MVP
IL

Getting started with Git and GitHub for the reluctant IT Pro

Thomas Vochten

Agenda

1. Introduction to git
2. The tools you need to get started
3. Getting basic stuff done with git
4. Where to go next?

Thomas Vochten

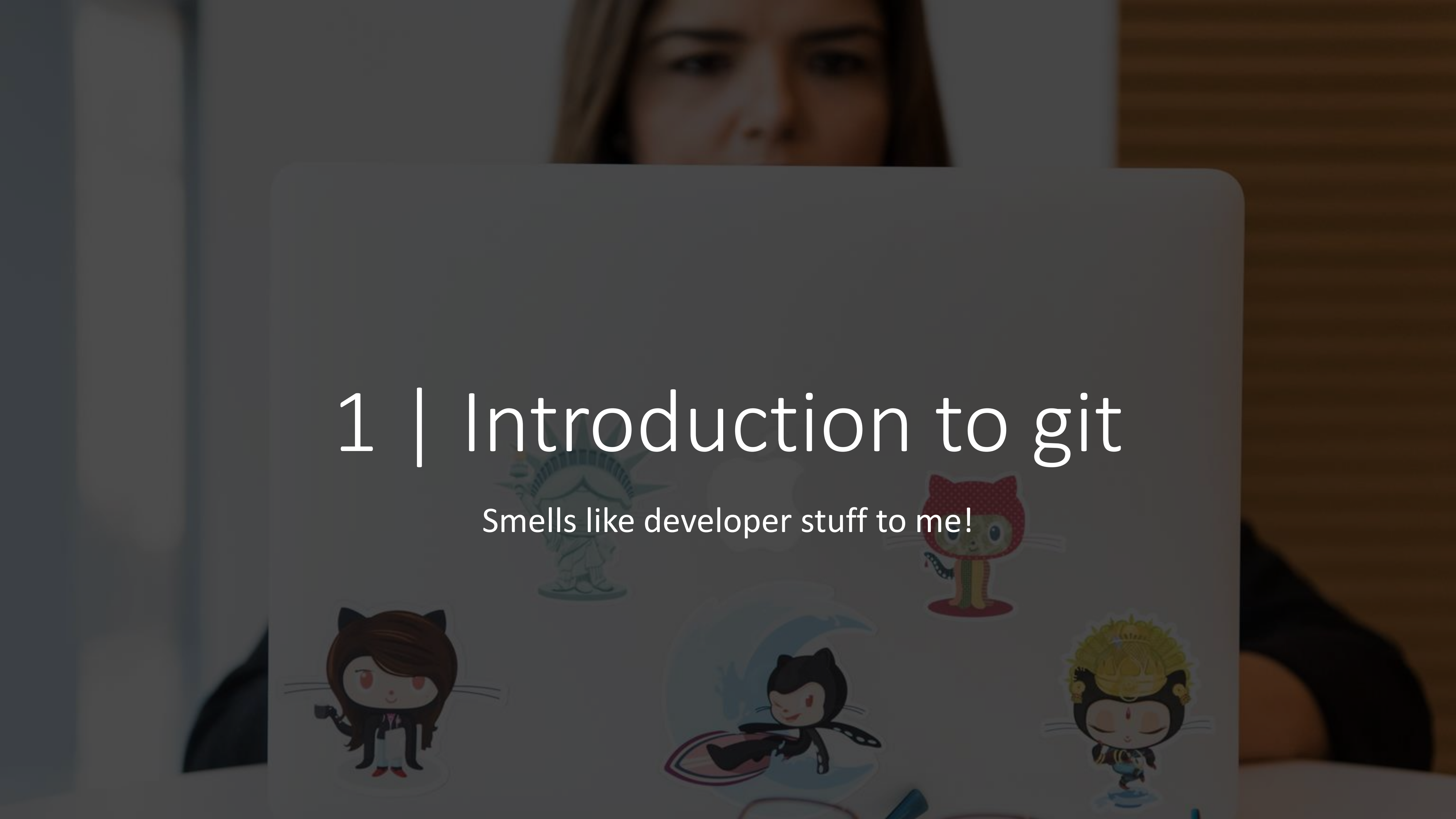
@thomasvochten
https://thomasvochten.com
mail@thomasvochten.com

MVP Microsoft® Most Valuable Professional

BIWUG

CRONOS GROEP

☁ Technology Evangelist ☕ 💻
#Microsoft365 #Azure #CommunityRocks



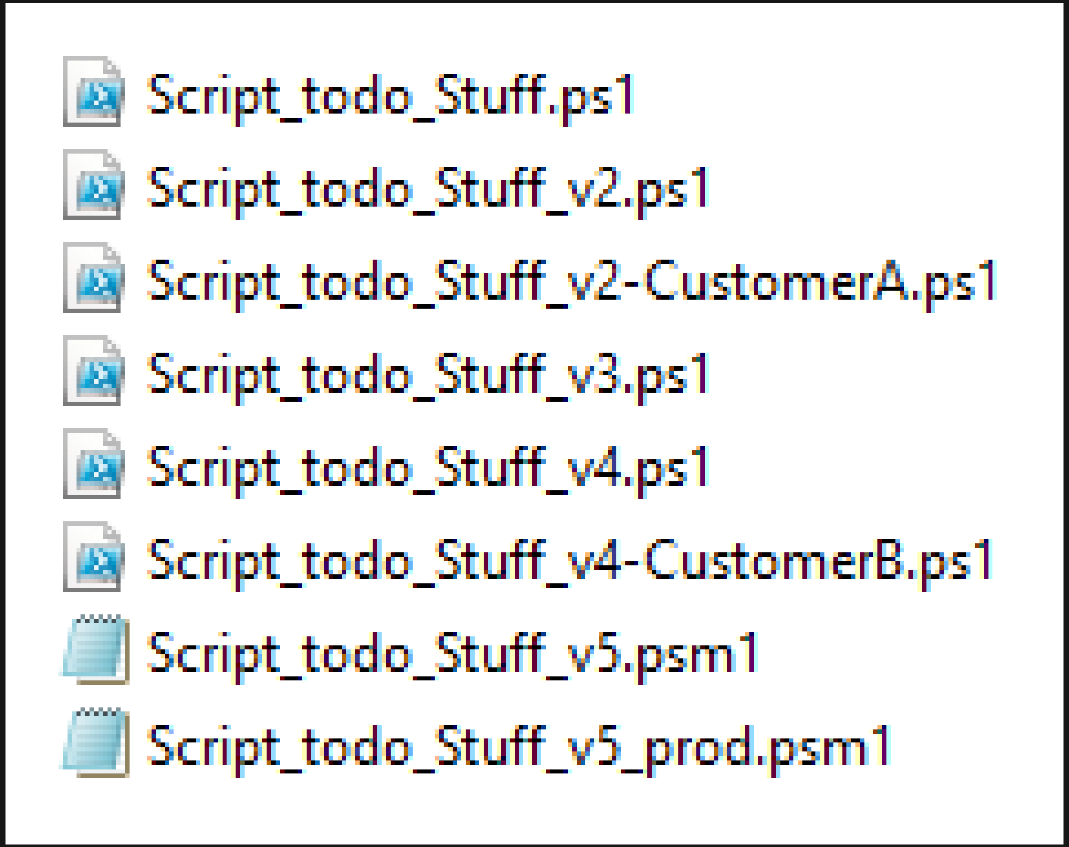
1 | Introduction to git

Smells like developer stuff to me!

Why exactly would I care?

Where and how do you store your stuff?

- Computer
- File share
- OneNote
- Cloud storage
- …



Script_todo_Stuff.ps1
Script_todo_Stuff_v2.ps1
Script_todo_Stuff_v2-CustomerA.ps1
Script_todo_Stuff_v3.ps1
Script_todo_Stuff_v4.ps1
Script_todo_Stuff_v4-CustomerB.ps1
Script_todo_Stuff_v5.psm1
Script_todo_Stuff_v5_prod.psm1

Source control to the rescue

- Like versioning in SharePoint, but different!
- Keep different versions of your files
- Ability to go back if you messed up
- Work together with other people
- Work on new stuff while keeping the existing files intact
- Make sure your new stuff stays up to date when you change the existing files
- …

git

Git is a free and open source distributed version control system

https://thvo.me/git

What about GitHub then?

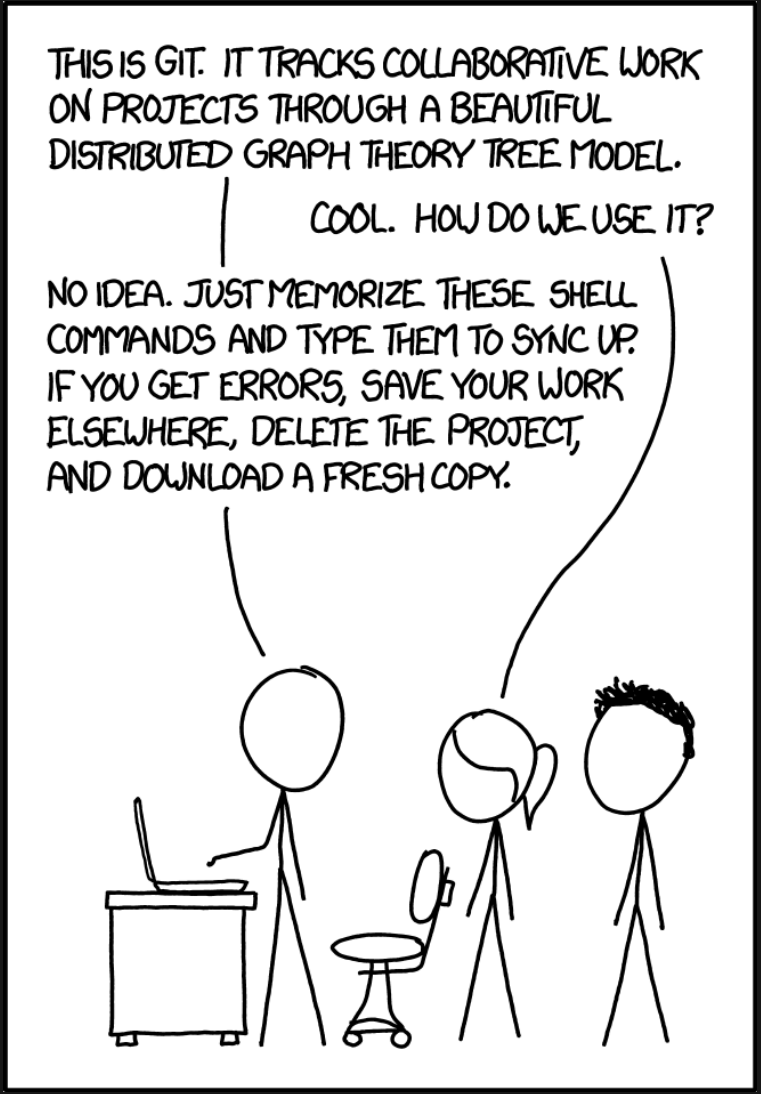
- Hosted git
- Acquired by Microsoft in 2018
- The de facto standard

Some alternatives

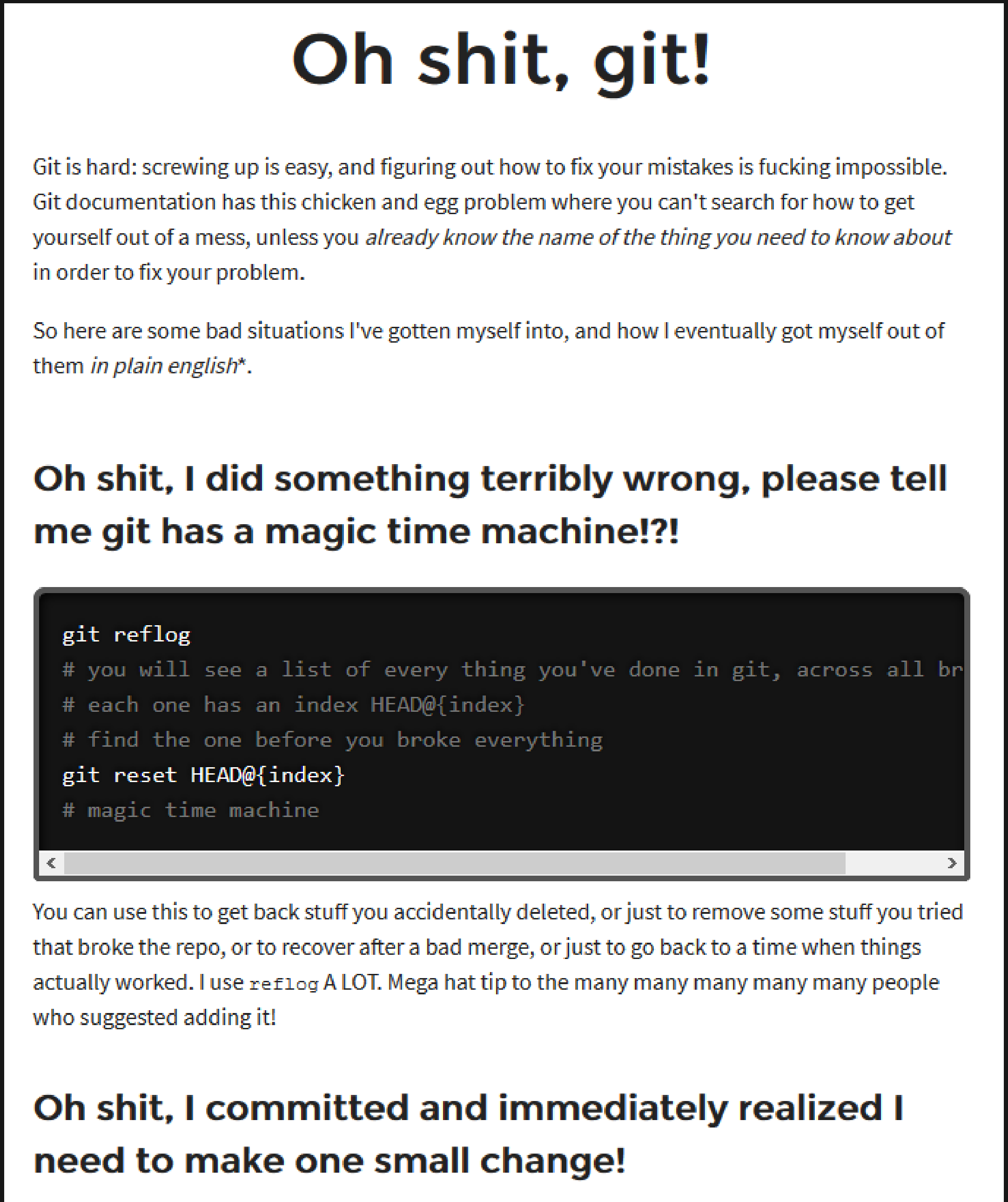
Bitbucket

Gitlab

Azure Repos


THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.
COOL. HOW DO WE USE IT?
NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.

https://ohshitgit.com/



Oh shit, git!

Git is hard: screwing up is easy, and figuring out how to fix your mistakes is fucking impossible. Git documentation has this chicken and egg problem where you can't search for how to get yourself out of a mess, unless you *already know the name of the thing you need to know about* in order to fix your problem.

So here are some bad situations I've gotten myself into, and how I eventually got myself out of them *in plain english**.

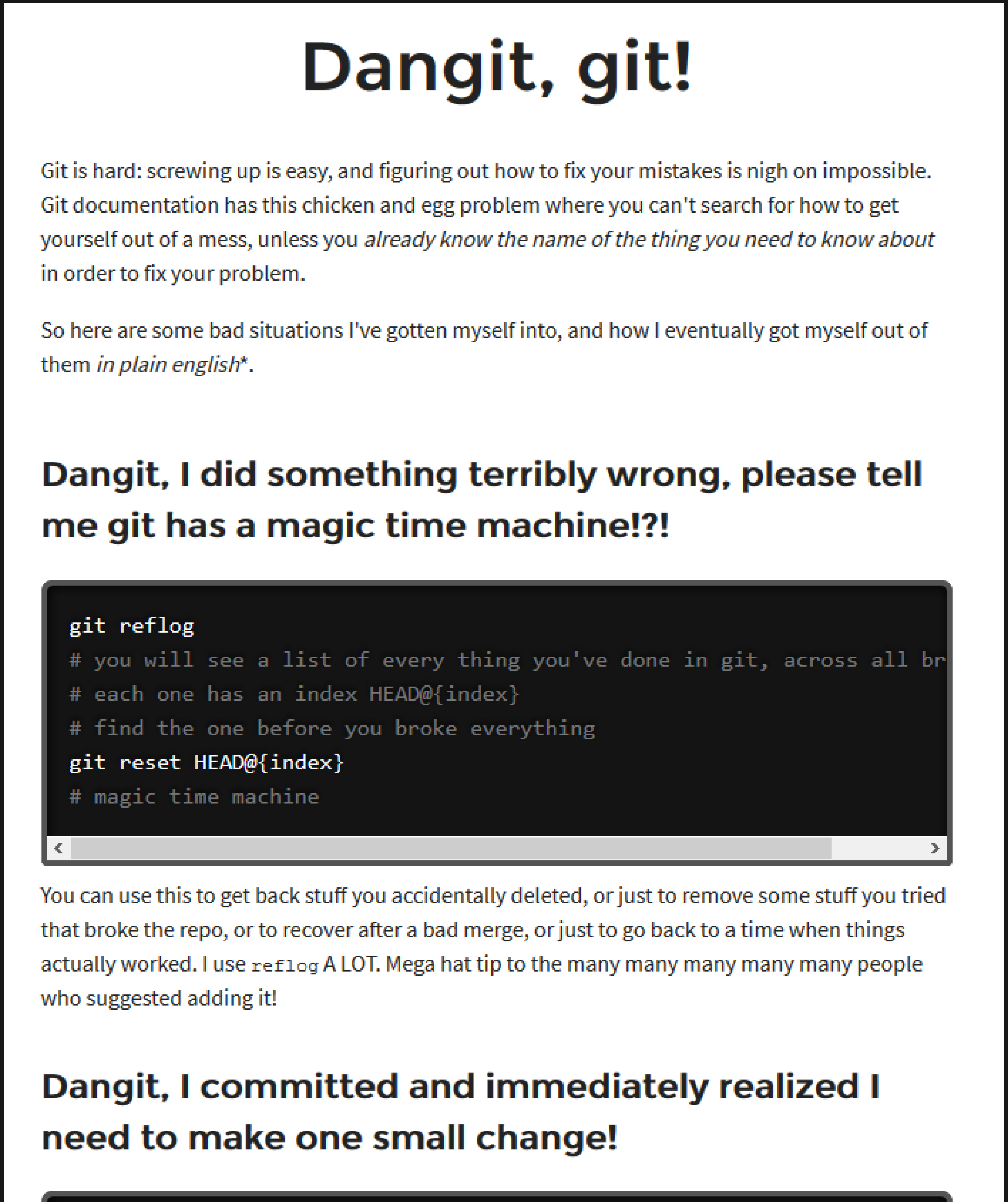
Oh shit, I did something terribly wrong, please tell me git has a magic time machine!?!

```
git reflog
# you will see a list of every thing you've done in git, across all br
# each one has an index HEAD@{index}
# find the one before you broke everything
git reset HEAD@{index}
# magic time machine
```

You can use this to get back stuff you accidentally deleted, or just to remove some stuff you tried that broke the repo, or to recover after a bad merge, or just to go back to a time when things actually worked. I use `reflog` A LOT. Mega hat tip to the many many many many many people who suggested adding it!

Oh shit, I committed and immediately realized I need to make one small change!

https://dangitgit.com/



Dangit, git!

Git is hard: screwing up is easy, and figuring out how to fix your mistakes is nigh on impossible. Git documentation has this chicken and egg problem where you can't search for how to get yourself out of a mess, unless you *already know the name of the thing you need to know about* in order to fix your problem.

So here are some bad situations I've gotten myself into, and how I eventually got myself out of them *in plain english**.

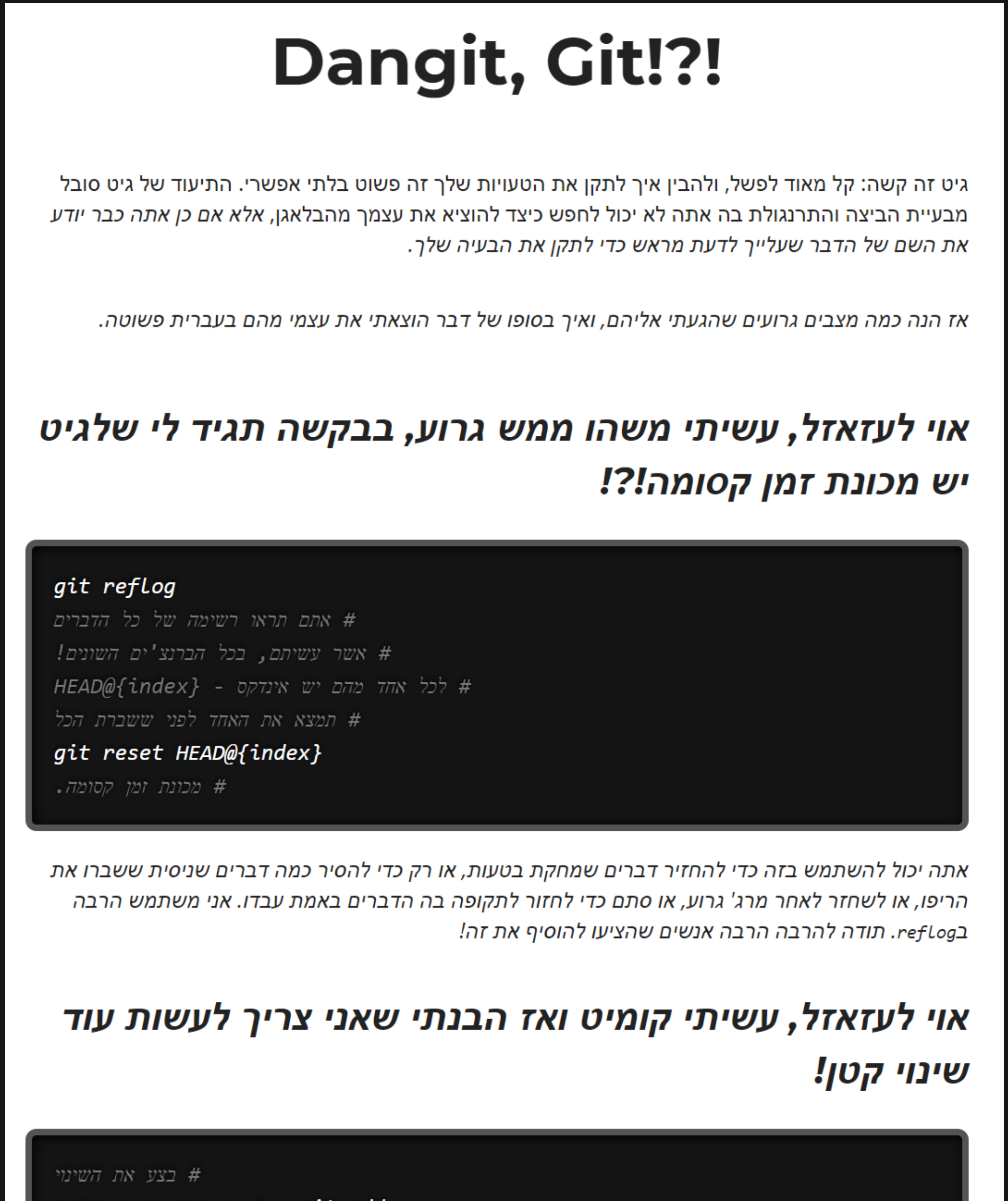
Dangit, I did something terribly wrong, please tell me git has a magic time machine!?!

```
git reflog
# you will see a list of every thing you've done in git, across all br
# each one has an index HEAD@{index}
# find the one before you broke everything
git reset HEAD@{index}
# magic time machine
```

You can use this to get back stuff you accidentally deleted, or just to remove some stuff you tried that broke the repo, or to recover after a bad merge, or just to go back to a time when things actually worked. I use `reflog` A LOT. Mega hat tip to the many many many many many people who suggested adding it!

Dangit, I committed and immediately realized I need to make one small change!

https://dangitgit.com/he



Dangit, Git!?!

גיט זה קשה: קל מאוד לפשל, ולהבין איך לתקן את הטעויות שלך זה פשוט בלתי אפשרי. התיעוד של גיט סובל מבעיית הביצה והתרנגולת בה אתה לא יכול לחפש כיצד להוציא את עצמך מהבלאגן, *אלא אם כן אתה כבר יודע את השם של הדבר שעלייך לדעת מראש כדי לתקן את הבעיה שלך.*

אז הנה כמה מצבים גרועים שהגעתי אליהם, ואיך בסופו של דבר הוצאתי את עצמי מהם בעברית פשוטה.

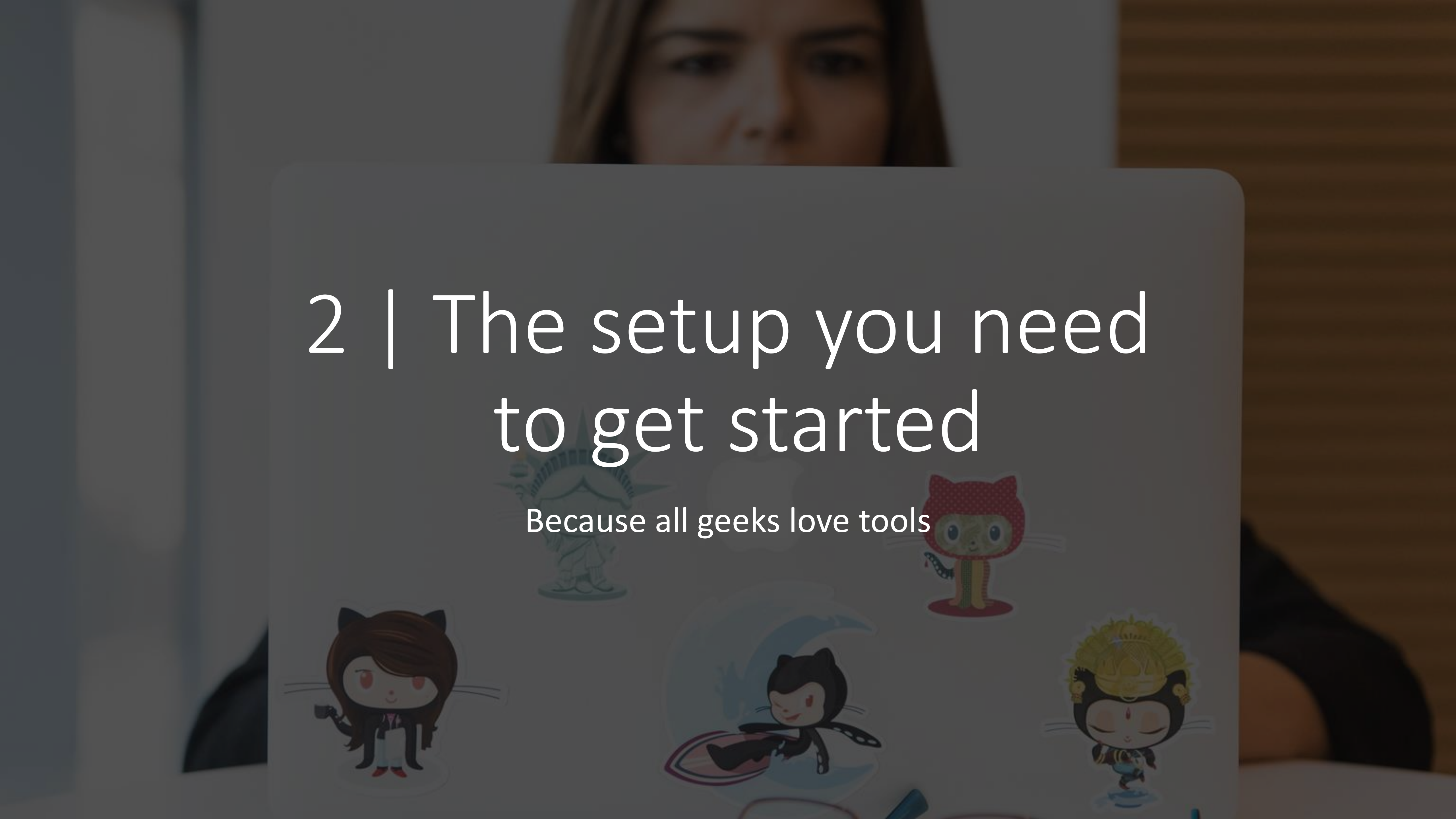
אוי לעזאזל, עשיתי משהו ממש גרוע, בבקשה תגיד לי שלגיט יש מכונת זמן קסומה!?!

```
git reflog
# אתם תראו רשימה של כל הדברים
# אשר עשיתם, בכל הברנצ'ים השונים!
# לכל אחד מהם יש אינדקס - HEAD@{index}
# תמצא את האחד לפני ששברת הכל
git reset HEAD@{index}
# מכונת זמן קסומה.
```

אתה יכול להשתמש בזה כדי להחזיר דברים שמחקת בטעות, או רק כדי להסיר כמה דברים שניסית ששברו את הריפו, או לשחזר לאחר מרג' גרוע, או סתם כדי לחזור לתקופה בה הדברים באמת עבדו. אני משתמש הרבה ב`reflog`. תודה להרבה הרבה אנשים שהציעו להוסיף את זה!

אוי לעזאזל, עשיתי קומיט ואז הבנתי שאני צריך לעשות עוד שינוי קטן!

```
# בצע את השינוי
```



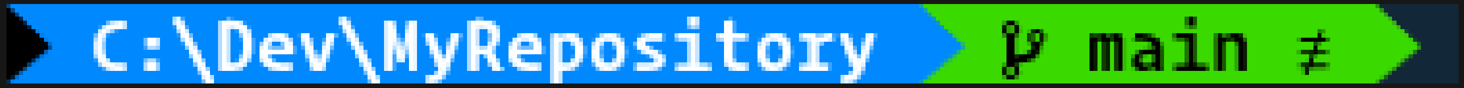
2 | The setup you need to get started

Because all geeks love tools

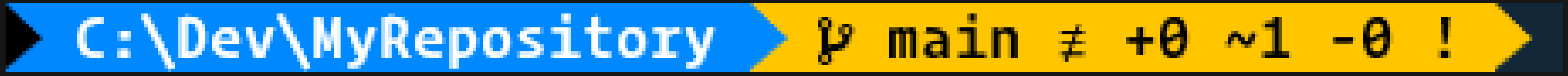
Windows Terminal

Visual Studio Code

Upgrade your command prompt



```
C:\Dev\MyRepository  main ≢
```



```
C:\Dev\MyRepository  main ≢ +0 ~1 -0 !
```

Upgrade your command prompt

- Install Windows Terminal
- Install a Powerline-capable font, e.g. from NerdFonts:

  https://thvo.me/nerdfonts

- Setup git & Powerline integration in PowerShell

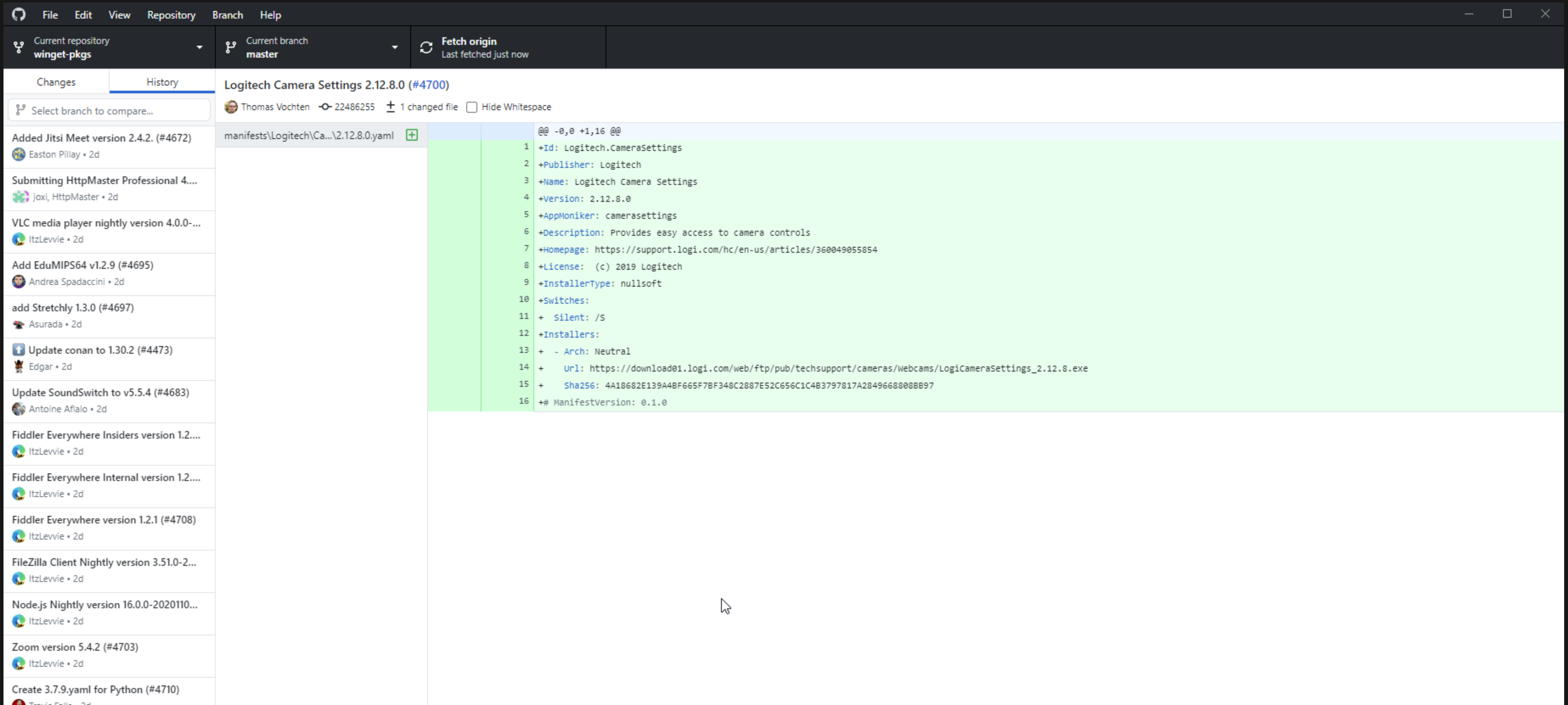
  ```
  Install-module posh-git
  Install-Module oh-my-posh
  Set-Theme Paradox
  ```

- Modify your Windows Terminal profile settings to use the font

  ```
  "fontFace": “FiraCode Nerd Font"
  ```

GitHub Desktop

https://thvo.me/ghdesktop



Added Jitsi Meet version 2.4.2. (#4672)
Easton Pillay • 2d

Submitting HttpMaster Professional 4....
joxi, HttpMaster • 2d

VLC media player nightly version 4.0.0-...
ItzLevvie • 2d

Add EduMIPS64 v1.2.9 (#4695)
Andrea Spadaccini • 2d

add Stretchly 1.3.0 (#4697)
Asurada • 2d

Update conan to 1.30.2 (#4473)
Edgar • 2d

Update SoundSwitch to v5.5.4 (#4683)
Antoine Aflalo • 2d

Fiddler Everywhere Insiders version 1.2....
ItzLevvie • 2d

Fiddler Everywhere Internal version 1.2....
ItzLevvie • 2d

Fiddler Everywhere version 1.2.1 (#4708)
ItzLevvie • 2d

FileZilla Client Nightly version 3.51.0-2...
ItzLevvie • 2d

Node.js Nightly version 16.0.0-2020110...
ItzLevvie • 2d

Zoom version 5.4.2 (#4703)
ItzLevvie • 2d

Create 3.7.9.yaml for Python (#4710)

Logitech Camera Settings 2.12.8.0 (#4700)

Thomas Vochten · 22486255 · 1 changed file

manifests\Logitech\Ca...\2.12.8.0.yaml

```
@@ -0,0 +1,16 @@
+Id: Logitech.CameraSettings
+Publisher: Logitech
+Name: Logitech Camera Settings
+Version: 2.12.8.0
+AppMoniker: camerasettings
+Description: Provides easy access to camera controls
+Homepage: https://support.logi.com/hc/en-us/articles/360049055854
+License:  (c) 2019 Logitech
+InstallerType: nullsoft
+Switches:
+  Silent: /S
+Installers:
+  - Arch: Neutral
+    Url: https://download01.logi.com/web/ftp/pub/techsupport/cameras/Webcams/LogiCameraSettings_2.12.8.exe
+    Sha256: 4A18682E139A4BF665F7BF348C2887E52C656C1C4B3797817A2849668808BB97
+# ManifestVersion: 0.1.0
```

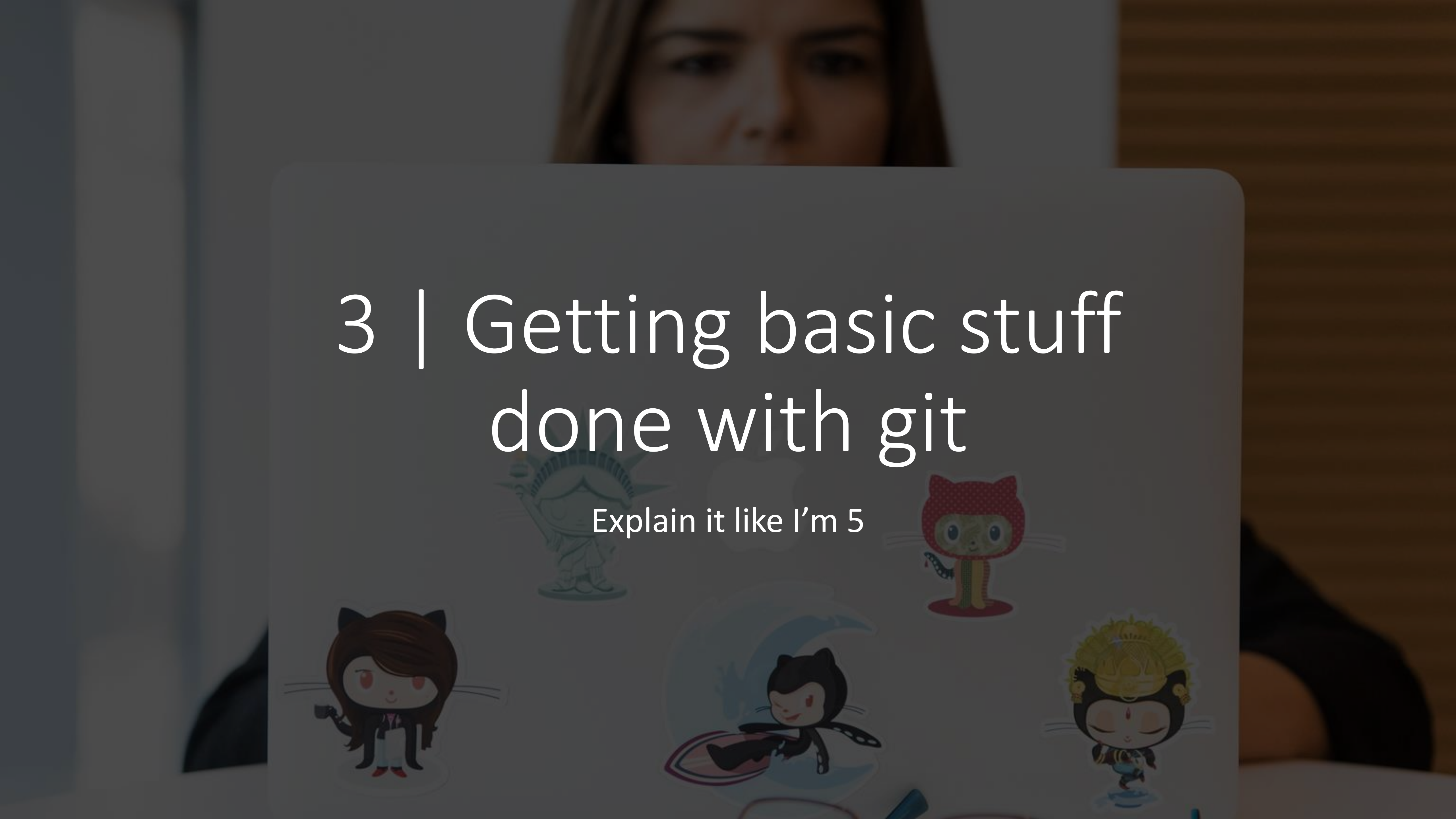
Integration with Visual Studio Code

- Supports the most common git commands by default
- Use the command line for more advanced scenarios
- Useful extensions:

GitLens
https://thvo.me/gitlens

Git History
https://thvo.me/githistory

GitHub Pull Requests & Issues
https://thvo.me/vscodegithub



3 | Getting basic stuff done with git

Explain it like I’m 5

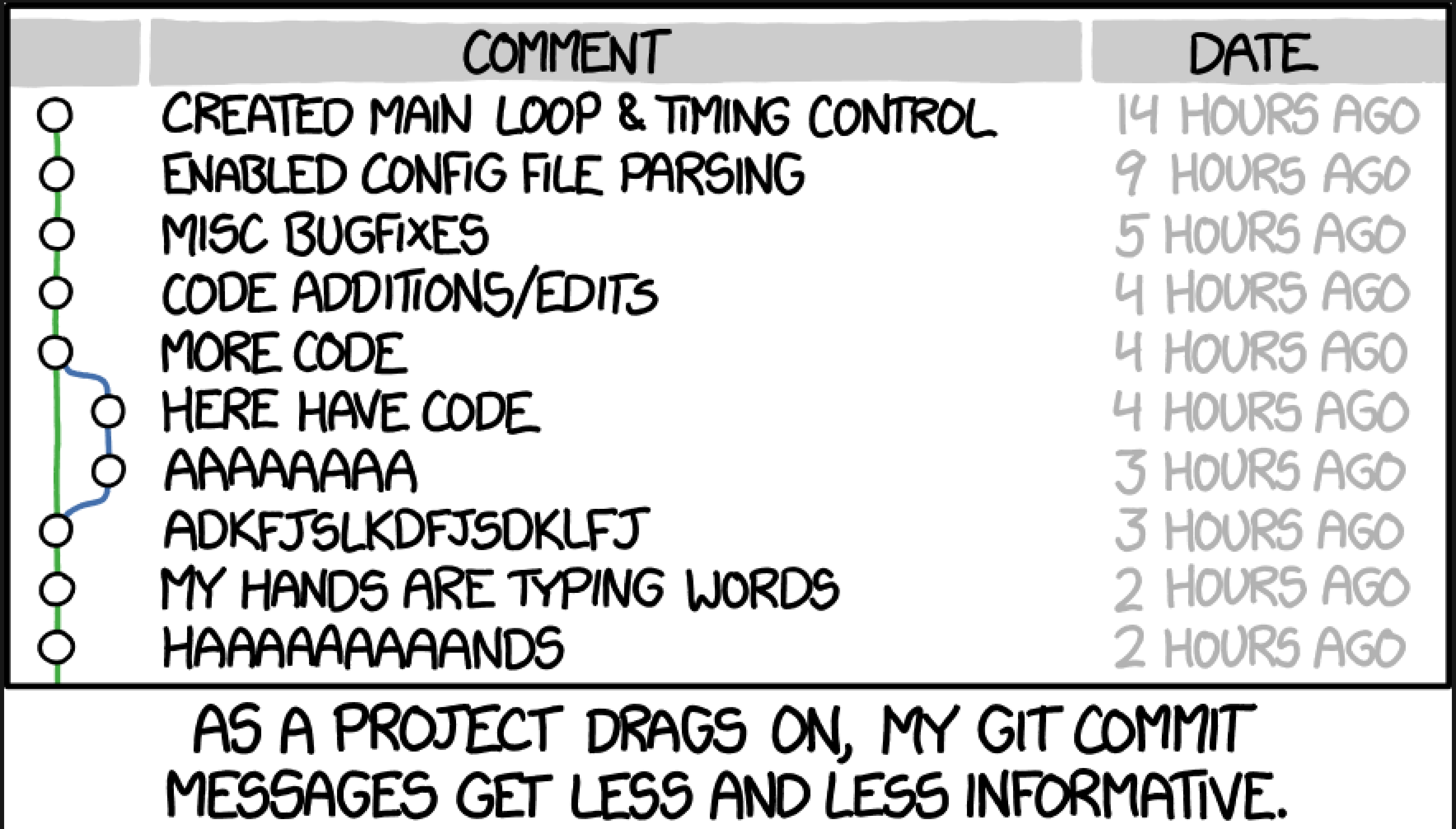
Essential Lingo

- Just “downloading” code from the internet (cloning)
- Creating your own version of existing code (forking)
- Creating a separate area to work on stuff (branching)
- Checking in your changes (committing)
- Move your changes to a remote repository (pushing)
- Getting changes from the remote repository (pulling)
- Integrating changes (merging)

ⓘ Many practices are just conventions

Demo - Common Operations

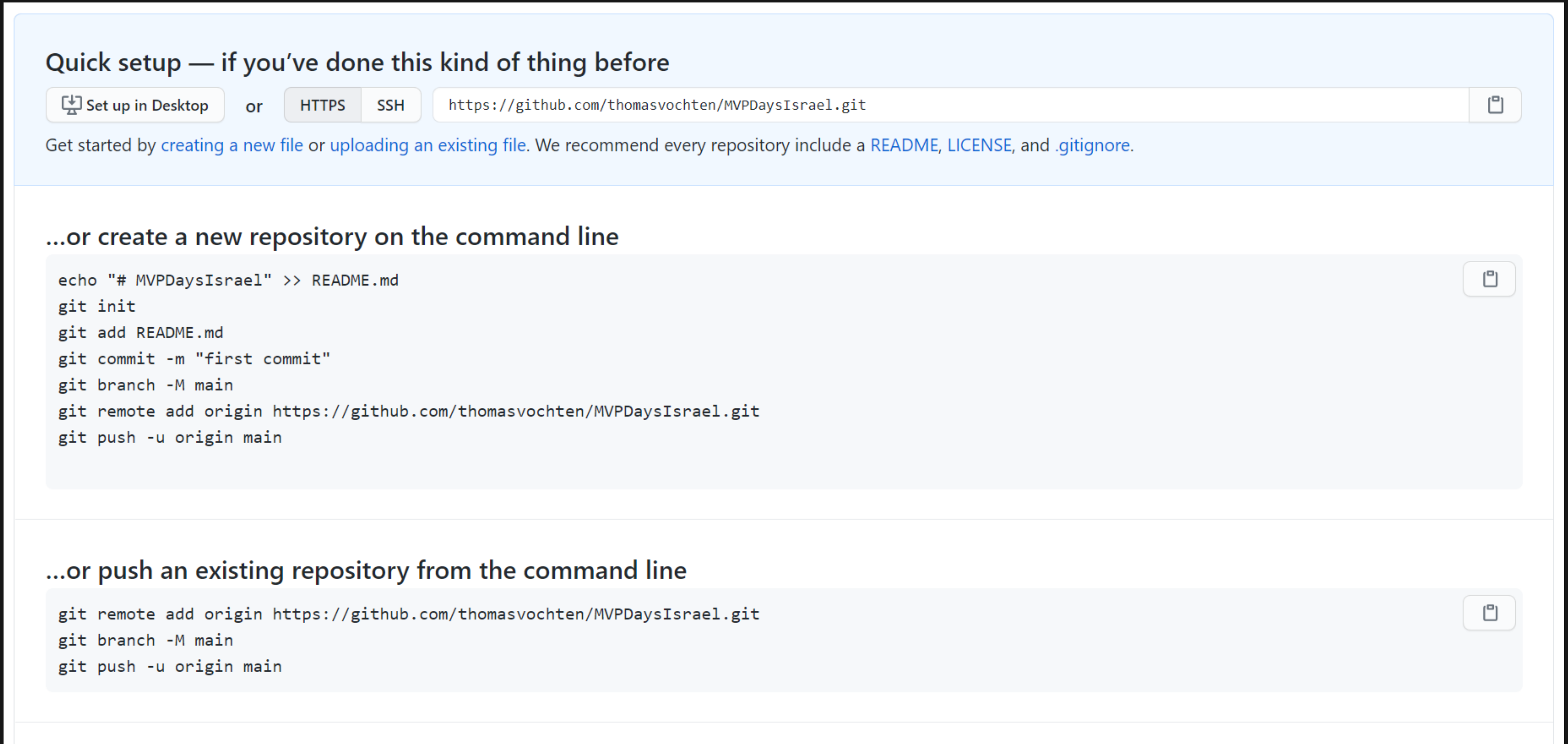
git clone, git status, git commit, git add, git push, git pull



	COMMENT	DATE
○	CREATED MAIN LOOP & TIMING CONTROL	14 HOURS AGO
○	ENABLED CONFIG FILE PARSING	9 HOURS AGO
○	MISC BUGFIXES	5 HOURS AGO
○	CODE ADDITIONS/EDITS	4 HOURS AGO
○	MORE CODE	4 HOURS AGO
○	HERE HAVE CODE	4 HOURS AGO
○	AAAAAAAA	3 HOURS AGO
○	ADKFJSLKDFJSDKLFJ	3 HOURS AGO
○	MY HANDS ARE TYPING WORDS	2 HOURS AGO
○	HAAAAAAAANDS	2 HOURS AGO

AS A PROJECT DRAGS ON, MY GIT COMMIT MESSAGES GET LESS AND LESS INFORMATIVE.

Initializing a (remote) repository



Quick setup — if you've done this kind of thing before

```
https://github.com/thomasvochten/MVPDaysIsrael.git
```

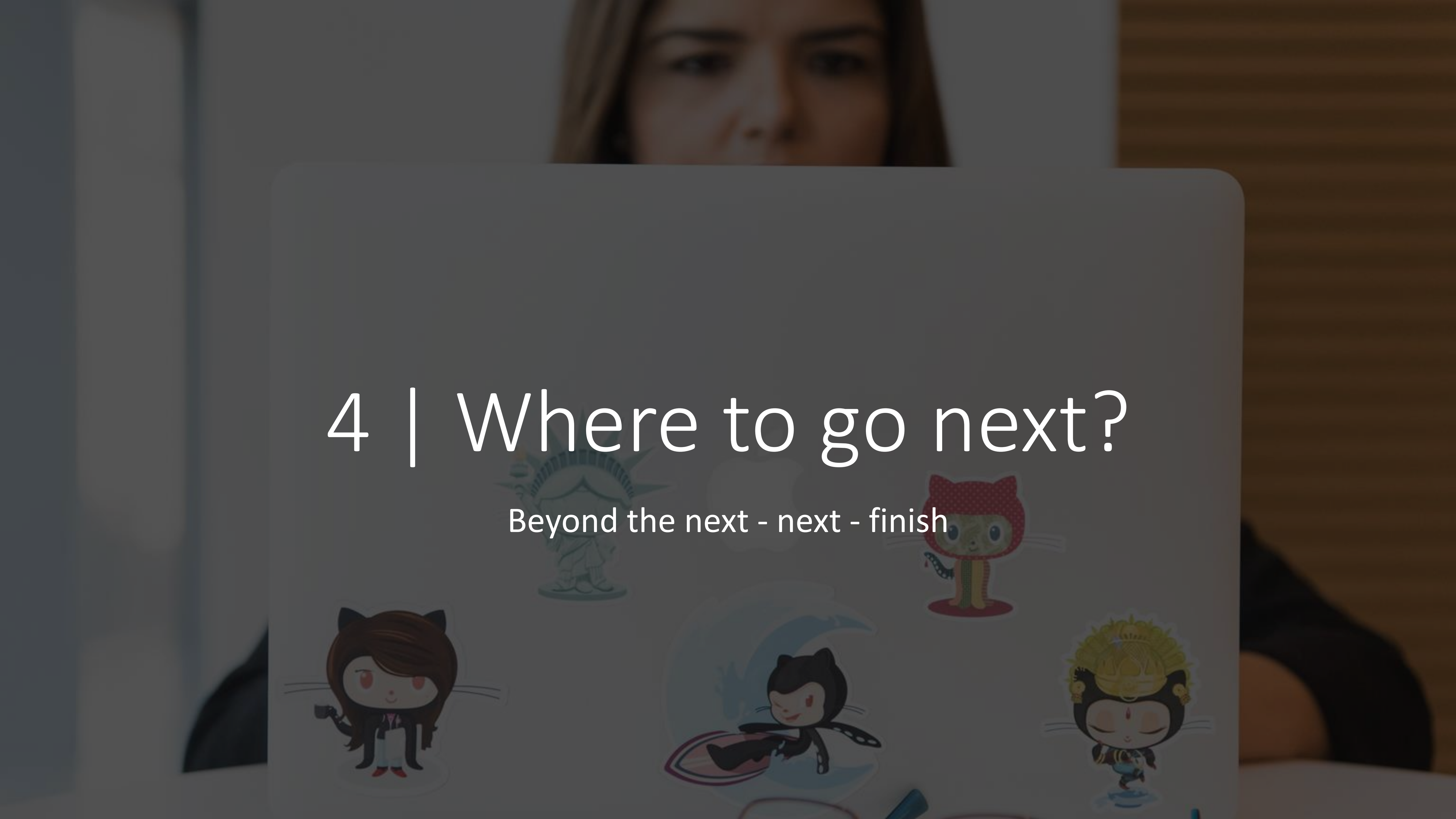
Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# MVPDaysIsrael" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/thomasvochten/MVPDaysIsrael.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/thomasvochten/MVPDaysIsrael.git
git branch -M main
git push -u origin main
```



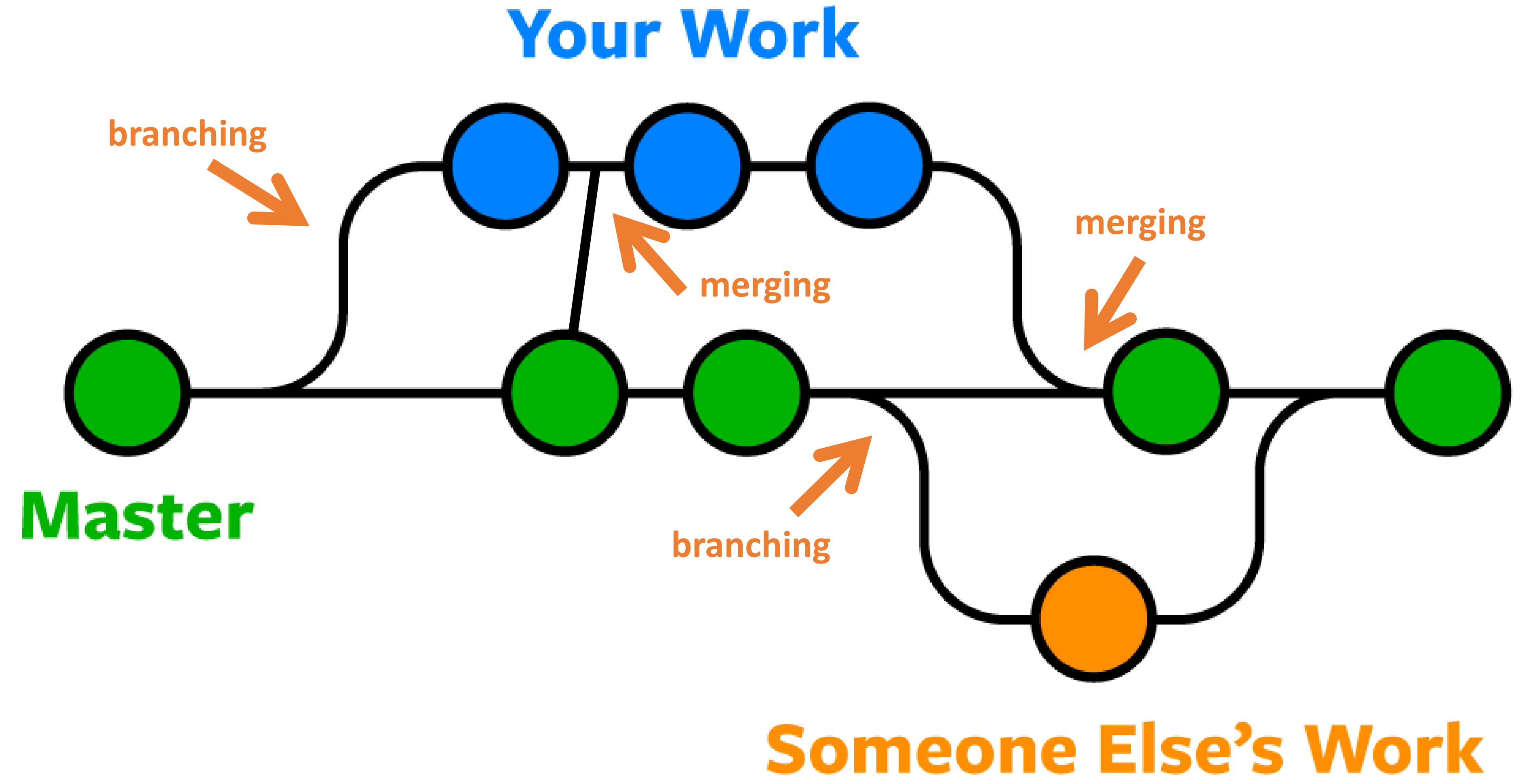
4 | Where to go next?

Beyond the next - next - finish

What to learn next

- Working with remote repositories
- Pull / Push mechanism
- Branching
- Forking
- Merging
- Pull Requests

Branching & Merging

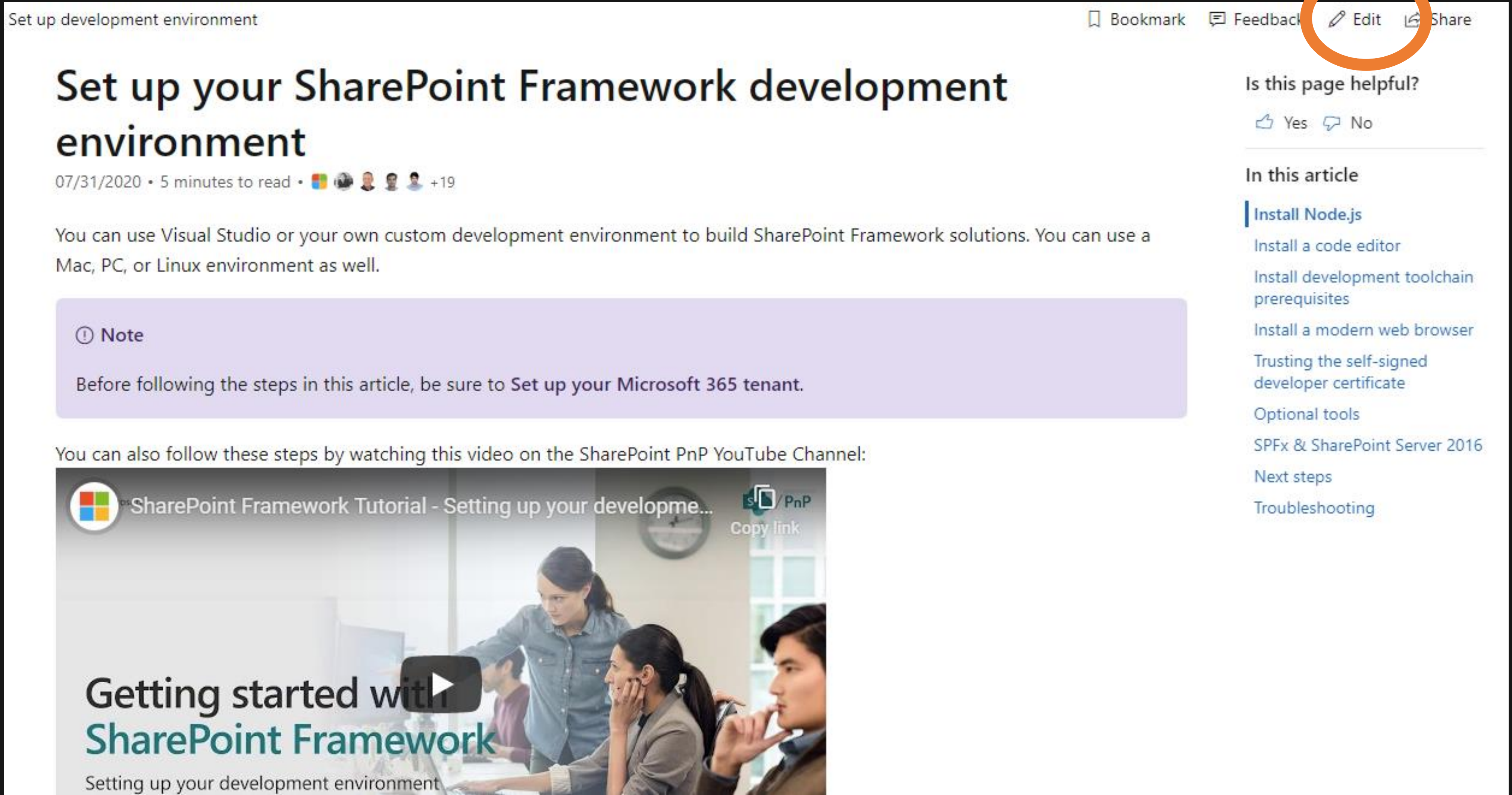


Sharing is caring!

Everyone can contribute.
Share the things you’re proud of:

- PowerShell scripts & modules
- Samples and tutorials
- Documentation
- …

Contributing to docs.microsoft.com



Set up your SharePoint Framework development environment

07/31/2020 • 5 minutes to read • +19

You can use Visual Studio or your own custom development environment to build SharePoint Framework solutions. You can use a Mac, PC, or Linux environment as well.

> Note
>
> Before following the steps in this article, be sure to **Set up your Microsoft 365 tenant.**

You can also follow these steps by watching this video on the SharePoint PnP YouTube Channel:

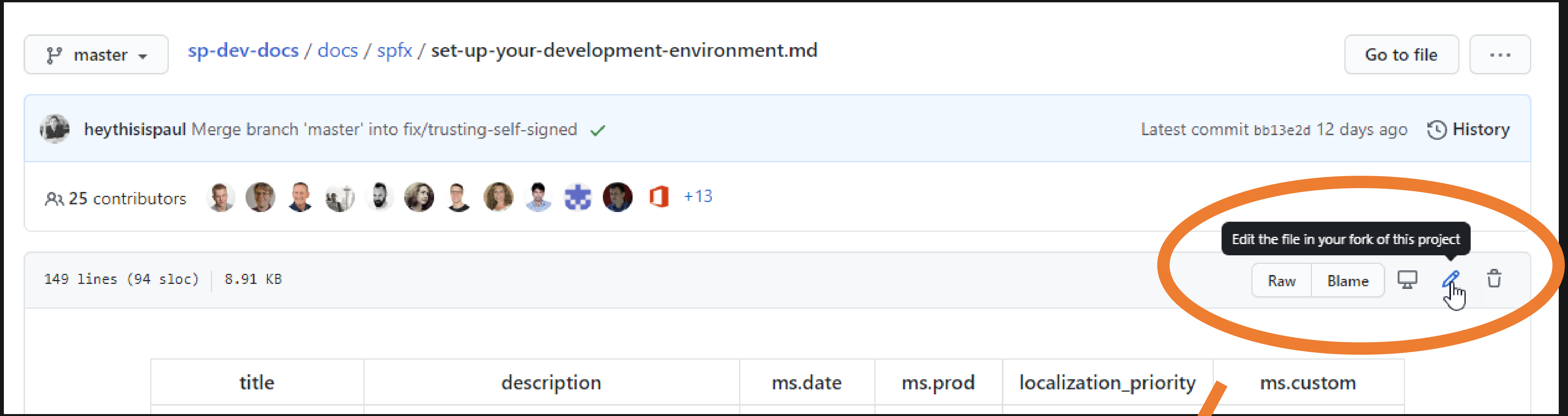
SharePoint Framework Tutorial - Setting up your developme...
PnP
Copy link
Getting started with
SharePoint Framework
Setting up your development environment

Is this page helpful?

Yes No

In this article

Install Node.js
Install a code editor
Install development toolchain prerequisites
Install a modern web browser
Trusting the self-signed developer certificate
Optional tools
SPFx & SharePoint Server 2016
Next steps
Troubleshooting



sp-dev-docs / docs / spfx / set-up-your-development-environment.md

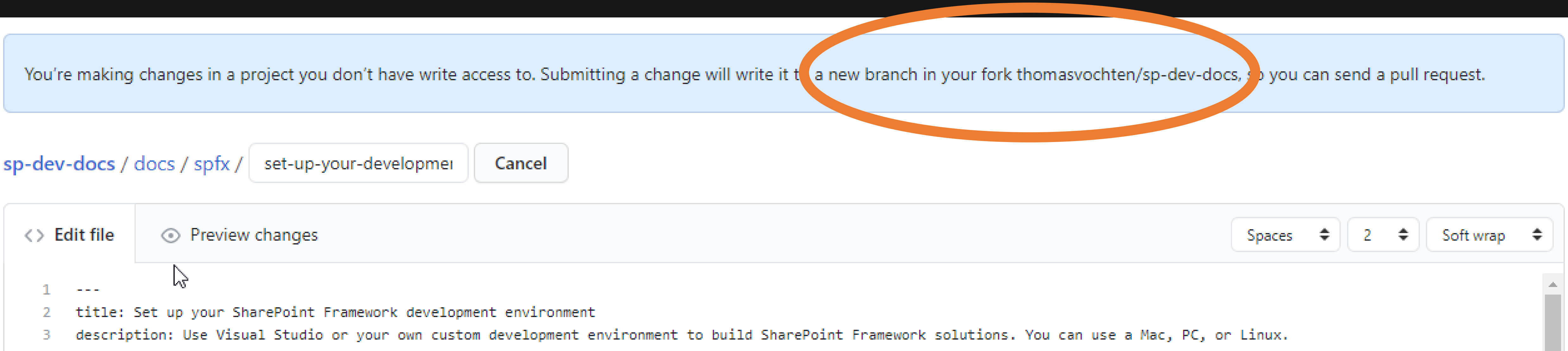
heythisispaul Merge branch 'master' into fix/trusting-self-signed ✓

Latest commit bb13e2d 12 days ago

Edit the file in your fork of this project

149 lines (94 sloc) | 8.91 KB

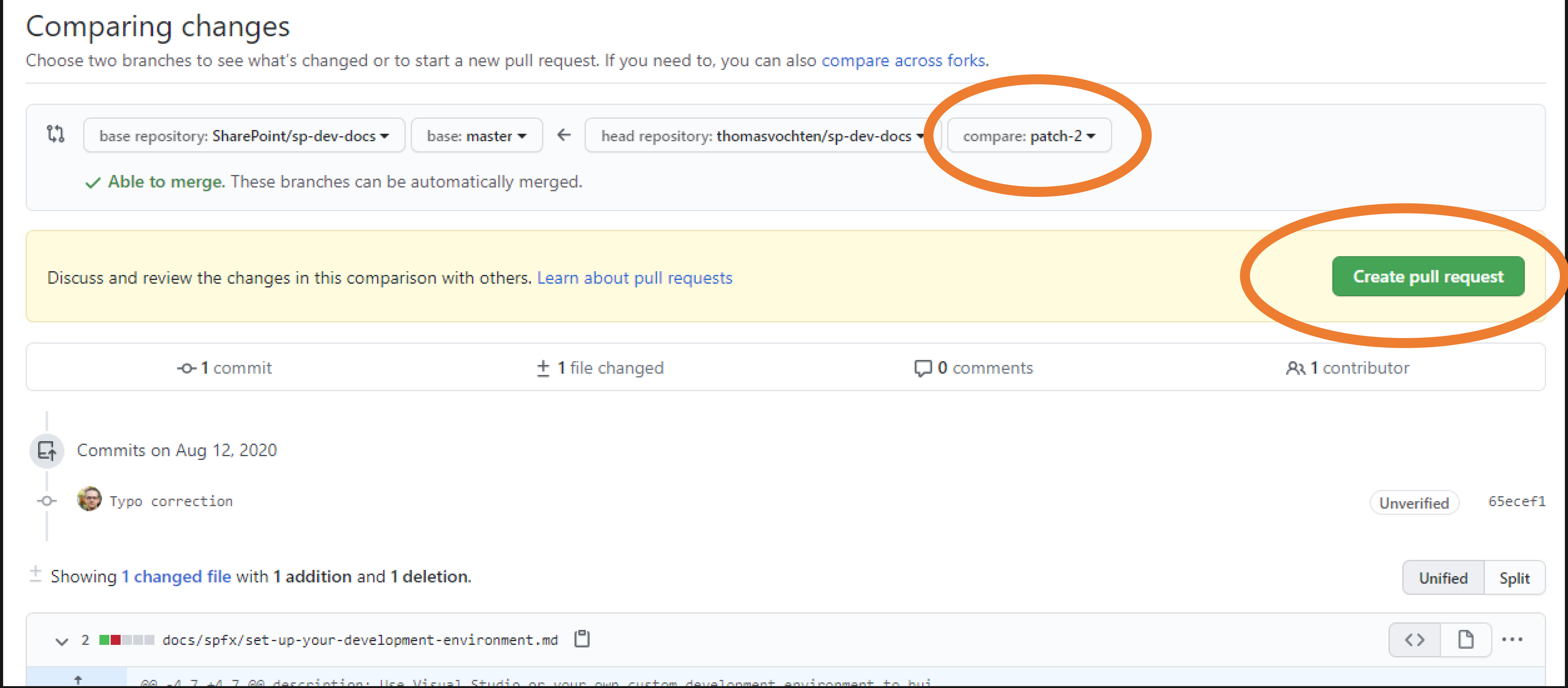
title	description	ms.date	ms.prod	localization_priority	ms.custom



You're making changes in a project you don't have write access to. Submitting a change will write it to a new branch in your fork thomasvochten/sp-dev-docs, so you can send a pull request.

sp-dev-docs / docs / spfx / set-up-your-developme

```
---
title: Set up your SharePoint Framework development environment
description: Use Visual Studio or your own custom development environment to build SharePoint Framework solutions. You can use a Mac, PC, or Linux.
```



Comparing changes

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.

base repository: SharePoint/sp-dev-docs ▾ base: master ▾ ← head repository: thomasvochten/sp-dev-docs ▾ compare: patch-2 ▾

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

Discuss and review the changes in this comparison with others. Learn about pull requests

1 commit · 1 file changed · 0 comments · 1 contributor

Commits on Aug 12, 2020

```
Typo correction
```

Unverified 65ecef1

Showing 1 changed file with 1 addition and 1 deletion.

```
2 docs/spfx/set-up-your-development-environment.md
@@ -4,7 +4,7 @@ description: Use Visual Studio or your own custom development environment to bui
```

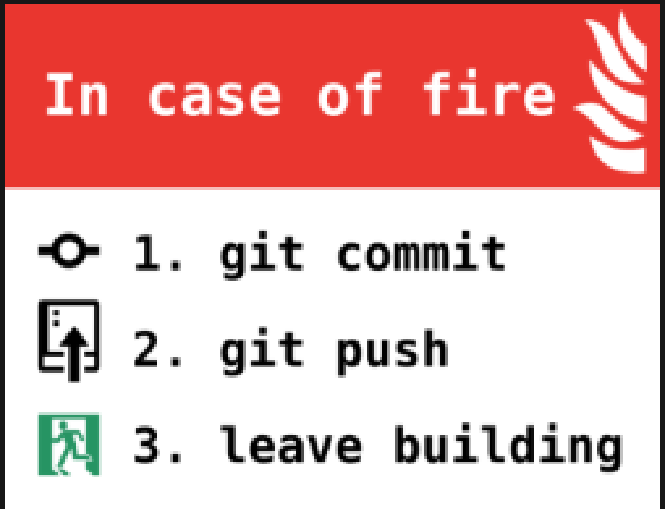
Typical workflow when contributing

- Fork the repo you want to contribute to
- Create a branch for your change
- Make & test your changes
- Create a pull request
- Wait for the pull request to be accepted
- Remove your fork

⚠ Learn about pull requests etiquette…

Key takeways

- Getting started with git is relatively simple
- The learning curve is quite dramatic once you go beyond the basics
- Don’t be afraid of the command line, but also:
- Take advantage of tools like VSCode or GitHub Desktop to help you
- Commit & pull as often as you can
- Don’t stop here!



In case of fire

1. git commit
2. git push
3. leave building

Additional resources

Video series by Scott Hanselman

https://thvo.me/scotthagit

Git documentation
https://thvo.me/gitdocs

Set up Powerline in Windows Terminal

https://thvo.me/wintermpowerline

@thomasvochten

THANK YOU

Download session materials