

git Instructions

Courtesy of Rachel ...

Log into gaul and go to the location where the repo is going to be held.
For our group it was:

```
cd ../../gaul/special/cs4480/cs4480-02
```

Create a directory there. Our group did:

```
mkdir IntrudersRun
```

Then change the permissions on that group:

```
chmod g+rxw IntrudersRun
```

Then you need to go into the directory:

```
cd IntrudersRun
```

and create the git

```
git init --bare --shared=true
```

Now there is a git!

Now from your personal system or just on the hard drive of the macs in the lab
(not logged into gaul) log into terminal.

Go to the place where you want the repo stored. I went to:

```
cd school/cs4480
```

And then clone the git onto the hard drive, my command was this:

```
git clone ssh://rgrimard@gaul.csd.uwo.ca/gaul/special/cs4480/cs4480-02/IntrudersRun
```

It will ask you for your gaul password.

Now you will have the directory you created on your hard drive. I know

have school/cs4480/IntrudersRun on my personal machine.

I created a test.txt on my personal machine and saved it in my IntrudersRun (its the repo) directory. Now you need to put the file in the repo.

In terminal, in the IntrudersRun folder on your personal machine do the following (only one person must do these exact steps):

```
git add .
```

this adds all files

```
git commit -m "commit message"
```

the -m makes it so you don't have to open a text file for a commit message

```
git push origin master
```

this push only needs to be done this once. After this you can just do a plain git push

Thats it. The repo is created and synced.

To update the files on your personal system do:

```
git pull
```

All other members of the group just need to do the git clone, and when they have things to add to the repo do a

```
git add .
```

```
git commit -m "message"
```

```
git push
```

These two files need to be created in the root folder of your git repo to stop the merge issues with the project file. I haven't had my team test it yet, but I think it will be fine.

`.gitignore`

```
# xcode noise  
build/*  
*.xcdeuser  
*.modelv3
```

```
# old skool  
.svn
```

```
# osx noise  
.DS_Store  
profile
```

`.gitattributes`

```
*.xcdeproj -crlf -diff -merge
```