

# Assignment for Cloud Engineer

1. Find a publicly available sample frontend application (e.g., Next.js or Vue.js), host it on an AWS EC2 instance (Linux OS), and set up a reverse proxy using a web server (Nginx or Apache). No backend is needed for this task.
2. Create a README file in GIT repository detailing the steps you took to complete this task with screen shots.

Note – You can submit whatever progress you have made at the end of 3 days from the date of receiving this assignment. Completing this assignment increases your chances of getting hired.

## # Assignments1

- Copy git repository

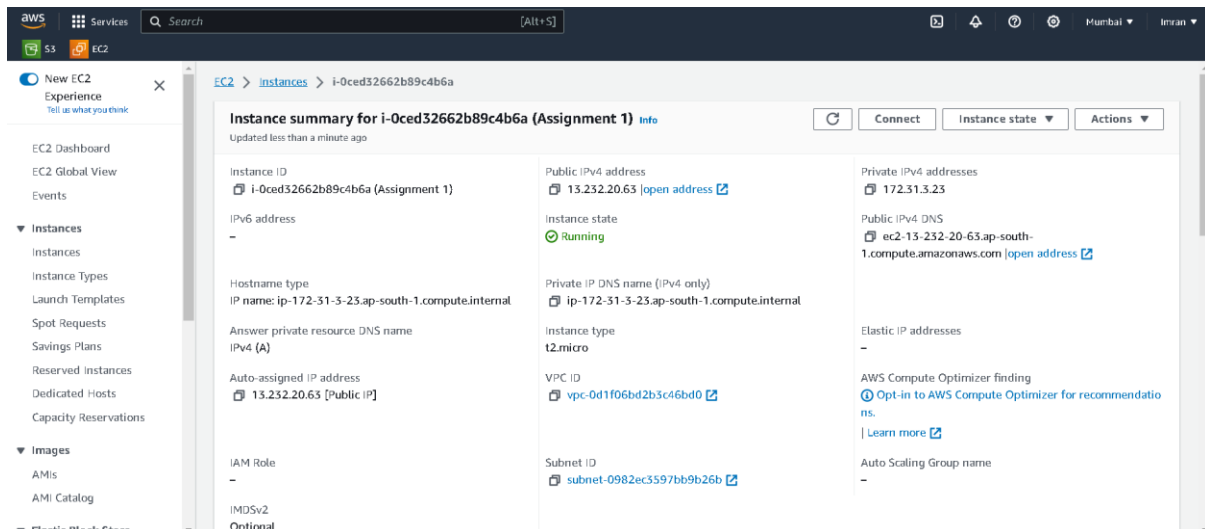
The screenshot shows the GitHub interface for a repository named 'assignment-4' by user 'Imran9198'. The repository is public and has 1 branch (main) and 0 tags. The commit history shows 5 commits by user 'warengonzaga' on August 14, 2022. The commit messages are: 'CHORE: update info', 'CHORE: add initial info', 'CHORE: add license', and 'CHORE: update info'. The files listed are: .github, pages, public, styles, .eslintrc.json, .gitignore, LICENSE, README.md, next.config.js, package.json, and yarn.lock. The commit messages for the files are: 'Initial commit from Create Next App' for .github, pages, public, styles, .eslintrc.json, .gitignore, next.config.js, package.json, and yarn.lock; and 'CHORE: update info' for LICENSE and README.md.

File	Commit Message	Commit Date	Commit Count
.github	Initial commit from Create Next App	last year	1
pages	Initial commit from Create Next App	last year	1
public	Initial commit from Create Next App	last year	1
styles	Initial commit from Create Next App	last year	1
.eslintrc.json	Initial commit from Create Next App	last year	1
.gitignore	Initial commit from Create Next App	last year	1
LICENSE	CHORE: add license	last year	1
README.md	CHORE: update info	last year	1
next.config.js	Initial commit from Create Next App	last year	1
package.json	Initial commit from Create Next App	last year	1
yarn.lock	Initial commit from Create Next App	last year	1

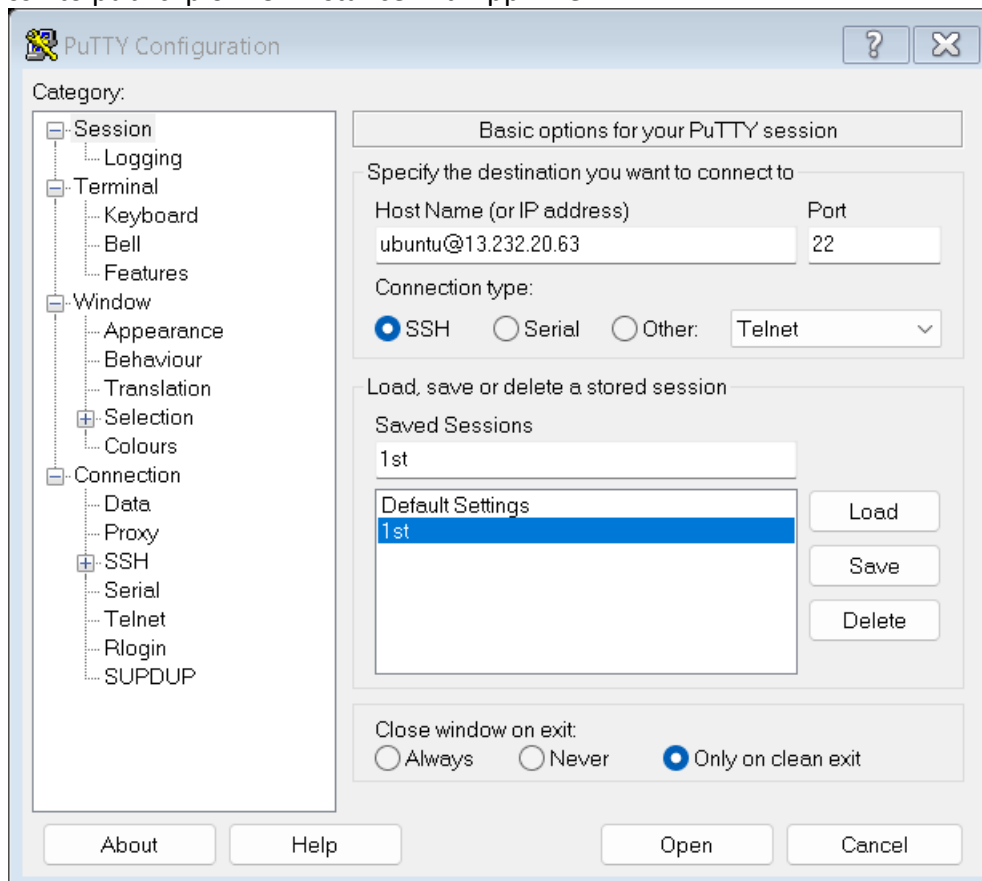
- Create EC2 instance on aws portal
- select Ubuntu 22.04 LTS OS ,t2 micro,with default vpc
- create .ppk key , create security group with all traffic allow inbound rule.

The screenshot shows the AWS Management Console interface. The left sidebar contains navigation links for 'New EC2 Experience', 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Instances', 'Instance Types', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', 'Dedicated Hosts', 'Capacity Reservations', 'Images', 'AMIs', 'AMI Catalog', and 'Elastic Block Store'. The main content area displays the 'Instances (1)' page. A table lists the instances, showing one instance named 'next.js' with Instance ID 'i-0ced52662b89c4b6a', Instance state 'Running', Instance type 't2.micro', Status check '2/2 checks passed', Alarm status 'No alarms', Availability Zone 'ap-south-1b', and Public IPv4 DNS 'ec2-52-66-176-...'. Below the table, there is a 'Select an instance' button.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
next.js	i-0ced52662b89c4b6a	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1b	ec2-52-66-176-...



- ssh to public ip of EC2 instance with .ppk file



- `sudo apt update` (to ensure your instance has the latest updates.)
- `curl -sL https://deb.nodesource.com/setup_16.x | sudo -E bash -` (copy file from web site)
- `sudo apt-get install -y nodejs` (install nodejs)
- `node -v` (check nodejs version)
- `git clone https://github.com/Imran9198/assignment-4.git` (Clone your Nextjs App from GitHub)
- `cd sample-nextjs-app/` (change directory application)
- `npm install` (Install required node modules)
- `npm run build` (to make application ready)

- `sudo npm install pm2 -g` (Install PM2 using the below command)
- `pm2` (to run application)

```
Last login: Sun Oct 8 11:13:08 2023 from 103.199.136.133
ubuntu@ip-172-31-3-23:~$ sudo su
root@ip-172-31-3-23:/home/ubuntu# pm2

-----
      /\      /\      /\      /\      /\      /\      /\      /\      /\      /\
     /\      /\      /\      /\      /\      /\      /\      /\      /\      /\
    /\      /\      /\      /\      /\      /\      /\      /\      /\      /\
   /\      /\      /\      /\      /\      /\      /\      /\      /\      /\
  /\      /\      /\      /\      /\      /\      /\      /\      /\      /\
 /\      /\      /\      /\      /\      /\      /\      /\      /\      /\
/\      /\      /\      /\      /\      /\      /\      /\      /\      /\
-----

Runtime Edition

PM2 is a Production Process Manager for Node.js applications
with a built-in Load Balancer.

Start and Daemonize any application:
$ pm2 start app.js

Load Balance 4 instances of api.js:
$ pm2 start api.js -i 4

Monitor in production:
$ pm2 monitor

Make pm2 auto-boot at server restart:
$ pm2 startup

To go further checkout:
http://pm2.io/

-----

usage: pm2 [options] <command>

pm2 -h, --help          all available commands and options
pm2 examples            display pm2 usage examples
pm2 <command> -h        help on a specific command

Access pm2 files in ~/.pm2
root@ip-172-31-3-23:/home/ubuntu#
```

- `pm2 start npm --name nextjs-app -- run start -- -p 3000` (Execute the below code to run Next.js with PM2 (**security group should allow HTTP on port 3000**))
- copy instance public ip and add port no. 3000

<http://13.232.20.63:3000/>



# Welcome to Sample **Next.js** App!

🇺🇸❤️🇮🇳 by Warren Gonzaga