

COMMUNITY HALL RESERVATION SYSTEM

A PROJECT REPORT

Submitted by

KALAISELVI S(220701116)

in partial fulfillment for the course

OAI1903 - INTRODUCTION TO ROBOTIC PROCESS AUTOMATION

for the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

RAJALAKSHMI NAGAR

THANDALAM

CHENNAI – 602 105

NOVEMBER 2024

RAJALAKSHMI ENGINEERING COLLEGE

CHENNAI - 602105

BONAFIDE CERTIFICATE

Certified that this project report “**Community Hall Reservation System**” is the bonafide work of “**KALAISELVI S(220701116)**” 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 **Community Hall Reservation System** is an RPA-based solution designed to automate the process of booking and managing community hall reservations. Built using UiPath, the system collects user details such as name, contact information, and desired time slots. It ensures secure bookings through OTP validation by generating and sending a unique code to the user's email, which must be verified before proceeding.

The system interacts with an Excel file to check the availability of time slots, updating the status to "Booked" if the slot is available and prompting the user to choose another slot if unavailable. Once a booking is confirmed, the system updates the Excel sheet and sends an automated email containing the booking details and payment reminders. Activities like Input Dialog, Read Range, Write Range, and Send SMTP Mail Message are utilized to achieve a seamless workflow.

This system is designed for community hall administrators and users, reducing manual effort and improving efficiency in the booking process. Scalable and customizable, it can be expanded to include features like payment integration and SMS notifications, ensuring a hassle-free and transparent experience for users.

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 Coordinators, **Dr. N.Durai Murugan, M.E., Ph.D.**, Associate Professor, and **Mr. B.Bhuvaneswaran, M.E.**, Assistant Professor (SG), Department of Computer Science and Engineering for their useful tips during our review to build our project.

KALAISELVI S(220701116)

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	iii
	LIST OF FIGURES	vi
1.	INTRODUCTION	1
	1.1 GENERAL	1
	1.2 EXISTING SYSTEM	3
	1.3 PROPOSED SYSTEM	3
2.	LITERATURE REVIEW	4
	2.1 GENERAL	4
3.	SYSTEM DESIGN	6
	3.1 SYSTEM FLOW DIAGRAM	6
	3.2 ARCHITECTURE DIAGRAM	7
	3.3 SEQUENCE DIAGRAM	8
4.	PROJECT DESCRIPTION	9
	4.1 CREATING PROJECT	9
	4.2 PACKAGES REQUIRED	9
	4.3 PROJECT WORKFLOW	10
	4.3.1 ACTIVITIES USED	10
	4.3.2 EXPLAINING SEQUENCE	11
5.	OUTPUT SREENSHOTS	13
6.	CONCLUSIONS	19
	APPENDICES	20
	REFERENCES	24

LIST OF FIGURES

Figure No	Figure Name	Page No
3.1	System Flow Design	6
3.2	Architecture Diagram	7
3.3	Sequence Diagram	8
4.3.2	Sequence Creation	11
5.1	Name Input Box	13
5.2	Phone no Input Box	13
5.3	Address Input Box	14
5.4	Email Id Input Box	14
5.5	Message Box	15
5.6	OTP from Email	15
5.7	OTP Validation Box	16
5.8	Time Slot Selection	16
5.9	Successful Email Delivery	17
5.10	Email Attachment	17

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Managing bookings for community halls can be a challenging task, often requiring manual coordination, repeated follow-ups, and ensuring availability for users. In today's fast-paced world, such manual processes can lead to inefficiencies, missed opportunities, and user dissatisfaction. By integrating technology into this domain, we can streamline and automate the reservation process for enhanced efficiency and convenience.

The **Community Hall Reservation System** leverages Robotic Process Automation (RPA) to provide an efficient and user-friendly platform for booking community halls. The system automates key tasks such as validating user credentials through OTP, checking slot availability, updating reservations dynamically, and sending email confirmations. This ensures a seamless booking experience while reducing manual effort for administrators.

Designed to be flexible and intuitive, the system integrates Excel for managing slot data and SMTP for real-time email communication. By automating these repetitive tasks using UiPath, the solution saves time, enhances user satisfaction, and promotes a hassle-free reservation experience. It serves as a practical example of how RPA can revolutionize traditional processes in a user-centric manner.

1.2 EXISTING SYSTEM

Existing systems for community hall reservations often rely on manual processes or basic online booking platforms. Manual systems involve extensive coordination, repeated follow-ups, and physical documentation, making the process time-consuming and prone to errors. On the other hand, basic online platforms may offer booking functionality but often lack features like real-time availability updates, automated notifications, or secure OTP-based validation. These limitations create inefficiencies for both users and administrators, highlighting the need for a more automated, reliable, and user-friendly solution to streamline the reservation process.

1.3 PROPOSED SYSTEM

The proposed system automates the community hall reservation process using Robotic Process Automation (RPA), providing a seamless and efficient solution. By integrating OTP-based validation and dynamic availability checks from an Excel file, the system ensures secure and real-time booking of slots. It automates tasks like sending personalized email notifications for booking confirmations and updating slot statuses, eliminating manual intervention and reducing errors. This system enhances the user experience while streamlining the management of reservations for administrators.

CHAPTER 2

LITERATURE REVIEW

2.1 GENERAL

The integration of automation in reservation and booking systems has gained significant traction in recent years. With increasing demand for efficient and user-friendly solutions, systems that streamline and secure the reservation process are highly sought after. Robotic Process Automation (RPA) presents a powerful solution by automating repetitive tasks and enhancing the overall efficiency of managing bookings.

Automation in reservation management has largely been implemented through online portals and mobile applications, which facilitate basic functions like booking slots and sending notifications. However, these systems often rely heavily on manual intervention, leading to inefficiencies and a higher likelihood of errors. RPA introduces a dynamic approach by automating tasks such as availability checks, user validation, and notification delivery, providing a more seamless experience for both users and administrators.

Traditional reservation systems, whether manual or semi-automated, struggle with scalability and adaptability. Manual processes for verification, booking updates, and communication often lead to delays and mismanagement. These systems face challenges in ensuring accuracy and user satisfaction, highlighting the need for an automated and secure alternative, such as RPA-driven reservation systems.

Automation in reservation systems boosts efficiency by handling repetitive tasks like availability checks, user validation, and notifications. This reduces manual effort and errors, leading to faster and more accurate bookings. By processing multiple reservations simultaneously, automated systems improve the user experience with quicker responses and fewer delays.

RPA-powered systems enhance customer satisfaction by providing real-time updates, faster confirmations, and fewer errors. Personalized reminders and timely notifications keep users engaged, leading to improved retention and loyalty. Automation ensures users receive accurate information without delays, resulting in a more reliable and efficient experience.

Automated reservation systems offer superior scalability compared to traditional systems. As booking volumes grow, automated systems can handle increased demand without the need for additional manual intervention. This scalability ensures that businesses can continue to operate efficiently even during busy periods. Furthermore, automated systems are adaptable to changing needs, such as modifications in available slots or new integrations with third-party services. This flexibility allows businesses to adjust quickly to shifts in user behavior or operational requirements, ensuring smooth and efficient reservations 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.

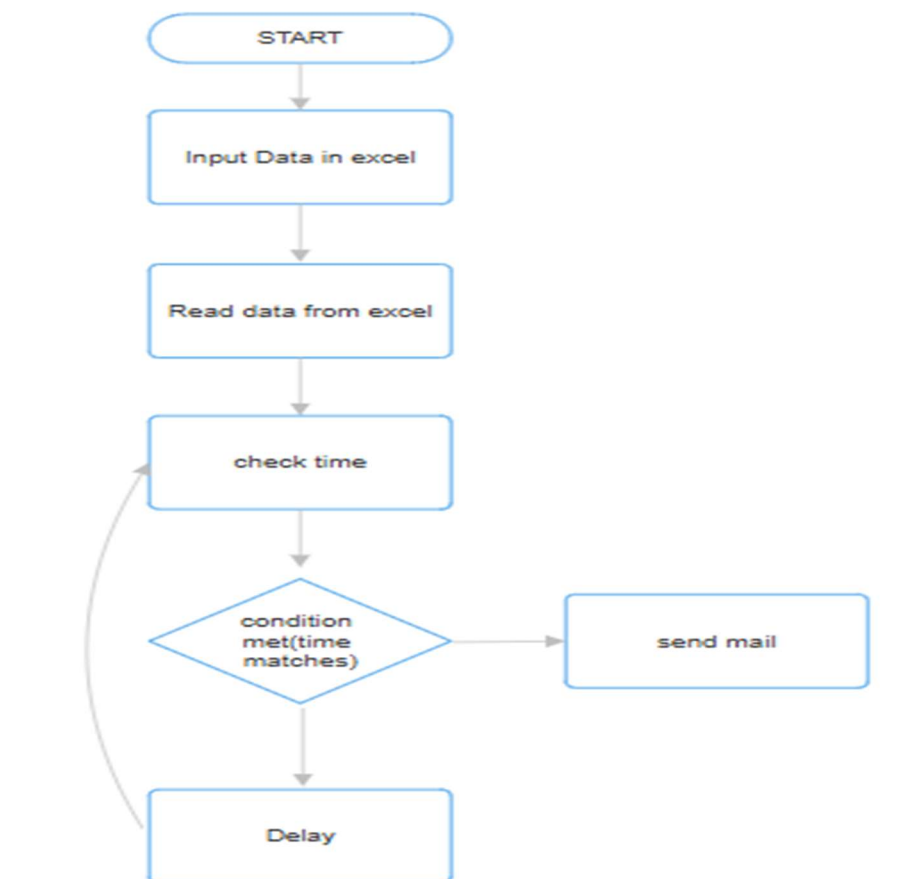


Fig 3.1 System Flow Design

3.2 ARCHITECTURE DIAGRAM

The **Architecture Diagram** for the **Community Hall Reservation System** visually represents the system's components and their interactions. It illustrates how user details are collected, time slot availability is checked, the reservation is confirmed, and notifications are sent via email. This diagram provides a clear understanding of the overall system design and flow, helping stakeholders grasp how each part contributes to the seamless reservation process.

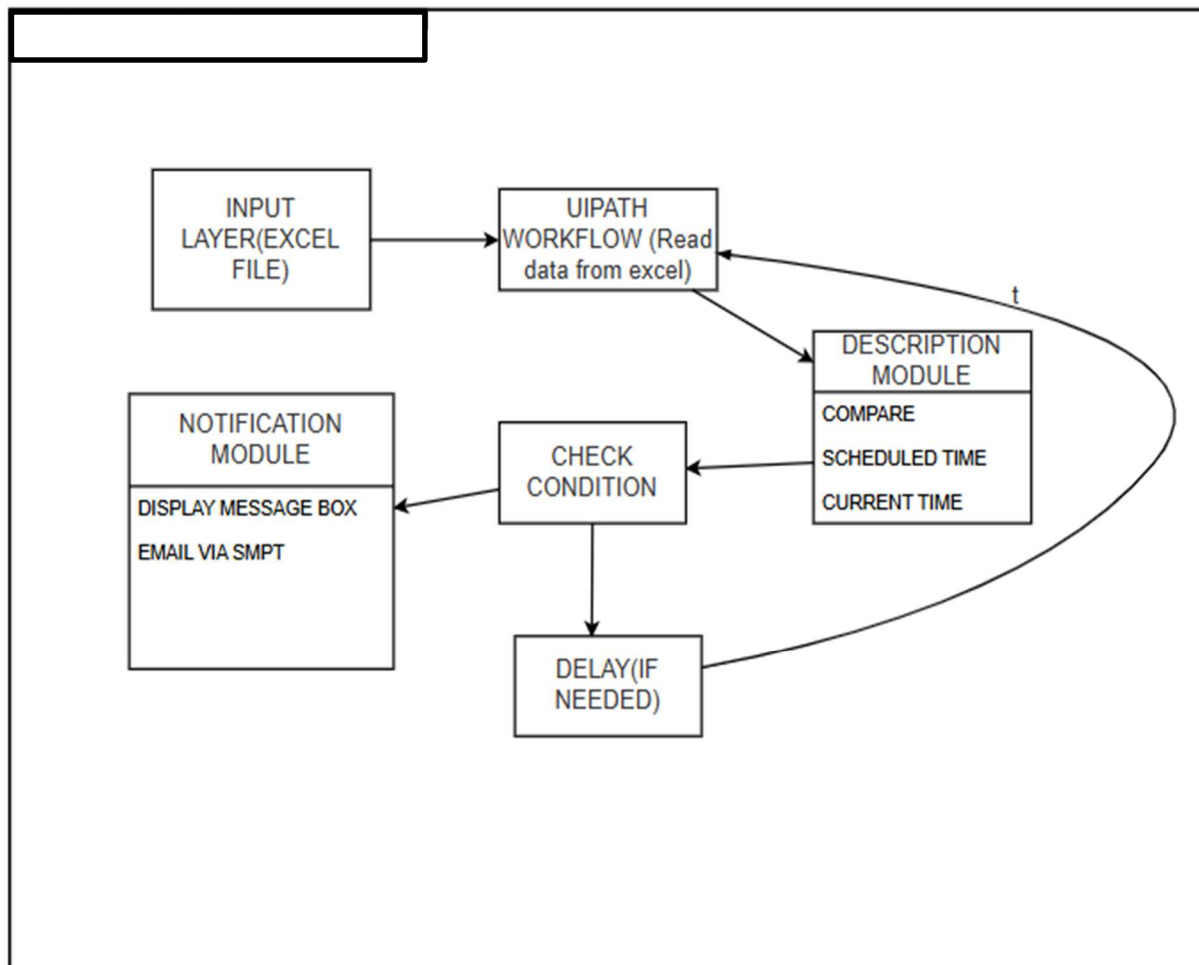


Fig 3.2 Architecture Diagram

3.2 SEQUENCE DIAGRAM

A sequence diagram is a type of interaction diagram because it describes how— and in what order—a group of objects works together. A sequence diagram is a type of UML (Unified Modeling Language) diagram that illustrates the interactions and messages exchanged between different components or objects in a system over time. It provides a dynamic view of a system, focusing on the order of interactions between objects or components.

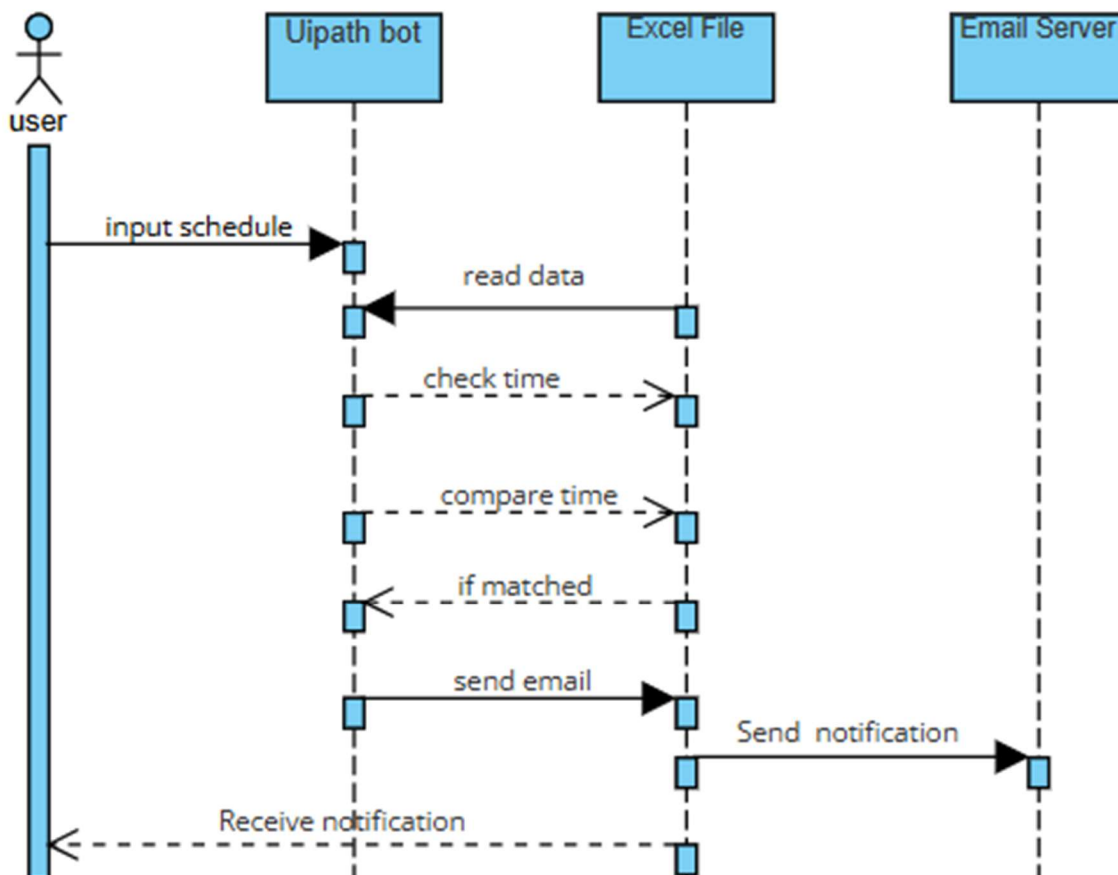


Fig 3.2 Sequence Diagram

CHAPTER 4

PROJECT DESCRIPTION

4.1 CREATING PROJECT

Open UiPath Studio and check for the version of the application. While it's not compulsory to work with the latest version, it is recommended as some features might have been added or few changes might have been made to the already existing Packages/Activities/Properties etc. Once the application is opened, create a new process, name the file and choose the directory where the UiPath files must be stored. Once you are done with the following steps, you will be good to continue with the next steps of actually Creating the Project.

4.2 PACKAGES REQUIRED

For the successful completion of the Community Hall Reservation System, it's crucial to download the necessary packages to enable the required activities. The following packages should be installed:

UiPath.Excel.Activities: To work with Excel files and organize flight data.

UiPath.Mail.Activities: For sending the Excel file via email.

UiPath.UIAutomation.Activities : For sending emails with attachments (Excel file).

UiPath.System.Activities: For basic workflow automation tasks like logging & exception.

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 ACTIVITIES USED

To create the project the following activities are required:

1. Excel Application Scope
2. Read Range
3. For Each Row in Data Table
4. Assign
5. If
6. Send SMTP Mail Message
7. Message box
8. Break
9. Do While
10. Write Range
11. Input Dialog

4.3.2 EXPLAINING SEQUENCE

Here's the sequence of the **Community Hall Reservation System** project, detailing each step in the workflow from start to finish:

The project was initiated by setting up a UiPath workflow and adding necessary dependencies such as **Excel** and **Mail** activities for handling time slot data and sending notifications.

An Excel file was created to store the time slot availability for the community hall, including columns like **Time Slot**, **Availability Status** (Available/Booked), **User Name**, and **Email**. This file acted as both the input for time slot management and the storage for reservation updates.

Use the **Read Range** activity to load the Excel file into a DataTable. This allowed the system to read all available time slots and the booking status, enabling to process each reservation request accordingly.

A **For Each Row** activity was used to iterate through each time slot entry in the DataTable. The bot processed one slot at a time to ensure that each reservation request was handled sequentially.

For each row, the bot extracted the **Time Slot**, **Availability Status**, **User Name**, and **Email Address**. It also parsed the **Time Slot** column to ensure the format was compatible with the current system time for comparison.

Use an **If** activity to check if the time slot was **Available**. If the slot was available, the bot proceeded to verify the user's reservation request..

If the user's selected time slot was available, then move on to send the reservation confirmation.

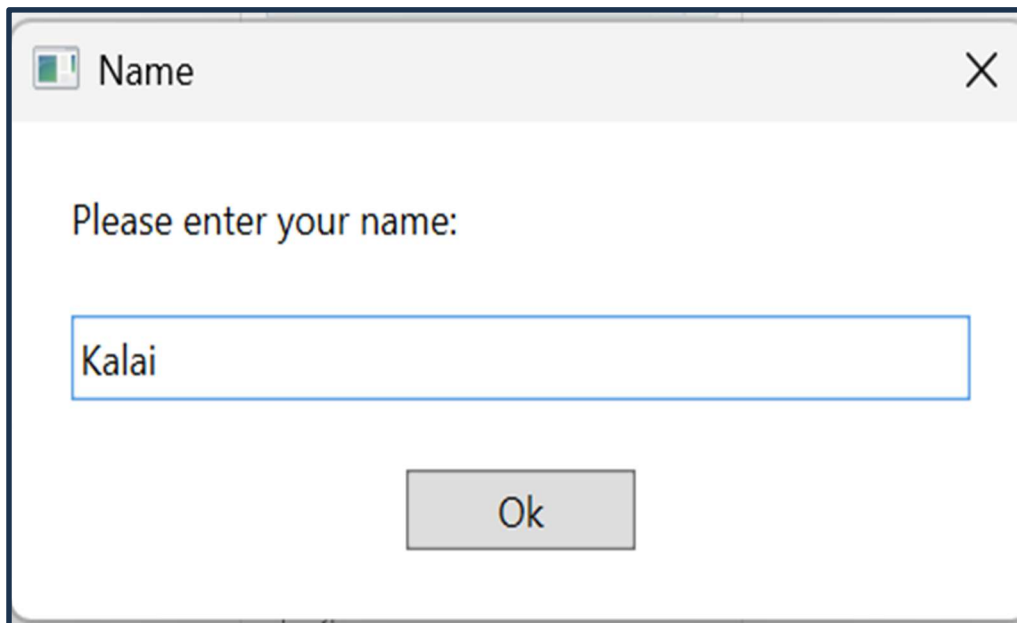
- **Email Notification:** If an email address was provided, then **Send SMTP Mail Message** activity to send a confirmation email to the user, detailing their booking information.

Once a reservation was confirmed, the bot updated the Excel file to mark the **Availability Status** of the booked time slot as **Booked**. This ensured that the system's availability was always up to date.

Once all confirmation were sent, the workflow was completed, and the program tested for accuracy and reliability.

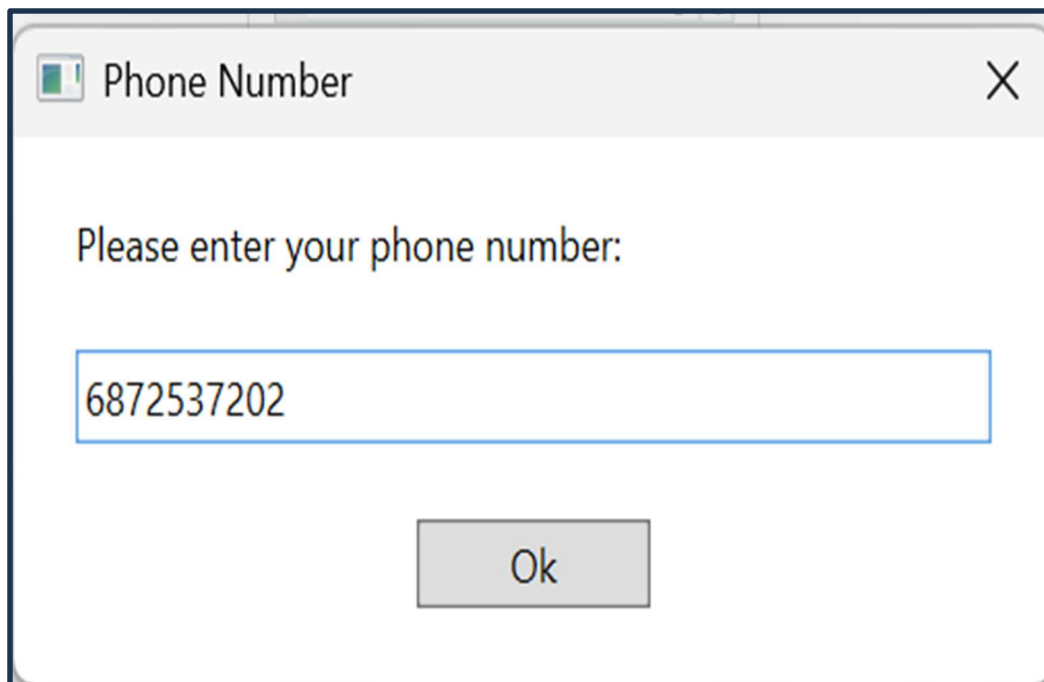
CHAPTER 5

OUTPUT SCREENSHOTS



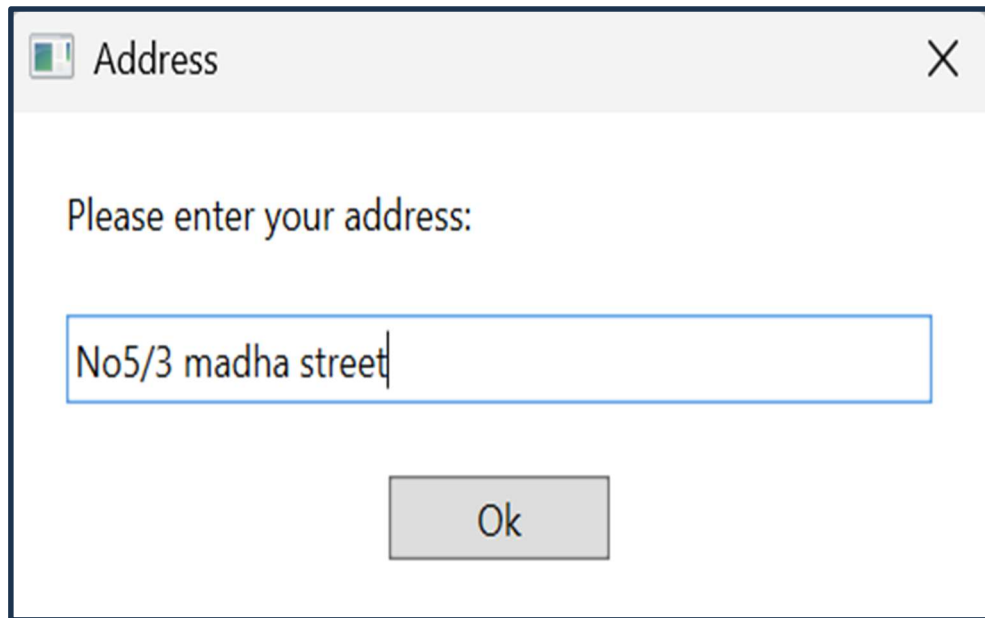
A screenshot of a Java-style dialog box titled "Name". The dialog box has a standard title bar with a close button (X) in the top right corner. Inside the dialog, the text "Please enter your name:" is displayed. Below this text is a text input field containing the name "Kalai". At the bottom center of the dialog is an "Ok" button.

Fig 5.1 Name Input Box



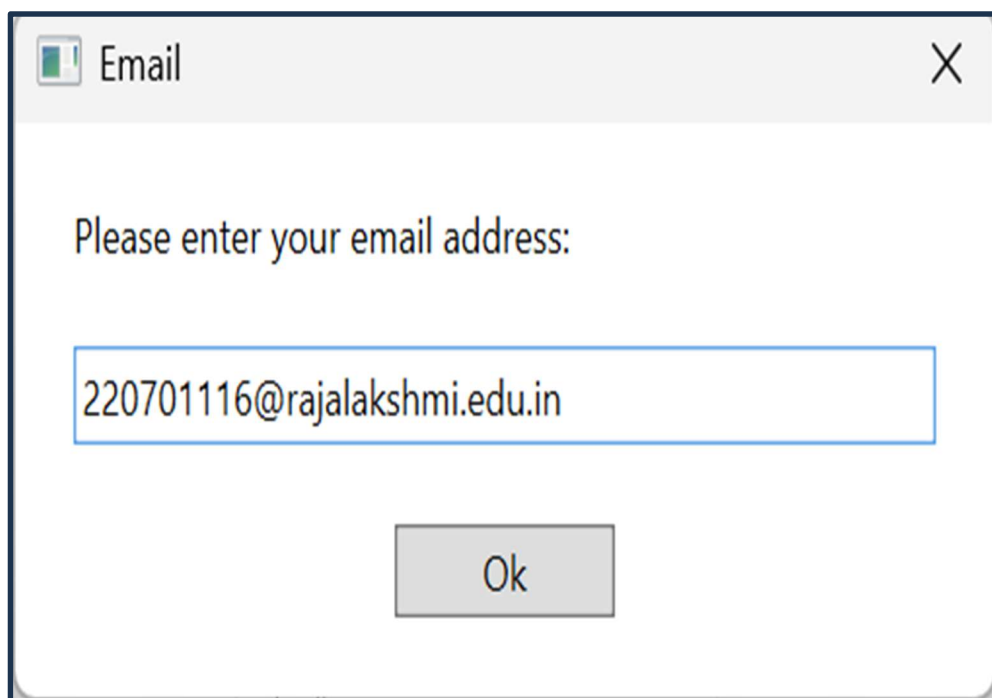
A screenshot of a Java-style dialog box titled "Phone Number". The dialog box has a standard title bar with a close button (X) in the top right corner. Inside the dialog, the text "Please enter your phone number:" is displayed. Below this text is a text input field containing the phone number "6872537202". At the bottom center of the dialog is an "Ok" button.

Fig 5.2 Phone no Input Box



A screenshot of a Java Swing dialog box titled "Address". The dialog has a standard title bar with a close button (X) in the top right corner. Inside the dialog, the text "Please enter your address:" is displayed. Below this text is a text input field containing the text "No5/3 madha street". At the bottom center of the dialog is an "Ok" button.

Fig 5.3 Address Input Box



A screenshot of a Java Swing dialog box titled "Email". The dialog has a standard title bar with a close button (X) in the top right corner. Inside the dialog, the text "Please enter your email address:" is displayed. Below this text is a text input field containing the email address "220701116@rajalakshmi.edu.in". At the bottom center of the dialog is an "Ok" button.

Fig 5.4 Email Input Box

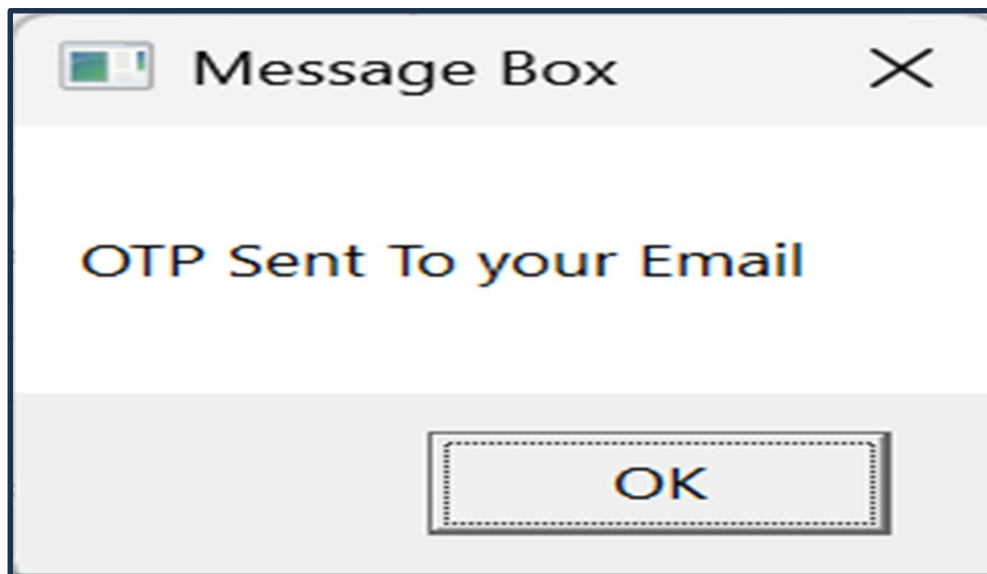


Fig 5.4 Message Box

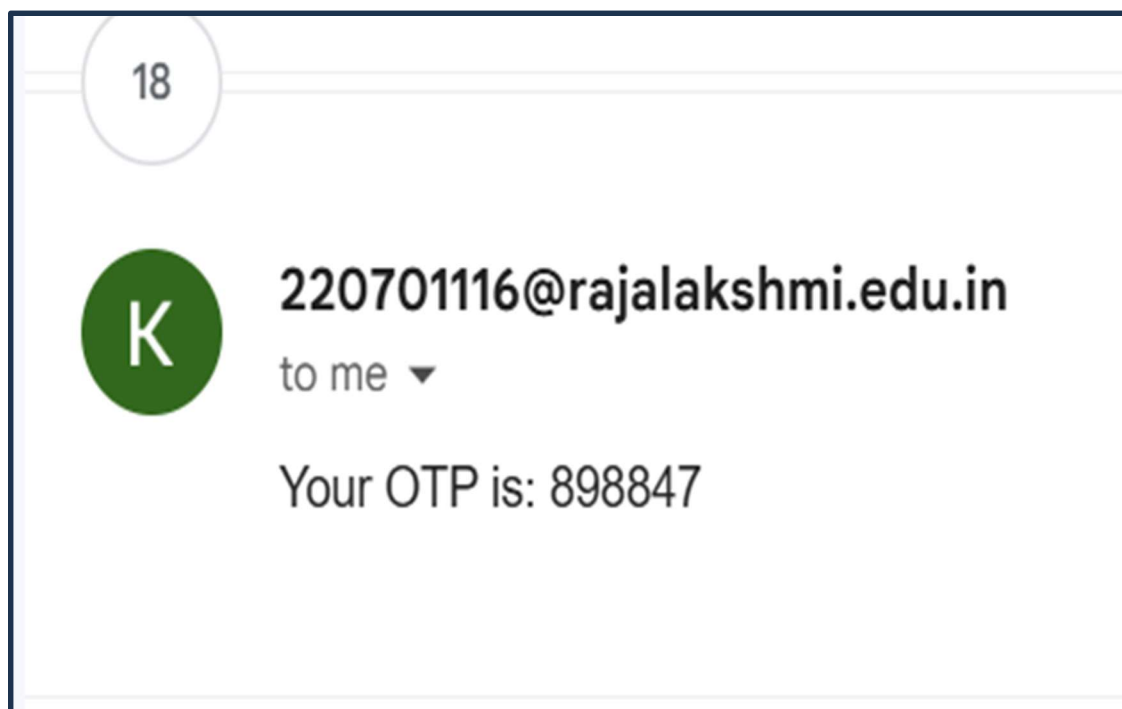


Fig 5.5 OTP from Email

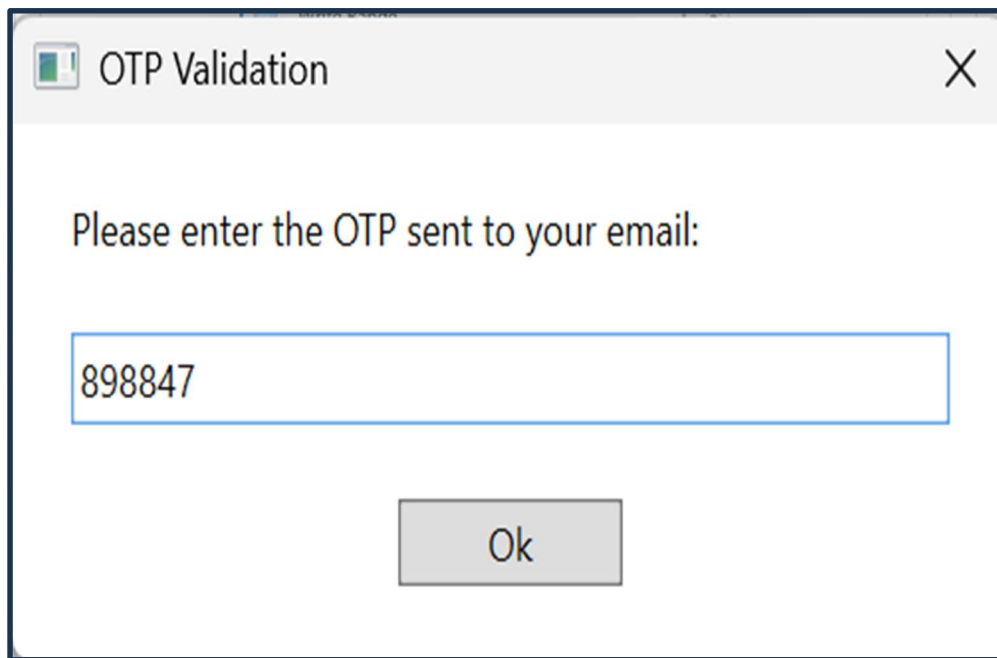


Fig 5.6 OTP Validation Box

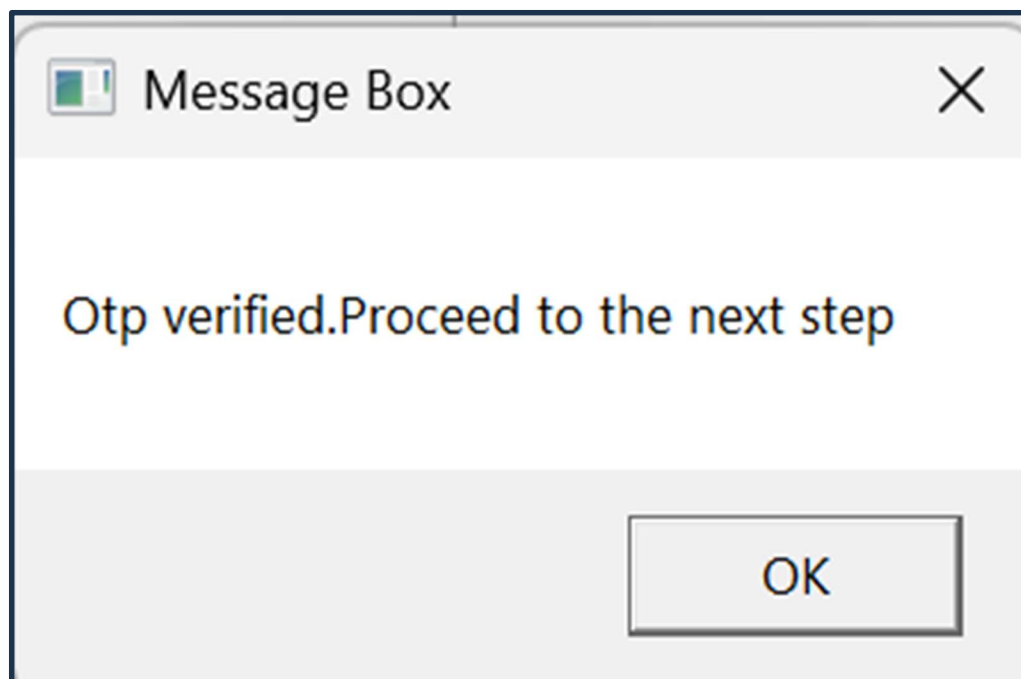


Fig 5.7 OTP verified

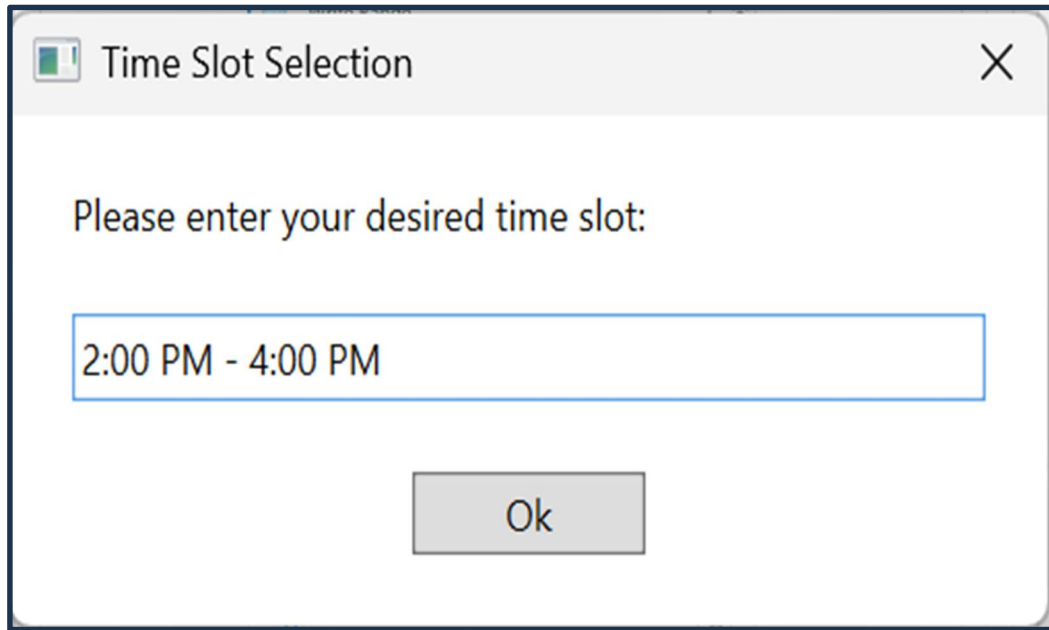


Fig 5.8 Time Slot Selection Box

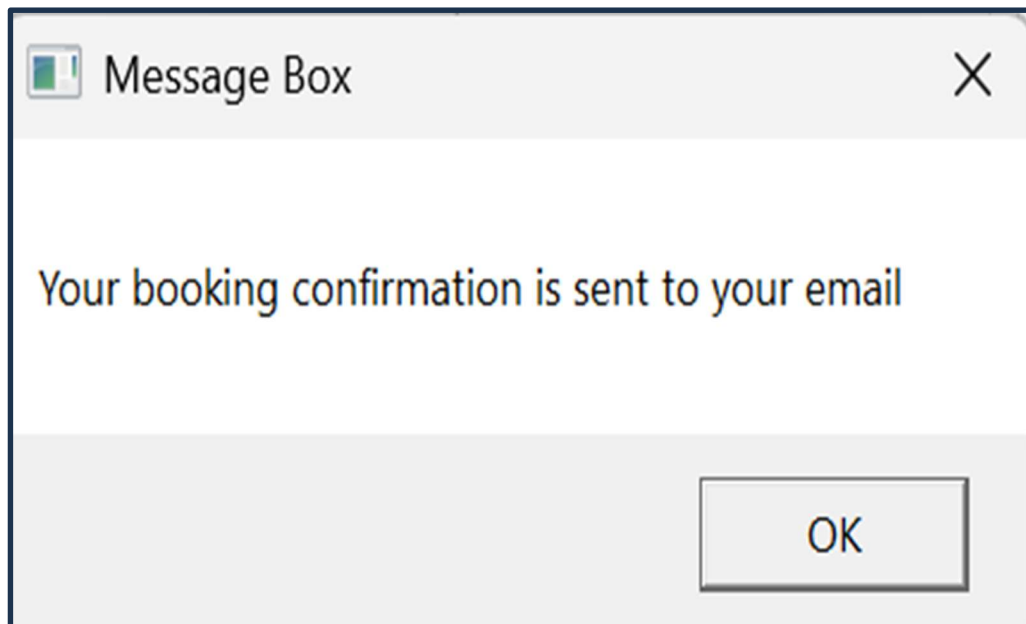


Fig 5.9 Succesfully Email Delivered

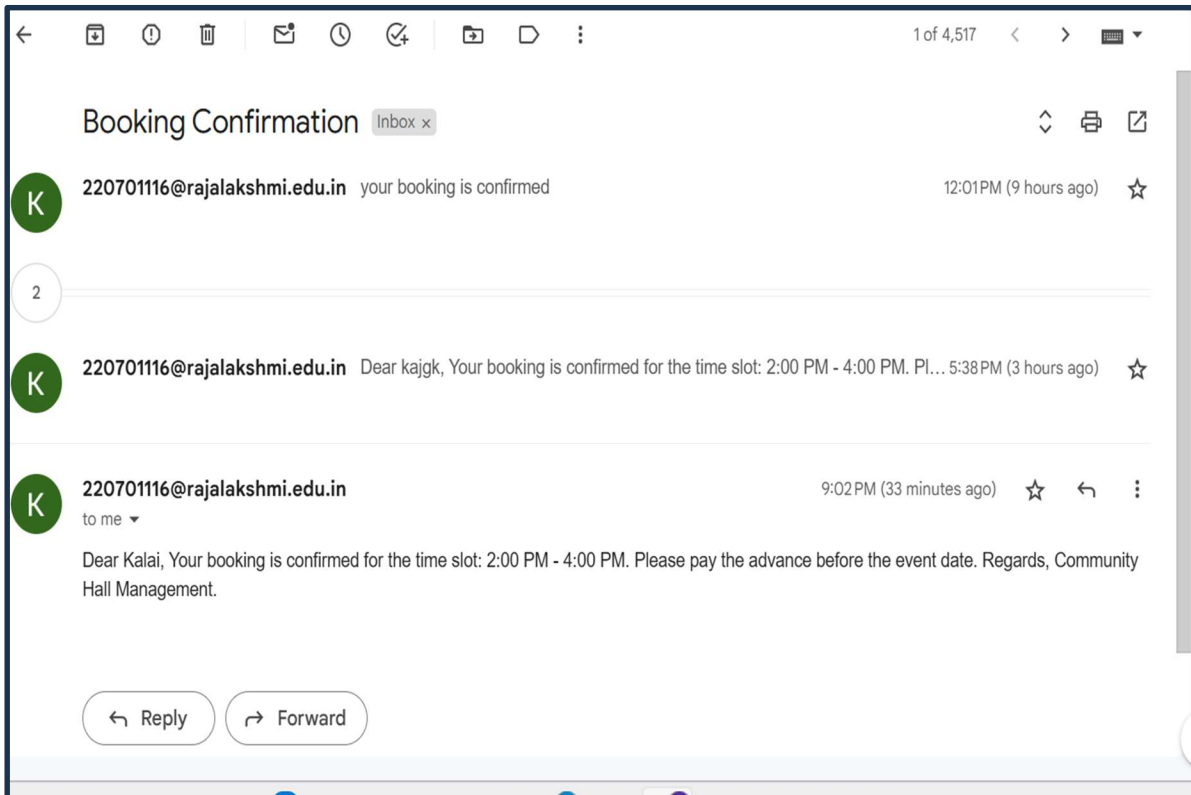


Fig 5.10 Confirmation in Email

File Home Insert Page Layout Formulas Data Review View Help				
<div> <div>Clipboard</div> <div> <div>Font</div> <div> Calibri 11 A⁺ A⁻ </div> <div> B I U </div> <div> </div> </div> <div> <div>Alignment</div> <div> </div> <div> </div> </div> <div> <div>Number</div> <div> General </div> <div> </div> </div> <div> <div>Conditional Formatting</div> </div> </div>				
D6				
	A	B	C	D
1	Time Slot	Status		
2	4:00 PM - 6:00 PM	Booked		
3	6:00 PM - 8:00 PM	Available		
4	8:00 PM - 10:00 PM	Available		
5	10:00 PM - 12:00 AM	Booked		
6	12:00 AM - 2:00 AM	Available		
7	2:00 AM - 4:00 AM	Booked		
8	4:00 AM - 6:00 AM	Available		
9	6:00 AM - 8:00 AM	Booked		
10	6:00 AM - 8:00 AM	Booked		
11	8:00 AM - 10:00 AM	Available		

Fig 5.10 Booking Details File

CHAPTER 6

CONCLUSION

The Community Hall Reservation System demonstrates the effective use of digital automation to streamline the process of booking and managing reservations for community events. By utilizing a user-friendly web interface and backend database management, the system automates the tasks of checking availability, processing bookings, and sending confirmation notifications, ensuring a smooth experience for both users and administrators. The integration of real-time conflict checks and data validation enhances the efficiency of the reservation process, reducing the likelihood of errors and double bookings.

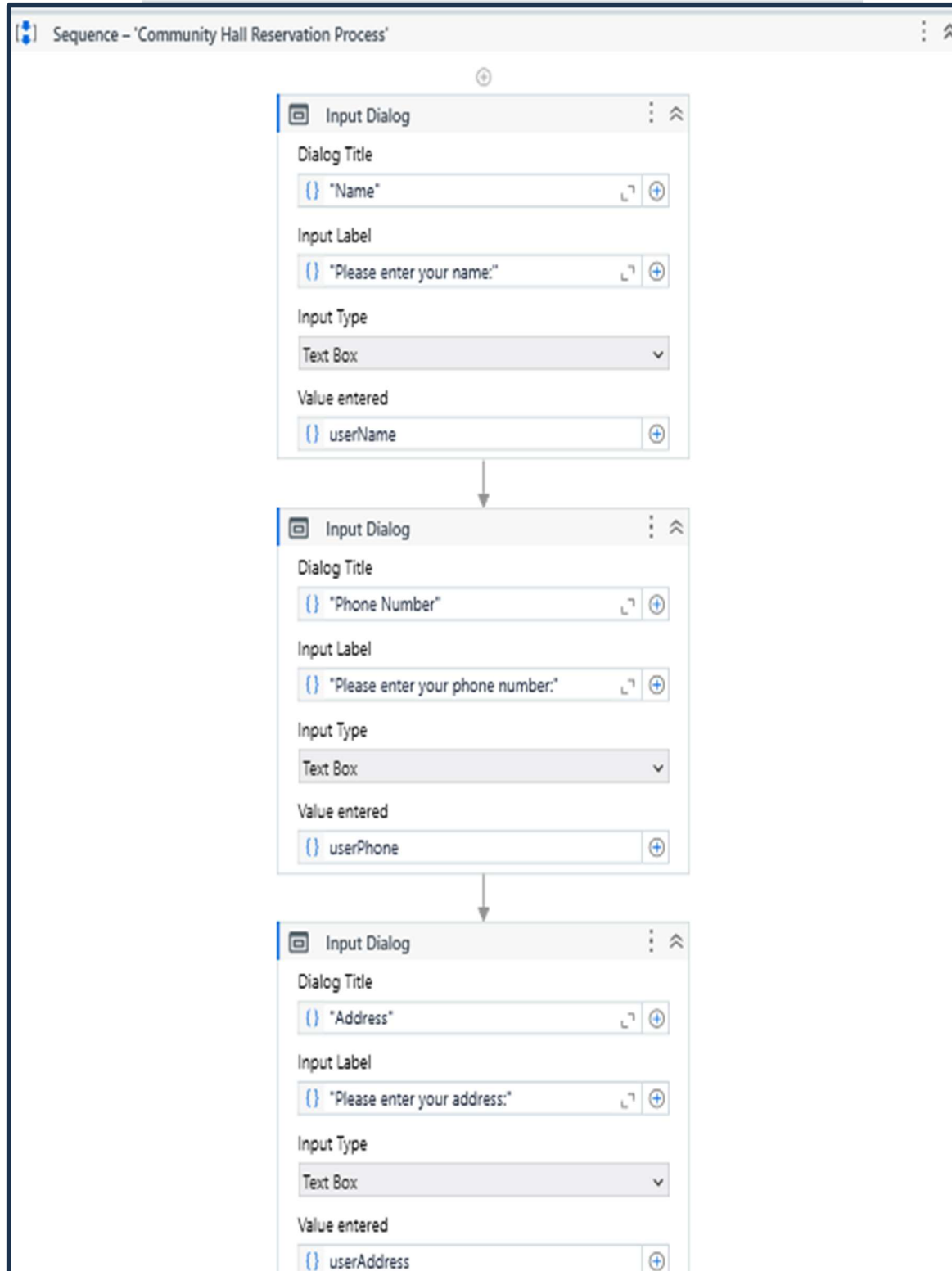
This project highlights the potential of automation in resource management and scheduling, significantly reducing manual effort while enhancing user satisfaction. With features such as easy modification or cancellation of bookings, real-time updates, and administrative oversight, the system caters to a wide range of user needs and event types.

The successful implementation of the Community Hall Reservation System proves its reliability and scalability, making it a valuable solution for managing community spaces. With potential enhancements, such as mobile app integration, payment processing, and advanced reporting tools, the system could evolve into a comprehensive platform for managing various types of reservations across multiple venues.

In conclusion, the project not only simplifies the reservation process but also offers a robust foundation for exploring further applications of automation in the management of shared resources.

APPENDIX

SAMPLE PROCESS



Input type
Text Box

Value entered
{ } userAddress

Input Dialog

Dialog Title
{ } "Email"

Input Label
{ } "Please enter your email address:"

Input Type
Text Box

Value entered
{ } userEmail

Assign

Save to		Value to save
{ } otpCode	=	{ } New Random().Next(1

Send SMTP Mail Message

To * { } userEmail

Subject { } "Your OTP for Community Hall Re

Body { } "Your OTP is: " + otpCode

[Attach Files](#)

Message Box

Text *
{ } "OTP Sent To your Email"

Input Dialog

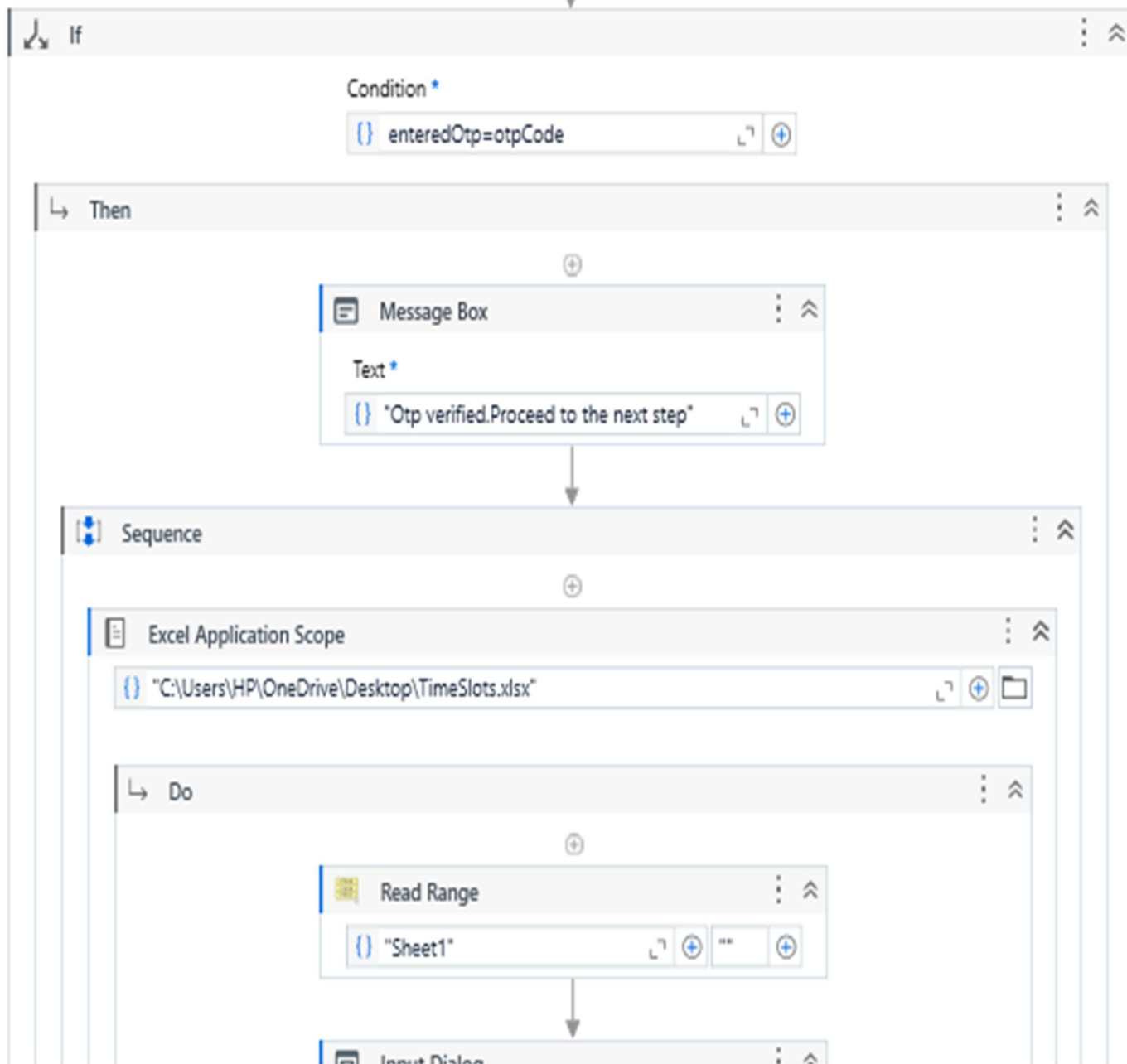
Input Dialog

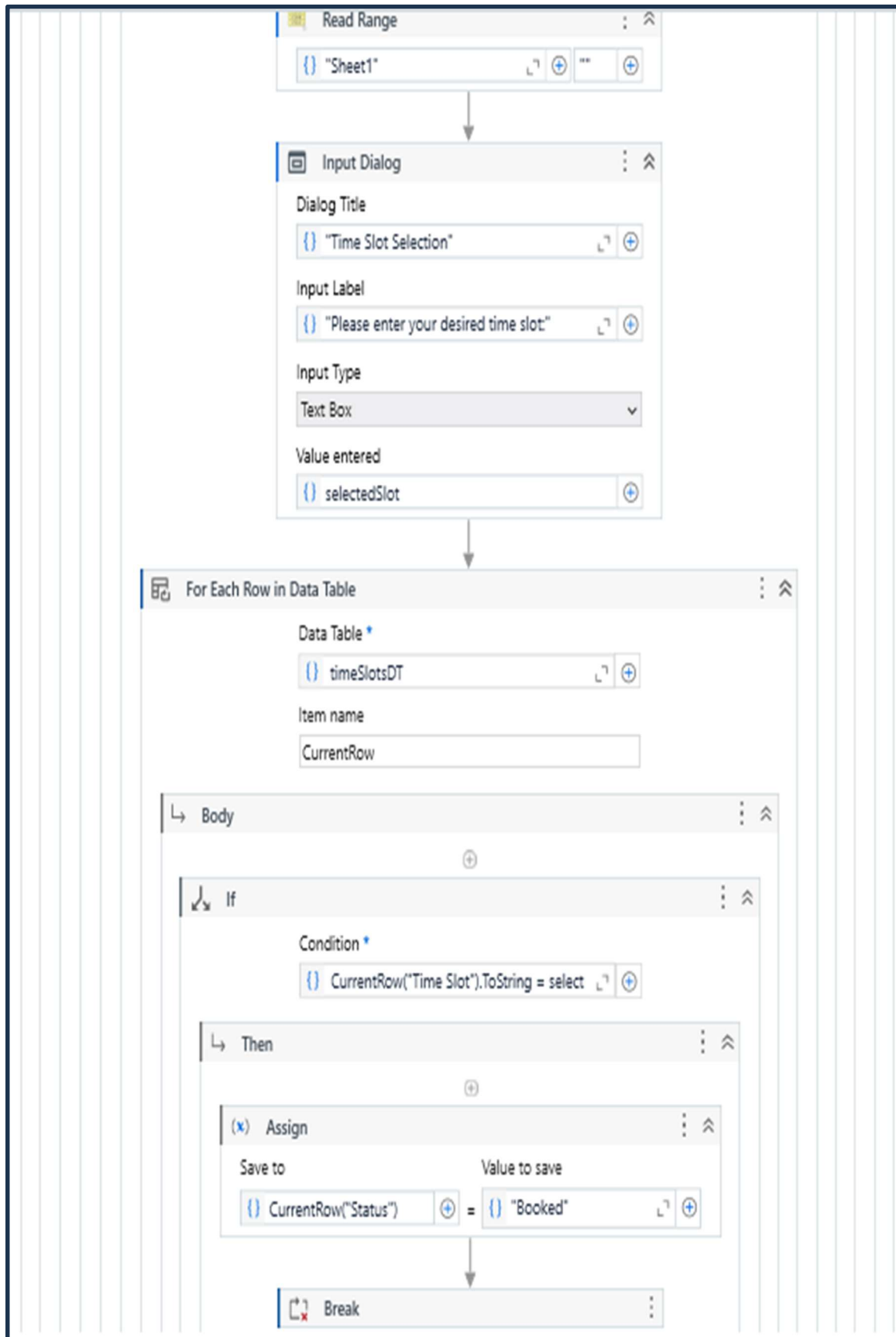
Dialog Title
{ } "OTP Validation" L¹ +

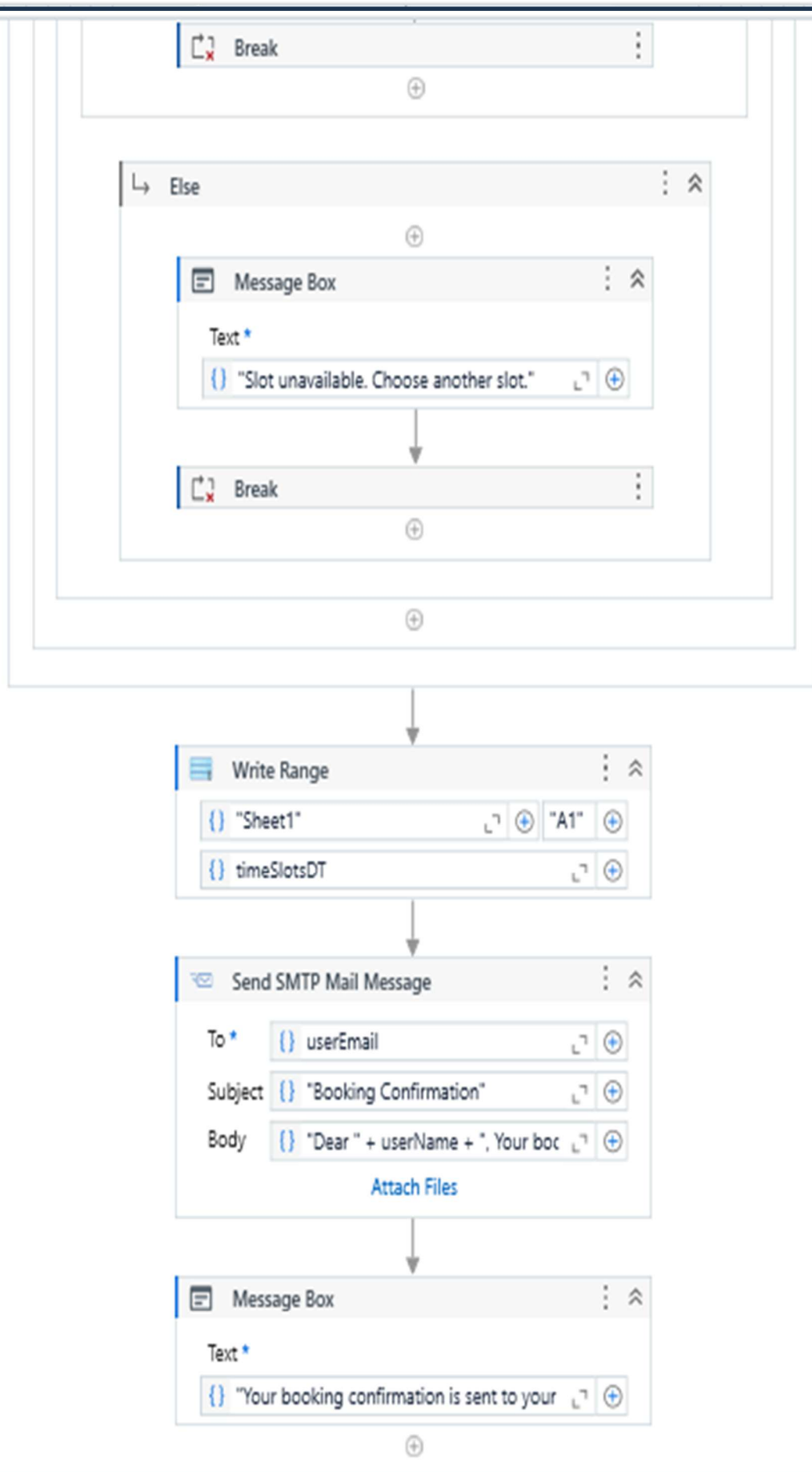
Input Label
{ } "Please enter the OTP sent to your email:" L¹ +

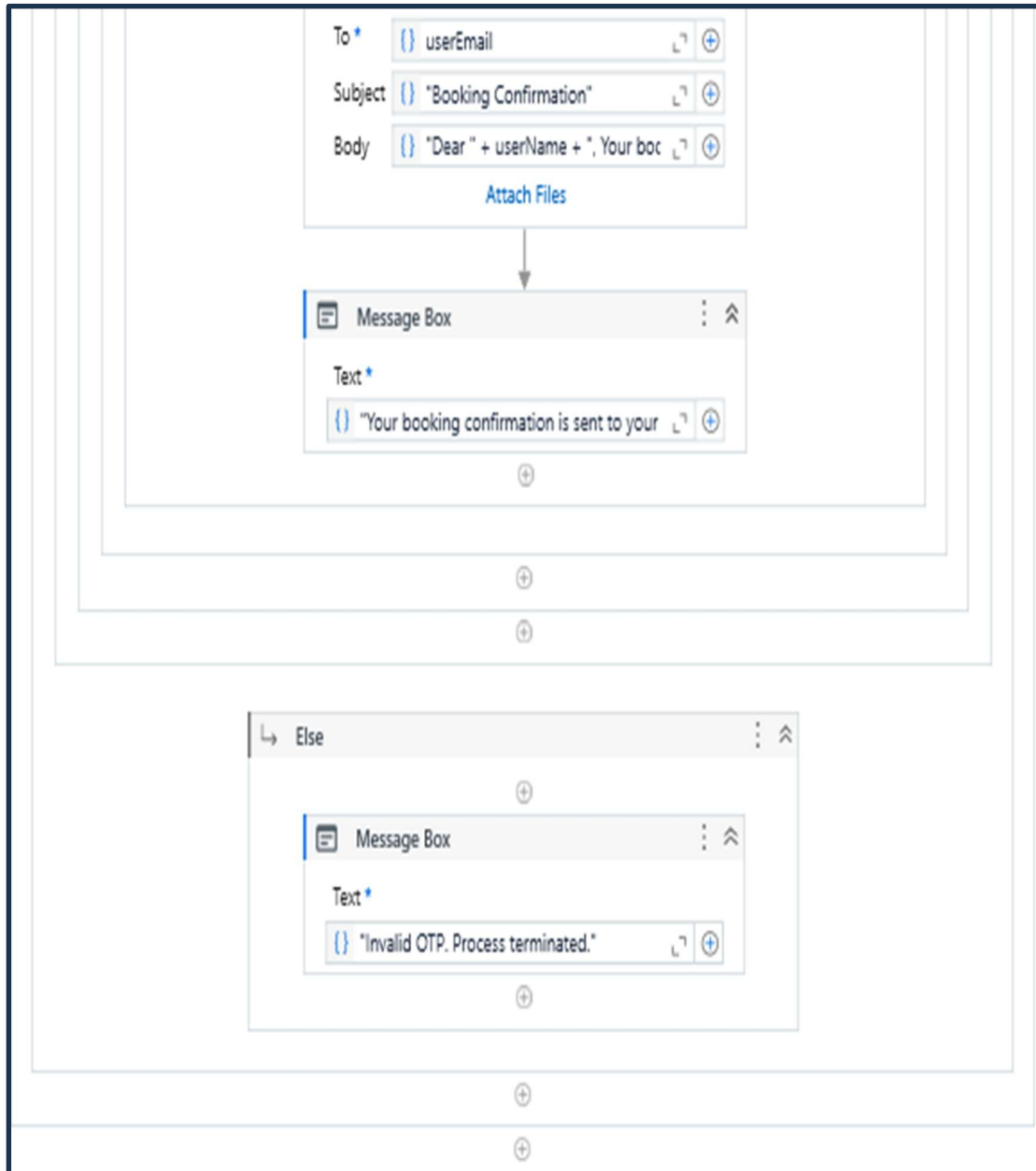
Input Type
Text Box

Value entered
{ } enteredOtp +









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/>