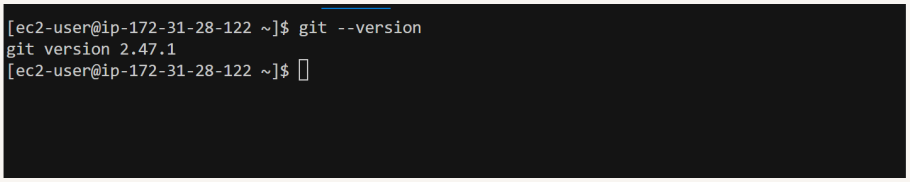
nextwork.org

Connect a GitHub Repo with AWS

Kayira Bertrand



Kayira Bertrand
NextWork Student

Introducing Today's Project!

Today I'm going to set up a Git Repository for my Web App's code. This is Project II in my 7 day DevOps Challenge. By the end of this project, the code that I write for my Java Web will be stored securely in GitHub.

Key tools and concepts

Services I used were GitHub, Amazon EC2, key pairs and VS Code. Key concepts I learnt include setting up a Git Repository, the difference between Git and GitHub and the commands for staging, saving and pushing changes to my code. I also learnt about upstream repositories and setting up a remote origin and branches.

Project reflection

This project too aproximately 5 hours including demo time, documentations.

I wanted to challenge myself with DevOps Skills and this project really help me a lot.

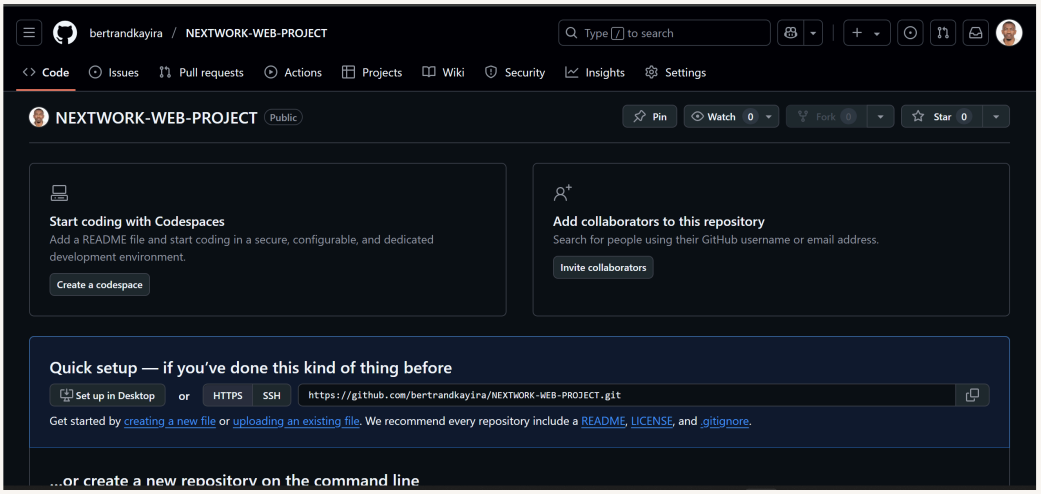
I will commence another project in few comming days.

Kayira Bertrand
NextWork Student

Git and GitHub

Git is a version control system used to track changes that we make to our code. It's also helpful for collaboration. With Git, you can see who made what change and the handover from development to deployment is smoother while looking at a share repo

Github is a platform that let's us store and share and collaborate on our code. It's called Github because it used Git as the tool for version control which is one of it's main features. Github adds much more that makes collaboration much easier.



My local repository

A Git repository is a storage space for your code, managed using Git, that tracks all changes made to the files within it. Hosting a repository in the cloud, like on GitHub, allows for collaboration with other engineers.

git init is a command that initialized git in our local repository. I ran git init in my Web App project folder, which tells our terminal that we want to track changes locally.

After running git init, the response from the terminal was that we initialized git and bt default we are using the 'main' branch. A branch in Git is like a version of your code you make changes to over a branch, and then merge those changes into main

```
/home/ec2-user/nextwork-web-project
[ec2-user@ip-172-31-28-122 nextwork-web-project]$ git init
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-172-31-28-122 nextwork-web-project]$
```

Kayira Bertrand
NextWork Student

To push local changes to GitHub, I ran three commands

git add

The first command I ran was git add . which adds our changes to a staging area. A staging area is like the place to review all changes made to the code so that you can decide would like to keep.

git commit

The second command I ran was git commit which is the command for saving the changes in our staging area ... Using '-m' means we are also leaving a message for that commit. Fro Example "Made changes to index.jsp "

git push

The third command I ran was `git push -u origin master`. This command pushes the code changes we saved to our remote origin. Using '-u' is a special flag that makes our remote origin the default UPSTREAM Repository.

Kayira Bertrand
NextWork Student

Authentication

When I commit changes to GitHub, Git asks for my credentials because it needs to authenticate me before letting me make code changes to the GitHub Repository. It needs to do this to know that I have the right to change the code.

Local Git identity

Git needs my name and email because it's a version control system which means it is used for tracking WHO made WHAT change to a piece of code. To really identify the people that made X changes, Git will need to know you name and email.

Running git log showed me that by default, Git is saving our code changes to a username called "EC2 Default User" instead of our actual name and details.

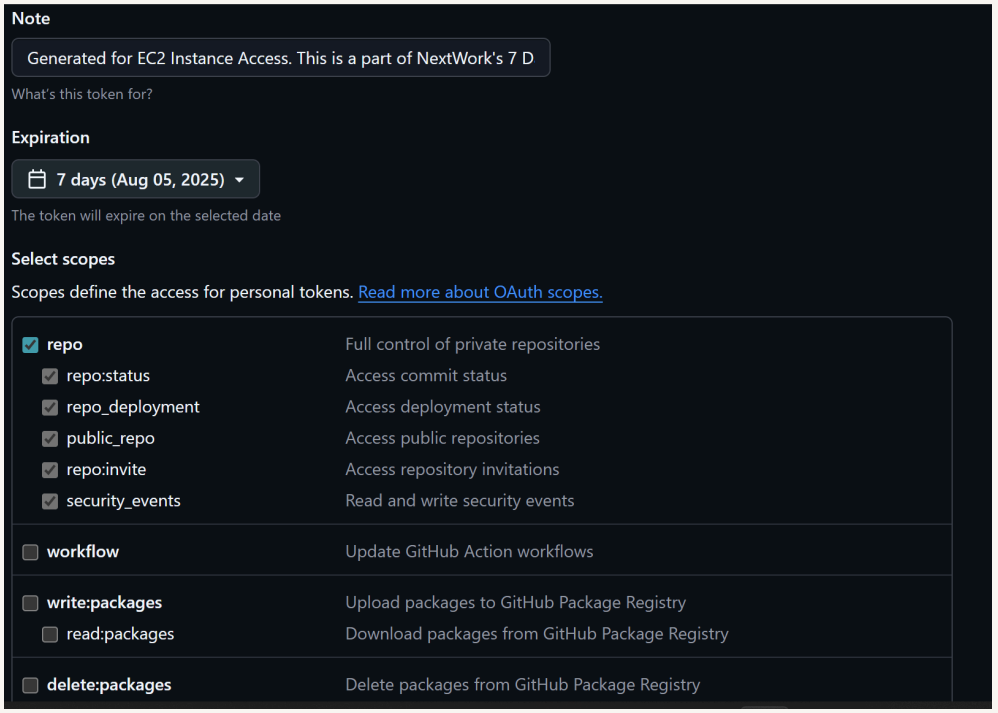
```
[ec2-user@ip-172-31-34-7 nextwork-web-project]$ git log
commit d51da230c72a9f0649b1910ac7bccf75757a3bb0 (HEAD -> master)
Author: EC2 Default User <ec2-user@ip-172-31-34-7.us-west-2.compute.internal>
Date:   Tue Jul 29 09:56:26 2025 +0000

    Updated index.jsp with new content
[ec2-user@ip-172-31-34-7 nextwork-web-project]$
```

GitHub tokens

GitHub authentication failed when I entered my password when I entered my password because password authentication support was already removed in 2021. There are too many security risks associated with entering the password over the terminal to GitHub, so more secure ways are now available.

A GitHub token is like a temporary password that grants access to your GitHub account. I'm using one in this project because it let me safely authenticate to my GitHub repository while in my EC2 Instance.

I could set up a GitHub token by visiting the developer settings in GitHub, and we set one up that expires in 7 days and only allow permissions to repositories

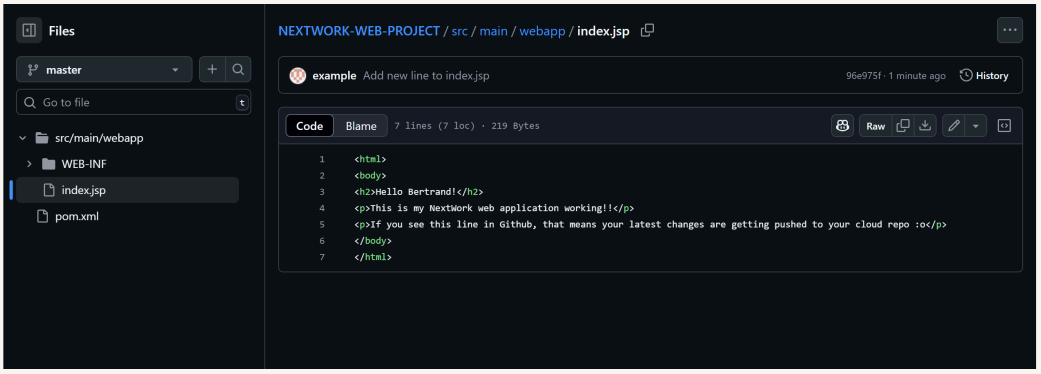

Note
Generated for EC2 Instance Access. This is a part of NextWork's 7 D
What's this token for?
Expiration
7 days (Aug 05, 2025)
The token will expire on the selected date
Select scopes
Scopes define the access for personal tokens. Read more about OAuth scopes.
repo
Full control of private repositories
repo:status
Access commit status
repo_deployment
Access deployment status
public_repo
Access public repositories
repo:invite
Access repository invitations
security_events
Read and write security events
workflow
Update GitHub Action workflows
write:packages
Upload packages to GitHub Package Registry
read:packages
Download packages from GitHub Package Registry
delete:packages
Delete packages from GitHub Package Registry

Kayira Bertrand
NextWork Student

Making changes again

I wanted to see Git working in action, so I mad eanother change to my index.jsp file. I couldn't see the changes in my GitHub repo initially because I had not added committed or pushed those changes.

I finally saw the changes in my GitHub repo after running the three commands and then refreshing index.jsp in my GitHub Repository.



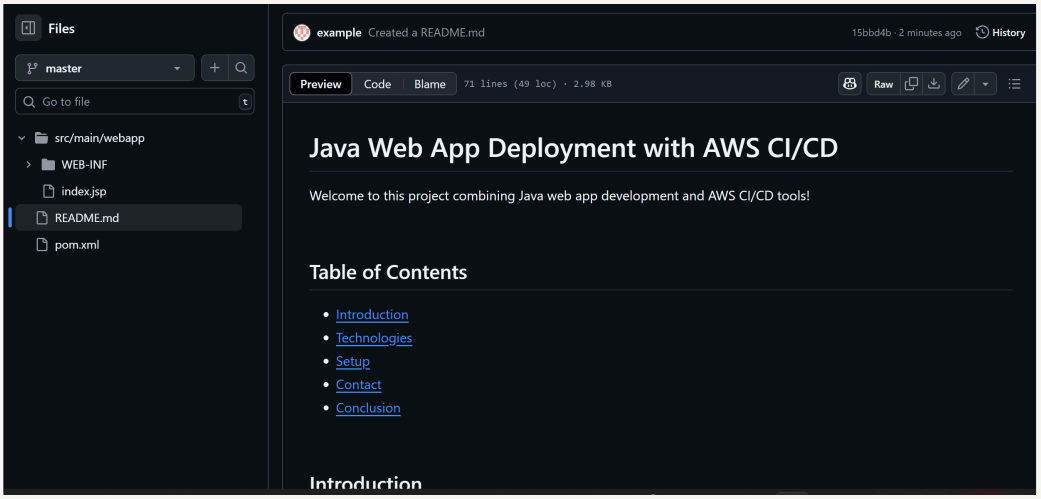
Setting up a README file

As a finishing touch to my GitHub repository, I added a README file, which is document that lives at the root of my repository and introduces my project, what it does, how to use it, tec. I added a README file by running "touch README.md" in my EC2 Instance's Terminal.

My README is written in Markdown because that is the common text language that is lets us format text in a document. Special characters can help you format text in Markdown, such as a backtick (') for code or double asterisks (**) for bolding and more.

My README file has 6 sections that outline a table of content, an introduction, technologies (i.e the AWS services I used), how to set up the repository locally and then a section for contacting me + a thank you message ...

Kayira Bertrand
NextWork Student



The place to learn & showcase your skills

Check out nextwork.org for more projects