МИНОБРНАУКИ РОССИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**ВЫСШЕГО ОБРАЗОВАНИЯ**

**“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”**

Факультет *компьютерных наук*

Кафедра *информационных систем*

*Система организации матчей в видеоиграх*

*Курсовой проект*

*по дисциплине*

*Технологии программирования*

*09.03.02. Информационные системы и технологии*

Преподаватель *\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.С. Тарасов* \_\_.\_\_.20\_\_

Обучающиеся \_\_\_\_\_\_\_\_\_\_\_\_\_\_ *Баженов В.С, Турченко К.А., Назаренко В.С 3 курс, д/о*

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_ *А.В. Нужных*

Воронеж2020

Оглавление:

[Введение 3](#_Toc35090276)

[Постановка задачи 4](#_Toc35090277)

[Анализ предметной области. 5](#_Toc35090278)

[Обоснование выбора средств разработки 6](#_Toc35090279)

[Диаграммы 6](#_Toc35090280)

[План тестирования 18](#_Toc35090281)

[Заключение 18](#_Toc35090282)

[Отчет о проделанной работе](#_Toc35090283) 18

****Введение****

В наше время многие пользователи онлайн-игр могут столкнуться с проблемой – нехватка соперников или товарищей в матчах. Особенно это касается пользователей мало популярных и давно вышедших игр. К сожалению, многим из них приходится ждать по несколько часов, чтобы набрать команду и, наконец, поиграть в любимую игру. Именно поэтому было принято решение создать веб-приложение, которое могло бы помочь и пользователям объединяться в команды указывать время проведения матча.

Данное приложение должно иметь логичный и понятный интерфейс, который сможет освоить любой пользователь ПК, а также определенную функциональность:

* Возможность для пользователя создать матч по интересующей его игре.
* Просмотр уже существующих матчей, и при желании участие в них.
* Фильтрация матчей по играм, по жанрам и поиск по названию.
* Возможность создателю матча оставлять сообщения общие для всех участников данного матча.

Данный курсовой проект посвящен разработке такого приложения, нацеленного на пользователей игровой индустрии, испытывающих трудности с поиском партнера по игре. Это приложение должно обладать понятным и легким интерфейсом, который будет удобен любому пользователю.

****Постановка задачи****

Цель проекта: создать систему, которая позволит организовывать матчи любых онлайн игр, в которых пользователи смогут поучаствовать.

Основание для разработки: организовать процесс создание матчей по любым онлайн играм более удобным и доступным для каждого образом, сделать платформу имеющую актуальность и интересную определенному кругу пользователей.

Приложение должно отвечать следующим требованиям:

* Выполнение основных задач приложения:
  + Добавления пользователями новых матчей;
  + Поиск созданных матчей;
  + Фильтрация по жанрам и по играм;
  + Участие в созданных матчах;
  + Возможность для создателя матча добавить сообщение для остальных игроков.
* Администрирование системы, а именно: редактирование списка пользователей, комнат, игр и жанров с помощью графического интерфейса;
* Отсутствие нагромождения и броских цветов в интерфейсе;
* Возможность перехода на все основные страницы приложения с главной страницы;
* Безопасность приложения обеспечивается обязательностью регистрации пользователя в системе. Только администратор имеет доступ к таблицам базы данных.

Требования к программному обеспечению: Система должна работать в Google Chrome 83.0.4103.97, Mozilla Firefox 76.0.1 и других браузерах.

Безопасность: Пользователи обязательно авторизуются для входа в web-приложение для дальнейшего выполнения всех операций. Сайт должен позволять только администраторам редактировать списки пользователей, игр и жанров.

Разработанный проект будет представлять собой веб-приложение, предназначенное для организации матчей в интересующую пользователя дату. Готовый сайт должен соответствовать всем поставленным выше условиям.

****Анализ предметной области.****

Разработка данного проекта будет актуальна особенно в ближайшие годы. Сейчас многие компании выпускают все новые онлайн игры или проекты нацеленные на онлайн использование, старые же игры зачастую теряют игроков, но, не смотря на все это, у многих старых онлайн игр остаются свои фанаты и те, кто еще не знаком с данной игрой тоже хотят испытать свои силы в ней. Часто в таких играх команды не успевают сформироваться из-за временных ограничений в самой игре. Пример – браузерная онлайн игра Heroes of War and Money, где максимальное время заявки на групповой бой – 30мин. На данный момент люди, любящие старые или малоизвестные проекты, могут испытывать трудности с формированием команды для игры. К сожалению, на данный момент нет платформы удовлетворяющей интересы именно этого круга людей. Существуют различные группы в соц. сетях (обычно, это сообщества игроков конкретных игр), где аудитория проявляет интерес к данной теме из-за отсутствия удобного интерфейса для формирования команд. На данный момент не существует платформы, где пользователи онлайн игр могли формировать команды для игры вместе. На основе этого анализа было принято решение о создании платформы, которая позволит пользователям искать новых товарищей для игры и вступать в команды, даже если она устарела. Данный веб-сайт будет давать возможность всем пользователям формировать группы игроков либо вступать в них. Именно поэтому он вызовет большой интерес у пользователей и будет обеспечен их постоянным вниманием.

****Обоснование выбора средств разработки****

Для реализации данного проекта был выбран язык программирования Python для облегчения обслуживания, тестирования и модернизации приложения, а также из-за его кроссплатформенности и поддержки большого числа библиотек и модулей.

Веб-фреймворк Flask был выбран в качестве основы реализуемого проекта. Основание – легковесность и гибкость, данный фреймворк не имеет жесткой структуры и позволяет выбирать модули под конкретные задачи и устанавливать их по мере необходимости.

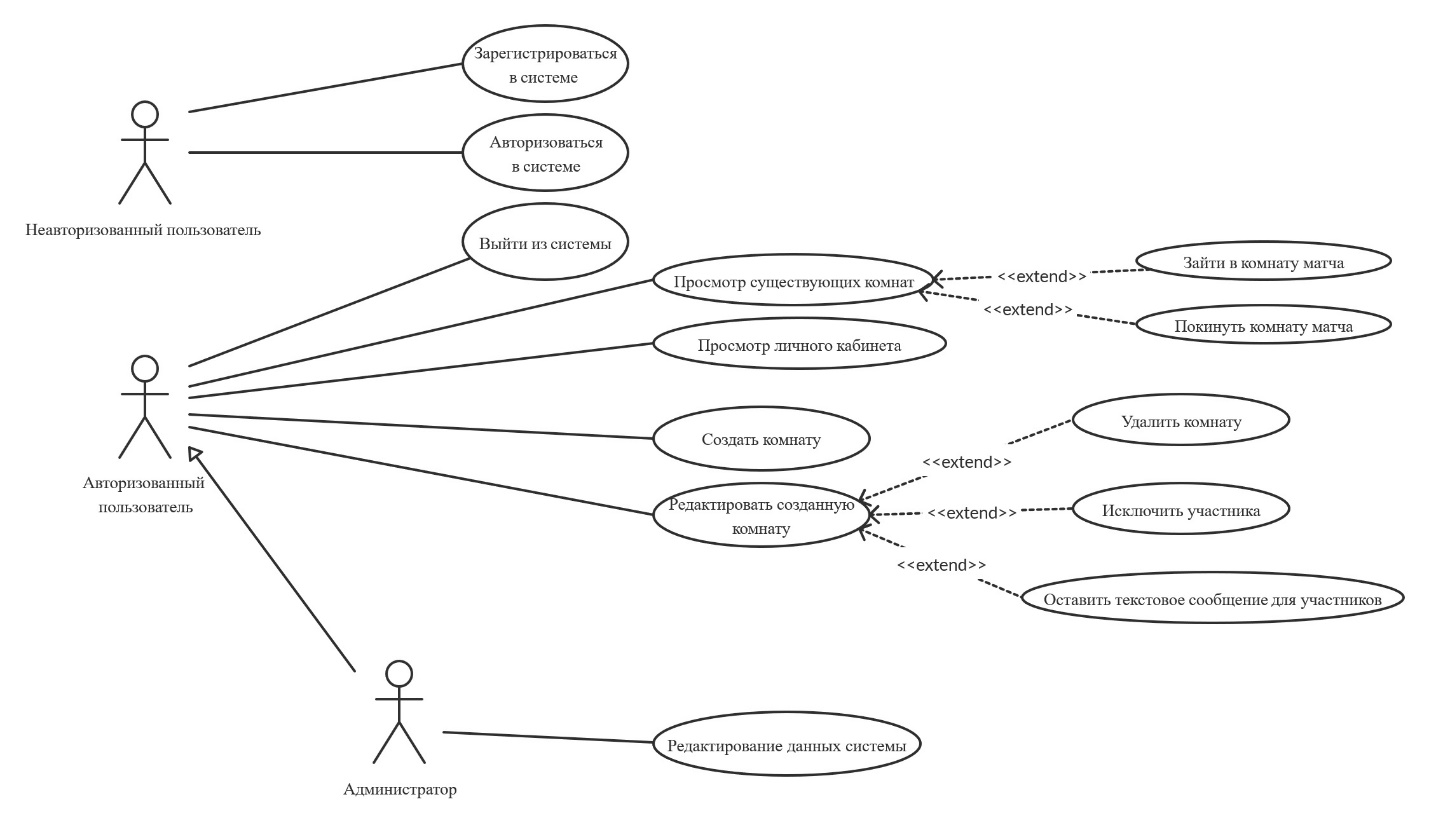
В данном проекте используется фреймворк Flask. Для организации работы базы данных был выбрана библиотека SQLAlchemy. В качестве СУБД была выбрана SQLite. СУБД SQLite была использована вследствие ее высокой производительности и простоте развертывания, а также способности SQLite и Flask взаимодействовать с помощью SQLAlchemy, что позволяет использовать подход ORM при разработке приложения.

При разработке front-end части были использованы широко распространенные HTML (в соответствии со стандартом 4.0) и CSS.

Реализация на языке Python позволила легко организовать связь back-end части (логика проекта, функционирование базы данных) с front-end частью (веб-формы HTML и стили CSS).

****Диаграммы****

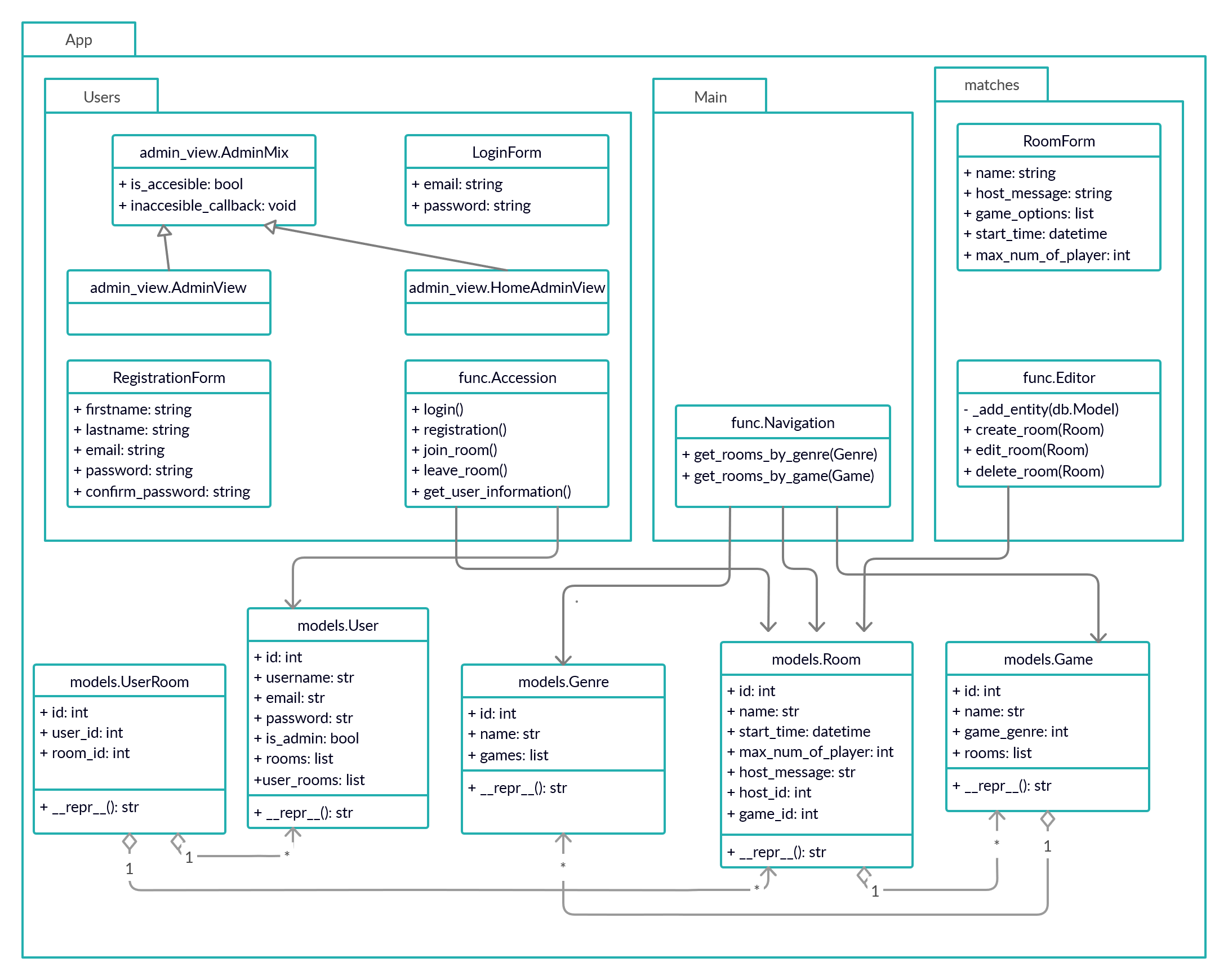
****Use-case****



**Архитектура приложения.**

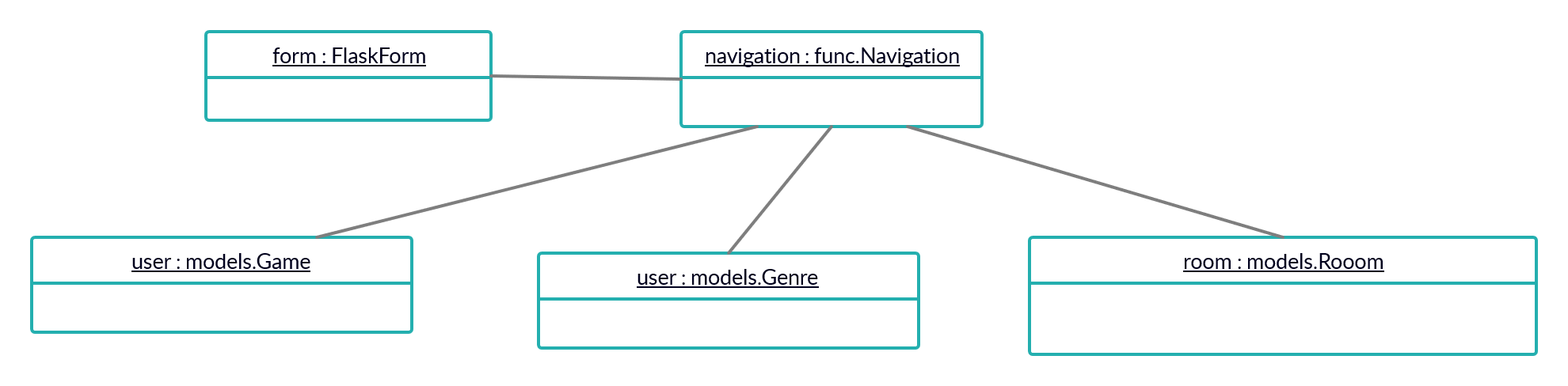
Flask использует концепцию blueprints для создания модулей приложения.  
Вся функциональность приложения разделена на 3 независимых модуля: Main, users, matches.  
Использование таких модулей позволит расширить возможности приложения в дальнейшем, зарегистрировав  
новый blueprint в приложении с новой функциональностью. И при этом структура приложения будет понятной для других разработчиков.

**Диаграмма Классов**

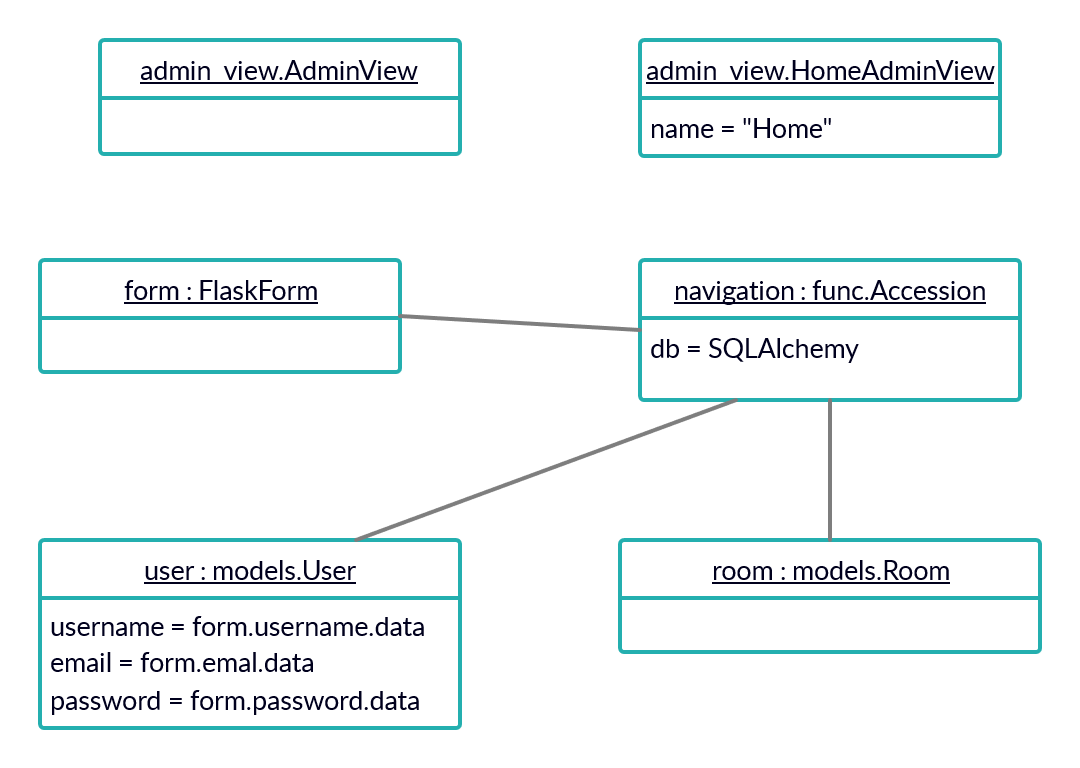


**Диаграммы объектов**

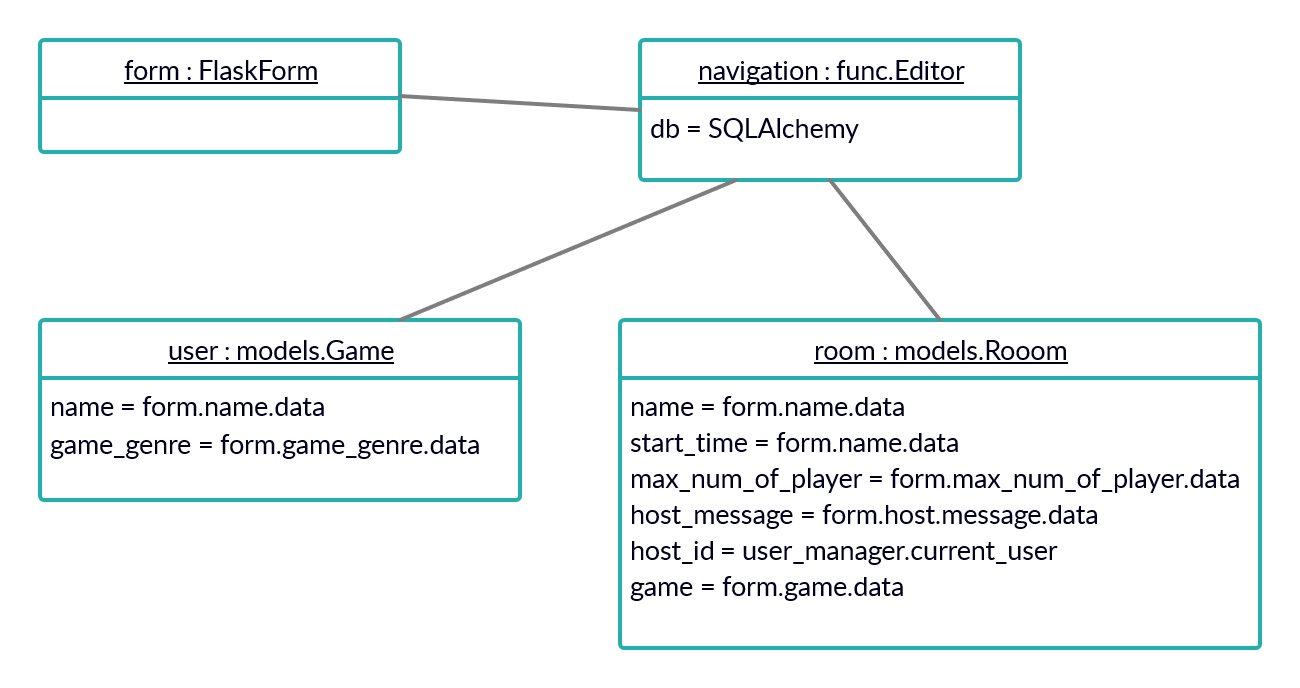
**Main**



**users**

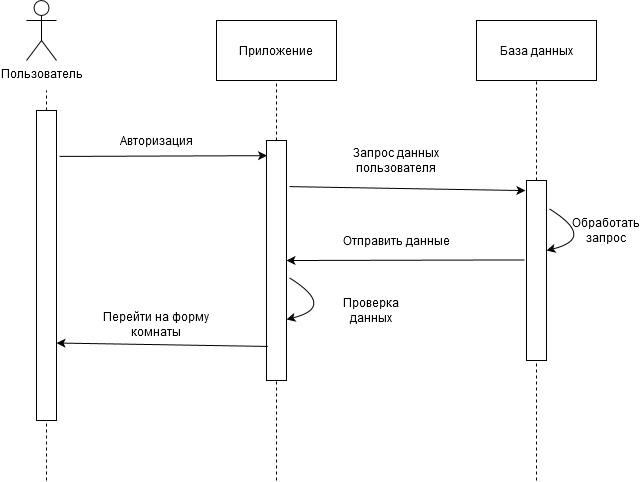


**matches**

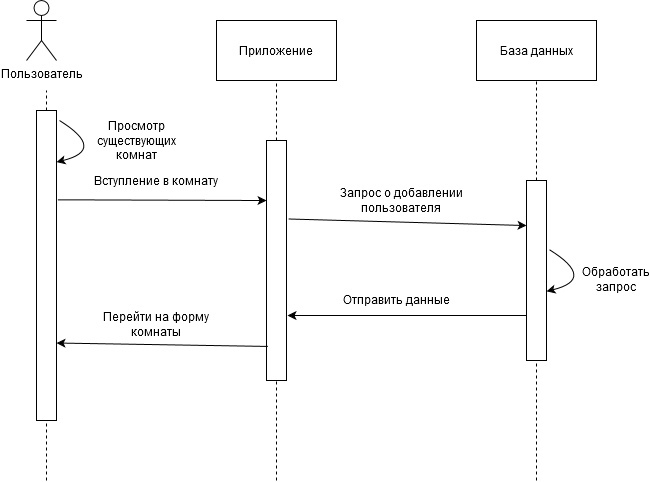


**Диаграммы последовательностей**

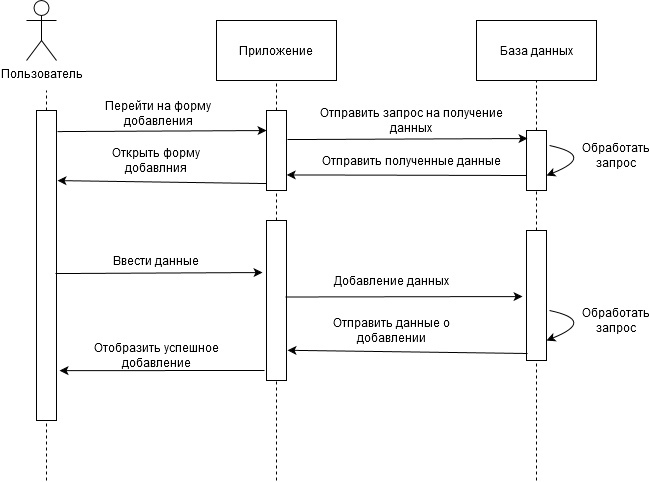
**Авторизация**



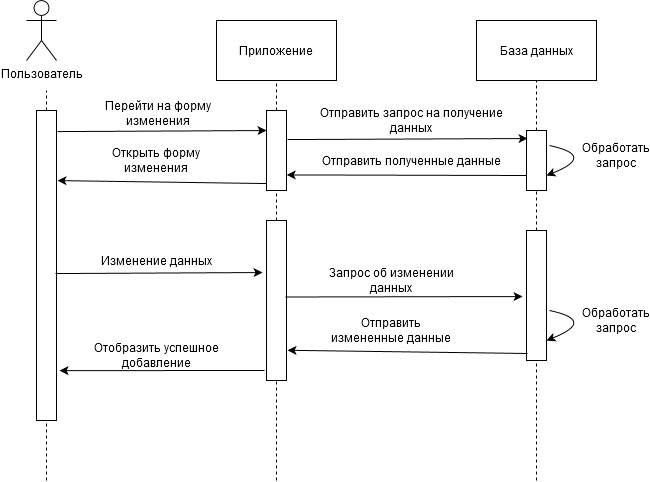
**Вступление в матч**



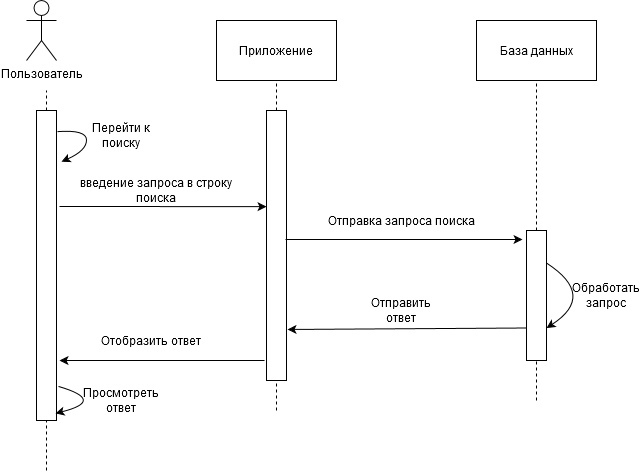
**Добавление**



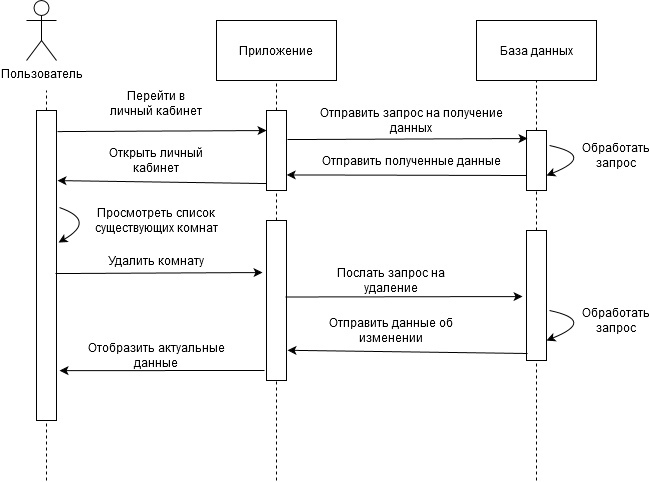
**Изменение**



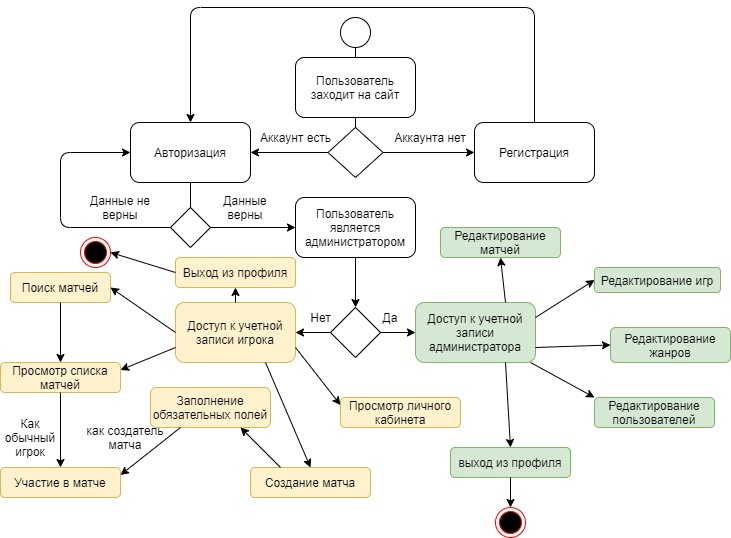
**Поиск**



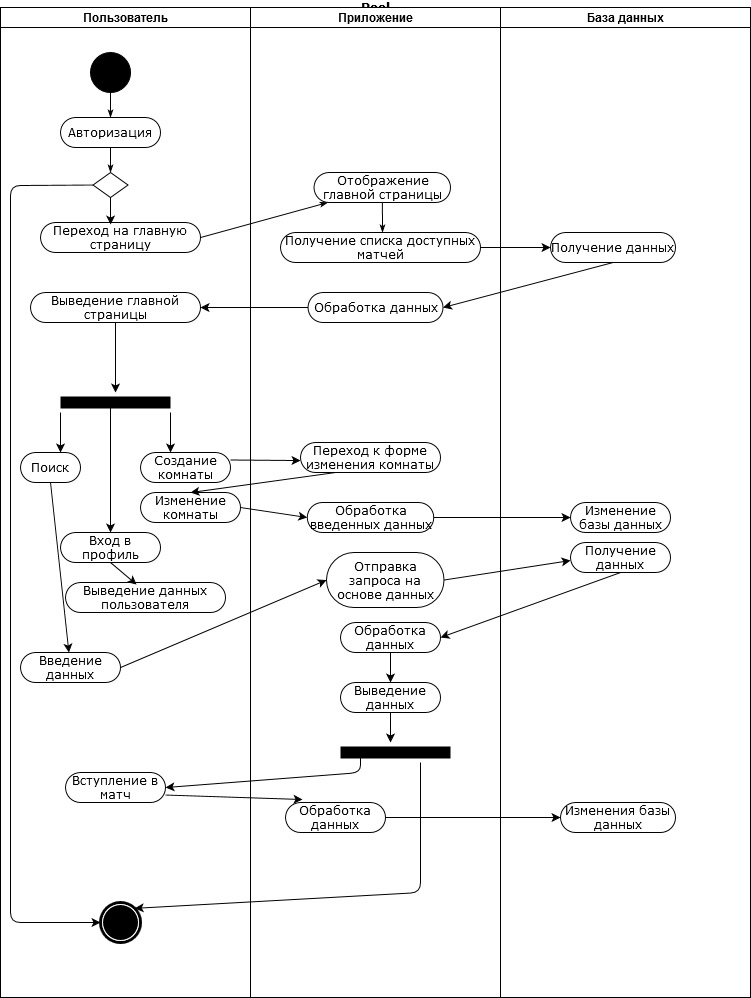
**Удаление**



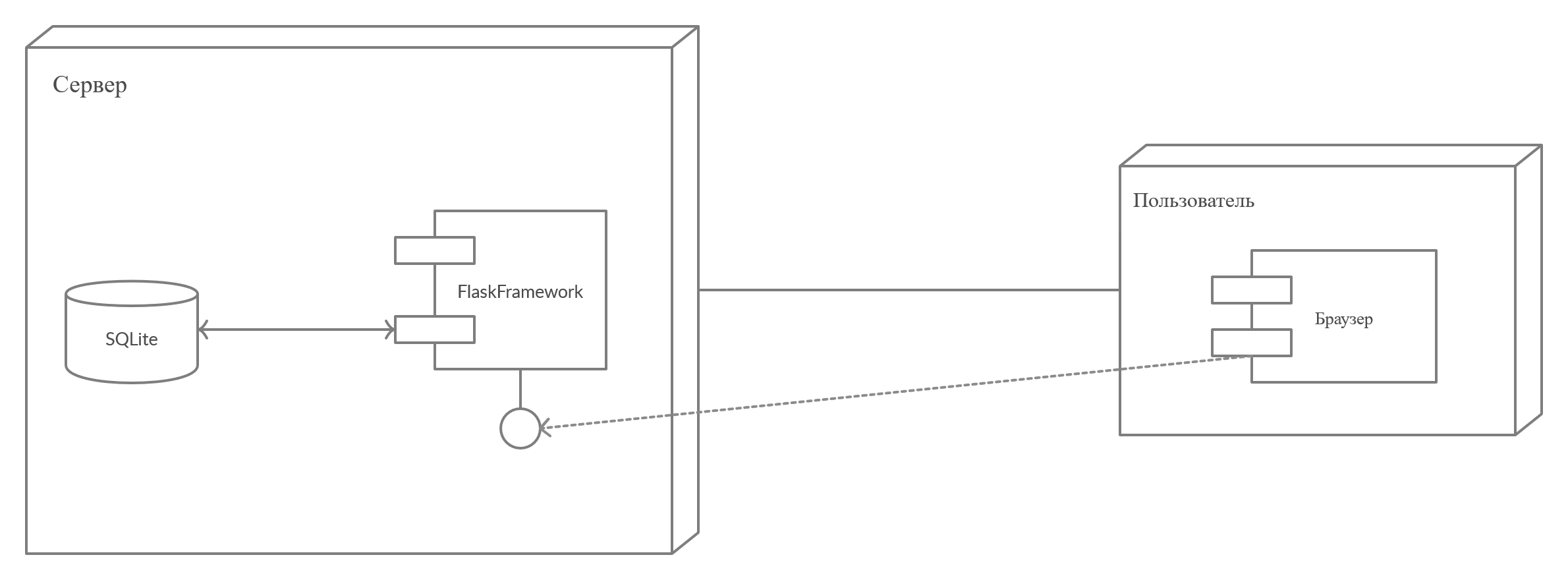
**Диаграмма состояний**



**Диаграмма активности**



**Диаграмма развертывания**



**Сценарии воронок конверсии**

Сценарий для вступления в комнату пользователя(игрок): авторизоваться -> перейти на главную страницу -> нажать кнопку «Вступить»

Сценарий для создания комнаты пользователем(игрок): авторизоваться -> перейти на главную страницу -> нажать кнопку «Создать комнату» -> нажать кнопку «Создать»

Сценарий для удаления комнаты пользователем(игрок): авторизоваться -> перейти на главную страницу -> нажать кнопку «Личный кабинет» -> нажать кнопку «Удалить»

Сценарий для отказа от участия в матче пользователем(игрок): авторизоваться -> перейти на главную страницу -> нажать кнопку «Личный кабинет» -> нажать кнопку «Покинуть»

****План тестирования****

Далее в курсовом проекте был выработан план тестирования, реализация которого приведена ниже.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Название | Предпосылка | Шаг | Ожидаемый результат | Отметка о прохождении | Комментарий |
| **1.1** | Регистрация пользователя | Пользователь не авторизован | 1. Нажатие на кнопку "Зарегистрироваться" 2. Ввод логина, почты и пароля 3. Нажатие на кнопку "Создать аккаунт" | Создание аккаунта в базе данных и перенаправление пользователя на страницу авторизации | Пройдено | Пользователь прошел регистарцию |
| **2.1** | Авторизация пользователя | Пользователь не авторизован | 1. Ввод логина и пароля 2. Нажатие на кнопку "Войти" | Вход в систему | Пройдено | Пользователь прошел авторизацию |
| **2.2** | Выход из системы | Пользователь авторизован и хочет выйти из системы | 1. Выполнение сценария 2.1  2. Пользователь нажимает кнопку "Выход" | Пользователь выходит из системы и перенаправляется на стартовую страницу авторизации | Пройдено | Пользователь вышел из системы |
| **3.1** | Вход в личный кабинет | Пользователь авторизован и хочет зайти в личный кабинет | 1. Выполнение сценария 2.1  2. Пользователь нажимает на кнопку "моя страница" | Пользователь попадает на страницу личного кабинета | Пройдено | Пользователь был переадресован на страницу личного кабинета |
| **3.2** | Регистрация в матче | Пользователь авторизован и хочет принять участие в матче | 1. Выполнение сценария 2.1 2. Пользователь выбирает понравившуюся ему комнату и нажимает кнопку "вступить" | Занесение пользователя в список участников матча и переадресация его на страницу соответствующей комнаты | Пройдено | Пользователь был переадресован на страинцу комнаты нужного ему матча и был занесен в список игроков |
| **4.1** | Поиск конкретного матча | Пользователь авторизован и хочет найти конкретный матч | 1. Выполнение сценария 2.1 2. Пользователь в строке с надписью "поиск комнаты" пишет интересующее его слово | Пользователь получает список комнат в названии которых содержится введенное им слово | Пройдено | Пользователь получил список комнат содержащих в названии искомое слово |
| **4.2** | Поиск матчей с играми определённого матча | Пользователь авторизован и хочет просмотреть матчи игр конкретного жанра | 1. Выполнение сценария 2.1 2. Пользователь выбирает интересующий его жанр из списка | Пользователь получает список комнат игр конкретного жанра | Пройдено | Пользователь получил список комнат игр конкретного жанра |
| **4.3** | Поиск матчей с определенными играми | Пользователь авторизован и хочет просмотреть матчи конкретных игр | 1. Выполнение сценария 2.1 2. Пользователь выбирает интересующую его игру из списка | Пользователь получает список комнат игр конкретной игры | Пройдено | Пользователь получил список комнат интересующей его игры |
| **5.1** | Просмотр комнаты матча в котором вы уже учавствуете | Пользователь авторизован и хочет посмотреть комнату матча в котором он учавствует | 1. Выполнение сценария 3.1  2. Пользователь нажимает кнопку "Подробнее" под интересующим его матчем в списке "Список матчей пользователя" | Пользователь попадает на страницу комнаты матча в котором он учавствует | Пройдено | Пользователь был переадресован на нужную ему комнату участником которой он является |
| **5.2** | Перестать быть участником комнаты матча | Пользователь авторизован и хочет покинуть матч, участником которого он являлся | 1. Выполнение сценария 3.1  2. Пользователь нажимает кнопку "Выйти" под интересующим его матчем | Пользователь покидает матч и получает обновленный список матчей, участником которых он является | Пройдено | Пользователь был удален из участников комнаты |
| **6.1** | Создать свой матч | Пользователь авторизован и хочет создать матч | 1. Выполнение сценария 3.1  2. Пользователь нажимает кнопку "Создать матч" 3. Пользователь заполняет поля и нажимает кнопку "сохранить" | Пользователь создает матч с введенными им данными и попадает в комнату ожжидания матча | Пройдено | Пользователь создал свою комнату матча |
| **6.2** | Просмотр комнаты матча, владельцем которой вы являетесь | Пользователь авторизован и хочет посмотреть комнату, владельцем которой он является | 1. Выполнение сценария 3.1 2. Пользователь нажимает кнопку "Подробнее" под интересующим его матчем в списке "Список созданных матчей" | Пользователь попадает на страницу комнаты матча, владельцем которой является | Пройдено | Пользователь был переадресован на нужную ему комнату владельцем которой он является |
| **6.3** | Удаление пользователя из своего матча | Пользователь авторизован и хочет удалить какого-либо пользователя из своей комнаты ожидания матча | 1. Выполнение сценария 6.2  2. Пользователь нажимает кнопку "Исключить" под интересующим его пользователем в списке "Список игроков" | Пользователь исключает игрока из своего матча | Пройдено | Пользователь удаляет игрока из своего матча и получает обновленный список игроков своего матча |
| **6.4** | Удаление своего матча | Пользователь авторизован и хочет удалить свой матч | 1. Выполнение сценария 3.1  2. Пользователь нажимает кнопку "удалить" под интересующим его матчем из списка "Список созданных матчей" | Пользователь удаляет матч, владельцем каторого он являлся | Пройдено | Пользователь удаляет матч, владельцем которого он являлся, из системы |
| **7.1** | Вход в администраторскую панель | Пользователь авторизован как администратор и попасть на страницу администраторской панели | 1. Выполнение сценария 2.1 как администратор 2. Администартор нажимает кнопку "Редактировать" | Администратор переадерсовывается на страницу администратора | Пройдено | Администратор был успешно переадресован |
| **7.2** | Создание чего-либо в базе данных через администраторскую панель | Пользователь авторизован как администратор и хочет добавить запись в базу данных системы с помощью администраторской панели | 1. Выполнение сценария 7.1  2. Администратор нажимает на кнопку интересующей его таблицы  3. Администратор получает список из нужной ему таблицы и нажимает кнопку "create" 4. Администратор заполняет поля и нажимает кнопку "Save" | Администратор вносит запись в базу данных | Пройдено | Администратор добавил новую забись в базу данных |
| **7.3** | Редактирование чего-либо в базе данных | Пользователь авторизован как администратор и хочет изменить запись в базе данных системы с помощью администраторской панели | 1. Выполнение сценария 7.1  2. Администратор нажимает на кнопку интересующей его таблицы  3. Администратор получает список из нужной ему таблицы и нажимает на иконку карандаша напротив интересующей его записи 4. Администратор редактирует поля и нажимает кнопку "Save" | Администратор изменияет запись в базе данных | Пройдено | Администратор обновил уже существующую запись в базе данных |
| **7.4** | Удаление чего-либо из базы данных | Пользователь авторизован и хочет изменить свой матч | 1. Выполнение сценария 7.1  2. Администратор нажимает на кнопку интересующей его таблицы  3. Администратор получает список из нужной ему таблицы и нажимает на иконку урны напротив интересующей его записи 4. Администратор подтверждает действие на странице | Администратор удаляет запись из базы данных | Пройдено | Администратор удалил ранее существующую запись из базе данных |
| **1.1.1.** | Негативный сценарий регистрации пользователя | Ввод уже существующего логина | 1. Ввод уже существующего логина 2. Нажатие на кнопку подтверждения | Предупреждение пользователя о том что он ввел уже существующий логин | Провалено | Сообщение об ошибке не высветилось |
| **1.1.2.** | Негативный сценарий регистрации пользователя | Ввод уже существующей почты | 1. Ввод почты, которая уже указывалась для регистрации ранее 2. Нажатие на кнопку подтверждения | Предупреждение пользователя о том что он ввел уже существующую почту | Провалено | Сообщение об ошибке не высветилось |
| **2.1.1.** | Негативный сценарий авторизации пользователя | Пользователь вводит некорректные логин или пароль | 1. Ввод некорректных данных2. Нажатие на кнопку подтверждения | Предупреждение пользователя об ошибке посредством сообщения | Провалено | Сообщение об ошибке высветилось |
| **0.0.0.** | Попытка перейти на URL, который не соответствует выбранной роли | Пользователь ввел URL, не соответсвующий его роли | 1. Выполнение сценария 2.1 пользователем 2. Ввод URL, соотвествующего шаблону /admin | Пользователь возвращается на свою домашнюю страницу | Пройдено | Перенаправления на страницу Администратора не произошло |
| **0.0.0.** | Попытка перейти на несуществующий URL | Пользователь ввел URL, не предусмотренный приложением | 1. Выполнение сценария 2.1 пользователем 2. Ввод URL , не предусмотренного приложением | Пользователь перенаправляется на страницу ошибки 404 | Пройдено | Высвечивается страница 404 |

****Заключение****

Заявленный в техническом задании проект был полностью реализован командой из трех участников. Он соответствует следующим требованиям:

1. Система помогает пользователям объединяться для участия в созданных ими матчах.
2. Система позволяет пользователям самим создавать матчи по интересующим их играм.
3. Система может привлечь пользователей возможностью поиграть в старые игры, которые им нравятся.

Данное приложение полностью удовлетворяет требованиям, излагаемым в техническом задании. Заданный функционал отлажен и работает.

Пользователь может заходить в личный кабинет, просматривать, искать, создавать и участвовать в матчах. Администратор же может редактировать список пользователей, комнат, игр и жанров.

В итоге удалось реализовать заявленное веб-приложение, а также отладить и оптимизировать его работу.

****Отчет о проделанной работе****

Техническое задание составили: Турченко Константин, Назаренко Виктория, Баженов Вадим.

Составление диаграмм между участниками команды было распределено следующим образом:

* Юз кейс диаграмма – Турченко Константин
* Диаграмма последовательностей – Назаренко Виктория
* Диаграмма классов - Турченко Константин
* Диаграмма состояний – Баженов Вадим
* Диаграмма объектов – Турченко Константин
* Диаграмма активностей – Назаренко Виктория
* Диаграмма развертывания – Турченко Константин

Работа над приложением:

* Front-end – Назаренко Виктория
* Back-end – Турченко Константин
* Тестирование – Баженов Вадим
* Составление сценариев воронок конверсии – Назаренко Виктория
* Подключение аналитики – Турченко Константин
* Размещение на хостинге – Турченко Константин