МИНОБРНАУКИ РОССИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**ВЫСШЕГО ОБРАЗОВАНИЯ**

**“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”**

Факультет *компьютерных наук*

*Кафедра технологий обработки и защиты информации*

*Приложение для аренды спортивного снаряжения «ProCat»*

*Курсовой проект*

*09.03.02 Информационные системы и технологии*

*Обработка информации и машинное обучение*

Допущен к защите

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_П.Л. Некрасов*, 3 курс, д/о*

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.Е. Яньшина*, 3 курс, д/о*

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.Ю. Сахаров*, 3 курс, д/о*

Воронеж 2021

Оглавление

[Оглавление 2](#_Toc67515089)

[Введение 3](#_Toc67515090)

[1. Постановка задачи 4](#_Toc67515091)

[2. Анализ предметной области 5](#_Toc67515092)

[2.1 Глоссарий 5](#_Toc67515093)

[2.2 Анализ существующих решений 5](#_Toc67515094)

[2.3 Анализ задачи 6](#_Toc67515095)

[2.3.1 Варианты использования приложения 6](#_Toc67515096)

[2.3.2 Взаимодействие компонентов системы 8](#_Toc67515097)

[2.3.3 Варианты состояния системы 8](#_Toc67515098)

[2.3.4 Варианты действия в системе 10](#_Toc67515099)

[2.3.5 Развертывание приложения 11](#_Toc67515100)

[3. Анализ средств реализации 12](#_Toc67515101)

[4. Реализация 12](#_Toc67515102)

[4.1 Сущности 12](#_Toc67515103)

[4.2 Сценарии воронок конверсии 12](#_Toc67515104)

[4.3 Графический интерфейс 14](#_Toc67515105)

[4.3.1 Начальная страница с выбором пункта проката 14](#_Toc67515106)

[4.3.2 Страница с выбором снаряжения 14](#_Toc67515107)

[4.3.3 Страница авторизации 14](#_Toc67515108)

[4.3.4 Страница просмотра заявок 14](#_Toc67515109)

[4.3.5 Страница оплаты 14](#_Toc67515110)

[4.3.6 Страницы администратора 14](#_Toc67515111)

[5. Тестирование 14](#_Toc67515112)

[5.1 Дымовое тестирование 14](#_Toc67515113)

[5.2 UI тесты 14](#_Toc67515114)

[Заключение 14](#_Toc67515115)

# Введение

В современном интернете есть различные сервисы, которые позволяют покупать или арендовать различные вещи. Но любому человеку нужен отдых, и самое очевидное и первостепенное, о чем думает каждый — это хобби. По статистике 42% людей всего мира занимаются спортом. В настоящее время существуют приложения и сайты, посвященные поиску и аренде спортивного снаряжения в каком-либо конкретном городе, но у каждого из них различные функциональные возможности, разный интерфейс. Сейчас набирают обороты именно мобильные приложения из-за их удобства и внешнего вида, именно поэтому было решено заняться именно разработкой мобильного приложения для Android.

Желаемое приложение должен облегчить пользователям поиск и аренду спортивного снаряжения. Тем самым необходимости в просмотре большого количества сайтов и установки приложений с арендой снаряжения не будет. Также приложение будет ориентироваться владельцев проката тоже, чтобы упростить приемку и выдачу спортивного товара.

Желаемое приложение должно предоставлять основную необходимую функциональность:

* Поиск необходимого снаряжения пользователем
* Оформление заявки, аренда и оплата
* Подтверждение начала и окончания времени бронирования администратором

Данный курсовой проект посвящен разработке именно такого, простого в освоении, но в то же время выполняющего самые необходимые функции приложения, способного уменьшить временные затраты каждого человека на поиск и аренду необходимого снаряжения.

# 1. Постановка задачи

Цель курсовой работы: реализовать приложение, которое отвечает следующим требованиям:

- регистрацию Гостей и авторизацию неавторизированных Пользователей

- просмотр списка доступного снаряжения для аренды Пользователем

- формирование заявки на аренду снаряжения Пользователем

- просмотр списка заявок Пользователем

- возможность досрочного закрытия заявки Пользователем с учетом подтверждения Администратора

- возможность формирования двух и более заявок на аренду одним Пользователем

- считывание деталей заказа с помощью QR-кода Администратором

- подтверждение начала и конца аренды Администратором

Для достижения данной цели были выделены следующие задачи:

Разработка Android части приложения, находящиеся на телефоне пользователя;

Разработка Back-end части сайта, развернутой на удаленном сервере сайта;

Создание связи между Android и Back-end частями сайта;

Разработка базы данных, расположенной на удаленном сервере.

# 2. Анализ предметной области

## 2.1 Глоссарий

***Заявка*** — заполненная пользователем информация, необходимая и достаточная для аренды спортивного снаряжения

## 2.2 Анализ существующих решений

1. <https://rentstation.ru/>

Достоинства:

* Большое разнообразие категорий спортивного снаряжения для аренды
* Лаконичное оформление
* Синхронизация с Facebook

Недостатки:

* Нет поисковой строки
* Отсутствие возможности оплаты на сайте
* Минимальный срок аренды - 1 неделя
* Отсутствие чата

1. <https://pro-interactive.ru/shop/>

Достоинства:

* Присутствие чата
* Синхронизация с различными соц. сетями

Недостатки:

* Неоднозначный дизайн и подбор цветов
* Перебои в работе сайта

1. <http://easysoftware.pro/projects/programma-ucheta-prokata/>

Достоинства:

* Обширный функционал
* Экспорт данных в Excel, PDF, Word
* Печать документов и отчетности
* Возможность интеграции приложения с сайтом

Недостатки:

* Сервис только для учета проката (со стороны владельца)
* Неоднозначный дизайн и подбор цветов

## 2.3 Анализ задачи

### 2.3.1 Варианты использования приложения

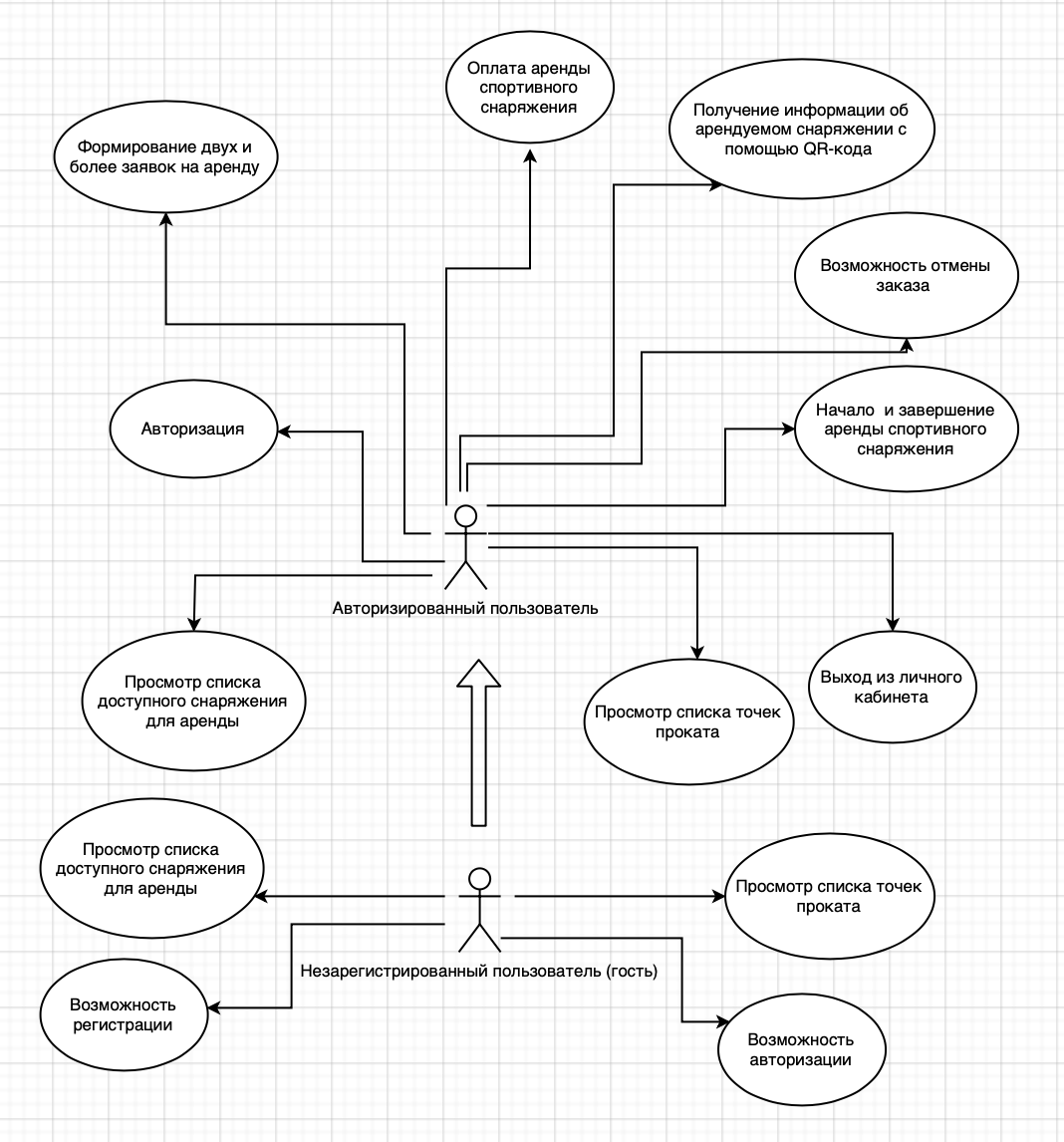


Рисунок 1. Диаграмма прецедентов.

При взаимодействии с сайтом у пользователя есть определенный

список возможностей, который более наглядно изображен на рисунке 1:

* Функционал гостя
* Просмотр главного экрана с выбором точек проката.
* Просмотр доступного для аренды снаряжения.
* Возможность регистрации
* Возможность авторизации
* Функционал авторизированного пользователя
* Просмотр главного экрана с выбором точек проката.
* Просмотр доступного для аренды снаряжения.
* Возможность авторизации
* Получение информации об арендуемом снаряжении при помощи QR-кода
* Формирование двух и более заявок на аренду
* Просмотр списка оформленных заявок
* Начало и завершение аренды спортивного снаряжения
* Оплата аренды спортивного снаряжения и получение чека
* Выход из личного кабинета

### 2.3.2 Взаимодействие компонентов системы

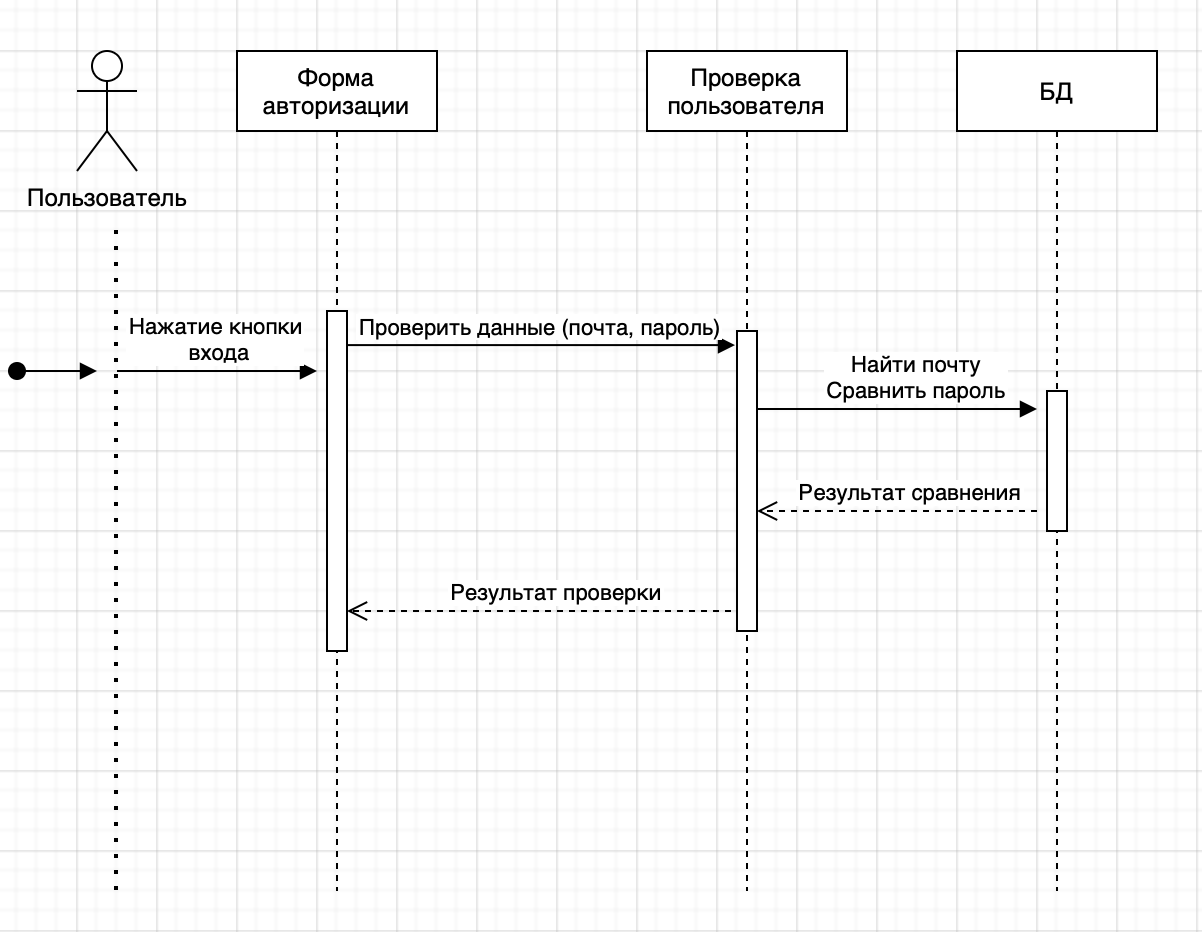


Рисунок 2. Диаграмма последовательностей.

На рисунке 2 показана диаграмма последовательности, на которой изображено упорядоченное во времени взаимодействие объектов при авторизации пользователя.  
 Для авторизации пользователь обращается к форме авторизации, которая передаёт введённые данные на проверку в модуль проверки пользователя. Тот в свою очередь проверяет существование данного пользователя в базе данных и совпадение введённого пароля с паролем, хранящимся в базе данных. Модуль проверки пользователя посылает статус проверки на форму авторизации, которая выводит пользователю результат авторизации.

### 2.3.3 Варианты состояния системы

### 

Рисунок 3. Диаграмма состояний.

Диаграмма состояний, изображенная на Рисунке 3, отражает возможные состояния заявки. Посредством сканирования QR-кода производится проверка существования заявки. Если заявки на данный товар еще не было, то создается новая заявка, и далее идет процесс пользования товаром, после окончания которого будет производиться повторная проверка существования заявки. Если заявка на данный товар уже была заведена, то происходит завершение заявки на аренду и оплата аренды.

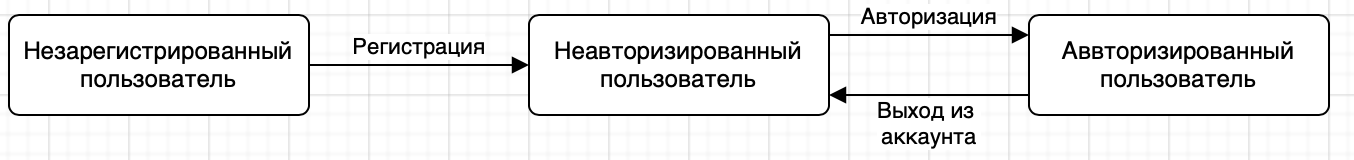


Рисунок 4. Диаграмма состояний.

Диаграмма состояний, изображенная на Рисунке 4, отражает возможные состояния пользователя. Изначально любой пользователь, не прошедший регистрацию, находится в состоянии незарегистрированного пользователя. После прохождения регистрации пользователь переходит в состояние неавторизованного пользователя. После авторизации пользователь переходит в состояние авторизованного пользователя. Если пользователь выходит из аккаунта он возвращается к состоянию неавторизованного пользователя и может авторизоваться снова. Если пользователь захочет пройти процесс регистрации, заполнив необходимые поля, система проверит входящие данные на корректность и в случае ошибки выдаст предупреждение. Если данные корректны, пользователь перейдет в личный кабинет.  
 При желании пользователя авторизоваться, заполнив необходимые поля, система так же проверит корректность введенных данных, в случае, если пользователя с такими данными нет, система предложит зарегистрироваться.

### 2.3.4 Варианты действия в системе

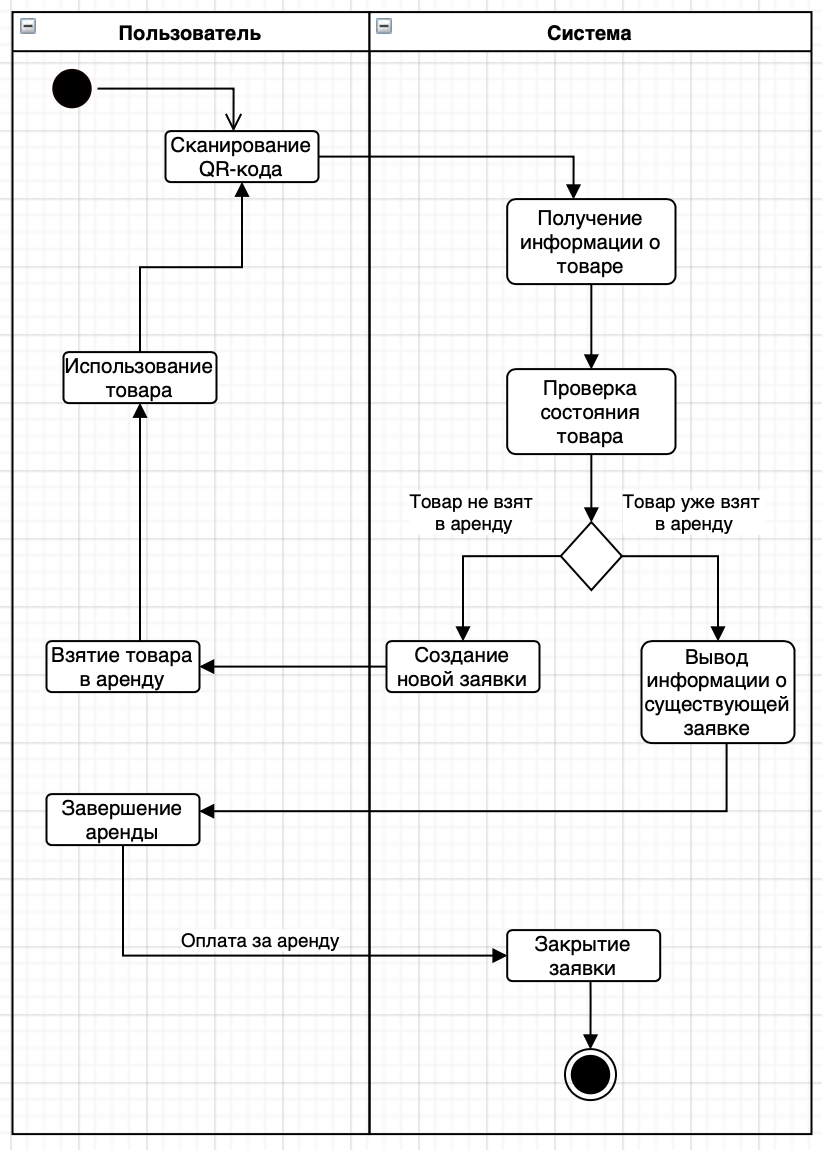


Рисунок 5. Диаграмма активности.

Диаграмма активности, изображенная на Рисунке 5, отражает действия авторизованного пользователя при создании заявки. После того, как авторизованный пользователь отсканирует QR-код, указанный на товаре, система получит данные о нем. Когда система проверит состояние товара, возможно два варианта развития событий:

1. В случае, если товар не был еще взят в аренду, то система создаст новую заявку на аренду снаряжения, после чего пользователь сможет взять в аренду товар. Попользовавшись товаром определенное время, пользователь должен отсканировать QR-код заново, чтобы впоследствии завершить аренду.
2. В случае, если товар уже был взят в аренду, система выведет информацию о существующей заявке, после чего пользователь сможет завершить аренду.

Далее пользователь оплачивает арендную плату, и заявка закрывается системой.

Рисунок 6. Диаграмма взаимодействий.

### 2.3.5 Развертывание приложения

Рисунок 7. Диаграмма развертывания.

На Рисунке 7 представлена диаграмма развертывания, чтобы определить какие аппаратные компоненты («узлы») существуют, какие программные компоненты работают на каждом узле и как различные части этого комплекса соединяются друг с другом. телефон и сервер, а в качестве узла среды выполнения выступает приложение. Развернуто Andriod-приложение, а на серверной части back-end и база данных.

# 3. Анализ средств реализации

В качестве средств реализации Android приложения были выбраны следующие технологии: Android SDK v30, Kotlin, RxJava 3, Dagger 2, Retrofit

В качестве СУБД была выбрана PostgreSQL. Она является хорошо масштабируемой, в равной степени легко может быть использована для работы, как с малыми, так и с большими объемами данных. А за счет упрощения некоторых используемых в ней стандартов система имеет высокую производительность.

В качестве языка разработки в back-end использовался язык Java 8, а также Maven, Spring Boot,Spring Data JPA

# 4. Реализация

## 4.1 Сущности

## 4.2 Сценарии воронок конверсии

1. Посетил начальную страницу - Перешел на страницу просмотра списка доступного оборудования
2. Посетил начальную страницу - Перешел на страницу просмотра списка доступного оборудования - Начал оформлять заявку
3. Посетил начальную страницу - Перешел на страницу просмотра списка доступного оборудования - Начал оформлять заявку
4. Посетил начальную страницу - Перешел на страницу просмотра списка доступного оборудования - Начал оформлять заявку - Зарегистрировался
5. Посетил начальную страницу - Перешел на страницу просмотра списка доступного оборудования - Начал оформлять заявку - Зарегистрировался – Оплатил
6. Посетил начальную страницу - Перешел на страницу просмотра списка доступного оборудования - Начал оформлять заявку - Зарегистрировался – Оплатил – Завел еще одну заявку

## 4.3 Графический интерфейс

### 4.3.1 Начальная страница с выбором пункта проката

### 4.3.2 Страница с выбором снаряжения

### 4.3.3 Страница авторизации

### 4.3.4 Страница просмотра заявок

### 4.3.5 Страница оплаты

### 4.3.6 Страницы администратора

# 5. Тестирование

## 5.1 Дымовое тестирование

## 5.2 UI тесты

# Заключение