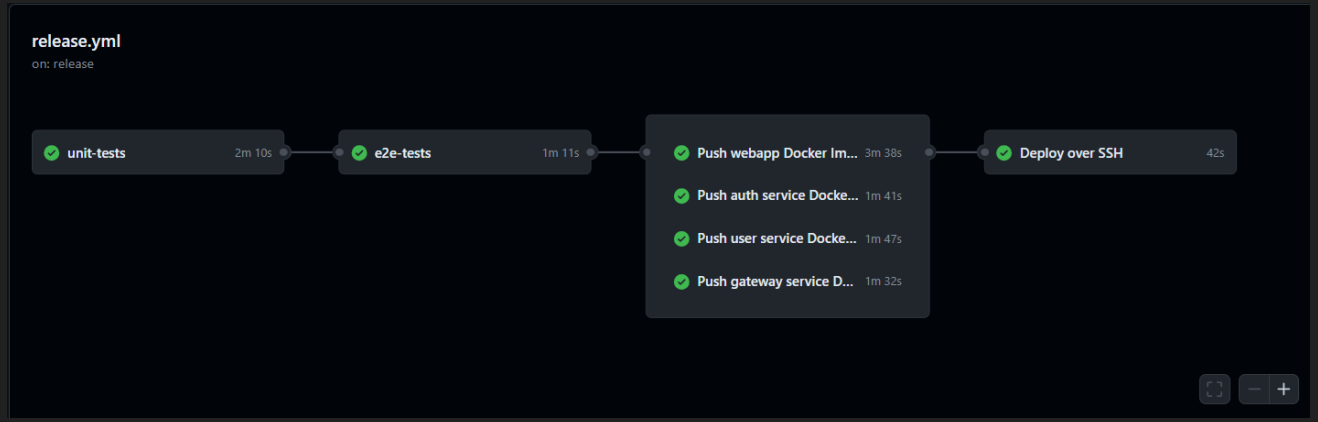
Github Actions

Darío Gutiérrez Mori (UO282435)
Diego Villanueva Berros (UO283615)
Diego Murias Suárez (UO290009)
Jorge Cano Martínez (UO289845)

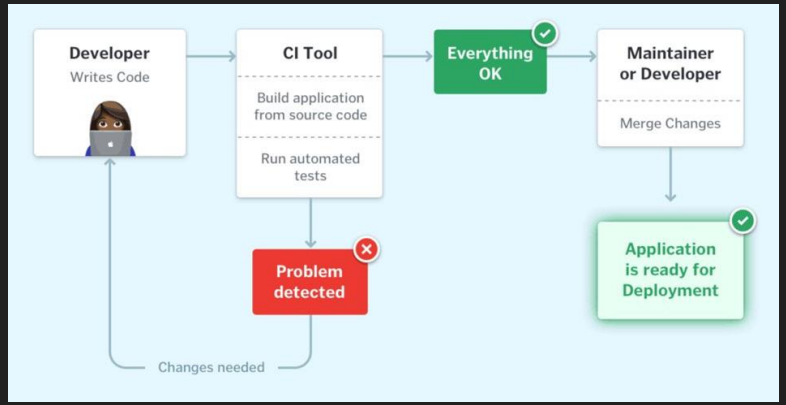
What are Github Actions?

- Customizable workflows that allow you to automate tasks within your GitHub repository
- GitHub Actions serves as an ideal solution for CI and CD.
- Configure workflows to automatically run tasks triggered by some events.
- Automate unit tests, integration tests, artifact building, and more for seamless integration of code changes.



Continuous Integration

- CI is essential for efficient software development.
- Without CI, code changes can lead to fragmentation and conflicts.
- CI, enabled via GitHub Actions, automates workflows triggered by events.
- Tasks include running tests and deploying the application.
- Early bug detection reduces time and effort for fixes.
- CI promotes collaboration and trust through automated testing.
- Recommending the use of dependabot



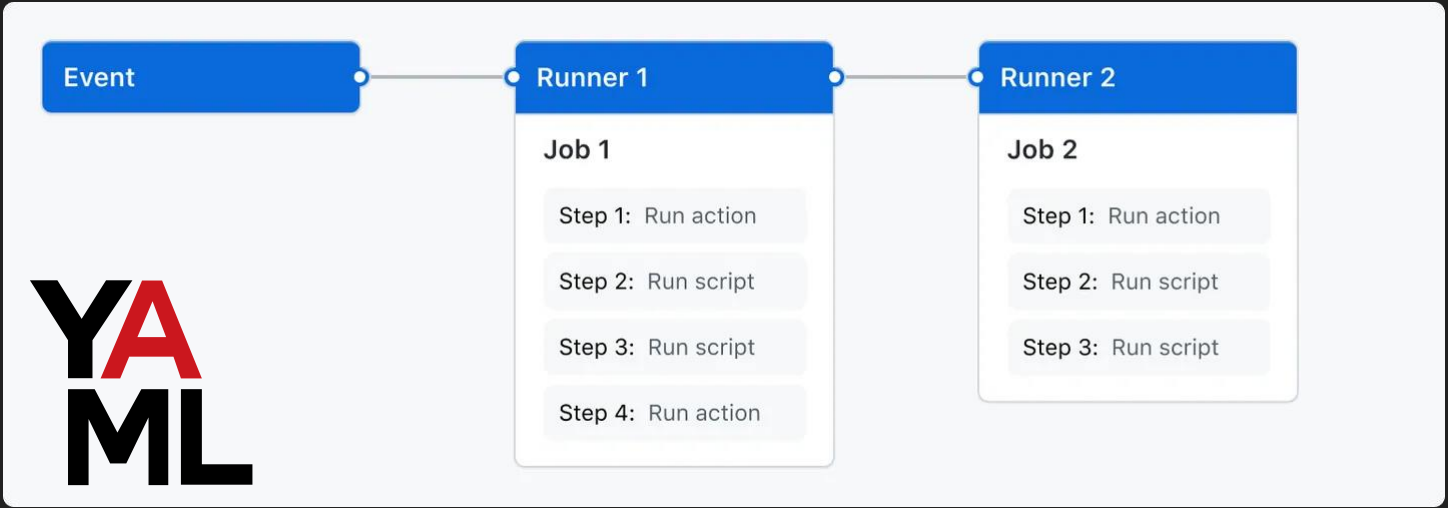
Continuous Deployment

Workflows:

Automated process that will run one or more jobs.

Jobs:

Set of steps in a workflow that is executed on the same runner



Continuous Deployment

Events:

A specific activity in a repository that triggers a workflow run.

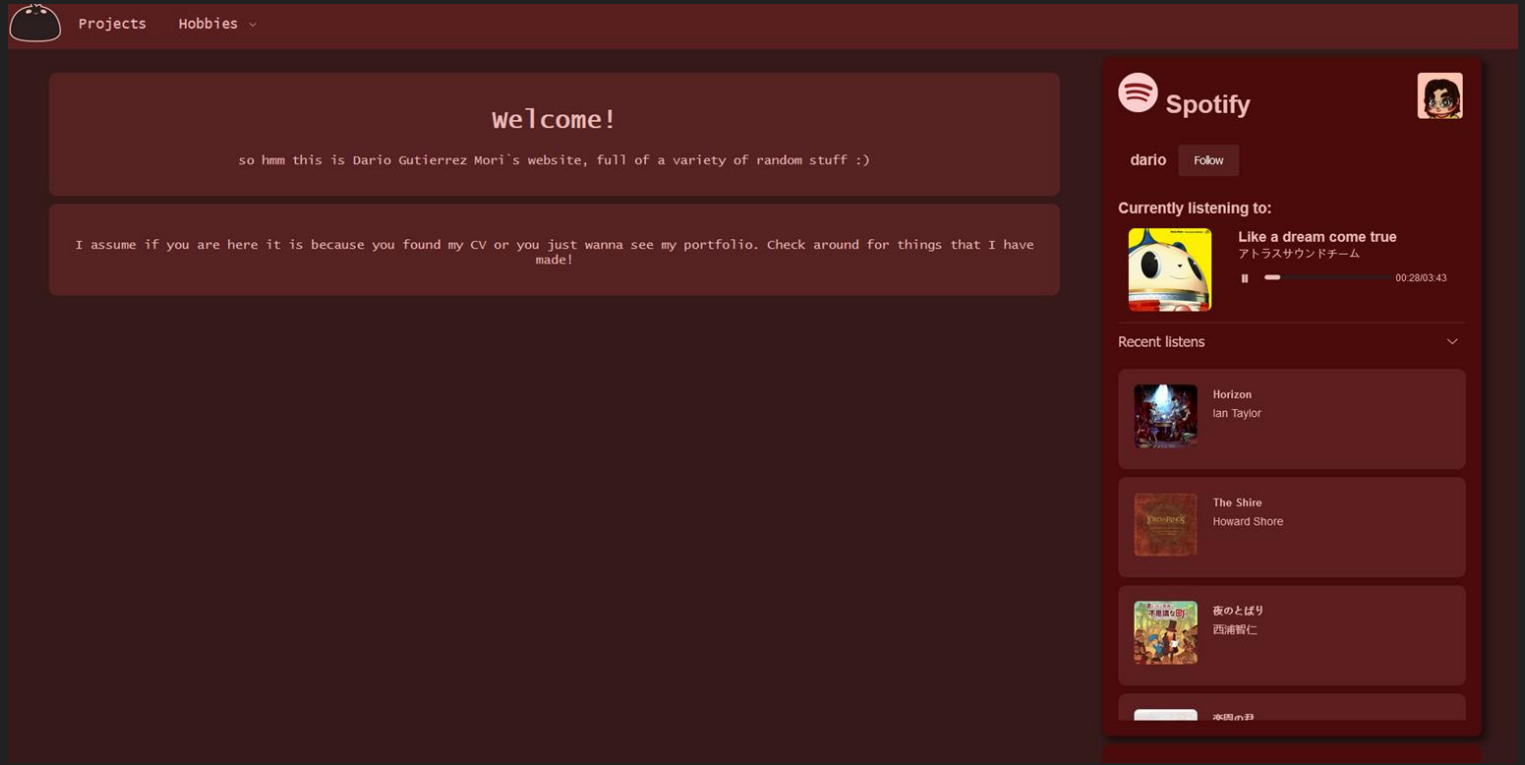
- Pull request
- Pushing a commit
- Opening an issue

Basically, anything that can happen in the repo

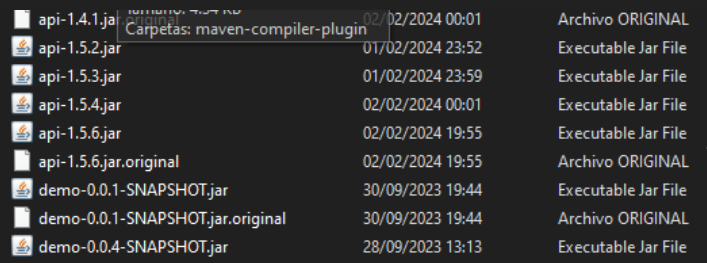
YAML

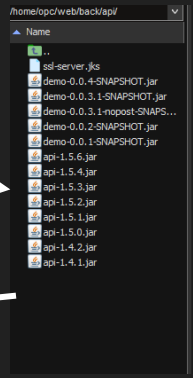
Ok, but why should I care about this?
(and how do I use it in my projects)

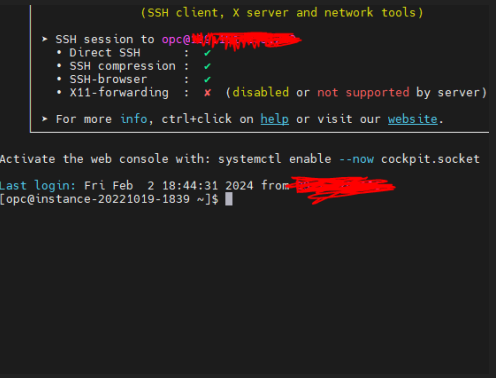
A practical example

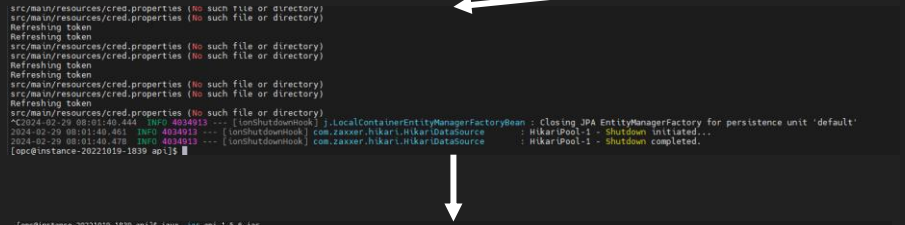


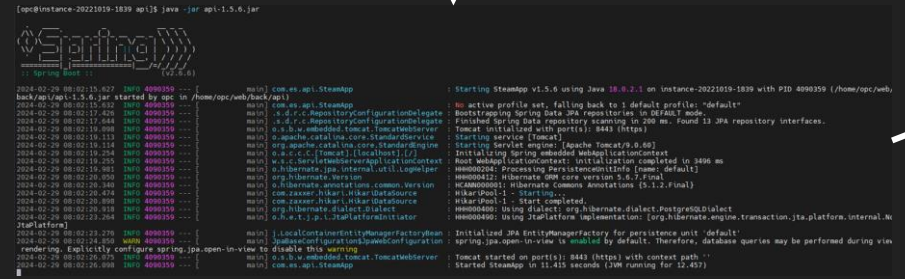
What I had (and still have) to do:



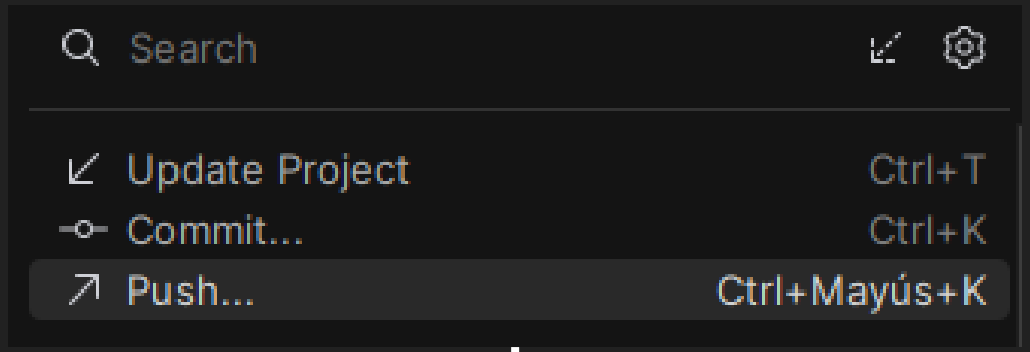


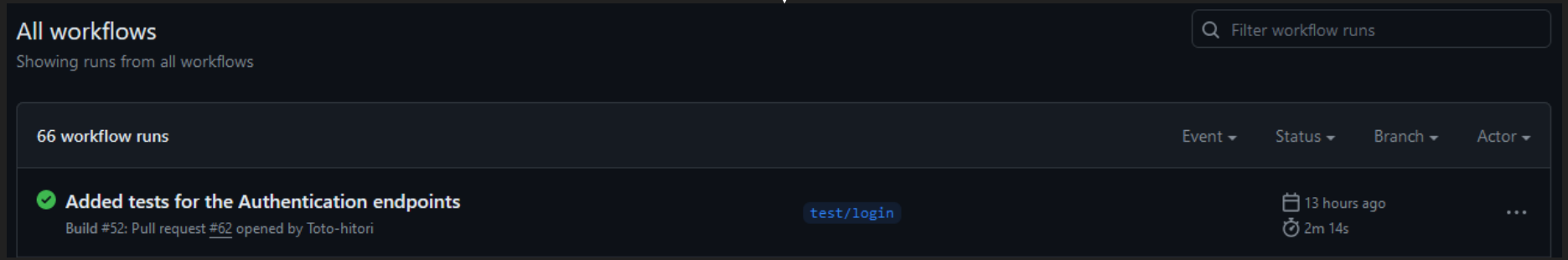






But if I had Github actions setup…





(and I would have tests!)

Ok but, where is the catch?

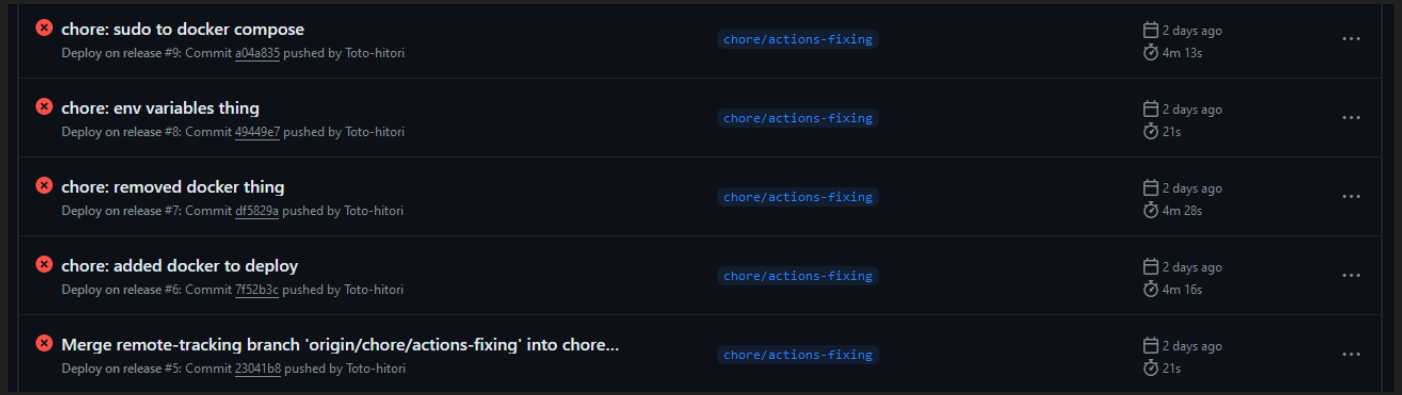
(here it is :/)

Plan	Storage	Minutos (por mes)
GitHub Free	500 MB	2.000
GitHub Pro	1 GB	3,000
GitHub Free para organizaciones	500 MB	2.000
GitHub Team	2 GB	3,000
GitHub Enterprise Cloud	50 GB	50.000

Tasas por minuto

Sistema operativo	vCPU	Tasa por minuto (USD)
Linux	2	0,008 USD

you have around 50 hours so don’t worry!



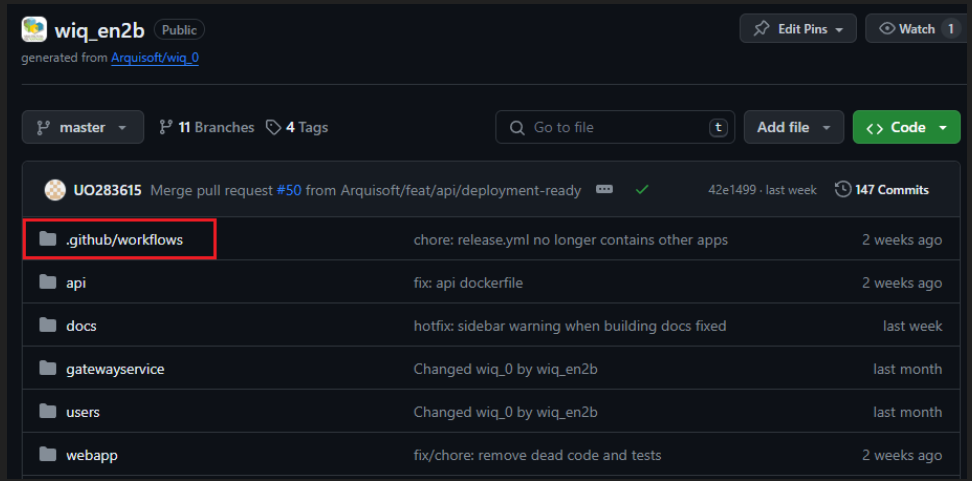
Where could I deploy my applications?

OR

ORACLE

Data Cloud

How are actions defined?



build.yml

```
name: Build
on:
  push:
    branches:
      - master
  pull_request:
    types: [opened, synchronize, reopened]
jobs:
  unit-tests:
    runs-on: ubuntu-latest
    steps:
    - uses: actions/checkout@v4
        with:
          fetch-depth: 0
    - uses: actions/setup-node@v4
      with:
        node-version: 20
    - run: npm --prefix users/authservice ci
    - run: npm --prefix users/authservice test --
--coverage
    - name: Analyze with SonarCloud
      uses: sonarsource/sonarcloud-github-
action@master
      env:
        GITHUB_TOKEN: ${{ secrets.GITHUB_TOKEN }}
        SONAR_TOKEN: ${{ secrets.SONAR_TOKEN }}
```

Some keywords

name: The identifier for your action

inputs: Data that will be used in runtime

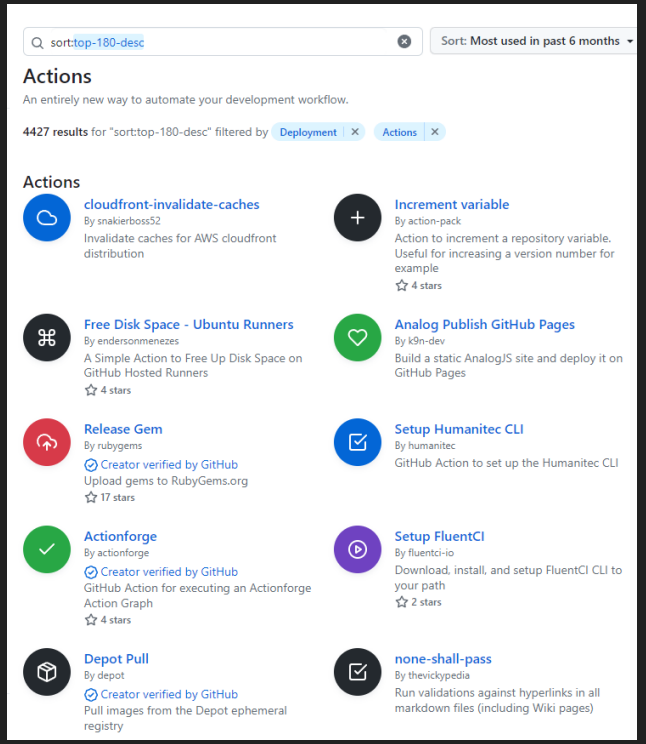
outputs: Data that an actions sets. **Can be used by following actions**

runs: Specifies if it is a JavaScript, Docker or composite action.

jobs: What will be done

Actions Marketplace

Someone has *probably* done that before



Thanks