GIT-Version control system by using Amazon web services

What is git?

Git is a version control system for tracking changes in computer files and is used to help coordinate work among several people on a project while tracking progress over time . in other words, it is a tool that facilitates source code management in software development.

Github- To provide internet hosting for version control and software development,github makes use of git.

Some popular git hosting services:

\*github

\*source forge

\*gitlab

\*bitbucket

Comman git commands:

\*git add <file name> -add file local area to staging area

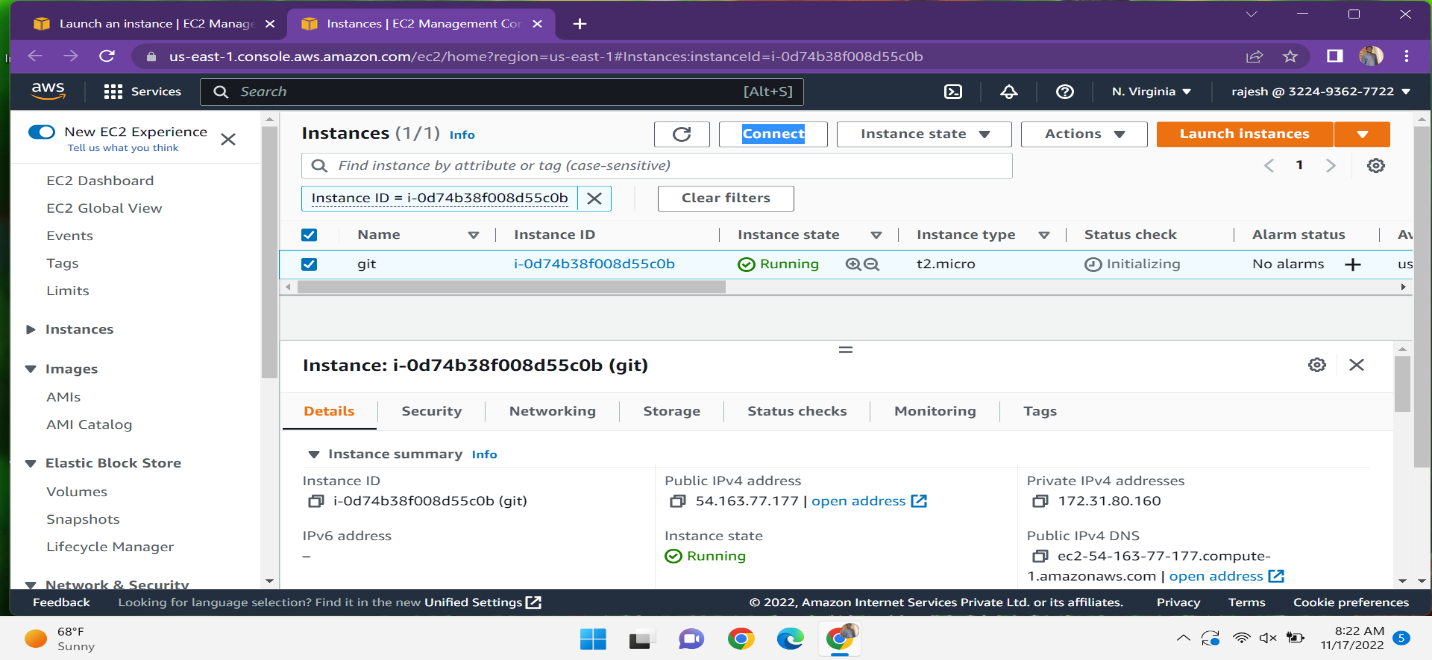
\*git commit -m “message” <file name> - file from staging area to remote repo

\*git push – add file to remote repo to central repository

PROJECT-1

SCENARIO-1: Creating ec2 instance

Login into your aws console and create a ec2 instance linux as a operating system and connect the server to terminal.



Instance connect to terminal

Text

Description automatically generated

Scenario:2 Create repo in local machine

\* Create a folder on local machine using command-git init folder

\* After we can switch to that folder using command -cd folder

\* In that create an empty file using touch command-touch file1.txt

\* Then after we can check the status whether if any changes are done but the status shows that

the file is present but it is untracked by git-git status

\*The file can be add to working area to staging area then the git can tracked the file-

Git add file1.txt

\*After we commit the changes add staging area to local repository the command –

Git commit -m “file” file1.txt

\*Check the status-git statusText

Description automatically generated

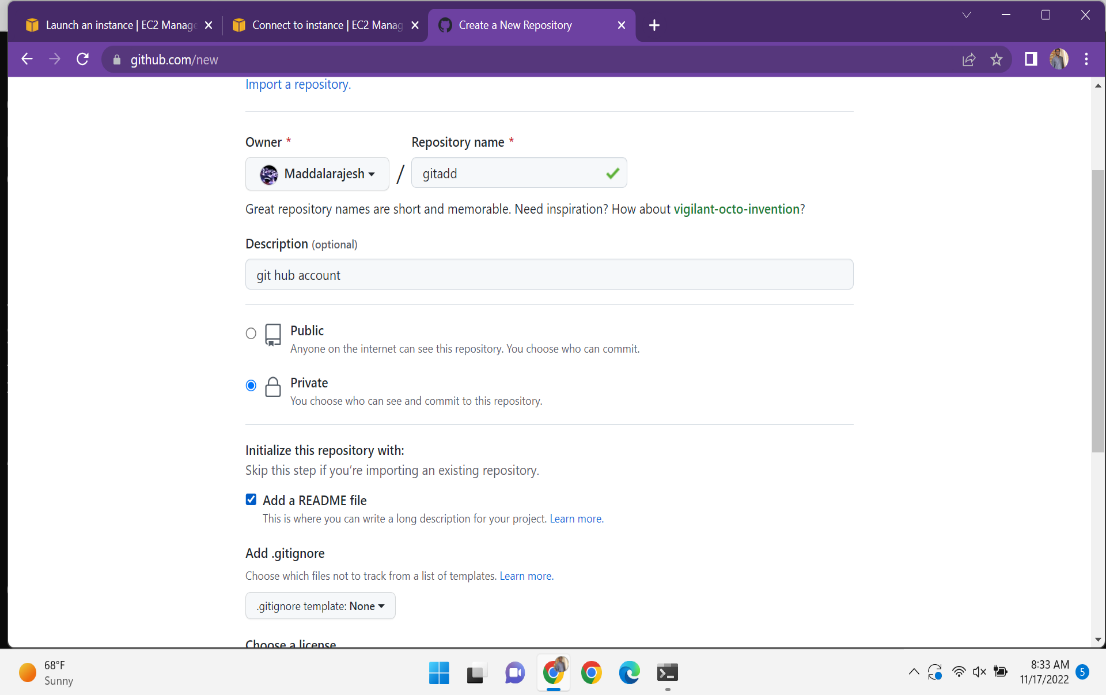
Text

Description automatically generated

Scenario:3 Creating repo in remote location-github

\*Go to the github account create a new repository then click on new give the repo name-gitadd after

then we have to initialize the README.md file and then finally click on the create repo.



Graphical user interface, text, application, email

Description automatically generated

Scenario:4 Working with remote repo

\* Copy the code and paste the code on local machine

Git clone(paste the URL that will copy from github)

\*it will be ask the username and password the passwork cannot work instead of password create

Personal access token(PAC).

\*Personal access token creation-go to our profile at last settings is there click on it and go to the

Developer settings in that we see a PAC choose the classic token and generate a new token.

\*Create a files by using command-touch file2.txt

\*After that the file can add from local to staging area-git add file2.txt

\*And from add staging area to repo-git commit -m “file” file2.txt

\*Git push it asks the user name and the password it moves from local to central repo

Graphical user interface, text, application, chat or text message, email

Description automatically generated

Scenario:5 Creating a new branch from main branch

Go to the github repo click on the main it shows the option create new branch and then

new branch will created after that you can change some modifications in the branch

and see that modifications will only on branch it cant be see in main.

Graphical user interface, text, application, email

Description automatically generated

Scenario:6 Pushing a locally created repo into github

Previously create a repo in your local machine and initialize it locally in lab-2 . with the same name create a repo in remote repo and it cannot be initialize . and the commands in local machine it is a master while in the remote repo it be a main-git branch -M main

Git remote add origin -paste the URL of your remote repo

To push your local branch to remote repo-git push -u origin main

Text

Description automatically generated

Graphical user interface, text, application, chat or text message

Description automatically generated

Scenario:7 pull all the branches in your local machine

Go to your local machine where you have the copy of your remote branch. Run the command git pull to pull all the changes such as branches from the remote location run git branch -a and then checkout the feature-git checkout(branch name) and see the status -git status and make some changes in the file and see the status.

Git add filename

Git commit -m “hello” file3.txt and git push it asks the user name and the password.and after got our github repo see if any changes are in main .

Graphical user interface, text, application

Description automatically generated

Scenario:8 merge our feature branch with main branch

Go to our git hub repository check the changes in your branch click on the create and pull request put your main branch and the feature branch in the respective block . click on the create pull request and it will ask for a comment,just click again on create pull request and merge it go to our code in mai branch and see the changes are now visible here.

A screenshot of a computer

Description automatically generated

Scenario:9 Go to local machine

Go to our local machine switch to the main-git checkout main

and run the command git pull it asks the permissions give it and see the changes are available in main.

Text

Description automatically generated