

CAPSTONE PROJECT REPORT

TITLE:

Microservices Orchestration and Management with Kubernetes

<https://github.com/mano257200/capstone-final.git>

BY:

Emmanuel Louis (DEVOPS)

Mahendravarman R (CyberSecurity)

Microservices Orchestration and Management with Kubernetes

Problem Statement: "TechSolutions" is modernizing its application infrastructure by moving to a microservices architecture. Your challenge is to deploy and manage their set of microservices using Kubernetes, ensuring resilience, scalability, and efficient inter-service communication.

Dataset: A set of interconnected microservices applications available on GitHub, simulating an e-commerce platform with services like User Management, Product Catalog, Cart, and Payment.

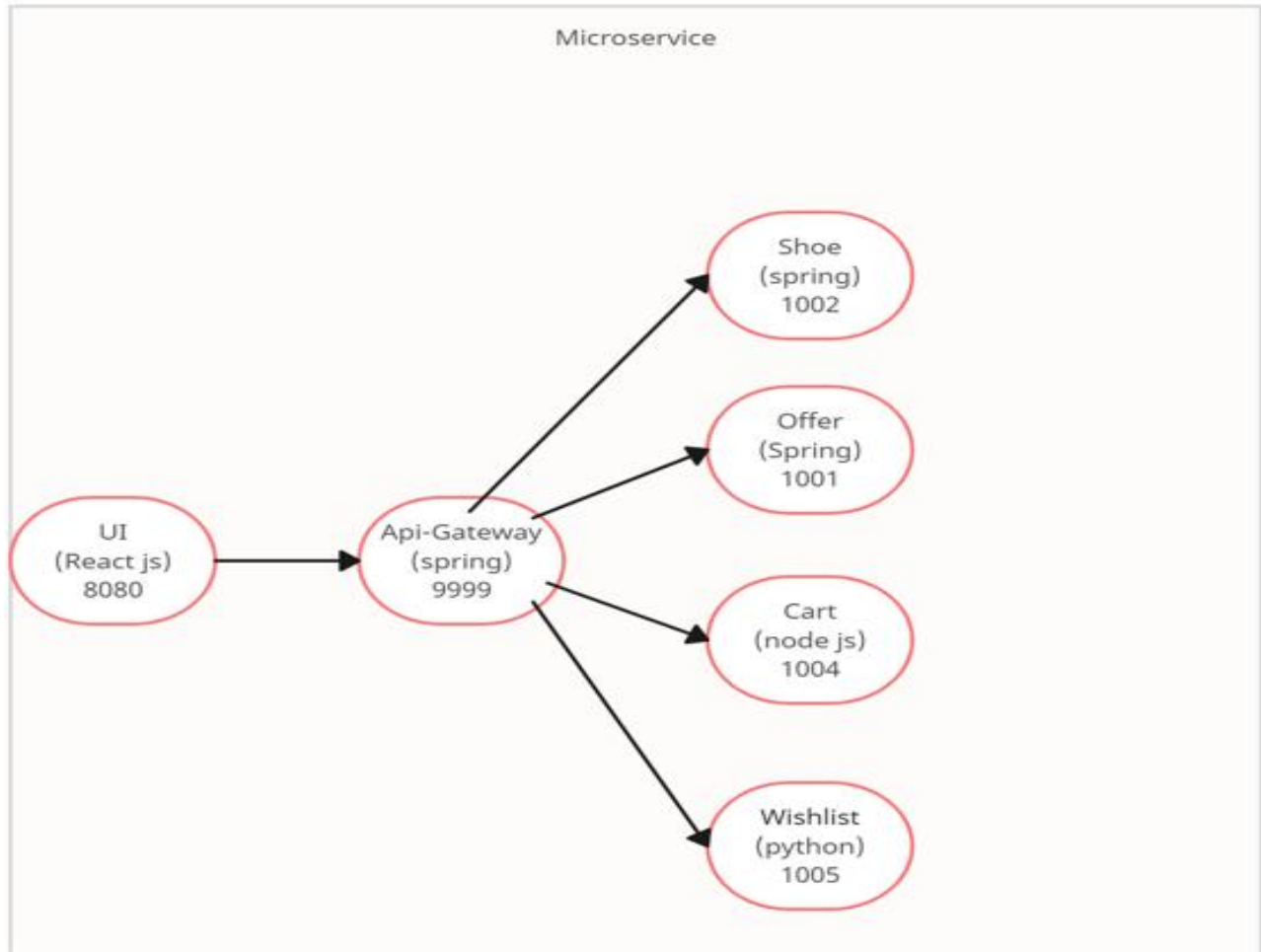
Step 1: Clone The Application From The Github Repo

- The Github repository contains six separate files as each file is a different microservice required for an e-commerce application.
- The microservices such as for e-commerce application user interface , api , offers , wishlist , cart , shoes .

Name	Last commit message	Last commit date
..		
cart-microservice-nodejs	Add files via upload	yesterday
offers-microservice-spring-boot	Add files via upload	yesterday
shoes-microservice-spring-boot	Add files via upload	yesterday
ui-web-app-reactjs	Add files via upload	yesterday
wishlist-microservice-python	Add files via upload	yesterday
zuul-api-gateway	Add files via upload	yesterday

Architecture of the microservice Application:

Architecture of sample multiple microservices developed in different technologies - Spring Boot, Node.js, Python, React.js in a project. Microservices connected by an API Gateway



Step 2: Containerizing The Microservices.

Containerization is a crucial step in deploying microservices using Kubernetes. It involves packaging each microservice into a container image.

Steps to containerize microservices :

- Create a Dockerfile for each microservice.
- In the Dockerfile, specify a base image that aligns with the microservice's requirements.
- Add instructions to copy the microservice's source code and any necessary dependencies into the container image.
- Define the command that should be executed when the container starts.

Containerizing e-commerce app user-interface

Dockerfile For Each Microservices:

UI (React js):

```
1 FROM node:8
2 WORKDIR /app
3 COPY . .
4 RUN npm install
5 RUN npm run build
6 EXPOSE 8080
7 CMD ["node", "server.js"]
```

Api (Spring):

```
1 FROM maven as build
2 WORKDIR /app
3 COPY . .
4 RUN mvn install
5
6 FROM openjdk:11.0.10-jre
7 WORKDIR /app
8 COPY --from=build /app/target/zuul-0.0.1-SNAPSHOT.jar /app
9 EXPOSE 8080
10 CMD ["java", "-jar", "zuul-0.0.1-SNAPSHOT.jar"]
```

Shoe (Spring):

```
1 FROM maven as build
2 WORKDIR /app
3 COPY . .
4 RUN mvn install
5
6 #jdk
7 FROM openjdk:11.0.10-jre
8 WORKDIR /app
9 COPY --from=build /app/target/shoes-0.0.1-SNAPSHOT.jar /app
10
11 EXPOSE 1002
12
13 CMD ["java", "-jar", "shoes-0.0.1-SNAPSHOT.jar"]
```

Offer (Spring):

```
1 FROM maven as build
2 WORKDIR /app
3 COPY . .
4 RUN mvn install
5
6 #jdk
7 FROM openjdk:11.0.10-jre
8 WORKDIR /app
9 COPY --from=build /app/target/offers-0.0.1-SNAPSHOT.jar /app
10
11 CMD ["java", "-jar", "offers-0.0.1-SNAPSHOT.jar"]
```

Cart (Node JS):

```
1 FROM node:14
2 WORKDIR /app
3 COPY . .
4 RUN npm install
5 CMD [ "node", "index.js" ]
```

Wishlist (Python):

```
1 FROM python:3
2 COPY . .
3 RUN pip install flask flask_cors
4 CMD ["python","index.py"]
```

Build and push image using Azure devops pipeline and push the image to the DockerHub and copy to Artifact











The screenshot shows the configuration page for a pipeline named 'ui1'. The left sidebar contains tabs for 'Tasks', 'Variables', 'Triggers', 'Options', 'History', 'Save & queue', 'Discard', 'Summary', 'Queue', and '...'. The main area is titled 'Pipeline' and 'Build pipeline'. It shows a list of tasks: 'Get sources' (Run on agent), 'Agent job 1' (Run on agent), 'Build an image' (Run on agent), 'Push an image' (Run on agent), 'Copy Files to: \$(Build.ArtifactStagingDirectory)' (Copy files), and 'Publish Artifact: drop' (Publish build artifacts). The right sidebar shows the 'Name' field set to 'ui1', the 'Agent pool' set to 'Azure Pipelines', and the 'Agent Specification' set to 'ubuntu-20.04'. The 'Parameters' section indicates that the pipeline doesn't have any parameters.

- Create a pipeline in Azure devops
- Add Agent Job to the pipeline like build an image , push an image , Copy files , Publish Artifact for continuous Integration













The screenshot shows the 'Jobs in run' page for pipeline #20230914.1. The left sidebar shows a list of jobs: 'Agent job 1' (10m 36s), 'Initialize job' (<1s), 'Checkout emmanuel55...' (8s), 'Build an image' (8m 18s), 'Push an image' (2m 4s), 'Copy Files to: C:\azure...' (<1s), 'Publish Artifact: drop' (1s), 'Post-job: Checkout e...' (<1s), and 'Finalize Job' (<1s). The 'Build an image' job is selected, and the right sidebar shows its execution log. The log includes the following commands and their output:

```
#7 sha256:a22eb1d90ebad4d8633c84928bb2a1edd3c1132a8ad0baee39c793a90da71812
#7 sha256:2cabec57fa3613116cbc641e1159ee0a7bde39ef9b3046a4e2122a5b390b7db5 0B / 17.46MB 0.2s
#7 sha256:9bb4e550f43682a246b903fdef6927f76ed9470a81b08a6867ee0f2b6047b927 0B / 153.80MB 0.4s
#7 ...
#13 [stage-1 1/3] FROM docker.io/library/openjdk:11.0.10-jre
#13 sha256:5de20d1464ef8cd105ca8f3cabf83f576a7793657e2d9f9d5f7843fbf02667ee
#13 DONE 1.1s
#7 [build 1/4] FROM docker.io/library/maven@sha256:64de69ce79d9af45a39f5167e8fc6256af59184f62f0ab46e15187585ef37614
#7 sha256:a22eb1d90ebad4d8633c84928bb2a1edd3c1132a8ad0baee39c793a90da71812
#7 ...
#6 [stage-1 2/3] WORKDIR /app
#6 sha256:c2e95fd9772985276c8e0f4c9b0b058cb56cace6c444e9d309f556a999271a36
#6 DONE 0.8s
#7 [build 1/4] FROM docker.io/library/maven@sha256:64de69ce79d9af45a39f5167e8fc6256af59184f62f0ab46e15187585ef37614
#7 sha256:a22eb1d90ebad4d8633c84928bb2a1edd3c1132a8ad0baee39c793a90da71812
#7 sha256:44ba2882f8eb14264e5f2f9f6ec55bcf5306527b637279f2cd9d4858762388af 2.10MB / 30.44MB 2.0s
#7 sha256:2cabec57fa3613116cbc641e1159ee0a7bde39ef9b3046a4e2122a5b390b7db5 1.05MB / 17.46MB 2.5s
#7 sha256:44ba2882f8eb14264e5f2f9f6ec55bcf5306527b637279f2cd9d4858762388af 4.19MB / 30.44MB 3.4s
#7 sha256:2cabec57fa3613116cbc641e1159ee0a7bde39ef9b3046a4e2122a5b390b7db5 2.10MB / 17.46MB 3.6s
#7 sha256:2cabec57fa3613116cbc641e1159ee0a7bde39ef9b3046a4e2122a5b390b7db5 3.15MB / 17.46MB 4.6s
#7 sha256:44ba2882f8eb14264e5f2f9f6ec55bcf5306527b637279f2cd9d4858762388af 6.29MB / 30.44MB 5.4s
#7 sha256:9bb4e550f43682a246b903fdef6927f76ed9470a81b08a6867ee0f2b6047b927 3.15MB / 153.80MB 5.4s
#7 sha256:2cabec57fa3613116cbc641e1159ee0a7bde39ef9b3046a4e2122a5b390b7db5 4.19MB / 17.46MB 5.5s
```

Pushed images into DockerHub:

Tags				
This repository contains 6 tag(s).				
Tag	OS	Type	Pulled	Pushed
 wishlist		Image	an hour ago	an hour ago
 cart		Image	an hour ago	an hour ago
 offer		Image	an hour ago	an hour ago
 ui1		Image	an hour ago	an hour ago
 shoe		Image	---	2 hours ago
See all		Go to Advanced Image Management		

CI Pipelines:

Pipelines			New pipeline
Recent All Runs			Filter pipelines
Recently run pipelines			
Pipeline	Last run		
 wishlist	#20230914.1 • Update and rename kube.yaml to kube-wash.yaml ↳ Manually triggered for  main		14m ago 55s
 cart	#20230914.1 • Update and rename kubecart.yaml to kube-cart.yaml ↳ Manually triggered for  main		22m ago 56s
 offer	#20230914.1 • Rename kubeoffer.yaml to kubeoffers.yaml ↳ Manually triggered for  main		31m ago 1m 20s
 ui1	#20230914.1 • Update kube.ui.yaml ↳ Manually triggered for  main		43m ago 2m 8s
 shoe	#20230914.2 • Update kubeshoe.yaml ↳ Manually triggered for  main		2h ago 1m 24s
 api	#20230914.2 • Update kubeshoe.yaml ↳ Manually triggered for  main		2h ago 1m 35s

Step 3: Deploy Microservices Application in Kubernetes Using Release pipeline in Azure Devops

Create Kubernetes cluster in Azure:

- Create Azure kubernetes service's and Enable public Ip in the nodes

```
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

Welcome to Azure Cloud Shell

Type "az" to use Azure CLI
Type "help" to learn about Cloud Shell

mahendravarman [ ~ ]$ az account set --subscription fc115d67-76b0-4888-8c8c-016128f5251c
mahendravarman [ ~ ]$ az aks get-credentials --resource-group E_rg --name project
Storage fileshare subscription fc115d67-76b0-4888-8c8c-016128f5251c is not registered to Microsoft.CloudShell Namespace. Please follow these instructions "https://aka.ms/gisterCloudShell" to register. In future, unregistered subscriptions will have restricted access to CloudShell service.

mahendravarman [ ~ ]$ az account set --subscription fc115d67-76b0-4888-8c8c-016128f5251c
mahendravarman [ ~ ]$ az aks get-credentials --resource-group E_rg --name project
Merged "project" as current context in /home/mahendravarman/.kube/config
mahendravarman [ ~ ]$ kubectl get pods
No resources found in default namespace.
mahendravarman [ ~ ]$ kubectl get nodes -o wide
NAME                                STATUS    ROLES    AGE   VERSION   INTERNAL-IP   EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION      CONTAINER-RUNTIME
aks-agentpool-39719263-vmss000000   Ready    agent    5m5s  v1.26.6   10.224.0.4    20.121.116.150 Ubuntu 22.04.3 LTS   5.15.0-1041-azure   containerd://1.7.1+azure-1
mahendravarman [ ~ ]$
```

Kubernetes manifest files for each microservices:

UI:

```
15     ---
16     apiVersion: apps/v1
17     kind: Deployment
18     metadata:
19       name: "react-ui"
20       labels:
21         app: react-ui
22     spec:
23       replicas: 1
24       selector:
25         matchLabels:
26           app: "react-ui"
27     template:
28       metadata:
29         labels:
30           app: react-ui
31       spec:
32         containers:
33           - name: "react-ui"
34             image: "emman777/e-commerce:ui"
35             imagePullPolicy: "IfNotPresent"
36             ports:
37               - containerPort: 8080
```



```
1  apiVersion: "v1"
2  kind: "Service"
3  metadata:
4    labels:
5      app: "react-ui"
6      name: "ui"
7  spec:
8    ports:
9      - name: "http"
10      port: 8080
11      targetPort: 8080
12    selector:
13      app: "react-ui"
14    type: "NodePort"
```

```
50  ---
51  apiVersion: autoscaling/v2
52  kind: HorizontalPodAutoscaler
53  metadata:
54    name: react-ui-autoscaler
55  spec:
56    scaleTargetRef:
57      apiVersion: apps/v1
58      kind: Deployment
59      name: react-ui
60    minReplicas: 1
61    maxReplicas: 5
62    metrics:
63      - type: Resource
64        resource:
65          name: cpu
66          target:
67            type: Utilization
68            averageUtilization: 50
69      - type: Resource
70        resource:
71          name: memory
72          target:
73            type: AverageValue
74            averageValue: 100Mi
```

API:

```
1  apiVersion: "v1"
2  kind: "Service"
3  metadata:
4    labels:
5      app: "zuul-api-gateway"
6      name: "zuul-api-gateway"
7  spec:
8    ports:
9      - name: "http"
10      port: 9999
11      targetPort: 9999
12      nodePort: 32470
13      type: "NodePort"
14    selector:
15      app: "zuul-api-gateway"
```

```
16  ---
17  apiVersion: apps/v1
18  kind: Deployment
19  metadata:
20    name: "zuul-api-gateway"
21    labels:
22      app: zulu
23  spec:
24    replicas: 1
25    selector:
26      matchLabels:
27        app: "zuul-api-gateway"
28    template:
29      metadata:
30        labels:
31          app: "zuul-api-gateway"
32      spec:
33        containers:
34          - name: "zuul-api-gateway"
35            image: "emman777/e-commerce:api"
36            imagePullPolicy: "IfNotPresent"
37            ports:
38              - containerPort: 9999
```

Cart:

```
1  ---
2  apiVersion: v1
3  kind: Service
4  metadata:
5    labels:
6      app: cart-ui
7    name: cart
8  spec:
9    ports:
10     - name: http
11       port: 1004
12       targetPort: 1004
13    selector:
14      app: cart-ui
15
```

```
15  ---
16  apiVersion: apps/v1
17  kind: Deployment
18  metadata:
19    name: cart-ui
20    labels:
21      app: cart-ui
22  spec:
23    replicas: 1
24    selector:
25      matchLabels:
26        app: cart-ui
27    template:
28      metadata:
29        labels:
30          app: cart-ui
31      spec:
32        containers:
33          - name: cart-ui
34            image: mahendravarman12/e-commerce:cart
35            imagePullPolicy: IfNotPresent
36            ports:
37              - containerPort: 1004
38            readinessProbe:
39              httpGet:
40                path: /
41                port: 1004
```

Offer:

```
1  ---
2  apiVersion: v1
3  kind: Service
4  metadata:
5    labels:
6      app: offer-ui
7    name: offer
8  spec:
9    ports:
10     - name: http
11       port: 1001
12       targetPort: 1001
13    selector:
14      app: offer-ui
```

```
15  ---
16  apiVersion: apps/v1
17  kind: Deployment
18  metadata:
19    name: offer-ui
20    labels:
21      app: offer-ui
22  spec:
23    replicas: 1
24    selector:
25      matchLabels:
26        app: offer-ui
27    template:
28      metadata:
29        labels:
30          app: offer-ui
31      spec:
32        containers:
33          - name: offer-ui
34            image: mahendravarman12/e-commerce:offer
35            imagePullPolicy: IfNotPresent
36            ports:
37              - containerPort: 1001
```

Shoe:

```
1  apiVersion: "v1"
2  kind: "Service"
3  metadata:
4    labels:
5      app: "shoe-ui"
6    name: "shoe"
7  spec:
8    ports:
9      - name: "http"
10      port: 1002
11      targetPort: 1002
12    selector:
13      app: "shoe-ui"
```

```
14  ---
15  apiVersion: apps/v1
16  kind: Deployment
17  metadata:
18    name: "shoe-ui"
19    labels:
20      app: shoe-ui
21  spec:
22    replicas: 1
23    selector:
24      matchLabels:
25        app: "shoe-ui"
26    template:
27      metadata:
28        labels:
29          app: shoe-ui
30      spec:
31        containers:
32          - name: "shoe-ui"
33            image: "mahendravarman12/e-commerce:shoe"
34            imagePullPolicy: "IfNotPresent"
35            ports:
36              - containerPort: 1002
```

Wish-list:

```
1  ---
2  apiVersion: v1
3  kind: Service
4  metadata:
5    labels:
6      app: wishlist-ui
7    name: wishlist
8  spec:
9    ports:
10     - name: http
11       port: 1003
12       targetPort: 5000
13     selector:
14       app: wishlist-ui
15
```

```
15  ---
16  apiVersion: apps/v1
17  kind: Deployment
18  metadata:
19    name: wishlist-ui
20    labels:
21      app: wishlist-ui
22  spec:
23    replicas: 1
24    selector:
25      matchLabels:
26        app: wishlist-ui
27    template:
28      metadata:
29        labels:
30          app: wishlist-ui
31      spec:
32        containers:
33          - name: shoe-ui
34            image: mahendravarman12/e-commerce:wish
35            imagePullPolicy: IfNotPresent
36            ports:
37              - containerPort: 5000
```

Continuous Deployment Using Release Pipeline:

14 / MiniProject / Pipelines / Releases

Search all pipelines

+ New

api release

Stage 1

New release pipeline
No deployments found

New release pipeline (2)
No deployments found

api release

Releases Deployments Analytics

Releases Created Stages

Release-1

20230914... 8⁹ main

9/14/2023, 12:18:49 PM

Stage 1

Edit Create release

All releases

Setup the artifact and connect to Azure kubernetes

api release > Release-1

Pipeline Variables History + Deploy Cancel Refresh Edit

Release

Manually triggered

by Mugundhan R

9/14/2023, 12:18 PM

Artifacts

_micro-api

20230914.1

main

Stages

Stage 1

Succeeded

on 9/14/2023, 12:19 PM

Set up kubernetes credentials and command

All pipelines > api release

Save Create release View releases ...

Pipeline Tasks Variables Retention Options History

Stage 1
Deployment process

Agent job
Run on agent

kubectl apply
Kubectl

Kubectl

Task version 0.*

Display name *
kubectl apply

Kubernetes service connection ⓘ | Manage

uiservice

Namespace ⓘ
default

Commands ^

Command ⓘ
apply

☒ Use Configuration files ⓘ

Configuration file * ⓘ
\$(System.DefaultWorkingDirectory)/_micro-api/drop/kubeapi.yaml

Create the Release

api release > Release-1 > Stage 1 ✓ Succeeded

← Pipeline Tasks Variables Logs Tests | Deploy Cancel Refresh Download all logs Edit ...

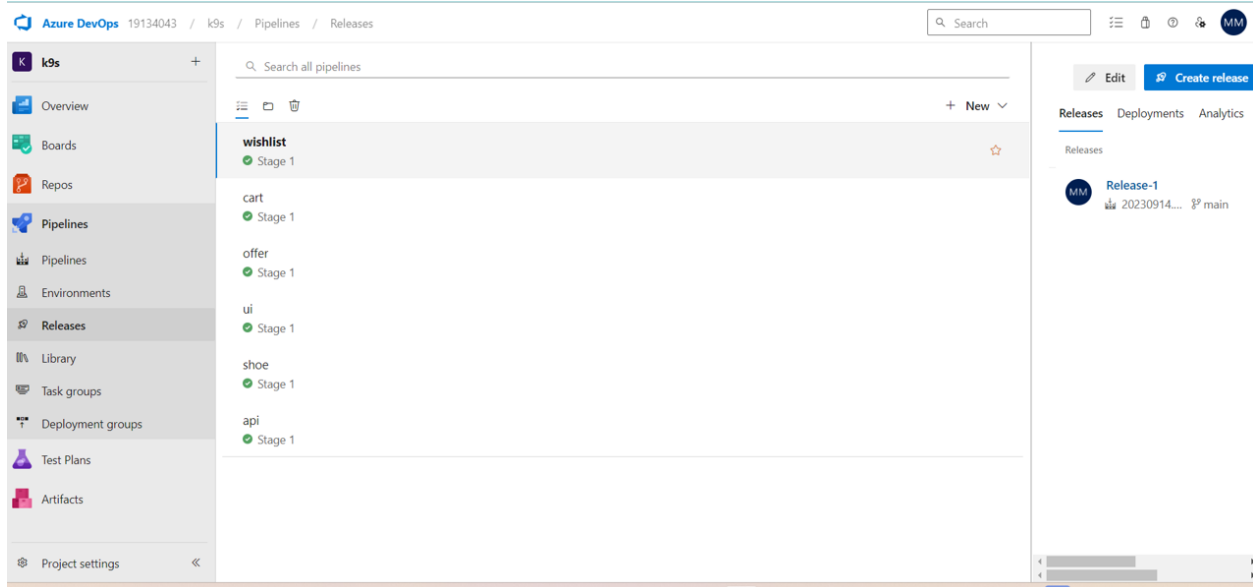
Deployment process
Succeeded

✓ Agent job
Succeeded

Agent job
Pool: MY_agent · Agent: myelagent
Started: 9/14/2023, 12:18:57 PM
... 9s

✓ Initialize job · succeeded	<1s
✓ Download artifact - _micro-api - drop · succeeded	3s
✓ kubectl apply · succeeded	4s
✓ Finalize Job · succeeded	<1s

All Release Pipeline for each microservices



Deployed pods in AKS

```
replicaset.apps/zuul-api-gateway-76c67ffccd 1 1 1 93m
mahendravarma [ ~ ]$ kubectl get all
```

NAME	READY	STATUS	RESTARTS	AGE
pod/cart-ui-75b6856b4f-xpp6s	1/1	Running	0	22m
pod/offer-ui-9776875fb-k5pw4	1/1	Running	0	28m
pod/shoe-ui-7659798b86-67z9k	1/1	Running	0	91m
pod/wishlist-ui-595f58c788-kplbc	1/1	Running	0	19m
pod/zuul-api-gateway-76c67ffccd-rt54	1/1	Running	0	94m

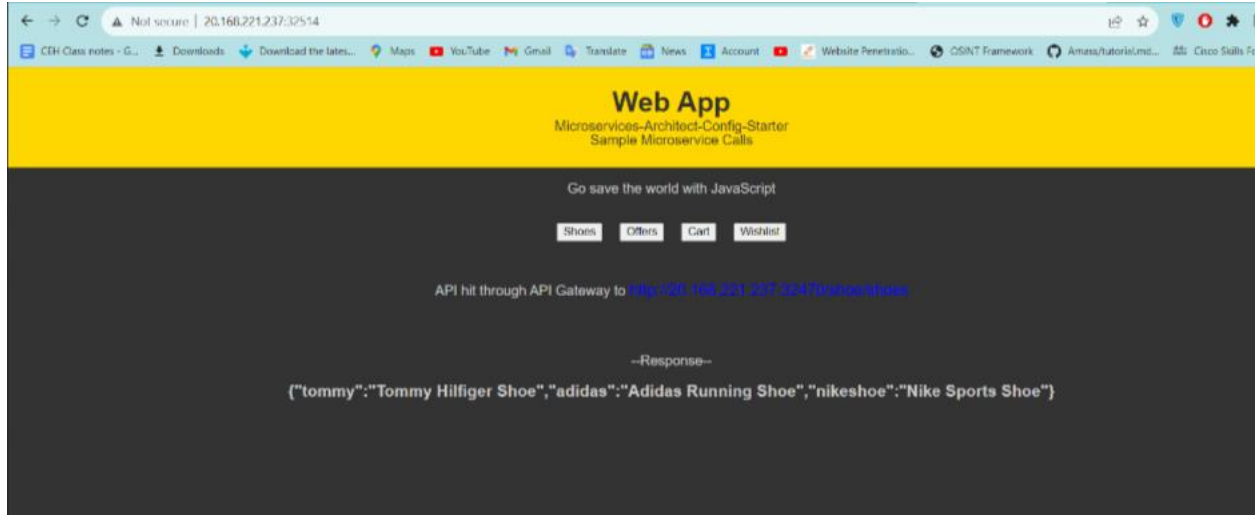
NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
service/cart	ClusterIP	10.0.250.87	<none>	1004/TCP	22m
service/kubernetes	ClusterIP	10.0.0.1	<none>	443/TCP	3h15m
service/offer	ClusterIP	10.0.55.192	<none>	1001/TCP	28m
service/shoe	ClusterIP	10.0.19.18	<none>	1002/TCP	91m
service/ui	NodePort	10.0.114.88	<none>	8080:32514/TCP	56m
service/wishlist	ClusterIP	10.0.67.106	<none>	1003/TCP	19m
service/zuul-api-gateway	NodePort	10.0.225.35	<none>	9999:32470/TCP	94m

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
deployment.apps/cart-ui	1/1	1	1	22m
deployment.apps/offer-ui	1/1	1	1	28m
deployment.apps/shoe-ui	1/1	1	1	91m
deployment.apps/wishlist-ui	1/1	1	1	19m
deployment.apps/zuul-api-gateway	1/1	1	1	94m

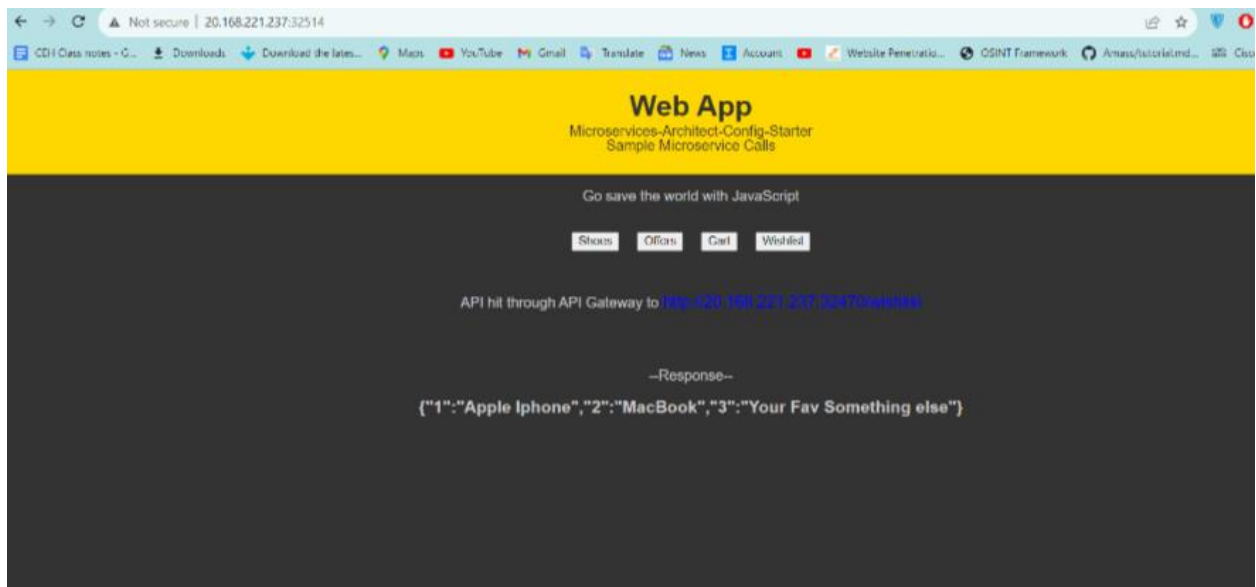
NAME	DESIRED	CURRENT	READY	AGE
replicaset.apps/cart-ui-75b6856b4f	1	1	1	22m
replicaset.apps/offer-ui-9776875fb	1	1	1	28m
replicaset.apps/shoe-ui-7659798b86	1	1	1	91m
replicaset.apps/wishlist-ui-595f58c788	1	1	1	19m
replicaset.apps/zuul-api-gateway-76c67ffccd	1	1	1	94m

Final Outputs:

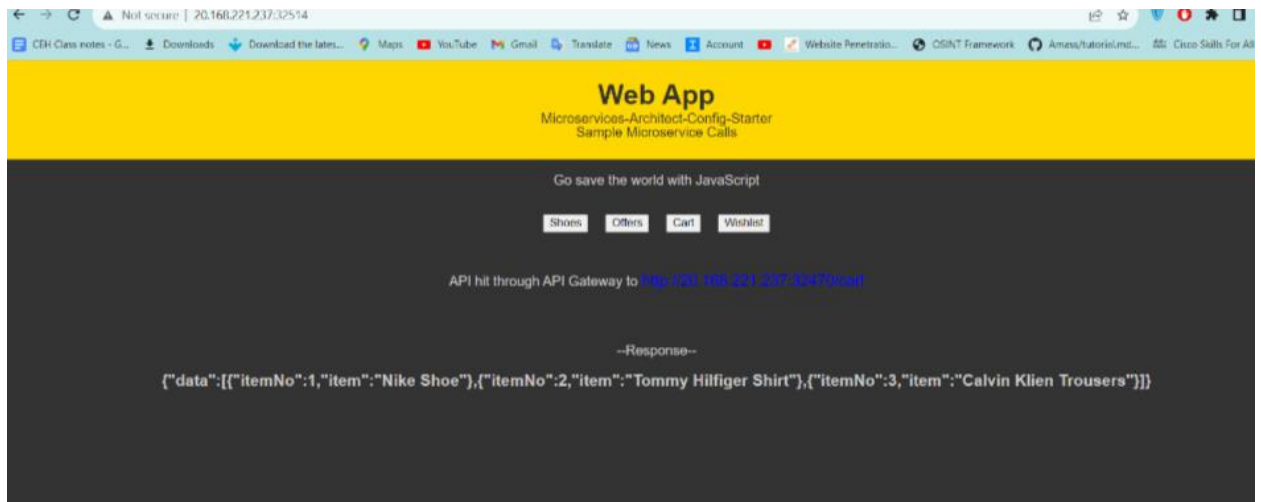
Shoe:



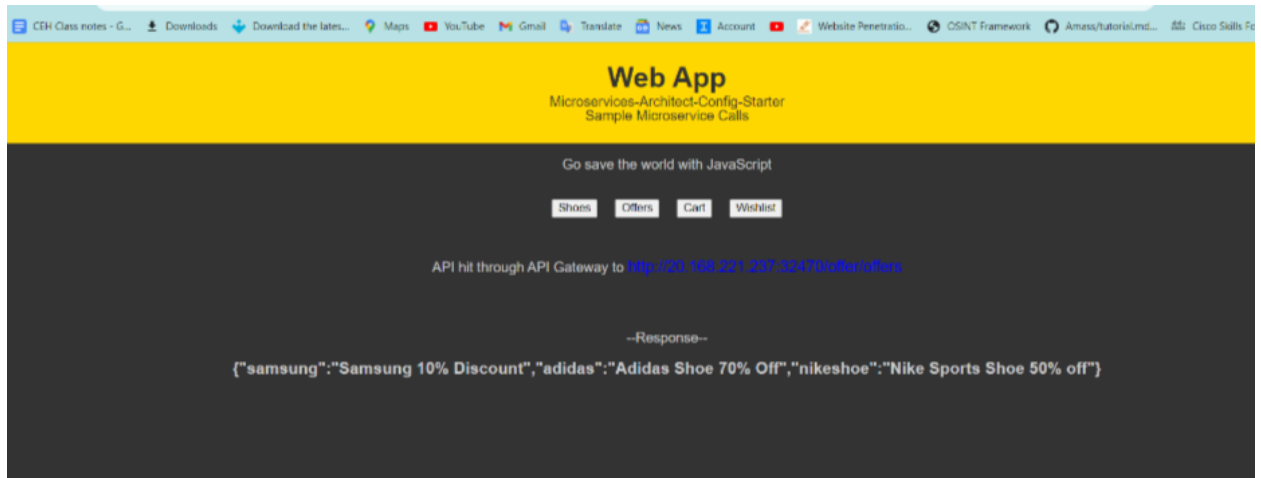
Wish-List:



Cart:



Offer:



Step 4: Monitor Microservices using EFK

- EFK is an acronym that stands for Elasticsearch, Fluentd, and Kibana. It is a popular open-source stack used for log and event data analysis and visualization.
- Monitor each pods logs in efk

capstone-final / EFK-Log monitoring /

Add file

...

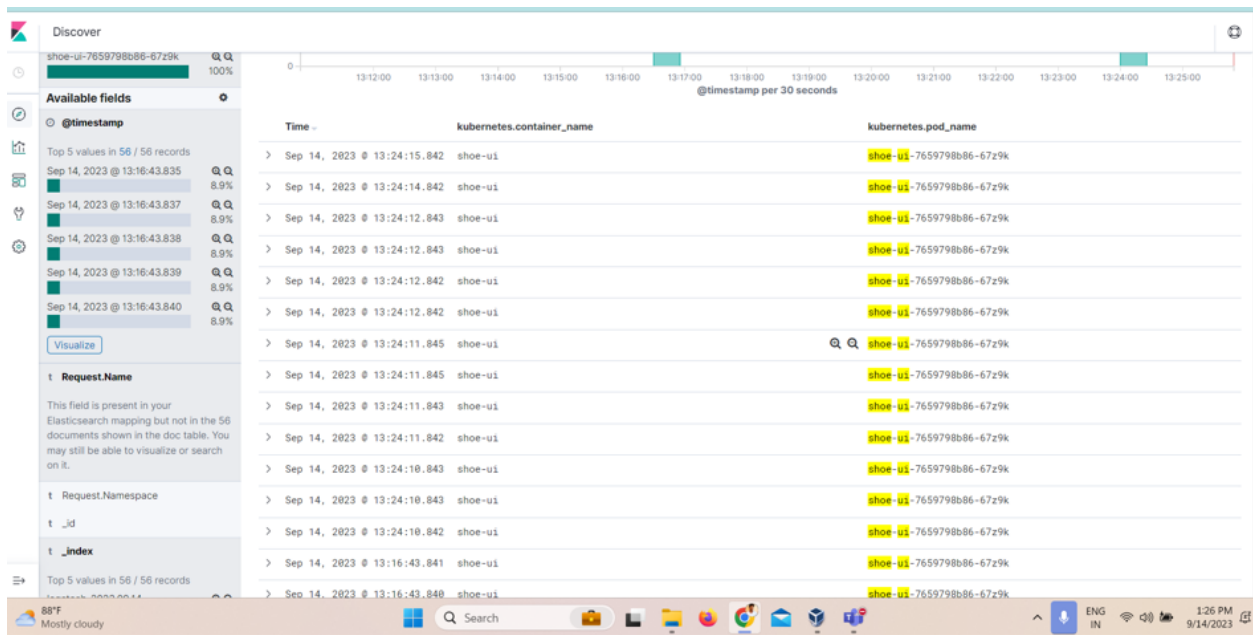
mano257200 Add files via upload

5fc3712 · 6 hours ago

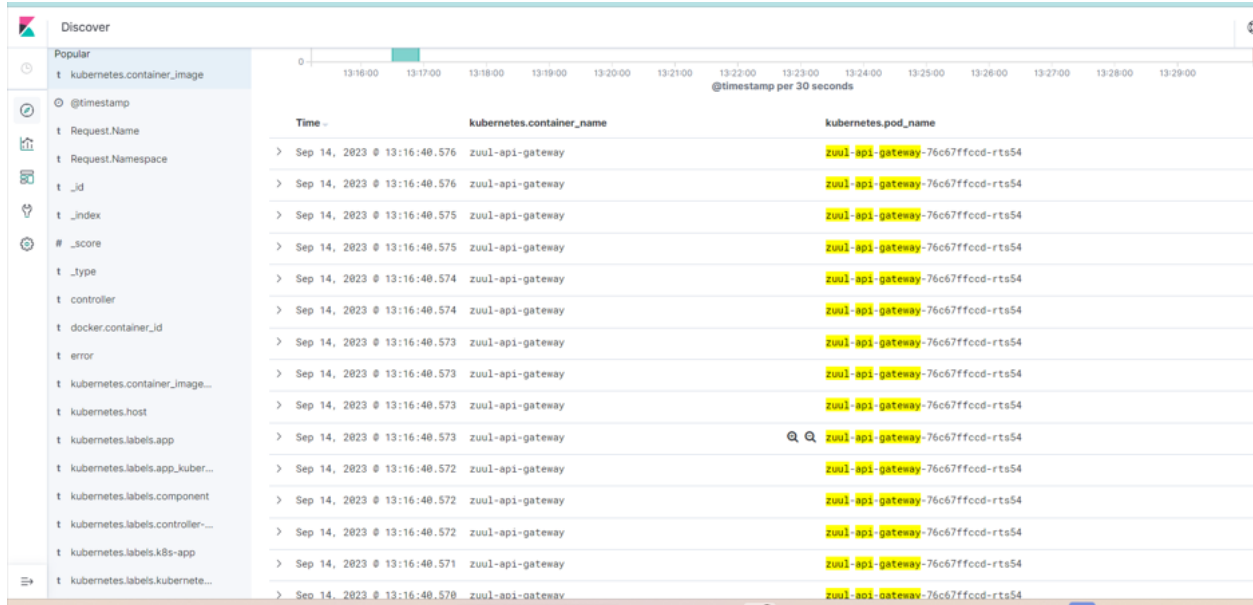
History

Name	Last commit message	Last commit date
..		
01_Namespace.yml	Add files via upload	6 hours ago
02_ElasticSearch_Service.yml	Add files via upload	6 hours ago
03_ElasticSearch_StatefulSet.yml	Add files via upload	6 hours ago
04_Fluentd_ConfigMap.yml	Add files via upload	6 hours ago
05_Fluentd_DaemonSet.yml	Add files via upload	6 hours ago
06_Kibana_Deployment.yml	Add files via upload	6 hours ago
07_Kibana_Service.yml	Add files via upload	6 hours ago

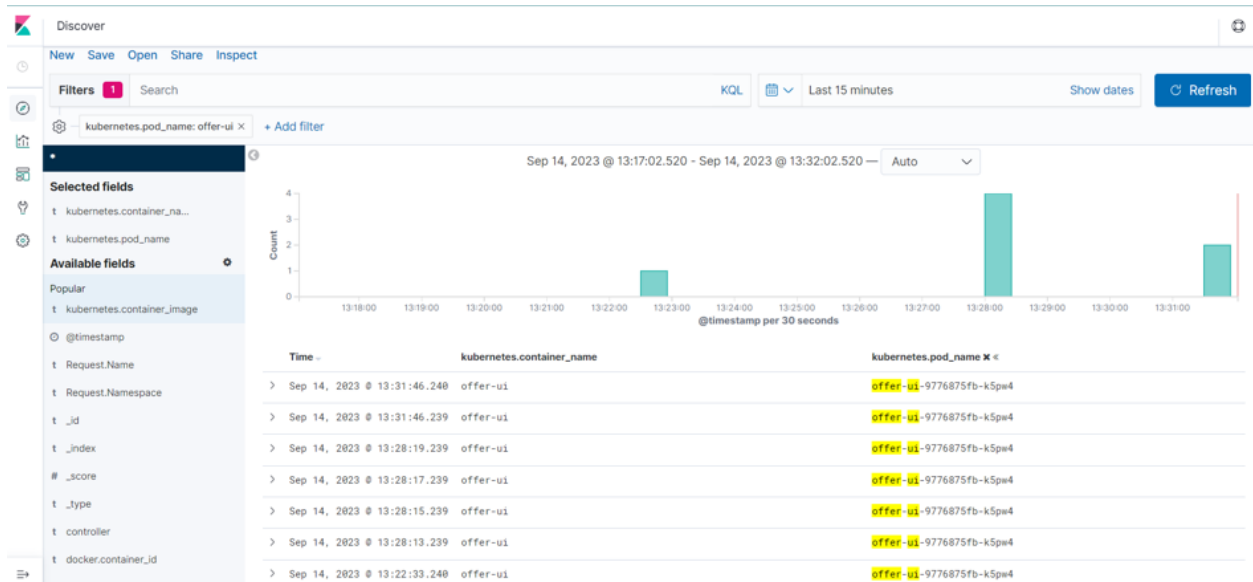
Pod shoe-ui:



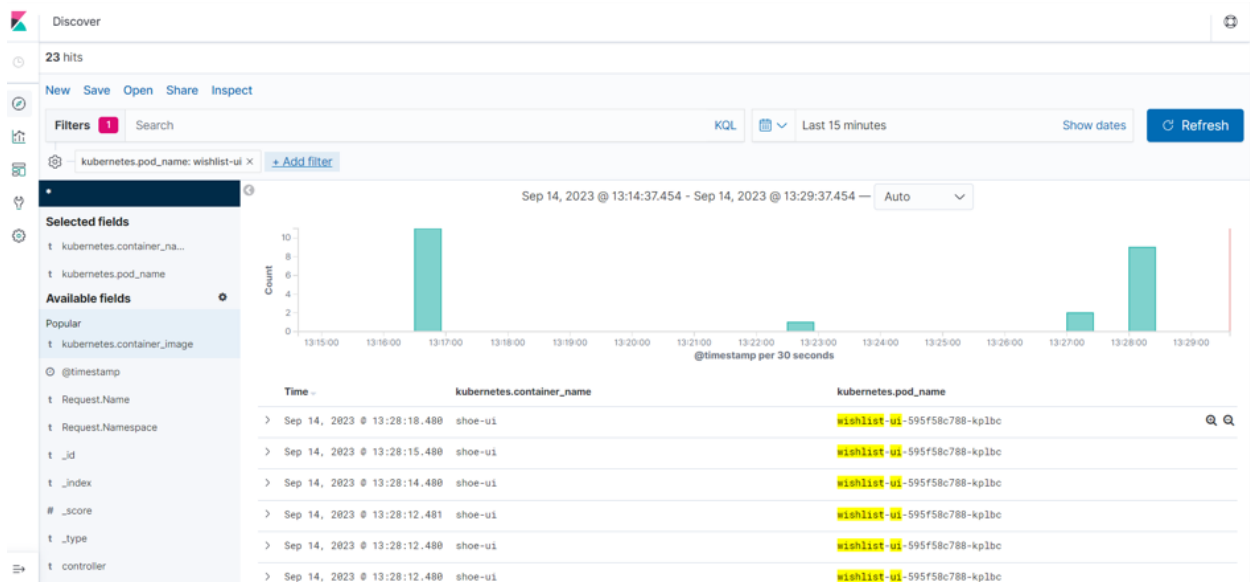
Pod zuul-api:



Pod Offer-ui:



Pod Wishlist-UI



Challenges Faced:

- Difficult to find the correct project git repo in github
- We get our code in github and In UI part the we Face code level problems like ip won't fetch automatically so we give statically ip using variables And In the UI part the code is written by very old version node 8 but latest version is node 18.14
- Setting up a reliable CI/CD pipeline for Kubernetes microservice
- Setting up and managing monitoring and logging solutions in Kubernetes

