Jenkins

⊗ **Console Output**   Download   Copy   View as plain text

- Status
- Changes
- Console Output
- Edit Build Information
- Timings
- Git Build Data
- Pipeline Overview
- Pipeline Console
- Thread Dump
- Pause/resume
- Replay
- Pipeline Steps
- Workspaces
- Previous Build

```
Started by user unknown or anonymous
Obtained Jenkinsfile from git https://github.com/khyron/Prueba_CICD.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/jenkins_home/workspace/Prueba_final_CICD
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential b61b048f-79a6-456f-8cbf-33cfaef2b971
 > git rev-parse --resolve-git-dir /var/jenkins_home/workspace/Prueba_final_CICD/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/khyron/Prueba_CICD.git # timeout=10
Fetching upstream changes from https://github.com/khyron/Prueba_CICD.git
 > git --version # timeout=10
 > git --version # 'git version 2.39.5'
using GIT_ASKPASS to set credentials git
 > git fetch --tags --force --progress -- https://github.com/khyron/Prueba_CICD.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision ac148abe1e92552e566800ca751095476b6311a8 (refs/remotes/origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f ac148abe1e92552e566800ca751095476b6311a8 # timeout=10
Commit message: "force test error"
 > git rev-list --no-walk 764f52463b76aa6b35a375be85939bec55d161a4 # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Checkout)
[Pipeline] echo
🐙 Clonando el repositorio...
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential b61b048f-79a6-456f-8cbf-33cfaef2b971
 > git rev-parse --resolve-git-dir /var/jenkins_home/workspace/Prueba_final_CICD/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/khyron/Prueba_CICD.git # timeout=10
Fetching upstream changes from https://github.com/khyron/Prueba_CICD.git
 > git --version # timeout=10
 > git --version # 'git version 2.39.5'
using GIT_ASKPASS to set credentials git
 > git fetch --tags --force --progress -- https://github.com/khyron/Prueba_CICD.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision ac148abe1e92552e566800ca751095476b6311a8 (refs/remotes/origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f ac148abe1e92552e566800ca751095476b6311a8 # timeout=10
Commit message: "force test error"
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] script
[Pipeline] {
[Pipeline] echo
⚙ Instalando dependencias...
[Pipeline] sh
+ npm install

up to date, audited 356 packages in 2s

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

found 0 vulnerabilities
[Pipeline] sh
+ npm run build

> ci_pipeline_project@1.0.0 build
> echo 'No hay proceso de compilación en este proyecto'

No hay proceso de compilación en este proyecto
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Test)
[Pipeline] script
[Pipeline] {
[Pipeline] echo
🧪 Ejecutando pruebas...
[Pipeline] sh
+ npm test

> ci_pipeline_project@1.0.0 test
> jest --coverage --forceExit

FAIL tests/app.test.js
  API Tests
    ✓ should return a list of users (78 ms)
    ✕ should return a single user (22 ms)

  ● API Tests › should return a single user
```

```
        expect(received).toEqual(expected) // deep equality

    Expected: 200
    Received: 404

    [0m [90m 12 |[39m   it([32m'should return a single user'[39m[33m,[39m [39m [36masync[39m () [33m=>[39m {
     [90m 13 |[39m     [36mconst[39m res [33m=[39m [39m [36mawait[39m request(app)[33m.[39m[39m[36mget[39m([32m'/users/3'[39m)[33m;[39m[39m
    [31m[1m>[22m[39m[90m 14 |[39m     expect(res[33m.[39mstatusCode)[33m.[39mtoEqual([35m200[39m)[33m;[39m
     [90m    |[39m                           [31m[1m^[22m[39m
     [90m 15 |[39m     expect(res[33m.[39mbody[33m.[39mname)[33m.[39mtoEqual([32m'Alice'[39m)[33m;[39m
     [90m 16 |[39m   })[33m;[39m
     [90m 17 |[39m })[33m;[39m[39m[0m

      at Object.toEqual (tests/app.test.js:14:28)

----------|---------|----------|---------|---------|-------------------
File      | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
----------|---------|----------|---------|---------|-------------------
All files |    92.3 |       50 |     100 |   91.66 |
 app.js   |    92.3 |       50 |     100 |   91.66 | 15
----------|---------|----------|---------|---------|-------------------
Test Suites: 1 failed, 1 total
Tests:       1 failed, 1 passed, 2 total
Snapshots:   0 total
Time:        1.89 s, estimated 2 s
Ran all test suites.
Force exiting Jest: Have you considered using `--detectOpenHandles` to detect async operations that kept running after all tests finished?
[Pipeline] error
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Declarative: Post Actions)
[Pipeline] echo
❌ El pipeline ha fallado
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
ERROR: ❌ Error en la etapa de Test
Finished: FAILURE
```