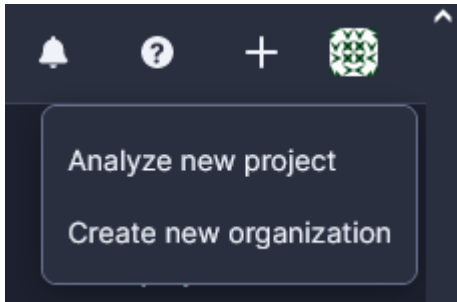


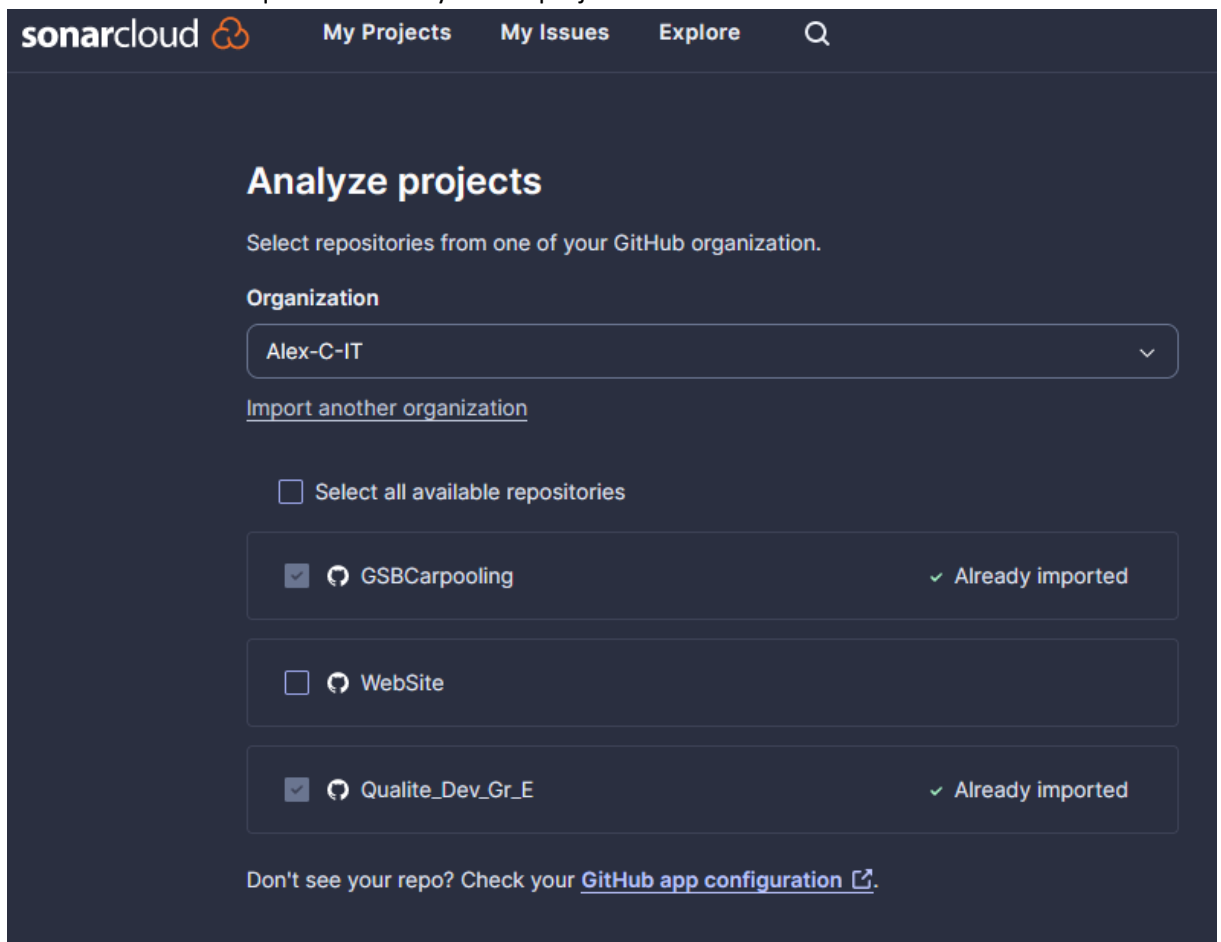
Installation et configuration de Sonar

SONARCLOUD

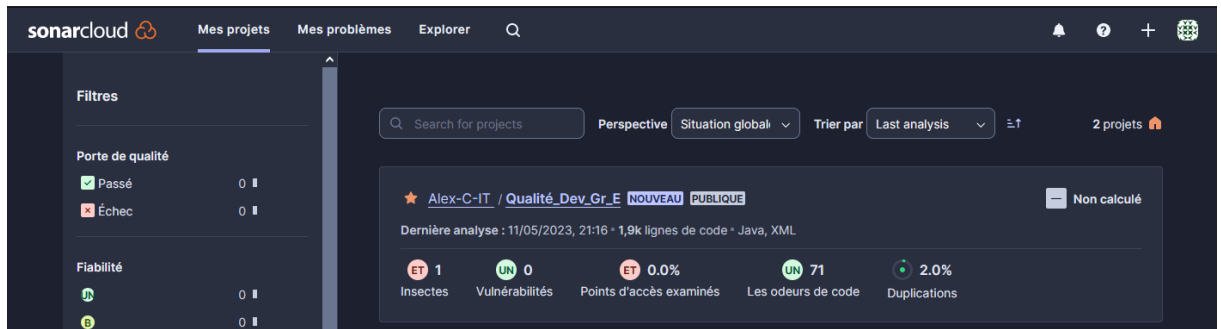
- Création d'un compte utilisateur sur SonarCloud
- Ajouter une organisation (Github)
- Ajouter un nouveau projet



En haut à droite > Cliquez sur « Analyze new project »



Sélectionnez le projet à analyser.



Le projet est chargé et a été préalablement et automatiquement analysé.

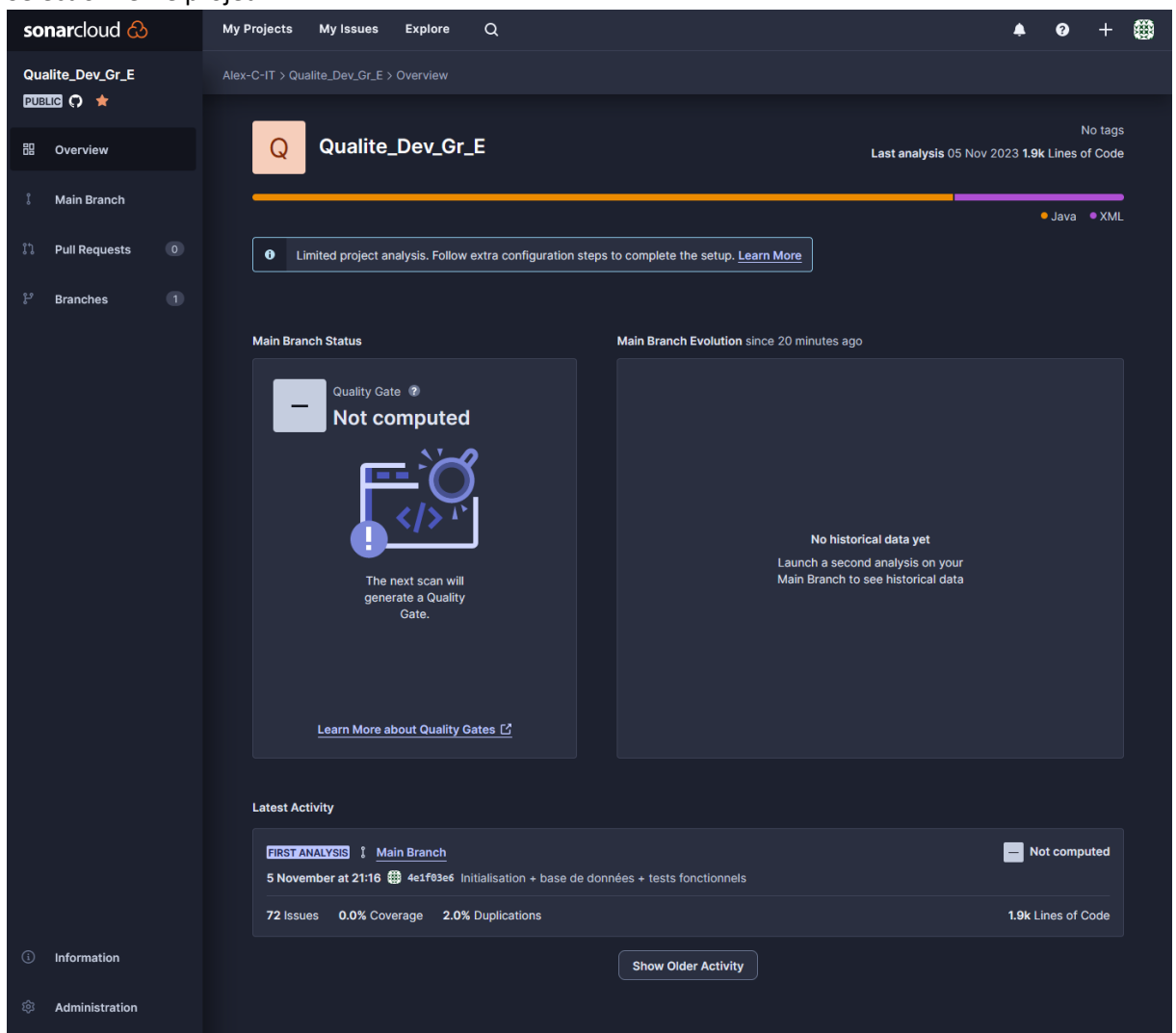
Celui-ci apparaît au sein du tableau de bord « My Project ».

L'analyse du projet est limitée.

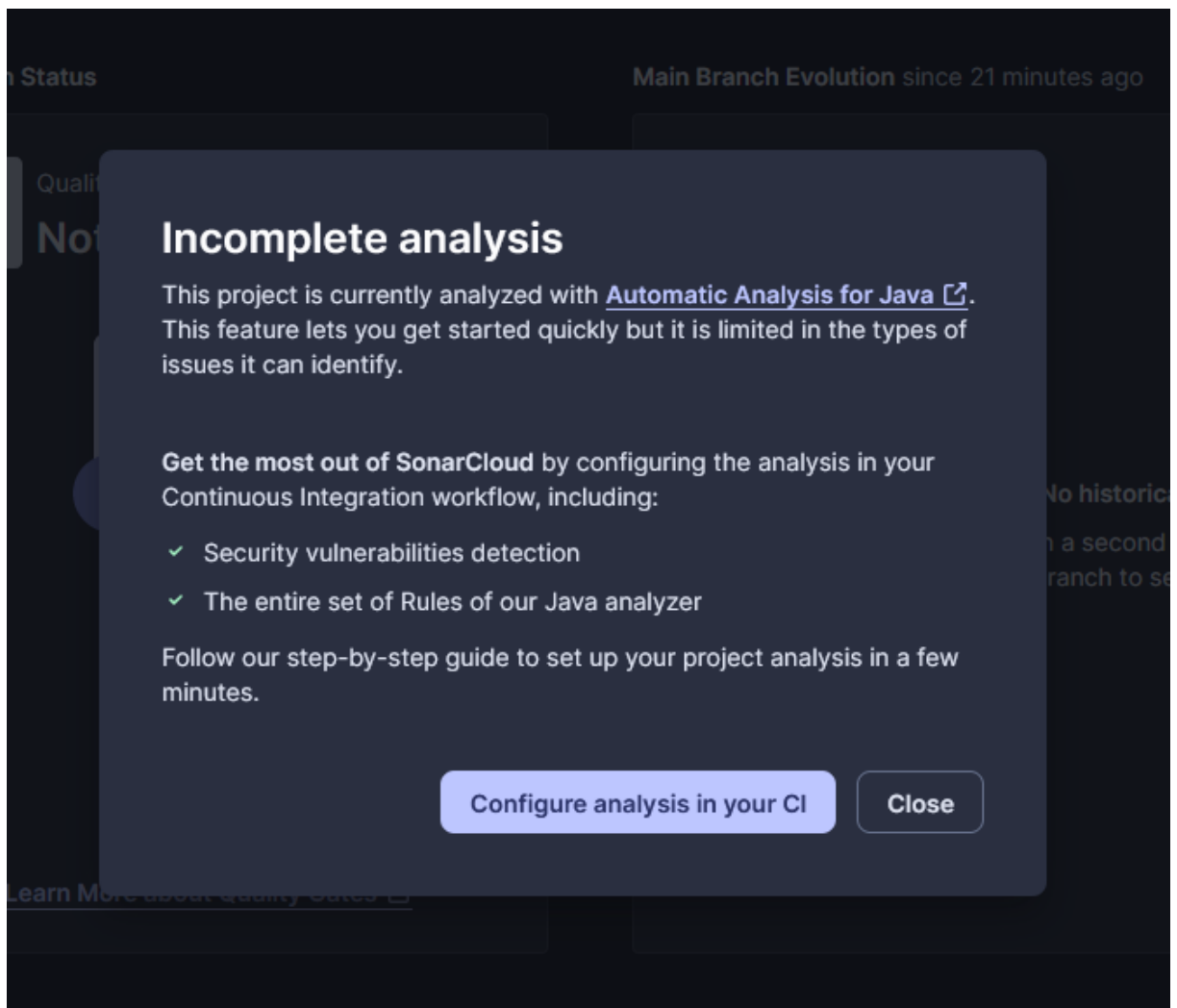
Ce projet est actuellement analysé avec « Automatic Analysis for Java ». Cette fonctionnalité vous permet de démarrer rapidement, mais elle est limitée dans les types de problèmes qu'elle peut identifier.

Différentes étapes de configuration supplémentaires sont nécessaires pour terminer la configuration.

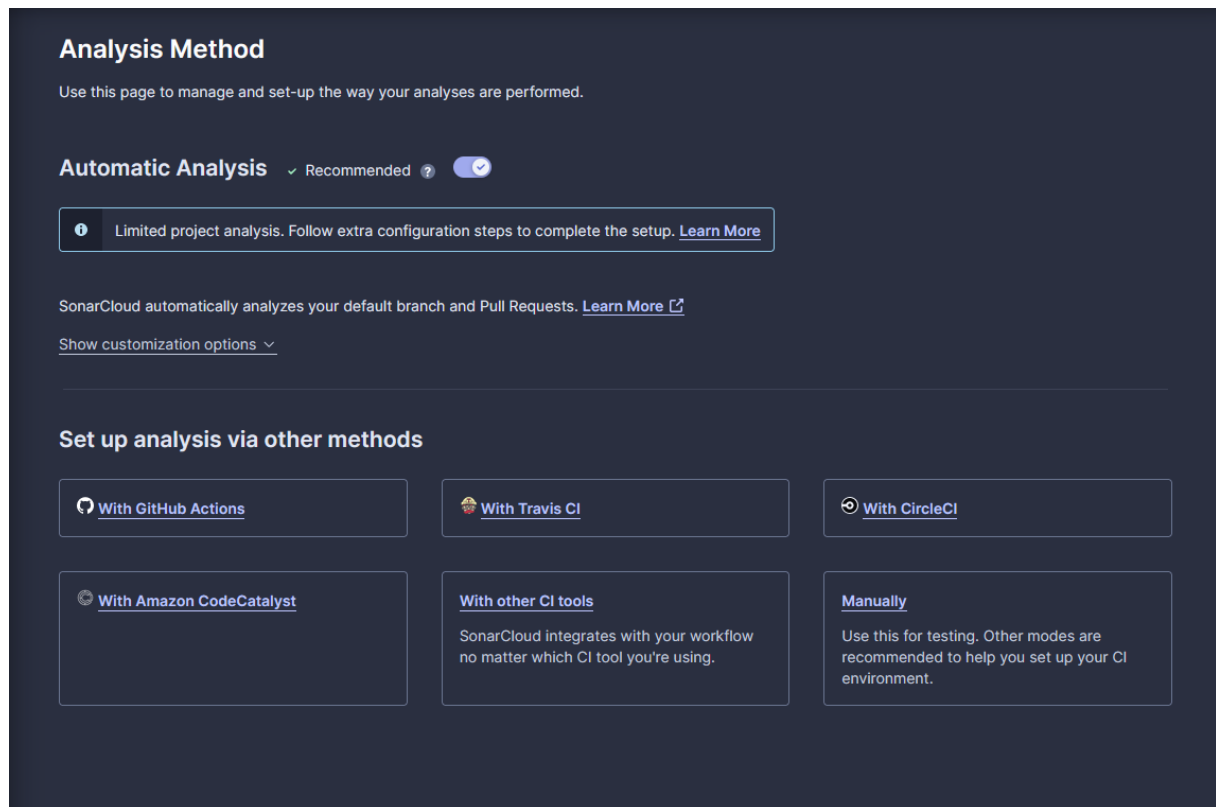
- Sélectionnez le projet



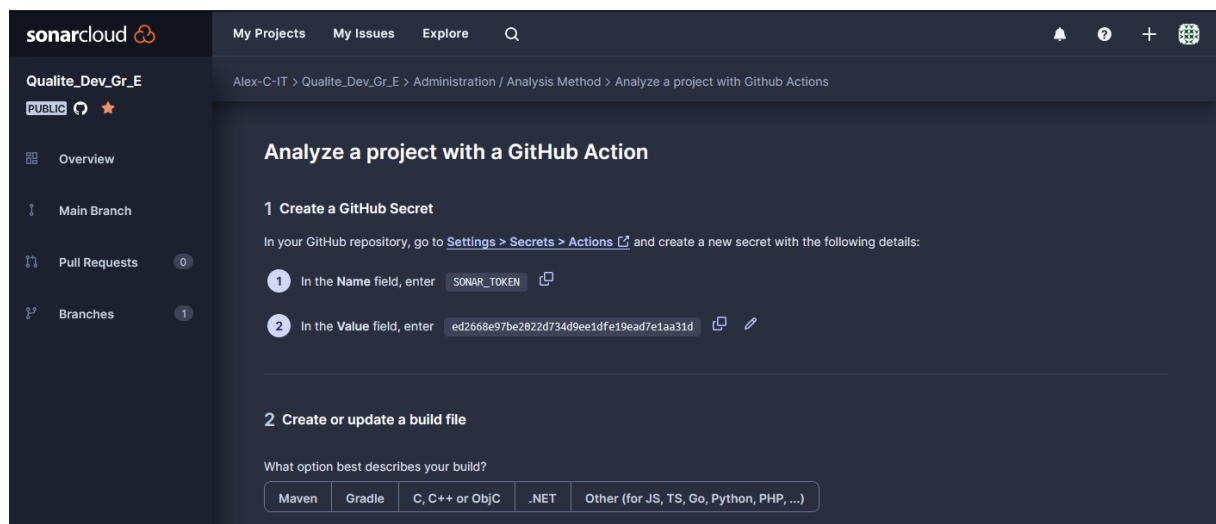
Cliquez sur « Learn More » pour finaliser la configuration de votre CI.



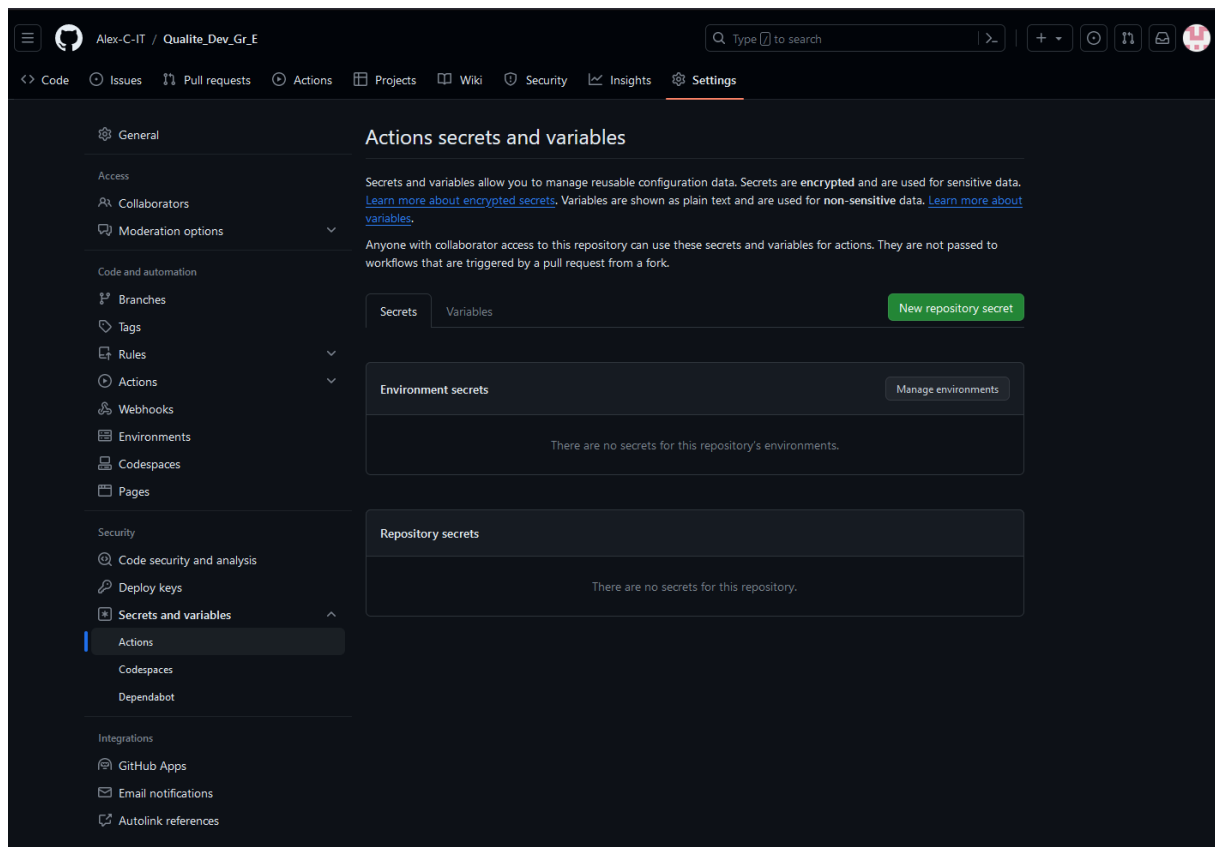
Cliquez sur « Configure analysis in your CI »



- Désactivez l'analyse automatique.
- Cliquez sur « With GitHub Actions »

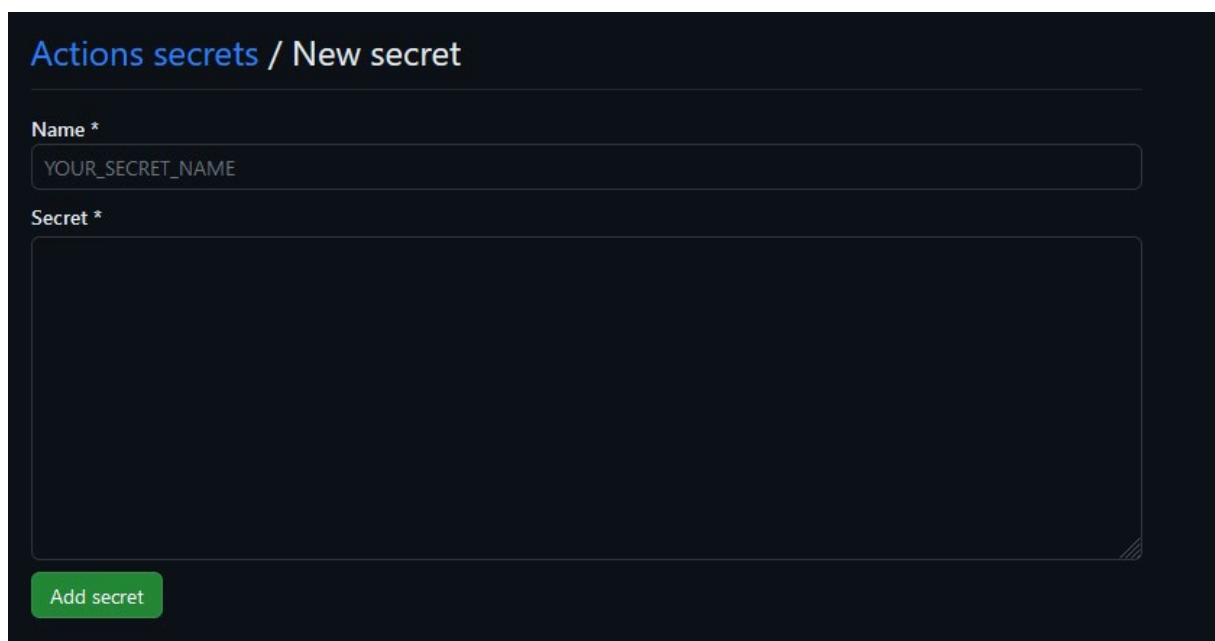


Ajouter une nouvelle clef privée secrète sur GitHub.



Dans Settings > Security > Actions :

- Cliquez sur « New repository secret ».



Ajoutez le nom et le token qui ont été préalablement fournis par SonarCloud.

Cliquez ensuite sur « Add secret ».

Retournez sur SonarCloud pour la mise à jour du fichier de build.

2 Create or update a build file

What option best describes your build?

Maven

Gradle

C, C++ or ObjC

.NET

Other (for JS, TS, Go, Python, PHP, ...)

Cliquez sur « Maven ».

2 Create or update a build file

What option best describes your build?

Maven

Gradle

C, C++ or ObjC

.NET

Other (for JS, TS, Go, Python, PHP, ...)

Update your `pom.xml` file with the following properties:

```
<properties>
  <sonar.organization>alex-c-it</sonar.organization>
  <sonar.host.url>https://sonarcloud.io</sonar.host.url>
</properties>
```

Copy

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

Here is a base configuration to run a SonarCloud 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.

```
name: SonarCloud
on:
  push:
    branches:
      - main
  pull_request:
    types: [opened, synchronize, reopened]
jobs:
  build:
    name: Build and analyze
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v3
        with:
          fetch-depth: 0 # Shallow clones should be disabled for a better relevancy of analysis
      - name: Set up JDK 17
        uses: actions/setup-java@v3
        with:
          java-version: 17
          distribution: 'zulu' # Alternative distribution options are available.
      - name: Cache SonarCloud packages
        uses: actions/cache@v3
        with:
          path: ~/.sonar/cache
          key: ${ runner.os }-sonar
          restore-keys: ${ runner.os }-sonar
      - name: Cache Maven packages
        uses: actions/cache@v3
        with:
          path: ~/.m2
          key: ${ runner.os }-m2-${ hashFiles('**/pom.xml') }
          restore-keys: ${ runner.os }-m2
      - name: Build and analyze
        env:
          GITHUB_TOKEN: ${ secrets.GITHUB_TOKEN } # Needed to get PR information, if any
          SONAR_TOKEN: ${ secrets.SONAR_TOKEN }
        run: mvn -B verify org.sonarsource.scanner.maven:sonar-maven-plugin:sonar -Dsonar.projectKey=Alex-C-IT-Qualite-Dev-Gr-E
```

Copy

Dans le fichier « pom.xml » du projet et au sein des balises <properties>, ajoutez le code associé :

```
11      <properties>
12          <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
13          <sonar.organization>alex-c-it</sonar.organization>
14          <sonar.host.url>https://sonarcloud.io</sonar.host.url>
15      </properties>
16
```

Pensez également à changer la version de Maven pour que celle-ci soit compatible avec SonarCloud.

```
      <plugin>
        <artifactId>maven-war-plugin</artifactId>
        <version>3.3.2</version>
        <configuration>
          <warSourceDirectory>WebContent</warSourceDirectory>
          <failOnMissingWebXml>false</failOnMissingWebXml>
          <webXml>WebContent/WEB-INF/web.xml</webXml>
        </configuration>
      </plugin>
```

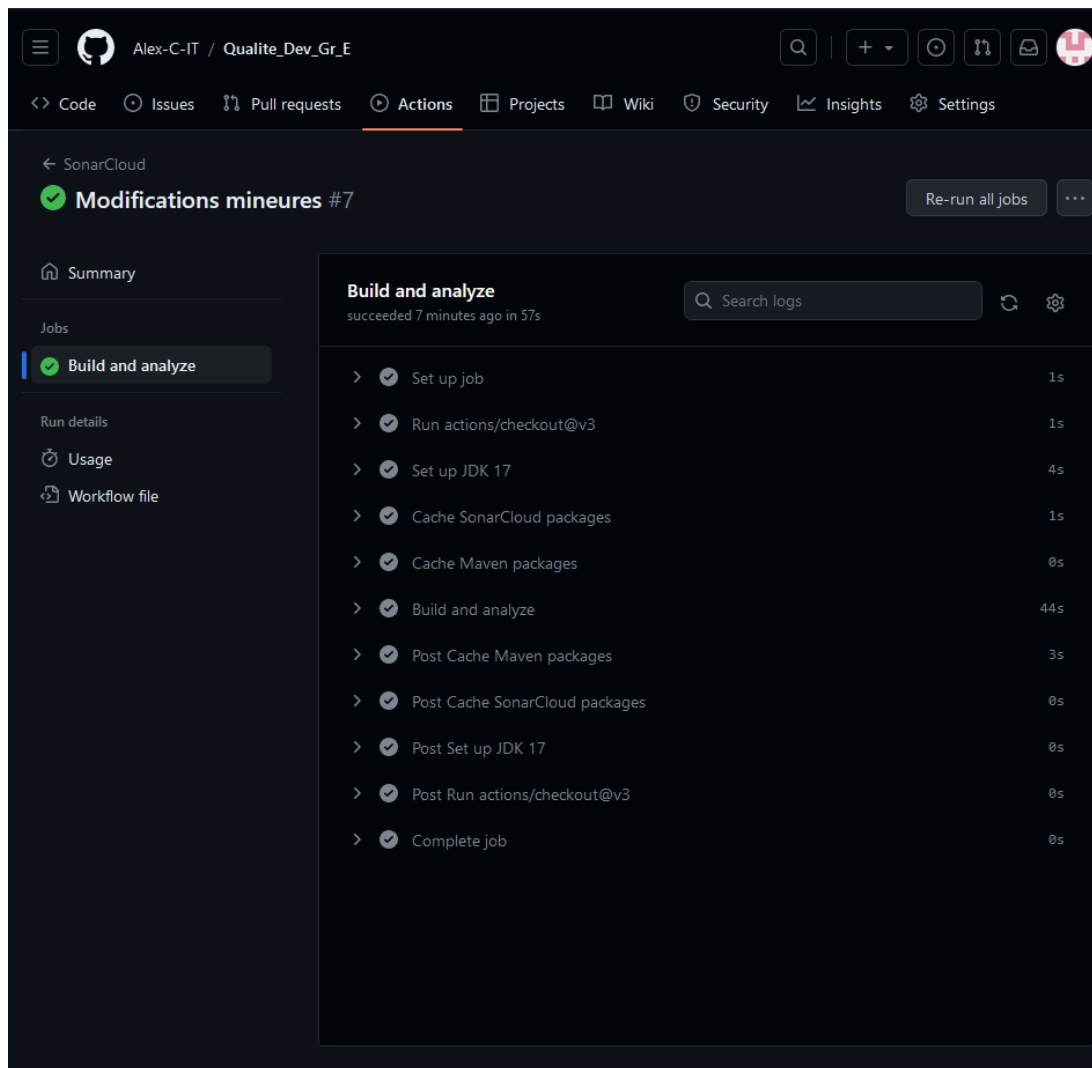
Créez un fichier « build.yml » à l'emplacement suivant (dans votre projet) :

.github/workflows/build.yml

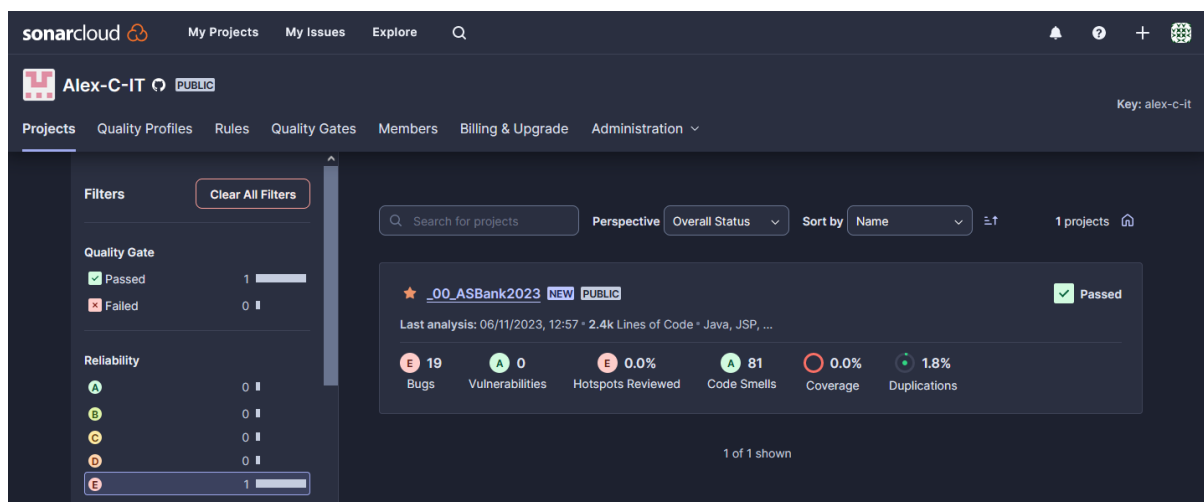
Collez-y la routine fournie par SonarCloud. Enregistrez.

Effectuez un push du projet.

Lors d'une nouvelle « pull request » ou un « push » sur la branche « master », il est possible de voir l'action s'effectuer dans GitHub :



Si la routine GitHub est correctement effectuée, l'analyse sur SonarCloud est visible.



FINITO PEPITOOOOO !!!