

Intro to GitHub



Chris, Miguel, Nathan, Nic, Tony

Part 1: What is GitHub, How is it Used?



What is GitHub?



GitHub is how people build software

We're supporting a community where more than 56 million* people learn, share, and work together to build software.

October 2007

First commit

San Francisco

Headquarters

100 million*

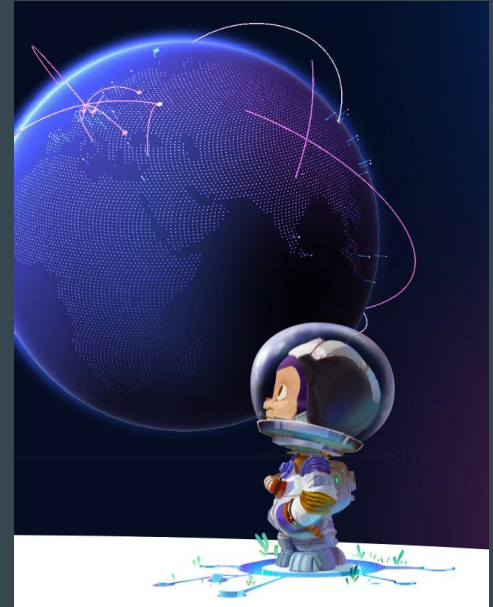
Repositories hosted

* As of August 2019

What is GitHub? (Cont.)

- Platform for Building Software
- Worldwide Development Capabilities
- Graphical Interface for Version Control Management
- Progress Tracker
- Cloud Storage and Development

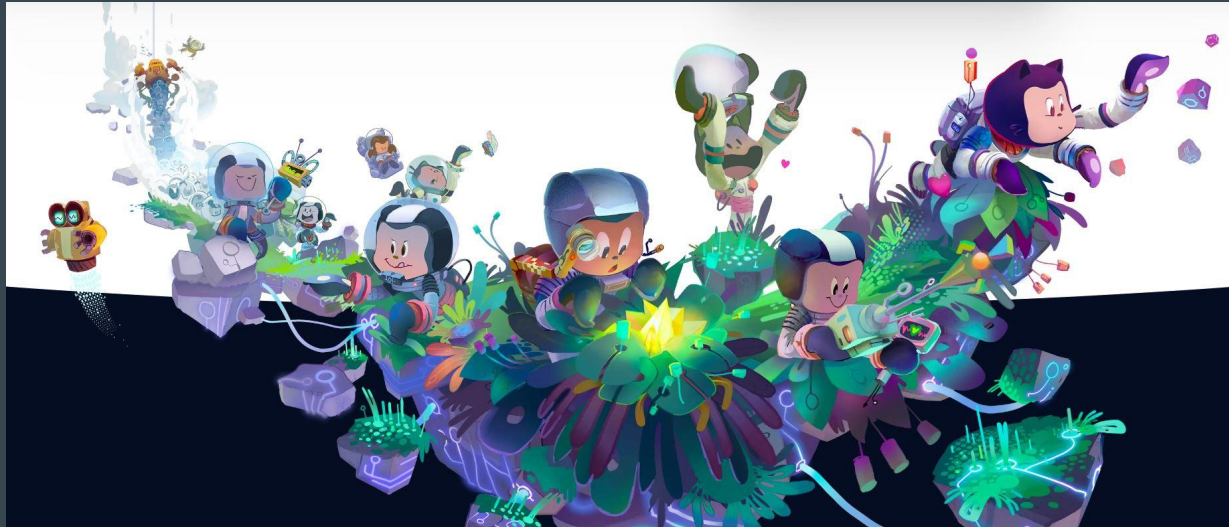
**Where the world
builds software**

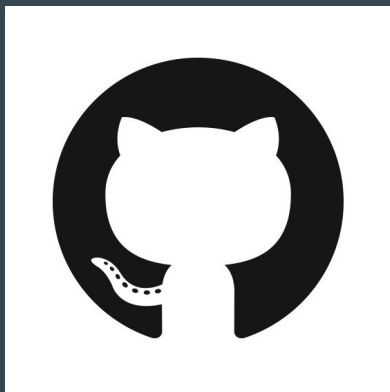


How is GitHub Used?



- Record or Rewind Changes
- Build on the Built, or Create New
- Code Review with Pull Requests
- Instant Developer Environments
- Find and Fix Vulnerabilities or Issues





Octocat



Octocat Figurine \$30



Size

5" Octocat Figurine ▾

Quantity

1

Add to Cart



Octocat Merch



Part 2: Setting up GitHub Desktop

Creating a GitHub account

- Go to <https://github.com/join>
- Fill in the information to create your GitHub account

Downloading and Installing GitHub Desktop

- GitHub Desktop allows you to interface with repositories from your desktop/laptop without having to use the command-line
- Go to <https://desktop.github.com/>
- Download the installer for your system (Windows (64bit), Windows (msi), macOS...)
- Open the installer
 - Follow the instructions (if any) to install GitHub Desktop
- Sign in to your GitHub account on GitHub Desktop
 - Using the toolbar on the top of the screen, open File -> Options -> Account
 - Sign in

[File](#)[Edit](#)[View](#)[Repository](#)[Branch](#)[Help](#)[New repository...](#) Ctrl+N[Add local repository...](#) Ctrl+O[Clone repository...](#) Ctrl+G[Options...](#)[Exit](#)

Current branch
main



Push origin
Last fetched 2 minutes ago



Options

[Accounts](#)[Integrations](#)[Git](#)[Appearance](#)[Prompts](#)[Advanced](#)

GitHub.com



lotkey
@lotkey

[Sign out](#)

GitHub Enterprise

If you have a GitHub Enterprise or AE account at work, sign in to it to get access to your repositories.

[Sign in](#)

View the files of your repository in Explorer
Repository menu or Ctrl Shift F

Part 3: Workflow with GitHub & GitHub Desktop



Creating Your First Repository (Repo)

- From GitHub.com
 - After registering a new account, click on the green new icon on the top left under your username

The screenshot shows the GitHub homepage for a user named A-Vela94. The top navigation bar includes the GitHub logo, a search bar, and links to Pull requests, Issues, Marketplace, and Explore. The left sidebar shows the user's profile, a list of repositories (including 'Unex-Uldaho/Cyberjunk2020'), and a section for teams. The main content area is divided into two sections. The top section, titled 'Learn Git and GitHub without any code!', contains a green box with a close button (X) in the top right corner. Inside this box, it says 'Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.' Below this text are two buttons: 'Read the guide' (green) and 'Start a project' (white). The bottom section, titled 'Discover interesting projects and people to populate your personal news feed.', contains a paragraph about the news feed and a button labeled 'Explore GitHub'.

Search or jump to... Pull requests Issues Marketplace Explore

A-Vela94

Repositories

Find a repository...

Unex-Uldaho/Cyberjunk2020

Your teams

You don't belong to any teams yet!

Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

Read the guide Start a project

Discover interesting projects and people to populate your personal news feed.

Your news feed helps you keep up with recent activity on repositories you [watch](#) and people you [follow](#).

Explore GitHub

ProTip! The feed shows you events from people you [follow](#) and repositories you [watch](#).

Subscribe to your news feed

- What is a repository?

- A repository, or repo, is a directory hosted by GitHub that contains all of your project files. You can choose to make your repo public or private based on the needs of your project.
- After clicking on New, you'll be presented with the following screen.

New Repository

Create a new repository

Owner *

Repository name *



/ TestRepo



Give your repository a name fitting for your project

Great repository names are short and memorable. Need inspiration? How about [turbo-rotary-phone](#)?

Description (optional)

Give your project a short description →

How to Create a new Repository Demo



Public

Anyone on the internet can see this repository. You choose who can commit.

Choose to make your project public for all to see, or private to invite specific collaborators



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☒ Add a README file

This is where you can write a long description for your project. [Learn more.](#)

The README file will be the first page displayed to users on your GitHub repository. Its purpose is to guide users/collaborators in how to use/collaborate on your application

☒ Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: **Unity**

You can choose to add a default README.md file as well as a .gitignore file for your type of project

GitHub provides templates for a .gitignore file for various types of projects. Since our project will be developed in Unity, we'll choose that template

☐ Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

You can also choose a license for your project

This will set **main** as the default branch. Change the default name in your [settings](#).

Create repository

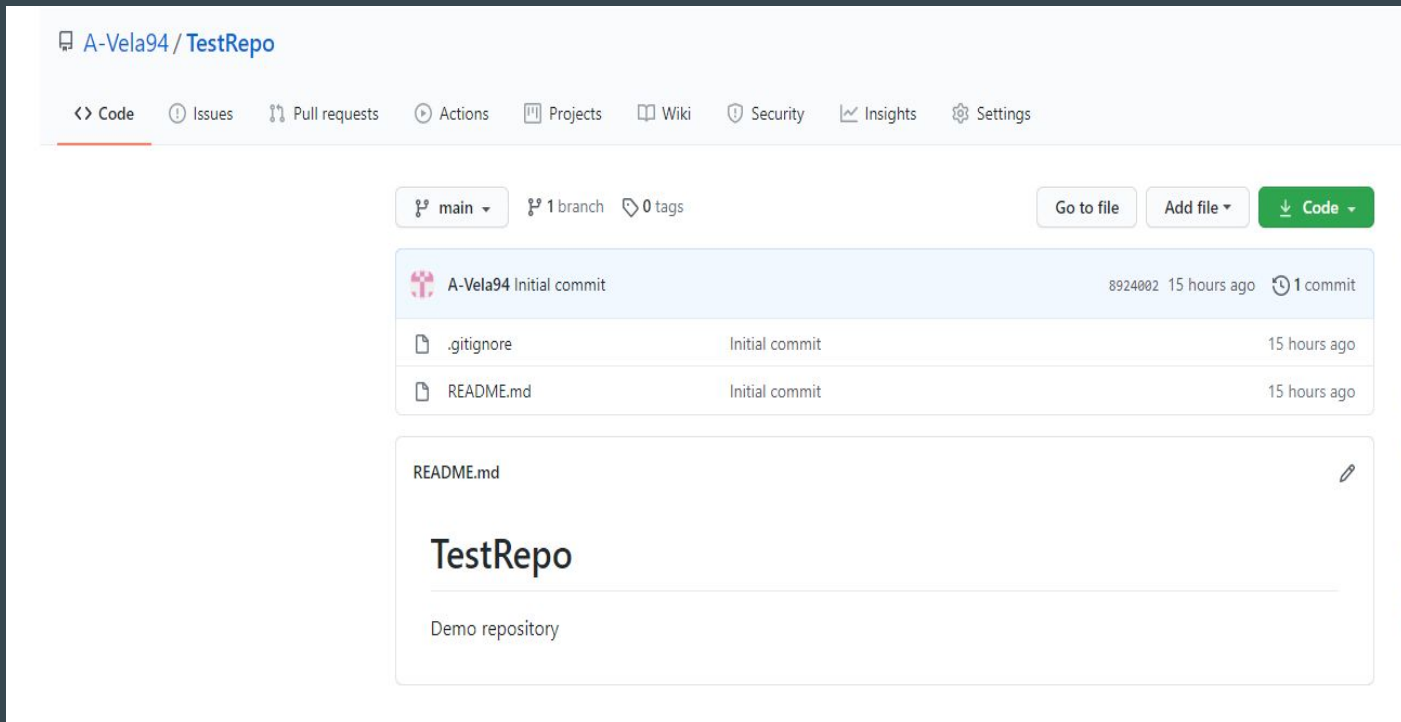
A Quick Note on README.md, .gitignore, and Licensing

- README.md
 - From the GitHub Documentation: docs.github.com/en/github/creating-cloning-and-archiving-repositories/about-readmes
 - 'You can add a README file to a repository to communicate important information about your project. A README, along with a repository license, contribution guidelines, and a code of conduct, communicates expectations for your project and helps you manage contributions.'
- .gitignore
 - From GitHub Documentation docs.github.com/en/github/using-git/ignoring-files#excluding-local-files-without-creating-a-gitignore-file
 - 'You can create a `.gitignore` file in your repository's root directory to tell Git which files and directories to ignore when you make a commit. To share the ignore rules with other users who clone the repository, commit the `.gitignore` file in to your repository.'
- Licensing
 - There are many different open-source Licenses available to use with your project. This article is a comprehensive guide to licensing your project. Read to learn more.
 - docs.github.com/en/github/creating-cloning-and-archiving-repositories/licensing-a-repository#searching-github-by-license-type



New Repo cont'd.

After creating your repository, you should see something similar to this.



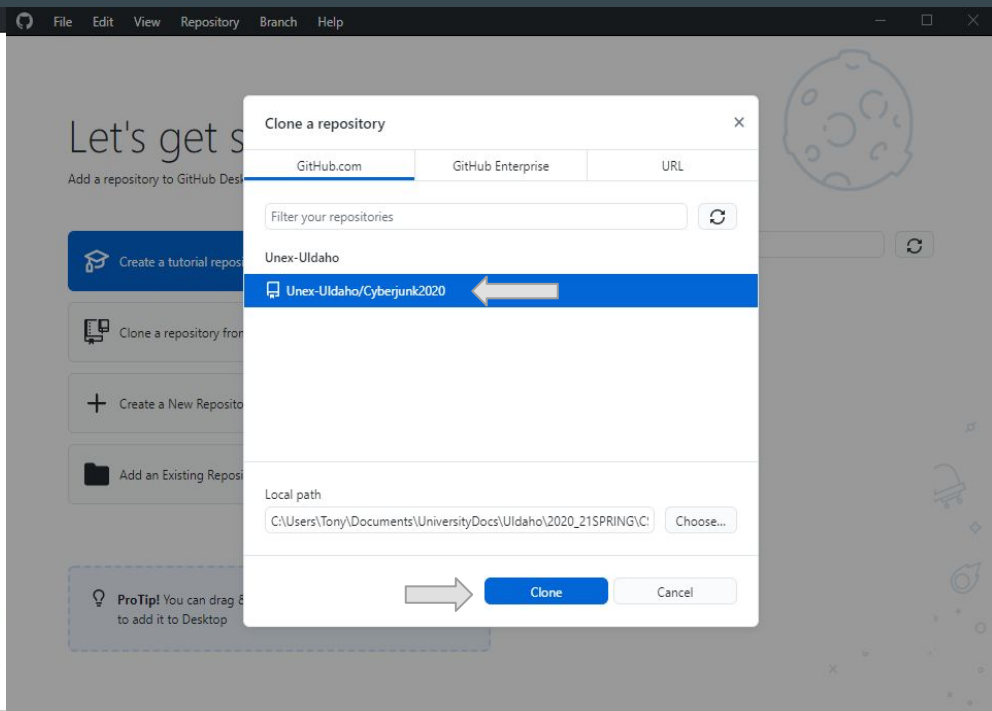
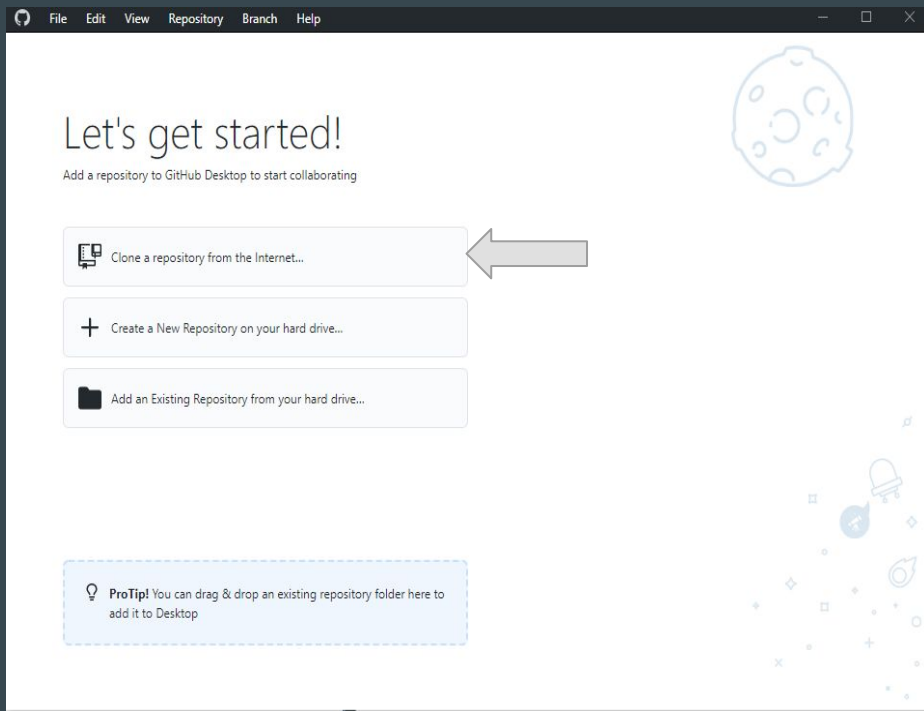


Cloning Your Repositories With GitHub Desktop

- *You should have already installed and configured GitHub Desktop prior to this step. If you haven't, return to that step in this guide. For more in-depth instructions visit docs.github.com/en/github/getting-started-with-github/github-desktop.*
- After configuring GitHub Desktop and signing in via your browser you'll see this screen.

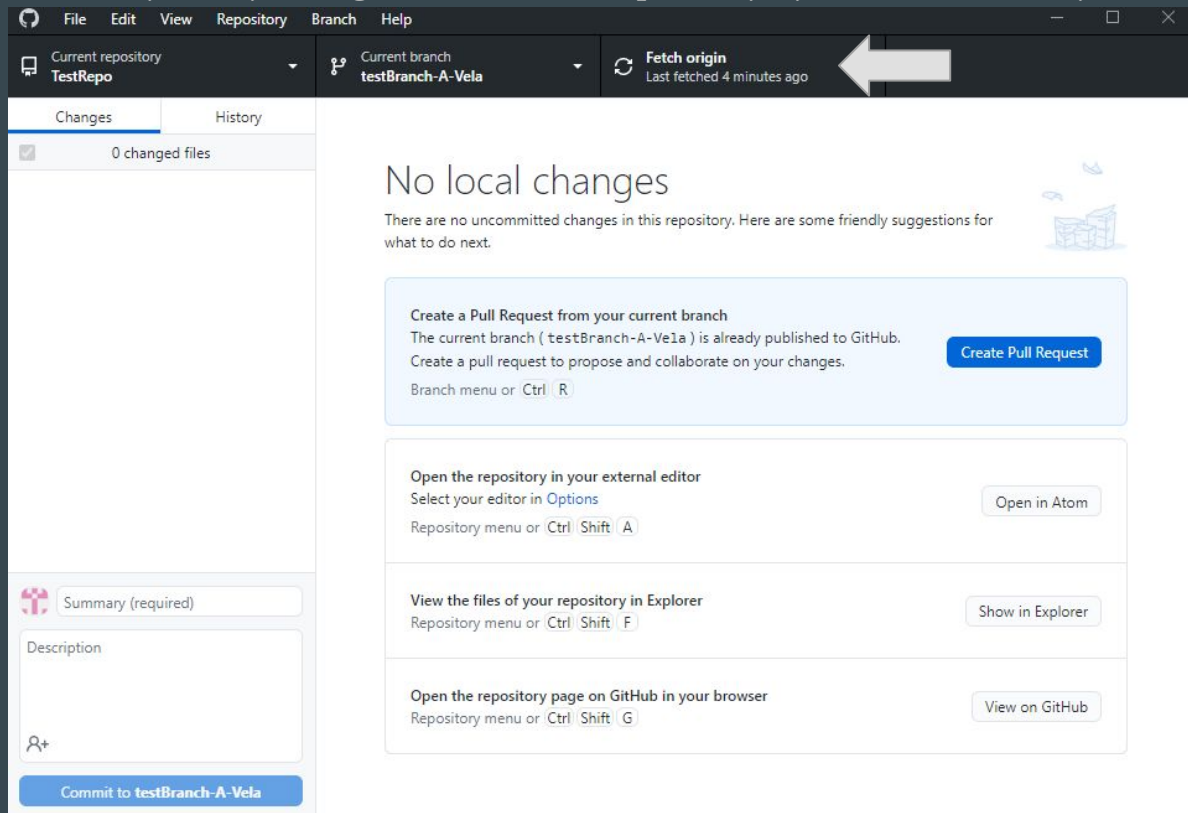
Click on "Clone a repository from the internet..." you'll be prompted to sign in to github.com if you've logged out.

Select the repository to be cloned from the list, the repo you created should be found here, and choose the directory where this repository will be stored on your local machine.



Fetching Origin GitHub Desktop

Whenever you start working on repository cloned to your local machine, remember to “Fetch” the origin of your project to resync any changes made to the repository by collaborators to your local project directory.

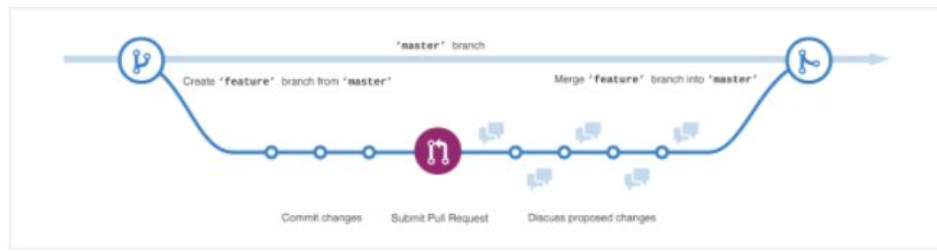


Getting Familiar with Branching

- What is branching?
 - As stated in the introductory guide on GitHub's web site: guides.github.com/activities/hello-world/
 - “Branching is the way to work on different versions of a repository at the same time”
 - When you create a new repository, the “main” branch gets created, this is considered the definitive branch that all other branches will get merged back into.
 - When a new branch is created off of the main, basically, a copy of the main branch is created and can be edited without committing those edits to the main branch.
 - If changes are made to the main branch while you're working on your separate branch, GitHub makes it possible to pull those changes into your branch.
- This diagram from GitHub's guide illustrates this workflow process

This diagram shows:

- The `main` branch
- A new branch called `feature` (because we're doing 'feature work' on this branch)
- The journey that `feature` takes before it's merged into `main`



Getting Familiar with Branching cont'd.

- Branching is like saving different versions of a file, with the original file being the main, and additional copies thought of as branches. As the creators of GitHub explain:

Have you ever saved different versions of a file? Something like:

- `story.txt`
- `story-joe-edit.txt`
- `story-joe-edit-reviewed.txt`

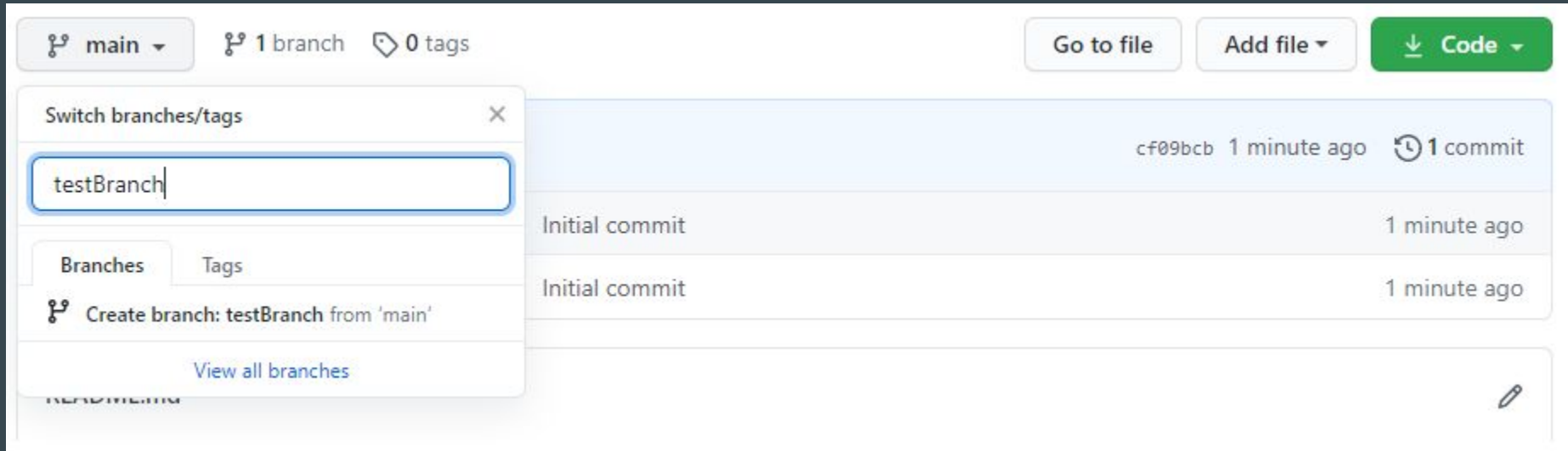
Branches accomplish similar goals in GitHub repositories.

Here at GitHub, our developers, writers, and designers use branches for keeping bug fixes and feature work separate from our `main` (production) branch. When a change is ready, they merge their branch into `main`.

- The use of branches makes collaborating on large projects easier to scale and maintain.

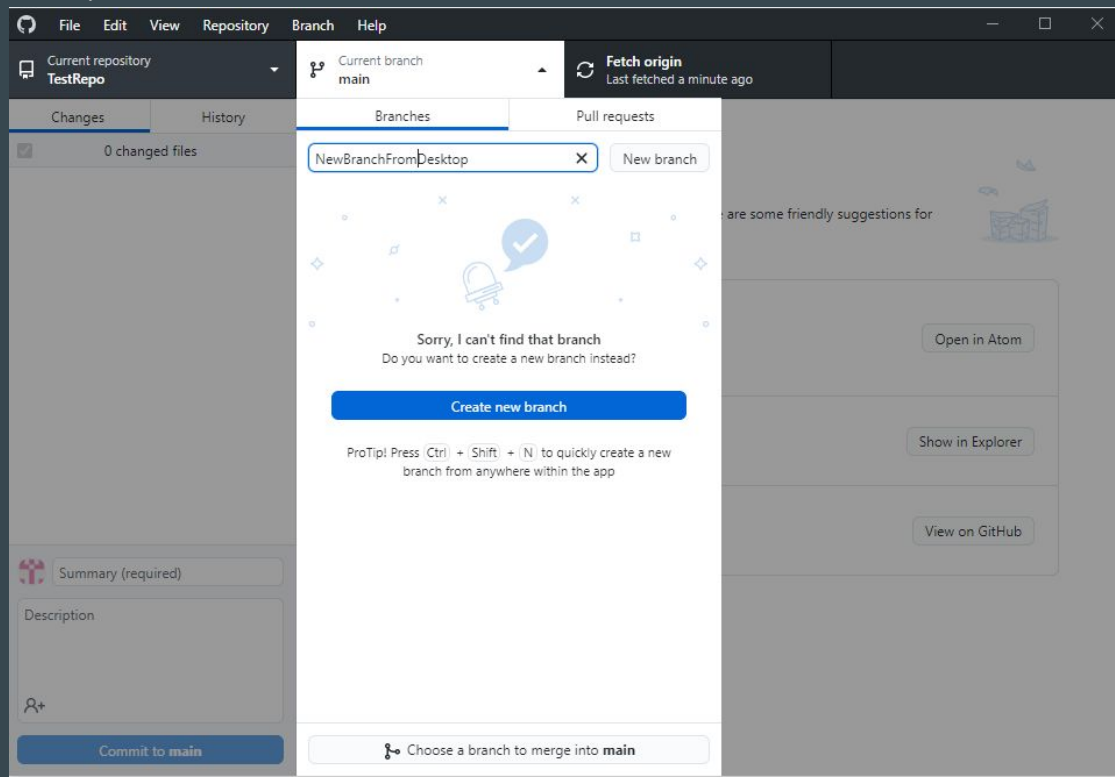
Creating a New Branch-Browser View

- To create a new branch in GitHub from the Browser:
 - GitHub in-browser:
 - Navigate to your repository
 - Click the drop-down arrow where your main branch is
 - Type the desired name of your new branch and hit enter and your new branch will be created



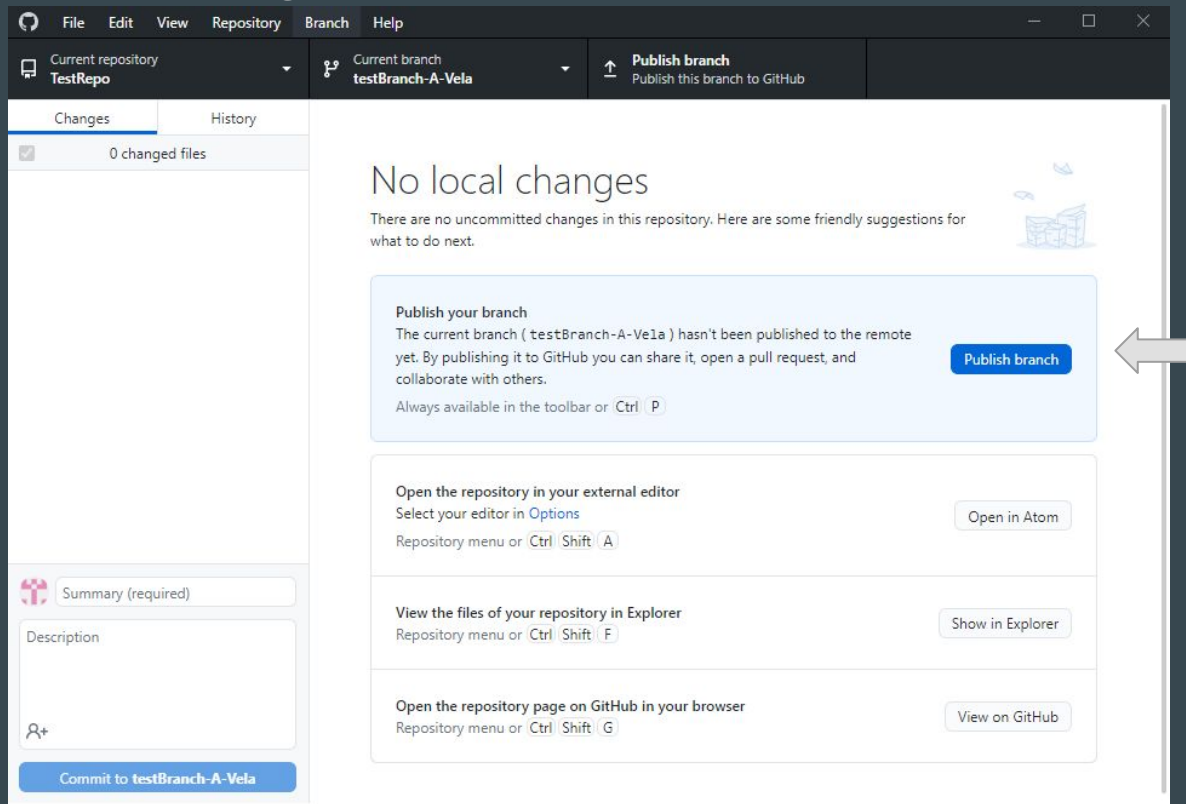
Creating a New Branch: GitHub Desktop

- Creating a new branch using GitHub Desktop is very similar
 - Open the application, click the drop-down arrow on the main branch, enter the name and hit enter.
 - If the repository has a branch created off of the main branch already, choose which branch you want to base your new branch off of.



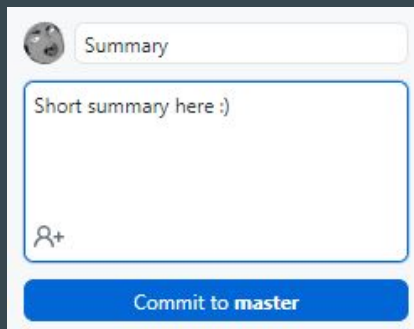
Publishing a New Branch: GitHub Desktop cont'd.

- To publish your new branch created in GitHub Desktop, click Publish branch. Now your new branch should be visible on github in-browser.



Making Edits and Commits

- Now that you have a new branch to start making changes to, you can start changing things up and making some commits.
 - Saved changes to files stored in a repository are called commits.
 - Each new commit generates a message with a description of the changes for all to see.
 - When a developer makes a commit, they should submit a short, descriptive message detailing changes.
- Edits/commits can be made via browser, or on a local machine with a cloned repository via GitHub Desktop



A screenshot of a web-based commit form. At the top left is a circular profile picture of a person. To its right is a text input field containing the word "Summary". Below this is a larger text area with the placeholder text "Short summary here :)". At the bottom left of the text area is a small icon of a person with a plus sign. At the bottom of the form is a blue button with the text "Commit to master".

Making Edits and Commits-Browser

- From your new branch: (click the drop-down on main and make sure to select the new branch)
 - Select the file in the new branch repository you wish to edit, we'll edit README.md below:

The screenshot shows the GitHub web interface for a repository named 'TestRepo'. At the top, there's a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below this, the 'Confirm your branch' section shows the current branch is 'testBranch', which is 2 branches away from main and has 0 tags. There are buttons for 'Go to file', 'Add file', and 'Code'. A message states 'This branch is even with main.' with links for 'Pull request' and 'Compare'. Below this is a list of commits. The first commit is 'TooYoungToDieHard Initial commit' from 36 minutes ago. It lists two files: '.gitignore' and 'README.md', both marked as 'Initial commit' and '36 minutes ago'. An arrow points to the 'README.md' file with the text 'Double-click desired file'. Below the commit list, the 'README.md' file content is shown, featuring the repository name 'TestRepo' and the description 'Another how-to demo repository'. On the right side, there are sections for 'About' (Another how-to demo repository), 'Releases' (No releases published, with a link to 'Create a new release'), and 'Packages' (No packages published, with a link to 'Publish your first package').

Confirm your branch → testBranch 2 branches 0 tags

Go to file Add file Code

This branch is even with main. Pull request Compare

TooYoungToDieHard Initial commit c-f09bcb 36 minutes ago 1 commit

.gitignore Initial commit 36 minutes ago

README.md Initial commit 36 minutes ago

Double-click desired file →

README.md

TestRepo

Another how-to demo repository

About Another how-to demo repository

Releases No releases published Create a new release

Packages No packages published Publish your first package

Making Edits and Commits-Browser cont'd.


Click the pencil icon on the right of the page to edit the file.


testBranch ▾


TestRepo / README.md

Go to file




...

 TooYoungToDieHard Initial commit

Latest commit cƒ09bcb 1 hour ago  History

 1 contributor

2 lines (2 sloc) | 42 Bytes

RawBlame

TestRepo

Another how-to demo repository

Making Edits and Commits-Browser cont'd.


Make your edits

TestRepo / README.md Cancel

<> Edit file Preview changes Spaces 2 Soft wrap

```
1 # TestRepo
2 Another how-to demo repository
3 **This is an edit ready to be committed**
4
```

Scroll to the bottom of the page and click Commit changes

 Commit changes

Update README.md

Add an optional extended description...

☒ Commit directly to the testBranch branch.

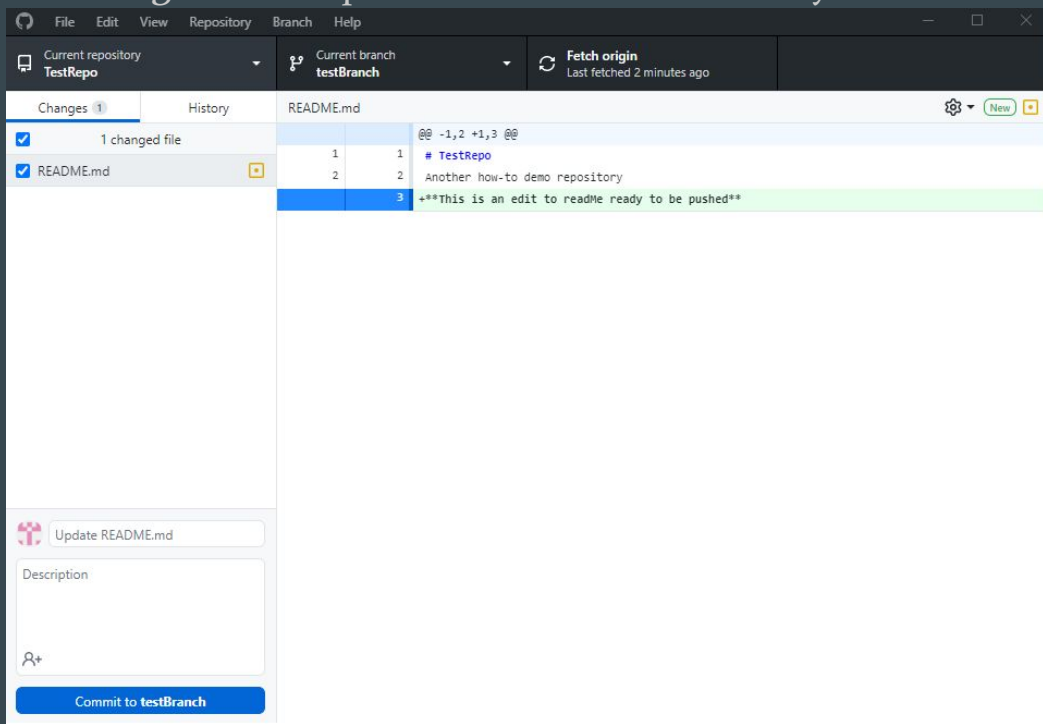
☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes Cancel

Your changes will only be saved to the README.md file in the new branch you created.

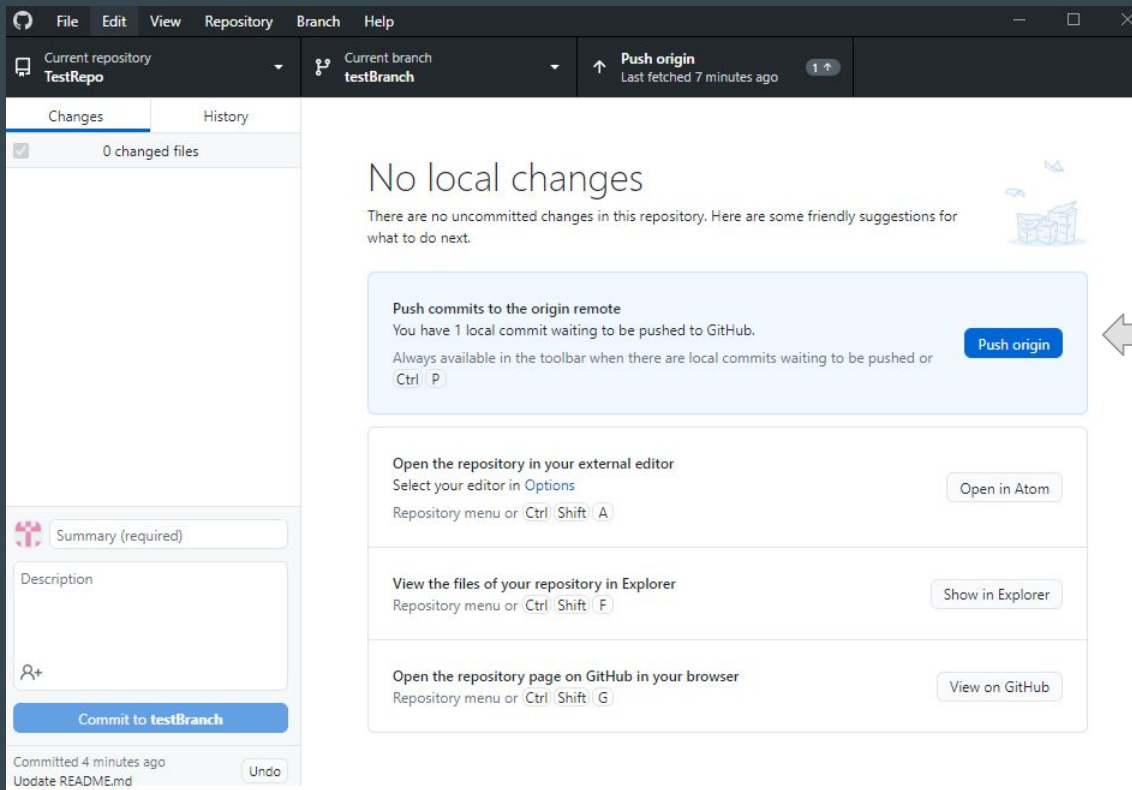
Making Edits and Commits-GitHub Desktop

- Commits can also be pushed using the GitHub Desktop app with a cloned repository
 - You can use whatever tools your prefer to make changes to code, text files, or other assets of your project, including Unity
- To do this, simply make the changes on your local machine, save the changes, return to GitHub Desktop, give your changes a description and click Commit to “yourBranchName”



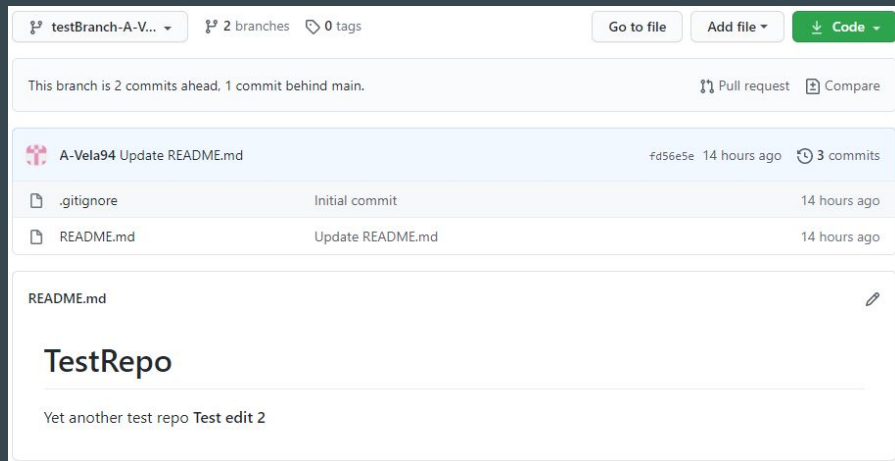
Making Edits and Commits-GitHub Desktop

- After making edits and committing them, your app screen will look like this:

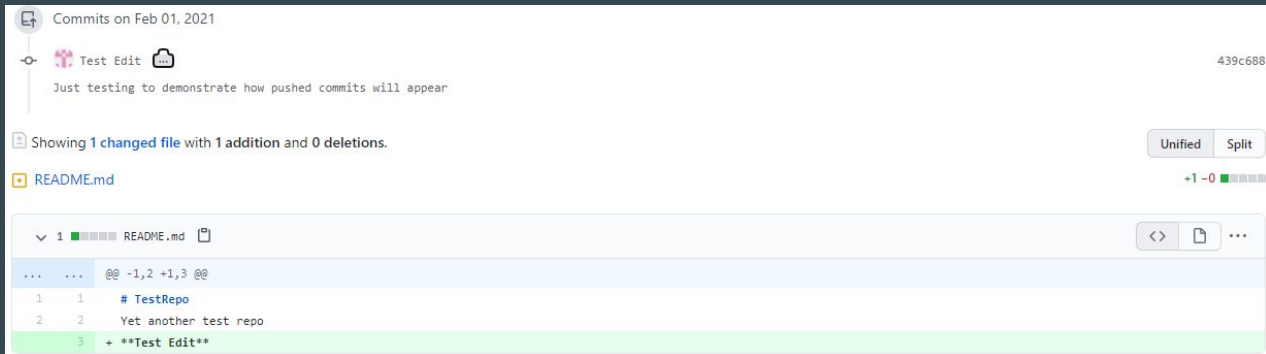


Viewing Commit History-Browser

- Commits will be logged and appear on both the repository page on github.com as well as GitHub Desktop.



Commit log, github.com



Viewing Commit History-GitHub Desktop

Commit log: GitHub Desktop - original

The screenshot shows the GitHub Desktop interface with the 'History' tab selected. The left sidebar shows the commit history for 'TestRepo' on the 'testBranch-A-Vela' branch. The main area displays the 'Initial commit' by A-Vela94, which includes a '.gitignore' file and a 'README.md' file. The commit message is 'Initial commit'.

Commit	Author	Files	Message
Initial commit	A-Vela94	2 changed files	Initial commit

Commit log: GitHub Desktop - changes made

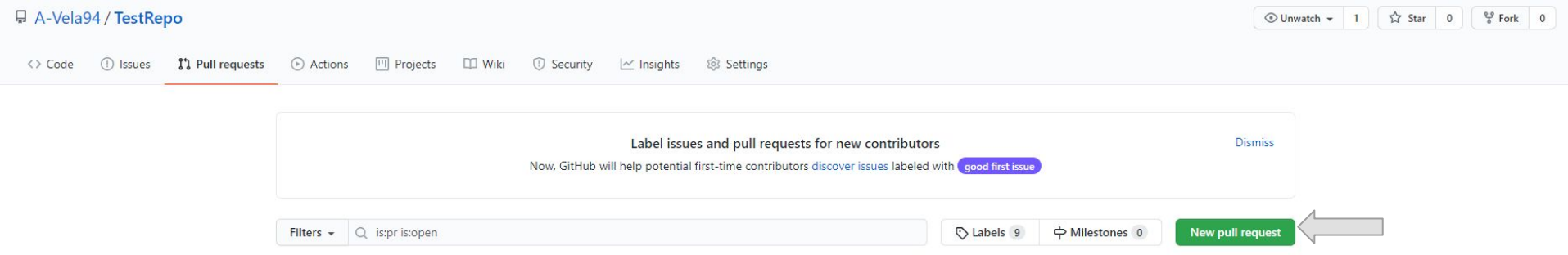
The screenshot shows the GitHub Desktop interface with the 'History' tab selected. The left sidebar shows the commit history for 'TestRepo' on the 'testBranch-A-Vela' branch. The main area displays the 'Test Edit' commit by A-Vela94, which includes a 'README.md' file. The commit message is 'Just testing to demonstrate how pushed commits will appear'.

Commit	Author	Files	Message
Test Edit	A-Vela94	1 changed file	Just testing to demonstrate how pushed commits will appear
Initial commit	A-Vela94	2 changed files	Initial commit

Pull Requests and Merging Branches

- The changes made to the README.md file in the testBranch in the previous slide only apply to that branch, for now.
- After changes are made to a branch off of main, GitHub allows developers to open what's called a *pull request*
 - A proposal to the rest of the collaborators working on a project requesting that your contribution be reviewed and merged back into either the main branch, or another branch depending on the nature of the contribution and the project goals.

Make a pull request on github.com



[Comprehensive documentation on pull requests: docs.github.com/en/github/collaborating-with-issues-and-pull-requests/about-pull-requests](https://docs.github.com/en/github/collaborating-with-issues-and-pull-requests/about-pull-requests)

Pull Request- Browser View

A-Vela94 / TestRepo

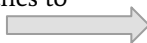
Unwatch 1 Star 0 Fork 0

<> Code ⓘ Issues 🚀 Pull requests ⚙️ Actions 📁 Projects 📖 Wiki 🛡️ Security 📊 Insights ⚙️ Settings

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

Comparing changes

Select branches to compare



base: main ← compare: testBranch-A-Vela ✓ Able to merge. These branches can be automatically merged.

🔗 Test branch a vela #2
Please look over my edits to see if they're good to merge. [View pull request](#)

Create another pull request to discuss and review the changes again. [Learn about pull requests](#) [Create pull request](#)

If satisfied with your changes, below, create your pull request



↔️ 2 commits 📄 1 file changed 💬 0 comments 👤 1 contributor

📅 Commits on Feb 01, 2021

- 🔗 Test Edit ... 439c688
- 🔗 Update README.md fd56e5e

📄 Showing 1 changed file with 1 addition and 0 deletions.

To view differences before and after the commit, select Split



Unified Split


```
1 # TestRepo
2 Yet another test repo
3 + **Test edit 2**
```

Pull Request-Browser View cont'd.

Give your pull request a title and a brief description of the proposed changes. Then create your request

Open a pull request


Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

 base: main

←

compare: testBranch-A-Vela

✓ Able to merge. These branches can be automatically merged.



Test branch a vela

Write


Preview

H B I

≡ <> 🔗

≡ ≡ ☑ @ 📎 ↶

Please look over my edits to see if they're good to merge.

Attach files by dragging & dropping, selecting or pasting them. 

Create pull request

Reviewers

No reviews

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Linked issues

Use **Closing keywords** in the description to automatically close issues

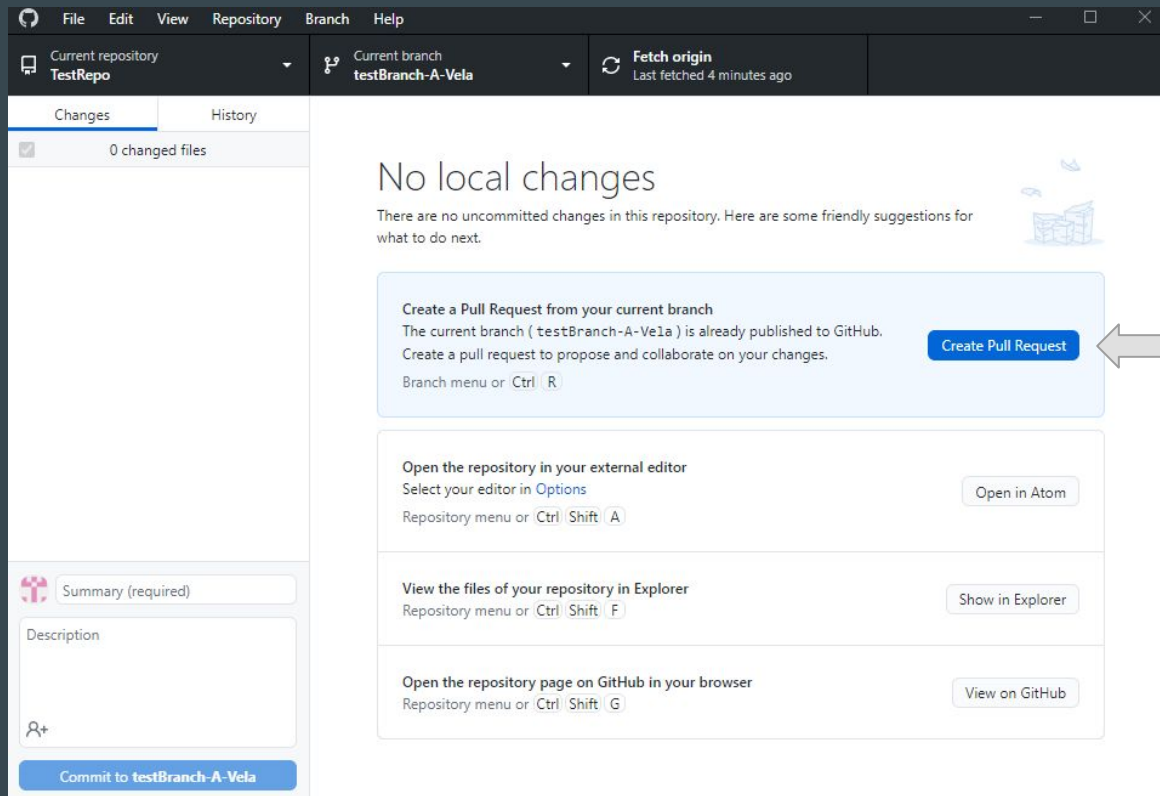
Helpful resources

[GitHub Community Guidelines](#)

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

Pull Request-GitHub Desktop View

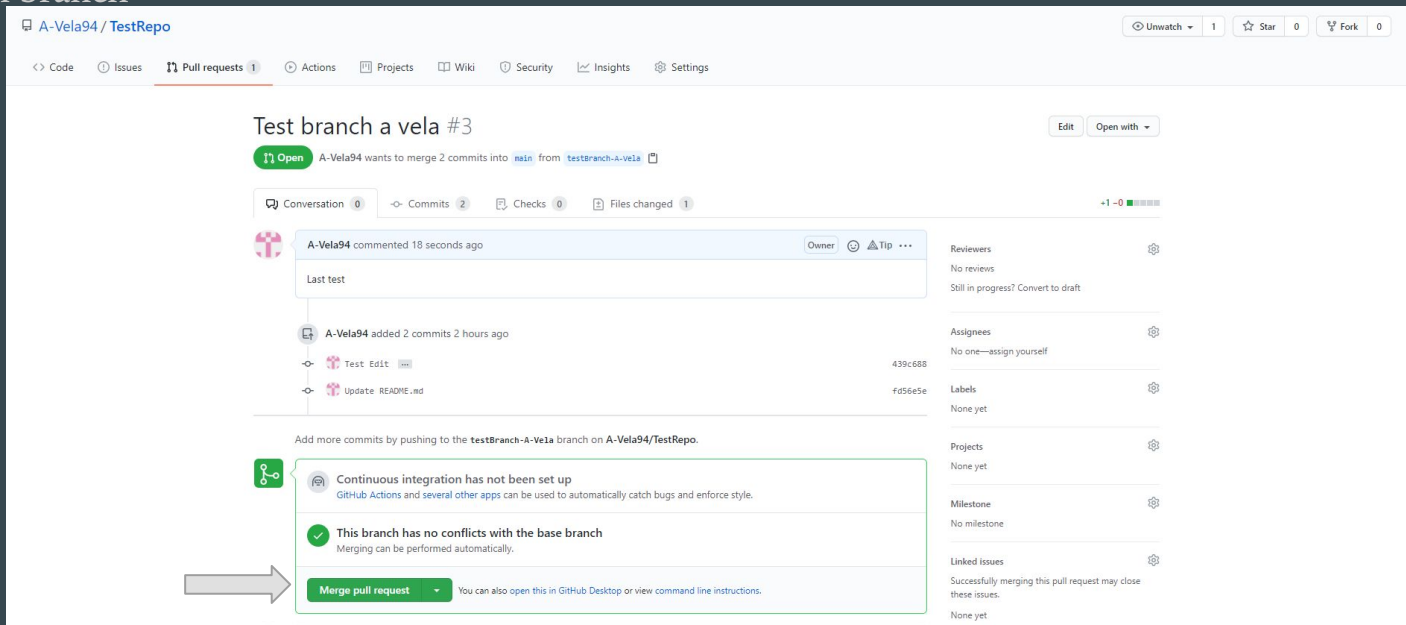
Pull requests can also be initiated from GitHub Desktop.



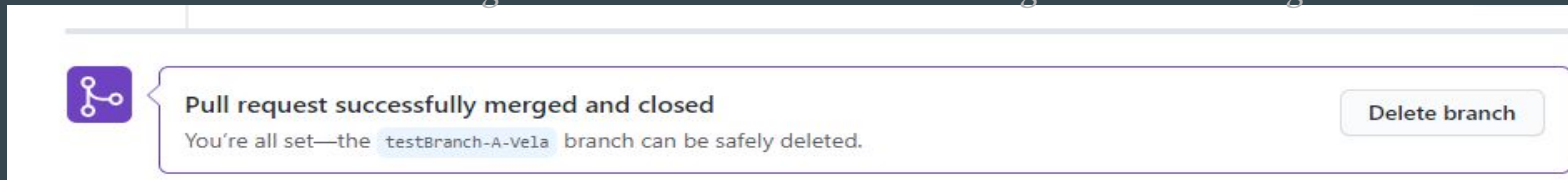
This will take you to the github.com to create your pull request as previously demonstrated

Merging Your Pull Request

- After all changes have been reviewed and approved, the new branch is ready to be merged back into the main branch



The branch used to make changes can now be deleted since its changes have been merged with main



Part 4: Disaster Recovery

Preparing for Disaster

- Back up your GitHub repository constantly
- Although you have access to each commit in your repository, you may accidentally override something when reverting changes
- Always a good idea to keep backups!
- Open your repository in a browser and download the ZIP file

File Edit View History Bookmarks Tools Help

Git & GitHub Tutorial for Begin X Google Slides X lotkey/guardians: 2D dungeon X +

https://github.com/lotkey/guardians

Search or jump to... Pull requests Issues Marketplace Explore

lotkey / guardians

Unwatch 2 Star 1 Fork 0

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 2 branches 0 tags

Go to file Add file Code

lotkey Update .gitignore

Assets project directory structure, gitignore

doc project directory structure, gitignore

.gitignore Update .gitignore

README.md Update README.md

Clone

HTTPS SSH GitHub CLI

https://github.com/lotkey/guardians.

Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop

Download ZIP

README.md

Guardians

Unity 2020.2.2f1

Clone URL: <https://github.com/lotkey/guardians.git>

2D dungeon crawler game by Pseudo Random Studios.

About

2D dungeon crawler game by Pseudo Random Studios.

Readme

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Contributors 2

lotkey

manicdisciple

README is a temporary placeholder. It will get the

Treating Disaster

- If you believe a specific commit is the source of your disaster, you can revert the commit
- Exercise caution whenever you revert your repository
- Good idea to make a separate backup before you revert
- In GitHub Desktop:
 - Navigate to the repository you wish to revert
 - Select “History”
 - Select the specific commit you wish to revert and right-click it
 - Select “Revert changes in commit”

FileEditViewRepositoryBranchHelp

Current repository
guardians

Current branch
main

Fetch origin
Last fetched just now

ChangesHistory

Update .gitignore

Select branch to compare...

lotkey dc95e70 ± 1 changed file ⚙

Update .gitignore
lotkey • 4h

Delete Guardians v0.1 directory
lotkey • 4h

Merge pull request #1 from lotkey/
lotkey • 4h

project directory structure, gitignore
Cade Disselkoe • 4h

Create .gitignore
lotkey • 4h

Update .gitignore
manicdisciple • 4h

Working on Animations
synthchris666 • 8h

.gitignore

Revert changes in commit

Create Tag...

Copy SHA

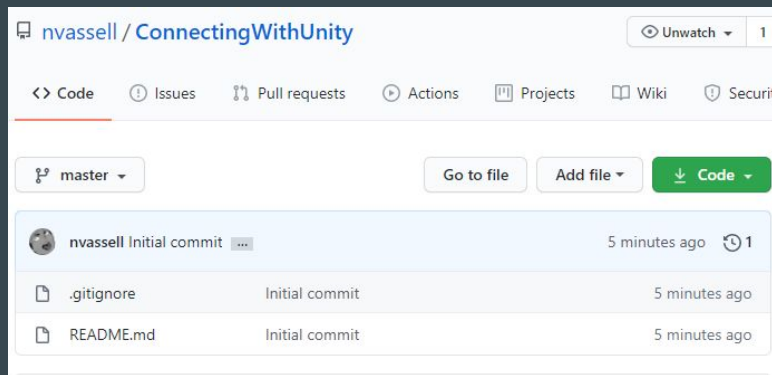
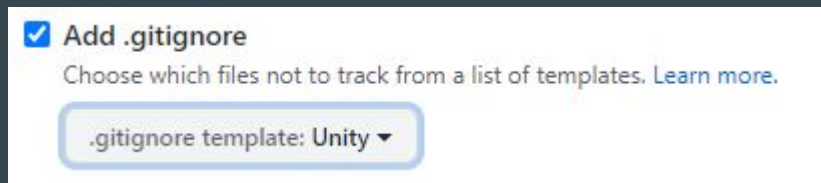
View on GitHub

		@@ -2,13 +2,13 @@
2	2	#
3	3	# Get latest from https://github.com/git
4	4	#
5		-[Ll]ibrary/
6		-[Tt]emp/
7		-[Oo]bj/
8		-[Bb]uild/
9		-[Bb]uilds/
10		-[Ll]ogs/
11		-[Mm]emoryCaptures/
	5	+/[Ll]ibrary/
	6	+/[Tt]emp/
	7	+/[Oo]bj/
	8	+/[Bb]uild/
	9	+/[Bb]uilds/
	10	+/[Ll]ogs/

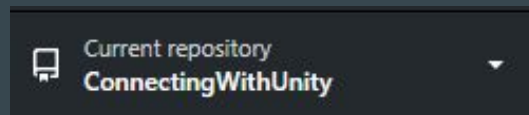
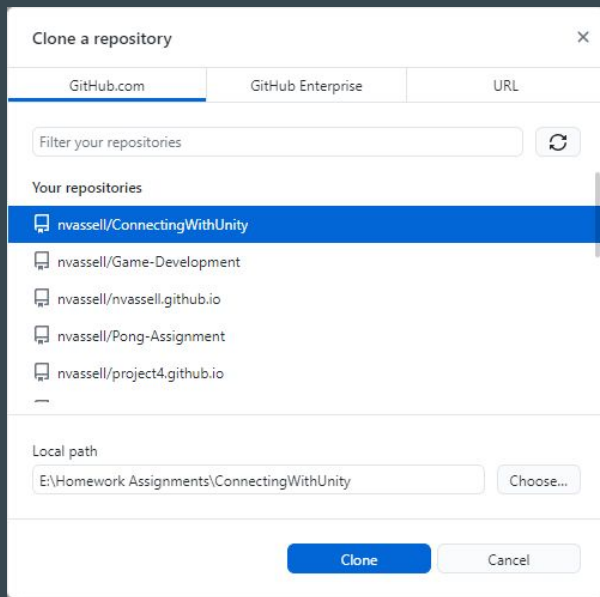
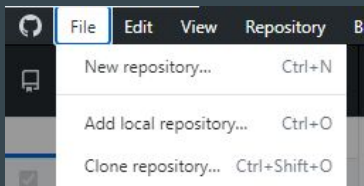
Part 5: Unity and Github

Getting a Unity project connected to GitHub

- When creating your GitHub repo, make sure that the “Add .gitignore” check is active with the “.gitignore template: Unity” so our .gitignore contains the correct items.

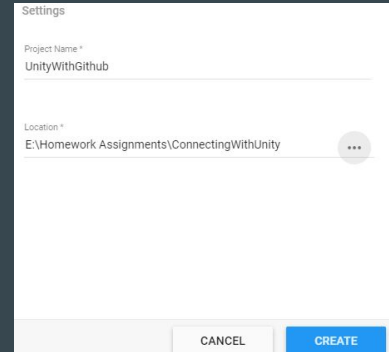
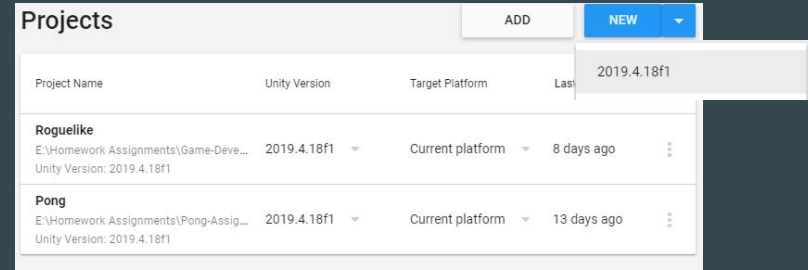


- Now that we have our project repo let's get it cloned to our GitHub Desktop:



With Unity Hub: Let's get this project created in our repo.

- Click: New
 - And your version of Unity
- When Creating our project, make sure that the location is pointing towards our newly created local repo.



- Looking at our GitHub Desktop we can see a new 8098 changed files (in our case)

<input checked="" type="checkbox"/>	8098 changed files	
<input checked="" type="checkbox"/>	UnityWithGithub\vs...\settings.json	
<input checked="" type="checkbox"/>	UnityWithGithub\As...\Scenes.meta	
<input checked="" type="checkbox"/>	UnityWithGith...\SampleScene.unity	
<input checked="" type="checkbox"/>	UnityWi...\SampleScene.unity.meta	
<input checked="" type="checkbox"/>	UnityWithGi...\AnnotationManager	
<input checked="" type="checkbox"/>	Unit...\project-dependencies.graph	
<input checked="" type="checkbox"/>	UnityWithGithub\Library\ArtifactDB	

- If we make the following changes to .gitignore: we can down the number of changes to a more reasonable amount.

```
-/[Ll]ibrary/  
-/[Tt]emp/  
-/[Oo]bj/  
-/[Bb]uild/  
-/[Bb]uilds/  
-/[Ll]ogs/  
-/[Mm]emoryCaptures/  
+[Ll]ibrary/  
+[Tt]emp/  
+[Oo]bj/  
+[Bb]uild/  
+[Bb]uilds/  
+[Ll]ogs/  
+[Mm]emoryCaptures/
```

<input checked="" type="checkbox"/>	27 changed files	
<input checked="" type="checkbox"/>	.gitignore	
<input checked="" type="checkbox"/>	UnityWithGithub\vs...\settings.json	
<input checked="" type="checkbox"/>	UnityWithGithub\As...\Scenes.meta	
<input checked="" type="checkbox"/>	UnityWithGith...\SampleScene.unity	
<input checked="" type="checkbox"/>	UnityWi...\SampleScene.unity.meta	