

React Redux Youtube Schema

```
index.js

import _ from 'lodash';
import React, {Component} from 'react';
import ReactDOM from 'react-dom';
import YTSearch from 'youtube-api-search';
import SearchBar from './components/search_bar';
import VideoList from './components/video_list';
import VideoDetail from './components/video_detail';
const API_KEY = 'GOOGLE_API_KEY';

class App extends Component{

  constructor(props){
    super(props);

    this.state = {videos:[], selectedVideo: null};
    this.videoSearch('canadian news');
  }

  videoSearch(term){
    YTSearch({key:API_KEY, term: term}, (videos) => {
      console.log(videos);
      this.setState({ videos: videos, selectedVideo: videos[0] });
    });
  }

  render() {
    const videoSearch = _.debounce((term) => { this.videoSearch(term) }, 1000);

    return (
      <div>
        <SearchBar onSearchTermChange={videoSearch} />
        <VideoDetail video={this.state.selectedVideo}/>
        <VideoList videos={this.state.videos} onVideoSelect={selectedVideo => this.setState({selectedVideo})} />
      </div>
    );
  }
}

ReactDOM.render(<App />, document.querySelector('.container'));
```

/src/components

```
search_bar.js

import React, {Component} from 'react';

class SearchBar extends Component {

  constructor(props){
    super(props);

    this.state = {term: ''};
  }

  render() {
    return (
      <div className="search-bar">
        <input value={this.state.term} onChange={event => this.onInputChange(event.target.value)} />
      </div>
    );
  }

  onInputChange(term){
    console.log(term);
    this.setState({term});
    this.props.onSearchTermChange(term);
  }

}

export default SearchBar;
```

```
video_list.js

import React, {Component} from 'react';
import VideoListItem from './components/video_list_item';

const VideoList = (props) => {

  const videoItems = props.videos.map(video => {
    return <VideoListItem key={video.etag} video={video} onVideoSelect={props.onVideoSelect} />
  });

  return (
    <ul className="col-md-4 list-group">
      {videoItems}
    </ul>
  );
};

export default VideoList;
```

```
video_detail.js

import React from 'react';

const VideoDetail = ({video}) => {

  if(!video){
    return <div>Loading...</div>;
  }

  const videoId = video.id.videoId;
  const url = `https://www.youtube.com/embed/${videoId}`;

  return(
    <div className="video-detail col-md-8">
      <div className="embed-responsive embed-responsive-16by9">
        <iframe className="embed-responsive-item" src={url}></iframe>
      </div>
      <div className="details">
        <h3 className="title">{video.snippet.title}</h3>
        <div className="description">{video.snippet.description}</div>
      </div>
    </div>
  );
};

export default VideoDetail;
```

```
video_list_item.js

import React from 'react';

const VideoListItem = ({video, onVideoSelect}) => {

  const imageUrl = video.snippet.thumbnails.default.url;

  return (
    <li className="list-group-item" onClick={() => onVideoSelect(video)}>
      <div className="video-list media">
        <div className="media-left">
          <img className="media-object" src={imageUrl} />
        </div>
        <div className="media-body">
          <div className="media-heading">{video.snippet.title}</div>
        </div>
      </div>
    </li>
  );
};

export default VideoListItem;
```