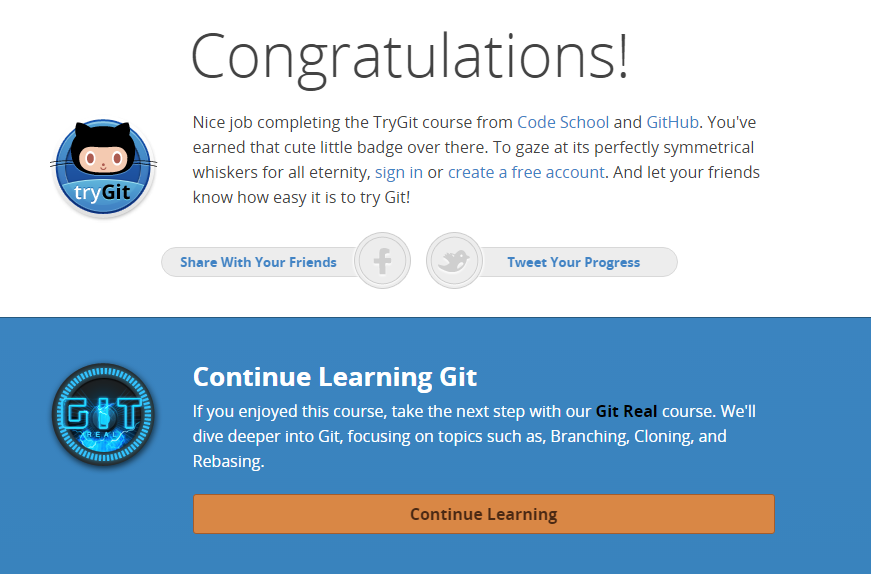
**GitHub**

GitHub is a platform that is used to store code for different sorts of projects. It is basically a repository that allows for developers to store and then collaborate on projects together. It also incorporates Git for version control which is what allows developers to work together without having conflicts with their code. It was originally released back in 2008 by Tom Preston-Werner, Chris Wanstrath, and PJ Hyett. The development began the year prior in 2007 which lead to a few beta releases prior to the main release. Other platforms with similar features include BitBucket, SourceForge, and GitLab. The main uses for GitHub would be to have proper documentation and control of source code, especially when working on a team. GitHub allows all members of the team contribute to the project while at the same time identifying and conflicts between versions and merges. GitHub also allows proper cloud storage of projects for future use.

**Part 4**



**Part 5**

Repository – Root location of where the code for the project is stored along with commit objects and references called heads.

Commit- Records changes to the local repository. This can either be done to staged (index) changes or directly from the workspace.

Push- Pushes or updates the local repository changes to the remote repository.

Branch- Separate copy of a repository usually checked out to do some work before committing, merging, and then deleting. Branch also provides commands to create, rename and delete branches.

Fork- This is a copy of a repository which is good to do if you would like to test out a new feature.

Merge- Allows the user to merge two different branches. This is usually done after commits have occurred and the master branch is trying to get all of the latest changes.

Clone- This basically copies the entire project into a new directory. This is usually a good way to jump into a project that is already up and running.

Pull- A combination of Git fetch and Git merge. The pull looks for any changes in the remote repository and then merges them into your own local repo.

Pull request- A pull request is a request sent by the user saying that the changes are complete and are ready to be merged with the main branch of the project.

**Part 7**

**Summary:** I used the fork feature to first make a second repository and then I checked out the repository and made the changes. I made the changes in a separate branch before merging and then deleting that branch. I then pushed all of my changes to the remote repository. Finally I submitted a pull request through GitHub for my changes. The actual steps are below.

1. Clicked on fork and made a repository in my name
2. Git clone project

--Waited a day because of class

1. Git remote add upstream “…course.git”
2. Git fetch upstream
3. Git status
4. Git checkout master
5. Git pull –rebase
6. Git checkout –b firstChange
7. Opened text editor and added my name
8. Git add README.md
9. Git status
10. Git commit –m “first commit”
11. Git checkout master
12. Git merge firstChange
13. Git branch –d firstChange
14. Git push
15. Used GitHub to assign a pull request