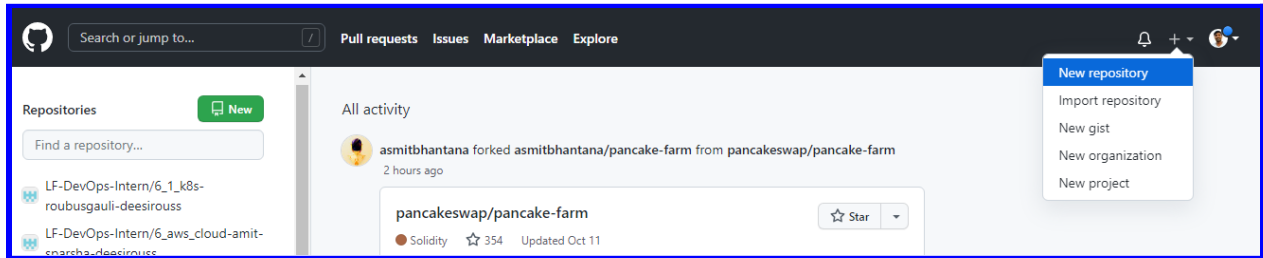


## Creating new repository for s3-react-cicd



## Name s3-react-cicd, Private repo

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


#### Repository template

Start your repository with a template repository's contents.


No template ▾

**Owner \***

**Repository name \***


 deesirouss ▾

/


s3-react-cicd 

Great repository names are short and memorable. Need inspiration? How about [animated-succotash?](#)

**Description (optional)**

☐  **Public**

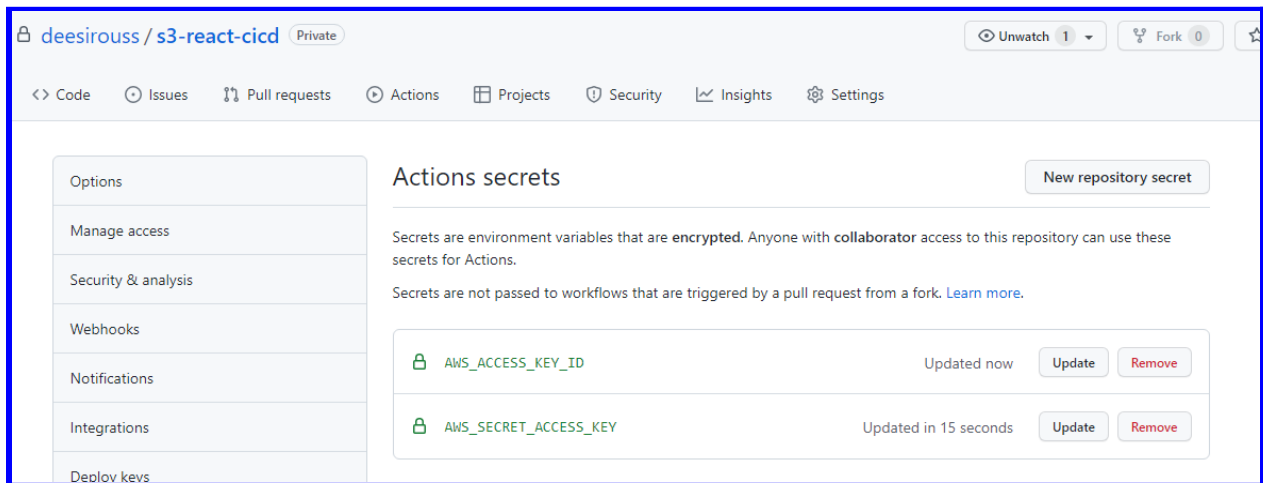
Anyone on the internet can see this repository. You choose who can commit.

☒  **Private**

You choose who can see and commit to this repository.

**Initialize this repository with:**  
Skip this step if you're importing an existing repository.

## Added two secrets (Access Key and Secret Key)



## Pushing Build directory to the GitHub repo

```
[ec2-user@ip-10-15-32-111 s3-react-cicd]$ git push origin main
Username for 'https://github.com': deesirouss
Password for 'https://deesirouss@github.com':
Enumerating objects: 28, done.
Counting objects: 100% (28/28), done.
Compressing objects: 100% (25/25), done.
Writing objects: 100% (27/27), 172.62 KiB | 4.54 MiB/s, done.
Total 27 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/deesirouss/s3-react-cicd.git
   c99e055..860c04e  main -> main
[ec2-user@ip-10-15-32-111 s3-react-cicd]$
```

## Added bucket-object-policy.json to github repository

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Statement1",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::react-bibek-s3/*"
    }
  ]
}
```

## Added s3.sh script to the github repository

```
#!/bin/bash

#checking if bucket exists using aws cli-V2-command
aws s3api head-bucket --bucket react-bibek-s3

if [ $? -ne 0 ];
then
    #if bucket doesn't exist, creating bucket
    aws s3 mb s3://react-bibek-s3 --region us-east-2

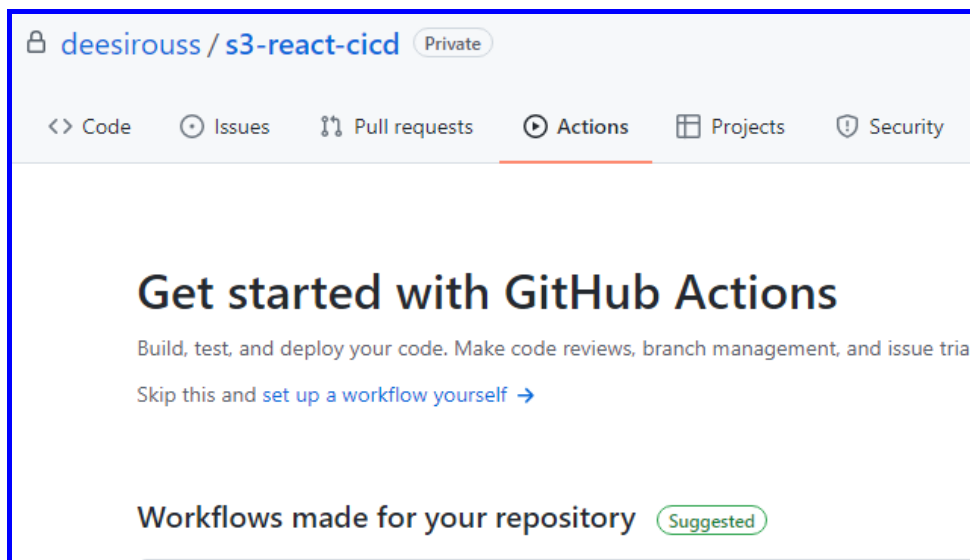
    #turning off block public access
    aws s3api put-public-access-block --bucket react-bibek-s3
--public-access-block-configuration
"BlockPublicAcls=false,IgnorePublicAcls=false,BlockPublicPolicy=false,RestrictPublicBuckets=f
alse"

    #making object public inside bucket
    aws s3api put-bucket-policy --bucket react-bibek-s3 --policy file://bucket-object-policy.json
fi

#syncing up build directory of react app
aws s3 sync build s3://react-bibek-s3

#hosting static website in this bucket
aws s3 website s3://react-bibek-s3 --index-document index.html --error-document error.html
```

## Creating GitHub Actions → Actions → Set up a workflow yourself



## Workflow yml file

```
# This is a basic workflow to help you get started with Actions

name: CI
# Controls when the workflow will run
on:
  # Triggers the workflow on push or pull request events but only for the main branch
  push:
    branches: [ main ]
  pull_request:
    branches: [ main ]

# Allows you to run this workflow manually from the Actions tab
workflow_dispatch:

# A workflow run is made up of one or more jobs that can run sequentially or in parallel
jobs:
  # This workflow contains a single job called "build"
  build:
    # The type of runner that the job will run on
    runs-on: ubuntu-latest

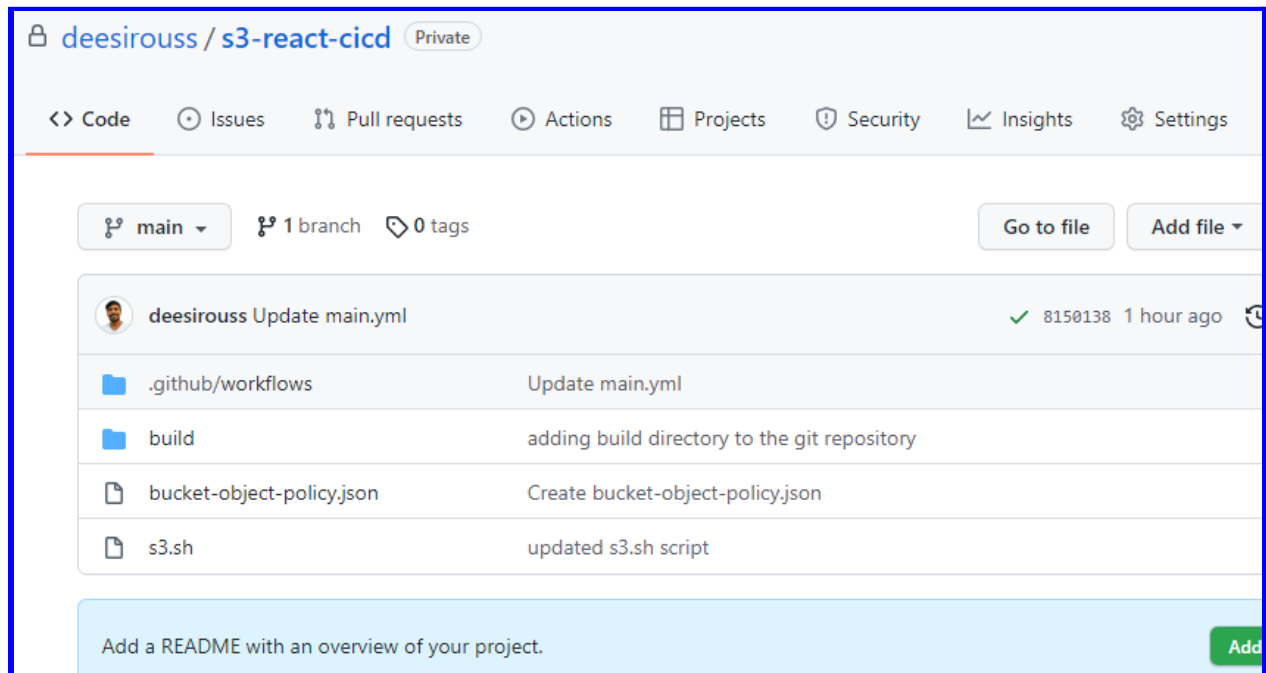
    # Steps represent a sequence of tasks that will be executed as part of the job
    steps:
      # Checks-out your repository under $GITHUB_WORKSPACE, so your job can access it
      - uses: actions/checkout@v2

      #AWS credentials
      - name: Configure AWS credentials
        uses: aws-actions/configure-aws-credentials@v1
        with:
          aws-access-key-id: ${ secrets.AWS_ACCESS_KEY_ID }
          aws-secret-access-key: ${ secrets.AWS_SECRET_ACCESS_KEY }
          # TODO Change your AWS region here!
          aws-region: us-east-2

      - name: Chmod s3.sh
        run: |
          chmod +x ./s3.sh

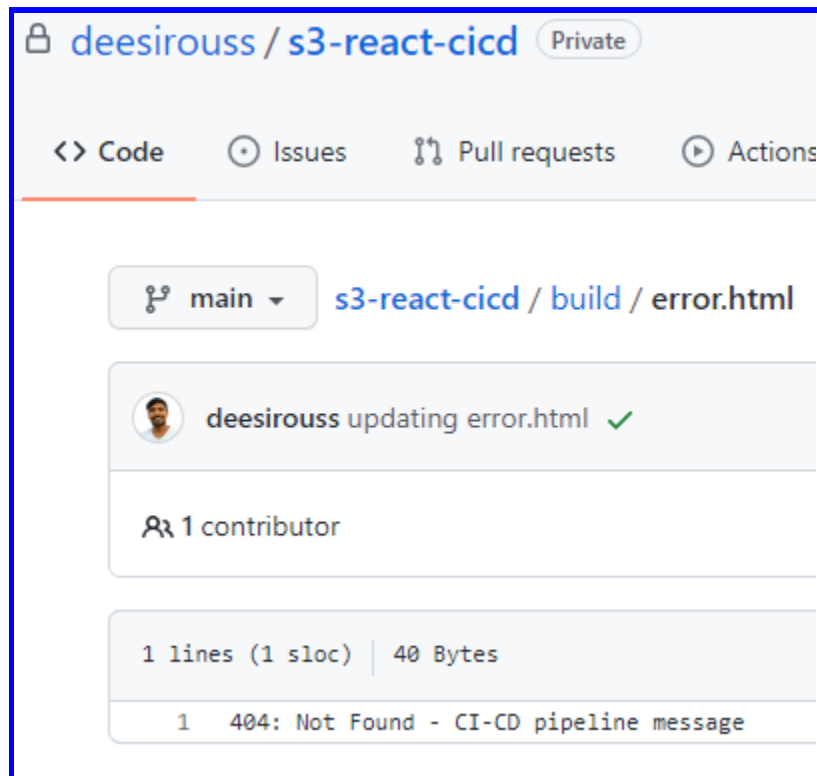
      - name: Running script s3.sh
        run: ./s3.sh
        shell: bash
```

We have all these files in the github repository

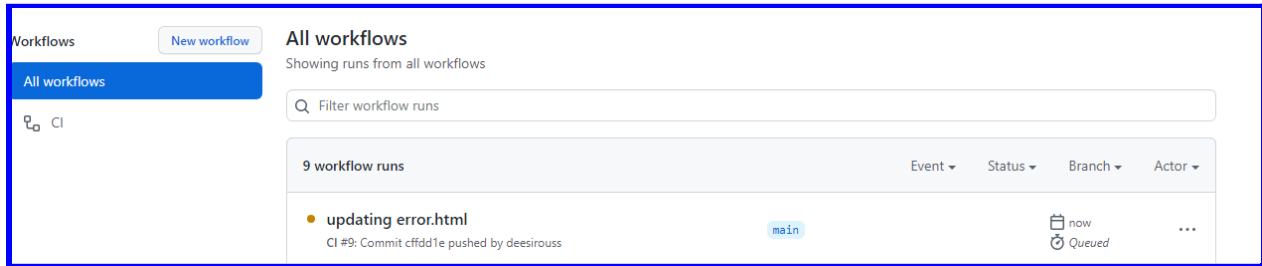


## Changing the message of error.html file

From “404: Not Found ” to “404: Not Found - CI-CD pipeline message”

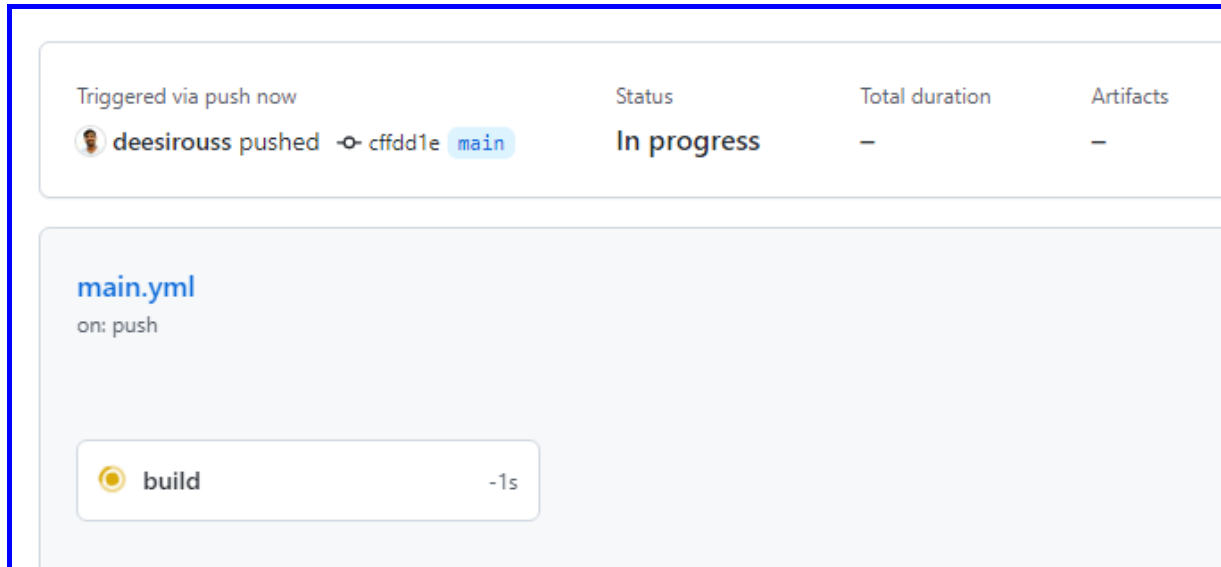


## Workflow triggered



The screenshot shows the GitHub Actions interface. On the left, the 'Workflows' sidebar is visible with 'All workflows' selected. The main area is titled 'All workflows' and shows 'Showing runs from all workflows'. A search bar is present. Below, a table lists '9 workflow runs'. The first run is highlighted, showing it was triggered by 'deesirouss' pushing commit 'cffdd1e' to the 'main' branch. The status is 'queued'.

Event	Status	Branch	Actor
updating error.html	queued	main	deesirouss



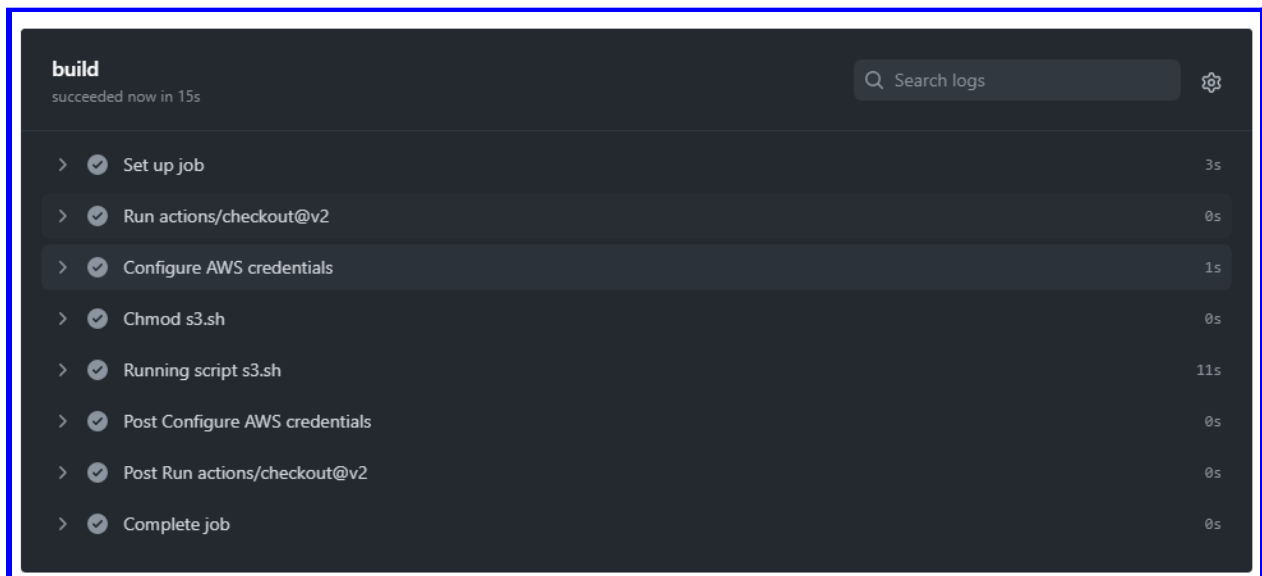
The screenshot shows the details of a workflow run. It indicates the workflow was 'Triggered via push now' by 'deesirouss' pushing commit 'cffdd1e' to the 'main' branch. The status is 'In progress'. The total duration is shown as a dash. Below, the 'main.yml' file is displayed, showing the trigger 'on: push'. A job named 'build' is listed with a duration of '-1s'.

Triggered via push now  
deesirouss pushed cffdd1e main  
Status: In progress  
Total duration: -  
Artifacts: -

main.yml  
on: push

build -1s

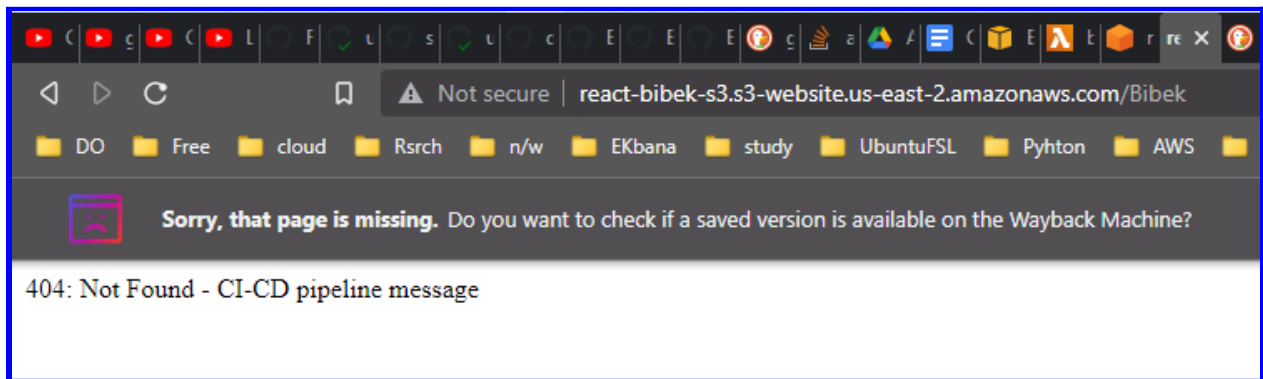
## All jobs executed successfully



The screenshot shows the 'build' job log. It indicates the job 'succeeded now in 15s'. A search bar for logs is present. The log lists the following steps, all of which were successful (indicated by checkmarks):

- Set up job (3s)
- Run actions/checkout@v2 (0s)
- Configure AWS credentials (1s)
- Chmod s3.sh (0s)
- Running script s3.sh (11s)
- Post Configure AWS credentials (0s)
- Post Run actions/checkout@v2 (0s)
- Complete job (0s)

## While browsing other than root for error.html message



Successfully runned the workflow and got the change respectively.

In this way we configured pipeline to update static hosting of s3 with react file