SonaQube Configuration

Using Github Actions

Alessandro MIDOLO, 2/18/25

Requirements

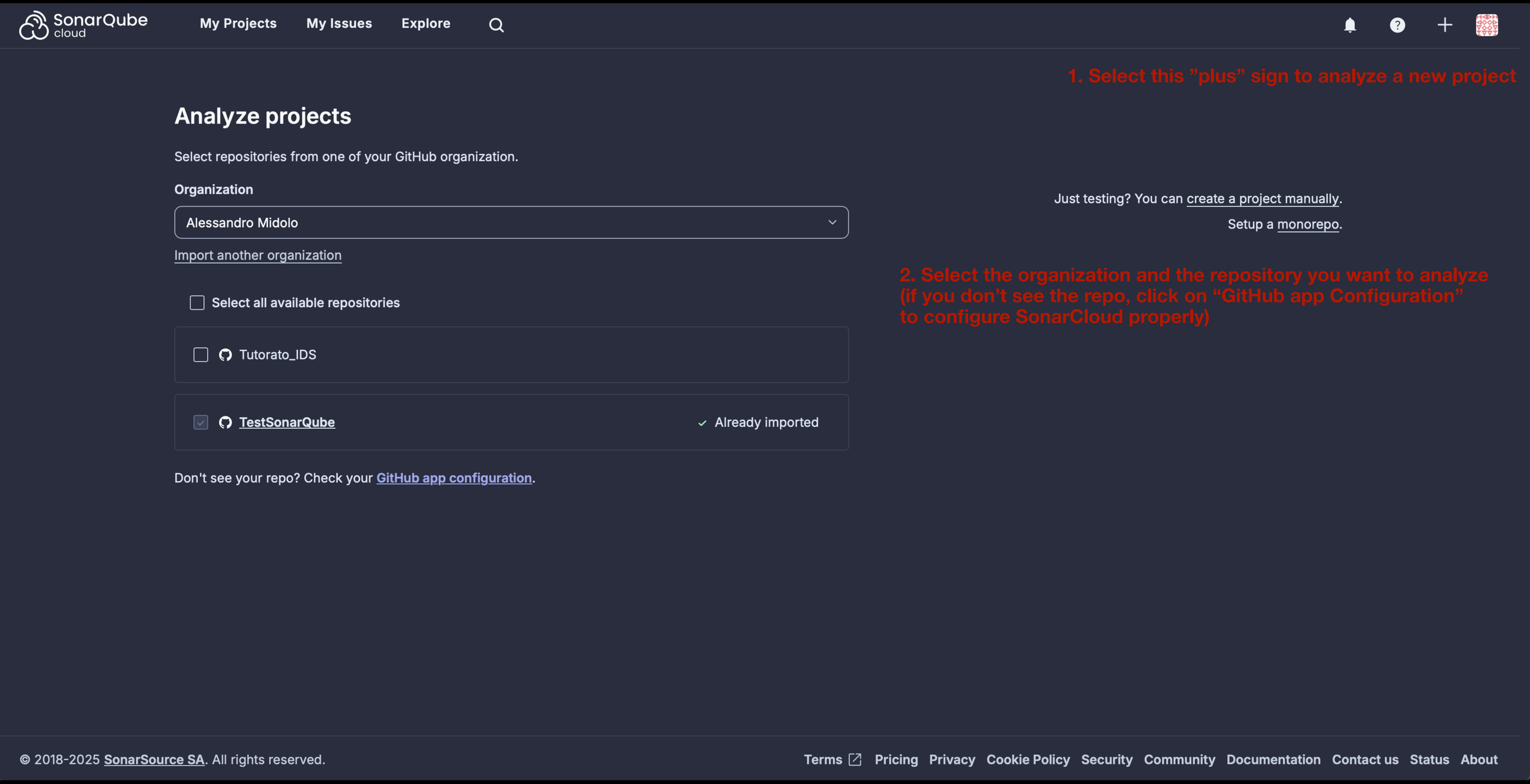
- GitHub Account (https://github.com/)
  - Make sure the repository you want to analyze is public
- SonarCloud Account (https://sonarcloud.io/) —> You should sign in with your GitHub account
  - Give SonarCloud access to your GitHub repositories (you are not grant access to all your repositories, you can select just the ones you want to analyze)

Github Actions

- We configure SonarCloud using GitHub Actions
- A GitHub action is an automated workflow that runs in response to events in a GitHub repository, such as pushing code, creating a pull request, or opening an issue
- You can create a PUBLIC repository where you will put the code you want to analyze (the repo should be empty, you will put the code LATER)
    - The repo should be public because you need to pay the membership if you want to analyze private repositories

SonarCloud Steps

- On your SonarCloud Dashboard, click “Analyze new Project”
- Select GitHub as your ALM (Application Lifecycle Management)
- Choose your GitHub organization (yourself) and repository
- See next slides for each STEP of the configuration


SonarQube cloud
My Projects
My Issues
Explore
1. Select this ”plus” sign to analyze a new project
Analyze projects
Select repositories from one of your GitHub organization.
Organization
Alessandro Midolo
Import another organization
Just testing? You can create a project manually.
Setup a monorepo.
2. Select the organization and the repository you want to analyze (if you don’t see the repo, click on “GitHub app Configuration” to configure SonarCloud properly)
Select all available repositories
Tutorato_IDS
TestSonarQube
Already imported
Don't see your repo? Check your GitHub app configuration.
© 2018-2025 SonarSource SA. All rights reserved.
Terms
Pricing
Privacy
Cookie Policy
Security
Community
Documentation
Contact us
Status
About

SonarQube cloud

My Projects | My Issues | Explore

Set up project for Clean as You Code

The new code definition sets which part of your code will be considered new code.

This helps you focus attention on the most recent changes to your project, enabling you to follow the Clean as You Code methodology.

Learn more: New Code Definition

Set a new code definition for your organisation to use it by default for all new projects

This can help you use the Clean as You Code methodology consistently across projects.

Alessandro Midolo - Administration - New Code

You can select “Previous Version.” This will setup SonarCloud to analyze the code you will pull in the repo starting from now

The new code for this project will be based on:

- (•) **Previous version**
  Any code that has changed since the previous version is considered new code.
  Recommended for projects following regular versions or releases.
- ( ) **Number of days**
  Any code that has changed in the last x days is considered new code. If no action is taken on a new issue after x days, this issue will become part of the overall code.
  Recommended for projects following continuous delivery.

You can change this at any time in the project administration

Back | Create project

Tutorato_IDS

PUBLIC

Overview
Main Branch
Pull Requests 0
Branches 0

Choose your Analysis Method

With GitHub Actions

With Travis CI

With CircleCI

With Amazon CodeCatalyst

With other CI tools

SonarQube Cloud integrates with your workflow no matter which CI tool you're using.

Manually

Use this for testing. Other modes are recommended to help you set up your CI environment.

Information
Administration
Collapse

SonarQube cloud

My Projects My Issues Explore

Tutorato_IDS

PUBLIC

Overview

Main Branch

Pull Requests 0

Branches 0

Information

Administration

Collapse

Alessandro Midolo > Tutorato_IDS > Administration / Analysis Method > Analyze a project with GitHub Actions

Analyze a project with a GitHub Action

1 Create a GitHub Secret

In your GitHub repository, go to Settings > Secrets > Actions and create a new secret with the following details:

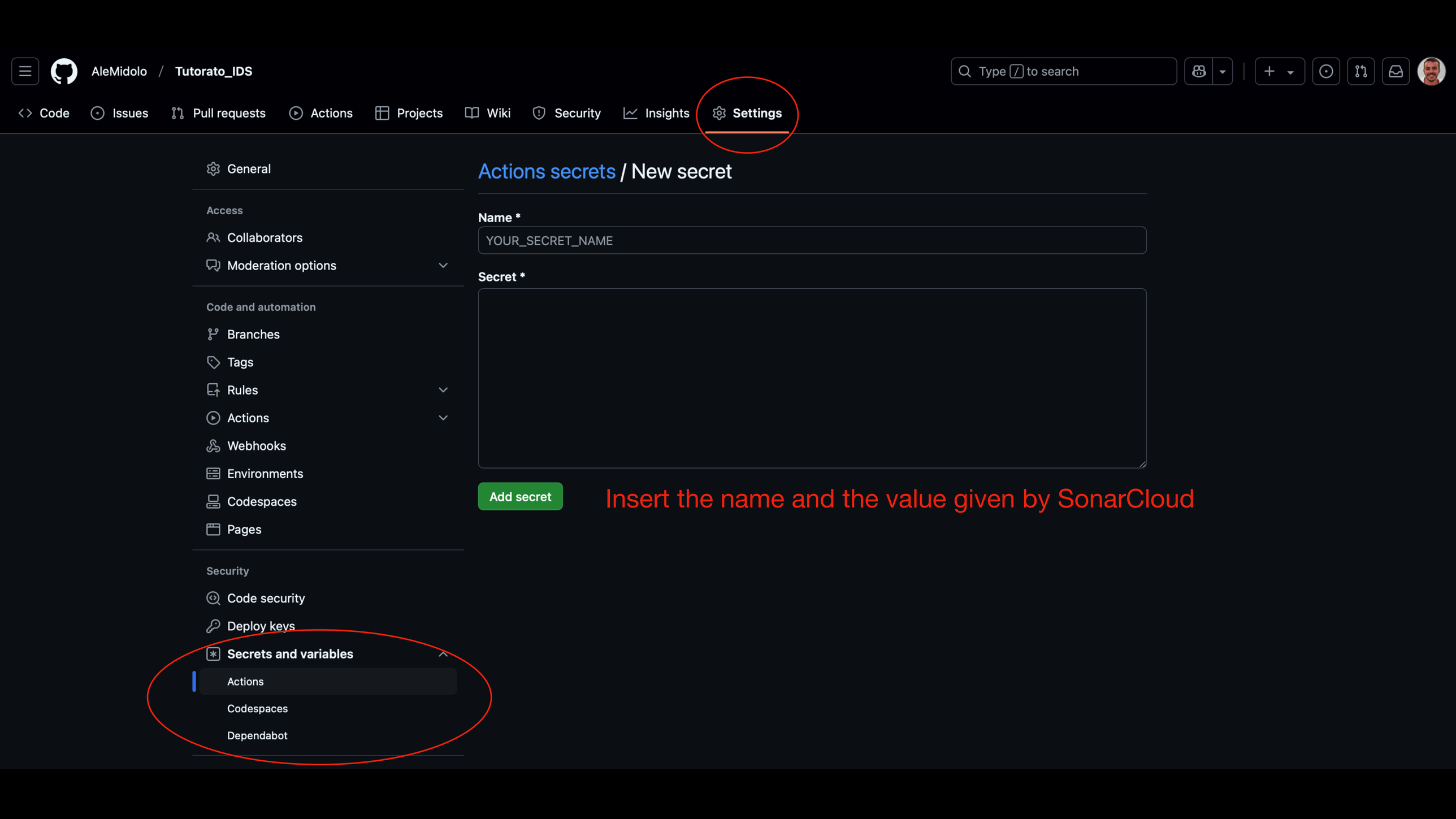
1. In the **Name** field, enter `SONAR_TOKEN`
2. In the **Value** field, enter `dec6f7b9073ff6da5b1ef4c094aba8a8a23acc69`

DO NOT USE THIS (use your secret)

2 Create or update a build file

What option best describes your project?

Maven	Gradle	.NET	C, C++ or ObjC	Flutter or Dart	Other (for JS, TS, Go, Python, PHP, ...)


AleMidolo / Tutorato_IDS
Type / to search
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
General
Access
Collaborators
Moderation options
Code and automation
Branches
Tags
Rules
Actions
Webhooks
Environments
Codespaces
Pages
Security
Code security
Deploy keys
Secrets and variables
Actions
Codespaces
Dependabot
Actions secrets / New secret
Name *
YOUR_SECRET_NAME
Secret *
Add secret
Insert the name and the value given by SonarCloud

Overview

Main Branch

Pull Requests 0

Branches 0

Information

Administration

Collapse

Analyze a project with a GitHub Action

1 Create a GitHub Secret

In your GitHub repository, go to Settings > Secrets > Actions and create a new secret with the following details:

1. In the **Name** field, enter `SONAR_TOKEN`
2. In the **Value** field, enter `dec6f7b9073ff6da5b1ef4c094aba8a8a23acc69`

2 Create or update a build file

What option best describes your project?

Maven | Gradle | .NET | C, C++ or ObjC | Flutter or Dart | Other (for JS, TS, Go, Python, PHP, ...)

Select the language you will use. Follow the instructions

Overview

Main Branch

Pull Requests 0

Branches 0

Information

Administration

Collapse

Maven | Gradle | .NET | C, C++ or ObjC | Flutter or Dart | **Other (for JS, TS, Go, Python, PHP, ...)**

Which OS do you run your build on?

Linux | Windows | macOS

This is an example. I want to analyze python code

Create or update your `.github/workflows/build.yml`

1. Clone your repository, and create (in the main directory) the directory ./github/workflows

Here is a base configuration to run a SonarQube Cloud analysis on your master branch and Pull Requests. If you already have some GitHub Actions, you might want to just add some of these new steps to an existing one.

2. Create a file “build.yml” and copy the code below

Copy

```yaml
name: Build
on:
  push:
    branches:
      - main
  pull_request:
    types: [opened, synchronize, reopened]
jobs:
  sonarqube:
    name: SonarQube
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
        with:
          fetch-depth: 0  # Shallow clones should be disabled for a better relevancy of analysis
      - name: SonarQube Scan
        uses: SonarSource/sonarqube-scan-action@v4
        env:
          SONAR_TOKEN: ${{ secrets.SONAR_TOKEN }}
```

3 Create a `sonar-project.properties` file

Create a configuration file in the root directory of the project and name it `sonar-project.properties`

Follow the instructions: create the file and copy the code in it

Copy

```
sonar.projectKey=AleMidolo_Tutorato_IDS
sonar.organization=alemidolo

# This is the name and version displayed in the SonarCloud UI.
#sonar.projectName=Tutorato_IDS
#sonar.projectVersion=1.0


# Path is relative to the sonar-project.properties file. Replace "\" by "/" on Windows.
#sonar.sources=.

# Encoding of the source code. Default is default system encoding
#sonar.sourceEncoding=UTF-8
```

And you are done!

If everything is running successfully, once the analysis is complete you'll be redirected to the Overview page of your project where the new analysis results will be displayed. This can take a few minutes.

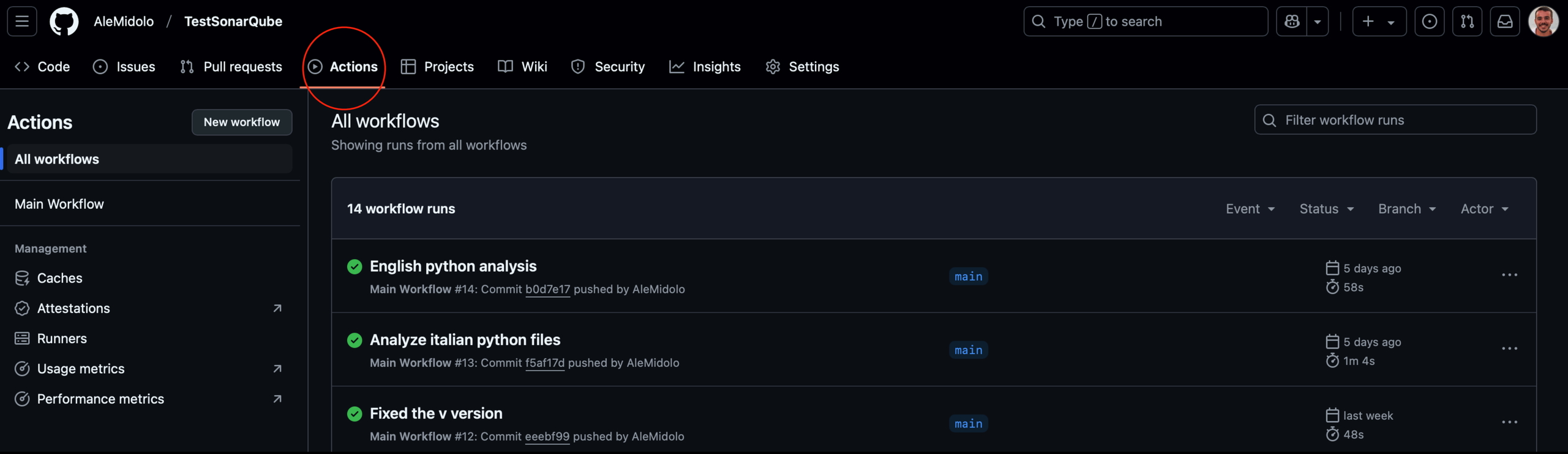
- ✓ You'll get an analysis of your default branch
- ✓ Each new push you make on a branch or Pull Request will automatically trigger a new analysis
- ✓ Each new Pull Request you create will automatically be analyzed

Start the analysis

- Now, to “trigger” the analysis, you should push the code you want to analyze in the repository —> this will automatically start the analysis of SonarCloud on the code you pushed
- The output of the analysis will be displayed on SonarCloud

This is an example of my repository

1. Push the code on the repository
2. Go to GitHub and click on the “Actions” tab
3. You will see the status of the analysis (you can even click on it to see each steps)
4. Once the analysis is completed (green mark), go on SonarCloud Dashboard to see the results



That’s all

Feel free to contact me if you need more assistance

- amidolo@unisannio.it