

Министерство образования и науки РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования «Ивановский государственный энергетический
университет имени В.И. Ленина»

Кафедра программного обеспечения компьютерных систем

Отчёт по лабораторной работе №7

Конструирование интернет-приложений

Использование библиотек компонентов

Выполнила студент гр. 3-42 Шарабанов Н.А.

Проверил Садыков А.М.

ИВАНОВО 2023

Цель лабораторной работы: научиться использовать библиотеки компонентов

Задания:

1. Добавить пакет Ant Design
2. Выбрать основную верстку
3. Реализовать верстку в компоненте Layout
4. Добавить компонент Form
5. Добавить модальное окно

Вывод

Ход Работы

1. Добавлен пакет Ant Design

Был добавлен пакет Ant Design с помощью команды `npm i -D antd`. Результат представлен на рисунке 1.

```
"name": "f-internet-shop-web-app",
"version": "0.1.0",
"private": true,
"dependencies": {
  "@testing-library/jest-dom": "^5.16.5",
  "@testing-library/react": "^13.4.0",
  "@testing-library/user-event": "^13.5.0",
  "antd": "^5.4.2",
  "react": "^18.2.0",
  "react-dom": "^18.2.0",
  "react-scripts": "5.0.1",
  "web-vitals": "^2.1.4"
```

Рис.1 – Добавление Ant Design

2. Выбрать основную верстку

Выбран подходящий для моей темы layout с сайта <https://ant.design/components/layout>. Он будет иметь следующий вид:

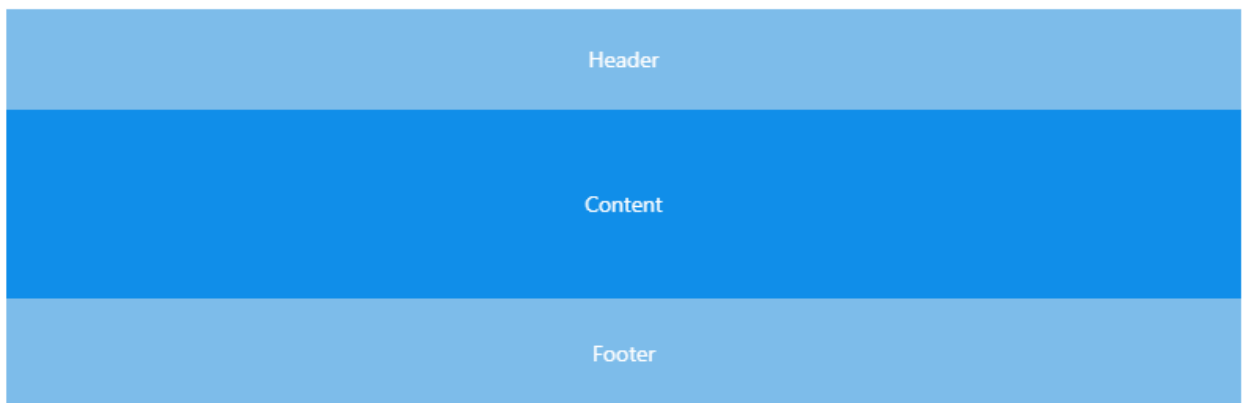


Рис.2 – Выбор Layout

3. Реализовать верстку в компоненте Layout

```
import React from 'react';
import {Outlet, Link} from 'react-router-dom';
import {Layout as LayoutAntd, Menu} from 'antd';
const {Header, Content, Footer} = LayoutAntd;
const items = [
  {
    label: <Link to={'/'}>Главная</Link>,
    key: '1',
  },
  {
    label: <Link to={'/OrderItems'}>Строки заказа</Link>,

```

```

    key: '2',
  },
  {
    label: <Link to={"/login"}>Вход</Link>,
    key: '3',
  },
  {
    label: <Link to={"/logout"}>Выход</Link>,
    key: '4',
  },
  {
    label: <Link to={"/register"}>Регистрация</Link>,
    key: '5',
  },
];
const Layout = ({user}) => {
  return (
    <LayoutAntd>
      <Header style={{position: 'sticky', top: 0, zIndex: 1, width: '100%'}}>
        <div
          style={{
            float: 'right',
            color: 'rgba(255, 255, 255, 0.65)',
          }}
        >
          {user.isAuthenticated ? (
            <strong>{user.userName}</strong>
          ) : (
            <strong>Гость</strong>
          )}
        </div>
        <Menu theme="dark" mode="horizontal" items={items} className="menu" />
      </Header>
      <Content className="site-layout" style={{padding: '0 50px'}}>
        <Outlet />
      </Content>
      <Footer style={{textAlign: 'center'}}>Blogging ©2023</Footer>
    </LayoutAntd>
  );
};
export default Layout;

```



Рис.3 – Результат работы приложения

4. Добавить компонент Form

Для того, чтобы добавить компонент Form необходимо внести изменения в компонент Login и Register:

```
/* eslint-disable max-len */
```

```

import React, {useState} from 'react';
import {useNavigate} from 'react-router-dom';
import {Button, Checkbox, Form, Input} from 'antd';
const LogIn = ({user, setUser}) => {
  const [errorMessages, setErrorMessages] = useState([]);
  const navigate = useNavigate();
  const logIn = async (formValues) => {
    console.log('Success:', formValues);
    const requestOptions = {
      method: 'POST',
      headers: {'Content-Type': 'application/json'},
      body: JSON.stringify({
        email: formValues.username,
        password: formValues.password,
        rememberme: formValues.remember,
      }),
    };
    return await fetch('api/account/login', requestOptions)
      .then((response) => {
        // console.log(response.status)
        response.status === 200 &&
        setUser({isAuthenticated: true, userName: ''});
        return response.json();
      })
      .then(
        (data) => {
          console.log('Data:', data);
          if (
            typeof data !== 'undefined' &&
            typeof data.userName !== 'undefined'
          ) {
            setUser({isAuthenticated: true, userName: data.userName, userRole: data.userRole});
            navigate('/');
          }
          typeof data !== 'undefined' &&
          typeof data.error !== 'undefined' &&
          typeof data.userRole !== 'undefined';
          setErrorMessages(data.error);
        },
        (error) => {
          console.log(error);
        },
      );
  };
  const renderErrorMessage = () =>
    errorMessages.map((error, index) => <div key={index}>{error}</div>);
  return (
    <>
      {user.isAuthenticated ? (
        <h3>Пользователь {user.userName} с ролью {user.userRole} успешно вошел в систему</h3>
      ) : (
        <>
          <h3>Вход</h3>
          <Form

```

```

    onFinish={logIn}
    name="basic"
    labelCol={{span: 8}}
    wrapperCol={{span: 16}}
    style={{maxWidth: 600}}
    initialValues={{remember: true}}
    onFinishFailed={renderErrorMessage}
    autoComplete="off"
  >
    <Form.Item
      label="Логин"
      name="username"
      rules={[
        {required: true, message: 'Please input your username!'},
      ]}
    >
      <Input />
    </Form.Item>
    <Form.Item
      label="Пароль"
      name="password"
      rules={[
        {required: true, message: 'Please input your password!'},
      ]}
    >
      <Input.Password />
    </Form.Item>
    <Form.Item
      name="remember"
      valuePropName="checked"
      wrapperCol={{offset: 8, span: 16}}
    >
      <Checkbox>Remember me</Checkbox>
      {renderErrorMessage()}
    </Form.Item>

    <Form.Item wrapperCol={{offset: 8, span: 16}}>
      <Button type="primary" htmlType="submit">
Отправить
      </Button>
    </Form.Item>
  </Form>
</>
)}
</>
);
};
export default LogIn;

```

Аналогичные изменения необходимо внести в компонента Register.

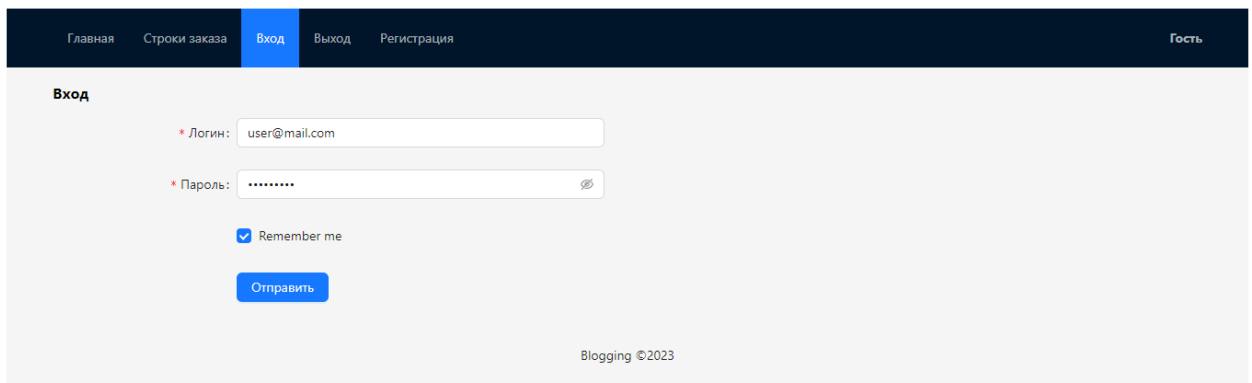


Рис.4 – Результат изменения компонента Login

5. Добавить модальное окно

```
import React, {useEffect, useState} from 'react';
import {useNavigate} from 'react-router-dom';
import {Modal} from 'antd';
const LogOff = ({setUser}) => {
  const [open, setOpen] = useState(false);
  const navigate = useNavigate();
  const showModal = () => {
    setOpen(true);
  };
  useEffect(() => {
    showModal();
  }, []);
  const logOff = async (event) => {
    event.preventDefault();
    const requestOptions = {
      method: 'POST',
    };
    return await fetch('api/account/logoff', requestOptions).then(
      (response) => {
        response.status === 200 &&
        setUser({isAuthenticated: false, userName: ''});
        response.status === 401 ? navigate('/login') : navigate('/');
        setOpen(false);
      },
    );
  };
  const handleCancel = () => {
    console.log('Clicked cancel button');
    setOpen(false);
    navigate('/');
  };
  return (
    <>
      <Modal title="Title" open={open} onOk={logOff} onCancel={handleCancel}>
        <p>Выполнить выход?</p>
      </Modal>
    </>
  );
};
```

export default LogOff;

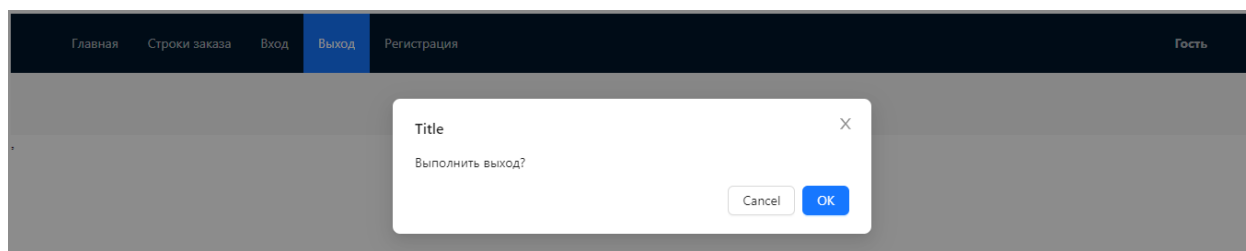


Рис.5 – Результат изменения компонента Logoff

Вывод

В ходе лабораторной работы я научился использовать библиотеки компонентов.