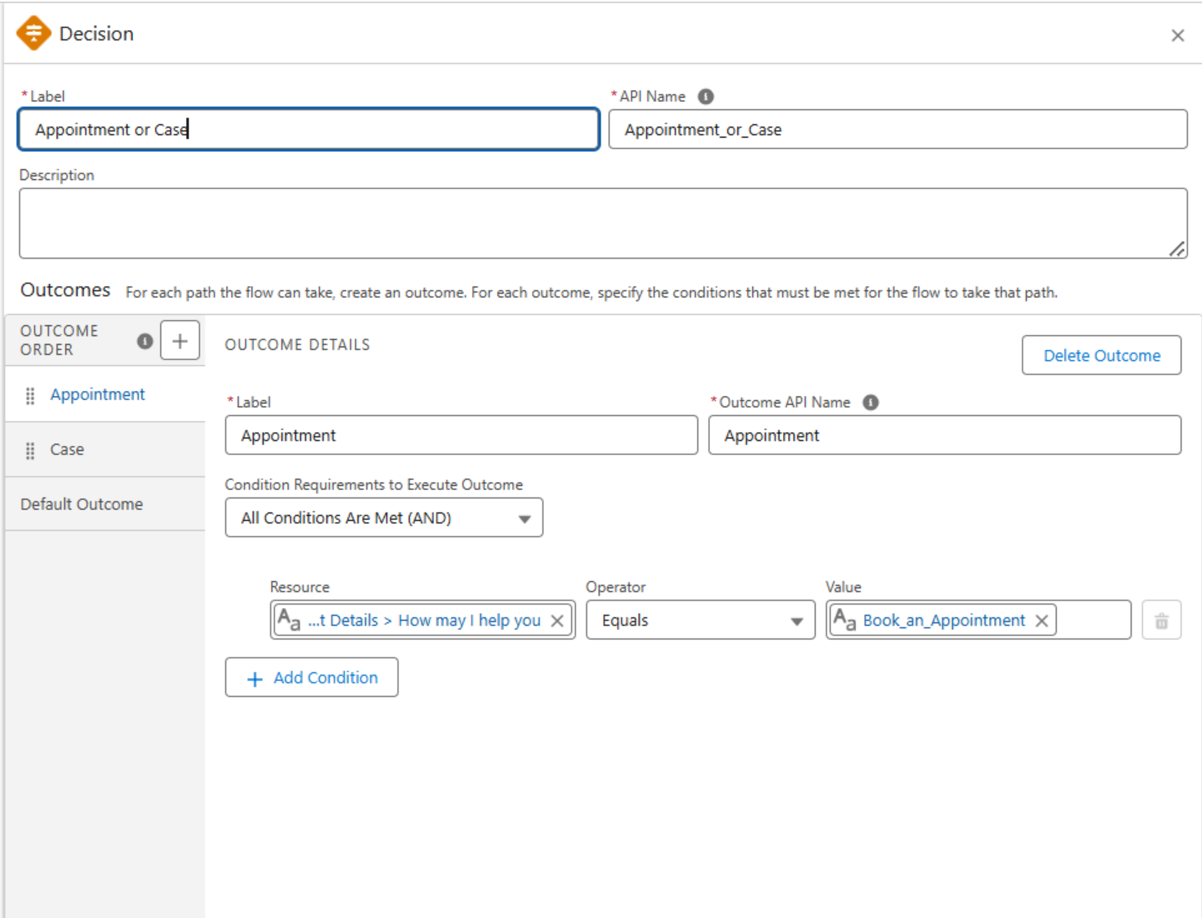
Add GET Record Element

1. Add a GET Record Element after Decision Element, under the IELTS path and label it as “Get Rec”.
2. Select Object : Student
   Condition Requirement : All Conditions are Met(AND)
      Field : Student Name
      Operator : Equals
      Value : {!Enter_Student_Name}
      Field : Email__c
      Operator : Equals
      Value : {!Enter_Student_Email}



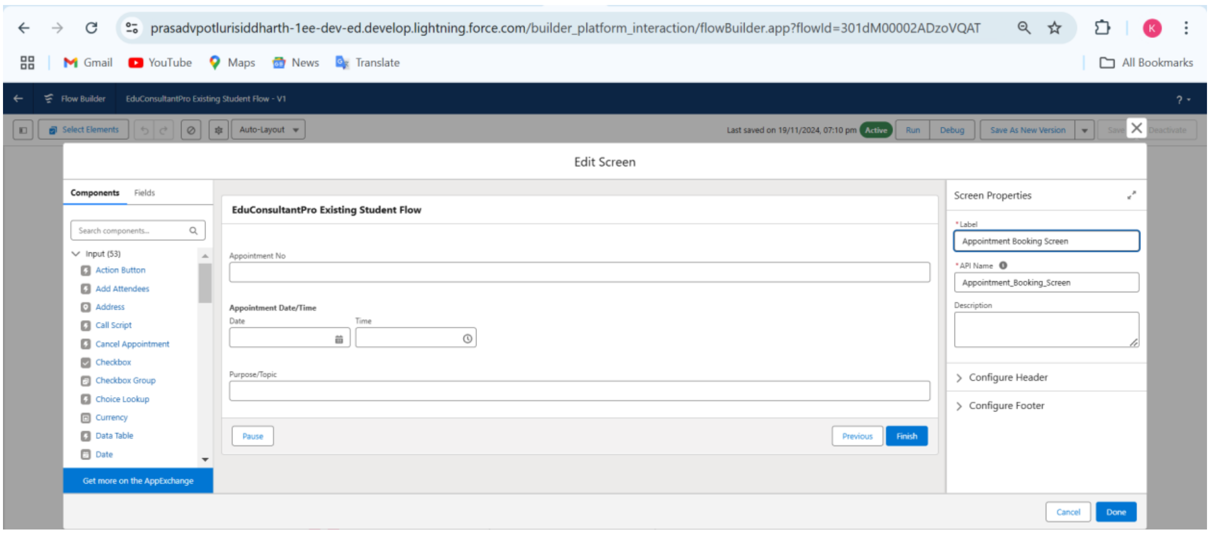
Add Decision Element

1. Add a Decision Element after Select Display Student Details Element, label it as "Appointment or Case".
2. Under outcome label it as "Appointment" and write the condition such as below:
   Resource : {!How_may_I_Help_you}
   Operator : Equals
   Value : {!Book_an_Appointment}
3. Click on the "+" icon and Repeat step 2 for Case options mentioned.



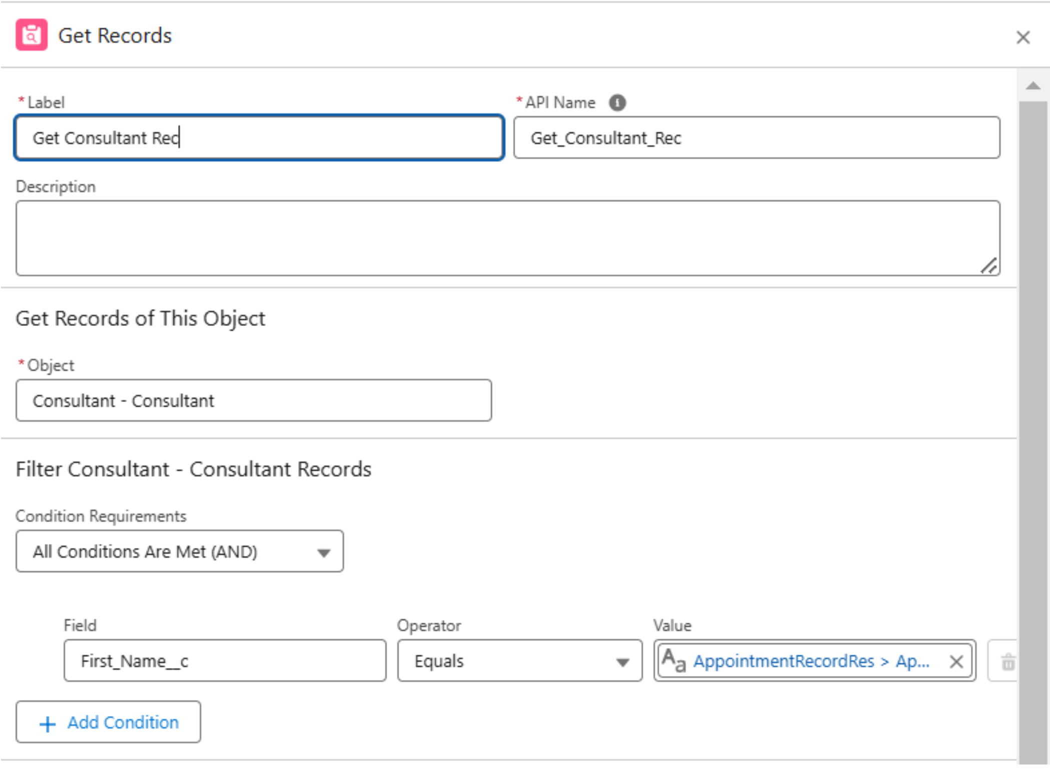
Add Screen Element

1. Add a Screen element after the Decision Element, on the Appointment path and label it as “Appointment Booking Screen".
2. Click on Fields, click on the record variable input and create a new Resource (AppointmentRecordRes) to display all the fields which are in the Appointment object.
3. Drag all the fields which are needed to add on the screen inorder to collect the student information.



Add GET Record Element

1. Add a GET Record Element after Decision Element, under the Appointment path and label it as “Get Consultant Rec”.
2. Select Object : Consultant
   Condition Requirement : All Conditions are Met(AND)
   Field : Name
   Operator : Equals
   Value : {!AppointmentRecordRes.Consultant_Name__c}



Create Appointment Record using Create Records Element

1. Add a Create element after the Get Consultant Rec element and label it as “Create Appointment”.
2. Select “one” under How many records to Create, and select “Use separate resources, and literal values” under How to Set the record fields.
3. Select Object : Appointment
   Field : Appointment_DateTime__c
   Value : {!AppointmentRecordRes.Appointment_DateTime__c}
   Field : Consultant__c
   Value : {!Get_Consultant_Rec.Id}
   Field : Notes__c
   Value : {!AppointmentRecordRes.Notes__c}
   Field : PurposeTopic__c
   Value : {!AppointmentRecordRes.PurposeTopic__c}
   Field : Student_Name__c
   Value : {!Get_Rec.Id}

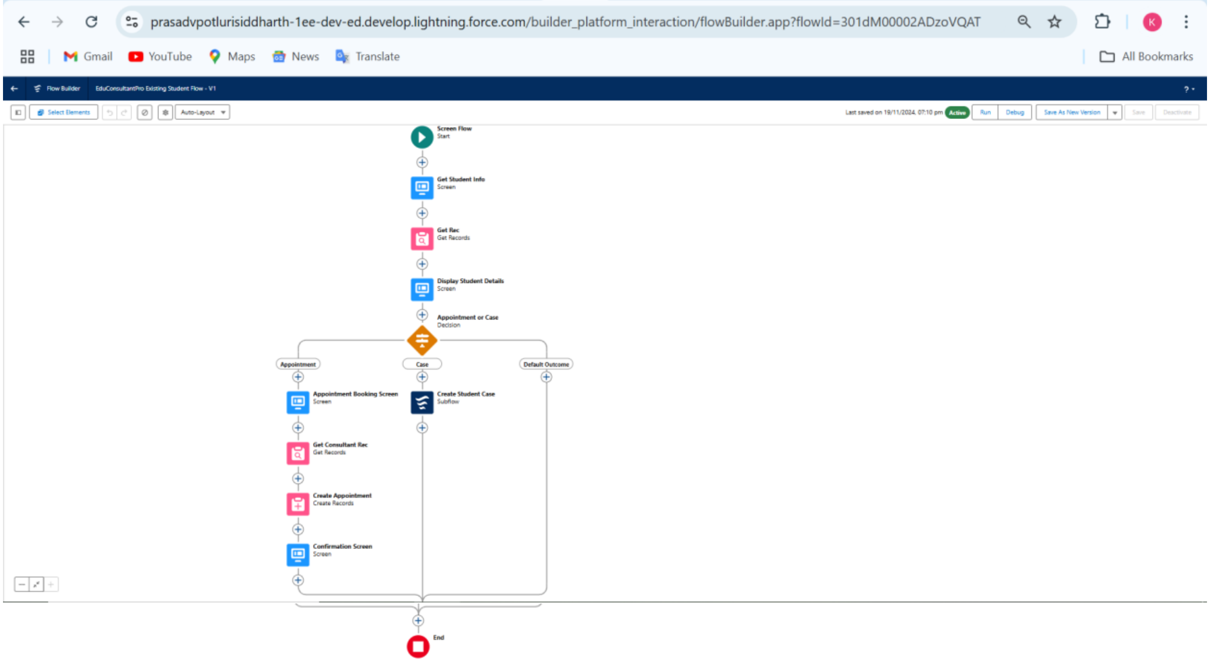
Add Screen Element

1. Add a Screen Element after the Send Email to Student Action Element, label it as “Confirmation Screen”.

2. From the left side panel search for the Display text component and drag it to the main panel, label it as “Appointment_Confirmation”.
3. Paste the below in the Resource picker box.
   Consultant Name : {!Get_Consultant_Rec.Name},
   Date & Time : {!AppointmentRecordRes.Appointment_DateTime__c},
   Notes : {!AppointmentRecordRes.Notes__c},

Add an SubFlow Element

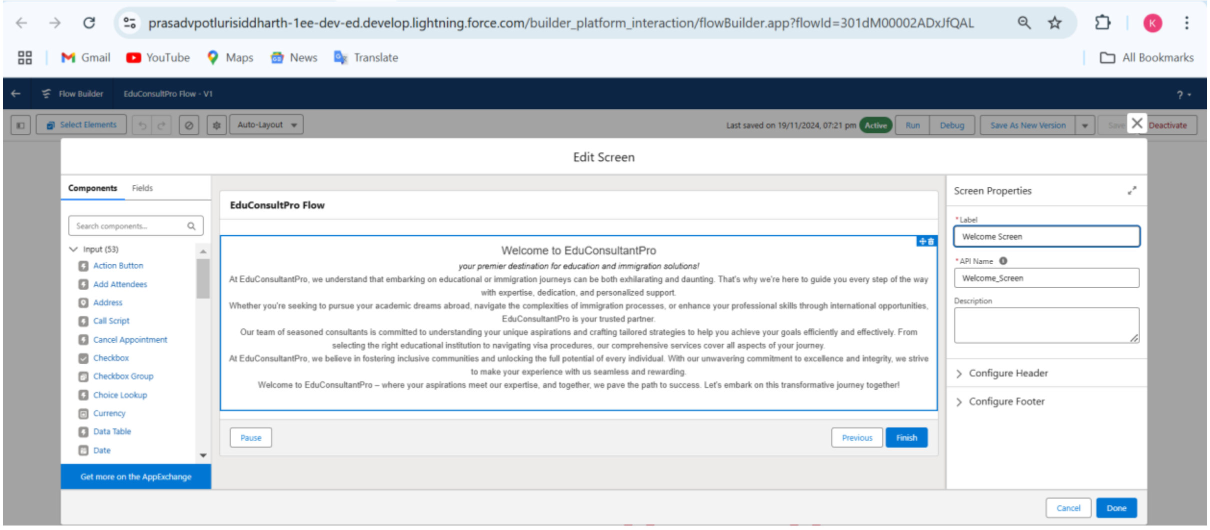
1. Add a subflow element after the Decision Element, on the Case path and search and Select for “Create a Case”, label it as “Create Student Case”.
2. Save the flow and label it as “EduConsultantPro Existing Student Flow”, you can use the below image for reference.



Milestone 7:Create a ScreenFlow to Combine all the flows at one place

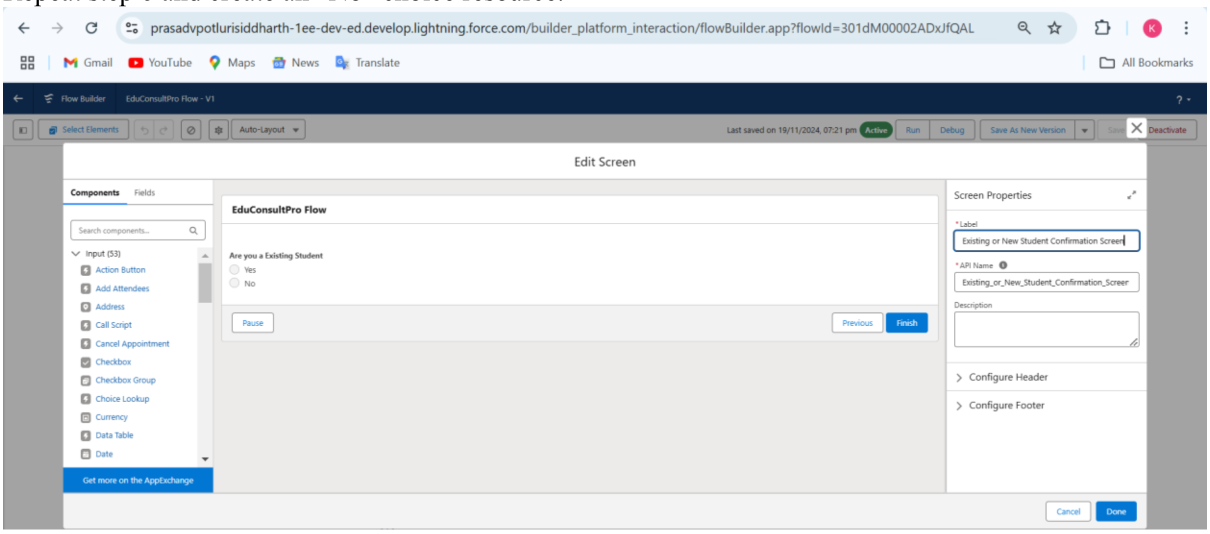
Add Screen Element

1. Add a Screen Element and label it as Welcome Screen.
2. From the left side panel search for the Display text component and drag it to the main panel, label it as “SuccessMessage”.
3. Paste the below in the Resource picker box.



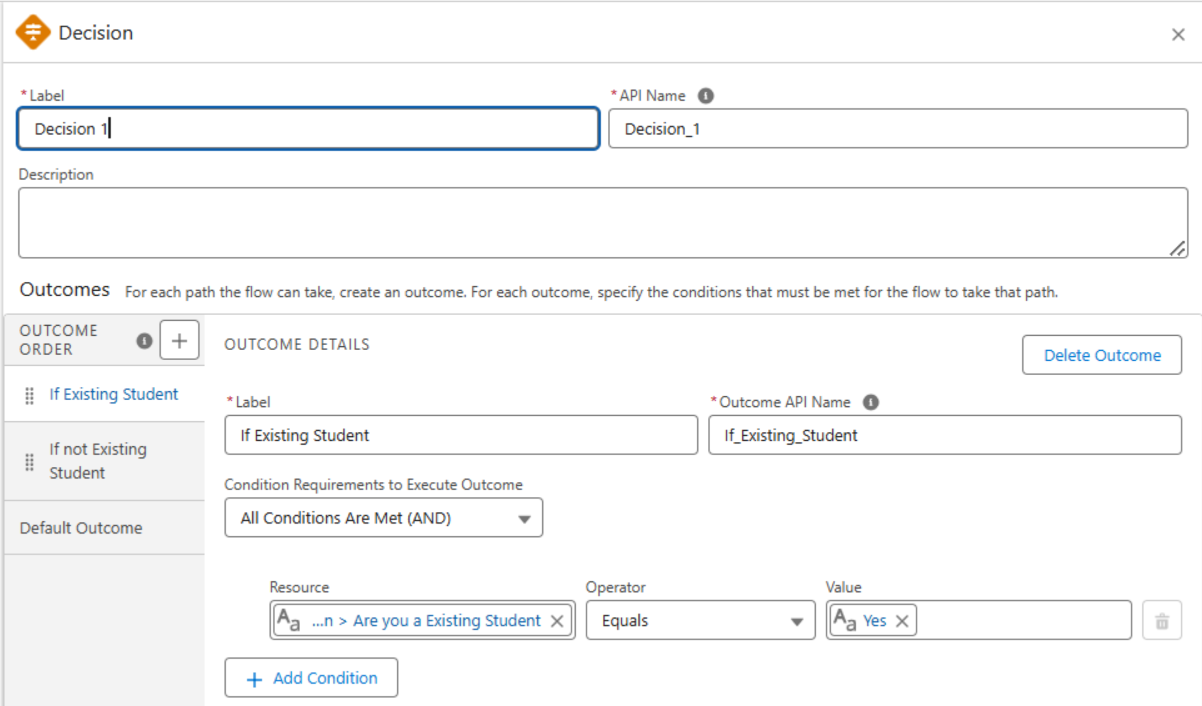
Add Screen Element

1. Add a Screen Element after the Welcome Screen Element, label it as “Existing or New Student Confirmation Screen”.
2. Add a radio button component from the left side panel,
   label : Are you a Existing Student
3. Click on Add Choice --> type “Yes” in the input field --> click Create Yes choice.
4. Repeat step 6 and create an “No” choice resource.



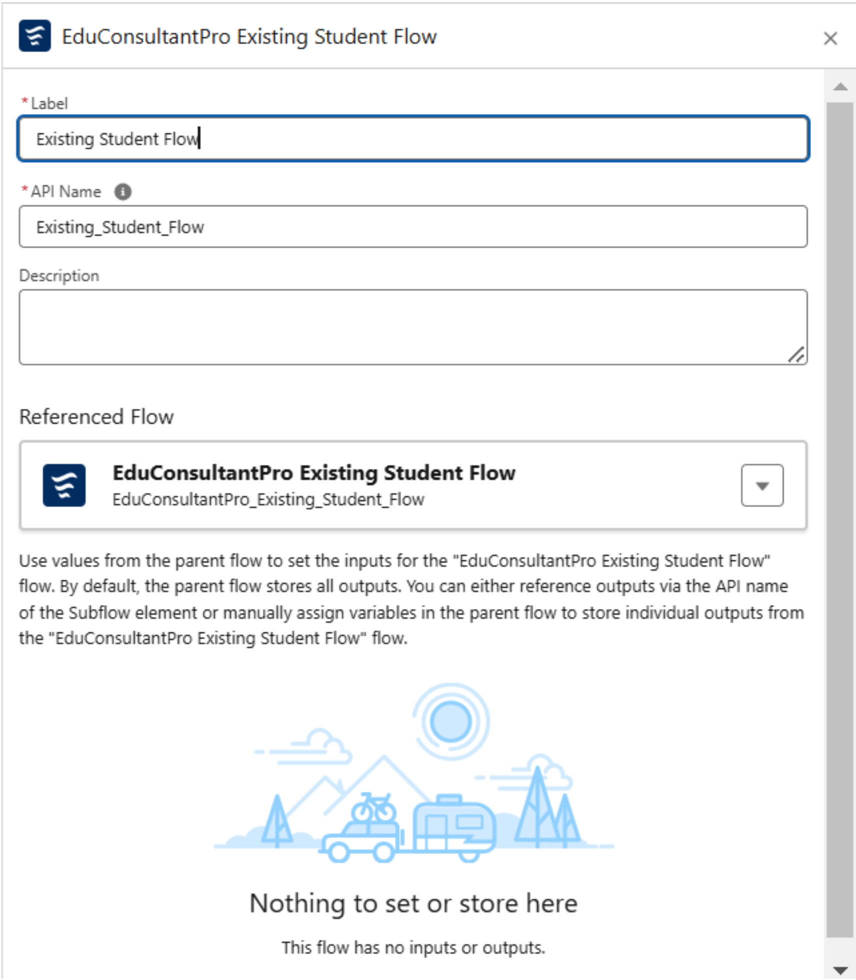
Add Decision Element

1. Add a Decision Element after Existing or New Student Confirmation Screen Element, label it as “Decision 1”.
2. Under outcome label it as “If Existing Student” and write the condition such as below:
   Resource : {!Are_you_a_Existing_Student}
   Operator : Equals
   Value : {!Yes}
3. Click on the “+” icon and Repeat step 2 for No options mentioned.



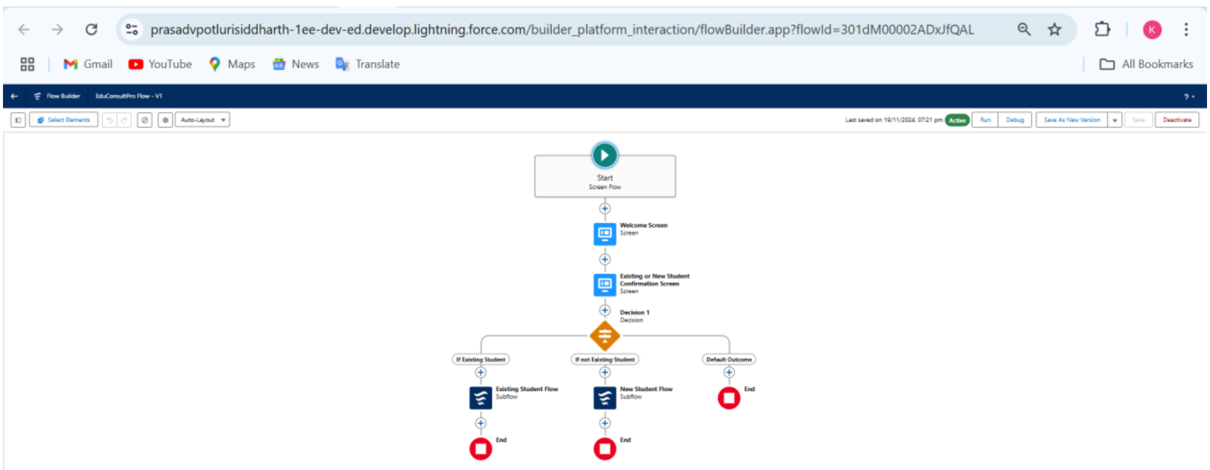
Add an SubFlow Element

1. Add a subflow element after the Decision 1 Element on the if Existing Student path and search and Select for “EduConsultantPro Existing Student Flow ", label it as “Existing Student Flow”.
2. Save the flow and label it as “EduConsultantPro Existing Student Flow”.



Add an SubFlow Element

1. Add a subflow element after the Decision 1 Element on the if Not an Existing Student path and search and Select for “EduConsultantPro Student Flow ", label it as “New Student Flow”.
2. Save the flow and label it as “EduConsultantPro Existing Student Flow”.



APEX AND TRIGGERS

Apex can be invoked by using triggers. Apex triggers enable you to perform custom actions before or after changes to Salesforce records, such as insertions, updates, or deletions. A trigger is Apex code that executes before or after the following types of operations:

● insert

● update

● delete

● merge

● upsert

● undelete

For example, you can have a trigger run before an object's records are inserted into the database, after records have been deleted, or even after a record is restored from the Recycle Bin. You can define triggers for top-level standard objects that support triggers, such as Contact or an Account, some standard child objects, such as a CaseComment, and custom objects. To define a trigger, from the object management settings for the object who triggers you want to access, go to Triggers. There are primarily two types of Apex Triggers: Before Trigger: This type of trigger in Salesforce is used either to update or validate the values of a record before they can be saved into the database. So, basically, the before trigger validates the record first and then saves it. Some criteria or code can be set to check data before it gets ready to be inserted into the database. After Trigger: This type of trigger in Salesforce is used to access the field values set by t system and affect any change in the record. In other words, the after trigger makes changes to the value from the data inserted in some other record.

Trigger Handler :

- The ValidateAppointmentTime Trigger validates that appointment times are within working hours (9:00 AM to 6:00 PM) and ensures there are no overlapping appointments for the same consultant.

- If a conflict is detected, it prevents the appointment from being saved and triggers the Apex Class to reschedule the appointment to the next available slot.

How to create a new trigger :

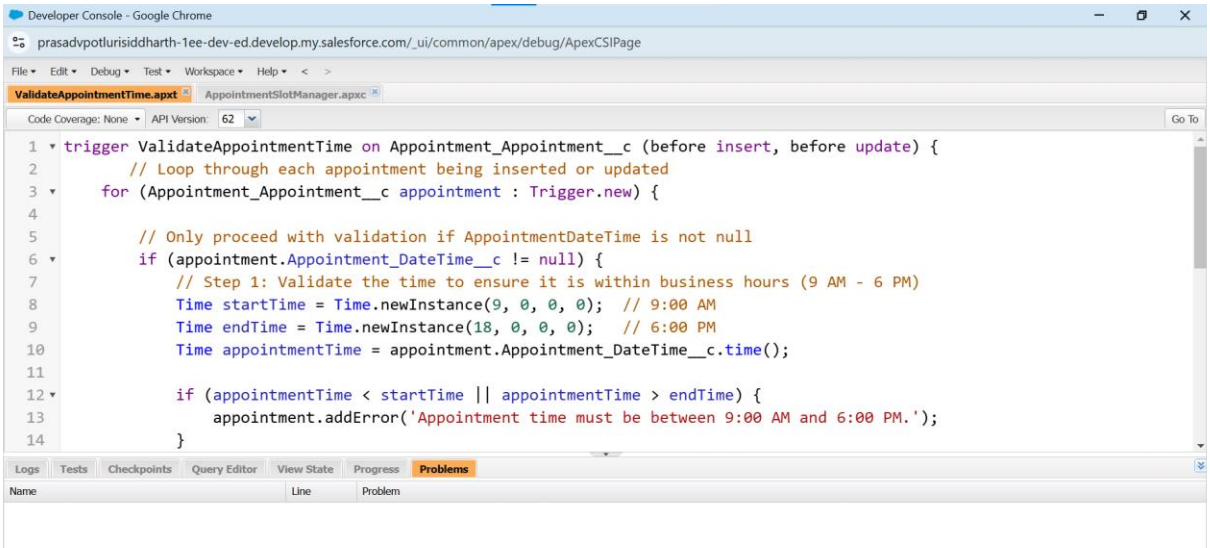
1. Click on developer console and you will be navigated to a new console window.

2. Click on File menu in the tool bar, and click on new→Trigger.

3. Enter the trigger name and the object to be triggered.

4. Name : ValidateAppointmentTime

5. sObject : Appointment__Appointment_c



Code:

```
trigger ValidateAppointmentTime on Appointment_Appointment__c (before insert, before
update) {

    for (Appointment_Appointment__c appointment : Trigger.new) {

        if (appointment.Appointment_DateTime__c != null) {

            Time startTime = Time.newInstance(9, 0, 0, 0);  // 9:00 AM

            Time endTime = Time.newInstance(18, 0, 0, 0);   // 6:00 PM

            Time appointmentTime = appointment.Appointment_DateTime__c.time();

             if (appointmentTime < startTime || appointmentTime > endTime) {

                appointment.addError('Appointment time must be between 9:00 AM and 6:00
PM.');

            }

            List<Appointment_Appointment__c> overlappingAppointments = [

                SELECT Id
```

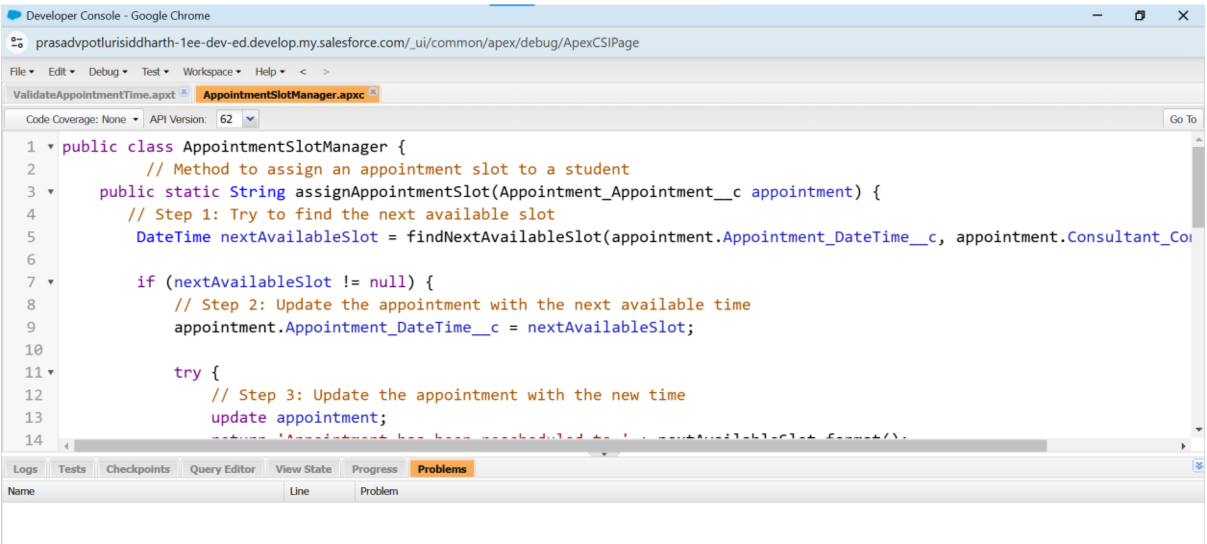
```
            FROM Appointment_Appointment__c
            WHERE Appointment_DateTime__c = :appointment.Appointment_DateTime__c
            AND Consultant_Consultant__c = :appointment.Consultant_Consultant__c
            AND Id != :appointment.Id
        ];

        if (!overlappingAppointments.isEmpty()) {
            String result = AppointmentSlotManager.assignAppointmentSlot(appointment);
            if (result != null) {
                appointment.addError(result); // Add error if rescheduling fails
            }
        }
    }
  }
}
```

Apex Handler

The AppointmentSlotManager Apex Class dynamically identifies the next available appointment slot for a consultant if there is a conflict and reschedules the conflicting appointment.

It updates the appointment record with the new time and ensures the consultant's schedule remains within capacity limits.



Code:

```
public class AppointmentSlotManager {

 public static String assignAppointmentSlot(Appointment_Appointment__c appointment) {

        DateTime nextAvailableSlot =
findNextAvailableSlot(appointment.Appointment_DateTime__c,
appointment.Consultant_Consultant__c);

           if (nextAvailableSlot != null) {

            appointment.Appointment_DateTime__c = nextAvailableSlot;

                try {

                        update appointment;

                return 'Appointment has been rescheduled to ' + nextAvailableSlot.format();

            } catch (Exception e) {

                return 'Error occurred while rescheduling the appointment: ' + e.getMessage();

            }

        } else {

            return 'No available slots for rescheduling.';

        }

    }
```

```
    public static DateTime findNextAvailableSlot(DateTime appointmentDateTime, Id consultantId) {

        List<Appointment_Appointment__c> nextAvailableAppointments = [

            SELECT Appointment_DateTime__c

            FROM Appointment_Appointment__c

            WHERE Consultant_Consultant__c = :consultantId

            AND Appointment_DateTime__c > :appointmentDateTime

            ORDER BY Appointment_DateTime__c ASC

            LIMIT 1

        ];

        if (!nextAvailableAppointments.isEmpty()) {

            return nextAvailableAppointments[0].Appointment_DateTime__c;

        } else {

                return null;

        }

    }

}
```

MileStone 8:Create a lightning app page

1. From Setup, enter App Builder in the Quick Find box, then click Lightning App Builder.
2. Click New, select Home Page, then click Next.
3. Step through the wizard and name the page “EduConsultPro Home Page”, select the Standard Home Page template, and then click Done.
4. Drag the Flow component to the top-right region.
5. Search for the “EduConsultantPro Flow” and click Save.