EXPENSE AUTOMATION BOT

A PROJECT REPORT

Submitted by

J.P.SAI SRAVANTHI - 220701902

in partial fulfilment for the course

OAI1903 - INTRODUCTION TO ROBOTIC PROCESS AUTOMATION

for the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

THANDALAM

CHENNAI – 602 105

NOVEMBER 2024

RAJALAKSHMI ENGINEERING COLLEGE

CHENNAI - 602105

BONAFIDE CERTIFICATE

Certified that this project report **"Expense Automation Bot"**
is the bonafide work of **"J.P.Sai Sravanthi(220701902)"** who carried out the project work for the subject OAI1903- Introduction to Robotic Process Automation under my supervision.

Mrs. J. Jinu Sophia, M.E., (Ph.D)

SUPERVISOR

Assistant Professor (SG)

Department of

Computer Science and Engineering

Rajalakshmi Engineering College

Thandalam

Chennai - 602105

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

INTERNAL EXAMINER EXTERNAL EXAMINER

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 **Thiru. S.Meganathan, B.E., F.I.E.,** our Vice Chairman **Mr. M.Abhay Shankar, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) Thangam Meganathan, M.A., M.Phil., 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.Kumar, M.E., Ph.D.,** Professor and Head of the Department of Computer Science and Engineering for his guidance and encouragement throughout the project work. We convey our sincere and deepest gratitude to our internal guides, Mrs. J. Jinu Sophia, M.E., Ph.D., Assistant Professor (SG) Department of Computer Science and Engineering for their valuable guidance throughout the course of the project. We are very glad to thank our Project Coordinator Professor, Dr. N. Duraimurugan, 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.

J.P.SAI SRAVANTHI (220701902)

ABSTRACT

The Expense Automation Bot is an innovative solution designed to streamline financial report generation for large organizations. Built using a combination of Flask, UiPath, and Google Sheets API, this project automates the processing of expense-related invoices and receipts, ensuring accurate and efficient data handling. The system allows users to upload multiple files, including PDFs and Excel documents, through a dynamic web interface. Each organization's data is segregated and processed separately to ensure confidentiality and accuracy.

The web application, developed using Flask, facilitates file uploads and creates a dedicated Google Sheet for each organization, sharing the access link via email. On the backend, UiPath plays a pivotal role by monitoring uploaded files, extracting key details such as invoice numbers, dates, and amounts using OCR and Regex, and appending this data to the respective Google Sheets. The extracted data can also be exported as a CSV file and emailed to the organization for further analysis.

This automation bot eliminates manual data entry, reducing errors and enhancing productivity. It is particularly beneficial for organizations managing high volumes of financial documents. The integration of robust technologies ensures scalability and ease of use, making it a valuable tool for financial reporting automation. Future enhancements include advanced data validation and integration with additional APIs for real-time expense tracking and analysis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Full Form
RPA	**-Robotic Process Automation**
OCR	**-Optical Character Recognition**
BOT	**- Robot or Automation Script**
API	**- Application Programming Interface**
GUI	**- Graphical User Interface**
ETL	**- Extract, Transform, Load**
URL	**- Uniform Resource Locator**
CSV	**- Comma-Separated Values**
XPath	**XML Path Language (used for web scraping/navigation)**

CHAPTER 1: INTRODUCTION

1.1 GENERAL:

Managing organizational expenses is a critical yet challenging task, especially for large organizations handling high volumes of invoices and receipts. The traditional manual processes involved in tracking, organizing, and reporting expenses are time-consuming and prone to errors. Such inefficiencies can lead to financial discrepancies, delays in reporting, and increased operational costs.

The Expense Automation Bot is a dynamic solution that addresses these challenges by leveraging advanced technologies like Robotic Process Automation (RPA), Flask, and Google Sheets API. Designed to automate the expense processing workflow, this system allows users to upload invoices and receipts through an intuitive web interface. Using UiPath, the bot extracts essential details from the uploaded documents, such as invoice numbers, dates, amounts, and payment information, and updates a centralized Google Sheet in real-time.

This project exemplifies the transformative potential of automation in financial management, significantly reducing manual effort while improving speed, accuracy, and scalability. By eliminating repetitive tasks, the Expense Automation Bot enables organizations to focus on strategic activities, enhancing overall operational efficiency.

1.2 OBJECTIVE:

The objective of the Expense Automation Bot project is to create an automated system for processing and managing organizational expenses. Key objectives include:

1. Automating the collection and extraction of data from invoices and receipts.
2. Generating financial summaries and reports dynamically, with real-time updates to Google Sheets.
3. Providing a user-friendly web interface for uploading and processing multiple files simultaneously.
4. Minimizing human intervention to reduce errors and operational overhead.
5. Ensuring secure and confidential handling of organizational data.

The project also aims to demonstrate the potential of RPA and related technologies in transforming traditional expense management into a seamless, efficient process. Ultimately, the system is designed to save time, improve accuracy, and provide actionable insights into financial data for organizations of all sizes.

1.3 EXISTING SYSTEM:

The current process of expense management in most organizations is largely manual, requiring staff to upload, organize, and analyze invoices and receipts. This involves several challenges, including:

- **Time-Consuming Processes**: Manually reviewing and extracting information from invoices is labor-intensive and delays financial reporting.

- **Error-Prone Operations**: Manual data entry increases the likelihood of errors, leading to inaccurate reports and potential financial losses.
- **Scalability Issues**: Handling large volumes of invoices and receipts becomes unmanageable, especially for growing organizations.
- **Lack of Automation**: Existing systems often lack automation, requiring repeated manual efforts for routine tasks such as categorizing expenses and generating summaries.
- **Limited Accessibility**: In some cases, financial data is stored in disparate systems, making it difficult to access or consolidate.

These inefficiencies highlight the need for an automated system that streamlines the expense management workflow while improving speed and accuracy.

1.4 PROPOSED SYSTEM:

The Expense Automation Bot addresses the limitations of the existing system by introducing a fully automated solution that integrates web-based technologies with RPA tools. This proposed system significantly enhances efficiency, accuracy, and scalability in managing organizational expenses.

Key Features of the Proposed System:

1. **Automated Data Extraction**: The system leverages Optical Character Recognition (OCR) and Regex to automatically extract key details such as invoice numbers, dates, amounts, and payment details from uploaded invoices and receipts.
2. **Dynamic Google Sheets Integration**: Extracted data is automatically organized and updated in Google Sheets, allowing real-time access to financial reports. Each organization receives a dedicated Google Sheet, ensuring data confidentiality.

3. **User-Friendly Web Interface**: Built using Flask, the web application enables users to upload multiple files simultaneously. The interface is intuitive, requiring minimal user effort.
4. **Secure and Segregated Data Handling**: Uploaded files are processed separately for each organization, ensuring that data remains isolated and secure.
5. **Email Notifications with Attachments**: Organizations receive an email containing the link to their Google Sheet and a CSV file summarizing their expense data.
6. **Error Reduction and Operational Efficiency**: The automated workflow eliminates manual errors and reduces processing time, allowing staff to focus on strategic activities.
7. **Scalable Design**: The system is designed to handle large volumes of invoices, making it suitable for organizations of all sizes.

CHAPTER 2: LITERATURE REVIEW

Expense automation systems and Robotic Process Automation (RPA) have transformed financial management by automating tasks such as data extraction, validation, and reporting. This review examines key advancements in these technologies and their impact on improving efficiency and accuracy in handling financial documents.

1. Expense Management Systems: Traditional expense management relied heavily on manual data entry, leading to high error rates and inefficiencies (Smith & Johnson, 2017). Modern systems have adopted cloud-based tools like Google Sheets, which simplify data sharing and accessibility (Gupta & Sharma, 2020). While effective, these systems often lack advanced automation for extracting data directly from financial documents, creating opportunities for RPA integration.

2. RPA in Expense Management: RPA has revolutionized expense management by automating repetitive tasks such as invoice processing and report generation. Nguyen et al. (2018) highlighted RPA's ability to reduce errors and processing times. Tools like UiPath, leveraging OCR and regex-based parsing, have proven effective in extracting structured data from invoices (Patel & Srinivasan, 2019). Additionally, RPA's integration with cloud platforms, such as Google Sheets, enables real-time updates to financial records (Rajasekar & Ravichandran, 2020).

3. Challenges in Automation: Inconsistent invoice formats and low-quality document scans remain significant challenges (Agarwal & Mehta, 2020; Sharma et al., 2019). These issues require extensive configuration and occasional human validation. Real-time system integration also poses technical challenges, requiring robust APIs and error-handling mechanisms (Lee et al., 2020). Data security concerns, particularly in cloud-based systems, necessitate encryption and secure authentication methods (Gupta & Singh, 2021).

4. Advances in Automation and Analytics: Emerging technologies like AI-powered analytics and predictive modeling are enhancing automated expense management. Kim & Park (2020) demonstrated how NLP could generate detailed financial summaries, while Lee & Chen (2019) highlighted predictive analytics for forecasting expense trends. Integrating these advancements with RPA could further streamline expense management workflows.

CHAPTER 3: SYSTEM DESIGN

The system design of the Expense Automation Bot provides a comprehensive framework to automate and streamline the process of expense management. The user interacts with the system through a dynamic web interface built using Flask, allowing the upload of multiple invoices and receipts simultaneously. These files are stored securely in organization-specific folders within the backend storage to maintain data segregation. Once files are uploaded, UiPath workflows are triggered to automate the extraction of critical financial details such as invoice numbers, dates, amounts, and payment due dates using advanced OCR and Regex techniques. The extracted data is then organized into a structured format and dynamically updated in Google Sheets, ensuring real-time accessibility and accuracy of financial reports. The system also incorporates a notification mechanism to send an email to the organization with a link to their Google Sheet, along with a CSV file summarizing the extracted data. By integrating robust automation technologies like UiPath and leveraging cloud services such as Google Sheets API, the system ensures accuracy, scalability, and efficiency in processing high volumes of financial data. This design reduces manual effort, enhances data accuracy, and provides users with a reliable and intuitive solution for managing organizational expenses.

3.1.1 SYSTEM FLOW DIAGRAM:

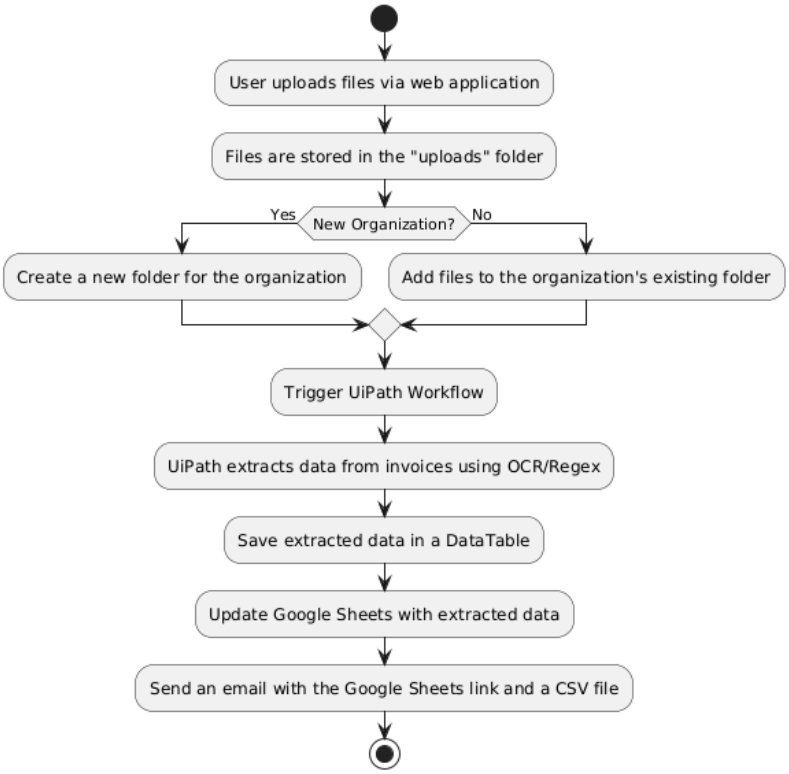

Fig 3.1

Description:

The system flow diagram represents the sequence of operations performed by the Expense Automation Bot.

- It begins with the user uploading financial documents through the web application. These files are stored in the appropriate organization folder.
- If the organization is new, a dedicated folder is created to maintain data segregation.
- A UiPath workflow is triggered to process these files, extract key details such as invoice numbers and amounts using OCR/Regex, and organize the data into a structured format (DataTable).

- The system updates the organization's Google Sheets file with this data and sends an email containing the sheet's link and a CSV file. This flow ensures a seamless and automated expense management process

3.1.2 ARCHITECTURE DIAGRAM:

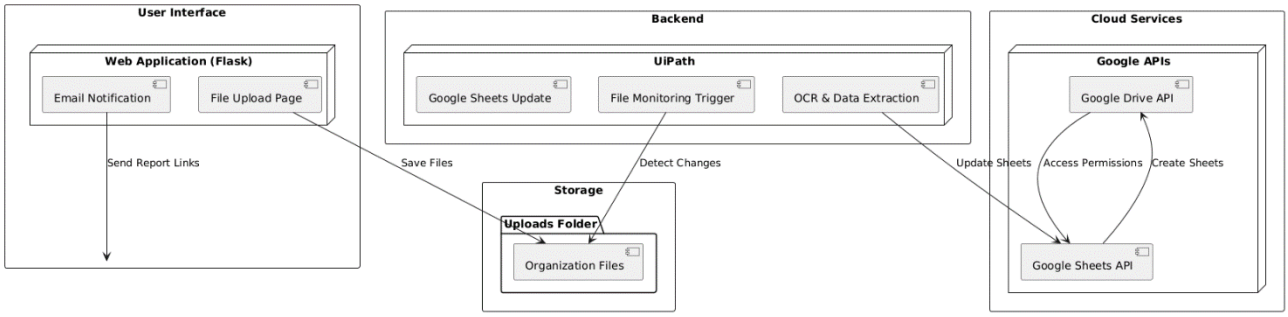

Fig 3.2

Description:

The architecture diagram illustrates the Expense Automation Bot's modular design and component interactions. The system features:

1. **User Interface**: Built using Flask, it includes a web page for uploading invoices and notifying users via email.
2. **Backend**: UiPath automates tasks such as monitoring uploads, extracting data from invoices, and updating Google Sheets.
3. **Cloud Services**: Google APIs (Sheets and Drive) handle data storage and sharing, ensuring secure and real-time access to financial reports.
4. **Storage**: The system organizes uploaded files in the "uploads" folder, creating separate directories for each organization to maintain confidentiality.

3.1.3 SEQUENCE DIAGRAM:

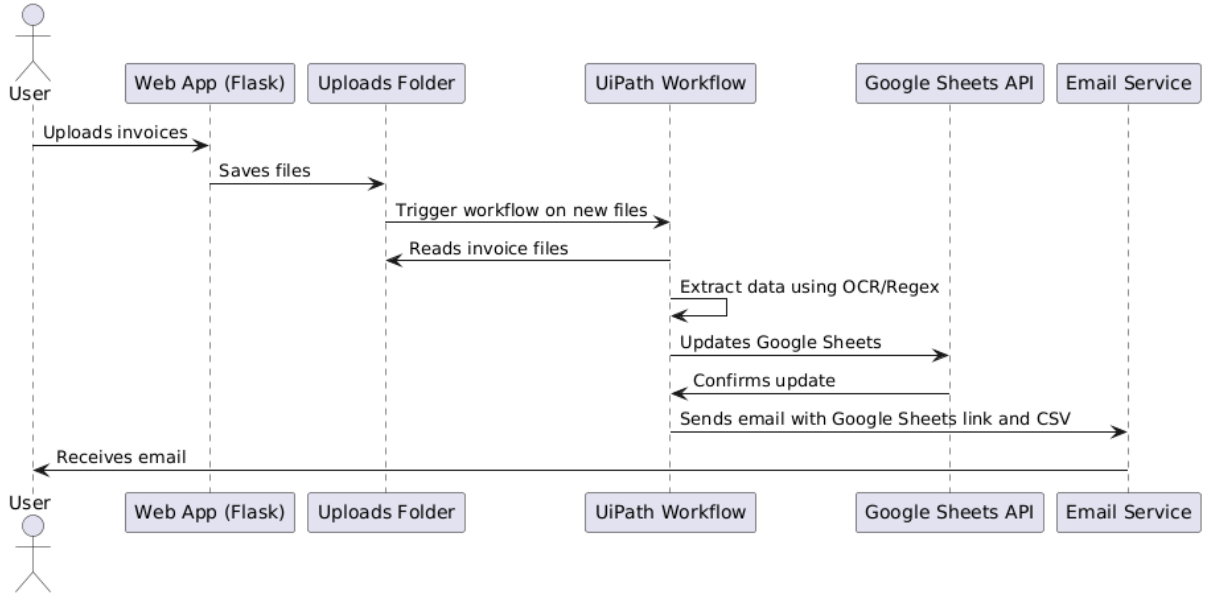

Fig 3.3

Description:

The sequence diagram details the interactions between the user and system components during the expense automation process:

1. **User Interaction**: The user uploads invoices through the web application.
2. **File Storage**: The application saves the uploaded files in the appropriate organization's folder.
3. **Workflow Trigger**: UiPath detects new files and initiates data extraction.
4. **Data Extraction**: UiPath reads the invoices using OCR/Regex to extract key information.
5. **Google Sheets Update**: The extracted data is appended to the corresponding Google Sheet.
6. **Email Notification**: A notification email is sent to the user, containing a link to the Google Sheet and a CSV summary of the extracted data.

CHAPTER 4: PROJECT DESCRIPTION

The Expense Automation Bot is an RPA-based automation solution designed to streamline and automate the process of extracting, processing, and reporting financial data from uploaded invoices and receipts. By using advanced technologies like UiPath, Flask, and Google Sheets API, the system automates the tedious tasks of data entry, categorization, and report generation. The system allows organizations to upload multiple invoices, and through the use of Optical Character Recognition (OCR) and data extraction techniques, the bot efficiently extracts key information such as invoice numbers, amounts, and payment dates, updating the data directly into Google Sheets. This solution aims to reduce manual effort, increase data accuracy, and provide real-time financial reporting.

4.1 METHODOLOGIES:

The methodology employed for the Expense Automation Bot follows the Robotic Process Automation (RPA) lifecycle, utilizing UiPath to automate the extraction, processing, and reporting of financial data.

Step 1: Requirement Gathering: Collect user requirements, including types of invoices to process (e.g., expense invoices, receipts), necessary fields for extraction (e.g., invoice number, date, amount), and target systems (e.g., Google Sheets) for report generation. Define the organizational requirements, including how data is stored, and which departments or users will be accessing the reports.

Step 2: Process Design: Design the process flow that includes all necessary steps: file upload (via web interface), triggering data extraction (via UiPath), extracting and structuring the data, updating Google Sheets, and sending email notifications. Use UiPath to create a flowchart and sequence diagram to map out the various activities, decisions, and the sequence of operations.

Step 3: Data Extraction (OCR & Regex): Using UiPath, create workflows to automate the extraction of key details from invoices and receipts using Optical Character Recognition (OCR) and Regular Expressions (Regex). The process will involve automating interactions with file storage, extracting data such as invoice numbers, dates, amounts, and payment details from the documents.

Step 4: Data Categorization and Validation: Once the data is extracted, categorize the information based on predefined categories (e.g., invoice number, amount, date, payment method). Use validation rules within UiPath to ensure data accuracy. For example, checking for missing or incorrect values, ensuring data consistency, and handling edge cases such as unreadable data or errors from OCR.

Step 5: Data Storage and Reporting: Store the validated and processed data in a structured format, such as a Google Sheets document, allowing real-time access to the generated reports. The system will update Google Sheets with the extracted data, ensuring that each organization has its own separate file, preventing data mixing from different organizations. Automatically generate a CSV file with a summary of the processed data for each organization.

Step 6: Notification and Result Presentation: Once the data has been successfully extracted and processed, an email notification will be sent to the organization, providing them with a link to their Google Sheet, along with the CSV report attached for offline access.

Step 7: Testing and Optimization: Test the entire automation process. This includes testing the OCR extraction, validation of the data, and the updating of Google Sheets. Optimize the system for performance, ensuring the process works efficiently even with large volumes of invoices or receipts. Address any issues related to different file formats or OCR inaccuracies.

Step 8: Deployment and Maintenance: Deploy the solution to production, where users can upload invoices and automatically receive accurate financial reports in real-time. Provide regular maintenance and updates to the system, such as incorporating improvements to the OCR extraction or handling new invoice formats. Update workflows periodically to ensure compatibility with changes in document formats or the Google Sheets API.

4.1.1 MODULES:

The Expense Automation Bot project can be divided into several key modules, each responsible for specific tasks in the process of automating expense data extraction, processing, and reporting. Below are the modules of the system:

Module 1: User Interface (UI) Module

- **Purpose**: To collect user inputs, such as uploading financial documents (invoices, receipts) and providing organizational details.
- **Features**:
    - A simple and intuitive user interface (can be built in Flask or UiPath with custom forms or input dialogs).
    - Provides the ability to upload multiple files at once.
    - Validates the data provided by the user (e.g., ensuring that uploaded files are in PDF or Excel format).
    - Offers feedback on successful or failed file uploads and system processing

Module 2: File Management and Storage Module

- **Purpose**: To manage and store uploaded files, ensuring they are properly segregated for each organization.
- **Features**:
    - Automatically saves files in organization-specific folders within a structured file storage system.
    - Ensures that new folders are created for new organizations and that the data for each organization is stored separately.
    - Handles the organization and archiving of documents for easy retrieval and management.

Module 3: Data Extraction Module

- **Purpose**: To extract relevant financial data from invoices and receipts using Optical Character Recognition (OCR) and Regular Expressions (Regex).
- **Features**:
    - Uses UiPath to automate the extraction of essential information from uploaded invoices (e.g., invoice number, dates, total amounts, tax amounts).
    - Applies OCR to PDF or scanned images of invoices and receipts to capture text.
    - Regex patterns are used to match and extract the required fields (e.g., invoice numbers, payment amounts).
    - Handles varying document formats and ensures accurate data extraction.

Module 4: Data Validation and Categorization Module

- **Purpose**: To validate and categorize the extracted data based on predefined rules and categories.
- **Features**:
    - Validates the extracted data to check for errors or inconsistencies (e.g., missing invoice numbers, mismatched dates).
    - Categorizes the data into relevant financial categories (e.g., total amounts, taxes, discounts) for easier reporting.
    - Identifies and flags incomplete or erroneous invoices for further processing or review.

Module 5: Data Processing and Storage Module

- **Purpose**: To organize the extracted data and store it in a structured format (e.g., Google Sheets).
- **Features**:
    - Automatically updates Google Sheets with the processed data.
    - Segregates data for each organization and ensures that each organization has its own dedicated Google Sheet.
    - Optionally stores the processed data in a local or cloud database for backup or future analysis.
    - Provides the ability to export the data to CSV or other formats for further analysis.

Module 6: Report Generation and Notification Module

- **Purpose**: To generate real-time financial reports and send notifications to the relevant stakeholders.
- **Features**:
    - Generates a detailed summary report of expenses based on the extracted data.
    - Sends automated email notifications to the organization, containing a link to the updated Google Sheet and an attached CSV report.
    - Optionally allows for the creation of downloadable reports that summarize financial data for the user's records.

Module 7: Logging and Monitoring Module

- **Purpose**: To track and log system actions, errors, and report generation activities.
- **Features**:
    - Maintains logs of file uploads, data extraction processes, and report generation activities.
    - Records any errors or exceptions that occur during the data processing workflow.
    - Provides a monitoring dashboard for system administrators to track progress and performance metrics.

Module 8: Error Handling and Recovery Module

- **Purpose**: To handle errors gracefully and ensure that the system can recover from failures without disrupting the workflow.
- **Features**:
    - If a document fails to be processed (e.g., OCR fails or data extraction is incomplete), the system retries the operation or notifies the user of the error.
    - Implements automatic retry mechanisms for failed processes or workflows.
    - Ensures that the system continues processing other files even if one fails, reducing downtime and improving reliability.
    - Logs errors and provides a mechanism for administrators to manually intervene when needed.

CHAPTER 5: IMPLEMENTATION AND OUTPUT

5.1 IMPLEMENTATION WORKFLOW

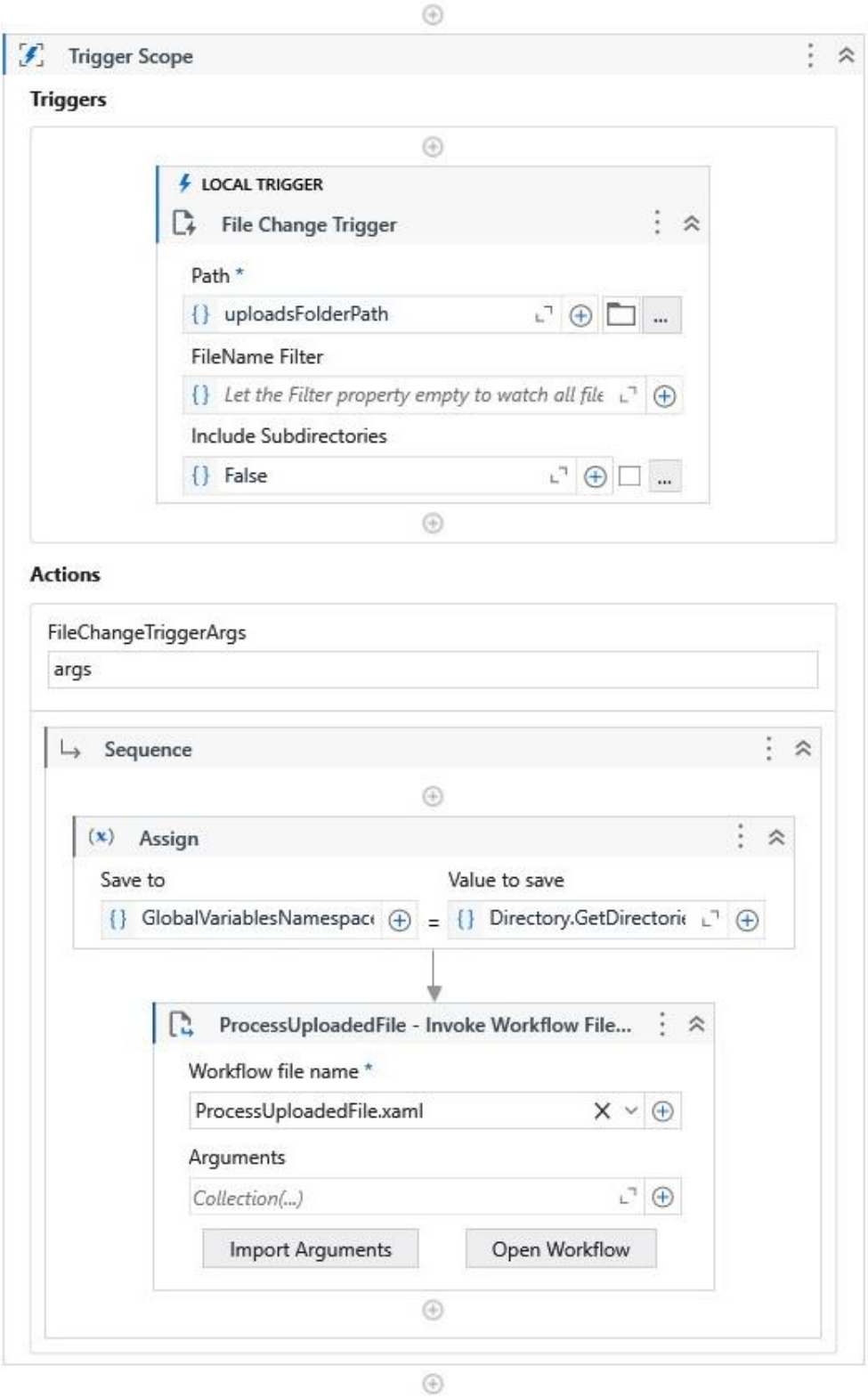

Fig 5.1

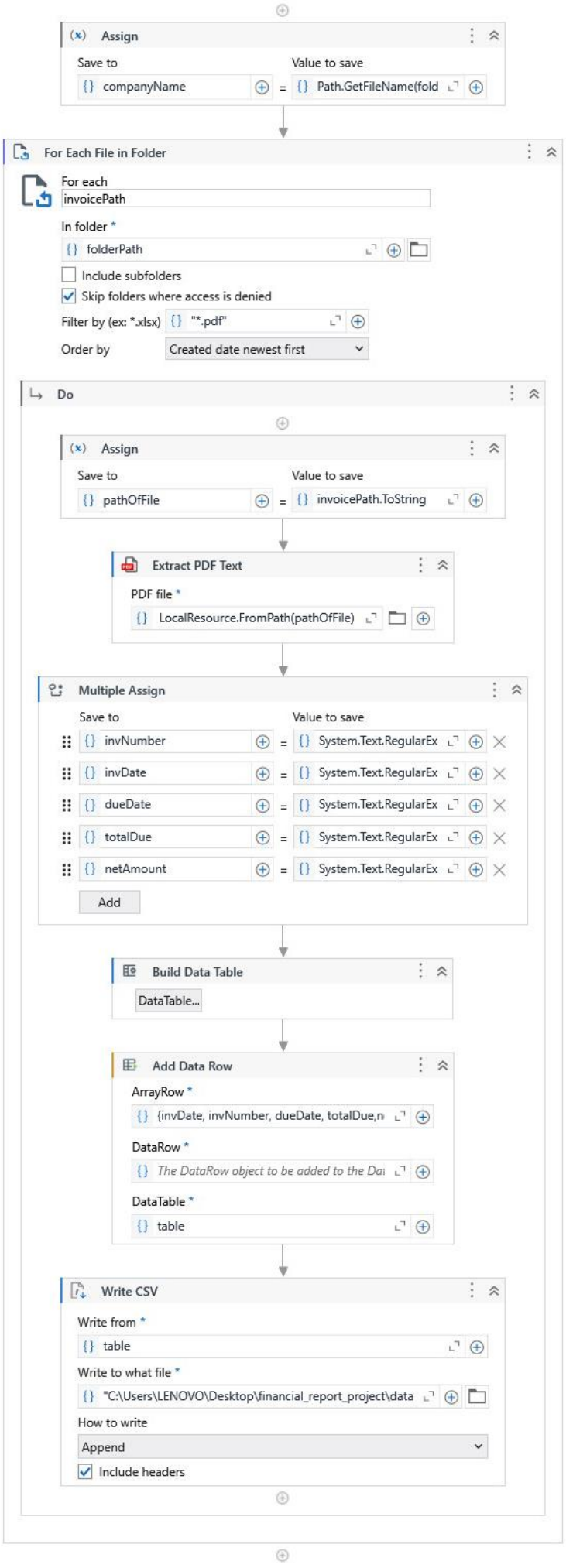
Assign
Save to
companyName
Value to save
Path.GetFileName(fold
For Each File in Folder
For each
invoicePath
In folder *
folderPath
Include subfolders
Skip folders where access is denied
Filter by (ex: *.xlsx)
"*.pdf"
Order by
Created date newest first
Do
Assign
Save to
pathOfFile
Value to save
invoicePath.ToString
Extract PDF Text
PDF file *
LocalResource.FromPath(pathOfFile)
Multiple Assign
Save to
Value to save
invNumber
invDate
dueDate
totalDue
netAmount
System.Text.RegularEx
Add
Build Data Table
DataTable...
Add Data Row
ArrayRow *
{invDate, invNumber, dueDate, totalDue,n
DataRow *
The DataRow object to be added to the Dai
DataTable *
table
Write CSV
Write from *
table
Write to what file *
"C:\Users\LENOVO\Desktop\financial_report_project\data
How to write
Append
Include headers

Fig 5.2

5.2. OUTPUT

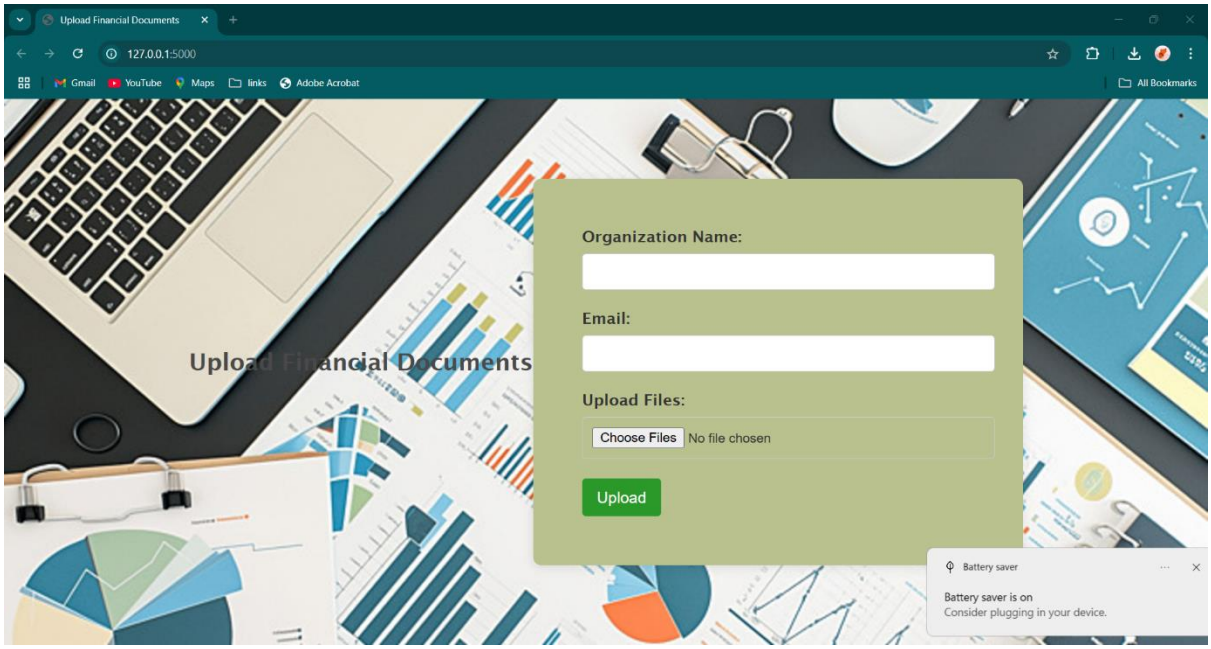

Fig 5.3

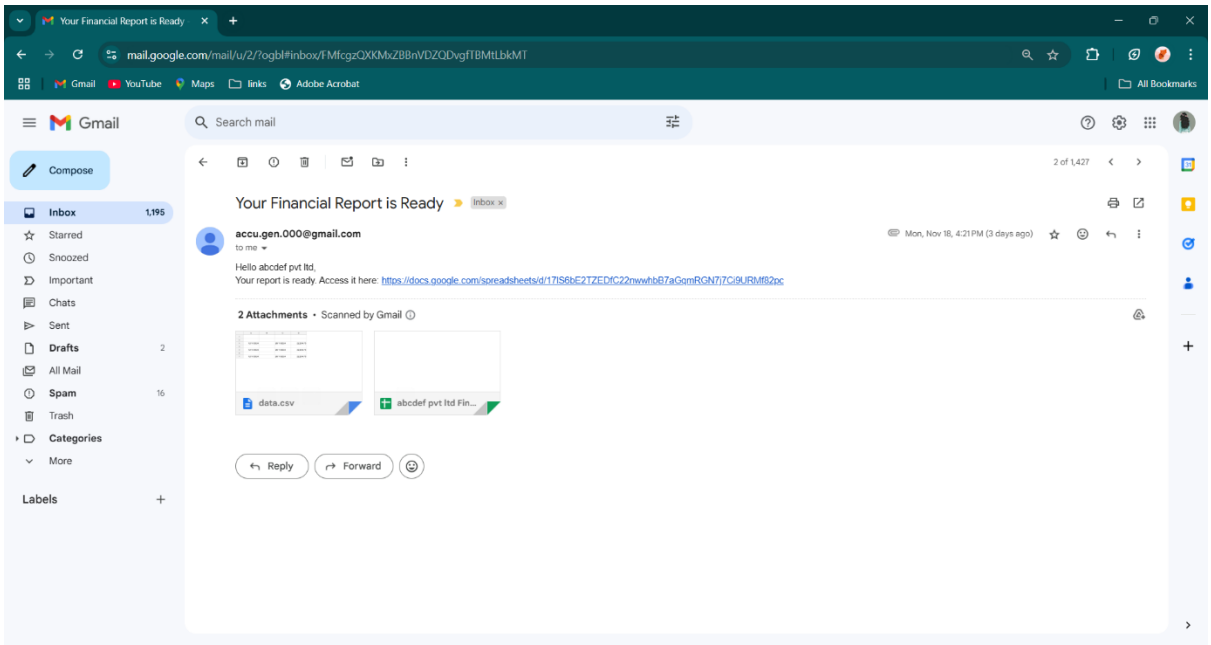

Fig 5.4

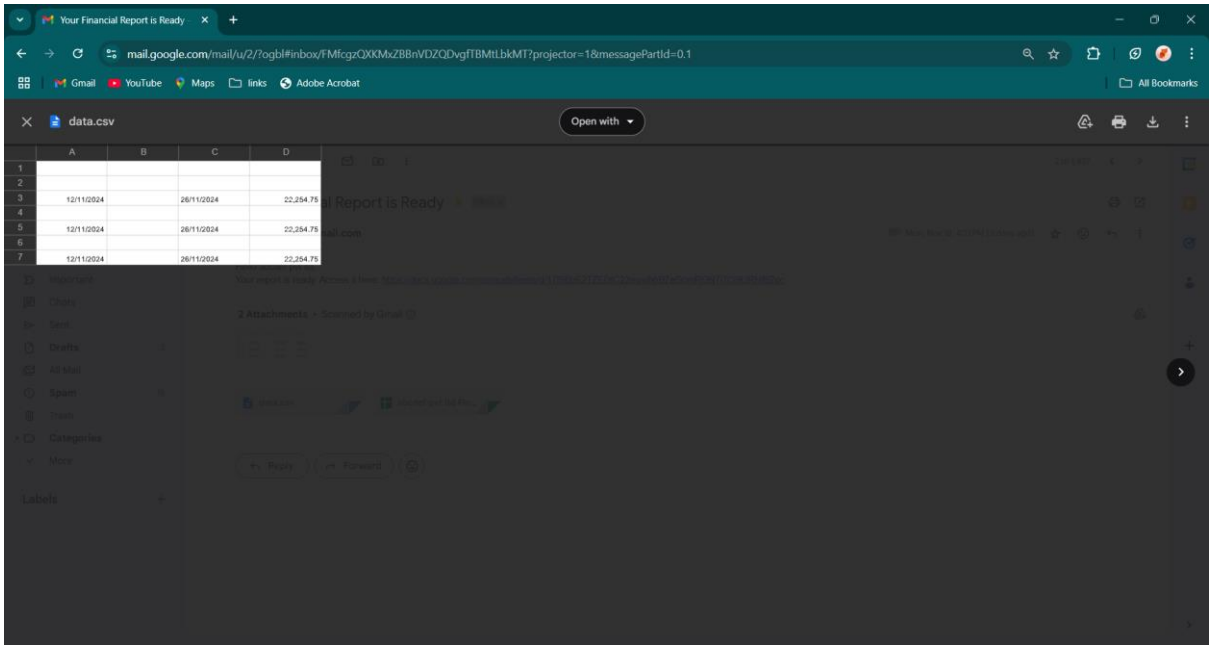
Your Financial Report is Ready
mail.google.com/mail/u/2/?ogbl#inbox/FMfcgzQXKMxZBBnVDZQDvgfTBMttLbkMT?projector=1&messagePartId=0.1
Gmail
YouTube
Maps
links
Adobe Acrobat
All Bookmarks
data.csv
Open with
A
B
C
D
12/11/2024
26/11/2024
22,254.75
12/11/2024
26/11/2024
22,254.75
12/11/2024
26/11/2024
22,254.75

Fig 5.5

CHAPTER 6: CONCLUSION

The Expense Automation Bot successfully demonstrates the power of Robotic Process Automation (RPA) in transforming the traditional methods of financial data extraction and reporting. By automating the process of extracting data from invoices and receipts, validating it, and updating it in real-time on platforms like Google Sheets, the system significantly reduces manual effort, minimizes human error, and accelerates financial reporting. This project uses UiPath to automate key tasks such as Optical Character Recognition (OCR) for text extraction, data categorization, and report generation. Additionally, by leveraging cloud-based platforms like Google Sheets, the system ensures real-time, secure, and easy access to financial data.

The implementation of this bot ensures scalability, reliability, and efficiency, making it an invaluable tool for organizations of all sizes. By automating repetitive tasks, the system frees up valuable human resources for more strategic activities, such as financial analysis and decision-making. The project sets the stage for future enhancements, including integration with more platforms, advanced analytics, and AI-based predictions. Overall, the Expense Automation Bot exemplifies how RPA can optimize operational workflows, improve accuracy, and enhance decision-making in financial management.

6.1. REFERENCES

- **UiPath Documentation**
  Official UiPath documentation for RPA development, covering topics such as OCR, automation workflows, data extraction, and error handling.
  https://docs.uipath.com
- **Google Sheets API**
  Official documentation for Google Sheets API, used for managing and updating financial data in Google Sheets.
  https://developers.google.com/sheets/api
- **UiPath Academy**
  A resource for learning UiPath tools, including tutorials on web scraping, data processing, and workflow automation.
  https://academy.uipath.com/
- **OCR Technology**
  Research and applications on Optical Character Recognition (OCR) for document processing and text extraction.
  https://en.wikipedia.org/wiki/Optical_character_recognition
- **Google Drive API**
  Documentation for Google Drive API
  https://developers.google.com/drive/api
- **RPA Process Design Best Practices**
  General guidelines for designing and optimizing RPA processes for real-world applications.
  https://www.uipath.com/rpa/robotic-process-automation
- **Machine Learning for Expense Categorization**
  Research on the application of machine learning to categorize and analyze financial data.
  https://www.geeksforgeeks.org/

6.2. APPENDICES

Appendix A: Key Components of the System

- **User Interface (UI)**: A simple, user-friendly web interface built using Flask, allowing users to upload invoices and receipts and provide organizational details. The interface validates the uploaded files and offers feedback on successful or failed processing.
- **Data Extraction Layer**: Uses UiPath for extracting key details (invoice number, amount, date, etc.) from uploaded invoices using OCR and Regex. The module ensures accurate and structured data extraction, even from varied formats.
- **Data Validation and Categorization Logic**: Validates the extracted data to ensure consistency and accuracy. It categorizes the data into relevant categories (e.g., total amounts, tax amounts, payment methods) for easier reporting and analysis.
- **Google Sheets Integration**:Updates Google Sheets in real-time with the processed data, ensuring each organization has its own dedicated file, preventing data mixing. Provides secure access and allows organizations to track their expenses effectively.
- **Email Notification System**: Sends emails with a link to the Google Sheets document and an attached CSV report summarizing the extracted data.

Appendix B: Process Flow Summary

1. **Input Gathering**: Users upload invoices and receipts via the web interface, providing organizational details.
2. **Data Extraction**: UiPath workflows automatically extract key information from the uploaded documents using OCR and Regex techniques.
3. **Data Validation** :The extracted data is validated for accuracy and completeness before being categorized and stored.
4. **Data Storage and Reporting**: The processed data is saved in Google Sheets for real-time access and tracking.
5. **Email Notification**: An email notification with the Google Sheets link and CSV summary is sent to the organization.

Appendix C: Tools and Technologies

- **UiPath Studio**:
  Primary tool for designing and executing RPA workflows.
- **Google Sheets API**:
  Used for managing the data and interacting with Google Sheets.
- **Google Drive API**:
  Provides secure cloud storage for uploaded invoices and reports.
- **Flask Framework**:
  Used to build the web interface for file uploads and notifications.
- **OCR (Optical Character Recognition)**:
  Used to extract text from scanned invoices and receipts.
- **Python**:
  Backend programming language for handling file management, email notifications, and integrations with the Google APIs.

Appendix D: Potential Enhancements

1. **Integration with Accounting Software**: Future integration with accounting software such as QuickBooks or Xero could automate the entire workflow from invoice upload to report generation.
2. **Advanced Analytics**: Incorporating machine learning to analyze spending patterns, predict future expenses, and provide actionable insights to the organization.
3. **AI-Based Data Validation**: Using AI to improve data extraction accuracy, especially for handwritten or poorly formatted invoices.
4. **Multi-Channel Support**: Expanding the system to support other document types and file formats, such as scanned receipts or emailed invoices.
5. **Real-Time Notifications**:Implementing real-time notifications for organizations when new expenses are processed or when the budget threshold is reached.