Intro to Git & GitHub

Sophie Shepherd

@sophshepherd

Welcome!

Girl Develop It is here to provide affordable and accessible programs to learn software through mentorship and hands-on instruction.

Rules

- We are here for you!
- Every question is important
- Help each other
- Have fun

Welcome!

- What's your name?
- What is your Git/GitHub experience?
- What do you hope to get out of the class?

What we will cover

- What is version control and why should we care?
- Setting up GitHub Desktop & GitHub.com
- Collaboration with others
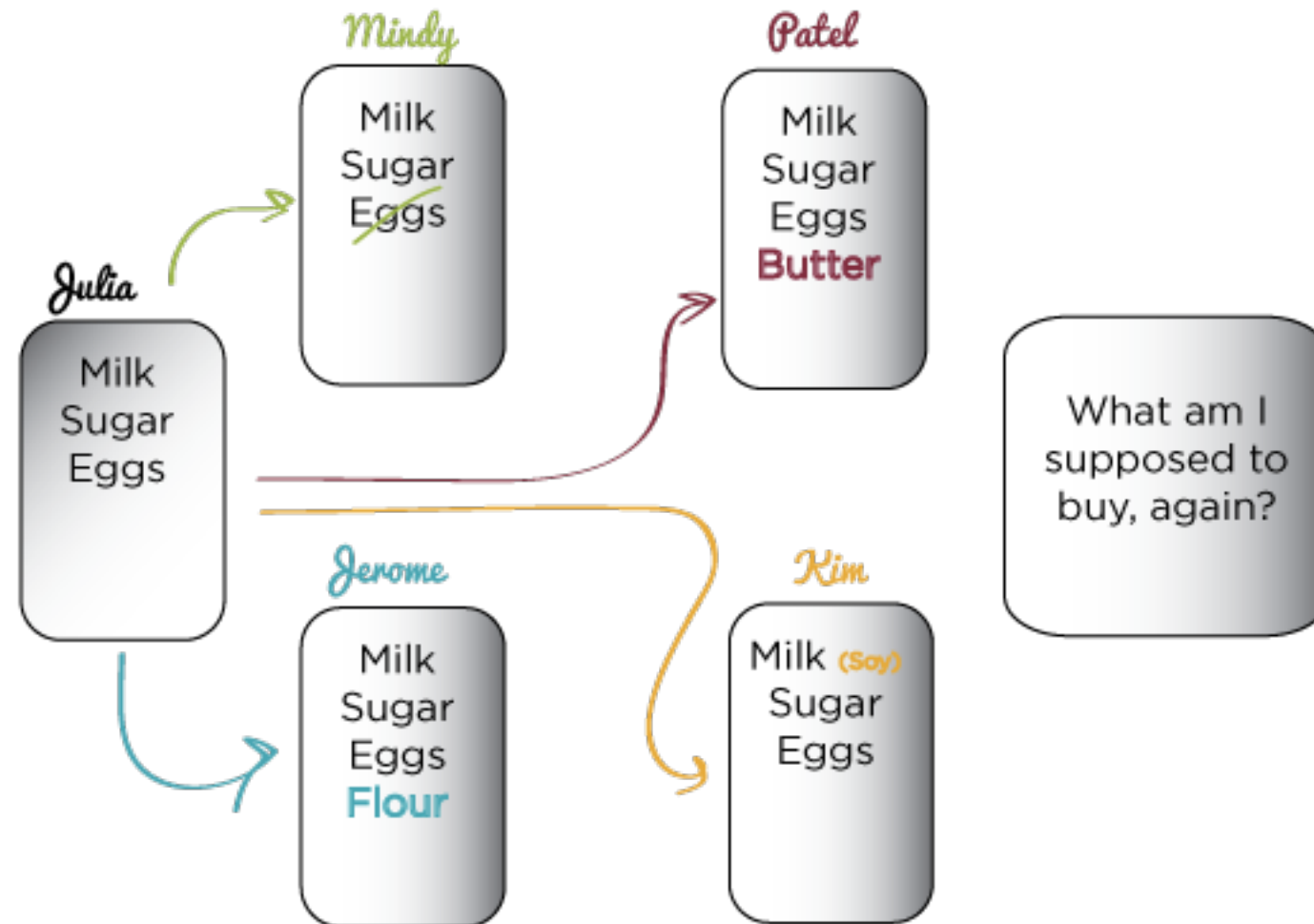
What is version control?

Track & Revert Changes

Mistakes happen. Wouldn't it be nice if you could see the changes that have been made and go "back in time" to fix something that went wrong?
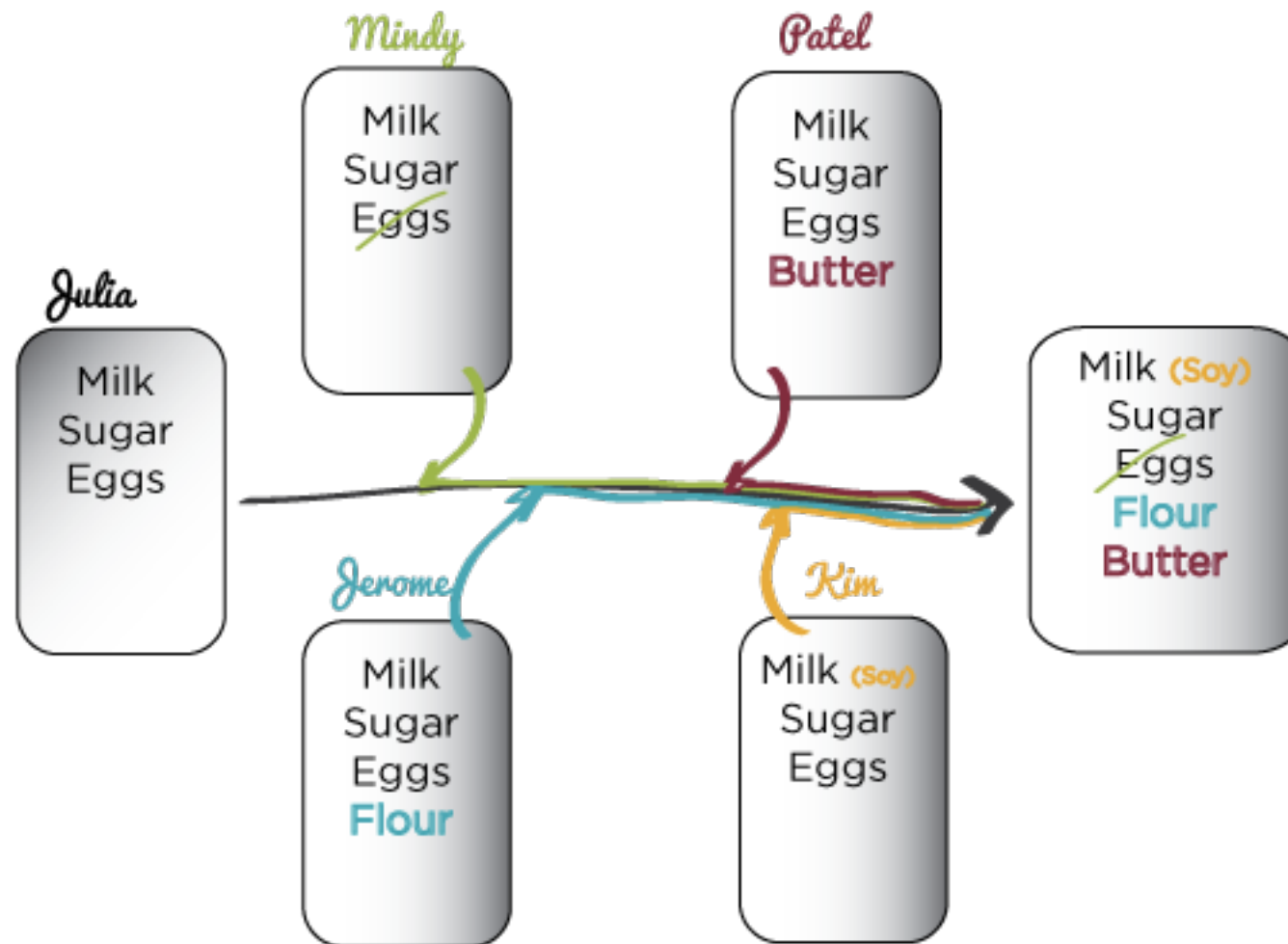
What is version control?

Collaborate

Create anything with other people, from academic papers to entire websites and applications.

What is version control?

What is version control?

History

- 1990s — CVS (Concurrent Version Systems)
- 2000s — SVN (Apache Subversion)
- 2005 — Git
- 2008 — GitHub Founded

Git Benefits

- Fast -- add to your team and code base quickly
- Distributed, every user has a local copy
- Each commit has a corresponding hash (track changes from everyone)
- Everyone has a local copy of the history

Installation

git-scm.com/downloads
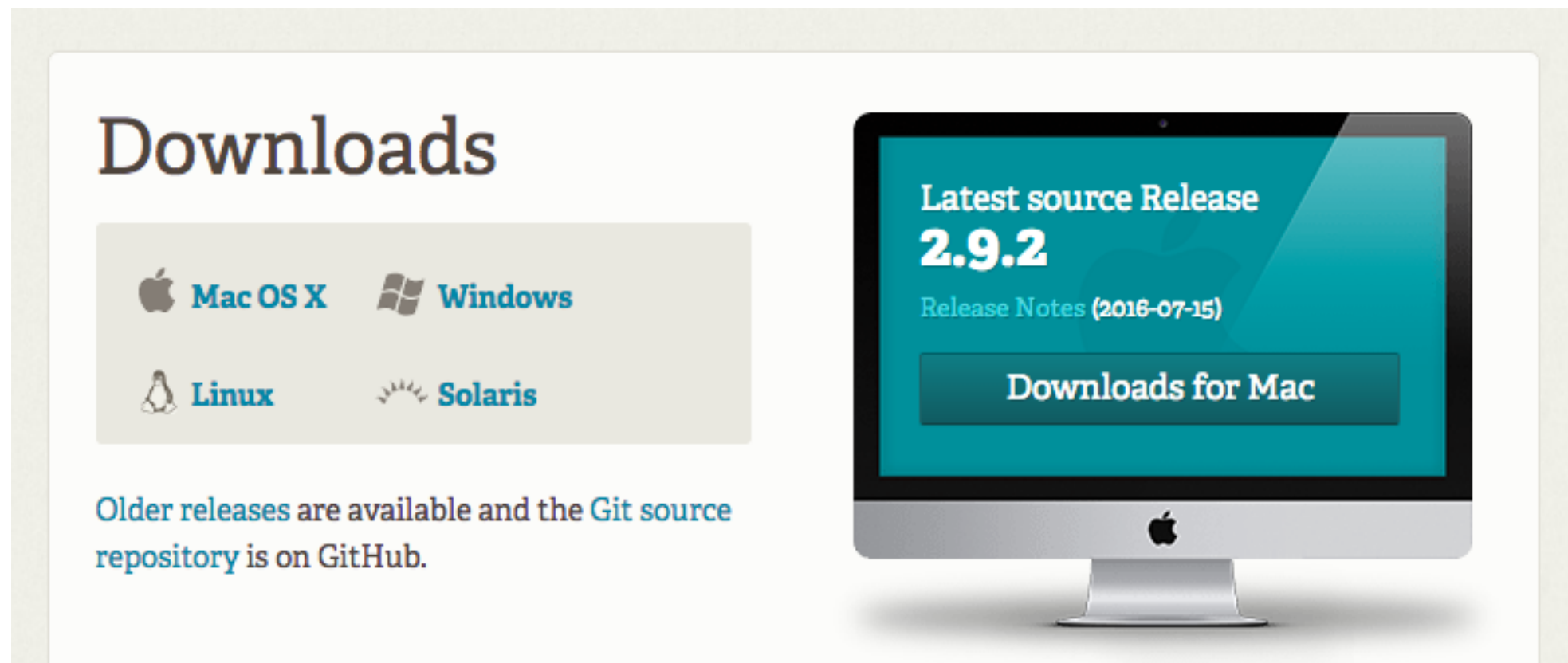
Sign Up for GitHub

github.com/join

Join GitHub
The best way to design, build, and ship software.

Step 1: Set up a personal account	Step 2: Choose your plan	Step 3: Tailor your experience

Create your personal account

Username

[   ]

This will be your username — you can enter your organization's username next.

Email Address

[   ]

You will occasionally receive account related emails. We promise not to share your email with anyone.

Password

You'll love GitHub
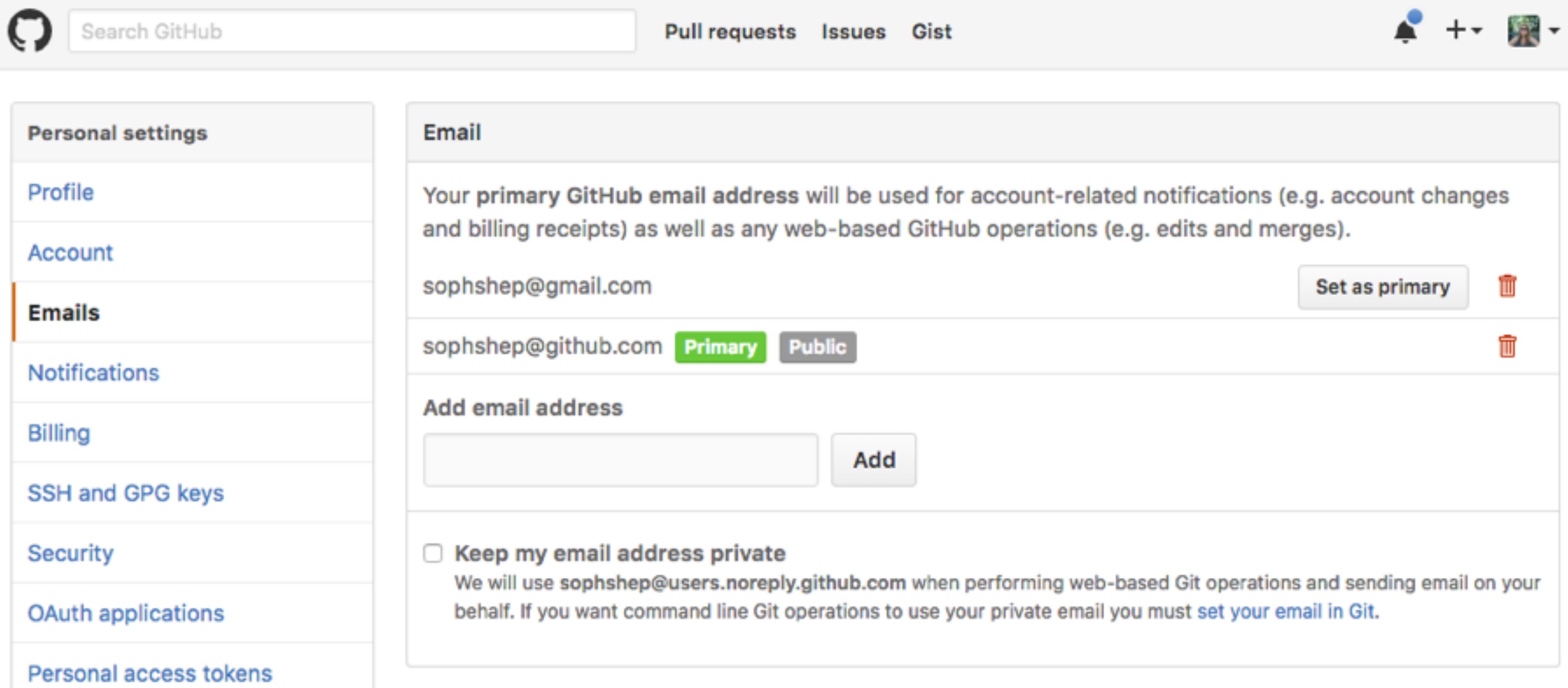
Unlimited collaborators

Unlimited public repositories

✓ Great communication

✓ Frictionless development

✓ Open source community

GitHub Settings

github.com/settings/email

Get GitHub Desktop

desktop.github.com
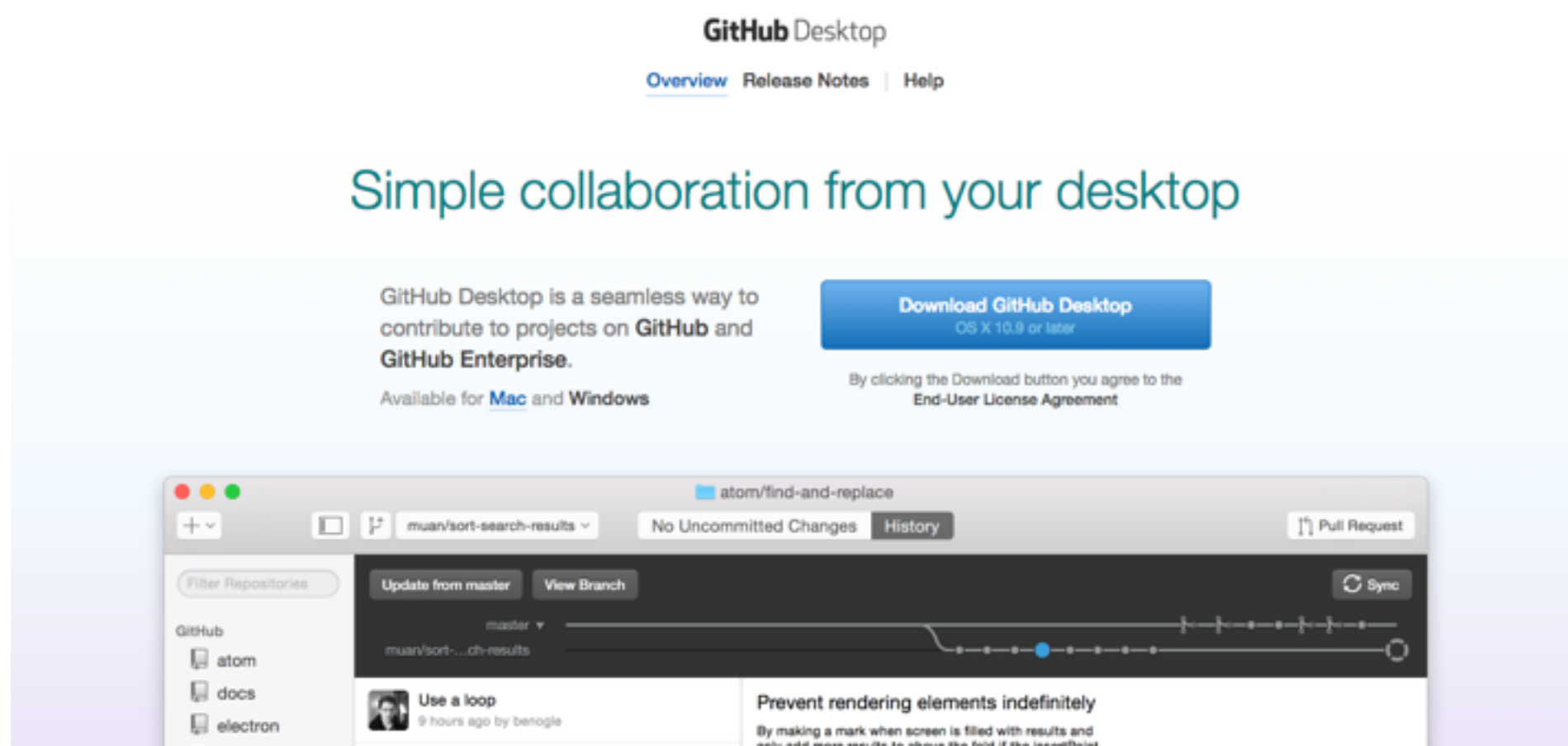
Connect GitHub & Desktop

- Select `GitHub Desktop` menu >
- `Preferences` >
- `Accounts` >
- Enter GitHub credentials

Connect GitHub & Desktop

- Preferences >
- Advanced >
- Enter name
- Enter email

Break?

Git Commands

services.github.com/resources/

GitHub
GIT CHEAT SHEET

Git is the open source distributed version control system that facilitates GitHub activities
desktop. This cheat sheet summarizes commonly used Git command line instructions for qu

INSTALL GIT
GitHub provides desktop clients that include a graphical user interface for the most common repository actions and an automatically updating command line edition of Git for advanced scenarios.

GitHub for Windows
https://windows.github.com

GitHub for Mac
https://mac.github.com

Git distributions for Linux and POSIX systems are available on the official Git SCM web site.

Git for All Platforms
http://git-scm.com

CONFIGURE TOOLING
Configure user information for all local repositories

MAKE CHANGES
Review edits and craft a commit transaction

`$ git status`

Lists all new or modified files to be committed

`$ git diff`

Shows file differences not yet staged

`$ git add [file]`

Snapshots the file in preparation for versioning

`$ git diff --staged`

Shows file differences between staging and th

`$ git reset [file]`

Unstages the file, but preserve its contents

`$ git commit -m "[descriptive messag`

Records file snapshots permanently in version

GIT CHEAT SHEET

REFACTOR FILENAMES
Relocate and remove versioned files

`$ git rm [file]`

Deletes the file from the working directory and stages the deletion

`$ git rm --cached [file]`

Removes the file from version control but preserves the file locally

`$ git mv [file-original] [file-renamed]`

Changes the file name and prepares it for commit

SUPPRESS TRACKING
Exclude temporary files and paths

```
*.log
build/
temp-*
```

A text file named .gitignore suppresses accidental versioning of files and paths matching the specified patterns

`$ git ls-files --other --ignored --exclude-standard`

Lists all ignored files in this project

SAVE FRAGMENTS
Shelve and restore incomplete changes

REVIEW HISTORY
Browse and inspect the evolution of project files

`$ git log`

Lists version history for the current branch

`$ git log --follow [file]`

Lists version history for a file, including renames

`$ git diff [first-branch]...[second-branch]`

Shows content differences between two branches

`$ git show [commit]`

Outputs metadata and content changes of the specified commit

REDO COMMITS
Erase mistakes and craft replacement history

`$ git reset [commit]`

Undoes all commits after [commit], preserving changes locally

`$ git reset --hard [commit]`

Discards all history and changes back to the specified commit

SYNCHRONIZE CHANGES
Register a repository bookmark and exchange version history

About Repositories

github.com/sophshep/gdi-intro-to-github

- A repository is a project
- Local & Remote
- "Watch" this repository
- Issues
- README.md

Collaborating

- **Forks:** A fork is a personal copy of another user's repository that lives on your account.
- **Branches:** A branch is a parallel version of a repository. It is contained within the repository, but does not affect the primary or master branch allowing you to work freely without disrupting the "live" version.

Clone Repo

git clone git@github.com:sophshep/gdi-intro-to-github.git

github.com/sophshep/gdi-intro-to-github
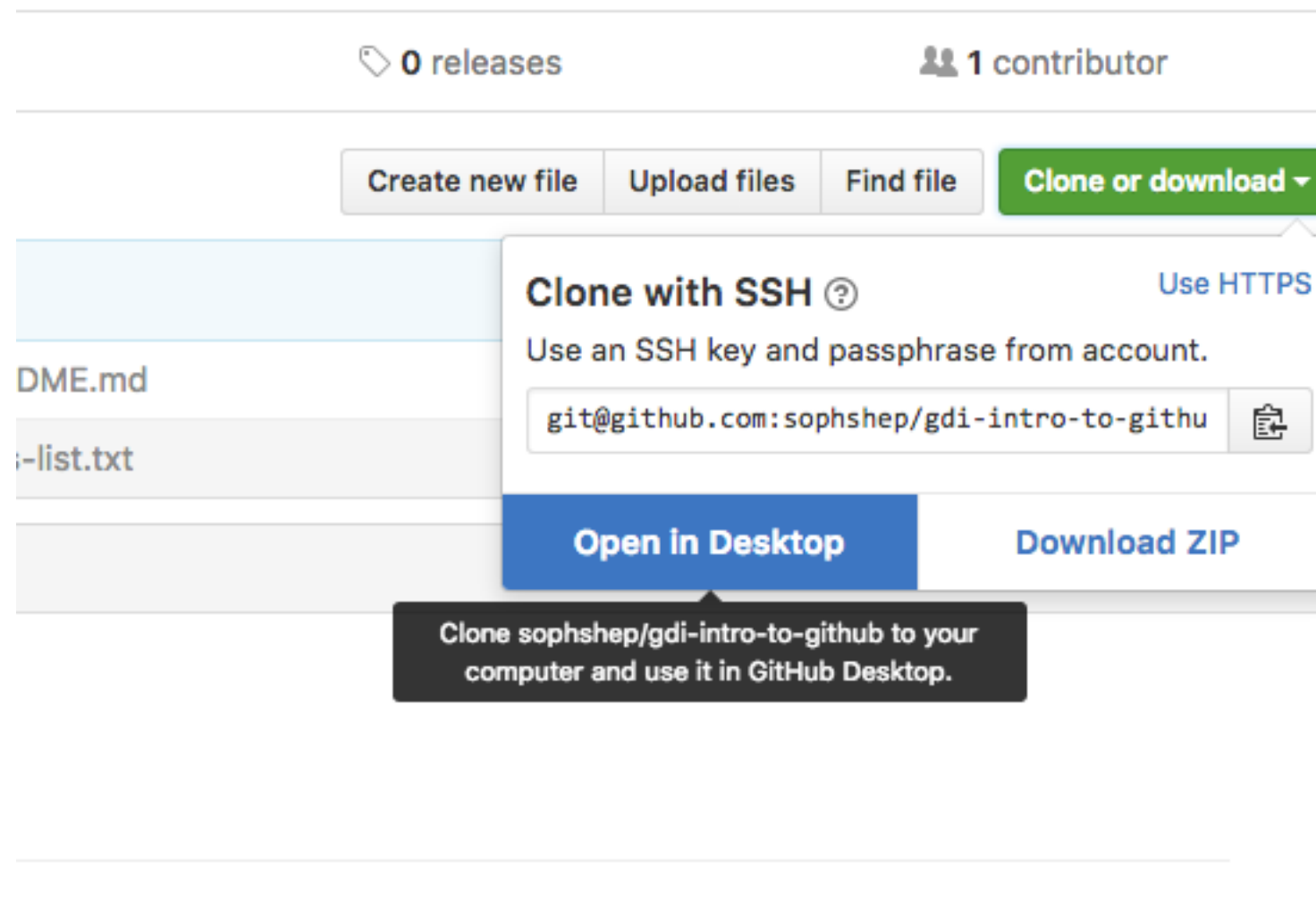
About Branches

- A branch represents an independent line of development, does not interfere with `master`
- New commits are recorded in the history for the current branch.

Make branch

`git checkout -b sophie-branch`
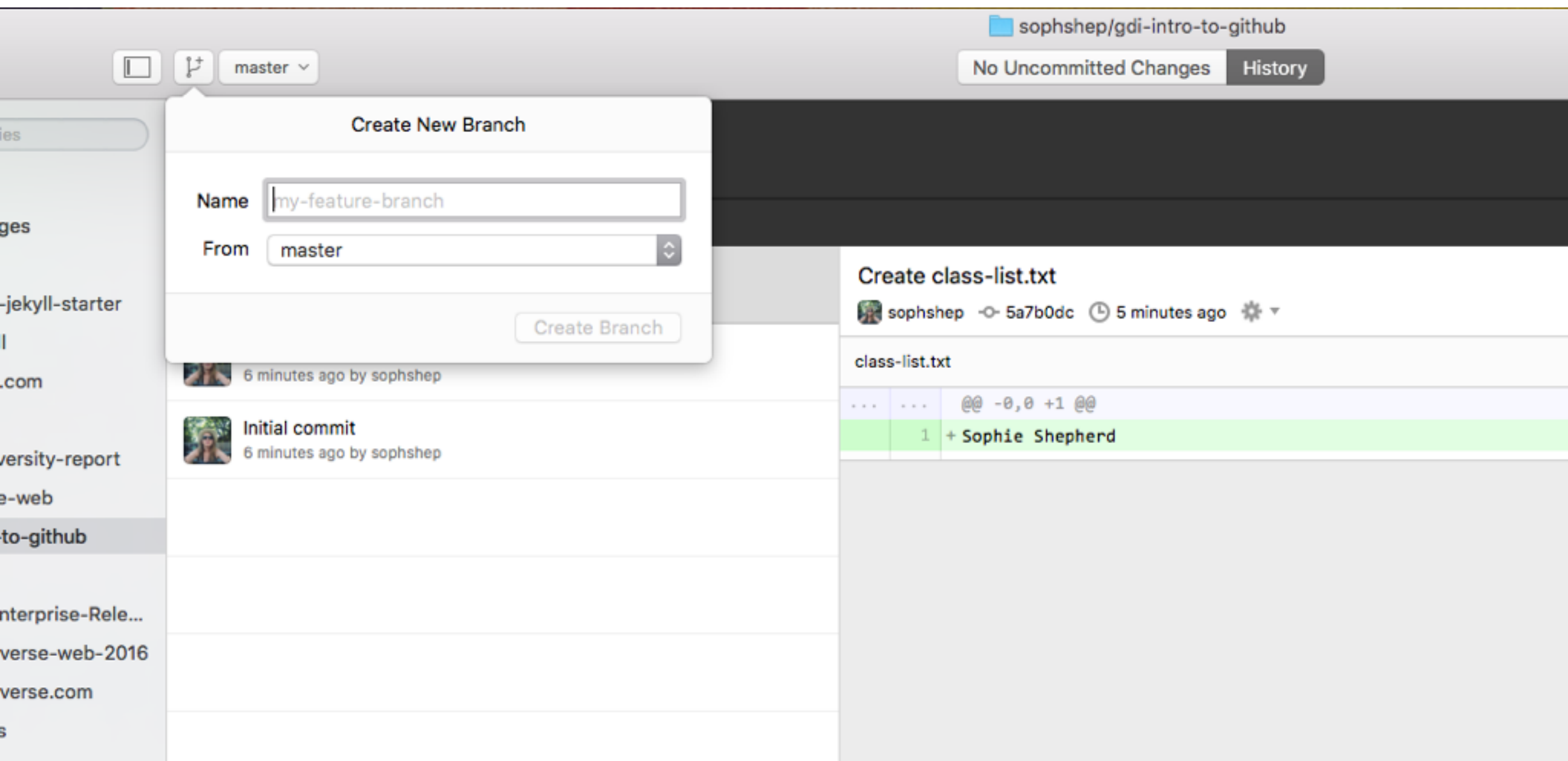
Make changes

- Add yourself to class list
- Add a new file

About Commits

- A commit, or "revision", is an individual change to a file (or set of files).
- Like a snapshot of the repository
- Commits should a commit message which is a brief description of what changes were made.

Commit changes

Publish Branch

(aka "push")
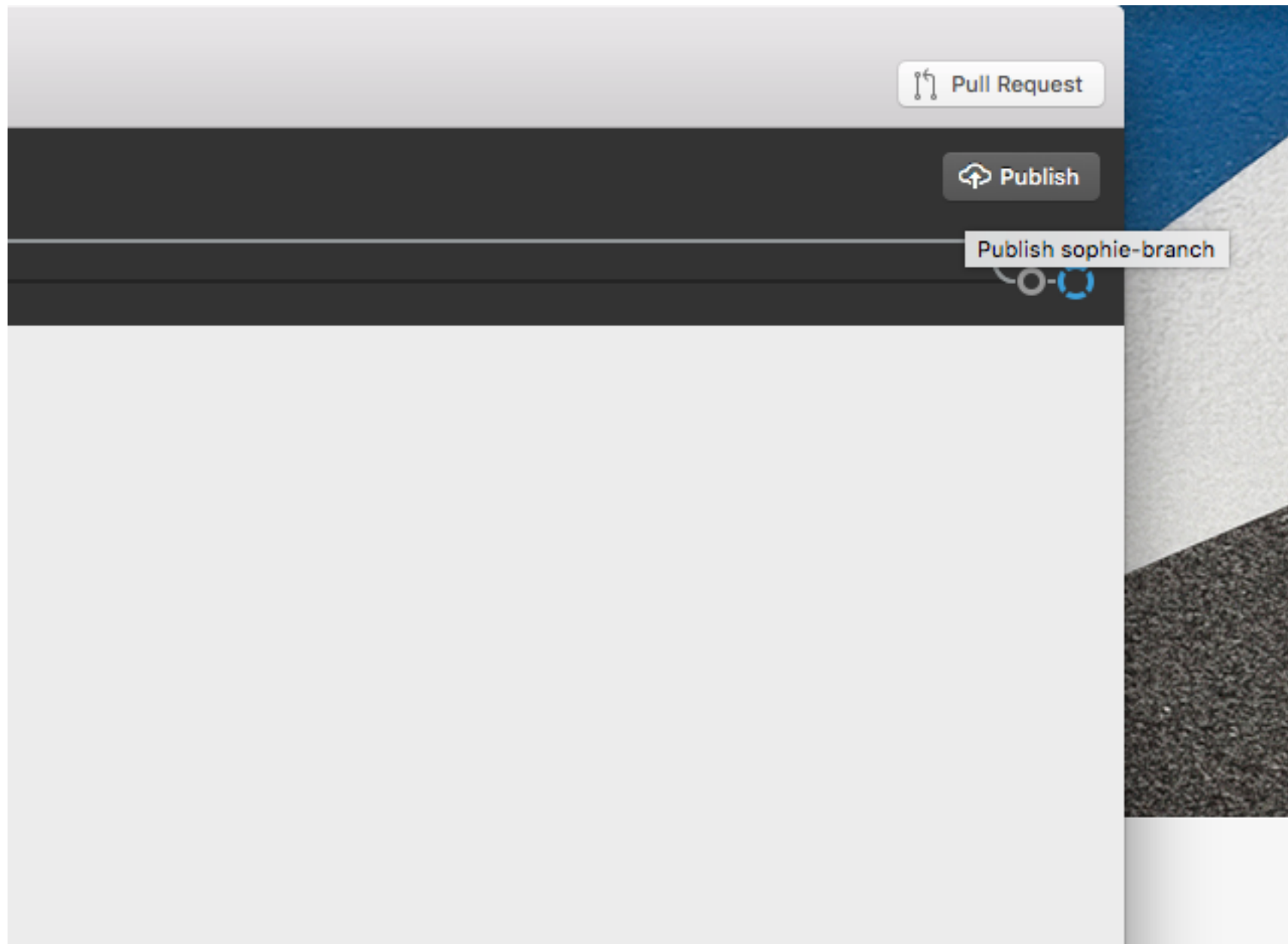
About Pull Requests

- Pull requests let you tell others about changes you've pushed to a repository on GitHub.
- Once a pull request is sent, interested parties can review the set of changes, discuss potential modifications, and even push follow-up commits if necessary.

Create Pull Request

Create Pull Request

sophshep / **gdi-intro-to-github**

⊙ Watch ▾ 0   ★ Star 0   ⑂ Fork 0

<> Code   ① Issues 0   Pull requests 0   ▤ Wiki   ⤳ Pulse   �ılı Graphs   ⚙ Settings

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also **compare across forks**.

⇅   base: **master** ▾ ... compare: **sophie-branch** ▾   ✓ **Able to merge.** These branches can be automatically merged.

Adding myself to list

Write   Preview   AA▾ B *i*   " <> ⊂⊃   ☰ ☰ ☱   ↰▾ @ ▱

This change adds me to the class list, and also includes some photos of my cat.|

Labels ⚙
None yet

Milestone ⚙
No milestone

Assignees ⚙
No one—assign yourself

Merge Pull Request

Learn More

help.github.com/articles/github-glossary/

GitHub Git Cheat Sheet

guides.github.com

atlassian.com/git