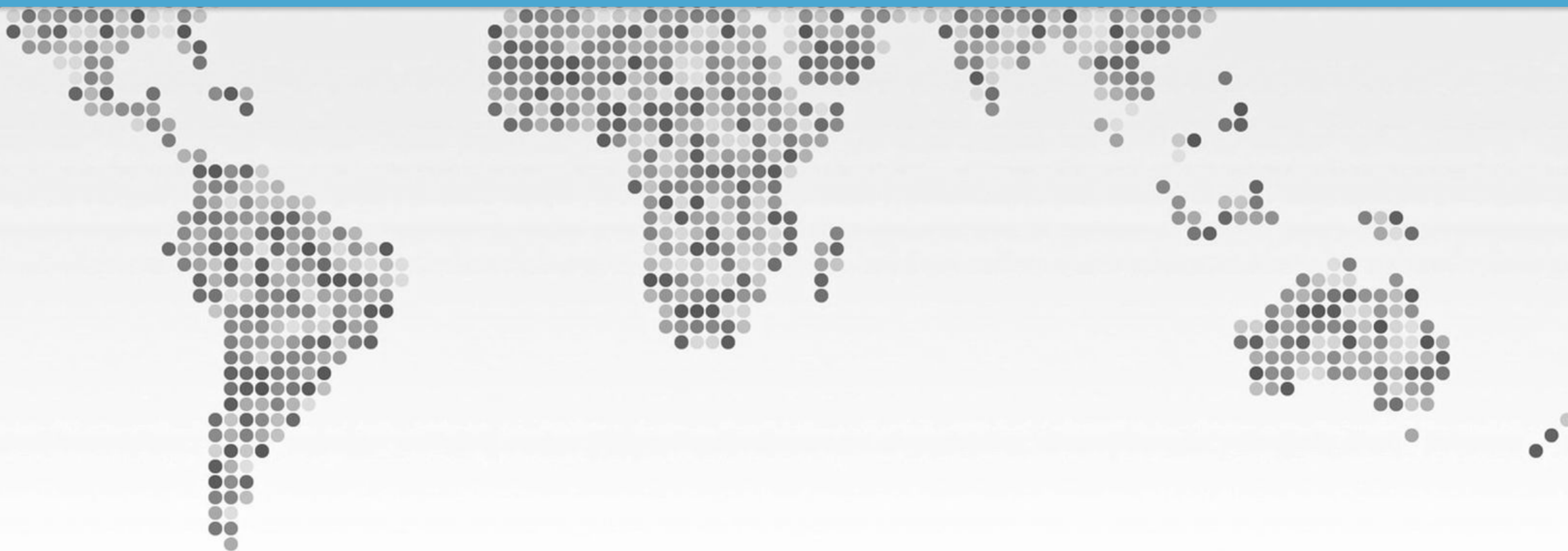


Continuous Integration & Continuous Delivery

GitHub Actions + Travis CI

Laboratorio di Ingegneria del Software



Continuous Integration - Che cos'è CI/CD

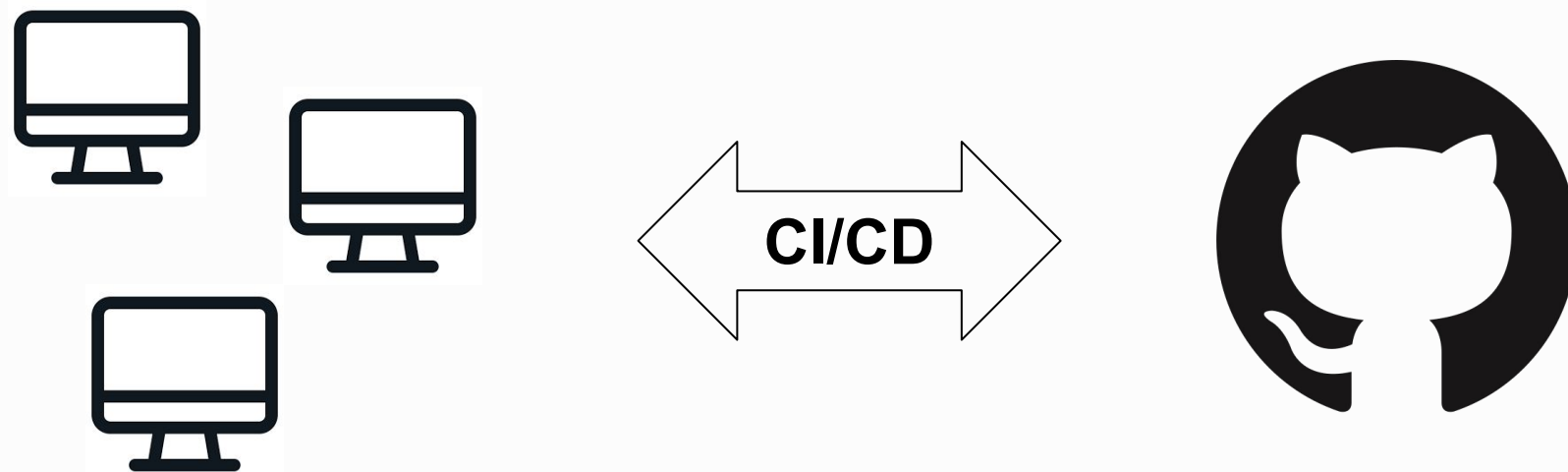
CI/CD è un approccio per lo sviluppo di software, finalizzato all'automazione delle procedure che portano il codice dallo sviluppo all'integrazione, dal test alla distribuzione e deployment finale.

CI fa riferimento alla metodologia di Continuous Integration.

CD a quella di Continuous Delivery.

Le modifiche al software vengono compilate, testate e unite in un repository condiviso (Continuous Integration), resolvendo così il problema dei conflitti tra le numerose diramazioni di un'applicazione in fase di sviluppo.

Con Continuous Delivery si intende il processo con il quale le modifiche apportate da uno sviluppatore vengono automaticamente testate, caricate in un repository ed, infine, distribuite al repository di produzione dove diventano fruibili ai clienti.

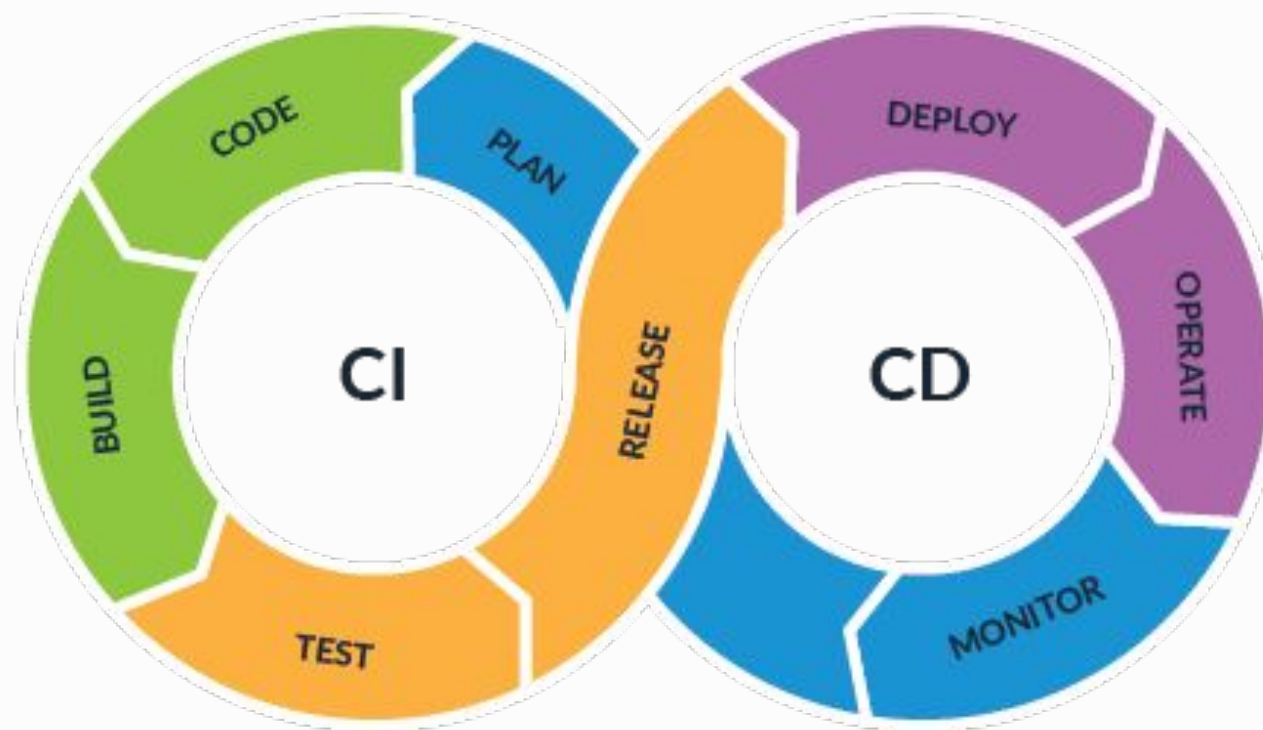


Continuous Integration – Pipeline di CI/CD

*Una **pipeline CI/CD** è un insieme di attività (spesso sequenziali) atte a garantire che i principi della CI/CD siano applicati ad un progetto software. Ogni attività costituisce una fase di pipeline.*

Le fasi di pipeline tipiche includono:

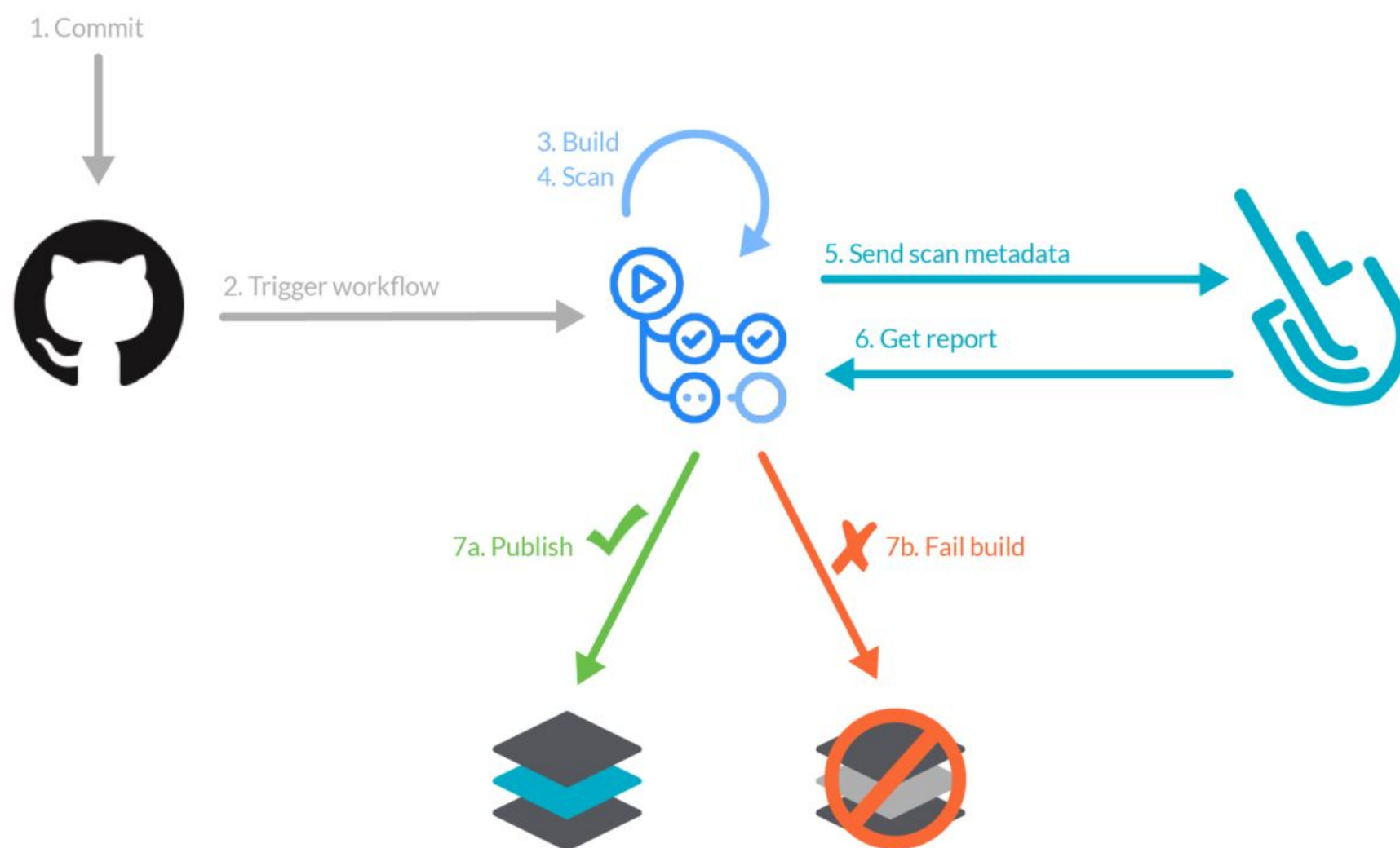
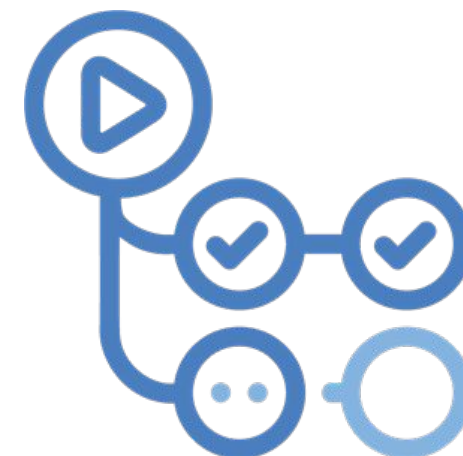
- **Build:** la fase di compilazione dell'applicazione.
- **Test:** la fase in cui il codice viene testato.
- **Rilascio:** la fase in cui l'applicazione viene fornita al repository.
- **Deployment:** in questa fase il codice viene distribuito al reparto di produzione.



Tool per CI/CD – GitHub Actions

GitHub Actions è una piattaforma di CI/CD che consente di automatizzare la pipeline di creazione, test e distribuzione.

È possibile creare workflows che eseguono azioni in risposta ad un evento, ad esempio una pull request o un push, o distribuiscono le pull request all'ambiente di produzione.



Prerequisiti

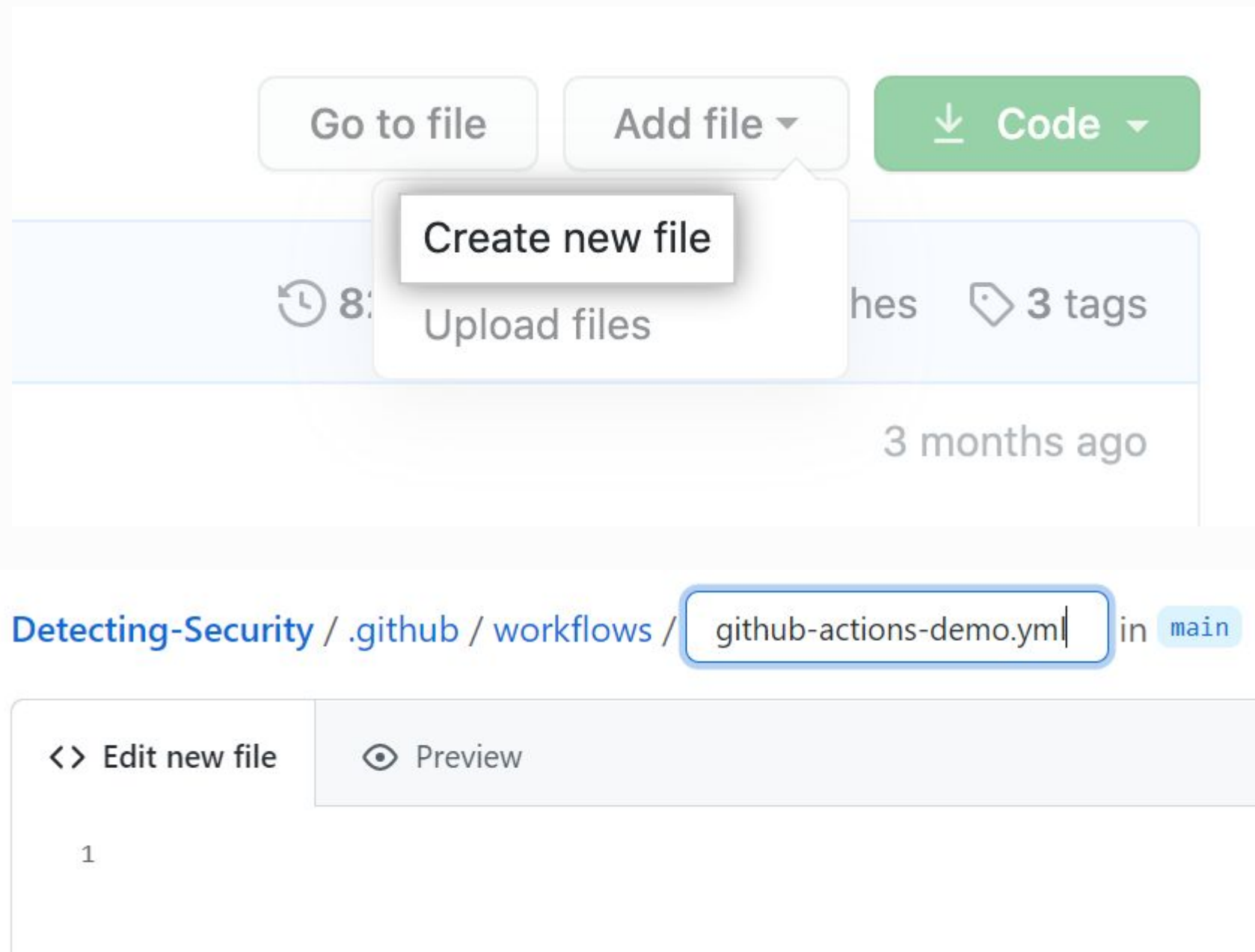
- Disporre di un repository GitHub.

Risorse utili

- Link a GitHub Actions: <https://docs.github.com/en/actions/>
- Link al *getting started* di GitHub Actions: <https://docs.github.com/en/actions/quickstart>

Tool per CI/CD – GitHub Actions – Setup

- 1 Creare una directory nel tuo repository GitHub e nominarla `.github/workflows`
- 2 Nella directory `.github/workflows`, creare un file chiamato `github-actions-demo.yml`



Tool per CI/CD – GitHub Actions – Setup

3

Inserire la specifica YAML nel file *github-actions-demo.yml*

Detecting-Security / .github / workflows / github-actions-demo.yml

Cancel changes

<> Edit new file

Preview

Spaces

2

No wrap

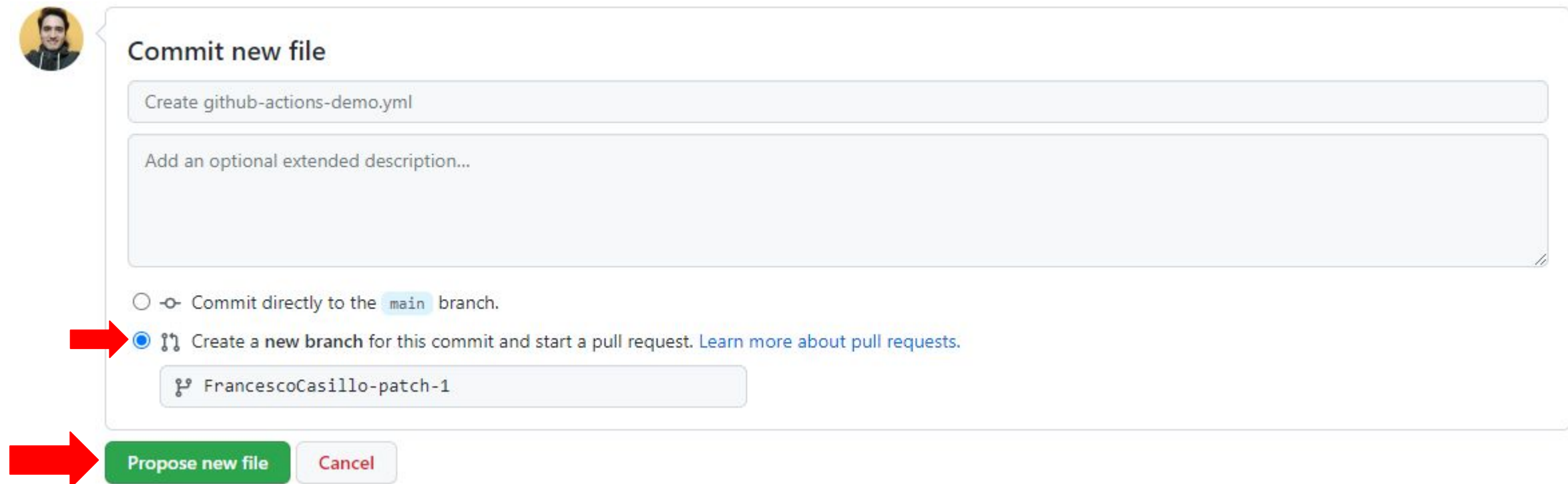
```
1 name: GitHub Actions Demo
2 run-name: ${{ github.actor }} is testing out GitHub Actions 🚀
3 on: [push]
4 jobs:
5   Explore-GitHub-Actions:
6     runs-on: ubuntu-latest
7     steps:
8       - run: echo "🎉 The job was automatically triggered by a ${{ github.event_name }} event."
9       - run: echo "👋 This job is now running on a ${{ runner.os }} server hosted by GitHub!"
10      - run: echo "🔍 The name of your branch is ${{ github.ref }} and your repository is ${{ github.repository }}."
11      - name: Check out repository code
12        uses: actions/checkout@v3
13      - run: echo "💡 The ${{ github.repository }} repository has been cloned to the runner."
14      - run: echo "🖨 The workflow is now ready to test your code on the runner."
15      - name: List files in the repository
16        run: |
17          ls ${{ github.workspace }}
18      - run: echo "🍏 This job's status is ${{ job.status }}."
```

Esempio

Tool per CI/CD – GitHub Actions – Setup

4

Scorrere fino alla fine della pagina e selezionare *Create a new branch for this commit and start a pull request*.
Quindi, per creare una richiesta di pull, fare clic su *Propose new file*.



Commit new file

Create github-actions-demo.yml

Add an optional extended description...

☐ Commit directly to the `main` branch.

☒ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

FrancescoCasillo-patch-1

Propose new file Cancel

**Il commit di un file in un branch del repository attiva(*trigger*)
l'evento *push* ed esegue il flusso di lavoro.**

Tool per CI/CD – GitHub Actions – Setup

5 Accettata la richiesta di pull, nella sezione *Actions* possiamo controllare l'esito dell'operazione, con l'esecuzione degli step indicati nel file YAML

FrancescoCasillo / Detecting-Security Public

<> Code Issues Pull requests **Actions** Projects Wiki Security Insights Settings

FrancescoCasillo is testing out GitHub Actions GitHub Actions Demo #1 Re-run all jobs

Summary

Jobs

Explore-GitHub-Actions

Explore-GitHub-Actions
succeeded 6 hours ago in 8s

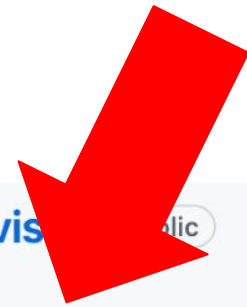
Search logs

- > Set up job 1s
- > Run echo "🚀 The job was automatically triggered by a push event." 0s
- > Run echo "📁 This job is now running on a Linux server hosted by GitHub!" 0s
- > Run echo "🌐 The name of your branch is refs/heads/main and your repository is FrancescoCasillo/Detecting-Security." 0s
- > Check out repository code 0s
- > Run echo "💡 The FrancescoCasillo/Detecting-Security repository has been cloned to the runner." 0s
- > Run echo "💻 The workflow is now ready to test your code on the runner." 0s
- > List files in the repository 0s
- > Run echo "🍏 This job's status is success." 0s
- > Post Check out repository code 0s
- > Complete job 0s

Tool per CI/CD – GitHub Actions – Setup Alternativo

1 Dalla repository GitHub, andare nella tab *Actions*

Flusso Alternativo



StefanoLambiasi / is2_tutorato_travis

<> Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

main

1 branch

0 tags

Go to file

Add file

Code



StefanoLambiasi Update maven.yml

✓ 7060dec 8 minutes ago 4 commits



.github/workflows

Update maven.yml

8 minutes ago



.idea

Initial commit

3 hours ago



src/main/java

Initial commit

3 hours ago



.gitignore

Initial commit

3 hours ago



.travis.yml

Create .travis.yml

2 hours ago



pom.xml

Initial commit

3 hours ago


Help people interested in this repository understand your project by adding a README.

Add a README

Tool per CI/CD – GitHub Actions – Setup Alternativo

2

Selezionare un template di workflows (nel nostro caso, quello Maven)

 StefanoLambiasi / is2_tutorato_travisCI Public

Pin Unwatch 1

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

Get started with GitHub Actions

Build, test, and deploy your code. Make code reviews, branch management, and issue triaging work the way you want. Select a workflow to get started.

Skip this and [set up a workflow yourself](#) →

Suggested for this repository

SLSA Generic generator
By Open Source Security Foundation (OpenSSF)
Generate SLSA3 provenance for your existing release workflows
[Configure](#) Go

Publish Java Package with Maven
By GitHub Actions
Build a Java Package using Maven and publish to GitHub Packages.
[Configure](#) Java

Java with Maven
By GitHub Actions
Build and test a Java project with Apache Maven.
[Configure](#) Java

Android CI
By GitHub Actions
Build an Android project with Gradle.
[Configure](#) Java

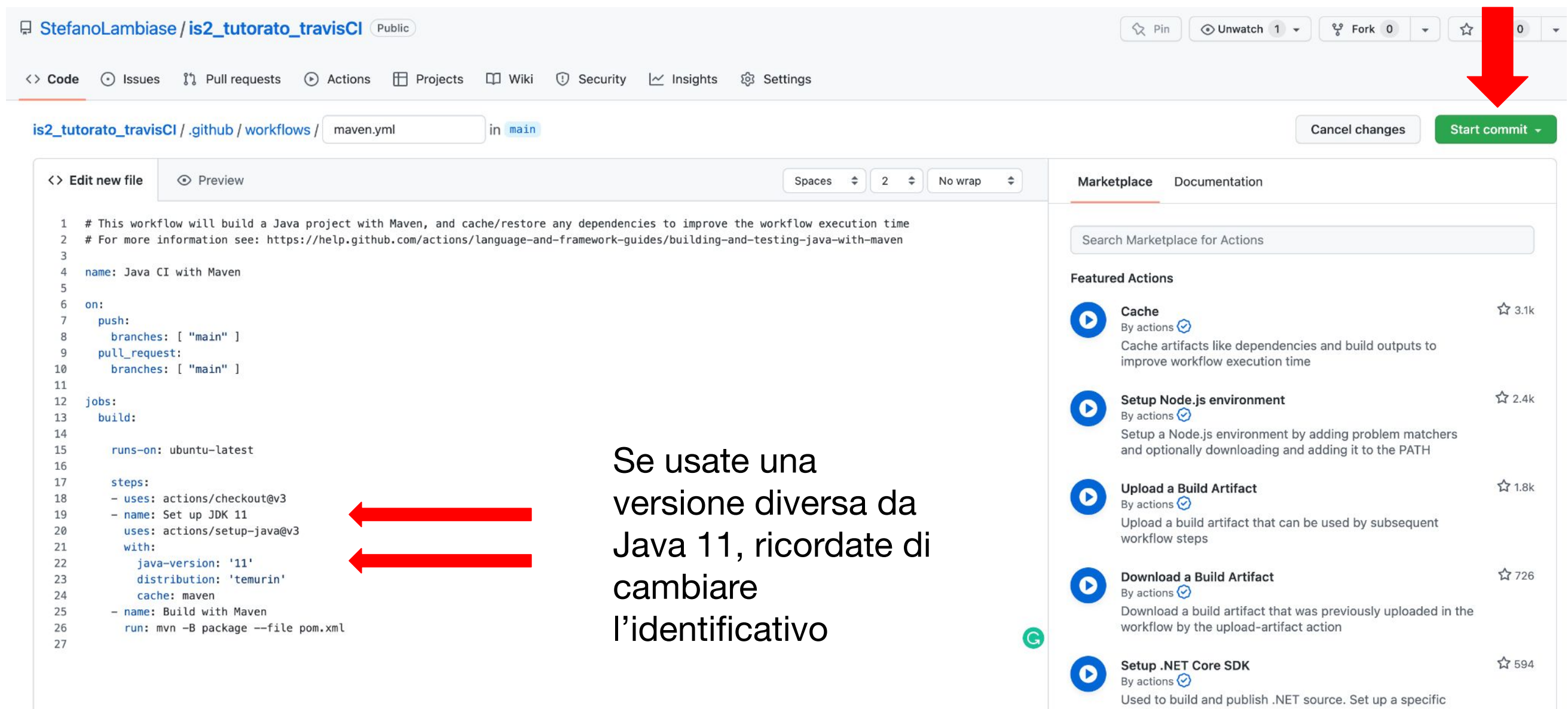
Java with Ant
By GitHub Actions
Build and test a Java project with Apache Ant.
[Configure](#) Java

Clojure
By GitHub Actions
Build and test a Clojure project with Leiningen.
[Configure](#) Clojure

Tool per CI/CD – GitHub Actions – Setup Alternativo

3

Fare il commit del file YAML



StefanoLambiasi / is2_tutorato_travisCI Public

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

is2_tutorato_travisCI / .github / workflows / maven.yml in main

Cancel changes Start commit

```
1 # This workflow will build a Java project with Maven, and cache/restore any dependencies to improve the workflow execution time
2 # For more information see: https://help.github.com/actions/language-and-framework-guides/building-and-testing-java-with-maven
3
4 name: Java CI with Maven
5
6 on:
7   push:
8     branches: [ "main" ]
9   pull_request:
10    branches: [ "main" ]
11
12 jobs:
13   build:
14
15     runs-on: ubuntu-latest
16
17     steps:
18     - uses: actions/checkout@v3
19     - name: Set up JDK 11
20       uses: actions/setup-java@v3
21       with:
22         java-version: '11'
23         distribution: 'temurin'
24         cache: maven
25     - name: Build with Maven
26       run: mvn -B package --file pom.xml
27
```

Marketplace Documentation

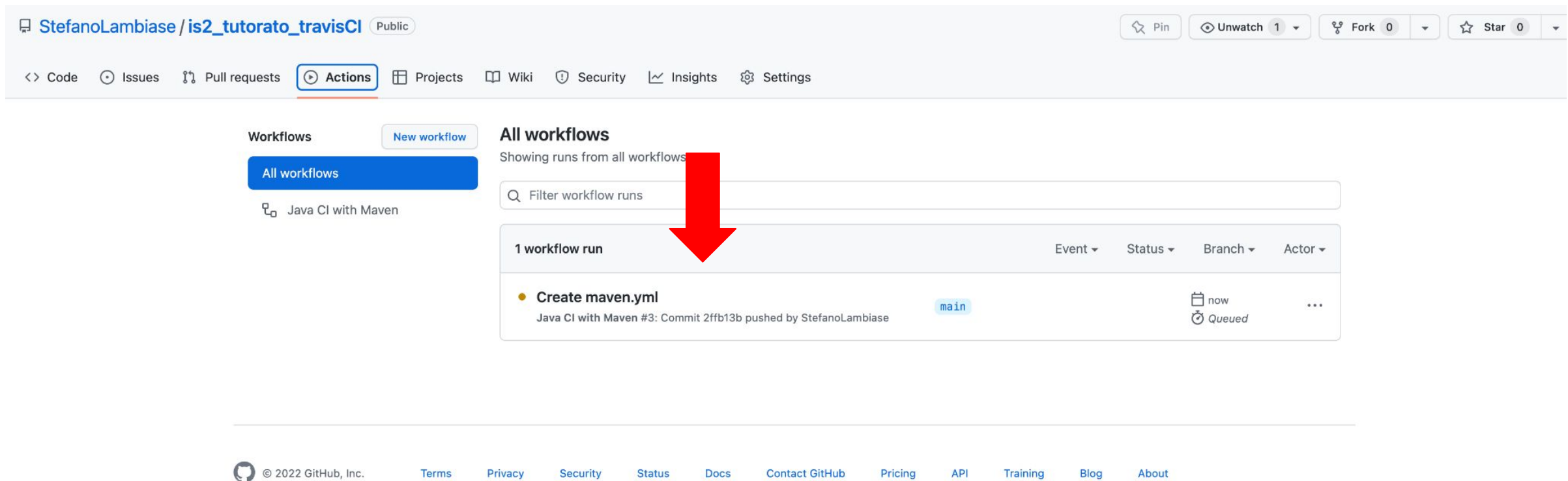
Search Marketplace for Actions

Featured Actions

- Cache** 3.1k
By actions ✓
Cache artifacts like dependencies and build outputs to improve workflow execution time
- Setup Node.js environment** 2.4k
By actions ✓
Setup a Node.js environment by adding problem matchers and optionally downloading and adding it to the PATH
- Upload a Build Artifact** 1.8k
By actions ✓
Upload a build artifact that can be used by subsequent workflow steps
- Download a Build Artifact** 726
By actions ✓
Download a build artifact that was previously uploaded in the workflow by the upload-artifact action
- Setup .NET Core SDK** 594
By actions ✓
Used to build and publish .NET source. Set up a specific

Se usate una versione diversa da Java 11, ricordate di cambiare l'identificativo

4 La Action apparirà nella tab associata



The screenshot shows the GitHub Actions interface for the repository `StefanoLambiasi/is2_tutorato_travisCI`. The `Actions` tab is selected, displaying a list of workflow runs. A red arrow points to the workflow run `Create maven.yml`, which is currently `Queued`.

Workflow Run Details:

Event	Status	Branch	Actor
Create maven.yml	Queued	main	now

The workflow run is triggered by a commit push to the `main` branch. The status is `Queued`, indicating it is waiting to be executed.

Tool per CI/CD — GitHub Actions

Esempio

Questo è un file YAML per un progetto Maven con JDK 18

```
# This workflow will build a Java project with Maven, and cache/restore any dependencies to improve the workflow execution time
# For more information see: https://help.github.com/actions/language-and-framework-guides/building-and-testing-java-with-maven
```

```
name: Java CI with Maven
```

```
on:
```

```
  push:
```

```
    branches: [ "main" ]
```

```
  pull_request:
```

```
    branches: [ "main" ]
```

```
jobs:
```

```
  build:
```

```
    runs-on: ubuntu-latest
```

```
    steps:
```

```
      - uses: actions/checkout@v3
```

```
      - name: Set up JDK 18
```

```
        uses: actions/setup-java@v3
```

```
        with:
```

```
          java-version: '18'
```

```
          distribution: 'temurin'
```

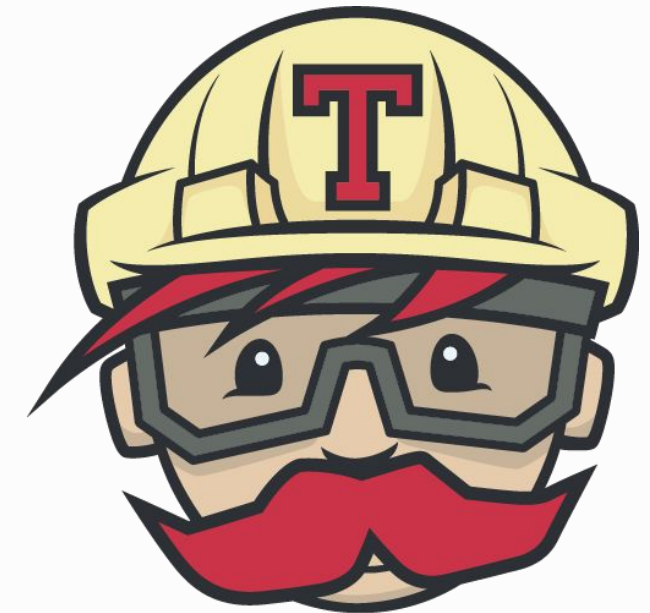
```
          cache: maven
```

```
      - name: Build with Maven
```

```
        run: mvn -B package --file pom.xml
```


Travis CI

***Travis CI** è un tool accessibile tramite il web che consente di automatizzare l'esecuzione di script su di un progetto software caricato su una repository condivisa (es. GitHub).*



Esempio di utilizzo

Ad esempio, utilizzando Travis CI è possibile automatizzare la build di un progetto al momento in cui avviene un push sulla repository su GitHub. In questa maniera, è possibile determinare se le modifiche fatte dallo sviluppatore introducono bug che impediscono la build del progetto. Sulla base del risultato, si potrebbe impedire che le modifiche vengano aggiunte alla repository condivisa.

Per accedere a Travis CI è possibile usare il proprio account GitHub!

Risorse utili

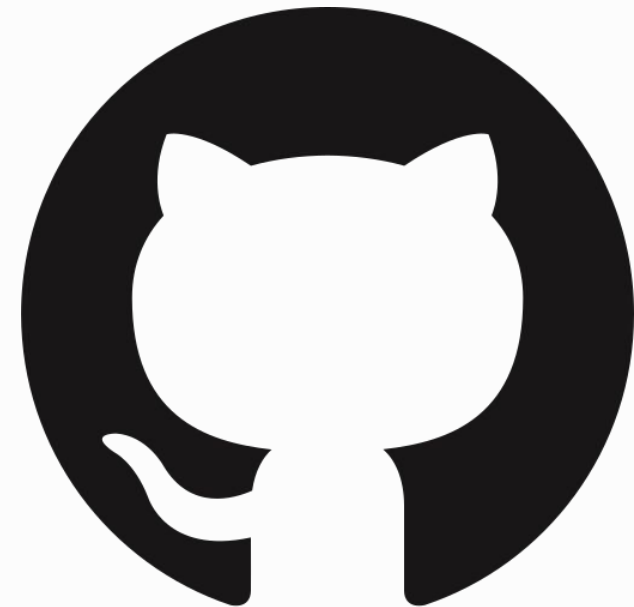
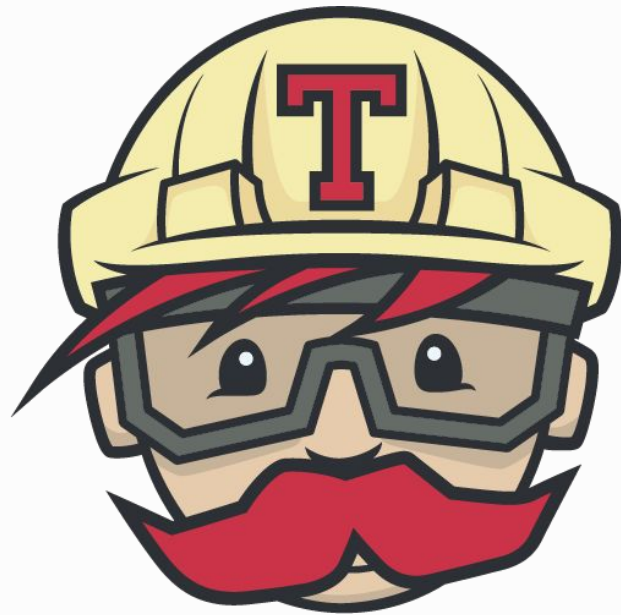
- Link a Travis CI: <https://www.travis-ci.com/>
- Link alla documentazione di Travis CI: <https://docs.travis-ci.com/>
- Link al *getting started* di Travis CI: https://app.travis-ci.com/getting_started

Travis CI + GitHub

Travis CI può essere integrato con GitHub

Prerequisiti:

1. Possedere un account di GitHub
2. Possedere un account di Travis CI (è possibile usare quello GitHub)



Tool per CI/CD – Travis CI

Travis CI + GitHub



- 1 Visita il sito https://app.travis-ci.com/getting_started

The screenshot shows the Travis CI dashboard interface. On the left, there's a sidebar with a search bar and a 'My Repositories' section showing 'Running (0/0)' and 'No repositories found'. The main content area is titled 'First time here? Let's get you started!' and contains three numbered steps:

- 1 Activate your GitHub repositories**
Once you're signed in, go to your profile page where you'll see all the organizations you're a member of.
You can install the GitHub App integration by clicking the Activate button for each organization you would like to use with Travis CI.
NOTE: You need to be an admin for any repositories you want to integrate with Travis CI.
[ACTIVATE ALL REPOSITORIES USING GITHUB APPS](#)
- 2 Add a .travis.yml file to your repository**
In order for Travis CI to build your project, you'll need to add a `.travis.yml` configuration file to the root directory of your repository.
If a `.travis.yml` is not in your repository, or is not valid YAML, Travis CI will ignore it.
Here you can find some of our [basic language examples](#).
- 3 Trigger your first build**
So you've configured the GitHub App integration, and added a `.travis.yml` file to your repository. All you need to do now is commit it to your local git history and

A circular inset shows a snippet of a `.travis.yml` file:

```
language: ruby
rvm:
  - "1.8.7"
  - "1.9.2"
  - "1.9.3"
  - jruby-18mode # JRuby
  - jruby-19mode # JRuby
  - rbx
# uncomment this line
# script: bundle
```

Tool per CI/CD – Travis CI

Travis CI + GitHub



- 2 Tramite il pulsante evidenziato è possibile attivare l'integrazione di Travis CI con GitHub

Travis CI Dashboard

Search all repositories

My Repositories Running (0/0) +

No repositories found

First time here? Let's get you started!

- 1 Activate your GitHub repositories**

Once you're signed in, go to your profile page where you'll see all the organizations you're a member of.

You can install the GitHub App integration by clicking on the button for each organization you would like to use with Travis CI.

NOTE: You need to be an admin for any repositories you would like to use with Travis CI.

[ACTIVATE ALL REPOSITORIES USING GITHUB APPS](#)
- 2 Add a .travis.yml file to your repository**

In order for Travis CI to build your project, you'll need to add a `.travis.yml` configuration file to the root directory of your repository.

If a `.travis.yml` is not in your repository, or is not valid YAML, Travis CI will ignore it.

Here you can find some of our [basic language examples](#).
- 3 Trigger your first build**

So you've configured the GitHub App integration, and added a `.travis.yml` file to your repository. All you need to do now is commit it to your local git history and

```
language: ruby
rvm:
  - "1.8.7"
  - "1.9.2"
  - "1.9.3"
  - jruby-18mode # JRuby
  - jruby-19mode # JRuby
  - rbx
# uncomment this line to run the bundler
# script: bundle
```

Tool per CI/CD – Travis CI

Travis CI + GitHub



3 Attivare il Free Plan nella tab apposita

The screenshot shows the Travis CI user interface for the user **valeriapontillo92**. The interface is divided into two main sections: a left sidebar and a main content area.

Left Sidebar:

- MY ACCOUNT:** Displays the user's profile picture and name **valeriapontillo92**. Below it is a **Sync account** button.
- ORGANIZATIONS:** States "You are not currently a member of any organization."
- MISSING AN ORGANIZATION?:** Provides a link to "Review and add your authorized organizations."

Main Content Area:

- User Profile:** Shows the user's name **valeriapontillo92** and their GitHub handle **@valeriapontillo92**.
- Navigation Tabs:** Includes **Repositories** (selected), **Settings**, **Plan** (highlighted by a red arrow), **Migrate**, and **Plan usage**.
- GitHub Apps Integration:** Features a button to **Manage repositories on GitHub** and a search bar labeled **Filter GitHub repositories**.
- Repository List:** A table of repositories with their names and associated settings buttons.

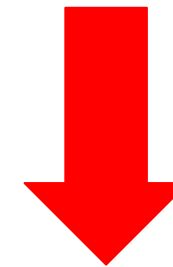
Repository Name	Settings
Curriculum	Settings
esercitazione1	Settings
esercitazione2-JFlex	Settings
esercitazione-3--Parser-Bottom-up	Settings
Ev_EnglishValidation_Student	Settings
libraryTest	Settings
Progetto_GPS_IS	Settings
tesiMagistrale	Settings
Unica-LaTeX	Settings

Tool per CI/CD – Travis CI

Travis CI + GitHub









- 5** Aggiungere il file `.travis.yml` nella root della directory



🔑 main ▾ 🔑 1 branch 🔑 0 tags

Go to file Add file ▾ Code ▾

 **StefanoLambiase** Create `.travis.yml` d21781b 9 minutes ago ⌚ 2 commits

 <code>.idea</code>	Initial commit	1 hour ago
 <code>src/main/java</code>	Initial commit	1 hour ago
 <code>.gitignore</code>	Initial commit	1 hour ago
 <code>.travis.yml</code>	Create <code>.travis.yml</code>	9 minutes ago
 <code>pom.xml</code>	Initial commit	1 hour ago

Help people interested in this repository understand your project by adding a README.

Add a README



Tool per CI/CD – Travis CI

Travis CI + GitHub



- 6** Tornare sulla pagina di Travis e selezionare il progetto a cui si è aggiunto il `.travis.yml` per innescare la build

valeriapontillo92 / libraryTest build unknown

Current Branches Build History Pull Requests Settings

General

☒ Build pushed branches ? ☐ Limit concurrent jobs ?

☒ Build pushed pull requests ? [User management](#)

More options

- Settings
- Requests
- Caches
- Trigger build

Auto Cancellation

Auto Cancellation allows you to only run builds for the latest commits in the queue. This setting can be applied to builds for Branch builds and Pull Request builds separately. Builds will only be canceled if they are waiting to run, allowing for any running jobs to finish.

☐ Auto cancel branch builds ☐ Auto cancel pull request builds

Clone or Import

☐ Allow files from this repository to be used in builds on other repositories

ON makes files from this repository to be used in builds on other repositories in case it is needed to pull down additional private dependencies.
OFF means that no information from the repository is shared and it is possible some builds using private dependencies between repos can break.

Tool per CI/CD – Travis CI

Travis CI + GitHub



7

Selezionare una versione della Java Development Kit per l'esecuzione dello script

My Repositories Running (0/0) +

✓ valeriapontillo92/libraryTest # 2
⌚ Duration: 1 min 9 sec
📅 Finished: less than a minute ago

○ valeriapontillo92/Unisa-LaTeX
⌚ Duration: -

○ valeriapontillo92/Ev_EnglishVali
⌚ Duration: -

○ valeriapontillo92/Progetto_GPS
⌚ Duration: -

○ valeriapontillo92/Progettazione
⌚ Duration: -

Current Branches Build History Pull Requests > Build #2 More options

✓ main add travis file #2 passed Restart build
🔗 Commit 3fb415e
🌿 Branch main
👤 valeriapontillo92
⌚ Ran for 39 sec
⌚ Total time 1 min 9 sec
📅 less than a minute ago

Build jobs View config

✓ # 2.1	AMD64	Xenial	</> JDK: openjdk8 Java	no environment variables set	⌚ 32 sec	⚙
✓ # 2.2	AMD64	Xenial	</> JDK: openjdk11 Java	no environment variables set	⌚ 37 sec	⚙

























































Tool per CI/CD – Travis CI

Travis CI + GitHub



8

Attendere l'esecuzione dell'operazione...

	 main  valeriapontillo	minor fix	 #6 received  c8db65f 	 -  -	
	 main  valeriapontillo	minor fix	 #5 failed  2ad5319 	 1 min 4 sec  2 years ago	
	 main  valeriapontillo	minor fix	 #4 passed  1998b9a 	 1 min 4 sec  2 years ago	
	 main  valeriapontillo	minor fix	 #3 failed  a6494ff 	 1 min 1 sec  2 years ago	
	 main  valeriapontillo	add travis file	 #2 passed  3fb415e 	 1 min 9 sec  2 years ago	
	 main  valeriapontillo	add travis file	 #1 passed  3fb415e 	 1 min 14 sec  2 years ago	

Tool per CI/CD – Travis CI

Travis CI + GitHub



9 Osservare con attenzione il processo di build

valeriapontillo92 / libraryTest  build passing

Current Branches Build History Pull Requests > Build #4 Job #4.1

More options

✓ main minor fix

🔗 #4.1 passed

🔄 Restart job

🔗 Commit 1998b9a

🕒 Ran for 31 sec

🐛 Debug job

🔗 Compare a6494ff...1998b9a

📅 less than a minute ago

🔗 Branch main

👤 valeriapontillo92

🔗 JDK: openjdk8 Java

🔗 AMD64

Job log

View config

```
1 Worker information
6
7 Build system information
161
162
163 Installing openjdk8
164
165 Installing SSH key from: default repository key
167 Using /home/travis/.netrc to clone repository.
168
169 $ git clone --depth=50 --branch=main https://github.com/valeriapontillo92/libraryTest.git valeriapontillo92/libraryTest
177
```


Continuous Integration & Continuous Delivery

GitHub Actions + Travis CI

Laboratorio di Ingegneria del Software

