**Городское бюджетное общеобразовательное учреждение**

**"Инженерная школа №1581", Москва**

**"Сайт для создания персональной карты мира"**

Работу выполнили:

Обучающиеся 10 К класса

ГБОУ Инженерная школа 1581

Палий Дарья Александровна

Камышан София Алексеевна

Докучаева Анжела Михайловна

Научный руководитель:

Гришина Арина Александровна

г. Москва, 2024

**ОГЛАВЛЕНИЕ**

[ВВЕДЕНИЕ 4](#_Toc185420775)

[**Методика выполнения работы** 6](#_Toc185420776)

[**1. Создание дизайна.** 6](#_Toc185420777)

[**2. Создание вёрстки сайта.** 7](#_Toc185420778)

[**3. Реализация основных функций.** 8](#_Toc185420779)

[Список литературы 14](#_Toc185420780)

# **ВВЕДЕНИЕ**

*Актуальность проекта*

В наше время цифровые технологии играют все более важную роль в повседневной жизни. Одним из интересных и перспективных проектов в этой области является создание персональной виртуальной карты мира. Этот проект позволит пользователям создать уникальную интерактивную карту, на которой они смогут отмечать свои путешествия, любимые места, интересные события и многое другое.

В нашем проекте мы хотим реализовать инновационную и удобную в использовании платформу, которая позволит каждому создать свою собственную виртуальную карту мира и делиться ею с друзьями и близкими.

Актуальность проекта подтверждается большой популярностью у туристов (путешественников) компьютерных программ, в том числе web-приложений, позволяющих обрабатывать топографические данные (информацию).

Такая виртуальная карта может быть полезной как для путешественников, так и для тех, кто любит исследовать различные уголки планеты. Кроме того, виртуальная карта мира может быть использована в образовательных целях, помогая ученикам и студентам изучать географию и культуру различных стран.

Таким образом, создание персональной виртуальной карты мира является актуальным и перспективным проектом, который может приносить пользу и удовольствие многим пользователям.

*Обоснование выбора темы*

Выбор темы проекта «Создание персональной виртуальной карты мира» обусловлен необходимостью разработки удобного и инновационного инструмента для путешественников и любителей географии. Современные технологии позволяют создать уникальное пространство, в котором пользователь сможет удобно отслеживать свои путешествия, отмечать интересные места и делиться ими с друзьями.

Кроме того, создание такой карты может способствовать развитию интереса к географии и культуре различных стран, а также стимулировать путешественников и исследователей к новым открытиям и приключениям. Таким образом, данная тема проекта не только актуальна, но и может привлечь широкую аудиторию и стать популярным инструментом для любителей путешествий и географии.

Цель:

Создать платформу для реализации персональной виртуальной карты мира, которая будет выполнять следующие функции:

1. Создание персонального профиля пользователя с возможностью добавления информации о себе.
2. Добавление меток на карту с указанием мест, которые пользователь хочет посетить или уже посетил.
3. Обмен картой и опытом с другими пользователями с использованием QR-кодов.
4. Возможность сохранения карты в личном кабинете для последующего доступа и редактирования.

Задачи проекта:

1. Создание пользовательского интерфейса и дизайна.
2. Создание верстки сайта.
3. Реализация основных функций.
4. Создание базы данных.
5. Запуск и продвижение среди пользователей.

# **1. Методика выполнения работы**

Методика выполнения индивидуального проекта включает несколько этапов: 

**1.1 Создание дизайна.**

Пользовательский интерфейс должен обеспечивать выполнение следующих четырёх основных функций:

управление компьютером путём действий пользователя: инициация, прерывание, отмена компьютерных процессов и т. п.;

﻿﻿﻿ввод данных, осуществляемый пользователем, и отклик системы.

отображение данных, включающее отображение данных, вводимых пользователем, который может управлять процессом отображения данных;

﻿﻿﻿поддержка оператора в процессе деятельности, осуществляемая по каналам обратной связи, в которых циркулирует информация об ошибочных или случайных (не по алгоритму) действиях пользователя.

Дизайн пользовательского интерфейса (UI) — это создание визуально привлекательных и удобных интерфейсов, которые улучшают общий пользовательский опыт.

Дизайн фокусируется на внешнем виде, восприятии и интерактивности продукта. Это включает в себя выбор подходящих цветовых схем, типографики и макета для создания целостного визуального восприятия.

Основные принципы дизайна пользовательского интерфейса:

- Последовательность. Благодаря ей пользователи могут предсказать поведение элементов, что повышает их уверенность и доверие к интерфейсу.

* *Визуальная иерархия*. Это расположение и организация элементов интерфейса таким образом, чтобы внимание пользователей в первую очередь обращалось на самую важную информацию.
* *Обратная связь.* Она помогает пользователям понять, каковы результаты их действий и были ли они успешными.

UI-дизайнеры работают с внешним видом интерфейса, такими элементами, как кнопки, иконки, шрифты и цвета, чтобы сделать интерфейс удобным для пользователя.

Для получения эффективного результата разработки пользовательского интерфейса используют различные подходы к проектированию:

* + - 1. Подход, ориентированный на пользователя (User Centered) - основным содержанием этого подхода является ориентация на пользователя, т. е. в первую очередь необходимо узнать, что хочет пользователь получить от проектируемого интерфейса. Далее в процессе проектирования полученные требования реализуются в продукте. При сборе информации используются методы наблюдения за работой пользователя, проводятся интервью.
      2. Системный подход (System). Пользователь рассматривается как маленькая интеллектуальная часть системы «человек - программный продукт».
      3. Деятельностный подход (Activity Centered). Изучается деятельность пользователя в целом, и постепенно оптимизируются её отдельные моменты.
      4. Итеративный подход (Agile) — метод последовательных приближений. Суть итеративного подхода заключается в создании изначально самого простейшего прототипа с целью показать заказчику и затем постепенно дорабатывать прототип, основываясь на реакции заказчика после каждого шага доработки.
      5. Экспертный подход (Genius). Заключается в следующем: эксперт собирает важную, по его мнению, информацию, ведёт переговоры с заказчиком, задаёт нужные вопросы. На основе полученной информации создаётся интерфейс.
      6. Целеориентированный подход проектирования (Goal Centered Design). Разработка интерфейса ориентируется на цель, которая будет достигаться данным программным продуктом.
      7. Средоориентированный подход. Разрабатывается среда интерфейса как место деятельности оператора.

При проектировании интерфейса мы обращали особое внимание на потребности и желания пользователей. Также провели анализ платформ, таких как «Золотое яблоко», «GoogleMaps» и других, с большой аудиторией. По результатам анализа нами выбраны основные направления создания интуитивно понятного пользователю интерфейса.

Сайт выполнен в классическом стиле в сине – зеленой цветовой гамме на фоне карты мира. Вверху страницы располагается логотип и название сайта. Справа находится sidebar с необходимыми страничками и функциями. Дизайн главной страницы представлен на рисунке 2. Остальные страницы выполнены в этом же стиле.



Рисунок 2 – Дизайн главной страницы

* 1. **Создание вёрстки сайта.**

В проекте используется свободный фреймворк для разработки быстрых и безопасных веб-приложений и сайтов на языке Python – Django.

В качестве среды разработки в проекте используется Visual Studio Code.

Верстка сайта в *Django* — это процесс создания интерфейса сайта с использованием *HTML, CSS* и *JavaScript*.

*Django* предоставляет возможность легко интегрировать верстку веб-страниц с логикой приложения.

Для создания верстки сайта в *Django* мы создали шаблоны *HTML,* в которых содержатся весь необходимый контент страницы. Эти шаблоны мы используем для создания всех страниц.

Пример использования кода в файле *urls.py* *Django* для определения маршрута главной страницы представлен на листинге 1.

Листинг 1 – Пример кода из кода *urls.py*

from django.urls import path

from .views import index

urlpatterns = [

path (‘’, index),

]

В этом примере импортируется функция *path*, которая связывает *URL* с функциями представления, и модуль *views* текущего пакета. Затем в списке *urlpatterns* вызывается функция *path*, первым параметром указывается пустая строка, а вторым — функция *index*.  Пустая строка соответствует маршруту главной страницы, и при обращении к ней будет вызываться функция представления *index*. Также через третий параметр функции *path* можно указать имя маршрута, например: *path('', views.index, name='home')*

Шаблон — это образец для формирования документа, который будет представлен клиенту: веб-страницы, файл в формате *XML, PDF* и пр. Подсистема *Django,* называемая шаблонизатором, загружает шаблон, объединяет его с данными, извлеченными из моделей, полученными от посетителя или сгенерированными в процессе работы, и формирует на основе всего этого полноценный документ, который и отправляется клиенту.

Применительно к веб-страницам, шаблон — это файл с *HTML* —кодом страницы, содержащий особые команды шаблонизатора: директивы, теги и фильтры. Директивы указывают поместить в заданное место *HTML*-кода какое-либо значение, теги управляют генерированием содержимого результирующего документа, а фильтры выполняют какие-либо преобразования указанного значения перед выводом.

При создании шаблонизатора мы использовали следующий код:

Листинг 2 — Пример кода шаблонизатора.

<!doctype html>

<html lang="ru">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, intial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="IE-edge">

<title>Мечты на карте </title>

</head>

<body>

<h1>код </h1>

</body>

## **Реализация основных функций.**

В нашем проекте реализованы следующие функции.

1. Функция авторизации на сайте. Пользователь может авторизоваться в программе, используя свои учетные данные, такие как логин и пароль. При входе на сайт пользователю предлагается форма для ввода учетных данных, которые затем проверяются на соответствие информации, хранящейся в базе данных сайта. Если введенные данные верны, пользователь получает доступ к своей учетной записи и может пользоваться всеми доступными функциями и сервисами сайта. В случае неверного ввода учетных данных, пользователю выводится сообщение об ошибке и предлагается повторить попытку.
2. Функция выбора стран. Авторизованный пользователь может выбрать страну, которую уже посетил и поставить флажок. Далее эта страна будет отображаться на странице «Мои визиты». Также, авторизованный пользователь может выбрать страну, которую хочет посетить и отметить флажком другого цвета. Избранные страны будут отмечены на странице «Избранное». Эта функция поможет пользователям создать интерактивную карту своих путешествий.

Пользователь может в поисковике написать страну далее отметить ее как посещенную, либо ту которую хочет посетить.

1. Функция написания отзыва о стране на сайте предполагает возможность для пользователей оставить свой отзыв о своем опыте посещения страны. При написании отзыва о стране, пользователи могут поделиться информацией о своем впечатлении от поездки, привлекательных местах для посещения, интересных культурных особенностях и традициях, а также другие полезные советы. Отзывы будут находится на странице «Мои впечатления».
2. Ссылка. Возможность пользователю поделиться своей персональной картой с друзьями, тем самым увеличить спрос продукта. Карта может стать одним из способом сохранить воспоминания о посещенных местах и делиться ими с другими.
   1. **Создание базы данных.**

Для хранения сведений о зарегистрированных на сайте пользователях, а также регистрации их действий (установка флажков, комментарии и.д.) используется база данных.

В качестве системы управления базами данных (СУБД) в проекте используется *PostgreSQL.*

Диаграмма базы данных сайта представлена на рисунке 2.

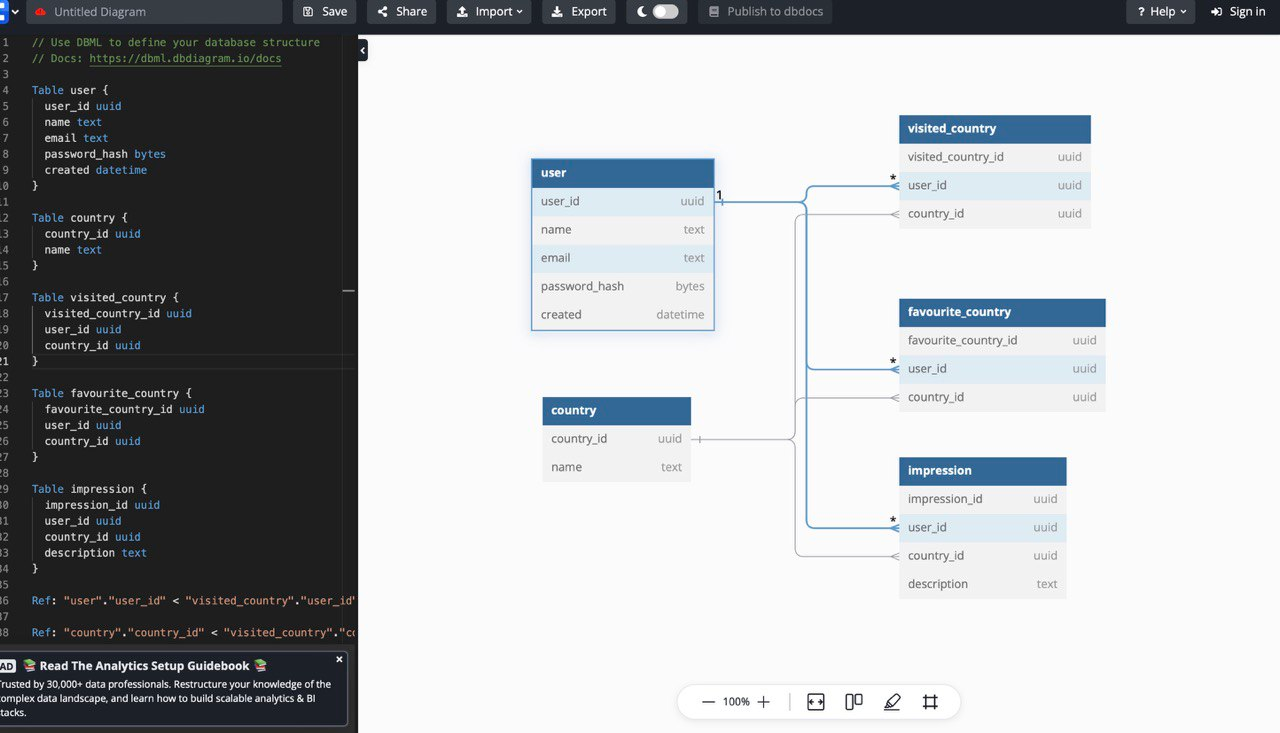


Рисунок 2 - Диаграмма базы данных сайта

База данных проекта состоит из 5 таблиц.

Таблица *User* – содержит сведения о зарегистрированных на сайте пользователях.

Таблица *Country* – справочник стран.

Таблица *Visited Country* – содержит страны, которые были посещены раннее.

Таблица *Favourite Country* – содержит список стран, которые пользователь планирует посетить.

Таблица *Impressions* – отзывы, впечатления о посещенной стране.

# **Тестирование.**

Тестирование сайта — это процесс проверки веб-приложений на наличие ошибок, багов и уязвимостей. Оно помогает убедиться, что сайт работает корректно, соответствует требованиям и обеспечивает хороший пользовательский опыт.

Методы тестирования:

Тестирование черного ящика​ представляет собой стратегию, в которой тестирование основано исключительно на требованиях и спецификациях. В отличие от своего дополнения, тестирования белого ящика, тестирование черного ящика не требует знания внутренних путей, структуры или реализации проверяемой программы.

Тестирование белого ящика​ представляет собой стратегию, в которой тестирование основано на внутренних путях, структуре и реализации проверяемой программы. В отличие от дополняющего тестирования черного ящика, тестирование белого ящика обычно требует детальных знаний в области программирования.

Метод тестирования за столом — это проверка исходного кода программы, выполняемая одним человеком.  Он читает текст программы, проверяет его по списку и пропускает через программу тестовые данные. Исходя из принципов тестирования, проверку за столом должен проводить человек, не являющийся автором программы.

Таблица 1. Функциональное тестирование.

|  |  |  |
| --- | --- | --- |
| № пользователя | Удобство использования | Удобство эксплуатации |
| 1 | 9 | 8 |
| 2 | 7 | 7 |
| 3 | 8 | 8 |
| 4 | 6 | 9 |
| Средняя оценка: | 7,5 | 8 |

Таблица 2. Оценочное тестирование.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Номер теста | Назначение теста | Значение исходных данных | Ожидаемый результат | Реакция программы | Вывод |
| 1 | Проверка корректности работы кнопки «Выбрать страну» | Нажатие на кнопку «Выбрать страну» | Ожидается открытие поисковика | Открытие поисковика | Программа работает верно |
| 2 | Проверка корректности работы кнопки «Мои визиты» | Нажатие на кнопку «Мои визиты» | Ожидается открытие страницы «Мои визиты» | Открытие страницы «Мои визиты» с посещенными странами | Программа работает верно |
| 3 | Проверка корректности работы кнопки «Вход в аккаунт» | Нажатие на кнопку «Вход в аккаунт» | Ожидается открытие страницы входа в аккаунт | Открытие страницы входа в аккаунт | Программа работает верно |

Вывод: Мы создали платформу для планирования путешествий с помощью виртуальной карты, которая будет выполнять следующие функции:

1. Создание персонального профиля пользователя с возможностью добавления информации о себе.
2. Добавление меток на карту с указанием мест, которые пользователь хочет посетить или уже посетил.
3. Обмен картой и опытом с другими пользователями с использованием QR-кодов.
4. Возможность сохранения карты в личном кабинете для последующего доступа и редактирования.

# Список литературы

1)Баранов, С. В. "Основы работы с Django." М.: Издательство, 2021.

2) Дронов, В. А. «Django 2.1 Практика создания веб-сайтов на Python.» Спб: «БХВ-Петербург», 2019.

3) Ли Копланд, *“A Practitioner's Guide to Software Test Design”*, Автор перевода: Уфимцева Галина, 2003.

2) Сергеев, С. Ф., Падерно, Н. А., Назаренко, Н. А. «Введение в проектирование интеллектуальных интерфейсов.» Спб: СПбГУ ИТМО, 2011.

2) Силин, П. А. "Проектирование и разработка веб-приложений." М.: Издательство, 2020.

3) Степанов, И. А. "Методы тестирования программного обеспечения." М.: Издательство, 2017.

4) Шабанов, Д. Ю. "Дизайн пользовательского интерфейса." М.: Издательство, 2020.