WebDev 5

CS571: Building User Interfaces

Cole Nelson

Today's Warmup

- Clone today's code to your machine.
  - Run the command `npm install` inside of the `starter` and `solution` folders.

Announcements

- Thank you for your feedback! :)
  - So far, so good!
  - ICA submission window will increase to 48 hours.
  - Will spend more time on the fundametals, esp. in CS272 offered this Fall!
- Midterm exam on Thursday! No class in the morning.

Midterm Exam

You should have received an email on Sunday with the exam details. Remember to bring...

- A #2 pencil
- Your double-sided notesheet
- Your Wiscard

Something come up? Let me know **ASAP!**

Learning Objectives

- Be able to memoize components to optimize web application performance.
- Be able to write reusable logic in the form of custom React hooks.
- Understand the bigger picture of the React eco-system, including the NPM community.
- Be able to deploy a web application.

Memoization

Not memorization!

Memoization

Storing the result so you can use it next time instead of calculating the same thing again and again

what the frik is: memoization

`useCallback` to memoize functions

`useMemo` to memoize calculated values

`memo` to memoize components

Note: This will be going away in React 19!

`useCallback` Hook

Consider the following functional component...

```
function MyApp() {
  const myComplicatedFunction = () => {
    // ...
  }
  return <>
    <button onClick={myComplicatedFunction}>Click Me</button>
  </>
}
```

How many times do we *create* the function `myComplicatedFunction`? We do on *every render*!

`useCallback` Hook

`useCallback` is used to 'memoize' a callback function.

```
function MyApp() {
  const myComplicatedFunction = useCallback(() => {
    // ...
  }, []);
  return <>
    <button onClick={myComplicatedFunction}>Click Me</button>
  </>
}
```

Takes a callback function to 'memoize' and an optional list of dependencies (e.g. when to re-'memoize').

`useMemo` Hook

Same thing as `useCallback`, except memoizes the *value* of a *callback* rather than the *callback* itself.

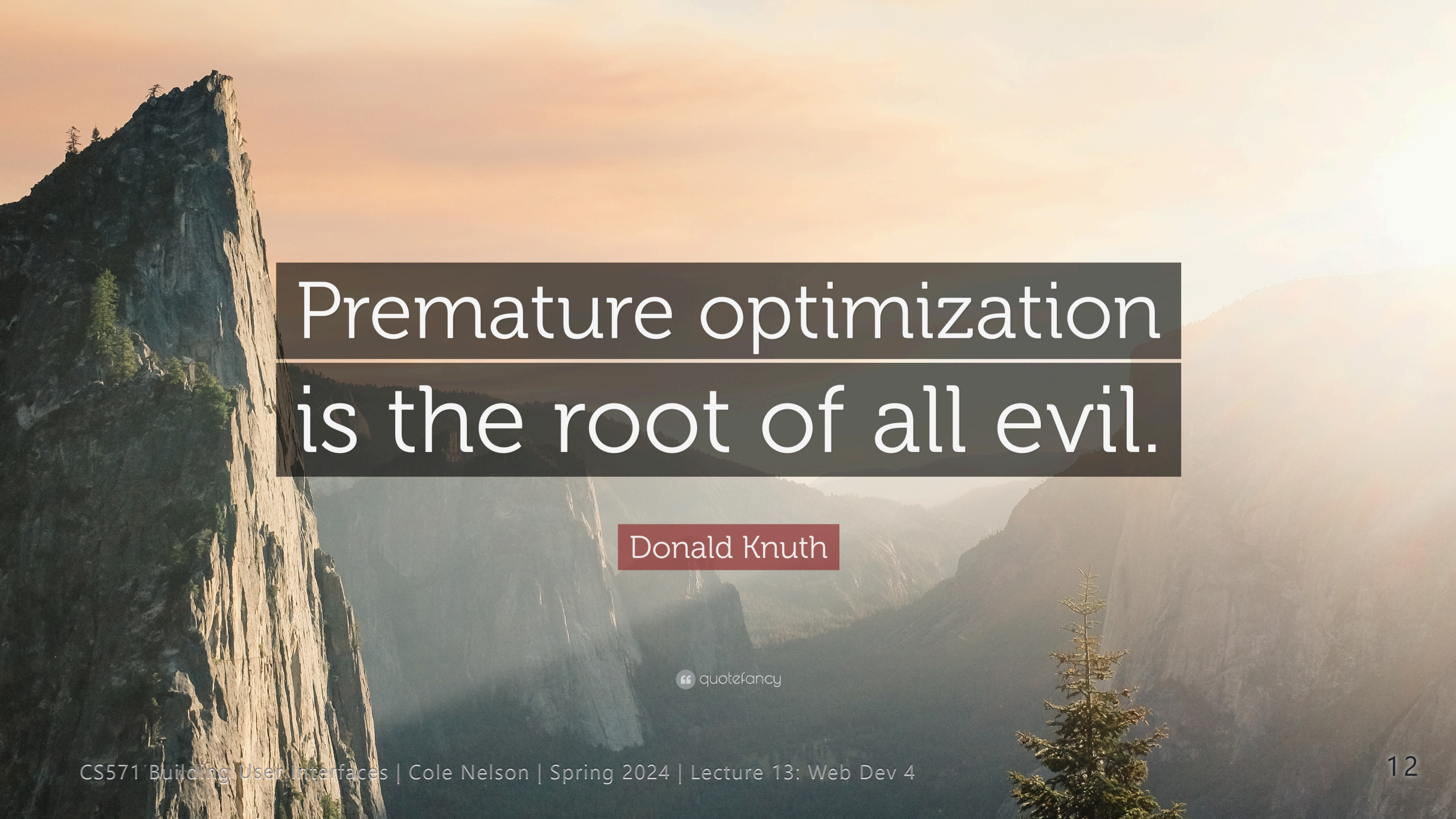
```
function MyApp() {
  const myComplicatedValue = useMemo(() => { /* Some complex call */}, []);

  return <>
    <p>{myComplicatedValue}</p>
  </>
}
```

`memo` -ized Components

Used for creating *purely functional* components. Given the same props, the function renders the same output.

```
//                      v--- Name of functional component!
export default memo(GroceryList, (prevProps, nextProps) => {
  return prevProps.apples === nextProps.apples &&
    prevProps.bananas === nextProps.bananas &&
    prevProps.coconuts === nextProps.coconuts;
})
```

See StackBlitz for `useCallback`, `useMemo`, and `memo`


Premature optimization is the root of all evil.
Donald Knuth
quotefancy

A Plea for Lean Software

Niklaus Wirth
ETH Zürich

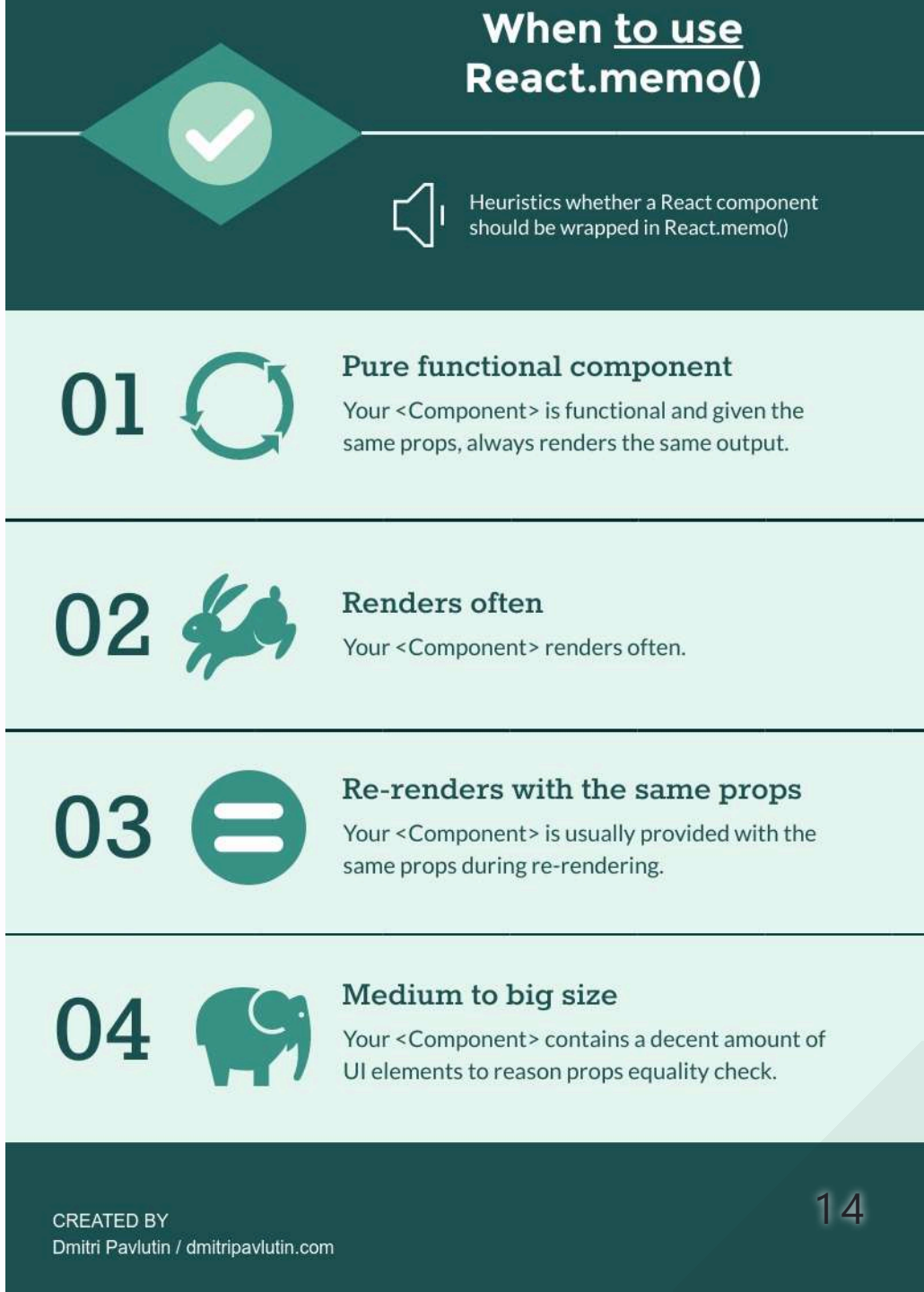
Memory requirements of today's workstations typically jump substantially—from several to many megabytes—whenever there's a new software release. When demand surpasses capacity, it's time to buy add-on memory. When the system has no more extensibility, it's time to buy a new, more powerful workstation. Do increased performance and functionality keep pace with the increased demand for resources? Mostly the answer is no.

About 25 years ago, an interactive text editor could be designed with as little as 8,000 bytes of storage. (Modern program editors request 100 times that much!) An operating system had to manage with 8,000 bytes, and a compiler had to fit into 32 Kbytes, whereas their modern descendants require megabytes. Has all this inflated software become any faster? On the contrary. Were it not for a thousand times faster hardware, modern software would be utterly unusable.

Finding a Balance

1. Given the same input, renders the same output.
2. Is rendered often.
3. Does not change often.
4. Is of substantial size.

Dmitri Pavlutin Blog Post



Your turn!

Expand on BadgerChat Mini from last week by optimizing the application with *memoization*.

Clone from here.

How Can We Reuse Logic?

Practicing **D**on't **R**epeat **Y**ourself (DRY)

Custom React Hooks

You can write your own custom hooks! These are just JavaScript functions that can use React's features!

- We use **custom components** to re-use **UI elements**.
- We use **custom hooks** to re-use **business logic**.

JSConf Talk

Let's Write a Custom Hook!

Writing reusable logic for persisting data.

StackBlitz Solution | Inspitation from WDS

Your turn!

Use this custom hook in BadgerChat Mini!

Congrats!

You are now a React Devloper! 🥳🎏🎉

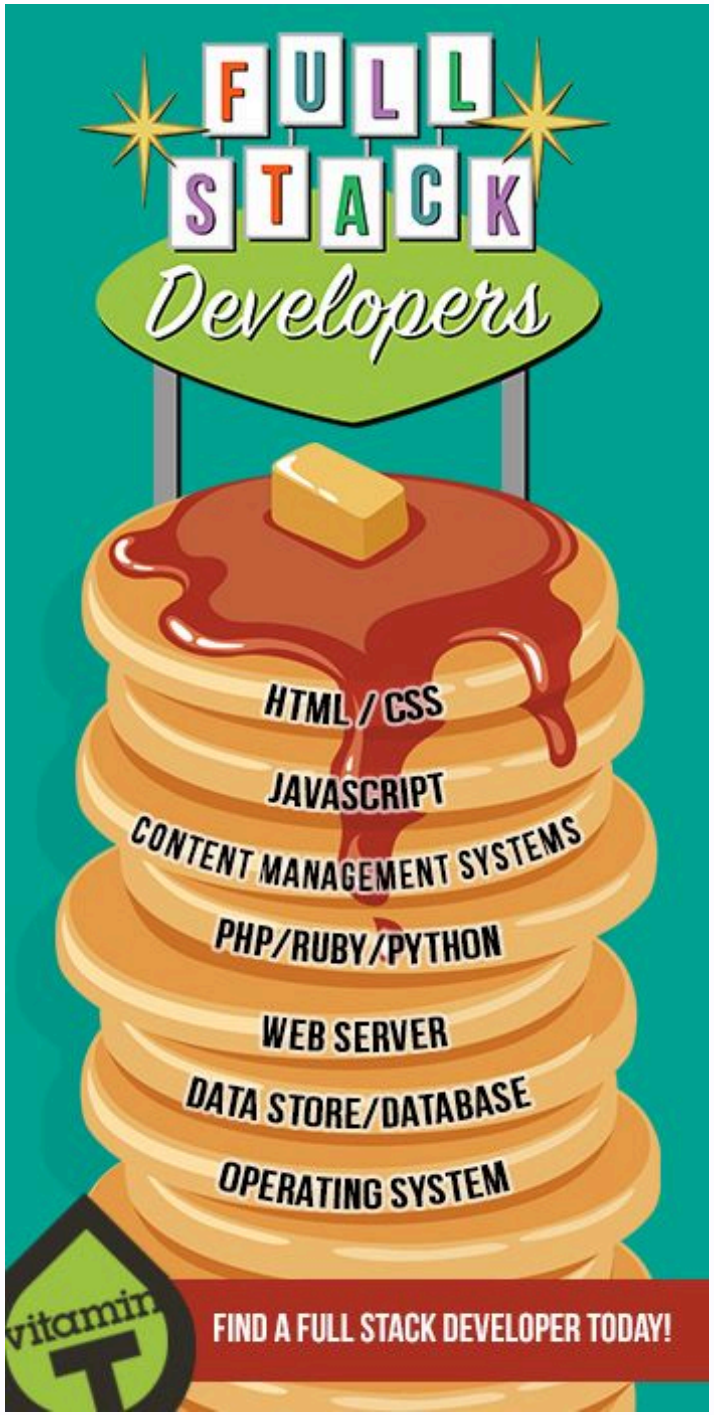
Questions You Should Ask Yourself...

- What even is "frontend development"?
- Where does this fit in to the software stack?
- How can I get my web app out in front of customers?
- What concerns should I have about my web app?
- What else can I do with these skills?
- How much of a raise should I ask for? 💰

Software Stack

Think of software like a stack of pancakes...

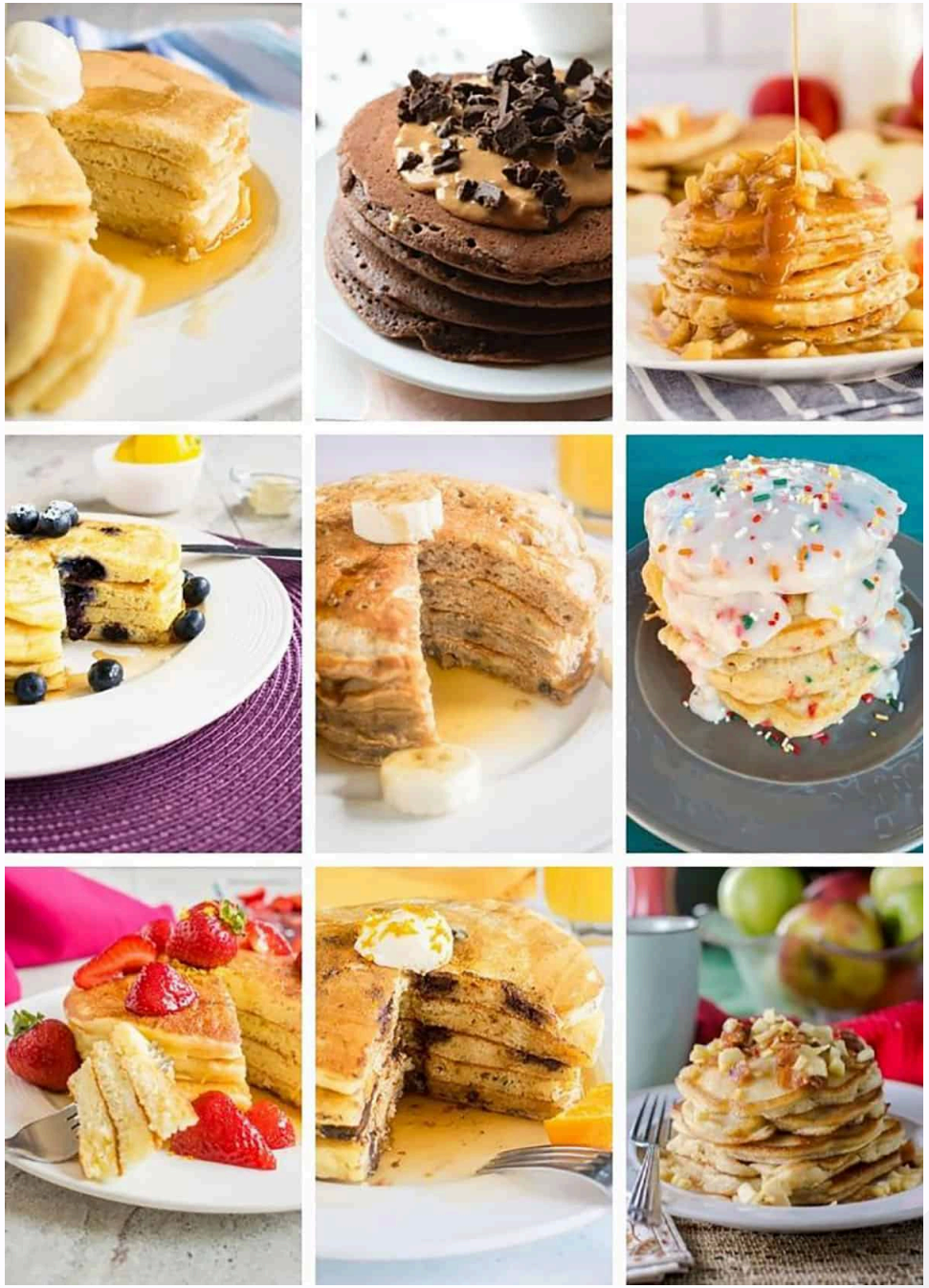
Image Source



Software Stack

... where each pancake can be its own flavor...

Image Source



Software Stack

... and can be cooked its own way...

Image Source

Software Stack

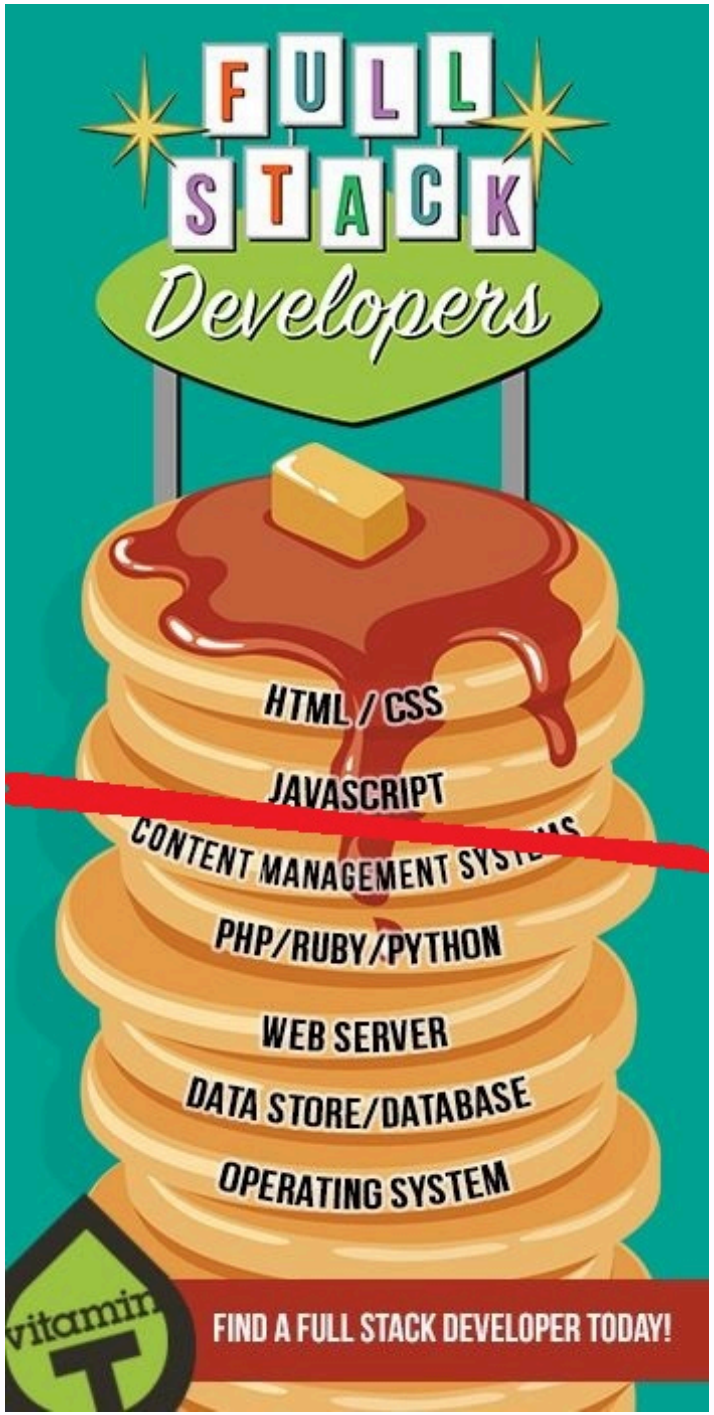
... with as many or as few as we want!

Image Source

Software Stack

We typically refer to the "frontend" as the content that gets delivered to the user...

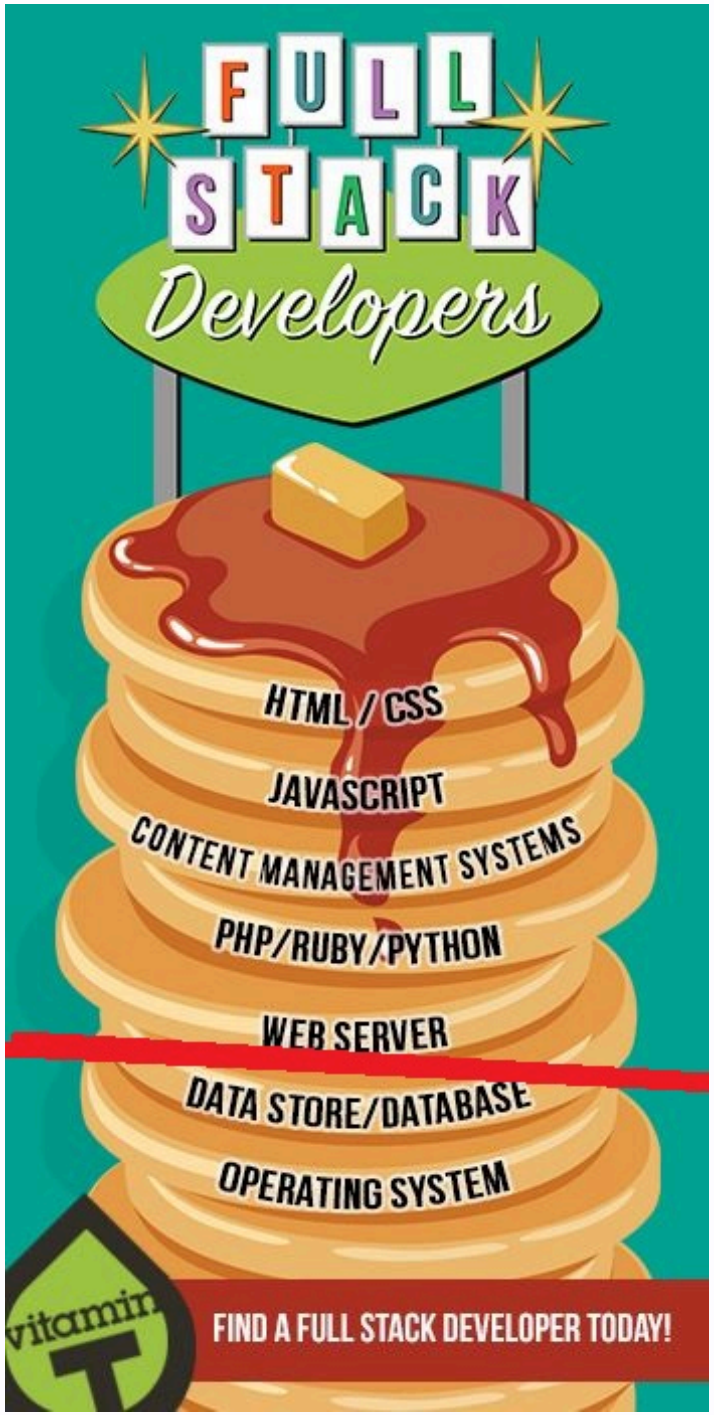
Image Source



Software Stack

...but this can change based on your perspective!

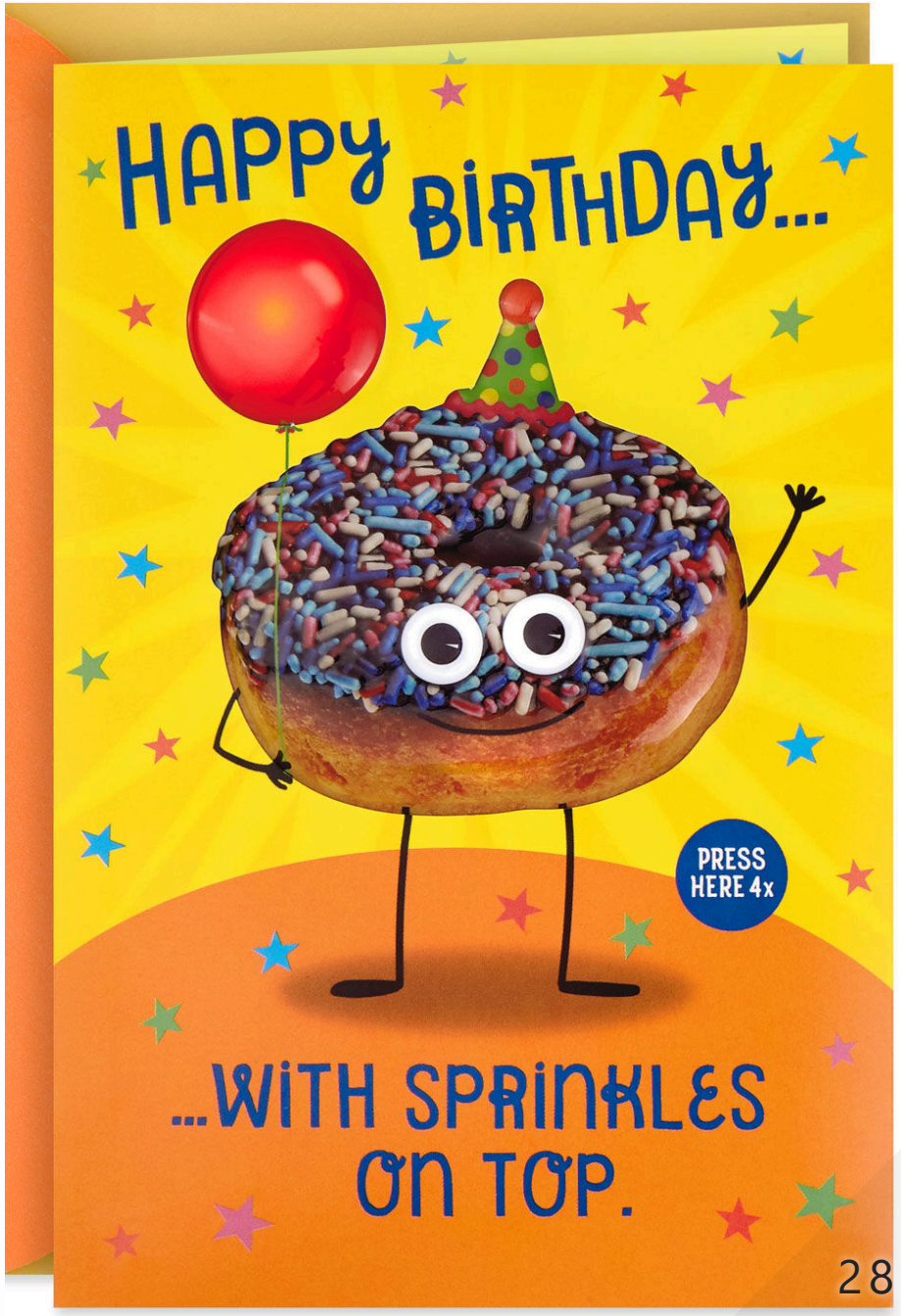
Image Source



The Browser

However, we are constrained to what the browser can interpret...

- HTML
- CSS
- JS



So... is React an Exception?

Facebook is influential, but not *that* influential!

Reminder: JSX

This React component displays Hello World on the webpage using JSX.

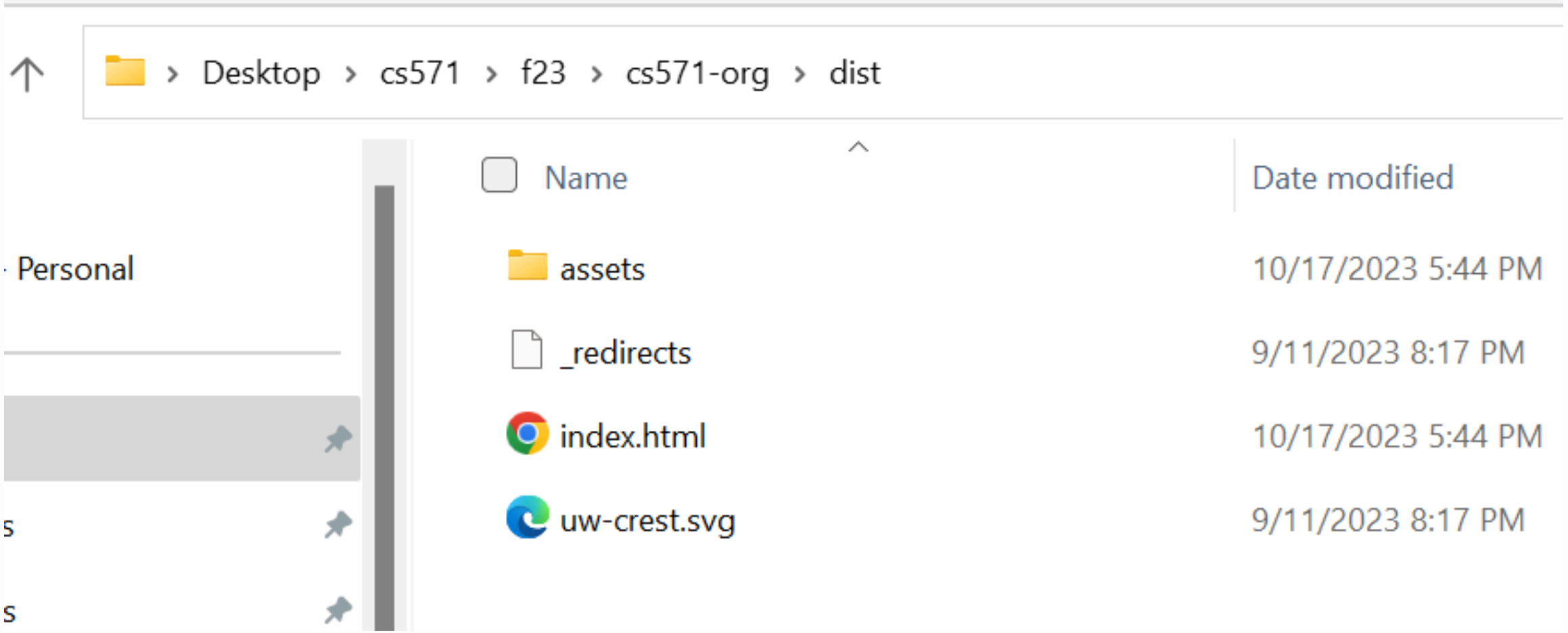
```
function Welcome() {
  return <h1>Hello World!</h1>;
}
```

This is transpiled into JS, CSS, and HTML.

Delivery of React App

We don't deliver our JSX code, we deliver HTML, CSS, and JS generated via `npm run build`!

This creates our "build bundle"...


Desktop
cs571
f23
cs571-org
dist
Personal
Name
Date modified
assets
10/17/2023 5:44 PM
_redirects
9/11/2023 8:17 PM
index.html
10/17/2023 5:44 PM
uw-crest.svg
9/11/2023 8:17 PM

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <link
      rel="stylesheet"
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
      integrity="sha384-9ndCyUaIbzAi2FUVXJi0CjmCapSmO7SnpJef0486qhLnuZ2cdeRhO02iuK6FUUVM"
      crossorigin="anonymous"
    />
    <link rel="icon" type="image/svg+xml" href="/uw-crest.svg" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>CS571</title>
    <script type="module" crossorigin src="/assets/index-10eed002.js"></script>
    <link rel="stylesheet" href="/assets/index-fe717832.css">
  </head>
  <body>
    <div id="root"></div>
  </body>
</html>
```

```
/*! For license information please see main.aae268c3.js.LICENSE.txt */
!function(){var e={694:function(e,t){var n;!function(){"use strict";var r={}.
hasOwnProperty;function l(){for(var e=[],t=0;t<arguments.length;t++){var n=arguments[t];
if(n){var a=typeof n;if("string"===a||"number"===a)e.push(n);else if(Array.isArray(n)){
if(n.length){var o=l.apply(null,n);o&&e.push(o)}}else if("object"===a)if(n.toString===
Object.prototype.toString)for(var u in n)r.call(n,u)&&n[u]&&e.push(u);else e.push(n.
toString())}}return e.join(" ")}e.exports?(l.default=l,e.exports=l):void 0===(n=function
(){return l}.apply(t,[]))||(e.exports=n)}()},618:function(e,t,n){var r;!function(){"use
strict";var l=!("undefined"===typeof window||!window.document||!window.document.
createElement),a={canUseDOM:l,canUseWorkers:"undefined"!==typeof Worker,
canUseEventListeners:l&&!(!window.addEventListener&&!window.attachEvent),canUseViewport:
l&&!!window.screen};void 0===(r=function(){return a}.call(t,n,t,e))||(e.exports=r)}()},
888:function(e,t,n){"use strict";var r=n(47);function l(){}function a(){}a.
resetWarningCache=l,e.exports=function(){function e(e,t,n,l,a,o){if(o!==r){var u=new
Error("Calling PropTypes validators directly is not supported by the `prop-types`
package. Use PropTypes.checkPropTypes() to call them. Read more at
http://fb.me/use-check-prop-types");throw u.name="Invariant Violation",u}}function t(){
return e}e.isRequired=e;var n={array:e,bigint:e,bool:e,func:e,number:e,object:e,string:e
,symbol:e,any:e,arrayOf:t,element:e,elementType:e,instanceOf:t,node:e,objectOf:t,oneOf:t
,oneOfType:t,shape:t,exact:t,checkPropTypes:a,resetWarningCache:l};return n.PropTypes=n,
n}},7:function(e,t,n){e.exports=n(888)()},47:function(e){"use strict";e.exports=
"SECRET_DO_NOT_PASS_THIS_OR_YOU_WILL_BE_FIRED"},463:function(e,t,n){"use strict";var r=n
(791),l=n(296);function a(e){for(var t=
```

Build Bundle Deployment

Let's do a deployment!

... to cs571.org

Developing Your Own App!

A lot of the starter code was given to you, but you can generate this yourself!

```
npm create vite@latest my-website -- --template react
cd my-website
npm install
npm run dev
```

This will get you started!

Vite Documentation

Developing Your Own App!

For Bootstrap, run...

```
npm install react-bootstrap bootstrap
```

But you'll also need to import and include a bootstrap stylesheet from the web...

React Bootstrap Documentation

Developing Your Own App!

In `main.jsx`, put...

```
import 'bootstrap/dist/css/bootstrap.min.css';
```

In `index.html`, put...

```
<link
  rel="stylesheet"
  href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css"
  integrity="sha384-T3c6CoIi6uLrA9TneNEoa7RxnatzjcDSCmG1MXxSR1GAsXEV/Dwwykc2MPK8M2HN"
  crossorigin="anonymous" />
```

React Bootstrap Documentation

Developing Your Own App!

For React Router, run...

```
npm install react-router-dom localforage match-sorter sort-by
```

You'll also need to list a base in `vite.config.js`

```
base: "/myprojectpath"
```

Note: For GitHub Pages you'll need to use a `HashRouter` rather than a `BrowserRouter`. Why?

React Router Documentation

See Workshop Video!

See Hongtao's video posted to Kaltura Gallery!

Concerns in Production...

- Reliability
- Performance
- Monitoring
- Business Value of Delivery
- Search Engine Optimization (SEO)

Reliability

Does our code work?

- Manual testing
- Automated testing
  - Jest
  - React Testing Library
- Static analysis
  - TypeScript: Used for type-checking
  - ESLint: Used for following best practice

Performance

Does our code work *well*?

- Be aware of the "bundle size"!
- Our code specifically...
  - Perf
  - Profiler
- Our code broadly...
  - Google Lighthouse
  - Chrome User Experience Report (CrUX)

Monitoring

Does our code *continue to* work well?

- Logging
  - *Not* `console.log`
  - Sentry
  - DataDog
- Cloud Tools
  - Cloud Monitoring Tools
  - DownDetector

Business Value of Delivery

- Core Questions
  - Are we making money? 
  - Are users making use of new features?
- Analysis Methods
  - A/B Testing
  - Customer Surveys
- Commercial Tools
  - Pendo

Search Engine Optimization (SEO)

The generated HTML doesn't have much in it!

What is a search engine crawler supposed to do?

Option: Server Side Rendering next.js

Your Considerations?

Using NPM & Node.js

The `package.json` specifies details about the project.
The `package-lock.json` specifies specific details about dependencies.

`npm install` installs dependencies in `node_modules`.
`npm run dev` runs your webserver w/ hot-reloading!
`npm run build` generates your build bundle.

Using NPM & Node.js

What's the difference?

- **Node** is the **runtime engine**
- **NPM** is the **package manager**

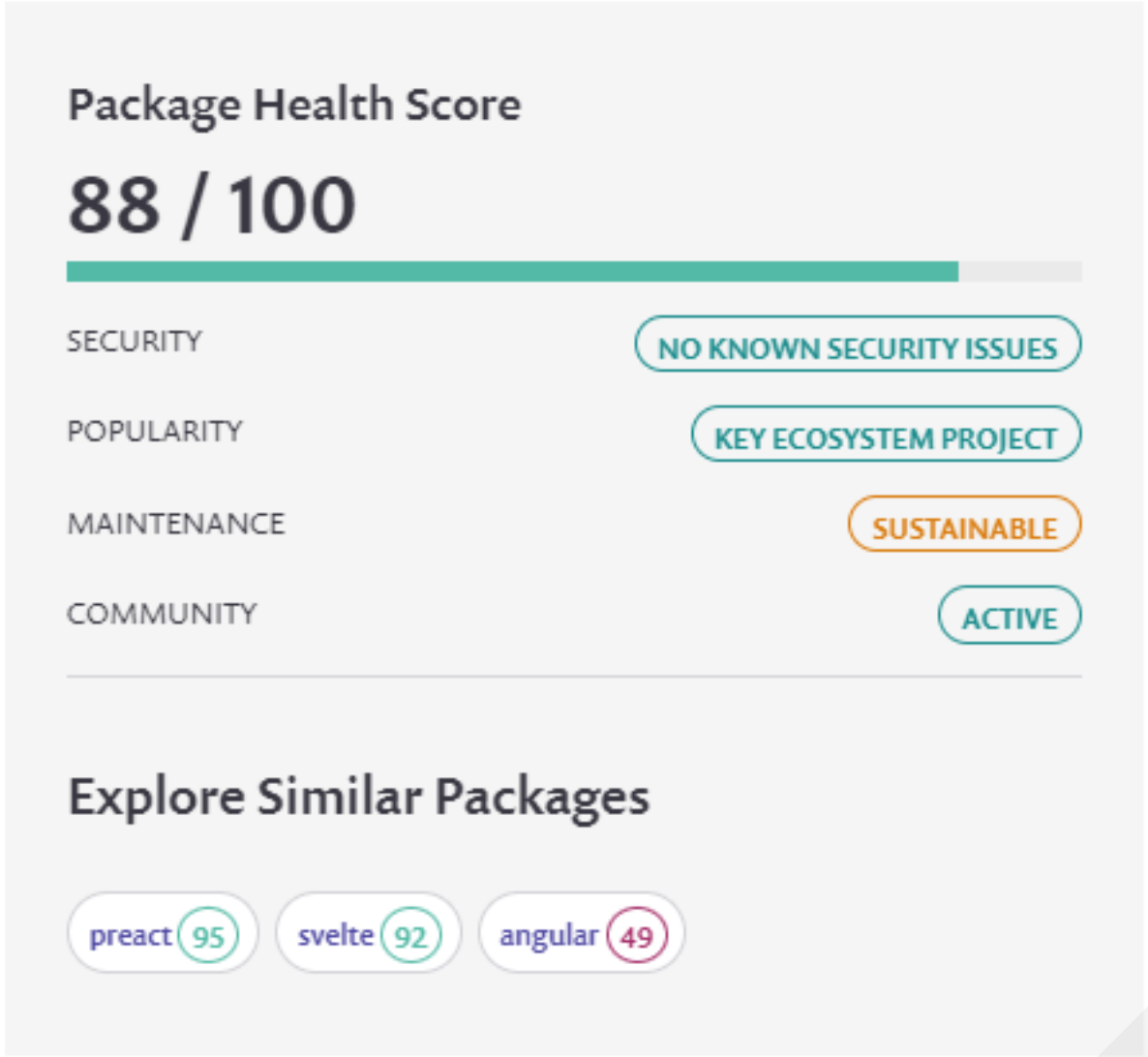
Other package managers such as pnpm and yarn exist.

NPM Community

Choose your packages carefully!

- Are they secure?
- Are they well-documented?
- Will they be maintained?

A helpful tool!



```
PS C:\Users\ctnelson1997\Desktop\cs571\f

up to date, audited 299 packages in 3s

99 packages are looking for funding
  run `npm fund` for details

2 vulnerabilities (1 moderate, 1 high)

To address all issues (including breakin
  npm audit fix --force

Run `npm audit` for details.
```

NPM Supply Chain Risk

Run `npm ls -all`.

Vulnerabilities are discovered over time.

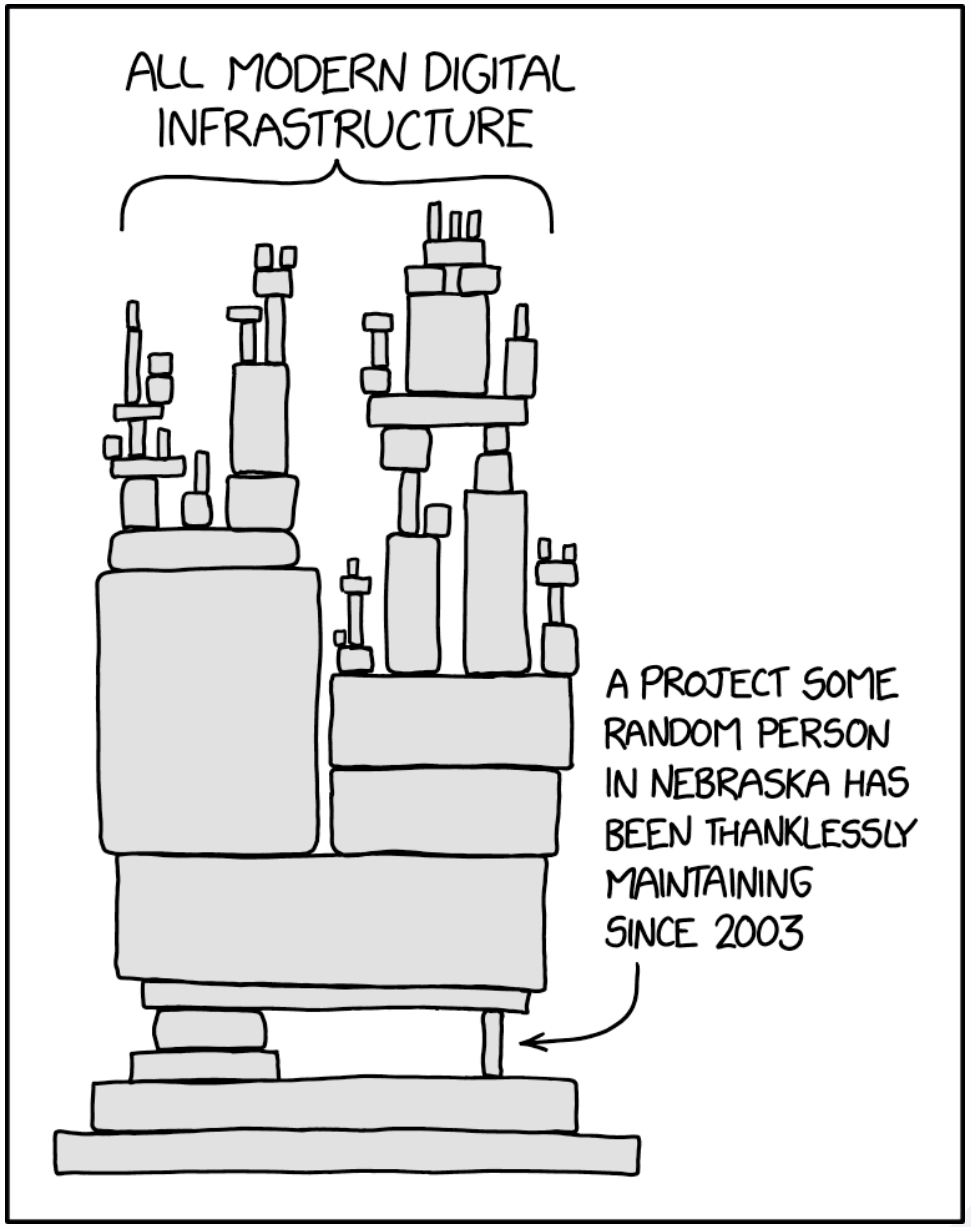
Not all are true positive

Contributing

Anyone can contribute!

Even the smallest contribution may be helpful :)

XKCD 2347



Up Next: React Native

React Native in 100 seconds

Questions?