

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МО ЭВМ**

**ОТЧЕТ**  
**по лабораторной работе №6**  
**по дисциплине «Веб-технологии»**  
**Тема: Модуль приложения “Покупка и продажа акций”**

Студент гр. 8304

\_\_\_\_\_

Сергеев А. Д.

Преподаватель

\_\_\_\_\_

Беляев С. А.

Санкт-Петербург

2020

## **Цель работы**

Изучение возможностей применения библиотеки React (<https://reactjs.org/>) для разработки интерфейсов пользователя web-приложений.

## **Основные теоретические сведения**

React – библиотека на JavaScript для построения интерфейса пользователя. React представляет удобным инструментом для создания масштабируемых web-приложений, особенно в тех ситуациях, когда приложение является одностраничным.

В основу React заложены принципы Redux, предлагающее предсказуемый контейнер хранения состояния web-приложения.

Вся структура веб-страницы может быть представлена с помощью DOM. Для решения проблемы производительности предложена концепция виртуального DOM, который представляет собой облегченную версию DOM. React работает именно с виртуальным DOM. Реализован механизм, который периодически сравнивает виртуальный DOM с реальным и вычисляет минимальный набор манипуляций для приведения реального DOM к состоянию, которое хранится в виртуальном DOM.

## **Общая формулировка задачи**

Необходимо создать web-приложение, обеспечивающее работу брокера, у него есть запас денежных средств, он имеет возможность купить или продать акции (любое доступное количество), а также контролировать изменение котировок акций. Брокеру должен отображаться баланс (запас денежных средств плюс стоимость акций), с которым он начал день, и текущее состояние.

### Основные требования:

1. Приложение получает исходные данные из модуля администрирования приложения «Биржа акций» в виде настроек в формате JSON-файла.
2. В качестве сервера используется Node.JS с модулем express (либо nginx в связке с PHP в качестве «back-end»).
3. Участники торгов подключаются к приложению «Покупка и продажа акций».
4. Предусмотрена HTML-страница администратора, на которой отображается перечень участников, для каждого участника отображается его баланс, количество акций каждого типа у каждого участника и количество, выставленное на торги.
5. Предусмотрена HTML-страница входа в приложение, где каждый участник указывает (или выбирает из допустимых) своё имя.
6. Предусмотрена HTML-страница, на которой участнику отображается общее количество доступных ему средств, количество и суммарная стоимость по каждому виду купленных акций. На ней же отображается количество выставленных на торги акций, их количество и стоимость. У участника есть возможность купить/продать интересующее его количество акций.

Участнику отображается суммарный доход на начало торгов и на текущий момент времени.

### Ход работы

1. Используя среду разработки JetBrains WebStorm, были установлены все необходимые расширения.
2. Используя модуль express, был создан и настроен сервер.
3. При помощи React было создано одностраничное приложение.

4. Для упрощения создания пользовательского интерфейса была использована библиотека bootstrap.
5. Разработка интерфейса пользователя:

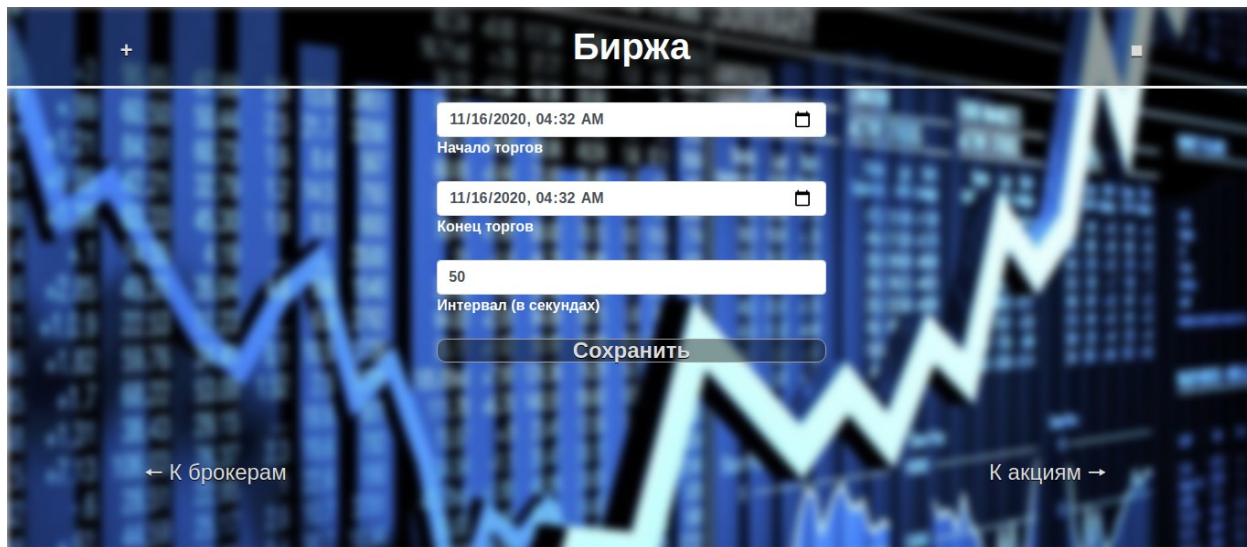


Рисунок 1 – Главная страница сайта.

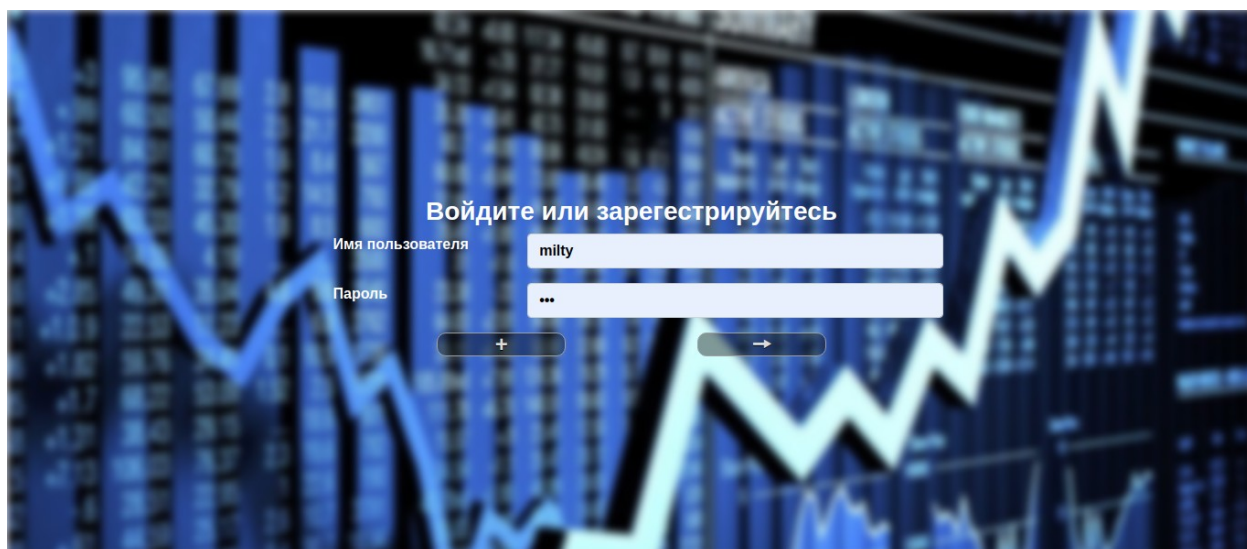


Рисунок 2 – Страница входа на сайт.

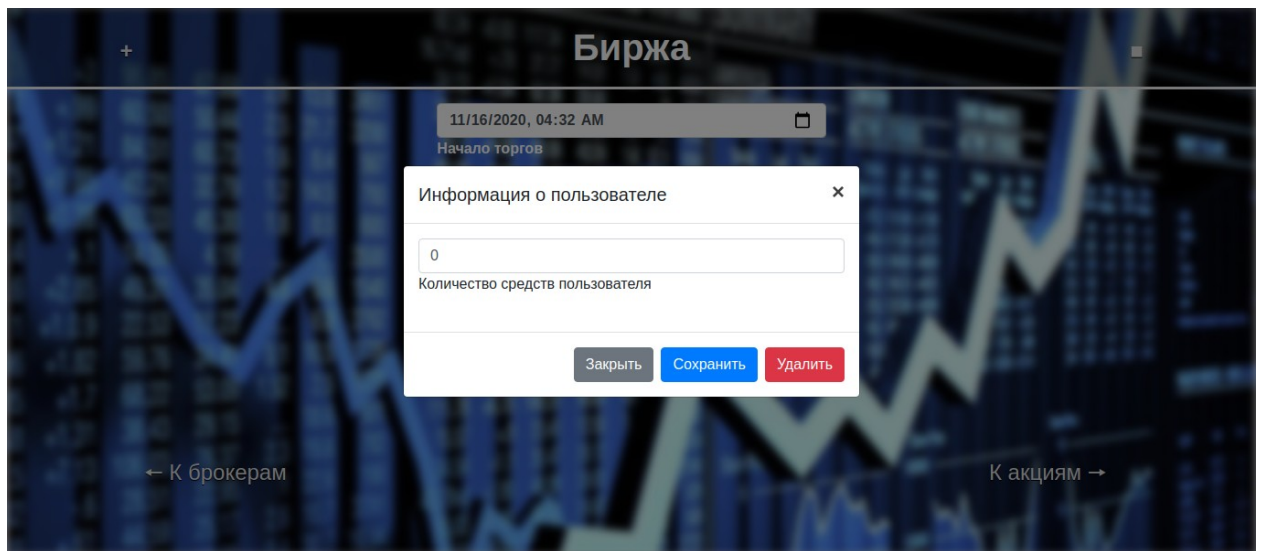


Рисунок 3 – Модальное окно текущего пользователя.

Имя	Баланс	Роль
Peter	390	admin
Tom	510	user
milty	0	admin
otter	0	user

Рисунок 4 — Страница информации о пользователях.



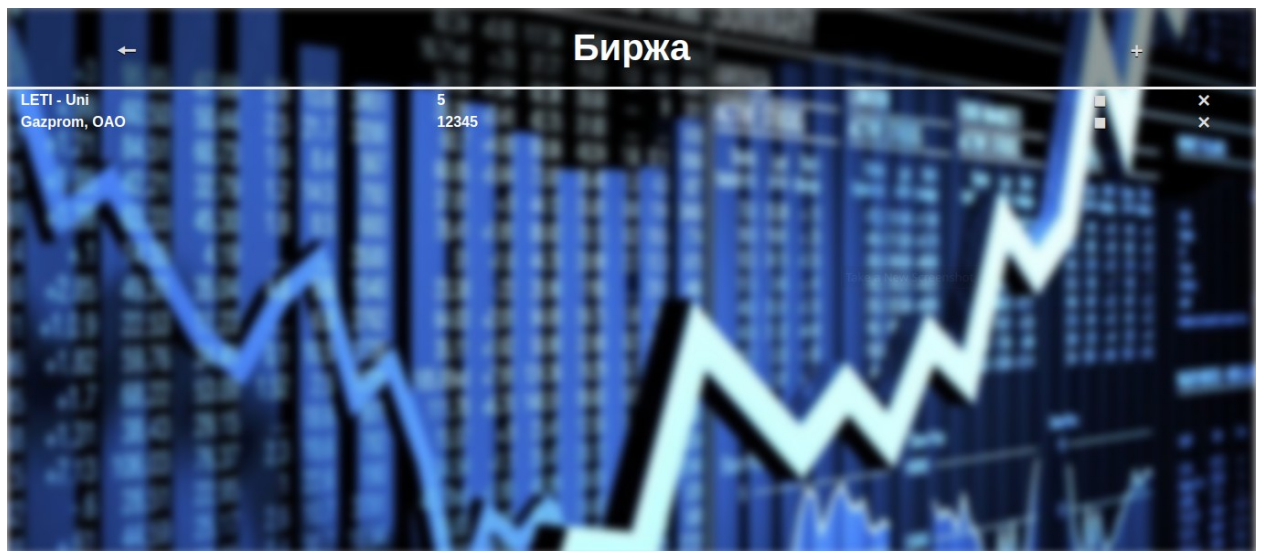


Рисунок 5 — Страница информации об акциях.

Рисунок 6 — Модальное окно добавления акции.

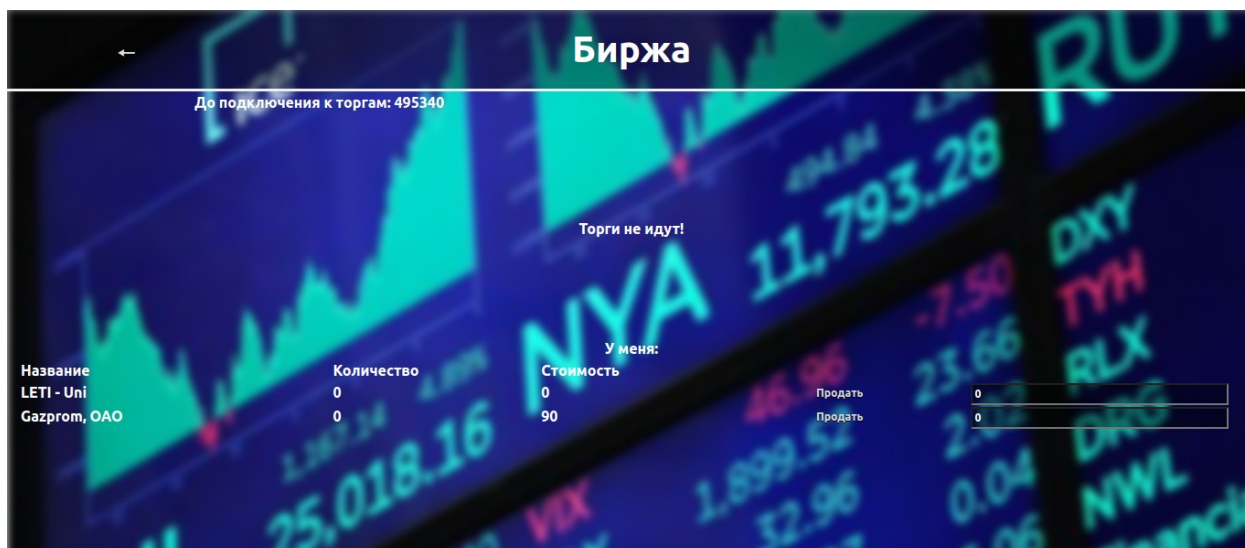


Рисунок 7 — Страница торгов.

6. Для проекта были созданы следующие файлы.

- 1) Была создана компонента `brokers`, для отображения интерфейса брокеров.
- 2) Была создана компонента `actions`, для отображения информации об акциях.
- 3) Была создана компонента `fzf`, для отображения не найденной страницы.
- 4) Была создана компонента `login`, для отображения страницы входа на сайт.
- 5) Была создана компонента `main`, для отображения главной страницы сайта.
- 6) Реализована маршрутизация при помощи изменения файла `app.component.ts`.
- 7) `brokers.component.ts`, `actions.component.ts`, `fzf.component.ts`, `login.component.ts`, `main.component.ts` – файлы представляющие собой интерфейс компонентов.
- 8) Создан сервис `provider.service.ts` для обмена информацией с сервером.

- 9) App.js – файл, содержащий в себе функции по работе с сервером и сокетами.
- 10) MoneySpan.js — компонент, отображающий количество денег пользователя до и после торгов.
- 11) index.js — базовый компонент react-приложения.
- 12) ActionSpace.js — компонент, отображающий акции на торгах и принадлежащие пользователю.
- 13) Socketizer.js — сервис, осуществляющий обмен информацией с сервером при помощи сокетов.

## **Вывод**

В ходе лабораторной работы был получен опыт работы с React на основе создания модуля приложения “Покупка и продажа акций”.