Git for Windows

This tool will allow multiple developers to collaboratively create or modify a project package offline and easily merge with a master package online. Conversely, a clone of the master package can be copied to a developer's local workstation.

Download

Git for Windows can be downloaded by selecting 'On Google Code' at: http://msysgit.github.com/

The current featured beta as of this writing is **Git-1.8.1.2-preview20130201.exe**. Save this executable to your workstation and run. During the Git Setup prompts, on prompt for **'Adjusting your PATH environment'**, select **'Run Git from the Windows Command Prompt'** option, as shown below:

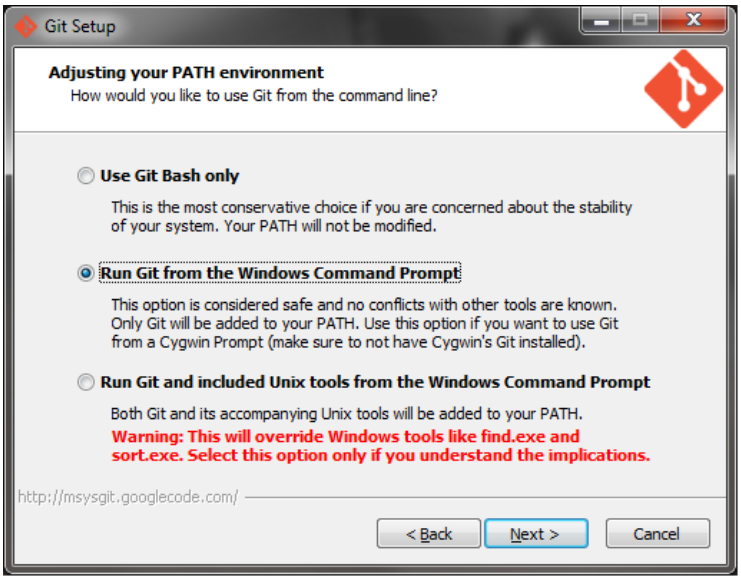

Figure 1: Adjusting your PATH environment prompt

Use OpenSSH, as shown below:

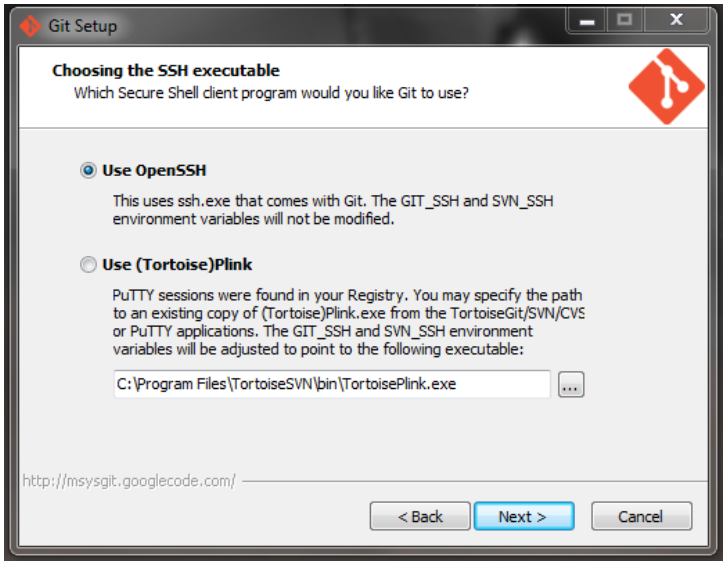

Figure 2: Choosing the SSH executable

Configure the line ending conversion, as such:

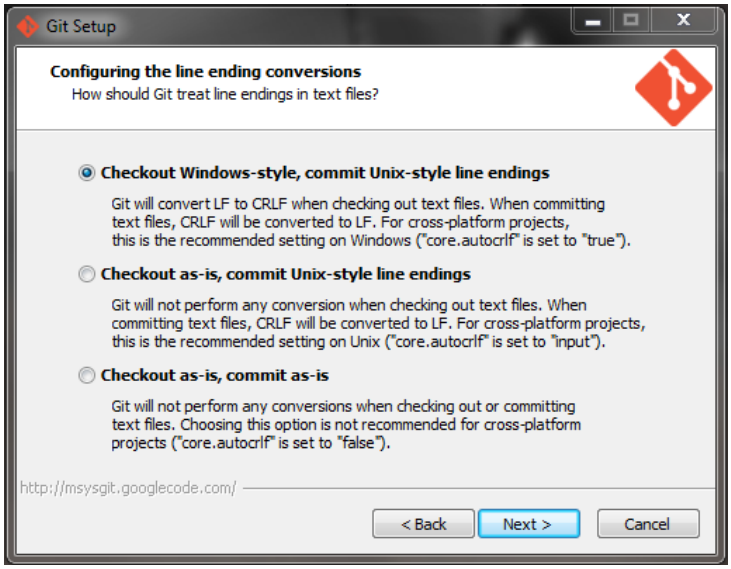

Figure 3: Configuring the line ending conversions

After the installer has successfully finished, we will configure Git for use in creating and modifying our project package.

Configure Git

Now we will configure Git with the use of **Git Bash**. There should be a shortcut on your desktop, but if not, it can be found in the Git directory that you just installed, for example, 'C:\Program Files (x86)\Git'. Access this shortcut.

Figure 4: Git Bash Shortcut

You should see a command-line that looks like this:

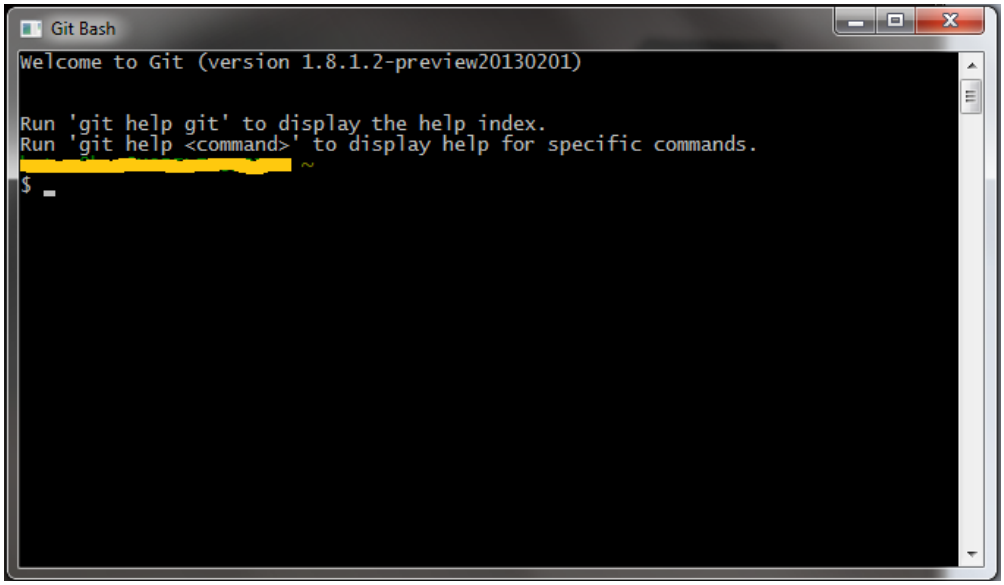

Figure 5: Git Bash Command-line tool

Now we want to configure the username and useremail:

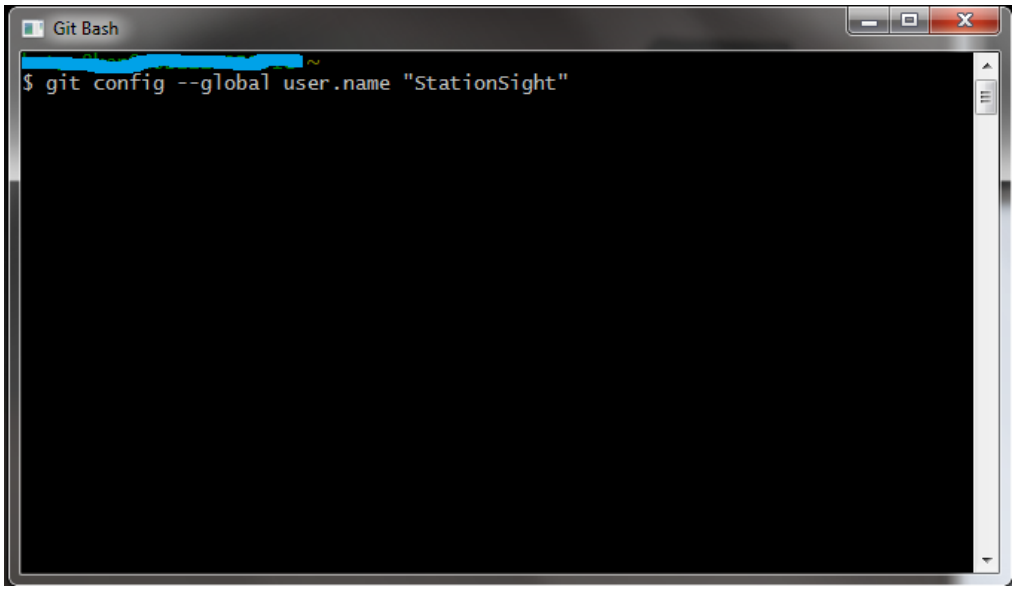

Figure 6: Set user.name

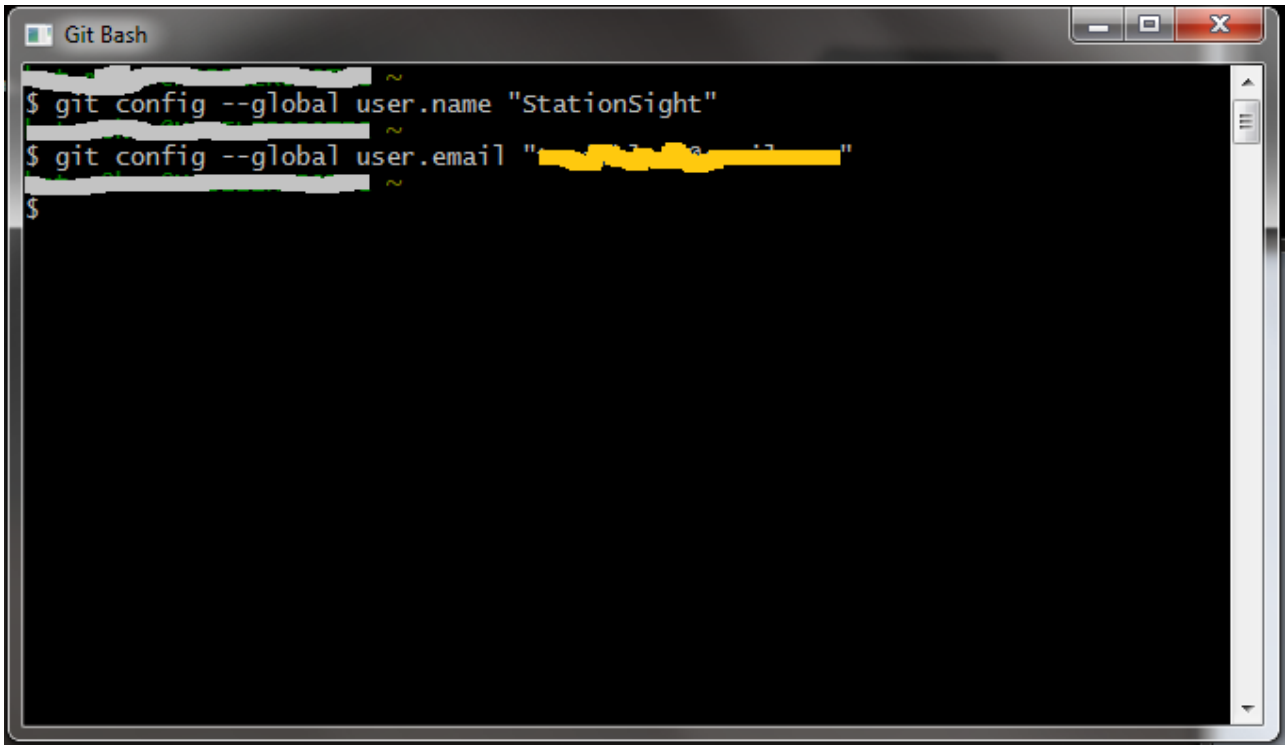

Figure 7: Set user.email

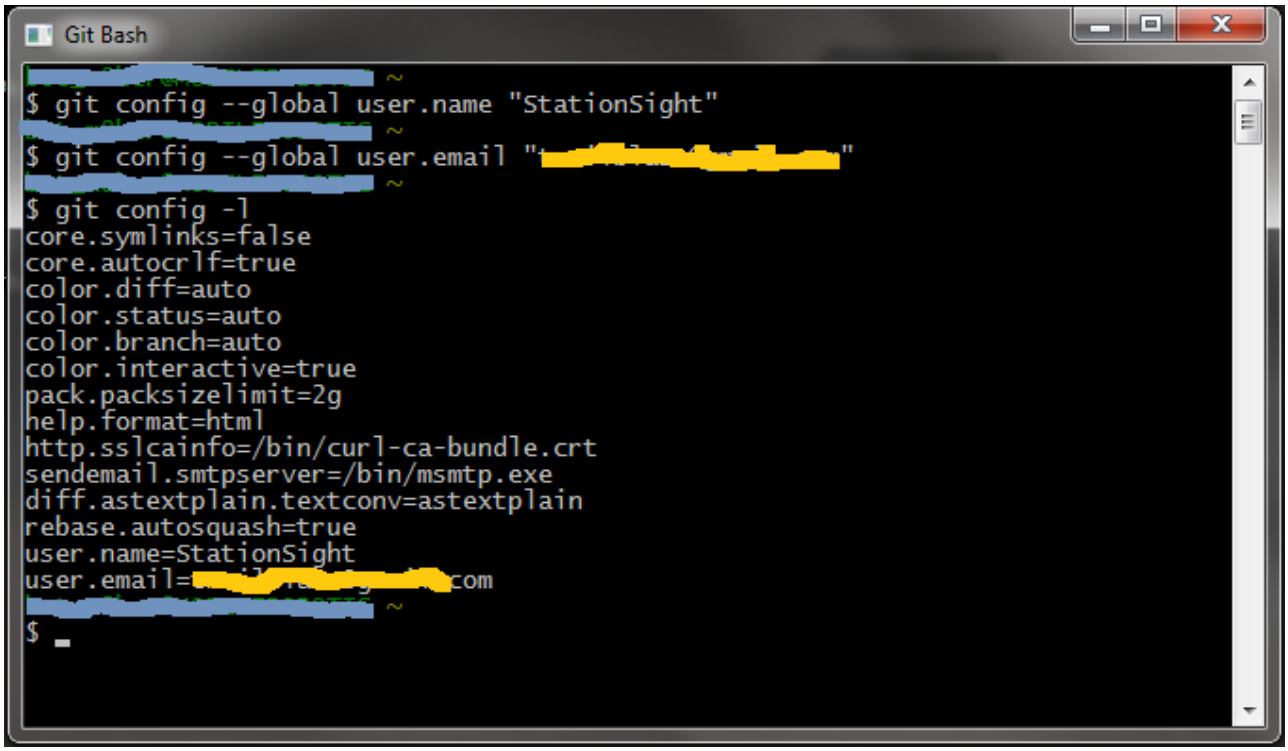

Figure 8: Configuration list

Clone a Repository

If there exist a repository, then we can clone the repository onto our local workstations. To accomplish this, we must use the command:

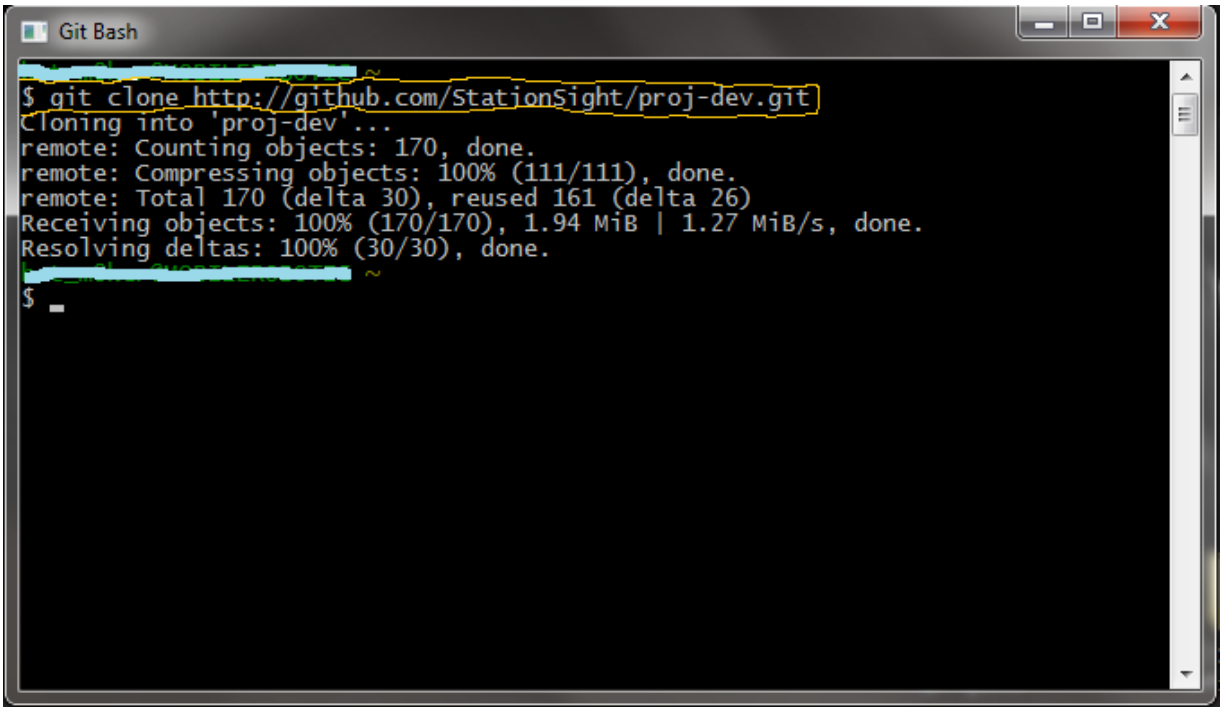

Figure 9: Successful clone of repository at github

The project should now be local to your workstation. You will likely find the project at **'C:\Users\<your_username>\<repository_name>**. Now you will be able to make advances with the code locally. Later, you can merge your improvements into the master package by uploading to the repository. Now, you should make some modifications to the project, such as: adding a file, modify a file, or remove a file, so that we can upload these changes to the repository.

Upload to Repository

If you have made changes to the local directory holding the project package, then we should be able to see a listing of these changes by calling the Git status command. You will need to be in the local project directory when calling the Git status command.

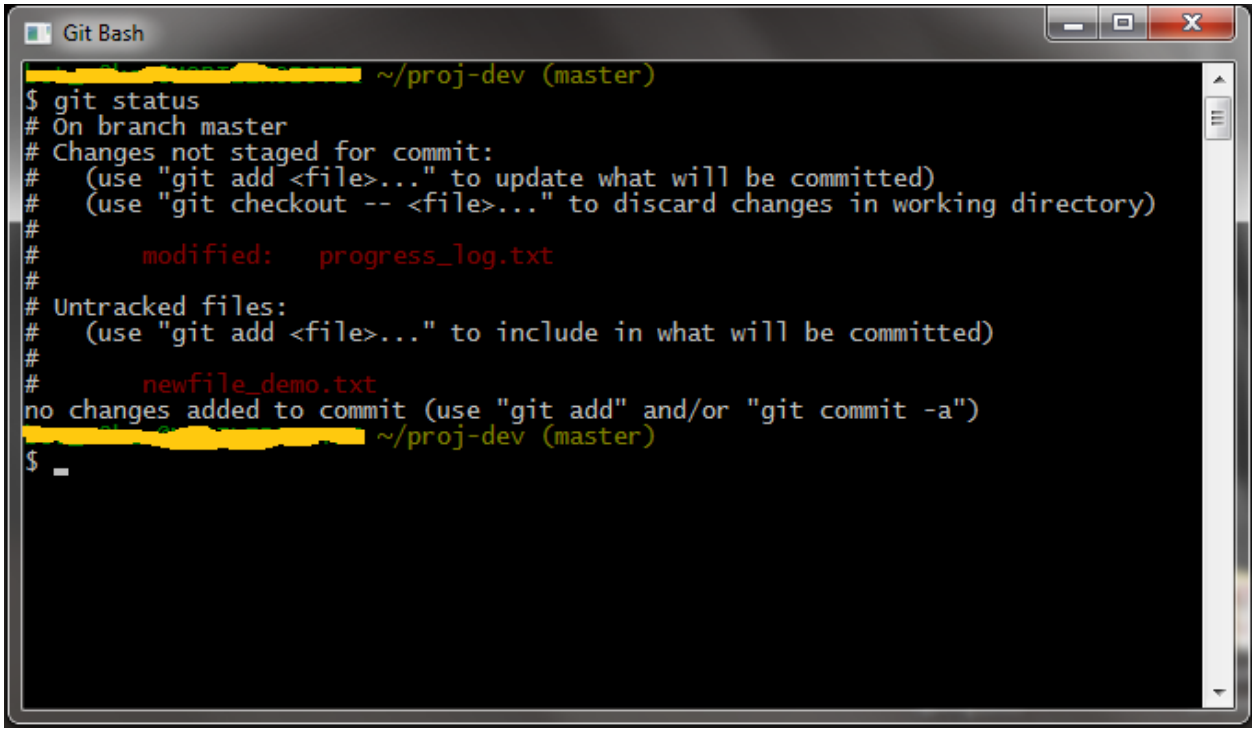

Figure 10: Git Status Command

Stage and Commit

There are three main sections, the working directory, staging area, and Git directory (repository).

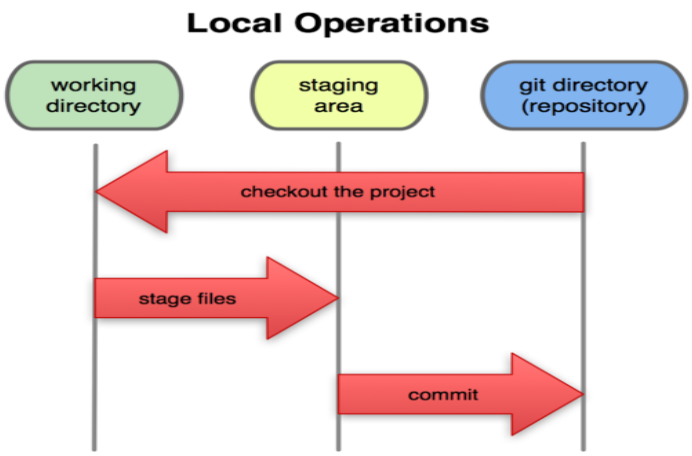

Figure 11: Project Sections

The modifications and additions that were made need to be placed in the staging area. We can accomplish this with the Git Add command.

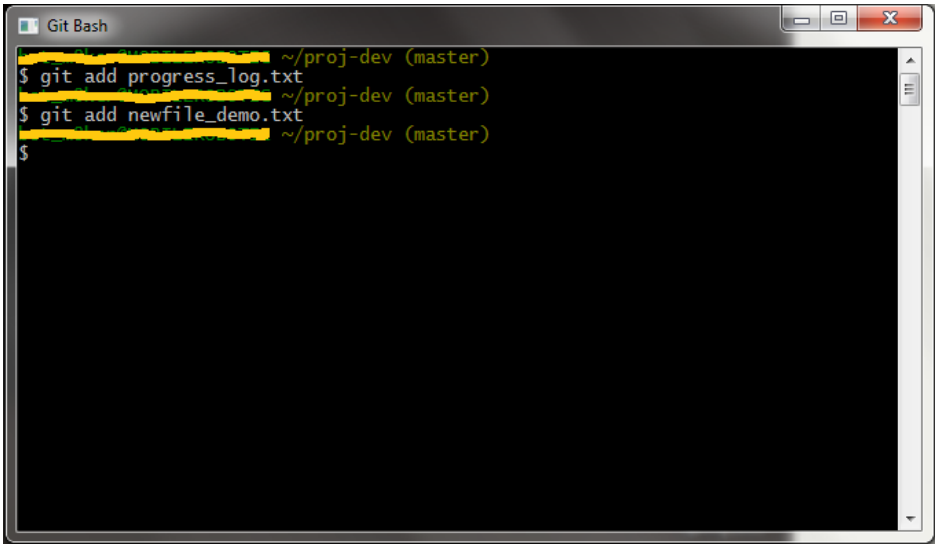

Figure 12: Git Add Command

Similarly, there is a Git Remove command that looks like this:

```
$ git rm <filename>
```

This will remove the specified file from the next commit, but not from previous snapshots. The files are now in the staging area awaiting a commit. To commit we use the Git Commit command that looks like this:

```
$ git commit -m "comment"
```

This will commit all files in the staging area into the next snapshot. The comment string will identify the next snapshot.

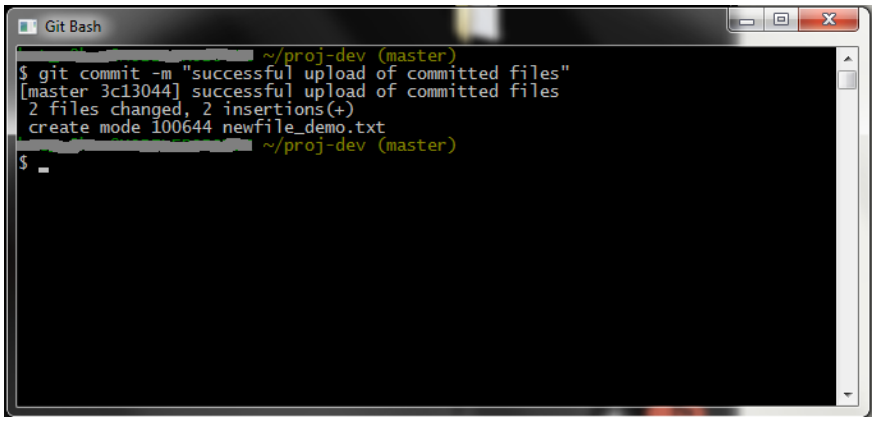

Figure 13: Git Commit command

Finally, we can upload all committed files to the master snapshot. We use the Git Push command to upload all committed files and it looks like this:

```
$ git push origin
```

The commits on your local workstations will be pushed to the online repository at 'origin', which is a variable containing the address of the repository from which the project was initially cloned. A successful upload will appear as follows:

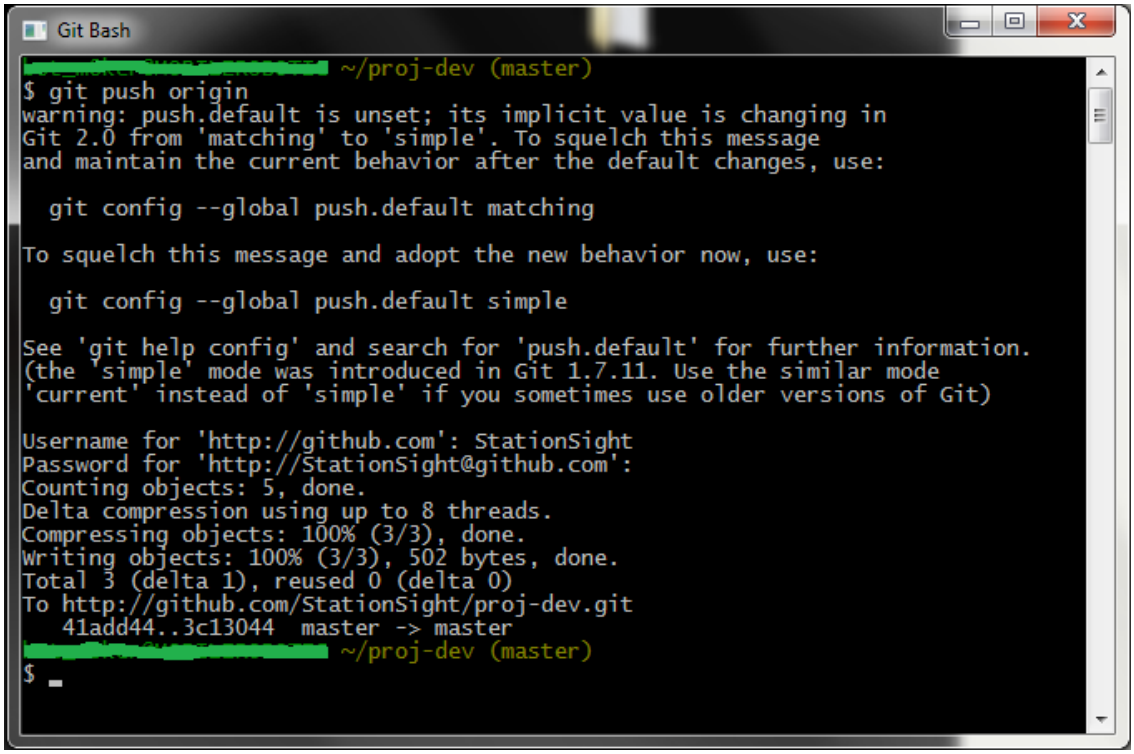

Figure 14: Successful Upload

Notice that a username and password will be required to complete the upload. Each upload will create a new snapshot in the online repository. The following image should aid you in visualizing the way the git repository manages version control.

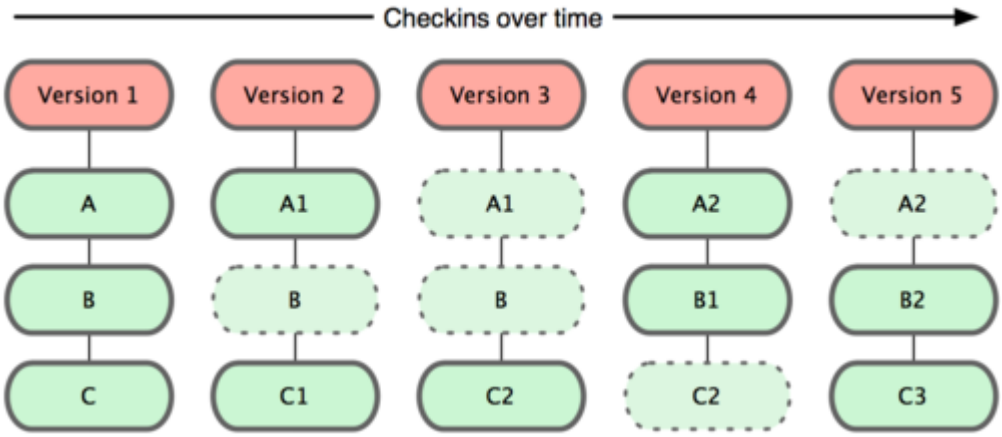

Figure 15: Git snapshots of the project over time

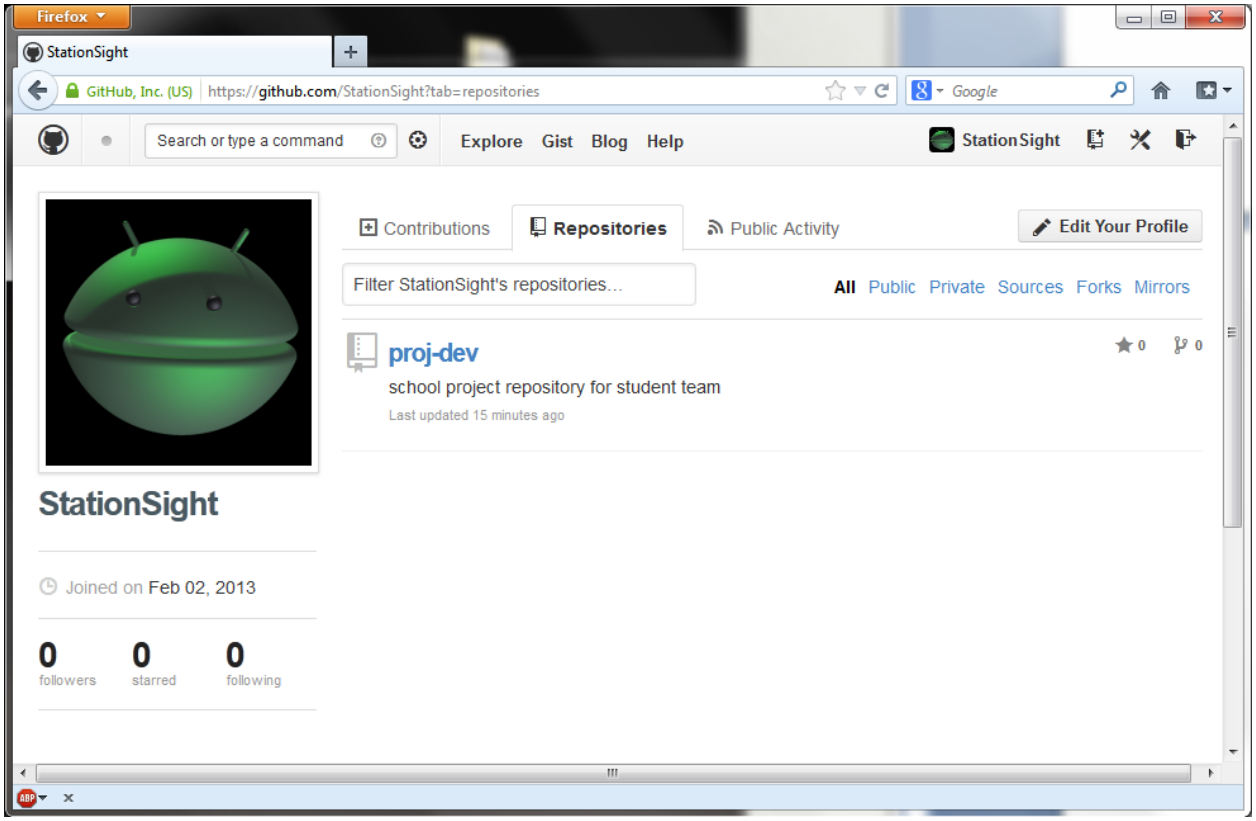

Figure 16: Screenshot of our github repository