

CONTINUOUS DEPLOYMENT OF A NODE.JS APPLICATION WITH GITLAB CI AND DOCKER AUTOMATION.

Check Branch Protection Rules in GitLab:

- Go to your project in GitLab.
- Navigate to **Settings > Repository**.
- Under **Protected Branches**, you should see the main branch listed. GitLab typically restricts direct pushes and force pushes to protected branches.
- You can either:
 - **Modify the protection settings** to allow force pushes, or

1. Check Existing Remotes:

```
git remote -v
```

2. Remove Existing Remote origin:

```
git remote remove origin
```

3. Re-add the Remote origin:

```
git remote add origin https://gitlab.com/divyam4111748/demo.git
```

4. Rename Current Branch to main:

```
git branch -M main
```

5. Push the Local Changes to main on GitLab:

```
git push -uf origin main
```

```
C:\Windows\System32\cmd.e  x  +  v

(c) Microsoft Corporation. All rights reserved.

C:\Users\ibmtr\Desktop\VTU DevOps\Week 10\NodeJS App_DevOps>git remote -v
origin  https://gitlab.com/divyam4111748/demo.git (fetch)
origin  https://gitlab.com/divyam4111748/demo.git (push)

C:\Users\ibmtr\Desktop\VTU DevOps\Week 10\NodeJS App_DevOps>git remote remove origin
warning: more than one branch.main.remote
warning: branch.main.remote has multiple values
warning: branch.main.merge has multiple values

C:\Users\ibmtr\Desktop\VTU DevOps\Week 10\NodeJS App_DevOps>git remote add origin https://gitlab.com/divyam4111748/demo.git

C:\Users\ibmtr\Desktop\VTU DevOps\Week 10\NodeJS App_DevOps>git branch -M main

C:\Users\ibmtr\Desktop\VTU DevOps\Week 10\NodeJS App_DevOps>git push -uf origin main
Enumerating objects: 61, done.
Counting objects: 100% (61/61), done.
Delta compression using up to 12 threads
Compressing objects: 100% (33/33), done.
Writing objects: 100% (61/61), 33.09 KiB | 11.03 MiB/s, done.
Total 61 (delta 25), reused 53 (delta 22), pack-reused 0
To https://gitlab.com/divyam4111748/demo.git
+ 01da493...729c079 main -> main (forced update)
warning: branch.main.remote has multiple values
warning: branch.main.merge has multiple values
branch 'main' set up to track 'origin/main'.

C:\Users\ibmtr\Desktop\VTU DevOps\Week 10\NodeJS App_DevOps>
```

Dockerfile:

FROM node:latest

WORKDIR /usr/src/app

COPY package.json ./

RUN npm install

COPY . .

EXPOSE 4000

CMD ["node", "index.js"]

.gitlab-ci.yml file:

image: docker:latest

services:

- docker:dind

stages:

- build

- test

- release

- deploy

variables:

CONTAINER_TEST_IMAGE: \$CI_REGISTRY_IMAGE:\$CI_COMMIT_REF_SLUG

CONTAINER_RELEASE_IMAGE: \$CI_REGISTRY_IMAGE:latest

DOCKER_HOST: tcp://docker:2375

DOCKER_DRIVER: overlay2

DOCKER_TLS_CERTDIR: ""

before_script:

- docker info

- docker login -u \$CI_REGISTRY_USER -p \$CI_REGISTRY_PASSWORD
\$CI_REGISTRY

build:

stage: build

script:

- docker build -t \$CONTAINER_TEST_IMAGE .

- docker push \$CONTAINER_TEST_IMAGE

test:

stage: test

script:

- docker pull \$CONTAINER_TEST_IMAGE

- timeout 300s docker run --rm \$CONTAINER_TEST_IMAGE npm test

cache:

paths:

- node_modules/

release-image:

stage: release

script:

- docker pull \$CONTAINER_TEST_IMAGE
- docker tag \$CONTAINER_TEST_IMAGE \$CONTAINER_RELEASE_IMAGE
- docker push \$CONTAINER_RELEASE_IMAGE

deploy:


stage: deploy



script:


- docker pull \$CONTAINER_RELEASE_IMAGE
- docker run -d -p 3000:3000 \$CONTAINER_RELEASE_IMAGE

main ▾demo / + ▾

HistoryFind fileEdit ▾Code ▾

 Update file .gitlab-ci.yml
Divya Murugan authored 2 minutes ago



 cea1b414 

Name	Last commit	Last update
test	Initial commit	18 minutes ago
.dockerignore	Initial commit	18 minutes ago
 .gitlab-ci.yml	Update file .gitlab-ci.yml	2 minutes ago
README.md	Initial commit	22 minutes ago
Dockerfile	Update file dockerfile	4 minutes ago
index.js	Initial commit	18 minutes ago
package-lock.json	Initial commit	18 minutes ago
package.json	Initial commit	18 minutes ago

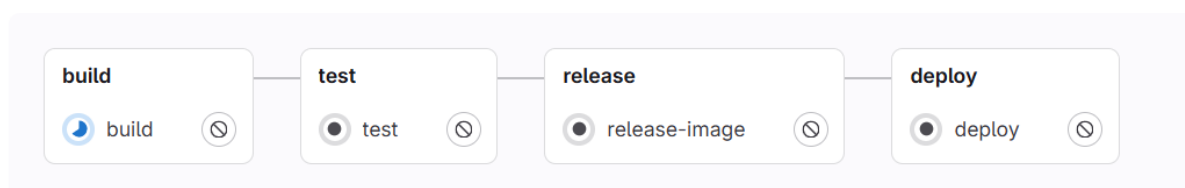
Update file .gitlab-ci.yml

 **Running** Divya Murugan created pipeline for commit cea1b414  just now

For **main**

latest  4 jobs  In progress, queued for 1 seconds

Pipeline Jobs 4 Tests 0



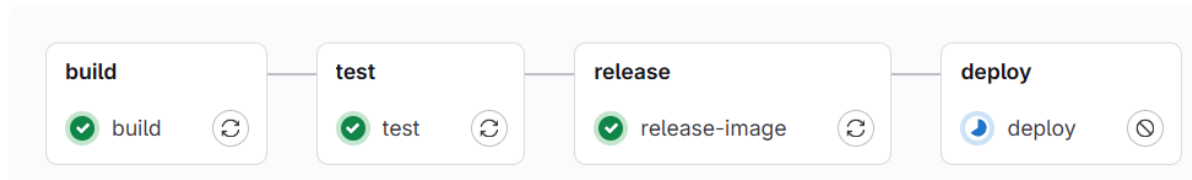
Update file .gitlab-ci.yml

✓ Passed **santhanam L** created pipeline for commit `895e356a` 4 minutes ago, finished just now

For `main`

latest 4 jobs 4.5 4 minutes 30 seconds, queued for 1 seconds

Pipeline Jobs 4 Tests 0



Pipeline Jobs 4 Tests 0



1. Check All Containers (including stopped ones)

To see all containers, including those that are stopped, use:

```
docker ps -a
```

This will list all containers, and you can check if the container is stopped or if there were any errors during the deployment process.

1. Check Logs of the Stopped Container

To understand why the container stopped, you can view the logs of the container. Use the following command:

```
docker logs a4c54a198495
```

This will display the logs of the container, which should help identify any errors or issues that caused it to stop.

2. Start the Container Manually

If you want to start the container again manually, use the following command:

```
docker start -a a4c54a198495
```

<input type="checkbox"/>	<input type="radio"/>	minikube	ed6e9a56f7d4	k8s-minikube/kicbase	http://localhost:3000	0%	6 days ago			
<input type="checkbox"/>	<input checked="" type="radio"/>	amazing_wescoff	a4c54a198495	my-nodejs-app	3000:3000	0%	15 seconds ago			

amazing_wescoff

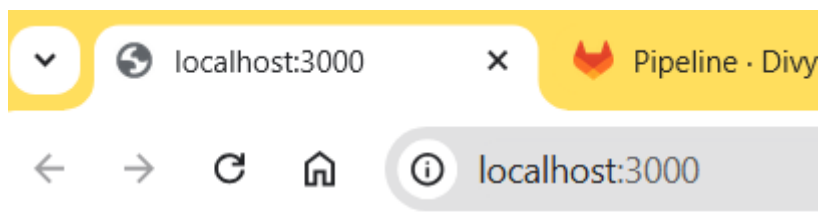
a4c54a198495

my-nodejs-app

3000:3000

Show all ports (5)

0% 9 min



```
{ "response": "Hello From Divya" }
```