```
tt()
```

Schedule

1. Web architecture

   1. Bash & Terminal
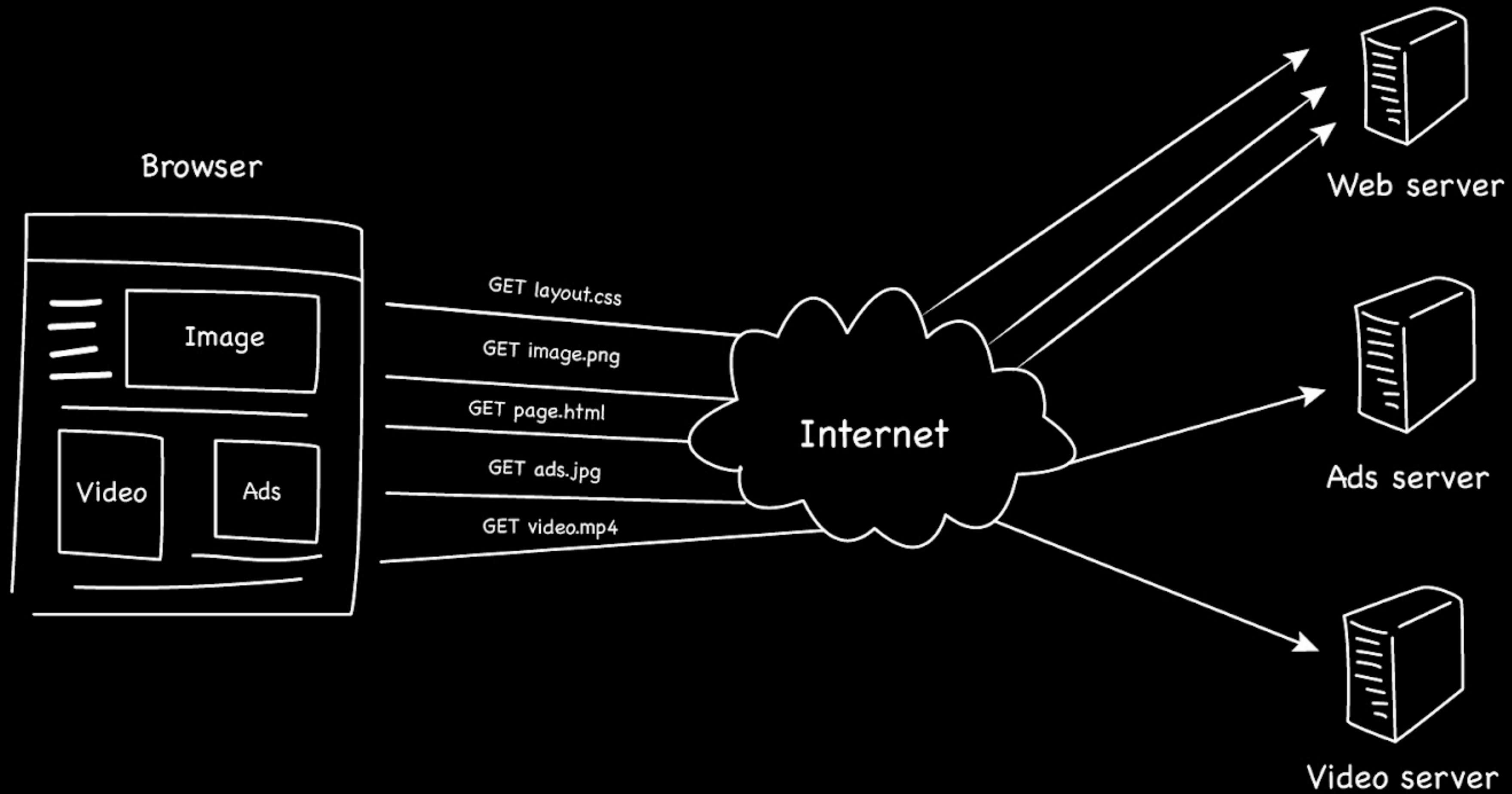
   2. Git & GitHub

   3. NPM & Node

   4. Bundlers

2. Boilerplate

3. What's next?

Schedule

Web architecture

Web architecture

Remember: the Tech Track is
for Front-end, not full stack!

Browser

Image

Video   Ads

GET layout.css

GET image.png

GET page.html

GET ads.jpg

GET video.mp4

Internet

CANCELLED

Schedule

1. Web architecture

   1. *Bash & Terminal*

   2. *Git & GitHub*

   3. *NPM & Node*

   4. *Bundlers*

2. Boilerplate

3. What's next?
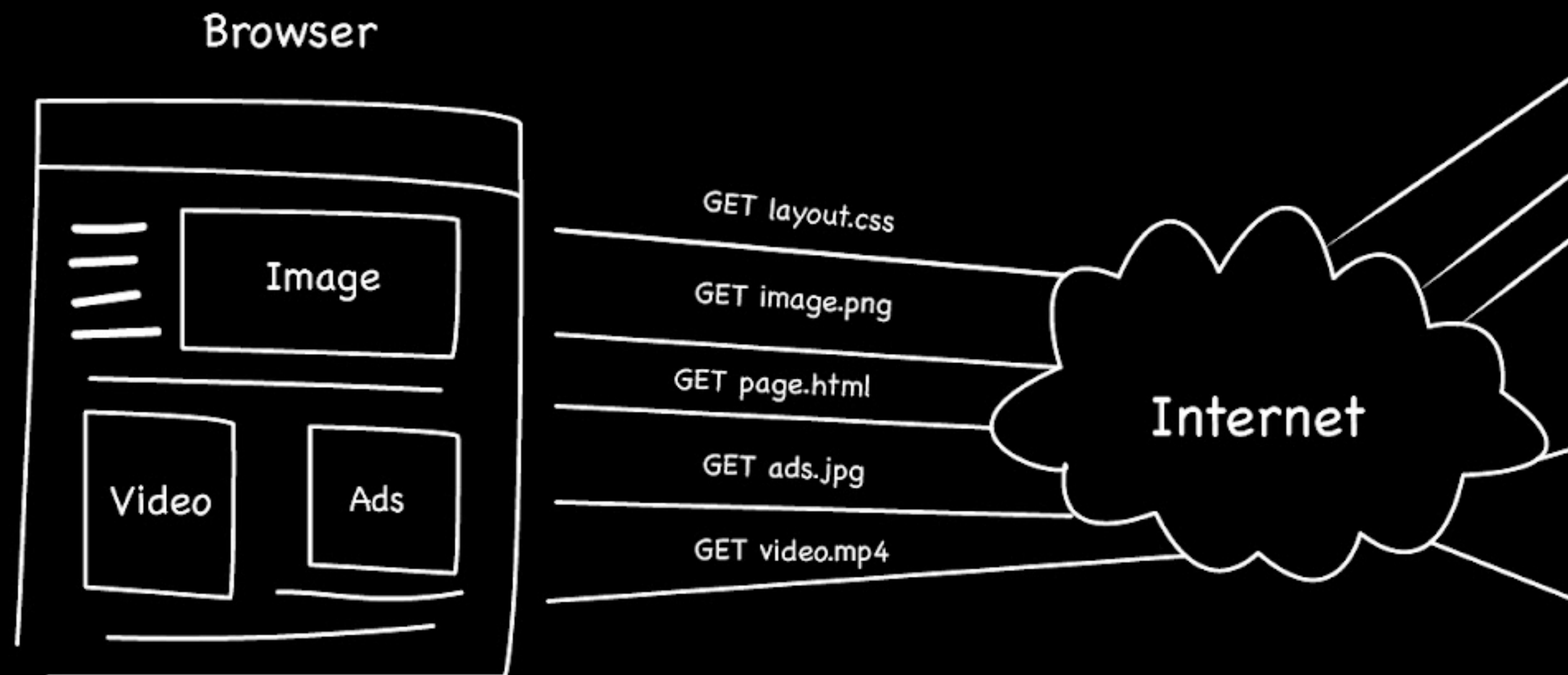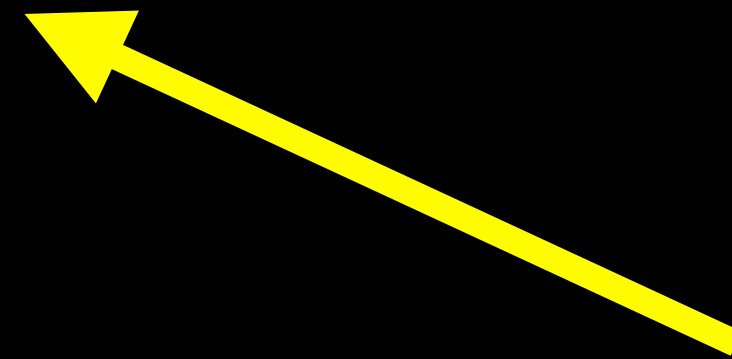
So why do we (re)visit these?

Web architecture

Alexander Joseph / JUN 9, 2022 / 3 comments

The Future Of Frontend Build Tools

📅 10 min read 🏷 JavaScript, Performance, DevTools

🐦 Share on Twitter, LinkedIn

QUICK SUMMARY ↝ *This article explores the concept of tooling for frontend development on the web. You will learn why we need frontend tooling, the various stages of evolution it has gone through, and the new developments that will shape the frontend build tools of the future. To follow along with this article, a general understanding of modern frontend development on the web is necessary.*

Web architecture

- Browser compatibility across users

- Using node_modules (more on that later…)

- Allows you to quickly shift between programming paradigms, and learn new frameworks that use them

Web architecture

Let's not reinvent the wheel
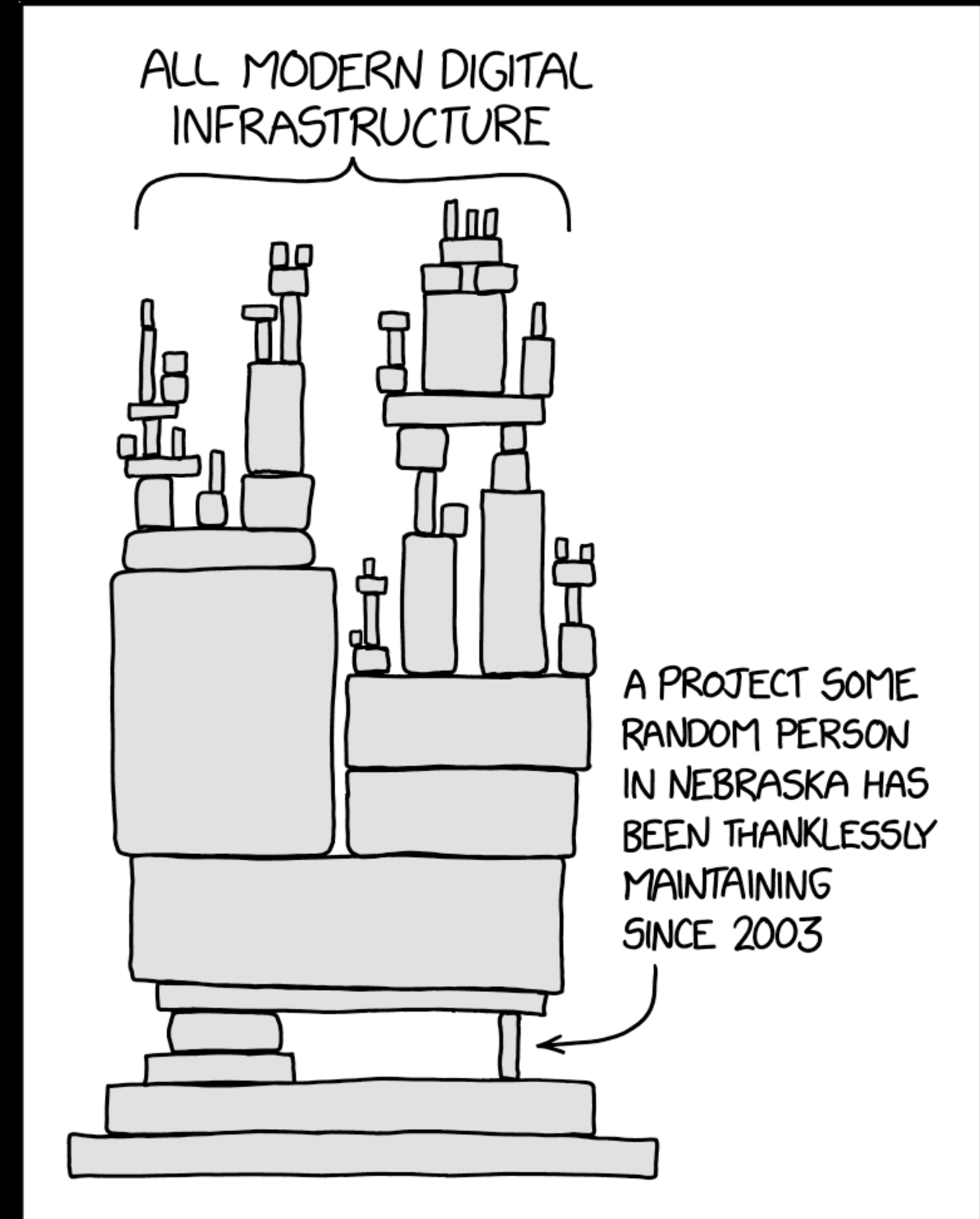
Schedule

1. Web architecture

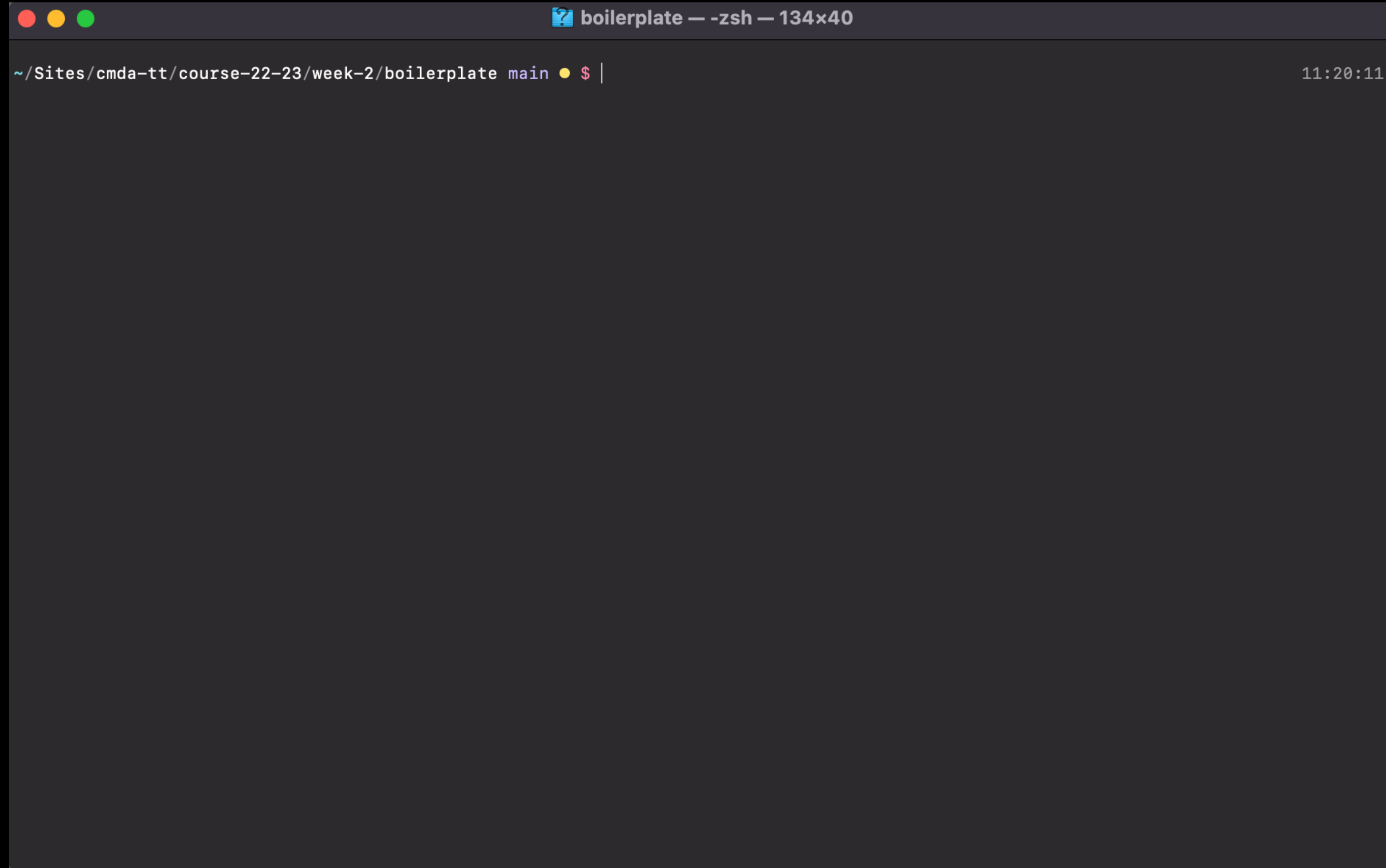   1. **Bash & Terminal**

   2. Git & GitHub

   3. NPM & Node

   4. Bundlers

2. Boilerplate

3. What's next?

Bash & Terminal

Bash & Terminal

Bash & Terminal


```
TERMINAL                              bash - hello-world   +   ⌄   ☐   🗑   ⌃   ✕

~ took 508ms
⬤ ❯ mkdir hello-world && cd hello-world

~/hello-world
⬤ ❯ git init
Initialized empty Git repository in /Users/daimms/hello-world/.git/

~/hello-world ⎇ main
⬤ ❯ echo "test" > test_file

~/hello-world ⎇ main [?]
⬤ ❯ git add . && git commit -m "Hello world!"
[main (root-commit) c1c1283] Hello world!
 1 file changed, 1 insertion(+)
 create mode 100644 test_file

~/hello-world ⎇ main
○ ❯
```
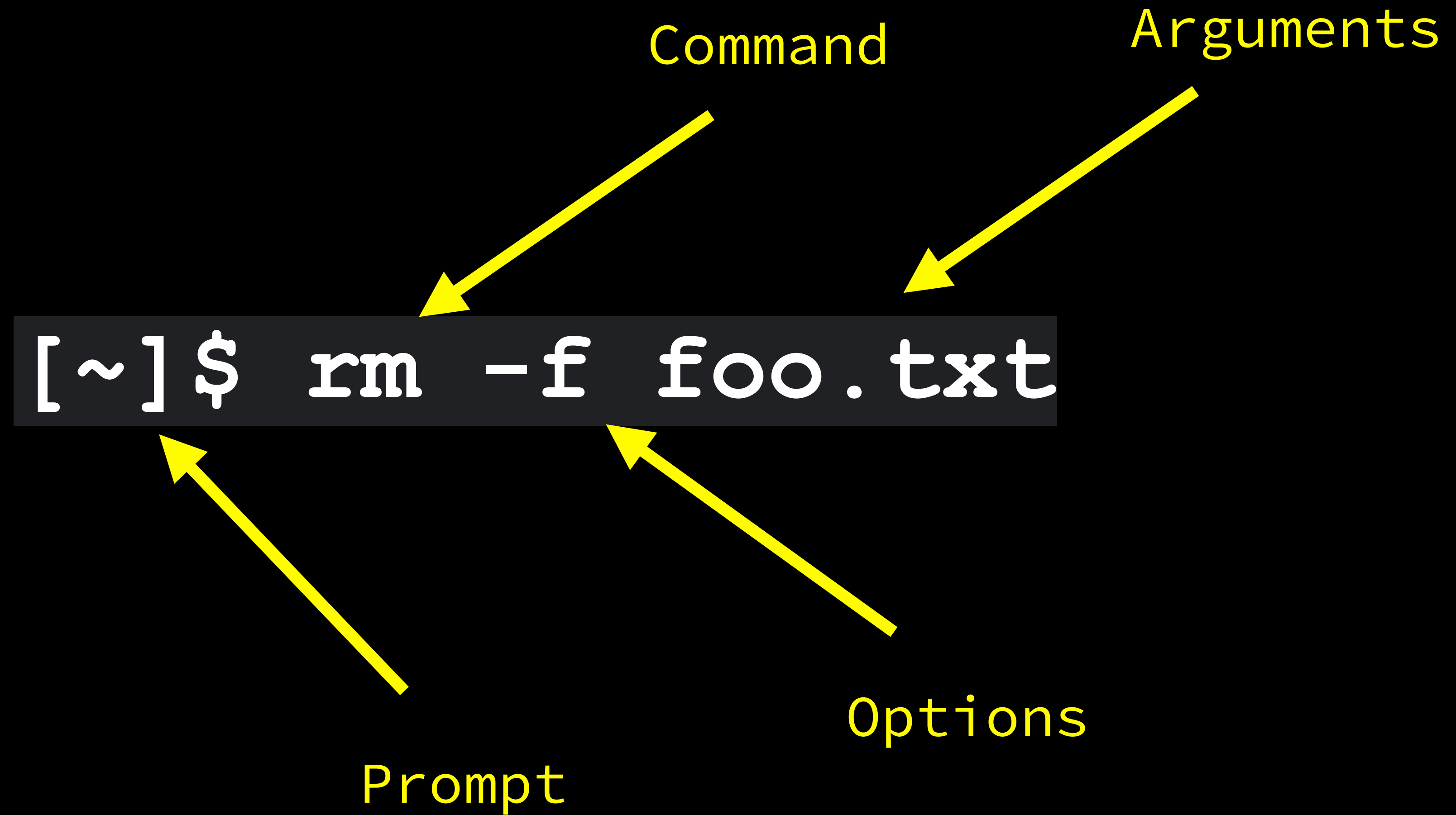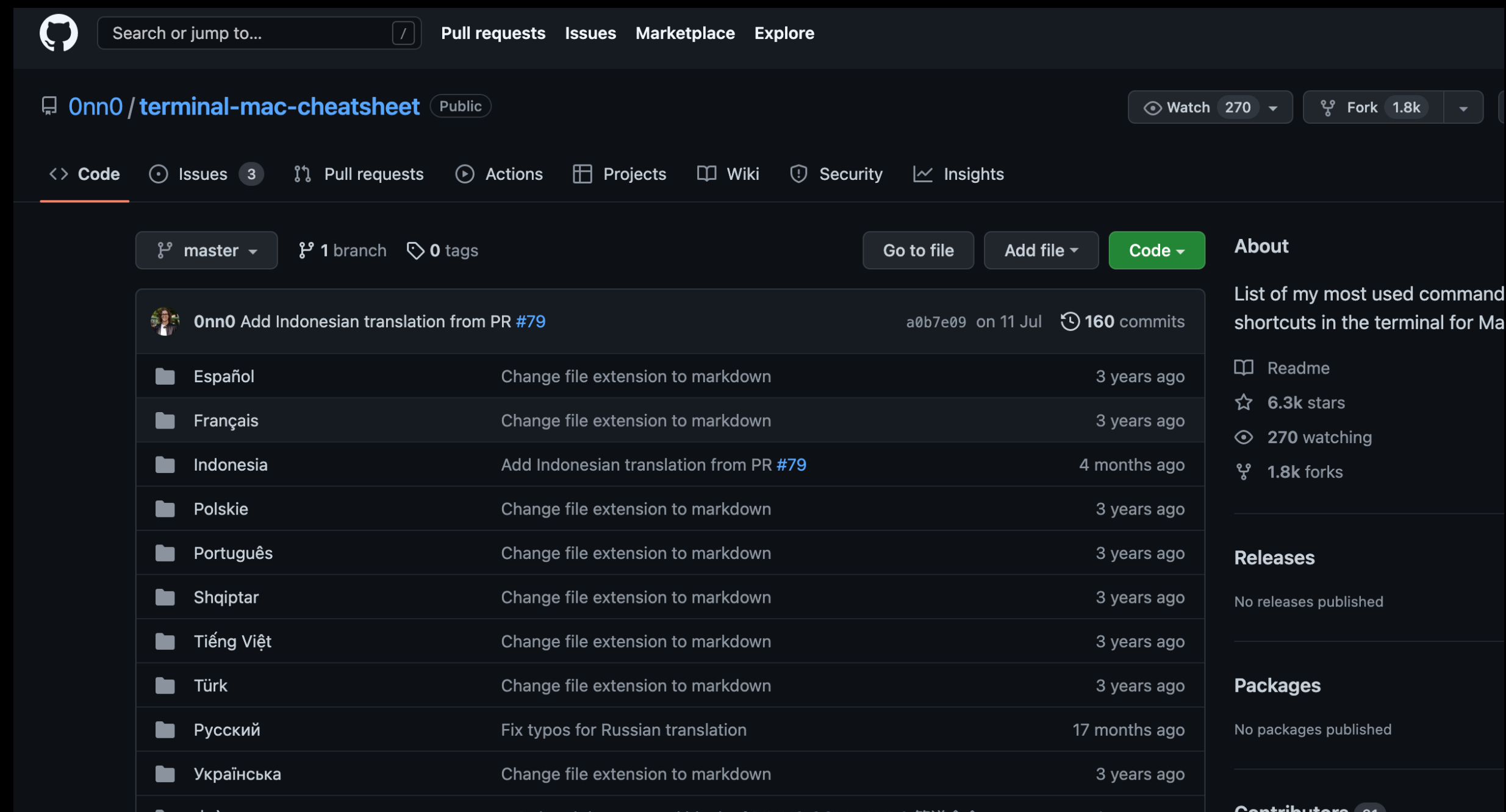
Bash & Terminal

```
[~]$ rm -f foo.txt
```

Bash & Terminal

Command

Arguments

`[~]$ rm -f foo.txt`

Prompt

Options

Bash & Terminal

Note: syntax between Mac/Unix and Windows may differ!

Schedule

Git

Git is free and open source software for distributed version control: tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development.

Git + GitHub?

Git + GitHub?

GitHub, Inc. is an Internet hosting service for software development and version control using Git. It provides the distributed version control of Git plus access control, bug tracking, software feature requests, task management, continuous integration, and wikis for every project.

Git + GitHub?
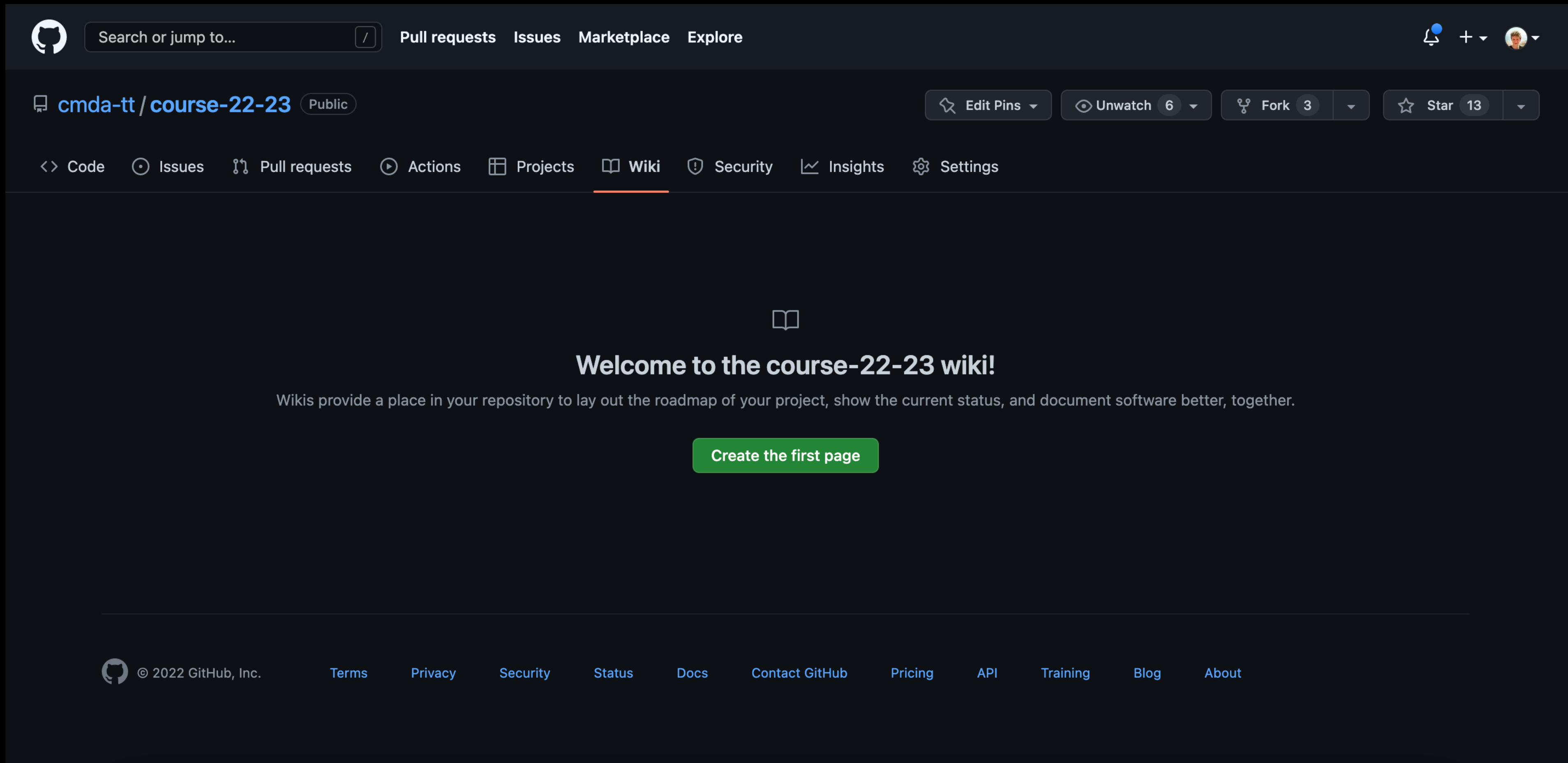
Git + GitHub

- We use Git as the **software that manages changes**

- We use GitHub to host and deploy / show our code

- Our code lives in a Git Repository, hosted online on GitHub

- Our wiki (documentation) is a separate Git Repository that is coupled with our code-repository

Assignment

- Install Git on your computer, if you haven't done it

- Install GitHub Desktop (or use the terminal if you're going hardcore) and link it to your GitHub account

- Blok Tech students: Help your fellow Tech-Trackers and make sure your existing config is still working!

- Clone your tech-track repository to your computer and commit something to your code and documentation repository!

Tutorials can be found on #tech-track at Teams!

Schedule

NodeJS

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on a JavaScript Engine (i.e. V8 engine) and executes JavaScript code outside a web browser, which was designed to build scalable network applications.

NodeJS

- We use NodeJS in this course for two reasons only:

  - To allow us to easily use NPM (more on that later…)

  - To enable us to use bundlers

NPM

npm (originally short for **Node Package Manager**)[4] is a package manager for the JavaScript programming language maintained by npm, Inc. npm is the default package manager for the JavaScript runtime environment Node.js.

NPM

NPM is a database of JavaScript modules, some for back-end, others for front-end. D3 (the library we'll be using next week) is hosted here as well for example.

Assignment

- Install Node using the Node Version Manager (most reliable way) on Mac, use the installer on the NodeJS website for Windows.

- Verify the installation, this also goes for blok-tech alumni, use the commands: `node -v` and `npm -v`

Tutorials can be found on #tech-track at Teams!

Schedule
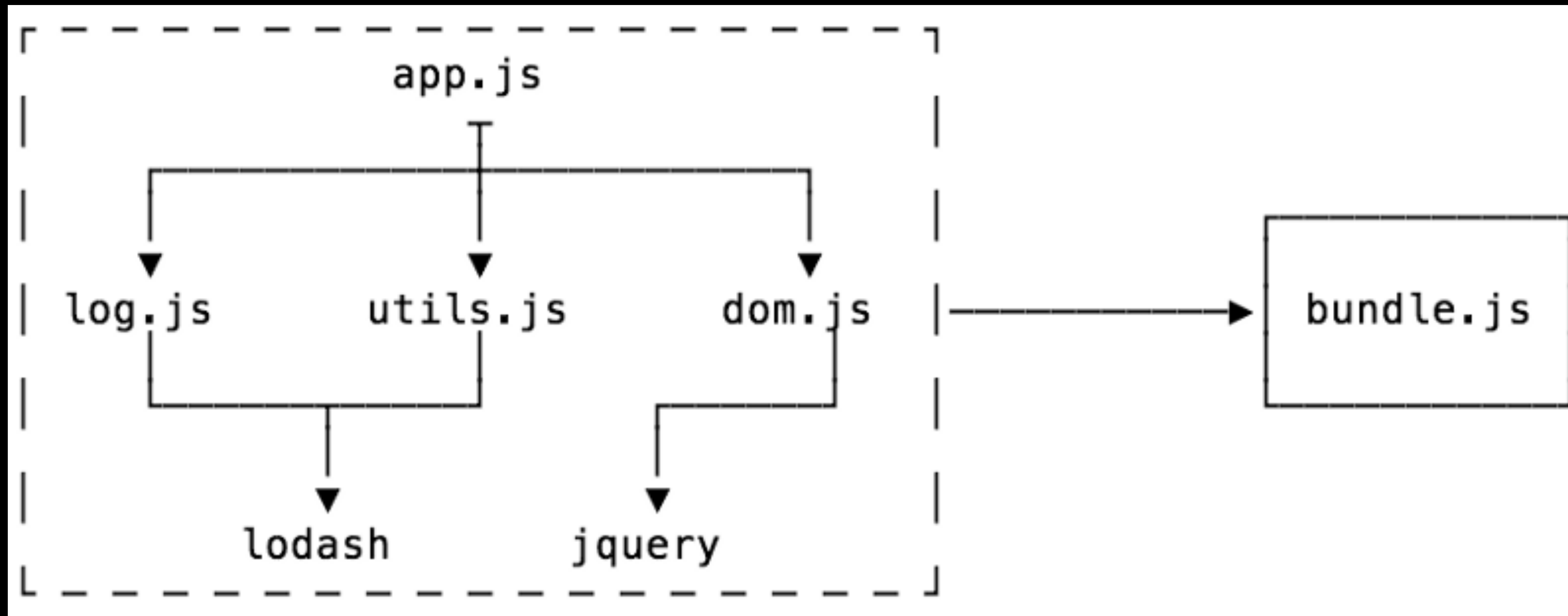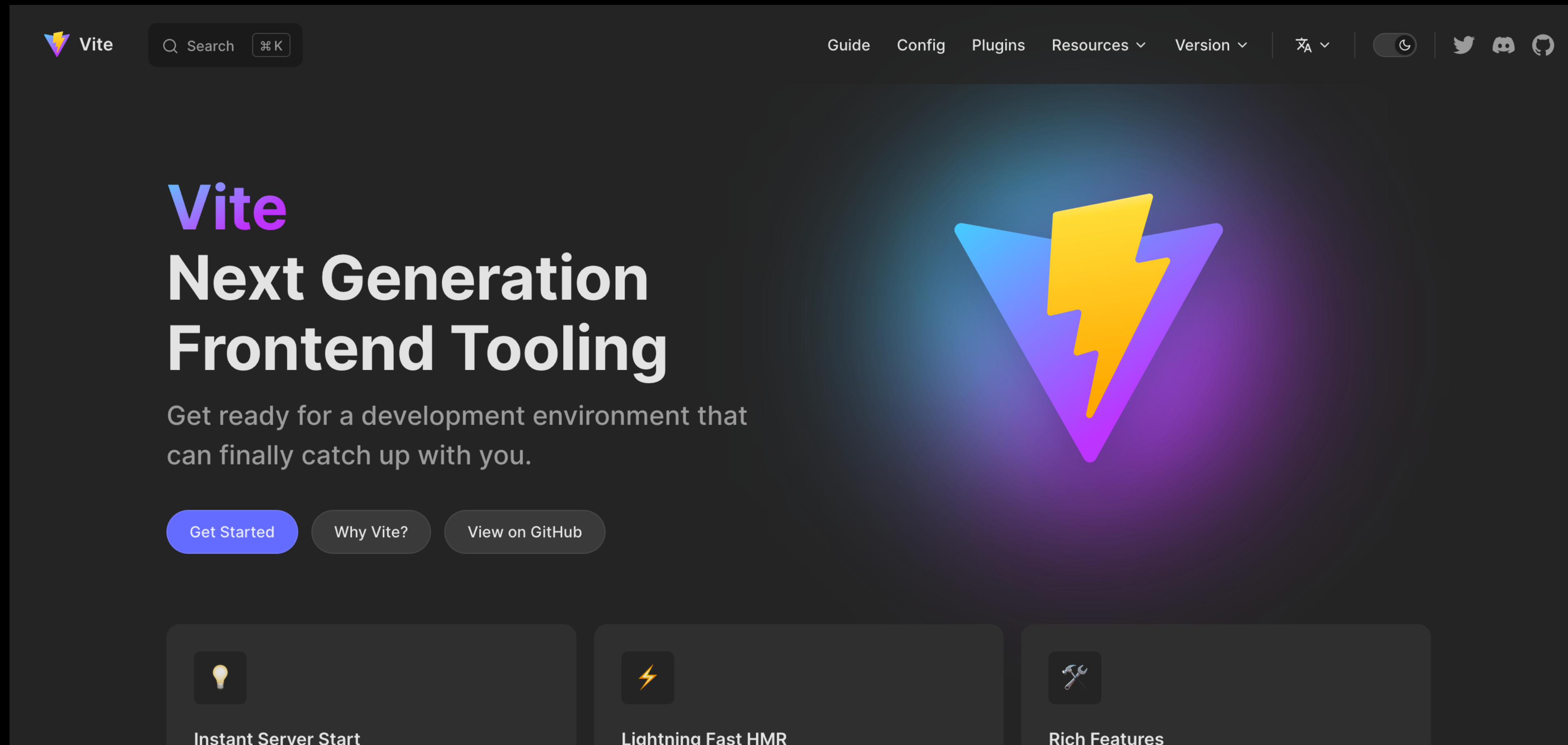
Bundlers

Bundlers

We used to have a lot of bundlers, Gulp, Grunt, Webpack
bla bla bla, etc…

But now:

Bundlers

Bundlers

Vite uses the command line and the NodeJS ecosystem to automatically perform some magical tasks for us, such as:

• Hosting a live server for us to see code

• Automatically bundle node_modules and our own modules into a single file

• Minify and optimise code

• Allow us to use Sass (CSS on steroids) or TypeScript (JavaScript on steroids).

• Easily integrate frameworks like React/Vue/Svelte

Schedule

Schedule

Maybe a small break?

Schedule

1. Web architecture

   1. Bash & Terminal

   2. Git & GitHub

   3. NPM & Node

   4. Bundlers

2. **Boilerplate**

3. What's next?
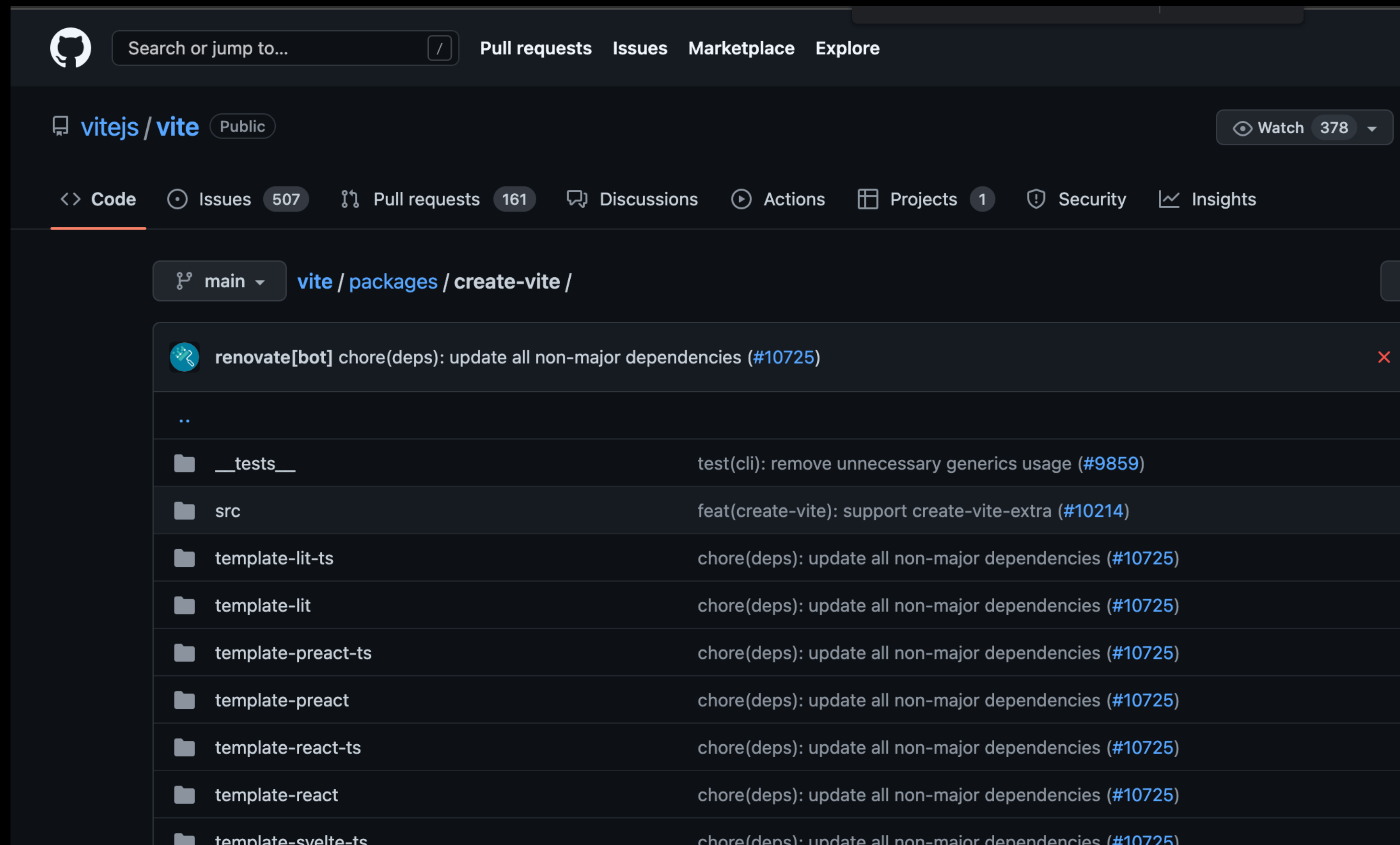
Boilerplate

▼ boilerplate
  ▼ dist
    ▼ assets
      {} index.8d4121f2.css
      (); index.b4b29d20.js
    ◻ favicon.ico
    <> index.html
  ▼ images
    ◻ tt.png
  ▶ node_modules
  ▼ public
    ◻ favicon.ico
  ▼ scripts
    (); main.js
  ▼ styles
    {} style.css
  ▤ .gitignore
  <> index.html
  ◎ package-lock.json
  ◎ package.json

Comes with a free live demo!

(After which you'll be charged 9,99 per month for extended support)

Boilerplate

Boilerplate

Schedule

What's next?

- Check the SCHEDULE.md file for the planning

- Make sure to do your homework for tomorrow!

- We'll meet Wednesday again for a guest lecture
  Please note: (sketch) notes are
  mandatory deliverables in your documentation!

Uncaught SyntaxError
Unexpected end of input