

Day 25 Task: Complete Jenkins CI/CD Project - Continued with Documentation

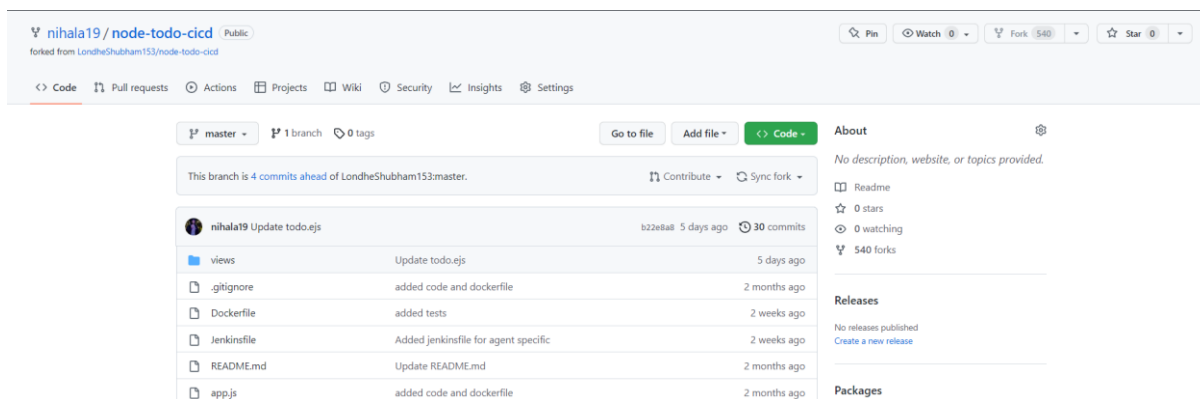
I can imagine catching up will be tough so take a small breather today and complete the Jenkins CI/CD project from Day 24 and add a documentation.

Task-01:

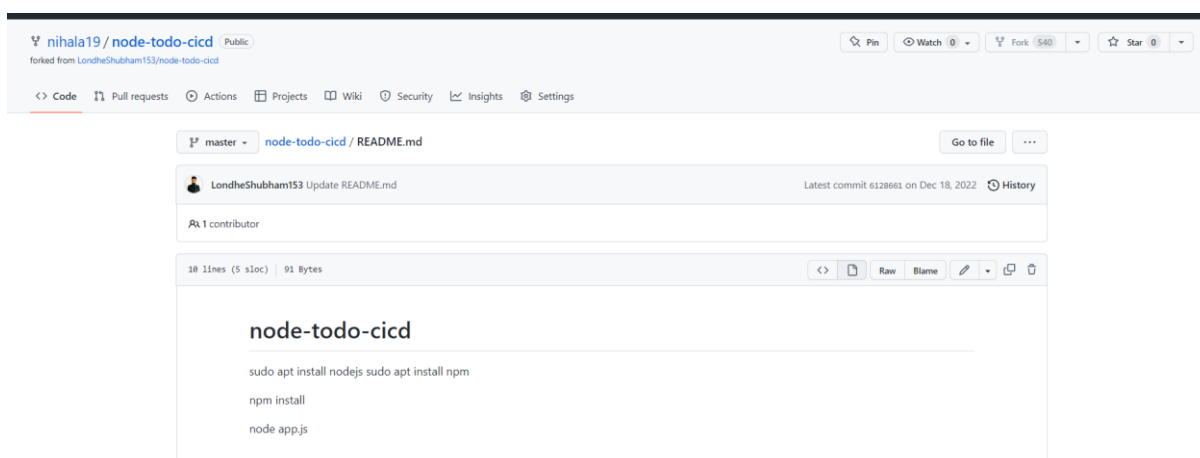
Document the process from cloning the repository to adding webhooks, and Deployment, etc. as a README, go through [this example](#)

A well written readme file will help others to understand your project and you will understand how to use the project again without any problems.

- Go to your GitHub account.
- And click on node-todo-cicd.



- Click on the **README.md** file and,
- Write on the processes in the **README.md** file that have done their day 24 tasks.



```

10
11 Step1: Fork the repository.
12 Step2: Generate the SSH keys for integrating your Jenkins project with your git repository. Use ssh-keygen command to create public and private key.
13 Step3: Now, go to your GitHub account settings.
14 Step4: Go to SSH and GPG keys, Add public key that we created using ssh-keygen and select key-type Authentication key.
15 Step5: Go to your GitHub repository and click on Settings.
16 Step6: Click on Webhooks and then click on Add webhook.
17 Step7: In the 'Payload URL' field, paste your Jenkins environment URL. At the end of this URL add /github-webhook/. In the 'Content type' select: 'application/json' and leave the 'Secret' field empty.
18 Step8: Now for Installing GitHub Integration plugin in Jenkins
19 Step9: Open your Jenkins dashboard.
20 Step10: Click on the Manage Jenkins button on your Jenkins dashboard
21 Step11: Click on Manage Plugins
22 Step12: Install GitHub Integration plugin
23 Step13: Configuring Jenkins:
24 Step14: Create a Jenkins job
25 Step15: Create node-todo-app freestyle project
26 Step16: In Configure, GitHub project URL write your project GitHub URL
27 Step17: In Git, add credentials for Jenkins
28 Step18: Click on the 'Build Triggers' tab and then on the 'GitHub hook trigger for GITScm polling'.
29 Step19: In the Execute shell run the application using Docker compose
30 Step20: You will have to make a Docker Compose file for this Project
31 Step21: After build you can check console output.
32 Step22: Using docker ps command, you can see container is created.
33 Step23: Browse public IP address with port no.8000
34

```

Attach files by dragging & dropping, selecting or pasting them.



Commit changes

nihala19 / node-todo-cicd
Public

forked from LondheShubham153/node-todo-cicd
Pin
Watch 0
Fork 542
Star 0

[Code](#)
[Pull requests](#)
[Actions](#)
[Projects](#)
[Wiki](#)
[Security](#)
[Insights](#)
[Settings](#)

master
node-todo-cicd / README.md
Go to file
...

nihala19 Update README.md
Latest commit 73d126d now
History

2 contributors

33 lines (28 sloc) | 1.61 KB
<>
Raw
Blame
Edit
Copy

node-todo-cicd

```

sudo apt install nodejs sudo apt install npm

npm install

node app.js

Step1: Fork the repository. Step2: Generate the SSH keys for integrating your Jenkins project with your git repository. Use ssh-keygen command to create public and private key. Step3: Now, go to your GitHub account settings. Step4: Go to SSH and GPG keys, Add public key that we created using ssh-keygen and select key-type Authentication key. Step5: Go to your GitHub repository and click on Settings. Step6: Click on Webhooks and then click on Add webhook. Step7: In the 'Payload URL' field, paste your Jenkins environment URL. At the end of this URL add /github-webhook/. In the 'Content type' select: 'application/json' and leave the 'Secret' field empty. Step8: Now for Installing GitHub Integration plugin in Jenkins Step9: Open your Jenkins dashboard. Step10: Click on the Manage Jenkins button on your Jenkins dashboard Step11: Click on Manage Plugins Step12: Install GitHub Integration plugin Step13: Configuring Jenkins: Step14: Create a Jenkins job Step15: Create node-todo-

```

Task-02:

Also it's important to keep smaller goals, as its a small task, think of a small Goal you can accomplish.

Write about it using [this template](#)

Have small goals and strategies to achieve them, also have a small reward for yourself.

February Goal Planner

My Goal

Gaining hands-on experience with automation tools and processes.

My Strategies

Building my own personal projects to gain practical experience and build my skills.

Note to self

Stay focused on continuous learning and improvement.

Reward

Opportunities for career advancement.

Thanks for reading!

Happy Learning 😊