

Birthday reminder bot  
**A PROJECT REPORT**

*Submitted by*

**AARTHI (220701001)**

*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 “**TITLE**” is the Bonafide work of “**NAME(ROLLNO)**” 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 \_\_\_\_\_.

## **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. 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.

**AARTHI (220701001).**

## **ABSTRACT :**

A birthday reminder bot is a system designed to simplify and automate the process of remembering and celebrating important dates. It collects user data, such as names, birth dates, and preferences for notifications, through manual input, integrations with calendars, or importing files like CSV. This data is stored securely in a structured database, which serves as the backbone of the system. Using a processing engine, the bot calculates upcoming birthdays, taking into account recurring events, user preferences, and special cases like leap years. It then schedules and delivers timely notifications through multiple channels, including SMS, email, push notifications, or chat platforms like WhatsApp and Telegram.

The bot also features a user-friendly interface, such as a mobile app, web dashboard, or chatbot, allowing users to interact with the system effortlessly. Users can add, edit, or delete birthdays, configure notification timings, and even personalize messages for each recipient. For added convenience, the bot can integrate with external systems, such as Google Calendar or social media platforms, to sync data and provide enhanced functionality. By combining automation, customization, and ease of use, the birthday reminder bot ensures users never miss a special day, making it a valuable tool for both personal and professional use.

## TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	<b>ABSTRACT</b>	<b>iv</b>
	<b>LIST OF TABLE</b>	<b>v</b>
	<b>LIST OF FIGURES</b>	<b>vi</b>
	<b>LIST OF ABBREVIATIONS</b>	<b>vii</b>
<b>1.</b>	<b>INTRODUCTION</b>	<b>8</b>
	1.1 GENERAL	8
	1.2 OBJECTIVE	9
	1.3 EXISTING SYSTEM	9
	1.4 PROPOSED SYSTEM	9
<b>2.</b>	<b>LITERATURE REVIEW</b>	<b>10</b>
	2.1 GENERAL	11
<b>3.</b>	<b>SYSTEM DESIGN</b>	<b>12</b>
	3.1 GENERAL	12
	3.1.1 SYSTEM FLOW DIAGRAM	12
	3.1.2 ARCHITECTURE DIAGRAM	13
	3.1.3 SEQUENCE DIAGRAM	14
<b>4.</b>	<b>PROJECT DESCRIPTION</b>	<b>15</b>
	4.1 METHODOLOGIE	15
	4.1.1 MODULES	16
<b>5.</b>	<b>OUTPUT SCREENSHOTS</b>	<b>18</b>
	5.1. Dispatcher of excel	19
	5.2. Overview Of RE Framework	20
	5.3.Framework Of RE Excel	21
	5.4. Sample Excel Sheet	21
	5.5. Sample Offer Letter	22
<b>6.</b>	<b>CONCLUSIONS</b>	<b>23</b>
	6.1 .GENERAL	24
	<b>APPENDICES</b>	<b>25</b>
	<b>REFERENCES</b>	<b>30</b>

## **LIST OF FIGURES :**

<b>Figure No</b>	<b>Title</b>	<b>Page No.</b>
<b>3.1.1</b>	<b>System Flow Diagram</b>	<b>12</b>
<b>3.1.2</b>	<b>Architecture Diagram</b>	<b>13</b>
<b>3.1.3</b>	<b>Sequence Diagram</b>	<b>14</b>
<b>5.1</b>	<b>Dispatcher Of Excel</b>	<b>19</b>
<b>5.2</b>	<b>Overview Of RE Framework</b>	<b>20</b>
<b>5.3</b>	<b>Framework Of RE Excel</b>	<b>21</b>
<b>5.4</b>	<b>Sample Excel Sheet</b>	<b>21</b>
<b>5.5</b>	<b>Sample Offer Letter</b>	<b>22</b>

## **LIST OF ABBREVIATIONS:**

<b>Abbreviation</b>	<b>Full Form</b>
<b>SMTP</b>	<b>Simple Mail Transfer Protocol</b>
<b>ERD</b>	<b>Entity Relationship Diagram</b>
<b>DFD</b>	<b>Data Flow Diagram</b>
<b>HR</b>	<b>Human Resources</b>
<b>API</b>	<b>Application Programming Interface</b>
<b>RE</b>	<b>Robotic Enterprise</b>
<b>RPA</b>	<b>Robotics Process Automation</b>

# **CHAPTER-1**

## **INTRODUCTION**

A birthday reminder bot is a digital solution designed to help users efficiently manage and celebrate birthdays without the worry of forgetting important dates. This bot automates the process of storing, tracking, and notifying users about upcoming birthdays, ensuring timely reminders for personal, social, or professional purposes. By leveraging automation, it simplifies birthday management, allowing users to focus on thoughtful celebrations and gestures.

The bot typically collects data such as names, birth dates, and notification preferences through user input or integrations with external systems like calendars or social media. It stores this information securely and processes it to identify birthdays in advance.

Notifications are sent through various channels, such as SMS, email, push notifications, or messaging apps, based on user preferences.

Equipped with a user-friendly interface, the bot allows easy addition, editing, or deletion of birthdays. Advanced features, such as personalized message templates, gift suggestions, and calendar integration, make it a versatile tool for individuals, families, or organizations aiming to strengthen personal connections. By automating reminders, the birthday reminder bot ensures no special day is missed, fostering better relationships and creating memorable moments.



## **1.2 OBJECTIVE**

The objective of a birthday reminder bot is to automate the process of tracking and notifying users about upcoming birthdays, ensuring that no important dates are forgotten. It aims to enhance personal and professional relationships by providing timely and accurate reminders, fostering connections through thoughtful gestures. The bot seeks to minimize the manual effort involved in remembering birthdays, offering users the convenience of automated notifications. It also allows for personalized greetings and customizable reminder settings, making each notification meaningful. Additionally, the bot integrates with calendars and communication platforms to sync birthday data, ensuring reliability and seamless user experience. Ultimately, the birthday reminder bot helps users stay organized, timely, and connected, ensuring that special occasions are always remembered.

## **1.3 EXISTINGSYSTEM**

Existing systems for birthday reminders are mostly manual or basic, such as using calendars, social media, or spreadsheets. These methods are time-consuming, prone to errors, and lack personalization. Mobile apps provide reminders but often lack integration with other tools and offer limited customization. Overall, these systems fail to automate, personalize, or centralize birthday management effectively, highlighting the need for a more advanced solution like a birthday reminder bot.

## **1.4 PROPOSEDSYSTEM**

The proposed birthday reminder bot offers an automated and personalized solution for managing birthdays. It collects user data, stores it in a structured database, and sends reminders via SMS, email, or messaging apps based on user preferences. The bot integrates with calendars and social media, automates notifications, and provides a user-friendly interface for managing birthdays. With features like customizable messages and scalable usage, the system ensures timely reminders, reduces manual effort, and offers a seamless, efficient way to track and celebrate birthdays.

## **CHAPTER-2**

### **LITERATURE\_REVIEW**

A literature review of birthday reminder systems highlights various approaches, from manual methods like using calendars and spreadsheets to automated solutions such as mobile apps and social media notifications. Traditional systems, while effective, are often time-consuming, error-prone, and lack personalization. Existing automated solutions, like mobile reminder apps, provide basic functionality but typically lack integration with external tools, such as calendars or communication platforms, limiting their efficiency. Recent advancements in automation and AI have paved the way for more sophisticated systems that offer customization, seamless integration, and personalized notifications. These systems address the shortcomings of previous methods, offering a more reliable and user-friendly approach to birthday management.

## CHAPTER-3

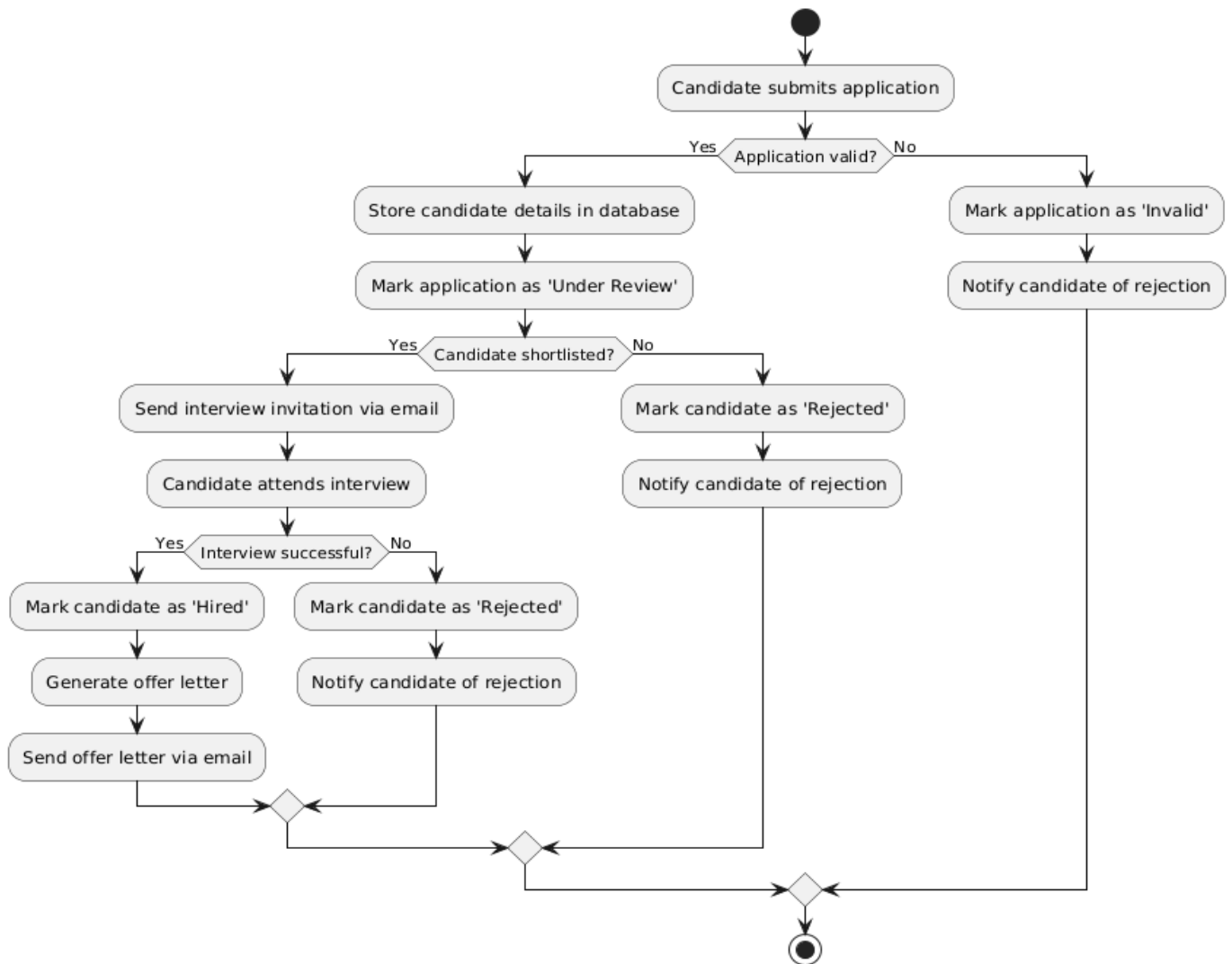
### SYSTEM DESIGN

#### 3.1.1 SYSTEM FLOW DIAGRAM

The **System Flow Diagram** outlines the overall flow of data and processes in the system. It demonstrates how user inputs, system processing, and outputs interact.

**Description:**

1. **Input** Name, Birthday, contact details (email/phone), and Reminder preferences (SMS/Email) are provided by the user through the frontend interface
2. **Process:**
  - User data is stored in a **database**.
  - The system checks daily for any upcoming birthdays (e.g., tomorrow).
  - The bot calculates which birthdays are due for reminder and prepares notifications.
3. **Output:** The bot sends birthday reminders via **SMS** or **Email** to the user based on their preferences, notifying them of upcoming birthdays.

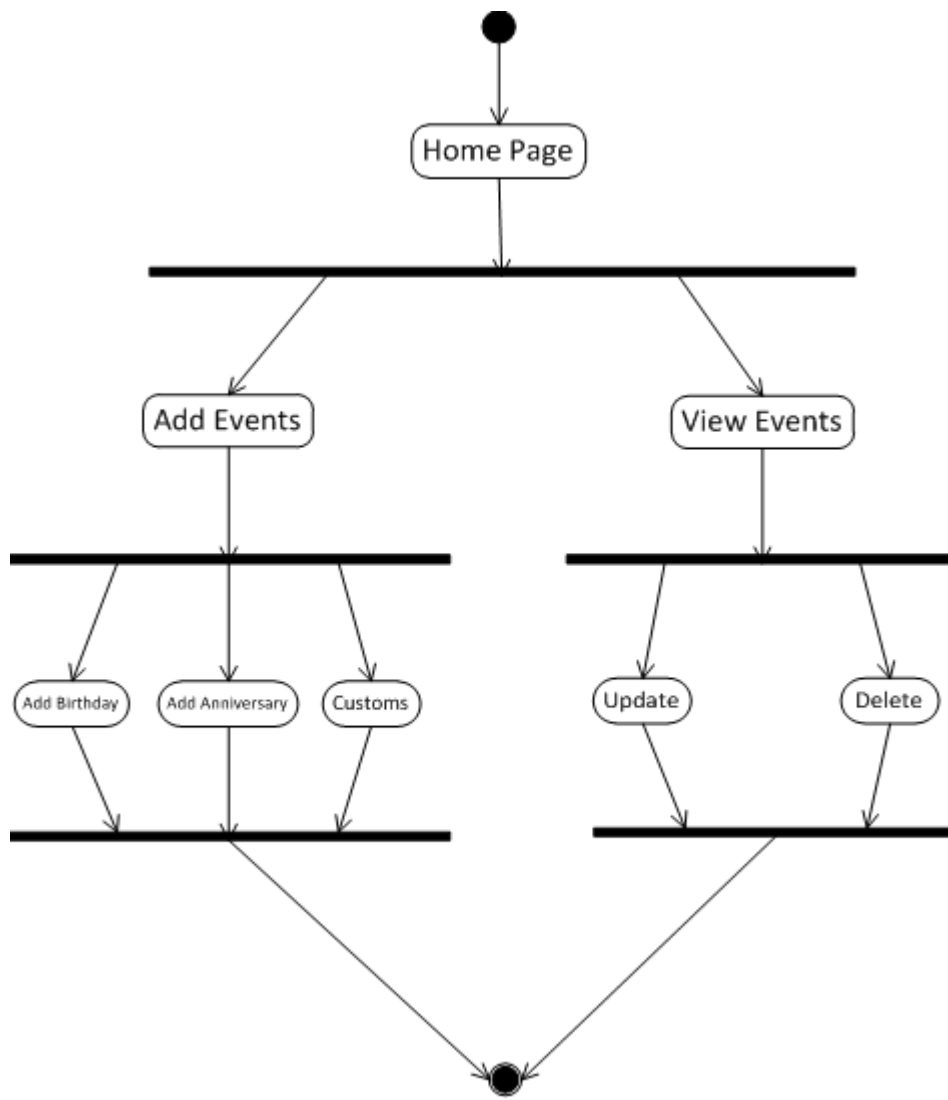


### 3.1.2 ARCHITECTURE DIAGRAM

The **Architecture Diagram** provides a high-level view of the system's structure and its components.

#### Components:

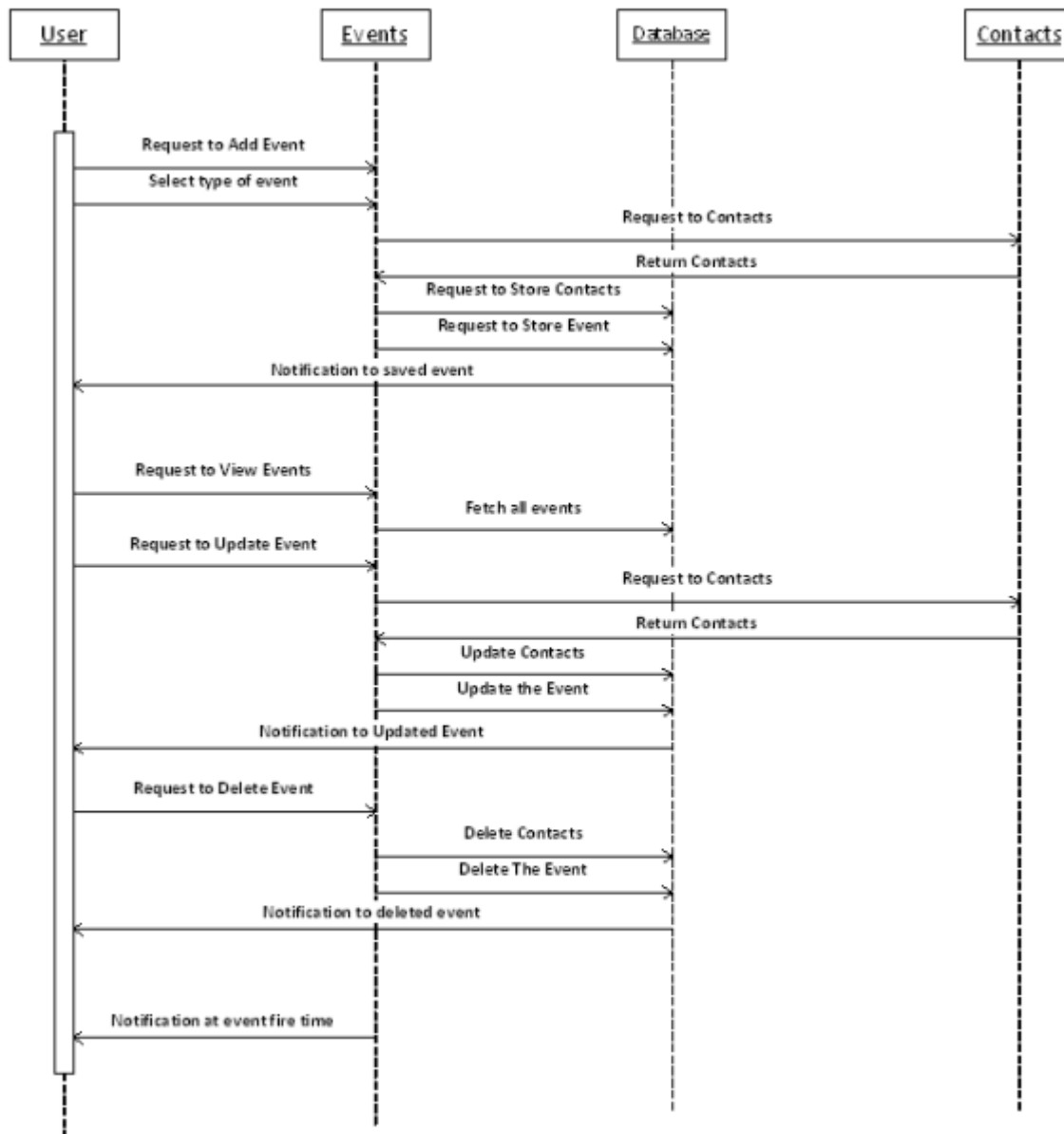
1. **Frontend:** User Interface where users input their data (name, birthday, contact details) and set reminder preferences (SMS/Email).
2. **Backend:** The server processes birthday data, calculates upcoming reminders, and triggers notifications based on user preferences
3. **Database/Storage:** Stores user information, birthdays, and contact details securely for easy retrieval by the backend.
4. **External Services:** SMS Service (e.g., Twilio): Sends birthday reminders via SMS.



### 3.1.3 SEQUENCE DIAGRAM

The **Sequence Diagram** shows the interaction between actors (HR personnel) and the system components in a sequential manner.

**Steps:**



## CHAPTER-4

### PROJECT DESCRIPTION

The **Birthday Reminder Bot** is an automated system designed to help users never forget important birthdays. By storing users' birthday information along with contact details and reminder preferences, the bot ensures timely reminders through **SMS, email, or push notifications**. The system allows users to input their personal and contact details via a **frontend interface**, which are then stored in a **database**. Daily, the **backend** checks for any upcoming birthdays, calculates when to send reminders, and triggers notifications using **external services** like **Twilio** (for SMS) or **SMTP** (for email). This automated process saves users time and effort, ensuring that they never miss an opportunity to wish a loved one on their birthday.

#### 4.1 METHODOLOGY

The development of the followed an agile methodology, ensuring iterative progress and flexibility in meeting project requirements. The system was built using UiPath's Robotic Process Automation (RPA) platform, utilizing its RE Framework to ensure structured execution, error handling, and scalability. The key steps in the methodology include the following:

1. **Requirements Gathering:** Collect user requirements for storing birthday data, reminder preferences, and notification types (SMS or Email).
2. **System Design:** Design the **frontend** for user input and interaction  
Develop the **backend** to handle business logic, data storage, and reminder calculations.



Integrate **external services** like **Twilio** for SMS and **SMTP** for sending emails.

### **Data Storage:**

Use a **database** to securely store user information, including birthdays and contact details, and ensure data is easily accessible for reminders.

3. **Testing & Deployment:** Test the system to ensure that data is correctly processed and reminders are sent on time through the selected communication channels. Deploy the system for use, ensuring it operates efficiently and accurately sends reminders based on user settings.

### **4.1.1 MODULES:**

1. **User Registration Module:** This module will automate data input collection. Users can fill out their details via a web form or an Excel sheet.

### **UiPath Activities:**

Use **Input Dialogs** or **Forms** to collect user information.

Data is stored in **Excel** or **SQL Database** using **Write Range** or **Insert Data** activities.

2. **Database Module:** This module runs daily to check for upcoming birthdays.

### **UiPath Activities:**

Use **Get Row** or **Execute Query** to fetch users' birthday data from the database.

Implement logic with **If-Else** or **Switch** statements to check if the birthday is tomorrow.

Use **Delay** activity to run this check at a scheduled time (e.g., every day at midnight).

### **Notification Module:**

Sends birthday reminders via **SMS** or **Email** based on user preference.

### **UiPath Activities:**

For **Email Notifications**, use **Send Outlook Mail** or **Send SMTP Mail** activities to send reminder emails.

For **SMS Notifications**, integrate **Twilio** with UiPath using **HTTP Request** activities to send SMS.

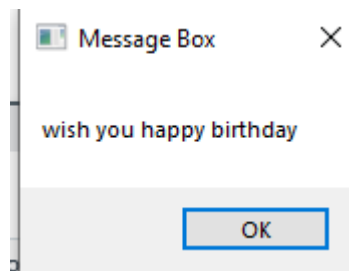
Use **If** conditions to check the user's preferred notification method (SMS or Email).

**User Interface Module:** UiPath can create a basic interface for user interaction through **Forms** or **Input Dialog**.

Use **Forms Activities** to create a simple form where users input their data.  
**Invoke Workflow** can be used to call different processes, such as registration or reminders.

## CHAPTER-5

### OUTPUT SCREENSHOT



## CHAPTER-6

### CONCLUSIONS

The **Birthday Reminder Bot** offers an efficient and automated solution to help users never forget important birthdays. By leveraging technologies like **UiPath**, **SMS services** (Twilio), and **email services** (SMTP), the bot ensures timely, personalized reminders through the user's preferred communication channel. The system consists of several integrated modules, including user registration, database management, reminder logic, and notifications, all working together to provide a seamless experience.

With this bot, users save time and effort, while also ensuring that they can stay connected with friends, family, and colleagues. The automation eliminates the risk of missing important dates, contributing to stronger relationships and a more organized personal or business life. By using an RPA solution like UiPath, the system is scalable, reliable, and easily customizable to meet future needs.

organizations to maintain compliance with internal policies and external regulations by ensuring that all offer letters are standardized and The project demonstrates how RPA can be applied to real-world business problems, making processes more efficient and allowing organizations to focus.

## APPENDICES

### Appendix A: Sample User Data Structure

This appendix provides **code snippets** for essential functionalities such as:

1. Reading data from an Excel file.
2. Generating personalized letters.
3. Sending emails via SMTP.

### Appendix B: UiPath Workflow Overview

This appendix includes a **Process Overview** generated by the UiPath, illustrating the dynamic data insertion and format customization.

#### User Registration Workflow:

**Input:** User provides name, birthday, email, phone number, and reminder preference.

**Process:** The data is validated and stored in the **database** using UiPath **Write Range** or **Database Insert** activities.

**Output:** Confirmation message or email confirming successful registration.

#### Reminder Check Workflow:

**Input:** Database query to retrieve users with upcoming birthdays.

**Process:** The workflow checks daily for birthdays and compares them with the current date.

**Output:** Triggered reminder for the next day's birthday via SMS or Email.

#### **Notification Workflow:**

**Input:** User's reminder preference (SMS/Email).

**Process:** Sends an SMS using Twilio or an email using SMTP.

**Output:** Birthday reminder sent to the user or recipient.

#### Appendix C: Twilio Integration for SMS

To integrate **Twilio** with **UiPath** for SMS reminders, follow these steps:

**Create a Twilio Account:** Obtain an **Account SID** and **Auth Token** from the Twilio Console.

**Install Twilio Package in UiPath:** Use the **Package Manager** to install the Twilio API package

**Create HTTP Request Activity:** Configure the **HTTP Request** activity in UiPath to send a POST request to Twilio's API with the required parameters (recipient phone number, message, etc.).

#### Appendix D: Email Notification Configuration (SMTP)

**SMTP Server Configuration:** Use a service like Gmail, Outlook, or any SMTP server.

### **Appendix 3: Testing Logs**

Contains a record of the **testing process**, including:

1. Test case IDs.
2. Test steps.
3. Expected vs. actual results.
4. Notes on identified issues and resolutions.