GIT Setup

**Getting started**

1. Download git from here<http://git-scm.com/download>
2. Create a new folder for the project.   
   **example** “C:\Users\Evan\Documents\virtual\_canvas”
3. Right click on the folder and click “git init here”
4. Right click on the folder and click “git bash” to open the git console
5. Extract virtual\_canvans\_ssh.zip to c:\user\your\_user  
   This zip contains the git server configuration as well as the rsa private key. This needs to be kept safe, and for that reason I will give you a copy of the file via usb key only.
6. Now in git console type “git remote add master virtualcanvas:~/git”
7. Now type “git pull master –f master”
   1. If it’s working it should ask to verify the authenticity of the host – type yes
8. Now type “git checkout –f master” – now in the folder you shout have a new hidden file called “.gitignore”

**Get latest files from master, and copy them to the current directory**

git pull master –f master

git checkout –f master

**Start a dev branch**

git branch **branchname**

**Add changed files**

git status - show what files have changed

git add*path/filename.blah* - add a single file to be updated, do this for each changed file

commit –m “*update comment*”

git push master –f *branch\_name*

**Help Git Checkout is giving me errors**

This usually happens because you’ve made a change to a branch which hasn’t been committed. Use “git reset –hard” to remove any changes this will remove all the changes you’ve made, be careful.

If you want to keep those changes and also get the latest files you have options

1. Merge changes using the tortoisemerge tool
2. Commit the changes, then swap to a different branch and checkout there. Best option if you want to abandon a development branch and create a new dev branch from the latest release.

**How do i revert to an older release?**

git log shows list of all updates  
git checkout *update\_number* checkout one of the releases from the log

**Merging**Firstly make sure you have a mergetoool. Download TortoiseSVN for the mergetool and configure git to use it with these tools.

git config --global merge.tool tortoisemerge

git config --global mergetool.tortoisemerge.cmd

now simply type “git merge” and merge each file, notice that the mergetool usually merges 99% of it automatically. Only lines with “????” need to be done manually.

After the merge process, test the code, if all good commit the code & push the release.

**Show branches on server**

git ls-remote --heads master

gi