### 2. How will the application handle state?

Generally speaking handling state involves:

* create constructor in stateful components and initialize state there, like:  
  this.state = { stateItem: value <, more key-value pairs> }
* make sure each stateful component class stores information as state
* make sure there are stateless components that display that state (‘**sibling**’); passing the state to it like:  
  <Video src={this.state.src}/>  
  and displaying the state in it like:  
  <video src={this.props.src} />
* make sure that for each stateful component class a *different* (‘**child**’) component class (exists that) displays a way to change that state, passing a method to do that with to it like:  
  <Menu chooseVideo={this.chooseVideo}/>  
  and make sure the method is called by some event there (tie eventHandler to an eventListener), like clicking a button, like:  
  <form onClick={this.props.handleClick}>   
  handleClick? yes handleClick! handleClick is a new method that must be written now in order to extract the content from the event (with event.target.value for instance) that needs to be passed upon calling chooseVideo(), e.g. like:  
  handleClick(e) { this.props.chooseVideo(e.target.value); }
* make sure to bind any method that will be passed to another component and that contains a reference to this. to the this of the component where it is defined!, like: <form onClick={**this.handleClick.bind(this)**}> or (usually) done in the constructor, like:  
  this.chooseVideo = this.chooseVideo.bind(this);
* if you need eventHandler methods (to pass information between components) make sure they look like this, containing this.setState call and placed outside of the components class’ render() method (because .setState() itself calls render()!).  
  class Cat extends React.Component {   
   speak() { this.setState({ noise: ‘Meow’ }); }  
   render() { … }   
  }
* make sure that if a stateful component needs to pass a prop to a stateless component, the stateless component exports its component class, like:  
  export class Tree extends React.Component  
  and the stateful component imports that component class, like:   
  import { Tree } from ‘./Tree’
* if a stateful component needs to pass a prop (with a piece of state) to a stateless component, that must be done within its render() method.

Now specifically…

App will be a stateful component, storing information:

* a list of songs being searched, say ‘searchResultTracks’ and
* a list of songs to be stored on Spotify as a playlist, say ‘playListTracks’ (I prefer a name that is a plural of something instead of playlist here, so items can referred to using the singular version of the same word).

SearchBar will be the child component, changing (parent) App’s state of ‘searchResultTracks’. SearchBar will itself maintain state also however, for the term being searched, say ‘term’.

The App’s state of ‘playListTracks’ however will be changed by clicking inside the Track components, either within the SearchResultTracks- or PlayListTracks-container components. Items will be deleted from the ‘playListTracks’ in the PlayListTrack components and added to ‘playListTracks’ in the SearchResultTrack components. The method to do that will need to be passed from ‘parent’ App through SearchResultTracks respectively PlayListTracks to SearchResultTrack respectively PlayListTrack.

App’s state info will be shown in stateless components (siblings) SearchResultTracks and PlayListTracks, that will in turn create instances of (stateless components) respectively SearchResultTrack and PlayListTrack to render each item in them.