Deploy your first JupyterLite website on GitHub Pages

Hint

If you first want to get familiar with the interface, check out the User Guide.

JupyterLite can easily be deployed on GitHub Pages, using the `jupyter-lite` CLI to add content and extensions.

Note

Deploying to GitHub Pages requires a Github account.

Generate a new repository from the template

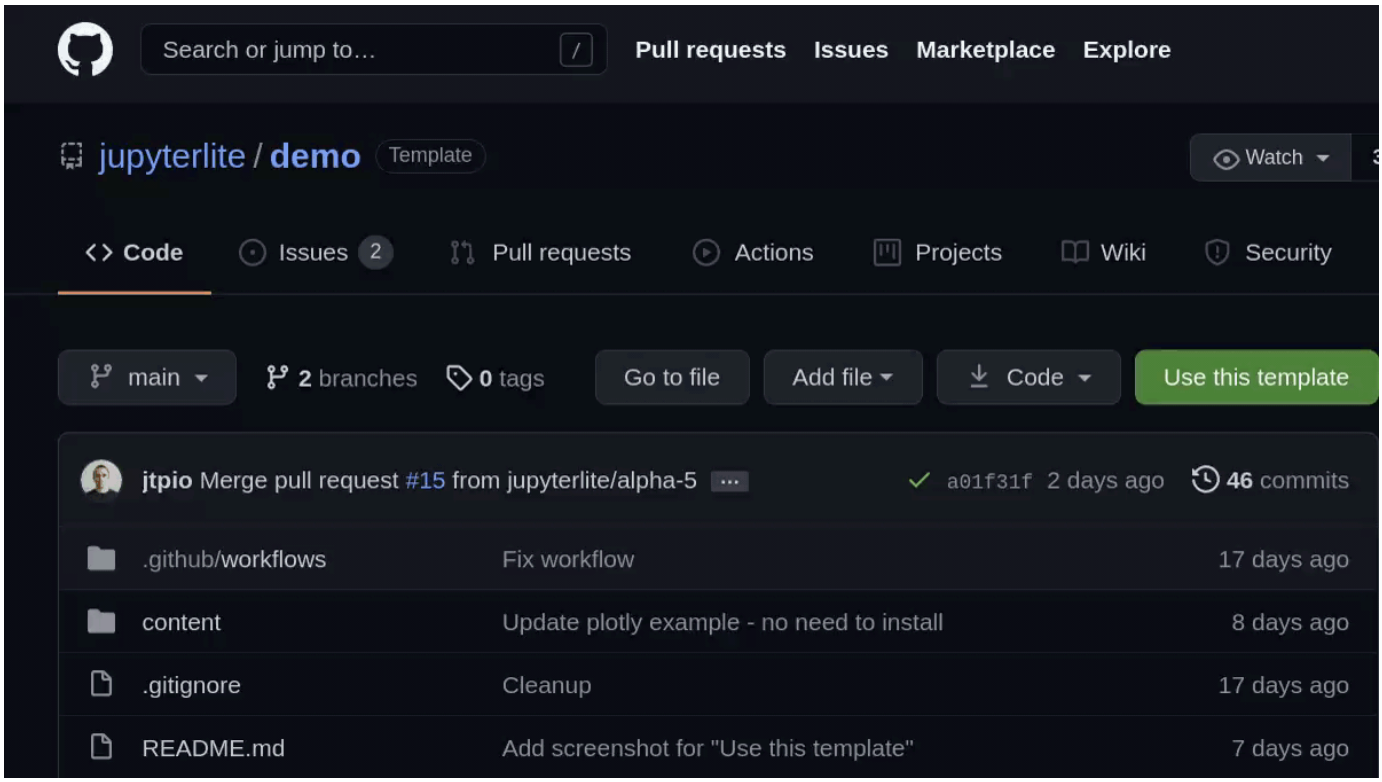
The jupyterlite demo repository is a template to easily:

build a JupyterLite website using prebuilt JupyterLite assets bundling a collection of pre-existing Jupyter Notebooks as part of the distribution

deploy the website to GitHub Pages

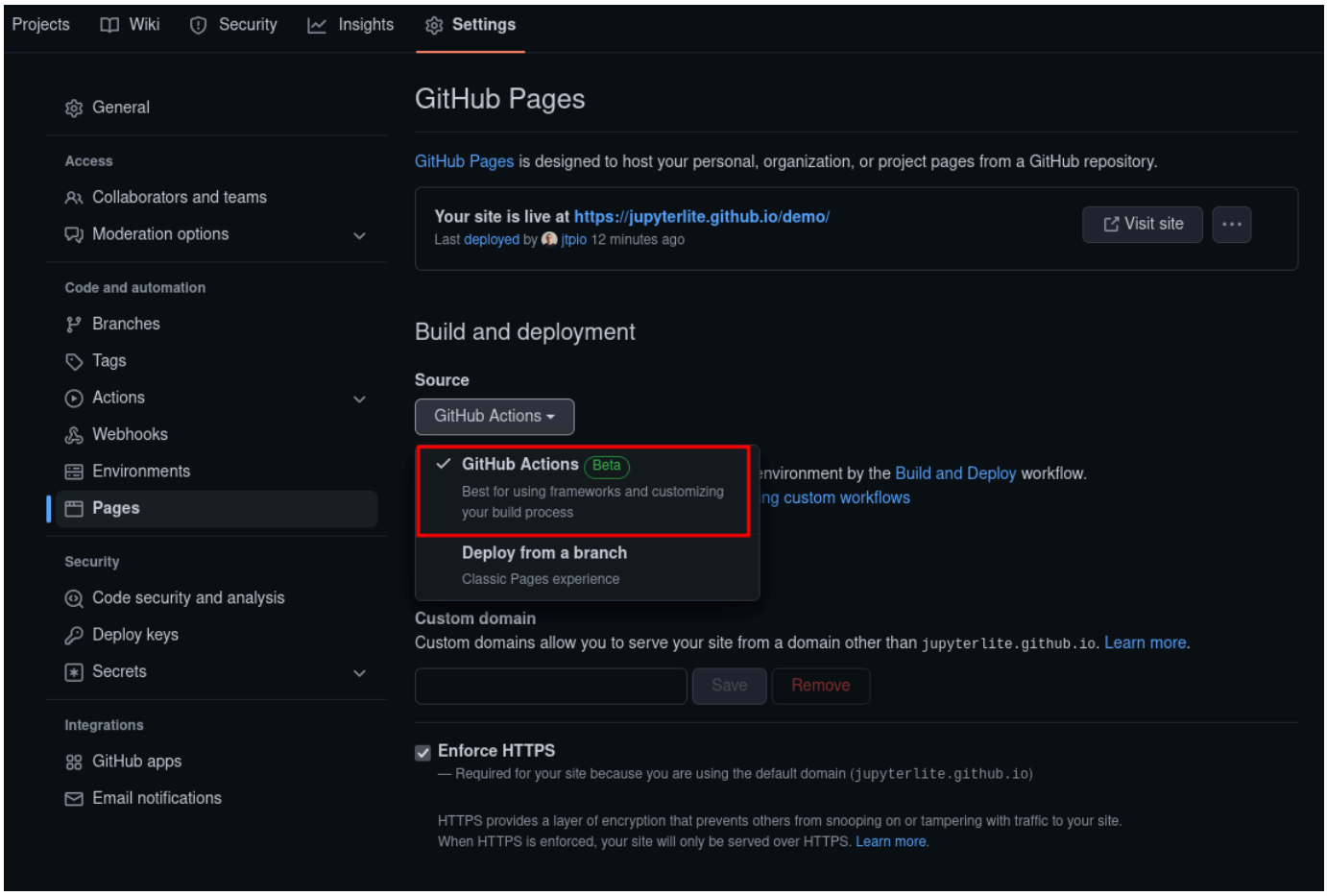
The process is automated using Github Actions.

Click on “Use this template” to generate a repository of your own from this template:



From the *Actions* tab on your repository, ensure that workflows are enabled. When you make a commit to the `main` branch, a Github Action will build your JupyterLite release and deploy it to the repository's Github Pages. By default, the Github Pages site will be found at `YOUR_GITHUB_USERNAME.github.io/YOUR_REPOSITORY-NAME`. *You can also check the URL from the Repository* `Settings` *tab* `Pages` *menu item.*

If the deployment failed, go to "Settings - Actions - General", in the "Workflow permissions" section, check "Read and write permissions". Check that you have Github Pages enabled for your repository: from your repository *Settings* tab, select the *Pages* menu item and ensure that the source is set to `GitHub Actions`:



When you commit a file, an updated website will be built and published on Github Pages.

Note

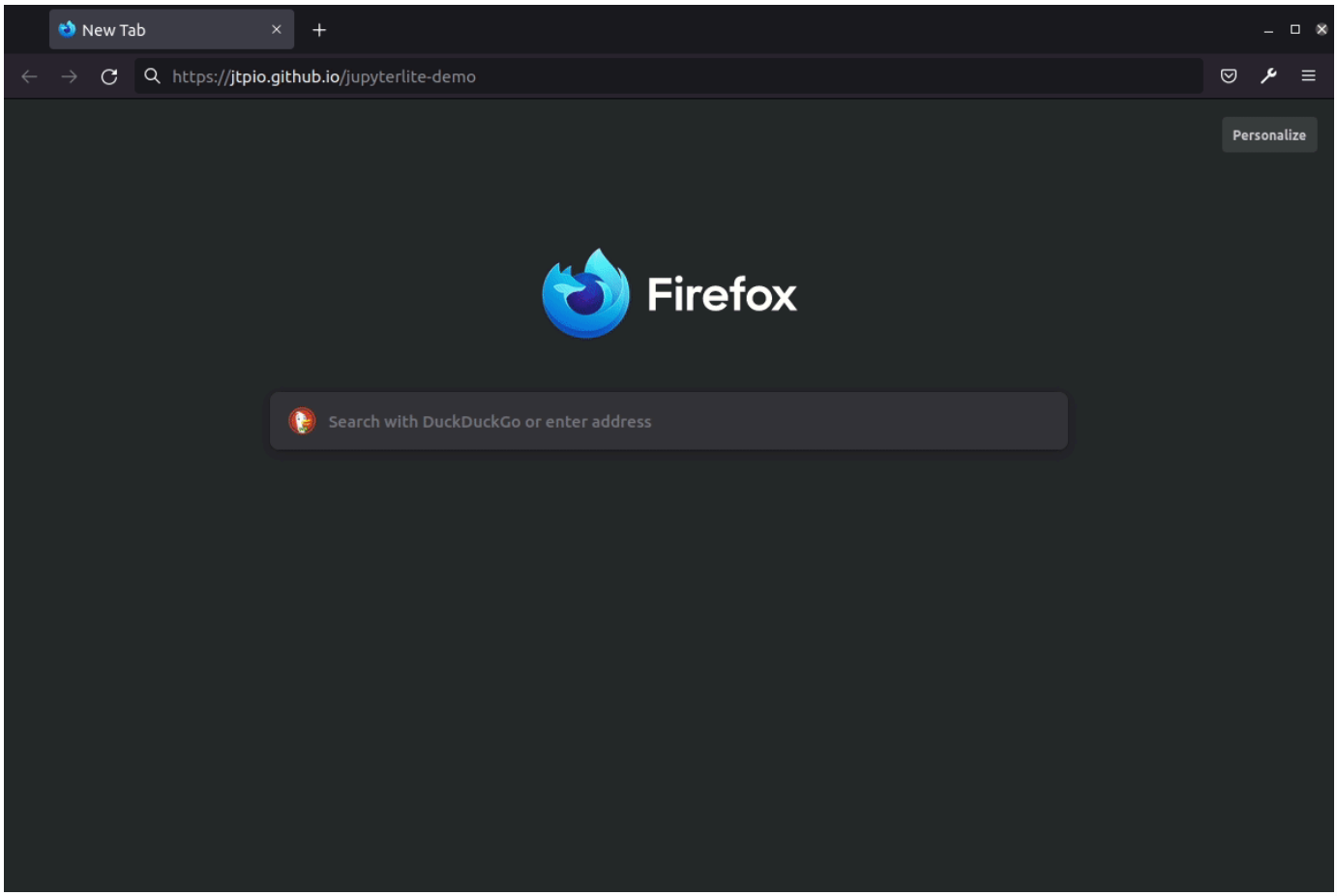
Note that it may take a few minutes for the Github Pages site to be updated. Do a hard refresh on your Github Pages site in your web browser to see the new version of the website.

Note

Alternatively, you can use the JupyterLite demo using xeus-python to publish a deployment on Github pages that uses xeus-python by default and allows to pre-install packages using `emscripten-forge` and `conda-forge`.

Accessing the JupyterLite website

After the build has completed, the site will be available on GitHub Pages. Go to `https://YOUR_GITHUB_USERNAME.github.io/YOUR_REPOSITORY-NAME` to access it:



Note

By default the deployment provided by the `jupyterlite/demo` repo includes a `.nojekyll` file to bypass Jekyll processing on GitHub Pages.

See this blog post for more information.

Deploy a new version of JupyterLite

To change the version of the prebuilt JupyterLite assets, update the `jupyterlite-core` package version in the `requirements.txt` file.

Commit and push the changes. The site will be deployed on the next push to the `main` branch.

Add additional requirements to the deployment

Note

The jupyterlite/demo repository uses a `requirements.txt` file to specify the dependencies. For demo purposes this file may contain extra kernels and extensions you might want to remove from your deployment. If that's the case you can stick to a more minimal `requirements.txt` file such as:

```
# core package for building the JupyterLite website
jupyterlite-core==0.1.0b19
# the Python kernel powered by Pyodide
jupyterlite-pyodide-kernel==0.0.5
```