|  |  |
| --- | --- |
| Изображение выглядит как эмблема, герб, нашивка, символ  Автоматически созданное описание | **Министерство науки и высшего образования Российской Федерации**  **Калужский филиал федерального государственного бюджетного образовательного учреждения высшего образования**  **«Московский государственный технический университет  имени Н.Э. Баумана  (национальный исследовательский университет)»**  **(КФ МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ ИНФОРМАТИКА И УПРАВЛЕНИЕ

КАФЕДРА СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К КУРСОВОМУ ПРОЕКТУ   
НА ТЕМУ:***

*TravelGuide – ваш гид по Калужской области*

|  |  |  |  |
| --- | --- | --- | --- |
| Студент группы ИУК5-52Б |  |  | Яцков А. |
|  | (подпись, дата) |  | (И.О. Фамилия) |
|  |  |  |  |
| Руководитель курсового проекта |  |  | Федоров В.О. |
|  | (подпись, дата) |  | (И.О. Фамилия) |

|  |
| --- |
| **Министерство науки и высшего образования Российской Федерации**  **Калужский филиал федерального государственного бюджетного образовательного учреждения высшего образования**  **«Московский государственный технический университет имени Н.Э. Баумана  (национальный исследовательский университет)»**  **(КФ МГТУ им. Н.Э. Баумана)** |

УТВЕРЖДАЮ

Заведующий кафедрой **\_\_ИУК5\_\_\_**

\_\_\_\_\_\_\_\_\_\_\_ (Е. В. Вершинин)

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г.

**ЗАДАНИЕ**

**на выполнение курсового проекта**

по дисциплине ***Модели данных***

Студент группы ***ИУК5-52Б Яцков Александр***

(фамилия, имя, отчество)

Тема курсового проекта ***TravelGuide – ваш гид по Калужской области***

Направленность КП ***учебный***

Источник тематики ***кафедра ИУК5***

***Задание***

*Разработать сайт для помощи в подборе достопримечательностей Калужской области.*

***Оформление курсового проекта***

*Расчетно-пояснительная записка на\_\_\_\_\_\_\_\_ листах формата А4.*

*Перечень графического материала КП (плакаты, схемы, чертежи и т. п.):*

*− Концептуальная схема – 1 лист формата А3;*

*− Логическая схема – 1 лист формата А3;*

Дата выдачи задания «\_\_\_» \_\_\_\_\_\_\_\_\_\_2024 г.

|  |  |  |  |
| --- | --- | --- | --- |
| Руководитель |  |  | В.О. Федоров |
|  | (подпись, дата) |  | (И.О. Фамилия) |
| Студент |  |  | А. Яцков |
|  | (подпись, дата) |  | (И.О. Фамилия) |

|  |
| --- |
| **Министерство науки и высшего образования Российской Федерации**  **Калужский филиал федерального государственного бюджетного образовательного учреждения высшего образования**  **«Московский государственный технический университет имени Н. Э. Баумана  (национальный исследовательский университет)»**  **(КФ МГТУ им. Н.Э. Баумана)** |

**КАЛЕНДАРНЫЙ ПЛАН**

**на выполнение курсового проекта**

по дисциплине ***Модели данных***

Студент группы ***ИУК5-52Б Яцков Александр***

(фамилия, имя, отчество)

Тема курсового проекта ***TravelGuide – ваш гид по Калужской области***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Наименование этапов | Сроки выполнения этапов | | Отметка о выполнении | |
| план | факт | Руководитель | Куратор |
| 1 | Задание на выполнение | 1-я нед. |  |  |  |
| 2 | Выполнение основных расчетов и эскизов графической части | 10-я нед. |  |  |  |
| 3 | Выполнение и окончательное оформление графической части и расчетно-пояснительной записки | 14-я нед. |  |  |  |
| 4 | Защита | 17-я нед. |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Студент |  | Руководитель |  |
|  | (подпись, дата) |  | (подпись, дата) |

Содержание

1. Техническое задание ……………………………………………………………5
2. Научно-исследовательская часть …………………………………………...… 8

2.1 Введение ……………………………………………………………………. 8

2.2 Постановка задачи проектирования …...…………………………………..8

2.3 Описание предметной области ……………….………………………….....8

2.4 Анализ аналогов ……………………………………………..………...…. 10

2.5 Требования к системе ……………………………………………….……..11

2.6 Обоснование выбора инструментов и платформы для разработки..…....13

1. Проектно-конструкторская часть…………………………………………….. 14  
   3.1. Логическая схема базы данных …………………………………………. 14  
   3.2. Разработка архитектуры приложения……………………………….……15  
   3.3. Наполнение базы данных …………………………………………….…..16  
   3.4. Взаимодействие с базой данных …………………………………….….. 19
2. Проектно-технологическая часть …………………………………………… 22  
   4.1. Порядок развертывания системы ………………………..………………22  
   4.2. Руководство пользователя ……………………………….……………….22

Заключение …………………………………………………………….……………. 26

Список используемых источников…………………………………………………. 27

**1.Техническое задание**

1.1.1 Название проекта: TravelGuide – Ваш гид по Калужской области

**1.1.2 Наименование организации-заказчика, разработчика системы Заказчик:** КФ МГТУ им. Н.Э. Баумана Разработчик: Студент КФ МГТУ им. Н.Э. Баумана группы ИУК5-52Б Яцков А.

1.1.3 Плановые сроки начала и окончания работы по созданию системы:

Плановый срок начала работ - 5 сентября 2024 года  
Плановый срок окончания работ - 1 декабря 2024 года

**1.2 Назначение и цели создания веб-приложения.** 1.2.1 TravelGuide предназначено для туристических агентств, организовывающих путешествия по Калуге и области и для людей, которые в поиске новых красот Калужской области.  
 1.2.2 Цель создания веб-приложения:  
- увеличить количество пользователей туристического агентства за счет веб-приложения TravelGuide в течение первого года.

**1.2.3 Объект автоматизации.**  
 Процесс подбора новых туристических мест, исключая все посещенные ранее места. В настоящее время люди не могут сразу сориентироваться куда можно поехать в свободное время и что посмотреть, тратят свое время на поиск новых парков, выставок, галерей и т.д. Куда проще посмотреть отзывы других людей в веб-приложении и выбрать что тебе по душе.

**5**

**1.3.1 Требования к системе в целом.**

1.3.1.1 Требования к эргономике и технической эстетике интерфейс веб-приложения TravelGuide должен быть интуитивно понятным, не перегруженным и не вводить пользователя в заблуждение. Все элементы интерфейса должны соответствовать их функционалу и пояснению.  
1.3.2 Требования к функциям (задачам), выполняемых веб-приложением:

**Функции для регистрации пользователей:** - простая и удобная форма регистрации (возможность регистрации через электронную почту; минимальный набор обязательных данных).  
 - безопасная система авторизации (шифрование данных пользователей).

**1.3.3 Требования к видам обеспечения.** 1.3.3.1 Требования к лингвистическому обеспечению веб-приложения Средством описания предметной области, так же, как и средством взаимодействия пользователя с веб-приложением является русский язык.

**1.3.3.2 Требования к входным и выходным данным.**  
 Входные данные: данные для регистрации (имя пользователя, email, пароль); данные для авторизации.  
 Выходные данные: регистрация (сообщение об успешной регистрации).

1.4 Состав и содержание работ по созданию (развитию) веб-приложения  
 Утверждение темы, задания на разработку, технического задания. Оформление ТЗ. Описание предметной области и требований к системе. Концептуальная схема.

**6**

Прототип или скетчи интерфейса.  
Оформление введения и исследовательской части. Обоснование выбора БД, логическая схема БД, физическая схема данных. Демонстрация работы макета системы с БД.  
Оформление проектно-конструкторской части. Демонстрация работающего веб-сайта. Презентация. Тестирование и отладка веб-сайта. Разработка эксплуатационной документации.  
Завершающее оформление документации согласно требованиям ГОСТ. Все ошибки и проблемы устранены. Подготовка доклада. Защита курсовой работы.

**Порядок контроля и приёмки веб-приложения.**  
 Веб-приложение разрабатывается с применением тестирования работоспособности функционала. При добавлении новых функций предыдущие тесты должны сохранять работоспособность. Приемка работы осуществляется комиссией КФ МГТУ им. Н.Э. Баумана кафедры ИУК5

**Требования к документированию.**

- Техническое задание в соответствии с ГОСТ 34.602-89  
 - Расчетно-пояснительную записку, включающую в исследовательскую часть, проектно-конструкторскую часть и проектно-технологическую часть, включающую в себя руководство пользователя и руководство программиста (администратора). Расчетно-пояснительная записка выполняется с учетом требований, предусмотренных ГОСТ 7.32-2001 и 2.105-95

**Источники разработки.**

**-** Технические спецификации и обучающий материал.

- Библиотеки и фреймворки для разработки пользовательского интерфейса.

**7**

**2.Научно-исследовательская часть**

**2.1 Введение**

Калужская область, расположенная в центральной части России, является уникальным регионом, богатым историческими памятниками, архитектурными ансамблями, природными достопримечательностями и культурным наследием. Это место привлекает внимание туристов и местных жителей, которые стремятся познакомиться с его уникальной атмосферой и увидеть самые интересные объекты. Однако, несмотря на обилие достопримечательностей, многие из них остаются малоизвестными, а информация о них разбросана и труднодоступна.

Современные технологии предоставляют возможность упростить поиск и планирование путешествий, сделав процесс более удобным и информативным. Одной из таких технологий является создание информационных систем, которые объединяют данные о туристических объектах, предоставляют возможность оставлять отзывы, а также помогают пользователям быстро находить нужную информацию.

**2.2 Постановка задачи проектирования**

Задачей проектирования данной курсовой работы является реализация веб-сервиса для поиска и выбора калужских достопримечательностей.

**2.3 Описание предметной области**

TravelGuide – это информационная система, предоставляющая пользователям возможность получать актуальную и разностороннюю информацию о туристических объектах, достопримечательностях, развлечениях и сервисах в Калужской области.

**8**

**Целевая аудитория:**- Туристы, планирующие путешествие по Калужской области.  
- Жители, интересующиеся достопримечательностями и развлечениями в нашем регионе.  
- Гиды, экскурсоводы, владельцы туристических объектов и сервисов, желающие привлечь внимание к своим предложениям.

**Основные задачи системы.**Предоставление информации:  
 - описания достопримечательностей, музеев, парков, памятников, архитектурных ансамблей и других туристических объектов.  
Интерактивность и социальность:   
 - отзывы пользователей о туристических объектах.

**Модель данных.**

Для реализации проекта необходимо разработать модель данных, включающую следующие сущности:

- Отзыв (текст отзыва).

- Событие (название, дата проведения)

- Администраторы (имя, электронная почта, пароль)

- Пользователи (ФИО, электронная почта, пароль).

- Достопримечательность (название, описание, фотография, адрес, контактная информация).

**9**

TravelGuide поможет раскрыть красоту и богатство Калужской области для широкой аудитории, сделать путешествия более удобными и интересными.

**2.4. Анализ аналогичных решений**

После проведения анализа аналогов было выявлено 2 аналога.

Сервисы подбора достопримечательностей, такие как **«**TripAdvisor» и «Яндекс.Путешествия». Однако **«**TripAdvisor» показывает достопримечательности всей Европы и различные предложения из других источников, которые довольно сильно мешают конкретному выбору определенно места для посещения близлежащих красот:

Изображение выглядит как текст, кровать, мебель, в помещении

Автоматически созданное описание

Рисунок 1 – Пример публикации сайта «TripAdvisor»

В том числе, кроме достопримечательностей в поиске могут попасться такие виды развлечений, как прогулки на круизных лайнерах, аренда автомобилей и т.д., что имеет мало отношения к поиску знаковых мест.

**10**

Второй аналог это скорее помощник в передвижении к уже выбранным достопримечательностям, чем помощник в их выборе. Там вы сможете выбрать даты поездки, транспорт или даже купить готовый тур.

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок 2 – Список возможностей выбора «Яндекс.Путешествия»

**2.5. Требования к системе**

- **Эргономика и интерфейс**: интерфейс должен быть интуитивно понятным, не перегруженным и соответствовать функционалу.

-  **Функции для регистрации пользователей**: простая форма регистрации, возможность авторизации с помощью электронной почты.

- **Интерактивность**: возможность оставлять отзывы.

**11**

Изображение выглядит как текст, диаграмма, снимок экрана, План

Автоматически созданное описание

Рисунок 3 – Концептуальная схема

Чтобы воспользоваться системой пользователь должен авторизоваться или зарегистрироваться, введя ФИО, почту и пароль.

После входа в систему пользователь может приступить к просмотру и дальнейшему выбору достопримечательности по его вкусу. Для этого он может просмотреть все представленные места в списке и перейти на заинтересовавшее. Далее он может ознакомиться более подробно и узнать местонахождение культурного объекта. У системы должны быть администраторы. Они могут следить за отзывами, событиями и общей работоспособностью сайта.

**12**

**2.6. Обоснование выбора инструментов и платформы для разработки**

Для разработки данного веб-приложения будет использоваться платформа, основанная на технологиях Flask. Этот выбор обоснован рядом ключевых преимуществ, которые обеспечивают оптимальное сочетание гибкости, производительности и удобства разработки. В контексте бэкенда, Flask предоставляет легковесный и масштабируемый фреймворк с простым синтаксисом, что упрощает создание эффективных АРІ. Flask позволит создать эффективное взаимодействие между серверной и клиентской частями приложения. Такой подход обеспечит четкое разделение обязанностей, упрощая сопровождение и масштабирование проекта в будущем. Экосистема инструментов обеспечит обширные возможности для расширения функциональности при необходимости. Таким образом, выбор данной платформы обеспечивает оптимальное сочетание простоты, гибкости и производительности.

**13**

**3. Проектно-конструкторская часть**

**3.1 Логическая схема базы данных**

**Изображение выглядит как текст, диаграмма, снимок экрана, Параллельный

Автоматически созданное описание**

Рисунок 4 – Логическая схема

**14**

**3.2 Разработка архитектуры приложения.**

**Изображение выглядит как диаграмма, снимок экрана, линия, текст

Автоматически созданное описание**

Рисунок 5 – Архитектура приложения

Для разработки данного приложения используется клиент-серверная архитектура.

Архитектура клиент-сервер - это распределенная архитектура, в которой приложение разделяется на две основные компоненты: клиент и сервер, Клиент - это часть приложения, выполняющаяся на стороне пользователя, а сервер - это часть, выполняющаяся на удаленной машине, предоставляющей какие-то услуги или ресурсы.

Клиент:

1. Интерфейс: клиентская часть обеспечивает пользовательский интерфейс, с которым взаимодействует пользователь. Это может быть графический интерфейс, веб-страница и т.д.

2. Запросы к серверу: клиент отправляет запросы на сервер для получения данных, обновлений или выполнения каких-то операций.

Сервер:

1. Бизнес-логика: сервер содержит бизнес-логику приложения. Это программный компонент, отвечающий за обработку запросов, выполнение необходимых операций, и возврат результатов клиенту.

**15**

2. Ответы на запросы: сервер принимает запросы от клиента, обрабатывает их, и отправляет обратно необходимую информацию или результат выполненной операции.

Взаимодействие между клиентом и сервером:

1. Установление соединения: клиент устанавливает соединение с сервером, используя определенный протокол (например, HTTP, WebSocket, TCP/IP).

2. Отправка запроса: клиент отправляет запрос серверу, содержащий информацию о том, что нужно сделать (например, получить данные, отправить данные, выполнить определенное действие).

3. Обработка запроса на сервере: сервер принимает запрос, а затем обрабатывает его, взаимодействуя с базой данных, выполняя вычисления и т.д.

4. Отправка ответа: после обработки запроса сервер отправляет ответ обратно клиенту. Ответ может содержать запрошенные данные или подтверждение выполнения операции.

5. Обработка ответа на клиенте: клиент принимает ответ и обрабатывает полученные данные, обновляя пользовательский интерфейс или предоставляя необходимую информацию пользователю. Этот процесс повторяется при каждом взаимодействии между клиентом и сервером.

**3.3 Наполнение базы данных**

Прежде всего пользователю необходимо зарегистрироваться, его данные запишутся в базу данных и в дальнейшем ему нужно будет авторизироваться по указанным данным.

**16**

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 6 – Регистрация

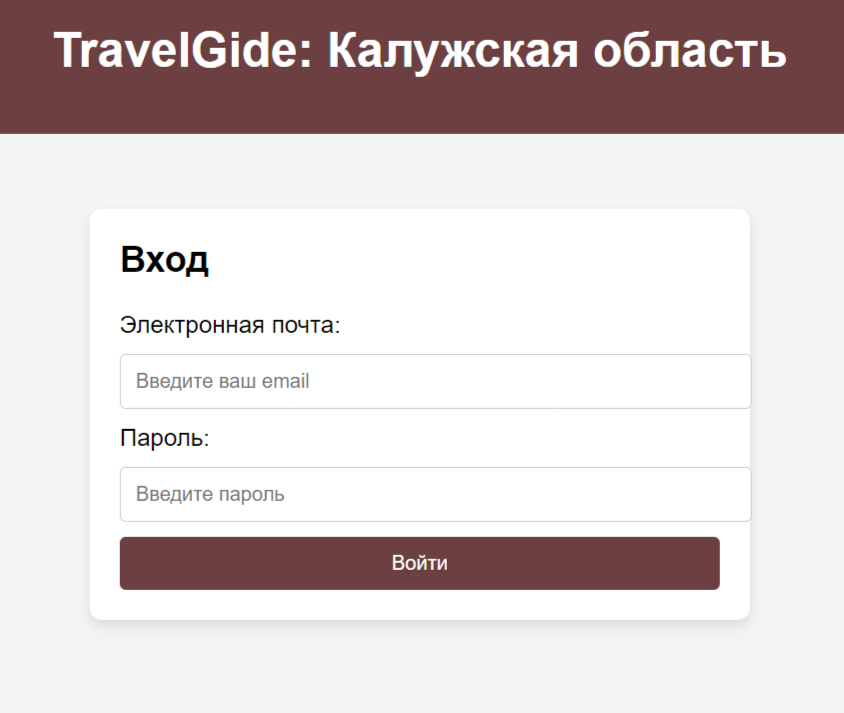


Рисунок 7 – Авторизация

**17**

После успешной регистрации пользователь попадает на основную страницу сайта (Рисунок 8).

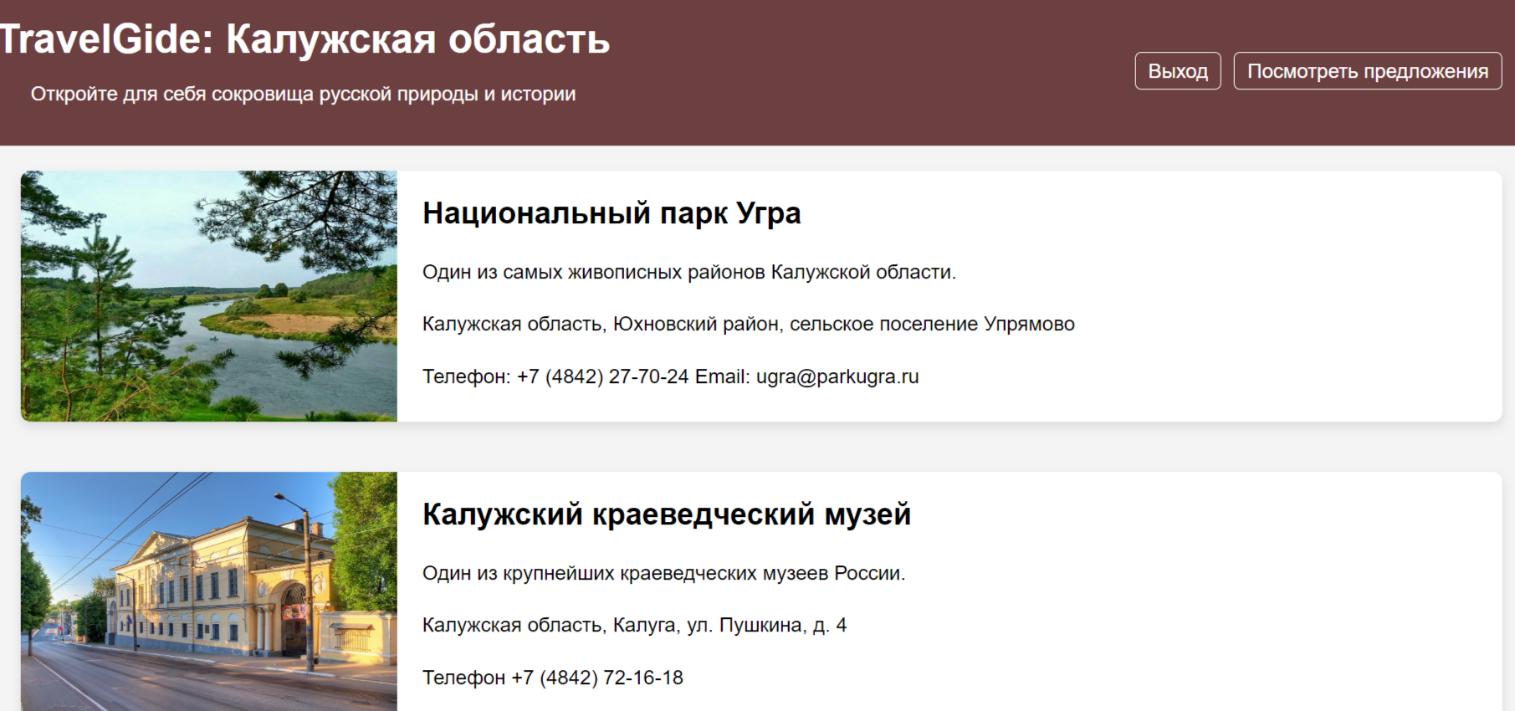


Рисунок 8 – Основная страница

После регистрации пользователю доступно предложить свои достопримечательности (Рисунок 8), а администратору доступно просмотреть предложенные пользователями достопримечательности (Рисунок 9).

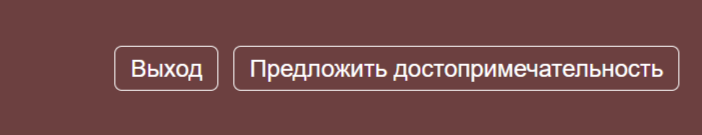


Рисунок 9 – Возможности пользователя

**18**

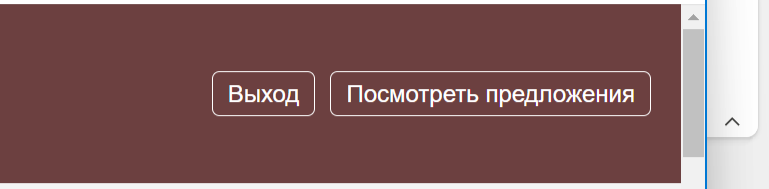


Рисунок 10 – Возможности администратора

**3.4 Взаимодействие с базой данных**

Для управления базой данных и хранения информации используется DB Browser SQLite в качестве инструмента администрирования. DB SQLite представляет собой реляционную систему управления базами данных (СУБД).

Реляционная база данных (РБД) — это система управления базами данных (СУБД), которая организует данные в виде таблиц (называемых отношениями), где каждая таблица состоит из строк (записей) и столбцов (атрибутов). Реляционная база данных является одной из самых популярных и широко используемых систем управления базами данных в мире. Она используется в самых разных сферах, от небольших веб-приложений до крупных корпоративных систем. Вот примеры компаний, организаций и проектов, которые используют реляционные базы данных:

* **Google:** хранит данные о пользователях, поисковых запросах и рекламных кампаниях в реляционных базах данных.
* **Microsoft:** использует реляционные базы данных для управления данными в Office 365, Azure и других продуктах.
* **Apple:** хранит данные о пользователях, приложениях и устройствах в реляционных базах данных.

**19**

То есть реляционные базы данных применимы везде, где важнейшим приоритетом является целостность и безопасность данных и где встречаются высокоструктурированные данные, соответствующие строгой, предсказуемой и предопределенной схеме.

Класс Sight описывает достопримечательности. Конструктор этого класса используется для создания объектов, представляющих достопримечательности.

class Sight(db.Model):

id = db.Column(db.Integer, primary\_key=True)

name = db.Column(db.String(100), nullable=False)

info = db.Column(db.Text, nullable=False)

adress = db.Column(db.Text)

contac\_info = db.Column(db.Text)

dop\_info = db.Column(db.Text)

 В Python конструктор класса — это метод \_\_init\_\_. Однако в данном случае явный конструктор не определен, поэтому используется конструктор по умолчанию, предоставляемый базовым классом db.Model.

Основные методы объекта запроса «query», которые часто используются:

- Filter\_by() - добавляет условия фильтрации к запросу

- AIl() - возвращает все результаты запроса в виде списка

- First - возвращает первый результат запроса

- Join() - объединение таблиц

**20**

Примеры использования методов на основе регистрации и авторизации с результатом отрисовки на клиентской части:

Регистрация. При обращении к соответствующему маршруту приложение данные о новом пользователе. Из полей «E-mail» и «Пароль» извлекаются эти данные. При вводе почты проверяется, чтобы в базе данных не было пользователя с такой почтой, в противном случае возвращается ошибка. В результате этой операции создается пользователь с указанной почтой и паролем. Данный пользователь добавляется в базу данных и база данных фиксирует изменения.

Авторизация. Программа смотрит, чтобы были введены оба поля. По введенному адресу почты происходит проверка наличия ее в базе данных. Если пользователь существует, то устанавливается сессия для этого пользователя. В случае успешной регистрации или авторизации пользователь заходит в систему. Если неверно указаны данные, то появится ошибка и потребуется заново заполнить поля.

**21**

**4.Проектно-технологическая часть**

**4.1. Порядок развертывания системы**

Процесс наполнения базы данных описан в пункте 3.3 с примерами пользовательских форм.

Развертывание системы состоит из следующих этапов:

1. Запуск базы данных: в приложении DB Browser (SQLite).

2. Запуск серверной части.

Данная система тестировалась в Yandex версии 24.10.4.756 (64-bit).

**4.2. Руководство пользователя**

После загрузки сайта вы окажетесь на главной странице, где отображаются достопримечательности Калужской области. В верхней части страницы находится шапка с названием сайта и кнопками для регистрации и входа (Рисунок 11).

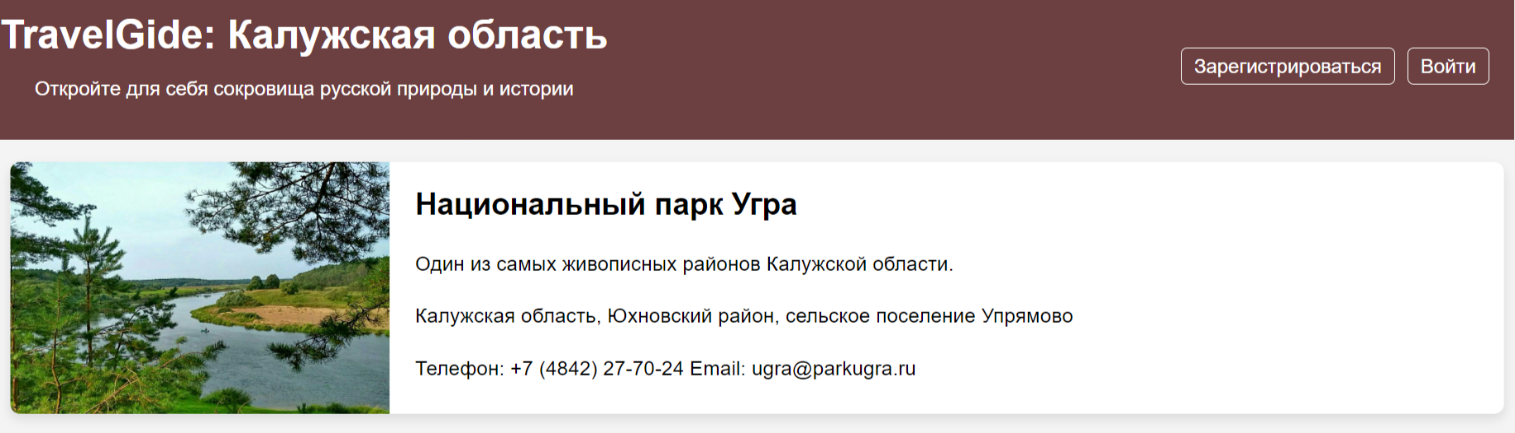


Рисунок 11 – Начальная страница

Если вы новый пользователь, нажмите на кнопку **"Зарегистрироваться"** в шапке сайта. В открывшейся форме заполните следующие поля (Рисунок 6):

**22**

**-** ФИО: введите ваше полное имя.

- Email: укажите ваш адрес электронной почты.

- Пароль: придумайте и введите пароль.

Нажмите кнопку **"Зарегистрироваться"**, чтобы завершить регистрацию.

Если вы уже зарегистрированы, нажмите на кнопку **"Войти"** в шапке сайта. В открывшейся форме введите ваш Email и Пароль (Рисунок 7). Нажмите кнопку **"Войти"**, чтобы войти в систему. После входа в систему в шапке сайта появится кнопка **"Выход"**. Нажмите на неё, чтобы выйти из аккаунта.

На главной странице отображаются все достопримечательности Калужской области. Каждая достопримечательность представлена в виде карточки с изображением, названием, описанием, адресом и контактной информацией. Нажмите на карточку, чтобы перейти к подробной информации о достопримечательности (Рисунок 12).

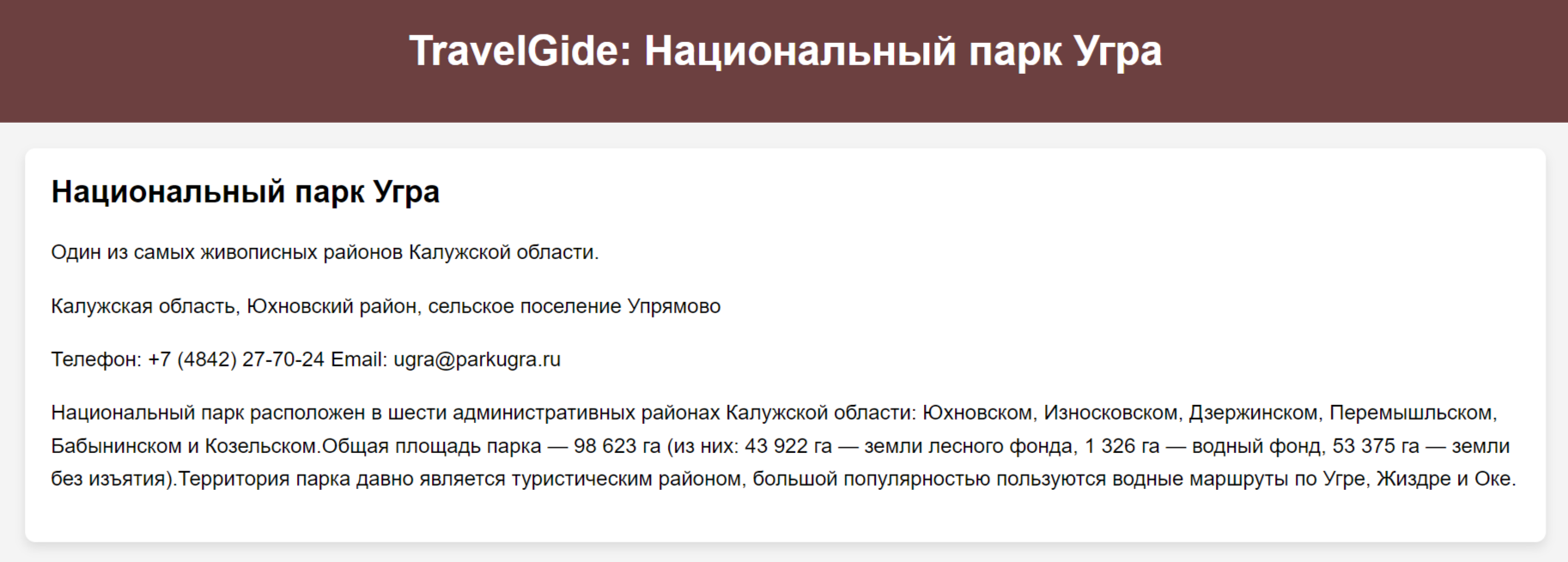
****

Рисунок 12 – информация о достопримечательности

На странице деталей достопримечательности вы увидите подробную информацию о месте, включая описание, адрес, контакты и дополнительную информацию. Если вы авторизованы, вы можете оставить отзыв о достопримечательности (Рисунок 13).

**23**

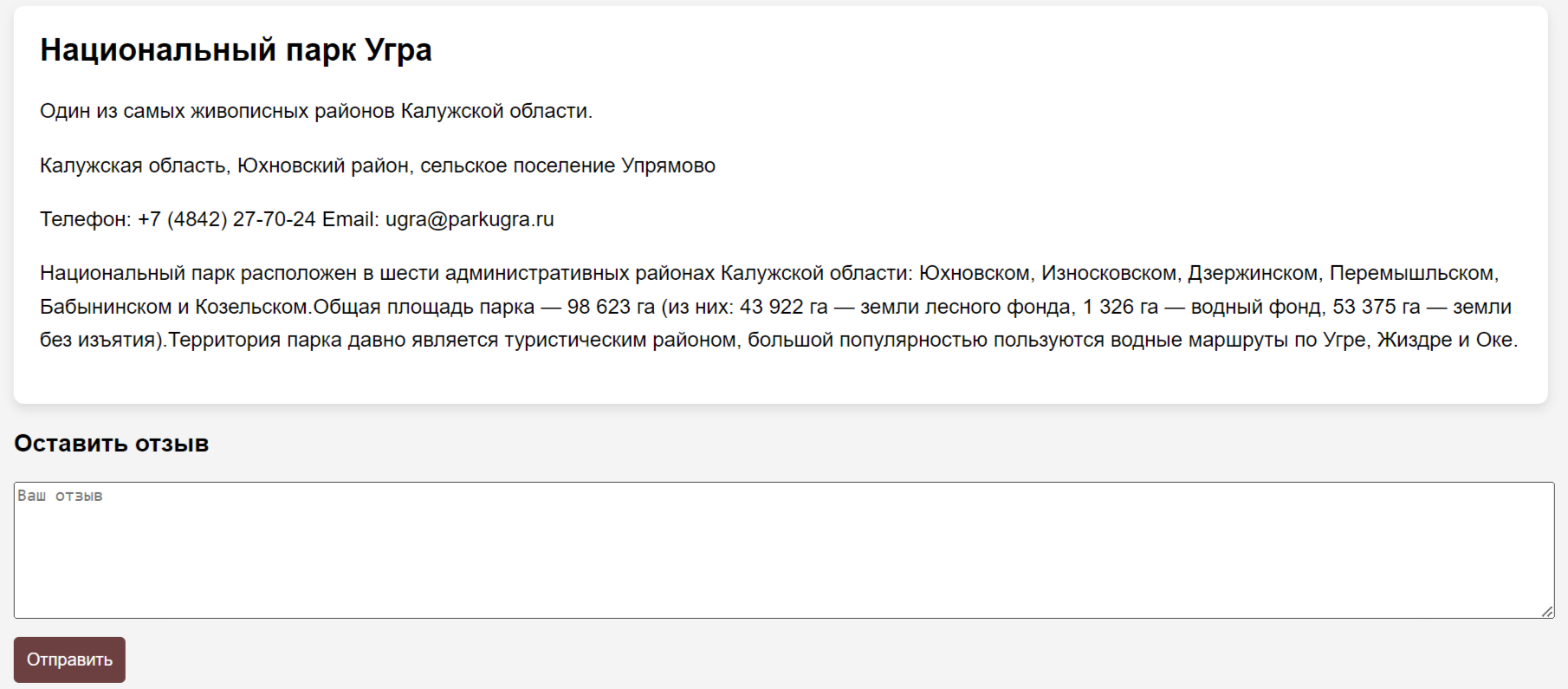


Рисунок 13 – добавление отзыва

Введите текст отзыва и нажмите кнопку **"Отправить"**.

Ваш отзыв будет отображаться на странице подробной информации о достопримечательности (Рисунок 14).

Рисунок 14 - Отзывы

Администратор может удалять один или сразу все нецензурные, и не относящиеся к определенной локации, отзывы, нажав кнопку **"Удалить"**. Если вы хотите предложить новую достопримечательность, нажмите на кнопку **"Предложить достопримечательность"** в шапке сайта.

**24**

В открывшейся форме заполните следующие поля: название, описание, адрес, контактная информация, дополнительная информация, фото.

Нажмите кнопку **"Отправить"**, чтобы отправить предложение администратору.

Также вы можете указать событие (например, фестивали, выставки и др.), которое будет происходить в данном месте. Администратор также проверит достоверность этого события и загрузит на страницу сайта с общим описанием места и отзывами (Рисунок 15).



Рисунок 15 – Предстоящие события

Если вы администратор, вы можете просмотреть все предложения новых достопримечательностей, нажав на кнопку **"Посмотреть предложения"** в шапке сайта. На странице предложений вы увидите список всех предложений, отправленных пользователями. Для каждого предложения вы можете нажать кнопку **"Утвердить"**, чтобы добавить достопримечательность в каталог, или кнопку **"Отклонить"**, чтобы отклонить предложение.

**25**

**Заключение**

При выполнении курсового проекта на тему «TravelGuide – ваш гид по Калужской области» была исследована и описана предметная область, проведен анализ аналогов данной системы, а также выбор инструментов и платформы для разработки.

Проведен анализ объектов автоматизации и разработаны методы решения технических задач. Также была разработана и реализована структура базы данных, серверная часть приложения и веб-интерфейс системы. База данных была наполнена тестовыми данными.

Результатом данного курсового проекта является рабочее веб-приложение.

**26**

**Список используемых источников.**

1. Сакулин, С. А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие / С. А. Сакулин. — Москва : МГТУ им. Н.Э. Баумана, 2017. — 112 с. - ISBN 978-5-7038-4724-4.

2. Проектирование объектов баз данных в среде Access : учебное пособие / Брешенков А. В., Губарь А. М.; МГТУ им. Н. Э. Баумана. - М. : Изд-во МГТУ им. Н. Э. Баумана, 2006. - 180 с. ил. - Библиогр. в конце кн. - ISBN 5-7038-2837-

3. Введение в пректирование баз данных : учебное пособие для вузов / Лукин В. Н. - М. : Вузовская книга, 2013. - Библиогр.: с. 140. - ISBN 978-5-9502-0741-9.

4. Концепции систем баз данных: Зильбершатц А., Корт Г.Ф., Сударшан С., 2019

5. Ревунков, Г.И. Проектирование баз данных - М.: МГТУ им. Н.Э. Баумана, 2009. — 20 с. — Режим доступа: <http://e.lanbook.com/book/52390>

6. Гринберг М. Разработка веб-приложений с использованием Flask на языке Python / М. Гринберг. - Москва : ДМК Пресс, 2014. - 272 с. - ISBN 978-5-97060-138- 9\_ int. - URL: https://ibooks.ru/bookshelf/372852/reading

**27**