# Giới thiệu

Sonarqube là một platform mã nguồn mở giúp chúng ta có thể kiểm tra chất lượng code của dự án, được viết bằng java nhưng nó hổ trợ nhiều ngôn ngữ khác nhau: PHP, Ruby, Java,…

Lợi ích

* Phát hiện bug
* Phát hiện code smell, duplicate
* Tính toán độ bao phủ của Unit test (Unit-test coverage)
* Tính toán technical debt
* So sánh chất lượng code so với các lần kiểm tra trước

# Các cách sử dụng Sonar

## SonarLint

Sonar Lint là gói cài đặt khá là đơn giản với các IDE: Eclipse, Intelli J,..

Sau khi cài đặt plugin này cho IDE thì sẽ có các phân tích tự động với code, hoặc người dùng có thể scan để check code với code mình thích

## Sonar Queue

Sonar Queue là một product cài đặt trên server. Người dùng có thể tải Sonar Queue và cài đặt lên máy tính hoặc lên server có lựa chọn phân tích code theo các rules mà Sonar Queue cài đặt cho project của mình

## Sonar Cloud

Sonar Cloud là bản cài đặt trên các ứng dụng Cloud và có thể phân tích code mạng phân tán, dựa vào việc cài đặt nó kết nối đến một nguồn project nào đó mà mình mong muốn.

Trong tài liệu này các bạn sẽ làm quen với các thành phần trên

# Sonar Cloud Online

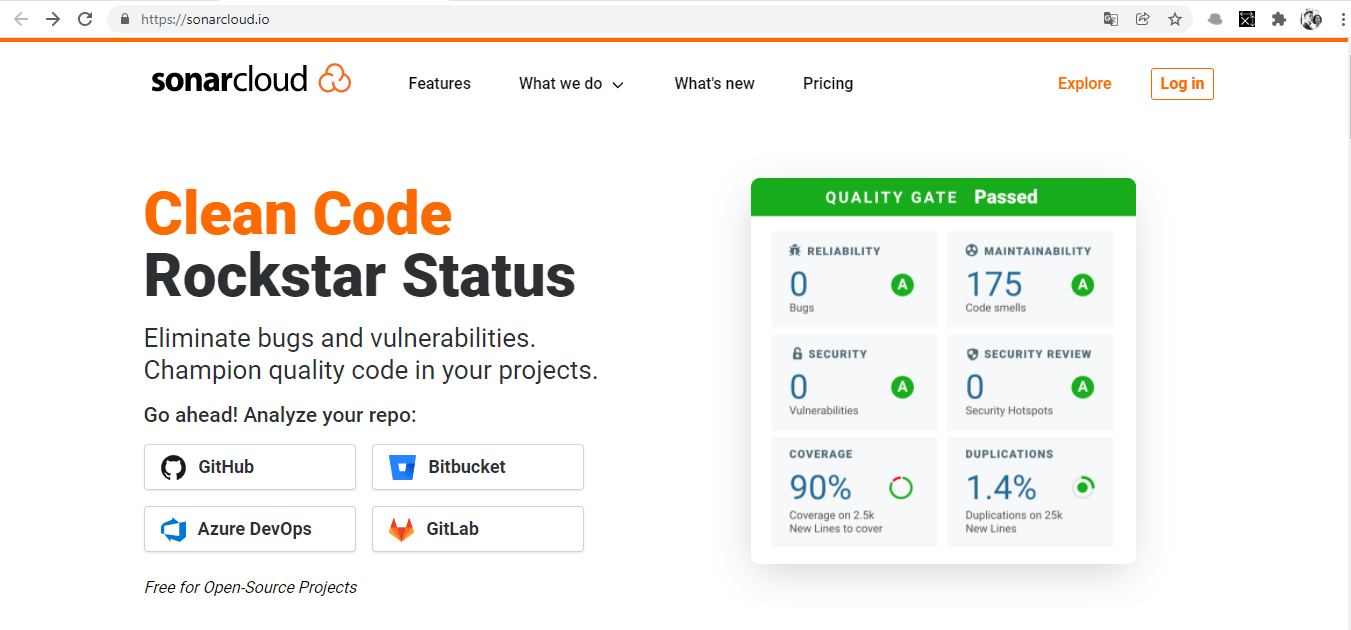
Sonar có bản Online và có thể kết nối với các repository Git mà chúng ta mong muốn phân tích code.

Các bước làm việc với Sonar Cloud được mô tả như sau

## Truy cập sonarcloud

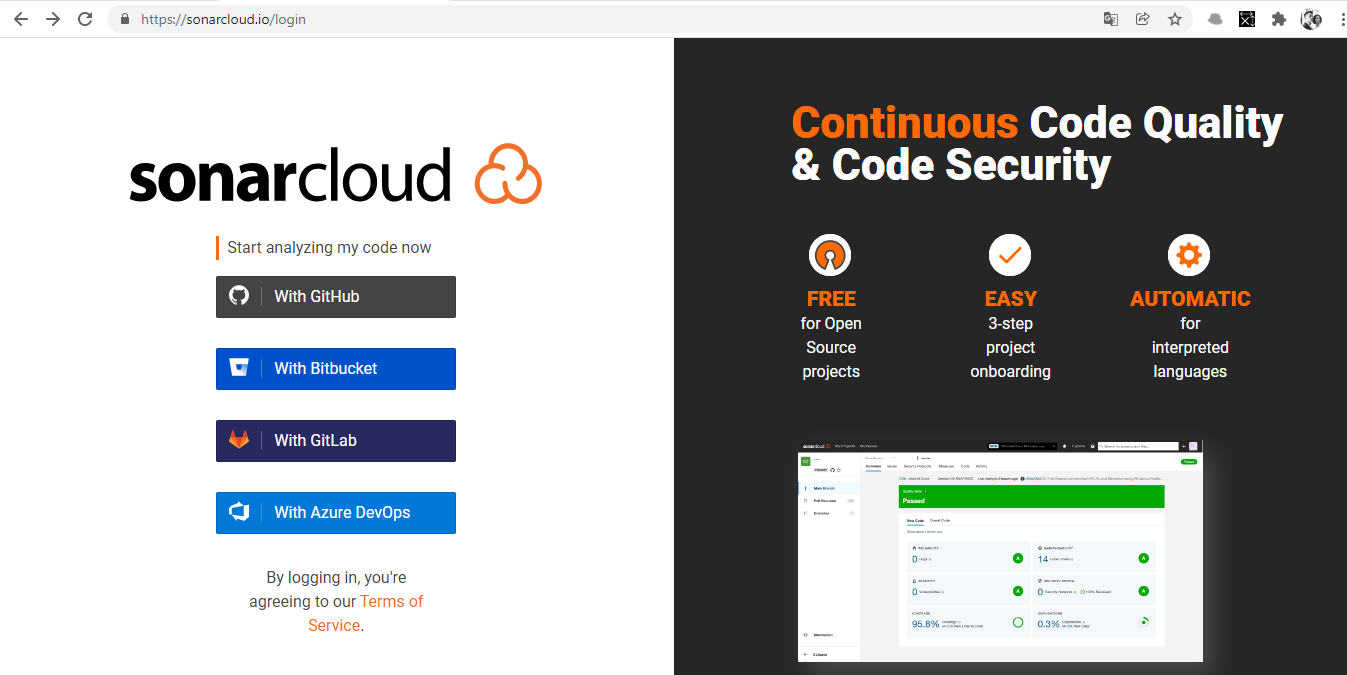
Vào hệ thống online của Sonar và cần user login

<https://sonarcloud.io/>

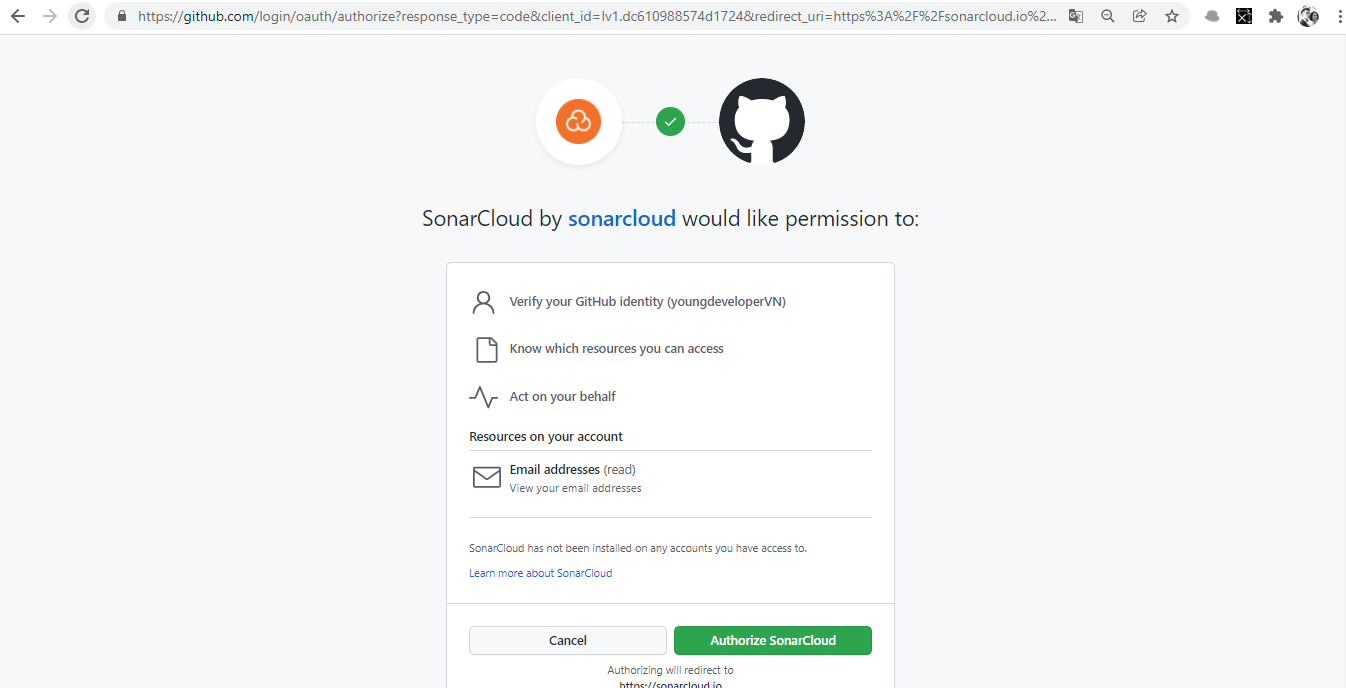


Trang chủ của Sonar Cloud

## Login với tài khoản Git và chọn repository tương ứng



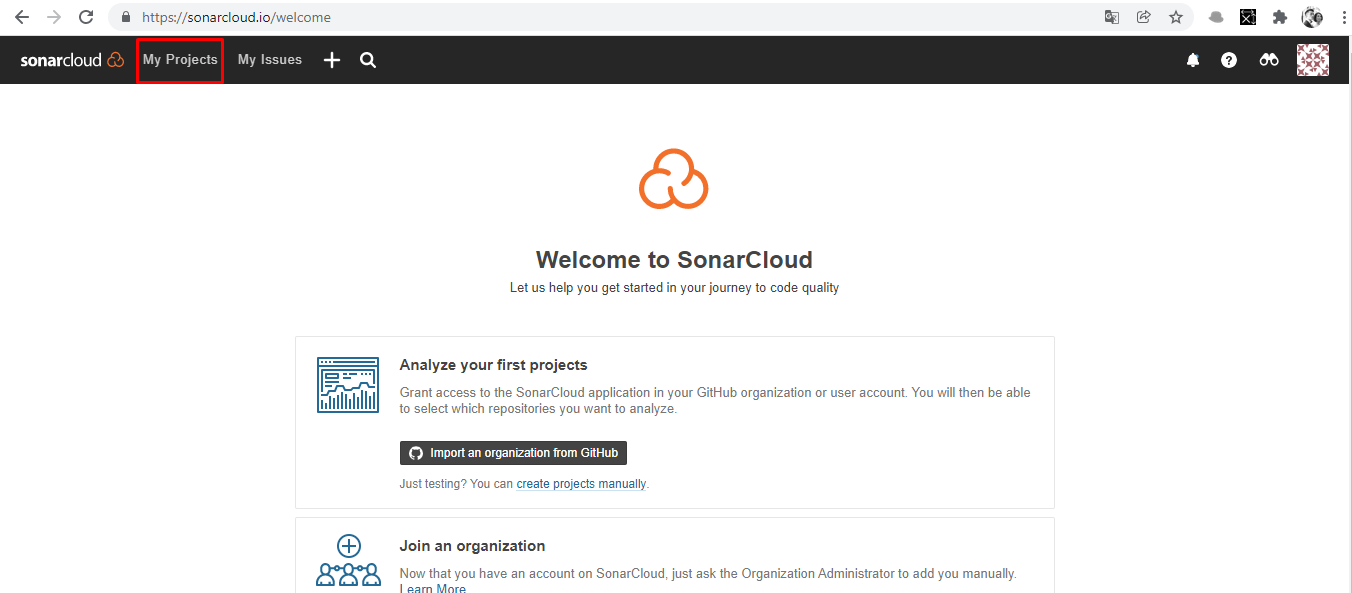
Login Sonar với Github của mình



Lựa chọn project của bản than

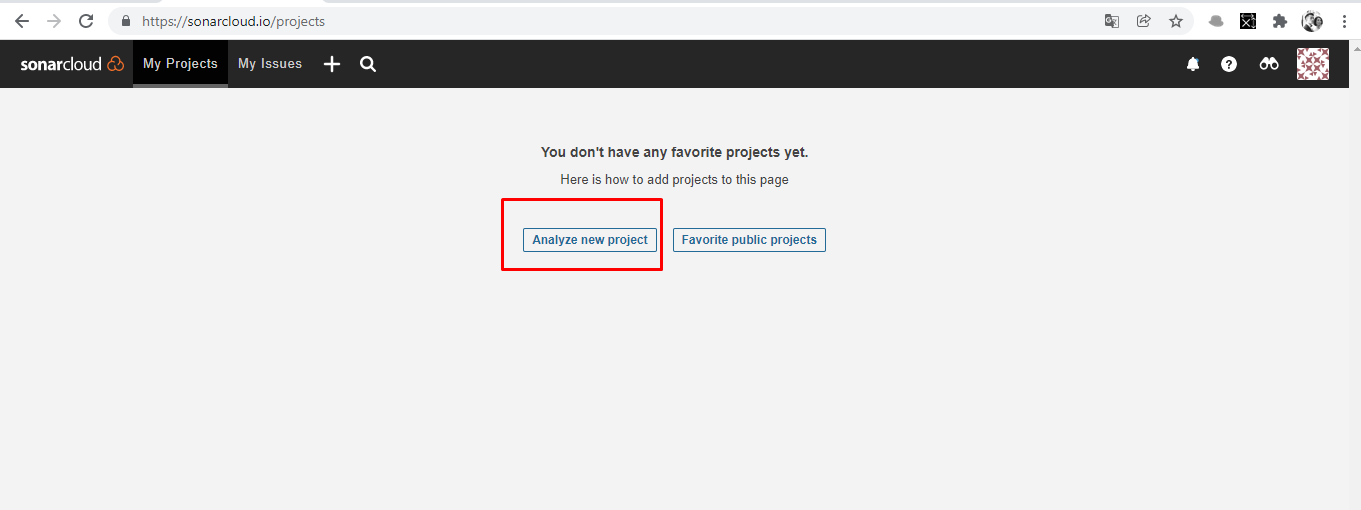
Sau khi lựa chọn việc cho phép Sonar dùng Git của mình, thì sẽ có một email gửi tới email cá nhân và chúng ta sẽ active cho phép Sonar.

Giao diện của trang Dashboard

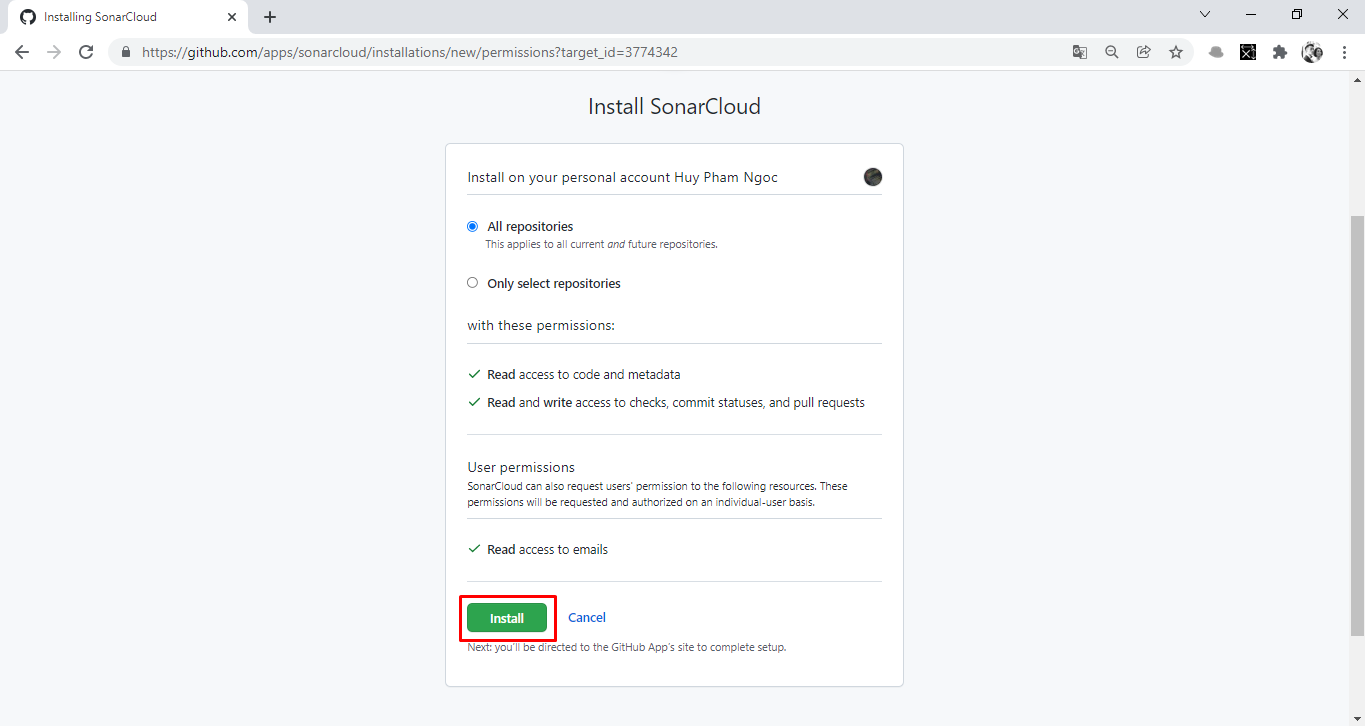


## Import project vào Sonar

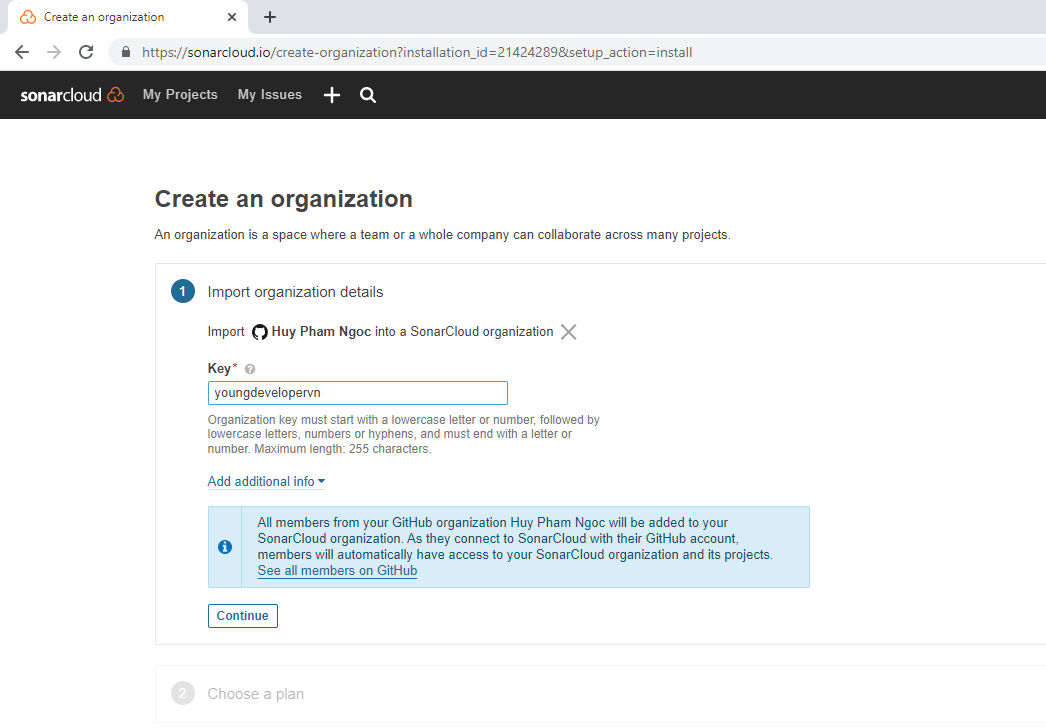
Lựa chọn My Project và chọn project mình cần Import vào Sonar

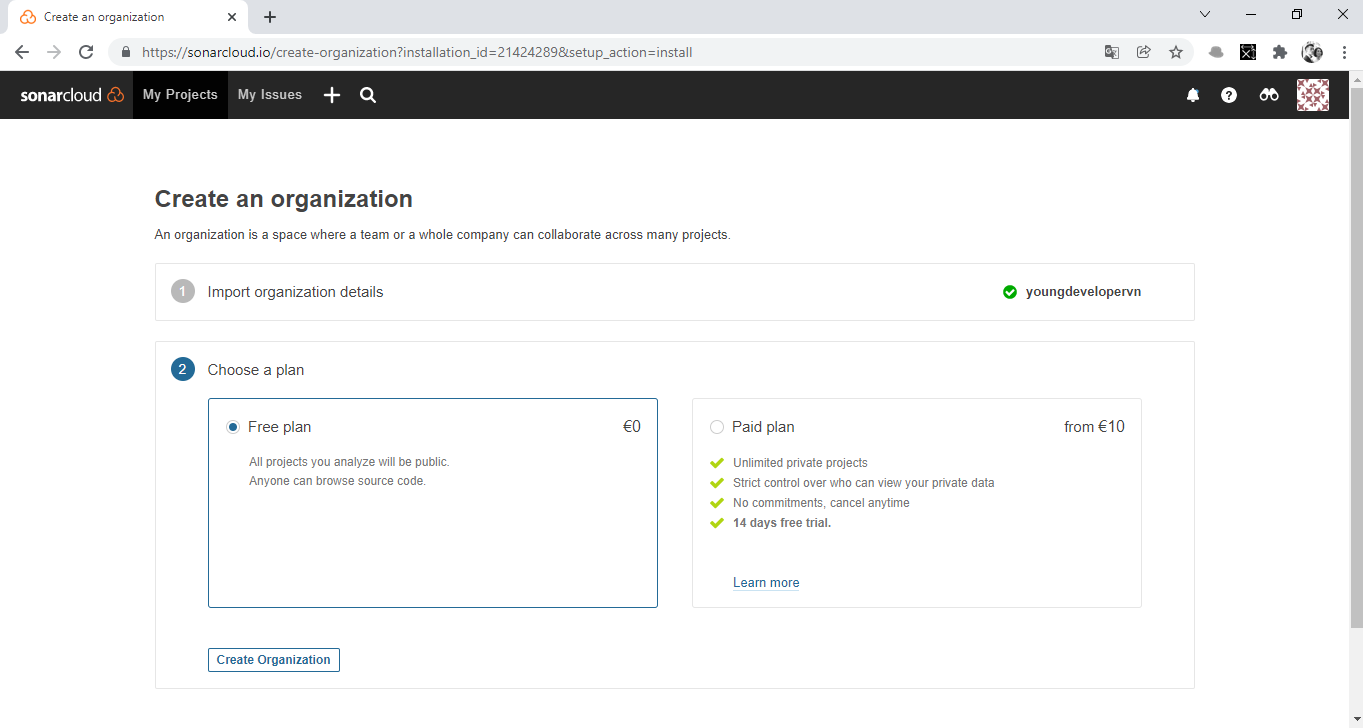


Sau đó lựa chọn Install Sonar với Git



Việc Login có thể được hỏi và các bạn sẽ cài đặt thành công Sonar với GitHub

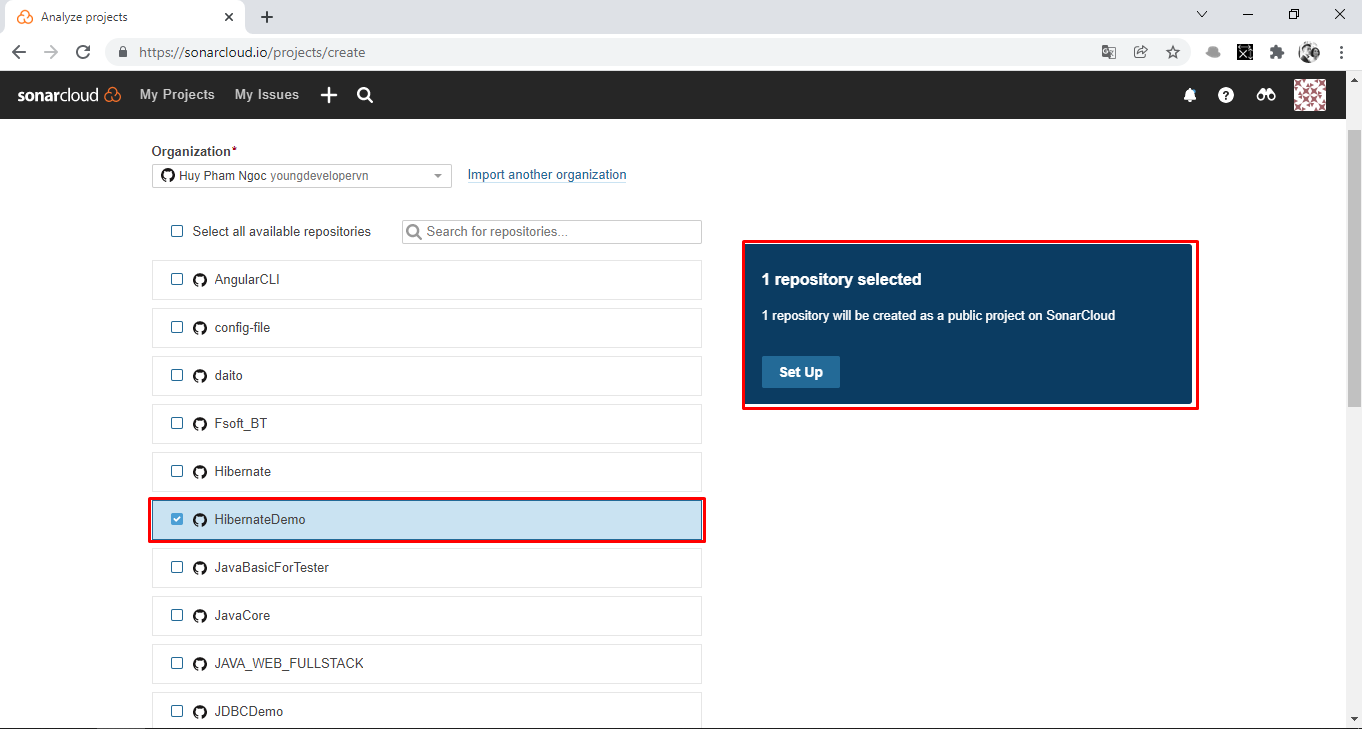




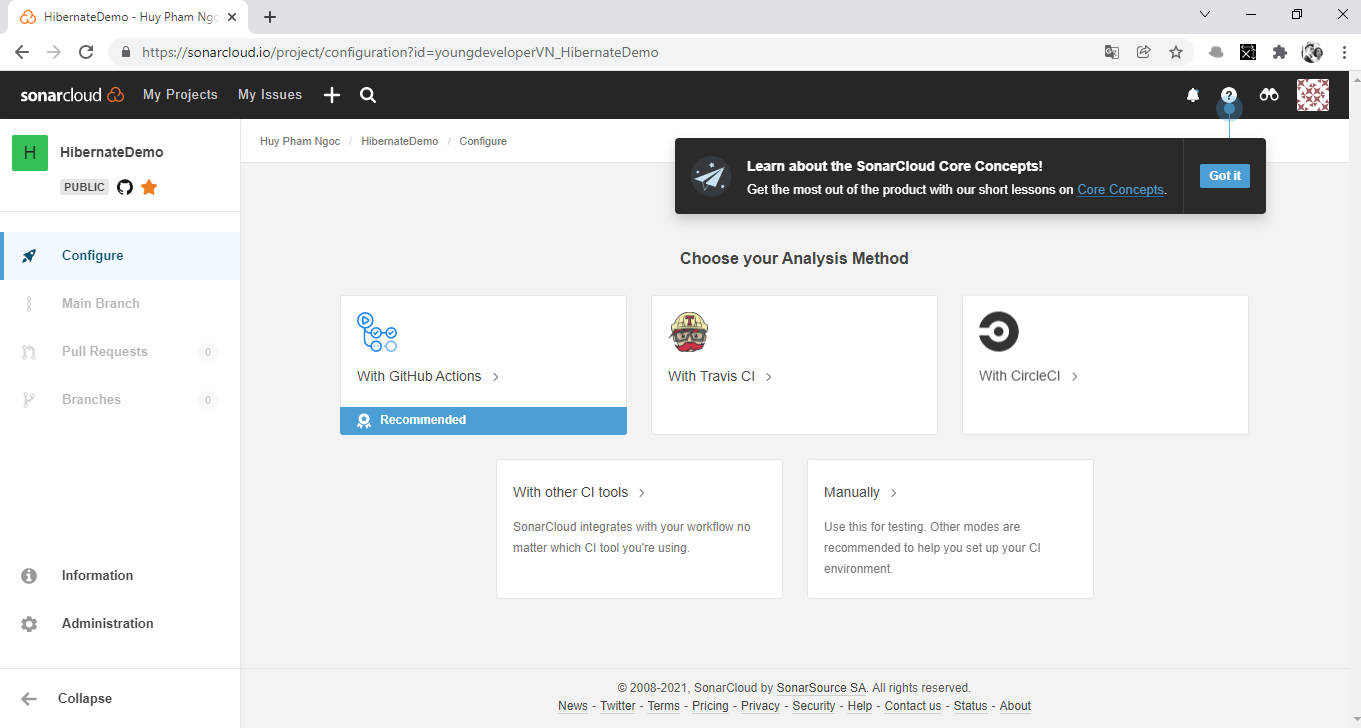
Sau khi lựa chọn Create Organization thì Sonar sẽ list các project mà các bạn có

Việc lựa chọn Scan project sẽ rất đơn giản

Ví dụ ở dưới là lựa chọn project Java

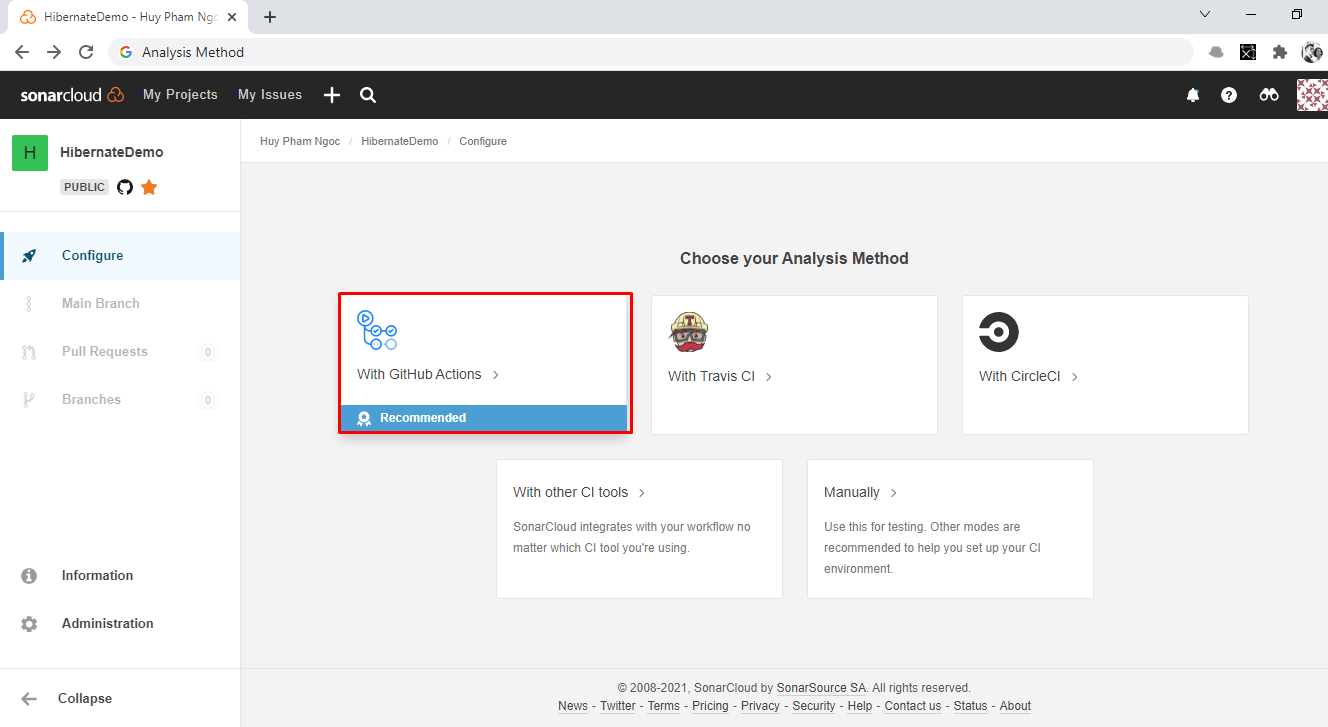


Sau đó các bạn chọn SetUp, sau khi chọn Setup sẽ ra giao diện như sau



## Cấu hình Analysis Method

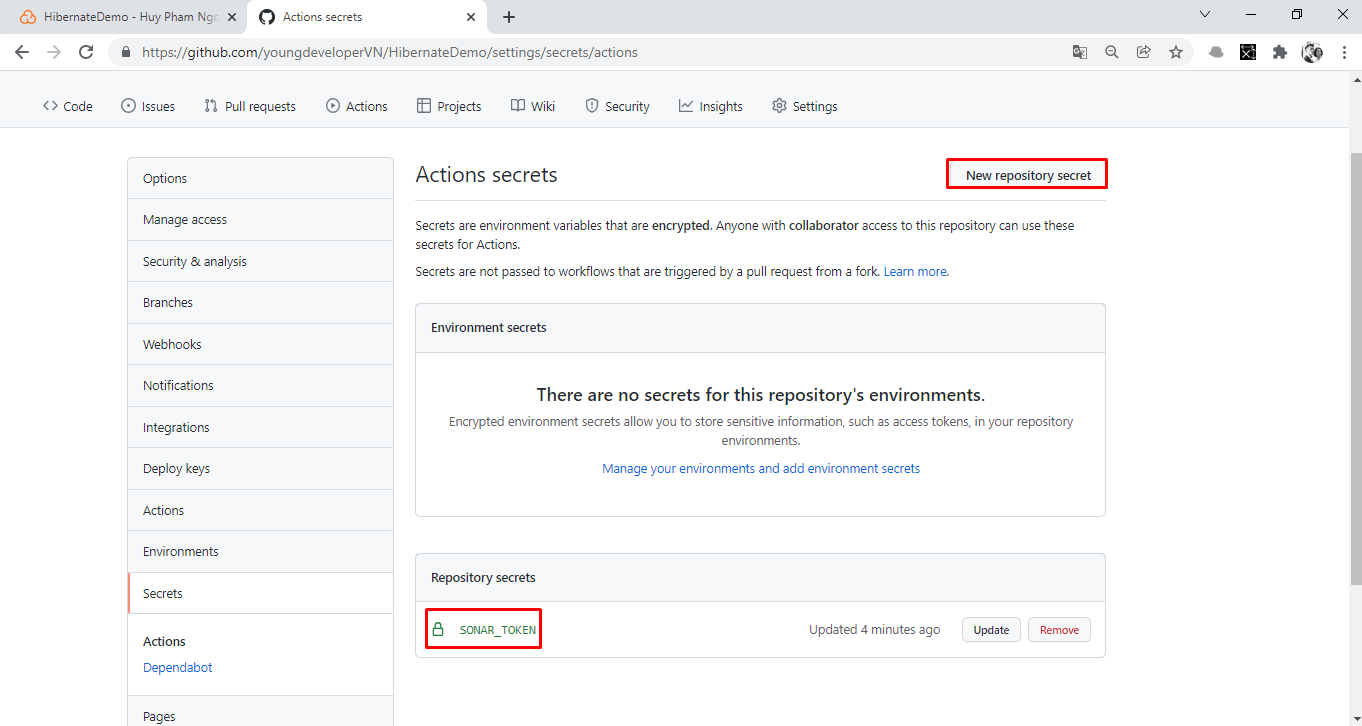
Tiếp theo chúng ta sẽ lựa chọn phương thức phân tích Analysis Method



## Cấu hình Secret Key trên Github

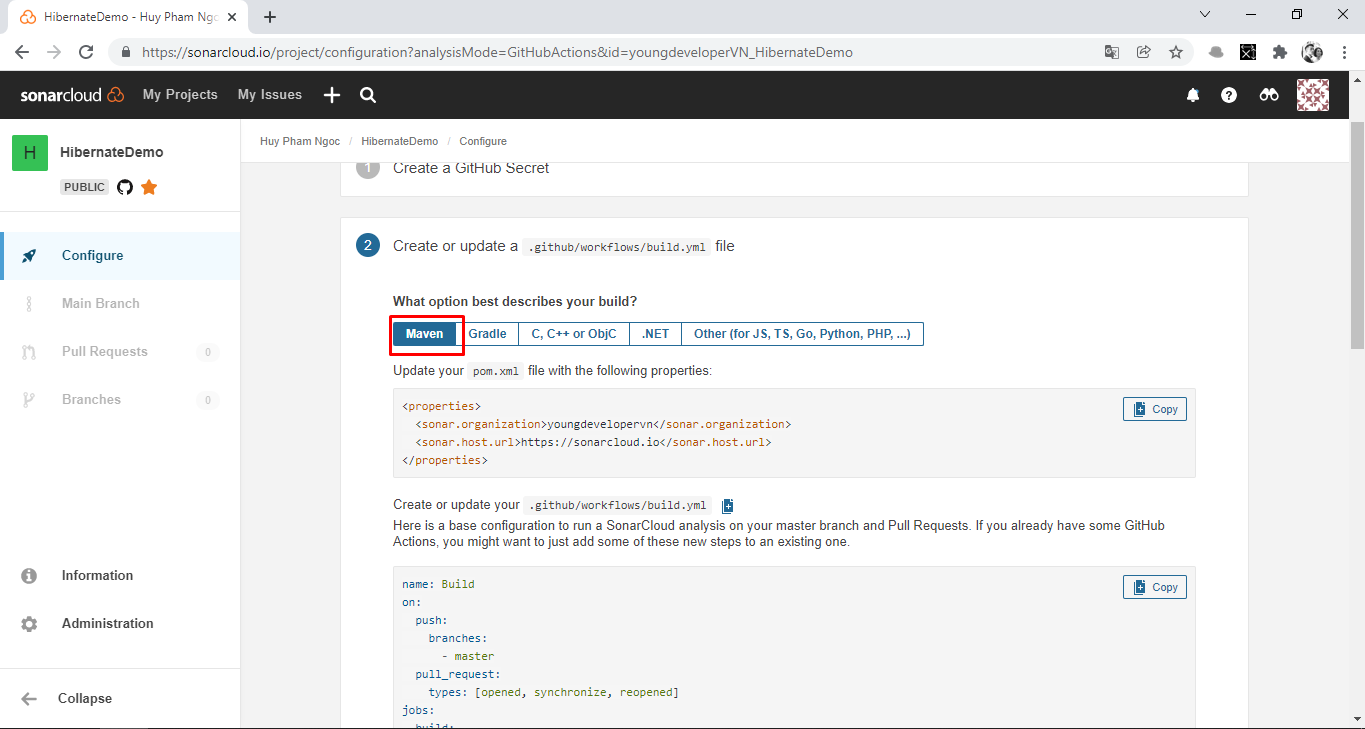
Hãy lựa chọn Github Actions và phần 1 sẽ cho các bạn một Secret Key.

Hãy Copy key và cấu hình vào project

Đặt tên key đúng với SONAR\_TOKEN giá trị đúng với mã Secret mà Sonar cho bạn

## Cấu hình POM.xml

Sau khi bước 1 là cấu hình Secret Key xong thì hãy Sửa lại file Pom.xml trong Github và thêm các thuộc tính mà Sonar gửi bạn



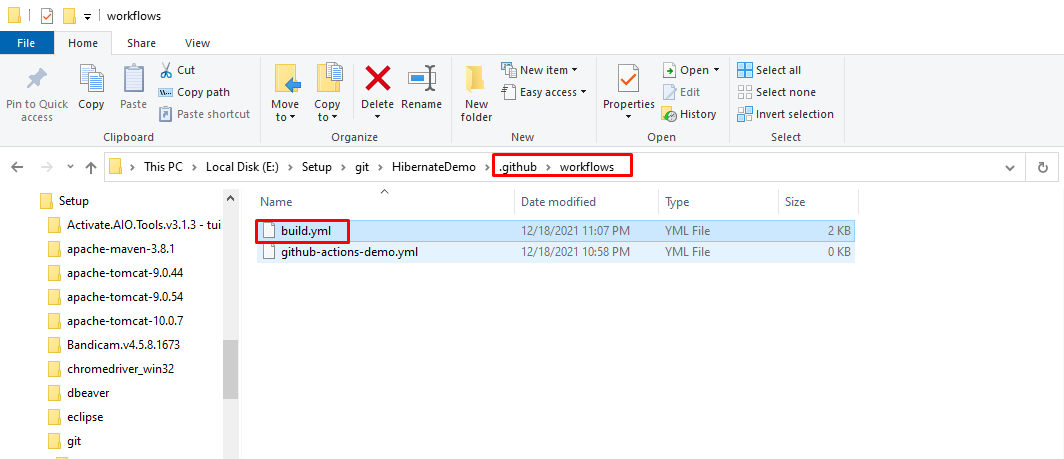
Trong phần này chúng ta có 2 file cần update

Pom.xml là file của Project chúng ta cần Dependency của Project

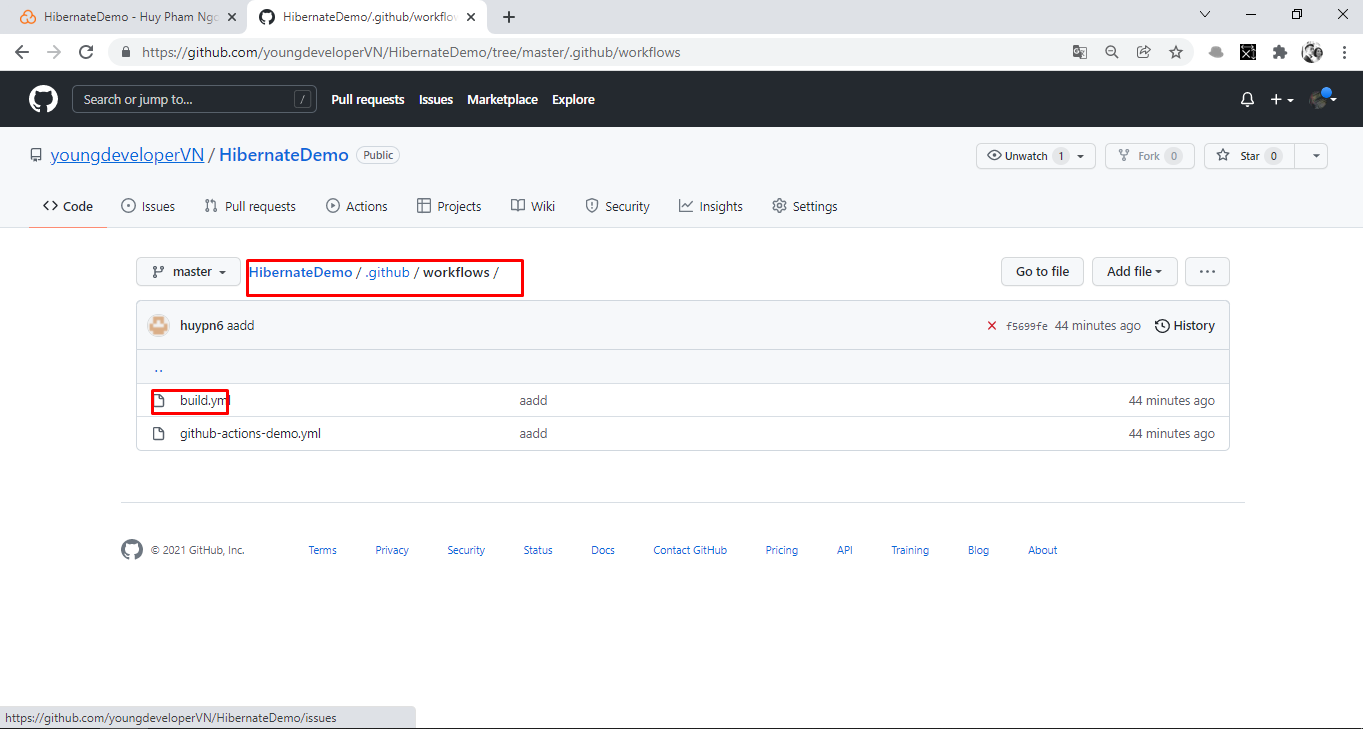
|  |
| --- |
| <properties>  <sonar.organization>youngdevelopervn</sonar.organization>  <sonar.host.url>https://sonarcloud.io</sonar.host.url>  </properties> |

## Tạo file build.yml

Trình build của Github sẽ cho đọc file build.yml để phân tích tự động sau mỗi lần bạn commit thay đổi code



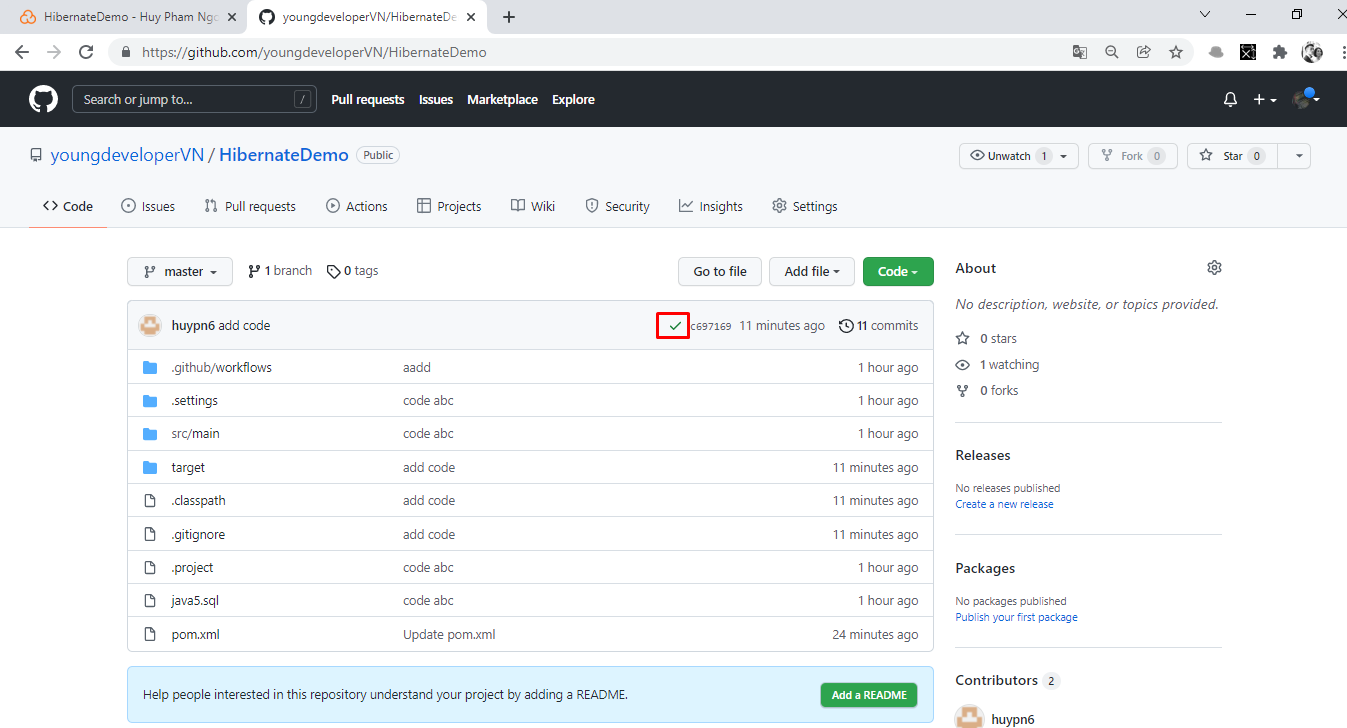
Hãy clone code thêm file và Push code ngược lại lên Github

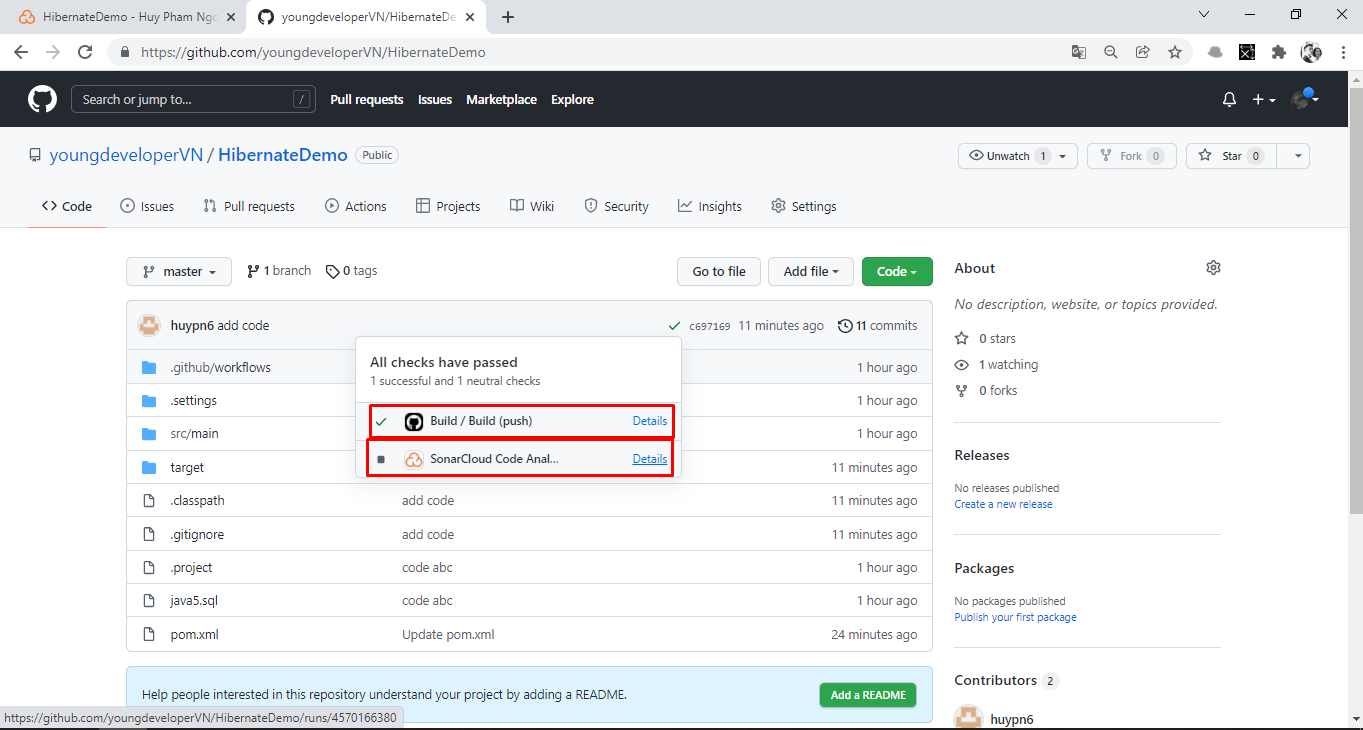


Sau đó thì chờ đợi Github sẽ Scan và load khi build.

## Kiểm tra Github build code

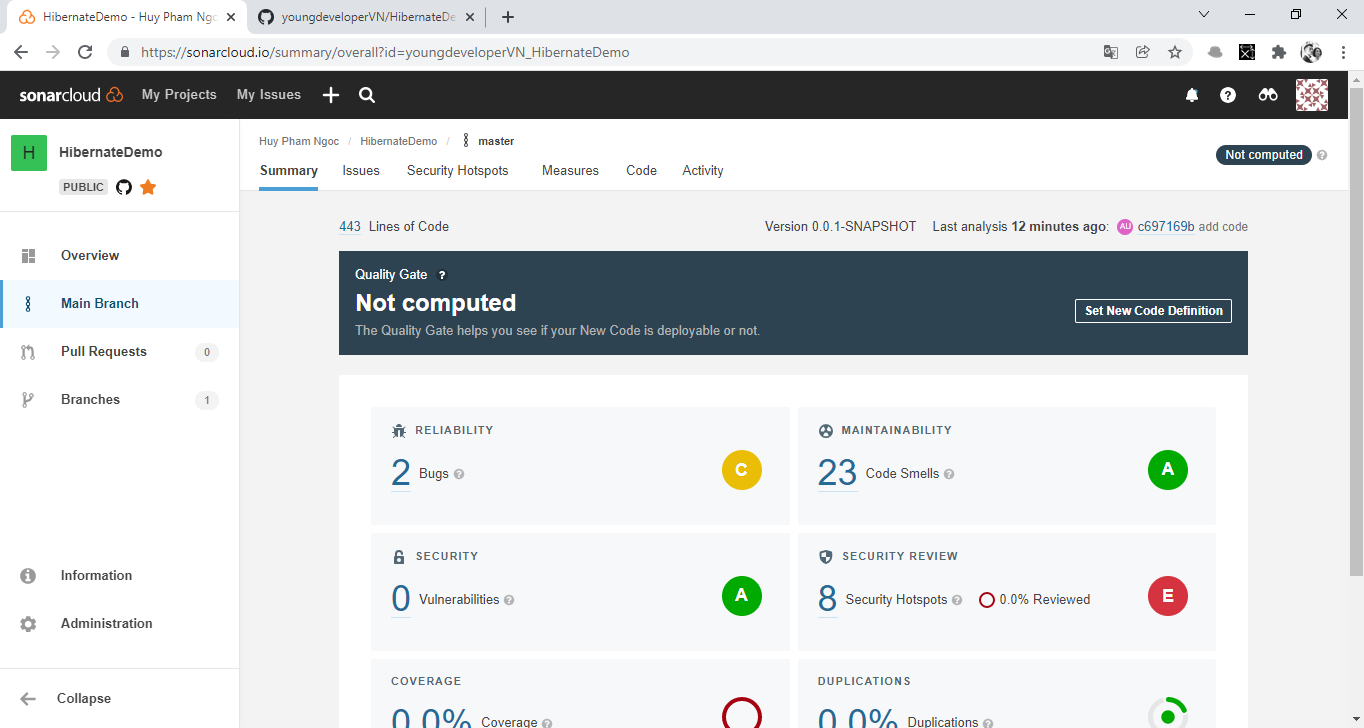
Sau mỗi lần commit code thì chúng ta sẽ được build file và kiểm tra bằng cách





Có thể lựa chọn detail để xem thông tin thành công hay không

Kiểm tra với phân tích code tự động bên Sonar Queue

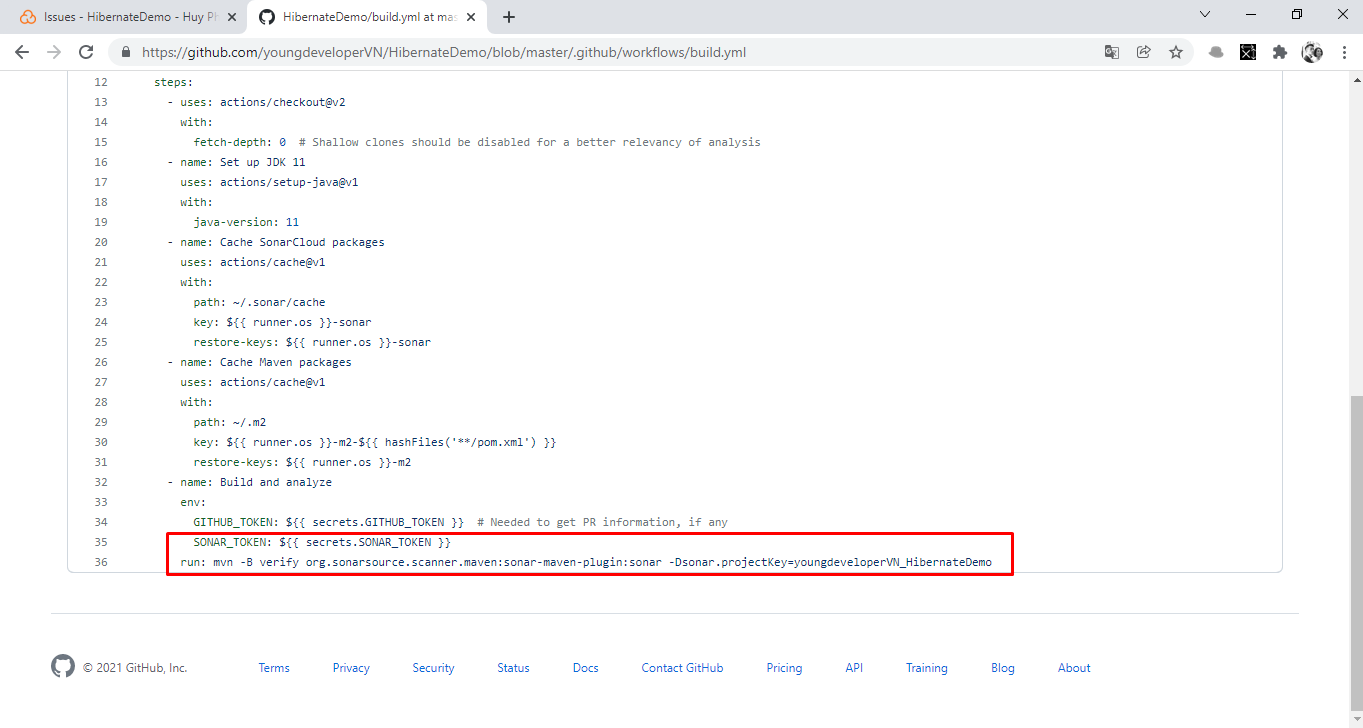


Việc phân tích code được diễn ra tự động và nó tự hiểu là code Java khi chúng ta Setting Maven Project

## Vì sau Github có thể build tự động tới Sonar.

Các trình build code tại Github sẽ build khi đọc file Build.xml

Tại file này chứa lệnh Build



Chính vì thế Github có thể tự động build và Sonar Queue cũng sẽ nắm bắt được các vấn đề code của bạn

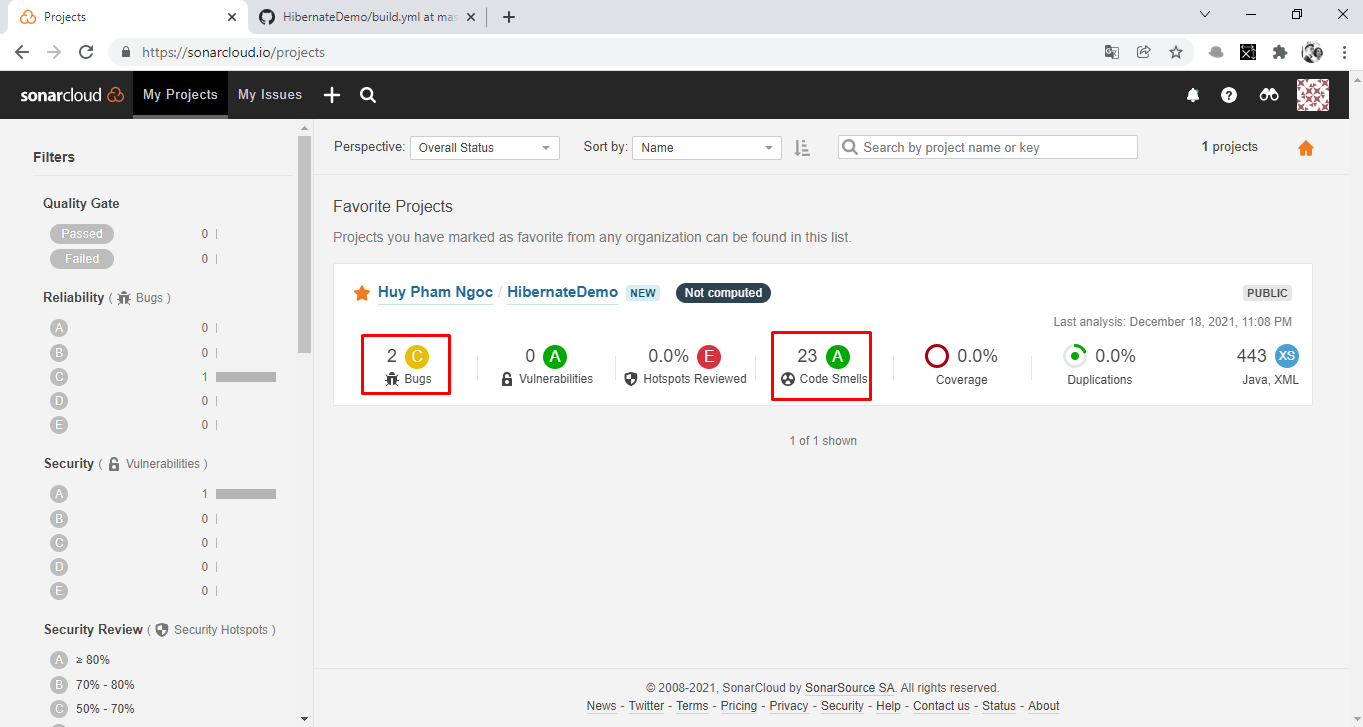
# Cách phân tích code với Sonar

Việc Sonar phân tích khá là đơn giản và tự động.

Chúng ta chỉ cần quan tâm tới việc Project chúng ta có vấn đề gì về Code

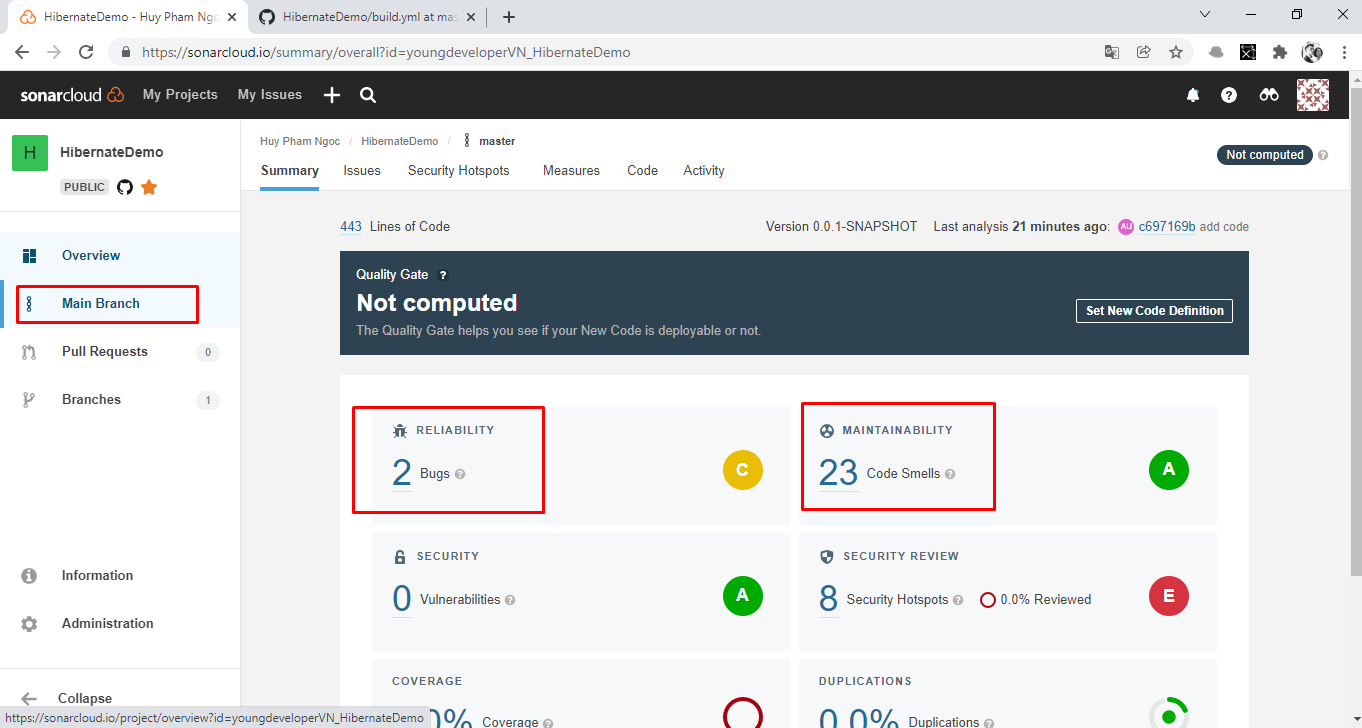
Các vấn đề của project mô tả khá chi tiết

Hãy truy cập link <https://sonarcloud.io/projects>

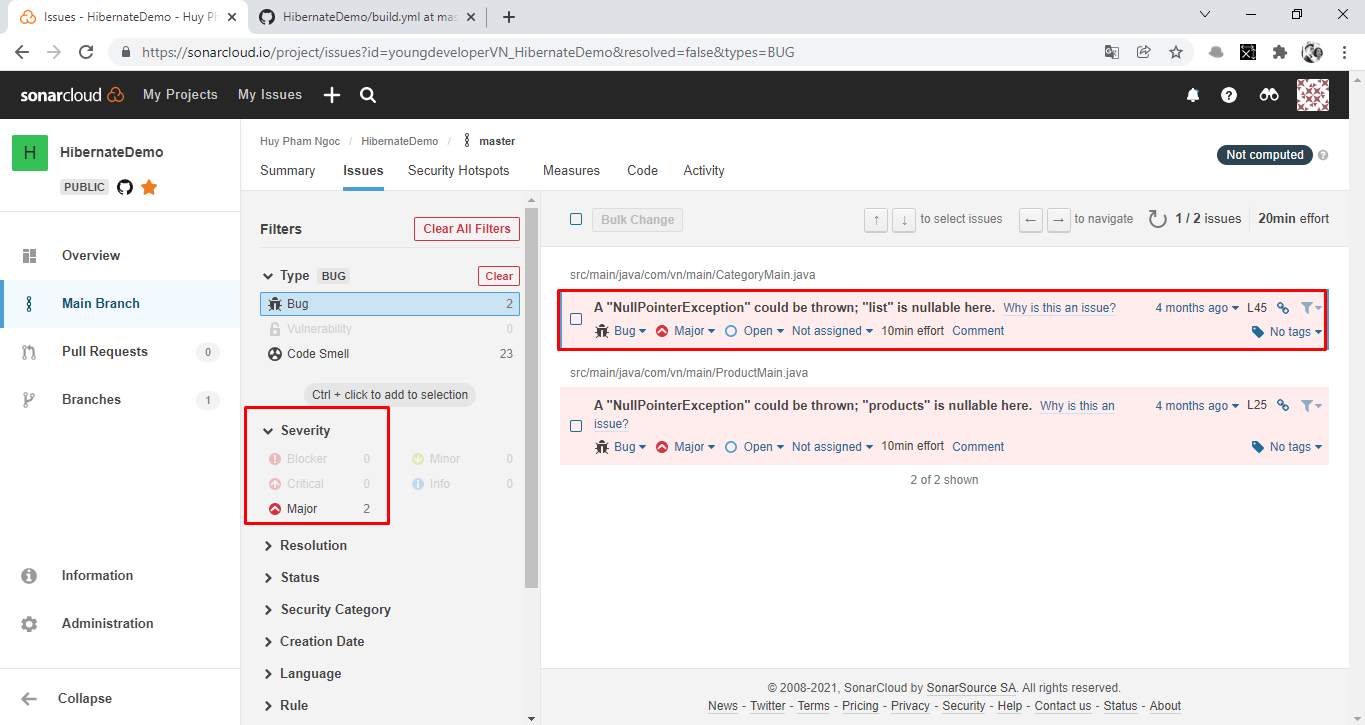


## Xem chi tiết các Bug

Hãy click vào project và lựa chọn



Chúng tao có thể xem được chi tiết các Bug và vấn đề code gặp phải



## Các Severity cần biết

Hãy ưu tiên xử lý các Severity như sau

* Blocker
* Critical
* Major
* Minor
* Info

Các bug viết khá chi tiết để các bạn có thể Fix

Các rule và các Serverity được mô tả khá chi tiết trong document của Sonar

<https://docs.sonarqube.org/latest/user-guide/rules/>

# Xem bug, ưu tiên bug và fix bug

## Kiểm tra sự thời gian phân tích code

Kiểm tra xem code đã được Sonar mới phân tích chưa, có khớp với code đã up lên chưa

Việc này tuy đơn giản nhưng khá quan trọng và việc sửa nó sẽ chính xác được.

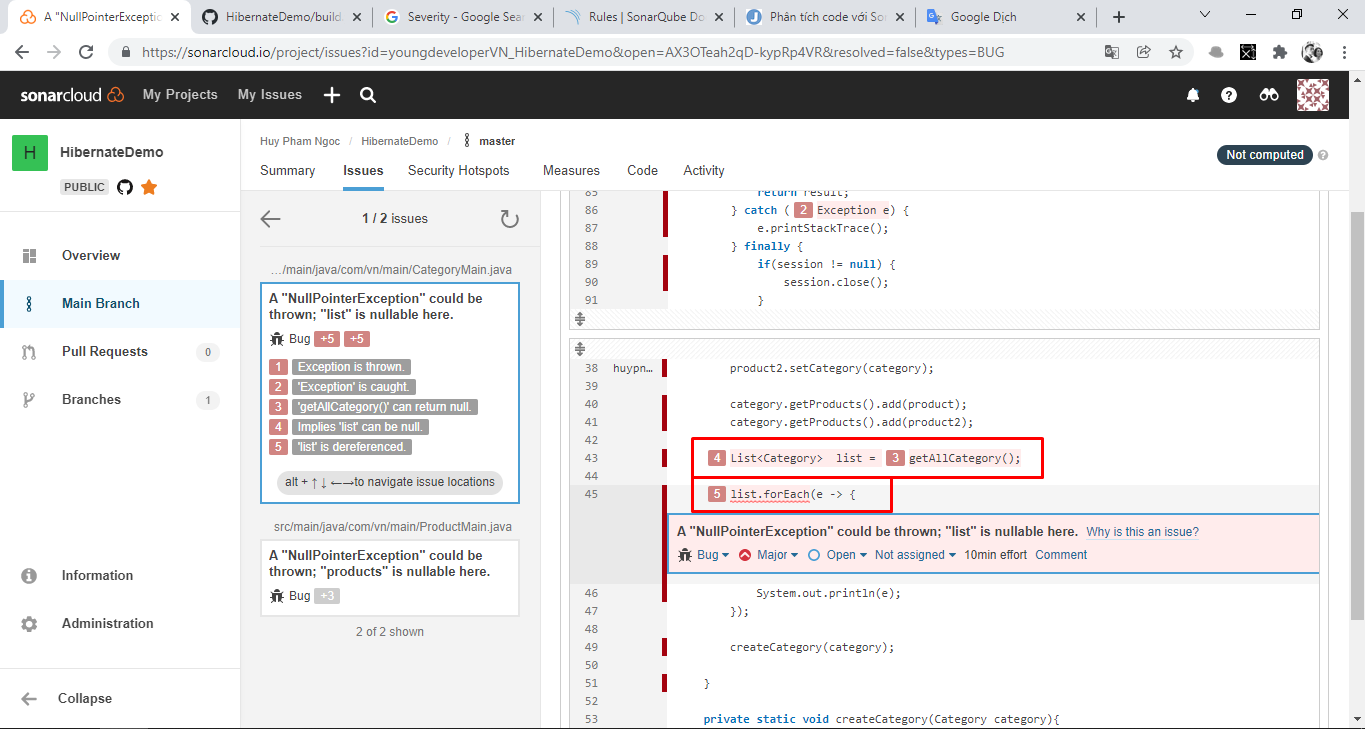
## Xem bug, hiểu bug

Ưu tiên các bug về code smells vì cái này khá dễ gặp

Tiếp theo ưu tiên các Bug: Block, Critical, Major vì các bug này thường là bug quan trọng

Xem tiêu đề và chi tiết bug

## Ví dụ 1

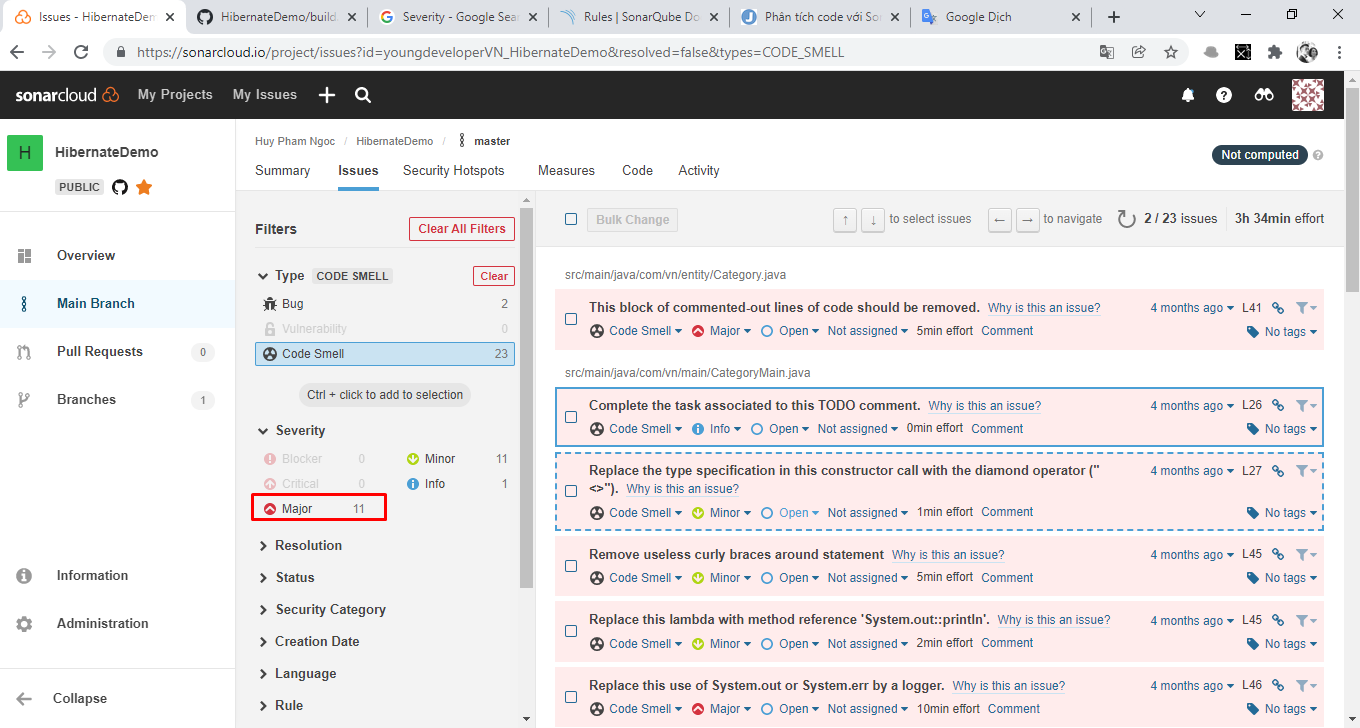


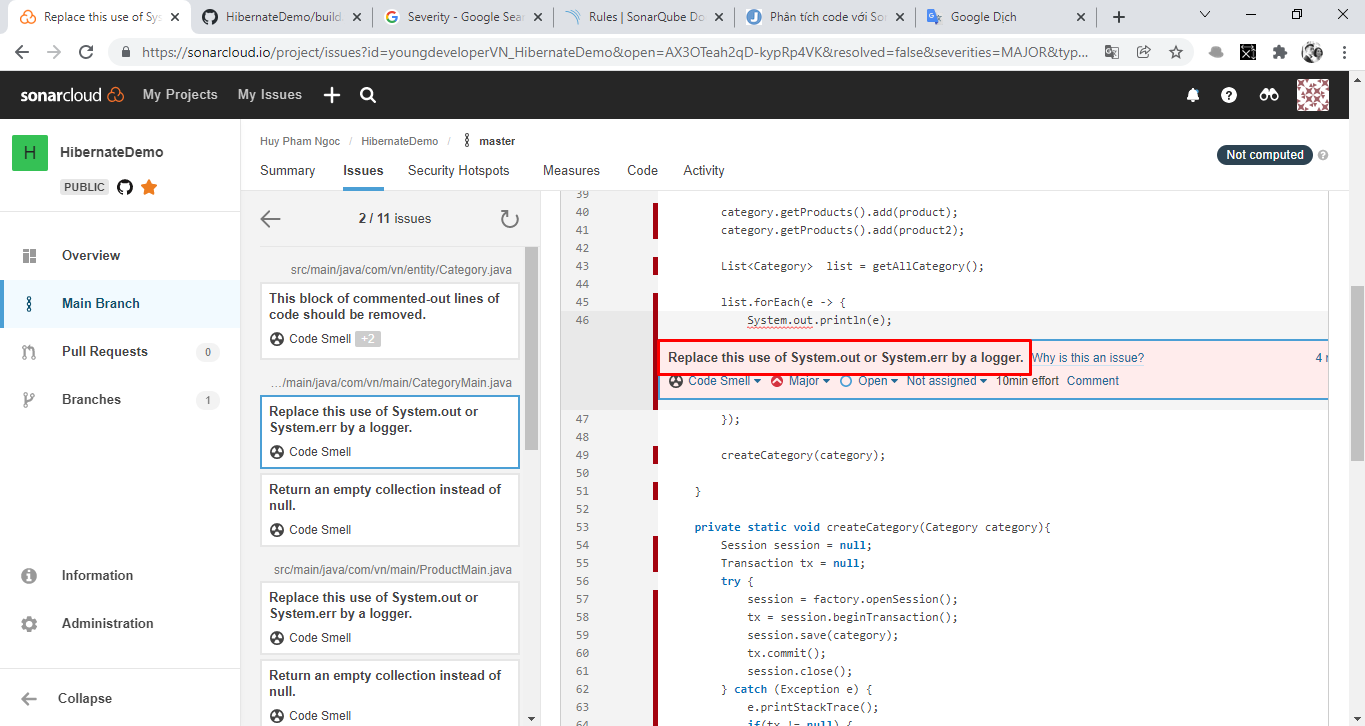
Việc đọc dữ liệu từ 3 có thể làm biến list có thể bị null và khiến dòng 5 sẽ bị lỗi.

Mọi người xem detail của bug có thể phân tích đươc

## Ví dụ 2

Lựa chọn Major fix bug (11 như trong ảnh)





Code cần chuyển sang Logger chứ không dùng System.out