



Universidad
CATÓLICA
de Pereira

ESPECIALIZACIÓN EN
**Desarrollo
de Software**

Proceso de Desarrollo de Software

Andrés Mauricio Martínez Hincapié

Actividad práctica #2: Pipelines con Jenkins

Objetivo: Crear un pipeline en Jenkins para un proyecto en GitHub, aplicado a un proyecto de React, y configurar las etapas de checkout, build (con npm install), y pruebas unitarias.

1. Instalar plugin NodeJS Plugin y Pipeline

🔍📄🔄📄🔄

Install	Name ↓	Released
<input checked="" type="checkbox"/>	NodeJS 1.6.4 npm NodeJS Plugin executes NodeJS script as a build step.	Hace 2 Mes 29 días

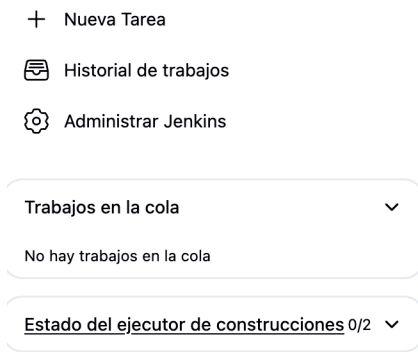
🔍📄🔄📄🔄

Install	Name ↓	Released
<input checked="" type="checkbox"/>	Pipeline 608.v67378e9d3db_1 Plugins para la interfaz de línea de comandos Plugins para el control de nodos Plugins lanzadores de tareas Varios A suite of plugins that lets you orchestrate automation, simple or complex. See Pipeline as Code with Jenkins for more details.	Hace 2 Mes 2 días

2. Configuración del Pipeline

A. Crear un Nuevo Pipeline

1. En Jenkins, haz clic en **Nueva Tarea** → **Pipeline**.



2. Nómbralo (ej: `react-pipeline-demo`) y selecciona **Pipeline**.



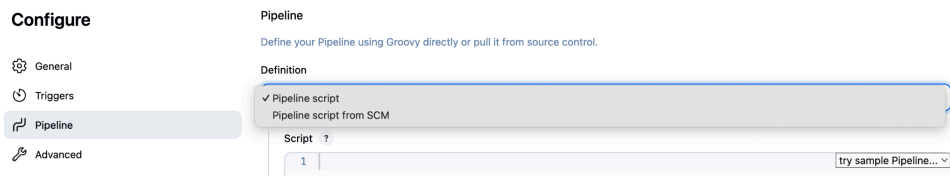
Pipeline

Gestiona actividades de larga duración que pueden abarcar varios agentes de construcción. Apropiado para construir pipelines (conocidas anteriormente como workflows) y/o para la organización de actividades complejas que no se pueden articular fácilmente con tareas de tipo freestyle.

B. Configurar el Pipeline desde SCM (GitHub)

1. En la sección **Pipeline**, elige:

- **Definition:** *Pipeline script from SCM.*



- **SCM:** *Git.*

- **Repository URL:** `https://github.com/username/ucp-app-react.git` (tu repositorio).



- **Branch:** `main` o `master`.

Branches to build ?

Branch Specifier (blank for 'any') ?

*/main

- **Script Path:** `Jenkinsfile` (crearemos este archivo en el repositorio).

Script Path ?

Jenkinsfile

☒ Lightweight checkout ?

[Pipeline Syntax](#)

- **Guardar**

Save

Apply

3. Crear el `Jenkinsfile` en el Repositorio

Crea un archivo `Jenkinsfile` en la raíz de tu proyecto React con este contenido:

```
pipeline {
  agent any

  tools {
    nodejs 'Node_24' // Nombre definido en Global Tool Configuration
  }

  stages {
    // Etapa 1: Checkout del código desde GitHub
    stage('Checkout') {
      steps {
        git branch: 'main', url:
'https://github.com/amartinezh/ucp-app-react.git'
      }
    }

    // Etapa 2: Instalar dependencias y build del proyecto
    stage('Build') {
      steps {
        sh 'npm install'
        sh 'npm run build' // Ejecuta el build de React
      }
    }

    // Etapa 3: Ejecutar pruebas unitarias
    stage('Unit Tests') {
      steps {
        sh 'npm test -- --watchAll=false --silent > test-output.txt || true'
        // Ejecuta pruebas sin modo interactivo
        // Muestra el reporte simple en la consola
        sh 'cat test-output.txt'
      }
      post {
        always {
          archiveArtifacts artifacts: 'test-output.txt',
allowEmptyArchive: true
        }
      }
    }
  }
}
```

```

    }
  }
}

// Post-actions (opcional)
post {
  success {
    echo '¡Pipeline ejecutado con éxito!'
  }
  failure {
    echo 'Pipeline fallido. Revisar logs.'
  }
}
}
}

```

Explicación del Jenkinsfile

- **Checkout:** Clona el repositorio GitHub.
- **Build:** Instala dependencias (`npm install`) y construye el proyecto (`npm run build`).
- **Unit Tests:** Ejecuta pruebas con Jest (`npm test`) y genera reportes en formato JUnit.
- **Post-actions:** Notificaciones de éxito/fallo.

4. Preparar la prueba unitaria

- Teniendo en cuenta nuestro componente principal:

```

src > JS App.js > App
Andres Mauricio Martinez Hincapie, 3 days ago | 1 author (Andres Mauricio Martinez Hincapie)
1  import React from 'react';
2
3  function App() {
4    return (
5      <div>
6        <h1>Universidad Católica de Pereira !</h1>
7        <p>Listado de integrantes - Proceso de Desarrollo de Software I</p>
8        <p>profe: Andrés Mauricio Martinez Hincapie</p>
9      </div>
10   );
11 }
12
13 export default App;

```

- **Agregaremos** nuestra prueba unitaria:

```

import { render, screen } from '@testing-library/react';
import App from './App';

test('renders learn react link', () => {
  render(<App />);
  const linkElement = screen.getByText(/¡Universidad Católica de Pereira !/i);
  expect(linkElement).toBeInTheDocument();
});

```

- Ejecutamos la prueba para validar que funcione:

```
MacBook-Air-de-Andres:ucp-app andresmauriciomartinezhincapie$ npm test -- --watchAll=false
> ucp-app@0.1.0 test
> react-scripts test --watchAll=false

PASS src/App.test.js
  ✓ renders learn react link (13 ms)

Test Suites: 1 passed, 1 total
Tests: 1 passed, 1 total
Snapshots: 0 total
Time: 0.718 s, estimated 1 s
Ran all test suites.
MacBook-Air-de-Andres:ucp-app andresmauriciomartinezhincapie$
```

5. Configurar Jenkins para el Pipeline

1. Asegúrate de que Node.js esté disponible:

- Ve a *Administrar Jenkins > Tool*.



Tools

Configure tools, their locations and automatic installers.

- Agrega Node.js (ej: Node_24) y marca **Install automatically**.

instalaciones de NodeJS

Añadir NodeJS

Nombre: Node_24

NodeJS

Nombre

Node 24

☒ Instalar automáticamente

Install from nodejs.org

Versión

NodeJS 24.0.2

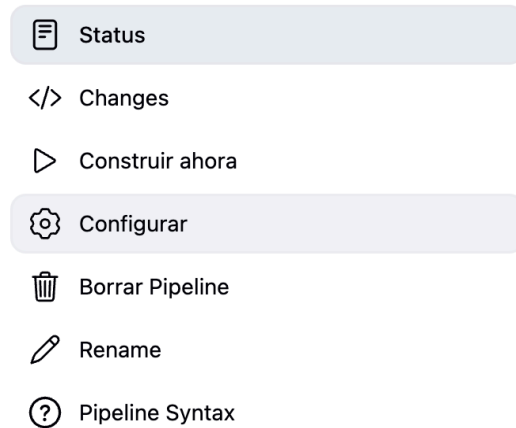
2. Configura credenciales para GitHub (si es un repo privado):

- En *Manage Jenkins > Credentials*, agrega un **Secret Text** con tu token de GitHub.
- Modifica el `Jenkinsfile` para usar credenciales:

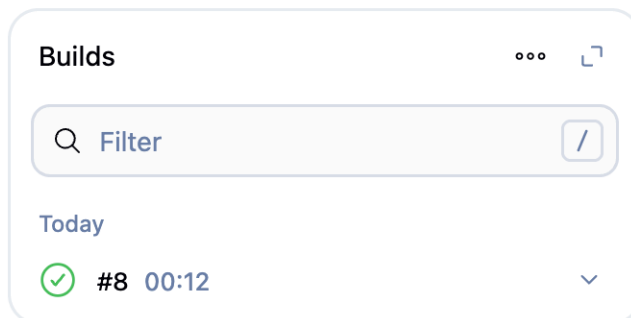
```
git branch: 'main', url:
'https://github.com/username/ucp-app-react.git', credentialsId:
'github-token'
```

6. Ejecutar el Pipeline

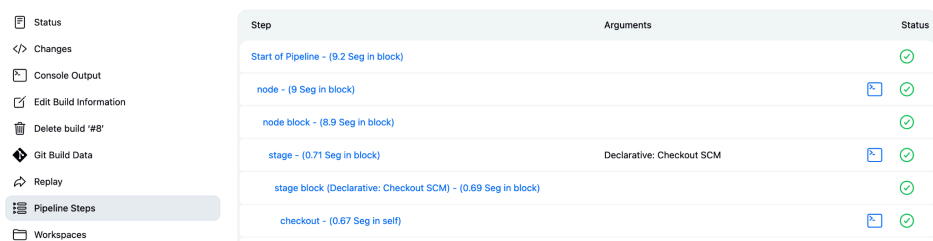
1. **Guarda** la configuración del pipeline en Jenkins.
2. Haz clic en **Construir ahora**.



3. **Visualiza el resultado del pipeline:**



- En la página del job, haz clic en **Pipeline Steps**



7. Verificar Resultados

- **Console Output:** Revisa los logs de cada etapa.

- **Test Reports:** En la página del build, ve a *Test Result* para ver detalles de las pruebas.
- **Build Artifacts:** Si generas archivos (ej: `build/`), guárdalos como artefactos.



Salida de consola

[Download](#)[Copy](#)[View as plain text](#)

```
Started by user unknown or anonymous
Obtained Jenkinsfile from git https://github.com/amartinezh/ucp-app-react.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/jenkins_home/workspace/react-pipeline-demo
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
```