

## GETTING STARTED WITH THE UCM JUPYTERHUB

This document contains an introduction to using UC Merced's JupyterHub, providing access to students, troubleshooting some of the most common issues, and additional resources to get started teaching a course with JupyterHub. You can experiment with a sample implementation here. If you have any questions about these resources, please reach out to Rosemarie Bongers (Applied Math) or Sarvani Chadalapaka (CIRT).

**What is JupyterHub?** The UC Merced JupyterHub allows you to host and distribute Jupyter notebooks to your students. These notebooks integrate text or markdown blocks with code blocks so that you can provide templates and code blocks to your classes. Students can then freely modify these templates to add their own text, code, and answers; this can be used for in-class coding demonstrations, computational explorations, or homework exercises. Students can submit notebooks to you either by downloading the modified notebook or exporting to a pdf. Some example notebooks can be found in the same folder as this document.

Once the notebook has been distributed to students, it can be used entirely within a web browser. Students can execute or step through code blocks using the notebook interface. You can also execute any block by putting a cursor in it and pressing Ctrl-Enter or Cmd-Enter. Code blocks can be in many different languages including Python and R.

**What do Jupyter notebooks look like?** See the reference notebooks here:

- A sample homework 0 to get students used to using the notebooks.
- A sample homework problem from Math 032 involving simulation.
- A sample in-class demonstration from Math 032 involving graphing.

**How do I set up JupyterHub for my class?** Once JupyterHub is set up, then it can be accessed directly via a link that you disseminate to students; from the student end, it is as simple as clicking and logging in. The following setup relies on having a linked GitHub repository; this is where you'll add or modify notebooks, and each time the student logs in to JupyterHub it will sync and pull in your latest content. The actual setup is fairly short:

- Create a GitHub repository; this is where you'll host the actual notebooks to distribute. (If you have not done this, you'll need to make a GitHub account). Make sure that the visibility is set to *public*.
- Use nbgitpuller to generate a direct link to JupyterHub.
- Disseminate the link to students, e.g. by posting it on Canvas.

More detailed instructions are at the end of this document.

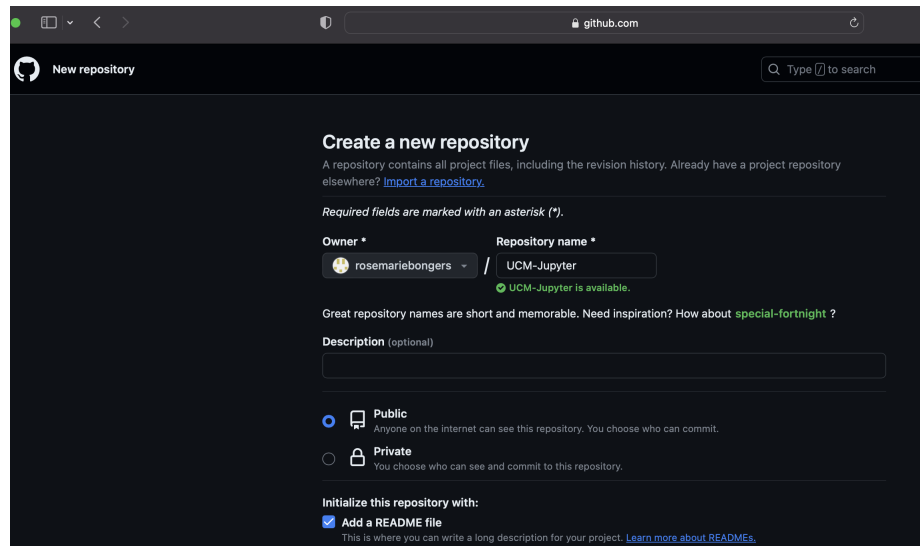
**What frequent issues do you run into?**

- Authentication issues with the campus JupyterHub. Sometimes these are simple (i.e. the student did not choose to log in with their UCM credentials and chose a different option), and sometimes they are due to an outage. In the case of an outage, it is helpful to have an alternative hosting space. One solution is to post the notebooks on Canvas (in .ipynb format), in which case they can download the notebooks and run them directly on Jupyter.org.
- Students lose their work or cannot recover the original version of the notebook. There are savepoints that students can revert to within Jupyter, but it is again helpful to have an original, static version that the students can refer to (i.e. posting the notebooks on Canvas).
- Student inexperience in coding. This is the hardest one to work with, but well-scaffolded notebooks can help.

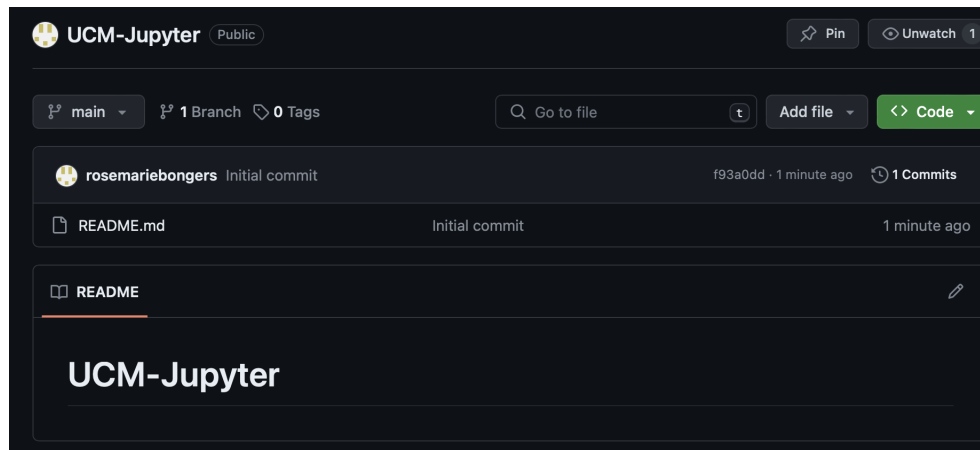
## DETAILED SETUP INSTRUCTIONS

**Step 0:** This PDF is interactive with a number of hyperlinks, although they may be disabled depending on your pdf viewer or if you are using it directly on GitHub. You can either download the PDF or see the **appendix** at the end of the document for a list of all the links.

**Step 1:** Create a new GitHub repository after logging into your account at GitHub. This repository's name is UCM-Jupyter:



After creation, it only contains one file, README.md. This repository is publically accessible here:



Although this is beyond the scope of this guide, you can also use GitHub through the command line directly. See [here](#) for generating a personal access token and [here](#) for how to push an update to a remote repository.

**Step 2:** Use the nbgitpuller link generator to generate a link that will sync with your latest GitHub repository updates. You'll need the URLs for both the campus JupyterHub (<https://ucmerced.2i2c.cloud>, accessible [here](#)) and your personal GitHub repository. Here's an example of the link creation:

The screenshot shows the 'nbgitpuller link generator' web interface. At the top, it says 'Use the following form to create your own nbgitpuller links.' Below this is a 'Note' box with a blue header and white text: 'Consider using the nbgitpuller link generator browser extension instead! Available for Firefox and Chrome.' The main form has three tabs: 'JupyterHub' (selected), 'Launch from Canvas', and 'Binder'. The 'JupyterHub' tab shows a text input for the 'JupyterHub URL' with the value 'https://ucmerced.2i2c.cloud/' and a green checkmark. Below this is a note: 'The JupyterHub to send users to nbgitpuller must be installed in this hub.' The 'Git Repository URL' field contains 'https://github.com/rosemariebongers/UCM-~' with a green checkmark. To its right is a 'branch' dropdown menu set to 'main' with a green checkmark. Below the dropdown is a note: 'Use main instead of master for new GitHub repositories'. The 'File to open' field contains 'index.ipynb' with a green checkmark. Below this is a note: 'This file or directory from within the repo will open when user clicks the link.' The 'Application to Open' section has a list of radio buttons: 'Classic Jupyter Notebook' (selected), 'RetroLab', 'JupyterLab', 'RStudio', 'Shiny', and 'Custom URI'.

**nbgitpuller link generator**

Use the following form to create your own `nbgitpuller` links.

**Note**

Consider using the [nbgitpuller link generator browser extension](#) instead! Available for [Firefox](#) and [Chrome](#).

**JupyterHub** [Launch from Canvas](#) [Binder](#)

`https://ucmerced.2i2c.cloud/hub/user-redirect/git-pull?repo=https%3A%2F%2Fgithub.com%2Frosemariebongers%2FUCM-~`

**JupyterHub URL** `https://ucmerced.2i2c.cloud/` ✓

The JupyterHub to send users to `nbgitpuller` must be installed in this hub.

**Git Repository URL** `https://github.com/rosemariebongers/UCM-~` ✓

**branch** `main` ✓

Use `main` instead of `master` for [new GitHub repositories](#)

**File to open** `index.ipynb` ✓

This file or directory from within the repo will open when user clicks the link.

**Application to Open**

- ☒ Classic Jupyter Notebook
- ☐ RetroLab
- ☐ JupyterLab
- ☐ RStudio
- ☐ Shiny
- ☐ Custom URI

The generated link is accessible [here](#).

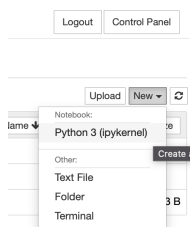
**Step 3:** Disseminate the link that you just generated. The first time your students click it, they will be sent to an authentication page:



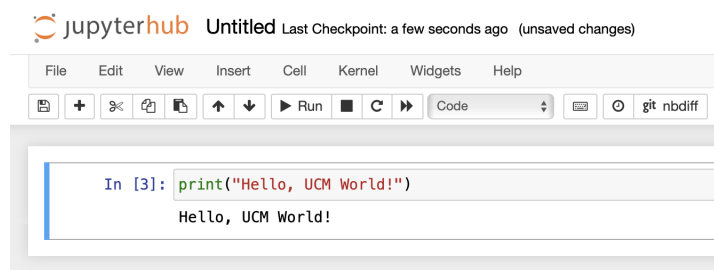
After you enter your UCM credentials, you may run into a prompt to select a kernel; choose whatever is appropriate for your applications (in this case, we're using Python). After a short sync, you should be able to see all of your files within Jupyter:



If you had any notebooks already, they would be accessible here. To get started writing a new one, click "New":



Once you've done this, happy coding!



## APPENDIX: LINKS AND RESOURCES

**Links used in this setup document:**

- GitHub: <https://github.com>
- UCM GitHub repository: <https://github.com/rosemariebongers/UCM-Jupyter>
- Generating a GitHub personal access token: <https://docs.github.com/en/authentication/keeping-your-account-and-data-secure/managing-your-personal-access-tokens>
- Pushing a commit to a remote repository: <https://docs.github.com/en/get-started/using-git/pushing-commits-to-a-remote-repository>
- Link generator (nbgitpuller): <https://nbgitpuller.readthedocs.io/en/latest/link.html>
- Default UCM JupyterHub landing page: <https://ucmerced.2i2c.cloud/>
- JupyterHub server implementing all the sample notebooks: <https://bit.ly/4bAQpc1>

**Resources and contact people:**

- UC Merced Cyberinfrastructure and Research Technologies: <https://it.ucmerced.edu/CIRT>
- Rosemarie Bongers (Applied Mathematics): [rosemariebongers@ucmerced.edu](mailto:rosemariebongers@ucmerced.edu)
- Sarvani Chadalapaka (CIRT): [schadalapaka@ucmerced.edu](mailto:schadalapaka@ucmerced.edu)