

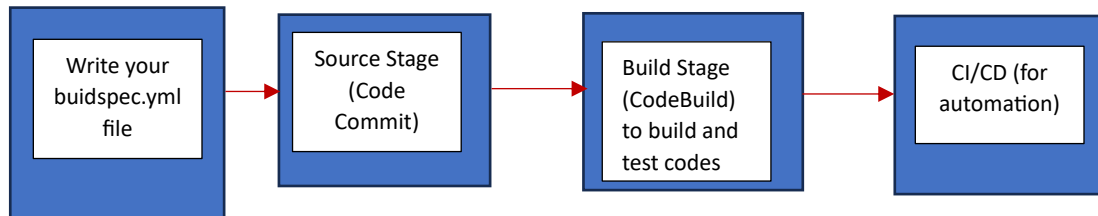
Project

Setup a two stages CI/CD pipeline using CodePipeline with build stage to simulate an application.

By

Aghogho Dre Aruoture

Road Map Schematics of the Project



Firstly, I wrote my buildspec.yml code and saved it as "buildspec.yml". It is good to note that by default, the buildspec file must be named buildspec.yml and placed in the root of your source directory.

```
! dreAppbuildSpec.yaml > {} phases > {} install > {} pre_build > [ ] commands
1  version: 0.2
2
3  phases:
4    install: # describes the app to be installed "Dre-App"
5      runtime-versions:
6        python: 3.9 # It will install python 3.8
7      pre_build: # Checks if all softwares has been installed successfully
8        commands:
9          - echo "CHECKING IF ALL SOFTWARE HAS BEEN INSTALLED"
10         - python3 --version #checks the version of python installed
11      build: # in here, actually runs the command to build your code
12        commands:
13          - echo "BUILD STARTED AT `date` "
14          - pwd
15          - echo "Dre-App is being Built"
16      post_build: # final stage to check if the code ran successfully
17        commands:
18          - echo "BUIL Dre-App COMPLETED at `date` "
19          - ls -l
20
21      artifacts: #these are the files from your source stage you want to produce as outputs
22        files:
23          - '**/*' # means everything in your source stage
```

Secondly, for my source code being AWS CodeCommit, I created a repository.

Hint:

- Make sure you have one of the git clients like git installed in your local system.
- Attach an IAM permission “AWSCodeCommitPowerUser” to the user you choose to choose.

The screenshot shows the AWS IAM console interface. On the left is a navigation menu with sections like 'Identity and Access Management (IAM)', 'Access management', 'Access reports', and 'Related consoles'. The main content area is titled 'admin_user_2023' and includes a 'Delete' button. Below the title is a 'Summary' section with details: ARN (arn:aws:iam:866756323262:user/admin_user_2023), Console access (Enabled without MFA), Access key 1 (Create access key), Created date (April 04, 2023, 23:09 UTC-02:30), and Last console sign-in (Today). Below the summary are tabs for 'Permissions', 'Groups (1)', 'Tags', 'Security credentials', and 'Access Advisor'. The 'Permissions' tab is active, showing 'Permissions policies (2)'. A table lists two policies: 'AdministratorAccess' (AWS managed - job function, attached via Group admin_group_2023) and 'IAMUserChangePassword' (AWS managed, attached directly). At the bottom, it says 'Permissions boundary (not set)'.

The screenshot shows the 'Add permissions' wizard in the AWS IAM console. The breadcrumb trail is 'IAM > Users > admin_user_2023 > Add permissions'. The left sidebar shows 'Step 1 Add permissions' and 'Step 2 Review'. The main content area is titled 'Add permissions' with a sub-header 'Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more'. Below this are 'Permissions options' with three radio buttons: 'Add user to group' (selected), 'Copy permissions', and 'Attach policies directly'. The 'Attach policies directly' option is highlighted with a blue border. Below the options is a section 'Permissions policies (1/1136)' with a search bar containing 'codeco' and a filter dropdown set to 'All types' showing '3 matches'. A table lists three policies: 'AWSCodeCommitFullAccess', 'AWSCodeCommitPowerUser' (which is selected with a blue checkmark), and 'AWSCodeCommitReadOnly'. All three are 'AWS managed' and have '0' attached entities. At the bottom right are 'Cancel' and 'Next' buttons.

[IAM](#) > [Users](#) > [admin_user_2023](#) > Add permissions

Step 1
[Add permissions](#)

Step 2
Review

Review

The following policies will be attached to this user. [Learn more](#)

User details

User name
admin_user_2023

Permissions summary (1)

Name	Type	Used as
AWSCodeCommitPowerUser	AWS managed	Permissions policy

Cancel Previous **Add permissions**

[Permissions](#) | [Groups \(1\)](#) | [Tags](#) | [Security credentials](#) | [Access Advisor](#)

Permissions policies (3)

Permissions are defined by policies attached to the user directly or through groups.

Filter by Type: All types

<input type="checkbox"/>	Policy name	Type	Attached via
<input type="checkbox"/>	AdministratorAccess	AWS managed - job function	Group admin_group_2023
<input type="checkbox"/>	AWSCodeCommitPowerUser	AWS managed	Directly

- Get a HTTPS “Git Credentials for ASW CodeCommit” from the security tab of the IAM user.

[IAM](#) > [Users](#) > [admin_user_2023](#)

admin_user_2023 [Info](#) [Delete](#)

Summary

ARN arn:aws:iam::866756323262:user/admin_user_2023 Created April 04, 2023, 23:09 (UTC-02:30)	Console access Enabled without MFA Last console sign-in Today	Access key 1 Create access key
-----------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------	---------------------------------------------------

[Permissions](#) | [Groups \(1\)](#) | [Tags](#) | [Security credentials](#) | [Access Advisor](#)

Identity and Access Management (IAM)

Dashboard

▼ Access management

User groups:

[Users](#)

Roles

Policies

Identity providers

Account settings

▼ Access reports

Access analyzer

Archive rules

Analyst

Settings

Credential report

Organization activity

Service control policies (SCPs)

Related consoles

SSH Key ID Uploaded Status

No SSH public keys

[Upload SSH public key](#)

HTTPS Git credentials for AWS CodeCommit (0)

Generate a user name and password you can use to authenticate HTTPS connections to AWS CodeCommit repositories. You can have a maximum of 2 sets of credentials (active or inactive) at a time. [Learn more](#)

[Actions](#) [Generate credentials](#)

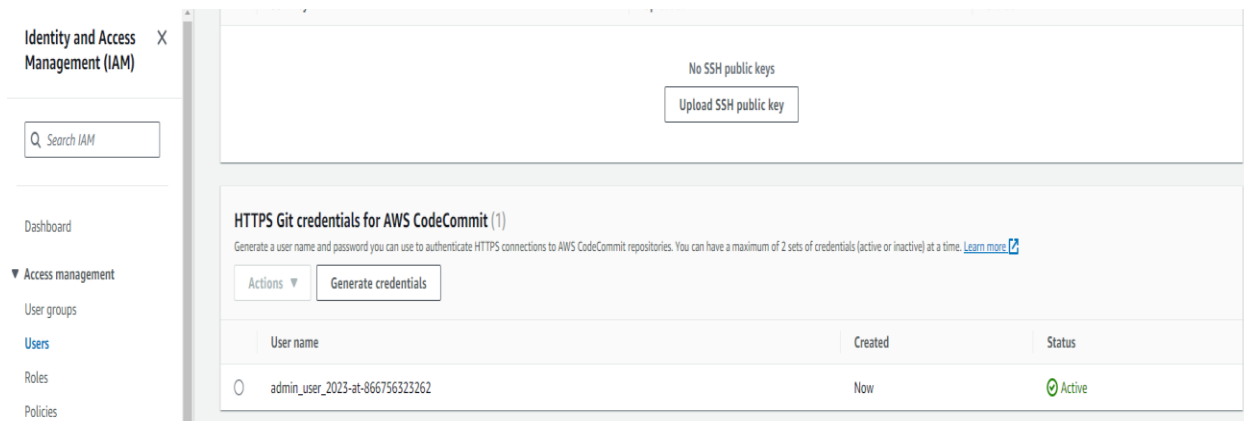
User name	Created	Status
No credentials		
Generate credentials		

Credentials for Amazon Keyspaces (for Apache Cassandra) (0)

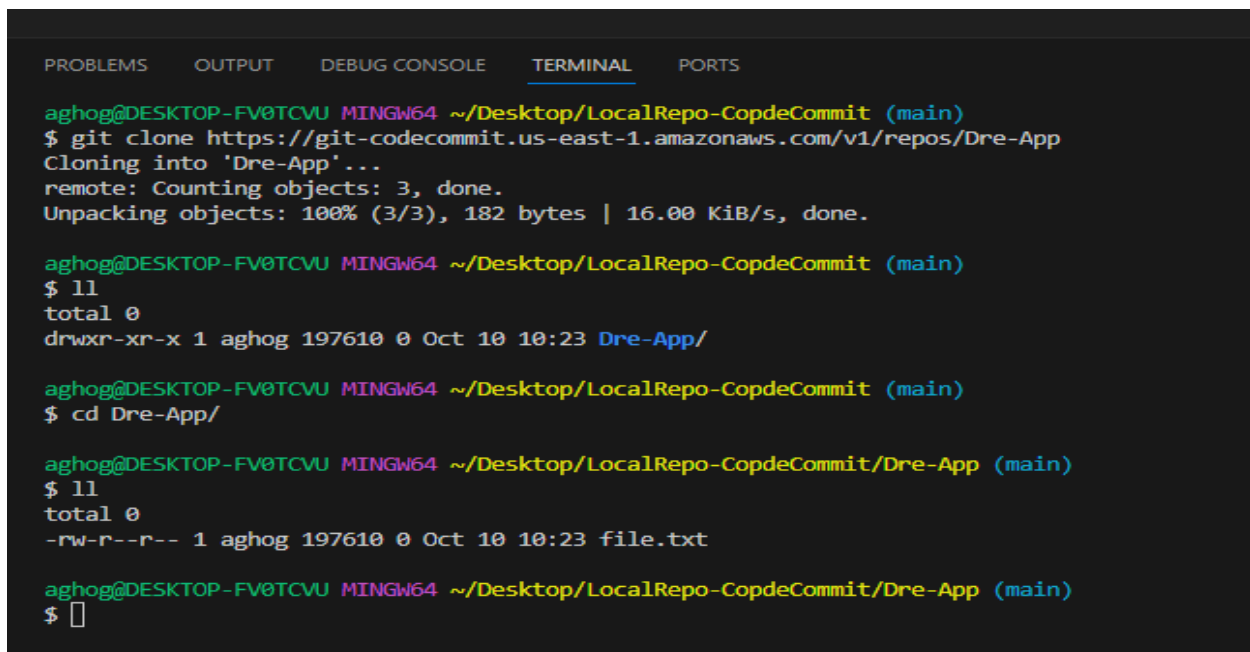
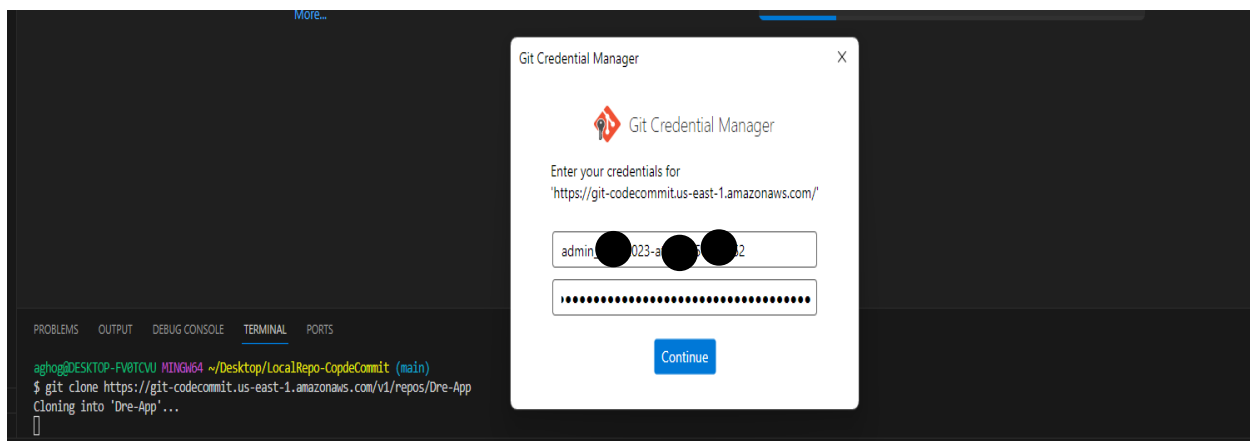
Generate a user name and password you can use to authenticate to Amazon Keyspaces. You can have a maximum of two sets of credentials (active or inactive) at a time. [Learn more](#)

[Actions](#) [Generate credentials](#)

User name	Created	Status
No credentials		
Generate credentials		



Next, you go ahead and create a codecommit repository and clone the http URL of the repository to a folder in your local computer, using your terminal.



Thirdly, you build your project using **CodeBuild**.

[Developer Tools](#) > [CodeBuild](#) > [Build projects](#) > Create build project

Create build project

Project configuration

Project name

A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and _.

Description - *optional*

Build badge - *optional*

☐ Enable build badge

Enable concurrent build limit - *optional*

Limit the number of allowed concurrent builds for this project.

☐ Restrict number of concurrent builds this project can start

► **Additional configuration**

tags

Source

Add source

Source 1 - Primary

Source provider

Repository

Reference type

Choose the source version reference type that contains your source code.

☒ Branch

☐ Git tag

☐ Commit ID

Branch

Choose a branch that contains the code to build.

Commit ID - *optional*

Choose a commit ID. This can shorten the duration of your build.

Source version [Info](#)

refs/heads/main

37e6fc7e [intro](#)

► **Additional configuration**

Git clone depth, Git submodules

Environment

Environment image

☒ **Managed image**
Use an image managed by AWS CodeBuild

☐ **Custom image**
Specify a Docker image

Operating system

Amazon Linux ▼

Runtime(s)

Standard ▼

Image

aws/codebuild/amazonlinux2-x86_64-standard:5.0 ▼

Image version

Always use the latest image for this runtime version ▼

Environment type

Linux EC2 ▼

Privileged

☐ Enable this flag if you want to build Docker images or want your builds to get elevated privileges

Service role

☒ **New service role**
Create a service role in your account

☐ **Existing service role**
Choose an existing service role from your account

Artifacts

Add artifact

Artifact 1 - Primary

Type

No artifacts ▼

You might choose no artifacts if you are running tests or pushing a Docker image to Amazon ECR.

► Additional configuration

Cache, encryption key

Logs

CloudWatch

☒ **CloudWatch logs - optional**
Checking this option will upload build output logs to CloudWatch.

Group name

Stream name

S3

☐ **S3 logs - optional**
Checking this option will upload build output logs to S3.

Cancel

Create build project

Creating a build project makes your codecommit as a source and creates a docker testing environment (a temporary ec2). After creating your build project, you can push your buildspec.yml file from your git to the codecommit repository.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

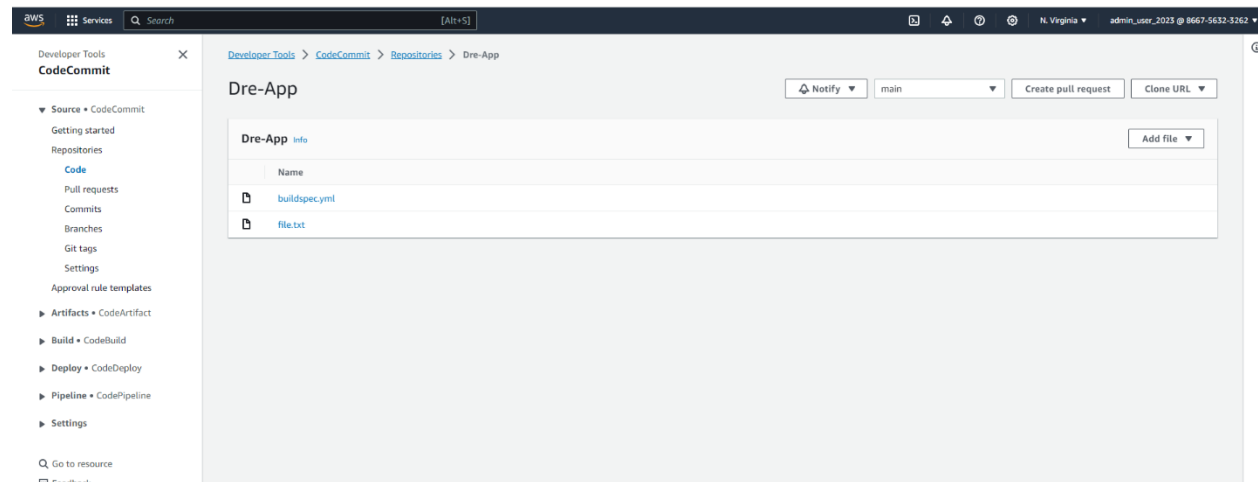
no changes added to commit (use "git add" and/or "git commit -a")

aghog@DESKTOP-FV8TCVU MINGW64 ~/Desktop/LocalRepo-CopdeCommit/Dre-App (main)
$ git add .

aghog@DESKTOP-FV8TCVU MINGW64 ~/Desktop/LocalRepo-CopdeCommit/Dre-App (main)
$ git commit -m "adding the buildspec to the codebuild"
[main 946b577] adding the buildspec to the codebuild
1 file changed, 0 insertions(+), 0 deletions(-)
rename DreAppBuildspec.yml => buildspec.yml (100%)

aghog@DESKTOP-FV8TCVU MINGW64 ~/Desktop/LocalRepo-CopdeCommit/Dre-App (main)
$ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 292 bytes | 292.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Validating objects: 100%
To https://git-codecommit.us-east-1.amazonaws.com/v1/repos/Dre-App
77b5528..946b577  main -> main

aghog@DESKTOP-FV8TCVU MINGW64 ~/Desktop/LocalRepo-CopdeCommit/Dre-App (main)
$ []
```



Fourth Step, you build a CI/CD pipeline as an orchestrator for automation of any changes made to your code “buildspec.yml”

aws

Services

Search

[Alt+S]

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Choose pipeline settings Info

Pipeline settings

Pipeline name

Enter the pipeline name. You cannot edit the pipeline name after it is created.

Dre-App-Pipeline

No more than 100 characters

Service role

☒ New service role

Create a service role in your account

☐ Existing service role

Choose an existing service role from your account

Role name

AWSCodePipelineServiceRole-us-east-1-Dre-App-Pipeline

Type your service role name

☒ Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

► Advanced settings

Cancel

Next

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Add source stage Info

Source

Source provider

This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

AWS CodeCommit

Repository name

Choose a repository that you have already created where you have pushed your source code.

Q Dre-App X

Branch name

Choose a branch of the repository

Q main X

Change detection options

Choose a detection mode to automatically start your pipeline when a change occurs in the source code.

☒ Amazon CloudWatch Events (recommended)

Use Amazon CloudWatch Events to automatically start my pipeline when a change occurs

☐ AWS CodePipeline

Use AWS CodePipeline to check periodically for changes

Output artifact format

Choose the output artifact format.

☒ CodePipeline default

AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.

☐ Full clone

AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full Git clone. Only supported for AWS CodeBuild actions.

Cancel

Previous

Next

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Add build stage [Info](#)

Build - optional

Build provider

This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.

AWS CodeBuild ▼

Region

US East (N. Virginia) ▼

Project name

Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.

Q Dre-App X or Create project ↗

Environment variables - optional

Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. [Learn more](#) ↗

Add environment variable

Build type

☒ Single build

Triggers a single build.

☐ Batch build

Triggers multiple builds as a single execution.

Cancel

Previous

Skip build stage

Next

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Add deploy stage [Info](#)

Deploy - optional

Deploy provider

Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.

▼

Cancel

Previous

Skip deploy stage

Next

RepositoryName
Dre-App
BranchName
main
PollForSourceChanges
false
OutputArtifactFormat
CODE_ZIP

Step 3: Add build stage

Build action provider

Build action provider
AWS CodeBuild
ProjectName
Dre-App

Step 4: Add deploy stage

Deploy action provider

Deployment stage
No deploy

Cancel

Previous

Create pipeline

Developer Tools

CodePipeline

► Source • CodeCommit

► Artifacts • CodeArtifact

► Build • CodeBuild

► Deploy • CodeDeploy

▼ Pipeline • CodePipeline

Getting started

Pipelines

Pipeline

History

Settings

► Settings

Go to resource

Feedback

Developer Tools > CodePipeline > Pipelines > DreApp-Pipeline

DreApp-Pipeline

Notify

Edit

Stop execution

Clone pipeline

Release change

Source

Succeeded

Pipeline execution ID: 1bd1156c-2b5a-4d66-b2ef-0aeed518ed64

Source

AWS CodeCommit

Succeeded - 3 minutes ago

946b677e

946b677e: Source: adding the buildspec to the codebuild

Disable transition

Build

Succeeded

Pipeline execution ID: 1bd1156c-2b5a-4d66-b2ef-0aeed518ed64

Build

AWS CodeBuild

Succeeded - 2 minutes ago

Details

View logs

946b677e: Source: adding the buildspec to the codebuild

✓

✓

Finally, a two stage CI/CD pipeline is created.

Screenshot of the Build Log.

Logs

⤴ Show previous logs

1 [Container] 2023/10/10 14:42:40 Waiting for agent ping

2 [Container] 2023/10/10 14:42:41 Waiting for DOWNLOAD_SOURCE

3 [Container] 2023/10/10 14:42:43 Phase is DOWNLOAD_SOURCE

4 [Container] 2023/10/10 14:42:43 CODEBUILD_SRC_DIR=/codebuild/output/src135598616/src

5 [Container] 2023/10/10 14:42:43 YAHML location is /codebuild/output/src135598616/src/buildspec.yml

6 [Container] 2023/10/10 14:42:43 No commands found for phase name: install

7 [Container] 2023/10/10 14:42:43 Setting HTTP client timeout to higher timeout for S3 source

8 [Container] 2023/10/10 14:42:43 Processing environment variables

9 [Container] 2023/10/10 14:42:43 Selecting 'python' runtime version '3.9' based on manual selections...

10 [Container] 2023/10/10 14:42:43 Running command echo "Installing Python version 3.9 ..."

11 Installing Python version 3.9 ...

12

13 [Container] 2023/10/10 14:42:43 Running command pyenv global \$PYTHON_39_VERSION

14

15 [Container] 2023/10/10 14:42:44 Moving to directory /codebuild/output/src135598616/src

16 [Container] 2023/10/10 14:42:44 Unable to initialize cache download: no paths specified to be cached

17 [Container] 2023/10/10 14:42:44 Configuring ssm agent with target id: codebuild:4c3dfc80-faad-4214-90a7-fc018fa7a0ba

18 [Container] 2023/10/10 14:42:44 Successfully updated ssm agent configuration

19 [Container] 2023/10/10 14:42:44 Registering with agent

20 [Container] 2023/10/10 14:42:44 Phases found in YAHML: 4

21 [Container] 2023/10/10 14:42:44 POST_BUILD: 2 commands

22 [Container] 2023/10/10 14:42:44 INSTALL: 0 commands

23 [Container] 2023/10/10 14:42:44 PRE_BUILD: 2 commands

24 [Container] 2023/10/10 14:42:44 BUILD: 3 commands

25 [Container] 2023/10/10 14:42:44 Phase complete: DOWNLOAD_SOURCE State: SUCCEEDED

26 [Container] 2023/10/10 14:42:44 Phase context status code: Message:

27 [Container] 2023/10/10 14:42:45 Entering phase INSTALL

28 [Container] 2023/10/10 14:42:45 Phase complete: INSTALL State: SUCCEEDED

29 [Container] 2023/10/10 14:42:45 Phase context status code: Message:

30 [Container] 2023/10/10 14:42:45 Entering phase PRE_BUILD

31 [Container] 2023/10/10 14:42:45 Running command echo "CHECKING IF ALL SOFTWARE HAS BEEN INSTALLED"

32 CHECKING IF ALL SOFTWARE HAS BEEN INSTALLED

33

34 [Container] 2023/10/10 14:42:45 Running command python3 --version

35 Python 3.9.17

Logs

36

37 [Container] 2023/10/10 14:42:45 Phase complete: PRE_BUILD State: SUCCEEDED

38 [Container] 2023/10/10 14:42:45 Phase context status code: Message:

39 [Container] 2023/10/10 14:42:45 Entering phase BUILD

40 [Container] 2023/10/10 14:42:45 Running command echo "BUILD STARTED AT 'date' "

41 BUILD STARTED AT Tue Oct 10 14:42:45 UTC 2023

42

43 [Container] 2023/10/10 14:42:45 Running command pwd

44 /codebuild/output/src135598616/src

45

46 [Container] 2023/10/10 14:42:45 Running command echo "Dre-App is being Built"

47 Dre-App is being Built

48

49 [Container] 2023/10/10 14:42:45 Phase complete: BUILD State: SUCCEEDED

50 [Container] 2023/10/10 14:42:45 Phase context status code: Message:

51 [Container] 2023/10/10 14:42:45 Entering phase POST_BUILD

52 [Container] 2023/10/10 14:42:45 Running command echo "BUIL Dre-App COMPLETED at 'date' "

53 BUIL Dre-App COMPLETED at Tue Oct 10 14:42:45 UTC 2023

54

55 [Container] 2023/10/10 14:42:45 Running command ls -l

56 total 4

57 -rw-r--r-- 1 root root 811 Oct 10 14:42 buildspec.yml

58 -rw-r--r-- 1 root root 0 Oct 10 14:42 file.txt

59

60 [Container] 2023/10/10 14:42:46 Phase complete: POST_BUILD State: SUCCEEDED

61 [Container] 2023/10/10 14:42:46 Phase context status code: Message:

62 [Container] 2023/10/10 14:42:46 Expanding base directory path: .

63 [Container] 2023/10/10 14:42:46 Assembling file list

64 [Container] 2023/10/10 14:42:46 Expanding .

65 [Container] 2023/10/10 14:42:46 Expanding file paths for base directory .

66 [Container] 2023/10/10 14:42:46 Assembling file list

67 [Container] 2023/10/10 14:42:46 Expanding */*

68 [Container] 2023/10/10 14:42:46 Found 2 file(s)

69 [Container] 2023/10/10 14:42:46 Phase complete: UPLOAD_ARTIFACTS State: SUCCEEDED

70 [Container] 2023/10/10 14:42:46 Phase context status code: Message:

71

Close