

Com publicar a *Github Pages*?

Gran part dels repositoris que es publiquen a [Github](#) són públics - accessibles i oberts - per a tothom. Això és una excel·lent notícia per a tothom que crea codi: tothom pot aprendre de tothom. Aquesta seria la gràcia, de fet. Però abans de començar: sabeu què és Git?

Abans de començar: aclarim conceptes

Git és un programari de **sistema de control de versions** que s'utilitza habitualment per al desenvolupament de programari. Es tracta d'un sistema de codi obert i gratuït que s'utilitza tant per crear codi obert com comercial, amb avantatges clars per les persones implicades en la realització de codi.

Git permet als desenvolupadors veure tota la cronologia dels seus canvis, decisions i progressió de qualsevol projecte en un sol lloc. Amb una eina com aquesta, tots els desenvolupadors implicats en un projecte poden treballar de manera asíncrona i, si cal, en diferents zones horàries.

Un programa de control de versions bàsicament ens permet saber coses com: quins canvis es van fer? Qui els va fer? Quan? O perquè eren necessaris aquests canvis?

Un **repositori**, o projecte Git, inclou tota la col·lecció de fitxers i carpetes associades a un projecte, juntament amb l'historial de revisions de cada fitxer. Treballar en repositoris manté els projectes de desenvolupament organitzats i protegits. A través de plataformes com **GitHub**, Git també proporciona més oportunitats per a la transparència i col·laboració del projecte.

GitHub és un **servei de hosting** de repositoris Git, que ofereix les funcionalitats de Git més algunes característiques pròpies. A diferència de Git, el qual és estrictament una eina de línia d'ordres, GitHub proporciona una **interfície gràfica** basada en web i escriptori així com integració mòbil.

Per a treballar en Git és necessari familiaritzar-se amb conceptes com **branch**, **commit**, **pull**, **push**... A continuació trobareu un petit glossari de termes que us poden servir per familiaritzar-vos amb els principals comandaments de Git, tot i que a la xarxa trobareu un munt de tutorials i manuals per a utilitzar-los amb tot el seu potencial. Per començar a introduir-vos amb el tema, us recomanem aquesta guia, [Git Handbook](#), en què trobareu també tots els **comandaments** necessaris per a treballar amb Git i Github.

PRINCIPALS COMANDAMENTS DE GIT

branch:

Aquest comandament s'utilitza per llistar, crear o esborrar branques.

```
git branch  
git branch -d <nom-branca>
```

init :

Aquest comandament s'utilitza per crear un nou repositori Git.

```
git init
```

add :

Aquest comandament s'utilitza per afegir arxius

```
git add nomarxiu.txt
```

commit :

Aquest comandament fa una instantània dels canvis preparats en aquest moment en el projecte. Les instantànies confirmades es poden considerar versions segures del projecte.

```
git commit -m "Missatge que acompanya el commit"
```

status:

Aquest comandament llista els arxius que s'han preparat, els que no i els arxius en els que Git no en farà cap seguiment.

```
git status
```

push :

Comandament dels més bàsics. Envia els canvis fets d'un repositori local a un de remot.

```
git push origin master
```

remote :

Aquest comandament s'utilitza per a connectar en un repositori remot.

```
git remote
```

pull :

Aquest comandament s'utilitza per a fusionar tots els canvis que s'han fet al repositori local.

```
git pull
```

merge :

Aquest comandament s'utilitza per a fusionar una branca amb una altra branca activa.

```
git merge <nom-branca>
```

fetch :

Aquest comandament permet cercar tots els objectes d'un repositori remot que no són actualment al repositori local.

```
git fetch origin
```

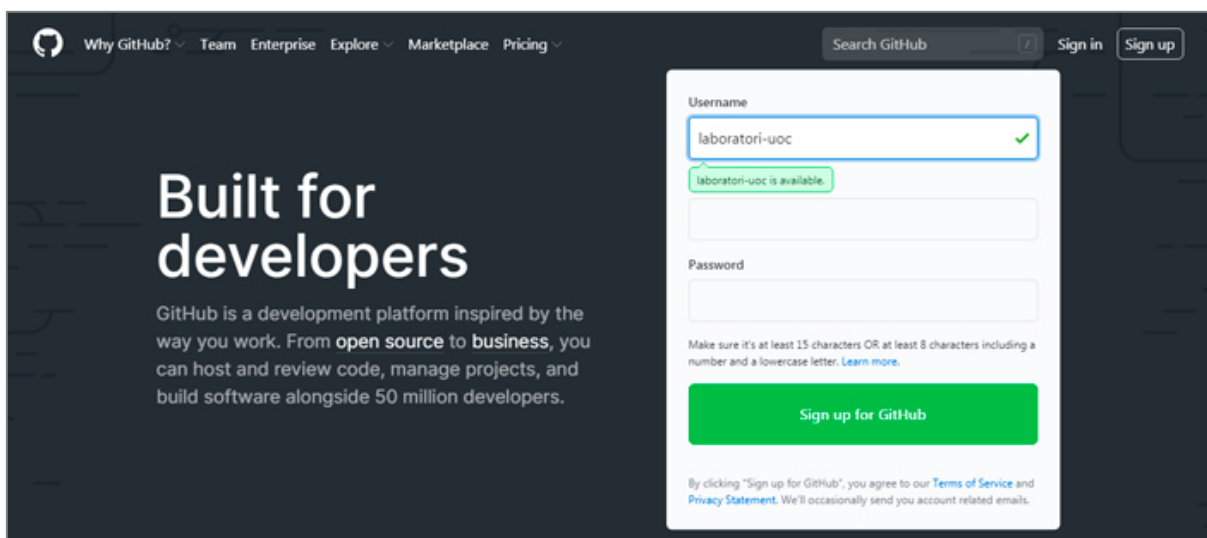
Per aquells que esteu aprenent a crear codi, Github té una funcionalitat molt útil anomenada **Github Pages**, que permet crear i publicar tot el codi d'un web públic allotjant-lo gratuïtament dins del seu domini, amb una estructura d'aquest estil: `elteusuari.github.io`.

En aquest manual veurem com, amb pocs passos i de manera molt senzilla, podem crear i publicar un repositori a Github.

1) Crear un compte a Github

El primer que haureu de fer és registrar-vos a Github.

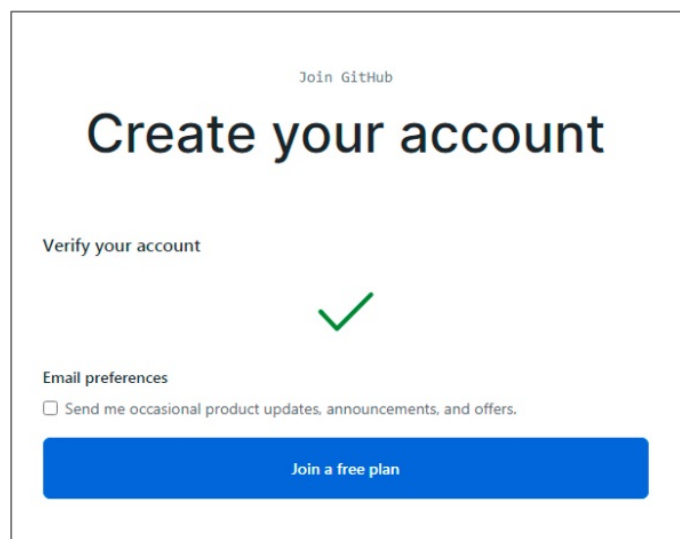
És molt important que, quan escolliu el vostre nom d'usuari, tingueu en ment que aquest serà el nom que apareixerà al vostre domini. Assegureu-vos d'escollir un nom disponible i introduïu el vostre correu electrònic i una contrasenya.



Aleshores haureu de crear un pla gratuït, clicant al botó que posa "Join a free plan". Només haureu de verificar que no sou una màquina, escollint una imatge, i ja podreu seguir amb la configuració del vostre compte.

A continuació, haureu de respondre algunes preguntes sobre què necessiteu, quins coneixements teniu i quins són els vostres llenguatges de preferència.

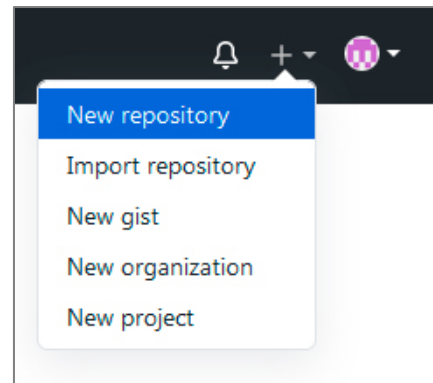
Just després rebreu un correu a la vostra bústia, per tal de verificar que la vostra adreça electrònica és correcta.



2) Crear un nou repositori a Github

Abans de començar a posar codi, el primer que haureu de fer és crear un repositori de codi que és on anirà agrupat tot el vostre codi per aquest nou projecte web. Al marge superior dret, teniu l'enllaç per afegir un nou repositori.

A continuació, escolliu un nom per aquest repositori, el marqueu com a públic i també escolliu que hi hagi un fitxer `readme.md`, que després podreu modificar.




Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner *

Repository name *


 laboratori-uoc ▾

/


web ✓

Great repository names are short and memorable. Need inspiration? How about [miniature-pancake?](#)

Description (optional)

☒  **Public**

Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**


You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☒ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

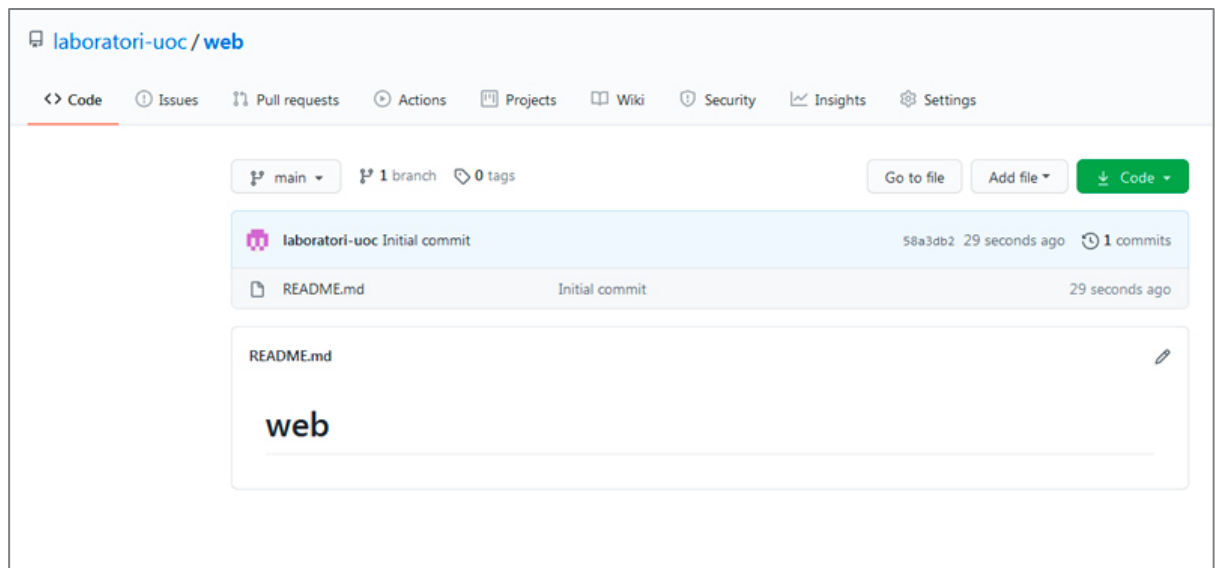
☐ **Add .gitignore**
Choose which files not to track from a list of templates. [Learn more.](#)

☐ **Choose a license**
A license tells others what they can and can't do with your code. [Learn more.](#)

This will set  **main** as the default branch. Change the default name in your [settings](#).

Create repository

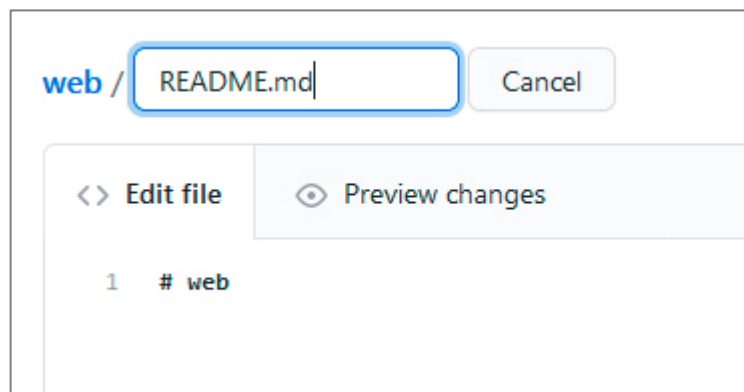
Ja tindreu creat el projecte que hem anomenat “web” - podeu posar-li el nom que vulgueu - i en el qual ara mateix només hi ha un arxiu `README.md`. Al lateral dret d'aquest arxiu, veureu que hi ha una icona amb un llapis.



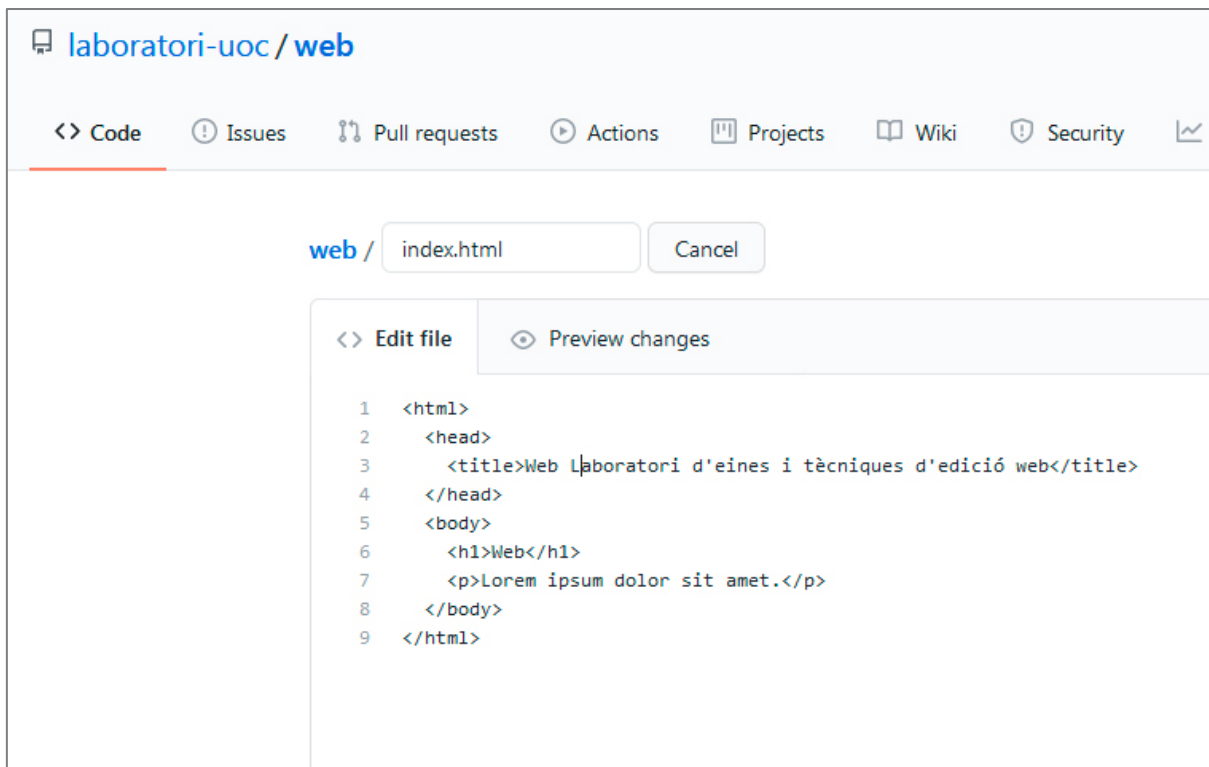
Editar aquest arxiu serà la manera més fàcil de tenir ja feta la primera pàgina del vostre projecte.

Un cop creat aquest repositori, el podreu descarregar i modificar. De fet, quan ja ho tingueu per mà , aquesta serà la manera més còmode de treballar-hi. Però, de moment, i per aquesta ocasió, l'editarem directament al web de *Github Pages*.

A la casella on posa el nom de l'arxiu, canvieu “README .md” per “**index.html**”.



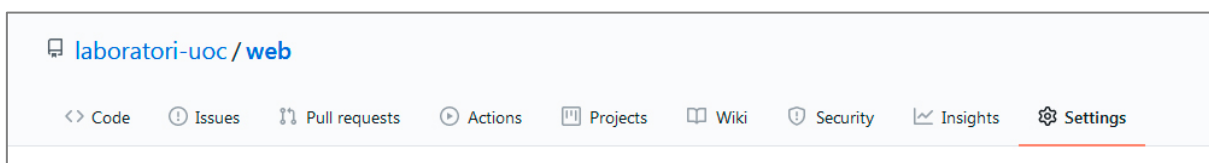
Aleshores, ja podeu editar el text que hi ha dins d'aquest arxiu i hi podeu posar el codi HTML que correspongui a la que serà la vostra primera pàgina.



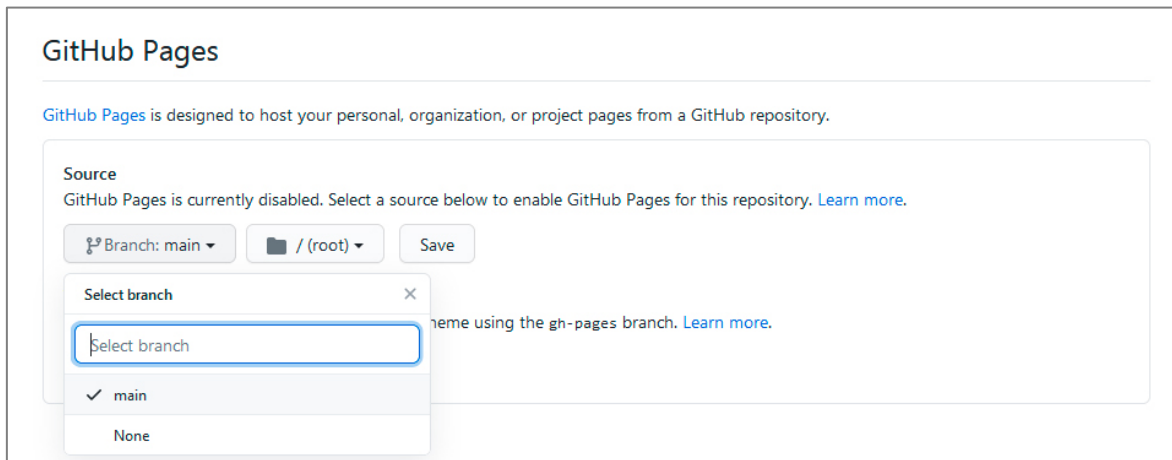
Finalment, amb l'scroll, desplaceu la pàgina cap avall fins a arribar on posa “**Commit changes**”. El canvi de `README.md` a `index.html` junt amb el canvi de contingut de la pàgina quedaran registrats com a canvis realitzats al repositori i sempre es podran consultar en l'historial de canvis.

3) Veure el vostre web publicat

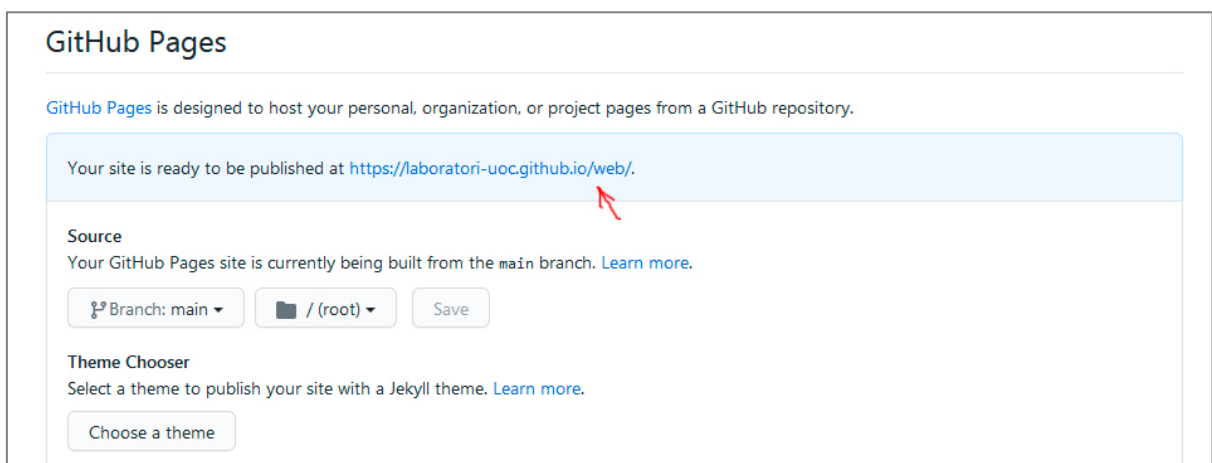
Si accediu al vostre web, publicat a Github i encara no el veieu, haureu d'accedir a la pestanya de “*settings*” i anar a l'apartat de **Github Pages**.



Escolliu el *branch* “**main**” (ja que no n’hem creat cap altre) i feu clic al botó que posa “**save**” per guardar els canvis.



A continuació, ja us sortirà publicada, just al damunt d’aquest apartat, la **URL** on se us ha publicat el vostre projecte web.



Si hi cliqueu, se us obrirà aquest web des del vostre navegador. Sinó també podeu anar directament a un navegador i teclejar l’adreça web del projecte que acabeu de crear.

La primera pàgina que hem creat, **index.html**, a la qual hem posat molt poc codi, ja es veuria així:



A partir d'aquí, ja podeu anar editant aquesta pàgina per posar-li més continguts i estils CSS i imatges relacionades. Podeu anar pujant arxius i treballar amb el projecte, fins a tenir un web del tot navegable.

Més informació i ajuda per a crear pàgines amb Github Pages

- ***Publicar tu web usando Github Pages***
<https://devcode.la/tutoriales/publicar-tu-web-usando-github-pages/>
- ***Creating a Github Pages site***
<https://docs.github.com/en/free-pro-team@latest/github/working-with-github-pages/creating-a-github-pages-site>
- ***Hosting your website on Github***
<https://www.khanacademy.org/computing/computer-programming/html-css/web-development-tools/a/hosting-your-website-on-github>
- ***Hello world***
<https://guides.github.com/activities/hello-world/>
- ***Getting started with Github Pages***
<https://guides.github.com/features/pages/>
- ***Using github pages***
https://developer.mozilla.org/es/docs/Learn/Using_Github_pages
- ***Git Handbook***
<https://guides.github.com/introduction/git-handbook/>