Collaborating with others on GitHub

Eirini Zormpa, 5 July 2023



Summary of last week

Last week we learned:

- That GitHub is an online website to host repositories tracked with git, something very useful for collaboration
- How to connect local and remote repositories (in this case, ones shared on GitHub)
- That branches are independent lines of development, useful for experimenting with new features without breaking anything
- W How to create and merge branches
- V How to open and merge pull requests to suggest changes

Questions from last time?

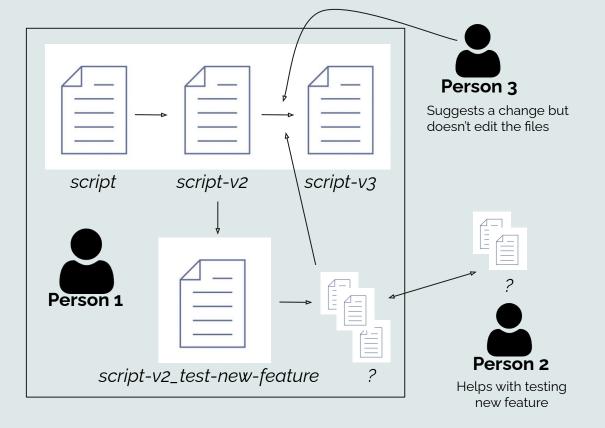
Learning objectives

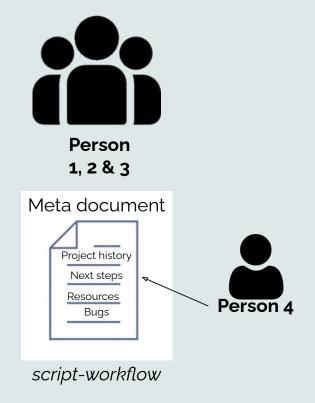
By the end of this session, you'll be able to:

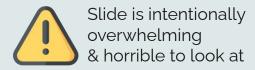
- Explain what a fork is and when to use it
- Request reviews for your pull requests
- Understand when merge conflicts happen and how to resolve them
- Set up a GitHub repository to facilitate collaboration
- Use issues to discuss and document changes

Collaboration mechanisms on GitHub









How to collaborate?

If you have access

The only person that, by default, can make changes to a personal repository is the person that **created it**.

The repo owner can add collaborators, who can contribute directly by committing to main, or by **creating a branch** and submitting a **pull request**.

If you don't have access

If you have not been added as a collaborator, you can make a copy of the repo, called a **fork**, into you own account.

You can then suggest the changes to the original (upstream) repo through a **pull request**.

Collaboration mechanisms on GitHub

No need to type along - just watch 👀



fork vs. branch

How to collaborate?

If you have access





The only person that, by default, can make changes to a personal repository is the person that **created it**.

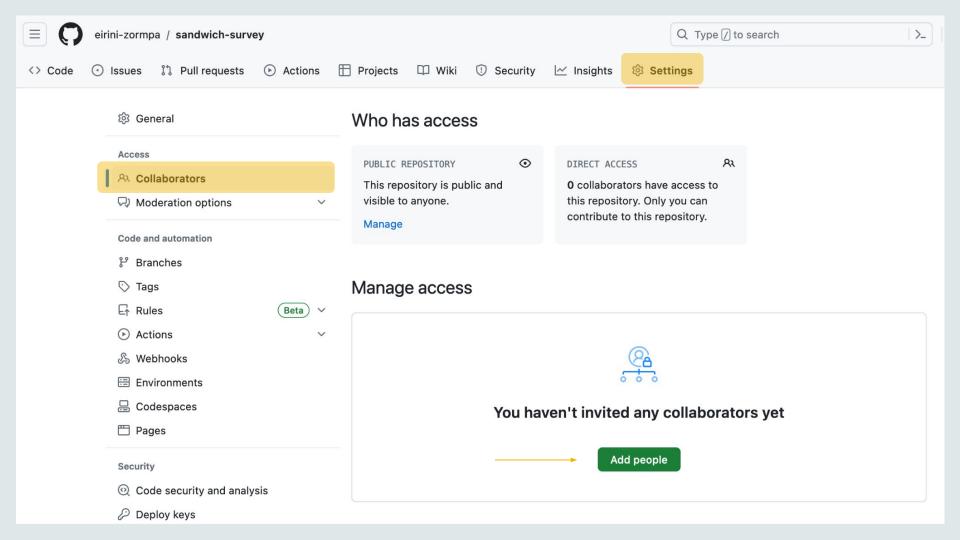
The repo owner can add collaborators, who can contribute directly by committing to main, or by **creating a branch** and submitting a **pull request**.

If you don't have access



If you have not been added as a collaborator, you can make a copy of the repo, called a **fork**, into you own account.

You can then suggest the changes to the original (upstream) repo through a **pull request**.



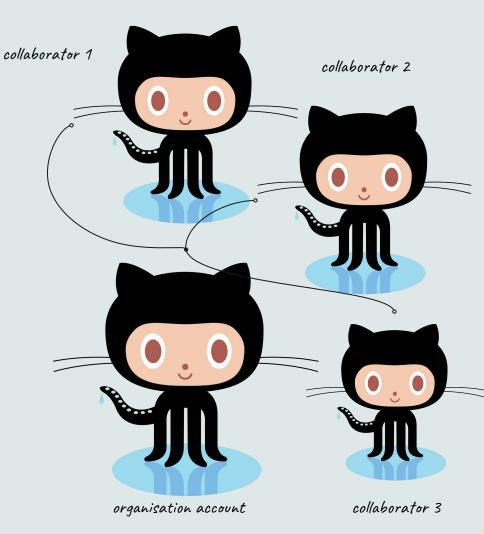
Repository 1 Oh no 😭 Person 3 Person 1, 2 & 3 Meta-document Person 1 itory 2 **Repository 3** Person 3 Person 3 Person 1, 2 & 3 Meta-document Person 1, 2 & 3 Meta-document Person 4 Person 1 Person 2 Person 4 Slide is intentionally Person 1 overwhelming

& horrible to look at

Person 2

GitHub Organisations

- allow you to work with the same collaborators on multiple projects
- keep a collaborations' work centralised with less disruption if one person stops being involved



Enterprise accounts

Account types on GitHub

Personal account

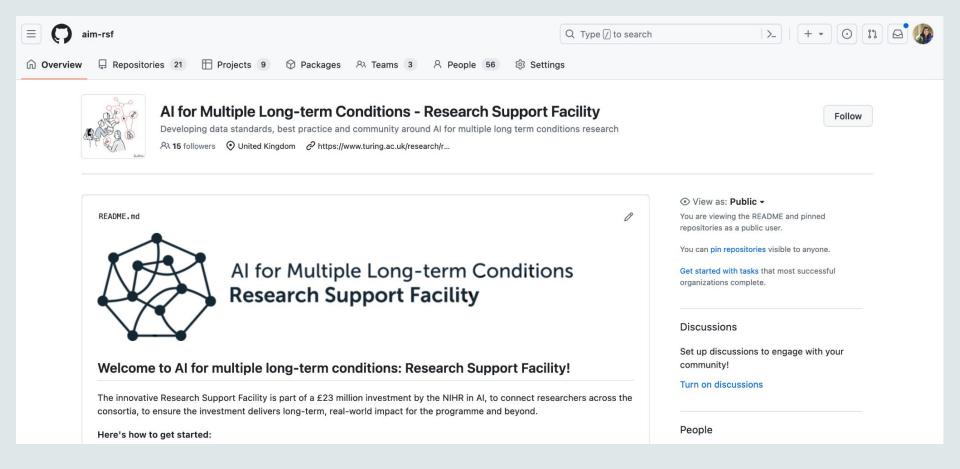
Everyone that uses GitHub has a personal account. Personal accounts can own resources such as repos.

Any time you do something on GitHub (make a commit, submit a pull request, etc), that action is associated with your account.

Organisation account

Organisations are shared accounts, where multiple people can collaborate across many projects at once.

Organisations can own resources, but they are not associated with actions; the user's personal account is.



Collaborate with issues

Issues are like posts in a repository.

Use them to flag things that should be changed: bugs or errors that need to be fixed or new features that could be added to enhance the repository.

When thinking about contributing to a repo, check the issues to see what's needed!



Deleted branches still show up in the branches dropdown #5893

New issue

⊘ Closed

IdanAdar opened this issue on Oct 13, 2018 · 46 comments



IdanAdar commented on Oct 13, 2018 ... Assignees No one assigned Description Labels I've created a branch via GitHub Desktop. Now I'm done with it and so I've deleted it from the GitHub.com UI. None yet In GitHub Desktop though it's "still there". How do I sync the list of branches? Does it sync at all? **Projects** None yet Version Milestone • GitHub Desktop: 1.4.3 beta 1 No milestone • Operating system: macOS 10.14 Development **Steps to Reproduce** No branches or pull requests 1. Create a branch via GitHub Desktop 2. Delete it from the GitHub.com UI **Notifications** Customize

Documenting for collaboration



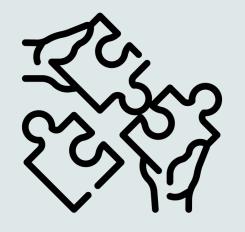
If your repo isn't documented, people won't know what it's about

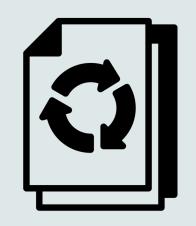
If your repo isn't documented, people won't know what it's about, how to use it

If your repo isn't documented, people won't know what it's about, how to use it and how to contribute.

Documentation









README.md

CONTRIBUTING.md

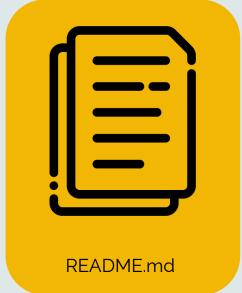
LICENCE

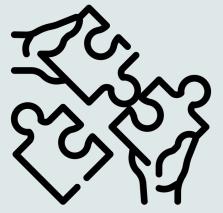
CODE_OF_CONDUCT.md

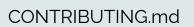
Used to describe the project and provide important information. "README" is recognised by GitHub and displayed as a landing page.

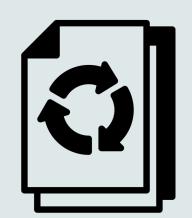
Essential

Documentation









LICENCE



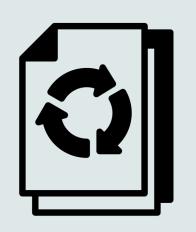
CODE_OF_CONDUCT.md

Used to give instructions to prospective collaborators on what you need help with, where they can suggest new ideas, what mechanism to use to propose actual changes.

Optional, but highly recommended

Documentation







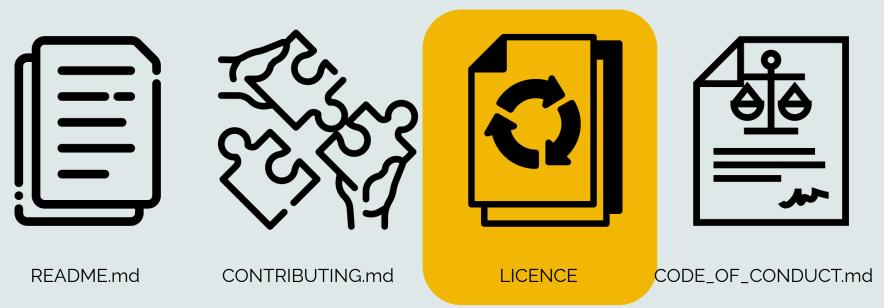
LICENCE

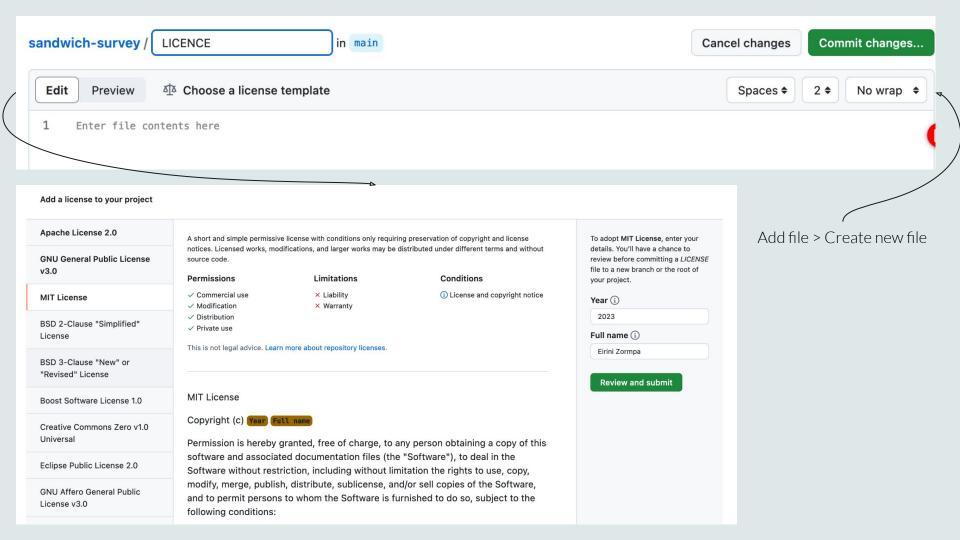
CODE_OF_CONDUCT.md

Used to set the project usage terms. Different projects require difference licence types; I would recommend OSI-approved for software, Creative Commons otherwise.

Essential

Documentation

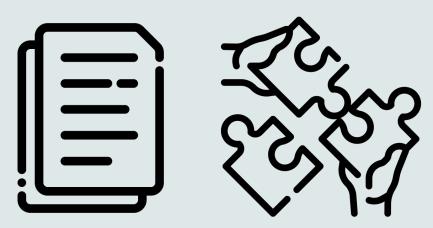




Used to describe the expected behaviour from collaborators and contributors. It should give examples of (un)desirable behaviours and the process for dealing with breaches.

Optional, but highly recommended

Documentation







LICENCE



CODE_OF_CONDUCT.md

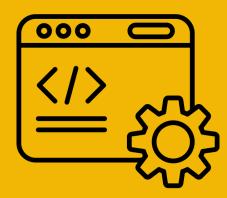
Acknowledging all contributions >>

GitHub will automatically track the contributors who have made commits to repository files. But not all contributors push code! Consider:

- project management
- Previewing pull requests
- • finding bugs
- **6** suggest improvements



Time to collaborate!



Exercise

- 1. Go to the <u>training repository</u> of the AIM RSF organisation
- 2. Fork the repository into your account
- 3. Navigate to the folder version control > hi.md
- 4. Follow the instructions in the document
- 5. Commit your change
- 6. Contribute the change to the upstream repository via a pull request. If you can see the option, add me (eirini-zormpa) or Rachael (RayStick) as reviewers.

Collaboration ideas 🧠 🔆



- Create documentation for your existing projects
- Publish one of your repositories on GitHub (if that is possible)
- Create an issue in a repository with a suggestion for improvement (or to fix a link, or correct typos)
- Comment on an issue that you could contribute to

See the HackMD for documentation templates and repository suggestions!

Open source



Open source software

"Open source software is software with source code that anyone can inspect, modify, and enhance."

Open source software development typically involves open sharing of work and collaboration.

Contributing to open source software can help you build your existing skills or develop new ones, advance your career, and meet people with similar interests

Quote from the "What is open source" blog post on opensource.com; see the Open Source Definition on the Open Source Initiative website; some content from the Open Source Software chapter of The Turing Way



Any final questions?



Scriberia