



NextWork.org

Deploy an App with CodeDeploy



Shikha Gupta



Shikha Gupta
NextWork Student

[NextWork.org](https://www.nextwork.org)

Introducing today's project!

What is AWS CodeDeploy?

AWS CodeDeploy automates and manages application deployments, supporting EC2, Lambda, and on-premises servers. It ensures reliable, consistent releases with features like rollback and flexible strategies.

How I'm using AWS CodeDeploy in this project

I used AWS CodeDeploy to automate EC2 deployments, handle rollbacks, and monitor the process for efficiency and reliability.

One thing I didn't expect...

I didn't expect the extent of detail required for configuring deployment strategies and handling rollbacks. It turned out to be crucial for ensuring smooth, error-free updates.

This project took me...

The project took about 3 hours, including setup, configuration, and testing of the deployment process with AWS CodeDeploy.



Shikha Gupta
NextWork Student

NextWork.org

Set up an EC2 instance

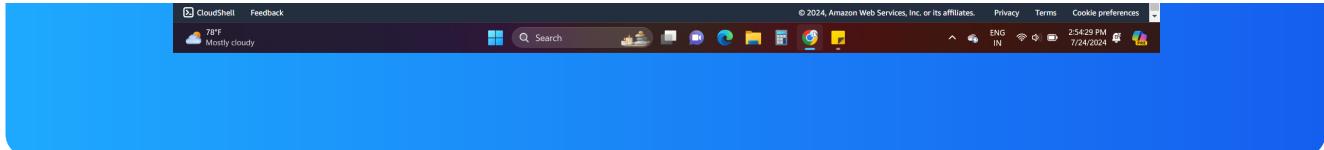
I set up an EC2 instance and VPC to securely host my application and manage network connectivity.

We manage production and development environments separately to ensure stability and avoid affecting live services while testing new features.

To set up my EC2 instance and VPC, I used AWS CloudFormation.

The screenshot shows the AWS CloudFormation console with the 'CloudFormation' tab selected. On the left, the 'Stacks' section lists 'CloudFormation > Stacks > NextWorkEC2VPCStack' with 2 stacks. The right panel displays the 'Events' tab for the 'NextWorkEC2VPCStack' stack, which has 40 events. The events table includes columns for Timestamp, Logical ID, Status, and Detailed status. The events listed are:

Timestamp	Logical ID	Status	Detailed status
2024-07-24 14:53:41 UTC+0530	NextWorkEC2VPCStack	CREATE_COMPLETE	-
2024-07-24 14:53:39 UTC+0530	DeployRoleProfile	CREATE_COMPLETE	-
2024-07-24 14:53:03 UTC+0530	WebServer	CREATE_COMPLETE	-
2024-07-24 12:47:49 UTC+0530	NextWorkEC2VPCStack	CREATE_IN_PROGRESS	CONFIGURATION_COMPLETE
2024-07-24 14:52:43 UTC+0530	NextWorkEC2VPCStack	CREATE_IN_PROGRESS	CONFIGURATION_COMPLETE
2024-07-24 14:52:43 UTC+0530	WebServer	CREATE_IN_PROGRESS	CONFIGURATION_COMPLETE



Shikha Gupta
NextWork Student

NextWork.org

Bash scripts

Scripts are sets of automated commands. Bash is a scripting language used in Unix-like systems to write these scripts.

I used three scripts for my project's deployment

The first script I created was `install_dependencies.sh` and it installs all the dependencies required.

The second script I created was `The start_server.sh` script and this is responsible for starting the Tomcat and HTTPD services on our EC2 instance.

The third script I created was `The stop_server.sh` script and it is used to stop the Tomcat and HTTPD services on our EC2 instance.



Shikha Gupta
NextWork Student

NextWork.org

Bash scripts

```

version: 0.0
os: linux
files:
  - source: /target/nextwork-web-project.war
    destination: /usr/share/tomcat/webapps/
hooks:
  - BeforeInstall:
    - location: scripts/install_dependencies.sh
      timeout: 300
      runsas: root
  - ApplicationStop:
    - location: scripts/start_server.sh
      timeout: 300
      runsas: root
  - ApplicationStop:
    - location: scripts/stop_server.sh
      timeout: 300
      runsas: root

```

bash - t@172-31-5-197 ~ %

```

Writing objects: 100% (3/3), 298 bytes | 298.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: VCSBackend: git
remote: VCSBackend: git
To https://git-codecommit.ap-south-1.amazonaws.com/v1/repos/nextwork-web-project
  242095...4e9131a main -> main
branch 'main' set up to track "origin/main".
Shikha-TAN-Admin:~/environment/nextwork-web-project (main) $ 
```



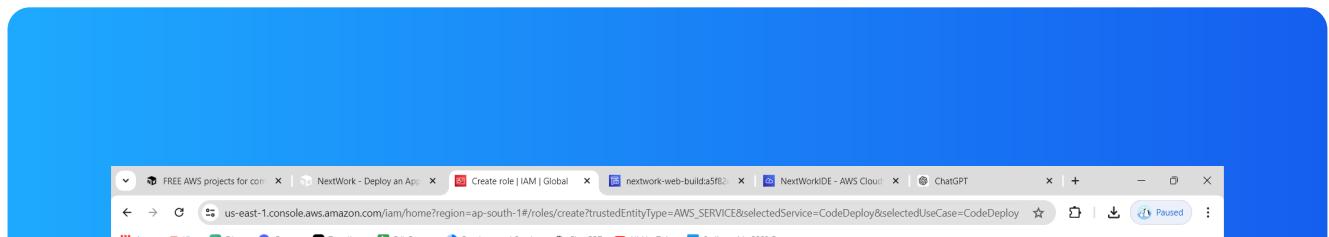
Shikha Gupta
NextWork Student

[NextWork.org](https://nextwork.org)

CodeDeploy's IAM Role

I created an IAM service role for CodeDeploy because CodeDeploy does not have access to EC2, so we're going to grant access using the AWS Managed AWSCodeDeployRole policy

To set up CodeDeploy's IAM role, I Choose CodeDeploy as the service and select CodeDeploy for the use case. Click Next: Permissions. You'll notice the CodeDeployRole default policy is suggested already—nice! That's all we need.



The screenshot shows the AWS IAM Policy editor interface. A policy named "AWSCodeDeployRole" is being edited. The policy document is as follows:

```

1- {
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "autoscaling:CompleteLifecycleAction",
        "autoscaling:DeleteLifecycleHook",
        "autoscaling:DescribeAutoScalingGroups",
        "autoscaling:DescribeLifecycleHooks",
        "autoscaling:PutLifecycleHook",
        "autoscaling:RegisterLifecycleActionHeartbeat",
        "autoscaling:ReplaceAutoScalingGroup",
        "autoscaling:CreateOrUpdateTags",
        "autoscaling:UpdateAutoScalingGroup",
        "autoscaling:EnableMetricsCollection",
        "autoscaling:DescribePolicies",
        "autoscaling:DescribeScheduledActions",
        "autoscaling:DescribeNotificationConfigurations",
        "autoscaling:SuspendProcesses"
      ]
    }
  ]
}

```

Below the policy document, there is a section titled "Set permissions boundary - optional" with a note: "Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting but can be used to delegate permission management to others. Learn more about permission boundaries".



Shikha Gupta
NextWork Student

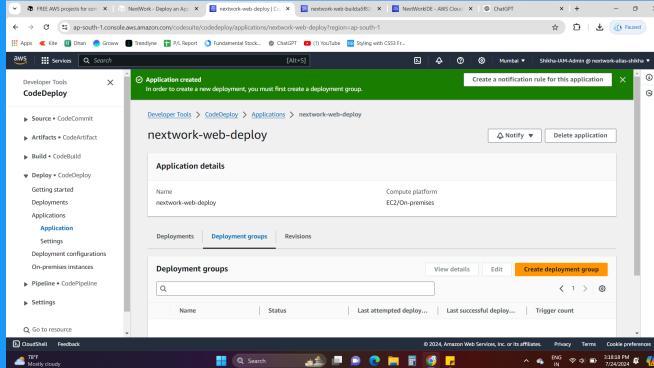
NextWork.org

CodeDeploy application

A CodeDeploy application works like a folder that holds all the settings it needs for a deployment. Think of this as setting up a template for deploying a web app. CodeDeploy applications helps to streamline the deployment process and ensures consist.

To create a CodeDeploy application, I had to select a compute platform, which means choosing where my application will be deployed, such as EC2 instances, Lambda functions, or on-premises servers.

The compute platform I chose was EC2 and on-premises because it offers flexibility for deploying on both cloud instances and existing servers.



Shikha Gupta
NextWork Student

[NextWork.org](https://www.nextwork.org)

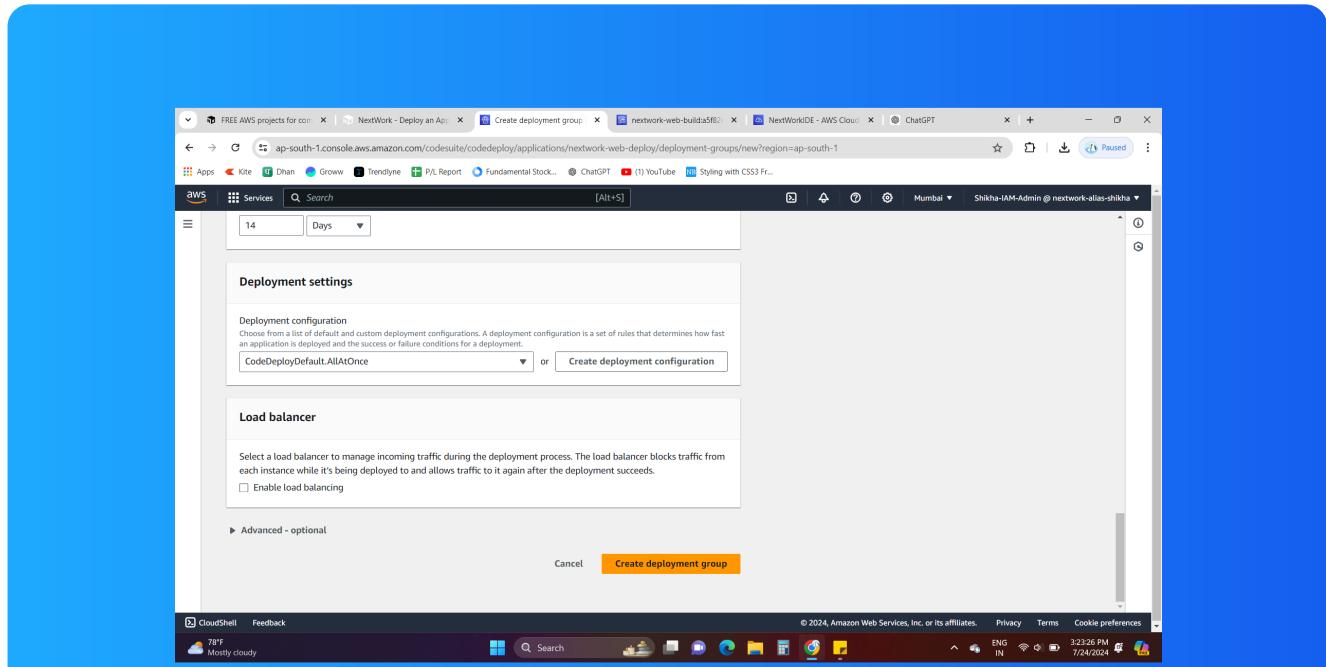
Deployment group

Deployment groups are the specific instructions for one particular deployment scenario. It defines which servers to use, how to deploy, and what settings to apply for that specific deployment.

Two key configurations for a deployment group

The environment configuration tells CodeDeploy what kind of servers you want to use for your application. In this project, it's Amazon EC2 instances!

The CodeDeploy Agent is like a helper installed on your servers that communicates with CodeDeploy to get the instructions for deploying your application. It ensures that the servers know what to do when a new version of your application need to deplo



Shikha Gupta
NextWork Student

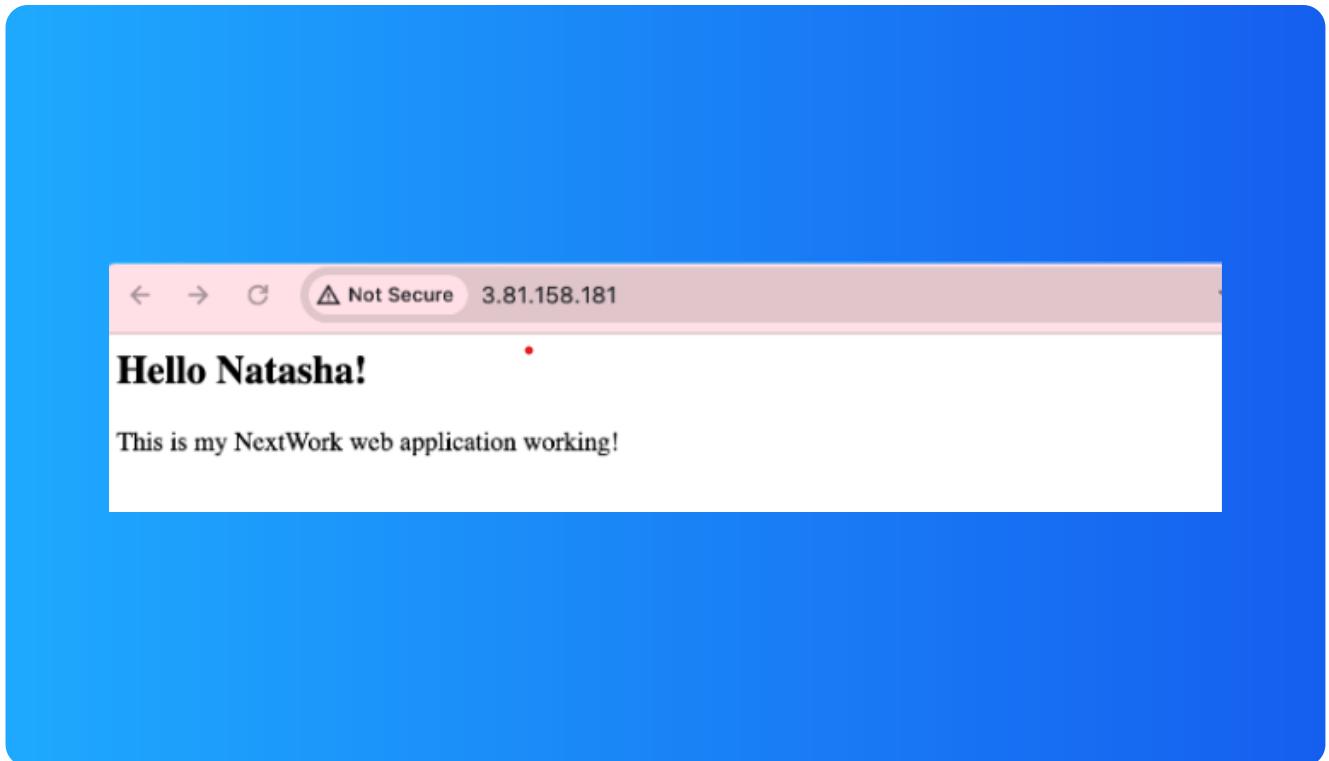
[NextWork.org](https://nextwork.org)

CodeDeploy application

To create my deployment, I had to set up a revision location, which means i had to copy uri of nextwork-web-build.zip

My revision location was my uri of nextwork-web-build.zip

To visit my web app, I had to visit <http://65.2.149.189/>



[NextWork.org](https://nextwork.org)

Everyone should be in a

job they love.

Check out nextwork.org for more projects

