GitHub Documentation

The DITI GitHub is hosted under the “NULabNortheastern” organization. We use the [Digital Assignment Showcase](https://github.com/NULabNortheastern/digitalassignmentshowcase) repository to store all the classroom materials we have created during partnerships.

The GitHub is the primary place to publish documentation about the materials so future DITI members and faculty can use and access them easily.

### GitHub Organization

The GitHub folders are organized like the [DITI 2-Lane: Module Materials](https://drive.google.com/drive/folders/16bBdNVAZ2T4A7POWJ1gu45LcRtqPbodY?usp=sharing) folders - by the following digital proficiencies:

* coding\_qualitative
* coding\_quantitative
* citation-management
* data-gathering
* data-management
* digital-archiving
* digital-scholarly-publishing
* website-building
* audio-editing\_podcasting
* data-visualization
* digital-communication\_presentation
* image-design\_photo-editing
* mapping
* multi-domain-modules(for any single-visit modules that fall into more than one domain category)
* text-analysis
* text-encoding
* surveys
* data-ethics
* data-modeling

Each *module* in a partnership will need its own GitHub folder to house the module materials (slides, handout(s), data or demo content, and a documentation ‘README.md’ markdown file). For partnerships that have multiple modules, please remember that each module will need to be filed by digital proficiency.

The naming conventions for GitHub module folders are as follows:

***semesteryear-faculty-coursenumber-module***

example: su22-fuchs-grmn3800-infographics

**A note on file names:** Try not to put any spaces in file names for any resources you create. You can use hyphens or underscores, consistently with other examples that you consult. Make sure not to use any special characters, such as quotation marks, in your file names. Try to use just hyphens, underscores, and letters.

### How GitHub Works

GitHub is a **cloud storage repository** that connects to your local computer. In order to make changes to our GitHub, you can either a) do so directly online on the GitHub page, or b) make the changes on your local file and push those onto our repository. The latter is the best way to create new folders, add new materials, etc., while the former is best for editing READ.me files.

To edit our GitHub, you *must* have a GitHub account and *must* be a contributor/collaborator.

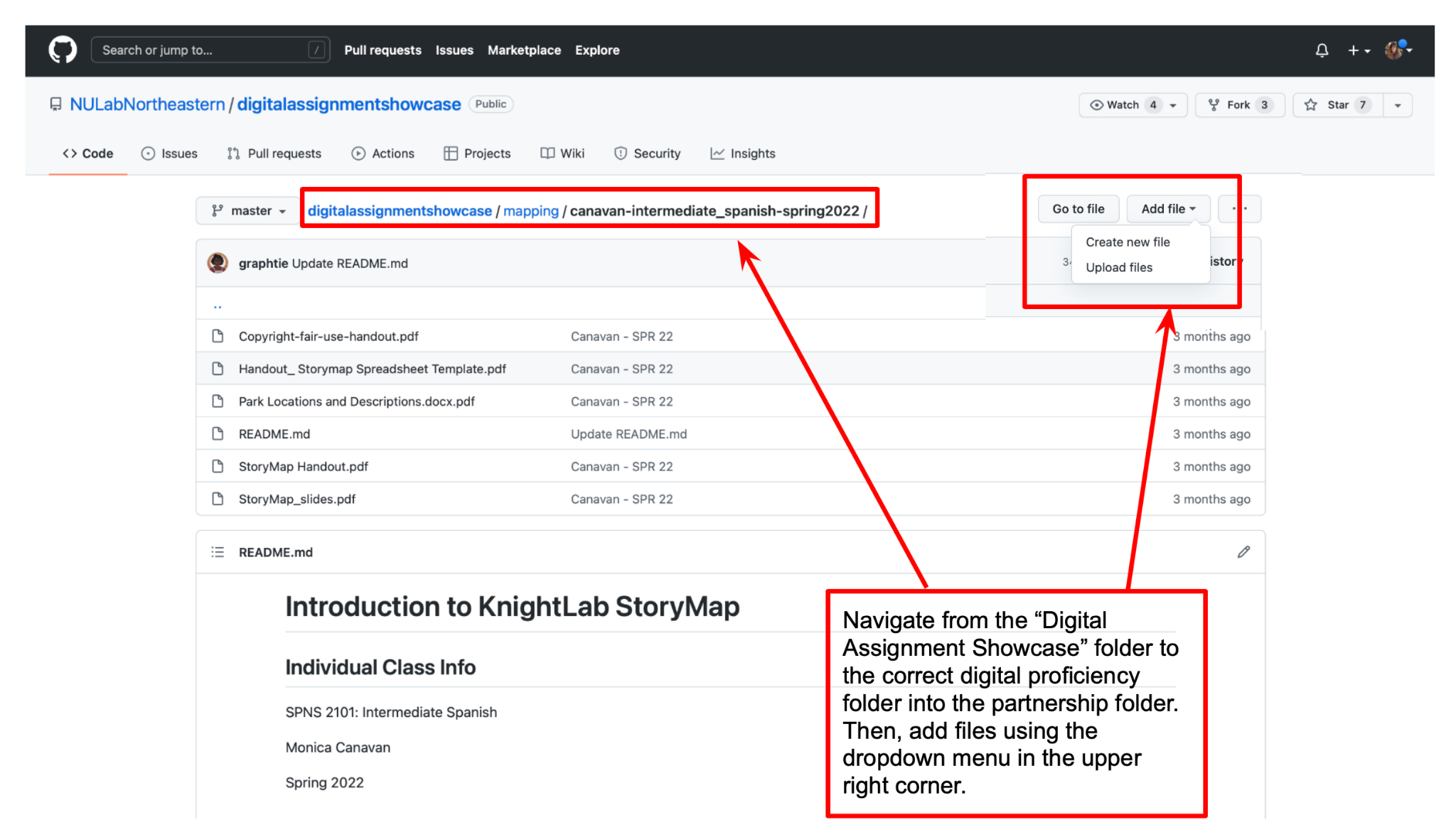
Before you can begin adding module materials to GitHub, you must download all of the materials (handouts, slides, etc.) in PDF format. For the README.md documentation file, use the information in the “[Module Name] Documentation” file. For example, in the “Mapping” folder in the GoogleDrive there is a document called “StoryMap Documentation.” Using a markdown editor (like Typora), create a README.md file with that information and add the class information and the DITI consultants names.

### GitHub Online

GitHub online is very simple to use, *particularly if you have gathered all module materials together in the proper formats* (PDFs for slide decks, handouts, data sets, etc., and a markdown file for the README.md documentation file) ahead of time.

To make changes (editing, uploading, deleting, etc) to the GitHub directly from your browser:

1. Be sure that you’re logged into your GitHub account.
2. From the ‘Digital Assignment Showcase’ navigate into the appropriate digital proficiency folder you need to add materials to, or edit.
3. You can drag and drop PDF or markdown files directly from your computer into the file by clicking the “add file” dropdown menu in the upper right corner of the window and selecting “upload file”.



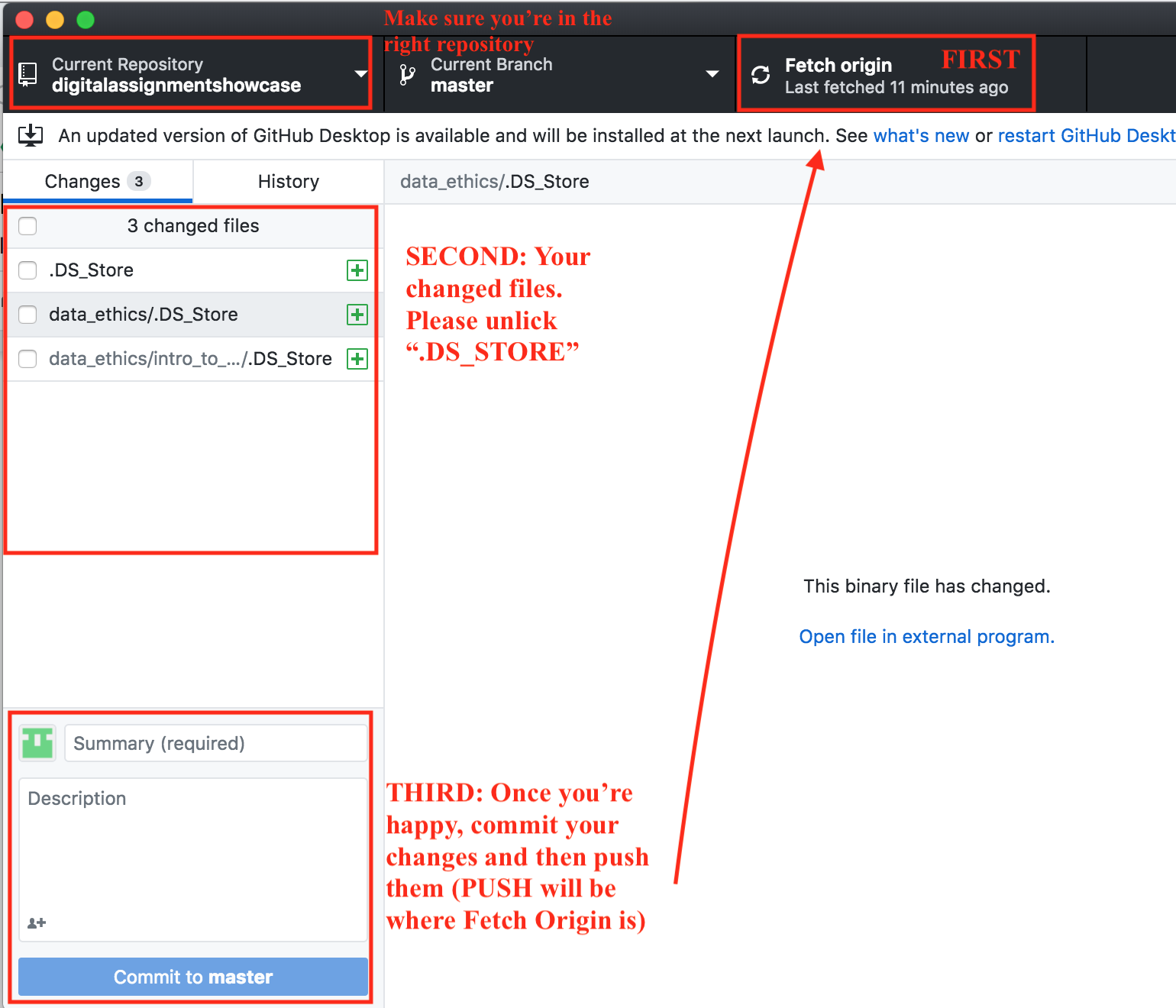
### 

### GitHub Desktop

GitHub Desktop is an application that connects the cloud repository to your local computer. First, you need to install GitHub Desktop and then connect your local computer to the cloud repository. In order to do this, in the “File” menu, click “Clone a Repository” and then use our repository’s URL (<https://github.com/NULabNortheastern/digitalassignmentshowcase.git>).

Follow these steps to make changes (editing, uploading, deleting, etc) to the GitHub:

1. You **MUST** **CLICK** “Fetch Origin”/”Pull” every time you want to make any edits, even if it’s a small one. This way, your local version is up-to-date with the cloud version. Otherwise, there could be GitHub conflicts.
2. Make your edits on your local file, whether that’s editing a README file, adding a new PDF, &c.
3. When you want to upload your new files/folders into the cloud repository:
   1. make sure the edited files are clicked (if you’re on a Mac, unclick anything with DS\_Store)
   2. Write a summary of what you are changing (like 3-5 words)
   3. Click “commit to master”
   4. Click “**Push**” and voila, it should be on GitHub! Feel free to check our GitHub on your browser to be sure.



### 

### 

### GitHub Command Lines

If you prefer, you can also use GitHub on the command line, which will pretty much do the same thing as GitHub Desktop, just without the graphical user interface. Follow these [guidelines](https://kbroman.org/github_tutorial/pages/routine.html), which explain how to connect your local computer to your GitHub account and specific repository as well as how to pull, commit, and push.

Don’t worry if you make a mistake! GitHub is a version control repository, so it’s really easy to be able to go back to another version pre-mistake.