Спецификация требований к ПО (техническое задание)

для

Система организации матчей в видеоиграх

**Версия 0.1**

**Содержание**

[1. Терминология 3](#_30j0zll)

[2. Назначение и цели создания сайта 3](#_1fob9te)

[2.1 Назначение 3](#_3znysh7)

[2.2 Объем проекта и функции продукта](#_2et92p0) 4

[3. Общее описание 4](#_tyjcwt)

[3.1 Общий взгляд на продукт 4](#_3dy6vkm)

[3.2 Классы и характеристики пользователей 4](#_1t3h5sf)

[3.3 Среда функционирования продукта 4](#_4d34og8)

[3.4 Операционная среда](#_2s8eyo1) 5

[3.5 Ограничения дизайна и реализации 5](#_17dp8vu)

[4. Функции системы 5](#_lnxbz9)

[4.1 Функции Игрока 5](#_35nkun2)

[4.2 Функции Администратора 6](#_1ksv4uv)

[4.3 Use-case диаграмма 6](#_44sinio)

[5. Требования к внешнему интерфейсу 7](#_2jxsxqh)

[5.1 Интерфейсы пользователей 7](#_z337ya)

[6. Другие нефункциональные требования 7](#_1y810tw)

[6.1 Требования к безопасности 7](#_4i7ojhp)

[6.2 Атрибуты качества ПО 7](#_1ci93xb)

# Терминология

Сайт – веб-страница в сети Интернет

Игрок – авторизованный пользователь системы

Администратор – пользователь, который может редактировать данные всей системы

Логин – уникальное имя пользователя в системе

Неавторизованный пользователь – пользователь, который не вошел в систему посредством ввода своего пароля и логина

Авторизованный пользователь – пользователь, который вошел в систему посредством ввода своего пароля и логина

Регистрация - процесс создания нового пользователя системы: запись всех обязательных данных о пользователе в базу данных системы

Матч – не начавшаяся или прошедшая партия в игру, в которую хотят сыграть или сыграли пользователи

Комната – веб-страница с информацией о матче, списке участников матча и сообщением от создателя этого матча

Слот – место которое пользователь может занять для участия в матче

Хостинг – предоставление ресурсов для корректной работы сайта

Фреймворк – рабочая среда, использующаяся для выполнения некоторых действий

База данных – систематизированные некоторым образом данные, использующиеся при работе сайта.

# Назначение и цели создания сайта

## Назначение

Эта спецификация требований к ПО описывает функциональные и нефункциональные требования к Системе Организации матчей в видеоиграх. Этот документ предназначен для команды, которая будут реализовывать и проверять корректность работы системы. Кроме специально обозначенных случаев, все указанные здесь требования имеют высокий приоритет.

## Объем проекта и функции продукта

Система организации матчей в видео играх позволит пользователям быстро находить соперников для проведения матчей в различных видеоиграх. Пользователи могут создавать матчи для других пользователей с интересующими их играми, а так же написать что-то в комнате ожидания матча, просматривать список матчей, искать матчи, фильтровать матчи по интересующим их жанрам, а также просматривать участников матча. Администратор имеет частичный доступ к базе данных.

# Общее описание

## Общий взгляд на продукт

Система организации матчей в видео играх — это система, которая позволяет объединятся по интересам.

## Классы и характеристики пользователей

|  |  |
| --- | --- |
| **Класс пользователей** | **Характеристика пользователей** |
| **Игрок** | Может заходить в личный кабинет, просматривать, искать, создавать и участвовать в матчах. |
| **Администратор** | Может редактировать список пользователей, комнат, игр и жанров. |

## Среда функционирования продукта

Хостинг осуществляется на ресурсах клиента;

Для создания приложения были использованы следующие средства и инструменты:

Python – язык программирования

Flask – фреймворк Python для создания web-приложений

Расширения Flask: flask\_sqlalchemy, flask\_migrate, flask\_bcrypt, flask\_login, flask\_admin, flask\_wtf

SQLite – база данных для хранения

SQLAlchemy – библиотека для работы с реляционными СУБД

WTForms – библиотека для работы с формами

Обоснование выбора данного инструментария:

При реализации приложения был использован фреймворк Flask по причине его легковесности, модульности, гибкости настройки и растущей популярности

Язык программирования Python был выбран для облегчения обслуживания, тестирования и модернизации приложения, а также вследствие его кроссплатформенности и поддержки большого числа библиотек и модулей

СУБД SQLite была использована вследствие ее высокой производительности, а также способности SQLite и Flask взаимодействовать с помощью SQLAlchemy, что позволяет использовать подход ORM при разработке приложения

## Операционная среда

1. Система работает со следующими Интернет-браузерами: Google Chrome (и др. на основе Chromium), Mozilla FireFox.
2. Система установлена на сервере, на котором установлено следующее ПО: http-сервер (apache или nginx), Python с модулями Flask, а также модули web-сервера для Python (cgi, wsgi, uwsgi).

## Ограничения дизайна и реализации

1. Система должна использовать базу данных SQLite.
2. Весь код HTML должен соответствовать стандарту HTML 4.0.
3. Все сценарии должны быть написаны на Python с использованием фреймворка Flask и соответствующими расширениями.

# Функции системы

Каждый пользователь имеет определенный набор функций. Данный раздел содержит полное описание функционала Системы.

## Функции Игрока

**4.1.1 Краткое описание**

Игрок может просматривать список матчей, искать их, создавать, а также участвовать в них, а также заходить в личный кабинет.

**4.1.2 Пошаговое описание**

1. Игрок авторизуется в системе.
2. Система перенаправляет Игрока на web-страницу, содержащую список активных матчей(если есть желание в нем поучаствовать есть у каждого матча отдельная кнопка), фильтр по жанрам, кнопку создания матча и поиск по названию матча.
3. После нажатия кнопки, Система отсылает Игрока на web-страницу содержащую список игроков этого матча и объявление от администратора.
4. Игрок может зайти в свой личный кабинет, где содержится информация о его профиле и список матчей, в которых он участвует.

**4.1.3 Пошаговое описание функции “создать матч”**

1. Игрок авторизуется в системе.
2. Система отсылает Игрока на страницу, содержащую список активных матчей, фильтр по играм и кнопку создания матча.
3. После нажатия кнопки создания матча, Система отсылает Игрока на Web-страницу содержащее поля для заполнения – название матча, жанр, игра, максимальное количество участников, время проведения.
4. После введения данных Система отсылает на Web-страницу ожидания матча(комната), здесь Игрок может просматривать список участников и писать для них объявления.

## Функции Администратора

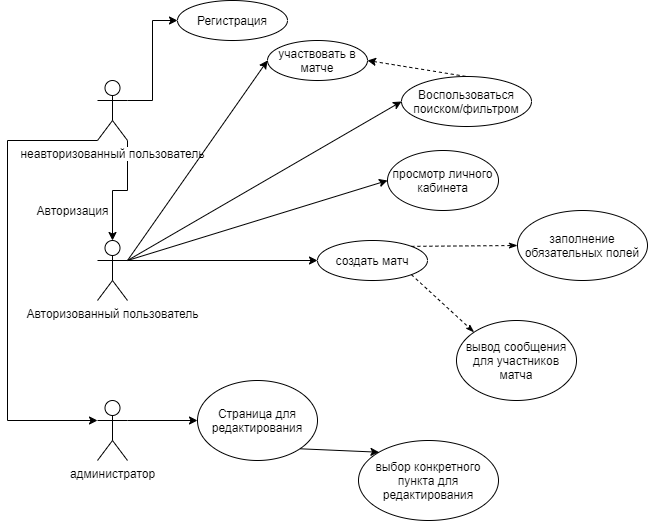
**4.2.1 Краткое описание**

Администратор имеет возможность редактировать данные системы: пользователей, матчи, игры и жанры.

**4.2.2 Пошаговое описание**

1. Администратор авторизуется в системе.
2. Система отсылает Администратора на страницу со списком редактируемых им пунктов (профили игроков, комнаты, игры и жанры).
3. Администратор выбирает из списка то, что он хочет изменить и нажимает на кнопку. Ему становится доступен список всех доступных игроков, комнат, жанров или игр в зависимости от того, что он выбрал.
4. После выбора Администратору доступна форма для заполнения тех данных, которые он хочет изменить.
5. После изменения Администратор должен нажать на кнопку, чтобы внесенные им изменения сохранились в базе данных.

## Use-case диаграмма



# Требования к внешнему интерфейсу

## Интерфейсы пользователей

1. Web-страницы должны предоставлять полную возможность навигации и выбор опций при помощи клавиатуры и мыши.

# Другие нефункциональные требования

## Требования к безопасности

1. Пользователи обязательно регистрируются для входа в Систему.
2. Только Администратор может просматривать базу данных и изменять ее.

## Атрибуты качества ПО

Система должна быть защищена от неверных действий пользователя.