|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К КУРСОВОЙ РАБОТЕ***

***НА ТЕМУ:***

***\_\_\_\_\_\_\_\_\_\_\_\_\_Трейдерское приложение\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

Студент \_\_\_\_\_ИУ5-65Б\_\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_М.Д. Гурова\_\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Студент \_\_\_\_\_ИУ5-65Б\_\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_А.С. Ищенко\_\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Студент \_\_\_\_\_ИУ5-65Б\_\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_Е.В. Юриков\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель курсовой работы **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_В.А. Галкин\_\_**

(Подпись, дата) (И.О.Фамилия)

Консультант **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О.Фамилия)

*2023 г.*

**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технический университет имени Н.Э. Баумана**

**(национальный исследовательский университет)»**

**(МГТУ им. Н.Э. Баумана)**

УТВЕРЖДАЮ

Заведующий кафедрой \_\_\_\_\_\_\_\_\_\_

(Индекс)

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

(И.О.Фамилия)

« \_\_\_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_ 20 \_\_\_\_ г.

**ЗАДАНИЕ**

**на выполнение курсовой работы**

по дисциплине \_\_\_\_\_\_\_\_\_Сетевые технологии в АСОИУ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Студенты группы \_\_\_\_ИУ5-65Б\_\_\_\_\_\_

\_\_\_\_Гурова Мария Дмитриевна, Ищенко Анастасия Сергеевна, Юриков Егор Владимирович\_

(Фамилия, имя, отчество)

Тема курсовой работы \_\_\_\_\_\_\_\_\_\_\_\_\_Трейдерское приложение\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

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

Направленность КР (учебная, исследовательская, практическая, производственная, др.)

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

Источник тематики (кафедра, предприятие, НИР) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

График выполнения работы: 25% к \_\_\_ нед., 50% к \_\_\_ нед., 75% к \_\_ нед., 100% к \_\_\_ нед.

***Задание*** \_\_\_\_\_\_Разработать распределённую систему для биржевой торговли\_\_\_\_\_\_\_\_\_\_\_\_\_

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

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

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

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

***Оформление курсовой работы:***

Расчетно-пояснительная записка на \_60\_\_ листах формата А4.

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

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

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

Дата выдачи задания « \_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.

**Руководитель курсовой работы**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **\_\_\_\_\_\_В.А. Галкин\_\_**

(Подпись, дата) (И.О.Фамилия)

**Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_М.Д. Гурова\_\_\_\_**

(Подпись, дата) (И.О.Фамилия)

**Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_А.С. Ищенко\_\_\_**

(Подпись, дата) (И.О.Фамилия)

**Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_Е.В. Юриков**\_\_

(Подпись, дата) (И.О.Фамилия)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

Оглавление

[Введение 4](#_gjdgxs)

[Назначение программного продукта 6](#_30j0zll)

[Архитектура программного продукта 8](#_1fob9te)

[Сервис бэкенда 10](#_3znysh7)

[Фронтенд 13](#_2et92p0)

[Сервис WebSocket 20](#_tyjcwt)

[Заключение 22](#_3dy6vkm)

[Список использованных источников 23](#_1t3h5sf)

[Приложения 24](#_4d34og8)

# Введение

Основной целью курсовой работы являлось создание распределённой системы на примере трейдерского приложения. Компоненты распределённой системы взаимодействуют между собой различными способами: при помощи HTTP запросов, websocket и gRPC.

Распределённая система “Трейдерское приложение” предназначена для обеспечения работы веб приложения для трейдинга.

В функционал данного приложения входит:

* Предоставление информации о ценных бумагах и их актуальных котировках с визуализацией в виде графика для всех посетителей сайта.
* Авторизация на сайте. Посетитель может авторизоваться в роли простого пользователя или менеджера. В зависимости от роли, авторизованному будет доступен разный функционал.
* Возможность покупки акций, просмотр истории операций и списка имеющихся акций для авторизованных пользователей сайта
* Возможность управления данными о ценных бумагах для менеджера. Подлежат изменению следующая информация: название ценной бумаги, краткое название, ее стоимость и описание. Также для менеджера открыт интерфейс создания новых акций и удаления существующих.

Актуальность котировок обеспечивается использованием протокола WebSocket, при помощи которого при изменении котировки на сервере, данные автоматически обновляются у пользователей по установленному соединению.

Данная работа была разбита на несколько составляющих частей:

* Фронтенд — реализовать приложение на React + Redux Toolkit + Axios + MUI. Необходимо реализовать окно регистрации и авторизации. Приложение должно общаться с веб-сервису с данными.
* Бэкенд — реализовать веб-сервис (API gateway), который будет предоставлять методы для фронтенда. Веб-сервис взаимодействует с базой данных. Необходимо ограничить доступ к методам сервиса через авторизацию.
* Интеграционная задача — реализовать протокол прикладного уровня для передачи по WebSocket котировок акций. Сервис ws взаимодействует с бэкендом по gRPC для получения/изменения данных. Гарантированная доставка сообщений при повторном подключении после разрыва соединения.

# Назначение программного продукта

Роли и функции пользователей программного продукта отражены на диаграмме прецедентов:

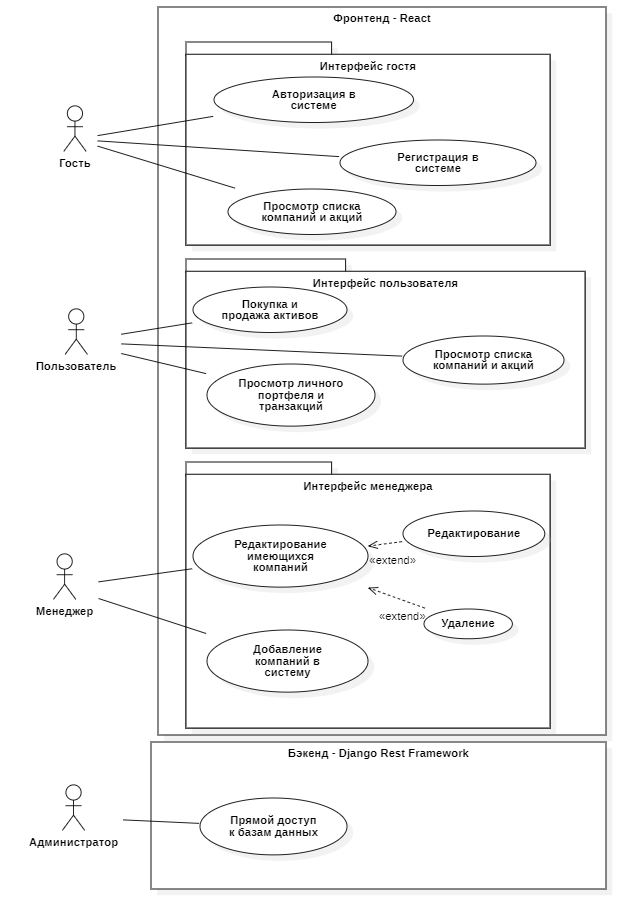


Рисунок 1 - Диаграмма прецедентов

Распределённая система:

1. Обеспечивает авторизацию и регистрацию пользователей;
2. Обеспечивает разделение пользователей на менеджеров, клиентов, посетителей и администраторов;
3. Обеспечивает просмотр акций компаний с графиками котировок авторизованными пользователями, гостями и менеджерами;
4. Обеспечивает интерфейс администраторов для доступа к базе данных;
5. Обеспечивает возможность менеджерами добавления новых компаний в систему;
6. Обеспечивает возможность покупки и продажи акций на аккаунт авторизованного пользователя;
7. Обеспечивает возможность просмотра портфеля пользователя, а также его списка транзакций.

# Архитектура программного продукта

Архитектура программного продукта представлена на диаграмме развёртывания.

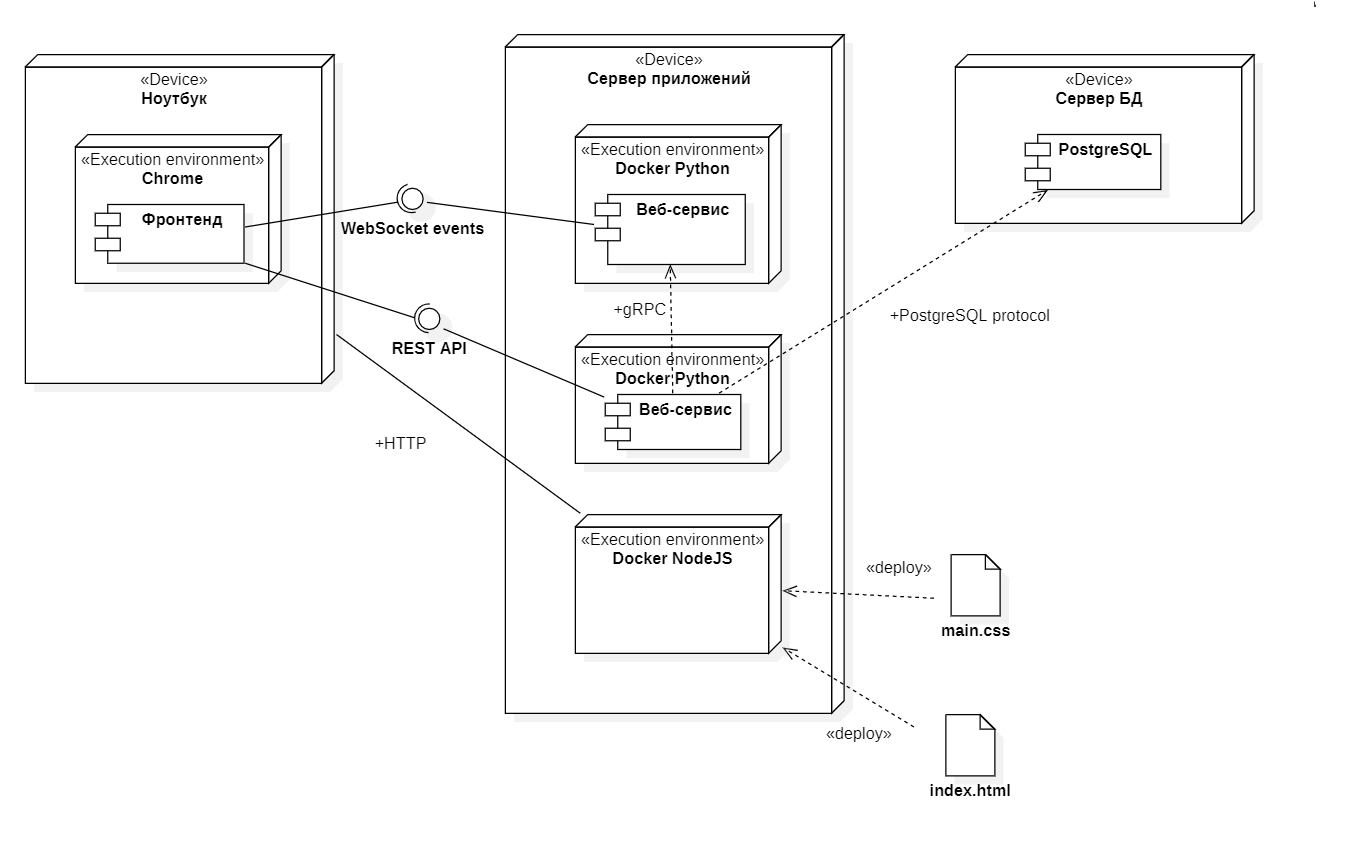


Рисунок 2 - Диаграмма развёртывания

Распределённая система состоит из бэкенда, фронтенда и сервиса Websocket. Пользователь взаимодействует с программой при помощи пользовательского интерфейса. С его помощью пользователь имеет ограниченный доступ к базе данных через бэкенд приложения. Взаимодействие бэкенда и фронтенда происходит при помощи AJAX запросов.

Для обновления котировок акций в реальном времени используется протокол Websocket. Для подключения клиентов ws разработан отдельный сервис. Взаимодействие фронтенда и сервиса ws происходит по протоколу Websocket. Для передачи новой котировки, полученной с бэкенда, бэкенд и сервис ws взаимодействуют по gRPC.

# Сервис бэкенда

Сервис бэкенда выполнен на языке Python с использованием Django Rest Framework. Взаимодействие с фронтендом происходит по Rest API, взаимодействие с сервисом Websocket происходит по gRPC. Сервис бэкенда используется для удалённого вызова процедур для обращения к базе данных.

Бэкенд взаимодействует с сервером базы данных. В качестве СУБД используется PostrgeSQL. Схема базы данных представлена на ER диаграмме.

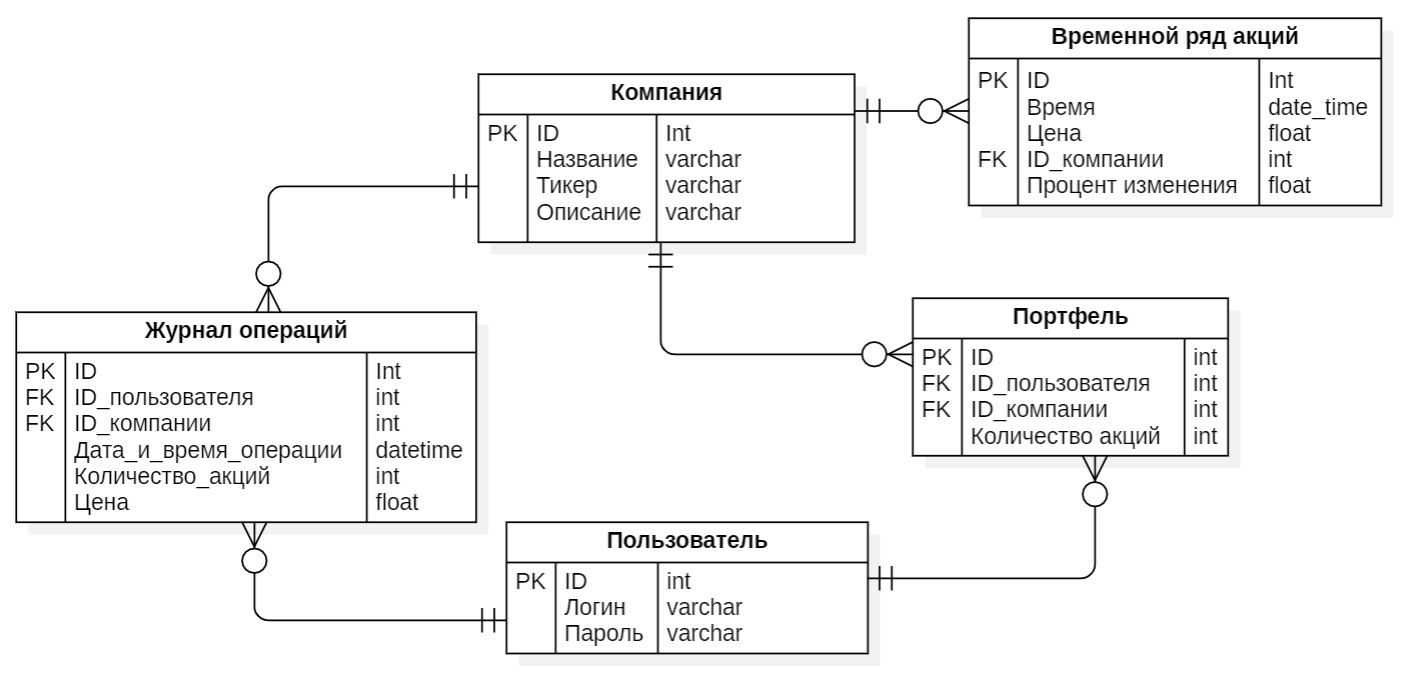


Рисунок 3 - ER диаграмма

База данных позволяет хранить всю необходимую информацию:

* логин и пароль пользователя;
* информацию о компаниях, представленных на бирже;
* журнал операций пользователя со временем совершения сделки, количеством акций и ценой на момент совершения сделки;
* портфели пользователей;
* временной ряд акций с ценой, временем и процентом изменения цены за каждый временной отрезок.

Бэкенд предоставляет фронтенду данные из базы данных в виде JSON, а также может делать новые записи в базу. Сервис предоставляет методы для:

* регистрации: пользователю необходимо указать логин и пароль, переданные логин и пароль сохраняются в базу данных;
* авторизации: пользователю необходимо указать логин и пароль, в ответ пользователю будут отправлены access и refresh токены;
* управления списком компаний: любой пользователь может получить список или одну компанию. Изменить, удалить или добавить компанию может только менеджер;
* покупки и продажи активов: пользователю необходимо указать id компании, и количество акций, так как метод доступен только авторизованному пользователю, id пользователя будет взят из JWT токена, соответствующая запись сохранится в базе данных, также будет добавлена новая операция в соответствующую таблицу;
* просмотра портфеля пользователя: пользователю необходимо отправить пустой запрос на соответствующий URL, так как метод доступен только авторизованному пользователю, id пользователя будет взят из JWT токена, в ответ пользователю будет отправлен его портфель;
* просмотра списка операций пользователя: пользователю необходимо отправить пустой запрос на соответствующий URL, так как метод доступен только авторизованному пользователю, id пользователя будет взят из JWT токена, в ответ пользователю будет отправлен его список операций;
* получения временного ряда списка компаний: пользователю необходимо отправить пустой запрос на соответствующий URL, в ответ пользователю будет отправлен временной ряд акций одной компании.

Доступные методы показаны на диаграмме последовательности:

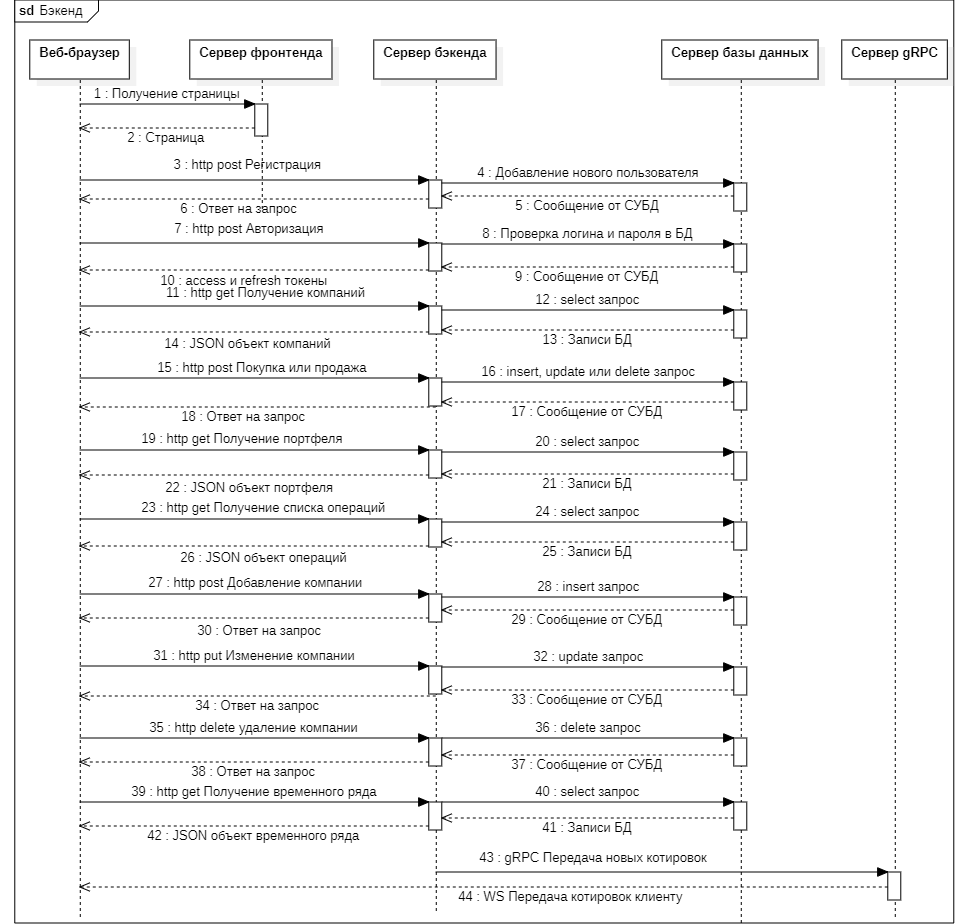


Рисунок 4 - Диаграмма последовательности бэкенда

Авторизация происходит по JWT, помимо служебной информации, в токене также хранится id пользователя, которому принадлежит токен, и является ли пользователь менеджером.

Сервис развёрнут в docker контейнере.

Сервис бэкенда также обращается к методу сервиса Websocket для передачи новых котировок акций по gRPC. Бэкенд генерирует новую цену акций для каждой компании, хранящейся в системе, сохраняет их в базу данных и передаёт сервису Websocket.

# Фронтенд

Макет приложения был создан в онлайн-редакторе для проектирования интерфейсов Figma. Все элементы, кроме графика цены акции, использованные в макете являются компонентами библиотеки MUI. MUI предоставляет простую, настраиваемую и доступную библиотеку компонентов React. MUI предлагает полный набор инструментов пользовательского интерфейса, которые помогают быстрее внедрять новые функции.

График цен ценной бумаги явяется компонентом библиотеки React Stockcharts. Он поддерживает уменьшение и увеличение масштаба, а также обеспечивает наилучшую визуализацию динамики цен, что делает пользовательский опыт лучше.

Фронтенд выполнен на языке JavaScript. При его разработке учитывались рекомендации по простоте, удобству и функциональности интерфейса. При разработке приложения использовалась библиотека React — библиотека JavaScript с открытым кодом для создания внешних пользовательских интерфейсов. В отличие от других библиотек JavaScript, предоставляющих полноценную платформу приложений, React ориентируется исключительно на создание представлений приложений через инкапсулированные единицы (компоненты), которые сохраняют состояние и генерируют элементы пользовательского интерфейса. Для сборки и локальной разработки использовалась утилита vite - интерфейса командной строки (command line interface, CLI), который позволяет создавать предварительно настроенные React-проекты. Предварительная настройка включает в себя, но не исчерпывается, настройкой esbuild.

Так как JavaScript не является типизированным языком, в проекте также использовался язык программирования TypeScript. Преимущества использования TypeScript состоят в том, что строгая типизация значительно сокращает количество возможных ошибок, удобно интегрируется с IDE, расширяет возможности JavaScript в ООП (TypeScript позволяет использовать наследование, абстрактные классы и многое другое), компилятор TS позволяет преобразовывать код в JS-код более старых стандартов языка, что повышает поддержку браузерами, а также повышает наглядность и документируемость кода.

Для управления состоянием в проекте используется библиотека redux-toolkit.

Для отправки запросов на бэкенд используется библиотека Axios. Axios — это облегченный клиент HTTP на базе сервиса $http с Angular.js, похожего на собственный JavaScript Fetch API. Благодаря использованию данной библиотеки, интерфейс взаимодействия между бэкендом и фронтендом значительно упрощается.

Для обеспечения актуальности стоимости акций используется протокол WebSocket, который поддерживает непрерывное соединение между сервером и клиентом, передавая новые данные при их обновлении в БД клиенту. Для навигации используется библиотека react-router-dom.

Взаимодействие пользователя с приложением представлено на диаграмме деятельности (Приложение 5 Диаграмма деятельности).

При запуске приложения посетитель попадает на главную страницу. До авторизации он считается гостем и может только просматривать информацию об акциях. Гость может авторизоваться как обычный пользователь или как менеджер, введя свои логин и пароль в соответствующих полях. Далее данные отправляются на бэкенд для проверки корректности. В случае верной авторизации, пользователю возвращаются access и refresh токены и статус менеджера, записывающиеся в локальное хранилище. Авторизованному пользователю доступен список акций, также ему открывается доступ к покупке и продаже активов и просмотру своего портфеля и транзакций. Для покупки и продажи акций пользователю необходимо перейти на страницу компании. Проверка корректности введённого значения проверяется как на фронтенде, так и на бэкенде. После успешной покупки или продажи операция отобразится в соответствующем списке на странице операций пользователя. Менеджерам доступно редактирование, удаление и и добавление компаний, но недоступны покупка и продажа. Статус менеджера, также как и токены, хранится в локальном хранилище. Для доступа к методам менеджера также необходим токен аккаунта менеджера.

При нажатии на кнопку с надписью «Stocks» в верхнем меню открывается главная страница со списком акций (рис. 4). Находясь на главном экране, пользователь может найти интересующую его акцию. При нажатии на карточку пользователь перейдёт на страницу конкретной акции с графиком и описанием. Для получения данных об акции со отравляется get-запрос.

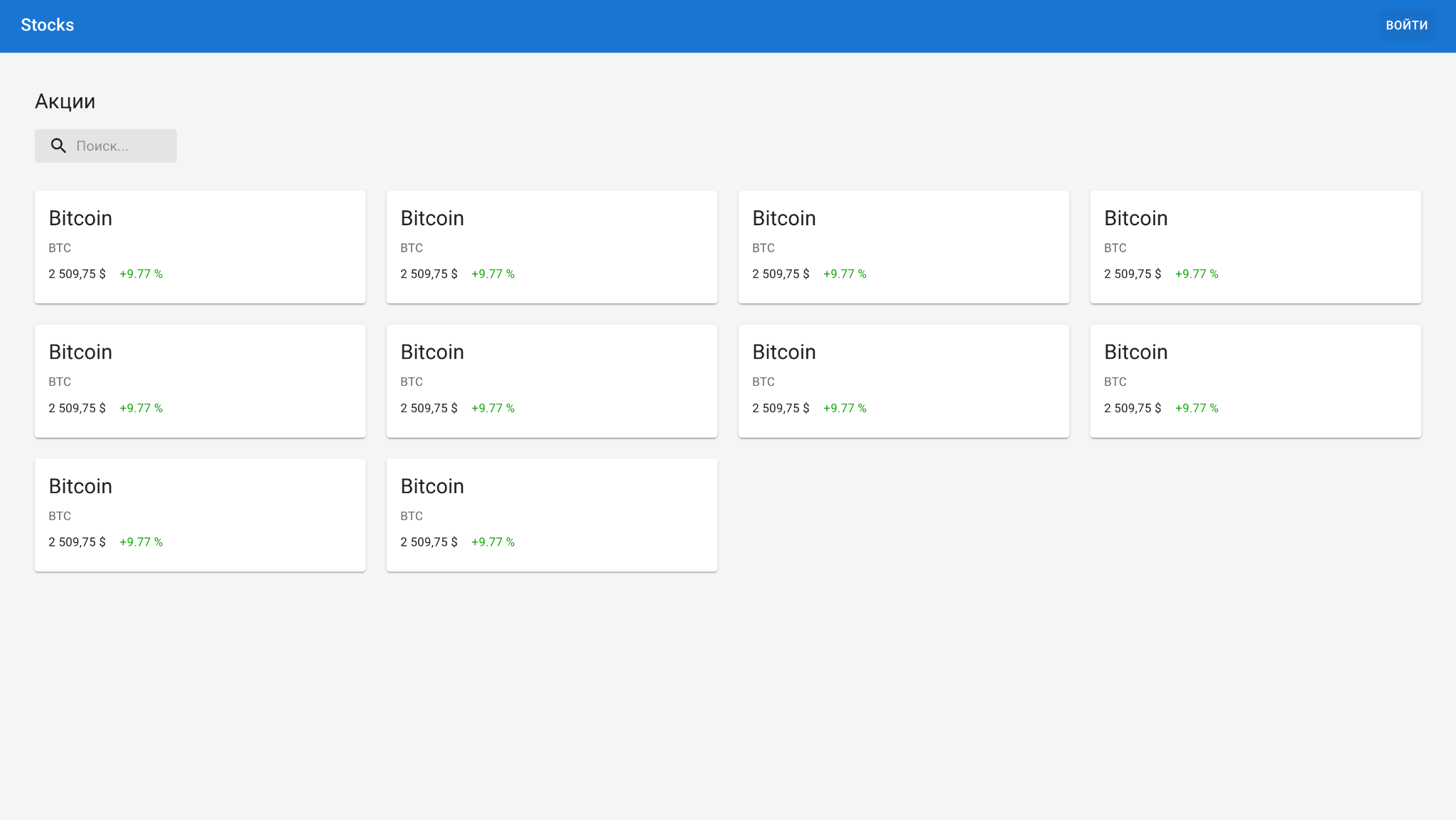


Рисунок 4 - Главный экран неавторизованного пользователя

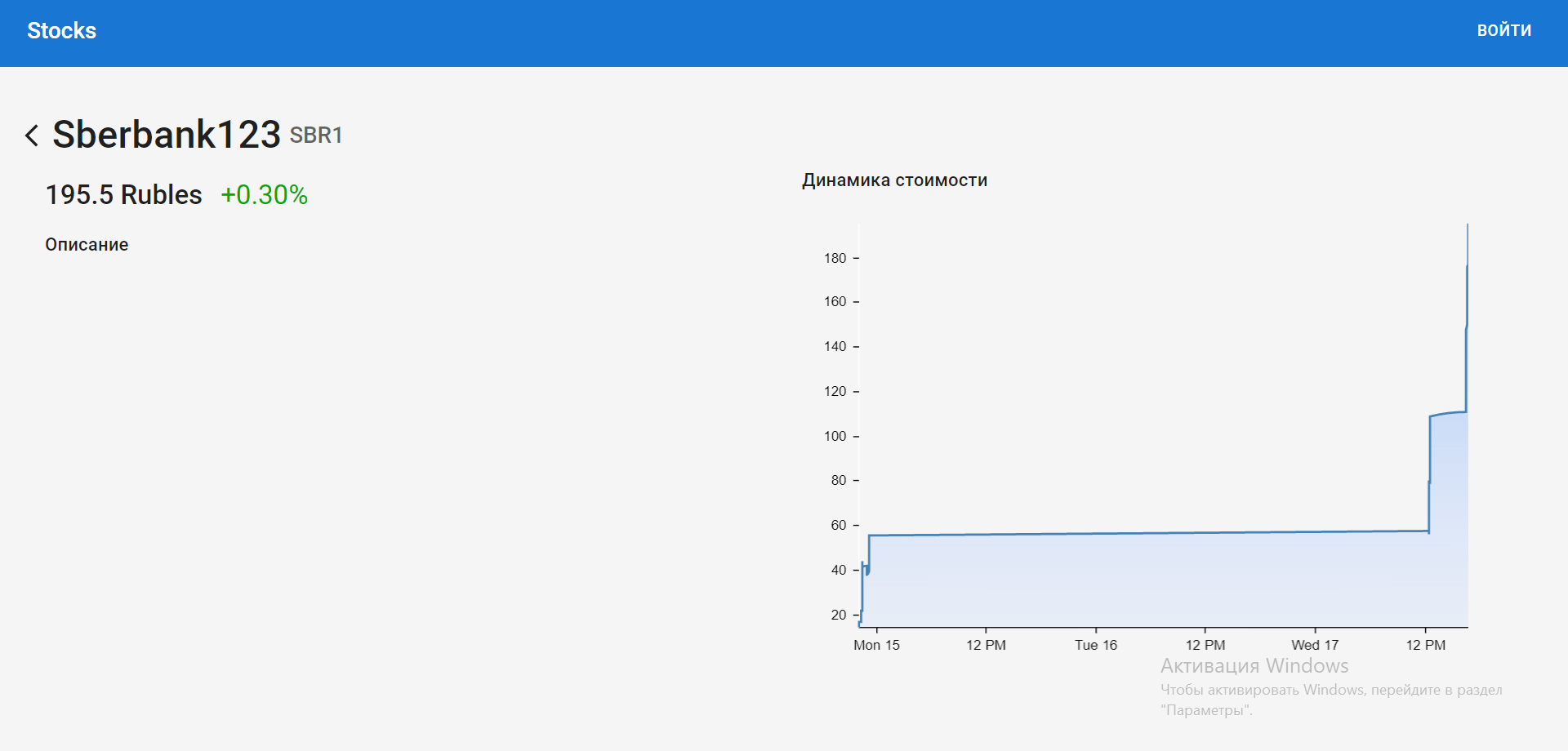


Рисунок 5 - Страница акции неавторизованного пользователя

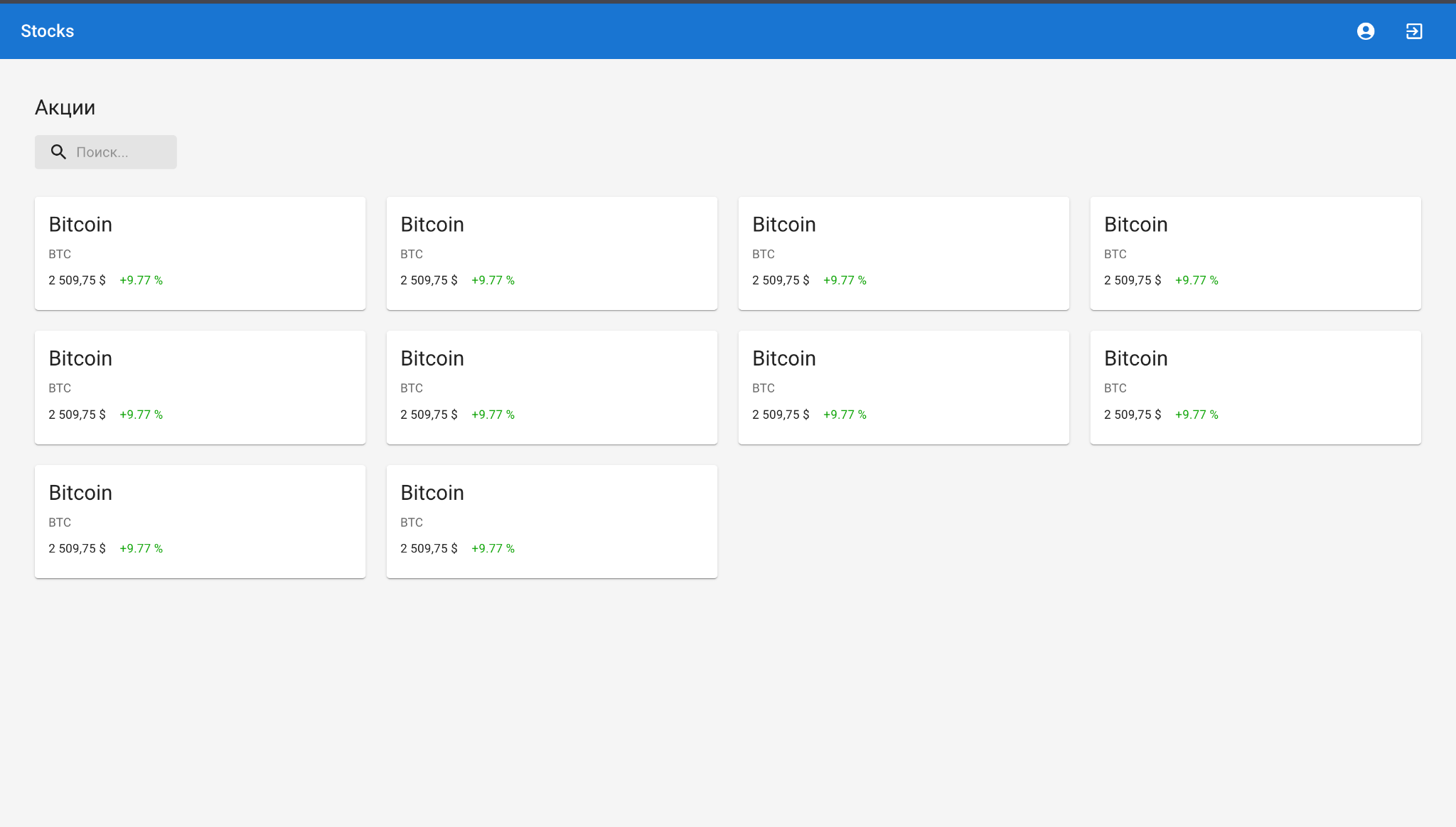
После авторизации у пользователя появляется доступ к кнопке для перехода на страницу просмотра личных транзакций и операций (рис 6). 

Рисунок 6 - Главное меню авторизованного пользователя

Также после авторизации у пользователя появляется доступ к покупке и продаже ценных бумаг (рис. 7). Проверить количество имеющихся у пользователя акций данного типа можно на этом же экране - сразу под стоимостью ценных бумаг отображается количество имеющихся акций. При совершении операций купли-продажи осуществляется post запрос на бэкенд.

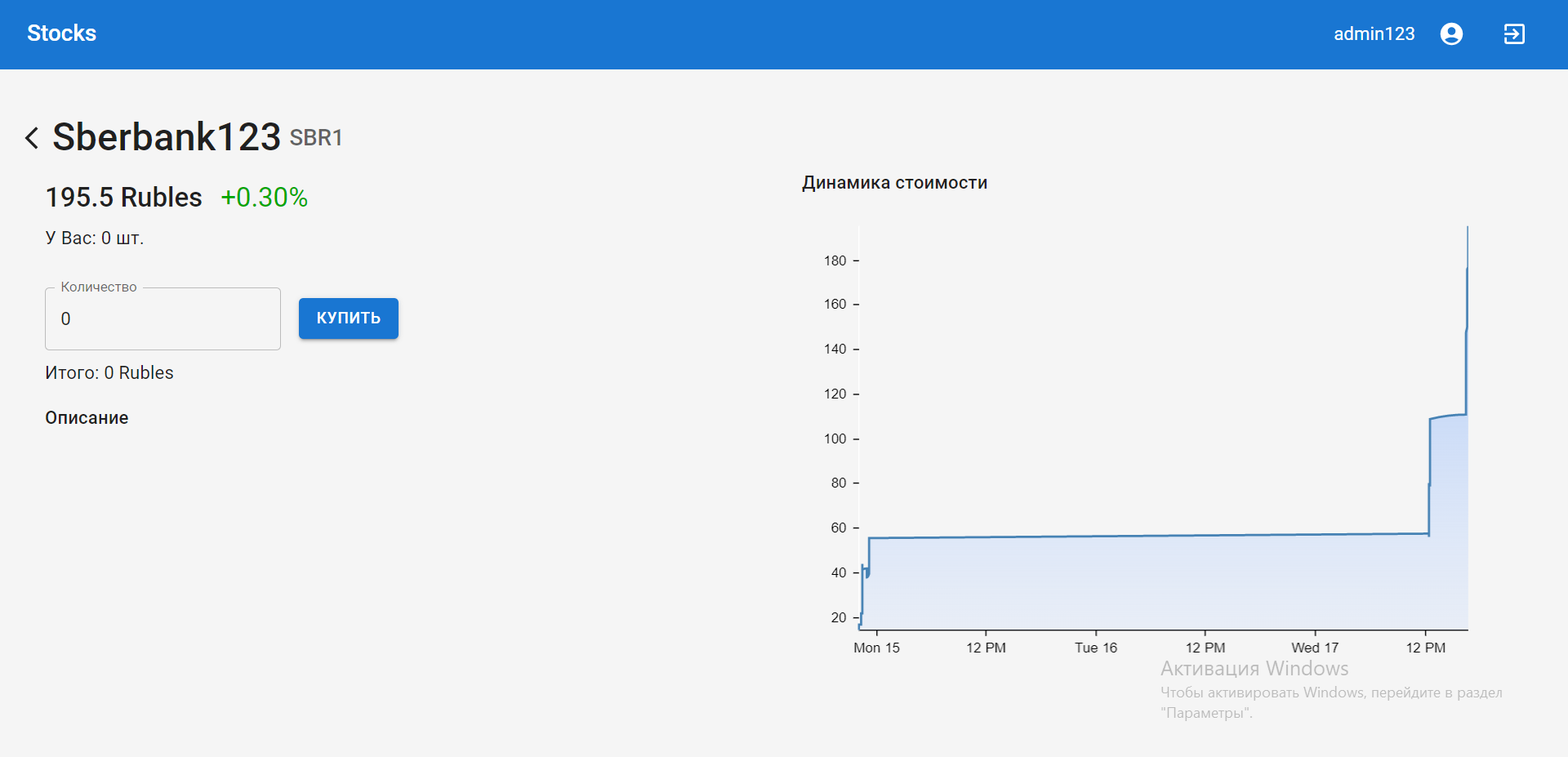


Рисунок 7 - Страница акции для авторизованного пользователя

         Если авторизованный пользователь обладает статусом менеджера, ему открывается доступ к редактированию и удалению акций. Для добавления новой акции, менеджер должен нажать на кнопку “+”, находящуюся на главном экране правее заголовка “Акции” (рис. 8).

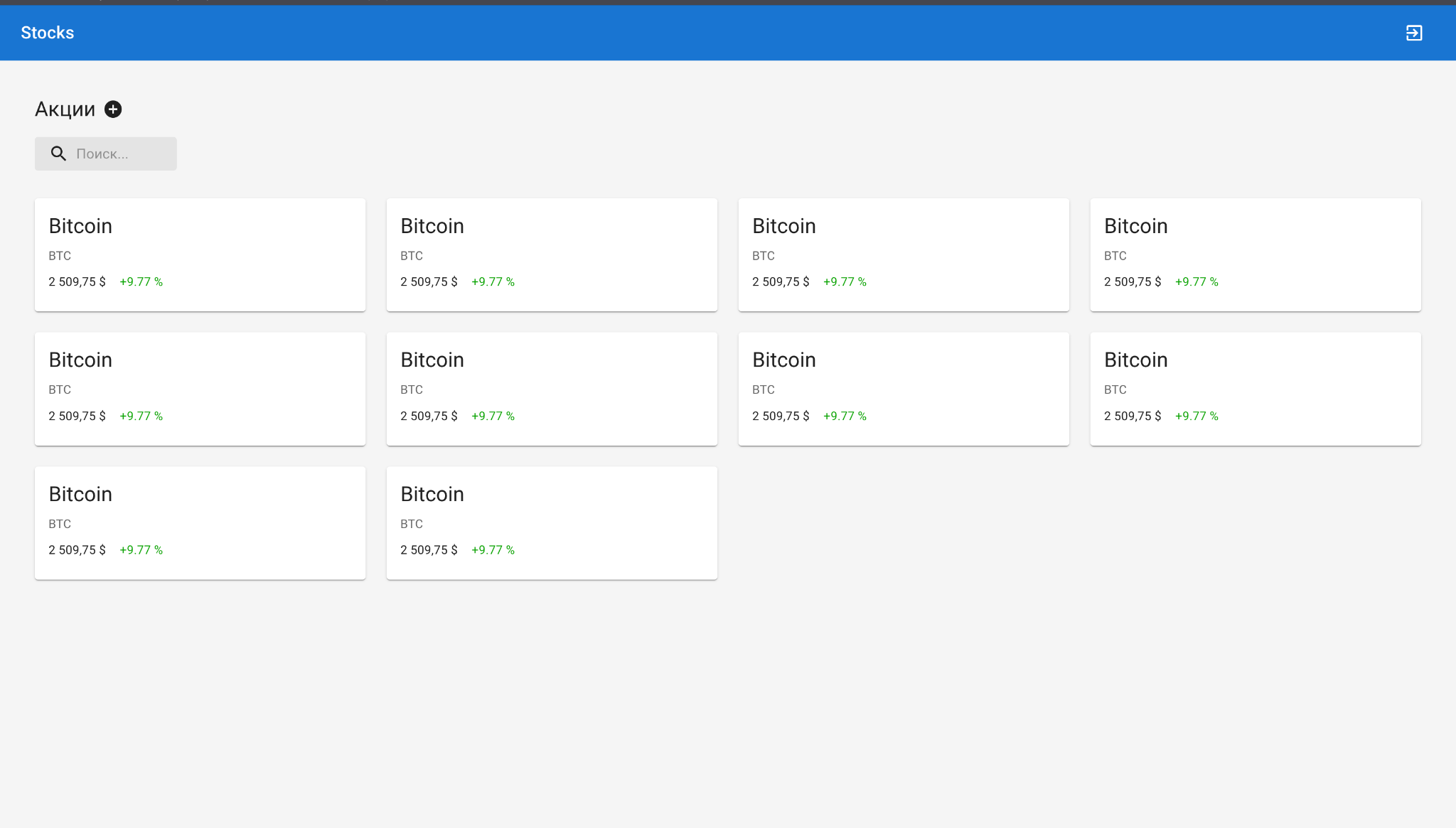


Рисунок 8 - Главное меню менеджера

Далее менеджеру необходимо заполнить данные - наименование акции, ее текущую стоимость и описание (рис 9). Для создания новой акции отправляется post запрос на бэкенд, содержащий введенную менеджером информацию.

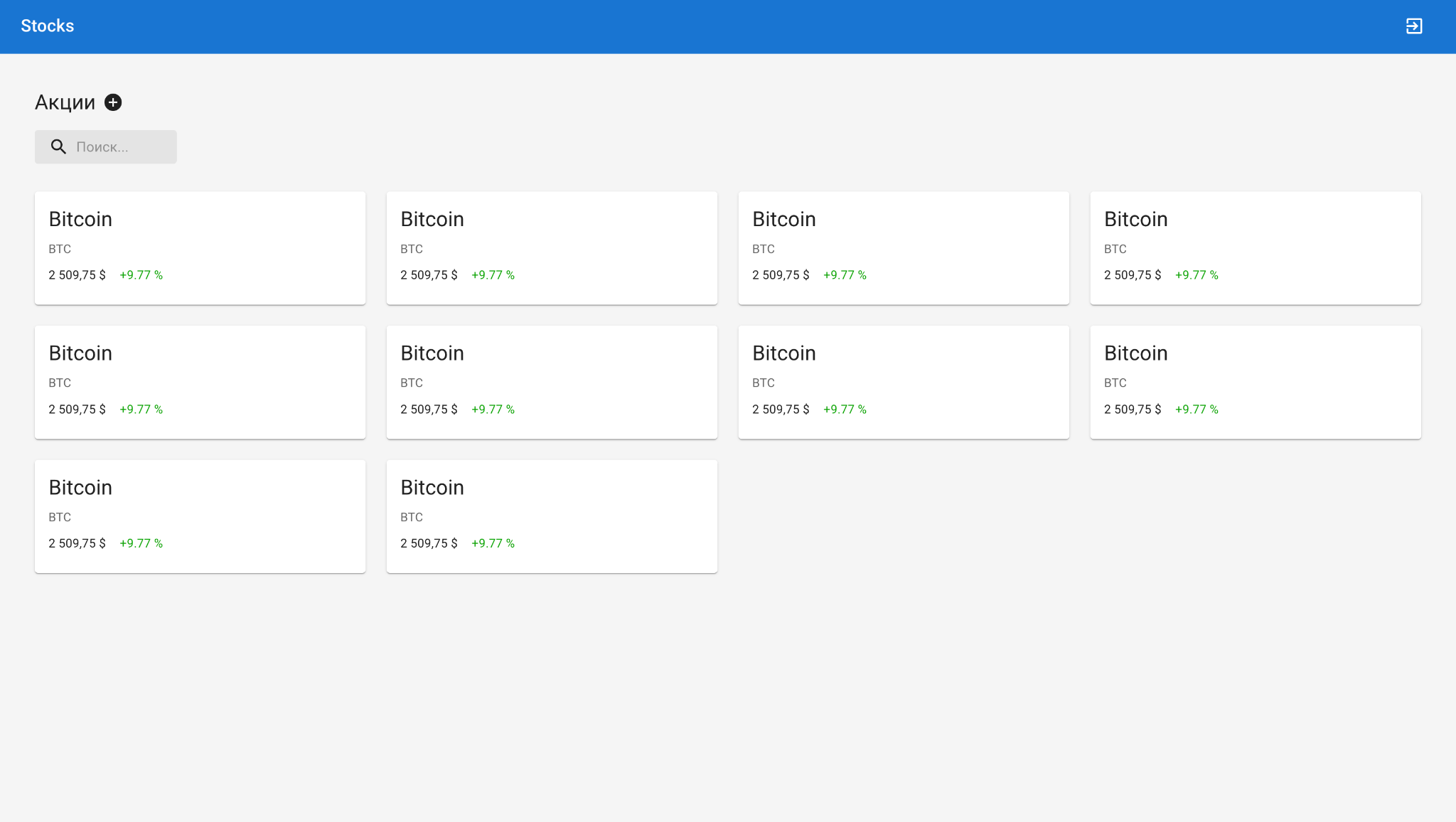


Рисунок 9 - Создание новой акции

Кроме добавления новой акции, для менеджера доступно изменение информации о существующих ценных бумагах. Для доступа к редактированию и удалению акции необходимо перейти на ее страницу и нажать соответствующие кнопки - “Редактировать” и “Удалить” (рис. 10). При редактировании данных о ценной бумаге на бэкенд отправляется put запрос, а при удалении delete запрос.

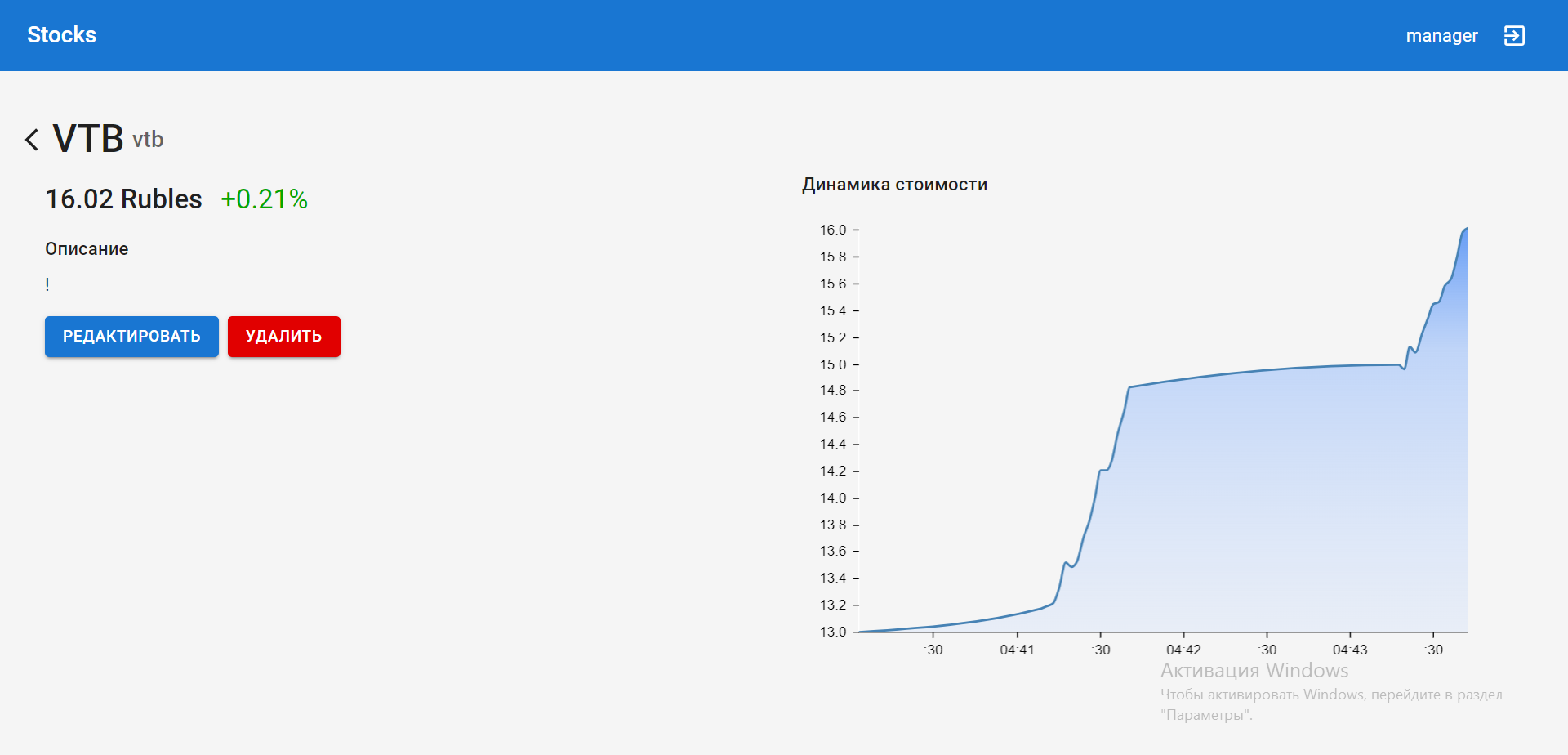


Рисунок 10 - Страница акции для менеджера

# Сервис WebSocket

Сервис WebSocket выполнен на языке Python с использованием библиотеки WebSockets. Сервис служит для передачи новых котировок акций, полученных от сервиса бэкенда на фронтенд в реальном времени. Взаимодействие с фронтендом происходит по протоколу ws, взаимодействие с сервисом бэкенда происходит по gRPC.

Websocket это протокол прикладного уровня, предоставляющий двунаправленный обмен данными между клиентом и сервером через одно логическое соединение TCP. Для подключения по ws одна из сторон отправляет HTTP запрос с заголовком upgrade, после ответа второй стороны с заголовком upgrade передача информации переходит на протокол Websocket. Далее подключение поддерживается при помощи пакетов ping/pong. Для закрытия одна из сторон отправляет пакет с кодом 1000, после подтверждения второй стороны соединение закрывается.

На сервисе WebSocket реализован метод gRPC, позволяющий бэкенду обращаться к нему и передавать новые котировки по протоколу protobuf.

gRPC - система удалённого вызова процедур, использующая HTTP/2 для передачи данных, и protobuf для их сериализации. Так как описание интерфейсов происходит на protobuf, gRPC поддерживает множество языков программирования. В отличие от API с JSON protobuf имеет строгую структуру и сериализуется в бинарные данные, что затрудняет понимание человеком, но уменьшает объём сообщений.

От сервиса WebSocket новые котировки поступают всем пользователям, при разрыве соединения полученные котировки будут кэшироваться и будут отправлены клиенту, когда оно будет восстановлено. Это обеспечивает гарантированную доставку котировок.

Работа сервиса Websocket показана на диаграмме последовательности Websocket:

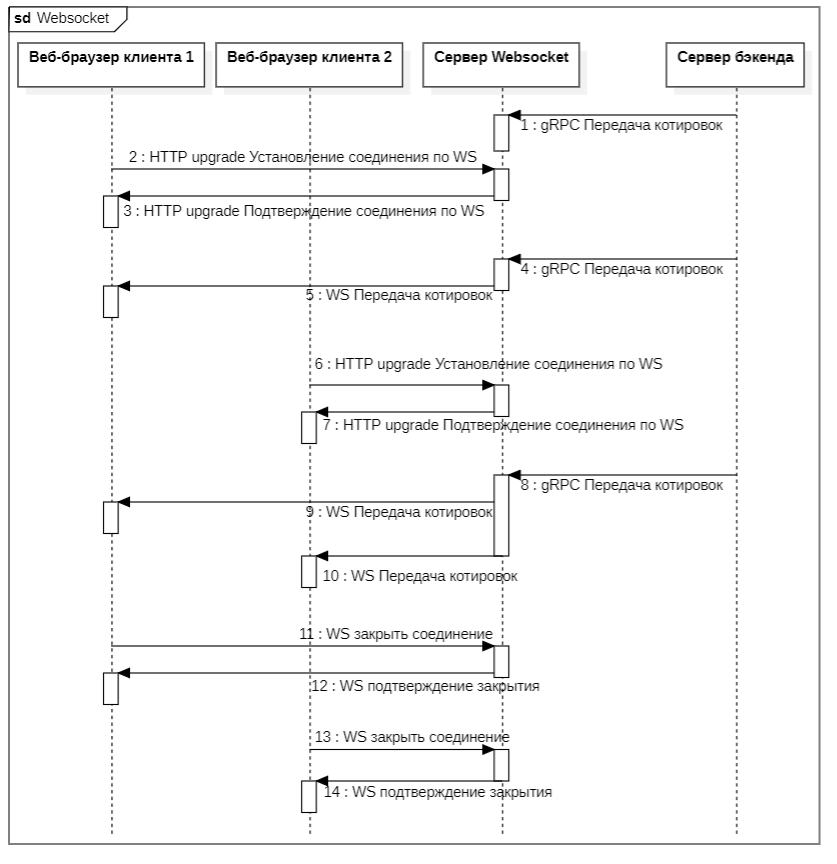


Рисунок 11 - Диаграмма последовательности Websocket

Сервис развёрнут в docker контейнере.

# Заключение

В результате выполнения курсовой работы была разработана распределённая система “Трейдерское приложение”, позволяющая хранить инфоомацию об акциях, осуществлять их покупку и продажу, а также хранить информацию об операциях и портфеле пользователя.

В процессе разработки системы были выполнены следующие задачи:

1. Сформировано техническое задание;
2. Продумано взаимодействие между компонентами системы;
3. Разработаны диаграммы в нотации UML для упрощения взаимодействия между участниками команды разработки;
4. Разработан пользовательский интерфейс веб-приложения;
5. Разработано руководство пользователя, содержащее информацию о взаимодействии конечного пользователя с системой;
6. Разработано руководство системного администратора, содержащее информацию о развёртывании системы;
7. Проведено тестирование каждого компонента системы по отдельности;
8. Выполнено пробное развёртывание системы;
9. Проведено полное тестирование развёрнутой системы;
10. Проведена отладка системы.

# Список использованных источников

1. Лутц М. Изучаем Python. Т. 1 // Диалектика .- 2019.- С 832
2. Робсон Э. Изучаем программирование на JavaScript // Издательский дом ПИТЕР .- 2022 .- С 640
3. Тиленс Т. React в действии // Издательский дом ПИТЕР .- 2019 .- С 360

# Приложения

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Н.Э. Баумана

Кафедра «Системы обработки информации и управления»

Утверждаю

\_\_\_\_\_\_\_\_\_\_\_\_\_Галкин В.А.

“\_\_\_” \_\_\_\_\_\_\_2023 г.

**Трейдерское приложение**

Техническое задание

(вид документа)

писчая бумага

(вид носителя)

7

(количество листов)

Исполнители:

Гурова М.Д.

Ищенко А.С.

Юриков Е.В.

Москва – 2023

1. Наименование:

Распределённая система, представляющая собой веб приложение для отслеживания цен акций в реальном времени.

1. Основание для разработки:

Основанием для разработки является учебный план МГТУ им. Баумана кафедры ИУ5 на 6 семестр.

1. Исполнители:

Ищенко А. С. (интеграция) – группа ИУ5-65Б

Гурова М. Д. (фронтенд) – группа ИУ5-65Б

Юриков Е. В. (бэкенд) – группа ИУ5-65Б

1. Цель разработки:

Разработать распределённую систему для отслеживания цен акций, с возможностью покупки и продажи.

1. Содержание работы:
   1. Функциональные требования:
      1. Фронтенд:
         1. Разработка страницы авторизации:

* На странице авторизации пользователю предлагается ввести свои учётные данные для входа в систему: логин и пароль;
* Доступ к странице имеют только неавторизованные пользователи.
  + - 1. Разработка страницы регистрации:
* На странице регистрации пользователю предлагается ввести новые учётные данные: логин и пароль;
* Доступ к странице имеют только неавторизованные пользователи.
  + - 1. Разработка главной страницы:
* Авторизованному пользователю доступен список акций, просмотр своего портфеля и просмотр своих операций;
* Неавторизованному пользователю доступен только список акций.
  + - 1. Разработка страницы просмотра котировок и графика цены акций конкретной компании:
* На странице пользователь получает данные о компании: текущую цену акции, график и описание компании;
* Авторизованному пользователю также доступны функции покупки и продажи.
  + - 1. Разработка страницы добавления и редактирования компаний:
* Доступ к странице имеют только менеджеры;
* На странице можно добавлять, удалять и редактировать имеющиеся компании.
  + - 1. Разработка модуля Websocket для фронтенда:
* Websocket-соединение устанавливается для всех пользователей, независимо от авторизации;
* По Websocket новая котировка передаётся фронтенду, далее фронтенд отображает изменившуюся цену акций на главной странице и на странице отдельной компании.
  + 1. Бэкенд:
       1. HTTP методы:
          1. Разработка метода регистрации, для создания аккаунта для взаимодействия с системой:

При регистрации запись пользователя заносится в базу данных, пароль шифруется;

Доступ к методу имеют только гости.

* + - * 1. Разработка метода авторизации для получения доступа к системе:

При авторизации пользователь получает JWT токены (access и refresh) access действителен 5 минут, refresh - 1 сутки;

Доступ к методу имеют только гости.

* + - * 1. Разработка метода для получения данных о компаниях, представленных на рынке, и котировок их акций:

Доступ к методу имеют все пользователи, в том числе гости.

* + - * 1. Разработка метода для покупки и продажи активов:

Доступ к методу имеют только авторизованные пользователи;

Пользователь имеет доступ только к своим активам.

* + - * 1. Разработка метода для просмотра портфеля пользователя:

Доступ к методу имеют только авторизованные пользователи;

Пользователи имеют доступ только к своему портфелю.

* + - * 1. Разработка метода для просмотра списка операций пользователя:

Доступ к методу имеют только авторизованные пользователи;

Пользователи имеют доступ только к своему списку операций.

* + - * 1. Разработка метода для добавления, редактирования и удаления компаний:

Доступ к методу имеют только менеджеры.

* + - * 1. Разработка метода получения временного ряда акций компании:

Доступ к методу имеют все пользователи, в том числе гости.

* + 1. WebSocket:
       1. Разработка метода передачи новых цен акций пользователю:
* Сервис Websocket отправляет подключенным к вебсокету пользователям, новые котировки акций;
* Передача данных происходит по протоколу Websocket.
  + - 1. gRPC методы:
         1. Разработка метода получения от сервиса бэкенда новых котировок акций.
  1. Пользователи и доступный им функционал:
     1. Гость:
        1. Просмотр списка компаний и акций;
        2. Регистрация в системе;
        3. Авторизация в системе.
     2. Авторизованный пользователь:
        1. Просмотр списка компаний и акций;
        2. Покупка и продажа активов;
        3. Просмотр личного портфеля.
     3. Менеджер:
        1. Добавление компаний в систему.
     4. Администратор:
        1. Прямой доступ к базам данных.

1. Требования к составу технических средств:

Программное изделие выполняется на Python (Django Rest Framework) и JavaScript (React) на компьютере под управлением ОС Windows. Также необходим сервер базы данных PostgreSQL. Для взаимодействия по Websocket используется Python библиотека Websockets.

Для демонстрации работы программы требуется пк для разворачивания системы и пк для её использования.

1. Этапы разработки:
   1. Разработать черновое ТЗ – 3 неделя;
   2. Разработать чистовое ТЗ, макет figma, диаграмму прецедентов, ER-диаграмму, диаграмму развёртывания и настроить Swagger – 8 неделя;
   3. Разработать полный комплект документов – 12 неделя;
   4. Разработать и отладить приложение – 15 неделя.
2. Техническая документация, предъявляемая по окончании работы:
   1. Технический проект:

Расчётно-пояснительная записка, включающая в приложении комплект технической документации на программный продукт, содержащий:

* Приложение 1 – Техническое задание
* Приложение 2 – Программа и методика испытаний
* Приложение 3 – Руководство пользователя
* Приложение 4 – Руководство системного администратора
* Приложение 5 – Диаграмма последовательности бэкенда
* Приложение 6 – Диаграмма последовательности Websocket
* Приложение 7 – Диаграмма деятельности фронтенд
* Приложение 8 – Диаграмма прецедентов
* Приложение 9 – ER-диаграмма
* Приложение 10– Диаграмма развёртывания
  1. Папка с технической и программной документацией в формате:

<группа>\_ <Фамилия И.О. студента> \_КР\_СТ\_в\_АСОИУ.zip.

1. Порядок приёма работы:

Прием работы осуществляется в соответствии с "Программой и методикой испытаний".

1. Дополнительные условия:

Данное Техническое Задание может дополняться и изменяться в установленном порядке.

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Н.Э. Баумана

Кафедра «Системы обработки информации и управления»

Утверждаю

\_\_\_\_\_\_\_\_\_\_\_\_\_Галкин В.А.

“\_\_\_” \_\_\_\_\_\_\_2023 г.

**Трейдерское приложение**

Программа и методика испытаний

(вид документа)

писчая бумага

(вид носителя)

19

(количество листов)

Исполнители:

Гурова М.Д.

Ищенко А.С.

Юриков Е.В.

Москва – 2023

# Объект испытаний:

Объектом испытаний является распределённая система "Трейдерское приложение"

# Цель испытаний

Целью проведения испытаний является доказательство работоспособности описанного в пункте 1 объекта испытаний.

# Требования к объекту испытаний

Требования к объекту испытаний представлены в документе «Техническое задание».

# Требования к программной документации

Во время проведения испытания должны быть представлены следующие документы:

1. Техническое задание;
2. Программа и методика испытаний.

# Средства и порядок испытаний

Перед тестированием должна быть проведена установка и настройка системы согласно документу “Руководство системного администратора”.

## Бэкенд

Для тестирования бэкенда необходимо в запустить сервис бэкенда и при помощи Postman отправлять запросы на соответствующие URL на порт 8000. Для получения доступа к некоторым методам необходима авторизация, для этого надо в хедерах запроса указать хедер Authorization со строкой “Token (access token)”.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Пункт ТЗ | Выполняемое действие | Результат | Иллюстрации |
| 1 | 5.1.2.1.1 | Регистрация:  Необходимо отправить POST запрос с именем пользователя и паролем в теле запроса. | Пользователю придёт ответ с введённым именем. | Рисунок 11 |
| 2 | 5.1.2.1.2 | Авторизация: Необходимо отправить POST запрос с именем пользователя и паролем в теле запроса. | Пользователю вернутся access и refresh токены | Рисунок 12 |
| 3 | 5.1.2.1.3 | Получение списка компаний: Необходимо отправить пустой GET запрос. | Пользователю будет отправлен массив объектов с данными о компаниях. | Рисунок 13 |
| 4 | 5.1.2.1.4 | Покупка и продажа активов: Необходимо авторизоваться и отправить POST запрос, в теле запроса указать id компании и количество акций, id пользователя берётся из токена. | Пользователю будет отправлена совершённая им операция. | Рисунок 14 |
| 5 | 5.1.2.1.5 | Просмотр портфеля пользователя: Необходимо авторизироваться и отправить пустой GET запрос. | Пользователю будет отправлен массив с данными о его активах. | Рисунок 15 |
| 6 | 5.1.2.1.6 | Просмотр списка операций пользователя: Необходимо авторизироваться и отправить пустой GET запрос. | Пользователю будет отправлен массив с данными о его операциях. | Рисунок 16 |
| 7 | 5.1.2.1.7 | Добавление, редактирование, удаление компаний: Для получения доступа к этим методам необходимо авторизоваться через аккаунт менеджера. Для добавления компании необходимо отправить POST запрос, в теле запроса указать данные добавляемой компании.  Для редактирования компании необходимо отправить PUT запрос, в теле запроса указать новые данные редактируемой компании.  Для удаления компании необходимо отправить пустой DELETE запрос. | Пользователю будет отправлена добавленная, изменённая или удалённая компания. | Рисунок 17  Рисунок 18  Рисунок 19 |
| 8 | 5.1.2.1.8 | Получение временного ряда акций компании: Для получения временного ряда акций необходимо отправить пустой GET запрос. | Временной ряд акций, компании, хранящейся в базе данных | Рисунок 20 |

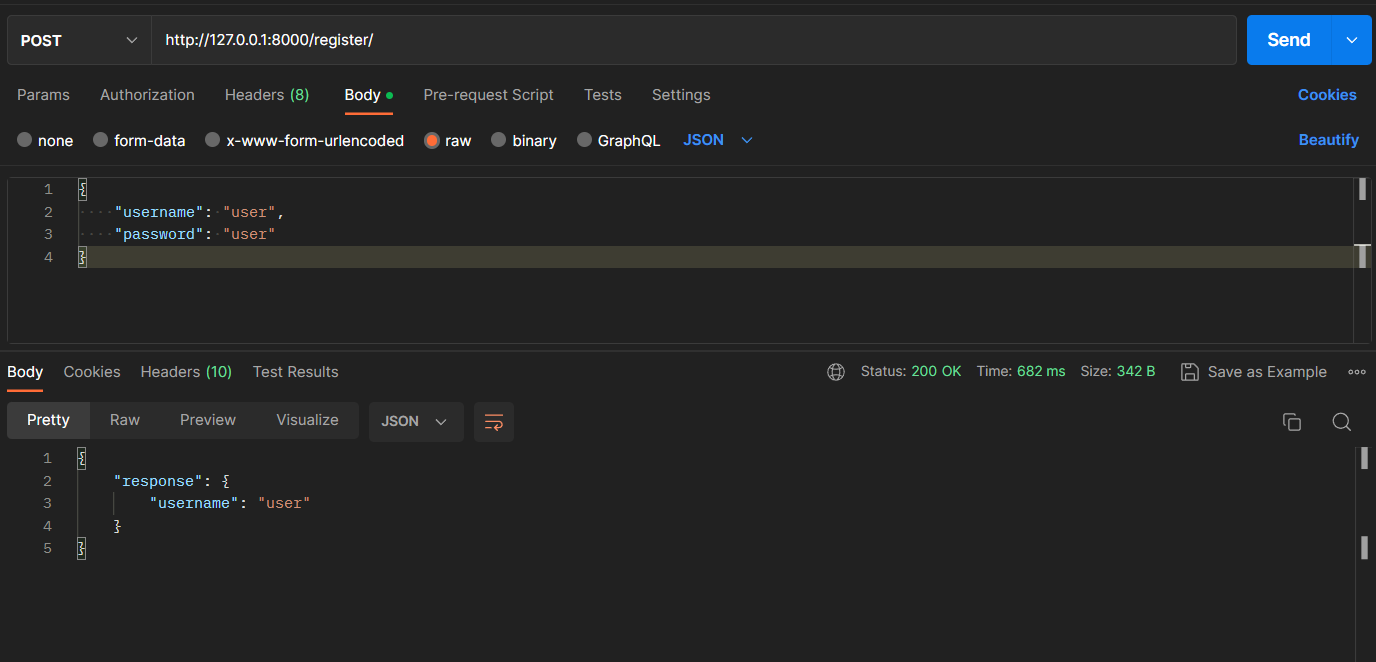


Рисунок 12 - Пример POST запроса (регистрация) и ответа на него

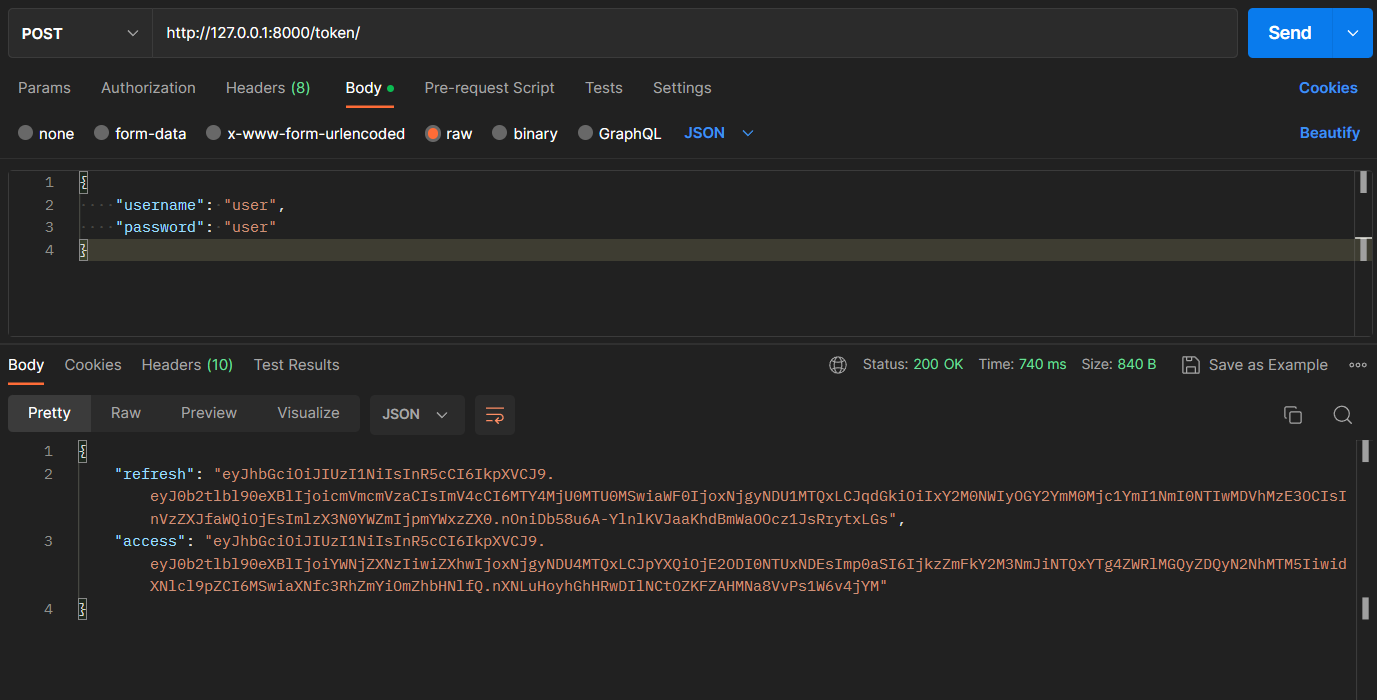


Рисунок 13 - Пример POST (авторизация) запроса и ответа на него

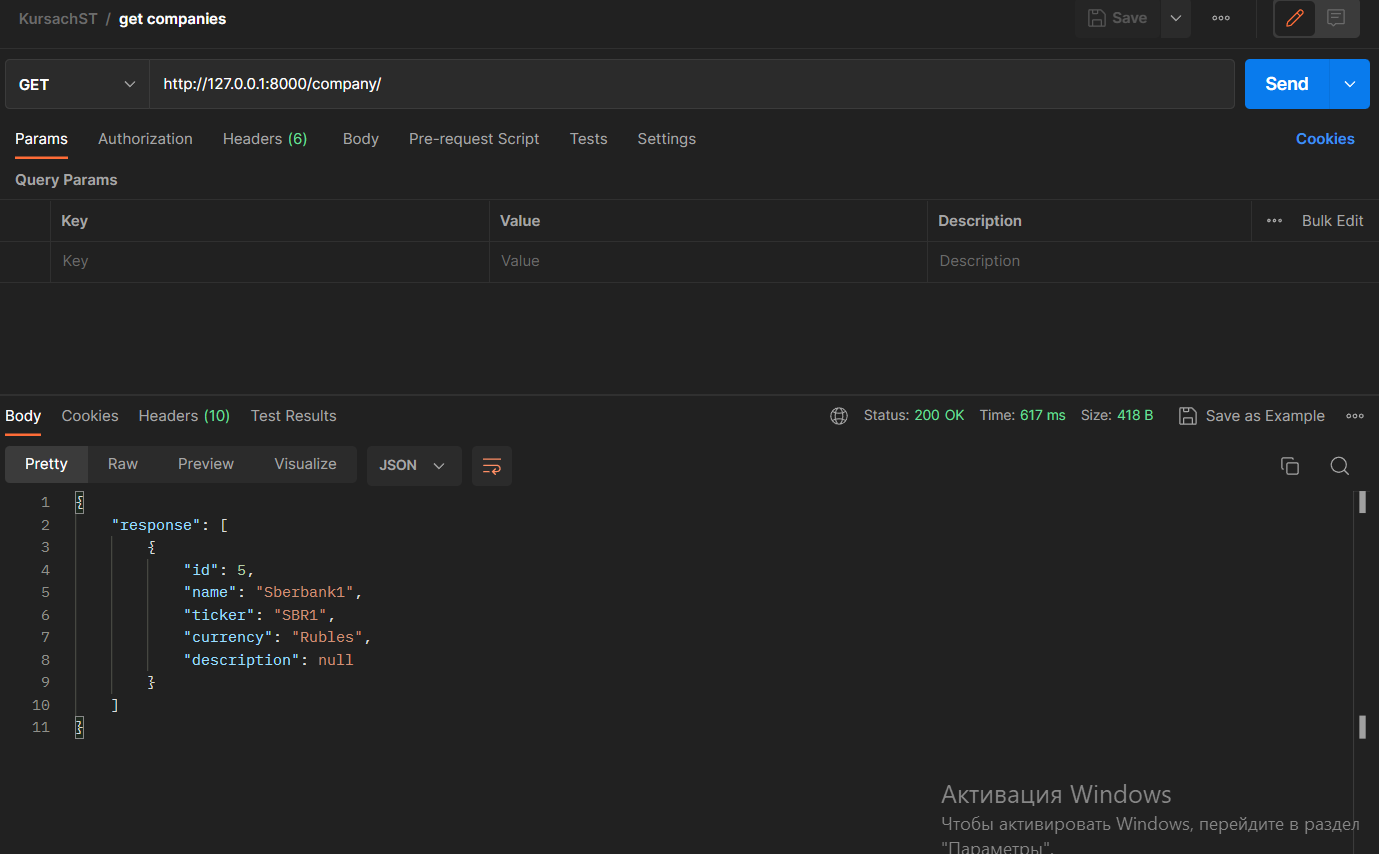


Рисунок 14 - Пример GET (получение списка компаний) запроса и ответа на него

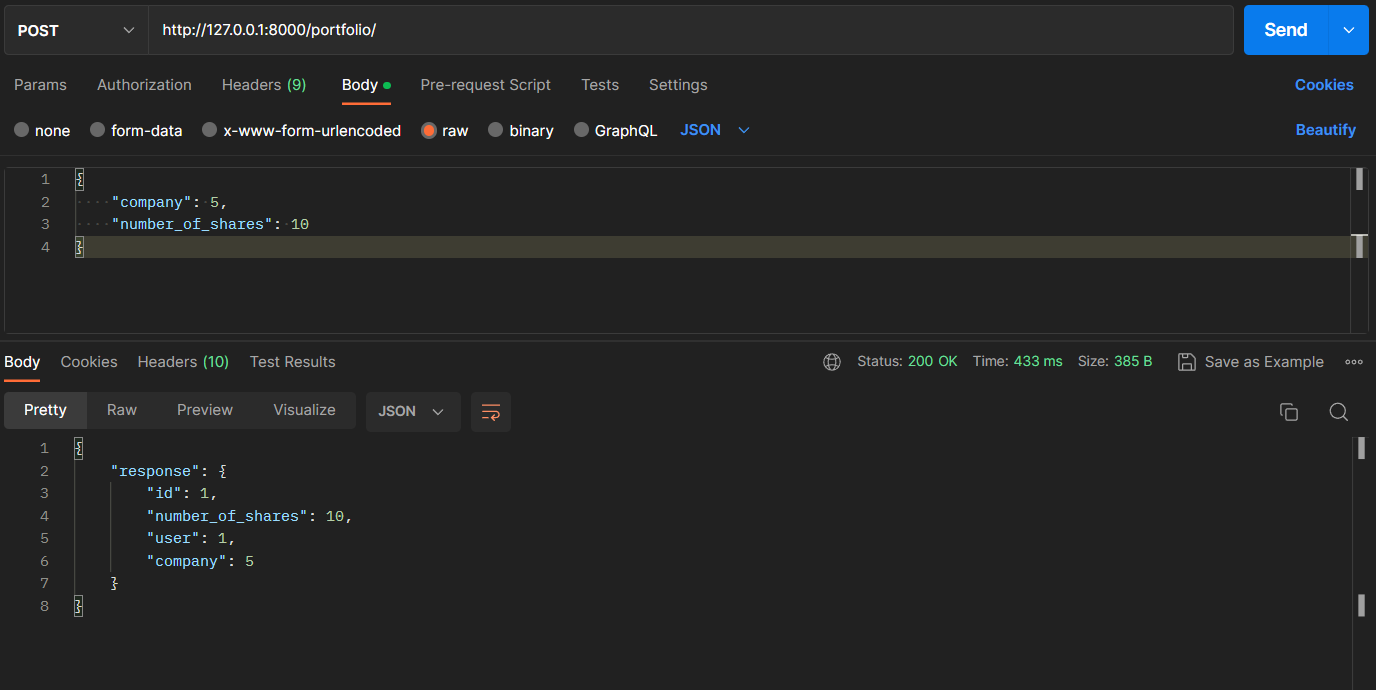


Рисунок 15 - Пример POST (покупка и продажа активов). запроса и ответа на него

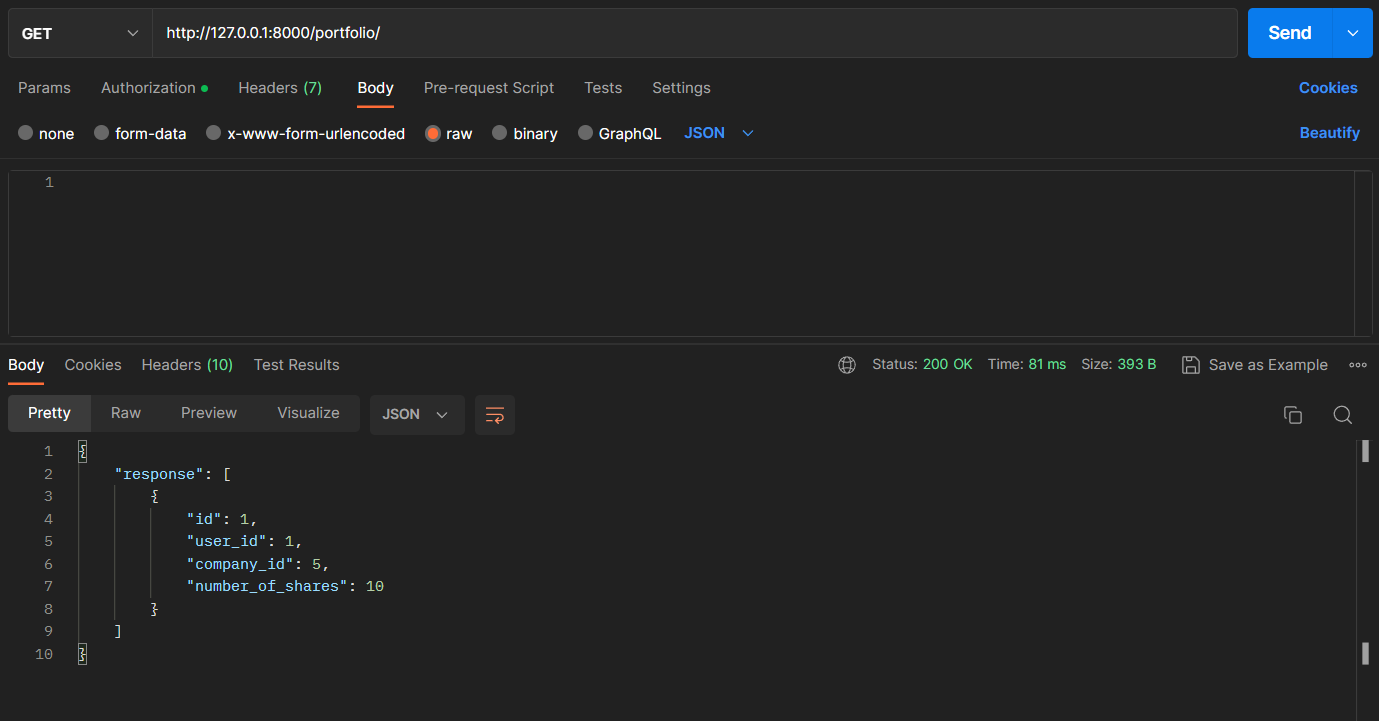


Рисунок 16 - Пример GET (просмотр портфеля пользователя) запроса и ответа на него

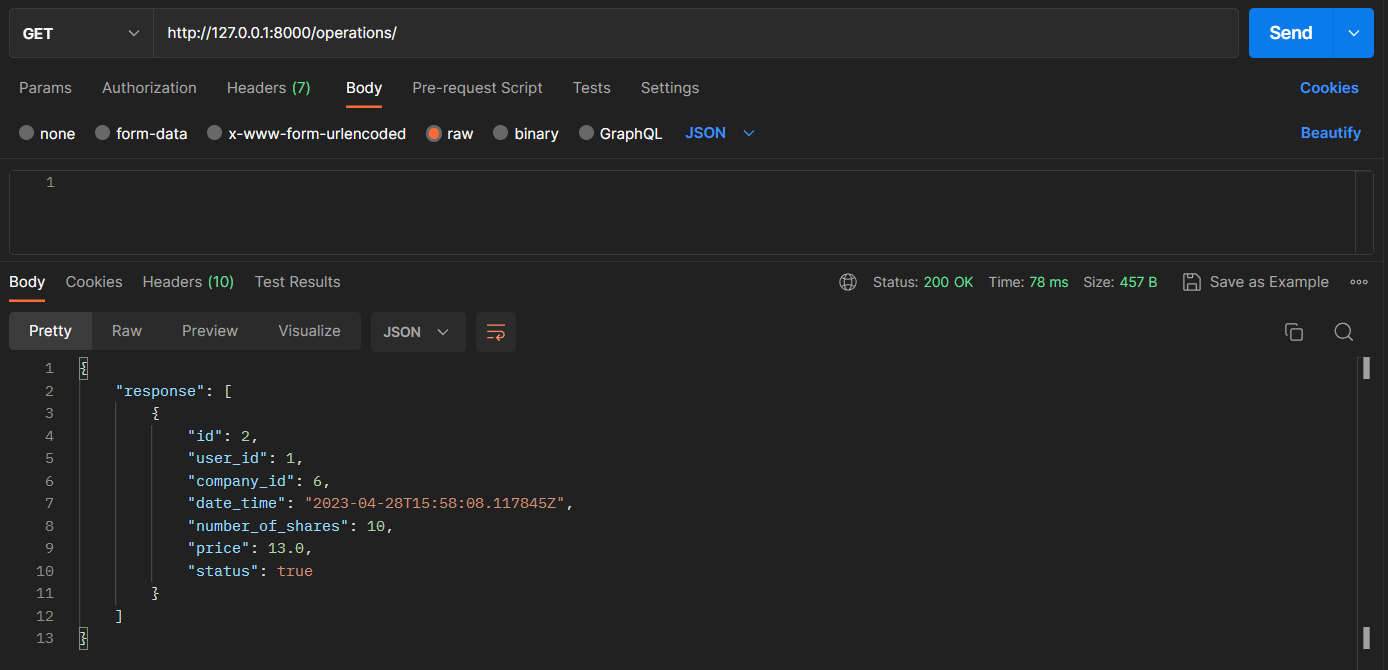


Рисунок 17 - Пример GET (просмотр операций пользователя) запроса и ответа на него

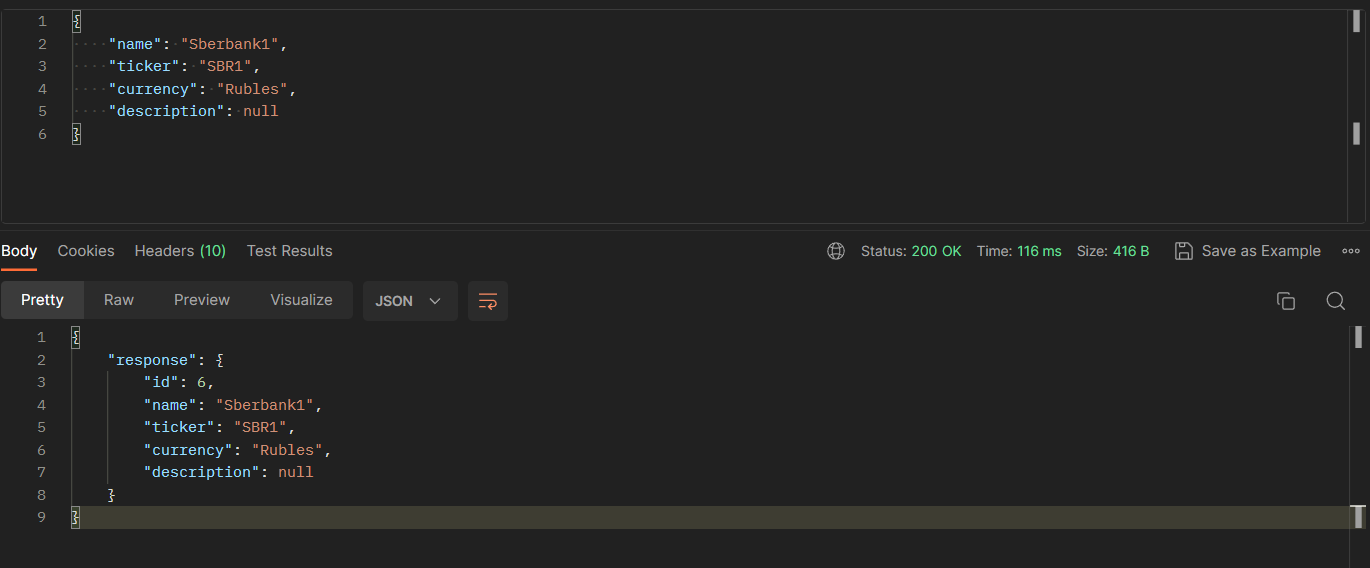


Рисунок 18 - Пример POST (добавление компании) запроса и ответа на него

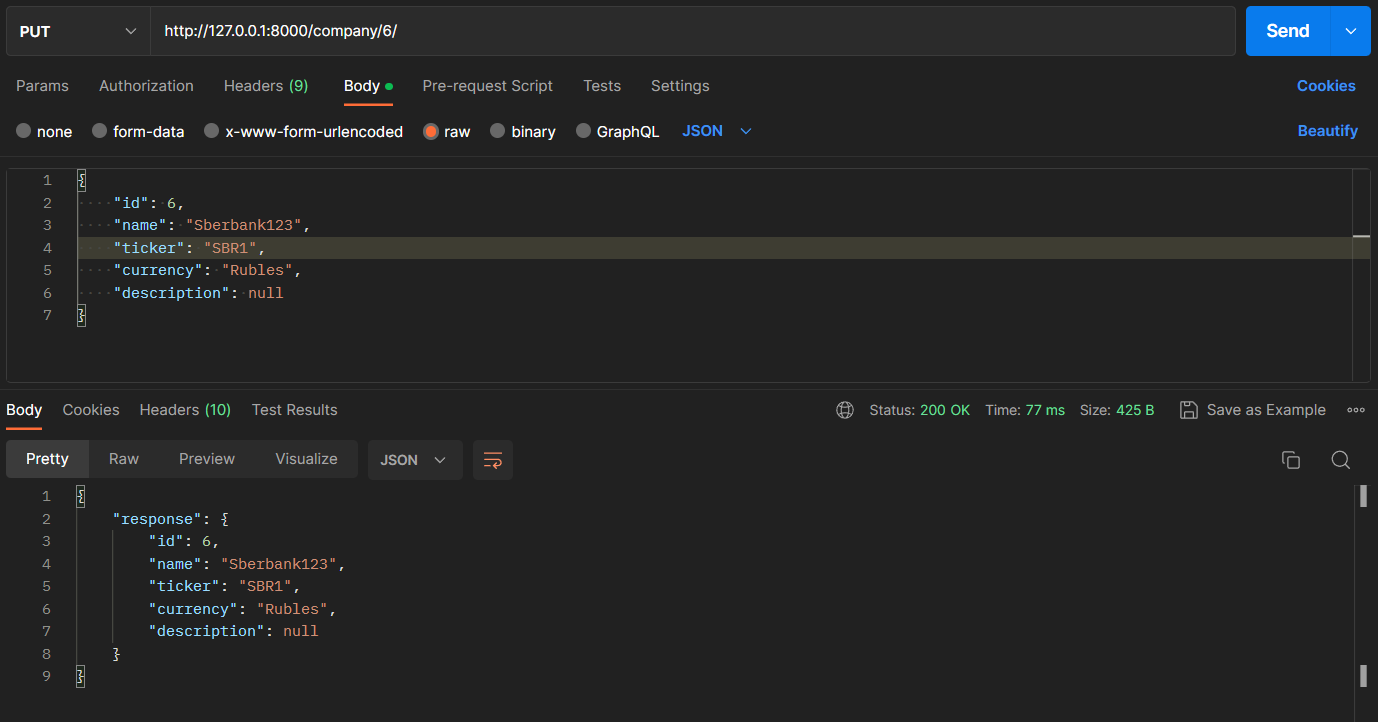


Рисунок 19 - Пример PUT (редактирование компании) запроса и ответа на него

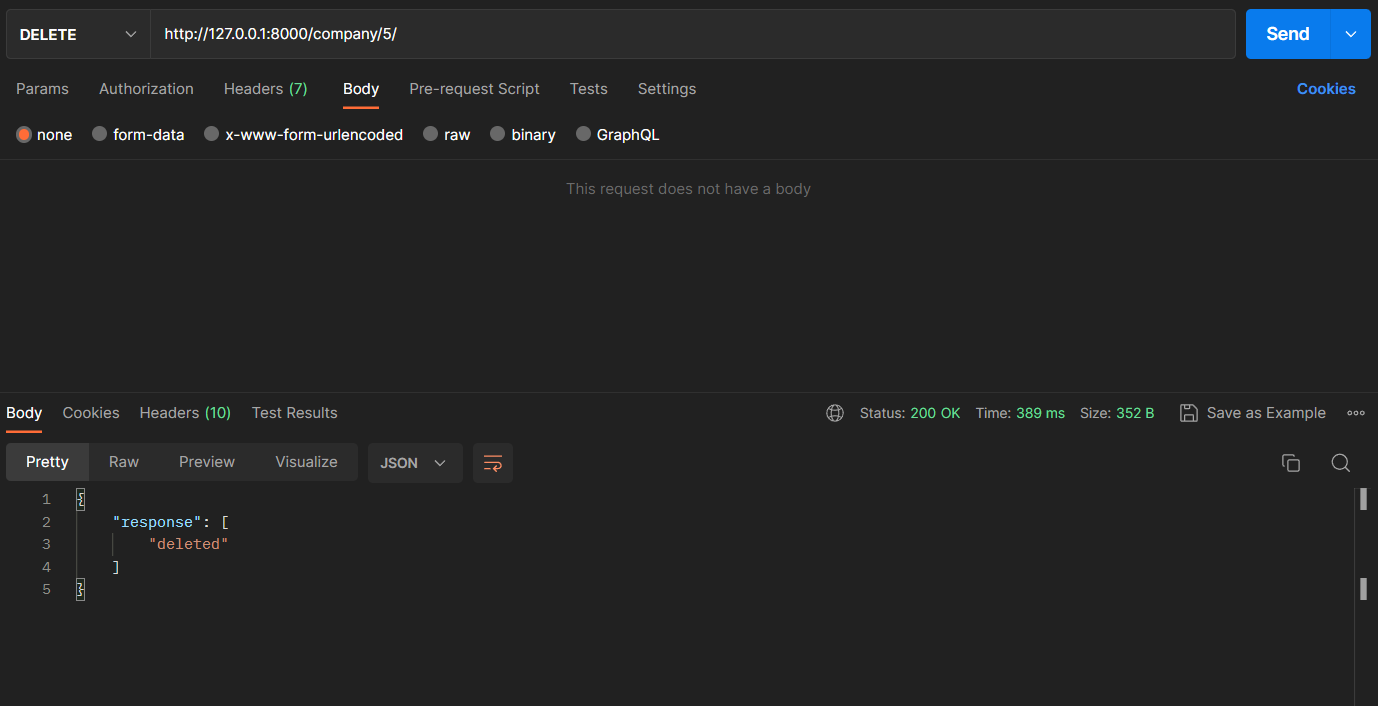


Рисунок 20 - Пример DELETE (удаление компании) запроса и ответа на него

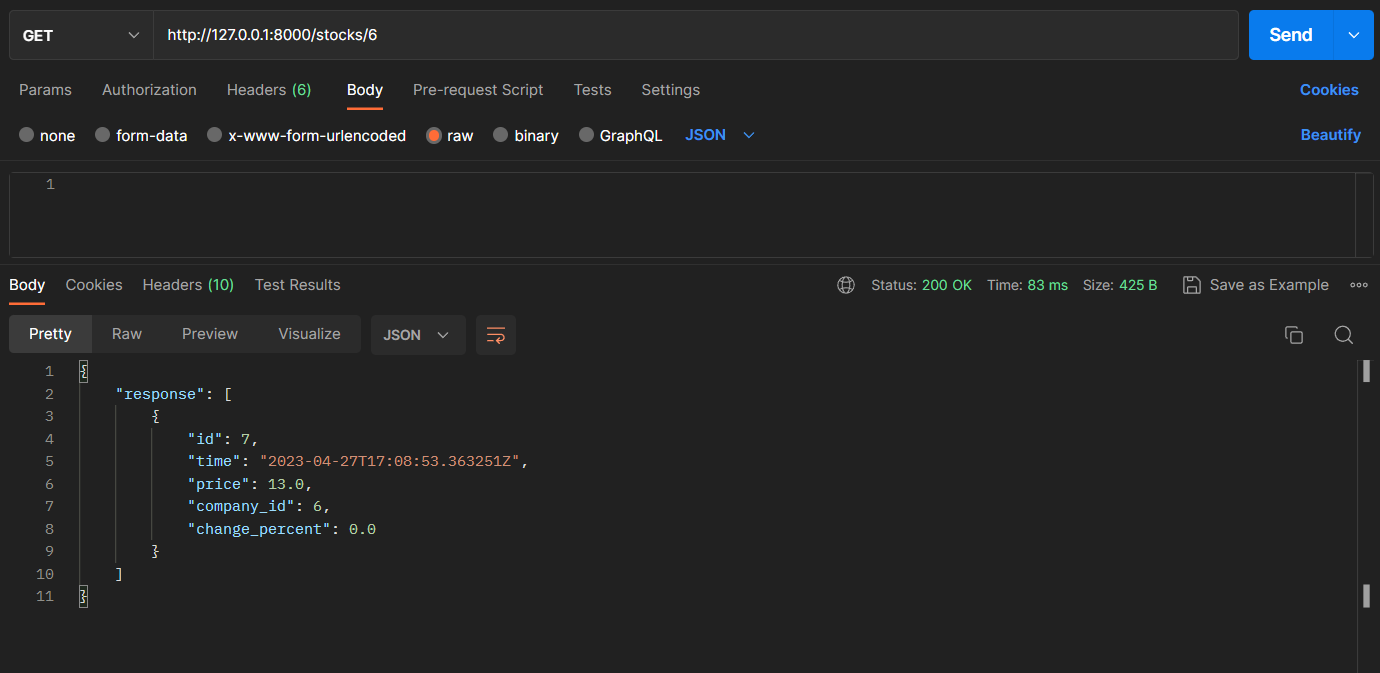


Рисунок 21 - Пример GET (просмотр временного ряда акций) запроса и ответа на него

## Фронтенд

Для тестирования фронтенда необходимо в браузере сервера фронтенда перейти на localhost:3000.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Пункт ТЗ | Выполняемое действие | Результат | Иллюстрации |
| 1 | 5.1.1.1 | Авторизация:  Ввод данных пользователя. | Авторизация и открытие главной страницы. | Рисунок 21  Рисунок 24  Рисунок 25 |
| 2 | 5.1.1.2 | Регистрация: Ввод новых данных пользователя. | Регистрация нового пользователя, авторизация в системе, открытие главной страницы. | Рисунок 22  Рисунок 24 |
| 3 | 5.1.1.3 | Получение списка компаний: Нажатие на кнопку Stocks в левом верхнем углу. | Открытие главной страницы со списком доступных компаний. | Рисунок 23 |
| 4 | 5.1.1.4 | Просмотр портфеля и списка транзакций: Авторизация без прав менеджера, нажатие на кнопку пользователя в правом верхнем углу. | Открытие страницы с портфелем и списком транзакций пользователя. | Рисунок 29 |
| 5 | 5.1.1.5 | Добавление компаний: Авторизация с правами менеджера, нажатие на кнопку +. | Переход на страницу добавления компании. | Рисунок 25 |
| 6 | 5.1.1.6 | Просмотр страницы компании: Нажатие на одну из карточек компаний на главной странице | Переход на страницу просмотра компании. | Рисунок 26  Рисунок 27 |
| 7 | 5.1.1.7 | Покупка и продажа активов:  Авторизоваться без прав менеджера, перейти на страницу компании, ввести количество акций, нажать кнопку купить. | Количество купленных акций отобразится на карточке компании. | Рисунок 27 |
| 8 | 5.1.1.8 | Удаление и редактирование кампаний:  Авторизоваться с правами менеджера, перейти на страницу компании и нажать удалить для удаления, и редактировать для редактирования. | Редактирование или удаление компании. | Рисунок 28 |

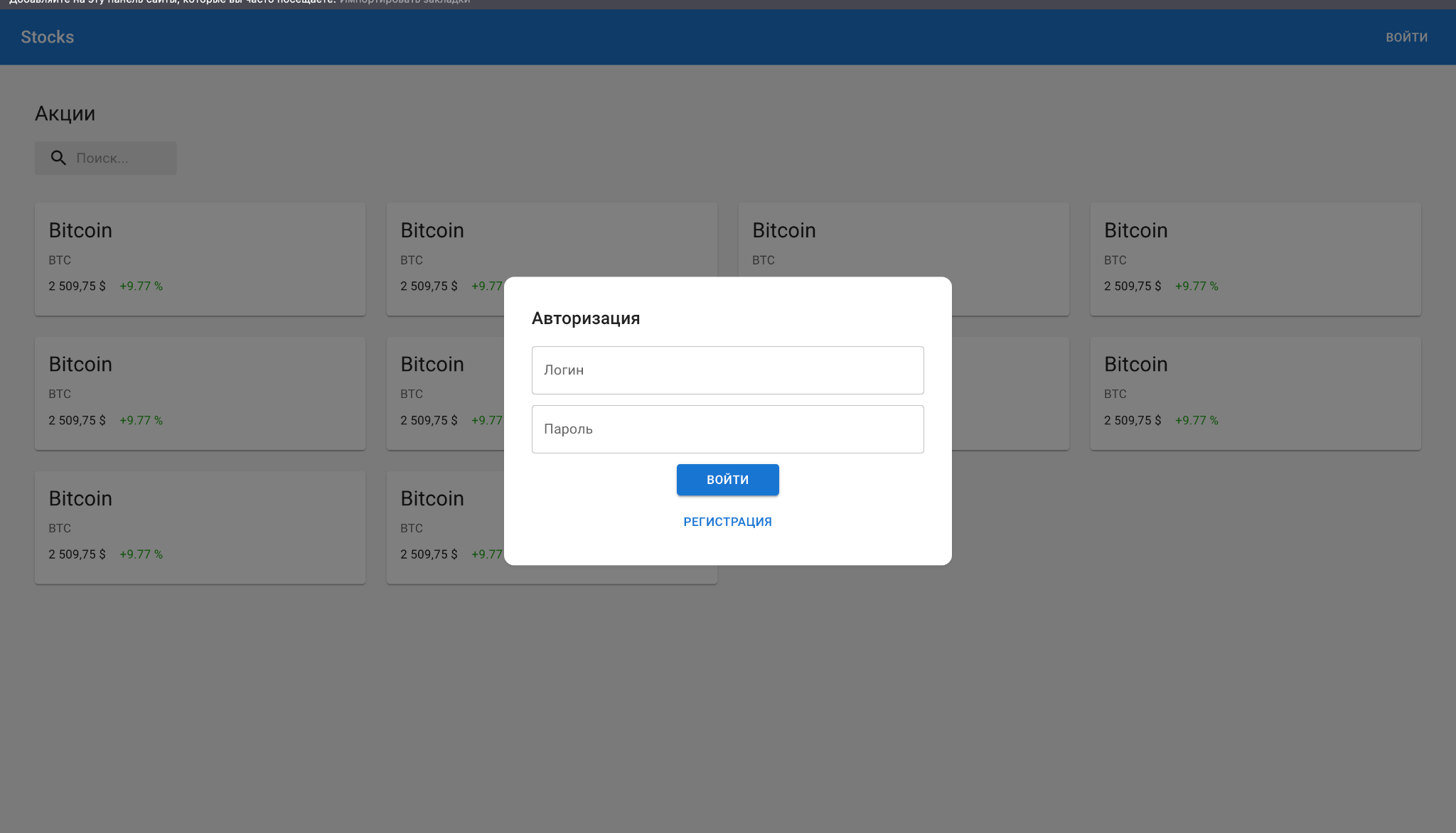


Рисунок 22 - Окно авторизации

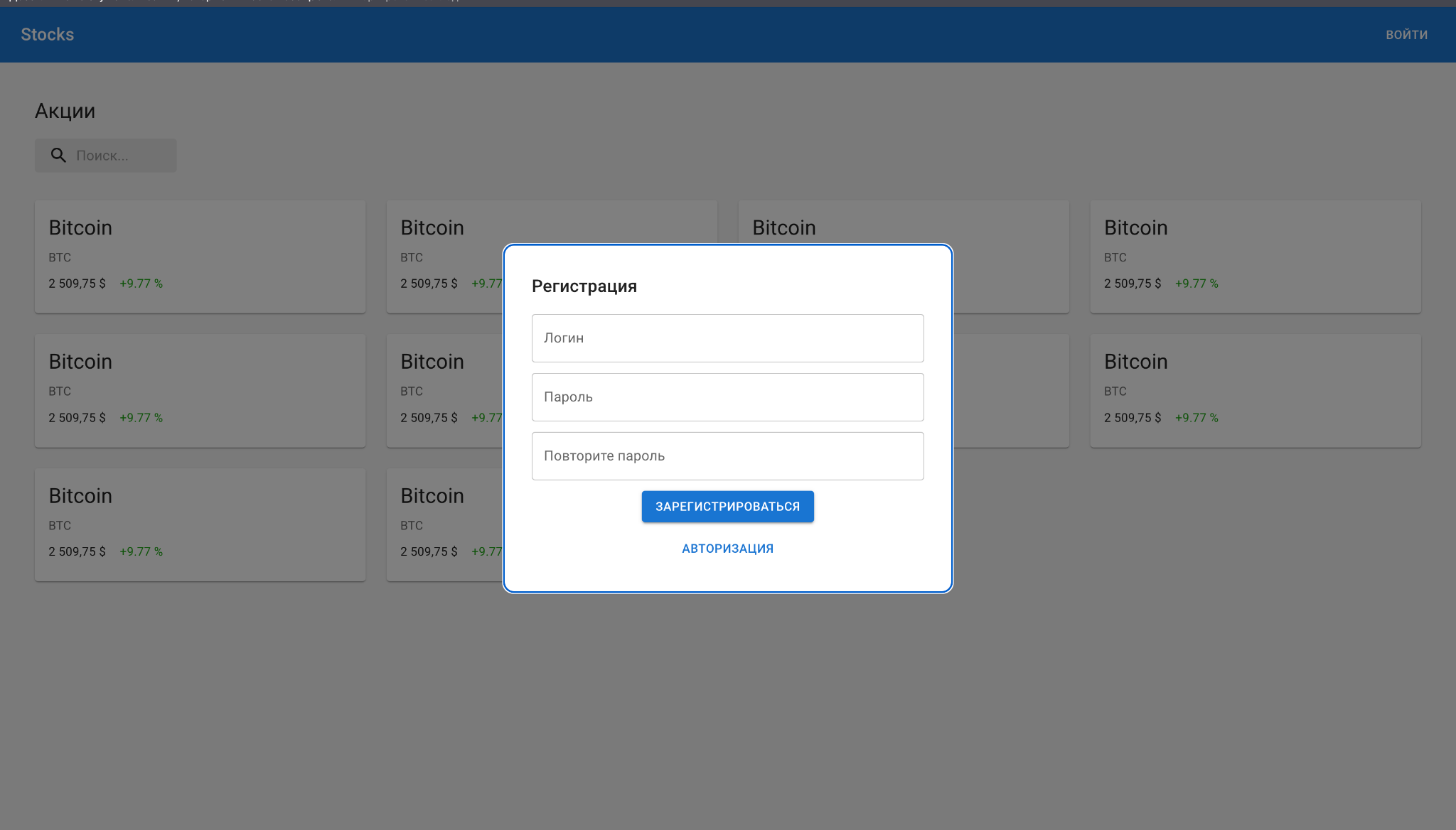
****

Рисунок 23 - Окно регистрации

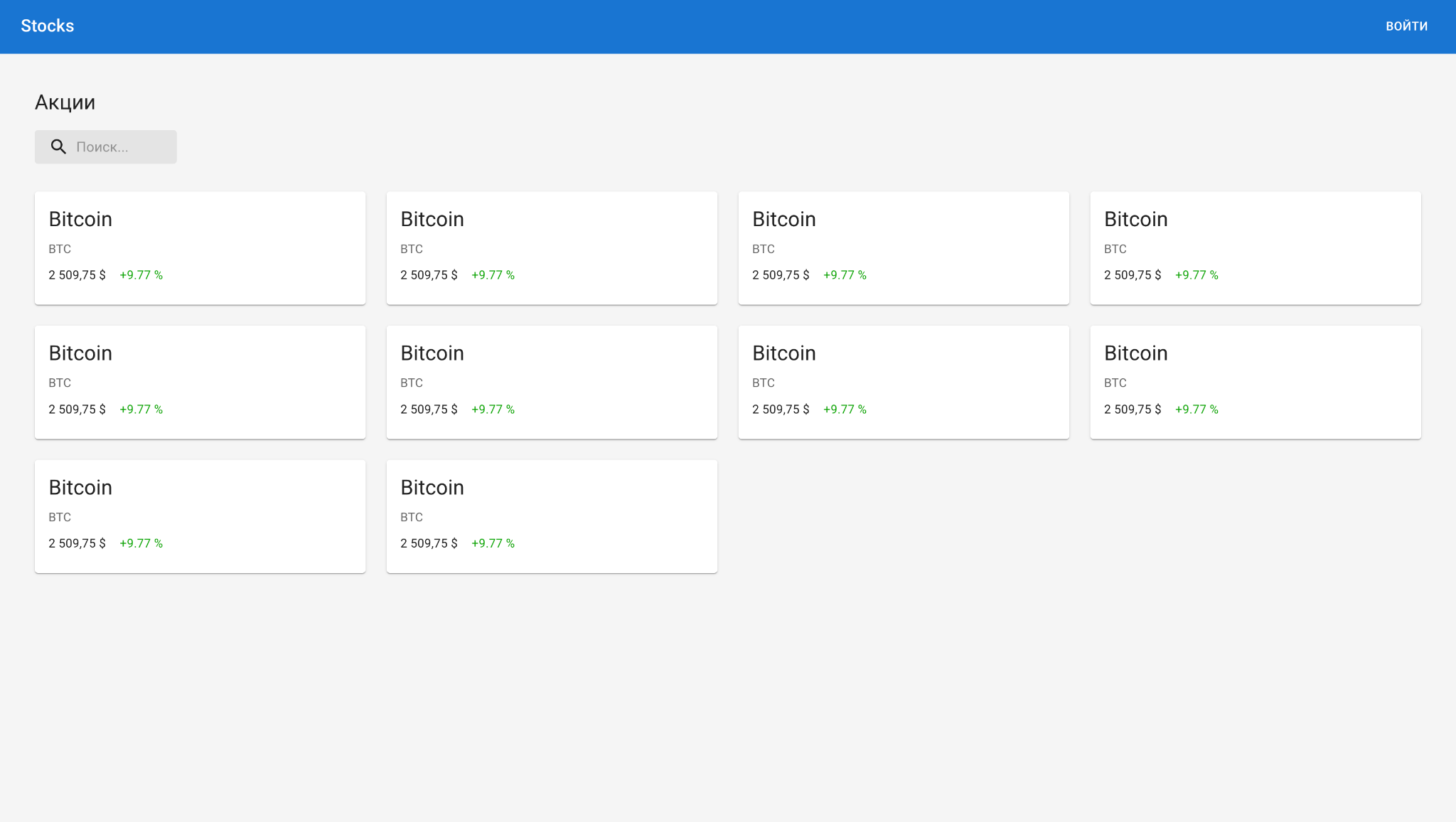


Рисунок 24 - Главная страница неавторизованного пользователя

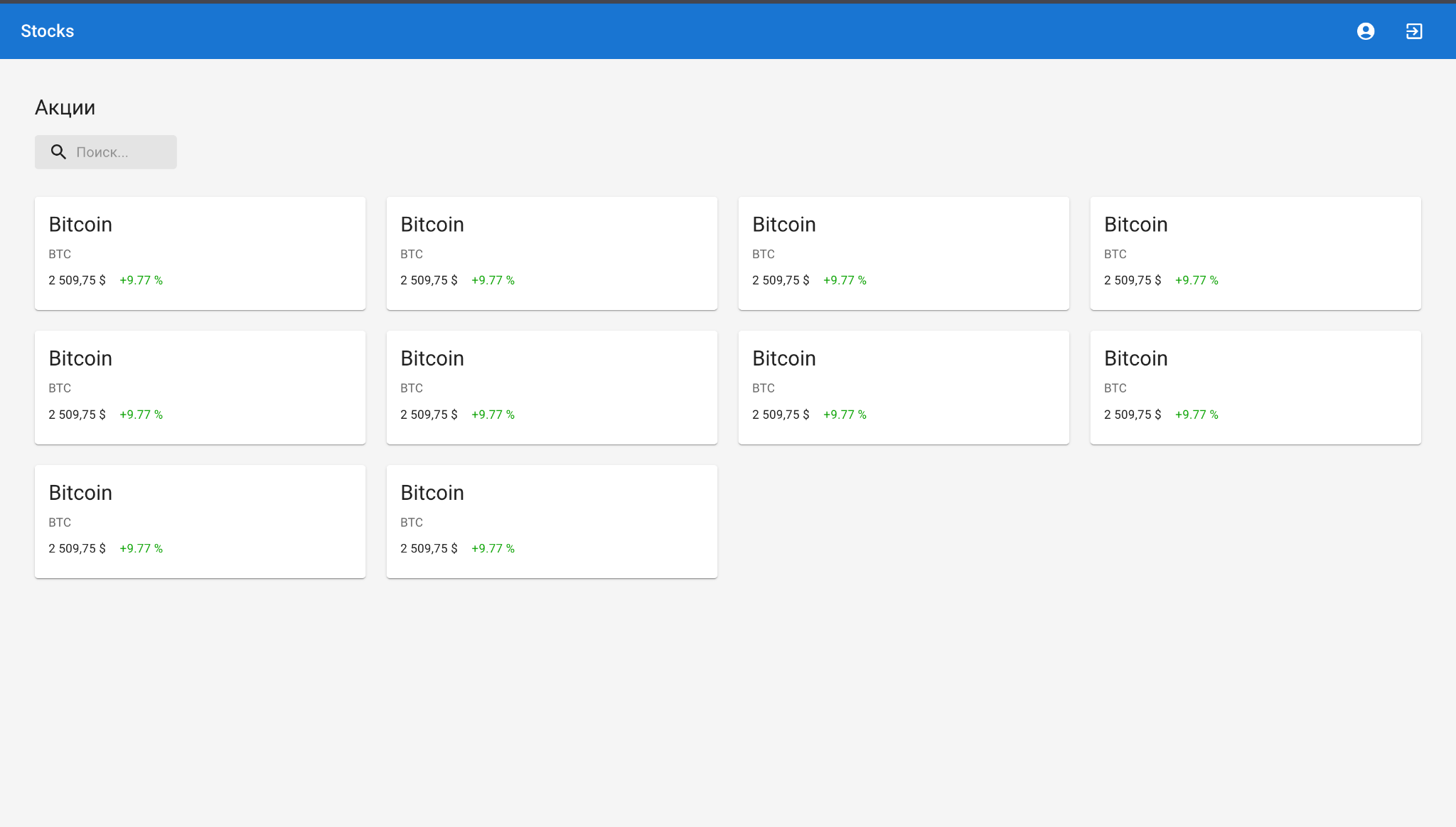


Рисунок 25 - Главная страница авторизованного пользователя

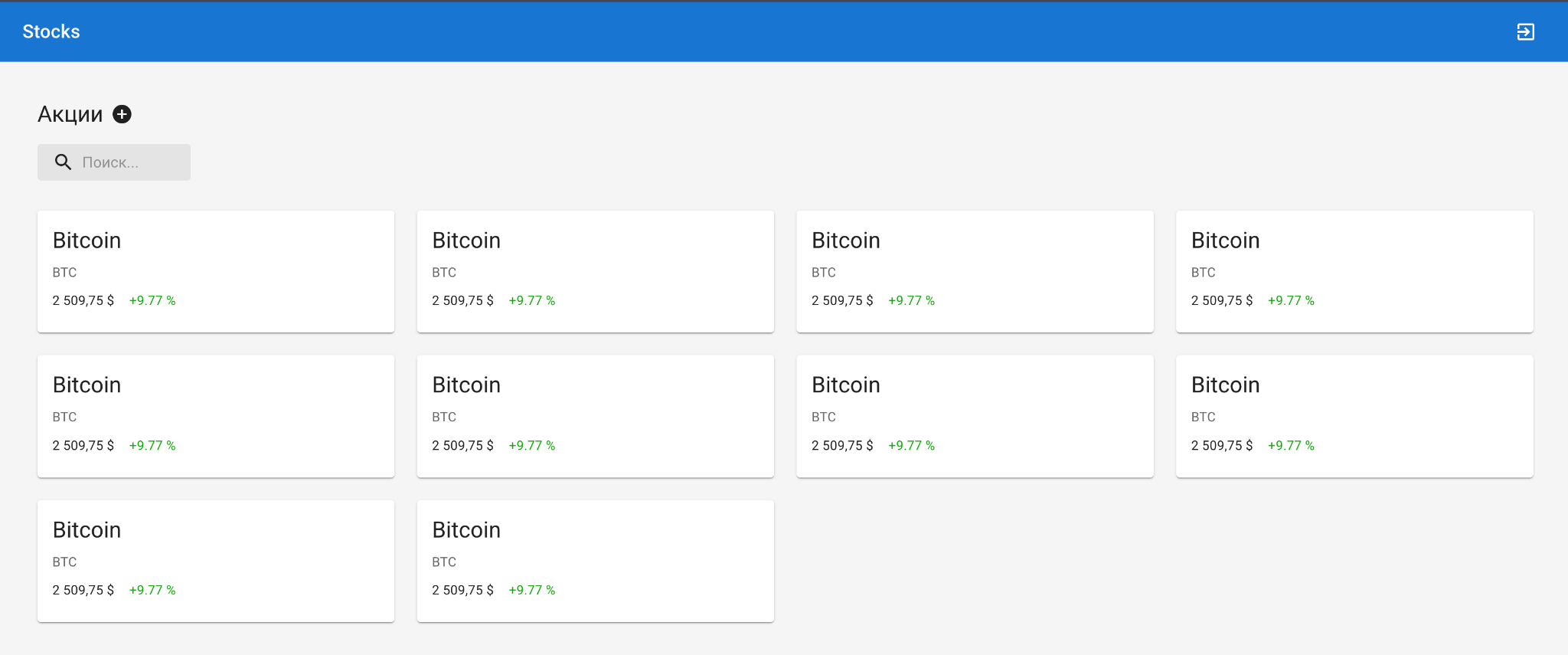


Рисунок 26 - Главная страница менеджера

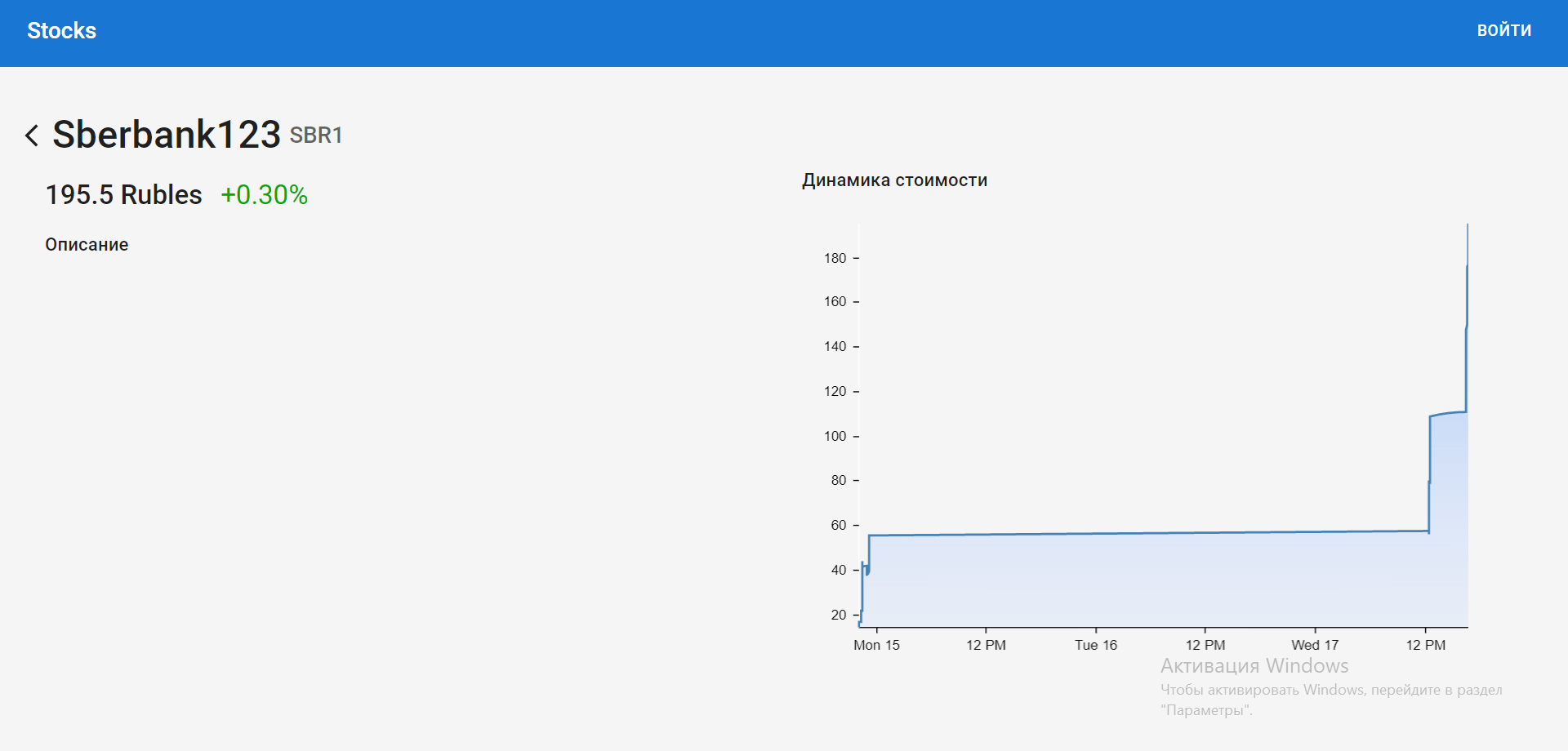


Рисунок 27 - Страница акции для неавторизованного пользователя

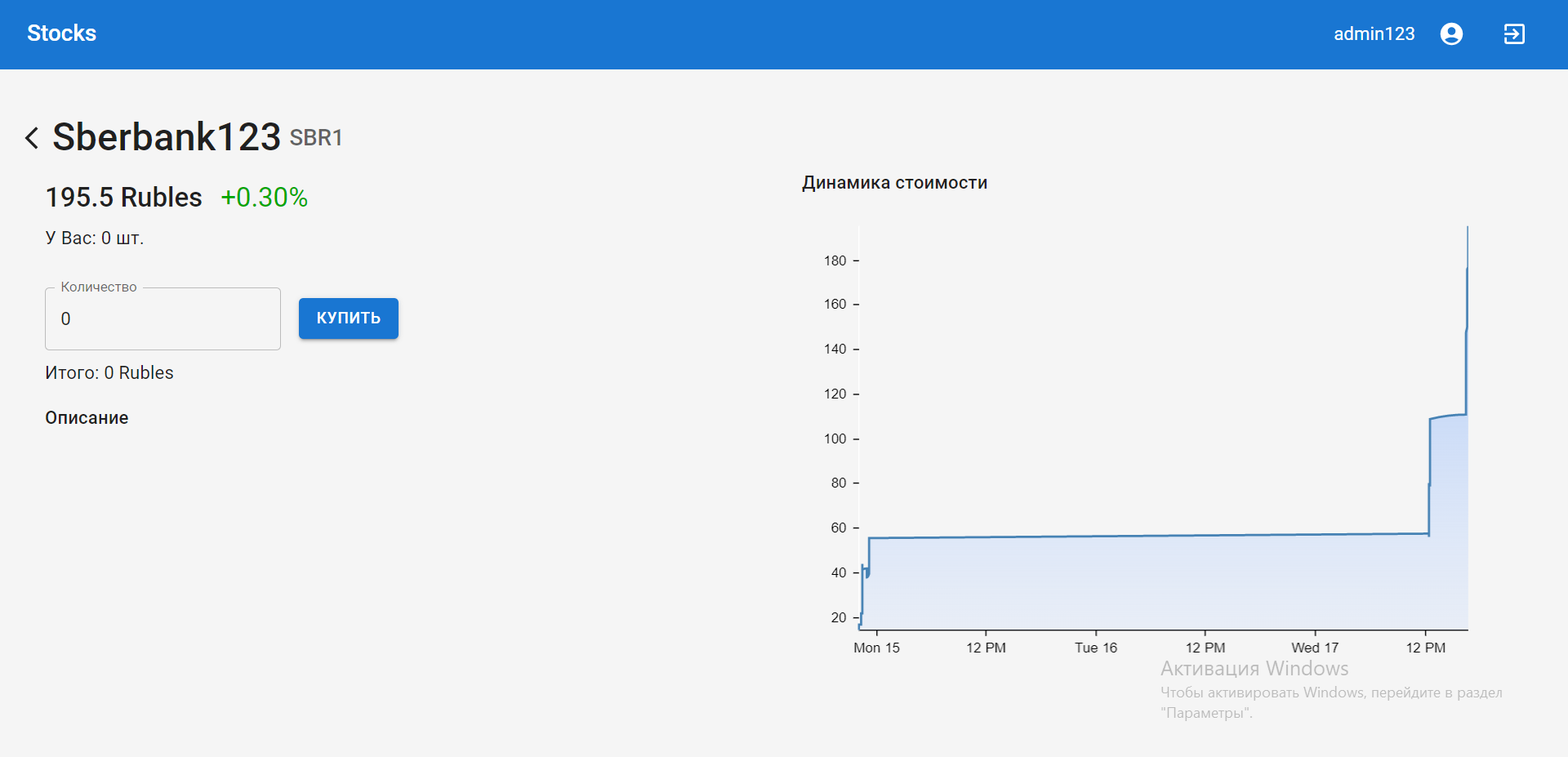


Рисунок 28 - Страница акции для авторизованного пользователя

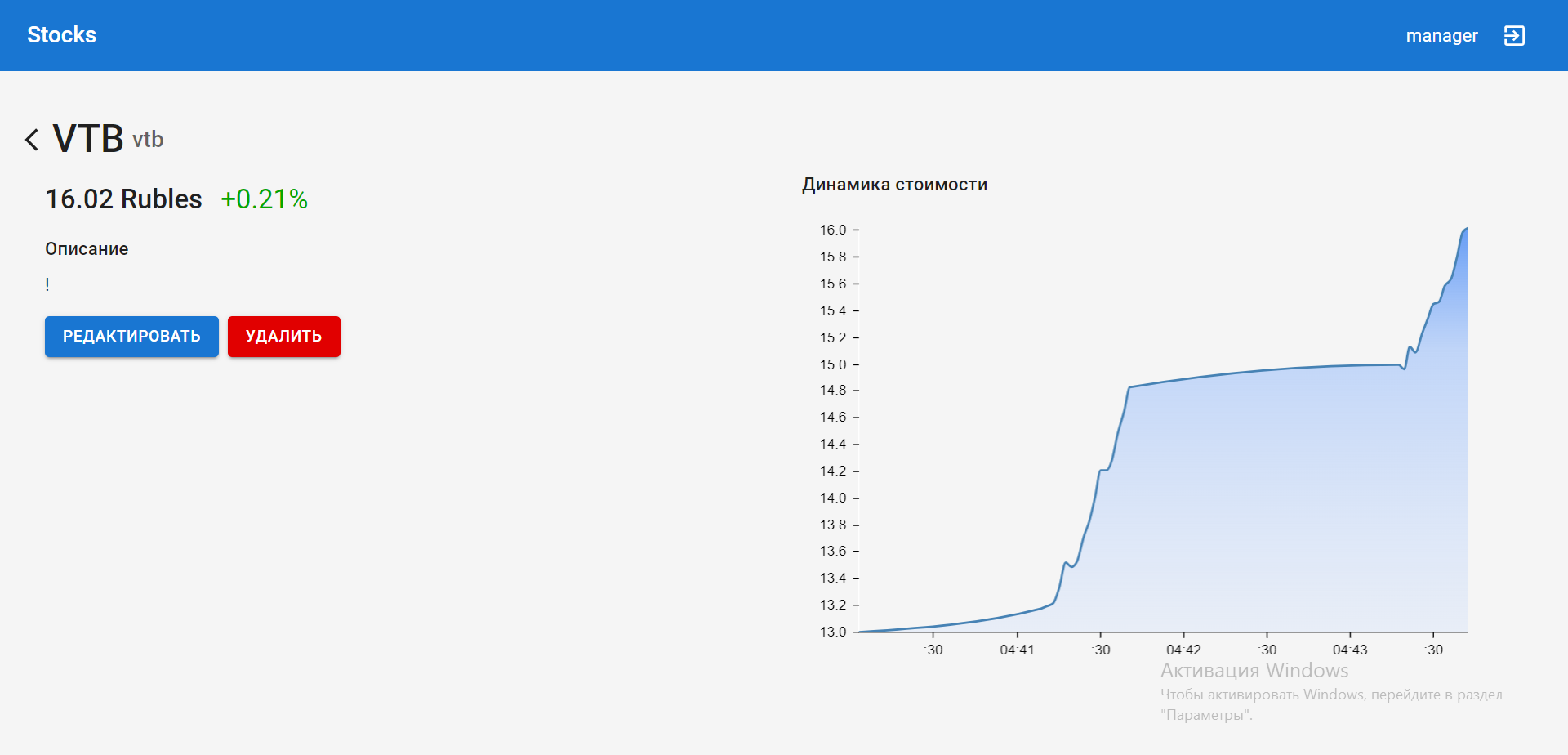


Рисунок 29 - Страница акции для менеджера

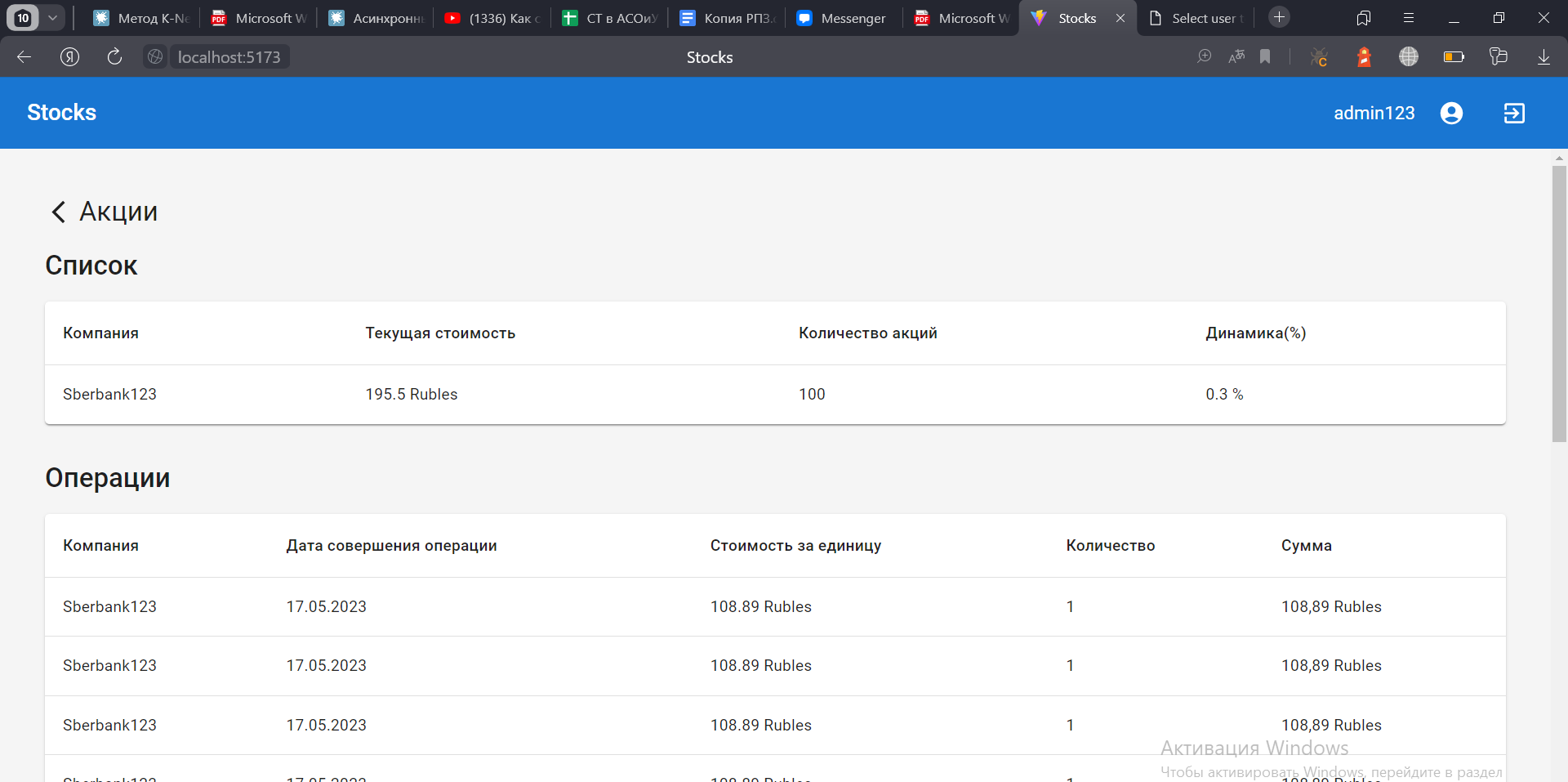


Рисунок 30 - Активы и транзакции пользователя

## Websocket:

Для тестирования Websocket необходимо запустить сервис Websocket. Отправлять запросы при помощи Postman на ip сервера Websocket на порт 9000.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Пункт ТЗ | Выполняемое действие | Результат | Иллюстрации |
| 1 | 5.1.3.1 | Передача котировок:  Подключиться к сервису по ws, дождаться получения котировки. | Получение котировки акции. | Рисунок 30 |
| 2 | 5.1.3.2 | Отправка котировок по gRPC: Необходимо отправить POST запрос, в теле запроса указать массив данных с новыми котировками. | В ответ вернётся количество переданных котировок. | Рисунок 31 |

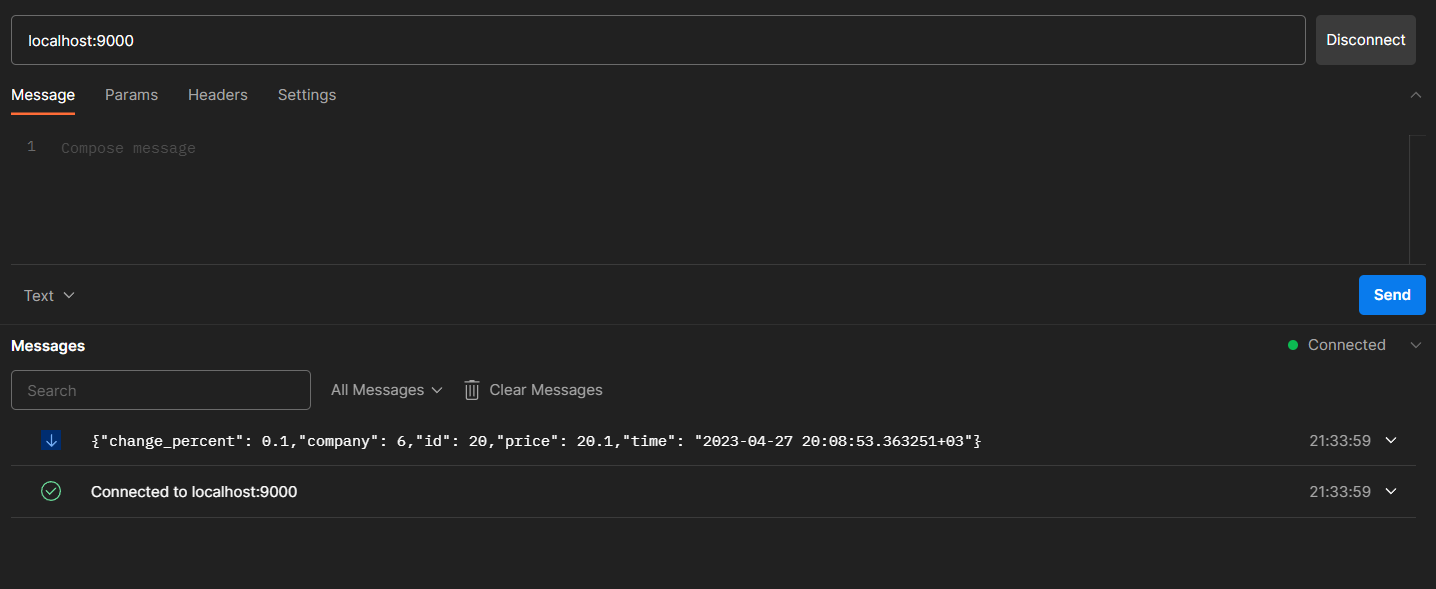


Рисунок 31 - Отправка котировок по Websocket

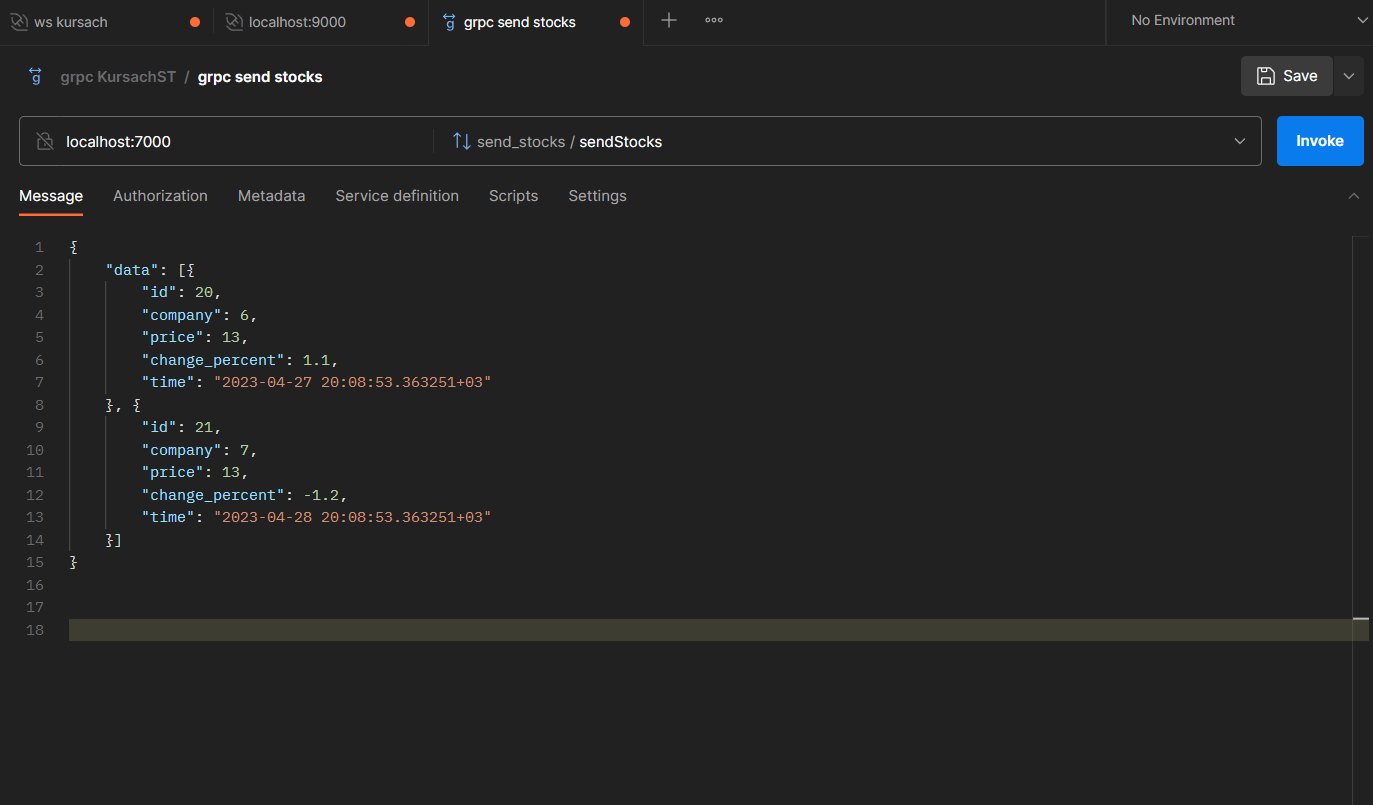


Рисунок 32 - Получение новых котировок по gRPC

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Н.Э. Баумана

Кафедра «Системы обработки информации и управления»

Утверждаю

\_\_\_\_\_\_\_\_\_\_\_\_\_Галкин В.А.

“\_\_\_” \_\_\_\_\_\_\_2023 г.

**Трейдерское приложение**

Руководство пользователя

(вид документа)

писчая бумага

(вид носителя)

9

(количество листов)

Исполнители:

Гурова М.Д.

Ищенко А.С.

Юриков Е.В.

Москва – 2023

# Введение

## Область применения

## Требования настоящего документа применяются при:

* предварительных комплексных испытаниях;
* опытной эксплуатации;
* приемочных испытаниях;
* промышленной эксплуатации.

## Краткое описание возможностей

Распределённая система “Трейдерское приложение”, представляющая собой веб-приложение для отслеживания цен акций в реальном времени и трейдинга.

РС “Трейдерское приложение” предоставляет возможность доступа к информации об акциях компаний без авторизации. При наличии авторизации, пользователь получает доступ к покупке/продаже ценных бумаг, а также к статистике аккаунта (содержимое его инвестиционного портфеля, история транзакций).

РС “Трейдерское приложение” предоставляет возможность авторизации в роли менеджера. При авторизации данным способом, пользователю становится доступным дополнительный функционал: добавление акции, изменение ее описания и ее удаление.

Актуальность цен акций поддерживается при помощи протокола WebSocket, который позволяет обновлять стоимость ценных бумаг в реальном времени.

# Назначение и условия применения РС “Трейдерское приложение”

## Назначение РС “Трейдерское приложение”

РС “Трейдерское приложение” предназначена для автоматизации информирования пользователем об изменении стоимости акций, процессов их купли-продажи, добавления/удаления новых ценных бумаг, а также своевременной корректировки информации о них.

Для использования РС “Трейдерское приложение” необходимо выполнение следующих условий:

## Системные требования

Для работы клиента необходим Yandex Browser.

1. **Условия выполнения программы**

Для работы программы требуется браузер Yandex Browser или любой иной поддерживающий современные функции JavaScript, а также стабильное интернет-соединение.

# Выполнение программы

* 1. Инсталляция/деинсталляция
     1. Потребуется Web-браузер, рекомендуется, Yandex Browser
  2. Запуск программы
     1. На компьютере необходимо запустить браузер и ввести в адресную строкуhttp://localhost:3000.

# Описание операций

## Просмотр информации о ценных бумагах

Доступно для: все пользователи.

Операция 1: Открыть веб страницу

Для просмотра краткой информации о ценных бумагах необходимо открыть приложение (рис. 1).

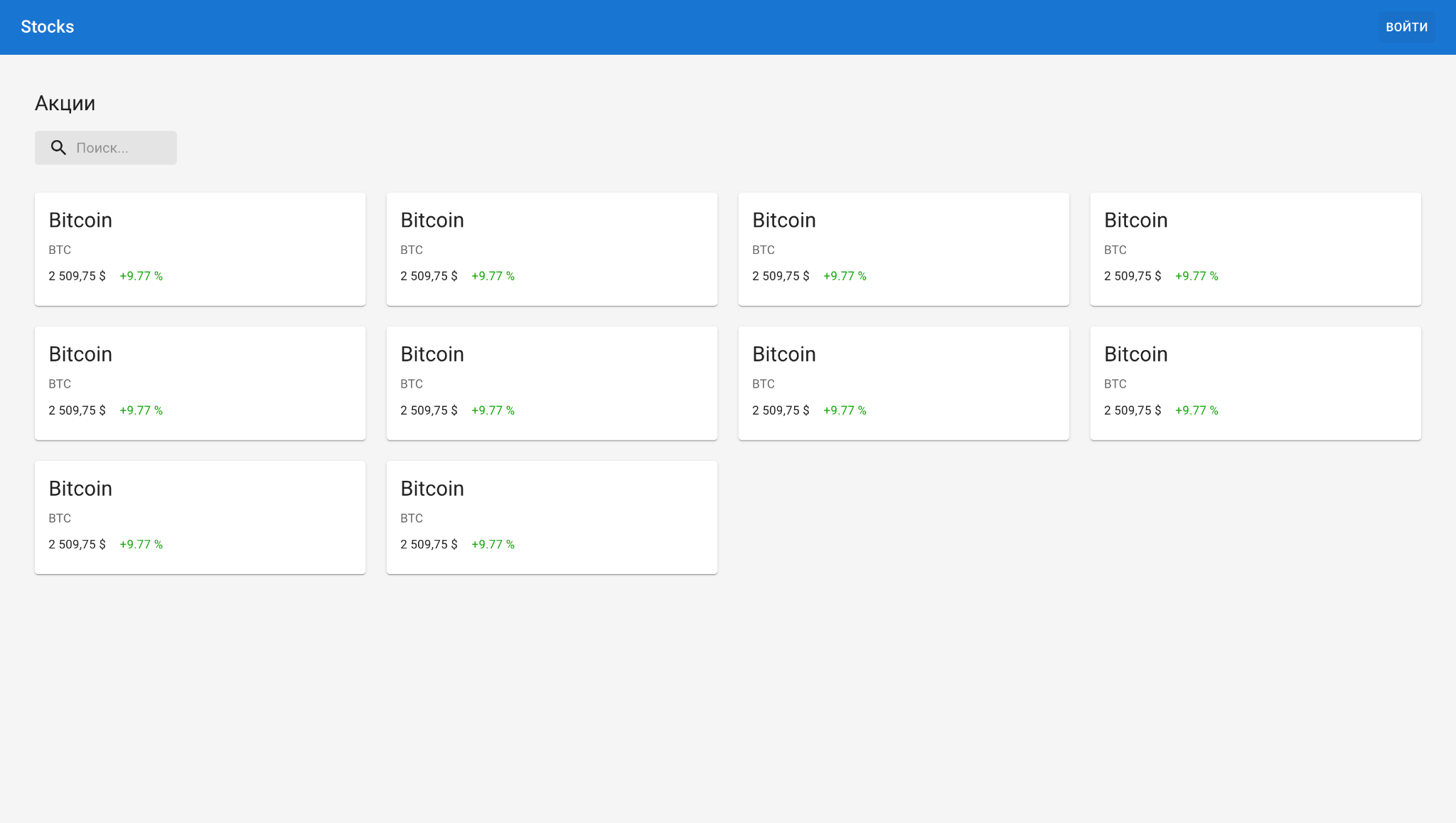


Рисунок 33 - Главный экран неавторизованного пользователя

Операция 2: Просмотр подробной информации по акции

Для просмотра подробной информации по акции необходимо на главном экране нажать на карточку с интересующей вас акцией (рис. 32).

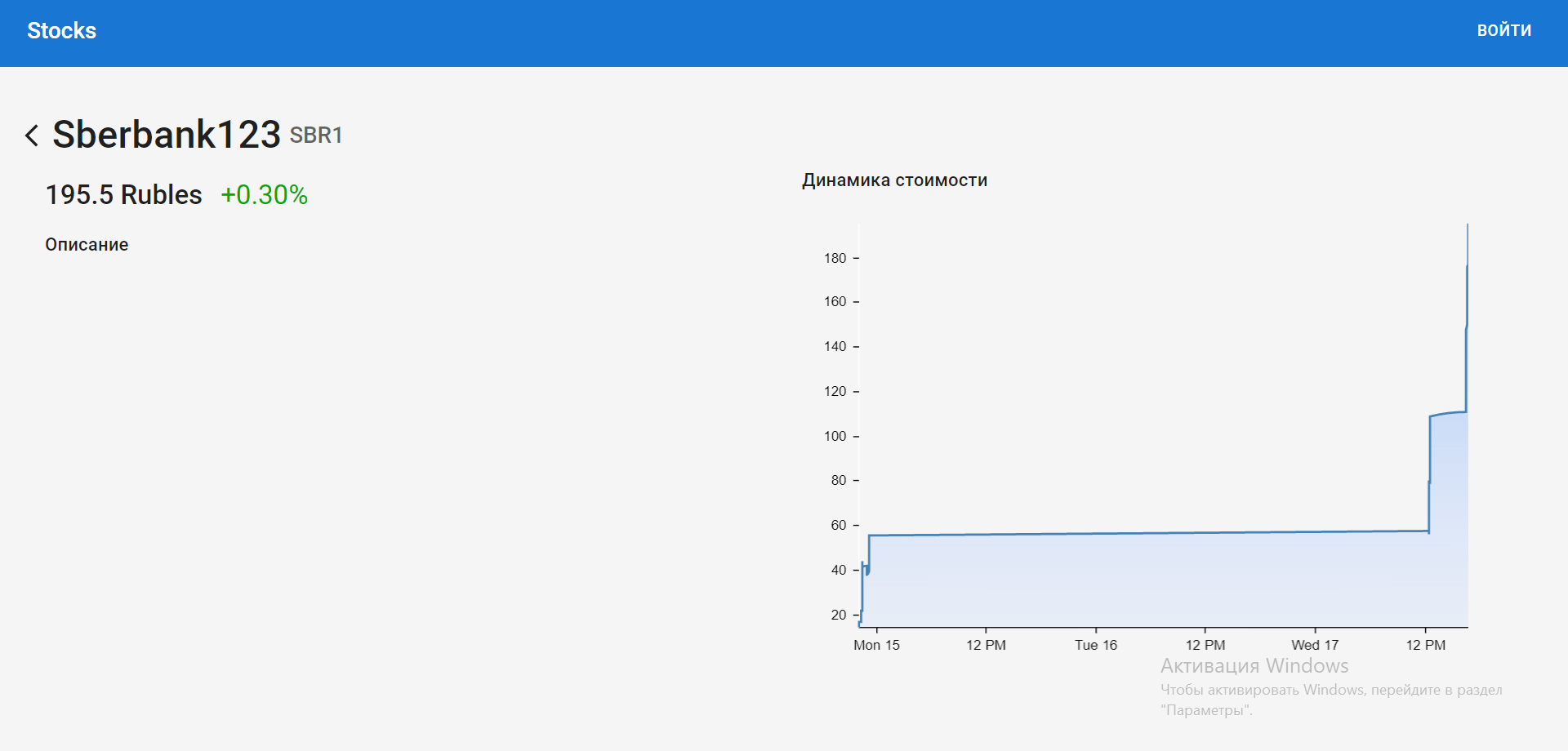


Рисунок 34 - Страница акции неавторизованного пользователя

## Выход из аккаунта

Доступно для: все пользователи.

Для выхода из аккаунта необходимо нажать на кнопку выхода в правом верхнем углу меню.

## Поиск ценных бумаг

Доступно для: все пользователи.

Операция 1: для поиска ценных бумаг по названию, воспользуйтесь полем “Поиск” на главном экране (рис.32, рис.35).

Операция 2: ввести название акции и нажать клавишу “Enter”.

## Регистрация

Доступно для: неавторизованные пользователи.

Операция 1: для регистрации в РС “Трейдерское приложение” необходимо в главном меню (рис. 32) нажать на кнопку “Войти” на верхнем меню.

Операция 2: после выполнения шага 1 откроется окно авторизации. Для регистрации необходимо нажать на соответствующую кнопку (рис. 33).

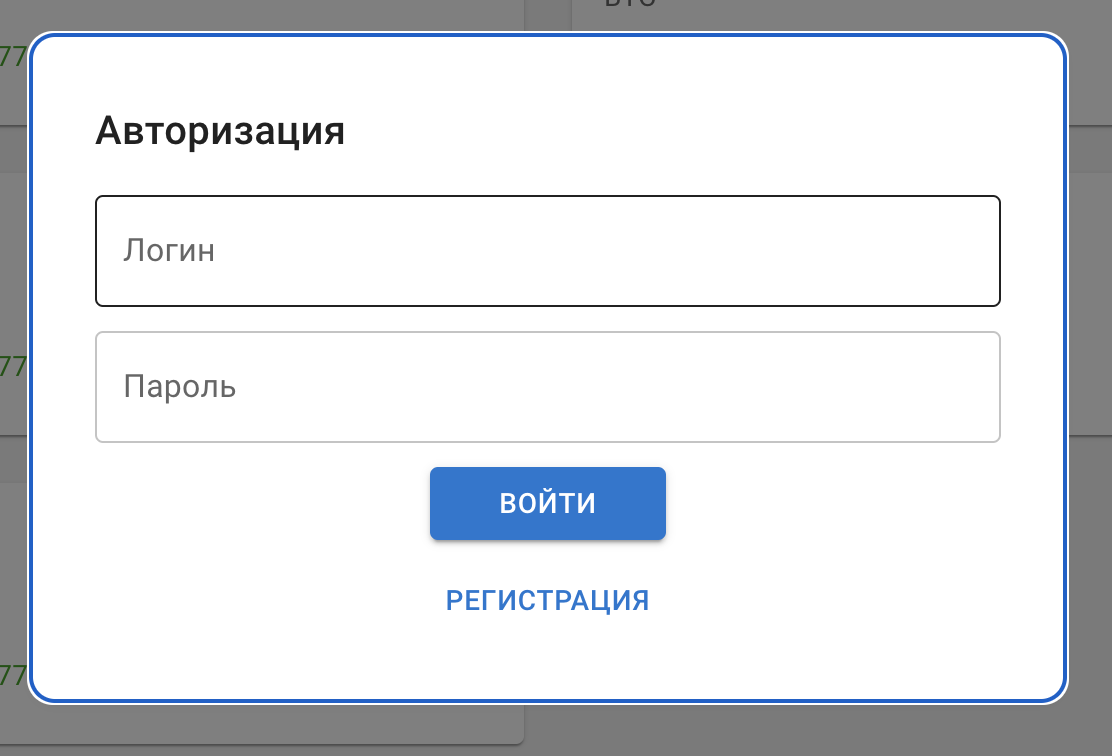


Рисунок 35 - Окно авторизации

Операция 3: в окне регистрации необходимо ввести логин и пароль (продублировать пароль необходимо в двух полях), которые будут использованы в дальнейшем для авторизации (рис. 34).

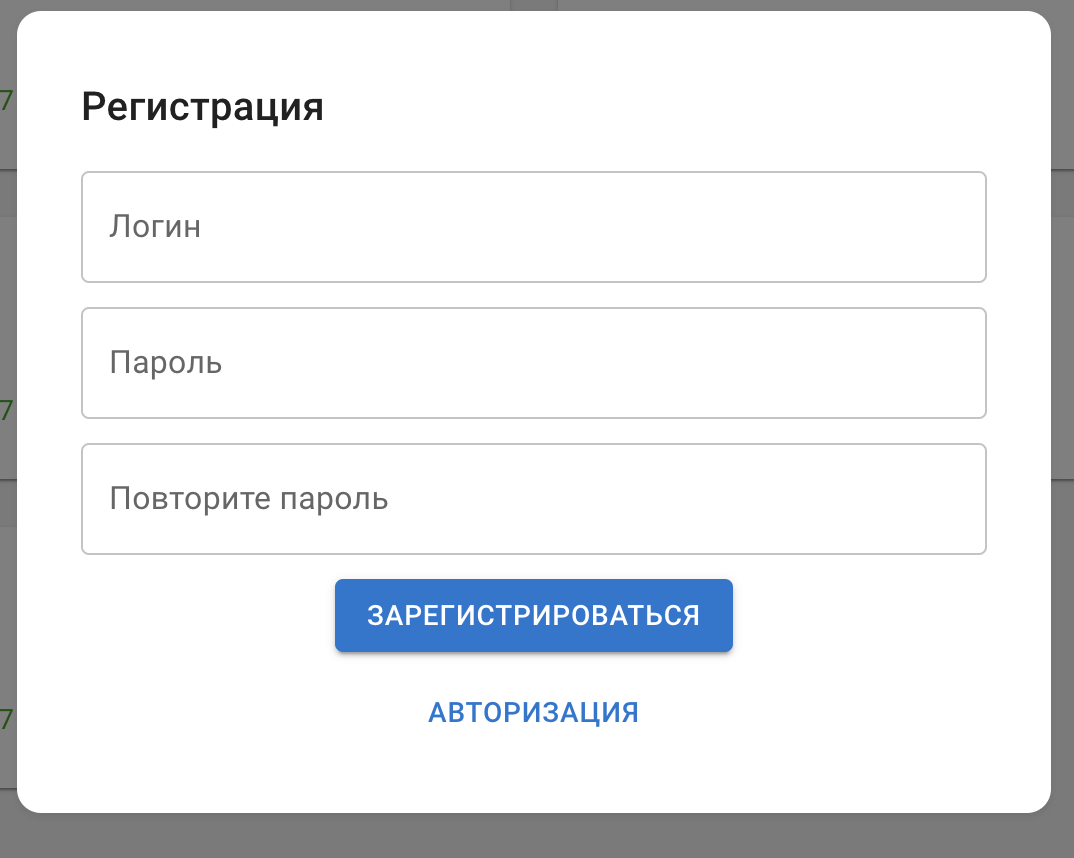


Рисунок 36 - Окно регистрации

Операция 4: после успешной регистрации будет выполнен вход в аккаунт и открыто главное меню. В правом верхнем углу отображается иконка для перехода к транзакциям и активам аккаунта (рис. 35).

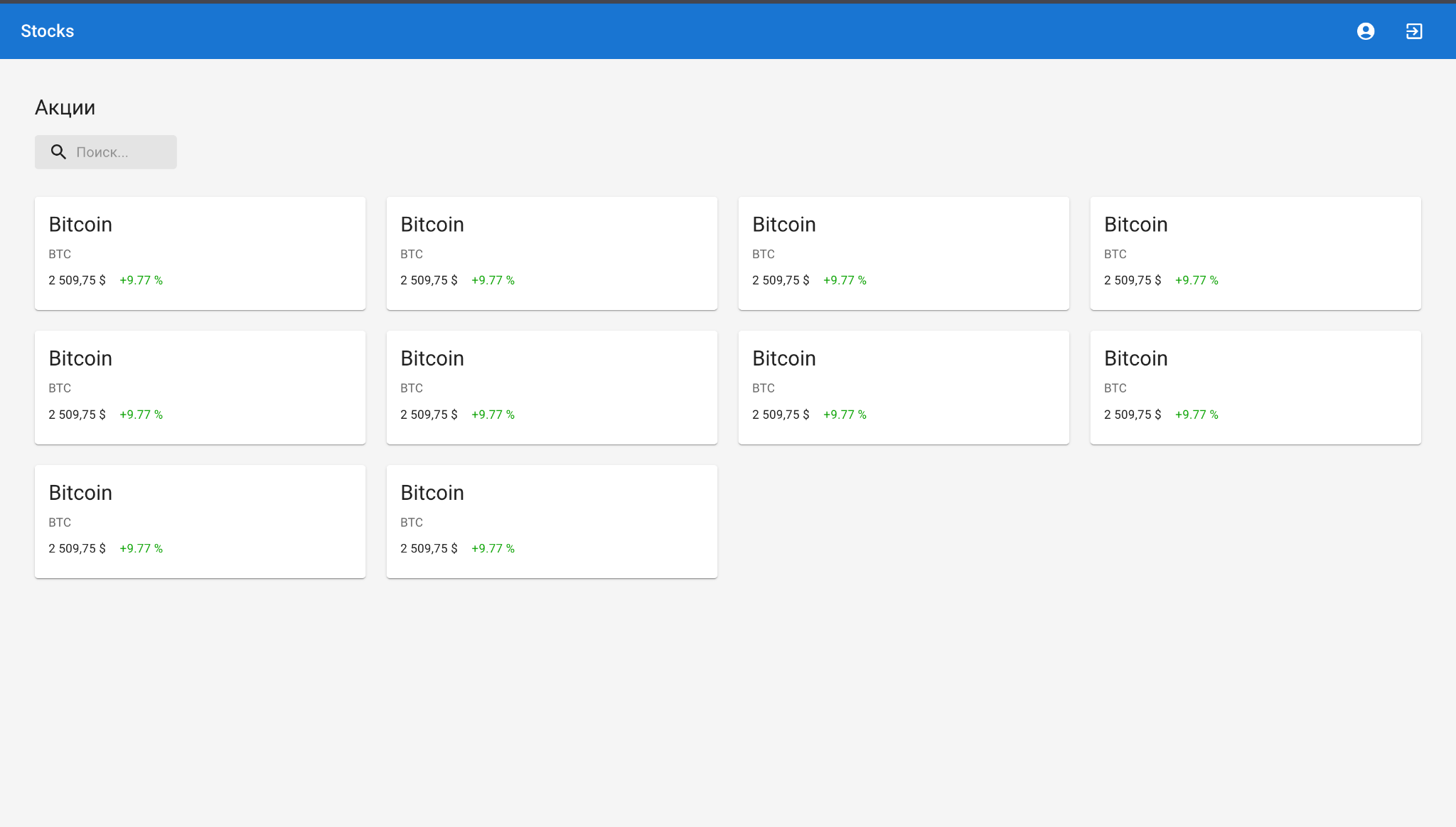


Рисунок 37 - Главное меню авторизованного пользователя

## Просмотр транзакций и активов

Доступно для: авторизованные пользователи без прав менеджера.

Операция 1: для просмотра произведенных транзакций и инвестиционного портфеля необходимо нажать на иконку аккаунта в правом верхнем углу (рис. 36).

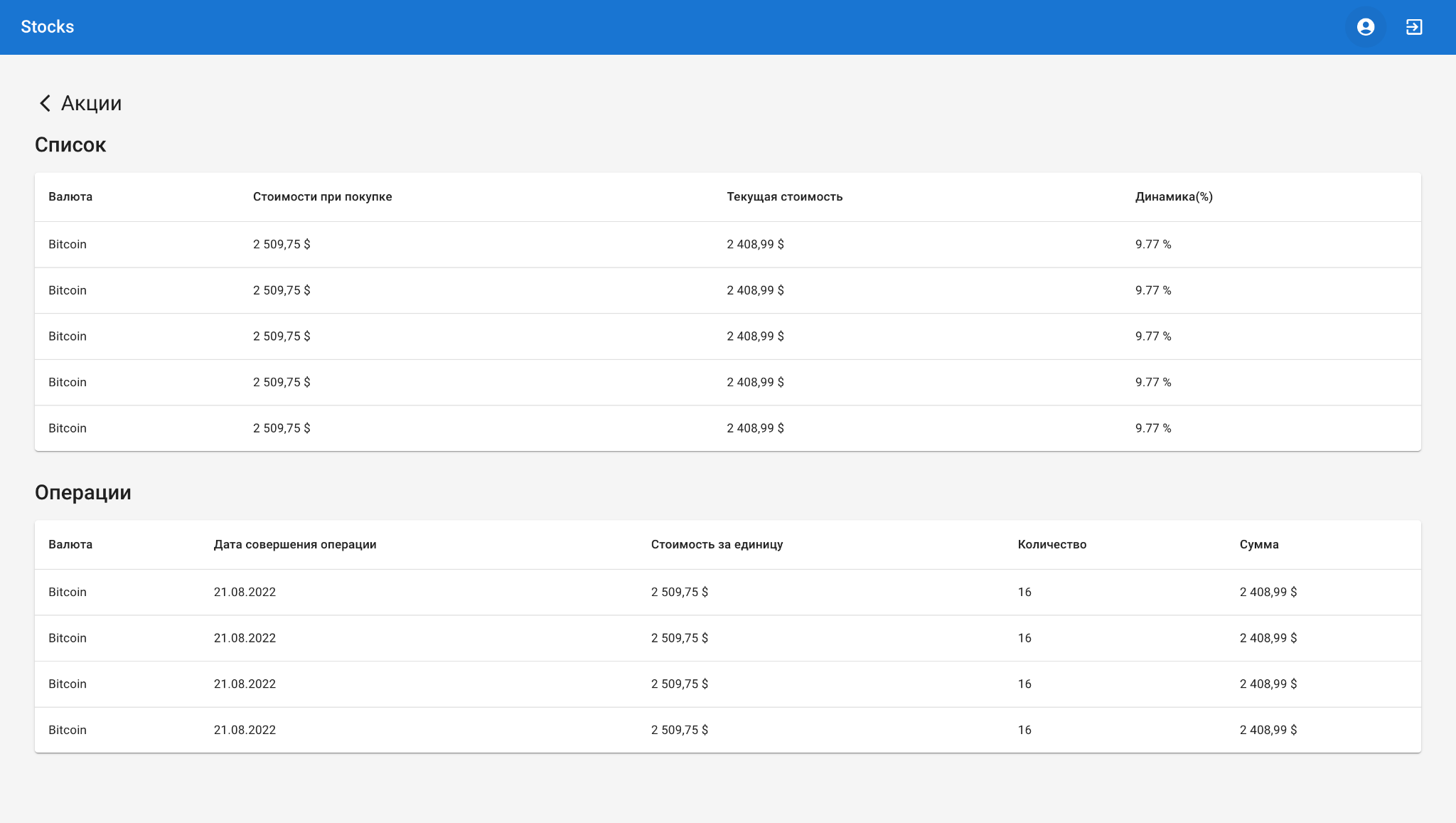


Рисунок 38 - Данные об операциях и купленных ценных бумагах пользователя

## Покупка/продажа ценных бумаг

Доступно для: авторизованные пользователи без прав менеджера.

Для покупки/продажи акции необходимо перейти на ее страницу через главное меню (пункт 4.1.), ввести количество акций для покупки и нажать на соответствующую кнопку (рис. 37).

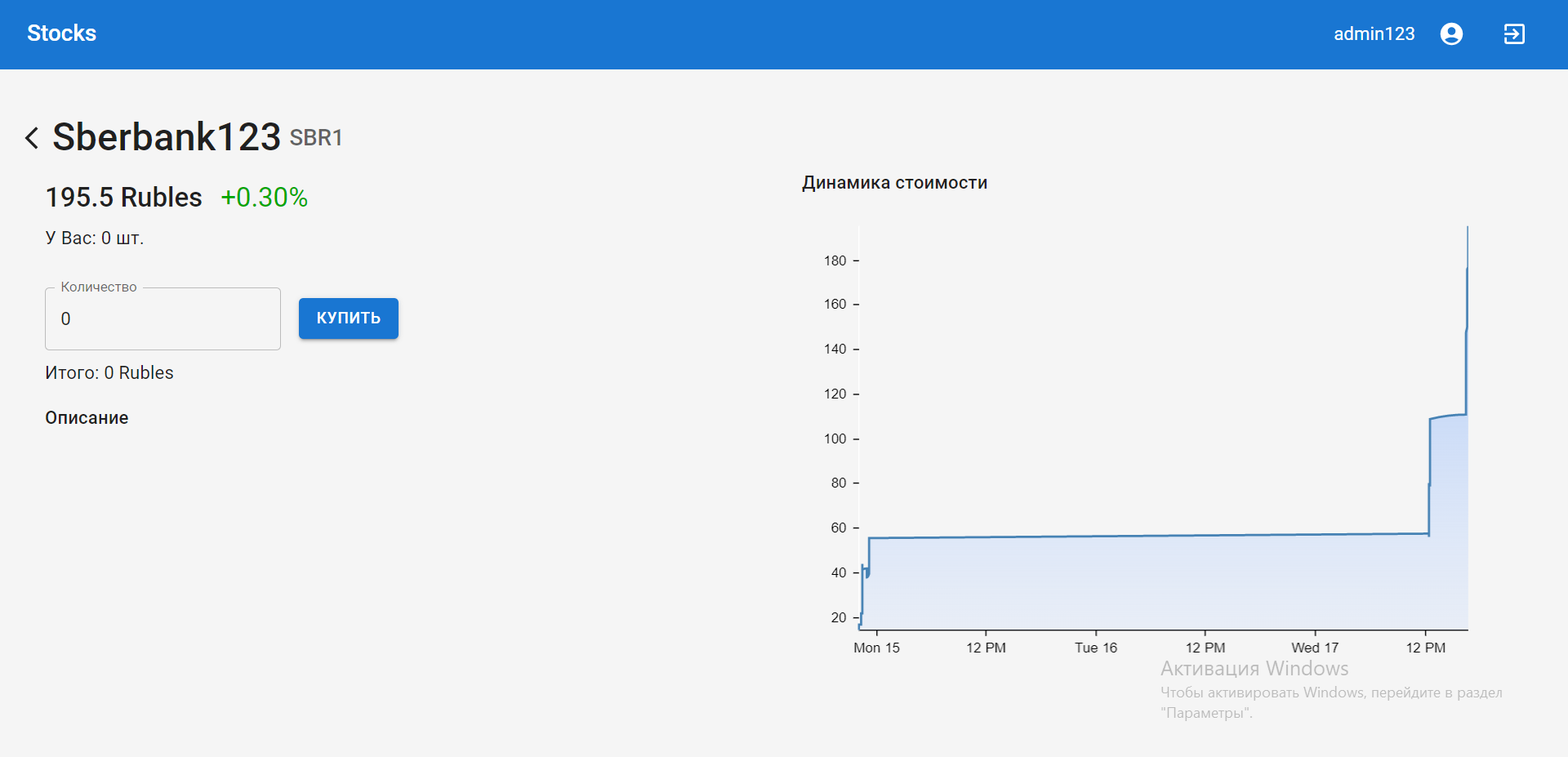


Рисунок 39 - Страница акции для авторизованного пользователя

## Добавление новой ценной бумаги

Доступно для: пользователи, авторизованные в роли менеджера.

Операция 1: для добавления новой акции необходимо авторизоваться как менеджер (рис. 40).

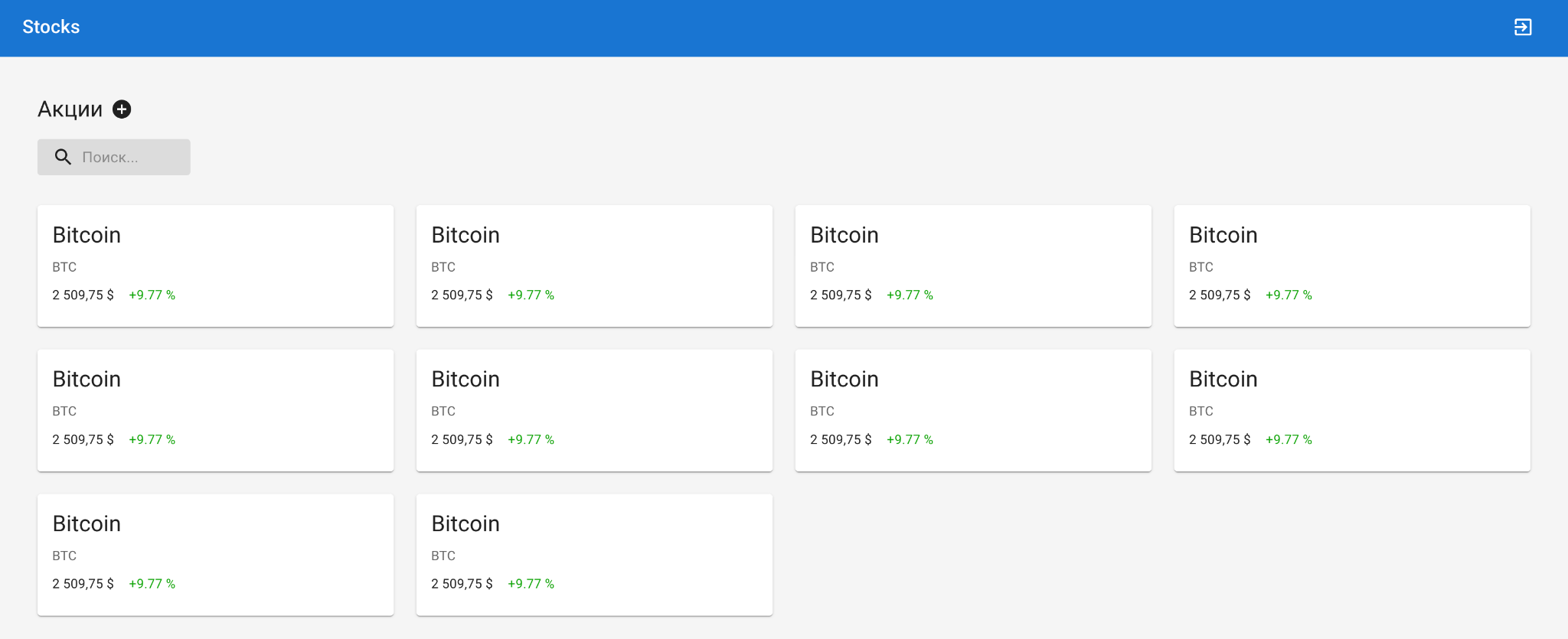
****

Рисунок 40 - Главное меню менеджера

Операция 2: после авторизации в главном меню справа от заголовка “Акции” появится кнопка “+” (рис. 40), при нажатии на нее откроется форма добавления новой акции. Для создания акции необходимо заполнить данные и нажать на соответствующую кнопку (рис. 41).

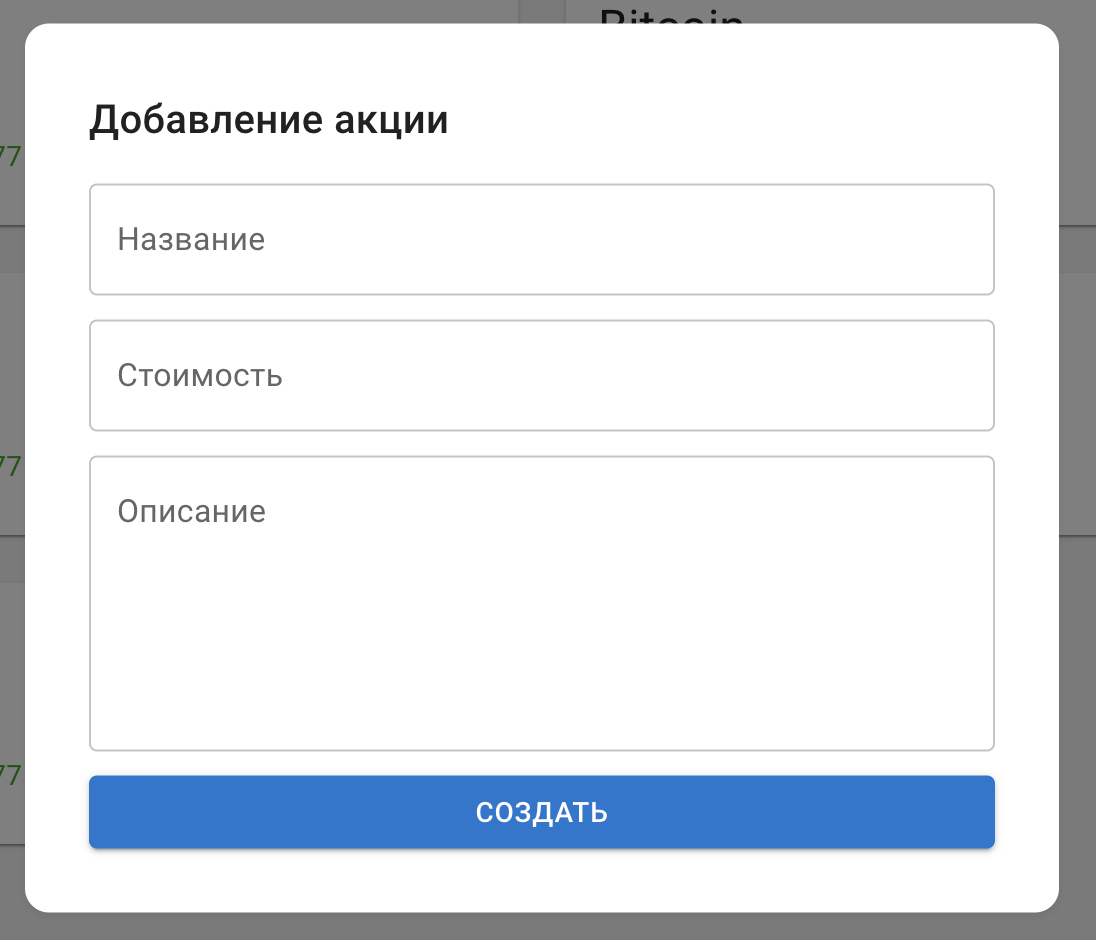


Рисунок 41 - Создание акции

## Редактирование ценной бумаги

Доступно для: пользователи, авторизованные в роли менеджера.

Для редактирования акции необходимо через главное меню перейти на страницу акции (пункт 4.1.) и нажать на кнопку “Редактировать”.

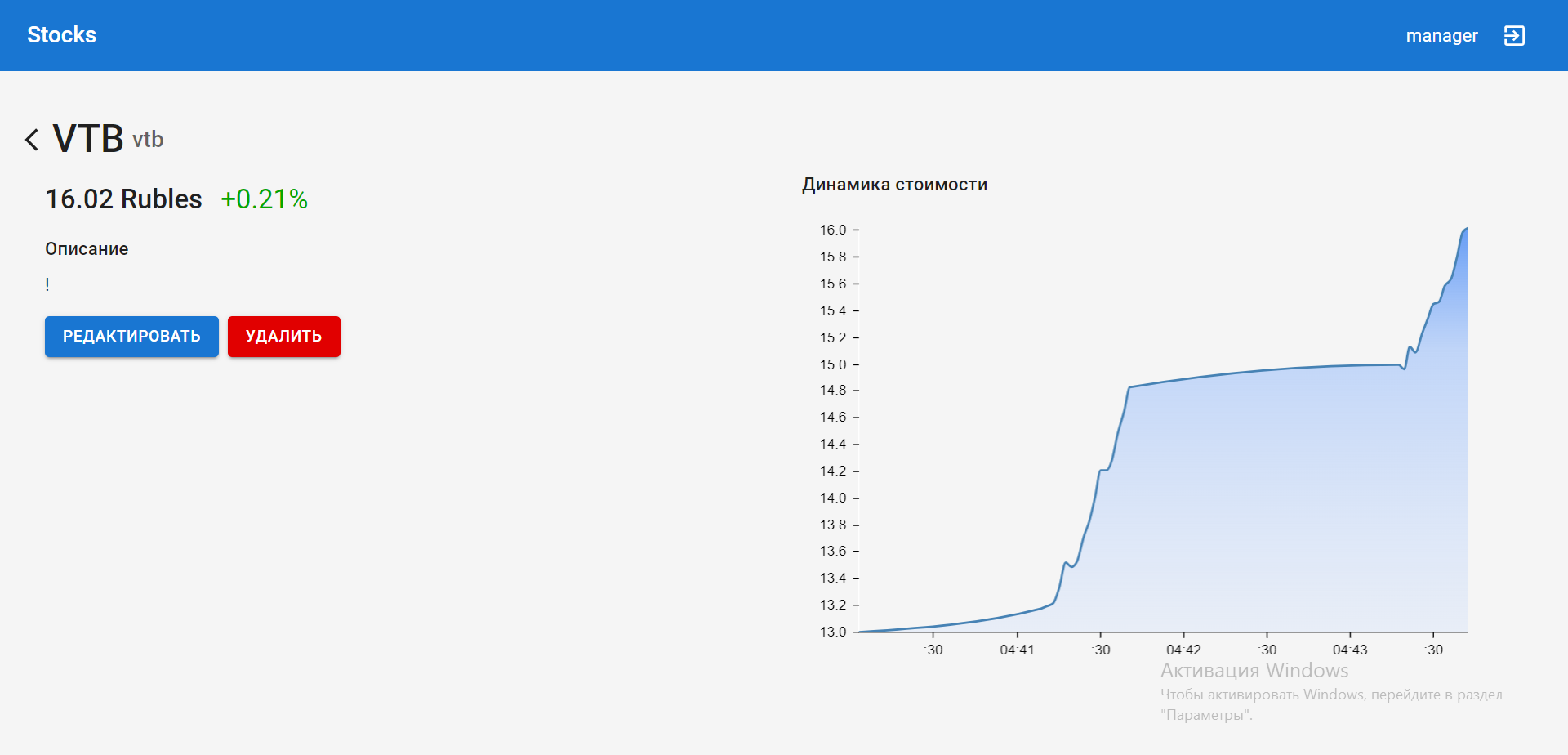


Рисунок 42 - Страница акции для менеджера

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Н.Э. Баумана

Кафедра «Системы обработки информации и управления»

Утверждаю

\_\_\_\_\_\_\_\_\_\_\_\_\_Галкин В.А.

“\_\_\_” \_\_\_\_\_\_\_2023 г.

**Трейдерское приложение**

Руководство системного админстратора

(вид документа)

писчая бумага

(вид носителя)

4

(количество листов)

Исполнители:

Гурова М.Д.

Ищенко А.С.

Юриков Е.В.

Москва – 2023

# Системные требования

Для работы клиента необходим Yandex Browser.

### Требования к ОС

* Windows

Windows 7, Windows 8, Windows 8.1, Windows 10 или более поздней версии.

Примечание: также поддерживаются следующие серверные ОС: Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2 или Windows Server 2016.

* MacOS

macOS High Sierra 10.13 или более поздней версии

* Linux

Ubuntu 18.04 (64-разрядная версия) или более поздней версии, Debian 10 или более поздней версии, openSUSE 15.2 или более поздней версии, Fedora Linux 32 или более поздней версии.

**Требования к оборудованию**

Центральный процессор: с частотой от 1 ГГц

Оперативная память: 8 Гб и выше

Пространство на жестком диске: 1 Гб

Платформа: Microsoft Windows, Mac OS, Linux x32/x64

### Требования к ПО

1. Сервер СУБД с установленным pgAdmin версии 6.19 и выше;
2. Сервер для запуска микросервисов с установленным Docker desktop версии 4.17.1.

# Порядок развёртывания серверов:

## Сервер СУБД

Развернуть СУБД PostgreSQL, создать базу данных kursach\_db, пользователя kursach\_dbuser с паролем asdf, порт PostgreSQL 5432.

## Сервис бэкенда

1. Выполнить команду   
   docker pull criptycpizza7/kursach\_stback;
2. Выполнить команду   
   docker run -it criptycpizza7/kursach\_stback sh;
3. Последовательно выполнить следующие команды:

* cd /usr/app/src/kursach\_back;
* python manage.py migrate;
* python manage.py runserver 0.0.0.0:8000.

После этого на порте 8000 будет доступен сервис бэкенда.

## Сервис Websocket

1. Выполнить команду docker pull criptycpizza7/kursach\_socket;
2. Выполнить команду docker run -it criptycpizza7/kursach\_socket sh;
3. Последовательно выполнить следующие команды:
   * cd /usr/app/src/kursach\_socket;
   * python websocket.py.

После этого на порте 9000 будет доступен сервис Websocket.

## Сервис фронтенда

* 1. Выполнить команду   
     docker pull criptycpizza7/kursach\_stfront;
  2. Выполнить команду   
     docker run -it criptycpizza7/kursach\_stfront sh
  3. Последовательно выполнить следующие команды:
* cd /usr/app/src/kursach\_front;
* yarn start.

После этого на порте 3000 будет доступен сервис фронтенда.

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Н.Э. Баумана

Кафедра «Системы обработки информации и управления»

Утверждаю

\_\_\_\_\_\_\_\_\_\_\_\_\_Галкин В.А.

“\_\_\_” \_\_\_\_\_\_\_2023 г.

**Трейдерское приложение**

Диаграмма деятельности

(вид документа)

писчая бумага

(вид носителя)

2

(количество листов)

Исполнители:

Гурова М.Д.

Ищенко А.С.

Юриков Е.В.

Москва – 2023

