

A CRM Application to Manage the Services offered by an Institution

BY

G. Ajith

gopuajith319@gmail.com

Project Abstract

In today's competitive environment, effective customer relationship management is vital for any institution aiming to provide superior service. This project focuses on developing a customized CRM application using Salesforce to manage and enhance the services offered by an institution. Salesforce, a leading CRM platform, offers robust tools and capabilities to streamline service management and improve customer satisfaction.

The CRM application will serve as a centralized platform where institutions can track and manage all customer interactions, from initial contact to post-service follow-ups. It will allow institutions to store and access detailed customer information, including contact details, service history, and preferences. This centralized database will enable staff to provide personalized and efficient services, leading to improved customer experiences. By implementing this CRM application, institutions can expect to see significant improvements in service management, customer satisfaction, and overall operational efficiency. The application will provide a structured approach to handling customer interactions, reducing the chances of miscommunication and missed opportunities. Additionally, the insights gained from the application's analytics will help institutions identify areas for improvement and optimize their services.

INDEX Page

Topics	Page No
Cover page	1
Project Abstract	2
Introduction	5
TASK-1 Create Objects from Spreadsheet	6-12
1.1 Create Course object	6
1.2 Create Remaining object	7
1.3 Create Relationship among the objects	8
1.4 Configure the Case Object	11
1.5 Create a Lightning App	12
TASK-2 Create a Screen Flow for Student Admission Application process	12-26
2.1 Add Screen Element	12
2.2 Create Student Record using Create Element	13
2.3 Add Screen Element	14
2.4 Add Decision Element	15
2.5 Add GET Record Element	16
2.6 Create Registration Record using Create Records Element	18
2.7 Create Email Text Template Variables for email body and subject	21
2.8 Add an Action Element	23
2.9 Add Screen Element	25
TASK-3 Create Users	26-27
3.1 User	26
3.2 Configure the User Settings	27
TASK-4 Create an Approval Process for Property Object	27-29
4.1 Create an Email Template	27
4.2 Create an Approval Process	28
TASK-5 Create a Record Triggered Flow	29-31
5.1 Configure the Start Element	29
5.2 Add an Action Element	30
TASK-6 Create a ScreenFlow for Existing Student to Book an Appointment	32-39
6.1 Add Screen Element	32
6.2 Add GET Record Element	32
6.3 Add Decision Element	33
6.4 Add Screen Element	34
6.5 Add GET Record Element	34
6.6 Create Appointment Record using Create Records Element	35
6.7 Add Screen Element	37

6.8 Add an SubFlow Element	38
TASK-7 Create a ScreenFlow to Combine all the flows at one place	39-44
7.1 Add Screen Element	39
7.2 Add Screen Element	40
7.3 Add Decision Element	41
7.4 Add an SubFlow Element	42
7.5 Add an SubFlow Element	43
TASK-8 Create a lightning app page	44-45
8.1 Create a lightning app page	44

Introduction

A CRM Application to Manage the Services offered by an Institution

EduConsultPro Institute is a leading educational institution offering a variety of courses and programs in diverse fields. With a growing number of prospective students seeking admission each year, the institute faces challenges in managing the admission process, students enquiry, and expert consulting services efficiently. To address these challenges, EduConsultPro Institute decides to leverage Salesforce CRM to streamline the admission process and enhance the overall experience for both students and admissions staff.

The use case focuses on the admission process for prospective students interested in enrolling in courses and programs offered by EduConsultPro Institute. The goal is to provide a seamless and transparent experience for students while enabling admissions staff to efficiently review and process admission applications, students enquiry and case management.

In today's competitive business environment, maintaining a strong relationship with customers is crucial for the success of any institution. A Customer Relationship Management (CRM) application is designed to streamline and enhance interactions between an institution and its customers. This project aims to develop a CRM application specifically tailored to manage the services offered by the institution, ensuring efficient service delivery and improving customer satisfaction.

The primary objective of the CRM application is to provide a comprehensive platform that integrates all aspects of customer interaction and service management. This includes tracking service requests, managing customer information, scheduling services, and providing insights through data analysis. By centralizing these functions, the CRM application will enable the institution to deliver a more personalized and responsive customer experience.

Task-1 Create Objects from Spreadsheet

Directly Creating Objects from Spreadsheet in Salesforce

1.1 Create Course object

- Go to your object manager and click on create object from spreadsheet
- Click on the link to get the spreadsheet, [Course](#).
- After downloading, upload the file, map the fields and upload to create an object.

The screenshot shows the Salesforce Object Manager page. At the top, there's a navigation bar with 'Setup', 'Home', and 'Object Manager'. The main area displays a table of objects with columns for 'LABEL', 'API NAME', 'TYPE', 'DESCRIPTION', and 'LAST MODIFIED'. The 'LAST MODIFIED' column shows 'Custom Object from Spreadsheet' for the last entry. Below the table, a message says 'Create a custom object from a spreadsheet'. Underneath, a section titled 'Select a spreadsheet' asks 'Select a source for your new object data.' It provides three options: 'Upload' (with a dashed box for dropping files), 'Google Sheet' (with a Google Sheets icon), and 'Office 365 or Drive' (with a cloud icon).

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED
Account	Account	Standard Object		
Activity	Activity	Standard Object		
Alternative Payment Method	AlternativePaymentMethod	Standard Object		
API Anomaly Event Store	ApiAnomalyEventStore	Standard Object		Custom Object from Spreadsheet

Create a custom object from a spreadsheet

Select a spreadsheet

Select a source for your new object data.

Upload .xlsx or .csv

or Drop File Here

Upload

Google Sheet

Office 365 or Drive

Create a custom object from a spreadsheet

Nice Work!



Now you can add your object to a Lightning app. You might need to refresh the object list to see it.

Import Overview

Object Created
Course

Fields Detected
5

Rows Detected
5

Fields Created
5

Rows Imported
5

[Import Another Object](#)

1.2 Create Remaining objects

- Follow the steps which we have followed for course object creation.
- Use the following sheets for remaining objects.
- [Consultant](#)
- [Student](#)
- [Appointment](#)

Create a custom object from a spreadsheet

Nice Work!



Now you can add your object to a Lightning app. You might need to refresh the object list to see it.

Import Overview

Object Created
Consultant - Consultant

Fields Detected
6

Rows Detected
3

Fields Created
6

Rows Imported
3

[Import Another Object](#)

Create a custom object from a spreadsheet

Nice Work!



Now you can add your object to a Lightning app. You might need to refresh the object list to see it.

Import Overview

Object Created	Student - Student
Fields Detected	12
Rows Detected	2
Fields Created	12
Rows Imported	2

[Import Another Object](#)

Create a custom object from a spreadsheet

Nice Work!



Now you can add your object to a Lightning app. You might need to refresh the object list to see it.

Import Overview

Object Created	Appointment - Appointment
Fields Detected	5
Rows Detected	0
Fields Created	5
Rows Imported	0

[Import Another Object](#)

1.3 Create Relationship among the objects

- Create lookup between appointment and student, appointment and consultant.
- Create an object to store the information student and course details with the name Registration.
- Also create a lookup between student and case to store the student queries for immigration or visa application.
- The data model should be similar to the below Data Model with fields & relationships:

SETUP > OBJECT MANAGER

Appointment

Fields & Relationships 11 items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment	Name	Text(80)		✓
Appointment Date/Time	Appointment_DateTime__c	Date/Time		
Appointment No	Appointment_No__c	Auto Number		
Consultant	Consultant__c	Lookup(Consultant)		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedBy	Lookup(User)		
Notes	Notes__c	Long Text Area(256)		
Owner	OwnerId	Lookup(User,Group)		✓
Purpose/Topic	PurposeTopic__c	Text(255)		
Status	Status__c	Picklist		

Appointment Custom Field
Student Name

[Back to Appointment](#) [Help](#)

[Validation Rules \[0\]](#)

Custom Field Definition Detail [Edit](#) [Set Field-Level Security](#) [View Field Accessibility](#) [Where is this used?](#)

Field Information

Field Label	Student Name	Object Name	Appointment
Field Name	Student_Name	Data Type	Lookup
API Name	Student_Name__c		
Description			
Help Text			
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			
Created By	2111CS010024 - GOPU AJITH, 7/29/2024, 9:08 AM	Modified By	2111CS010024 - GOPU AJITH, 7/30/2024, 4:52 AM

Lookup Options

Related To	Student	Child Relationship Name	Appointments
Related List Label	Appointments		
Required	<input type="checkbox"/>		
What to do if the lookup record is deleted?	Clear the value of this field.		

Appointment Custom Field
Consultant

[Back to Appointment](#)

[Validation Rules \[0\]](#)

Custom Field Definition Detail [Edit](#) [Set Field-Level Security](#) [View Field Accessibility](#) [Where is this used?](#)

Field Information

Field Label	Consultant	Object Name	Appointment
Field Name	Consultant	Data Type	Lookup
API Name	Consultant__c		
Description			
Help Text			
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			
Created By	2111CS010024 - GOPU AJITH, 7/30/2024, 4:48 AM	Modified By	2111CS010024 - GOPU AJITH, 7/30/2024, 4:48 AM

Lookup Options

Related To	Consultant	Child Relationship Name	Appointments
Related List Label	Appointments		
Required	<input type="checkbox"/>		
What to do if the lookup record is deleted?	Clear the value of this field.		

SETUP > OBJECT MANAGER

Registration

Fields & Relationships

7 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXE
Course Name	Course_Name__c	Lookup(Course)		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Registration Name	Name	Text(80)		✓
Registration No	Registration_No__c	Number(18, 0)		
Student Name	Student_Name__c	Lookup(Student)		✓

Case Custom Field
Student_LookUp

[Back to Case Fields](#)

[Validation Rules \[0\]](#)

Custom Field Definition Detail

[Edit](#) | [Set Field-Level Security](#) | [View Field Accessibility](#) | [Where is this used?](#)

Field Information

Field Label	Student_LookUp	Object Name	Case
Field Name	Student_LookUp	Data Type	Lookup
API Name	Student_LookUp__c		
Description			
Help Text			
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			
Created By	2111CS010024 - GOPU AJITH, 7/30/2024, 6:51 AM	Modified By	2111CS010024 - GOPU AJITH, 7/30/2024, 6:51 AM

Lookup Options

Related To	Student	Child Relationship Name	Cases
Related List Label	Cases		
Required	<input type="checkbox"/>		

What to do if the lookup record is deleted? [Clear the value of this field.](#)

- Go to home >quick find type schema builder

Setup Home Object Manager

Schema Builder

[Close](#) | [Auto-Layout](#) | [View Options ▾](#)

Elements Objects

Select objects to display on the builder.

Select from [All Objects](#) [Quick Find...](#) [Select All](#) | [Clear All](#)

The diagram illustrates the relationships between various Salesforce objects:

- Registration** has fields: Course Name, Created By, Last Modified By, Owner, Registration Name, Registration No, and Student Name.
- Student** has fields: Address, City, Created Date, Date of Birth, Description, First Name, Gender, Last Modified By, Last Name, Owner, Phone, Qualification, Student Name, Student Record, University Name, and Year of Passing.
- Course** has fields: Course, Course Name, Description, End Date, Instructor, Last Modified By, Owner, and Start Date.
- Appointment** has fields: Appointment, Appointment Date/Time, Appointment No, Consultant, Last Modified By, Notes, Owner, Purpose Topic, Status, and Student Name.
- Case** has fields: Account Name, Asset, Backup Agent, Business Hours, Case Number, Case Origin, Case Owner, Case Reason, Case When Created, Contact Email, Contact Fax, Contact Mobile, Contact Name, Contact Phone, Created By, Date/Time Closed, Date/Time Opened, Description, Engineering Rev Number, Engineering Rev Type, Endorsement Name, Endorsement Process End Date, Endorsement Process Start Date, and Show More Fields.
- Consultant** has fields: Address, Consultant, Created By, Email, Expertise, First Name, Last Modified By, Last Name, Owner, and Phone.

1.4 Configure the Case Object

- Go to object manager, edit case object.
- Select the “Type” field and add the values in it.
Immigration
Visa Application
- Now Select the “Status” field and add the values in it.
Open
In-progress

Action	Values	API Name	Closed	Default	Chart Colors	Modified By
Edit Del Deactivate	New	New	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 5/9/2024, 4:46 AM
Edit Del Deactivate	Working	Working	<input type="checkbox"/>	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 5/9/2024, 4:46 AM
Edit Del Deactivate	Escalated	Escalated	<input type="checkbox"/>	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 5/9/2024, 4:46 AM
Edit Del Deactivate	Closed	Closed	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 5/9/2024, 4:46 AM
Edit Del Deactivate	Open	Open	<input type="checkbox"/>	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 7/30/2024, 10:04 AM
Edit Del Deactivate	In-progress	In-progress	<input type="checkbox"/>	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 7/30/2024, 10:04 AM

Action	Values	API Name	Default	Chart Colors	Modified By
Edit Del Deactivate	Mechanical	Mechanical	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 5/9/2024, 4:46 AM
Edit Del Deactivate	Electrical	Electrical	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 5/9/2024, 4:46 AM
Edit Del Deactivate	Electronic	Electronic	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 5/9/2024, 4:46 AM
Edit Del Deactivate	Structural	Structural	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 5/9/2024, 4:46 AM
Edit Del Deactivate	Other	Other	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 5/9/2024, 4:46 AM
Edit Del Deactivate	Immigration	Immigration	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 7/30/2024, 9:57 AM
Edit Del Deactivate	Visa Application	Visa Application	<input type="checkbox"/>	Assigned dynamically	2111CS010024 - GOPU AJITH 7/30/2024, 9:57 AM

1.5 Create a Lightning App

- Go to Setup, search for the App Manager in quick find
- Click on New Lightning App
- Give app name as “EduConsultPro”, click Next, Next, Next
- Add Home, Students, Courses, Consultants, Appointments, Registrations, and Cases from the Available Items to Selected Items.
- Add “System Administrator” profile from Available Profiles to Selected Profiles, click Save & Finish.

The screenshot shows the Salesforce Lightning Experience App Manager. At the top, there's a header with 'Object Manager' and 'SETUP'. Below it is a sub-header 'Lightning Experience App Manager' with buttons for 'New Lightning App' and 'New Connected App'. The main area displays a table of 30 items, sorted by App Name. The columns include App Name, Developer Name, Description, Last Modified, App Type, and Vi... (View). The table lists various standard Salesforce tools like Ant Migration Tool, App Launcher, Automation, Bolt Solutions, etc., along with custom apps like EduConsultPro and Force.com IDE.

30 items • Sorted by App Name • Filtered by All appmenuitems - TabSet Type						
App Name ↑	Developer Name	Description	Last Modified ...	App Type	Vi... ▾	⋮
3 Ant Migration Tool	Forcecom_Migration_Tool	The Force.com Migration Tool is a Java/Ant-based command-line utility...	5/9/2024, 4:46 AM	Connected (Managed)	✓	4
4 App Launcher	AppLauncher	App Launcher tabs	5/9/2024, 4:46 AM	Classic	✓	▼
5 Automation	FlowsApp	Automate business processes and repetitive tasks.	6/15/2024, 2:16 PM	Lightning	✓	▼
6 Bolt Solutions	LightningBolt	Discover and manage business solutions designed for your industry.	5/9/2024, 4:46 AM	Lightning	✓	▼
7 Community	Community	Salesforce CRM Communities	5/9/2024, 4:46 AM	Classic	✓	▼
8 Content	Content	Salesforce CRM Content	5/9/2024, 4:46 AM	Classic	✓	▼
9 Data Manager	DataManager	Use Data Manager to view limits, monitor usage, and manage recipes.	5/9/2024, 4:46 AM	Lightning	✓	▼
10 Dataloader Bulk	Dataloader_Bulk	The Data Loader is an easy to use graphical tool that helps you to get y...	5/9/2024, 4:46 AM	Connected (Managed)	▼	4
11 Dataloader Partner	Dataloader_Partner	The Data Loader is an easy to use graphical tool that helps you to get y...	5/9/2024, 4:46 AM	Connected (Managed)	▼	4
12 Digital Experiences	SalesforceCMS	Manage content and media for all of your sites.	5/9/2024, 4:46 AM	Lightning	✓	▼
13 EduConsultPro	EduConsultPro	The Force.com IDE is a powerful client application for creating, modifyi...	7/30/2024, 10:23 A...	Lightning	✓	▼
14 Force.com IDE	Forcecom_IDE	The Force.com IDE is a powerful client application for creating, modifyi...	5/9/2024, 4:46 AM	Connected (Managed)	▼	4

TASK-2 Create a ScreenFlow for Student Admission Application process

2.1 Add Screen Element

- From Setup, enter Flow Builder in quick find, select new flow --> ScreenFlow.
- Add a Screen element.
- In the Screen Properties pane, for Label, enter “Student Info”.
- 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.

Edit Screen

EduConsultPro Student Flow

First Name
Last Name
City
Address
Date of Birth
Email
Gender
Phone

Components Fields

Input (30)

- Action Button (Beta)
- Address
- Call Script
- Checkbox
- Checkbox Group
- Choice Lookup
- Currency
- Data Table
- Date
- Date & Time
- Dependent Picklists
- Display Image
- Email
- Enhanced Message
- File Upload
- Long Text Area
- Lookup
- Multi-Select Picklist
- Name

Get more on the AppExchange

Screen Properties

* Label: Student Info

* API Name: Student_Info

Description:

> Configure Header
> Configure Footer

Cancel Done

2.2 Create Student Record using Create Element

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

Auto-Layout ▾ Version 1: Last modified 6 days ago **Active** Run Debug Deactivate Save As Save

Create Records

* Label Create Student Record * API Name Create_Student_Record

Description

* How to set record field values From a Record Variable

How Many Records to Create

One Multiple

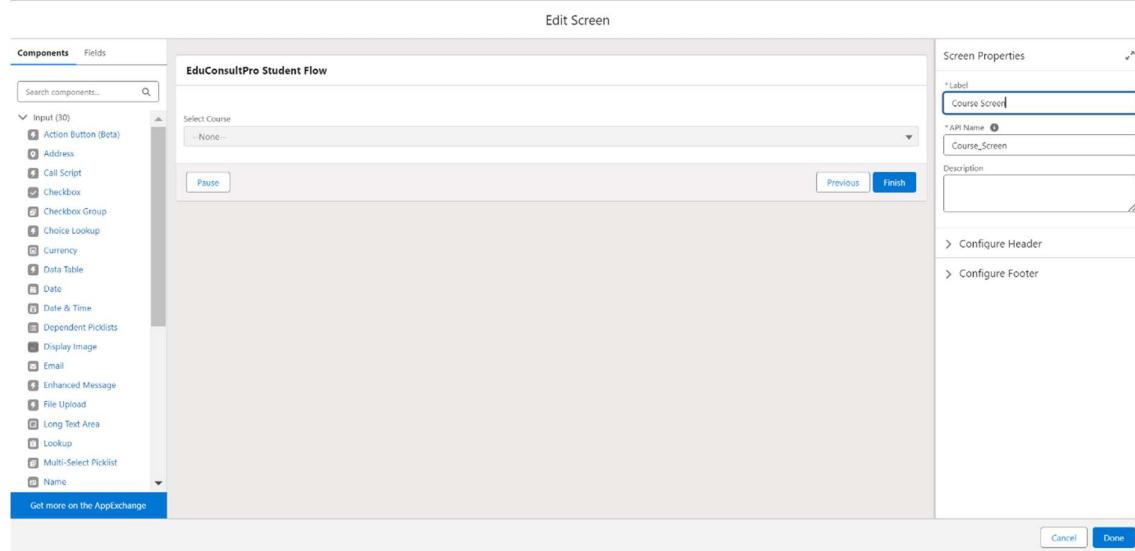
Create a Record from These Values

* Record StudentRecordRes

Make sure that ID is blank. After the flow creates the records, ID is set to match the record that was created. ⓘ

2.3 Add Screen Element

- Add a Screen Element after Create Student Record Element and label it as Course Screen.
- 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.
- Repeat the same for GRE, GMAT, Duolingo, TOEFL.



2.4 Add Decision Element

- Add a Decision Element after Select Course Screen Element, label it as Selecting Course.
- 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)
- Click on the “+” icon and Repeat step 2 for other options mentioned as below:
GRE
GMAT
DuoLingo
TOEFL
- Click Done.

Decision

* Label	* API Name <small>i</small>																					
Selecting Course	Selecting_Course																					
Description																						
<p>Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.</p> <table border="1"> <thead> <tr> <th>OUTCOME ORDER</th> <th>OUTCOME DETAILS</th> <th>Delete Outcome</th> </tr> </thead> <tbody> <tr> <td>Selected IELTS</td> <td>* Label Selected IELTS</td> <td>* Outcome API Name <small>i</small> Selected_IELTS</td> </tr> <tr> <td>Selected GRE</td> <td colspan="2">Condition Requirements to Execute Outcome All Conditions Are Met (AND)</td> </tr> <tr> <td>Selected GMAT</td> <td colspan="2"></td> </tr> <tr> <td>Selected Duolingo</td> <td colspan="2"></td> </tr> <tr> <td>Selected TOEFL</td> <td colspan="2"></td> </tr> <tr> <td>Default Outcome</td> <td colspan="2">+ Add Condition</td> </tr> </tbody> </table>		OUTCOME ORDER	OUTCOME DETAILS	Delete Outcome	Selected IELTS	* Label Selected IELTS	* Outcome API Name <small>i</small> Selected_IELTS	Selected GRE	Condition Requirements to Execute Outcome All Conditions Are Met (AND)		Selected GMAT			Selected Duolingo			Selected TOEFL			Default Outcome	+ Add Condition	
OUTCOME ORDER	OUTCOME DETAILS	Delete Outcome																				
Selected IELTS	* Label Selected IELTS	* Outcome API Name <small>i</small> Selected_IELTS																				
Selected GRE	Condition Requirements to Execute Outcome All Conditions Are Met (AND)																					
Selected GMAT																						
Selected Duolingo																						
Selected TOEFL																						
Default Outcome	+ Add Condition																					

2.5 Add GET Record Element

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

 Get Records X

* Label * API Name i

Description

Get Records of This Object

* Object

Filter Course Records

Condition Requirements

Field Operator Value X [trash]

+ Add Condition

Sort Course Records

Sort Order ⚠ If you store only the first record, filter by a unique field, such as ID.

How Many Records to Store

Only the first record
 All records

How to Store Record Data

Automatically store all fields
 Choose fields and let Salesforce do the rest
 Choose fields and assign variables (advanced)

 Get Records

* Label

* API Name

Description

Get Records of This Object

* Object

Filter Course Records

Condition Requirements

Field	Operator	Value
<input type="text" value="Name"/>	<input type="text" value="Equals"/>	<input type="text" value="Aa Select_Course X"/> <input type="button" value="X"/>

Sort Course Records

Sort Order

 If you store only the first record, filter by a unique field, such as ID.

How Many Records to Store
 Only the first record
 All records

How to Store Record Data
 Automatically store all fields
 Choose fields and let Salesforce do the rest
 Choose fields and assign variables (advanced)

2.6 Create Registration Record using Create Records Element

- Add a Create element after the Get IELTS Rec element and label it as “Create IELTS Registration Rec”.
- Select “one” under How many records to Create, and select “Use separate resources, and literal values” under How to Set the record fields.
- Select Object : Registration
 - Field : Course_Name__c
 - Value : {!GetIELTSSRec.Id}

Field : Student_Name__c

Value : {!StudentRecordRes.Id}

- Repeat the steps 1 & 2 for the GRE, GMAT, TOEFL, Duolingo paths.

 Create Records X

* Label * API Name i

Description

* How to set record field values

Create a Record of This Object

* Object

Set Field Values for the Registration

Field <input type="text" value="Course_Name__c"/>	Value <input type="text" value="Aa Course from GetIELTS_Rec > Record ID X"/> ← Aa Course from GetIELTS_Rec > Record ID X X Delete
Field <input type="text" value="Student_Name__c"/>	Value <input type="text" value="Aa StudentRecordRes > Record ID X"/> ← Aa StudentRecordRes > Record ID X X Delete

+ Add Field Manually assign variables

Check for Matching Records Disabled

 Create Records X

* Label * API Name i

Description

* How to set record field values

Create a Record of This Object

* Object

Set Field Values for the Registration

Field	Value
<input type="text" value="Course_Name__c"/>	<input type="text" value="Aa Course from Get_GRE_Rec > Record ID X"/> 
<input type="text" value="Student_Name__c"/>	<input type="text" value="Aa StudentRecordRes > Record ID X"/> 

+ Add Field Manually assign variables

Check for Matching Records  Disabled

 Get Records

* Label

* API Name

Description

Get Records of This Object

* Object

Filter Course Records

Condition Requirements

Field	Operator	Value
<input type="text" value="Name"/>	<input type="text" value="Equals"/>	<input type="text" value="Aa Select_Course X"/> X

[+ Add Condition](#)

Sort Course Records

Sort Order

⚠ If you store only the first record, filter by a unique field, such as ID.

How Many Records to Store

Only the first record

All records

How to Store Record Data

Automatically store all fields

Choose fields and let Salesforce do the rest

Choose fields and assign variables (advanced)

2.7 Create Email Text Template Variables for email body and subject

- Click on the toggle toolbox on the left corner, click “New Resource”, then select “Text Template” as Resource Type.
- Give the API name as “StuRegistrationEmailTextTempBody”, select “view as plain text” and paste the below text in body.

“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.

At EduConsultantPro, we understand the importance of your academic and career goals, and we are committed to providing you with the highest level of support and guidance throughout your journey.

Here are a few key points to help you get started:

Explore Our Resources : Take some time to explore the wide range of resources, tools, and services available on the EduConsultantPro platform. From educational insights to immigration advice, we offer comprehensive support tailored to your needs.

Connect with Our Consultants : Our team of experienced consultants is here to assist you at every stage of your educational and immigration endeavors. Don't hesitate to reach out to us with any questions, concerns, or inquiries you may have. We're here to help!

Stay Updated : Keep an eye on your inbox for important updates, announcements, and exclusive opportunities from EduConsultantPro. We'll ensure that you're informed about the latest developments and relevant information to support your journey.

Engage with the Community : Join our vibrant community of students, professionals, and experts who share your passion for education and global opportunities. Connect with like-minded individuals, participate in discussions, and expand your network.

Once again, congratulations on taking this important step towards realizing your academic and career aspirations. We are thrilled to have you as part of the EduConsultantPro family and look forward to supporting you on your journey to success.

If you have any questions or need assistance, please don't hesitate to contact us.

Thank you.”

- Click Done.
- Repeat steps 1 & 2 to create an email text template for the email subject, label it as “StuRegistrationEmailTextTempSub”, write a text message in the body and save it.

Send Email

* Label
Send Email to Student

* API Name [i](#)
Send_Email_to_Student

Description

Send Email [i](#)
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	<input type="checkbox"/>	Not Included
○ Add Threading Token to Subject	<input type="checkbox"/>	Not Included
Aa Body i  StuRegistrationEmailTextTempBody X	<input checked="" type="checkbox"/>	Included
Aa Email Template ID	<input type="checkbox"/>	Not Included
○ Log Email on Send	<input type="checkbox"/>	Not Included
Aa Recipient Address Collection	<input type="checkbox"/>	Not Included

2.8 Add an Action Element

- Add an Action Element after all the Decision paths, label it as “Send Email to Student”.

- Under “Set input values for selected action”, include body, Recipient Address List and Subject.
- For input Body : {!StuRegistrationEmailTextTempBody},
- Recipient Address List : {!StudentRecordRes.Email__c},
- Subject : {!StuRegistrationEmailTextTempSub}.

Send Email

 StuRegistrationEmailTextTempBody	Included
A_a Email Template ID	<input type="checkbox"/>
⌚ Log Email on Send	<input type="checkbox"/>
A_a Recipient Address Collection	<input type="checkbox"/>
A_a Recipient Address List <small>i</small>	<input checked="" type="checkbox"/> Included
A_a StudentRecordRes > Email	
A_a Recipient ID	<input type="checkbox"/>
A_a Related Record ID	<input type="checkbox"/>
⌚ Rich-Text-Formatted Body	<input type="checkbox"/>
A_a Sender Email Address	<input type="checkbox"/>
A_a Sender Type	<input type="checkbox"/>
A_a Subject <small>i</small>	<input checked="" type="checkbox"/> Included
 StuRegistrationEmailTextTempSub	

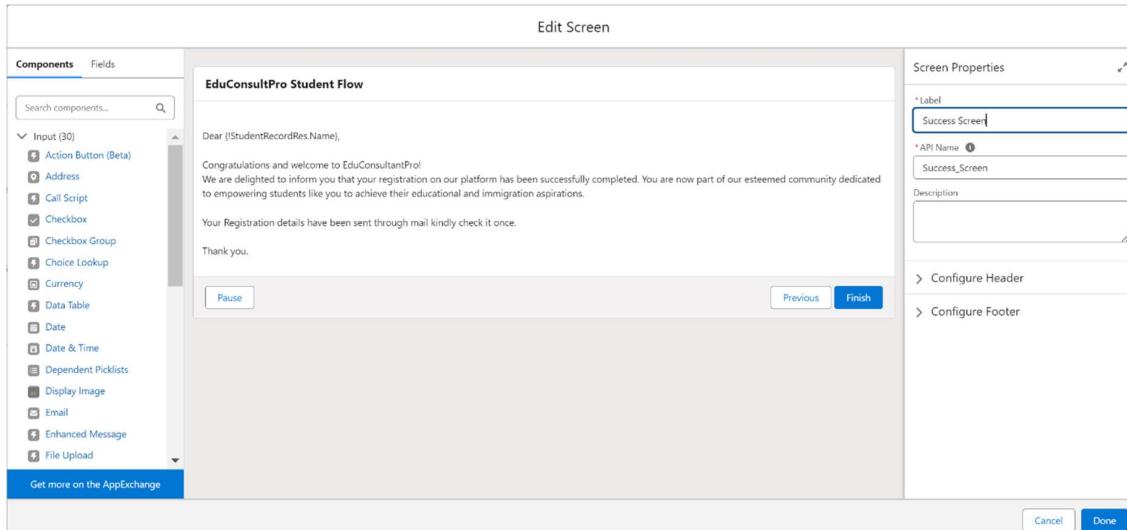
2.9 Add Screen Element

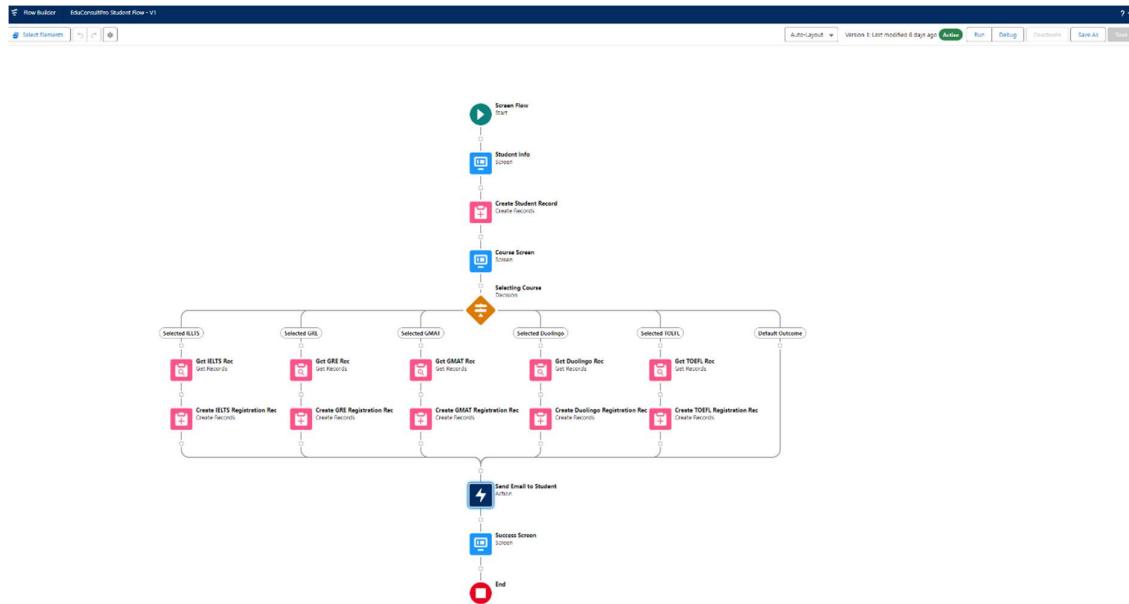
- Add a Screen Element after the Send Email to Student Action Element, label it as Success Screen.
- From the left side panel search for the Display text component and drag it to the main panel, label it as “SuccessMessage”.
- Paste the below in the Resource picker box.

“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.”

- Click Done.
- Save the flow and name it as “EduConsultPro Student Flow”. Your flow will look as shown below:





Task-3 Create Users

3.1 User

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

User Detail		Role	
Name	Consultant	User License	Salesforce Platform
Alias	cons	Profile	Standard Platform User
Email	gopuajith319@gmail.com [Verify] i	Active	<input checked="" type="checkbox"/>
Username	gopuajith319@gmail.com	Marketing User	<input type="checkbox"/>
Nickname	User17225157712497805899 i	Offline User	<input type="checkbox"/>
Title		Knowledge User	<input type="checkbox"/>
Company		Flow User	<input type="checkbox"/>
Department		Service Cloud User	<input type="checkbox"/>
Division		Site.com Contributor User	<input type="checkbox"/>
Address	India	Site.com Publisher User	<input type="checkbox"/>
Time Zone	(GMT-07:00) Pacific Daylight Time (America/Los_Angeles)	WDC User	<input type="checkbox"/>
Locale	English (United States)	Mobile Push Registrations	View
Language	English	Data.com User Type	i
Delegated Approver		Accessibility Mode (Classic Only)	<input type="checkbox"/>
Manager		Debug Mode	<input type="checkbox"/> i
Receive Approval Request Emails	Only if I am an approver	High-Contrast Palette on Charts	<input type="checkbox"/> i
Federation ID		Load Lightning Pages While Scrolling	<input checked="" type="checkbox"/> i
App Registration: One-Time Password Authenticator	i	Salesforce CRM Content User	<input checked="" type="checkbox"/>
App Registration: Salesforce Authenticator	i	Receive Salesforce CRM Content Email Alerts	<input checked="" type="checkbox"/>
Security Key (U2F or WebAuthn)	i	Receive Salesforce CRM Content Alerts as Daily Digest	<input checked="" type="checkbox"/>
Lightning Login	i	Make Setup My Default Landing Page	<input type="checkbox"/>
Temporary Verification Code (Expires in 1 to 24 Hours)	[Generate] i	Allow Forecasting	<input type="checkbox"/>
		No MRU Updates	<input type="checkbox"/> i
		Call Center	<input type="checkbox"/>
		Phone	<input type="checkbox"/>

3.2 Configure the User Settings

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

Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/> Edit	Chatter_Expert	Chatter	chatty_00idak000006jpolea0_dhufor4krd@chatter.salesforce.com		✓	Chatter Free User
<input type="checkbox"/> Edit	Consultant	cons	ogruweth19@mail.com		✓	Standard Platform User
<input type="checkbox"/> Edit	GOPAL JITH_2111CS010024_-	ZGORPU	2111cs010024@reseller-recocon-eo@zmc.com		✓	System Administrator
<input type="checkbox"/> Edit	User_Integration	Integ	integration@00idak000006jpolea0.com		✓	Analytics Cloud Integration User
<input type="checkbox"/> Edit	User_Security	sec	insightssecurity@00idak000006jpolea0.com		✓	Analytics Cloud Security User

Task-4. Create an Approval Process for Property Object

4.1 Create an Email Template

- From Setup, enter Templates in the Quick Find box, and then select Lightning Email Templates, toggle on
- go to app launcher, search for “Email Templates”, Create a new folder with the desired name.
- 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 two more Email templates for Approval and Rejection of Request similar to the previous one.

Email Templates						
Recent	3 items					
EMAIL TEMPLATES	Email Template Name	Description	Folder	Last Modified By	Last Modified Date	⋮
Recent	Approval Template		Appointment Templates	2111CS010024 - GOPU AJITH	8/1/2024, 6:18 AM	▼
Created by Me	Rejection Template		Appointment Templates	2111CS010024 - GOPU AJITH	8/1/2024, 6:19 AM	▼
Private Email Templates	Submission Template		Appointment Templates	2111CS010024 - GOPU AJITH	8/1/2024, 6:20 AM	▼

4.2 Create an Approval Process

- From Setup, enter Approval in the Quick Find box, and then select Approval Processes.
- In Manage Approval Processes For, select Appointment.
- Click Create New Approval Process --> Use Jump Start Wizard.
- Configure the approval process.
- Process Name - Appointment Approval, Under Select Approver, Select Manager for the option :“Automatically assign an approver using a standard or custom hierarchy field.”
- Click next and “Next Automated Approver Determined By” --> Select Manager.

- From Record Editability Properties --> Click on Administrators OR the currently assigned approver can edit records during the approval process.
- Save the approval process.
- Click View Approval Process Detail Page.
- Under Initial Submission Actions, click Add New --> Field Update, and configure it with these values.

Field	Value
Name	Submitted
Field to Update	Appointment: Status
A Specific value	Pending

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

Final Approval Actions		Add Existing	Add New
Action	Type	Description	
Edit	Record Lock	Lock the record from being edited	
Edit Remove	Email Alert	Approval Email Alerts	
Edit Remove	Field Update	Approved	

Final Rejection Actions		Add Existing	Add New
Action	Type	Description	
Edit	Record Lock	Unlock the record for editing	
Edit Remove	Field Update	Rejected	
Edit Remove	Email Alert	Rejection Email Alert	

Task 5 Create a Record Triggered Flow

5.1 Configure the Start Element

- From Setup, enter Flows in the Quick Find box, then select Flows.
- Click New Flow.
- Select Record-Triggered Flow.
- Click Create. The Configure Start window opens.
- For Object, select Appointment.

- For Trigger the Flow When, select A record is created. The flow will look like this:

The screenshot shows the 'Configure Start' screen of the Salesforce Flow Builder. It includes sections for 'Select Object' (Appointment), 'Configure Trigger' (A record is created), 'Set Entry Conditions' (None selected), and 'Optimize the Flow for' (Actions and Related Records selected). A checkbox for 'Include a Run Asynchronously path' is also shown.

Select Object
Select the object whose records trigger the flow when they're created, updated, or deleted.
*Object
Appointment

Configure Trigger
*Trigger the Flow When:
 A record is created
 A record is updated
 A record is created or updated
 A record is deleted

Set Entry Conditions
Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.
If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the Only when a record is updated to meet the condition requirements option for When to Run the Flow for Updated Records.

Condition Requirements
None

*Optimize the Flow for:
Fast Field Updates
Update fields on the record that triggers the flow to run. This high-performance flow runs before the record is saved to the database.
Actions and Related Records
Update any record and perform actions, like send an email. This more flexible flow runs after the record is saved to the database.

Include a Run Asynchronously path to access an external system after the original transaction for the triggering record is successfully committed

5.2 Add an Action Element

- Add an Action element after the Start Element and Select the Submit for approval action, label it as "Approval SubFlow".
- Set the RecordId to "{\$Record.Id}"
- Save the Flow, label it as "EduConsultPro Approval Flow" and Click on Activate.

Submit for Approval

* Label

Approval SubFlow

* API Name ⓘ

Approval_SubFlow

Description



Submit for Approval ⓘ

submit-submit

Use values from earlier in the flow to set the inputs for the "Submit for Approval" core action. To use its outputs later in the flow, store them in variables.

Set Input Values for the Selected Action

A_a * Record ID ⓘ

A_a Triggering Appointment_c > Record ID X

EduConsultPro Approval Flow - V1

ts ⏪ ⏩ ⚙

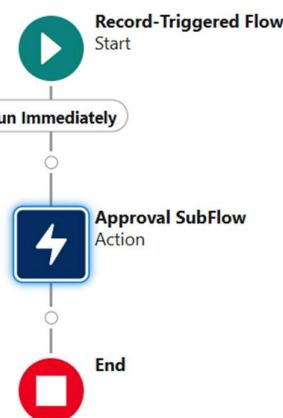
Auto-Layout

Version 1: Last modified 6 days ago

Active

Run

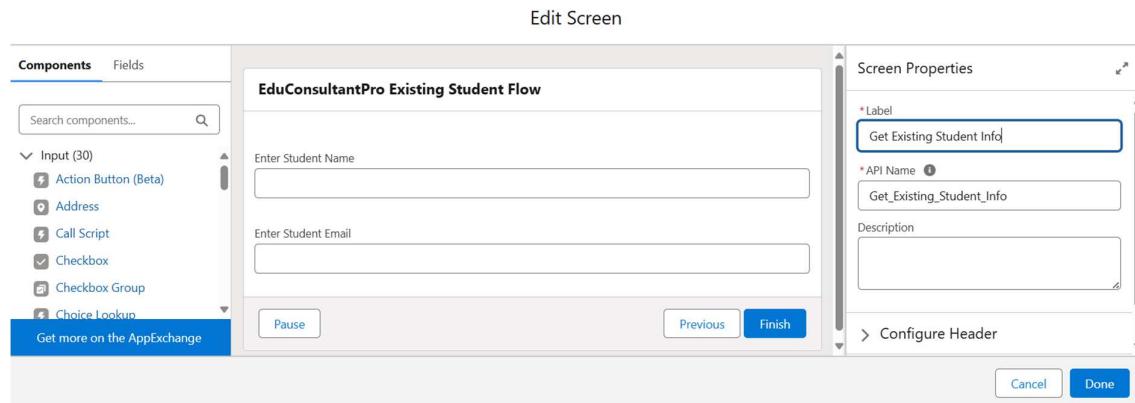
Debug



Task-6 Create a ScreenFlow for Existing Student to Book an Appointment

6.1 Add Screen Element

- From Setup, enter Flow Builder in quick find, select new flow ? ScreenFlow.
- Add a Screen element.
- In the Screen Properties pane, for Label, enter “Get Student Info”.
- 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
- Click on Done.



6.2 Add GET Record Element

- Add a GET Record Element after Decision Element, under the IELTS path and label it as “Get Rec”.
- 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}

 Get Records

* Label * API Name

Description

Get Records of This Object

* Object

Filter Student Records

Condition Requirements

Field	Operator	Value
Name	Equals	<input type="text" value="Aa Enter_Student_Name"/> <input type="button" value="X"/> <input type="button" value="Delete"/>
AND Email_c	Equals	<input type="text" value="Aa Enter_Student_Email"/> <input type="button" value="X"/> <input type="button" value="Delete"/>

6.3 Add Decision Element

- Add a Decision Element after Select Display Student Details Element, label it as “Appointment or Case”.
- 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}
- Click on the “+” icon and Repeat step 2 for Case options mentioned.

Decision

* Label	* API Name																											
Appointment or Case	Appointment_or_Case																											
Description																												
<p>Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.</p> <table border="1"> <thead> <tr> <th>OUTCOME ORDER</th> <th>OUTCOME DETAILS</th> <th>Delete Outcome</th> </tr> </thead> <tbody> <tr> <td>Appointment</td> <td>* Label</td> <td>* Outcome API Name</td> </tr> <tr> <td>Case</td> <td>Appointment</td> <td>Appointment</td> </tr> <tr> <td>Default Outcome</td> <td colspan="2">Condition Requirements to Execute Outcome</td> </tr> <tr> <td></td> <td colspan="2">All Conditions Are Met (AND)</td> </tr> <tr> <td></td> <td>Resource</td> <td>Operator</td> <td>Value</td> </tr> <tr> <td></td> <td>A_a ...Details > How_may_I_Help_you</td> <td>Equals</td> <td>A_a Book_an_Appointment</td> </tr> <tr> <td></td> <td colspan="3">+ Add Condition</td> </tr> </tbody> </table>		OUTCOME ORDER	OUTCOME DETAILS	Delete Outcome	Appointment	* Label	* Outcome API Name	Case	Appointment	Appointment	Default Outcome	Condition Requirements to Execute Outcome			All Conditions Are Met (AND)			Resource	Operator	Value		A_a ...Details > How_may_I_Help_you	Equals	A_a Book_an_Appointment		+ Add Condition		
OUTCOME ORDER	OUTCOME DETAILS	Delete Outcome																										
Appointment	* Label	* Outcome API Name																										
Case	Appointment	Appointment																										
Default Outcome	Condition Requirements to Execute Outcome																											
	All Conditions Are Met (AND)																											
	Resource	Operator	Value																									
	A_a ...Details > How_may_I_Help_you	Equals	A_a Book_an_Appointment																									
	+ Add Condition																											

6.4 Add Screen Element

- Add a Screen element after the Decision Element, on the Appointment path and label it as “Appointment Booking Screen”.
- 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.
- Drag all the fields which are needed to add on the screen inorder to collect the student information.
- Click on Done.

Edit Screen

Components	Fields									
Search components...										
Input (30)										
Action Button (Beta)										
Address										
Call Script										
Checkbox										
Checkbox Group										
Choice Lookup										
Get more on the AppExchange										
<p>EduConsultantPro Existing Student Flow</p> <table border="1"> <tr> <td>Appointment Date/Time</td> <td>Date</td> <td>Time</td> </tr> <tr> <td colspan="3">Appointment</td> </tr> <tr> <td colspan="3">Notes</td> </tr> </table>		Appointment Date/Time	Date	Time	Appointment			Notes		
Appointment Date/Time	Date	Time								
Appointment										
Notes										
<p>Screen Properties</p> <p>* Label</p> <p>Appointment Booking Screen</p> <p>* API Name</p> <p>Appointment_Booking_Screen</p> <p>Description</p> <p>Configure Header</p>										
<p>Cancel Done</p>										

6.5 Add GET Record Element

- Add a GET Record Element after Decision Element, under the Appointment path and label it as “Get Consultant Rec”.

- Select Object : Consultant

Condition Requirement : All Conditions are Met(AND)

- Field : Name

Operator : Equals

Value : {!AppointmentRecordRes.Consultant_Name__c}

The screenshot shows the configuration of a 'Get Records' element. At the top, there are fields for 'Label' (Get Consultant Rec) and 'API Name' (Get_Consultant_Rec). Below these are 'Description' and 'Object' fields. Under 'Condition Requirements', it says 'All Conditions Are Met (AND)' and shows a single condition requirement with fields: 'Field' (Name), 'Operator' (Equals), and 'Value' (Aa AppointmentRecordRes > Con...). A blue '+' button is visible for adding more conditions.

6.5 Create Appointment Record using Create Records Element

- Add a Create element after the Get Consultant Rec element and label it as “Create Appointment”.
- Select “one” under How many records to Create, and select “Use separate resources, and literal values” under How to Set the record fields.
- 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}

 Get Records X

* Label * API Name

Description

Get Records of This Object

* Object

Filter Appointment Records

Condition Requirements

Field	Operator	Value
<input type="text" value="Appointment_DateTime__c"/>	<input type="text" value="Equals"/>	<input type="text" value="AppointmentRecordRes > App..."/> X Edit
AND <input type="text" value="Consultant__c"/>	<input type="text" value="Equals"/>	<input type="text" value="A_a Consultant from Get_Consulta..."/> X Edit

AND Notes_c Equals Aa AppointmentRecordRes > Not... X

AND PurposeTopic_c Equals Aa AppointmentRecordRes > Pur... X

AND Student_Name_c Equals Aa Student from Get_Rec > Recor... X

[+ Add Condition](#)

Sort Appointment Records

Sort Order

Not Sorted ▾ ⚠ If you store only the first record, filter by a unique field, such as ID.

How Many Records to Store

Only the first record
 All records

How to Store Record Data

Automatically store all fields
 Choose fields and let Salesforce do the rest

6.6 Add Screen Element

- Add a Screen Element after the Send Email to Student Action Element, label it as “Confirmation Screen”.
- From the left side panel search for the Display text component and drag it to the main panel, label it as “Appointment_Confirmation”.
- Paste the below in the Resource picker box.

Consultant Name : {!Get_Consultant_Rec.Name},

Date & Time : {!AppointmentRecordRes.Appointment_DateTime__c},

Notes : {!AppointmentRecordRes.Notes__c},

- Click Done.

6.7 Add an SubFlow Element

- 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”.
- Save the flow and label it as “EduConsultantPro Existing Student Flow”, you can use the below image for reference.

Create a Case

* Label
Create Student Case

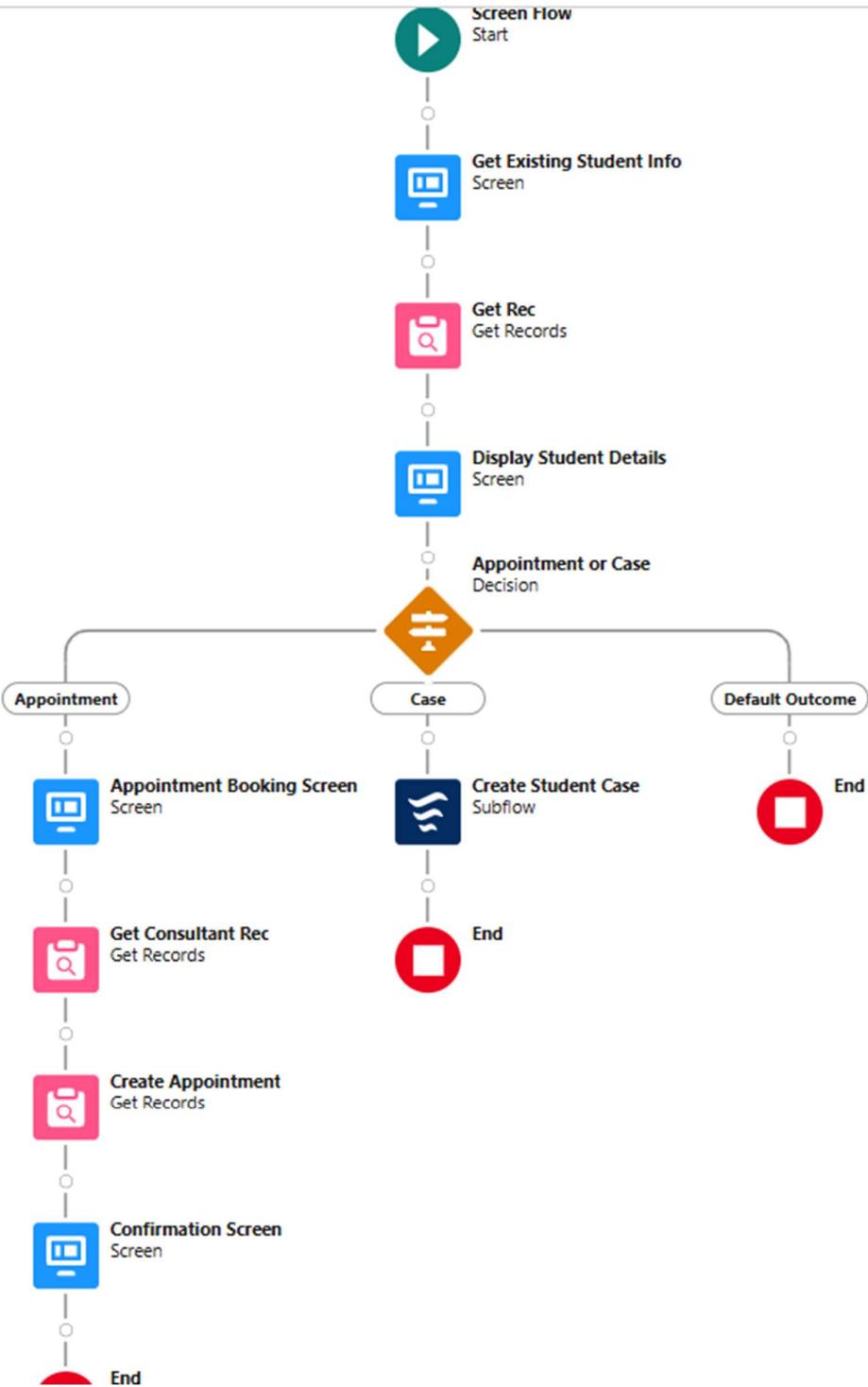
* API Name
Create_Student_Case

Description

Referenced Flow

Create a Case
setup_service_experience_Create_Case

Use values from the parent flow to set the inputs for the "Create a Case" flow. By default, the parent flow stores all outputs. You can either reference outputs via the API name of the Subflow element or manually assign variables in the parent flow to store individual outputs from the "Create a Case" flow.



Task-7 Create a ScreenFlow to Combine all the flows at one place

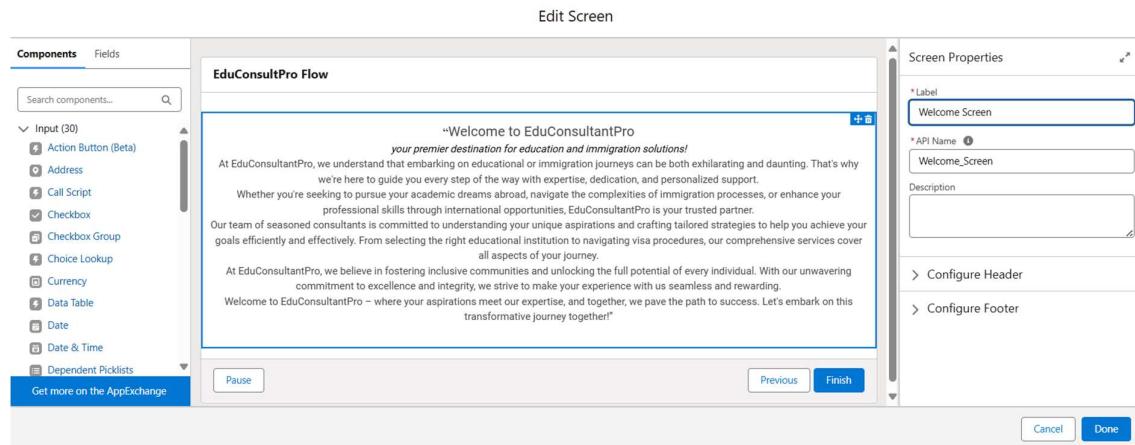
7.1 Add Screen Element

- Add a Screen Element and label it as Welcome Screen.
- From the left side panel search for the Display text component and drag it to the main panel, label it as “SuccessMessage”.

- Paste the below in the Resource picker box.
- “Welcome to EduConsultantPro
 your premier destination for education and immigration solutions!
 At EduConsultantPro, we understand that embarking on educational or immigration journeys can be both exhilarating and daunting. That's why we're here to guide you every step of the way with expertise, dedication, and personalized support.
 Whether you're seeking to pursue your academic dreams abroad, navigate the complexities of immigration processes, or enhance your professional skills through international opportunities, EduConsultantPro is your trusted partner.
 Our team of seasoned consultants is committed to understanding your unique aspirations and crafting tailored strategies to help you achieve your goals efficiently and effectively. From selecting the right educational institution to navigating visa procedures, our comprehensive services cover all aspects of your journey.
 At EduConsultantPro, we believe in fostering inclusive communities and unlocking the full potential of every individual. With our unwavering commitment to excellence and integrity, we strive to make your experience with us seamless and rewarding.

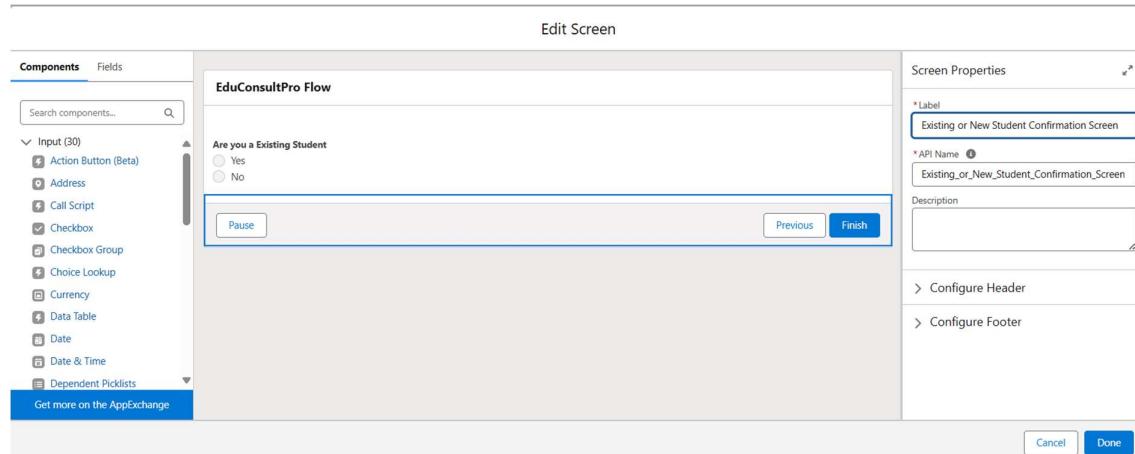
Welcome to EduConsultantPro – where your aspirations meet our expertise, and together, we pave the path to success. Let's embark on this transformative journey together!”

- Click Done.



7.2 Add Screen Element

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



7.3 Add Decision Element

- Add a Decision Element after Existing or New Student Confirmation Screen Element, label it as “Decision 1”.
- 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}
- Click on the “+” icon and Repeat step 2 for No options mentioned.

Decision

* Label	* API Name <small>i</small>
Decision 1	Decision_1
Description	
<div style="border: 1px solid #ccc; height: 60px; width: 100%;"></div>	

Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

OUTCOME ORDER	OUTCOME DETAILS	Delete Outcome
1 If Existing Student	* Label: If Existing Student * Outcome API Name: If_Existing_Student Condition Requirements to Execute Outcome: All Conditions Are Met (AND)	Delete Outcome
2 if not Existing Student		
Default Outcome	Resource: Aa ... > Are_you_a_Existing_Student X Operator: Equals Value: Aa Yes X	Delete

7.4 Add an SubFlow Element

- 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”.
- Save the flow and label it as “EduConsultantPro Existing Student Flow”.
- Click Done

EduConsultantPro Existing Student Flow

* Label
Existing Student Flow

* API Name ⓘ
Existing_Student_Flow

Description

Referenced Flow

EduConsultantPro Existing Student Flow
EduConsultantPro_Existing_Student_Flow

Use values from the parent flow to set the inputs for the "EduConsultantPro Existing Student Flow" flow. By default, the parent flow stores all outputs. You can either reference outputs via the API name of the Subflow element or manually assign variables in the parent flow to store individual outputs from the "EduConsultantPro Existing Student Flow" flow.

Set Input Values

AppointmentRecordRes

Not Included

7.5 Add an SubFlow Element

- 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”.
- Save the flow and label it as “EduConsultantPro Existing Student Flow”.
- Click Done.
- Save the flow and label it as “EduConsultPro Flow”, you can use the below image for reference.

 EduConsultantPro Student Flow X

* Label
New Student Flow

* API Name i
New_Student_Flow

Description

Referenced Flow

 **EduConsultantPro Student Flow**
 EduConsultPro_Student_Flow ▼

Use values from the parent flow to set the inputs for the "EduConsultantPro Student Flow" flow. By default, the parent flow stores all outputs. You can either reference outputs via the API name of the Subflow element or manually assign variables in the parent flow to store individual outputs from the "EduConsultantPro Student Flow" flow.



Task-8 Create a lightning app page

- From Setup, enter App Builder in the Quick Find box, then click Lightning App Builder.
- Click New, select Home Page, then click Next.
- Step through the wizard and name the page “EduConsultPro Home Page”, select the Standard Home Page template, and then click Done.
- Drag the Flow component to the top-right region.
- Search for the “EduConsultantPro Flow” and click Save.
- Click Activate, Click App and Profile, then click Assign to Apps and Profiles.
- Select the Sales app, then click Next.
- Scroll down the list of profiles and select System Administrator, then click Next.

- Review the assignment, and then click Save.

