Phase 4: Process Automation (Admin)

Introduction

In this phase, we implemented **automation features in Salesforce** to streamline the workflow of our Skill Development & Employment Portal. The goal was to minimize manual tasks, improve accuracy, and ensure timely notifications for users (Students, Employers, Trainers, and Admins).

Salesforce provides multiple tools for process automation, and we applied them to enforce business rules, trigger actions, and guide the overall process.

Step 1 — Preparing org and create test data (required before automation)

Purpose: enable email delivery & notifications and create sample records you'll use to test all automations (Job, Training, Student, Employer, Application).

in my Trailhead Playground (Lightning Experience):

1. **Launch Playground / Dev Org**
- Go to Trailhead → click your avatar → **Hands-on Orgs** / **Launch** your Trailhead Playground (or open your Developer Edition org).
- Confirm you're in the right org (org name shown in the top-right).
- *(Why)*: all configuration and screenshots must be done inside the org where you'll build automations.
2. **Open Setup**
- Click the **Gear (⚙) icon** → **Setup**.
3. **Enable Email Deliverability** (so email alerts actually send)
- In Setup Quick Find, type **Deliverability** → open **Deliverability**.
- Set **Access Level** (or Access to Send Email) to **All Email** → **Save**.

- *Why:* Email Alerts use your org's deliverability setting — Trailhead orgs default to "System email only" which blocks test emails.

4. **Create 2 simple Lightning Email Templates** (you'll wire these to Email Alerts later)
- In Setup Quick Find, type **Email Templates → Email Templates**.
- Click **New Email Template** (Lightning). Create:
- **Template 1**
- Name: `Job Application – To Employer`
- Subject: `New application for: {!Job__c.Name}` *(replace merge field with your Job field if different)*
- Body: short message: "A new candidate has applied. Applicant: {!Contact.Name}. Job: {!Job__c.Name}. View the application: {!Record.Link}"

Job Application – To Employer

Preview your email template below.

Email Template Detail [Edit] [Delete] [Clone]

Email Templates from Salesforce	Unfiled Public Classic Email Templates		
Email Template Name	Job Application – To Employer	Available For Use	✓
Template Unique Name	Job_Application_To_Employer	Last Used Date	
Encoding	Unicode (UTF-8)	Times Used	
Author	T. Rupasree [Change]		
Description			
Created By	T. Rupasree, 9/19/2025, 9:30 AM	Modified By	T. Rupasree, 9/19/2025, 9:30 AM

[Edit] [Delete] [Clone]

Email Template [Send Test and Verify Merge Fields]

| Subject | New application for: {!Job__c.Name}

Plain Text Preview

"A new candidate has applied. Applicant: {!Contact.Name}. Job: {!Job__c.Name}. View the application: {!Record.Link}"

- **Template 2**
- Name: `Application Status – To Student`
- Subject: `Your application status for {!Job__c.Name}`
- Body: "Hi {!Contact.FirstName}, your application status is now: {!Application__c.Status__c}."
- Save templates. (You don't need perfect merge fields now — placeholders are fine; we'll fix exact merge fields when wiring alerts.)

Text Email Template
Application Status – To Student

Preview your email template below.

Email Template Detail [Edit] [Delete] [Clone]

Email Templates from Salesforce	Unfiled Public Classic Email Templates		
Email Template Name	Application Status – To Student	Available For Use	✓
Template Unique Name	Application_Status_To_Student	Last Used Date	
Encoding	Unicode (UTF-8)	Times Used	
Author	T. Rupasree [Change]		
Description			
Created By	T. Rupasree, 9/19/2025, 9:31 AM	Modified By	T. Rupasree, 9/19/2025, 9:31 AM

[Edit] [Delete] [Clone]

Email Template   [Send Test and Verify Merge Fields]

| Subject | Your application status for {!Job__c.Name} |
| **Plain Text Preview** | |

"Hi {!Contact.FirstName}, your application status is now: {!Application__c.Status__c}."

5. **Create Custom Notification Types** (for in-app notifications)
- In Setup Quick Find, type **Notification Builder** → open **Custom Notifications**.
- Click **New** and create two types:
- `New Application` (API name `New_Application`) — Channels: Desktop & Mobile (select available channels).
- `Application Status Changed` (API name `Application_Status_Changed`) — Channels: Desktop & Mobile.
- Save.

... and security practices of third party systems or applications like Google Cloud Messaging or Apple Push Notification Service.

Custom Notification Types   [New]

Send custom notifications using Flows or Process Builder

NOTIFICATION NAME	API NAME	NAMESPACE	DESKT
enablement_coaching_feedback_ready	enablement_coaching_feedback_ready		✓
New Application	New_Application		✓

6. **Create sample test records (so automations can be tested)**
- Use App Launcher (grid icon) → search for your custom objects
  (e.g., **Jobs**, **Trainings**, **Applications**, **Contacts**).
- Create:
- **Employer Account** (Accounts) — e.g., `Acme Hiring`
- **Contact / Student** — e.g., `Rupa Sree` (email a working email you control for testing)
- **Job Posting** — Title: `Frontend Developer - Test`, Location: `Hyderabad`, Salary Range: `20,000-40,000`, Closing Date: (choose a future date)
- **Training Program** — Name: `Full-Stack Bootcamp Test`, Start Date: choose date
- **Job Application** — Link Applicant (Contact) to Job Posting, Status = `Pending`
- Save each record.

Automation Components

Validation Rules

- **Purpose**: Ensure data integrity by restricting incorrect or incomplete data entry.
- **Examples in Project**:
- Prevent students from registering without entering a valid email and phone number.
- Ensure that job postings must include **Salary Range** and **Job Location**.
- Block duplicate skill names in the Skills object.
- Validation Rules ensure users enter correct data before saving records.

In our **Skill Development & Employment Portal**, we'll create rules like:

- **Students must enter a valid Email address**
- **Job Posting must include Salary Range**

◆ Procedure (do this in my Trailhead Playground):

Rule 1: Candidate Email Validation

1. Go to **Setup** → Quick Find: **Object Manager**.
2. Select your **Candidate(Contact)** object (if you're using Contact for students).
3. In the left panel, click **Validation Rules** → **New**.
4. Enter:
- **Rule Name**: `Valid_Email`
- **Error Condition Formula**:
- `NOT(REGEX( Email , "^[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\\.[A-Za-z]{2,4}$"))`
- **Error Message**: Enter valid email
- **Error Location**: Field → **Email**.

5. Save.

Rule 2: Job Posting Salary Range Required

1. In **Object Manager**, open your **Job** (custom object).
2. Go to **Validation Rules → New**.
3. Enter:
* **Rule Name**: `Salary_Range_Required`
* **Error Condition Formula**:
* `ISBLANK( Salary_Range__c )`
* **Error Message**: `Salary Range must be provided for all job postings.`
* **Error Location**: Field → **Salary Range**.
4. Save.

◆ Testing the Rules

1. Go to **App Launcher** → open **Students (Contacts)**.
- Try creating a new student record with an invalid email (e.g., `test@abc`).
- Confirm Salesforce blocks the save and shows your error message.
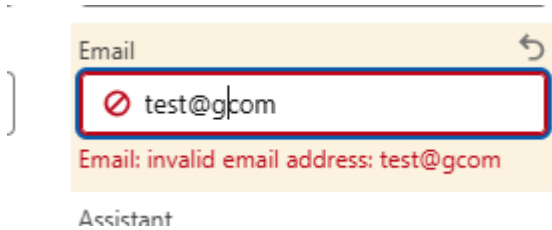
2. Go to **Job Postings** → create a new job posting **without entering Salary Range**.
- Confirm Salesforce blocks save and shows your error message.

⊞ Workflow Rules

  - o **Purpose**: Automate simple "if-then" actions based on record changes.
  - o **Examples in Project**:
  - o When a **Job Application** is submitted, send an **email alert** to the employer.
  - o If a training session is updated with a new schedule, notify registered students.

  - ➢ Workflow Rules help you **send alerts, update fields, or create tasks** automatically when conditions are met.
  - ➢ For our **Skill Development & Employment Portal**, let's build one:

◆ Use Case Example: Notify Employer when a Job Application is Submitted

When a student submits a new **Job Application**, the employer should immediately get an **Email Alert**.

Procedure
1. Go to **Setup** → Quick Find: **Workflow Rules**.

2. Click **New Rule**.
3. Select **Object**: `Job` (your custom object).
4. Click **Next**.
5. Enter:
- **Rule Name**: `Notify_Employer_New_Application`
- **Evaluation Criteria**: `created` (Evaluate the rule when a record is created).
- **Rule Criteria**:
- Field: `Application Status`
- Operator: `equals`
- Value: `Pending`

- Click **Save & Next**.

Workflow Rule

Notify_Employer_New_Application

Help for this Page ❓

ⓘ Go with the flow! With Flow Builder, the future of low-code automation, you can do everything you do with workflow rules — and more! Salesforce plans to retire workflow rules and recommends building automation in Flow Builder. Tell Me More |
Migrate your workflow rules to flows

Workflow Rule Detail   [Edit] [Delete] [Clone] [Activate]

Rule Name	Notify_Employer_New_Application	Object	Job
Active	☐	Evaluation Criteria	Evaluate the rule when a record is created
Description			
Rule Criteria	Job: Application Status EQUALS Pending		
Created By	T. Rupasree, 9/19/2025, 10:45 PM	Modified By	T. Rupasree, 9/19/2025, 10:45 PM

Workflow Actions   [Edit]

Immediate Workflow Actions

No workflow actions have been added.

Time-Dependent Workflow Actions   See an example

ⓘ No workflow actions have been added. Before adding a workflow action, you must have at least one time trigger defined.

---

1. In Workflow Actions, click **Add Workflow Action → New Email Alert**.
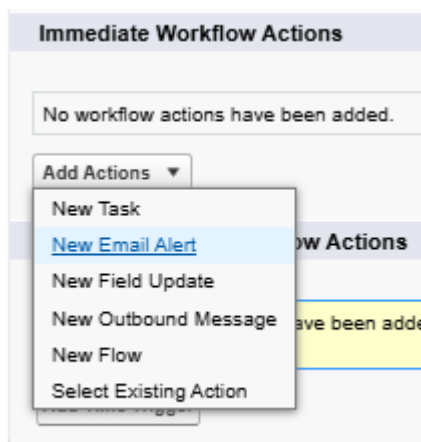
2. Enter:
- **Description**: New Job Application Alert to Employer

- **Email Template**: Select `Job Application – To Employer` (we created in Step 1).
- **Recipients**: Add **Employer Email** (lookup field on Job Posting → Employer → Contact Email).
- Save.

1. Click **Done**.
2. On Workflow Rules list → select your new rule.
3. Click **Activate**.

Testing the Workflow Rule

1. Go to **App Launcher** → Open **Job Applications**

New Job

Select a record type

○ Full-Time Job

◉ Internship

Cancel   Next

2. Create a **new Job Application** for a Student → choose Job Posting → set Status = `Pending`.

3. Save.

* Job Name ↺	Owner
Developer	🔲 T. Rupasree

Job Title

[   ]

Job Department ↺

[IT   ▼]

Location

Latitude

[   ]

Longitude

[   ]

Locationn

[--None--   ▼]

Required Experience ↺

[2   ]

Job Description

[   ]

salary range

[   ]
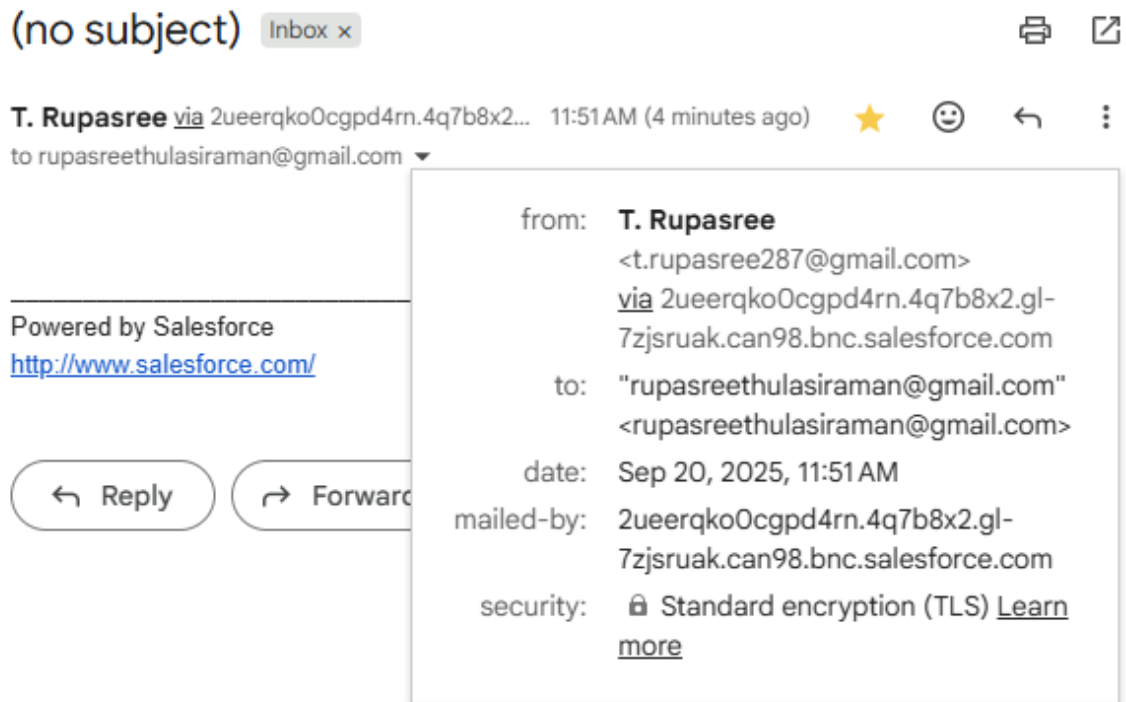
Application Status ↺

[Pending   ]

4. Check the Employer's email inbox (or if testing with your email, check your inbox)

- Workflow Rule detail page (showing criteria + email action).
- Email received in inbox.

🪧Process Builder

- **Purpose**: Build more advanced automation with multiple conditions and actions.
- **Examples in Project**:
- Automatically assign a **status = "Pending Review"** when a new Job Application is submitted.
- Create a **Task for Admin** when a new Training Program is added.
- Send a **Custom Notification** to students when their application status changes.

Why Process Builder?
Workflow Rules are limited (one action at a time). Process Builder allows **multiple conditions + multiple actions** in a single automation.

For our **Skill Development & Employment Portal**, we'll build a process that:

1. When a **Job Application** is created or updated,
- Automatically set its **Status = "Pending Review"**.
- Create a **Task for Admin** to review the application.
- Send a **Custom Notification** to the student.

◆ Procedure

Create New Process
1. Go to **Setup** → Quick Find: **Process Builder**.
2. Click **New**.
- **Process Name**: `Application_Submission_Process`
- **API Name**: auto-filled
- **Description**: `Automates actions when a job application is submitted.`
- **The process starts when**: `A record changes.`
- Click **Save**.

Process Name*

Application_Submission_Process

API Name* ⓘ

Application_Subr

Description

Automates actions when a job application is submitted.

Select Object & Criteria
1. Click **+ Add Object**.
- Choose **Job Application** object.
- Start the process: `when a record is created or edited`.
- Save.
2. Under Criteria → Click **+ Add Criteria**.
- Criteria Name: `New_Application_Criteria`
- Criteria for Executing Actions: `Conditions are met`
- Field: `Application Status`
- Operator: `equals`
- Value: `Pending`
- Save.

START

Job

+ Add Criteria

FALSE

STOP

Define Criteria for this Action Group

Criteria Name* ⓘ

New_Application_Criteria

Criteria for Executing Actions*

🔘 Conditions are met

⚪ Formula evaluates to true

⚪ No criteria—just execute the actions!

Set Conditions

	Field*	Operator*	Type*	Value*	
1	[Job__c].Applica... 🔍	Equals ▼	String ▼	Pending	✕

+ Add Row

Conditions*

🔘 All of the conditions are met (AND)

⚪ Any of the conditions are met (OR)

⚪ Customize the logic

> Advanced

Save   Cancel

Add Immediate Actions

(A) Field Update → Status

- Action Type: **Update Records**.
- Name: `Update_Status_PendingReview`.
- Record Type: `Select the Job Application record that started the process.`
- Field: `Status` → Value = `Pending Review`.
- Save.

(B) Create a Task for Admin

- Action Type: **Create a Record**.
- Name: `Create_Admin_Review_Task`.
- Record Type: `Task`.
- Set Fields:
- Subject = `Review New Job Application`
- Assigned To = Admin User (lookup your username if you're Admin).
- Status = `Not Started`
- Priority = `High`.
- Save.

Action Type*

Create a Record ▼

Action Name* ⓘ

Create_Admin_Review_Task

Record Type*

Task ▼

Set Field Values

Field*		Type*		Value*	
Assigned To ID	▼	ID	▼	Rupasree	✕
Priority	▼	Picklist	▼	High	✕
Status	▼	Picklist	▼	Not Started	✕

+ Add Row

Save   Cancel

(C) Send Custom Notification to Student

- Action Type: **Send Custom Notification**.
- Name: `Notify_Student_Application`.
- Notification Type = `Application Status Changed` (from Step 1).
- Recipient: **Student (Contact)** related to the application.
- Title = `Your job application has been received!`
- Body = `Your application is now under review.`
- Save.

Select and Define Action

enablement_coaching_feedback_rea▼

Only notification types without predefined content are supported in Process Builder. Learn More in Help

Notification Recipient*   User*

| User ▼ | User Field from a Record ▼ | [Job__c].Owner.User.Id 🔍 |

Notification Title*

| Merge Field 🔍 |

Your job application has been received!

Notification Body*

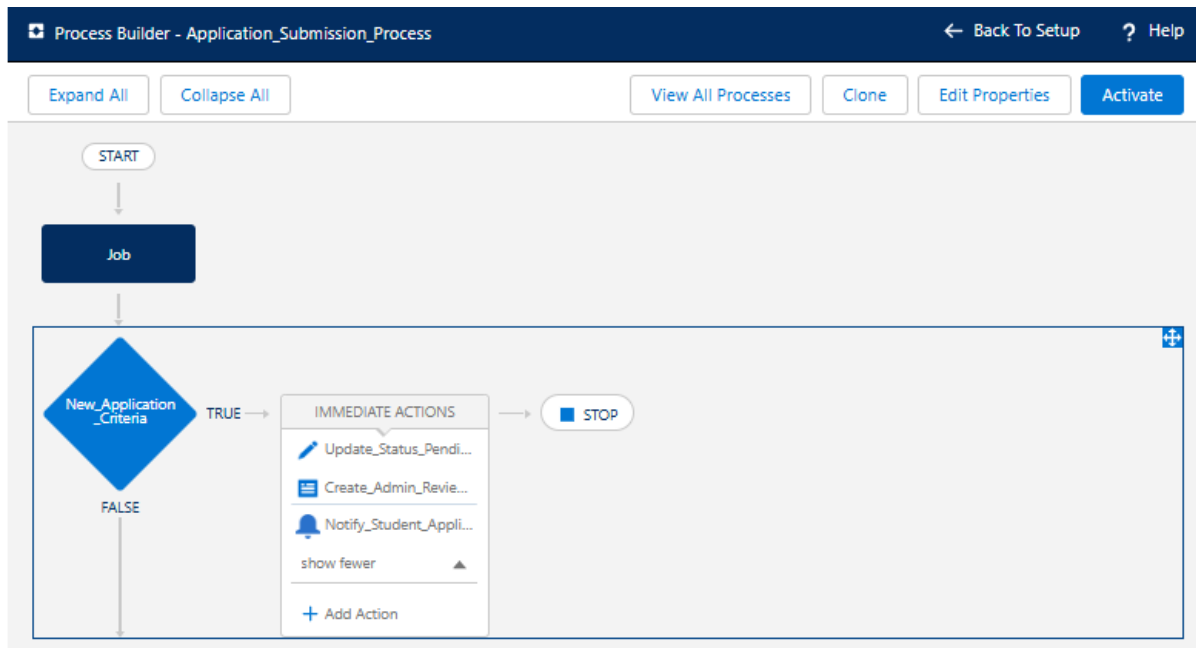| Merge Field 🔍 |

Your application is now under review

Target Object*
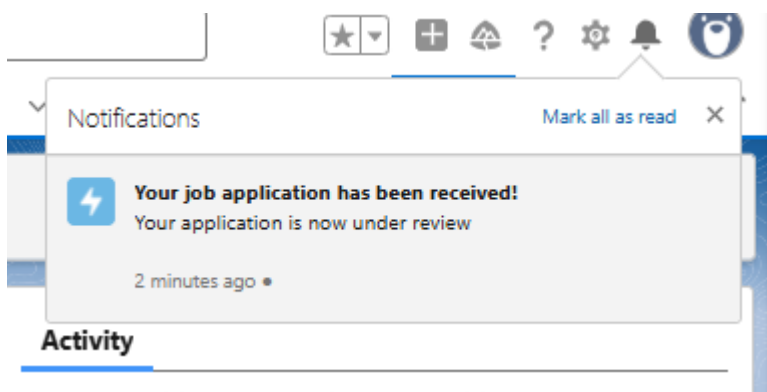🔘 Job__c Object that started the process

[Save] Cancel

Activate the Process

- Click **Activate** at the top.

◆ Testing the Process

1. Go to **Job Applications** → create a new record.
- Student = (pick your test Contact).
- Job Posting = (pick your test Job).
- Status = `Pending`.
- Save.
2. Verify Results:
- The **Status** auto-updates to `Pending Review`.
- A **Task** is created and assigned to Admin.
- The **Student** receives a custom notification (check bell icon in Salesforce).

Auto updated as pending review

🔧 Approval Process

- o **Purpose**: Automate record approval workflows.
- o **Examples in Project**:
- o Employers' job postings go through an **Admin approval process** before being visible to students.
- o Training programs created by trainers require approval before publishing.

- o **Why Approval Process?**
  It ensures that important records (like Job Postings or Training Programs) go through proper review before becoming active.

For our **Skill Development & Employment Portal**, let's create one for **Job Postings**:

- Employer submits a Job Posting.

- Admin reviews & approves.
- If approved → Status = "Approved".
- If rejected → Status = "Rejected".
- Employer gets email notification.

◆ Procedure

Create Approval Process

1. Go to **Setup** → Quick Find: **Approval Processes**.
2. Click **Approval Process Wizard** → **Jump Start Wizard** (simpler for demo).
3. Select **Object**: `Job Posting`.
4. Enter:
- **Name**: `Job_Posting_Approval`
- **Unique Name**: auto-filled.
- **Entry Criteria**: `Status equals Submitted`
- **Approver**: `Automatically assign to user` → Select Admin (your user).
- **Email Template**: Use a Job Approval template (you can reuse `Job Application - To Employer` template, or create a new one like `Job Posting Approval Notification`).
- Save.

Add Field Updates

We'll set **Status** field to change based on outcome.

1. In Approval Process → Click **Final Approval Actions** → **Add New** → **Field Update**.
- Name: `Set_Status_Approved`.
- Object: `Job Posting`.
- Field: `Status`.
- Value: `Approved`.
- Save.
2. In **Final Rejection Actions** → Add **Field Update**.
- Name: `Set_Status_Rejected`.
- Field: `Status`.
- Value: `Rejected`.

- Save.

Activate Approval Process

1. Click **Activate**.

◆ Testing the Approval Process

1. Go to **App Launcher → Job Postings**.
2. Create a new Job Posting:
- Title = `Backend Developer Test`.
- Salary Range = `20,000–40,000`.
- Status = `Submitted`.
- Save.
3. On the record detail page → click **Submit for Approval**.
4. Login as Admin (your user should already be approver).
- Open the Approval Request.
- Click **Approve**.
- Confirm **Status field = Approved**.

Flow Builder

Salesforce **Flows** allowed us to create more advanced automations, including screens, record-triggered actions, and scheduled tasks.

- **Screen Flow**: Guided student registration with conditional questions (e.g., education background).
- **Record-Triggered Flow**: When a job is closed, automatically update related applications to **"Not Selected"**.
- **Scheduled Flow**: Send reminders to students about upcoming training sessions.
- **Auto-Launched Flow**: Auto-assign default skills to new students upon registration.

Step 6: Flow Builder — the most powerful automation tool in Salesforce.

We'll do this step-by-step with **four types of Flows**:

1. **Record-Triggered Flow** → auto actions when a record changes.
2. **Screen Flow** → guided wizard for users.
3. **Scheduled Flow** → runs on a schedule.
4. **Auto-launched Flow** → called by other automation (Process Builder, Apex).

Use Case:
When a Job Application is **Approved**, automatically:

- Enroll the Student into a default Training Program.
- Send them a notification.

◆ Procedure

1. **Go to Flow Builder**
- Setup → Quick Find: **Flows** → Click **New Flow**.
- Select **Record-Triggered Flow**.
- Object = `Job Application`.
- Trigger = `When a record is updated`.
- Condition = `Status equals Approved`.
- Optimize for = **Actions and Related Records**.
- Click **Done**.

Approval Processes
Job: Job_Posting_Approval
« Back to Approval Process List

Process Definition Detail | Edit ▾ | Clone | Delete | Activate |

Process Name	Job_Posting_Approval	Active	☐
Unique Name	Job_Posting_Approval	Next Automated Approver Determined By	Manager of Record Submitter
Description			
Entry Criteria	Job: Application Status EQUALS Submitted		
Record Editability	Administrator ONLY	Allow Submitters to Recall Approval Requests	☐
Approval Assignment Email Template			
Initial Submitters	Job Owner		
Created By	T. Rupasree, 9/20/2025, 12:12 AM	Modified By	T. Rupasree, 9/20/2025, 12:12 AM

Initial Submission Actions ℹ️ | Add Existing | Add New ▾ |

Action	Type	Description
	Record Lock	Lock the record from being edited

Approval Steps ℹ️ | New Approval Step |

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions | Edit | Del	1	Step 1			Manager	Final Rejection

Final Approval Actions ℹ️ | Add Existing | Add New ▾ |

Action	Type	Description
Edit	Record Lock	Lock the record from being edited

Final Rejection Actions ℹ️ | Add Existing | Add New ▾ |

Action	Type	Description

---

2. **Add Action: Create Record (Training Enrollment)**
- In canvas, click **+ → Create Records**.
- Label = `Enroll Student in Training`.
- Record Type = `Training Enrollment` (or junction object between Student & Training, depending on your schema).
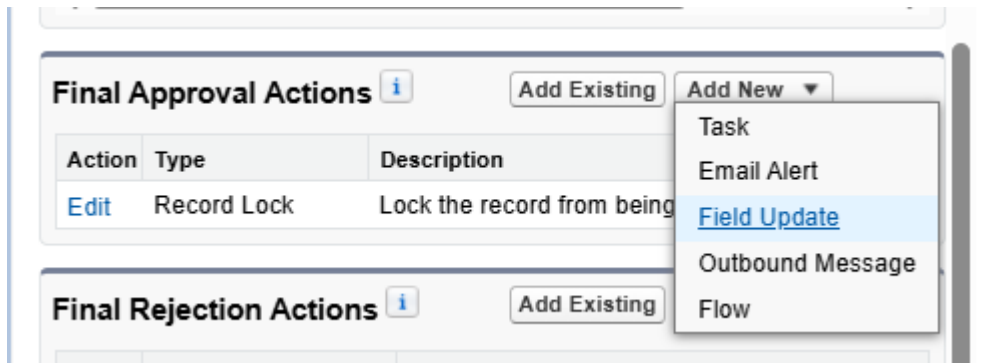
- Set Field Values:
- `Student__c` = `{!$Record.Student__c}`
- `Training__c` = "Full-Stack Bootcamp Test" (pick your default Training record).
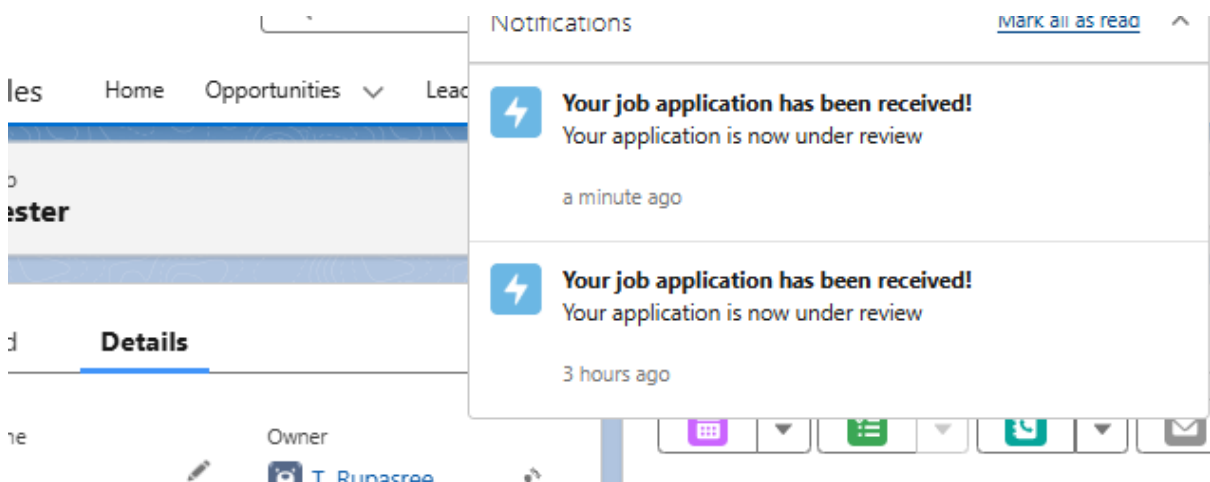- Save.
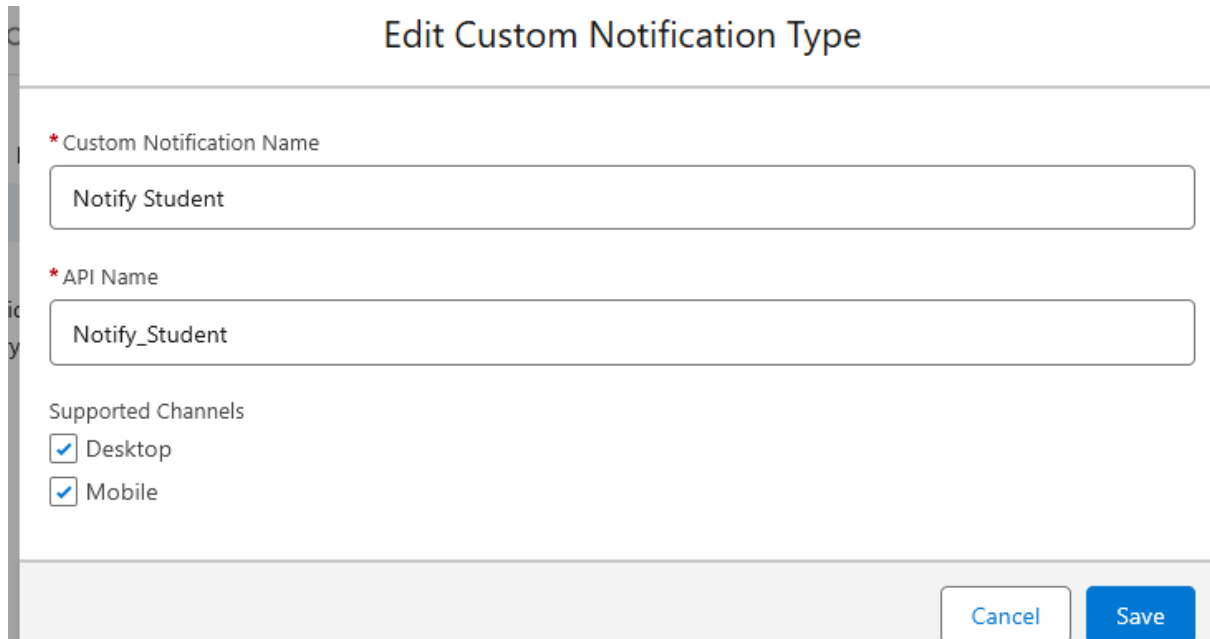
3. **Add Action: Send Notification**
- Click **+** → **Action**.
- Action Type = **Send Custom Notification**.
- Label = `Notify Student`.
- Notification Type = `Application Status Changed`.
- Recipient = `{!$Record.Student__c}`.
- Title = `Congrats!`
- Body = `You have been enrolled in the training program.`
- Save.

4. **Save & Activate Flow**
- Flow Label: `Auto_Enroll_Student_On_Approval`.
- Save → **Activate**.

◆ Testing

1. Go to **Job Applications** → Pick one with Status = Pending Review.
2. Update Status = **Approved** → Save.
3. Check:
- Student auto-enrolled in Training (new Training Enrollment record created).
- Student receives Notification in Salesforce (bell icon).

🪟 Email Alerts

- o Used to send automated emails for different actions:
- o Job Application submitted → Employer notified.
- o Training Program approved → Trainer notified.
- o Application accepted → Student notified.

Objective

- o To **send automatic email notifications** to users when specific conditions are met.
- o In our project use case (Skill Development Employment Portal):

- When a **Candidate is shortlisted for a Job**, send them an automated email.
- When a **Job Application is Rejected**, send a rejection email.
- When a **Training Enrollment is Approved**, send confirmation to the Candidate.

◆ Procedure

Step 1: Create an Email Template

1. In Salesforce Setup, go to **App Launcher → Search "Email Templates" → New Email Template**.
2. Enter details:
- **Email Template Name**: *Candidate Shortlist Notification*
- **Subject**: *Congratulations! You have been shortlisted*
- **Email Body**:
- `Hello {!Candidate__c.Name},`
-
- `Congratulations! You have been shortlisted for the job: {!Job__c.Name}.`
- `Please log in to the portal for next steps.`
-
- `Regards,`
- `Skill Development Employment Portal Team`
- Save.

New Email Template

* = Required Information

Information

* Email Template Name

Candidate Shortlist Notification

Related Entity Type

Job ▼

Description

Folder

Public Email

Templates

Select Folder

Message Content

Subject

: Congratulations! You have been shortliste

Enhanced Letterhead

Search Enhanced Letterheads... 🔍

HTML Value

- Hello {!Candidate__c.Name},
-
- Congratulations! You have been shortlisted for the job: {!Job__c.Name}.
- Please log in to the portal for next steps.
-

Cancel   Save

Step 2: Create an Email Alert

1. In Setup, search **Email Alerts** → **New Email Alert**.
2. Fill details:
- **Description**: *Notify Candidate when Shortlisted*
- **Email Template**: *Candidate Shortlist Notificat*
- *ion*

- **Recipient Type**: Select **Related User → Candidate Email field**
- Save.

SETUP
Email Alerts

Email Alert
Notify Candidate when Shortlisted

Help for this Page

Rules Using This Email Alert [0] | Approval Processes Using This Email Alert [0] | Entitlement Processes Using This Email Alert [0]

Email Alert Detail   Edit Delete Clone

Description	Notify Candidate when Shortlisted	Email Template	Application Status – To Student
Unique Name	Notify_Candidate_when_Shortlisted	Object	Candidate
From Email Address	Current User's email address		
Recipients	User: Candidate Youth		
Additional Emails			
Created By	T. Rupasree, 9/20/2025, 2:45 AM	Modified By	T. Rupasree, 9/20/2025, 2:45 AM

Edit Delete Clone

Rules Using This Email Alert   Rules Using This Email Alert Help ?

This alert is currently not used by any rules

Approval Processes Using This Email Alert   Approval Processes Using This Email Alert Help ?

This alert is currently not used by any approval processes

Entitlement Processes Using This Email Alert

This alert is currently not used by any entitlement processes

Flows Using This Email Alert

Flow Name	Version	Description	Object	Active

^ Back To Top   Always show me ▼ more records per related list

Step 3: Add Email Alert to Automation

You can attach this **Email Alert** to:

- **Workflow Rule** (when Job Application Status = Shortlisted)
- **Process Builder** (if we want more conditions, e.g., Skill Match ≥ 80%)
- **Flow Builder** (for advanced logic, e.g., send both Candidate & Recruiter emails).

Use Case

When a **Job Application's Status = Shortlisted**, the candidate automatically receives the **shortlist notification email**.

◆ Step-by-Step Procedure

Step 1: Go to Workflow Rules

1. In **Setup**, search for **Workflow Rules**.
2. Click **New Rule**.

Step 2: Choose Object

1. Select **Job Application (Custom Object)** (assuming you created this in earlier phases).
2. Click **Next**.

Step 3: Define Rule Criteria

1. Rule Name: **Shortlist Notification Rule**
2. Evaluation Criteria: **created, and every time it's edited**
3. Rule Criteria:
- Field: **Status**
- Operator: **equals**
- Value: **Shortlisted**
4. Save & Next.

Step 4: Add Workflow Action → Email Alert

1. Under **Immediate Workflow Actions**, click **Add Workflow Action → New Email Alert**.
2. Choose the previously created **Candidate Shortlist Notification Email Alert**.
3. Save.

Step 5: Activate the Workflow

1. Click **Activate**.
2. Done ✓

Workflow Rule
Shortlist Notification Rule

> ℹ️ Go with the flow! With Flow Builder, the future of low-code automation, you can do everything you do with workflow rules — and more! Salesforce plans to retire workflow rules and recommends building automation in Flow Builder. <u>Tell Me More</u> | <u>Migrate your workflow rules to flows</u>

Workflow Rule Detail   [Edit] [Delete] [Clone] [Activate]

Rule Name	Shortlist Notification Rule	Object	J [Activate]
Active	☐	Evaluation Criteria	Evaluate the rule when a record is created, and every time it's edited
Description			
Rule Criteria	Job: Application Status EQUALS Shortlisted		
Created By	T. Rupasree, 9/20/2025, 2:48 AM	Modified By	T. Rupasree, 9/20/2025, 2:48 AM

Workflow Actions   [Edit]

Immediate Workflow Actions

Type	Description
Email Alert	Notify Candidate when Shortlisted

Time-Dependent Workflow Actions   See an example

> ℹ️ You cannot add time-dependent workflow actions because your evaluation criteria is "Every time a record is created or edited". Change Evaluation Criteria

---

◆ Workflow Flow Diagram

```
Job Application Status = Shortlisted
        ↓
Workflow Rule Triggers
        ↓
Email Alert → Sends "You are shortlisted" email to Candidate
```

Now, whenever a recruiter updates a candidate's application to **Shortlisted**, the system will send an **automated email** 🎉

🗂️Field Updates

- Automated field updates to reduce manual work:
- Application Status auto-updates from **"Pending"** → **"Under Review"** → **"Accepted/Rejected"**.
- Job Posting status auto-updates to **"Expired"** after the closing date.

Objective

Field Updates allow us to **automatically update a field's value** when certain conditions are met.
This reduces manual work and ensures data consistency.

◆ Use Case in Project

In the **Skill Development Employment Portal**, let's implement:

- When a **Job Application Status = Selected**, automatically update the **Placement Status** field on the Candidate record to **Placed**.

This way, admins and recruiters don't need to manually mark a candidate as "Placed."

◆ Step-by-Step Procedure
Step 1: Go to Workflow Rules
1. In **Setup**, search for **Workflow Rules**.

2. Click **New Rule**.
3. Select **Job Application (Custom Object)**.
4. Click **Next**.

Step 2: Define Rule Criteria

1. Rule Name: **Placement Update Rule**
2. Evaluation Criteria: **created, and every time it's edited**
3. Rule Criteria:
- Field: **Status**
- Operator: **equals**
- Value: **Selected**
4. Save & Next.

Step 3: Add Workflow Action → Field Update

1. Under **Immediate Workflow Actions**, click **Add Workflow Action → New Field Update**.
2. Action Name: **Update Candidate Placement Status**
3. Field to Update: **Candidate → Placement Status**
4. New Value: **Placed**
5. Save.

Step 4: Activate the Workflow

1. Click **Activate**.
2. Done ✓

◆ Workflow Flow Diagram

```
Job Application Status = Selected
            ↓
Workflow Rule Triggers
            ↓
Field Update → Candidate.Placement_Status = "Placed"
```

This ensures **automatic synchronization** between job application status and candidate placement status.

We already saw this field updates in our earlier steps while creating workflow rules

🧩Tasks

- **Automated Task creation** for Admins and Trainers:
- Admin gets a task to review new Job Postings.
- Trainer gets a task to prepare study materials when a new Training Session is created.

Objective

Tasks allow Salesforce to **automatically create to-do items** for users (Recruiters, Admins, Trainers, etc.) when specific conditions are met.
This ensures **no important step is missed**.

◆ Use Case in Project

In the **Skill Development Employment Portal**, let's implement:

- When a **Candidate applies for a Job**, automatically create a **Task for the Recruiter** to review the application.

  This helps recruiters stay on top of new applications.

◆ Step-by-Step Procedure

Step 1: Go to Workflow Rules
1. In **Setup**, search **Workflow Rules**.
2. Click **New Rule**.
3. Select **Job Application (Custom Object)**.
4. Click **Next**.

Step 2: Define Rule Criteria
1. Rule Name: **Candidate Application Task Rule**
2. Evaluation Criteria: **created**
3. Rule Criteria:
- Field: **Status**
- Operator: **equals**
- Value: **Applied**
4. Save & Next.

Step 3: Add Workflow Action → New Task
1. Under **Immediate Workflow Actions**, click **Add Workflow Action → New Task**.

Edit Rule Candidate Application Task Rule

Step 3: Specify Workflow Actions

Specify the workflow actions that will be triggered when the rule criteria are met. See an exam

Rule Criteria   Job: Application Status EQUALS Applied

Evaluation Criteria   Evaluate the rule when a record is created

Immediate Workflow Actions

No workflow actions have been added.

Add Actions ▼
- New Task
- New Email Alert
- New Field Update
- New Outbound Message
- New Flow
- Select Existing Action

2. Fill details:
- **Assigned To**: Recruiter (Related User or specific role)
- **Subject**: *Review New Candidate Application*
- **Due Date**: Rule Trigger Date + 2 Days
- **Priority**: High
- **Status**: Not Started
3. Save.

SETUP
Tasks

New Task

Configure Task

Save Save & New Cancel

Create a task to associate with one or more workflow rules, approval processes, or entitlement processes. When changing a task, any modifications will apply to all rules, approvals, or entitlement processes associated with it.

Edit Task   = Required Information

Object	Job	Status	Not Started ▾
Assigned To	Recruiter	Priority	High ▾
Subject	Review New Candidate App		
Unique Name	Review_New_Candidate_A ⓘ		
Due Date	Rule Trigger Date ▾ plus ▾ 2		
	days		
Protected Component	☐		

Description Information

Comments

Step 4: Activate the Workflow

1. Click **Activate**.
2. Done ✓

◆ Workflow Flow Diagram

```
Job Application Created (Status = Applied)
            ↓
Workflow Rule Triggers
            ↓
Task Created for Recruiter: "Review Candidate Application"
```

This ensures that **every application gets recruiter attention** without delay.

🔔 Custom Notifications

- **In-App Notifications** were configured for real-time alerts:
- Student receives a notification when their application is shortlisted.
- Employer gets a notification when a new candidate applies.

Objective

Custom Notifications allow Salesforce to **send real-time alerts** directly inside the Salesforce app (web or mobile), instead of only through email.
This keeps users (Recruiters, Admins, Trainers, Candidates) informed immediately.

◆ Use Case in Project

In the **Skill Development Employment Portal**, let's implement:

- When a **Candidate is Placed in a Job**, send a **Custom Notification** to the Recruiter and Admin saying:
  "Candidate [Name] has been successfully placed in [Job Title]."

This makes the platform more interactive and responsive.

◆ Step-by-Step Procedure
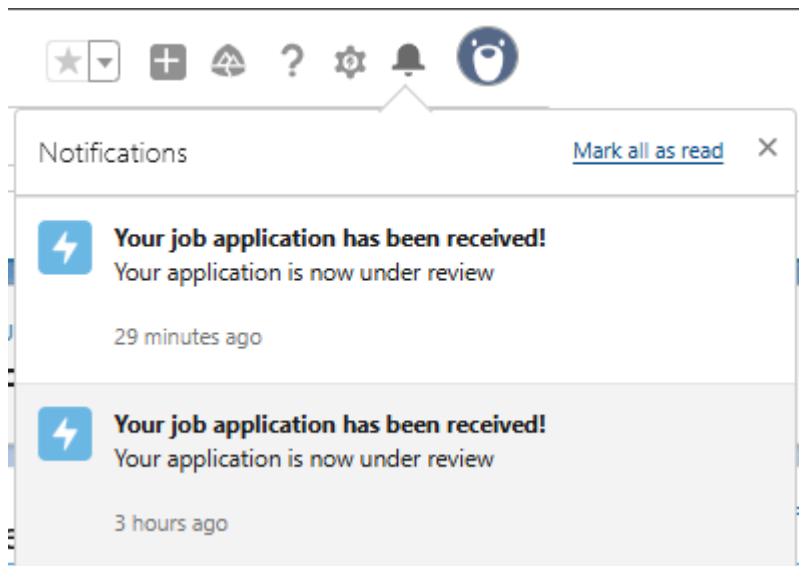
Step 1: Enable Custom Notifications
1. Go to **Setup** → Search **Notification Builder** → Click **Notification Types**.
2. Click **New**.
- **Label**: *Placement Success Notification*
- **API Name**: Placement_Success_Notification
- **Supported Channels**: Desktop + Mobile
3. Save.

Step 2: Create Process/Flow to Trigger Notification

1. Go to **Setup** → **Flow Builder** (recommended instead of Workflow for Custom Notifications).
2. Create a **Record-Triggered Flow** on **Job Application (Custom Object)**.
3. Entry Criteria:
- Field: **Status**
- Operator: **equals**
- Value: **Selected**
4. Add an **Action** → **Send Custom Notification**.
- Notification Type: *Placement Success Notification*
- Recipient: **Recruiter (Related User)** + **Admin Role**
- Notification Title: *Candidate Placement Update*
- Notification Body: *Candidate {!Candidate__c.Name} has been placed in Job {!Job__c.Name}.*
5. Save & Activate.

Step 3: Test It

1. Change a **Job Application Status** → **Selected**.
2. Recruiter & Admin should instantly see a **Salesforce Notification (bell icon top-right)**.

Already did in previous steps to test

◆ **Workflow Flow Diagram**

```
Job Application Status = Selected
        ↓
Record-Triggered Flow
        ↓
Custom Notification → Recruiter + Admin
```

Process Automation Phase (Admin) is complete:

- Validation Rules
- Workflow Rules
- Process Builder
- Approval Process
- Flow Builder
- Email Alerts
- Field Updates
- Tasks
- Custom Notifications

Benefits of Process Automation in the Project

- Reduced manual effort for Admins and Trainers.
- Faster response times for students and employers.
- Ensured **data accuracy** through validation rules.
- Improved **user experience** with timely notifications.
- Standardized approval flows to maintain quality.