Assignment 5

Part B

Meet Patel (B00899516)

Dalhousie University

Subject

CSCI 5410 (Serverless Data Processing)

Professor

Dr. Saurabh Dey

Project Git Repository

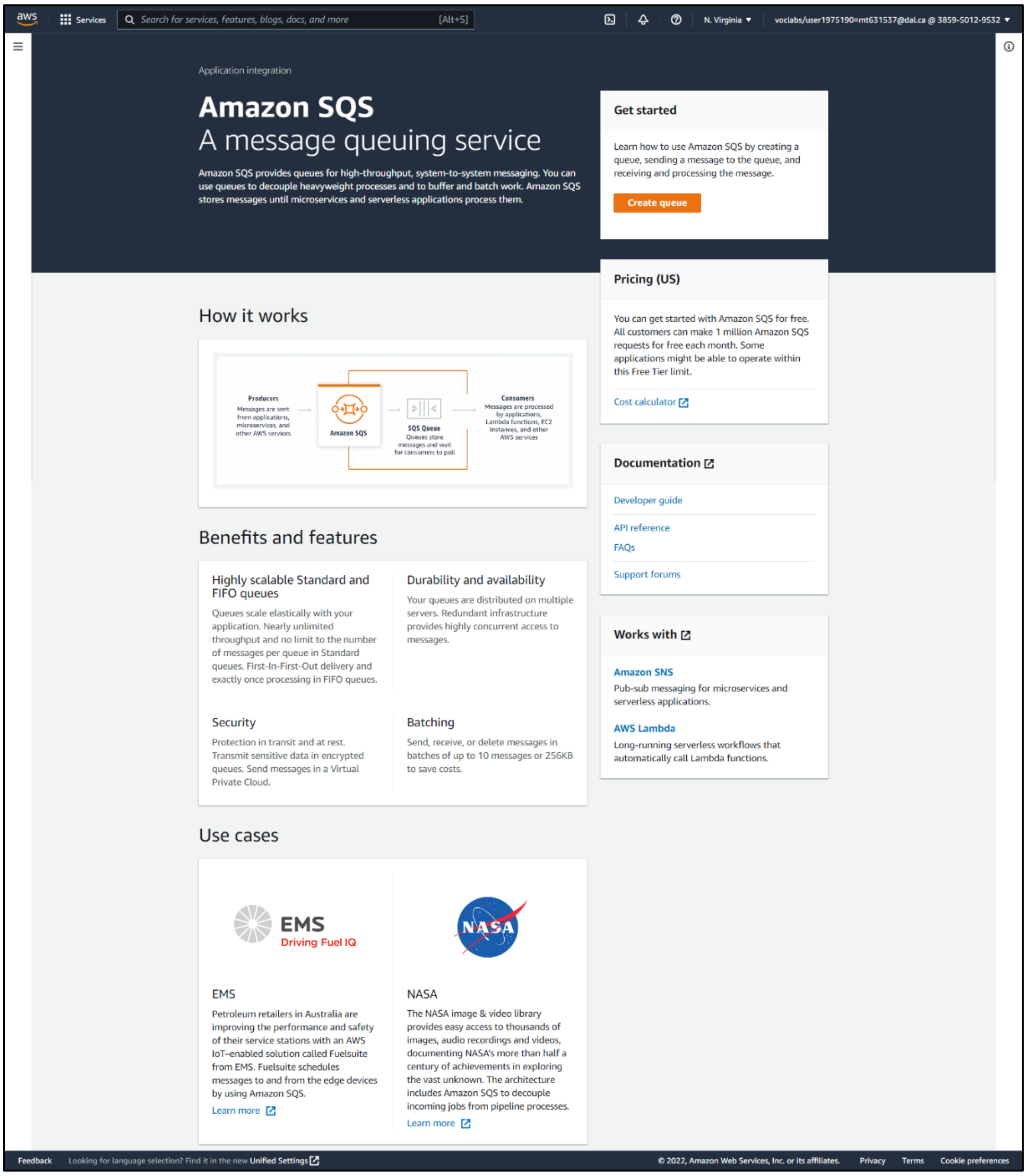
Gitlab Repository Link: https://git.cs.dal.ca/patel13/csci5410_b00899516_meet_patel.git

Use AWS Lambda-SQS-SNS.

Screenshots of all the steps performed

Figures 2 to **22** is responsible for showing the initial steps. I used AWS lambda, AWS SNS, and AWS SQS to complete this assignment. These all steps are performed on AWS console.

Figure 2 is responsible for showing **Amazon SQS** start page where we need to click on "**Create Queue**" button.



aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532

Application integration

Amazon SQS
A message queuing service

Amazon SQS provides queues for high-throughput, system-to-system messaging. You can use queues to decouple heavyweight processes and to buffer and batch work. Amazon SQS stores messages until microservices and serverless applications process them.

Get started

Learn how to use Amazon SQS by creating a queue, sending a message to the queue, and receiving and processing the message.

Create queue

Pricing (US)

You can get started with Amazon SQS for free. All customers can make 1 million Amazon SQS requests for free each month. Some applications might be able to operate within this Free Tier limit.

Cost calculator

How it works

Producers
Messages are sent from applications, microservices, and other AWS services

Amazon SQS

SQS Queue
Queues store messages and wait for consumers to poll

Consumers
Messages are processed by applications, Lambda functions, EC2 Instances, and other AWS services

Documentation

Developer guide

API reference

FAQs

Support forums

Benefits and features

Highly scalable Standard and FIFO queues

Queues scale elastically with your application. Nearly unlimited throughput and no limit to the number of messages per queue in Standard queues. First-In-First-Out delivery and exactly once processing in FIFO queues.

Durability and availability

Your queues are distributed on multiple servers. Redundant infrastructure provides highly concurrent access to messages.

Security

Protection in transit and at rest. Transmit sensitive data in encrypted queues. Send messages in a Virtual Private Cloud.

Batching

Send, receive, or delete messages in batches of up to 10 messages or 256KB to save costs.

Works with

Amazon SNS
Pub-sub messaging for microservices and serverless applications.

AWS Lambda
Long-running serverless workflows that automatically call Lambda functions.

Use cases

EMS
Driving Fuel IQ

EMS

Petroleum retailers in Australia are improving the performance and safety of their service stations with an AWS IoT–enabled solution called Fuelsuite from EMS. Fuelsuite schedules messages to and from the edge devices by using Amazon SQS.

Learn more

NASA

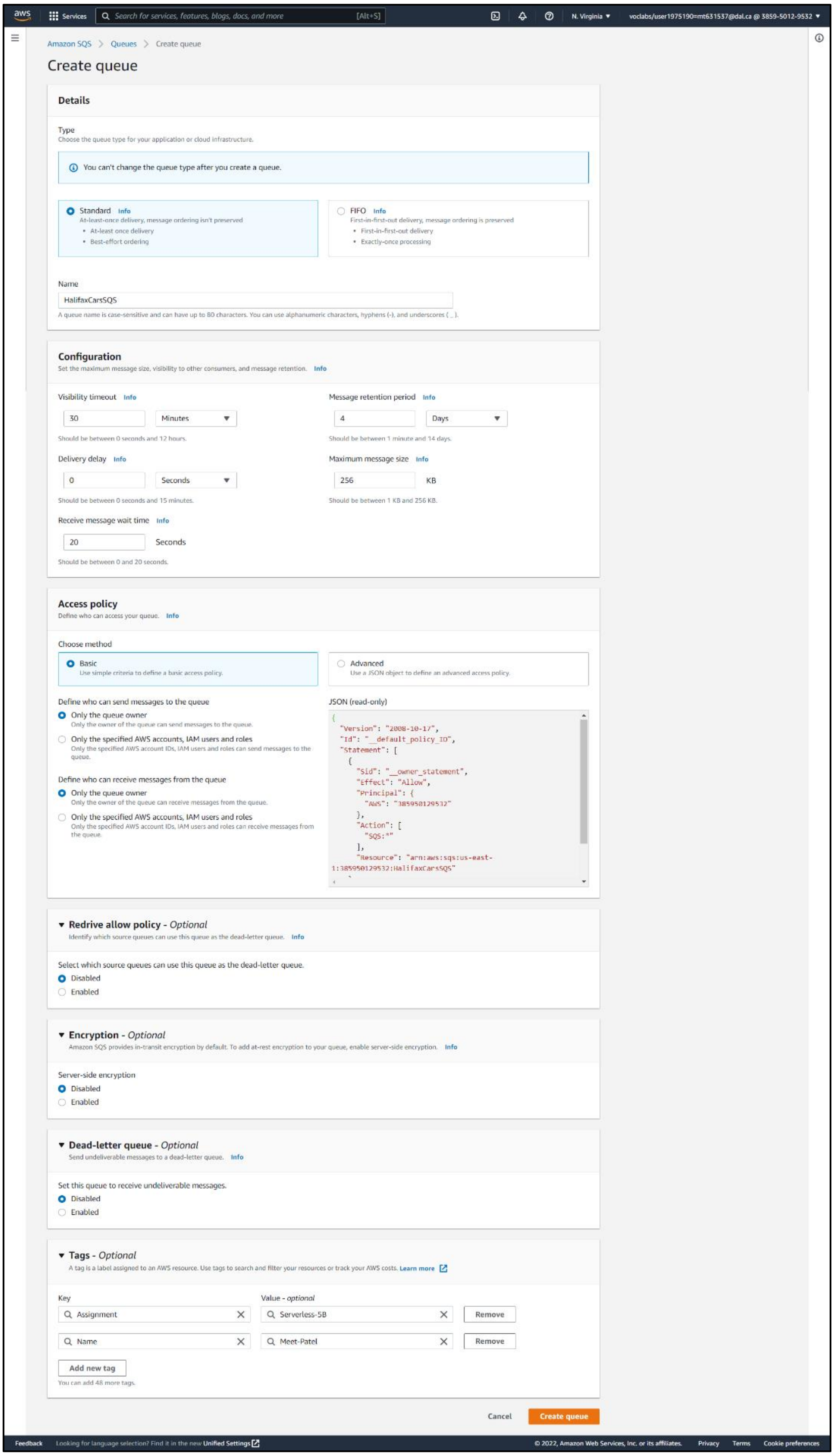
The NASA image & video library provides easy access to thousands of images, audio recordings and videos, documenting NASA's more than half a century of achievements in exploring the vast unknown. The architecture includes Amazon SQS to decouple incoming jobs from pipeline processes.

Learn more

Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Figure 1: Amazon SQS start page

Figure 3 is responsible for showing the configuration page of “Create Queue” page. I filled the information that is showed in screenshot.


aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia ▼
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532 ▼
Amazon SQS > Queues > Create queue

Create queue

Details

Type
Choose the queue type for your application or cloud infrastructure.

You can't change the queue type after you create a queue.

● Standard Info
At-least-once delivery, message ordering isn't preserved
- At-least once delivery
- Best-effort ordering

○ FIFO Info
First-in-first-out delivery, message ordering is preserved
- First-in-first-out delivery
- Exactly-once processing

Name: HalifaxCarsSQS
A queue name is case-sensitive and can have up to 80 characters. You can use alphanumeric characters, hyphens (-), and underscores (_).

Configuration
Set the maximum message size, visibility to other consumers, and message retention. Info

Visibility timeout Info: 30 Minutes
Should be between 0 seconds and 12 hours.

Message retention period Info: 4 Days
Should be between 1 minute and 14 days.

Delivery delay Info: 0 Seconds
Should be between 0 seconds and 15 minutes.

Maximum message size Info: 256 KB
Should be between 1 KB and 256 KB.

Receive message wait time Info: 20 Seconds
Should be between 0 and 20 seconds.

Access policy
Define who can access your queue. Info

Choose method
● Basic
Use simple criteria to define a basic access policy.
○ Advanced
Use a JSON object to define an advanced access policy.

Define who can send messages to the queue
● Only the queue owner
Only the owner of the queue can send messages to the queue.
○ Only the specified AWS accounts, IAM users and roles
Only the specified AWS account IDs, IAM users and roles can send messages to the queue.

Define who can receive messages from the queue
● Only the queue owner
Only the owner of the queue can receive messages from the queue.
○ Only the specified AWS accounts, IAM users and roles
Only the specified AWS account IDs, IAM users and roles can receive messages from the queue.

JSON (read-only)

```
{
  "Version": "2008-10-17",
  "Id": "__default_policy_ID",
  "Statement": [
    {
      "Sid": "__owner_statement",
      "Effect": "Allow",
      "Principal": {
        "AWS": "385950129532"
      },
      "Action": [
        "SQS:*"
      ],
      "Resource": "arn:aws:sqs:us-east-
1:385950129532:HalifaxCarsSQS"
```

Redrive allow policy - *Optional*
Identify which source queues can use this queue as the dead-letter queue. Info

Select which source queues can use this queue as the dead-letter queue.
● Disabled
○ Enabled

Encryption - *Optional*
Amazon SQS provides in-transit encryption by default. To add at-rest encryption to your queue, enable server-side encryption. Info

Server-side encryption
● Disabled
○ Enabled

Dead-letter queue - *Optional*
Send undeliverable messages to a dead-letter queue. Info

Set this queue to receive undeliverable messages.
● Disabled
○ Enabled

Tags - *Optional*
A tag is a label assigned to an AWS resource. Use tags to search and filter your resources or track your AWS costs. Learn more

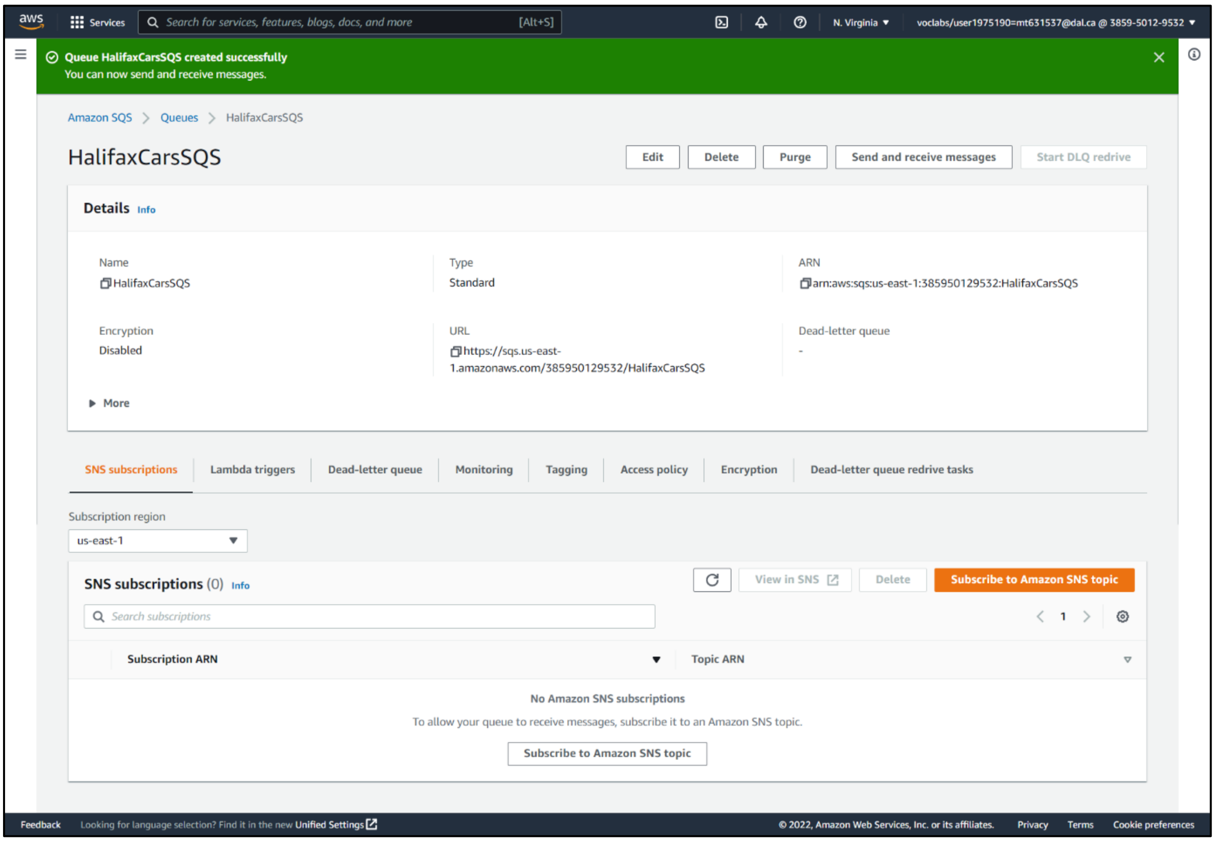
Key	Value - *optional*
Assignment	Serverless-5B
Name	Meet-Patel

You can add 48 more tags.

Remove
Remove
Add new tag
Cancel
Create queue
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Figure 2: Configuration page of “Create Queue”

Figure 4 and 5 is responsible for showing the successful creation of **HalifacCarsSQS** queue. The created order will be added to this queue.


aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532
Amazon SQS > Queues > HalifaxCarsSQS
Edit
Delete
Purge
Send and receive messages
Start DLQ redrive
More
SNS subscriptions
Lambda triggers
Dead-letter queue
Monitoring
Tagging
Access policy
Encryption
Dead-letter queue redrive tasks
View in SNS
Delete
Subscribe to Amazon SNS topic
Search subscriptions
1
Subscribe to Amazon SNS topic
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Queue HalifaxCarsSQS created successfully
You can now send and receive messages.

HalifaxCarsSQS

Details Info

Name: HalifaxCarsSQS

Type: Standard

ARN: arn:aws:sqs:us-east-1:385950129532:HalifaxCarsSQS

Encryption: Disabled

URL: https://sqs.us-east-1.amazonaws.com/385950129532/HalifaxCarsSQS

Dead-letter queue: -

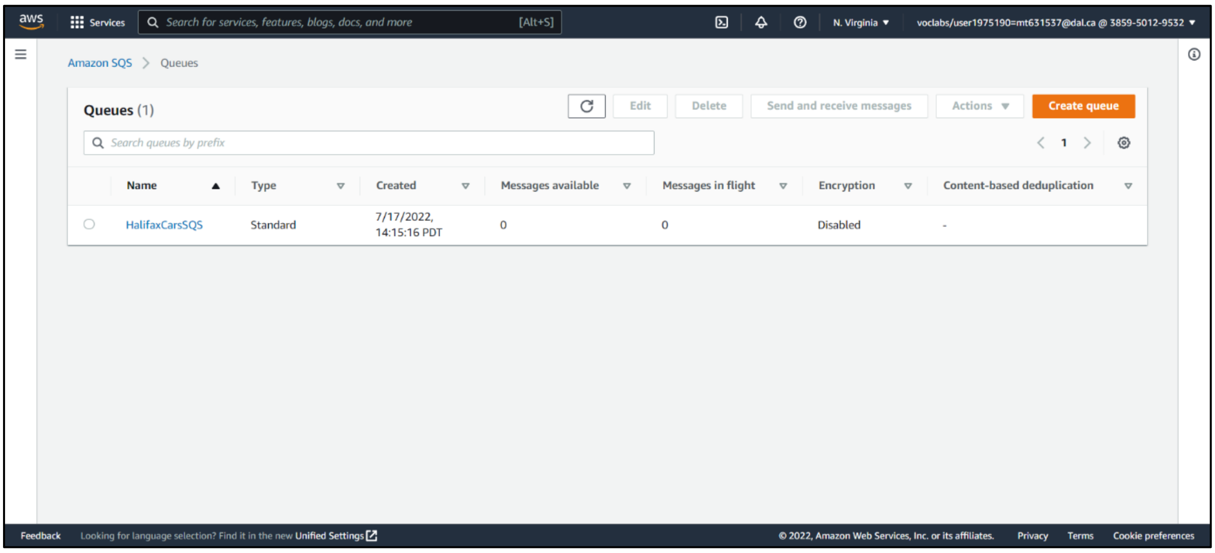
Subscription region: us-east-1

SNS subscriptions (0) Info

Subscription ARN	Topic ARN

No Amazon SNS subscriptions
To allow your queue to receive messages, subscribe it to an Amazon SNS topic.

*Figure 3: Successful creation of **HalifaxCarsSQS** queue*

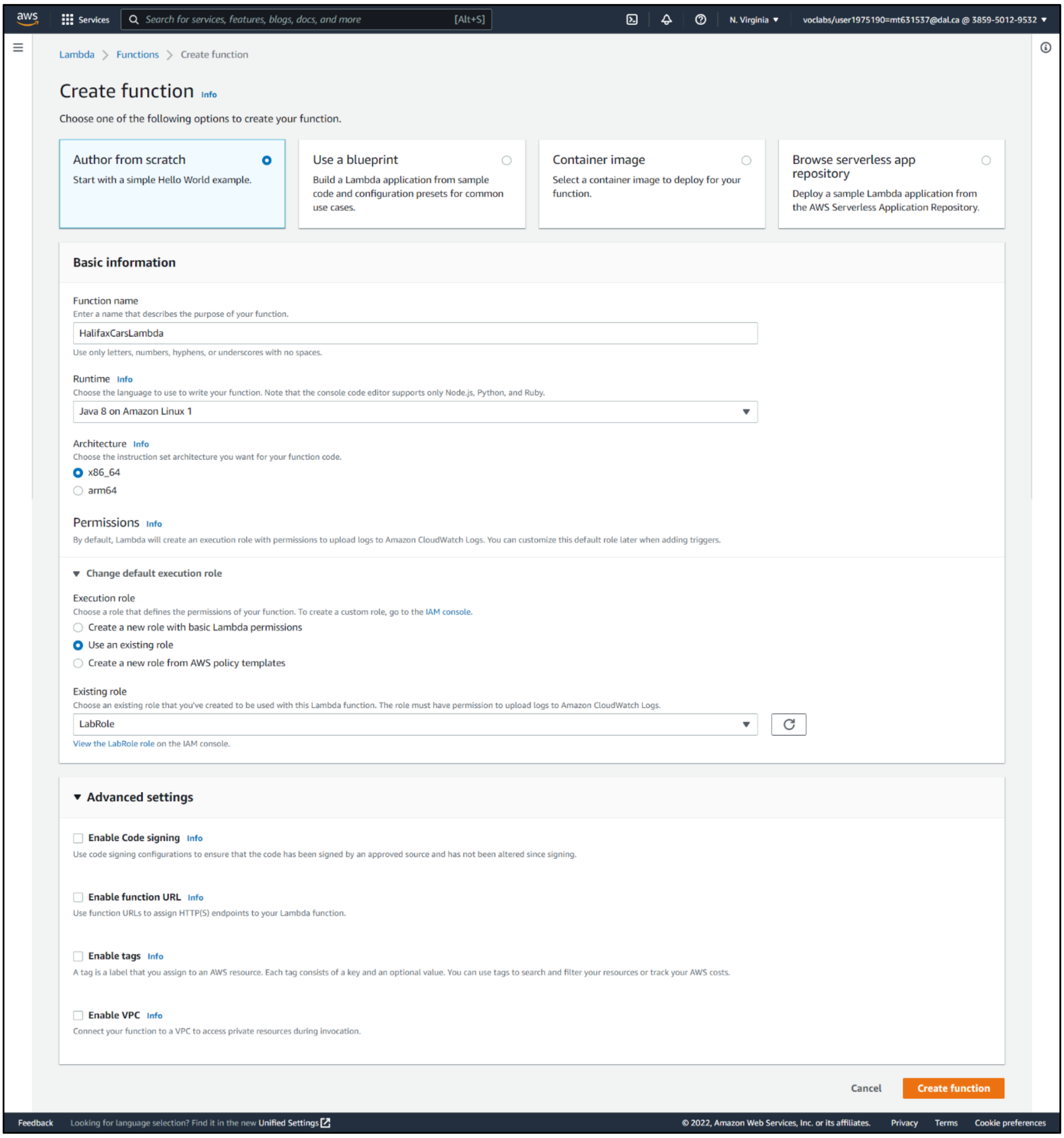

aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532
Amazon SQS > Queues
Edit
Delete
Send and receive messages
Actions
Create queue
Search queues by prefix
1
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Queues (1)

	Name	Type	Created	Messages available	Messages in flight	Encryption	Content-based deduplication
○	HalifaxCarsSQS	Standard	7/17/2022, 14:15:16 PDT	0	0	Disabled	-

*Figure 4:Successful creation of **HalifaxCarsSQS** queue*

Figure 6 is responsible for showing the creation of Lambda Function with name **HalifaxCarsLambda.**


aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532
Lambda > Functions > Create function

Create function Info

Choose one of the following options to create your function.

● Author from scratch
Start with a simple Hello World example.

○ Use a blueprint
Build a Lambda application from sample code and configuration presets for common use cases.

○ Container image
Select a container image to deploy for your function.

○ Browse serverless app repository
Deploy a sample Lambda application from the AWS Serverless Application Repository.

Basic information

Function name
Enter a name that describes the purpose of your function.
HalifaxCarsLambda
Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Java 8 on Amazon Linux 1

Architecture Info
Choose the instruction set architecture you want for your function code.
● x86_64
○ arm64

Permissions Info
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

▼ Change default execution role

Execution role
Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console.
○ Create a new role with basic Lambda permissions
● Use an existing role
○ Create a new role from AWS policy templates

Existing role
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.
LabRole
View the LabRole role on the IAM console.

▼ Advanced settings

□ Enable Code signing Info
Use code signing configurations to ensure that the code has been signed by an approved source and has not been altered since signing.

□ Enable function URL Info
Use function URLs to assign HTTP(S) endpoints to your Lambda function.

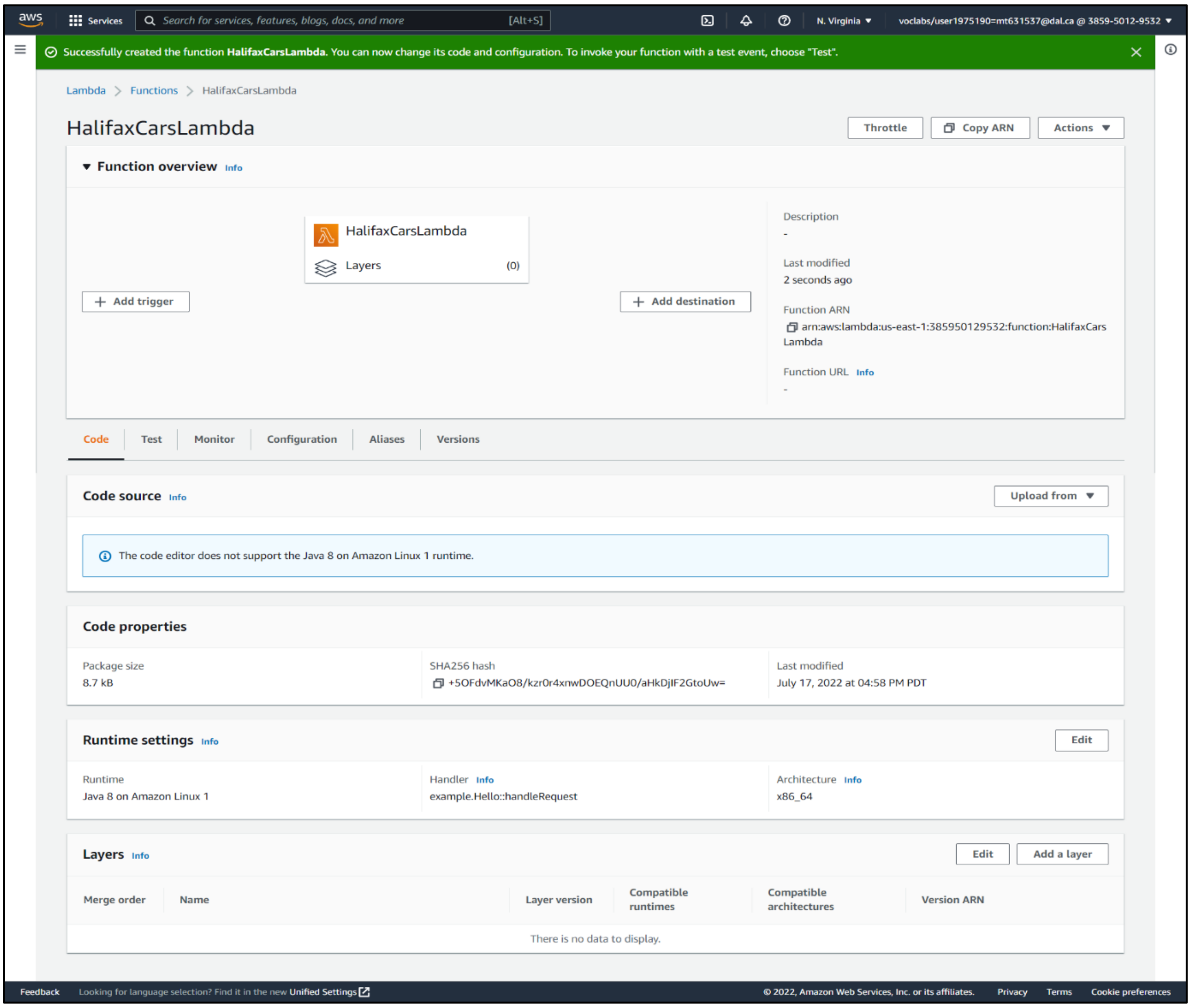
□ Enable tags Info
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

□ Enable VPC Info
Connect your function to a VPC to access private resources during invocation.

Cancel
Create function
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

*Figure 5: Creation of Lambda Function with name **HalifaxCarsLambda***

Figure 7 is responsible for showing the successful creation of **HalifaxCarsLambda** lambda function



Successfully created the function **HalifaxCarsLambda**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

Lambda > Functions > HalifaxCarsLambda

HalifaxCarsLambda

Function overview

Description
-

Last modified
2 seconds ago

Function ARN
arn:aws:lambda:us-east-1:385950129532:function:HalifaxCarsLambda

Function URL
-

Code source

The code editor does not support the Java 8 on Amazon Linux 1 runtime.

Code properties

Package size	SHA256 hash	Last modified
8.7 kB	+5OFdvMKaO8/kzr0r4xnwDOEQnUU0/aHkDjIF2GtoUw=	July 17, 2022 at 04:58 PM PDT

Runtime settings

Runtime	Handler	Architecture
Java 8 on Amazon Linux 1	example.Hello::handleRequest	x86_64

Layers

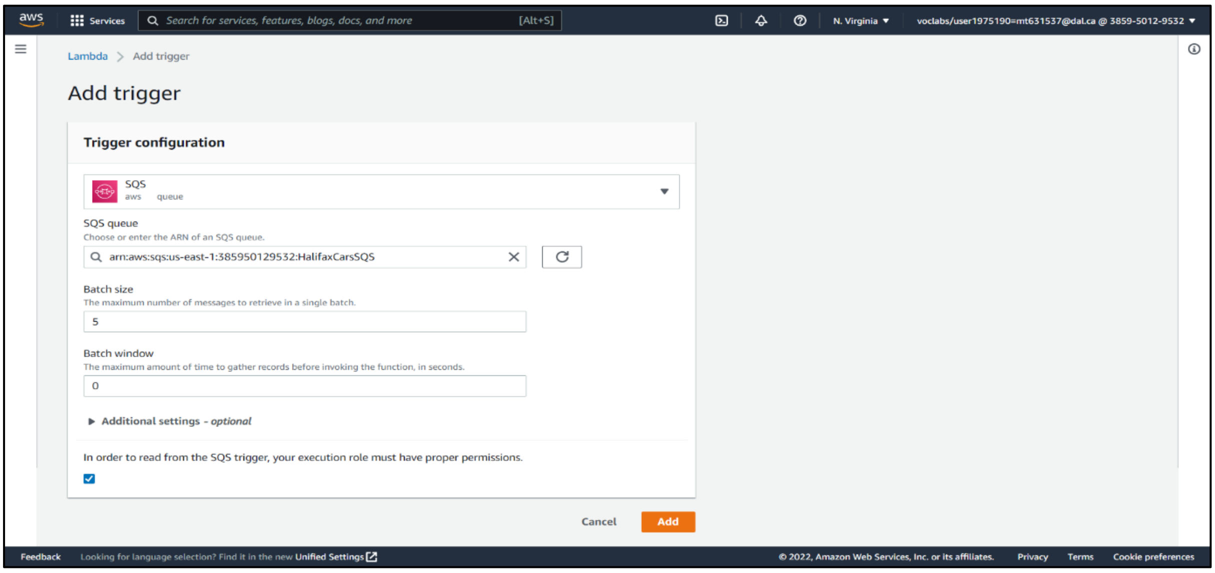
Merge order	Name	Layer version	Compatible runtimes	Compatible architectures	Version ARN

There is no data to display.

aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532
Throttle
Copy ARN
Actions
Info
HalifaxCarsLambda
Layers
(0)
Add trigger
Add destination
Code
Test
Monitor
Configuration
Aliases
Versions
Upload from
Edit
Add a layer
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

*Figure 6: Successful creation of **HalifaxCarsLambda** lambda function*

Figure 8 is responsible for showing successful creation **HalifaxCarsSQS** trigger set to the lambda function **HalifaxCarsLambda** Lambda Function.



Lambda > Add trigger

Add trigger

Trigger configuration

SQS

SQS queue
Choose or enter the ARN of an SQS queue.
arn:aws:sqs:us-east-1:385950129532:HalifaxCarsSQS

Batch size
The maximum number of messages to retrieve in a single batch.
5

Batch window
The maximum amount of time to gather records before invoking the function, in seconds.
0

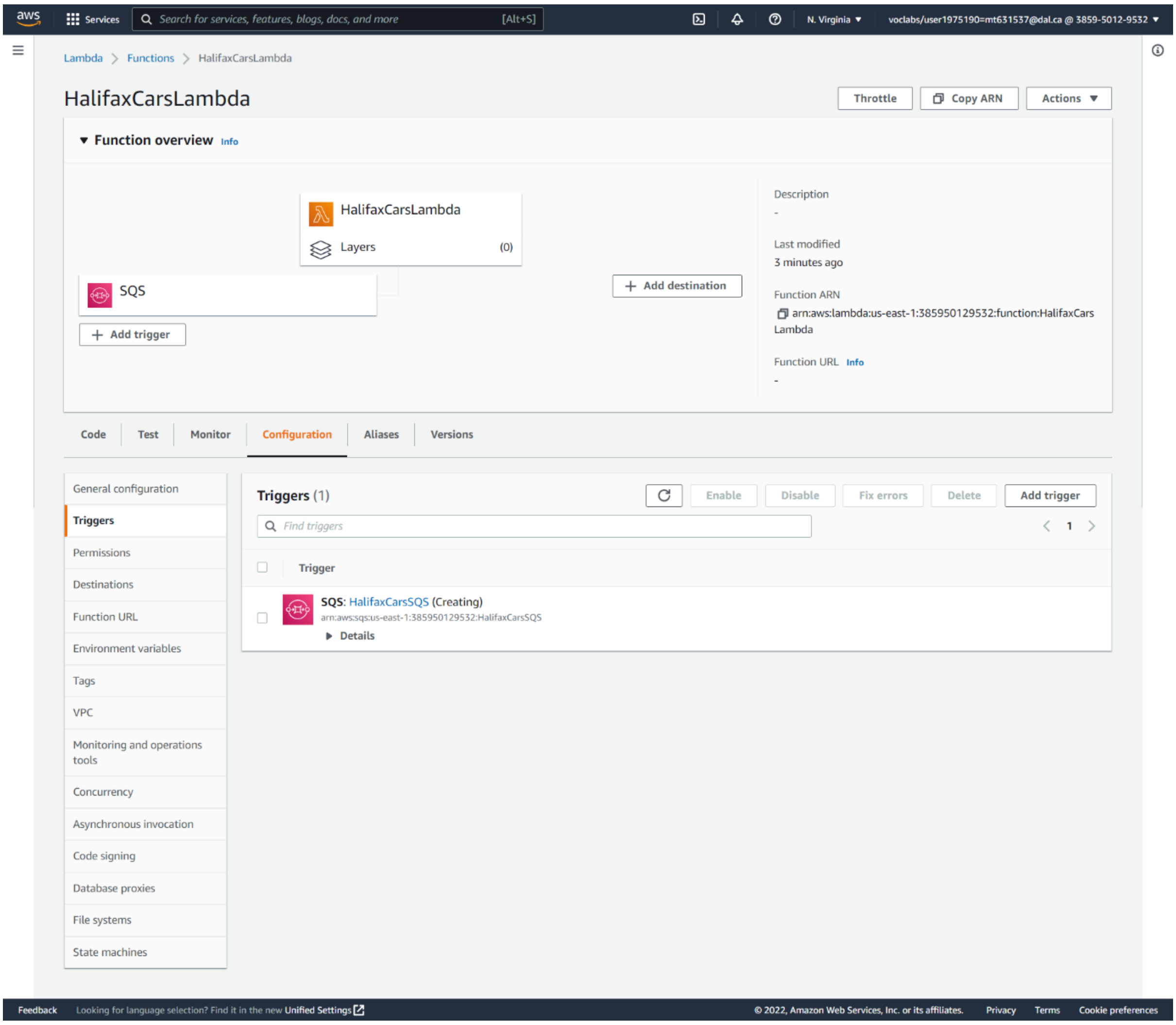
Additional settings - *optional*

In order to read from the SQS trigger, your execution role must have proper permissions.
☑

aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532
aws
queue
Cancel
Add
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Figure 7: HalifaxCarsSQS dashboard displaying the trigger HalifaxCarsLambda set to it

Figure 9 is responsible for showing the dash of the lambda function **HalifaxCarsLambda** Function along with the trigger **HalifaxCarsSQS** trigger.



HalifaxCarsLambda

Description
-

Last modified
3 minutes ago

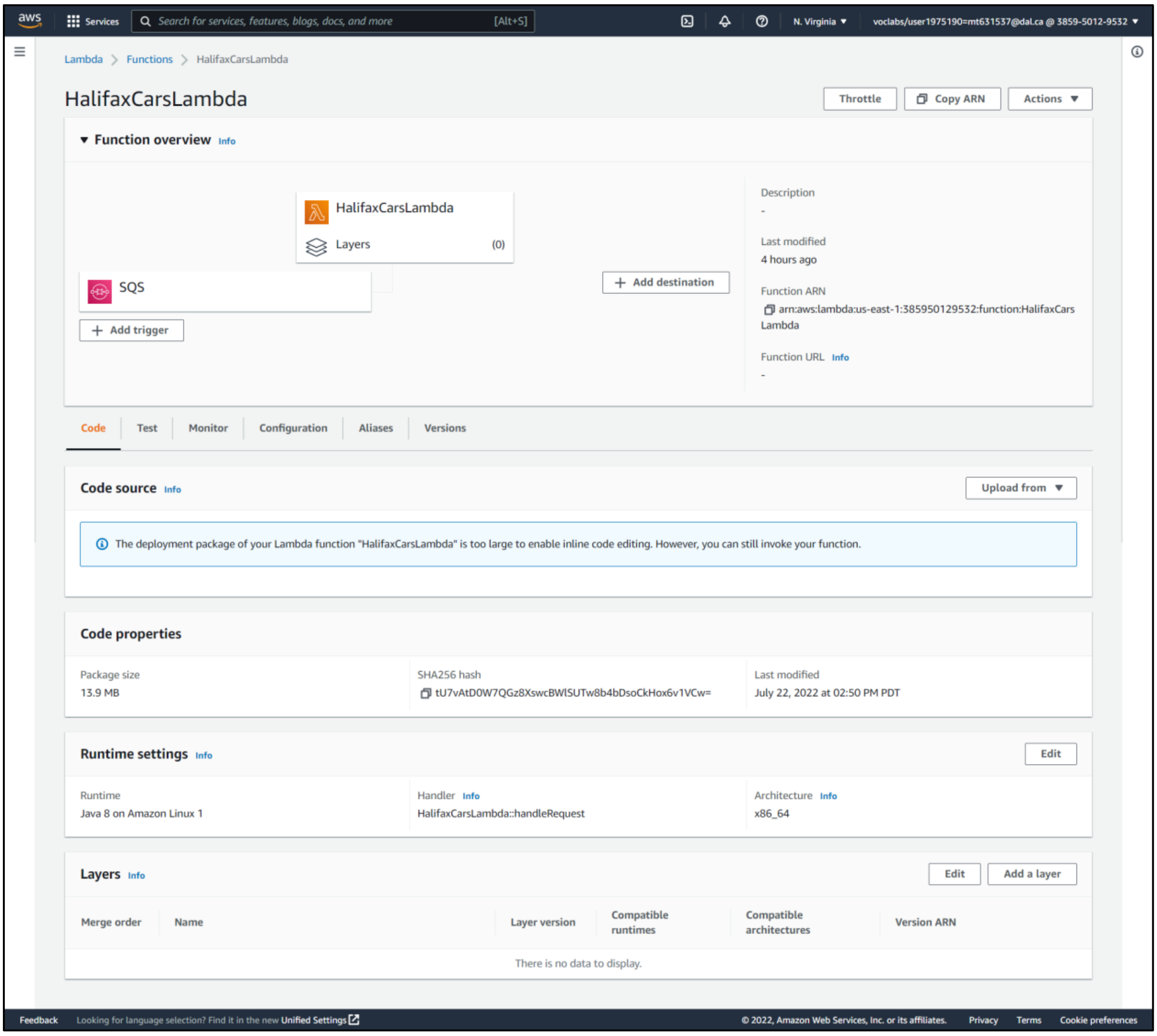
Function ARN
arn:aws:lambda:us-east-1:385950129532:function:HalifaxCarsLambda

Function URL Info
-

SQS: HalifaxCarsSQS (Creating)
arn:aws:sqs:us-east-1:385950129532:HalifaxCarsSQS

Figure 8: HalifaxCarsSQS dashboard displaying the trigger HalifaxCarsLambda set to it

Figure 10 is responsible for showing **HalifaxCarsLambda** configuration page with SQS trigger set to it


aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia ▼
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532 ▼
Lambda > Functions > HalifaxCarsLambda
Throttle
Copy ARN
Actions ▼
HalifaxCarsLambda
Layers (0)
SQS
+ Add trigger
+ Add destination
Code
Test
Monitor
Configuration
Aliases
Versions
Upload from ▼
Edit
Edit
Add a layer
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

HalifaxCarsLambda

▼ Function overview Info

Description
-

Last modified
4 hours ago

Function ARN
arn:aws:lambda:us-east-1:385950129532:function:HalifaxCarsLambda

Function URL Info
-

Code source Info

ⓘ The deployment package of your Lambda function "HalifaxCarsLambda" is too large to enable inline code editing. However, you can still invoke your function.

Code properties

Package size	SHA256 hash	Last modified
13.9 MB	tU7vAtD0W7QGz8XswcBWlSUTw8b4bDsoCkHox6v1VCw=	July 22, 2022 at 02:50 PM PDT

Runtime settings Info

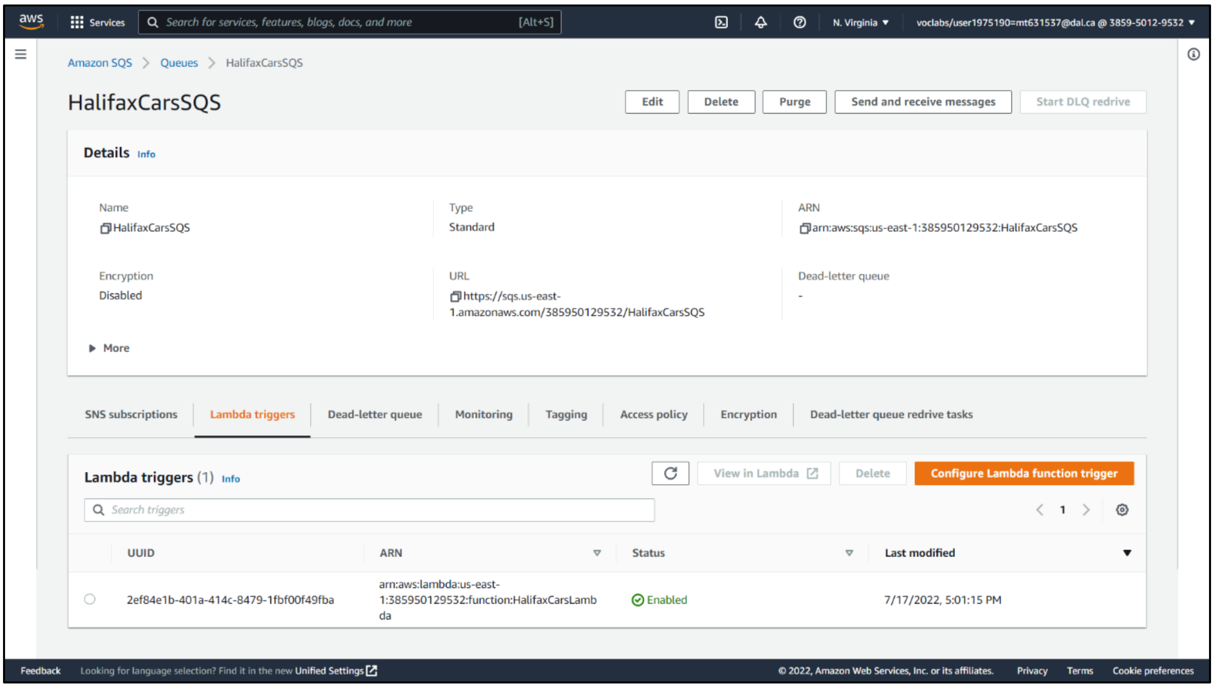
Runtime	Handler Info	Architecture Info
Java 8 on Amazon Linux 1	HalifaxCarsLambda::handleRequest	x86_64

Layers Info

Merge order	Name	Layer version	Compatible runtimes	Compatible architectures	Version ARN
There is no data to display.					

*Figure 9: **HalifaxCarsLambda** function Dashboard*

Figiure 11 is responsible for representing the **HalifaxCarsSQS** dashboard with the trigger **HalifaxCarsLambda**


aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532
Amazon SQS > Queues > HalifaxCarsSQS
Edit
Delete
Purge
Send and receive messages
Start DLQ redrive
More
SNS subscriptions
Lambda triggers
Dead-letter queue
Monitoring
Tagging
Access policy
Encryption
Dead-letter queue redrive tasks
View in Lambda
Delete
Configure Lambda function trigger
Search triggers
1
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

HalifaxCarsSQS

Details Info

Name	Type	ARN
HalifaxCarsSQS	Standard	arn:aws:sqs:us-east-1:385950129532:HalifaxCarsSQS

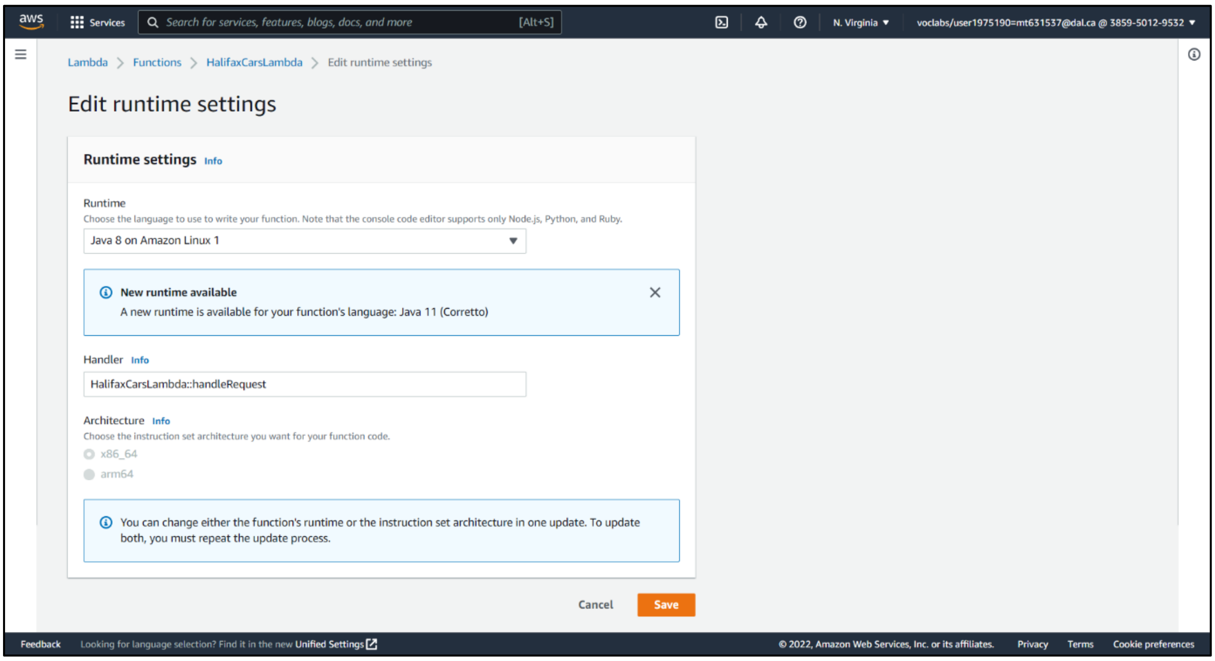
Encryption	URL	Dead-letter queue
Disabled	https://sqs.us-east-1.amazonaws.com/385950129532/HalifaxCarsSQS	-

Lambda triggers (1) Info

	UUID	ARN	Status	Last modified
○	2ef84e1b-401a-414c-8479-1fbf00f49fba	arn:aws:lambda:us-east-1:385950129532:function:HalifaxCarsLambda	Enabled	7/17/2022, 5:01:15 PM

*Figure 10: **HalifaxCarsSQS** dashboard with the trigger **HalifaxCarsLambda***

Figure 12 is responsible for displaying the “Edit runtime settings” of **HalifaxCarsLambda** function with the HalifaxCarsLamnda::handleRequest set to it


aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532
Lambda > Functions > HalifaxCarsLambda > Edit runtime settings
Cancel
Save
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Edit runtime settings

Runtime settings Info

Runtime
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Java 8 on Amazon Linux 1

New runtime available
A new runtime is available for your function's language: Java 11 (Corretto)

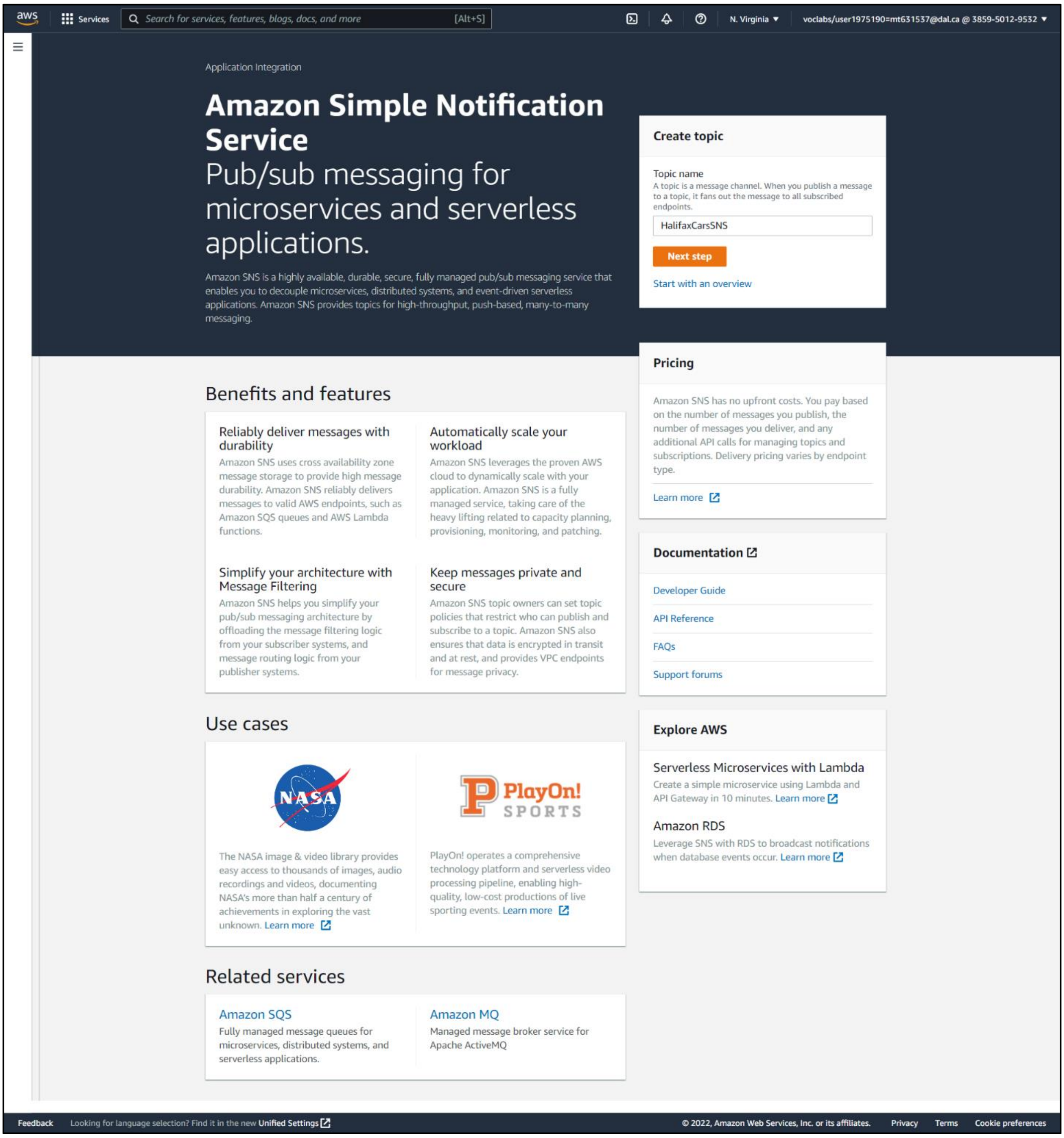
Handler Info
HalifaxCarsLambda::handleRequest

Architecture Info
Choose the instruction set architecture you want for your function code.
○ x86_64
○ arm64

You can change either the function's runtime or the instruction set architecture in one update. To update both, you must repeat the update process.

Figure 11: HalifaxCarsLambda function with the HalifaxCarsLamnda::handleRequest set to it

Figure 13 is responsible for showing the Amazon SNS start page with “Create Topic” Page.



aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532

Application Integration

Amazon Simple Notification Service

Pub/sub messaging for microservices and serverless applications.

Amazon SNS is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and event-driven serverless applications. Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging.

Create topic

Topic name

A topic is a message channel. When you publish a message to a topic, it fans out the message to all subscribed endpoints.

HalifaxCarsSNS

Pricing

Amazon SNS has no upfront costs. You pay based on the number of messages you publish, the number of messages you deliver, and any additional API calls for managing topics and subscriptions. Delivery pricing varies by endpoint type.

Benefits and features

Reliably deliver messages with durability
Amazon SNS uses cross availability zone message storage to provide high message durability. Amazon SNS reliably delivers messages to valid AWS endpoints, such as Amazon SQS queues and AWS Lambda functions.

Automatically scale your workload
Amazon SNS leverages the proven AWS cloud to dynamically scale with your application. Amazon SNS is a fully managed service, taking care of the heavy lifting related to capacity planning, provisioning, monitoring, and patching.

Simplify your architecture with Message Filtering
Amazon SNS helps you simplify your pub/sub messaging architecture by offloading the message filtering logic from your subscriber systems, and message routing logic from your publisher systems.

Keep messages private and secure
Amazon SNS topic owners can set topic policies that restrict who can publish and subscribe to a topic. Amazon SNS also ensures that data is encrypted in transit and at rest, and provides VPC endpoints for message privacy.

Explore AWS

Serverless Microservices with Lambda
Create a simple microservice using Lambda and API Gateway in 10 minutes. Learn more

Amazon RDS
Leverage SNS with RDS to broadcast notifications when database events occur. Learn more

Use cases

The NASA image & video library provides easy access to thousands of images, audio recordings and videos, documenting NASA's more than half a century of achievements in exploring the vast unknown. Learn more

PlayOn! operates a comprehensive technology platform and serverless video processing pipeline, enabling high-quality, low-cost productions of live sporting events. Learn more

Related services

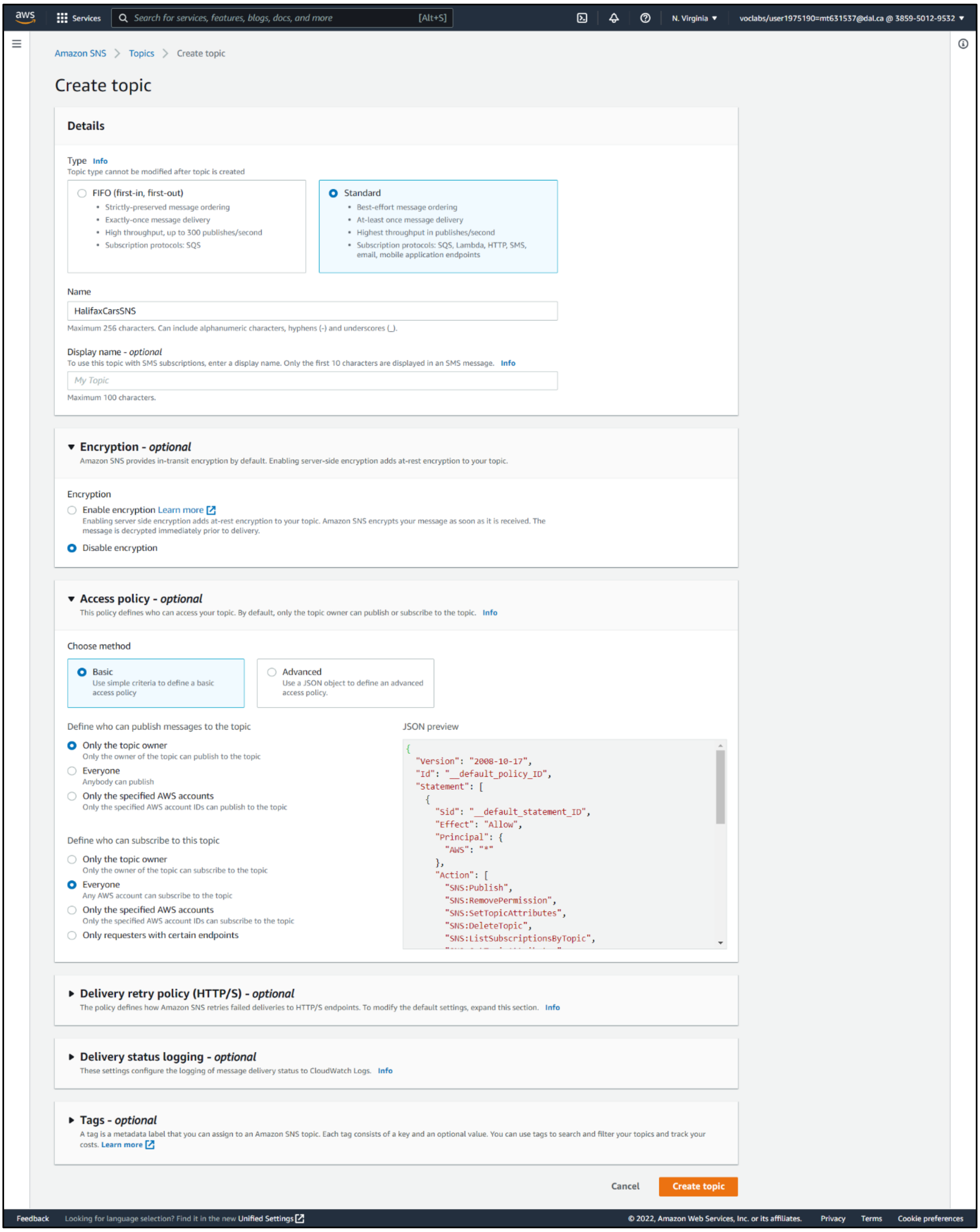
Amazon SQS
Fully managed message queues for microservices, distributed systems, and serverless applications.

Amazon MQ
Managed message broker service for Apache ActiveMQ

Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Figure 12: Amazon SNS start page with “Create Topic” Page.

Figure 14 is responsible for showing the configuration page for Amazon SNS topic.


aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia ▾
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532 ▾
Amazon SNS > Topics > Create topic

Create topic

Details

Type Info
Topic type cannot be modified after topic is created

○ FIFO (first-in, first-out)
- Strictly-preserved message ordering
- Exactly-once message delivery
- High throughput, up to 300 publishes/second
- Subscription protocols: SQS

● Standard
- Best-effort message ordering
- At-least once message delivery
- Highest throughput in publishes/second
- Subscription protocols: SQS, Lambda, HTTP, SMS, email, mobile application endpoints

Name
HalifaxCarsSNS
Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_).

Display name - *optional*
To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message. Info
My Topic
Maximum 100 characters.

▼ **Encryption - *optional***
Amazon SNS provides in-transit encryption by default. Enabling server-side encryption adds at-rest encryption to your topic.

Encryption
○ Enable encryption Learn more
Enabling server side encryption adds at-rest encryption to your topic. Amazon SNS encrypts your message as soon as it is received. The message is decrypted immediately prior to delivery.
● Disable encryption

▼ **Access policy - *optional***
This policy defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic. Info

Choose method
● Basic
Use simple criteria to define a basic access policy
○ Advanced
Use a JSON object to define an advanced access policy.

Define who can publish messages to the topic
● Only the topic owner
Only the owner of the topic can publish to the topic
○ Everyone
Anybody can publish
○ Only the specified AWS accounts
Only the specified AWS account IDs can publish to the topic

Define who can subscribe to this topic
○ Only the topic owner
Only the owner of the topic can subscribe to the topic
● Everyone
Any AWS account can subscribe to the topic
○ Only the specified AWS accounts
Only the specified AWS account IDs can subscribe to the topic
○ Only requesters with certain endpoints

JSON preview

```
{
  "Version": "2008-10-17",
  "Id": "__default_policy_ID",
  "Statement": [
    {
      "Sid": "__default_statement_ID",
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "SNS:Publish",
        "SNS:RemovePermission",
        "SNS:SetTopicAttributes",
        "SNS:DeleteTopic",
        "SNS:ListSubscriptionsByTopic",
```

▶ **Delivery retry policy (HTTP/S) - *optional***
The policy defines how Amazon SNS retries failed deliveries to HTTP/S endpoints. To modify the default settings, expand this section. Info

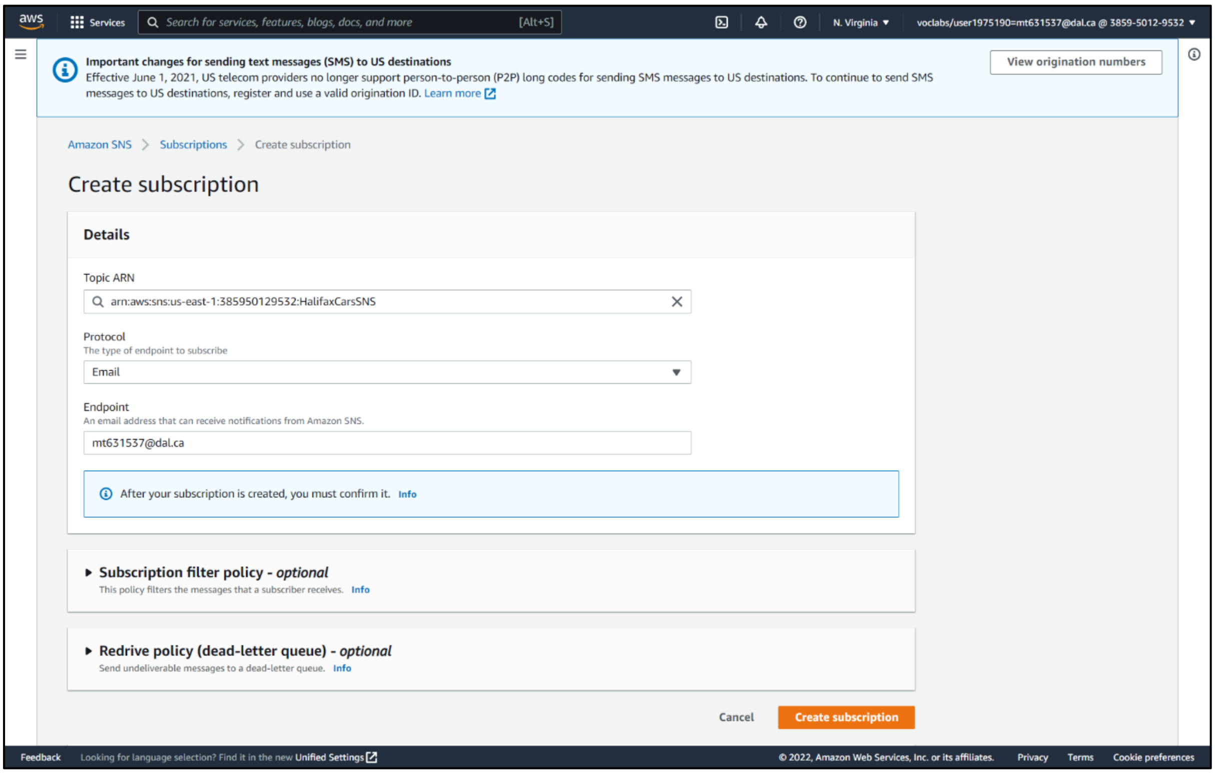
▶ **Delivery status logging - *optional***
These settings configure the logging of message delivery status to CloudWatch Logs. Info

▶ **Tags - *optional***
A tag is a metadata label that you can assign to an Amazon SNS topic. Each tag consists of a key and an optional value. You can use tags to search and filter your topics and track your costs. Learn more

Cancel
Create topic
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Figure 13: Configuration page for Amazon SNS topic

Figure 15 is responsible for showing the **HalifaxCarsSNS's** subscription page. The endpoint is configured to mt631537@dal.ca email as I want the order to come on this email.



Important changes for sending text messages (SMS) to US destinations
Effective June 1, 2021, US telecom providers no longer support person-to-person (P2P) long codes for sending SMS messages to US destinations. To continue to send SMS messages to US destinations, register and use a valid origination ID. Learn more

Amazon SNS > Subscriptions > Create subscription

Create subscription

Details

Topic ARN: arn:aws:sns:us-east-1:385950129532:HalifaxCarsSNS

Protocol
The type of endpoint to subscribe
Email

Endpoint
An email address that can receive notifications from Amazon SNS.
mt631537@dal.ca

After your subscription is created, you must confirm it. Info

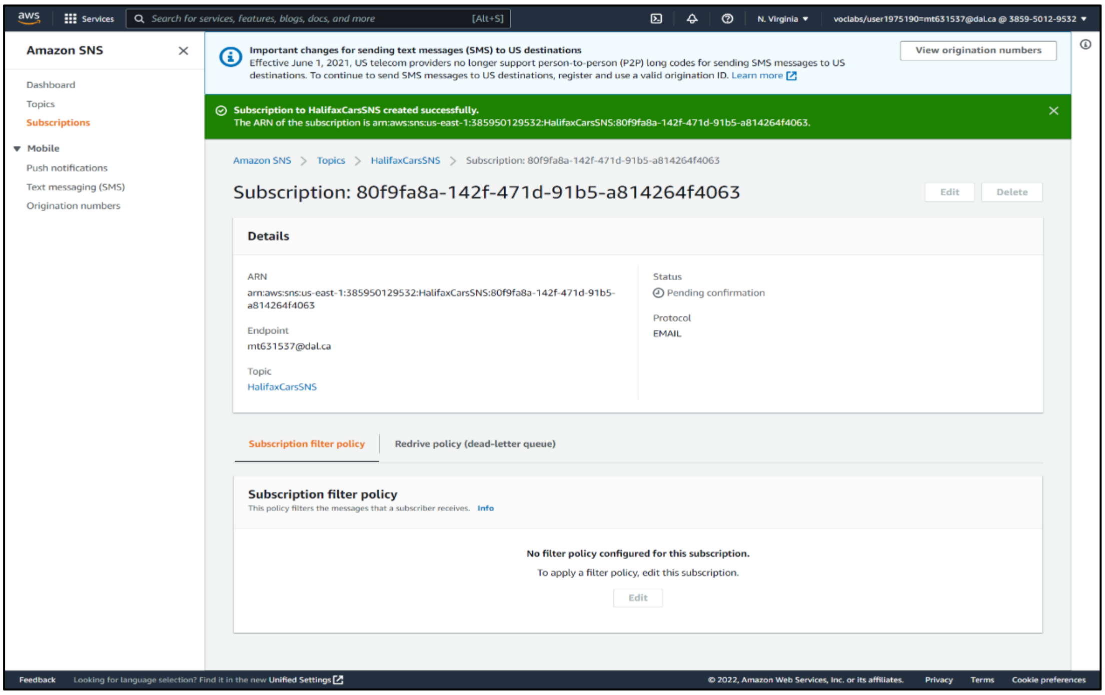
Subscription filter policy - *optional*
This policy filters the messages that a subscriber receives. Info

Redrive policy (dead-letter queue) - *optional*
Send undeliverable messages to a dead-letter queue. Info

aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532
View origination numbers
Cancel
Create subscription
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Figure 14: Create subscription page with the protocol set to email address mt631537@dal.ca

Figure 16 and 17 is responsible for successful creation of a subscription and currently the status is at "Pending Confirmation"



Important changes for sending text messages (SMS) to US destinations
Effective June 1, 2021, US telecom providers no longer support person-to-person (P2P) long codes for sending SMS messages to US destinations. To continue to send SMS messages to US destinations, register and use a valid origination ID. Learn more

Subscription to HalifaxCarsSNS created successfully.
The ARN of the subscription is arn:aws:sns:us-east-1:385950129532:HalifaxCarsSNS:80f9fa8a-142f-471d-91b5-a814264f4063.

Amazon SNS > Topics > HalifaxCarsSNS > Subscription: 80f9fa8a-142f-471d-91b5-a814264f4063

Subscription: 80f9fa8a-142f-471d-91b5-a814264f4063

Details

ARN: arn:aws:sns:us-east-1:385950129532:HalifaxCarsSNS:80f9fa8a-142f-471d-91b5-a814264f4063

Endpoint: mt631537@dal.ca

Topic: HalifaxCarsSNS

Status: Pending confirmation

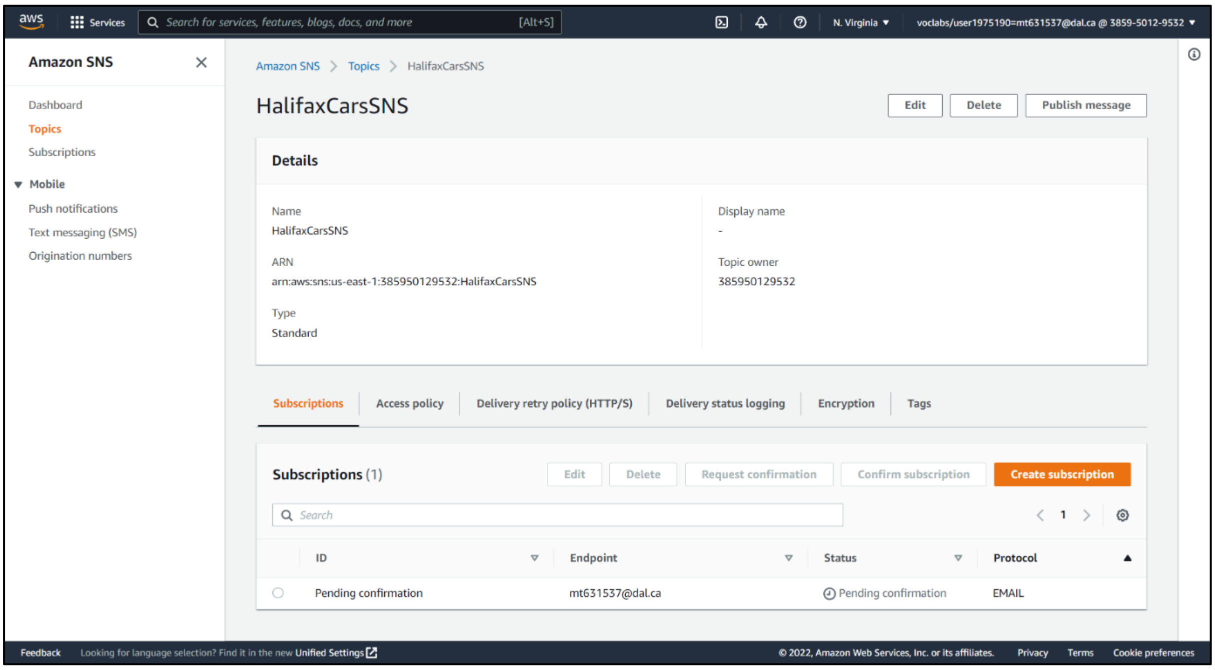
Protocol: EMAIL

Subscription filter policy
This policy filters the messages that a subscriber receives. Info

No filter policy configured for this subscription.
To apply a filter policy, edit this subscription.

aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532
Amazon SNS
Dashboard
Topics
Subscriptions
Mobile
Push notifications
Text messaging (SMS)
Origination numbers
View origination numbers
Edit
Delete
Subscription filter policy
Redrive policy (dead-letter queue)
Edit
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Figure 15: Successful creation of a subscription with status "Pending Confirmation"



HalifaxCarsSNS

Details

Name
HalifaxCarsSNS

Display name
-

ARN
arn:aws:sns:us-east-1:385950129532:HalifaxCarsSNS

Topic owner
385950129532

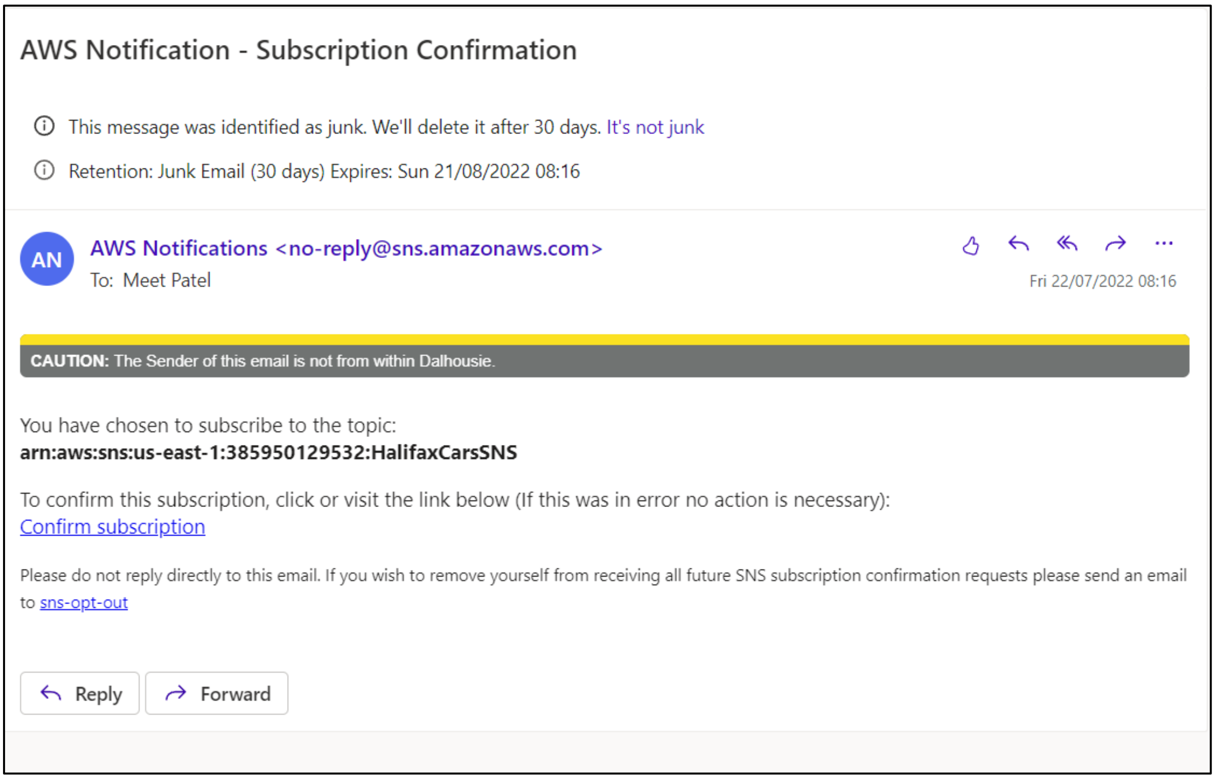
Type
Standard

Subscriptions (1)

	ID	Endpoint	Status	Protocol
○	Pending confirmation	mt631537@dal.ca	Pending confirmation	EMAIL

Figure 16: : Successful creation of a subscription with status "Pending Confirmation"

Figure 18 is responsible for showing the subscription email ai the provided the email id: mt631537@dal.ca



AWS Notification - Subscription Confirmation

This message was identified as junk. We'll delete it after 30 days. It's not junk

Retention: Junk Email (30 days) Expires: Sun 21/08/2022 08:16

AWS Notifications <no-reply@sns.amazonaws.com>
To: Meet Patel
Fri 22/07/2022 08:16

CAUTION: The Sender of this email is not from within Dalhousie.

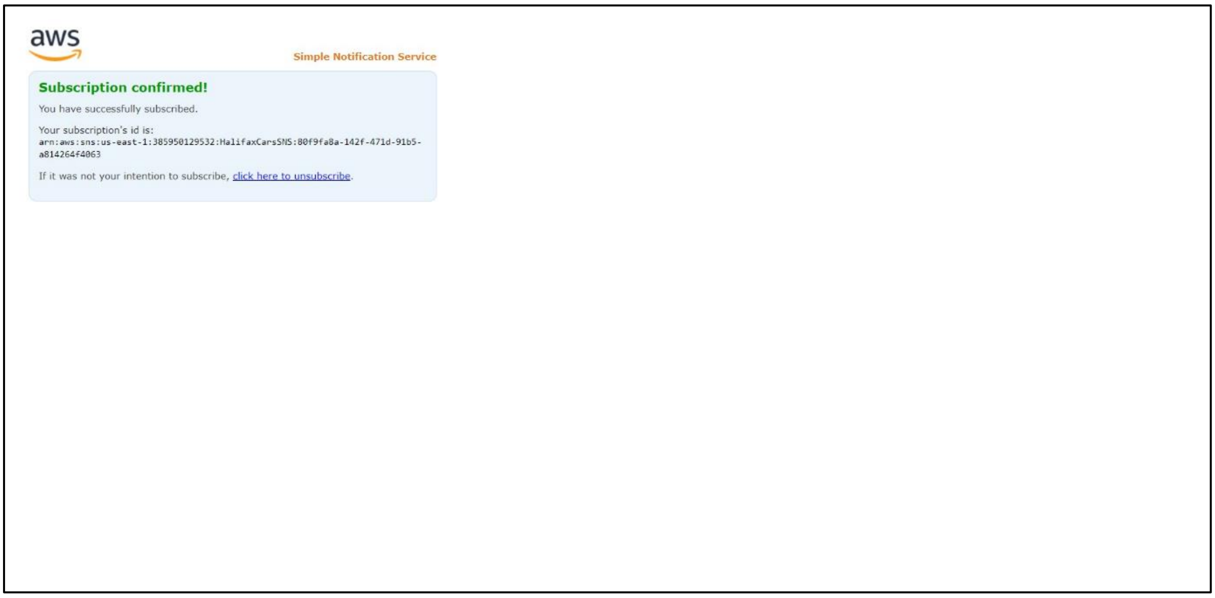
You have chosen to subscribe to the topic:
arn:aws:sns:us-east-1:385950129532:HalifaxCarsSNS

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):
Confirm subscription

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to sns-opt-out

Figure 17; Subscription email ai the provided the email id: mt631537@dal.ca

Figure 19 is responsible for showing the confirmation page of the subscription



Subscription confirmed!

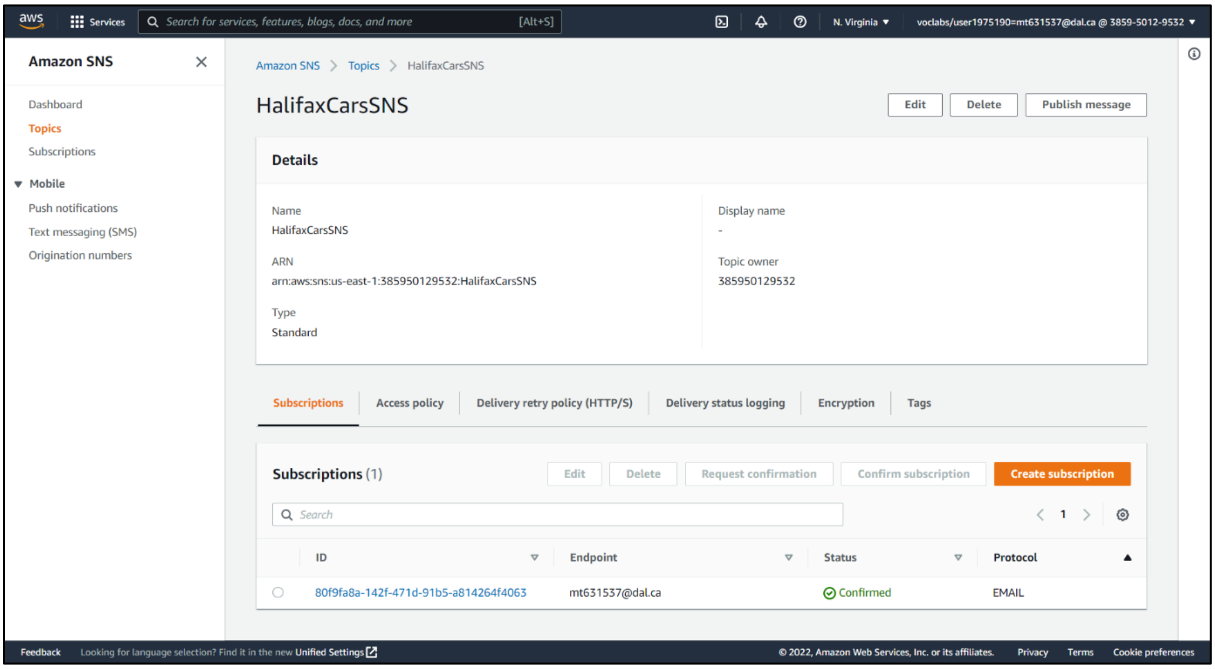
You have successfully subscribed.

Your subscription's id is:
arn:aws:sns:us-east-1:385950129532:HalifaxCarsSNS:80f9fa8a-142f-471d-91b5-a814264f4063

If it was not your intention to subscribe, click here to unsubscribe.

Figure 18:Confirmation page of the subscription

Figure 20 is responsible for showing the **HalifaxCarsSNS** dashboard with confirmed status for emai id mt631537@dal.ca



HalifaxCarsSNS

Details

Name
HalifaxCarsSNS

ARN
arn:aws:sns:us-east-1:385950129532:HalifaxCarsSNS

Type
Standard

Display name
-

Topic owner
385950129532

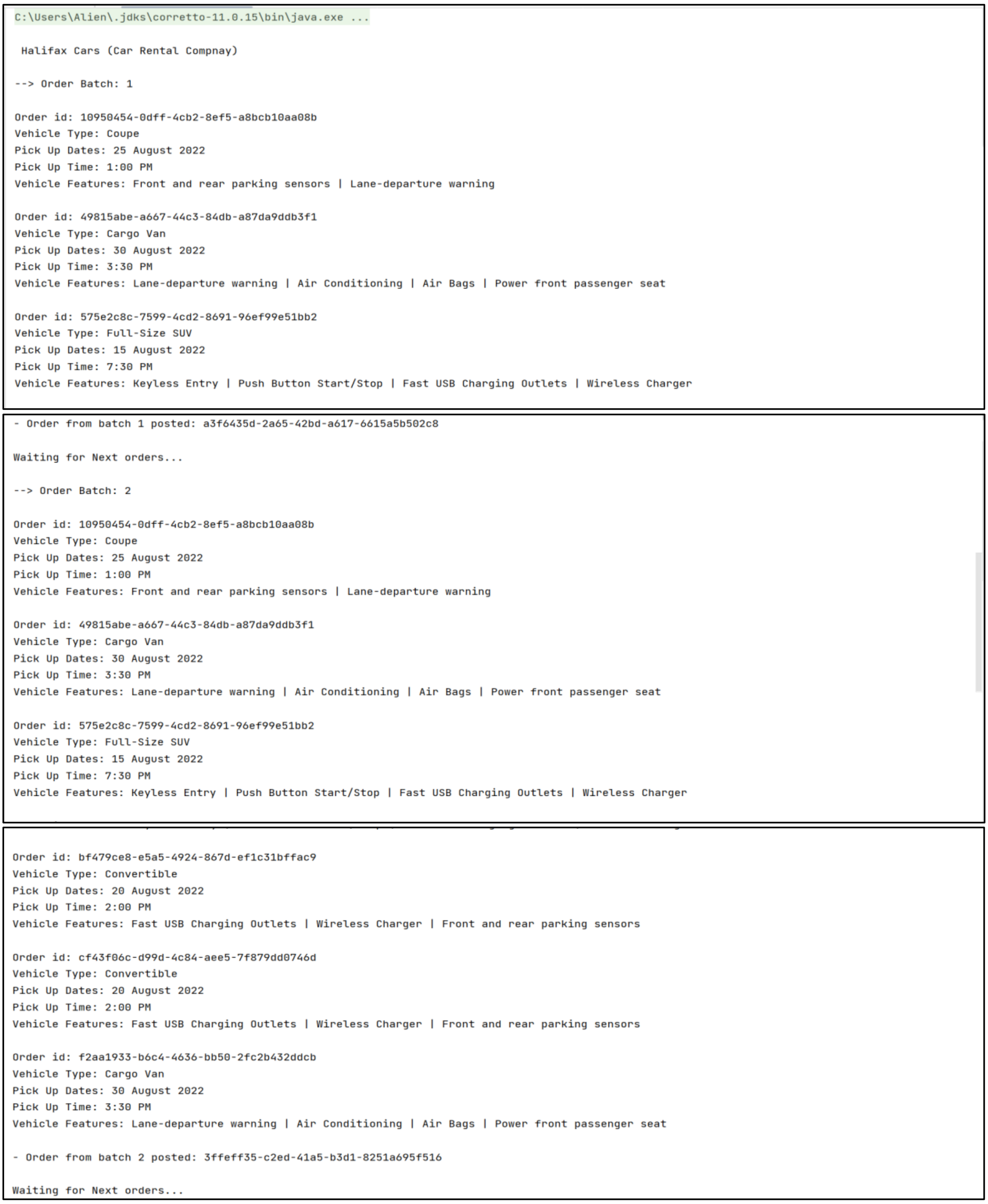
Subscriptions (1)

	ID	Endpoint	Status	Protocol
○	80f9fa8a-142f-471d-91b5-a814264f4063	mt631537@dal.ca	✓ Confirmed	EMAIL

Figure 19: HalifaxCarsSNS dashboard with confirmed status for email id mt631537@dal.ca

Program Output

Figure 21 below shows the program execution. Orders are sent in batches and each batch consist of 2 to 6 orders. Here **batch 1** consist of 3 orders whereas **batch 2** consist of 6 orders.



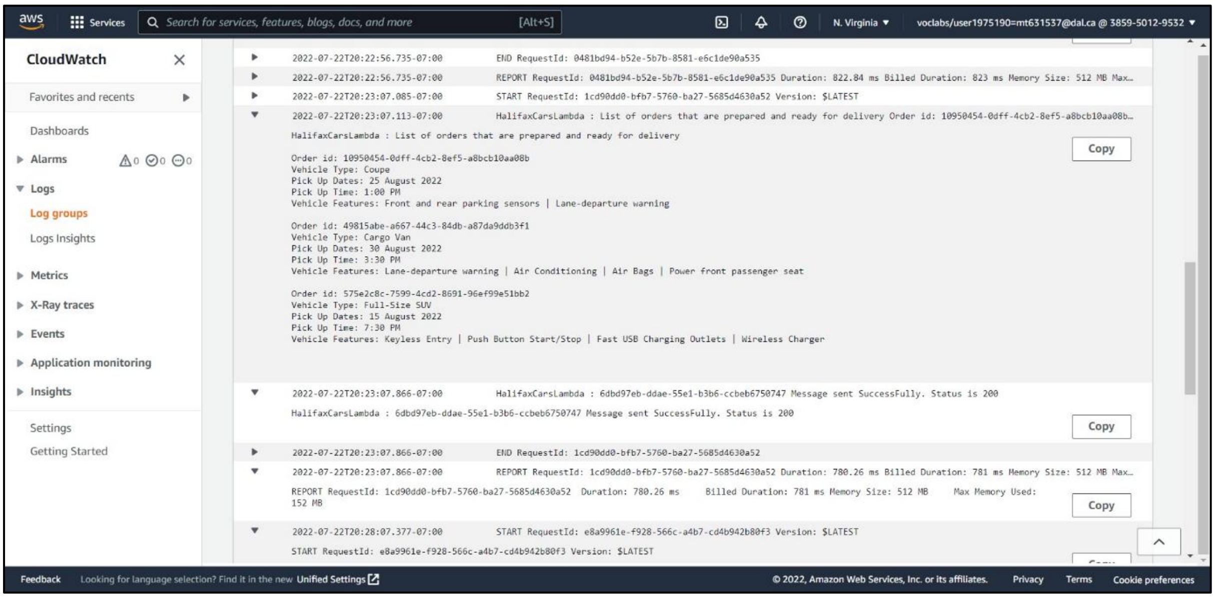
```
C:\Users\Alien\.jdks\corretto-11.0.15\bin\java.exe ...

 Halifax Cars (Car Rental Compnay)

--> Order Batch: 1

Order id: 10950454-0dff-4cb2-8ef5-a8bcb10aa08b
Vehicle Type: Coupe
Pick Up Dates: 25 August 2022
Pick Up Time: 1:00 PM
Vehicle Features: Front and rear parking sensors | Lane-departure warning

Order id: 49815abe-a667-44c3-84db-a87da9ddb3f1
Vehicle Type: Cargo Van
Pick Up Dates: 30 August 2022
Pick Up Time: 3:30 PM
Vehicle Features: Lane-departure warning | Air Conditioning | Air Bags | Power front passenger seat

Order id: 575e2c8c-7599-4cd2-8691-96ef99e51bb2
Vehicle Type: Full-Size SUV
Pick Up Dates: 15 August 2022
Pick Up Time: 7:30 PM
Vehicle Features: Keyless Entry | Push Button Start/Stop | Fast USB Charging Outlets | Wireless Charger
```

```
- Order from batch 1 posted: a3f6435d-2a65-42bd-a617-6615a5b502c8

Waiting for Next orders...

--> Order Batch: 2

Order id: 10950454-0dff-4cb2-8ef5-a8bcb10aa08b
Vehicle Type: Coupe
Pick Up Dates: 25 August 2022
Pick Up Time: 1:00 PM
Vehicle Features: Front and rear parking sensors | Lane-departure warning

Order id: 49815abe-a667-44c3-84db-a87da9ddb3f1
Vehicle Type: Cargo Van
Pick Up Dates: 30 August 2022
Pick Up Time: 3:30 PM
Vehicle Features: Lane-departure warning | Air Conditioning | Air Bags | Power front passenger seat

Order id: 575e2c8c-7599-4cd2-8691-96ef99e51bb2
Vehicle Type: Full-Size SUV
Pick Up Dates: 15 August 2022
Pick Up Time: 7:30 PM
Vehicle Features: Keyless Entry | Push Button Start/Stop | Fast USB Charging Outlets | Wireless Charger
```

```
Order id: bf479ce8-e5a5-4924-867d-ef1c31bffac9
Vehicle Type: Convertible
Pick Up Dates: 20 August 2022
Pick Up Time: 2:00 PM
Vehicle Features: Fast USB Charging Outlets | Wireless Charger | Front and rear parking sensors

Order id: cf43f06c-d99d-4c84-aee5-7f879dd0746d
Vehicle Type: Convertible
Pick Up Dates: 20 August 2022
Pick Up Time: 2:00 PM
Vehicle Features: Fast USB Charging Outlets | Wireless Charger | Front and rear parking sensors

Order id: f2aa1933-b6c4-4636-bb50-2fc2b432ddcb
Vehicle Type: Cargo Van
Pick Up Dates: 30 August 2022
Pick Up Time: 3:30 PM
Vehicle Features: Lane-departure warning | Air Conditioning | Air Bags | Power front passenger seat

- Order from batch 2 posted: 3ffeff35-c2ed-41a5-b3d1-8251a695f516

Waiting for Next orders...
```

Figure 20: Program execution output

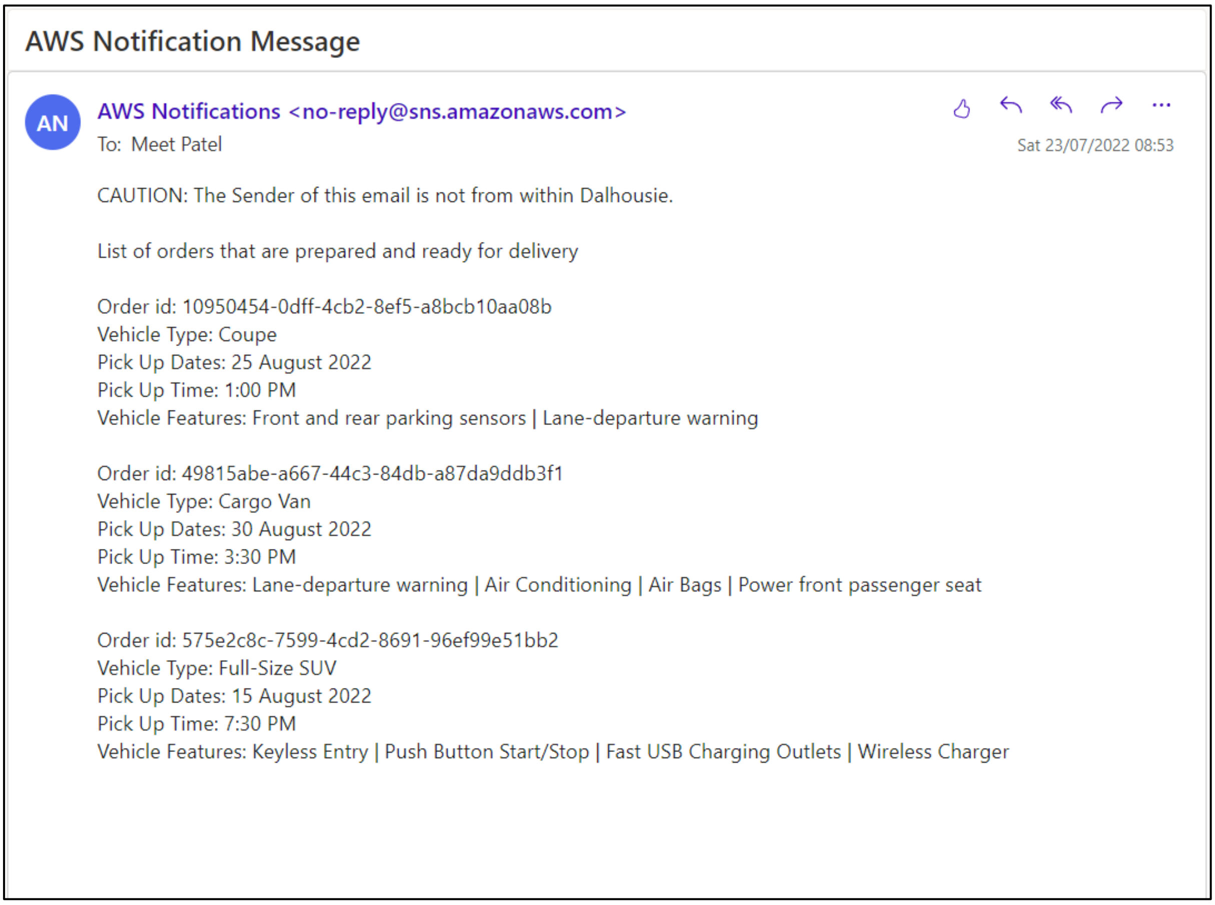
Figure 22 is responsible for showing the logs recorded when lambda function **HalifaxCarsLambda** was triggered.


aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
voclabs/user1975190=mt631537@dal.ca @ 3859-5012-9532
CloudWatch
Favorites and recents
Dashboards
Alarms
Logs
Log groups
Logs Insights
Metrics
X-Ray traces
Events
Application monitoring
Insights
Settings
Getting Started
Copy
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

```
2022-07-22T20:22:56.735-07:00    END RequestId: 0481bd94-b52e-5b7b-8581-e6c1de90a535
2022-07-22T20:22:56.735-07:00    REPORT RequestId: 0481bd94-b52e-5b7b-8581-e6c1de90a535 Duration: 822.84 ms Billed Duration: 823 ms Memory Size: 512 MB Max…
2022-07-22T20:23:07.085-07:00    START RequestId: 1cd90dd0-bfb7-5760-ba27-5685d4630a52 Version: $LATEST
2022-07-22T20:23:07.113-07:00    HalifaxCarsLambda : List of orders that are prepared and ready for delivery Order id: 10950454-0dff-4cb2-8ef5-a8bcb10aa08b…
HalifaxCarsLambda : List of orders that are prepared and ready for delivery

Order id: 10950454-0dff-4cb2-8ef5-a8bcb10aa08b
Vehicle Type: Coupe
Pick Up Dates: 25 August 2022
Pick Up Time: 1:00 PM
Vehicle Features: Front and rear parking sensors | Lane-departure warning

Order id: 49815abe-a667-44c3-84db-a87da9ddb3f1
Vehicle Type: Cargo Van
Pick Up Dates: 30 August 2022
Pick Up Time: 3:30 PM
Vehicle Features: Lane-departure warning | Air Conditioning | Air Bags | Power front passenger seat

Order id: 575e2c8c-7599-4cd2-8691-96ef99e51bb2
Vehicle Type: Full-Size SUV
Pick Up Dates: 15 August 2022
Pick Up Time: 7:30 PM
Vehicle Features: Keyless Entry | Push Button Start/Stop | Fast USB Charging Outlets | Wireless Charger

2022-07-22T20:23:07.866-07:00    HalifaxCarsLambda : 6dbd97eb-ddae-55e1-b3b6-ccbeb6750747 Message sent SuccessFully. Status is 200
HalifaxCarsLambda : 6dbd97eb-ddae-55e1-b3b6-ccbeb6750747 Message sent SuccessFully. Status is 200
2022-07-22T20:23:07.866-07:00    END RequestId: 1cd90dd0-bfb7-5760-ba27-5685d4630a52
2022-07-22T20:23:07.866-07:00    REPORT RequestId: 1cd90dd0-bfb7-5760-ba27-5685d4630a52 Duration: 780.26 ms Billed Duration: 781 ms Memory Size: 512 MB Max…
REPORT RequestId: 1cd90dd0-bfb7-5760-ba27-5685d4630a52  Duration: 780.26 ms     Billed Duration: 781 ms Memory Size: 512 MB     Max Memory Used:
152 MB
2022-07-22T20:28:07.377-07:00    START RequestId: e8a9961e-f928-566c-a4b7-cd4b942b80f3 Version: $LATEST
START RequestId: e8a9961e-f928-566c-a4b7-cd4b942b80f3 Version: $LATEST
```

Figure 21: logs recorded when lambda function HalifaxCarsLambda was triggered

Figure 23 and 24 is representing the email received



AWS Notification Message

AN AWS Notifications <no-reply@sns.amazonaws.com>
To: Meet Patel
Sat 23/07/2022 08:53

CAUTION: The Sender of this email is not from within Dalhousie.

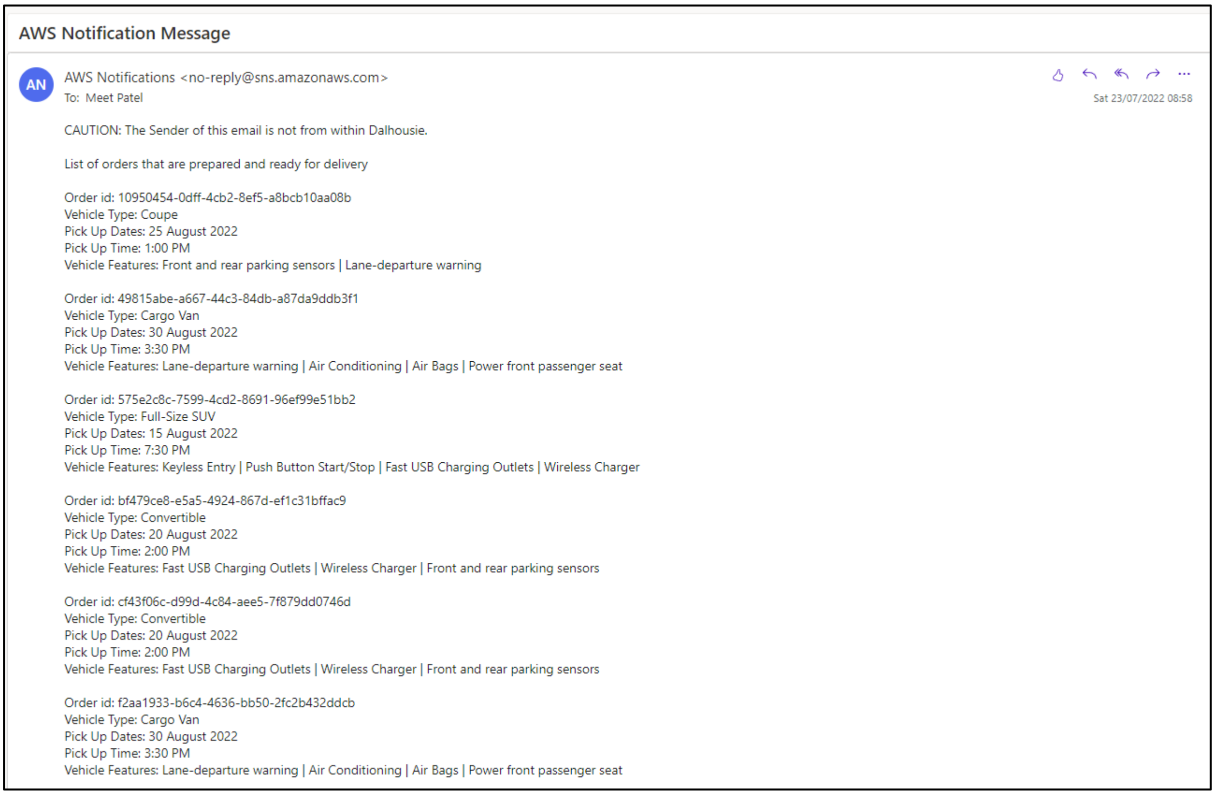
List of orders that are prepared and ready for delivery

Order id: 10950454-0dff-4cb2-8ef5-a8bcb10aa08b
Vehicle Type: Coupe
Pick Up Dates: 25 August 2022
Pick Up Time: 1:00 PM
Vehicle Features: Front and rear parking sensors | Lane-departure warning

Order id: 49815abe-a667-44c3-84db-a87da9ddb3f1
Vehicle Type: Cargo Van
Pick Up Dates: 30 August 2022
Pick Up Time: 3:30 PM
Vehicle Features: Lane-departure warning | Air Conditioning | Air Bags | Power front passenger seat

Order id: 575e2c8c-7599-4cd2-8691-96ef99e51bb2
Vehicle Type: Full-Size SUV
Pick Up Dates: 15 August 2022
Pick Up Time: 7:30 PM
Vehicle Features: Keyless Entry | Push Button Start/Stop | Fast USB Charging Outlets | Wireless Charger

Figure 22: Email Received



AWS Notification Message

AWS Notifications <no-reply@sns.amazonaws.com>
To: Meet Patel

Sat 23/07/2022 08:58

CAUTION: The Sender of this email is not from within Dalhousie.

List of orders that are prepared and ready for delivery

Order id: 10950454-0dff-4cb2-8ef5-a8bcb10aa08b
Vehicle Type: Coupe
Pick Up Dates: 25 August 2022
Pick Up Time: 1:00 PM
Vehicle Features: Front and rear parking sensors | Lane-departure warning

Order id: 49815abe-a667-44c3-84db-a87da9ddb3f1
Vehicle Type: Cargo Van
Pick Up Dates: 30 August 2022
Pick Up Time: 3:30 PM
Vehicle Features: Lane-departure warning | Air Conditioning | Air Bags | Power front passenger seat

Order id: 575e2c8c-7599-4cd2-8691-96ef99e51bb2
Vehicle Type: Full-Size SUV
Pick Up Dates: 15 August 2022
Pick Up Time: 7:30 PM
Vehicle Features: Keyless Entry | Push Button Start/Stop | Fast USB Charging Outlets | Wireless Charger

Order id: bf479ce8-e5a5-4924-867d-ef1c31bffac9
Vehicle Type: Convertible
Pick Up Dates: 20 August 2022
Pick Up Time: 2:00 PM
Vehicle Features: Fast USB Charging Outlets | Wireless Charger | Front and rear parking sensors

Order id: cf43f06c-d99d-4c84-aee5-7f879dd0746d
Vehicle Type: Convertible
Pick Up Dates: 20 August 2022
Pick Up Time: 2:00 PM
Vehicle Features: Fast USB Charging Outlets | Wireless Charger | Front and rear parking sensors

Order id: f2aa1933-b6c4-4636-bb50-2fc2b432ddcb
Vehicle Type: Cargo Van
Pick Up Dates: 30 August 2022
Pick Up Time: 3:30 PM
Vehicle Features: Lane-departure warning | Air Conditioning | Air Bags | Power front passenger seat

Figure 23: Email Received

Program Script

Application Files

AWSConnection.java

```
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;

public class AWSConnection {
    private static final String AWS_ACCESS_KEY_ID = "ASIAVTXDLTV6LD3JP7XP";
    private static final String AWS_SECRET_ACCESS_KEY = "oAfnE+mL5PbyTSlq3cQznVZaz74m9z/fp8sdZsFd";
    private static final String AWS_SESSION_TOKEN = "FwoGZXIvYXdzEDMaDJuUS8TVX1ZwJ0pu7yLAAY37ekZ" +
            "/k27osh/TAmAR8b+EBrpYzEvl1pBH9mljWJKqeTgTBsdFXsG1oRPLEi5htvCthxbiSDIaokheN" +
            "k1g0AXr9e5eupa2hNuFvMRQvjkYxsruBUE27DrxIE68gR747QitPQV8UttqGWOpbQkZXHF2CzQ" +
            "ZCs9gQR+YMXWKwcz+2LVV5tJYm4nAwsG8ybY+0cVW6bz4/cSWVrtF/LCVsRfh+Pga1Cs0PAkgeA" +
            "ZnBhVITNg66Gy7RF2LoZVBlPH7ayjtp+2WBjItHAXixqxz3I98j9WRvJhDlrr7n7fvDrI/XwgAoD" +
            "AkNjADj56PH1mxrwcxt5nc";
    private static final BasicSessionCredentials AWS_CREDENTIALS = new BasicSessionCredentials(AWS_ACCESS_KEY_ID,
AWS_SECRET_ACCESS_KEY, AWS_SESSION_TOKEN);

    public AmazonSQS createAmazonSQSClientBuilder() {
        return AmazonSQSClientBuilder.standard()
                .withCredentials(new AWSStaticCredentialsProvider(AWS_CREDENTIALS))
                .withRegion(Regions.US_EAST_1)
                .build();
    }
}
```

OrderModel.java

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

public class OrderModel {
    private List<VehicleListModel> vehicleListModels;
    private List<String> dates;

    private List<String> times;

    public OrderModel(ArrayList<VehicleListModel> vehicleListModels, ArrayList<String> dates, ArrayList<String>
times) {
        this.vehicleListModels = vehicleListModels;
        this.dates = dates;
        this.times = times;
    }
    public void addVehicleListModels(VehicleListModel vehicleListModel){
        vehicleListModels.add(vehicleListModel);
    }
    public List<VehicleListModel>getVehicleListModels(){
        return vehicleListModels;
    }

    public void addInDates(String date){
        dates.add(date);
    }

    public List<String> getDates() {
        return dates;
    }

    public void addInTime(String time){
        dates.add(time);
    }

    public List<String> getTimes() {
        return times;
    }
}
```

VehicleData.java

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

public class VehicleData {

    public static List<VehicleListModel> getVehicleList() {

        List<VehicleListModel> vehicleList = new ArrayList<>();

        vehicleList.add(new VehicleListModel("Small Pickup Truck",
                "Automatic TransmissionAir Conditioning" +
                        "| Air Bags | AM/FM Stereo", "25 July 2022", "12:00 PM"));

        vehicleList.add(new VehicleListModel("Mid Size SUV",
                " Air Bags | AM/FM Stereo " +
                        "| Touchscreen infotainment system | Panoramic Sunroof", "30 July 2022",
"8:00 AM"));

        vehicleList.add(new VehicleListModel("Large Pickup Truck",
                "Touchscreen infotainment system | Panoramic Sunroof " +
                        "| Apple CarPlay and Android Auto | Music System", "05 August 2022", "7:00
PM"));

        vehicleList.add(new VehicleListModel("7 Passenger Minivan",
                "Apple CarPlay and Android Auto | Music System" +
                        "| Keyless Entry | Push Button Start/Stop", "10 August 2022", "9:00 PM"));


        vehicleList.add(new VehicleListModel("Full-Size SUV",
                "Keyless Entry | Push Button Start/Stop " +
                        "| Fast USB Charging Outlets | Wireless Charger", "15 August 2022", "7:30
PM"));


        vehicleList.add(new VehicleListModel("Convertible",
                "Fast USB Charging Outlets | Wireless Charger " +
                        "| Front and rear parking sensors", "20 August 2022", "2:00 PM"));


        vehicleList.add(new VehicleListModel("Coupe",
                "Front and rear parking sensors " +
                        "| Lane-departure warning", "25 August 2022", "1:00 PM"));

        vehicleList.add(new VehicleListModel("Cargo Van",
                "Lane-departure warning | Air Conditioning " +
                        "| Air Bags | Power front passenger seat", "30 August 2022", "3:30 PM"));
        return vehicleList;
    }
}
```

VehicleListModel.java

```
public class VehicleListModel {
    private final String vehicleType;
    private final String vehicleDescription;
    private final String pickUpDate;

    private final String pickUpTime;

    public VehicleListModel(String vehicleType, String vehicleDescription, String pickUpDate, String
pickUpTime) {
        this.vehicleType = vehicleType;
        this.vehicleDescription = vehicleDescription;
        this.pickUpDate = pickUpDate;
        this.pickUpTime = pickUpTime;
    }

    public String getVehicleType() {
        return vehicleType;
    }
    public String getVehicleDescription() {
        return vehicleDescription;
    }
    public String getPickUpDate() {
        return pickUpDate;
    }

    public String getPickUpTime(){
        return pickUpTime;
    }
}
```

HalifaxCarsProgram.js

```
import com.amazonaws.services.sqs.AmazonSQS;
import com.amazonaws.services.sqs.model.GetQueueUrlRequest;
import com.amazonaws.services.sqs.model.SendMessageRequest;
import java.util.ArrayList;
import java.util.List;
import java.util.Random;
import java.util.UUID;

public class HalifaxCarsProgram {
    StringBuilder vehicleOrderMessage = new StringBuilder();
    public void processingVehicleOrder() {
        System.out.print("\n Halifax Cars (Car Rental Compnay)\n\n");
        AmazonSQS amazonSQSClientBuilder = new AWSConnection().createAmazonSQSClientBuilder();
        String SQSQueue = amazonSQSClientBuilder.getQueueUrl(new
GetQueueUrlRequest().withQueueName("HalifaxCarsSQS")).getQueueUrl();
        int orderBatch = 0;
        while (true) {
            System.out.print("--> Order Batch: " + orderBatch++ + "\n\n");
            for (int i = 1; i <= new Random().nextInt(5) + 2; ++i) {
                List<VehicleListModel> haliifaxCarsList = VehicleData.getVehicleList();
                OrderModel orderModel = new OrderModel(new ArrayList<>(), new ArrayList<>(), new
ArrayList<>());
                orderModel.addVehicleListModels(haliifaxCarsList.get(new
Random().nextInt(haliifaxCarsList.size())));
                StringBuilder sb = new StringBuilder();
                sb.append("Order id: ").append(UUID.randomUUID()).append("\n");
                for (int i1 = 0; i1 < orderModel.getVehicleListModels().size(); ++i1) {
                    sb.append("Vehicle Type:
").append(orderModel.getVehicleListModels().get(i1).getVehicleType()).append("\n");
                    sb.append("Pick Up Dates:
").append(orderModel.getVehicleListModels().get(i1).getPickUpDate()).append("\n");
                    sb.append("Pick Up Time:
").append(orderModel.getVehicleListModels().get(i1).getPickUpTime()).append("\n");
                    sb.append("Vehicle Features:
").append(orderModel.getVehicleListModels().get(i1).getVehicleDescription()).append("\n\n");
```

```
                }
                vehicleOrderMessage.append(sb);
            }
            System.out.print(vehicleOrderMessage);
            SendMessageRequest sendMessageRequest = new SendMessageRequest()
                    .withQueueUrl(SQSQueue)
                    .withMessageBody(vehicleOrderMessage.toString());
            System.out.print("- Order from batch " + orderBatch + " posted: " +
amazonSQSClientBuilder.sendMessage(sendMessageRequest).getMessageId() + "\n\n");
            try {
                System.out.print("Waiting for Next orders...\n\n");
                Thread.sleep(5 * 60 * 1000);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}
```

Main,java

```
public class Main {
    public static void main(String[] args) {
        HalifaxCarsProgram halifaxDineProgram = new HalifaxCarsProgram();
        halifaxDineProgram.processingVehicleOrder();
    }
}
```

HalifaxCarsLambda.java

```
import com.amazonaws.services.lambda.runtime.Context;
import com.amazonaws.services.lambda.runtime.RequestHandler;
import com.amazonaws.services.lambda.runtime.events.SQSEvent;
import com.amazonaws.services.sns.AmazonSNSClientBuilder;
import com.amazonaws.services.sns.model.AmazonSNSException;
import com.amazonaws.services.sns.model.CreateTopicRequest;
import com.amazonaws.services.sns.model.CreateTopicResult;
import com.amazonaws.services.sns.model.ListTopicsRequest;
import com.amazonaws.services.sns.model.PublishRequest;
import com.amazonaws.services.sns.model.Topic;
import java.util.List;

public final class HalifaxCarsLambda implements RequestHandler<SQSEvent, Void> {
    @Override
    public Void handleRequest(final SQSEvent sqsEvent,
                              final Context context) {
        StringBuilder emailContent = new StringBuilder();
        for (SQSEvent.SQSMessage sqsMessage : sqsEvent.getRecords()) {
            emailContent.append("List of orders that are prepared and ready for delivery")
                    .append("\n\n")
                    .append(sqsMessage.getBody())
                    .append("\n\n");;
        }
        context.getLogger().log(HalifaxCarsLambda.class.getSimpleName() + " : " + emailContent);
        try {
            String SNSTopicARN = null;
            boolean completed = false;
            List<Topic> topics = AmazonSNSClientBuilder.defaultClient().listTopics(new
ListTopicsRequest()).getTopics();
            for (Topic topic : topics) {
                String[] topicARNSplit = topic.getTopicArn().split(":");
                if (topicARNSplit[topicARNSplit.length - 1].equals("HalifaxCarsSNS")) {
                    SNSTopicARN = topic.getTopicArn();
                    completed = true;
                    break;
                }
            }
            SNSTopicARN = checkStatus(SNSTopicARN, completed);
            if (SNSTopicARN == null) return null;
            PublishRequest request = new PublishRequest();
            request.setMessage(emailContent.toString());
            request.setTopicArn(SNSTopicARN);
            context.getLogger().log(HalifaxCarsLambda.class.getSimpleName() + " : " +
                    AmazonSNSClientBuilder.defaultClient().publish(request)
                            .getMessageId() + " Message sent SuccessFully. " +
                    "Status is " +
AmazonSNSClientBuilder.defaultClient().publish(request).getSdkHttpMetadata().getHttpStatusCode());
        } catch (AmazonSNSException e) {
            e.printStackTrace();
        }
        return null;
    }

    private String checkStatus(String SNSTopicARN, boolean completed) {
        if (!completed) {
            CreateTopicResult result1 = AmazonSNSClientBuilder.defaultClient().createTopic(new
CreateTopicRequest().withName("HalifaxCarsSNS"));
            SNSTopicARN = result1.getTopicArn();
        }
        if (SNSTopicARN == null) {
            return null;
        }
        return SNSTopicARN;
    }
}
```

References

[1] AWS, "AWS Lambda," Amazon, [Online]. Available: https://aws.amazon.com/lambda/ [Accessed 22 July 2022].

[2] AWS, "Amazon Simple Queue Service," Amazon, [Online]. Available: https://aws.amazon.com/sqs/ [Accessed 22 July 2022].

[3] AWS, "Amazon Simple Notification Service," Amazon, [Online]. Available: https://aws.amazon.com/sns/ [Accessed 22 July 2022].

[5] AWS, "Amazon CloudWatch," Amazon, [Online]. Available: https://aws.amazon.com/cloudwatch/ [Accessed 22 July 2022]