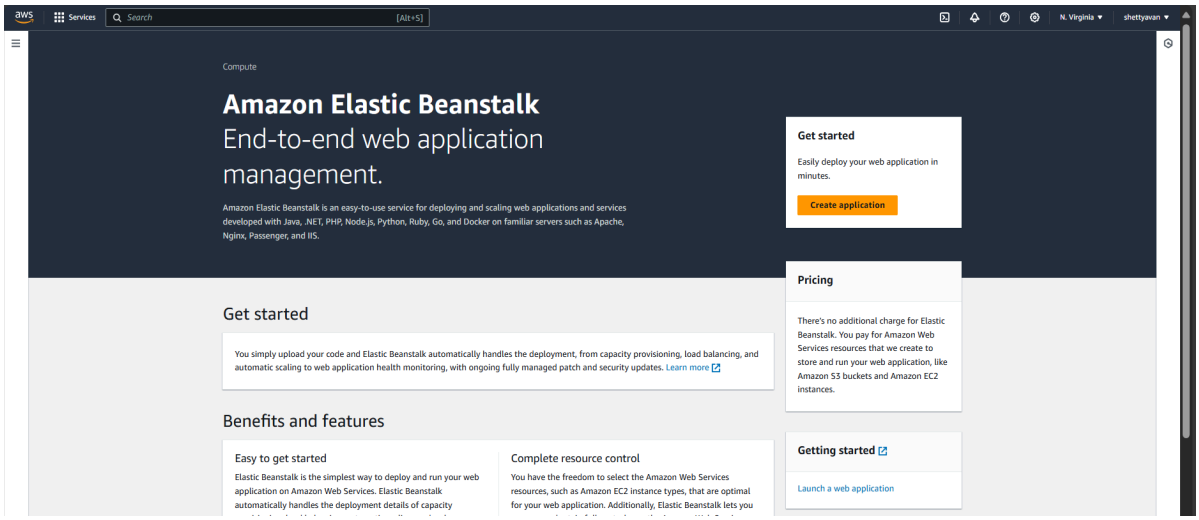
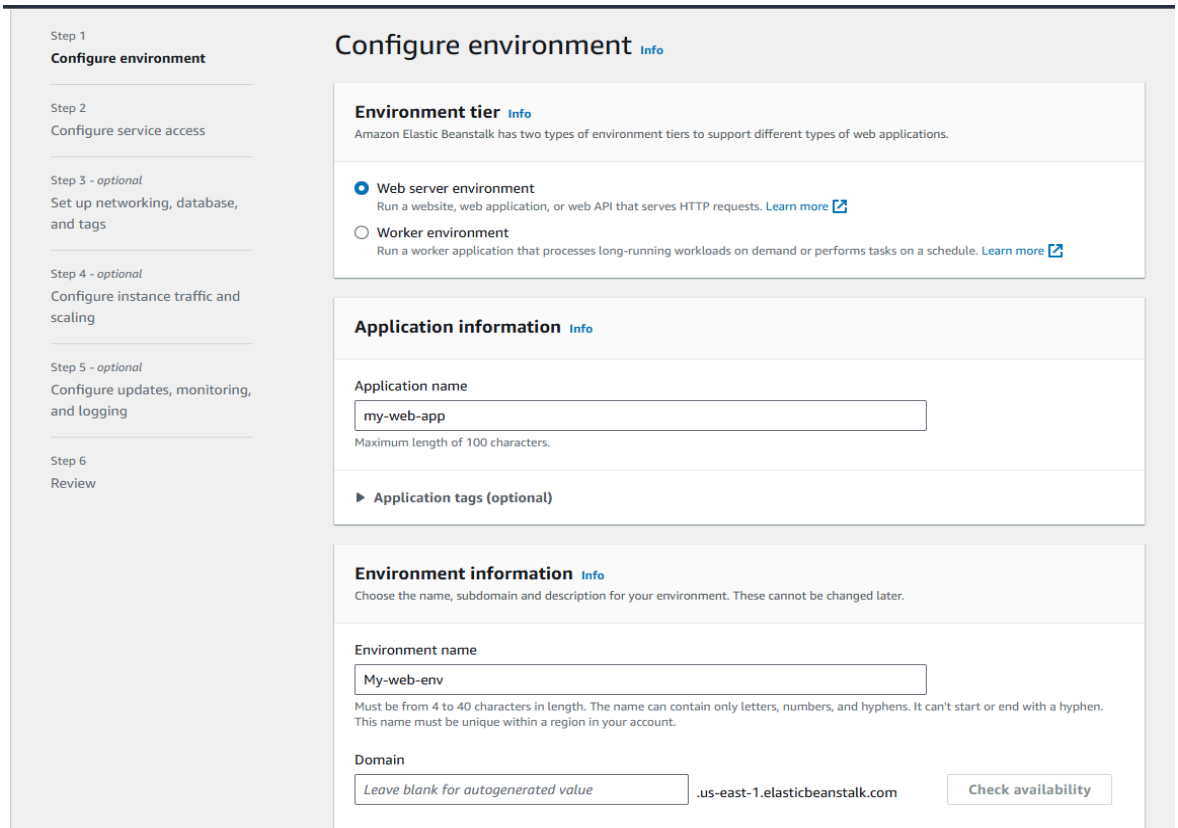
Aim : To Build Your Application using AWS CodeBuild and Deploy on S3 / SEBS using AWS CodePipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.

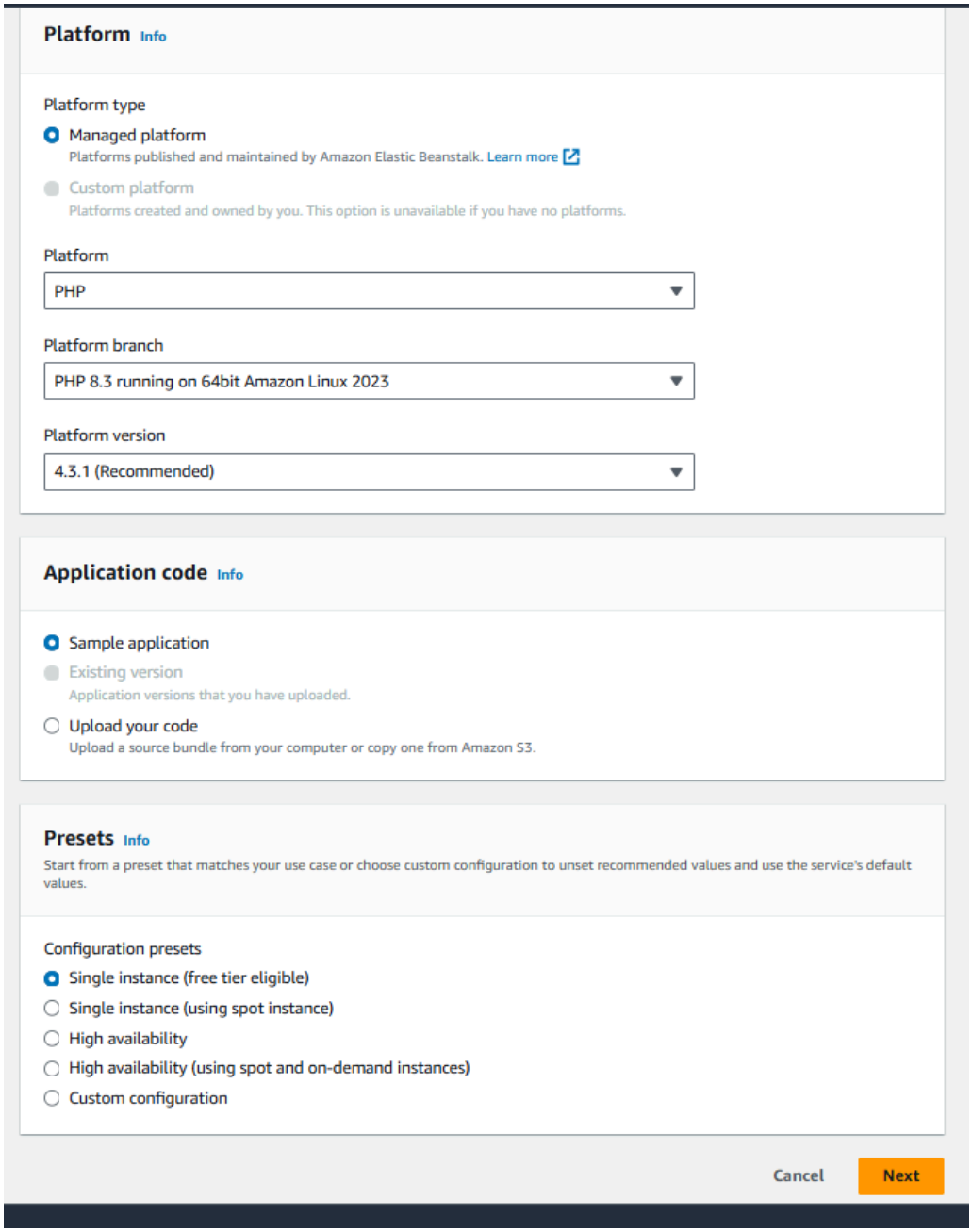
1. Open the aws console and then search Elastic Beanstalk (Opens a dashboard as seen below)



2. Click on create application and configure the environment by adding your application name

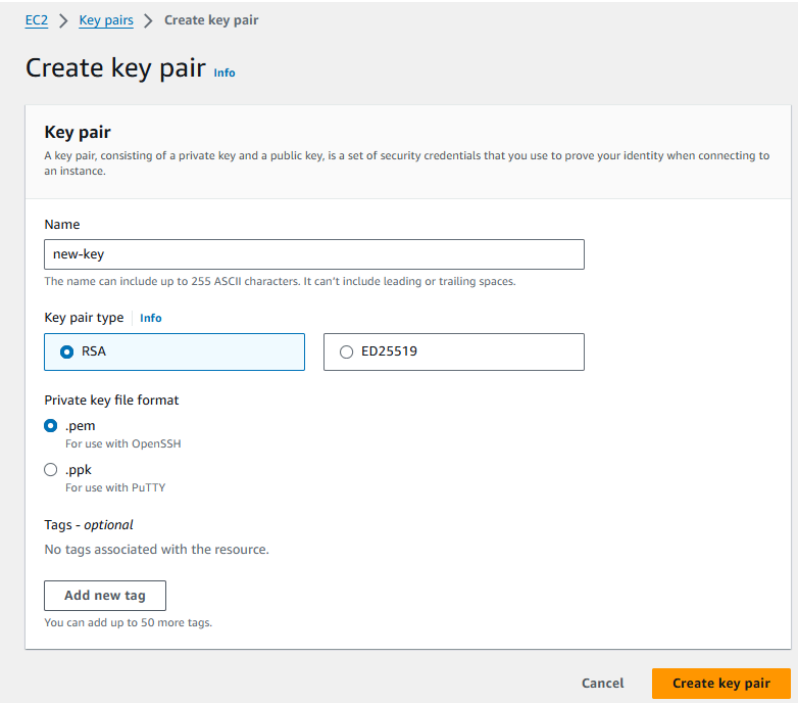


3. Choose PHP from the drop-down menu and click next

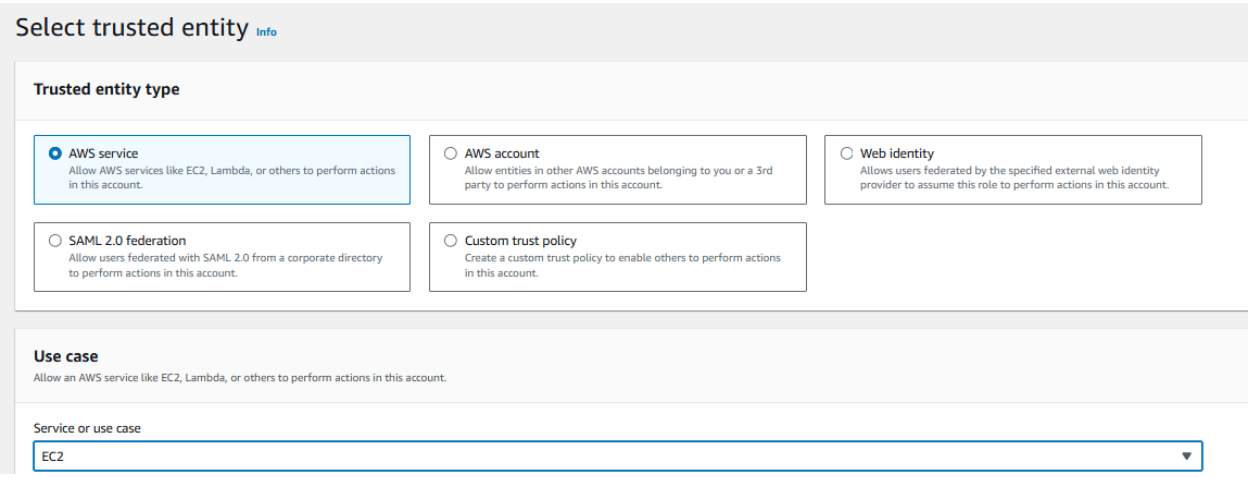


4. In this step you have to create a key pair

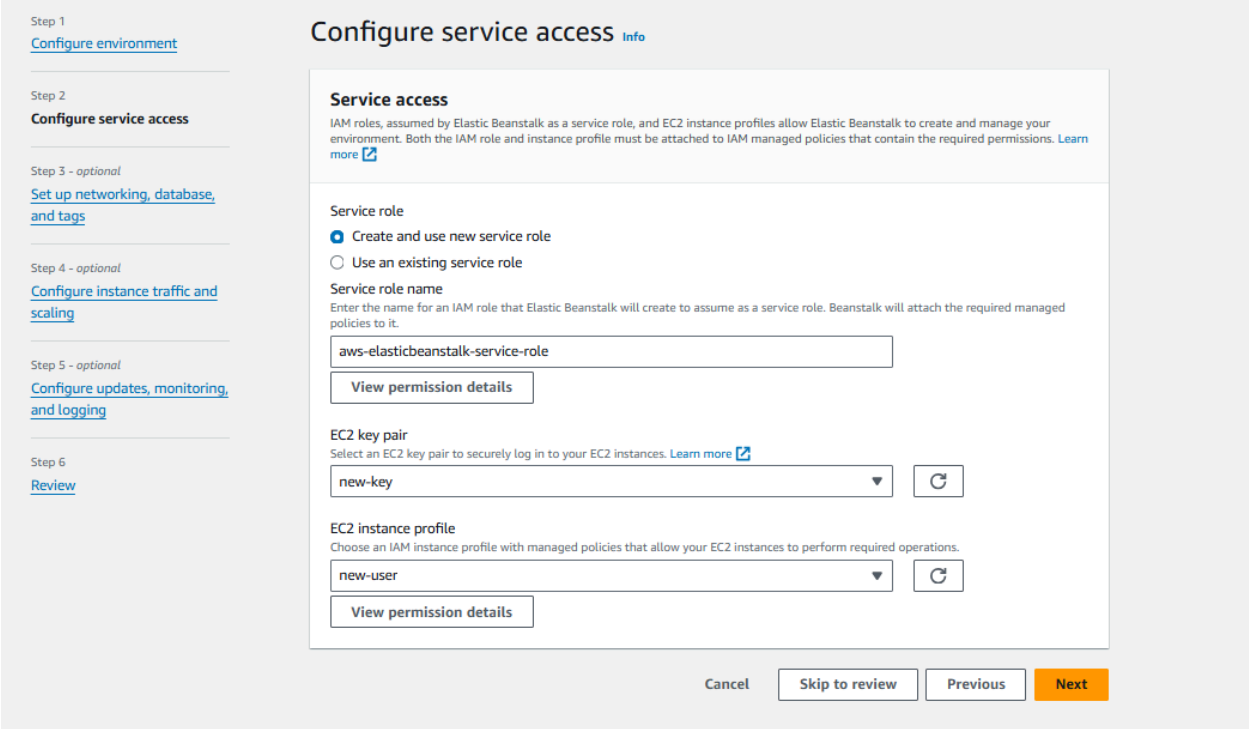
a. Go to EC2 instance tab and from the let panel create a key pair. As the key pair might be useful for the further process



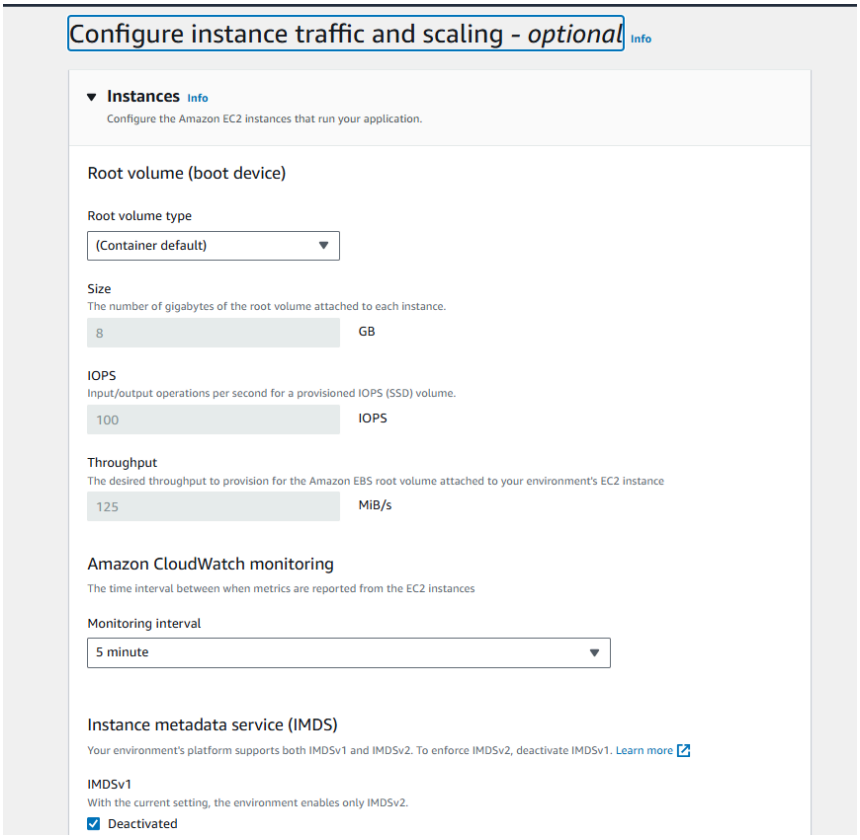
b. In the same fashion go to IAM and then under roles section click create role and then Select AWS Service and under instances select EC2. Your EC2 Instance role is created

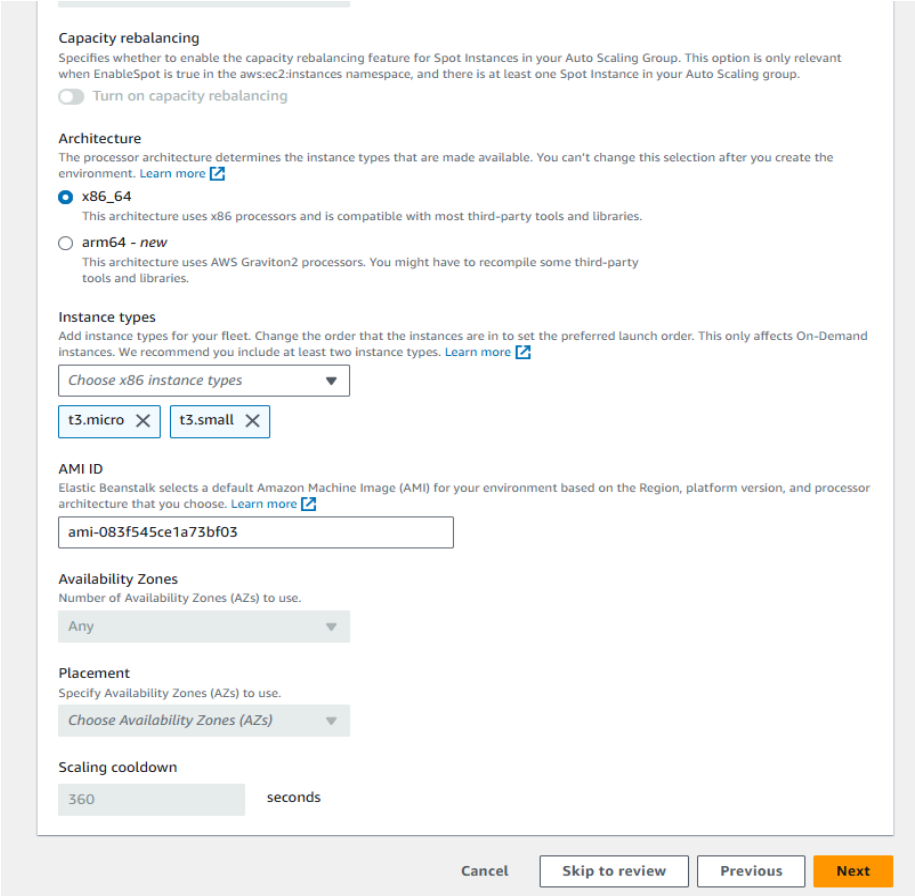


5. Now comeback to Elastic Beanstalk page and from the drop down menu select the newly created key pair and instance profile

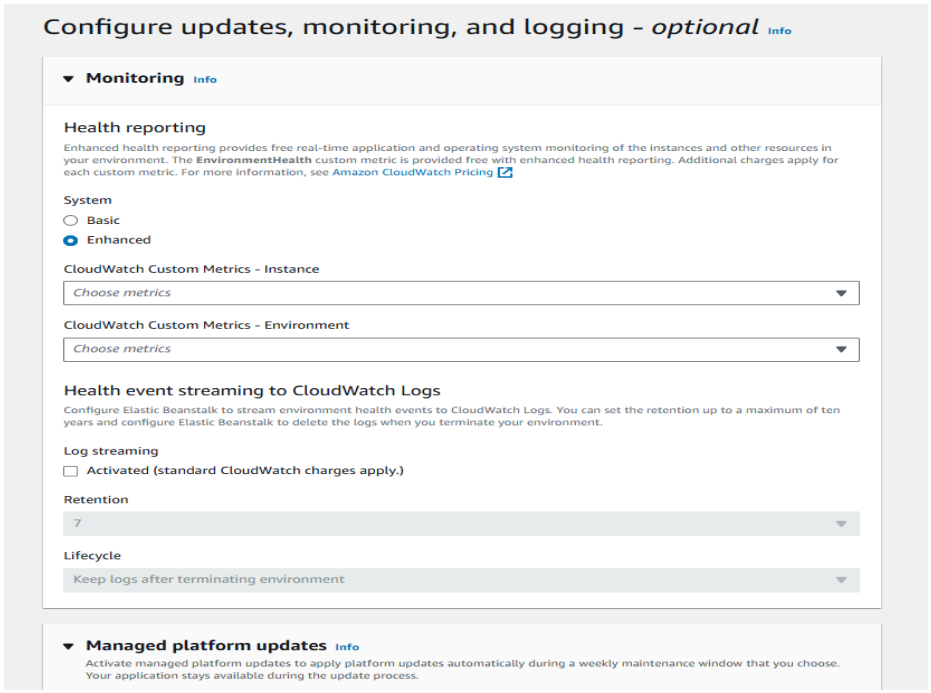


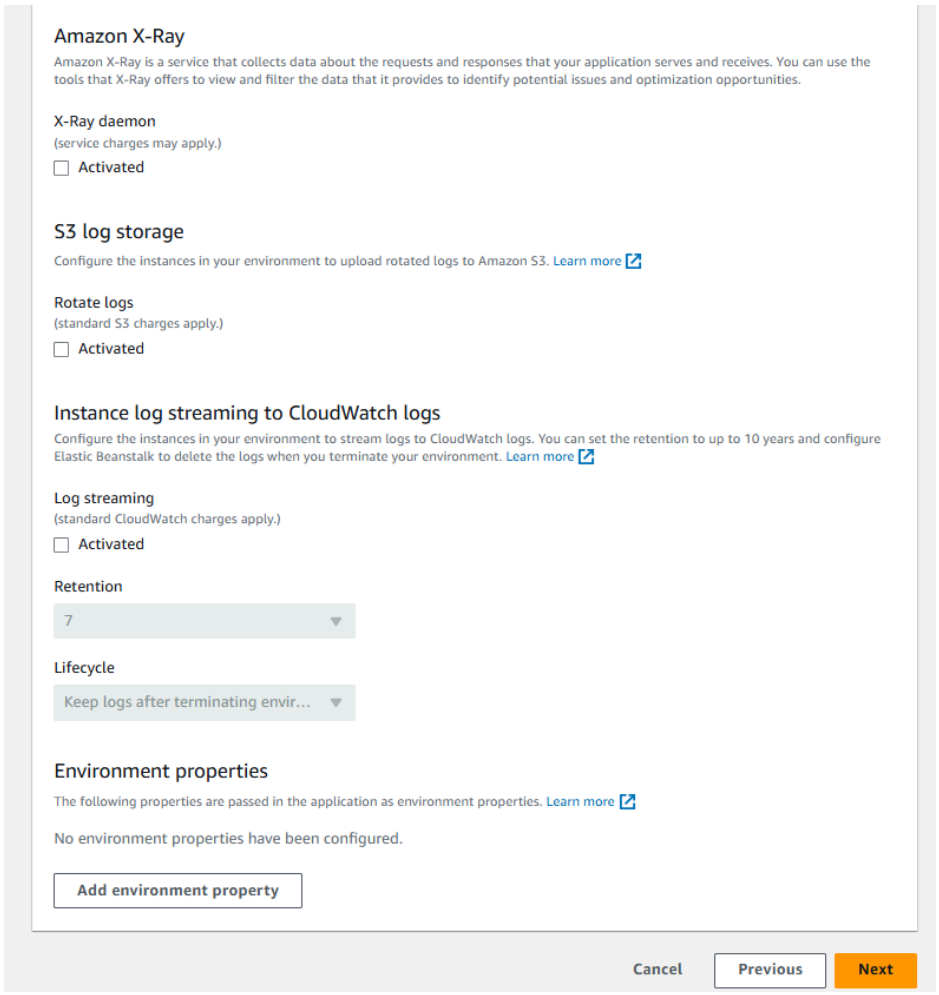
6. No changes further in the Configure instance traffic section



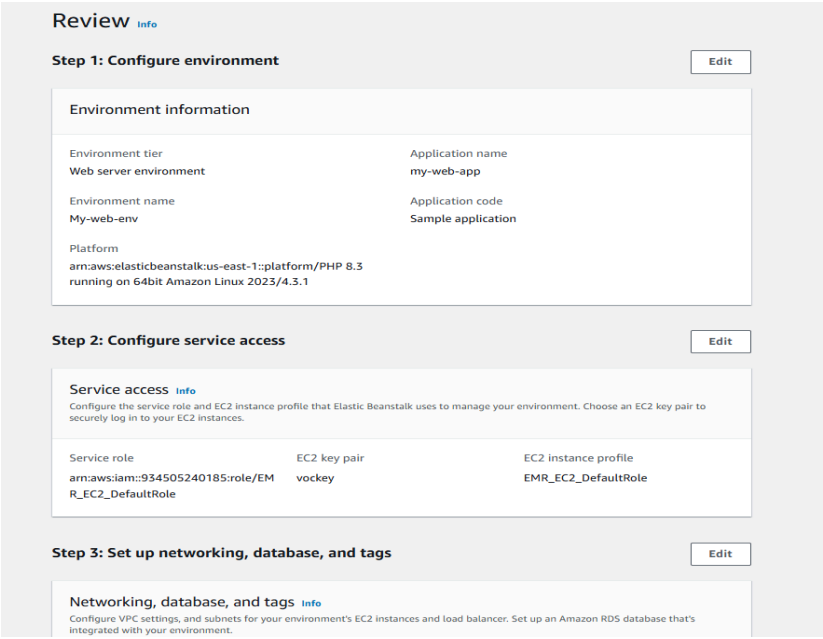


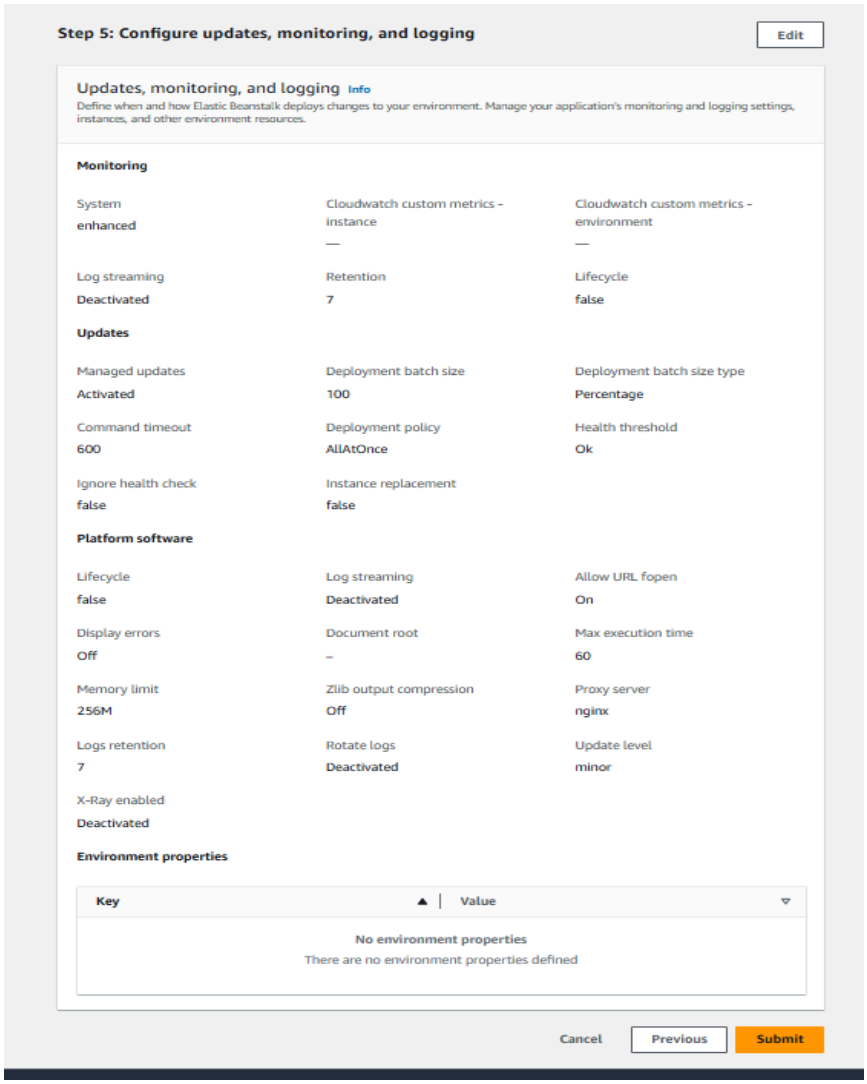
7. Again no changes in the configure updates, monitoring, and logging part just click on next.


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
Lifecycle
Keep logs after terminating environment
Managed platform updates Info
Activate managed platform updates to apply platform updates automatically during a weekly maintenance window that you choose. Your application stays available during the update process.

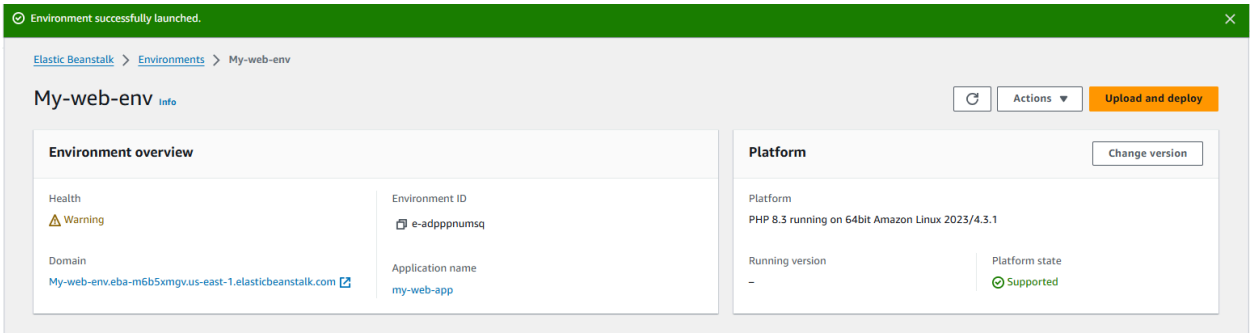

Amazon X-Ray
Amazon X-Ray is a service that collects data about the requests and responses that your application serves and receives. You can use the tools that X-Ray offers to view and filter the data that it provides to identify potential issues and optimization opportunities.
X-Ray daemon
(service charges may apply.)
Activated
S3 log storage
Configure the instances in your environment to upload rotated logs to Amazon S3. Learn more
Rotate logs
(standard S3 charges apply.)
Activated
Instance log streaming to CloudWatch logs
Configure the instances in your environment to stream logs to CloudWatch logs. You can set the retention to up to 10 years and configure Elastic Beanstalk to delete the logs when you terminate your environment. Learn more
Log streaming
(standard CloudWatch charges apply.)
Activated
Retention
7
Lifecycle
Keep logs after terminating envir...
Environment properties
The following properties are passed in the application as environment properties. Learn more
No environment properties have been configured.
Add environment property
Cancel
Previous
Next

8. Now review the changes made and click on create application





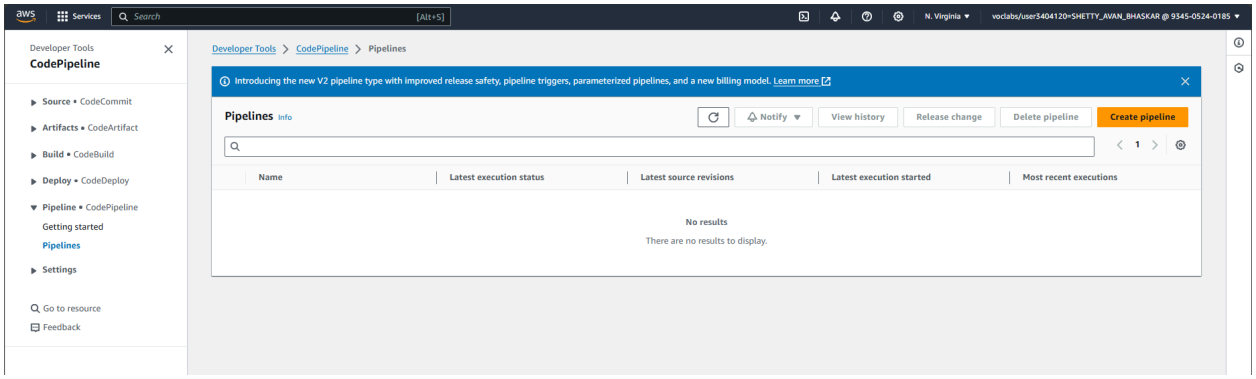
9. Your sample environment is created for you to deploy your application. By default, it creates an EC2 instance, a security group, an Auto Scaling group, an Amazon S3 Bucket, Amazon CloudWatch alarms and a domain name for your Application.



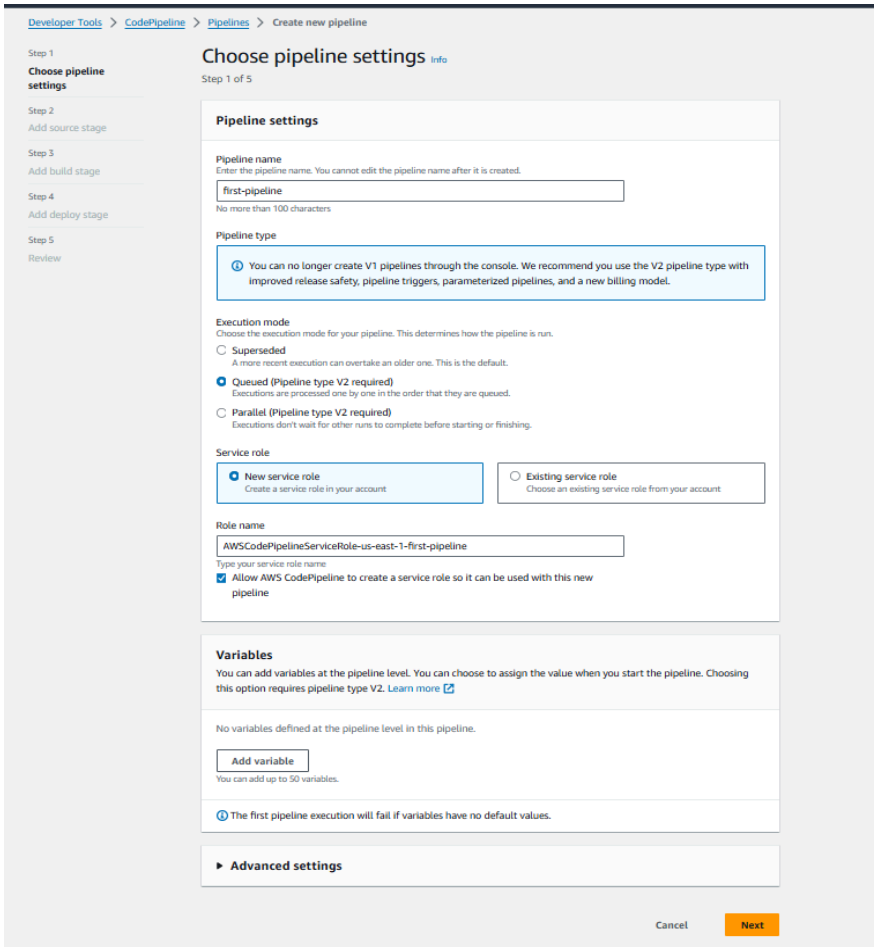
Pipeline creation:

1. Fork a github repo for aws codepipeline available as The pipeline takes code from the source and then performs actions on it.We don't need to code from scratch in this manner

2. Go to developer tools and select CodePipeline and create a new pipeline

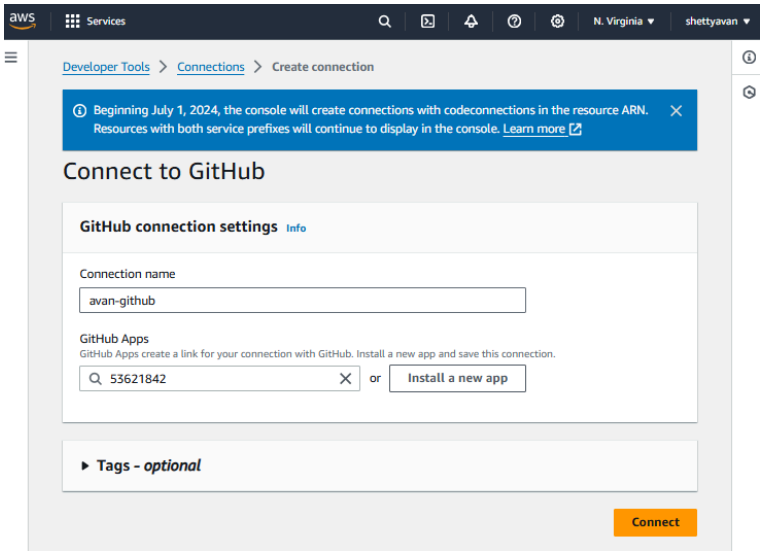


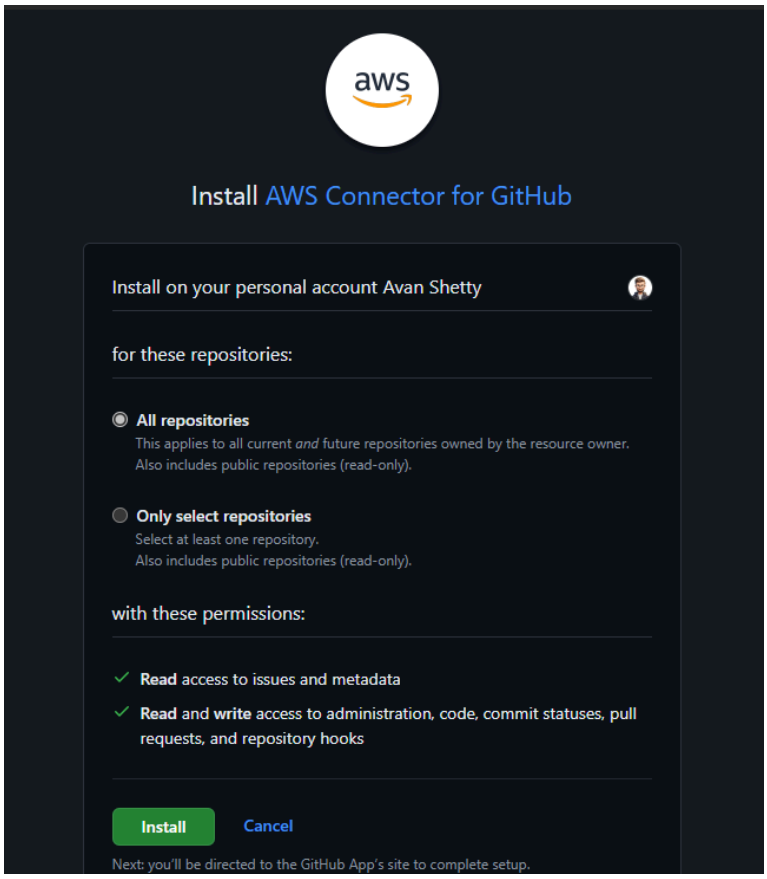
3. Name your pipeline and select the desired service role



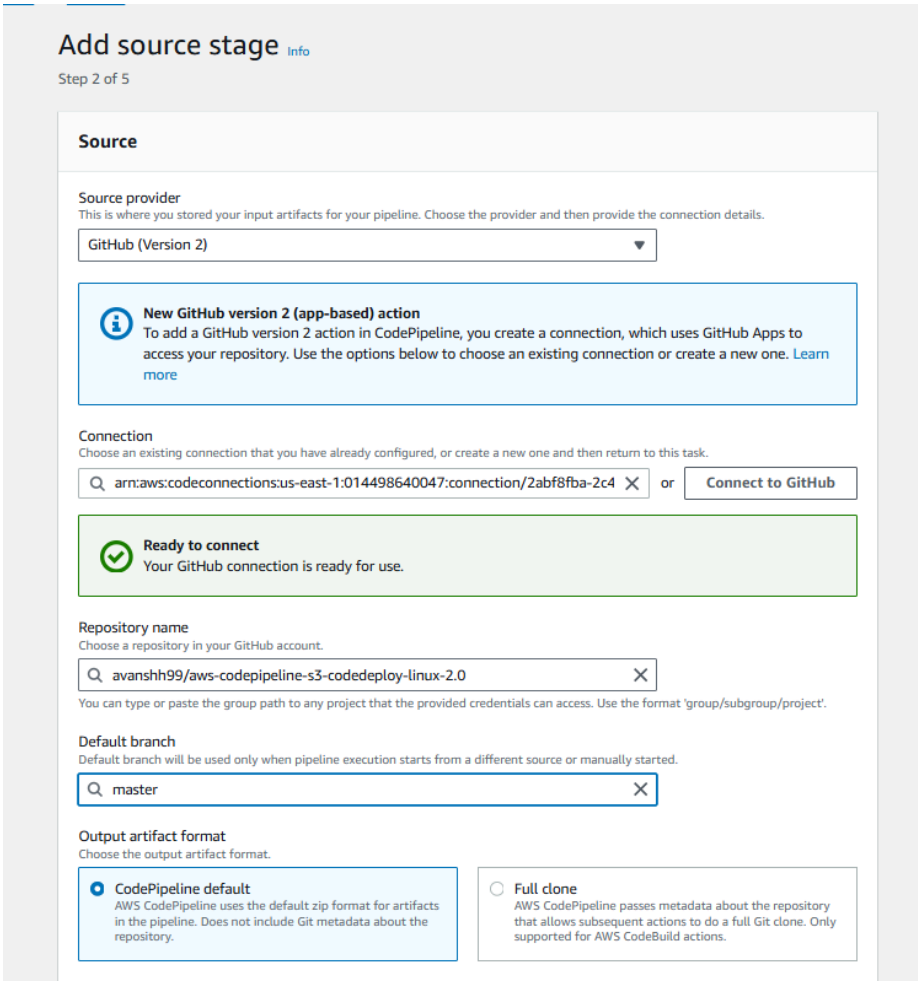
4. In the source stage select Github v2 as the provider and then connect your github connect so that the pipeline can access the forked source code .

a. For this purpose create aws github connection and with your credentials install the AWS under the forked repository

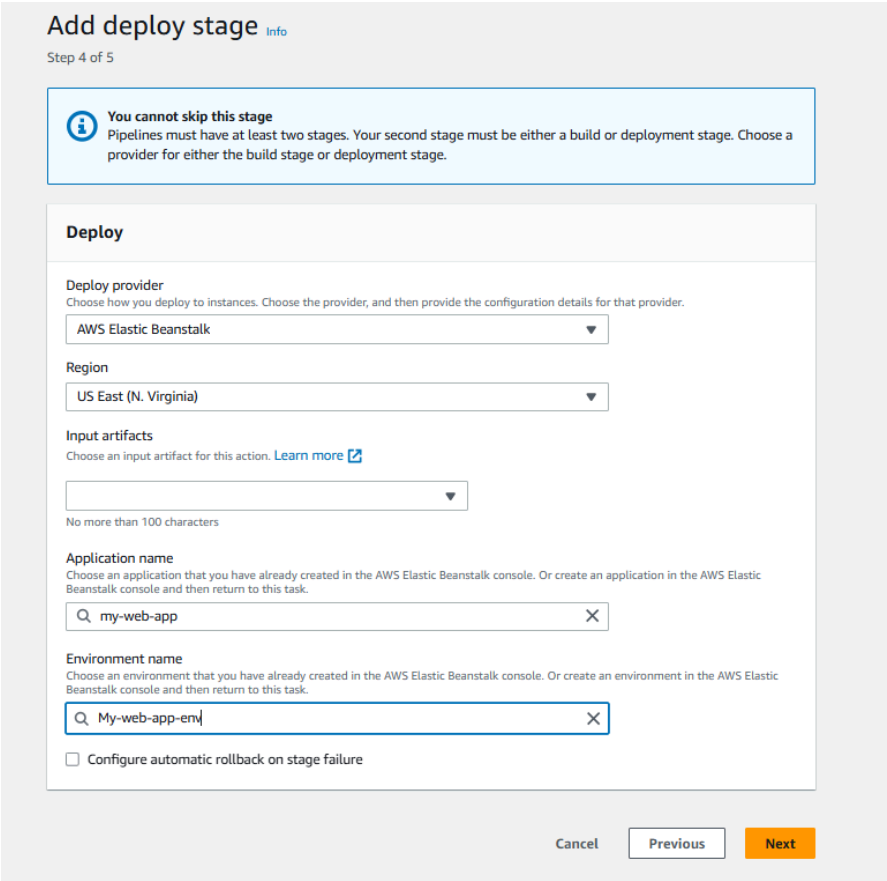




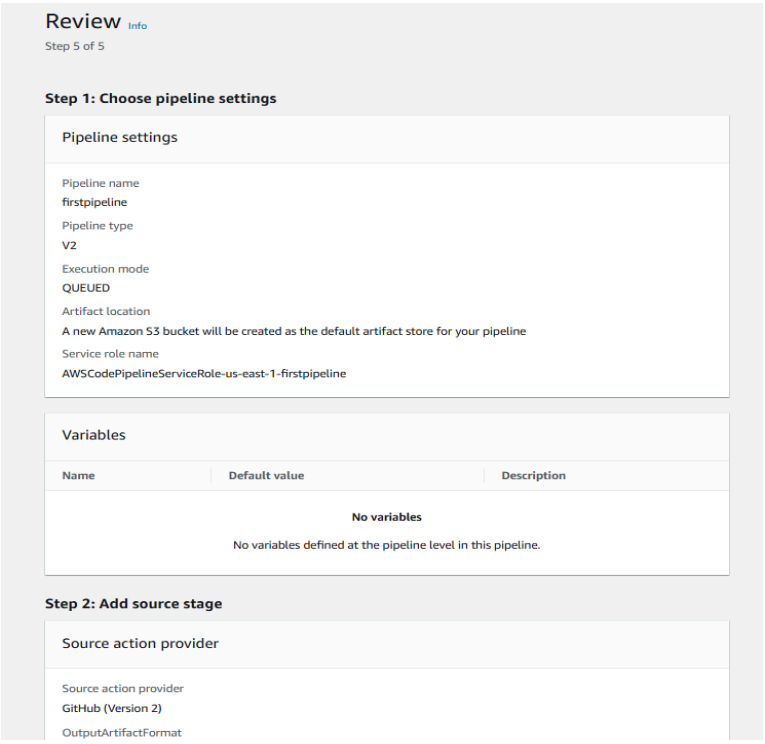
5. Once the connection is established from the drop down menu select the repository and the branch

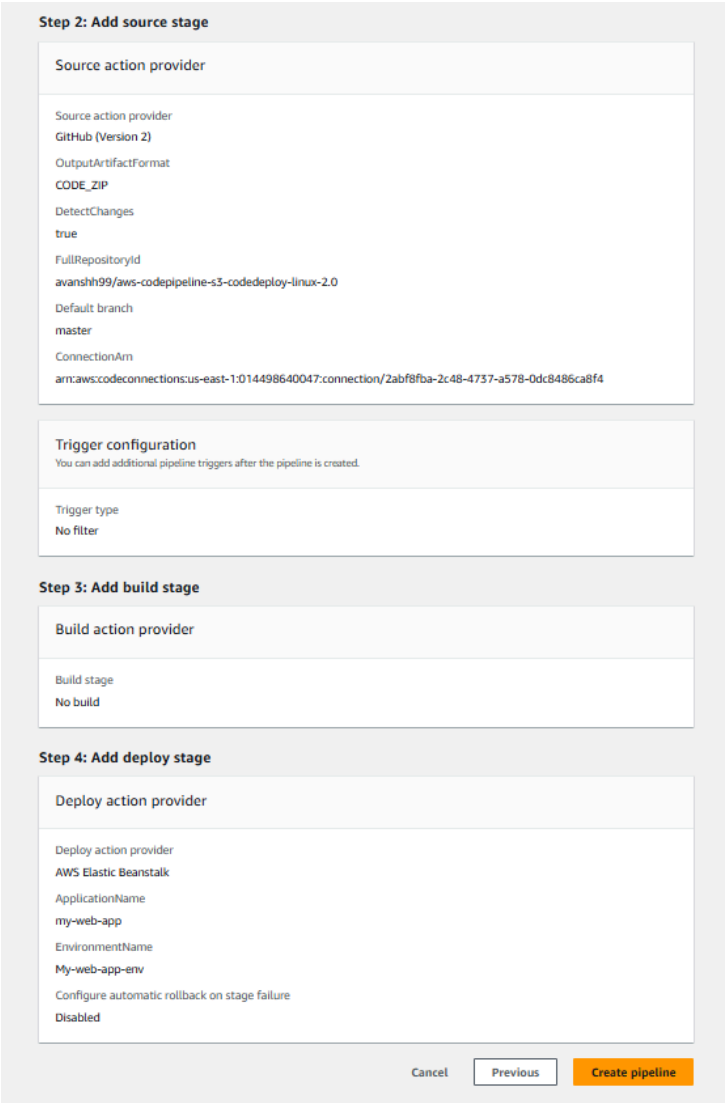


6. Skip the build stage part as we are not plugging in any build provider and in choose Beanstalk as the Deploy Provider, same region as the Bucket and Beanstalk, name and environment name.

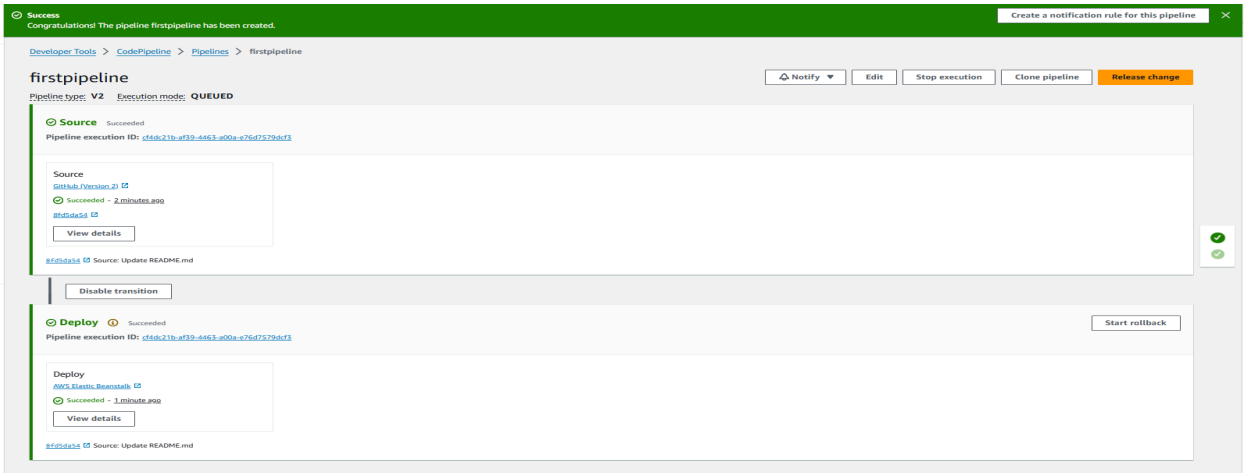


7. Review the settings and click on create pipeline

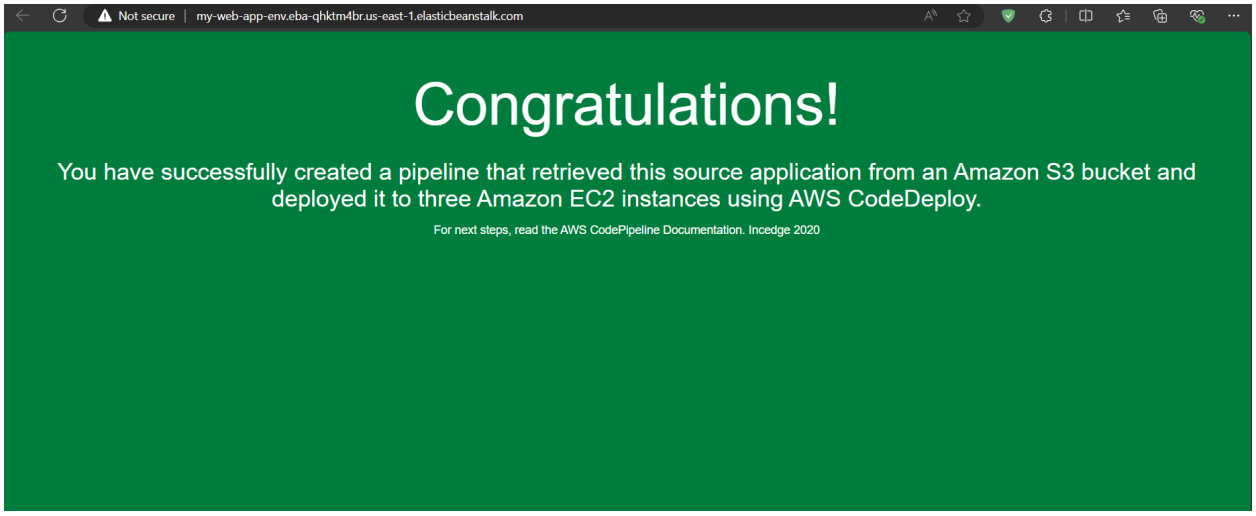




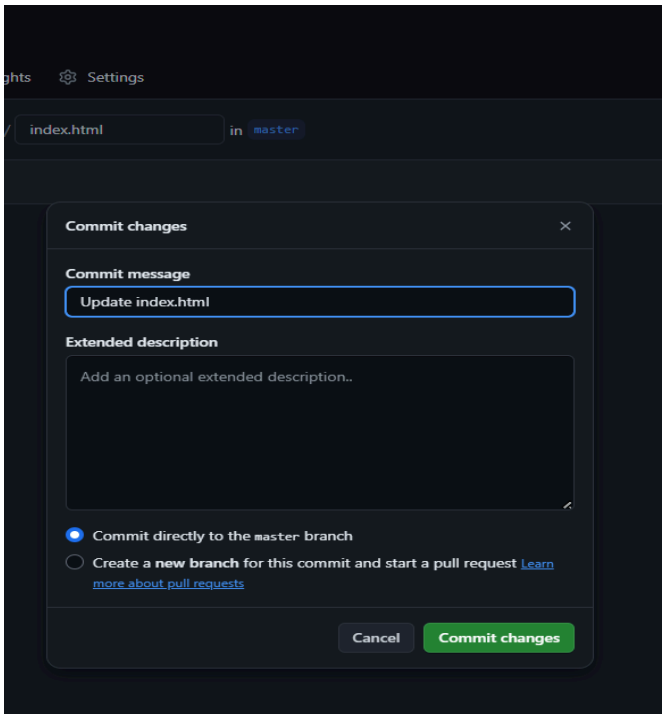
8. Once the Successfully created message appears, your pipeline is created.Then go ahead and check the URL provided in the EBS environment.



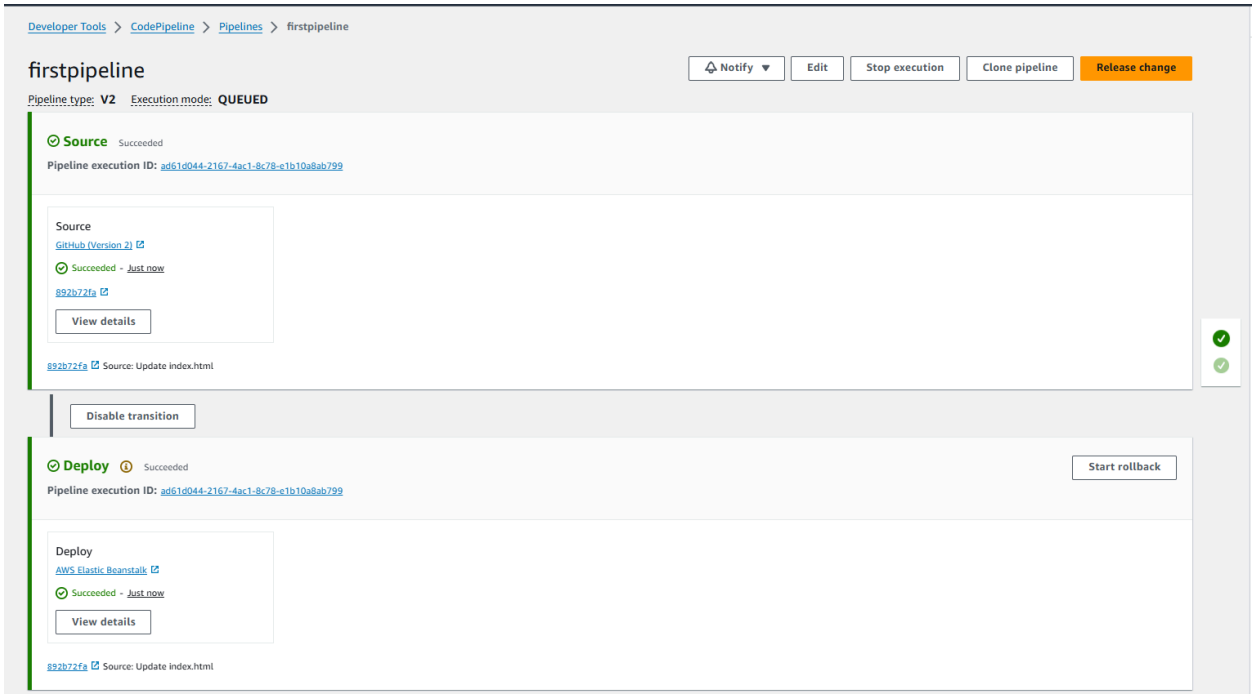
9. This is the website hosted from that forked repo in our beanstalk environment

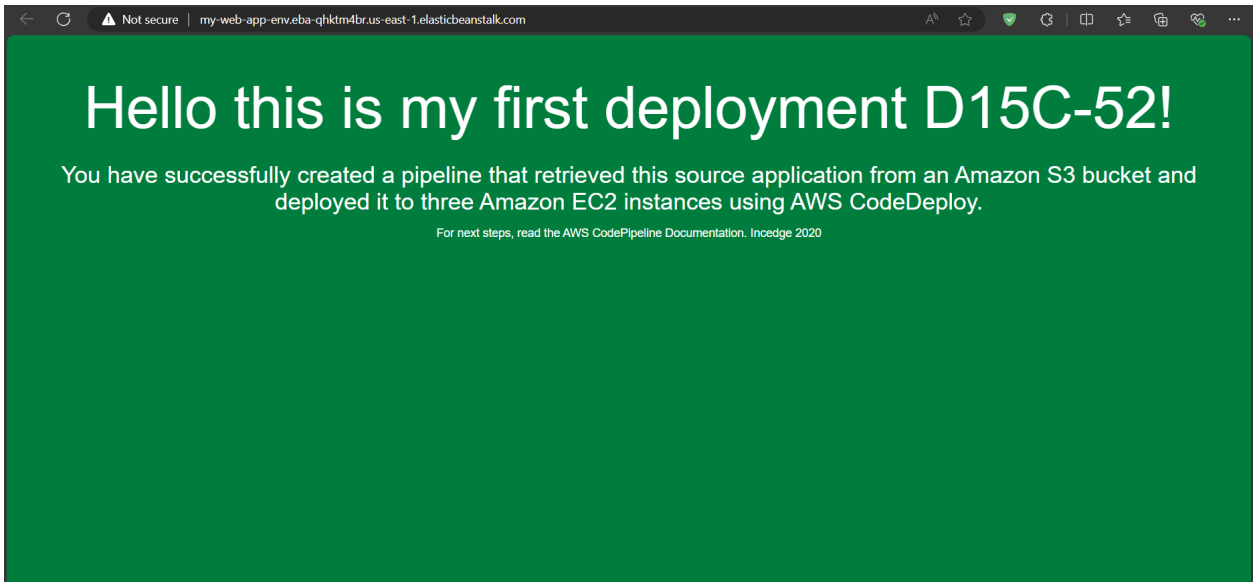


10. Go to the repository and make the changes in the index.html file and commit them



11.The changes that are committed can be noticed in the source panel in real time and to view the changes check the url (refresh it) and you can view the changes once the deployment section shows success.


Developer Tools > CodePipeline > Pipelines > firstpipeline
firstpipeline
Notify
Edit
Stop execution
Clone pipeline
Release change
Pipeline type: V2 Execution mode: QUEUED
Source Succeeded
Pipeline execution ID: ad61d044-2167-4ac1-8c78-e1b10a8ab799
Source
GitHub (Version 2)
Succeeded - Just now
892b72fa
View details
892b72fa Source: Update index.html
Disable transition
Deploy Succeeded
Start rollback
Pipeline execution ID: ad61d044-2167-4ac1-8c78-e1b10a8ab799
Deploy
AWS Elastic Beanstalk
Succeeded - Just now
View details
892b72fa Source: Update index.html


Not secure | my-web-app-env.eba-qhktm4br.us-east-1.elasticbeanstalk.com
Hello this is my first deployment D15C-52!
You have successfully created a pipeline that retrieved this source application from an Amazon S3 bucket and deployed it to three Amazon EC2 instances using AWS CodeDeploy.
For next steps, read the AWS CodePipeline Documentation. Incedge 2020