

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

Факультет Компьютерных Наук
Кафедра информационных систем

Техническое задание
на разработку веб-приложения
«TripToVoronezh»

Исполнители

_____ Д.В. Свиридов

_____ А.Ю. Кунахова

Заказчик

_____ В.С. Тарасов

Воронеж 2024

СОДЕРЖАНИЕ

СОДЕРЖАНИЕ	2
1 Терминология	4
2 Общие сведения.....	6
2.1 Полное наименование системы и название сайта.....	6
2.2 Наименование исполнителя и заказчика системы	6
2.2.1 Наименование заказчика	6
2.2.2 Наименование исполнителя	6
2.2.3 Перечень документов, на основании которых создается система	7
2.3 Состав и содержание работ по созданию приложения.....	7
2.4 Порядок контроля и приемки автоматизированной системы.....	7
2.5 Назначение и цели создания автоматизированной системы	9
2.5.1 Назначение сайта.....	9
2.5.2 Цели создания сайта.....	9
2.5.3 Задачи, решаемые при помощи сайта	9
3 Требования к автоматизированной системе	11
3.1 Требования к архитектуре сайта.....	11
3.2 Общие требования к оформлению и вёрстке страниц.....	11
3.3 Требования к численности и квалификации персонала, обслуживающего сайт.....	12
3.4 Требования к системе администрирования	13
4 Структура сайта.....	14
4.1 Динамические страницы.....	14

5	Языковые версии сайта.....	15
6	Группы пользователей.....	16
7	Дизайн приложения	17
8	Навигация по сайту	18
8.1	Описание динамических страниц.....	18
8.1.1	Главная страница.....	18
8.1.2	Страница профиля.....	18
8.1.3	Страница доступных гидов	18
8.1.4	Страница сохранённых гидов	18
8.1.5	Страница поиска и фильтрации	19
8.1.6	Страница аутентификации	19
8.1.7	Страница регистрации	19
9	Функциональные возможности приложения	20
10	Порядок контроля и приемки работ	22
	Реквизиты и подписи сторон	23
	Приложение	24

1 Терминология

Проект, система – разрабатываемое веб-приложение.

Веб-приложение — клиент-серверное приложение, в котором клиент взаимодействует с веб-сервером при помощи браузера

Фреймворк – программная платформа, определяющая структуру программной системы; программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта.

Пользователь – авторизованный в системе человек, пользующийся функционалом веб-приложения.

Личный кабинет пользователя – веб-страница, где отражаются персональная информация пользователя, введенная на этапе регистрации и/или отредактированная позже, а также переход к странице редактирования личных данных.

Администратор – пользователь, обслуживающий веб-приложение, осуществляющий связь с клиентами и данные которого были заранее занесены в систему под отличительным доменом.

Back-end – программно–аппаратная часть веб-приложения. Отвечает за функционирование внутренней части веб-приложения.

Front-end – клиентская часть приложения. Отвечает за получение информации с программно–аппаратной части и отображение ее на устройстве пользователя.

Сервер, серверная часть – компьютер, обслуживающий другие компьютеры и предоставляющий им свои ресурсы для выполнения определенных задач.

Клиентская сторона – компьютер, использующий ресурсы сервера и предоставляющий пользователю возможность взаимодействия с системой.

GitHub – веб-сервис для хостинга IT-проектов и их совместной разработки.

REST API (REST) – стиль архитектуры программного обеспечения для построения масштабируемых веб-приложений.

Java — это высокоуровневый язык программирования общего назначения с строгой статической типизацией и автоматическим управлением памятью посредством сборщика мусора.

Spring — это мощный Java веб-фреймворк, который позволяет создавать надежные, безопасные и поддерживаемые веб-приложения с высокой скоростью разработки.

Hibernate — это мощный Java ORM-фреймворк, который позволяет эффективно работать с базами данных, создавая надежные и поддерживаемые приложения с высокой скоростью разработки.

Система управления базами данных (СУБД) – это набор программ, которые управляют структурой БД и контролируют доступ к данным, хранящимся в БД.

PostgreSQL – это реляционная СУБД с открытым кодом.

JavaScript – это язык программирования, предназначенный для front-end-разработки и использующийся для взаимодействия с пользователем.

CSS – формальный язык описания внешнего вида веб-страницы, написанного с использованием языка разметки (HTML).

React – JavaScript-библиотека для создания пользовательских интерфейсов.

HTML – стандартизированный язык разметки для просмотра веб-страниц в браузере.

2 Общие сведения

2.1 Полное наименование системы и название сайта

Полное наименование: «Платформа для создания персонализированных путеводителей по городу Воронеж и Воронежской области на основе интересов пользователя».

Наименование приложения: «TripToVoronezh».

2.2 Наименование исполнителя и заказчика системы

2.2.1 Наименование заказчика

Заказчик: Старший преподаватель Тарасов Вячеслав Сергеевич. Воронежский Государственный Университет, Факультет компьютерных наук, кафедра Программирования и Информационных Технологий.

2.2.2 Наименование исполнителя

Исполнитель: «5» команда группы «3»

Состав команды исполнителя:

- Свиридов Данила Владимирович, студент. Воронежский Государственный Университет, Факультет компьютерных наук, кафедра Информационных Систем.
- Кунахова Алина Юрьевна, студент. Воронежский Государственный Университет, Факультет компьютерных наук, кафедра Информационных Систем.

2.2.3 Перечень документов, на основании которых создается система

- Федеральный закон "О персональных данных" от 27.07.2006 N 152–ФЗ;
- Закон РФ от 07.02.1992 N 2300-1 (ред. от 11.06.2021) "О защите прав потребителей".

2.3 Состав и содержание работ по созданию приложения

Состав и содержание работ по созданию приложения включают в себя следующие этапы:

- Сбор необходимой информации, постановка целей, задач системы, которые в будущем должны быть реализованы 16.02.24 – 01.03.24.
- Анализ предметной области, анализ конкурентов и построение структуры требований, ведущих к решению поставленных задач и целей 01.03.24 – 13.03.24.
- Построение модели программы, описание спецификаций данных, определение связей между сущностями, разработка модели БД 13.03.24 – 01.04.24.
- Разработка рабочего проекта, состоящего из написания кода, отладки и корректировки кода программы 16.04.24 – 01.05.24.
- Проведение тестирования программного обеспечения 16.05.24 – 01.06.24.

2.4 Порядок контроля и приемки автоматизированной системы

Предварительные отчёты по работе будет проводиться во время рубежных аттестаций:

1 аттестация (конец марта 2024) - создан репозиторий проекта на GitHub, распределены задачи проекта в таск-менеджере Trello, создан проект Miro с общей логикой системы, предоставлены промежуточные результаты по курсовому проекту и готовое техническое задание;

2 аттестация (конец апреля 2024) - написана основополагающая часть кода приложения, реализована БД и ее взаимодействие с сервером, проведена отладка и доработка кода, проведено тестирование по работе системы;

3 аттестация (конец мая 2024) - разработан курсовой проект, выполнены завершающие работы по доработке приложения, предоставлена готовая система.

2.5 Назначение и цели создания автоматизированной системы

2.5.1 Назначение сайта

Предоставление возможности пользователям создавать персонализированные путеводители по городу Воронеж и Воронежской области на основе их интересов и предпочтений, а также содействие туристической активности в регионе путем обеспечения информации о достопримечательностях, маршрутах и местах отдыха.

2.5.2 Цели создания сайта

Целями создания сайта являются:

- Обеспечение пользователей удобным и богатым функционалом для интуитивно понятного планирования и организации своего досуга на территории Воронежа и его области;
- Продвижение туристического потенциала Воронежского региона;
- Предоставление персонализированных рекомендаций и советов пользователям с учётом их интересов и предпочтений.

2.5.3 Задачи, решаемые при помощи сайта

- Систематизация и обновление информации о местах для посещения;
- Обеспечение возможности пользователям ознакомиться с широким спектром досуга (рестораны, музеи, кинотеатры, памятники и др);
- Обеспечение возможности пользователям создавать персонализированные путеводители на основе большого количества фильтров;

- Обеспечение возможности пользователям добавлять интересные для посещения места и гиды в «Мои гиды»;
- Реализация алгоритмов рекомендаций, предлагающих пользователям новые места;
- Разработка удобного и интуитивно понятного интерфейса для пользователей.

3 Требования к автоматизированной системе

3.1 Требования к архитектуре сайта

Сайт должен иметь архитектуру, соответствующую модели клиент-серверного взаимодействия на основе REST API.

Для реализации серверной части сайта будут использоваться следующие средства:

- Язык программирования Java.
- Веб-фреймворк Spring Boot Framework.
- Библиотека Hibernate.
- ПО Docker.
- СУБД PostgreSQL.

Для реализации клиентской части сайта будут использоваться следующие средства:

- Язык гипертекстовой разметки HTML.
- Формальный язык описания внешнего вида документа CSS.
- Язык программирования JavaScript.
- Библиотека React.

3.2 Общие требования к оформлению и вёрстке страниц

Сайт должен быть оформлен в одной цветовой палитре с использованием ограниченного набора шрифтов. У страниц сайта должен быть единый стиль. В оформлении сайта должно присутствовать разработанное название.

Необходимо корректное и одинаковое отображение страниц сайта в следующих браузерах:

- Yandex Browser, версия 23.3.7.24 или новее;
- Google Chrome, версия 118.0.5993.71 или новее;
- Mozilla Firefox, версия 115 и 115 ESR или новее;
- Microsoft Edge, версия 88.0.705.68 или новее;
- Safari, версия 16.5.2.

Функции, выполняемые сайтом:

- Получение информации о новых местах Воронежа и области.
- Регистрация и аутентификация пользователей;
- Возможность просмотра событий и интересных мероприятий, планируемых на конкретные даты;
- Обновление тематических подборок интересных мест;
- Добавление пользователем путеводителей в «Мои гиды»;

3.3 Требования к численности и квалификации персонала, обслуживающего сайт

Сайт должен обслуживаться и поддерживаться как минимум 1 администратором, ознакомленным с правилами работы с сайтом. Он обязан отвечать за добавление, удаление, просматривание или редактирование данных через веб-интерфейс, настраивание прав пользователей.

3.4 Требования к системе администрирования

Система администрирования сайта создается при помощи фреймворка Spring Security. Это мощный инструмент управления, который позволяет добавлять, удалять, просматривать или редактировать любую модель базы данных через веб-интерфейс, т.е. управлять данными по путеводителям.

4 Структура сайта

Web-приложение должно содержать следующие страницы/виды страниц:

4.1 Динамические страницы

- Главная страница;
- Страница профиля;
- Страница доступных гидов;
- Страница сохранённых гидов;
- Страница поиска и фильтрации;
- Страница аутентификации;
- Страница регистрации.

5 Языковые версии сайта

Все страницы сайта должны быть реализованы с поддержкой русской языковой версии.

6 Группы пользователей

Приложение содержит следующие группы пользователей:

- Неавторизованные пользователи;
- Авторизованные пользователи.

7 Дизайн приложения

- Все страницы должны быть выполнены в едином стиле, преимущественно в светлых и тёмных тонах зелёного цвета;
- Все страницы должны корректно отображаться на экранах любых устройств;
- Верстка страниц должна соответствовать ТЗ;
- Все элементы управления сайтом должны быть чётко видны.

8 Навигация по сайту

8.1 Описание динамических страниц

8.1.1 Главная страница

На главной странице пользователь увидит доступный функционал сервиса, оценит его основные возможности и преимущества.

Также здесь будут размещены изображения или слайды с основными достопримечательностями города Воронежа и области.

8.1.2 Страница профиля

Здесь пользователь сможет просмотреть свой профиль, включая личные данные, предпочтения и историю действий.

Также на этой странице будет возможность редактирования профиля.

8.1.3 Страница доступных гидов

Пользователи смогут просмотреть все актуальные для последней версии веб-приложения гиды и выбрать для просмотра их по категориям интересов, интерфейсу и другим параметрам.

8.1.4 Страница сохранённых гидов

Здесь пользователи увидят гиды, которые ранее были добавлены в данную подборку для последующего повторного просмотра.

8.1.5 Страница поиска и фильтрации

На странице поиска и фильтрации пользователи могут найти интересующие их места и путеводители по заданным критериям, таким как категория, тип, расположение и т. д.

8.1.6 Страница аутентификации

Страница входа для зарегистрированных пользователей, где они могут ввести свои учетные данные (логин и пароль) для входа на сайт.

8.1.7 Страница регистрации

Здесь новые пользователи могут заполнить форму регистрации, введя свои личные данные и создав учетную запись на сайте.

9 Функциональные возможности приложения

Для неавторизованного пользователя:

— Регистрация и авторизация в системе:

Пользователь может создать новую учетную запись

Для авторизованного пользователя:

— Возможность выхода из учетной записи:

Пользователь может безопасно выйти из своего аккаунта.

— Редактирование профиля:

Пользователь может обновить информацию о себе и изменить настройки своего профиля.

— Создание персонализированных путеводителей:

Возможность выбирать персонализированные маршруты и путеводители, учитывая интересы пользователя.

— Добавление мест в избранное:

Пользователь может добавлять понравившиеся места и события в список избранного для быстрого доступа.

— Поиск мест и событий:

Возможность выполнять поиск интересующих мест и событий по различным критериям.

Для администратора (управляет сайтом через сервер):

— Управление новостями:

Создание, редактирование и удаление новостей о событиях и мероприятиях города и области.

— Управление пользователями:

Просмотр, добавление, редактирование и удаление пользовательских аккаунтов.

— Управление местами и событиями:

Добавление, редактирование и удаление информации о местах, достопримечательностях и событиях.

— Управление категориями и тегами:

Создание и редактирование категорий и тегов для классификации мест и событий.

— Управление настройками сайта:

Изменение основных настроек сайта, таких как язык, тема оформления и другие параметры.

10 Порядок контроля и приемки работ

Приложение предоставляется для проверки во время занятий по предмету и рубежных аттестаций. После передачи готового сайта заказчику, встречи с преподавателем-практиком, а также обсуждения текущего этапа разработки с заказчиком, осуществится приемка работ на защите проекта. После окончания работ приложение с документацией будет представлено заказчику в назначенные сроки. Вся документация о системе будет доступна в печатном и электронном виде, а также храниться на платформе GitHub.

Реквизиты и подписи сторон

ЗАКАЗЧИК:

Ст. преп. Тарасов В.С.

_____/_____/

« ____ » _____ 20 ____ г.

ИСПОЛНИТЕЛЬ:

Разработчик Свиридов Д.В.

_____/_____/

Разработчик Кунахова А.Ю.

_____/_____/

« ____ » _____ 20 ____ г.

Приложение

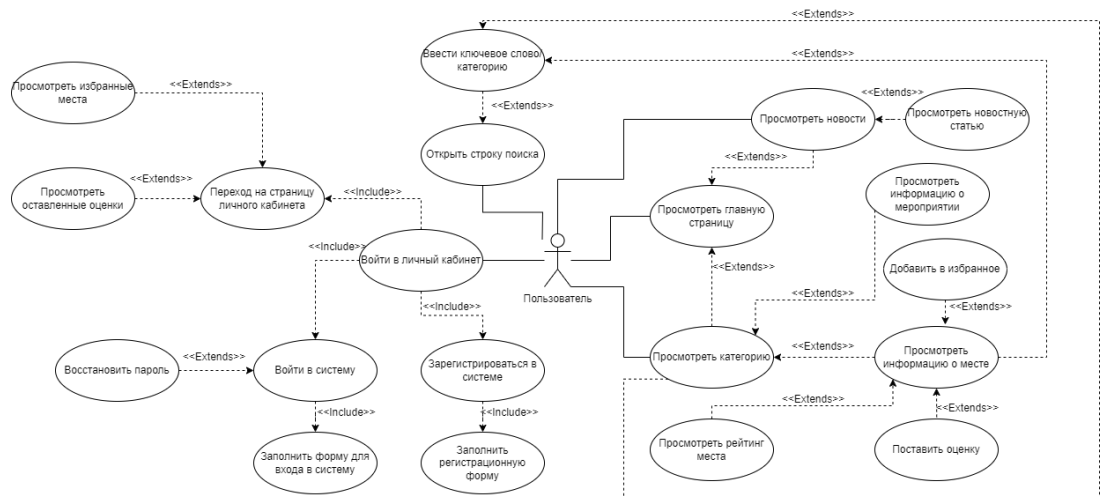


Рисунок 1 - Диаграмма Use-Case со стороны Клиента

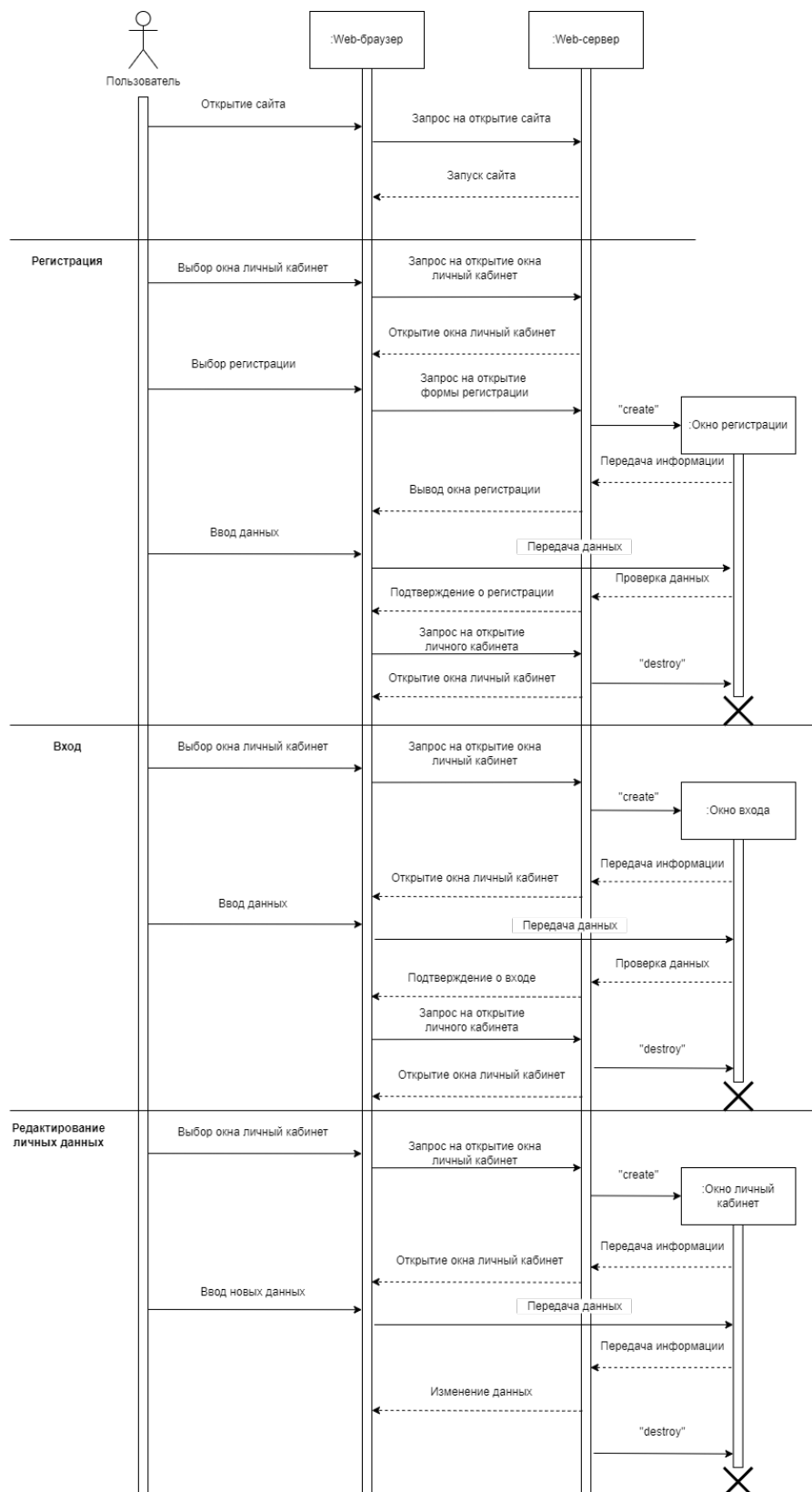


Рисунок 2 - Диаграмма последовательностей

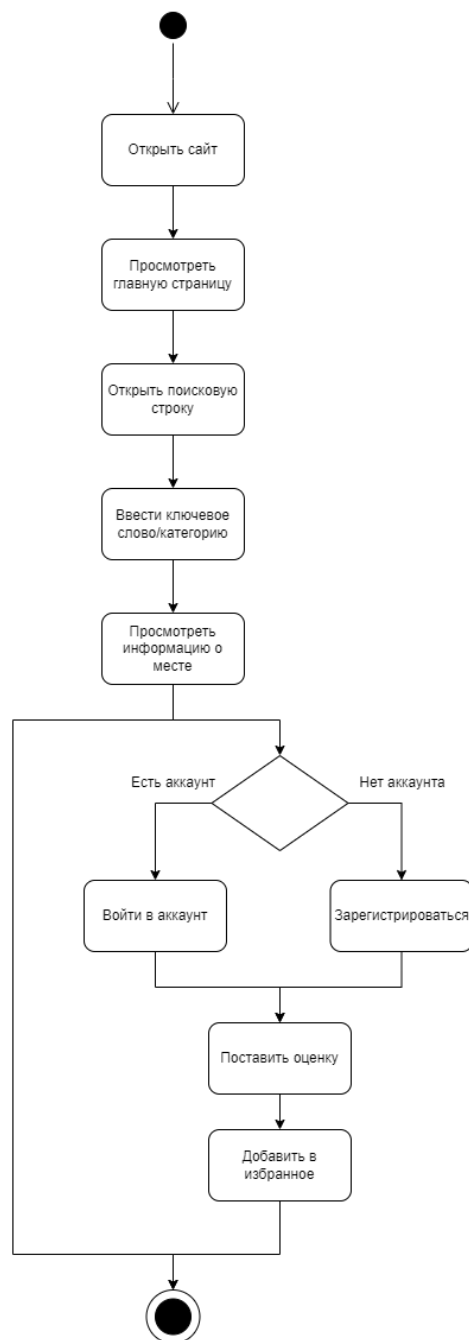


Рисунок 3 - Диаграмма активности

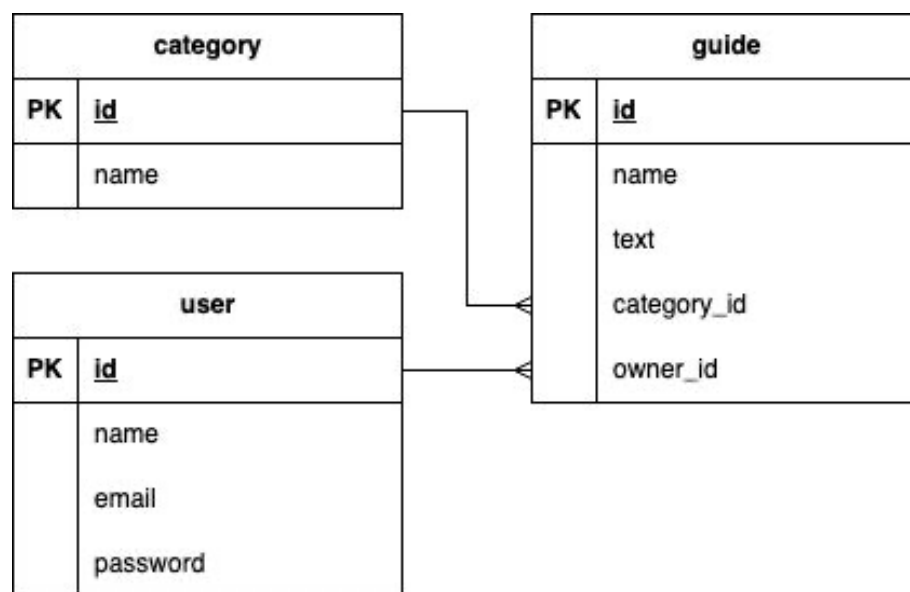


Рисунок 4 - ER-диаграмма

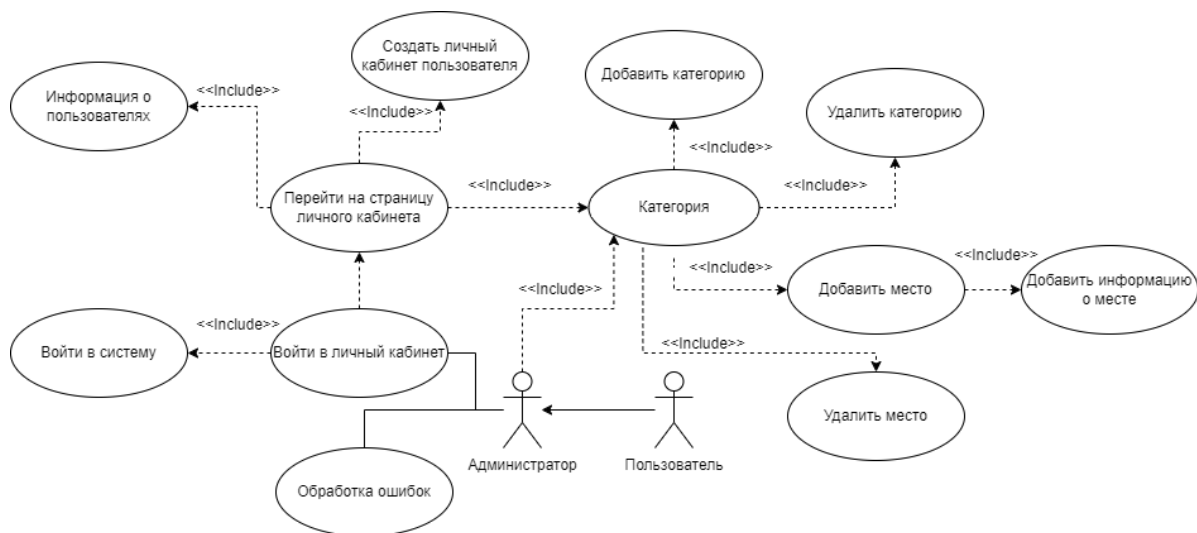


Рисунок 5 - Диаграмма Use-Case со стороны администратора