

create-react-app

From Getting Started to
Advanced Configuration

March 27, 2017

<https://github.com/nanohop/react-indy-march-2017>

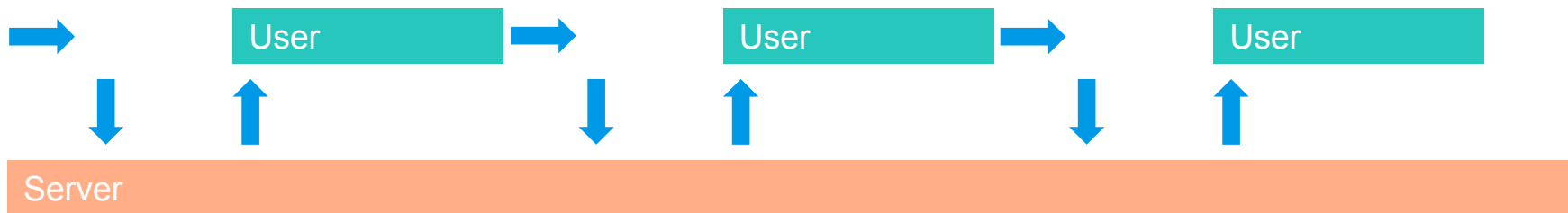
create-react-app

<https://github.com/facebookincubator/create-react-app>

- “Create React apps with no build configuration.”
 - Historically setup was very time consuming
 - Only a single dev dependency*
- This is not the only way
- Single Page Application

Single Page Applications

Traditional Web Application



Single Page Application



What You Get

- Babel: JSX, ES6
- Webpack: dev server, build script
- ESLint
- Jest for testing
- Autoprefixed CSS, declare dependencies in js
- Ability to 'eject' for advanced configuration

Installing

Need Node 4+, 6+ recommended: <https://nodejs.org>

- Doesn't assume Node backend - server can be anything
-

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

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

```
cd my-app/
```














```
npm start
```











Then open <http://localhost:3000>

DEMO

step1

Folder Structure

- ▼  public
 -  favicon.ico
 -  index.html
- ▼  src
 -  App.css
 -  App.js
 -  App.test.js
 -  index.css
 -  index.js
 -  logo.svg
-  .gitignore
-  package.json
-  README.md

- ▼  src
 - ▶  actions
 - ▶  api
 - ▶  components
 - ▶  images
 - ▶  pages
 - ▶  reducers
 - ▶  tests
 -  index.css
 -  index.js

Installing Dependencies

- Install npm packages
 - `npm install react-router-dom --save`
 - `npm install react-bootstrap --save`
 - `npm install bootstrap@3 --save`
- Link css in src/index.js:
 - `import 'bootstrap/dist/css/bootstrap.css';`
 - `import 'bootstrap/dist/css/bootstrap-theme.css';`

Using Dependencies

```
import { BrowserRouter as Router, Route, Link } from 'react-router-dom';  
import { Navbar, Nav, NavItem } from 'react-bootstrap';
```

```
    <Router>  
      <div>  
        <Navbar>  
          <Nav>  
            <ListItemLink to="/">Home</ListItemLink>  
            <ListItemLink to="/about">About</ListItemLink>  
          </Nav>  
        </Navbar>  
  
        <Route exact path="/" component={Home}/>  
        <Route path="/about" component={About}/>  
      </div>  
    </Router>
```

Absolute Imports

```
import logo from '../images/logo.svg';  
import '../.../styles/App.css';  
import apiUtils from '../.../utils/api_utils.js'
```

```
import logo from 'pages/App/images/logo.svg';  
import 'styles/App.css';  
import apiUtils from 'utils/api_utils.js'
```

- Make a .env file and add:
 - NODE_PATH=src

DEMO

step2

Making a “Real” App

The screenshot shows a web application with a light gray header containing 'Home' and 'About' links. On the left, a sidebar has a blue 'People' button and a list of names: 'John Doe', 'Sara Test', and 'Chris'. The main area displays three input fields, each with a name label above it: 'Sara Test' (containing 'Hello!'), 'John Doe' (containing 'Hello!'), and 'Chris' (containing 'Hello!'). Below these is a wide empty input field and a green 'Send' button.

- Connection to server
- Props and state
- Lifecycle methods
- ES6 syntax
- Different ways to style
- Component breakdown

Server Connection

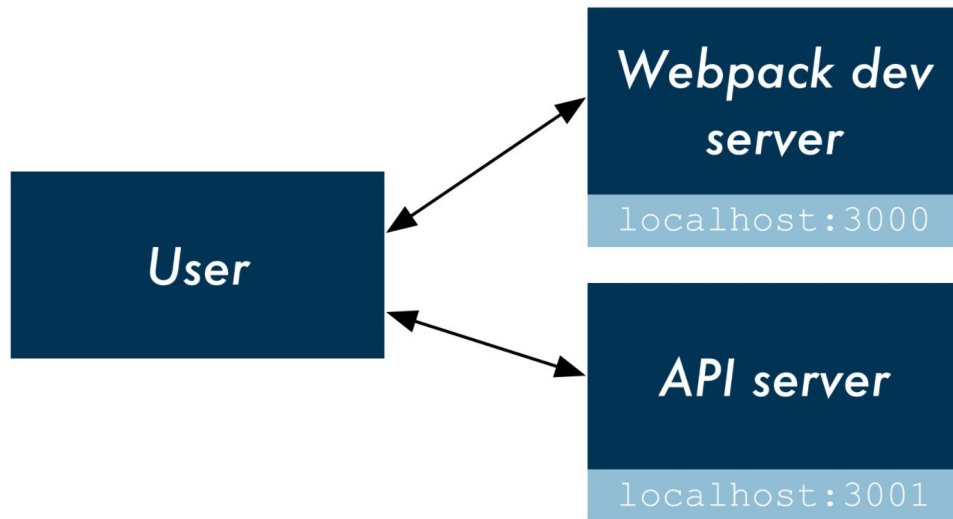
- Want to do something like this:

```
fetch(`/api/people.json`)
```

- Problem: in dev, /api doesn't exist

Problem

<https://www.fullstackreact.com/articles/using-create-react-app-with-a-server/>



Solution

<https://www.fullstackreact.com/articles/using-create-react-app-with-a-server/>



- Turn on proxy in package.json:
 - "proxy": "http://localhost:3001/"

APIs in Production?

- Two choices:
 - Have production server serve bundle
 - Enable CORS

DEMO

step3

Testing

- npm run test
- Uses Jest
 - Can use Enzyme, Chai, Sinon, etc
- Runs any .test.js, .spec.js, or in __test__ folder

DEMO

step4

Packaging for Production

- npm run build
- Generates a static build folder with assets

DEMO

step5

Eject

- Option to “Eject”
 - Allows you to modify configuration
- Permanent change to project
- `npm run eject`
- General wisdom: avoid if possible

DEMO

step6

Extra Credit

Hot Module Replacement

ReactCasts #7 - Hot Module Replacement in Create-React-App

<https://www.youtube.com/watch?v=f-ctxG2qEps&t=353s>

- Without ejecting
 - Lose internal state with every change
 - You might be able to preserve redux state*
- With ejecting
 - Full hot module replacement while keeping state

Progressive Web App

<https://github.com/jeffposnick/create-react-pwa>

- Adds a service worker
- Adds a web app manifest

Electron App

<https://medium.freecodecamp.com/building-an-electron-application-with-create-react-app-97945861647c#.xr2q4e94w>

Google: “create-react-app electron”

- No ejecting necessary

Questions?