React App Development

SENG 4640
Software Engineering for Web Apps
Winter 2023

Sina Keshvadi
Thompson Rivers University

Review

- React allows us to create web applications by developing reusable, modular components
- So far, we've seen how to define components within the HTML pages
- How do we develop larger applications with multiple components?

Node.js - Introduction

- **Node.js** is a free, open source platform and framework built in JavaScript

- Includes suite of tools that allows user to prepare JavaScript (and thus React) applications for deployment

- Utilizes Node.js Package Manager (**npm**) to install programs and manage dependencies

Node.js - Benefits

- Instead of including all JavaScript code in a **<script>** tag, now we can separate the components into different files to make code more modular

- Node.js allows us to incorporate dependencies of the code within the current file

```
var React = require('react');
var ReactDOM = require('react-dom');

import MyComponent from './MyComponent.js';
```

Node.js - Installation

- Navigate to https://nodejs.org/en/download/ and download and install Node.js and appropriate packages

- Although **npm** always comes with a Node.js installation, be sure to update the version to the most recent with the following command.

```
npm install npm -g
```

Creating a React App - Considerations

- Including dependencies (React, React-DOM libraries, etc.)
- Making code compatible with browsers that only support older versions of JavaScript
- Transforming JSX into JavaScript
- Modularity: implementing modules in separate files, bundling them as dependencies

Creating a React App with Node.js - Setup

- Fortunately, there exists a package (through npm) that takes all of the above into consideration when creating a React app
  - Incorporates **Babel** for JSX and ES6 transformation
  - Incorporates **Webpack** for bundling

```
npm install -g create-react-app
```

Creating a React App with Node.js - Setup

- To create new React app, run the following command in the desired parent directory of new application

```
npm install -g create-react-app
```

```
create-react-app my-app
```

on Unix systems, use “sudo” command

Anatomy of a React App

- **`package.json`:** information about app, lists of dependencies, shortcuts for scripts

- **`public`:** directory containing HTML files, images, other static web content

- **`src`:** directory containing JavaScript and CSS files

Starting React App with Node.js

- You can start the default app as follows:

```
cd my-app/
npm start
```

Starting React App with Node.js

- You can start the default app as follows:

```
cd my-app/
npm start
```

- This will start a web server that listens for incoming HTTP requests on port 3000 on your computer

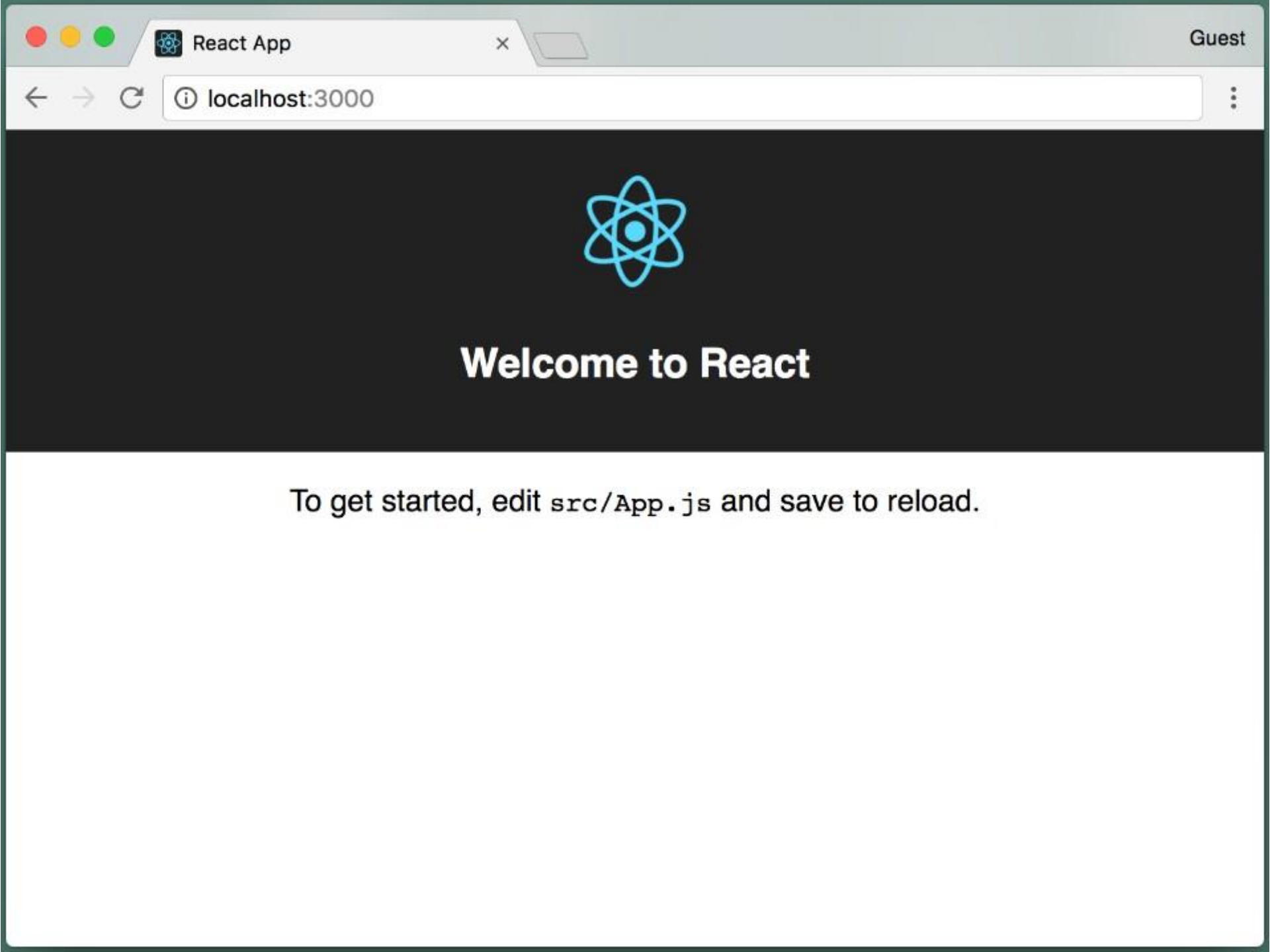
Starting React App with Node.js

- You can start the default app as follows:

```
cd my-app/
npm start
```

- This will start a web server that listens for incoming HTTP requests on port 3000 on your computer

- You can access the web server by accessing http://localhost:3000/ from your computer


React App
Guest
localhost:3000
Welcome to React
To get started, edit src/App.js and save to reload.

Incorporating Components

- We can now create separate JavaScript files for each component.
- **`src/Counter.js`** would look like this:

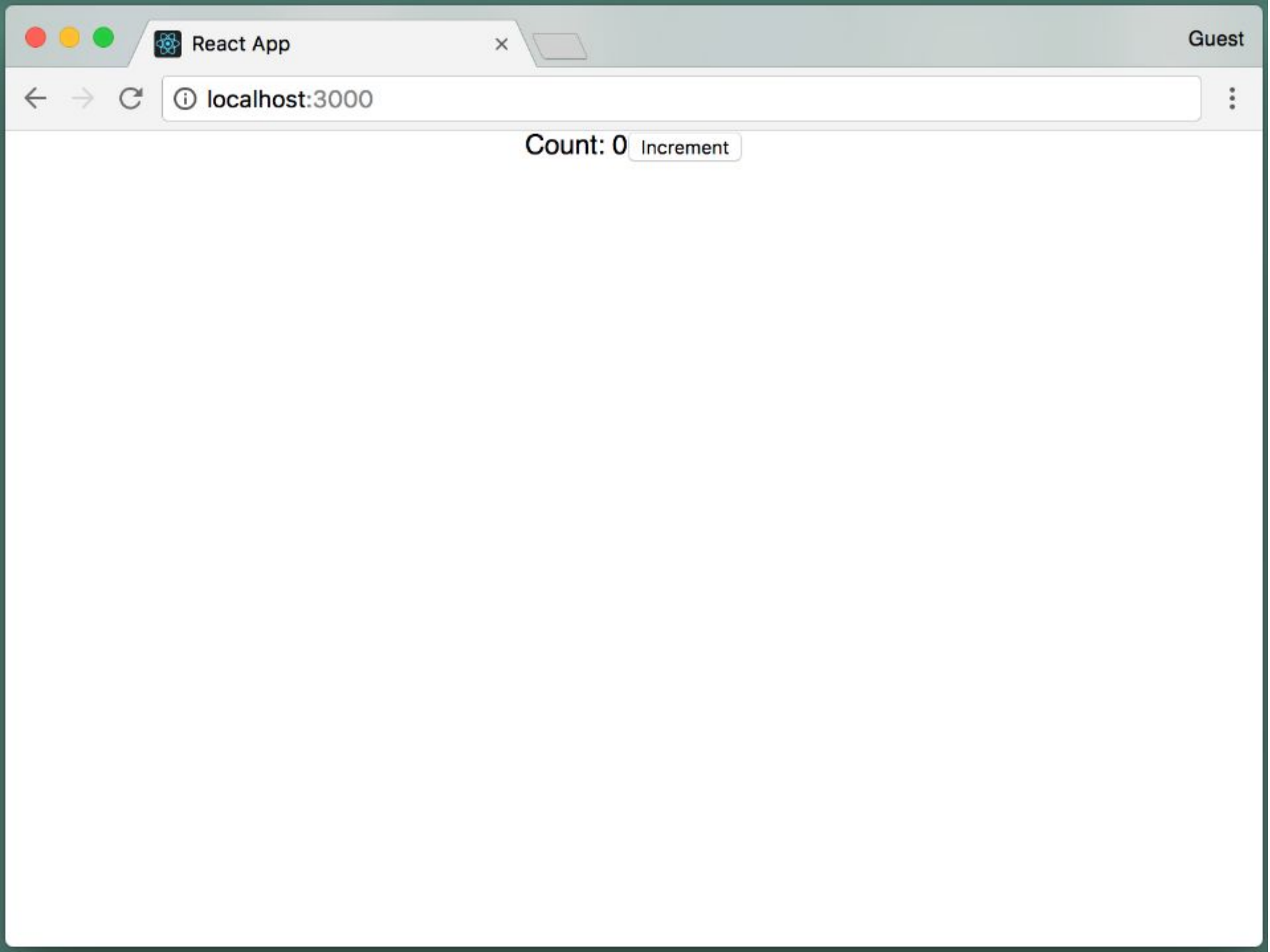
```
var React = require('react');

class Counter extends React.Component {
  constructor(props) { . . . }

  incrementCount() { . . . }

  render() { . . . }
};

export default Counter;
```

Incorporating Components into the App

- Edit **`src/App.js`** as follows:

```
import Counter from './Counter.js';
```

```
class App extends Component {
  render() {
   return (
      <div className="App">
         <Counter />
       </div>
   );
  }
}
```


React App
Guest
localhost:3000
Count: 0
Increment

Dog List example

React Dogs

Guest

localhost:3000

Good Dogs

Princess: Corgi X

Riley: Husky X

Add Dog

Name

Image

Breed

Submit

React Dogs

Guest

localhost:3000

Good Dogs

Princess: Corgi X

Riley: Husky X

Add Dog

Name

Cooper

Image

https://upload.wikimedia

Breed

Catahoula Leopard

Submit

React Dogs

Guest

localhost:3000

Good Dogs

Princess: Corgi X

Riley: Husky X

Add Dog

Name

Cooper

Image

https://upload.wikimedia

Breed

Catahoula Leopard

Submit

React Dogs

Guest

localhost:3000

Good Dogs

Princess: Corgi X

Riley: Husky X

Cooper: Catahoula Leopard X

Add Dog

Name

Cooper

Image

React Dogs

Guest

localhost:3000

Good Dogs

Princess: Corg X

Riley: Husky X

Cooper: Catahoula Leopard X

Add Dog

Name

Cooper

Image

React Dogs

Guest

localhost:3000/#

Good Dogs

Riley: Husky X

Cooper: Catahoula Leopard X

Add Dog

Name

Cooper

Image

https://upload.wikimedia

Breed

Catahoula Leopard

Submit

How do you handle the complexity of this web app?
How many component are you planning to create?

React Dogs

Guest

localhost:3000/#

Good Dogs

Riley: Husky X

Cooper: Catahoula Leopard X

Add Dog

Name

Cooper

Image

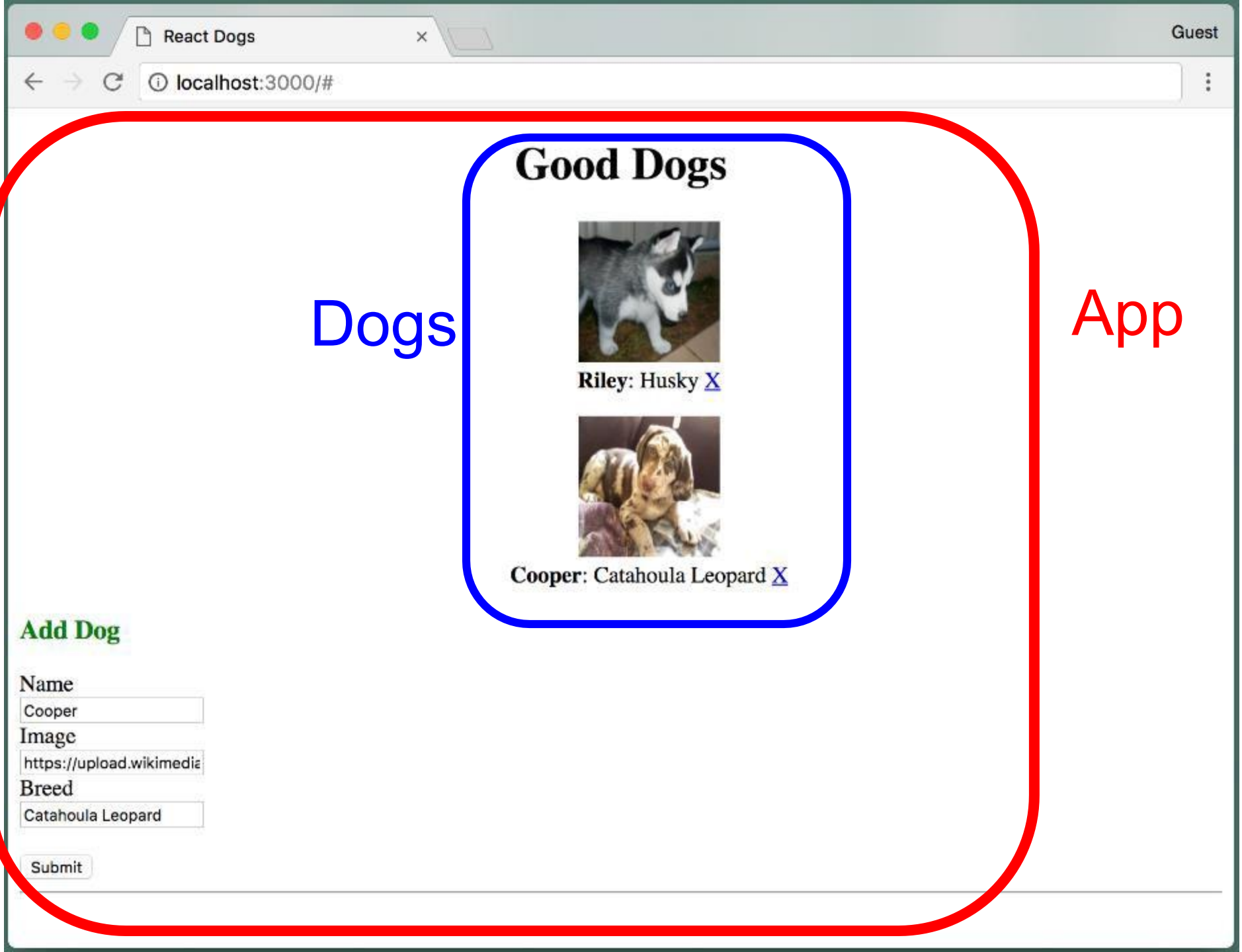
https://upload.wikimedia

Breed

Catahoula Leopard

Submit

App


React Dogs
Guest
localhost:3000/#
Good Dogs
Riley: Husky X
Cooper: Catahoula Leopard X
Dogs
App
Add Dog
Name
Cooper
Image
https://upload.wikimedia
Breed
Catahoula Leopard
Submit

React Dogs

Guest

localhost:3000/#

App

Dogs

Good Dogs

Riley: Husky X

Cooper: Catahoula Leopard X

DogItem

Add Dog

Name

Cooper

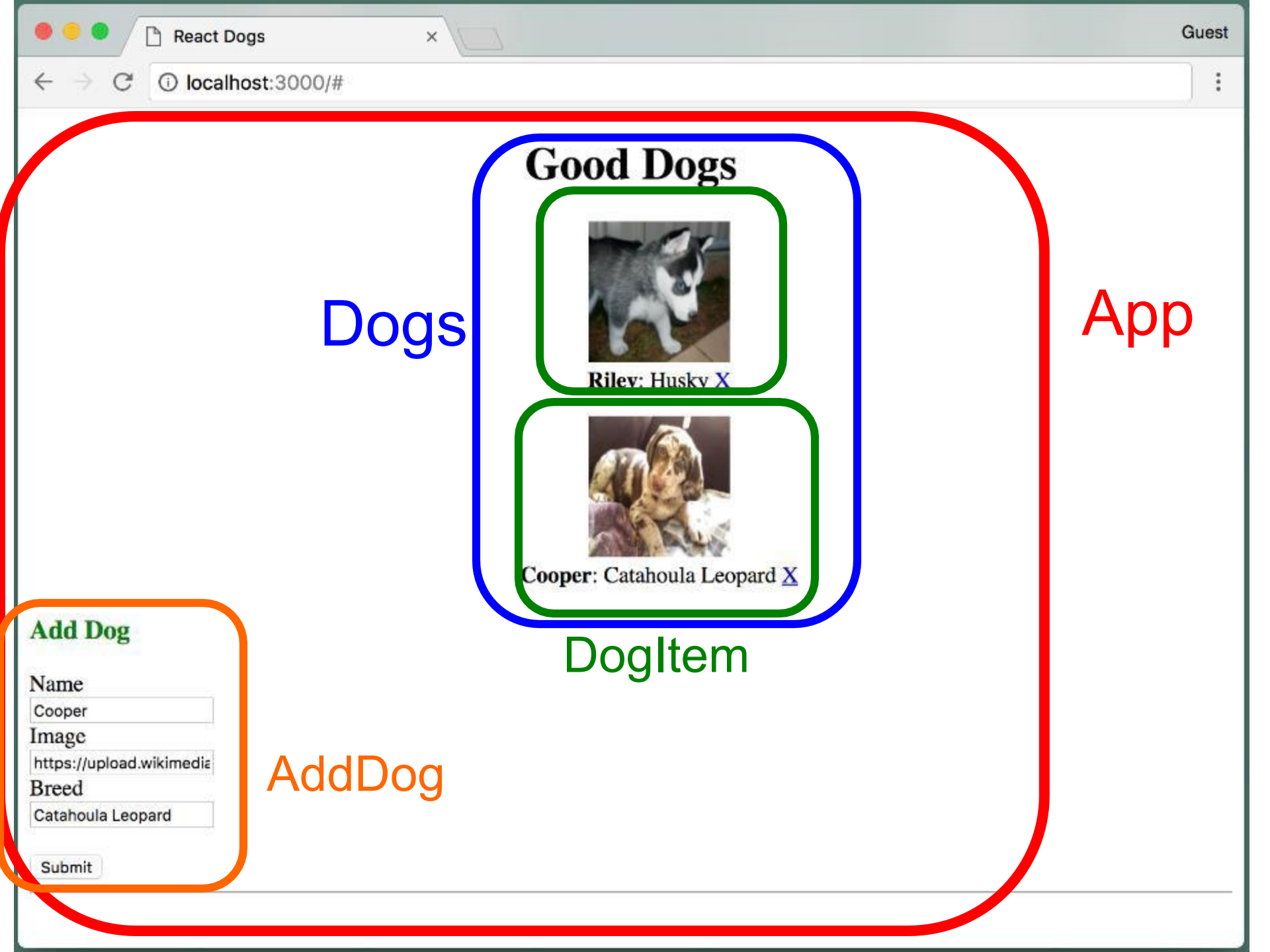
Image

https://upload.wikimedia

Breed

Catahoula Leopard

Submit


React Dogs
Guest
localhost:3000/#
Good Dogs
Riley: Husky X
Cooper: Catahoula Leopard X
Dogs
DogItem
App
Add Dog
Name
Cooper
Image
https://upload.wikimedia
Breed
Catahoula Leopard
Submit
AddDog

what would be your initial step?

```
import React, { Component } from 'react';
import Dogs from './components/Dogs';
import AddDog from './components/AddDog';
import './App.css';

class App extends Component {
  constructor() { . . . }
  handleAddDog(dog) { . . . }

  handleDeleteDog(name) { . . . }

  render() {

     return (
      <div className="App">
      <Dogs dogs={this.state.dogs}
            onDelete={this.handleDeleteDog.bind(this)} />
      <AddDog onAddDog={this.handleAddDog.bind(this)} />
      <hr />
      </div>
    );
   }
};
```

```
import React, { Component } from 'react';
import Dogs from './components/Dogs';
import AddDog from './components/AddDog';
import './App.css';

class App extends Component {
  constructor() { . . . }
  handleAddDog(dog) { . . . }

  handleDeleteDog(name) { . . . }

  render() {

     return (
      <div className="App">
      <Dogs dogs={this.state.dogs}
            onDelete={this.handleDeleteDog.bind(this)} />
      <AddDog onAddDog={this.handleAddDog.bind(this)} />
      <hr />
      </div>
    );
   }
};
```

```
import React, { Component } from 'react';
import Dogs from './components/Dogs';
import AddDog from './components/AddDog';
import './App.css';

class App extends Component {
  constructor() { . . . }
  handleAddDog(dog) { . . . }

  handleDeleteDog(name) { . . . }

  render() {

     return (
      <div className="App">
      <Dogs dogs={this.state.dogs}
            onDelete={this.handleDeleteDog.bind(this)} />
      <AddDog onAddDog={this.handleAddDog.bind(this)} />
      <hr />
      </div>
    );
  }
};
```

```
import React, { Component } from 'react';
import Dogs from './components/Dogs';
import AddDog from './components/AddDog';
import './App.css';

class App extends Component {
  constructor() { . . . }
  handleAddDog(dog) { . . . }

  handleDeleteDog(name) { . . . }

  render() {

     return (
      <div className="App">
      <Dogs dogs={this.state.dogs}
            onDelete={this.handleDeleteDog.bind(this)} />
      <AddDog onAddDog={this.handleAddDog.bind(this)} />
      <hr />
      </div>
    );
  }
};
```

```
import React, { Component } from 'react';
import Dogs from './components/Dogs';
import AddDog from './components/AddDog';
import './App.css';

class App extends Component {
  constructor() { . . . }
  handleAddDog(dog) { . . . }

  handleDeleteDog(name) { . . . }

  render() {

     return (
      <div className="App">
      <Dogs dogs={this.state.dogs}
            onDelete={this.handleDeleteDog.bind(this)} />
      <AddDog onAddDog={this.handleAddDog.bind(this)} />
      <hr />
      </div>
    );
  }
};
```

```
import React, { Component } from 'react';
import Dogs from './components/Dogs';
import AddDog from './components/AddDog';
import './App.css';

class App extends Component {
  constructor() { . . . }
  handleAddDog(dog) { . . . }

  handleDeleteDog(name) { . . . }

  render() {

     return (
      <div className="App">
      <Dogs dogs={this.state.dogs}
            onDelete={this.handleDeleteDog.bind(this)} />
      <AddDog onAddDog={this.handleAddDog.bind(this)} />
      <hr />
      </div>
    );
   }
};
```

```
import React, { Component } from 'react';
import Dogs from './components/Dogs';
import AddDog from './components/AddDog';
import './App.css';

class App extends Component {
  constructor() { . . . }
  handleAddDog(dog) { . . . }

  handleDeleteDog(name) { . . . }

  render() {

     return (
      <div className="App">
      <Dogs dogs={this.state.dogs}
            onDelete={this.handleDeleteDog.bind(this)} />
      <AddDog onAddDog={this.handleAddDog.bind(this)} />
      <hr />
      </div>
    );
  }
};
```

```
import React, { Component } from 'react';
import Dogs from './components/Dogs';
import AddDog from './components/AddDog';
import './App.css';

class App extends Component {
  constructor() { . . . }
  handleAddDog(dog) { . . . }

  handleDeleteDog(name) { . . . }

  render() {

     return (
      <div className="App">
      <Dogs dogs={this.state.dogs}
            onDelete={this.handleDeleteDog.bind(this)} />
      <AddDog onAddDog={this.handleAddDog.bind(this)} />
      <hr />
      </div>
    );
   }
};
```


React Dogs
Guest
localhost:3000/#
Good Dogs
Riley: Husky X
Cooper: Catahoula Leopard X
Dogs
App
Add Dog
Name
Cooper
Image
https://upload.wikimedia
Breed
Catahoula Leopard
Submit
AddDog