



**Ganpat
University**

॥ विद्यया समाजोत्कर्षः ॥

**Institute of
Computer
Technology**

Name: Tushar Panchal

En.No: 21162101014

Sub: EADC (Enterprise Application Development for Cloud)

Branch: CBA

Batch:61

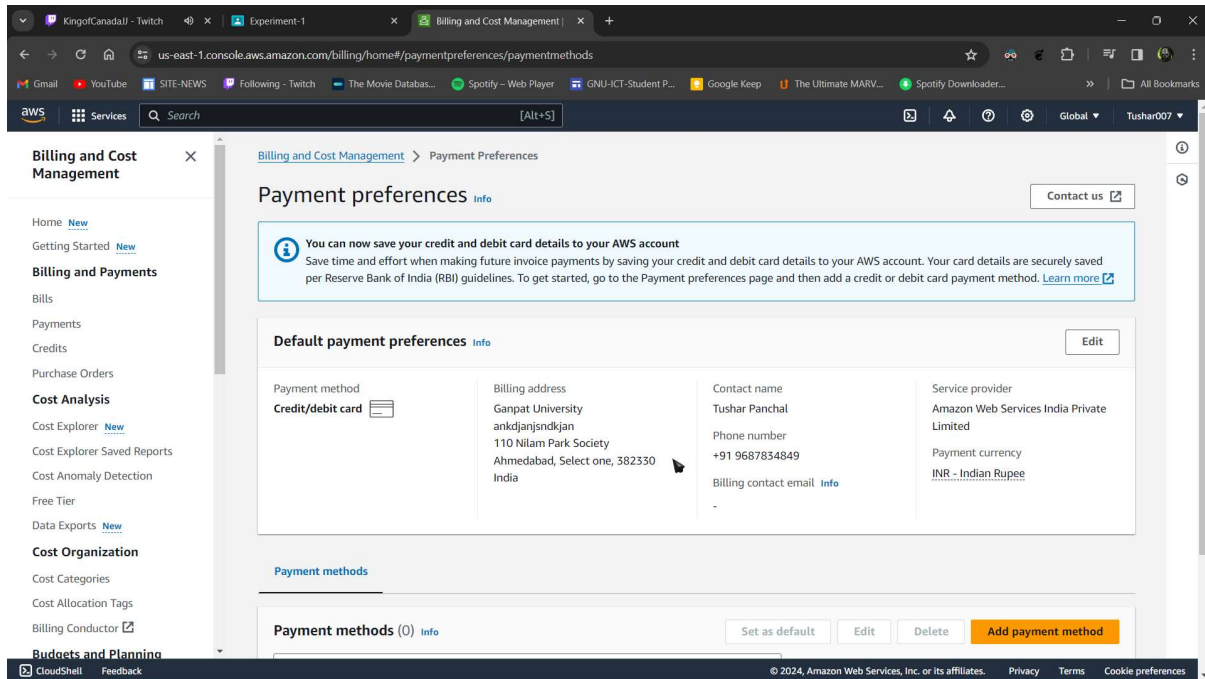
PRACTICAL 01

❖ Question :

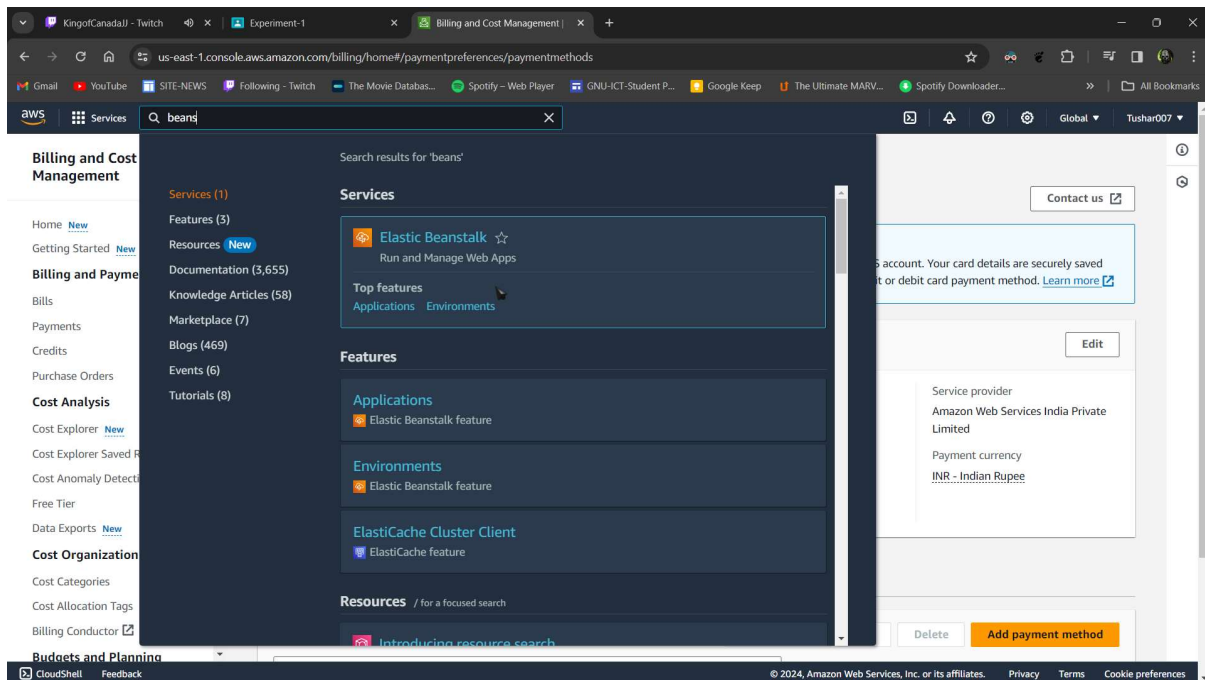
You have a requirement for deploying an existing NodeJS-based application to AWS Cloud. There is a need for automatic scaling for the underlying environment. Implement the AWS Cloud service and resources used to deploy this environment in the quickest way possible.

» First of all I deployed sample application. Here's the procedure of making it :

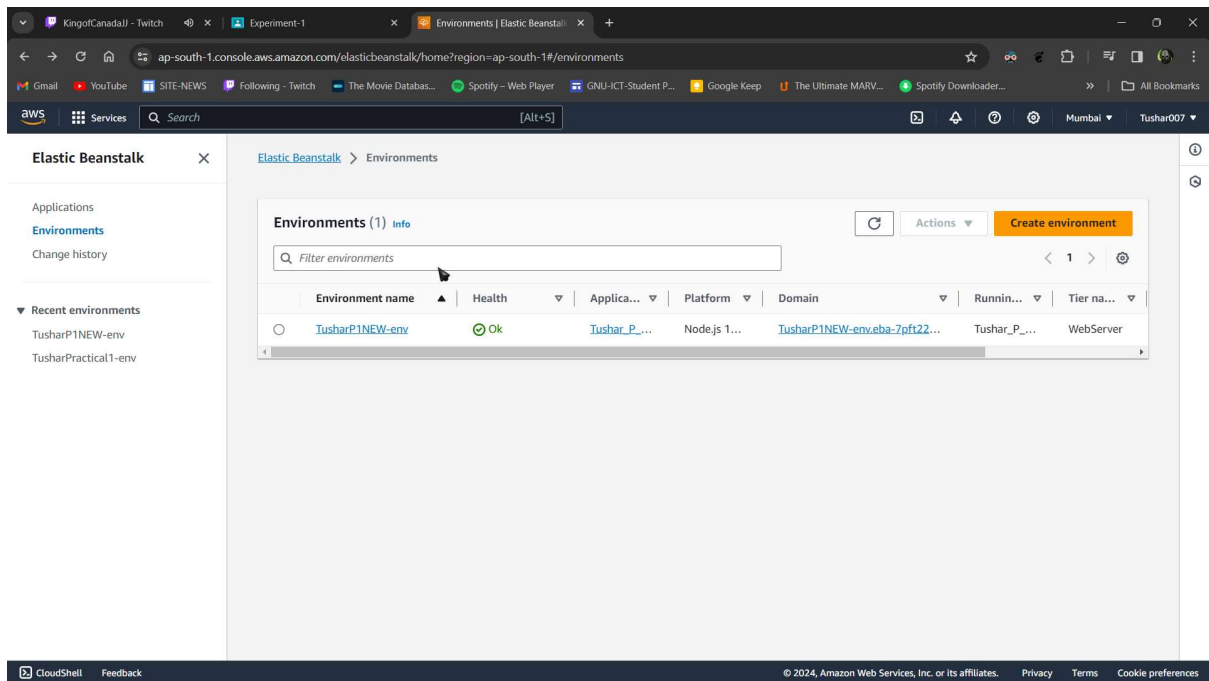
- First I Created the AWS account and logged in to the console :



- Then search for the service named "beanstalk" :

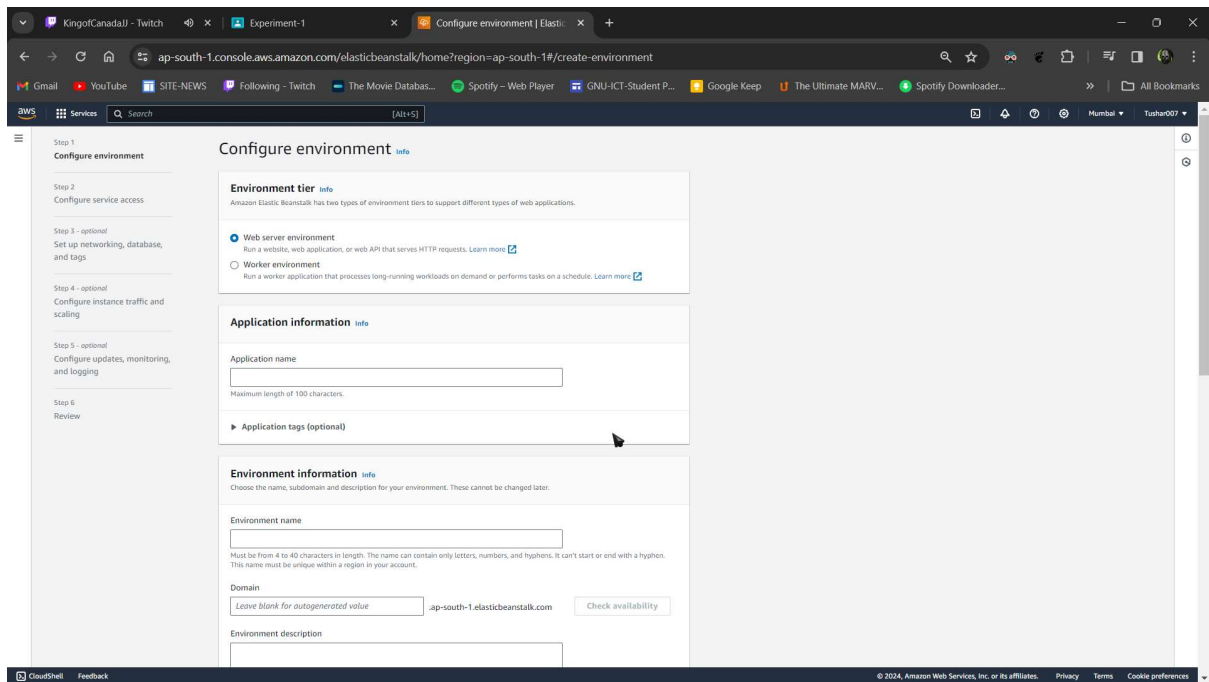


➤ Selected Beanstalk Service :



The screenshot shows the AWS Elastic Beanstalk console. The left sidebar has a menu with 'Applications', 'Environments', and 'Change history'. The main content area is titled 'Environments (1) Info'. It features a search bar and a table of environments. The table has columns: Environment name, Health, Application, Platform, Domain, Running, and Tier name. One environment is listed: 'TusharP1NEW-env' with a status of 'Ok', application 'Tushar_P...', platform 'Node.js 1...', domain 'TusharP1NEW-env.eba-7pft22...', running status 'Tushar_P...', and tier name 'WebServer'.

➤ Then clicked on the “Create Application” :



The screenshot shows the 'Configure environment' page in the AWS Elastic Beanstalk console. The left sidebar shows a progress bar with steps: Step 1: Configure environment (active), 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, and Step 6: Review. The main content area is titled 'Configure environment Info'. It has sections for 'Environment tier' (Web server environment selected), 'Application information' (Application name field), and 'Environment information' (Environment name and Domain fields). The Domain field is pre-filled with 'ap-south-1.elasticbeanstalk.com' and has a 'Check availability' button.

➤ Then gave the name for the application :

Step 1: Configure environment

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

Configure environment

Environment tier

Amazon Elastic Beanstalk has two types of environment tiers to support different types of web applications.

- ☒ Web server environment: Run a website, web application, or web API that serves HTTP requests.
- ☐ Worker environment: Run a worker application that processes long-running workloads on demand or performs tasks on a schedule.

Application information

Application name: (Maximum length of 100 characters)

Application tags (optional)

Environment information

Choose the name, subdomain and description for your environment. These cannot be changed later.

Environment name: (Must be from 4 to 40 characters in length. The name can contain only letters, numbers, and hyphens. It can't start or end with a hyphen. This name must be unique within a region in your account.)

Domain: .ap-south-1.elasticbeanstalk.com

Environment description:

➤ Selected Platform for Node Js and in application code i kept "Sample Application" :

Step 1: Configure environment

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

Configure environment

Platform

Platform type: ☒ Managed platform (Platforms published and maintained by Amazon Elastic Beanstalk.)

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

Platform:

Platform branch:

Platform version:

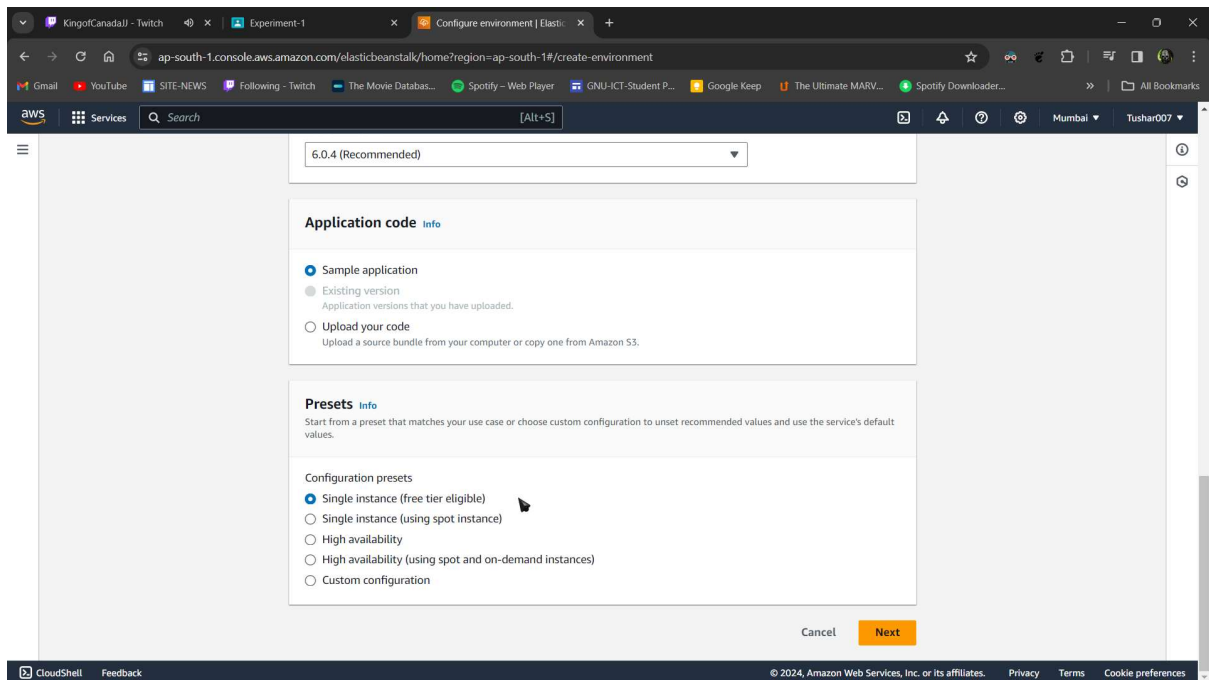
Application code

☒ Sample application

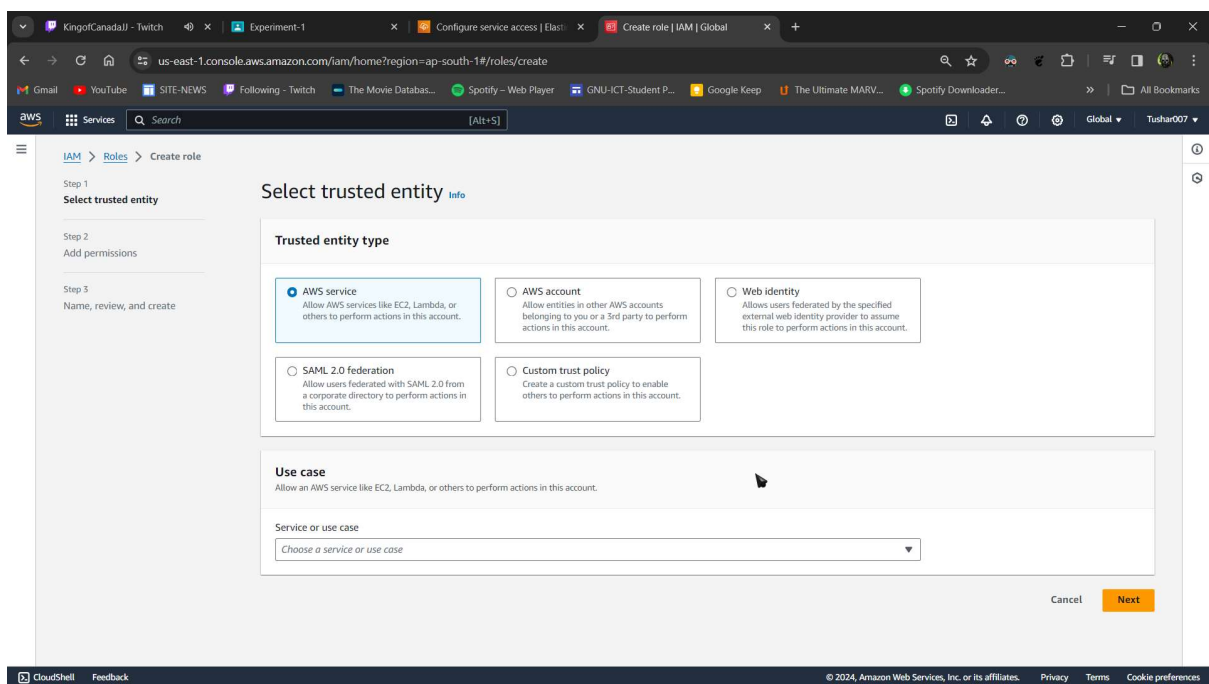
☐ Existing version (Application versions that you have uploaded.)

☐ Upload your code (Upload a source bundle from your computer or copy one from Amazon S3.)

- Selected “single instance (free tier eligible)”, so that we use free instance given by AWS :



- Now clicked on the next button and then for creating role i opened a new tab and again logged in to the AWS console and then searched for the IAM service and then opened IAM dashboard:



➤ Selected “elastic beanstalk” option in Use case:

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

Elastic Beanstalk

Choose a use case for the specified service.

Use case

☒ Elastic Beanstalk - Customizable
Allows Elastic Beanstalk to create and manage AWS resources on your behalf.

☐ Elastic Beanstalk
Allows Elastic Beanstalk to create and manage AWS resources on your behalf.

Cancel Next

➤ Gave name to the role and created the role:

us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#/roles/create?selectedUseCase=ElasticBeanstalk&trustedEntityType=AWS_SERVICE&selecte...

Services Search [Alt+S]

IAM > Roles > Create role

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Name, review, and create

Role details

Role name
Enter a meaningful name to identify this role.

EC2

Maximum 64 characters. Use alphanumeric and "+,=,_,@,-" characters.

Description
Add a short explanation for this role.

Allows Elastic Beanstalk to create and manage AWS resources on your behalf.

Maximum 1000 characters. Use alphanumeric and "+,=,_,@,-" characters.

Step 1: Select trusted entities

Trust policy

```

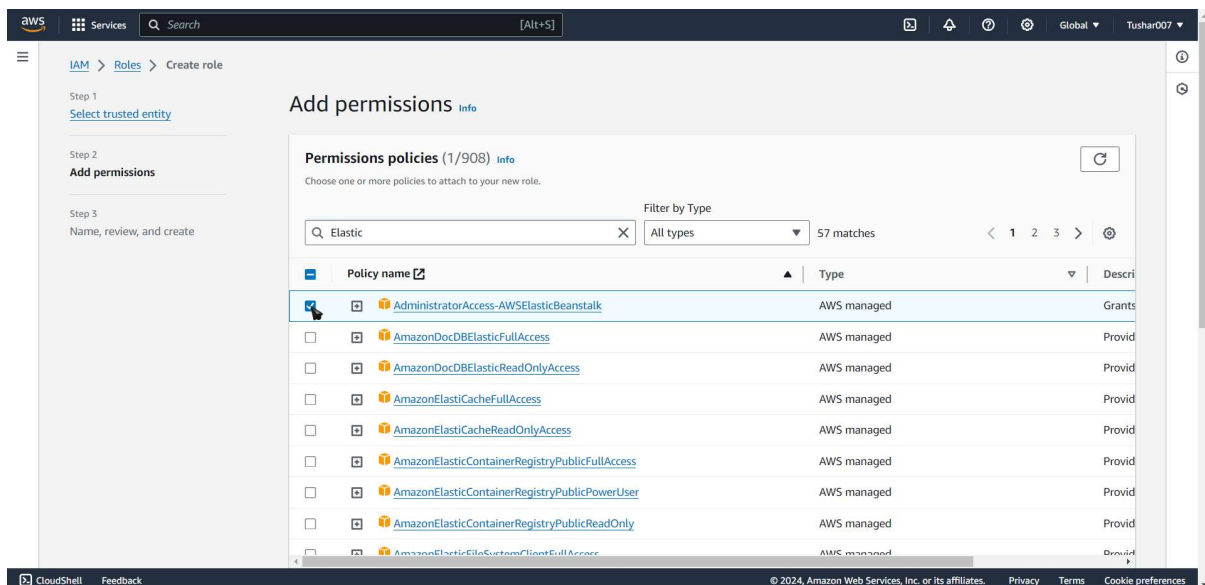
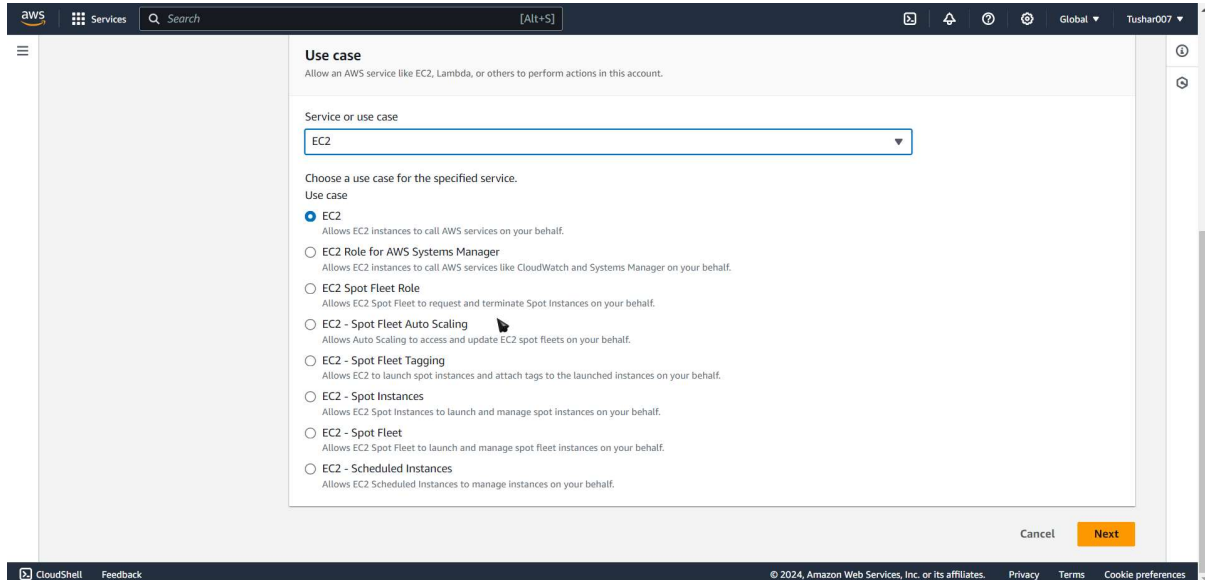
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "",
6       "Effect": "Allow",
7       "Principal": {
8         "Service": [
9           "elasticbeanstalk.amazonaws.com"
10        ]
11      }
12    ]
13  }

```

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Now creating another role of EC2 and then added the policies. Added five policies (open duplicate tab again then create new role as EC2) :



Add permissions Info

Permissions policies (2/908) Info

Choose one or more policies to attach to your new role.

Filter by Type: All types 2 matches

Policy name	Type	Description
<input type="checkbox"/> AmazonRDSEnhancedMonitoringRole	AWS managed	Provides
<input checked="" type="checkbox"/> AWSElasticBeanstalkEnhancedHealth	AWS managed	AWS Elast

► Set permissions boundary - optional

Cancel Previous Next

Add permissions Info

Permissions policies (5/908) Info

Choose one or more policies to attach to your new role.

Filter by Type: All types 30 matches

Policy name	Type	Description
<input checked="" type="checkbox"/> AdministratorAccess-AWSElasticBeanstalk	AWS managed	Grants account administrative permissions. Explicitly allows ...
<input type="checkbox"/> AWSElasticBeanstalkCustomPlatformforEC2Role	AWS managed	Provide the instance in your custom platform builder environ...
<input checked="" type="checkbox"/> AWSElasticBeanstalkEnhancedHealth	AWS managed	AWS Elastic Beanstalk Service policy for Health Monitoring s...
<input type="checkbox"/> AWSElasticBeanstalkManagedUpdatesCustomerRolePolicy	AWS managed	This policy is for the AWS Elastic Beanstalk service role used ...
<input type="checkbox"/> AWSElasticBeanstalkMultiContainerDocker	AWS managed	Provide the instances in your multicontainer Docker environ...
<input type="checkbox"/> AWSElasticBeanstalkReadOnly	AWS managed	Grants read-only permissions. Explicitly allows operators to ...
<input type="checkbox"/> AWSElasticBeanstalkRoleCore	AWS managed	AWSElasticBeanstalkRoleCore (Elastic Beanstalk operations f...
<input type="checkbox"/> AWSElasticBeanstalkRoleCWL	AWS managed	(Elastic Beanstalk operations role) Allows an environment to ...
<input type="checkbox"/> AWSElasticBeanstalkRoleECS	AWS managed	(Elastic Beanstalk operations role) Allows a multicontainer D...
<input type="checkbox"/> AWSElasticBeanstalkRoleRDS	AWS managed	(Elastic Beanstalk operations role) Allows an environment to ...
<input type="checkbox"/> AWSElasticBeanstalkRoleSNS	AWS managed	(Elastic Beanstalk operations role) Allows an environment to ...
<input checked="" type="checkbox"/> AWSElasticBeanstalkRoleWorkerTier	AWS managed	(Elastic Beanstalk operations role) Allows a worker environm...
<input checked="" type="checkbox"/> AWSElasticBeanstalkWebTier	AWS managed	Provide the instances in your web server environment access ...
<input checked="" type="checkbox"/> AWSElasticBeanstalkWorkerTier	AWS managed	Provide the instances in your worker environment access to u...
<input type="checkbox"/> AWSElasticDisasterRecoveryAgentInstallationPolicy	AWS managed	This policy allows installing the AWS Replication Agent, whic...
<input type="checkbox"/> AWSElasticDisasterRecoveryAgentPolicy	AWS managed	This policy allows using the AWS Replication Agent, which is ...
<input type="checkbox"/> AWSElasticDisasterRecoveryAgentFullAccess	AWS managed	This policy provides full access to all public APIs of AWS El...

➤ Gave name and created the role :

Name, review, and create

Role details

Role name

Enter a meaningful name to identify this role.

AWS-elasticbean-EC2-role

Maximum 64 characters. Use alphanumeric and '+', '@', '-' characters.

Description

Add a short explanation for this role.

Allows EC2 instances to call AWS services on your behalf.

Maximum 1000 characters. Use alphanumeric and '+', '@', '-' characters.

➤ Then again go to main console where we were progressing first, then select the role named "AWS-elasticbeans-EC2-role" and click next :

Configure service access [Info](#)

Service access
IAM roles, assumed by Elastic Beanstalk as a service role, and EC2 instance profiles allow Elastic Beanstalk to create and manage your environment. Both the IAM role and instance profile must be attached to IAM managed policies that contain the required permissions. [Learn more](#)

Service role
☐ Create and use new service role
☒ Use an existing service role
Existing service roles
 Choose an existing IAM role for Elastic Beanstalk to assume as a service role. The existing IAM role must have the required IAM managed policies.
 [Refresh](#)

EC2 key pair
 Select an EC2 key pair to securely log in to your EC2 instances. [Learn more](#)
 [Refresh](#)

EC2 instance profile
 Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.
 [Refresh](#)
 [Checkmark](#)

[Cancel](#) [Skip to review](#) [Previous](#) [Next](#)

- Then don't modify "setup networking, database and tags", next tab that i.e "configure instance traffic and scaling", as well as next tab "configure updates, monitoring and logging".
- Then finally reviewed all the settings and then clicked submit button.

Configure updates, monitoring, and logging - optional [Info](#)

Monitoring [Info](#)

Health reporting
 Enhanced health reporting provides free real-time application and operating system monitoring of the instances and other resources in your environment. The **EnvironmentHealth** custom metric is provided free with enhanced health reporting. Additional charges apply for each custom metric. For more information, see [Amazon CloudWatch Pricing](#).

System
☐ Basic
☒ Enhanced

CloudWatch Custom Metrics - Instance
 [Refresh](#)

CloudWatch Custom Metrics - Environment
 [Refresh](#)

Health event streaming to CloudWatch Logs
 Configure Elastic Beanstalk to stream environment health events to CloudWatch Logs. You can set the retention up to a maximum of ten years and configure Elastic Beanstalk to delete the logs when you terminate your environment.

Log streaming
☐ Activated (standard CloudWatch charges apply)

Retention

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

aws

Services

Search

[Alt+S]

Mumbai

Tushar007

Step 1
Configure environment

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

Review

Step 1: Configure environment

Environment information

Environment tier	Application name
Web server environment	Tushar_Practical1_Sample_APP
Environment name	Application code
TusharPractical1SampleAPP-env	Sample application
Platform	
arn:aws:elasticbeanstalk:ap-south-1::platform/Node.js 18 running on 64bit Amazon Linux 2023/6.0.4	

Step 2: Configure service access

Service access

Configure the service role and EC2 instance profile that Elastic Beanstalk uses to manage your environment. Choose an EC2 key pair to securely log in to your EC2 instances.

Service role	EC2 instance profile
--------------	----------------------

Command timeout	Deployment policy	Health threshold
600	AllAtOnce	Ok
Ignore health check	Instance replacement	
false	false	
Platform software		
Lifecycle	Log streaming	Proxy server
false	Deactivated	nginx
Logs retention	Rotate logs	Update level
7	Deactivated	minor
X-Ray enabled		
Deactivated		
Environment properties		
Key		
Value		
No environment properties		
There are no environment properties defined		

Cancel Previous Submit

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences

➤ Finally we have created the sample application.

The screenshot shows the AWS Elastic Beanstalk console. A green banner at the top states "Environment successfully launched." The main content area displays the details for the environment "TusharPractical1SampleAPP-env".

Environment overview:

- Health: Ok
- Environment ID: e-zwwiugexv3
- Domain: TusharPractical1SampleAPP-env.eba-acnamitw.ap-south-1.elasticbeanstalk.com
- Application name: Tushar_Practical1_Sample_APP

Platform:

- Platform: Node.js 18 running on 64bit Amazon Linux 2023/6.0.4
- Running version: -
- Platform state: Supported

Events (11):

Filter events by text, property or value

The screenshot shows a "Congratulations" message on a green background. The text reads:

Congratulations

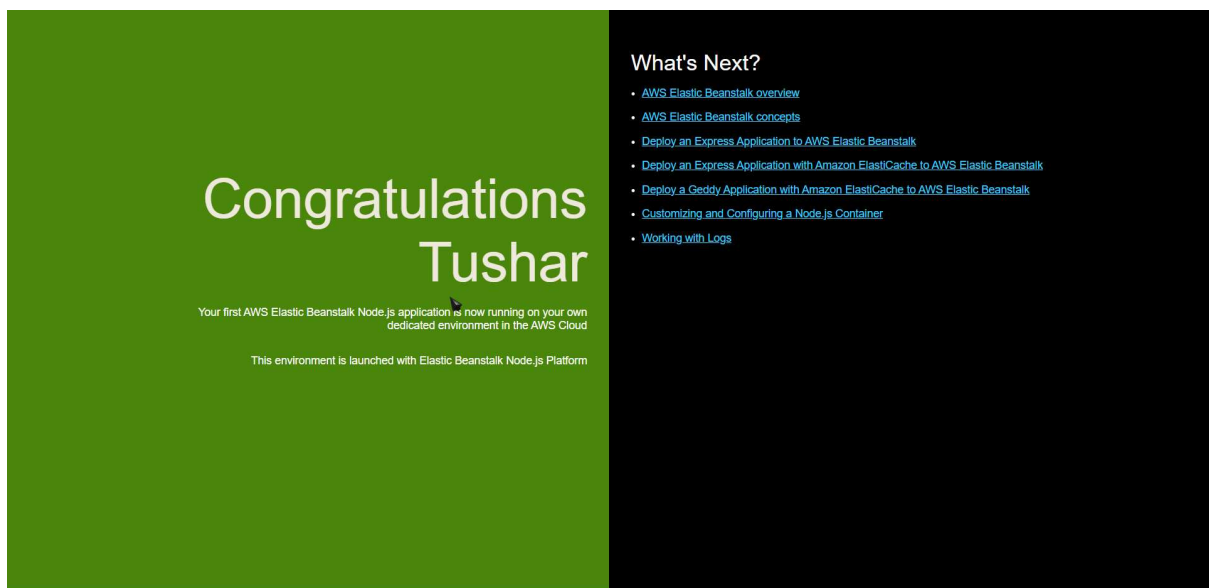
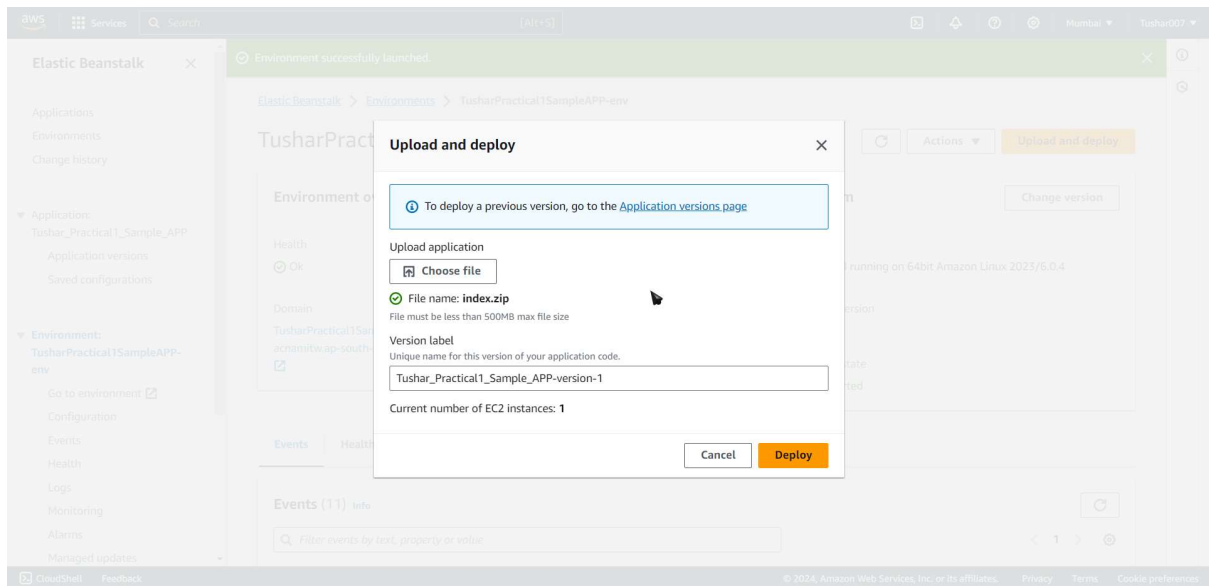
Your first AWS Elastic Beanstalk Node.js application is now running on your own dedicated environment in the AWS Cloud

This environment is launched with Elastic Beanstalk Node.js Platform

What's Next?

- [AWS Elastic Beanstalk overview](#)
- [AWS Elastic Beanstalk concepts](#)
- [Deploying an Express Application to AWS Elastic Beanstalk](#)
- [Deploying an Express application with clustering to Elastic Beanstalk](#)
- [Customizing and Configuring a Node.js Container](#)
- [Working with Logs](#)

- After that i modified the html file and again deployed and uploaded in the instance again. Here's the output of same:

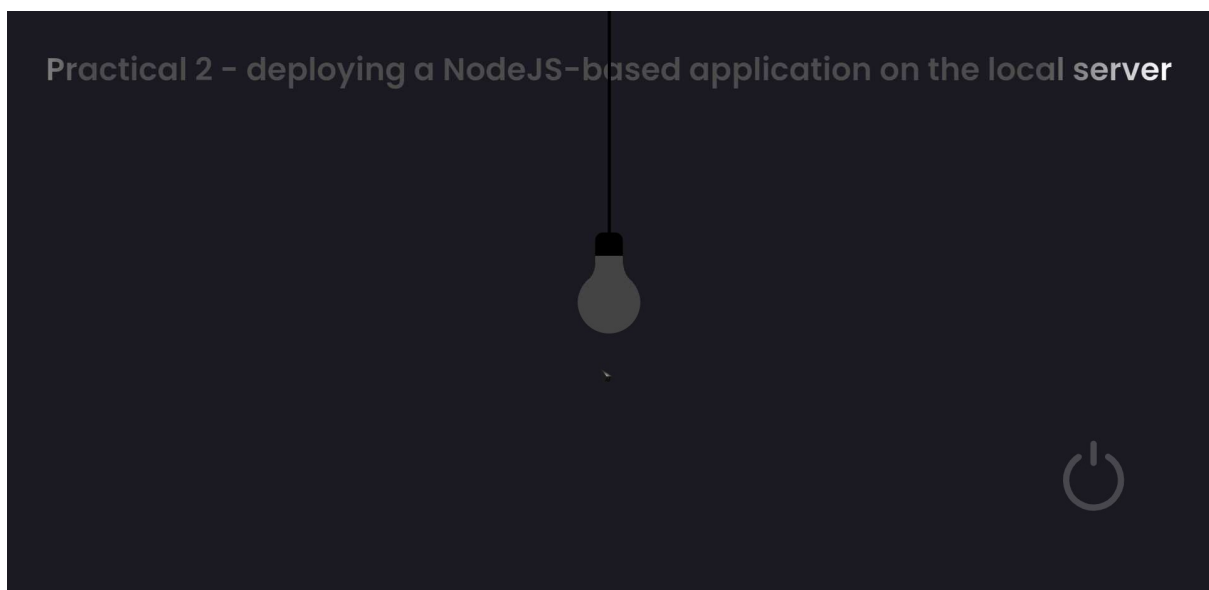


➡ **1.1: Developing :**

NodeJS application to create any website GUI (use HTML, CSS, JavaScript) using AWS Cloud and AWS Elastic beanstalk Sample GUI should contain two buttons, with one button to turn ON the bulb provided in the user interface and a second button to turn OFF the bulb. :

» **I already have files of this practical so I will create zip file of it and then I created new application for it as by following sample application's steps .**

» **Output :**



👉 **Domain Link :**

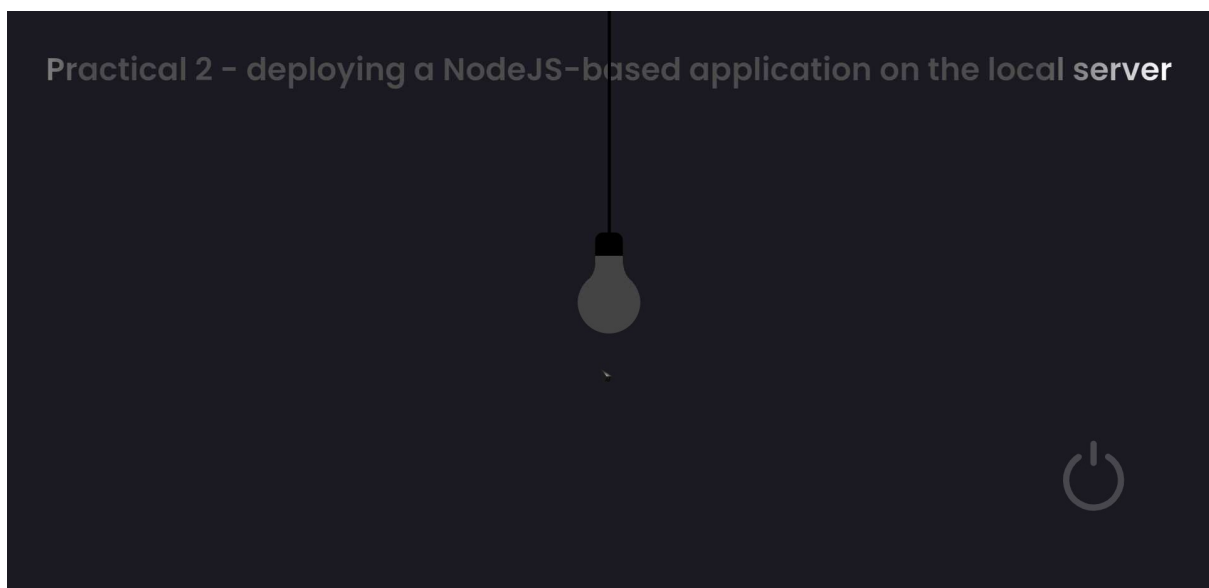
<http://tusharp1new-env.eba-7pft22fz.ap-south-1.elasticbeanstalk.com/>

➡ **1.2: Developing NodeJS :**

application to create website GUI (use HTML, CSS, JavaScript) using AWS Cloud and AWS Elastic Beanstalk to Click on the light bulb to turn on/off the light :

» **I already have files of this practical so I will create zip file of it and then I created new application for it as by following sample application's steps .**

» **Output :**



👉 **Domain Link :**

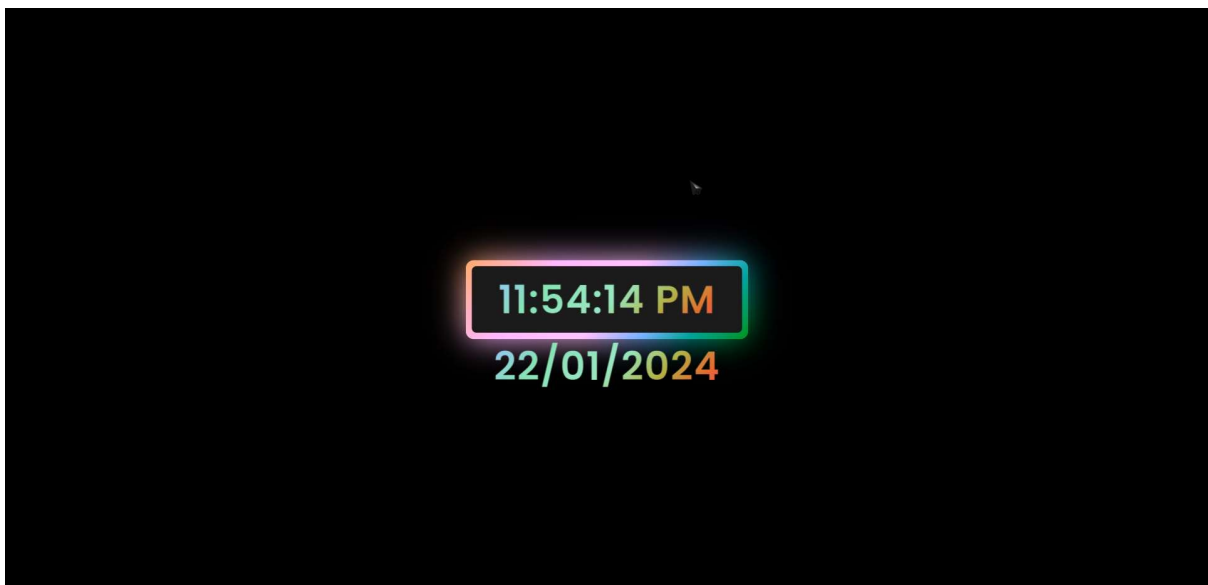
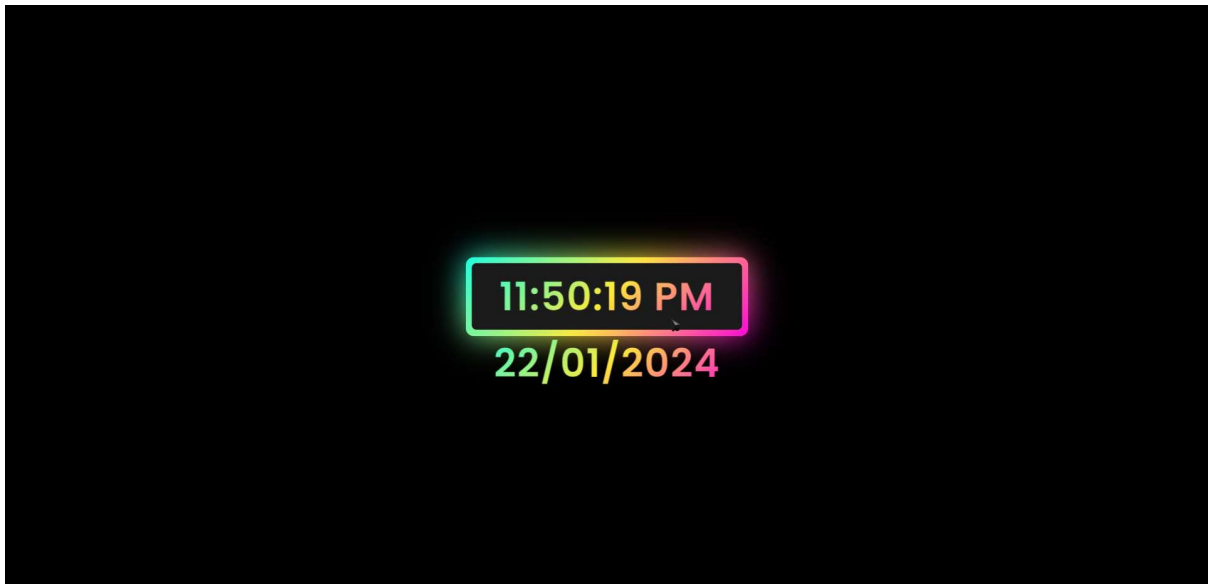
<http://tusharp1new-env.eba-7pft22fz.ap-south-1.elasticbeanstalk.com/>

➡ **1.3 Develop HTTP Module :**

World NodeJS application on the AWS Cloud using the HTTP module create your module and include it in the NodeJS application to print the current date and time:

» **I already have files of this practical so I will create zip file of it and then I created new application for it as by following sample application's steps.**

» **Output :**



👉 **Domain Link :**

<http://tusharpractical1task3-env.eba-m83erkgd.ap-south-1.elasticbeanstalk.com/>

» **Here's the screenshot of all the applications running on different instances:**

The screenshot displays the AWS Elastic Beanstalk 'Environments' page. The top section shows a summary of three environments, all with a health status of 'Ok'. Below this, a detailed table lists the environments with their names, health, application names, platforms, domains, runtimes, and tiers.

Environment name	Health	Applica...	Platform	Domain	Runnin...	Tier na...
TusharP1NEW-env	Ok	Tushar_P...	Node.js 1...	TusharP1NEW-env.eba-7pft22...	Tushar_P...	WebServer
TusharPractical1SampleAPP-env	Ok	Tushar_Pr...	Node.js 1...	TusharPractical1SampleAPP-e...	Tushar_Pr...	WebServer
TusharPractical1Task3-env	Ok	Tushar_Pr...	Node.js 1...	TusharPractical1Task3-env.eba...	Tushar_Pr...	WebServer

The bottom section of the screenshot shows the full AWS console interface, including the left-hand navigation menu with 'Elastic Beanstalk' selected, and the main content area displaying the same 'Environments' table. The footer of the console shows 'CloudShell', 'Feedback', and copyright information for Amazon Web Services, Inc. (© 2024).