**ОТЧЕТ**

по лабораторной работе №3

по курсу «Разработка web-приложений»

на тему «React hooks»

Выполнили:

студенты группы 17ВВ2

Живаев В.В.

Приняла:

Карамышева Н.С.

Пенза 2021

Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

### Цель работы: изучение способов работы с хуками библиотеки ReactJS.

**Задание**: В исходный проект необходимо добавить функцию поиска, т.е. необходимо добавить поле ввода, при изменении данных в котором будет выполняться поиск по элементам массива (принцип его работы основывается на поиске первого вхождения подстроки в строке). Результат поиска должен заменять отображение начального списка на найденный(-ые) элемент(-ы) из этого списка (пока строка пустая, отображается начальный вариант списка).

**Ход работы**:

Изучили виды react-хуков (useState, useEffect) и их применение.

Написали тестовые функциональные компоненты, в которых ознакомились с возможностями хуков, например, тестовый компонент UseStateHookExample, возвращающий контейнер с кнопкой, имеющей внутреннее состояние:

import React, { useState } from 'react';  
 export default function UseStateHookExample() {  
 // Объявляем новую переменную состояния "count"  
 const [count, setCount] = useState(0);  
 return (  
 <div>  
 <p>Вы нажали {count} раз</p>  
 <button onClick={() => setCount(count + 1)}>

Нажми на меня  
 </button>  
 </div>  
 );  
 }

Листинги файлов с JSX компонентами по заданию представлен ниже.

### Листинг:

#### CustomIterable.js

import React from "react";

import ListElement from "./CustomElement";

const ListComponent = (props) => {

let container = [];

for (let val of props.array) {

container.push(<ListElement value={val}/>);

}

return <div align={"center"}>{container}</div>;

}

export default ListComponent;

#### CustomElement.js

import React from "react";

const ListElement = (props) => {

return (

<li id={props.value.id}

style={{

backgroundColor: "lightgray",

borderRadius: 3,

margin: 4,

maxWidth:100

}}>

{props.value.value}

</li>

)

}

export default ListElement;

#### App.js

import React, {useEffect, useState} from "react";

import './App.css';

import './CustomIterable'

import ListComponent from "./CustomIterable";

function App() {

const arr = [

{id: "1", value: "Element 1"},

{id: "2", value: "Element 2"},

{id: "3", value: "Element 3"}

];

return (

<div className="App">

<ListComponent array={arr}/>

<Search arr={arr}/>

</div>

);

}

const Search = ({arr}) => {

const [currentValue, setCurrentValue] = useState({filtered: [], input: ""});

useEffect(() => {

if (currentValue.filtered.length !== 0) {

document.title = `Найдено ${currentValue.filtered.length} элементов`;

} else {

document.title = `UseEffect React hook example`;

}

})

return (

<>

<input type="text" value={currentValue.input} onChange={

(e) =>

setCurrentValue(() => {

let input = e.target.value;

let proceed = input ? arr.filter(el => el.value.includes(input)).map(item => item.value)

: arr.map(item => item.value);

return {filtered: proceed, input: input};

})

}/>

<div>

{

currentValue.filtered

}

</div>

</>

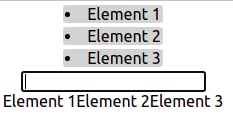
)

};

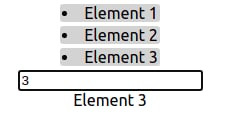
export default App;

### Результаты выполнения работы:

На рисунках ниже продемонстрированы результаты работы:



**Рисунок 1 – Результат поиска при пустой строке.**



**Рисунок 2 – Результат поиска с вхождением элемента.**



**Рисунок 3 – Измененный title документа при пустой поисковой выдаче.**



**Рисунок 4 – Измененный title документа при вхождении 3 элементов.**

Для отображения списка, созданного по заданию был расширен существующий функциональный компонент из App.js и дописаны простые стили внутри компонента JSX.

### Выводы

### В ходе данной лабораторной работы были получены знания о способах работы с хуками библиотеки ReactJS.