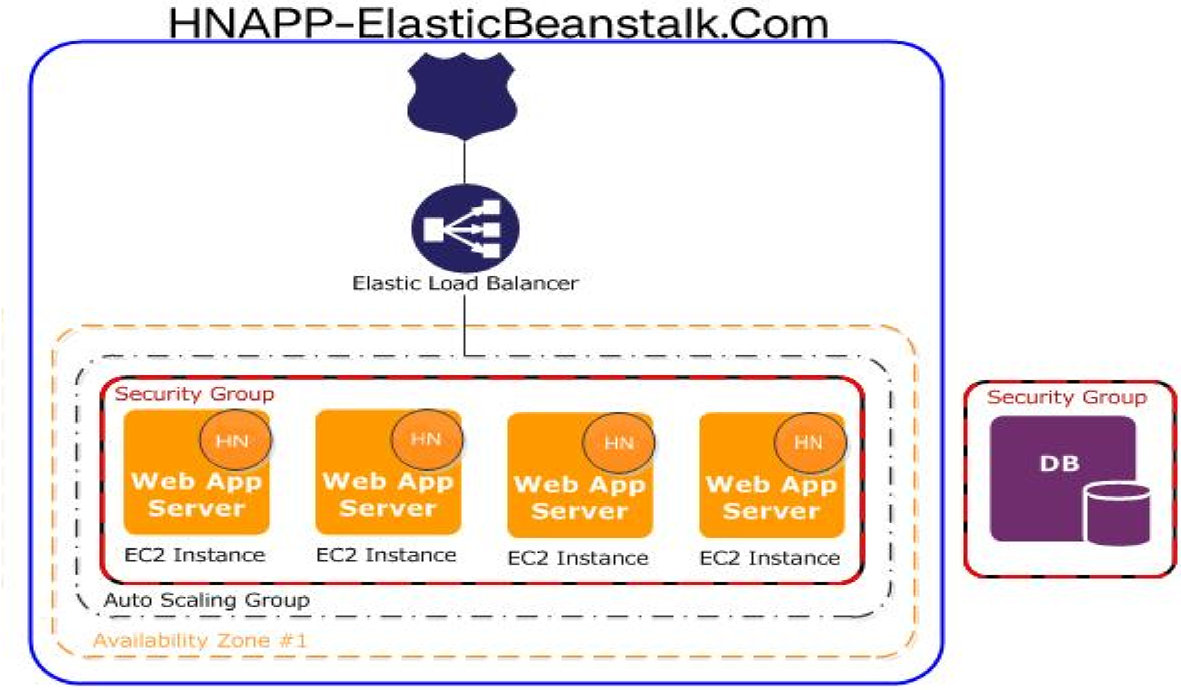
Project



Deploy a Web Application Using Elastic Beanstalk

07.11.2023

—

Himanshu Nimje
Nagpur-440017

Overview

AWS Elastic Beanstalk quickly deploys and manages applications in the AWS Cloud without worrying about the infrastructure that runs those applications. AWS Elastic Beanstalk reduces management complexity without restricting choice or control. You simply upload your application, and AWS Elastic Beanstalk automatically handles the details of capacity provisioning, load balancing, scaling, and application health monitoring.

Benefits

1. Fast and simple:- Developers can deploy applications in minutes without managing infrastructure or resource configuration.
2. Scalable:- Applications can be automatically scaled up and down based on needs.
3. Reduces complexity:- Elastic Beanstalk manages application versioning, deployment, auto scaling, load balancing, and health monitoring.
4. Focus on code:- Developers can focus on writing code instead of managing servers, databases, load balancers, firewalls, and networks.

Service Used

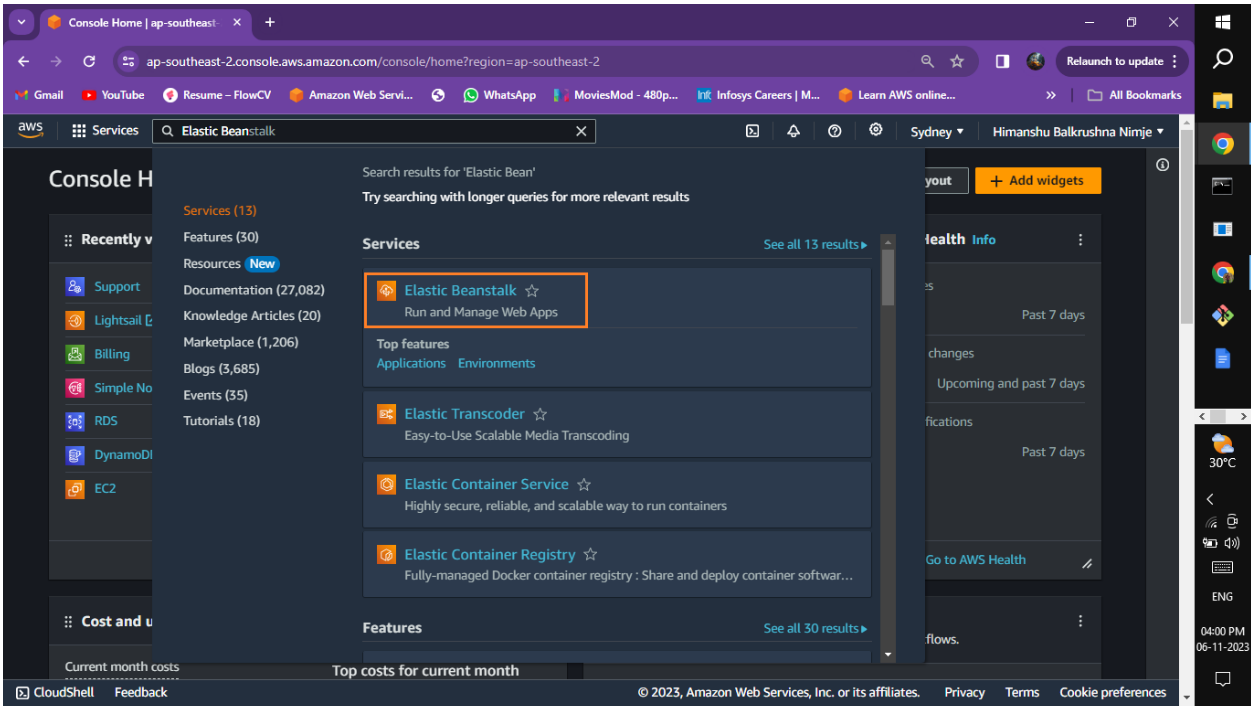
AWS Elastic Beanstalk

Steps

1. Prepare Your Application Code
2. Create an Elastic Beanstalk Application
3. Create an Environment
4. Review and Launch
5. Choose Web Server Environment
6. Configure Your Environment

Implementation⇩

Go to Elastic Beanstalk



Search results for 'Elastic Bean'

Try searching with longer queries for more relevant results

Elastic Beanstalk — Run and Manage Web Apps

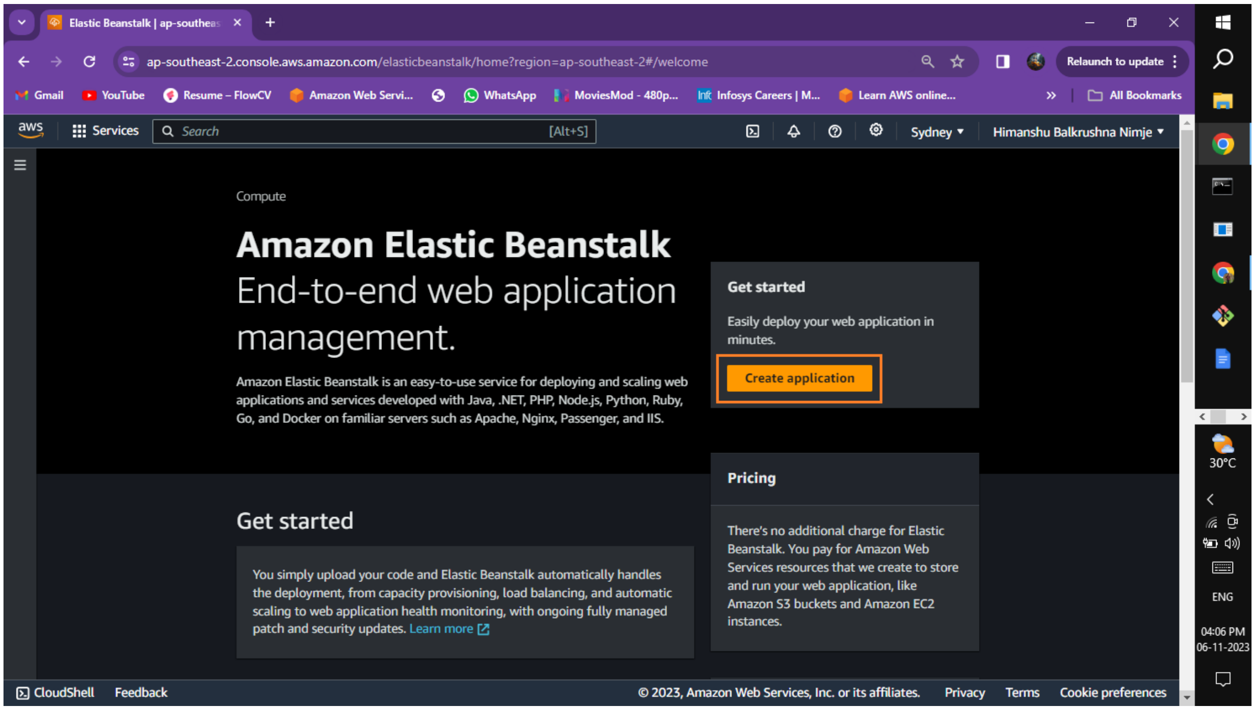
Elastic Transcoder — Easy-to-Use Scalable Media Transcoding

Elastic Container Service — Highly secure, reliable, and scalable way to run containers

Elastic Container Registry — Fully-managed Docker container registry : Share and deploy container softwar...

© 2023, Amazon Web Services, Inc. or its affiliates.

And Then click on



Amazon Elastic Beanstalk

End-to-end web application management.

Amazon Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.

Get started

Easily deploy your web application in minutes.

Pricing

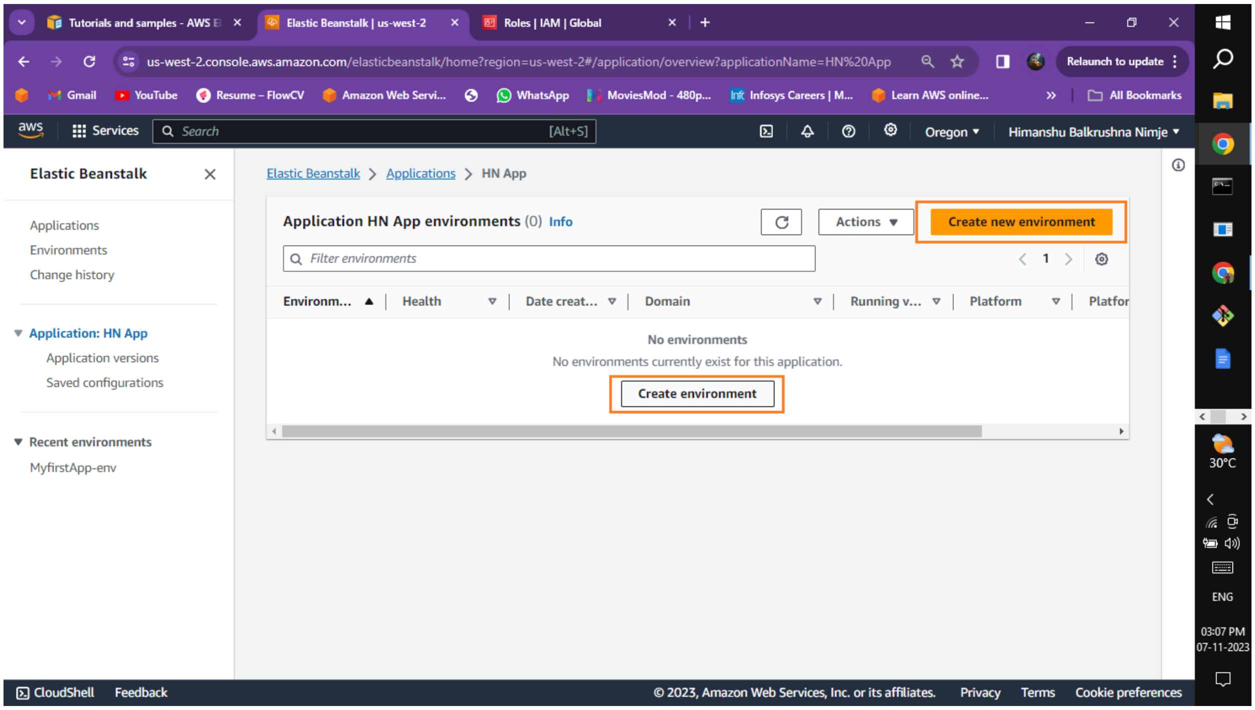
There's no additional charge for Elastic Beanstalk. You pay for Amazon Web Services resources that we create to store and run your web application, like Amazon S3 buckets and Amazon EC2 instances.

Get started

You simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, and automatic scaling to web application health monitoring, with ongoing fully managed patch and security updates. Learn more

© 2023, Amazon Web Services, Inc. or its affiliates.

Click on create environment



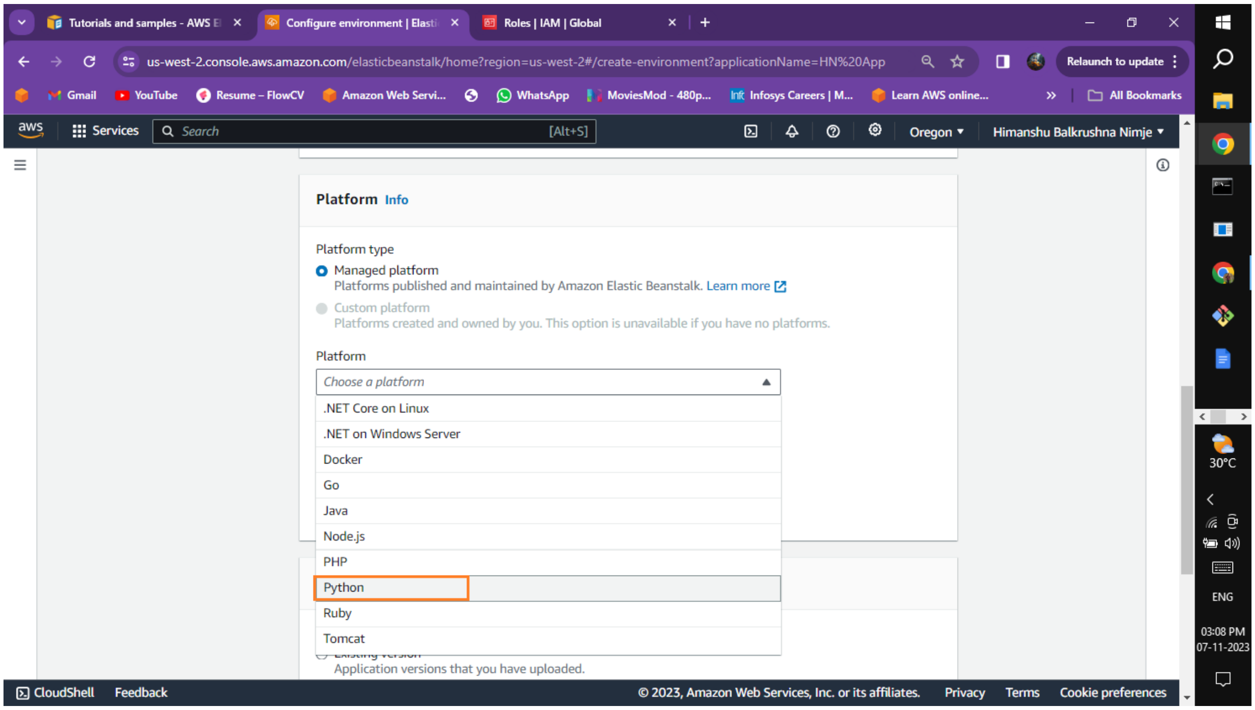
Application HN App environments (0) Info

No environments

No environments currently exist for this application.

© 2023, Amazon Web Services, Inc. or its affiliates.

Select a platform, select Python



Platform Info

Platform type

● Managed platform
Platforms published and maintained by Amazon Elastic Beanstalk. Learn more

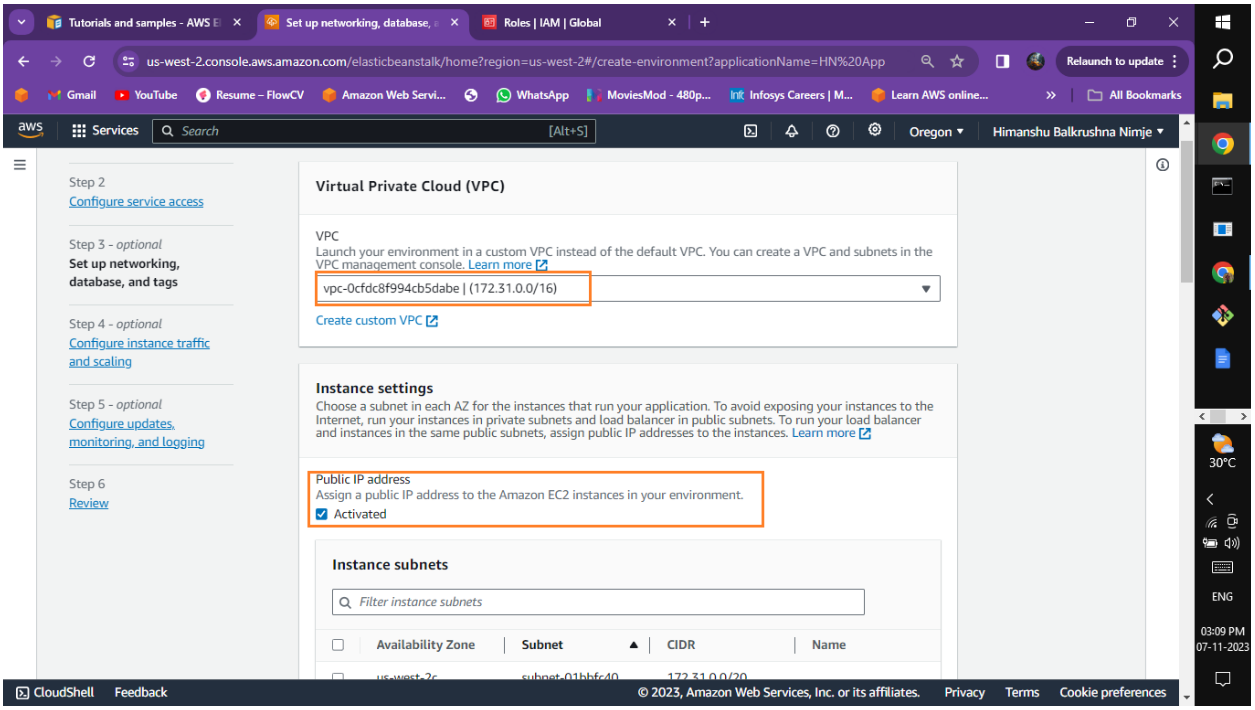
○ Custom platform
Platforms created and owned by you. This option is unavailable if you have no platforms.

Platform

Existing version
Application versions that you have uploaded.

© 2023, Amazon Web Services, Inc. or its affiliates.

Select VPC, Click on Public IP address



Step 2
Configure service access

Step 3 - *optional*
Set up networking, database, and tags

Step 4 - *optional*
Configure instance traffic and scaling

Step 5 - *optional*
Configure updates, monitoring, and logging

Step 6
Review

Virtual Private Cloud (VPC)

VPC
Launch your environment in a custom VPC instead of the default VPC. You can create a VPC and subnets in the VPC management console. Learn more

vpc-0cfdc8f994cb5dabe | (172.31.0.0/16) ▾

Create custom VPC

Instance settings
Choose a subnet in each AZ for the instances that run your application. To avoid exposing your instances to the Internet, run your instances in private subnets and load balancer in public subnets. To run your load balancer and instances in the same public subnets, assign public IP addresses to the instances. Learn more

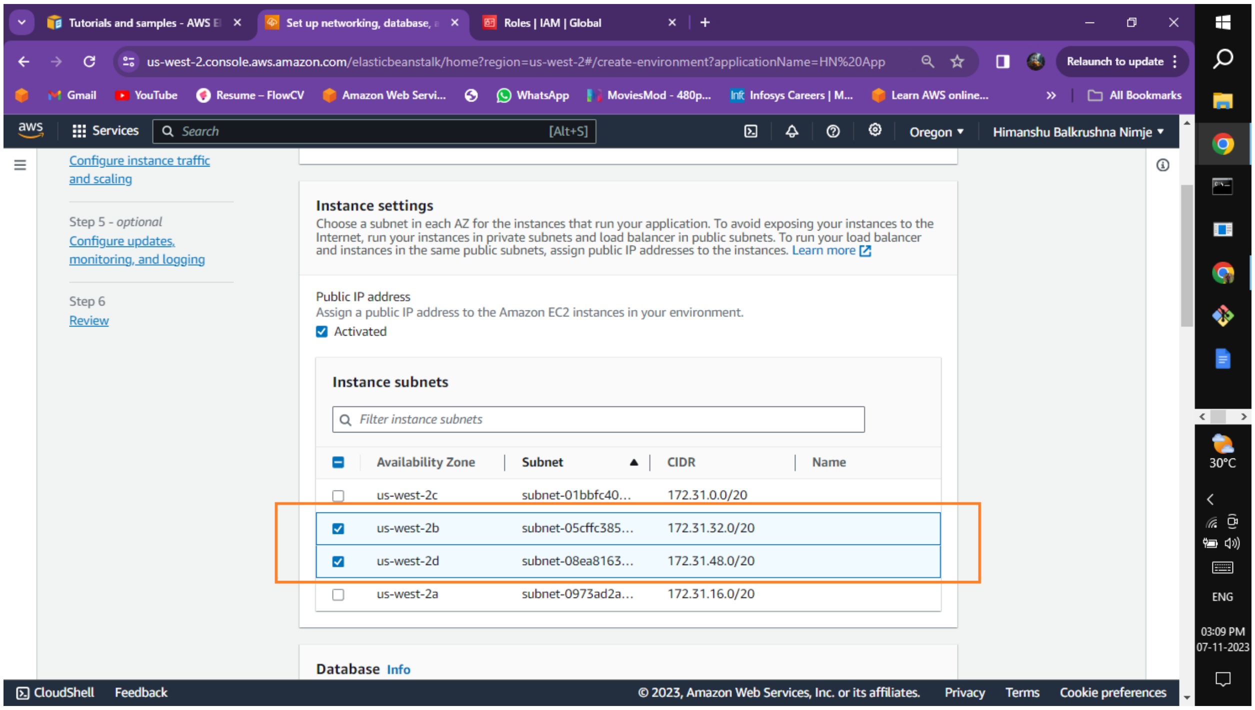
Public IP address
Assign a public IP address to the Amazon EC2 instances in your environment.
☑ Activated

Instance subnets

Filter instance subnets

□	Availability Zone	Subnet ▲	CIDR	Name
□	us-west-2c	subnet-01bbfc40	172.31.0.0/20	

Select subnet



Configure instance traffic and scaling

Step 5 - *optional*
Configure updates, monitoring, and logging

Step 6
Review

Instance settings
Choose a subnet in each AZ for the instances that run your application. To avoid exposing your instances to the Internet, run your instances in private subnets and load balancer in public subnets. To run your load balancer and instances in the same public subnets, assign public IP addresses to the instances. Learn more

Public IP address
Assign a public IP address to the Amazon EC2 instances in your environment.
☑ Activated

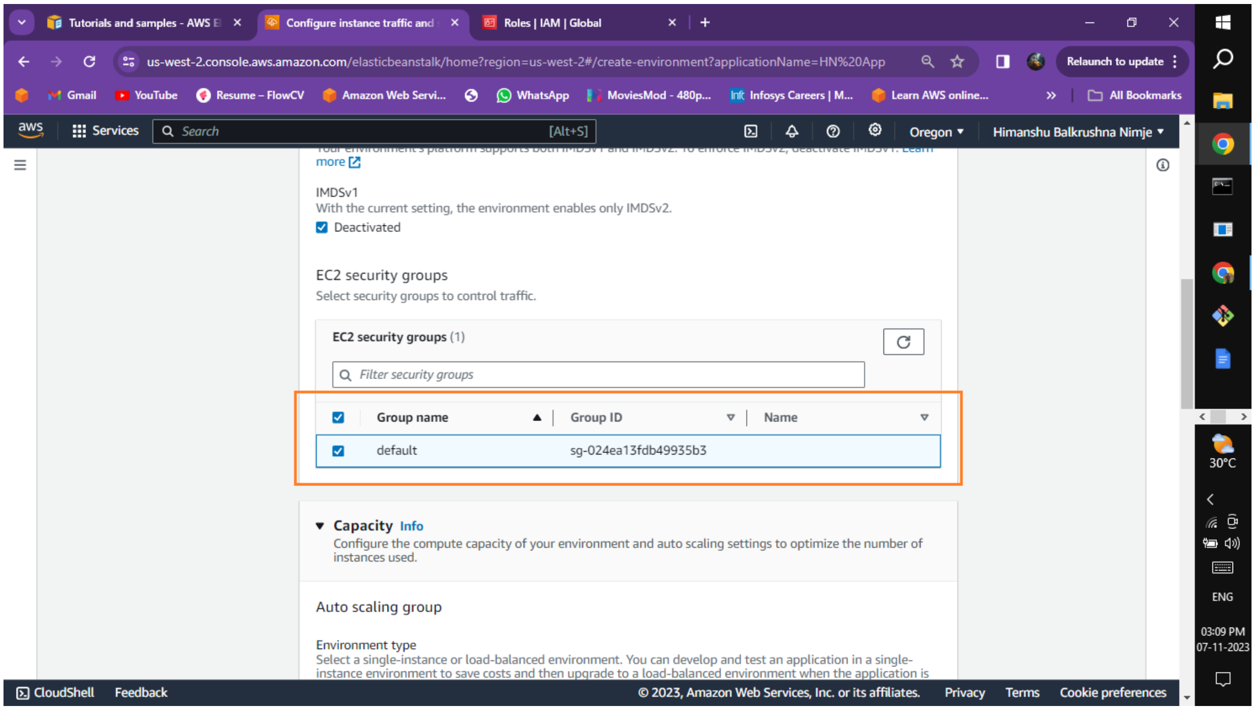
Instance subnets

Filter instance subnets

☐	Availability Zone	Subnet ▲	CIDR	Name
□	us-west-2c	subnet-01bbfc40...	172.31.0.0/20	
☑	us-west-2b	subnet-05cffc385...	172.31.32.0/20	
☑	us-west-2d	subnet-08ea8163...	172.31.48.0/20	
□	us-west-2a	subnet-0973ad2a...	172.31.16.0/20	

Database Info

Select Default Security Group



more

IMDSv1
With the current setting, the environment enables only IMDSv2.
☑ Deactivated

EC2 security groups
Select security groups to control traffic.

EC2 security groups (1)

Filter security groups

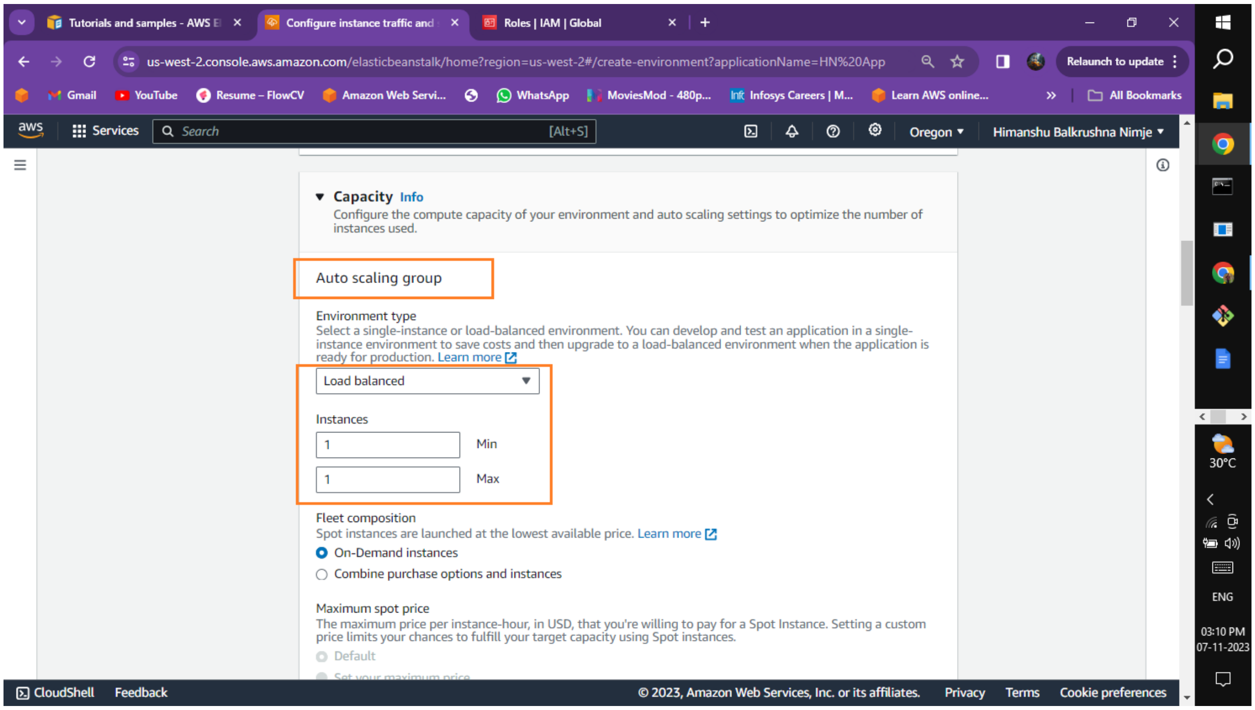
☑	Group name ▲	Group ID ▽	Name ▽
☑	default	sg-024ea13fdb49935b3	

▼ **Capacity** Info
Configure the compute capacity of your environment and auto scaling settings to optimize the number of instances used.

Auto scaling group

Environment type
Select a single-instance or load-balanced environment. You can develop and test an application in a single-instance environment to save costs and then upgrade to a load-balanced environment when the application is

In Auto Scaling group , Select Application Load Balancer then Select 1 minimum instance and 1 maximum instance for Handling the Load



▼ **Capacity** Info
Configure the compute capacity of your environment and auto scaling settings to optimize the number of instances used.

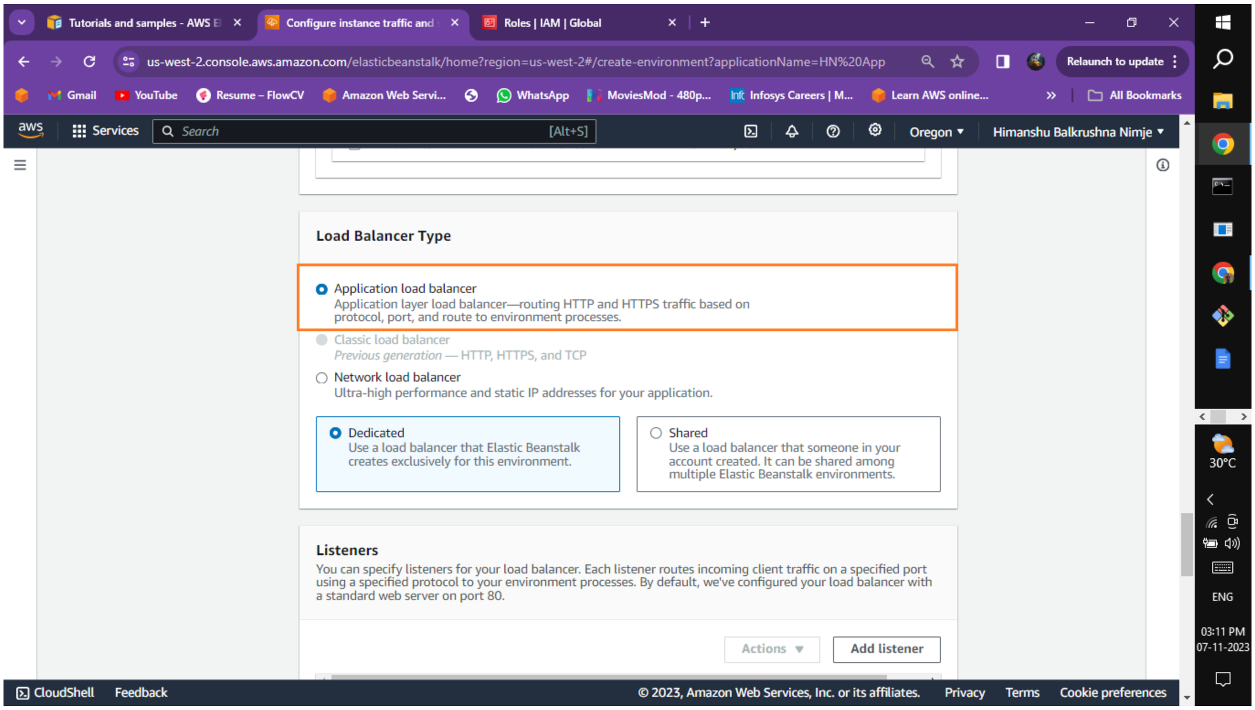
Auto scaling group

Environment type
Select a single-instance or load-balanced environment. You can develop and test an application in a single-instance environment to save costs and then upgrade to a load-balanced environment when the application is ready for production. Learn more
Load balanced ▾

Instances
1 Min
1 Max

Fleet composition
Spot instances are launched at the lowest available price. Learn more
● On-Demand instances
○ Combine purchase options and instances

Maximum spot price
The maximum price per instance-hour, in USD, that you're willing to pay for a Spot Instance. Setting a custom price limits your chances to fulfill your target capacity using Spot instances.
○ Default
○ Set your maximum price



Load Balancer Type

● Application load balancer
Application layer load balancer—routing HTTP and HTTPS traffic based on protocol, port, and route to environment processes.

○ Classic load balancer
Previous generation — HTTP, HTTPS, and TCP

○ Network load balancer
Ultra-high performance and static IP addresses for your application.

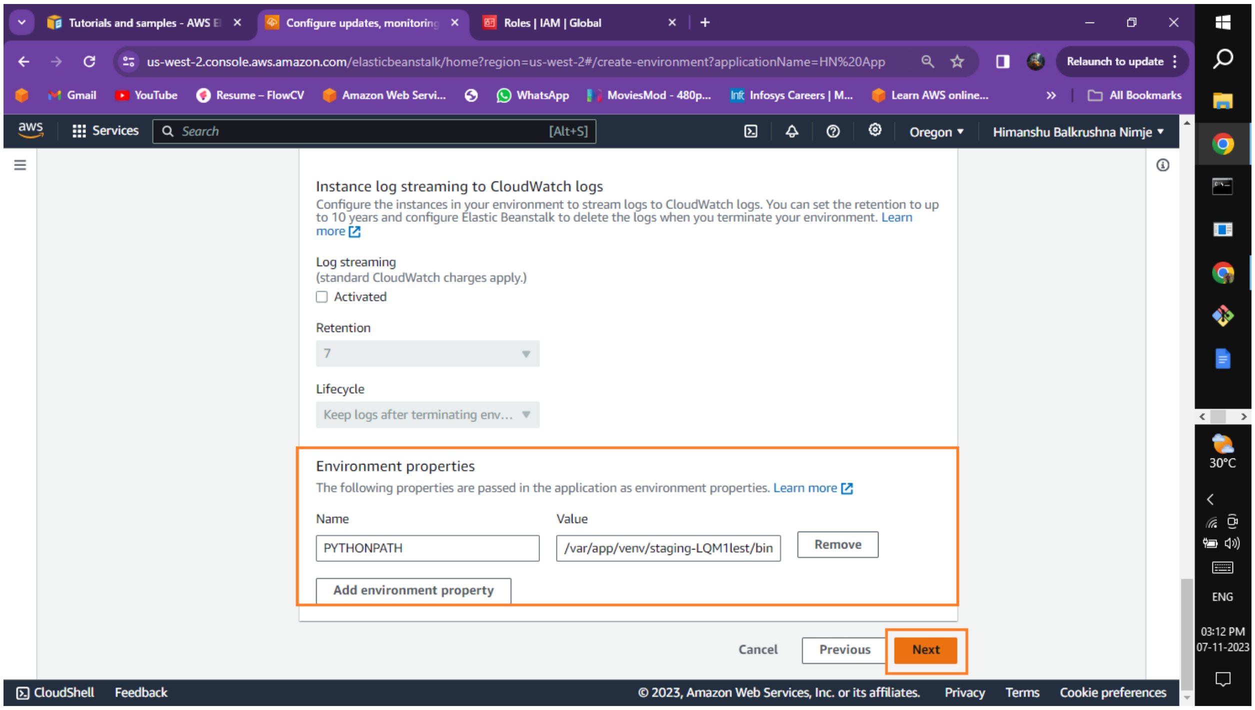
● Dedicated
Use a load balancer that Elastic Beanstalk creates exclusively for this environment.

○ Shared
Use a load balancer that someone in your account created. It can be shared among multiple Elastic Beanstalk environments.

Listeners
You can specify listeners for your load balancer. Each listener routes incoming client traffic on a specified port using a specified protocol to your environment processes. By default, we've configured your load balancer with a standard web server on port 80.

Actions
Add listener
us-west-2.console.aws.amazon.com/elasticbeanstalk/home?region=us-west-2#/create-environment?applicationName=HN%20App
Oregon
Himanshu Balkrushna Nimje
© 2023, Amazon Web Services, Inc. or its affiliates.
03:11 PM
07-11-2023

Click on the Next



Instance log streaming to CloudWatch logs
Configure the instances in your environment to stream logs to CloudWatch logs. You can set the retention to up to 10 years and configure Elastic Beanstalk to delete the logs when you terminate your environment. Learn more

Log streaming
(standard CloudWatch charges apply.)
□ Activated

Retention: 7

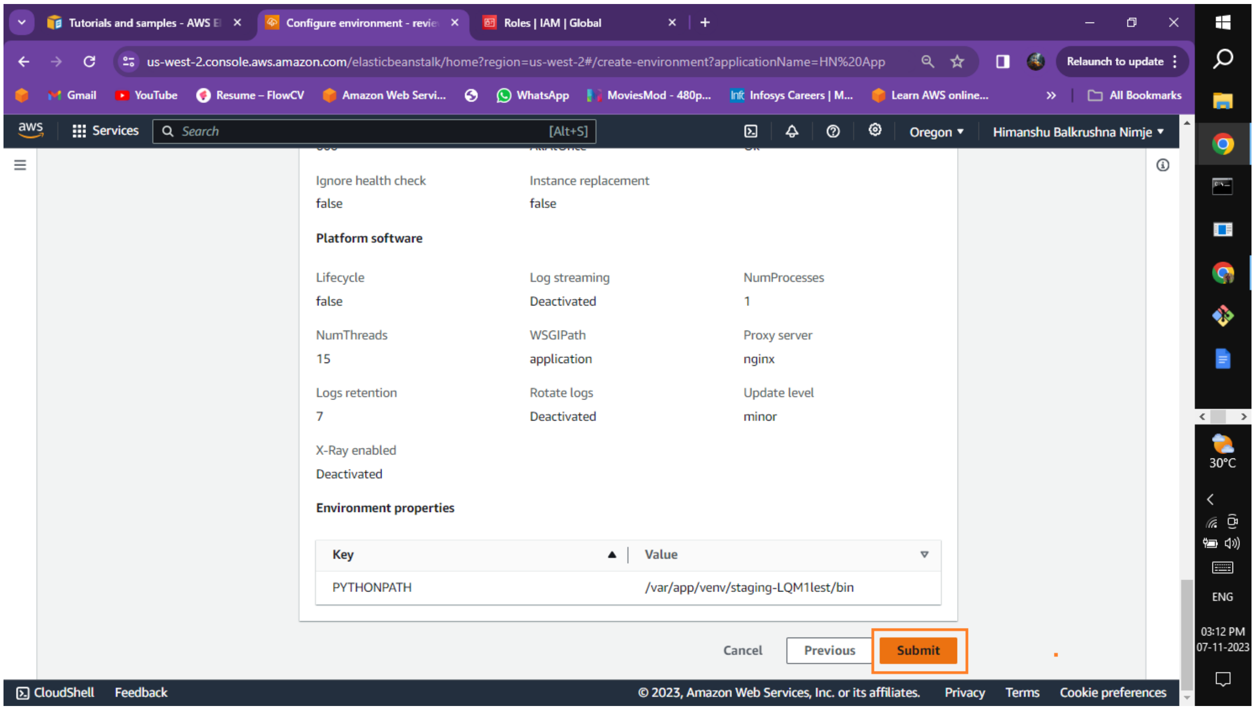
Lifecycle: Keep logs after terminating env…

Environment properties
The following properties are passed in the application as environment properties. Learn more

Name	Value
PYTHONPATH	/var/app/venv/staging-LQM1lest/bin

Remove
Add environment property
Cancel
Previous
Next
us-west-2.console.aws.amazon.com/elasticbeanstalk/home?region=us-west-2#/create-environment?applicationName=HN%20App
Oregon
Himanshu Balkrushna Nimje
© 2023, Amazon Web Services, Inc. or its affiliates.
03:12 PM
07-11-2023

Click on Submit



Ignore health check false	Instance replacement false	

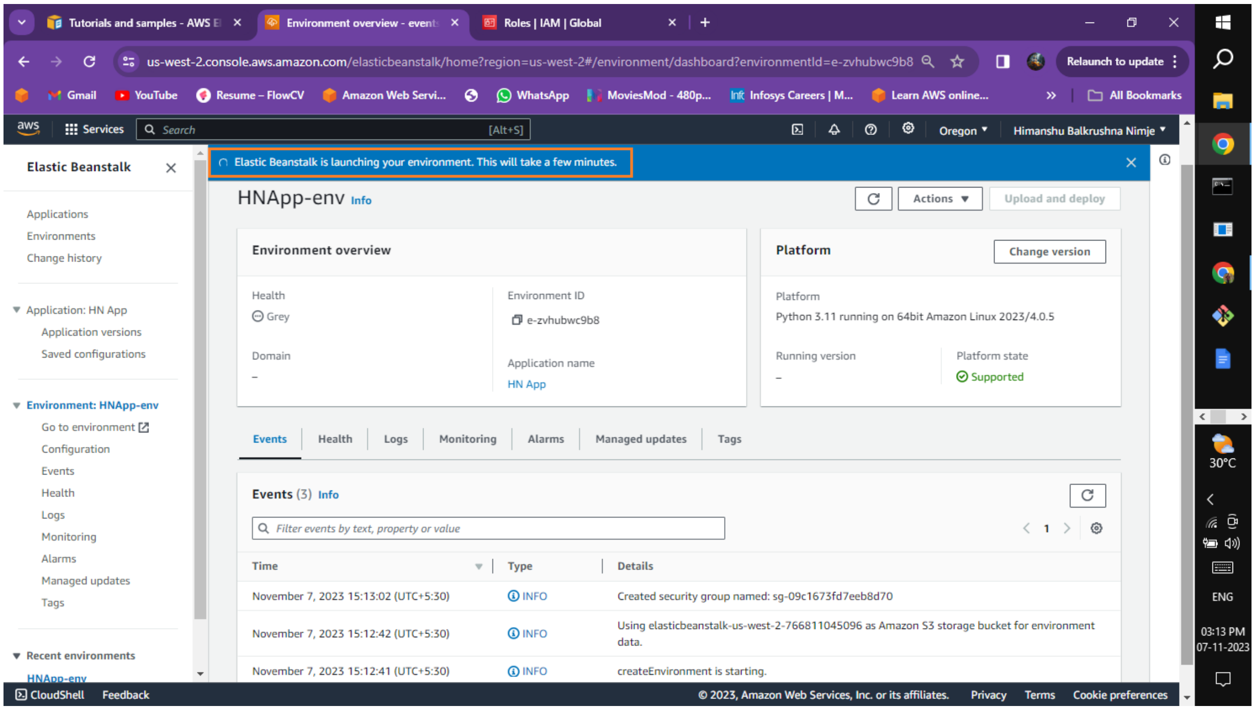
Platform software

Lifecycle false	Log streaming Deactivated	NumProcesses 1
NumThreads 15	WSGIPath application	Proxy server nginx
Logs retention 7	Rotate logs Deactivated	Update level minor
X-Ray enabled Deactivated		

Environment properties

Key	Value
PYTHONPATH	/var/app/venv/staging-LQM1lest/bin

Elastic Beanstalk is launching environment



Elastic Beanstalk is launching your environment. This will take a few minutes.

HNApp-env Info

Environment overview

Health Grey	Environment ID e-zvhubwc9b8
Domain –	Application name HN App

Platform

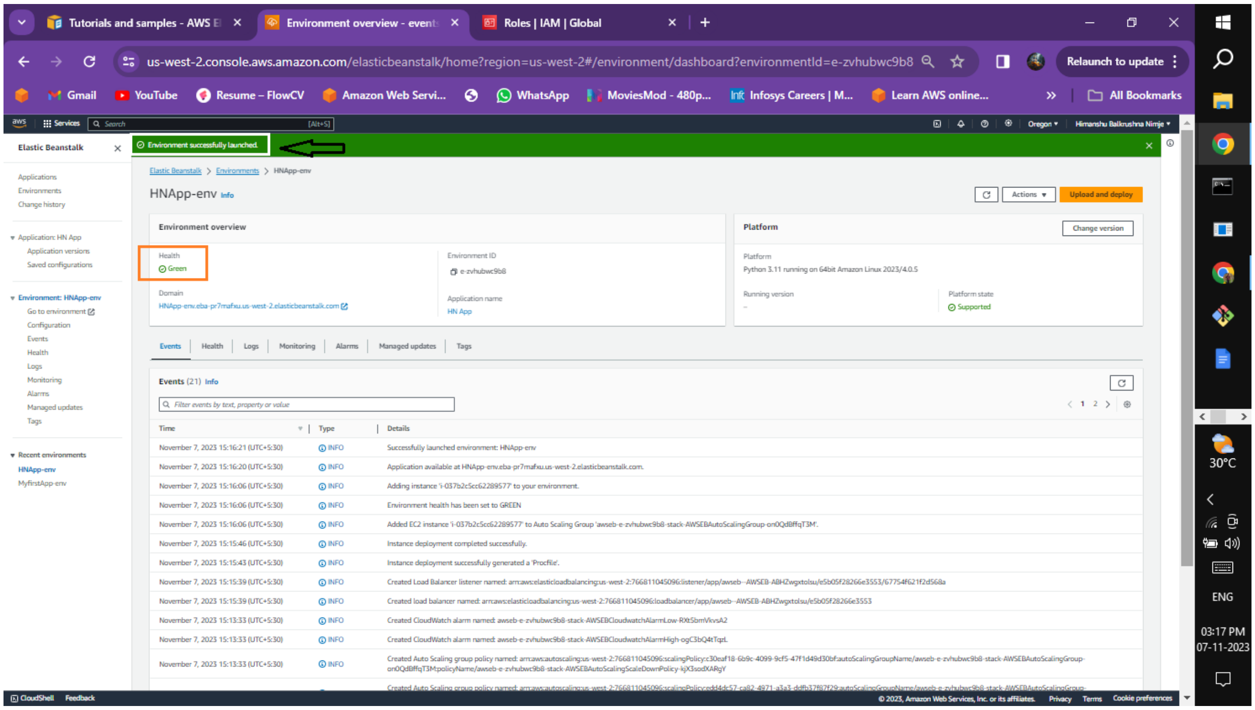
Platform
Python 3.11 running on 64bit Amazon Linux 2023/4.0.5

Running version –	Platform state Supported

Events (3) Info

Time	Type	Details
November 7, 2023 15:13:02 (UTC+5:30)	INFO	Created security group named: sg-09c1673fd7eeb8d70
November 7, 2023 15:12:42 (UTC+5:30)	INFO	Using elasticbeanstalk-us-west-2-766811045096 as Amazon S3 storage bucket for environment data.
November 7, 2023 15:12:41 (UTC+5:30)	INFO	createEnvironment is starting.

After 5 Minutes later Environment successfully launched



Environment successfully launched.

Elastic Beanstalk > Environments > HNApp-env

HNApp-env

Environment overview

Health: Green

Environment ID: e-zvhubwc9b8

Domain: HNApp-env.eba-pr7mafxu.us-west-2.elasticbeanstalk.com

Application name: HN App

Platform

Platform: Python 3.11 running on 64bit Amazon Linux 2023/4.0.5

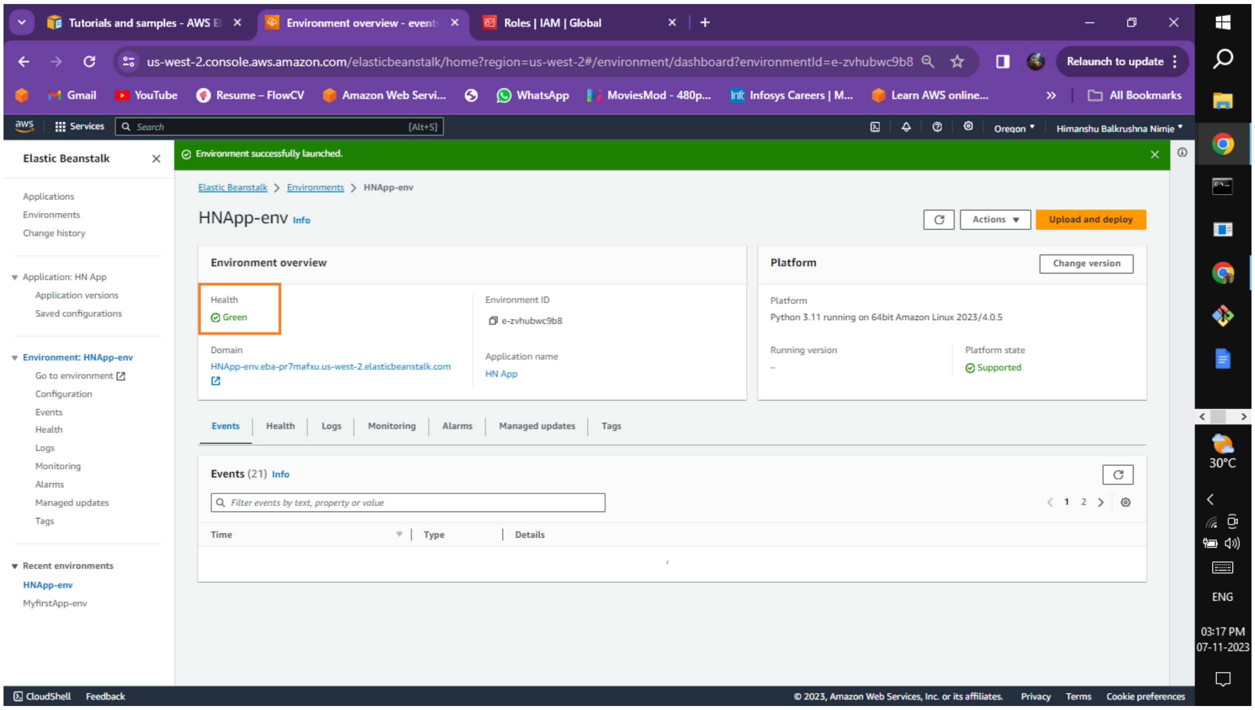
Running version: –

Platform state: Supported

Events (21)

Time	Type	Details
November 7, 2023 15:16:21 (UTC+5:30)	INFO	Successfully launched environment: HNApp-env
November 7, 2023 15:16:20 (UTC+5:30)	INFO	Application available at HNApp-env.eba-pr7mafxu.us-west-2.elasticbeanstalk.com.
November 7, 2023 15:16:06 (UTC+5:30)	INFO	Adding instance 'i-037b2c5cc62289577' to your environment.
November 7, 2023 15:16:06 (UTC+5:30)	INFO	Environment health has been set to GREEN
November 7, 2023 15:16:06 (UTC+5:30)	INFO	Added EC2 instance 'i-037b2c5cc62289577' to Auto Scaling Group 'awseb-e-zvhubwc9b8-stack-AWSEBAutoScalingGroup-on0QdBffqT3M'.
November 7, 2023 15:15:46 (UTC+5:30)	INFO	Instance deployment completed successfully.
November 7, 2023 15:15:43 (UTC+5:30)	INFO	Instance deployment successfully generated a 'Procfile'.
November 7, 2023 15:15:39 (UTC+5:30)	INFO	Created Load Balancer listener named: arn:aws:elasticloadbalancing:us-west-2:766811045096:listener/app/awseb--AWSEB-ABHZwgxtolsu/e5b05f28266e3553/67754f621f2d568a
November 7, 2023 15:15:39 (UTC+5:30)	INFO	Created load balancer named: arn:aws:elasticloadbalancing:us-west-2:766811045096:loadbalancer/app/awseb--AWSEB-ABHZwgxtolsu/e5b05f28266e3553
November 7, 2023 15:13:33 (UTC+5:30)	INFO	Created CloudWatch alarm named: awseb-e-zvhubwc9b8-stack-AWSEBCloudwatchAlarmLow-RXtSbmVkvsA2
November 7, 2023 15:13:33 (UTC+5:30)	INFO	Created CloudWatch alarm named: awseb-e-zvhubwc9b8-stack-AWSEBCloudwatchAlarmHigh-ogC3bQ4tTqzL
November 7, 2023 15:13:33 (UTC+5:30)	INFO	Created Auto Scaling group policy named: arn:aws:autoscaling:us-west-2:766811045096:scalingPolicy:c30eaf18-6b9c-4099-9cf5-47f1d49d30bf:autoScalingGroupName/awseb-e-zvhubwc9b8-stack-AWSEBAutoScalingGroup-on0QdBffqT3M:policyName/awseb-e-zvhubwc9b8-stack-AWSEBAutoScalingScaleDownPolicy-kjX3sodXARgY
		Created Auto Scaling group policy named: arn:aws:autoscaling:us-west-2:766811045096:scalingPolicy:edd4dc57-ca82-4971-a3a3-ddfb37f87f29:autoScalingGroupName/awseb-e-zvhubwc9b8-stack-AWSEBAutoScalingGroup-

The health of environment is showing Green



Environment successfully launched.

Elastic Beanstalk > Environments > HNApp-env

HNApp-env

Environment overview

Health: Green

Environment ID: e-zvhubwc9b8

Domain: HNApp-env.eba-pr7mafxu.us-west-2.elasticbeanstalk.com

Application name: HN App

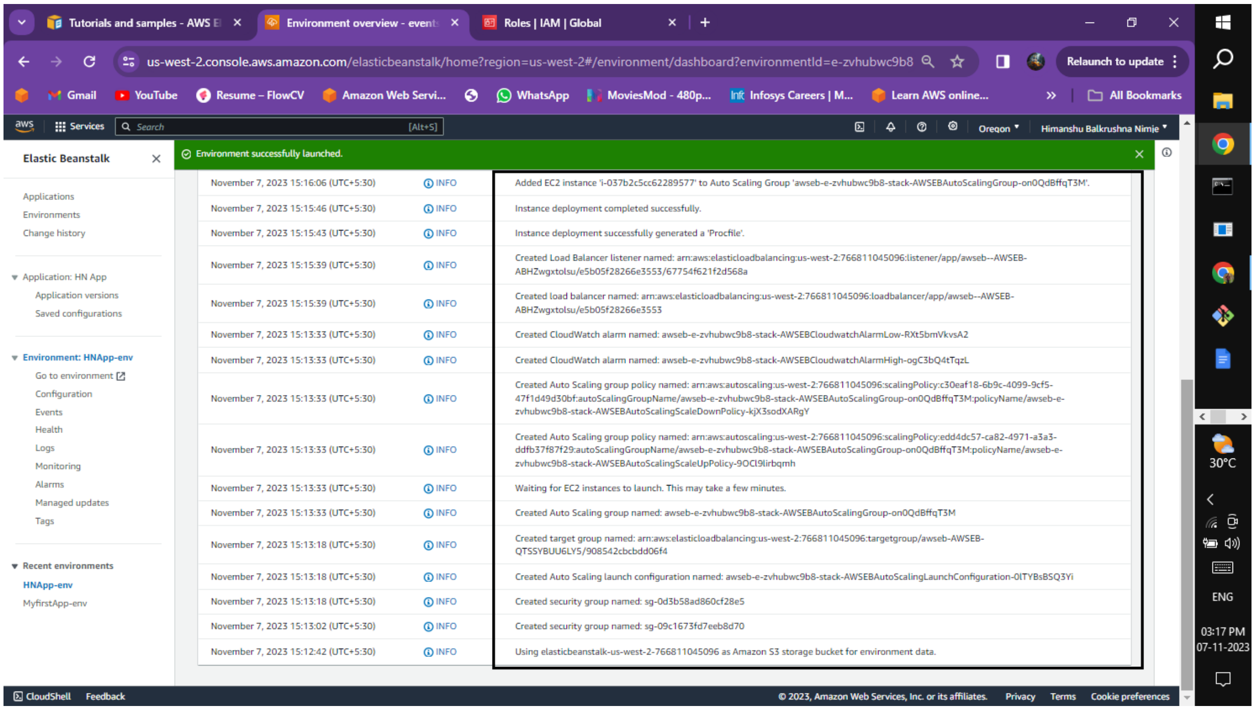
Platform

Platform: Python 3.11 running on 64bit Amazon Linux 2023/4.0.5

Running version: –

Platform state: Supported

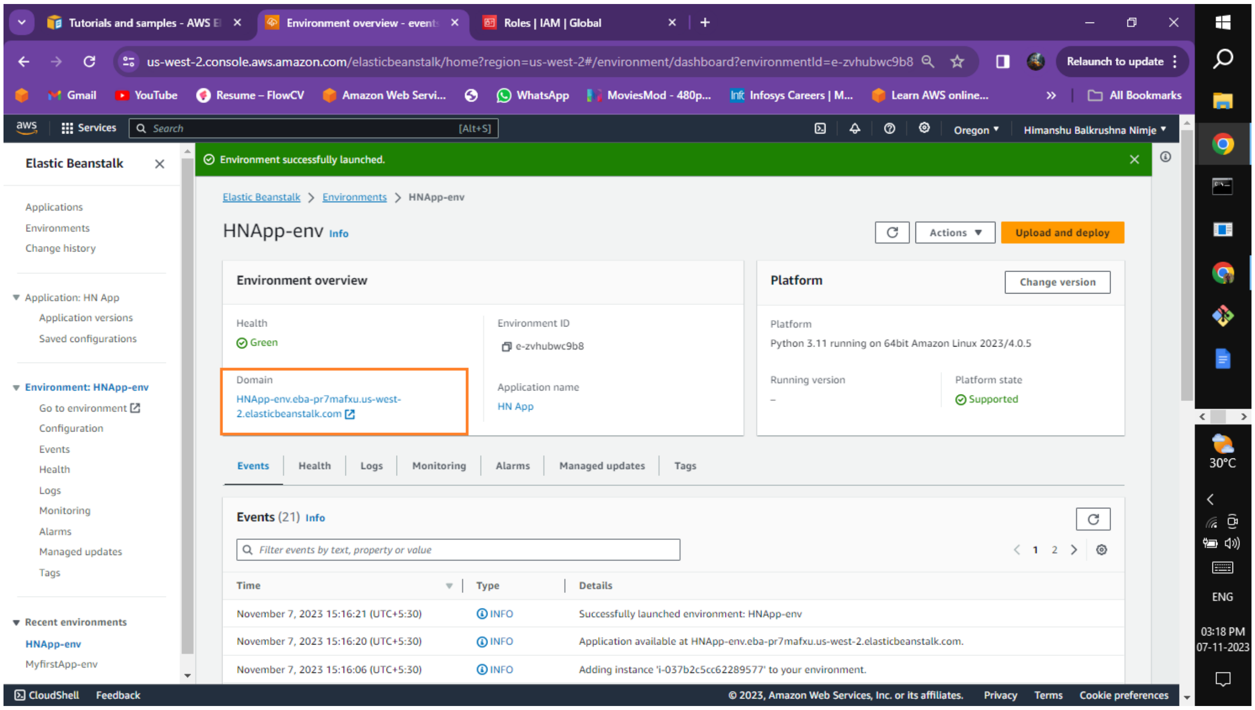
Events (21)



Environment successfully launched.

November 7, 2023 15:16:06 (UTC+5:30)	INFO	Added EC2 instance 'i-037b2c5cc62289577' to Auto Scaling Group 'awseb-e-zvhubwc9b8-stack-AWSEBAutoScalingGroup-on0QdBffqT3M'.
November 7, 2023 15:15:46 (UTC+5:30)	INFO	Instance deployment completed successfully.
November 7, 2023 15:15:43 (UTC+5:30)	INFO	Instance deployment successfully generated a 'Procfile'.
November 7, 2023 15:15:39 (UTC+5:30)	INFO	Created Load Balancer listener named: arn:aws:elasticloadbalancing:us-west-2:766811045096:listener/app/awseb--AWSEB-ABHZwgxtolsu/e5b05f28266e3553/67754f621f2d568a
November 7, 2023 15:15:39 (UTC+5:30)	INFO	Created load balancer named: arn:aws:elasticloadbalancing:us-west-2:766811045096:loadbalancer/app/awseb--AWSEB-ABHZwgxtolsu/e5b05f28266e3553
November 7, 2023 15:13:33 (UTC+5:30)	INFO	Created CloudWatch alarm named: awseb-e-zvhubwc9b8-stack-AWSEBCloudwatchAlarmLow-RXt5bmVkvsA2
November 7, 2023 15:13:33 (UTC+5:30)	INFO	Created CloudWatch alarm named: awseb-e-zvhubwc9b8-stack-AWSEBCloudwatchAlarmHigh-ogC3bQ4tTqzL
November 7, 2023 15:13:33 (UTC+5:30)	INFO	Created Auto Scaling group policy named: arn:aws:autoscaling:us-west-2:766811045096:scalingPolicy:c30eaf18-6b9c-4099-9cf5-47f1d49d30bf:autoScalingGroupName/awseb-e-zvhubwc9b8-stack-AWSEBAutoScalingGroup-on0QdBffqT3M:policyName/awseb-e-zvhubwc9b8-stack-AWSEBAutoScalingScaleDownPolicy-kjX3sodXARgY
November 7, 2023 15:13:33 (UTC+5:30)	INFO	Created Auto Scaling group policy named: arn:aws:autoscaling:us-west-2:766811045096:scalingPolicy:edd4dc57-ca82-4971-a3a3-ddfb37f87f29:autoScalingGroupName/awseb-e-zvhubwc9b8-stack-AWSEBAutoScalingGroup-on0QdBffqT3M:policyName/awseb-e-zvhubwc9b8-stack-AWSEBAutoScalingScaleUpPolicy-9OCl9lirbqmh
November 7, 2023 15:13:33 (UTC+5:30)	INFO	Waiting for EC2 instances to launch. This may take a few minutes.
November 7, 2023 15:13:33 (UTC+5:30)	INFO	Created Auto Scaling group named: awseb-e-zvhubwc9b8-stack-AWSEBAutoScalingGroup-on0QdBffqT3M
November 7, 2023 15:13:18 (UTC+5:30)	INFO	Created target group named: arn:aws:elasticloadbalancing:us-west-2:766811045096:targetgroup/awseb-AWSEB-QTSSYBUU6LY5/908542cbcbdd06f4
November 7, 2023 15:13:18 (UTC+5:30)	INFO	Created Auto Scaling launch configuration named: awseb-e-zvhubwc9b8-stack-AWSEBAutoScalingLaunchConfiguration-0lTYBsBSQ3Yi
November 7, 2023 15:13:18 (UTC+5:30)	INFO	Created security group named: sg-0d3b58ad860cf28e5
November 7, 2023 15:13:02 (UTC+5:30)	INFO	Created security group named: sg-09c1673fd7eeb8d70
November 7, 2023 15:12:42 (UTC+5:30)	INFO	Using elasticbeanstalk-us-west-2-766811045096 as Amazon S3 storage bucket for environment data.

click on the Domain



Environment successfully launched.

Elastic Beanstalk > Environments > HNApp-env

HNApp-env

Environment overview

Health: Green

Environment ID: e-zvhubwc9b8

Domain: HNApp-env.eba-pr7mafxu.us-west-2.elasticbeanstalk.com

Application name: HN App

Platform

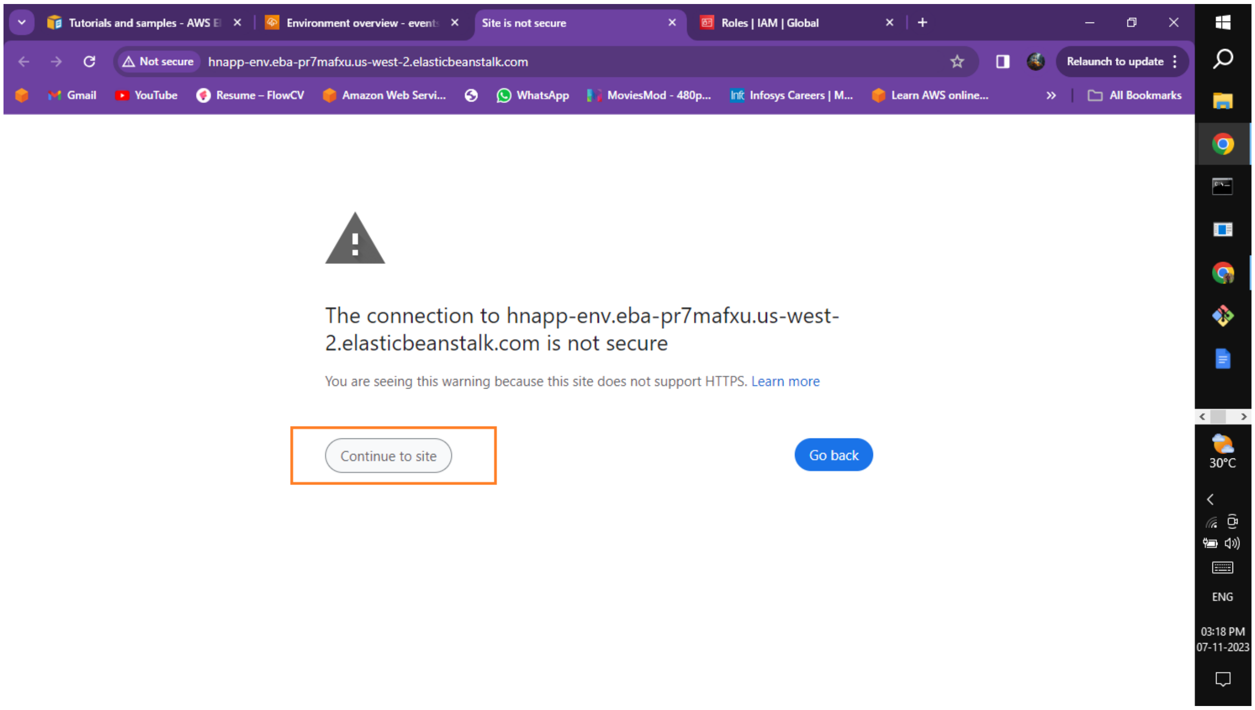
Platform: Python 3.11 running on 64bit Amazon Linux 2023/4.0.5

Running version: –

Platform state: Supported

Events (21)

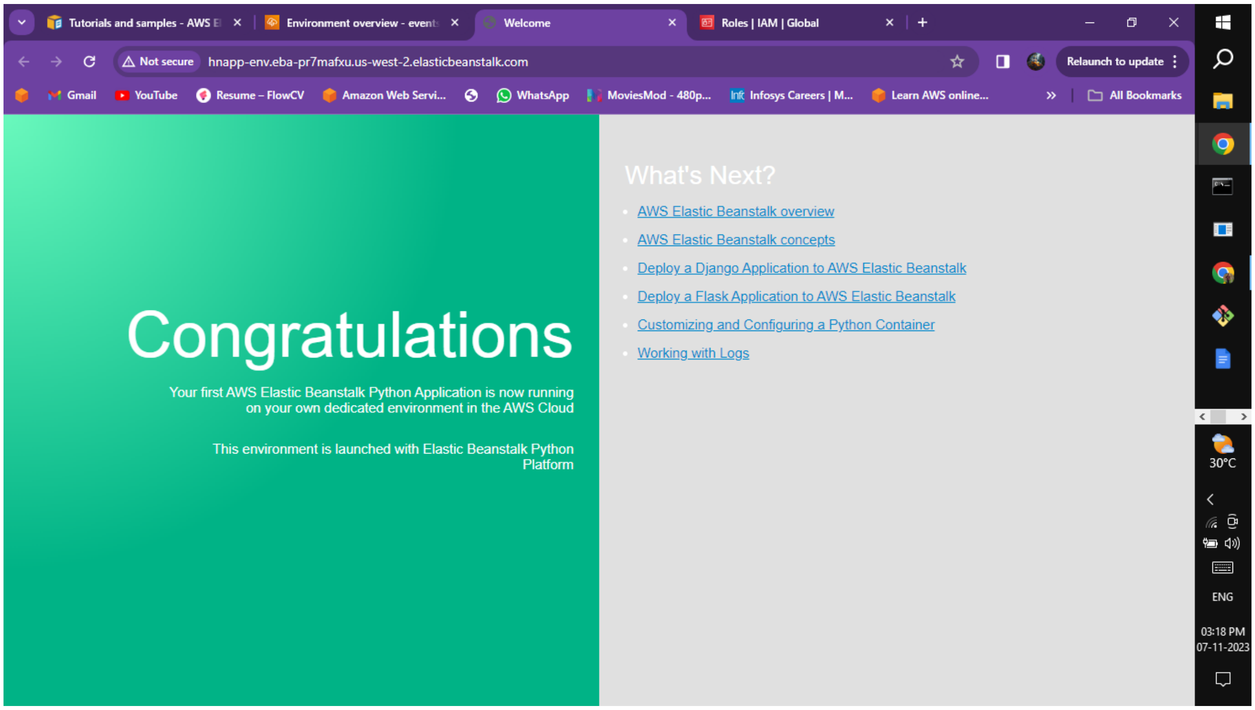
Time	Type	Details
November 7, 2023 15:16:21 (UTC+5:30)	INFO	Successfully launched environment: HNApp-env
November 7, 2023 15:16:20 (UTC+5:30)	INFO	Application available at HNApp-env.eba-pr7mafxu.us-west-2.elasticbeanstalk.com.
November 7, 2023 15:16:06 (UTC+5:30)	INFO	Adding instance 'i-037b2c5cc62289577' to your environment.



The connection to hnapp-env.eba-pr7mafxu.us-west-2.elasticbeanstalk.com is not secure

You are seeing this warning because this site does not support HTTPS. Learn more

App Environment has Launched



Congratulations

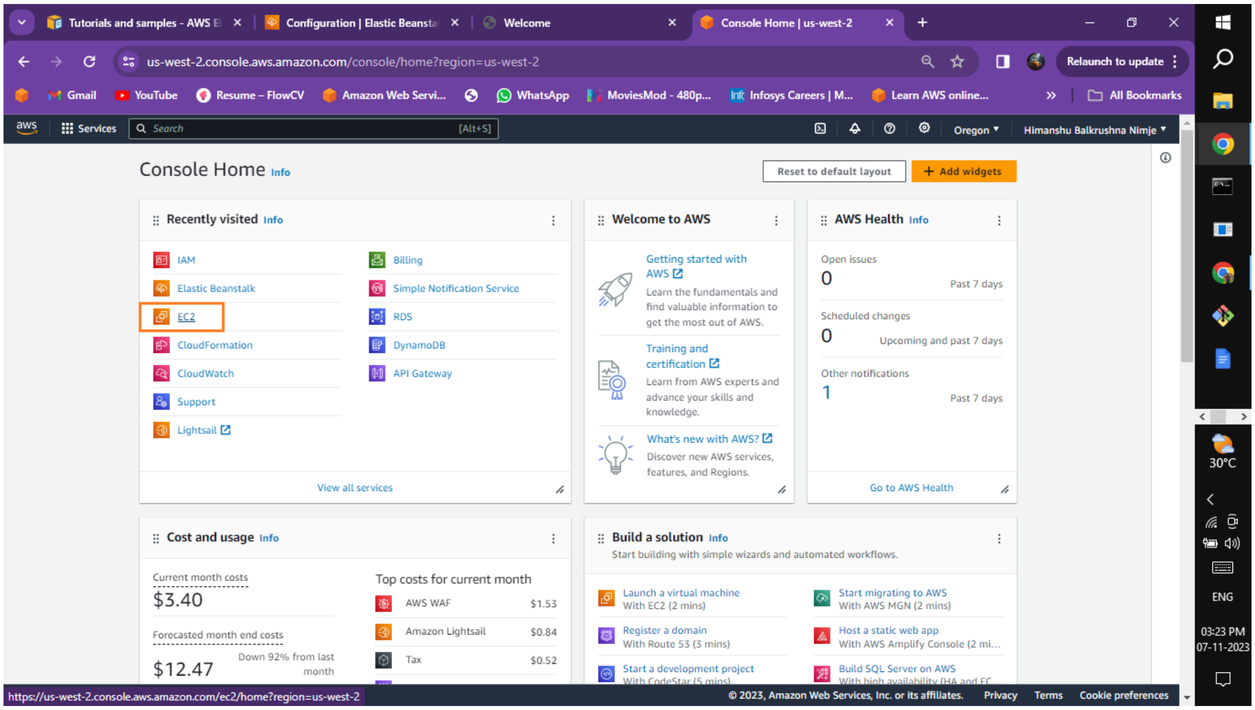
Your first AWS Elastic Beanstalk Python Application is now running on your own dedicated environment in the AWS Cloud

This environment is launched with Elastic Beanstalk Python Platform

What's Next?

- AWS Elastic Beanstalk overview
- AWS Elastic Beanstalk concepts
- Deploy a Django Application to AWS Elastic Beanstalk
- Deploy a Flask Application to AWS Elastic Beanstalk
- Customizing and Configuring a Python Container
- Working with Logs

Let's see the services Behind the app, Go to EC2 and Click on



Console Home Info

Reset to default layout | + Add widgets

Recently visited Info

IAM	Billing
Elastic Beanstalk	Simple Notification Service
EC2	RDS
CloudFormation	DynamoDB
CloudWatch	API Gateway
Support	
Lightsail	

View all services

Welcome to AWS

Getting started with AWS
Learn the fundamentals and find valuable information to get the most out of AWS.

Training and certification
Learn from AWS experts and advance your skills and knowledge.

What's new with AWS?
Discover new AWS services, features, and Regions.

AWS Health Info

Open issues
0 Past 7 days

Scheduled changes
0 Upcoming and past 7 days

Other notifications
1 Past 7 days

Go to AWS Health

Cost and usage Info

Current month costs
$3.40

Forecasted month end costs
$12.47 Down 92% from last month

Top costs for current month

AWS WAF	$1.53
Amazon Lightsail	$0.84
Tax	$0.52

Build a solution Info
Start building with simple wizards and automated workflows.

Launch a virtual machine
With EC2 (2 mins)

Start migrating to AWS
With AWS MGN (2 mins)

Register a domain
With Route 53 (3 mins)

Host a static web app
With AWS Amplify Console (2 mi...

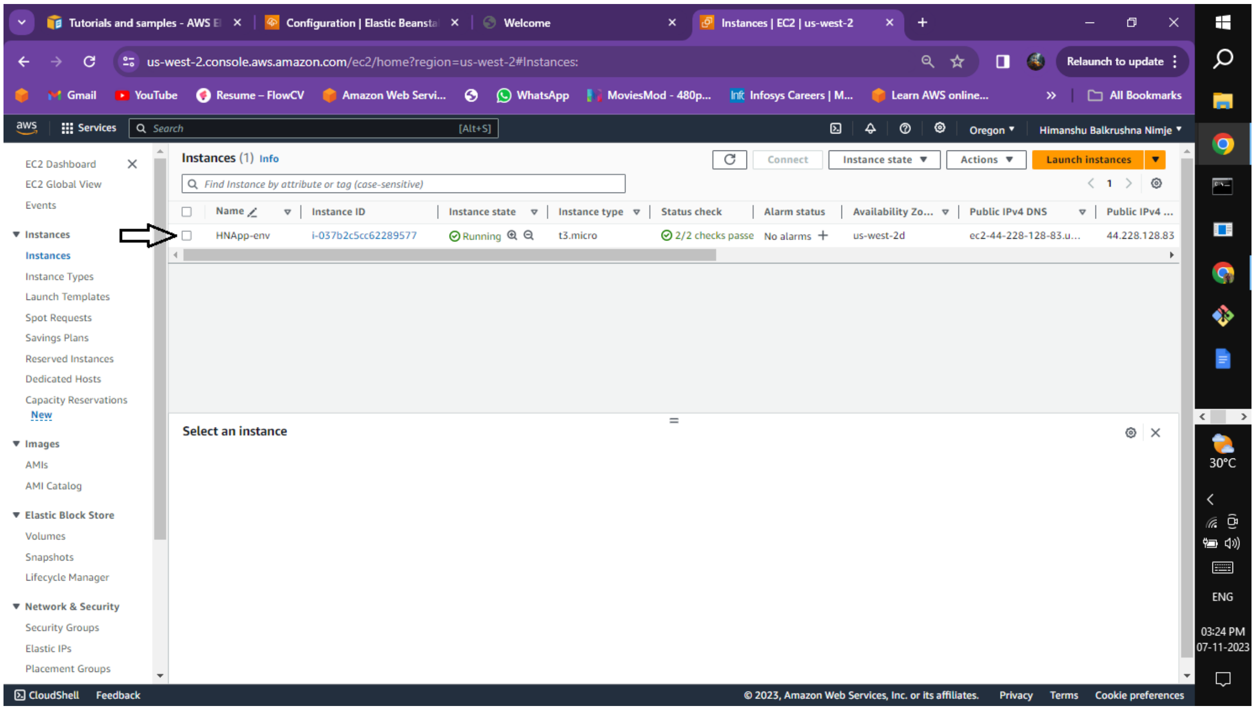
Start a development project
With CodeStar (5 mins)

Build SQL Server on AWS
With high availability (HA and FC

https://us-west-2.console.aws.amazon.com/ec2/home?region=us-west-2

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Here is App Instances



Instances (1) Info

Connect | Instance state ▼ | Actions ▼ | Launch instances ▼

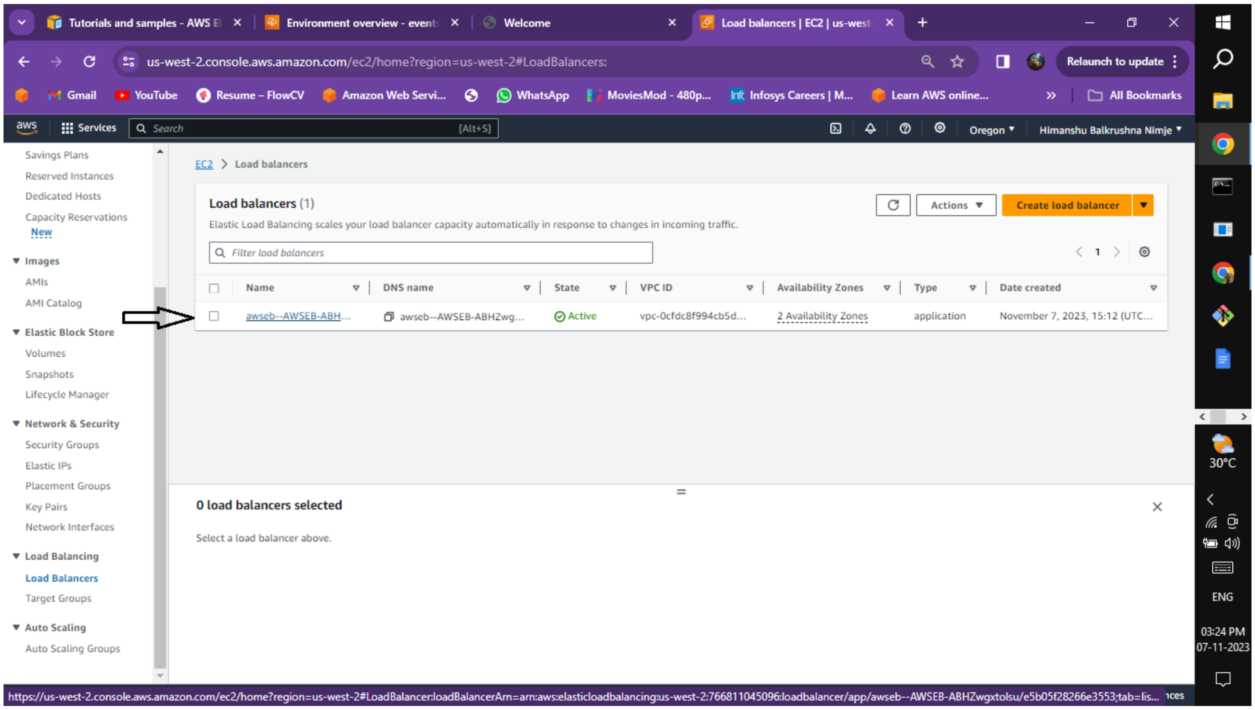
Find Instance by attribute or tag (case-sensitive)

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zo...	Public IPv4 DNS	Public IPv4 ...
□	HNApp-env	i-037b2c5cc62289577	Running	t3.micro	2/2 checks passe	No alarms +	us-west-2d	ec2-44-228-128-83.u...	44.228.128.83

Select an instance

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Here is App LoadBalancer



EC2 > Load balancers

Load balancers (1)

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

Actions ▼ | Create load balancer ▼

Filter load balancers

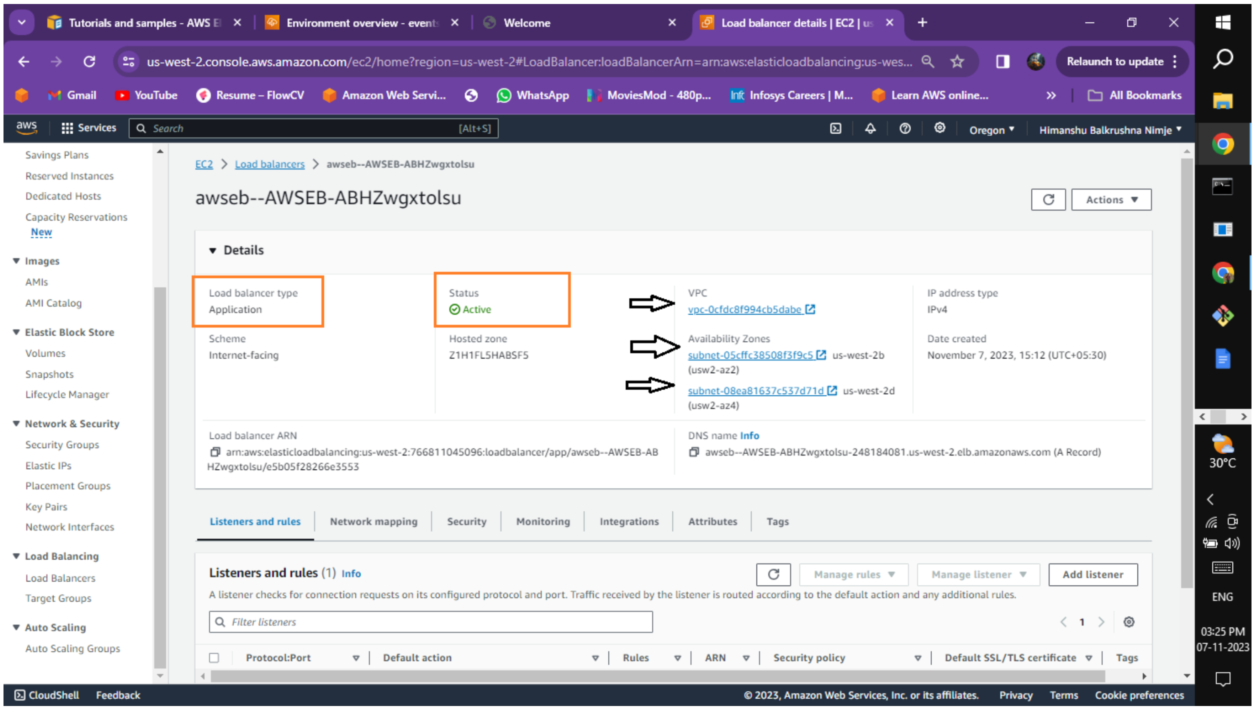
□	Name	DNS name	State	VPC ID	Availability Zones	Type	Date created
□	awseb--AWSEB-ABH...	awseb--AWSEB-ABHZwg...	✓ Active	vpc-0cfdc8f994cb5d...	2 Availability Zones	application	November 7, 2023, 15:12 (UTC...

0 load balancers selected

Select a load balancer above.

https://us-west-2.console.aws.amazon.com/ec2/home?region=us-west-2#LoadBalancer:loadBalancerArn=arn:aws:elasticloadbalancing:us-west-2:766811045096:loadbalancer/app/awseb--AWSEB-ABHZwgxtolsu/e5b05f28266e3553;tab=lis...

Here is VPC Availability Zones and Subnets



EC2 > Load balancers > awseb--AWSEB-ABHZwgxtolsu

awseb--AWSEB-ABHZwgxtolsu

Actions ▼

Details

Load balancer type: Application

Status: ✓ Active

VPC: vpc-0cfdc8f994cb5dabe

IP address type: IPv4

Scheme: Internet-facing

Hosted zone: Z1H1FL5HABSF5

Availability Zones: subnet-05cffc38508f3f9c5 us-west-2b (usw2-az2); subnet-08ea81637c537d71d us-west-2d (usw2-az4)

Date created: November 7, 2023, 15:12 (UTC+05:30)

Load balancer ARN: arn:aws:elasticloadbalancing:us-west-2:766811045096:loadbalancer/app/awseb--AWSEB-ABHZwgxtolsu/e5b05f28266e3553

DNS name Info: awseb--AWSEB-ABHZwgxtolsu-248184081.us-west-2.elb.amazonaws.com (A Record)

Listeners and rules | Network mapping | Security | Monitoring | Integrations | Attributes | Tags

Listeners and rules (1) Info

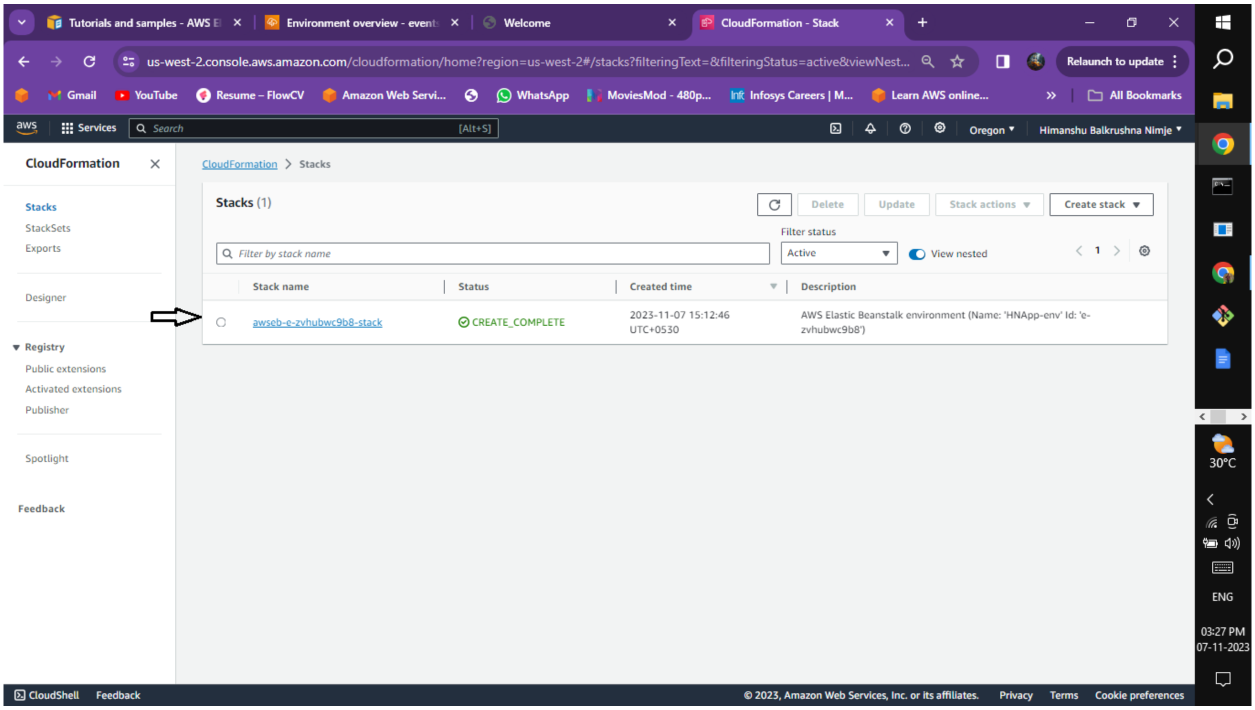
Manage rules ▼ | Manage listener ▼ | Add listener

A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

Filter listeners

□	Protocol:Port	Default action	Rules	ARN	Security policy	Default SSL/TLS certificate	Tags

Design of App Environment

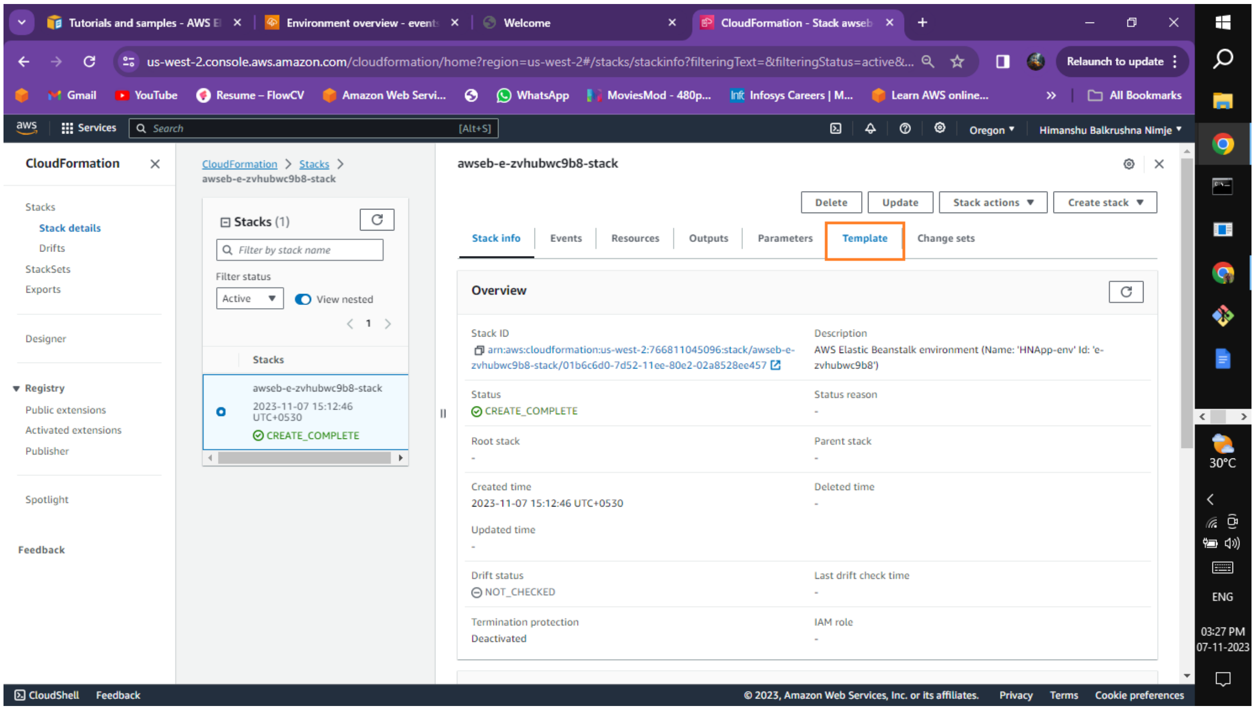


Stacks (1)

Filter status: Active

Stack name	Status	Created time	Description
awseb-e-zvhubwc9b8-stack	CREATE_COMPLETE	2023-11-07 15:12:46 UTC+0530	AWS Elastic Beanstalk environment (Name: 'HNApp-env' Id: 'e-zvhubwc9b8')

click on the cloudformation go to stack option, Then go to Template Option



Stacks: awseb-e-zvhubwc9b8-stack, 2023-11-07 15:12:46 UTC+0530, CREATE_COMPLETE

awseb-e-zvhubwc9b8-stack

Overview

Stack ID: arn:aws:cloudformation:us-west-2:766811045096:stack/awseb-e-zvhubwc9b8-stack/01b6c6d0-7d52-11ee-80e2-02a8528ee457

Description: AWS Elastic Beanstalk environment (Name: 'HNApp-env' Id: 'e-zvhubwc9b8')

Status: CREATE_COMPLETE

Status reason: -

Root stack: -

Parent stack: -

Created time: 2023-11-07 15:12:46 UTC+0530

Deleted time: -

Updated time: -

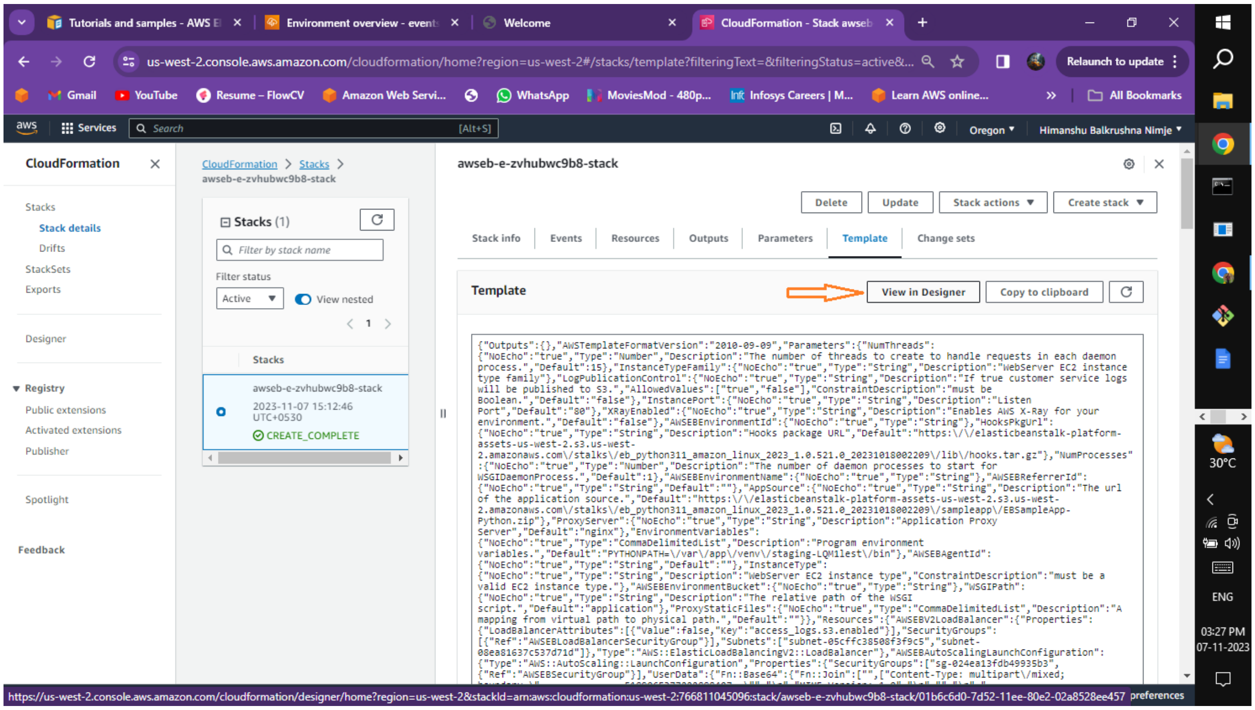
Drift status: NOT_CHECKED

Last drift check time: -

Termination protection: Deactivated

IAM role: -

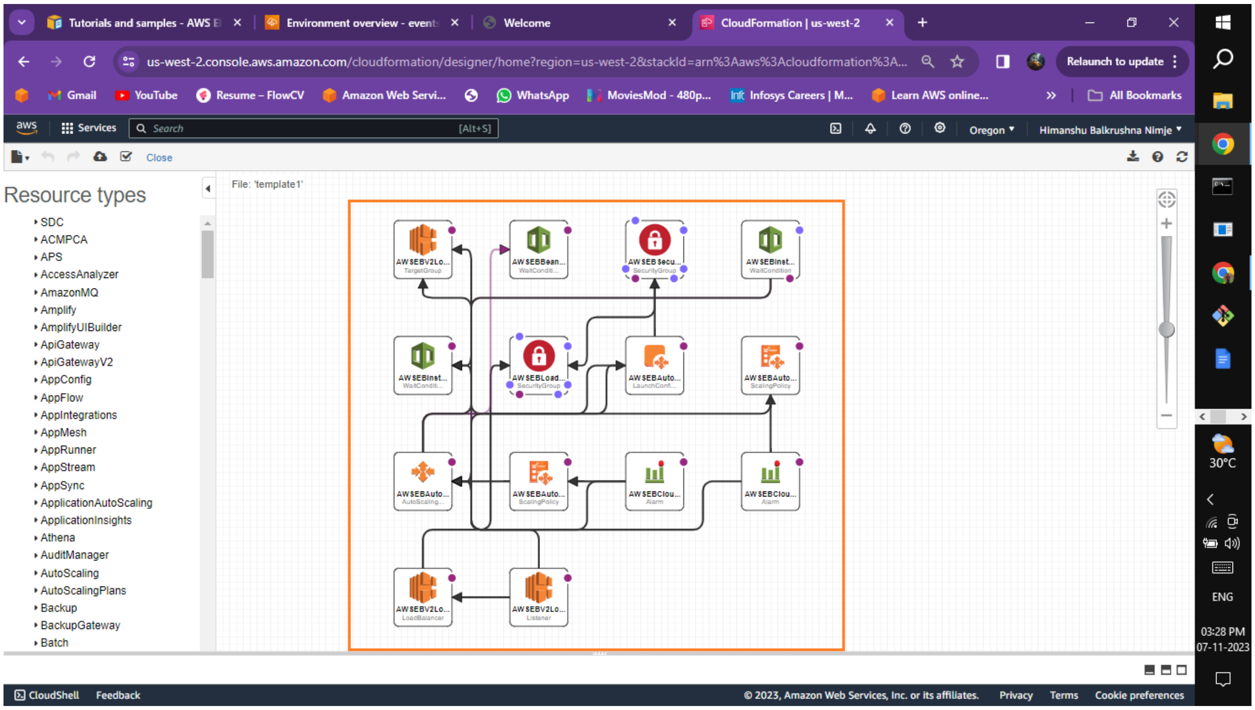
click on View in Designer option



```
{"Outputs":{},"AWSTemplateFormatVersion":"2010-09-09","Parameters":{"NumThreads":
{"NoEcho":"true","Type":"Number","Description":"The number of threads to create to handle requests in each daemon
process.","Default":15},"InstanceTypeFamily":{"NoEcho":"true","Type":"String","Description":"WebServer EC2 instance
type family"},"LogPublicationControl":{"NoEcho":"true","Type":"String","Description":"If true customer service logs
will be published to S3.","AllowedValues":["true","false"],"ConstraintDescription":"must be
Boolean.","Default":"false"},"InstancePort":{"NoEcho":"true","Type":"String","Description":"Listen
Port","Default":"80"},"XRayEnabled":{"NoEcho":"true","Type":"String","Description":"Enables AWS X-Ray for your
environment.","Default":"false"},"AWSEBEnvironmentId":{"NoEcho":"true","Type":"String"},"HooksPkgUrl":
{"NoEcho":"true","Type":"String","Description":"Hooks package URL","Default":"https:\/\/elasticbeanstalk-platform-
assets-us-west-2.s3.us-west-
2.amazonaws.com\/stalks\/eb_python311_amazon_linux_2023_1.0.521.0_20231018002209\/lib\/hooks.tar.gz"},"NumProcesses"
:{"NoEcho":"true","Type":"Number","Description":"The number of daemon processes to start for
WSGIDaemonProcess.","Default":1},"AWSEBEnvironmentName":{"NoEcho":"true","Type":"String"},"AWSEBReferrerId":
{"NoEcho":"true","Type":"String","Default":""},"AppSource":{"NoEcho":"true","Type":"String","Description":"The url
of the application source.","Default":"https:\/\/elasticbeanstalk-platform-assets-us-west-2.s3.us-west-
2.amazonaws.com\/stalks\/eb_python311_amazon_linux_2023_1.0.521.0_20231018002209\/sampleapp\/EBSampleApp-
Python.zip"},"ProxyServer":{"NoEcho":"true","Type":"String","Description":"Application Proxy
Server","Default":"nginx"},"EnvironmentVariables":
{"NoEcho":"true","Type":"CommaDelimitedList","Description":"Program environment
variables.","Default":"PYTHONPATH=\/var\/app\/venv\/staging-LQM1lest\/bin"},"AWSEBAgentId":
{"NoEcho":"true","Type":"String","Default":""},"InstanceType":
{"NoEcho":"true","Type":"String","Description":"WebServer EC2 instance type","ConstraintDescription":"must be a
valid EC2 instance type."},"AWSEBEnvironmentBucket":{"NoEcho":"true","Type":"String"},"WSGIPath":
{"NoEcho":"true","Type":"String","Description":"The relative path of the WSGI
script.","Default":"application"},"ProxyStaticFiles":{"NoEcho":"true","Type":"CommaDelimitedList","Description":"A
mapping from virtual path to physical path.","Default":""}},"Resources":{"AWSEBV2LoadBalancer":{"Properties":
{"LoadBalancerAttributes":[{"Value":false,"Key":"access_logs.s3.enabled"}],"SecurityGroups":
[{"Ref":"AWSEBLoadBalancerSecurityGroup"}],"Subnets":["subnet-05cffc38508f3f9c5","subnet-
08ea81637c537d71d"]},"Type":"AWS::ElasticLoadBalancingV2::LoadBalancer"},"AWSEBAutoScalingLaunchConfiguration":
{"Type":"AWS::AutoScaling::LaunchConfiguration","Properties":{"SecurityGroups":["sg-024ea13fdb49935b3",
{"Ref":"AWSEBSecurityGroup"}],"UserData":{"Fn::Base64":{"Fn::Join":["",["Content-Type: multipart\/mixed;
```

https://us-west-2.console.aws.amazon.com/cloudformation/designer/home?region=us-west-2&stackId=arn:aws:cloudformation:us-west-2:766811045096:stack/awseb-e-zvhubwc9b8-stack/01b6c6d0-7d52-11ee-80e2-02a8528ee457

Here is The Architecture Diagram Of App environment


Tutorials and samples - AWS E
Environment overview - event
Welcome
CloudFormation | us-west-2
us-west-2.console.aws.amazon.com/cloudformation/designer/home?region=us-west-2&stackId=arn%3Aaws%3Acloudformation%3A...
Relaunch to update
Services
Search
[Alt+S]
Oregon ▼
Himanshu Balkrushna Nimje ▼
Close
Resource types
SDC
ACMPCA
APS
AccessAnalyzer
AmazonMQ
Amplify
AmplifyUIBuilder
ApiGateway
ApiGatewayV2
AppConfig
AppFlow
AppIntegrations
AppMesh
AppRunner
AppStream
AppSync
ApplicationAutoScaling
ApplicationInsights
Athena
AuditManager
AutoScaling
AutoScalingPlans
Backup
BackupGateway
Batch
File: 'template1'
AWSEBV2Lo... TargetGroup
AWSEBBean... WaitConditi...
AWSEBSecu... SecurityGroup
AWSEBInst... WaitCondition
AWSEBInst... WaitConditi...
AWSEBLoad... SecurityGroup
AWSEBAuto... LaunchConfi...
AWSEBAuto... ScalingPolicy
AWSEBAuto... AutoScaling...
AWSEBAuto... ScalingPolicy
AWSEBClou... Alarm
AWSEBClou... Alarm
AWSEBV2Lo... LoadBalancer
AWSEBV2Lo... Listener
CloudShell
Feedback
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
30°C
ENG
03:28 PM
07-11-2023

What I Learned

1. Elastic Beanstalk Environment:- Understanding what an Elastic Beanstalk environment is and how it provides a platform for deploying and running web applications without managing the underlying infrastructure.
2. Elastic Beanstalk Console:- Navigating the Elastic Beanstalk console in the AWS Management Console and the key sections such as environments, applications, and configurations.
3. Creating an Elastic Beanstalk Application:- Creating an Elastic Beanstalk application, giving it a name, and understanding how it serves as a container for your web application.
4. Environment Configuration:- Configuring your Elastic Beanstalk environment, including settings like instance type, instance count, environment variables, and more.
5. Scaling Options:- Learning about the automatic scaling capabilities of Elastic Beanstalk and how to configure scaling options based on your application's needs.
6. Health Monitoring:- Monitoring the health of application and environment, including understanding how Elastic Beanstalk manages instance health and replaces unhealthy instances.
7. Clean-Up and Termination:- Learning how to terminate an Elastic Beanstalk environment when it's no longer needed to avoid incurring unnecessary costs.

By going through the process of deploying a web application with Elastic Beanstalk, I have gained practical experience in deploying, managing, and scaling applications in a simplified and automated manner within the AWS environment.