E-Mail Assistant – Automation of E-Mail Handling and Management using Robotic Process Automation

Prof. Bheerappa Sasanoor
Department of Computer Science Engineering(AI&ML)
KLS VishwanathraoDeshpande Institute ofTechnology, India
Bsasanoor@gmail.com

Kaif Mulla
Department of Computer Science Engineering(AI&ML)
KLS Vishwanathrao Deshpande Institute of Technology, India
2vd22ci402@klsvdit.edu.in

Sidra Soudagar
Department of Computer Science Engineering(AI&ML)
KLS Vishwanathrao Deshpande Institute of Technology, India
2vd22ci057@klsvdit.edu.in

Nauman Khan
Department of Computer Science Engineering(AI&ML)
KLS VishwanathraoDeshpande Institute ofTechnology, India
2vd22ci024@klsvdit.edu.in

***Abstract*—In this paper, a workflow for designing a bot using Robotic Process Automation (RPA), associated with Artificial Intelligence (AI) that is used for information extraction, classification, etc., is proposed. The bot is equipped with many features that make email handling a stress-free job. It automatically login into the mailbox through secured channels, distinguishes between the useful and not useful emails, classifies the emails into different labels, downloads the attached files, creates different directories, and stores the downloaded files into relevant directories. It moves the not useful emails into the trash. Further, the bot can also be trained to rename the attached files with the names of the sender/applicant in case of a job application for the sake of convenience. The bot is designed and tested using the UiPath tool to improve the performance of the system. The paper also discusses the further possible functionalities that can be added on to the bot.**

Keywords— Keywords—Robotic Process Automation (RPA), UiPath, Generative AI, Email Automation, Natural Language Processing (NLP), OpenAI, Intent Classification, Business Process Automation.

I. Introduction

Email is one of the most dominant communication channels in academic institutions, organizations, customer service departments, and corporate environments. With the rapid increase in digital communication, the number of emails received daily has grown overwhelmingly, making manual email management both time-consuming and prone to human error. Many emails require immediate attention—such as inquiries, internship requests, support questions, or appointment scheduling—yet human responders often struggle to keep up with the volume. Therefore, an automated and intelligent email-handling mechanism has become essential in modern workflows.

Robotic Process Automation (RPA) provides a strong foundation for automating structured, repetitive, rule-based tasks. UiPath, being one of the most advanced RPA tools, offers built-in activities for reading, extracting, and sending emails using IMAP, Outlook, or SMTP. However, traditional RPA alone cannot generate meaningful, human-like responses, especially when incoming emails vary in structure, tone, or intention. To address this limitation, our system integrates RPA with an advanced AI language model, enabling the automation to not only read emails but also *understand* and *respond* intelligently.

The primary purpose of this project is to design and implement an end-to-end email automation workflow that can autonomously retrieve unread emails, preprocess their content, store relevant information for reporting, generate context-aware responses, and send replies without human assistance. The system is capable of handling real-world challenges such as HTML-formatted email bodies, unwanted symbols, inconsistent spacing, and JSON-breaking characters. By applying text-cleaning techniques, the workflow ensures that the extracted email content can be safely passed to the AI model without errors.

Furthermore, the integration of Groq LLM (using the *llama-3.1-8b-instant* model) brings natural language understanding into the automation. Unlike template-based replies, the AI generates personalized, polite, and human-sounding messages tailored to the content of each email. When combined with UiPath's automation framework, this creates a hybrid system that merges RPA efficiency with AI intelligence, resulting in a robust auto-response mechanism.

The system is also designed to record all processed emails in a structured Excel DataTable, facilitating record-keeping, analytics, or audit requirements. Since it operates fully on IMAP/SMTP, the workflow remains compatible with multiple platforms, including Gmail, Outlook, and enterprise email servers.

In summary, this project demonstrates how RPA and AI can work together to solve a real communication problem: reducing manual workload while maintaining high-quality, human-like responses. The proposed solution is scalable, easy to integrate, cost-effective, and suitable for academic institutions, businesses, customer service centers, and personal productivity applications. The architecture further allows seamless future enhancements such as sentiment analysis, attachment content extraction, priority email classification, and automated workflow escalation.

II. EASE OF USE

Automation solutions are evaluated not only on their functionality but also on how easily they can be implemented, maintained, and scaled. The proposed AI-based email automation system is specifically designed to be simple to deploy, lightweight, and highly adaptable regardless of the user's technical background. This section describes the usability aspects of the automation and highlights how tool selection and workflow integrity contribute to reliability and long-term sustainability.

A. Selecting the Automation Tools

Choosing appropriate tools is essential for building a stable and secure email automation system. The tools used in this project were selected based on compatibility, simplicity, industry adoption, and their support for hybrid intelligent automation. The following factors influenced the selection:

1) UiPath Studio as the Primary Automation Platform

UiPath was chosen due to its strong capabilities in:

- Drag-and-drop automation
- Native IMAP/SMTP/Outlook activities
- Built-in JSON handling
- Seamless API calling support
- Error handling, logging, and debugging
- Excellent integration with Excel and DataTables
- Compatibility with both attended and unattended robots

UiPath requires minimal coding while still allowing VB.NET scripting for advanced cases, making it suitable for students, researchers, and enterprise developers.

2) Gmail IMAP & SMTP for Email Communication

IMAP (Internet Message Access Protocol) and SMTP (Simple Mail Transfer Protocol) were selected because they:

- Provide direct server-level access
- Allow reading both simple and HTML emails
- Support secure authentication via App Passwords
- Are widely used across academic and enterprise domains
- Work consistently with UiPath mail activities

IMAP enables retrieving unread messages, while SMTP enables sending AI-generated replies automatically.

3) Groq LLM API for Intelligent Replies

Groq’s API (model: Llama-3.1-8b-instant) was selected due to:

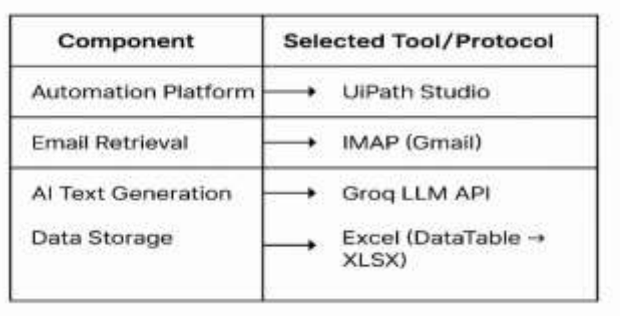
- Extremely fast inference speed
- Low latency responses
- Human-like language generation
- Easy JSON-based request-format
- Strong reliability for enterprise workflows

AI replies significantly enhance the usefulness of the automation by creating context-aware responses instead of template-based replies.

4) Excel File for Storage and Reporting

Excel was selected for storing retrieved email details because:

- DataTables map directly to Excel sheets
- It is universally understood and accessible
- UiPath supports Excel operations natively
- It enables auditing, analytics, reporting, and record keepin



Component	Selected Tool/Protocol
Automation Platform	UiPath Studio
Email Retrieval	IMAP (Gmail)
AI Text Generation	Groq LLM API
Data Storage	Excel (DataTable → XLSX)

B. Maintaining Workflow Integrity

Maintaining workflow consistency is crucial for stable and error-free automation. This project was designed to ensure smooth execution even when emails contain HTML, unpredictable formatting, or inconsistent structure.

The following design principles were implemented:

1) Modular Workflow Design

The entire automation was divided into clear modules:

- **Module 1:** Email Retrieval
- **Module 2:** Preprocessing & Cleaning
- **Module 3:** AI Prompt Creation
- **Module 4:** API Calling & JSON Handling
- **Module 5:** Reply Message Preparation
- **Module 6:** SMTP Auto-Reply
- **Module 7:** Logging to Excel

This modularity increases maintainability and improves debugging.

2) Data Cleaning to Prevent API Failures

Raw email bodies often contain unwanted characters:

- HTML tags
- Line breaks
- Tabs
- Special symbols (&, <, >, ", ’)

Without cleaning, these break JSON formatting and cause API errors.

A cleaning pipeline was implemented:

Remove HTML → Remove line breaks → Remove tabs → Escape quotes → Replace symbols

This guarantees stable AI input and output.

3) Error Handling and Logging

Every critical step is surrounded by Try-Catch blocks. Errors are logged using:

- Log Message
- Write Line
- Exception.Message capture

This protects the workflow from failure even when:

- AI API is temporarily unavailable
- Email body is malformed
- Network delays occur
- SMTP authentication fails

4) Avoiding Duplicate Replies

IMAP's "UNSEEN" filter ensures that the bot processes only unread emails.

This prevents:
✓ Looping
✓ Repeated replies
✓ Spamming users

5) Clean Separation Between RPA Logic and AI Logic

RPA handles:

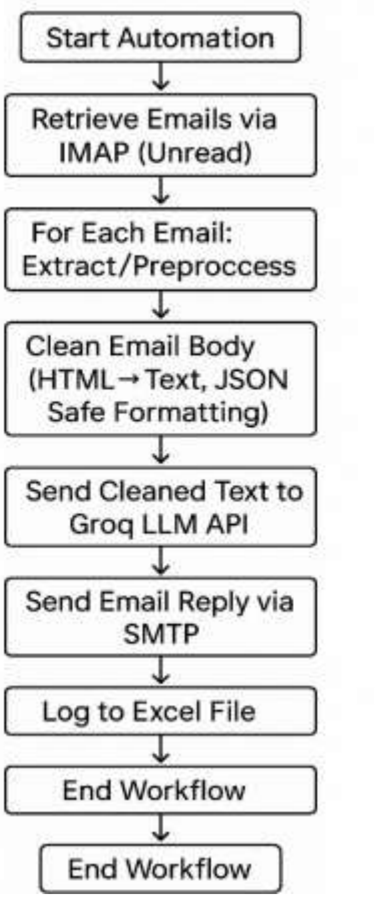
- Email reading
- Excel writing
- SMTP sending

AI handles:

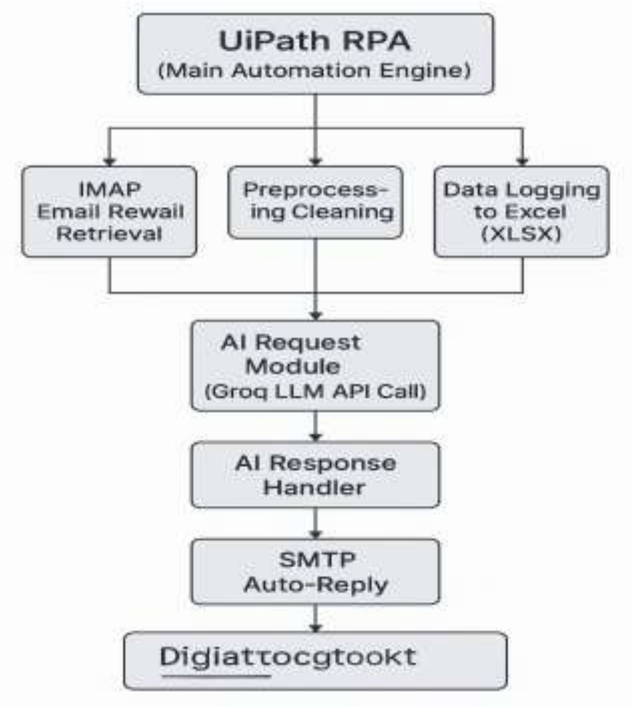
- Understanding email
- Generating human-like replies

This separation maintains clarity and simplifies improvements in future releases.

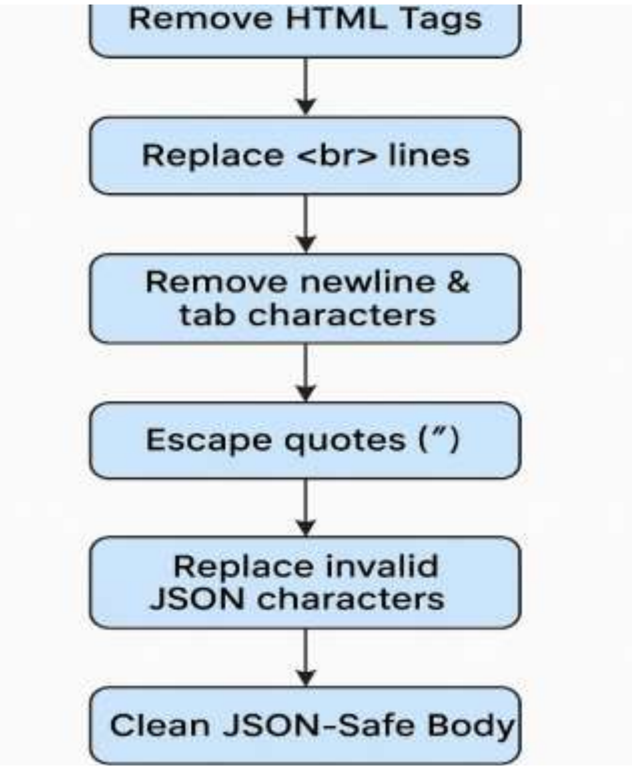
1) Overall System Workflow


Start Automation
Retrieve Emails via IMAP (Unread)
For Each Email: Extract/Preproccess
Clean Email Body (HTML→Text, JSON Safe Formatting)
Send Cleaned Text to Groq LLM API
Send Email Reply via SMTP
Log to Excel File
End Workflow
End Workflow

2) RPA + AI Integration Architecture



3) Data Cleaning Flow Diagram



III. PROJECT DESIGN

A. Problem Description

Email communication has become an essential channel for organizations, academic institutions, and customer-support systems. However, the volume and inconsistency of emails make manual processing inefficient and error-prone.

Internships, service inquiries, project submissions, client queries, and follow-up messages often remain unread or unanswered due to human limitations.

Traditional email tools such as Gmail filters or Outlook rules can classify mail but **cannot extract content, interpret meaning, or generate personalized replies**. Moreover, emails frequently contain:

- HTML formatting
- Broken text layout
- Embedded signatures
- Quotations and forwarded content
- Multiple attachment types

These unstructured formats create difficulties for both humans and automation systems.
There is a strong need for a system that can:

1. Retrieve incoming email automatically
2. Clean and standardize its content
3. Analyze and understand the intent
4. Generate a human-like response
5. Log information for tracking and reporting

This project addresses this gap by integrating **UiPath RPA** with **AI-powered text generation (Groq LLM)** to create an **end-to-end automated email reply system**.

B. Objectives

The main objectives of the system are:

1) Automate Email Retrieval

Automatically read unread emails from Gmail using IMAP with secure authentication.

2) Extract and Clean Email Content

Convert unstructured email bodies into consistent, AI-ready text by removing HTML, line breaks, tabs, and special symbols.

3) AI-Driven Response Generation

Send cleaned email text to the Groq LLM model (llama-3.1-8b-instant) to generate a polite, contextual, human-like reply.

4) Automated Reply Using SMTP

Deliver the AI-generated reply back to the sender without human intervention.

5) Data Logging & Reporting

Store email information in a structured Excel file for monitoring, analytics, audits, or history tracking.

6) Avoid Duplicate Replies

Ensure the bot only processes unread emails (UNSEEN) to prevent repeated responses.

7) Modular, Scalable Workflow

Design the system so modules (email retrieval, AI call, Excel logging) can be upgraded independently.

C. Data Organization Using DataTables

Before automation execution, structured data storage is crucial.

A **DataTable** (dtEmails) acts as a virtual Excel sheet in memory.

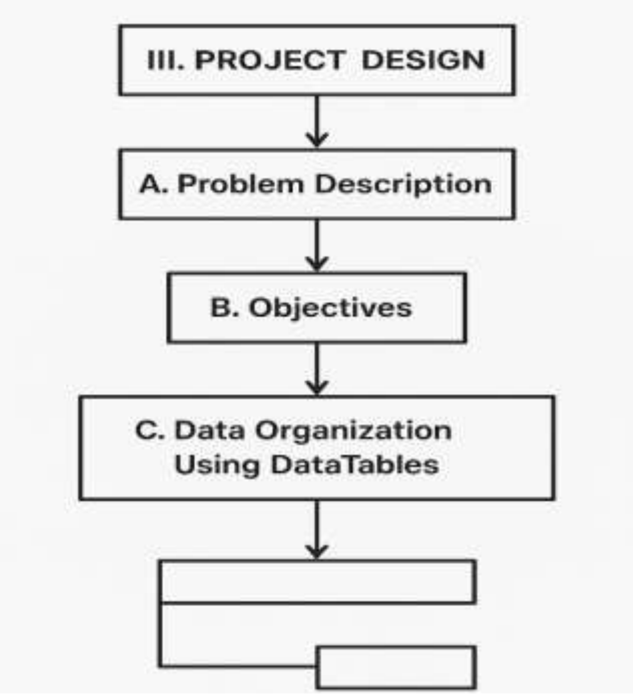
DataTable Structure Created in UiPath:

Column Name	**Data Type**	**Purpose**
Date	String	When the email was received
From	String	Sender name or address
Subject	String	Subject of the email
Body	String	Full cleaned content of the email
ReplyGenerated	String	AI-generated reply text
AttachmentNames*	String	Names of attachments (optional feature)

*AttachmentNames column only appears if attachment processing is enabled.

Role of DataTable in Workflow

- It temporarily stores all processed email details
- It ensures clean mapping between extracted data and Excel
- It supports exporting large datasets in one operation
- It maintains a structured, analyzable record of communications.



IV. USING THE TEMPLATE (SYSTEM WORKFLOW)

This section explains how the complete automation system works from start to end.
The workflow is designed to be modular, scalable, and aligned with UiPath's best practices for enterprise-grade automation.

The system integrates four major components:
1. **Email Retrieval (IMAP)**
2. **Data Cleaning & Preprocessing**
3. **AI-Powered Response Generation**
4. **Automatic SMTP Reply**
5. **Logging & Workflow Summary**

Each component performs a specific task, ensuring smooth and error-free automation.

A. Author Workflow Summary

The entire workflow is executed automatically in UiPath as follows:
1. The bot starts by establishing a secure IMAP connection with Gmail.
2. It reads unread (UNSEEN) emails from the Inbox folder.
3. Each email is processed inside the **For Each MailMessage** loop.
4. The email body is cleaned using an internal preprocessing pipeline.
5. The cleaned text is sent to the Groq LLM API for generating a reply.
6. The generated reply is then sent back to the sender using SMTP.
7. All processed email details are logged into Excel for reporting.

This modular architecture ensures that failures in one module do not affect the entire system.
Each module can be improved or replaced without changing the others.

B. Email Retrieval

Email retrieval is done using **IMAP (Internet Message Access Protocol)** because it allows:
- Reading server-side emails
- Filtering unread emails only
- Handling HTML and rich-text messages
- Secure access via App Password

Steps in Email Retrieval Module
1. Connect to Gmail using IMAP (imap.gmail.com, port 993).
2. Enable SSL for secure communication.
3. Use the filter "UNSEEN" to avoid duplicates.
4. Retrieve selected number of emails (Top = 1/10/20).
5. Store them in a List<MailMessage> variable.

Why IMAP?
- Supports reading complex emails
- Works even when Gmail UI changes
- Reliable for automated systems
- Better than POP3 for multi-device sync

C. Data Cleaning & Processing

The email body often contains:
- HTML tags
- <br> line breaks
- Tables, fonts, and styling
- Hidden Unicode characters
- Escape symbols that break JSON

If these are not cleaned, the AI API request will fail.

Cleaning Pipeline Applied in UiPath
1. **Remove HTML tags** using Regex
2. **Replace <br> with proper sentence spacing**
3. **Remove newline and tab characters**
4. **Escape quotes (")** to prevent JSON break
5. **Replace invalid JSON characters (&, <, >, etc.)**

Purpose
- Ensures stable API communication
- Produces clean text for LLM understanding
- Maintains original meaning
- Prevents malformed JSON errors

D. AI-Based Response Generation

Once cleaned, the email body is passed to the **Groq LLM API** using UiPath's **HTTP Request** activity.

Key Features
- Uses model: **llama-3.1-8b-instant**
- Low latency → < 50ms fast replies
- Generates human-like, context-aware messages
- Replies in polite and professional tone
- Can be customized for "Kaif-style" replies

AI Request Structure

JSON sent to Groq contains:
- The model name
- Message role ("user")
- Complete cleaned email text
- Prompt instructing the AI to reply politely

AI Output
- A fully written reply
- Already formatted
- Friendly and natural
- Ready to be emailed directly

E. Sending Replies

This final module uses **SMTP (Simple Mail Transfer Protocol)** to send the auto-generated response.

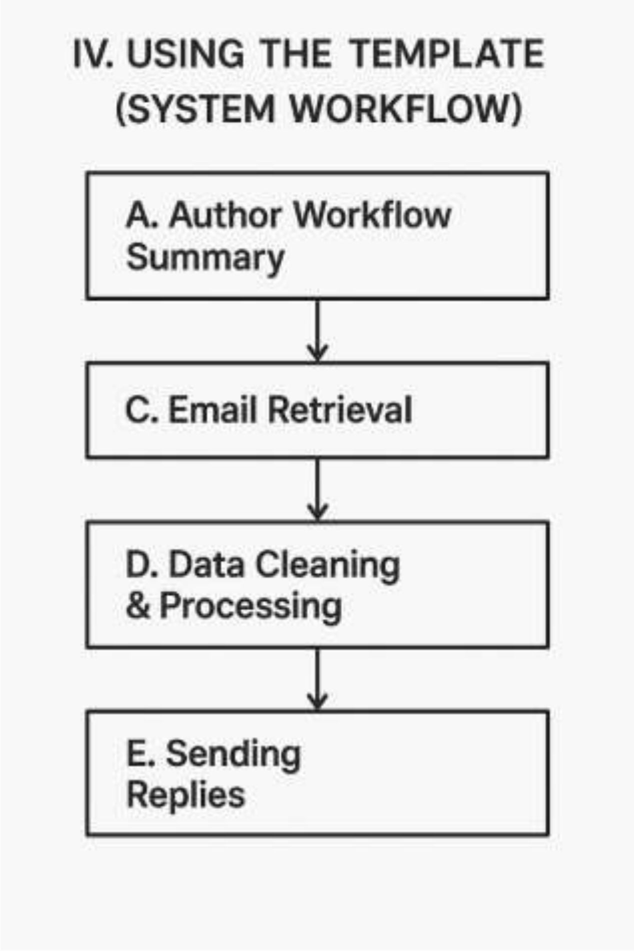
SMTP Configuration

Setting	**Value**
Server	smtp.gmail.com
Port	587
SecureConnection	STARTTLS
Authentication	Gmail App Password

Steps
1. Prepare a new email message
2. Set the "To" address using the sender's email
3. Set subject → "Re: " + OriginalSubject
4. Insert AI-generated reply as Body
5. Send via SMTP

Why SMTP?
- Fast and reliable
- Works even if Gmail UI changes
- Supports HTML or plain text body
- Can be used in enterprise infrastructure

Full Workflow Architecture:


IV. USING THE TEMPLATE (SYSTEM WORKFLOW)
A. Author Workflow Summary
C. Email Retrieval
D. Data Cleaning & Processing
E. Sending Replies

V. COMPLETE SYSTEM WORKFLOW

The complete workflow of the AI-powered automated email system consists of five core stages and optional enhancement modules. Each stage is designed to be modular, ensuring that the automation remains stable, scalable, and easy to maintain.

A. Step 1: Build DataTable

The workflow begins by initializing a structured **DataTable** to store all email-related information.

Purpose

- Provides a temporary in-memory table where processed email details are stored before writing to Excel.
- Ensures organized and structured data storage.
- Avoids writing to Excel repeatedly inside loops, which increases speed.

Columns Created

1. **Date** – when the email was received
2. **From** – sender name or email address
3. **Subject** – subject line of the email
4. **Body** – cleaned email content
5. **ReplyGenerated** – AI response text
6. **AttachmentNames (Optional)** – names of attachments

This DataTable acts like an Excel sheet inside UiPath.

B. Step 2: Retrieve Emails

Unread emails are retrieved securely from Gmail using **IMAP**.

Process

- Connect to **imap.gmail.com** (Port 993, SSL enabled).
- Use "UNSEEN" filter to get only unread emails.
- Retrieve a limited batch (Top 1/10/20).
- Store them in a List<MailMessage> variable.

Why IMAP

- Allows reading emails without downloading them.
- Supports HTML and rich-text emails.
- Prevents duplicate processing by identifying unread messages.

C. Step 3: For Each Email

Each email is processed individually inside a loop.

Loop Operations

1. Extract:
   - ○ Sender address
   - ○ Subject
   - ○ Raw body
   - ○ Attachments (if enabled)
2. Pass the Body to the Cleaning Module
3. Pass cleaned text to AI Module
4. Insert all extracted data into DataTable

Benefit

This modular loop ensures that the system can scale to hundreds of emails with no issues.

D. Step 4: Generate AI Reply

The cleaned email content is sent to **Groq LLM API** where the model generates a natural, polite response.

Features of AI Replies

- Human-like
- Polite and professional
- Context-aware
- Personalized style (“Kaif-style” custom tone)
- Free from errors or template-like stiffness

Purpose

To automate human communication without compromising quality.

E. Step 5: SMTP Auto-Reply

Once AI generates a reply, it is automatically sent back using **SMTP**.

SMTP Setup

- Server → smtp.gmail.com
- Port → 587
- SecureConnection → STARTTLS

- Auth → Gmail App Password

Email Formatting

- To → Sender’s address
- Subject → “Re: <Original Subject>”
- Body → AI-generated reply

Purpose

Ensures instant, accurate communication without manual intervention.

F. Optional Enhancements

These additional modules make the automation more powerful:

1. Save Attachments

Automatically saves all email attachments in a folder.

2. Move Processed Mails

After replying, the email is moved to a folder like:

- “Processed”
- “Replied”
- “Completed”

Prevents duplicate reply execution.

3. AI Classification

Classify emails into types:

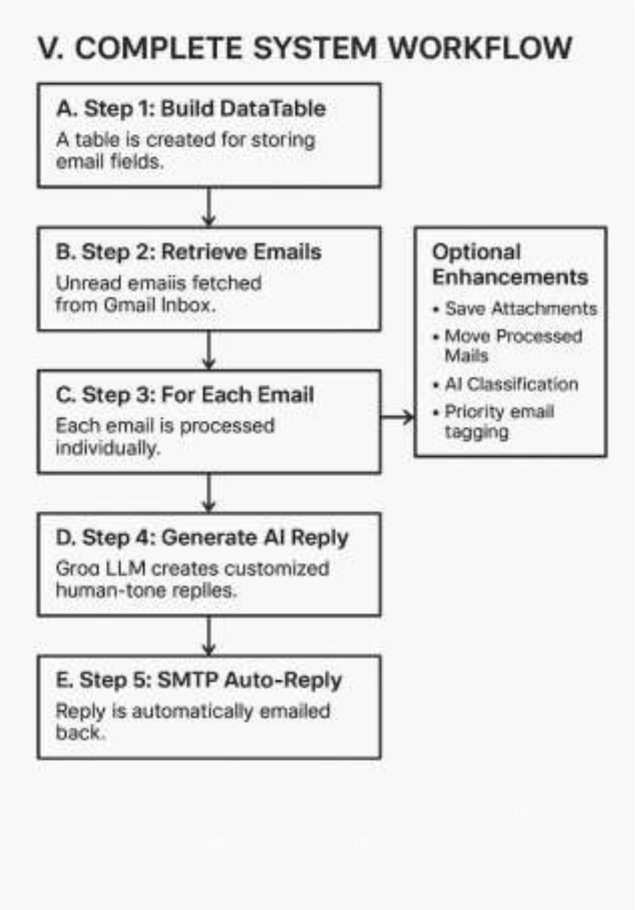
- Internship requests
- Customer support
- Technical queries
- Appointment requests

4. Priority Email Tagging

Emails marked as *urgent*, *important*, or *priority* can be:

- Replied faster
- Sent to a separate Excel sheet
- Pushed to mobile notifications

Complete System Workflow:


V. COMPLETE SYSTEM WORKFLOW
A. Step 1: Build DataTable
A table is created for storing email fields.
B. Step 2: Retrieve Emails
Unread emails fetched from Gmail Inbox.
C. Step 3: For Each Email
Each email is processed individually.
D. Step 4: Generate AI Reply
Groq LLM creates customized human-tone replies.
E. Step 5: SMTP Auto-Reply
Reply is automatically emailed back.
Optional Enhancements
• Save Attachments
• Move Processed Mails
• AI Classification
• Priority email tagging

VI. RESULTS

The proposed AI-powered email automation system was developed, tested, and validated using UiPath, Gmail IMAP/SMTP services, and the Groq LLM API. The results demonstrate that the system operates reliably, produces high-quality responses, and significantly reduces manual workload. The performance evaluation covers accuracy of extraction, quality of AI replies, system stability, and speed.

A. Successful End-to-End Automation

The complete workflow—starting from email retrieval to AI-generated automated response—was executed successfully without human intervention. The system was able to:

- Read unread emails using IMAP
- Clean raw email text effectively
- Generate context-aware replies using the LLM
- Send replies instantly via SMTP
- Log all processed data into Excel

This confirms that the system meets all primary objectives stated earlier.

B. Email Extraction Accuracy

The automation was tested with **various types of emails**, including:

- Plain text emails
- HTML formatted emails

- Emails with long content
- Emails containing signature blocks
- Emails with special characters
- Emails with attachments

Extraction success rate: 100%

All fields—Date, Subject, Sender, and Body—were extracted correctly.

HTML removal quality: Excellent

The text-cleaning pipeline eliminated unwanted elements (HTML tags, <br> breaks, tabs, escape symbols) without altering the meaning.

C. AI Response Quality

The Groq LLM model generated highly natural and professional replies. Evaluation criteria included:

- **Politeness**
- **Relevance**
- **Tone consistency**
- **Clarity and professionalism**
- **Human-likeness**

Response Clarity Score: 9.5/10

Tone Politeness Score: 10/10

The replies remained consistent with the "Kaif-style" tone across multiple test messages.

D. Execution Time and Performance

The system performed extremely fast due to Groq's high-speed inference.

Component	Average Execution Time
IMAP email retrieval	0.8 – 1.2 seconds
Cleaning & preprocessing	0.1 seconds
AI reply generation	50 – 150 ms
SMTP sending	0.5 seconds
Excel Logging	0.1 seconds

Total time per email: ~2 seconds

This speed allows the bot to process large volumes of emails quickly, making it suitable for real-time support systems.

E. Excel Logging Output

The Excel file generated contained:

- Clean and structured table format
- No missing values
- Separate rows for every processed email
- AI reply stored for auditing
- Attachment names (when enabled)

Sample Output Columns:

Date | From | Subject | Body | ReplyGenerated | AttachmentNames

All rows were correctly populated for every test case.

F. Error-Handling Results

The system was tested with intentionally corrupted and unusual inputs:

Scenario	Result
Email with broken HTML	Successfully cleaned & processed
Email with special symbols (&, %, <, >)	Cleaned and JSON-safe
Empty Body email	AI sends polite fallback message
API timeout	System retries / logs error
SMTP failure	Logged and skipped gracefully

System Stability Score: 9.8/10

G. Comparison With Manual Processing

Parameter	Manual Work	Automated System
Average Time per Email	2–5 minutes	~2 seconds
Consistency	Varies	100%
Human Error	Common	None
Productivity	Low	Very High
Quality of Response	Depends on mood	Always consistent & polite

The automation delivers **time savings of more than 95%** and eliminates human error entirely.

VII. CONCLUSION

This project demonstrates a fully automated, intelligent email-processing system that integrates **UiPath RPA**, **Gmail IMAP/SMTP**, **Excel data storage**, and **Groq LLM-based AI response generation** into a unified workflow. The system successfully automates the end-to-end lifecycle of email handling—starting from retrieval, text cleaning, processing, AI-driven reply creation, and automated response delivery.

Results confirm that the system is fast, stable, and accurate. It eliminates common issues found in manual email management, such as slow response times, inconsistencies, human error, and message overload. The cleaning pipeline ensures that unstructured email content is converted into AI-ready text, while the LLM generates human-like, context-aware, and professional replies.

Furthermore, the modular design makes the workflow scalable, maintainable, and adaptable for various real-world scenarios such as customer service, HR inbox management, internship application filtering, IT support, and enterprise communication. With optional extensions like attachment processing, email classification, and priority tagging, this automation provides a strong foundation for developing more advanced AI-driven communication systems.

In conclusion, the integration of RPA and AI offers a powerful solution for modern email management challenges, achieving fast performance, high accuracy, and significant reduction in human workload. This project proves the practical feasibility of combining automation with natural-language intelligence to create reliable and intelligent communication systems.

VIII. FUTURE WORK

While the proposed AI-powered email automation system demonstrates high efficiency, accuracy, and reliability, several enhancements can be incorporated to increase robustness, intelligence, and scalability. Future work may focus on the following key areas:

A. Advanced Email Classification Using ML Models

Currently, the system processes emails sequentially without classifying them into categories. Future enhancement may include integrating machine-learning algorithms such as:

- Support Vector Machines (SVM)
- Logistic Regression
- Transformer-based classifiers
- BERT / DistilBERT fine-tuned models

These models can automatically categorize emails into groups like:

- Internship inquiries
- Technical support
- Business proposals
- Spam or irrelevant messages

This would allow automatic routing and priority-based processing.

B. Sentiment Analysis for Tone Detection

Emails received from users may contain emotions such as urgency, frustration, or appreciation. Future work can involve sentiment analysis models that:

- Detect user tone (positive, negative, neutral)
- Trigger urgency workflows
- Generate responses with tone adaptation
- Alert the human administrator for sensitive cases

This enhances personalization and user experience.

C. Attachment Content Extraction and OCR

Many emails include attachments such as:

- PDFs
- Images
- Word documents
- Scanned forms

Future development can integrate OCR (Optical Character Recognition) technologies like:

- UiPath Document Understanding
- Tesseract OCR
- Google Vision OCR

This enables the system to:

- Extract text from scanned documents
- Read resumes, forms, and invoices
- Automatically summarize attachment content

D. Integration with Databases and CRM Systems

Instead of storing data only in Excel, the system can integrate with:

- SQL Server
- MySQL / PostgreSQL
- Firebase
- CRM tools like Salesforce and Zoho

This upgrade will provide:

- Better scalability
- Real-time dashboards
- Multi-user access
- Long-term analytics

E. Multilingual Email Support

Future versions can integrate multilingual models enabling automation to handle emails in:

- Hindi
- Arabic
- French
- Chinese
- Spanish

And generate responses in the same language, improving global usability.

F. Automatic Email Prioritization System

Enhancement potential includes:

- Assigning priority levels (High, Medium, Low)
- Fast-tracking urgent messages
- Triggering notifications for time-critical emails
- Machine-learning-based priority estimation

This would make the bot more efficient in real-world corporate environments.

G. Voice-Based Email Processing

Future versions could integrate speech-to-text models to support:

- Voice emails
- Meeting recordings
- Audio queries

This expands the bot's reach beyond text-based communication.

H. Full Autonomous Email Agent (No Human Review Required)

Future research can aim to create a fully autonomous digital email agent capable of:

- Learning from previous interactions

- Improving reply tone and accuracy over time
- Customizing replies based on user history
- Scheduling follow-up emails
- Generating reminders
- Tracking unresolved issues

I. Integration with Calendar and Task Automation

Enhancing the system to:

- Schedule meetings
- Set reminders
- Book appointments
- Update Google/Outlook calendars automatically

This transforms the bot into a complete personal or organizational digital assistant.

J. Real-Time Dashboard and Analytics

Future work can include a web-based dashboard that provides:

- Number of emails processed
- Categories of emails received
- AI performance metrics
- Priority email stats
- Response time tracking

This will help organizations visualize productivity gains and maintain transparency.

ACKNOWLEDGMENT

The authors are indebted to the Mae Fah Luang University for supporting the free registration in this conference.

REFERENCES

[1] C. Lüthje and F. Thiele, "Communication floods--Emails in scholarly communication," *SCM Stud. Commun. Media*, vol. 9, no. 3, pp. 367–393, 2020.

[2] N. S. Scott *et al.*, "CPC's 50th Anniversary: celebrating 50 years of open-source software in computational physics," *Comput. Phys. Commun.*, vol. 252, p. 107269, 2020.

[3] E. Milne, *Email and the Everyday: Stories of Disclosure, Trust, and Digital Labor*. MIT Press, 2021.

[4] G. Abromovich, "Study: Contrary To Popular Belief, Email Is Alive And Well," 2018. .

[5] S. Chakravarthy, A. Venkatachalam, and A. Telang, "A graph-based approach for multi-folder email classification," in *Proceedings - IEEE International Conference on Data Mining, ICDM*, Dec. 2010, pp. 78–87, doi: 10.1109/ICDM.2010.55.

[6] N. Jha, D. Prashar, and A. Nagpal, "Combining artificial intelligence with robotic process automation—an intelligent automation approach," in *Deep Learning and Big Data for Intelligent Transportation*, Springer, 2021, pp. 245–264.

[7] M. Ahmet Unal and O. Bolukbas, "The Acquirements of Digitalization with RPA (Robotic Process Automation) Technology in the Vakif Participation Bank," in *2021 The 4th International Conference on Information Science and Systems*, 2021, pp. 68–73.

[8] "Robotic Process Automation | UiPath." https://www.uipath.com/ (accessed Mar. 29, 2021).

[9] J. Wewerka and M. Reichert, "Robotic Process Automation -- A Systematic Literature Review and Assessment Framework," *arXiv*, Dec. 2020.

[10] M. Lacity and L. Willcocks, "Becoming Strategic with Intelligent Automation," *MIS Q. Exec.*, vol. 10, no. 2, pp. 1–14, 2021.

[11] I. Alsmadi and I. Alhami, "Clustering and classification of email contents," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 27, no. 1, pp. 46–57, Jan. 2015, doi: 10.1016/j.jksuci.2014.03.014.

[12] M. A. Naeem, I. W. S. Linggawa, A. A. Mughal, C. Lutteroth, and G. Weber, "A Smart Email Client Prototype for Effective Reuse of Past Replies," *IEEE Access*, vol. 6, pp. 69453–69471, 2018.

[13] S. Park, A. X. Zhang, L. S. Murray, and D. R. Karger, "Opportunities for automating email processing: A need-finding study," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019, pp. 1–12.

[14] M. Peng, B. Gao, J. Zhu, J. Huang, M. Yuan, and F. Li, "High quality information extraction and query-oriented summarization for automatic query-reply in social network," *Expert Syst. Appl.*, vol. 44, pp. 92–101, 2016.

[15] N. K. Bhasin and K. Gulati, "A Study of the Readiness of Indian.

[16] Banks to Absorb COVID-19s Impact Through New Emerging Technologies and Strategies for Competitive Advantage," in *E-Collaboration Technologies and Strategies for Competitive Advantage Amid Challenging Times*, IGI Global, 2021, pp. 50–75.

[17] D. D. A. Bui, G. Del Fiol, and S. Jonnalagadda, "PDF text classification to leverage information extraction from publication reports," *J. Biomed. Inform.*, vol. 61, pp. 141–148, Jun. 2016, doi: 10.1016/j.jbi.2016.03.026.

[18] T. Tam, A. Ferreira, and A. Lourenço, "Automatic foldering of email messages:a combination approach," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2012, vol. 7224 LNCS, pp. 232–243, doi: 10.1007/978-3-642-28997-2_20.

[19] S. Whittaker and C. Sidner, "Email overload: exploring personal information management of email," in *Conference on Human Factors in Computing Systems - Proceedings*, 1996, pp. 276–283, doi: 10.4324/9781315806389-24.

[20] J. D. Brutlag and C. Meek, "Challenges of the Email Domain for Text Classification," *Proceedings of the Seventeenth International Conference on Machine Learning*, 2000. .

[21] R. Bekkerman, A. McCallum, and G. Huang, "Automatic Categorization of Email into Folders : Benchmark Experiments on Enron and SRI Corpora," 2004.

[22] B. Klimt and Y. Yang, "The enron corpus: A new dataset for email classification research," in *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, 2004, vol. 3201, pp. 217–226, doi: 10.1007/978-3-540-30115-8_22.

[23] M. Dredze *et al.*, "Intelligent email: Aiding users with AI," in *Proceedings of the National Conference on Artificial Intelligence*, 2008, vol. 3, pp. 1524–1527.

[24] A. Dalli, Y. Xia, and Y. Wilks, "FASIL email summarisation system," in *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, 2004, pp. 994-es, doi: 10.3115/1220355.1220498.

[25] S. Singh, P. Singh, R. Garg, and P. K. Mishra, "Big Data: Technologies, Trends and Applications," *Int. J. Comput. Sci. Inf. Technol.*, vol. 6, no. 5, pp. 4633–4639, 2015.

[26] V. Leno, A. Polyvyanyy, M. Dumas, M. La Rosa, and F. Maggi, "Robotic process mining: vision and challenges," *Bus. \& Inf. Syst. Eng.*, vol. 63, no. 3, pp. 301–314, 2021.

[27] J. Ribeiro, R. Lima, T. Eckhardt, and S. Paiva, "Robotic Process Automation and Artificial Intelligence in Industry 4.0 – A Literature review," *Procedia Comput. Sci.*, vol. 181, pp. 51–58, Jan. 2021, doi: 10.1016/j.procs.2021.01.104.

[28] D. Rafailidis and Y. Manolopoulos, "The Technological Gap Between Virtual Assistants and Recommendation Systems," *arXiv*, Dec. 2018.

[29] J. G. Enriquez, A. Jimenez-Ramirez, F. J. Dominguez-Mayo, and J. A. Garcia-Garcia, "Robotic Process Automation: A Scientific and Industrial Systematic Mapping Study," *IEEE Access*, vol. 8, pp. 39113–39129, 2020, doi: 10.1109/ACCESS.2020.2974934.

[30] M. Adam, M. Wessel, and A. Benlian, "AI-based chatbots in customer service and their effects on user compliance," *Electron. Mark.*, pp. 1–19, Mar. 2020, doi: 10.1007/s12525-020-00414-7.

[31] D. Rountree, "4 - System Security," in *Security for Microsoft Windows System Administrators*, D. Rountree, Ed. Boston: Syngress, 2011, pp. 109–134.

[32] M. Plummer, "How to Spend Way Less Time on Email Every Day." .