

CS423 DevOps

Assignment 3

Abdur Rahman Goraya, 2022035

Ubaid ur Rehman, 2022599

About the assignment and project:

The project was cloned from <https://github.com/eljamaki01/ReactNodeTesting>, it contained a React + Express app that was a simple shopping cart application. Very basic, but contained a front-end and back-end part with tests written for both.

The assignment object was to setup CI/CD deployment of a given app. The app had tests written and used ESLint. These provided code structure and guidelines.

We set up the GitHub workflows, two separate ones for testing and staging. These workflows triggered with relevant triggers and updated their deployed code in their respective servers.

Both of these workflows also contained the integral CI part where we ran jest and eslint to ensure our code quality is up to standard.

They also included notification via emails.

Difficulties we faced:

1. After installing node via nvm, libatomic1 was missing for some reason from the default install of the linux OS. We thought I was an issue with node but the fix was as simple as

```
sudo apt install libatomic1
```

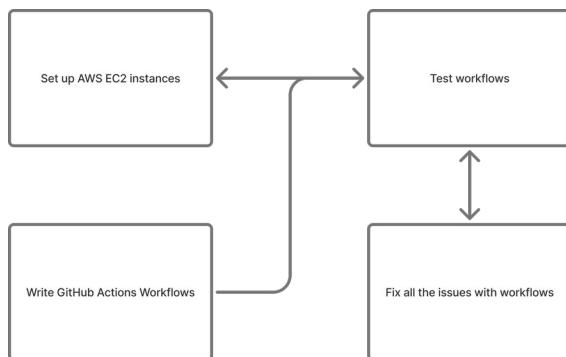
2. We both couldn't remember the mail trapping service that Sir used in class and that took some time to remember (we asked chatgpt because 10 minutes of googling didn't work, thanks AI). We used ethereal mail (mailtrap.io was not working)

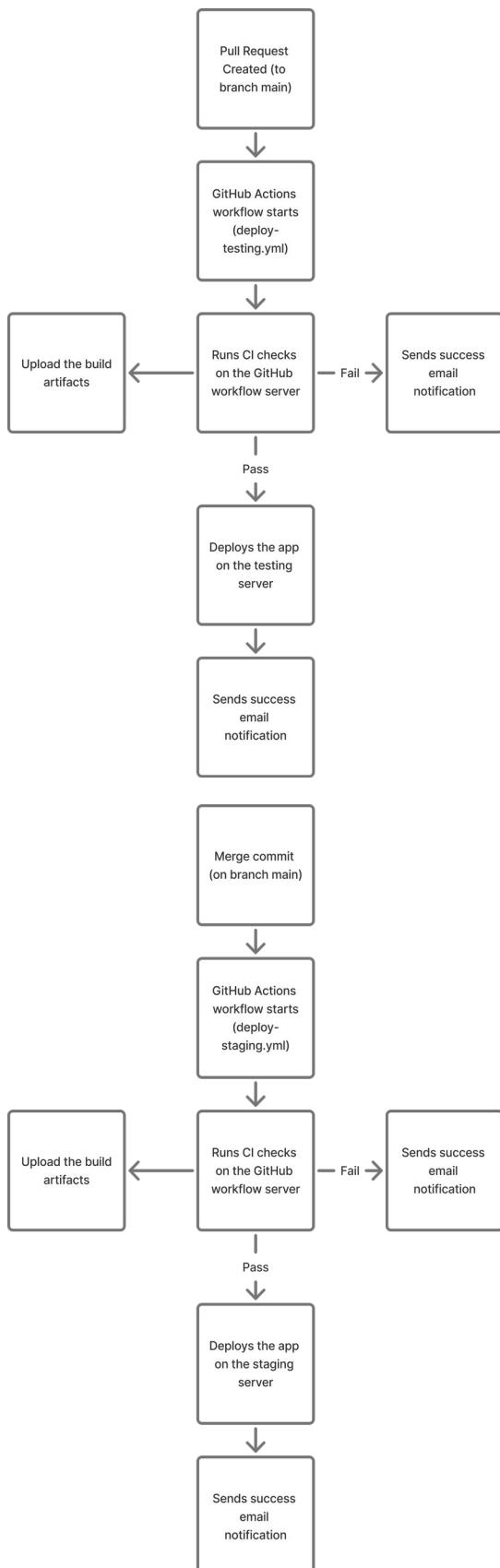
3. When using the actions script appleboy/ssh-action@v1.0.3, it took us 30 minutes to figure out that the script does not source the .bashrc located in the user folder. We had to insert this at the start of it so it could find npm and pm2

```
source ~/.bashrc || source ~/.profile || true  
[ -s "$HOME/.nvm/nvm.sh" ] && source "$HOME/.nvm/nvm.sh" || true
```

Flows:

Not sure if we are asked to present what we did, or what is the CI/CD process of the application after the assignment but here's both.





Screenshots:

1. Creating the instances

The screenshot shows two side-by-side AWS interface panels. On the left, the 'Launch an instance' wizard is displayed, allowing users to create a new EC2 instance. It includes sections for 'Name and tags', 'Application and OS Images (Amazon Machine Image)', and 'Amazon Machine Image (AMI)'. The AMI selection section lists various operating systems like Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, and Debian. A specific AMI entry for 'Ubuntu Server 24.04 LTS (HVM), SSD Volume Type' is selected, showing its details and being marked as 'Free tier eligible'. On the right, the 'Summary' section of the instance creation wizard is shown, detailing the chosen configuration: 1 instance, Canonical Ubuntu 24.04, t2.micro instance type, and a new security group. Below this, the 'Instances (2)' list is shown, displaying two running servers: 'Devops Testing Server' and 'Devops Staging Server'.

2. Writing and pushing the workflows

This screenshot displays a series of GitHub commit history entries, likely from a CI/CD pipeline. The commits are as follows:

- Merge pull request #1 from flankedgonerogue/dev
- Fix SSH script to load user environment for npm/pm2
- Update workflows to use npm install instead of npm ci
- Update package-lock.json for reproducible CI builds
- Add package-lock.json for reproducible CI builds
- Updated README.md
- Add CI/CD workflow for staging environment deployment
- Add CI/CD workflow for testing environment deployment

Each commit is accompanied by a green 'Verified' badge, indicating successful CI validation. The commits were made by the user 'flankedgonerogue' at various times, with some being just a few minutes ago and others up to 2 hours ago.

3. Storing the secrets

Repository secrets		New repository secret
Name	Last updated	
🔒 APP_DIRECTORY	31 minutes ago	 
🔒 AWS_SSH_PRIVATE_KEY	1 hour ago	 
🔒 AWS_SSH_USERNAME	1 hour ago	 
🔒 QA_EMAIL	1 hour ago	 
🔒 SMTP_FROM_EMAIL	54 minutes ago	 
🔒 SMTP_PASSWORD	1 hour ago	 
🔒 SMTP_PORT	1 hour ago	 
🔒 SMTP_SERVER	1 hour ago	 
🔒 SMTP_USERNAME	1 hour ago	 
🔒 STAGING_SERVER_IP	8 minutes ago	 
🔒 TESTING_SERVER_IP	9 minutes ago	 

4. Setting up the instances

```
ubuntu@ip-172-31-25-24:~/app$ curl -o https://raw.githubusercontent.com/nvm-sh/nvm/v0.40.3/install.sh | bash
% Total    % Received % Xferd  Average Speed   Time     Time      Time  Current
                                         Dload  Upload Total Spent   Left  Speed
100 16631  100 16631    0     0   307k    0  --::--:-- --::--:-- --::--:--  312k
=> Downloading nvm from git to '/home/ubuntu/.nvm'
=> Cloning into '/home/ubuntu/.nvm'...
remote: Enumerating objects: 384, done.
remote: Counting objects: 100% (384/384), done.
remote: Compressing objects: 100% (328/328), done.
remote: Total 384 (delta 42), reused 181 (delta 28), pack-reused 0 (from 0)
Receiving objects: 100% (384/384), 392.50 KiB | 9.81 MiB/s, done.
Resolving deltas: 100% (42/42), done.
* (HEAD detached at FETCH_HEAD)
  master
=> Compressing and cleaning up git repository

=> Appending nvm source string to /home/ubuntu/.bashrc
=> Appending bash_completion source string to /home/ubuntu/.bashrc
=> Close and reopen your terminal to start using nvm or run the following to use it now:

export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh" # This loads nvm
[ -s "$NVM_DIR/bash_completion" ] && \. "$NVM_DIR/bash_completion" # This loads nvm bash_completion
ubuntu@ip-172-31-25-24:~/app$ bash
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-25-24:~/app$ nvm install node
Downloading and installing node v25.2.1...
Downloading https://nodejs.org/dist/v25.2.1/node-v25.2.1-linux-x64.tar.xz...
#####
Computing checksum with sha256sum
Checksums matched!
Now using node v25.2.1
Creating default alias: default -> node (> v25.2.1)
```

```
ubuntu@ip-172-31-25-24:~$ git clone https://github.com/flankedgonerogue/devops-assignment.git ./app
Cloning into './app'...
remote: Enumerating objects: 154, done.
remote: Counting objects: 100% (154/154), done.
remote: Compressing objects: 100% (85/85), done.
remote: Total 154 (delta 68), reused 150 (delta 64), pack-reused 0 (from 0)
Receiving objects: 100% (154/154), 805.53 KiB | 12.20 MiB/s, done.
Resolving deltas: 100% (68/68), done.
ubuntu@ip-172-31-25-24:~$ cd app
```

```
ubuntu@ip-172-31-25-24:~/app$ npm install -g pm2  
  
added 133 packages in 7s  
  
13 packages are looking for funding  
  run `npm fund` for details
```

5. Testing the workflows

```

  ✓ Deploy to Testing Server
  4s

1 ► Run appleboy/ssh-action@v1.0.3
21 /usr/bin/docker run --name a4cd386e354cd575144b987966cc04c8e14c4_f32d66 --label a4cd38 --workdir /github/workspace --rm -e "INPUT_HOST" -e "INPUT_USERNAME" -e "INPUT_KEY" -e "INPUT_SCRIPT" -e "INPUT_PORT" -e "INPUT_PASSPHRASE" -e "INPUT_PASSWORD" -e "INPUT_SYNC" -e "INPUT_USE_INSECURE_CIPHER" -e "INPUT_CIPHER" -e "INPUT_TIMEOUT" -e "INPUT_COMMAND_TIMEOUT" -e "INPUT_KEY_PATH" -e "INPUT_FINGERPRINT" -e "INPUT_PROXY_HOST" -e "INPUT_PROXY_PORT" -e "INPUT_PROXY_USERNAME" -e "INPUT_PROXY_PASSWORD" -e "INPUT_PROXY_CIPHER" -e "INPUT_PROXY_USE_INSECURE_CIPHER" -e "INPUT_SCRIPT_STOP" -e "INPUT_ENVS" -e "INPUT_ENVS_FORMAT" -e "INPUT_DEBUG" -e "INPUT_ALLENVS" -e "INPUT_REQUEST_PTY" -e "HOME" -e "GITHUB_JOB" -e "GITHUB_REF" -e "GITHUB_SHA" -e "GITHUB_REPOSITORY" -e "GITHUB_REPOSITORY_OWNER" -e "GITHUB_RUN_ID" -e "GITHUB_RUN_NUMBER" -e "GITHUB_RUN_NUMBER"
22 ======CND=====
23 cd ***
24 git fetch origin
25 git checkout dev
26 git pull origin dev
27 npm install
28 npm run build-react
29 pm2 restart all || pm2 start index.js --name "react-node-app"
30 ======END=====
31 err: Already on 'dev'.
32 out: Your branch is up to date with 'origin/dev'.
33 err: From https://github.com/flankedgonerogue/devops-assignment
34 err:   * branch          dev      -> FETCH_HEAD
35 err:   * branch          dev      -> FETCH_HEAD
36 out: Already up to date.
37 err: bash: line 5: npm: command not found
38 err: bash: line 6: npm: command not found
39 err: bash: line 7: pm2: command not found
40 err: bash: line 7: pm2: command not found
41 2024/12/21 16:07:23 Process exited with status 127

```

6. Ta-da apps running

