

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГУ»)

Факультет компьютерных наук  
Кафедра программирования и информационных технологий

Веб приложение  
«Система для проведения онлайн-аукционов произведений искусства  
LUXELOOT»

Курсовой проект  
09.03.04 Программная инженерия  
Профиль «Информационные системы и сетевые технологии»

Зав. кафедрой \_\_\_\_\_ С.Д. Махортов д. ф.-м. н., профессор \_\_ . \_\_.2024

Обучающийся \_\_\_\_\_ П. Е. Пастуханов

Обучающийся \_\_\_\_\_ А. Е. Рыков

Обучающаяся \_\_\_\_\_ Е. М. Клышникова

Руководитель \_\_\_\_\_ В.С. Тарасов ст. преподаватель

Воронеж 2024

## Содержание

Введение .....	3
2 Анализ предметной области .....	5
2.1 Терминология .....	5
2.2 Функциональные требования .....	6
2.2.1 Вход в приложение .....	6
2.2.2 Авторизация и регистрация .....	6
2.3 Пользовательские сценарии .....	7
3 Требования к приложению и программному обеспечению .....	9
3.1 Требования к структуре приложения в целом .....	9
3.1.1 Требования к архитектуре .....	9
3.1.2 Перспективы развития и модернизации приложения .....	11
3.2 Требования к функциям приложения .....	11
3.3 Управление аккаунтом .....	12
Приложение А .....	13
Приложение Б .....	14
Приложение В .....	18

## Введение

Каждый, кто профессионально занимается коллекционированием предметов современного искусства, рано или поздно обращается к практике покупки арт-объектов на аукционе. Участие в аукционе позволяет не только приобрести уникальную вещь, но и почувствовать себя на своем месте, среди таких же ценителей искусства. Сегодня распространена практика приобретения картин современных художников на онлайн-аукционах, что значительно упрощает весь процесс покупки лота – это особенно важно для занятых людей.

Веб приложения, созданные в помощь человеку, значительно упрощают процесс покупки произведений искусства в онлайн.

Целями создания приложения являются:

- расширение клиентской базы;
- повышение конкуренции на ставках;
- увеличение времени пребывания пользователя в аукционе при его проведении на 10%;
- увеличение прибыли от продажи произведений искусства на 5%.

Для достижения поставленных целей, выделяются следующие задачи:

- приложение должно быть реализовано с использованием трехуровневой клиент-серверной архитектуры, где клиентский уровень взаимодействует с серверным уровнем через REST API. В серверной части должен применяться монолитный подход.
- связь между клиентом и сервером должна осуществляться посредством HTTPS протокола.
- приложение должно иметь базу данных для хранения личных данных пользователей.
- должна присутствовать интеграция с платежным сервисом.

Приложение позволяет:

- управлять состоянием лотов и аукционов;
- создавать лоты и подавать их заявку на аукционы;
- создавать аукционы и настраивать время их проведения;
- конкурировать ставками с другими пользователями, участвующими в аукционе;
- проводить предварительную модерацию лотов.

Мобильное приложение реализовано на языке Java, React.js, JavaScript.

## **2 Анализ предметной области**

### **2.1 Терминология**

**Серверная часть** – это программное обеспечение, которое работает на сервере и обеспечивает его функционирование.

**Клиентская часть** – это программное обеспечение, которое работает на устройстве пользователя.

**Администратор** – пользователь, имеющий доступ к расширенному функционалу по модерации базы данных через панель администрации.

**Пользовательские сценарии** – описания типичных ситуаций использования продукта или услуги, которые помогают понять, как пользователи будут взаимодействовать с ним в различных контекстах.

**Клиент-сервер** – современная технология, предполагающая распределение нагрузки и заданий между теми, кто предоставляет услуги, и теми, кто их использует.

**Клиент** – машинное оборудование, в обязанности которого входит отправка запросов на предоставление определенной информации или решение пользовательских задач серверу.

**Сервер** – аппаратное обеспечение, принимающее, обрабатывающее и выполняющее запросы, полученные от клиента.

**REST API** (Representational State Transfer Application Programming Interface) – программный интерфейс, который позволяет взаимодействовать с удаленными серверами и обмениваться данными между клиентом и сервером в формате, основанном на принципах архитектурного стиля REST.

**HTTPS** (Hypertext Transport Protocol Secure) – протокол, который обеспечивает целостность и конфиденциальность данных при их передаче между сайтом и устройством пользователя.

**База данных** – упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе.

**Модерация** – контроль выполнения требований, установленных владельцем сайта.

**Фреймворк** – готовый набор инструментов, который помогает разработчику быстро создать продукт: сайт, приложение, интернет-магазин, CMS-систему.

**СУБД** (Система управления базами данных) – программное обеспечение, предназначенное для создания, управления, обновления и анализа баз данных.

**Идентификация** – процесс установления уникальной идентичности конкретного пользователя или субъекта в рамках системы, приложения или сервиса.

**Авторизация** – процесс проверки прав доступа пользователя к определенным ресурсам, функциям или данным в информационной системе.

## **2.2 Функциональные требования**

В системе пользователи будут иметь одну из ролей:

- неавторизованный пользователь;
- авторизованный пользователь;
- администратор.

### **2.2.1 Вход в приложение**

Входной точкой в приложение для любого типа пользователя должен являться splash-экран, на котором будут запрашиваться системные разрешения.

### **2.2.2 Авторизация и регистрация**

- Экран авторизации открывается при первом запуске приложения;
- если пользователь зарегистрирован, то он вводит свои почту и пароль, входит в свой аккаунт;
- если пользователь не зарегистрирован, он может перейти на экран регистрации, чтобы завести свой аккаунт.

## 2.3 Пользовательские сценарии

Анонимный пользователь может просматривать основные страницы приложения, на которых возможности этой роли ограничены. Для неавторизованного пользователя доступна регистрация, авторизация и просмотр аукционов. Также неавторизованный пользователь имеет возможность ограниченно просмотреть лоты. На рисунке 1 показана его use case диаграмма. Анонимный пользователь может просматривать основные страницы приложения, на которых возможности этой роли ограничены. Для неавторизованного пользователя доступна регистрация, авторизация и просмотр аукционов. Также неавторизованный пользователь имеет возможность ограниченно просмотреть лоты. На рисунке 1 показана его use case диаграмма.

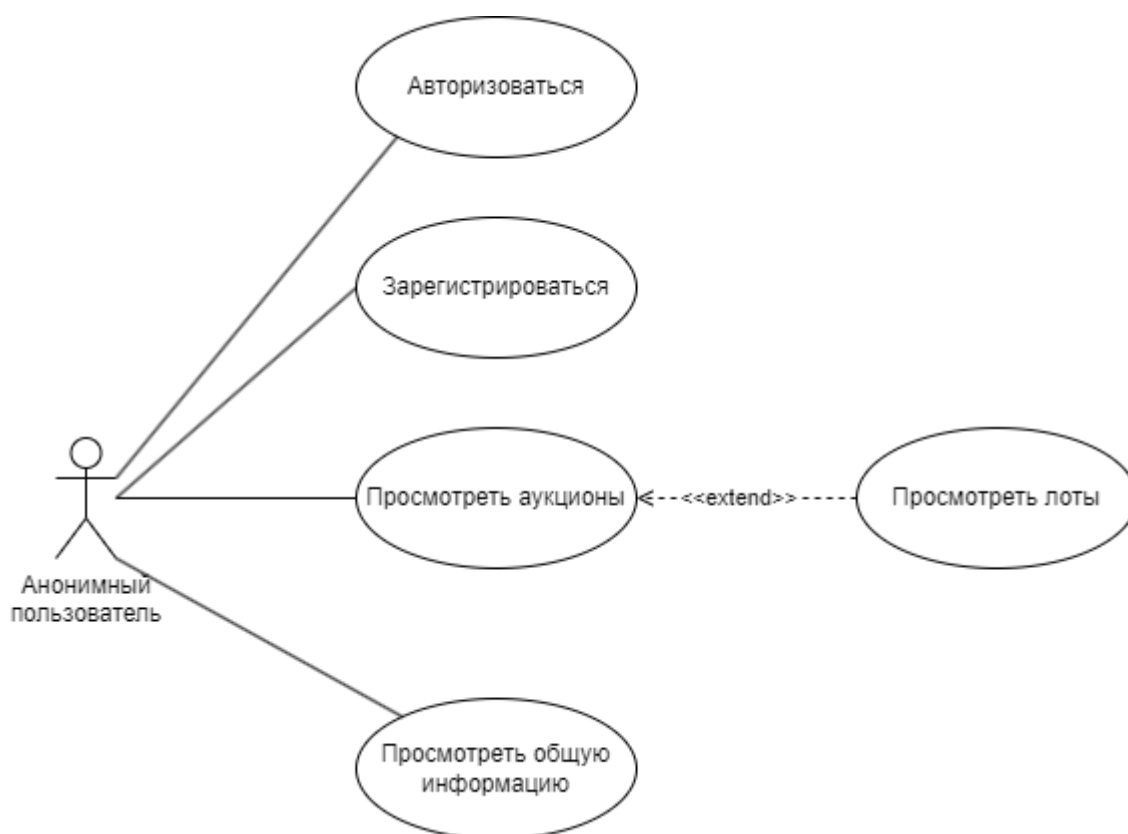


Рисунок 1 - Диаграмма для анонимного пользователя

Авторизованный пользователь имеет расширенные возможности анонимного пользователя, исключая авторизацию и регистрацию. На рисунке 2 показана его use case диаграмма.

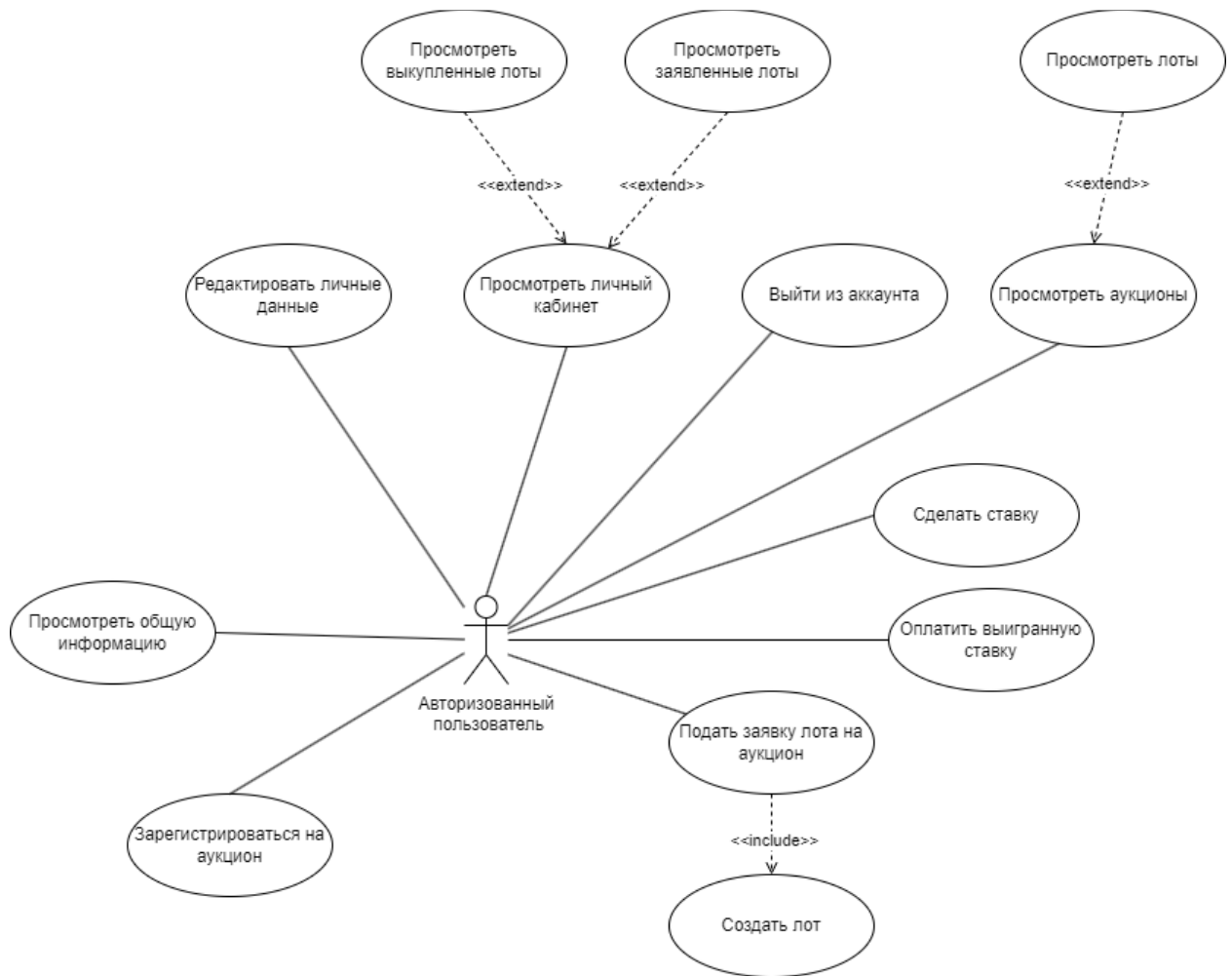


Рисунок 2 - Диаграмма для авторизованного пользователя

Администратор имеет расширенные возможности авторизованного пользователя. На рисунке 3 показана его use case диаграмма.



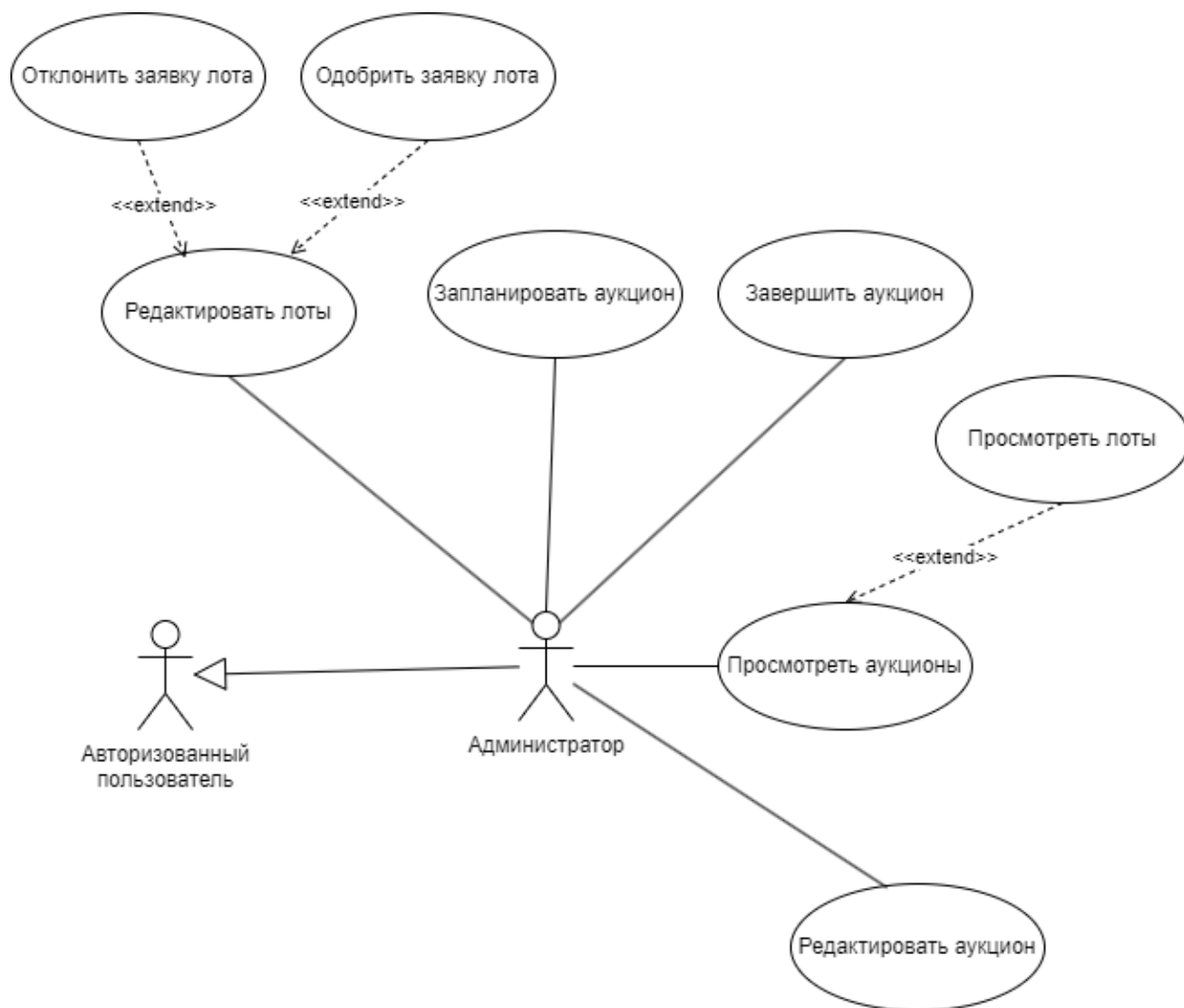


Рисунок 3 - Диаграмма для администратора

### 3 Требования к приложению и программному обеспечению

#### 3.1. Требования к структуре приложения в целом

##### 3.1.1 Требования к архитектуре

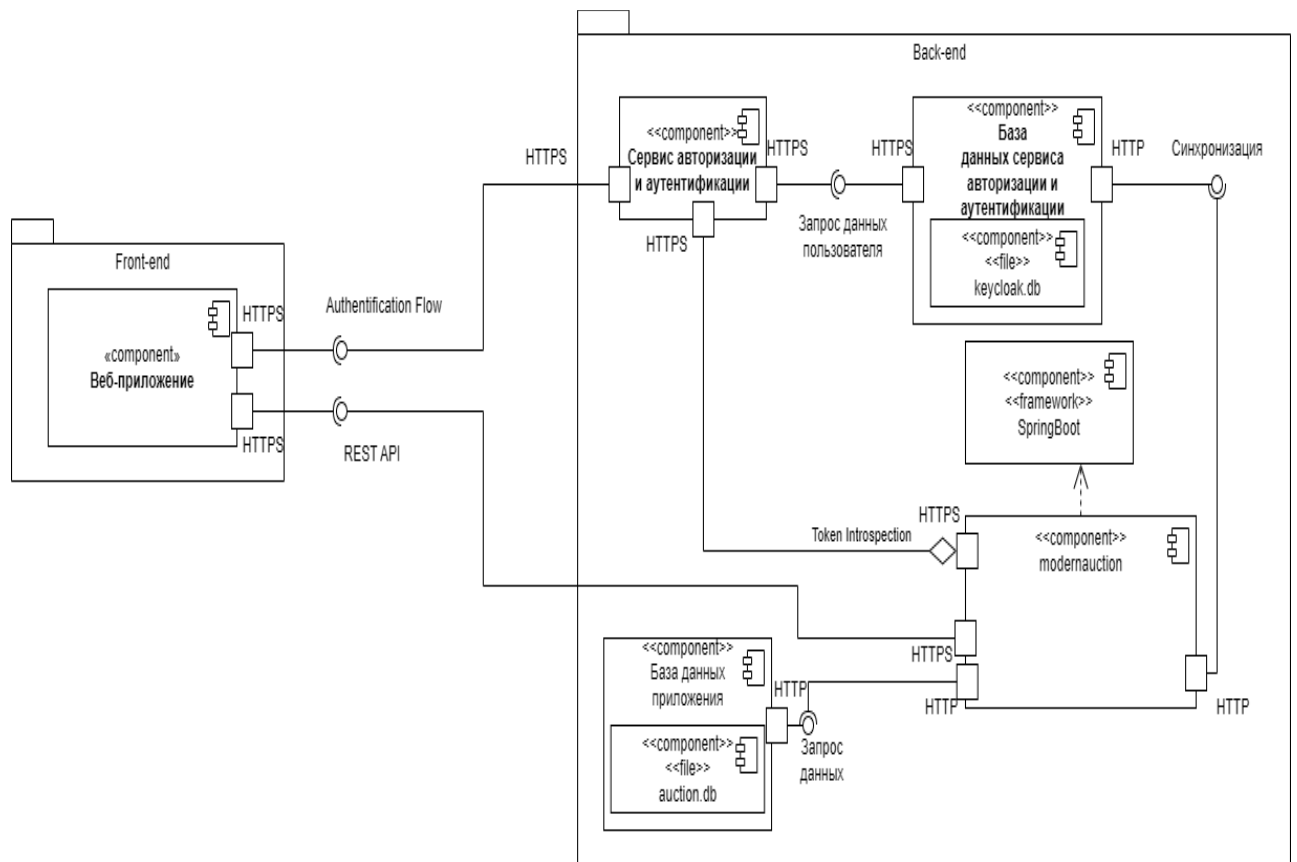
Приложение должно быть реализовано с использованием трехуровневой клиент-серверной архитектуры, где клиентский уровень взаимодействует с серверным уровнем через REST API. В серверной части должен применяться монолитный подход.

Связь между клиентом и сервером осуществляется посредством HTTPS протокола.

Приложение должно иметь базу данных для хранения личных данных пользователей.

Должна присутствовать интеграция с платежным сервисом.

Также для иллюстрации архитектуры системы были составлены диаграмма компонентов и развертывания, которые представлены на рисунках ниже.



#### Рисунок 4 - Диаграмма компонентов

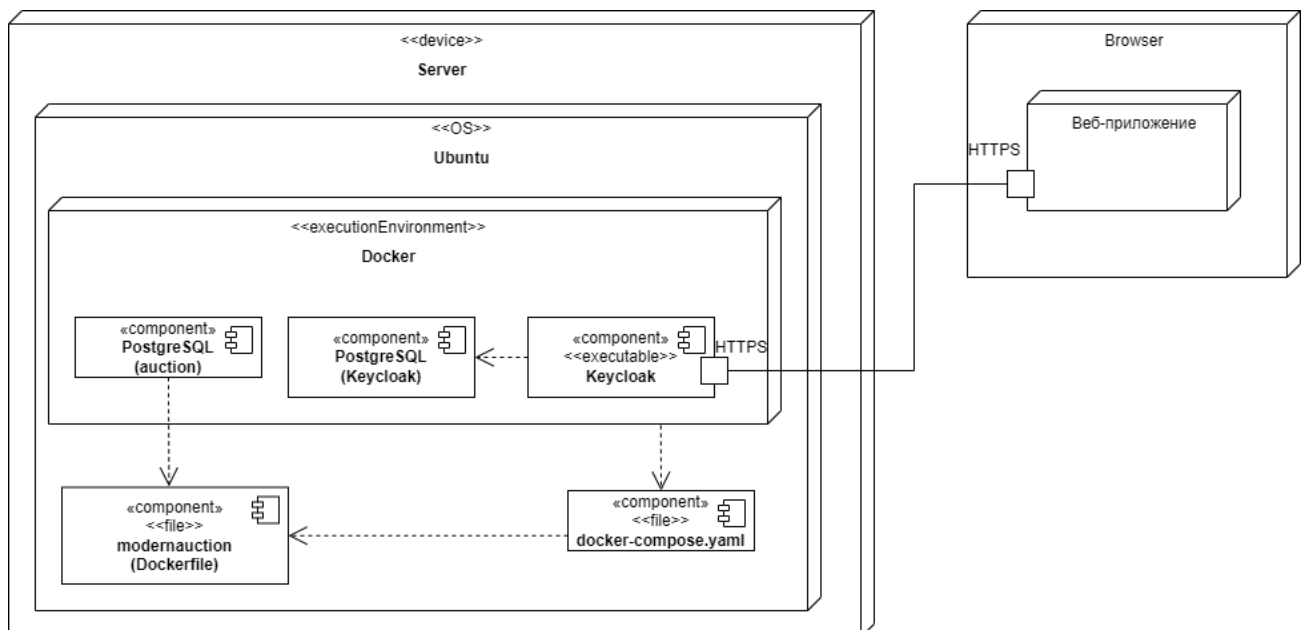


Рисунок 5 - Диаграмма разворачивания

### **3.1.2 Перспективы развития и модернизации приложения**

В дальнейшем для реализации могут быть рассмотрены следующие направления:

- создание платных подписок и разделение пользователей на категории относительно них;
- закрытые аукционы, доступные по приглашению;
- добавление залога для участия в аукционе;
- добавление реализации бана пользователей;
- возможность создания аукционов-событий с нестандартными правилами.

### **3.2 Требования к функциям приложения**

Для иллюстрации функций, которые должны быть реализованы в приложении была создана функциональная схема, которая представлена в приложении А.

Также для детализации выполнения некоторых процессов были созданы диаграммы последовательности. Их можно посмотреть в приложении Б.

Список процессов, на которые представлены диаграммы последовательностей:

- вход в аукцион;
- просмотр лотов для авторизованного пользователя;
- выставление ставки;
- выставление лота;
- редактирование личных данных;
- модерация лотов;
- создание аукциона;
- редактирование аукциона;
- завершение аукциона.

Помимо этого, для иллюстрации взаимодействия с лотом была составлена диаграмма активности, которая представлена в приложении В

### **3.3 Управление аккаунтом**

- Управление аккаунтом пользователя должно реализовываться на экране "Личного кабинета";
- пользователь должен иметь возможность просмотреть статистику;
- пользователь должен иметь возможность добавлять лоты с указанием запрашиваемых параметров.

## ПРИЛОЖЕНИЕ А ФУНКЦИОНАЛЬНАЯ СХЕМА

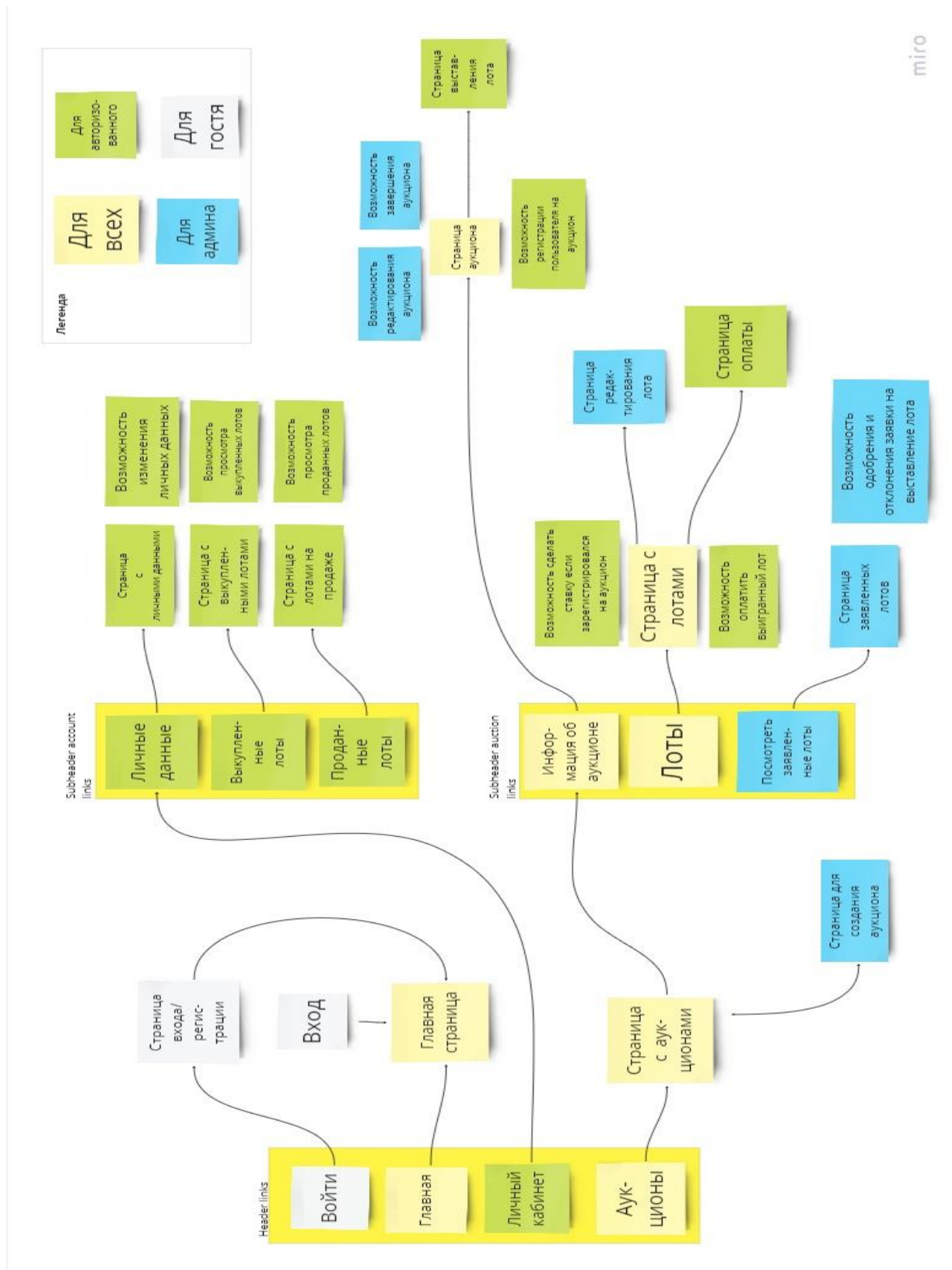


Рисунок 6 - Функциональная схема

## ПРИЛОЖЕНИЕ Б ДИАГРАММЫ ПОСЛЕДОВАТЕЛЬНОСТЕЙ

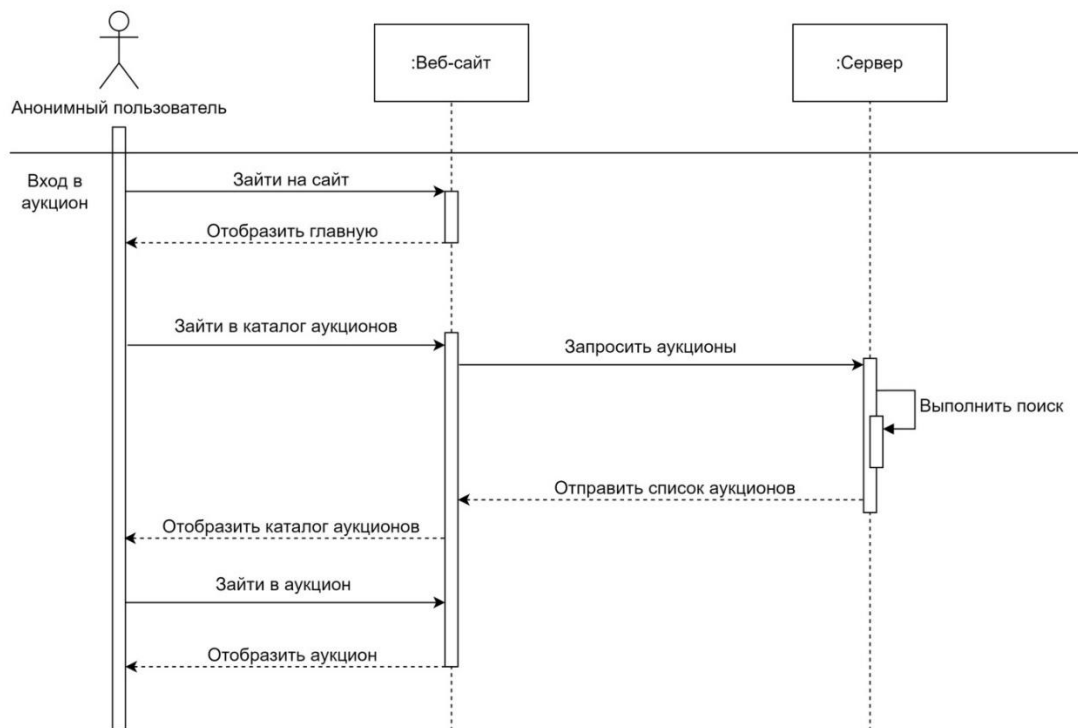


Рисунок 7 - Вход в аукцион

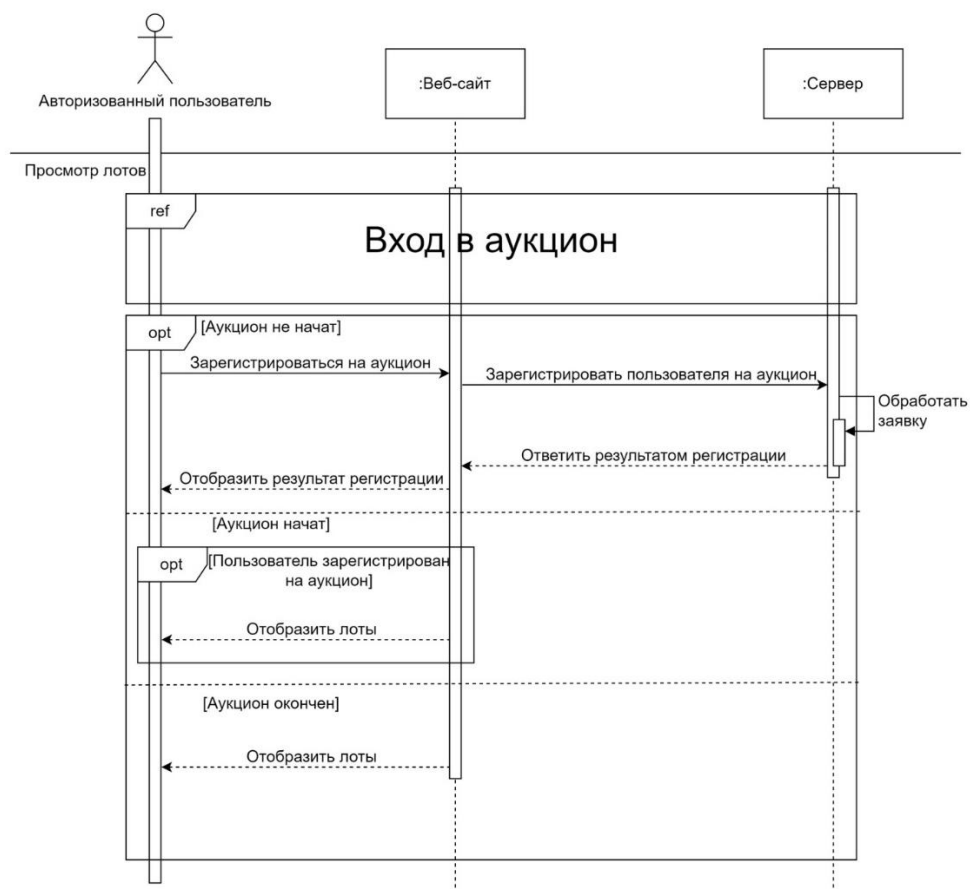


Рисунок 8 - Просмотр лотов

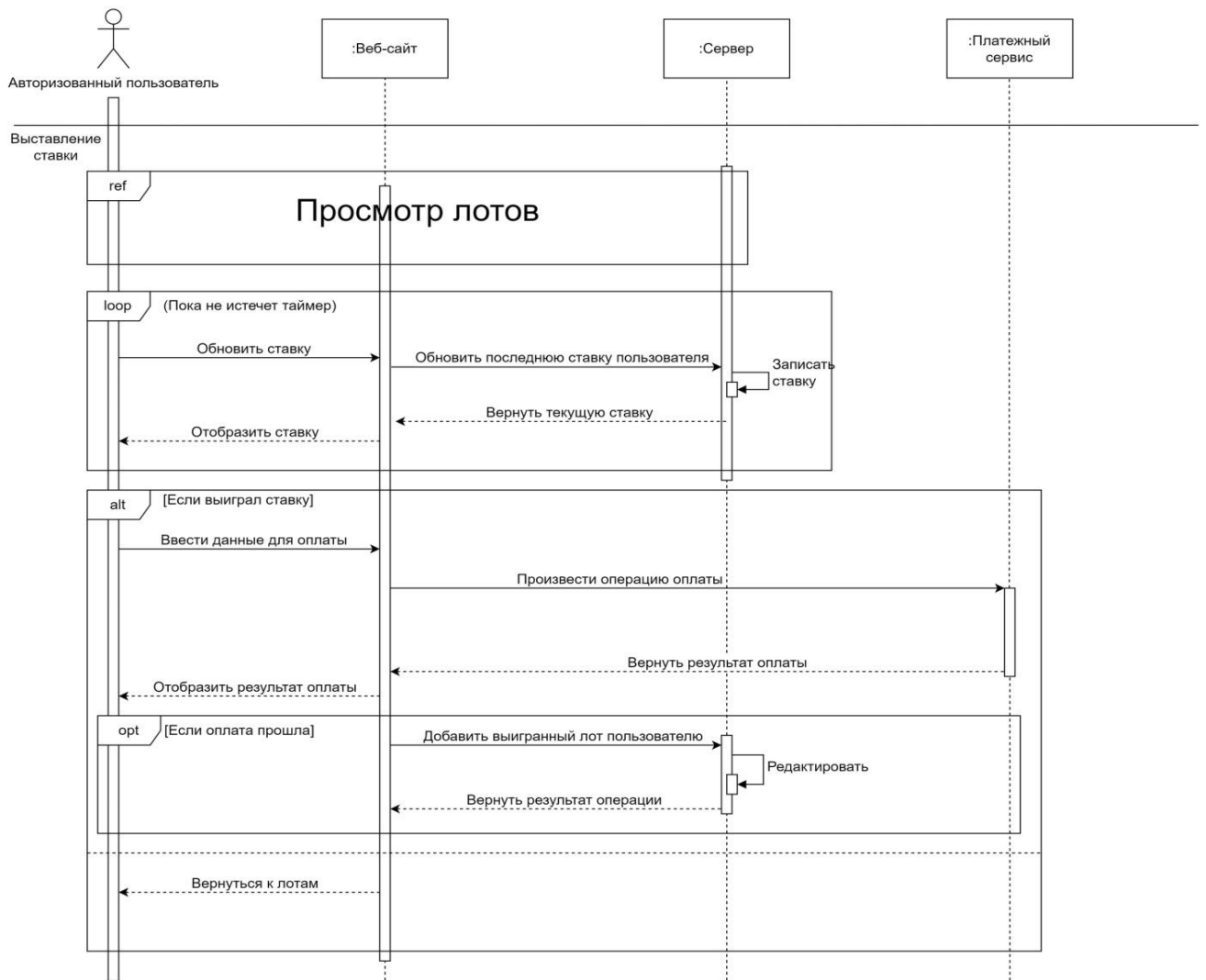


Рисунок 9 - Выставление ставки



Рисунок 10 - Выставление лота

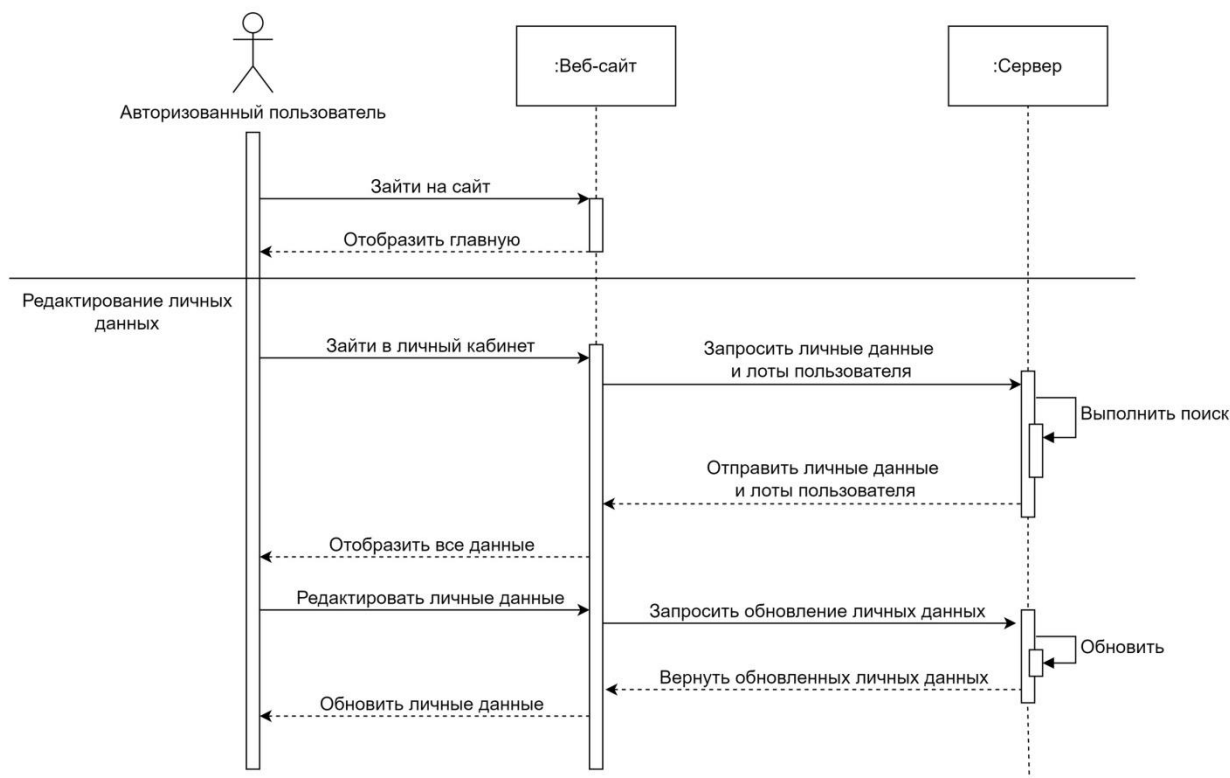


Рисунок 11 - Редактирование личных данных

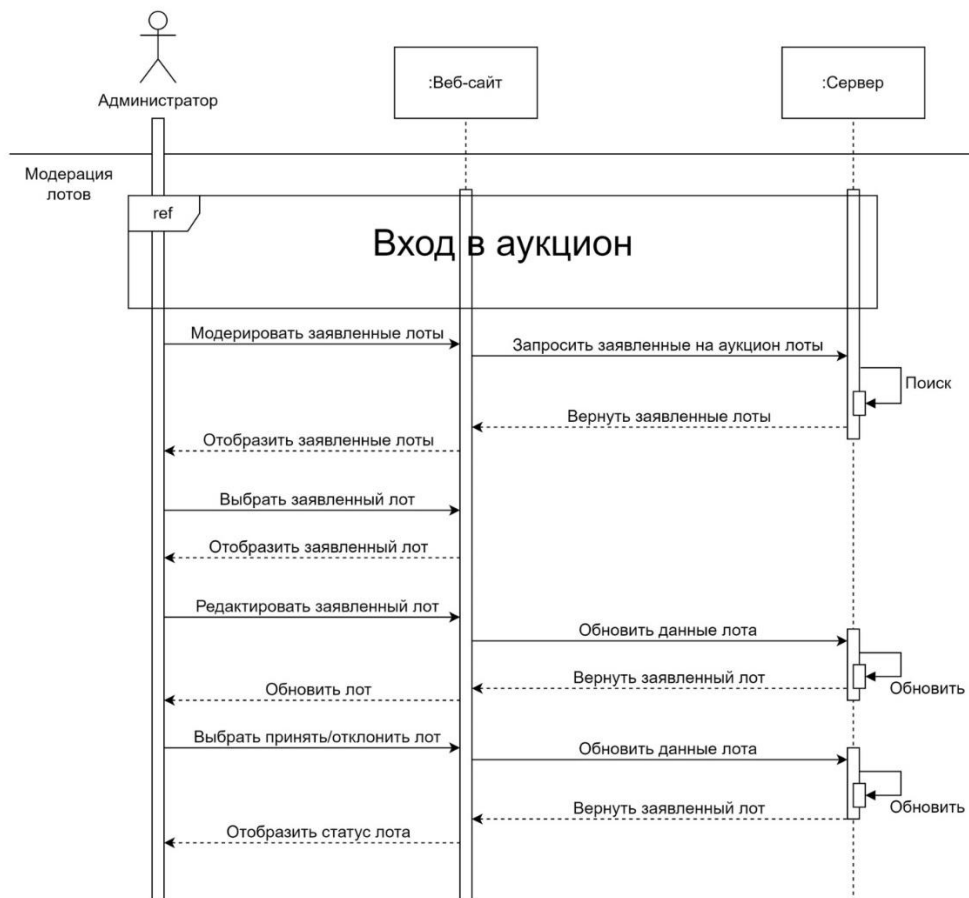


Рисунок 12 - Модерация лотов



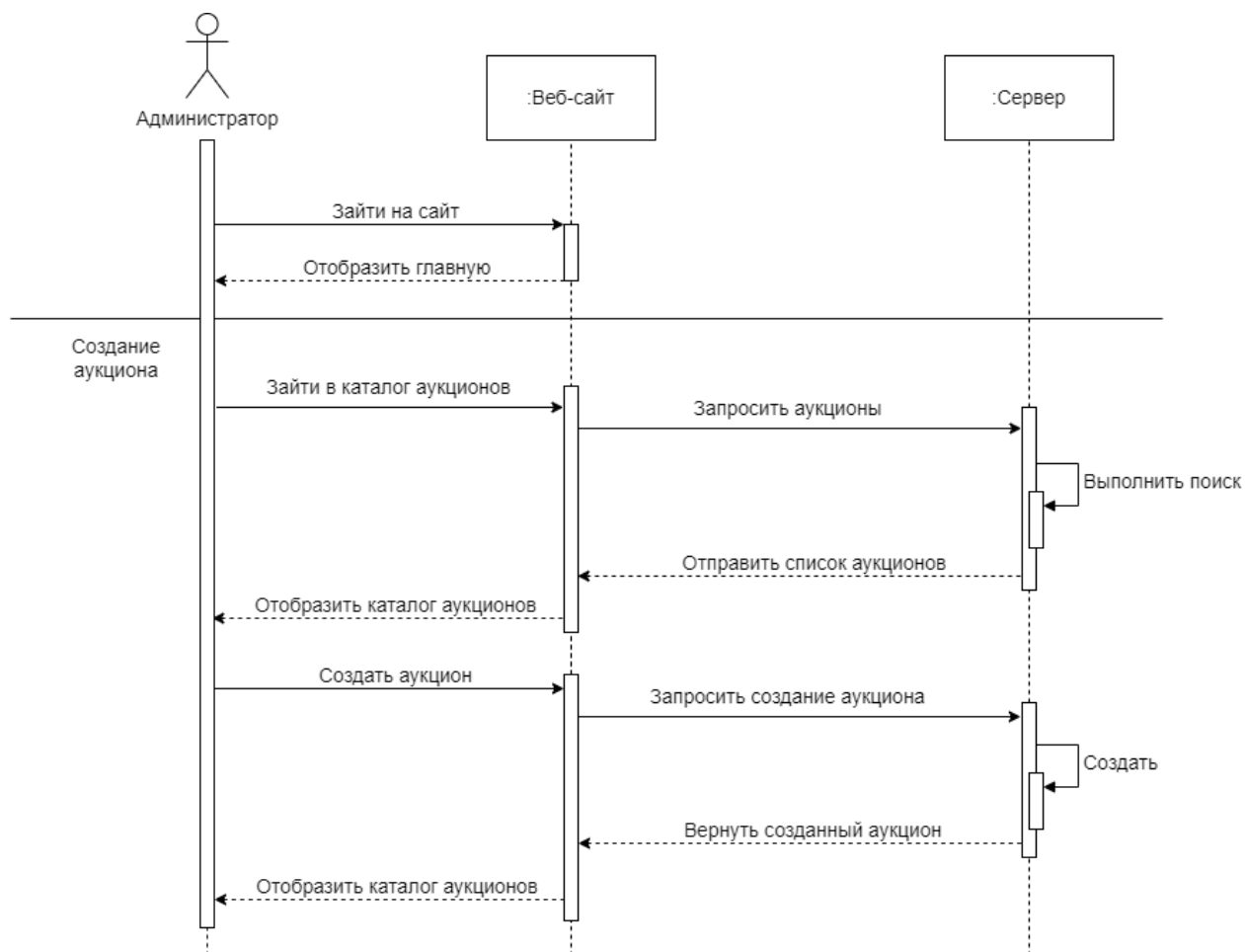


Рисунок 13 - Создание аукциона

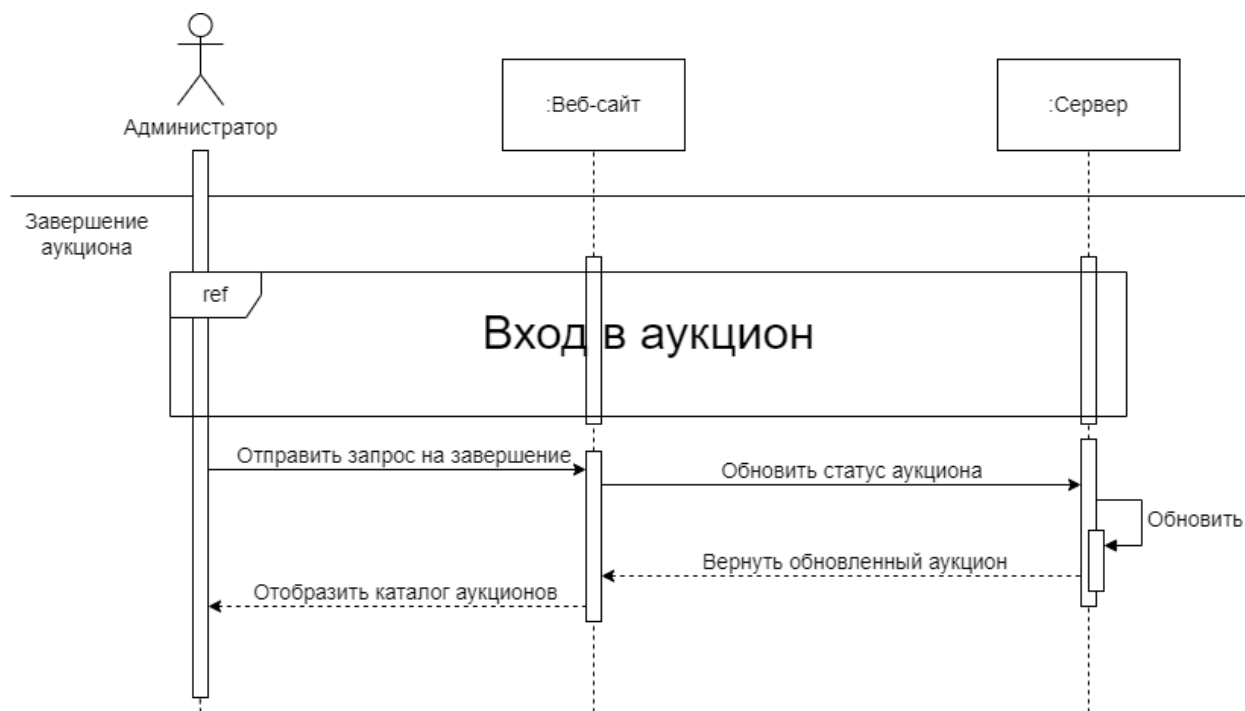


Рисунок 14 - Завершение аукциона

## ПРИЛОЖЕНИЕ В ДИАГРАММА АКТИВНОСТИ

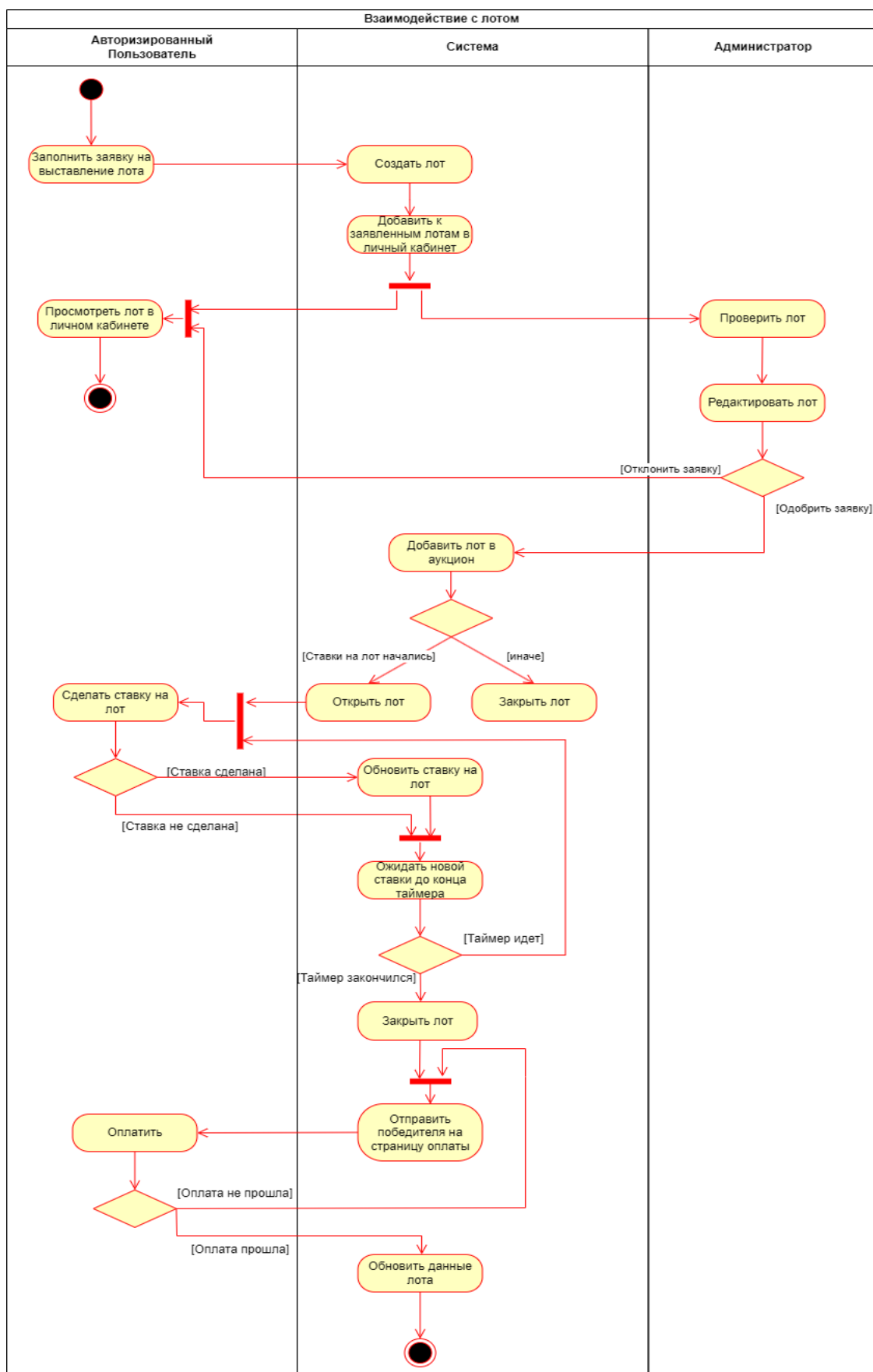


Рисунок 15 - Диаграмма активности