



Departamento de Lenguajes y  
Sistemas Informáticos

# Uso de Github y métricas software

## Arquitectura - Práctica 1

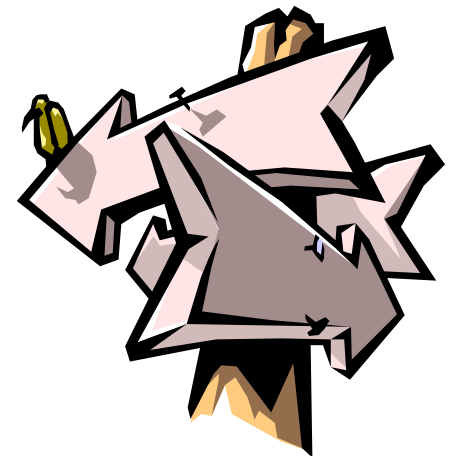


Arquitectura e Integración de Sistemas Software

2020/2021

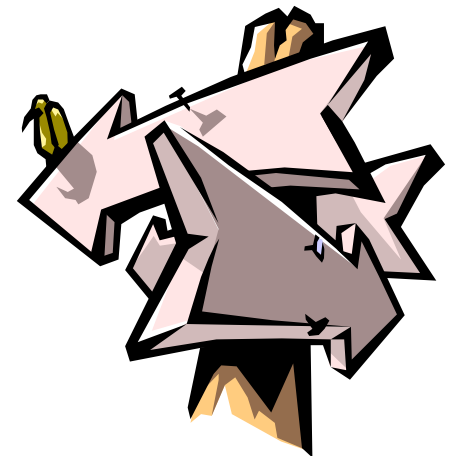
# Índice

- Introducción
- Uso de Github
- Métricas de arquitectura software



# Índice

- **Introducción**
- Uso de Github
- Métricas de arquitectura software



## Git básico

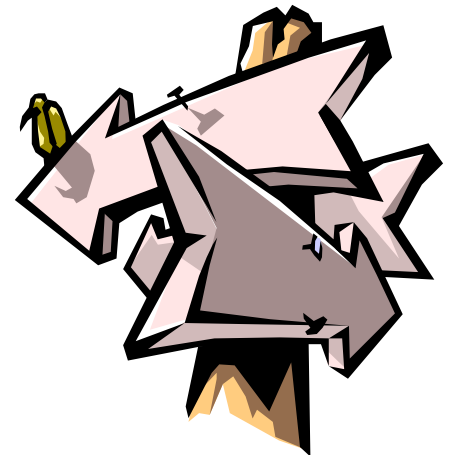
- Copia de trabajo & Clone
- Commit
- Push, Fetch, & Pull
- Conflicto
- Merge
- Fast-forward

# GitHub



# Índice

- Introducción
- **Uso de Github**
  - Organizaciones y equipos
  - Repositorios plantilla y forks
  - Issues
  - Pull requests
- Métricas de arquitectura software



## Organizaciones (y equipos)

- Las organizaciones son **cuentas compartidas** donde las empresas y los grupos de trabajo de código abierto pueden **colaborar en varios proyectos** a la vez.
- El *propietario* y los *administradores* pueden administrar el acceso de los **miembros** de la organización a la información y los proyectos.
- Los miembros de la organización pueden estructurarse formando en **equipos** que reflejen la estructura de la empresa o grupo, con permisos de acceso en comunes para el equipo.

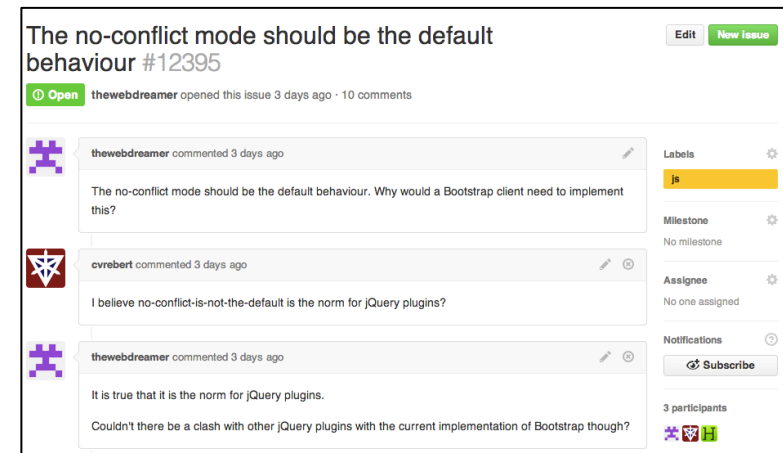
## Forks

- Un **fork** en el contexto del desarrollo software supone copiar un proyecto y partir de otro, para poder hacerle modificaciones.
- En repositorios **Git**, supone hacer una copia propia del proyecto que:
  - **Queda ligada** mediante una referencia **al proyecto original**
  - Tenemos permisos de administración sobre la copia.
  - Podemos **transferir (pull) cambios** desde el proyecto general a nuestro fork para mantenerlo actualizado
  - Podemos hacer **pull requests** desde nuestro proyecto al original



# Issues

- En github una Issue suele asociarse a una tarea:
  - una mejora a incorporar,
  - un error a resolver
  - Documentación a generar/escribir
- Una issue tiene una discusión/foro asociado y puede ser comentada, compartida, referenciada desde otra, etc.
- Las issues están por defecto abiertas y cuando se terminan las tareas se cierran.



## Issues (II)

- La issues pueden categorizarse mediante :
  - **Labels:** una issue puede tener asociadas varias etiquetas, que permiten establecer una categorización ej: bug, feature, css, ...
  - **Assignee:** Es el miembro del proyecto involucrado en que la realización de las tarea avance
  - **Milestone:** Son grupos de issues con fecha de fin que se corresponde con una fase del proyecto. Una issue puede estar asociada a un único hito.

The screenshot shows a GitHub issue page for the title "Open modal is shifting body content to the left #9855". At the top right are "Edit" and "New Issue" buttons. Below the title, a green "Open" status icon is followed by the text "mat0r opened this issue 5 months ago · 88 comments". The main content area displays two comments. The first comment, by user "mat0r", describes a problem where content shifts left on macOS when a modal is active, with a link to the Bootstrap modal documentation. The second comment, by user "jamescostian", confirms the issue on Chrome for Linux and mentions the "3.0.0-wip" branch. On the right side, the "Labels" section shows three tags: "confirmed" (red), "css" (purple), and "js" (yellow). The "Milestone" section shows a green bar for "v3.1.0". The "Assignee" section shows "No one assigned".

< Open modal is shifting body content to the left #9855 Edit New Issue

Open mat0r opened this issue 5 months ago · 88 comments

mat0r commented 5 months ago

When launching the modal component (<http://getbootstrap.com/javascript/#modals>) the entire content will slightly move to the left on mac OS (haven't tried it on windows yet). With the active modal the scrollbar seem to disappear, while the content width still changes.

You can observer the problem on the bootstrap page

jamescostian commented 5 months ago

I can confirm that I see this also on Chrome for Linux, both on getbootstrap.com and on the 3.0.0-wip branch

Labels

- confirmed
- css
- js

Milestone

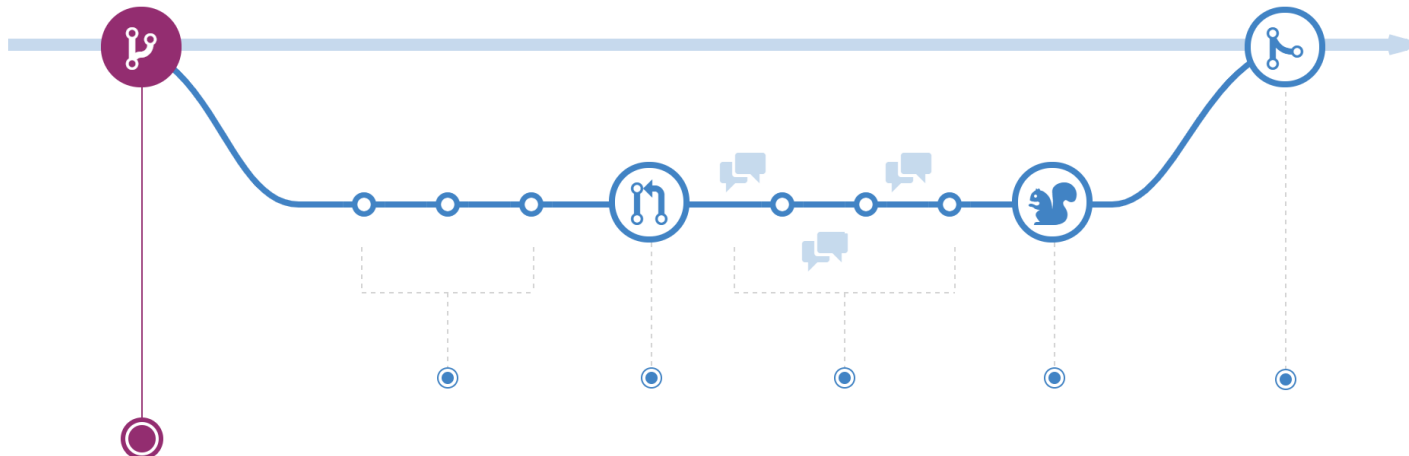
- v3.1.0

Assignee

No one assigned

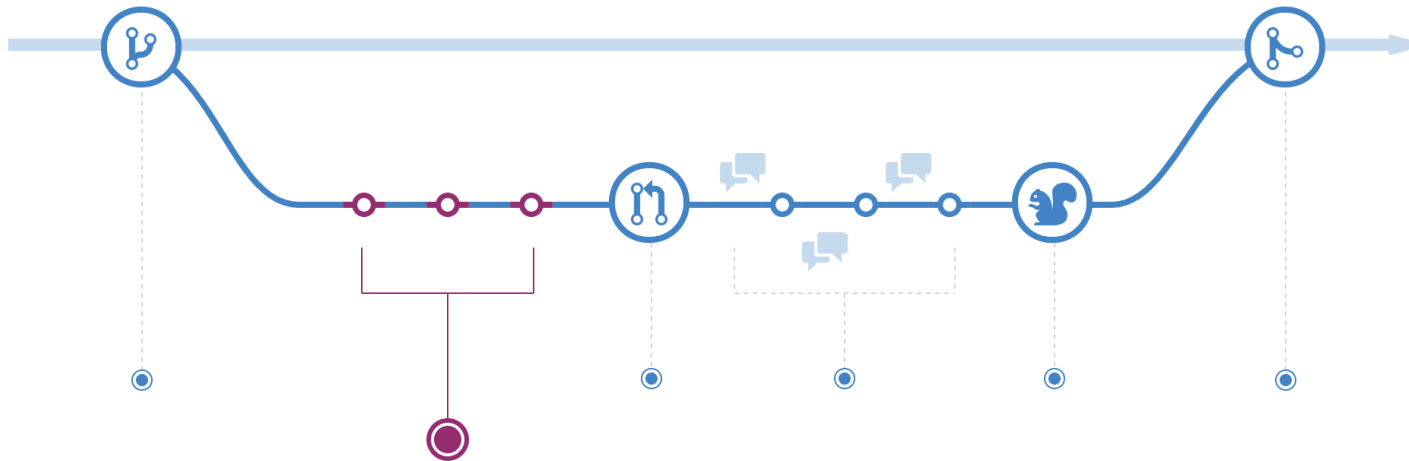
## El flujo de trabajo en Github

- En github se propone el siguiente ciclo de vida para contribuir a un proyecto:
  1. **Crear una rama o fork para trabajar en él**
  2. Confirmar nuestros cambios (commit & push)
  3. Abrir una pull-request para discutir y revisar los cambios
  4. Probar e incorporar (mergerar) tus cambios



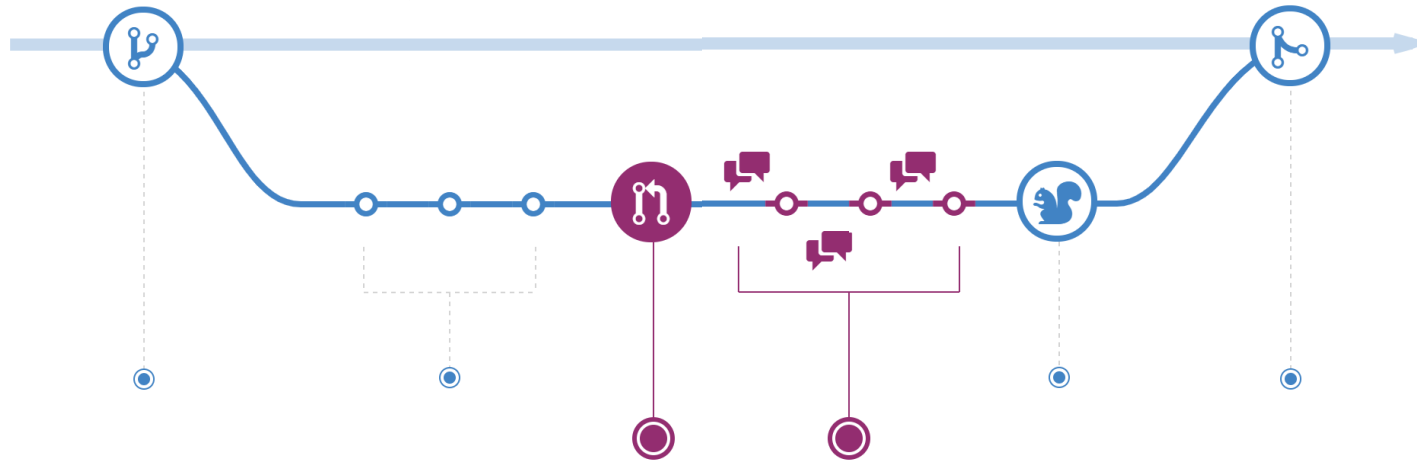
# El flujo de trabajo en Github

- En github se propone el siguiente ciclo de vida para contribuir a un proyecto:
  1. Crear una rama o fork
  - 2. Confirmar nuestros cambios (commit & push)**
  3. Abrir una pull-request para discutir y revisar los cambios
  4. Probar e incorporar (mergerar) tus cambios



# El flujo de trabajo en Github

- En github se propone el siguiente ciclo de vida para contribuir a un proyecto:
  1. Crear una rama o fork para trabajar en él
  2. Confirmar nuestros cambios (commit & push)
  - 3. Abrir una pull-request para discutir y revisar los cambios**
  4. Probar e incorporar (mergerar) tus cambios

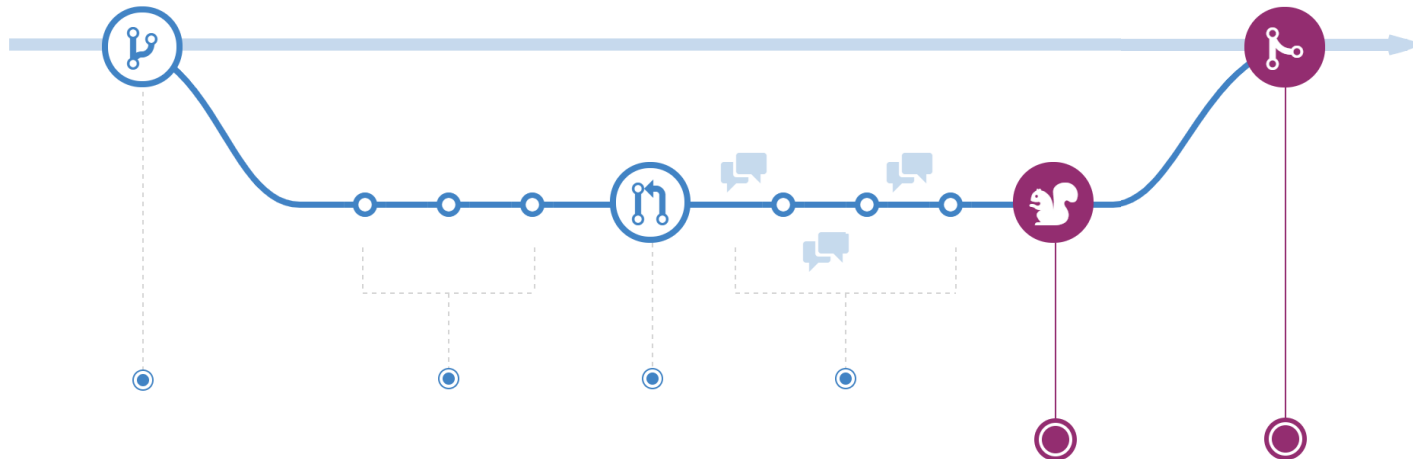


## Pull Requests

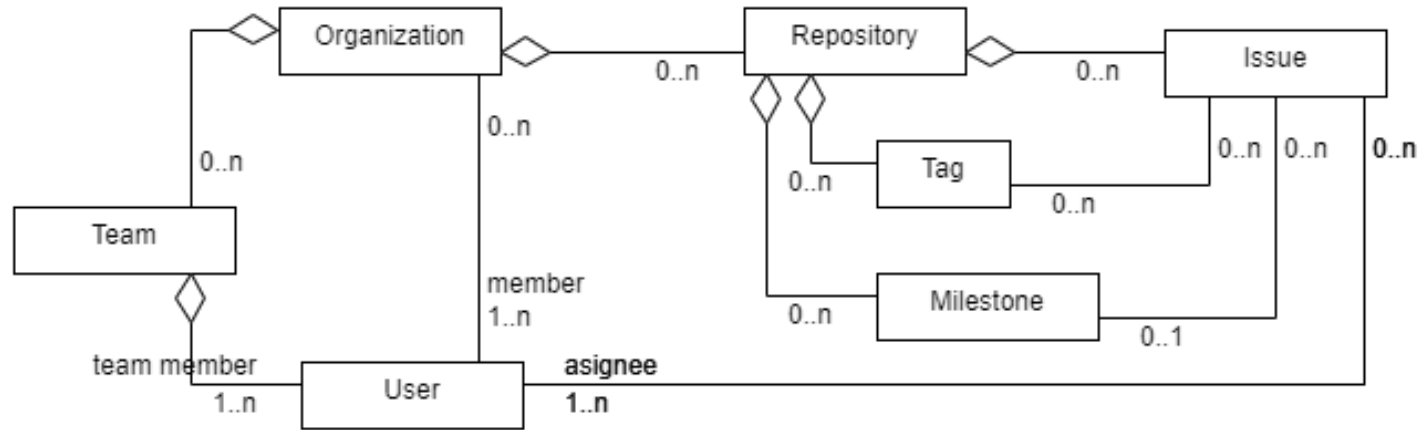
- Proporcionan una manera de notificar a los gestores/responsables de un proyecto los cambios que queremos que consideren.
- Arranca un debate/evaluación sobre nuestros commits, y solicita la incorporación de dichos cambios. Evidentemente puede ser rechazada.
- Si la solicitud es evaluada correctamente, se realizará un *merge* de los commits en la rama o proyecto, y nuestros cambios formarán parte del código oficialmente. Seremos contribuidores al proyecto.

## El flujo de trabajo en Github

- En github se propone el siguiente ciclo de vida para contribuir a un proyecto:
  1. Crear una rama o fork para trabajar en él
  2. Confirmar nuestros cambios (commit & push)
  3. Abrir una pull-request para discutir y revisar lo cambios
  4. **Probar e incorporar (mergerar) tus cambios**



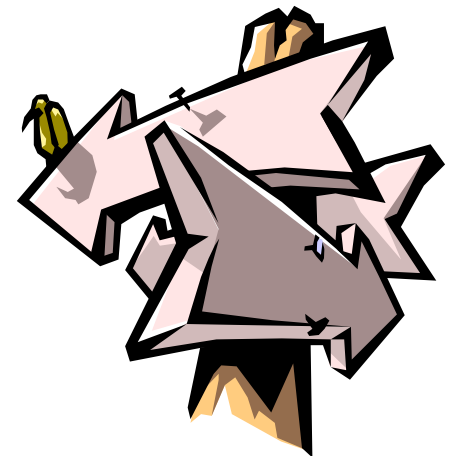
# Diagrama conceptual general sobre GitHub





# Índice

- Introducción
- Uso de Github
- **Métricas de arquitectura software**




## Métricas software en Github

- Podemos usar la plataforma de github para sacar algunas métricas interesantes como por ejemplo:
  - Número de forks y estrellas.
  - Número total de issues e issues abiertas.
  - Número total/% de issues por etiqueta.
  - Número de contribuidores, commits, y releases.
  - Etc.

# Métricas de software en Github

 [gii-is-AISS](#) / [extjwnl](#)

forked from [extjwnl/extjwnl](#)

 Watch ▾

0


 Star

0

 Fork

43

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

 master ▾

 1 branch

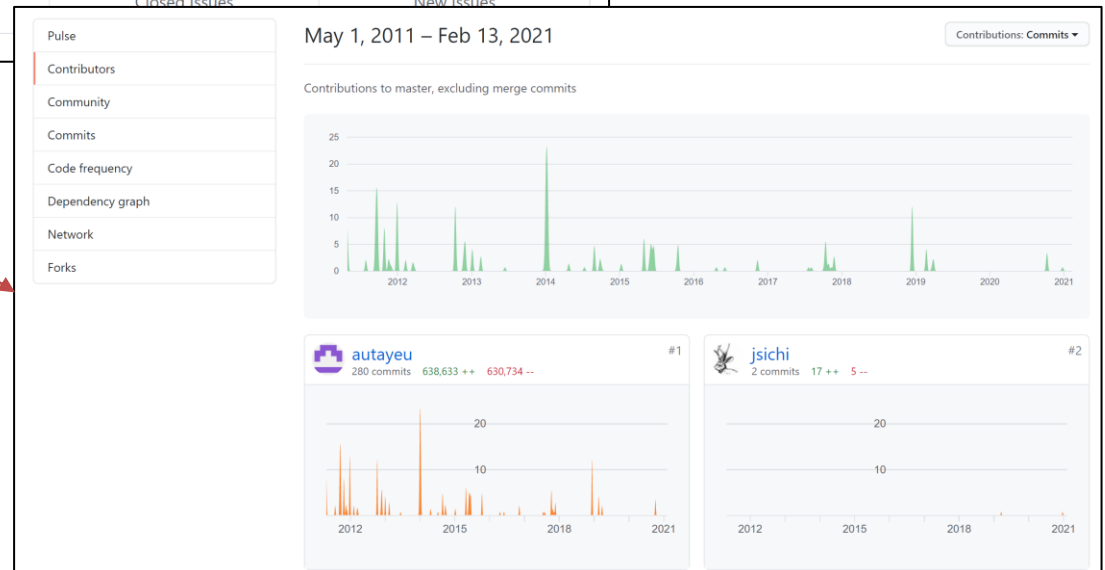
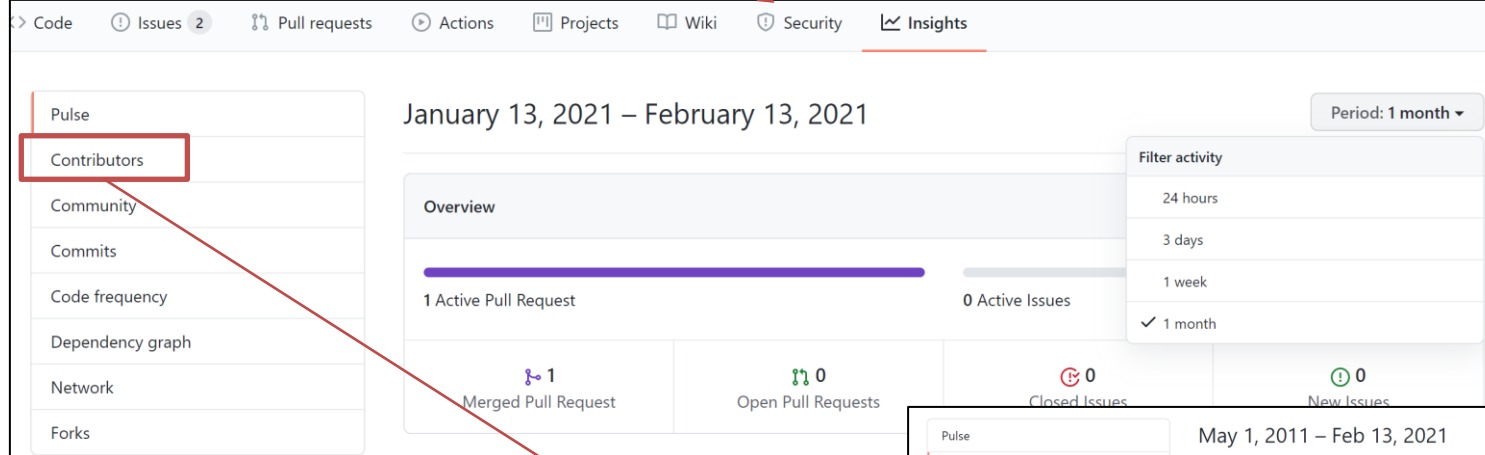
 13 tags

[Go to file](#)

[Add file ▾](#)

[Code ▾](#)

[About](#)



# Enlaces

- Guías de uso de github:

<https://guides.github.com/>

- Documentación oficial de Sonar Cloud

<https://sonarcloud.io/documentation>

# Disclaimer and Terms of Use

All material displayed on this presentation is for teaching and personal use only.

Many of the images that have been used in the presentation are Royalty Free images taken from <http://www.everystockphoto.com/>. Other images have been sourced directly from the Public domain, from where in most cases it is unclear whether copyright has been explicitly claimed. Our intention is not to infringe any artist's copyright, whether written or visual. We do not claim ownership of any image that has been freely obtained from the public domain. In the event that we have freely obtained an image or quotation that has been placed in the public domain and in doing so have inadvertently used a copyrighted image without the copyright holder's express permission we ask that the copyright holder writes to us directly, upon which we will contact the copyright holder to request full written permission to use the quote or images.