

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ  
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО  
ОБРАЗОВАНИЯ «ВОРОНЕЖСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (ФГБОУ ВО  
«ВГУ»)

ТЕХНИЧЕСКОЕ ЗАДАНИЕ  
К КУРСОВОМУ ПРОЕКТУ  
«TeamScanner»

Воронеж, 2020

## Оглавление

1. Термины .....	4
2. Общие положения .....	5
2.1 Название сайта .....	5
2.2 Наименование разработчика и заказчика системы .....	5
2.3. Основание для разработки .....	5
2.4. Плановые сроки начала и окончания работ по созданию системы .....	5
2.5. Порядок оформления и предъявления заказчику результатов работ по созданию системы .....	5
3. Назначение и цели создания приложения .....	7
3.1 Назначение приложения .....	7
3.2 Цели создания приложения .....	7
4. Требования к сайту и программному обеспечению .....	8
4.1. Требования к приложению в целом .....	8
4.1.1 Требования к структуре и функционированию системы .....	8
4.1.1.1 Перечень подсистем, их назначение и основные характеристики .....	8
4.1.1.2 Требования к способам и средствам связи для информационного обмена между компонентами системы .....	9
4.1.2 Требования к численности и квалификации персонала системы и режиму его работы .....	9
4.1.3 Требования к надежности .....	9
4.1.4 Требования по эргономике и технической эстетике .....	9
4.1.5 Требования к патентной чистоте .....	10
4.1.6. Требования по стандартизации и унификации .....	10
4.1.7 Требования к тестированию системы .....	10
4.1.7.1. Дымовое тестирование. ....	10
4.1.7.2. Тестирование пользовательского интерфейса (UI тесты) .....	10
4.1.8. Требования к оформлению и верстке страниц .....	10
4.2 Требования к функциям (задачам), выполняемым системой .....	11
4.2.1. Перечень функций, подлежащих автоматизации .....	11
4.2.2. Временной регламент реализации каждой функции. ....	11

4.3 Требования к видам обеспечения .....	11
4.3.1. Требования по применению систем управления базами данных .....	11
4.3.2. Требования к лингвистическому обеспечению системы .....	11
5. Структура сайта .....	13
6. Языковые версии сайта .....	13
7. Группы пользователей .....	13
8. Дизайн сайта .....	13
9. Навигация сайта.....	14
9.1 Основное навигационное меню .....	14
9.2 Дополнительная навигация по сайту.....	14
10.Описание страниц сайта.....	15
10.1 Описание статических страниц .....	15
10.1.1.Главная страница.....	15
10.1.2. Страница регистрации.....	15
10.1.3. Страница авторизации.....	15
10.1.4. Личный кабинет.....	15
10.2. Описание динамических страниц .....	16
10.2.1. Страница поиска .....	16
10.2.2. Страница подробного просмотра события.....	16
11. Функционал .....	17
11.1. Действующие лица .....	17
11.2. ВИ “Регистрация и управление профилями”.....	17
11.3. ВИ “Создание объявления об игре”. .....	18
11.4. ВИ “Поиск игры и регистрация на нее” .....	19
11.5. ВИ “Модерация системы”.....	20
12. Порядок контроля и приемки работ. ....	21

## 1. Термины

Участник - Авторизованные пользователи, подписавшиеся на объявление о сборе команды.

Доска комментариев - Специальное место на странице объявления, где Участники могут оставлять свои комментарии.

ВИ - Вариант Ипользования или Use Case, описание см.  
[https://ru.wikipedia.org/wiki/Сценарий\\_использования](https://ru.wikipedia.org/wiki/Сценарий_использования).

Система - Сервис помощи создания команды “TeamScanner”, требования к которому указаны в данном документе.

Личный кабинет - раздел, доступный только конкретному авторизованному пользователю.

Фреймворк – программная платформа, определяющая структуру программной системы.

Валидация – проверка вводимых пользователем данным на корректность.

Модерация – процесс контроля действий пользователей на соответствие принципам ресурса.

## **2. Общие положения**

### **2.1 Название сайта**

Сервис помощи создания команды “TeamScanner”

### **2.2 Наименование разработчика и заказчика системы**

Разработчик:

Студент Антоненко Илья Игоревич, кафедра информационных систем управления предприятием.

Студент Волков Александр Григорьевич, кафедра программирования и информационных технологий

Студент Маликов Станислав Алексеевич, кафедра программирования и информационных технологий

Заказчик:

Преподаватель Тарасов Вячеслав Сергеевич, кафедра программирования и информационных технологий.

### **2.3 Основание для разработки**

Необходимость создания курсового проекта по дисциплине «Технологии программирования» для успешного прохождения курса.

### **2.4 Плановые сроки начала и окончания работ по созданию системы**

Плановый срок начала работ – Февраль 2020 г.

Плановый срок окончания работ – Июнь 2020 г.

### **2.5 Порядок оформления и предъявления заказчику результатов работ по созданию системы**

Предусматривается предъявление заказчику как промежуточных результатов по созданию системы в рамках рубежных аттестаций, так и конечного продукта во время защиты проекта. Также Заказчик может

самостоятельно ознакомиться с текущим состоянием проекта по ссылке

<https://github.com/StasMalikov/TeamScanner>

### **3. Назначение и цели создания приложения**

#### **3.1 Назначение приложения**

Автоматизация процесса поиска команды для спортивных игр

#### **3.2 Цели создания приложения**

Создание web - приложения, с помощью которого пользователи смогут найти себе сокомандников для игр разных видов спорта, откликаясь на объявления о наборе, которые создают другие пользователи.

## 4. Требования к сайту и программному обеспечению

### 4.1 Требования к приложению в целом

#### 4.1.1 Требования к структуре и функционированию системы

##### 4.1.1.1 Перечень подсистем, их назначение и основные характеристики

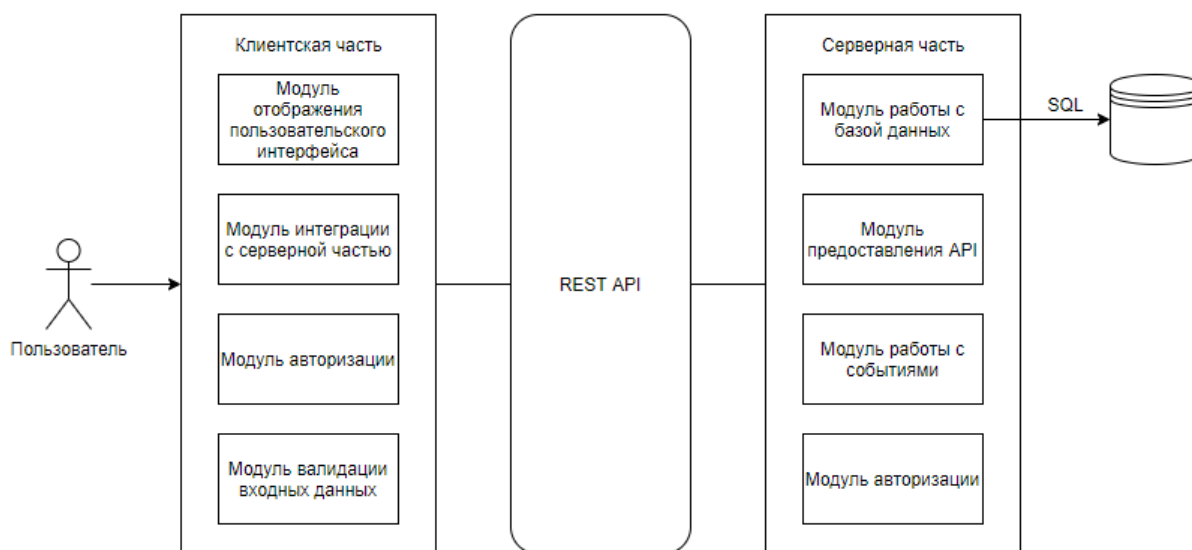


Рисунок 1. Модульная структура приложения

Приложение должно иметь архитектуру, основанную на шаблоне проектирования Model-View-Controller, а также иметь разделение на front-end и back-end.

Базы данных (Model) представляет собой фундаментальные данные, необходимые для работы приложения.

Клиентская часть (View) является подписчиком на событие изменения значений свойств, запросов или команд, предоставляемых серверной частью. Если пользователь воздействует на какой-либо элемент интерфейса, вызывается соответствующая команда, предоставляемая серверной частью.

Серверная часть (Controller) предоставляет обёртку данных из базы данных, которые подлежат связыванию. Таким образом, она является



посредником между клиентской частью и базой данных, предоставляя пользователю возможность для обработки вводимой им информации, извлекая данные из базы данных и форматируя их для отображения в графическом интерфейсе.

#### **4.1.1.2 Требования к способам и средствам связи для информационного обмена между компонентами системы**

Взаимодействие между серверной и клиентской частями осуществляется с помощью REST API.

#### **4.1.2 Требования к численности и квалификации персонала системы и режиму его работы**

Предполагается, что один модератор сможет сопровождать систему в 100 пользователей. При достижении каждой дальнейшей сотни пользователей, необходимо вводить в систему нового модератора.

Как модерация, так и обычное использование системы не требует технической подготовки.

#### **4.1.3 Требования к надежности**

При работе системы возможны следующие аварийные ситуации, влияющие на надежность работы системы:

- отсутствие ответа источников данных
- аварийное прерывание работы программы

#### **4.1.4 Требования по эргономике и технической эстетике**

Система должна иметь удобный для конечного пользователя интерфейс. Для этого:

- Должна быть выдержана одна цветовая гамма
- Использование чрезмерно ярких цветов допускается только для выделения

- При возникновении ошибки в работе системы выводится уведомление пользователю.

#### **4.1.5 Требования к патентной чистоте**

Данный проект должен не нарушать никаких лицензий и патентов. В случае нарушения всю ответственность несет сторона Исполнителя.

#### **4.1.6 Требования по стандартизации и унификации**

Разработка системы планируется с использованием каскадной модели разработки программного обеспечения

Для разработки клиентской части используется Angular 8

Для разработки серверной части Java - Spring Boot

Для работы с базой данных PostgreSQL

#### **4.1.7 Требования к тестированию системы**

Должны быть проведены следующие виды тестирования:

##### **4.1.7.1 Дымовое тестирование.**

Необходимо проверить работоспособность приложения на всех основных сценариях использования:

##### **4.1.7.2 Тестирование пользовательского интерфейса (UI тесты)**

Необходимо проверить работоспособность всех интерфейсных элементов приложения, с которыми должен взаимодействовать конечный пользователь.

#### **4.1.8 Требования к оформлению и верстке страниц**

- Все поля ввода должны валидироваться, а ошибки валидации выделяться.
- Все страницы сайта должны быть выдержаны в едином стиле.
- Макет должен быть сверстан с применением фреймворка bootstrap4+

- Для разметки используются теги HTML5

## **4.2 Требования к функциям (задачам), выполняемым системой**

### **4.2.1 Перечень функций, подлежащих автоматизации**

В режиме функционирования система должна обеспечивать:

- Регистрацию, авторизацию, возможность просмотра личного кабинета, выхода из Учетной записи.
- Возможность создания объявления о поиске игроков (только для авторизованных пользователей), просмотра списка Участников, обмена комментариями на Доске комментариев.
- Поиск объявлений по нескольким критериям :категория вида спорта, город, дата проведения.
- Возможность откликнуться на объявление о поиске игроков и оставить комментарий на Доске комментариев(только для авторизованных пользователей).
- Существование роли администратора. Он может редактировать и удалять существующие объявления.

### **4.2.2 Временной регламент реализации каждой функции.**

Весь требуемый функционал необходимо реализовать до момента защиты курсового проекта.

## **4.3 Требования к видам обеспечения**

### **4.3.1 Требования по применению систем управления базами данных**

База данных должна управляться с помощью СУБД PostgreSQL

### **4.3.2 Требования к лингвистическому обеспечению системы**

При реализации клиентской части системы должен использоваться Angular 8.

Для реализации серверной части должен использоваться высокоуровневый язык программирования Java с использованием фреймворка Spring Boot.

Для реализации алгоритмов манипулирования данными в системе поиска команд необходимо использовать стандартный язык запроса к данным SQL.

Для организации диалога системы с пользователем должен применяться графический пользовательский интерфейс.

## 5 Структура сайта

В качестве структуры выбрана древовидная.

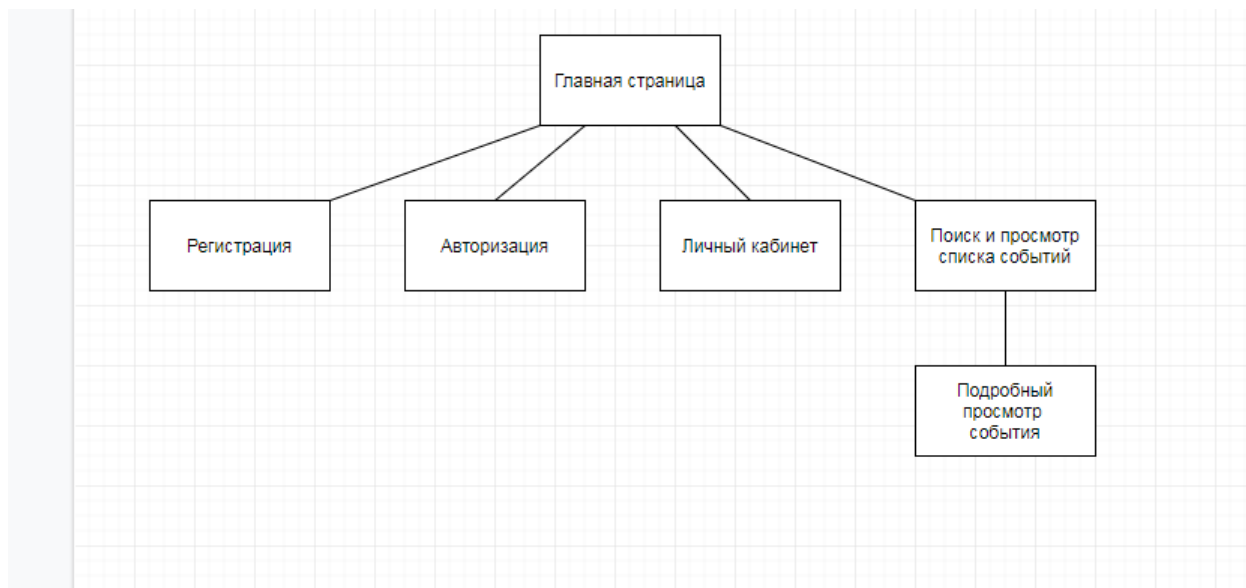


Рисунок 2. Структура сайта

## 6 Языковые версии сайта

Предполагается реализация только русского интерфейса.

## 7 Группы пользователей

Сайт должен предполагать три роли пользователей:

1. Базовый - для конечных пользователей системы
2. Модератор - для персонала, обслуживающего систему
3. Администратор - для персонала, обслуживающего систему

## 8 Дизайн сайта

Необходимо добиться интуитивно понятного дизайна. На главной странице сайта должны располагаться карточки видов спорта, при нажатии на которые будет переход на страницу поиска. В верхней части каждой страницы ,над контентом ,по центру, размещается логотип сайта. Общий фон сделать светлый, выдержать его на всех страницах.

## **9 Навигация сайта**

### **9.1 Основное навигационное меню**

С главной страницы на страницу поиска пользователь попадает путем нажатия на необходимую ему карточку вида спорта.

Со страницы поиска на страницу подробного просмотра события пользователь должен переходить, нажимая на запись интересующего объявления из списка, который составляется при поиске. Чтобы попасть обратно на главную страницу пользователь должен нажать на логотип TeamScanner'a, находящийся над контентом.

### **9.2 Дополнительная навигация по сайту**

В личный кабинет пользователь должен попадать с главной страницы сайта. На странице личного кабинета необходимо расположить иконку выхода из учетной записи. Если неавторизованный пользователь попытается стать Участником события, он будет перенаправлен на страницу регистрации.

## 10 Описание страниц сайта

### 10.1 Описание статических страниц

#### 10.1.1 Главная страница

При открытии сайта пользователь попадает на Главную страницу. При входе неавторизованного пользователя ему должна быть предоставлена возможность перейти на страницы авторизации и регистрации.

На главной странице расположены иконки карточек разных видов спорта. Пользователь выбирает интересующую его карточку и переходит на страницу поиска.

#### 10.1.2 Страница регистрации

Таблица 1. Требования к странице регистрации.

Название	Тип	Описание
Фамилия	Строка	Фамилия пользователя
Имя	Строка	Имя пользователя
Логин	Строка	Строка не менее 6 символов
Пароль	Строка	Строка не менее 6 символов
Подтверждение пароля	Строка	Строка не менее 6 символов, должна совпадать с паролем

#### 10.1.3 Страница авторизации

Таблица 2. Требования к странице авторизации.

Название	Тип	Описание
Логин	Строка	Строка не менее 6 символов
Пароль	Строка	Строка не менее 6 символов

#### 10.1.4 Личный кабинет

На странице личного кабинета должны находиться личные данные пользователя (фамилия, имя, город), список подписок на игры и игр,

созданных пользователем и иконка выхода из учетной записи. Также из личного кабинета пользователь может создать новое объявление об игре.

## **10.2 Описание динамических страниц**

### **10.2.1 Страница поиска**

Страница поиска должна быть разделена на два сектора. На меньшем по размеру секторе пользователь выбирает критерий для поиска: дата проведения, город, категория вида спорта. После, пользователь нажимает на кнопку и видит список доступных игр на большем секторе. Кликнув по интересующему, пользователь переходит на страницу просмотра объявления об этой игре.

### **10.2.2 Страница подробного просмотра события**

На странице просмотра объявления находится следующая информация о событии:

- Название
- Описание
- Дата создания
- Дата проведения
- Количество зарегистрировавшихся участников
- Необходимое количество участников
- Доска комментариев
- Кнопка «Подписаться»
- Кнопка «Отписаться»



## 11 Функционал

На Диаграммах вариантов использования представлены основные варианты использования системы.

### 11.1 Действующие лица

На данной диаграмме представлена иерархия всех пользователей системы. Связь обобщения следует читать следующим образом: пользователь наследует все поведение своего родителя, а также имеет свое поведение в системе. Модератор обладает всеми правами пользователя, но может еще и блокировать, и разблокировать как объявления пользователя, так и самого Пользователя. Администратор обладает всеми правами модератора, но может еще и назначать модератора.

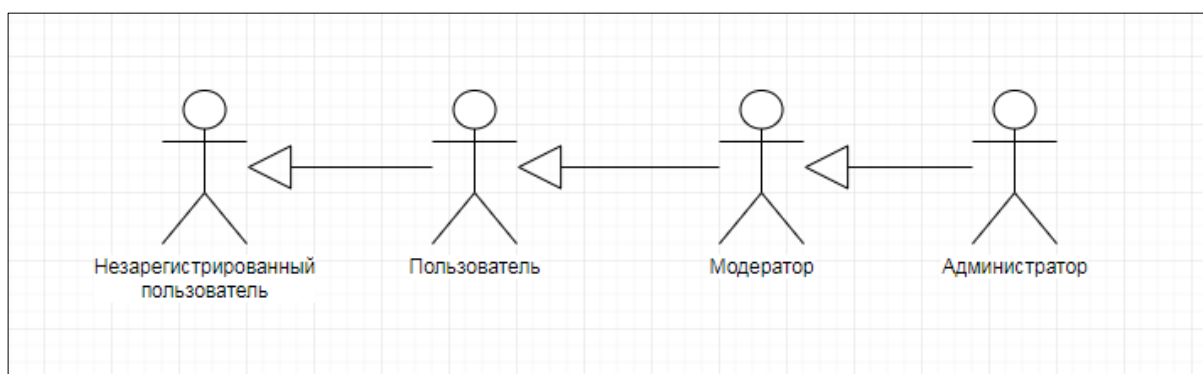


Рисунок 3. Действующие лица.

### 11.2 ВИ “Регистрация и управление профилями”.

Исходя из п.11.1.1 управление учетными записями пользователей необходимо построить следующим образом.

Незарегистрированный пользователь должен создать свою учетную запись путем заполнения формы регистрации, см. п. 10.1.2.

Зарегистрированный пользователь должен иметь возможность войти в учетную запись с помощью формы авторизации, см п. 10.1.3. После авторизации он может внести изменения в профиль или выйти из системы.

Деятельность пользователей контролируется модераторами. В их полномочия должны входить блокировки и разблокировки пользователей. Модераторов устанавливает администратор.

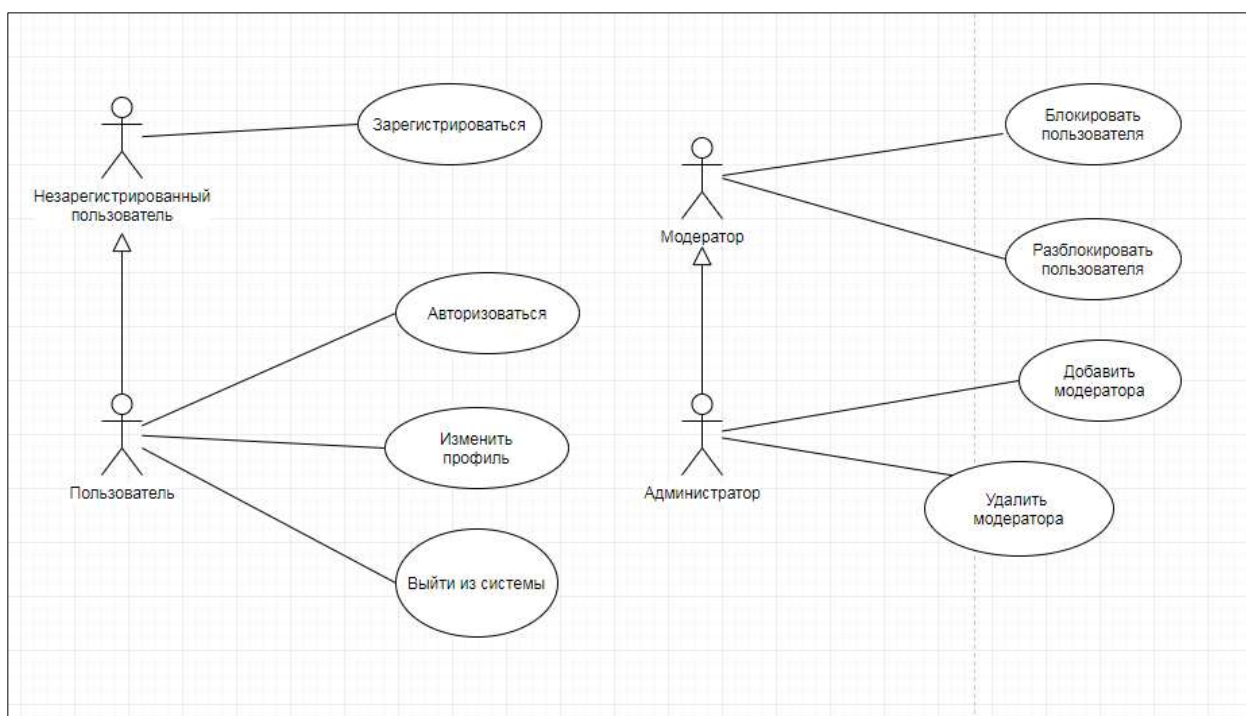


Рисунок 4. ВИ регистрации и управления профилями.

### 11.3 ВИ “Создание объявления об игре”.

Авторизованный аользователь (далее Пользователь) должен иметь возможность создать новое объявление о наборе команды для игры. Он вводит данные, см. пункт 10.2.2, по которым другие пользователи смогут найти объявление. После создания, это объявление можно просмотреть, редактировать и оставить комментарий на доске комментариев.

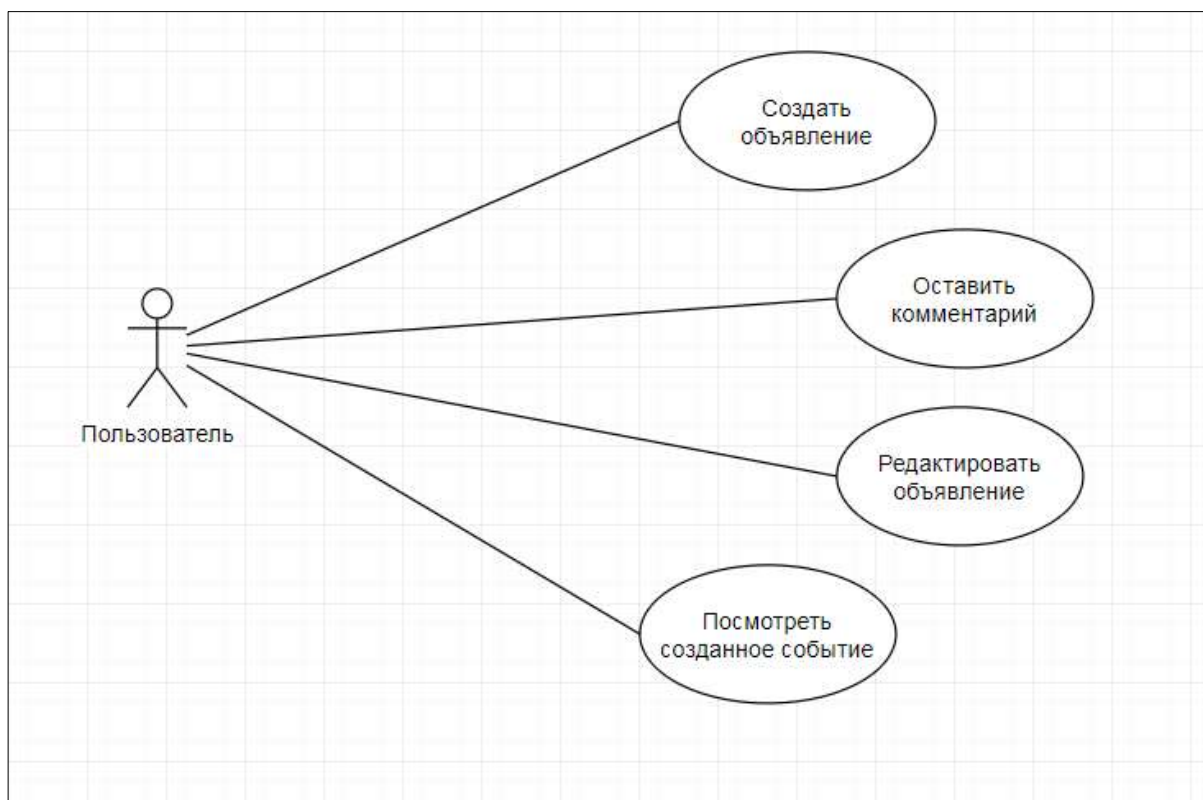


Рисунок 5. ВИ создания объявления об игре

#### 11.4 ВИ “Поиск игры и регистрация на нее”

В создаваемой системе поиск и просмотр созданных объявлений об играх доступно как незарегистрированным (или неавторизованным), так и авторизованным пользователям. Но откликнуться на объявление могут только авторизованные пользователи. После того, как они становятся участниками, пользователи могут оставить комментарий на доске комментариев или отказаться от участия в игре.

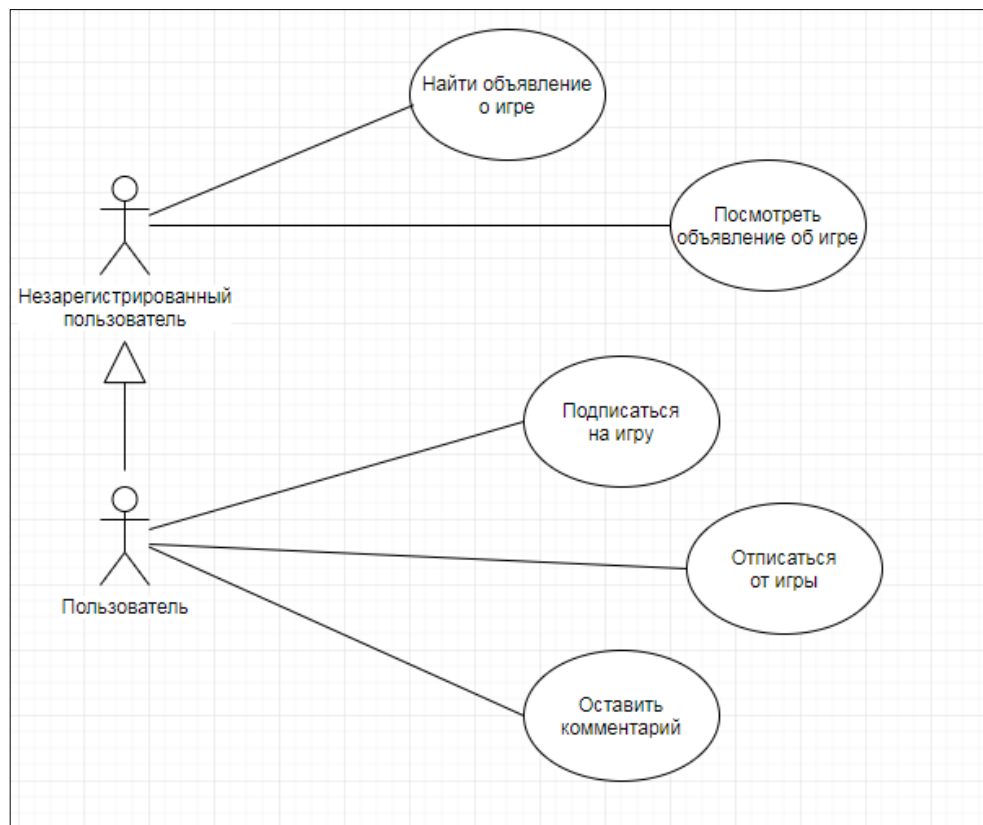


Рисунок 6. ВИ поиск игры и регистрации на нее.

### 11.5 ВИ “Модерация системы”

Для контроля действий пользователя существует роль модератора. Они могут посмотреть любое объявление и, при нарушениях, удалить его. Также модераторы имеют право на разблокировку объявлений.

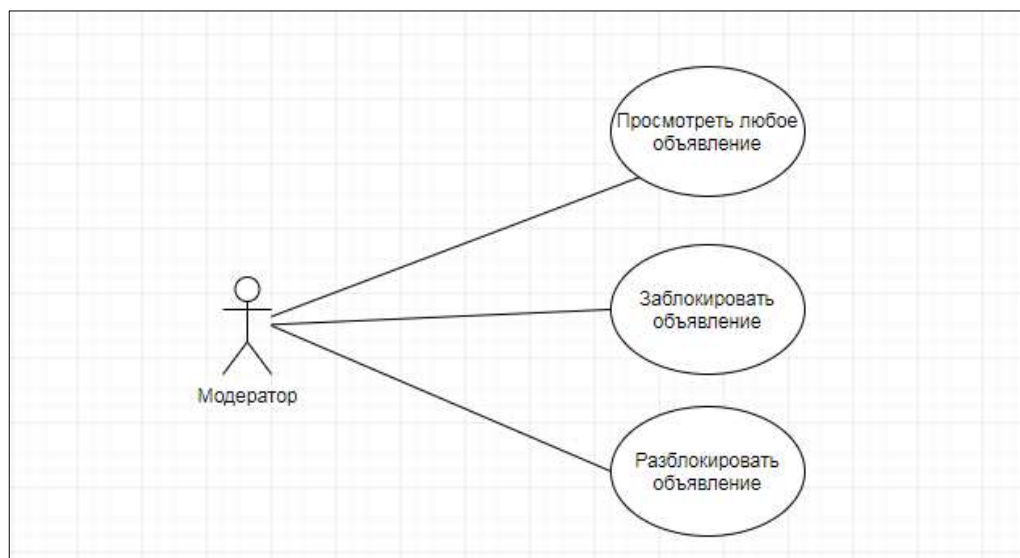


Рисунок 7. ВИ модерации системы.

## **12. Порядок контроля и приемки работ.**

Контроль разработки системы осуществляется посредством предоставления необходимой Заказчику отчетности Исполнителем. Готовая система с полной документацией должна быть представлена Заказчику в требуемые им сроки. Заказчик определяет соответствие системы требованиям и осуществляет её приём.

При приеме системы Исполнитель обязан предоставить:

- Техническое задание
- Исходный код системы
- Демонстрационное видео проекта со всеми ключевыми сценариями
- Тест кейсы
- Курсовой проект
- Презентация проекта