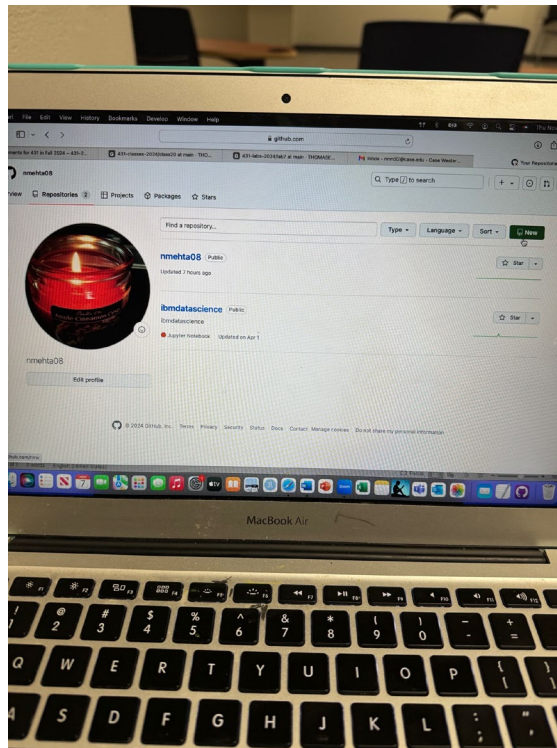


Advice on How to Publish Your Website on GitHub, by Nihit Mehta  
Nihit's site is found at <https://nmehta08.github.io/nmehta08/>

Marvin Schmitt provides a full tutorial and template at <https://www.marvinschmitt.com/blog/website-tutorial-quarto/> and the template is at <https://marvinschmitt.github.io/quarto-website-template> but some people have trouble getting their site working at GitHub after following those instructions.

### Why did I create this document?

It took me nearly three hours of troubleshooting to figure out how to publish a Lab 7 website using the GitHub platform. To save you time and frustration, I've put together this step-by-step guide to help you complete the task without running into the same issues I faced. I hope you find it useful! Happy "Websiting"!



**Task 1** once you have created the GitHub Account, click on the green button to create a new repository.

Required fields are marked with an asterisk (\*).

Owner \* nmehta08 / Repository name \*

Great repository names are short and memorable. Need inspiration? How about [ubiquitous-octo-eureka](#) ?

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:

☐ **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: None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

☐ You are creating a public repository in your personal account.

[Create repository](#)

**Task 2** I used my username as the repository name, and that worked well for me. Feel free to leave the default settings as shown in the photo above. Once you've done this, click the green button in the bottom-right corner to create your repository.

**nmehta082**
Public

[Pin](#)
[Unwatch 1](#)
[Fork 0](#)
[Star 0](#)

**Set up GitHub Copilot**  
 Use GitHub's AI pair programmer to autocomplete suggestions as you code.  
[Get started with GitHub Copilot](#)

**Add collaborators to this repository**  
 Search for people using their GitHub username or email address.  
[Invite collaborators](#)

**Quick setup — if you've done this kind of thing before**  

Set up in Desktop
 or
 

HTTPS
 SSH

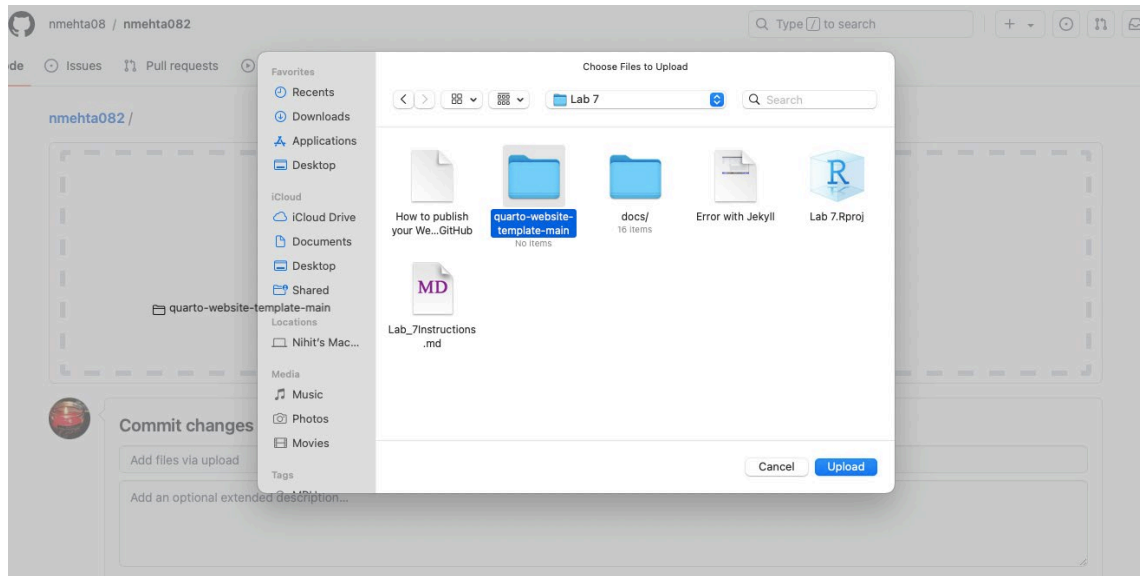
 Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

**...or create a new repository on the command line**  

```

echo "# nmehta082" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/nmehta08/nmehta082.git
      
```

**Task 3** Click on the Upload an existing file, below the Quick Set up.



**Task 4** I used the Marvin Schmitt template, which creates a separate folder, and I uploaded the "docs" folder inside it. If you're using his template, make sure to upload the "docs" folder, not the entire template folder.

My second issue was that GitHub doesn't allow you to upload folders directly—only individual files—if you use the traditional upload method. To get around this, I dragged and dropped the folder into the dashed box on GitHub. Once the upload starts, it will preserve the folder structure. At that point, click the "Cancel" button near the blue upload button in the bottom-right corner to stop the upload.

The screenshot shows the GitHub Pages configuration interface. On the left is a sidebar with navigation links: General, Access (Collaborators, Moderation options), Code and automation (Branches, Tags, Rules, Actions, Webhooks, Environments, Codespaces), Pages (highlighted), Security (Code security, Deploy keys, Secrets and variables), and Integrations (GitHub Apps, Email notifications). The main content area is titled 'GitHub Pages' and includes the following sections:

- General:** States 'Your site is live at <https://nmehta08.github.io/nmehta08/>' and shows it was last deployed by 'nmehta08' 7 hours ago. A 'Visit site' button is present.
- Build and deployment:**
  - Source:** A dropdown menu is set to 'Deploy from a branch'.
  - Branch:** A message states the site is built from the '/docs' folder in the 'main' branch, with a link to 'Learn more about configuring the publishing source for your site.' Below this are dropdowns for 'main' and '/docs', and a 'Save' button.
- Custom domain:** A message states the site was last deployed to the 'github-pages' environment by the 'pages build and deployment' workflow, with a link to 'Learn more about deploying to GitHub Pages using custom workflows'. Below is a text input field for a custom domain, 'Save' and 'Remove' buttons.
- Enforce HTTPS:** A checkbox is checked, with a note: 'Required for your site because you are using the default domain (nmehta08.github.io)'.

**Task 5** Change the source to "Deploy from a Branch," set the branch to "main," and enter "/"docs" for the folder. Then, click "Save." It may take four to five minutes, but once the page refreshes, you'll see an updated page with your website link.