

Please follow each step carefully to avoid any weird bugs. Merging errors can be really annoying to deal with so let's try not to cause any :(

If you are not clear about any git commands or have technical issues, consult me asap before you change any files in the repository.

## **Clone the repo to your local machine**

1) Download and install Git on your PC if you haven't already.

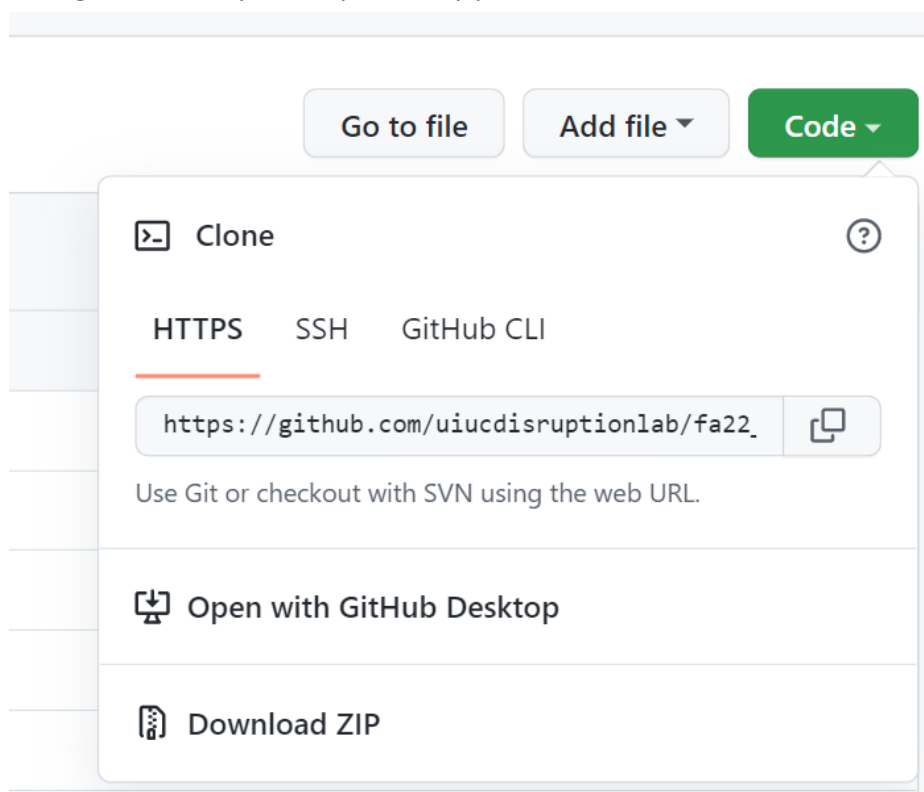
- download link: <https://git-scm.com/downloads>

2) Download and install GitLFS on your PC. This will be used to upload large files that exceed Github's storage limit.

- download link: <https://git-lfs.github.com/>

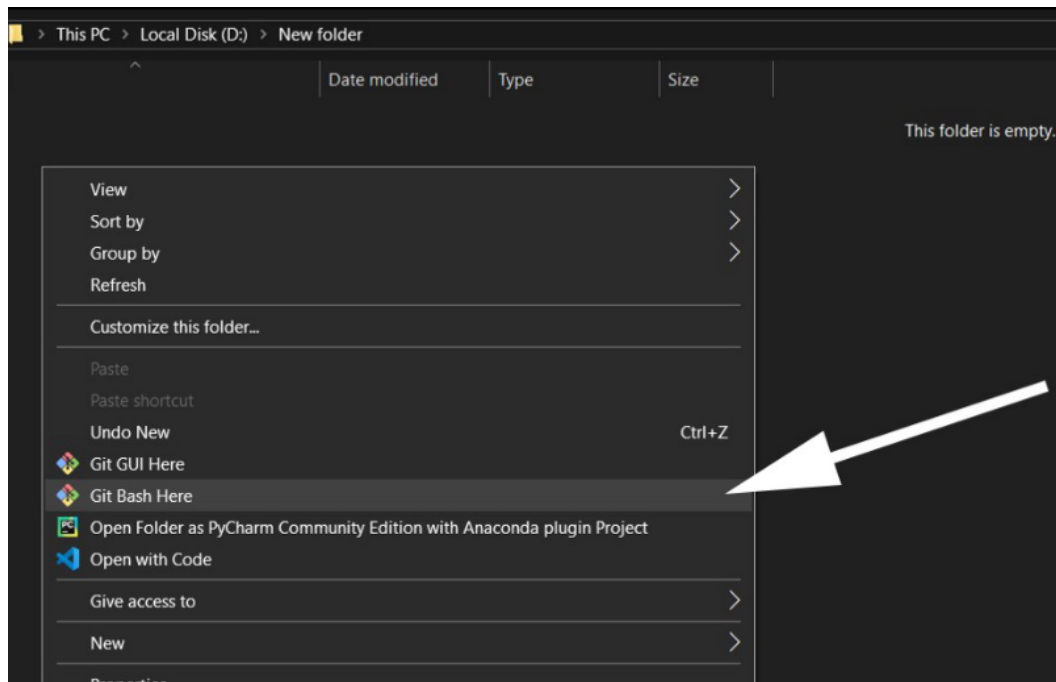
3) Now clone the github repo to your local machine.

- First go to our repository and copy the 'clone/download' URL.



Repo link: [https://github.com/uiucdisruptionlab/fa22\\_giesVR.git](https://github.com/uiucdisruptionlab/fa22_giesVR.git)

- Create a new folder anywhere on your pc (on a disk that has enough storage).  
(The size of the project file is already at 11 gigabytes now, and it might double or even triple in the future. So just make sure to keep it somewhere with enough capacity. )
- Right-click in your new folder and click "Git Bash Here".



- Now, add your credentials by using the following commands:

```
$ git config -- global user.email youremail@example.com
```

```
$ git config -- global user.password yourpassword
```

```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (main)
$ git config --global user.email aoruli2@illinois.edu

Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (main)
$ git config --global user.password 12345678
```

Be sure to use your illinois.edu email or an email with access to this repo.

If you are getting an “Authentication Failed” error, use these commands to print your credentials and see if the information is correct.

```
$ git config user.email
```

```
$ git config user.password
```

```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (main)
$ git config user.email
aoruli2@illinois.edu

Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (main)
$ git config user.password
12345678
```

If you’re still having troubles connecting to the repo with the correct credentials, contact me asap.

- Now, run the following command in the bash opened. Replace the link with our repo link.

```
$ git clone link you copied
```

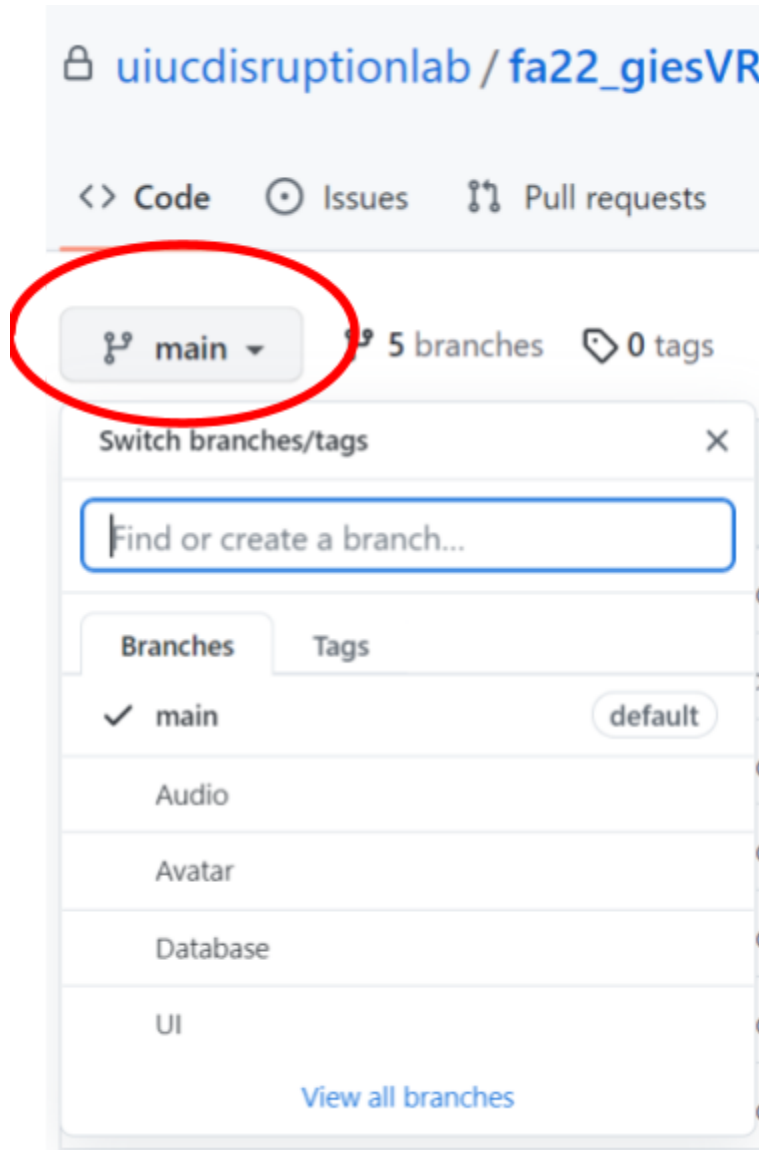
```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (main)
$ git clone https://github.com/uiucdisruptionlab/fa22_giesVR.git
```

- Now you have successfully cloned the repository to your local machine!

## Branches

To avoid merging conflicts and overwriting files, I have created a branch for each group.

Go to our repository and click on “main”, and you can see all existing branches. You can view and edit files in a branch by clicking on its name.



## 1) Check which branch you are on

After you have cloned the repository, you are most likely on the “main” branch.

- Use the command `$ git branch` to see which branch you are on.

```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (main)
$ git branch
* main
```

- Use the command `$ git branch -a` to see all existing branches in the repository.

```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (main)
$ git branch -a
* main
remotes/origin/Avatar
remotes/origin/Database
remotes/origin/HEAD -> origin/main
remotes/origin/UI
remotes/origin/main
```

## 2) Switch to other branches

- Use the command `$ git checkout your-branch-name` to get to other branches.

```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (main)
$ git checkout Avatar
Updating files: 100% (16/16), done.
Switched to a new branch 'Avatar'
Branch 'Avatar' set up to track remote branch 'Avatar' from 'origin'.
Encountered 1 file(s) that should have been pointers, but weren't:
    Library/PackageCache/com.unity.burst@1.6.6/.Runtime/hostlin/11d
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (Avatar)
$ |
```

For the UI team, you might get this error because there's a folder in the repo also called "UI", and the terminal got confused.

```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (Avatar)
$ git checkout UI
fatal: 'UI' could be both a local file and a tracking branch.
Please use -- (and optionally --no-guess) to disambiguate
```

Use `$ git checkout UI --` or `$ git switch UI` instead.

```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (Avatar)
$ git checkout UI --
Updating files: 100% (2/2), done.
Switched to a new branch 'UI'
Branch 'UI' set up to track remote branch 'UI' from 'origin'.
Encountered 1 file(s) that should have been pointers, but weren't:
    Library/PackageCache/com.unity.burst@1.6.6/.Runtime/hostlin/11d
```

```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (main)
$ git switch UI
Updating files: 100% (15/15), done.
Switched to branch 'UI'
Your branch is up to date with 'origin/UI'.
```

(Remember: Only pull from and push to your own branch!)

## Pull & Push

Before you do any of these things, make sure to **communicate with people you're working with**. Let them know what you're working on and what files you have modified. Try to avoid having more than one person working on a file at the same time as it might lead to merging errors.

### Pull

(Warning: Always pull first before doing anything to the project, even opening up Unity editor!!!!!!)

The Unity editor manages hundreds of temporary files. Merely opening or closing the editor will create, delete, or modify some of these files. So to avoid any merge conflicts, always run the **git pull** command first in your local repository, then open up the editor.

Now, I will show you how to pull new changes:

**1) Make sure you're on your own branch!**

- No matter if you are pulling or pushing, always make sure you are on your own branch by using the `$ git branch` command. This prevents you from overwriting someone else's files or having your files overwritten by someone else.

- 2) Use the command `$ git pull` to pull new changes from the repository.

```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (Avatar)
$ git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 660 bytes | 73.00 KiB/s, done.
From https://github.com/uiucdisruptionlab/fa22_giesVR
  9d0a4a34..790f3f4f  main      -> origin/main
```

- Check your local repository for newly added files. After you have successfully pulled the new changes, you can start working on the project!

## Push

After you are done working on the project, you will need to push your work from your local repository to the remote repository.

Now, I will show you how to push to remote repository:

### 1) Make sure you're on your branch!

- Same as pulling, check if you are on your own branch so you won't mess up someone else's files.

### 2) Add, Commit, Push

Follow the following steps to push to remote repository:

- Use the command `$ git add .` or `$ git add -A` to add and track new files.

```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (main)
$ git add .
warning: LF will be replaced by CRLF in Library/CurrentLayout-default.dwlt.
The file will have its original line endings in your working directory
```

It is very likely you will get this “LF will be replaced by CRLF” warning. It is safe to just ignore it.

- Use the command `$ git commit -m "Your message here"` to commit your changes. Write a meaningful commit message to indicate what you have done.

```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (main)
$ git commit -m "Gies VR"
[main 9d0a4a34] Gies VR
16 files changed, 240310 insertions(+), 120467 deletions(-)
rewrite Library/PackageCache/com.unity.burst@1.6.6/.Runtime/hostlin/1ld (99%)
rewrite Library/gfu/cache/gitstatusentries.yaml (76%)
```



- Use the command `$ git push origin Your-branch-name` to push to remote repository.

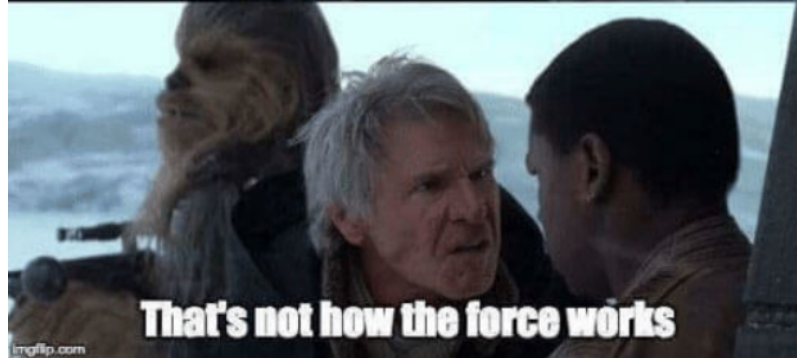
```
Li@DESKTOP-ILR4QKL MINGW64 /e/fall22_giesVR/fa22_giesVR (Avatar)
$ git push origin Avatar
Uploading LFS objects: 100% (1/1), 61 MB | 0 B/s, done.
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 12 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 735 bytes | 735.00 KiB/s, done.
Total 8 (delta 5), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (5/5), completed with 5 local objects
To https://github.com/uiucdisruptionlab/fa22_giesVR.git
96ff39a7..497bb038 Avatar -> Avatar
```

Remember to never push to the “main” branch. I will do code review on each branch first, put together the entire project, then push to main at the end of every week.

If you are having trouble with any of these steps, please let me know and I’ll find a way to resolve it. Please never use the `git push --force` command to force push to repo as it might mess up the entire project :(



git push --force



**That's not how the force works**