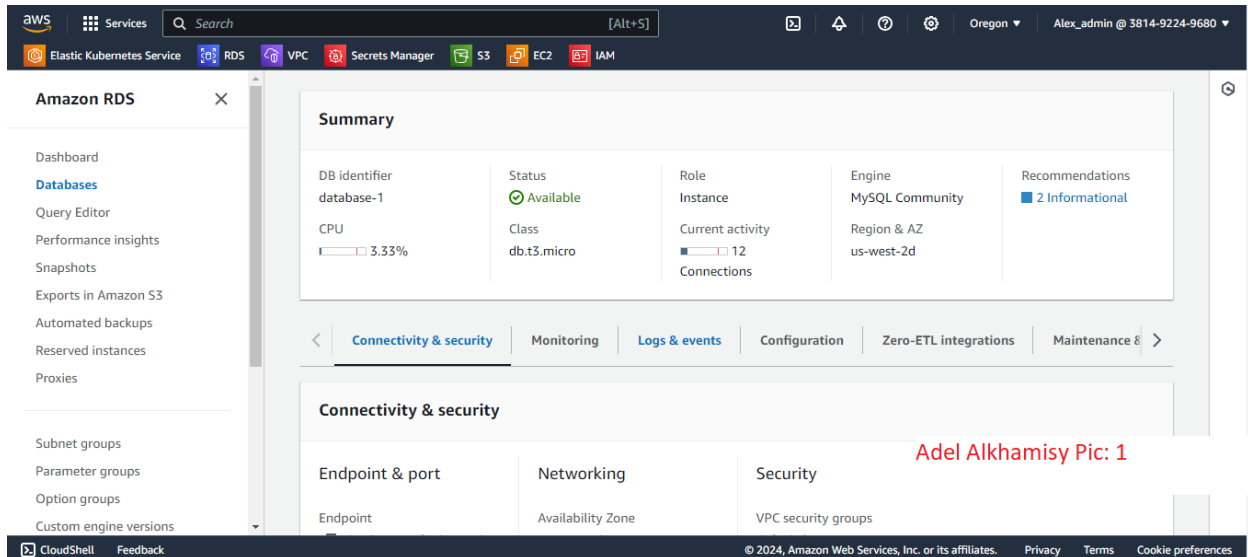
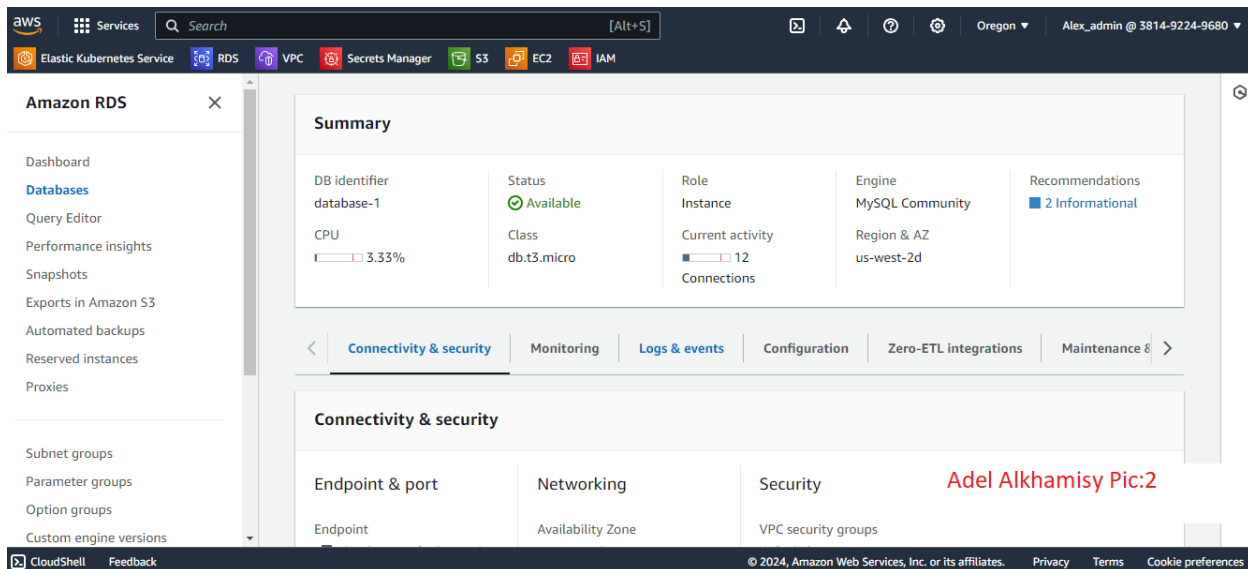


Jenkinsfile for CI/CD Pipeline with EKS Deployment for Spring Boot App

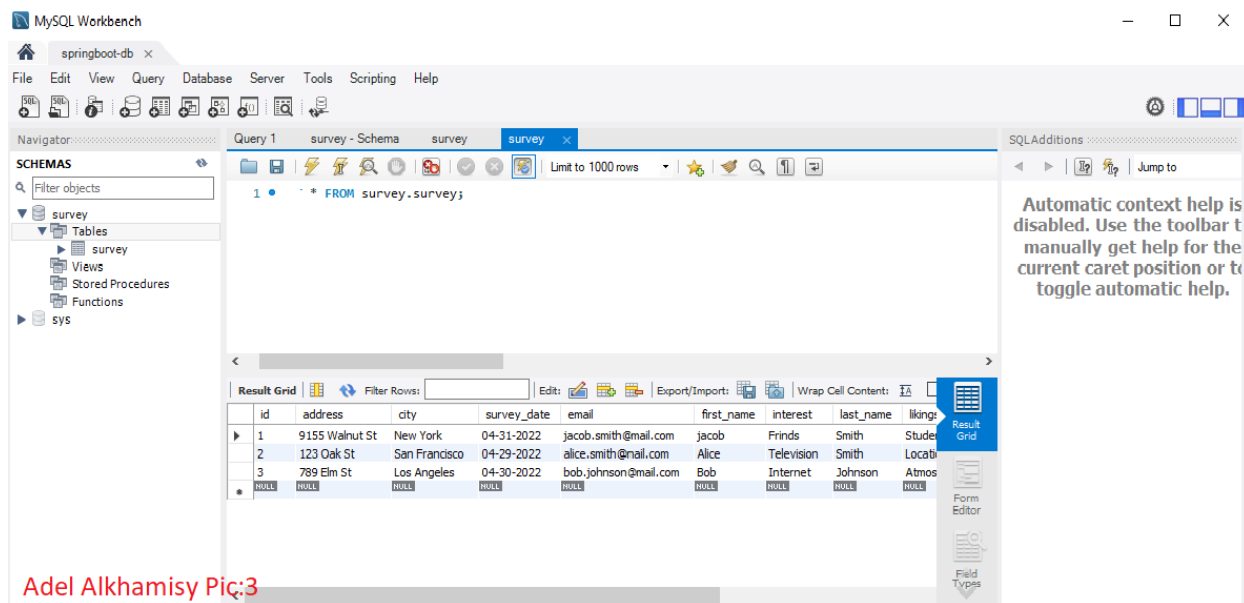
1: Create MySQL RDS in US West (Oregon)us-west-2 region



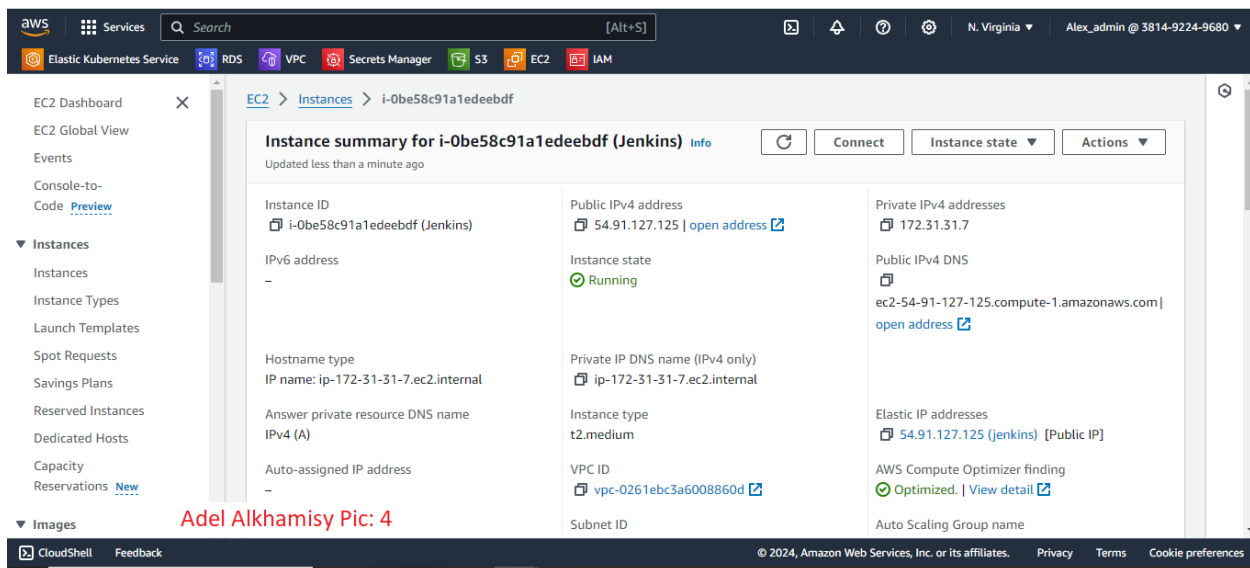
2: MySQL RDS configuration



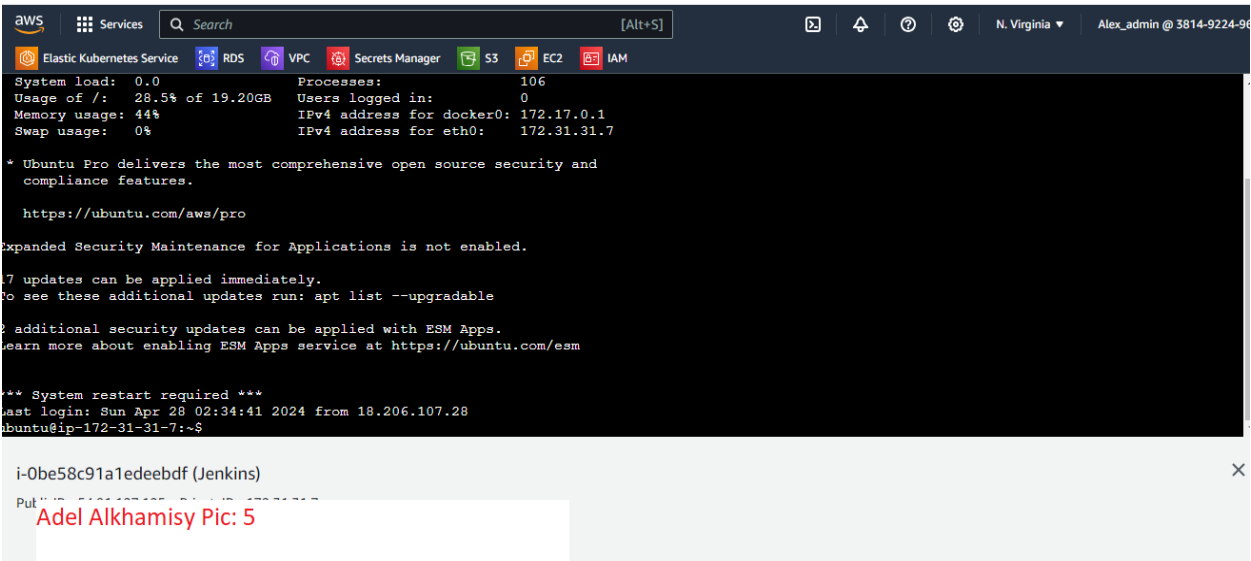
3: MySQL Workbench



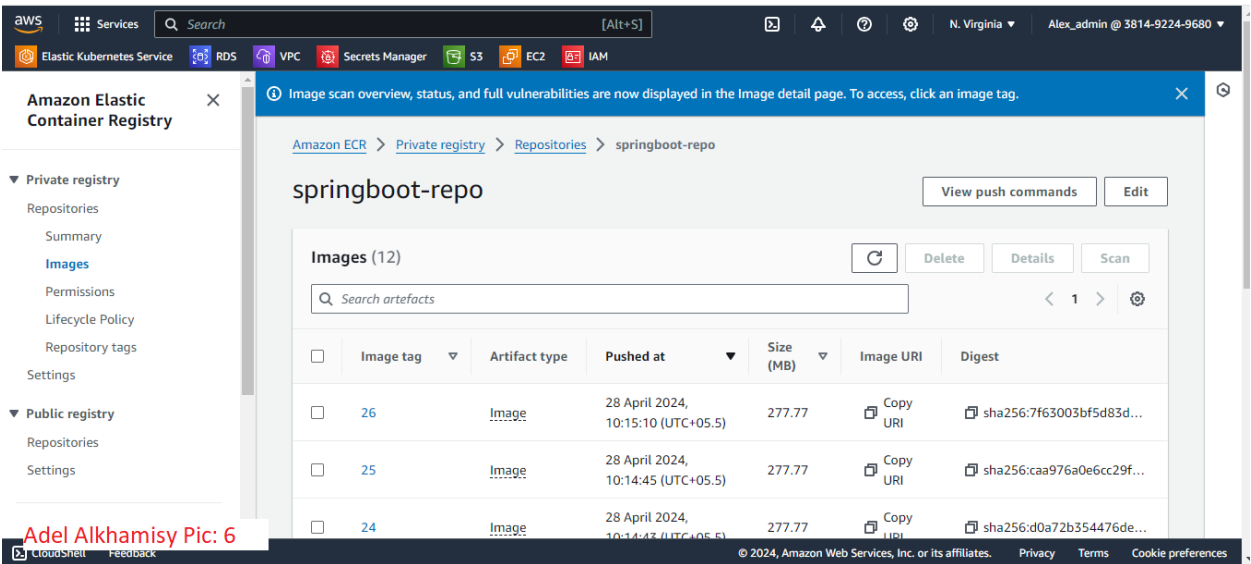
4: Create ec2 instance to install Jenkins, docker and maven. In us-east-1 with configuration: 2 vCPU, 4 GB RAM, 20 GB EBS volume



5: Ec2 console connect

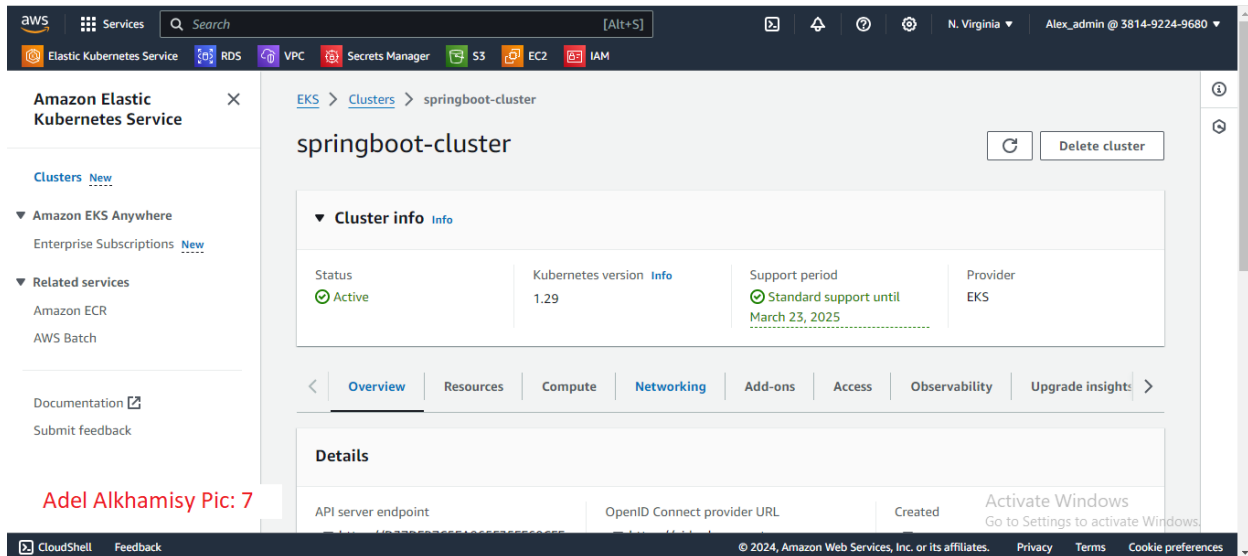


6. ECR to manage docker images



7. Create EKS to deploy springboot app

Configuration: Two node, Instance type for both: t3.small, region: us-east-1



This screenshot shows the AWS Management Console for the 'springboot-cluster' EKS cluster. The left sidebar displays the 'Amazon Elastic Kubernetes Service' navigation menu. The main content area shows the 'Cluster info' section with the following details:

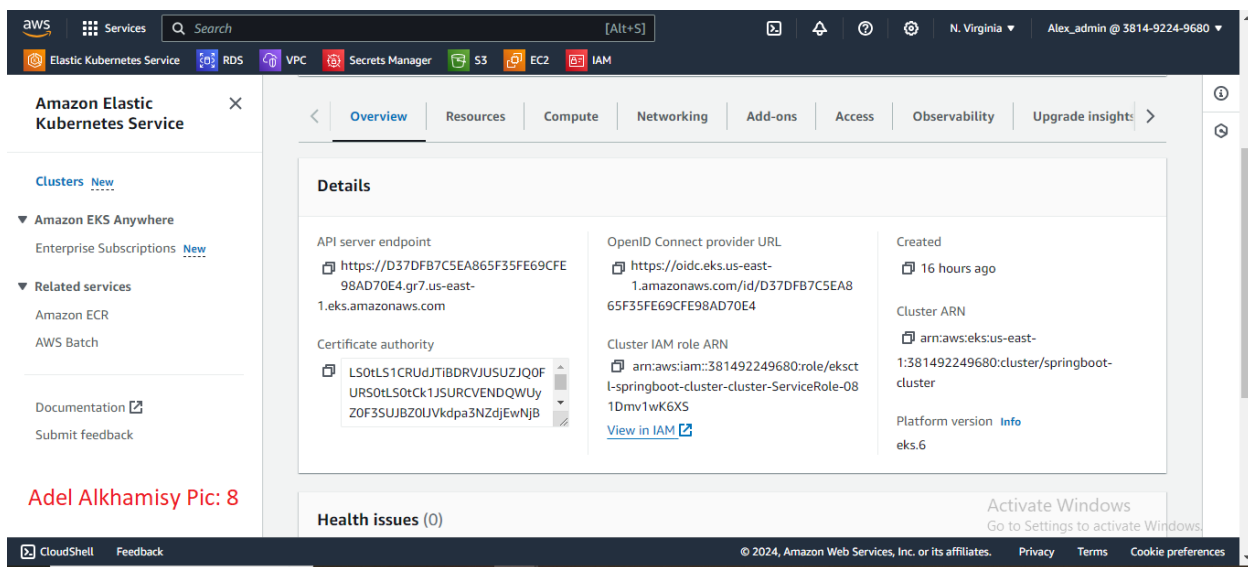
Cluster info	Info
Status	Active
Kubernetes version	1.29
Support period	Standard support until March 23, 2025
Provider	EKS

Below the 'Cluster info' section, there are tabs for 'Overview', 'Resources', 'Compute', 'Networking', 'Add-ons', 'Access', 'Observability', and 'Upgrade insights'. The 'Details' section is currently selected, showing the following information:

Details		
API server endpoint	OpenID Connect provider URL	Created
https://d37dfb7c5ea865f35fe69cfe98ad70e4.gr7.us-east-1.eks.amazonaws.com	https://oidc.eks.us-east-1.amazonaws.com/id/D37DFB7C5EA865F35FE69CFE98AD70E4	16 hours ago

The bottom of the console shows the 'CloudShell' and 'Feedback' buttons, along with the copyright notice for Amazon Web Services, Inc. or its affiliates.

Adel Alkhamisy Pic: 7



This screenshot shows the AWS Management Console for the 'springboot-cluster' EKS cluster, specifically the 'Details' tab. The left sidebar displays the 'Amazon Elastic Kubernetes Service' navigation menu. The main content area shows the 'Details' section with the following information:

Details		
API server endpoint	OpenID Connect provider URL	Created
https://d37dfb7c5ea865f35fe69cfe98ad70e4.gr7.us-east-1.eks.amazonaws.com	https://oidc.eks.us-east-1.amazonaws.com/id/D37DFB7C5EA865F35FE69CFE98AD70E4	16 hours ago
Certificate authority	Cluster IAM role ARN	Cluster ARN
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURCVENDQWUyZ0F3SUVBZ0lVkdpa3NZdjEwNjB	arn:aws:iam::381492249680:role/eksctl-springboot-cluster-cluster-ServiceRole-081Dmv1wK6XS	arn:aws:eks:us-east-1:381492249680:cluster/springboot-cluster
	View in IAM	Platform version
		eks.6

Below the 'Details' section, there is a 'Health issues (0)' section. The bottom of the console shows the 'CloudShell' and 'Feedback' buttons, along with the copyright notice for Amazon Web Services, Inc. or its affiliates.

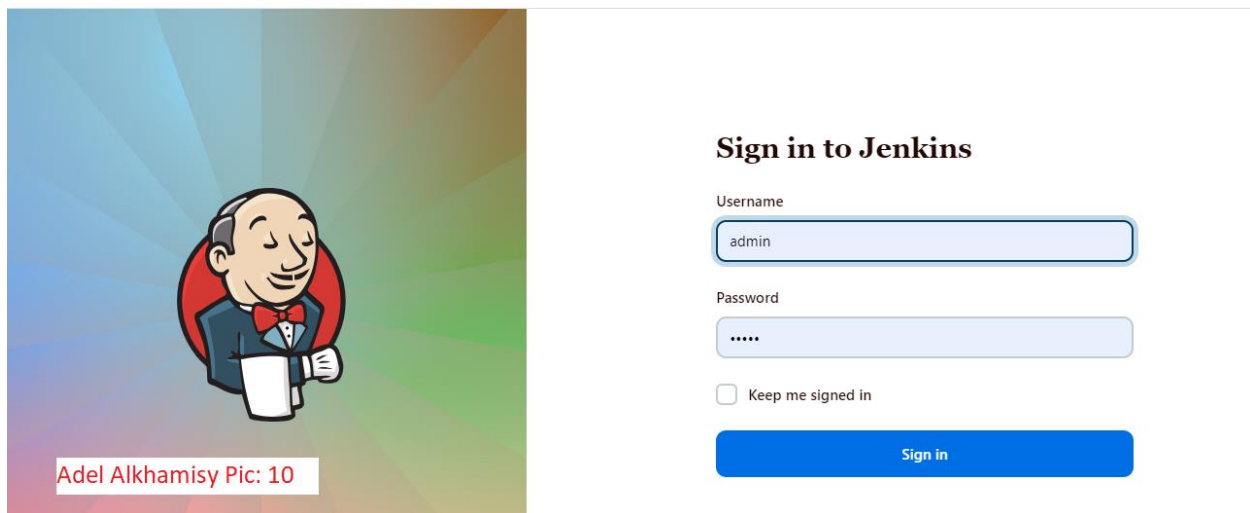
Adel Alkhamisy Pic: 8

8. Create Dockerfile to build image

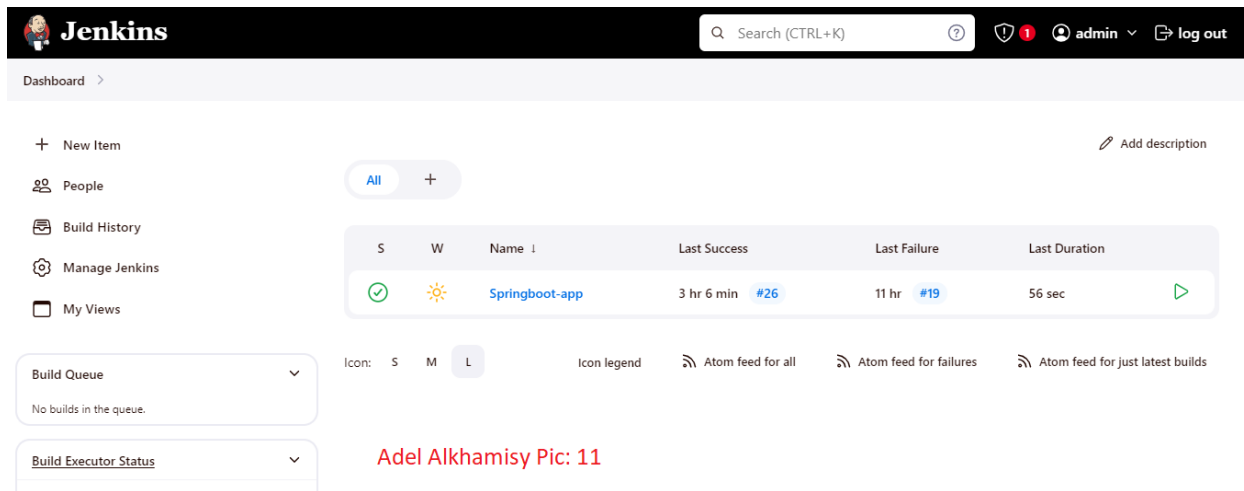
```
Dockerfile x
1 FROM openjdk:17-jdk
2
3 LABEL author.name="Adel Alkhamisy"
4
5 EXPOSE 8080
6
7 ARG JAR_FILE=./target/survey-0.0.1-SNAPSHOT.jar
8
9 COPY ${JAR_FILE} .
10
11 CMD [ "java", "-jar", "/survey-0.0.1-SNAPSHOT.jar"]
```

Adel Alkhamisy Pic: 9

9. Configure Jenkins to build pipeline



10. Jenkins dashboard

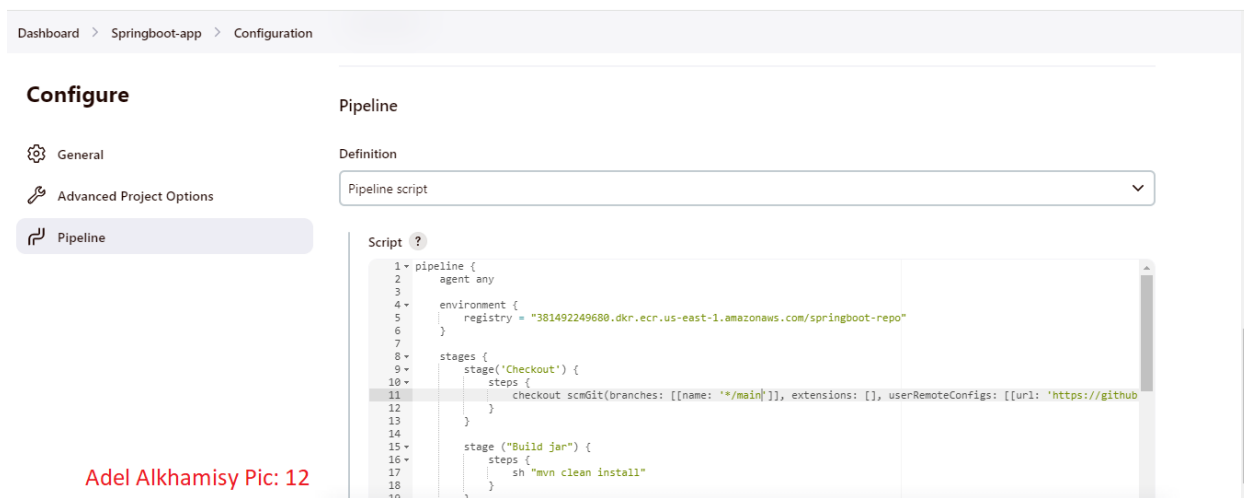


The screenshot shows the Jenkins dashboard. At the top is a navigation bar with the Jenkins logo, a search bar, and user information (admin). Below the navigation bar is a sidebar with links: New Item, People, Build History, Manage Jenkins, and My Views. The main content area displays a table of builds for the 'Springboot-app' job. The table has columns for status (S), warnings (W), name, last success, last failure, and last duration. The current build is #26, which is successful and took 56 seconds. Below the table, there are sections for 'Build Queue' (empty) and 'Build Executor Status'.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀	Springboot-app	3 hr 6 min #26	11 hr #19	56 sec

Adel Alkhamisy Pic: 11

11. Jenkins project configuration. Complete Jenkins file is in GitHub repository



The screenshot shows the Jenkins configuration page for the 'Springboot-app' job. The 'Pipeline' tab is selected. The 'Definition' section shows the 'Pipeline script' definition. The script is a Jenkinsfile that defines a pipeline with two stages: 'Checkout' and 'Build jar'. The 'Checkout' stage uses the 'checkout' step to fetch the code from a GitHub repository. The 'Build jar' stage uses the 'sh' step to run 'mvn clean install'.

```
1 pipeline {
2   agent any
3
4   environment {
5     registry = "381492249680.dkr.ecr.us-east-1.amazonaws.com/springboot-repo"
6   }
7
8   stages {
9     stage('Checkout') {
10      steps {
11        checkout scmGit(branches: [[name: '*/main']], extensions: [], userRemoteConfigs: [[url: 'https://github.com/adelalkhamisy/springboot-app.git']])
12      }
13    }
14
15    stage('Build jar') {
16      steps {
17        sh "mvn clean install"
18      }
19    }
20  }
```

Adel Alkhamisy Pic: 12

12. Jenkins build pipeline

The screenshot shows the Jenkins web interface. At the top, there's a navigation bar with the Jenkins logo, a search bar, and user information. Below the navigation bar, the breadcrumb trail shows 'Dashboard > Springboot-app >'. The main content area is divided into two sections. On the left, there's a sidebar with various actions: 'Status' (selected), 'Changes', 'Build Now', 'Configure', 'Delete Pipeline', 'Full Stage View', 'Rename', and 'Pipeline Syntax'. On the right, the 'Springboot-app' pipeline is shown with a green checkmark icon. Below the pipeline name, there's a 'Stage View' section. It displays a table of stages and their durations. The stages are 'Checkout', 'Build jar', 'Build image', 'Push Image', and 'Helm Deploy'. The durations are: Checkout (356ms), Build jar (17s), Build image (3s), Push Image (4s), and Helm Deploy (1s). Below the table, there's a section for 'Average stage times' and 'Average full run time: ~33s'. The 'Average stage times' section shows a bar chart with the following values: Checkout (401ms), Build jar (16s), Build image (2s), Push Image (4s), and Helm Deploy (1s).

Springboot-app

Stage View

Stage	Duration
Checkout	356ms
Build jar	17s
Build image	3s
Push Image	4s
Helm Deploy	1s

Average stage times:
(Average full run time: ~33s)

Stage	Duration
Checkout	401ms
Build jar	16s
Build image	2s
Push Image	4s
Helm Deploy	1s

13. List of nodes

The screenshot shows the AWS CLI terminal. The user is logged in as 'Alex_admin' in the 'N. Virginia' region. The terminal shows the command 'kubectl get nodes' being executed. The output is as follows:

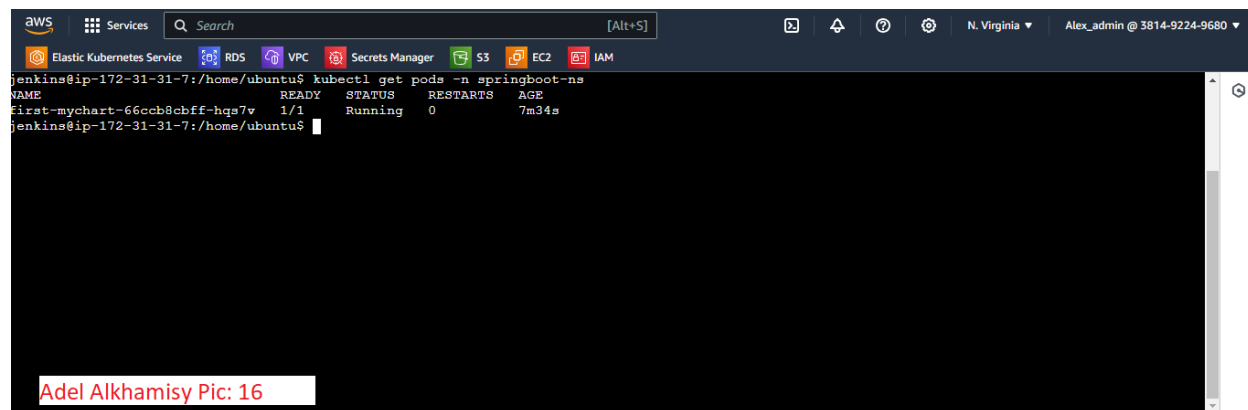
```
jenkins@ip-172-31-31-7:/home/ubuntu$ kubectl get nodes
NAME                                STATUS    ROLES    AGE    VERSION
ip-192-168-18-25.ec2.internal        Ready    <none>    16h    v1.29.0-eks-5e0fdde
ip-192-168-34-104.ec2.internal       Ready    <none>    16h    v1.29.0-eks-5e0fdde
jenkins@ip-172-31-31-7:/home/ubuntu$
```

14. List of namespaces

The screenshot shows the AWS CLI terminal. The user is logged in as 'Alex_admin' in the 'N. Virginia' region. The terminal shows the command 'kubectl get ns' being executed. The output is as follows:

```
jenkins@ip-172-31-31-7:/home/ubuntu$ kubectl get ns
NAME              STATUS    AGE
default           Active    16h
kube-node-lease   Active    16h
kube-public        Active    16h
kube-system        Active    16h
springboot-ns     Active    12h
jenkins@ip-172-31-31-7:/home/ubuntu$
```

15. Kubernetes pod




The screenshot shows an AWS CLI terminal window with the command `kubectl get pods -n springboot-ns` executed. The output displays a single pod named `first-mychart-66ecb8cbff-hgs7v` in a `Running` state. The terminal interface includes the AWS Services menu at the top and a red watermark at the bottom.

```
jenkins@ip-172-31-31-7:/home/ubuntu$ kubectl get pods -n springboot-ns
NAME                                READY    STATUS    RESTARTS   AGE
first-mychart-66ecb8cbff-hgs7v      1/1      Running   0           7m34s
jenkins@ip-172-31-31-7:/home/ubuntu$
```

Adel Alkhamisy Pic: 16

16. Kubernetes service



The screenshot shows an AWS CLI terminal window with the command `kubectl get svc -n springboot-ns` executed. The output shows a service named `first-mychart` of type `LoadBalancer`. The terminal interface includes the AWS Services menu at the top and a red watermark at the bottom.

```
jenkins@ip-172-31-31-7:/home/ubuntu$ kubectl get svc -n springboot-ns
NAME      TYPE           CLUSTER-IP      EXTERNAL-IP                                PORT(S)      AGE
first-mychart LoadBalancer  10.100.255.58    a30c22c64d59a4ff387d6d41dc329df5-1200801463.us-east-1.elb.amazonaws.com  80:31927/TCP  11h
jenkins@ip-172-31-31-7:/home/ubuntu$
```

Adel Alkhamisy Pic: 17

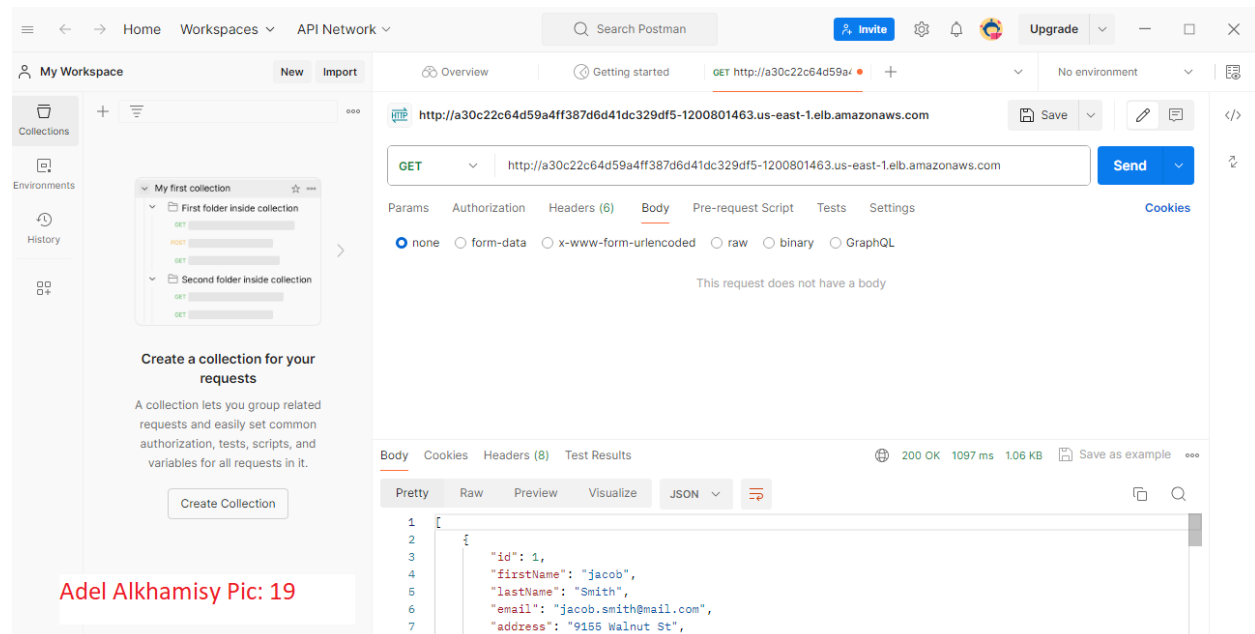
17. Output

Pretty print ☐

```
[{"id":1,"firstName":"jacob","lastName":"Smith","email":"jacob.smith@mail.com","address":"9155 Walnut St","city":"New York","state":"NY","zip":"22031","telephone":"8844448888","dateOfSurvey":"04-31-2022","recommendation":"Very Likely","interest":"Frinds","likings":"Students"}, {"id":2,"firstName":"Alice","lastName":"Smith","email":"alice.smith@mail.com","address":"123 Oak St","city":"San Francisco","state":"CA","zip":"94102","telephone":"9999999999","dateOfSurvey":"04-29-2022","recommendation":"Likely","interest":"Television","likings":"Location"}, {"id":3,"firstName":"Bob","lastName":"Johnson","email":"bob.johnson@mail.com","address":"789 Elm St","city":"Los Angeles","state":"CA","zip":"90001","telephone":"7777777777","dateOfSurvey":"04-30-2022","recommendation":"Unlikely","interest":"Internet","likings":"Atmosphere"}]
```

Adel Alkhamisy Pic: 18

18. GET method



Output:

```
[
  {
    "id": 1,
    "firstName": "jacob",
    "lastName": "Smith",
    "email": "jacob.smith@mail.com",
    "address": "9155 Walnut St",
    "city": "New York",
    "state": "NY",
    "zip": "22031",
    "telephone": "8844448888",
    "dateOfSurvey": "04-31-2022",
    "recommendation": "Very Likely",
    "interest": "Frinds",
    "likings": "Students"
  },
  {
    "id": 2,
    "firstName": "Alice",
    "lastName": "Smith",
    "email": "alice.smith@mail.com",
    "address": "123 Oak St",
    "city": "San Francisco",
    "state": "CA",
```

```

    "zip": "94102",
    "telephone": "9999999999",
    "dateOfSurvey": "04-29-2022",
    "recommendation": "Likely",
    "interest": "Television",
    "likings": "Location"
  },
  {
    "id": 3,
    "firstName": "Bob",
    "lastName": "Johnson",
    "email": "bob.johnson@mail.com",
    "address": "789 Elm St",
    "city": "Los Angeles",
    "state": "CA",
    "zip": "90001",
    "telephone": "7777777777",
    "dateOfSurvey": "04-30-2022",
    "recommendation": "Unlikely",
    "interest": "Internet",
    "likings": "Atmosphere"
  }
]

```

19. POST method

The screenshot shows the Postman interface with a POST request configured. The URL is `http://a30c22c64d59a4ff387d6d41dc329df5-1200801463.us-east-1.elb.amazonaws.com`. The request body is a JSON array of two user objects. The response is a JSON object representing a user.

Request Body (JSON):

```

[
  {
    "id": 6,
    "firstName": "Jacob",
    "lastName": "Johnson",
    "email": "jacob.johnson@mail.com",
    "address": "789 Elm St",
    "city": "Los Angeles",
    "state": "CA",
    "zip": "90001",
    "telephone": "7777776666",
    "dateOfSurvey": "06-30-2022",
    "recommendation": "Unlikely"
  }
]

```

Response Body (JSON):

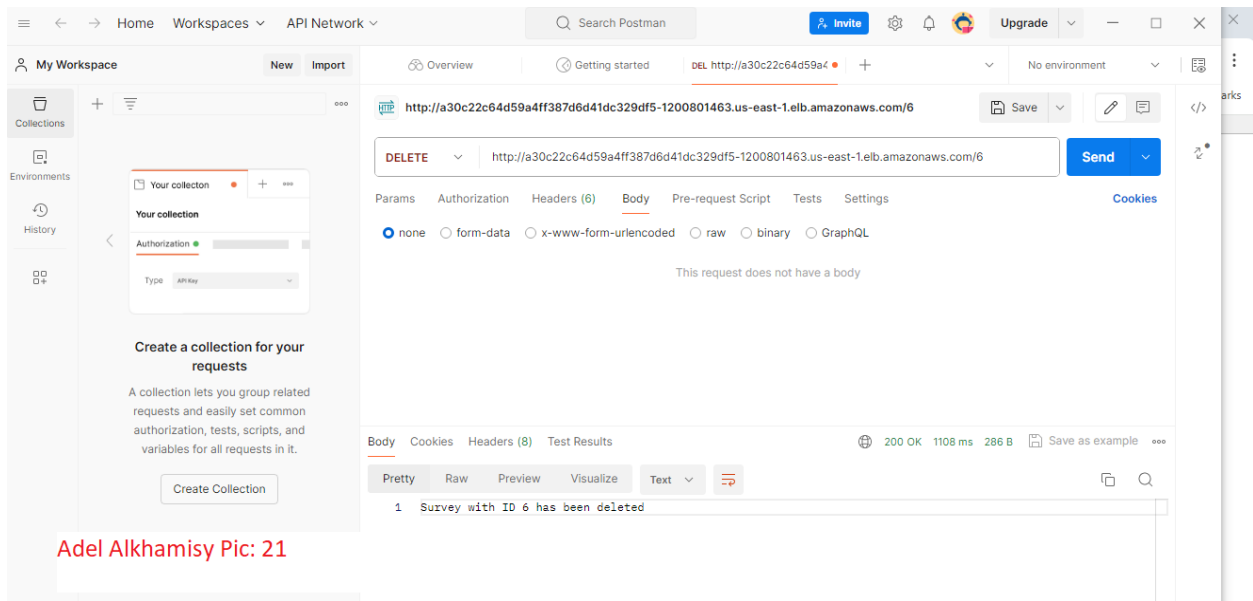
```

{
  "id": 6,
  "firstName": "Jacob",
  "lastName": "Johnson",
  "email": "jacob.johnson@mail.com",
  "address": "789 Elm St",
  "city": "Los Angeles",
  "state": "CA",
  "zip": "90001",
  "telephone": "7777776666",
  "dateOfSurvey": "06-30-2022",
  "recommendation": "Unlikely"
}

```

Adel Alkhamisy Pic: 20

20. DELETE method



Video URL:

<https://www.loom.com/share/09b3f2fd8a9343afa4eaf0ce93c78a9c?sid=29f0f74d-8b25-4418-b0a8-e731154a1168>