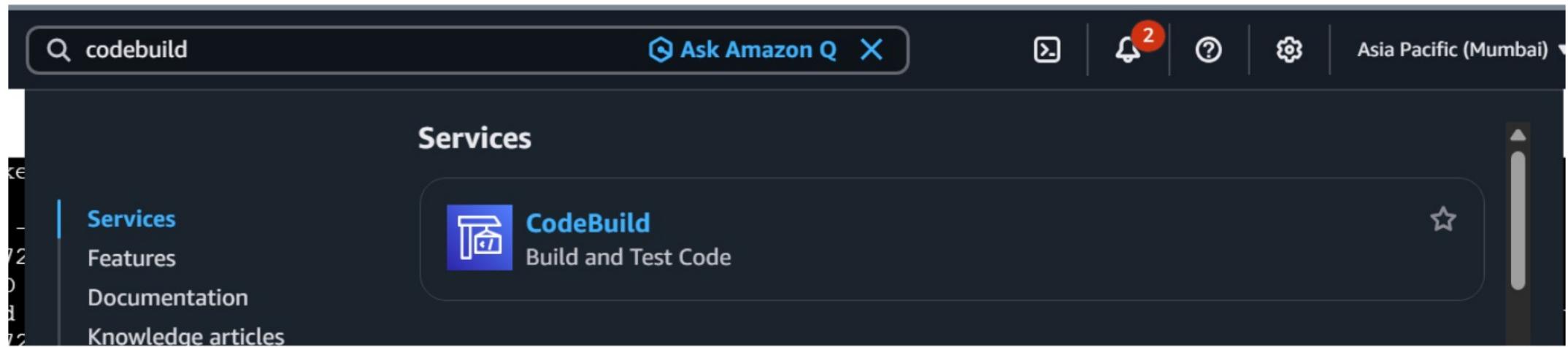


# **4. Creation of Codebuild Project**

# 1. Go to codebuild option in aws



**2. Click on create project**



**Create project**

# 3. Name the project

Create build project

## Project configuration

### Project name

brain-tasks-build

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 \_.

# 4. Select github as a source provider

## Source 1 - Primary

Source provider

GitHub



Credential

## 5. Select repository in my github account option

**Repository**



Repository in my GitHub account

# 6. Create a repository in github

## Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

---

*Required fields are marked with an asterisk (\*).*

Owner \*

 deepeshchandnani ▼

Repository name \*

/ Brain-Tasks-App

✓ Brain-Tasks-App is available.

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.



## 7. Copy the repo link

Local      Codespaces      No

Clone      ?

HTTPS    SSH    GitHub CLI

<https://github.com/deepeshchandnani/Brain-Tasks> 

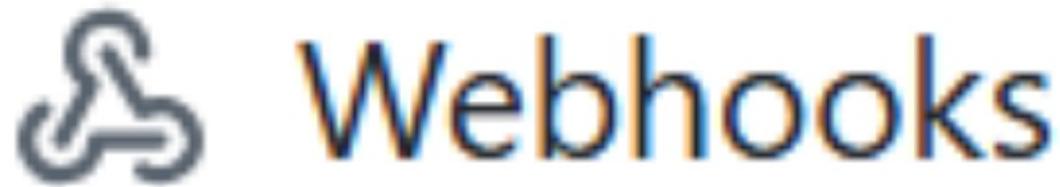

# 8. Paste it on codebuild repo path section

## GitHub repository

<https://github.com/deepeshchandnani/Brain-Tasks-App.git>

<https://github.com/<user-name>/<repository-name>>

**9. Go to webhooks section in github repo**



# 10. Add webhook

## Webhooks

---

Add webhook

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

# 11. Choose single build option

## ▼ Primary source webhook events Info

Webhook - *optional* Info 

- Rebuild every time a code change is pushed to this repository

### Build type

Single build

Triggers single build

Batch build

Triggers multiple builds as single execution

### Comment approval

ALL\_PULL\_REQUESTS



### Approver roles



GITHUB\_WRITE

GITHUB\_MAINTAIN

GITHUB\_ADMIN

## 12. Enable report build status option

### **Build Status - *optional***

- Report build statuses to source provider when your builds start and finish

# 13. Click on create build project

Cancel

Create build project

# 14. Create github app connection

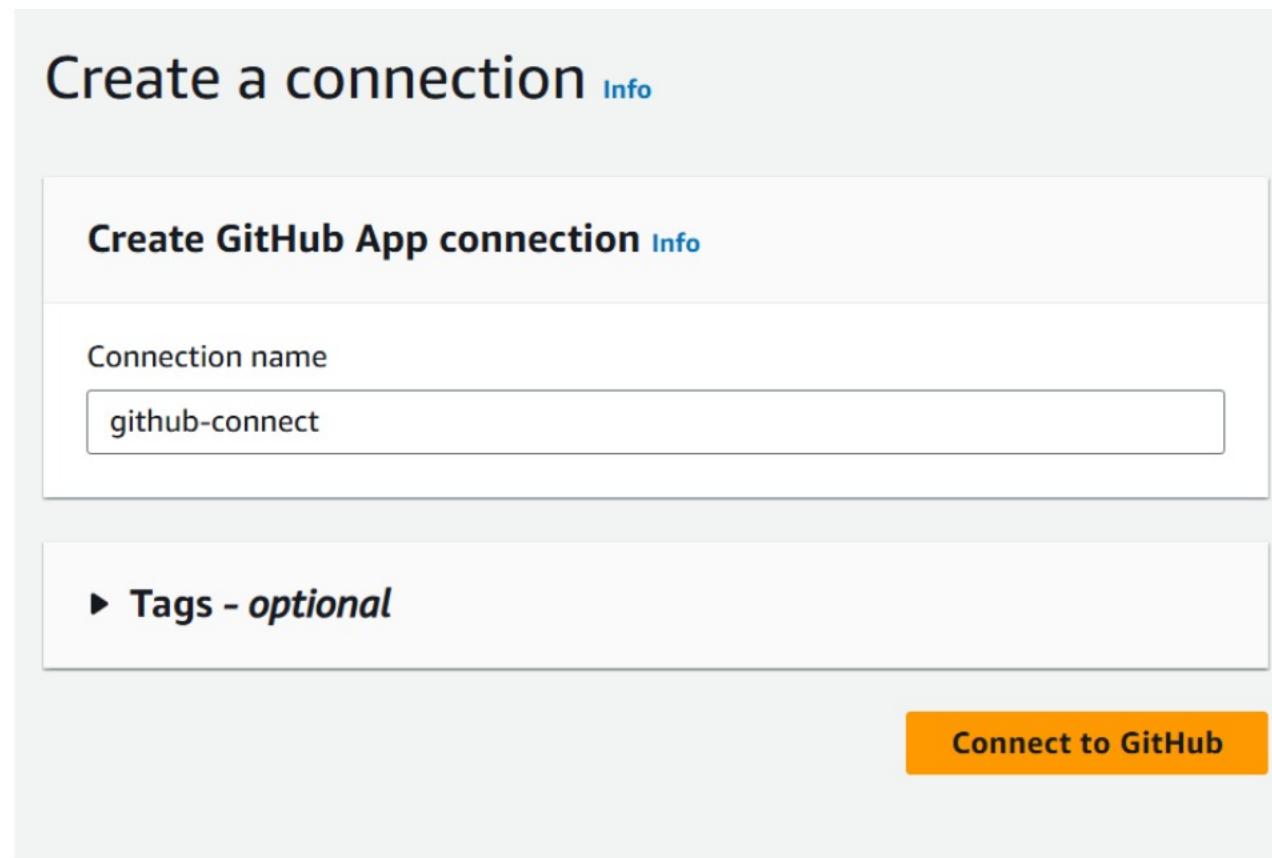
Create a connection [Info](#)

Create GitHub App connection [Info](#)

Connection name

► Tags - *optional*

**Connect to GitHub**



# 15. Choose github app as a source provider

**Manage default source credential**

**Source Provider**

GitHub

**Credential type**

GitHub App  
Connect project to GitHub using an AWS managed GitHub App

Personal access token  
Connect project to GitHub using a personal access token

OAuth app  
Connect project to GitHub using an OAuth app

**Connection**

You can [create a new GitHub connection](#) by using an AWS managed GitHub App

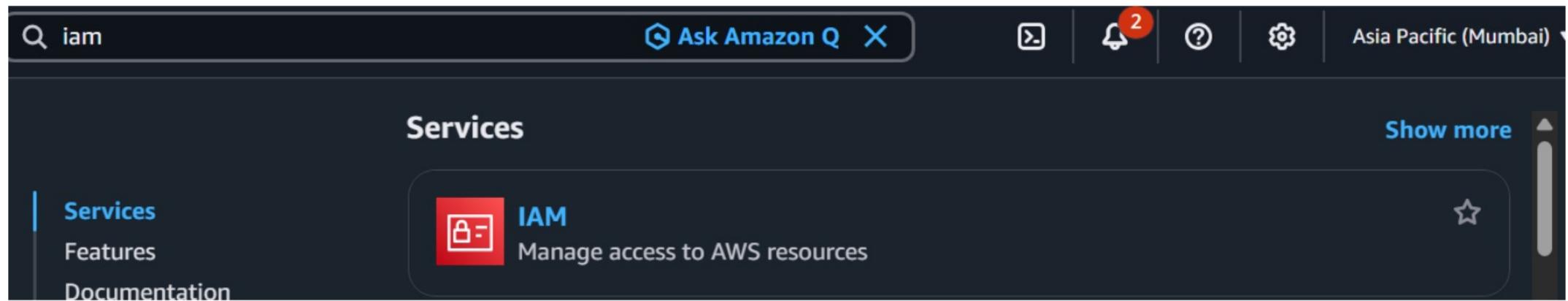
arn:aws:codeconnections:ap-south-1:284419413658:connection/d7759011-e2bb-4f43-9e9

# 16. Enable privilege option to build docker images

## Privileged

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

# 17. Go to iam page in aws



## 18. Give the required permission to the role

- AmazonEC2ContainerRegistryPowerUser
- CloudWatchLogsFullAccess

# 19. Generate new token in github



**No fine-grained tokens created**

1 API token for scripts or testing? Generate a personal access token for quick access GitHub API.

[Generate new token](#)

# 20. Name the token

---

## New fine-grained personal access token

---

Create a fine-grained, repository-scoped token suitable for personal API use and for using Git over HTTPS.

Token name \*

 **'deepesh' is available.**

A unique name for this token. May be visible to resource owners or users with possession of the token.

## 21. Click on generate token



This token will be ready for use immediately.

# 22. Copy personal access token

Fine-grained personal access tokens

[Generate new token](#)

These are fine-grained, repository-scoped tokens suitable for personal [API](#) use and for using Git over HTTPS.



deepesh

Never used • Expires on Fri, Jan 16 2026

[Delete](#)



Make sure to copy your personal access token now as you will not be able to see this again.

github\_pat\_11AUNT5BQ0UIMKYbujW6JS\_q3KgQgk4cztzViGcbqTx6j4hxIAb7If0r9YacZVjzv20EH25WGAnrdKv



# 23. Connect from local repo to git repo

```
ubuntu@ip-172-31-3-167:~/Brain-Tasks-App$ git remote add origin https://deepeshchandnani:github_pat_11AUNT5BQ0UIMKYbujW6JS_q3KgQgk4cZtzViGCb  
qTx6j4hxIAb7If0r9YacZVjzv2OEH25WGAnrdKvNj1@github.com/deepeshchandnani/Brain-Tasks-App.git  
ubuntu@ip-172-31-3-167:~/Brain-Tasks-App$ []
```

# 24. Create buildspec file

```
ubuntu@ip-172-31-3-167:~/Brain-Tasks-App$ nano buildspec.yml
```

# 25. Write build code in buildspec file

```
GNU nano 7.2                                     buildspec.yml *
```

```
version: 0.2

env:
  variables:
    AWS_DEFAULT_REGION: ap-south-1
    ECR_REPO: brain-tasks-nginx
    IMAGE_TAG: v1

phases:
  pre_build:
    commands:
      - echo "Logging in to Amazon ECR..."
      - aws ecr get-login-password --region $AWS_DEFAULT_REGION | docker login --username AWS --password-stdin \
        $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com

  build:
```

## 26. Add all files in stage area

```
ubuntu@ip-172-31-3-167:~/Brain-Tasks-App$ git add .
```

# 27. Commit the files

```
ubuntu@ip-172-31-3-167:~/Brain-Tasks-App$ git commit -m "Add Dockerfile, nginx config and buildspec for CodeBuild"
[main 7d9739b] Add Dockerfile, nginx config and buildspec for CodeBuild
Committer: Ubuntu <ubuntu@ip-172-31-3-167.ap-south-1.compute.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

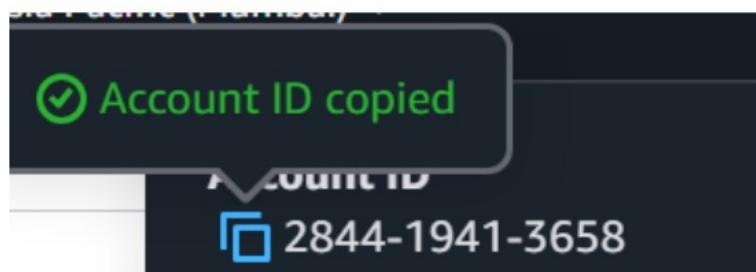
git commit --amend --reset-author

3 files changed, 59 insertions(+)
```

# 28. push the files from local to remote

```
ubuntu@ip-172-31-3-167:~/Brain-Tasks-App$ git push origin main
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 2 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 1.05 KiB | 1.05 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/deepeshchandnani/Brain-Tasks-App.git
  db063b2..9470408  main -> main
ubuntu@ip-172-31-3-167:~/Brain-Tasks-App$ █
```

# 29. Now add Environment variables in build section



## Environment variables

Name	Value	Type	
AWS_ACCOUNT_ID	284419413658	Plaintext	<button>Remove</button>
AWS_DEFAULT_REGION	ap-south-1	Plaintext	<button>Remove</button>
ECR_REPO	brain-tasks-nginx	Plaintext	<button>Remove</button>
IMAGE_TAG	v1	Plaintext	<button>Remove</button>
<a href="#">Add environment variable</a>			

# 30. Enable cloudwatch logs option for build output logs

## ▼ Logs

### CloudWatch



CloudWatch logs - *optional*

Checking this option will upload build output logs to CloudWatch.

# 31. Choose buildspec file option

## ▼ Buildspec ⚠

### Build specifications

Insert build commands

Store build commands as build project configuration

Use a buildspec file

Store build commands in a YAML-formatted buildspec file

### Buildspec name - optional

By default, CodeBuild looks for a file named buildspec.yml in the source code root directory. If your buildspec file uses a different name or location, enter its path from the source root here (for example, buildspec-two.yml or configuration/buildspec.yml).

buildspec.yml

**32. Now goto webhook section in  
github**



# 32.1. Copy payload and secret to add in the webhook

## Create webhook

X

You must create a webhook for your GitHub repository.

### Payload URL

`https://codebuild.ap-south-1.amazonaws.com/webhooks?t=eyJlbmNyeXB0ZWREYXRhljoiY254K0Uydnk3cIBkSUt`

[Copy payload URL](#)

### Secret

`nJt1q0HV6RdtpLkg-KTmZc8hKu5qqmPiwM5-ZOS2naKEzmZqERtJVrJRwXG0qgtzUu4lMufb4S_yKcyzUCnfBakR-II`

[Copy secret](#)

#### Webhook creation steps

In order to manually create a GitHub webhook, follow the below instructions. For more information about GitHub webhooks, you can reference the [GitHub webhook documentation](#):

1. Navigate to the GitHub settings page for the GitHub resource associated with your CodeBuild project
2. Select **Webhooks** and click **Add webhook**
3. Add the above **Payload URL** value under **Payload URL**
4. Set the **Content type** to `application/json`
5. Add the above **Secret** value under **Secret**
6. Under **Which events would you like to trigger this webhook?**, select **Let me select individual events**

# 33. Add payload url and secret in webhook

[Webhooks](#) / [Add webhook](#)

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in [our developer documentation](#).

**Payload URL \***

`https://codebuild.ap-south-1.amazonaws.com/webhooks?t=eyJlbmNyeXB0ZWREYXRhljoiY254K0Uydnk3cIBkSUdLZV`

**Content type \***

`application/json`

**Secret**

`nJt1q0HV6RdtpLkg-KTmZc8hKu5qqmPiwM5-ZOS2naKEzmZqERtJVrJRwXG0qgtzUu4lMufb4S_yKcyzUCnfBakR-ILfbaiP'`

# 34. Here webhook successfully connected to aws codebuild

[Webhooks](#) / Manage webhook

Settings

Recent Deliveries



3e254432-db12-11f0-9df1-8dd79dabf29a ping

2025-12-17 12:03:06 ...

Request

Response **200**

Redeliver

Completed in 0.84 seconds.

Headers

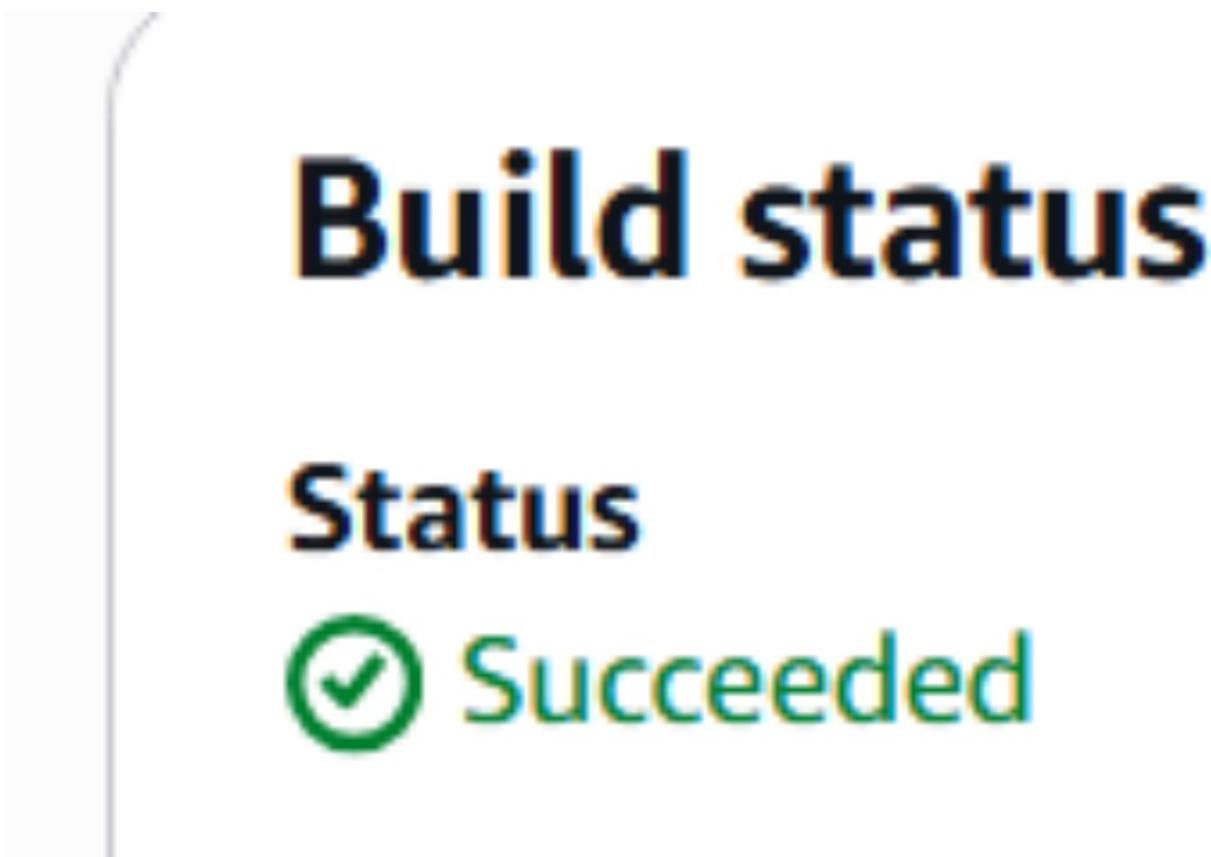
**Content-Length:** 56

**Content-Type:** application/json

**Date:** Wed, 17 Dec 2025 06:33:06 GMT

**X-Amzn-Requestid:** 3dc294a2-627d-40e6-b8a7-2db05fd5092b

## 35. now build succeeded



# 36. In the build logs all states are showing succeeded

```
164
165 [Container] 2025/12/17 16:27:25.148916 Phase complete: POST_BUILD State: SUCCEEDED
166 [Container] 2025/12/17 16:27:25.148936 Phase context status code: Message:
167 [Container] 2025/12/17 16:27:25.202680 Set report auto-discover timeout to 5 seconds
168 [Container] 2025/12/17 16:27:25.202734 Expanding base directory path: .
169 [Container] 2025/12/17 16:27:25.206226 Assembling file list
170 [Container] 2025/12/17 16:27:25.206243 Expanding .
171 [Container] 2025/12/17 16:27:25.209760 Expanding file paths for base directory .
172 [Container] 2025/12/17 16:27:25.209775 Assembling file list
173 [Container] 2025/12/17 16:27:25.209781 Expanding */
174 [Container] 2025/12/17 16:27:25.213683 No matching auto-discover report paths found
175 [Container] 2025/12/17 16:27:25.213707 Report auto-discover file discovery took 0.011026 seconds
176 [Container] 2025/12/17 16:27:25.213776 Phase complete: UPLOAD_ARTIFACTS State: SUCCEEDED
177 [Container] 2025/12/17 16:27:25.213784 Phase context status code: Message:
178
```