***Project Name:*** *Parcel Monkey.*

***Use Case:***

Develop a bot using Automation Anywhere to automate the Process and Business Rules outlined below:

*Website to be used*: https://www.parcelmonkey.co.uk/

Input File Required: ParcelMonkey.xlsx

***Process Use Case***:

1. Read information/data from existing input file required.

2. Based on business rules / conditions mentioned below, extract the data and produce output as stated below.

***Business Rules / Conditions***

1. Carry out the following update on the form:

a. Collect from-Read from input file

b. Delivering to- Read from input file

c. Parcel Weight- Read from input file

Note: Update 15 rows in excel

2. Filter with Rating: Highest first

3. Extract delivery service name and insert into database.

4. If you stop the task during the execution and want to run the task, the extraction should continue

from the last incomplete record in the list.

5. Send an email with the output file.

***Introduction:***

Bot is intended to update the delivery service name in the Input Excel File with required information acquired from the Parcel Monkey website. The Bot can handle various scenarios that can arise during the process, and has been tested on a number of test cases.

Bot Reads the data from the Input Excel file i.e. Source and Destination of the package, submits them in the Parcel Monkey website and fetches the details about the quote, updates them in the Input Excel file under the column name status.

***Configuring Bot:***

To deploy bot onto a system, it requires expected folder structure to be set up and the presence of required files at mentioned locations. i.e.

This bot has two dependency files viz Input Excel file and the config file.

***Configuration file:***

Name: “ParcelMonkeyConfig.xml”.

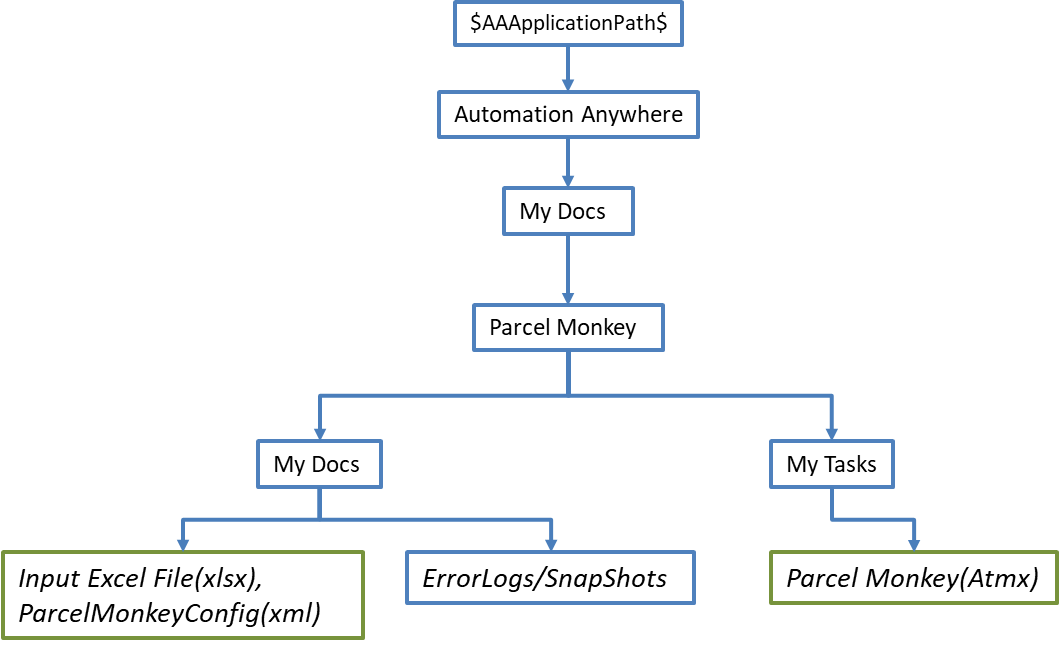
Location: “$AAApplicationPath$\Automation Anywhere\My Docs\ParcelMonkey\My Docs”.

***Input Excel File:***

Name & Location: Can be configured using Configuration File.

***Folder Structure Expected:***

The presence of below folder structure is important for the execution of the bot, ErrorLogs folder should consist of sub folder named Snapshots to store the error snapshots. And the ErrorLogs folder stores the Log files in csv form.

****** The Input Excel sheet’s location and name can be managed using the Xml config file, and should consist of four columns.

***Variables & Configuration:***

*Use the following information to configure your bot.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parameter Name** | **Type** | | **Direction** | **Additional Info** |
| **Input Excel File** | | | | |
| Input Excel file | File | | Input | Provide input records to be updated according to the shipping details. |
| **Content of CSV file** | | | | |
| Source | Column Name | | Input | Source of the Package. |
| Destination | Column Name | | Input | Destination of the Package. |
| Weight | Column Name | | Input | Weight of the Package. |
| Status | Column Name | | Output | Extracted from the Website. |
| **Config File** | | | | |
| ParcelMonkeyConfig.xml | File | | Input | Provides the Email ID and the path of the input excel file. |
| **Content of xml file** | | | | |
| Excel File Path | | String | Input | Provides the file path to the input excel file. |
| Email | | String | Input | Provides Email address to be used inside the bot. |
| URL | | String | Input | Points to the website of Parcel Monkey. |
| **Other fields** | | | | |
| vInputPath | | String | Input | It is taken from the config file and points to the Input excel file location. |
| vEmail | | String | Input | It is taken from the config file and provides the Email to be used. |
| vURL | | String | Input | It is taken from the config file and points to the website of parcel monkey. |
| vCounter | | Value | Input | It works along with counter variable, considering the presence of headers. |
| vCollectFrom | | String | Input | It is taken from the input excel file and submitted to the website. |
| vDeliveringTo | | String | Input | It is taken from the input excel file and submitted to the website |
| vDeliveryServiceName | | String | Output | This is Extracted from the website and is used to update information in the Excel Sheet. |
| vWeight | | Value | Input | This is taken from the input excel file and is submitted to the website. |

***Bot Process:***

The Bot follows the below mentioned flow to process the Input Excel File:

***Step1****: Verifies if configuration file is present or not, logs that configuration file is not present and stops the task, if present continues on.*

***Step2****: Reads the configuration details required from the configuration file, and check the presence of excel file, if absent, logs the same and stops the task.****Step3****:Opens the browser for Parcel Monkey URL, if failed logs the same and stops the task.****Step4****: Checks if the fourth column of the record i.e. Status is empty, if it is not, moves onto the next record, if it is, it gets the package details and submits them in website.*

***Step5****: Waits for the website to load and captures the required details and updates the status in the excel sheet. If service is available, orders the quotes based on rating and picks the one with best rating.*

***Step6****: If Service is not available or If extra details are required updates the same in excel sheet.*

***Step7****: If all the records are processed, then saves the excel sheet and closes it, along with browser and sends an email to the customer about the completion, with excel file attached.*

***Error Handling:***

*The Bot’s Error Handling Mechanism is as follows:*

*Log Files are present in the ErrorLogs folder and log file will be created for days on which Bot is executed, and name of the file will in the format day-month-year, and the snapshots will be stored in snapshots folder, name of the file will be in the format day-month-year-hour-minute-second.*

*If an unexpected exception is encountered the task will log and snapshot the details and stop running.*

***Logging & SnapShot:*** *Error Line Number and description along with time stamp will be appended to the ErrorLog created on that day, and Snapshot of the same will be stored.*

***Different Cases:***

1. **Extra Details Required:**