What is GitHub?

After working in programming for a bit of time you might find yourself hearing the term GitHub being thrown around. Looking into it afterwards can be daunting and learning about how it works and why developers love using it can often feel like an information overload.

GitHub at its core is a host. It is a space for all sorts of developers to upload their code for everyone to see it. It is most commonly used for developers to work together with one another on a specific program and version control. For you to understand exactly how GitHub works you must understand two key principles:

- Git
- Version control

Git is an open-source version control system. It allows developers to control the different versions and commit changes using command line. For newcomers to version control software it can be extremely difficult to learn at first, which is why most companies opt to using GitHub instead.

Version control allows people track the history of a program. This works through two methods: branching and merging.

- **Branching:** branching allows a developer to clone a program so they can edit the code for themselves without affecting the original program.
- **Merging:** merging allows the developer to merge the original code and their branch together. This process can be reverted if necessary.

After understanding these two key principles behind GitHub we can explain how it works using these new terms! GitHub is a user-friendly interface which allows users to utilize Git without learning the command line codes for it. It allows users to branch and merge different programs to edit and contribute to a project!
There are many new terms you must learn when first using GitHub. Below is a list of words that you will very quick run into and their definition:

**Command-line:** a text-interface use to process commands.

**Branching:** creating copies of a program to work in parallel

**Merging:** bringing together two different copies of a program together

**Repository:** name and location of all the project files used in a program. It is often abbreviated as repo and also contains the entire history of changes.

**Commit:** uploading a file to a branch

**Branch:** a duplicate of the main program which can be edited

**master-branch:** the original program, where the final program is located

**Pull request:** a request to edit a branch operated by another person

**Push:** commits local changes to the repository

**Fork:** clones a repository into your profile, essentially creating a copy where you can edit everything

---

**FAU Courses**

CEN4010 – Principles of Software Engineering
CEN4214 – Software-Hardware Co-Design
CEN4910 – Software Engineering Project
EEE3300 – Electronics 1

**Sources**

[https://kinsta.com/knowledgebase/what-is-github/](https://kinsta.com/knowledgebase/what-is-github/)
[https://github.com/](https://github.com/)
[https://thenounproject.com/term/fork/60030/](https://thenounproject.com/term/fork/60030/)
[https://git-scm.com/](https://git-scm.com/)

---

For more information on the HSI program at FAU visit: [http://eng.fau.edu/research/hsi/](http://eng.fau.edu/research/hsi/)