

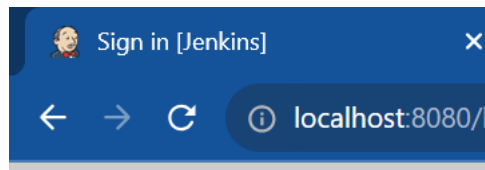
Trabajo Práctico N°7

Ejercicio 1

```
Usuario@DESKTOP-8G1HL0R MINGW64 ~/OneDrive/Escritorio/2023/Ing SW 3/tp7
$ docker build -t jenkins-with-dotnetcore -f Dockerfile.jenkins .
[+] Building 2.1s (9/9) FINISHED                                docker:default
=> [internal] load .dockerignore                                0.1s
=> => transferring context: 2B                                   0.0s
=> [internal] load build definition from Dockerfile.jenkins    0.1s
=> => transferring dockerfile: 456B                              0.0s
=> [internal] load metadata for docker.io/jenkins/jenkins:lts 1.8s
=> [auth] jenkins/jenkins:pull token for registry-1.docker.io 0.0s
=> [1/4] FROM docker.io/jenkins/jenkins:lts@sha256:b705323eaf70a7da4c1eed8b816f33dff2d5c8c3671170a2c17cf77aa4f15 0.0s
=> CACHED [2/4] RUN apt-get update && apt-get install -y      apt-transport-https software-properties-common 0.0s
=> CACHED [3/4] RUN wget -q https://packages.microsoft.com/config/ubuntu/20.04/packages-microsoft-prod.deb -O pa 0.0s
=> CACHED [4/4] RUN apt-get install -y dotnet-sdk-7.0         0.0s
=> exporting to image                                          0.0s
=> => exporting layers                                          0.0s
=> => writing image sha256:65890153844fd505f92b56b360024f7625086eeadd41971cf930f446d7e6f4e0 0.0s
=> => naming to docker.io/library/jenkins-with-dotnetcore    0.0s

What's Next?
View summary of image vulnerabilities and recommendations → docker scout quickview
```

```
Usuario@DESKTOP-8G1HL0R MINGW64 ~/OneDrive/Escritorio/2023/Ing SW 3/tp7
$ docker run -d -p 8080:8080 -p 50000:50000 --name jenkins -v /c/jenkins:/var/jenkins_home jenkins-with-dotnetcore
ef02474c26a980d4a323a3a556ec2049497992321922eae66421e9657eafb6c3
```



Create First Admin User

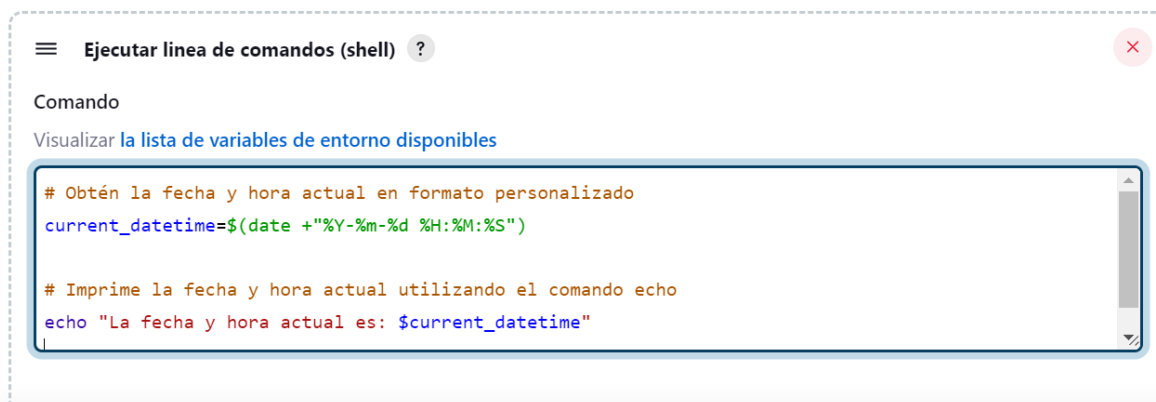
Usuario

Contraseña

Confirma la contraseña

Ejercicio 3

Build Steps



✓ Salida de consola

```
Started by user agustina
Running as SYSTEM
Building in workspace /var/jenkins_home/workspace/first-job
[first-job] $ /bin/sh -xe /tmp/jenkins7278965154484193005.sh
+ date +%Y-%m-%d %H:%M:%S
+ current_datetime=2023-09-26 18:21:19
+ echo La fecha y hora actual es: 2023-09-26 18:21:19
La fecha y hora actual es: 2023-09-26 18:21:19
Finished: SUCCESS
```

1. `Building in workspace /var/jenkins_home/workspace/first-job` : El trabajo se está ejecutando en el espacio de trabajo (`workspace`) llamado "first-job". Este es el lugar donde Jenkins almacena los archivos relacionados con este trabajo en particular.
2. `[first-job] $ /bin/sh -xe /tmp/jenkins7278965154484193005.sh` : Jenkins está ejecutando un script de shell (`/bin/sh`) en modo verbose (`x`) ubicado en `/tmp/jenkins7278965154484193005.sh` .
3. `+ date +%Y-%m-%d %H:%M:%S` : Este es un comando de shell para obtener la fecha y hora actual.
4. `+ current_datetime=2023-09-26 18:21:19` : El resultado del comando anterior (la fecha y hora actual) se almacena en una variable llamada `current_datetime` .
5. `Finished: SUCCESS` : El trabajo ha finalizado correctamente (`SUCCESS`). Esto significa que el trabajo se ejecutó sin errores y completó todas sus tareas con éxito.

Ejercicio 4

Un pipeline en Jenkins es una forma de definir y automatizar un conjunto de pasos que deben seguirse para construir, probar y desplegar una aplicación.

Pipeline

Definition

Pipeline script

Script ?

```
1 pipeline {  
2   agent any  
3  
4   stages {  
5     stage('Hello') {  
6       steps {  
7         echo 'Hello World'  
8       }  
9     }  
10  }  
11 }  
12
```

Hello World



Salida de consola

```
Started by user agustina  
[Pipeline] Start of Pipeline  
[Pipeline] node  
Running on Jenkins in /var/jenkins_home/workspace/hello-world  
[Pipeline] {  
[Pipeline] stage  
[Pipeline] { (Hello)  
[Pipeline] echo  
Hello World  
[Pipeline] }  
[Pipeline] // stage  
[Pipeline] }  
[Pipeline] // node  
[Pipeline] End of Pipeline  
Finished: SUCCESS
```

Step	Arguments	Status
Start of Pipeline - (11 Seg in block)		✓
node - (4.2 Seg in block)		✓
node block - (0.79 Seg in block)		✓
stage - (0.55 Seg in block)	Hello	✓
stage block (Hello) - (0.39 Seg in block)		✓
echo - (23 Ms in self)	Hello World	✓

1. `[Pipeline] Start of Pipeline` : Indica el inicio del pipeline.
2. `[Pipeline] node` : Este paso indica que el pipeline se está ejecutando en un nodo de Jenkins (agente) que está configurado para manejar trabajos.
3. `Running on Jenkins in /var/jenkins_home/workspace/hello-world` : Indica que el pipeline se está ejecutando en un espacio de trabajo llamado "hello-world" ubicado en el directorio `/var/jenkins_home/workspace` .
4. `[Pipeline] {` : Marca el comienzo del bloque principal del pipeline.
5. `[Pipeline] stage` : Define un escenario dentro del pipeline. Los escenarios son etapas o pasos individuales que componen el proceso completo.
6. `[Pipeline] { (Hello)` : Indica el inicio del escenario llamado "Hello".
7. `[Pipeline] echo` : Este comando imprime un mensaje en la consola del pipeline.
8. `Hello World` : El mensaje "Hello World" es impreso en la consola.
9. `[Pipeline] }` : Marca el fin del escenario "Hello".
10. `[Pipeline] // stage` : Indica el fin de la etapa actual.
11. `[Pipeline] }` : Marca el fin del bloque principal del pipeline.
12. `[Pipeline] // node` : Indica el fin de la ejecución en el nodo.
13. `[Pipeline] End of Pipeline` : Indica el final del pipeline.
14. `Finished: SUCCESS` : El pipeline ha finalizado correctamente (`SUCCESS`). Esto significa que todos los pasos y etapas definidos en el pipeline se ejecutaron sin errores y el proceso completo fue exitoso.

Ejercicio 5

Pipeline

Definition

Pipeline script

Script ?

```
1 pipeline {
2   agent any
3
4
5   stages {
6     stage('Build') {
7       steps {
8         // Get some code from a GitHub repository
9         git branch: 'main', url: 'https://github.com/ingsoft3ucc/SimpleWebAPI'
10      }
11    }
12  }
13 }
14 }
```

try sample Pipeline... ▾

```
Started by user agustina
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/jenkins_home/workspace/github-job
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/ingsoft3ucc/SimpleWebAPI
> git init /var/jenkins_home/workspace/github-job # timeout=10
Fetching upstream changes from https://github.com/ingsoft3ucc/SimpleWebAPI
> git --version # timeout=10
> git --version # 'git version 2.39.2'
> git fetch --tags --force --progress -- https://github.com/ingsoft3ucc/SimpleWebAPI +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/ingsoft3ucc/SimpleWebAPI # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 8726ce58a1894af134cd22fcfb7ecb38ad9444 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 8726ce58a1894af134cd22fcfb7ecb38ad9444 # timeout=10
> git branch -a -v --no-abbrev # timeout=10
> git checkout -b main 8726ce58a1894af134cd22fcfb7ecb38ad9444 # timeout=10
Commit message: "Update WeatherForecastController.cs"
First time build. Skipping changelog.
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Este job es un pipeline que está configurado para clonar un repositorio de Git desde GitHub, específicamente el repositorio <https://github.com/ingsoft3ucc/SimpleWebAPI>

Clonación del Repositorio Git (`[Pipeline] git`): En esta etapa, Jenkins clona el repositorio de Git desde <https://github.com/ingsoft3ucc/SimpleWebAPI>.

- **No credentials specified**: Indica que no se han especificado credenciales para acceder al repositorio. Esto puede ser un problema si el repositorio es privado o requiere autenticación.
- **Checking out Revision 8726ce58a1894af134cd22fcfb7ecb38ad9444**: Jenkins está verificando la revisión (commit) con el hash `8726ce58a1894af134cd22fcfb7ecb38ad9444`.
- **Commit message: "Update WeatherForecastController.cs"**: Muestra el mensaje del commit.

- **First time build. Skipping changelog.**: Indica que este es el primer build, por lo que se omite el registro de cambios (changelog).

Ejercicio 6

Este job realizado en Jenkins ejecutó una serie de pasos para construir y publicar un proyecto de API en .NET Core llamado "SimpleWebAPI" desde un repositorio de GitHub.

```
Started by user agustina
Running as SYSTEM
Building in workspace /var/jenkins_home/workspace/git-netcore-job
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/jenkins_home/workspace/git-netcore-job/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/aaraya0/SimpleWebAPI # timeout=10
Fetching upstream changes from https://github.com/aaraya0/SimpleWebAPI
> git --version # timeout=10
> git --version # 'git version 2.39.2'
> git fetch --tags --force --progress -- https://github.com/aaraya0/SimpleWebAPI +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 030c545ec18b57edcc19c087b23856e69f57e47c (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 030c545ec18b57edcc19c087b23856e69f57e47c # timeout=10
Commit message: "primer commit"
> git rev-list --no-walk 4de8a948dd1befb52249bcc52579dbb4da62bfc0 # timeout=10
[git-netcore-job] $ dotnet build SimpleWebAPI.sln
MSBuild version 17.7.3+8ec440e68 for .NET
Determining projects to restore...
Restored /var/jenkins_home/workspace/git-netcore-job/SimpleWebAPI/SimpleWebAPI.csproj (in 13.45 sec).
SimpleWebAPI -> /var/jenkins_home/workspace/git-netcore-job/SimpleWebAPI/bin/Debug/net7.0/SimpleWebAPI.dll

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:25.94
.NET Command Completed - Exit Code: 0

[git-netcore-job] $ dotnet publish SimpleWebAPI.sln --self-contained false
MSBuild version 17.7.3+8ec440e68 for .NET
Determining projects to restore...
All projects are up-to-date for restore.
SimpleWebAPI -> /var/jenkins_home/workspace/git-netcore-job/SimpleWebAPI/bin/Debug/net7.0/SimpleWebAPI.dll
SimpleWebAPI -> /var/jenkins_home/workspace/git-netcore-job/SimpleWebAPI/bin/Debug/net7.0/publish/
.NET Command Completed - Exit Code: 0

Finished: SUCCESS
```

1. Clonación del Repositorio:

- Se clona el repositorio desde la URL `https://github.com/aaraya0/SimpleWebAPI`.

2. Chequeo y Cambio de Revisión:

- Se verifica y cambia la revisión (commit) a `030c545ec18b57edcc19c087b23856e69f57e47c`.

3. Construcción del Proyecto:

- Se ejecuta el comando `dotnet build SimpleWebAPI.sln` para construir el proyecto `SimpleWebAPI`. La construcción es exitosa y el proyecto se compila sin errores.
- Se informa que no hay advertencias ni errores.
- El tiempo total de construcción fue de aproximadamente 25.94 segundos.

4. Publicación del Proyecto:

- Se ejecuta el comando `dotnet publish SimpleWebAPI.sln --self-contained false` para publicar el proyecto `SimpleWebAPI`. Esto significa que se prepara para su distribución, pero no se incluye el entorno de ejecución.
- La publicación se completa con éxito y el proyecto está listo para ser desplegado.

Finalmente, el estado del job es "SUCCESS", lo que indica que todos los pasos se ejecutaron sin errores y el proyecto fue construido y publicado correctamente. El proyecto resultante se encuentra en el directorio `/var/jenkins_home/workspace/git-netcore-job/SimpleWebAPI/bin/Debug/net7.0/publish/`.

Monitorear Cambios en el repo de GitHub

```
Usuario@DESKTOP-8G1HL0R MINGW64 ~/OneDrive/Escritorio/2023/Ing SW 3/SimpleWebAPI (main)
$ git add .

Usuario@DESKTOP-8G1HL0R MINGW64 ~/OneDrive/Escritorio/2023/Ing SW 3/SimpleWebAPI (main)
$ git commit -m "cambios read me para jenkins!"
[main 98678f2] cambios read me para jenkins!
1 file changed, 1 insertion(+), 1 deletion(-)

Usuario@DESKTOP-8G1HL0R MINGW64 ~/OneDrive/Escritorio/2023/Ing SW 3/SimpleWebAPI (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 302 bytes | 302.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/aaraya0/SimpleWebAPI.git
030c545..98678f2 main -> main
```

... #4

| (pending—Finished waiting)

Cambios

Summary

1. cambios read me para jenkins! ([details](#))

Commit 98678f2398f94b381d8f0b17c5d669871e371b3b by agusaraya131
cambios read me para jenkins!

 [README.md \(diff\)](#)

Explicar por que no es posible tener un WebHook en nuestro TP

- Con el sondeo (polling), Jenkins verifica el repositorio a intervalos regulares y ejecuta el job si detecta cambios.
- Con un WebHook, GitHub notifica a Jenkins en tiempo real sobre eventos específicos, lo que resulta en una respuesta más rápida.

No se puede configurar un WebHook en el entorno del trabajo práctico por diversas razones de seguridad y configuración. Esto puede incluir restricciones de red o firewall en entornos restringidos, limitaciones de permisos de usuario para configurar WebHooks en el repositorio de GitHub, configuraciones específicas del entorno que restringen ciertas comunicaciones, políticas de seguridad estrictas, falta de acceso a Internet en entornos de pruebas aislados y restricciones adicionales impuestas para mantener el entorno de pruebas controlado y aislado. Estas medidas se implementan para proteger la infraestructura y los datos de posibles vulnerabilidades o riesgos de seguridad.

Explicar Diferencia con ejecutar periódicamente

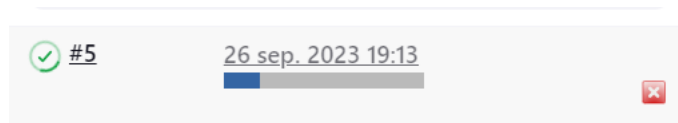
```

Usuario@DESKTOP-8G1HL0R MINGW64 ~/OneDrive/Escritorio/2023/Ing SW 3/SimpleWebAPI (main)
$ git add .

Usuario@DESKTOP-8G1HL0R MINGW64 ~/OneDrive/Escritorio/2023/Ing SW 3/SimpleWebAPI (main)
$ git commit -m "cambios en read me para jenkins parte 2!"
[main e6d72ef] cambios en read me para jenkins parte 2!
1 file changed, 1 insertion(+), 1 deletion(-)

Usuario@DESKTOP-8G1HL0R MINGW64 ~/OneDrive/Escritorio/2023/Ing SW 3/SimpleWebAPI (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 316 bytes | 316.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/aaraya0/SimpleWebAPI.git
 98678f2..e6d72ef main -> main

```



Compilar #5 (26 sep. 2023 19:13:00)

Conservar esta ejecución pa

añadir descripción

Comenzó hace 46 Seg

Tardó 16 Seg



Changes

1. cambios en read me para jenkins parte 2! ([details](#) / [githubweb](#))



Ejecutado por el programador



Revision: e6d72ef88d25b05ba2eff55f35bd541374d8d133

Repository: <https://github.com/aaraya0/SimpleWebAPI>

- refs/remotes/origin/main

- **Polling (Sondeo):** En el enfoque de sondeo, Jenkins consulta periódicamente el repositorio de control de versiones (como Git) para verificar si hay cambios. Si se detectan cambios, Jenkins inicia la construcción del proyecto. Esta es la técnica utilizada en este caso y es útil cuando no es posible configurar WebHooks.
- **Ejecución periódica:** En este enfoque, Jenkins ejecuta el job según un horario predefinido, independientemente de si ha habido cambios en el repositorio o no. Esto es útil cuando se desea ejecutar un job en intervalos regulares, independientemente de si hay cambios recientes en el repositorio.

La principal diferencia es que el enfoque de sondeo verifica activamente el repositorio en busca de cambios, mientras que la ejecución periódica simplemente ejecuta el job en un horario predeterminado. El sondeo es más eficiente y responde a los cambios en tiempo real, mientras que la ejecución periódica es útil para tareas que deben realizarse regularmente, incluso si no hay cambios recientes en el repositorio.

Ejercicio 6.3

✓ Provide Node & npm bin/ folder to PATH

NodeJS Installation

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

NodeJS

npmrc file

- use system default -

Cache location

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

Build Steps

≡ Ejecutar linea de comandos (shell) ?

Comando

Visualizar [la lista de variables de entorno disponibles](#)

```
npm install
npm run build
```

Avanzado ▾

git-node-job > #9 > Salida de consola

✓ Salida de consola

ut

in text

ción de la ejecución

#9'

/ia

```
Started by user agustina
Running as SYSTEM
Building in workspace /var/jenkins_home/workspace/git-node-job
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/jenkins_home/workspace/git-node-job/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/aaraya0/my-app # timeout=10
Fetching upstream changes from https://github.com/aaraya0/my-app
> git --version # timeout=10
> git --version # 'git version 2.39.2'
> git fetch --tags --force --progress -- https://github.com/aaraya0/my-app +refs/heads/*:refs/remotes/origin/* #
timeout=10
```

```
Started by user agustina
Running as SYSTEM
Building in workspace /var/jenkins_home/workspace/git-node-job
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/jenkins_home/workspace/git-node-job/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/aaraya0/my-app # timeout=10
```

```

Fetching upstream changes from https://github.com/aaraya0/my-app
> git --version # timeout=10
> git --version # 'git version 2.39.2'
> git fetch --tags --force --progress -- https://github.com/aaraya0/my-app +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 888a973234e3dfbe14460f7cdd87de055b414cf4 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 888a973234e3dfbe14460f7cdd87de055b414cf4 # timeout=10
Commit message: "primer commit my-app"
> git rev-list --no-walk 888a973234e3dfbe14460f7cdd87de055b414cf4 # timeout=10
Unpacking https://nodejs.org/dist/v16.15.0/node-v16.15.0-linux-x64.tar.gz to /var/jenkins_home/tools/jenkins.plugins.nodejs.tools.Node
[git-node-job] $ /bin/sh -xe /tmp/jenkins4319811280114648665.sh
+ npm install
npm WARN deprecated workbox-cacheable-response@6.6.0: workbox-background-sync@6.6.0
npm WARN deprecated w3c-hr-time@1.0.2: Use your platform's native performance.now() and performance.timeOrigin.
npm WARN deprecated stable@0.1.8: Modern JS already guarantees Array#sort() is a stable sort, so this library is deprecated. See the c
npm WARN deprecated source-map-codec@1.4.8: Please use @jridgewell/source-map-codec instead
npm WARN deprecated rollup-plugin-terser@7.0.2: This package has been deprecated and is no longer maintained. Please use @rollup/plugin
npm WARN deprecated svgo@1.3.2: This SVGO version is no longer supported. Upgrade to v2.x.x.
npm WARN deprecated @babel/plugin-proposal-private-methods@7.18.6: This proposal has been merged to the ECMAScript standard and thus t
npm WARN deprecated @babel/plugin-proposal-optional-chaining@7.21.0: This proposal has been merged to the ECMAScript standard and thus
npm WARN deprecated @babel/plugin-proposal-nullish-coalescing-operator@7.18.6: This proposal has been merged to the ECMAScript standar
npm WARN deprecated @babel/plugin-proposal-class-properties@7.18.6: This proposal has been merged to the ECMAScript standard and thus i
npm WARN deprecated @babel/plugin-proposal-numeric-separator@7.18.6: This proposal has been merged to the ECMAScript standard and thus

added 1521 packages, and audited 1522 packages in 2m

245 packages are looking for funding
  run `npm fund` for details

6 high severity vulnerabilities

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
+ npm run build

> my-app@0.1.0 build
> react-scripts build

Creating an optimized production build...
[0;33mOne of your dependencies, babel-preset-react-app, is importing the
"@babel/plugin-proposal-private-property-in-object" package without
declaring it in its dependencies. This is currently working because
"@babel/plugin-proposal-private-property-in-object" is already in your
node_modules folder for unrelated reasons, but it [1mmay break at any time[0;33m.

babel-preset-react-app is part of the create-react-app project, [1mwhich
is not maintained anymore[0;33m. It is thus unlikely that this bug will
ever be fixed. Add "@babel/plugin-proposal-private-property-in-object" to
your devDependencies to work around this error. This will make this message
go away.[0m

Compiled successfully.

File sizes after gzip:

  46.61 kB   build/static/js/main.46f5c8f5.js
  1.78 kB    build/static/js/787.28cb0dcd.chunk.js
  541 B      build/static/css/main.073c9b0a.css

The project was built assuming it is hosted at /.
You can control this with the homepage field in your package.json.

The build folder is ready to be deployed.
You may serve it with a static server:

  npm install -g serve
  serve -s build

Find out more about deployment here:

  https://cra.link/deployment

Finished: SUCCESS

```

Instalación de Node.js:

- Se desempaquetó e instaló Node.js en el entorno de Jenkins.

Ejecución de comandos npm:

- Se ejecutó npm install para instalar las dependencias del proyecto.
- Se ejecutó npm run build para construir la aplicación de React.

Resultados de la construcción:

- La construcción fue exitosa, se creó una versión optimizada de la aplicación.