20180420 :

**Used technologies**:

* React
* ES6
* Webpack
* Babel

**prop-types** : <https://github.com/facebook/prop-types#prop-types>

white-app\admin-on-rest-demo-master

Envir : babel-polyfill, admin-on-rest fwk : ES6, [React](https://facebook.github.io/react/) and [Material Design](https://material.io/)

* TODO Supports any authentication provider (REST API, OAuth, Basic Auth, ...)

TODO Vs Angular https://github.com/marmelab/ng-admin

Source Forum : <https://stackoverflow.com/questions/tagged/reactjs>

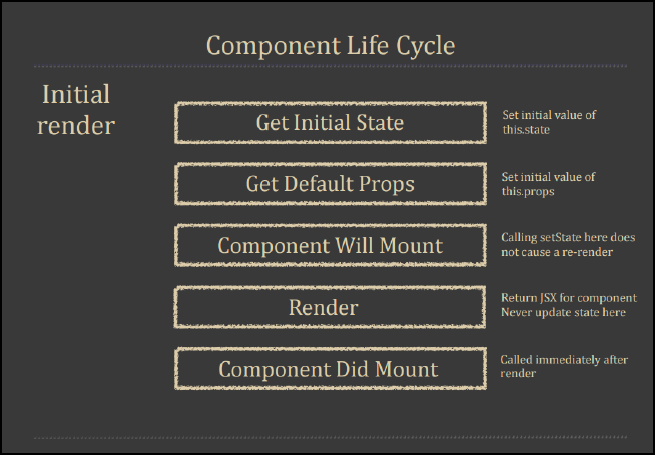
Microservice :

Source : <https://stackoverflow.com/search?q=%5Breactjs%5D+microservice>

 A client connect to my (Java) Web API Gateway using a browser, the Web API Gateway will call each (Java) microservice to get their JSON data and return it to the UI component that made the request on the client.

Source 1 : https://reactjs.org/tutorial/tutorial.html

Cycle de vie d’un composant :



https://reactjs.org/docs/lists-and-keys.html#\_\_\_gatsby

get array data from state react

**TODO Forms :**

**PureComponent**

20180413 :

<https://www.udemy.com/react-the-complete-guide-incl-redux/>

**Redux**

Redux is an open source JavaScript library, designed to allow for state management of JavaScript applications. Inspired by Elm, Redux is a debugging tool and supports robust application data-flow architecture. Redux is frequently used in combination with React.

TODO data-flow architecture : <https://redux.js.org/basics/data-flow>

Source 1-1 : <https://blog.risingstack.com/the-react-way-getting-started-tutorial/>

Used technologies:

* React
* ES6
* Webpack
* Babel
* Jest

Goal ?

* ES6 React
* virtual DOM
* Component-driven development
* Immutability
* Top-down rendering
* Rendering path and optimization
* Common tools/libs for bundling, ES6, request making, debugging, routing, etc.
* Isomorphic React

Source 1-2 : https://blog.risingstack.com/the-react-js-way-flux-architecture-with-immutable-js/