



# Set up a Git Repo with AWS



Nikhil Bhan

The screenshot shows a GitHub repository interface for a project named "nextwork-web-project". The repository path is "Developer Tools > CodeCommit > Repositories > nextwork-web-project". The current branch is "main". The "index.jsp" file is displayed with the following content:

```
1 <html>
2
3 <body>
4
5 <h2> Hello Nikhil </h2>
6
7 <p> This is my NextWork web application working! </p>
8 <p> Yo! If you see this line in CodeCommit, your latest changes are saved in the origin. </p>
9
10 </body>
11
12 </html>
13
```



# Introducing Today's Project!

## What is AWS CodeCommit?

AWS CodeCommit is a service that enables users to host their Git repositories securely in the cloud. It allows developers to collaborate together online and keep their files up to date while working on their local workstations simultaneously.

## How I'm using CodeCommit in this project

In this project I saved the local repository of my code in AWS Cloud9 to the origin repository in AWS CodeCommit. I was able to view changes in my file from CodeCommit to verify that the repository was saved.

## One thing I didn't expect...

I wasn't expecting how easy it was to send changed files and have them committed to the origin repository. The commands were simple to understand and I was able to run them in the CLI terminal, which was fun.

## This project took me...

This project took me 45 minutes to complete. In addition, I also used 15 minutes to write my documentation.



# Create a Git repository

## What is Git?

Git is a version control system; it's a way for developers to write their code and track changes that happen over time.

## What is a Git repository?

A Git repository is a folder where projects and files can be stored. This folder is accessible for developers to collaborate on their projects and track changes in their code, as well as reviewing old versions of their work in case they need it.

The screenshot shows the AWS CodeCommit interface after a repository has been successfully created. A green success bar at the top indicates 'Repository successfully created'. The main content area shows the repository details for 'nextwork-web-project'. It includes sections for 'Connection steps' (with tabs for HTTPS, SSH, and HTTPS (GRC), currently set to HTTPS), 'Prerequisites' (warning about Git client version), 'Git credentials' (warning about IAM user setup), and 'Clone the repository' (providing a command-line clone URL). A 'Clone URL' dropdown button is also visible.



# My first commit

I initialized a Git repo by running the command "git init -b main"

## I used three commands to push local changes to CodeCommit

The first command I ran was "git add". This command adds changed files to a location where they're kept before being sent out.

The second command I ran was "git commit". This command saves the changes that were made in the local repository and a message can be included with 'm' so other developers can understand why that change was made.

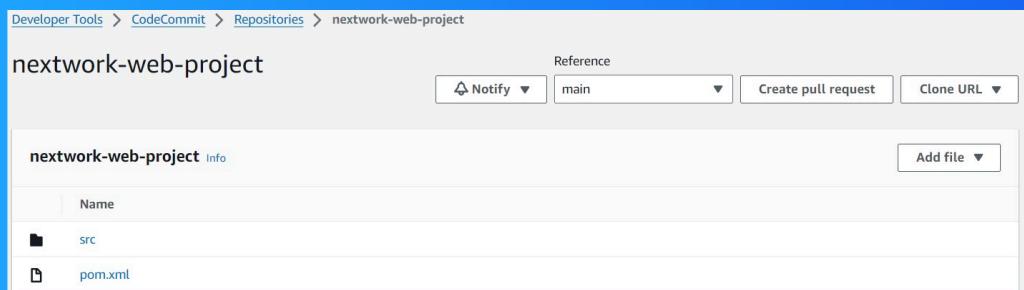
The third command I ran was "git push". This command uploads the changed files to the origin repository, which is "nextwork-web-project" in AWS CodeCommit.

N

Nikhil Bhan  
NextWork Student

[NextWork.org](http://NextWork.org)

# Creating a Repository





# Git in action

## Making changes in Cloud9

I wanted to see Git working in action, so I added two lines to my index.jsp file.

## The importance of committing changes

I tried seeing these changes in my CodeCommit repository, but this didn't work because my changes were saved to the local repository in Cloud9. They were never sent upstream towards my origin repository in CodeCommit.

I finally saw the changes in my CodeCommit repository after I ran 3 commands in the Cloud9 terminal. I needed to run the git commands, which were: - git add . - git commit - git push

The screenshot shows the Cloud9 developer tools interface. The path is "Developer Tools > CodeCommit > Repositories > nextwork-web-project". The repository name is "nextwork-web-project". The "Reference" dropdown is set to "main". The "Edit" button is visible. The code editor displays the "index.jsp" file content:

```
1 <html>
2
3 <body>
4
5 <h2> Hello Nikhil </h2>
6
7 <p> This is my NextWork web application working! </p>
8 <p> Yo! If you see this line in CodeCommit, your latest changes are saved in the origin. </p>
9
10 </body>
11
12 </html>
13
```



NextWork.org

# Everyone should be in a job they love.

Check out nextwork.org for  
more projects

