**Working** **with GitHub**

First off, a bit of an introduction to GitHub. GitHub happens to be one of the most effortless graphical interfaces for the Git programming language in the world. What is a Git you might be wondering? Well, a Git is a version control software which allows changes to a project without overwriting any part of the project.   
  
For example:

Say a team of programmers and yourself are working on updating or reprogramming a page of a website at the same time. When you make your changes you upload them back to the site and all is well; until one of the other team member’s uploads their changes, then all of your updates have been overwritten.  
 GitHub prevents this from happening, you and your team can upload revisions to the same page and GitHub will save multiple copies, later you can merge the changes together without losing any of the work. Addition to that, you can also revert to an earlier version at any time due to GitHub taking a “snapshot” of any change ever made.



Here is a project homepage, at the top in point **one** is:

- The project name  
  
- How many people are watching it  
  
- How many people have starred it (giving it a vote of confidence)  
  
- How many people have forked it, meaning to make changes of their own or to contribute in some way.

In point **two**, you’ll see:

* **Code:** The view you are on by default, showing every file inside the project.
* **Issues:** An effective and simple issue tracker, you and your team can report bugs, problems or even make requests for extra/new features.
* **Pulse:** Statistics about the project, all of the open and closed issues, here shows you how the activity of the project is coming along.
* **Graphs:** The timeline of commits, then a breakdown of commits by the individual contributor. You can see here the project activity in detail. Based on a series of key metrics including least to most active days for contributions and code frequency and so on.
* **Wiki:** A wiki for documenting the project in more extensive detail than a standard README file gives you.

In point **three** you’ll see:

* The number of commits to the branch & the number of branches.
* The number of releases
* The number of contributions

In point **four**, you’ll see:

* The branch picker and then below there’s a listing of the top level files inside the project.