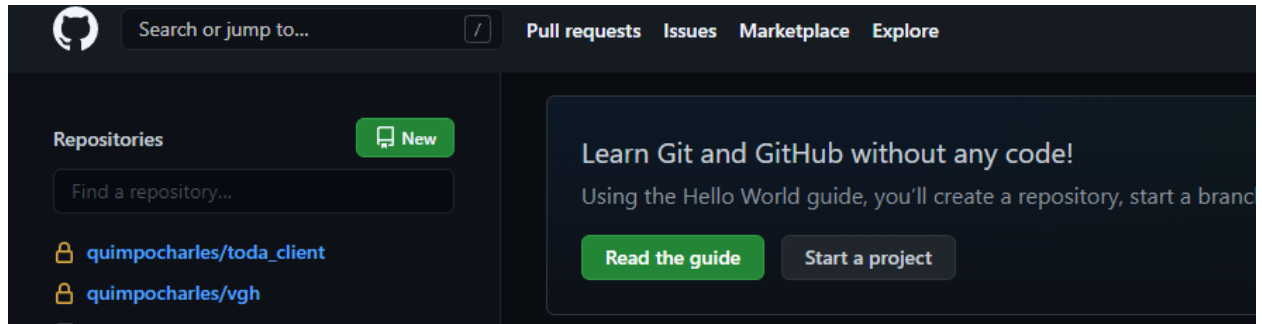
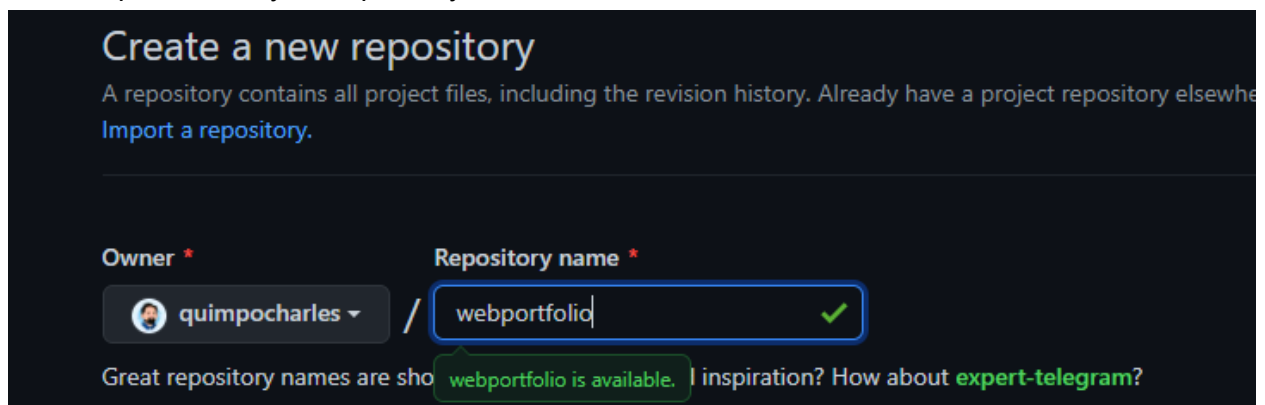


Github deployment

1. Create a Github account [<https://github.com/join>]
2. Login to your Github account [<https://github.com/login>]
3. In your Personal (Dashboard), click on the Start a Project button.



4. Add webportfolio as your repository name.




5. Set the visibility level for the repository to Public

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 *


 quimpocharles ▾

/


webportfolio ✓

Great repository names are short and memorable. Need inspiration? How about **expert-telegram**?

Description (optional)

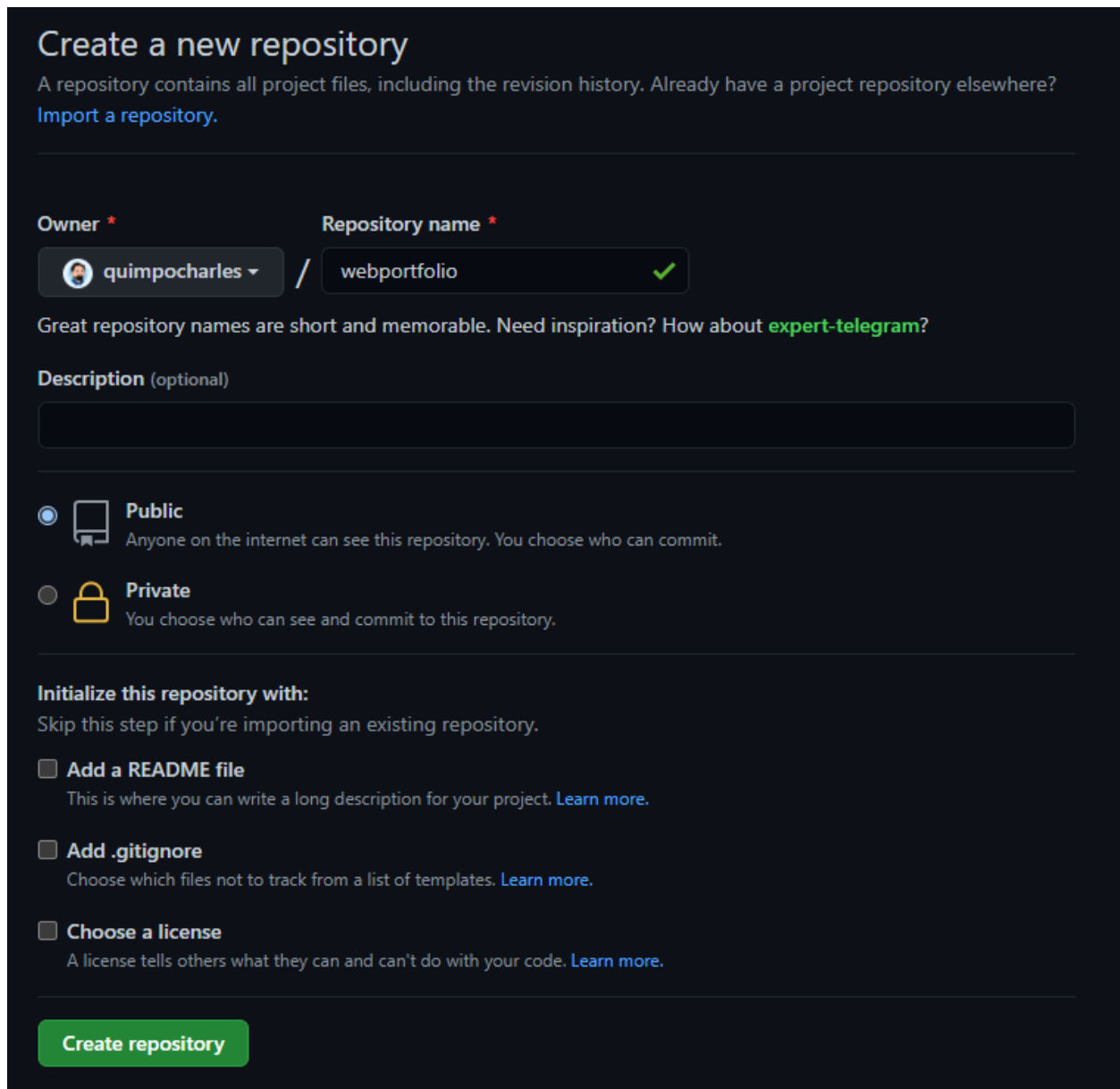
☒  **Public**

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

☐  **Private**

You choose who can see and commit to this repository.

6. Click on Create Repository



The screenshot shows the 'Create a new repository' page on GitHub. At the top, it says 'Create a new repository' and provides a brief explanation of what a repository is. Below this, there are two main input fields: 'Owner' and 'Repository name'. The 'Owner' field is set to 'quimpocharles' and the 'Repository name' field is set to 'webportfolio', with a green checkmark indicating it's valid. A suggestion for 'expert-telegram?' is shown below the name field. There is a 'Description (optional)' text area below the name field. Further down, there are two radio button options: 'Public' (selected) and 'Private'. Below these are three checkboxes for repository initialization: 'Add a README file', 'Add .gitignore', and 'Choose a license'. At the bottom, there is a green 'Create repository' button.

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 *

quimpocharles / webportfolio ✓

Great repository names are short and memorable. Need inspiration? How about **expert-telegram?**

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.

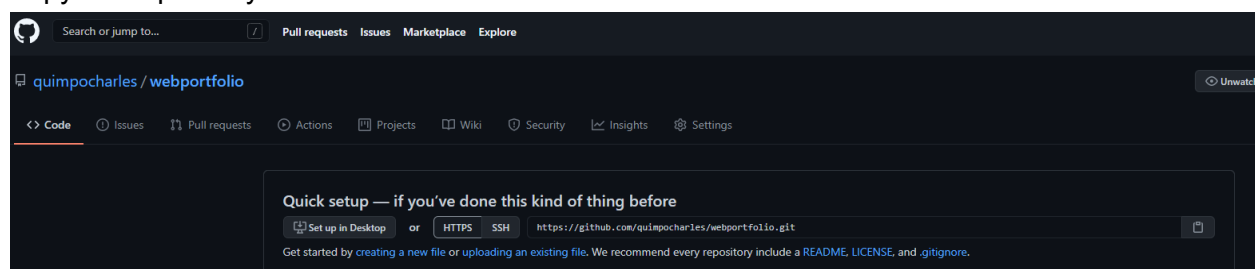
☐ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

☐ **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.](#)

Create repository

7. Copy the repository's SSH URL



Optional: Add an SSH Key for your Github Account

[\[https://docs.github.com/en/github/authenticating-to-github/connecting-to-github-with-ssh/adding-a-new-ssh-key-to-your-github-account\]](https://docs.github.com/en/github/authenticating-to-github/connecting-to-github-with-ssh/adding-a-new-ssh-key-to-your-github-account)

- Open your Terminal/Gitbash and go to your project folder
- Run the command `git remote -v` to check the aliases and URLs associated with your local project.

```
charles@DESKTOP-HNR769E MINGW64 /e/zuitt/csp1 (master)
$ git remote -v
origin  git@gitlab.com:zuitt-coding-bootcamp-curricula/courses/wdc028v1.5/csp1.git (fetch)
origin  git@gitlab.com:zuitt-coding-bootcamp-curricula/courses/wdc028v1.5/csp1.git (push)
```

- Add the Github remote repository by executing the command `git remote add github <paste the url from the Github repository>`. This will add an alias called `github` to your local project which represents the URL to your Github repository.

```
charles@DESKTOP-HNR769E MINGW64 /e/zuitt/csp1 (master)
$ git remote -v
origin  git@gitlab.com:zuitt-coding-bootcamp-curricula/courses/wdc028v1.5/csp1.git (fetch)
origin  git@gitlab.com:zuitt-coding-bootcamp-curricula/courses/wdc028v1.5/csp1.git (push)

charles@DESKTOP-HNR769E MINGW64 /e/zuitt/csp1 (master)
$ git remote add github https://github.com/quimpocharles/webportfolio.git

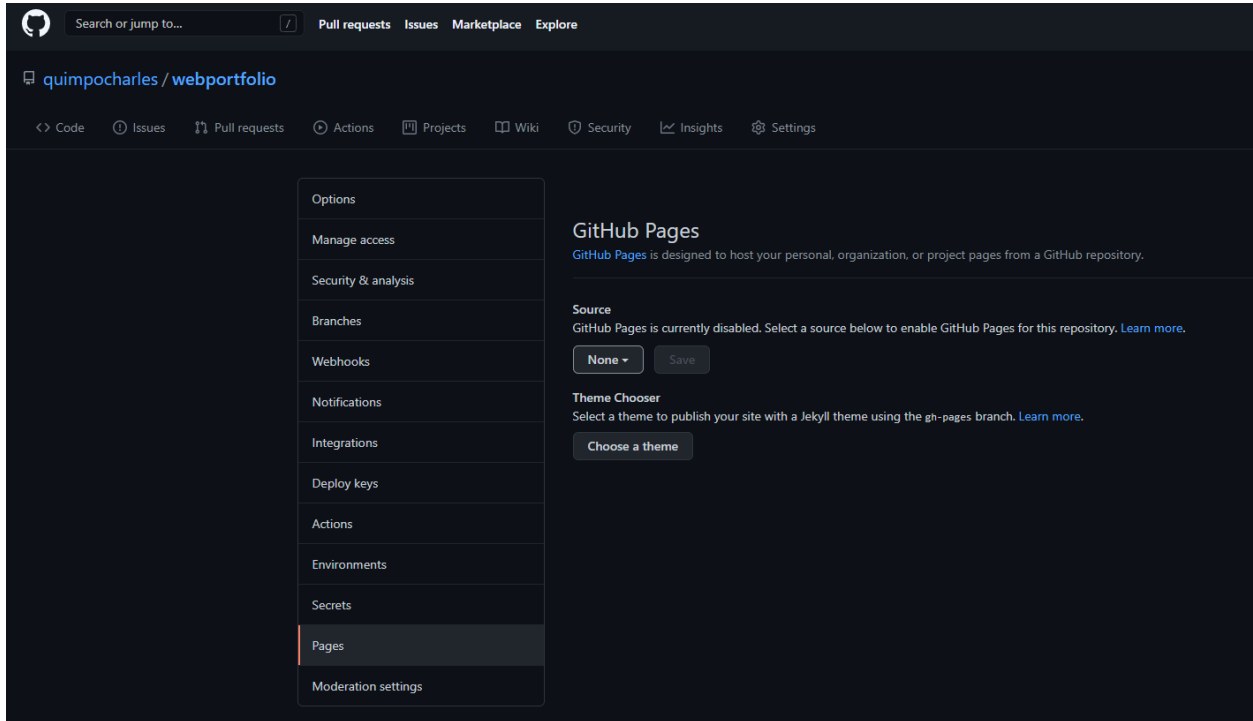
charles@DESKTOP-HNR769E MINGW64 /e/zuitt/csp1 (master)
$ git remote -v
github  https://github.com/quimpocharles/webportfolio.git (fetch)
github  https://github.com/quimpocharles/webportfolio.git (push)
origin  git@gitlab.com:zuitt-coding-bootcamp-curricula/courses/wdc028v1.5/csp1.git (fetch)
origin  git@gitlab.com:zuitt-coding-bootcamp-curricula/courses/wdc028v1.5/csp1.git (push)
```

- Push the project to your github repository.

```
charles@DESKTOP-HNR769E MINGW64 /e/zuitt/csp1 (master)
$ git remote -v
github  https://github.com/quimpocharles/webportfolio.git (fetch)
github  https://github.com/quimpocharles/webportfolio.git (push)
origin  git@gitlab.com:zuitt-coding-bootcamp-curricula/courses/wdc028v1.5/csp1.git (fetch)
origin  git@gitlab.com:zuitt-coding-bootcamp-curricula/courses/wdc028v1.5/csp1.git (push)

charles@DESKTOP-HNR769E MINGW64 /e/zuitt/csp1 (master)
$ git push github master
info: please complete authentication in your browser...
Enumerating objects: 41, done.
Counting objects: 100% (41/41), done.
Delta compression using up to 8 threads
Compressing objects: 100% (40/40), done.
Writing objects: 100% (41/41), 627.74 KiB | 31.39 MiB/s, done.
Total 41 (delta 4), reused 26 (delta 1), pack-reused 0
remote: Resolving deltas: 100% (4/4), done.
To https://github.com/quimpocharles/webportfolio.git
 * [new branch]      master -> master
```

12. Go to the repository's Settings Tab and click on Pages

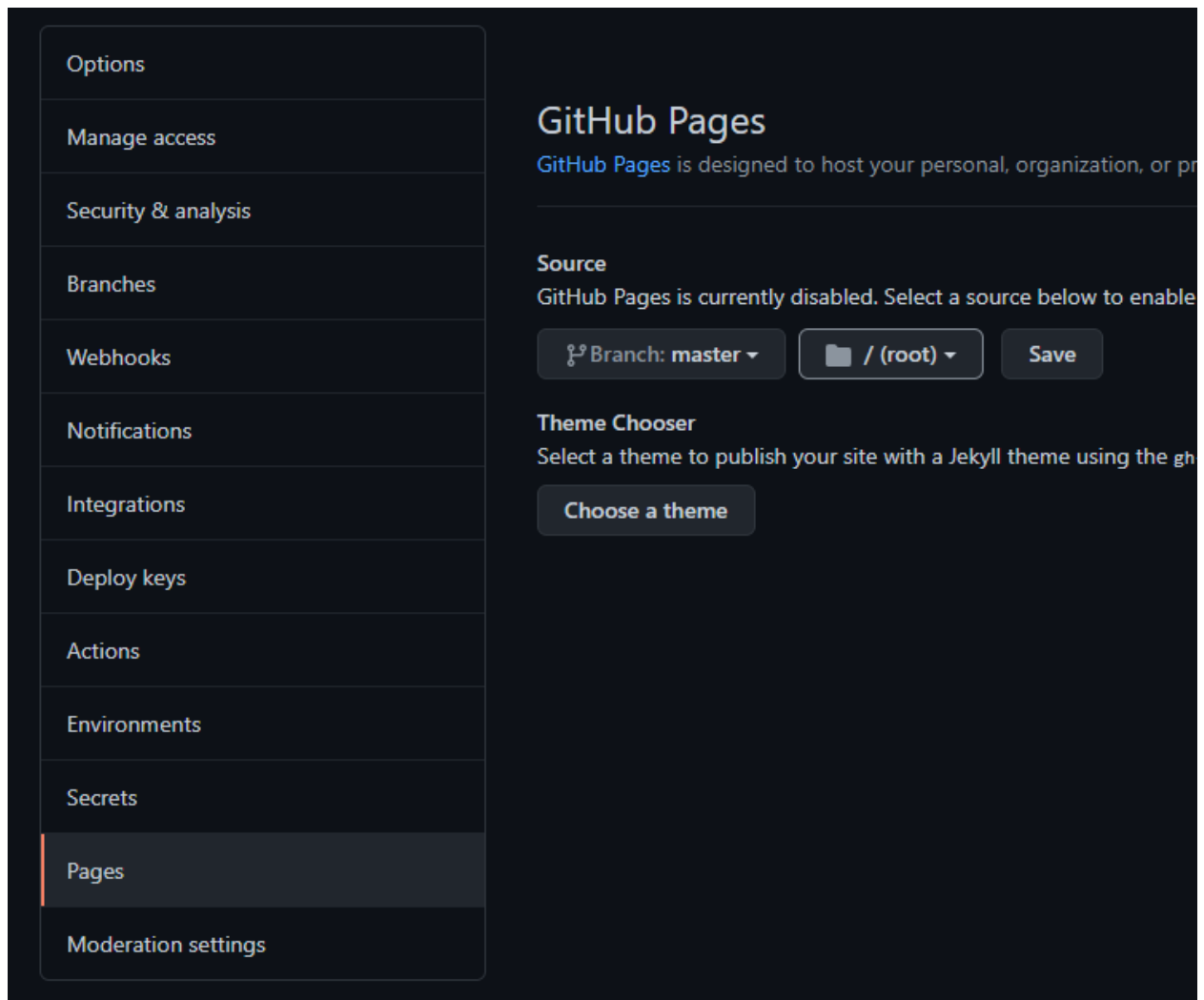


The screenshot shows the GitHub repository settings page for the repository 'quimpocharles/webportfolio'. The left sidebar contains a list of settings categories: Options, Manage access, Security & analysis, Branches, Webhooks, Notifications, Integrations, Deploy keys, Actions, Environments, Secrets, Pages (highlighted with an orange bar), and Moderation settings. The main content area is titled 'GitHub Pages' and includes the following information:

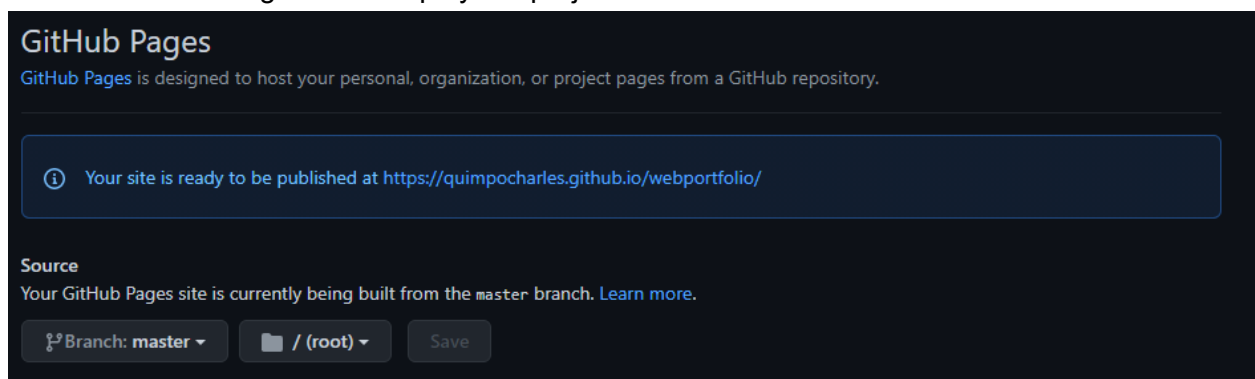
- Source:** GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. [Learn more.](#)
- Theme Chooser:** Select a theme to publish your site with a Jekyll theme using the gh-pages branch. [Learn more.](#)

Below the text, there are two buttons: 'None' (with a dropdown arrow) and 'Save'.

13. Select the master branch as the Source and root as your root folder.



14. Click on Save. Doing this will deploy the project and will create a new link.



15. Sample Output: [<https://quimpocharles.github.io/webportfolio/>]