Git

Now you need to install Git – a version control system to manage your software development conducted by your group. Follow the following instructions to install and use git:

Install git
```
$ yum install git
```

Create a local git repository
```
$ cd path_to_directory/
$ git init
```

At first, any files in this directory will be untracked by git, so you will have to add them
```
$ git add .
```

Now that you have your files tracked, you can make your first commit
```
$ git commit -m 'Initial commit'
```

You can use the `git log` command to look at the commit history.

Now that you have set up the local git repository, you can create a new remote repository in GitHub so that your team members can collaborate on this project. If you don't already have one, create a GitHub profile.

Create a new git repository in the GitHub main page and call it `COMP-3500`.
The new GitHub repository will be empty when it is initialized, so now you have to push up your existing repository so that it will be available on GitHub:

In your local git directory, add the newly created GitHub repository's remote:
```
$ git remote add origin https://github.com/username/COMP-3500.git
```

Be sure to replace `'username'` above with your github username.

Now that your local repository knows where it's remote location is, you can push up all of the original files:
```
$ git push -u origin master
```

The command above specifies that you want to push up your local changes to the remote we named `'origin'`, to the git branch named `'master'`.

Note:
All repositories start with a `master` branch. A branch is useful for making drastic changes to your codebase while maintaining a stable "official" master branch, especially while working on a project with others. A branch gives you a new place to commit and push changes without affecting `master`.

Once you are happy with your code changes you can use $ `git merge branch_name` to merge changes from the current branch, and into another.

You can create new branches at any time with $ `git branch branch_name`
Switching between branches is as easy as $ `git checkout branch_name`

You now have your git repository hosted on GitHub. You can now refresh the repository page in GitHub, and you should see all of your files present.

If you chose to make the repository public, anyone can view the repository, but only you can commit and push changes. You'll want to add your teammates as collaborators by navigating in GitHub to **Settings > Collaborators**, and adding your group members' GitHub usernames.