



**Министерство науки и высшего образования Российской Федерации
Мытищинский филиал
Федерального государственного бюджетного образовательного учреждения
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МФ МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ Информатика, искусственный интеллект и системы управления
КАФЕДРА ИУ7. Программное обеспечение ЭВМ и информационные технологии

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

НА ТЕМУ:

ПО для поиска поблизости интересных мест

Подготовил Жижин Н.И.

2022 г.

Содержание

[Введение](#)

[Аналитическая часть](#)

[Существующие аналоги](#)

[Категории мест](#)

[Задачи](#)

[Требования к разрабатываемому ПО:](#)

[Конструкторская часть](#)

[Места, категории мест](#)

[API](#)

[Возможности Яндекс API](#)

[Взаимодействие с Яндекс API](#)

[Портал открытых данных Правительства Москвы \[7\]](#)

[Базы данных](#)

[Взаимодействие с пользователем](#)

[Географические координаты](#)

[Алгоритм рекомендации мест пользователю](#)

[Технологическая часть](#)

[Методы разработки баз данных](#)

[Применение различных API](#)

[Поиск информации о точках интереса](#)

[Заключение](#)

[Было реализовано:](#)

[Среди дальнейших планов можно выделить:](#)

[Список источников](#)

Введение

В мире существует огромное количество мест для культурного отдыха и мероприятий, однако большая часть из них остается для людей неизвестными и они предпочитают ходить в те места, где уже были. При этом, даже по одному и тому же определенному маршруту каждый раз.

Для рекомендации интересных и новых мест пользователю для посещения могут использоваться программные средства.

Аналитическая часть

Существующие аналоги

Среди программных средств для рекомендации интересных мест для посещения есть «Яндекс.Афиша»[1], «KUDAGO»[2], а также «Qreeq»[3].

Сервис «KUDAGO» (рис. 1) предоставляет пользователю выбрать необходимую информацию о текущих событиях в Москве и других городах. На первых местах на главной странице находятся крупнейшие события, праздники текущего и следующих дней. А также, среди дополнительных функций сервиса пользователь может прочитать новости в стране, по решать интересные головоломки и задачи, увидеть скидки на товары в супермаркетах, посмотреть на новые экспозиции на выставках, запись на экскурсии по городу, узнать о новых интересных занятиях с детьми. Но этот сервис не учитывает расстояние до мест, предлагаемых пользователю.

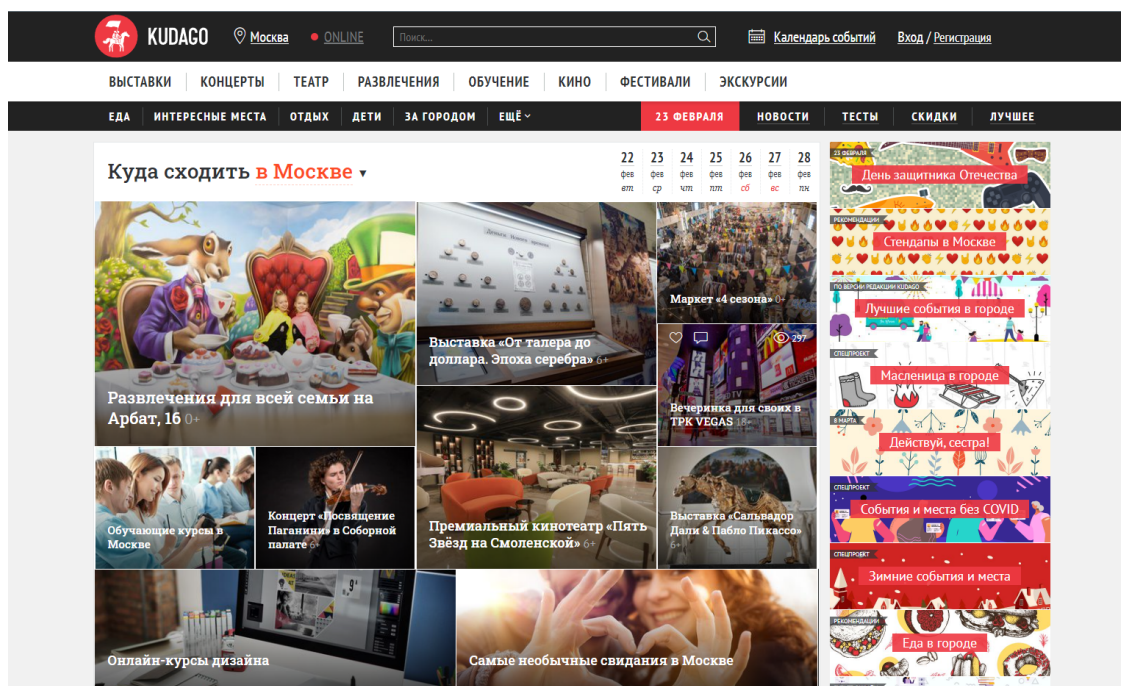


Рис. 1. Главная страница сервиса «KUDAGO».

Сервис «Яндекс.Афиша» (рис. 2) на главной странице предлагает пользователю подборки предстоящих концертов, фильмов и театральных постановок. Здесь можно узнать информацию об интересных видах спорта в данный сезон года, акциях и билетах на различные события. Но в то же время можно выделить невозможность увидеть список интересующих мест, таких как парки, музеи. Нельзя построить маршрут до какого-то места.

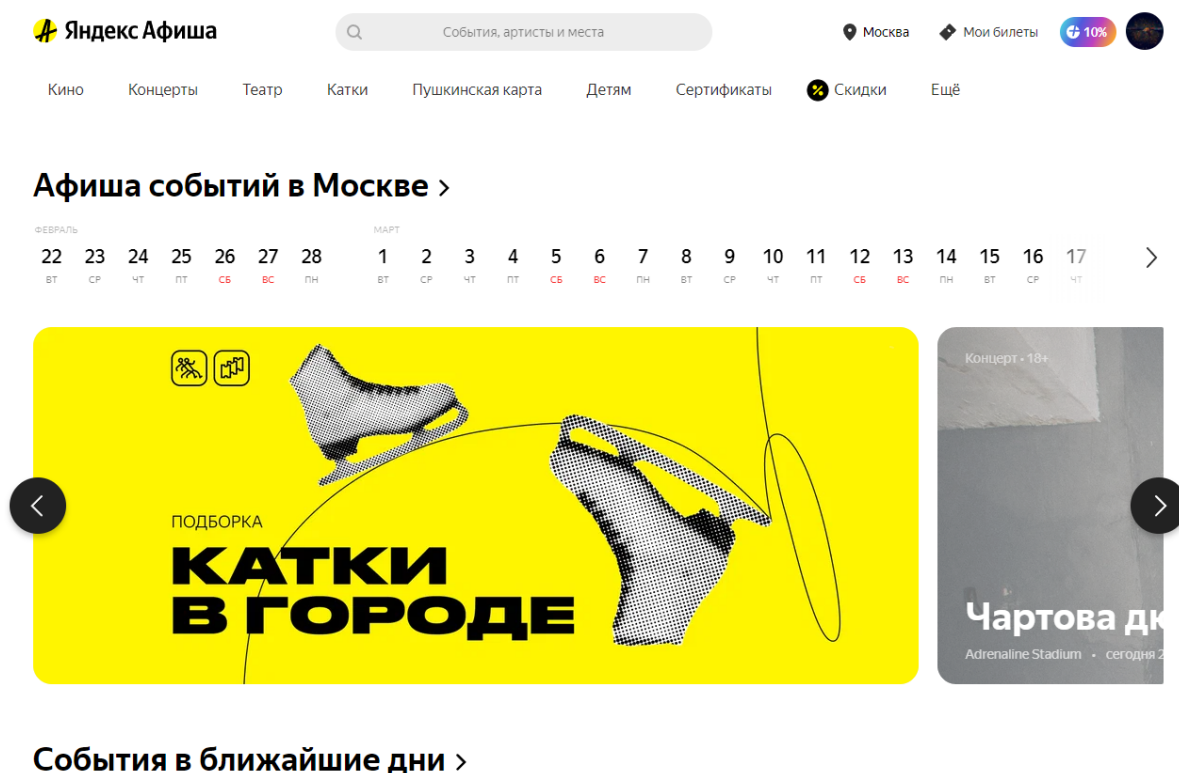


Рис. 2. Главная страница сервиса «Яндекс.Афиша»

Программное средство «Qreeq» (рис. 3) - мобильное приложение для поиска и просмотра ближайших понравившихся ресторанов, кафе и других мест пользователям сети. Но также, можно выделить существенный недостаток, заключающийся в том, что невозможно применить разнообразные фильтры рекомендуемых мест и построить маршрут до них.

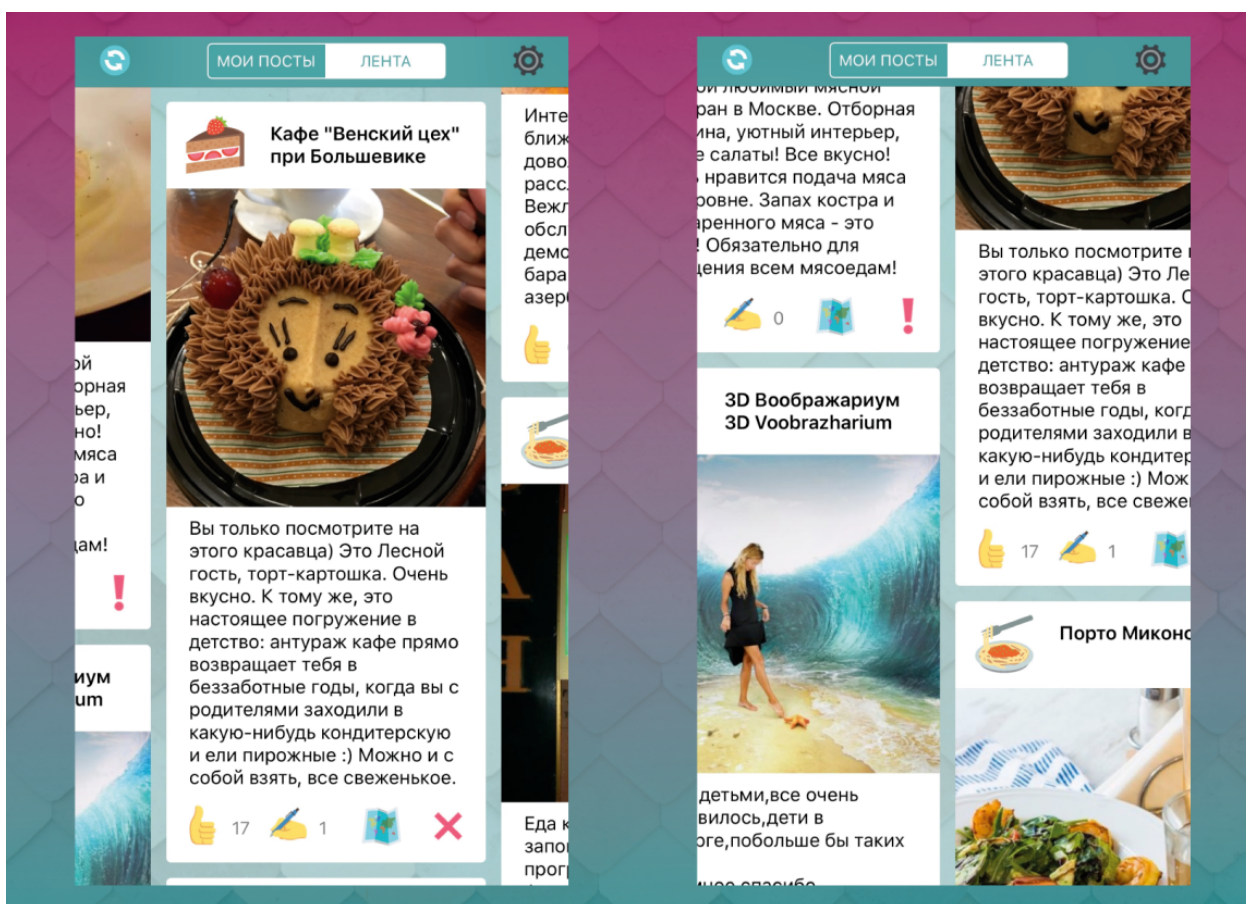


Рис. 3. Интерфейс мобильного приложения «Qreeq».

Категории мест

Различные места могут быть разделены на множество категорий, одними из них являются музеи, парки аттракционов, дикие местности, аквариумы, артворки, кемпы, национальные парки, пикниковые зоны, площадки, защищенные зоны, тематические парки, точки обзора, аквапарки.

У реализованных программных средств присутствует существенный недостаток в том, что они не учитывают расстояние от пользователя до места проведения мероприятия или других мест культурного отдыха.

Из-за данного недостатка возникает необходимость в разработке нового программного обеспечения.

Задачи

- Исследовать работу с картами
- Исследовать работу с координатами, вычисление расстояния между двумя координатами
- Определить способ хранения данных, наиболее подходящий для этого программного средства

Требования к разрабатываемому ПО:

- Возможность выбора необходимых или понравившихся категорий мест для последующей их обработки и генерации.
- Отображение интерактивной карты.
- Выбор максимальной дистанции до нового, выбранного случайным образом места.
- Автоматическое определение точного местоположения пользователя.
- Отображение адреса и расстояния до этого места.

Конструкторская часть

Места, категории мест

Самые популярные места сейчас – это парки, кафе, антикафе, музеи, аквапарки, различные виды аттракционов, и так далее.

API

API (Application Programming Interface) — это совокупность инструментов и функций в виде интерфейса для создания новых приложений, благодаря которому одна программа будет взаимодействовать с другой. Это позволяет разработчикам расширять функциональность своего продукта и связывать его с другими. Большинство крупных компаний разрабатывают API для сторонних разработчиков.

Возможности Яндекс API

Яндекс API – программный интерфейс для работы с сервисом «Яндекс». Он предоставляет множество технологий, которые доступны сторонним разработчикам. Одними и самыми популярными из них являются:

- API Карт. Позволяет программисту использовать картографические данные в проектах.

- API Директа. Предоставляет инструменты для управления рекламными кампаниями.
- API Диска. Предназначено для обращения к сервису Яндекс.Диск из программного обеспечения.
- ClickHouse. Предоставляет аналитическую СУБД.
- Yandex.Cloud. Облачная платформа, где можно создавать свои проекты.
- AppMetrica. Предоставляет сервис аналитики для мобильных приложений.

Взаимодействие с Яндекс API

Для начала работы с сервисом, необходимо получить токен Яндекс API. Токен - уникальный идентификатор, выдаваемый пользователю и позволяющий системе производить его авторизацию для возможности создавать запросы API. Для этого необходимо зарегистрироваться в Яндекс.

Работа с любым API заключается в передачи задачи, а также входных данных, которые необходимо обработать серверу, который предоставляет эту возможность.

Портал открытых данных Правительства Москвы [7]

Это общедоступный веб-ресурс, предоставляющий в свободном доступе информацию о деятельности органов государственной власти города Москвы, а также информацию о местоположении многих мест, сгруппированных по категориям в сети Интернет. Информация представлена

в виде таблиц, машиночитаемом видах (в формате .csv), а также в виде карты (Электронный Атлас города Москвы).

Базы данных

В настоящее время существуют различные типы баз данных:

- Реляционные - используют табличное представление данных,
- Иерархические - используют структуру данных в виде деревьев,
- Сетевые – используют структуру в виде сети с множественными связями между данными, т. е. в виде графа.

Для хранения в базе данных записей о местах (идентификатор, координаты, тип места) является наилучшим выбором реляционный тип базы данных.

Взаимодействие с пользователем

Для начала работы с программой пользователь запускает ПО, указывает свое местоположение, если его необходимо корректировать. Затем выбирает интересующие его категории мест, которые будут использованы для генерации. Выбирает предельную дистанцию до интересного места. Запускает генерацию, и на карте отображается маршрут до сгенерированного места. Если пользователь пропустил выбор и ввод таких важных данных, как категории мест, расстояние до места или текущее местоположение, то программа выведет ошибку о том, что одно или несколько полей пропущены и не заполнены.

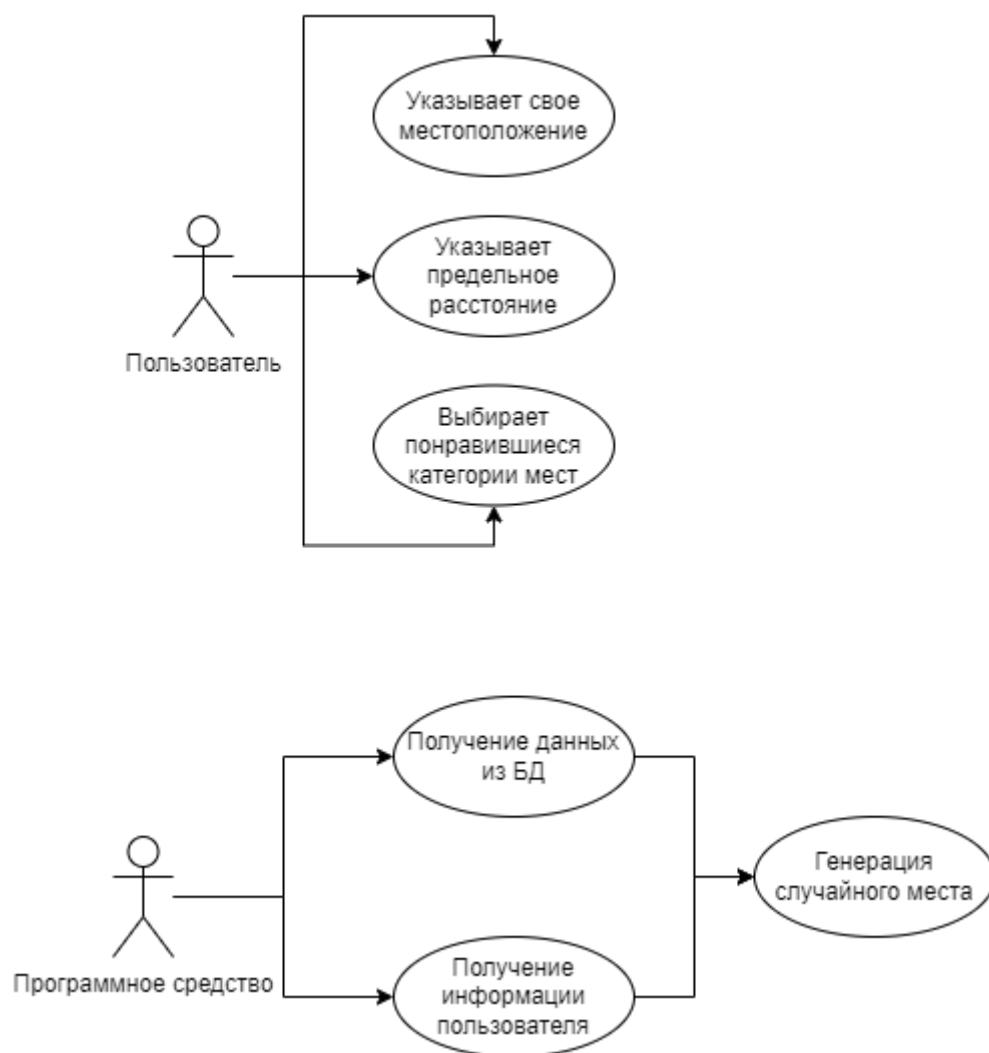


Рис. 4. Диаграмма USECASE

Для работы программы существует необходимость, заключающаяся в обработке координат, и вычислении расстояния между точками, которые находятся в этих координатах.

Географические координаты

Географические координаты – два значения: широта и долгота, которые определяют положение точки на земной поверхности. Земля принимается за шар.

$$\Delta\sigma = \arctan \frac{\sqrt{(\cos \phi_2 \sin(\Delta\lambda))^2 + (\cos \phi_1 \sin \phi_2 - \sin \phi_1 \cos \phi_2 \cos(\Delta\lambda))^2}}{\sin \phi_1 \sin \phi_2 + \cos \phi_1 \cos \phi_2 \cos(\Delta\lambda)}.$$

Рис. 4. Формула для расчета ортодромного расстояния на сфере.

Для вычисления расстояния на земной поверхности существует несколько способов, например, ортодромное расстояние на сфере (расстояние по большому кругу) (рис. 4), которое реализовано при помощи тригонометрических формул.

Алгоритм рекомендации мест пользователю

Процесс генерации рекомендуемого места начинается с определения случайной категории из списка выбранных пользователем. Затем из базы данных выбираются только те места, категория которых совпадает со сгенерированной. Начинается выбор случайной координаты из этого списка. Если предельное расстояние не превышено, пользователю выводятся координаты места и карта. Если предельное расстояние превышено, то это место исключается и будут генерироваться новые места, пока оно не будет превышено. По окончании генерации производится API-запрос к геокодеру сервиса Яндекс для определения названия рекомендуемого места, после чего производится загрузка страницы Яндекс.Карт для отображения интерактивной карты в приложении.

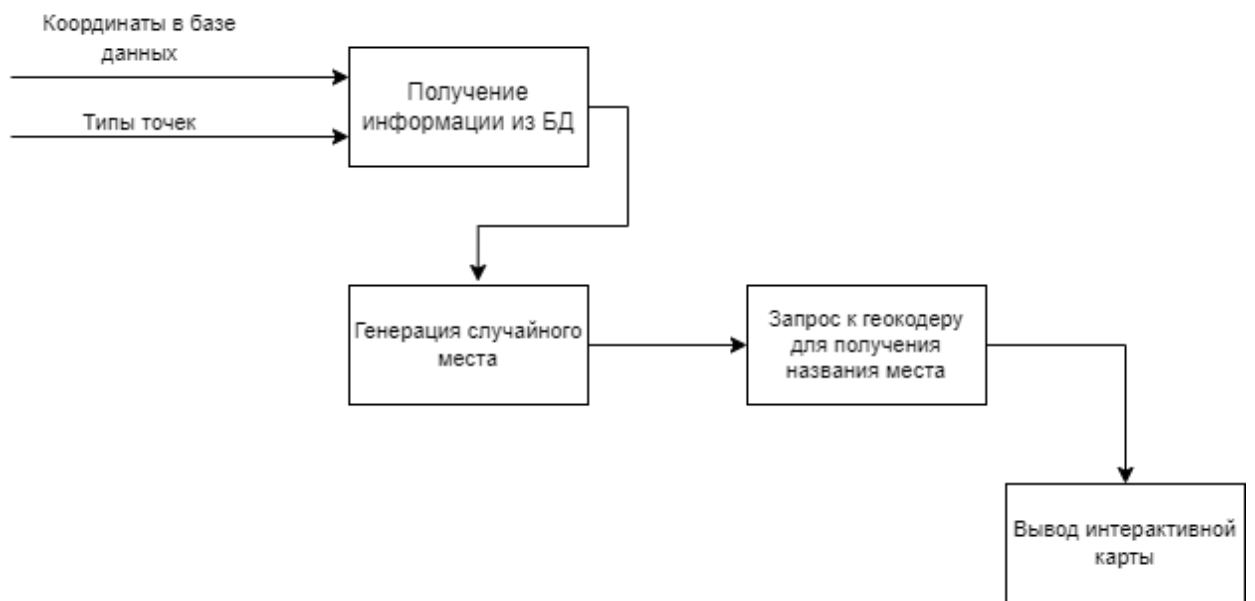


Рис. 5. Диаграмма IDEF0

Технологическая часть

Средством реализации данного программного обеспечения был выбран язык C# в связи с дальнейшим планируемым переходом проекта в дополнительные форматы мобильного и веб-приложений, из соображений сохранения единообразия стека технологий.

Методы разработки баз данных

Существует два основных подхода разработки баз данных. Code-First и Database-First.

Подход, называемый Code-First (сначала код), предполагает минимальное участие в проектировании сущностей базы данных программистом. Он просто пишет код, остальное делает Microsoft Entity Framework. Он подходит в случаях если главное в проекте – бизнес логика, а база данных – это способ хранения данных.

Подход Database-First (сначала данные), позволяет писать приложения для уже существующих баз данных. Базы данных в реальных приложениях довольно быстро становятся сложными и пытаться создать модель для существующей базы данных, которую могут понять разработчики, довольно трудно. Еще тяжелее написать код использования модели, в котором происходит взаимодействие с базой данных.

Для взаимодействия с базой данных был использован Microsoft Entity Framework Core для обращения к данным из программного кода и для отказа от языка запросов SQL во избежание появления уязвимостей, таких как SQL - инъекция. А также в связи с тем, что изначально данных не было.

В качестве основной базы данных была выбрана PostgreSQL, так как она является реляционной и свободной к распространению.

Применение различных API

Яндекс API был использован с целью предоставления карт, а также геокодинга (преобразования координат в полное название местности).

Текущее местоположение определяется при помощи сервиса Geolocation API, которое позволяет определить его с точностью до дома, что является важным критерием при разработке этого ПО.

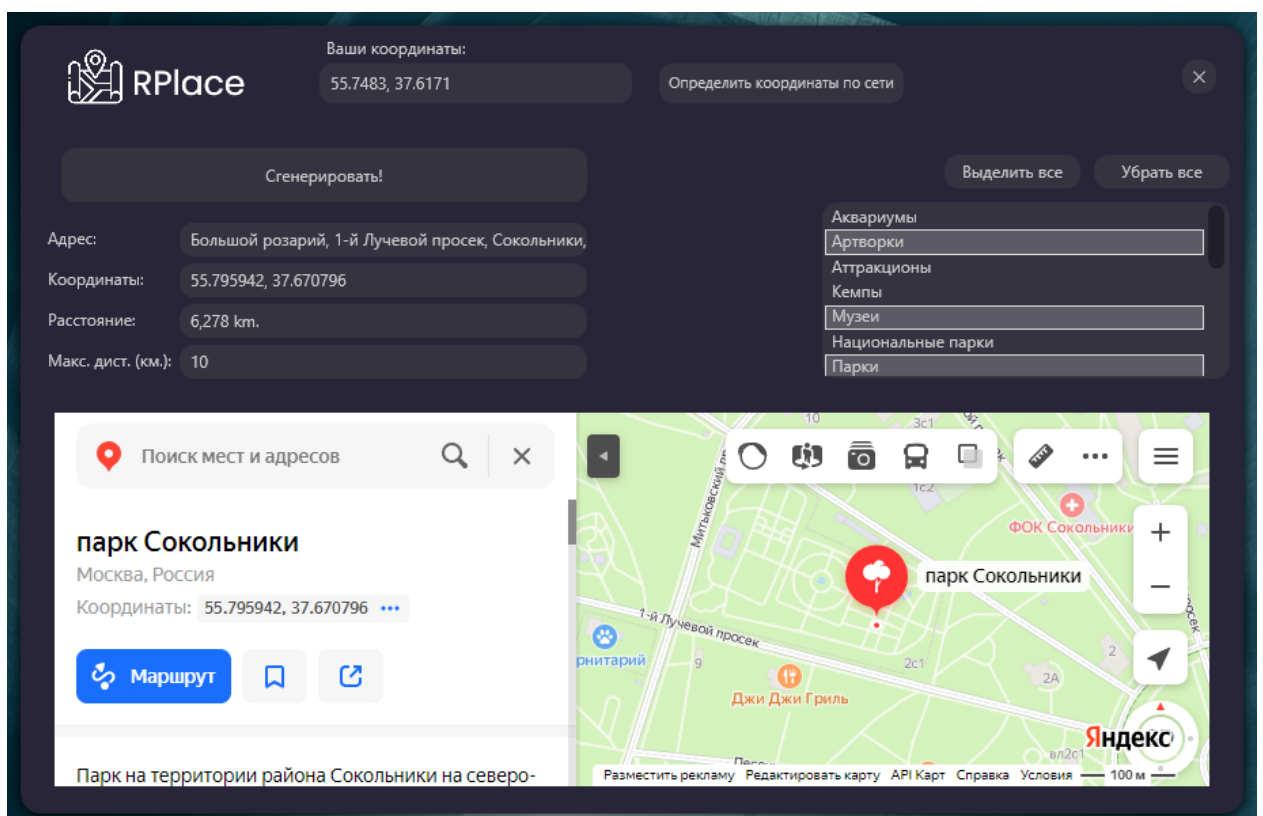


Рис. 6. Текущий интерфейс ПО.

Поиск информации о точках интереса

В начале, был найден портал открытых данных Москвы и Московской области, где можно получить доступ к сборкам файлов с точками интереса (POI) и другой полезной информации об объектах. Для получения координат точек был разработан скрипт на языке Python, который производил парсинг данных с портала открытых данных правительства Москвы. Но в связи с тем, что количество полученных данных было незначительно, пришлось отказаться от использования этого ресурса в пользу сервиса OSM (OpenStreetMap).

Для обработки данных с этого сервиса, необходимо загрузить полную карту Москвы и московской области со всеми объектами, а затем сделать фильтрацию только тех данных, которые были необходимы для работы программы. При помощи языка C# была написана программа, позволяющая фильтровать и загружать данные из скачанной карты в базу данных PostgreSQL.

Изначально, было принято решение использовать несколько групп файлов для хранения координат, разбитых по типам. Но в связи с тем, что скорость работы с таким количеством файлов очень низкая. С переносом всех данных в базу данных, значительно увеличилась скорость чтения всех координат, что повлекло за собой уменьшение времени, отведенного на генерацию нового случайного места.

Заключение

Было реализовано:

- Возможность выбора понравившихся категорий мест
- Отображение интерактивной карты.
- Выбор максимальной дистанции до генерируемого места
- Автоматическое определение местоположения пользователя
- Автоматическое определение названия места по его координатам
- Вычисление расстояния между двумя точками на поверхности Земли

Реализованное программное средство является прототипом, в связи с чем можно составить план дальнейших нововведений в ПО.

Среди дальнейших планов можно выделить:

- Разработка веб сервиса для удобного использования этого программного средства с любого устройства без необходимости скачивания.
- Определение точного местоположения по GPS для мобильных устройств.

Список источников

1. Яндекс.Афиша. — Текст : электронный // : [сайт]. — URL: <https://afisha.yandex.ru/moscow> (дата обращения: 22.02.2022).
2. Qreeq. — Текст : электронный // : [сайт]. — URL: <https://vc.ru/tribuna/22396-qreeq> (дата обращения: 22.02.2022).
3. KUDAGO. — Текст : электронный // : [сайт]. — URL: <https://kudago.com/msk/> (дата обращения: 22.02.2022).
4. OpenStreetMap. — Текст : электронный // : [сайт]. — URL: <https://www.openstreetmap.org/> (дата обращения: 22.02.2022).
Точки интереса. — Текст : электронный // OpenStreetMap : [сайт]. — URL: https://wiki.openstreetmap.org/wiki/Points_of_interest (дата обращения: 22.02.2022).
5. Сервис определения текущего местоположения. — Текст : электронный // : [сайт]. — URL: <https://www.w3.org/TR/geolocation/> (дата обращения: 22.02.2022).
6. Определение дистанции по прямой на Земле. — Текст : электронный // : [сайт]. — URL: https://en.wikipedia.org/wiki/Great-circle_distance (дата обращения: 22.02.2022).
7. Портал открытых данных Правительства Москвы. — Текст : электронный // : [сайт]. — URL: <https://data.mos.ru/> (дата обращения: 22.02.2022).
8. Yandex.Maps Static API. — Текст : электронный // : [сайт]. — URL: <https://yandex.ru/dev/maps/staticapi/> (дата обращения: 22.02.2022).
9. Как я делал сервис для построения интересных пешеходных маршрутов. — Текст : электронный // : [сайт]. — URL: <https://habr.com/ru/post/414433/> (дата обращения: 22.02.2022).