

### Web App Design with React Week 15 Coding Assignment

Points possible: 75

URL to GitHub Repository: <a href="https://github.com/dmoylan1/week15Project.git">https://github.com/dmoylan1/week15Project.git</a>

**Instructions:** In VS Code, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document, with your JavaScript project code, to the repository. Add the URL for this week's repository to this document where instructed and submit this document to your instructor when complete.

### **Coding Steps:**

1. Using the Houses API, or any open API of your choice you can find online, create a single page that allows for all 4 crud operations to be performed on a resource from that API. Create a React component (or more, if needed) to represent the resource. Make all forms and other necessary UI pieces their own components as reasonable.

#### **Screenshots of Code:**



### App.js

```
JS team.js U
src > JS App.js > ♣ App > ♠ addPlayer
       import React from "react";
       import './App.css';
       import Team from "./team";
       const TEAM_ENDPOINT = 'https://631be1534fa7d3264ca3334f.mockapi.io/teams_api/team';
       export default class App extends React.Component {
        constructor(props) {
           super(props);
           this.addPlayer = this.addPlayer.bind(this);
           this.deletePlayer = this.deletePlayer.bind(this);
         render() {
           const teams = this.state
             ? this.state.teams.map((team, index) =>
             <Team
               key={index}
               data={team}
               addPlayer = {this.addPlayer}
               deletePlayer = {this.deletePlayer} />)
               : null;
               return (
                   {teams}
                 </div>
         componentDidMount() {
           fetch(TEAM ENDPOINT)
             .then(res => res.json())
             .then(data => {
```

```
JS App.js M X JS team.js U X JS index.js M JS new-player-form.js U
src > JS App.js > ♣ App > ♠ addPlayer
                 teams:data
               });
             });
         deletePlayer(e, team, player) {
           const index = team.players.indexOf(player);
           team.players.splice(index, 1);
           updateTeam(team)
             .then(() => {
               this.setState(state => {
                 for (let t of state.teams) {
                   if (t._id === team._id) {
                     let t = team;
                     break;
                 return state;
               });
             });
             e.preventDefault();
         addPlayer(e, team, player) {
           team.players.push(player)
           updateTeam(team)
             .then(() => {
               this.setState(state => {
                 for (let t of state.teams) {
                   if (t._id === team._id) {
                     let t = team;
                     break;
```





### Team.js

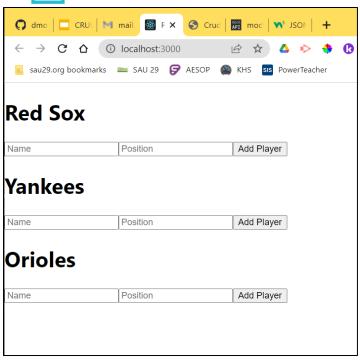
```
JS team.js U X JS index.js M
                                               JS new-player-form.js U
src > Js team.js > 😝 Team > 🛇 render
      import React from "react";
      import NewPlayerForm from "./new-player-form";
      export default class Team extends React.Component {
          render() {
              const players = this.props.data.players
                   ? this.props.data.players.map((player, index) =>
                   key={index}>
                       {player.name} Position: {player.position}
                       <button onClick={e =>
                       this.props.deletePlayer(e, this.props.data, player)
                   }>Delete/button>
                   )
                   : null;
                  return (
                           <h1>{this.props.data.name}</h1>
                               {players}
                           <NewPlayerForm
                               addPlayer={this.props.addPlayer} data={this.props.data} />
                       </div>
```

New-player-form.js

```
JS team.js U
                                  JS index.js M JS new-player-form.js U X
src > JS new-player-form.js > 😝 NewPlayerForm > 🛇 handleClick > 🔑 name
       import React from "react";
       export default class NewPlayerForm extends React.Component {
            constructor(props) {
                super(props);
                this.state = {
                     nameValue: '',
                     positionValue: ''
                this.handleNameChange = this.handleNameChange.bind(this);
                this.handlePositionChange = this.handlePositionChange.bind(this);
                this.handleClick = this.handleClick.bind(this);
            handleNameChange(e) {
                this.setState({nameValue: e.target.value});
            handlePositionChange(e) {
                this.setState({positionValue: e.target.value});
            handleClick(e) {
                this.props.addPlayer(e, this.props.data,
                     {name: this.state.nameValue, position: this.state.positionValue});
                this.setState({nameValue: '', positionValue: ''});
  > JS new-player-form.js > ધ NewPlayerForm > 🕅 handleClick > 🔑 name
                 <input type="text" placeholder="Name" onChange={this.handleNameChange} value={this.state.nameValue} />
                 <input type="text" placeholder="Position" onChange={this.handlePositionChange} value={this.state.positionValue} />
                 <button onClick={this.handleClick}>Add Player</button>
```

**Screenshots of Running Application:** 





I can add a player, but cannot get it to display. Error is 'cannot read properties of undefined (reading 'push')