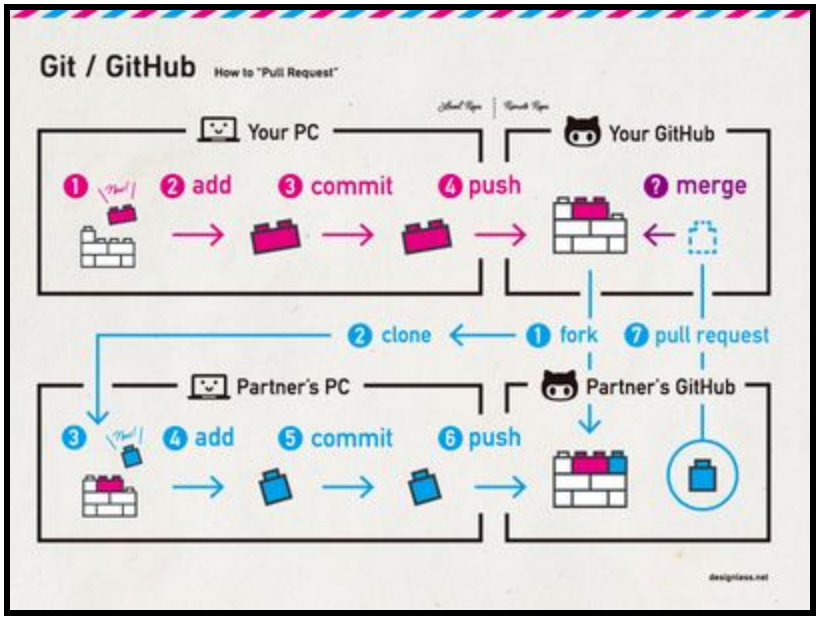
Github Technical Details

Github is an agile development service that allows users to collaborate on large projects as a team. It uses a branching system where each individual user can make changes to their own copy of a file without messing up the master file. The master file can only be changed if either ⅔ of the contributing users sign off on the document or the owner of the repository signs off on it. Github also allows private and public repositories so if a company doesn't want to open source their research or technology, they can make the repository private to only allow contributors to view and make changes to its contents. Overall Github is what most of the Software, Hardware, Electronic, and Systems Engineering companies use to manage their products both before release and after release.

There are other VCS (Version Control Systems) out there other than Github. Github is free so Matador Engineering can use it's free features to test drive it and see if they would like to upgrade to the paid version where they would be allowed to create private repositories.

- Git - a version control system based on cloud storage where users can access content anywhere at anytime.
- GitHub - a code hosting platform based on *Git* for version control and collaboration. It lets you and others work together on projects from anywhere.
- Repository - commonly called a "repo" is a *git* file system that keeps track of all changes to all files within a directory.

- Branch - a version of a *repository* that the user can change without changing the master *branch* that will eventually be delivered to the client.
- Clone - when a *repository* is cloned, it is copied to the users local computer so they can contribute to it.
- Checkout - when a user wants to see or make changes to a specific or someone else's branch, they can *checkout* the *branch* to maintain the structure of their own *branch*.
- Add - when a user adds a file to the directory or removes one.
- Commit - when a user adds a file to the local *branch,* but doesn't push it to the remote *repository*.
- Push - after a *commit* is made, the user will want other people to see their changes so they *push* it to the remote branch in the cloud.
- Pull request - when the SCRUM master has approved of changes that have been *pushed* to a *branch*, a *pull request* is opened to *merge* these changes to the master branch.
- Merge - When two *branches* have differences or things are being added or removed from them, you can *merge* the branches and choose what contents of each file will remain in the master branch.


Git / GitHub
How to "Pull Request"
Your PC
Your GitHub
1
2 add
3 commit
4 push
7 merge
2 clone
1 fork
7 pull request
Partner's PC
Partner's GitHub
3
4 add
5 commit
6 push

Hello World Guide

```
git clone [insert URL of your website]

git checkout -b [insert name of your branch you want to create]

*add some files to this directory on your computer*

git add *

git commit

git push origin [insert branch name]
```