

React

Client Side JS Library

Runs on the browser

Created and maintained by Facebook

Component based

More comparable to Angular than JQuery

Most people refer to it as a framework

Most common React, View JS, Angular

¿Why we use it?

Allow to build very powerful apps on the front end

Code Organization

Reusable components

Speed & performance

Makes things much easier

Very popular in the industry

Look at each element as individual components.

Organized structure

Logic, output & style

Components can have “Props” & “State”

Download and instal:

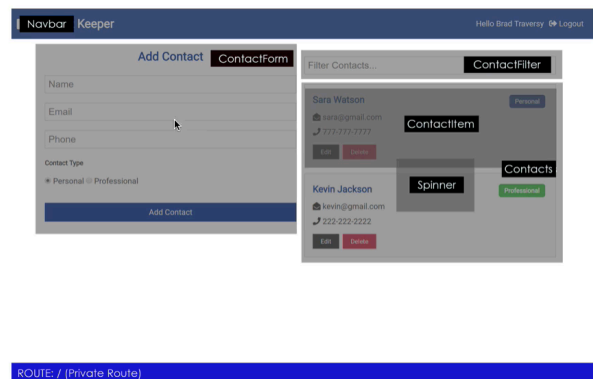
- Node JS
- GIT
- React Developer Tool Chrome Extension
- Redux DevTools Chrome Extension
- Postman

VSCoDe Extensions:

- ES7 React/Redux/GraphQL/React-Native snippets

Component Based UI

- Look at each element as individual components
- Organized structure
- Logic, output & style
- Components can have “props” & “state”



Project 1: Start

Code & Resources

The final code for this project can be found in this Github repo:

<https://github.com/bradtraversy/github-finder>

Docs for the Github API:

<https://developer.github.com/v3/>

To Register Your Github App & Get Keys:

<https://github.com/settings/applications/new>

API Endpoints We Will Be Working With:

<https://api.github.com/users>

<https://api.github.com/search/users?q=brad>

<https://api.github.com/users/bradtraversy>

<https://api.github.com/users/bradtraversy/repos>

`npx create-react-app`

Components could be functions or classes

JSX - JavaScript Syntax Extension

```
class App extends Component {  
  render() {  
    return (  
      <div className='App'>  
        <h1>Hello from React</h1>  
      </div>  
    );  
  }  
}
```

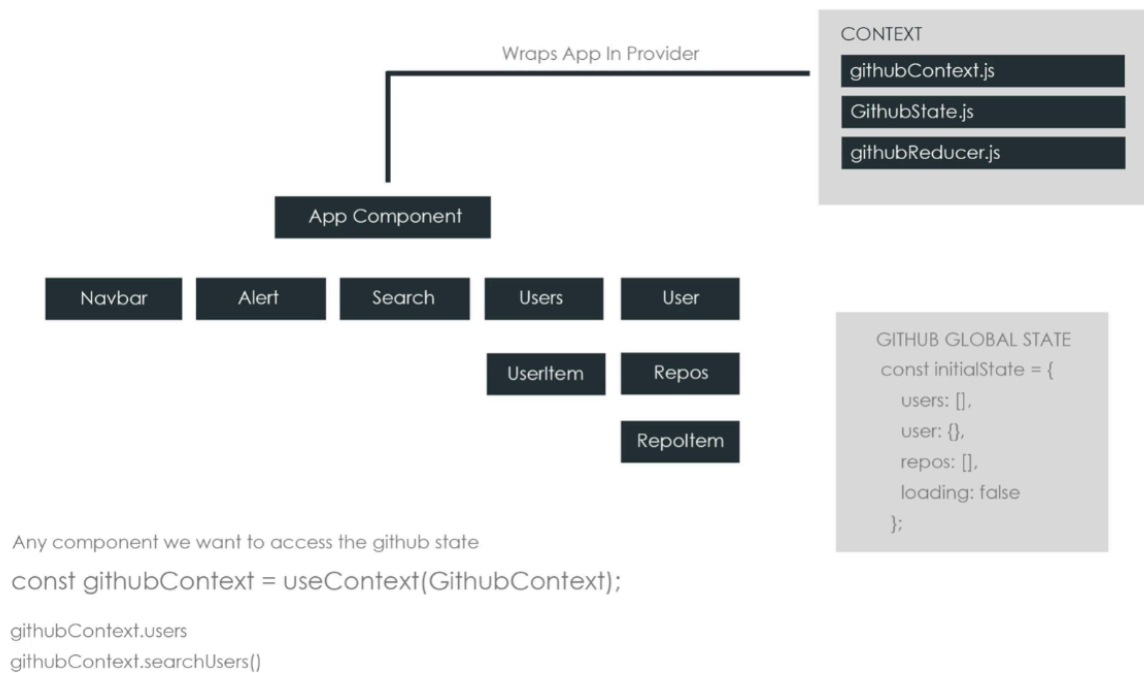
Has to have one Parent element

```
export default App;
```

```
class App extends Component {  
  render() {  
    return (  
      <React.Fragment className='App'>  
        <h1>Hello from React</h1>  
        <h2>GoodBye</h2>  
      </React.Fragment>  
    );  
  }  
}
```

Without JSX

```
▼ <body> == $0  
  <noscript>You need to enable  
    JavaScript to run this app.  
  </noscript>  
  ▼ <div id="root">  
    <h1>Hello from React</h1>  
    <h2>Goodbye</h2>  
  </div>
```



Hooks

Functions that let us hook into react state and lifecycle features from a function component.

Import { useState, useEffect } from 'react';

- useState
- useEffect
- useContext
- useReducer
- useRef...

Also you can create custom Hooks

2. Contact Keeper app

1. npm init -y
2. Configure package.json
3. Install dependencies:
 1. npm i express (web framework)
 2. bcryptjs (handle hashing passwords)
 3. jsonwebtoken (authentication)
 4. config (global variables)
 5. express-validator (validate email, passwords from user)
 6. mongoose (add, read, data, delete from DB)
4. Dev dependencies:
 1. Npm i -D nodemon (keep watching server, not manually server restart)
 2. concurrently (run front and back react server at the same time).
5. Add scripts package.json
<https://jwt.io/>

Front-End

Npx create-react-app client

Config json file

Npm run dev

NPM install dependencies

npm i axios for using http client

react-router-dom

uuid for creating id

react-transition-group for front-end animations

... spread operator

Redux

npm i redux react-redux redux-thunk redux-devtools-extension

<https://github.com/bradtraversy/it-logger/blob/master/src/actions/logActions.js>

<https://github.com/typicode/json-server>

<https://materializecss.com/>