Getting Started to Build Secure React Applications

Adhithi Ravichandran

Software Consultant, Author, Speaker

@AdhithiRavi www.adhithiravichandran.com

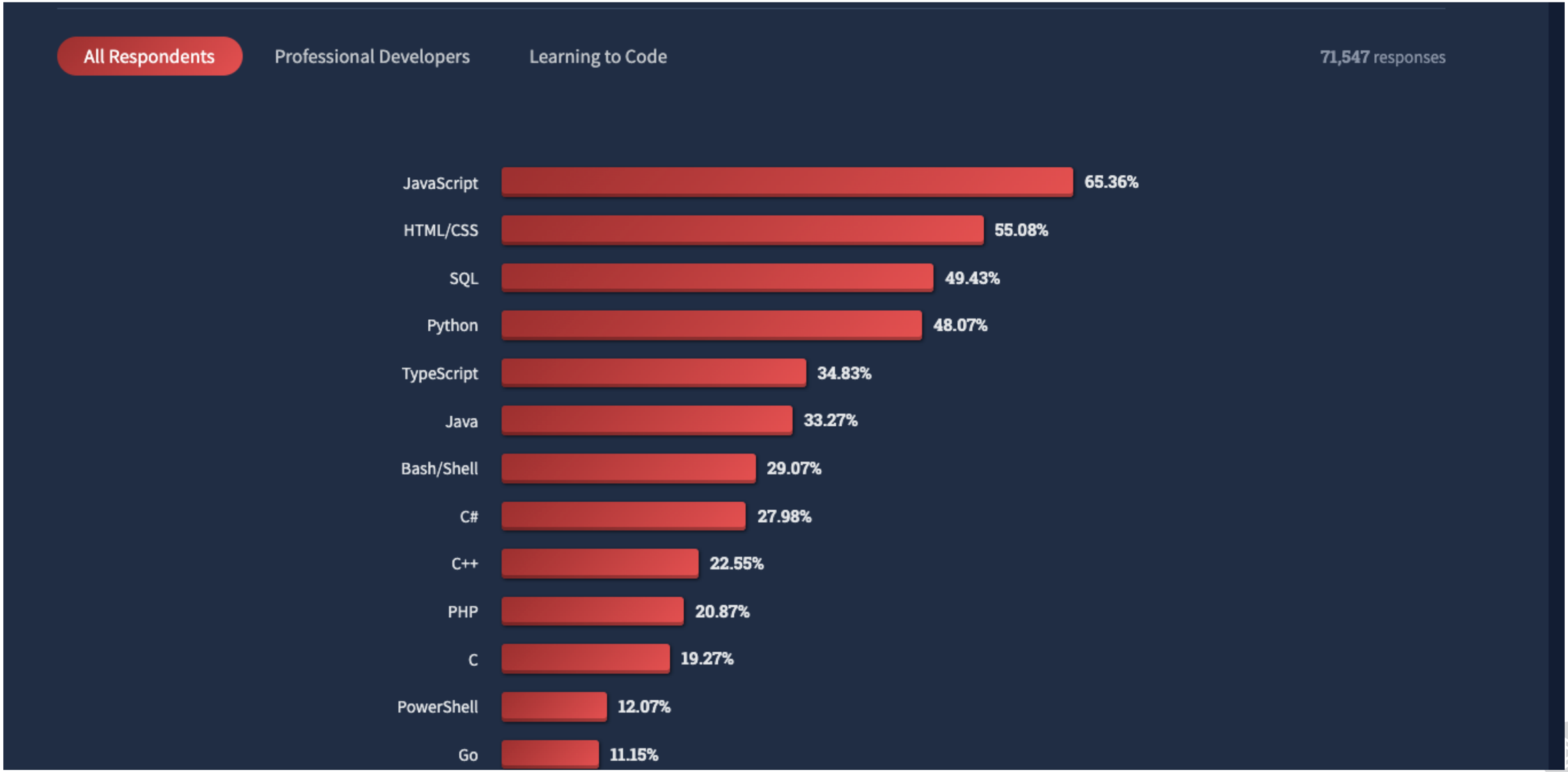
Prerequisites

- JavaScript
- ES 2017
- HTML
- CSS

Code in JavaScript

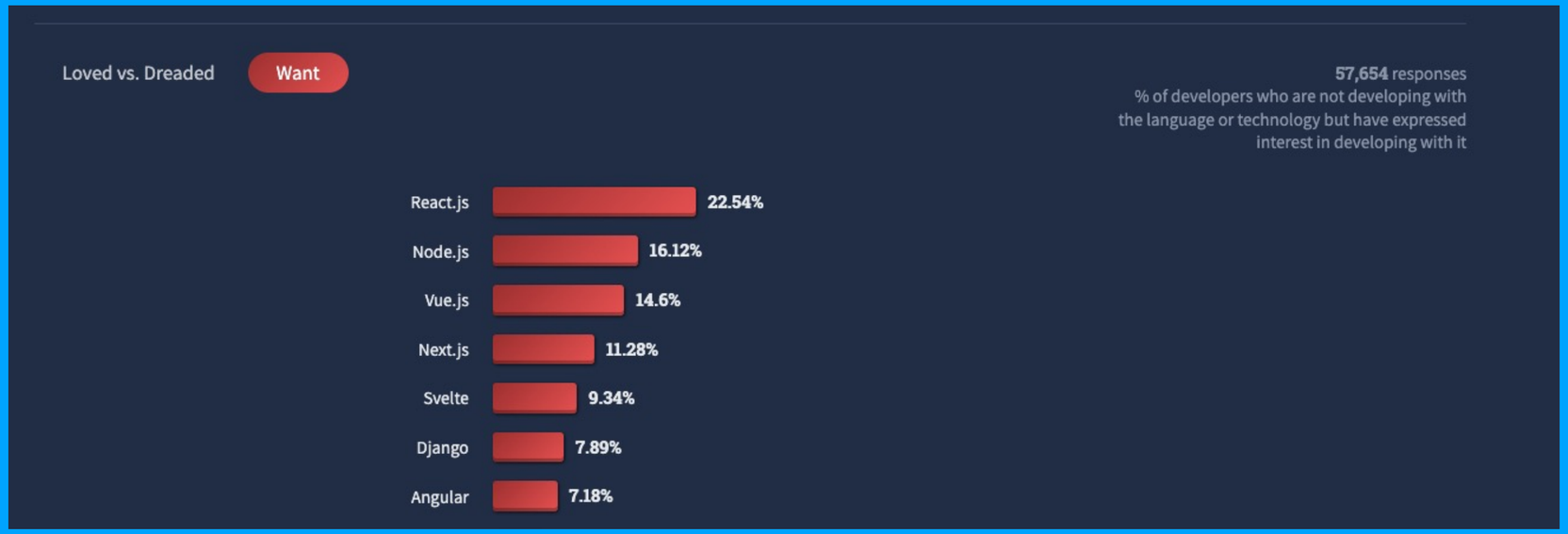
According to Stack Overflow's Annual Survey of 2022, for the 10th year in a row, JavaScript is ranked as the most commonly used programming language.

Stack Overflow Survey Results 2022



What is React?

React is a JavaScript library for building rich user interfaces.



Most Wanted Web Frameworks 2022

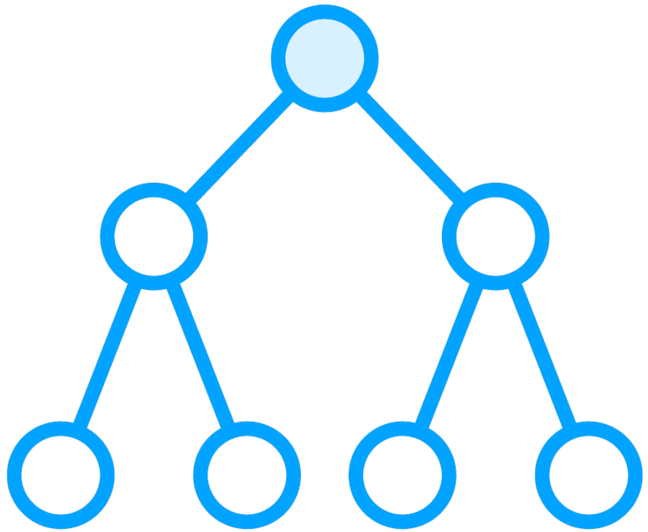
React.js completes its fifth year as most wanted
in the stack overflow developer survey!

Everything Is a Component

Component-based architecture

Reuse of components

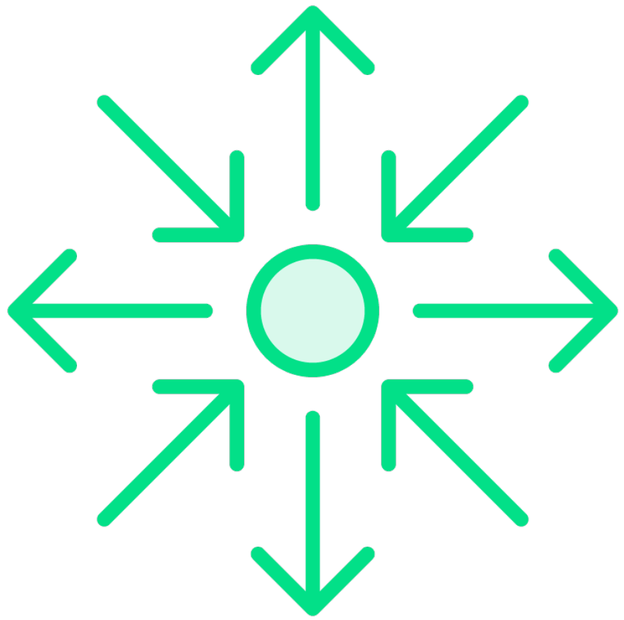
One directional data flow



Flexibility

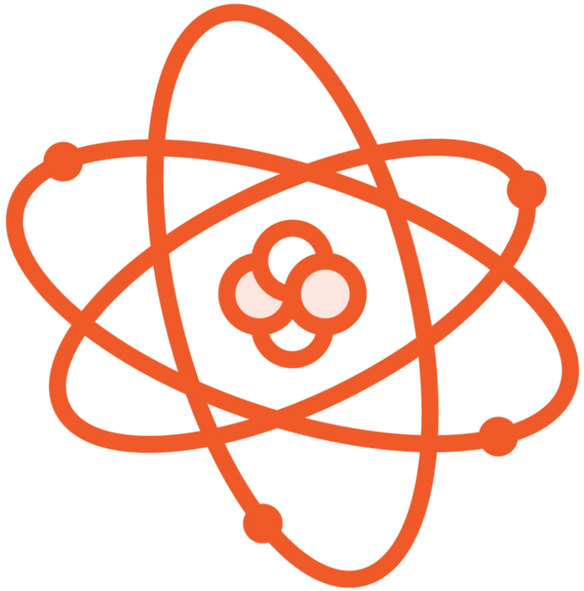
Offers lots of flexibility

Freedom to choose state management, navigation, server-side rendering and more



Framework vs. Library

Angular

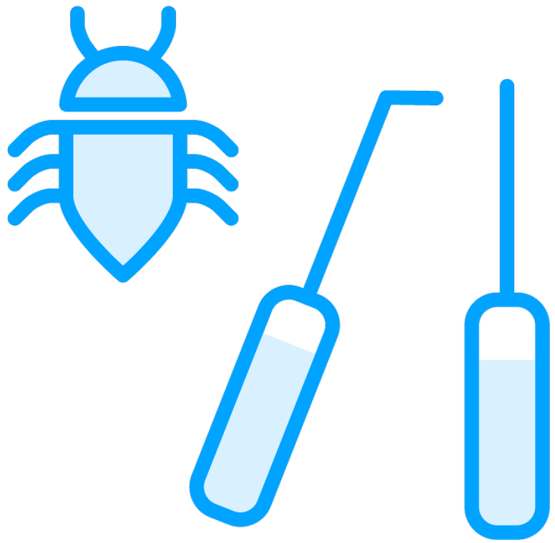


React

Testing Is Fun

Ease of testing

Plenty of testing options like Jest, Cypress, etc..

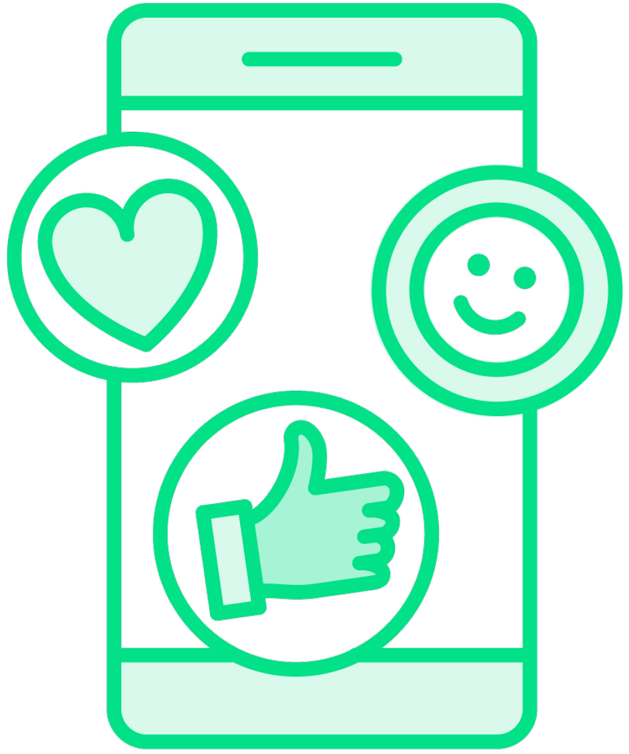


Builds Rich UI

Seamless UI

Highly responsive and rich UI

Quick loading times enabling a smooth user experience



Code Sharing

Code sharing between web and mobile

Saves cost of development and maintenance

React Components

Components

Components let you split the UI into independent, reusable pieces that are easier to write, test, and maintain.

Search Twitter

Home

Explore

Notifications

Messages

Bookmarks

Lists

Profile

More

Tweet

you | #Tokyo2020 | Trending | COVID-19 | News | Sports | Entertainme

Bloomberg Quicktake · August 13, 2021

Fencing gains popularity in Hong Kong after Olympic gold win

Bloomberg Quicktake · August 12, 2021

This Olympian won gold after a #Tokyo2020 volunteer paid for his taxi when he went to the wrong venue

Sports Insider · August 12, 2021

Meet the man in charge of timing the Olympics

Bloomberg Quicktake · August 9, 2021

Tokyo 2020 Olympians receive a hero's welcome from hometown fans

Who to follow

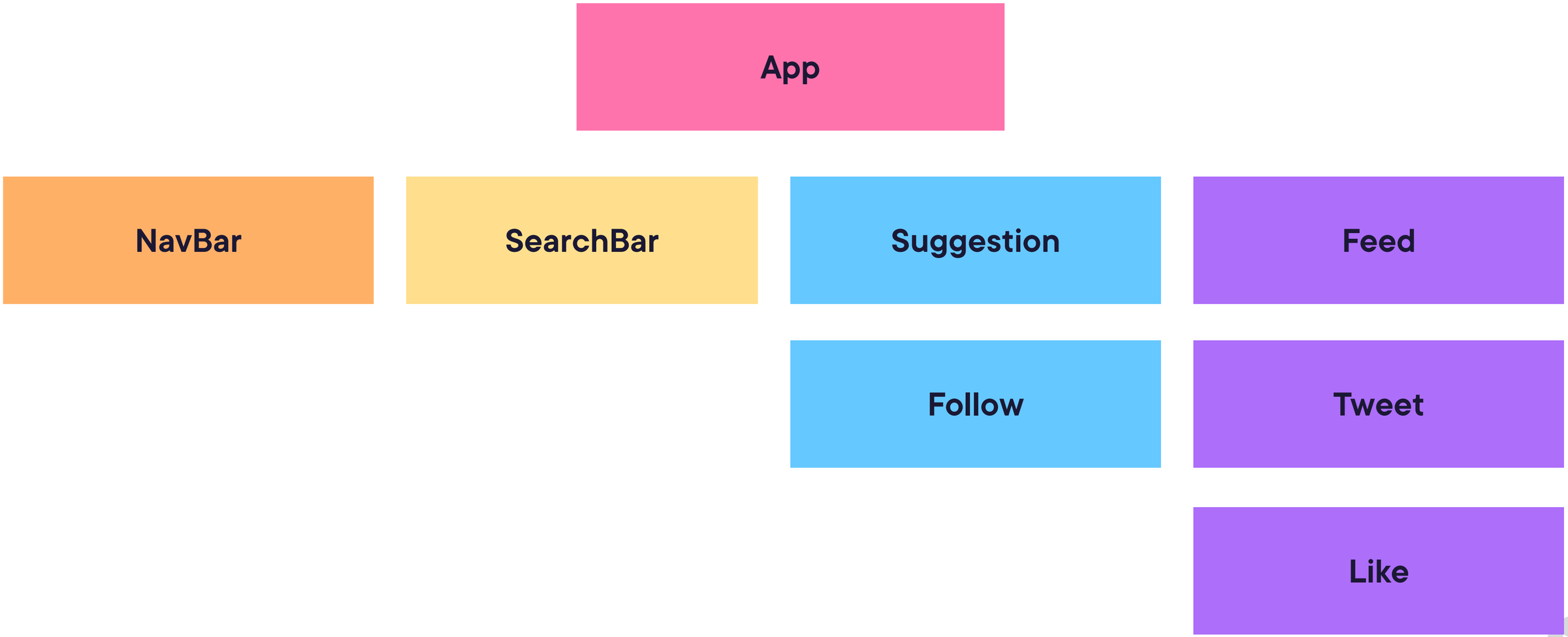
Mary Grygleski
@mgrygles
Follow

Developer Relations
@JobsInDevRel
Follow

Kate Inyeong Kim
@kateinkim
Follow

Show more

Components Structure



Each component is a building block that is a reusable piece of UI. Putting them together results in a complete application!

```
class Welcome extends React.Component {
  render() {
    return <h1>Hello Welcome to this presentation on React</h1>;
  }
}
```

React Component

Define a component as an ES6 class

```jsx
function Welcome() {
  return <h1>Hello Welcome to this presentation on React</h1>;
}
```

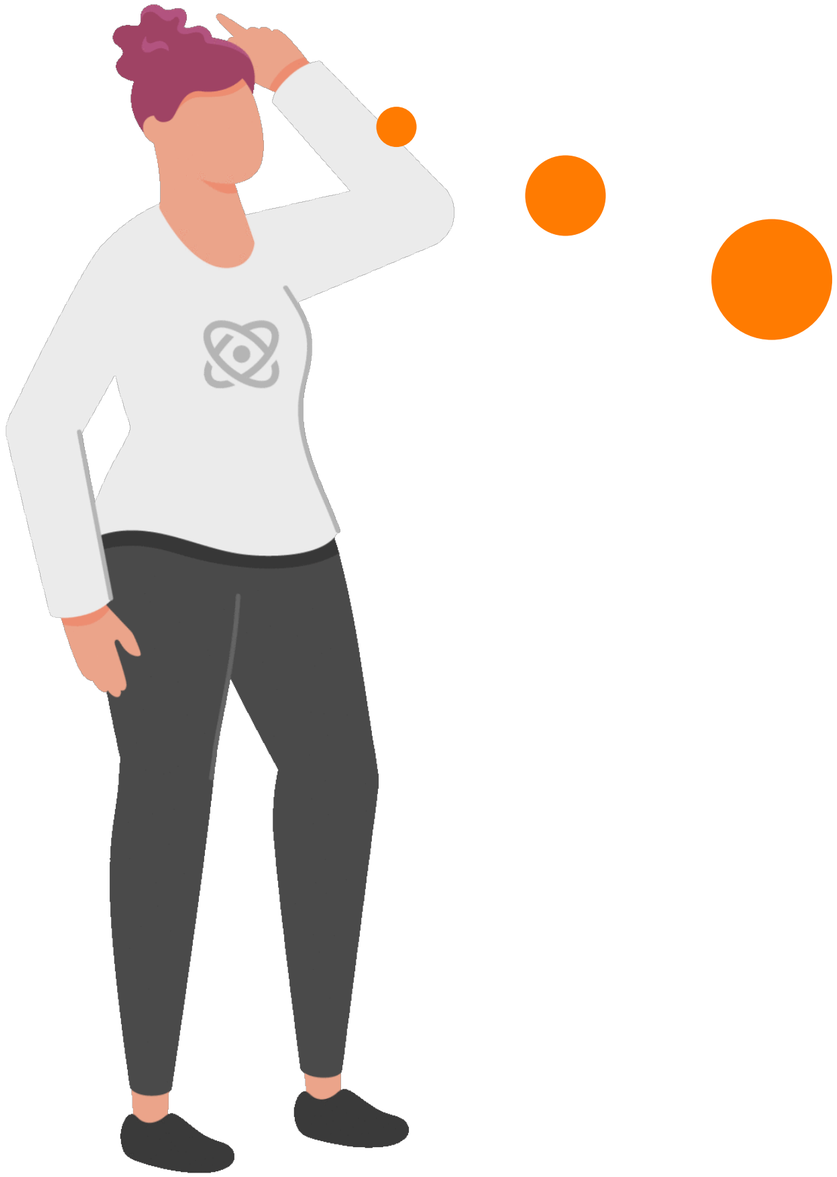
React Component

Define a component with a JS function.

```
function formatName(user) {
  return user.firstName + ' ' + user.lastName;
}

const user = {
  firstName: 'Harper',
  lastName: 'Perez',
};

const element = <h1>Hello, {formatName(user)}!</h1>;

ReactDOM.render(element, document.getElementById('root'));
```

JSX

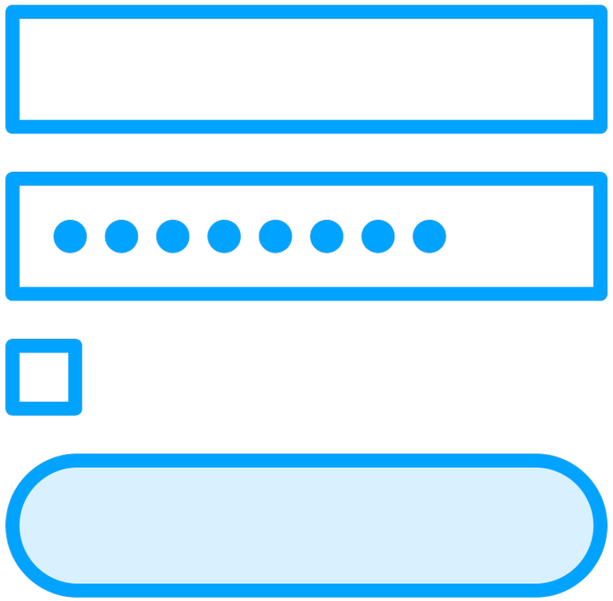
JSX - JavaScript Expressions, not HTML
It is a syntax extension to JavaScript
Can use JSX inside if statements, for loops, accept is as arguments, and return it from functions!



What are the best practices to build secure React apps?

1. Secure Authentication

Authentication



Validating if the app is accessed by the right user (Login)

Sign In, Register

Auth Provider – auth0

SDK available for both web and mobile (React and React Native)

Universal Login

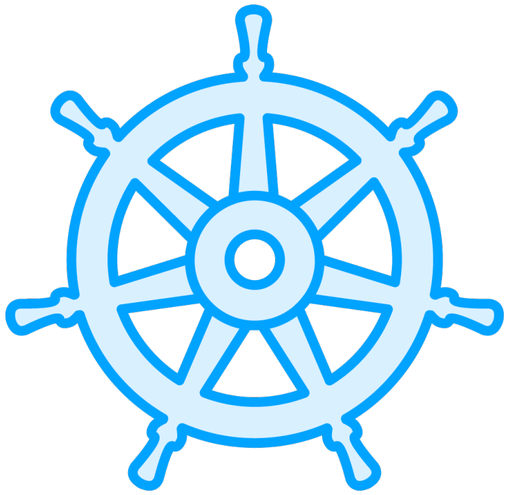
Single Sign On (SSO)

Multifactor authentication

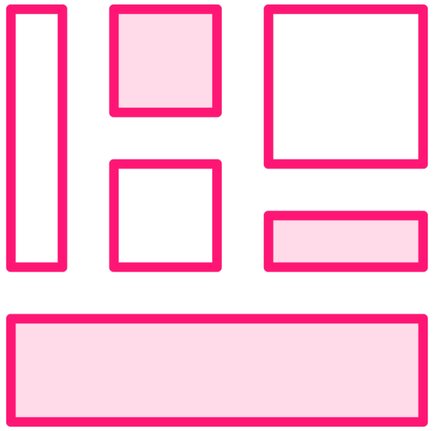
Passwordless authentication

Breached Passwords Detection

Benefits of Auth Provider?



You don’t need to re-invent the wheel!

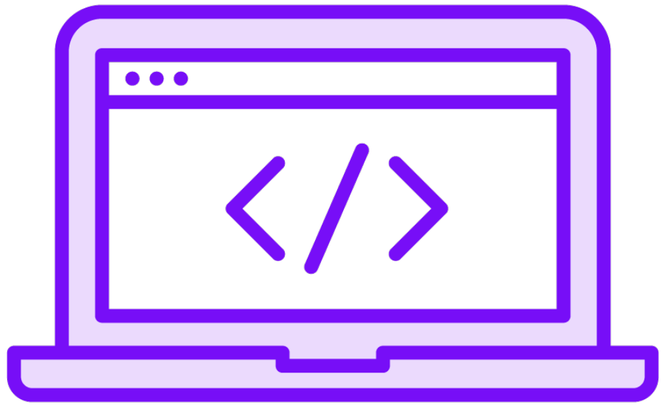


Customizable to your needs

Cost effective with free plans to start with

2. Ensure code is resilient

Resilient Code



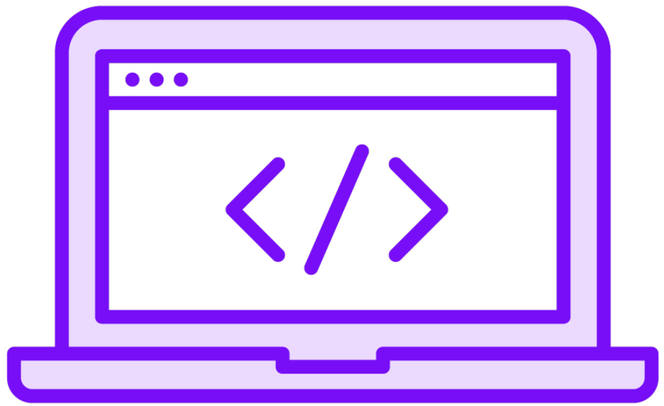
Use JSX Auto Escaping

- React automatically escapes all the strings you are displaying in order to prevent a wide range of XSS attacks.

Avoid Direct DOM Access

- Avoid using refs and findDomNode() to access rendered DOMelements to directly inject content via innerHTML and similar properties and methods.

Resilient Code



Avoid JSON injection attacks

- Common to send JSON data along with server-side rendered React pages.
- Always escape < characters with a benign value to avoid injection attacks.

Use Linter configurations

- Use ESLint React security config.
- Automatically detects security issues in code and offers advice.

3. Be vigilant while using 3rd party libraries

Vulnerabilities from Dependencies

Malicious packages with similar names to popular packages.

Dependency hell!

Dangerous library code (directly inserting HTML into DOM)

Vulnerabilities from Dependencies

Do not install libraries you do not need!

Vet libraries being added to your codebase.

Review packages using:

- Openbase

- Snyk Advisor

Use security linters on your node_modules to detect unsafe patterns in library code

react-day-picker

React DayPicker is a customizable date picker component for React, with native TypeScript support.

MIT TypeScript Definitions: Built-In

GitHub Stars	Weekly Downloads	Last Commit	User Rating	Top Feedback
4.7K	653K	8mos ago	4.6/5 (12)	9 Great Documentation 7 Easy to Use 4 Highly Customizable

react-datepicker

A simple and reusable datepicker component for React

MIT TypeScript Definitions: DefinitelyTyped

GitHub Stars	Weekly Downloads	Last Commit	User Rating	Top Feedback
6.8K	1.6M	8mos ago	4.4/5 (11)	17 Easy to Use 14 Easy to Use 13 Great Documentation

react-calendar

Ultimate calendar for your React app.

MIT TypeScript Definitions: DefinitelyTyped

GitHub Stars	Weekly Downloads	Last Commit	User Rating	Top Feedback
2.6K	374K	6mos ago	4.5/5 (21)	22 Great Documentation 22 Easy to Use 13 Highly Customizable

Search results for "react date time picker"

Search by algolia

rc-picker

3.0.0-4 published 7 days ago

Package Health Score 81 / 100

@dselmanovic/rc-picker

1.0.0 published 2 years ago

Package Health Score 48 / 100

react-calendar-mg

2.0.1 published 2 months ago

Package Health Score 57 / 100

tzolkin

2.1.1 published 4 years ago

Package Health Score 40 / 100

bonree-picker

1.0.1 published 1 month ago

Package Health Score 70 / 100

Snyk

Sign up

Can analyze projects (both public and private repos)

Run npm audit fix

Snyk Vulnerability Scanner - VS Code extension – Automatically fix vulnerabilities in source code

4. Storing sensitive data

Never store sensitive data in your React code

Store sensitive data in the server

API Keys

- Secure key with API provider
- Call API through server to obtain API keys

General Best Practices in React

TypeScript is a life saver!

Why TypeScript?

Catch problems early on

Intellisense is accurate

Easier to refactor code

Readable code

Easier to maintain and test

High quality documentation (TSDoc)

```tsx
interface MessageProps {
  text: string;
  important: boolean;
}

export const Message = ({ text, important }: MessageProps) => {
  return (
    <div>
      {important ? "Important message: " : "Regular message: "}
      {text}
    </div>
  );
};
```

Component in TypeScript

Interface defines the *props* that are accepted by the component using an object type

MessageProps is the interface that describes the *props* the component accepts

Break down your components – when needed!

When do I break down components?

Managing state is a nightmare

Performance concerns with re-rendering of application

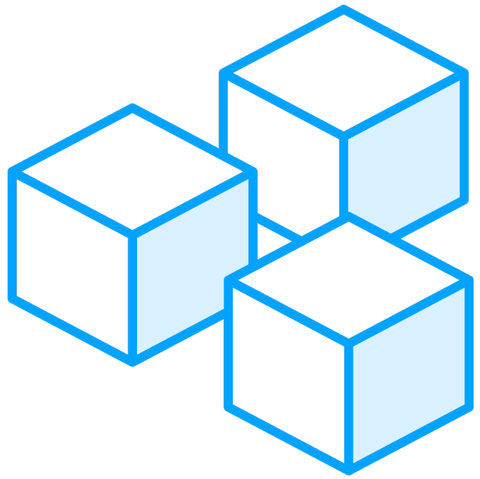
Code readability takes a hit

Working with multiple developers on the codebase becomes challenging

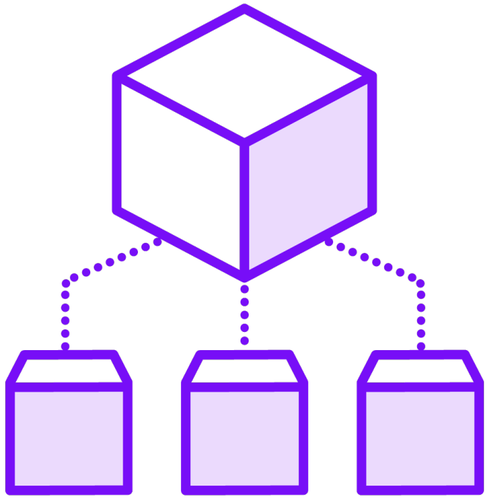
Testing code is harder

Start with local state!

State Management

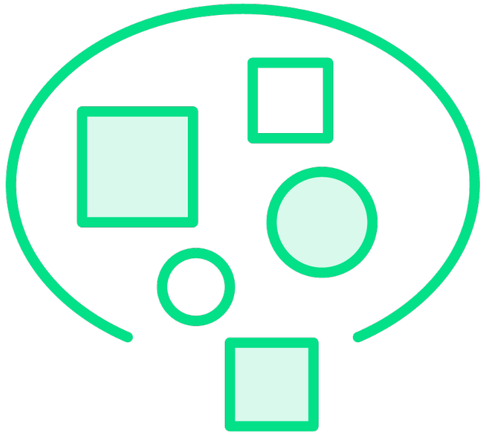


Start with local state first



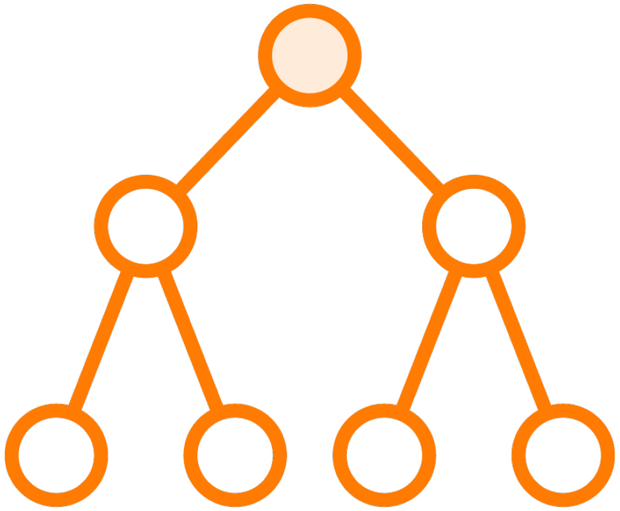
Pass down state via *props* if child component needs it

Lift state up if non-child component needs the data



Next choice: Context or external state management

useContext Hook



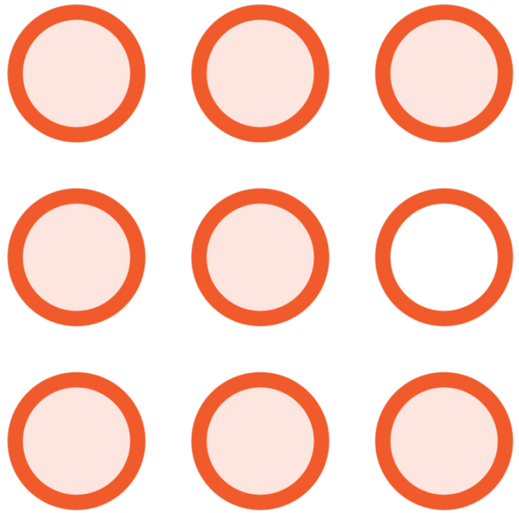
Context is designed to share data that is global for a tree of React components.

Avoid passing props through multiple layers of components.

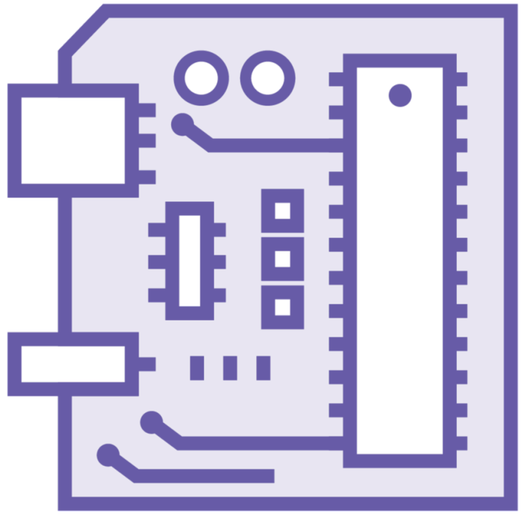
Examples: Theme, demographic information, language, etc..

Test, Test and Test!

Types of Testing



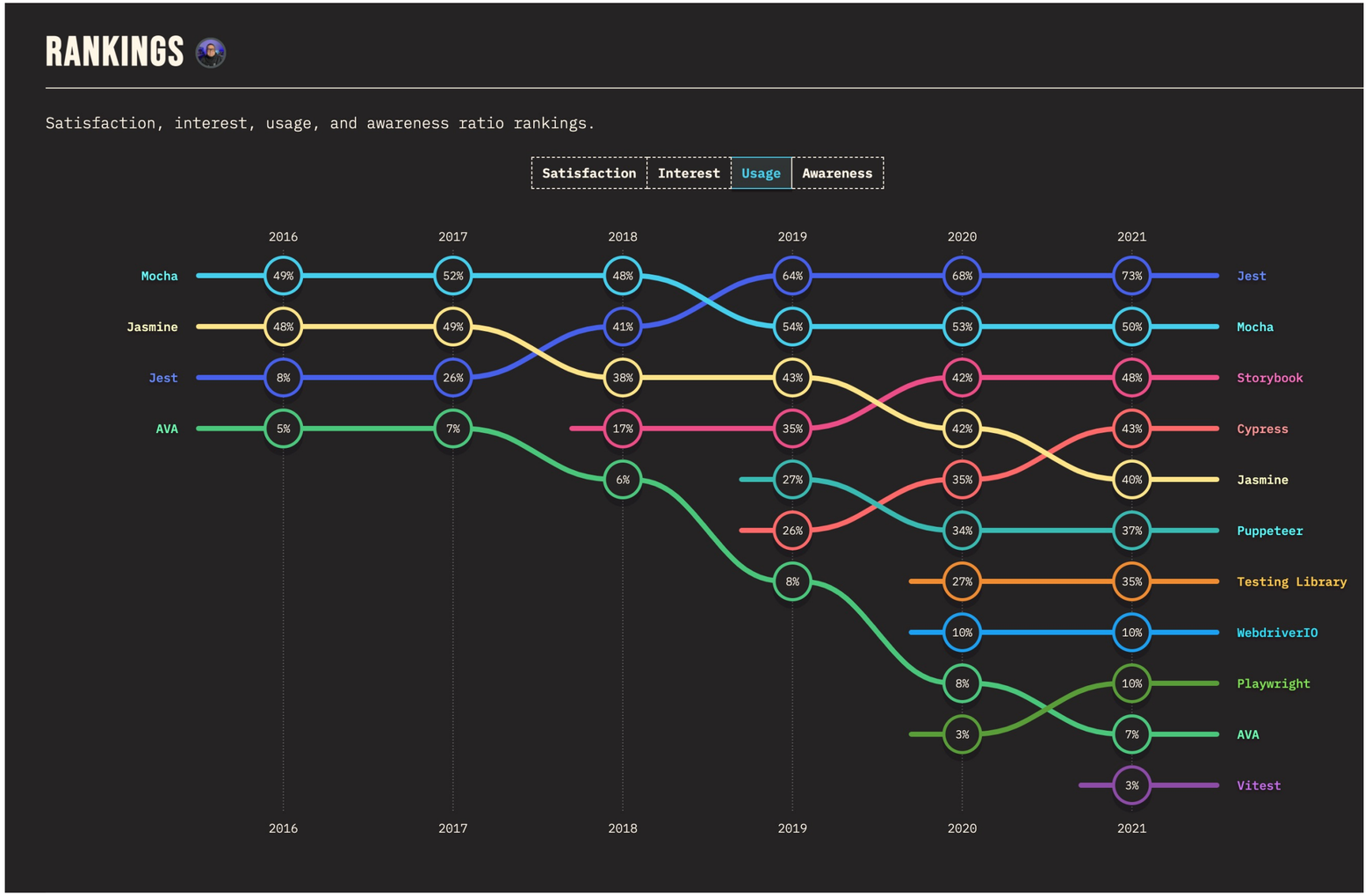
Unit testing - Jest



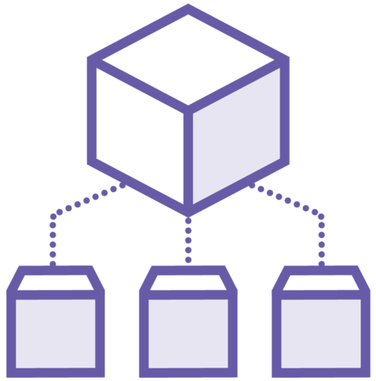
Component testing – React testing library

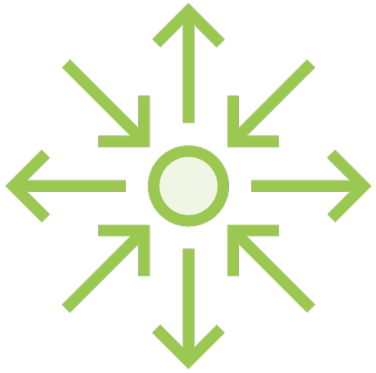
Automated End-to-end tests - Cypress

State of JS Survey 2021



React Framework?





Developer Decisions

- Routing
- Data fetching
- Server-side rendering
- Search Engine Optimization
- Image Optimization
- Bundling
- Code Splitting
- Deployment

What is Next.js?

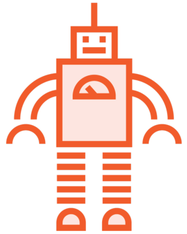
Next.js is a flexible React framework that gives you building blocks to create fast web applications

Next Features

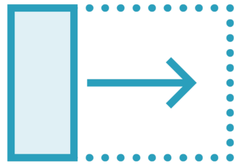
File-system Routing

Image and font optimization

Hybrid: SSG and SSR



Automatic bundle splitting and lazy loading

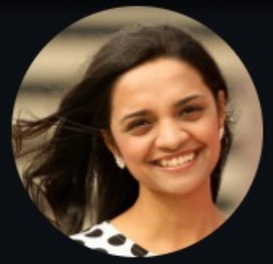


Incremental Static Regeneration

Automatic TypeScript support

Thank You!

@AdhithiRavi



Adhithi Ravichandran

Pluralsight Author

Following

439 Followers

Adhithi Ravichandran is a Software Consultant, Author and Speaker based in Kansas City. She is the owner and founder of Surya Consulting, Inc. through which she provides her expertise in Software Architecture, Development and Training. She provides clients, consulting services in architecting...

Show more...

adhithiravichandran.com

Twitter

LinkedIn

CONTENT AUTHORED

7

All time

TOPICS AUTHORED

GraphQL

Content authored

Cypress 9 Fundamentals NEW!

Course · Beginner · 1 hr 59m · May 23, 2022 · ★★★★½ (24)

Understanding Culture Intelligence

Course · Intermediate · 55m · Jul 11, 2021 · ★★★★☆ (15)

React Native: The Big Picture

Course · Beginner · 1 hr 3m · Jan 27, 2021 · ★★★★☆ (194)

React Native 0.63: Components Playbook

Course · Intermediate · 2 hr 31m · Jan 12, 2021 · ★★★★☆ (31)

Consuming a GraphQL API with Apollo Client 3 and React

Course · Intermediate · 2 hr 1m · Sep 23, 2020 · ★★★★½ (79)

Cypress 4: End-to-end JavaScript Testing

Course · Intermediate · 2 hr 9m · May 5, 2020 · ★★★★☆ (254)

GraphQL: The Big Picture

Course · Beginner · 1 hr 17m · Aug 7, 2019 · ★★★★½ (672)

Checkout my courses on Pluralsight on React Native, GraphQL, Cypress, etc..