# **Pipeline For Node app**

Last updated by | shakibesaib | Aug 15, 2023 at 1:01 PM GMT+6

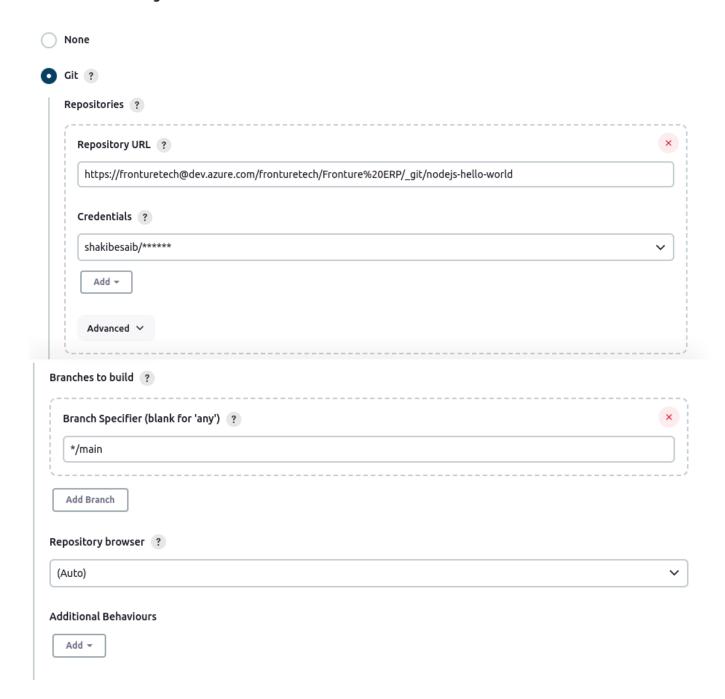
## List of tasks to be done to build a node app and deploy it on local machine-

- In a repository keep the node app, make sure Node modules has been ignore by git.
- · Create a item
- Add repository to jenkins which has the node app
- Trigger build so that jenkins can pull any changes if made in source code.
- Add Execute Shell in build steps
- Give necessary commands to build and deploy the app.

## Steps-

1. Adding our nodejs app repository <a href="https://dev.azure.com/fronturetech/Fronture ERP/\_git/nodejs-hello-world">https://dev.azure.com/fronturetech/Fronture ERP/\_git/nodejs-hello-world</a> in Source Code Management.

## Source Code Management



2.

## **Build Triggers**

	Trigger builds remotely (e.g., from scripts) ?
	Build after other projects are built ?
	Build periodically ?
	GitHub hook trigger for GITScm polling ?
<b>~</b>	Poll SCM ?
	Schedule ?
	****
	Do you really mean "every minute" when you say "* * * * * *"? Perhaps you meant "H * * * * *" to poll once per hour Would last have run at Wednesday, August 9, 2023 at 12:46:06 PM Bangladesh Standard Time; would next run at Wednesday, August 9, 2023 at 12:46:06 PM Bangladesh Standard Time.
	Ignore post-commit hooks ?

3.

## **Build Steps**

```
Execute shell ?

Command

See the list of available environment variables

Tm -rf *.tar.gz
npm install
tar czf node_app.tar.gz node_modules app.js package.json
chmod +r /var/lib/jenkins/workspace/node_app/node_app.tar.gz
cp /var/lib/jenkins/workspace/node_app/node_app.tar.gz /home/devops/Desktop/
cd /home/devops/Desktop
tar -xzvf node_app.tar.gz
npm install

BUILD_ID=dontKillMe nohup npm start &

curl -I http://localhost:3003/
```

Save and select build now.



```
Started by an SCM change
    Running as SYSTEM
    Building in workspace /var/lib/jenkins/workspace/node app
    The recommended git tool is: NONE
    using credential 86fc3e93-a491-484f-81a4-424dfaab6238
     > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/node_app/.git # timeout=10
    Fetching changes from the remote Git repository
     > git config remote.origin.url https://fronturetech@dev.azure.com/fronturetech/Fronture%20ERP/ git/nodejs-hello-
    world # timeout=10
    Fetching upstream changes from https://fronturetech@dev.azure.com/fronturetech/Fronture%20ERP/ git/nodejs-hello-
     > git --version # timeout=10
     > git --version # 'git version 2.34.1'
    using GIT_ASKPASS to set credentials
    > git fetch --tags --force --progress --
    https://fronturetech@dev.azure.com/fronturetech/Fronture%20ERP/ git/nodejs-hello-world
    +refs/heads/*:refs/remotes/origin/* # timeout=10
     > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
    Checking out Revision ab6262c956ddbc73d550164d957693483ccbbb70 (refs/remotes/origin/main)
found 0 vulnerabilities
+ curl -I http://localhost:3003/
+ BUILD ID=dontKillMe nohup npm start
  % Total % Received % Xferd Average Speed
                                                Time
                                                         Time
                                                                  Time Current
                                 Dload Upload Total
                                                         Spent
                                                                  Left Speed
        0
                                            0 --:--:--
      15
             Θ
                              0
                                     0
                                            0 --:--:--
HTTP/1.1 200 OK
X-Powered-By: Express
Content-Type: text/html; charset=utf-8
Content-Length: 15
ETag: W/"f-RtfnH1PrzrfzTp4m10YZ0t+EWE0"
Date: Wed, 09 Aug 2023 06:25:29 GMT
Connection: keep-alive
Keep-Alive: timeout=5
> dev node@1.0.0 start
> node app.js
Archiving artifacts
Finished: SUCCESS
```

#### Code:

```
rm -rf *.tar.gz
npm install
tar czf node_app.tar.gz node_modules app.js package.json
chmod +r /var/lib/jenkins/workspace/node_app/node_app.tar.gz
cp /var/lib/jenkins/workspace/node_app/node_app.tar.gz /home/devops/Desktop/
cd /home/devops/Desktop
tar -xzvf node_app.tar.gz
npm install
BUILD_ID=dontKillMe nohup npm start &
curl -I http://localhost:3003/
```

#### **Explanation Of the code:**

- rm -rf \*.tar.gz , this will remove the tar.gz file of previous build.
- npm install, to run node app we need to do npm install so it will download all the dependencies
- tar czf node\_app.tar.gz node\_modules app.js package.json , zipping the files to deploy.
- chmod +r /var/lib/jenkins/workspace/node\_app/node\_app.tar.gz , giving write permission to the to copy the file.
- cp /var/lib/jenkins/workspace/node\_app/node\_app.tar.gz /home/devops/Desktop/, copying the zip file to another directory of the local machine. Here we have to keep in mind that the folder we are copying to, jenkins have enough permission to that folder.
- cd /home/devops/Desktop, directing terminal to the directory where we will deploy the app.
- tar -xzvf node app.tar.gz , extracting the zip files.
- npm install, to download the dependencies on that directory
- BUILD\_ID=dontKillMe nohup npm start & , starting npm and making npm run on the back stage, while jenkins do other tasks.
- curl -I <a href="http://localhost:3003/">http://localhost:3003/</a>, using curl checking the status code, if the app deployment is okay.