## Practical-5 Platform as a service using AWS

Name: Shriomi Gite

Roll no: A015

## 1)Platform as a Service (PaaS)

Platform as a Service (PaaS) is a cloud computing model that provides a ready-to-use environment for developers to build, run, and manage applications without worrying about the underlying infrastructure. PaaS offers pre-configured tools, libraries, and frameworks that simplify the development and deployment process, helping developers focus on coding and application logic rather than server management and infrastructure maintenance. Examples of PaaS include AWS Elastic Beanstalk, Google App Engine, and Microsoft Azure App Service.

## 2)AWS Elastic Beanstalk

AWS Elastic Beanstalk is an easy-to-use PaaS solution that automates the deployment, scaling, and management of applications in the AWS Cloud. With Elastic Beanstalk, developers can deploy applications written in popular programming languages (like Python, Java, Node.js, PHP, Ruby, and more) by simply uploading code. Beanstalk handles tasks such as provisioning the necessary infrastructure, managing load balancing, auto-scaling, monitoring, and more. It's particularly useful for fast deployments and quick iterations while maintaining flexibility over the underlying AWS services.

## 3) Components of Elastic Beanstalk

Elastic Beanstalk comprises several key components that work together to create and manage the environment in which applications run. These include:

- 1. Application: A container that holds different environments. Each environment runs a version of the application, allowing for testing and production environments within the same application container.
- 2. Environment: The specific infrastructure resources and settings used to run an application version. Each environment is associated with a URL, load balancer, and other resources that Elastic Beanstalk configures based on the application's needs.
- 3. Environment Configuration: Settings applied to the environment, such as instance type, security groups, scaling policies, and software versions. These configurations can be updated as the application evolves.
- 4. Application Version: A deployable version of the code. Multiple versions can be stored and rolled back if necessary.
- 5. Environment Tier: The type of environment, either Web Server (used for web applications) or Worker (for asynchronous, background tasks).
- 6. Elastic Beanstalk CLI (EB CLI): A command-line tool that facilitates the management of applications and environments directly from the command line.

## 4) IAM (Identity and Access Management)

AWS Identity and Access Management (IAM) is a security service that enables users to securely control access to AWS services and resources. With IAM, users can manage permissions and access controls for various AWS resources, ensuring that only authorized individuals or applications can perform specific actions. Key IAM components include:

- 1. Users: Individual user accounts representing a person or service.
- 2. Groups: Collections of users to simplify permission management. Permissions applied to a group apply to all members.
- 3. Roles: Roles allow users or services to assume permissions temporarily, ideal for cross-account or temporary access.
- 4. Policies: JSON-based documents that define permissions and can be attached to users, groups, or roles to control access.

IAM plays a crucial role in Elastic Beanstalk by securing access to application environments and associated resources, allowing developers to manage permissions granularly.

1. We'll be using Elastic Beanstalk - Python



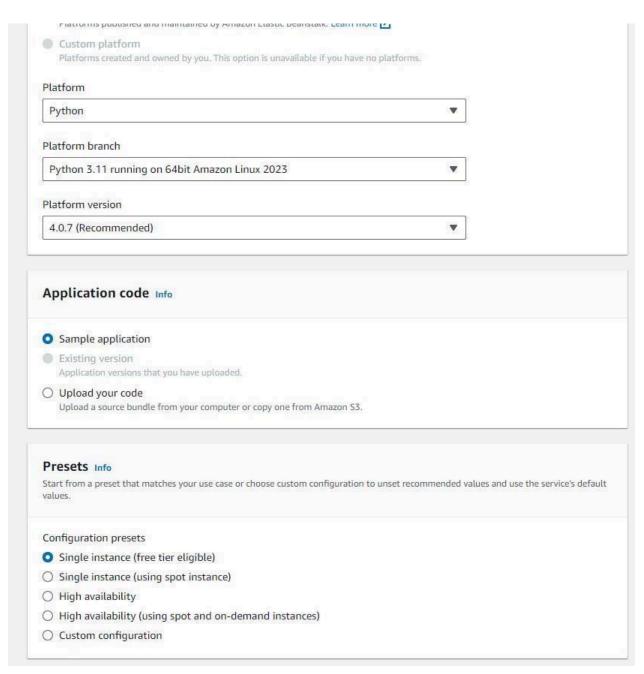
3. Fill in the details

Configure environment		
Step 2		
Configure	e service access	
Step 3 - <i>op</i>	ntional	
Set up ne	etworking, database,	
and tags		
Step 4 - <i>op</i>	otional	
Configur	e instance traffic and	
scaling		
Step 5 - <i>op</i>	otional	
Configure	e updates, monitoring,	
and logg	ing	
Step 6		
Review		

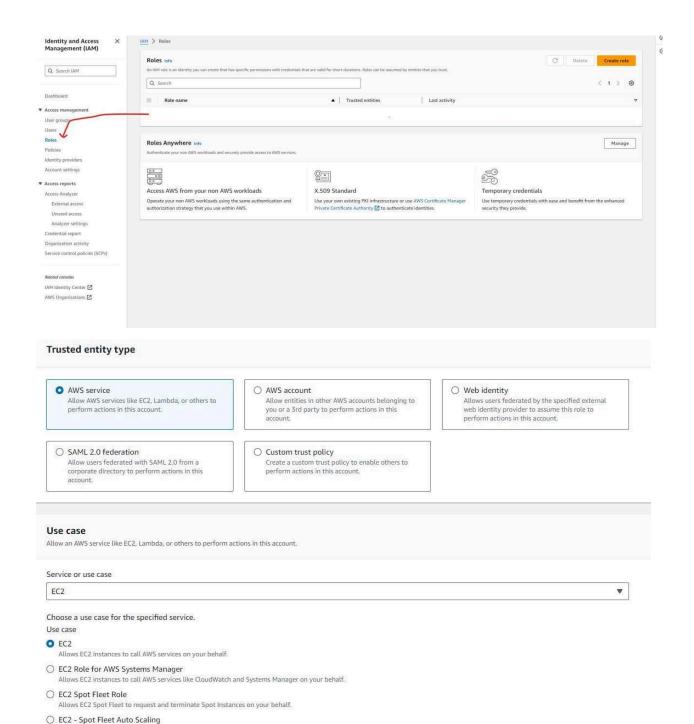
### 3. Fill in the details

Environment tier Info		
	ment tiers to support different types of web application	ns.
Web server environment     Run a website, web application, or web API that	t serves HTTP requests. Learn more	
○ Worker environment	unning workloads on demand or performs tasks on a so	chedule. Learn more 🔼
Application information Info		
Application name		
WebAPP		
Maximum length of 100 characters.		
► Application tags (optional)		
Environment information Info Choose the name, subdomain and description for y	our environment. These cannot be changed later.	
► Application tags (optional)  Environment information Info Choose the name, subdomain and description for y  Environment name  WebAPP-env	our environment. These cannot be changed later.	
Environment information Info Choose the name, subdomain and description for y Environment name WebAPP-env Must be from 4 to 40 characters in length. The nam This name must be unique within a region in your a	ne can contain only letters, numbers, and hyphens. It ca	in't start or end with a hyphen.
Environment information Info Choose the name, subdomain and description for y Environment name WebAPP-env Must be from 4 to 40 characters in length. The name	ne can contain only letters, numbers, and hyphens. It ca	in't start or end with a hyphen.  Check availability

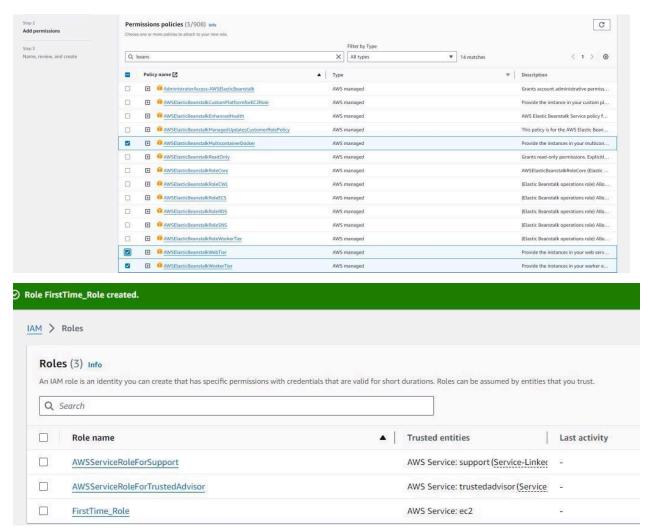
4. Select the platform



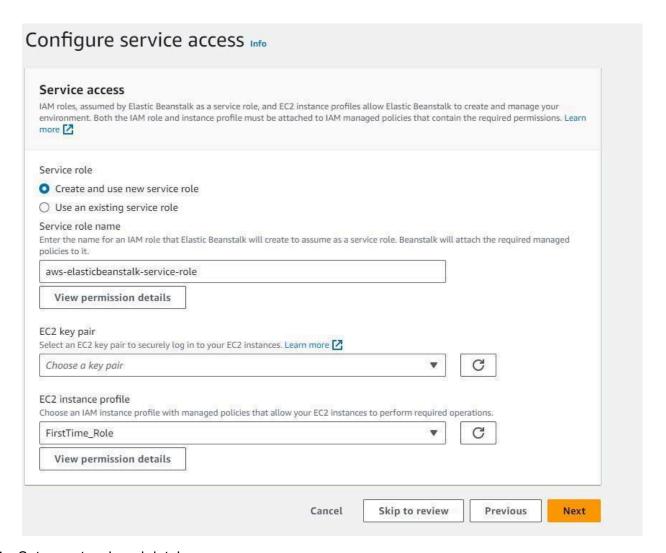
5. Go to the IAM services -> Create role



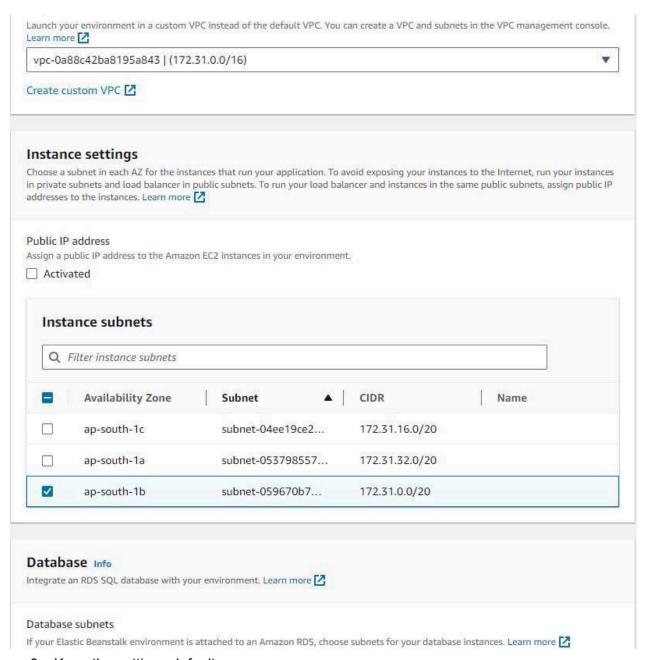
Allows Auto Scaling to access and update EC2 spot fleets on your behalf.



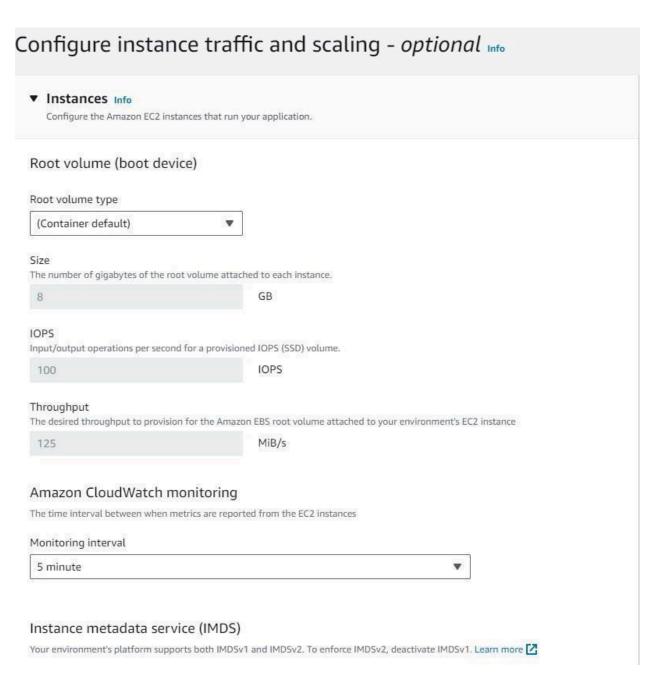
6. Create new service role, and select the EC instance profile



7. Set up network and database



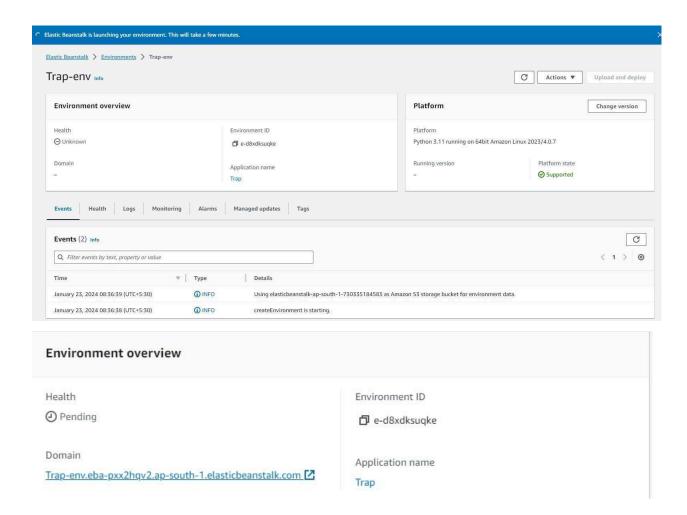
### 8. Keep the settings default



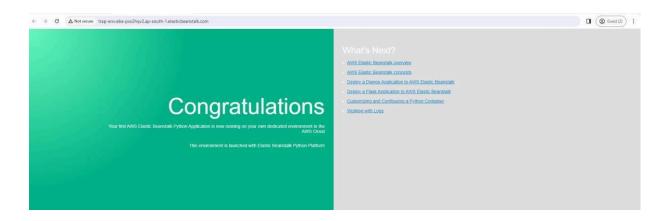
9. Keep the settings default

# 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 [2] System O 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 Lifecycle Keep logs after terminating environment

- 10. Review and submit!
- 11. The environment will start launching



#### The webapp is launched!



1. We'll be using Elastic Beanstalk - Java



2. There will be the steps

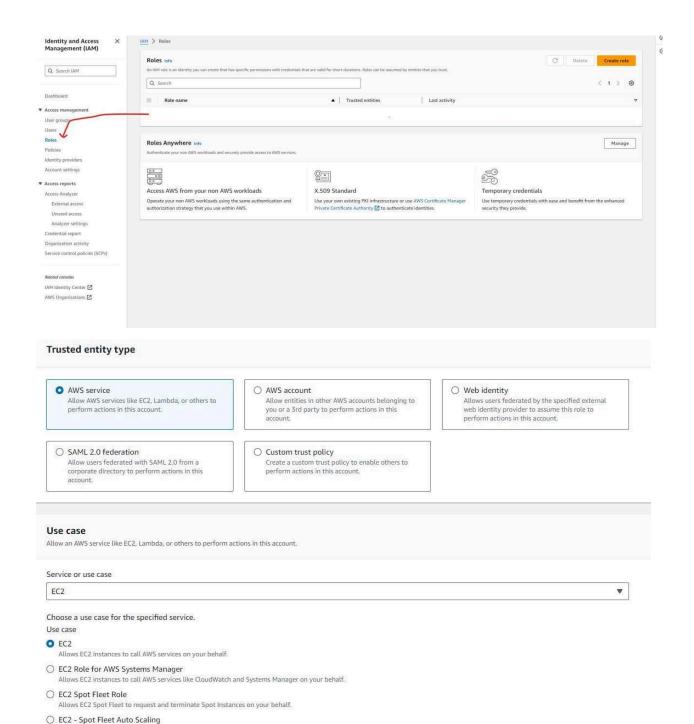
Configure	environment
Step 2	
Configure	service access
Step 3 - opt	tional
Set up net	tworking, database,
and tags	
Step 4 - opt	tional
Configure	instance traffic and
scaling	
Step 5 - opt	tional
Configure	updates, monitoring,
and loggii	ng
Step 6	
Review	

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	
<ul> <li>Worker environment</li> <li>Run a worker application that processes long-running workloads on demand or performs tasks on a sched</li> </ul>	ule Learn more [Z
Application information Info	
Application name	
AWS-Java	
Maximum length of 100 characters.	
► Application tags (optional)	

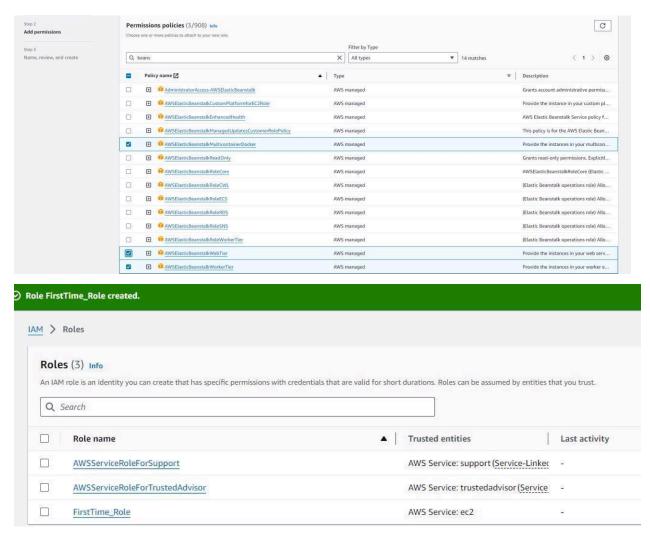
## 12. Select the platform

Platforms created and owned by you. This option is unavailable if you have no platforms.	ia.
form	
thon	▼
form branch	
thon 3.11 running on 64bit Amazon Linux 2023	•
form version	
0.7 (Recommended)	•
plication 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.	
esets Info	
t from a preset that matches your use case or choose custom configuration to unset reco es.	ommended values and use the service's defa
figuration presets	
Single instance (free tier eligible)	
Single instance (using spot instance)	
High availability	
High availability (using spot and on-demand instances)	

13. Go to the IAM services -> Create role



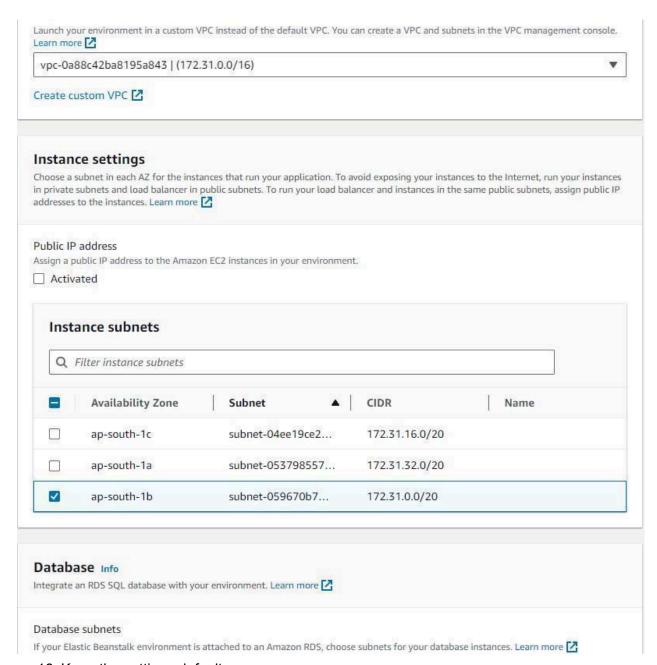
Allows Auto Scaling to access and update EC2 spot fleets on your behalf.



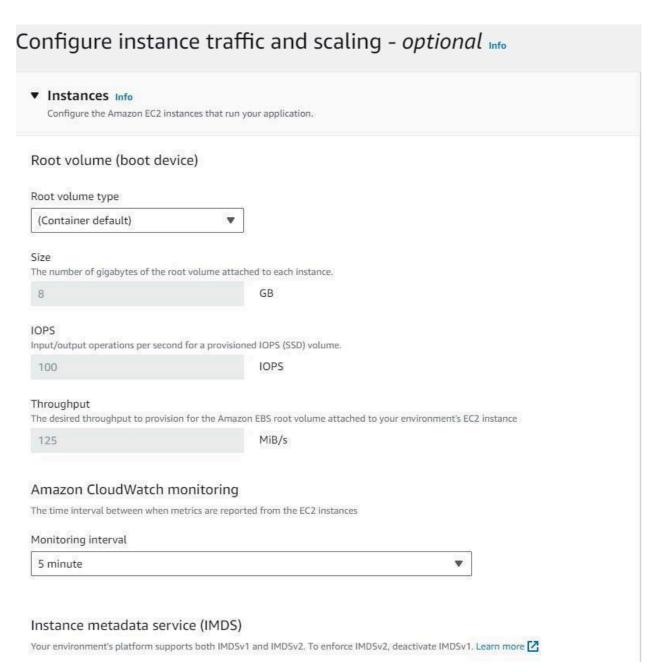
14. Create new service role, and select the EC instance profile

Service role	
Create and use new service role	
O 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 policies to it.	e. Beanstalk will attach the required manager
aws-elasticbeanstalk-service-role	
View permission details	70
EC2 key pair	
Select an EC2 key pair to securely log in to your EC2 instances. Learn more 🔼	
Choose a key pair	▼ C
EC2 instance profile  Choose an IAM instance profile with managed policies that allow your EC2 instances to per	form required operations.
Java-Server	▼ C

15. Set up network and database



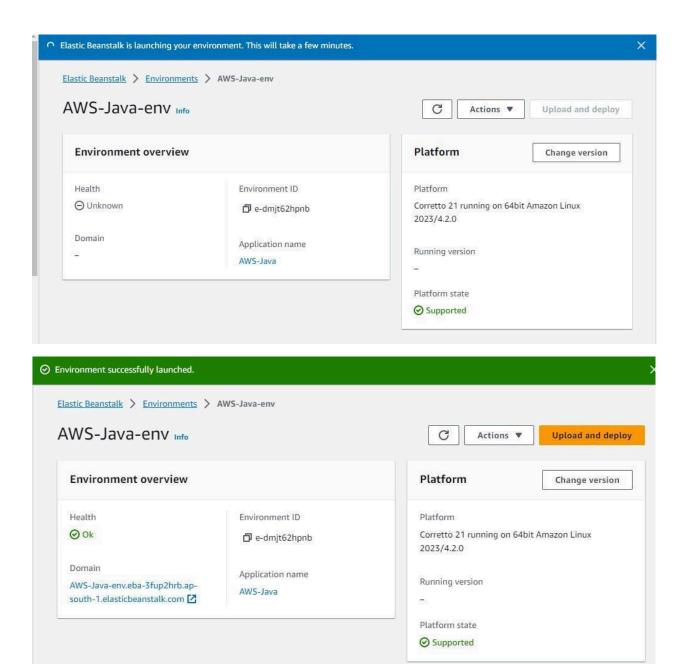
### 16. Keep the settings default



#### 17. Keep the settings default

# 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 [2] System O 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 Lifecycle Keep logs after terminating environment

- 18. Review and submit!
- 19. The environment will start launching



The webapp is launched!

