ARTMIP GitHub + MyST Workflow

A beginner-friendly guide to build a website for your own atmospheric river detection algorithm, using cutting-edge tools that promote effective documentation, reproducibility, and transparency

Jimmy Butler, Fernando Pérez, March 2025

Getting Started

- Log into GitHub (in your browser), go to https://github.com/jbbutler/artmip_template, and click on the "use this template" arrow on the right and pick the "create new repository" option:
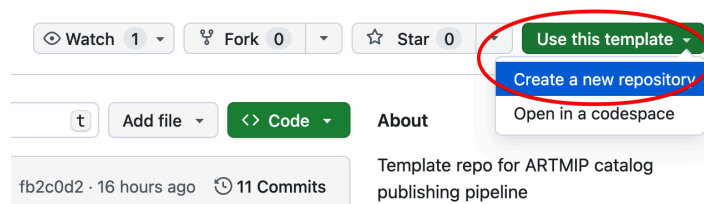
- You will get a screen like this, we suggest you name it `artmip-publish-<IDENTIFIER>` , where you can pick `<IDENTIFIER>` to be any short mnemonic that identifies either you, or your algorithm/catalog, or paper, or team, etc. Examples could be `TEMPEST-2020`, `Viale-etal-2018`, etc. We're simply thinking of a pattern that makes it easy to later automate the integration of many similar repositories into a collection.
- Log into the **CryoCloud Hub** at: https://hub.cryointhecloud.com.
- In your home directory, you will see a subfolder called `shared-public`. From there, navigate to the `fperez/` folder, where you can find a notebook called `GHAUTH.ipynb`. Copy this notebook to your home (top level) folder, and run the cells to connect GitHub with CryoCloud.
  - You will need to run this notebook once each time you log into the Hub if you intend to work with a github repository. It saves you from having to type your github credentials over and over. You can add it to your "Favorites" (right-click on it) so it's always easy to find it:
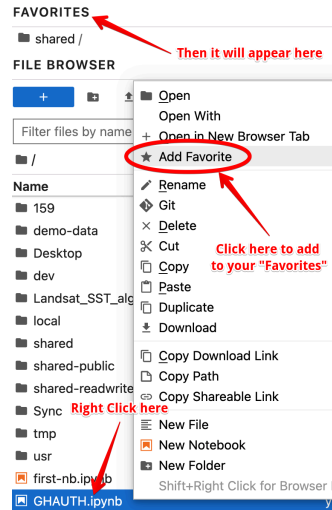
- ○ The notebook will ask you to complete two steps:
  1. Paste the provided 8-digit code into the provided GitHub link, choose your GitHub account with the repo, and authorize access

```
import gh_scoped_creds
%ghscopedcreds
```

```
The code DA6B-7BEF has been copied to your clipboard.
You have 15 minutes to go to https://github.com/login/device and paste it there.
```

  2. A success message will appear, along with a link to allow CryoCloud to push to your repo. Click this link, then click **Configure**, the account for which you would like to install CryoCloud GitHub Access, and then choose either All repositories or Only select repositories (if you choose the latter, choose your new repo in the dropdown)
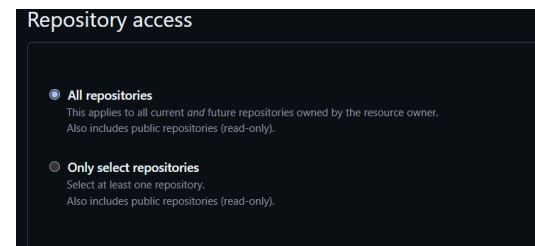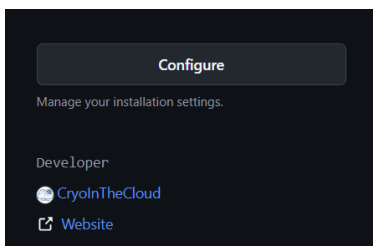
- Open a terminal window in the JupyterLab interface in CryoCloud.
- `git clone` your repository into CryoCloud. You will find the address of your repository here on your github page:
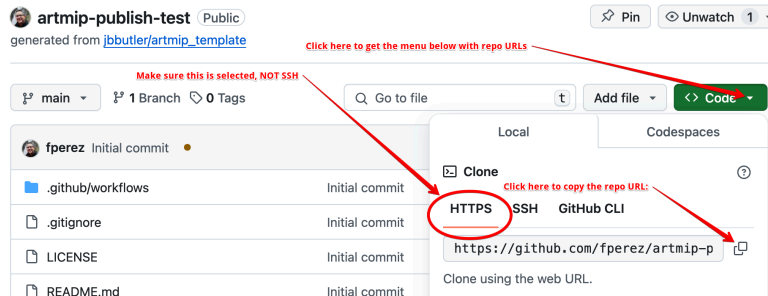
- The repository template already contains a GitHub action to deploy the content to a GitHub pages hosted website. You will notice, however, that there will be an automatic failed deployment to GitHub pages (lower right hand corner of in-browser repository home page). This is because, by default, the repository is not configured to be deployed to GitHub pages. To configure this, go to **Settings**, **Pages**, and under the **Source** dropdown, click **GitHub Actions**.
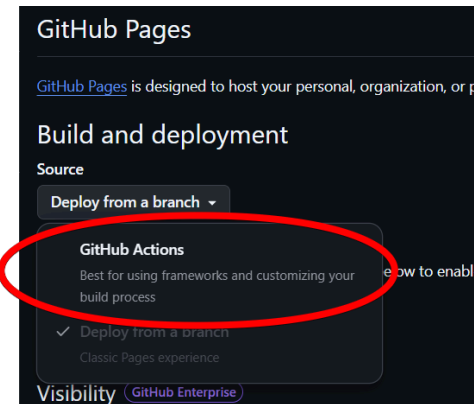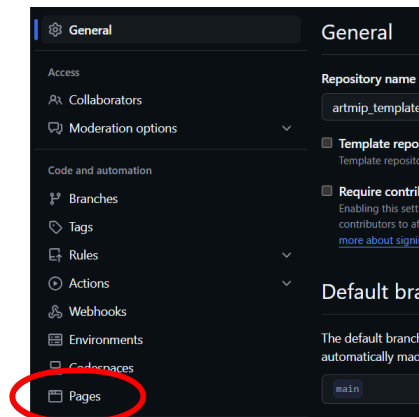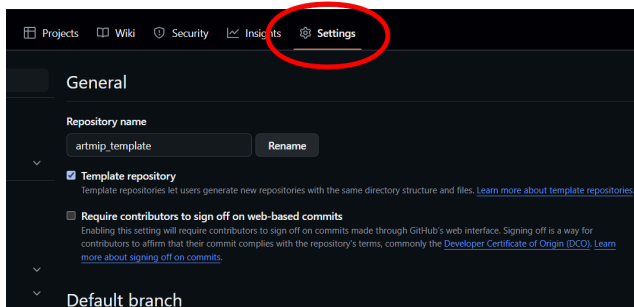

- On your next commit to the repository, your website will automatically attempt to deploy again, and this time it should work!
  - **NOTE:** if this is your first time using GitHub on CryoCloud, you will need to configure git by giving your name and email address. Run the following lines in a Terminal window on CryoCloud (you will only need to do this once for any git-based work you do on CryoCloud)
    1. `git config --global user.name "<your name>"`
    2. `git config --global user.email "<your email>"`
  - To deploy your website, you can do an empty commit from the terminal in CryoCloud
    1. `git commit --allow-empty -m'empty commit to deploy'`
    2. `git push`

- Now, the deployments tab should indicate a successful deployment to GitHub pages! Click on the deployment link, and you should see a link to your website; click that link and you should see the template website!
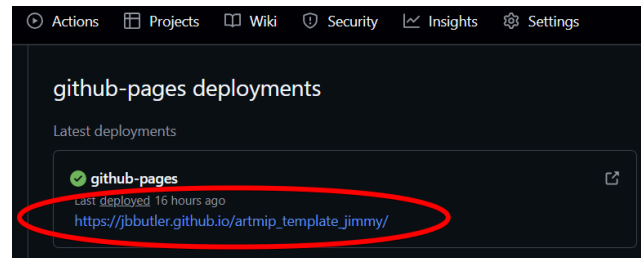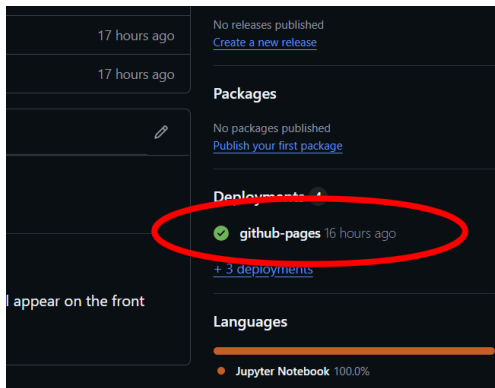

- Each time you push changes to the repo, either through the Desktop or on your copy in CryoCloud, the website will redeploy with the updated content. Refresh the site to see your changes! It's good practice to include the link to the site in the About section of your repo. To do so, click the gear icon on your home page, paste the URL in the Website field, and save.
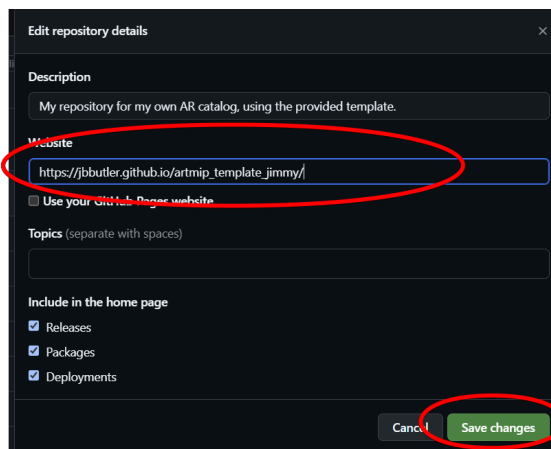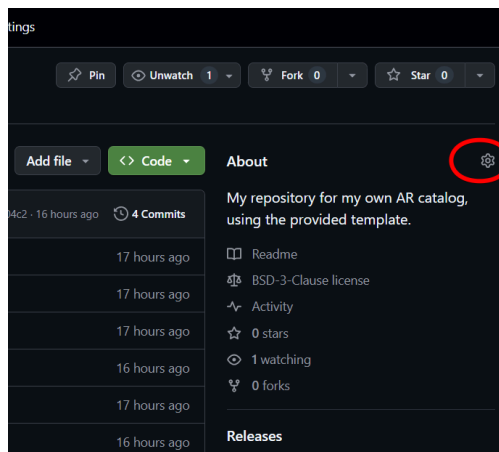
Adding a Binder Link to an Interactive Jupyter Notebook

- In the README (and on the homepage of your site), you will notice a button that says Binder, which when clicked will open the `catalog_demo.ipynb` on a cloud-hosted server.
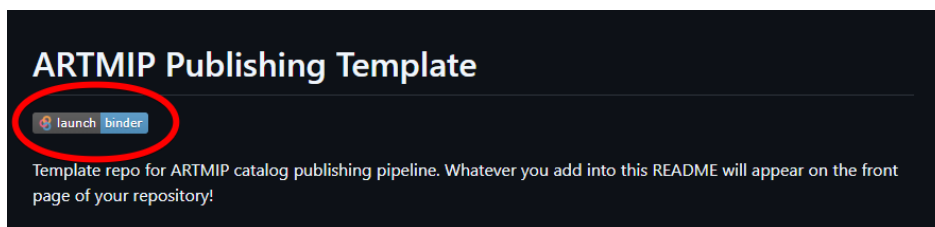
- However, as it stands, this binder link is configured to the copy of the notebook on the template repository, not your own repository. This can be seen in the link

- To configure it to the copy of the notebook on your own repository, open the README and change the `jbbutler/artmip_template` portion of the hyperlink to `<your GitHub username>/<repository name>`
  - You can do this either in the browser on GitHub, or on your copy of the repository on CryoCloud. If the former, click the green 'Commit' button and then 'Commit changes' with a message of your choosing! If the latter, make sure you add, commit, and push your changes!

- Click the button now, and after a few minutes, it should take you to an interactive version of the notebook in your repo!
  - Note: you may be wondering how we were able to set up the environment to import the relevant packages in our notebook. The `environment.yml` file in the repo contains all the packages necessary to run the notebook in Binder. If you edit the notebook to include more packages, make sure you update this file to include these new packages!

Successfully built websites

- Kyle; https://ksmattingly.github.io/artmip-publish-mattingly_v2
- Jimmy: https://jbbutler.github.io/artmip-publish-jimmy
- Serena: https://serenarosemary.github.io/artmip-publish-serena
- Christine: https://shieldsca.github.io/artmip-publish-shields
- Fernando: https://fperez.github.io/artmip-publish-test