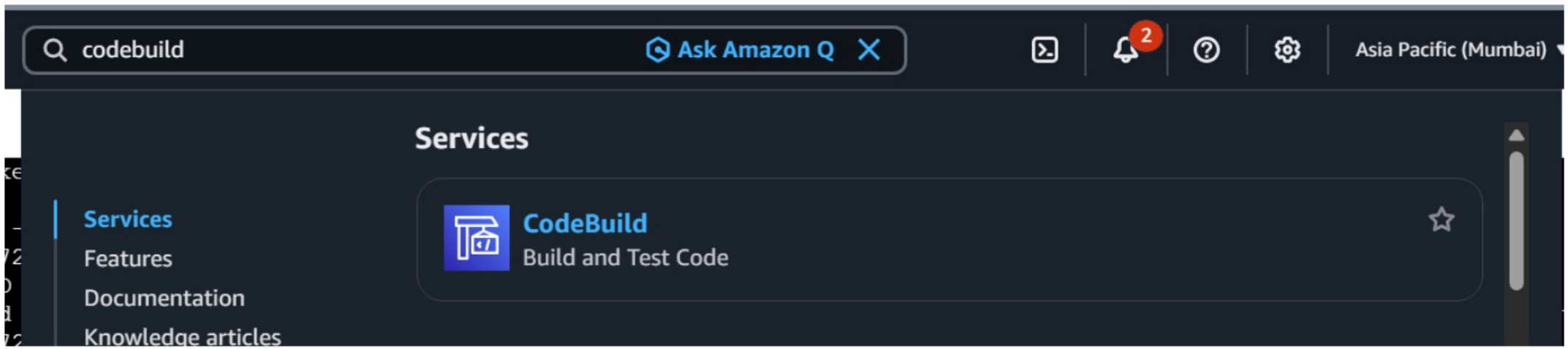
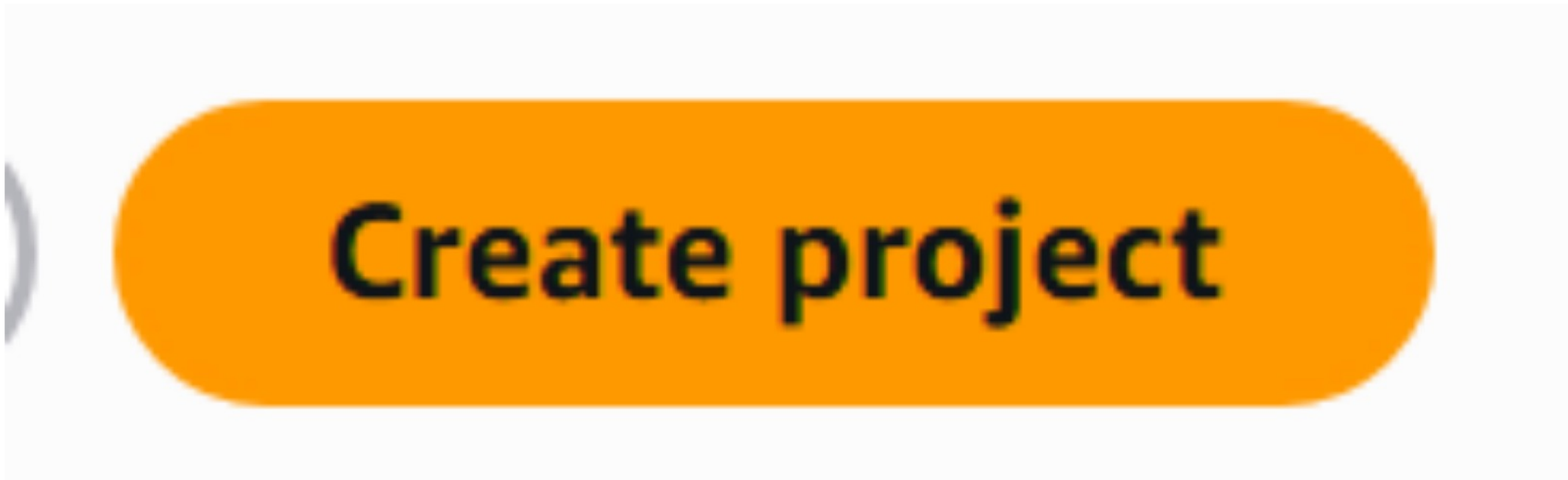
4. Creation of Codebuild Project

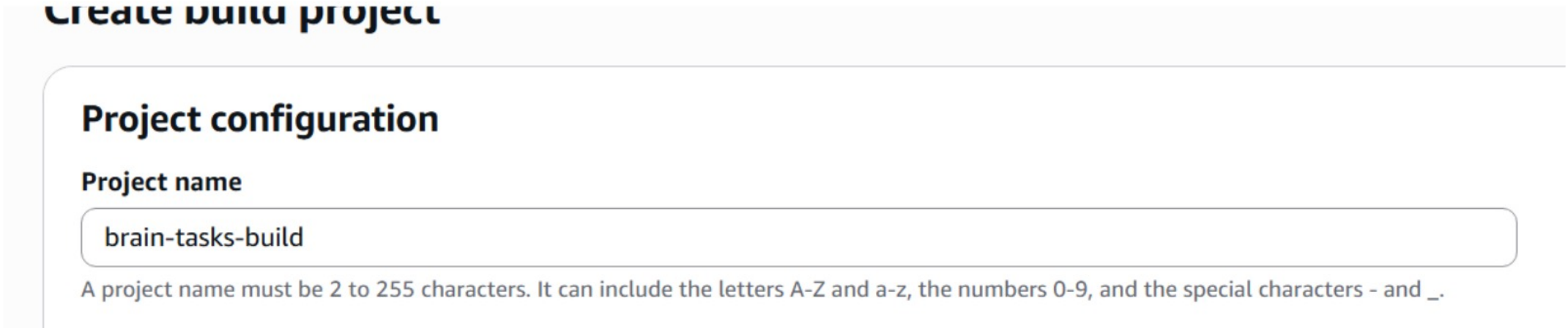
1. Go to codebuild option in aws



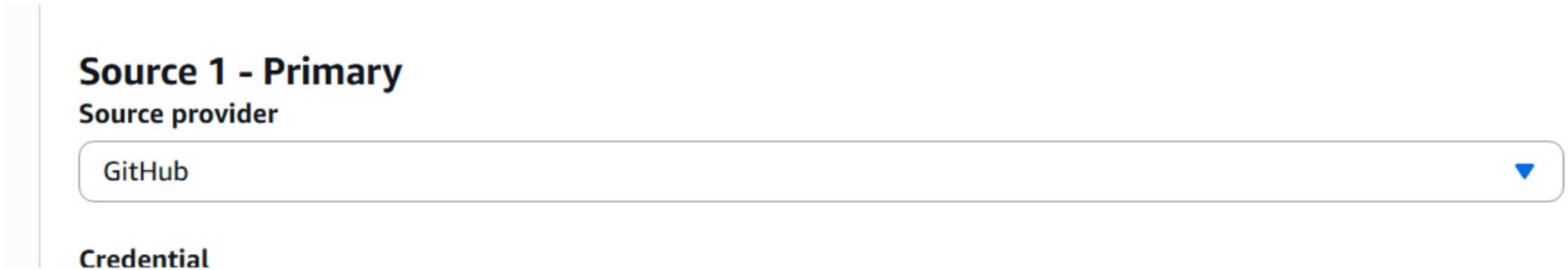
2. Click on create project



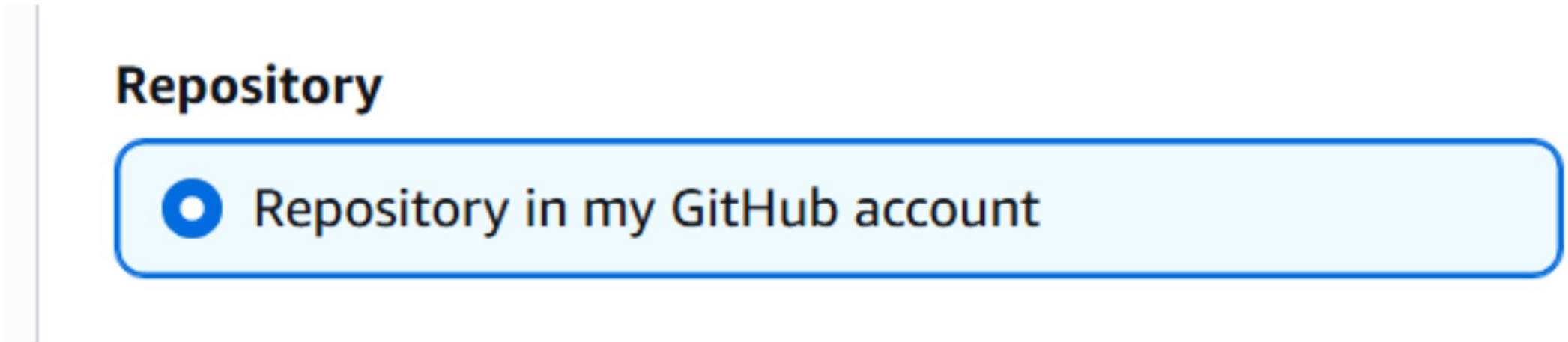
3. Name the project



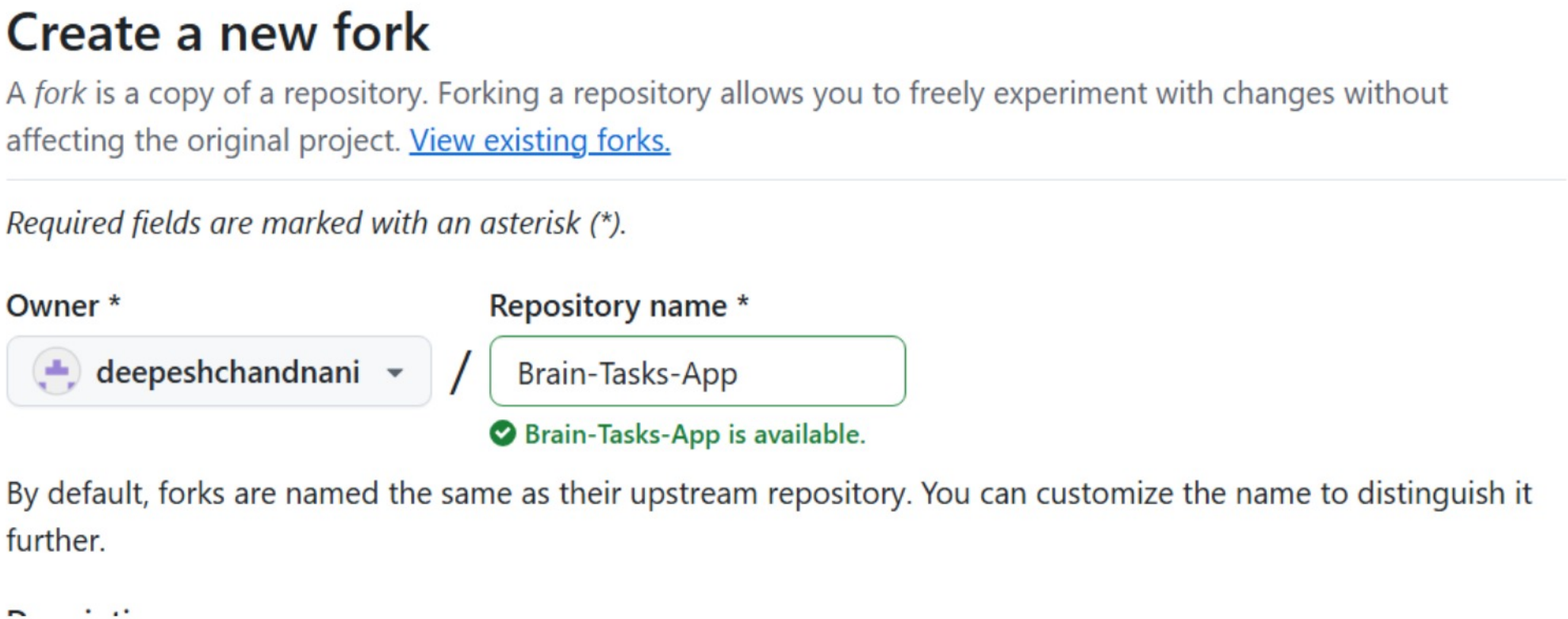
4. Select github as a source provider



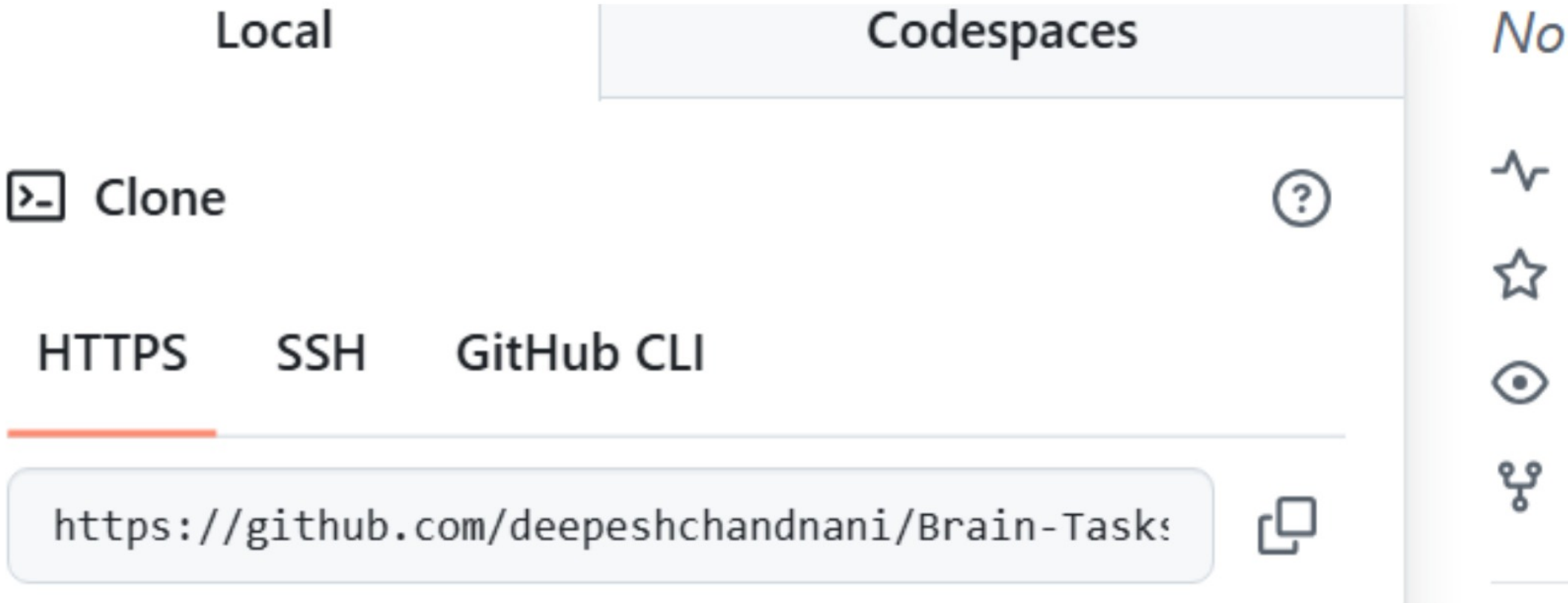
5. Select repository in my github account option



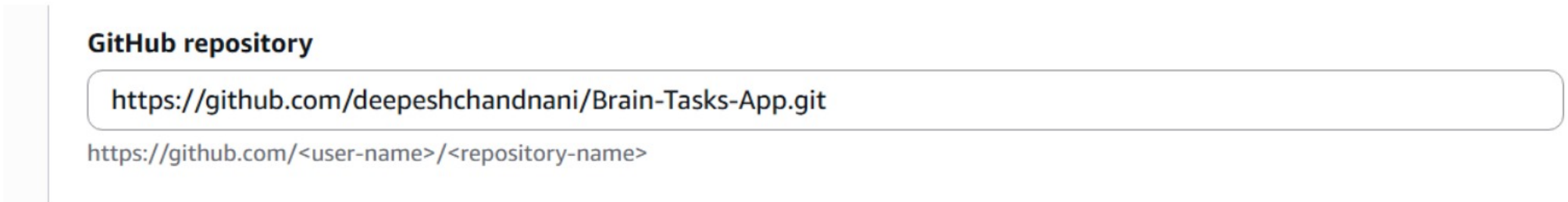
6. Create a repository in github



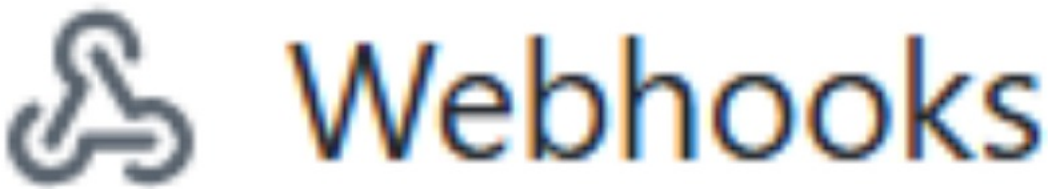
7. Copy the repo link



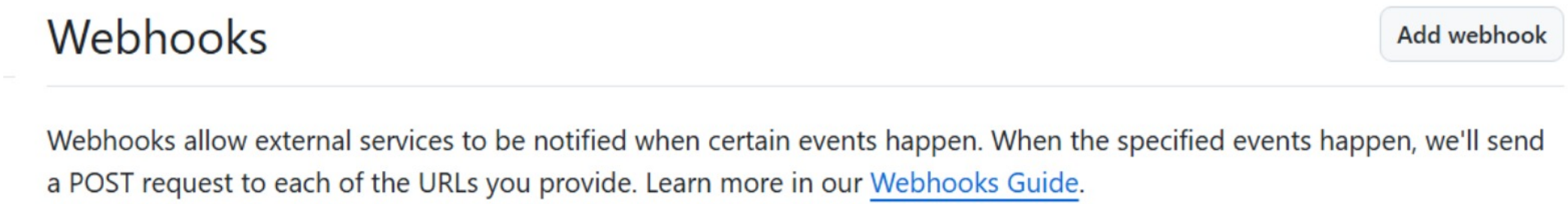
8. Paste it on codebuild repo path section



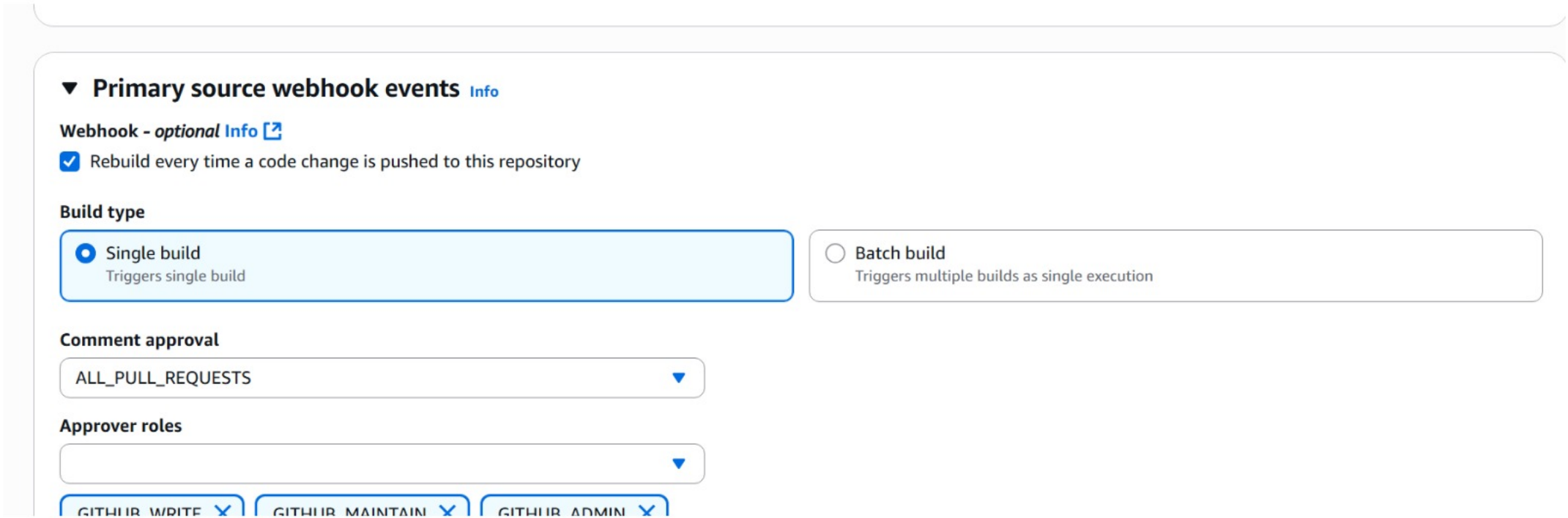
9. Go to webhooks section in github repo



10. Add webhook



11. Choose single build option

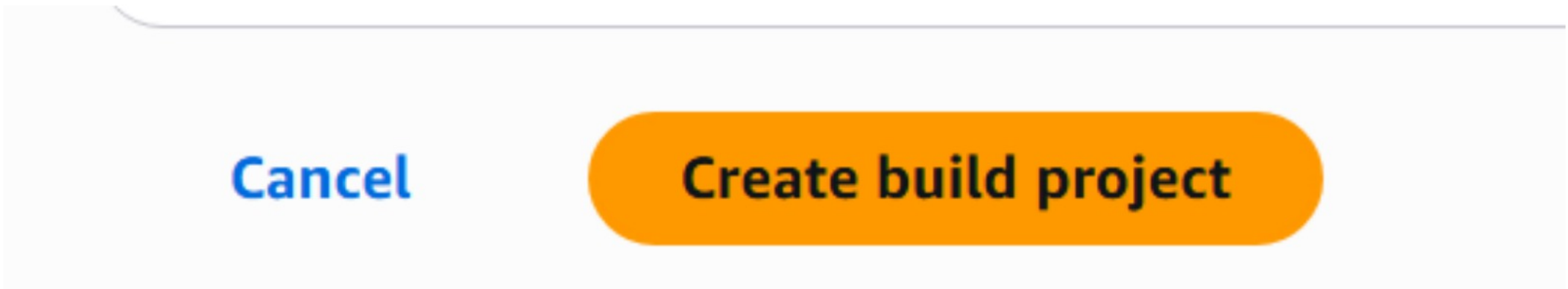


12. Enable report build status option

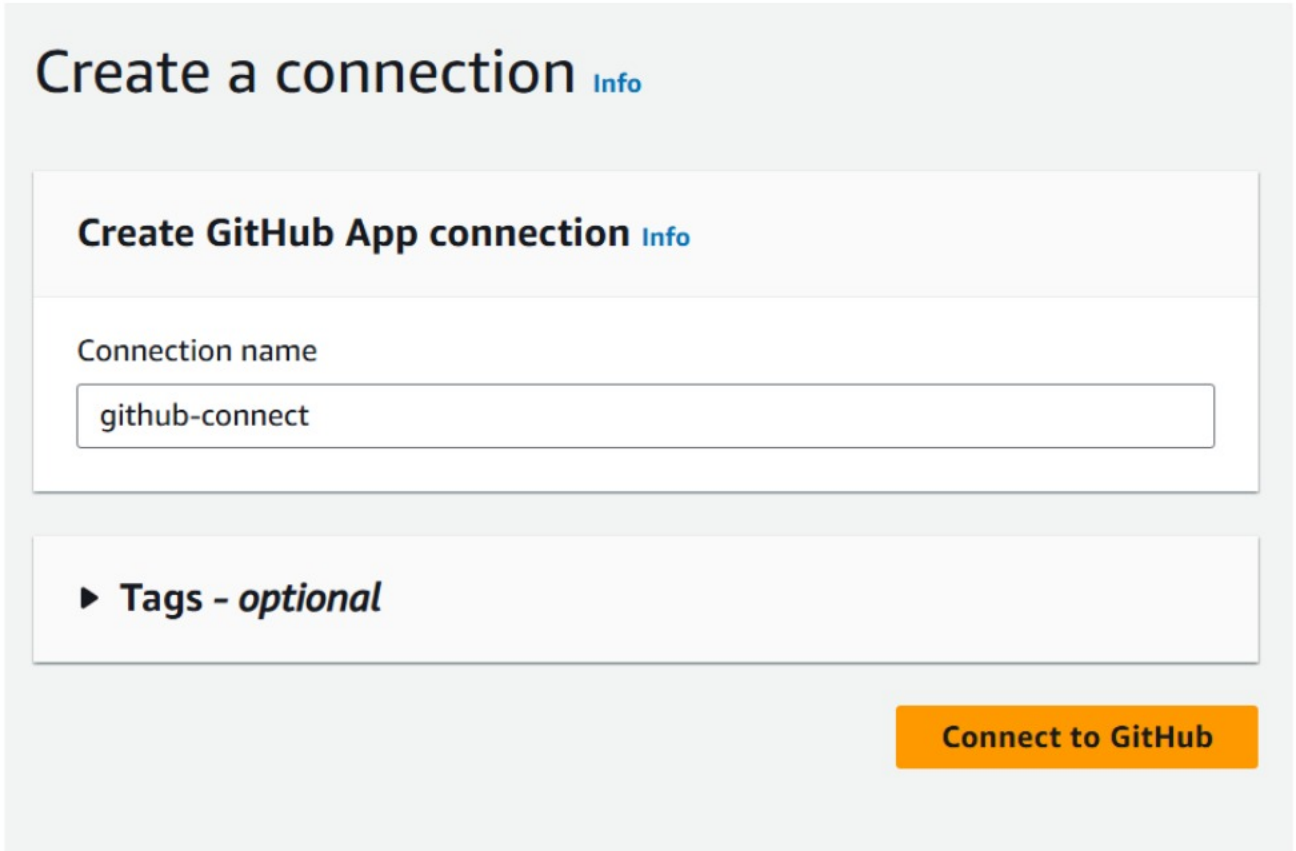
Build Status - *optional*

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

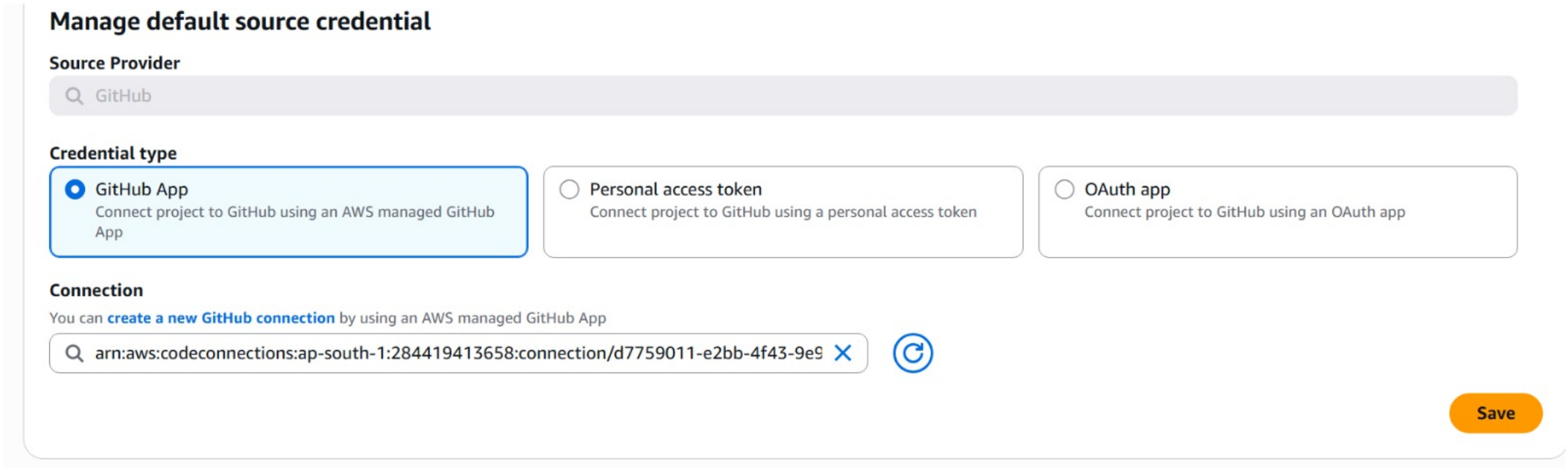
13. Click on create build project



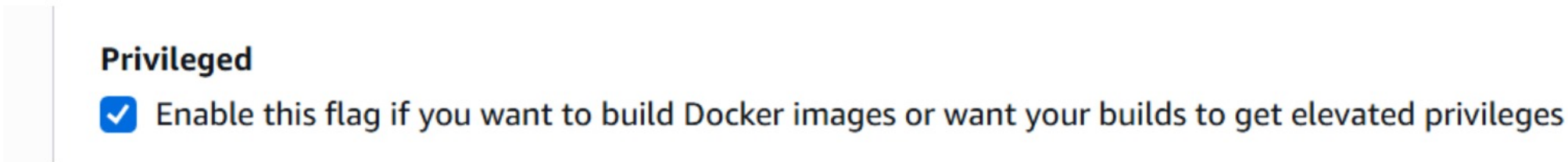
14. Create github app connection



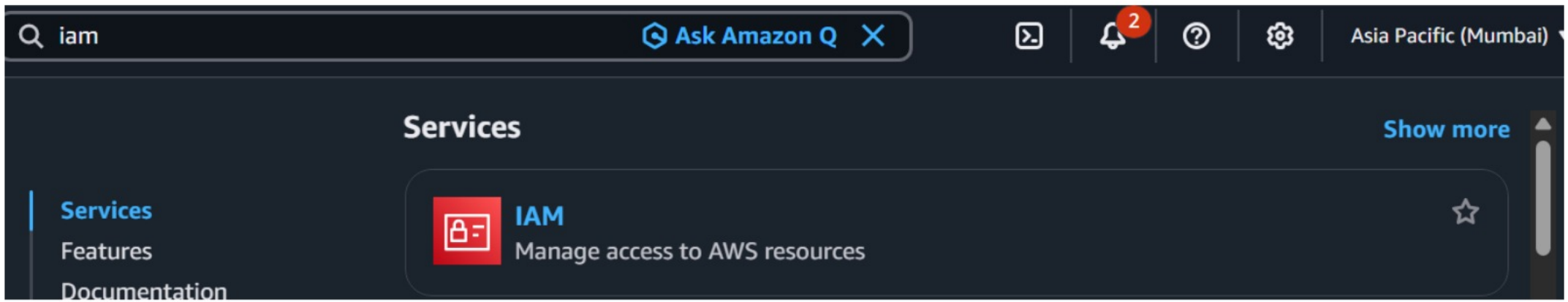
15. Choose github app as a source provider



16. Enable priviledge option to build docker images



17. Go to iam page in aws



18. Give the required permission to the role

- ✔ `AmazonEC2ContainerRegistryPowerUser`
- ✔ `CloudWatchLogsFullAccess`

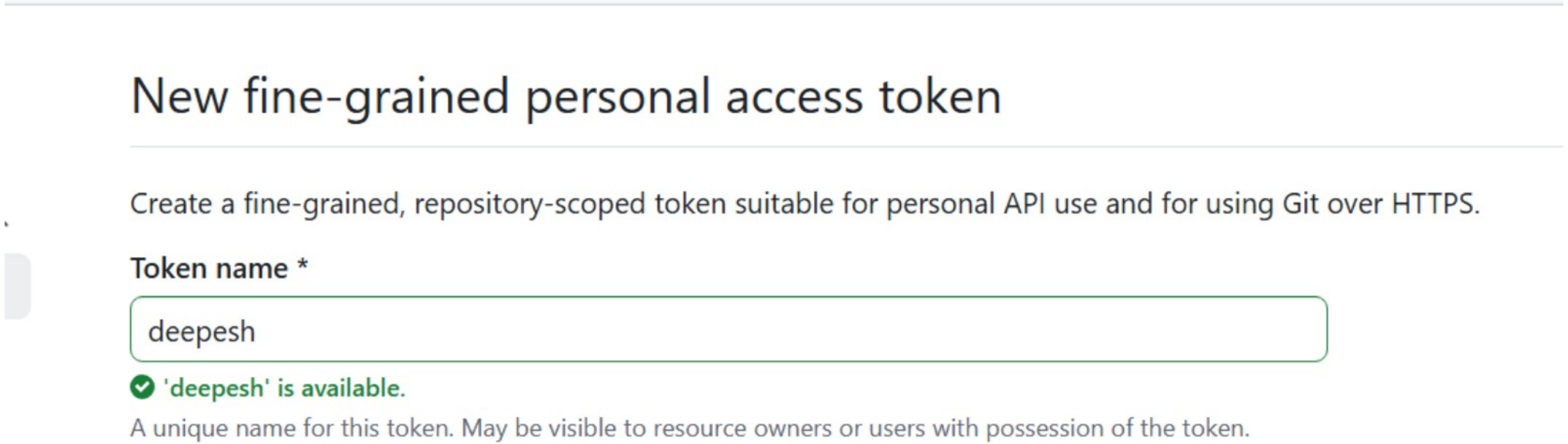
19. Generate new token in github

No fine-grained tokens created

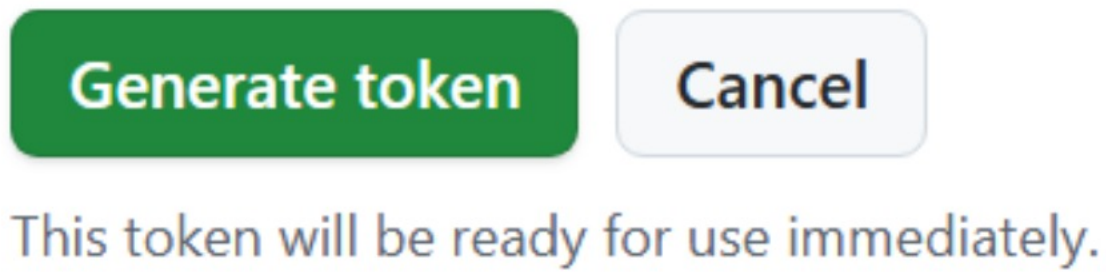
ı API token for scripts or testing? Generate a personal access token for quick access GitHub API.

Generate new token

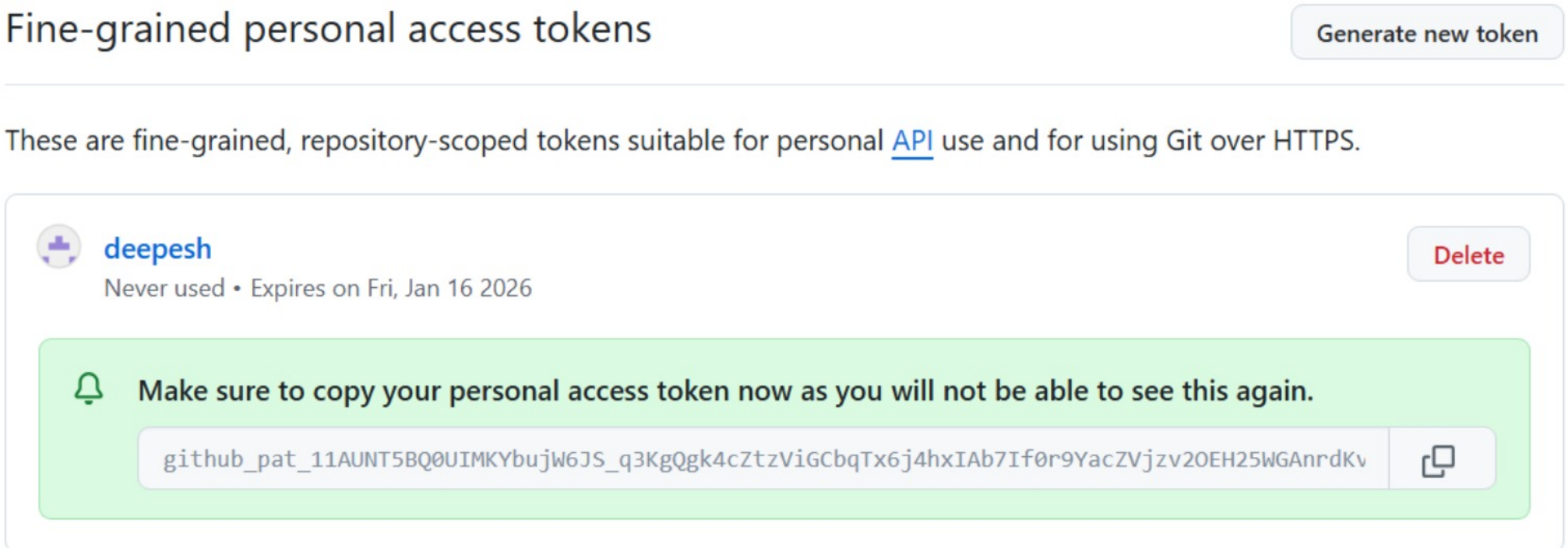
20. Name the token



21. Click on generate token



22. Copy personal access token



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 .
```

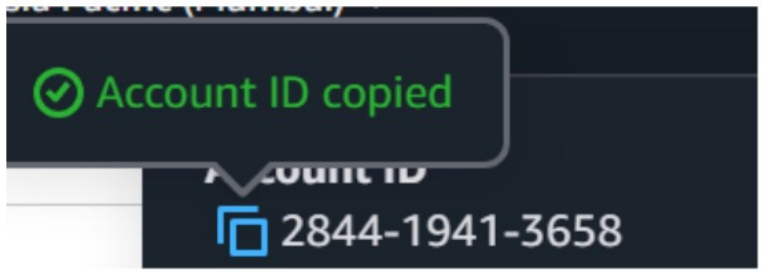
27. Commit the files

```
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 varibles in build section

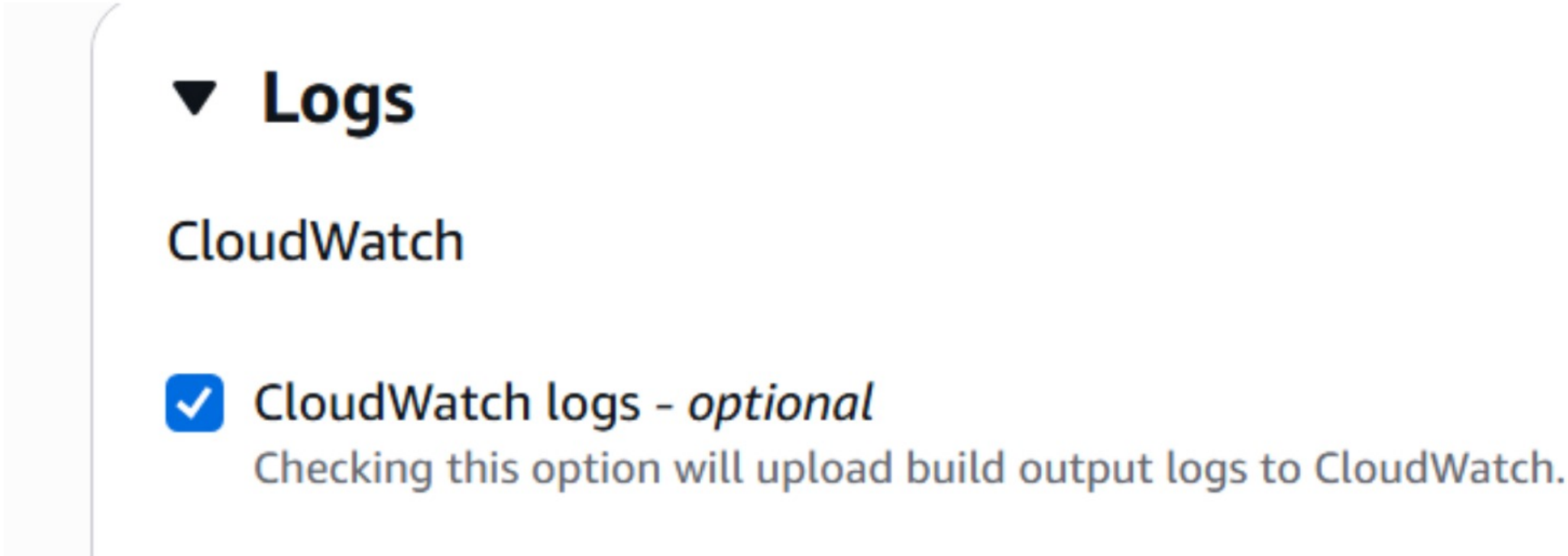


Environment variables

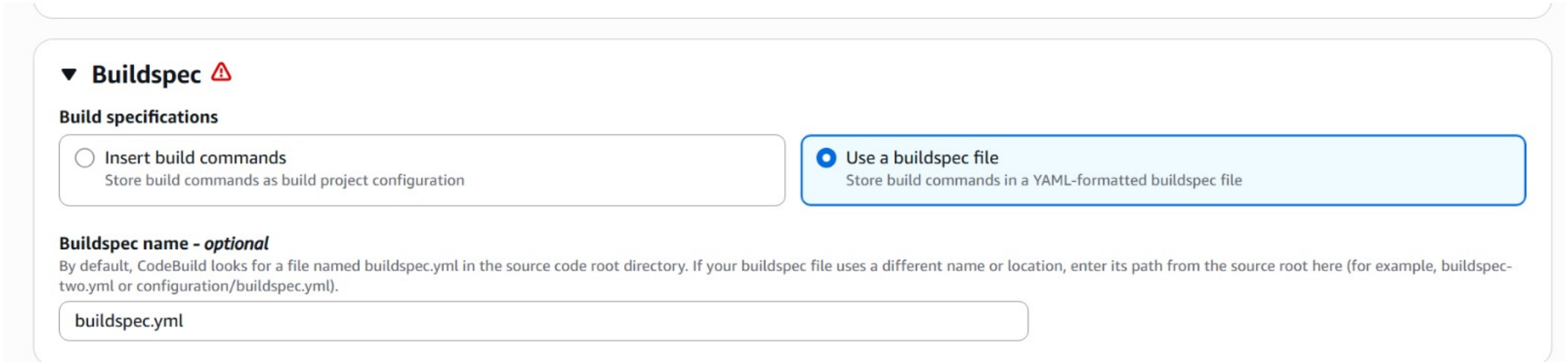
Name	Value	Type	
AWS_ACCOUNT_ID	284419413658	Plaintext	Remove
AWS_DEFAULT_REGION	ap-south-1	Plaintext	Remove
ECR_REPO	brain-tasks-nginx	Plaintext	Remove
IMAGE_TAG	v1	Plaintext	Remove

Add environment variable

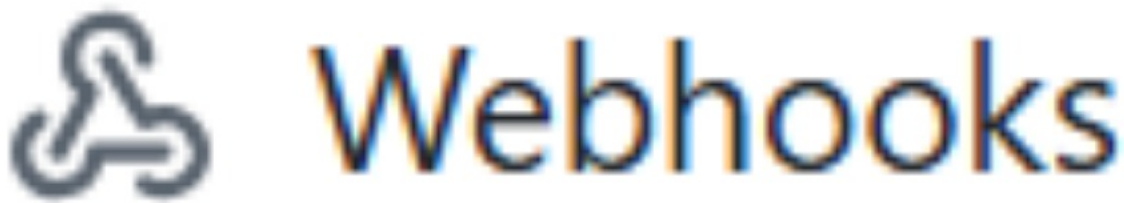
30. Enable cloudewatch logs option for build output logs



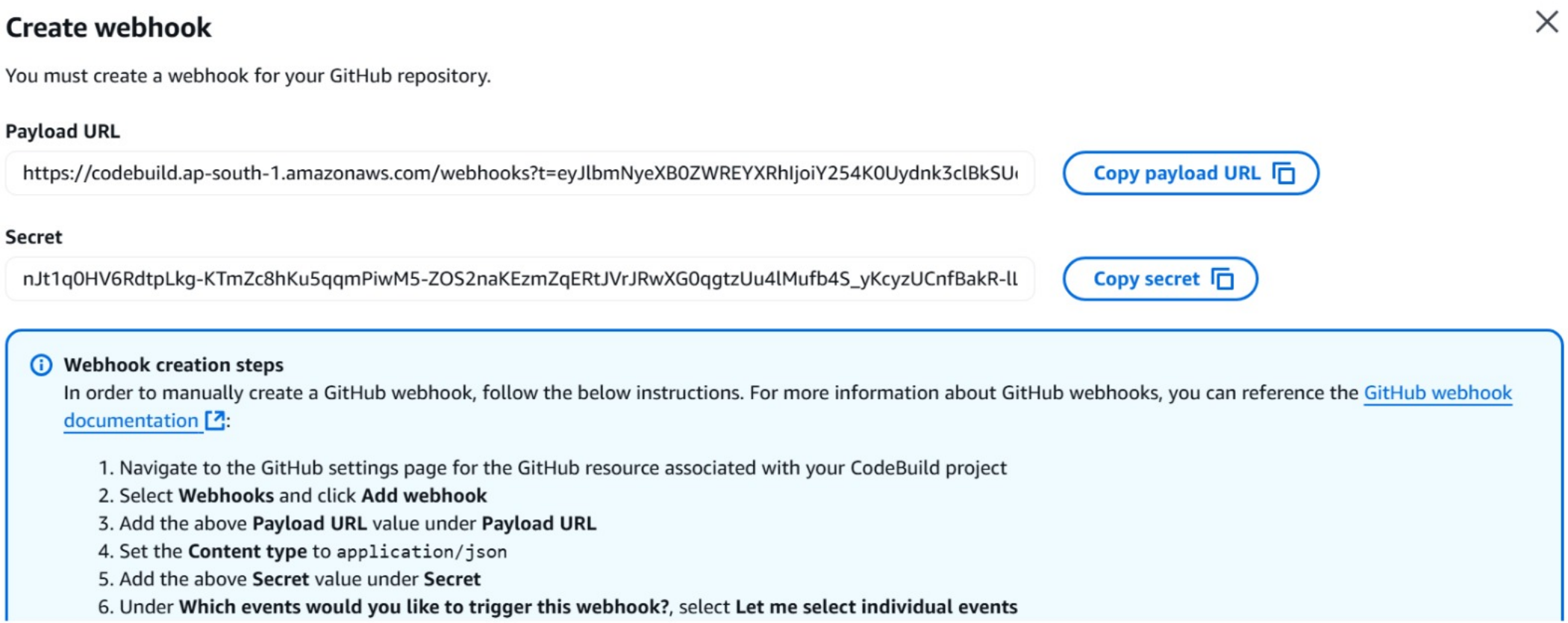
31. Choose buildspec file option



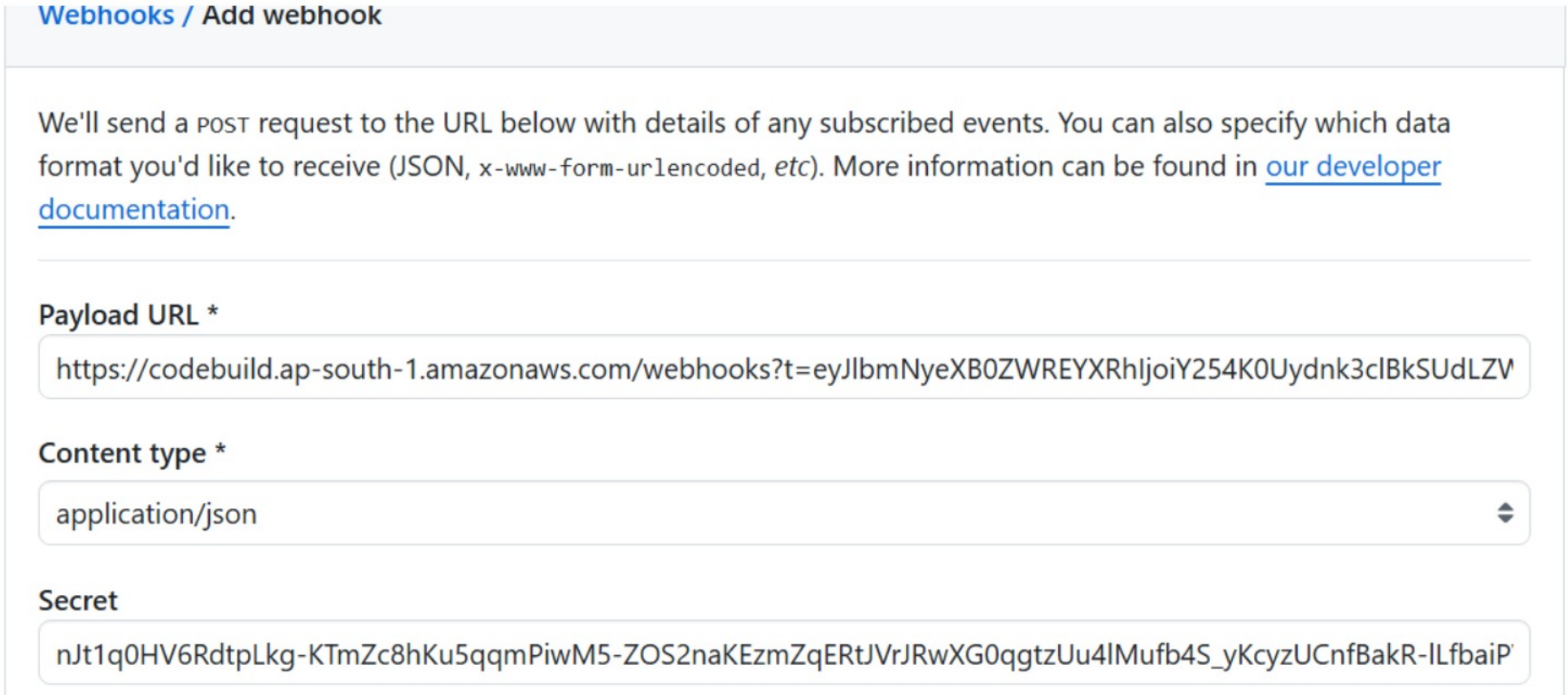
32. Now goto webhook section in github



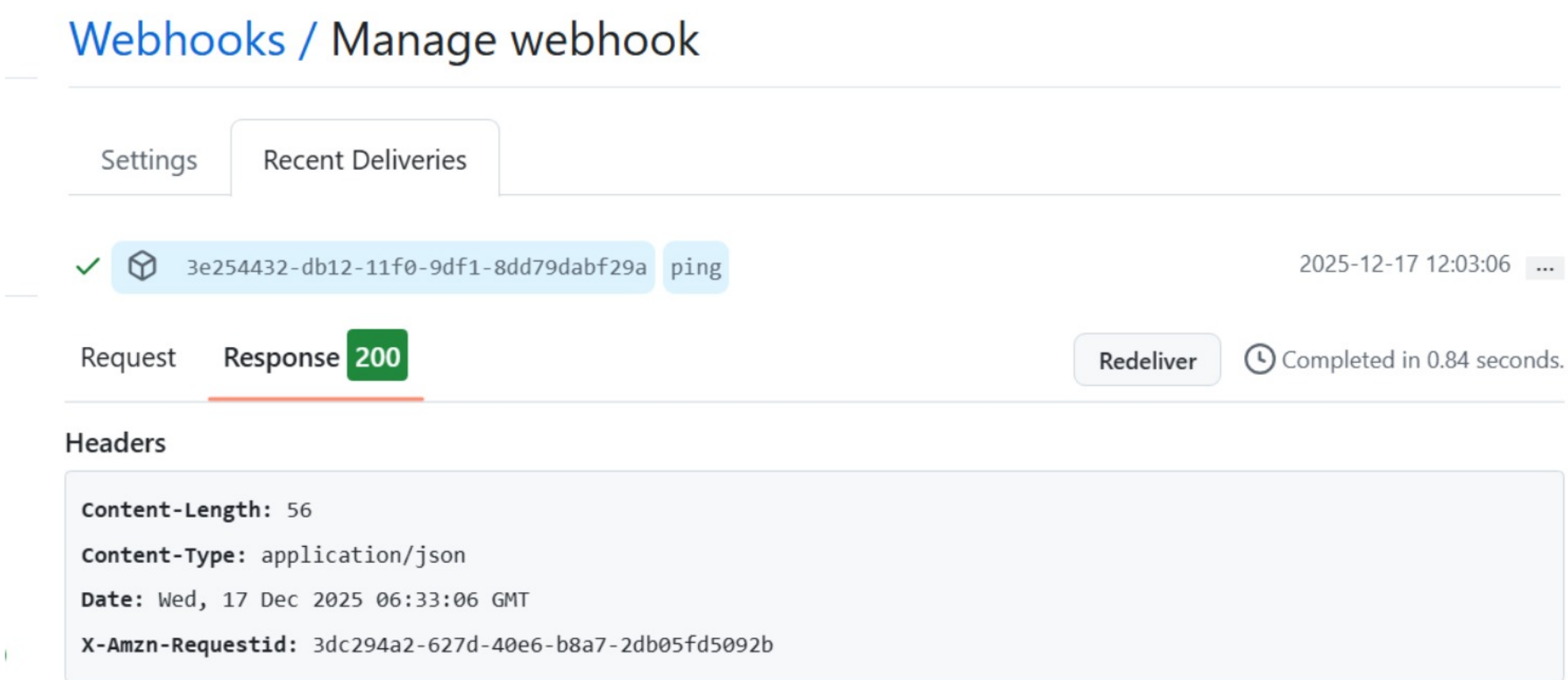
32.1. Copy payload and secret to add in the webhook



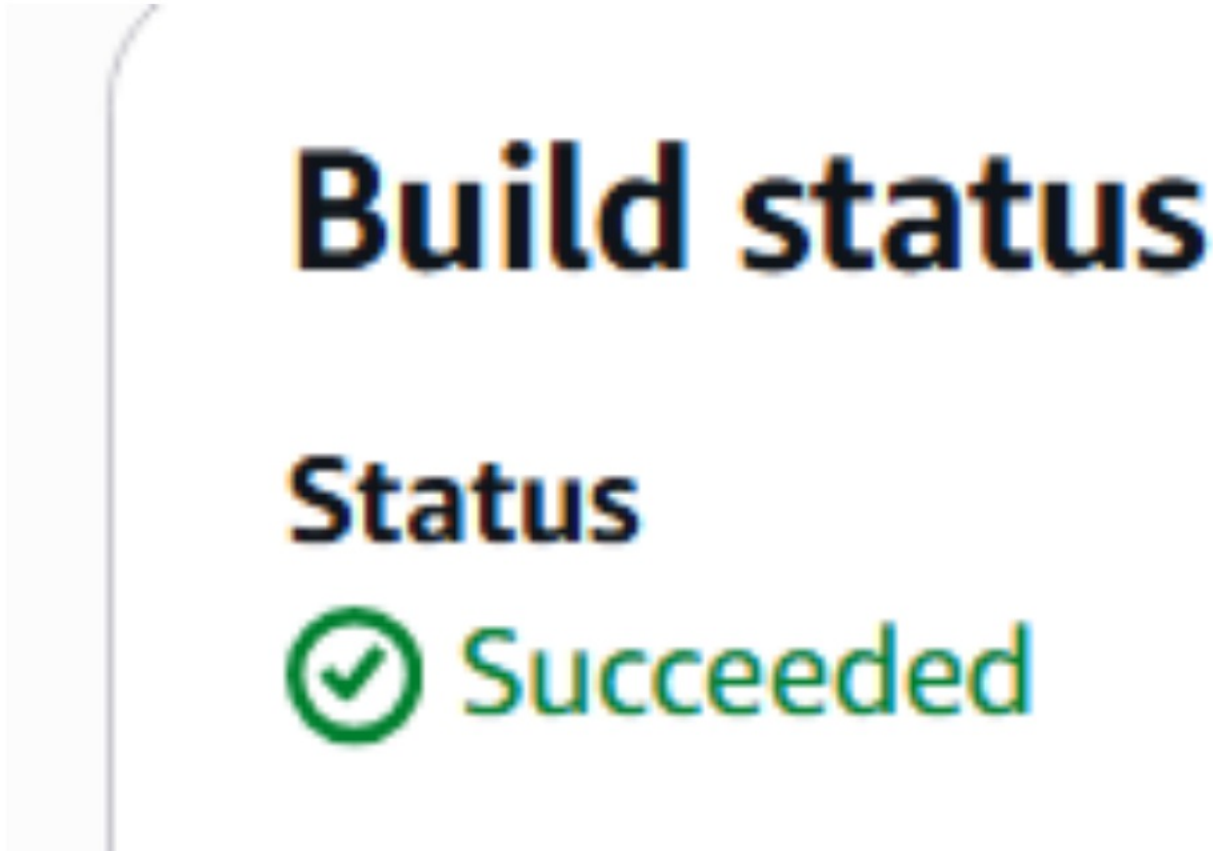
33. Add payload url and secret in webhook



34. Here webhook successfully connected to aws codebuild



35. now build succeeded



36. In the build logs all states are showing succeded

```
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
```