Github Practice:

V2.0.1 Feb,2019

1. Go over the github tutorials online. <https://guides.github.com/activities/hello-world/>
2. Register an account and try the hello-world tutorial. Use an email you check regularly, since you will be notified via email about any updates.

All the above is web-based, now try the command line tool.

1. Install git for Ubuntu, sudo apt install git
2. Please find out how to do the following tasks. I’ll leave it to your guys to explore.
   1. Access my repository online and **fork** my repository : <https://github.com/taoluwork/hello-world>

< this is a testing repository, don’t be afraid to try out things>

* 1. **Clone** your own fork to local disk
  2. Open your local folder with **vscode** (please install if you don’t have this).
  3. **Edit** the README file, whatever you want.
  4. **Add** a file with your name, put whatever in it.
  5. **Commit** your changes to local disk file. You can either use git commit/git add or use build-in function of vscode.
  6. **Push** your changes to your online repository by git push
  7. Generate a **pull request** to push your changes to my repository, use web this step.
  8. I’ll be sent an email to review the code and **merge** the PR.

Later, I’ll do some modification on my end, and you should be able to pull the repository to sync with me. ( You will get emails from github)

Touch the real project, use caution from here.

1. Fork my repository <https://github.com/taoluwork/Eth_testing_sys>

this repository contains design descriptions and client\_description, **please have a look at this and let me know if you have any questions.**

1. Fork my repository <https://github.com/taoluwork/gitLab>

This contains the real design files. We are currently working on the bcai\_v2.0. Others could serve as examples to learn.

Fork is essential in open-source project because you keep an identical copy to edit but not influent the original one. This is always the case you try to work on others project without having to notify them. The bad thing is if there is conflict, you cannot merge PR. You can try it here.

In summary, 3 copies of project is kept.

A. Original copy: Project creator own and manage it. You can change by start PR.

B. Your fork: You own and control it, but it’s public others can see and fork.

C. Your disk copy: it’s like temp version you are developing, not ready to publish.

Git operations:

Make changes:

Git commit / add : lock your editing changes to your disk copy.

Git push: sync your disk changes to your online repository (your fork)

On web start a PR: sync your changes to original copy.

Sync with others:

On web start a PR: compare original with your fork.

Git pull: update your disk copy to your fork.

Thanks.

Tao