

nextwork.org

Connect a GitHub Repo with AWS

RD

rdanu.inbox@gmail.com

The screenshot shows a terminal window with the following content:

```
index.jsp
src > main > webapp > index.jsp > ...
1  <html>
2
3  <body>
4
5  <h2>Hello <DHANUSH>!</h2>
6
7  <p>This is my NextWork web application working!</p>
8
9  </body>
10 </html>
11
12
13

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS
Verifying      : perl-TermReadKey-2.38-9.amzn2023.0.2.x86_64
Verifying      : perl-lib-0.65-477.amzn2023.0.6.x86_64

7/8
8/8

Installed:
git-2.47.1-1.amzn2023.0.2.x86_64          git-core-2.47.1-1.amzn2023.0.2.x86_64          git-core-doc-2.47.1-1.amzn2023.0.2.noarch
perl-Error-1:0.17029-5.amzn2023.0.2.noarch    perl-File-Find-1.37-477.amzn2023.0.6.noarch  perl-Git-2.47.1-1.amzn2023.0.2.noarch
perl-TermReadKey-2.38-9.amzn2023.0.2.x86_64    perl-lib-0.65-477.amzn2023.0.6.x86_64

Complete!
[ec2-user@ip-172-31-11-122 nextwork-web-project]$ git --version
git version 2.47.1
[ec2-user@ip-172-31-11-122 nextwork-web-project]$
```

Introducing Today's Project!

What is GitHub?

GitHub is a cloud-based platform for version control and collaboration using Git. I used it to create a repository, connect my local project, and push code changes from EC2.

One thing I didn't expect...

One thing I didn't expect in this project was the need to use a GitHub token instead of a password for authentication. It was a new way of securely accessing my repository.

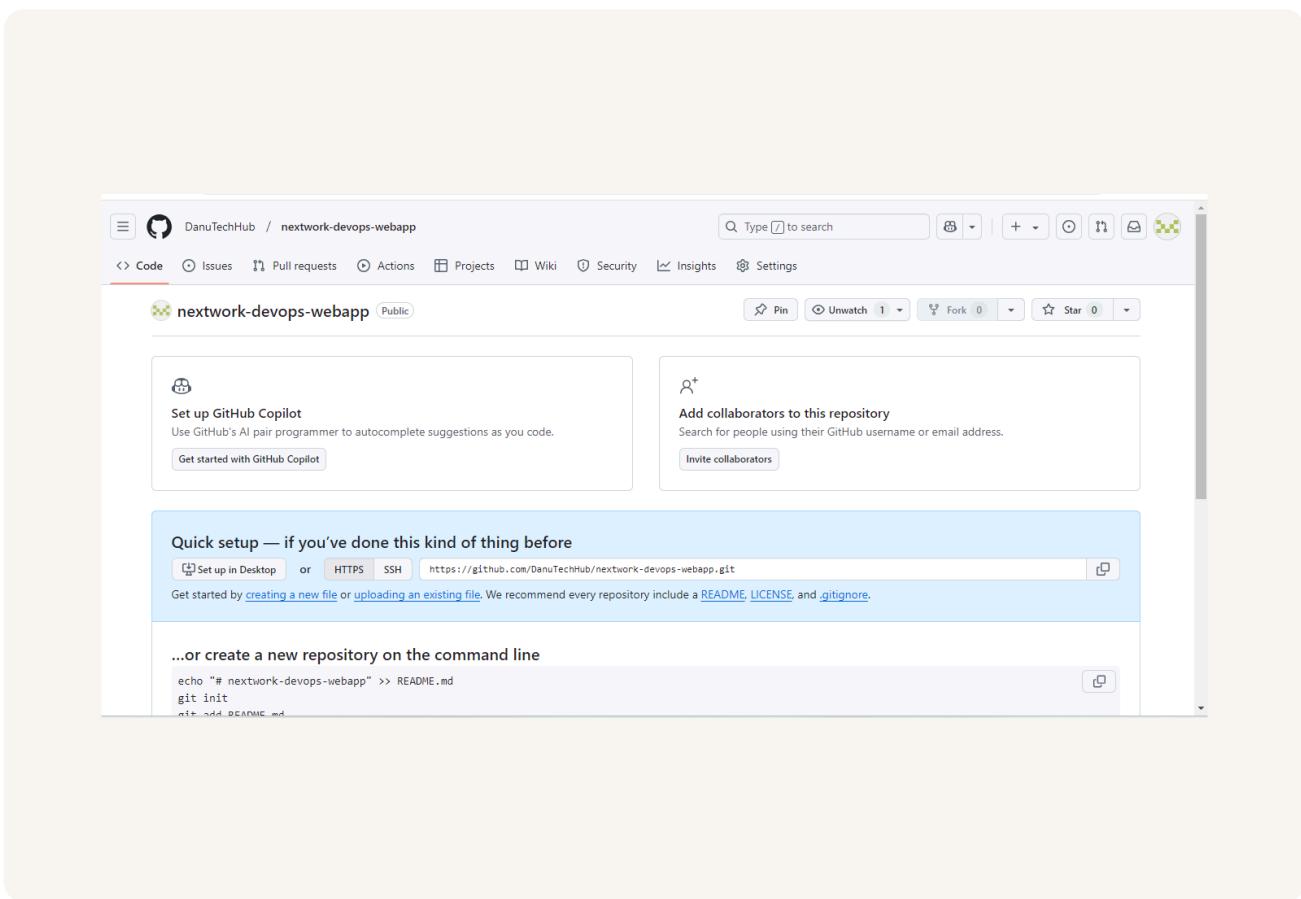
This project took me...

This project took me approximately [your estimated time] to complete, including setting up Git, connecting GitHub, and pushing code changes. It took longer than expected due to authentication and SSH setup issues.

Git and GitHub

Git is a version control system for tracking code changes and collaboration. I installed it using: sudo dnf update -y sudo dnf install git -y

GitHub is a cloud-based platform for version control and collaboration using Git. I'm using GitHub in this project to store code, track changes, and integrate with AWS for deployment.

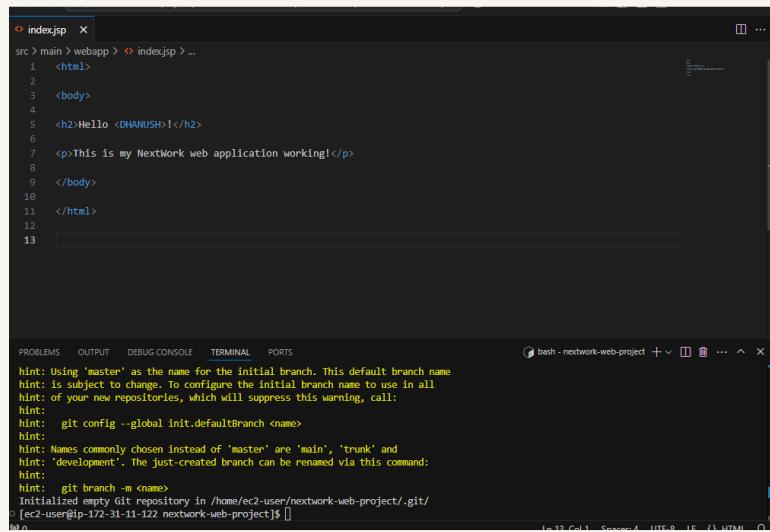


My local repository

A Git repository is a storage location that tracks and manages code changes, allowing collaboration and version control. It can be local on a computer or hosted on platforms like GitHub.

git init is a command that initializes a new Git repository in a directory, allowing version control for the project. I ran git init in my web app folder to start tracking changes and manage my code efficiently.

After running git init, the response from the terminal was that a new Git repository was initialized. A branch in Git is an independent line of development that allows you work on new features without affecting the main code until changes are merged.



The screenshot shows a terminal window with the following text:

```
index.jsp  x
src > main > webapp > index.jsp > ...
1  <html>
2
3  <body>
4
5  <h2>Hello <DHANUSH>!</h2>
6
7  <p>This is my NextWork web application working!</p>
8
9  </body>
10 </html>
11
12
13

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
hint: Using "master" as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/ec2-user/nextwork-web-project/.git/
[ec2-user@ip-172-31-11-122 nextwork-web-project]$
```

To push local changes to GitHub, I ran three commands

git add

The first command I ran was git remote add origin <https://github.com/DanuTechHub/nextwork-devops-webapp.git>, which links my local repository to GitHub. A staging area is where Git stores changes before committing them. Next git add to save changes.

git commit

The second command I ran was git commit -m "Updated index.jsp with new content", which saves the staged changes to the local repository with a message. Using -m means adding a commit message to describe the changes made.

git push

The third command I ran was git push -u origin master, which uploads my committed changes to the GitHub repository. Using -u means setting the upstream branch, so future pushes can be done with just git push.

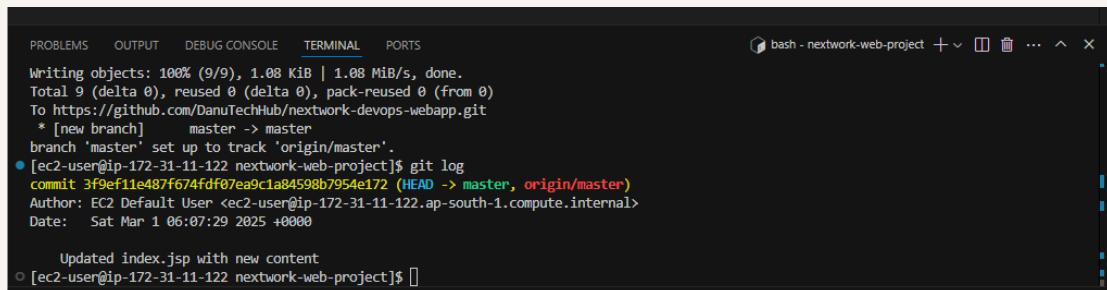
Authentication

When I commit changes to GitHub, Git asks for my credentials because authentication is required to verify my identity and ensure I have permission to push changes to the repository. This helps secure the repository and prevent unauthorized access.

Local Git identity

Git needs my name and email because it uses them to identify me as the author of commits. This information is recorded in the commit history to track contributions and collaboration.

Running `git log` showed me the history of commits in my repository, including commit IDs, author details, dates, and messages describing the changes made.



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Writing objects: 100% (9/9), 1.08 KiB | 1.08 MiB/s, done.
Total 9 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/DanuTechHub/nextwork-devops-webapp.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
● [ec2-user@ip-172-31-11-122 nextwork-web-project]$ git log
commit 3f9ef11e487f674fdf07ea9c1a84598b7954e172 (HEAD -> master, origin/master)
Author: EC2 Default User <ec2-user@ip-172-31-11-122.ap-south-1.compute.internal>
Date:   Sat Mar 1 06:07:29 2025 +0000

    Updated index.jsp with new content
○ [ec2-user@ip-172-31-11-122 nextwork-web-project]$
```

GitHub tokens

GitHub authentication failed when I entered my password because GitHub no longer supports password authentication for Git operations. Instead, it requires a personal access token (PAT) for secure authentication.

A GitHub token is a secure way to authenticate and access repositories without using a password. I'm using one because GitHub requires tokens for safer authentication in Git operations.

I could set up a GitHub token by going to GitHub Settings > Developer Settings > Personal Access Tokens, generating a new token with the required permissions, and copying it for authentication in Git operations.

RD

rdanu.inbox@gmail.com
NextWork Student

NextWork.org

The screenshot shows the GitHub interface for creating a new personal access token. The left sidebar has sections for GitHub Apps, OAuth Apps, Personal access tokens (selected), Fine-grained tokens (Preview), and Tokens (classic). The main area is titled "New personal access token (classic)". It includes a note about personal access tokens functioning like OAuth tokens, a note field containing "Generated for EC2 Instance Access. NextWork DevOps project serie", a dropdown for expiration set to "30 days" (expiring Mar 31 2025), and a "Select scopes" section. The "repo" scope is selected, granting full control of private repositories. Other available scopes include repo:status, repo_deployment, public_repo, repo:invite, security_events, workflow, access commit status, access deployment status, access public repositories, access repository invitations, read and write security events, and update GitHub Action workflows.

New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

Generated for EC2 Instance Access. NextWork DevOps project serie

What's this token for?

Expiration *

30 days ▾ The token will expire on Mon, Mar 31 2025

Select scopes

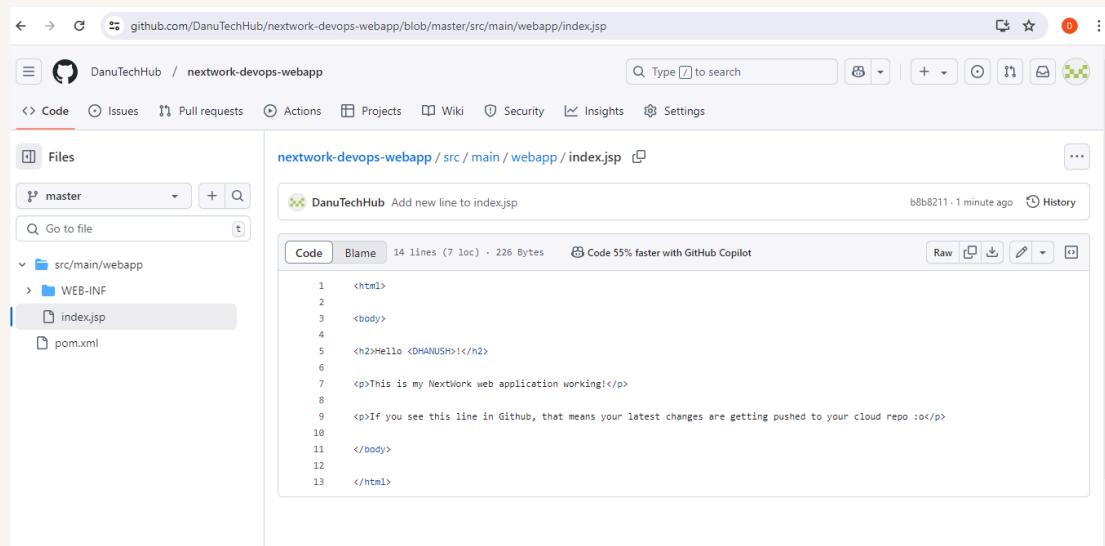
Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

Scopes	Description
<input checked="" type="checkbox"/> repo	Full control of private repositories
<input type="checkbox"/> repo:status	Access commit status
<input type="checkbox"/> repo_deployment	Access deployment status
<input type="checkbox"/> public_repo	Access public repositories
<input type="checkbox"/> repo:invite	Access repository invitations
<input type="checkbox"/> security_events	Read and write security events
<input type="checkbox"/> workflow	Update GitHub Action workflows

Making changes again

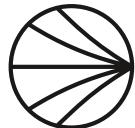
I wanted to see Git working in action, so I updated the index.jsp file in the nextwork-web-project. I couldn't see the changes in my GitHub repo initially because I hadn't committed and pushed the updates yet.

I finally saw the changes in my GitHub repo after staging them with git add ., reviewing the differences using git diff --staged, committing with git commit -m "Add new line to index.jsp", and pushing the updates using git push.



A screenshot of a GitHub repository page for 'DanuTechHub / nextwork-devops-webapp'. The 'Code' tab is selected in the navigation bar. On the left, the file structure shows 'master' and 'src/main/webapp/index.jsp'. The main pane displays the content of 'index.jsp':

```
1 <html>
2 <body>
3 <h2>Hello <DHANUSH>!</h2>
4 <p>This is my NextWork web application working!</p>
5 <p>If you see this line in Github, that means your latest changes are getting pushed to your cloud repo :o</p>
6
7
8
9
10
11
12
13 </html>
```



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

