Using Crontab & Bash Scripts to do Automated Git Push Daily

Hillary Wando
.
Follow
3 min read
.
Mar 3

The Problem

So in this scenario, we don’t have a DevOps pipeline setup. However, we make constant changes to th code on the server such that the web app we are working on always has the latest. This is such that w can develop web apps quite quickly and have them deployed almost as soon as the development process starts. The problem we currently have is the lack of code backups since there is no use of Git and GitHub in our development process. Hence the need to setup Git on the server and make use of t Crontab service and a bash script to commit code and push it to GitHub at a given time every day. Thi such that we will always have version control and a backup of the server code on GitHub.

As a prerequisite before we begin, we need to have setup SSH push access to our repository on GitHu so that our bash script does not keep asking us for our username and password.

Solution Step 1

The first step is to come up with the bash script for handling the git commits and pushing to GitHub. T bash script is presented below.

The script gets the current date and uses it to create a commit message. It then changes the current directory to the folder where we have our code. In this case we are using a dummy folder so that will l the ‘gitBashScript’ folder. Inside this folder, the scripts runs the ‘git add’ and ‘git commit’ commands while appending the generated commit message to the ‘git commit’ command. The script then gets th output of the

```
git status -b --porcelain
```

command and checks it against a given string. This check is to make sure we don’t do a git push when there are no new commits in the working tree. If the working tree is clean the script simply prints the text ‘IT IS CLEAN’ to a text file, otherwise it goes ahead and commits our code to GitHub.

Solution Step 2

The next step is to make our script executable and then add it to the Linux crontab list so that it can b executed automatically. To make the script executable we can run the following command while in the script’s parent directory.

```
chmod +x script.sh
```

After this we can also use the following commands to copy the script into a directory that our system expects to contain executable scripts and code.

```
sudo cp git-push.sh /usr/bin/git-push.sh
sudo cp git-push.sh /usr/local/bin
```

Before we add our script to the crontab list we can run the following command to make sure the cron service is running on Linux.

```
sudo service cron status
```

This should give you the following output if the service is up and running.

```
 * cron is running
```

If the service is not running you can start it up using the following command

```
sudo service cron start
```

Finally, we can add our script to the crontab list using the following command.

```
crontab -e
```

This should open up the crontab list for you to edit. If the command gives you back a list of editors instead just select the first one on the list by entering 1 to edit the crontab list using the Nano editor. Once we have access to the crontab list we can enter the following line into it to add our script.

```
0 2 * * * bash ~/gitBashScript/script.sh
```

This should set it to run the script at 2 am every day. And with that we should now have everything setup to backup our server code on GitHub every day.

Conclusion

This concludes this small tutorial on how to setup automated daily git push to GitHub from the server. an additional step for tracking purposes, you can also add email notifications to your GitHub repositor such that an email is sent to your email address every time the script pushes to the repo.

Git

Github

Bash Script

Crontab

--

Follow

Written by Hillary Wando

15 Followers

More from Hillary Wando

Hillary Wando

How to Create a Basic CRUD API using Python Flask

CRUD is a common acronym used in software development. It stand for the 4 services provided by a REST API, tha is: Create, Read, Update...

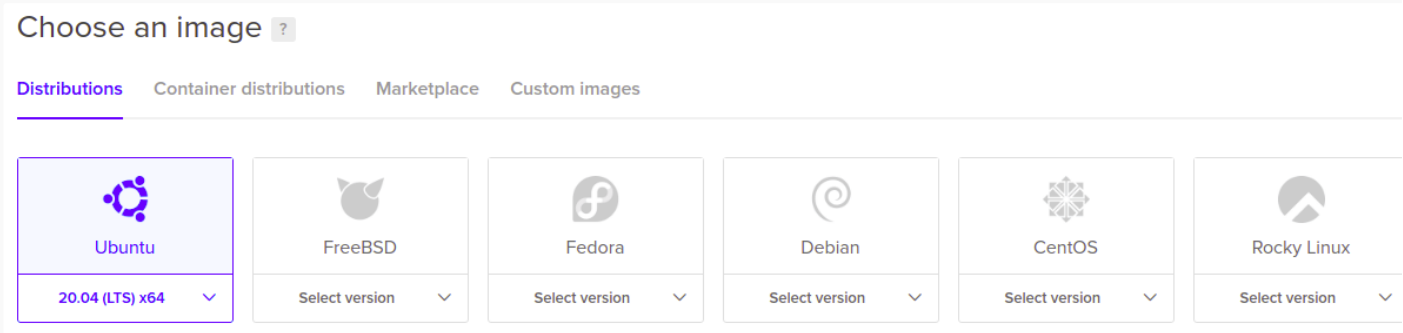
5 min read

·

Aug 25, 2022

--

2



Hillary Wando

Hosting a FullStack Javascript Web App on Digital Ocean Part 1

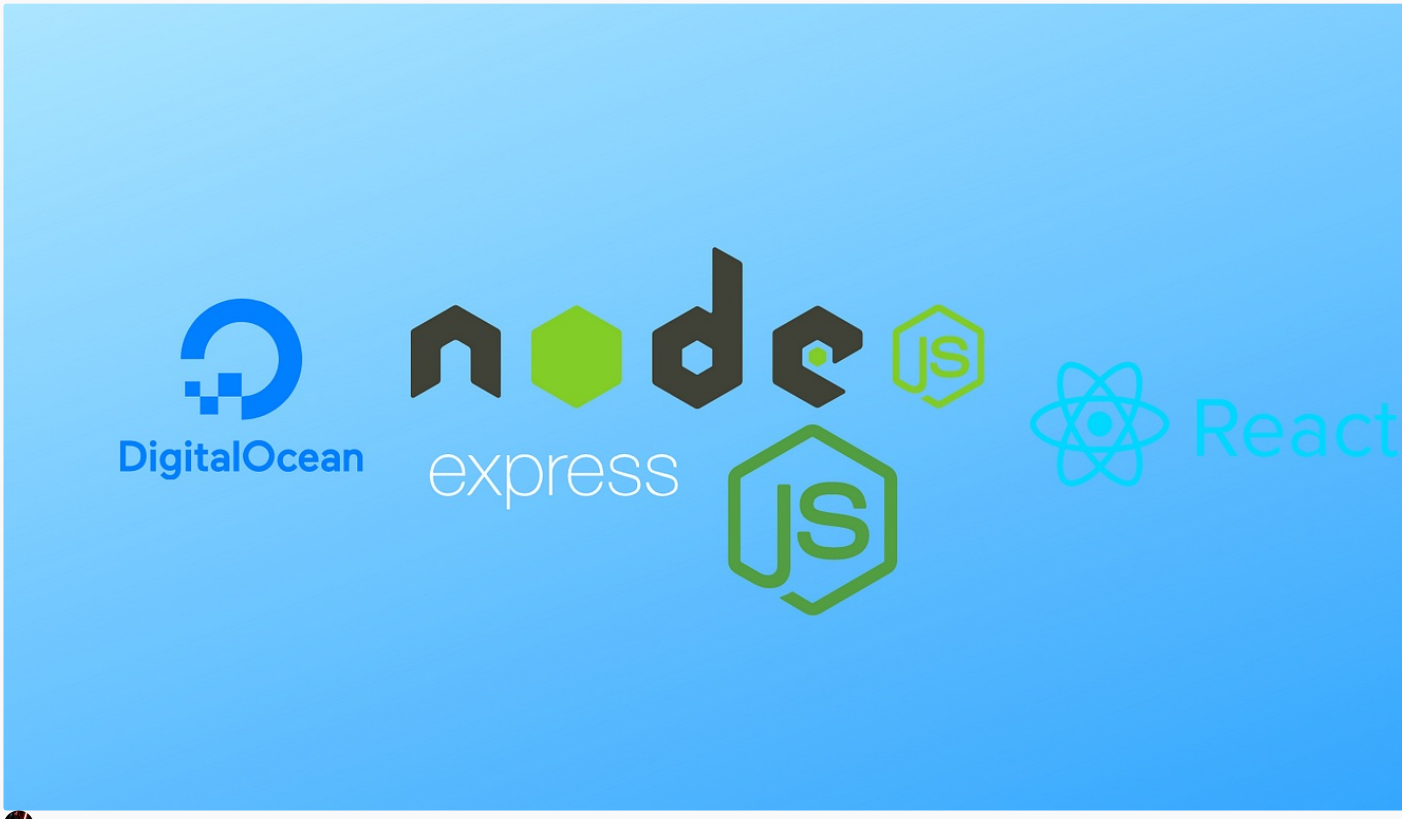
This is the first of a three part series showing how to host a fullstack MERN app, both the Node JS back-end and React front-end on one...

5 min read

·

Mar 10, 2022

--



Hillary Wando

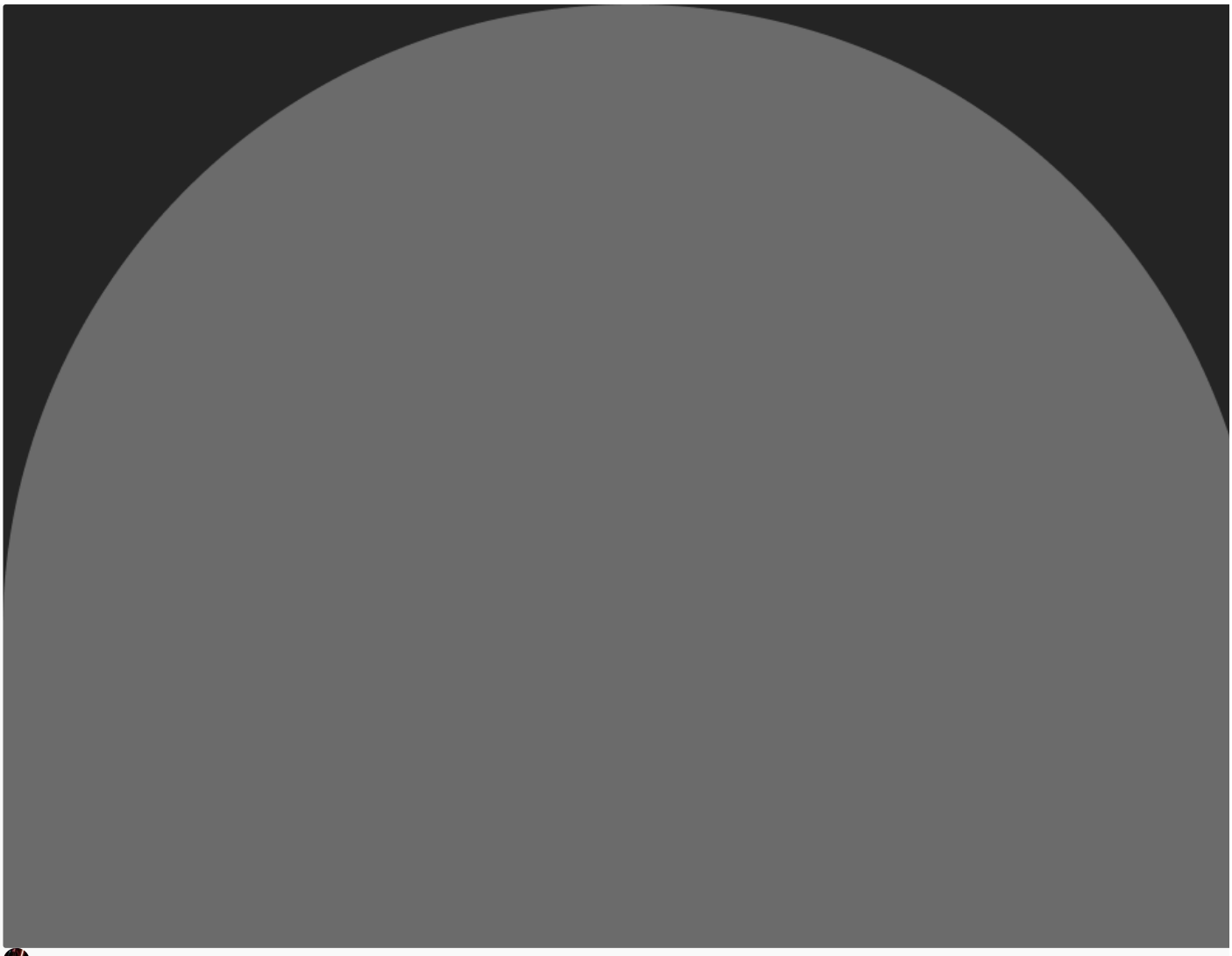
Hosting a Full Stack Javascript Web App on Digital Ocean

This is a guide on how to host a fullstack javascript web app on a single Digital Ocean Droplet. The stack we are working with is...

1 min read

Mar 10, 2022

--



Hillary Wando

Hosting a FullStack Javascript Web App on Digital Ocean Part 3

This is part 3 of a guide on how to host a fullstack Javascript web app (NodeJS/ExpressJS back-end, ReactJS front-er
on a single Digital...

4 min read

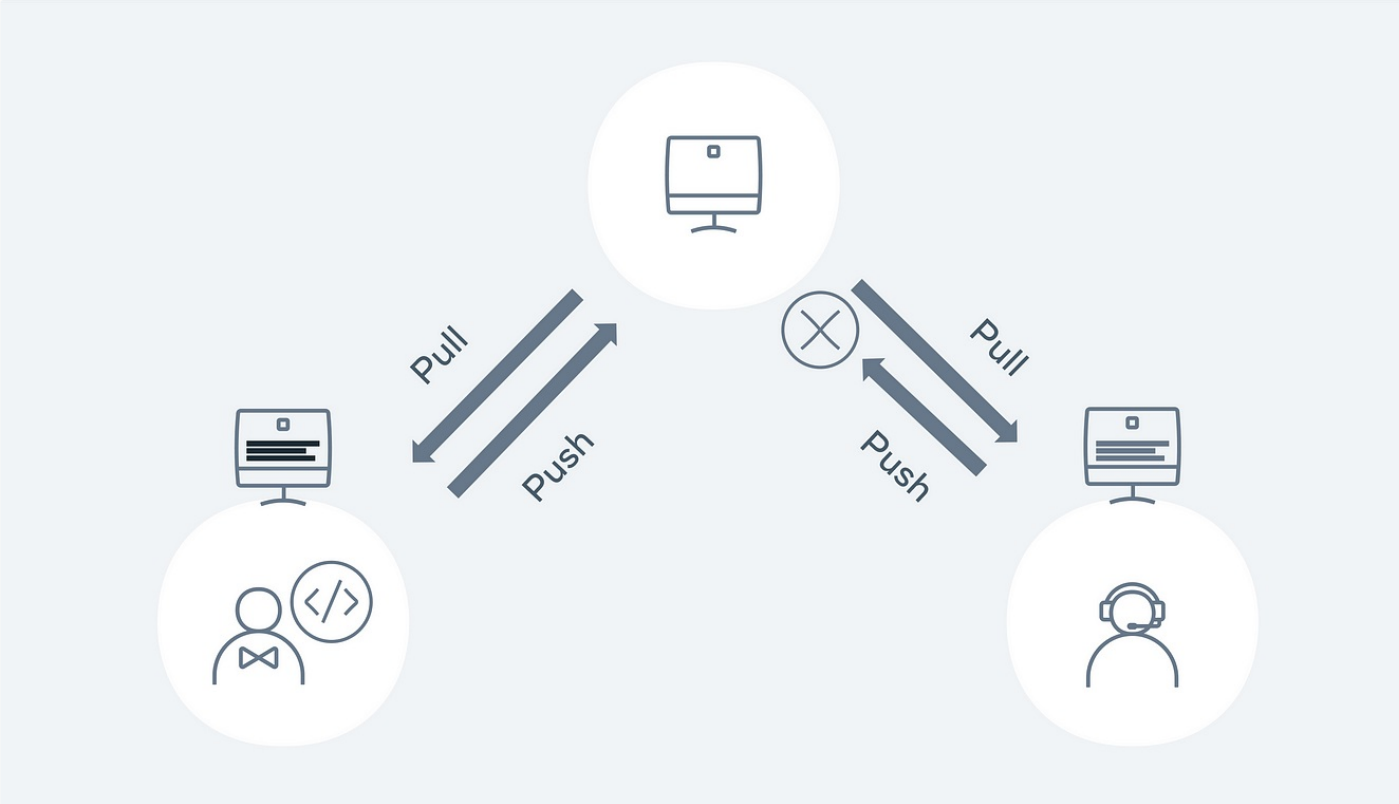
·

Aug 5, 2022

--

See all from Hillary Wando

Recommended from Medium



Suhas Tum...

Resolving conflicts in Git

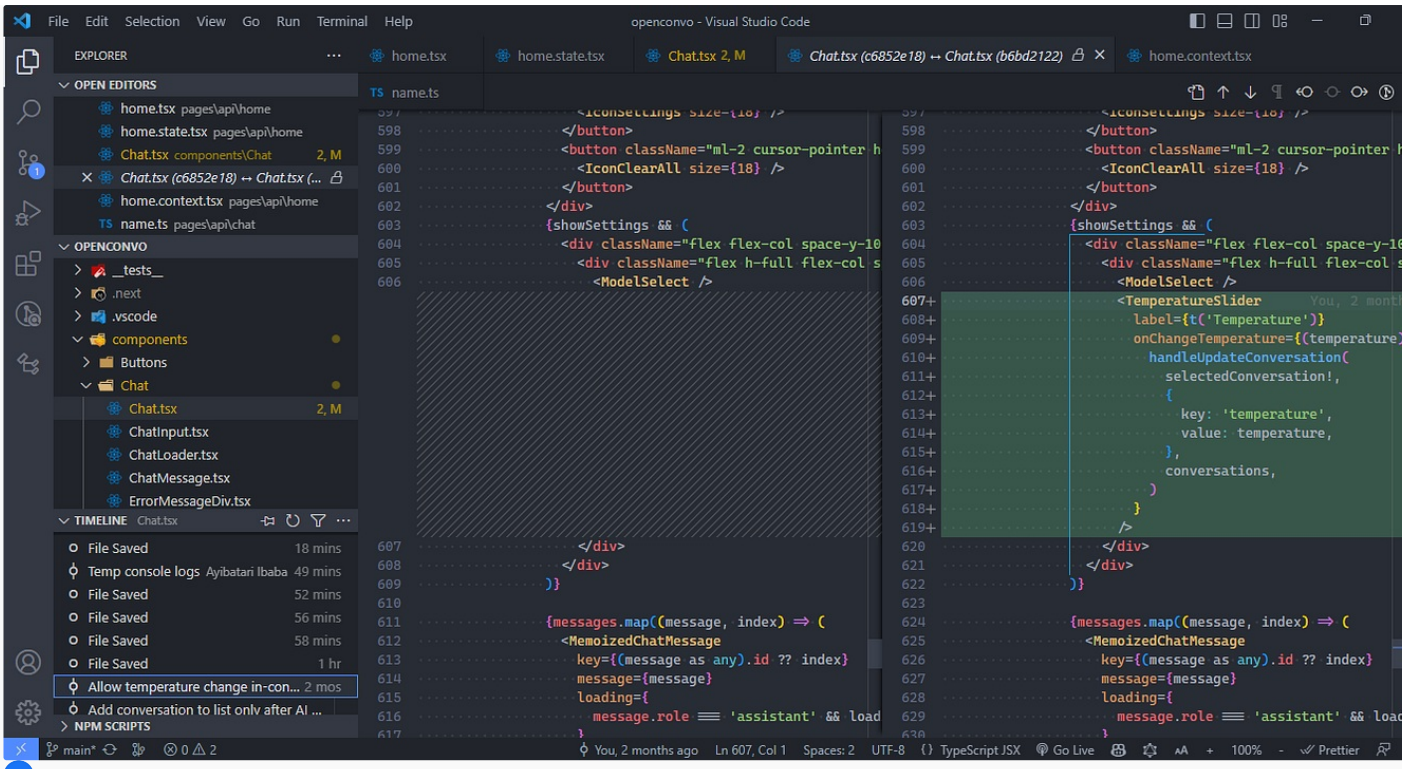
You know that conflicts will normally occur when you try to merge a branch that may have competing changes. Git will normally try to...

3 min read

·

Apr 3

--



Coding Beauty

in

Dev Genius

10 essential VS Code tips & tricks for greater productivity

Boost your productivity with VS Code: discover key features to enhance your coding experience and achieve your goals faster than ever.

10 min read

.

Aug 20

--

19

Lists

Icon Design

30 stories

.

84 saves

Generative AI Recommended Reading

52 stories

·

231 saves

New_Reading_List

174 stories

·

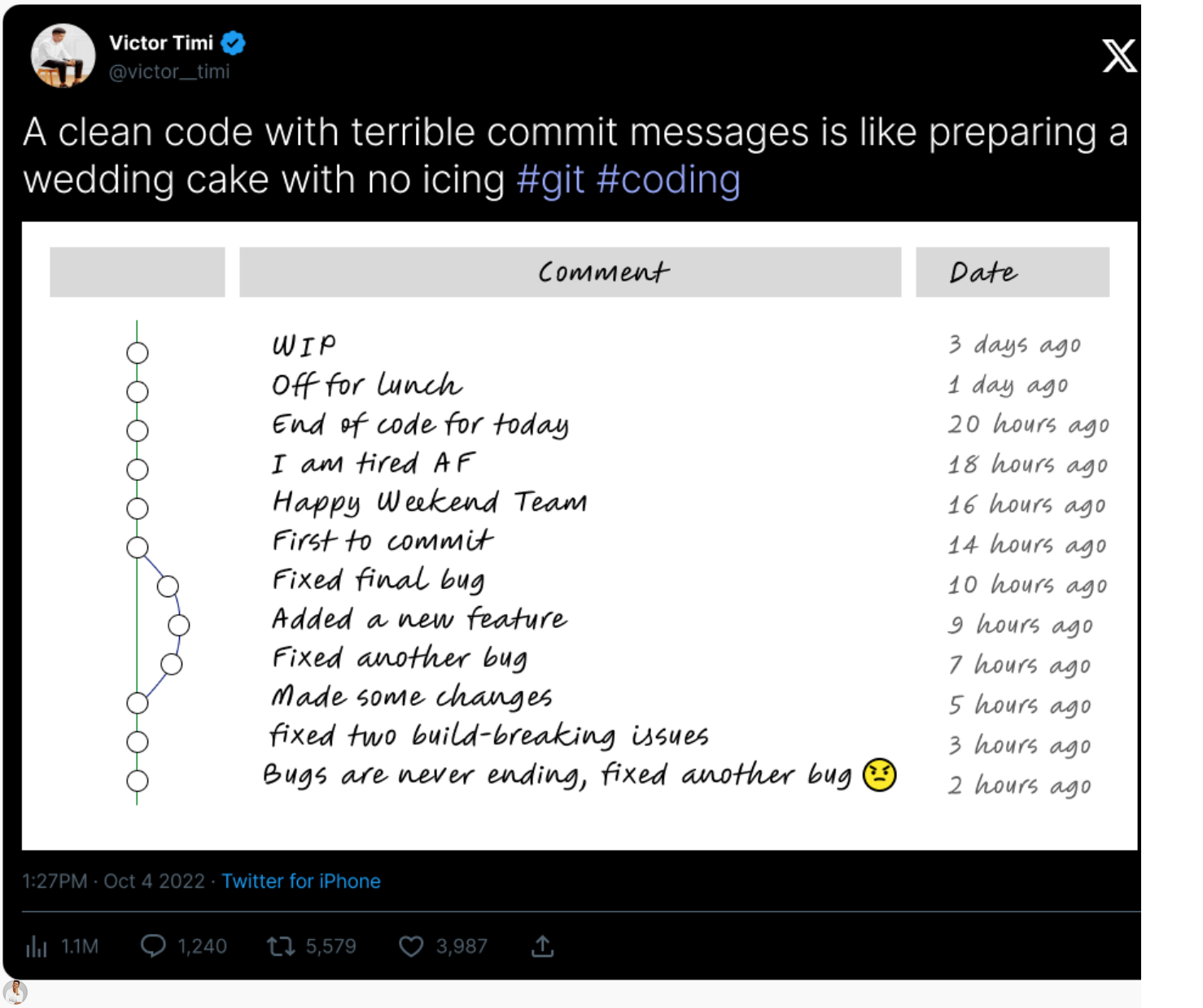
105 saves

Natural Language Processing

607 stories

·

220 saves



Victor Timi
in
Level Up Coding

“Good Commit” vs “Your Commit”: How to Write a Perfect Git Commit Message

A good commit shows whether a developer is a good collaborator — Peter Hutterer, Linux.

8 min read
·
Sep 5

--

24

Harsh Rajotya

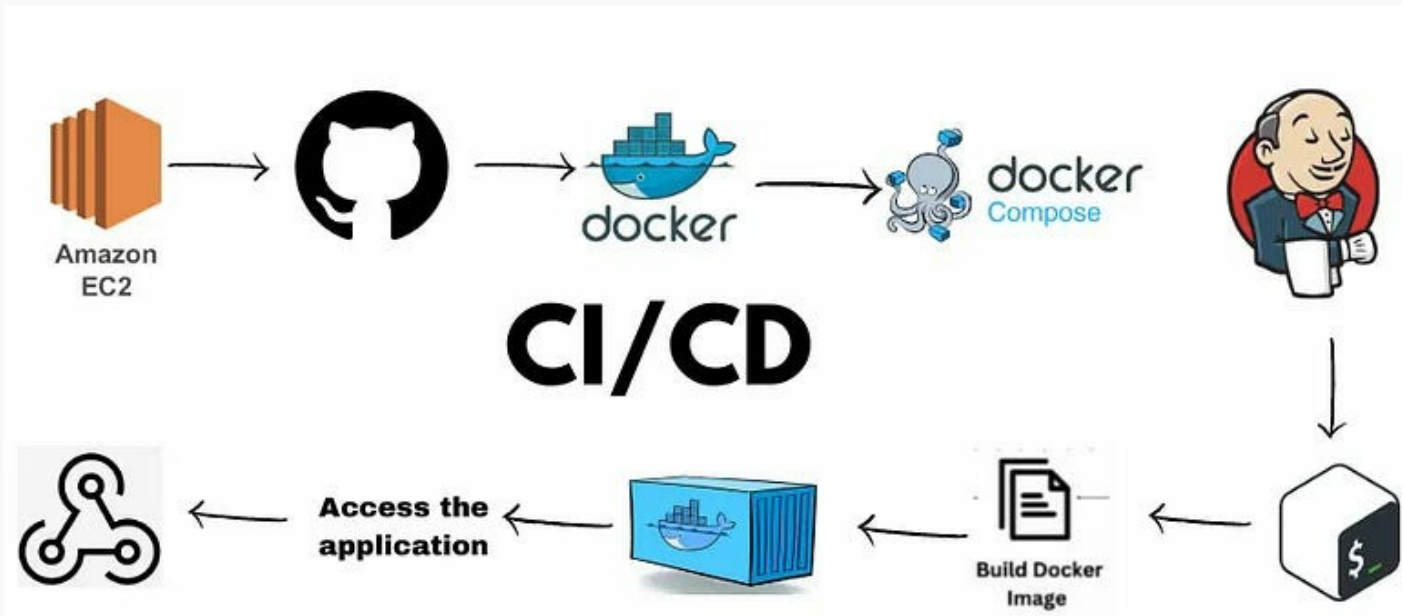
Advanced Linux Shell Scripting for DevOps Engineers with User management #Da 5

TABLE OF CONTENTS

3 min read
·
Jul 19

--

2



jabir ahammed

Complete Jenkins CI/CD Project

6 min read

Jul 27

--

```
→  gpush git:(demo) gpush
[?] What type of commit is this?: feat
   fix
 > feat
   docs
   ci

[?] Does the commit include breaking changes?: Yes
 > Yes
   No

[?] What's your commit message: Example commit message
```

tjtharrison

Git conventional commits

Here’s a description of conventional commits by Google’s Bard, in the style of a Slipknot song:

4 min read

·

Jun 19

--