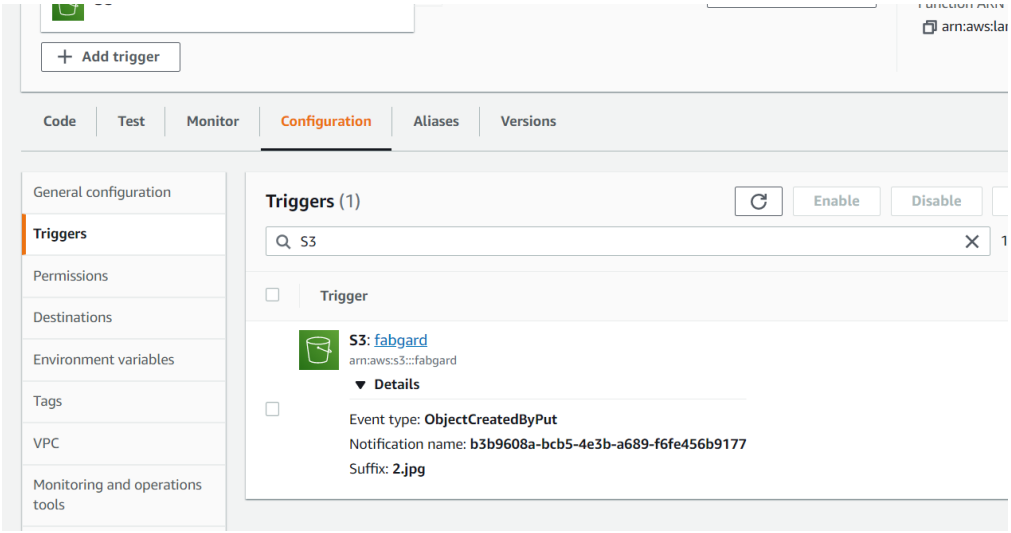
Github link -https://github.com/arubittu/fabguard

Configuring lambda_function.py

- Make a lambda function on aws lambda , attach the given layer - 'fgboto' having all libraries. Fgboto has to be uploaded on a S3 bucket.
  https://drive.google.com/drive/folders/1c0BsFw8s1o6TNdolMeOW5Eecau8NV0nI?usp=sharing
  Download from above drive link^ (same given on github readme)

- Ensure that you have set up nat gateway access and proper internet gateways required to access internet capabilities - from this link -https://blog.theodo.com/2020/01/internet-access-to-lambda-in-vpc/
- On line 29 of **lambda_function.py** you have to give the name of the google api secret file. Also you need to upload this file as a zip file in lambda editor so it can be accessed by aws.
- On line 172 you have to enter your mongodb connection link, obtained from mongodb website.
- **Configuring s3 event-** The **lambda_function.py** activated whenever a file is uploaded on s3 by the user. So S3 has to be selected as an event trigger . currently i have added the following suffix also. So the function gets activated when a filename with given suffix is uploaded.
- 

Configuring mongodb

- Make a mongodb account
- Create a table
- Create collection under the above table
- Get the api url and put it at line 172 in **lambda_function.py**

RPA.py

- At line 9 and 10 , change the amazon api keys and put your own. This is required if you want to access aws resources from code editor like vscode.

```
rpa.py > ...
2   from selenium import webdriver
3   import boto3
4   import pymongo
5   from pymongo import MongoClient
6   import time
7   #from boto3.s3.key import Key
8   ################ aws access keys
9   Access_Key_ID='AKIA4RDZJU3SBN7UTW3Y'
10  Secret_Access_Key='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'
11  #################
12  # uploading image on aws s3 after getting it from user
13  def aws_call(Access_Key_ID,Secret_Access_Key,path,name='test_img.jpg'):
14      s3 = boto3.client(
15          's3',
16          aws_access_key_id=Access_Key_ID,
17          aws_secret_access_key=Secret_Access_Key
18          )
19          #aws_session_token=SESSION_TOKEN
20      rs=boto3.resource('s3')
21      response = s3.list_buckets()
22      s3.upload_file(path, 'fabgard', name)
```

- At line 22, in s3.upload_file function, the 2nd argument is the name of the s3 bucket. So put it accordingly.
- Again in line 31 you have to put the applicable mongodb url to access the database
- I have used selenium as web automation tool. Which after you download the library and set it up on your pc , change the path at line 46, accordingly put location of your path.
- Line 49 is where the website link is written. This can be changed to any other link on which rpa is to be performed.

Streamlit_app.py

- Change line 14 /15 and put your aws keys

```
12
13  ###################### aws keys
14  Access_Key_ID='AKIA4RDZJU3SBN7UTW3Y'
15  Secret_Access_Key='SoQx5BuC69n9hAGabOuKTZoAi5dhPifTdklwcldA'
16  ####################
17
```

- Line 44 and 39 is where i have saved the user obtained images from website to local computer with the given names.

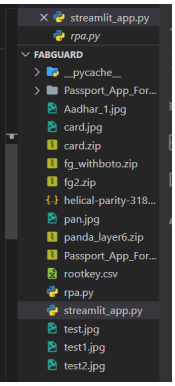
```
file2 = st.file_uploader('Upload An pan Image')
# operate on images obtained from file
if file1 and file2:
    # preprocess images
    img1 = Image.open(file1)
    resized_image1 = img1.resize((678,381))
    resized_image1.save("test1.jpg")
    st.image(resized_image1)

    img2 = Image.open(file2)
    resized_image2 = img2.resize((995,638))
    resized_image2.save("test2.jpg")
    st.image(resized_image2)

    # upload files on aws
    aws_call(Access_Key_ID,Secret_Access_Key,'test1.jpg',name='test_img.jpg')
    aws_call(Access_Key_ID,Secret_Access_Key,'test2.jpg',name='test_img2.jpg')

    time.sleep(6)

    # get recent result from mongo db
    item=get_mongo()
```



- Line 48 and 49 is where i am uploading the two images obtained from user from website to amazon s3 bucket. And with the given names. So once these images are uploaded on s3. It will trigger an aws event which will activate **lambda_function.py.** Which will perform ocr on it in the cloud and upload details on mongodb database.

```
    # upload files on aws
    aws_call(Access_Key_ID,Secret_Access_Key,'test1.jpg',name='test_img.jpg')
    aws_call(Access_Key_ID,Secret_Access_Key,'test2.jpg',name='test_img2.jpg')

    time.sleep(6)
```

- I have put a sleep timer of 6 seconds since that is the average time it takes for the lambda function on the cloud to run and upload results on mongodb. So try experimenting with this time and see what works.