

# Cloud Computing

## Prac 2: Platform as a service using AWS.

### Writeup

- Platform as a service:

Platform as a Service (PaaS) provides a runtime environment. It allows programmers to easily create, test, run, and deploy web applications. You can purchase these applications from a cloud service provider on a pay-as-per use basis and access them using the Internet connection. In PaaS, back end scalability is managed by the cloud service provider, so end- users do not need to worry about managing the infrastructure.

PaaS includes infrastructure (servers, storage, and networking) and platform (middleware, development tools, database management systems, business intelligence, and more) to support the web application life cycle.

Example: Google App Engine, Force.com, Joyent, Azure.

- Elastic beanstalk:

AWS Elastic Beanstalk is an AWS-managed service for web applications. Elastic Beanstalk is a pre-configured EC2 server that can directly take up your application code and environment configurations and use it to automatically provision and deploy the required resources within AWS to run the web application. Unlike EC2 which is Infrastructure as a service, Elastic Beanstalk is a Platform As A Service (PAAS) as it allows users to directly use a pre-configured server for their application. Of course, you can deploy applications without ever having to use elastic beanstalk but that would mean having to choose the appropriate service from the vast array of services offered by AWS, manually provisioning these AWS resources, and stitching them up together to form a complete web application. Elastic Beanstalk abstracts the underlying configuration work and allows you as a user to focus on more pressing matters.

This raises a concern that if elastic Beanstalk configures most of the resources itself and abstracts the underlying details. Can developers change the configuration if needed? The answer is Yes. Elastic Beanstalk is provided to make application deployment simpler but at no level will it restrict the developers from changing any configurations.

- Components of Beanstalk:

### **1. Deployment**

Elastic Beanstalk, for the most part, simplifies the process of deploying an application on the Amazon cloud. The service allows developers the ability to upload and manage different versions of their apps, and switch between them in different environments like development, test, and production.

### **2. Application**

An application in Elastic Beanstalk is basically a collection of environments, versions, and everything else related to them, like events. In other words, an Elastic Beanstalk application is conceptually similar to a folder. Most users normally create a separate EB application for each of their applications, and although this is not required, it does help streamline management.

### **3. Version**

A version is the deployable code of an application. Depending on your programming platform of choice, you will have a file, or a set of files that you upload, with a label and description. You can then see where it is deployed, in which environment, and even download the file or files, if needed.

### **4. Environment**

As you may have guessed, an environment is a deployed version of specific instances, load balancers, and scaling groups, etc. A typical workflow is creating one environment for testing, and another for production. Though you can, of course, create as many as you need, as much as your budget allows. Amazon provides access to your environment via a specific URL, and provides different health status so you can quickly get an idea of how things are up there. Green is okay, yellow when your environment has not responded within the last 5 minutes, red if it hasn't responded for more than 5 minutes, and grey, unknown.

### **5. Events**

Events tell you what is going on with your environment. They are either informative, warnings, or errors, letting you know details like when an environment successfully launches, or an instance is close to utilising its resources. You can view the events in a web console, or have them sent to you via email.

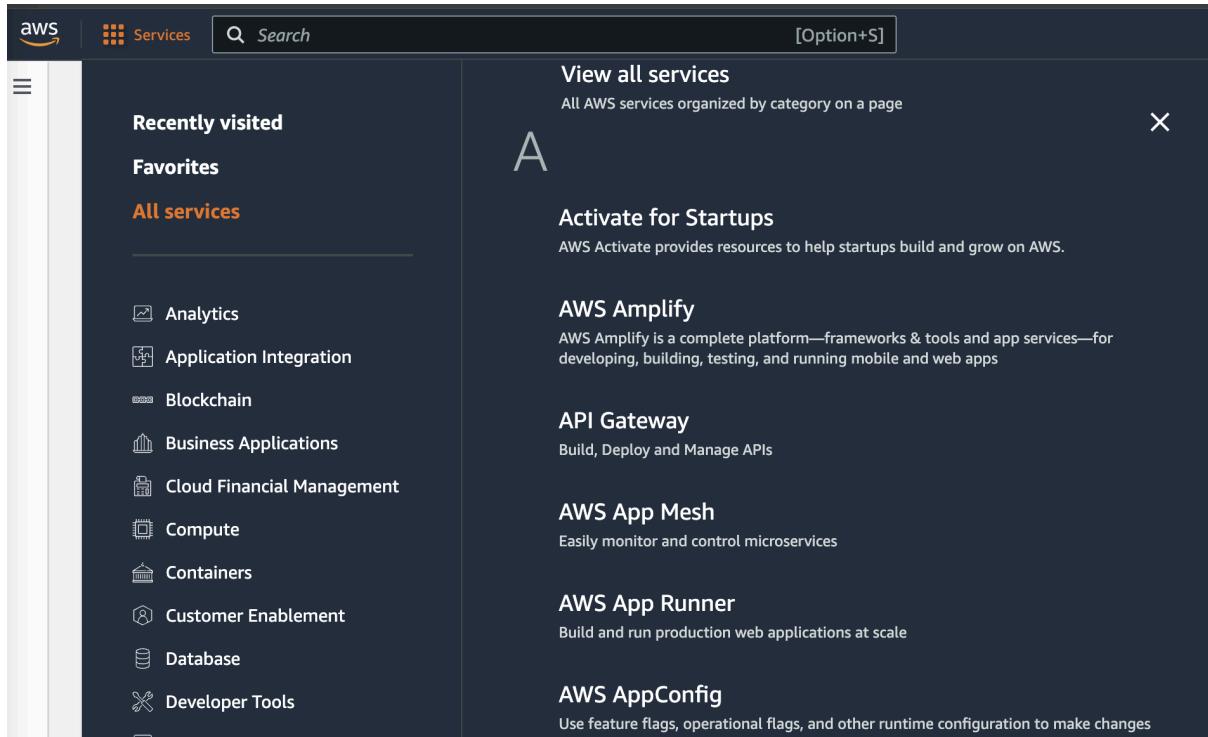
- IAM:

IAM is a combination of policies and technologies that allows organisations to identify users and provide the right form of access as and when required. There has been a burst in the market with new applications, and the requirement for an organisation to use these applications has increased drastically. The services and resources you want to access can be specified in IAM. IAM doesn't provide any replica or backup. IAM can be used for many purposes such as, if one wants to control access of individual and group access for your AWS resources. With IAM policies, managing permissions to your workforce and systems to ensure least-privilege permissions becomes easier. The AWS IAM is a global service.

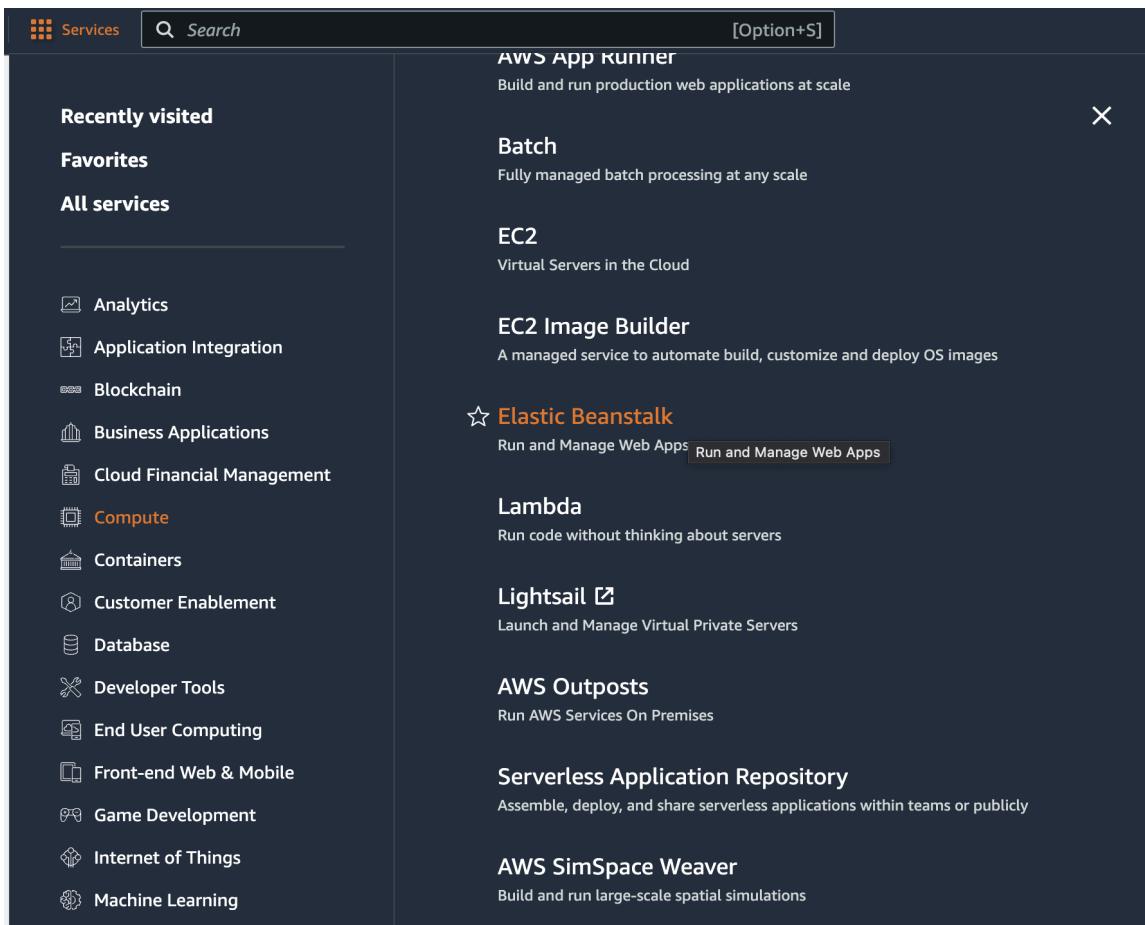
## Implement PaaS using elastic beanstalk for Python

1. Sign in to your AWS account.

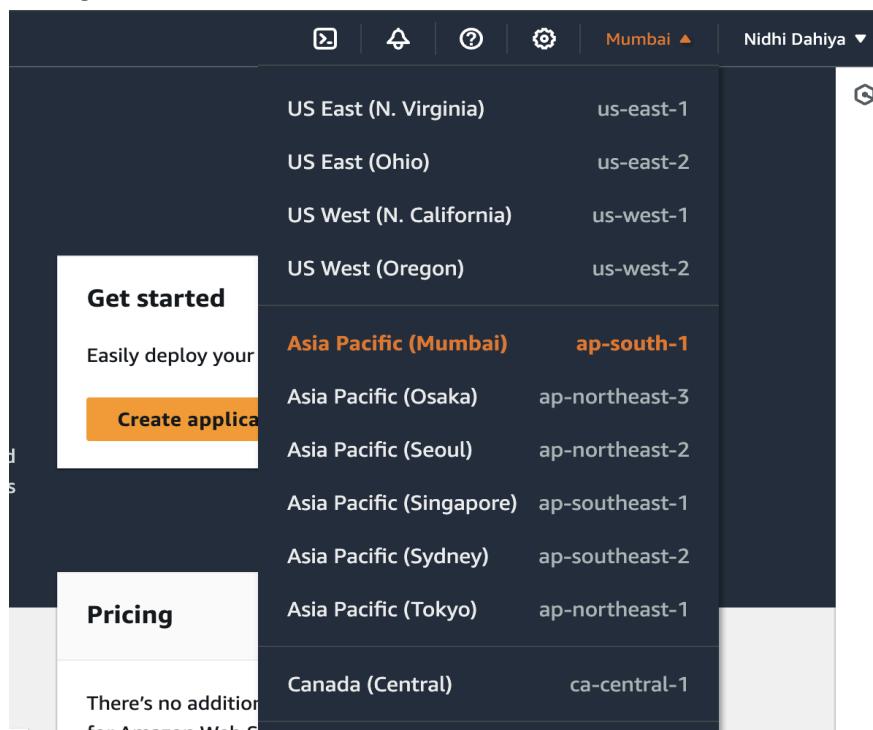
2. Select All Services.



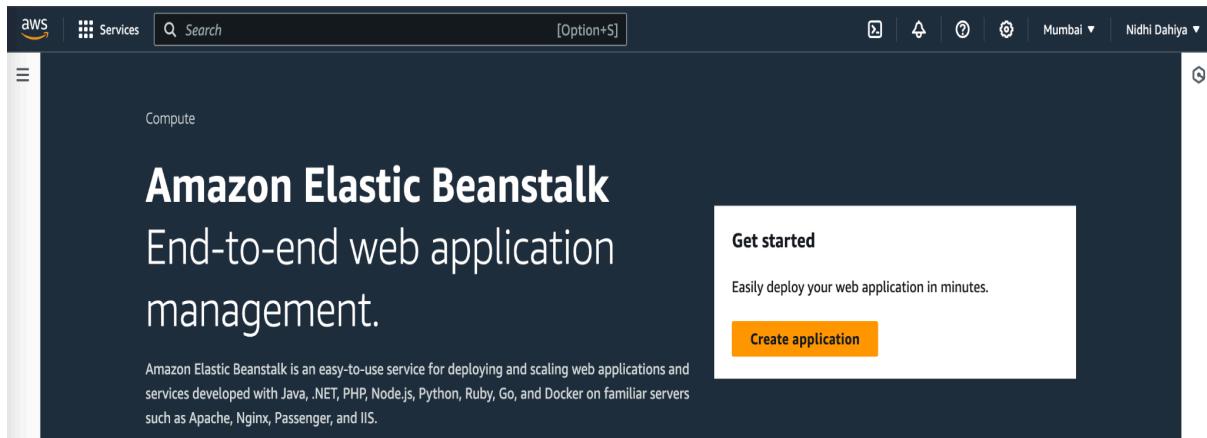
### 3. Select Elastic Beanstalk.



### 4. Select region Mumbai.



5. Click on create application.



6. Click on the web server environment.

A screenshot of the 'Configure environment' step in the AWS Elastic Beanstalk wizard. The title 'Configure environment' is at the top, followed by a link 'Info'. Below it, there's a section titled 'Environment tier' with a 'Info' link. It states that Amazon Elastic Beanstalk has two types of environment tiers to support different types of web applications. Two options are shown: 'Web server environment' (selected, indicated by a blue radio button) and 'Worker environment' (indicated by an empty radio button). Each option has a brief description and a 'Learn more' link with a blue icon.

7. Give the application name and description. (give the domain or else it'll be given by default)

**Application information** Info

Application name

Maximum length of 100 characters.

► Application tags (optional)

**Environment information** Info

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
 Check availability

Environment description

My sample application to demonstrate PaaS.

8. Choose the platform i.e, 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

Python

Platform branch

Python 3.11 running on 64bit Amazon Linux 2023

Platform version

4.0.8 (Recommended)

9. Select sample application (no code to upload), and single instance.

**Application code** [Info](#)

- 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.

**Presets** [Info](#)

Start from a preset that matches your use case or choose custom configuration to unset recommended values and use the service's default values.

Configuration presets

- Single instance (free tier eligible)
- Single instance (using spot instance)
- High availability
- High availability (using spot and on-demand instances)
- Custom configuration

[Cancel](#) [Next](#)

10. Click on next.

11. Click on create a new service role.

**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

**Service role name**

Enter the name for an IAM role that Elastic Beanstalk will create to assume as a service role. Beanstalk will attach the required managed policies to it.

12. For creating a new role, Go to IAM in ALL SERVICES. (use a new tab)

The screenshot shows the AWS Services dashboard. On the left, there's a sidebar with 'Recently visited' (Amazon Honeycode), 'Favorites' (empty), and 'All services'. Below that is a list of various AWS services with icons: Analytics, Application Integration, Blockchain, Business Applications, Cloud Financial Management, Compute, Containers, Customer Enablement, Database, Developer Tools, End User Computing, Front-end Web & Mobile, and Game Development. On the right, the 'IAM' service is highlighted with a star icon and the text 'Manage access to AWS resources'. Other services listed include Amazon Honeycode, IAM Identity Center, Incident Manager, Amazon Inspector, Amazon Interactive Video Service, and IoT 1-Click.

13. Click on roles and then create role.

The screenshot shows the 'Roles' page under the 'Identity and Access Management (IAM)' service. The left sidebar has 'Access management' expanded, showing 'User groups', 'Users', and 'Roles'. The 'Roles' section is selected. The main area shows a table of roles with columns for 'Role name', 'Trusted entities', and 'Last activity'. Two roles are listed: 'AWSServiceRoleForSupport' (trusted by 'AWS Service: support (Service-Linker)') and 'AWSServiceRoleForTrustedAdvisor' (trusted by 'AWS Service: trustedadvisor (Service-Linker)'). A 'Create role' button is located at the top right of the table. Below the table is a section titled 'Roles Anywhere' with a 'Manage' button.

## 14. Click on AWS service.

IAM > Roles > Create role

Step 1  
Select trusted entity

Step 2  
Add permissions

Step 3  
Name, review, and create

Select trusted entity Info

**Trusted entity type**

- AWS service**  
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
- AWS account**  
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- Web identity**  
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
- SAML 2.0 federation**  
Allows users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- Custom trust policy**  
Create a custom trust policy to enable others to perform actions in this account.

## 15. Select EC2 and then next.

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

**Service or use case**

EC2

Choose a use case for the specified service.

**Use case**

- EC2**  
Allows EC2 instances to call AWS services on your behalf.
- EC2 Role for AWS Systems Manager**  
Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.
- EC2 Spot Fleet Role**  
Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.
- EC2 - Spot Fleet Auto Scaling**  
Allows Auto Scaling to access and update EC2 spot fleets on your behalf.
- EC2 - Spot Fleet Tagging**  
Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.
- EC2 - Spot Instances**  
Allows EC2 Spot Instances to launch and manage spot instances on your behalf.
- EC2 - Spot Fleet**  
Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.
- EC2 - Scheduled Instances**  
Allows EC2 Scheduled Instances to manage instances on your behalf.

Cancel **Next**

16. Search for beanstalk rules, and select following 3 and next.

Filter by Type			
<input type="text"/> beanstalk	<input type="button"/>	All types	14 matches
	Policy name	Type	Description
<input type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AdministratorAccess-AWSEI...</a>	AWS managed	Grants account administrative permis...
<input type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkCustom...</a>	AWS managed	Provide the instance in your custom pl...
<input type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkEnhanc...</a>	AWS managed	AWS Elastic Beanstalk Service policy f...
<input type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkManag...</a>	AWS managed	This policy is for the AWS Elastic Bean...
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkMultico...</a>	AWS managed	Provide the instances in your multicon...
<input type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkReadOnly</a>	AWS managed	Grants read-only permissions. Explicitl...
<input type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkRoleCore</a>	AWS managed	AWSElasticBeanstalkRoleCore (Elastic ...
<input type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkRoleCWL</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkRoleECS</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkRoleRDS</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkRoleSNS</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkRoleWo...</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkWebTier</a>	AWS managed	Provide the instances in your web serv...
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">AWSElasticBeanstalkWorker...</a>	AWS managed	Provide the instances in your worker e...

17. Name the role.

### Name, review, and create

<b>Role details</b>
<p><b>Role name</b>            Enter a meaningful name to identify this role.  <input type="text" value="beanstalk-role"/>  <small>Maximum 64 characters. Use alphanumeric and '+,-,@,_' characters.</small> </p> <p><b>Description</b>            Add a short explanation for this role.  <input type="text" value="Allows EC2 instances to call AWS services on your behalf."/>  <small>Maximum 1000 characters. Use alphanumeric and '+,-,@,_' characters.</small> </p>
<b>Step 1: Select trusted entities</b>
<p><b>Trust policy</b></p> <pre> 1  "Version": "2012-10-17", 2  "Statement": [ 3    { 4      "Effect": "Allow", 5      "Action": [ 6        ... </pre>

18. Click on create role.

The screenshot shows the 'Step 2: Add permissions' section of the AWS IAM 'Create New Role' wizard. At the top, there's a table titled 'Permissions policy summary' with three rows:

Policy name	Type	Attached as
<a href="#">AWSElasticBeanstalkMulticontainerDocker</a>	AWS managed	Permissions policy
<a href="#">AWSElasticBeanstalkWebTier</a>	AWS managed	Permissions policy

Below the table, there's a section titled 'Step 3: Add tags' with a note: 'Add tags - optional' and a message: 'Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.' It shows 'No tags associated with the resource.' and a button 'Add new tag'. A note below says 'You can add up to 50 more tags.' At the bottom right are buttons for 'Cancel', 'Previous', and 'Create role' (which is highlighted).

19. Role created.

The screenshot shows the 'Roles' list screen in the AWS IAM console. At the top, a green banner says 'Role beanstalk-role created.' with buttons for 'View role' and 'X'. Below it, the 'IAM > Roles' navigation path is shown. The main table has columns: 'Role name', 'Trusted entities', and 'Last activity'. It lists three roles:

Role name	Trusted entities	Last activity
<a href="#">AWSServiceRoleForSupport</a>	AWS Service: support (Service-Linked)	-
<a href="#">AWSServiceRoleForTrustedAdvisor</a>	AWS Service: trustedadvisor (Service-Linked)	-
<a href="#">beanstalk-role</a>	AWS Service: ec2	-

20. Go to EC2, click on the key pairs and create a key pair.

The screenshot shows the AWS Management Console navigation bar on the left, with 'Key pairs' selected under 'Network & Security'. The main area is titled 'Key pairs' with an 'Info' link. It features a search bar labeled 'Find Key Pair by attribute or tag'. Below the search bar is a table header with columns: Name, Type, Created, Fingerprint, and ID. A message 'No key pairs to display' is centered below the table. At the top right of the main area are 'Actions' and 'Create key pair' buttons.

21. Create a key pair.

The screenshot shows the 'Create key pair' wizard. The first step, 'Key pair', is completed. The second step, 'Name', has 'sample-key-pair' entered. The third step, 'Key pair type', has 'RSA' selected. The fourth step, 'Private key file format', has '.ppk' selected. The fifth step, 'Tags - optional', shows a button 'Add new tag' and a note about adding up to 50 more tags. The final step, 'Create key pair', is at the bottom right.

22. Go to the previous tab, and refresh and select the role and key pair and click on 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

**Service role name**  
 Enter the name for an IAM role that Elastic Beanstalk will create to assume as a service role. Beanstalk will attach the required managed policies to it.  
 aws-elasticbeanstalk-service-role

[View permission details](#)

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

sample-key-pair [▼](#) [C](#)

**EC2 instance profile**  
 Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.

beanstalk-role [▼](#) [C](#)

[View permission details](#)

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

23. Select VPC.

### Set up networking, database, and tags - *optional* Info

**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-025f7c4e1bc4e0cba | (172.31.0.0/16) [▼](#)

[Create custom VPC](#)

24. Select the same IP address in instance as of VPC (172.31.0.0).

The screenshot shows the AWS VPC configuration interface. At the top, there is a dropdown menu with the option "vpc-025f7c4e1bc4e0cba | (172.31.0.0/16)". Below it is a link "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**

	Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/>	ap-south-1c	subnet-018470deb...	172.31.16.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-040070104...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1a	subnet-0dd23cf18...	172.31.32.0/20	

25. Select the same IP address in the database as of VPC.

**Database Info**

Integrate an RDS SQL database with your environment. [Learn more](#)

**Database subnets**

If your Elastic Beanstalk environment is attached to an Amazon RDS, choose subnets for your database instances. [Learn more](#)

**Choose database subnets (3)**

	Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/>	ap-south-1c	subnet-018470deb...	172.31.16.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-040070104...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1a	subnet-0dd23cf18...	172.31.32.0/20	

Enable database

**Restore a snapshot - optional**

Restore an existing snapshot from a previously used database.

Snapshot

26. Click on next.

27. Set as default, no changes and next.

## Configure instance traffic and scaling - *optional* Info

### ▼ Instances Info

Configure the Amazon EC2 instances that run your application.

#### Root volume (boot device)

##### Root volume type

##### Size

The number of gigabytes of the root volume attached to each instance.

 GB

##### IOPS

Input/output operations per second for a provisioned IOPS (SSD) volume.

 IOPS

##### Throughput

The desired throughput to provision for the Amazon EBS root volume attached to your environment's EC2 instance

 MiB/s

#### Amazon CloudWatch monitoring

The time interval between when metrics are reported from the EC2 instances

##### Monitoring interval

 ▾

28. Set as default, no changes and next.

### 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

[Choose metrics](#)

##### CloudWatch Custom Metrics - Environment

[Choose metrics](#)

##### 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

7

29. Review.

### Review Info

#### Step 1: Configure environment

[Edit](#)

##### Environment information

Environment tier	Application name
Web server environment	test-web-application
Environment name	Application code
Test-web-application-env	Sample application

Platform  
arn:aws:elasticbeanstalk:ap-south-1::platform/Python  
3.11 running on 64bit Amazon Linux 2023/4.0.8

#### Step 2: Configure service access

[Edit](#)

##### Service access Info

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 key pair	EC2 instance profile
arn:aws:iam::544606390347:role/service-role/aws-elasticbeanst	sample-key-pair	beanstalk-role

### 30. Submit.

Command timeout	Deployment policy	Health threshold				
600	AllAtOnce	Ok				
Ignore health check	Instance replacement					
false	false					
<b>Platform software</b>						
Lifecycle	Log streaming	NumProcesses				
false	Deactivated	1				
NumThreads	WSGIPath	Proxy server				
15	application	nginx				
Logs retention	Rotate logs	Update level				
7	Deactivated	minor				
X-Ray enabled						
Deactivated						
<b>Environment properties</b>						
<table border="1"> <thead> <tr> <th>Key</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>PYTHONPATH</td> <td>/var/app/venv/staging-LQM1lest/bin</td> </tr> </tbody> </table>			Key	Value	PYTHONPATH	/var/app/venv/staging-LQM1lest/bin
Key	Value					
PYTHONPATH	/var/app/venv/staging-LQM1lest/bin					
<a href="#">Cancel</a> <a href="#">Previous</a> <a href="#" style="background-color: orange; color: white; border: 1px solid orange;">Submit</a>						

### 31. Launching.

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

[Elastic Beanstalk](#) > [Environments](#) > Test-web-application-env

**Test-web-application-env** [Info](#) [C](#) [Actions ▾](#) [Upload and deploy](#)

<b>Environment overview</b>	<b>Platform</b> <a href="#">Change version</a>
Health Unknown	Platform Python 3.11 running on 64bit Amazon Linux 2023/4.0.8
Domain -	Running version -
	Platform state Supported

[Events](#) [Health](#) [Logs](#) [Monitoring](#) [Alarms](#) [Managed updates](#) [Tags](#)

### 32. Environment is created.

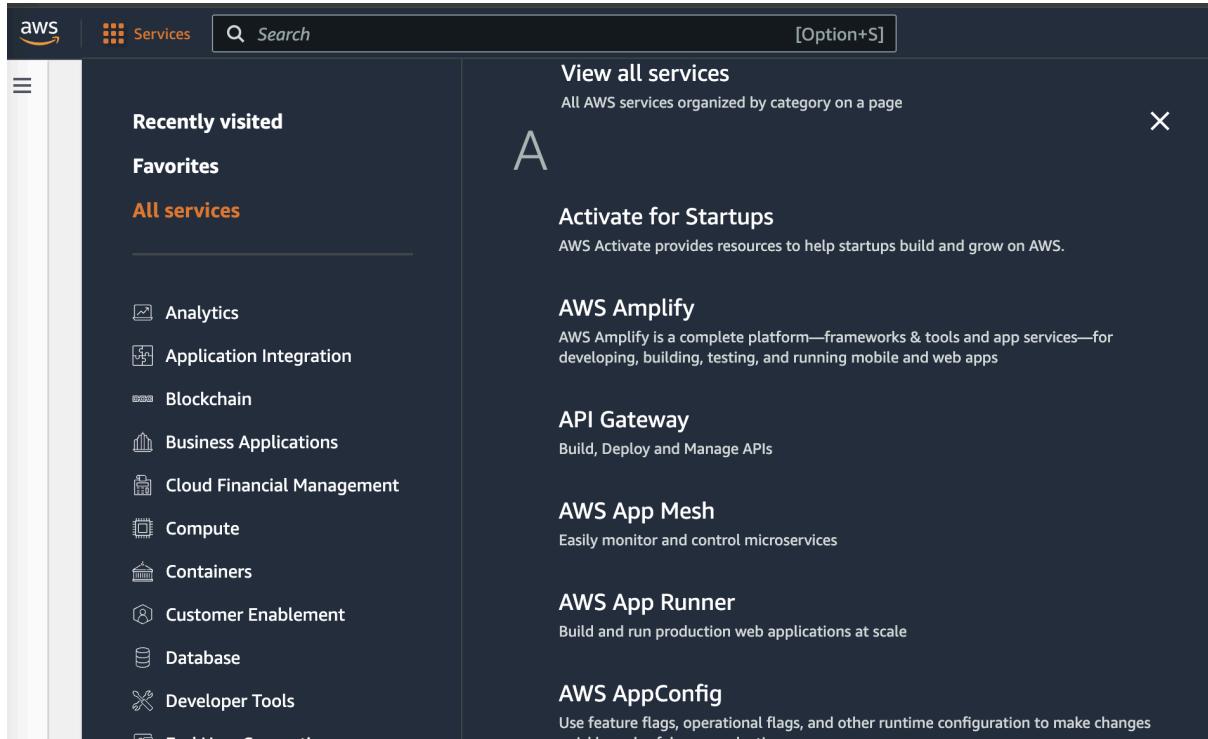
The screenshot shows the AWS Elastic Beanstalk interface. On the left, a sidebar lists 'Applications', 'Environments' (which is selected), and 'Change history'. Below this is a 'Recent environments' section containing 'Test-web-application-env'. The main content area has a green header bar with the text 'Environment successfully launched.' and a circular icon. Below this is a breadcrumb trail: 'Elastic Beanstalk > Environments'. The main table is titled 'Environments (1) Info' and includes columns for 'Environment name', 'Health', 'Application', 'Platform', 'Domain', and 'Running...'. A single row is shown for 'Test-web-application-env', which is marked as 'Ok' with a green status indicator. At the top right of the table are 'Actions' and 'Create environment' buttons.

The screenshot shows a 'Congratulations' message on the left side of the screen. It says: 'Your first AWS Elastic Beanstalk Python Application is now running on your own dedicated environment in the AWS Cloud' and 'This environment is launched with Elastic Beanstalk Python Platform'. To the right, under the heading 'What's Next?', there is a list of links: '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', and 'Working with Logs'.

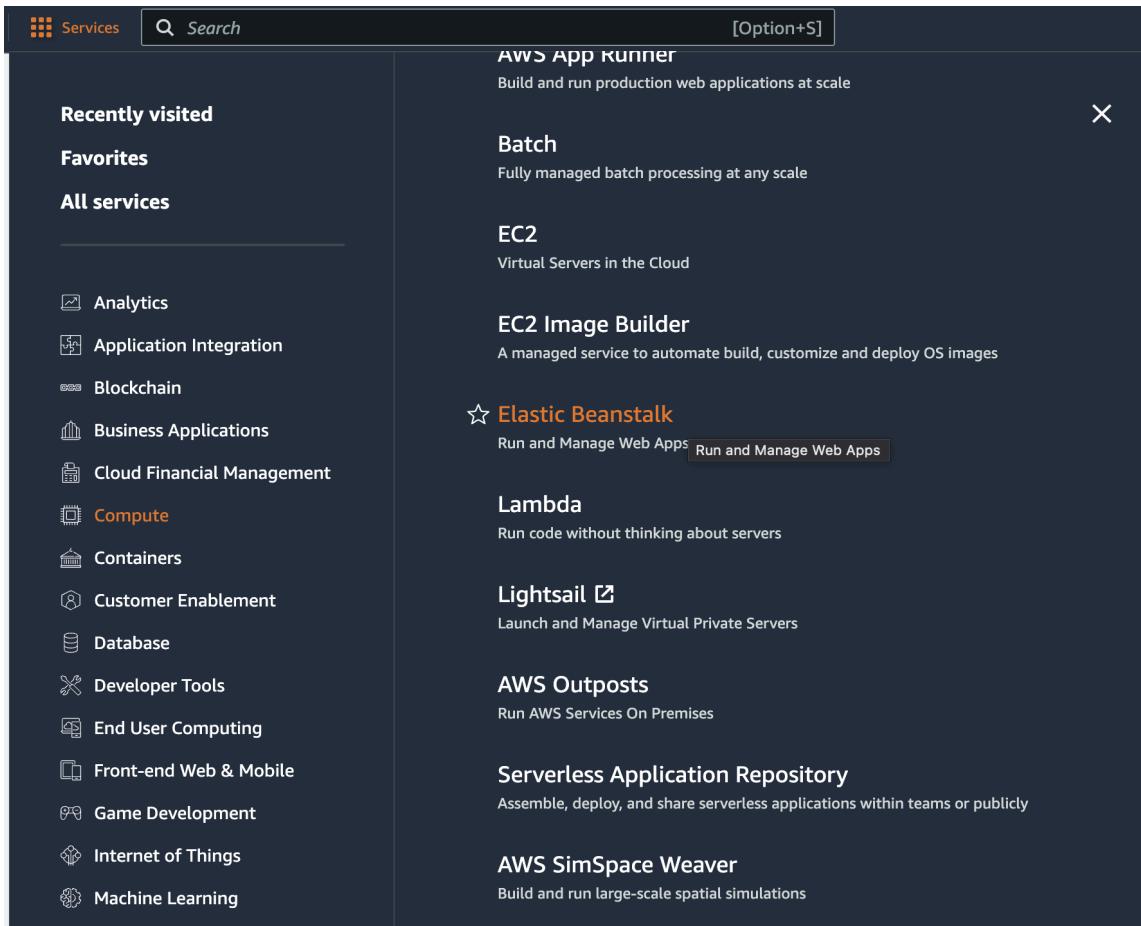
## Implement PaaS using elastic beanstalk for Java

1. Sign in to your AWS account.

2. Select All Services.



### 3. Select Elastic Beanstalk.



### 4. Click on create application.

The screenshot shows the 'Applications' page in the AWS Elastic Beanstalk console. The top navigation bar includes 'Elastic Beanstalk > Applications'. The main area displays a table titled 'Applications (1) Info'. The table has columns for 'Application name', 'Environments', 'Date created', and 'Last modified'. A single row is shown for the application 'test-web-application', which has an environment 'Test-web-application-env'. The 'Date created' is 'January 29, 2024 20:01:40' and the 'Last modified' is 'January 29, 2024 20:01:40'. The 'Actions' dropdown menu and a 'Create application' button are visible at the top right of the table.

## 5. Name the application.

**Configure environment** Info

**Environment tier** Info

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. [Learn more](#)
- Worker environment**  
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule. [Learn more](#)

**Application information** Info

**Application name**  
  
Maximum length of 100 characters.

**► Application tags (optional)**

**Environment information** Info

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

**Environment name**

## 6. Give the description.

**Environment information** Info

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**

**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**

## 7. Choose the platform.

**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

Java

Platform branch

Corretto 21 running on 64bit Amazon Linux 2023

Platform version

4.2.0 (Recommended)

## 8. Choose Upload your code and download a java code war file and upload the same.

**Application code Info**

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.

Version label  
Unique name for this version of your application code.  
sample-java

Source code origin. Maximum size 500 MB

Local file

Upload application

 File name: **SampleWebApp.war**  
File must be less than 500MB max file size

Public S3 URL

**Presets Info**

Start from a preset that matches your use case or choose custom configuration to unset recommended values and use the service's default values.

Configuration presets

Single instance (free tier eligible)

9. Click on next.

10. Click on create a new service role.

## 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

[Create role now](#)

11. For creating a new role, Go to IAM in ALL SERVICES. (use a new tab)

The screenshot shows the AWS Services navigation bar at the top with 'Services' selected. Below it is a search bar and a keyboard shortcut '[Option+S]'. The main content area displays a list of services under 'All services'. The 'IAM' service is highlighted with a star icon and a callout box containing the text 'Manage access to AWS resources'. Other services listed include Analytics, Application Integration, Blockchain, Business Applications, Cloud Financial Management, Compute, Containers, Customer Enablement, Database, Developer Tools, End User Computing, Front-end Web & Mobile, and Game Development. Each service has a small icon and a brief description below it.

12. Click on roles and then create role.

The screenshot shows the AWS IAM 'Roles' page. At the top, there's a breadcrumb navigation from 'IAM' to 'Roles'. Below it, a header says 'Roles (5) Info' with a 'Delete' button and an orange 'Create role' button. A search bar and a page navigation area with a single item are also present. The main content area lists five roles, each with a small icon, a name, and a 'View details' link.

13. Select EC2 and next.

The screenshot shows the 'Create New Role' wizard, Step 1: Set permissions. It has a 'Use case' section with a note about allowing AWS services like EC2, Lambda, or others to perform actions. Below is a 'Service or use case' dropdown set to 'EC2'. The 'Use case' section contains several options, each with a description:

- EC2**  
Allows EC2 instances to call AWS services on your behalf.
- EC2 Role for AWS Systems Manager**  
Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.
- EC2 Spot Fleet Role**  
Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.
- EC2 - Spot Fleet Auto Scaling**  
Allows Auto Scaling to access and update EC2 spot fleets on your behalf.
- EC2 - Spot Fleet Tagging**  
Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.
- EC2 - Spot Instances**  
Allows EC2 Spot Instances to launch and manage spot instances on your behalf.
- EC2 - Spot Fleet**  
Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.
- EC2 - Scheduled Instances**  
Allows EC2 Scheduled Instances to manage instances on your behalf.

At the bottom right are 'Cancel' and 'Next' buttons.

14. Select the following and next.

Filter by Type			
	<input type="text" value="beanstalk"/>	<input type="button" value="X"/>	All types
<input type="checkbox"/>	<a href="#">AdministratorAccess-AWSE...</a>	AWS managed	Grants account administrative permis...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkCustom...</a>	AWS managed	Provide the instance in your custom pl...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkEnhanc...</a>	AWS managed	AWS Elastic Beanstalk Service policy f...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkManag...</a>	AWS managed	This policy is for the AWS Elastic Bean...
<input checked="" type="checkbox"/>	<a href="#">AWSElasticBeanstalkMultico...</a>	AWS managed	Provide the instances in your multicon...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkReadOnly</a>	AWS managed	Grants read-only permissions. Explicitl...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleCore</a>	AWS managed	AWSElasticBeanstalkRoleCore (Elastic ...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleCWL</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleECS</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleRDS</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleSNS</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleWo...</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input checked="" type="checkbox"/>	<a href="#">AWSElasticBeanstalkWebTier</a>	AWS managed	Provide the instances in your web serv...
<input checked="" type="checkbox"/>	<a href="#">AWSElasticBeanstalkWorker...</a>	AWS managed	Provide the instances in your worker e...

15. Name the role and click on create role.

### Name, review, and create

<b>Role details</b>
<b>Role name</b> Enter a meaningful name to identify this role. <input type="text" value="java-role"/> <small>Maximum 64 characters. Use alphanumeric and '+,-,_' characters.</small>
<b>Description</b> Add a short explanation for this role. <div style="border: 1px solid #ccc; padding: 5px; height: 60px; margin-top: 5px;"> <small>Allows EC2 instances to call AWS services on your behalf.</small> </div> <small>Maximum 1000 characters. Use alphanumeric and '+,-,_' characters.</small>
<b>Step 1: Select trusted entities</b>
<b>Trust policy</b> <pre> 1 - [ 2   "Version": "2012-10-17", 3   "Statement": [ 4     { 5       "Effect": "Allow", 6       ... 7     } 8   ] 9 ] </pre>

## 16. Role created.

The screenshot shows the AWS IAM Roles list. At the top, a green banner displays the message "Role java-role created." Below this, the IAM navigation bar is visible, followed by the Roles page title "Roles (6) Info". A search bar and filter buttons for "Trusted entities" and "Last activity" are present. The table lists six roles, including the newly created "java-role".

Role name	Trusted entities	Last activity
<a href="#">aws-elasticbeanstalk-service-role</a>	AWS Service: elasticbeanstalk	18 minutes ago
<a href="#">AWSServiceRoleForAutoScaling</a>	AWS Service: autoscaling (Service-Linker)	26 minutes ago
<a href="#">AWSServiceRoleForSupport</a>	AWS Service: support (Service-Linker)	-
<a href="#">AWSServiceRoleForTrustedAdvisor</a>	AWS Service: trustedadvisor (Service-Linker)	-
<a href="#">beanstalk-role</a>	AWS Service: ec2	16 minutes ago
<a href="#">java-role</a>	AWS Service: ec2	-

## 17. Go to EC2, click on the key pairs and create a key pair.

The screenshot shows the AWS Key Pairs list. At the top, a green banner displays the message "Key pairs (1) Info". Below this, the Key Pairs page title "Key pairs (1) Info" is shown. A search bar and filter buttons for "Name", "Type", "Created", and "Fingerprint" are present. The table lists one key pair, "sample-key-pair".

Name	Type	Created	Fingerprint	ID
sample-key-pair	rsa	2024/01/29 20:32 GMT+5:30	42:f0:5a:d4:ec:2d:f1:d0:9e:e6:1...	key-0e769a5a5cd33fee7

## 18. Create a key pair.

The screenshot shows the "Create key pair" wizard. The first step, "Key pair", is displayed. It includes a description of what a key pair is and a note about character restrictions. The "Name" field is set to "say-java-key". The "Key pair type" section shows "RSA" selected. Under "Private key file format", ".ppk" is selected, with a note that it's for use with PuTTY. The "Tags - optional" section indicates no tags are associated with the resource. At the bottom, there are buttons for "Cancel" and "Create key pair".

19. Created.

Key pairs (2) <a href="#">Info</a>					
<input type="checkbox"/>	Name	Type	Created	Fingerprint	ID
<input type="checkbox"/>	say-java-key	rsa	2024/01/29 22:22 GMT+5:30	7c:24:57:24:51:73:8d:81:37:7b:6d:ba:2...	key-06cd0...
<input type="checkbox"/>	sample-key-pair	rsa	2024/01/29 20:32 GMT+5:30	42:f0:5a:d4:ec:2d:f1:d0:9e:e6:18:11:e2:...	key-0e769...

20. Go to the previous tab, and refresh and select the role and key pair and click on 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

**Service role name**  
Enter the name for an IAM role that Elastic Beanstalk will create to assume as a service role. Beanstalk will attach the required managed policies to it.

[View permission details](#)

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

[C](#)

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

[View permission details](#)

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

21. Select VPC.

## Set up networking, database, and tags - *optional* [Info](#)

### 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-025f7c4e1bc4e0cba | (172.31.0.0/16) 

[Create custom VPC](#) 

22. Select the same IP address in instance as of VPC (172.31.0.0).

### 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
<input type="checkbox"/> ap-south-1c	subnet-018470deb...	172.31.16.0/20	
<input checked="" type="checkbox"/> ap-south-1b	subnet-040070104...	172.31.0.0/20	
<input type="checkbox"/> ap-south-1a	subnet-0dd23cf18...	172.31.32.0/20	

23. Select the same IP address in the database as of VPC.

The screenshot shows the 'Database Info' configuration step in the AWS Elastic Beanstalk console. It displays a table of database subnets:

	Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/>	ap-south-1c	subnet-018470deb...	172.31.16.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-040070104...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1a	subnet-0dd23cf18...	172.31.32.0/20	

Below the table, there is a checkbox labeled 'Enable database'. Under 'Restore a snapshot - optional', it says 'Restore an existing snapshot from a previously used database.' A dropdown menu for 'Snapshot' is set to 'None'. At the bottom, under 'Database settings', it says 'Choose an engine and instance type for your environment's database.' A dropdown menu for 'Engine' is set to 'None'.

24. Click on next.

25. Set as default, no changes and next.

## Configure instance traffic and scaling - *optional* [Info](#)

### ▼ Instances [Info](#)

Configure the Amazon EC2 instances that run your application.

#### Root volume (boot device)

##### Root volume type

(Container default)



##### Size

The number of gigabytes of the root volume attached to each instance.

8

GB

##### IOPS

Input/output operations per second for a provisioned IOPS (SSD) volume.

100

IOPS

##### Throughput

The desired throughput to provision for the Amazon EBS root volume attached to your environment's EC2 instance

125

MiB/s

#### Amazon CloudWatch monitoring

The time interval between when metrics are reported from the EC2 instances

##### Monitoring interval

26. Set as default, no changes and next.

## 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

*Choose metrics* 

##### CloudWatch Custom Metrics - Environment

*Choose metrics* 

#### 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

7 

## 27. Review.

**Review [Info](#)**

**Step 1: Configure environment** [Edit](#)

Environment information	
Environment tier	Application name
Web server environment	this-java
Environment name	Application code
This-java-env	SampleWebApp.war
Platform	
arn:aws:elasticbeanstalk:ap-south-1::platform/Corretto 21 running on 64bit Amazon Linux 2023/4.2.0	

**Step 2: Configure service access** [Edit](#)

Service access <a href="#">Info</a>		
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 key pair	EC2 instance profile
arn:aws:iam::544606390347:role/service-role/aws-elasticbeanstalk-	say-java-key	java-role

## 28. Submit.

Rotate logs	Update level	X-Ray enabled								
Deactivated	minor	Deactivated								
<b>Environment properties</b>										
<table border="1"> <thead> <tr> <th>Key</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>GRADLE_HOME</td> <td>/usr/local/gradle</td> </tr> <tr> <td>M2</td> <td>/usr/local/apache-maven/bin</td> </tr> <tr> <td>M2_HOME</td> <td>/usr/local/apache-maven</td> </tr> </tbody> </table>			Key	Value	GRADLE_HOME	/usr/local/gradle	M2	/usr/local/apache-maven/bin	M2_HOME	/usr/local/apache-maven
Key	Value									
GRADLE_HOME	/usr/local/gradle									
M2	/usr/local/apache-maven/bin									
M2_HOME	/usr/local/apache-maven									

[Cancel](#) [Previous](#) **Submit**

## 29. Launching.

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

[Elastic Beanstalk](#) > [Environments](#) > This-java-env

### This-java-env [Info](#)

[C](#) [Actions ▾](#) [Upload and deploy](#)

Environment overview		Platform	<a href="#">Change version</a>
Health	Environment ID	Platform	
(⌚ Pending)	e-pifn2ksbp7	Corretto 21 running on 64bit Amazon Linux 2023/4.2.0	
Domain	Application name	Running version	-
-	this-java	Platform state	<span style="color: green;">Supported</span>

[Events](#) [Health](#) [Logs](#) [Monitoring](#) [Alarms](#) [Managed updates](#) [Tags](#)

## 30. Environment is created.

[Applications \(2\) \[Info\]\(#\)](#)

[C](#) [Actions ▾](#) [Create application](#)

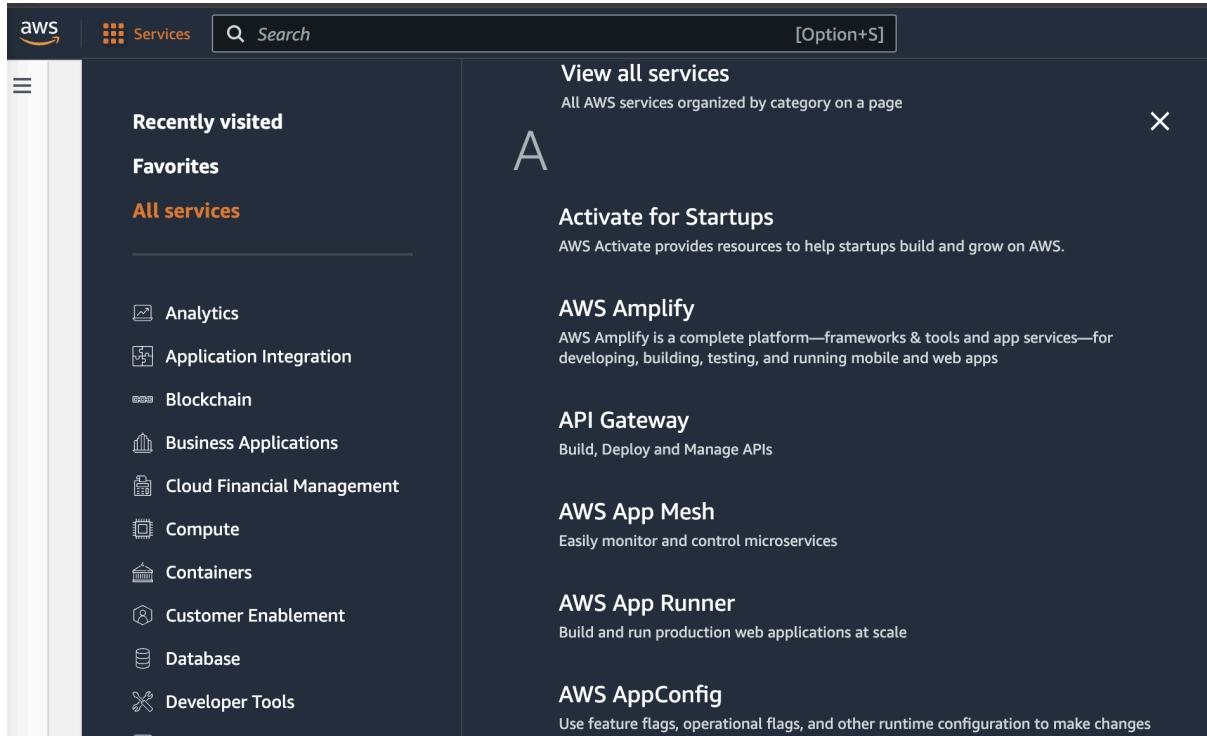
Filter results matching the display value

Application name	Environments	Date created	Last modified
<a href="#">test-web-application</a>	<a href="#">Test-web-application-env</a>	January 29, 2024 20:01:40 (...)	January 29, 2024 20:01:40 (...)
<a href="#">this-java</a>	<a href="#">This-java-env</a>	January 29, 2024 22:04:47 (...)	January 29, 2024 22:04:47 (...)

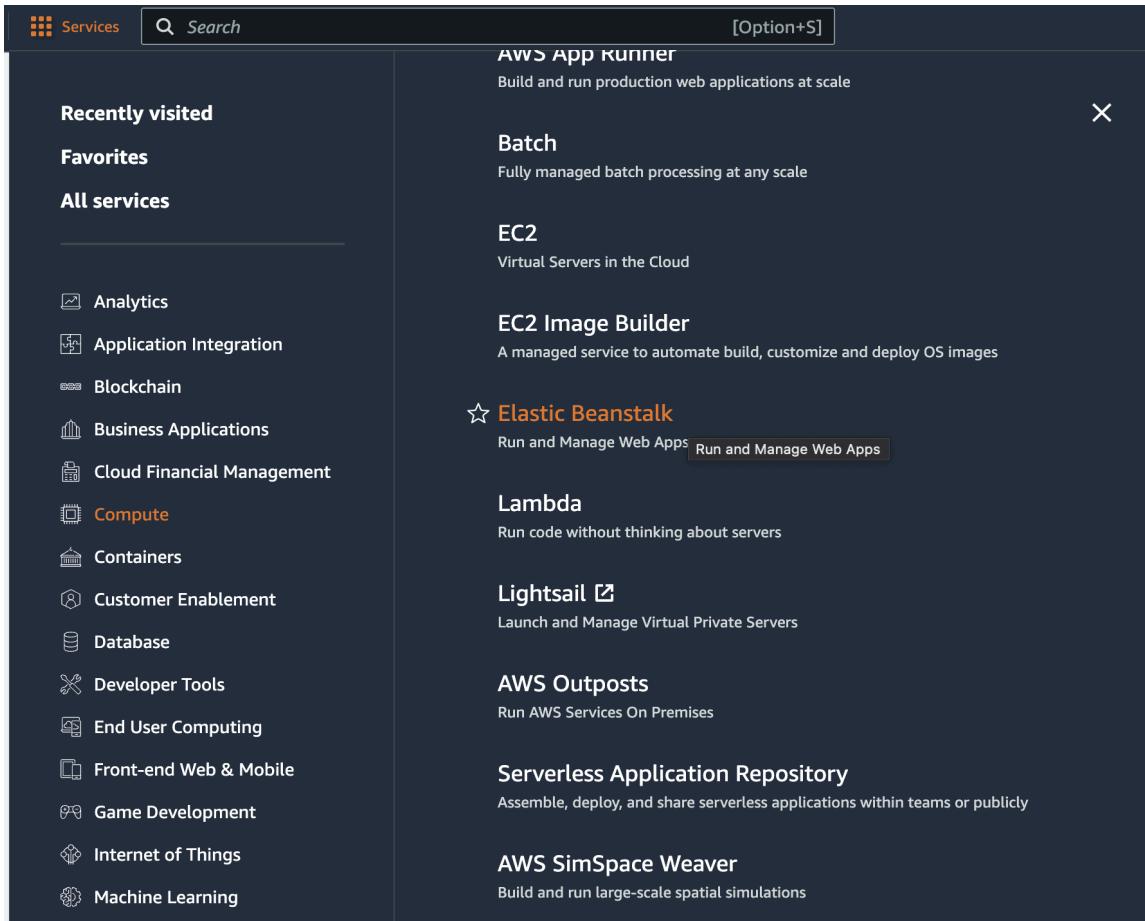
## Implement PaaS using elastic beanstalk with Tomcat Application

1. Sign in to your AWS account.

2. Select All Services.



### 3. Select Elastic Beanstalk.



### 4. Create an application.

The screenshot shows the Amazon Elastic Beanstalk landing page. At the top, it says 'Compute' and features a large heading 'Amazon Elastic Beanstalk' with the subtext 'End-to-end web application management.'. Below this is a description: '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.' To the right, there's a 'Get started' section with a 'Create application' button and a note: 'Easily deploy your web application in minutes.' Further down, there's a 'Pricing' section stating: '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.'

5. Add the application name and description.

## Configure environment Info

**Environment tier Info**  
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. [Learn more](#)

**Worker environment**  
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule. [Learn more](#)

## Application information Info

Application name  
  
Maximum length of 100 characters.

▶ Application tags (optional)

## Environment information Info

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

Environment name

## 6. Upload the platform, Tomcat.

**Environment description**

elastic beanstalk with tomcat application.

---

**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**

Tomcat

**Platform branch**

Tomcat 10 with Corretto 17 running on 64bit Amazon Linux 2023

**Platform version**

5.1.3 (Recommended)

## 7. Upload the code. (WAR file)

**Application code Info**

**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.

**Version label**  
Unique name for this version of your application code.  
sample-code-is

Source code origin. Maximum size 500 MB

**Local file**

Upload application

**File name: Calendar.war**  
File must be less than 500MB max file size

**Public S3 URL**

---

**Presets Info**  
Start from a preset that matches your use case or choose custom configuration to unset recommended values and use the service's default values.

**Configuration presets**

**Single instance (free tier eligible)**

**Single instance (using root instance)**

8. Click on next.

9. Click on create a new service role.

The screenshot shows the 'Configure service access' wizard. Under the 'Service role' section, the 'Create and use new service role' option is selected. A note below states: '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.' A 'Learn more' link is also present.

10. For creating a new role, Go to IAM in ALL SERVICES. (use a new tab)

The screenshot shows the AWS Services dashboard. The 'All services' section is open, displaying various AWS services. The 'IAM' service is highlighted with a yellow star icon and a callout box that reads 'Manage access to AWS resources'. Other visible services include Analytics, Application Integration, Blockchain, Business Applications, Cloud Financial Management, Compute, Containers, Customer Enablement, Database, Developer Tools, End User Computing, Front-end Web & Mobile, and Game Development.

11. Click on roles and then create role.

The screenshot shows the AWS IAM 'Roles' page. At the top, there's a breadcrumb navigation 'IAM > Roles'. Below it, a title 'Roles (5) Info' is followed by a description: 'An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.' A search bar labeled 'Search' is present. On the right side, there are three buttons: 'Create role' (highlighted in orange), 'Delete', and a refresh icon. Below these buttons are navigation arrows and a gear icon. The main area lists 5 roles, each with a small preview icon, the role name, and a 'View details' link.

12. Select EC2 and next.

The screenshot shows the 'Select EC2' step in the AWS IAM Role creation wizard. It starts with a 'Use case' section describing allowing AWS services like EC2, Lambda, or others to perform actions in the account. Below this is a 'Service or use case' dropdown menu set to 'EC2'. The main area lists several 'Use case' options, each with a description:

- EC2**: Allows EC2 instances to call AWS services on your behalf.
- EC2 Role for AWS Systems Manager**: Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.
- EC2 Spot Fleet Role**: Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.
- EC2 - Spot Fleet Auto Scaling**: Allows Auto Scaling to access and update EC2 spot fleets on your behalf.
- EC2 - Spot Fleet Tagging**: Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.
- EC2 - Spot Instances**: Allows EC2 Spot Instances to launch and manage spot instances on your behalf.
- EC2 - Spot Fleet**: Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.
- EC2 - Scheduled Instances**: Allows EC2 Scheduled Instances to manage instances on your behalf.

At the bottom right, there are 'Cancel' and 'Next' buttons.

13. Select the following and next.

Filter by Type			
<input type="text"/> beanstalk		Type	14 matches
		Policy name	Description
<input type="checkbox"/>	<a href="#">AdministratorAccess-AWSE...</a>	AWS managed	Grants account administrative permis...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkCustom...</a>	AWS managed	Provide the instance in your custom pl...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkEnhanc...</a>	AWS managed	AWS Elastic Beanstalk Service policy f...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkManag...</a>	AWS managed	This policy is for the AWS Elastic Bean...
<input checked="" type="checkbox"/>	<a href="#">AWSElasticBeanstalkMultico...</a>	AWS managed	Provide the instances in your multicon...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkReadOnly</a>	AWS managed	Grants read-only permissions. Explicit...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleCore</a>	AWS managed	AWSElasticBeanstalkRoleCore (Elastic ...)
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleCWL</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleECS</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleRDS</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleSNS</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input type="checkbox"/>	<a href="#">AWSElasticBeanstalkRoleWo...</a>	AWS managed	(Elastic Beanstalk operations role) Allo...
<input checked="" type="checkbox"/>	<a href="#">AWSElasticBeanstalkWebTier</a>	AWS managed	Provide the instances in your web serv...
<input checked="" type="checkbox"/>	<a href="#">AWSElasticBeanstalkWorker...</a>	AWS managed	Provide the instances in your worker e...

14. Name the role and click on create role.

# Name, review, and create

## Role details

**Role name**  
Enter a meaningful name to identify this role.  
Maximum 64 characters. Use alphanumeric and '+=\_,@-' characters.

**Description**  
Add a short explanation for this role.  
Maximum 1000 characters. Use alphanumeric and '+=\_,@-' characters.

## Step 1: Select trusted entities

Trust policy

```
1 - {  
2     "Version": "2012-10-17",  
3     "Statement": [  
4         {  
5             "Action": "sts:AssumeRole",  
6             "Effect": "Allow",  
7             "Principal": "arn:aws:iam::123456789012:root"  
8         }  
9     ]  
10 }
```

15. Role created.

Role tomcat-role created. [View role](#)

[IAM](#) > Roles

**Roles (8) [Info](#)**

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

[Create role](#)

16. Go to EC2, click on the key pairs and create a key pair.

Key pairs (1) <a href="#">Info</a>					
<input type="text"/> Find Key Pair by attribute or tag				<a href="#">Actions</a> ▾	<a href="#">Create key pair</a>
<input type="checkbox"/>	Name	Type	Created	Fingerprint	ID
<input type="checkbox"/>	sample-key-pair	rsa	2024/01/29 20:32 GMT+5:30	42:f0:5a:d4:ec:2d:f1:d0:9e:e6:1...	key-0e769a5a5cd33fee7

17. Create a key pair.

[EC2](#) > [Key pairs](#) > [Create key pair](#)

### Create key pair [Info](#)

**Key pair**  
A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance.

**Name**  
  
The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

**Key pair type** [Info](#)  
 RSA       ED25519

**Private key file format**  
 .pem  
For use with OpenSSH  
 .ppk  
For use with PuTTY

**Tags - optional**  
No tags associated with the resource.  
[Add new tag](#)  
You can add up to 50 more tags.

18. Upload role and key pair and 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.

aws-elasticbeanstalk-service-role ▼

**EC2 key pair**

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

tomcat-ehhh ▼

**EC2 instance profile**

Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.

tomcat-role ▼

[View permission details](#)

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

19. Select VPC.

**Set up networking, database, and tags - optional** Info

**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-025f7c4e1bc4e0cba | (172.31.0.0/16) ▼

[Create custom VPC](#)

20. Select the same IP address in instance as of VPC (172.31.0.0).

**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
<input type="checkbox"/>	ap-south-1c	subnet-018470deb...	172.31.16.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-040070104...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1a	subnet-0dd23cf18...	172.31.32.0/20	

21. Select the same IP address in the database as of VPC.

The screenshot shows the 'Database Info' configuration step in the AWS Elastic Beanstalk console. It displays a table of database subnets:

Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/> ap-south-1c	subnet-018470deb...	172.31.16.0/20	
<input checked="" type="checkbox"/> ap-south-1b	subnet-040070104...	172.31.0.0/20	
<input type="checkbox"/> ap-south-1a	subnet-0dd23cf18...	172.31.32.0/20	

Below the table, there is a checkbox labeled 'Enable database'. Under 'Restore a snapshot - optional', it says 'Restore an existing snapshot from a previously used database.' A dropdown menu for 'Snapshot' shows 'None'. At the bottom, under 'Database settings', it says 'Choose an engine and instance type for your environment's database.' A dropdown menu for 'Engine' shows 'None'.

22. Click on next.

23. Set as default, no changes and next.

## Configure instance traffic and scaling - *optional* Info

### ▼ Instances Info

Configure the Amazon EC2 instances that run your application.

#### Root volume (boot device)

##### Root volume type

(Container default)



##### Size

The number of gigabytes of the root volume attached to each instance.

8

GB

##### IOPS

Input/output operations per second for a provisioned IOPS (SSD) volume.

100

IOPS

##### Throughput

The desired throughput to provision for the Amazon EBS root volume attached to your environment's EC2 instance

125

MiB/s

#### Amazon CloudWatch monitoring

The time interval between when metrics are reported from the EC2 instances

##### Monitoring interval

24. Set as default, no changes and next.

## 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

*Choose metrics*

##### CloudWatch Custom Metrics - Environment

*Choose metrics*

#### 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

7

## 25. Review.

**Review** [Info](#)

**Step 1: Configure environment** [Edit](#)

Environment information	
Environment tier	Application name
Web server environment	say-tom-cat
Environment name	Application code
Say-tom-cat-env	Calendar.war
Platform	
arn:aws:elasticbeanstalk:ap-south-1::platform/Tomcat 10 with Corretto 17 running on 64bit Amazon Linux 2023/5.1.3	

**Step 2: Configure service access** [Edit](#)

Service access		
<a href="#">Info</a>		
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 key pair	EC2 instance profile
arn:aws:iam::544606390347:role/ser	tomcat-ehhh	tomcat-role

26. Click on submit.

The screenshot shows the 'Platform software' section with the following settings:

Lifecycle	Log streaming	Initial JVM heap size (Xms)
false	Deactivated	256m
Max JVM heap size (Xmx)	JVM options	Proxy server
256m	-	nginx
Logs retention	Rotate logs	Update level
7	Deactivated	minor
X-Ray enabled		
Deactivated		

**Environment properties**

Key	Value
JDBC_CONNECTION_STRING	

Buttons at the bottom: Cancel, Previous, Submit (highlighted in orange).

27. Launching.

The screenshot shows the 'Environment overview' and 'Platform' sections for the environment 'Say-tom-cat-env'.

**Environment overview**

Health	Environment ID
⊖ Unknown	⌚ e-dqmmpsfh2a
Domain	Application name
-	say-tom-cat

**Platform**

Platform	Change version
Tomcat 10 with Corretto 17 running on 64bit Amazon Linux 2023/5.1.3	
Running version	-
Platform state	_supported_

## 27. Launched.

Say-tom-cat-env [Info](#)

[Actions](#) [Upload and deploy](#)

Environment overview		Platform	<a href="#">Change version</a>
Health	Environment ID	Platform	
<span>Ok</span>	e-dqmmpsfh2a	Tomcat 10 with Corretto 17 running on 64bit Amazon Linux 2023/5.1.3	
Domain	Application name	Running version	
<a href="#">Say-tom-cat-env.eba-hz76tbuv.ap-south-1.elasticbeanstalk.com</a>	say-tom-cat	sample-code-is	
		Platform state	
		<span>Supported</span>	

## GWT Calendar

Click on day to get date popup. Example Datepicker. Built with the tomcat war builder.  
<http://code.google.com/p/gwt-examples/>

