WHATSAPP BOT

# A PROJECT REPORT

***Submitted by***

**J JANANI (220701097)**

***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 **“WHATSAPP BOT”** is the Bonafede work of **“J JANANI (220701097)”** who carried out the project work for the subject OAI1903-Introduction to Robotic Process Automation under my supervision.

Mrs. J. Jinu Sophia

# 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 .

**TABLE OF CONTENTS**

CHAPTER NO. TITLE PAGE NO. ABSTRACT 5

[LIST OF TABLES 6](#_TOC_250002)

[LIST OF FIGURES 7](#_TOC_250001)

[LIST OF ABBREVIATIONS 9](#_TOC_250000)

1. INTRODUCTION 10
   1. GENERAL
   2. OBJECTIVE
   3. EXISTING SYSTEM
   4. PROPOSED SYSTEM
2. **LITERATURE REVIEW 12**
   1. GENERAL
3. **SYSTEM DESIGN 14**
   1. GENERAL
      1. SYSTEM FLOW DIAGRAM
      2. ARCHITECTURE DIAGRAM
      3. SEQUENCE DIAGRAM
      4. WORKFLOW
4. **PROJECT DESCRIPTION 18**
   1. METHODOLOGIES
      1. MODULE
5. **OUTPUT SCREENSHOTS**
6. **CONCLUSIONS 20**
   1. GENERAL

**REFERENCES 21**

**APPENDICES 22**

# 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.

# J JANANI (220701097)

**ABSTRACT**

The WhatsApp Bot developed in UiPath Studio automates the process of sending scheduled messages to specified contacts. The bot prompts the user to enter a phone number or contact name, the message to be sent and the preferred date and time for delivery. Using this information, the bot automatically searches for the contact on WhatsApp and ensures the message is sent at the specified time, eliminating the need for manual intervention.

This project enhances communication efficiency by streamlining the scheduling of messages, ensuring they are delivered precisely as planned. By automating this task, it offers time-saving benefits to individuals and businesses who need to send reminders, notifications, or messages at specific times without being online. The bot simplifies repetitive messaging tasks and provides a reliable solution for automating WhatsApp communications, making it highly suitable for marketing, reminders, and personal use.

# LIST OF TABLES

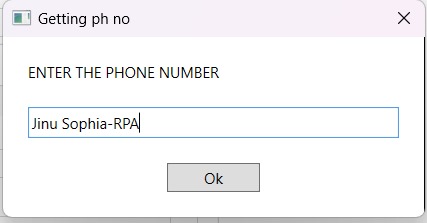
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **TABLE NO.** | **FIELDS** | **DESCRIPTION** | |  | Variables Used in the WhatsApp Bot Project | Details of all the variables used in the bot, including their types, scope, and purpose. | |  | Module Implementation Details | Overview of the modules implemented (e.g., input collection, validation, scheduling). | |  | Testing Scenarios and Outcomes | Scenarios tested during the project (e.g., valid vs invalid inputs, contact search success/failure) and their outcomes. | |  | User Input Validation Rules | Rules and constraints for user inputs, such as phone number format, message length, and date-time validation. | |  | Error Handling and Notifications | Error-handling strategies and how the bot notifies users of invalid inputs or failed message delivery. | |  |  |

# Description:

* **Purpose**: The tables provide structured information about variables, modules, testing scenarios, and other key aspects of the WhatsApp Bot project, ensuring clarity and organization.
* **Utility**: They serve as a reference for understanding the data flow, functionality, and validation mechanisms implemented in the project.

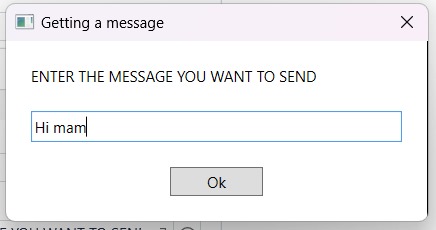
# LIST OF FIGURES

1. Enter Phone Number:



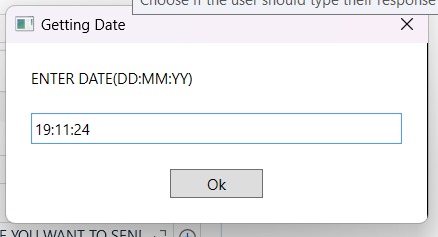
**Fig.1**

1. Enter Message:



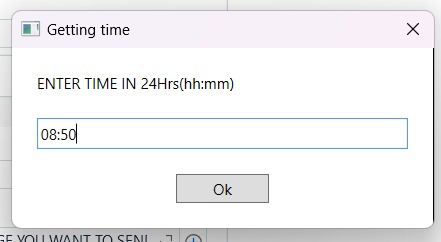
**Fig.2**

1. Enter Date:



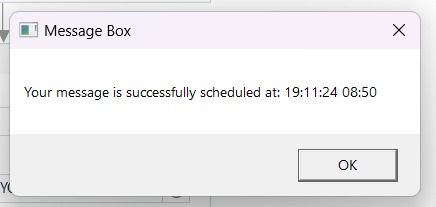
**Fig.3**

1. Enter Time:



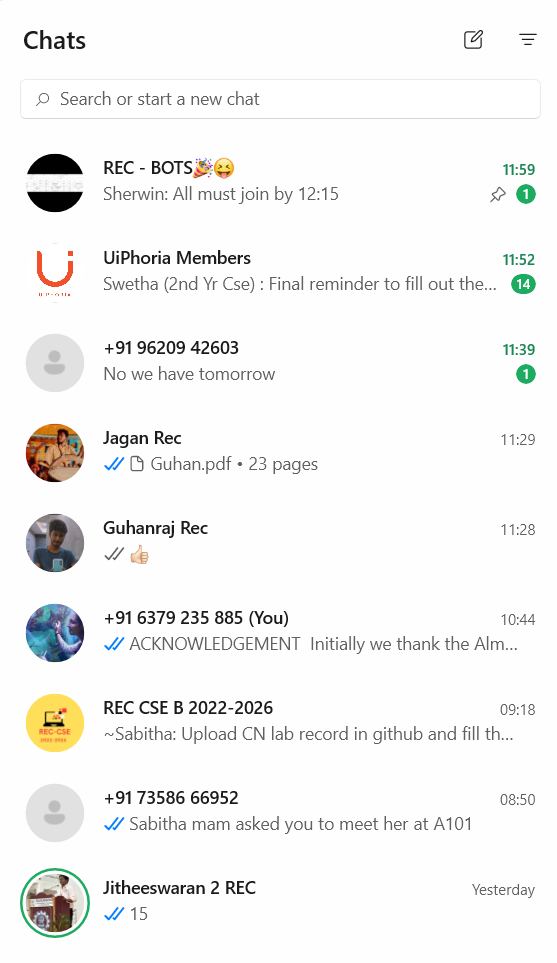
**Fig.4**

1. Scheduled Message:



**Fig.5**

1. WhatsApp Search:



**Fig.6**

# LIST OF ABBREVIATIONS

|  |  |
| --- | --- |
| **Variable** | **Abbreviation** |
| **CLI** | Command Line Interface |
| **JSON** | JavaScript Object Notation |
| **UI** | User Interface |
| **HTTP** | Hypertext Transfer Protocol |
| **RPA** | Robotic Process Automation |
| **AI** | Artificial Intelligence |
| **BOT** | Robot |
| **API** | Application Programming Interface |
| **GUI** | Graphical User Interface |
| **IDE** | Integrated Development Environment |

**INTRODUCTION**

In today's fast-paced digital world, effective communication is key, especially when it comes to managing messages in a timely manner. This bot automates the process of sending scheduled messages by allowing users to input the contact information, the message content, and the scheduled date and time.

Once the data is provided, the bot automatically searches for the contact on WhatsApp and sends the message at the specified time. This solution enhances convenience by eliminating the need for manual intervention, thus making communication more efficient and reliable, particularly for routine or time-sensitive messages.

* 1. **GENERAL**

The proposed WhatsApp bot, created using UiPath Studio, is designed to automate the process of sending scheduled messages. Users input the contact's phone number or name, the message content, and the preferred date and time for delivery. The bot then handles the rest by searching for the contact on WhatsApp and sending the message at the designated time. This system offers a more streamlined, reliable, and hands-off approach to messaging, ensuring timely delivery without user involvement

* 1. **OBJECTIVE**

The main objective of this project is to develop an automated system that allows users to schedule WhatsApp messages in advance. The bot prompts the user for the necessary details, such as contact information, message content, and the desired time, and then sends the message automatically without any further manual effort. This project aims to improve efficiency by automating communication tasks, thus saving time and ensuring that important messages are sent on time.

* 1. **EXISTING SYSTEM**

Currently, WhatsApp does not provide a built-in feature for scheduling messages. Users often rely on external applications or manual reminders to send messages at specific times, but these methods are either cumbersome or unreliable. Third-party tools that claim to offer message scheduling often require the user to keep the application open or active, which is inconvenient and does not offer a fully automated solution.

* 1. **PROPOSED SYSTEM**

The proposed WhatsApp bot system will address the limitations of existing systems by fully automating the process of scheduling and sending messages. Users only need to provide the recipient's contact information, the message, and the scheduled time. The bot will take care of the rest, searching for the contact on WhatsApp and sending the message at the scheduled time without requiring any further input. This system provides a reliable, convenient, and hands-off solution for message scheduling.

# LITERATURE REVIEW

The literature review highlights the challenges and limitations of existing systems for scheduling messages on WhatsApp. Currently, WhatsApp does not provide a native feature for scheduling messages, forcing users to rely on manual methods or third-party tools. Robotic Process Automation (RPA) platforms like UiPath provide a reliable framework for automating such processes, ensuring accurate and timely communication without manual intervention. These findings underscore the need for an automated WhatsApp bot that can overcome the limitations of existing systems and provide a secure, efficient, and user-friendly solution for message scheduling.

**2**.**1 GENERAL**

A comprehensive review of existing literature and systems related to WhatsApp message scheduling was conducted. Key findings include:

**Manual Scheduling Challenges:**

WhatsApp lacks a built-in feature for scheduling messages, leading users to manually send messages or use third-party apps.

Manual scheduling is prone to human error, such as forgetting to send messages on time, particularly for repetitive tasks.

**Third-Party Tools:**

Several third-party applications claim to offer message scheduling but have significant limitations:

Most require the app to remain active or the device to be unlocked at the scheduled time. Security concerns arise as some tools request unnecessary permissions or require access to user data. Compatibility issues are common due to frequent WhatsApp updates.

**Automation in Messaging Systems:**

Automation has been successfully implemented in other messaging platforms and tasks, demonstrating the feasibility of automating WhatsApp message scheduling. Robotic Process Automation (RPA) tools like UiPath provide a secure and robust framework for building such automation, leveraging UI interactions and time-based triggers.

**Advantages of Automation:**

Automating message scheduling ensures timely communication without the need for user intervention.

It reduces the risk of human error and enhances efficiency for both personal and professional messaging needs.

Automation eliminates dependency on unreliable third-party applications.

**SYSTEM DESIGN**

**3.1 GENERAL**

This section details the overall system design of the WhatsApp Bot, providing diagrams and workflows to explain the architecture and execution.

**3.1.1 SYSTEM FLOW DIAGRAM**

The system flow diagram illustrates the process flow:

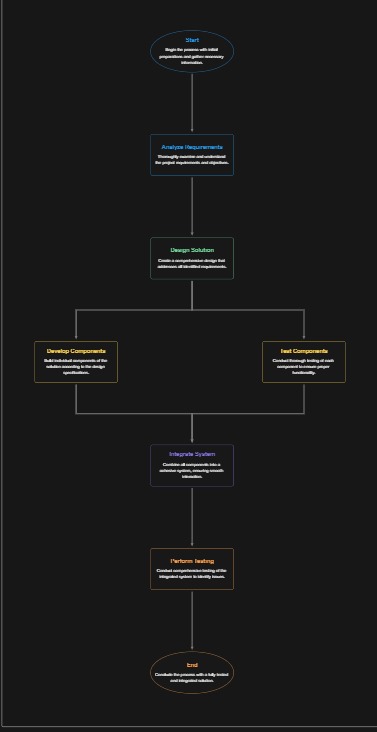
*Input Gathering:* User enters the contact details, message, and scheduled date/time.

*Data Validation*: The bot validates the contact and schedule information.

*Message Scheduling*: The message is scheduled using the input parameters.

*WhatsApp Interaction*: The bot searches for the contact on WhatsApp and sends the message.

*Completion and Feedback*: Confirmation is provided to the user.



**Fig.7**

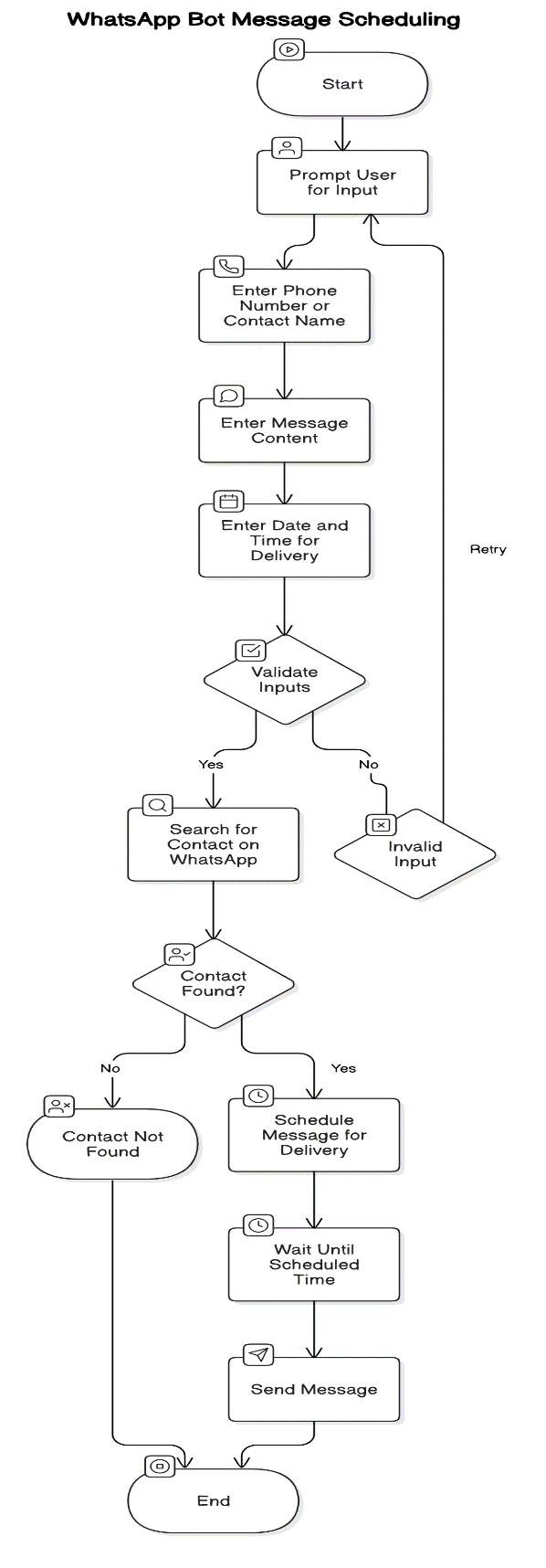
**3.1.2 ARCHITECTURE DIAGRAM**

The architecture diagram outlines the bot's components:

*User Interface*: Interface for inputting details (UiPath forms or Excel/CSV).

*Orchestrator*: Manages task schedules and triggers.

*UiPath Bot*: Executes the automation process, interacts with the WhatsApp desktop/web app.

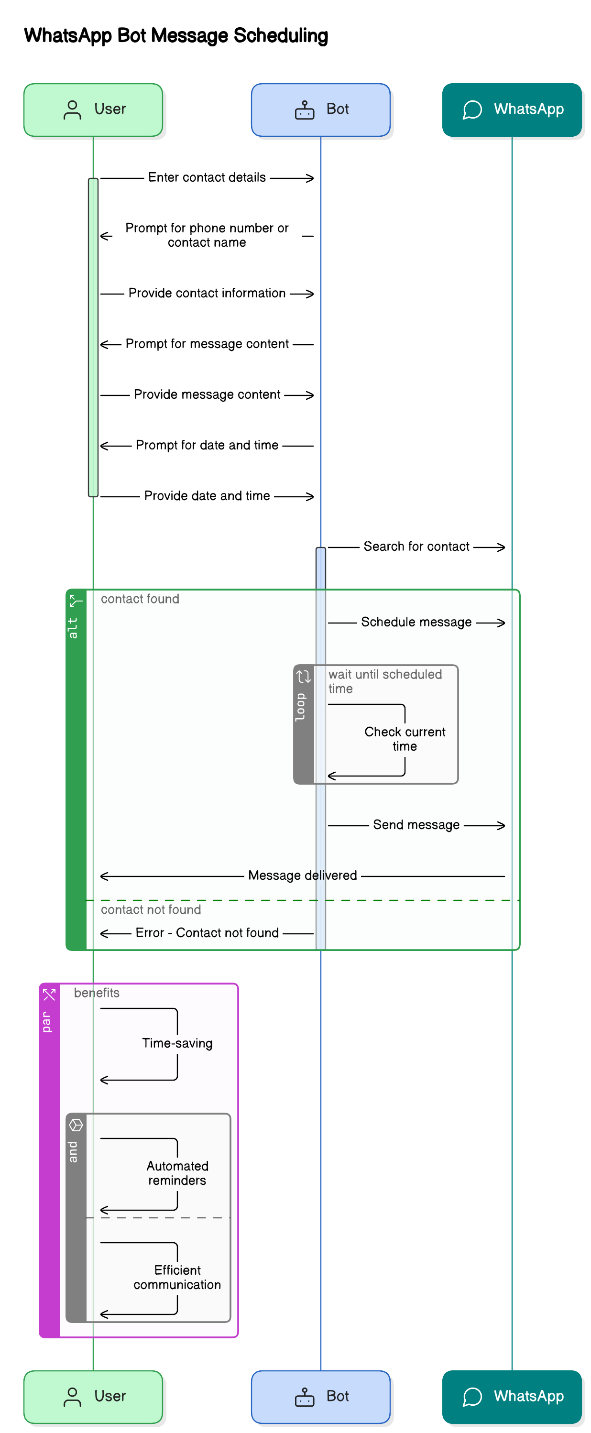


**Fig.8**

**3.1.3 SEQUENCE DIAGRAM**

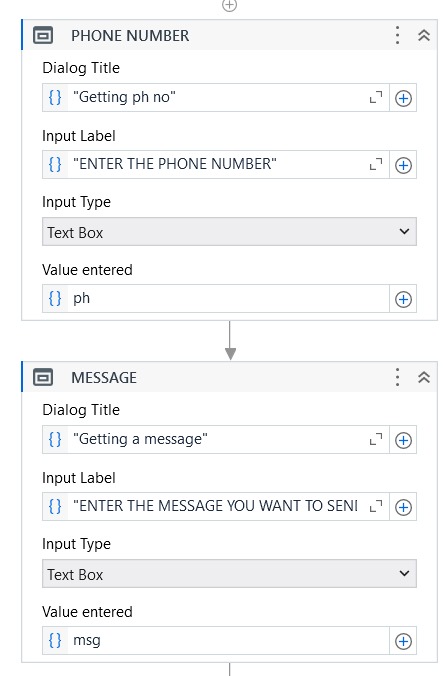
The sequence diagram demonstrates interaction between system components:

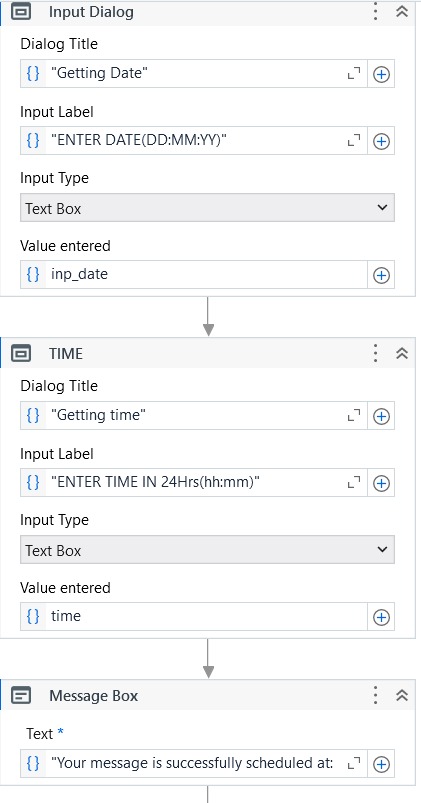
* User inputs data into the bot.
* Bot processes and validates data.
* Scheduler queues the task.
* WhatsApp GUI/API processes and sends the message.
* User receives delivery confirmation.

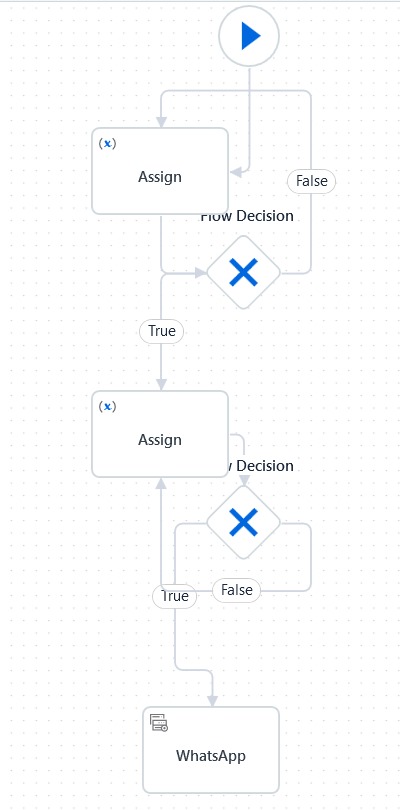


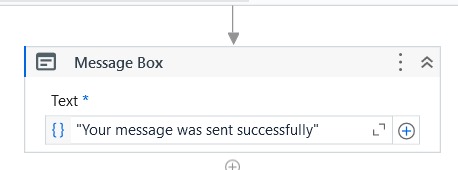
**Fig.9**

3.1.4 WORKFLOW









**Fig.10**

**PROJECT DESCRIPTION**

* 1. **METHODOLOGIES**

The WhatsApp Bot is structured following a logical methodology that guarantees, its feasibility, reliability, and user-friendliness in the following aspects:

**Requirement Gathering:**

* Understand what specific needs are to be met when it comes to scheduling and automating WhatsApp messages.
* Identify expectations that can be built into it, such as inputting contact details, customizing messages, and delivery schedules.
* Study on the limitations of WhatsApp and the possibility of using UiPath Studio automation.

**Design**

* Design the system flow that represents how the bot processes input and executes tasks.
* Draw architecture diagrams for structuring module interaction.
* Use sequence diagrams to define how the bot will operate step-by-step from collecting inputs to delivering messages.

**Implementation**

* Design automation workflows in UiPath Studio with components that involve user inputs, validation, scheduling, and message delivery
* Design an application interface based on Graphical User Interface interaction which shall be used to interface the WhatsApp interaction and ensure smooth, secure message delivery.
* Test cross-version and scenario compatibility with WhatsApp

**4.1.1 MODULES**

The project is divided into some interdependent modules to ensure smooth functioning as follows:

**User Interface Module**

* It shall provide the user interface for inputting the contact details either the phone number or name, message content, and delivery schedule. It shall ensure intuitive design so that input errors are minimized.

**Validation Module**

* It verifies the accuracy of user inputs such as cross validation with respect to phone numbers, verifying the date and time to ensure it is not in the past.
* Prevents scheduling errors by rejecting incomplete or incorrect data.

**Scheduler Module**

* Manages the scheduling logic, ensuring messages are triggered at the specified date and time.
* Handles time-based events using UiPath's timer activities and workflows.
* Ensures the system remains idle until the specified trigger time.

# CONCLUSION

# The bot also demonstrated the capacity to schedule, verify, and dynamically interact with the WhatsApp interface. It gives users an opportunity to efficiently send reminders, notifications, or marketing messages. This innovation is more specifically relevant to industries where timely communication is critical-such as in healthcare (appointment reminders), education (event notifications), or business (customer engagement).

Beyond its functionality, this bot showcases the UiPath Studio's ability as a development platform for creating meaningful RPA solutions. This project is scalable, allowing for further integrations with more messaging platforms and APIs. Its modular design ensures that the bot can be easily adapted for personal use or enterprise-level implementations. In addition, its user-friendly interface makes it accessible to those with non-technical skills.

In conclusion, the WhatsApp Bot serves as a robust and innovative solution for addressing the challenges of timely communication. It bridges the gap between manual effort and automation, offering users a seamless and efficient alternative to traditional messaging workflows. As communication needs evolve, this project lays the groundwork for further enhancements, paving the way for smarter and more automated messaging systems.

# REFERENCES

**Title:** Multi-Channel Chatbot and Robotic Process Automation

**Authors:** Gota Dan; Domuta Claudiu; Fanca Alexandra; Pop-Puscasiu Adela; Stan Ovidiu; Valean Honoriu

**Publisher:** IEEE

**Link :** [**https://ieeexplore.ieee.org/abstract/document/9801960**](https://ieeexplore.ieee.org/abstract/document/9801960)

**Title:** Adopted Acceptance Test-Driven Development (ATDD) to produce Robotic Process Automation (RPA) for reducing teaching workload

**Authors:** Nuttaporn Phakdee; Wantana Sisomboon; Jirayus Arbking

**Publisher:** IEEE

**Link:** [**https://ieeexplore.ieee.org/abstract/document/10460654**](https://ieeexplore.ieee.org/abstract/document/10460654)

**APPENDICES**

### **Main Workflow Script (e.g., Main.xaml or Main.py):**

* Manages the core logic of the WhatsApp Bot, including:

1. Message scheduling and validation of input details (contact name/number, message content, and schedule).
2. Automated interaction with the WhatsApp interface for sending messages

### **INPUT Folder:**

* Stores essential files required for the bot's operations, such as:
  1. **ScheduledMessages.xlsx**: Contains a list of messages with respective contact details and time schedules for batch processing.

### **OUTPUT Folder:**

* Stores execution logs, sent message details, and status reports, for auditing and debugging purposes. Examples:
  1. **MessageLog.txt**: A text file logging messages sent, with timestamps and recipient details.
  2. **ErrorReport.json**: Captures any errors encountered during bot execution for troubleshooting.