



ARQUITECTURA DEL SOFTWARE

2025-26

Jose Emilio Labra Gayo
Pablo González
Diego Martín
Celia Melendi

Laboratorio 1

Introducción a la práctica
Organización de equipos
Git
Github



Escuela de
Ingeniería
Informática



Universidad de Oviedo



Introducción a la práctica

¿Qué vamos a hacer?

Diseñar y desarrollar un juego online basado en el juego Y.

¿Qué **recursos** debemos utilizar?

- El website de Arquitectura del Software donde se encontrará toda la documentación de la asignatura.
- Campus virtual.
- La especificación de YOVI.
- Github del proyecto repositorios .

¿Cómo se va a **evaluar** la práctica?

70% - Trabajo en grupo

30% - Trabajo individual

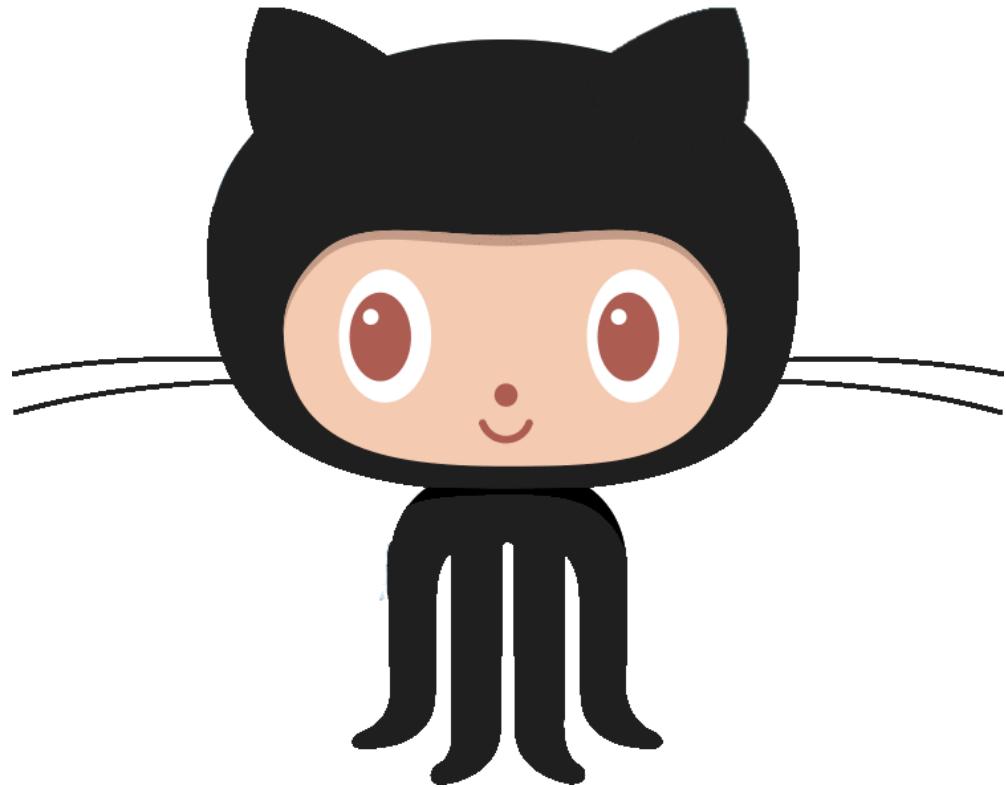
● Organización de equipos

Actas de reuniones

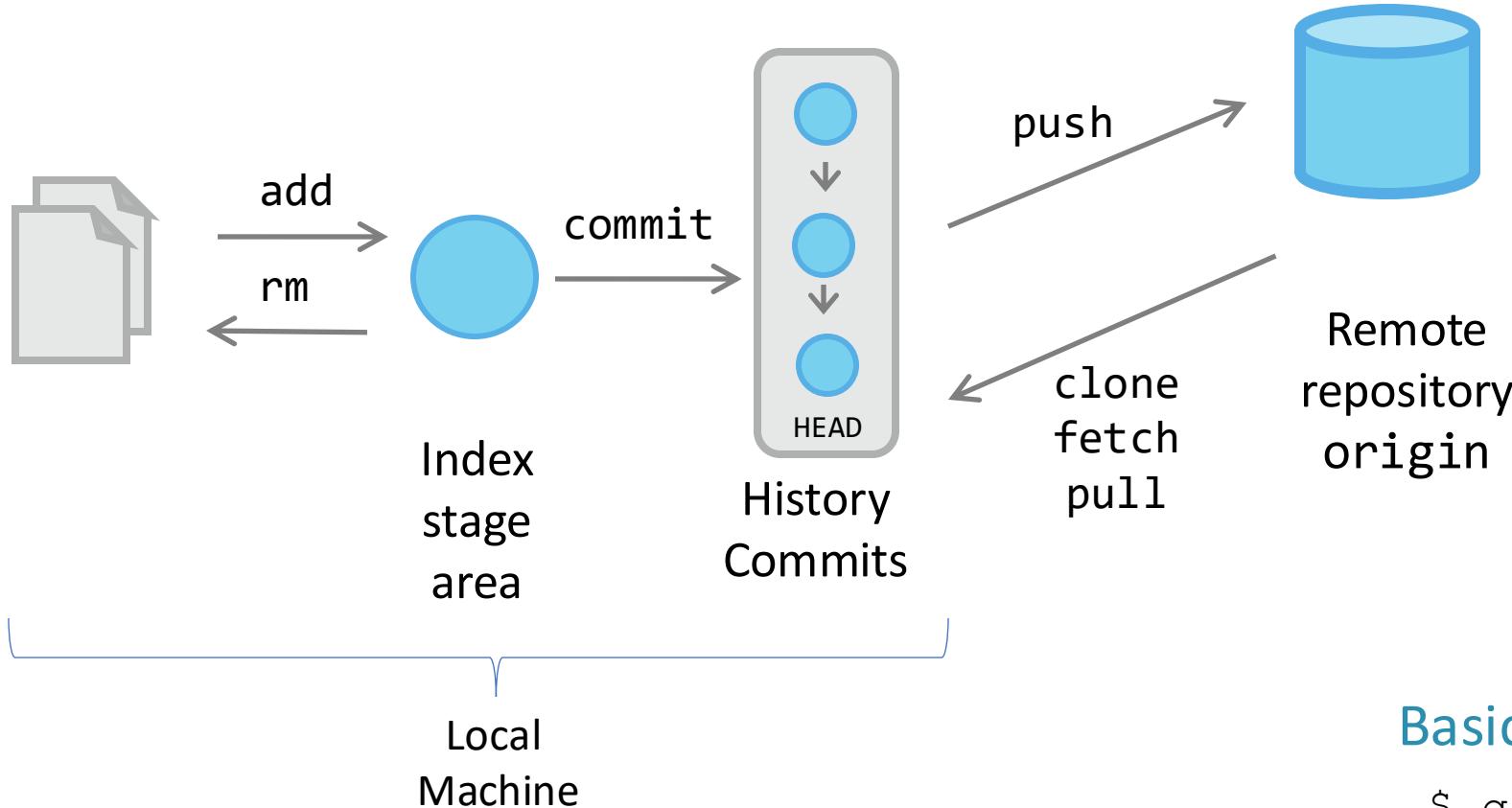
- Clases de práctica == reunión.
 - Se pueden hacer otras reuniones aparte de las clases de laboratorio
- **Obligatorio** tomar acta de las reuniones.
- Se utilizará la sección “wiki” del repositorio para las actas.
- Formato mínimo **obligatorio**:
 - Fecha 
 - Lista de participantes 
 - Acuerdo en el Reparto de trabajo adoptados para la próxima sesión (issues abiertos) 
 - Revisión de estado de tareas en reuniones anteriores
 - Enlaces a **Issues** y **Pull requests** 
 - Breve descripción decisiones tomadas
 - Preferible enlazar a registros de decisiones arquitectónicos (<https://adr.github.io/>)



Git



Git



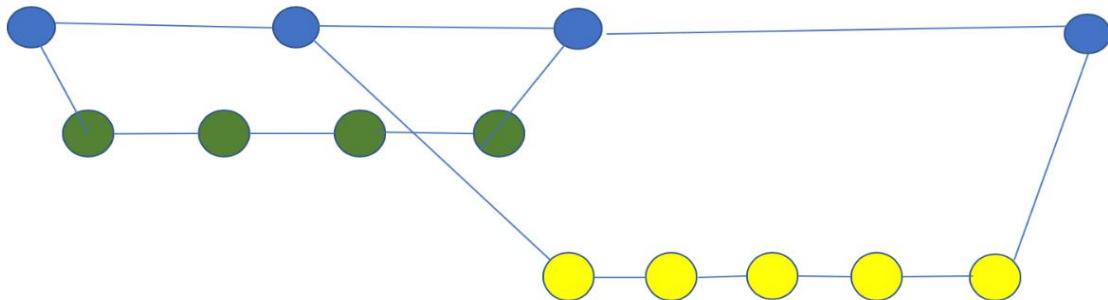
Basic workflow

```
$ git init
$ git clone urlRepository
$ git add .
$ git commit -m "message"
$ git push origin master
```



Git

Trabajar con ramas



- Crear una rama:
`$ git checkout -b rama1`
- Ver en qué rama estamos
`$ git branch`
- Cambiar de rama
`$ git checkout master`
- Ver los cambios entre ramas
`$ git diff --stat master rama1`
- Fusionar ramas
`$ git checkout master`
`$ git merge --no-ff rama1`
- Eliminar la rama
`$ git branch -d rama1`
- Crear rama **develop**
`$ git checkout -b develop`
- Añadir la rama a nuestro repositorio
`$ git push origin develop`

Git

Estrategias de ramificación

- Todas las estrategias se basan esencialmente en la forma en la que van a crear o no ramas y fusionarlas a la rama principal.
- **No existe** una estrategia perfecta 😢. Depende de muchos factores desde la veteranía del equipo hasta las tecnologías utilizadas o las políticas de la organización.
 - <https://martinfowler.com/articles/branching-patterns.html#Trunk-basedDevelopment>
- Varias estrategias:
 - Git Flow. Vincent Driessen en 2010: [A successful Git branching model](#)” (Tiene código comandos git).
 - [GitHub Flow](#).
 - Trunk-based development: <https://trunkbaseddevelopment.com/>



Pull Request

Pull Request es la aportación más importante que ha hecho GitHub:
Poder integrar a un repositorio código hecho por otros usuarios.

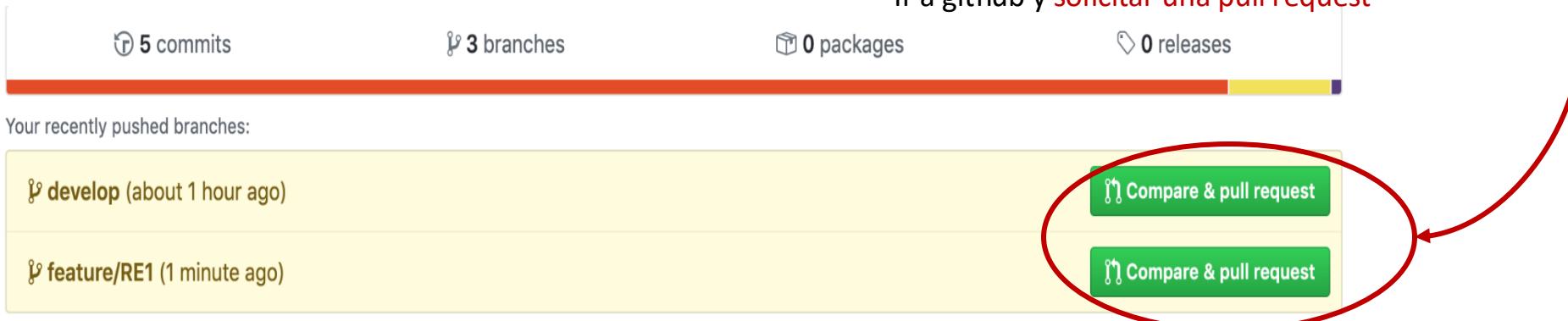
Pasos

- Crear la rama

```
$ git flow feature start RE1 develop
$ git checkout -b feature-RE1 develop
```
- Añade tu nombre en **README.md** en el apartado *Colaboradores*
- Subir los cambios en local

```
$ git add .
$ git commit
```
- Subir los cambios

```
$ git push --set-upstream origin feature-RE1
```
- Ir a **github** y **solicitar una pull request**





Pull request

The screenshot shows the GitHub interface for creating a pull request. The top navigation bar includes 'Code', 'Issues 4', 'Pull requests 2', 'Actions', 'Projects', 'Wiki', 'Security 1', 'Insights', and 'Settings'. The main section is titled 'Open a pull request' with the sub-instruction: 'Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#)'. The pull request configuration area has two dropdowns: 'base: master' (marked 1) and 'compare: feature-new-cool-thing' (marked 2). A note states: 'Able to merge. These branches can be automatically merged.' (marked 2). The pull request details box contains a user profile picture (marked 3), the title 'Fix problem when creating an user', and a rich text editor with 'Write' and 'Preview' tabs. The content area (marked 4) includes '# Changes' with the text 'Use uuid v4 instead a manual generated id with Math.random to avoid problems when using the id in some places.', '# Fixed issues' with the text 'Fixed #30 on generating new users', and an attachment instruction 'Attach files by dragging & dropping, selecting or pasting them.' At the bottom is a green 'Create pull request' button (marked 5).

Rama destino.

Añadir contexto adicional.

Rama que queremos fusionar

Crear pull request

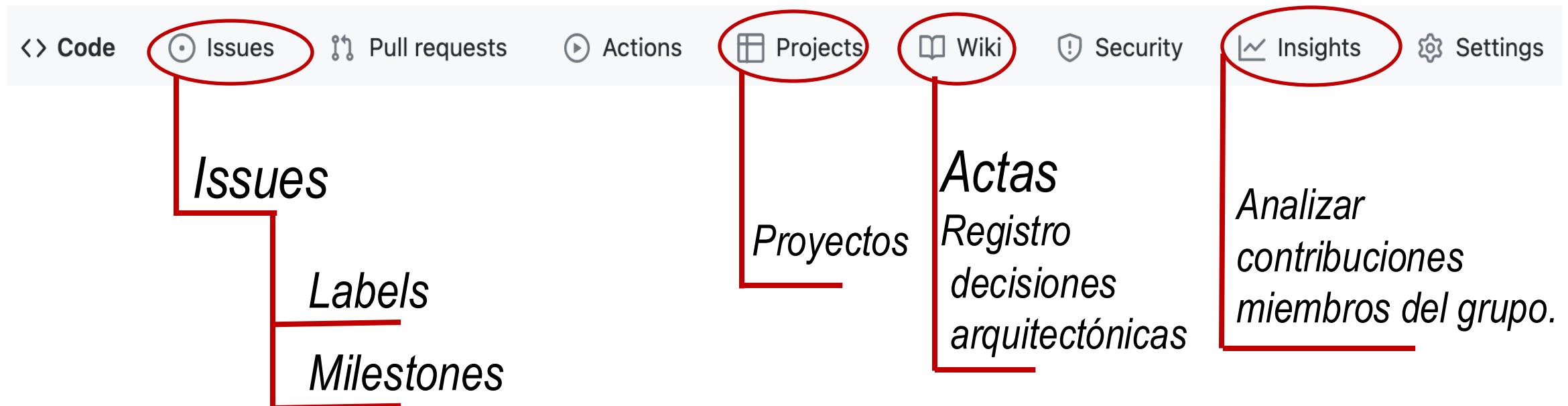


Ventajas de la gestión de proyectos

- Planificación del proyecto (**Futuro**).
- Controlar el estado actual del proyecto. (**Presente**)
 - Detectar cuellos de botella.
 - Carga de trabajo de los miembros del equipo.
 - Problemas actuales.
- Reporte del desempeño (**Pasado**).
 - Evaluación de la contribución de los miembros del equipo.



Herramientas de Github para gestión de proyectos





GitHub

root > Projects > MagiDoc - Development

+ Add column

Backlog	To Do	In Progress	Done
<ul style="list-style-type: none">Create Docker image for project: magidoc-admin-service#10 opened by jflory? enhancement priority:lowThrow errors for missing dependencies: magidoc-admin-service#11 opened by jflory? enhancement high priorityCurling results in empty response, server crashes: magidoc-admin-service#12 opened by jflory? bug priority:high priority:criticalServe aggregations by admin level: magidoc-admin-service#13 opened by mikelab enhancementSave API in Swagger Spec Has Extra /: magidoc-openapi#14 opened by shentao enhancementUse arguments to overwrite default configurations: magidoc-args#15 opened by jflory? enhancement priority:lowReturn hasProtoData in API response for schools: magidoc-openapi#16 opened by jflory? enhancementexpired token warning gets overridden by tokenHandler library and return fatal instead of just: magidoc-openapi#17 opened by mikelab enhancement	<ul style="list-style-type: none">Colombia shapefile (impl.json) should be fetched from api rather than be required in react app: magidoc-react#18 opened by mikelab enhancementFix "the_lines" issue in mock-app's public/lib.js: magidoc-react#19 opened by jflory? bugComment and refactor code in actions: select-country#20 opened by mikelab enhancementAdd disclaimer popup on load: magidoc-react#21 opened by mikelab enhancementRewire login code to work in production: magidoc-react#22 opened by mikelab enhancement	<ul style="list-style-type: none">Add contributing guidelines placeholder (closed #23) magidoc-react#23 opened by jflory? enhancementDetermine where to begin writing tests and refactoring: magidoc-react#24 opened by mikelab enhancementMap relationships between apps, hypers, and azure resources: magidoc#15 opened by mikelab enhancement internal organization priority:critMobility: Magi should display data for mobility being displayed: magidoc#16 opened by mikelab enhancement	<ul style="list-style-type: none">Resume test suite (run test-all), add comments: magidoc-openapi#17 opened by jflory? enhancementRefactor WebGLayers: magidoc-maps#17 opened by jflory? bugCreate production stage including deployment slots for Mobility Mapping: magidoc#18 opened by mikelab enhancement internal organization priority:lowSet up automatic CI testing for new pull requests: magidoc#19 opened by jflory? enhancement priority:highAdd loading spinner to mobility map: magidoc-maps#17 opened by mikelab enhancementAdd spinner mobility: magidoc-maps#18 opened by carloscarlos enhancementAdd CodeClimate maintainability + test coverage checks to CI: magidoc-openapi#18 opened by jflory? enhancementAdd TravisCI and CodeClimate tests: magidoc-openapi#19 opened by jflory? enhancement

Automated in Backlog Merge Automated in In progress Merge Automated in Done Merge



Tablero de proyecto

- Podemos crear un tablero KanBan.
- Muy Flexible. Podemos hacer un tablero muy simple (y de gestión manual)
- GitHub permite automatizar el flujo de trabajo a través de la gestión de Issues y pull requests.
- Podemos crear **diferentes** tableros (documentación, backend,...)

The screenshot shows the GitHub interface for managing projects. At the top, there are navigation links: Code, Issues, Pull requests, Actions, Projects (which is highlighted with a red oval), and Below these, there's a search bar with the query 'is:open'. A 'Projects' dropdown menu is open, showing a 'New project' button, which is also highlighted with a red oval. The main area displays a message: 'There aren't any projects yet'.

The screenshot shows the 'Create a new project' form. It starts with a heading 'Create a new project' and a sub-instruction: 'Coordinate, track, and update your work in one place, so projects stay transparent and on schedule.' Below this is a 'Project board name' field, which is highlighted with a red oval. There's also a 'Description (optional)' field with a text area. A large red arrow points from the 'New project' button in the GitHub interface to this 'Project board name' field. At the bottom of the form, there's a 'Project template' section with a note: 'Save yourself time with a pre-configured project board template', a 'Template: None' dropdown, and a large green 'Create project' button, which is also highlighted with a red oval. A red annotation on the right side of the form reads 'Opciones de automatización' (Automation options).



Issues

Puede entenderse como un post-it en el tablero **Kanban**.

👉 Cualquier problema o tarea que nos encontremos con el proyecto debe tener un Issue.

Utilizan **Markdown**.

Escribir solo lo necesario para que se entienda el problema. Se puede acompañar imágenes o hiperenlaces en la explicación.

Es el eje la gestión de un proyecto. Cuantos más Issues, mayor información acerca del estado del proyecto.

Un Issue puede relacionarse con otras entidades del repositorio de GitHub. Esta interacción potencia enormemente la información.

- Responsables 
- Etiquetas 
- Milestones 



SOLO SE EVALUA LA COMUNICACIÓN QUE ESTÉ EN EL REPOSITORIO DE GITHUB



GitHub



Investigate to what extent HTTPS should be mandatory #1091

[New issue](#)

Estado Issue

Open

RubenVerborgh opened this issue on 9 Dec 2021 · 2 comments

Explicación
del
problema

Comentarios



RubenVerborgh commented
on 9 Dec 2021

Member



...

In some previous testing, I have come across preliminary evidence that some Solid-related functionality only works over HTTPS. In particular, when running Mashlib as the on-server UI, authentication seems to break because the server is not running over HTTPS.

Whereas this is actually a question for the bigger Solid ecosystem, we can test some assumptions on CSS and turn them into recommendations.

If the answer is that some functionality only works over HTTPS, then we might want to make CSS start over HTTPS out of the box (e.g., by auto-generating `localhost` certificates etc.).



RubenVerborgh added the `task` label on 9 Dec 2021

Assignees

RubenVerborgh

Asignado a

Labels

task

Etiquetas

Projects

None yet

Tablero de Proyecto

Milestone

No milestone

Milestones

Linked pull requests

Successfully merging a pull request may close this issue.

None yet

Pull request
linkado

Log del Issue

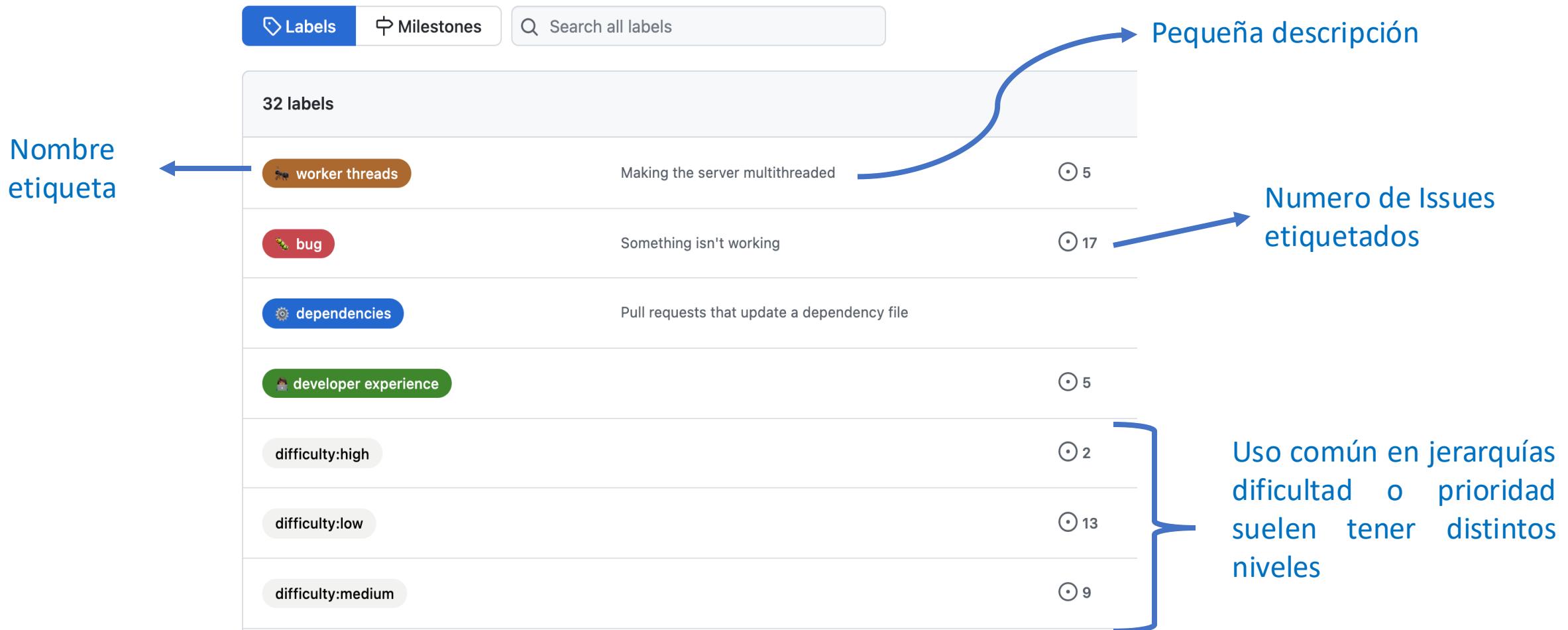
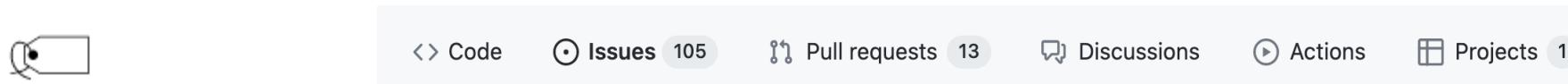


ETIQUETAS



- Sirven para catalogar los issues.
- GitHub provee unas etiquetas genéricas.
- Podéis crear las etiquetas que queráis.(backend, frontend, bbdd, jerarquías)
- Cuanto más organizado mejor.
- Cuidado con no pasarse. Si se crean demasiadas pierden su función que es categorizar.
- GitHub permite personalizar el color y utilizar emojis para dar estilo y ser de ayuda visual .

GitHub



Uso de FAQ de arquisoft para preguntas

<https://github.com/Arquisoft/faq/issues>

Compartir preguntas del curso

- Se permite añadir cualquier issue que describa alguna pregunta relacionada con el curso en inglés o español
 - Cualquiera puede contribuir respondiendo la pregunta o añadiendo comentarios
 - Los que contribuyan deben seguir un código de conducta que respete las consideraciones éticas de un curso de la Universidad de Oviedo
- Los profesores podrán borrar cualquier issue o comentario que consideren inapropiado para el curso

Doc adicional

- Diapositivas de Labra sobre [Git](#).
- [Guía rápida](#) de Pablo González
- [introducción a git](#) de Hugo Lebreiro
- [guía](#)
- [learngitbranching.js.org](#)

Doc adicional

- Contenido [Miguel Angel Durán](#).
- Libro de Miguel Angel [Aprendiendo Git](#).

Youtube

- [Explicación conceptos de Git en 15 minutos](#)
- [Curso/tutorial](#) desde 0
- [GitHub vs GitLab](#) Este video explica la importancia actual de estas empresas.

Práctica de este año



Imagen generada con ChatGPT

Enunciado: <https://arquisoft.github.io/course2425.html#labs>

Primer entregable

- Documentación, versión 0.1
- Primera prueba de concepto
- Aplicación desplegada
 - Aplicación web conectada con GameY
 - Aplicación web accediendo a base de datos

