RPA Test Project Report

APRIL 23

Authored by: Rakshit Bhadoria

Table of Contents

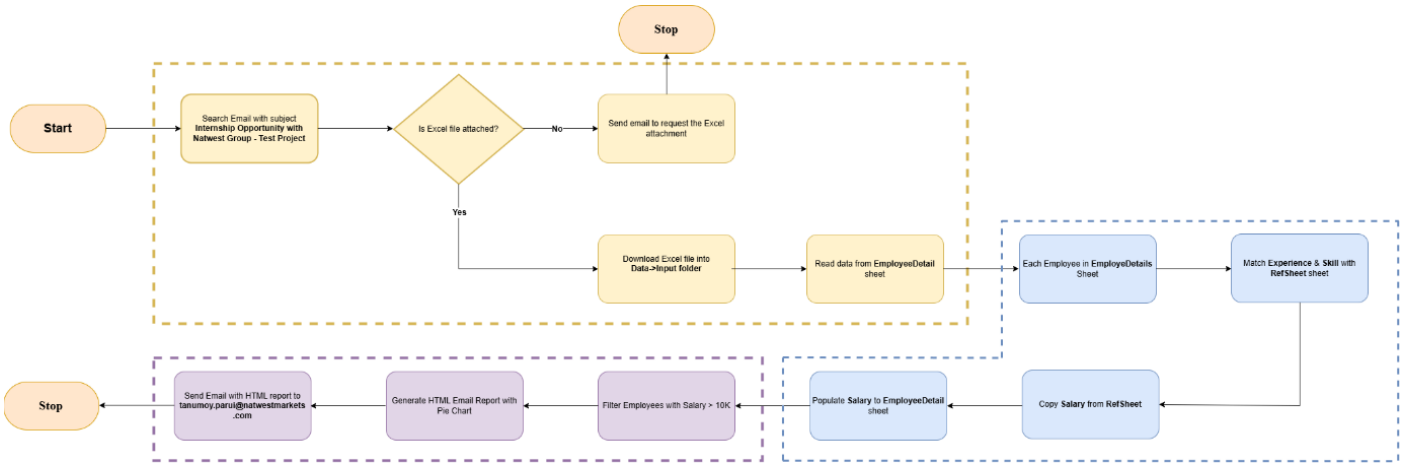
Project Objective

The goal of this RPA project is to automate the process of **updating employee salaries** based on their **experience and skillset**, generate a visual report for employees with **salary greater than 10K**, and send this report via email to the concerned stakeholder. Additionally, the bot is to be **deployed in UiPath Orchestrator** and scheduled to **run twice a week**.

This automated solution is developed using **REFramework**.

Task 1: HLSD

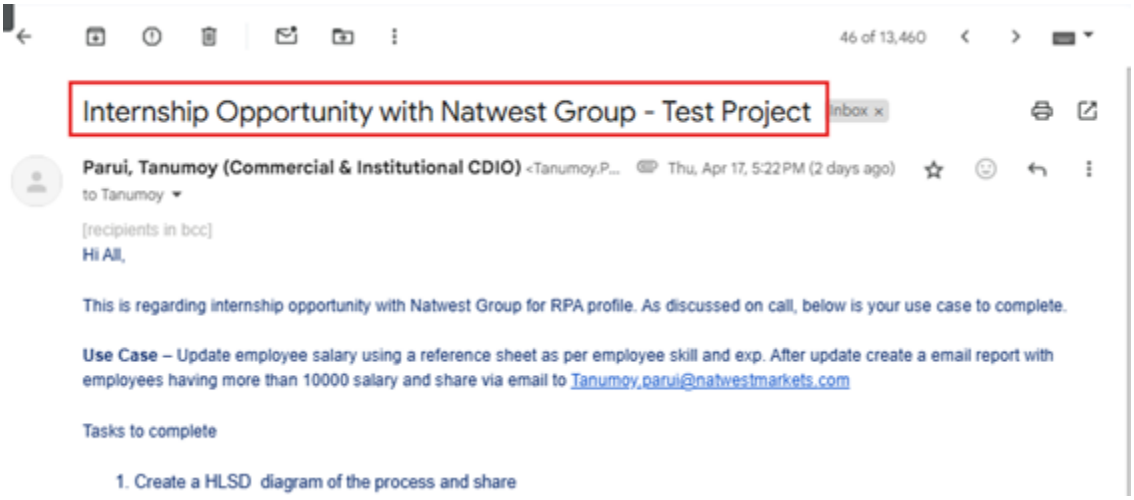
A **High-Level Solution Design (HLSD)** was created to visualize the end-to-end automation process highlights the core steps including reading input, processing logic, report generation, and emailing



Process:

1. **Search for Email**

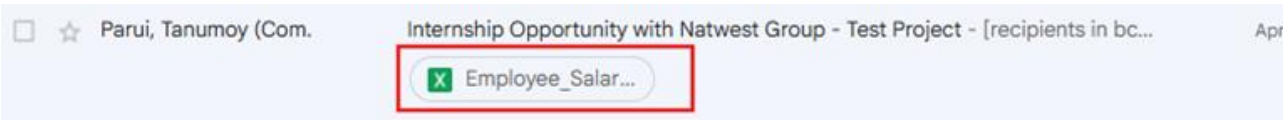
   Search in mail inbox for an email with the subject **"Internship Opportunity with Natwest Group - Test Project".**



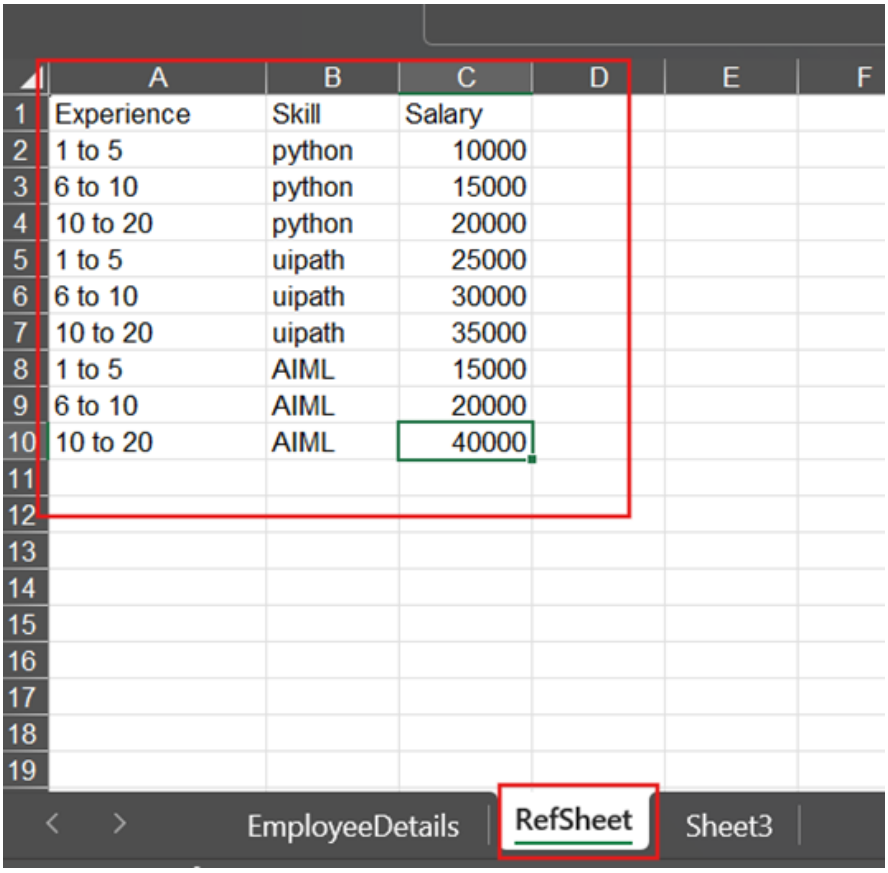
2. **Check Excel Attachment**

   Verify whether the email has the necessary Excel file attached:

   - If yes, save the file to the **Data → Input folder.**
   - If not, halt the process by sending a follow-up email asking for the attachment.



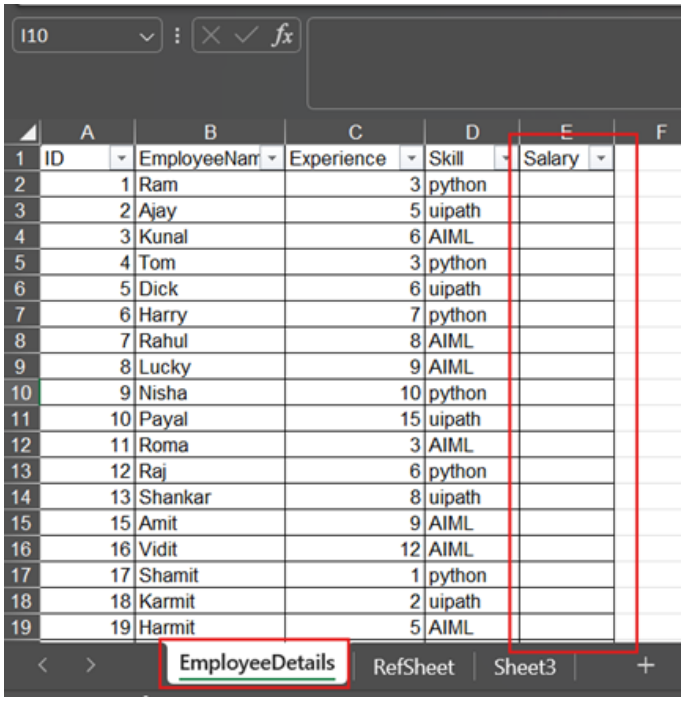
3. **Update Salary**
    - Compare the experience and skill of each employee (in EmployeeDetail sheet) with t RefSheet sheet (Screenshot 1).



	A	B	C	D	E	F
1	Experience	Skill	Salary			
2	1 to 5	python	10000			
3	6 to 10	python	15000			
4	10 to 20	python	20000			
5	1 to 5	uipath	25000			
6	6 to 10	uipath	30000			
7	10 to 20	uipath	35000			
8	1 to 5	AIML	15000			
9	6 to 10	AIML	20000			
10	10 to 20	AIML	40000			
11						
12						
13						
14						
15						
16						
17						
18						
19						

EmployeeDetails | RefSheet | Sheet3

    - Populate the salary in the Salary column (in EmployeeDetail sheet, screenshot 2) afte obtaining it from the RefSheet.



	A	B	C	D	E	F
1	ID	EmployeeNam	Experience	Skill	Salary	
2	1	Ram	3	python		
3	2	Ajay	5	uipath		
4	3	Kunal	6	AIML		
5	4	Tom	3	python		
6	5	Dick	6	uipath		
7	6	Harry	7	python		
8	7	Rahul	8	AIML		
9	8	Lucky	9	AIML		
10	9	Nisha	10	python		
11	10	Payal	15	uipath		
12	11	Roma	3	AIML		
13	12	Raj	6	python		
14	13	Shankar	8	uipath		
15	15	Amit	9	AIML		
16	16	Vidit	12	AIML		
17	17	Shamit	1	python		
18	18	Karmit	2	uipath		
19	19	Harmit	5	AIML		

EmployeeDetails | RefSheet | Sheet3

4. **Filter Employee – Have More than 10K Salary**

   Employees with salaries over 10K should be filtered out from **EmployeeDetail** Sheet and saved for reporting.

	A	B	C	D	E	F
1	ID	EmployeeName	Experience	Skill	Salary	
2	1	Ram	3	python	10000	
3	2	Ajay	5	uipath	25000	
4	3	Kunal	6	AIML	20000	
5	4	Tom	3	python	10000	
6	5	Dick	6	uipath	30000	
7	6	Harry	7	python	15000	
8	7	Rahul	8	AIML	20000	
9	8	Lucky	9	AIML	20000	
10	9	Nisha	10	python	20000	
11	10	Payal	15	uipath	35000	
12	11	Roma	3	AIML	15000	
13	12	Raj	6	python	15000	
14	13	Shankar	8	uipath	30000	
15	15	Amit	9	AIML	20000	
16	16	Vidit	12	AIML	40000	
17	17	Shamit	1	python	10000	
18	18	Karmit	2	uipath	25000	
19	19	Harmit	5	AIML	15000	

EmployeeDetails | RefSheet | Sheet3

5. **Generate HTML Report and send email**
    - Make a pie chart in an HTML report that displays the salary distribution of workers making above 10K.
    - Use the SMTP/Outlook to send the HTML report to tanumoy.parui@natwestmarkets.com via email.

Succes | Updated Salary Report Inbox ×

Rakshit Bhadoria <RakshitBhadoria67@gmail.com> Fri, Apr 18, 4:55 PM (20 hours ago)
to me

Hello,

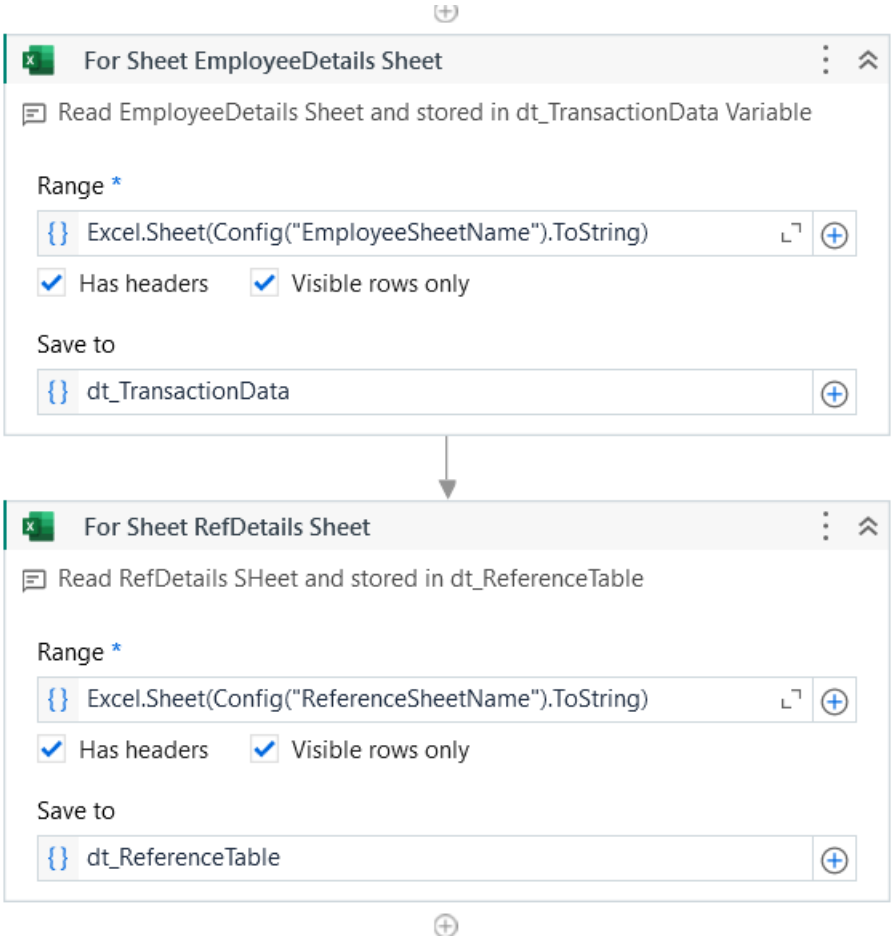
The automation task has been completed successfully.
Below is a summary of employees whose updated salary exceeds ₹10,000:

ID	EmployeeName	Experience	Skill	Salary
1	Ram	3	python	10000
2	Ajay	5	uipath	25000
3	Kunal	6	AIML	20000
4	Tom	3	python	10000
5	Dick	6	uipath	30000
6	Harry	7	python	15000
7	Rahul	8	AIML	20000
8	Lucky	9	AIML	20000
9	Nisha	10	python	20000
10	Payal	15	uipath	35000
11	Roma	3	AIML	15000
12	Raj	6	python	15000
13	Shankar	8	uipath	30000
15	Amit	9	AIML	20000
16	Vidit	12	AIML	40000
17	Shamit	1	python	10000
18	Karmit	2	uipath	25000
19	Harmit	5	AIML	15000
20	Deepa	6	uipath	30000

Task 2: Read Data from Excel

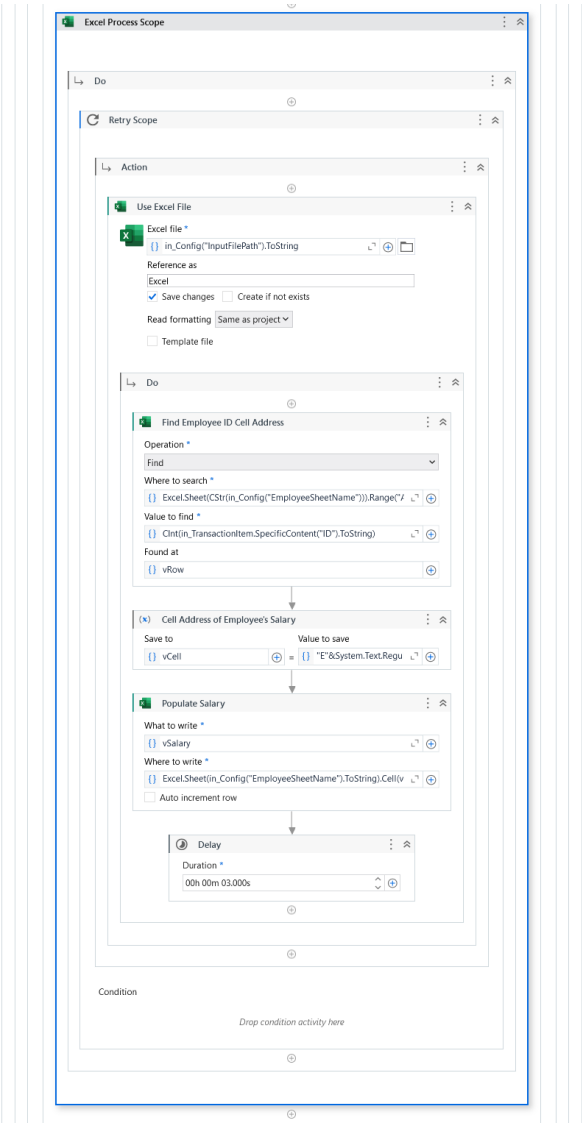
The automation begins by reading two sheets from the Excel file:

- **employeeDetails Sheet**: Contains ID, Name, Experience, Skill, and (initially blank) Salary
- **RefDetail Sheet:** Maps experience ranges and skills to respective salary slabs
- Used **UiPath's Excel Application Scope and Read Range** activities.



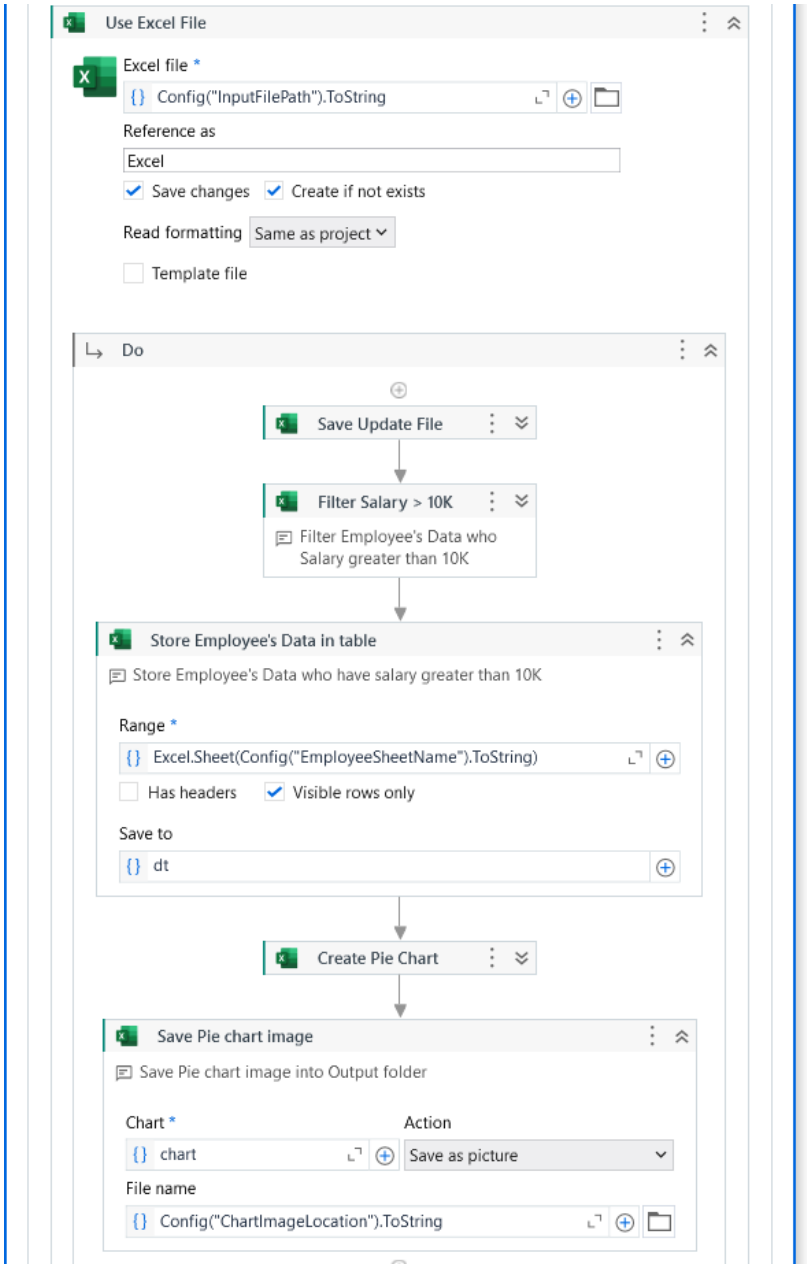
Task 3: Update Salary in Excel (employeeDetails sheet)

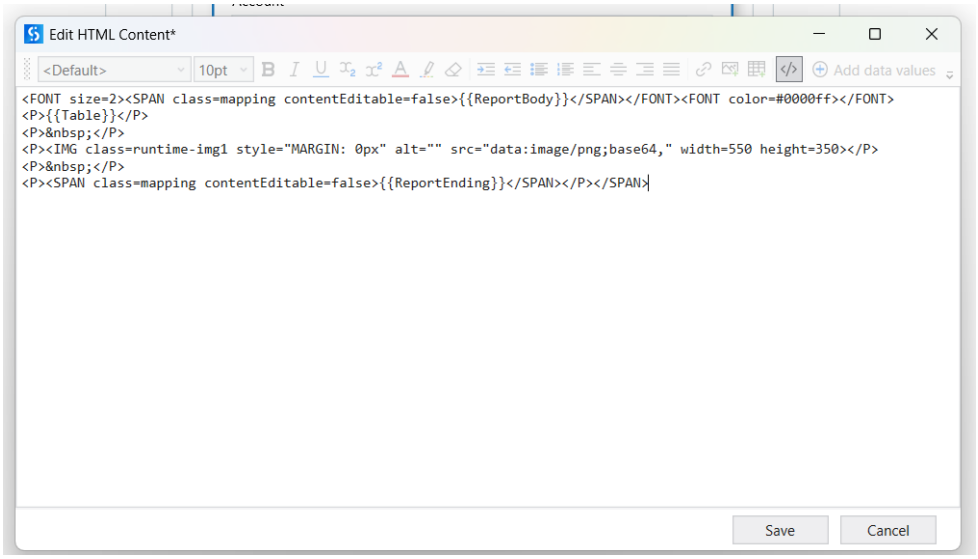
The Each employee’s experience and skill was matched against the reference sheet logic:



Task 4: Generate HTML Report (Salary > 10K)

Employees with salaries above **10K** were filtered into a report. **Embedded pie chart using Excel.**



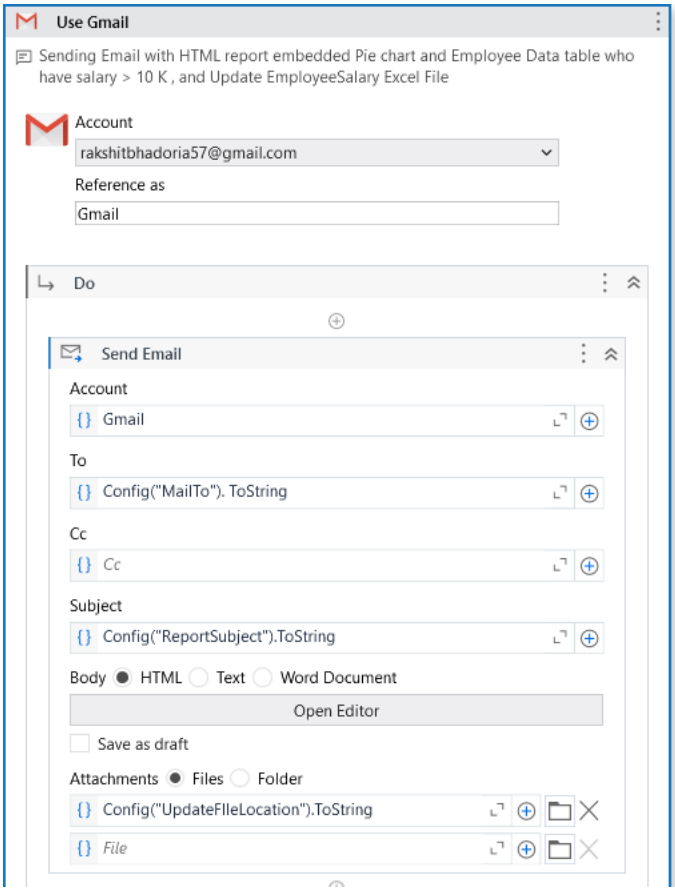

Edit HTML Content*
<Default>
10pt
Add data values
<FONT size=2><SPAN class=mapping contentEditable=false>{{ReportBody}}</SPAN></FONT><FONT color=#0000ff></FONT>
<P>{{Table}}</P>
<P> </P>
<P><IMG class=runtime-img1 style="MARGIN: 0px" alt="" src="" width=550 height=350></P>
<P> </P>
<P><SPAN class=mapping contentEditable=false>{{ReportEnding}}</SPAN></P></SPAN>
Save
Cancel

Task 5: Send Report via Email

The report was sent using UiPath's Send Use Gmail activity.

Email Configuration:

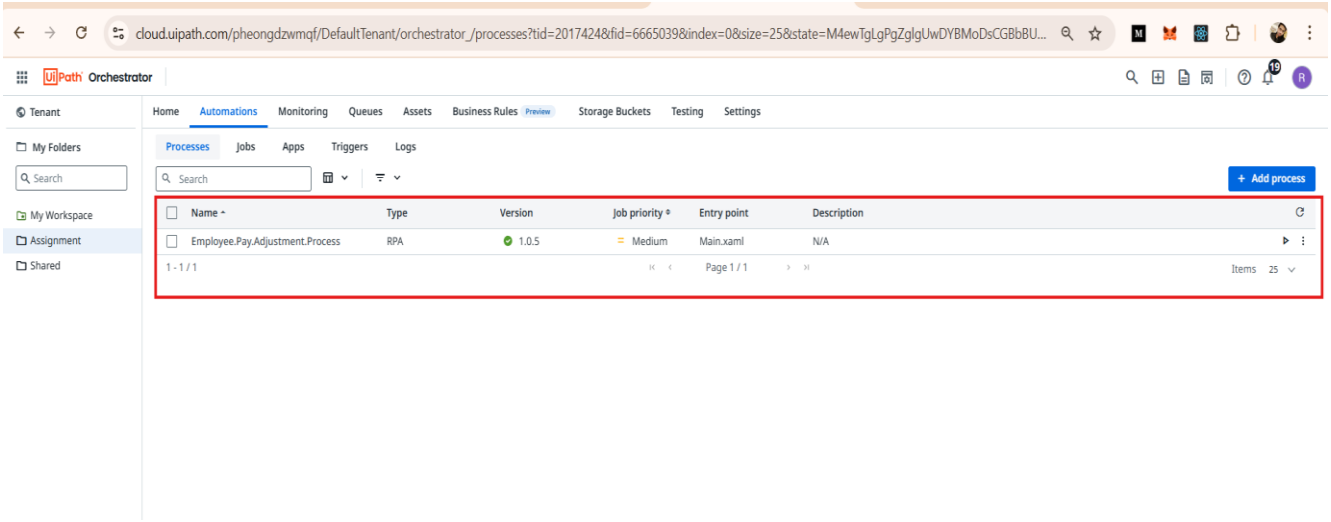
- **To:** Tanumoy.parui@natwestmarkets.com
- **Attachment:** Updated EmployeeSalary File



Task 6: Deploy Process in Orchestrator

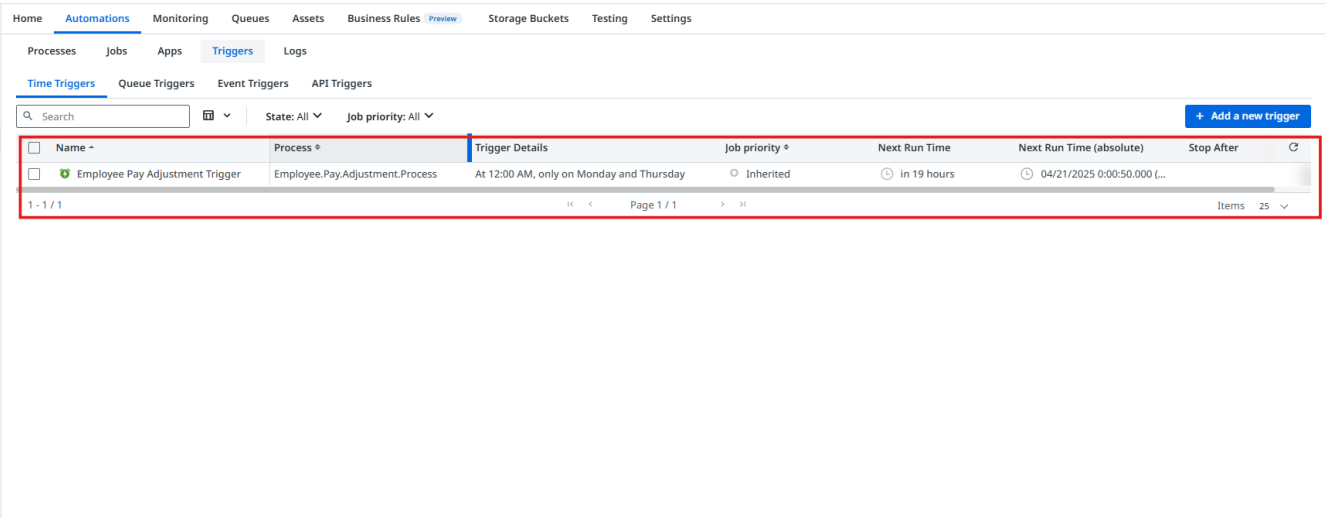
The workflow was successfully published from **UiPath Studio to UiPath Orchestrator**.

- **Process name:** Employee.Pay.Adjustment.Process



Task 7: Create Trigger (2 Times Per Week)

- **Trigger Frequency:** Twice a week (every Monday & Thursday at 12 AM)
- **Trigger Type:** Time-based recurring trigger



Files, Screenshots & Repository

PDD	PDD.pdf
HLSD Diagram	HLSD.png
Trigger_Screenshot	Trigger_Screenshot.png
Process_Deployment_Screenshot	Process_Deployment_Screenshot.png
Project Code	Source Code