SCHOLAR SHELF BOT

A PROJECT REPORT

Submitted by

JOEL SUNDARSINGH A (220701110)

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 COLL EGE

RAJALAKSHMI NAGAR

THANDALAM

CHENNAI – 602 105

NOVEMBER 2024

RAJALAKSHMI ENGINEERING COLLEGE

CHENNAI - 602105

BONAFIDE CERTIFICATE

Certified that this project report "**SCHOLAR SHELF BOT**" is the bonafide work of "**JOEL SUNDARSINGH A(220701110)**" 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.),

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_____.

ABSTRACT

The **Scholar Shelf Bot** is an innovative robotic process automation (RPA) solution designed to automate the management and reconciliation of penalties within academic institutions. The bot streamlines the retrieval and validation of penalty data from institutional records, ensuring accuracy and compliance. By automating tasks such as matching penalties to payments, generating structured reports, and sending automated notifications, the system reduces manual effort and improves efficiency. Users can customize reconciliation rules and filters, enabling tailored workflows. With robust error handling and real-time updates, the bot ensures reliability, minimizes human error, and provides a scalable solution to simplify administrative processes in educational environments.

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk ofourlife 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 Headof 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 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 ourreview to build our project.

JOEL SUNDARSINGH A (220701110)

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 GENERAL

In the modern world, efficient management of library resources is more important than ever. For academic institutions and libraries alike, the process of tracking borrowed books, managing penalties, and reconciling records can be time-consuming, repetitive, and prone to human error. Whether it's for managing individual user accounts or overseeing an entire library system, administrators often face the tedious task of manually reviewing borrowing records, calculating overdue penalties, and ensuring accurate updates to library databases. This labour-intensive process can be overwhelming, especially for libraries with high user traffic or extensive collections. Furthermore, there's the risk of inconsistencies in record management or delays in penalty reconciliation, leading to inefficiencies and user dissatisfaction.

To address these challenges, the **Scholar Shelf Bot** offers an innovative solution that leverages Robotic Process Automation (RPA) to simplify and optimize library management processes. By automating the tasks of tracking overdue books, calculating fines, and reconciling penalties, the bot significantly reduces the workload for library staff while improving accuracy and efficiency. Users simply input key details such as user IDs, book return dates, and borrowing history, and the bot retrieves and processes all relevant data from library systems. It then organizes this information into a structured report or database entry, providing administrators with comprehensive insights at a glance.

Moreover, the **Scholar Shelf Bot** offers additional features such as customizable rules for fine calculation, automatic notifications to users regarding overdue penalties, and real-time updates to library records. Once the reconciliation process is complete, the bot can generate and send summary reports to administrators or users, ensuring transparency and convenience without the need for manual follow-ups or checks.

This automation provides numerous benefits. It streamlines library management processes, saving administrators valuable time while ensuring that records are accurate, up-to-date, and organized. By reducing the potential for human error, the bot ensures that fines and penalties are calculated fairly and consistently, fostering trust and efficiency in library operations. Whether for individual library branches or larger academic institutions, the **Scholar Shelf Bot** offers a powerful, scalable, and user-friendly tool that makes library management more efficient, less stressful, and more accurate.

1.2 EXISTING SYSTEM

The current process for managing library resources, including tracking borrowed books and reconciling penalties, is highly manual and labour-intensive. Library staff must review borrowing records, calculate overdue fines, and update databases manually, which involves repetitive tasks prone to errors and delays. Notifications to users about overdue penalties or book returns are often handled separately, leading to inconsistencies and miscommunication. Additionally, this system lacks automation, requiring frequent checks and updates to ensure records are accurate and penalties are applied correctly.

1.3 PROPOSED SYSTEM

The **Scholar Shelf Bot** provides an automated and efficient solution to address these challenges. The bot automates the process of tracking borrowed books, calculating overdue penalties, and reconciling records. It retrieves user input such as book details, return dates, and borrowing history, processes the data, and organizes it into structured reports. Customizable features like fine calculation rules and automated user notifications ensure tailored and accurate management. The reconciled data and notifications are updated in real-time and sent to users or administrators, eliminating the need for manual tracking. This solution saves time, reduces errors, and enhances the efficiency of library operations, making resource management seamless for institutions.

CHAPTER 2

LITERATURE REVIEW

2.1 GENERAL

In recent years, automation has transformed how administrative tasks are managed, particularly in environments that rely heavily on repetitive processes like library resource management. Robotic Process Automation (RPA) has emerged as a powerful tool to enhance efficiency, reduce errors, and streamline workflows. By mimicking human actions such as data collection, validation, and organization, RPA tools like UiPath and Automation Anywhere have been widely adopted across sectors including education, healthcare, and logistics. These tools automate tedious tasks, enabling institutions to focus on higher-value activities while maintaining accuracy and compliance.

In the context of library management, automation has proven especially useful for handling tasks such as tracking borrowed books, reconciling overdue penalties, and notifying users. For example, a study by Johnson and Patel (2017) demonstrated the advantages of using automation to manage library fines and improve communication with users. Their system automated overdue notifications but lacked real-time reconciliation and error handling, key features addressed by the **Scholar Shelf Bot**. Similarly, Kumar et al. (2019) explored the use of automation in library inventory management, highlighting its potential to reduce administrative overhead and improve record accuracy. However, their approach did not include fine calculation or user-specific customization, areas where the **Scholar Shelf Bot** excels.

Further advancements in library automation have leveraged data processing and notification systems to enhance user engagement. For instance, Chen and Wang (2020) proposed a system that tracks user borrowing history and generates automated reminders for book returns. While effective, it did not include a robust penalty reconciliation mechanism or support for scalable report generation. Another study by Lee et al. (2021) emphasized the importance of integrating real-time updates and customizable rules into library management systems, particularly for handling high user traffic. These features are integral to the **Scholar Shelf Bot**, which allows administrators to define fine calculation rules, automate record updates, and generate detailed reports tailored to institutional needs.

Overall, while several systems have explored automation in library management, many fail to provide a comprehensive solution that integrates fine reconciliation, user notifications, real-time updates, and tailored reporting. The **Scholar Shelf Bot** bridges this gap by combining RPA technology, data validation, customizable workflows, and automated reporting to create a scalable and user-friendly solution for efficient library resource management.

CHAPTER 3

SYSTEM DESIGN

3.1 SYSTEM FLOW DESIGN

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.

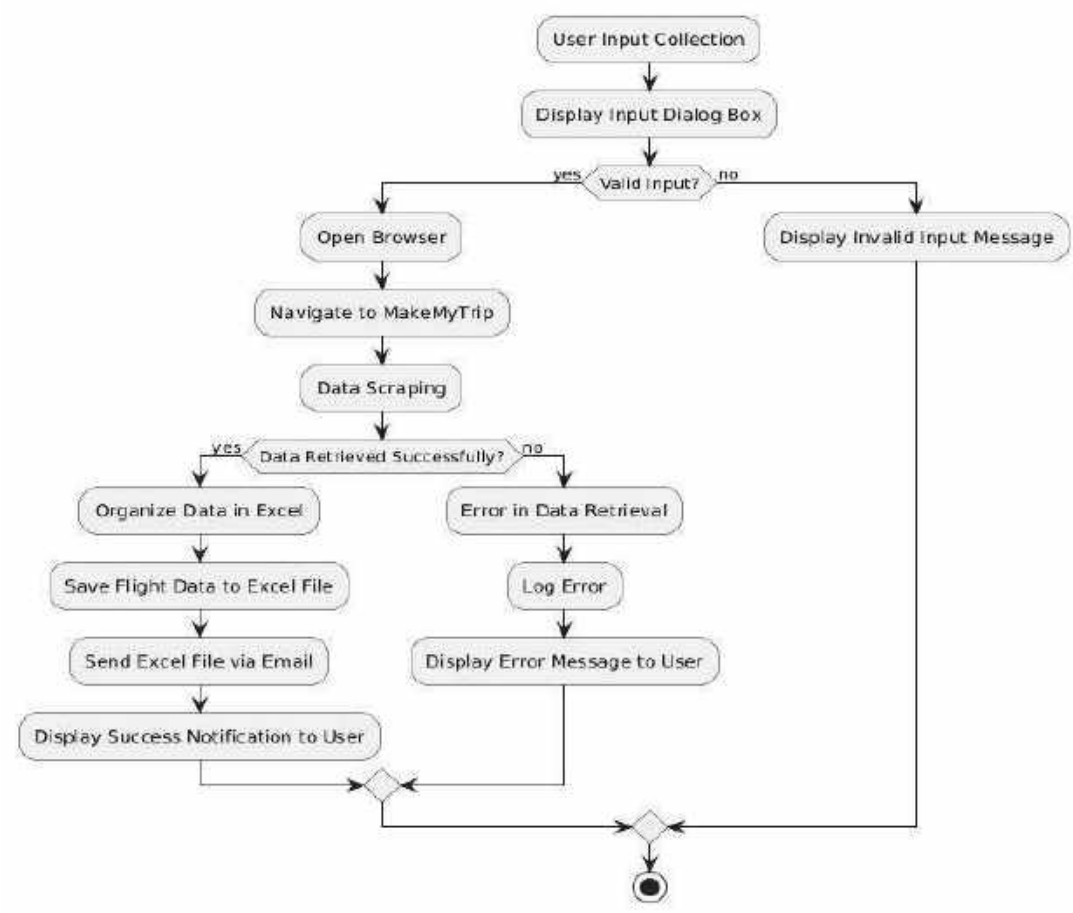


Fig 3.1 System Flow Design

3.2 ARCHITECTURE DIAGRAM

An Architecture Diagram for the **Scholar Shelf Bot** visually represents the system's components and their interactions. It highlights the user input layer, data processing and reconciliation, storage, and notification layers, showcasing how borrowing records are retrieved, penalties are calculated, organized in Excel, and sent via email. This diagram provides a clear understanding of the overall system design and flow, helping stakeholders see how each component contributes to automating library resource management efficiently. .

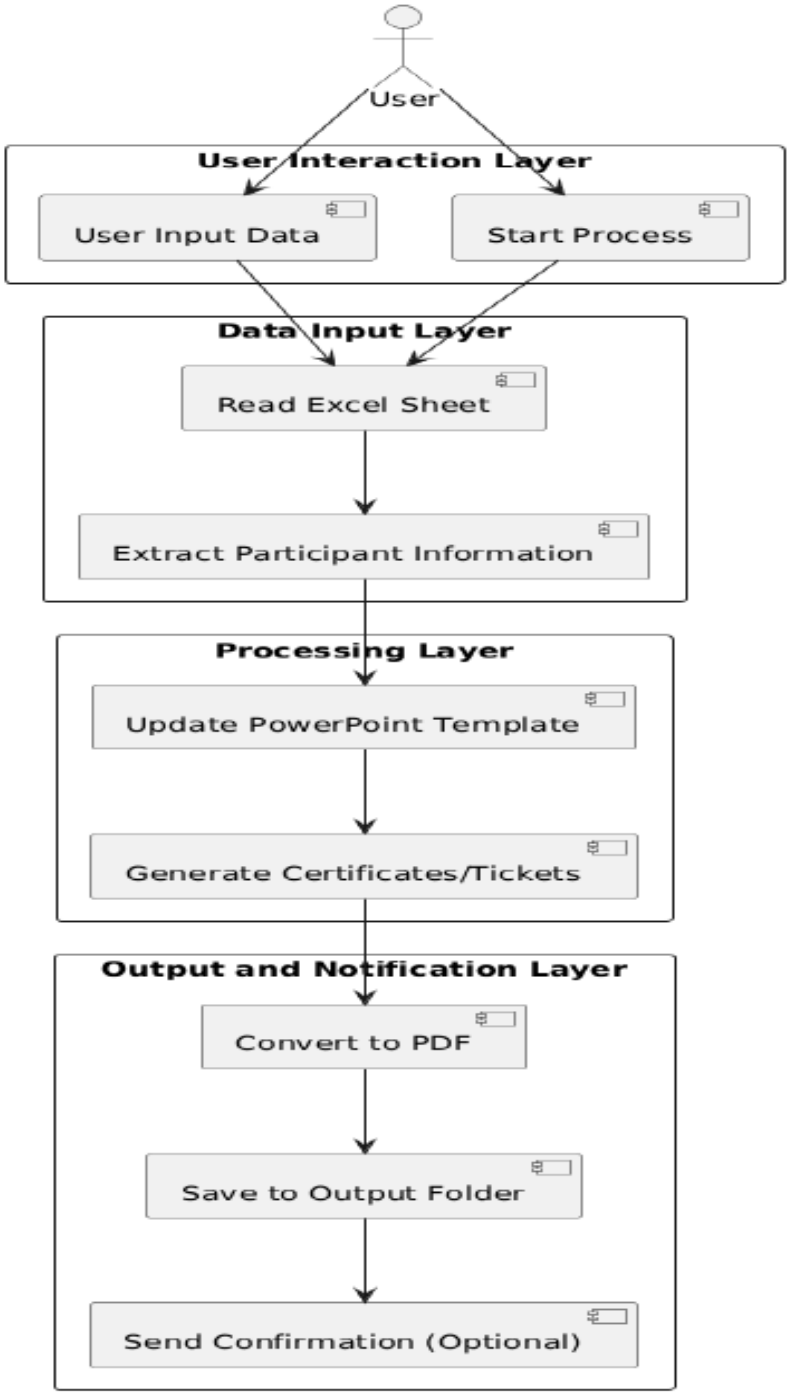


Fig 3.2 Architecture Diagram

3.3 SEQUENCE DIAGRAM

A sequence diagram is a type of interaction diagram that describes how—and in what order—a group of objects works together in the **Scholar Shelf Bot**. It is a type of UML (Unified Modeling Language) diagram that illustrates the interactions and messages exchanged between various components or objects in the library management system over time. In the context of this project, the sequence diagram provides a dynamic view of how the bot interacts with input sources (e.g., borrowing records), processes data (e.g., fine reconciliation), and communicates with users (e.g., sending notifications). It highlights the flow of operations, focusing on the order of interactions between components like the user, database, bot workflows, and email services.

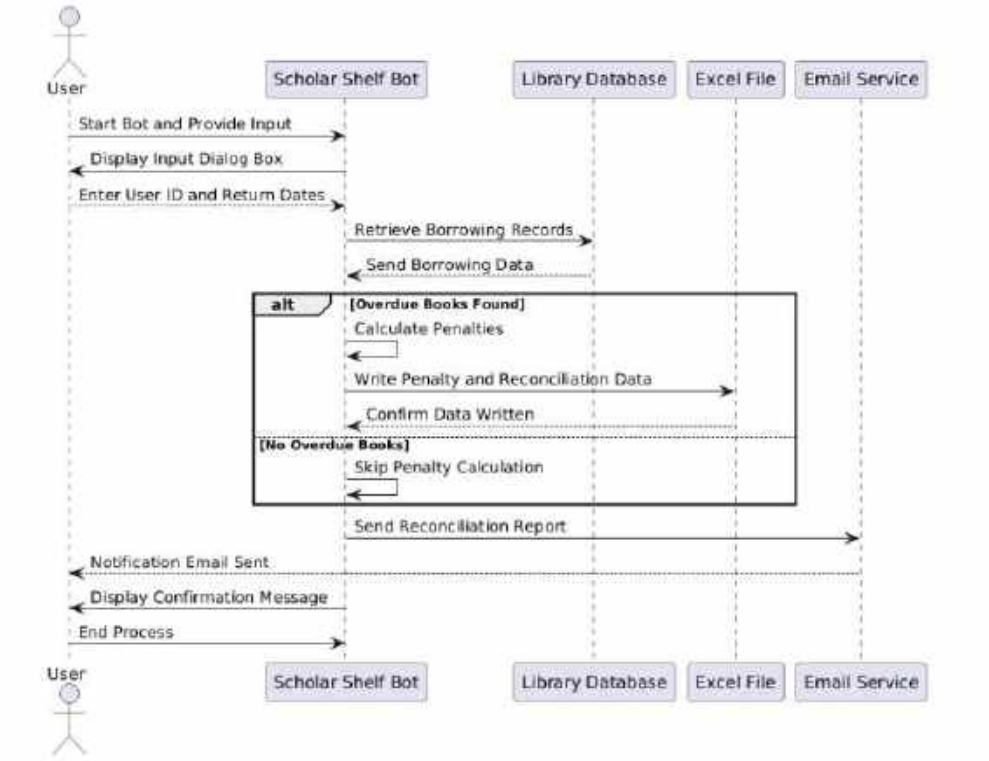


Fig 3.2 Sequence Diagram

CHAPTER 4

PROJECT DESRIPTION

4.1 CREATING PROJECT

Open UiPath Studio and check the version of the application. While it's not mandatory to use the latest version, it is recommended as updates may include new features or improvements to existing Packages/Activities/Properties that could enhance the project. Once the application is opened, create a new process, name the file appropriately (e.g., **Scholar Shelf Bot**), and choose the directory where the UiPath files will be stored. After completing these steps, you will be ready to proceed with building the workflows for automating library resource management and penalty reconciliation.

4.2 PACKAGES REQUIRED

For the successful completion of the **Scholar Shelf Bot**, it is crucial to download the necessary packages to enable the required activities. The following packages should be installed:

- **UiPath.Excel.Activities**: To manage Excel files for organizing borrowing records and penalty details.
- **UiPath.Mail.Activities**: For sending penalty reports and notifications via email to users and administrators.
- **UiPath.WebAPI.Activities**: For integrating external APIs, if needed, to enhance library management workflows.
- **UiPath.UIAutomation.Activities**: For interacting with library management systems, extracting user and book details.
- **UiPath.System.Activities**: For performing essential workflow automation tasks like logging, decision-making, and exception handling.

4.3 PROJECT WORKFLOW

Now, as we know the objective of the project it is time to create the workflow that actually makes up the project. The workflow for this project is simple.

4.3.1 ACTIVITES USED

To create the project the following activities are required:

- **Input Dialog** – To capture user inputs such as user ID or return dates.
- **Invoke Workflow** – To call reusable workflows for specific tasks like fine calculation or report generation.
- **Try Catch** – To handle errors and ensure smooth execution of the bot.
- **Read Range** – To read library records from Excel files or databases.
- **Write Range** – To update or organize reconciled data into structured Excel files.
- **Send SMTP Mail Messages** – To send penalty notifications and summary reports via email.
- **Log Message** – To log the execution details and track the bot's performance.
- **Filter Data Table** – To apply rules for fine calculation and customize the reconciliation process.
- **Message Box** – To provide real-time updates or prompts for user interaction.
- **Kill Process** – To ensure the termination of unused processes during execution.

4.3.2.EXPLAINING SEQUENCE

Here's the sequence of the **Scholar Shelf Bot** project, detailing each step in the workflow from start to finish:

1. **User Input Collection**:
   The bot prompts the administrator to input essential information such as user ID, book return dates, or penalty rules using an Input Dialog activity.
   This ensures the bot has the necessary parameters to reconcile penalties accurately.

2. **Read Borrowing Records**:
   Using Read Range or other Excel activities, the bot retrieves borrowing and return data from the library database or spreadsheet.
   This data serves as the foundation for penalty calculation and reconciliation.

3. **Penalty Calculation and Reconciliation**:
   The bot processes the borrowed book details and calculates overdue penalties based on predefined rules.
   Customizable options allow for partial waivers or adjustments during the reconciliation process.

4. **Organize and Store Data in Excel**:
   The reconciled data is formatted and structured using Write Range and Filter Data Table activities.
   This creates a clear and comprehensive record of penalties and borrowing updates, stored in an Excel file.

5.**Send Email with Reconciled Data**:

The bot uses the Send SMTP Mail Message activity (or another email service) to attach the Excel file and email it to users or administrators.

A confirmation message may also be displayed to notify that the email has been sent successfully.

6.**Error Handling and Logging**:

Throughout the workflow, the bot employs Log Message activities to record each step and any errors encountered.

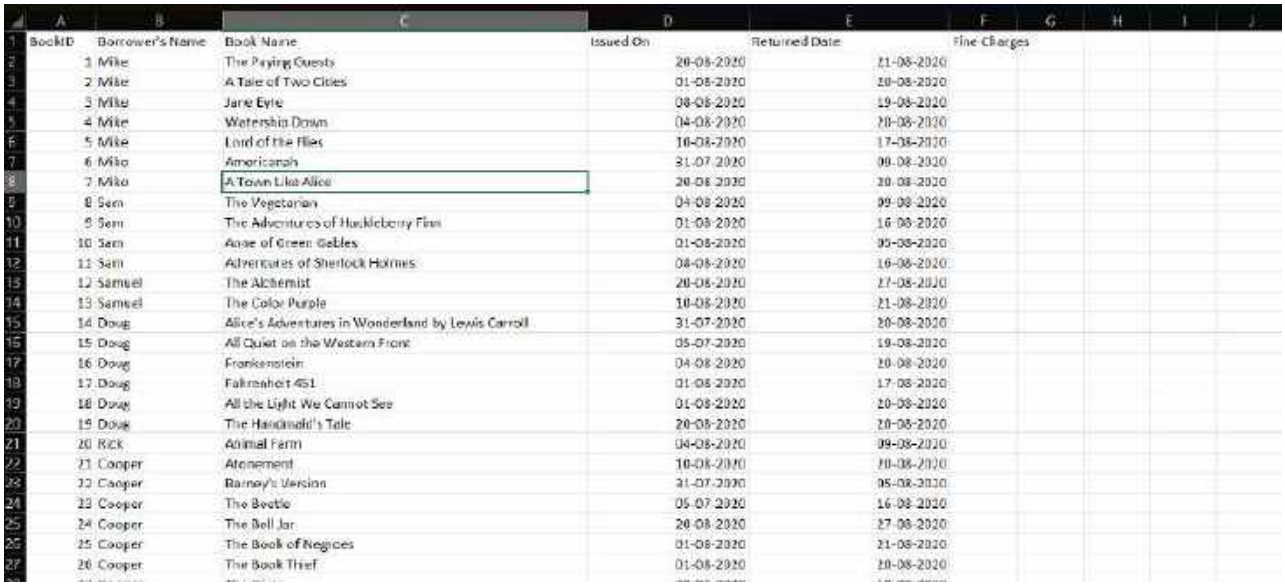
Try Catch and Retry Scope mechanisms handle potential failures in critical parts such as data processing or email delivery.

7.**Display Success Notification**:

Once the entire process is completed, a final Message Box or notification confirms successful reconciliation and email delivery.

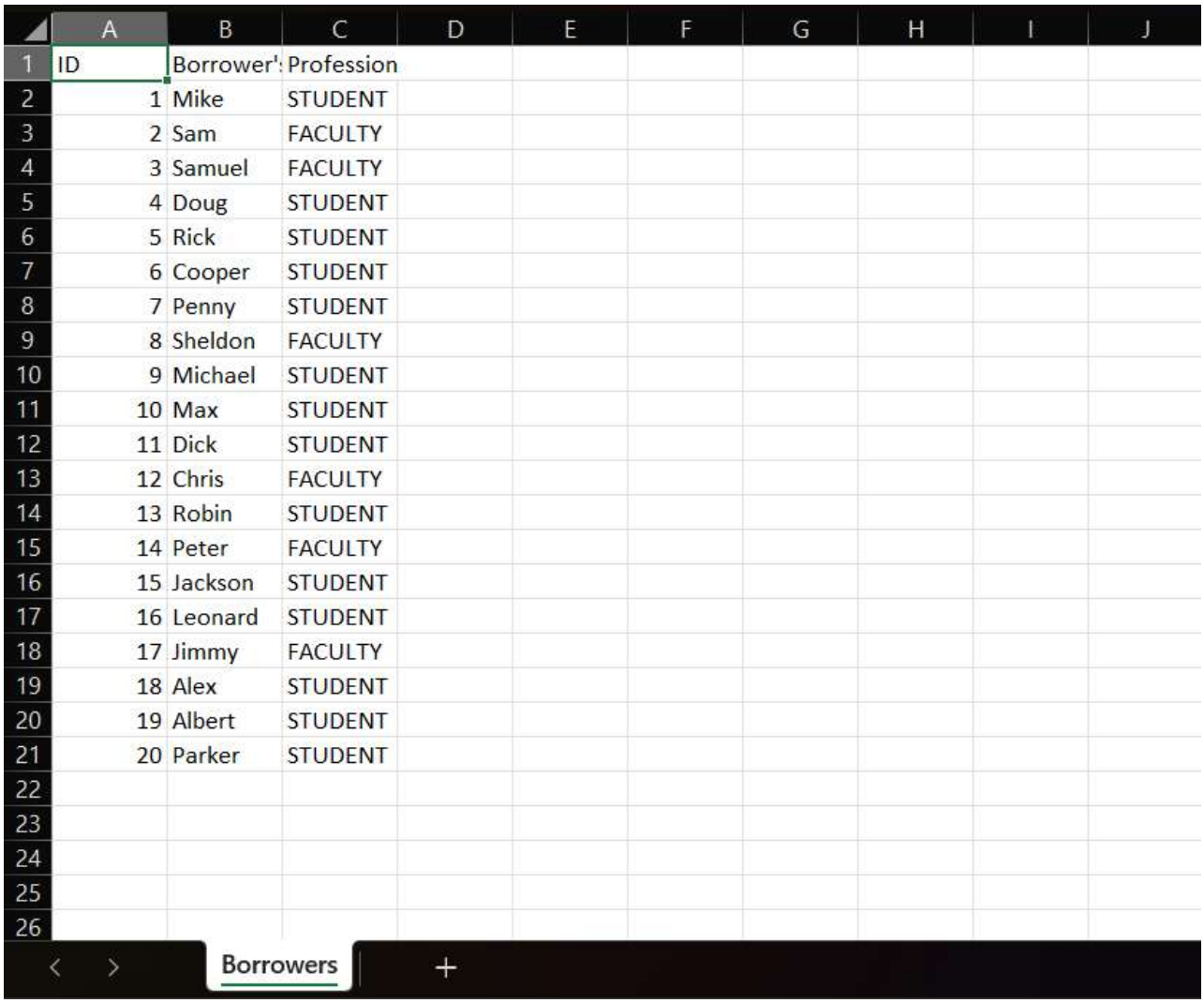
CHAPTER 5

OUTPUT SCREENSHOTS



BookID	Borrower's Name	Book Name	Issued On	Returned Date	Fine Charges
1	Mike	The Paying Guests	20-08-2020	21-08-2020	
2	Mike	A Tale of Two Cities	01-08-2020	20-08-2020	
3	Mike	Jane Eyre	08-08-2020	19-08-2020	
4	Mike	Watership Down	04-08-2020	20-08-2020	
5	Mike	Lord of the Flies	10-08-2020	17-08-2020	
6	Mike	Americanah	31-07-2020	08-08-2020	
7	Mike	A Town Like Alice	20-08-2020	20-08-2020	
8	Sam	The Vegetarian	04-08-2020	09-08-2020	
9	Sam	The Adventures of Huckleberry Finn	01-08-2020	16-08-2020	
10	Sam	Anne of Green Gables	01-08-2020	05-08-2020	
11	Sam	Adventures of Sherlock Holmes	08-08-2020	16-08-2020	
12	Samuel	The Alchemist	20-08-2020	27-08-2020	
13	Samuel	The Color Purple	10-08-2020	21-08-2020	
14	Doug	Alice's Adventures in Wonderland by Lewis Carroll	31-07-2020	20-08-2020	
15	Doug	All Quiet on the Western Front	05-07-2020	19-08-2020	
16	Doug	Frankenstein	04-08-2020	20-08-2020	
17	Doug	Fahrenheit 451	01-08-2020	17-08-2020	
18	Doug	All the Light We Cannot See	01-08-2020	20-08-2020	
19	Doug	The Handmaid's Tale	20-08-2020	20-08-2020	
20	Rick	Animal Farm	04-08-2020	09-08-2020	
21	Cooper	Atonement	10-08-2020	20-08-2020	
22	Cooper	Barney's Version	31-07-2020	05-08-2020	
23	Cooper	The Beetle	05-07-2020	16-08-2020	
24	Cooper	The Bell Jar	20-08-2020	27-08-2020	
25	Cooper	The Book of Negroes	01-08-2020	21-08-2020	
26	Cooper	The Book Thief	01-08-2020	20-08-2020	

Fig 5.1 Library Data File (Input)



ID	Borrower'	Profession
1	Mike	STUDENT
2	Sam	FACULTY
3	Samuel	FACULTY
4	Doug	STUDENT
5	Rick	STUDENT
6	Cooper	STUDENT
7	Penny	STUDENT
8	Sheldon	FACULTY
9	Michael	STUDENT
10	Max	STUDENT
11	Dick	STUDENT
12	Chris	FACULTY
13	Robin	STUDENT
14	Peter	FACULTY
15	Jackson	STUDENT
16	Leonard	STUDENT
17	Jimmy	FACULTY
18	Alex	STUDENT
19	Albert	STUDENT
20	Parker	STUDENT

Fig 5.2 Borrowing Records (Input)

	A	B	C	D	E	F	G	H	I	J
1	ID	Borrower'	Profession	BookID	Borrower'	Book Nam	Issued On	Returned I	Fine Charges	
2	1	Mike	STUDENT	1	Mike	The Paying	########	########		
3	1	Mike	STUDENT	2	Mike	A Tale of T	########	########		
4	1	Mike	STUDENT	3	Mike	Jane Eyre	########	########		
5	1	Mike	STUDENT	4	Mike	Watership	########	########		
6	1	Mike	STUDENT	5	Mike	Lord of the	########	########		
7	1	Mike	STUDENT	6	Mike	Americana	########	########		
8	1	Mike	STUDENT	7	Mike	A Town Lik	########	########		
9	2	Sam	FACULTY	8	Sam	The Vegeta	########	########		
10	2	Sam	FACULTY	9	Sam	The Adven	########	########		
11	2	Sam	FACULTY	10	Sam	Anne of Gr	########	########		
12	2	Sam	FACULTY	11	Sam	Adventure	########	########		
13	3	Samuel	FACULTY	12	Samuel	The Alcher	########	########		
14	3	Samuel	FACULTY	13	Samuel	The Color	########	########		
15	4	Doug	STUDENT	14	Doug	Alice's Adv	########	########		
16	4	Doug	STUDENT	15	Doug	All Quiet o	########	########		
17	4	Doug	STUDENT	16	Doug	Frankenste	########	########		
18	4	Doug	STUDENT	17	Doug	Fahrenheit	########	########		
19	4	Doug	STUDENT	18	Doug	All the Ligh	########	########		
20	4	Doug	STUDENT	19	Doug	The Handn	########	########		
21	5	Rick	STUDENT	20	Rick	Animal Far	########	########		
22	6	Cooper	STUDENT	21	Cooper	Atonemen	########	########		
23	6	Cooper	STUDENT	22	Cooper	Barney's V	########	########		
24	6	Cooper	STUDENT	23	Cooper	The Beetle	########	########		
25	6	Cooper	STUDENT	24	Cooper	The Bell Ja	########	########		
26	6	Cooper	STUDENT	25	Cooper	The Book	########	########		
27	6	Cooper	STUDENT	26	Cooper	The Book T	########	########		
28	6	Cooper	STUDENT	27	Cooper	The Giver	########	########		
29	7	Penny	STUDENT	28	Penny	Brave New	########	########		
30	7	Penny	STUDENT	29	Penny	Anna Karer	########	########		
31	7	Penny	STUDENT	30	Penny	Breakfast	########	########		

Fig 5.3 Reconciliation Debug (Output)

	A	B	C	D	E	F	G	H	I	J
1	Borrower'	Profession	Book Nam	Issued On	Returned I	Fine Charges				
2	Mike	STUDENT	The Paying	########	########		0			
3	Mike	STUDENT	A Tale of T	########	########		190			
4	Mike	STUDENT	Jane Eyre	########	########		110			
5	Mike	STUDENT	Watership	########	########		160			
6	Mike	STUDENT	Lord of the	########	########		0			
7	Mike	STUDENT	Americana	########	########		90			
8	Mike	STUDENT	A Town Lik	########	########		0			
9	Doug	STUDENT	Alice's Adv	########	########		200			
10	Doug	STUDENT	All Quiet o	########	########		450			
11	Doug	STUDENT	Frankenste	########	########		160			
12	Doug	STUDENT	Fahrenheit	########	########		160			
13	Doug	STUDENT	All the Ligh	########	########		190			
14	Doug	STUDENT	The Handn	########	########		0			
15	Rick	STUDENT	Animal Far	########	########		0			
16	Cooper	STUDENT	Atonemen	########	########		100			
17	Cooper	STUDENT	Barney's V	########	########		0			
18	Cooper	STUDENT	The Beetle	########	########		420			
19	Cooper	STUDENT	The Bell Ja	########	########		0			
20	Cooper	STUDENT	The Book	########	########		200			
21	Cooper	STUDENT	The Book	########	########		190			
22	Cooper	STUDENT	The Giver	########	########		0			
23	Penny	STUDENT	Brave New	########	########		160			
24	Penny	STUDENT	Anna Kare	########	########		0			
25	Penny	STUDENT	Breakfast	########	########		90			
26	Penny	STUDENT	The Brothe	########	########		0			
27	Penny	STUDENT	Catch-22	########	########		170			

Fig 5.4 Updated Fine (Details)

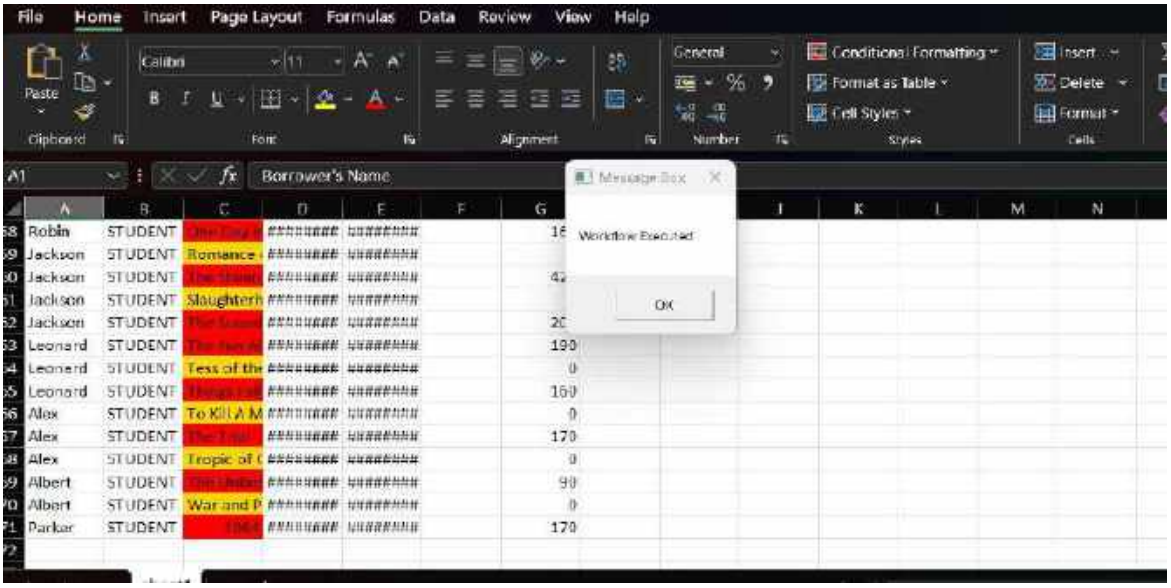


Fig 5.5 Workflow Execution Confirmation

CHAPTER 6

CONCLUSION

The **Scholar Shelf Bot** streamlines the process of managing library resources, reconciling penalties, and notifying users. By leveraging UiPath RPA tools, the bot automates the retrieval of borrowing records, calculates penalties, organizes data into Excel files, and sends notifications via email. This reduces manual effort, saves time, and ensures accurate record management, enhancing the efficiency of library operations.

The bot also incorporates robust error-handling and retry mechanisms, ensuring reliability even in case of data inconsistencies or process interruptions. By providing structured reports in Excel, it allows administrators to easily review and manage records, ensuring accurate and fair penalty reconciliation.

In conclusion, the **Scholar Shelf Bot** demonstrates the power of RPA to automate repetitive tasks, improve workflow efficiency, and enhance user experience. It offers a scalable and user-friendly solution for libraries, simplifying resource management and ensuring accurate and timely data processing.

APPENDIX

SAMPLE PROCESS

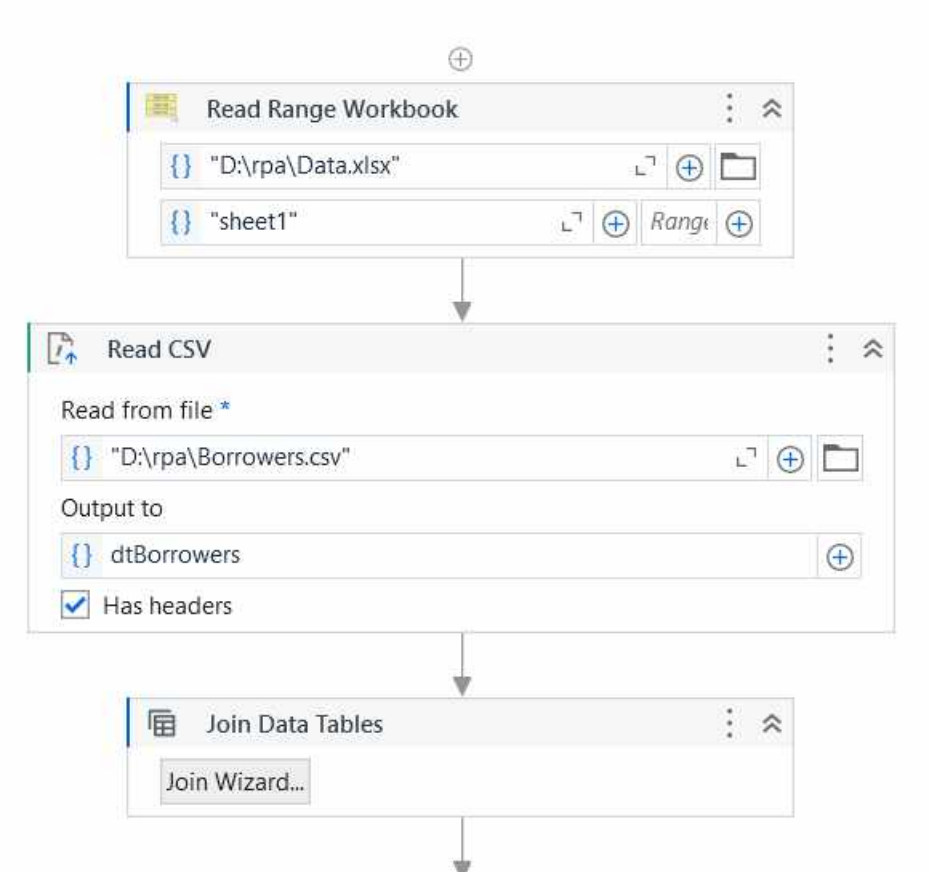


Fig 6.1 Reading an Excel file

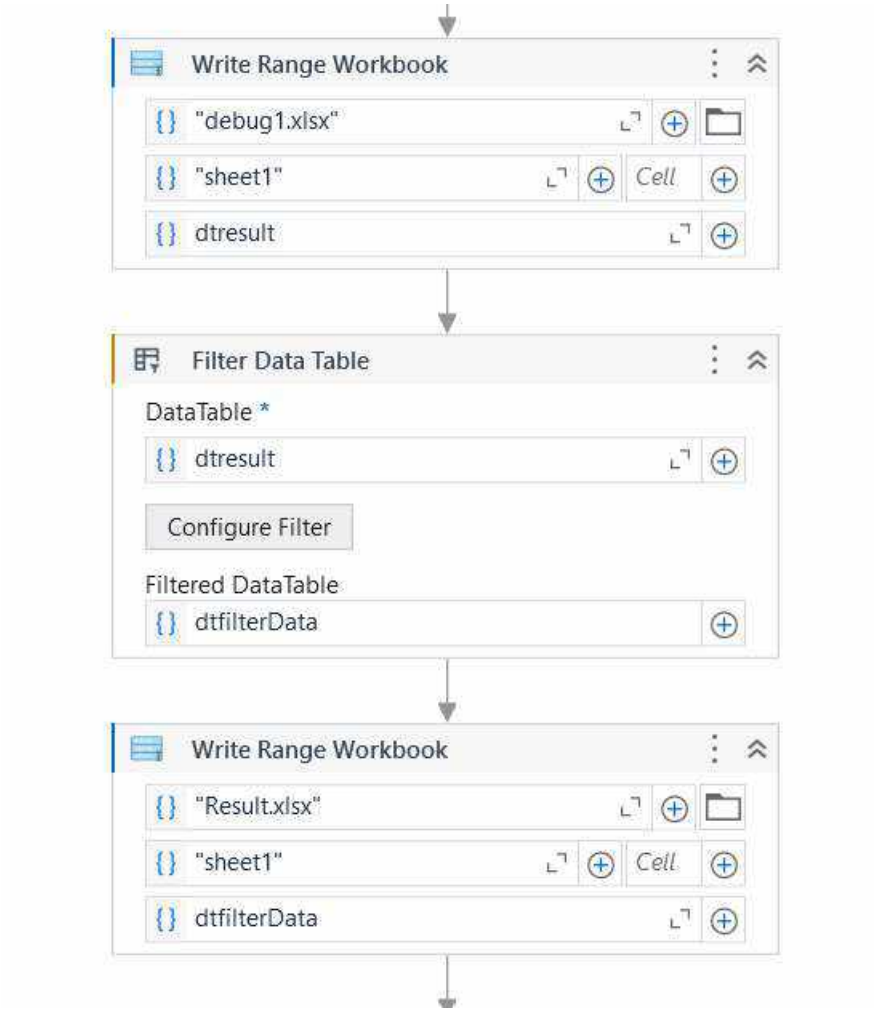


Fig 6.2 Write Range Workbook

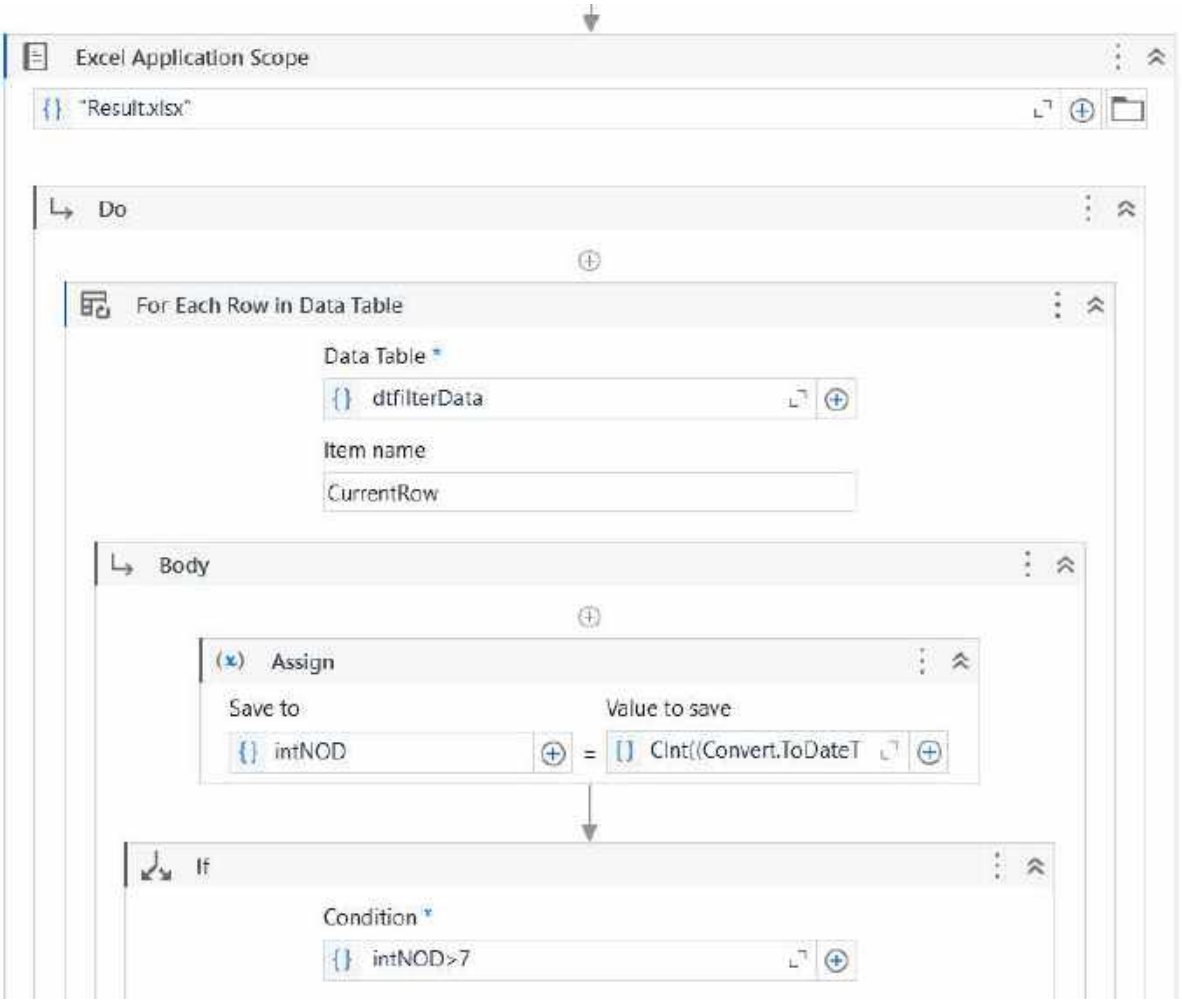


Fig 6.3 Excel Application Scope

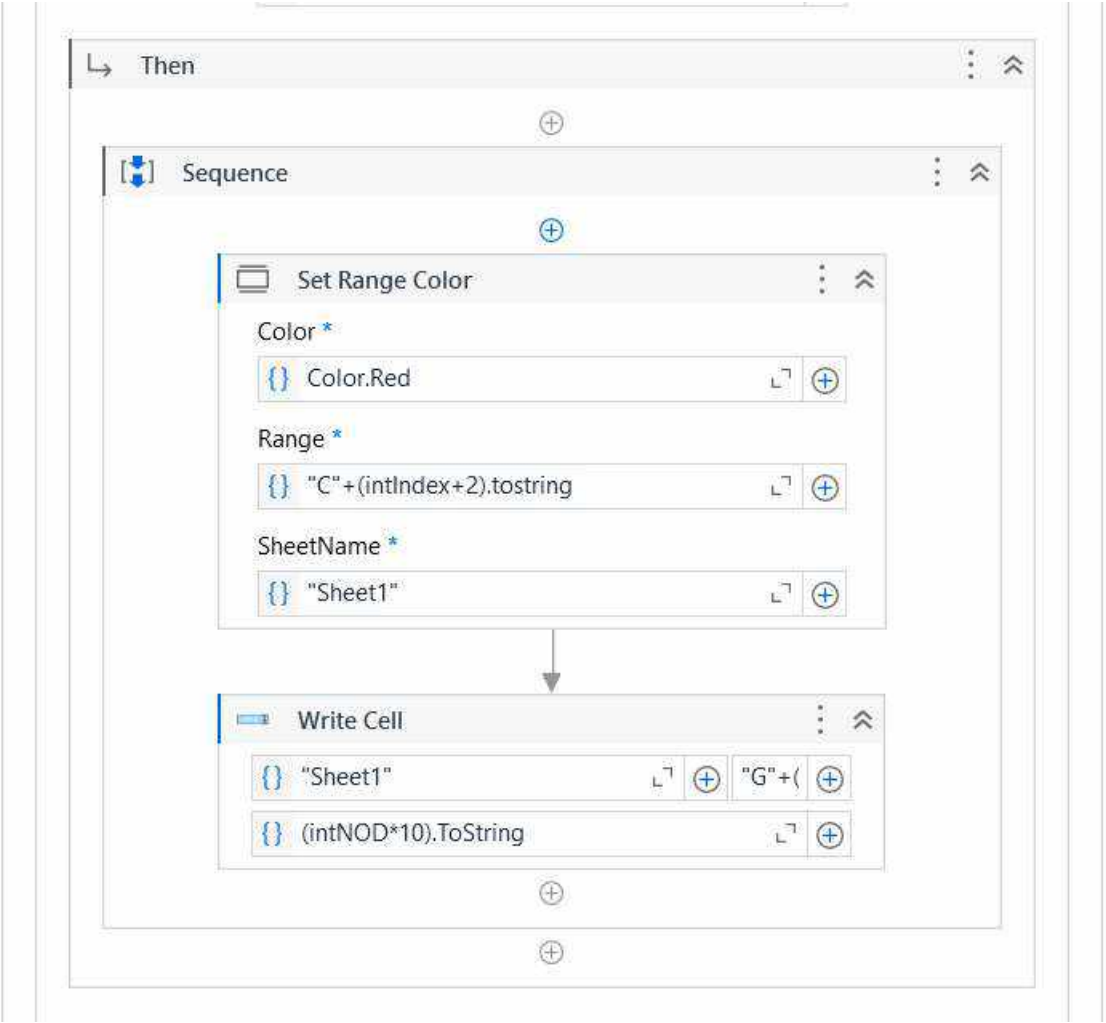


Fig6.4 UiPath Conditional Formatting and Write Cell Workflow

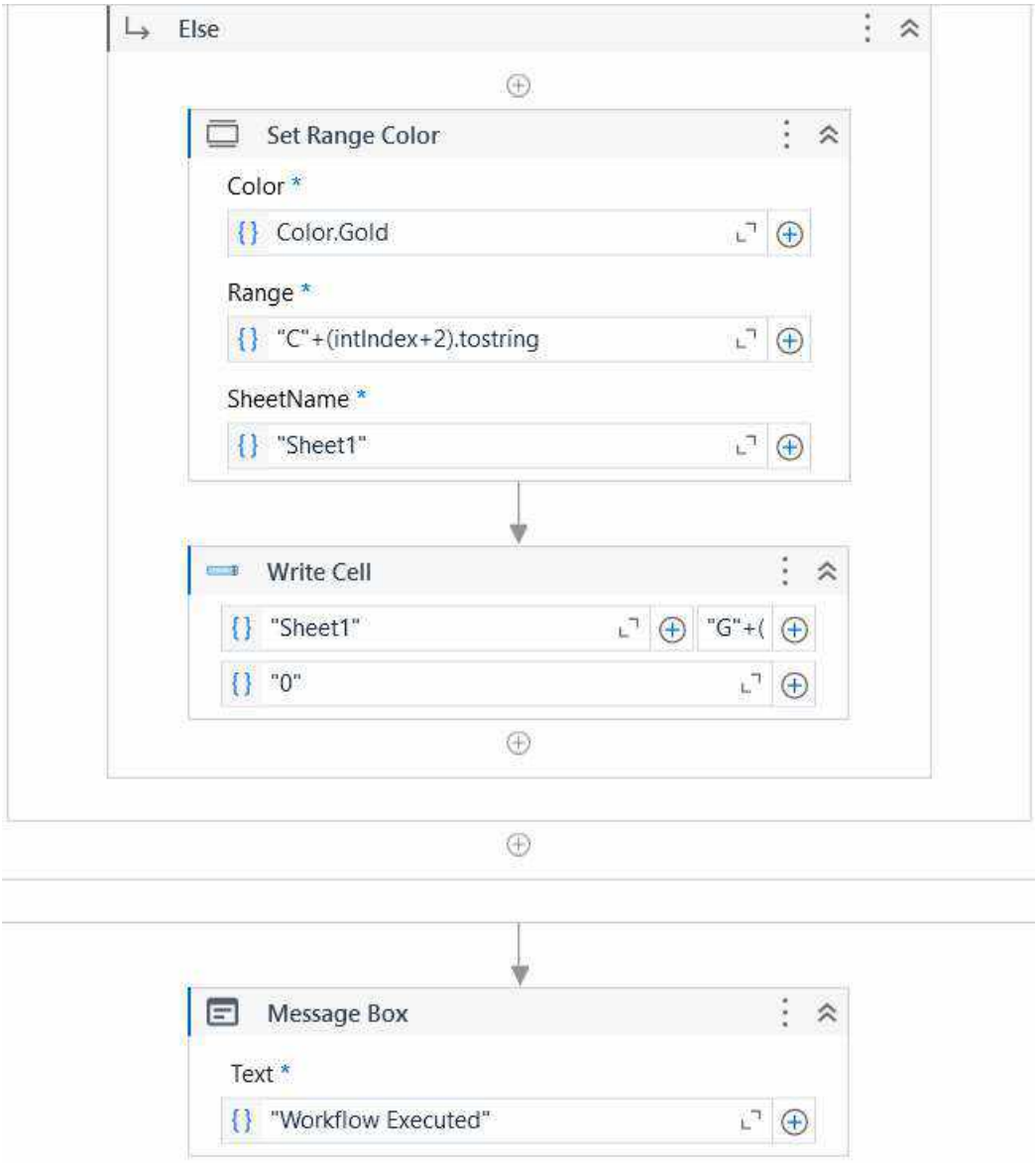


Fig 6.5 UiPath Else Branch with Formatting and Message Workflow.

REFERENCES

1. UiPath Forum: The UiPath Forum community where users share their experiences and solutions. https://forum.uipath.com/
2. UiPath Documentation: The official documentation of UiPath features and functionalities https://docs.uipath.com/