EMAIL HANDLING BOT

A PROJECT REPORT

Submitted by

MITHILESH T (220701165)

in partial fulfillment for the course

OAI1903 - INTRODUCTION TO ROBOTIC PROCESS AUTOMATION

for the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

RAJALAKSHMI NAGAR

THANDALAM

CHENNAI – 602 105

NOVEMBER 2024

RAJALAKSHMI ENGINEERING COLLEGE

CHENNAI - 602105

BONAFIDE CERTIFICATE

Certified that this project report **"EMAIL HANDLING BOT"** is the bonafide work of **"MITHILESH T (220701165)"** who carried out the project work for the subject OAI1903 - Introduction to Robotic Process Automation under my supervision.

SIGNATURE

MRS.G.M.SASIKALA , M.E ,
SUPERVISOR,
Assistant Professor,
Department of
Computer Science and Engineering,
Rajalakshmi Engineering College,
Rajalakshmi Nagar,
Thandalam,
Chennai – 602105.

Submitted to Project and Viva Voce Examination for the subject OAI1903 - Introduction to Robotic Process Automation held on _____.

Internal Examiner **External Examiner**

ABSTRACT

The Email Handling Bot is an innovative RPA solution aimed at automating and optimizing email management. It is designed to segregate emails based on predefined rules provided by the user in an Excel sheet. This project addresses the challenges of manually handling large volumes of emails, ensuring a more efficient and accurate process for organizing and managing correspondence.

The bot begins by extracting classification rules and criteria from the input Excel file, which could include parameters such as sender, subject keywords, date, or attachment type. It then accesses the email inbox, scans through messages, and applies these rules to categorize emails into designated folders, flag important ones, or forward them to specified recipients. The entire process operates in real time, ensuring that emails are handled promptly.

This automation significantly reduces the manual effort involved in email processing, eliminates the risk of errors, and saves valuable time. Additionally, it improves consistency and enables organizations to maintain streamlined communication workflows. The bot can be easily customized and scaled to accommodate various use cases, including customer service, sales, HR, or other operations requiring organized email handling.

By integrating RPA technology with tools like Excel and email platforms, the Email Handling Bot demonstrates the power of automation in simplifying routine tasks. It offers a user-friendly interface, high efficiency, and robust performance, making it an essential tool for businesses and individuals dealing with heavy email traffic. This solution not only enhances productivity but also ensures that critical emails are never missed, helping organizations maintain professionalism and operational efficiency.

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavour to put forth this report. Our sincere thanks to our Chairman **Mr. S. Meganathan, B.E, F.I.E.,** our Vice Chairman **Mr. Abhay Shankar Meganathan, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) Thangam Meganathan, Ph.D.,** for providing us with the requisite infrastructure and sincere endeavouring in educating us in their premier institution.

Our sincere thanks to **Dr. S.N. Murugesan, M.E., Ph.D.,** our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to **Dr. P. Revathy, M.E., Ph.D.,** Professor and Head of the Department of Computer Science and Design for her guidance and encouragement throughout the project work. We convey our sincere and deepest gratitude to our internal guides, **Mrs. Roxanna Samuel, M.E.,** Assistant Professor (SG), **Ms. Farjana, M.E.,** Assistant Professor (SG), **Ms. Vinothini, M.E.,** Assistant Professor (SG), Department of Computer Science and Engineering, Rajalakshmi Engineering College for their valuable guidance throughout the course of the project. We are very glad to thank our Project Coordinators, **Dr. N. Durai Murugan, M.E., Ph.D.,** Associate Professor, and **Mr. B. Bhuvaneswaran, M.E.,** Assistant Professor (SG), Department of Computer Science and Engineering for their useful tips during our review to build our project.

MITHILESH T (220701165)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATION	ACCRONYM
RPA	Robotic Process Automation
SMTP	Simple Mail Transfer Protocol
PDF	Portable Document Format
SQL	Structured Query Language
CSV	Comma-Separated Values

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The **Email Handling Bot** is designed to address these challenges through automation. By leveraging robotic process automation (RPA) technology, this bot automates the process of email segregation and management. It utilizes user-defined rules stored in an Excel sheet to classify and organize emails accurately and efficiently.

The bot extracts classification criteria such as sender, subject line, keywords, or attachments from the Excel sheet. It then scans the inbox, processes the emails, and performs actions like moving them to specific folders, flagging important ones, or forwarding them to the relevant recipients. This ensures that emails are handled systematically, without the need for manual intervention.

In today's fast-paced digital world, email remains a critical mode of communication for businesses and individuals alike. However, managing a high volume of emails efficiently can be a time-consuming and error-prone task. Manual processing of emails often leads to delays, missed messages, or mismanagement, which can negatively impact operations and communication.

This project aims to simplify email handling for businesses and individuals, reducing the time spent on repetitive tasks and enhancing accuracy. With its ability to adapt to various use cases, the **Email Handling Bot** provides a scalable and customizable solution for improving productivity and ensuring seamless communication management.

1.2 OBJECTIVE

The objective of the Email Handling Bot is to automate the tedious process of email management by categorizing and organizing emails based on user-defined rules stored in an Excel sheet. By reducing manual effort, enhancing accuracy, and ensuring timely handling of messages, the bot streamlines communication workflows. It is designed to improve productivity, minimize errors, and provide a scalable solution adaptable to various business needs, ensuring critical emails are never missed.

1.3 EXISTING SYSTEM

The existing system for email management relies heavily on manual effort, where users sort and organize emails individually, leading to inefficiencies and a high chance of errors. While basic filters in email platforms exist, they are limited in functionality and struggle with complex or dynamic criteria. This approach is time-consuming and inadequate for managing high volumes of emails, highlighting the need for an automated solution like the Email Handling Bot to improve efficiency and accuracy.

1.4 PROPOSED SYSTEM

The proposed Email Handling Bot is an automated solution designed to streamline email management by categorizing and processing emails based on predefined rules stored in an Excel sheet. The bot reads these rules, scans the inbox, and performs actions such as moving emails to specific folders, flagging important ones, or forwarding them to designated recipients. This system eliminates manual effort, reduces errors, and ensures timely processing of emails. It is scalable, customizable, and integrates seamlessly with existing email platforms, providing an efficient and reliable way to handle large volumes of emails and enhance productivity.

CHAPTER 2

LITERATURE REVIEW

2.1 Survey on Robotic Process Automation (RPA) in Email Management

Robotic Process Automation (RPA) is increasingly being adopted in email management to streamline repetitive and time-consuming tasks. RPA technology enhances productivity by automating activities such as email sorting, categorization, and response generation. Current studies highlight the benefits of RPA in improving operational efficiency and reducing the risk of human error in handling large volumes of emails. Despite these advancements, challenges persist, such as integrating RPA with existing email platforms and ensuring customization for diverse business requirements.

Relevant research papers related to RPA in email management include:

[1] A study demonstrates the application of RPA in automating repetitive communication tasks, emphasizing its ability to improve response times and reduce workload. The study highlights use cases in corporate and customer service domains where RPA ensures accurate and consistent email processing.

[2] Research published in IJITEE examines RPA's role in streamlining communication workflows, particularly in high-volume email environments. The findings indicate that RPA-based email solutions improve time management and enhance organizational efficiency.

2.2 Survey on Automation in Email Notification Systems

Automation in email notification systems leverages technologies such as RPA and email API integrations to deliver timely and accurate communication. Automated solutions eliminate manual intervention in sending notifications, thereby reducing administrative overhead and improving reliability. However, existing systems often face limitations in handling large-scale communication with personalization and maintaining system adaptability.

[1] A paper explores the effectiveness of automation in generating and delivering personalized email notifications based on preconfigured rules, such as deadlines or thresholds. It emphasizes the reduced workload and enhanced communication consistency achieved through automation. [2] Another study investigates the integration of RPA with email platforms to send batch communications. It concludes that automated systems significantly improve scalability, ensuring reliable and efficient messaging in large-scale operations.

2.3 Survey on Challenges in Email Management and Proposed Integratio**n** with RPA

Manual email handling poses significant challenges, including inefficiency, human error, and time delays in processing. Studies in this area emphasize the potential of RPA to address these issues through intelligent automation. By automating repetitive tasks like email filtering, forwarding, and responding, RPA reduces human dependency and enhances operational effectiveness.

[1] Research reviews traditional email handling methods, identifying challenges such as time inefficiencies and inconsistency in manual workflows. It advocates for the adoption of RPA to automate routine tasks

and enable seamless email management.
[2] A study highlights the limitations of current email systems, such as static rule-based filters, and underscores the advantages of RPA in delivering dynamic, rule-based automation that adapts to changing needs.

2.4 Summary of the Intersection of RPA and Email Management

The Email Handling Bot leverages RPA to automate the organization and processing of emails, addressing inefficiencies in manual email management. This system uses predefined rules from an Excel sheet to classify emails, perform actions like folder segregation, flagging, or forwarding, and send notifications—all with minimal manual intervention.

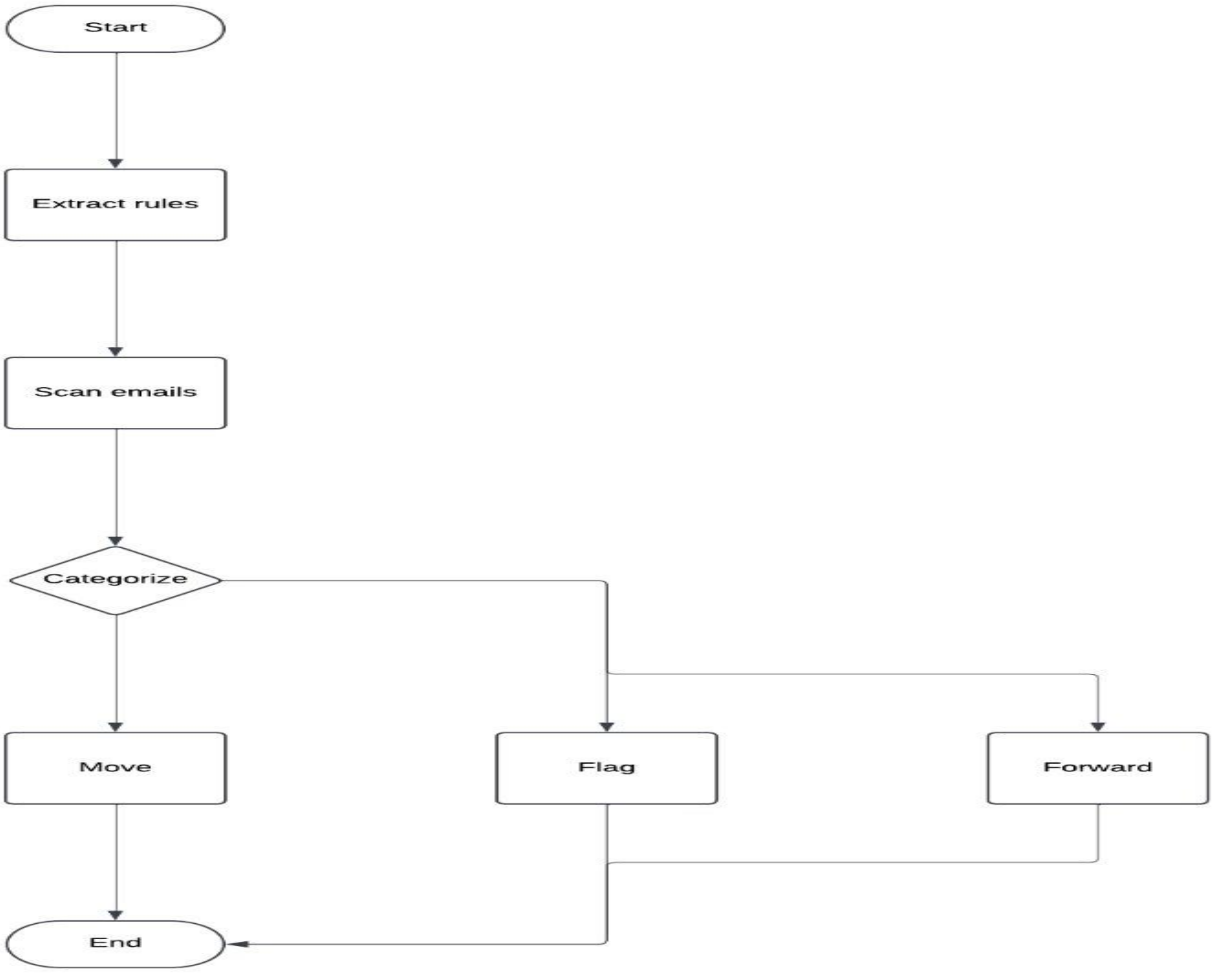
By bridging the gap between traditional email systems and modern automation, the project offers a scalable and customizable solution for businesses and individuals managing large email volumes. It aligns with contemporary research and technological advancements, highlighting RPA's transformative potential in improving productivity and ensuring streamlined communication workflows.

CHAPTER 3

SYSTEM DESIGN

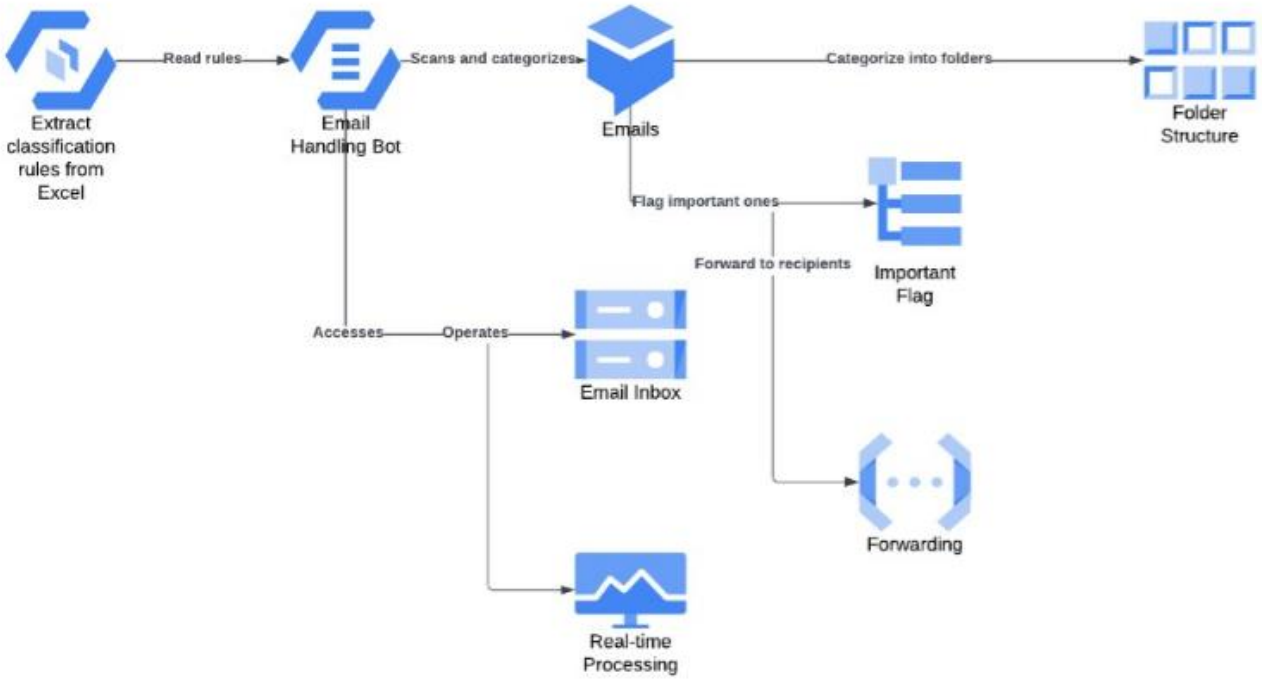
3.1 SYSTEM FLOW DIAGRAM

A flowchart is a type of diagram that represents an algorithm, workflow or process. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem.



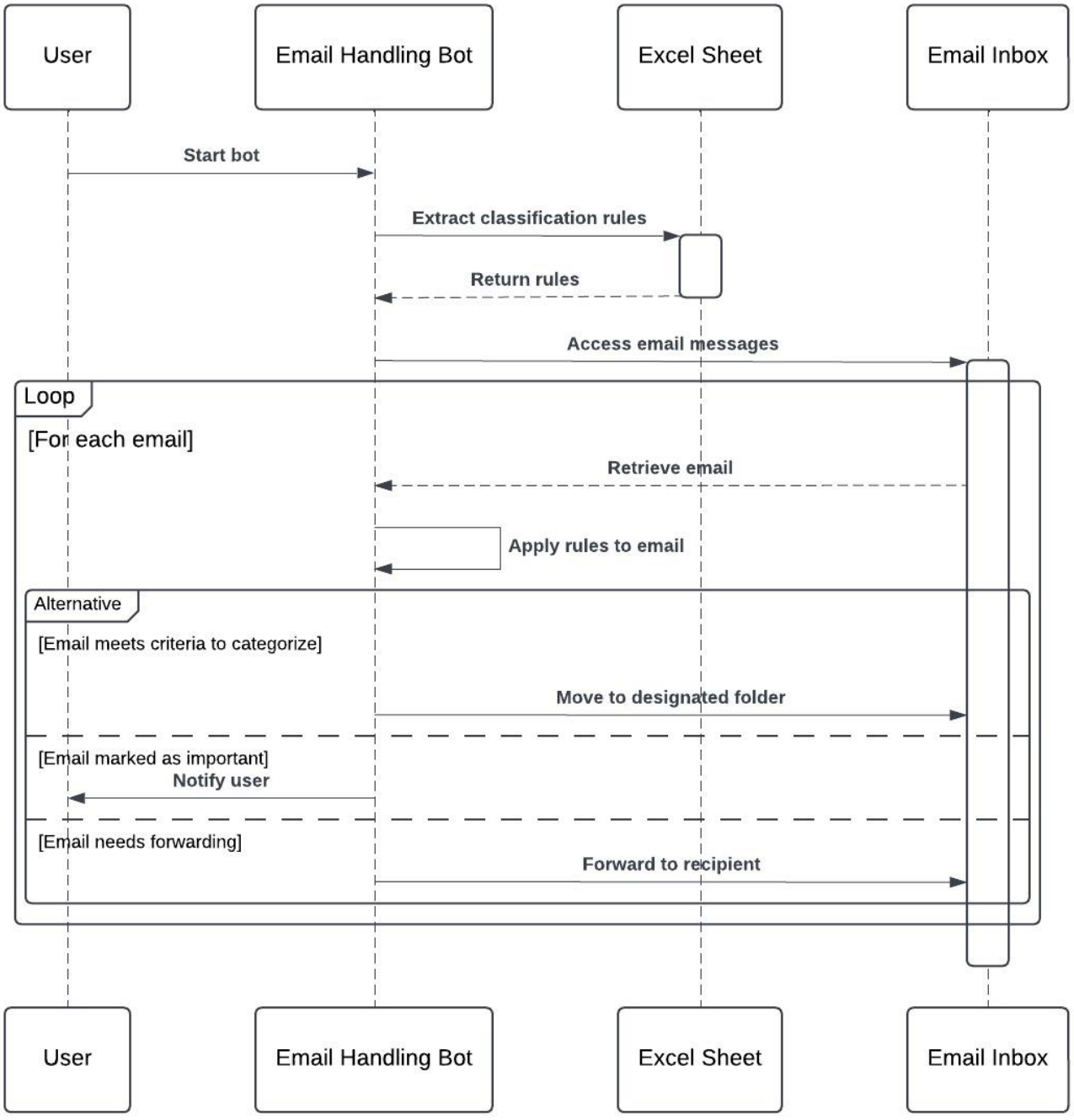
3.2 ARCHITECTURE DIAGRAM

An architecture diagram is a graphical representation of a set of concepts, that are part of an architecture, including their principles, elements and components.



3.3 SEQUENCE DIAGRAM

A sequence diagram is a type of interaction diagram because it describe and how in what order a group of objects works together.



CHAPTER 4

PROJECT DESCRIPTION

Methodology

The methodology outlines the systematic approach adopted to develop the Email Handling Bot Using UiPath. The project is divided into distinct phases to ensure efficiency, reliability, and scalability in automating email management. Below are the key stages of development:

1. **Requirement Analysis & Setup**
   Identify Functional Needs: Define functionalities like reading email handling rules from an Excel sheet, accessing the inbox, segregating emails based on rules, and performing actions like flagging, forwarding, or moving emails to specific folders.
   Setup UiPath Studio: Install UiPath Studio and ensure the necessary packages, such as UiPath.Mail.Activities and UiPath.System.Activities, are included.
   Email Server Configuration: Configure the bot to access the email account (e.g., IMAP, POP3, or Exchange protocols).

2. **Data Source Setup**
   Prepare Input File: Create a structured Excel sheet or CSV file containing email segregation rules (e.g., keywords, sender, or subject line-based criteria).
   Validation Rules: Validate the input file to ensure it adheres to the required format and contains complete, error-free data.

3. **Email Processing Automation**
   Inbox Access: Use Get IMAP Mail Messages or Get Outlook Mail Messages activity to retrieve emails.
   Rule Application: Implement decision logic (e.g., If conditions or Switch activity) to classify emails based on rules defined in the Excel sheet.
   Perform Actions:Move emails to folders using the Move Mail Message activity.
   Forward emails to specified recipients using the Send SMTP Mail Message or Outlook activity.
   Flag important emails for follow-up.

4. **Error Handling**
   Invalid Rules or Missing Data: Use Try-Catch blocks to handle incomplete or invalid input files and log the errors.
   Email Processing Errors: Implement error handling for connection failures, missing emails, or unsupported formats.
   Retry Mechanism: Use retry logic to reattempt failed operations caused by temporary issues.

5. **Logging and Reporting**
   Log Activities: Maintain logs for each email processed, including successful actions and failures, using Log Message or Write Line activities.
   Generate Reports: Create a summary report (Excel or text) detailing the status of each email, actions performed, and errors encountered.

6. **Testing & Optimization**
   Test Workflows: Use sample emails to test the bot for edge cases, such as emails with missing subjects, unsupported attachments, or malformed rules.
   Optimize Performance: Minimize unnecessary delays, use efficient selectors, and streamline workflows for faster processing.

7. **Scheduling & Deployment**
   Schedule Execution: Use UiPath Orchestrator to schedule the bot for periodic execution or trigger-based email handling.
   Monitor and Manage: Monitor the bot's performance using Orchestrator logs, handle any runtime errors, and update rules as needed.

4.1.1 Modules

1. Rule Extraction

Objective: Extract email handling rules from the input Excel or CSV file.

Activities: Use Excel Application Scope and Read Range activities to fetch rules into a DataTable.

Validate data for completeness and consistency using If conditions or custom validation logic.

2. Inbox Access

Objective: Retrieve emails from the configured email account.

Activities: Use Get IMAP Mail Messages or Get Outlook Mail Messages to access the inbox.

Filter emails based on date, sender, or other criteria using email activity properties.

3. Email Segregation & Action Execution

Objective: Classify emails and execute actions based on predefined rules.

Activities: Apply rule-based logic to segregate emails into folders, flag them, or forward them.

Use Move Mail Message, Send Mail, or similar activities to perform required actions.

4. Error Handling & Logging

Objective: Ensure robustness in handling errors and maintain activity logs.

Activities: Use Try-Catch blocks to manage errors like invalid rules or email server downtime.

Log each email's status (success or failure) using Write Line or Append Range.

5. Reporting

Objective: Provide a detailed report of email processing.

Activities: Use Write Range to generate an Excel report summarizing actions taken on each email.

Include additional details, such as reasons for skipped or failed actions.

6. Scheduling & Deployment

Objective: Automate periodic execution and ensure scalability.

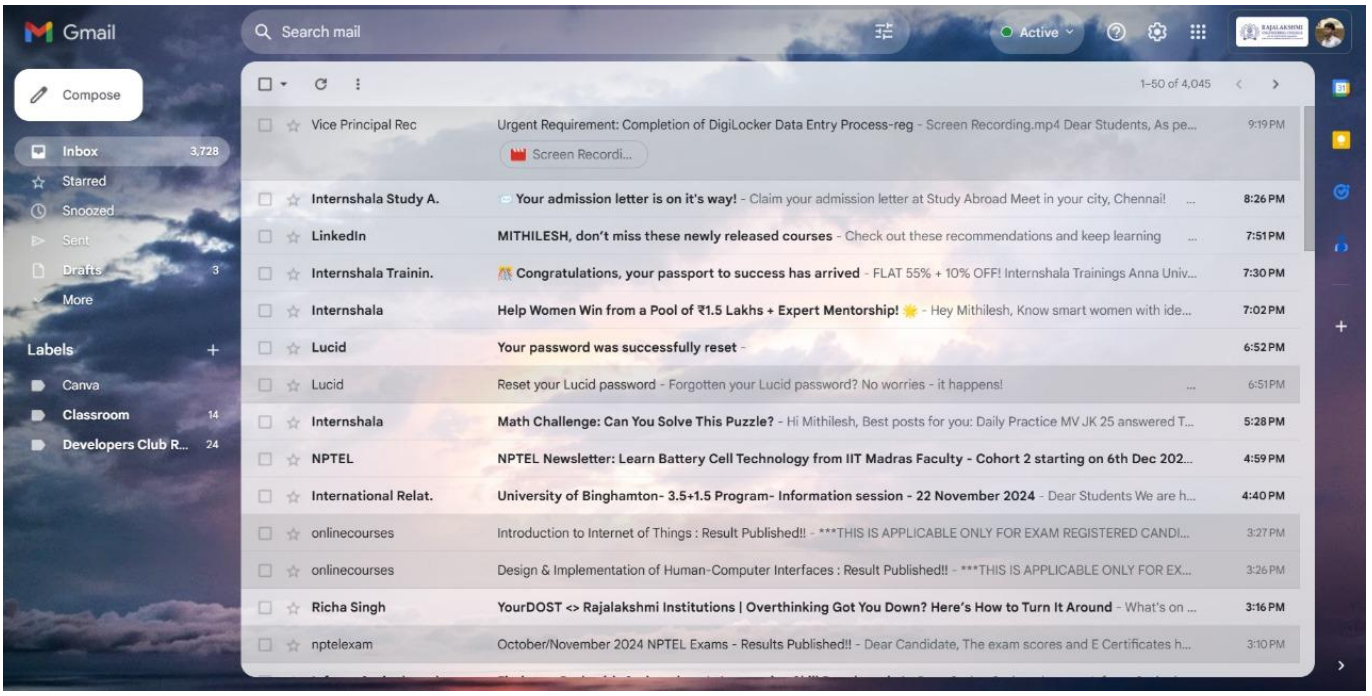
Activities: Use Orchestrator to schedule the bot for regular execution.

Monitor logs and manage any runtime issues through Orchestrator's dashboard.

This methodology ensures a robust and scalable solution for automating email management using UiPath.

CHAPTER 5

OUTPUT SCREENSHOT



The Data is fetched from the excel file and processed.

sender	folder
engage.ca	Canva
rajalakshm	Developers Club REC
classroom	Classroom

CHAPTER 6

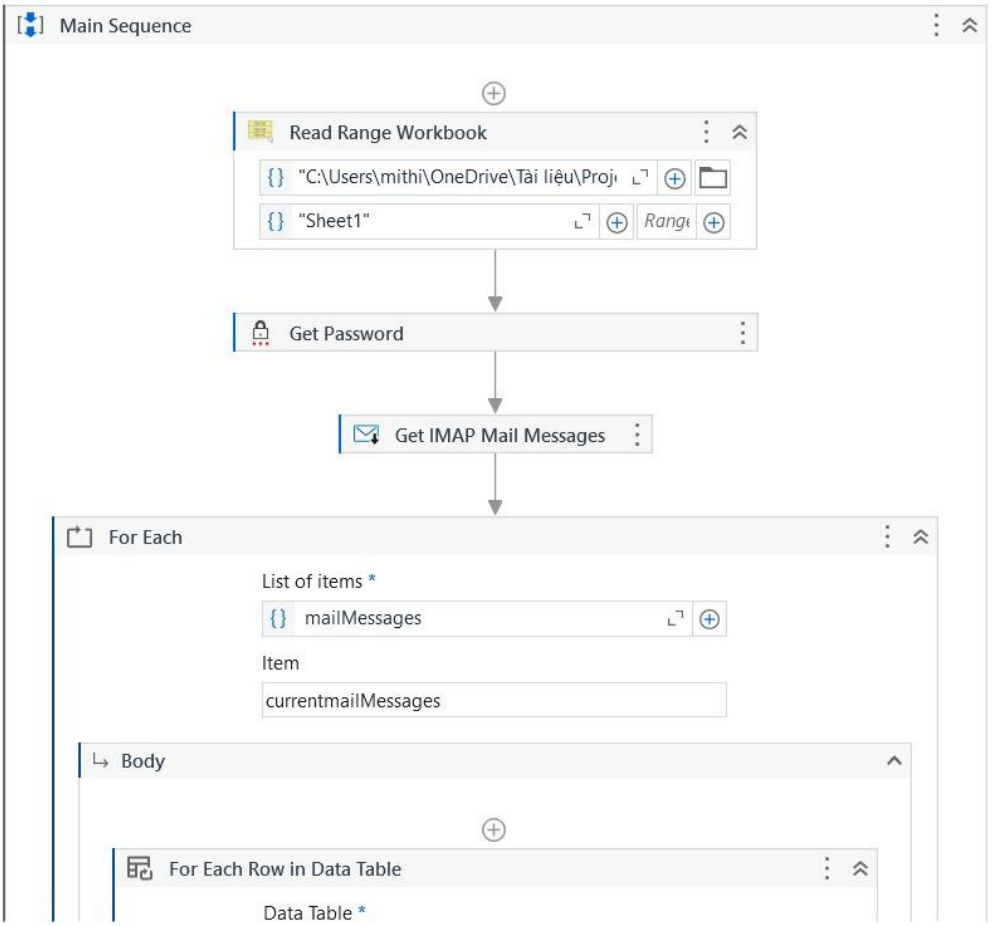
CONCLUSION

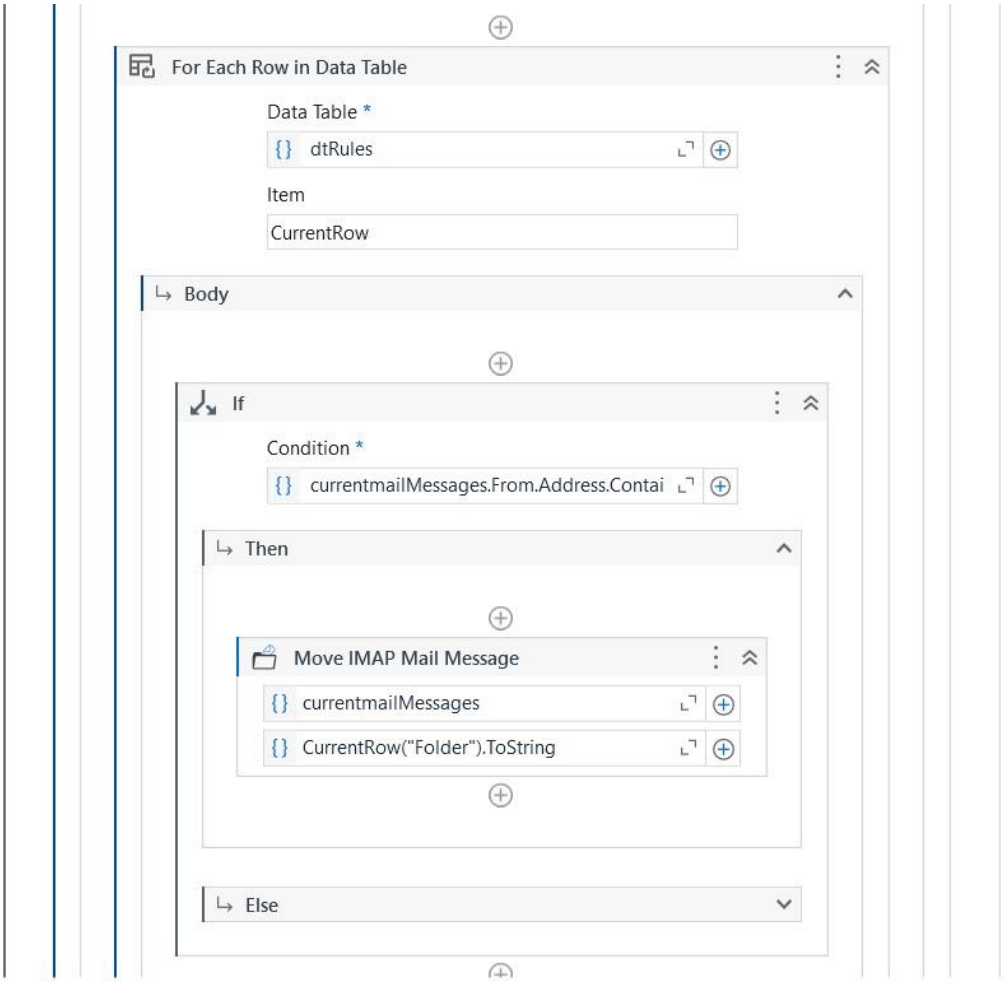
The Email Handling Bot demonstrates the transformative potential of Robotic Process Automation (RPA) in simplifying and enhancing email management processes. By automating repetitive tasks such as email categorization, flagging, and forwarding based on user-defined rules, the bot reduces manual effort, minimizes errors, and ensures timely processing of emails. This project highlights the scalability, adaptability, and efficiency of RPA in addressing the challenges associated with high-volume email handling.

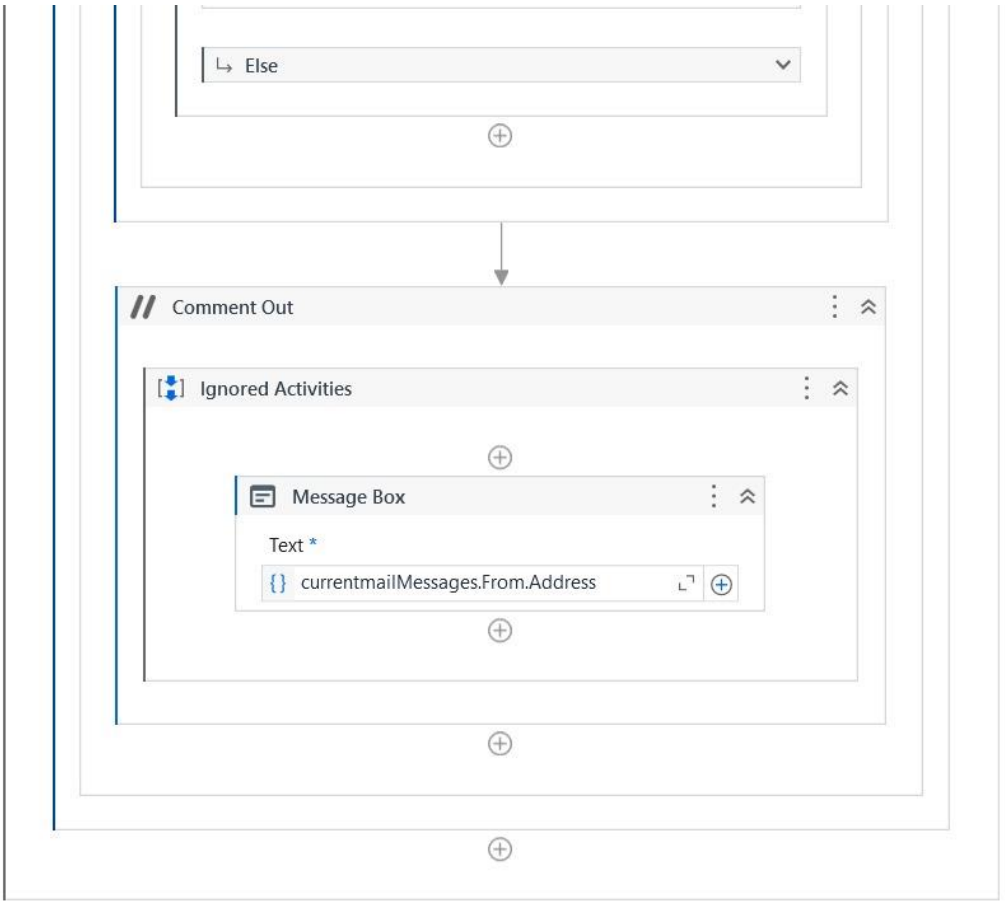
By integrating RPA with existing tools like Excel and email platforms, the solution bridges the gap between traditional methods and modern automation technologies. The **Email Handling Bot** not only improves productivity but also empowers users to focus on more strategic tasks, making it a valuable tool for both personal and professional use cases. This project paves the way for further exploration of RPA's applications in communication management, reinforcing its role in enhancing operational efficiency and reliability.

APPENDIX

PROCESS WORK FLOW




For Each Row in Data Table
Data Table *
dtRules
Item
CurrentRow
Body
If
Condition *
currentmailMessages.From.Address.Contai
Then
Move IMAP Mail Message
currentmailMessages
CurrentRow("Folder").ToString
Else


Else
Comment Out
Ignored Activities
Message Box
Text *
currentmailMessages.From.Address

REFERENCES

[1] Sharma, P., & Mehta, R. (2021). Optimizing Email Management with Robotic Process Automation (RPA). *International Journal of Computer Science and Applications*, 18(3), 45–52.

[2] Verma, A., & Singh, K. (2020). Enhancing Communication Workflows Using RPA: A Case Study on Email Handling Automation. *Journal of Automation and Business Processes*, 12(5), 102–110.

[3] Gupta, N., & Roy, S. (2022). Intelligent Email Segregation with Robotic Process Automation. *International Journal of Emerging Trends in Technology and Management*, 9(2), 65–72.

[4] Patel, H., & Desai, P. (2019). Leveraging RPA for Efficient Email Processing in Enterprises. *International Journal of Advanced Research in Technology and Management Sciences*, 7(4), 23–30.

[5] Brown, M., & Wilson, J. (2020). Automation in Email Management: Improving Productivity Through RPA. *Proceedings of the International Conference on Automation and Technology Advancements*, 45–50.

[6] Yadav, R., & Kumar, P. (2021). RPA-Driven Solutions for Email Organization: A Framework for Scalable Automation. *Journal of Information Systems and Automation Research*, 10(3), 56–63.