## CI/CD en Jenkins con nodeJS y un webhook de GitHub para realizar pruebas de building automatizado

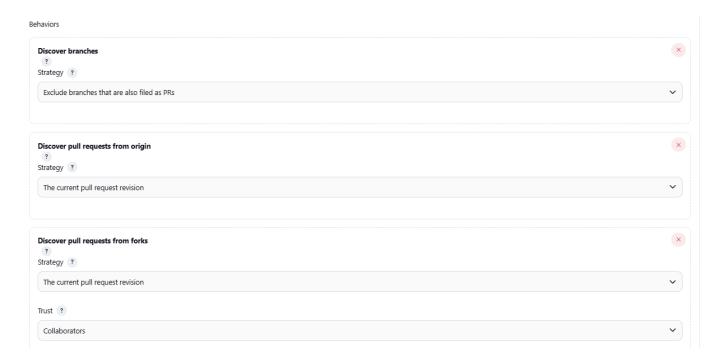
 El objetivo principal es automatizar los Pull Requests o git push de los desarrolladores para que el webhook de un repositorio ejecute un buid en Jenkins y testee un proyecto de nodeJS cada vez que ocurra alguno de los dos eventos.

#### Configuración en Jenkins:

- Instalar Jenkins como controlador en el entorno que más se adapte a las necesidades del proyecto. Ver documentación de referencia.
- Corroborar que esté instalado o instalar el plugin "GitHub Branch Source".



- Crear un nuevo item, darle un nombre y elegir "Multibranch Pipeline".
- En la configuración del nuevo job, en "Branch Sources" agregar la URL del repositorio de GitHub con el que Jenkins se comunicará. Si tiene credenciales, agregarlas también.
- En la sección "Build Configuration" asegurarse que la ruta del Jenkinsfile coincida con la ubicación en el repositorio de GitHub.
- Importante: Permisos para Pull requests. Se debe especificar el comportamiento a la hora de descubrir los Pull Requests, sobretodo en los forks. En este caso se confía en los colaboradores del repositorio para aceptar los PR.



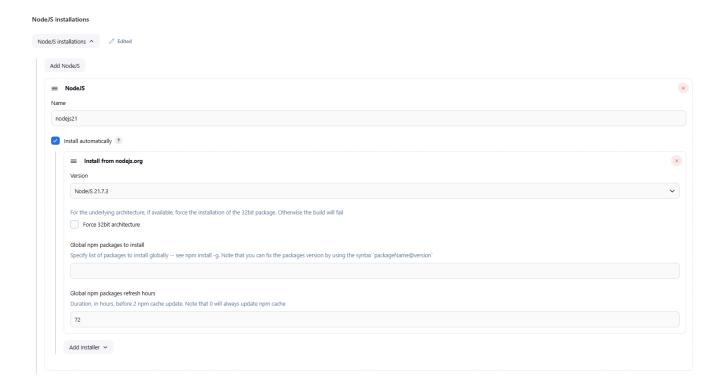
Aplicar y guardar.

### **Configuraciones adicionales:**

Al ejecutar una aplicación de nodeJS se debe instalar el plugin "NodeJS":



- Una vez instalado dirigirse a System Configuration >> Tools >> NodeJS Installations >> Add NodeJS
- El nombre de la nueva instalación debe coincidir con el nombre ingresado en el Jenkinsfile en el bloque tools {}
- En este caso sería "nodejs21"
- Elegir la versión de nodeJS que se requiera para el entorno, en este caso se elige
  "NodeJS 21.7.3"



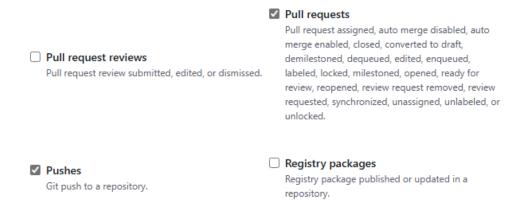
Aplicar y guardar.

## Configuración en ngrok:

- Instalar ngrok en la plataforma donde está instalado el controlador de Jenkins. Ver instalación en la documentación de referencia.
- Exponer el servicio de Jenkins a internet para que pueda ser accedido por GitHub.

## Configuración en GitHub:

- Crear un webhook en el repositorio que se comunicará con Jenkins especificando el "Payload URL" brindado por ngrok.
- Seleccionar los eventos individuales "Pull requests" y "Pushes":



# Comandos a ejecutar en cada build especificado en el Jenkinsfile:

#### Instalar dependencias:

npm install

#### Ejecutar las pruebas:

npm test

• En caso que falle la prueba, no se ejecutará la aplicación.

#### Ejecutar la aplicación:

npm start

#### Realizar un requerimiento para que devuelva el mensaje:

curl http://localhost:3000

#### Parar la aplicación:

pkill -f "node"

 Con esta configuración, Jenkins ejecutará el job de manera correcta cada vez que haga un git push o un pull request.



#### Referencias:

- Instalación de Jenkins: <a href="https://www.jenkins.io/doc/book/installing/">https://www.jenkins.io/doc/book/installing/</a>
- Branch Source Plugin: <a href="https://docs.cloudbees.com/docs/cloudbees-ci/latest/cloud-admin-guide/github-branch-source-plugin">https://docs.cloudbees.com/docs/cloudbees-ci/latest/cloud-admin-guide/github-branch-source-plugin</a>
- Instalación de ngrok: <a href="https://ngrok.com/docs/getting-started/">https://ngrok.com/docs/getting-started/</a>