# Index

Assumptions:
Process execution time:
nstructions to Run the Bot:
nstructions to Install the Bot in the Environment2
NTD RPA CHALLENGE - UIPATH2

# NTD RPA CHALLENGE - UIPATH

**Author: ROBSON ELIAS GONCALVES** 

E-mail: regonca1@hotmail.com

LinkedIn profile: <a href="https://www.linkedin.com/in/robson-elias-goncalves-833a9b19/">https://www.linkedin.com/in/robson-elias-goncalves-833a9b19/</a>

#### Instructions to Install the Bot in the Environment

1. Create Assets below:

Use a Modern Folder. In this example the Folder is called: Shared

Name	Туре	Value
NTD_Google_Calendar_Login_URL	Text	https://calendar.google.com
NTD_Google_Calendar_Logout_URL	Text	https://accounts.google.com/Logout
NTD_Google_Credentials	Credential	Inform Username and Password.
		The E-mail and Password must be
		valid.
		Asset name* Type  IND_Congin_Credentials  Credential  Understand the invalidation or invalidation or will receive it, unknown specifically permittee in
		Description Description

# 2. Create a Queue

Name	Value	
NTD_Google_Calendar_Queue	Shared > Queues > Update Queue: NTD_Google_Cale General details	x Schema Definitions
	Name* NTD_Google, Calendar, Queue	Specific Data JSON Schema
	Description	Output Data (ION Schama
	Additional options  [ friforce unique references	Analysics Data (SON Schema
	Store in encrypted format  Store in encrypted format	SLA Predictions
	Auto Retry	SA for movely added quarie items that don't have a deadline.  Tendes SLA for this ignorie  Retention policy
	☐ Abendoned items ①	Retention policies control when and how old queue items are removed. Read the documentation

#### Instructions to Run the Bot:

- Create the Assests Above
- Create the Queue Above
- Run Dispatcher (Main.xaml file)
  - o Check the LogProcess.txt file
  - Check the GoogleEventsList.xlsx file
- Check the Items added to the Queue
- Run Performer (Main.xaml file)
- Check the Results in Performer Folder: \Data\Output:
  - o EventsDetails.xlsx
  - EventsSummary.xlsx
  - LogProcess.txt

#### Process execution time:

• Dispatcher: ~ 2 minutes

Performer: ~ 13 seconds

# Assumptions:

- The E-mail and Password must be valid.
- The Language of the Google Account must be English.
- Default Browser is Google Chrome.
- If any Google Account is connect, the bot LogOut to ensure the correct login (e-mail) will be used by the process.
- Google Chrome Process is killed in the begging of the process.
- The Project uses ReFramework, Dispatcher and Performer built in Studio version: 2024.4.0
- The correct order to get all data correctly is to Run the Dispatcher (Main.xaml) and after Run the Performer (Main.xaml).
- Projects:
  - o NTD.GoogleCalender.Dispatcher
    - Dependencies:

```
    UiPath.Excel.Activities = 2.23.3-preview
    UiPath.GSuite.Activities = 2.7.22
    UiPath.System.Activities = 24.11.0-preview
    UiPath.Testing.Activities = 24.4.0-preview
    UiPath.UlAutomation.Activities = 24.4.1-preview
```

- NTD.GoogleCalendar.Performer
  - Dependencies:

```
    ➤ NTD.GoogleCalendar.Performer
    ➤ Dependencies (Windows)
    > ☐ UiPath.Excel.Activities = 2.23.3-preview
    > ☐ UiPath.System.Activities = 24.3.1
    > ☐ UiPath.Testing.Activities = 24.4.0-preview
    > ☐ UiPath.UlAutomation.Activities = 24.4.1-preview
```

- Dispatcher generates 2 files in the folder: \Data\Output:
  - GoogleEventsList.xlsx -> List of the Events including the columns: Day, Month, Time, Title. This file was created to easily see a complete list of all Events from the Google Account informed.
  - LogProcess.txt -> List of all Log generate by the process.
- Performer reads a Queue and generate 3 files in the folder: \Data\Output:
  - EventsDetails.xlsx -> List of the Events including the columns: Day, Month,
     Time, Title. It is the Result from all Items from the Queue. It must be the same
     result as GoogleEventsList.xlsx file.
  - EventsSummary.xlsx ->
    - Summary of the events grouped by Weekday. Columns: Day, EventCount.
    - Total Events: Count of all Events.
    - Most Busy Day.
- Garbage Collection. All the files are deleted before the Bot processing.
- Log File. All the process records log into a file.

# **Bonus**

I have created the activity to get Google Events using GrapAPI ou API.

