**Kamil Sabiron**

**28.01.2022**

**Sprawozdanie nr 05**

**Temat: React**

**Przebieg zadania:**

Część praktyczna (kod, screeny, opis działania funkcji, opis przebiegu zadania)

**Kod:**

**App.js**

import React from 'react';

import "./App.css";

export default class App extends React.Component{

state = {

loading: true,

Cars: []

};

async componentDidMount() {

const url = "<http://127.0.0.1:8000/modele/api/modeles/?format=json>";

const response = await fetch(url)

const data = await response.json();

this.setState({Cars: data, loading: false})

console.log(data[0]);

}

render(){

if(this.state.loading)

return <div>loading...</div>

if(!this.state.Cars.length)

return <div>Didn't get a car!</div>

const EveryCar = this.state.Cars.map(Car => (

<div id='box' key={Car.id} >

<div id='left'>

<div id='markaL'><label id='text'>Marka: </label></div>

<div id='modelL'><label id='text'>Model: </label></div>

<div id='rokL'><label id='text'>Rok: </label></div>

<div id='przebiegL'><label id='text'>Przebieg: </label></div>

<div id='pojemnoscL'><label id='text'>Pojemność Silnika: </label></div>

<div id='rodzajL'><label id='text'>Rodzaj Paliwa: </label></div>

<div id='opisL'><label id='text'>Opis: </label></div>

</div>

<div id='right'>

<div id='marka'>{Car.Marka}</div>

<div id='model'>{Car.Model}</div>

<div id='rok'>{Car.Rok}</div>

<div id='przebieg'>{Car.Przebieg} km</div>

<div id='pojemnosc'>{Car.Pojemnosc\_Silnika} cm3</div>

<div id='rodzaj'>{Car.Rodzaj\_Paliwa}</div>

<div id='opis'>{Car.Opis}</div>

</div>

<div id='right2'>

<img src={Car.Zdjecie} id='zdjecie' width="300px" />

</div>

</div>

));

return (

<div>{EveryCar}</div>

);

}

}

//http://127.0.0.1:8000/modele/api/modeles/?format=json

**App.css**

\*{

background: black;

color: whitesmoke;

}

#box{

background: #303030;

padding: 4rem;

font-family: sans-serif, Arial;

display: flex;

border: 3px solid black;

border-bottom: 1.5px solid black;

text-align: center;

border-radius: 4em;

flex-direction: row;

}

#text{

color: rgb(209, 24, 24);

font-weight: 600;

background: #303030;

}

img{

border: 3px solid white;

text-align: center;

vertical-align: middle;

}

#left{

display: block;

padding-right: 2em;

width: 15%;

background: #303030;

}

#right{

display: block;

text-align: left;

padding-left: 2rem;

padding-right: 3rem;

width: 55%;

background: #303030;

}

#right2{

width: 30%;

text-align: center;

vertical-align: middle;

background: #303030;

}

#marka, #markaL{

font-size: 3rem;

font-weight: 700;

background: #303030;

}

#model, #modelL{

font-size: 2rem;

font-weight: 700;

background: #303030;

}

#opis, #opisL{

padding-top: 10px;

background: #303030;

}

#pojemnosc, #pojemnoscL{

padding-top: 10px;

background: #303030;

}

#przebieg, #przebiegL{

padding-top: 10px;

background: #303030;

}

#rodzaj, #rodzajL{

padding-top: 10px;

background: #303030;

}

#rok ,#rokL{

padding-top: 10px;

background: #303030;

}

/\* Animation \*/

@keyframes start {

from {transform: translateX(100%)}

to {transform: translateX(0%)}

}

div#box {

animation-name: start;

animation-duration: 3s;

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Opis działania funkcji:**



Nadajemy tutaj stan zmmiennych



Nadajemy w stałych wartości url - link do API, response odbieramy link, date odbieramy w wersji json. Dalej ustalamy stan zmiennych i poniżej Konsol log w celach sprawdzenia drogi do poszczegolnych akcji itp.



W funkcji render sprawdzamy przez stany czy strona sie ładuje, jeśli tak to bedzie widnieć napis "loading...", a po załadowaniu strony wyświetli się norlmalnie strona. Sprawdzamy też czy mamy jakieś dane jeśli tak to przechodzimy dalej, a jak nie to wyświetli sie komunikat "Didn't get a car!" (jeśli jest pusta baza to wtedy sie wyświetli).







Ten const ma za zadanie pobrać w 1 dużym bloku 2 mniejsze podzielone na zwykle label'y (lewa strona) i dane wyciagniete z API (prawa strona). Wyświetla całą tablice a nie pojedyńcze elementy. Jeśli dodamy kolejny obiekt to zaktualizuje sie to automatycznie na stronie i bedzie on tam odrazu widnieć.



Wyświetlnie wcześniej omówionego const'a

**Opis przebiegu zadania:**

Zadanie zacząłem od stworzenia projektu, następnie obejrzałem dużu filmików dotyczące React aby zrozumieć z czym mam doczynienia. Nareszcze przystapiłem do pracy. Napotkałem duży problem dotyczący połaczenia mojego API z React'em. Zmarnował on mi naprawdę dużo czasu bo początkowo myślałem, że to ja coś źle robie i postanowiłem wyciagnąc dane z jakiegoś publicznego API i odziwo sie udało. Więc szukałem problemu i dostarłem do CORS które było moim całym problemem. Probowałem sam poradzić sobie z tym problemem ale nie wychodziło. Poprosiłem o pomoc kolege ale ten też zbytnio nie mial pomysłów jak to naprawić, a jak miał okazywały się one błednę. W końcu zdecydowałem się napisać do opiekuna naszej grupy o pomoc i dzięki niemu mogłem prześć dalej. Gdy w końcu udało sie mi wyświetlić jeden obiekt, zacząłem próbować odrazu całą tablice wyświetlić. Po dluższym czasie z dokumentacja i forami udało sie dokonać tego. Następnym krokiem była modyfikacja API w taki sposyb aby lepiej i wiecej informacji mógł wyświetlić np. zdjęcie. W miedzy czasie zrobiłem prostego css, ale dopiero teraz tak naprawde się za niego zabrałem. Zrobiłem prosta animacje wyjezdzania obiektów i strone bardziej przejrzystą. Pod koniec przeczytałem o router'ach i probowałem je zainplementować w swoim projekcie ale zbytnio to już nie wychodziło (prawdopodbnie przez zmęczenie).