



GitHub

Creación de Proyectos mediante CrowdSourcing

Seminario de Github: Sistema de Control de Versiones (VCS)

<https://github.com/>

Miguel Gea (mgea@ugr.es)
Dpt. Lenguajes y Sistemas Informáticos
Creación Multimedia Interactiva
Facultad de Bellas Artes
Universidad de Granada
<http://utopolis.ugr.es/medialab>
@media4ux



Act. marzo, 2020



¿Qué es un Sistema de Control Versiones (VCS)?

UN VCS sirve para:

- **Gestión de documentos** / ficheros / proyectos a lo largo del tiempo
- Facilita el **control de versiones** (modificaciones a lo largo del tiempo)
- Permite **compartir** y **colaborar** con otros usuarios / desarrolladores
- UN VCS (como GitHub) permite:
 - Crear **copias de seguridad** y restauración
 - **Sincronizar** versiones entre varios desarrolladores
 - **Deshacer** cambios y cuestiones relacionadas con el tiempo
 - Control de **autoría** (mediante una declaración de licencias explícita)
 - **Gestión del proyecto** (comentar, planificar, valorar, etc.)

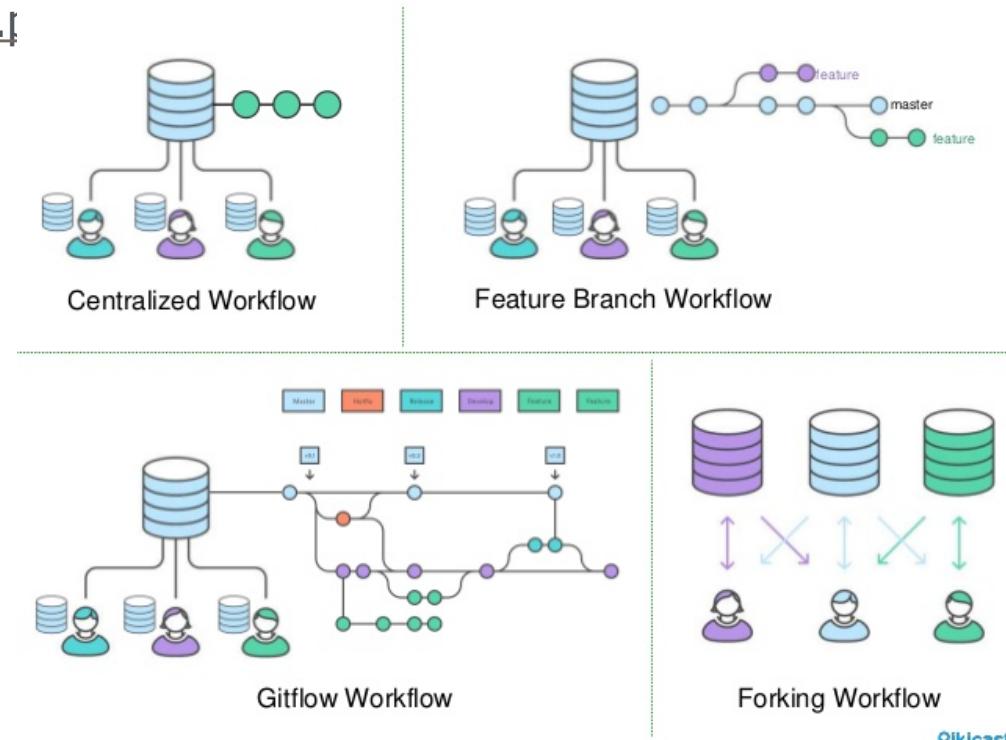
GitHub (<http://github.com>) es uno de los repositorios de software (y contenido) Open Source gratuito más importante (lanzamiento en 2008). Permite organizar el trabajo de versiones y en equipo. Funcionamiento distribuido (un repositorio publico con un proyecto y ramificaciones/branches) y muchas copias privadas .



GitHub para otros usos

El uso principal de Github es como **repositorio distribuido de gestión de proyectos** de software, sin embargo, su estructura es adecuada para otros usos como **la escritura creativa bajo la filosofía CrowdSourcing**. (<https://www.goodrebels.com/why-crowdsourcing-should-be-part-of-your-creative-strategy/>)

- Permite escritura colaborativa y trabajo en equipos con varias alternativas
- Hay incluso herramientas adaptadas de Github específicas para escritores
<https://www.jamstack.org/>



Ref: <https://medium.com/new-media-art-science/open-source-literature-in-the-wild-b2ab54aef69>



GitHub Tareas Básicas

BASICO

- Crear cuenta Github
- **Crear repository**
 - Añadir ficheros
 - Invitar colaboradores (Settings -> Collaborators)
- Comenzar a gestionar copia del proyecto **branch**
- Realizar cambios y subirlos (push)/notificarlos como **commit**
- Abrir y reunificar proyectos con un **pull request/ merge**

DOCUMENTAL / GESTION PROYECTO

- Añadir documentación básica (README.md, LICENSE)
- Añadir sugerencias / ideas / tareas por hacer (**issues**)
- Hacer una planificación de Task-to-do (**project**) según método Kan-ban
- Documentar (**wiki**)
- Valoración (**insight**)

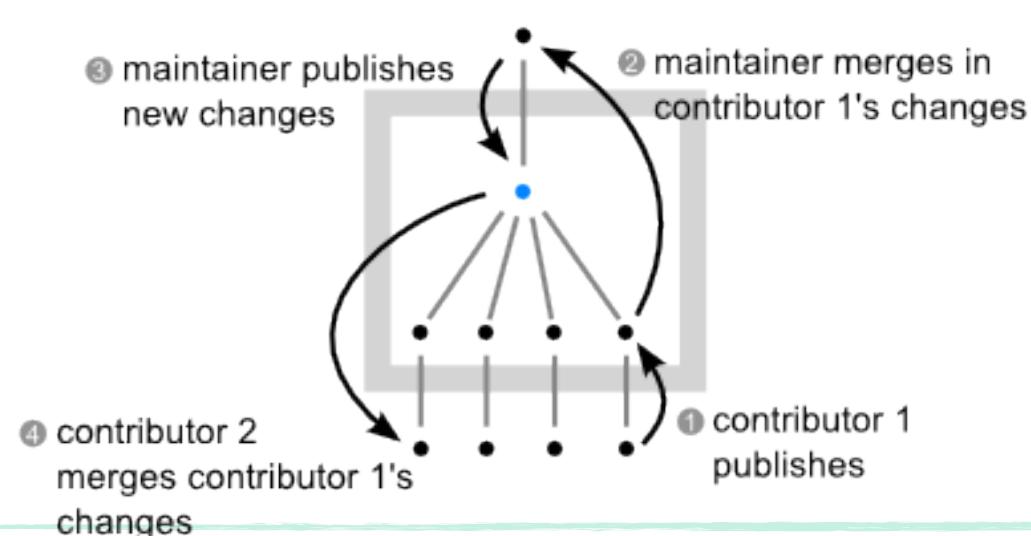
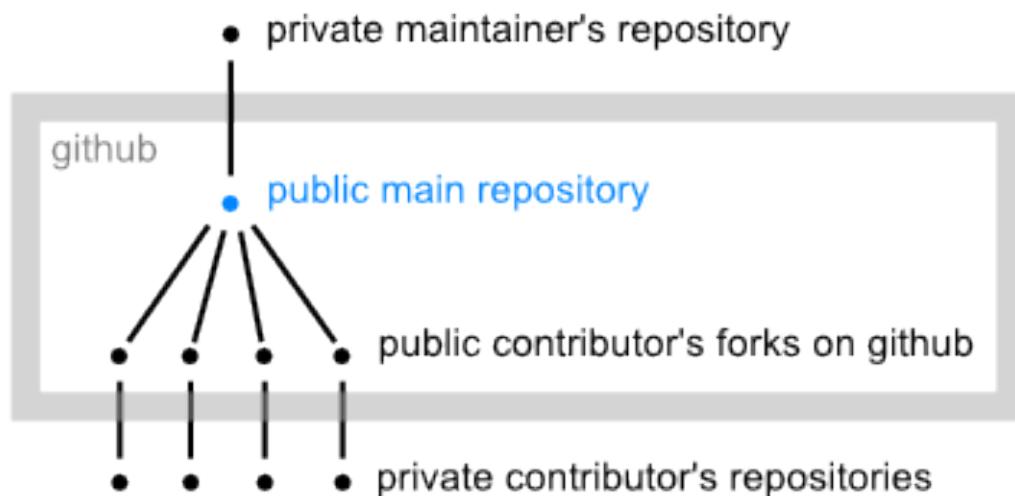
HERRAMIENTAS APOYO

- **Repositorio Web** con capacidad de gestión de contenidos (subir, editar...)
- **Repositorio local** software (Windows/Mac, <https://desktop.github.com/>) + entorno de comandos (Git) para Linux preferentemente.

Organización Distribuida

ORGANIZACIÓN DE CONTENIDO:

- Creación de un **repositorio (remoto)** en la Web (es gratis si es OpenSource) por el administrador/creador
- Creación de **múltiples copias (forks) de contribuciones** (en repositorio en web) de otros desarrolladores
- Estas copias pueden tener una conexión con **repositorio local** (para facilitar los cambios y sincronización) (4)
- Las múltiples instancias (forks) se van combinando (**merge**) (3) mediante solicitudes de publicación (**push**) (1) que notifican al creador (2)





Glosario GitHub

Repositorio (repository). Es la “carpeta” con el proyecto (ficheros+documentación). Cada repositorio conserva un histórico de revisiones. Pueden y pueden ser públicos or privados (Github sólo admite proyectos Open Source). Cada proyecto dispone de:

- **Code** / Ficheros. Fuentes que forman parte del repositorio. Generalmente se añade un README.md (documento que explica repositorio) y la licencia del proyecto (ej. MIT CC).
- Ideas/problemas (**issues**) a resolver. Colección de recomendaciones para el proyecto. Pueden generar un debate / comentarios, y etiquetarse (resueltas...)
- Colaboradores (**collaborators**) Lista de múltiples colaboradores con capacidad de leer/escribir en proyecto (deben ser invitar a colaborar desde “Settings”)
- Proyecto (**project**) Espacio donde organizar hitos bajo un modelo de tarjetas tipo Kansan (por hacer / en progreso / hecho)
- Espacio documentación (**wiki**)
- Indicadores (**insight**) de progreso (likes, stars, contribuciones, etc.)

The screenshot shows a GitHub repository page for 'mgea / media4ux'. The top navigation bar includes links for 'This repository', 'Search', 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. Below the header, there are sections for 'Code' (2 commits), 'Issues' (0), 'Pull requests' (0), 'Projects' (0), 'Wiki', 'Settings', and 'Insights'. The main content area displays the repository's description as 'New media & digital storytelng repository' and its license as 'MIT'. It lists three files: 'LICENSE', 'README.md', and another 'README.md'. At the bottom right, there is a link to 'https://help.github.com/articles/github-glossary/'.



Glosario GitHub

Crear una ramificación (**fork**): Es una ramificación online en Github de la versión del proyecto que se almacenan en el repositorio, pero no afectan al principal. Se pueden crear y mantener sub-versiones (**branch**) donde la principal se llama [master]

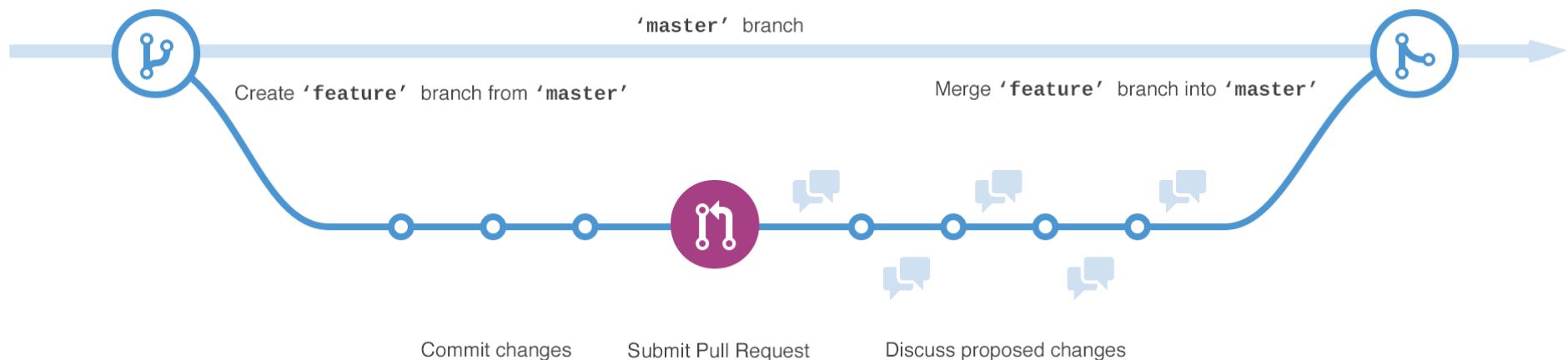
Clone. Crea una copia local (en tu ordenador) del repositorio, en lugar de en tu GitHubID, y también pueden estar sincronización con la original [master] subiéndolos con un “remote”.

The screenshot shows a GitHub repository page for 'mgea / media4ux'. At the top right, there are 'Watch' (0), 'Star' (0), and 'Fork' (0) buttons. A large orange callout bubble labeled 'Fork' points to the 'Fork' button. Below the header, there are tabs for 'Code', 'Issues (0)', 'Pull requests (0)', 'Projects (0)', 'Wiki', 'Settings', and 'Insights'. The 'Code' tab is selected. The repository description is 'New media & digital storytelling repository'. There are buttons for 'Edit', 'Add topics', '2 commits', '1 branch', '0 releases', '1 contributor', and 'MIT'. A green 'Clone or download' button is at the bottom right. A dropdown menu titled 'Switch branches/tags' with a search bar 'Find or create a branch...' is open. A callout bubble labeled 'Clone' points to the 'Clone or download' button. The commit history shows an 'Initial commit' made 'a day ago'.



Glosario GitHub

Commit. Es una "revision", un cambio individual en un(os) fichero(s), y quibble a "guardar". Se crea un ID que premise mantener el control de versiones, de cuando se han efectuado los cambios y por quién. Se recomienda incluir mensajes con las descripciones de los cambios efectuados.



En caso de consolidar cambios con original

Pull Request (PR) es una forma de indicar al propietario que se hacen modificaciones al código en la una de las sub-versiones (fork/branch). Esto permite su valoración para acordar incluirlo al proyecto [master] si se unen (merge)

<https://help.github.com/articles/github-glossary/>

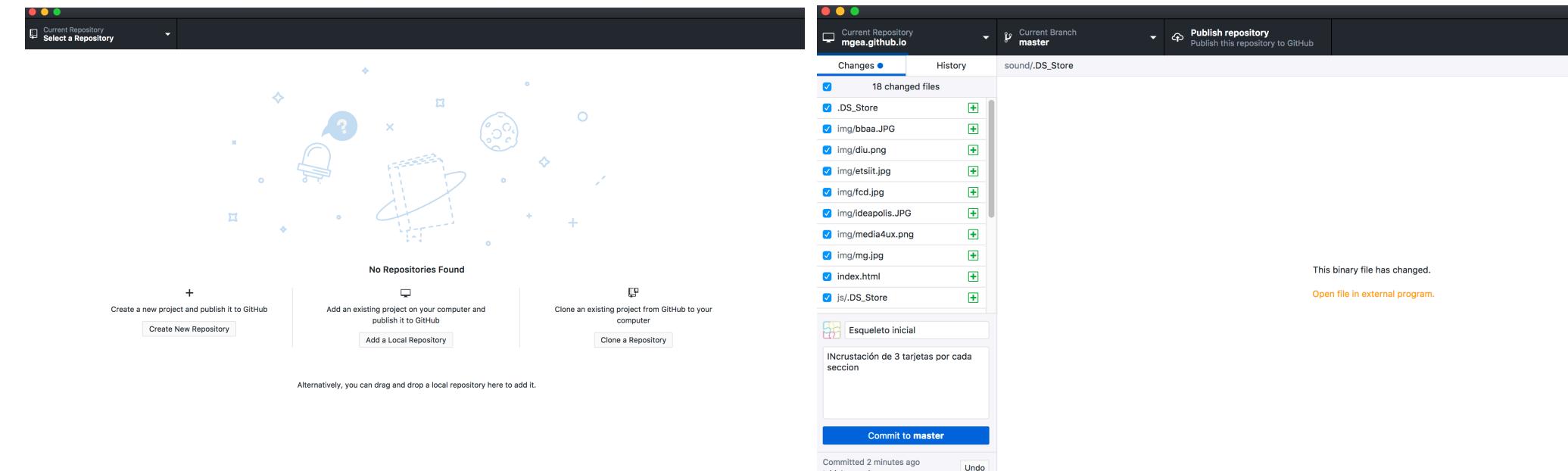


GitHub Desktop (programa local)

El repositorio remoto puede estar “sincronizado” con documentos locales. Se puede utilizar para ello un programa GitHub (PC/Mac/Linux) que te permite controlar las versiones locales.

- Dentro de una carpeta en repositorio denominada **/docs**
- Se pueden sincronizar (documentos locales) con GitHub (commit to master)

(para el gestionar proyecto sólo usaremos el entorno remoto)





Github pages (html)

Se pueden crear **páginas en formato HTML** con algunas limitaciones (*).

Modos de creación:

- Creando un repositorio denominado **username.github.io** donde 'username' es el nombre del usuario de Github.
- Crear una versión (branch) del repositorio denominada **gh-pages**
- CAMBIAR EN PESTAÑA: SETTINGS

The screenshot shows the GitHub repository creation interface. The 'Repository name' field is filled with 'ipsum.github.io'. Below it, a note says 'Great repository names are short and memorable. Need inspiration? How about [refactored](#)'. The 'Description (optional)' field contains the placeholder 'All the lorems you can shake a dolor sit amet at.' At the bottom, there are two options: 'Public' (selected) and 'Private', with descriptions: 'Anyone can see this repository. You choose who can commit.' and 'Only you and your team can see and commit to this repository.'

(*) Limitación de tamaño de HTML < 1GB / ancho banda de visitas de 100GB/mes...

<https://help.github.com/articles/what-is-github-pages/>

Referencia: <https://pages.github.com>

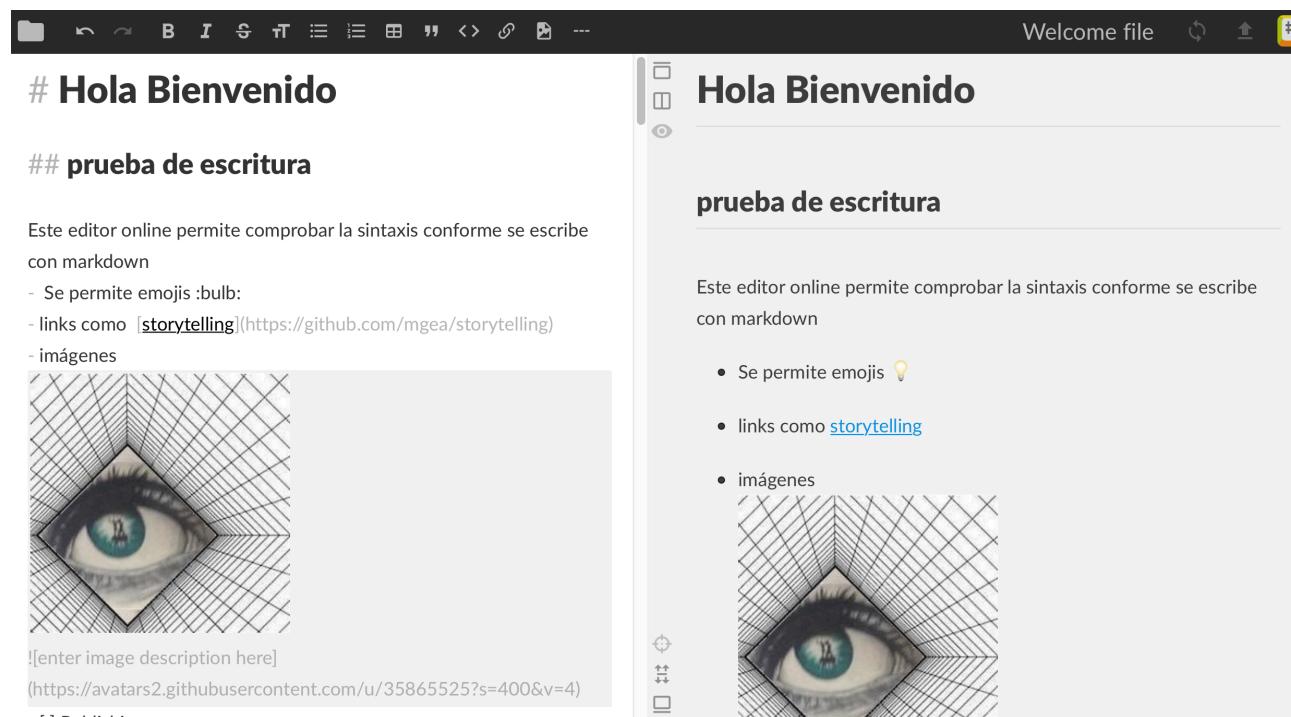


Documentos MD (markdown)



Además de HTML, el repositorio puede contener documentos con información (p.e. `readme.md`) que pueden escribirse con un lenguaje ligero y enriquecido (denominado markdown) que permite crear documentos con una apariencia similar a HTML, pero con una sintaxis más sencilla (y limitada).

- Los documentos tienen **extensión .md**
- Existen editores online para creación de documentos MarkDown correctos. In-browser Markdown editor: <https://stackedit.io/>





Editando MarkDown (desde Github)

Se pueden editar directamente los ficheros **.md (markdown)** en github. Se pueden previsualizar para ver los cambios producidos. El lenguaje de etiquetas es similar (simplificado) de HTML. Directamente se pone el nombre (también se puede poner el nombre de la carpeta donde queremos que se incluya)

The screenshot shows a GitHub repository page for 'mgea / storytelling'. A large orange arrow points from the top left towards the 'Edit file' button in the main content area. The content area displays the file 'Plantilla-proyecto.md' with the following Markdown code:

```
1 # Proyecto Hackers
2
3
4 
5 
6 
7 
8 --
9
10 Nombre del Equipo: :blue_heart: :blue_heart:
11
12 Proyecto
13
14 Resumen:
15
16
17 - Contexto:
18
--
```



CheatSheet markdown



<https://guides.github.com/pdfs/markdown-cheatsheet-online.pdf>

<https://markdown.es/sintaxis-markdown/>

<https://www.webpagefx.com/tools/emoji-cheat-sheet/>

HEADERS

```
# This is an <h1> tag
## This is an <h2> tag
##### This is an <h6> tag
```

EMPHASIS

```
*This text will be italic*
_This will also be italic_

**This text will be bold**
__This will also be bold__

*You **can** combine them*
```

BLOCKQUOTES

As Grace Hopper said:

> I've always been more interested
> in the future than in the past.

As Grace Hoppers said:

I've always been more interested
in the future than in the past.

LISTS

Unordered

- * Item 1
- * Item 2
- * Item 2a
- * Item 2b

Ordered

1. Item 1
2. Item 2
3. Item 3
 - * Item 3a
 - * Item 3b

EMOJI

To see a list of every image we support, check out
www.emoji-cheat-sheet.com

GitHub supports emoji!
 :+1: :sparkles: :camel: :tada:
 :rocket: :metal: :octocat:

GitHub supports emoji!



IMAGES

![GitHub Logo](/images/logo.png)

Format: ! [Alt Text] (url)

LINKS

<http://github.com> - automatic!

[GitHub](http://github.com)

TABLES

You can create tables by assembling a list of words and dividing them with hyphens - (for the first row), and then separating each column with a pipe | :

First Header	Second Header
Content cell 1	Content cell 2
Content column 1	Content column 2

First Header	Second Header
Content cell1	Content cell2
Content column1	Content column2

Caracteres especiales. Barra invertida (\) + *, *, \{ , \# , ...



Anexo I

CREAR UN PROYECTO EN GITHUB

1. Crearemos una CUENTA EN GITHUB para tener repositorios de proyectos

2. Crearemos un Primer Proyecto (repositorio)
 - a) Para ello, **partiremos de uno ya existente** de la asignatura
 - b) Lo **clonaremos (mediante un fork)** para mejorar/adaptarlo

2.a)

<https://github.com/mgea/cmi>

The screenshot shows the GitHub repository page for 'mgea/cmi'. At the top, there's a navigation bar with links for Code, Issues (0), Pull requests (0), Actions, Projects (0), Wiki, Security, Insights, and Settings. Below the navigation bar, the repository name 'Plantilla CMI' is displayed along with a 'Manage topics' link and an 'Edit' button. A summary bar shows statistics: 51 commits, 1 branch, 0 packages, 0 releases, 1 environment, 1 contributor, and MIT license. Further down, there are buttons for Branch: master, New pull request, Create new file, Upload files, Find file, and Clone or download. The main area lists three files: LICENSE, README.md, and index.html, each with their respective details like commit history and last update times.

File	Description	Last Commit
LICENSE	Initial commit	2 years ago
README.md	Update README.md	15 hours ago
index.html	Update index.html	10 months ago

2.b)



Anexo I - Crear nuestro PROYECTO

1. Crear cuenta en GitHub. Espacio ilimitado para proyectos (open source)

1a. Crear un nuevo proyecto (repositorio) público

2. Crear copia de un proyecto ya existente (buscar en Github):

<https://github.com/mgea/cmi>

2a. Clonar a modo local (al ordenador) (en modo Zip o en el 'Github Desktop')

2b. Fork (crear ramificación) en mi espacio GitHub (también se puede crear automáticamente añadiendo ficheros!)

Join GitHub
The best way to design, build, and ship software.

Step 1: Create personal account Step 2: Choose your plan Step 3: Tailor your experience

Create your personal account

Username

This will be your username. You can add the name of your organization later.

Email address

We'll occasionally send updates about your account to this inbox. We'll never share your email address with anyone.

Password

Use at least one lowercase letter, one numeral, and seven characters.

By clicking on "Create an account" below, you are agreeing to the [Terms of Service](#) and the [Privacy Policy](#).

[Create an account](#)

(1a)

Owner: hubot / Repository name: hello-world

Description (optional): Just another repository

Public: Anyone can see this repository. You choose who can commit.

Private: You choose who can see and commit to this repository.

Initialize this repository with a README: This will allow you to git clone the repository immediately. Skip this step if you have already run git init locally.

Add .gitignore: None Add a license: None

[Create repository](#)

<https://guides.github.com/activities/hello-world/>
<https://github.com/new>

Buscar: <https://github.com/mgea/cmi>

mgea / PeriodismoMultimedia_Template

Plantilla para la creación de un proyecto Multimedia

7 commits 1 branch 0 packages 0 releases 1 contributor MIT

Branch: master New pull request Create new file Upload files Find file Clone or download

mgea Update README.md Latest commit 670c484 14 hours ago

LICENSE Initial commit 17 hours ago

(2a)

(2b)

Fork your own copy of mgea/PeriodismoMultimedia_Template to your account

Clone with HTTPS Use SSH
Use Git or checkout with SVN using the web URL.
<https://github.com/mgea/storytelling.g>

Open in Desktop Download ZIP



Anexo I - Repositorio en GitHub (editar local)

3. Se crea un repositorio (fork del original) en tu ESPACIO GitHub, y posee dos branch (la original de donde se copió, y el creado con el nuevo contenido)

https://github.com/mi_nombre/cmi

3a) podemos editar (3c) los ficheros (3b) de la ramificación que hemos creado

3: Espacio (proyecto) que soy “propietario”, e indica de dónde se hizo “copia”

The screenshot shows a GitHub repository page for 'mi_nombre/cmi'. A red arrow points from the top-left towards the repository name 'mi_nombre/cmi' in the header. Another red arrow points down to the commit list. A third red arrow points to the edit icon for the 'README.md' file.

mi_nombre/cmi

Plantilla CMI

Manage topics

51 commits 1 branch 0 packages 0 releases 1 environment 1 contributor MIT

Branch: master New pull request Create new file Upload files Find file Clone or download

mgea Update README.md ✓ Latest commit dab060b 15 hours ago

LICENSE Initial commit 2 years ago

README.md Update README.md 15 hours ago

index.html Update index.html 10 months ago

README.md

(3b: lista de fichero)

(3c: lápiz para editar fichero)

cmi.io

(Plantilla de Proyecto)

Seminario: GitHub. <http://utopolis.ugr.es/medialab>



Anexo I - Crear contenidos y recursos

4. En el proyecto debemos EDITAR el fichero **readme.md** (escrito en MarkDown)

El fichero readme.md está en formato Markdown (tener cuidado con la sintaxis)

Se recomienda usar editores online de MarkDown (<https://stackedit.io/>) para la edición del contenido

Se pueden crear nuevos fichero (se recomienda que sean en formato Markdown), para ello hay que escribir el nombre del nuevo fichero incluyendo su extensión .md

The screenshot shows the GitHub interface for editing the README.md file of a repository named 'PeriodismoMultimedia_Template'. The code editor displays a Markdown document with sections like 'Metodología' and 'Etapa 1: Ideación de proyecto'. A red arrow points from the text area up to a 'Commit changes' dialog at the bottom. This dialog contains fields for 'Update README.md', an optional description, and two radio buttons: one for committing directly to the master branch and another for creating a new branch. Both buttons are highlighted with orange circles.

The screenshot shows the GitHub repository page for 'PeriodismoMultimedia_Template'. It displays basic repository statistics: 8 commits, 1 branch, 0 packages, 0 releases, 2 contributors, and the MIT license. A red arrow points from the 'Create new file' button in the top right corner towards the central content area.

The screenshot shows the GitHub interface for creating a new file. A red arrow points from the 'Create new file' button on the previous screen to this one. The dialog shows a single line of text '1' in the editor area. The file name 'nuevo_fichero.md' is highlighted with an orange oval. The URL in the browser bar also shows the new file name.

4a) Editar (en markdown)
Y confirmar (commit)

4b) Nuevo fichero (en markdown)

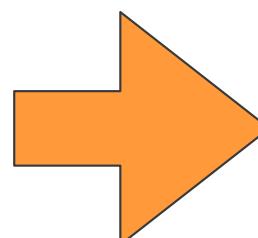


Anexo I - Crear contenidos y recursos

5. Podemos subir ficheros (imágenes, documentos...) arrastrando a la zona de ficheros y haciendo un “commit”

5a) Arrastrar ficheros a este espacio (add files)

A screenshot of a GitHub repository page for 'media4UX / PeriodismoMultimedia_Template'. The page shows a commit history with one commit from 'Initial commit' and two files: 'LICENSE' and 'README.md'. A red arrow points from the top-left towards the file list area.



A screenshot of the GitHub commit dialog for the same repository. It shows a list of files ('fatcat.png', 'girl_globe.png') added via upload. Below is a 'Commit changes' form with fields for adding files via upload and an optional extended description. Two radio button options are shown: 'Commit directly to the master branch.' and 'Create a new branch for this commit and start a pull request.' A green 'Commit changes' button is highlighted with a red oval and a red arrow pointing up to it.

5b) Commit (confirmar cambio)
Se pueden añadir comentarios y anotaciones



Referencias

- P. Bell, B. Beer (2015) Introducing GitHub A Non-Technical Guide. O'Reilly Media
- I. Martínez Ortiz. Taller de Git y GitHub desde cero. UCM. <https://www.lawebdelprogramador.com/pdf/6885-Taller-de-Git-y-GitHub-desde-cero.html>
- Github tutorial. <https://guides.github.com/activities/hello-world/>
- Github glosario <https://help.github.com/articles/github-glossary/>
- Xataka. <https://www.xataka.com/makers/como-y-por-que-se-esta-usando-github-mas-allá-del-software>
- Mastering Markdown. <https://guides.github.com/features/mastering-markdown/.>
- MarkDown Cheatsheet. <https://guides.github.com/pdfs/markdown-cheatsheet-online.pdf>
- MarkDown Sintaxis. <https://markdown.es/sintaxis-markdown/>
- MarkDown Emoji. <https://www.webpagefx.com/tools/emoji-cheat-sheet/>

- Editor Online: In-browser Markdown editor: <https://stackedit.io/>
- **Typora:** Programa OpenSource para Mac/Windows de creación de fichero Markdown. <https://typora.io/>