Tutorial: Github Custom Tutorial Speedrun

Made by: A Really Bad Time

Jan 2024

1 Introduction

This document is meant to outline a brief summary of what **Github** is, what is it used for, why it's useful and steps on how to install and use it. I made this document since Github is primarily used by Software Engineers to keep track of code. This means most of the tutorials online are heavily directed towards Software Engineers and writing code. However, Github since can be used to keep track of **file and folder changes** this means it can serve as a useful tool to other developers/creators which needs to keep track of different versions of their work. This means Github can serve as a valuable tool to have different versions of your work which you can always go back to at any point in time.

On top of this, Github is also used as a way for multiple collaborators to have a copy of the same project files and folders. This is good since each collaborator can make changes to the project and these changes will be seen and updated for everyone working on the project.

2 Github Summary

As a quick summary, Github is an application used to keep track of changes within your workspace. This is done by the creation of something called a Github repository. Github repositories are projects stored online which collaborators use in order to make a copies it in their own computers (local machines). An example of this can be seen in Figure 1 in which each collaborator has their own local copy of the "Game Project" repository. All the collaborators working on the project can also update and see changes other collaborators have made on the "Game Project" repository which ensures they all have the most up-to-date files

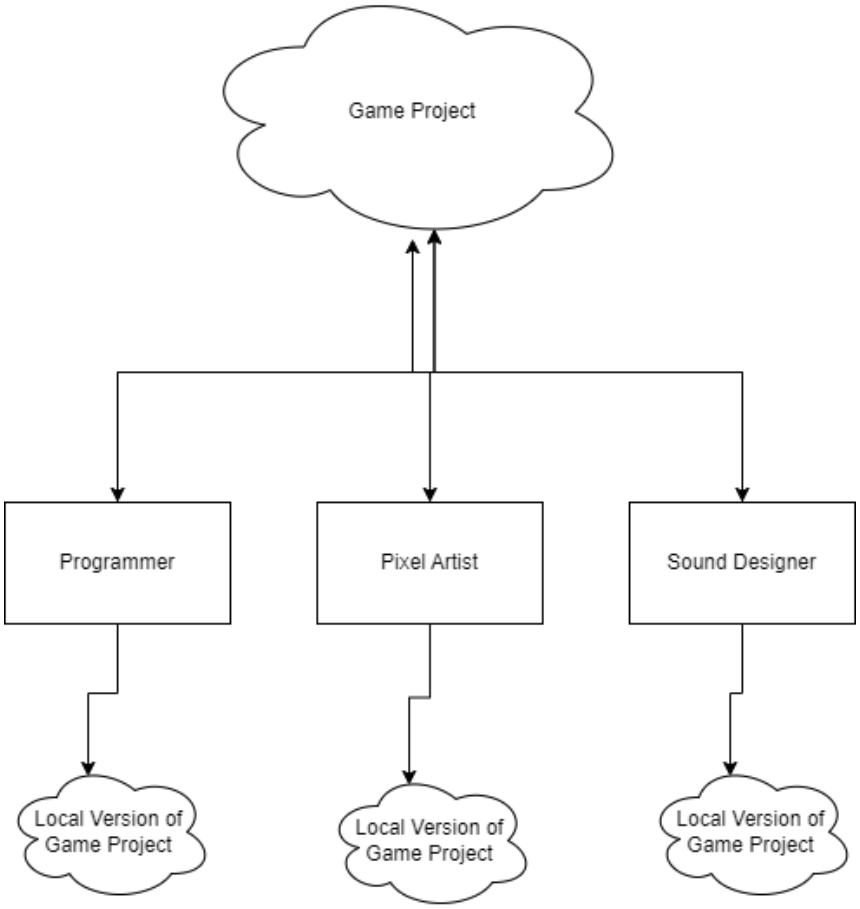


Figure 1: Github Repo Example

Github repositories contain files and folders which Github tracks. This means if any folder or file get added, renamed, deleted **on your own com-**

puter, Github allows you to track these changes and take a "snapshot" of the current state of your project. You can then upload these changes/snapshots on the **online** repository so that everyone else in the team also gets the updates you've made. Figure 2 shows a diagram example of an artist working on the concept art of an enemy. As you can see they have made 4 commits on their workspace (this is Githubs terminology of taking 4 snapshots), with the most recent being "Game Project v1.3". We can see that this is the most recent not only due to the arrows being pointed but also because of where the "Head" indicator is. In Github, the HEAD commit simply means the most recent (up to date) "snapshot" of your files and folders.

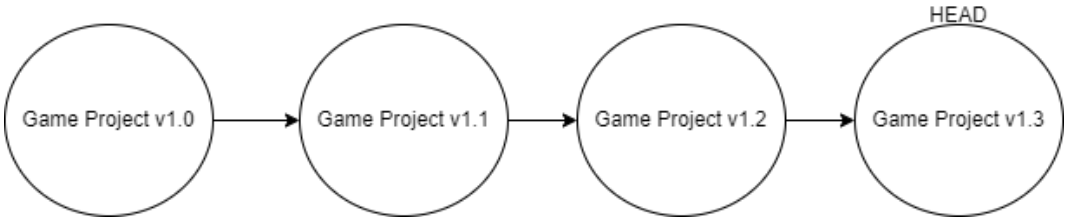


Figure 2: Simple file change example

A more detailed version of Figure 2 can be seen on Figure 3 where it can be seen that 2 files are added (File 4 and File 5). The amazing thing about Github is that it stories versions of every single commit you tell it to make. This means the pixel artist can go back to any of the versions they have made beforehand and access their files exactly how it was when they made the commit.

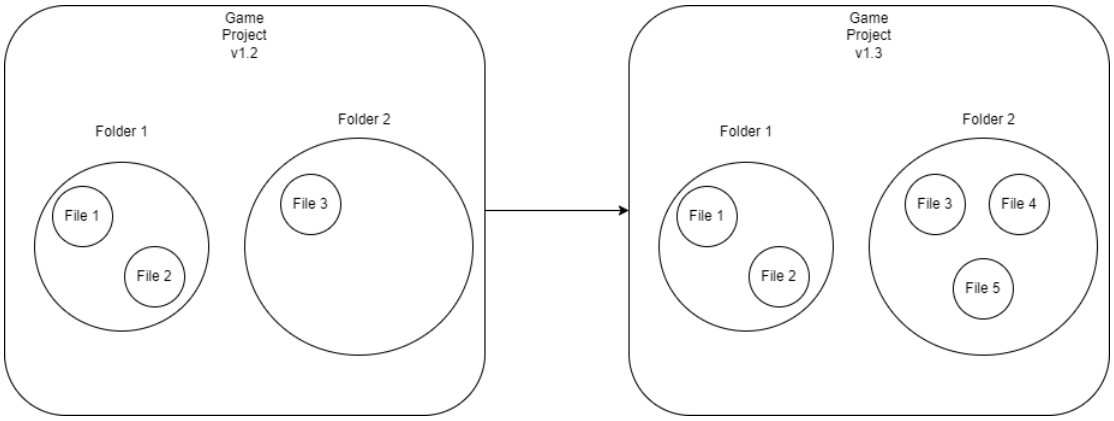
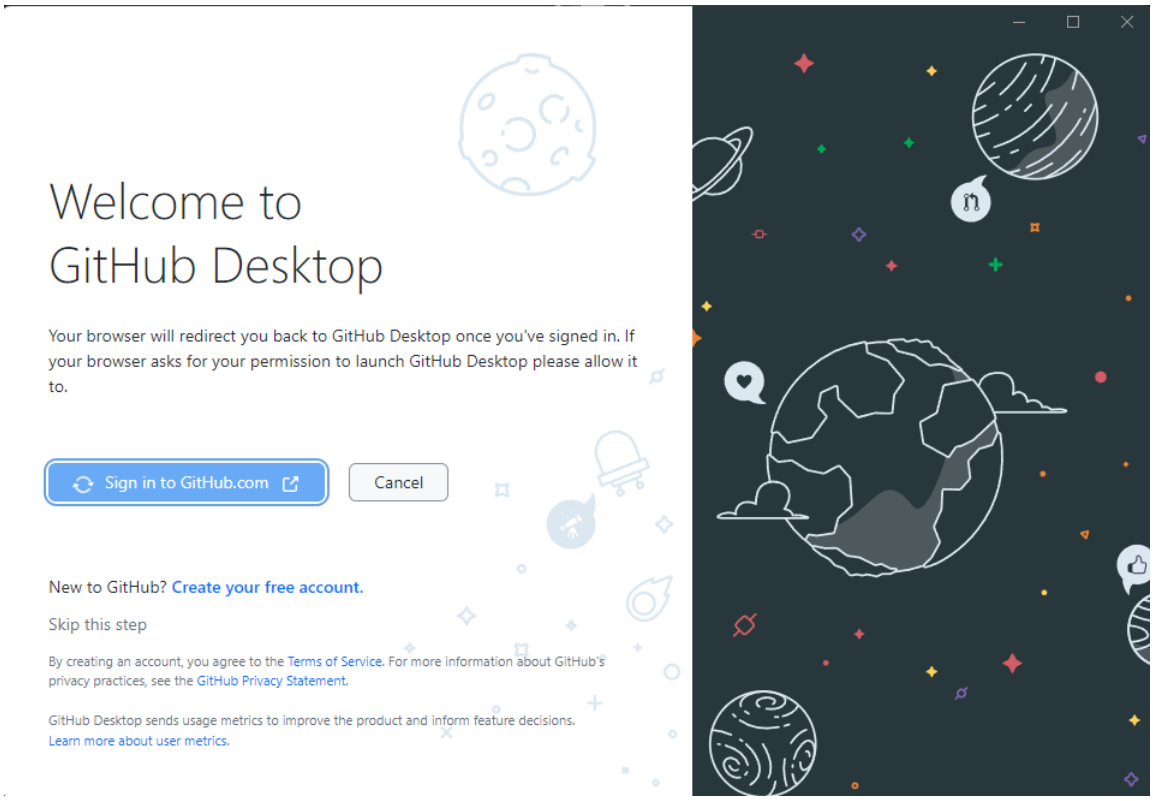


Figure 3: More detailed view of changes in Figure 2

3 Setting Up Github

1. Navigate to https://github.com and register an account if you haven't already
2. Navigate to https://desktop.github.com to download Github Desktop and run the installer. This should look like Figure 2

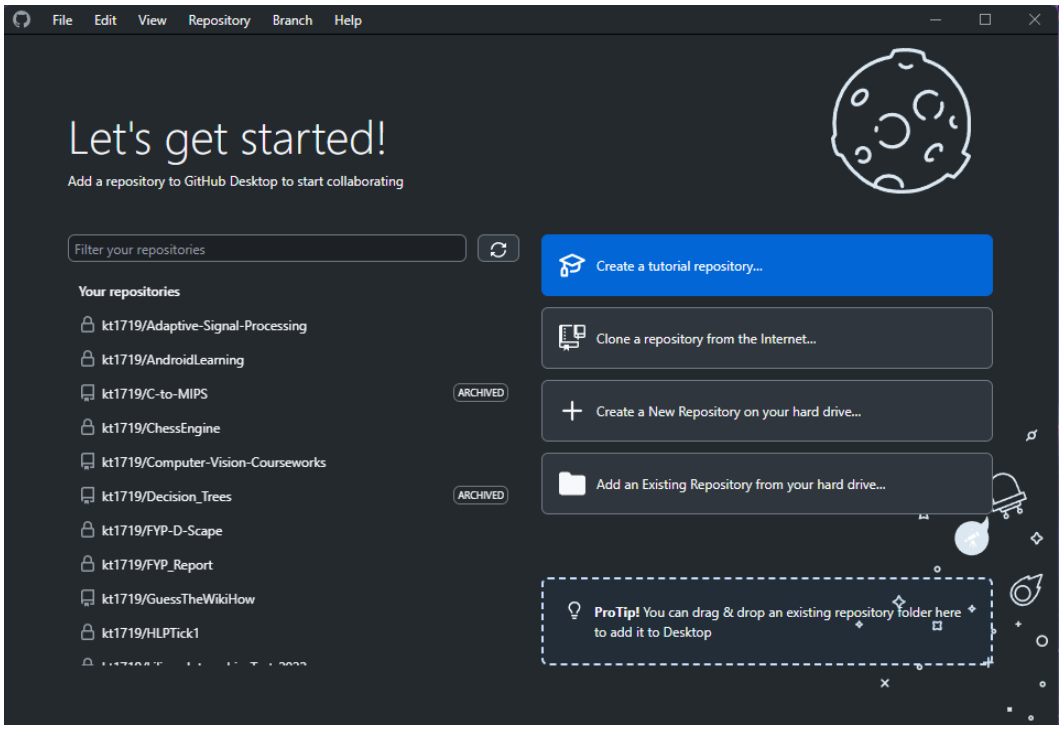


3. Press "Sign in to Github.com" which should send you to a page to login.
4. If you are on Windows you should be able to click "Open GithubDesktop" on your browser which should redirect you back to the Github Desktop App
5. Your name and email should also be automatically filled out however still make sure that the email and names filled in are correct and click Finish. That's it! Now you have Github

4 Using Github Desktop

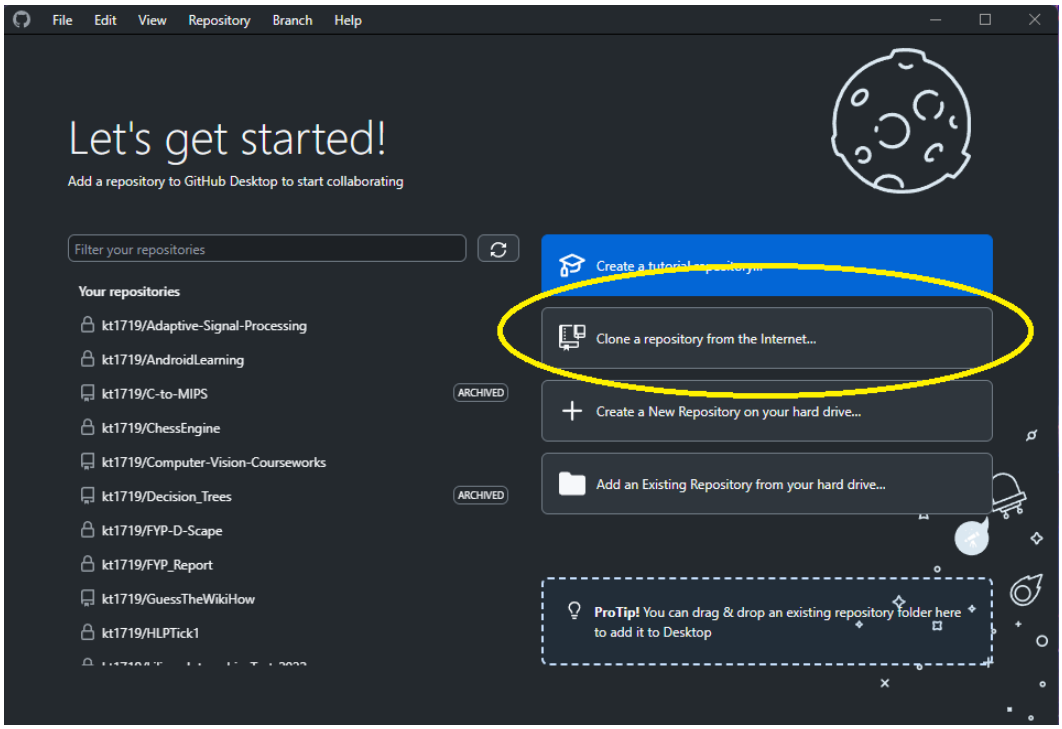
- Make sure before entering this step you have already either created or joined a repository.

Once you have joined a repository, go to the repository base URL. Normally this will be on the form similar to:
https://github.com/$exampleuser$/$examplerepo$
After installing Github desktop your screen should be met with this:

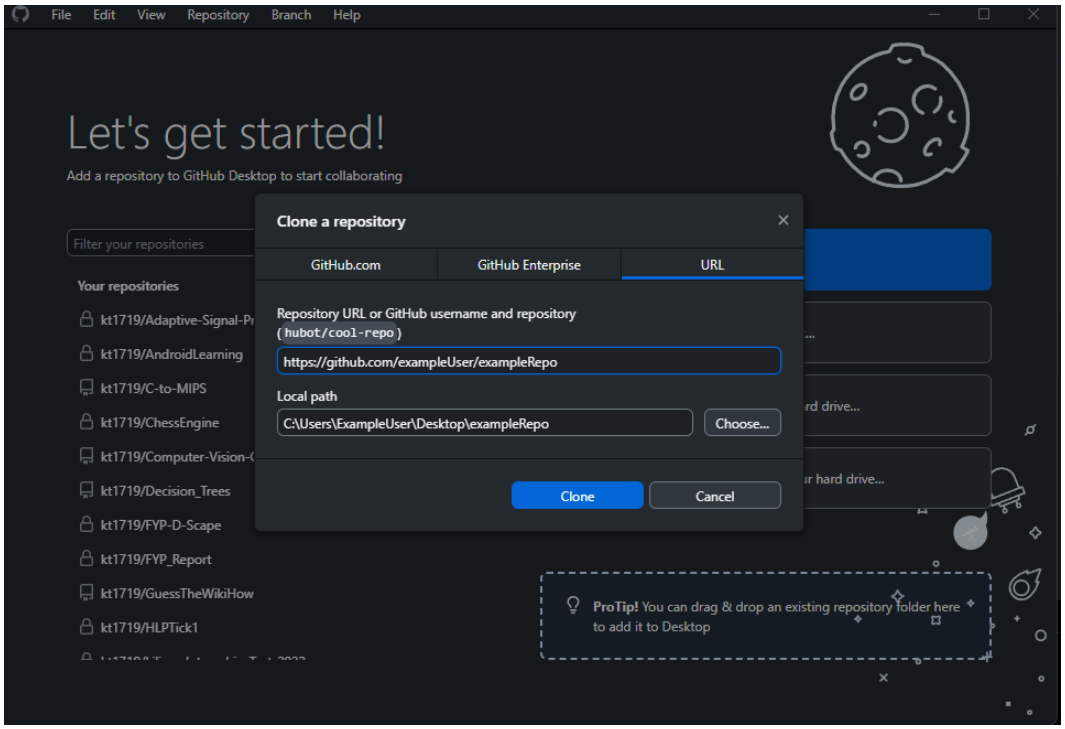


Now you need to make a clone of the online repository in your machine.

1. Click on the "Clone a repository from the internet..." button as shown below:



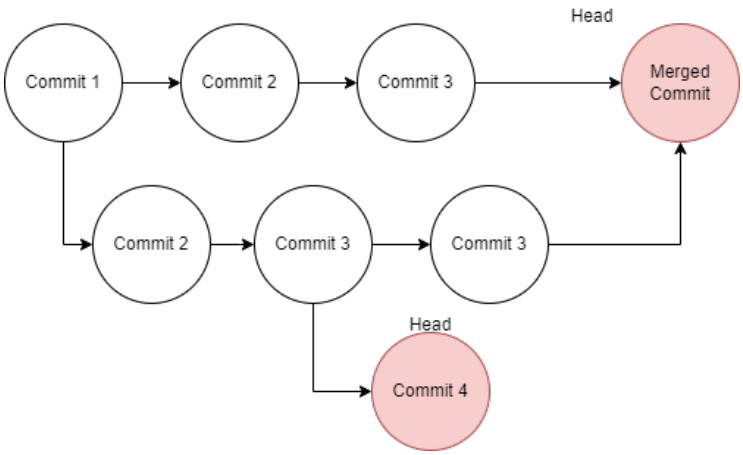
2. Next, click on the "URL" section on cloning the repository and paste the URL of the repository page. An example can be seen below:



3. Click Clone and you should have a clone of the repository on your local computer!

5 Basic Github Term Summary

1. Fetch - Check if there are any changes made on the repository online.
2. Pull - Update the files on your local machine to match the files on the repository online.
3. Push - Push changes you've made to the repository online.
4. HEAD - Most up to date commit for a specific timeline.
5. Branch - Another "timeline", reference image below.



To be updated...