

CSCI 5410: Assignment 5

Part B: Use AWS Lambda-SQS-SNS

Code:

GitLab Link: https://git.cs.dal.ca/drshah/dhrumilrakeshshah_csci5410/-/tree/main/Assignment-5

SQSTask.java:

```
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.BasicSessionCredentials;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.sqs.AmazonSQS;
import com.amazonaws.services.sqs.AmazonSQSClientBuilder;
import com.amazonaws.services.sqs.model.SendMessageRequest;

import java.util.ArrayList;
import java.util.List;

/**
 * Author: Dhrumil Rakesh Shah
 * Version: 1.0
 * Class: The AWS Java SDK for SQS Task
 */
public class SQSTask {

    // Driver method
    public static void main(String[] args) {

        // Try block
        try {

            // Storing the AWS Credentials
            final String ACCESS_KEY = "ASIAYN2RGTSOP4FFNENI";
            final String SECRET_KEY = "+vR2zZGR/LyqYdy9d9sWmIIyvTOlix75vSDNfpvgg";
            final String ACCESS_TOKEN = "FwoGZXIvYXZlEMv////////wEaDCER+kWNIHJcnRZ" +
                "EcSK9ATogWleDhA0qndzFJRTxLS1bNkTe5j9Mh/PYNDhg2qdfa+inVS9bB2eUbzeS/a" +
                "BL/uiJk5XdaCAQSVvAprTkALEeZPYeGYvoJnMdmwuAP1XtD/qJ8NHvrjgTIPorQwx3H" +
                "WZKEevk9y5tqEzPdLOWyBkQHkWs4ulJw7/CGMAaQW4HizWmf3Iugpe4sjrrevblhH4X" +
                "s0g4nVzHCy7pvOmGtdyKts4Gdavg9xtP3QXwrwEDVrIT14js3BizxVsUoSjs9YCNBjI" +
                "tm9qlgxAr0g87IOM8enmYZ9J3xtsNG9fJ8jQ0pt6A+eQcVSQrDbcUTQr5Df4u";

            // Storing the SQS Queue Name
            final String QUEUE_NAME = "HalifaxDine";

            // Creating an AWS Credentials object used to connect to the AWS Services
            AWSCredentials awsCredentials = new BasicSessionCredentials(
                ACCESS_KEY, SECRET_KEY, ACCESS_TOKEN);

            // Creating a AWS SQS object using the AWS Credentials
            AmazonSQS amazonSQS = AmazonSQSClientBuilder
                .standard()
                .withCredentials(new AWSStaticCredentialsProvider(awsCredentials))
```

```
.withRegion(Regions.US_EAST_1).build();

// Creating the queueURL that fetches it from the Queue Name
String queueURL = amazonSQS.getQueueUrl(QUEUE_NAME).getQueueUrl();

// Creating two ArrayLists storing the foodOrder and orderQuantity
List<Integer> orderQuantity = new ArrayList<>();
List<String> foodOrder = new ArrayList<>();

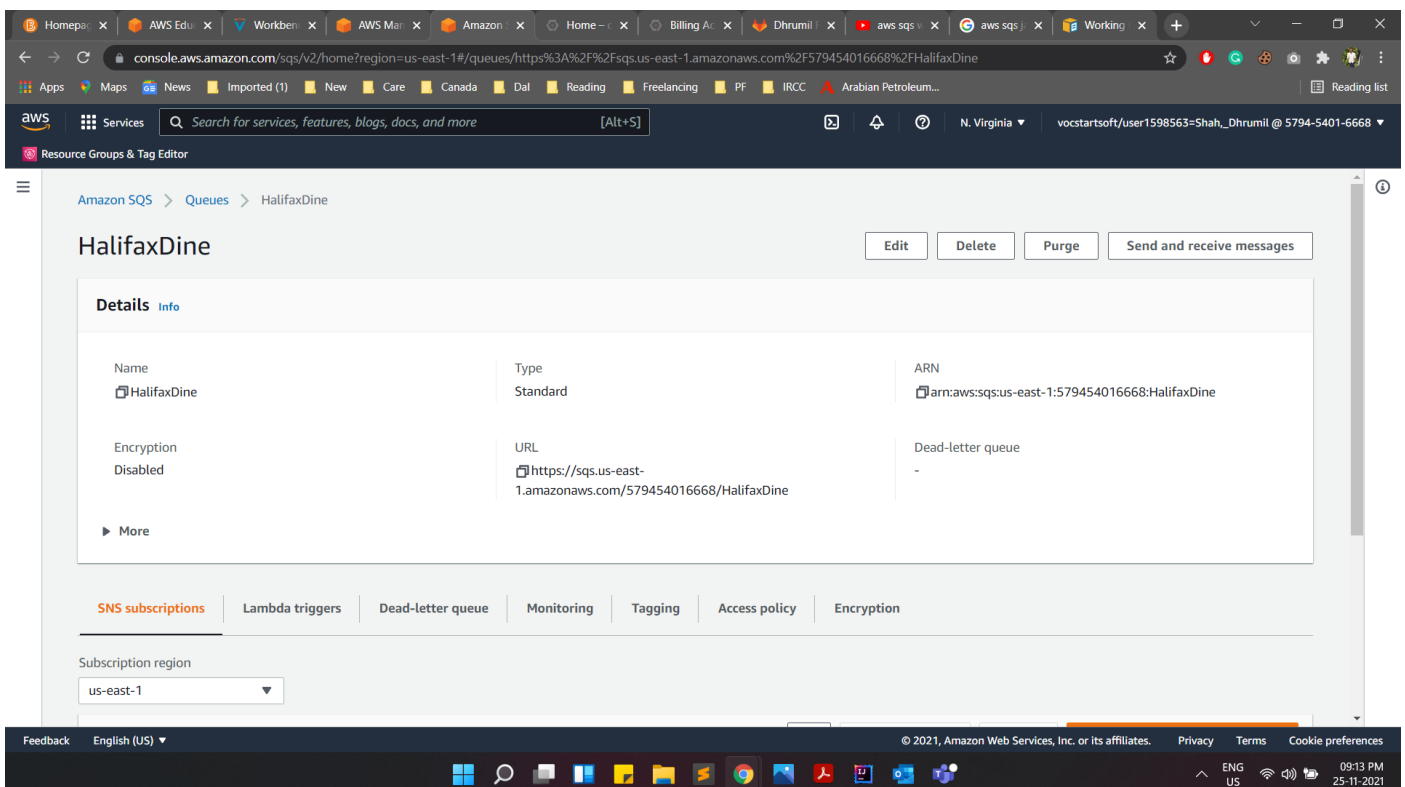
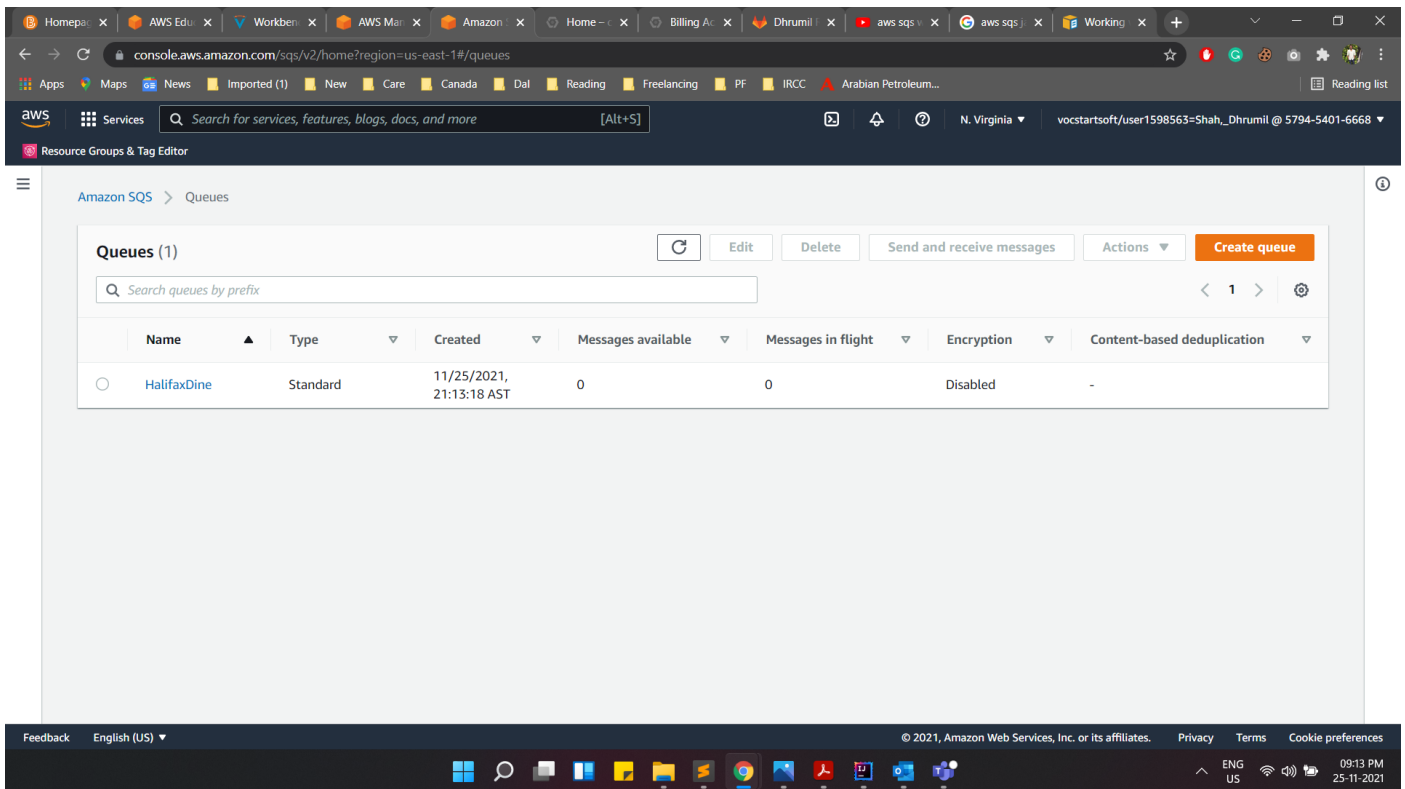
// Adding the food items in the foodOrder ArrayList
foodOrder.add("Pav Bhaji");
foodOrder.add("Vada Pav");
foodOrder.add("Pani Puri");
foodOrder.add("Sandwich");
foodOrder.add("Sizzler");
foodOrder.add("Lassi");

// Adding the order quantity in the orderQuantity ArrayList
orderQuantity.add(1);
orderQuantity.add(2);
orderQuantity.add(3);
orderQuantity.add(4);
orderQuantity.add(5);
orderQuantity.add(6);

// Looping through the ArrayLists and randomly adding orders to the SQS Queue
for (int i = 0; i < orderQuantity.size(); i++) {
    System.out.println(foodOrder.get(i) + ": " + orderQuantity.get(i));

    // Creating a sendMessageRequest object to make the MessageBody
    // and send to the AWS SQS
    SendMessageRequest sendMessageRequest = new SendMessageRequest()
        .withQueueUrl(queueURL)
        .withMessageBody(foodOrder.get(i) + " " + orderQuantity.get(i))
        .withDelaySeconds(5);
    amazonSQS.sendMessage(sendMessageRequest);
    System.out.println(sendMessageRequest);
}
} catch (Exception e) {
    System.err.println(e.getMessage());
}
}
```

ScreenShots:



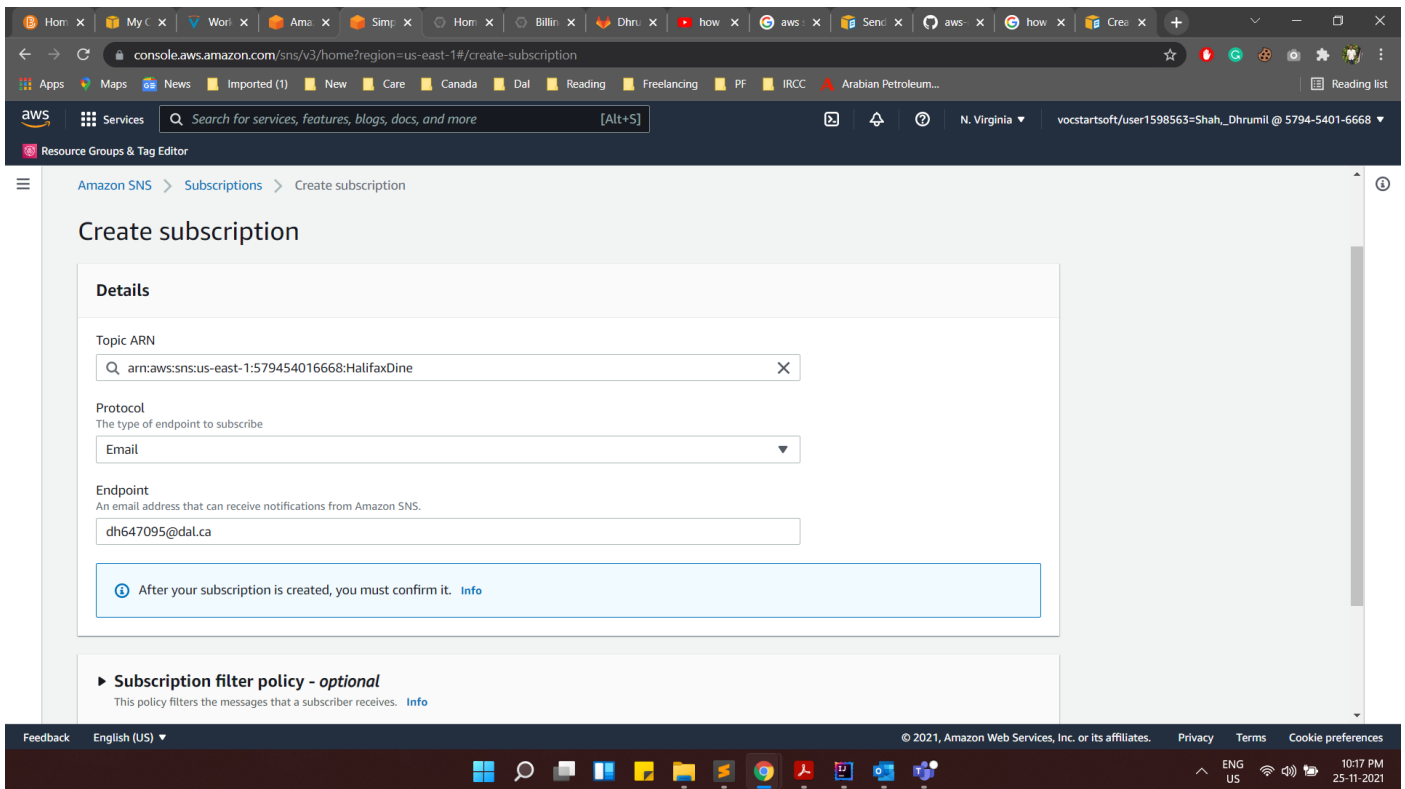


Figure 3: Creating Subscription for HalifaxDine SQS Queue for Email – dh647095@dal.ca

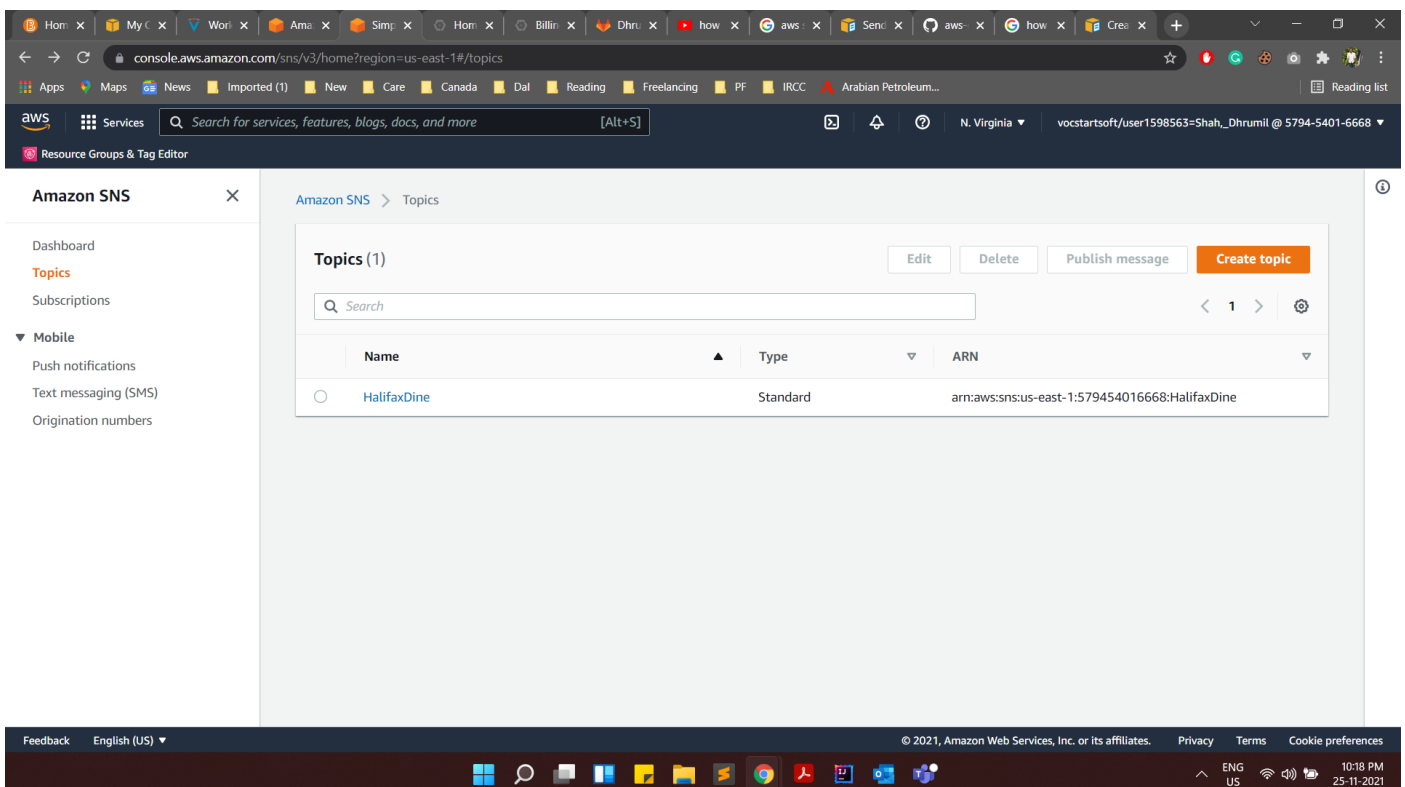


Figure 4: HalifaxDine Topic created in AWS SNS

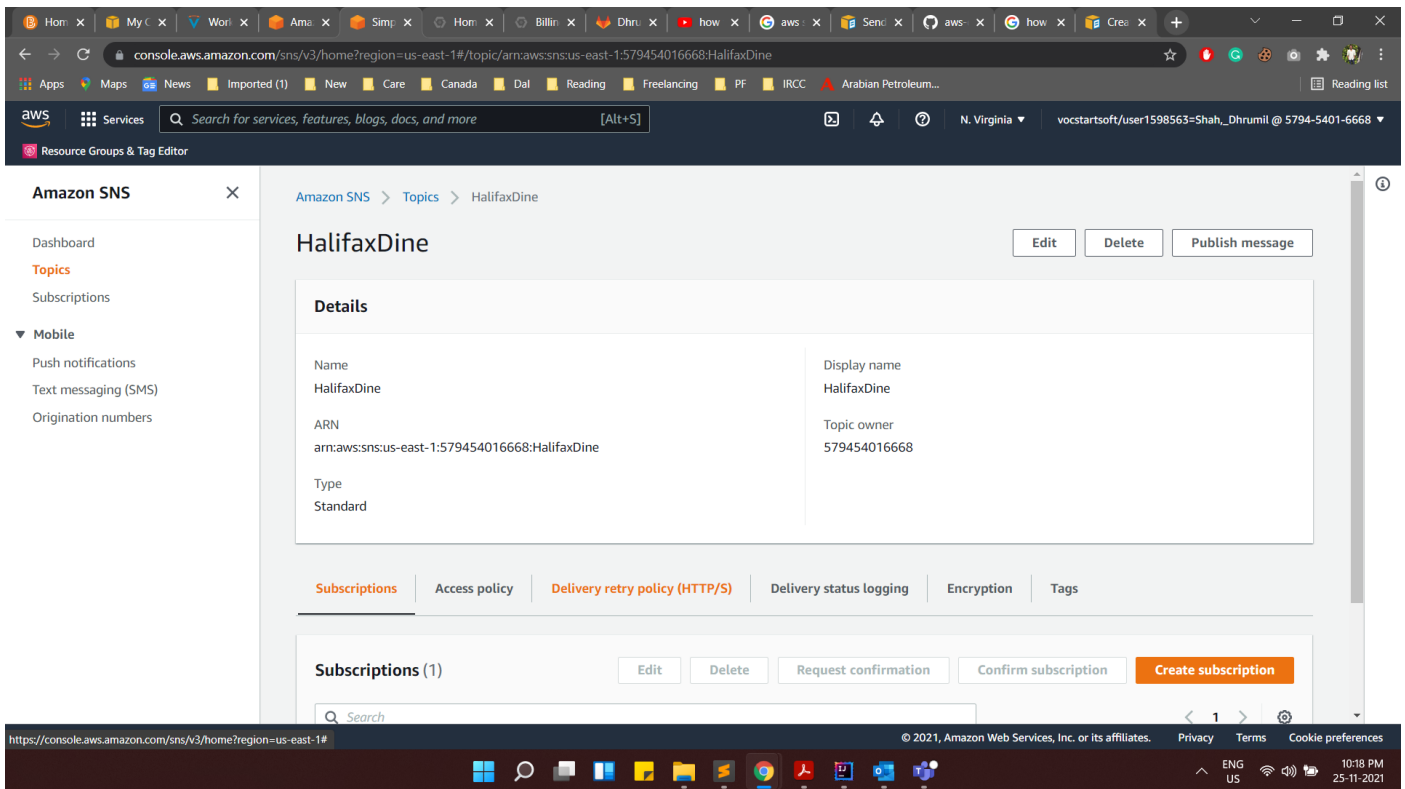


Figure 5: HalifaxDine Subscription created for HalifaxDine Topic

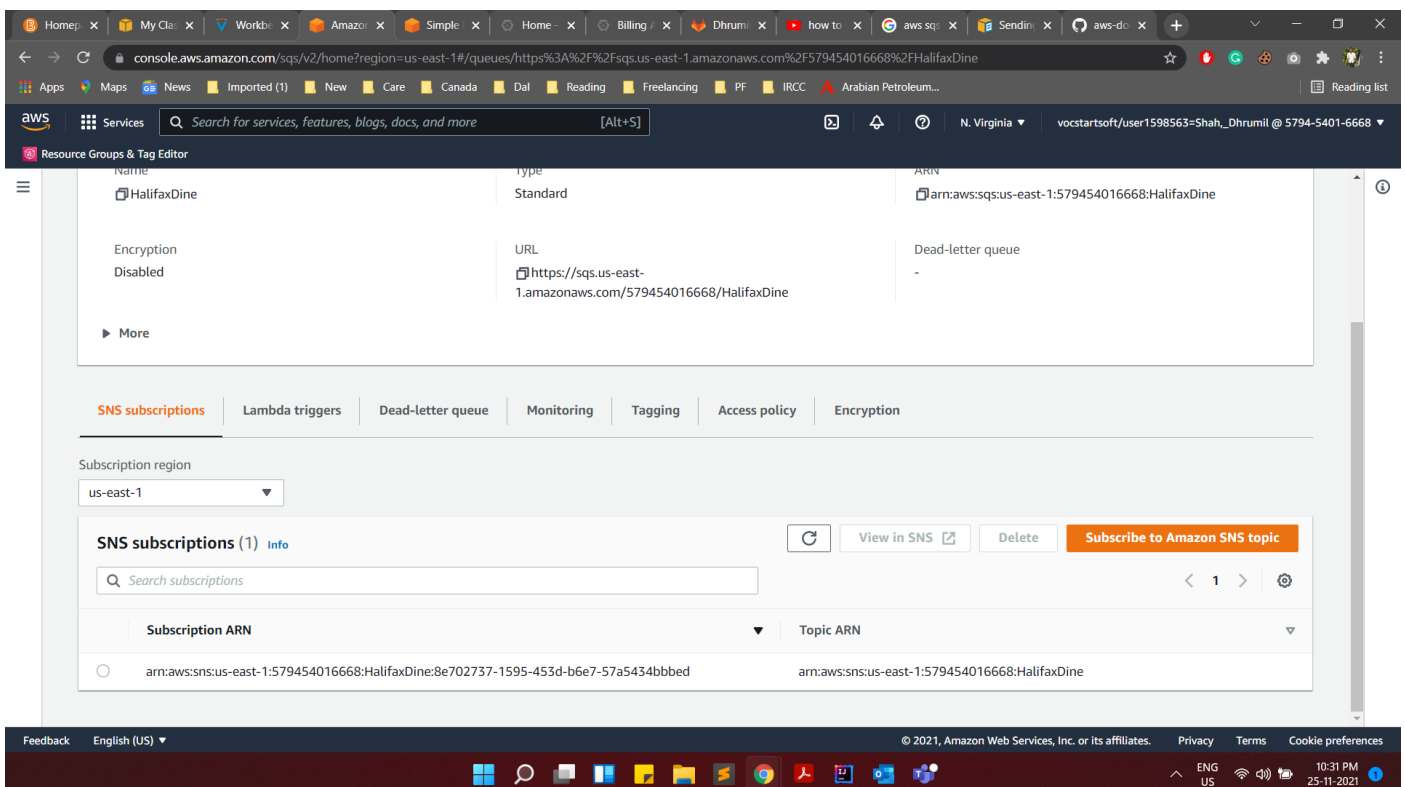


Figure 6: HalifaxDine SNS Subscription added to the HalifaxDine SQS Queue

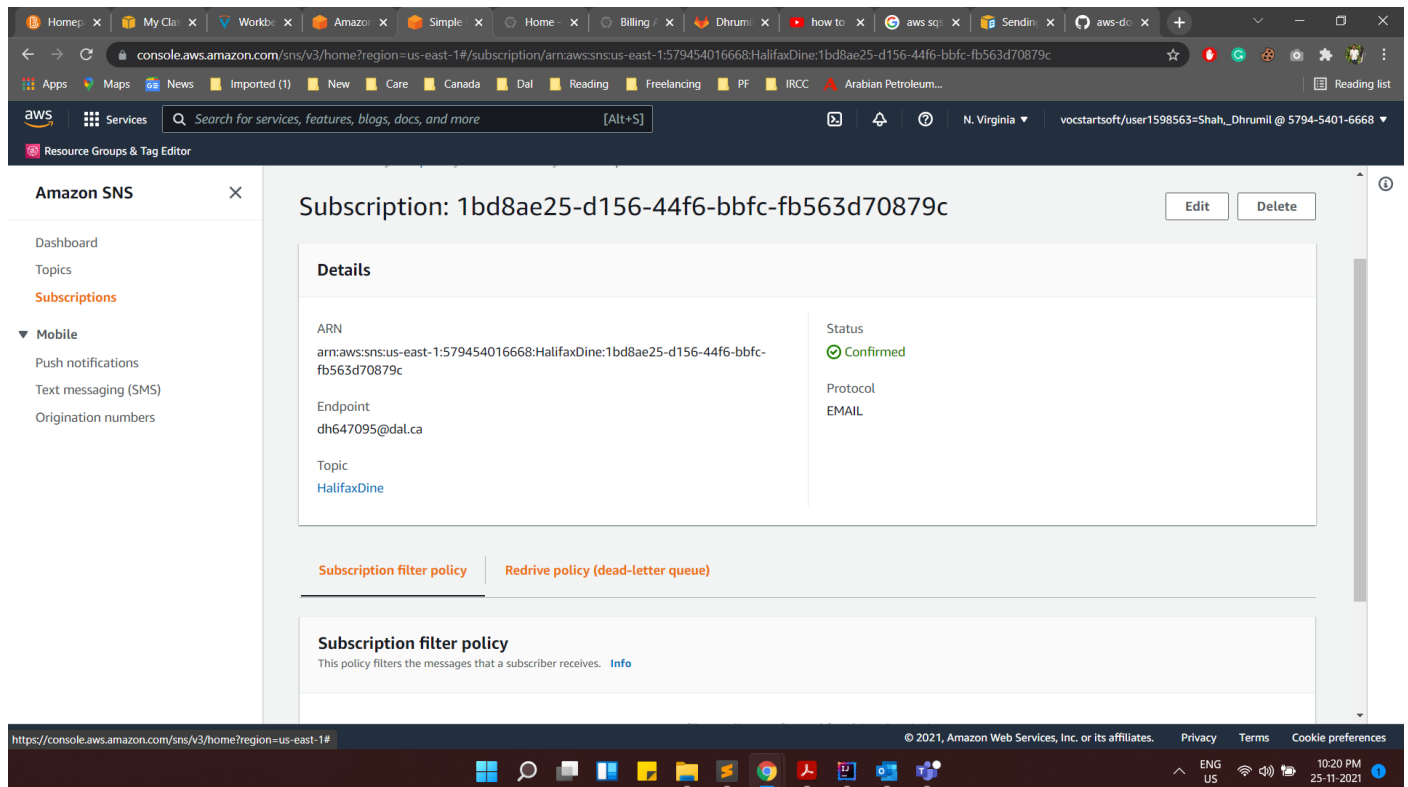


Figure 7: SNS Subscription addition confirmation

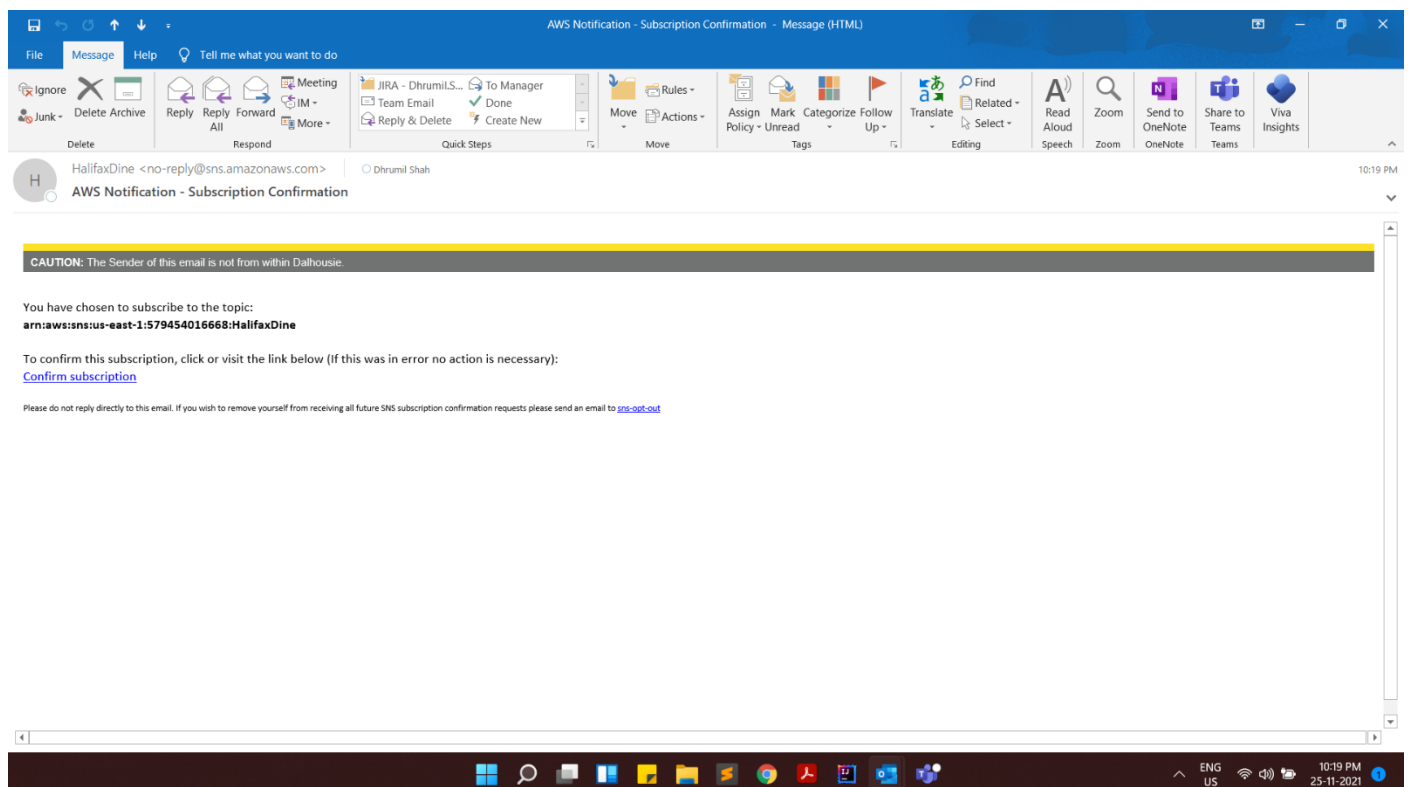


Figure 8: SNS Subscription Email Confirmation

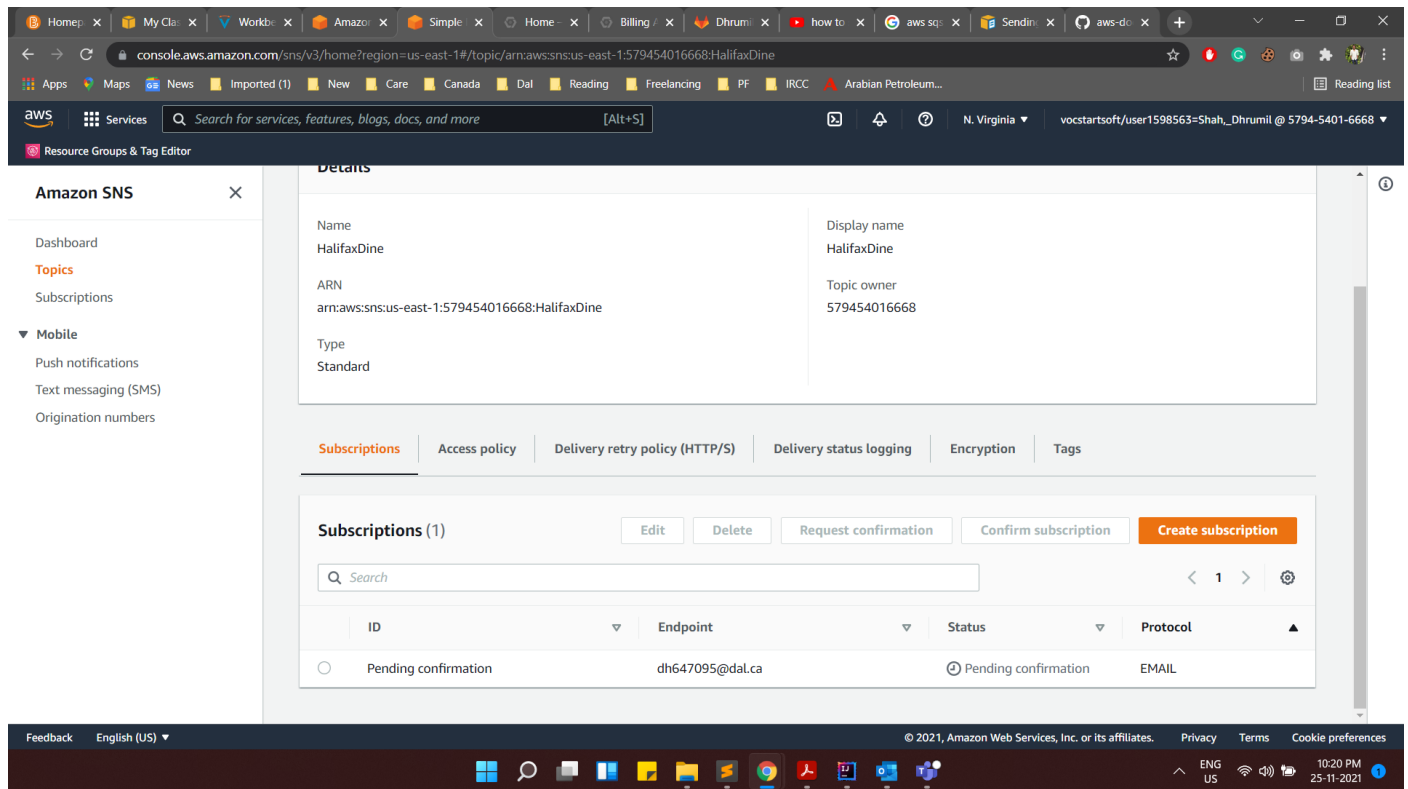


Figure 9: SNS Subscription Email Confirmation Pending Confirmation

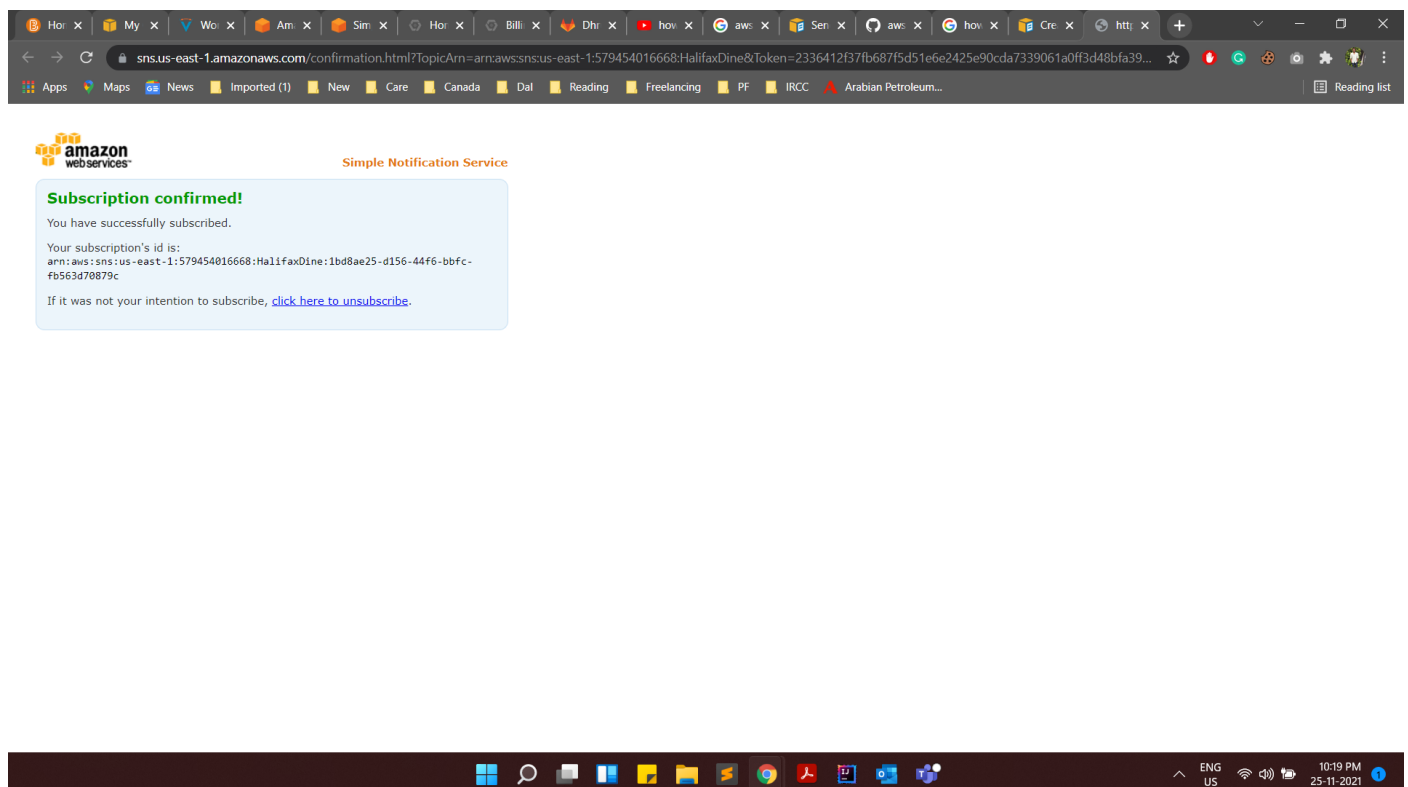


Figure 10: SNS Subscription Confirmed

The screenshot shows the AWS Management Console for an Amazon SNS topic named 'HalifaxDine'. The left sidebar contains navigation links for Dashboard, Topics, Subscriptions, and Mobile. The main content area displays the topic details, including Name, Display name, ARN, Topic owner, and Type. Below the details, the 'Subscriptions' tab is active, showing a table with one confirmed email subscription.

ID	Endpoint	Status	Protocol
1bd8ae25-d156-44f6-bbfc-fb563d70879c	dh647095@dal.ca	Confirmed	EMAIL

Figure 11: SNS Email Subscription Confirmed

The screenshot shows the AWS Management Console for the same Amazon SNS topic 'HalifaxDine'. The 'Subscriptions' tab is active, showing a table with two confirmed subscriptions: one email and one SQS.

ID	Endpoint	Status	Protocol
1bd8ae25-d156-44f6-bbfc-fb563d70879c	dh647095@dal.ca	Confirmed	EMAIL
8e702737-1595-453d-b6e7-57a5434bbbed	arn:aws:sqs:us-east-1:579454016668:HalifaxDine	Confirmed	SQS

Figure 12: SNS Subscription Confirmed for both SQS and Email

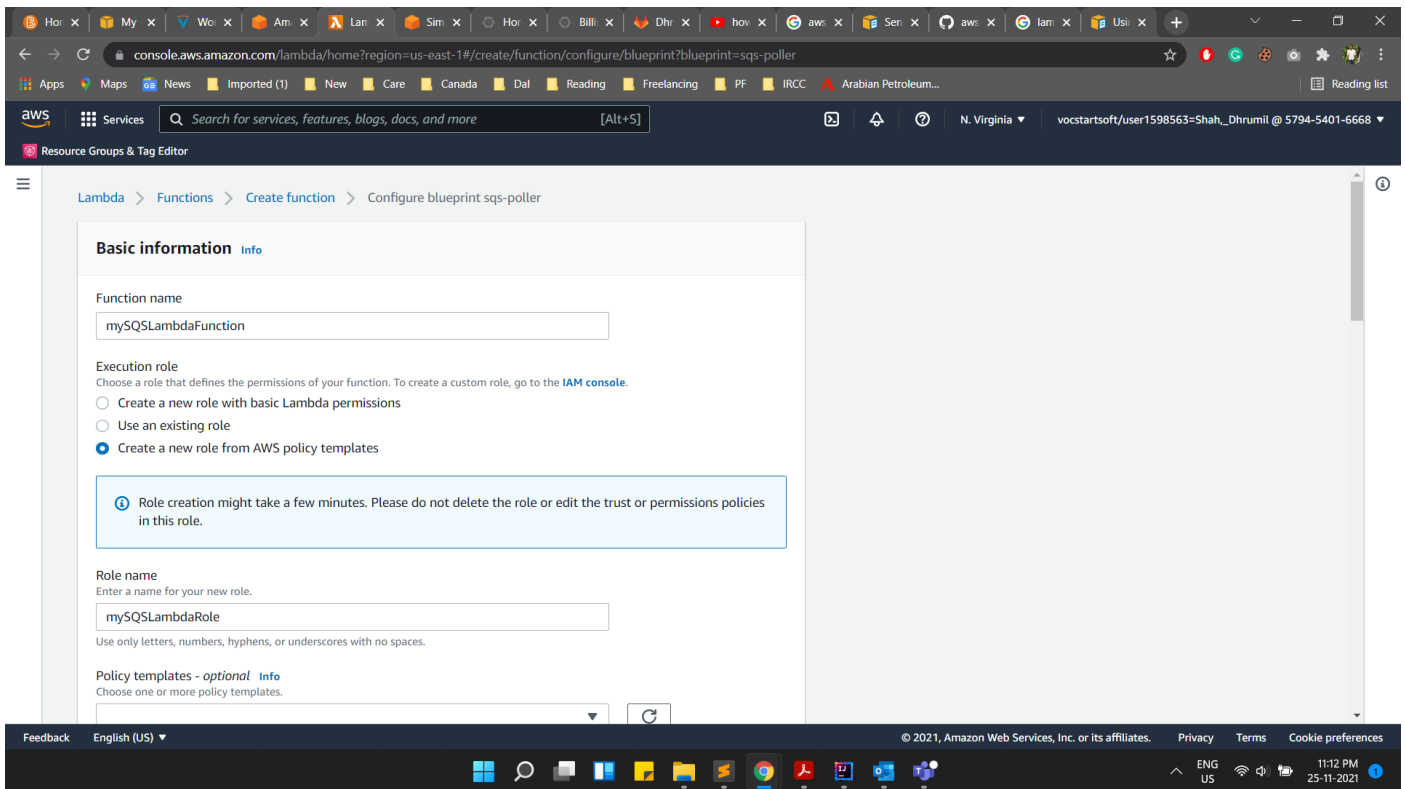


Figure 13: Lambda Function for SQS Trigger

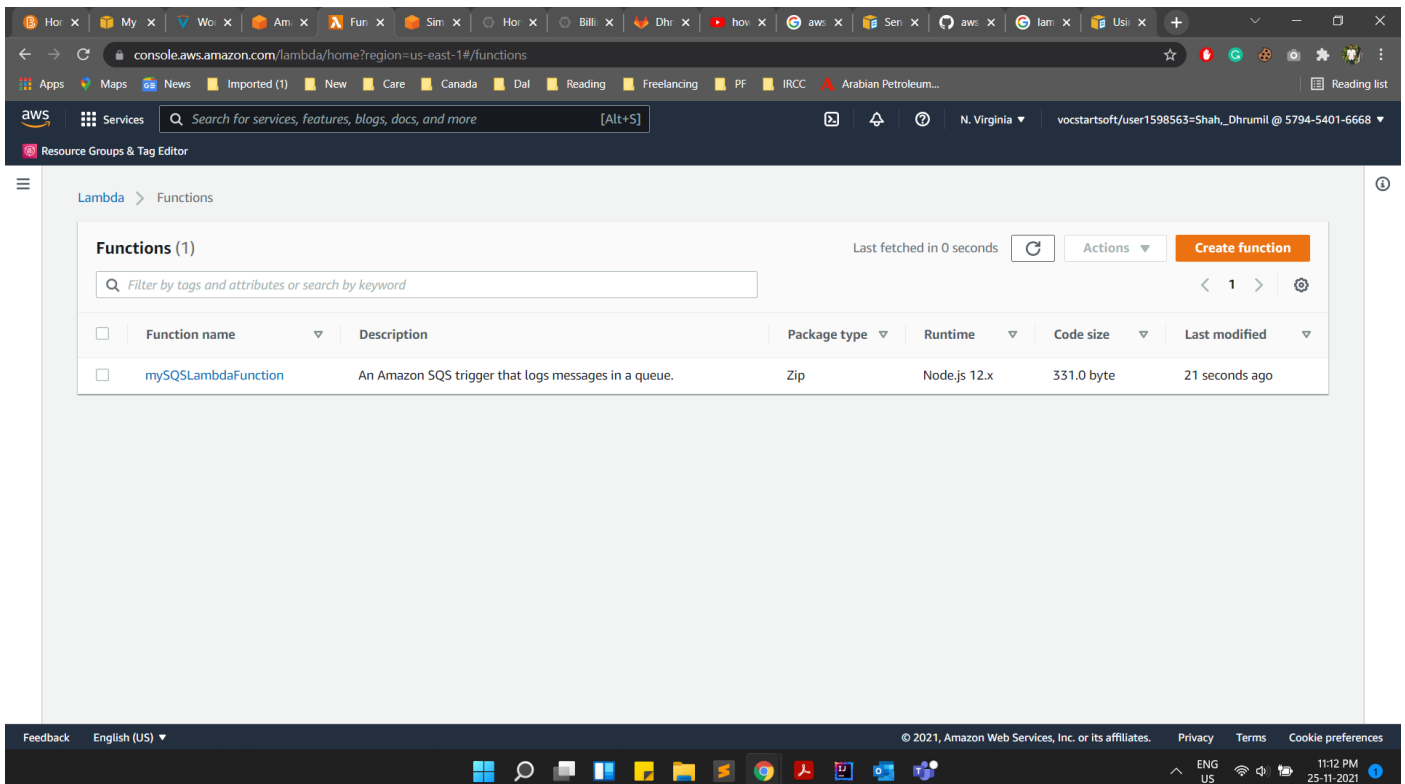


Figure 14: mySQLLambdaFunction Created

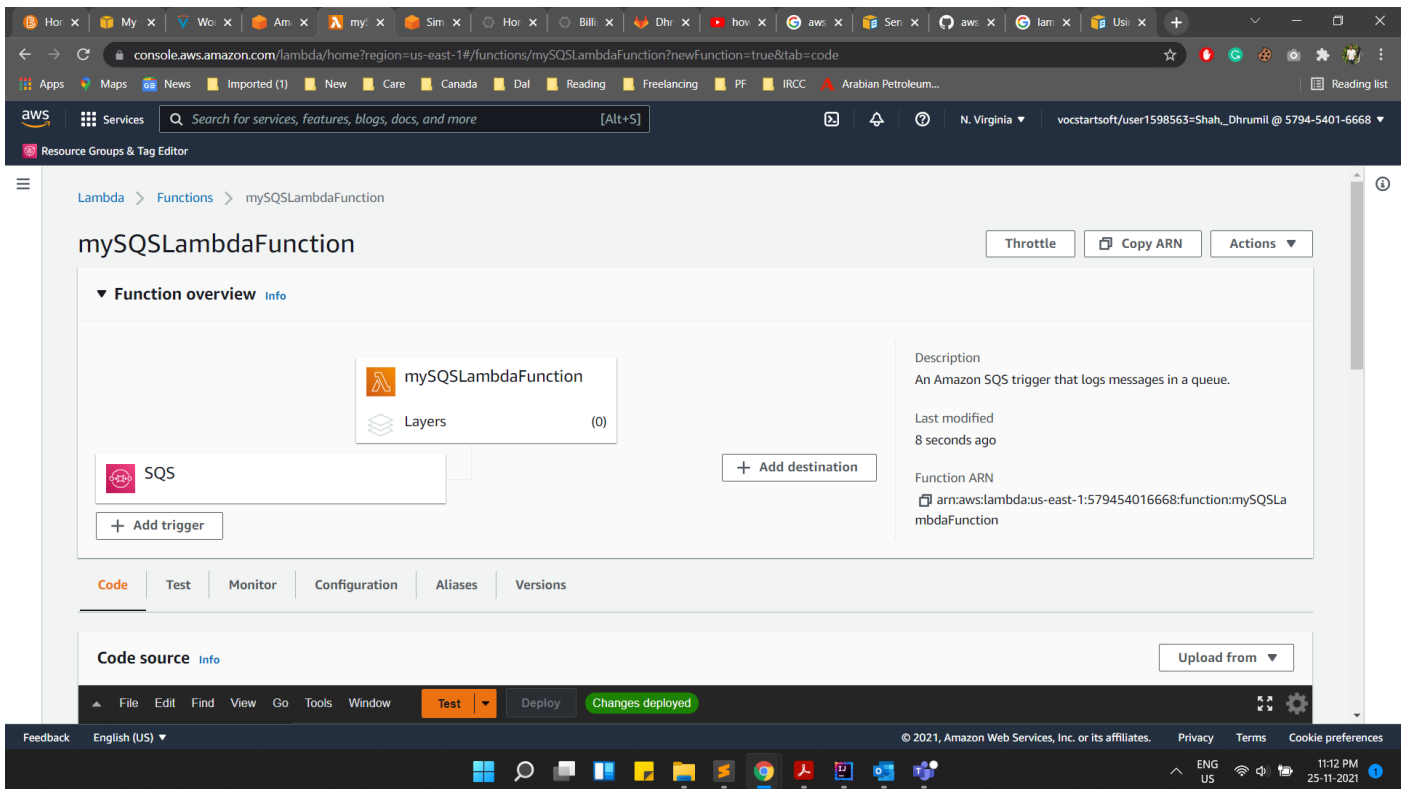


Figure 15: mySQLLambdaFunction with SQS Trigger

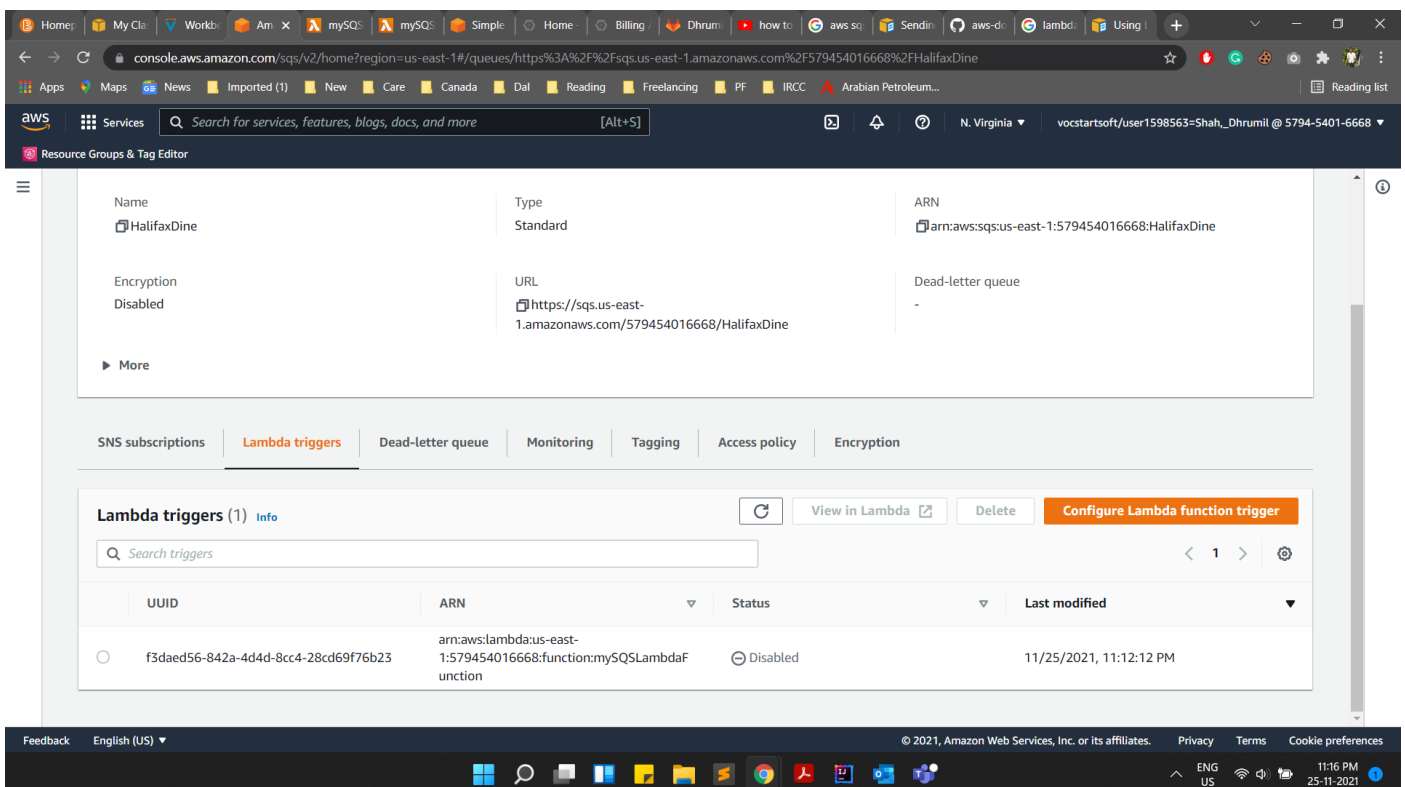


Figure 16: Lambda Trigger by default Disabled in SQS Queue

The screenshot displays the AWS Management Console interface for an Amazon SQS queue named 'HalifaxDine'. The console shows the queue's details, including its name, type (Standard), ARN, encryption status (Disabled), and URL. Below the details, the 'Lambda triggers' tab is selected, showing a single trigger that is explicitly enabled. The trigger's status is 'Enabled', and it was last modified on 11/26/2021 at 12:33:07 AM. The console also shows the 'Configure Lambda function trigger' button and the 'View in Lambda' link.

Queue Name: HalifaxDine
Type: Standard
ARN: arn:aws:sqs:us-east-1:579454016668:HalifaxDine
Encryption: Disabled
URL: https://sqs.us-east-1.amazonaws.com/579454016668/HalifaxDine
Dead-letter queue: -

Lambda triggers (1) info

UUID	ARN	Status	Last modified
f3daed56-842a-4d4d-8cc4-28cd69f76b23	arn:aws:lambda:us-east-1:579454016668:function:mySQLLambdaFunction	Enabled	11/26/2021, 12:33:07 AM

Figure 17: Lambda Function Trigger explicitly Enabled in SQS Queue