

## РАЗРАБОТКА СИСТЕМЫ ПОСТРОЕНИЯ СОБЫТИЙНЫХ ПЕШЕХОДНЫХ МАРШРУТОВ

**Бестужев К.О.** (федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»)

**Научный руководитель – к.т.н. Мухина К.Д.**

(федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»)

В данной работе проводится анализ развития систем построения маршрутов, на основе публикаций за последние 5 лет. Практически рассматривается проблема сбора данных о местах и событиях из общедоступной базы данных API KudaGo, с целью последующего построения туристического маршрута учитывающего временные ограничения.

**Введение.** Общемировое развитие различных видов туризма приводит как к появлению новых типов сервисов для туристов, так и развитию уже существующих. В настоящее время существует немало популярных средств автоматического построения маршрута из пункта А в пункт Б. Однако, специализированные сервисы, позволяющие построить путь между большим количеством точек, с определенными ограничениями на результат, распространены в меньшей степени.

Данной проблеме посвящено немало публикаций научных коллективов в разных странах, которые предлагают варианты решения задачи нахождения оптимального маршрута, в зависимости от многочисленных факторов: насколько необходимо посещение определенной точки, сколько времени должен занять конечный маршрут, какими денежными средствами располагает пользователь для покрытия расходов по перемещению и др.

Как наиболее комплексный, можно выделить подход, предлагающий к вышеперечисленным ограничениям учитывать так называемые "временные окна" - промежутки времени доступности точки, например - часы работы музея. При этом, потенциальные точки пути рассматриваются не просто как географические координаты, но и как события во времени.

Приняв во внимание публикации на тему конструирования маршрутов различных типов и для разных целей, можно выделить общие для этих работ шаги:

- найти непосредственную практическую проблему;
- собрать информацию о возможных точках маршрута;
- применить наиболее подходящий для решения задачи алгоритм-конструктор.

В качестве практической задачи выбрано построение пеших туристических маршрутов. Авторы близких по теме публикаций, как правило, используют следующие источники данных: соцсети и сервисы хранения фотографий содержащих геотеги, публично доступные спутниковые координаты и собственные базы данных транспортных компаний. Для построения маршрута, в рассмотренных работах использовались эвристические алгоритмы, наиболее часто выбирались оптимизация методами муравьиной колонии или пчелиного роя.

**Основная часть.** Данная работа посвящена системе построения событийных пешеходных маршрутов, поскольку рассматривает временную доступность объекта как один из основных критериев для добавления в туристический маршрут. В качестве источника данных рассмотрена бесплатная база событий и мест API KudaGo, где публикуются концерты, спектакли, выставки, мастер-классы и курсы, события для детей и масштабные городские фестивали.

Была запрошена максимально полная информация о местах и событиях в городах Санкт-Петербург и Москва. Это позволило оставить решение какую именно информацию использовать до момента четкой математической формулировки конечной задачи.

Полученные по запросу результаты предполагалось сохранять постранично, как список словарей. Однако в процессе сбора данных с API KudaGO была выявлена проблема - в

некоторых объектах, из-за неправильно расставленных html-тэгов не удавалось сохранять полученные результаты в требуемом формате .json. В качестве решения было применено разбиение страницы полученной с API KudaGo (содержащей от 20 до 100 записей в зависимости от запроса) на объекты и проверка возможности сохранения каждого из них в требуемом формате. Это позволило, в случае сбоев, отбрасывать не все объекты на странице, а только некорректно отформатированные. Кроме того, допускался вывод ID отброшенных объектов в отдельный список, по которому можно сделать повторный запрос, уже не включающий вызывающие ошибку при обработке поля, после чего добавить результат в общую базу.

Впоследствии, для построения маршрута по заданному алгоритму, можно сделать выборку потенциальных точек, в которую добавлять только места или события отвечающие заданным критериям, например по категории.

Список категорий объекта API KudaGO не всегда удобно использовать в исходном виде. С точки зрения рассмотрения объекта для добавления в маршрут, удобным представляется оставить одну категорию из возможного списка, наиболее востребованную для гостей города. Исходя из этих соображений категории объектов были переопределены по следующему принципу:

- Найти все уникальные категории объектов записанные в базе данных API KudaGO.
- Вручную определить их приоритет с точки зрения востребованности.
- Оставить для каждого из объектов только одну категорию, имеющую максимальный приоритет из всех записанных в его поле "categories".
- Объединить часть категорий содержащих небольшое количество объектов, которые несут схожую смысловую нагрузку, в одну. При этом название общей категории выбирается из объединяемых или записывается новое, исходя из роли входящих в нее объектов для туриста.

**Выводы.** В данной работе проведен анализ источников данных о точках, представляющих потенциальный интерес для туристов, алгоритмов построения пути по ним, а так же возможных практических применениях систем построения маршрутов.

Из базы API KudaGo были собраны сведения о местах и событиях, проведена переработка этих данных, с целью более удобного их использования при построении пешеходных маршрутов. Поскольку в качестве главного различия туристических объектов выступала принадлежность к некоторой категории, было произведено перераспределение и переименование категорий с точки зрения более логичного и осмысленного деления на группы. В качестве наглядного результата, приведены несколько типов визуализации на карте города переработанных данных, включая общие теплокарты распределения потенциальных туристических мест и расположение объектов для каждой категории.