React.js

When, Where and How to Use It

Brad Marsh @bbqhacker

- Senior Front End Developer at SmartThings
- Hobbyist Game Developer

Resources for this talk

Slides and code: https://github.com/dottertrotter/MidwestJs

What is React

View framework from Facebook

http://facebook.github.io/react/index.html

High Level Benefits

- Automatically reduces constant DOM updating by keeping a virtual DOM and only updating those parts that have changed or those that you explicitly say to update
- Produces highly reusable components
- Provides a “pure” javascript templating option

What React Is Not

- React is not an all inclusive MVC framework like Angular
  - Controllers can be created in whatever fashion you choose
  - Components have states and props, but are not Models

Hello World

- http://0.0.0.0:3000/examples/hello

JSX

Docs describe JSX in the following way:

“JSX lets you create JavaScript objects using HTML syntax.”

http://facebook.github.io/react/docs/displaying-data.html#jsx-syntax

JSX

- For speedy development react provides the JSXTransformer that interprets the JSX as javascripts at run time
- For production environments JSX should be compiled to straight javascript
- It is possible to use React without using JSX

Components

- React Classes are capitalized to make the XML compatible and so it is not mistaken for an HTML tag
- React Class render method returns a SINGLE element
- Single element can contain many nested elements including other components

Components (continued)

- When finished loading will call:
  - componentDidMount

Hello World Take 2

- http://0.0.0.0:3000/examples/hello-2

State and Props

- States are component data that the component sets itself via:
  - getInitialState
  - this.setState
- Props are select pieces of data that are passed to child components from a parent and are immutable by the child
  - <HelloWorld message=”Hello Cruel World” />

State and Props (continued)

- When deciding whether to use state or props
  - http://facebook.github.io/react/docs/interactivity-and-dynamic-uis.html#what-components-should-have-state

Question?

Isn’t having all of my html contained within my javascript counterproductive or anti MVC?

JSX vs Handlebars

- From http://handlebarsjs.com/

```
<div class="entry">
  {{!-- only output this author names if an author exists --}}
  {{#if author}}
    <h1>{{firstName}} {{lastName}}</h1>
  {{/if}}
</div>
```

- http://0.0.0.0:3000/examples/user-info

Add Ons

A separate build of React with some addition “experimental” utility features.

- Animation
- Performance Tools
- Testing
- etc.
- http://facebook.github.io/react/docs/addons.html

Tic Tac Toe

A slightly more complex Hello World

http://0.0.0.0:3000/

Annoyances - Nothing’s Perfect

- Lists must contain keys
- Requiring render to return a single element tends to lead to extra DOM elements
- I tend to forget JSX isn’t HTML and thus forget to use attributes such as className

React Native

Offshoot of the React project that allows you to use javascript to write native iOS applications

https://facebook.github.io/react-native/

Additional Reading

- In depth tutorial on creating a voting site using React, Mongo DB and Socket.IO
  - http://sahatyalkabov.com/create-a-character-voting-app-using-react-nodejs-mongodb-and-socketio/
- React.js Introduction For People Who Know Just Enough jQuery To Get By
  - http://reactfordesigners.com/labs/reactjs-introduction-for-people-who-know-just-enough-jquery-to-get-by/

Additional Reading (continued)

- Angular vs. React - the tie breaker
  - https://www.airpair.com/angularjs/posts/angular-vs-react-the-tie-breaker

Questions

Hopefully you have some.