

Real-Time Inventory Optimization Using AWS Lambda and Amazon Kinesis Configuration Manual

MSc Research Cloud Computing

Abhishek Yadav Student ID: 23153423

School of Computing National College of Ireland

Supervisor:Dr. Shivani Jaswal

National College of Ireland



MSc Project Submission Sheet

School of Computing

Student Name:	Abhishek Yadav		
Student ID:	23153423		
Programme:	Cloud Computing	Year:	2024
Module:	MSc Research Project		
Lecturer: Submission Due	Dr. Shivani Jaswal		
Date:	12/12/2024		
Project Title:	Real-Time Inventory Optimization Using A Amazon Kinesis	WS Lam	bda and

Word Count: Page Count: 11

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

<u>ALL</u> internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Abhishek Yadav
Date:	11/12/2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Abhishek Yadav Student ID: 23153423

1 Introduction

The configuration manual provides a step-by-step instruction to how to setup the cloud services for the inventory management services for the user to perform the application work in local live environment based on the lit review performed by various paper and analysing the methodology of done by different researchers to make the prediction and forecast.

2 Steps for Configure the steup

Certation of account :

Visit AWS Registration to step up the account and provide all relevant information. Now got to billing preferences to monitor costs chart.

Then setup of data Streaming with Kinesis

In the AWS Management Console search Kinesis and click on Kinesis Data Streams and follow the below mention steps.

aws Services Q Search	[Option+S] D 🗘 🤣 Ireland ▼ Abhishek Yadav •
Amazon Kinesis $ imes$	Amazon Kinesis > Data streams
Dashboard	Data streams (1) Info Process data in real time Create a Firehose stream Actions Create data stream
Data streams Amazon Data Firehose 🎦 New	Q. Find data streams < 1 > @
Managed Apache Flink 2 New Resources CloudFormation templates	Name▲StatusCapacity modeProvision ed shardsSharing policyData retention
AWS Glue Schema Registry [2]	□ inventry-data ⊘ Active On-demand - No 1 day

Figure1:Setup for Kinesis.

For creating the data stream:

- Click on create data stream.
- Give the name to your stream (RealTimeInventoryStream).
- In this we have to Specify the number of shards based.

Configure stream retention:

• Set the retention period (default is 24 hours).

Now moving to the intergradation of kinesis producers:

• Use the Kinesis SDK in your application to push inventory data into the stream.

Setting	Value	Editable after creation	
Capacity mode	On-demand	⊘ Yes	
Data retention period	1 day	⊘ Yes	
Server-side encryption	Disabled	⊘ Yes	
Monitoring enhanced metrics	Disabled	⊘ Yes	
Data stream sharing policy	No policy	⊘ Yes	
Tags - optional Info You can add tags to organize your AWS resour No tags associated with the Kinesis Data			

Figure2: Kinesis data configuration.

Setting up AWS Lambda for for function trigger.

First we need to create a lambda function for that go to AWS Lambda and click on create a functions.

aws	G Search		[Option+S]	<u>ک</u>	⑦ 皎 Ireland ▼ Al	ohishek Yadav 🔻
≡ La	mbda > Functions				Ū	9 2
Func	tions (3)		L	ast fetched 46 seconds ago	C Actions Create	function
Q Fi	ilter by tags and attributes or searc	h by keyword			< .	1 > 🕸
	Function name	▼ Description	▼ Package type ▼	Runtime	▼ Last modified	∇
	InventoryProcessor	-	Zip	Python 3.9	4 weeks ago	
	InventoryAdjustmentFunction	-	Zip	Python 3.9	2 weeks ago	
	CheckItemCount	-	Zip	Python 3.9	1 month ago	

Figure3:Lambda function creation.

- Give the function name (Inventory process data).
- select the programming language as we choose (Python).
- Now we can create kinesis trigger:

• click to add triggers button and select the service as kinesis.

\equiv Lambda $>$ Functions $>$ InventoryAdjustmentFunction	0 2 0
InventoryAdjustmentFunction	Throttle Copy ARN Actions V
▼ Function overview Info	Export to Infrastructure Composer Download V
Diagram Template	Description
InventoryAdjustme ntFunction Image: Layers (0) + Add trigger	+ Add destination Function URL Info

Figure4:Connection with trigger services.

• Here we can link the created which we have already created.

0 4 0

Trigger configuration Info
Kinesis aws analytics event-source-mapping polling streaming
Kinesis stream Select a Kinesis stream to listen for updates on. To select a stream in another shared AWS account, enter its Amazon Resource Name (ARN).
Q belect a stream in your AWS account or enter a shared ARN
inventry-data Select an optional consumer 🙆 of your stream to listen for updates on. To select a consumer in another shared AWS account, enter its ARN.
Q Select a consumer in your AWS account or enter a shared ARN
Event source mapping configuration
Activate trigger Select to activate the trigger now. Keep unchecked to create the trigger in a deactivated state for testing (recommended).
Enable metrics Monitor your event source with metrics. You can view those metrics in CloudWatch console. Enabling this feature incurs additional costs. Learn more

Figure 5: Kinesis configuration with lambda function.

Now we have to write a lambda code which is the function to process incoming data and store results in DynamoDB.

Co	de source Info		Ipload from 🔻
		> P InventoryAdjustmentFunction	
=	EXPLORER	♦ lambda_function.py ×	□ …
¢	 INVENTORYADJUSTMENTFUNCTION lambda_function.py 	Iambda_function.py I import boto3 Amazon Q Tip 1/3: Start typing to get suggestions ([ESC] to exit) import ison	HOLD P-
Q		3 from decimal import Decimal # Import Decimal for DynamoDB compatibility 4	de editor based on Code-C ource)
¢>		5 # Initialize the DynamoDB resource dynamodb = boto3.resource('dynamodb') # Fix: Initialize this resource at the global lo 7 sagemaker_runtime = boto3.client('sagemaker-runtime')	vel
₽		<pre>8 sns_client = boto3.client('sns') 9</pre>	Bill Chron Come
5	Deploy (소#U)	<pre>10 # Configuration 11 ENDPOINT_NAME = 'InventoryPredictionEndpoint'</pre>	
æ	Test (O≭i)	1 burduni junc = Inventory/relactionenpoint TABLE_UMME = 'Inventory/relaction SAFETY_STOCK_LEVEL = 50 # Minimum stock to avoid stockouts 00 VERSTOCK_THREEHOLE = 300 # Maximum stock threshold to avoid overstock REGNDER_POINT = 100 # Reorder when stock falls below this level 16 LEAD_TIRE_DAYS = 7 # Time it takes to restock 17 def lambda_handler(event, context):	
	✓ TEST EVENTS ✓ △ Private saved events InventoryTestEvent	19 try:	

Figure6:Fuction process of code.

Moving Forward Data Storage on DynamoDB and S3.

- 1. For setting up DynamoDB follow the below mention steps.
- Go to amazon DynamoDB and create the table.
- Give the name to the table (InventoryData).
- Now we have to set a partition key (ItemID).

reate table Table details info DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table. Table name This will be used to identify your table. Enter name for table Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (.). Partition key The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability. Enter the partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability. Enter the partition key is part of the table's primary key. The sort key allows you to sort or search among all items sharing the same partition key. Enter the sort key name In 255 characters and case sensitive. String ▼ In 255 characters and case sensitive. Determine the sort key name In 255 characters and case sensitive. String ▼ In 255 characters and case sensitive.	DynamoDB > Tables > Create table	· 도 (
DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table. Table name This will be used to identify your table. Enter name for table Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (,). Partition key The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability. Enter the partition key name 1 to 255 characters and case sensitive. Sort key - optional You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key. Enter the sort key name String Vou	eate table	
Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (.). Partition key The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability. Enter the partition key name 1 to 255 characters and case sensitive. Sort key - optional You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key. Enter the sort key name String	DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table. Table name	
Partition key The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability. Enter the partition key name String 1 to 255 characters and case sensitive. Sort key - optional You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key. Enter the sort key name String	Enter name for table	
The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability. Enter the partition key name String 1 to 255 characters and case sensitive. Sort key - optional You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key. Enter the sort key name String	Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (.).	
Sort key - optional You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key. Enter the sort key name String V	The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.	
You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key. Enter the sort key name String	1 to 255 characters and case sensitive.	
to 255 characters and rase sensitive	You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.	
	1 to 255 characters and case sensitive.	

Figure7:Creation of DynamoDB Table

- 2. For S3 Bucket setup:
- Go to S3 and Create Bucket.
- Give suitable name to the bucket (InventoryDataStorage).
- After that set permissions as public access.

Gener	al purpose buckets	Directory buckets		
Gene		ts (4) Info All AWS Regions	bucket	
Buckets	s are containers for data st	tored in S3.		
Q F	ind buckets by name			〈 1 〉 戀
	Name	▲ AWS Region ▼	IAM Access Analyzer	Creation date $\mathbf{\nabla}$
0	inventory-data-historica	L Europe (Ireland) eu-west-1	View analyzer for eu-west- 1	October 31, 2024, 08:26:23 (UTC+00:00)
0	inventrybucket	Europe (Ireland) eu-west-1	View analyzer for eu-west- 1	November 10, 2024, 22:00:42 (UTC+00:00)
0	sagemaker-eu-west-1- 703671932762	Europe (Ireland) eu-west-1	View analyzer for eu-west- 1	October 31, 2024, 08:48:33 (UTC+00:00)
0	sagemaker-studio- 703671932762- mgj7mbvn7k	Europe (Ireland) eu-west-1	View analyzer for eu-west- 1	October 31, 2024, 08:48:27 (UTC+00:00)

Figure8:Setting and creation of S3 bucket.

- 3. The Intergradation of S3 with Lambda:
 - Make the changes inside the Lambda function to save the processed data of inventory into the S3 bucket.

Machine Learning model integration with amazon SageMaker

- Go to Amazon SageMaker and click on create notebook instance.
- Name the instance "InventoryOptimizationModel".
- Select the instance which show type ml.t3.medium.

nazon SageMaker Al > Notebook instances > InventoryManagement Delete	Stop Open Jupyter	Open JupyterLab
Notebook instance settings		Edit
Name	Notebook instance type	
InventoryManagement	ml.t3.medium	
ARN	Elastic Inference	
arn:aws:sagemaker:eu-west-1:703671932762:notebook- instance/InventoryManagement	-	
. , ,	Volume Size	
Lifecycle configuration	5GB EBS	
	Platform identifier	
Status	Amazon Linux 2, Jupyter Lab 3	
⊘ InService	(notebook-al2-v2)	
Creation time	Minimum IMDS Version	
Oct 31, 2024 08:59 UTC	2	
Last updated		
Dec 09, 2024 13:39 UTC		

Figure9:Stage Maker juptyper notebook configuration.

- 1. To Preparing and training dataset for prediction follow the given steps.
- Use SageMaker to preprocess the data inside the notebook.

File Edit	/iew Insert Cell Kernel Widgets Help	Not Trusted	(MA)	conda_	python3
1 + % 4	▲ ▶ Run ■ C ▶ Code ♦ ☑ O nbdiff				
In [2]	<pre>: import pandas as pd import boto3 from stasmodels.tsa.arima.model import ARIMA import matplotlib.pyplot as plt s3 = boto3.client('s3') bucket_name = 'inventory-data-historical' file_name = 'historical_inventory.csv' s3.download_file(bucket_name, file_name, '/tmp/historical_inventory.csv') data = pd.read_csv('/tmp/historical_inventory.csv', parse_dates=['date'], index_col='date') print(data.head()) model = ARIMA(data['stockLevel'], order=(1,1,1)) model_fit = model.fit() forecast = model_fit.forecast(steps=10) print("Forecasted values:", forecast) data['stockLevel'].plot(label='Historical Stock Level') forecast.plot(label='Forecast', color='red') plt.legend() plt.show()</pre>				

Figure 10: Stage Maker jupyter notebook configuration with S3 bucket.

For Model training follow these steps.

• Use built-in algorithms Autoregressive Integrated Moving Average for train the model on inventory trends.

Trai	ning jobs Info		C	Actions v	Create training	j job
Q	Search training jobs				< 1 >) ()
	Name 🗢	Creation time 🔻	Duration	Job status ⊽	Warm pool status	Tim left
0	inventory-forecasting-job- copy-11-03	11/3/2024, 5:01:45 PM	2 minutes	⊘ Completed	-	-
0	inventory-forecasting-job	11/3/2024, 4:54:17 PM	2 minutes	Failed	-	-

Figure11:Stage Maker model training completion .

• Deploy the model train model as an endpoint for real-time prediction.

nazon SageMaker Al > Training jobs > inventory-forecasting-job-copy-11-03				
Clone Create model package Stop	Create model			
Job settings				
Job name	SageMaker metrics time series			
inventory-forecasting-job-copy-11-03	Disabled			
ARN	Training time (seconds)			
arn:aws:sagemaker:eu-west-1:703671932762:training- job/inventory-forecasting-job-copy-11-03	79			
	Billable time (seconds)			
Status O Completed	79			
View history	Managed spot training savings 0%			
Creation time Nov 03, 2024 17:01 UTC	Tuning job source/parent			
Last modified time Nov 03, 2024 17:03 UTC	IAM role ARN arn:aws:iam::703671932762:role/service-			
	role/AmazonSageMakerServiceCatalogProductsUseRole			

Figure12:Deploying the train model .

• we will use this endpoint to forecast inventory hits coming from the kinesis.

ventoryPredictionEn	dpoint	Delet
Endpoint summary		
Name	Status	Туре
nventoryPredictionEndpoint	⊘ InService	Real-time
ARN	Creation time	Last updated
arn:aws:sagemaker:eu-west-	Sun Dec 08 2024 10:43:34	Sun Dec 08 2024 10:46:35
1:703671932762:endpoint/nventory PredictionEndpoint	GMT+0000 (Greenwich Mean Time)	GMT+0000 (Greenwich Mean Time)
URL	Model container logs	Alarms
https://runtime.sagemaker.eu-west-	/aws/sagemaker/endpoints/nventor	0 alarms
1.amazonaws.com/endpoints/nvent oryPredictionEndpoint/invocations	yPredictionEndpoint	
Learn more about the API		

Figure13:Endpoint to demonstration for prediction.

After the machine model integration we have to set up visualizing Data so that user can understand the trend therefore we will step up Amazon QuickSight

- For the QuickSightint setup go to Amazon QuickSight and Sign Up.
- Then choose the standard edition for a single user license.

🛱 Datasets

• Connect to Data Sources and then add the DynamoDB and S3 as data sources.

Find analyses & more Q	Datasets			New	v dataset
Favorites	Name		Owner	Last Modified $$	
(Recent	results.csv	SPICE	Me	17 days ago	:
•	🏮 InventoryDataS3	SPICE	Me	22 days ago	:
My folders	🏮 People Overview	SPICE	Me	a month ago	:
☐ Shared folders	🏮 Business Review	SPICE	Me	a month ago	:
II. Dashboards	🏮 Sales Pipeline	SPICE	Me	a month ago	:
🛱 Data stories	🐞 Web and Social Media Analytics	SPICE	Me	a month ago	:
Analyses					

Figure14:QuickSight dataset demonstration setup.

• Now we will create a dashboards to visualize inventory trends and predictions.

🗾 QuickSight 🛱 InventoryDat	taS3 analysis		<u>s</u>
File Edit Data Insert Sheets	s Objects Search		TO WIDTH 🔻 PUBLISH
∽ ~ 8 0 7 # 0	🕼 ADD: 🖍 T 🕅 🖳 📼]	
🖯 Data X	00 Visuals ×	Sheet 1 ~ +	0
Dataset 100% SPICE InventoryDataS3	+ ADD	Sum of Stocklevel by Date and Itemid	
Search fields Q + CALCULATED FIELD	CHANGE VISUAL TYPE		• item123
💾 date	X AXIS		
itemId	date (MINUTE)		
# stockLevel	VALUE stockLevel (Sum)		0
	color (itemld :)		
	SMALL MULTIPLES		//

Figure15:QuickSight dashboard creation.

Once's the visual setup is done we move to setting up the Monitoring and Optimization in Amazon CloudWatch For that follow the below mention steps.

• Go to CloudWatch and click on create an alarm and select the custom Metrics .

	=
Add math 🔻 Add query 🔻	
Browse Multi source query Graphed metrics Options	Source
▼ Custom namespaces	
/aws/sagemaker/Endpoints 12	CustomMetrics 1
DynamoDBCustomMetrics 1	

Figure16:Cloud watch alarm custom metrics selection.

- Click on the create alarms button and give name and defined the alarm type.
- Turn on the notifications via Amazon SNS to alert .



Figure17:Alarm configuration all settings.

1. Simulation Performances Steps

1. For Simulate Inventory first need to connect the EC2 instances with local terminal to perform update on inventory, so go to the connect button and click on it.

Q Find Ins	tance by attribute or tag (ca	se-sensitive)		All state	s 🔻		
Instance sta	te = running X	Clear filters)			< 1	>
🗹 Nam	e 🖉 🛛 🔻 📔 Instance I	D	Instance state	▼ Instance type	▼ 9	Status check	
🔽 Inver	itoryPredi i-08ffadf4	bbfa78fe5	🕑 Running 🛛 🔍	🗨 t2.micro	(2/2 checks pa	ssec
08ffadf4bbfa	78fe5 (InventoryPred	ictionModel)	=				鐐

Figure18:EC2 instances connection.

2. After this it will open SSH client in which we have to copy the path of Example

Connect to instance Info

Connect to instance i-08ffadf4bbfa78fe5 (InventoryPredictionModel) using any of these options

EC2 Instance Connect	Session Manager	SSH client	EC2 serial console
Instance ID			
i-08ffadf4bbfa78fe5 (Inv	entoryPredictionModel)		
1. Open an SSH client.			
2. Locate your private ke	y file. The key used to laund	h this instance is i	nventorypredictionmodel.pem
	necessary, to ensure your ke torypredictionmodel.pem"	ey is not publicly v	iewable.
4. Connect to your instar	ice using its Public DNS: 175.eu-west-1.compute.am	azonaws.com	
Example:			
ssh -i "inventorypredicti	onmodel.pem" ec2-user@e	c2-18-201-228-17	5.eu-west-1.compute.amazonaws.com
(i) Note: In most cases, the username.	ne guessed username is cor	rect. However, rea	d your AMI usage instructions to check if the AMI owner has changed the default AMI

Figure19:SSH key for connection.

3. After this past this path into local terminal.

Last login: Mon Dec 9 14:36:05 on ttys000 [abhishek@Abhisheks-MacBook-Air ~ % cd Downloads]	
[abhishek@Abhisheks-Ma	acBook-Air Downloads % ec2-18-201-228-175.eu-west-1.compute.amazonaws.com
zsh: command not four	nd: ec2-18-201-228-175.eu-west-1.compute.amazonaws.com
abhishek@Abhisheks-MacBook-Air Downloads % ssh -i "inventorypredictionmodel.pem" ec2-user@ec2-18-201-228-175.	
[eu-west-1.compute.amazonaws.com]	
Last login: Mon Dec	9 14:36:19 2024 from 37.228.244.140
, #_	
~_ ####_	Amazon Linux 2
~~ `_#####\	
~~ \###	AL2 End of Life is 2025-06-30.
~~ \#/	
~~ V~''->	
~~~ /	A newer version of Amazon Linux is available!
~~/	
	Amazon Linux 2023, GA and supported until 2028–03–15.
_/m/'	https://aws.amazon.com/linux/amazon-linux-2023/
3 package(s) needed for security, out of 6 available Run "sudo yum update" to apply all updates. [ec2-user@ip-172-31-27-152 ~]\$	

Figure20:Terminal demonstration of Connection.

- 4. The connection with the terminal and DynamoDB has been made now run the below mention code to check two parameter of Overstock and Understock situation which is defined to forece fully update the data into loop and check the situation of the inventory
  - Overstock
     for i in {1..500}; do
     aws dynamodb put-item \
     --table-name InventoryTable \
     --item "{\"itemId\": {\"S\": \"SampleItemId\$i\"}, \"attribute1\": {\"S\":
     \"Value\$i\"}}"
     done

```
    Understock
        for i in {1..500}; do
            aws dynamodb delete-item \
                --table-name InventoryTable \
                --key "{\"itemId\": {\"S\": \"SampleItemId$i\"}}"
            done
```

5. After running above mention code user need to go on cloud watch and see the alarms which has been set for give an update on each threshold parameter and demonstrate the uses of CPU, lambda utilization of handling the situation and user get responses on email of each trigger and max and min limits of utilization.