



AUTOMATING EMAIL WORKFLOWS: INTEGRATING UIPATH, OUTLOOK, AND POWER AUTOMATE CLOUD FLOWS FOR SEAMLESS EMAIL MONITORING AND UNATTENDED BOT DEPLOYMENT

Sai Madhur Potturu

Robotics Center of Excellence (CoE)

Zoetis Inc.

Parsippany, United States

ABSTRACT

Robotic Process Automation (RPA) has emerged as a transformative technology, enabling businesses to streamline operations and enhance productivity. UiPath, a prominent RPA and Intelligent Automation platform, has played a pivotal role in this landscape. With a user-friendly interface, AI-driven capabilities, and a robust ecosystem, UiPath empowers organizations to automate repetitive tasks and drive digital transformation.

This research paper explores the integration of UiPath platform with Outlook and Power Automate to enable continuous monitoring of shared mailbox folders and deploy unattended bots in response to incoming emails.

This integration is achieved by leveraging the UiPath Start Jobs API, allowing different systems to trigger UiPath bots. The paper discusses two distinct methods for achieving this integration, highlighting the benefits and applications of the solution in various organizational processes. The first approach involves setting up VBA code in Outlook to monitor shared mailbox folders and trigger the UiPath Start Jobs API. This ensures the deployment of unattended bots when emails arrive in designated folders. The second approach utilizes Power Automate cloud flows to monitor shared mailbox folders and deploy unattended bots using the UiPath Start Jobs API.

The integration of Outlook, Power Automate, and UiPath provides several advantages, including timely responses to customer inquiries, improved automation communication, and increased efficiency in email-based processes. Moreover, it optimizes the usage of unattended bot runners, enabling the deployment of scalable and high ROI automation solutions on these runners.

By following the outlined steps and leveraging the capabilities of these tools, organizations can streamline their email-based processes, enhance productivity, and achieve greater automation efficiency.

The integration of UiPath with Outlook and Power Automate provides a comprehensive solution for continuous email monitoring and efficient utilization of automation resources.

Keywords: Robotic Process Automation (RPA), UiPath, Start Jobs API, Orchestrator APIs, VBA, Outlook, Power Automate Cloud Flows, Bot Deployment.

Cite this Article: Sai Madhur Potturu, Automating Email Workflows: Integrating Uipath, Outlook, and Power Automate Cloud Flows for Seamless Email Monitoring and Unattended Bot Deployment, International Journal of Robotics Research (IJRR), 1(1), 2023, pp. 9–25.

<https://iaeme.com/Home/issue/IJRR?Volume=1&Issue=1>

I. INTRODUCTION

UiPath is a leading Robotic Process Automation (RPA) and Intelligent Automation platform that revolutionizes business operations by automating repetitive tasks, freeing up human resources for more strategic endeavors. Its intuitive interface, AI-powered intelligence, and comprehensive offerings enable seamless process automation and scalability, while prioritizing data security and privacy. UiPath boasts a vast and thriving community of users, developers, and partners, creating a rich ecosystem for knowledge sharing, collaboration, and support. The platform's extensive offerings encompass a wide range of automation solutions, catering to various industries and business functions, making it a versatile and comprehensive tool for organizations seeking to optimize their processes and embrace digital transformation [1][2].

A common use of RPA involves triggering UiPath bots to automate tasks when an email arrives in an Outlook shared mailbox. UiPath's Integration Services simplify the automation of third-party apps by handling authorization, authentication, and API connections, making integration into SaaS platforms quicker. The three components of the Integration Service are Connectors, Connections, and Triggers [3].

The Triggers provide a mechanism for subscribing to specific events from third-party applications, giving the flexibility to automatically start a bot in Orchestrator. Automations can be initiated based on events in connected systems by using triggers [4]. UiPath offers the Microsoft Outlook 365 connector and various triggers to monitor events in Outlook in the user's primary mailbox and deploy a bot based on an event [5]. However, the Microsoft Outlook 365 connector currently lacks a trigger to monitor folders within Outlook shared mailboxes and deploy an unattended bot on a bot runner when an email arrives in a specific folder.

This article explores two distinct mechanisms for triggering UiPath bots. The first mechanism involves configuring VBA code in Outlook to monitor folders within shared mailboxes [6][7] and trigger the UiPath bot using the Start Jobs API [8][9][10]. This ensures the deployment of unattended bots when emails arrive in designated folders. The second approach utilizes Power Automate cloud flows [11] to monitor shared mailbox folders [12] and deploy unattended bots using the UiPath Start Jobs API. This integration enables continuous monitoring of folders in shared mailboxes and facilitates the deployment of a UiPath bot on an unattended bot runner using the Start Jobs API when an email arrives in a designated folder within a shared mailbox.

The UiPath Start Jobs API is a powerful REST API provided by UiPath, which serves as a fundamental tool for deploying and managing UiPath robots and automation processes within the UiPath platform. This API allows organizations to trigger the execution of automation jobs programmatically, offering a seamless integration between UiPath and external applications or systems.

Through these integration methods, the need for developing and scheduling a separate bot exclusively for frequent email monitoring tasks is eliminated. This integration optimizes the utilization of the unattended Bot runner, allowing for the deployment and scalability of high ROI automation solutions. This integration enhances efficiency, enables timely responses to customer inquiries, and improves overall automation communication.

II. SOLUTION

In this solution, we explore two distinct methods for continuous email monitoring and deploying unattended bots in UiPath. The first method shows how to trigger an unattended UiPath bot using VBA code in Outlook, using the UiPath Start Jobs API when an email arrives in a shared Outlook mailbox folder. The second method demonstrates triggering an Unattended UiPath bot using Power Automate cloud flows and the Start Jobs API.

A. Outlook and UiPath Integration

In this method, VBA code is set up in the users Outlook Windows application to monitor the shared mailbox folders and trigger an UiPath bot, utilizing the Start Jobs API.

1) Prerequisites:

a) Outlook windows Application

- The users outlook windows application session should be active and running for the VBA code to execute and respond to events such as receiving new emails in the shared mailbox.
- The user should have access to shared mailbox.

b) Automation Cloud Orchestrator APIs

UiPath Automation Cloud has two mechanisms for consuming APIs [13].

- getting the API access information from Automation Cloud.
- registering an external application to use the OAuth flow.

This solution utilizes the API access information from the Automation cloud to trigger the bots. To retrieve the API access information from the automation cloud, perform the steps below.

- Log in to your Automation Cloud account.
- Navigate to “Admin > Tenants.” The Tenants page lists all existing tenants.
- Select the Tenant to display its available services.
- Click “API Access ()” for the Orchestrator service. The API Access window opens and includes the following service-specific information:
- User Key - allows you to generate unique login keys to be used with APIs or with 3rd party applications to log in and perform actions on your behalf. This was previously known as your refresh token.
- Organization ID - your organization name. It is the name after the base URL (that is, <https://cloud.uipath.com/>).
- Tenant Name - the display name of the tenant.
- Client Id - specific to the Orchestrator application itself, is the same for all users and tenants on a specific platform. For example, all the tenants on cloud.uipath.com have the same Client Id value.

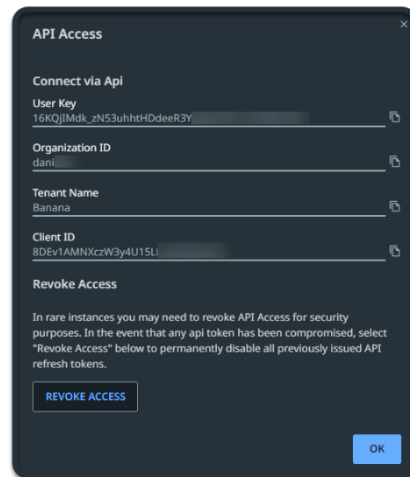


Figure 1. API Access Information [13]

2) VBA Code functionality in Outlook:

VBA code is setup in outlook for monitoring emails in a shared mailbox and trigger UiPath bot using Start Jobs API.

a) Steps for setting up the VBA code

- Open Outlook and press Alt + F11 or click Developer tab then click Visual Basic to open the VBA editor [7].

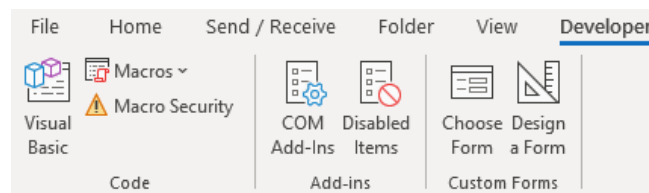


Figure 2. Visual Basic in Developer tab

- Enable References, Microsoft WinHTTP Services, version 5.1, and Microsoft XML, V6.0. These references allow the VBA code to trigger HTTP requests.

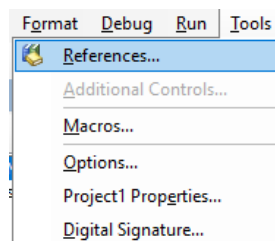


Figure 3. References in Tools tab

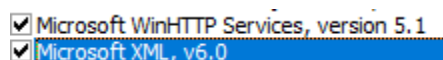


Figure 4. References

- In the Project Explorer window, locate and expand the project (Project 1).

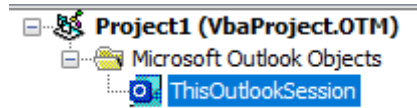


Figure 5. Project window

- Double-click “ThisOutlookSession” to open the code module for Outlook.
- In the code window, write the VBA code.

b) VBA Code and explanation

- Declare Private variables to access them with in the module.

// objInboxItems monitors shared mailboxes' inbox folder for any new emails. The WithEvents enables the object to respond to events (ItemAdd), which is triggered when a new item is added to the collection (Outlook.Items).

```
Private WithEvents objInboxItems As Outlook.Items
```

// Declare objProcessedFolder variable to hold a reference to an Outlook folder

```
Private objProcessedFolder As Outlook.Folder
```

- The "Application_Startup" Private Subroutine is triggered upon starting the Outlook session in Windows. It initializes the necessary objects and sets up the event handler to monitor the Inbox of the shared mailbox.

```
Private Sub Application_Startup()
```

//Reference Outlook MAPI name space

```
Dim objNamespace As Outlook.NameSpace
```

//Reference to access shared mailbox and its contents

```
Dim objSharedMailbox As Outlook.Recipient
```

//Reference Inbox folder

```
Dim objInboxFolder As Outlook.Folder
```

```
Set objNamespace = Application.GetNamespace("MAPI")
```

// Replace {Sharedmail_Box_Address} placeholder with your shared mailbox box address

```
Set objSharedMailbox = objNamespace.CreateRecipient("{Sharedmail_Box_Address}")
```

//Reference and get the Inbox folder of the shared mailbox

```
Set objInboxFolder = objNamespace.GetSharedDefaultFolder(objSharedMailbox, olFolderInbox)
```

```
Set objInboxItems = objInboxFolder.Items
```

```
End Sub
```

- The Private Sub objInboxItems_ItemAdd event handler is triggered whenever a new item (email) is added to the Inbox folder of the shared mailbox.

```
Private Sub ObjInboxItems_ItemAdd(ByVal Item As Object)
```

//Check if the received item is a MailItem

```
If TypeOf Item Is Outlook.MailItem Then
```

```
//Call StartUiPathProcess function to trigger a bot
```

```
    Call StartUiPathProcess
```

```
End If
```

```
End Sub
```

- The 'StartUiPathProcess' function is 'called' within the objInboxItems_ItemAdd subroutine to trigger an unattended bot in the bot runner VDI using the UiPath Start Jobs API.

```
Private Function StartUiPathProcess ()
```

```
//Call AuthenticateTenant function
```

```
    Dim vAccessToken As String
```

```
    vAccessToken = AuthenticateTenant()
```

```
//Call TriggerBot function and pass Access Token as input. This function triggers the API and deploys the specified bot on specified unattended bot user and machine.
```

```
    Call TriggerBot(vAccessToken)
```

```
End Function
```

- The 'AuthenticateTenant' function is called with in the 'StartUiPathProcess' function to authenticate the Automation Cloud Orchestrator Tenant with the tenant's API access information and generate an authorization token.

```
Private Function AuthenticateTenant () As String
```

```
//Declare Variables
```

```
    Dim vPostRequest As New MSXML2.ServerXMLHTTP60
```

```
    Dim vBody As String
```

```
    Dim AuthenticationURL As String
```

```
    Dim AuthenticationAPIResponse As String
```

```
    Dim AuthorizationToken As String
```

```
//Authentication API URL
```

```
    AuthenticationURL = "https://account.uipath.com/oauth/token"
```

```
// Authenticate API headers.
```

```
vPostRequest.Open "POST", AuthenticationURL, False
```

```
vPostRequest.setRequestHeader "Content-Type", "application/json"
```

```
vPostRequest.setRequestHeader "X-UIPATH-TenantName", "{Tenant}" //Replace {Tenant} place holder with the actual tenant's name.
```

```
// The API access information, client ID, and User key retrieved from the Automation Cloud will be formatted in JSON within the Body of the POST method [13].
```

```
// Replace {client_ID}, and {user_key} place holders with the actual Client ID and User Key from API access information
```

```
vBody = "{""grant_type"": ""refresh_token"", ""client_id"": ""{client_ID}"" , ""refresh_token"": ""{user_key}""}"
```

```
//POST Request
```

```
vPostRequest.Send vBody
```

```
//POST request response
```

```
AuthenticationAPIResponse = vPostRequest.ResponseText
```

```
// Parse JSON response to retrieve access token.
```

```
AuthorizationToken = "Bearer " + Mid(AuthenticationAPIResponse,  
InStr(AuthenticationAPIResponse, "access_token") + 15, InStr(AuthenticationAPIResponse,  
"id_token") - 21)
```

```
//Return token as function output
```

```
AuthenticateTenant = AuthorizationToken
```

```
End Function
```

- The 'TriggerBot' function is called with in the 'StartUiPathProcess' function to trigger an RPA bot using UiPath Start Jobs API.

```
Private Function TriggerBot (Token As String)
```

```
//Declare variables
```

```
Dim vPostRequest As New MSXML2.ServerXMLHTTP60
```

```
Dim vBody As String
```

```
Dim StartJobsAPIURL As String
```

```
Dim StartJobsAPIResponse As String
```

```
Dim vTemp As String
```

```
Dim vProcessId As String
```

```
//Start Jobs API URL
```

```
//Replace the {Org} and {Tenant} placeholders in the URL with the actual organization name and  
tenant name.
```

```
StartJobsAPIURL =
```

```
"https://cloud.uipath.com/{Org}/{Tenant}/odata/Jobs/UiPath.Server.Configuration.OData.StartJobs"
```

```
//POST Request Headers
```

```
vPostRequest.Open "POST", StartJobsAPIURL, False
```

```
vPostRequest.SetRequestHeader "Content-Type", "application/json"
```

```
vPostRequest.SetRequestHeader "X-UIPATH-TenantName", "{Tenant}" //Replace {Tenant}  
placeholder with the actual tenant's name.
```

```
vPostRequest.SetRequestHeader "X-UIPATH-OrganizationUnitId", "{Folder_ID}" //Replace the  
{Folder_ID} place holder with the actual orchestrator folder ID. To retrieve the folder's ID, click on  
the folder in the UiPath Orchestrator. The folder ID is displayed after "fid=" in the orchestrator URL.
```

```
vPostRequest.SetRequestHeader "Authorization", Token
```

```
//POST Request Body Schema
```

// To deploy a UiPath process on an unattended bot runner/machine using the Start Jobs API, constant parameters: Tenant Name, Modern Folder ID, Process Release key, Robot ID, and Machine ID are required. Once retrieved, these parameters remain unchanged and can be reused for repeated process deployment [9][10][14][15].

```
vBody = "{""startInfo"":{""ReleaseKey"":
""{Process_Release_key}"",""JobsCount"":1,""Strategy"":
""ModernJobsCount"",""MachineRobots"": [ {""RobotId"": {Robot_ID}, ""MachineId"":
{Machine_ID}} ]}"
//Replace {Process_Release_Key}, {Robot_ID}, and {Machine_ID} place holders with the actual
Process Release Key, Robot ID, and Machine ID.
//POST Request
vPostRequest.Send vBody
//POST request response
StartJobsAPIResponse = vPostRequest.ResponseText
//Retrieve Process ID from the JSON response. Process ID can be used to further monitor the status of
the process.
vTemp = Right(StartJobsAPIResponse, Len(StartJobsAPIResponse) –
InStrRev(StartJobsAPIResponse, "Id") – 3)
vProcessId = Left(vTemp, Len(vTemp) – 3)
End Function
```

- When the API request call is successful, the response code “201” is returned. The API response includes a “value” array where the value of the "Id" object represents the Process deployment ID. This ID is used to check the bot deployment status.

B. Power Automate and UiPath Integration

In this method, a power automate flow is setup for the O365 user to monitor the shared mailbox folders and trigger an unattended UiPath bot, utilizing the UiPath Start Jobs API.

1) Prerequisites

a) Power Automate Cloud flows

- The user should have access to Power Automate cloud flows and Premium connectors [16].
- The user should have a Premium license to use the HTTP connector in power automate [17].

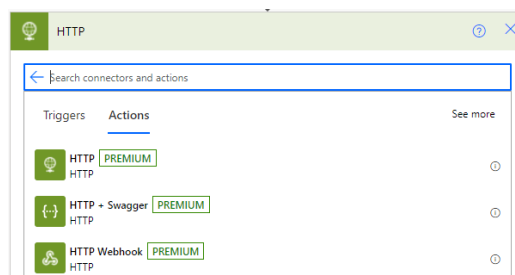


Figure 6. HTTP operation in Power Automate cloud flows

b) Automation Cloud Orchestrator APIs

The requirements for using the Automation Cloud Orchestrator APIs are identical to those detailed in the "Outlook and UiPath Integration" section.

2) Power Automate Cloud flows functionality

a) Setup up an “Automated cloud flow”

- Create a new flow and select “Automated Cloud flow” under “My flows” tab [18].

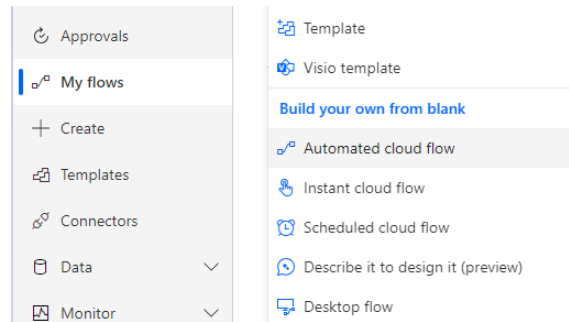


Figure 7. Select Automated cloud flow

- Define name for the flow and select “When a new email arrives in a shared mailbox” trigger.

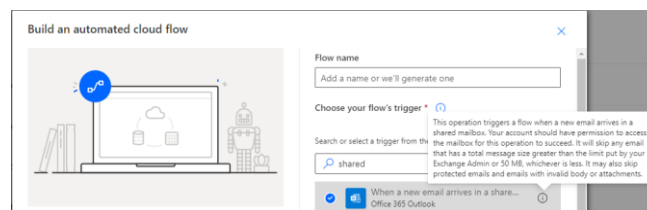


Figure 8. Provide name and select trigger

- Click create to create and start developing the cloud flow.



Figure 9. Create cloud flow

b) Power Automate Cloud flow development steps

- Select Shared mailbox and Folder.

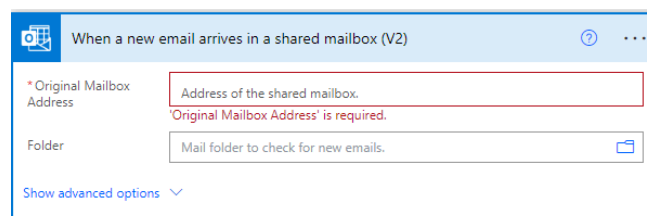


Figure 10. Configure email trigger

- Add Step, choose HTTP operation and select HTTP action to call UiPath Authentication API.

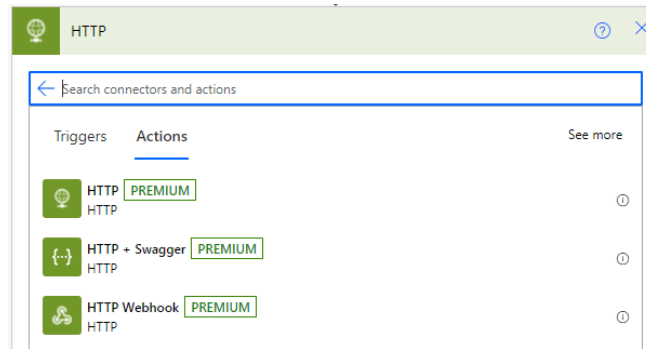


Figure 11. HTTP operation and action

- Select POST method in the HTTP action.



Figure 12. Select method for authentication

- Enter UiPath Authentication URL, “https://account.uipath.com/oauth/token”.



Figure 13. Authentication API URL

- Headers:

Content-Type: Set the “Content-Type” header value as “application/json”.

X-UIPATH-TenantName: Set the value for the "X-UIPATH-TenantName" header as your tenant's name.

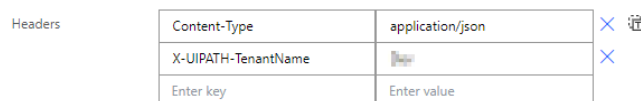


Figure 14. Authentication API headers

- Body: The API access information, client ID, and User key retrieved from the Automation Cloud will be formatted in JSON within the Body of the POST method.

```
{
  "grant_type": "refresh_token",
  "client_id": "{client_ID}", // Replace with the Client ID from API access information
  "refresh_token": "{user_key}" // Replace with the User Key from API Access information
}
```

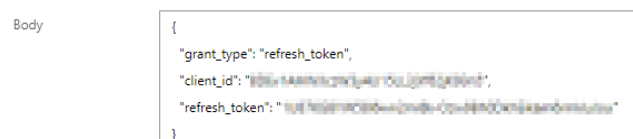


Figure 15. Authentication API body

Automating Email Workflows: Integrating Uiopath, Outlook, And Power Automate Cloud Flows for Seamless Email Monitoring and Unattended Bot Deployment

- Save the flow, click “Test” button, and send a test email to shared mailbox to test the flow. The flow will be activated upon receiving the email in the designated folder within the shared mailbox. It will then execute the HTTP action, calling the Authentication API and generating a JSON response. Copy this JSON output to a notepad for further reference.

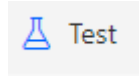


Figure 16. Test button

- Edit the flow and add “Data Operation” step. Select Parse JSON action in Data operation.

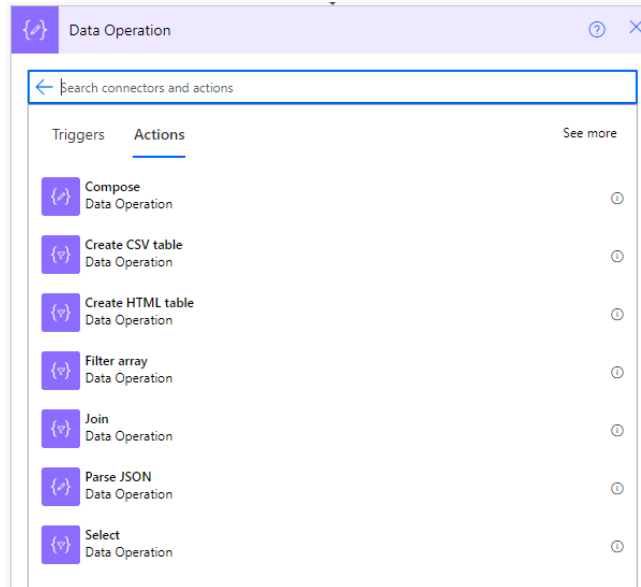


Figure 17. Parse JSON

- Select “Body” (output) from previous HTTP action as “Content” for parse JSON action. This step will retrieve authorization/access token from Authentication API JSON response.

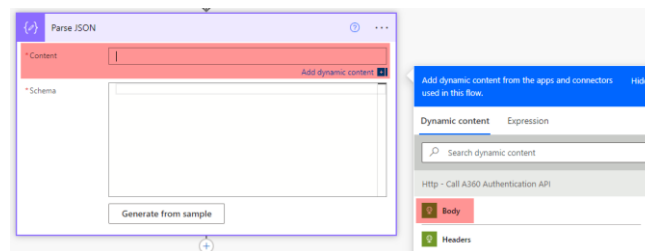


Figure 18. Parse JSON Contet

- Click “Generate from Sample” button.



Figure 19. Generate from sample

- Paste the previously saved Authentication API JSON output from HTTP step on to the “Insert a sample JSON payload” window and click done. This will generate JSON schema to parse the JSON and get hold of “Token” [19].

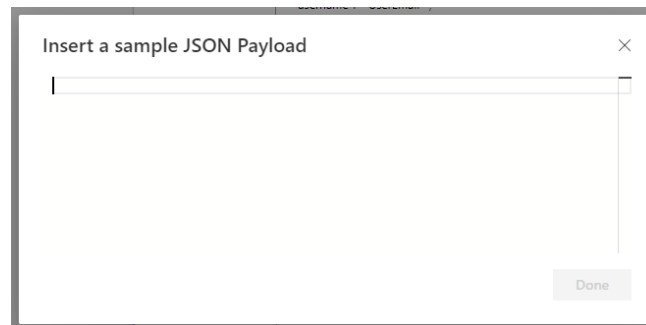


Figure 20. Insert JSON payload

- Add another step and select HTTP operation and HTTP action to call Start Jobs API.
- Select POST method in HTTP action.



Figure 21. Select method for deployment

- URL: The Start Jobs API URL is "https://cloud.uipath.com/{Org}/{Tenant}/odata/Jobs/UiPath.Server.Configuration.OData.StartJobs ". Replace the {Org} and {Tenant} placeholders in the URL with the actual organization name and tenant name.

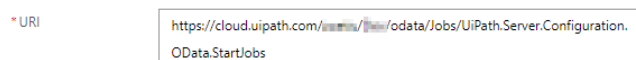


Figure 22. Start Jobs API URL

- Enter Headers, Content-type, X-UIPATH-TenantName, X-UIPATH- OrganizationUnitId, and Authorization. The authorization token retrieved from the previous step; "Parse JSON" is passed as an input for the "Authorization" header.
- Content-Type: Set the "Content-Type" header value as "application/json"
- X-UIPATH-TenantName: Set the value for the "X-UIPATH-TenantName" header as your tenant's name.
- X-UIPATH- OrganizationUnitId: Assign the value of the Orchestrator folder ID to the "X-UIPATH-OrganizationUnitId" header. To retrieve the folder's ID, click on the folder in the UiPath Orchestrator. The folder ID is displayed after "fid=" in the orchestrator URL.

Headers	
Content-Type	application/json
X-UIPATH-TenantName	
Authorization	Bearer access_token
X-UIPATH-OrganizationUnitId	
Enter key	Enter value

Figure 23. Start Jobs API headers

- To deploy a UiPath process on an unattended bot runner/machine using the Start Jobs API, constant parameters: Tenant Name, Modern Folder ID, Process Release key, Robot ID, and Machine ID are required. Once retrieved, these parameters remain unchanged and can be reused for repeated process deployment [9][10][14][15].
- Body: The JSON body schema including the Process Key, Robot ID, and Machine ID is as follows [20].

{

```
"startInfo": {  
  "ReleaseKey": "{Process Release Key}", // Replace with the actual process release key."  
  "JobsCount": 1,  
  "Strategy": "ModernJobsCount",  
  "MachineRobots": [  
    {  
      "RobotId": {Unattended Robot ID}, // Replace with the actual Unattended Robot ID.  
      "MachineId": {Machine ID} // Replace with the actual Machine ID.  
    }  
  ]  
}
```

Body

```
{  
  "startInfo": {  
    "ReleaseKey": "11111111-1111-1111-1111-111111111111",  
    "JobsCount": 1,  
    "Strategy": "ModernJobsCount",  
    "MachineRobots": [  
      {  
        "RobotId": "11111",  
        "MachineId": "11111"  
      }  
    ]  
  }  
}
```

Figure 24. Start Jobs API Body

- When the API request call is successful, the response code “201” is returned. The API response includes a “value” array where the value of the "Id" object represents the Process deployment ID. This ID is used to check the bot deployment status.
- Save the flow and click run to test the functionality of the flow.

III. APPLICATIONS OF THE SOLUTION IN VARIOUS ORGANIZATIONAL PROCESSES:

The solution of integrating UiPath with Outlook and Power Automate to enable continuous monitoring of shared mailbox folders and deploy unattended bots in response to incoming emails has broad applications across various organizational processes. Here are some use cases of this solution in different departments and processes [2][6][21][22][23].

A. Customer Support and Ticketing

The solution can be applied to automate customer support processes. Bots can monitor a shared mailbox for customer inquiries, automatically categorize and prioritize them, and trigger appropriate actions or responses. This can lead to faster response times and improved customer satisfaction.

B. Data Entry and Processing

Many business processes involve data entry and processing. By integrating UiPath with Outlook and Power Automate, organizations can automate tasks like data extraction from incoming emails, updating databases, and generating reports. This can significantly reduce manual data entry efforts and minimize errors.

C. Notification and Alerts

Organizations can use the solution to set up automated notifications and alerts based on specific email events. For example, a bot could monitor emails from a monitoring system and trigger alerts when certain conditions are met, such as server downtime or critical events.

D. Workflow Orchestration

The solution can serve as a foundation for orchestrating complex workflows that involve multiple systems and stakeholders. Bots can initiate processes in response to emails, interact with other systems through APIs, and ensure seamless execution of end-to-end workflows.

E. HR and Employee Onboarding

Organizations can automate HR processes such as employee onboarding by using the solution to trigger bots that assign tasks to different departments, and update employee records in various systems.

F. Finance and Expense Processing

The solution can be used to automate finance-related processes, such as expense report processing. Bots can monitor shared mailboxes for expense reports, validate receipts, calculate reimbursements, and update financial systems.

These are just a few examples of how the integrated solution can be applied to various organizational processes across different departments. The versatility of the solution allows organizations to streamline their email-based workflows, enhance productivity, and achieve greater automation efficiency in a wide range of business functions.

IV. BENEFITS OF THE SOLUTION

This integration has several benefits that enhance efficiency and effectiveness in various organizational processes. Here are the key benefits of the presented solution [2][6][24].

A. Efficiency and Productivity

By automating processes through UiPath, Outlook, and Power Automate integration, organizations can achieve higher levels of efficiency and productivity. Tasks that previously required manual intervention can now be executed automatically and swiftly, leading to faster turnaround times and reduced processing delays.

B. Real-time Updates:

The solution enables real-time updates to systems and databases based on specific events, such as incoming emails. This ensures that information is current and accurate, leading to better decision-making [8].

C. Faster Response Times:

The ability to trigger automated actions in response to specific events, such as customer inquiries or low inventory levels, ensures timely responses and actions. This can lead to improved customer satisfaction and better operational performance.

D. Customization and Flexibility:

The solution can be tailored to suit specific business needs and workflows, ensuring that automation aligns with the organization's unique requirements.

E. Cross-System Integration:

The solution enables seamless integration between different systems and applications, promoting data flow and communication across the organization [8].

V. CONCLUSION

In conclusion, this research paper has explored and presented a robust solution for bridging the gap in Robotic Process Automation (RPA) by enabling continuous monitoring of shared mailbox folders and deploying unattended bots in UiPath for enhanced efficiency and timely actions. The integration of UiPath with Outlook and Power Automate offers significant benefits and opens new possibilities for organizations seeking to optimize their automation processes.

By leveraging VBA code in Outlook and Power Automate cloud flows, the proposed solution eliminates the limitations of UiPath in monitoring shared mailbox folders and responding to incoming emails. The seamless integration between these platforms ensures that unattended bots are efficiently deployed, enabling organizations to respond promptly to customer inquiries and automate a wide range of email-based tasks.

Furthermore, the optimization of unattended bot runners ensures cost-effectiveness and scalability in automation initiatives. With the ability to deploy high Return on Investment (ROI) automation solutions on demand, organizations can achieve significant cost savings and streamline their operations.

Overall, the integrated solution presented in this research paper offers a comprehensive and user-friendly approach to continuous email monitoring and unattended bot deployment in UiPath. By following the outlined steps and leveraging the capabilities of Outlook, Power Automate, and UiPath, organizations can unlock the full potential of RPA, enhance their email-based workflows, and achieve greater automation efficiency.

Conflict of Interest Statement

The author, Sai Madhur Potturu, is employed at Zoetis Inc., specifically in the Robotics Center of Excellence (CoE) department. Zoetis Inc. is a company that provides animal healthcare products and services. The development and implementation of the digital solution presented in this manuscript align with the author's role and responsibilities within the organization. The author declares no financial or personal relationships that may have influenced the content or findings presented in this manuscript.

Data Availability Statement

The data used to support the findings of this study are available from the corresponding author upon reasonable request. The data include the PowerApps application design, RPA solution implementation details, and relevant datasets used for testing and evaluation. Access to the data will be provided to researchers or individuals to replicate the study findings or conduct further analyses related to the presented digital solution.

REFERENCES

- [1] Tripathi, A. M. (2018). Learning Robotic Process Automation: Create Software robots and automate business processes with the leading RPA tool–UiPath. Packt Publishing Ltd.
- [2] Mullakara, N., & Asokan, A. K. (2020). Robotic process automation projects: build real-world RPA solutions using UiPath and automation anywhere. Packt Publishing Ltd.
- [3] UiPath (n.d.). Integration Service User Guide. Docs.Uipath.com. <https://docs.uipath.com/integration-service/automation-cloud/latest/user-guide/introduction>
- [4] UiPath (n.d.). Triggers. Docs.Uipath.com. <https://docs.uipath.com/integration-service/automation-cloud/latest/user-guide/triggers>
- [5] UiPath (n.d.). Microsoft Outlook 365 events. Docs.Uipath.com. <https://docs.uipath.com/integration-service/automation-cloud/latest/user-guide/uipath-microsoft-outlook365-events>
- [6] Sai Madhur Potturu, Streamlining Email Automation: Seamless Integration of Automation 360 with Outlook and Power Automate Cloud Flows for Continuous Email Monitoring and Unattended Bot Deployment, International Journal of Computer Engineering and Technology (IJCET), 14(2), 2023, pp. 97-111 doi: <https://doi.org/10.17605/OSF.IO/TKAGV>
- [7] Data Numen (n.d.). How to Run VBA Code in Your Outlook. Wwww.Datanumen.com. <https://www.datanumen.com/blogs/run-vba-code-outlook/>
- [8] UiPath (n.d.). UiPath Add-ins Guide. Postman.Uipath.Rocks. <https://postman.uipath.rocks/#intro>
- [9] UiPath (n.d.). Start a Job using Orchestrator API. Forum.Uipath.com. <https://forum.uipath.com/t/start-a-job-using-orchestrator-api/307208>
- [10] UiPath (n.d.). Start A Job On Specific Machine Using Modern Folder Via API Call. Forum.Uipath.com. <https://forum.uipath.com/t/start-a-job-on-specific-machine-using-modern-folder-via-api-call/503009/1>
- [11] Microsoft (n.d.). Overview of cloud flows. Learn.Microsoft.com. <https://learn.microsoft.com/en-us/power-automate/overview-cloud>
- [12] Microsoft (n.d.). Overview of using Outlook and Power Automate. Learn.Microsoft.com. <https://learn.microsoft.com/en-us/power-automate/email-overview>
- [13] UiPath (n.d.). Consuming Cloud API. Docs.Uipath.com. <https://docs.uipath.com/orchestrator/automation-cloud/latest/api-guide/consuming-cloud-api>
- [14] [UiPath Video Tutorials Made by Cristian Negulescu]. (2020, December 3). Postman read process result from UiPath Orchestrator (REST API) [Video]. Wwww.Youtube.com. <https://www.youtube.com/watch?v=WWBwxstGtgM>
- [15] [UiPath Video Tutorials Made by Cristian Negulescu]. (2020, March 11). Start UiPath Process from Postman (Orchestrator REST API) [Video]. Wwww.Youtube.com. https://www.youtube.com/watch?v=84Wlzn1CK_Y&list=PLtmJ1hI2wYBl8ch2k4bwB_2I_Fiu1t5bn&t=6s
- [16] Microsoft (n.d.). List of all Premium tier connectors. Learn.Microsoft.com. <https://learn.microsoft.com/en-us/connectors/connector-reference/connector-reference-premium-connectors>
- [17] Microsoft (n.d.). Introduction. Learn.Microsoft.com. <https://learn.microsoft.com/en-us/training/modules/http-connectors/1-introduction>
- [18] Microsoft (n.d.). Create a cloud flow in Power Automate. Learn.Microsoft.com. <https://learn.microsoft.com/en-us/power-automate/get-started-logic-flow>

- [19] tomriha. (n.d.). Where to get Schema for the Parse JSON action in Power Automate. Tomriha.com. <https://tomriha.com/where-do-i-get-schema-for-the-parse-json-action-in-power-automate/>
- [20] UiPath (n.d.). Start job process via API in Modern Folders - The job's associated process could not be found. Forum.Uipath.com. <https://forum.uipath.com/t/start-job-process-via-api-in-modern-folders-the-jobs-associated-process-could-not-be-found/478483>
- [21] Sai Madhur Potturu. Revolutionizing Field Data Collection and Management Using PowerApps and Robotics Process Automation (RPA), Software Engineering. Volume 10, Issue 1, March 2023 , pp. 6-14. doi: 10.11648/j.se.20231001.12
- [22] beachnet (n.d.). The Benefits of Automation for Different Industries. Wwww.Beachnet.com. <https://www.beachnet.com/industries-automation-benefits/>
- [23] infomentum (n.d.). API+RPA. Wwww.Infomentum.com. <https://www.infomentum.com/api-rpa-automation-integration>
- [24] (n.d.). 7 Biggest Benefits of RPA (Robotic Process Automation). Wwww.Kofax.com. <https://www.kofax.com/learn/blog/benefits-of-rpa>

Citation: Article: Sai Madhur Potturu, Automating Email Workflows: Integrating UiPath, Outlook, and Power Automate Cloud Flows for Seamless Email Monitoring and Unattended Bot Deployment, International Journal of Robotics Research (IJRR), 1(1), 2023, pp. 9–25

Abstract: https://iaeme.com/Home/article_id/IJRR_01_01_002

Article Link:

https://iaeme.com/MasterAdmin/Journal_uploads/IJRR/VOLUME_1_ISSUE_1/IJRR_01_01_002.pdf

Copyright: © 2023 Authors. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

This work is licensed under a **Creative Commons Attribution 4.0 International License (CC BY 4.0)**.



 editor@iaeme.com