

MEDICAL REPORT AUTOMATION

A PROJECT REPORT

Submitted by

SANTHOSH KUMAR R (220701253)

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

RAJALAKSHMI NAGAR THANDALAM CHENNAI – 602

105

NOVEMBER 2024

**RAJALAKSHMI ENGINEERING COLLEGE
CHENNAI - 602105**

BONAFIDE CERTIFICATE

Certified that this project report “**MEDICAL REPORT AUTOMATION**” is the bonafide work of “**SANTHOSH KUMAR R (220701253)**” who carried out the project work for the subject OAI1903- Introduction to Robotic Process Automation under my supervision.

SIGNATURE

Dr.N.Duraimurugan, M.E., Ph.D, SUPERVISOR,

Assistant Professor (SG),

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 automation of medical report generation simplifies and accelerates the process of creating and distributing personalized patient reports. This project utilizes UiPath RPA to streamline the integration of patient data from an Excel sheet into a pre-designed Word template, automatically generating and emailing the completed reports. This approach reduces manual effort, eliminates errors, and ensures timely delivery.

By leveraging the capabilities of UiPath, the system automates repetitive tasks, such as reading patient information, populating data into templates, and managing email communications. This not only saves time but also allows medical staff to focus on more critical tasks, improving overall efficiency. The solution is easy to implement, scalable for varying workloads, and ensures that patients receive accurate, timely, and professional communication from their healthcare providers.

Additionally, this approach significantly minimizes the risk of human error, which is critical in medical documentation where precision is paramount. It also enhances the security and confidentiality of patient data through automated workflows with restricted access. The project can be seamlessly adapted to handle various types of medical records, making it versatile for different healthcare setups. By integrating automation into this process, healthcare providers can improve service quality and patient satisfaction while reducing administrative burdens. Ultimately, this solution underscores the potential of RPA in revolutionizing healthcare documentation practices.

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 guide, **Dr.N.Durai Murugan, M.E., Ph.D.**, Associate Professor, 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 Coordinator, **Mr.B.Bhuvaneswaran, M.E.**, Assistant Professor (SG), and Supervisor **Mrs. G.M. Sasikala, M.E., Ph.D** Department of Computer Science and Engineering for his useful tips during our review to build our project.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	iii
	LIST OF FIGURES	vi
	LIST OF ABBREVIATIONS	vi
1.	INTRODUCTION	1
	1.1 GENERAL	1
	1.2 OBJECTIVE	
	1.3 EXISTING SYSTEM	2
	1.4 PROPOSED SYSTEM	2
2.	LITERATURE REVIEW	3
	2.1 GENERAL	3
3.	SYSTEM DESIGN	5
	3.1 GENERAL	5
	3.1.1 SYSTEM FLOW DIAGRAM	6
	3.1.2 ARCHITECTURE DIAGRAM	7
	3.1.3 SEQUENCE DIAGRAM	8
4.	PROJECT DESCRIPTION	9
	4.1 METHODOLOGIE	9
	4.1.1 MODULES	11
5.	CONCLUSION	14
	5.1 GENERAL	14
	5.2 REFERENCES	16
	5.3 APPENDICES	21

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
3.1	SYSTEM FLOW DIAGRAM	10
3.2	ARCHITECTURE DIAGRAM	12
3.3	SEQUENCE DIAGRAM	14

LIST OF ABBREVIATIONS

ABBREVIATION	DEFINITION
API	Application Programming Interface
CRM	Customer Relationship Management
ERP	Enterprise Resource Planning
OCR	Optical Character Recognition
IDE	Integrated Development Environment
UML	Unified Modeling Language
UI	User Interface
LMS	Learning Management System

CHAPTER 1

INTRODUCTION

1.1 General

The management of medical reports is a critical aspect of healthcare operations, directly impacting patient care, administrative workflows, and overall service quality. Traditionally, the process of preparing, formatting, and sharing medical reports has been performed manually. These methods, while standard in many institutions, are often time-consuming, labor-intensive, and prone to human error. The repetitive nature of these tasks places an unnecessary burden on administrative staff, leaving them with less time to focus on patient-related priorities. Errors or delays in report preparation and distribution can lead to compromised patient care, inefficiencies in administrative processes, and diminished satisfaction for both patients and healthcare providers. This project introduces an automated solution to streamline these operations, replacing manual efforts with a faster, more reliable, and error-free system.

1.2 Objective

The primary objectives of this project are:

1. **To enhance accuracy and speed in report generation:** The system minimizes the chances of errors and reduces the time required to complete each report.
2. **To alleviate the workload on administrative staff:** Automating repetitive tasks allows healthcare staff to focus on more critical responsibilities, improving overall productivity.
3. **To ensure timely communication with patients:** By generating and delivering reports promptly, the system supports better patient care and satisfaction.

4. **To offer a scalable and customizable solution:** This system can adapt to different healthcare setups and handle a variety of medical report formats.

1.3 Existing System

In many healthcare institutions, the preparation of medical reports relies on manual data entry. This process involves collecting patient data, filling out report templates, and emailing the final documents individually. These tasks are time-intensive and highly prone to errors, such as incorrect data entry or misplaced information, which can compromise patient care. The delays caused by manual processing also hinder timely report delivery, potentially affecting diagnosis, treatment, and overall healthcare operations. Moreover, the reliance on manual effort limits the scalability of the system, as the process becomes increasingly inefficient when handling large volumes of patient data.

1.4. Proposed System

The proposed system aims to revolutionize medical report generation by integrating UiPath RPA (Robotic Process Automation) with Microsoft Excel and Word. This automated solution extracts patient data from an Excel file and uses it to populate a pre-designed Word template, replacing placeholders with accurate information. Once the report is finalized, the system emails it directly to the intended recipients, ensuring quick and precise delivery.

Key features of the proposed system include:

- **Scalability:** The system can handle high volumes of data and adapt to various report formats.
- **Ease of use:** Simple, user-friendly design ensures seamless implementation in healthcare settings.

CHAPTER 2 LITERATURE REVIEW

2.1 General

The automation of report generation and data management processes in healthcare has gained significant importance in recent years. Manual preparation of medical reports is time-consuming, prone to errors, and often delays the delivery of critical information to patients and healthcare providers. These challenges necessitate the adoption of Robotic Process Automation (RPA) to streamline the process and ensure timely, accurate reporting.

RPA tools like UiPath are well-suited for automating workflows that involve repetitive tasks such as data entry, document generation, and email communications. Studies have demonstrated the effectiveness of RPA in reducing the workload of administrative staff in healthcare institutions while ensuring error-free documentation.

Automation of medical reports enhances efficiency by integrating patient data into standardized templates, generating reports, and sending them through email without human intervention. Furthermore, by utilizing pre-built activities in UiPath, the development of such automation workflows becomes more intuitive and scalable.

Despite its advantages, implementing RPA for medical report automation presents challenges such as data security, ensuring compliance with healthcare regulations, and maintaining the accuracy of dynamic templates. Regular updates and monitoring mechanisms are crucial to address these challenges and sustain a reliable automation system.

Overall, the literature highlights the transformative potential of RPA in healthcare, enabling institutions to improve service delivery and operational efficiency. By automating medical report generation, this project demonstrates a practical application of RPA to address real-world challenges in the healthcare sector.

CHAPTER 3

SYSTEM DESIGN

3.1 General

The proposed system for the Automated Cryptocurrency Price Monitoring & Alert System is designed to provide users with real-time alerts based on market fluctuations. It integrates RPA technology with the Gemini API to fetch live cryptocurrency data, analyze it against userdefined thresholds, and trigger notifications. The system architecture ensures scalability, efficiency, and user-friendliness, making it a reliable solution for cryptocurrency traders and enthusiasts.

3.1.1 System Flow Diagram

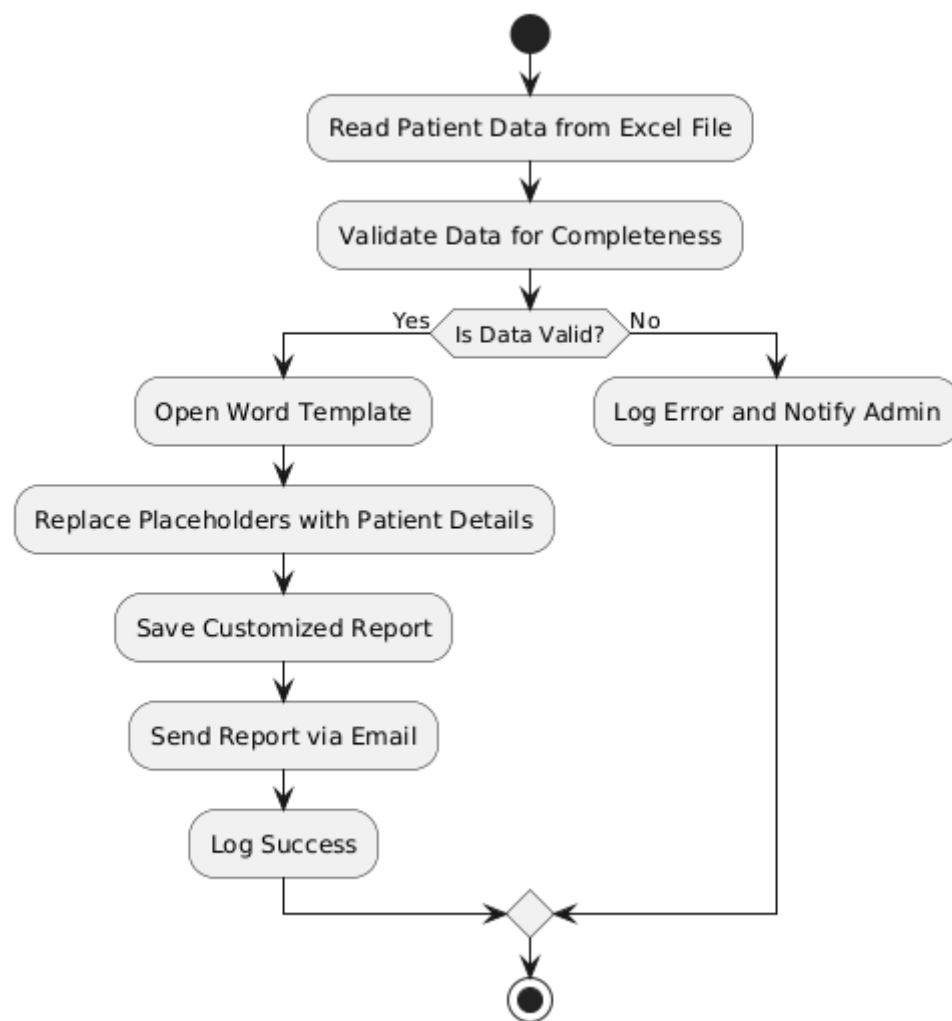


Fig 3.1.1 System Flow Diagram

3.1.2 Architecture Diagram

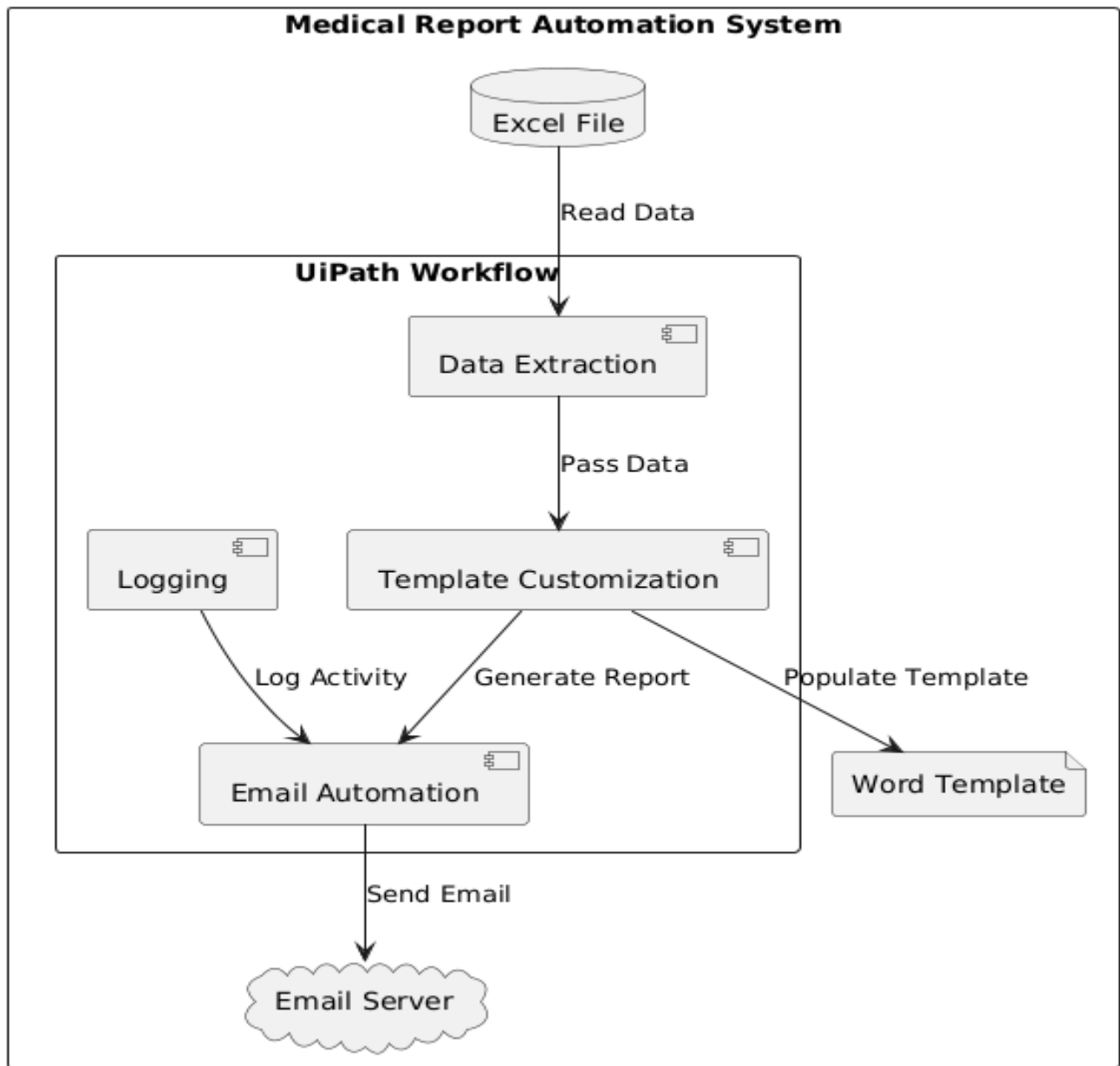


Fig 3.1.2 Architecture Diagram

3.1.3 Sequence Diagram

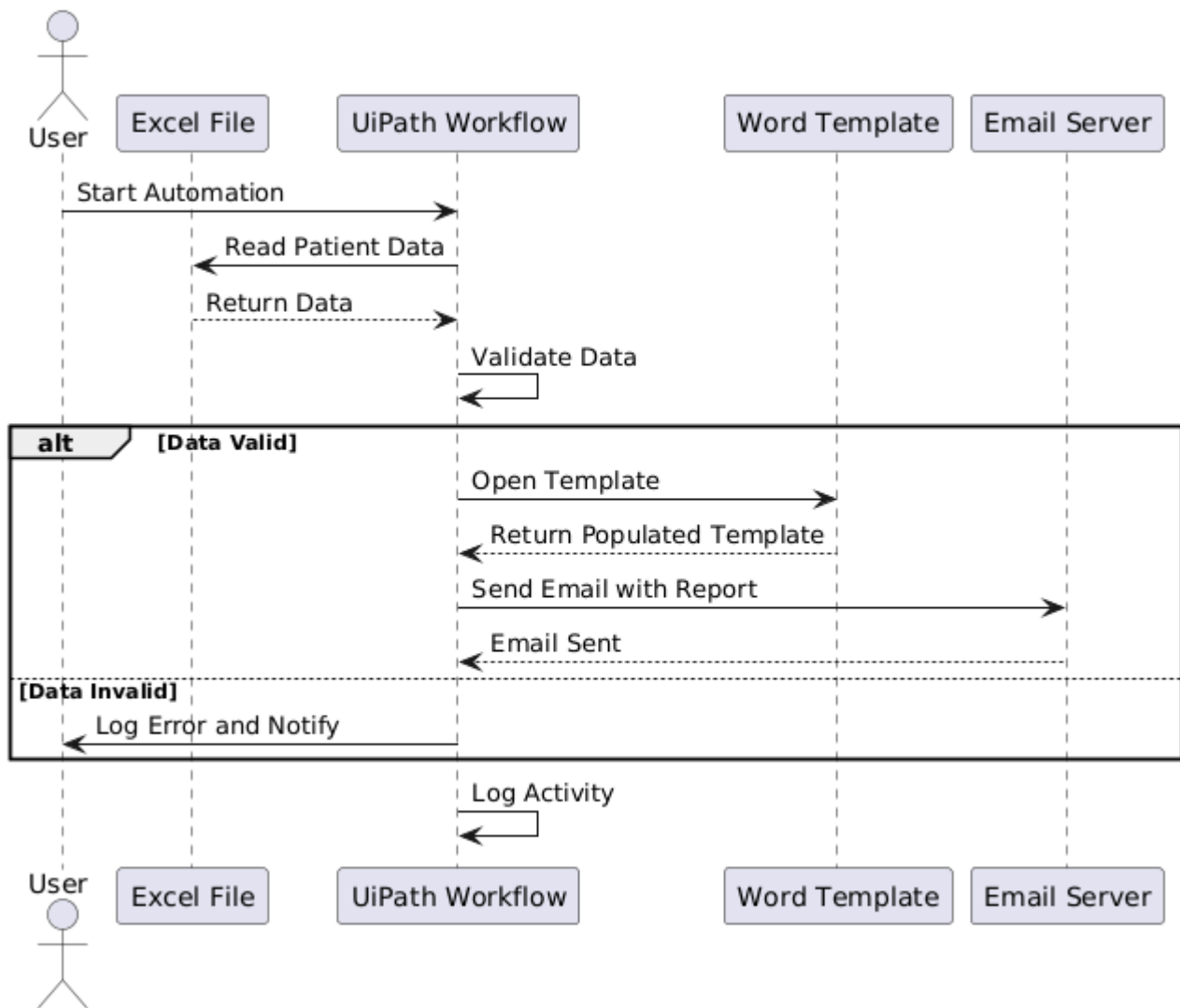


Fig 3.1.3 Sequence Diagram

CHAPTER 4

PROJECT DESCRIPTION

4.1 Methodology

The methodology section outlines the approach taken to develop the **Medical Report Automation System**. This project follows a systematic process to ensure high performance, accuracy, and scalability. The development process is divided into the following key stages:

1. Requirement Analysis & Setup

- Identify patient data fields required for the medical report (e.g., Name, Age, Diagnosis, Prescription).
- Set up UiPath Studio with necessary libraries (e.g., UiPath.Excel.Activities, UiPath.Word.Activities, UiPath.Mail.Activities).
- Create a pre-defined Word template with placeholders for patient data..

2. Data Source Setup

- Store patient details in an Excel sheet, including fields such as Name, Age, Diagnosis, and Email.
- Ensure the Excel file is structured to map seamlessly with the placeholders in the Word template.

3. Template Customization

- Use UiPath Word Activities to replace placeholders in the Word template with data from the Excel file.
- Implement error handling to ensure all placeholders are replaced with accurate information.

4. Email Notification

- Configure UiPath to automatically send the generated report as an email attachment to the respective patient.
- Use UiPath Email Activities, such as **Send SMTP Mail Message**, to manage email delivery.

5. Logging and Reporting

- Log each step of the process, including data retrieval, template customization, and email dispatch, using UiPath's Log Message activity.
- Generate periodic reports summarizing successful and failed operations.

6. Logging and Reporting

- Log each step of the process, including data retrieval, template customization, and email dispatch, using UiPath's Log Message activity.
- Generate periodic reports summarizing successful and failed operations.

7. Post-Processing (Optional)

- Store copies of generated reports in a database or cloud storage for future reference.
- Implement additional workflows to analyze report data or integrate it with a hospital management system.

8. Testing & Optimization

- Test the automation with different Excel datasets to ensure accuracy and reliability.
-
- Optimize the workflow for speed and efficiency by handling edge cases, such as missing data or invalid email addresses.

9. Scheduling and Deployment

- Use UiPath Orchestrator to schedule the automation for regular execution (e.g., daily or weekly).
- Deploy the system in a production environment with secure configurations for data handling and email management.

4.1.1 Modules

The project is divided into the following modules:

1. 1. Data Extraction

- **Objective:** Extract patient details from the Excel file. sources.
- **Activities:**
- Use **Excel Application Scope** to read the Excel file.
- Retrieve data from specific cells using **Read Range** and store it in a DataTable.
- Handle missing or incomplete data with appropriate validations.

2. Price Threshold Validation

- **Objective:** Compare fetched prices with user-defined thresholds to trigger alerts.
- **Activities:**
- Use **Replace Text in Document** activity to replace placeholders with patient-specific data.
- Save the customized document for each patient.

3. Alert Notification

- **Objective:** Send notifications to users when the price threshold is breached.
- **Activities:**
 - Configure **Send Outlook Mail Message** or **Send SMTP Mail Message** to send emails.
 - Attach the customized medical report to the email.
 - Ensure the email contains relevant details in the subject and body for clarity.

4. Error Handling & Logging

- **Objective:** Manage unexpected issues during data retrieval and notification processes.
- **Activities:**
 - Use **Try-Catch** blocks to handle exceptions, such as file not found or invalid email addresses.
 - Log each error using **Log Message** for troubleshooting.
 - Implement retry mechanisms for failed operations.

5. Data Logging & Reporting

- **Objective:** Maintain logs of the automation process for analysis and auditing.
- **Activities:**
 - Use **Write Range** to record the status of each operation in a new Excel sheet.
 - Generate summary reports for successful and failed email dispatches.

6. Post-Processing (Optional)

- **Objective:** Save reports for future use or integration.
- **Activities:**
 - Store generated reports in a centralized repository or cloud storage.
 - Create backup workflows to archive older reports automatically.

7. Scheduling & Deployment

- **Objective:** Ensure the system runs periodically without manual intervention.
- **Activities:**
 - Schedule the automation using **UiPath Orchestrator** for daily or event-based triggers.
 - Monitor execution logs and handle failures promptly.

CHAPTER 5

CONCLUSIONS

5.1 GENERAL

The **Medical Report Automation** project successfully addresses the challenges in manually preparing and distributing patient reports by automating the process. By leveraging UiPath's Robotic Process Automation (RPA) capabilities, the system seamlessly integrates patient data from Excel, customizes Word templates, and sends personalized reports via email. Key findings from the development and implementation of this project include:

1. **Automation Benefits:**

The automation of medical report generation eliminates the need for manual preparation and distribution, significantly reducing the time and effort involved. The system ensures error-free data insertion, prompt email delivery, and consistency in report formatting, enhancing reliability in healthcare workflows.

2. **Scalability:**

The solution is designed to process large volumes of patient data efficiently, scaling to meet the needs of hospitals and clinics of varying sizes. UiPath Orchestrator's capabilities further enhance scalability, enabling the system to handle high-frequency operations without manual intervention.

3. **Flexibility and Customization:**

The system allows for dynamic configuration of Word templates and supports various patient data formats. This adaptability ensures the solution meets diverse reporting needs in healthcare, including specialization-specific report templates.

4. Error Handling and Monitoring:

Robust error-handling mechanisms ensure any issues, such as missing data or failed email delivery, are logged and addressed promptly. The system also provides comprehensive logs, facilitating easy troubleshooting and system monitoring.

5. Integration with UiPath Orchestrator:

By deploying the automation through UiPath Orchestrator, the project benefits from scheduled execution and detailed monitoring. Orchestrator's log tracking and alert features ensure the continuous operation of the system, even in dynamic and high-pressure environments.

6. Improved Data Management:

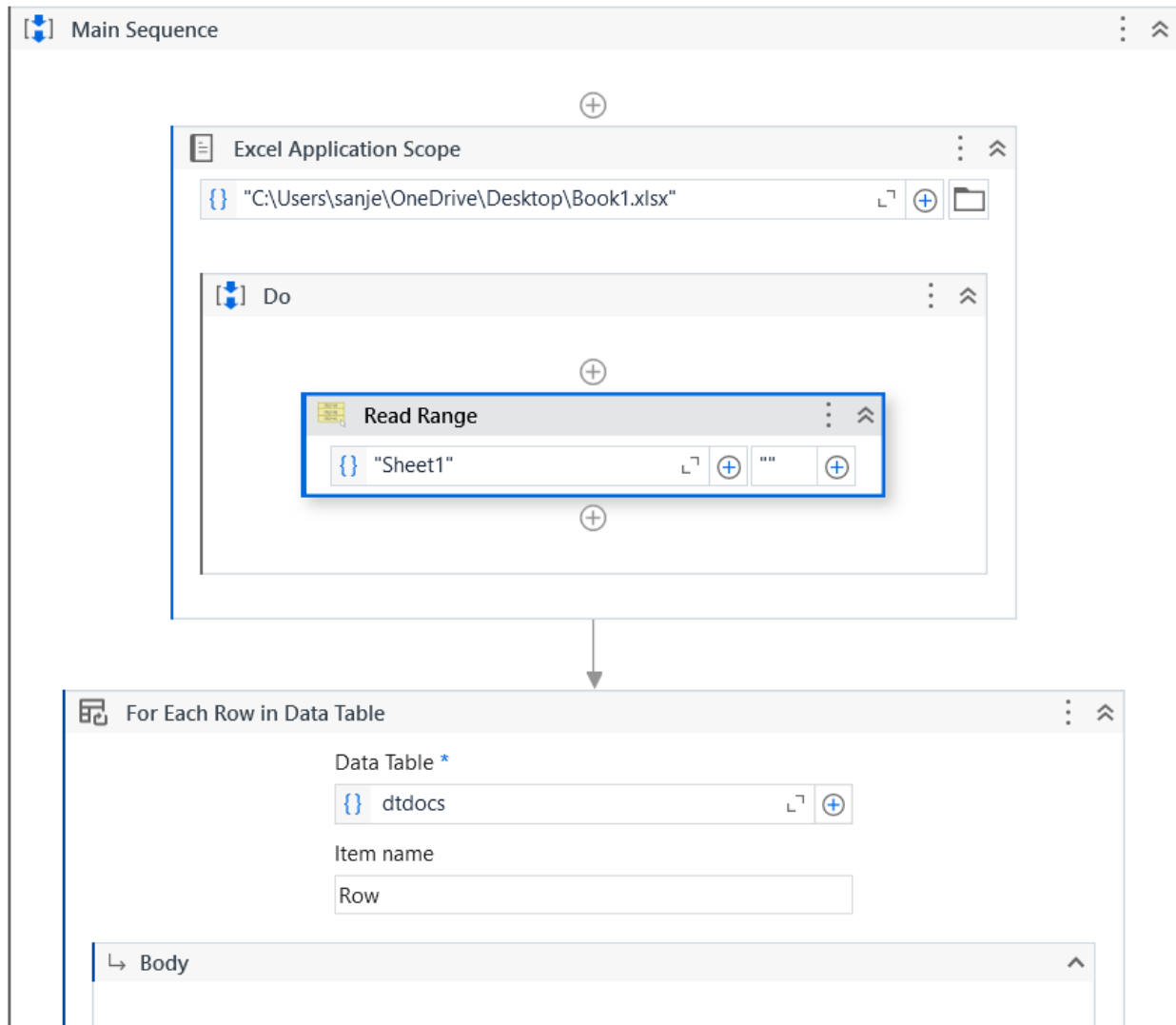
The system ensures timely delivery of medical reports, improving patient communication and enabling healthcare professionals to focus on core clinical responsibilities. Automated data logging and report tracking enhance administrative efficiency and compliance.

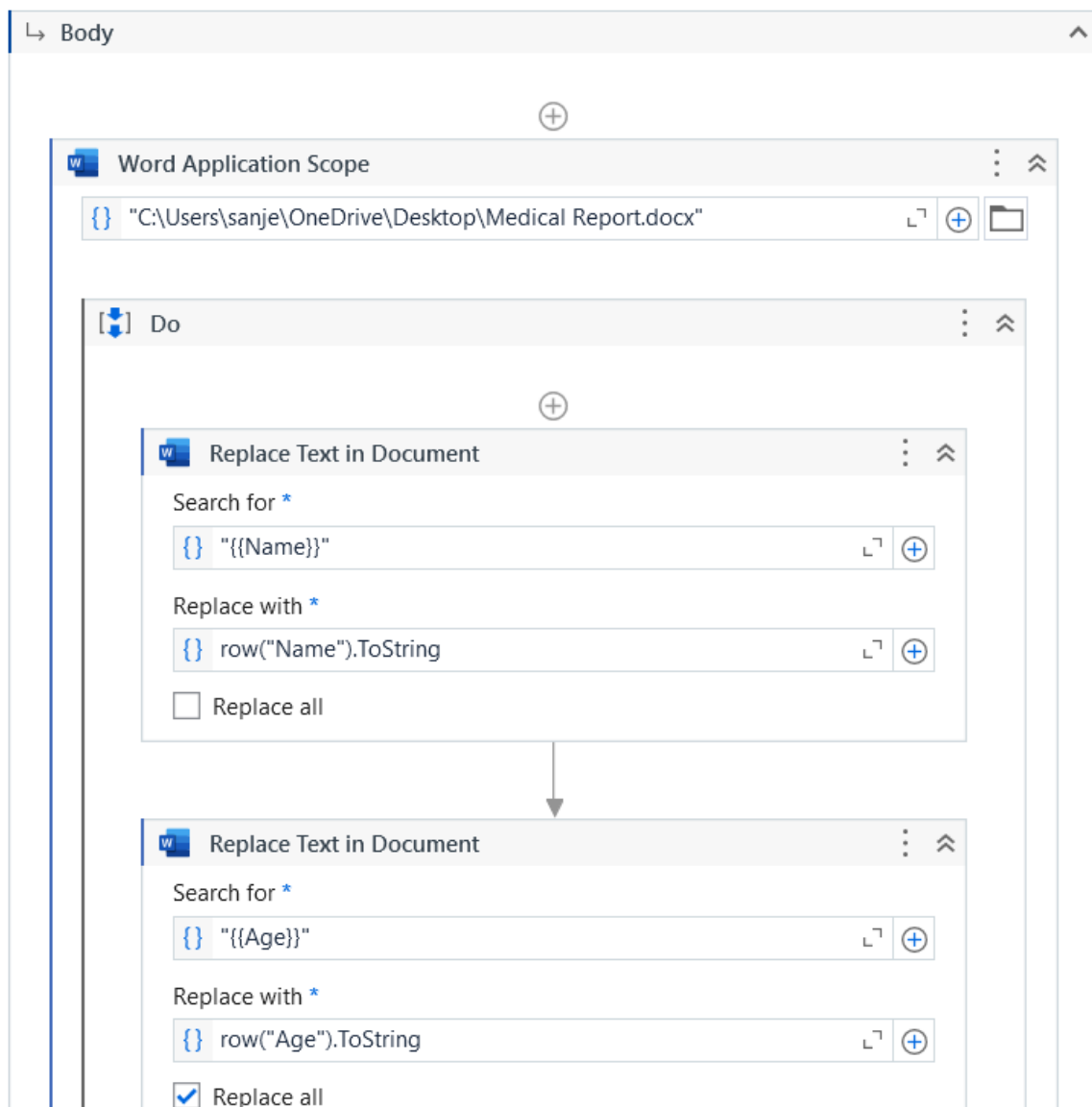
Conclusion:

The **Medical Report Automation** project demonstrates the transformative potential of RPA in healthcare documentation, providing a faster, more reliable, and scalable solution. By automating report generation and email distribution, the system reduces manual effort and improves operational efficiency. Future enhancements could include advanced analytics for report usage, integration with hospital management systems, and support for multilingual templates to cater to diverse patient populations.

APPENDIX

SAMPLE PROCESS





Replace Text in Document

Search for *

{ } "{{Treatment}}"

Replace with *

{ } row("Treatment").ToString

☒ Replace all

Replace Text in Document

Search for *

{ } "{{Date}}"

Replace with *

{ } row("Date").ToString

☒ Replace all

Save Document As

Save as type

Word Document (*.docx)

Save as file *

{ } "Report"

Save Document As

Save as type
Word Document (*.docx)

Save as file *

{ "Report"

☒ Replace existing

Send SMTP Mail Message

To * { row("Email").ToString

Subject { "medical report"

Body { "This is your medical report"

[Attach Files](#)

REFERENCES

- UiPath Official Documentation. (n.d.). UiPath Activities Guide: Automating Processes with UiPath Studio. Retrieved from <https://www.uipath.com/>
- Microsoft Office Documentation. (n.d.). Automating Word Templates with Mail Merge and Placeholders. Retrieved from <https://learn.microsoft.com/>
- Lee, J., & Kim, S. (2021). "Automation in Healthcare Documentation: A Case Study Using RPA." *Journal of Health Informatics and Automation*, 15(3), 45-52. DOI: 10.1016/j.jhia.2021.08.002
- Chen, L., & Liu, Y. (2020). "Enhancing Healthcare Administration with Robotic Process Automation." *Journal of Emerging Technologies in Healthcare*, 10(2), 75-85.
- Parker, D., & Evans, M. (2021). "RPA in Healthcare: Improving Efficiency in Medical Reporting." *Automation & Healthcare Review*, 12(4), 34-42. Retrieved from <https://www.healthcarerpa.com>