THE UNIVERSITY *of* EDINBURGH

Using Git and GitHub for Team Collaboration

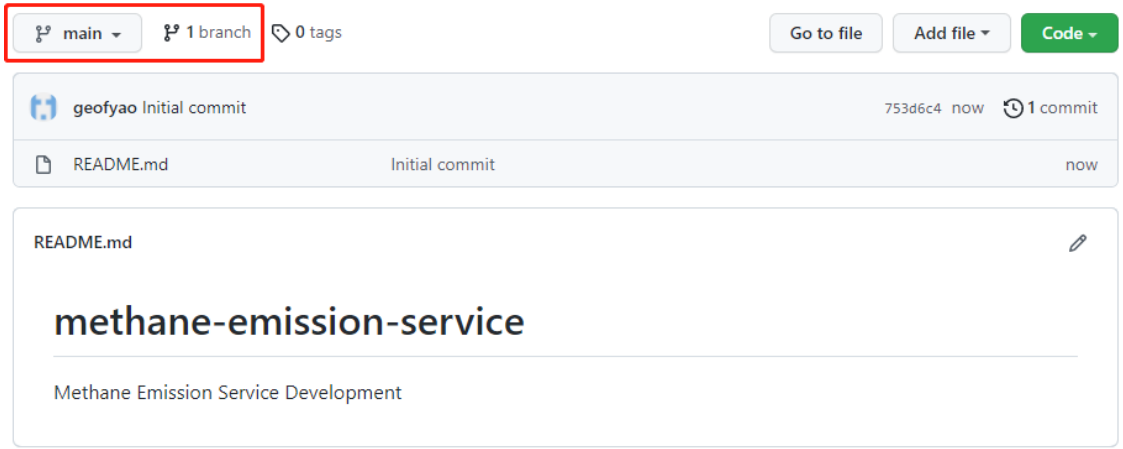
Fei Yao
28/02/2022

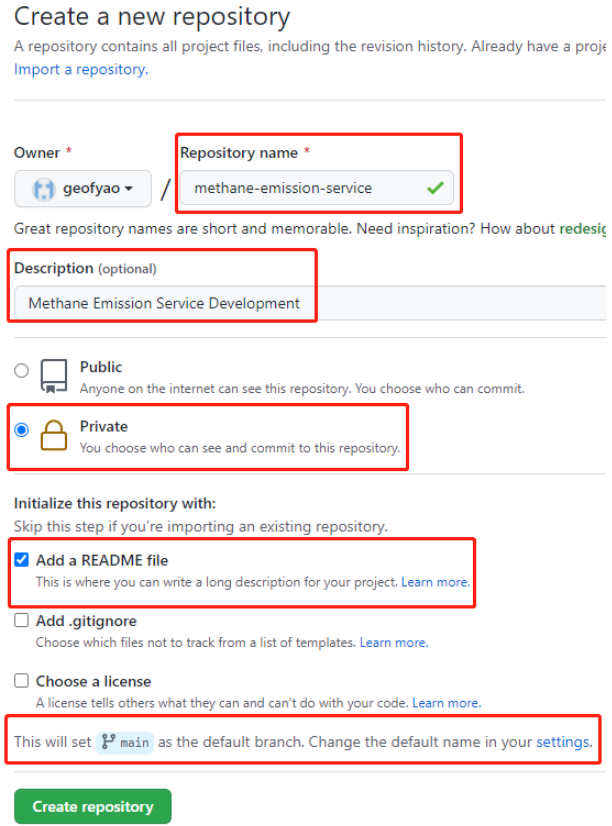
Git and GitHub

- Git and GitHub are not the same thing.
  - Git is an open-source, version control tool created in 2005 by developers working on the Linux operating system;
  - GitHub is a company founded in 2008 that makes tools which integrated git.
- You do not need GitHub to use git, but you cannot use GitHub without using git.
- Alternatives to GitHub include GitLab, BitBucket, and etc. All of these are referred, in git-speak, as "remotes" that will make sharing code with others easier.

An example workflow

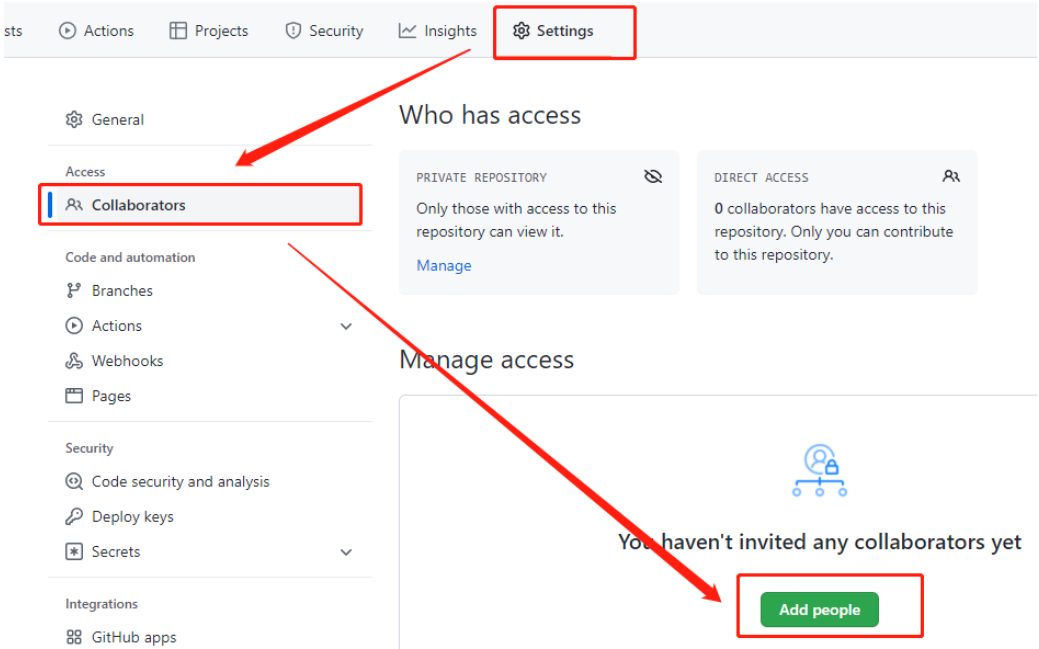
- The project leader creates a repository (repo).
  - Add a README file to set main branch as the default branch.
  - Can modify the README contents later on.
  - Add .gitignore and the licence later on.

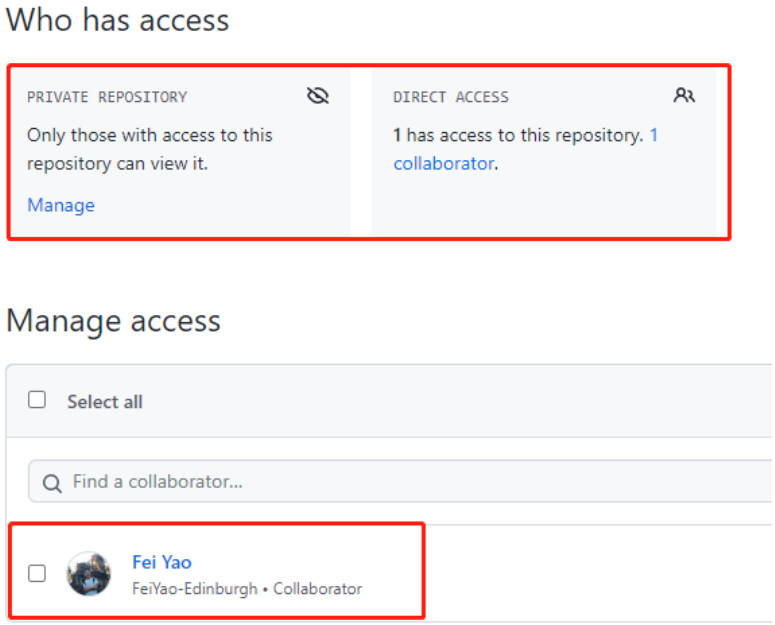




An example workflow

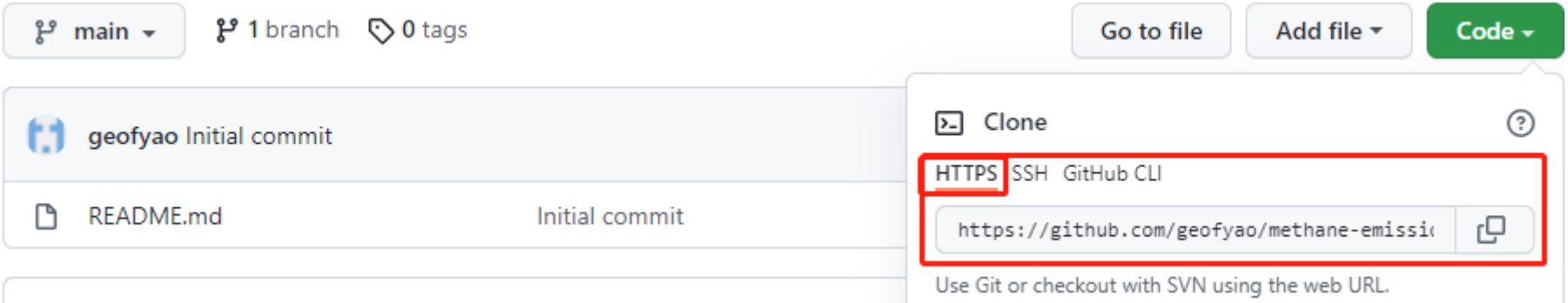
- The project leader adds collaborators to the repo.
  - The collaborators have push access to the repo.





An example workflow

- The collaborators (including the repo owner) clone the repo to their local machines.
  - As collaborators, they can type their own username and password when prompted to do so.
  - HTTPS requires username and personal access token (PAT).
  - SSH may require SSH keys (similar to JASMIN although not tested).
  - cd
  - git clone https://github.com/geofyao/methane-emission-service.git



An example workflow

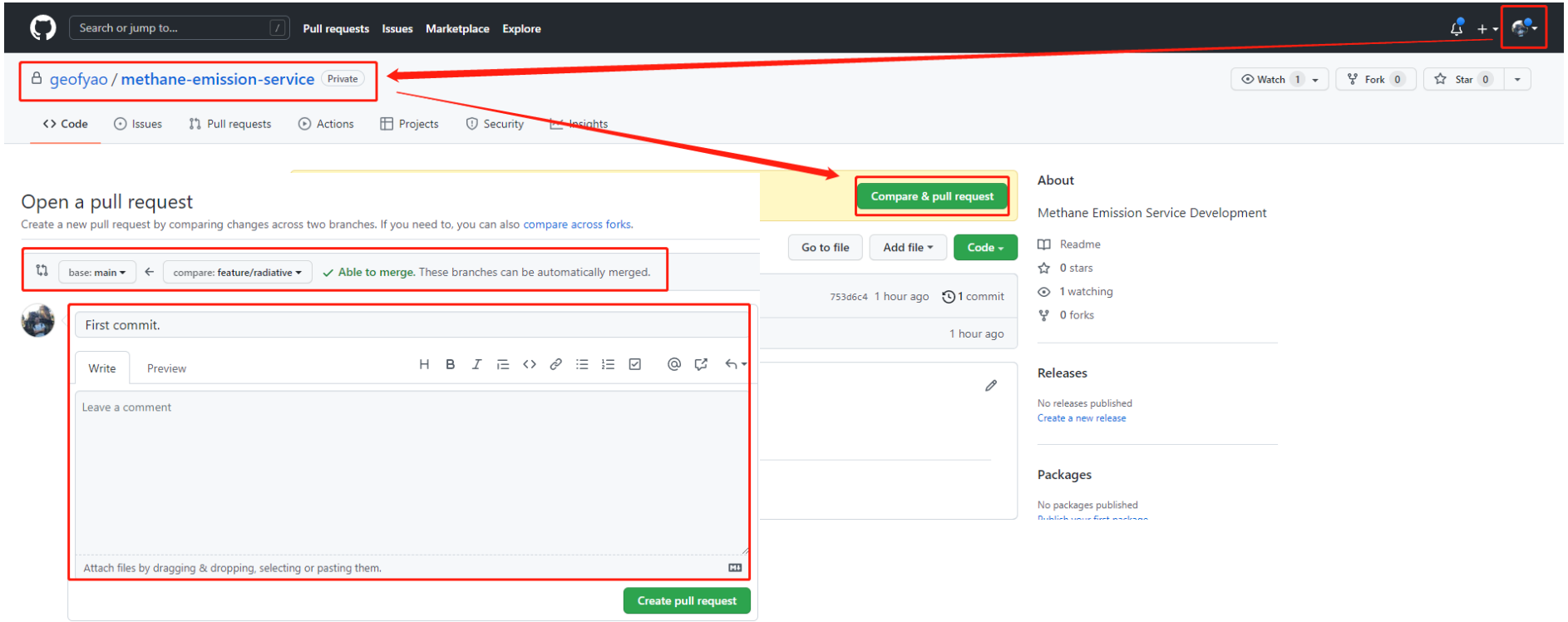
- The collaborators always create a new branch where they will develop codes **without affecting the main branch**.
  - cd methane-emission-service
  - git remote -v # show remote URL.
  - git branch # See that you are in the main branch.
  - git branch feature/radiative # Better to use a branch name indicating what the collaborators are working.
  - git checkout feature/radiative # Switch to the new branch.
  - # Can combine the above two lines of codes to "git checkout -b feature/radiative".
  - mkdir -p radiative # Now make some changes under the new branch.
  - cd radiative
  - touch README.md # Each subdir can have its own README.md.

The collaborators push their branch to the remote

- Track changes: add to the staging environment => make commits.
  - git status # Git has noted the changes but won't track it until we explicitly tells it to.
  - git add --all # Or can add one by one.
  - git status # Git has added the changes to the staging environment but not a commit although it is about to.
  - git commit -m 'First commit.' # Package all files that have been added to the staging environment to a commit.
  - git log # To see the commit.
  - git log origin/main..HEAD # See un-pushed commits if necessary.
  - # origin is an alias of the remote repo. HEAD can be thought as the current branch.
  - git pull origin feature/radiative # In case other commits are made to this branch during the period.
  - git push origin feature/radiative # Package all commits that you'd like to push to the remote.

An example workflow

- The collaborators make a pull request (PR) from their GitHub page.



An example workflow

- Other collaborators will be notified about the PR.

[geofyao/methane-emission-service] First commit. (PR #1)

From : "Fei Yao" <notifications@github.com> (Forward by noreply@github.com)

To : "geofyao/methane-emission-service" <methane-emission-service@noreply.github.com>

Cc : "Subscribed" <subscribed@noreply.github.com>

You can view, comment on, or merge this pull request online at:

https://github.com/geofyao/methane-emission-service/pull/1

Commit Summary

- 7284bf9 First commit.

File Changes (1 file)

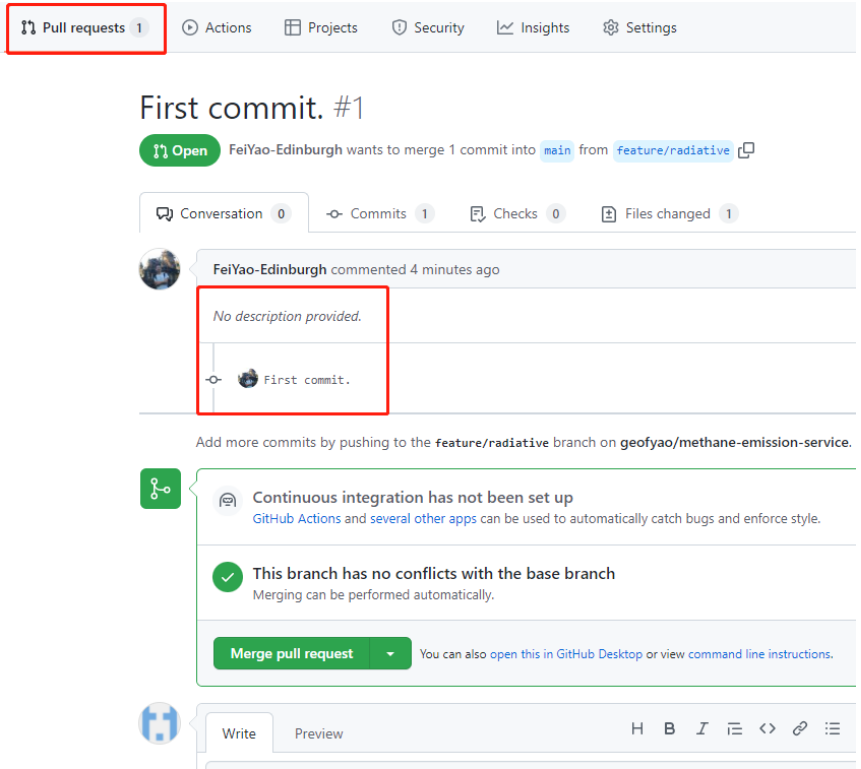
- **A** radiative/README.md (0)

Patch Links:

- https://github.com/geofyao/methane-emission-service/pull/1.patch
- https://github.com/geofyao/methane-emission-service/pull/1.diff

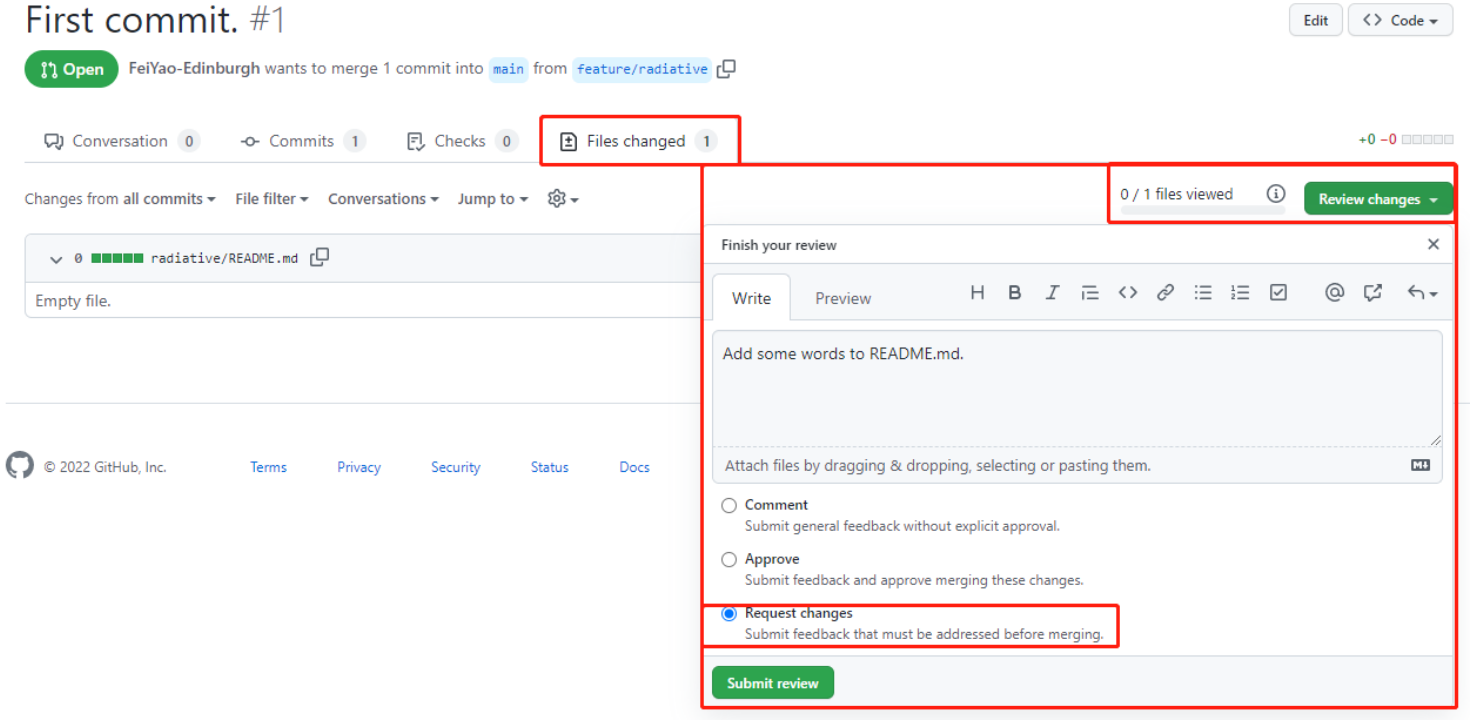
—

Reply to this email directly, view it on GitHub, or unsubscribe.
Triage notifications on the go with GitHub Mobile for iOS or Android.
You are receiving this because you are subscribed to this thread.



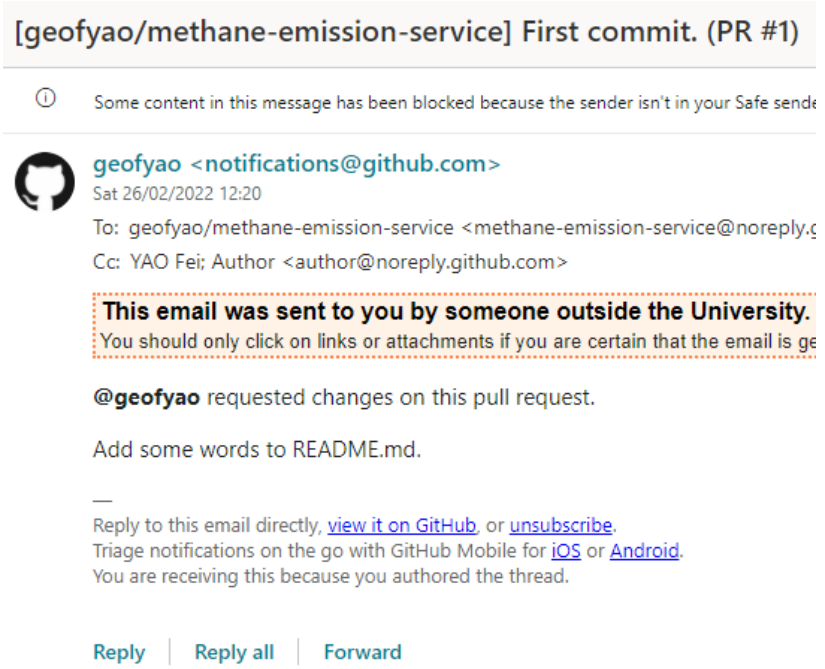
An example workflow

- Other collaborators review the PR and may make additional changes.



An example workflow

- The collaborators make additional changes per request and re-push.
  - vi README.md
  - git add --all
  - git commit -m 'Second commit.'
  - # git log origin.main..HEAD
  - git pull origin feature/radiative
  - git push origin feature/radiative



[geofyao/methane-emission-service] First commit. (PR #1)

ⓘ Some content in this message has been blocked because the sender isn't in your Safe sende

geofyao <notifications@github.com>
Sat 26/02/2022 12:20
To: geofyao/methane-emission-service <methane-emission-service@noreply.g
Cc: YAO Fei; Author <author@noreply.github.com>

This email was sent to you by someone outside the University.
You should only click on links or attachments if you are certain that the email is ge

@geofyao requested changes on this pull request.

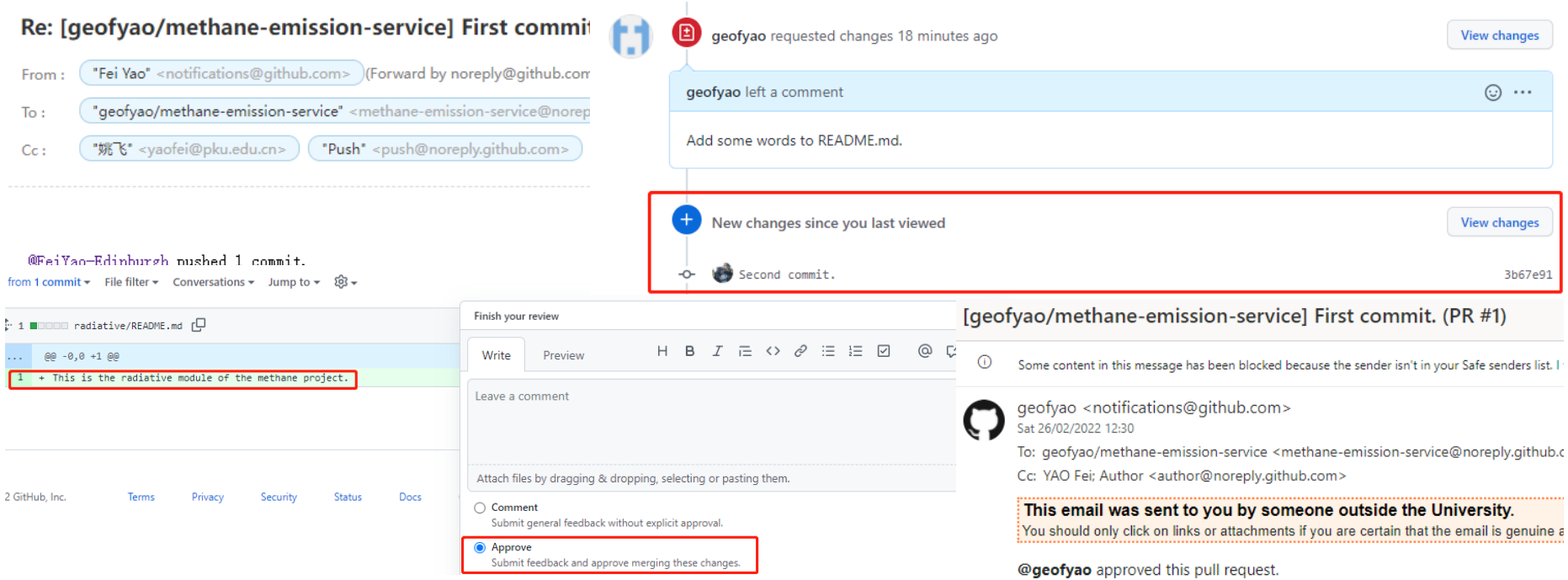
Add some words to README.md.

—
Reply to this email directly, view it on GitHub, or unsubscribe.
Triage notifications on the go with GitHub Mobile for iOS or Android.
You are receiving this because you authored the thread.

Reply | Reply all | Forward

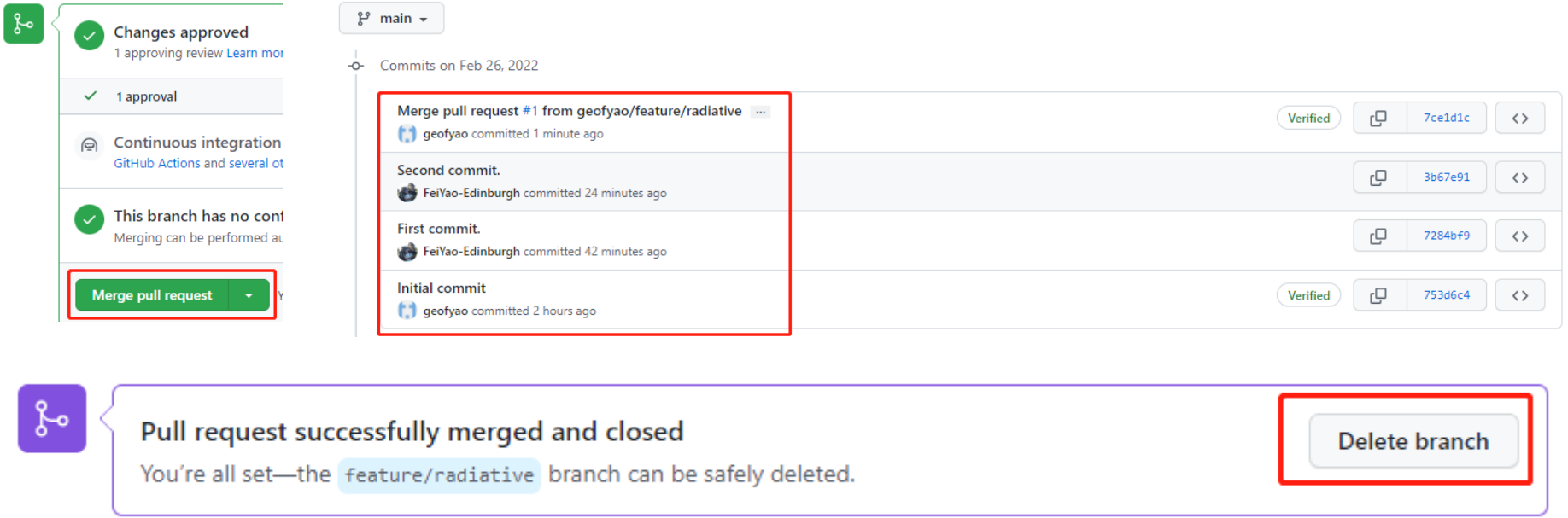
An example workflow

- Other collaborators will be notified again to remind them to review and may approve the additional changes made to the previous PR.



An example workflow

- All collaborators agree with the working branch to be merged to main branch and subsequently delete the working branch.

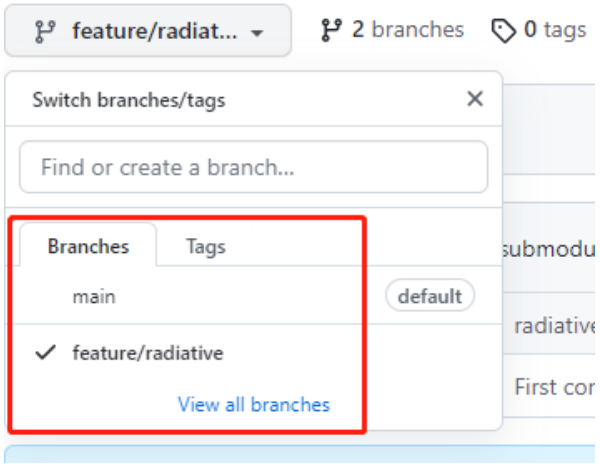


An example workflow

- The collaborators sync the remote changes back to their local machines and start next run. All collaborators behave similarly.
  - cd .. # After switch to main radiative no longer exists.
  - git checkout main # Always know where you are before running git-related codes.
  - git pull origin main # Keep update with remote. Or may git merge feature/radiative in main branch.
  - git log # To see completely identical as remote.
  - git branch -d feature/radiative # Remove already done branches just like the remote operation.
  - git checkout -b new-branch # New branch to work on.

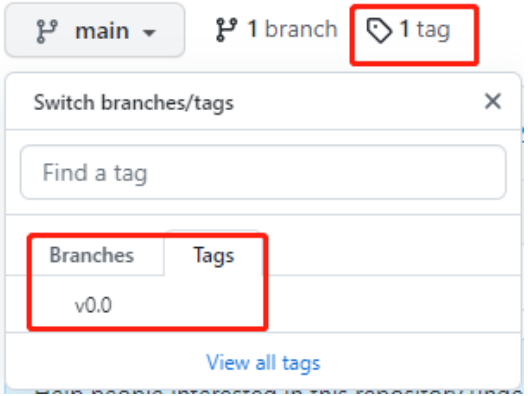
Recap

- A common git and GitHub collaboration workflow:
  - Fetch and merge changes from the remote.
  - Create a branch to work on a new project feature.
  - Develop the feature on the branch and commit the work.
  - Fetch and merge from remote again (in case new commits were made).
  - Push branch up to the remote for review.
- A safeguard against merge conflicts:
  - Collaborators work on separate subdirs.
  - They negotiate when have to work on common files.



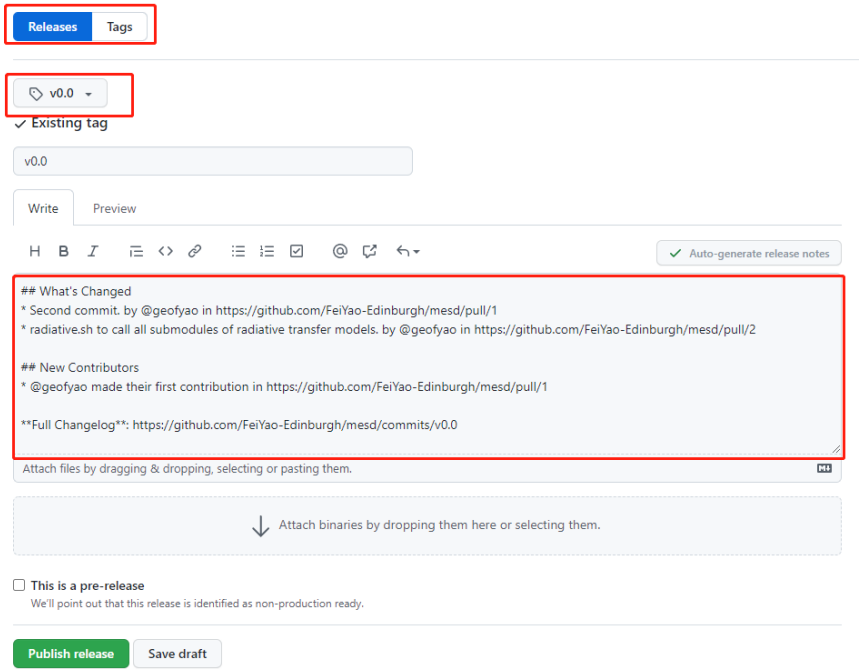
Tags: keeping track of project milestones

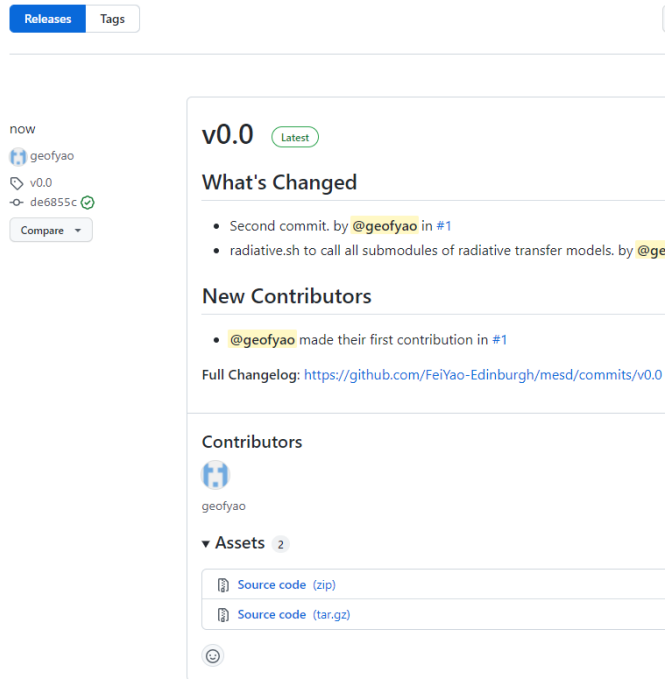
- A tag is similar to a branch; it's simply a lightweight moveable pointer to one of the commits.
  - git tag --list
  - git tag -a v0.0 –m "The 0.0 version."
  - git show v0.0
  - git push origin v0.0 # Always push code first.
  - # Create and checkout a new branch "feature/xxx" at tag "v0.0".
  - git checkout -b feature/xxx v0.0
  - git tag -d v0.0 # Can delete unwanted local tags.
  - git push origin --delete v0.0 # Can delete unwanted remote tags.



Release

- Releases, based on tags, are deployable software iterations you can package and make available for a wider audience to download and use.



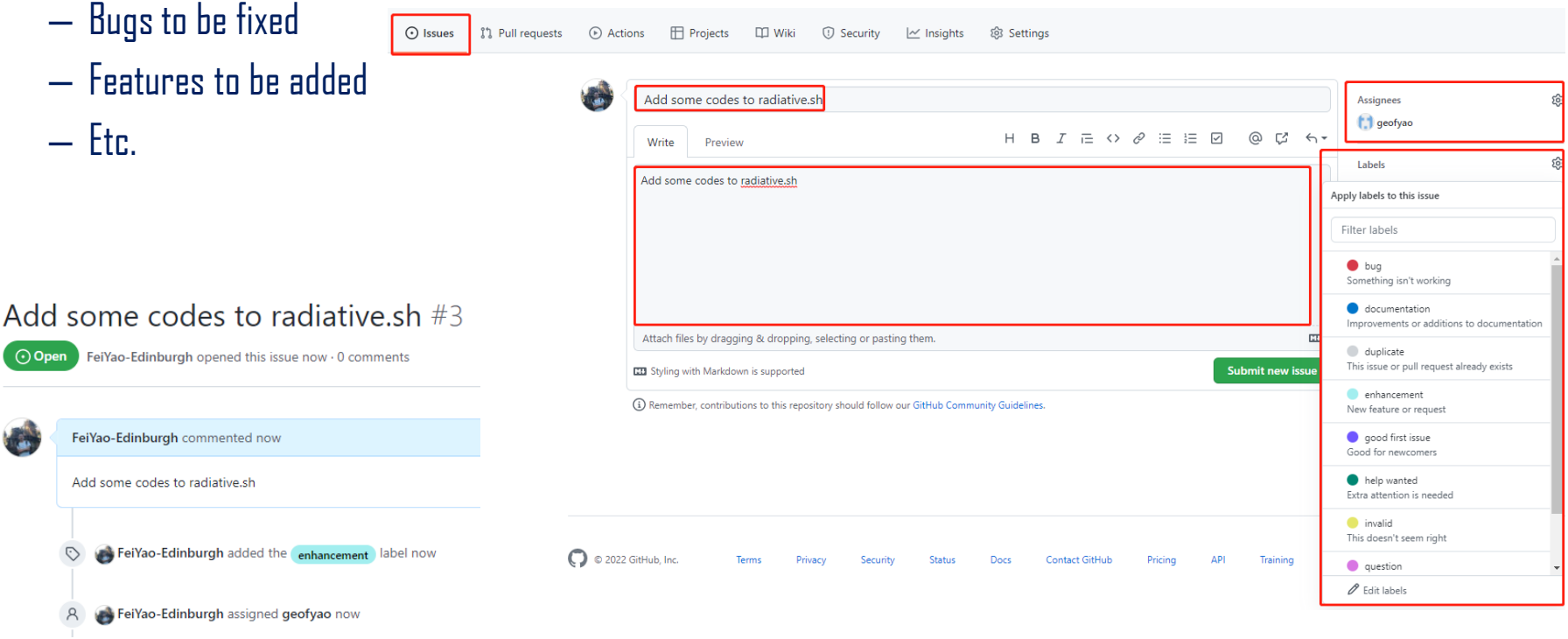


.gitignore: prevent unwanted file types

- Can reside in root and subdirs
- Prevent intermediate/backup files.
  - .ipynb_checkpoints
  - __pycache__
- Prevent large data files and so forth.
  - *.nc
  - *.nc4
  - Etc.
- Etc.

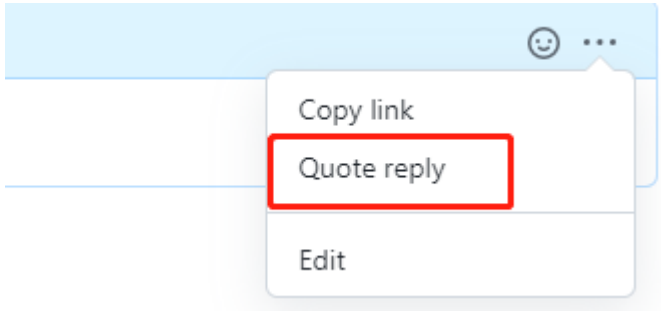
Issues

- Assign specific issues/tasks to specific collaborators:
  - Bugs to be fixed
  - Features to be added
  - Etc.



Issues

- Make scientific discussions
  - Simple but powerful markdown
  - Quote reply



Noting down some thoughts about mach

- Not sure if the current model can
- ML in GEE seems to require outlin
  classification
- Might need to retrain a new mod
  - This will require a new tra
  - We would need to outline
- This would produce a more usefu
- Allows further development of th
  proximity to infrastructure etc)
- Current training data set will look
  data that isn't seen in TROPOMI

*Training a new model:

- Labour intensive but possible short cuts
- If we use the old dataset of images (which I still have), upload them to GEE then outline them we don't have to search for new plumes
- We would need to manipulate the images to be uploaded (e.g., provide fake coordinates & times) and can then draw round the plumes using GEE
- Don't need 'no plume' examples as it'll just use the surrounding areas
- Possibly more errors involved because we need to decide the extent of a plume

3.2 Multi-band–single-pass (MBSP) retrieval

Our second method is a multi-band retrieval that estimates methane enhancements from differences between the band-11 and band-12 reflectances measured on a single satellite pass. For this method, the fractional change in reflectance is given by

$$\Delta R_{\mathrm{MBSP}} = \frac{cR_{12} - R_{11}}{R_{11}}, \tag{3}$$

where c is now determined by least-squares fitting of R_{12} against R_{11} across the scene. The fractional absorption model is then

$$m_{\mathrm{MBSP}}(\Delta\Omega) = \frac{T_{12}(\Omega+\Delta\Omega) - T_{12}(\Omega)}{T_{12}(\Omega)} - \frac{T_{11}(\Omega+\Delta\Omega) - T_{11}(\Omega)}{T_{11}(\Omega)}. \tag{4}$$

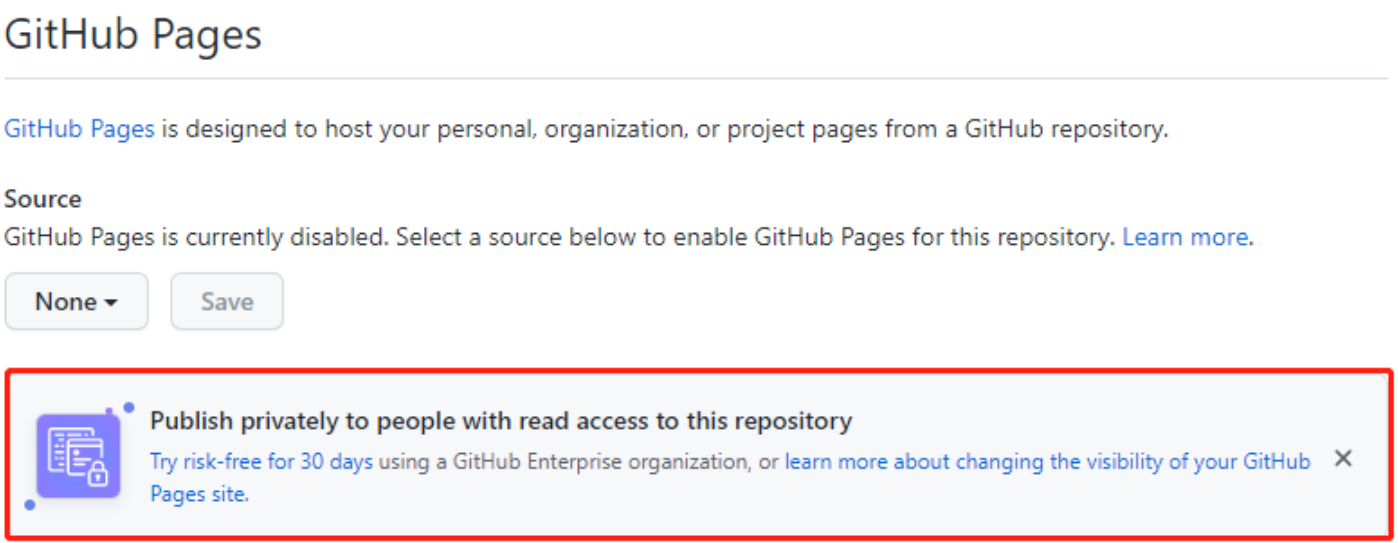
This approach relies on surface reflectance similarities between the two adjacent bands. The empirical scaling fac-

Licence

- The LICENSE, LICENSE.md, or LICENSE.txt file is often used in a repository to indicate how the contents of the repo may be used by others.
  - The license is best chosen from the get-go, even if for a repository that is not public.
  - Choosing a license that is in common use makes life easier for contributors and users, because they are more likely to already be familiar with the license and don't have to wade through a bunch of jargon to decide if they're ok with it.
  - People who incorporate General Public License (GPL'd) software into their own software must make their software also open under the GPL license; most other open licenses do not require this.
  - People who are not lawyers should not try to write licenses from scratch.

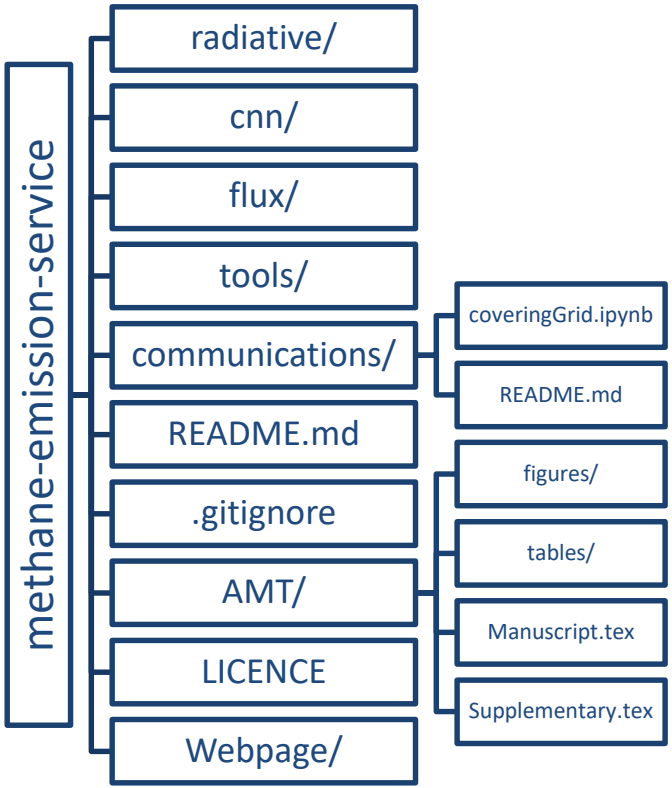
GitHub Pages

- GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.
  - Public versus private



The organization of the project

- A good top-design can make life easier.
- Can link with Overleaf:
  - Link the entire project to Overleaf.
  - Link the paper subdir (e.g. AMT/) to Overleaf. This will involve some further git commands.
  - Either way, we place all codes including Latex scripts within one place!



Reference

- An Intro to Git and GitHub for Beginners (Tutorial)
  - https://product.hubspot.com/blog/git-and-github-tutorial-for-beginners
- Version Control with Git
  - https://swcarpentry.github.io/git-novice/
- Using Git Tags To Version Coding Tutorials
  - https://medium.com/@emmabostian/using-git-tags-to-version-coding-tutorials-cf9ff28fad4f
- Git and GitHub cheat sheet
  - https://education.github.com/git-cheat-sheet-education.pdf

Additional slides

- Initialize a repo from local machines and link it to remote.

Initialize a repo from local machines and link it to remote

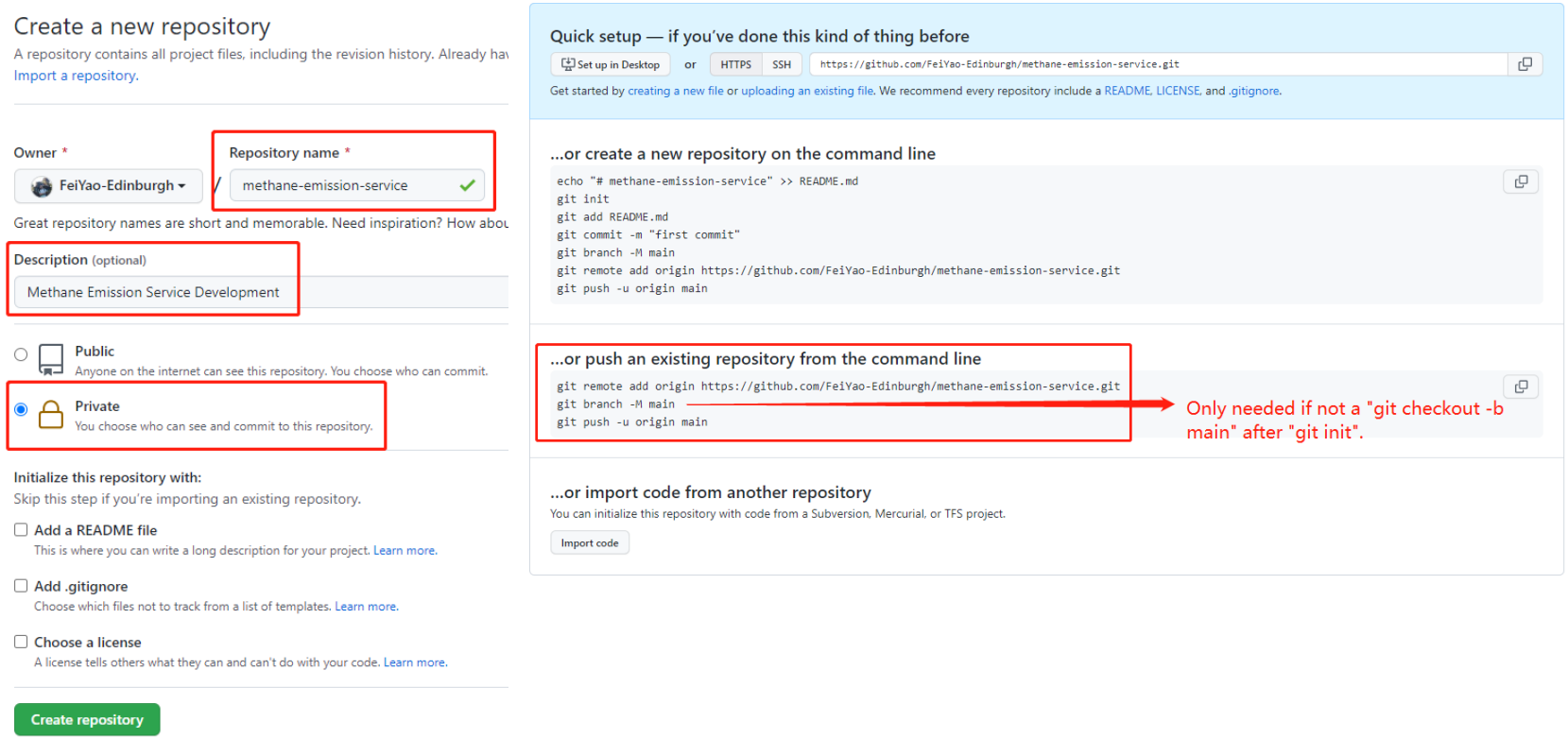
- Create a local git repository.
  - cd
  - mkdir methane-emission-service # methane emission service development
  - cd methane-emission-service
  - git init
  - git checkout -b main # primary branch name

Initialize a repo from local machines and link it to remote

- Add a new file (README.md) to the repo.
  - touch README.md
  - git status # git has noted the file but won't track it unless we explicitly add it in a commit.
- Add the file to the staging environment.
  - git add README.md
  - git status # git has added the file to a staging environment but not a commit although it's about to.
- Package all files added to the staging environment to a commit.
  - git commit -m 'Initial commit.' # leaving a clear explanation of your changes will be extremely helpful for future programmers (perhaps future you!)
  - git log # see that the commit has been added to the log

Initialize a repo from local machines and link it to remote

- Create a remote repo.



Initialize a repo from local machines and link it to remote

- Link local and remote repo.
  - git remote add origin https://github.com/FeiYao-Edinburgh/methane-emission-service.git
  - git push origin main # origin is an alias of remote repo!
  - git remote -v