Scuola universitaria professionale della Svizzera italiana
Dipartimento tecnologie innovative

SUPSI

Master of Science in Engineering

Full-Stack Web Development with React Specialization (Hong Kong University)

Student

Olmo Barberis

Module

HK_Seminar

Data

July 28, 2021

The purpose of this document is to provide on overview of all the topics covered during the *Full-Stack Web Development with React Specialization* on *Coursera*[1]. The original course is composed of four parts, but the *HK_Seminar* will cover only two parts: an introduction to React, a front-end framework for developing single-page web application,and server-side development with NodeJS, Exrepss and MongoDB.

Front-End Web Development with React

This course provide an introduction to the React framework, its components and some basic elements like routing, forms and animations. Shared state management, with the corresponding tools like Redux, is another important topic covered alongside fetching data.

React Overview

React is a JavaScript library for building component-based user interfaces using a declarative approach. React focuses only on user interface and is designed for speed, simplicity and scalability.
A React component is a JavaScript class or function that is imported from the React Module. It is then rendered using the React's rendering function `ReactCOM.render(...)`.

Components

A component is an indipendent and reusable set of React elements that should appear on screen. Every component can accepts any input and is composed of React tags, that always start with a capital letter, and native tags, which start with lowercase letters and are treated as DOM tags. Every component has a local state which can hold multiple information that can be passed to children components using `props`. The state is an immutable object that can be updated using the `setState(...)` directive. This function accept the property to update and merge it with the actual state. Only class components can have a local state. The state must never be manipulated directly.
Handling event is possible in a similar way as on DOM elements:
In Figure 1 is possible to see how map a function to a click event on the React element `Card`.

Lifecycle React provides life cycle hooks/methods that can be invoked to perform certain operations. Component is created and then mounted in the application and when it's not required anymore is unmounted. There are three stages of lifecycle: mounting, updating and unmounting. Each stage provide, when component is declared as class, several methods like a construcor (mounting), `componentDidMount()` (called after mounting is finished), `render()` (call when rendering UI) and many others.

[1] https://www.coursera.org/specializations/full-stack-react

```
function onDishSelect() { ... };
<Card onClick={() => this.onDishSelect(dish)}></Card>
<Card onClick={this.onDishSelect}></Card>
```

Figure 1: Event handling when clicking on div

Functonal vs Class Until the release of `React 16.8` in 2018 there were two ways of declaring a component: class component or functional component. Functional components are s JavaScript function that returns a React element, can receive props but cannot provide local state or lifecycle hooks.
In React 16.8 it was introduced React Hooks to use lifecycle hooks also in functional component. This topic is not covered in courses, but will be discussed in the appendix of this document.

Document Object Model In the browser there is an object called `Browser DOM` (Document Object Model) and is the representation of the structure and data of a webpage. React uses a lightweight version of DOM called `Virtual DOM`, which uses an in-memory tree data structure of plain JavaScript objects, is extremely fast to manipulate compared to browser DOM and is fully recreated on every setState.
There is a diffing algorithm that detect which nodes are changed and updates the only the minimum number of components in the sub-tree that is updated.

React Router

React Router gives the ability to navigate between views using links. It's a module that need to be additionally installed into the React application called `react-router-dom`. It is a collection of navigational components that enable navigation among views and support a browser-based bookmarkable URLs to navigate in the web app. It is also possible to pass optional parameters.

```
const Menu = ({ dishes, onClick }) => {
  return (
    <div className="container">
            <div className="row">{menu}</div>

    </div>
  )
}

// or

function Menu({ dishes, onClick }) {
  return (
    <div className="container">
            <div className="row">{menu}</div>

    </div>
  )
}
```

Figure 2: Ways to define a functional components as function or arrow function