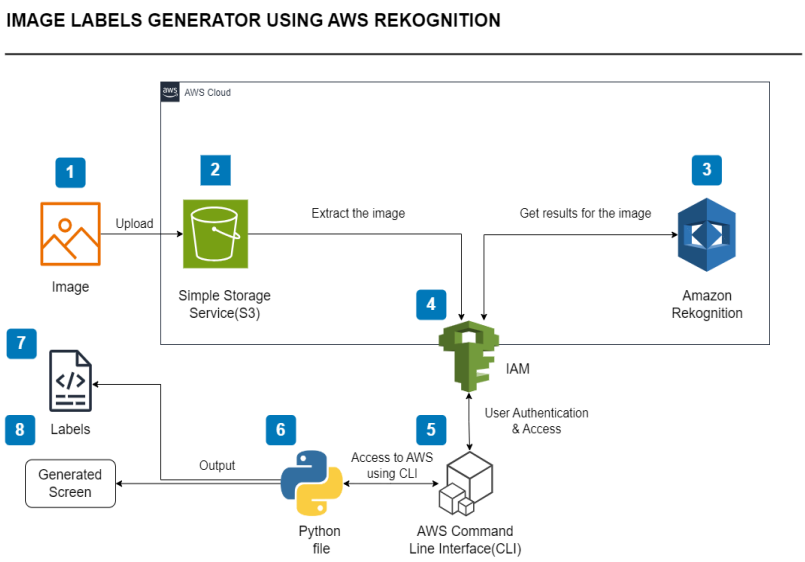
BUILD AN IMAGE LABELS GENERATOR USING AMAZON REKOGNITION

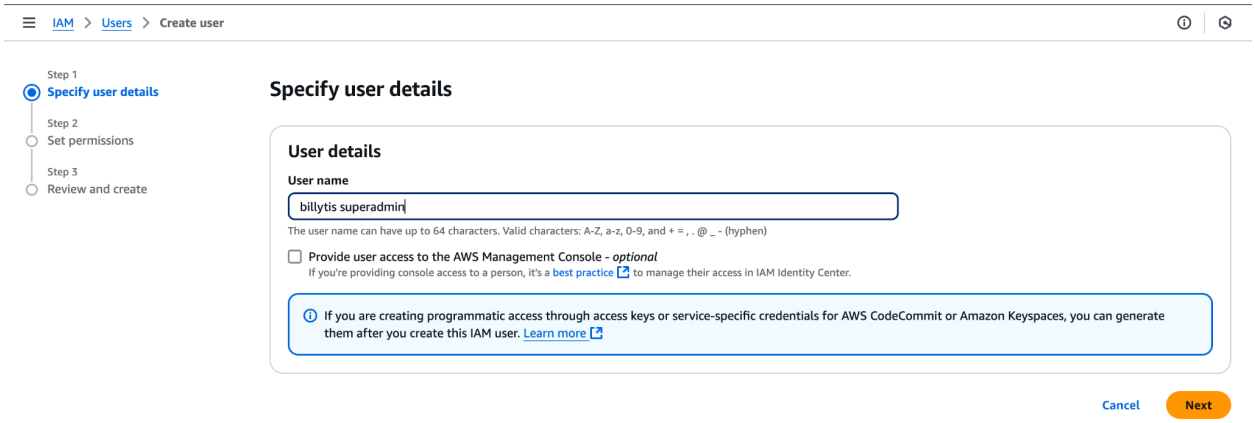
Here is the architectural plan of my application:



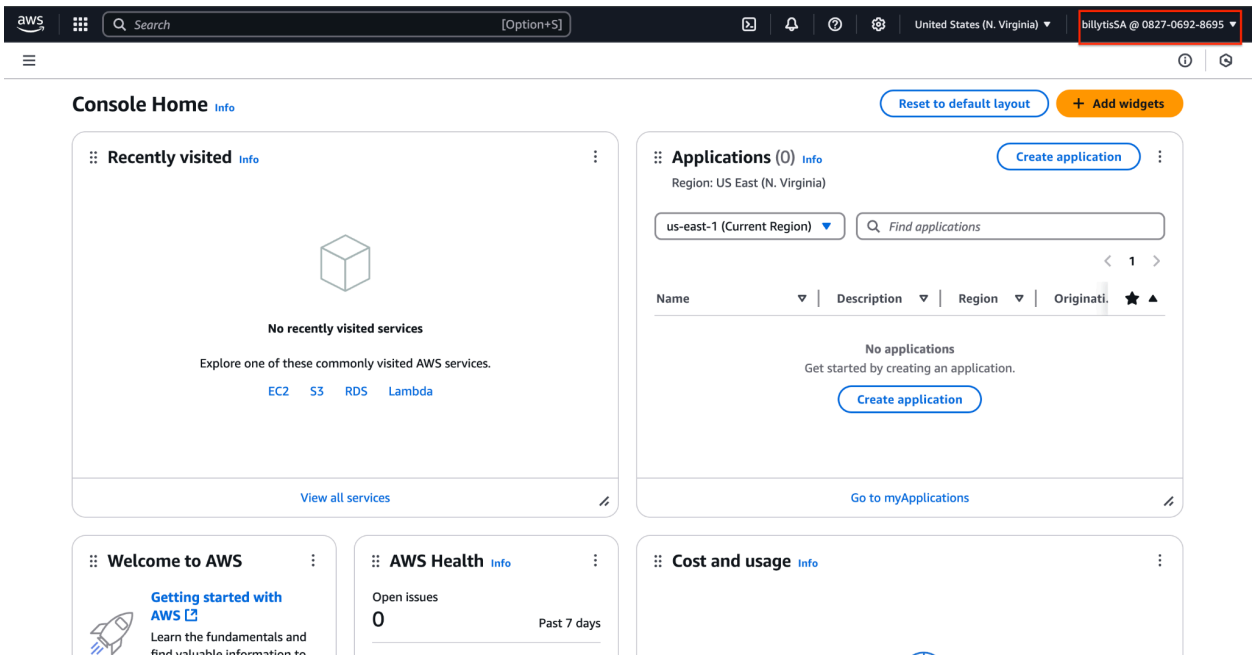
STEP 1:

Create an Admin user

Before i started my project i created an user admin via IAM in aws console so i don't have to use my root account for security purposes:

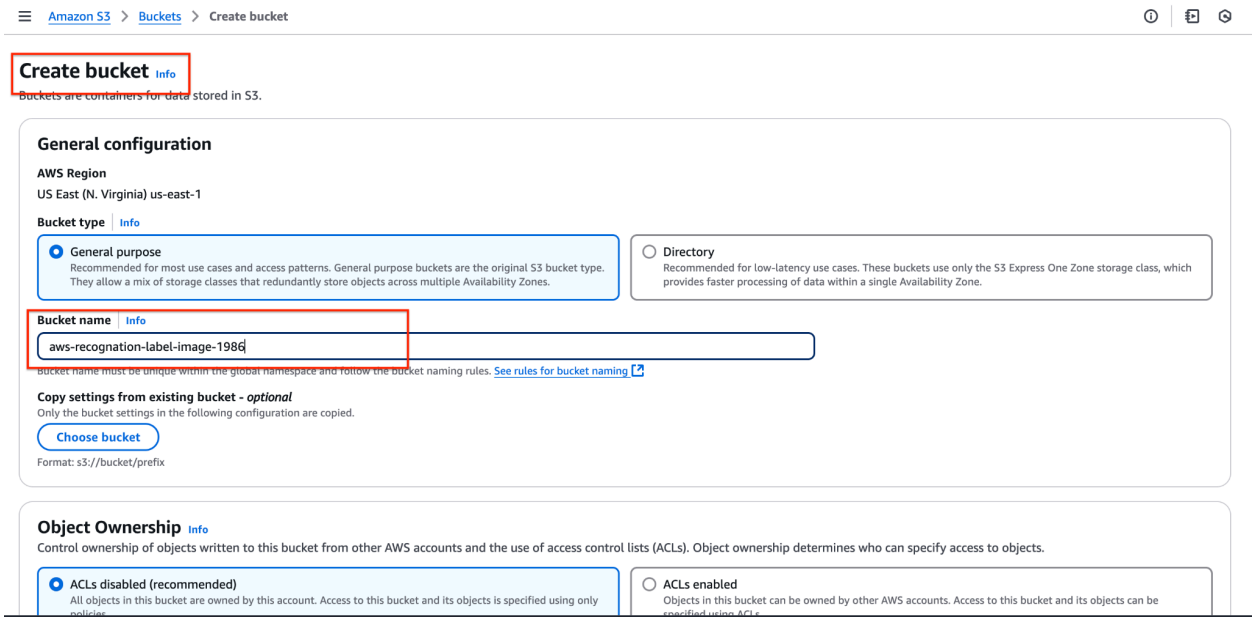


Here i am logged into the aws console as an admin:

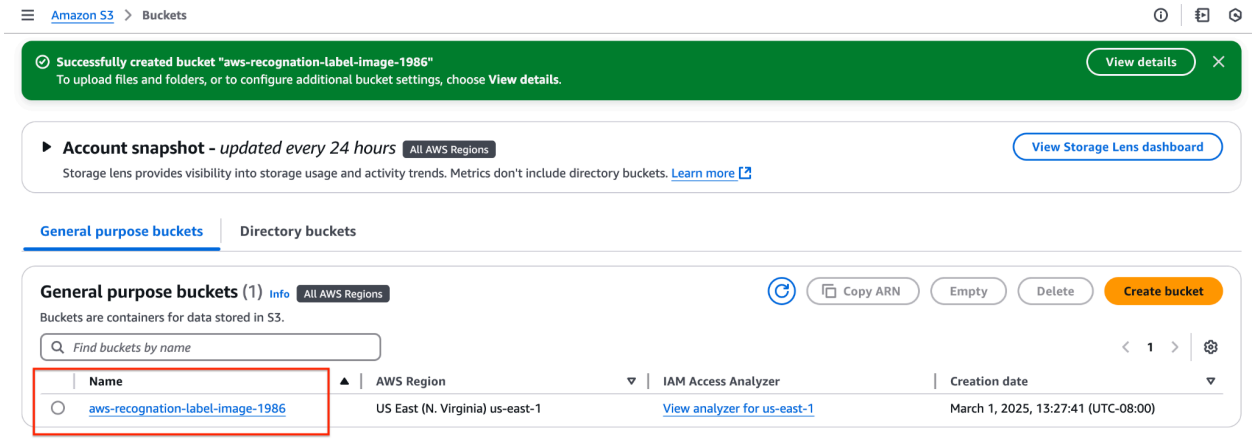


STEP 2:

Create a S3 bucket and upload images

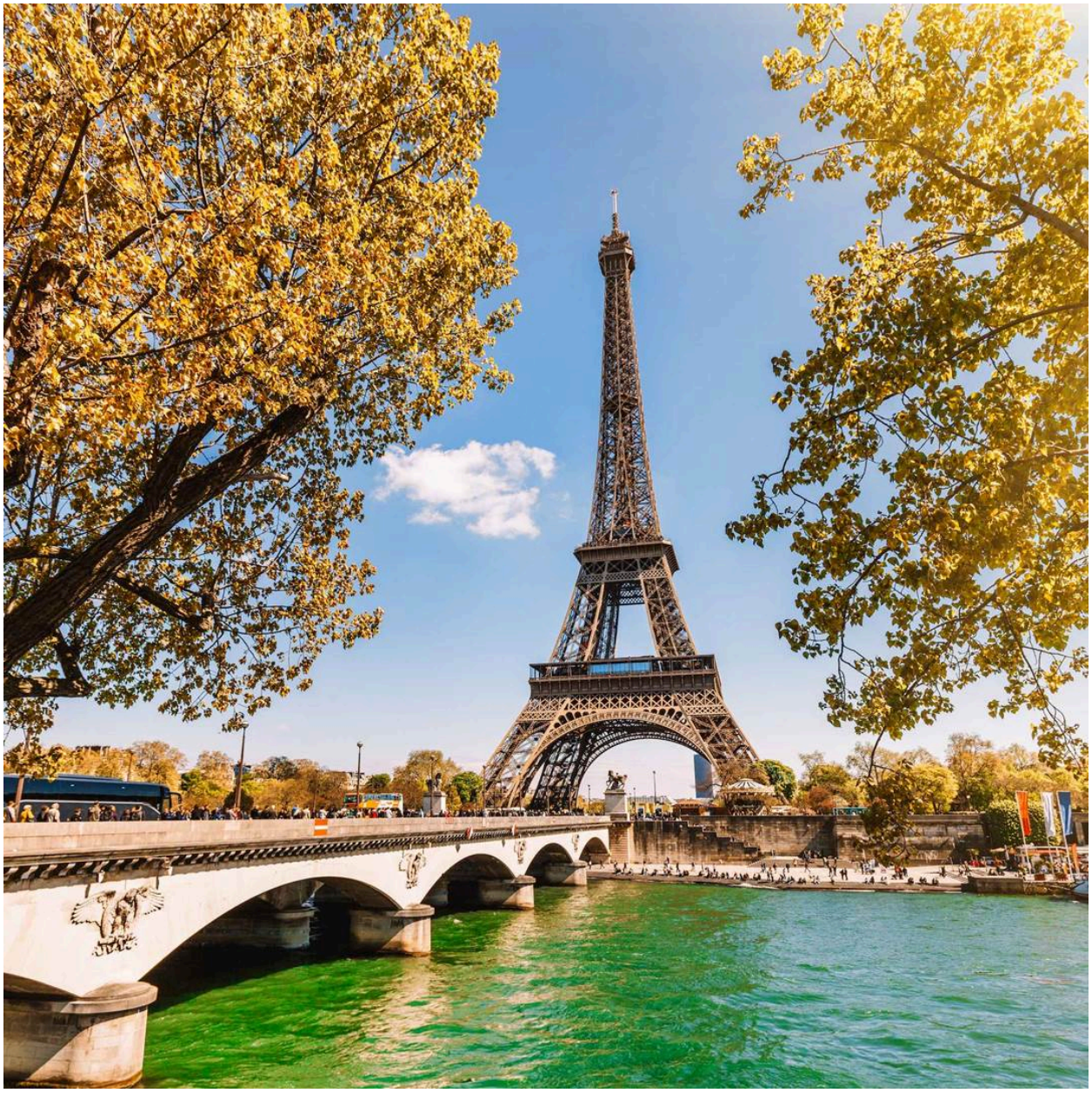


Bucket is created



Now it's time to upload the image we want to analyse:

These are pictures that i uploaded:



Amazon S3 > Buckets > aws-recognation-label-image-1986

aws-recognation-label-image-1986 Info

Objects | Metadata | Properties | Permissions | Metrics | Management | Access Points

Objects (0)

Copy S3 URI | Copy URL | Download | Open | Delete | Actions | Create folder | Upload

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Find objects by prefix

1

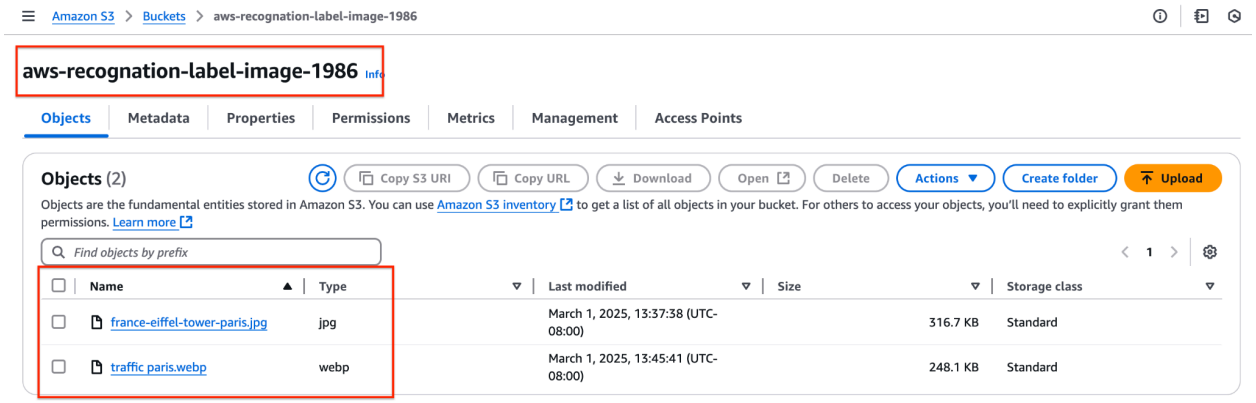
Name	Type	Last modified	Size	Storage class

No objects

You don't have any objects in this bucket.

Upload

Images have been uploaded



STEP 3

Installing the AWS Command line interface(CLI) ON MAC

a) I had to download the package via this command line (*the most challenging was to find the latest version of that package)*:

curl "https://awscli.amazonaws.com/AWSCLIV2.pkg" -o "AWSCLIV2.pkg"

b) Install the package:

sudo installer -pkg AWSCLIV2.pkg -target /

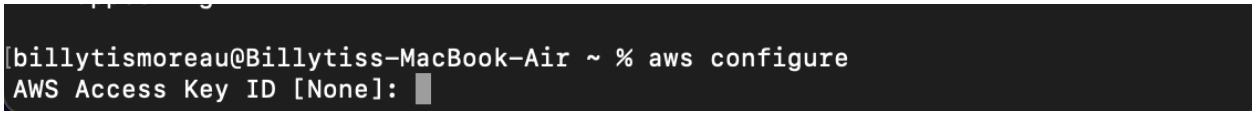
c) Verify if the package got installed:

aws --version

Everything been successfully installed

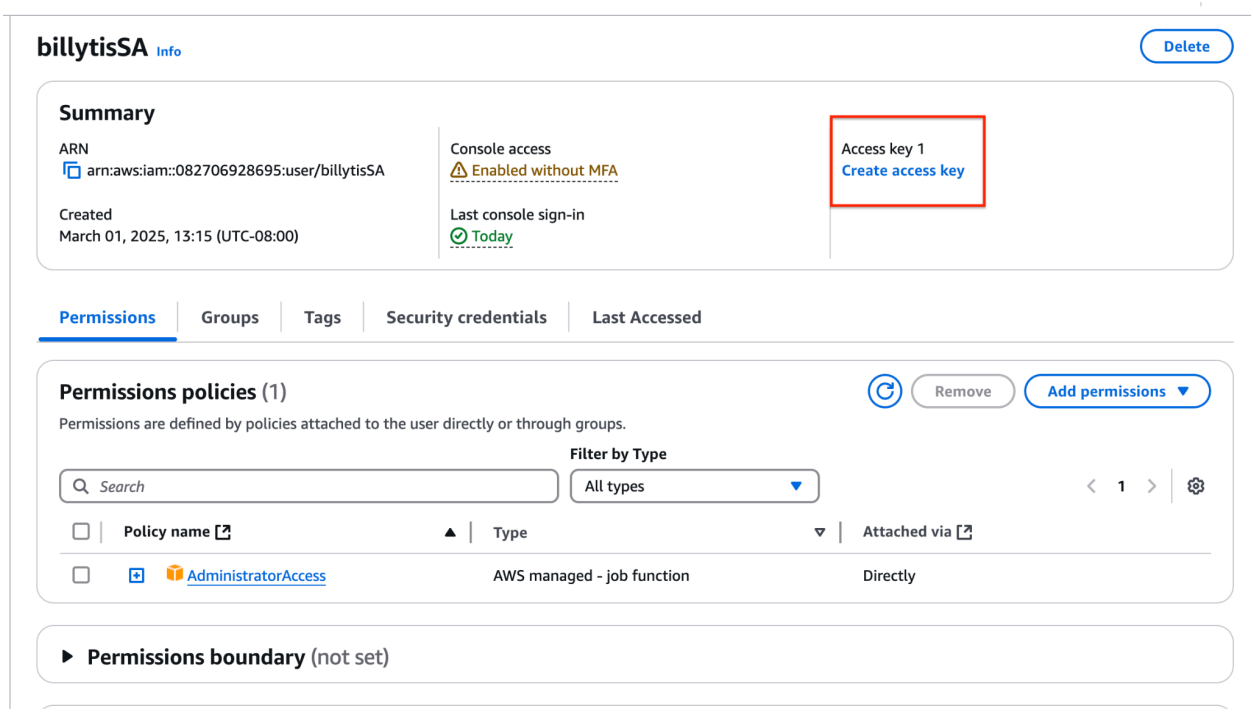
d) Configuration of AWS CLI:

aws configure

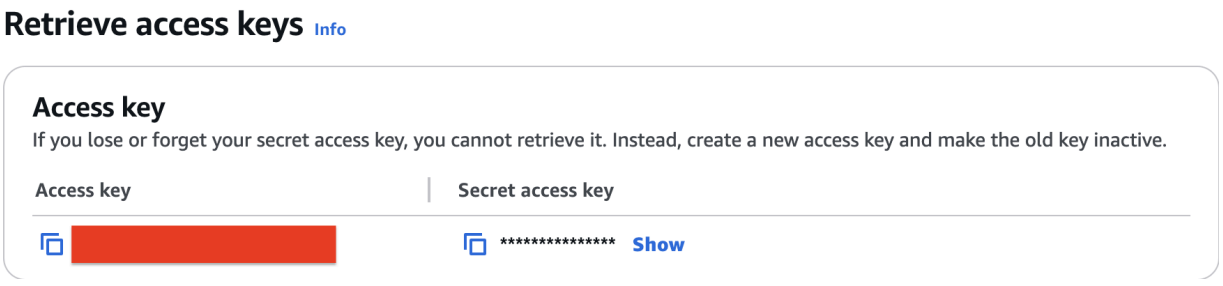


I had to create an access key and secret access key so i can get access to my aws account via AWS CLI.

From my root account I created an access key and secret access key for my Admin user so I can connect to my account via my terminal.



Access key and secret access key created:



My AWS CLI has been configured

```
billytismoreau@Billytiss-MacBook-Air ~ % aws configure
AWS Access Key ID [None]: [redacted]
AWS Secret Access Key [None]: [redacted]
Default region name [None]: us-est-1
Default output format [None]:
```

For the next step I wrote the python code for extracting pictures from S3 bucket and applied **detect_labels** operation from Rekognition to generate the labels with their confidence score.

STEP 4

Import Libraries

I had to add these libraries (**boto3 and matplotlib)** to my terminal but i had to make sure i have the latest version of python:

By typing these commands on my terminal:

python3 --version
pip3 --version

And then install the library with this command:

pip3 install boto3 matplotlib

And then to finish i had to check if Boto3 and Matplotlib are installed by running this command:

python3 -c "import boto3; print(boto3.__version__)" python3 -c "import matplotlib; print(matplotlib.__version__)"

STEP5

Define Functions

a) I define a function called **detect_labels**
b) I created a Rekognition client using boto3
c) I used the detect_labels method of the Rekognition client to detect labels in the given photo
d) I printed the detected labels along with their confidence levels
e) I loaded the image from the S3 bucket using boto3 and PIL

f) I used matplotlib to display the image and draw bounding boxes around the detected objects

DETECT_LABELS_FUCTION CODE:

detect_labels_fuction.py

```python
def detect_labels(photo, bucket):
    # Create a Rekognition client
    client = boto3.client('rekognition')

    # Detect labels in the photo
    response = client.detect_labels(
        Image={'S3Object': {'Bucket': bucket, 'Name': photo}},
        MaxLabels=10)

    # Print detected labels
    print('Detected labels for ' + photo)
    print()
    for label in response['Labels']:
        print("Label:", label['Name'])
        print("Confidence:", label['Confidence'])
        print()

    # Load the image from S3
    s3 = boto3.resource('s3')
    obj = s3.Object(bucket, photo)
    img_data = obj.get()['Body'].read()
    img = Image.open(BytesIO(img_data))

    # Display the image with bounding boxes
    plt.imshow(img)
    ax = plt.gca()
    for label in response['Labels']:
        for instance in label.get('Instances', []):
            bbox = instance['BoundingBox']
            left = bbox['Left'] * img.width
            top = bbox['Top'] * img.height
            width = bbox['Width'] * img.width
            height = bbox['Height'] * img.height
            rect = patches.Rectangle((left, top), width, height, linewidth=1, edgecolor='r',
facecolor='none')
            ax.add_patch(rect)
            label_text = label['Name'] + ' (' + str(round(label['Confidence'], 2)) + '%)'
            plt.text(left, top - 2, label_text, color='r', fontsize=8, bbox=dict(facecolor='white',
alpha=0.7))
    plt.show()

    return len(response['Labels'])
```

Next, I wrote a main function to test my detect_labels function. I specify a sample photo and bucket name, then called the detect_labels function with these parameters.

MAIN_FUCTION CODE:

```
def main():
    photo = 'image_file_name'
    bucket = 'bucket_name'
    label_count = detect_labels(photo, bucket)
    print("Labels detected:", label_count)

if __name__ == "__main__":
    main()
```

Once the code was written I changed my ‘image_file_name’ and ‘bucket_name’ to my actual configured naming.

```
def main():
    photo = 'france-eiffel-tower-paris.jpg'
    bucket = 'aws-recognation-label-image-1986'
    label_count = detect_labels(photo, bucket)
    print("Labels detected:", label_count)

if __name__ == "__main__":
    main()
```

STEP 6

Running your Project

I created a file Named detect_labels.py on my MacOs terminal, then copy/Paste the code in that file, saved it, exited out. And ran that command: **python3 detect_labels.py**

Here is the final result of the

```
~ $ python3 detect_labels.py
Detected labels for france-eiffel-tower-paris.jpg

Label: Architecture
Confidence: 96.84970092773438

Label: Building
Confidence: 96.84970092773438

Label: Eiffel Tower
Confidence: 96.84970092773438

Label: Landmark
Confidence: 96.84970092773438

Label: Tower
Confidence: 96.84970092773438

Label: Bridge
Confidence: 77.93797302246094

Label: Bus
Confidence: 70.84432220458984

Label: Transportation
Confidence: 70.84432220458984

Label: Vehicle
Confidence: 70.84432220458984

Label: Person
Confidence: 62.174354553222656

Labels detected: 10
~ $
```

END

CHALLENGES DURING THIS PROJECT

The challenges that I faced during this project was when I ran the command **python3 detect_labels.py** I had an error message because the region of the bucket had to be explicit in my code and I had a syntax error at line 46.