



# *web* *interfaces* *design*



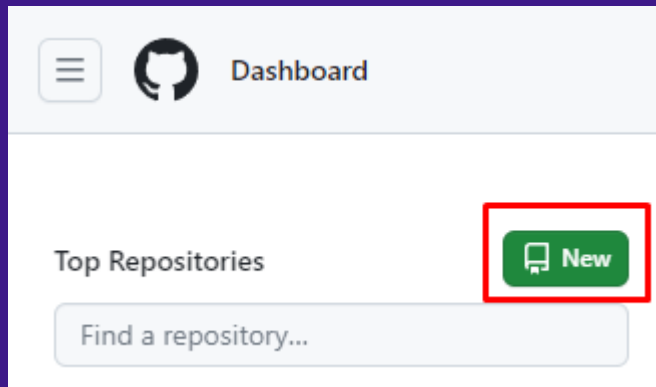
*GitHub*  
*Create page*

ALEX MARTIN URUEÑA

Create a GitHub repository .....	2
Create page .....	4

## Create a GitHub repository

Once we have logged into our private access to GitHub, we will create a new repository.



We enter the desired name, we can put a description about the subject of the project and indicate whether we want it to be public or private.

If public is selected, anyone can view the repository content, but the creator chooses who can modify it.

If private is selected, the author selects who can view and edit the content.

A screenshot of the 'Create a new repository' form on GitHub. The title is 'Create a new repository'. Below it, a subtitle says 'A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)'. A note states 'Required fields are marked with an asterisk (\*)'. The form has two main sections: 'Owner \*' with a dropdown menu showing 'AlexxKai' and 'Repository name \*' with a text input field containing 'example'. Below the repository name field, a green checkmark indicates 'example is available.'. There's a tip: 'Great repository names are short and memorable. Need inspiration? How about [super-duper-fiesta](#) ?'. The 'Description (optional)' section has a text area with the text 'This project is created for the subject "Web interface design".'. At the bottom, there are two radio button options: 'Public' (selected) and 'Private'. The 'Public' option is described as 'Anyone on the internet can see this repository. You choose who can commit.' and the 'Private' option as 'You choose who can see and commit to this repository.'.


Now we have the option to add the README file (where you can see the description we have written above) and choose the license we want our project to have.


Initialize this repository with:

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

Add .gitignore  
.gitignore template: None ▾  
Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license  
License: Apache License 2.0 ▾  
A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

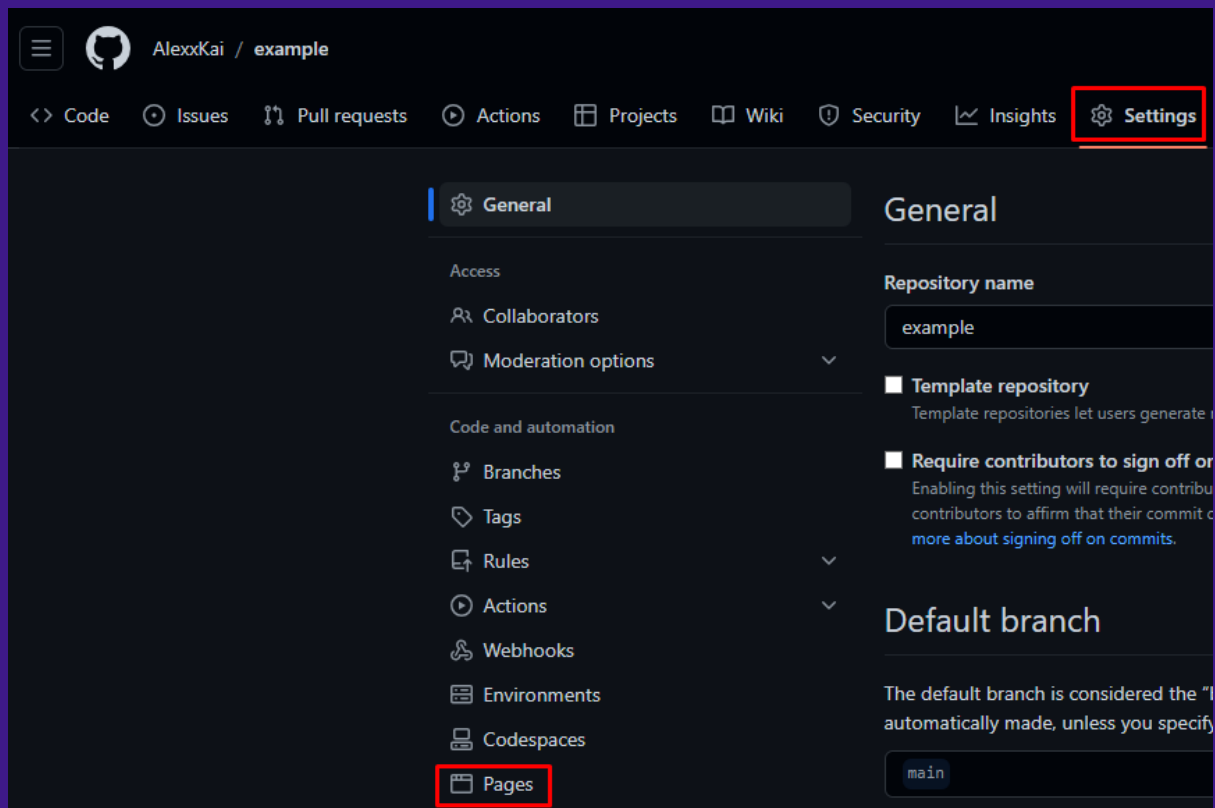
This will set  main as the default branch. Change the default name in your [settings](#).

 You are creating a public repository in your personal account.

Create repository

## Create page

From inside the menu of our repository we will go to the configuration and select the "Pages" option on the left side.



We will indicate from where it has to display the source and select the branch in which it is located.

In the green line we could indicate the custom domain we want to have, although in this case we leave it as default.



### Build and deployment

**Source**

Deploy from a branch ▾

**Branch**

Your GitHub Pages site is currently being built from the `main` branch. [Learn more about configuring the publishing source for your site.](#)

 main ▾  / (root) ▾ Save

[Learn how to add a Jekyll theme to your site.](#)

Your site was last deployed to the `github-pages` environment by the `pages` build and deployment workflow. [Learn more about deploying to GitHub Pages using custom workflows](#)

### Custom domain


**Custom domain**

Custom domains allow you to serve your site from a domain other than `alexkai.github.io`. [Learn more about configuring custom domains.](#)

Save Remove

At the end of these steps and reload the page, we can check at the top of the same section of "Pages" the website of our page.

Your site is live at <https://alexkai.github.io/example/>

Last deployed by  AlexKai yesterday

Visit site ...

The following is the link to the site: <https://alexkai.github.io/example/>