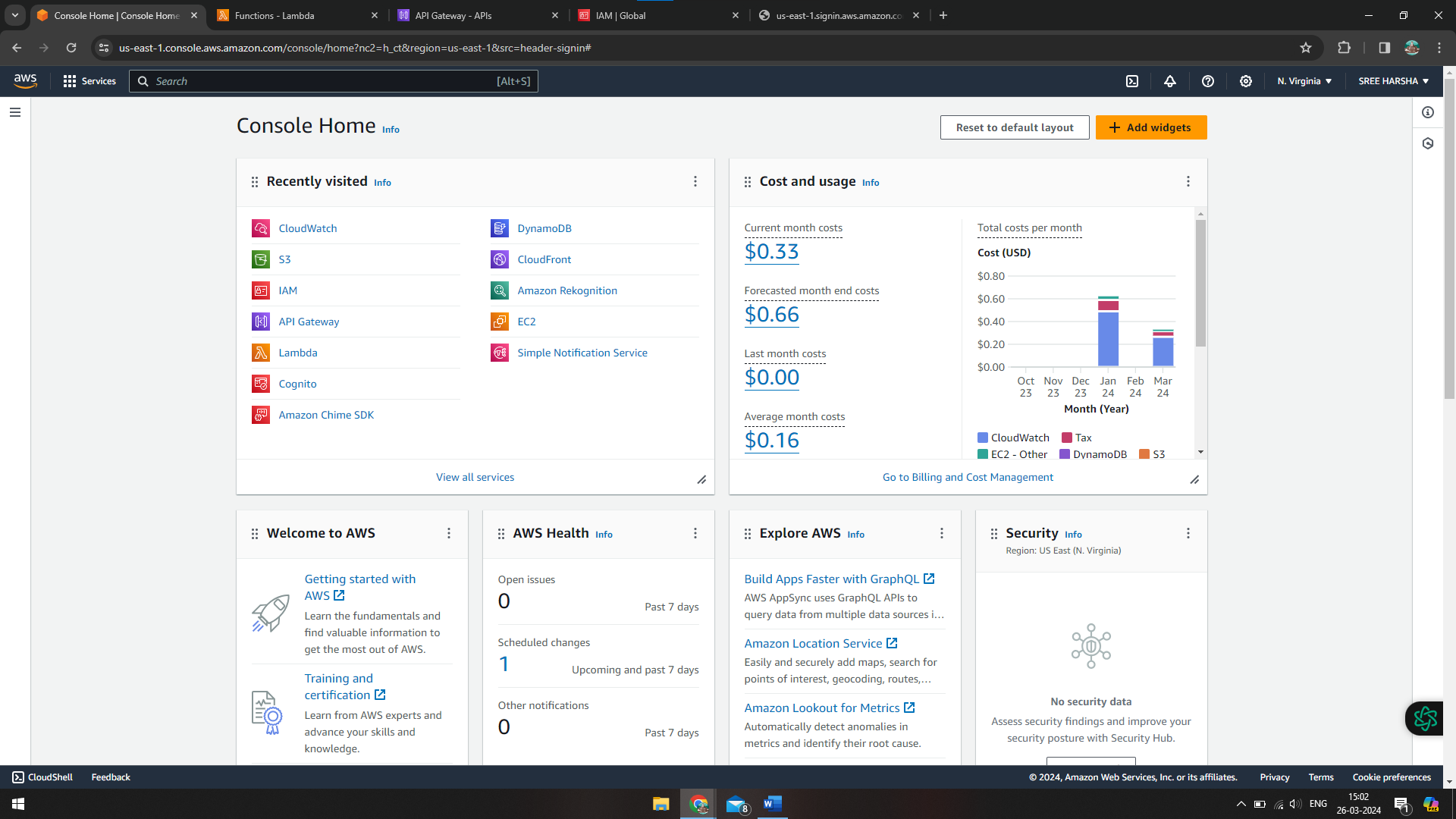
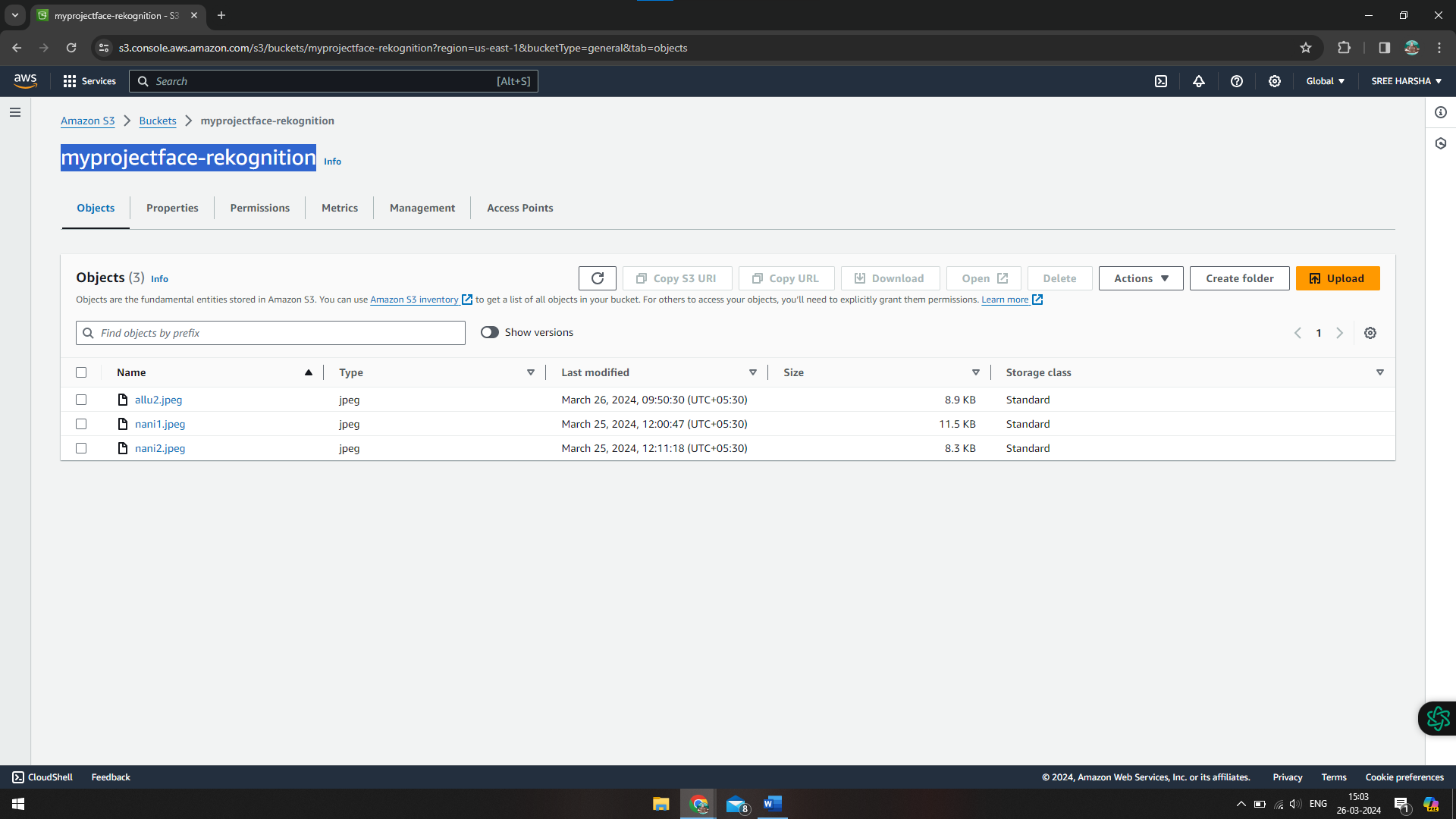
**AWS IMAGE REKOGNIZATION USING FIVE HUMANFACE(CLOUDWATCH,LAMBDA,S3)**

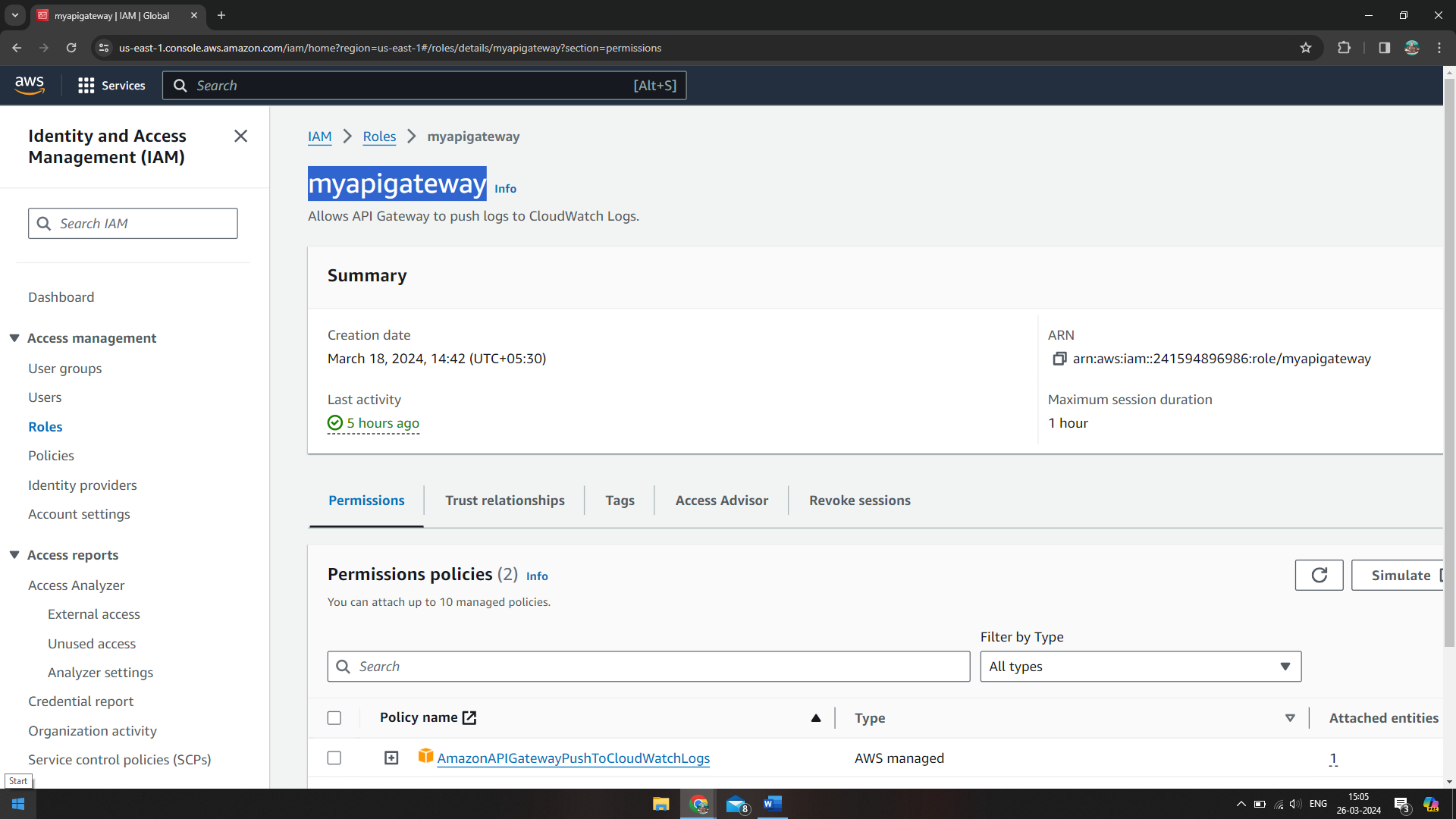
AWS MANAGEMENT CONSOLE:



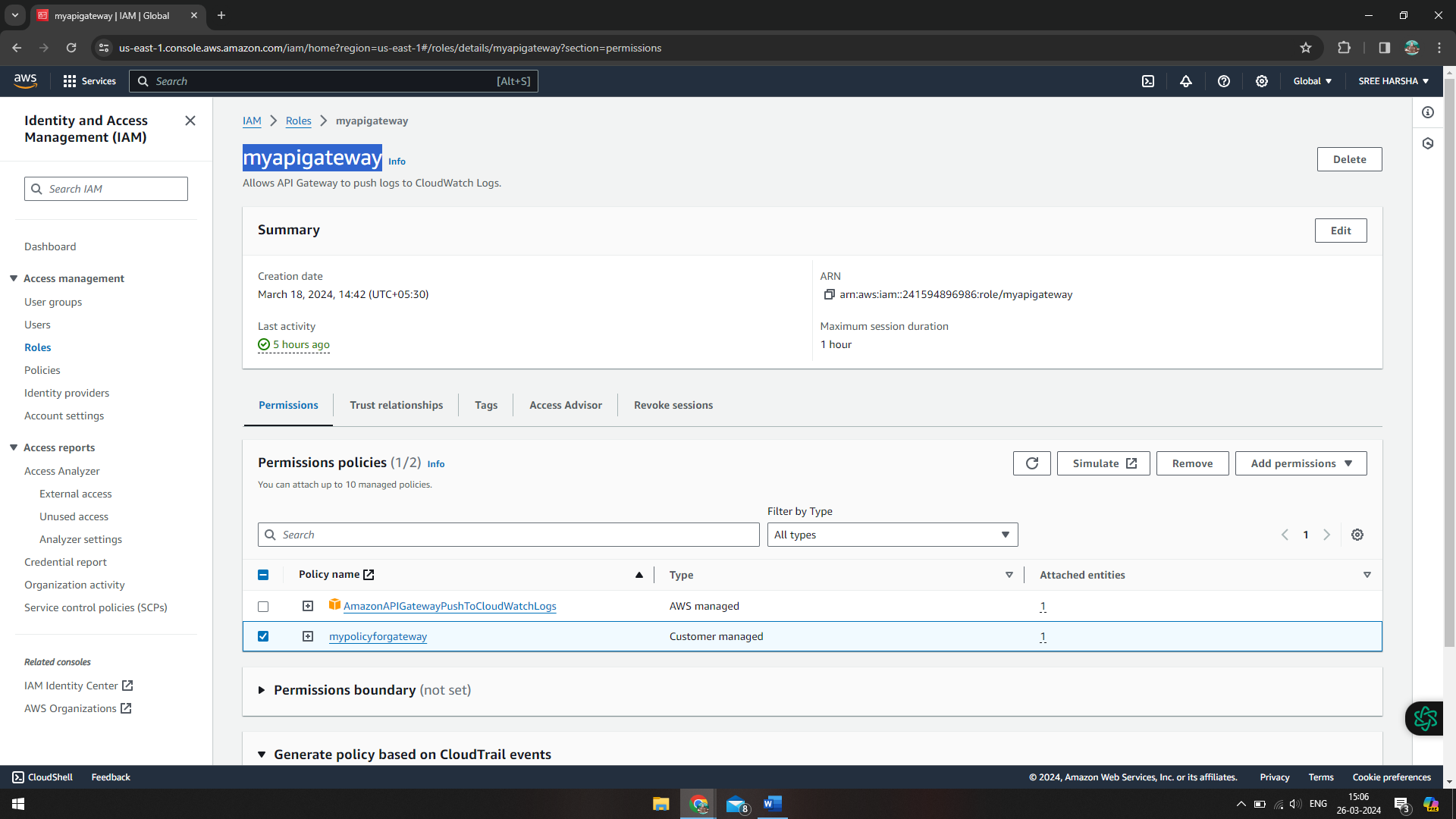
S3 BUCKET:



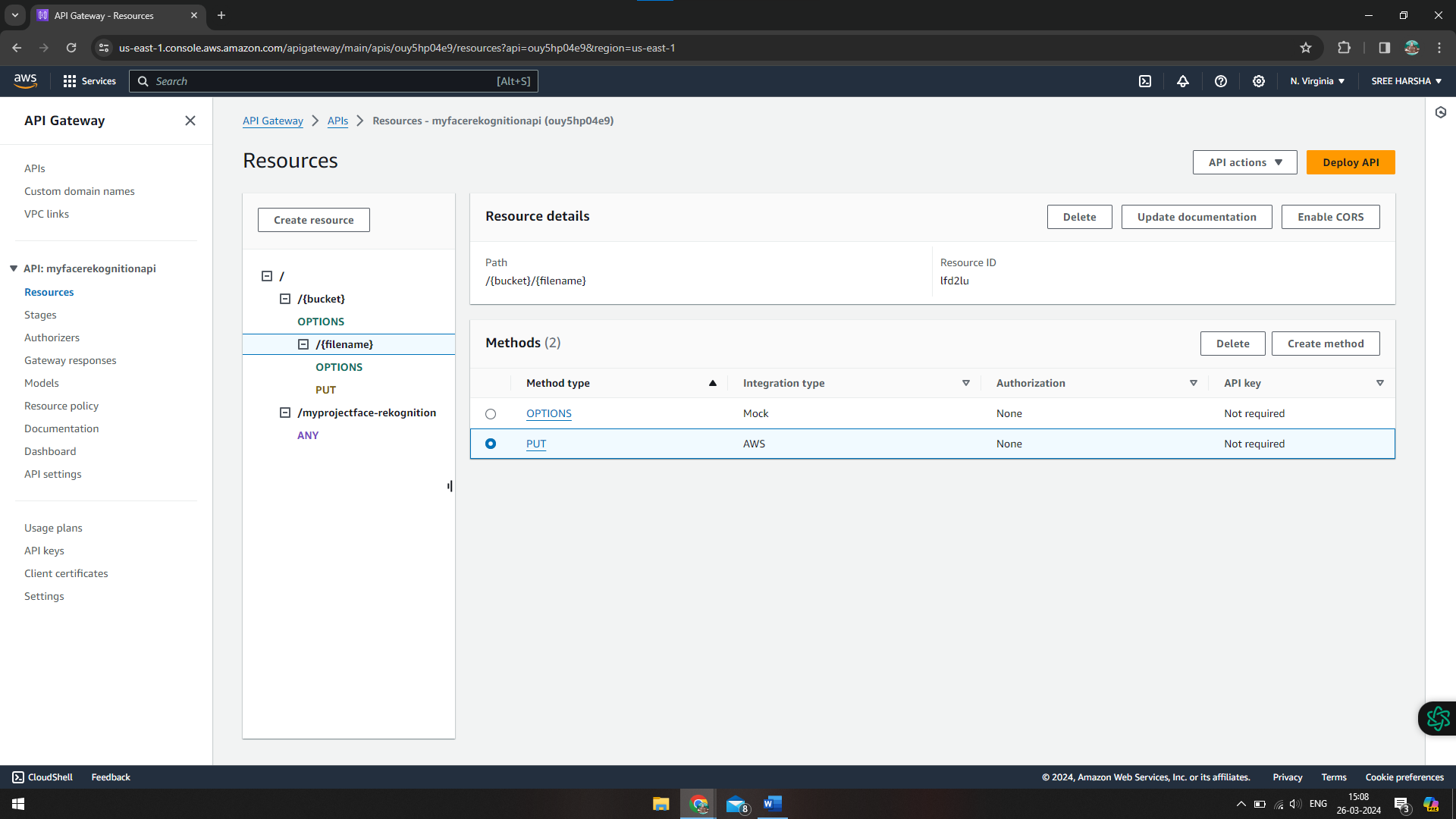
ROLE:



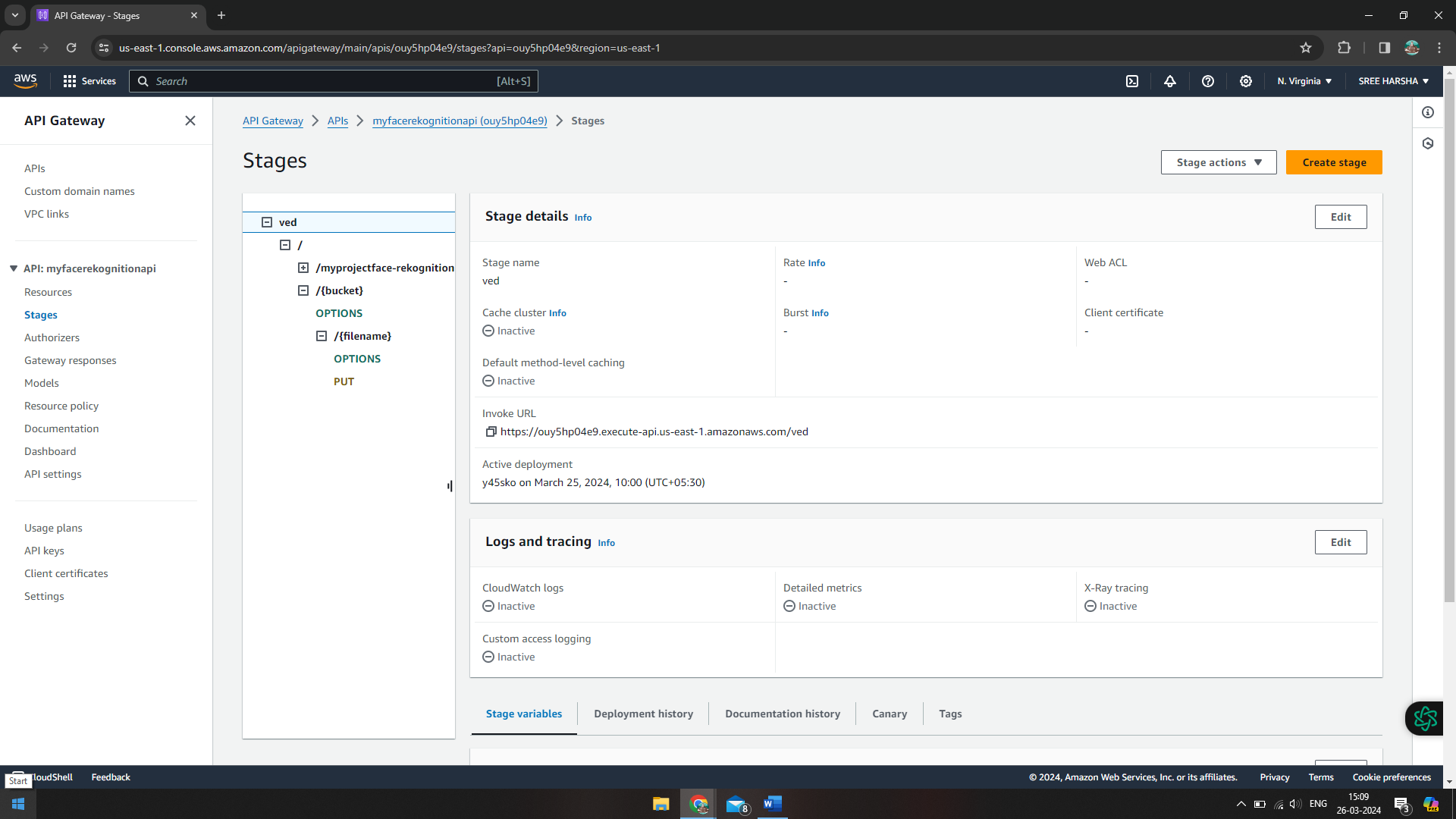
ATTACHED POLICY:



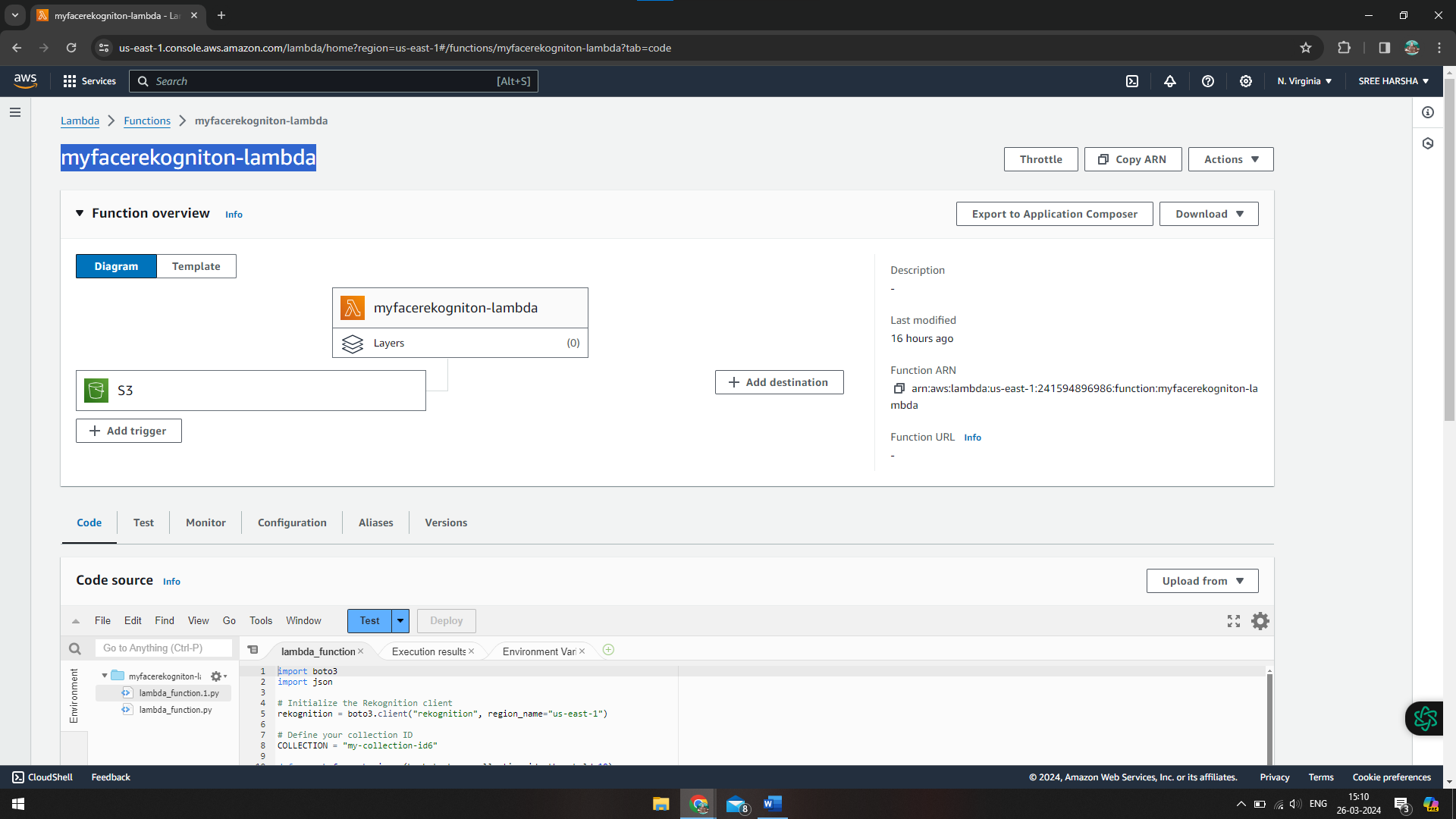
API GATEWAY:



->INVOKE URL:



LAMBDA:



CODE:

# import boto3

# def lambda\_handler(event, context):

# COLLECTION = "my-collection-id6"

# rekognition = boto3.client("rekognition", "us-east-1")

# # Note: you have to create the collection first!

# rekognition.create\_collection(CollectionId=COLLECTION)

import boto3

#index faces (not done)

BUCKET = "myprojectface-rekognition"

KEY = "srinivas.jpeg"

IMAGE\_ID = KEY # S3 key as ImageId

COLLECTION = "my-collection-id6"

# rekognition = boto3.client("rekognition", "us-east-1")

# Note: you have to create the collection first!

# rekognition.create\_collection(CollectionId=COLLECTION)

def index\_faces(bucket, key, collection\_id, image\_id=None, attributes=(), region="us-east-1"):

rekognition = boto3.client("rekognition", region)

response = rekognition.index\_faces(

Image={

"S3Object": {

"Bucket": bucket,

"Name": key,

}

},

CollectionId=collection\_id,

ExternalImageId=image\_id,

DetectionAttributes=attributes,

)

return response['FaceRecords']

def lambda\_handler(event, context):

for record in index\_faces(BUCKET, KEY, COLLECTION, IMAGE\_ID):

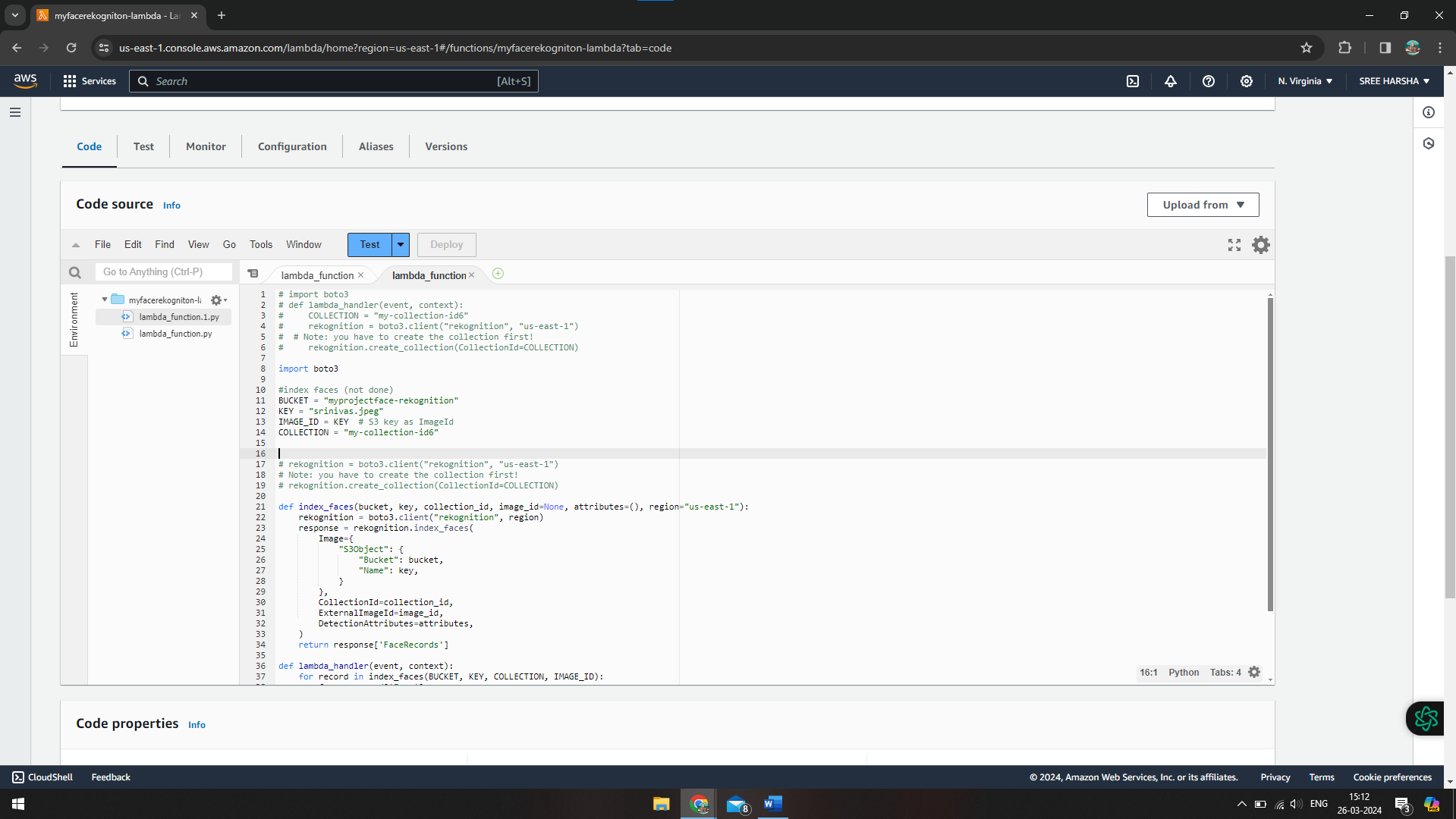
face = record['Face']

# details = record['FaceDetail']

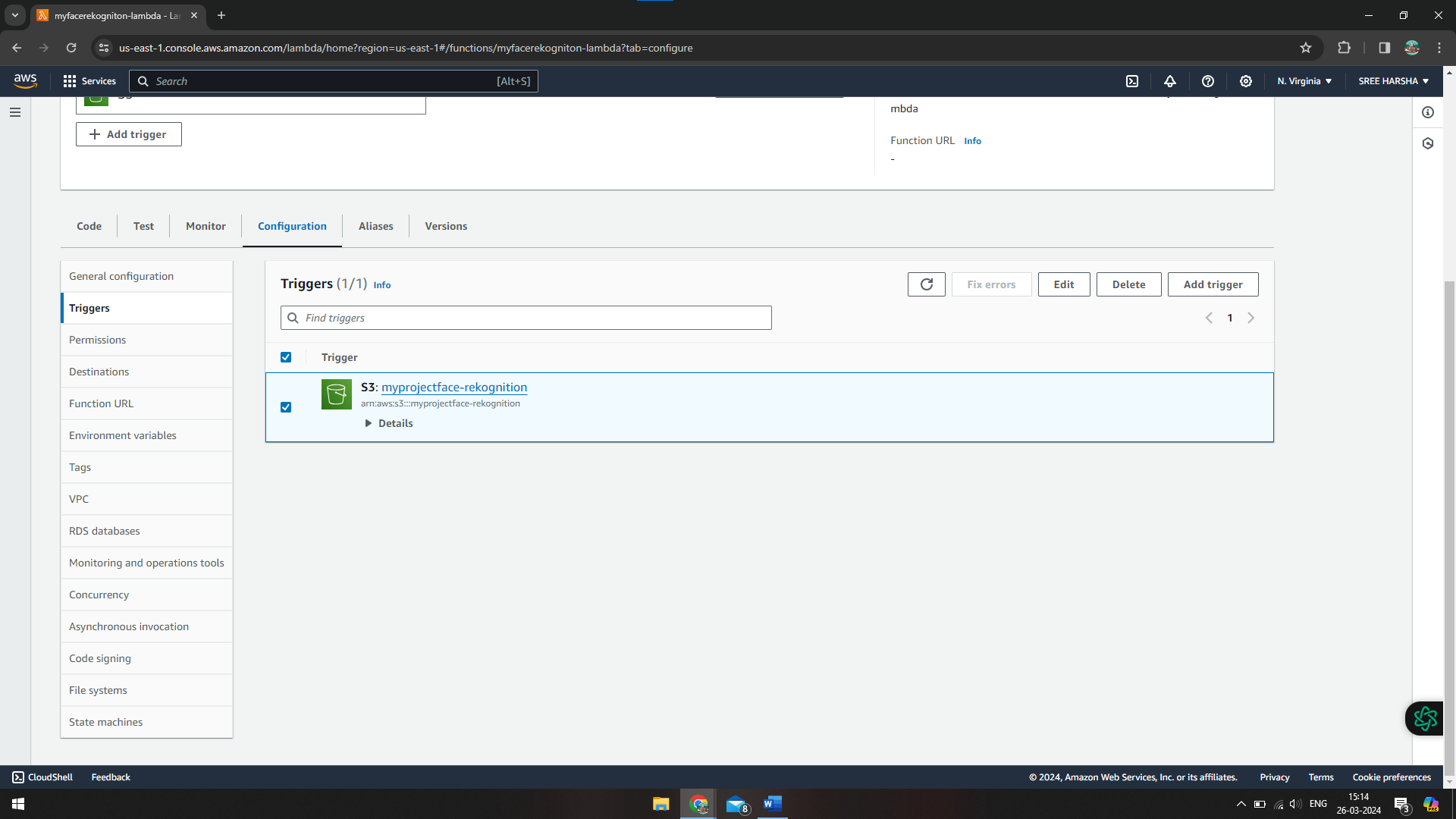
print( "Face ({}%)".format(face['Confidence']))

print( " FaceId: {}".format(face['FaceId']))

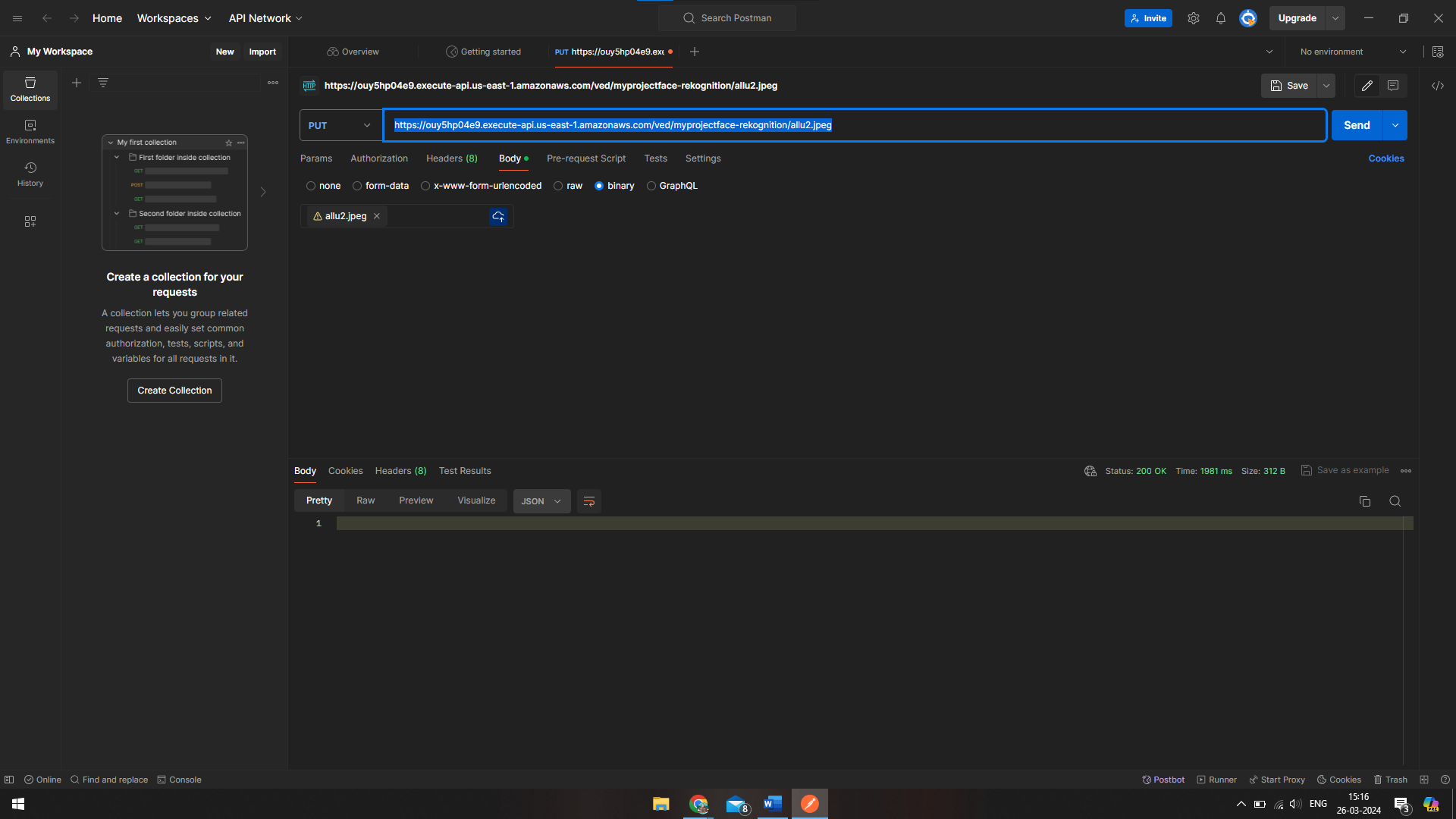
print( " ImageId: {}".format(face['ImageId']))



TRIGGER:



POSTMAN:



CLOUDWATCH:

