SWIGGY ORDERING BOT

A PROJECT REPORT

Submitted by

A K GOUTHAM (220701077)

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 **"SWIGGY ORDERING BOT"** is the Bonafide work of **"A K GOUTHAM (220701077)"** who carried out the project work for the subject OAI1903-Introduction to Robotic Process Automation under my supervision.

Mrs. J. Jinu Sophia(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_____.

TABLE OF CONTENTS

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 our Vice Chairma and our respected Chairperson for providing us with the requisite infrastructure and sincere endeavouring in educating us in their premier institution.

Our sincere thanks to our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to 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, Assistant Professor and 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**,** Professor**,** Associate Professor and Assistant Professor (SG), Department of Computer Science and Engineering for their useful tips during our review to build our project.

A K GOUTHAM (220701077)

ABSTRACT

In today's fast-paced world, automation has become essential to streamline everyday tasks, including online food ordering. This project leverages UiPath Robotic Process Automation (RPA) to automate the ordering process on the Swiggy platform, offering a more efficient and user-friendly experience. The workflow begins by collecting user inputs, such as the city, preferred dish, and food type (Veg or Non-Veg), using interactive dialogs. Based on these inputs, the system searches for restaurants, filters results, and identifies the desired dish.

The automation dynamically adjusts the quantity of the selected dish by iterating through a loop until the specified number is reached. Once the dish is added to the cart, the workflow moves to the checkout phase. Key UiPath activities like **Input Dialog**, **Type Into**, **Click**, and **Data Scraping** enable seamless interaction with the Swiggy website. Decision-making logic ensures the correct selection of items based on user preferences, while error-handling mechanisms address scenarios such as invalid inputs or missing data.

This project demonstrates a robust and scalable approach to automating online food ordering, reducing manual effort and improving speed and accuracy. By eliminating repetitive actions, this solution not only enhances user convenience but also showcases the potential of RPA in optimizing processes across digital platforms.

LIST OF TABLES

Field Name	Data Type	Description
Extract Data Table	DataTable	It is used to store the structured data extracted from a webpage, specifically from the Swiggy search results page. It contains the extracted menu from the page.

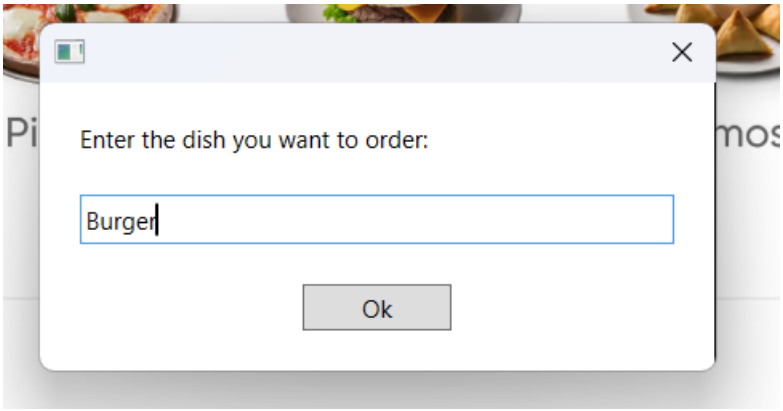
DESCRIPTION:

- To dynamically extract and organize menu details for further processing in the automation. To allow the bot to interact with specific dishes based on user preferences (e.g., Veg/Non-Veg, price range, ratings).To support decision-making steps like filtering and selecting dishes programmatically.

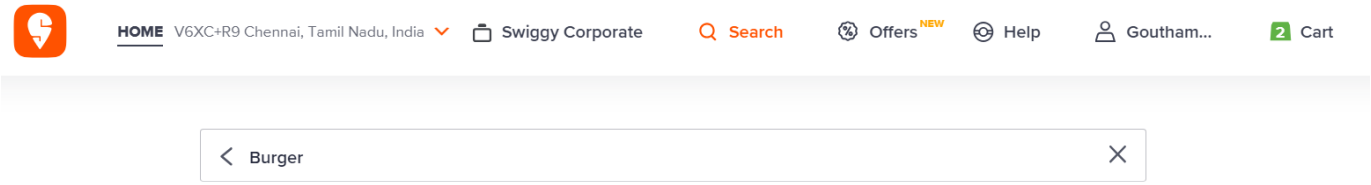
LIST OF FIGURES:

1.Process:

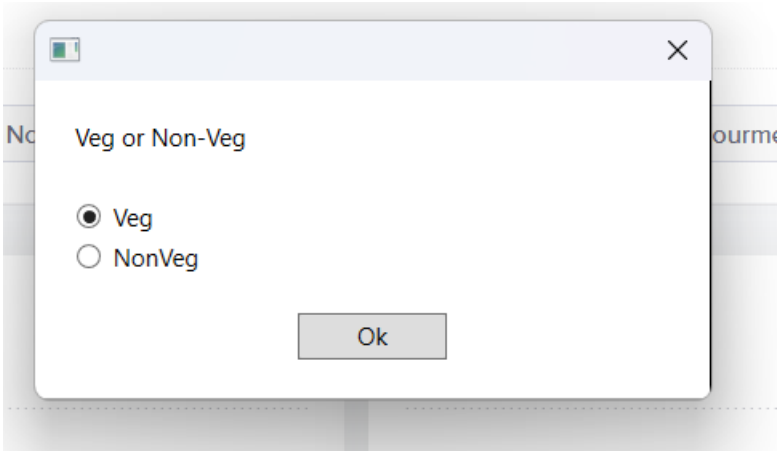
Enter the inputs:



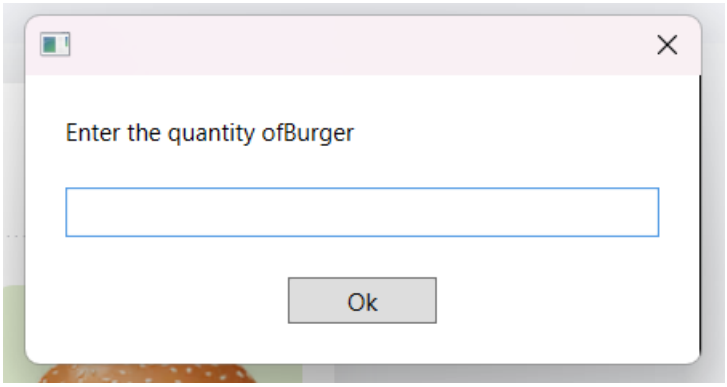
2.Searches for the order:



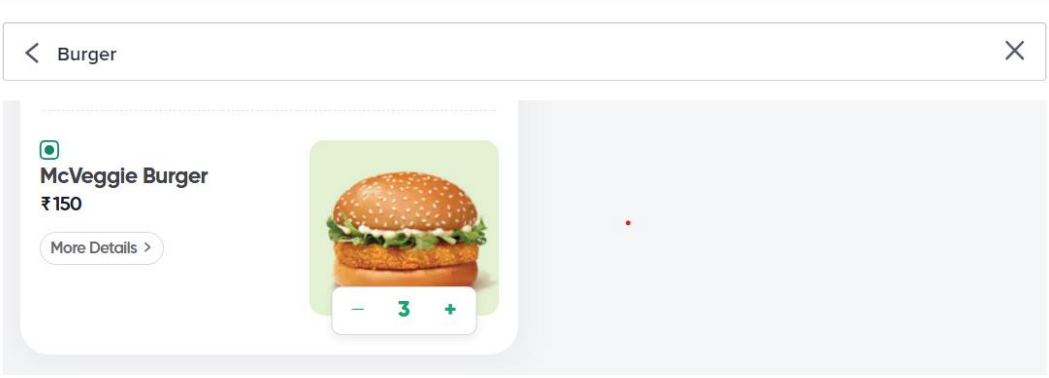
3.Choose Veg or Non-Veg:



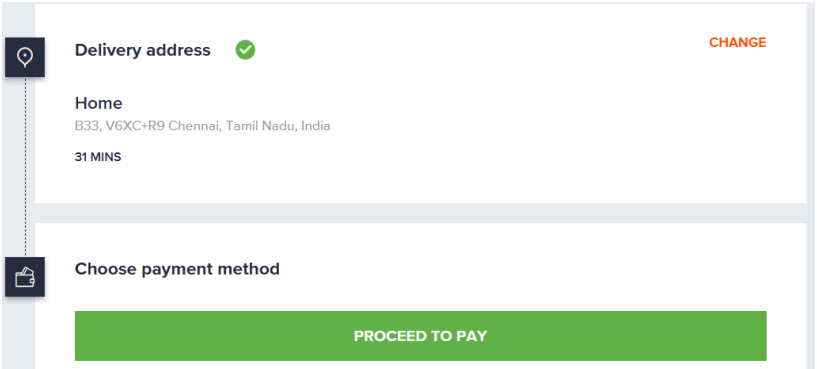
4.Enter quantity :



5.Order Added :



6.Proceed to Checkout :



INTRODUCTION

This UiPath-based RPA project automates the process of ordering food online from Swiggy, streamlining the entire workflow from dish selection to order placement. The bot begins by interacting with the user, prompting them to input the name of the dish they wish to order, along with preferences like Veg/Non-Veg and quantity. It then navigates to the Swiggy website, searches for the specified dish, and extracts structured data such as dish names, prices, and ratings from the search results. Using this information, the bot applies filters based on user preferences, selects the desired dish, and adds it to the cart in the specified quantity using iterative logic. Finally, the bot proceeds to the checkout page to finalize the order. This project demonstrates the power of RPA in automating repetitive web-based tasks, reducing manual intervention, and enhancing user convenience in real-world scenarios.

1.GENERAL

This project demonstrates the use of Robotic Process Automation (RPA) to automate a repetitive and time-consuming task—placing online food orders. The automation interacts with a food delivery platform, guiding the user through a streamlined process. It prompts the user to enter specific details, such as the dish name, preferences (e.g., Veg/Non-Veg), and quantity. Based on this input, the bot searches for the dish, extracts relevant data like names, prices, and ratings, and selects the appropriate option according to user preferences. The bot then adds the desired quantity to the cart and navigates to the checkout page to complete the order. This project showcases how RPA can be used to simplify daily tasks, improve efficiency, and minimize human effort in online transactions.

2.OBJECTIVE

This project is designed to automate the online food ordering process using Robotic Process Automation (RPA) with UiPath. The primary goal is to streamline repetitive tasks such as searching for a dish, applying filters based on user preferences, selecting the desired quantity, and placing the order. By leveraging the capabilities of RPA, this project demonstrates how routine, time-consuming activities can be handled with greater efficiency and minimal human intervention.

The automation starts with a user-friendly interaction where the bot collects essential inputs such as the dish name, preferences (e.g., Veg/Non-Veg), and quantity. The bot then navigates to the Swiggy website, searches for the dish, and extracts structured data like dish names, prices, and ratings. Based on the extracted information and user preferences, the bot selects the appropriate dish, adds the specified quantity to the cart, and proceeds to checkout to finalize the order.

1. **Automate Online Food Ordering**: Simplify the entire process, from searching for a dish to placing an order, by automating it with UiPath.
2. **Reduce Manual Effort**: Minimize the need for user intervention in repetitive web-based tasks.
3. **Enhance Efficiency**: Ensure faster and more accurate execution of tasks compared to manual handling.
4. **Demonstrate Practical RPA Implementation**: Highlight the use of UiPath for solving real-world problems like online food ordering.

3.EXISTING SYSTEM

In the existing system, ordering food online is a manual process that requires significant user effort and time. Users must navigate to a food delivery platform, search for their desired dish, apply filters like Veg/Non-Veg preferences, browse through multiple options, select the dish, specify the quantity, add it to the cart, and finally proceed to checkout. Each step involves manual interaction, such as typing, clicking, and decision-making, which can be repetitive and prone to human errors.

Key challenges of the existing system include:

1. **Time-Consuming Process**: Users must go through several steps, from searching to ordering, which can take a significant amount of time.
2. **Prone to Errors**: Manual inputs can result in mistakes, such as selecting the wrong dish or quantity.
3. **Lack of Efficiency**: Repetitive actions, such as filtering preferences or adding multiple items, require repeated user attention.
4. **No Automation**: There is no mechanism to handle repetitive tasks automatically, making the process cumbersome for users.

4.PROPOSED SOLUTION

The proposed solution aims to automate the online food ordering process using Robotic Process Automation (RPA) with UiPath. By leveraging automation, the repetitive and time-consuming steps involved in placing a food order are streamlined, reducing manual effort and improving overall efficiency. The system interacts with the user to collect inputs, performs web-based operations such as searching for dishes, applying filters, and placing orders, all without the need for constant user involvement.

Key Features of the Proposed Solution:

1. User Interaction Automation**:** The bot collects user inputs, such as the dish name, Veg/Non-Veg preference, and quantity, through interactive dialog boxes.
2. Web Automation**:** The bot navigates to the Swiggy website, searches for the specified dish, and extracts structured data like dish names, prices, and ratings from the search results.
3. Dynamic Filtering**:** Based on user preferences (e.g., Veg/Non-Veg), the bot automatically filters the search results to select the desired dish.
4. Automated Order Placement**:** The bot adds the specified quantity of the selected dish to the cart and proceeds to the checkout page, finalizing the order.

Benefits of the Proposed Solution:

- Enhanced Efficiency**:** Automating repetitive tasks improves the speed and accuracy of the process.
- User Convenience**:** By handling all intermediate steps, the bot simplifies the user experience.
- Reduced Manual Effort**:** The system requires minimal user input, freeing users from tedious tasks.
- Scalability**:** The solution can be adapted to handle multiple orders or integrate with other food delivery platforms.

LITERATURE REVIEW:

The concept of Robotic Process Automation (RPA) has gained significant traction in recent years due to its ability to mimic human actions and automate repetitive, rule-based tasks. RPA tools, such as UiPath, have been extensively used in various domains, including healthcare, banking, logistics, and e-commerce, to improve efficiency and reduce operational costs. The literature highlights the potential of RPA to enhance user experience by automating mundane activities that require minimal cognitive effort.

In the domain of online food ordering, automation has been explored to address challenges such as repetitive data entry, filtering large datasets, and navigating dynamic web pages. Studies have shown that automation can reduce human errors, save time, and improve overall task accuracy. Web scraping and data extraction techniques are frequently utilized in RPA implementations to collect and process structured data from websites dynamically.

One significant area of focus in RPA literature is user interaction. Dialog-based systems are often integrated into workflows to collect user inputs dynamically, enabling bots to execute tasks based on real-time preferences. Research also emphasizes the importance of decision-making logic, such as condition-based filtering, to personalize the automated process for better user satisfaction.

Furthermore, case studies on automation in e-commerce suggest that automating tasks like order placement, cart management, and checkout can drastically reduce the workload on users while ensuring faster task execution. The literature also points out the scalability of such solutions, allowing the same workflow to be reused across multiple platforms with minimal modifications.

GENERAL

1. Automation in Task Management Systems

Automating repetitive tasks is crucial for improving operational efficiency in modern systems. The depicted workflow demonstrates automation for processes like data entry, form interaction, and decision-based navigation. Literature highlights how such automation minimizes human intervention and errors, enabling faster task execution and resource optimization.

2. Data Extraction and Integration

A core element of automation workflows is the extraction and integration of structured data from various sources. In this workflow, data like search inputs and results are captured and processed systematically. Research emphasizes the importance of structured data formats (e.g., CSV, databases) to ensure smooth data flow between automation systems and reduce errors during transitions.

3. Dynamic Decision-Making in Automation

The workflow incorporates conditional logic to handle dynamic decision-making (e.g., selecting "Veg" or "Non-Veg" options). Literature underscores the importance of incorporating logic-based decision trees in automation to make workflows adaptable and capable of handling varied user inputs and system states effectively.

4. Enhancing User Experience through Automation

The automation process in this workflow interacts with the user inputs and system elements like web pages, search fields, and buttons. Studies highlight that user-centric automation ensures seamless interaction with systems, delivering faster and more accurate results, ultimately improving user satisfaction.

SYSTEM ARCHITECTURE

1. Input Layer

    - The system receives inputs through dialog boxes (e.g., "Enter the dish you want to order" and "Select Veg/Non-Veg").
    - Inputs are used to dynamically control the subsequent steps in the automation process.

2. Process Layer

- Web Browser Interaction:
    - Uses an RPA tool (e.g., UiPath) to interact with a web browser.
    - Includes actions like opening the browser, typing into search fields, clicking buttons, and extracting data.
- Data Processing:
    - Captures and processes data extracted from the website (e.g., search results, product details).
    - Implements conditional logic for decision-making (e.g., choosing between "Veg" and "Non-Veg").

3. Data Layer

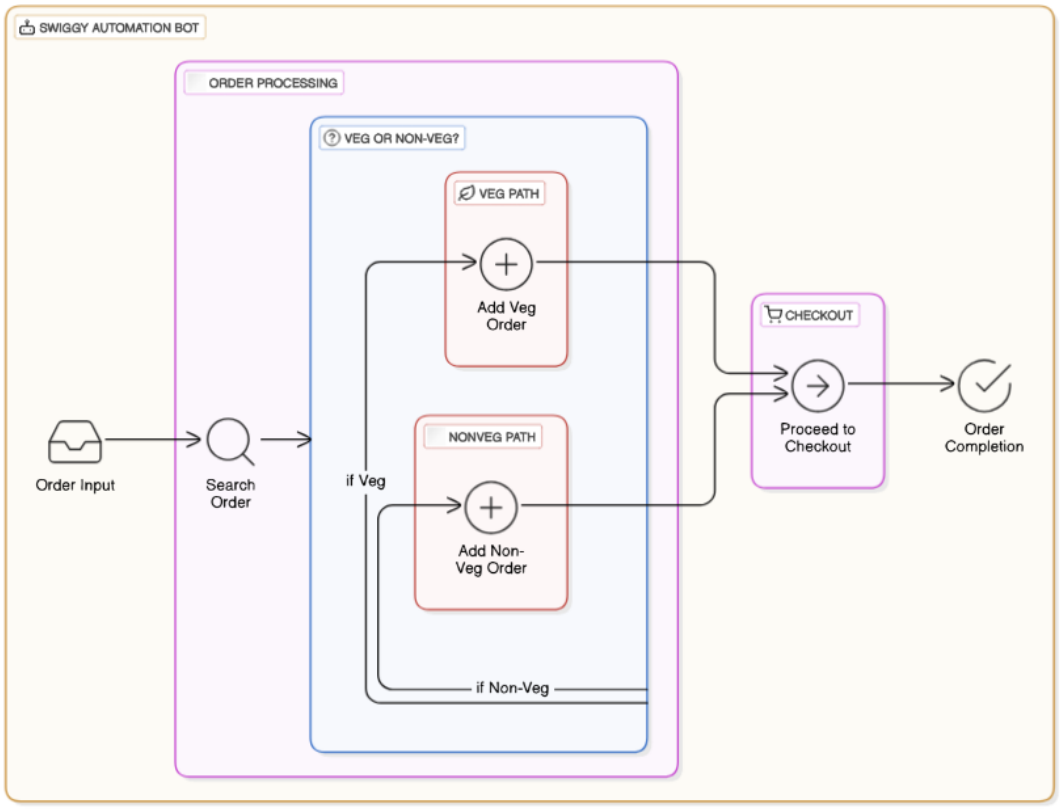
    - Variables are used to store and update dynamic values (e.g., counters for cart items).
    - Extracted data is temporarily stored for processing or passed as parameters between activities.

4. Output Layer

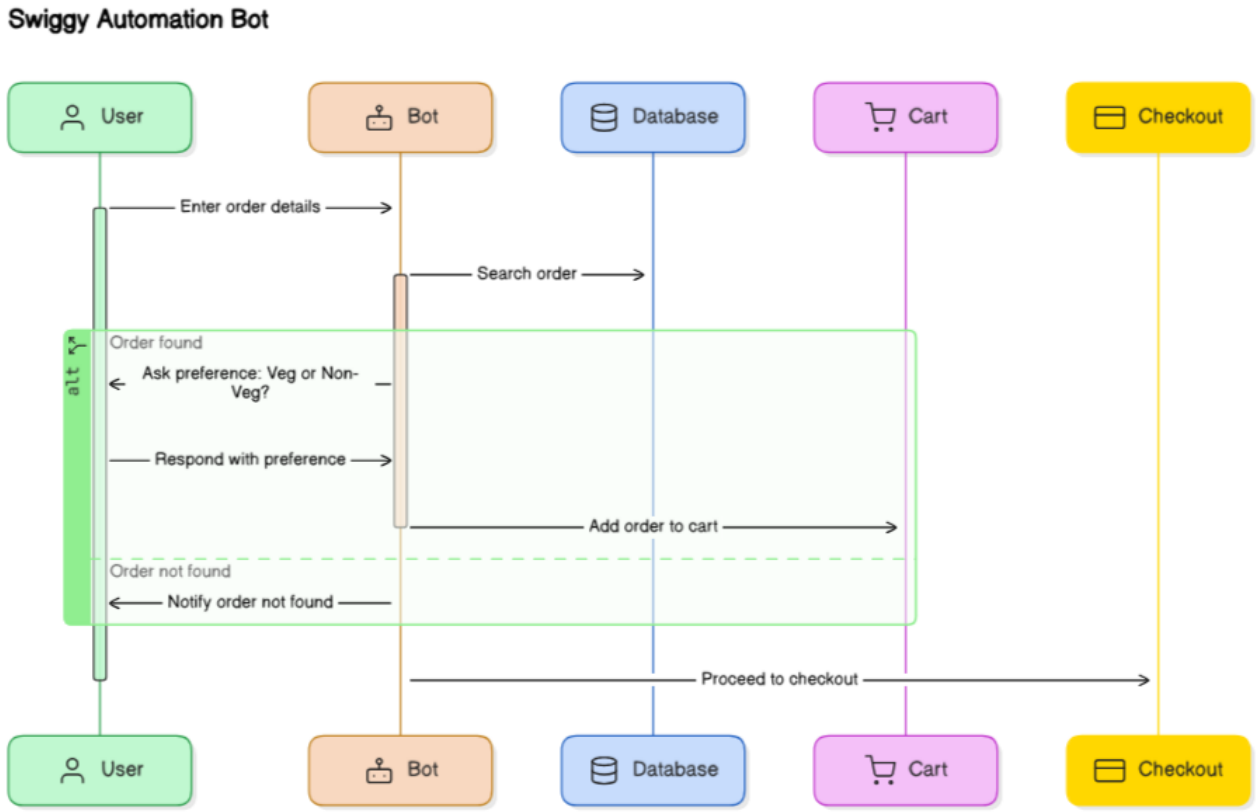
    - Executes final actions like navigating to the cart or confirming orders.
    - Automates end-to-end processes, such as navigating pages, adding items, and completing actions like placing order

SYSTEM FLOW DIAGRAM:

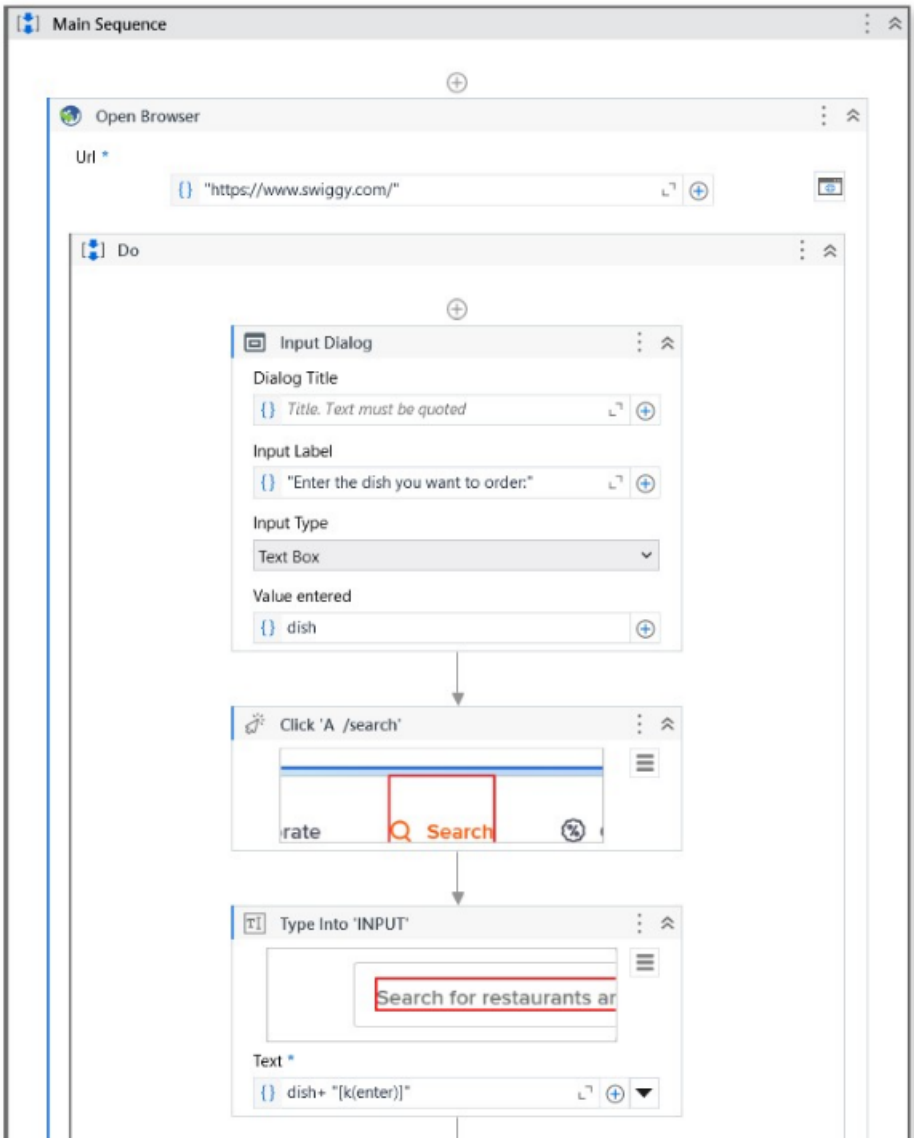
ARCHITECTURE DIAGRAM :

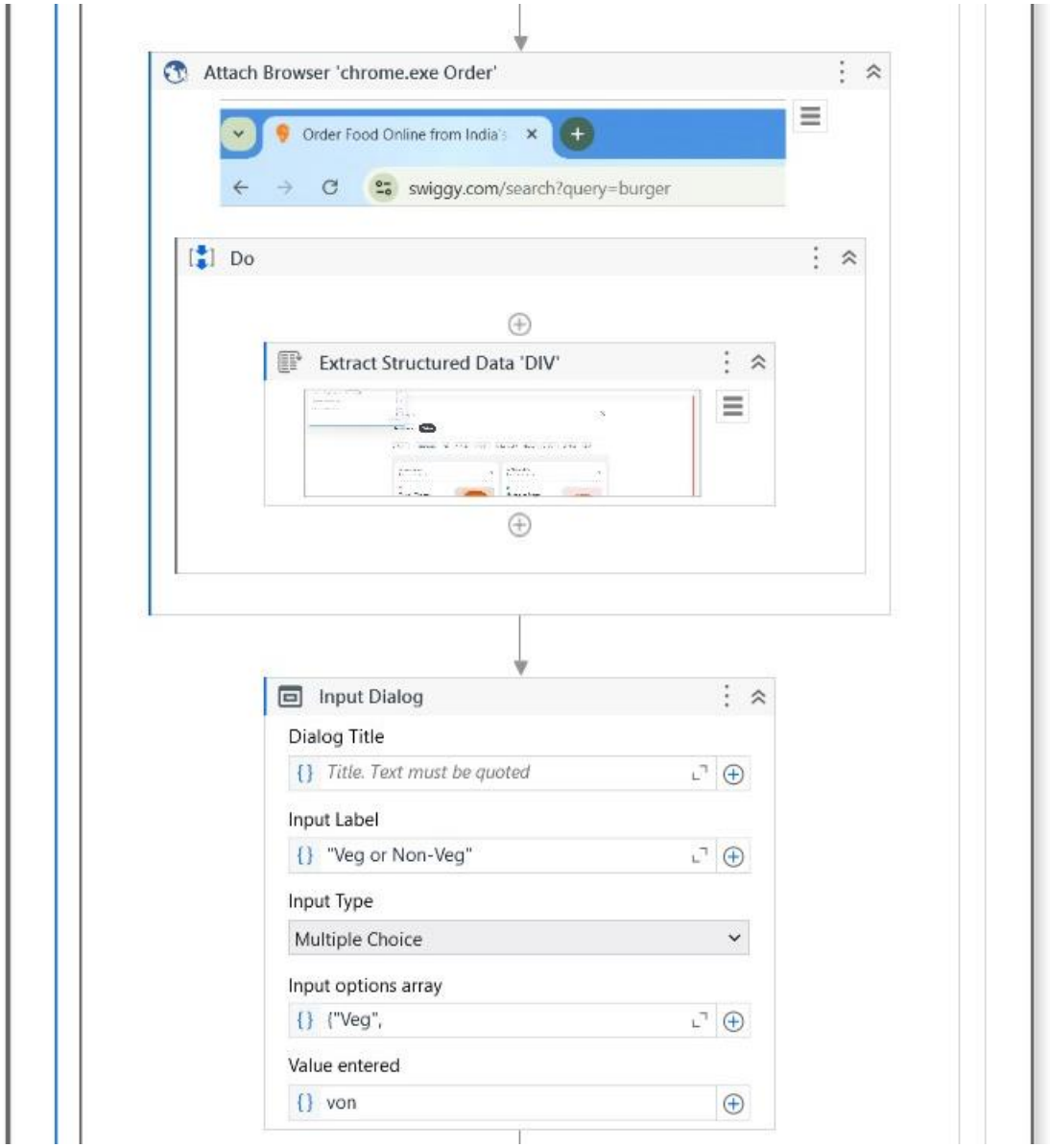


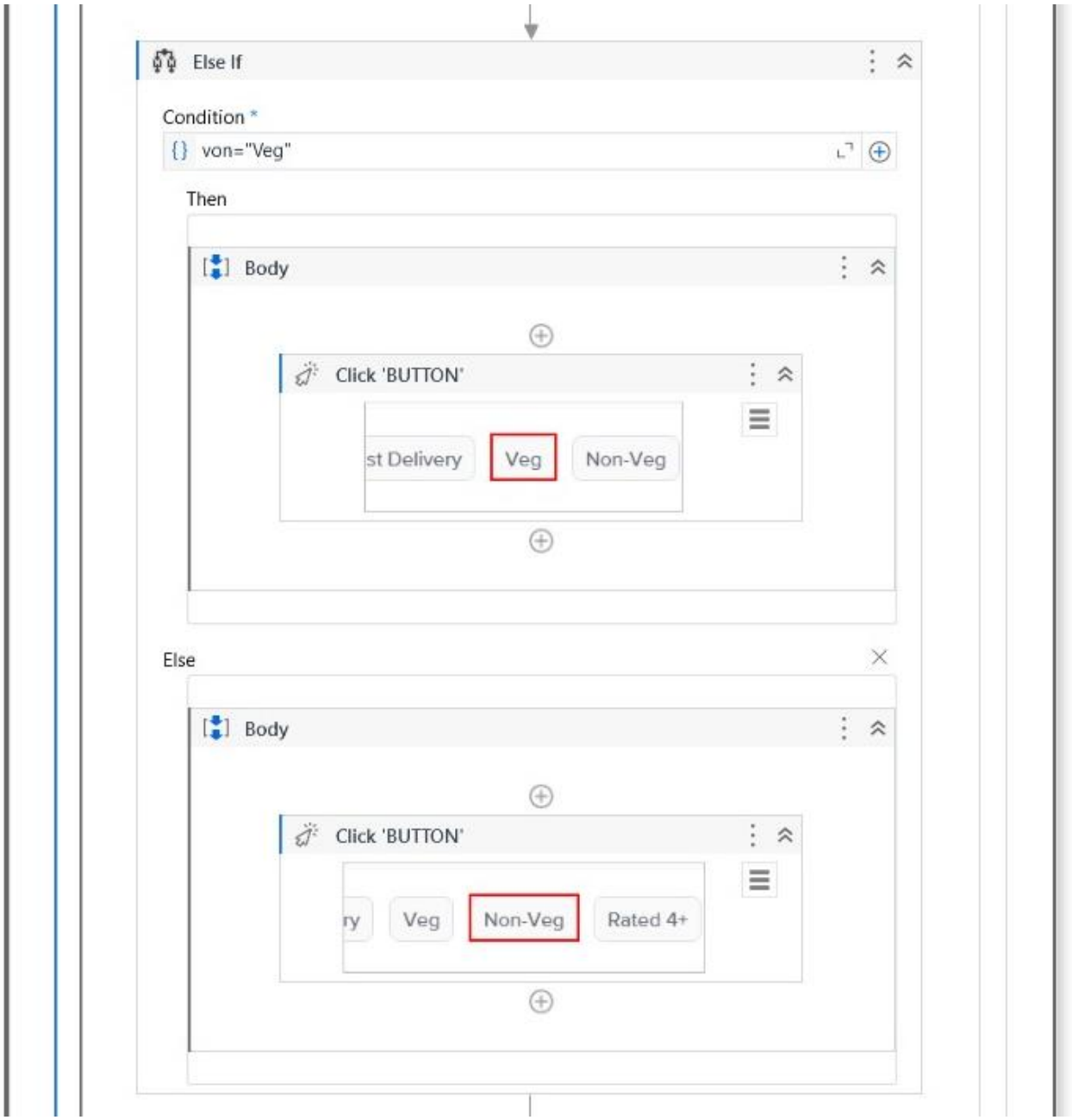
SEQUENCE DIAGRAM :

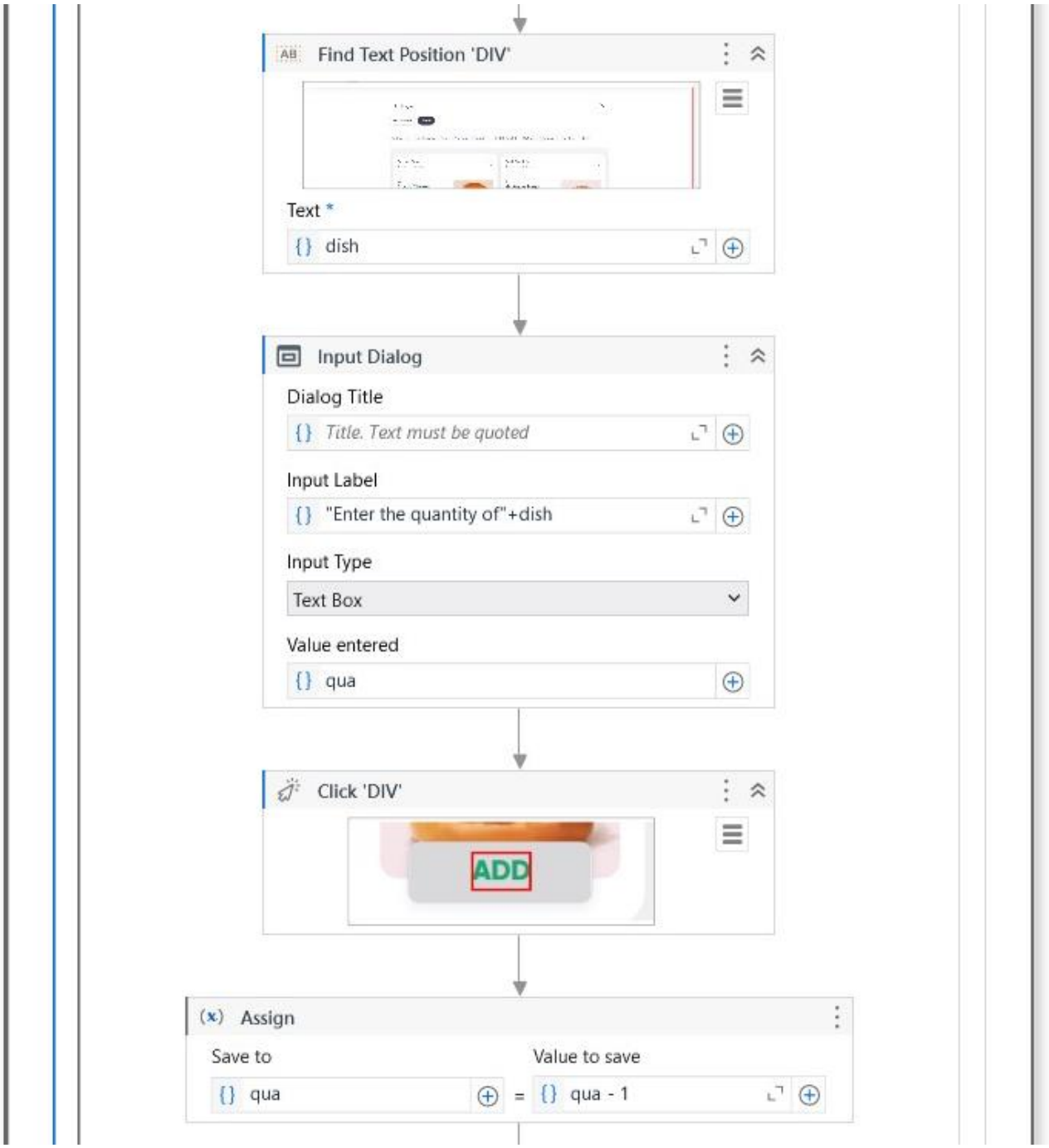


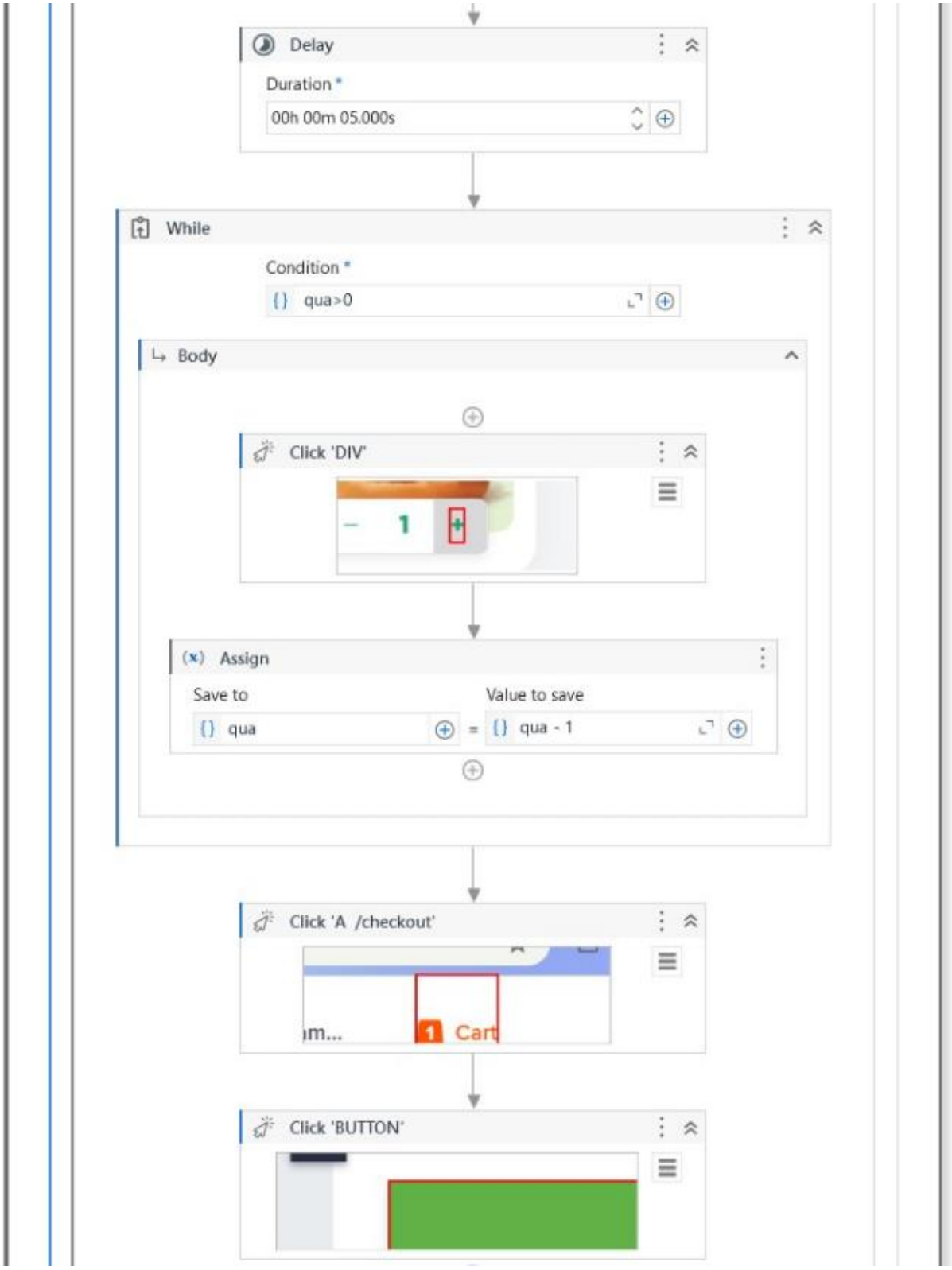
WORKFLOW :


Main Sequence
Open Browser
Url *
"https://www.swiggy.com/"
Do
Input Dialog
Dialog Title
Title. Text must be quoted
Input Label
"Enter the dish you want to order:"
Input Type
Text Box
Value entered
dish
Click 'A /search'
rate
Search
Type Into 'INPUT'
Search for restaurants ar
Text *
dish+ "[k(enter)]"


Attach Browser 'chrome.exe Order'
Order Food Online from India
swiggy.com/search?query=burger
Do
Extract Structured Data 'DIV'
Input Dialog
Dialog Title
Title. Text must be quoted
Input Label
"Veg or Non-Veg"
Input Type
Multiple Choice
Input options array
{"Veg",
Value entered
von


Else If
Condition *
von="Veg"
Then
Body
Click 'BUTTON'
st Delivery
Veg
Non-Veg
Else
Body
Click 'BUTTON'
ry
Veg
Non-Veg
Rated 4+


Find Text Position 'DIV'
Text *
dish
Input Dialog
Dialog Title
Title. Text must be quoted
Input Label
"Enter the quantity of"+dish
Input Type
Text Box
Value entered
qua
Click 'DIV'
ADD
Assign
Save to
Value to save
qua
=
qua - 1


Delay
Duration *
00h 00m 05.000s
While
Condition *
{} qua>0
Body
Click 'DIV'
1
Assign
Save to
Value to save
{} qua
=
{} qua - 1
Click 'A /checkout'
1 Cart
Click 'BUTTON'

PROJECT DESCRIPTION

METHODOLOGIES

1. Requirement Analysis

- Objective: Define the scope of automation and understand the user requirements.
- Identify the key tasks to automate:
    - Search for a dish.
    - Select preferences (e.g., Veg or Non-Veg).
    - Add the desired quantity of items to the cart.
    - Navigate to the cart for order finalization.

2. Workflow Design

- Tools and Techniques:
    - Use UiPath Studio for designing the automation workflow.
    - Modular design ensures scalability and easy debugging.
- Key Components of the Workflow:
    - Input Collection: Dialog boxes collect user inputs like dish names and preferences.
    - Web Automation: Interact with the website for searching, clicking, and extracting data.
    - Conditional Logic: Implement If-Else and While loops to handle decisions like item selection or quantity updates.

3. Implementation

- Browser Automation:
    - Open the browser and navigate to the food delivery website.
    - Use UI selectors to identify elements like search fields, buttons, and menu items.
    - Automate interactions like typing, clicking, and extracting text.
- Dynamic Decision-Making:
    - Conditional statements handle variations in user preferences (e.g., Veg/Non-Veg selection).
    - Automate the addition of items based on the user's quantity input.
- Data Handling:
    - Variables and counters store intermediate values (e.g., cart item count).

4. Testing and Debugging

- Objective: Ensure the workflow handles edge cases and functions as expected.
- Test cases include:
    - Invalid inputs (e.g., non-existent dishes).
    - UI element changes (e.g., website layout modifications).
    - Handling timeouts or errors during browser interactions.

5. Deployment and Integration

- Deploy the workflow on a production environment or integrate it with other systems if required.

6. Monitoring and Maintenance

- Monitor the automation's performance and success rate.
- Update the workflow as needed.

CONCLUSION:

The project successfully demonstrates the potential of Robotic Process Automation (RPA) to streamline repetitive and time-consuming tasks, such as ordering items from a food delivery platform. By automating processes like searching for dishes, selecting preferences, and adding items to the cart, the workflow reduces human effort, minimizes errors, and enhances operational efficiency. The dynamic decision-making capabilities, such as Veg/Non-Veg selection, and the ability to handle user inputs make the automation adaptable and user-friendly.

This automation highlights the broader applicability of RPA in similar scenarios, such as e-commerce, inventory management, and customer service, where repetitive tasks are prevalent. With proper maintenance and scalability, this solution can evolve to integrate advanced features, such as machine learning for personalized recommendations or API integrations for seamless operations.

REFERENCES :

UiPath Official Website

- Comprehensive guides, tutorials, and community forums.
- https://www.uipath.com/resources

Automation Anywhere Blog

- Articles on automation trends, best practices, and use cases.
- https://www.automationanywhere.com/blog

Blue Prism Learning Resources

- Resources to explore RPA concepts and implementations.
- https://www.blueprism.com/resources/

Stack Overflow

- A go-to platform for troubleshooting RPA workflows and code-related issues.
- https://stackoverflow.com/

GitHub RPA Repositories

- Open-source RPA scripts, projects, and tools.
- https://github.com/topics/rpa

APPENDICES :

Main.xaml

- The primary workflow file that manages the overall automation process, including browser interactions, user inputs, decision-making logic, and order placement.

Input Folder:

- DishList.xlsx:

  An Excel file containing columns like:
  - Serial Number
  - Dish Name
  - Cuisine Type (Veg/Non-Veg)
  - Price
  - Ratings

Config File

- A JSON or Excel file that stores key project settings such as:
  - Path to input Excel files.
  - Default Veg/Non-Veg preference.
  - Number of retries for failed activities.
  - Timeout durations for selectors.