

## Phase 4: Process Automation for EduConsultPro

### Step 1: Define Goal

- Automate Student Application Status updates when Admission Application is created/updated.
- Reduce manual work, ensure real-time updates, minimize errors.

### Step 2: Create Record-Triggered Flow

1. Setup → Flows → New Flow.
2. Select **Record-Triggered Flow** → Create.
3. Object: **Appointment**.
4. Trigger: **When a record is created**.
5. Add **Action Element** → Select **Submit for Approval** → Label: "Approval SubFlow".
6. Set **RecordId** = `{!$Record.Id}`.

The screenshot displays the Salesforce Flow Builder interface. On the left, a canvas shows a flow starting with a 'Record-Triggered Flow' start element, followed by a 'Run Immediately' connector, and ending with an 'End' element. The 'Record-Triggered Flow' element is highlighted with a blue border. On the right, the 'Configure Start' panel is open, showing the following configuration:

- 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 conserve your org's resources.**
- Condition Requirements:** None

## Action Element:

The screenshot shows a flow builder interface on the left and a configuration panel for the 'Submit for Approval' action on the right.

**Flow Builder (Left):**

- Start node: Record-Triggered Flow
- Connector: Run Immediately
- Action node: approval subflow (highlighted with a blue border)
- Connector: +
- End node: End

**Configuration Panel (Right):**

- Title: Submit for Approval
- \* Label: approval subflow
- \* API Name: approval\_subflow
- Description: (empty text area)
- Text: 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.
- Section: Set Input Values
- Input: A<sub>a</sub> \* Record ID (with a dropdown menu showing 'A - ...ent Info > Enter Student E')

## Step 3: Screen Flow for Existing Student

1. Setup → Flow Builder → New Flow → Screen Flow.

The screenshot shows a flow builder interface on the left and a configuration panel for the 'Get Records' action on the right.

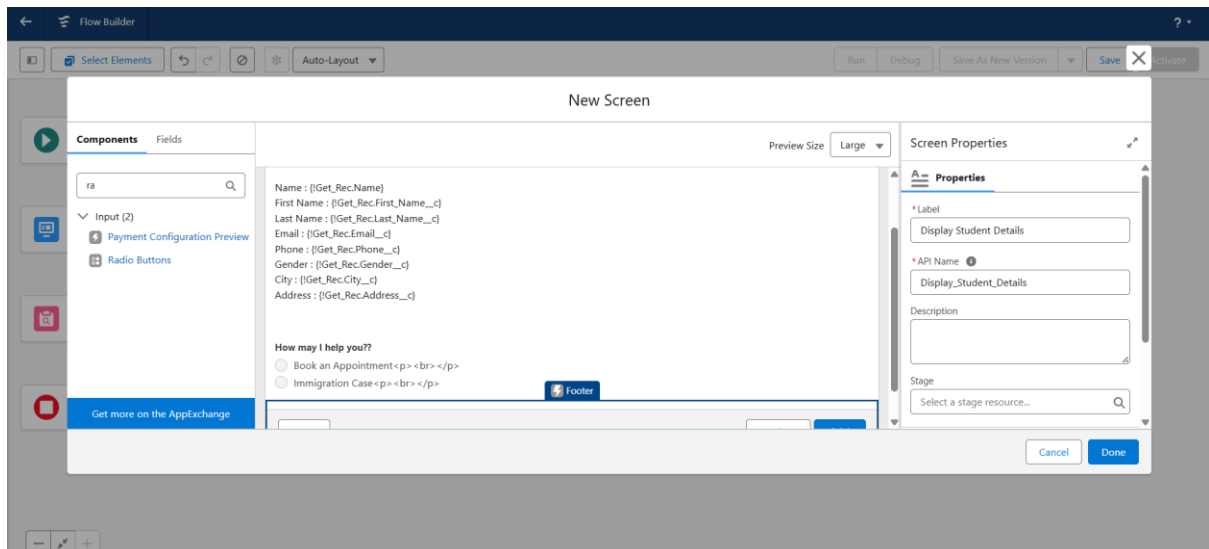
**Flow Builder (Left):**

- Start node: Screen Flow
- Connector: +
- Screen node: Get Student Info
- Connector: +
- Action node: Get Rec (highlighted with a blue border)
- Connector: +
- End node: End

**Configuration Panel (Right):**

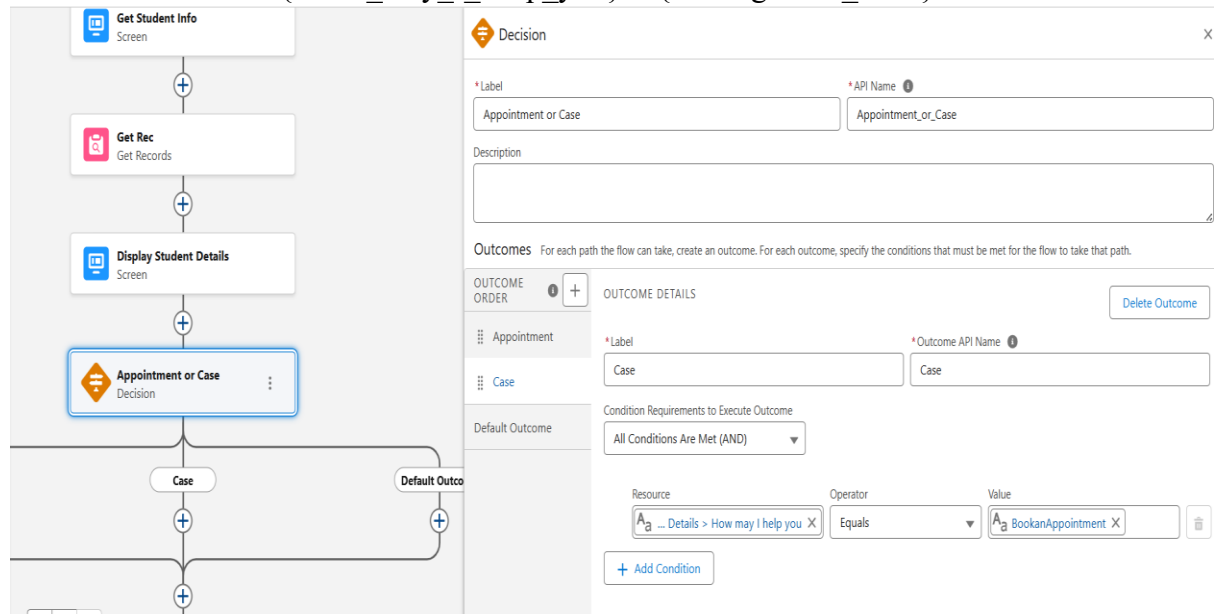
- Title: Get Records
- \* Label: Get Rec
- \* API Name: Get\_Rec
- Description: (empty text area)
- Section: Get Records of This Object
- \* Object: Student
- Section: Filter Student Records
- Condition Requirements: All Conditions Are Met (AND)
- Field: A<sub>a</sub> Student Name
- Operator: Equals
- Value: A<sub>a</sub> ...nt Info > Enter Student N
- Field: AND A<sub>a</sub> Email
- Operator: Equals
- Value: A<sub>a</sub> ...ent Info > Enter Student E
- + Add Condition

2. Add **Screen Element** → Label: "Get Student Info".
  - Add 2 Text Components → Labels: "Enter Student Name", "Enter Student Email".



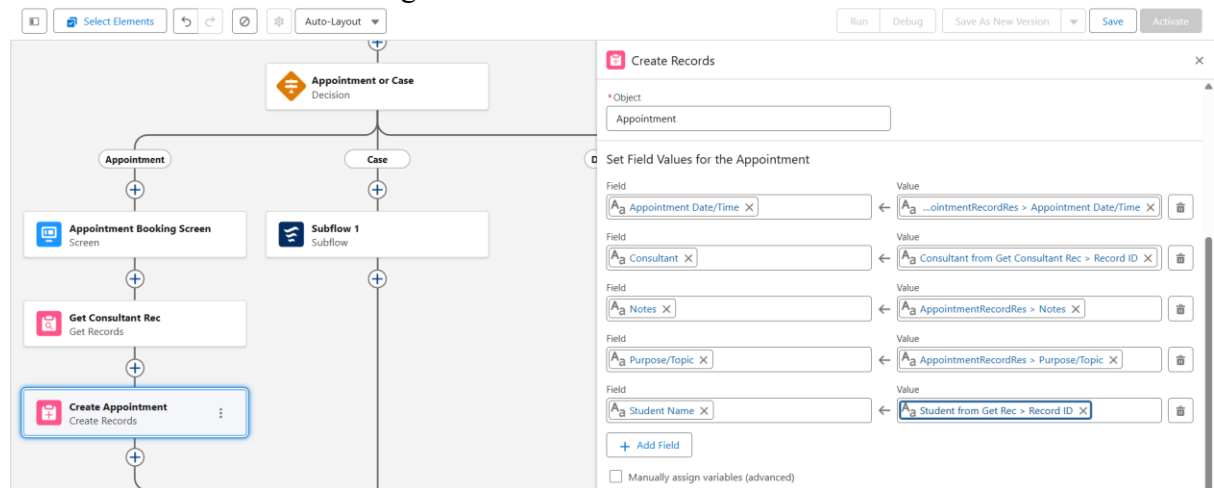
3. Add **Get Record Element** → Label: "Get Rec".
  - Object: **Student**.
  - Conditions: Student Name = {!Enter\_Student\_Name}, Email\_\_c = {!Enter\_Student\_Email}.
4. Add **Screen Element** → Label: "Display Student Details".
  - Display Text with Student fields (Name, Email, Phone, etc.).
  - Add **Radio Buttons** → Label: "How may I help you?"
  - Create Choices: "Book an Appointment", "Immigration Case".
5. Add **Decision Element** → Label: "Appointment or Case?"
  - Outcome 1: Appointment → {!How\_may\_I\_Help\_you} = {!Book\_an\_Appointment}.

- Outcome 2: Case →  $\{!How\_may\_I\_Help\_you\} = \{!Immigration\_Case\}$ .



## Step 4: Check Application Status

1. Add **Decision Element** → Label: "Check Status".
  - Outcome 1: Approved →  $\{\$Record.Status\_c\} = "Approved"$ .
  - Outcome 2: Rejected →  $\{\$Record.Status\_c\} = "Rejected"$ .
  - Outcome 3: Default → Pending/Other.



## Step 5: Create Email Templates

1. Setup → Templates → Lightning Email Template (ON).
2. Create Folder → Create New Template.
3. Add **Submission Template** (confirmation email).

4. Create **Approval Template**.
5. Create **Rejection Template**.

Email Templates

Recent

3 items

EMAIL TEMPLATES

Recent

Created by Me

Private Email Templates

Public Email Templates

All Email Templates

FOLDERS

All Folders

Created by Me

Email T... ▾

Descrip... ▾

Folder ▾

Last M... ▾

Last M... ▾

Student Rejection Template

CRM NM

Jagarapu y

26/9/2025, 12:35 pm

▾

Student Request Template

CRM NM

Jagarapu y

26/9/2025, 12:33 pm

▾

Student Approval Template

CRM NM

Jagarapu y

26/9/2025, 12:24 pm

▾

## Step 6: Create Approval Process

1. Setup → Approval Processes → Appointment → New → Jump Start Wizard.
2. Process Name: **Appointment Approval**.
3. Select Approver → **Manager**.
4. Record Editability: **Admins/Assigned Approver**.
5. Save.
6. Configure Actions:
  - **Initial Submission Actions:**
    - Field Update → Status = Pending.
    - Email Alert → Submission Email Template.
  - **Final Approval Actions:**
    - Field Update → Status = Approved.
    - Email Alert → Approval Template.
  - **Final Rejection Actions:**
    - Field Update → Status = Rejected.
    - Email Alert → Rejection Template.

Q approval

✓ Data

Mass Transfer Approval Requests

✓ Feature Settings

Approval Settings

✓ Process Automation

Approval Processes

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

SETUP

Approval Processes

12:48 pm

1:12 pm

Initial Submission Actions

Add Existing

Add New

Action	Type	Description
	Record Lock	Lock the record from being edited
<a href="#">Edit</a>   <a href="#">Remove</a>	Field Update	<a href="#">Submitted</a>
<a href="#">Edit</a>   <a href="#">Remove</a>	Email Alert	<a href="#">Submission Email Alert</a>

Approval Steps

New Approval Step

Action	Step Number	Name	Description	Criteria	Assigned Approver
<a href="#">Show Actions</a>   <a href="#">Edit</a>   <a href="#">Del</a>	1	Step 1			<a href="#">Manager</a>

Final Approval Actions

Add Existing

Add New

Action	Type	Description
<a href="#">Edit</a>	Record Lock	Lock the record from being edited
<a href="#">Edit</a>   <a href="#">Remove</a>	Field Update	<a href="#">Submitted</a>
<a href="#">Edit</a>   <a href="#">Remove</a>	Email Alert	<a href="#">Submission Email Alert</a>

### Step 7: Update Student Status

1. Add **Update Records Element** → Label: "Update\_Student\_Status".
2. Record: Student (from Get\_Student).
3. Field: Status    c → Value: Approved / Rejected / Pending (based on Decision).

### Step 8: Connect Flow Elements

- Start → Get\_Student → Check Status → Update\_Student\_Status → End.
- Connect Approved, Rejected, Default paths.

### Step 9: Save & Activate Flow

- **Flow Label:** EduConsultantPro Existing Student Flow
- **API Name:** EduConsultantPro Existing Student Flow

- **Description:** Automatically updates Student\_\_c status based on AdmissionApplication\_\_c status when the record is created/updated.

Save → Activate.

