Bharati Vidyapeeth’s

Institute of Management & Information Technology

C.B.D. Belapur, Navi Mumbai 400614

Vision:
Providing high quality, innovative and value-based education in information technology to build competent professionals.

Mission:

M1. To provide students for career in the industry and pursue research in all branches of computing field through effective teaching learning process.

M2. To strengthen the industry institute interaction by providing up-to-data programs.

M3. To embibe amongst the students ethical usage of technical knowledge beneficial to society.

M4. To provide an environment that fosters a framework for promoting collaborative and multidisciplinary activities.

CERTIFICATE

This is to certify that the journal is the work of Mr./Ms. **Vivek.Pokharankar** Roll No. **47** of **MCA** (**Sem - II** Div: **A**) for the academic year 2023 - 2024

Subject Solution:: **MCALE233**

Subject Name: **Robotic Process Automation**

Subject-in-charge

Principal

Date: ______________

External Examiner

Date: ______________

Bharati Vidyapeeth's Institute of Management & Information Technology
MCA Semester II AY 2023-24
MCALE233: Robotic Process Automation
INDEX
Name: Vivek Pokharankar MCA SEM II Division : Batch: A1 (Thursday)

CO1 Define the key concepts of Robotic Process Automation and evolution.
CO2 Demonstrate development of BOT with specific tools.
CO3 Apply RPA commands to automate tasks.
CO4 Designing BOT for Cognitive Analytics and Workforce Analytics.

Sr No.	Date	Topic	CO	CO-PO MAP	Timely Submission(5)	Completeness(10)	Understanding(10)	Total(25)	Sign
1		**Introduction to GUI of Tool**	**CO1**	**PO1**					
1.1	16.2.24	Introduction to Automation Anywhere A2019							
1.2	16.2.24	Actions in Automation Anywhere A2019							
2		**Automation anywhere Basic Commands:**	**CO1,CO2, CO3**	**PO1,PO2,PO3,PO5,PSO1,PSO2**					
2.1	22.2.24	Launching a website (Facebook)							
2.2	22.3.24	BoT to work with MSPaint							
2.3	29.2.24	Automate action of getting the title of active window							
2.4	7.3.24	Automate action of closing a notepad window							
3		**Automation anywhere Advanced Commands:**	**CO1,CO2, CO3**	**PO1,PO2,PO3,PO5,PSO1,PSO2**					

3.1	18.4.24	To automate the task of extracting the data from multiple PDF documents and storing the data into a text file.							
3.2	18.4.24	To automate the task of extracting the data from multiple PDF documents and storing the data into a csv file.							
3.3	25.3.24	Automate task of replacing few characters from a string							
3.4	18.4.24	Automate task of copying files from a source folder to destination folder							
3.5	2.5.24	Extract a table from webpage							
3.6	21.3.24	BoT to extract text from notepad file and copying it into excel							
3.7	14.3.24	Extracting Text from Image							
4		**RPA for Excel Commands:**	**CO1,CO2, CO3**	**PO1,PO2,PO3,PO5,PSO1,PSO2**					
4.1	4.4.24	To automate the task of extracting the data from an Excel File according to some condition and storing the extracted data in another File.							
5		**Advanced RPA Techniques:**	**CO1,CO2, CO3**	**PO1,PO2,PO3,PO5,PSO1,PSO2**					

5.1	25.4.24	Design IQ BOT	CO1,CO2,CO3,C04	PO1,PO2,PO3,PO5,PSO1,PSO2					
5.2	7.3.24	Error Handling	CO1,CO2,CO3,C04	PO1,PO2,PO3,PO5,PSO1,PSO2					
6	2.5.24	**Use of UI Path Tool**	**CO1,CO2,CO3,C04**	**PO1,PO2,PO3,PO5,PSO1,PSO2**					

SEMESTER II

Robotic Process Automation

Course Code: MCALE23

Practical 1.1

Topic : Introduction to Automation Anywhere A2019

Define RPA:

Robotic Process Automation is entity which mimics human actions with sequence of steps automatically i.e. without human intervention.

Why to use RPA:

1. Customer Satisfaction – reduce manual errors and faster services
2. Analytics – improve data quality and increase scope of data collection
3. Compliance – reduce human contact with sensitive data and reduce fraud.

Benefits of RPA:

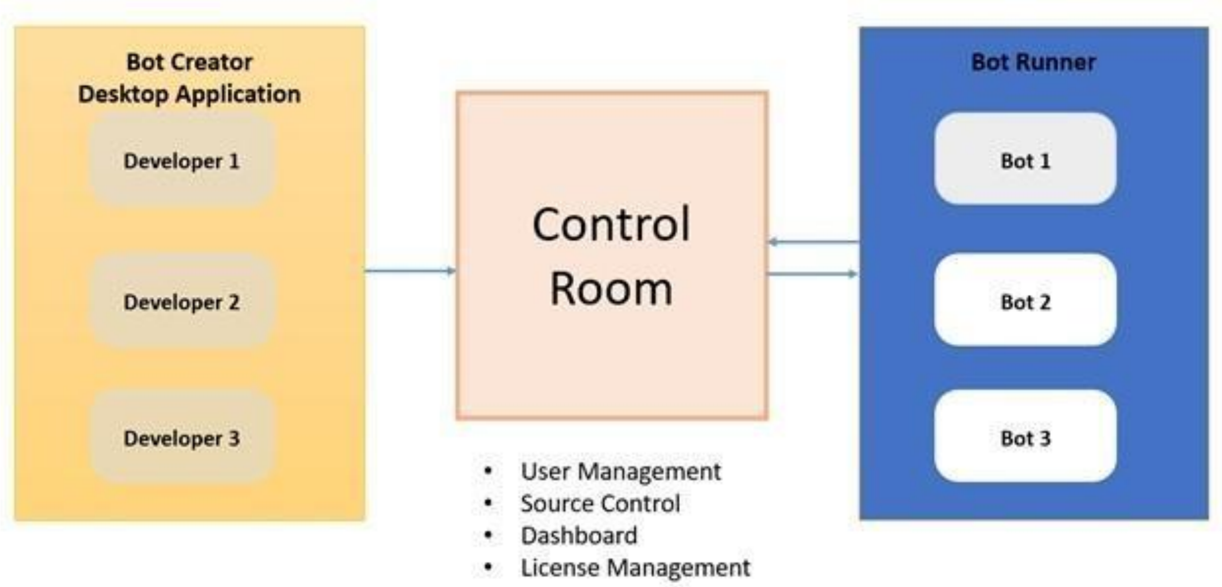
1. Reduce Burden on IT: Does not disturb underlying legacy systems
2. Reliability: Bots can work 24*7 effectively
3. Cost Cutting Technology: Reduces the size of the manual workforce and hence reduce costs
4. No Coding Required: Programming skills are not necessary
5. Accurate: Less prone to errors and functions with accuracy and uniformity.
6. Productivity Rate: Execution time much faster than manual process approach
7. Compliance: Follows rules to provide audit free trail
8. Consistency: Repetitive tasks are performed in the same manner each time

What can we automate using Robotic Process Automation?

- Excel HTML
- JAVA
- ERP
- Email
- PDF
- Web App
- Desktop App
- Image

Automation Anywhere Architecture:

- Client Server Architecture
- Bot Creator is the Developer Client
- Bot Runner is the runner client
- Control Room is the server



Create your First Bot:

Steps to follow

1. Open Community Edition Account: https://www.automationanywhere.com/products/enterprise/community-edition
2. Login to Control Room
3. Install BotAgent and connect device

Bot: **Display Hello Message**

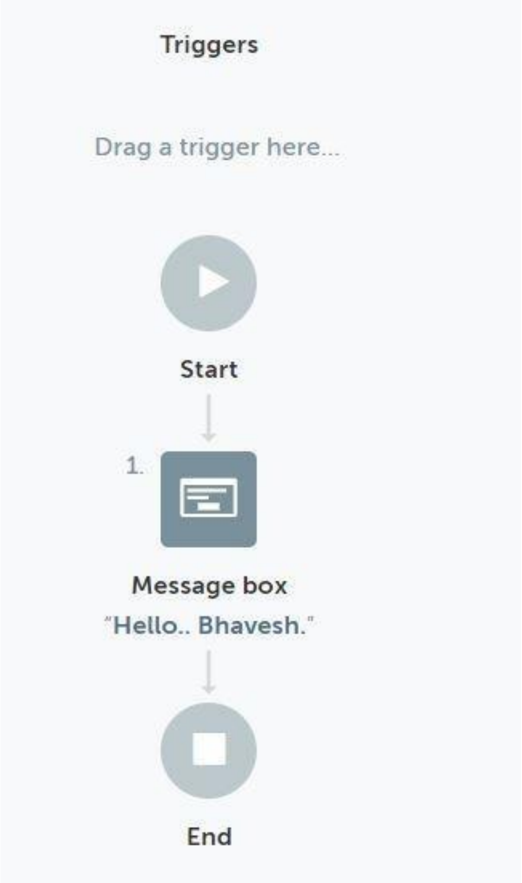
Step 1: Add Message box action and Enter the message: “Hello..!!!”

Bot Name: HelloMessagebot

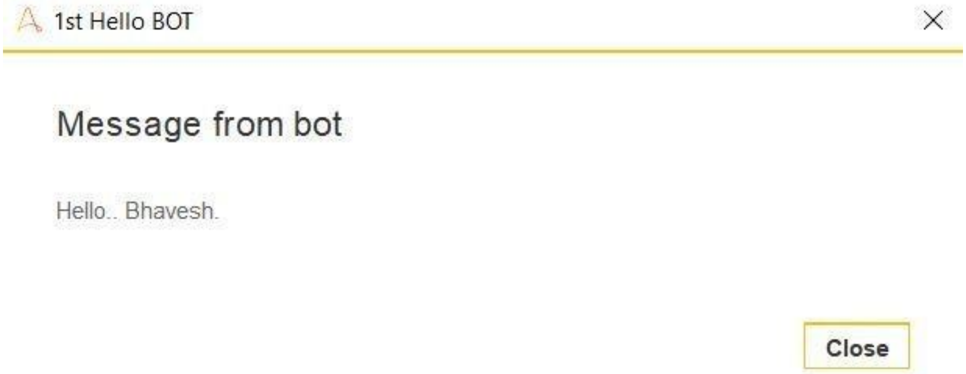
Recorder:

- Can works on Web and desktop application
- Recorder is object oriented as it captures object controls such as Buttons, textboxes etc.
- Captures Keystrokes and mouse click

Flow



Output:



Message from bot

Hello.. Bhavesh.

MCA

SEMESTER II

Robotic Process Automation

Course Code: MCALE23

Practical 1.2

Topic : Actions in Automation Anywhere A2019

Variable:

It is the containers where you can store any type of data e.g. string, numbers etc.

Types of Variable:

1. User (Local) Variables
2. System Variables
3. Credential Variables

User Variable:

Used for a specific task or set of tasks
A user variable can hold one value or multiple values

System Variables:

In the System variables, there are pre-defined variables that are provided by Automation Anywhere. System variables are available to be used in all automation tasks. System Variable types include the following components as follows.

- **Date and Time System Variables** –
  System-related date and time variables.
- **System Variables** –
  Loop: Useful in conjunction with Loop commands.
- **Excel System Variables** –
  Useful for Excel automation.
- **Email System Variables** –
  Useful for Email automation.
- **Trigger Variables** –
  Useful in conjunction with triggers.
- **PDF System Variables** –
  Useful to be used with PDF command.
- **System Variables** –
  Specific to System Settings/Parameters Variables specific to a particular client machine.

Credential Variables:

Credential variables store sensitive information that's used when running automation tasks. Credential variables can be used only by the Bot Creator and only in Command fields that need a credential input.

Actions:

These are essential building blocks of a Bot. They are the commands that tell a Bot what to do.

Actions are grouped into packages.

For example, the **File package** contains File related actions, such as Create, Open, Delete, Print Multiple files and Rename that you can insert into the Bot editor to automate a process working with files.

Excel advanced package contains Excel-related actions, such as **Open**, **Go to cell**, and **Delete cells**, that you can insert in to the Bot editor to automate a spreadsheet process.

String Operation:

Methods to Manipulate and Extract Strings

- **Assign** – assigns a value or a variable value to another string variable
- **Extract Text** –
  Use the before, before and/or after, or the after settings to extract a sub-string from a body of text.
  - Before: You provide the text that precedes the text you're interested in extracting, and the action will return everything that comes after.
  - Before and After: You provide the text that precedes the text you're interested in as well as the text which comes after the text you're interested in. The action will return all the text that comes between the two.
  - Before or After: You provide the text that may precede the text you're interested in as well as the text which may come after the text you're interested in. The action will attempt to return the preceding/subsequent text.
    - Note: Be sure to do lots of testing when using Before or After so you know exactly what kind of text the bot should expect as a return
  - After: You provide the text that comes after the text you're interested in extracting, and the action will return everything that comes before.

- **Compare** –
  Compares two strings and returns True if the strings are identical.
- **Uppercase** –
  Converts the source string to upper case.
- **Length**
  Gets the length of a string.
- **Split** –
  Splits the source string into multiple strings using a delimiter.

- **To Boolean**
  Splits the source string into multiple strings using a delimiter.
- **Substring** –
  Extract part of a larger body of text based on a start position and optional length. The optional length lets you specify how many characters you intend to extract; the alternative is that text from the starting position through the end of the provided source string would be extracted.
- **Find** –
  - returns the position of matching text within a larger body of text.
  - Can be hard-coded text, or could be a regular expression.
- **Replace** -
  Splits the source string into multiple strings using a delimiter.
- **Lowercase –**
  Splits the source string into multiple strings using a delimiter.
- **Reverse** –
  Splits the source string into multiple strings using a delimiter.
- **Trim** –
  Splits the source string into multiple strings using a delimiter.
- **To Number –**
  Splits the source string into multiple strings using a delimiter.

If statement:
Conditional Statement which executes a task if the condition is either true or false

Conditions for if Statement:
Number/String/Boolean
Windows exists
Web Control Exists
Task/Script Exists
File/Folder Exists
Data Table
Date Time

Bot: **To open calculator if it isn't open**

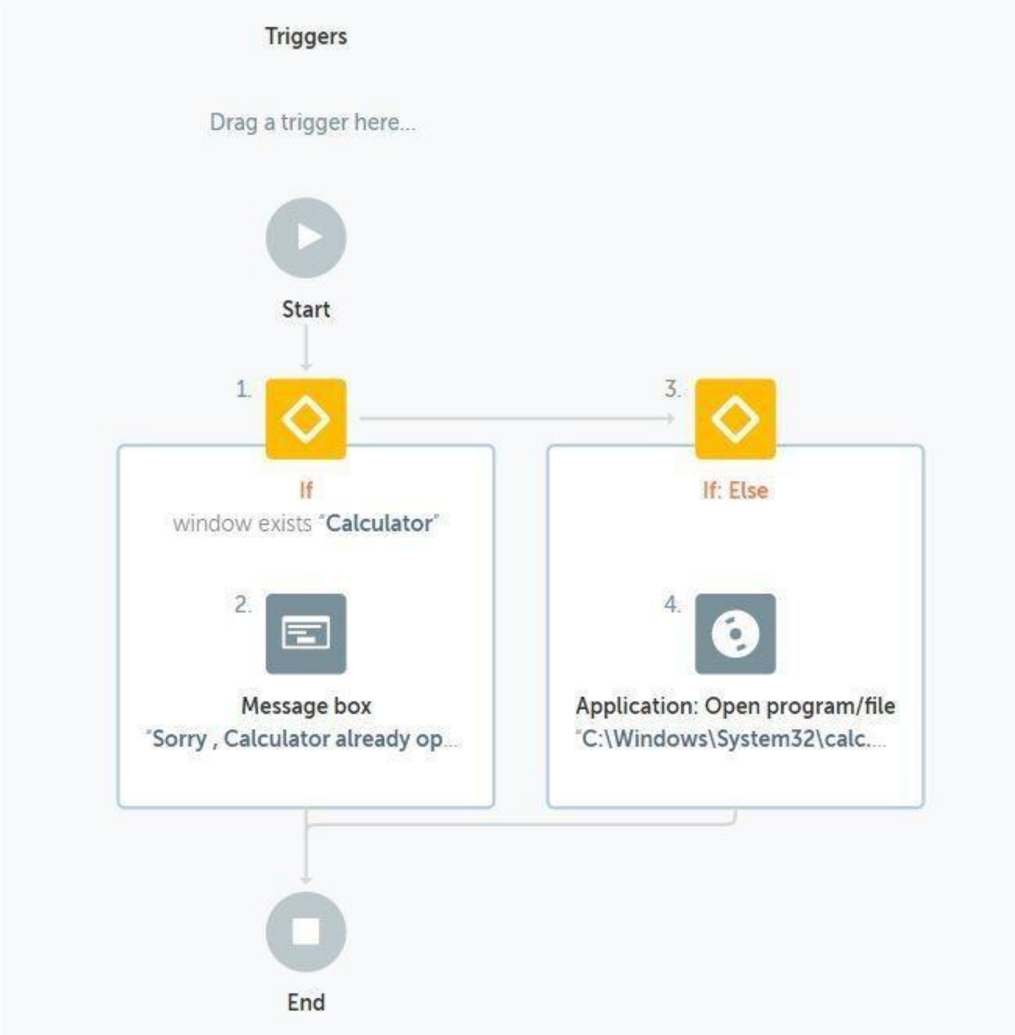
Step 1: Check if the Calculator is open. In **If action** select the condition **Window exists.** Select the calculator application from the Windows option

a. Display **Message** "Calculator already open"

Step 2: If not open the calculator. Add **Else** action and have **Application:open** action to open the calculator application
Step 3: Close the program. Close the appilication

Bot Name: If_Statement
If :

Else :

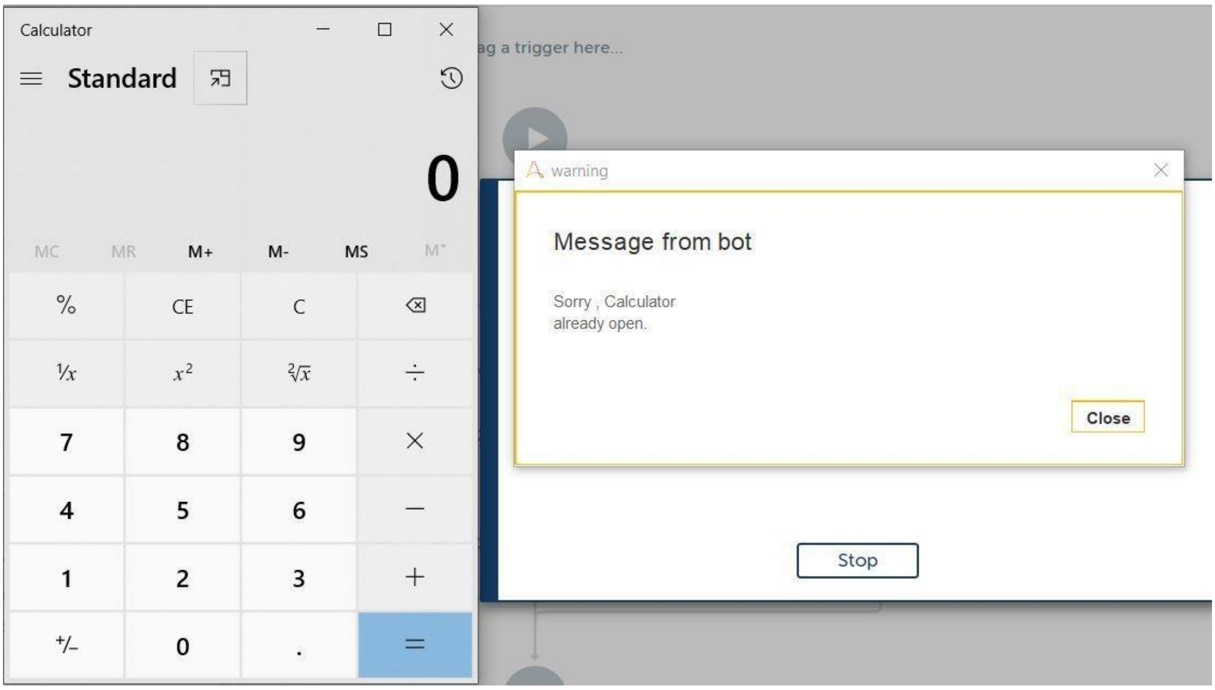


1. If
window exists "Calculator"

2. Message box
"Sorry , Calculator already op...

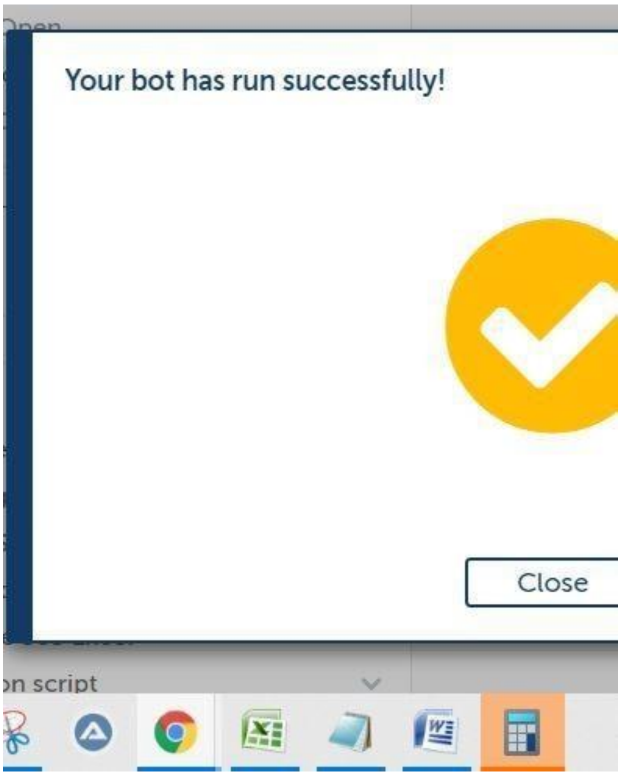
3. If: Else

4. Application: Open program/file
"C:\Windows\System32\calc....



Message from bot

Sorry , Calculator
already open.



Your bot has run successfully!

SEMESTER II
Robotic Process Automation Course Code: MCALE23
Practical 2.1

Bot : Launching a website (Facebook)

Step 1: Open the website using Browser: Open action

a) Select the Browser tab

b) Provide Link to open https://www.facebook.com/

Step 2: Using Recorder: Capture to simulate an object's click event

a) Using Capture: Object to capture the input box to provide the email id

b) Select Action as Set text

c) Enter Keystrokes with the email id with which you wish to log in

Step 3: Using Recorder: Capture to simulate an object's click event

a) Using Capture: Object to capture the input box to provide the password

b) Select Action as Set text

c) Enter Keystrokes with the password with which you wish to log in

Step 4: Using Recorder: Capture to simulate an object's click event

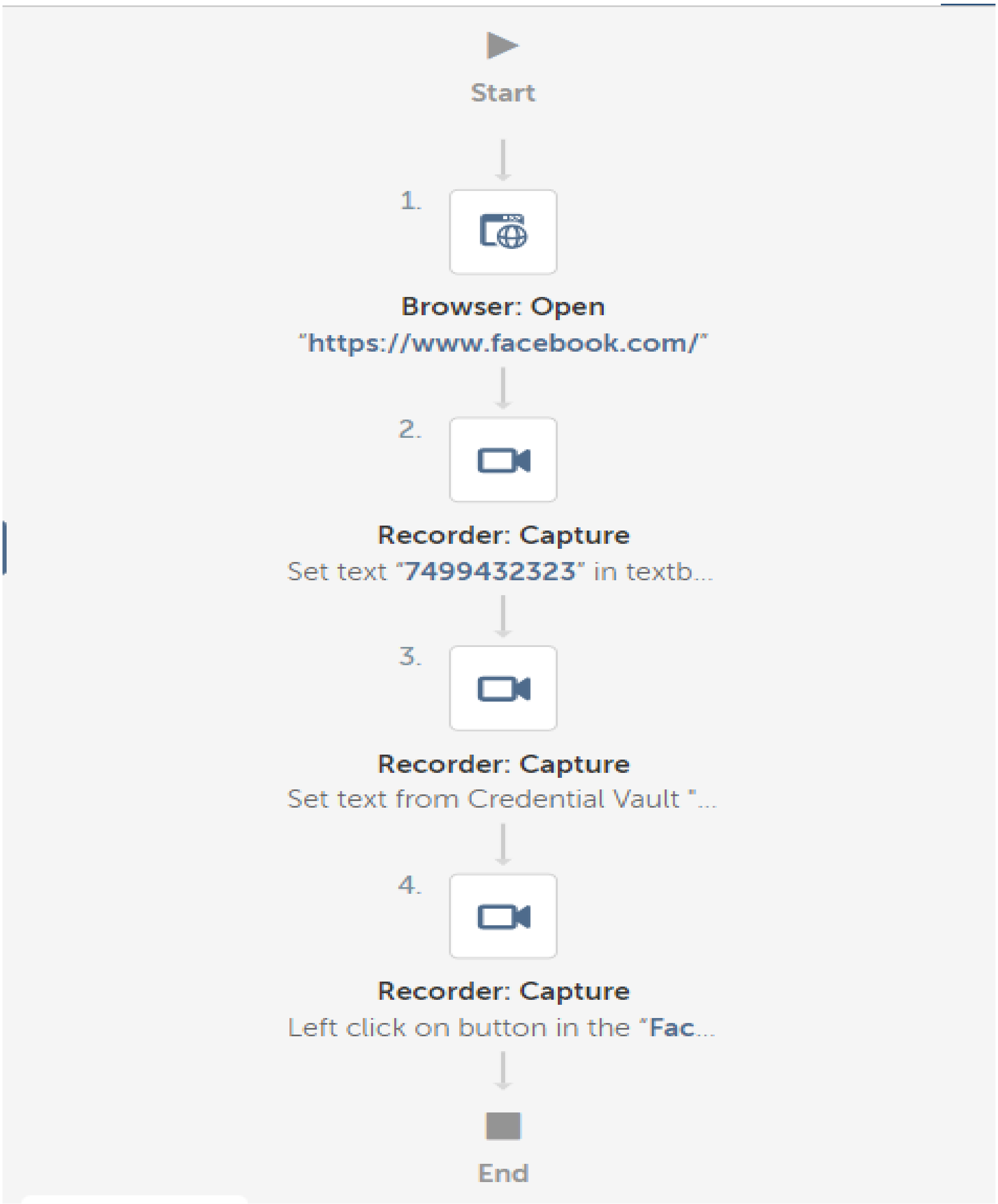
a) Using Capture: Object to capture the Login Button click

b) Select Action as Left Click

c) Enter Keystrokes with the password with which you wish to log in

Bot Name: Facebook Launcher

Flow:



Start

1. Browser: Open
"https://www.facebook.com/"

2. Recorder: Capture
Set text "7499432323" in textb...

3. Recorder: Capture
Set text from Credential Vault "...

4. Recorder: Capture
Left click on button in the "Fac...

End

SEMESTER II

Robotic Process Automation Course Code: MCALE23

Practical 2.2

Bot: Bot to work with MsPaint

Procedure

1. Open a new bot:

    a. From the Automation Anywhere Enterprise web interface, select Bots > Mybots.

b. Click Create new > Bot.

c. In the Create Task Bot window, enter a bot name.

d. Accept the default folder location \Bots\. To change where your bot is stored, click Choose and followthe prompts.

e. Click Create and edit.

2. Insert a Application: Open program/file action.

a. Click Actions.

b. Search for the Application package.

Click in the Actions search box and enter Application. Click the arrow toexpand the Message Box options.

c. Double-click or drag the Open action to the Bot editor (open space to theright).

3. In the dialog box on the right, specify the fields for the Open program/file action.

a. In the Location of the program/file field, browse the path for MSpaint.exe

4. Insert Recorder: Capture action

a. Click Actions.

b. Search Capture Action from Recorder package

c. In the dialog box set the Paint application.

d. In Capture object, capture any colour from the palate. Select Left Click fromAction

5. Insert Recorder: Capture action

e. Click Actions.

f. Search Capture Action from Recorder package

g. In the dialog box set the Paint application.

h. In Capture object, capture Fill with colour tool. Select Left Click fromAction

6. Insert Recorder: Capture action

i. Click Actions.

j. Search Capture Action from Recorder package

k. In the dialog box set the Paint application.

l. In Capture object, capture the area to be filled with the colour. Select Left Click from Action

7. Click Save.

Your bot is now ready to run.

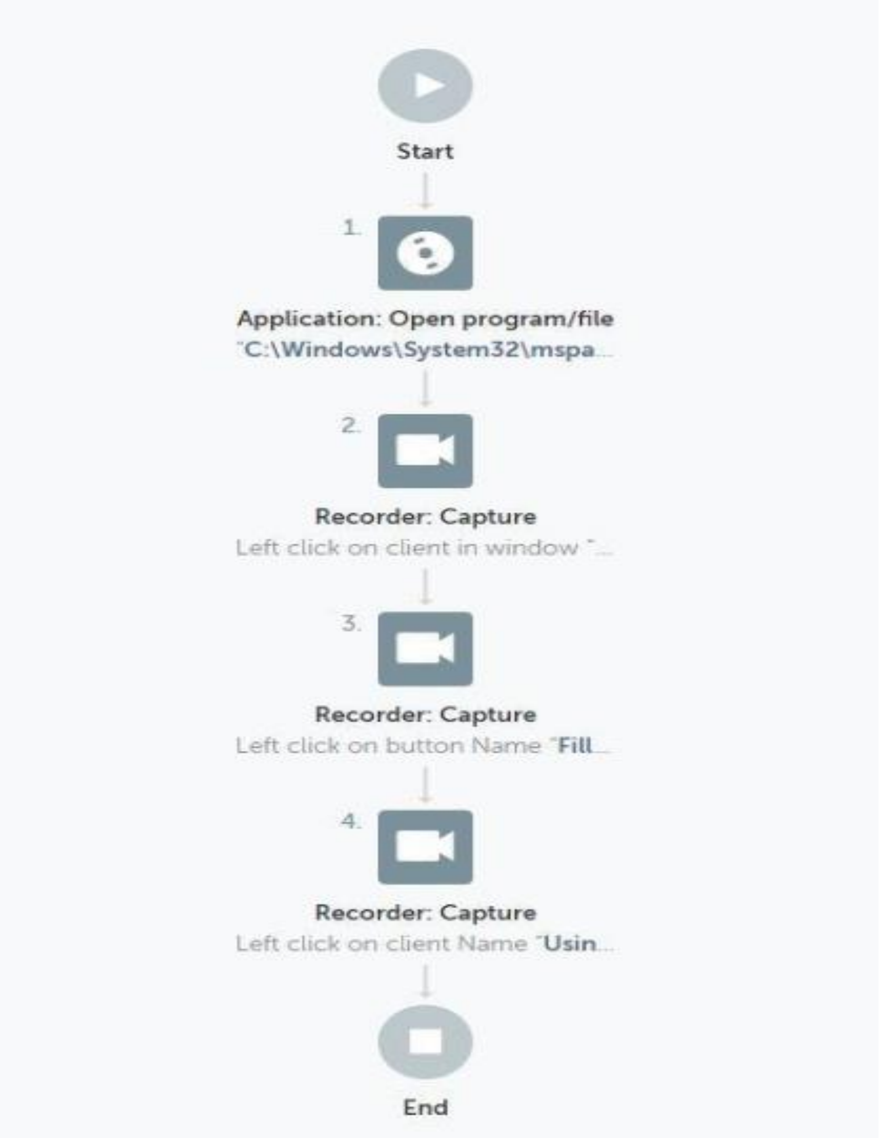
8. Test your bot.

Click Run at the top right.

The bot fills the area with the colour selected from the colour palate using the fill withcolour tool.

Bot Name: MSPaintBot

Flow:


Start
1.
Application: Open program/file
"C:\Windows\System32\mspa...
2.
Recorder: Capture
Left click on client in window "...
3.
Recorder: Capture
Left click on button Name "Fill...
4.
Recorder: Capture
Left click on client Name "Usin...
End

SEMESTER II
Robotic Process Automation Course Code: MCALE23

Practical 2.3

Bot: To get the Title of Active Window

Procedure

1. Open a new bot:
   a. From the Automation Anywhere Enterprise web interface, select Bots > Mybots.
   b. Click Create new > Bot.
   c. In the Create Task Bot window, enter a bot name.
   d. Accept the default folder location \Bots\.

      To change where your bot is stored, click Choose and follow the prompts.

    e. Click Create and edit.

2. Insert a Window action.
    a. Click Actions.
    b. Search for the Window package.
        Click in the Actions search box and enter window.
        Click the arrow to expandthe Window options.
    c. Double-click or drag the Activate action to the Bot editor (open space to theright).
3. In the dialog box on the right, specify the conditions for the Activate action.
    a. In the Choose an application window field, select word document file whichis open
4. In the dialog box on the right, specify the conditions for the Minimize action.
    a. In the Choose an application window field, select word document file whichis open
5. In the dialog box on the right, specify the conditions for the Maximize action.
    a. In the Choose an application window field, select word document file whichis open
6. In the dialog box on the right, specify the conditions for the Get Active WindowTitle action.
    a. In the Assign the windows title variable field, enter variable nameWindowsTitle1 to have Title bar details

1. In the dialog box on the right, specify the conditions for the Get Active WindowTitle action.
    a. In the Assign the windows title variable field, enter variable nameWindowsTitle1 to have Title bar details
2. Click Save.
    Your bot is now ready to run.
3. Test your bot.
    Click Run at the top right.
    The bot displays a Message box with the text Go be great!.

In the following steps, you configure a variable and insert it in the Message box

6. Create a variable.
    a. Click Variables from the accordion menu on the left panel.
    b. Click Create variable (the + sign).
    c. Enter vHelloWorld in the name field.

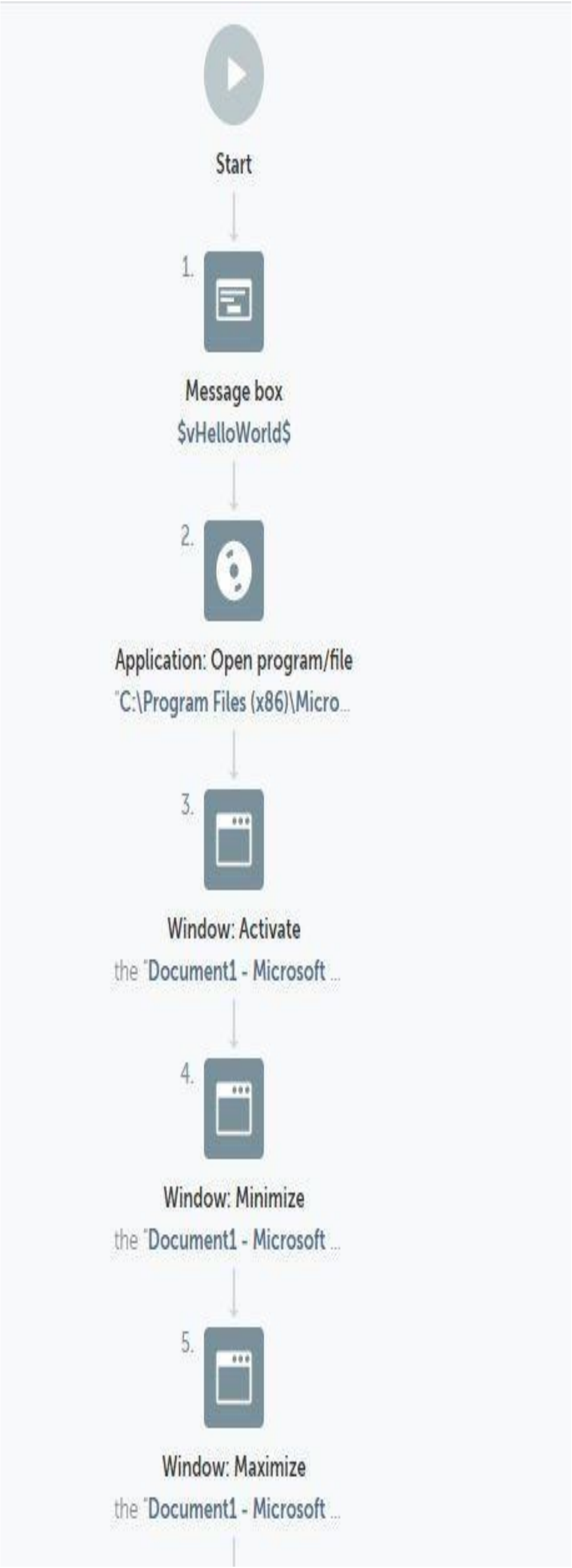
    d. Select the Use as output option.
    e. Enter Say Go be Great! with a variable in the Default value field.
    f. Click Create.

7. Assign the vHelloWorld variable to the Enter the message to display field.
    a. Click the Message box action in the Bot editor.
    b. In the dialog box, delete the text from the Enter the message to display field.
    c. Either press the F2 key or click the Insert variable icon.

       The Insert variable icon is located on the right-side of the text field.
    d. Select vHelloWorld from the drop-down list.
    e. Click Yes, insert.
8. Click Save.
9. Click Run.

The bot displays a Message box with the text Say Go be Great! with a variable

Flow :



Start

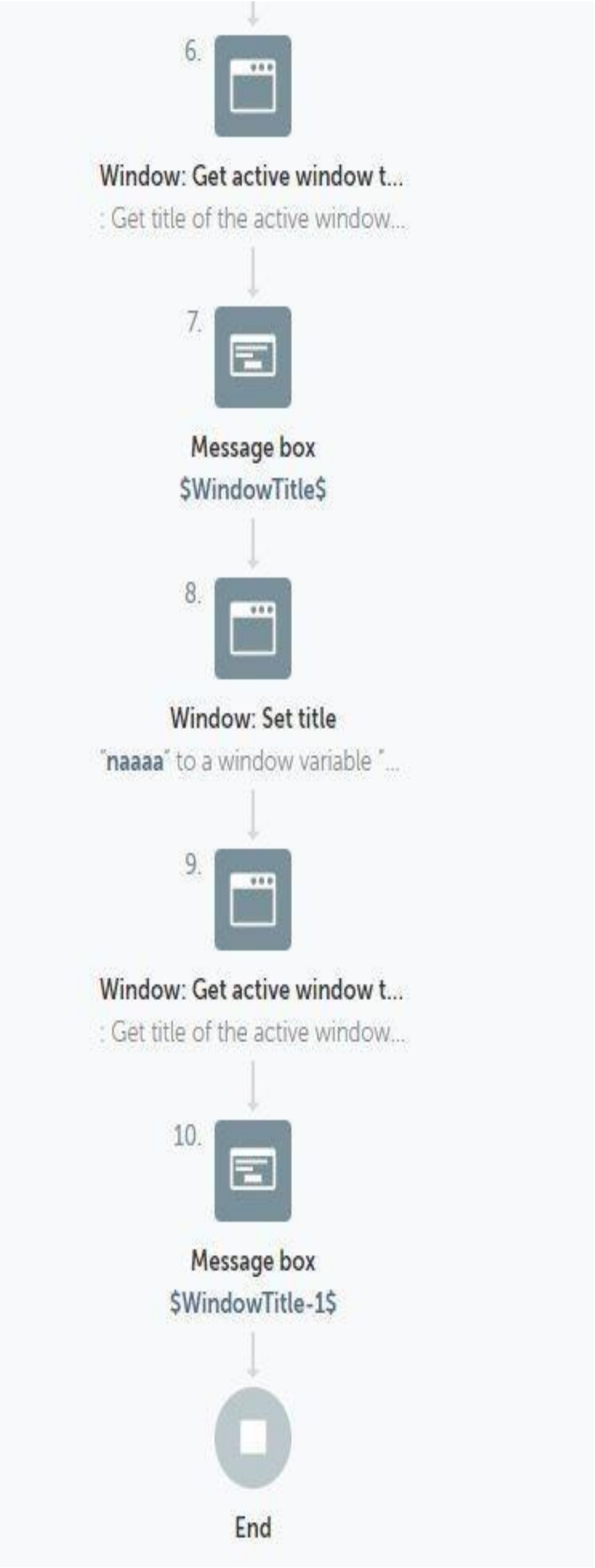
1. Message box
$vHelloWorld$

2. Application: Open program/file
"C:\Program Files (x86)\Micro...

3. Window: Activate
the "Document1 - Microsoft ...

4. Window: Minimize
the "Document1 - Microsoft ...

5. Window: Maximize
the "Document1 - Microsoft ...



6. Window: Get active window t...
: Get title of the active window...

7. Message box
$WindowTitle$

8. Window: Set title
"naaaa" to a window variable "...

9. Window: Get active window t...
: Get title of the active window...

10. Message box
$WindowTitle-1$

End

SEMESTER II
Robotic Process Automation Course Code: MCALE23
Practical 2.4

Bot: To close notepad

Procedure

Follow these steps to add a Click action:

1. In the Actions palette, double-click or drag the Click action from the Mouse package.
2. Specify the window in which to capture the mouse click: Choose from the Application, Browser, or Variable tab.
3. Click Capture coordinate. The selected window appears.
4. Drag the mouse to select the click spot and left-click to capture it.

The captured spot appears in the Preview section with the coordinates underneath.

5. In the Button option, specify the button to click.

Choose from Left Button, Right Button, or Middle Button.

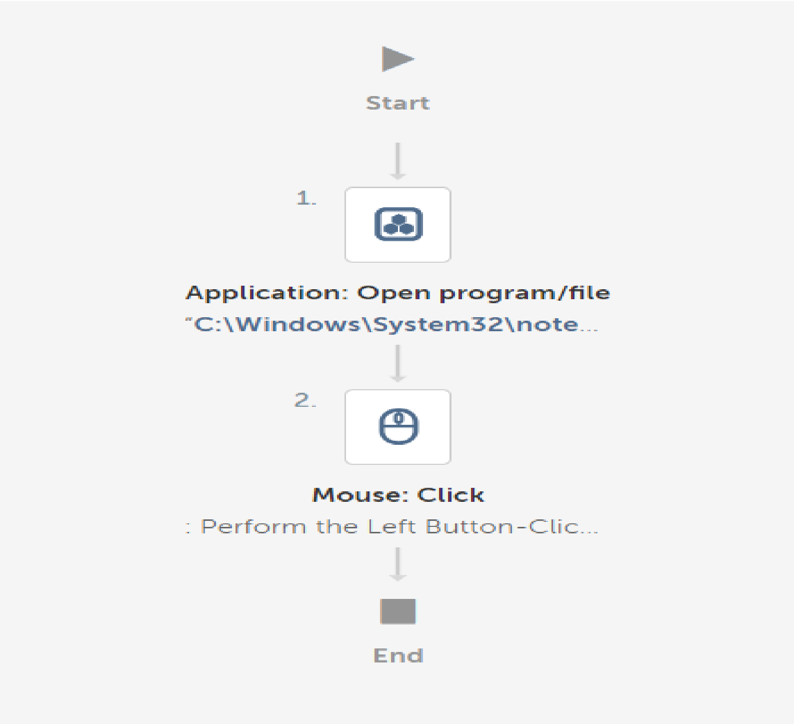
6. In the Event option, specify the event.

Choose from Click, Double click, Button up, or Button down.

7. Click Save.

Bot Name: Close Notepad

Flow:



1. Application: Open program/file "C:\Windows\System32\note...

2. Mouse: Click : Perform the Left Button-Clic...

Bot :To automa ments and storing the data

Step 1: Using L

a) Set Itera
b) Specify the Folder Path D:\Sudeshna2021\RPA\Purchase Order containing thePDF Files
c) Assign File name and extension to a dictionary variable FilesInFolder2

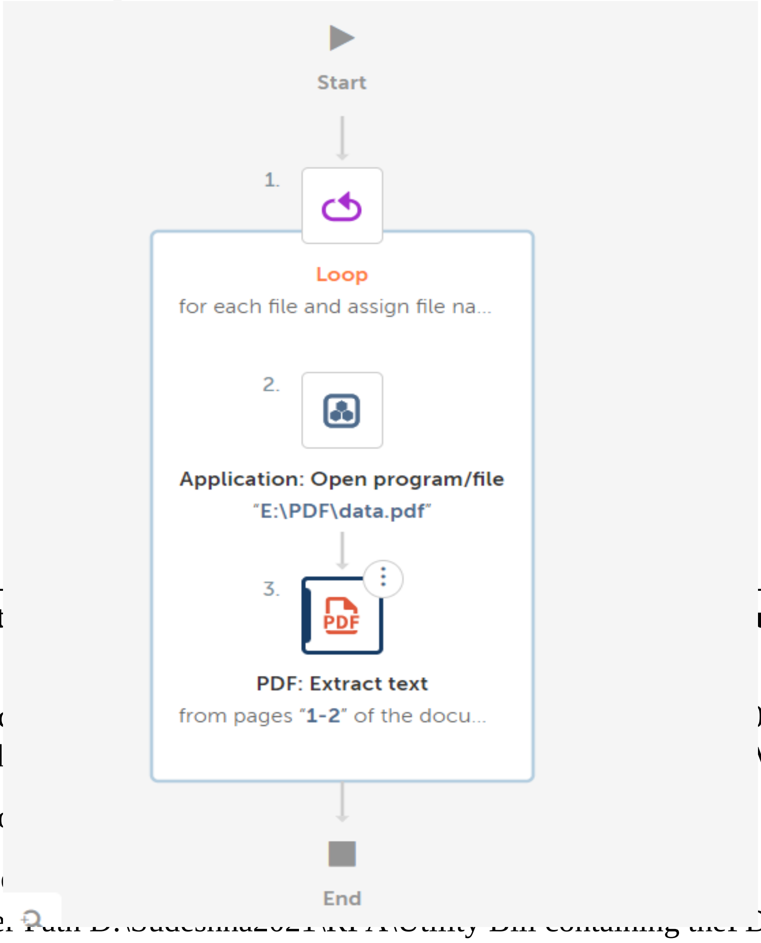
Step 2: Inside the loop

a) Using Application: Open open each of the program file in D:\Sudeshna2021\RPA\Purchase

b) Order\$FilesInFolder2{name}$.$FilesInFolder2{extension}$

c) Using PDF: Extract text export data to a text file D:\Sudeshna2021\RPA\LogFile\Log File for image text.txt

Bot Name: ExtractMultiplePDFtoText



Start

1. Loop
for each file and assign file na...

2. Application: Open program/file
"E:\PDF\data.pdf"

3. PDF: Extract text
from pages "1-2" of the docu...

End

Bot :To automat uments and storing the data

Step 1: Using Exc)21\RPA\Log File\Bill.xlsx. Sel Vrite mode only

Step 2: Using Loo

a. Set Iterator as Fil

b. Specify the Folde. Path D:\Sudeshna2021\RPA\Utility Bill containing the PDF Files

c. Assign File name and extension to a dictionary variable FilesInFolder3

Step 3: Inside the loop

a) Using PDF:Extract field action specify the path D:\Sudeshna2021\RPA\Utility Bill \$FilesInFolder3{name}$.$FilesInFolder3{extension}$. Extract fields Invoice Number and Invoice Date from the file. Map each key name extracted to a variablename.

b) Using Excel advanced: Set cell add the $InvoiceNo$ variable to the active cell

c) Using Excel advanced: Go to cell go to one cell right to the active cell

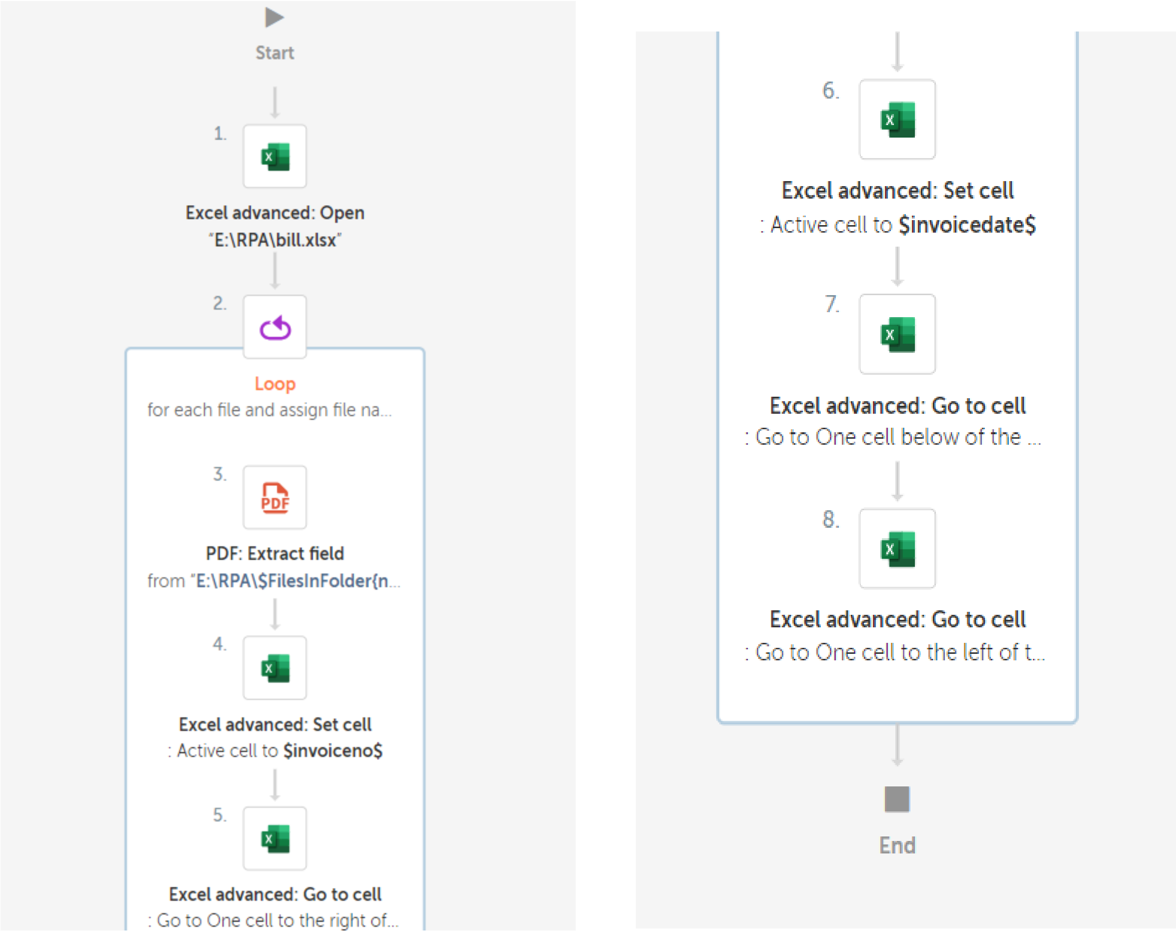
d) Using Excel advanced: Set cell add the $InvoiceDate$ variable to the active cell

e) Using Excel advanced: Go to cell go to one cell below to the active cell

f) Using Excel advanced: Go to cell go to one cell left to the active cell

Bot Name: ExtractMultiplePDFtoExcel

Flow:



Start

1. Excel advanced: Open "E:\RPA\bill.xlsx"

2. Loop
for each file and assign file na...

3. PDF: Extract field
from "E:\RPA\$FilesInFolder(n...

4. Excel advanced: Set cell
: Active cell to $invoiceno$

5. Excel advanced: Go to cell
: Go to One cell to the right of...

6. Excel advanced: Set cell
: Active cell to $invoicedate$

7. Excel advanced: Go to cell
: Go to One cell below of the ...

8. Excel advanced: Go to cell
: Go to One cell to the left of t...

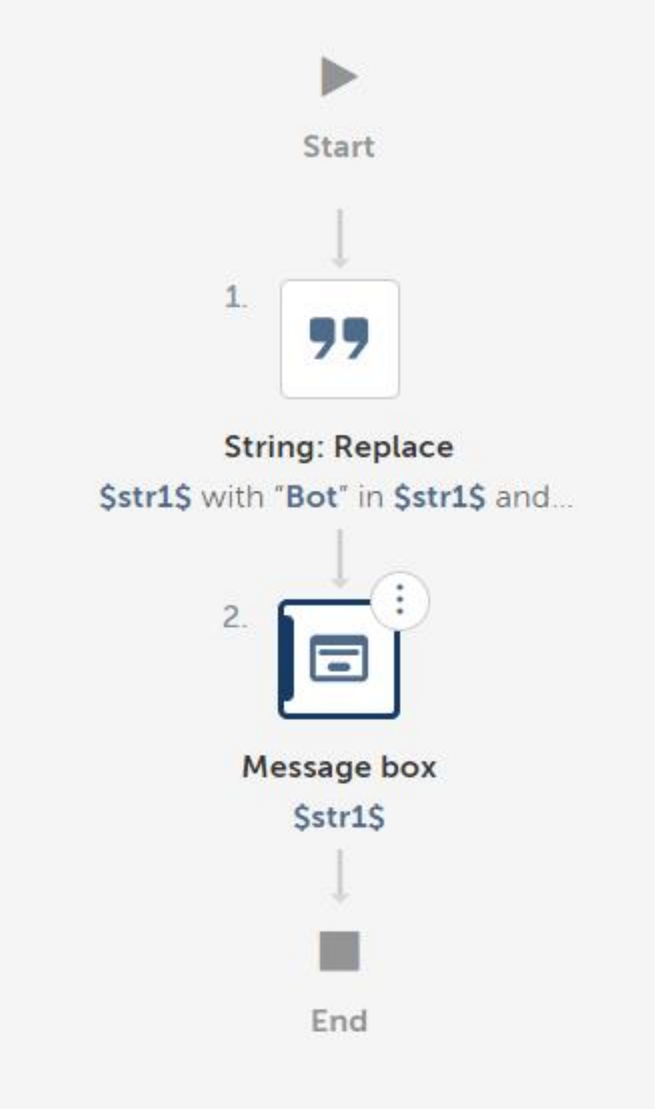
End

Practical 3.3

Bot : Automate task of replacing few characters from a string

Procedure:
Step 1: Create a String variable $str1$ and assign value to it
Step 2: Using String:Replace replaces specified part of a 'Source string' witha 'Replacement string'
3. Display the resultant string in the message box

Bot Name: ReplaceTextInString
Flow:



Start

1. String: Replace
$str1$ with "Bot" in $str1$ and...

2. Message box
$str1$

End

Practical 3.4

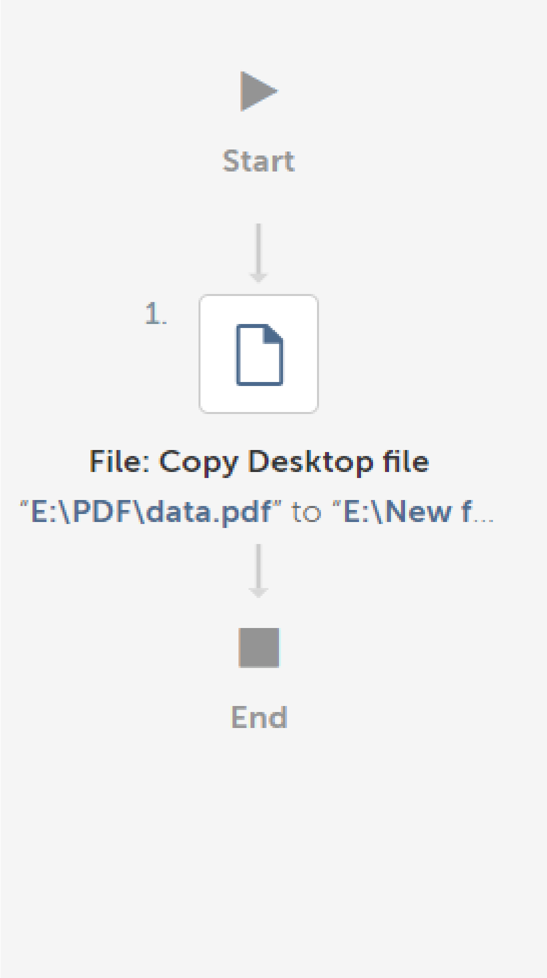
Bot: To Automate task of copying files from a source folder to destination folderTask:

1. Select source folder to copy
2. Create destination folder
3. Copy files
4. Zip the copied files

Procedure

5. Create folder from folder package (Give path)
6. Create file from file package (Give Path)
7. Copy file from file package (Give source and destination path)
8. Copy action from folder package (Mention source file and destination folder)
9. Use zip action from folder. (Specify files to compress, destination zipped file)
10. Save

Bot Name: copyFolder



1. File: Copy Desktop file
“E:\PDF\data.pdf” to “E:\New f...

Practical 3.5

Bot: Extract a table from webpage

This example uses actions from the Browser, Data Table, Recorder, and Window packages.

To extract data from a table, do the following steps:

Procedure

1. Open a new bot:
    a. From the Automation Anywhere Enterprise web interface, select Bots > Mybots.
    b. Click Create new > Bot.
    c. In the Create Task Bot window, enter a bot name.
    d. Accept the default folder location \Bots\.

        To change where your bot is stored, click Choose and follow the prompts.

    e. Click Create and edit.
2. Open a browser window to the web page from which you will extract the table.
    a. Double-click or drag the Browser > Open action.
    b. In the URL field, enter https://en.wikipedia.org/wiki/United_States_at_the_Olympics
    c. Select your preferred browser.
    d. Click Save.
    e. Click Run.

        The bot opens the window.

3. Specify the table.
    a. Double-click or drag the Recorder > Capture action.

        Click the Browser tab and select https://en.wikipedia.org/wiki/United_States_at_the_Olympics from the drop-down list.

        If the window title does not appear in the list, click Refresh.

    b. Click Capture object.
    c. Hover over the table in the page.

An orange box appears, surrounding the table

a. Click the table.

    The Object Processing message box appears.

b. Return to the Enterprise Control Room.

c. In the Object properties table, verify the Control Type is TABLE.If it is not, click Recapture object.

d. From the Action drop-down list, select Get table.

e. In the Assign output to variable field, create the tHostingGaames. The Daily Stock Market Overview window is saved as the variable window-1.

4. Specify the file where to save the data.
    a. Double-click or drag the Data Table > Write to file action.
    b. From the Data table name list, select tHostingGaames.
    c. Provide a file path to create a CSV file.

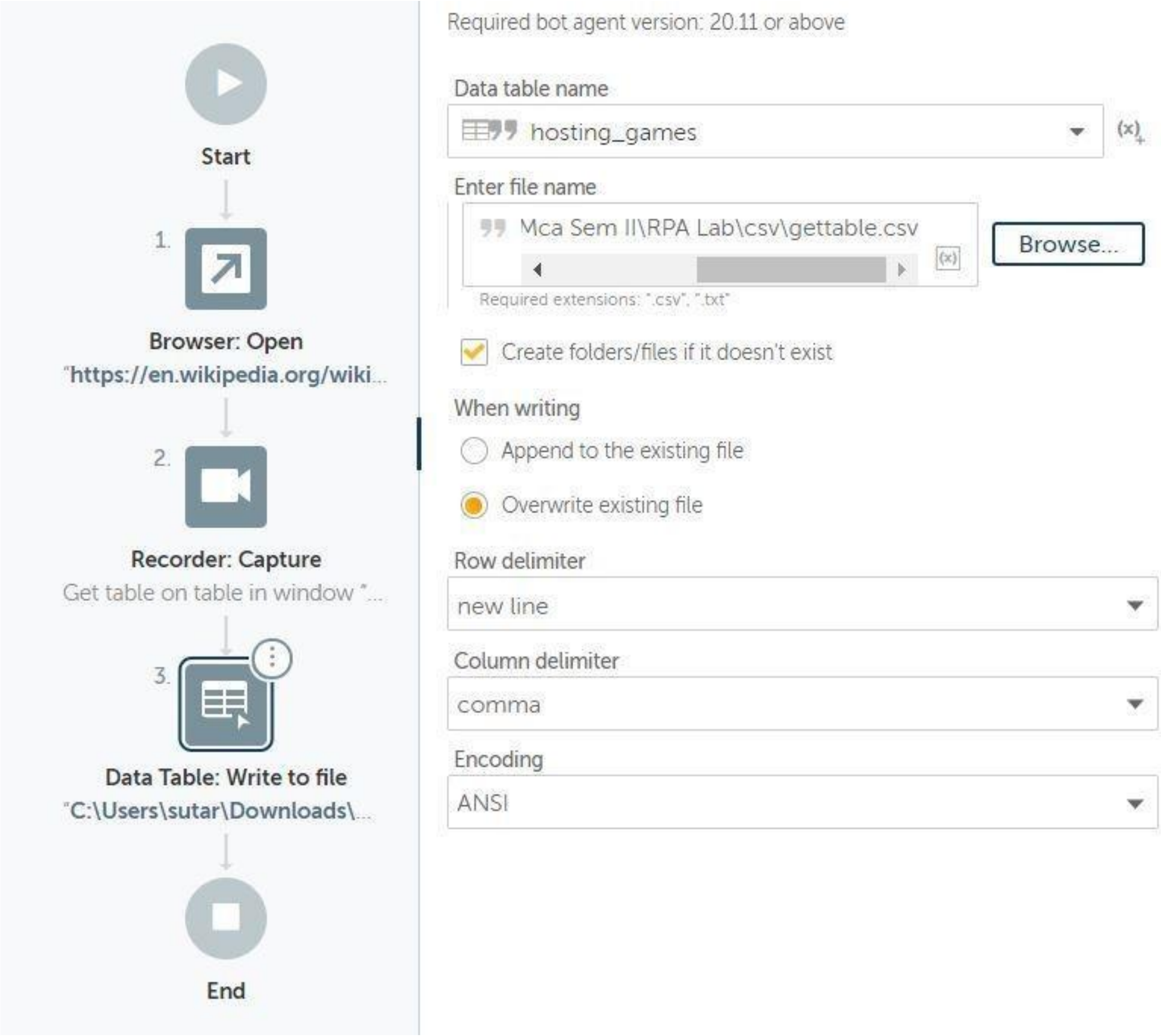
       For example, G:\Teaching2021\SEM002\RPA\Practical\RPA Orientaton

       Matrial_NMITD\RPA Orientaton Matrial\RPA Practicals\HostingGames.csv.
    d. Select the Create folders/files if it doesn't exist option.
    e. Select to overwrite the existing file.
5. Click Save.
6. Click Run.

   The bot creates a CSV file on the desktop with data on seven indexes, their values,and net change.

Bot Name: WebTableExtractBot

Flow:



1. Browser: Open "https://en.wikipedia.org/wiki...

2. Recorder: Capture Get table on table in window "...

3. Data Table: Write to file "C:\Users\sutar\Downloads\...

Required bot agent version: 20.11 or above

Data table name: hosting_games

Enter file name: Mca Sem II\RPA Lab\csv\gettable.csv

Required extensions: ".csv", ".txt"

☑ Create folders/files if it doesn't exist

When writing

○ Append to the existing file

● Overwrite existing file

Row delimiter: new line

Column delimiter: comma

Encoding: ANSI

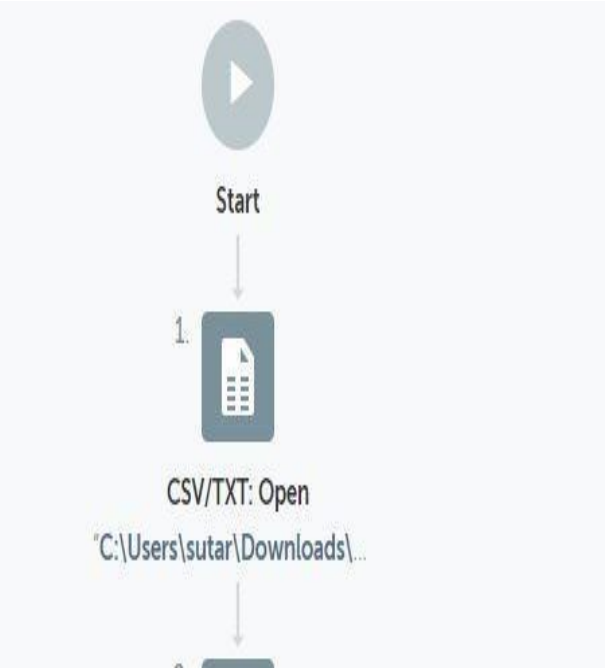
SEMESTER II
Robotic Process Automation Course Code: MCALE23
Practical 3.6

Bot: To extract string from TXT file and copy it into excel fileProcedure

1. Open TXT file (CSV/TXT package)
2. Open Excel File (Basic Excel PAckage)
3. Select Cell in Excel sheet from where c
Basic ExcelPackage)
4. Loop (For each row in CSV/TXT)
5. Messagebox (Just to display contents of
6. Set Cell mention variable name with inc
Cell” BasicExcel Package)
7. GoTo Cell (for next data in next cell)
8. Set Cell.
9. Repeat for each column
10. For next row, Go To cell: (mention “Gc
11. Click Save.
12. Close file

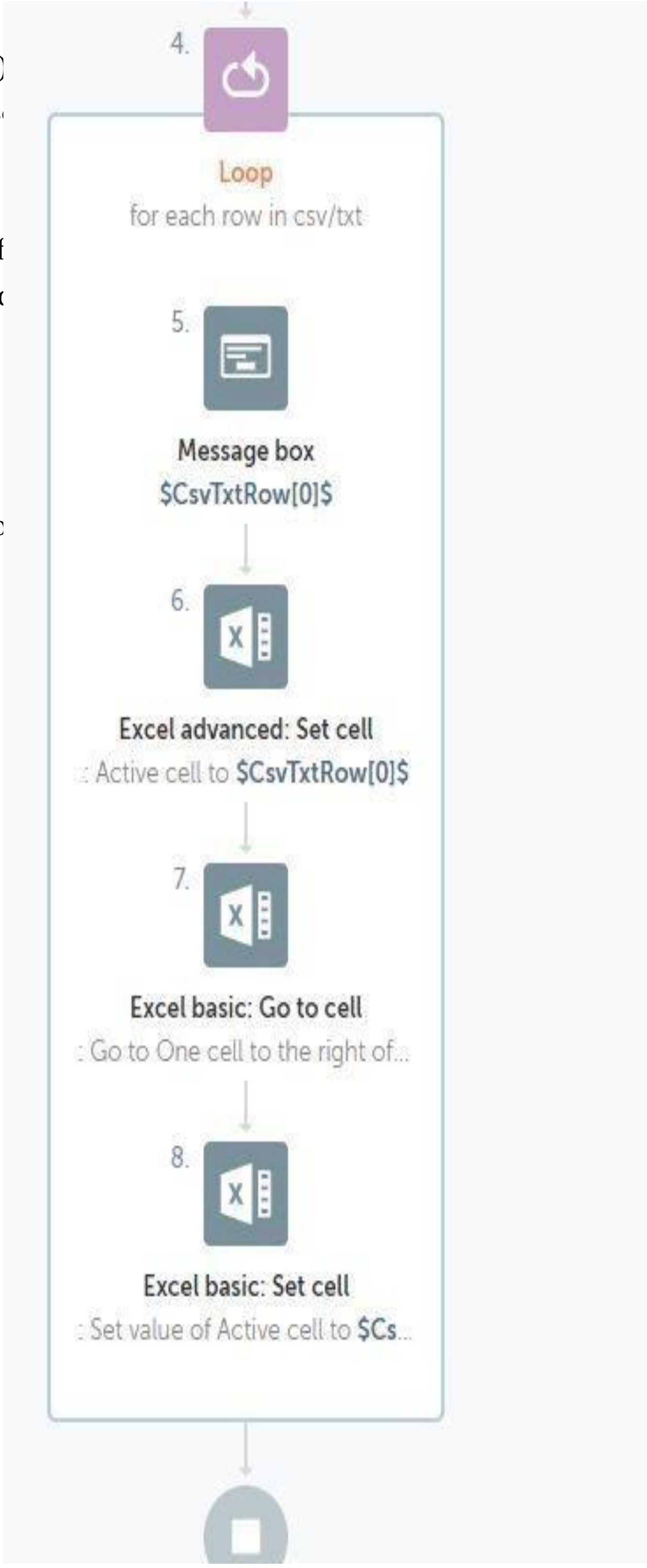
Bot Name: ImpBot

Flow:



Start

1. CSV/TXT: Open
"C:\Users\sutar\Downloads\...



4. Loop
for each row in csv/txt

5. Message box
$CsvTxtRow[0]$

6. Excel advanced: Set cell
: Active cell to $CsvTxtRow[0]$

7. Excel basic: Go to cell
: Go to One cell to the right of...

8. Excel basic: Set cell
: Set value of Active cell to $Cs...

SEMESTER II
Robotic Process Automation Course Code: MCALE23
Practical 3.7

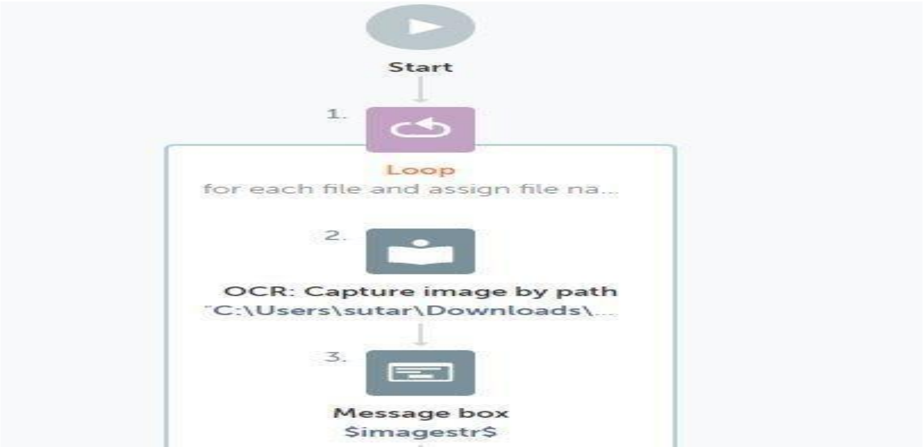
Bot: To capture text from imagePart 1:

Procedure

1. Store images into folder
2. Use loop: for each file in folder:
   2.1 Mention folder path
   2.2 Create variable of type dictionary
3. Use capture image by path from OCR package
   3.1 Mention image path in this format: E:\sudeshna\RPA\Automation nywhere\OCRCErtificates\$FilesInFolder-1{name}$.$FilesInFolder-1{extension}$
   3.2 Assign value to variable: Use string variable
4. Messagebox to display string
5. Log to File :
   5.1 Give pathname to create log in TXT file
   5.2 Enter text to log: mention variable: from 3.2

6. Click Save.
7. Close file

Bot Name: TextFromImage
Flow:



Start

1. Loop
for each file and assign file na...

2. OCR: Capture image by path
"C:\Users\sutar\Downloads\...

3. Message box
$imagestr$

SEMESTER II
Robotic Process Automation Course Code: MCALE23
Practical 4.1

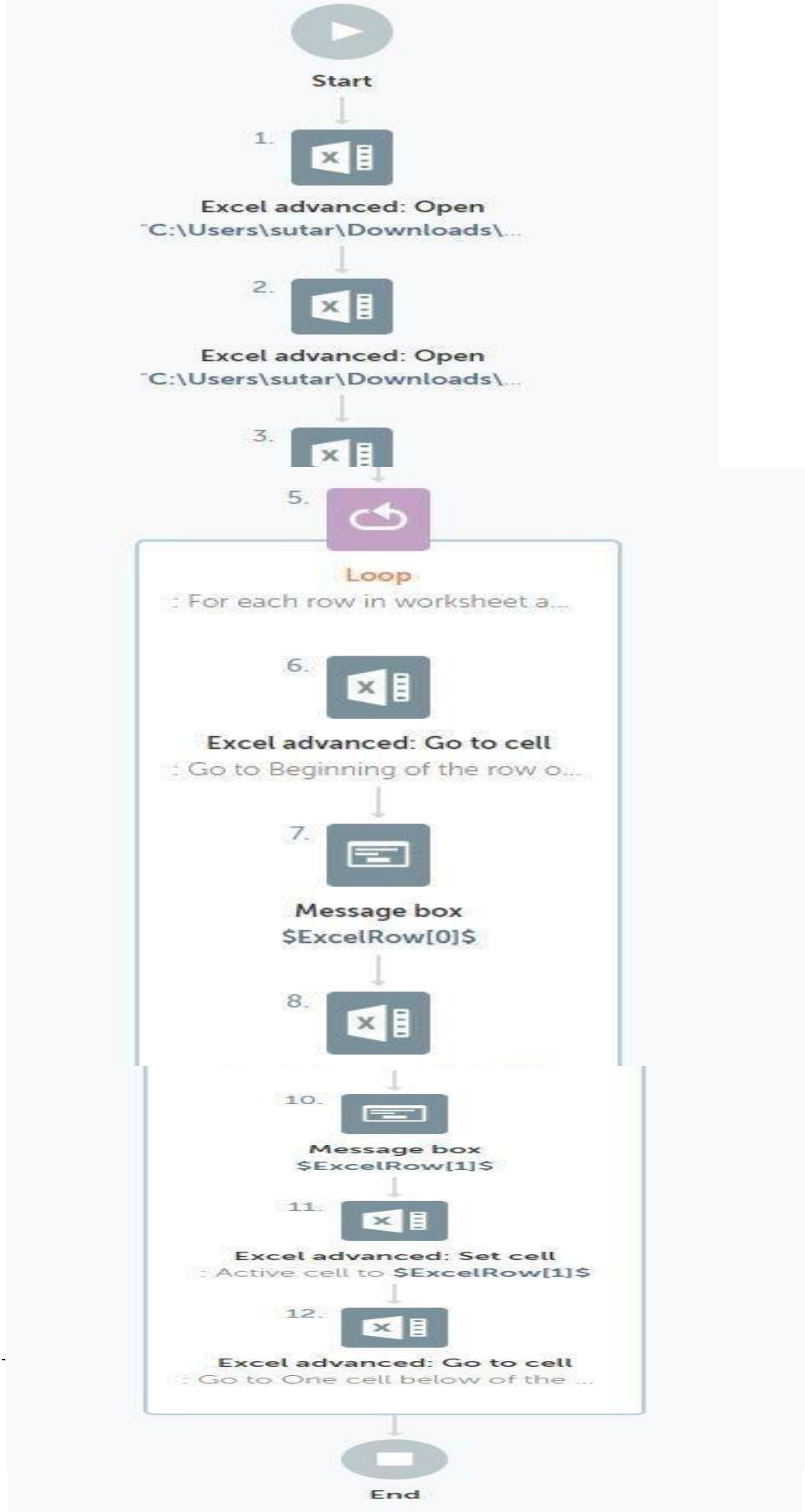
Bot: To Extract Data from Excel

Extracting a particular Cell Value:
Step 1: Using Excel advanced:Open action open the Excel file from the path D:\Sudeshna2021\RPA\Hosting Games Demo.xlsx
Step 2: Using Excel advanced: Go to Cell action place the cursor in a specific cell C2 in order extract Dates of the event from the excel file.
Step 3: Using Excel advanced: Get single cell read a particular cell value into a variable $Data1$
Step 4: Display the Variable value using MessageBox action
Extracting a Multiple Cell Values:
Step 1: Using Excel advanced:Open action open the Excel file from the path D:\Sudeshna2021\RPA\Hosting Games Demo.xlsx
Step 2: Using Excel advanced: Get multiple cell action read the contents from cell C2 to cell C9 in to a table variable $ExcelSheetData$
Step 3: Loop through each row in a table $ExcelSheetData$
a) Assign the value of every record in the iteration into Record variable $DateRecord$
b) Display the Date value using MessageBox action

Extracting Records based on condition and saving it in an Excel:
Step 1: Using Excel advanced:Open action open the Excel file from the path D:\Sudeshna2021\RPA\Hosting Games Demo.xlsx in Session 1
Step2: Using Excel advanced:Open action open the Excel file from the path D:\Sudeshna2021\RPA\Output\Hosting Game Output1.xlsx in Session 2
Step 3: Using Excel advanced: Get multiple cells retrieve the values of multiple cells in an excel worksheet opened in Session 1
Step 4: Use Loop iterating for each row in worksheet. It loops through all rows in excel file of Session 1 and retrieves each record in a variable $ExcelRow1$
Step 5: Create a number variable $var1$ and assign it a value 2
a. If number $ExcelRow1[4]$ Greater Than (>) 1000
I. Use Excel advanced: Set Cell assign specific cell A$var1.Number:toString$ to $ExcelRow1[0]$ in Session 2
II. Use Excel advanced: Set Cell assign specific cell B$var1.Number:toString$ to $ExcelRow1[1]$ in Session 2
III. Use Excel advanced: Set Cell assign specific cell C$var1.Number:toString$ to $ExcelRow1[2]$ in Session 2
IV. Use Excel advanced: Set Cell assign specific cell D$var1.Number:toString$ to $ExcelRow1[3]$ in Session 2
V. Use Excel advanced: Set Cell assign specific cell E$var1.Number:toString$ to $ExcelRow1[4]$ in Session 2
VI. Use Excel advanced: Set Cell assign specific cell F$var1.Number:toString$ to $ExcelRow1[5]$ in Session 2
VII. Perform Number: Increment to increment $var1$ by 1

Bot Name: ExtractFromExcel

Flow:



Start

1. Excel advanced: Open "C:\Users\sutar\Downloads\...

2. Excel advanced: Open "C:\Users\sutar\Downloads\...

3.

5. Loop : For each row in worksheet a...

6. Excel advanced: Go to cell : Go to Beginning of the row o...

7. Message box $ExcelRow[0]$

8.

10. Message box $ExcelRow[1]$

11. Excel advanced: Set cell : Active cell to $ExcelRow[1]$

12. Excel advanced: Go to cell : Go to One cell below of the ...

End

variety of

to pretrainedAI models (domains) for reading invoices, purchase orders, and other documents. It also allows you to train your own domains to support new document types.

Here are four necessary steps to complete:

create
classify
train
production

1. **Upload**: Prepare and upload sample documents, create an IQ Bot Learning Instance.
2. **Analyze**: IQ Bot will automatically organize similar documents into groups and mapfields with areas on the document.
3. **Review**: You will review field mapping for each group and make necessary corrections. IQ Bot will learn from these corrections and improve the accuracy of data extraction.
4. **Publish**: After completing IQ Bot training, you can use the Learning Instance inproduction by calling it from RPA Task bot.

Bot : Design IQ Bot

Step 1: Goto **IQ Bot>Learning Instances** and then click **Train Other Documents**

a. Click **Get Started** Button

Step 2: **Create New Learning Instance** and provide the general information

a. Provide **Instance Name** Instance1_demo1**, Document Type** Invoice**, Upload your documents** upload 2 files

b. Deselect all fields under Items Repeated

c. Wait till the documents are analyzed and classified(i.e. Apply classifier, Perform semantic analysis, Learning from aliases, Recognize patterns, Extracts data with respect to every document uploaded)

Step 3: **Review the mapping and To train a group:**

- Click each field (left panel) to verify the label and value (center panel) is the same as the document (right panel).
- Preview the mapping after all fields are verified.
- Save and go to the next group.
- Finally save the current training

Step 4: Set to Production
IQ Bot Learning instance: Instance1_demo1

Create a RPA Task Bot to upload documents for processing to IQBots

Step 1: Use a **Loop** Iterating for **each file in folder** and create a dictionary variable $FilesInvoives1$

a. Use **IQ Bot: Upload Document** and select the **Learning Instance Name** Instance1_demo1 and provide **path** as C:\Users\admin\Desktop\IQBotSampleDocs\Invoices\**$FilesInvoices1{name}$**.**$FilesInvoic es1{extension}$,** Create a variable to hold the result as $ResultOfIQBot$

b. Add a **Message Box** to display the message Invoice uploaded for processing**$ResultOfIQBot$**

Step 2: Use **IQ Bot: Download All Documents**

a. Provide the **server Instance Name** Instance1_demo1

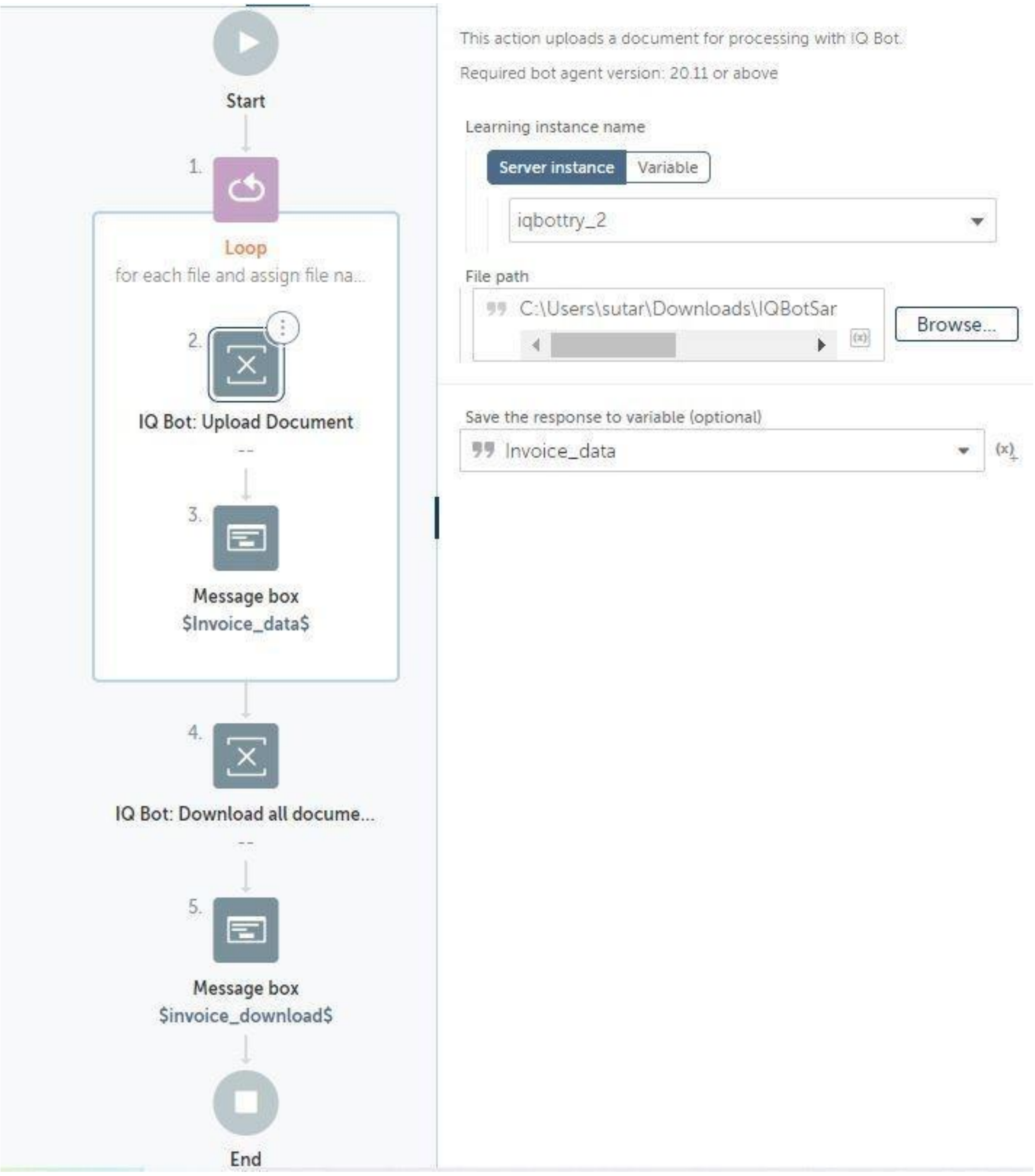
b. Provide the **Path for Local Output Folder**

C:\Users\admin\Desktop\IQBotSampleDocs\Output

c. Save the responses to a variable $IQBotDownload$

d. Add a **MessageBox** to diaplat the message Result Downloaded: **$IQBotDownload$**

Bot Name: IQBotDemo



This action uploads a document for processing with IQ Bot.

Required bot agent version: 20.11 or above

Learning instance name

Server instance | Variable

iqbottry_2

File path

C:\Users\sutar\Downloads\IQBotSar

Save the response to variable (optional)

Invoice_data

SEMESTER II
Robotic Process Automation Course Code: MCALE23
Practical 6

Introduction to UiPath in Robotic Process Automation (RPA)

Robotic Process Automation (RPA) is the automation of workflows using softwares or robots. This process reduces human interaction in these workflows.

RPA allows users to configure robots or softwares that enforce business processes through the integration of human actions with digital systems. RPA software collects data and orchestrates applications using the user interface. They interpret data and communicate with other applications to perform various repetitive tasks.

UiPath is one such RPA tool that can be used for various forms of automation such as Citrix automation, PDF automation, Web automation, and Windows desktop automation.

In web automation, UiPath enables the web developers to perform various actions like data extraction, application transfer, report generation, and website testing.

This tool can be integrated with technologies such as Artificial Intelligence (AI), Business Process Management (BPM), and Enterprise Resource Planning (ERP). Its end-to-end solutions enable users to automate redundant tasks. The main programming language used in this tool is Visual Basic (VB).

UiPath components

The following are the components of UiPath:

- **UiPath Studio**: This component uses visual aids such as diagrams and flowcharts to design automation processes. It also involves the coding and sequencing of these processes. It consists of Graphical User Interface (GUI) buttons, drag and drop features, and pre-built templates.
- **UiPath Robot**: In this component, multiple bots are deployed to execute the automation processes using pre-defined rules.
- **UiPath Orchestrator**: This component functions as a web-based application that schedules, deploys, and manages processes.

Why UiPath is widely used?

The following are the reasons why UiPath is widely used by developers and organizations:

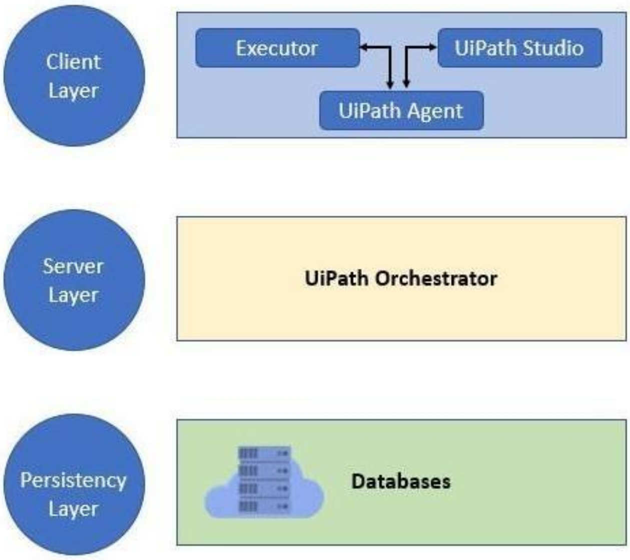
- **Compliance**: It can help an organization to comply with industry regulations, since it supports internal review.
- **Cost-effectiveness**: This tool is cost-effective because there are minimal operational costs.
- **Security**: The security features of UiPath prevents users from various forms of attacks. When a user tries to log into the system using a different computer, it will automatically end the previous session (login).
- **Community-centric**: It offers a free trial in its community edition with an active community support with tutorials and tips.
- **Speed and accuracy**: It completes tasks faster and with greater precision as compared to humans.
- **Customer experience**: It is an effective way of improving customer experience.
- **Productivity**: It boosts productivity in an organization, since many tasks can be completed simultaneously within a short time.

Features of UiPath

- **In-built recorder**: Users can record their activities using its in-built recorder.

- **Scalability**: It has effective debugging handling mechanisms. It can also handle various AI applications.
- **Hosting options**: Hosting can be done in a virtual terminal or a cloud.
- **Central repository**: It consists of a centralized repository that enhances simultaneous handling of robots by users.
- **Application compatibility**: UiPath is compatible with various applications, including desktop and web applications.
- **Scrapping functionality**: It has an advanced scrapping functionality (wizard-driven) that allows users to perform data scrapping. Users can scrape data from various applications.

How UiPath works?
Understanding the UiPath architecture can help us understand how it works. The image below shows the architecture of UiPath:



UiPath works in three layers: the client layer, server layer, and persistency layer.

Client Layer
This layer consists of UiPath components that can be accessed by users. These include UiPath Robots, UiPath Studio, executors, browsers, and the UiPath agent. A developer can use these components to design various tasks that need to be automated.
The UiPath agent plays the role of a mediator between the server and the client. It provides an avenue for communicating information. Users can see available jobs in the system tray using this tool.
The agent can change the system settings and enable the current jobs to start or stop. The software robots uses the executor service to execute tasks in the window application.
Server Layer
This layer displays all the server details. The whole project is uploaded here once the robots have been fully developed for the execution of tasks.
The server layer consists of an orchestrator that performs various functions such as monitoring activities, scheduling robots, and generating reports.
Persistency layer
This layer contains database servers that store the configuration details of the software robots. These include the users' data, logging information, asset details, and assigned tasks.