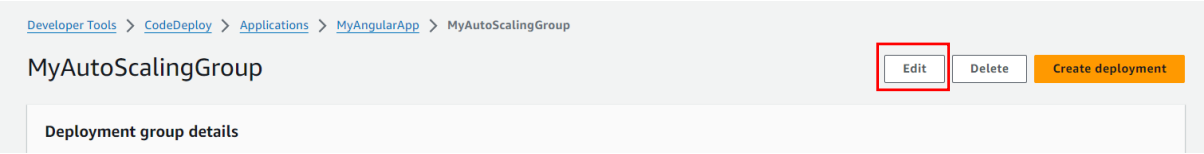
😊 In-Place Rolling Deployment

The text explains an improved method for deploying updates to a web application hosted on multiple EC2 instances behind a load balancer. Instead of updating all instances at once, which causes the application to be temporarily unavailable, the *in-place rolling deployment* method updates the instances one at a time. This ensures that the application remains online during the update process, with some instances still serving the old version while others are being updated to the new version.

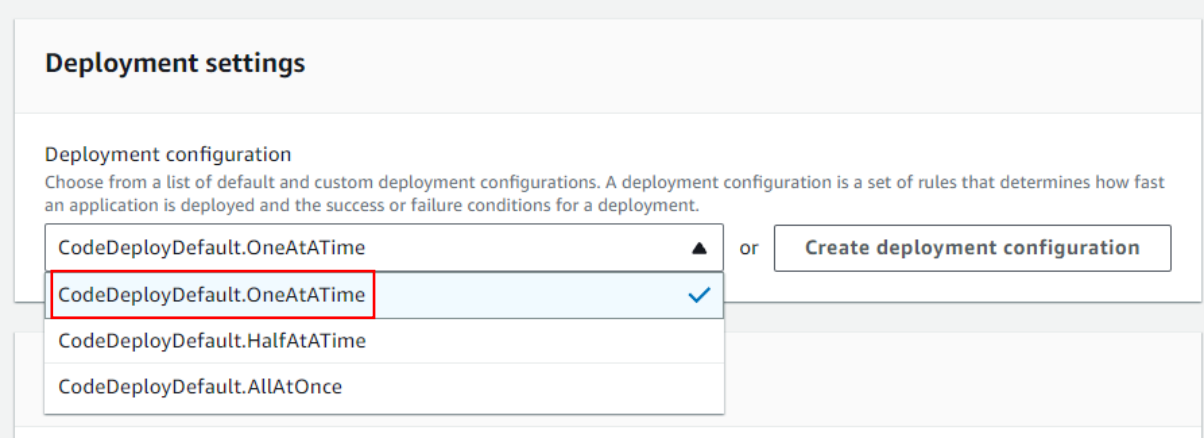
The step-by-step guide walks you through changing the deployment settings in AWS CodeDeploy to enable this rolling deployment, updating the application version, triggering the pipeline, and verifying the deployment progress. The goal is to ensure that the web application remains accessible without downtime during updates, making it more reliable for production environments.

😄 To begin with the Lab:

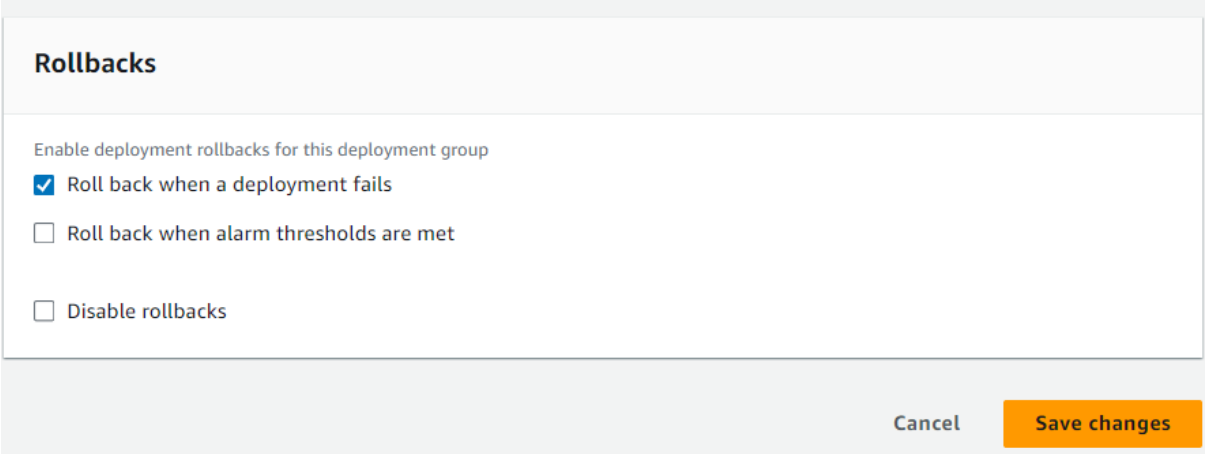
1. In the previous lab, we used in-place all-at-once deployment method to deploy our application to multiple EC2 instances behind a load balancer. However, we saw that during a deployment all instances are taken offline and our web application becomes unavailable for a few minutes. This is not acceptable for a production-level application. Then, do we have any alternative to this deployment method? Well, we can deploy to the instances one by one or even ten by ten if we have dozens of them. Again, the deployments will be performed on the existing instances, and we call these types of deployments in-place rolling deployments on CodeDeploy.
2. The first thing that you need to do is go to the Application in CodeDeploy and open your Deployment group then click on edit button.



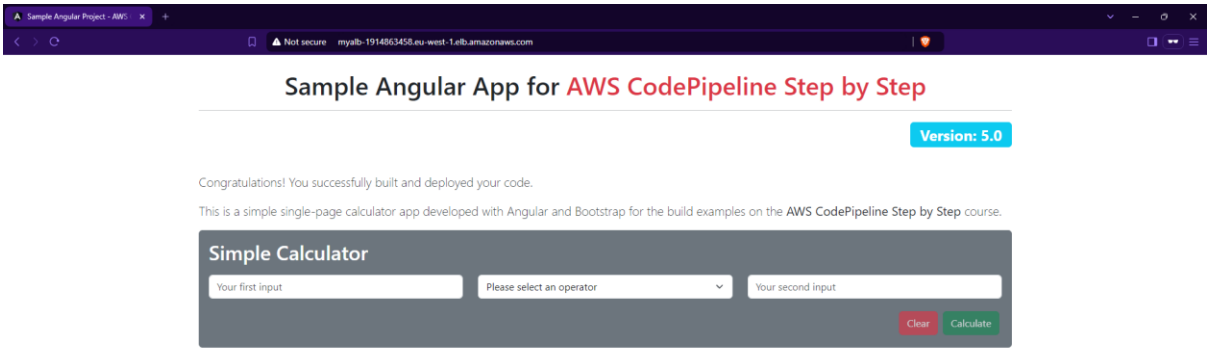
3. Now you need to scroll down to deployment settings and then expand the deployment configuration, after that, you need to choose One at a time



4. Then scroll down to the bottom and expand the advanced settings option, move to the bottom of the page, here you will see an option for Rollback, enable this feature and click on Save changes.



5. Currently you can see that the current version of our website is 5.0 so, let's go to our project in VS Code and change the Version to 6.0. Then commit your changes and push your changes to the repository.
6. It will trigger the pipeline, and your pipeline will be executed.



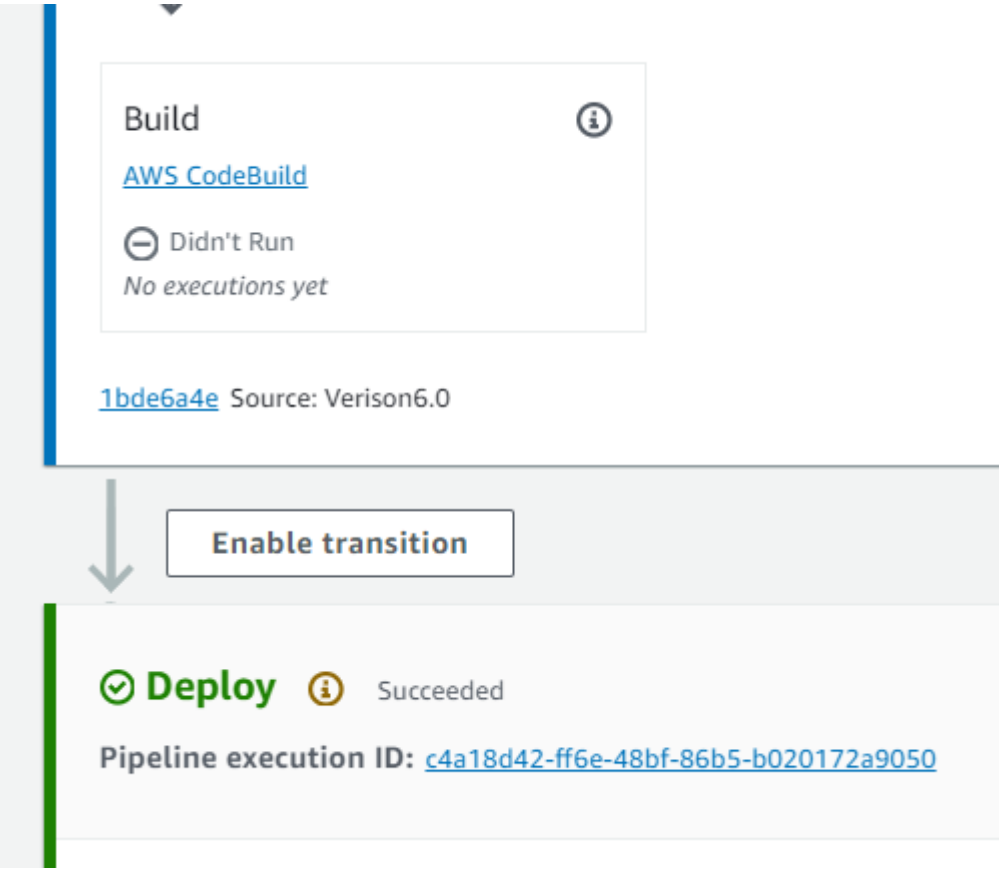
7. Below we updated the version and pushed the changes to our repository.

   git commit -a -m "Version6.0"
   git push origin master

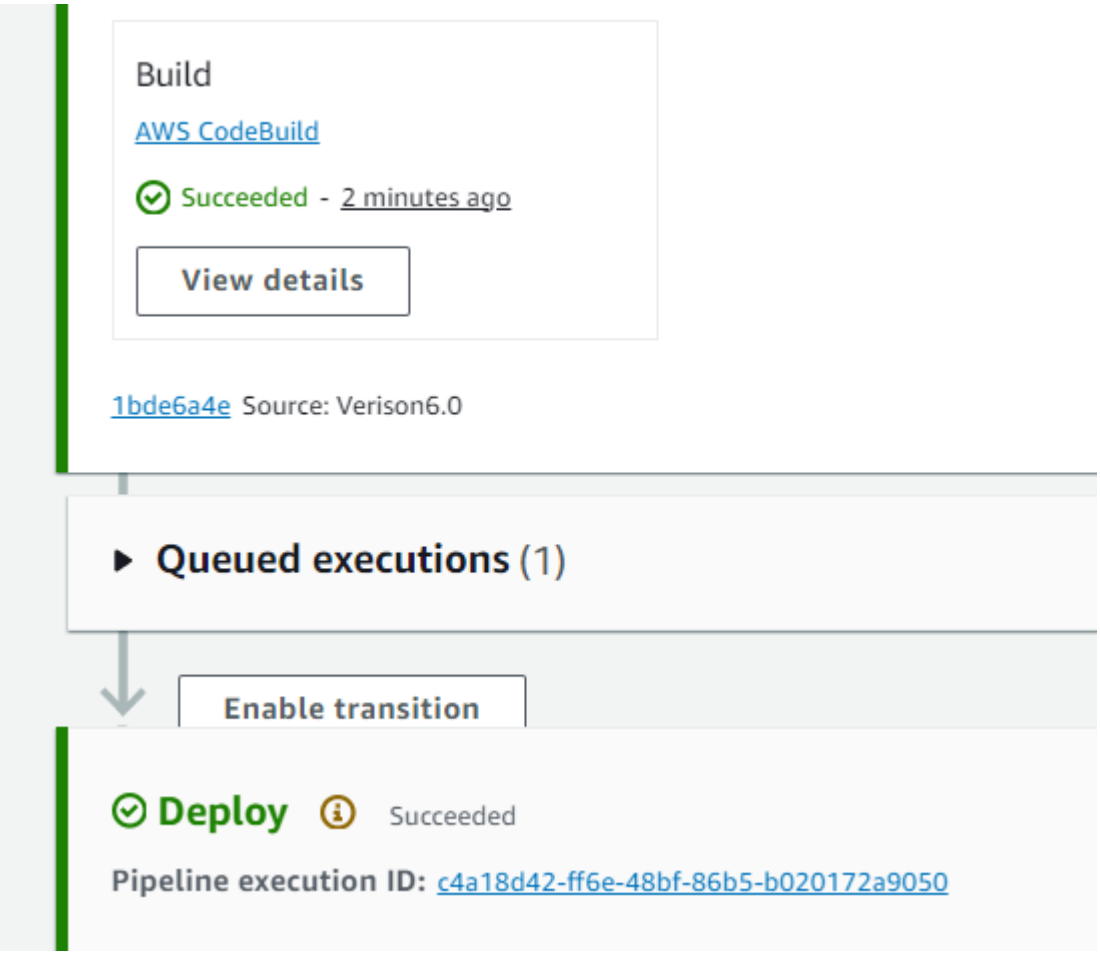
```
PS C:\Users\PULKIT\Downloads\my-angular-project> git commit -a -m "Verison6.0"
warning: in the working copy of 'src/app/app.component.html', LF will be replaced by CRLF the next time Git touches it
[master 1bde6a4] Verison6.0
 1 file changed, 1 insertion(+), 1 deletion(-)
PS C:\Users\PULKIT\Downloads\my-angular-project> git push origin master
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 12 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 434 bytes | 434.00 KiB/s, done.
Total 5 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Validating objects: 100%
To https://git-codecommit.eu-west-1.amazonaws.com/v1/repos/DemoAngularRepo
   ded03f0..1bde6a4  master -> master
PS C:\Users\PULKIT\Downloads\my-angular-project>
```

8. Now once your pipeline gets executed you need to disable the transition between your build stage and deploy stage and wait until your build stage gets executed.

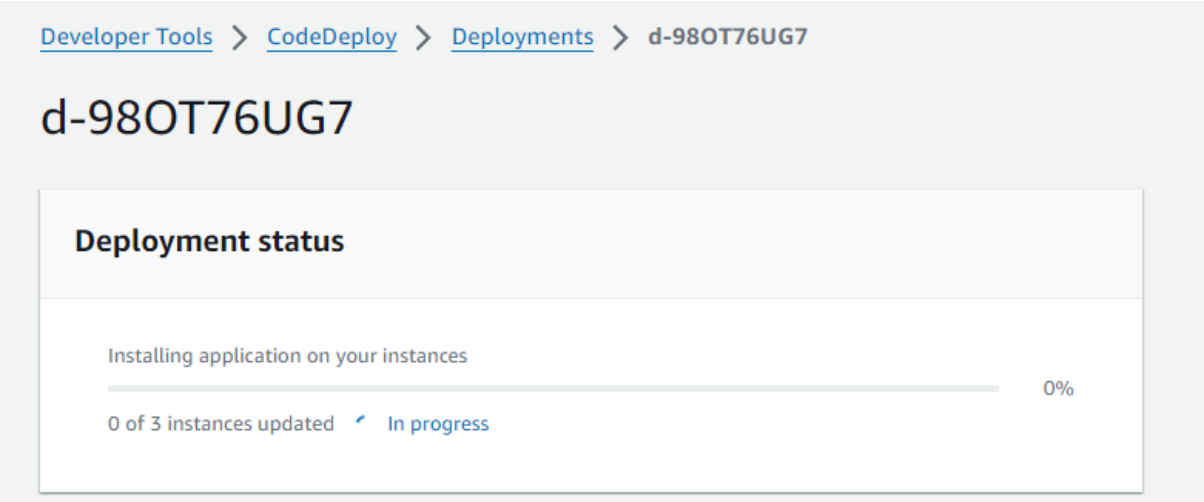
9. After that you need to enable the transition and wait until your deploy stage gets he option to view details.



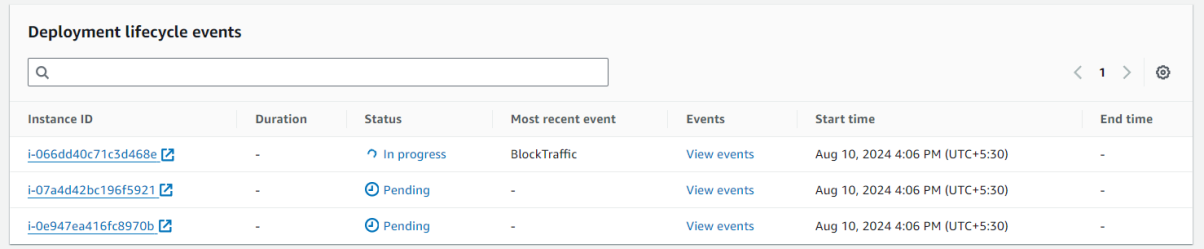
10. Now enable your queued execution and wait for view details options in deploy stage.



11. Once you have enabled the transition then you need to wait until the view details option is showing. You need to click on it then you will find the link to go into CodeDeploy, here you will see the deployment status.



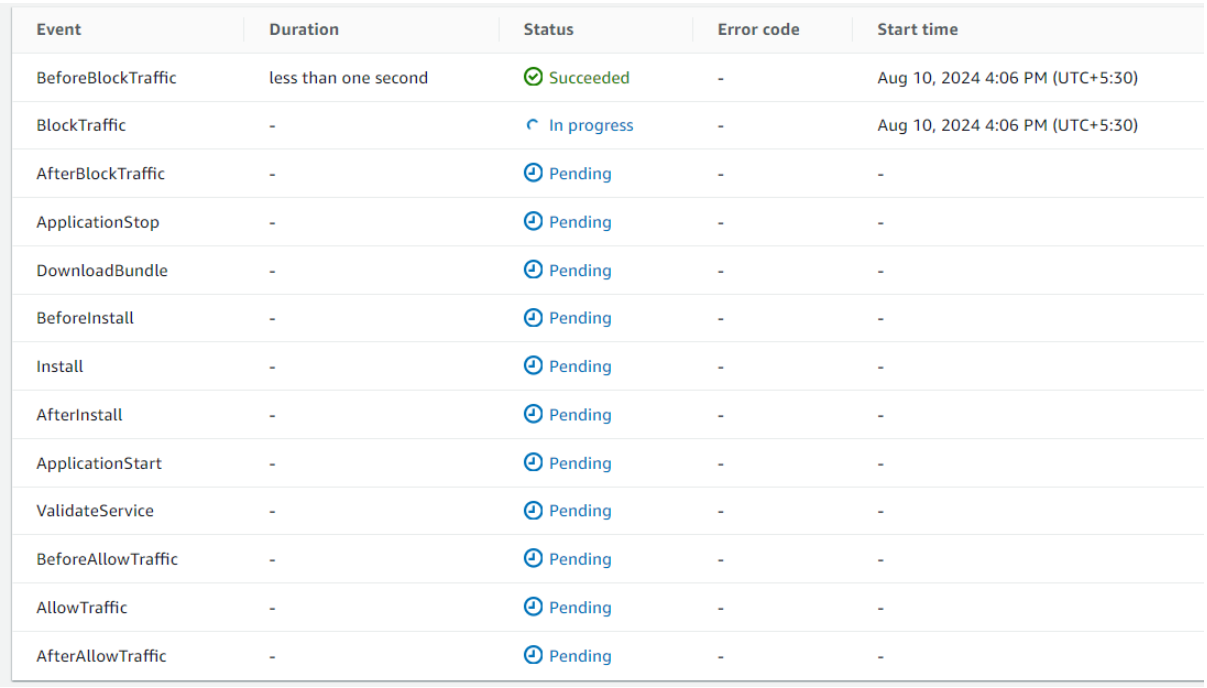
12. Then scroll down and you will see that one instance is currently in progress and others are in a pending state, which was not the case with In-place all-at-once deployment.



Deployment lifecycle events

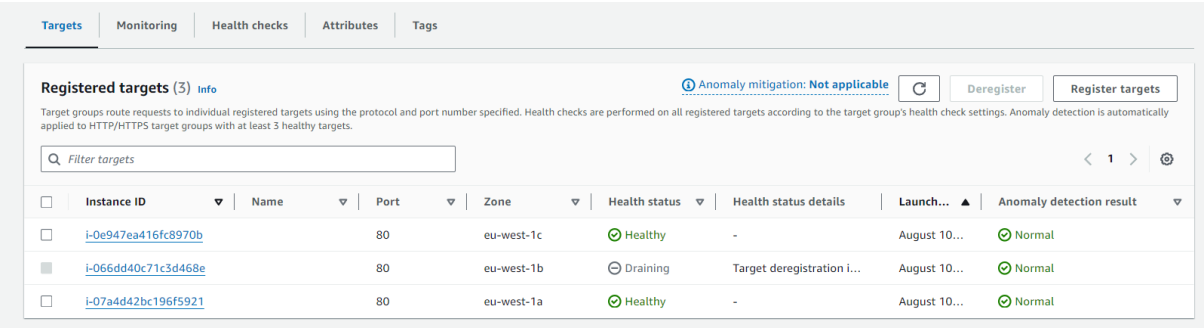
Instance ID	Duration	Status	Most recent event	Events	Start time	End time
i-066dd40c71c3d468e	-	In progress	BlockTraffic	View events	Aug 10, 2024 4:06 PM (UTC+5:30)	-
i-07a4d42bc196f5921	-	Pending	-	View events	Aug 10, 2024 4:06 PM (UTC+5:30)	-
i-0e947ea416fc8970b	-	Pending	-	View events	Aug 10, 2024 4:06 PM (UTC+5:30)	-

13. Now if you click on view events on your instance then you can see the status of the events happening on your instance.



Event	Duration	Status	Error code	Start time
BeforeBlockTraffic	less than one second	Succeeded	-	Aug 10, 2024 4:06 PM (UTC+5:30)
BlockTraffic	-	In progress	-	Aug 10, 2024 4:06 PM (UTC+5:30)
AfterBlockTraffic	-	Pending	-	-
ApplicationStop	-	Pending	-	-
DownloadBundle	-	Pending	-	-
BeforeInstall	-	Pending	-	-
Install	-	Pending	-	-
AfterInstall	-	Pending	-	-
ApplicationStart	-	Pending	-	-
ValidateService	-	Pending	-	-
BeforeAllowTraffic	-	Pending	-	-
AllowTraffic	-	Pending	-	-
AfterAllowTraffic	-	Pending	-	-

14. Now go to the Target group in EC2, here you will see that only one instance is draining, and others are still working. Now if you go to your website and refresh the page this time it is online because we have two more instances serving the previous version while one of them is being updated



Targets | Monitoring | Health checks | Attributes | Tags

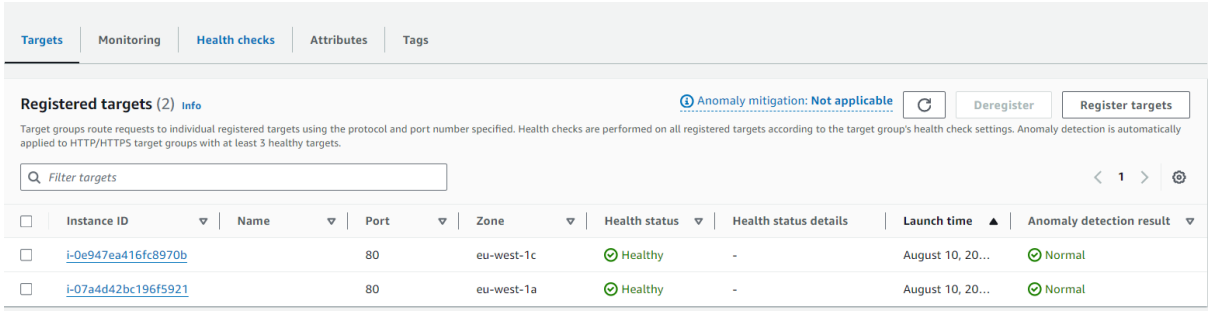
Registered targets (3) Info

Anomaly mitigation: Not applicable | Deregister | Register targets

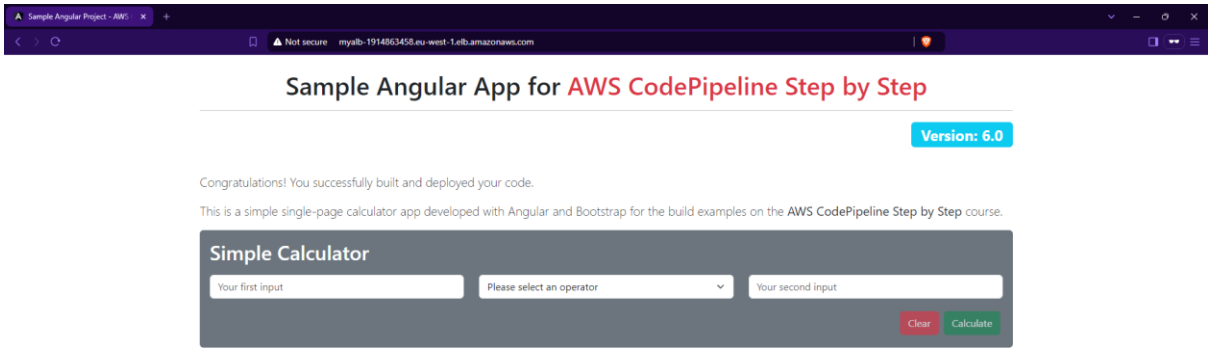
Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.

Filter targets

Instance ID	Name	Port	Zone	Health status	Health status details	Launch...	Anomaly detection result
i-0e947ea416fc8970b		80	eu-west-1c	Healthy	-	August 10...	Normal
i-066dd40c71c3d468e		80	eu-west-1b	Draining	Target deregistration i...	August 10...	Normal
i-07a4d42bc196f5921		80	eu-west-1a	Healthy	-	August 10...	Normal



15. Once one of your instances is updated then go to your website and refresh the page, here you will see that the version has changed to 6.0 as soon as one instance is updated.



16. Similarly, you can see the same process for other instances.