SOP on CI/CD of TL-Meet-Develop & TL-Meet-Develop-Post-build-Docker

Table of Contents

Contents

Document Control

Version History

Version Detail	Date	Prepared By	Reviewed By
V1	17/10/2023	Priya Apotikar	Anand Jain & Ganesh Jadhao
-	-	-	-

TL-Meet-Develop

Aim:

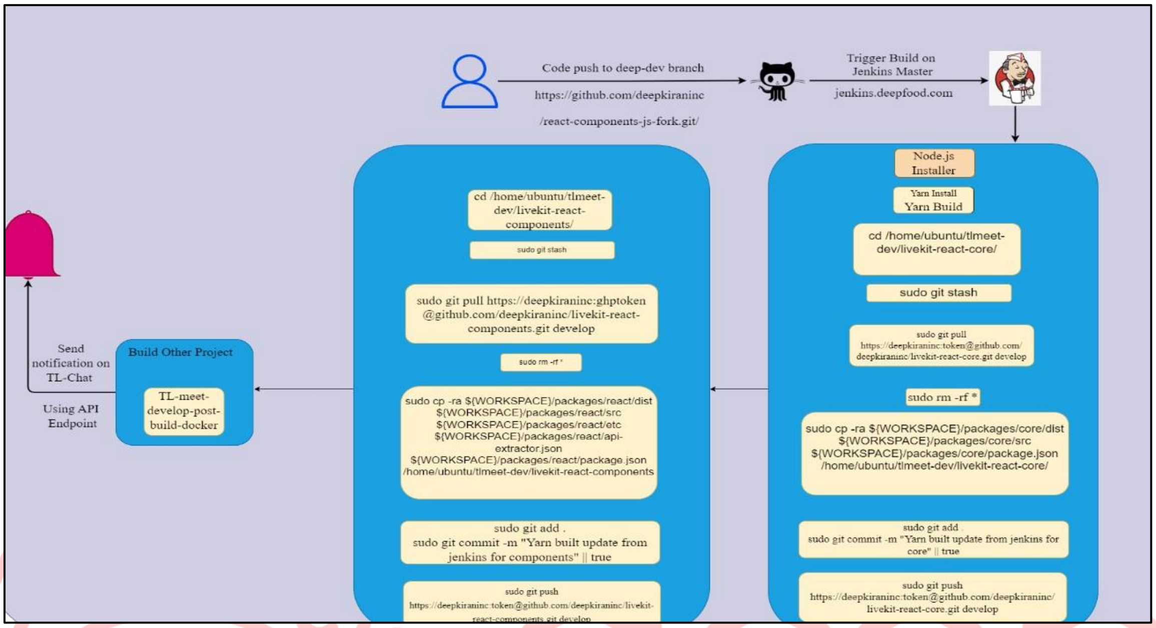

Code push to deep-dev branch
https://github.com/deepkiraninc
/react-components-js-fork.git/
Trigger Build on
Jenkins Master
jenkins.deepfood.com
Node.js
Installer
Yarn Install
Yarn Build
cd /home/ubuntu/tlmeet-dev/livekit-react-core/
sudo git stash
sudo git pull
https://deepkiraninc:token@github.com/deepkiraninc/livekit-react-core.git develop
sudo rm -rf *
sudo cp -ra ${WORKSPACE}/packages/core/dist ${WORKSPACE}/packages/core/src ${WORKSPACE}/packages/core/package.json /home/ubuntu/tlmeet-dev/livekit-react-core/
sudo git add .
sudo git commit -m "Yarn built update from jenkins for core" || true
sudo git push
https://deepkiraninc:token@github.com/deepkiraninc/livekit-react-core.git develop
cd /home/ubuntu/tlmeet-dev/livekit-react-components/
sudo git stash
sudo git pull https://deepkiraninc:ghptoken @github.com/deepkiraninc/livekit-react-components.git develop
sudo rm -rf *
sudo cp -ra ${WORKSPACE}/packages/react/dist ${WORKSPACE}/packages/react/src ${WORKSPACE}/packages/react/etc ${WORKSPACE}/packages/react/api-extractor.json ${WORKSPACE}/packages/react/package.json /home/ubuntu/tlmeet-dev/livekit-react-components
sudo git add .
sudo git commit -m "Yarn built update from jenkins for components" || true
sudo git push
https://deepkiraninc:token@github.com/deepkiraninc/livekit-react-components.git develop
Build Other Project
TL-meet-develop-post-build-docker
Send notification on TL-Chat
Using API Endpoint

Figure 1: Architecture of TL-Meet-Develop

Prerequisite:

- NodeJs 18.16.0
- Git URL

Navigate to Jenkins

jenkins.deepfoods.com

Steps to create pipeline for TL-Meet-Develop

Step 1: Create a Job

On the Jenkins dashboard, click on 'New Item' at the left top side of the dashboard.

Step 2: Choose Freestyle Project

Enter a name under 'Enter an item name'.i.e, TL-Meet-Develop

Then select 'Freestyle project' and click on 'OK'.

Step 3: Configure the Project

We can set the project name, description, and other settings.

Project: **Teamlocus**

Pipeline Author’s: Priya Apotikar & Anand Jain

Responsible Developer's: **Apurv Bhavsar**

Step 4: Add Project URL:

https://github.com/deepkiraninc/react-components-js-fork.git/

Step 5: Add Source Code Management

Git: Add Repository URL and Credentials

https://github.com/deepkiraninc/react-components-js-fork.git/

Step 6: Add Branch:

*/deep-dev

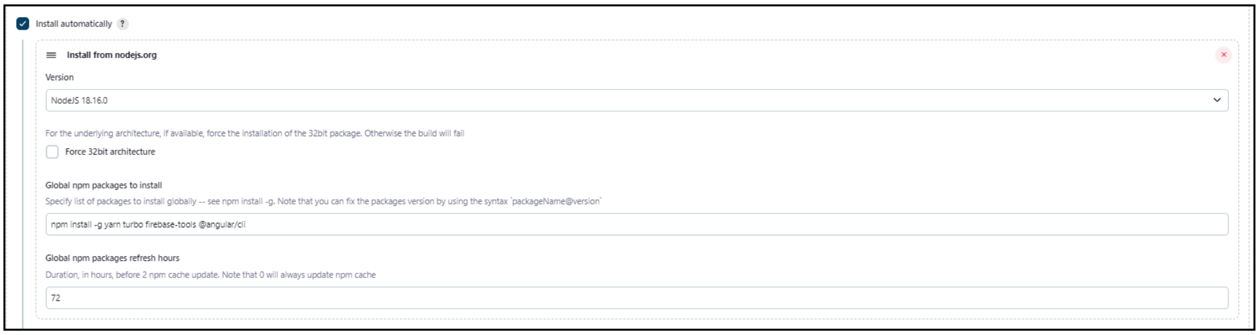
Step 7: Build Triggers:

Github hook trigger for GIT SCM polling

Step 9: Build Environment

Provide Node & npm bin/ folder to Path

- ➢ Add Nodejs from Manage Jenkins > Tool>
- ➢ Provide required information to add NodeJS
- ➢ Provide NodeJS 18.16.0



☑ Install automatically

Install from nodejs.org

Version

NodeJS 18.16.0

For the underlying architecture, if available, force the installation of the 32bit package. Otherwise the build will fail

□ Force 32bit architecture

Global npm packages to install

Specify list of packages to install globally -- see npm install -g. Note that you can fix the packages version by using the syntax `packageName@version`

npm install -g yarn turbo firebase-tools @angular/cli

Global npm packages refresh hours

Duration, in hours, before 2 npm cache update. Note that 0 will always update npm cache

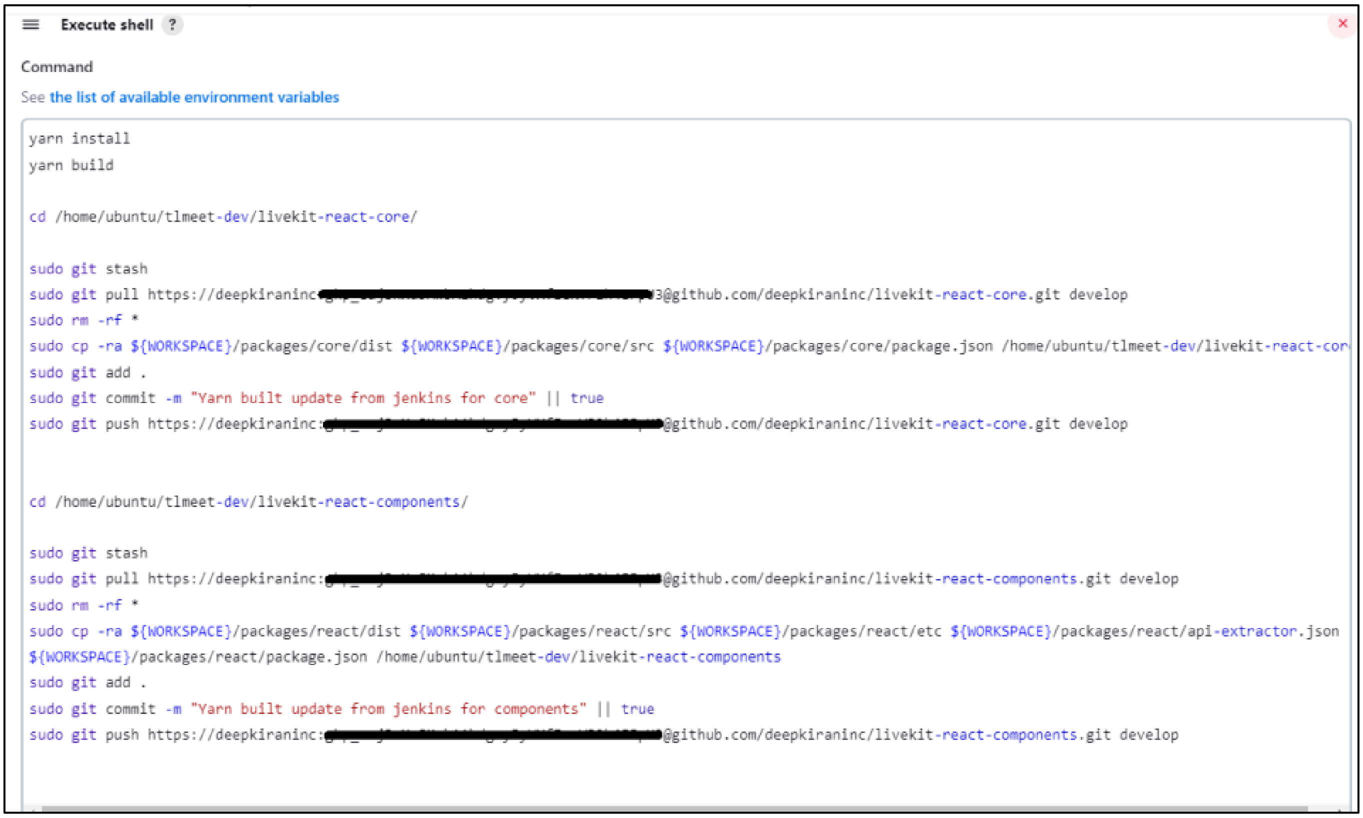
72

Figure 2: Add NodeJS in Manage Jenkins

Step 10: Build Steps:

Add Execute shell Command

This job builds on Jenkins master



```
yarn install
yarn build

cd /home/ubuntu/tlmeet-dev/livekit-react-core/

sudo git stash
sudo git pull https://deepkiraninc[redacted]@github.com/deepkiraninc/livekit-react-core.git develop
sudo rm -rf *
sudo cp -ra ${WORKSPACE}/packages/core/dist ${WORKSPACE}/packages/core/src ${WORKSPACE}/packages/core/package.json /home/ubuntu/tlmeet-dev/livekit-react-cor
sudo git add .
sudo git commit -m "Yarn built update from jenkins for core" || true
sudo git push https://deepkiraninc:[redacted]@github.com/deepkiraninc/livekit-react-core.git develop


cd /home/ubuntu/tlmeet-dev/livekit-react-components/

sudo git stash
sudo git pull https://deepkiraninc:[redacted]@github.com/deepkiraninc/livekit-react-components.git develop
sudo rm -rf *
sudo cp -ra ${WORKSPACE}/packages/react/dist ${WORKSPACE}/packages/react/src ${WORKSPACE}/packages/react/etc ${WORKSPACE}/packages/react/api-extractor.json
${WORKSPACE}/packages/react/package.json /home/ubuntu/tlmeet-dev/livekit-react-components
sudo git add .
sudo git commit -m "Yarn built update from jenkins for components" || true
sudo git push https://deepkiraninc:[redacted]@github.com/deepkiraninc/livekit-react-components.git develop
```

Figure 3: Execute shell Command

PURPOSE OF ABOVE COMMANDS

#yarn install

#yarn build

yarn install and yarn build are two common commands used in JavaScript and Node.js development, particularly when managing and building projects using the Yarn package manager. These commands serve different purposes.

- *yarn install:*

  After running yarn install, the dependencies listed in package.json will be downloaded and installed, making project ready to run

- *yarn build:*

  yarn build is used for building and preparing project for production deployment. It is a custom script defined in the scripts section of your package.json file.

#cd /home/ubuntu/tlmeet-dev/livekit-react-core/:

This command changes the current working directory to the specified path, "/home/ubuntu/tlmeet-dev/livekit-react-core/," where the project is located.

#sudo git stash:

This command stashes any uncommitted changes in the working directory, allowing you to save your changes temporarily without committing them.

#sudo git pull origin develop:

This command pulls the latest changes from the remote Git repository "https://github.com/deepkiraninc/livekit-react-core.git" into the current branch "develop". It updates your local branch with the changes from the remote repository.

#sudo rm -rf *:

This command recursively removes all files and directories in the current directory

# sudo cp -ra ${WORKSPACE}/packages/core/dist ${WORKSPACE}/packages/core/src ${WORKSPACE}/packages/core/package.json /home/ubuntu/tlmeet-dev/livekit-react-core/:

This command copies the specified files and directories from ${WORKSPACE} to the destination directory "/home/ubuntu/deepkiraninc/livekit-react-core/". It involves copying files related to the "core" package of your project.

#sudo git add .:

This command stages all changes in the current directory for the next commit. It prepares the changes made by the previous cp command for committing.

#sudo git commit -m "Yarn built update from Jenkins for core" || true:

This command commits the staged changes with the provided commit message. The || true is used to ensure that the command returns a success status code even if there are no changes to commit.

#sudo git push origin develop:

This command pushes the committed changes to the remote Git repository, specifically to the "develop" branch.

#cd /home/ubuntu/tlmeet-dev/livekit-react-components/:

This command changes the current working directory to the specified path, "/home/ubuntu/tlmeet-dev/livekit-react-components/," where the project is located.

#sudo git stash:

This command stashes any uncommitted changes in the working directory, allowing you to save your changes temporarily without committing them.

#sudo git pull origin develop:

This command pulls the latest changes from the remote Git repository "https://github.com/deepkiraninc/livekit-react-components.git" into the current branch "develop". It updates your local branch with the changes from the remote repository.

*#sudo rm -rf *:*

This command recursively removes all files and directories in the current directory.

#sudo cp -ra ${WORKSPACE}/packages/react/dist ${WORKSPACE}/packages/react/src ${WORKSPACE}/packages/react/etc ${WORKSPACE}/packages/react/api-extractor.json ${WORKSPACE}/packages/react/ package.json /home/ubuntu/tlmeet-dev/livekit-react-components:

This command copies the specified files and directories from ${WORKSPACE} to the destination directory "/home/ubuntu/tlmeet-dev/livekit-react-components." It appears to involve copying files related to the "react" package of your project.

#sudo git add . :

This command stages all changes in the current directory for the next commit.

#sudo git commit -m "Yarn built update from Jenkins for components" || true:

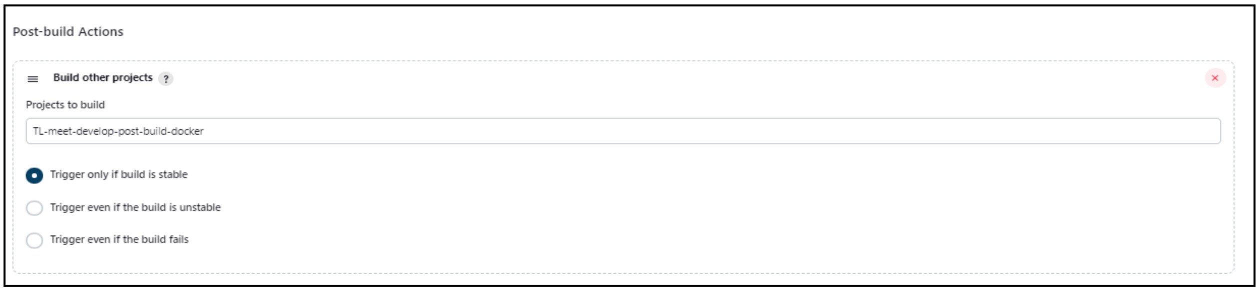
This command commits the staged changes with the provided commit message. The || true is used to ensure that the command returns a success status code even if there are no changes to commit.

#sudo git push origin develop:

This command pushes the committed changes to the remote Git repository, specifically to the "master" branch

Step 10: Post-Build Actions:

"Post-Build Actions" are a set of actions that are executed after the main build process has completed, regardless of whether the build was successful or failed. These actions provide a way to perform additional tasks, such as archiving artifacts, sending notifications, or triggering other jobs, once the build is finished.



Post-build Actions

Build other projects

Projects to build

TL-meet-develop-post-build-docker

● Trigger only if build is stable

○ Trigger even if the build is unstable

○ Trigger even if the build fails

Figure4: Post-Build Action

Here we Trigger the TL-Meet-develop-Post-Build Pipeline.

Step 11:

Once you've added all required details, scroll to the bottom of the page and click Apply, then Save. Here TL-meet-develop-post-build job will trigger and build.

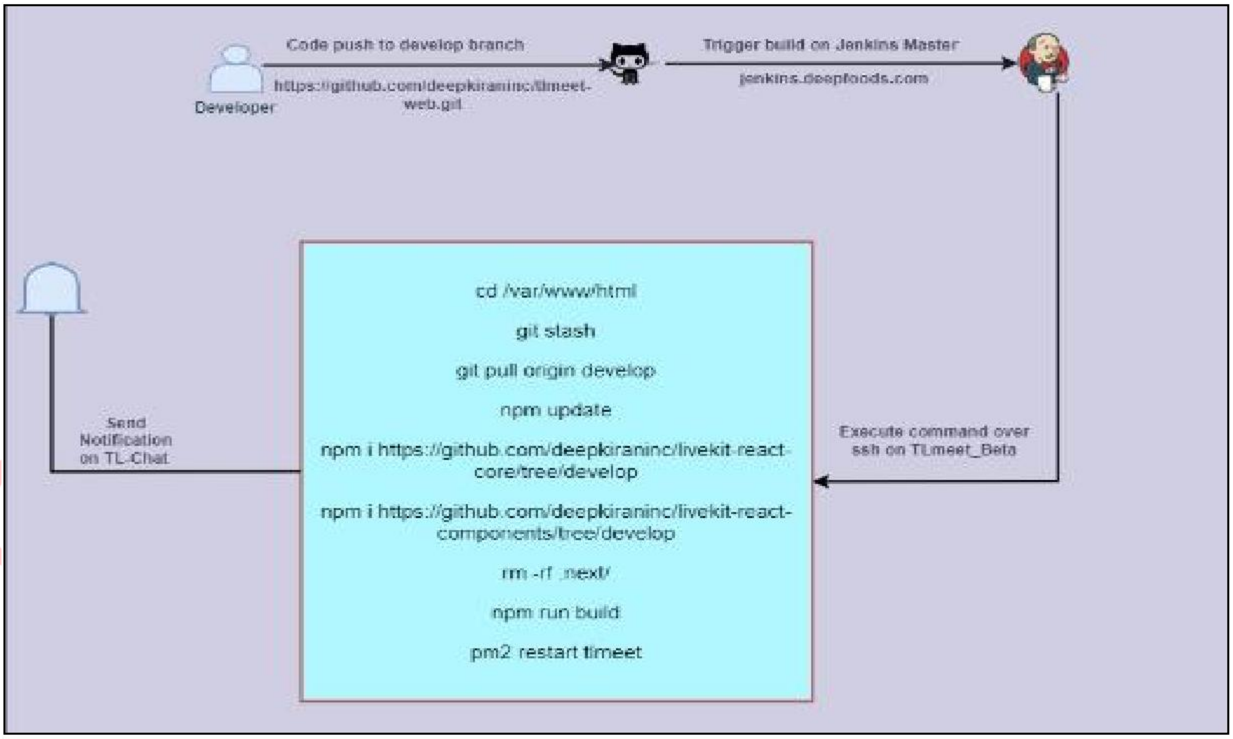
Step 12: Add Job Notifications:

We received CI/CD status change update notification on Teamlocus-Chat.

TL-Meet-Develop-Post-Build-Docker

This Pipeline trigger after TL-Meet-Develop will be successful.

Aim:


Code push to develop branch
Developer
https://github.com/deepkiraninc/tlmeet-web.git
Trigger build on Jenkins Master
jenkins.deepfoods.com
Execute command over ssh on TLmeet_Beta
Send Notification on TL-Chat

```
cd /var/www/html
git stash
git pull origin develop
npm update
npm i https://github.com/deepkiraninc/livekit-react-core/tree/develop
npm i https://github.com/deepkiraninc/livekit-react-components/tree/develop
rm -rf .next/
npm run build
pm2 restart tlmeet
```

Figuíe 5: TL-Meet-Develop-Post-Build-Docker

Prerequisite:

- Git URL
- SSH Seíveí
- NodeJs 18.16.0

Navigate to Jenkins

jenkins.deepfoods.com

Steps to create pipeline for TL-Meet-Post-Build-Docker

Step 1: Create a Job

On the Jenkins dashboard, click on 'New Item' at the left top side of the dashboard.

Step 2: Choose Freestyle Project

Enter a name under 'Enter an item name'. i.e, TL-meet-develop-post-build-docker

Then select 'Freestyle project' and click on 'OK'.

Step 3: Configure the Project

We can set the project name, description, and other settings.

Step 4: Add Project URL:

https://github.com/deepkiraninc/tlmeet-web.git/

Step 6: Add Source Code Management

Git: Add Repository URL and Credentials

https://github.com/deepkiraninc/tlmeet-web.git/

Step 7: Add Branch:

*/develop

Step 8: Build Triggers:

Github hook trigger for GITScm polling

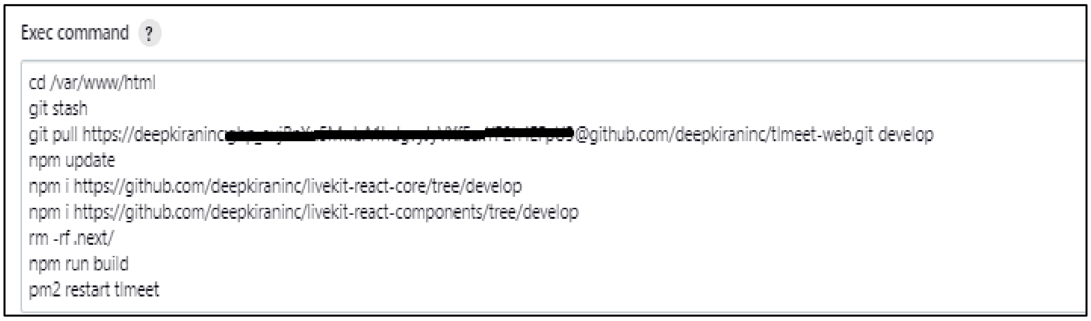
Step 9: Build Environment

Send files or execute commands over SSH after the build runs

Add SSH Publishers: TLmeet_Beta

Add Exec command:

Following commands added in executive shell. Commands will execute inside the container.



```
cd /var/www/html
git stash
git pull https://deepkiraninc[redacted]@github.com/deepkiraninc/tlmeet-web.git develop
npm update
npm i https://github.com/deepkiraninc/livekit-react-core/tree/develop
npm i https://github.com/deepkiraninc/livekit-react-components/tree/develop
rm -rf .next/
npm run build
pm2 restart tlmeet
```

Figure 6: Executive Shell Command

Purpose of above commands

#cd /var/www/html

Change your current directory to /var/www/html.

#git stash:

This command stashes any uncommitted changes in the working directory

#git pull origin develop:

This command pulls the latest changes from the remote Git repository

#npm update:

This command runs the npm update command using a specific version of Node.js (v19.9.0). It's used to update the packages (dependencies) in your Node.js project to their latest compatible versions as specified in your package.json file.

# npm i https://github.com/deepkiraninc/livekit-react-core/tree/develop:

This command installs a package directly from a GitHub repository. It's trying to install the "livekit-react-core" package from the "develop" branch.

# npm i https://github.com/deepkiraninc/livekit-react-components/tree/develop:

Installs the "livekit-react-components" package from the "develop" branch on GitHub.

#rm -rf .next/:

This command removes the .next directory. In the context of a Next.js application, the .next directory typically contains the output of the Next.js build process, including the compiled JavaScript files, assets, and other build artifacts.

#npm run build:

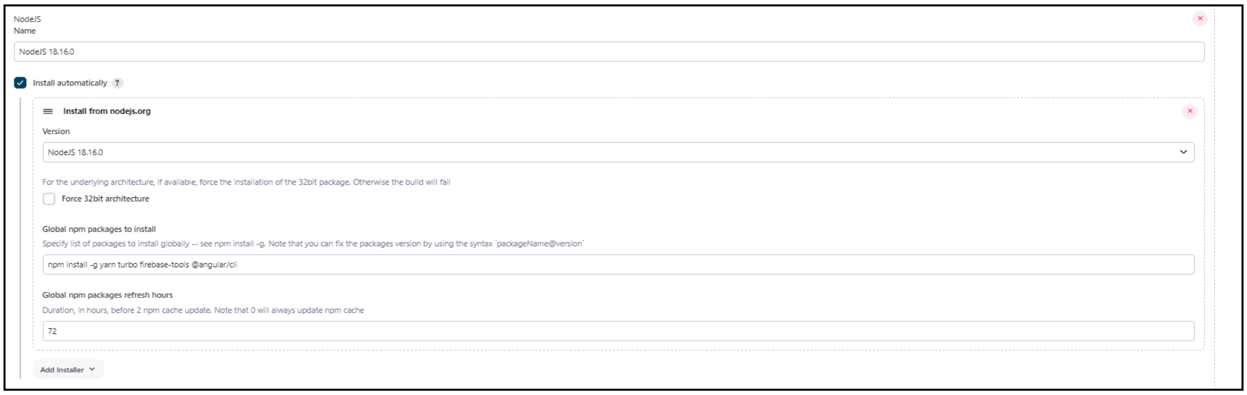
Build process for your Node.js application. It may compile and package your code for deployment.

#pm2 restart tlmeet:

Restarts your Node.js application using PM2, a process manager for Node.js applications. The "tlmeet" is assumed to be the name of the application managed by PM2.

Provide Node & npm bin/ folder to Path

- Add Nodejs from Manage Jenkins > Tool>
- Provide required information to add NodeJS
- Provide NodeJS 18.16.0



NodeJS
Name
NodeJS 18.16.0

☑ Install automatically ?

Install from nodejs.org

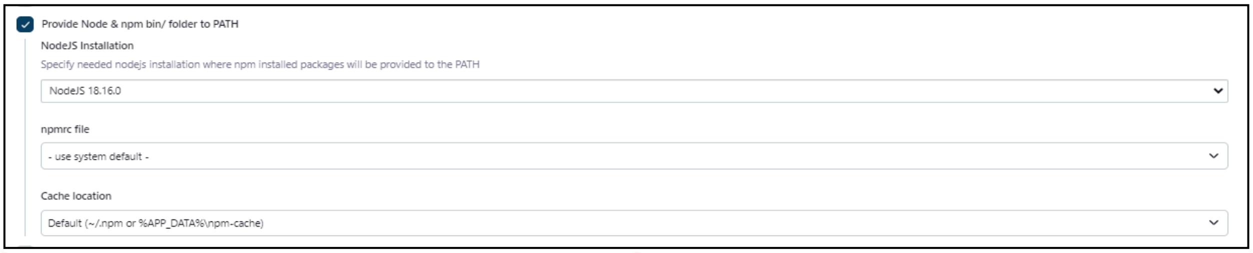
Version
NodeJS 18.16.0

For the underlying architecture, if available, force the installation of the 32bit package. Otherwise the build will fail
□ Force 32bit architecture

Global npm packages to install
Specify list of packages to install globally -- see npm install -g. Note that you can fix the packages version by using the syntax `packageName@version`
npm install -g yarn turbo firebase-tools @angular/cli

Global npm packages refresh hours
Duration, in hours, before 2 npm cache update. Note that 0 will always update npm cache
72

Figure 7: Add NodeJS in Manage Jenkins



☑ Provide Node & npm bin/ folder to PATH

NodeJS Installation
Specify needed nodejs installation where npm installed packages will be provided to the PATH
NodeJS 18.16.0

npmrc file
- use system default -

Cache location
Default (~/.npm or %APP_DATA%\npm-cache)

Figure 8: Add NodeJS in Build Environment

Step 10: Once you have added, scroll to the bottom of the page and click Apply, then Save.

Step 12: Add Job Notifications:

We received CI/CD status change update notification on Teamlocus-Chat