React Training

Follow training SOPs properly. Nothing should be missed. This part is where you need to show how much habits you have built around following best practices and writing good code and making architectural decisions. All of this will be evaluated.

First of all, close all old PRs on your project, merge them(after needed fixes etc) and do any other work you are assigned to clean-up in the project before React.

#### **First complete this ES6 tutorials (max 1 day):** <https://www.freecodecamp.org/learn/javascript-algorithms-and-data-structures/es6/>

#### **Now lets learn react(max 2 days, avg 1 day):**

Please use official docs to learn react because they are well written in a friendly manner

Here is the [link](https://reactjs.org/docs/hello-world.html) to the docs. Use the right sidebar to switch/navigate to different topics.

**Topics to cover:**

* Introduction
* Installation
* Main concepts(in the right side bar)
* Hooks

### **Learner should must know/be able to answer the following by the end:**

1. What is virtual dom?
2. Based on what, react re-renders a page?
3. Set of life cycle actions called when a state is updated?
4. Set of life cycle actions called when a prop is updated?
5. Why is a key prop necessary for a list rendering?
6. How to re-render a parent from a child?
7. How to pass data from parent to child?
8. What is meant by synchronous and asynchronous javascript (async, await, then etc)
9. Difference between classful and classless components?
10. Some good folder structures for managing components and API calls, css etc.

**Mini Project:**  
<https://react-builder-burger.firebaseapp.com/>  
Make a clone of the above app, for styling you can copy from the above url. This project will also be evaluated.

**React routing:**  
As introduction read [this article](https://www.educative.io/blog/react-router-tutorial)   
Further Details  
<https://www.digitalocean.com/community/tutorials/react-react-router>  
<https://www.digitalocean.com/community/tutorials/react-react-router-parameters>  
<https://www.digitalocean.com/community/tutorials/react-react-router-ssr>  
  
And Calling API  
<https://www.digitalocean.com/community/tutorials/react-axios-react>

**eslint** should be properly configured with some boilerplate configuration, like <https://github.com/standard/standard> (configure it as per your frontend stack/framework at the time).

**React pages from your test project:**  
In this part you need to build pages assigned from your rails test project in react, and data for these pages will be provided by your test project backend.

You will build API endpoints in your test project, and consume those to retrieve data and do required operations in the React app. Follow the instructions provided for this part.

**JS Basics (Cover these from the MERN Document link given below):**

[MERN Training Document](https://docs.google.com/document/d/1Fa-ZfW2U9umW7vJabYhWo3NpUKPZNnzsV7C9770_gyw/edit)

### **BONUS - REDUX** (only follow if advised by training department) (1day):

<https://www.freecodecamp.org/learn/front-end-libraries/redux/>  
<https://www.freecodecamp.org/learn/front-end-libraries/react-and-redux/>

* redux
* redux-thunk

### **Learner should must know/be able to answer the following by its end:**

1. What is the complete cycle of re-rendering a component using redux?
2. How a reducer gets called?
3. Everything that is achieved with redux, can it be achieved in react without using redux? If yes how, if not then what features do you think are unachievable.
4. Difference between Store and State.
5. Which lifecycle action is best fit for getting some initial data from api?