



Continuous Integration (CI)

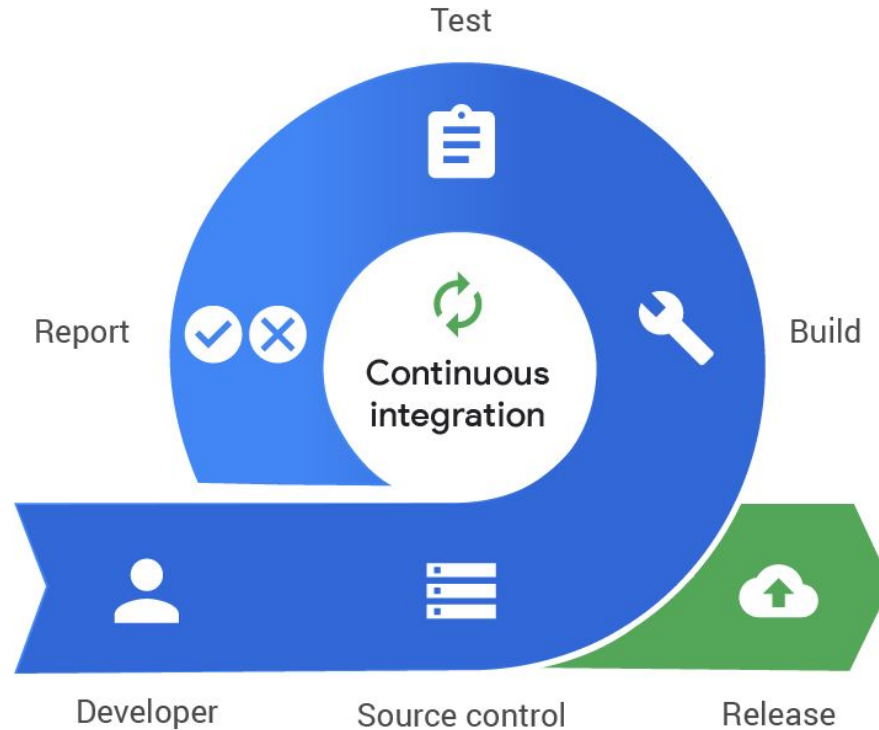
Alejandro García Perdomo (alu0101312101)
Jorge Cabrera Rodríguez (alu0101351773)

Tips to be covered

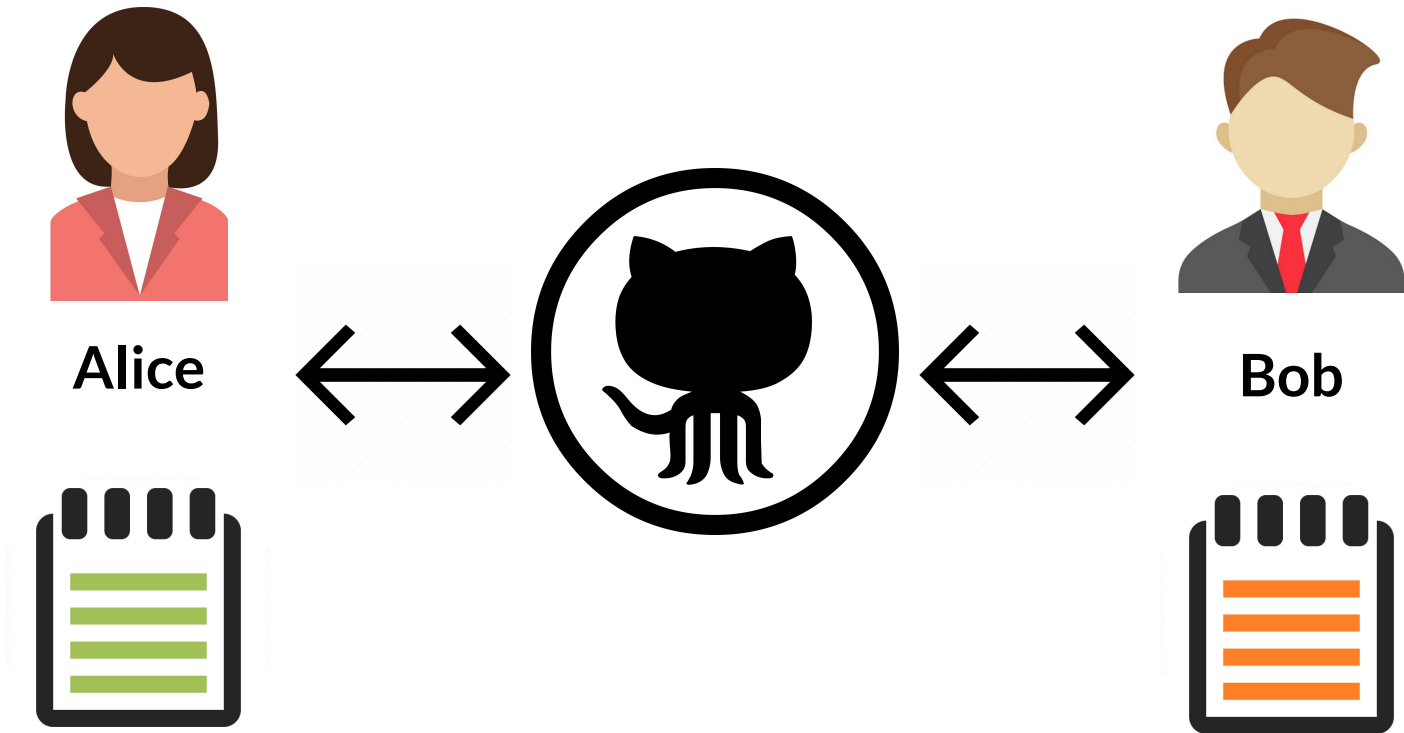


1. Introduction
2. What is CI
3. CI vs CD vs CDel
4. Phases of CI
5. CircleCI
 - a. Docker
 - b. Steps
 - c. Jobs
 - d. Workflow
 - e. Relevant features

A brief introduction to CI



An introduction example



Push frequently, save lives

In case of fire



1. git commit

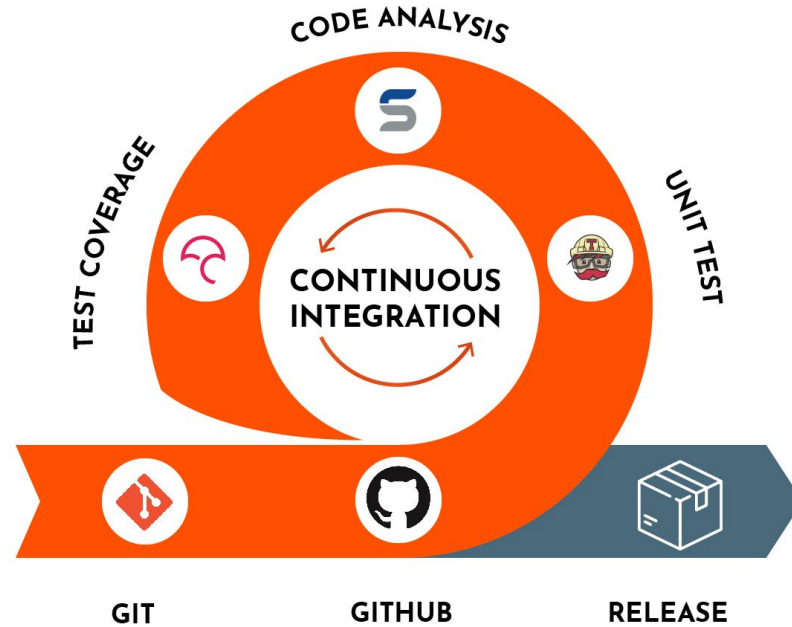


2. git push



3. leave building

Use CI!



What is CI?

Development practice where developers **integrate code** into a shared repository **frequently**.



Each integration can then be verified by an **automated build** and **automated tests**.



Advantages and Disadvantages




Advantages

- Merging is much easier.
- There is always executable code.
- Bugs are easier to find.
- More communication between team members.

Disadvantages

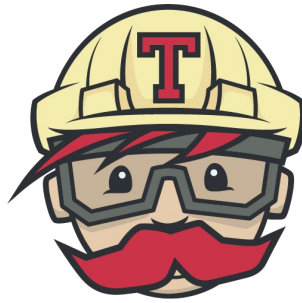
- May be slower.
- Needs a lot of push and pull actions.



“Continuous Integration doesn’t get rid of bugs, but it does make them dramatically easier to find and remove.”

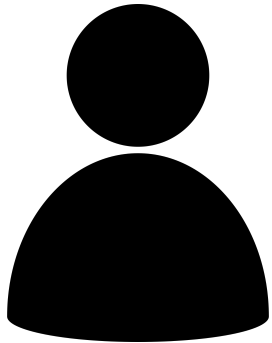
-Martin Fowler, software developer-

CI is independent of software



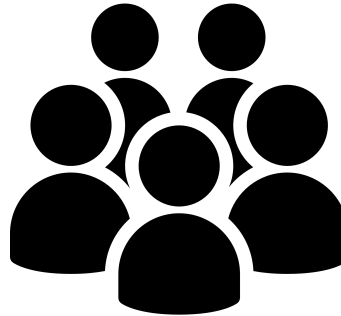
Practical application of CI

Small projects




- Might help with development
- Very situational
- Not required

Big projects



- Mandatory for teamwork
- Requires an administrator
- Make the projects success

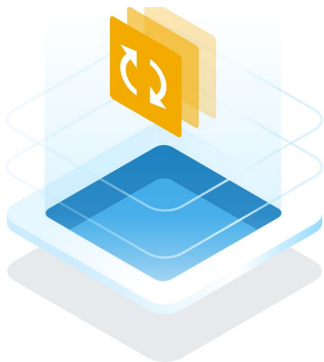


Continuous Integration VS Continuous Deployment VS Continuous Delivery

Differences between each model

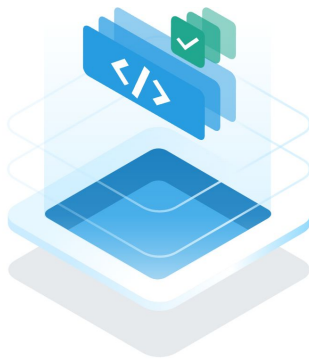
Continuous Integration (CI)

Integrating changes into a mainline as early as possible.



Continuous Deployment (CD)

deployment of the master branch to a production environment following automated testing.

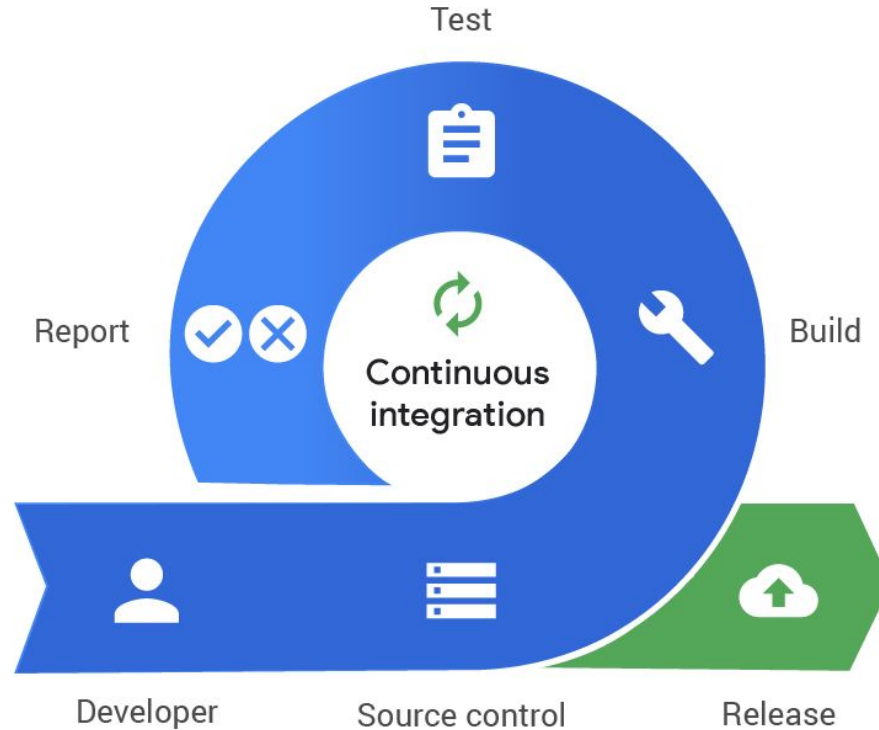


Continuous Delivery (CD)

being able to reliably release application changes (code) at any time.



Phases of CI





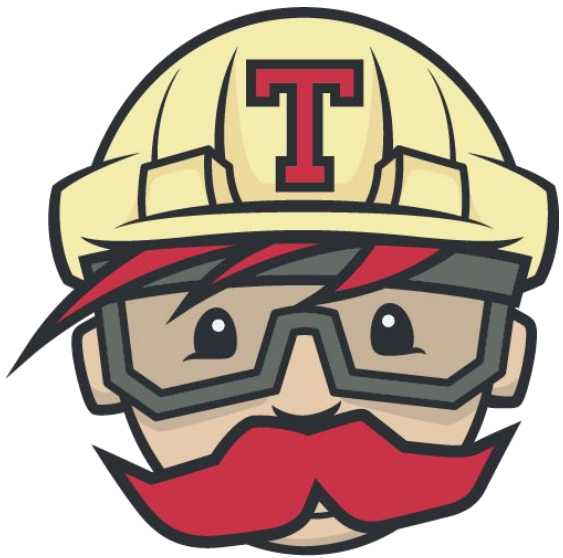
Tools for CI



GitHub Actions



Jenkins



Travis CI

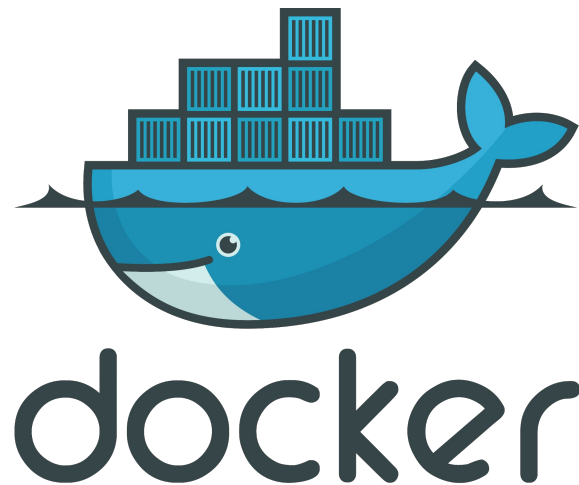


circleci



Basics of CircleCI

- Based on Docker
- Allow us to make automatic tasks in our project
- Easy to use
- Extremely documented
- Activated in PAI organization



Setting up a repo

ULL-ESIT-PAI-...
Jorge Cabrera Rodríg...

Dashboard

Projects

Insights

Organization Settings

Plan

Can't find an organization?
[Check permissions](#) and update access to the ones you want.

Notifications 2

Status **OPERATIONAL**

Set up new projects or follow projects already building on CircleCI.
Following a project adds it to your dashboard.

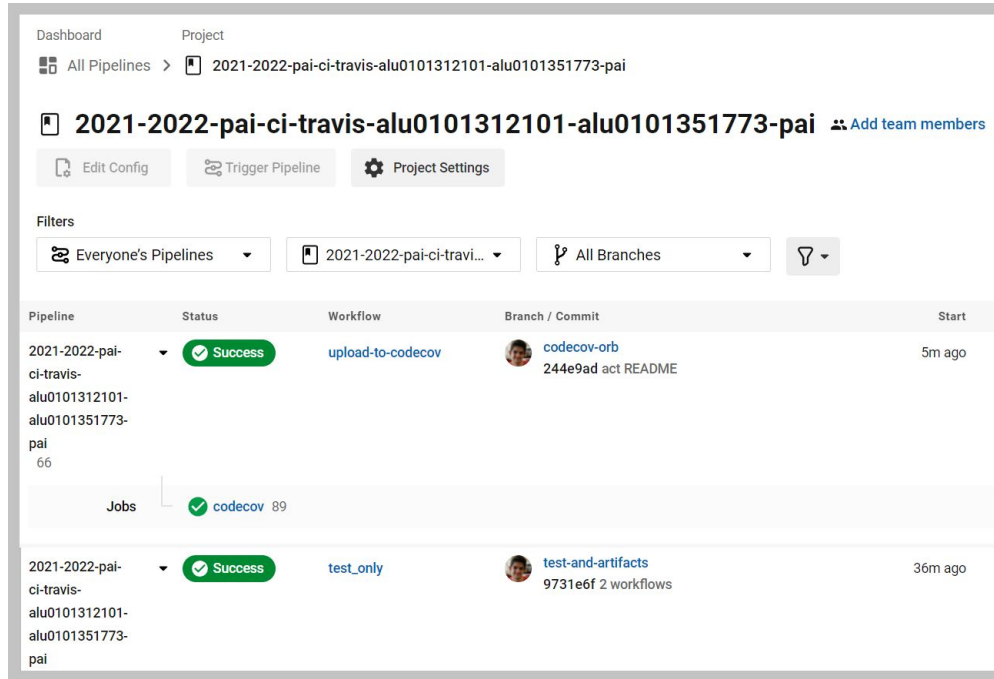
Projects

Repo name

Follow All ...

Repo	
2021-2022-pai-ci-travis-alu0101312101-alu0101351773-pai	Unfollow Project ...
2021-2022-pai-p05-debugging-exercism-alu0101351773	Set Up Project ...
2021-2022-pai-p06-files-json-alu0101351773	Set Up Project ...
2021-2022-pai-p04-unit-testing-exercism-alu0101351773	Set Up Project ...
2021-2022-pai-p02-simpleprograms2-exercism-alu0101351773	Set Up Project ...

Main page of our CI



The screenshot displays the Jenkins CI dashboard for a project named "2021-2022-pai-ci-travis-alu0101312101-alu0101351773-pai". The interface includes a top navigation bar with "Dashboard" and "Project" tabs. Below the navigation bar, there are buttons for "Edit Config", "Trigger Pipeline", and "Project Settings". A "Filters" section allows users to select "Everyone's Pipelines", the specific project, and "All Branches". The main content area shows a table of pipeline runs with columns for Pipeline, Status, Workflow, Branch / Commit, and Start time. The first pipeline run is "2021-2022-pai-ci-travis-alu0101312101-alu0101351773-pai" with a "Success" status, workflow "upload-to-codecov", and started 5m ago. Below the pipeline table, there is a "Jobs" section showing a "Success" status for "codecov" with 89 jobs. The second pipeline run is "2021-2022-pai-ci-travis-alu0101312101-alu0101351773-pai" with a "Success" status, workflow "test_only", and started 36m ago.

Dashboard Project

All Pipelines > 2021-2022-pai-ci-travis-alu0101312101-alu0101351773-pai

2021-2022-pai-ci-travis-alu0101312101-alu0101351773-pai Add team members

Edit Config Trigger Pipeline Project Settings

Filters

Everyone's Pipelines 2021-2022-pai-ci-travi... All Branches

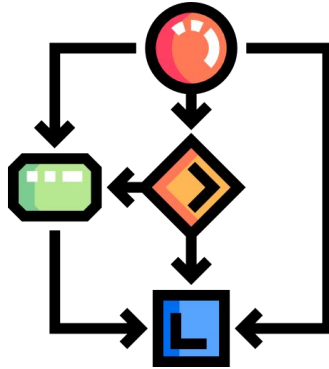
Pipeline	Status	Workflow	Branch / Commit	Start
2021-2022-pai-ci-travis-alu0101312101-alu0101351773-pai 66	Success	upload-to-codecov	codecov-orb 244e9ad act README	5m ago
Jobs				
Success codecov 89				
2021-2022-pai-ci-travis-alu0101312101-alu0101351773-pai	Success	test_only	test-and-artifacts 9731e6f 2 workflows	36m ago

CI basic concepts

Pipelines



Workflows



Jobs

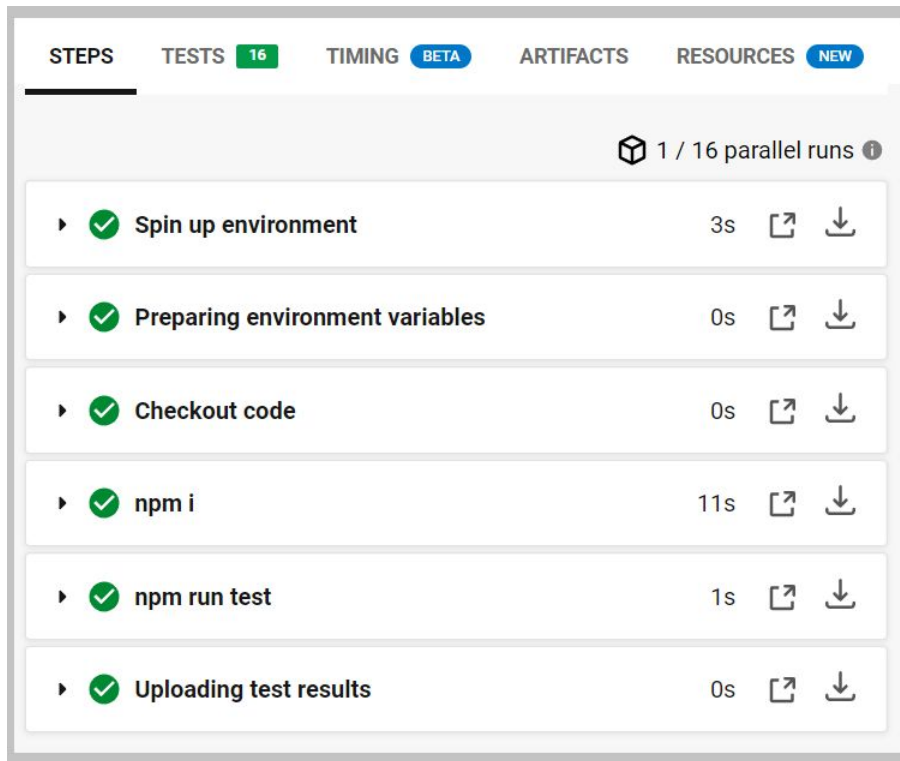


Steps



Steps

- Command of the system
- Allow us to run tasks in our virtual machine








The screenshot displays a CI/CD interface with a top navigation bar containing tabs: STEPS, TESTS (16), TIMING (BETA), ARTIFACTS, and RESOURCES (NEW). Below the tabs, a header indicates '1 / 16 parallel runs' with an information icon. The main content area lists six steps, each with a green checkmark, a right-pointing arrow, the step name, a duration, and icons for opening and downloading logs.

Step	Duration
Spin up environment	3s
Preparing environment variables	0s
Checkout code	0s
npm i	11s
npm run test	1s
Uploading test results	0s

Jobs

- Set of steps


 **test_only** ✓ Success

Duration / Finished	Branch	Commit	Author & Message
 21s / 8m ago	 test-and-artifacts	 9731e6f	 2 workflows

✓ tests 18s

Workflows

- Structure for managing jobs
- Allow us to manage which jobs we want to be runned

Pipeline	Status	Workflow
2021-2022-pai- ci-travis- alu0101312101- alu0101351773- pai 64	▼  Success	build_and_artifacts

Each time we run a job...





Very important!

All steps in the
same job are
executed
sequentially.

All jobs in the
same workflow
are executed
parallelly.



**Let's see our first
automatization!**

Relevant features

Test and Documentation



Relevant features

Artifacts


Duration / Finished


Queued



Executor / Resource Class


Branch


Commit

 14s / 2d ago


 0s

 Docker / [Large](#) 

 [test-and-artifacts](#)

 [dacfbe1](#)

Author & Message

 Update valid.spec.js


STEPS

TESTS

TIMING BETA

ARTIFACTS

RESOURCES NEW

▼  Parallel Run 0

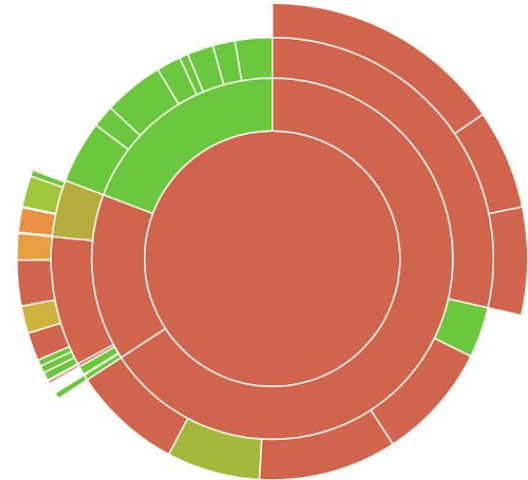
[docs/global.html](#)

[docs/index.html](#)

[docs/valid_dates.js.html](#)

Relevant features

Code coverage



Other interesting features

Environment Variables

```
jobs:
  build-job:
    docker:
      - image: cimg/base:2020.01
    environment:
      LITTLE_CORVETTE: red
```

Context

Context: my-context

\$MY_ENV_VAR = myvalue

```
steps: # use env var from Context
  - run: echo $MY_ENV_VAR
```

Other interesting features

Multiple Branches Execution

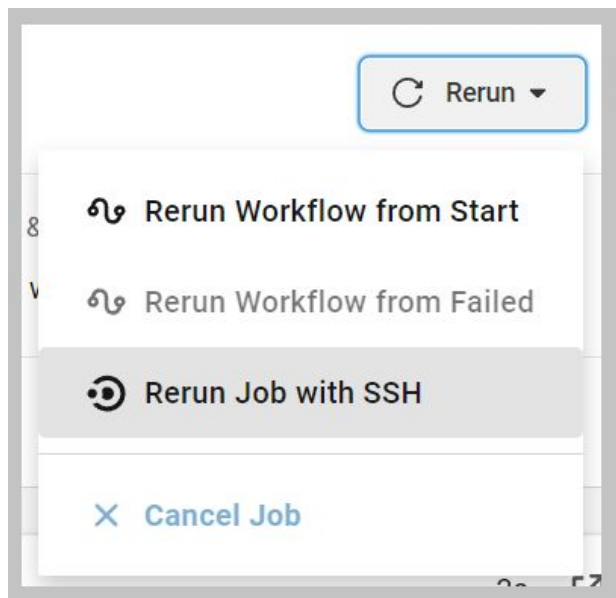
```
filters:
  branches:
    only:
      - dev
      - /user-.* /
```

Job approval

```
- hold:
    type: approval
    requires:
      - build
      - test
- deploy:
    requires:
      - hold
```

Other interesting features

SHH



Schedules

```
nightly:
  triggers:
    - schedule:
        cron: "0 0 * * *"
        filters:
```



Our personal opinion

Bibliografía



- YT Video - [What is Continuous Integration?](#)
- Wikipedia - [Continuous Integration definition](#)
- [Another definition of CI and its uses](#)
- [CircleCI](#)
- [CircleCI getting started](#)
- [14 Examples of CI tools](#)

Bibliografía (CircleCI)



- Workflows
- Orbs
- Docker
- Store test results
- Artifacts
- Code Coverage



Questions ?

alejandro.garcia.perdomo.18@ull.edu.es

cabrera.rodriguez.12@ull.edu.es