

USER GUIDE

Common Requirements:

1. Operating System: Windows (Windows 10 used)
2. Install Nodejs from the web link: <https://nodejs.org/en/download/> (If not already installed. This is a one-time task.)
3. Open Command Prompt, type 'npm install' and hit enter (First time only)
4. Download the project from the GitHub link: <https://github.com/lakshmi4296/ISA-CustomerSupportAutomation>
5. Extract the project folder
6. In Command Prompt navigate to the 'Account' folder

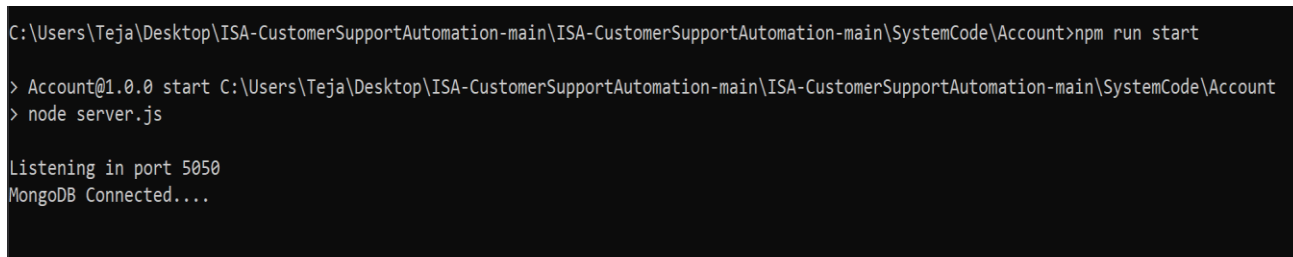


```
Command Prompt
Microsoft Windows [Version 10.0.19041.630]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Teja>cd C:\Users\Teja\Desktop\ISA-CustomerSupportAutomation-main\ISA-CustomerSupportAutomation-main\SystemCode\Account

C:\Users\Teja\Desktop\ISA-CustomerSupportAutomation-main\ISA-CustomerSupportAutomation-main\SystemCode\Account>
```

7. Type 'npm run start' and hit enter (This should start running the server and connect to the MongoDB)

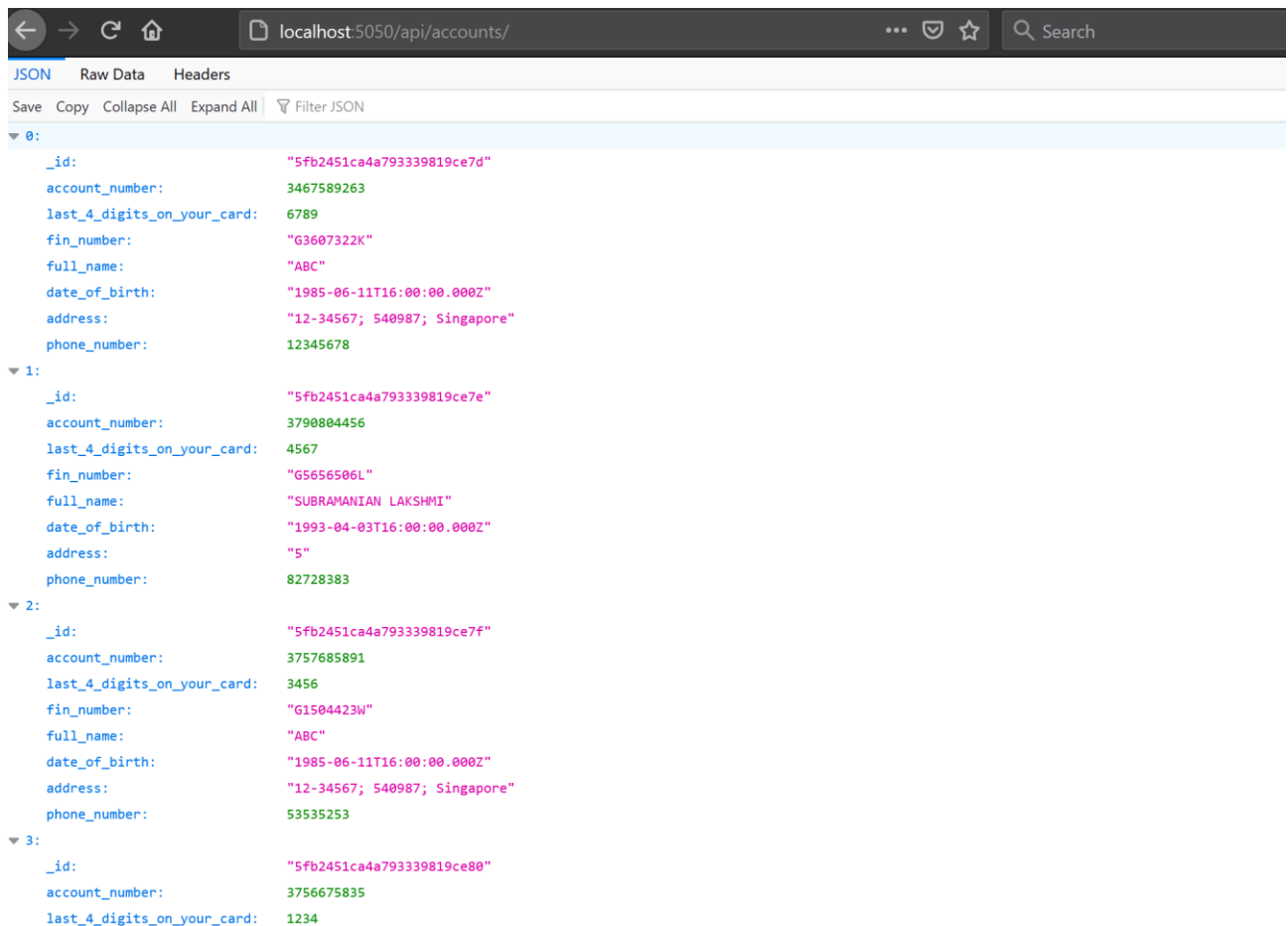


```
C:\Users\Teja\Desktop\ISA-CustomerSupportAutomation-main\ISA-CustomerSupportAutomation-main\SystemCode\Account>npm run start

> Account@1.0.0 start C:\Users\Teja\Desktop\ISA-CustomerSupportAutomation-main\ISA-CustomerSupportAutomation-main\SystemCode\Account
> node server.js

Listening in port 5050
MongoDB Connected....
```

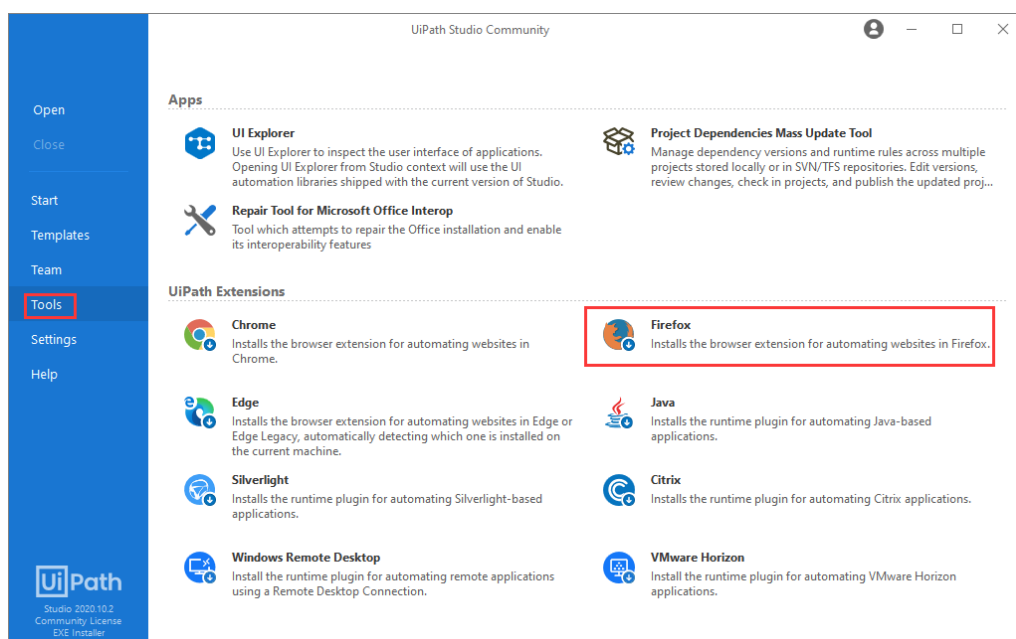
8. You can use the link: <http://localhost:5050/api/accounts/> to see the current data stored in MongoDB database



Email Automation Requirements:

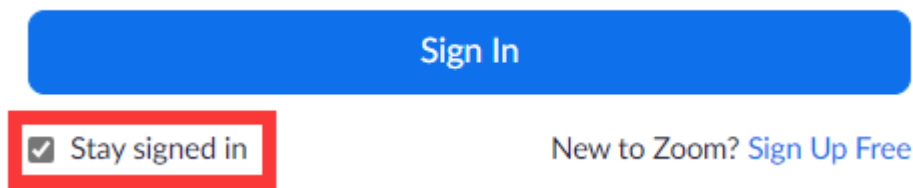
Please ensure you have the following ready before proceeding with the steps below.

1. Software: UiPath
2. Plugin: UiPath Firefox Plugin

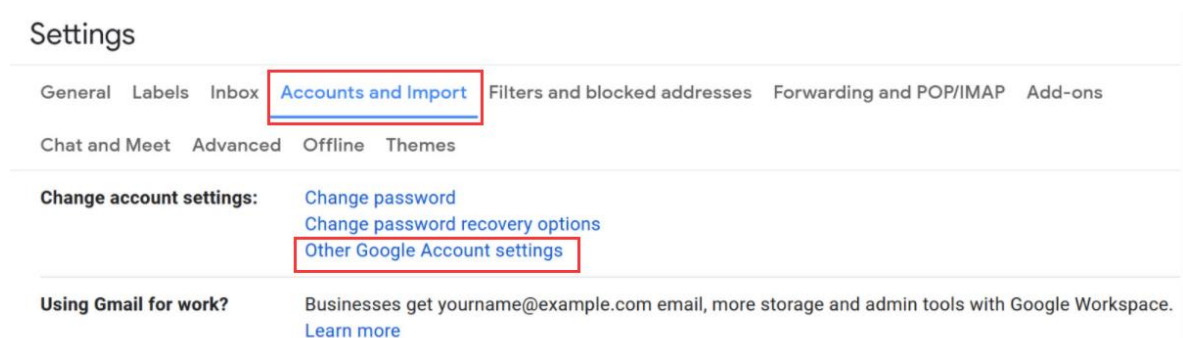
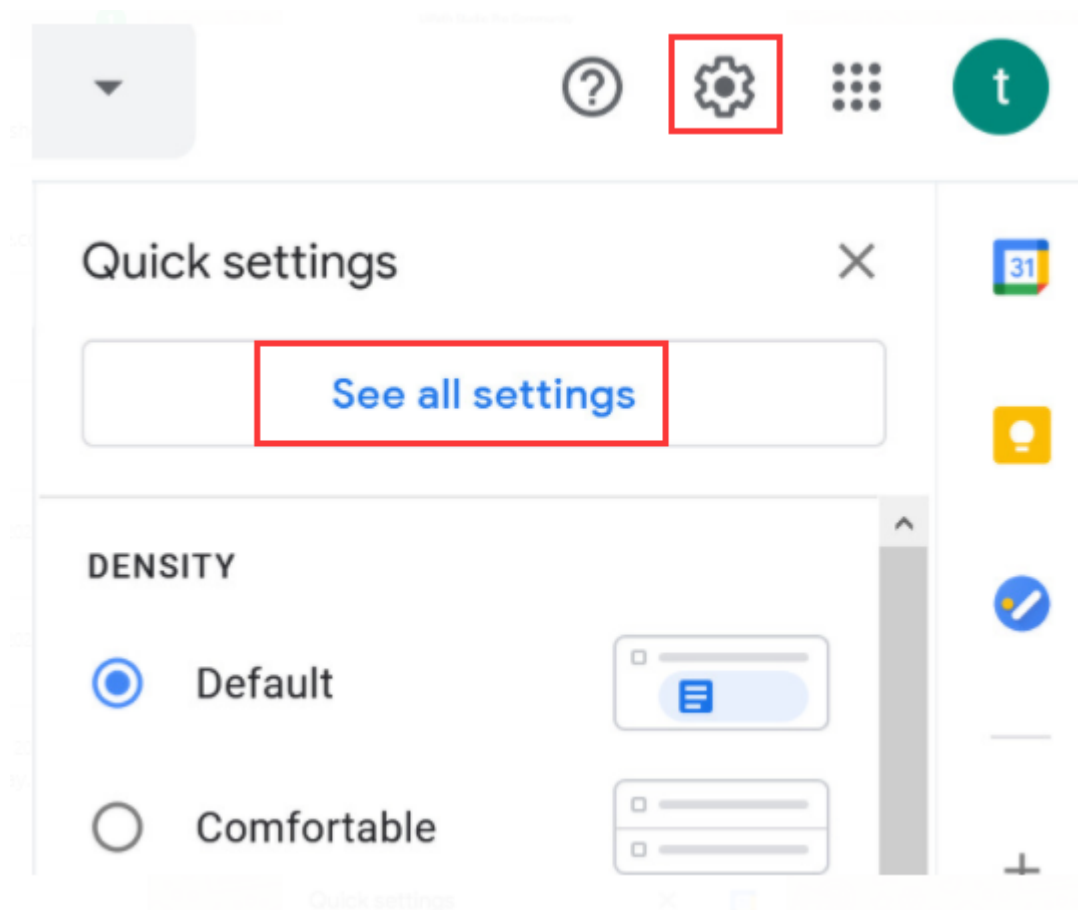


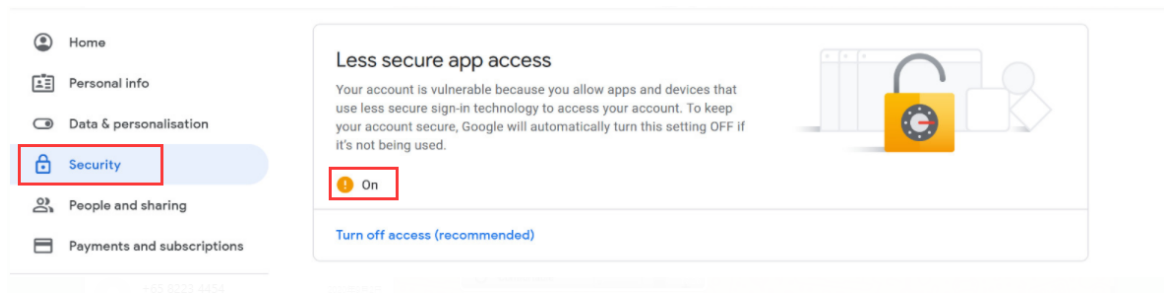
3. Zoom account should be signed in. Sign up for a zoom account (if not already available) and Sign in with your email address and password on [Sign In - Zoom](#), click the

“Stay signed in” box below the “Sign In” button.



4. Two dummy Gmail Accounts to simulate the ‘customer account’ and the ‘customer support team’ account.
5. Make sure that the ‘Less Secure App Access’ under ‘Security’ configuration of the Gmail is turned ‘ON’

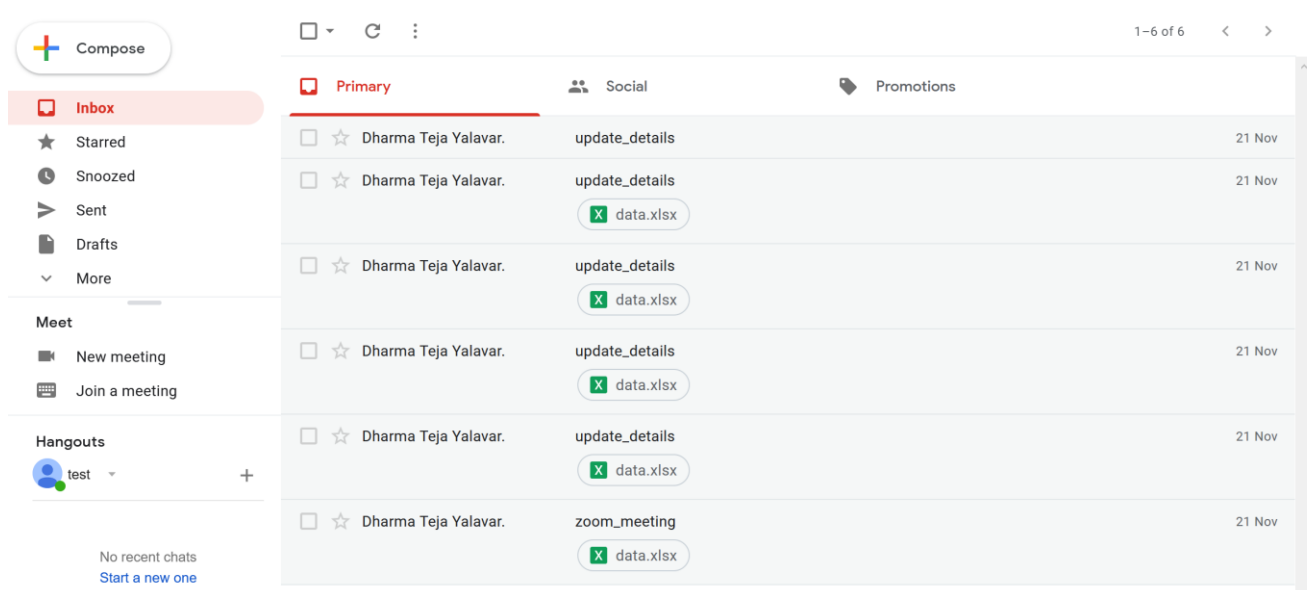




6. Send few mails from the 'customer account' to the 'customer support team' Gmail account which should follow the below mentioned requirements for the RPA bot to action upon.

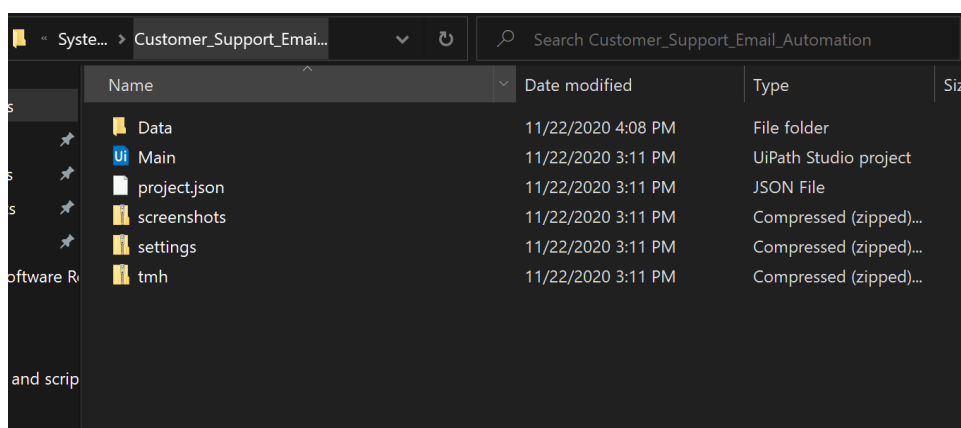
- **Email Requirements:**

- i. Subject line of the email sent should be either 'update_details' (to get the details updated in the database) or 'zoom_meeting' (to request for a zoom meeting appointment with the customer service executive). Other subject lines are ignored by the bot.
- ii. In order to request for updating details, fill the details in 'sheet 1' of the file 'data.xlsx' which is under the path 'SystemCode > Customer_Support_Email_Automation > Data' and use the same as attachment for the email being sent to the 'customer support team' account
- iii. Input data for the fields 'account_number', 'last_4_digits_on_your_card' and 'fin_number'. The data entered in these three fields should match the data in the database otherwise the customer would receive mail stating that the corresponding details are incorrect and hence the update cannot be done.
- iv. For the fields 'full_name', 'date_of_birth', 'address' and 'phone_number', edit the corresponding cells only if you wish to update the respective details in the database. If you do not wish to update, leave them as is (with 0 and NR)
- v. In order to request for a zoom meeting appointment, fill the details in 'sheet 2' of the same file 'data.xlsx'. Enter the date and time in the prescribed format.

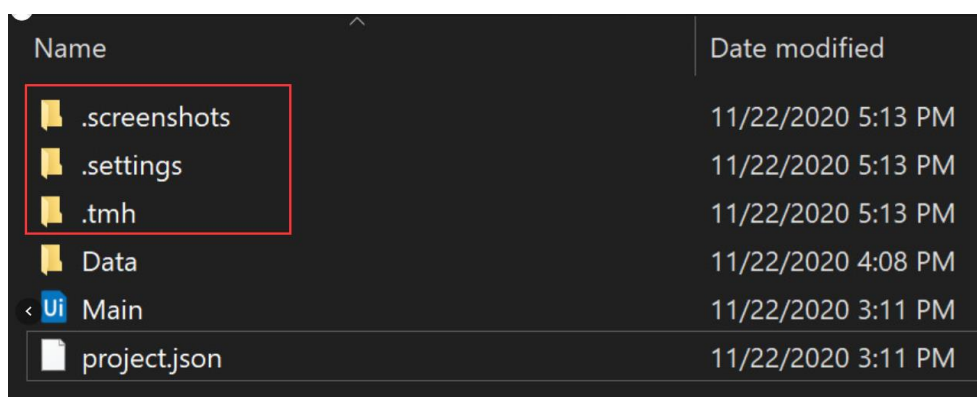


Email Automation Steps:

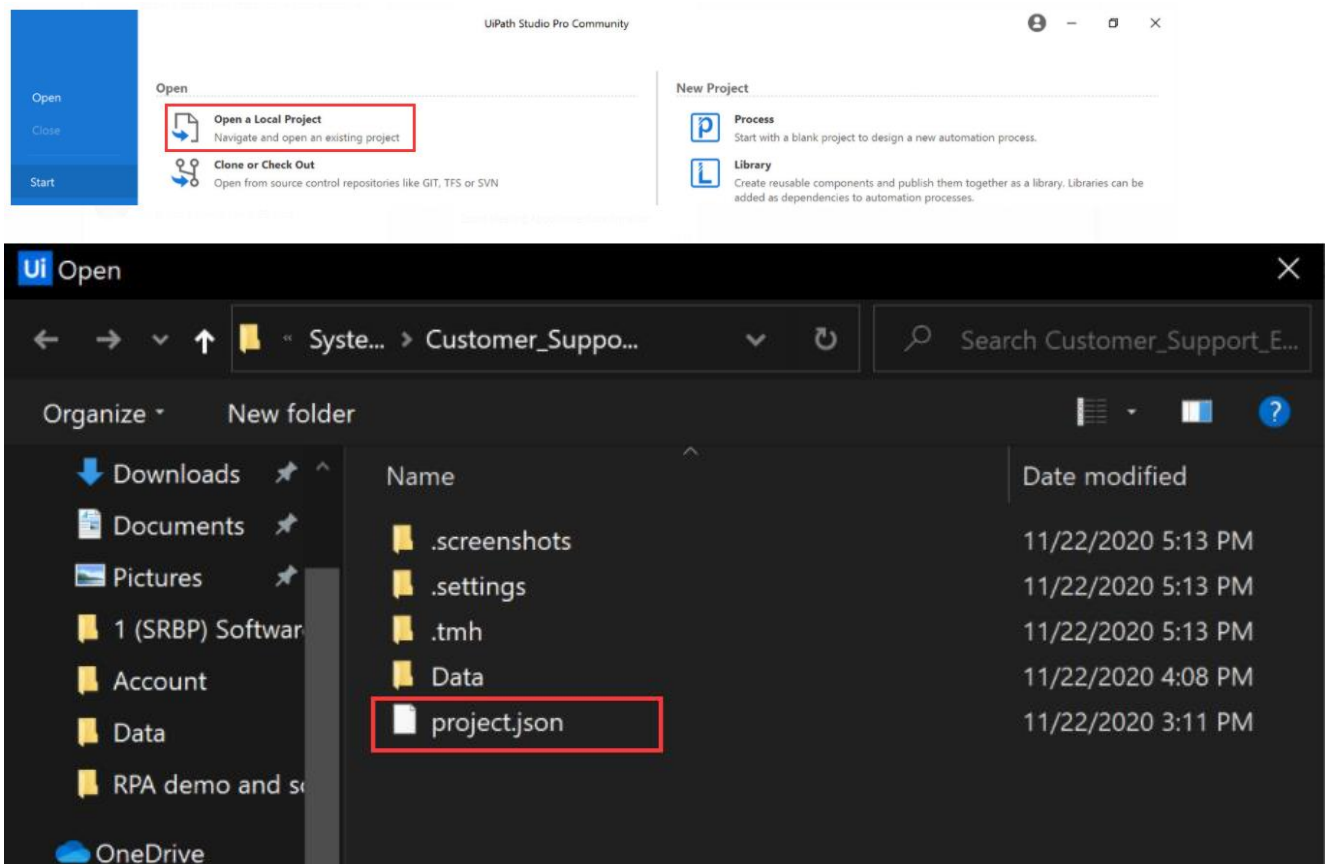
1. Go to the folder SystemCode > Customer_Support_Email_Automation



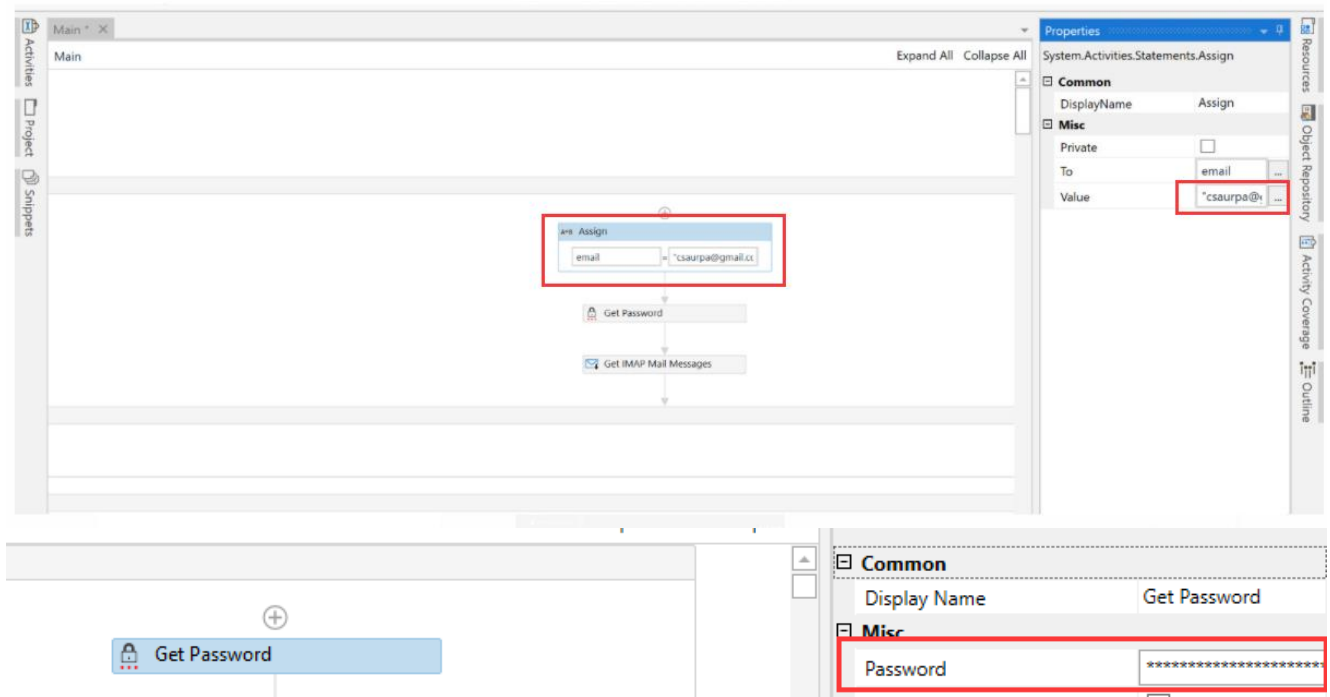
2. Extract the three Zip files (screenshots, settings, tmh) into the same folder



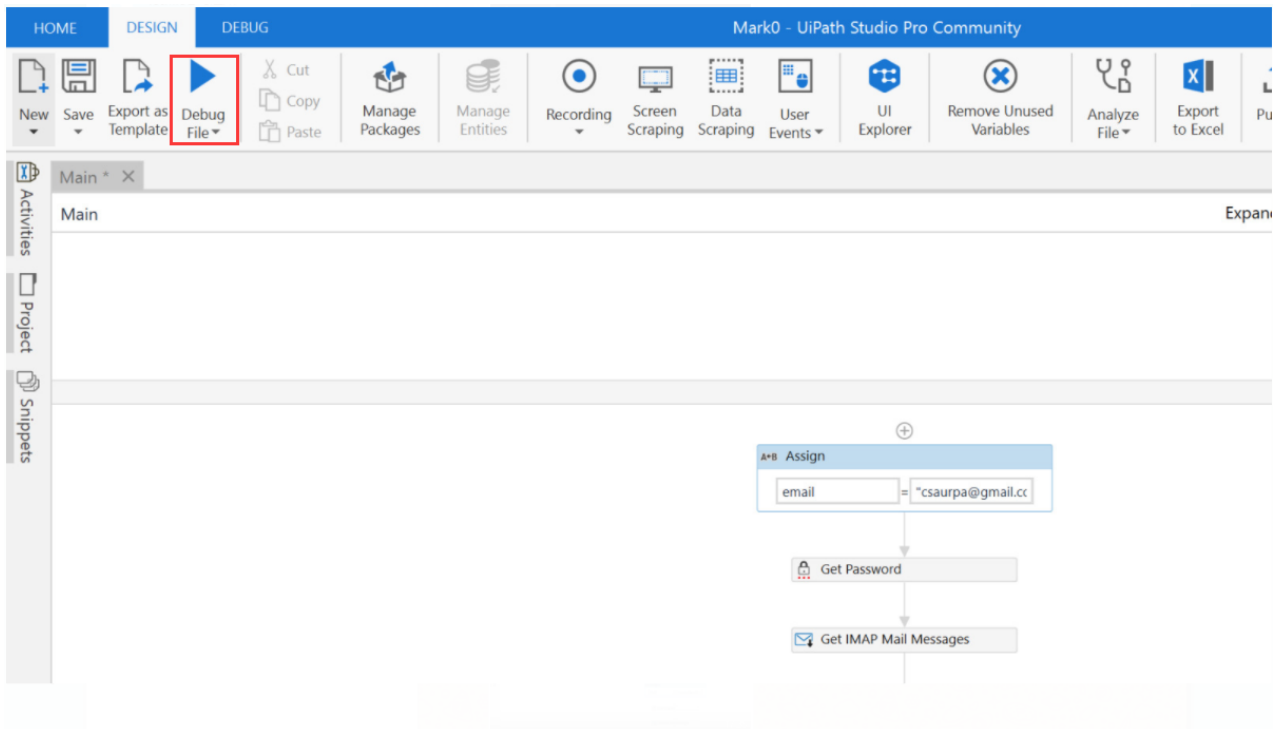
3. Open the project in UiPath



4. Update the 'Customer Support Team' Email id and password accordingly



5. Run the RPA bot to simulate and view the Customer Support Automation



6. Once the execution stops, you can verify the email responses sent by the bot to the customer account

Virtual Assistant Requirements:

- You are using python 3.5 and above (Version 3.7.6 used for development)
- Telegram app has been installed on your android or iOS smartphone.
- You have a registered account on <https://dialogflow.com/>

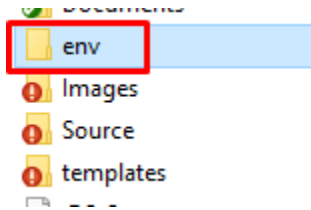
Virtual Assistant

Unzip into a location without whitespace eg. folder path names should not contain spaces like "D:/cc rpa/Project3a" it should be "D:/cc_rpa/Project3a"

Once you have installed python, open a command prompt and CD to the project root folder("<filepath>/Project3a/SystemCode").

1. Create a python env. Once created, you should see an 'env' folder in your project's folder

Use "python3 -m venv env" OR use "python -m venv env"



2. Activate the python env. Once activated, you should see (env) next to your command line.

For Mac: "source env/bin/activate"

For Windows: "env\Scripts\activate"

3. In the command prompt/terminal CD to the project folder "<your-file-path>/Project3a/SystemCode". Enter "pip install -r requirements.txt --user" OR "pip3 install -r requirements.txt". This will install all the required dependencies
4. Once installation is complete, type in "python app.py" and press enter. This will deploy your server locally on your pc.
5. Navigate to "<your-file- path>/Project3a/SystemCode/Account" and open cmd and type "npm install" and enter, for first time installation.
6. To start server for the Customer Care UI, run the following in cmd, "npm run start".
7. Download and install ngrok. Open command prompt in the location where ngrok is downloaded, enter "ngrok http localhost:5000". Copy the https link, it should look something like this <https://d7edbebd2187.ngrok.io>.
8. Go to <https://dialogflow.cloud.google.com/> and log in with your account. Proceed to import the dialogflow agent on dialogflow. To do this, first create a new agent and give it a name e.g., " CC_RPA_Agent". Once created click on the settings gear icon next the agent name and click the "Export & Import" tab. Click "Import from Zip" and select the CC_RPA_Agent.zip. Type "IMPORT" into the text box and click "IMPORT".
9. Once the agent has been imported and training is done, click on the "Fulfillment" option on the left menu bar (*figure 7.1*). Enable the webhook (If it is not enabled) and copy and paste the ngrok https link on the URL field. Scroll to the bottom and click save. Give it some time to save your settings.

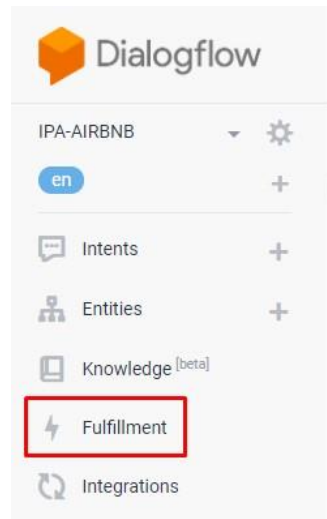


Figure 7.1 Fulfillment option

⚡ Fulfillment

Webhook

ENABLED ☒

Your web service will receive a POST request from Dialogflow in the form of the response to a user query matched by intents with webhook enabled. Be sure that your web service meets all the [webhook requirements](#) specific to the API version enabled in this agent.

URL*	<input type="text" value="https://d7edbebd2187.ngrok.io"/>	
BASIC AUTH	<input type="text" value="Enter username"/>	<input type="text" value="Enter password"/>
HEADERS	<input type="text" value="Enter key"/>	<input type="text" value="Enter value"/>
	+ Add header	
SMALL TALK	<input type="checkbox"/> Disable webhook for Smalltalk	

Figure 7.2 Enable and setting webhook

10. Set up the link between telegram and dialogflow. From dialogflow, click Integrations. Then, check the Telegram box. A pop-up will appear, enter the bot_token key and click start. This will integrate the bot and dialogflow together.

Bot Token key: 1455843468:AAG0kVfnJxpjNP6k3kUdJKgyX3RUmzc-Kyk

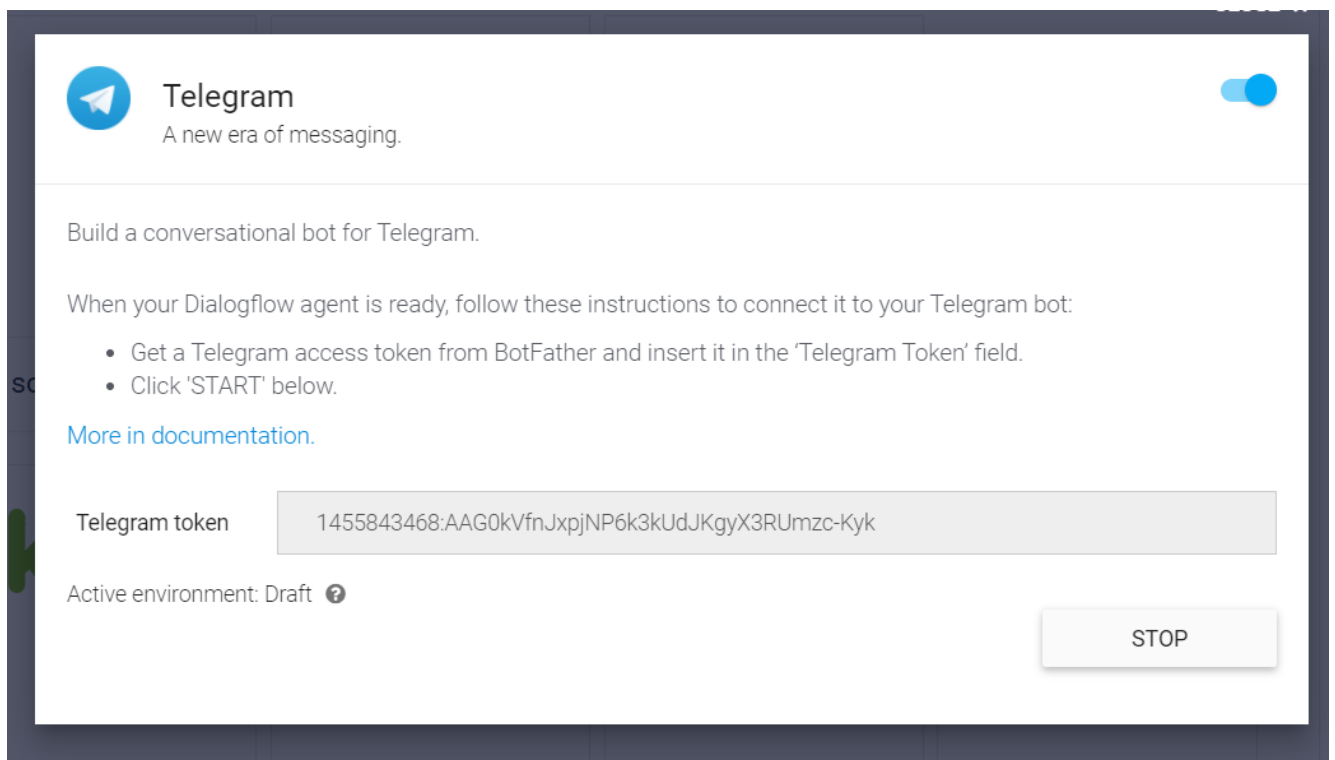


Figure 8.1 Telegram option

11. Now open your telegram app from the smartphone. From the search

bar of your chat page, type "CCRPAbot" click on "CCRPAbot". A chat window will be created with the bot, click 'Start'.

Temporarily the credentials for accessing gdrive is in credentials.json which is linked to the google forms owner. The credentials for google cloud vision api is saves as lak_cred.json. These credentials may expire and need to be replaced with valid credentials.

For testing -

Use Account number - [3757685891](#) Personal Security Number - 3456 and upload the below image saved in your local



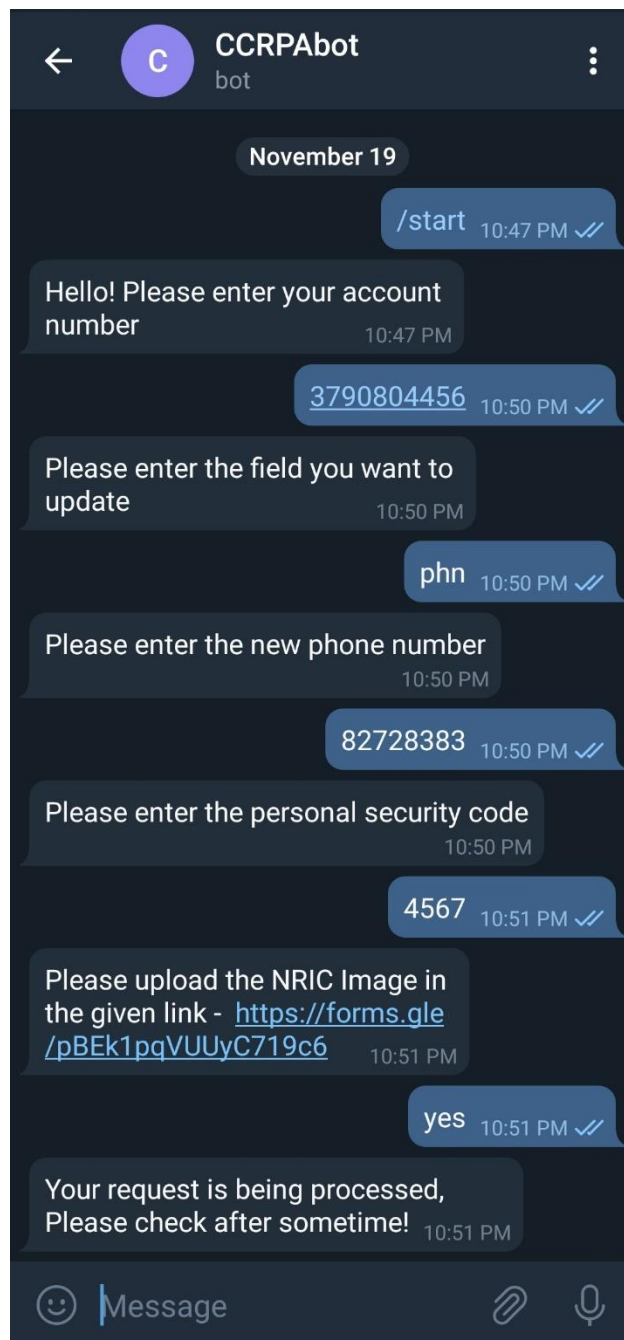


Figure 11.1 Telegram conversation