

School: CLOUDBOOSTA	Course: DevOps
Name: AKOSA SAMUEL ONYEJEKWE	Assignment-5-of-5_ Jenkins-2
Student No: CBALC0168	Week: 3 of 8
	Issue Date: May 19, 2026

Tasks:

1. Configure AWS Terraform backend infrastructure
2. Create Amazon S3 Terraform state bucket
3. Configure DynamoDB Terraform state locking table
4. Verify Terraform backend configuration
5. Create Amazon ECR repository
6. Configure Amazon EKS cluster infrastructure
7. Verify Amazon Elastic Load Balancer configuration
8. Launch Jenkins EC2 server instance
9. Configure Jenkins EC2 networking and security groups
10. Configure Jenkins inbound security group rules
11. Install Terraform and AWS CLI using EC2 user data
12. Connect to Jenkins EC2 server instance
13. Retrieve Jenkins initial administrator password
14. Unlock Jenkins administration console
15. Install Jenkins suggested plugins
16. Configure Jenkins administrator account
17. Configure Jenkins instance URL settings
18. Verify Jenkins installation completion
19. Install Jenkins AWS and Docker pipeline plugins
20. Configure Jenkins AWS credentials
21. Generate GitHub personal access token
22. Configure Jenkins GitHub authentication credentials
23. Verify Jenkins global credentials configuration
24. Fork E-CommerceWebsite GitHub repository
25. Clone GitHub repository into local development environment
26. Configure Terraform backend infrastructure code
27. Initialize Terraform backend configuration
28. Execute Terraform infrastructure planning
29. Apply Terraform backend infrastructure deployment
30. Configure Jenkins CI/CD pipeline job
31. Configure Jenkins GitHub SCM integration
32. Configure Jenkins pipeline Jenkinsfile execution
33. Execute Jenkins E-Commerce CI/CD pipeline

34. Verify Jenkins failed deployment execution logs
35. Verify Jenkins successful deployment execution
36. Configure Jenkins GitHub webhook trigger
37. Configure GitHub repository webhook integration
38. Verify GitHub webhook delivery configuration
39. Update E-Commerce frontend application source code
40. Push frontend application updates to GitHub repository
41. Verify automated Jenkins webhook pipeline execution
42. Verify deployed E-Commerce application interface
43. Verify deployed application load balancer endpoint
44. Execute Jenkins Terraform destroy pipeline workflow
45. Verify Terraform infrastructure destruction workflow
46. Validate Jenkins CI/CD automation lifecycle

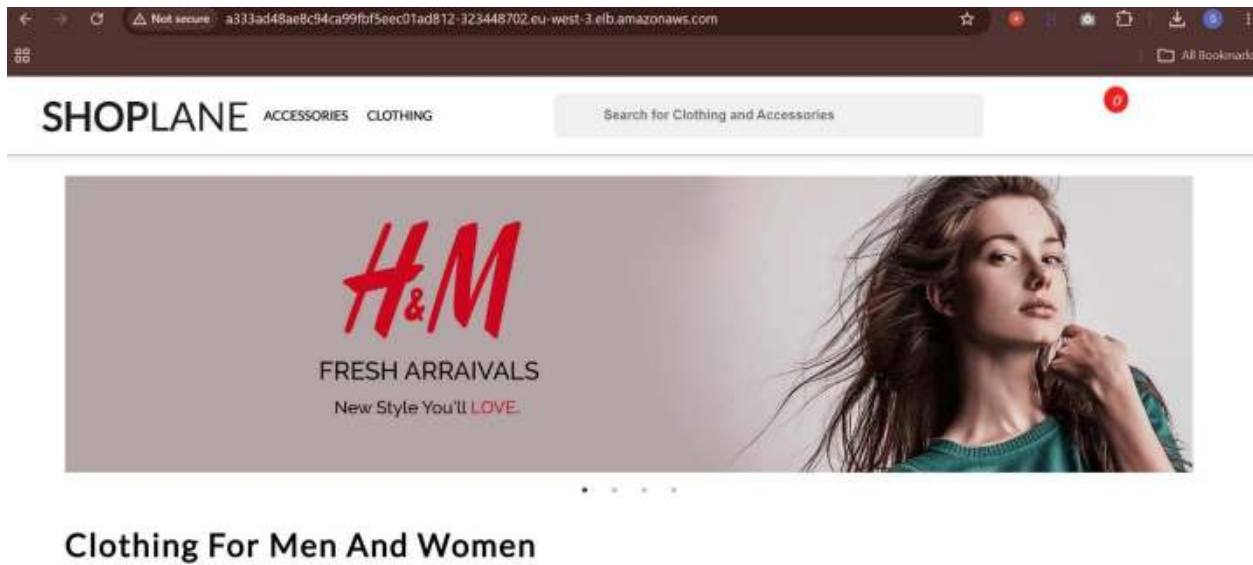
Goal:

To configure a secure cloud-native Jenkins CI/CD automation environment using Terraform, AWS EKS, Amazon ECR, GitHub integration, Jenkins pipeline automation, webhook-triggered deployment workflows, Terraform backend infrastructure management, and automated DevOps orchestration within an AWS cloud infrastructure environment.

Result:

- Amazon S3 Terraform backend bucket configured successfully
- DynamoDB Terraform state locking table configured successfully
- Amazon ECR repository configured successfully
- Amazon EKS cluster configured successfully
- Jenkins EC2 server launched successfully
- Jenkins security groups configured successfully
- Terraform and AWS CLI installed successfully
- Jenkins administrator account configured successfully
- Jenkins plugins installed successfully
- Jenkins AWS credentials configured successfully
- GitHub personal access token configured successfully
- Jenkins GitHub credentials configured successfully
- Jenkins CI/CD pipeline configured successfully
- Terraform backend infrastructure initialized successfully
- Terraform infrastructure deployment executed successfully

- Jenkins GitHub SCM integration configured successfully
- Jenkins webhook trigger configured successfully
- GitHub webhook delivery configured successfully
- E-Commerce application deployed successfully
- Jenkins deployment logs validated successfully
- Jenkins automated pipeline execution validated successfully
- Terraform destroy workflow executed successfully
- Jenkins CI/CD lifecycle automation validated successfully





Clothing For Men And Women



Men Navy Blue Solid
Sweatshirt

United Colors Of
Benetton

Rs 2599



Men Black MAMGP T7
Sweat Sporty Jacket

Puma

Rs 7999



Men Black Action
Parkview Lifestyle Shoes

Hush Puppies

Rs 6999



Women Black Solid
Lightweight Leather
Jacket

BARESKIN

Rs 9999



Women Blue Solid Shirt
Dress

SASSAFRAS

Rs 5200

Accessories For Men And Women



Unisex Silver-Toned Series
3 Smart Watch

Apple

Rs 31999



Unisex Black & Green
Reflex 2.0 Smart Band

Fastrack

Rs 1999



Unisex Black Galaxy Fit
Fitness Band

Samsung

Rs 9990



Gear IconX Black Cord-
Free Fitness Earbuds

Samsung

Rs 13990



White 2nd Gen AirPods
With Charging Case

Apple

Rs 14999

ONLINE STORE

MEN CLOTHING
WOMEN CLOTHING
MEN ACCESSORIES
WOMEN ACCESSORIES

HELPFUL LINK

HOME
ABOUT
CONTACT

PARTNERS

ZARA
PANTALOONS
LEVIS
UCB
+ MANY MORE

ADDRESS

BUILDING 101
CENTRAL AVENUE
LA - 902722
UNITED STATES

Amazon S3

General purpose buckets (18) Info

Buckets are containers for data stored in S3.

Search: ecom 1 match

Name	AWS Region	Creation date
ecommerce-terraform-state-009852102027	Europe (Paris) eu-west-3	May 23, 2026, 18:37:58 (UTC+03:00)

DynamoDB

Tables (1) Info

Share your feedback on Amazon DynamoDB

Find tables

Filter by tag: Any tag key

Filter by tag value: Any tag value

Name	Status	Partition key	Sort key	Indexes	Replication Regions	Deletion
ecommerce-terraform-locks	Active	LockID (S)	-	0	0	Off

Amazon ECR

Private registries

Private repositories (1)

Search by repository substring

Repository name	URI	Created at	Tag immutability	Encryption type
projectme-ak	009850210027.dkr.ecr.eu-west-3.amazonaws.com/projectme-ak	May 23, 2026, 18:05:26 (UTC+03)	Mutable	AES-256

Amazon Elastic Kubernetes Service

Clusters (1) Info

Filter clusters

Cluster name	Status	Kubernetes version	Support period	Upgrade policy
ecom-cluster	Active	1.35	Standard support until March 27, 2027	Extended support

Instances (1) Info

Connect

Instance state

Actions

Launch instances

Find Instance by attribute or tag (case-sensitive)

Name	Instance ID	Instance state	Instance type	Status check	Availability Zone
Jenkins-Server	i-0cf98ace9b59b4fd9	Running	t2.medium	2/2 checks passed	eu-west-3a

EC2 > Load balancers

Load balancers

Network & Security

Load balancers (1) [What's new?](#)

Actions

Create load balancer

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Filter load balancers

< 1 >

☐	Name	State	Type	Scheme	IP addr...	VPC ID	Availab
☐	a333ad48ae8c94ca99fb...	-	classic	-	-	vpc-05c50ea1c4a62fb04	2 Availa



Clothing For Men And Women



Men Navy Blue Solid
Sweatshirt

United Colors Of
Benetton

Rs 2599



Men Black MAMGP T7
Sweat Sporty Jacket

Puma

Rs 7999



Men Black Action
Parkview Lifestyle Shoes

Hush Puppies

Rs 6999



Women Black Solid
Lightweight Leather
Jacket

BARESKIN

Rs 9999



Women Blue Solid Shirt
Dress

SASSAFRAS

Rs 5200

Accessories For Men And Women



Unisex Silver-Toned Series
3 Smart Watch

Apple

Rs 31999



Unisex Black & Green
Reflex 2.0 Smart Band

Fastrack

Rs 1999



Unisex Black Galaxy Fit
Fitness Band

Samsung

Rs 9990



Gear IconX Black Cord-
Free Fitness Earbuds

Samsung

Rs 13990



White 2nd Gen AirPods
With Charging Case

Apple

Rs 14999

ONLINE STORE

MEN CLOTHING
WOMEN CLOTHING
MEN ACCESSORIES
WOMEN ACCESSORIES

HELPFUL LINK

HOME
ABOUT
CONTACT

PARTNERS

ZARA
PANTALOONS
LEVIS
UCB
+ MANY MORE

ADDRESS

BUILDING 101
CENTRAL AVENUE
LA - 90272
UNITED STATES

Action:

Step 1: Verified Terraform Backend Infrastructure

- Opened: Amazon S3 Console
- Verified Terraform backend bucket created successfully
- Verified DynamoDB Terraform locking table configured successfully
- Confirmed Terraform remote backend infrastructure operational successfully

Step 2: Created Amazon ECR Repository

- Opened: Amazon ECR → Private repositories
- Configured repository successfully
- Verified ECR repository created successfully
- Confirmed container repository available successfully

Step 3: Verified Amazon EKS Cluster Configuration

- Opened: Amazon Elastic Kubernetes Service → Clusters
- Verified EKS cluster status: Active
- Confirmed Kubernetes cluster configured successfully

Step 4: Verified Elastic Load Balancer Configuration

- Opened: EC2 → Load Balancers
- Verified load balancer configured successfully
- Confirmed load balancer DNS generated successfully

Step 5: Launched Jenkins EC2 Server Instance

- Opened: EC2 → Launch Instance
- Configured instance name: Jenkins-Server
- Selected Ubuntu Server 24.04 LTS AMI
- Selected instance type: t2.medium
- Verified Jenkins EC2 server launched successfully

Step 6: Configured Jenkins EC2 Networking

- Configured VPC successfully
- Configured subnet successfully

- Enabled public IP assignment successfully
- Confirmed EC2 networking configured successfully

Step 7: Configured Jenkins Security Group

- Configured security group name: jenkins_SG
- Configured security group description successfully
- Verified security group created successfully

Step 8: Configured Jenkins Inbound Security Rules

- Added SSH rule:
 - Port: 22
 - Source: Anywhere
- Added HTTP rule:
 - Port: 80
 - Source: Anywhere
- Added Jenkins TCP rule:
 - Port: 8080
 - Source: Anywhere
- Confirmed Jenkins inbound security rules configured successfully

Step 9: Configured EC2 User Data Automation

- Configured Terraform installation commands successfully
- Configured AWS CLI installation commands successfully
- Verified EC2 bootstrap automation configured successfully

Step 10: Verified Jenkins EC2 Instance Status

- Opened: EC2 → Instances
- Verified Jenkins instance state: Running
- Confirmed EC2 health checks passed successfully

Step 11: Connected to Jenkins EC2 Server

- Opened: Connect to Instance

- Selected: EC2 Instance Connect
- Connected to Jenkins EC2 instance successfully

Step 12: Retrieved Jenkins Initial Administrator Password

- Executed Jenkins password retrieval command successfully
- Verified Jenkins administrator password generated successfully

Step 13: Unlocked Jenkins Administration Console

- Opened Jenkins browser URL successfully
- Entered administrator password successfully
- Verified Jenkins unlock completed successfully

Step 14: Installed Jenkins Suggested Plugins

- Selected: Install suggested plugins
- Verified Jenkins plugin installation started successfully
- Confirmed plugin installation completed successfully

Step 15: Configured Jenkins Administrator Account

- Configured administrator username successfully
- Configured administrator password successfully
- Configured email address successfully
- Verified Jenkins administrator account created successfully

Step 16: Configured Jenkins Instance URL

- Configured Jenkins URL successfully
- Saved Jenkins instance configuration successfully
- Verified Jenkins installation completed successfully

Step 17: Installed Jenkins AWS and Docker Plugins

- Opened: Manage Jenkins → Plugins → Available Plugins
- Installed:
 - AWS Global Configuration

- Pipeline AWS Steps
 - Docker Pipeline
 - GitHub Integration
- Confirmed Jenkins plugins installed successfully

Step 18: Verified Jenkins Plugin Download Progress

- Opened: Plugins Download Progress
- Verified successful plugin installations
- Confirmed Jenkins restart completed successfully

Step 19: Configured Jenkins AWS Credentials

- Opened: Manage Jenkins → AWS
- Added AWS access credentials successfully
- Configured:
 - AWS Region: Europe (Paris)
- Verified Jenkins AWS credentials configured successfully

Step 20: Generated GitHub Personal Access Token

- Opened: GitHub → Developer Settings → Personal Access Tokens
- Configured GitHub token scopes successfully
- Generated GitHub access token successfully
- Verified GitHub token creation completed successfully

Step 21: Configured Jenkins GitHub Credentials

- Opened: Jenkins → Credentials → Global
- Added Username with password credential successfully
- Configured:
 - Credential ID: github-token
- Verified Jenkins GitHub credentials configured successfully

Step 22: Verified Jenkins Global Credentials Storage

- Verified configured credentials:
 - AWS credentials
 - GitHub credentials
- Confirmed Jenkins credentials stored securely successfully

Step 23: Forked E-CommerceWebsite Repository

- Opened: GitHub Repository
- Selected: Create Fork
- Verified repository fork created successfully

Step 24: Cloned GitHub Repository into Local Environment

- Executed Git clone command successfully
- Verified repository cloned into Ubuntu environment successfully

Step 25: Configured Terraform Backend Code

- Opened Terraform backend configuration files successfully
- Configured:
 - S3 backend bucket
 - DynamoDB locking table
- Verified Terraform backend code configured successfully

Step 26: Initialized Terraform Backend Infrastructure

- Executed: terraform init
- Verified Terraform initialization completed successfully
- Confirmed Terraform backend operational successfully

Step 27: Executed Terraform Infrastructure Planning

- Executed: terraform plan
- Verified infrastructure changes generated successfully
- Confirmed Terraform execution plan validated successfully

Step 28: Applied Terraform Infrastructure Deployment

- Executed: terraform apply --auto-approve
- Verified AWS resources created successfully
- Confirmed Terraform infrastructure deployment completed successfully

Step 29: Configured Jenkins CI/CD Pipeline Job

- Opened: Jenkins Dashboard → New Item
- Configured:
 - Pipeline name: E-Commerce CICD Pipeline
- Selected: Pipeline
- Verified Jenkins pipeline created successfully

Step 30: Configured Jenkins GitHub SCM Integration

- Configured GitHub repository URL successfully
- Configured Jenkins GitHub credentials successfully
- Configured:
 - Branch specification: */master
- Verified GitHub SCM integration configured successfully

Step 31: Configured Jenkins Jenkinsfile Pipeline Execution

- Configured:
 - Script Path: Jenkinsfile

- Enabled lightweight checkout successfully
- Saved Jenkins pipeline configuration successfully

Step 32: Executed Jenkins E-Commerce Pipeline

- Opened: Build with Parameters
- Configured:
 - AWS account
 - AWS region
- Executed Jenkins CI/CD pipeline successfully

Step 33: Verified Failed Jenkins Pipeline Execution

- Opened Jenkins Console Output
- Verified failed Terraform destroy execution logs successfully
- Confirmed Jenkins failure logs captured successfully

Step 34: Verified Successful Jenkins Pipeline Execution

- Opened Jenkins pipeline status dashboard
- Verified successful build execution successfully
- Confirmed Jenkins deployment workflow executed successfully

Step 35: Verified Elastic Load Balancer Endpoint

- Opened AWS Load Balancer Console
- Verified generated DNS endpoint successfully
- Confirmed deployed application endpoint active successfully

Step 36: Configured Jenkins GitHub Webhook Trigger

- Opened: Jenkins Pipeline → Configure → Triggers
- Enabled:
 - GitHub hook trigger for GITScm polling
- Saved Jenkins trigger configuration successfully

Step 37: Configured GitHub Repository Webhook

- Opened: GitHub Repository → Settings → Webhooks
- Configured:
 - Jenkins webhook URL
 - Content type: application/json
- Verified GitHub webhook configured successfully

Step 38: Verified GitHub Webhook Delivery

- Opened: GitHub → Recent Deliveries
- Verified webhook payload delivery successful
- Confirmed Jenkins webhook integration operational successfully

Step 39: Updated E-Commerce Frontend Application

- Opened frontend source code successfully
- Modified frontend application content successfully
- Verified frontend application updates completed successfully

Step 40: Pushed Frontend Updates to GitHub Repository

- Executed:
 - git add
 - git commit
 - git push
- Verified GitHub repository updated successfully

Step 41: Verified Automated Jenkins Webhook Execution

- Opened Jenkins pipeline dashboard
- Verified automated webhook-triggered build execution successfully
- Confirmed Jenkins automation workflow functioning successfully

Step 42: Verified E-Commerce Application Deployment

- Opened deployed Shoplane application successfully
- Verified frontend interface rendered successfully
- Confirmed E-Commerce application deployment validated successfully

Step 43: Verified Application Footer and Product Interface

- Verified:
 - Clothing product cards
 - Accessories product cards
 - Footer navigation links
- Confirmed frontend application rendering validated successfully

Step 44: Executed Jenkins Terraform Destroy Workflow

- Opened Jenkins → Build with Parameters
- Enabled:
 - DESTROY parameter
- Executed Terraform destroy workflow successfully

Step 45: Verified Terraform Infrastructure Destruction

- Opened Jenkins console execution logs
- Verified Terraform destroy completed successfully
- Confirmed AWS infrastructure resources removed successfully

Step 46: Validated Jenkins CI/CD Automation Lifecycle

- Verified:
 - Jenkins automation workflows
 - GitHub webhook automation
 - Terraform infrastructure lifecycle
 - Automated deployment orchestration
- Confirmed end-to-end DevOps CI/CD lifecycle validated successfully

Summary

- Terraform backend infrastructure configured successfully
- Amazon S3 and DynamoDB backend services validated successfully
- Jenkins EC2 cloud-native environment configured successfully
- Jenkins AWS and GitHub integrations configured successfully
- Jenkins pipeline automation configured successfully
- GitHub webhook automation validated successfully
- Terraform deployment and destroy workflows validated successfully
- E-Commerce application deployment validated successfully
- Jenkins CI/CD cloud automation workflow validated successfully
- End-to-end DevOps orchestration lifecycle implemented successfully

Lab Objective Achieved:

Implemented Jenkins CI/CD automation, Terraform backend infrastructure management, AWS EKS orchestration, Amazon ECR repository management, Jenkins GitHub integration, webhook-triggered deployment automation, Terraform infrastructure lifecycle management, secure Jenkins credential configuration, automated E-Commerce application deployment, and cloud-native DevOps orchestration successfully within AWS cloud infrastructure and Jenkins CI/CD environment.



Clothing For Men And Women

H&M

FRESH ARRIVAALS

New Style You'll LOVE.



Clothing For Men And Women



Men Navy Blue Solid
Sweatshirt

United Colors Of
Benetton

Rs 2599



Men Black MAMGP T7
Sweat Sporty Jacket

Puma

Rs 7999



Men Black Action
Parkview Lifestyle Shoes

Hush Puppies

Rs 6999



Women Black Solid
Lightweight Leather
Jacket

BARESKIN

Rs 9999



Women Blue Solid Shirt
Dress

SASSAFRAS

Rs 5200

Accessories For Men And Women



Unisex Silver-Toned Series
3 Smart Watch

Apple

Rs 31999



Unisex Black & Green
Reflex 2.0 Smart Band

Fastrack

Rs 1999



Unisex Black Galaxy Fit
Fitness Band

Samsung

Rs 9990



Gear IconX Black Cord-
Free Fitness Earbuds

Samsung

Rs 13990



White 2nd Gen AirPods
With Charging Case

Apple

Rs 14999

ONLINE STORE

MEN CLOTHING
WOMEN CLOTHING
MEN ACCESSORIES
WOMEN ACCESSORIES

HELPFUL LINK

HOME
ABOUT
CONTACT

PARTNERS

ZARA
PANTALOONS
LEVIS
UCB
+ MANY MORE

ADDRESS

BUILDING 101
CENTRAL AVENUE
LA - 902722
UNITED STATES

Amazon S3

Amazon S3

Buckets

General purpose buckets

Directory buckets

Table buckets

Vector buckets

Files

General purpose buckets (18) Info

Copy ARN

Empty

Delete

Create bucket

Search ecom1 match

<1>⚙️

Name	AWS Region	Creation date
ecommerce-terraform-state-009852102027	Europe (Paris) eu-west-3	May 23, 2026, 18:37:58 (UTC+03:00)

DynamoDB

Tables

Dashboard

Tables

Explore items

PartiQL editor

Backups

Exports to S3

Imports from S3

Integrations

Reserved capacity

Settings

Share your feedback on Amazon DynamoDB

Your feedback is an important part of helping us provide a better customer experience. Take this short survey to let us know how we're doing.

Share feedback

Tables (1) Info

Last updated May 23, 2026, 18:28 (UTC+3:00)

Actions

Delete

Create table

Find tables

Filter by tagAny tag key

Filter by tag valueAny tag value

< 1 >

Name	Status	Partition key	Sort key	Indexes	Replication Regions	Deletion
ecommerce-terraform-locks	Active	LockID (S)	-	0	0	Off

Amazon ECR

Private registry

Repositories

Amazon Elastic Container Registry

Private registry

Repositories

Features & Settings

Public registry

Repositories

Private repositories (1)

View push commands

Delete

Actions

Create repository

Search by repository substring

Repository name	URI	Created at	Tag immutability	Encryption type
 projectme-ak	 009850210027.dkr.ecr.eu-west-3.amazonaws.com/projectme-ak	May 23, 2026, 18:05:26 (UTC+03)	Mutable	AES-256

Amazon Elastic Kubernetes Service

Dashboard

Clusters

▼ Settings

Amazon Elastic Kubernetes Service

> Clusters

Clusters (1) Info

Filter clusters

Cluster name ▲

Status ▼

Kubernetes version ▼

Support period ▼

Upgrade policy ▼

ecom-cluster

Active

1.35

Standard support until March 27, 2027

Extended support

Refresh

Delete

Create cluster

Instances (1)

Info

Connect

Instance state ▾

Actions ▾

Launch instances

▾

Saved filter sets

Choose filter set ▾

Find Instance by attribute or tag (case-sensitive)

<

1

>

☐	Name	Instance ID	Instance state ▾	Instance type ▾	Status check	Availability Zone
☐	Jenkins-Server	i-0cf98ace9b59b4fd9	Running	t2.medium	2/2 checks passed	eu-west-3a

Instances (1)

Info

Connect

Instance state ▾

Actions ▾

Launch instances

▾

Saved filter sets

Choose filter set ▾

Find Instance by attribute or tag (case-sensitive)

<

1

>

☐	Name	Instance ID	Instance state ▾	Instance type ▾	Status check	Availability Zone
☐	Jenkins-Server	i-0cf98ace9b59b4fd9	Running	t2.medium	2/2 checks passed	eu-west-3a

EC2 > Load balancers

Snapshots

Lifecycle Manager

▼ Network & Security

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Network Interfaces

Load balancers (1) [What's new?](#)

Actions

Create load balancer

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Filter load balancers

< 1 >

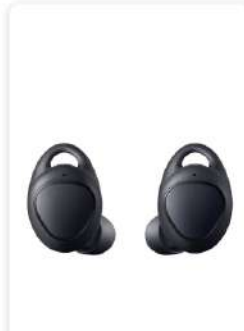
☐	Name	State	Type	Scheme	IP addr...	VPC ID	Availabi
☐	a333ad48ae8c94ca99fb...	-	classic	-	-	vpc-05c50ea1c4a62fb04	2 Availa



Clothing For Men And Women

 Men Navy Blue Solid Sweatshirt United Colors Of Benetton Rs 2599	 Men Black MAMGP T7 Sweat Sporty Jacket Puma Rs 7999	 Men Black Action Parkview Lifestyle Shoes Hush Puppies Rs 6999	 Women Black Solid Lightweight Leather Jacket BARESKIN Rs 9999	 Women Blue Solid Shirt Dress SASSAFRAS Rs 5200
--	---	--	--	--

Accessories For Men And Women

 Unisex Silver-Toned Series 3 Smart Watch Apple Rs 31999	 Unisex Black & Green Reflex 2.0 Smart Band Fastrack Rs 1999	 Unisex Black Galaxy Fit Fitness Band Samsung Rs 9990	 Gear IconX Black Cord-Free Fitness Earbuds Samsung Rs 13990	 White 2nd Gen AirPods With Charging Case Apple Rs 14999
---	---	--	--	---

ONLINE STORE

MEN CLOTHING
WOMEN CLOTHING
MEN ACCESSORIES
WOMEN ACCESSORIES

HELPFUL LINK

HOME
ABOUT
CONTACT

PARTNERS

ZARA
PANTALOONS
LEVIS
UCB
+ MANY MORE

ADDRESS

BUILDING 101
CENTRAL AVENUE
LA - 902722
UNITED STATES

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name

Jenkins-Server

Add additional tags

▼ Summary

Number of instances [Info](#)

1

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd64...[read more](#)

ami-0adb8ca49015e0901

Virtual server type (instance type)

Recents

Quick Start

Amazon Linux



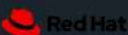
Ubuntu



Windows



Red Hat



SUSE Linux



Debian





Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type

ami-0adb8ca49015e0901 (64-bit (x86)) / ami-0d4483a9dfc5a29c6 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Canonical, Ubuntu, 24.04, amd64 noble image

▼ Instance type [Info](#) | [Get advice](#)

Instance type

t2.medium

Family: t2 2 vCPU 4 GiB Memory Current generation: true

On-Demand Windows base pricing: 0.0708 USD per Hour

On-Demand Ubuntu Pro base pricing: 0.0563 USD per Hour

On-Demand Linux base pricing: 0.0528 USD per Hour

On-Demand SUSE base pricing: 0.1528 USD per Hour

On-Demand RHEL base pricing: 0.0816 USD per Hour

Additional costs apply for AMIs with pre-installed software

▼ **Network settings** [Info](#)

VPC - required | [Info](#)

vpc-07e61444329818154
172.31.0.0/16

(default) ▼



Subnet | [Info](#)

subnet-0fd062fa0f12fe9ab
VPC: vpc-07e61444329818154 Owner: 009850210027
Availability Zone: eu-west-3a (euw3-az1) Zone type: Availability Zone
IP addresses available: 4091 CIDR: 172.31.0.0/20

▼

[Create new subnet ↗](#)

Auto-assign public IP | [Info](#)

Enable ▼

Firewall (security groups) | [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

Security group name - *required*

jenkins_SG

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and . _ - : / () # , @ [] + = & ; { } ! \$ *

Description - *required* | [Info](#)

Jenkins Security group

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

Remove

Type Info	Protocol Info	Port range Info
ssh ▼	TCP	22
Source type Info	Source Info	Description - optional Info
Anywhere ▼	Q Add CIDR, prefix list or security group ID 0.0.0.0/0 X	e.g. SSH for admin desktop

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0)

Remove

Type | Info

HTTP ▼

Protocol | Info

TCP

Port range | Info

80

Source type | Info

Anywhere ▼

Source | Info

Q Add CIDR, prefix list or security group ID

0.0.0.0/0 X

Description - optional | Info

e.g. SSH for admin desktop

▼ Security group rule 3 (TCP, 8080, 0.0.0.0/0)

Remove

Type	Info	Protocol	Info	Port range	Info	
Custom TCP ▼		TCP		8080		
Source type	Info	Source	Info	Description - optional		Info
Anywhere ▼		Q Add CIDR, prefix list or security group ID		e.g. SSH for admin desktop		
		0.0.0.0/0 X				

User data - *optional*

[Info](#)

Upload a file with your user data or enter it in the field.

 **Choose file**

```
https://apt.releases.hashicorp.com $ apt-get update && apt-get install terraform &&  
| sudo tee /etc/apt/sources.list.d/hashicorp.list > /dev/null
```

```
sudo apt-get update -y  
sudo apt-get install -y terraform
```

```
# _____  
# AWS CLI v2  
# _____  
echo ">>> Installing AWS CLI..."  
cd /tmp  
curl -fsSL "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o  
"awscliv2.zip"  
unzip -o awscliv2.zip  
sudo ./aws/install --update  
rm -rf awscliv2.zip aws/
```

EC2 > Instances

EC2

Dashboard

AWS Global View

Events

Instances

Instances (1/3) Info

Last updated less than a minute ago

Connect

Instance state

Actions

Launch instances

Saved filter sets

Choose filter set

Find Instance by attribute or tag (case-sensitive)

< 1 >

	Name	Instance ID	Instance state	Instance type	Status check	Availability Zone
☒	Jenkins-Server	i-0cf98ace9b59b4fd9	Running	t2.medium	2/2 checks passed	eu-west-3a

Instance ID

 i-0cf98ace9b59b4fd9

IPv6 address

—

Public IPv4 address

 35.180.54.169 | [open address](#) 

Instance state

 **Running**

aws

Search

[Alt+S]

Europe (Paris)

Samuel A. O. (009850210027)

cba_admin

EC2

Instances

i-0cf98ace9b59b4fd9

Connect to instance

Connect

Info

Connect to an instance using the browser-based client.

Instance ID

i-0cf98ace9b59b4fd9 (Jenkins-Server)

VPC ID

vpc-07e61444329818154

Security groups

sg-05f7df93c28a2e521 (jenkins_SG)

IAM role

-

EC2 Instance Connect

SSM Session Manager

SSH client

EC2 serial console

Instance ID

i-0cf98ace9b59b4fd9 (Jenkins-Server)

Connection type

Connect using a Public IP

Connect using a public IPv4 or IPv6 address

Connect using a Private IP

Connect using a private IP address and a VPC endpoint

Public IPv4 address

35.180.54.169

CloudShell

Feedback

Console Mobile App

© 2026, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences



[Alt+S]





Europe (Paris) ▼

Samuel A. O. (009850210027) ▼

cba_admin

```
ubuntu@ip-172-31-10-183:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
8aeb1cbc661a4e18aa6938d69aafb91b
ubuntu@ip-172-31-10-183:~$
```

i-0cf98ace9b59b4fd9 (Jenkins-Server)

PublicIPs: 35.180.54.169 PrivateIPs: 172.31.10.183



 CloudShell

 Feedback

 Console Mobile App

© 2026, Amazon Web Services, Inc. or its affiliates.

[Privacy](#)

[Terms](#)

[Cookie preferences](#)

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

```
/var/lib/jenkins/secrets/initialAdminPassword
```

Please copy the password from either location and paste it below.

Administrator password

Continue

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Getting Started

✓ Folders	Formatter	Ant	Gradle	Jenkins API
✓ Timestampers	Workspace Cleanup			Folders
○ Pipeline	GitHub Branch Source	Pipeline: GitHub Groovy Libraries	Pipeline Graph View	OWASP Markup Formatter
○ Git	SSH Build Agents	Matrix Authorization Strategy	LDAP	** ASM API
○ Email Extension	Mailer	Dark Theme		** JSON Path API
				** Struts
				** Pipeline: Step API
				** commons-text API
				** Token Macro
				Build Timeout
				** h2database API
				** Credentials
				** Plain Credentials
				** Variant
				** SSH Credentials
				Credentials Binding
				** SCM API
				** Pipeline: API
				Timestampers
				** Caffeine API
				** Script Security
				** Pipeline: Supporting APIs
				** Plugin Utilities API
				** Font Awesome API
				** Bootstrap 5 API
				** Jackson Annotations 2 API
				** SnakeYAML engine API
				** Woodstox Core API
				** Jackson 3 API
				** JQuery API
				** ECharts API
				** Prism API
				** Display URL API
				** Checks API
				** JUnit
				** Matrix Project
				** Resource Dispatcher
				Workspace Cleanup
				Ant
				** Okhttp

Create First Admin User

Username

phoenixsamuel

Password

Confirm password

Full name

Akosa Samuel Onyejekwe

E-mail address

samuelsonyejekwe852@gmail.com

Instance Configuration

Jenkins URL:

`http://35.180.54.169:8080/`

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the `BITBUILDER_URL` environment variable provided to build steps.

The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

Jenkins is ready!

Your Jenkins setup is complete.

[Start using Jenkins](#)

Plugins

Search: docker pipe

Install [Dropdown] Refresh

- Updates
- Available plugins
- Installed plugins
- Advanced settings
- Download progress

Install	Name 1	Released	Health
☒	AWS Global Configuration 159.v882a_ecd03fe6 Library plugins (for use by other plugins) A Jenkins plugin to configure AWS related settings	2 mo 6 days ago	100
☒	S3 publisher 520.v4381b_80b_1969 Artifact Uploaders aws This is a plugin to upload files to Amazon S3 buckets.	9 hr 28 min ago	94
☒	Pipeline: AWS Steps 1.45 pipeline aws This plugins adds Jenkins pipeline steps to interact with the AWS API.	2 yr 1 mo ago	87
☒	GitHub Integration 0.7.3 email-ext Build Triggers GitHub Integration Plugin for Jenkins	5 mo 13 days ago	87
☒	Docker Pipeline 634.vedic7242b_eda_7 pipeline DevOps Deployment docker Build and use Docker containers from pipelines. This plugin is up for adoption! We are looking for new maintainers. Visit our Adopt a Plugin initiative for more information.	7 mo 11 days ago	76

Plugins

- Updates
- Available plugins
- Installed plugins
- Advanced settings
- Download progress**

Download progress

Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

commons-lang3 v3.x Jenkins API	✓ Success
Icons API	✓ Success
Folders	✓ Success
OWASP Markup Formatter	✓ Success
ASM API	✓ Success
JSON Path API	✓ Success
Structs	✓ Success
Pipeline: Step API	✓ Success
commons-text API	✓ Success
Token Macro	✓ Success
Build Timeout	✓ Success
bouncycastle API	✓ Success
Credentials	✓ Success
Plain Credentials	✓ Success
Variant	✓ Success
SSH Credentials	✓ Success
Credentials Binding	✓ Success
SCM API	✓ Success
Pipeline: API	✓ Success
Timestamp	✓ Success
Caffeine API	✓ Success
Script Security	✓ Success
Pipeline: Supporting APIs	✓ Success
Plugin Utilities API	✓ Success
Font Awesome API	✓ Success
Bootstrap 5 API	✓ Success
Jackson Annotations 2 API	✓ Success
SnakeYAML Engine API	✓ Success
Woodstox Core API	✓ Success
Jackson 3 API	✓ Success
jQuery3 API	✓ Success
ECharts API	✓ Success
Prism API	✓ Success
Display URL API	✓ Success
Checks API	✓ Success
JUnit	✓ Success
Matrix Project	✓ Success
Resource Disposer	✓ Success
Workspace Cleanup	✓ Success
Ant	✓ Success
OkHttp	✓ Success
JavaBeans Activation Framework (JAF) API	✓ Success
JAXB	✓ Success
JSON Api	✓ Success
Jakarta Activation API	✓ Success
SnakeYAML API	✓ Success
Jakarta XML Binding API	✓ Success
Jackson 2 API	✓ Success
Apache HttpComponents Client 4.x API	✓ Success
Pipeline: Job	✓ Success
Gradle	✓ Success

Pipeline: Milestone Step	✓ Success
Durable Task	✓ Success
Pipeline: Nodes and Processes	✓ Success
Pipeline: Build Step	✓ Success
Pipeline: SCM Step	✓ Success
Pipeline: Groovy	✓ Success
Pipeline: Groovy Libraries	✓ Success
Joda Time API	✓ Success
Pipeline: Model API	✓ Success
Pipeline: Stage Step	✓ Success
Pipeline: Declarative Extension Points API	✓ Success
Jakarta Mail API	✓ Success
Instance Identity	✓ Success
Mailer	✓ Success
Branch API	✓ Success
Pipeline: Multibranch	✓ Success
Pipeline: Stage Tags Metadata	✓ Success
Pipeline: Input Step	✓ Success
Pipeline: Basic Steps	✓ Success
Pipeline: Declarative	✓ Success
Pipeline	✓ Success
Java JSON Web Token (JWT)	✓ Success
GitHub API	✓ Success
Mina SSHD API :: Common	✓ Success
Mina SSHD API :: Core	✓ Success
Gson API	✓ Success
Git client	✓ Success
Git	✓ Success
GitHub	✓ Success
GitHub Branch Source	✓ Success
Pipeline: GitHub Groovy Libraries	✓ Success
Metrics	✓ Success
Pipeline Graph View	✓ Success
Git	✓ Success
EDDSA API	✓ Success
Trilead API	✓ Success
SSH Build Agents	✓ Success
Matrix Authorization Strategy	✓ Success
LDAP	✓ Success
jsoup API	✓ Success
Email Extension	✓ Success
Mailer	✓ Success
Theme Manager	✓ Success
Dark Theme	✓ Success
Loading plugin extensions	✓ Success
Amazon Web Services SDK 2 :: Core	✓ Success
Amazon Web Services SDK :: Minimal	✓ Success
Dev Tools Symbols API	✓ Success
Amazon Web Services SDK 2 :: EC2	✓ Success
Amazon Web Services SDK :: EC2	✓ Success
AWS Credentials	✓ Success
AWS Global Configuration	✓ Success
Netty API	✓ Success
AWS Java SDK :: HTTP Clients :: Netty Non-Blocking I/O	✓ Success
Amazon Web Services SDK 2 :: S3	✓ Success
Copy Artifact	✓ Success
S3 publisher	✓ Success
Amazon Web Services SDK :: SQS	✓ Success
Amazon Web Services SDK :: SNS	✓ Success

Amazon Web Services SDK :: Api Gateway	✓ Success
Amazon Web Services SDK :: CloudFormation	✓ Success
Amazon Web Services SDK :: Elastic Beanstalk	✓ Success
Amazon Web Services SDK :: Elastic Load Balancing V2	✓ Success
Amazon Web Services SDK :: IAM	✓ Success
Amazon Web Services SDK :: ECR	✓ Success
Amazon Web Services SDK :: CloudFront	✓ Success
Amazon Web Services SDK :: Lambda	✓ Success
Amazon Web Services SDK :: Organizations	✓ Success
Amazon Web Services SDK :: CodeDeploy	✓ Success
Pipeline: AWS Steps	✓ Success
GitHub Integration	✓ Success
Authentication Tokens API	✓ Success
Docker Commons	✓ Success
Docker Pipeline	✓ Success
Loading plugin extensions	⋮ Running
Restarting Jenkins	⋮ Pending

→ [Go back to the top page](#)

(you can start using the installed plugins right away)

→ ☒ Restart Jenkins when installation is complete and no jobs are running

REST API

Jenkins 2.555.2



AWS

Configuration for Amazon Web Services

Amazon Credentials

Region ?

Europe (Paris)

Amazon Credentials ?

IAM instance Profile/user AWS config

Save

Apply

Add AWS Credentials

Global (Jenkins, nodes, items, all child items, etc)

ID ?

access-key

Description ?

Login details to access my AWS Account

Access Key ID ?

AKIAQESY6I3VY3DHXKO3

Secret Access Key

.....

These credentials are valid and have access to 3 availability zones



AWS

Configuration for Amazon Web Services.

Amazon Credentials

Region ?

Europe (Paris)

Amazon Credentials ?

AKIAQESY6L3VY3DHXKO3 (Login details to access my AWS Account)

+ Add

Save

Apply

- GitHub Apps

OAuth Apps

Personal access tokens

Time-grained tokens
Tokens (classic)

New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

jenkins-token

What's this token for?

Expiration

No expiration

GitHub strongly recommends that you set an expiration date for your token to help keep your information secure. [Learn more](#)

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

☒ repo	Full control of private repositories
☒ repo:status	Access commit status
☒ repo_deployment	Access deployment status
☒ public_repo	Access public repositories
☒ repo:invite	Access repository invitations
☒ security_events	Read and write security events
☒ workflow	Update GitHub Action workflows
☐ write:packages	Upload packages to GitHub Package Registry
☐ read:packages	Download packages from GitHub Package Registry
☐ delete:packages	Delete packages from GitHub Package Registry
☐ admin:org	Full control of orgs and teams; read and write org projects
☐ write:org	Read and write org and team membership, read and write org projects
☒ read:org	Read org and team membership, read org projects
☐ manage_runners:org	Manage org runners and runner groups
☐ admin:public_key	Full control of user public keys
☐ write:public_key	Write user public keys
☐ read:public_key	Read user public keys
☒ admin:repo_hook	Full control of repository hooks
☒ write:repo_hook	Write repository hooks
☒ read:repo_hook	Read repository hooks
☐ admin:org hook	Full control of organization hooks
☐ gist	Create gists
☐ notifications	Access notifications
☐ user	Update ALL user data
☐ read:user	Read ALL user profile data
☐ user:email	Access user email addresses (read-only)
☐ user:follow	Follow and unfollow users
☐ delete_repo	Delete repositories
☐ write:discussion	Read and write team discussions
☐ read:discussion	Read team discussions
☐ admin:enterprise	Full control of enterprises
☐ manage_runners:enterprise	Manage enterprise runners and runner groups
☐ manage_billing:enterprise	Read and write enterprise billing data
☐ read:enterprise	Read enterprise profile data
☐ scim:enterprise	Provisioning of users and groups via SCIM
☐ audit_log	Full control of audit log
☐ read:audit_log	Read access of audit log
☐ codespace	Full control of codespaces

☐ codespace:secrets	Ability to create, read, update, and delete codespace secrets
☐ copilot	Full control of GitHub Copilot settings and seat assignments
☐ manage_billing:copilot	View and edit Copilot Business seat assignments
☐ write:network_configurations	Write org hosted compute network configurations
☐ read:network_configurations	Read org hosted compute network configurations
☐ project	Full control of projects
☐ read:project	Read access of projects
☐ admin:pgp_key	Full control of public user GPG keys
☐ write:pgp_key	Write public user GPG keys
☐ read:pgp_key	Read public user GPG keys
☐ admin:ssh_signing_key	Full control of public user SSH signing keys
☐ write:ssh_signing_key	Write public user SSH signing keys
☐ read:ssh_signing_key	Read public user SSH signing keys

Generate token

Cancel



- GitHub Apps
- OAuth Apps
- Personal access tokens ^
- Fine-grained tokens
- Tokens (classic)

Personal access tokens (classic)

Generate new token ▾

Tokens you have generated that can be used to access the [GitHub API](#).



Make sure to copy your personal access token now. You won't be able to see it again!



ghp_YU4k7XT1dnUp8uks739vwFhTf0Bgvt0yTStk

Delete

 Jenkins

Manage Jenkins

Credentials

Global

Credentials that should be available everywhere

access-key: AKIAQJ579L3VY3D1K5

Login details to access my AWS Account

REST API

Jenkins 2.555.2

Add Credentials

Select a type of credential

 Username with password

Commonly used for authentication to services like Git, APIs, or registries.

 AWS Credentials

 GitHub App

 SSH Username with private key

 Secret file

 Secret text

 X.509 Client Certificate

 Certificate

Upload a PKCS#12 or PEM encoded certificate and private key.

Next



Global

Credentials that should be available everywhere



access-key: ANAGL5Y9L3VY3LKH

Click on details to access my AWS Account

+ Add Credentials



<<<

REST API

Jenkins 2.555.2

< Add Username with password



Scope ?

Global (Jenkins, nodes, items, all child items, etc)



Username

samuelakosaonyejekwe

☐ Treat username as secret ?

Password

.....

ID ?

github token

Description ?

GitHub token for Jenkins CI/CD

Create



Global

+ Add Credentials

Credentials that should be available everywhere.



access-key AKIAQ2SY9I3VY3LHXRKQ3

Login details to access my AWS Account



github-token samuelkosaonyejekwe/*****

Git lab token for Jenkins CI/CD



Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks](#).

Required fields are marked with an asterisk ().*

Owner *

👤 samuelakosaonyejekwe

/

Repository name *

E-CommerceWebsite

✔ E-CommerceWebsite is available.

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description

A sample of E-CommerceWebsite using HTML, CSS, JavaScript and API

65 / 350 characters

☒ Copy the `main` branch only

Contribute back to SafeEHA/E-CommerceWebsite by adding your own branch. [Learn more.](#)

📌 You are creating a fork in your personal account.

Create fork

E-CommerceWebsite

Public

forked from [SafeEHA/E-CommerceWebsite](#)

📌 Pin

👁 Watch 0

🍴 Fork 0

★ Star 0

🔗 master

🔗 1 Branch

🏷 0 Tags

🔍 Go to file

➕ Add file

<> Code

This branch is up to date with [SafeEHA/E-CommerceWebsite:master](#).

👤 Contribute

🔄 Sync fork

👤 SafeEHA	Fix clothing and accessories link text	1a61513 · last month	🕒 76 Commits
📁 TerraformBackend	update bucket name on main.tf	last month	
📁 TerraformDep	changed back	last month	
📁 webapp	Fix clothing and accessories link text	last month	
📄 .gitattributes	Initial commit	7 years ago	
📄 .gitignore	jenkinsfile updated	last year	
📄 Jenkinsfile	Refactor Jenkinsfile for improved clarity and structure	last month	
📄 Jenkinsfile-destroy	Corrected Terraform and added destroy job	last year	
📄 err.md	Added S3 and DynamoDB backend provisioning	last year	

📖 README



Add a README

Help people interested in this repository understand your project.

➕ Add a README

About

A sample of E-CommerceWebsite using HTML, CSS, JavaScript and API

[shoplane-by-lassie.netlify.app/](#)

🔗 Activity

★ 0 stars

👁 0 watching

🍴 0 forks

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Contributors

No contributors

Languages

📊 HCL 32.0% 📊 HTML 25.7%

📊 CSS 22.0% 📊 JavaScript 20.1%

📊 Dockerfile 0.2%

Suggested workflows

Based on your tech stack.



SLSA Generic generator

Configure

Generate SLSA3 provenance for your existing release workflows



Deno

Configure

Test your Deno project



Node.js

Configure

Build and test a Node.js project with npm.

[More workflows](#)

[Dismiss suggestions](#)



```
akosa@AKOSA:~$ git clone https://github.com/samuelakosaonyekwe/E-CommerceWebsite.git
Cloning into 'E-CommerceWebsite'...
remote: Enumerating objects: 305, done.
remote: Counting objects: 100% (119/119), done.
remote: Compressing objects: 100% (63/63), done.
Receiving objects: 100% (305/305), 1.13 MiB | 1.95 MiB/s, done.
remote: Total 305 (delta 82), reused 56 (delta 56), pack-reused 186 (from 3)
Resolving deltas: 100% (155/155), done.
akosa@AKOSA:~$ |
```

File Edit Selection View Go Run ...

E-CommerceWebsite [WSL: Ubuntu-24.04]

EXPLORER

- E-COMMERCEWEBSITE [WSL: UBUNTU-24.04]
 - TerraformBackend
 - .terraform.lock.hcl
 - backend.tf
 - TerraformDep
 - webapp
 - .gitattributes
 - .gitignore
 - err.md
 - Jenkinsfile
 - Jenkinsfile-destroy

backend.tf

```
70 output "dynamodb_table_name" {
71     value      = aws_dynamodb_table.terraform_locks.name
72 }
73
74
75 variable "region" {
76     description = "AWS region"
77     type        = string
78     default     = "eu-west-2"
79 }
80
81 variable "bucket_name" {
82     description = "Name of the S3 bucket for Terraform state"
83     type        = string
84     default     = "ecommerce-terraform-state-009850210027"
85 }
86
87 variable "dynamodb_table_name" {
```

PROBLEMS TERMINAL ...

bash - E-CommerceWebsite

```
akosa@AKOSA:~/E-CommerceWebsite$ aws sts get-caller-identity --query Account --ou
tput text
009850210027
akosa@AKOSA:~/E-CommerceWebsite$
```

```
akosa@AKOSA:~/E-CommerceWebsite/TerraformBackend$ terraform init
Successfully configured the backend "local"! Terraform will automatically
use this backend unless the backend configuration changes.
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Installing hashicorp/aws v5.100.0...
- Installed hashicorp/aws v5.100.0 (signed by HashiCorp)
Terraform has made some changes to the provider dependency selections recorded
in the .terraform.lock.hcl file. Review those changes and commit them to your
version control system if they represent changes you intended to make.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
akosa@AKOSA:~/E-CommerceWebsite/TerraformBackend$
```

File Edit Selection View Go Run ...

E-CommerceWebsite [WSL: Ubuntu-24.04]

EXPLORER

- ✓ E-COMMERCEWEBSITE [WSL: UBUNTU-24.04]
 - ✓ TerraformBackend
 - > .terraform
 - ≡ terraform.lock.hcl M
 - backend.tf M
 - > TerraformDep
 - > webapp
 - ◆ .gitattributes
 - ◆ .gitignore
 - ✚ err.md
 - 🔒 Jenkinsfile
 - 🌿 Jenkinsfile-destroy

PROBLEMS OUTPUT TERMINAL ...

bash - TerraformBackend

```
akosa@AKOSA:~/E-CommerceWebsite/TerraformBackend$ terraform plan
+ id      = (known after apply)

+ rule {
+   apply_server_side_encryption_by_default {
+     sse_algorithm = "AES256"
+     # (1 unchanged attribute hidden)
+   }
+ }

# aws_s3_bucket_versioning.terraform_state_versioning will be created
+ resource "aws_s3_bucket_versioning" "terraform_state_versioning" {
+   bucket = (known after apply)
+   id      = (known after apply)

+   versioning_configuration {
+     mfa_delete = (known after apply)
+     status     = "Enabled"
+   }
+ }

Plan: 5 to add, 0 to change, 0 to destroy.

Changes to Outputs:
+ dynamodb_table_name = "ecommerce-terraform-locks"
+ s3_bucket_name       = "ecommerce-terraform-state-009850210027"
```

File Edit Selection View Go Run ...

EXPLORER

E-COMMERCEWEBSITE [WSL: UBUNTU-24.04]

TerraformBackend

.terraform

.terraform.lock.hcl

backend.tf

terraform.tfstate

TerraformDep

webapp

.gitattributes

.gitignore

err.md

Jenkinsfile

Jenkinsfile-destroy

OUTLINE

TIMELINE

PROBLEMS OUTPUT TERMINAL

bash - TerraformBackend

akosa@AKOSA:~/E-CommerceWebsite/TerraformBackend\$ terraform apply -auto-approve

}
}

Plan: 5 to add, 0 to change, 0 to destroy.

Changes to Outputs:

+ dynamodb_table_name = "ecommerce-terraform-locks"
+ s3_bucket_name = "ecommerce-terraform-state-009850210027"

aws_s3_bucket.terraform_state: Creating...
aws_dynamodb_table.terraform_locks: Creating...
aws_s3_bucket.terraform_state: Creation complete after 4s [id=ecommerce-terraform-state-009850210027]
aws_s3_bucket_public_access_block.terraform_state_access: Creating...
aws_s3_bucket_server_side_encryption_configuration.terraform_state_encryption: Creating...
aws_s3_bucket_versioning.terraform_state_versioning: Creating...
aws_s3_bucket_public_access_block.terraform_state_access: Creation complete after 0s [id=ecommerce-terraform-state-009850210027]
aws_s3_bucket_server_side_encryption_configuration.terraform_state_encryption: Creation complete after 0s [id=ecommerce-terraform-state-009850210027]
aws_s3_bucket_versioning.terraform_state_versioning: Creation complete after 1s [id=ecommerce-terraform-state-009850210027]
aws_dynamodb_table.terraform_locks: Creation complete after 10s [id=ecommerce-terraform-locks]

Apply complete! Resources: 5 added, 0 changed, 0 destroyed.

Outputs:

dynamodb_table_name = "ecommerce-terraform-locks"
s3_bucket_name = "ecommerce-terraform-state-009850210027"

akosa@AKOSA:~/E-CommerceWebsite/TerraformBackend\$

WSL: Ubuntu-24.04

master*

0 0 0

akinwunmi-akinimisi (11 months ago)

Ln 83, Col

File Edit Selection View Go Run ...

EXPLOLER

E-COMMERCEWEBSITE [WSL: UBUNTU-24.04]

TerraformBackend

TerraformDep

.terraform.lock.hcl

main.tf M

variable.tf

webapp

.gitattributes

.gitignore

err.md

Jenkinsfile

Jenkinsfile-destroy

OUTLINE

TIMELINE

main.tf M X

TerraformDep > main.tf

```
2 terraform {
3   required_providers {
14     version = "~> 0.9
15   }
16 }
17 backend "s3" {
18   bucket      = "ecommerce-terraform-state-009850210027"
19   key         = "terraform.tfstate"
20   region      = "eu-west-2"

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

bash - TerraformBackend

akosa@AKOSA:~/E-CommerceWebsite/TerraformBackend\$ git add . git commit -m "update bucket name on main.tf" git push

-u, --[no-]update update tracked files

--[no-]renormalize renormalize EOL of tracked files (implies -u)

-N, --[no-]intent-to-add record only the fact that the path will be added later

-A, --[no-]all add changes from all tracked and untracked files

--[no-]ignore-removal ignore paths removed in the working tree (same as --no-all)

--[no-]refresh don't add, only refresh the index

--[no-]ignore-errors just skip files which cannot be added because of errors

--[no-]ignore-missing check if - even missing - files are ignored in dry run

--[no-]sparse allow updating entries outside of the sparse-checkout cone

--[no-]chmod (+|-)x override the executable bit of the listed files

--[no-]pathspec-from-file <file> read pathspec from file

--[no-]pathspec-file-nul with --pathspec-from-file, pathspec elements are separated with NUL character

akosa@AKOSA:~/E-CommerceWebsite/TerraformBackend\$

WSL: Ubuntu-24.04

master*

0 0 0

safeeha (1 month ago)

Ln 22, Col 44

Spaces: 4

UTF-8

New Item

Enter an item name

E-Commerce CI/CD Pipeline

Select an item type



Pipeline

Build, test, and deploy using pipelines. Supports stages, parallel work, and running on multiple agents.



Freestyle project

Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.



Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.



Folder

Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.



Multibranch Pipeline

Creates a set of Pipeline projects according to detected branches in one SCM repository.



Organization Folder

Creates a set of multibranch project subfolders by scanning for repositories.

OK

Configure

⚙️ General

🕒 Triggers

🔧 Pipeline

🔑 Advanced

General

Enabled 

Description

Plain text [Preview](#)

- ☐ Discard old builds ?
- ☐ Do not allow concurrent builds
- ☐ Do not allow the pipeline to resume if the controller restarts
- ☐ GitHub project
- ☐ Permission to Copy Artifact
- ☐ Pipeline speed/durability override ?
- ☐ Preserve stashes from completed builds ?
- ☐ This project is parameterized ?
- ☐ Throttle builds ?

Triggers

Set up automated actions that start your build based on specific events, like code changes or scheduled times.

- ☐ Build after other projects are built ?
- ☐ Build periodically ?
- ☐ GitHub Branches
- ☐ GitHub Pull Requests ?
- ☐ GitHub hook trigger for GITScm polling ?
- ☐ Poll SCM ?
- ☐ Trigger builds remotely (e.g., from scripts) ?

Pipeline

Define your Pipeline using Groovy directly or pull it from source control.

Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

<https://github.com/samuelakosaonyekwe/E-CommerceWebsite.git>

Credentials ?

samuelakosaonyekwe/***** (GitHub token for Jenkins CI/CD)

+ Add

Advanced ▾

+ Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

*/master

+ Add Branch

Repository browser ?

(Auto)

Additional Behaviours

+ Add

Script Path ?

Jenkinsfile

☒ Lightweight checkout ?

[Pipeline Syntax](#)

Advanced

Advanced ▾

Save

Apply

-  Status
-  Changes
-  Build with Parameters
-  Configure
-  Delete Pipeline
-  Stages
-  Rename
-  Pipeline Syntax

Builds 

 Filter



Today

Pipeline E-Commerce CICD Pipeline

This build requires parameters:

AWS_ACCOUNT_ID

AWS Account ID

009850210027

AWS_REGION

AWS Region

eu-west-3

IMAGE_TAG

Docker image tag (leave blank to use build number and commit hash)

 Build

Cancel

```
0m0s1m0s32m
Destroy complete! Resources: 0 destroyed.0m
[Pipeline] }
[Pipeline] // dir
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
ERROR: script returned exit code 1
Finished: FAILURE
```

Status

Changes

Build with Parameters

Configure

Delete Pipeline

Stages

Rename

Pipeline Syntax

✖ E-Commerce CICD Pipeline

Add description

Permalinks

- [Last build \(#1\), 3 min 58 sec ago](#)
- [Last failed build \(#1\), 3 min 58 sec ago](#)
- [Last unsuccessful build \(#1\), 3 min 58 sec ago](#)
- [Last completed build \(#1\), 3 min 58 sec ago](#)

Builds >

...

Filter

/

Today



#1 2:45 PM





Status



Changes



Build with Parameters



Configure



Delete Pipeline



Stages



Rename



Pipeline Syntax



E-Commerce CICD Pipeline



Add description

Permalinks

- [Last build \(#8\), 12 min ago](#)
- [Last stable build \(#8\), 12 min ago](#)
- [Last successful build \(#8\), 12 min ago](#)
- [Last failed build \(#2\), 14 hr ago](#)
- [Last unsuccessful build \(#2\), 14 hr ago](#)
- [Last completed build \(#8\), 12 min ago](#)

EC2 > Load balancers

Network & Security

Load Balancing

Auto Scaling

Load balancers (1/1) [What's new?](#)

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Filter load balancers

☒	Name	Type	Scheme	IP addr...	VPC ID	Availability Zones
☒	a333ad48ae8c94ca99fb...	classic	-	-	vpc-05c50ea1c4a62fb04	2 Availability Zone

Load balancer: a333ad48ae8c94ca99fbf5eec01ad812

DNS name [Info](#)

[a333ad48ae8c94ca99fbf5eec01ad812-323448702.eu-west-3.elb.amazonaws.com](#) (A Record)

cha_admin

Configure

-  General
-  Triggers
-  Pipeline
-  Advanced

Triggers

Set up automated actions that start your build based on specific events, like code changes or scheduled times.

- ☐ Build after other projects are built ?
- ☐ Build periodically ?
- ☐ GitHub Branches
- ☐ GitHub Pull Requests ?
- ☒ GitHub hook trigger for GITScm polling ?
- ☐ Poll SCM ?
- ☐ Trigger builds remotely (e.g., from scripts) ?

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Models

Preview

Webhooks

Copilot

Environments

Codespaces

Pages

Security and quality

Advanced Security

Deploy keys

Secrets and variables

Integrations

GitHub Apps

Email notifications

Webhooks / Add webhook

We'll send a `POST` request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, `x-www-form-urlencoded`, etc). More information can be found in [our developer documentation](#).

Payload URL *

http://35.180.54.169:8080/github-webhook/

Content type *

application/json

Secret

SSL verification

By default, we verify SSL certificates when delivering payloads.

☒ Enable SSL verification ☐ Disable (not recommended)

Which events would you like to trigger this webhook?

☒ Just the push event.

☐ Send me everything.

☐ Let me select individual events.

☒ Active

We will deliver event details when this hook is triggered.

Add webhook



samuelakosaonyejekwe / E-CommerceWebsite

Q Type / to search

+

<> Code

Pull requests

Agents

Actions

Projects

Wiki

Security and quality

Insights

Settings

Okay, that hook was successfully created. We sent a ping payload to test it out! Read more about it at <https://docs.github.com/webhooks/#ping-event>.

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Models

Webhooks

Add webhook

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

✓ [http://35.180.54.169:8080/github-w...](#) (push)

Last delivery was successful.

EditDelete

samuelakosaonyejekwe / E-CommerceWebsite

🔍 Type / to search

<> Code

🔗 Pull requests

👤 Agents

🕒 Actions

📁 Projects

📖 Wiki

🛡️ Security and quality

📈 Insights

⚙️ Settings

⚙️ General

Access

👤 Collaborators

🗨️ Moderation options

Code and automation

🌿 Branches

🏷️ Tags

📜 Rules

🕒 Actions

🔗 Models

🔗 Webhooks

Webhooks / Manage webhook

Settings

Recent Deliveries

✓

🔑

0e9310f4-573c-11f1-906f-97c026512a2c

ping

2026-05-24 09:44:48

...

FileEditSelectionViewGoRun...E-CommerceWebsite [WSL: Ubuntu-24.04]

EXPLORER

E-COMMERCEWEBSITE [WSL: UBUNTU-24.04]

TerraformBackendTerraformDep.terraform.lock.hclmain.tfvariable.tfwebapp

src

cssimgjs

cart.htmlcart.jscontent.htmlcontent.jscontentDetails.htmlcontentDetails.js

tf

terraform.tfstatebackend.tfhheader.html M Xvariable.tfJenkinsfile-destroyJenkinsf

webapp > src > header.html > html > body > header > section > div#container > div#search > input#input

2<html lang="en">15<body>1617<header>18<section>19<!-- MAIN CONTAINER -->20<div id="container">21<!-- SHOP NAME -->22<div id="shopName"> SHOPLANE </div>23<!-- COLLECTIONS ON WEBSITE -->24<div id="collection">25<div id="clothing"> CLOTHING </div>26<div id="accessories"> ACCESSORIES </div>27</div>28<!-- SEARCH SECTION -->29<div id="search">30<i class="fas fa-search search"></i>31<input type="text" id="input" name="searchBox" placeholder="Search for Clothing and Accessories">32</div>

```
akosa@AKOSA:~/E-CommerceWebsite$ cd ~/E-CommerceWebsite && git add . && git commit -m "Updated frontend and Terraform configuration"
&& git push origin master
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

nothing to commit, working tree clean
akosa@AKOSA:~/E-CommerceWebsite$ cd ~/E-CommerceWebsite && git push origin master
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 486 bytes | 243.00 KiB/s, done.
Total 5 (delta 3), reused 1 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/samuelakosaonyekwe/E-CommerceWebsite.git
   ed0ed36..c12155d  master -> master
akosa@AKOSA:~/E-CommerceWebsite$ |
```

- Status
- </> Changes
- ▶ Build with Parameters
- ⚙️ Configure
- 🗑️ Delete Pipeline
- 📖 Stages
- ✎️ Rename
- ❓ Pipeline Syntax
- 📄 GitHub Hook Log

✓ E-Commerce CICD Pipeline

✎️ Add description

Permalinks

- [Last build \(#10\), 0.12 sec ago](#)
- [Last stable build \(#9\), 2 min 31 sec ago](#)
- [Last successful build \(#9\), 2 min 31 sec ago](#)
- [Last failed build \(#2\), 16 hr ago](#)
- [Last unsuccessful build \(#2\), 16 hr ago](#)
- [Last completed build \(#9\), 2 min 31 sec ago](#)



Clothing For Men And Women



Men Navy Blue Solid
Sweatshirt

United Colors Of
Benetton

Rs 2599



Men Black MAMGP T7
Sweat Sporty Jacket

Puma

Rs 7999



Men Black Action
Parkview Lifestyle Shoes

Hush Puppies

Rs 6999



Women Black Solid
Lightweight Leather
Jacket

BARESKIN

Rs 9999



Women Blue Solid Shirt
Dress

SASSAFRAS

Rs 5200

Accessories For Men And Women



Unisex Silver-Toned Series
3 Smart Watch

Apple

Rs 31999



Unisex Black & Green
Reflex 2.0 Smart Band

Fastrack

Rs 1999



Unisex Black Galaxy Fit
Fitness Band

Samsung

Rs 9990



Gear IconX Black Cord-
Free Fitness Earbuds

Samsung

Rs 13990



White 2nd Gen AirPods
With Charging Case

Apple

Rs 14999

ONLINE STORE

MEN CLOTHING
WOMEN CLOTHING
MEN ACCESSORIES
WOMEN ACCESSORIES

HELPFUL LINK

HOME
ABOUT
CONTACT

PARTNERS

ZARA
PANTALOONS
LEVIS
UCB
+ MANY MORE

ADDRESS

BUILDING 101
CENTRAL AVENUE
LA - 90272
UNITED STATES



Today		
✓	#10	7:04 AM
✓	#9	7:01 AM
✓	#8	5:20 AM
May 23, 2026		
✗	#2	2:50 PM
✗	#1	2:45 PM

CLUSTER_NAME

EKS Cluster Name

TEST_PORT

Host port for testing Docker image (use 0 for random port)

☒ DESTROY

Check to destroy resources instead of deploying

▶ Build

Cancel



Jenkins

/ E-Commerce CI/CD Pipeline



#11



```
[WS-CLEANUP] Deleting project workspace...  
[WS-CLEANUP] Deferred wipeout is used...  
[WS-CLEANUP] done  
[Pipeline] }  
[Pipeline] // script  
[Pipeline] echo  
Terraform destroy completed successfully.  
[Pipeline] }  
[Pipeline] // stage  
[Pipeline] }  
[Pipeline] // withEnv  
[Pipeline] }  
[Pipeline] // node  
[Pipeline] End of Pipeline  
Finished: SUCCESS
```