



Институт интеллектуальных кибернетических систем

КАФЕДРА КИБЕРНЕТИКИ

Проект по курсу  
**Технология программирования кибернетических  
систем**

Веб-портал подбора и составления маршрутов для  
путешествий “YourTrip”

---

Руководители: \_\_\_\_\_ ( Козырев В. П., Мальцев И. Ю. )

(подпись)

(ФИО)

Состав команды:

1. Девяткина А.С., Б14-505
2. Козин И.А., Б14-505
3. Рогачев А.И., Б14-505
4. Саенко Д.С., Б14-505
5. Эйдлина М.А., Б14-505

**10.11.2017**

## **Оглавление**

<b>1. Цель</b>	<b>4</b>
<b>2. Глоссарий</b>	<b>4</b>
<b>3. Ключевые особенности</b>	<b>6</b>
<b>4. Проектный план на семестр</b>	<b>7</b>
<b>5. Классы пользователей</b>	<b>8</b>
<b>6. Требования и ограничения системы</b>	<b>8</b>
6.1. Функциональные требования	8
Модуль №1 (Авторизационный сервис)	8
Модуль №2 (Сервис ведения блога)	9
Модуль №3 (Сервис поиска пользователями маршрутов)	9
Модуль №4 (Сервис комментирования и оценивания маршрутов)	10
6.2. Функциональные ограничения	11
6.3. Системные требования	13
<b>7. Определение процессов разработки</b>	<b>13</b>
7.1. Выбор модели жизненного цикла	13
7.2. Выбор методологии разработки	14
<b>8. Проектирование системы</b>	<b>15</b>
8.1 Выбор CASE-средства для проектирования Use-case диаграммы	15
8.2 Use-case диаграмма	15
8.3 Use-case сценарии	18
Модуль №1 (Авторизационный сервис)	18
1.1 Регистрация незарегистрированного пользователя (гостя)	18
1.2 Авторизация зарегистрированного пользователя	19
1.3 Редактирование личной информации авторизованным пользователем	20
Модуль №2 (Сервис ведения блога)	21
2.1 Публикация маршрута	21
2.2 Автоматическое определение координат каждой достопримечательности	22
2.3 Автоматическое определение общей длительности и протяженности маршрута	22
2.4 Просмотр списка личных маршрутов	23
2.5 Добавления в маршрут фотографий	24
Модуль №3 (Сервис поиска маршрутов)	25
3.1 Поиск маршрутов пользователями портала	25
3.2 Сортировка результатов поиска маршрута	26
3.3 Подбор похожих маршрутов	27
3.4 Просмотр определенного маршрута	27
3.5 Возможность поделиться ссылкой на маршрут в vkontakte	28

Модуль №4 (Сервис комментирования и оценивания маршрутов)	29
4.1 Добавление комментариев к маршрутам	29
4.2 Оценивание комментариев других пользователей к маршруту	29
4.3 Удаление своих комментариев к маршруту	30
4.4 Редактирование своих комментариев к маршруту	31
4.5 Удаление отдельных комментариев и маршрутов пользователей	31
4.6 Оформление подписки на публикации новых маршрутов других пользователей	32
4.7 Рассылка уведомлений о публикации новых маршрутов пользователями	32
8.4. Извлечение существительных	33
8.4.1. Первичное извлечение существительных из use-case сценариев	33
8.4.2. Перечень существительных-кандидатов в классы	33
8.5. Диаграмма классов	34
<b>9. База данных системы</b>	<b>35</b>
9.1. Выбор CASE-средства для проектирования модели базы данных	36
9.2. Физическая модель базы данных системы	36
9.3. Оценка размеров базы данных	39
<b>10. Архитектура системы</b>	<b>41</b>
10.1. Общее описание архитектуры системы	41
10.2. Описание акторов, взаимодействующих с системой	42
10.2.1. Google Places API	42
10.2.2. Disqus	43
10.2.3. Amazon RDS	43
10.2.4. Amazon S3	43
10.3. Пропускная способность системы	43
<b>История версий</b>	<b>44</b>

В рамках рассматриваемого проекта разрабатывается веб-портал, предназначенный для формирования пользователями маршрутов посещения достопримечательностей (с возможностью оценивания всего пройденного маршрута, а также возможностью оценивания достопримечательностей в отдельности). Портал будет предоставлять площадку для ведения туристического блога, представляющего собой набор пройденных пользователем маршрутов. Портал ориентирован на пользователей любой возрастной категории, активно путешествующих по миру и желающих поделиться опытом своих путешествий.

## 1. Цель

Разработать систему подбора и составления решений для путешествий в виде подробных маршрутов.

## 2. Глоссарий

Термин	Значение
<b>Веб-портал</b>	сайт в компьютерной сети, который предоставляет пользователю различные веб-сервисы. Разрабатываемый портал включает следующие сервисы: <ul style="list-style-type: none"><li>● авторизационный сервис;</li><li>● сервис ведения личного туристического блога;</li><li>● сервис поиска подходящего маршрута путешествия по городам и меткам;</li><li>● сервис комментирования и оценивания просмотренных маршрутов других пользователей с возможностью подписки на блоги пользователей веб-портала</li></ul>
<b>Маршрут</b>	выбираемая пользователем по контрольным точкам на карте последовательность достопримечательностей, которые были им посещены в рамках путешествия, содержащая также: <ul style="list-style-type: none"><li>● название;</li><li>● координаты посещенных достопримечательностей;</li></ul>

	<ul style="list-style-type: none"> <li>• даты посещения достопримечательностей;</li> <li>• информацию о возможных способах перемещения между пунктами;</li> <li>• примерное время, затрачиваемое на передвижение между пунктами;</li> <li>• оценку (для всего маршрута), представляемую в виде системы из трех оценок (по 5-балльной шкале) – оценка сложности прохождения, оценка культурной насыщенности маршрута и оценка развлекательной составляющей маршрута;</li> <li>• отметку о завершенности/незавершенности (если пользователь заполняет маршрут во время его прохождения);</li> <li>• категорию отдыха (семейный, активный, романтический, познавательный);</li> <li>• отметку о необходимости опубликования (опубликовать/скрыть)</li> <li>• информацию о впечатлениях пользователя (фотографии, рассказ) - опционально;</li> </ul>
<b>Блог</b>	личный раздел сайта, в котором отображаются сформированные данным пользователем маршруты.
<b>Достопримечательность</b>	место на Google Maps, доступное через Google Places API, с указанием координат и оценки пользователем.
<b>Оценка маршрута его создателем</b>	<p>указываемые пользователем баллы в интервале от 1 до 5 по характеристикам:</p> <ul style="list-style-type: none"> <li>• сложность прохождения</li> <li>• культурная насыщенность</li> <li>• развлекательная насыщенность.</li> </ul>
<b>Оценка маршрута</b>	субъективное мнение пользователя, просмотревшего маршрут, об этом маршруте (нравится/не нравится).

<b>другим пользователем</b>	
<b>Оценка достопримечательности</b>	субъективное мнение пользователя, создавшего маршрут, об интересности достопримечательности, оценивается в интервале от 1 до 5.
<b>Оценка комментария</b>	субъективное мнение пользователя об оставленном комментарии к маршруту (нравится/не нравится).
<b>Поиск</b>	<p>нахождение маршрутов:</p> <ul style="list-style-type: none"> <li>• либо по населенному пункту (указывается радиус вокруг этого пункта);</li> <li>• либо по входящим в маршрут достопримечательности;</li> <li>• либо по нескольким населенным пунктам на протяжении маршрута;</li> <li>• по длительности и протяженности маршрута (опционально);</li> <li>• расстоянию, преодолеваемому в маршруте (опционально);</li> <li>• количеству достопримечательностей в маршруте (опционально);</li> <li>• завершенности (опционально);</li> <li>• категории отдыха (семейный, активный, романтический, познавательный) (опционально);</li> </ul> <p>выдаваемые результаты сортируются в зависимости от указанных приоритетов (оценка маршрутов и дата создания)</p>

### 3. Ключевые особенности

1. Функция формирования пользователями маршрутов путешествий (с возможностью оценивания маршрута в целом) в формате блога, а также оценивания отдельных достопримечательностей.

2. Функция поиска маршрутов.

3. Функция добавления подписок на посты других участников.
4. Функция оценивания (в шкале от 1 до 5) маршрутов других пользователей, их комментирования и оценки этих комментариев (нравится/не нравится).
5. Интеграция с различными соц. сетями: instagram, vkontakte.

#### 4. Проектный план на семестр

В рамках текущего семестра планируется реализовать авторизационный сервис и функцию формирования пользователями маршрутов путешествий в формате блога с возможностью автоматического нахождения и определения координат каждой достопримечательности с помощью Google Places API. Также планируется реализация функции добавления комментариев под каждым маршрутом с возможностью оценивания полезности комментария другими пользователями (по системе “нравится/не нравится”). В число реализуемых задач также входит функция удаления модераторами комментариев и маршрутов, нарушающих правила портала.

Рассматриваемую задачу предлагается разбить на подзадачи, указанные в таблице:

Номер задачи	Задачи	Сроки выполнения
1.	Уточнить системные и функциональные требования;	9 неделя
2.	Определение общей архитектуры;	9 неделя
3.	Проектирование базы данных;	10 неделя
4.	Разработка авторизационного сервиса, предоставляющего возможность регистрации и авторизации пользователей портала;	11 неделя
5.	Проектирование макетов интерфейсов пользователя;	12 неделя
6.	Разработка сервиса создания маршрутов пользователей;	13 неделя
7.	Добавление возможности комментирования и оценки маршрутов.	14 неделя

## 5. Классы пользователей

1. Гость
2. Авторизованный пользователь
3. Модератор

## 6. Требования и ограничения системы

### 6.1. Функциональные требования

#### Модуль №1 (Авторизационный сервис)

**Описание:** Авторизационный сервис, обеспечивающий возможность регистрации и авторизации на веб-портале.

**Роли пользователей:** Гость, авторизованный пользователь, модератор.

**Требования:**

1. Регистрация незарегистрированного пользователя (гостя) путем заполнения форм, указанных в функциональных ограничениях.

При успешном заполнении всех форм (обязательных для заполнения) будет отправляться сообщение о регистрации на указанный гостем адрес электронной почты. Для подтверждения регистрации пользователю необходимо перейти по ссылке, указанной в письме.

При регистрации пользователя класса «Модератор» проверка данных и внесение изменений в БД осуществляется при помощи оператора базы данных.

**Приоритет:** высокий.

2. Пользователи, прошедшие регистрацию, могут пройти авторизацию путем ввода логина и пароля.

**Приоритет:** высокий.

3. Пользователи, прошедшие авторизацию, могут редактировать личную информацию.

**Приоритет:** высокий.



## Модуль №2 (Сервис ведения блога)

**Описание:** Сервис формирования пользователями маршрутов путешествий в формате блога.

**Роли пользователей:** Авторизованный пользователь, модератор.

### Требования:

1. Ведение пользователями блога посредством публикации маршрутов путешествий.

**Приоритет:** высокий.

2. Автоматическое нахождение и определение координат каждой достопримечательности с помощью Google Places API.

**Приоритет:** высокий.

3. Автоматическое определение общей длительности и протяженности маршрута на основе указанных пользователем видов транспорта посредством Google Maps API и Google Places API:

а) Выбор при составлении маршрута возможности автоматического получения примерных длительности и протяженности перемещений между достопримечательностями с использованием Google Maps API и Google Places API.

б) Показ в составленных маршрутах, помимо указанного составителем способа перемещения, альтернативных способов, а также примерного времени и расстояния перемещения для этих способов (полученных с помощью Google Maps API).

**Приоритет:** средний.

4. Интеграция с соц. сетями:

а) Возможность добавления фотографий из instagram в маршрут (суммарное количество добавленных фотографий из instagram и с устройства пользователя - не более 10).

б) Возможность поделиться ссылкой на маршрут в vkontakte

**Приоритет:** низкий.

## Модуль №3 (Сервис поиска пользователями маршрутов)

**Описание:** Сервис поиска пользователями маршрутов посещения достопримечательностей.

**Роли пользователей:** Гость, авторизованный пользователь, модератор.

**Требования:**

1. Поиск пользователями портала маршрутов:

- либо по населенному пункту (указывается радиус вокруг этого пункта);
- либо по входящим в маршрут достопримечательности;
- либо по нескольким населенным пунктам на протяжении маршрута;
- по длительности и протяженности маршрута (опционально);
- расстоянию, преодолеваемому в маршруте (опционально);
- количеству достопримечательностей в маршруте (опционально);
- завершенности (опционально);
- категории отдыха (семейный, активный, романтический, познавательный) (опционально);

**Приоритет:** средний.

2. Сортировка результатов поиска: по дате составления маршрута, по оценке маршрута.

**Приоритет:** средний.

3. Подбор похожих маршрутов по наибольшему количеству совпадающих с просматриваемым маршрутом достопримечательностей.

**Приоритет:** средний.

#### **Модуль №4 (Сервис комментирования и оценивания маршрутов)**

**Описание:** Сервис комментирования и оценивания просмотренных маршрутов (составленными другими пользователями) с возможностью подписки на блоги пользователей веб-портала.

**Роли пользователей:** Авторизованный пользователь, модератор.

**Требования:**

1. Возможность добавления под каждым маршрутом комментариев и оценивания полезности комментария другими пользователями (по системе “нравится/не нравится”)

**Приоритет:** высокий.

2. Возможность редактирования собственных комментариев.

**Приоритет:** высокий.

3. Возможность удаления собственных комментариев.

**Приоритет:** высокий.

4. Возможность удаления модераторами комментариев и маршрутов, нарушающих правила портала

**Приоритет:** высокий.

5. Подписка на новые маршруты в блоге определенного пользователя

**Приоритет:** средний.

6. Получение уведомлений о новых (добавленных после последнего посещения данным пользователем портала) маршрутах в блогах, на которые данный пользователь подписан.

**Приоритет:** средний.

## 6.2. Функциональные ограничения

1. Регистрация незарегистрированного пользователя (гостя) осуществляется путем заполнения следующих форм:

- никнейм\*;
- адрес электронной почты\*;
- логин для входа на веб-портал\*;
- пароль\*;
- подтверждение пароля\*;
- фотография (для аккаунта пользователя с его изображением) в формате .jpg или .png, размером не больше 6МБ.

\* - обязательные поля.

2. Блог пользователя веб-портала представляет собой набор созданных им маршрутов. При создании нового маршрута пользователь обязан указать:

- a) название маршрута;
- b) последовательность достопримечательностей, составляющих маршрут (не менее 2);
- c) даты посещения пользователем каждой достопримечательности;
- d) используемый способ перемещения между каждой парой достопримечательностей;
- e) оценку составляемого маршрута (по 5-балльной шкале);
- f) оценку посещенных достопримечательностей (по 5-балльной шкале);
- g) завершенность маршрута (составляется, завершен);

- h) категорию отдыха (семейный, активный, романтический, познавательный);
- i) указание на то, должен ли этот маршрут быть опубликован (опубликовать/скрыть).

Кроме того, пользователь при создании маршрута может поделиться впечатлениями о маршруте в целом и отдельных достопримечательностей, прикрепить фотографии (не более 10 штук). Поддерживаемые форматы изображений: jpg, png. Максимальный размер загружаемых фотографий - 6МБ.

3. Пользователь может формировать подписки на блоги других участников портала. После этого он будет получать уведомления о публикации новых маршрутов этих пользователей.

4. Оценивание маршрута может производиться как его составителем (согласно его впечатлениям, по 5-балльной шкале), так и другими пользователями, просмотревшими этот маршрут (по системе “нравится/не нравится”).

5. Пользователи и гости портала могут осуществлять поиск по созданным другими пользователями маршрутам. При этом пользователь уточняет:

- в районе какого населенного пункта (указывается радиус вокруг этого пункта) находятся достопримечательности маршрута;
- либо несколько входящих в маршрут достопримечательностей;
- либо несколько населенных пунктов на протяжении маршрута;

Опционально указывается:

- протяженность искомого маршрута (диапазон);
- длительность маршрута (диапазон);
- количество достопримечательностей в маршруте (диапазон);
- является ли маршрут завершенным или составляемым в данный момент;
- желаемая категория отдыха (семейный, активный, романтический, познавательный).

Также пользователь может указать параметры сортировки результатов поиска: сортировка по оценке создателя маршрута, по оценкам этого маршрута других пользователей (по системе “нравится/не нравится”), по дате создания.

6. Пользователь может включать в составляемый маршрут только достопримечательности, доступные через Google Places API.

7. Автоматическое определение общей длительности и протяженности маршрута на основе указанных пользователем видов транспорта осуществляется с той точностью, с которой это происходит при помощи Google Places API.

### **6.3. Системные требования**

#### **1. Требования к оборудованию**

##### **1.1 Для доступа клиентов к веб-порталу:**

1.1.1. Возможность подключения устройства клиента к сети Интернет;

1.1.2. Наличие на устройстве клиента веб-браузера;

##### **1.2. Для развертывания базы данных и серверного приложения:**

1.2.1. Рабочая станция на базе ОС семейства Linux Debian-based дистрибутива;

1.2.2. Возможность подключения устройства к сети Интернет;

#### **2. Требования к пользователям системы**

##### **2.1 К пользователям с ролью “Гость” и “Авторизованный пользователь”:**

2.1.1. Базовые навыки обращения с используемым устройством и пользования веб-сайтами;

##### **2.2. К пользователям с ролью “Модератор”:**

2.2.1. Навыки уверенного обращения с используемым устройством;

2.2.2. Базовое понимание внутренней структуры веб-портала и базы данных;

2.2.3. Знание основных правил модерации контента портала.

## **7. Определение процессов разработки**

### **7.1. Выбор модели жизненного цикла**

Модель жизненного цикла - **инкрементальная**.

Причины использования данной модели в нашем проекте следующие:

- Разработка в несколько итераций, в результате чего формируются более четкие требования посредством включения в процесс пользователей;
- Быстрое получение готового прототипа, что позволяет скорректировать функционал системы в процессе ее разработки;

- Получение в результате выполнения каждого инкремента функционального продукта;
- В конце каждой инкрементной поставки существует возможность пересмотреть риски, связанные с соблюдением установленного графика;
- Потребности клиента лучше поддаются управлению, поскольку время разработки каждого инкремента очень незначительно.

Другие модели жизненного цикла не используются по следующим причинам:

- Объем работ не позволяет использовать модель “проб и ошибок”;
- Требования могут корректироваться в процессе разработки в зависимости от предпочтений пользователей, что исключает возможность использования каскадной модели, которая запрещает возврат к предыдущим этапам;
- Спиральная модель не используется из-за необходимости владения разработчиками управлением рисками, а также данная модель пригодна лишь для внутренних проектов крупного размера;
- В объектно-ориентированной модели зачастую процесс коррекции преобладает над основными процессами – проектированием и анализом.

## 7.2. Выбор методологии разработки

Методология жизненного цикла - **SCRUM (гибкая методология разработки)**.

Причины использования данной методологии:

- Изменение требований заказчиком в результате участия на каждом этапе создания проекта позволяет создать более качественный продукт, в большей степени удовлетворяющий его.
- Возможность тесного взаимодействия с другими участниками команды, что позволяет держать коллектив в курсе каких-либо изменений.
- Каждая итерация называется спринтом. Во время выполнения спринта требования и цели спринта не могут быть изменены, но его можно остановить, если цели перестали быть актуальны или невозможно их выполнить в срок. А за счет краткости спринтов - 2-4 недели - достигается гибкость процесса разработки.

Жесткие методологии не используются по следующим причинам:

- Необходимости определения четких границ полномочий и ответственности каждого;
- Ориентации на предсказуемый процесс разработки с четко обозначенными целями;
- Необходимости ролевого разделения труда команды;
- Стремления жестких методологий обеспечить разработчиков инструкциями, не требующими обсуждения и корректировок.

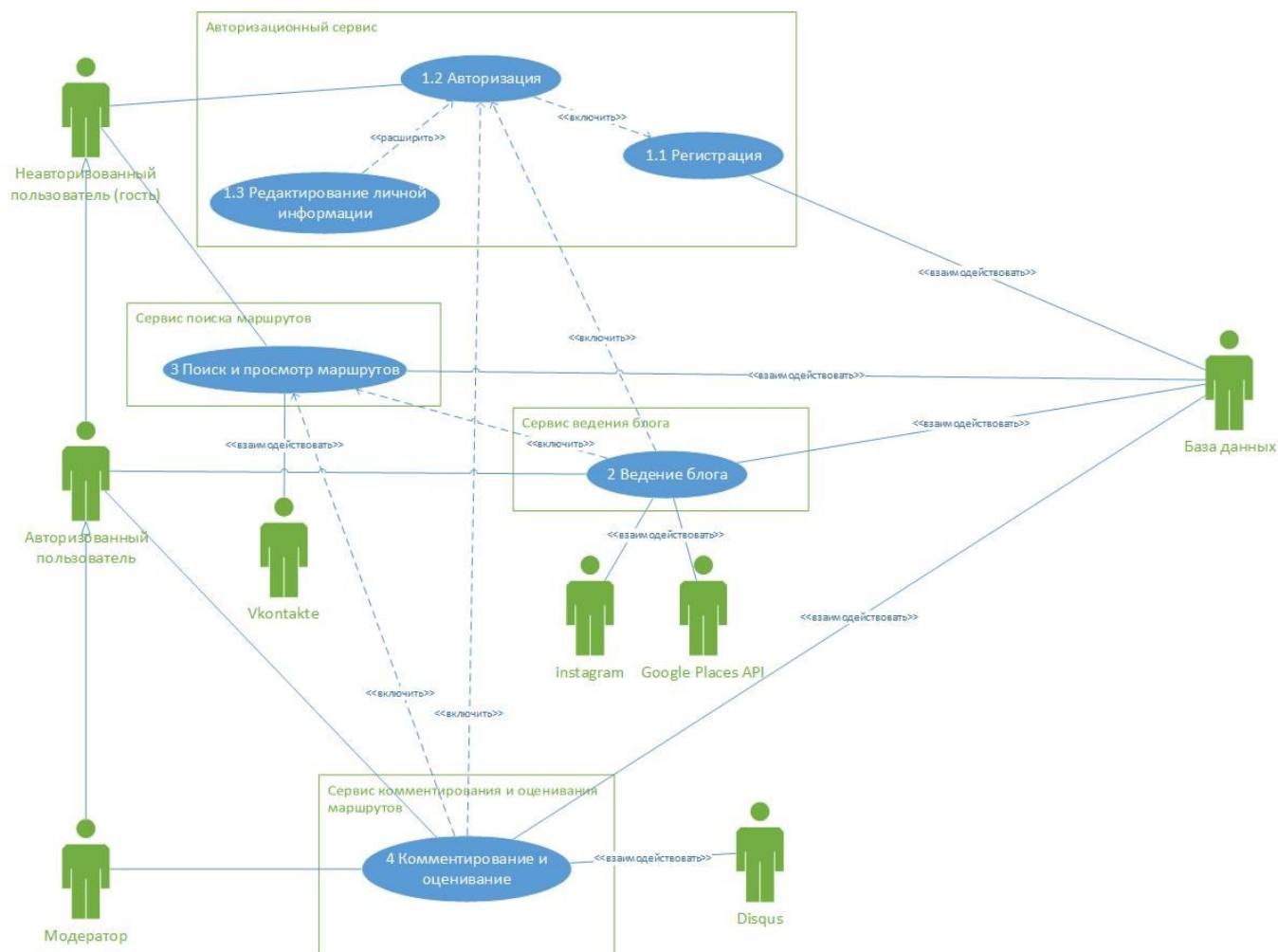
## **8. Проектирование системы**

### **8.1 Выбор CASE-средства для проектирования Use-case диаграммы**

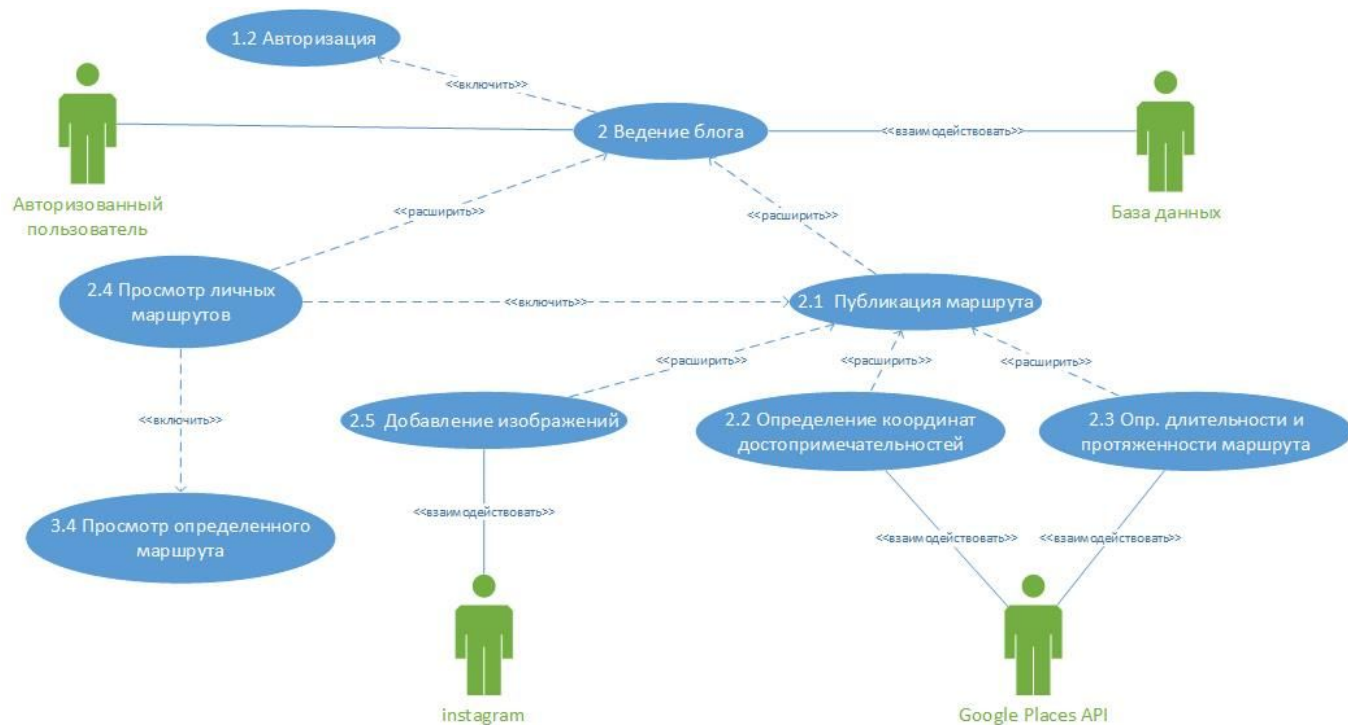
В качестве CASE-средства для проектирования диаграммы прецедентов (вариантов использования, use-case диаграммы) было выбрано универсальное средство моделирования данных и приложений Microsoft Office Visio 2013, поддерживающее создание объектно-ориентированных моделей сложных программных комплексов в рамках шаблона “Схема модели UML”.

### **8.2 Use-case диаграмма**

На схеме ниже представлена диаграмма вариантов использования(прецедентов) или use-case диаграмма, которая описывает множество всех актеров, взаимодействующих с системой при помощи так называемых вариантов использования (прецедентов).

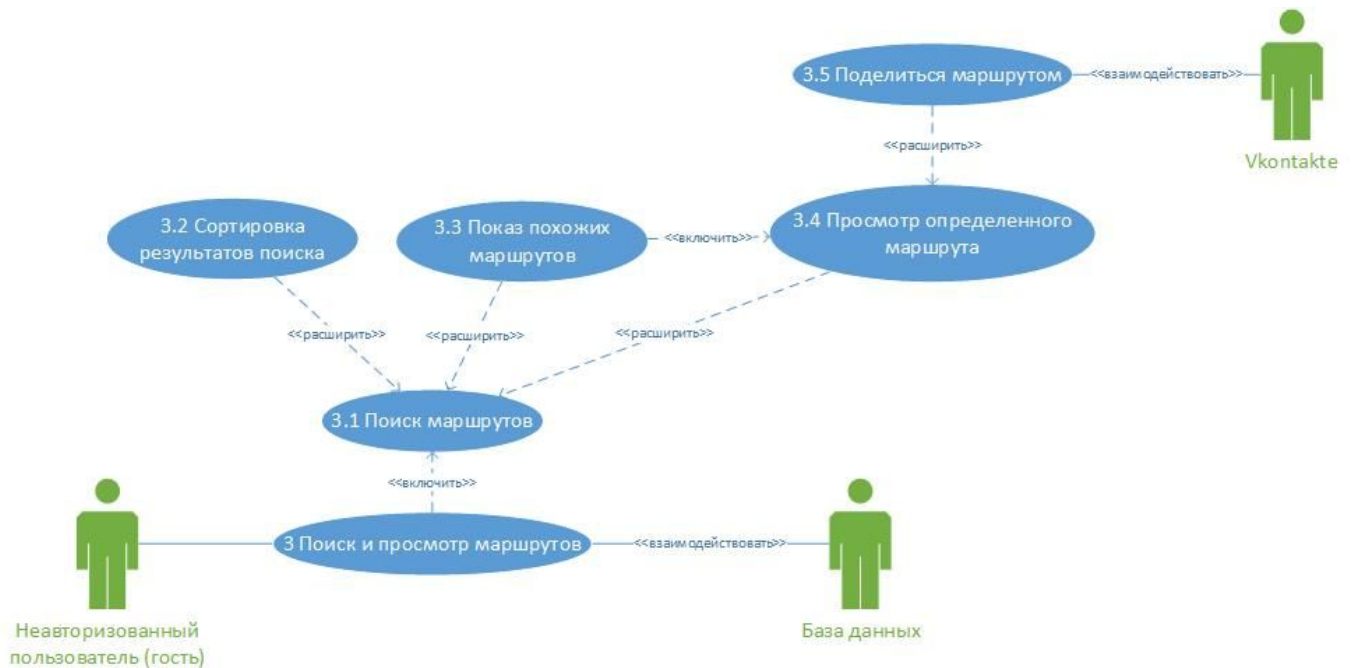


На следующей схеме представлена декомпозиция прецедента “2 Ведение блога” (подсистема “Сервис ведения блога”):

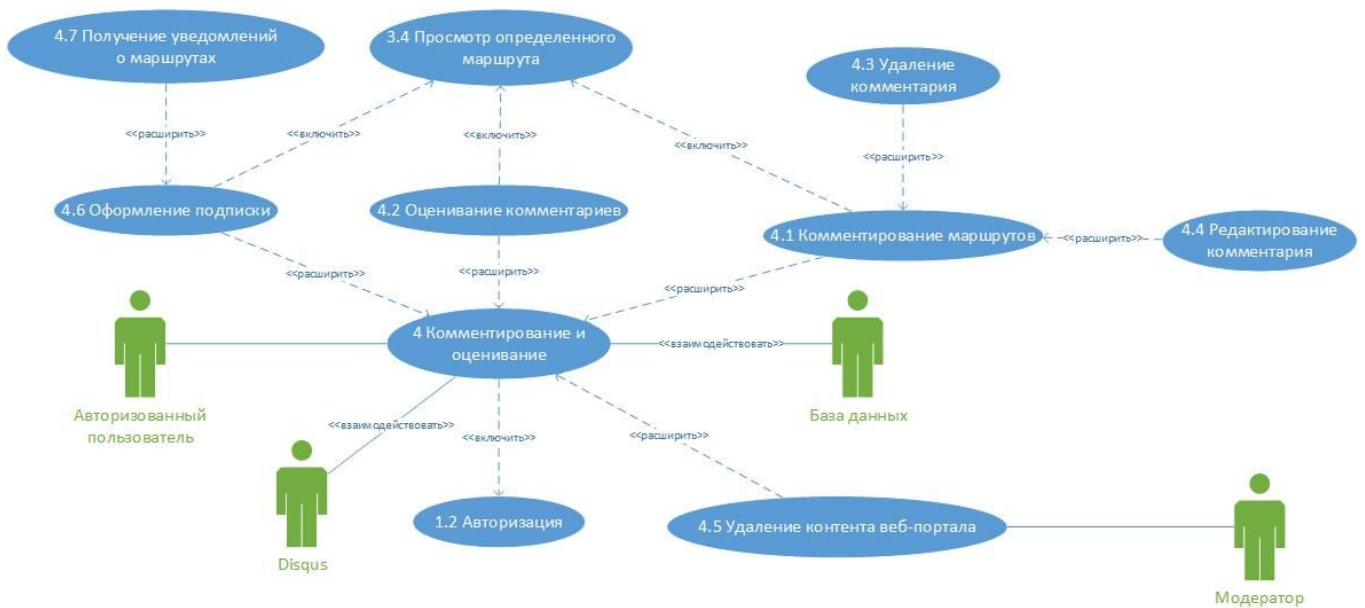




На схеме ниже представлена декомпозиция прецедента “3 Поиск и просмотр маршрутов” (подсистема “Сервис поиска маршрутов”):



На следующей схеме представлена декомпозиция прецедента “4 Комментирование и оценивание” (подсистема “Сервис комментирования и оценивания маршрутов”):



## 8.3 Use-case сценарии

### Модуль №1 (Авторизационный сервис)

#### 1.1 Регистрация незарегистрированного пользователя (гостя)

На диаграмме прецедентов: Регистрация

Контекст использования	Зарегистрироваться в системе для получения возможности последующей авторизации
Действующие лица	Гость, система
Вспомогательное действующее лицо	База данных
Основной сценарий	<p>1. Пользователь заполняет поля форм:</p> <ul style="list-style-type: none"><li>• никнейм*;</li><li>• адрес электронной почты*;</li><li>• логин для входа на веб-портал*;</li><li>• пароль* (не менее 8 символов, можно использовать только буквы латинского алфавита, обязательно использование хотя бы одной цифры, хотя бы одной буквы);</li><li>• подтверждение пароля*;</li><li>• фотография (для аккаунта пользователя с его изображением) в формате .jpg или .png, размером не больше 8МБ.</li></ul> <p>* - обязательные поля.</p> <p>2. Пользователь нажимает кнопку подтверждение регистрации</p>
Расширения	<p><i>a. “Внутренняя ошибка, попробуйте позже”</i> (Нет доступа к БД, система выдает сообщение 500 Internal Server Error) Результат: пользователь не может войти</p> <p><i>b. “Введенный логин занят, пожалуйста, выберите другой”</i> (Логин занят, система выдает сообщение 400 Bad Request) Результат: система предлагает выбрать другой логин</p> <p><i>c. “Email введен неверно, пожалуйста, введите существующий адрес электронной почты”</i> (Введенный адрес электронной почты некорректен, система выдает сообщение 400 Bad Request) Результат: система предлагает использовать другой электронный адрес</p>

Предусловие	Пользователь не имеет учетную запись, пользователь переходит на страницу регистрации
Постусловие	1. Создание в базе данных учетной записи пользователя 2. На почту пользователя отправляется сообщение с подтверждением регистрации 3. Переход к сценарию 1.2. “Авторизация зарегистрированного пользователя”

## 1.2 Авторизация зарегистрированного пользователя

На диаграмме прецедентов: Авторизация

Контекст использования	Авторизоваться в системе для дальнейшей работы с системой
Действующие лица	Зарегистрированный пользователь, система
Вспомогательное действующее лицо	База данных
Основной сценарий	1. Пользователь заполняет поля форм: <ul style="list-style-type: none"> <li>логин для входа на веб-портал;</li> <li>пароль;</li> </ul> 2. Пользователь нажимает кнопку авторизации
Расширения	<i>a. “Внутренняя ошибка, попробуйте позже”</i> (Нет доступа к БД, система выдает сообщение 500 Internal Server Error) Результат: пользователь не может войти <i>b. “Неверный логин или пароль”</i> (Неверный логин и/или пароль, система выдает сообщение 401 Unauthorized) Результат: Запрет авторизации
Предусловие	Успешное выполнение регистрации (сценарий 1.1. “Регистрация незарегистрированного пользователя”), пользователь перешел на страницу авторизации
Постусловие	1. Выдача пользователю авторизационного токена 2. Переход к сценарию 2.4. “Просмотр списка личных маршрутов путешествий”

### 1.3 Редактирование личной информации авторизованным пользователем

На диаграмме прецедентов: Редактирование личной информации

Контекст использования	Редактирование личной информации
Действующие лица	Авторизованный пользователь, модератор, система
Вспомогательное действующее лицо	База данных
Основной сценарий	<p>1. Пользователь вносит изменения в содержимое заполненных полей (хотя бы одного):</p> <ul style="list-style-type: none"><li>• никнейм*;</li><li>• адрес электронной почты*;</li><li>• пароль* (не менее 8 символов, можно использовать только буквы латинского алфавита, обязательно использование хотя бы одной цифры, хотя бы одной буквы);</li><li>• подтверждение пароля*;</li><li>• фотография (для аккаунта пользователя с его изображением) в формате .jpg или .png, размером не больше 8МБ.</li></ul> <p>* - обязательные поля (нельзя оставлять пустыми).</p> <p>2. Пользователь нажимает кнопку подтверждения изменений</p>
Расширения	<p>d. <i>“Внутренняя ошибка, попробуйте позже”</i> (Нет доступа к БД, система выдает сообщение 500 Internal Server Error) Результат: пользователь не может войти</p> <p>e. <i>“Email введен неверно, пожалуйста, введите существующий адрес электронной почты”</i> (Введенный адрес электронной почты некорректен, система выдает сообщение 400 Bad Request) Результат: система предлагает использовать другой электронный адрес</p>
Предусловие	Успешное выполнение авторизации (сценарий 1.2 “Авторизация зарегистрированного пользователя”), переход на страницу редактирования личной информации
Постусловие	Переход к сценарию 2.4. “Просмотр списка личных маршрутов путешествий”

## Модуль №2 (Сервис ведения блога)

На диаграмме прецедентов: Ведение блога

### 2.1 Публикация маршрута

На диаграмме прецедентов: Публикация маршрута

Контекст использования	Опубликовать новый маршрут
Действующие лица	Авторизованный пользователь, система
Вспомогательное действующее лицо	База данных
Основной сценарий	<p>1. Авторизованным пользователем заполнены поля для формирования маршрута:</p> <ul style="list-style-type: none"><li>• название маршрута;</li><li>• последовательность посещенных достопримечательностей (не менее двух);</li><li>• даты посещения достопримечательностей;</li><li>• информация о возможных способах перемещения между пунктами;</li><li>• оценка (для всего маршрута), представляемая в виде системы из трех оценок (по 5-балльной шкале) – оценка сложности прохождения, оценка культурной насыщенности маршрута и оценка развлекательной составляющей маршрута;</li><li>• отметка о завершенности/незавершенности;</li><li>• категория отдыха (семейный, активный, романтический, познавательный).</li><li>• информация о впечатлениях пользователя (фотографии, рассказ) - опционально;</li><li>• указание на то, должен ли этот маршрут быть опубликован (опубликовать/скрыть).</li></ul> <p>2. Автоматическим путем заполняются (см. функциональные требования № 2, 3, модуль № 2) координаты каждой достопримечательности, примерная общая длительность и протяженность маршрута.</p> <p>3. Пользователь нажимает кнопку подтверждения публикации нового маршрута</p>
Расширения	<p><i>а. “Внутренняя ошибка, попробуйте позже”</i> (Нет доступа к БД, система выдает сообщение 500 Internal Server Error) Результат: пользователь не может создать свой маршрут</p>

Предусловие	—
Постусловие	1. Добавление в базу данных информации о маршруте 2. Переход к сценарию 2.2 “Автоматическое определение координат каждой достопримечательности”

## 2.2 Автоматическое определение координат каждой достопримечательности

На диаграмме прецедентов: Определение координат достопримечательностей

Контекст использования	Получить данные о координатах достопримечательностей
Действующие лица	Авторизованный пользователь, система
Вспомогательное действующее лицо	База данных, Google Places API
Основной сценарий	1. Определена последовательность посещенных достопримечательностей (не менее двух) 2. Пользователь нажимает кнопку автоматического определения координат достопримечательностей
Расширения	—
Предусловие	Сценарий 3.4. “Публикация маршрута”
Постусловие	1. Автоматическое определение, отображение и сохранение координат каждой достопримечательности с помощью Google Places API 2. Переход к сценарию 2.3. “Автоматическое определение общей длительности и протяженности маршрута”

## 2.3 Автоматическое определение общей длительности и протяженности маршрута

На диаграмме прецедентов: Опр. длительности и протяженности маршрута

Контекст использования	Получение данных о длительности и протяженности маршрута
Действующие лица	Авторизованный пользователь, система

Вспомогательное действующее лицо	База данных, Google Places API
Основной сценарий	<ol style="list-style-type: none"> <li>1. Определена последовательность посещенных достопримечательностей (не менее двух)</li> <li>2. Пользователь нажимает кнопку автоматического определения общей длительности и протяженности маршрута</li> </ol>
Расширения	—
Предусловие	Сценарий 2.2. “Автоматическое определение координат каждой достопримечательности”
Постусловие	<ol style="list-style-type: none"> <li>1. Автоматическое нахождение и отображение длительности и протяженности маршрута, получаемое с помощью Google Places API.</li> <li>2. Обновление данных об альтернативных способах перемещения между достопримечательностями, а также добавление информации о длительности и протяженности участков маршрута для этих способов (полученных с помощью Google Maps API)</li> <li>3. Переход к сценарию 2.4. “Просмотр списка личных маршрутов путешествий”</li> </ol>

## 2.4 Просмотр списка личных маршрутов

На диаграмме прецедентов: Просмотр личных маршрутов

Контекст использования	Просмотр списка личных маршрутов пользователя
Действующие лица	Авторизованный пользователь, система
Вспомогательное действующее лицо	База данных
Основной сценарий	Пользователь просматривает личный список маршрутов
Расширения	—
Предусловие	Сценарий 1.2. “Авторизация зарегистрированного пользователя”
Постусловие	—

## 2.5 Добавления в маршрут фотографий

На диаграмме прецедентов: Добавление изображений

Контекст использования	Добавление в маршрут фотографий с устройства пользователя или из instagram
Действующие лица	Авторизованный пользователь, система
Вспомогательное действующее лицо	База данных, instagram
Основной сценарий	1. Пользователь выбирает изображения (не более 10 штук), которые желает закрепить за маршрутом 2. Пользователь нажимает кнопку подтверждения загрузки
Расширения	<i>a. “Сохранение файлов невозможно, попробуйте позже”</i> (Ошибка сохранения файлов, система выдает сообщение 500 Internal Server Error) Результат: пользователь не может прикрепить изображения к маршруту <i>b. “Внутренняя ошибка, попробуйте позже”</i> (Нет доступа к БД, система выдает сообщение 500 Internal Server Error) Результат: отмена загрузки файлов
Предусловие	Сценарий 2.1. “Публикация маршрута”
Постусловие	1. Загрузка выбранных фотографий в файловую систему и добавление ссылки на них в базу данных 2. Переход к сценарию 2.1. “Публикация маршрута” с отображением изображений в уменьшенном представлении



### Модуль №3 (Сервис поиска маршрутов)

На диаграмме прецедентов: Поиск и просмотр маршрутов

#### 3.1 Поиск маршрутов пользователями портала

На диаграмме прецедентов: Поиск маршрутов

Контекст использования	Поиск маршрута
Действующие лица	Гость, авторизованный пользователь, система
Вспомогательное действующее лицо	База данных
Основной сценарий	<p>1. Пользователь заполняет поля:</p> <p>а) <i>«Поиск по населенным пунктам»:</i></p> <ul style="list-style-type: none"><li>● Поле «Населенный пункт*» путем выбора названия населенного пункта из выпадающего меню. Пользователем может быть указан как один, так и несколько населенных пунктов (но не более пяти), включенных в маршрут.</li><li>● Для случая поиска маршрута по одному населенному пункту заполнено поле «Радиус удаленности*» от заданного населенного пункта на протяжении всего маршрута – указывается расстояние в километрах (целое число).</li></ul> <p>б) <i>«Поиск по достопримечательностям»:</i></p> <ul style="list-style-type: none"><li>● Поле «Достопримечательность*» путем выбора названия достопримечательности из выпадающего меню. Пользователем может быть указана как одна, так и несколько достопримечательностей (но не более пяти), включенных в маршрут.</li></ul> <p><i>Пользователем заполняются следующие опциональные фильтры поиска маршрутов:</i></p> <ul style="list-style-type: none"><li>● Диапазон протяженности маршрута (в километрах).</li></ul> <p>По умолчанию нижняя граница диапазона протяженности равна нулю, а верхняя граница диапазона – протяженности самого длинного маршрута, имеющегося в базе данных.</p> <ul style="list-style-type: none"><li>● Диапазон длительности маршрута.</li></ul> <p>По умолчанию нижняя граница диапазона длительности равна нулю, а верхняя граница диапазона – длительности самого продолжительного маршрута, имеющегося в базе данных.</p>

	<ul style="list-style-type: none"> <li>● Диапазон количества достопримечательностей, входящих в маршрут.</li> </ul> <p>По умолчанию нижняя граница диапазона равна единице, а верхняя граница диапазона – максимальному числу достопримечательностей маршрута, имеющегося в базе данных.</p> <ul style="list-style-type: none"> <li>● Завершенность маршрута: завершен / в путешествии /редактируется.</li> <li>● Категория отдыха: все/семейный/активный/романтический/экстремальный/познавательный.</li> </ul> <p>2. Нажатие кнопки подтверждения поиска маршрута с заданными параметрами</p>
Расширения	—
Предусловие	—
Постусловие	1. Осуществляется поиск среди всех имеющихся в базе данных маршрутов тех, что удовлетворяют заданным критериям поиска. В случае успешности поиска на экран выводится перечень найденных маршрутов, в противном случае – сообщение «Поиск не дал результатов».

### 3.2 Сортировка результатов поиска маршрута

На диаграмме прецедентов: Сортировка результатов поиска

Контекст использования	Сортировка результатов поиска
Действующие лица	Гость, авторизованный пользователь, система
Вспомогательное действующее лицо	База данных
Основной сценарий	<p>1. Пользователь нажимает кнопки выбора параметров сортировки:</p> <ul style="list-style-type: none"> <li>● По дате: от новых к старым / от старых к новым.</li> <li>● По рейтингу: по возрастанию / по убыванию.</li> </ul>
Расширения	—

Предусловие	1. Пользователь осуществил поиск маршрута по некоторым критериям (см. функциональные требования, модуль №3, требование 1) 2. Сценарий 3.1. “Поиск маршрутов пользователями портала”
Постусловие	1. Происходит изменение порядка следования маршрутов в списке результатов в соответствии с заданными пользователем параметрами сортировки. 2. Переход к сценарию 3.1. “Поиск маршрутов пользователями портала” с измененным порядком следования маршрутов

### 3.3 Подбор похожих маршрутов

На диаграмме прецедентов: Подбор похожих маршрутов

Контекст использования	Подбор похожих маршрутов
Действующие лица	Гость, авторизованный пользователь, система
Вспомогательное действующее лицо	База данных
Основной сценарий	1. Нажатие кнопки «Найти похожие маршруты»
Расширения	—
Предусловие	Сценарий 3.4. “Просмотр определенного маршрута”
Постусловие	Осуществляется поиск в базе данных маршрутов с наибольшим числом совпадающих с просматриваемым маршрутом достопримечательностей. На экран пользователя выводится список найденных маршрутов, отсортированный по убыванию степени схожести

### 3.4 Просмотр определенного маршрута

На диаграмме прецедентов: Просмотр определенного маршрута

Контекст использования	Показ определенного маршрута
Действующие лица	Гость, авторизованный пользователь, система

Вспомогательное действующее лицо	База данных
Основной сценарий	1. Пользователь просматривает страницу с конкретным маршрутом
Расширения	—
Предусловие	—
Постусловие	—

### 3.5 Возможность поделиться ссылкой на маршрут в vkontakte

На диаграмме прецедентов: Поделиться маршрутом

Контекст использования	Поделиться маршрутом с пользователями социальной сети vkontakte
Действующие лица	Авторизованный пользователь, система
Вспомогательное действующее лицо	База данных, vkontakte
Основной сценарий	Пользователь нажимает кнопку «Поделиться маршрутом»
Расширения	—
Предусловие	Сценарий 3.4. “Просмотр определенного маршрута”
Постусловие	Перенаправление на сайт vkontakte (с авторизацией пользователя, если это необходимо) с автоматической публикацией ссылки на выбранный маршрут на стене vkontakte.

## Модуль №4 (Сервис комментирования и оценивания маршрутов)

На диаграмме прецедентов: Комментирование и оценивание

### 4.1 Добавление комментариев к маршрутам

На диаграмме прецедентов: Комментирование маршрутов

Контекст использования	Добавление комментариев к маршруту
Действующие лица	Авторизованный пользователь, система
Вспомогательное действующее лицо	Disqus, База данных
Основной сценарий	Нажатие кнопки подтверждения публикации комментария
Расширения	—
Предусловие	<ol style="list-style-type: none"><li>1. В форме для добавления комментариев указаны:<ul style="list-style-type: none"><li>• Текст комментария* (максимальное количество символов – 16834);</li><li>• Графические изображения в формате .jpg или .png (не более 10 изображений для одного комментария).</li></ul></li><li>2. Сценарий 3.4. “Просмотр определенного маршрута”</li></ol>
Постусловие	Осуществляется проверка авторизации пользователя. Если пользователь авторизован, то комментарий успешно публикуется на странице маршрута, иначе – на экран выводится приглашения для авторизации пользователя в портале или через сторонние сервисы. В случае успешного прохождения авторизации, комментарий публикуется на странице маршрута.

### 4.2 Оценивание комментариев других пользователей к маршруту

На диаграмме прецедентов: Оценивание комментариев

Контекст использования	Оценивание комментариев других пользователей
Действующие лица	Авторизованный пользователь, система

Вспомогательное действующее лицо	Disqus, База данных
Основной сценарий	Нажатие кнопки выбора оценки: Нравится/Не нравится
Расширения	—
Предусловие	Сценарий 3.4. “Просмотр определенного маршрута”
Постусловие	Осуществляется проверка авторизации пользователя. Если пользователь авторизован – общий рейтинг комментария изменяется в зависимости от поставленной пользователем оценки, иначе – на экран выводится приглашения для авторизации пользователя в портале или через сторонние сервисы. В случае успешного прохождения авторизации, оценка комментария успешно фиксируется.

#### 4.3 Удаление своих комментариев к маршруту

На диаграмме прецедентов: Удаление комментария

Контекст использования	Удаление своих комментариев к маршруту
Действующие лица	Авторизованный пользователь, система
Вспомогательное действующее лицо	База данных, Disqus
Основной сценарий	Нажатие кнопки удаления комментария
Расширения	—
Предусловие	Сценарий 3.4. “Просмотр определенного маршрута”
Постусловие	Осуществляется проверка авторизации пользователя. Если пользователь авторизован, то производится запрос на подтверждение удаления комментария. В случае подтверждения удаления, комментарий удаляется.

#### 4.4 Редактирование своих комментариев к маршруту

На диаграмме прецедентов: Редактирование комментария

Контекст использования	Редактирование своих комментариев к маршруту
Действующие лица	Авторизованный пользователь, система
Вспомогательное действующее лицо	Disqus, База данных
Основной сценарий	Нажатие кнопки редактирования комментария
Расширения	—
Предусловие	Сценарий 3.4. “Просмотр определенного маршрута”
Постусловие	Осуществляется проверка авторизации пользователя. Если пользователь авторизован, то пользователю предоставляется возможность изменения комментария в поле редактирования комментария. В случае подтверждения пользователем изменений происходит сохранение новой версии комментария.

#### 4.5 Удаление отдельных комментариев и маршрутов пользователей

На диаграмме прецедентов: Удаление контента веб-портала

Контекст использования	Удаление отдельных комментариев и маршрутов пользователей
Действующие лица	Модератор, система
Вспомогательное действующее лицо	База данных, Disqus
Основной сценарий	1. Модератор просматривает информацию о конкретном маршруте или комментарии, который нарушает правила веб-портала 2. Нажатие кнопки подтверждения удаления маршрута или комментария
Расширения	—

Предусловие	1. Пользователь веб-портала авторизован как модератор 2. Сценарий 3.4. “Просмотр определенного маршрута”
Постусловие	1. Осуществляется удаление всей информации о маршруте или комментарии

#### 4.6 Оформление подписки на публикации новых маршрутов других пользователей

На диаграмме прецедентов: Оформление подписки

Контекст использования	Оформление подписки на публикации новых маршрутов других пользователей
Действующие лица	Авторизованный пользователь, система
Вспомогательное действующее лицо	База данных
Основной сценарий	Нажатие кнопки подтверждения подписки на публикации маршрутов от данного пользователя
Расширения	—
Предусловие	Сценарий 3.4. “Просмотр определенного маршрута”
Постусловие	—

#### 4.7 Рассылка уведомлений о публикации новых маршрутов пользователями

На диаграмме прецедентов: Получение уведомлений о маршрутах

Контекст использования	Рассылка уведомлений о публикации новых маршрутов
Действующие лица	Авторизованный пользователь, система
Вспомогательное действующее лицо	База данных
Основной сценарий	Пользователь произвел авторизацию на портале
Расширения	—



Предусловие	Пользователь портала оформил подписку на публикации маршрутов другого пользователя (см. функциональные требования, модуль №4, требование 4)
Постусловие	В разделе уведомлений отображается список всех уведомлений о публикациях новых маршрутов от пользователей, на которых оформлена подписка, с момента последнего посещения портала

## 8.4. Извлечение существительных

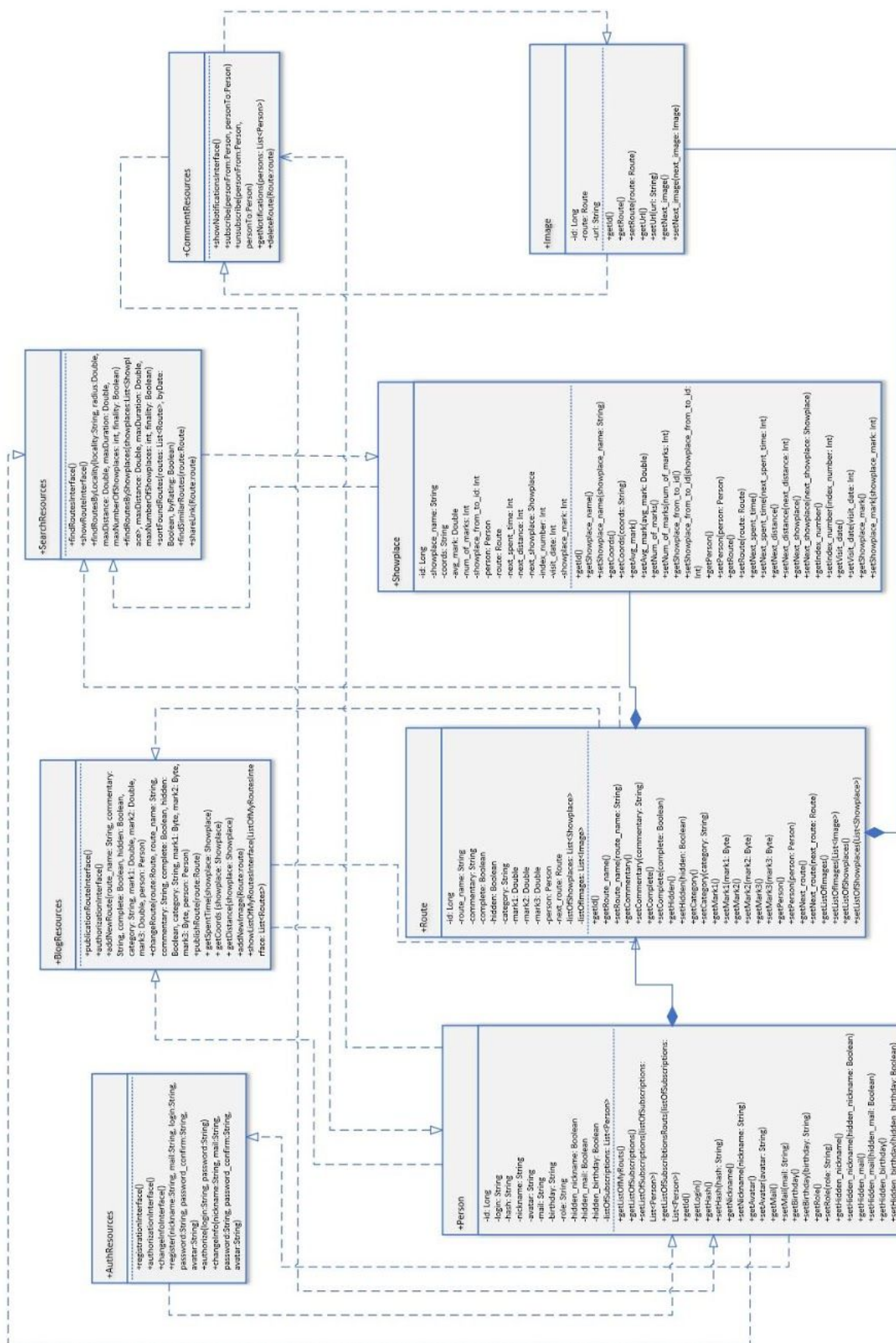
### 8.4.1. Первичное извлечение существительных из use-case сценариев

1. незарегистрированный пользователь
2. авторизованный пользователь
3. модератор
4. авторизация
5. регистрация
6. блог
7. маршрут
8. подписка
9. достопримечательность
10. публикация маршрута
11. участки маршрута
12. изображение
13. комментарий
14. подписка
15. перечень маршрутов
16. похожие маршруты

### 8.4.2. Перечень существительных-кандидатов в классы

1. пользователь
2. маршрут
3. подписка
4. достопримечательность
5. изображение
6. подписка

## 8.5. Диаграмма классов



## 9. База данных системы

- Разрабатываемый веб-портал имеет базу данных. База данных будет реляционной, состоять из нескольких таблиц и содержать следующие данные:

- а) Данные о зарегистрированных пользователях (логин и пароль, личные данные, подписки на других пользователей, роль - зарегистрированный пользователь или модератор);
- б) Маршруты (сам маршрут, логины пользователей, опубликовавших маршрут, оценки маршрута и достопримечательностей, ссылки на прикрепленные к маршруту фотографии);
- с) Комментарии (тексты комментариев, логины пользователей, оставивших комментарии, оценки комментариев, ссылки на маршрут, к которому относятся комментарии).

Все пользователи, включая гостей, имеют возможность взаимодействовать с БД, осуществляя поиск по достопримечательностям. При регистрации на портале данные о пользователях будут добавлены в БД. После этого авторизованные пользователи смогут взаимодействовать с БД, оставляя на портале маршруты и комментарии, а также их оценки. Кроме того, пользователи с ролью “модератор” будут иметь возможность удаления комментариев и маршрутов, нарушающих правила сайта.

В качестве системы управления базой данных была выбрана MySQL версии 5.7.20. Причинами выбора данной СУБД являются:

- бесплатность данной системы;
- наличие сервера и клиента MySQL в операционных системах на базе ядра Linux по умолчанию;
- простота настройки;
- быстроедействие по сравнению с другими бесплатными решениями;
- наличие у разработчиков опыта использования MySQL в прошлом.

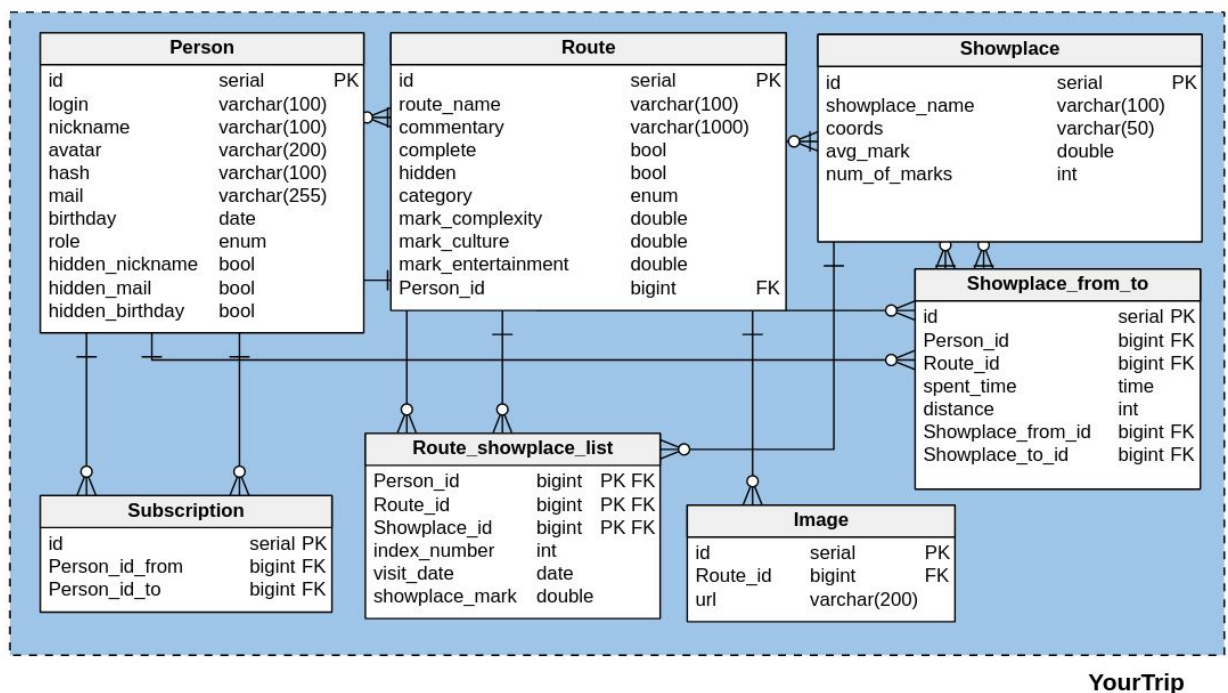
В качестве системы хранения данных будет использоваться система InnoDB, так как в ней поддерживаются внешние ключи, есть транзакционный движок, блокировка осуществляется на уровне строк. Выбранная СУБД содержит ограничения для системы хранения данных InnoDB: таблица может содержать максимум 1017 столбцов,

максимальный размер таблицы - 64ТБ, максимальное число одновременных подключений к MySQL - 151.

## 9.1. Выбор CASE-средства для проектирования модели базы данных

В качестве CASE-средства для проектирования модели базы данных был выбран инструмент Vertabelo. Основными причинами выбора данной системы стали ее бесплатность, отсутствие необходимости установки клиента (Vertabelo доступна онлайн), а также возможность совместной работы над моделью нескольких разработчиков.

## 9.2. Физическая модель базы данных системы



Краткое описание атрибутов таблиц базы данных:

- Таблица Person:
  - id - уникальный идентификатор пользователя
  - login - логин пользователя, предназначенный для авторизации на портале
  - nickname - отображаемое на портале имя пользователя

- avatar - урл изображения, которое пользователь установил в качестве аватара
- hash - хеш пароля пользователя
- mail - электронная почта пользователя
- birthday - дата рождения пользователя
- role - роль пользователя (зарегистрированный пользователь или модератор)
- hidden\_nickname - атрибут, показывающий, будет ли отображаться имя пользователя на его странице
- hidden\_mail - атрибут, показывающий, будет ли отображаться электронная почта пользователя на его странице
- hidden\_birthday - атрибут, показывающий, будет ли отображаться дата рождения пользователя на его странице
- Таблица Subscription:
  - id - уникальный идентификатор подписки одного пользователя на другого
  - person\_id\_to - идентификатор пользователя, на которого оформлена данная подписка
  - person\_id\_from - идентификатор пользователя, который оформил данную подписку
- Таблица Route:
  - id - уникальный идентификатор маршрута
  - route\_name - название маршрута
  - commentary - комментарий пользователя, составившего маршрут
  - complete - переменная, показывающая, полностью ли преодолен маршрут пользователем, или он в процессе пополнения
  - hidden - переменная, показывающая, скрыт ли маршрут для других пользователей (в случае, если пользователь, оставивший маршрут еще не закончил его составлять)
  - category - категория отдыха (семейный, активный, романтический, познавательный)
  - mark\_complexity - оценка сложности пройденного маршрута составившим его пользователем
  - mark\_culture - оценка культурной составляющей пройденного маршрута составившим его пользователем

- mark\_entertainment - оценка развлекательной составляющей пройденного маршрута составившим его пользователем
- Person\_id - идентификатор пользователя, составившего маршрут (внешний ключ)
- Таблица Route\_showplace\_list:
  - Person\_id - идентификатор пользователя, посетившего достопримечательность (внешний ключ)
  - Route\_id - идентификатор маршрута, к которому относится посещение данной достопримечательности (внешний ключ)
  - Showplace\_id - идентификатор посещенной достопримечательности (внешний ключ)
  - index\_number - номер достопримечательности в маршруте
  - visit\_date - дата посещения достопримечательности
  - showplace\_mark - оценка данным пользователем достопримечательности
- Таблица Showplace:
  - id - уникальный идентификатор достопримечательности
  - showplace\_name - название достопримечательности
  - coords - координаты достопримечательности
  - avg\_mark - средняя оценка достопримечательности
  - num\_of\_marks - количество оценок достопримечательности
- Таблица Showplace\_from\_to:
  - id - уникальный идентификатор перемещения между достопримечательностями
  - Person\_id - идентификатор пользователя (внешний ключ)
  - Route\_id - идентификатор маршрута, к которому относится перемещение (внешний ключ)
  - spent\_time - затраченное на перемещение время
  - distance - расстояние между достопримечательностями
  - showplace\_from\_id - идентификатор достопримечательности, из которой происходило перемещение (внешний ключ)
  - showplace\_to\_id - идентификатор достопримечательности, в которую происходило перемещение (внешний ключ)
- Таблица Image:

- id - уникальный идентификатор таблицы изображений
- route\_id - идентификатор маршрута (внешний ключ)
- url - адрес в сети, по которому размещено изображение

### 9.3. Оценка размеров базы данных

Отношение	Атрибут	Тип данных	Размер, байт	Среднее количес тво	Объем, байт
Person	# id	SERIAL	8	100.000	152.500 0.000
	# login	VARCHAR(100)	200		
	nickname	VARCHAR(100)	200		
	avatar	VARCHAR(200)	400		
	hash	VARCHAR(100)	200		
	mail	VARCHAR(255)	510		
	birthday	DATE	3		
	hidden_nickname	BOOL	1		
	hidden_mail	BOOL	1		
	hidden_birthday	BOOL	1		
	role	ENUM	1		

Route	# id	SERIAL	8	1.000.000	2.243.000.000
	route_name	VARCHAR(100)	200		
	commentary	VARCHAR(1000)	2000		
	complete	BOOL	1		
	hidden	BOOL	1		
	category	ENUM	1		
	mark_complexity	DOUBLE	8		
	mark_culture	DOUBLE	8		
	mark_entertainment	DOUBLE	8		
	# Person_id	BIGINT	8		
Showplace	# id	SERIAL	8	5.000.000	1.600.000.000
	showplace_name	VARCHAR(100)	200		
	coords	VARCHAR(50)	100		
	avg_mark	DOUBLE	8		
	num_of_marks	INTEGER	4		
Showplace_from_to	# id	SERIAL	8	5.000.000	235.000.000
	# Person_id	BIGINT	8		
	# Route_id	BIGINT	8		
	spent_time	TIME	3		



	distance	INTEGER	4		
	Showplace_from_id	BIGINT	8		
	Showplace_to_id	BIGINT	8		
Route_showplace_list	# Person_id	BIGINT	8	8.000.000	312.000.000
	# Route_id	BIGINT	8		
	# Showplace_id	BIGINT	8		
	index	INTEGER	4		
	visit_date	DATE	3		
	showplace_mark	DOUBLE	8		
Subscription	# id	SERIAL	8	2.000.000	48.000.000
	# Person_id_from	BIGINT	8		
	# Person_id_to	BIGINT	8		
Image	# id	SERIAL	8	3.000.000	1.248.000.000
	# Route_id	BIGINT	8		
	url	VARCHAR(200)	400		

Средний размер базы данных равен 5.838.500.000 байт, т. е. примерно 5.5 Гигабайтам.

## 10. Архитектура системы

### 10.1. Общее описание архитектуры системы

Для данного проекта выбрана клиент-серверная архитектура с тонким клиентом. Тонким клиентом является браузер пользователя.

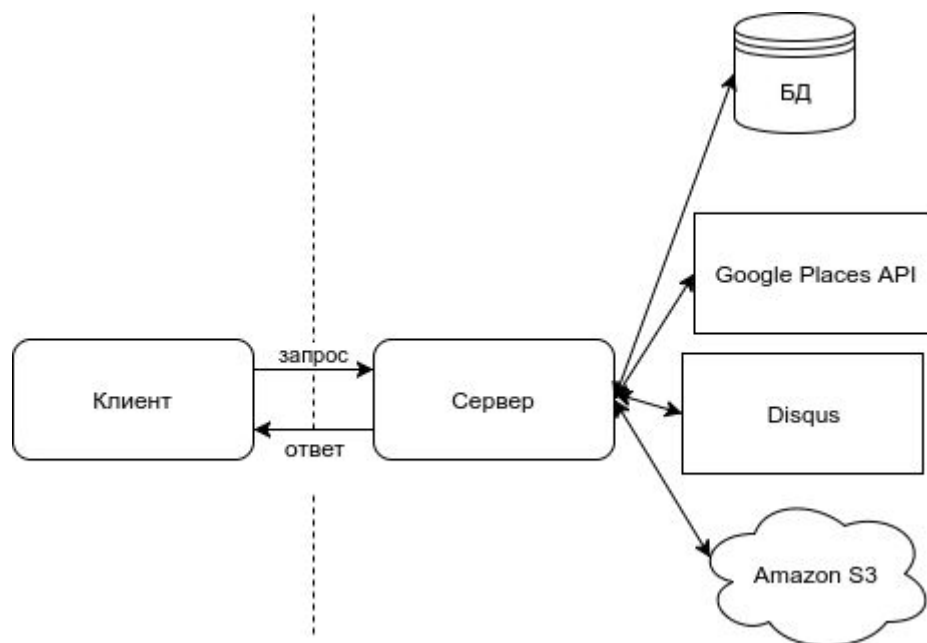


Рис.1. Визуальное представление архитектуры системы

Клиент-серверная архитектура выбрана, поскольку все пользователи системы работают с общей базой данных. Кроме того, в таком случае бизнес-логика приложения хранится на сервере, и тем самым избегается необходимость в загрузке пользователями на свои устройства кода для исполнения приложения, а также гарантируется невозможность внесения в систему данных вне рамок бизнес-правил. Отсутствие клиентского приложения позволяет вносить изменения в систему без необходимости загрузки пользователями обновленных версий программы. Наконец, клиентские приложения могут напрямую не общаться с внешними интерфейсами (в данном случае, Google Places API), и обработка получаемой из этих источников данных осуществляется на сервере.

Помимо непосредственно сервера, обрабатывающего HTTP-запросы, и базы данных, также система (с помощью сервера) будет производить запросы к внешним источникам - Google Places API для получения названий и координат достопримечательностей, Disqus для комментирования маршрутов и оценки комментариев и облачным хранилищем Amazon S3 для хранения фотографий, прикрепленных к маршрутам.

## 10.2. Описание акторов, взаимодействующих с системой

### 10.2.1. Google Places API

Google Places API - бесплатный веб-сервис, предоставляющий данные о достопримечательностях по всему миру. С помощью него может быть получена информация о конкретной достопримечательности, включая ее координаты и другую смежную информацию. С помощью данного API может быть также реализован интеллектуальный поиск по достопримечательностям, предлагающий подсказки мест по мере ввода их названия в строке поиска. Кроме того, в данном проекте используются функции для определения удаленности между несколькими достопримечательностями, получения альтернативных вариантов выбора способов перемещения между ними, а также вывод ожидаемого времени в пути. Эти функции также задействуют функционал данного сервиса.

### 10.2.2. Disqus

Disqus - сервис для подключения системы комментирования. Он помогает вовлекать аудиторию на сайт за счет того, что все комментарии сохраняются во внутренней социальной сети и пользователи предварительно авторизуются через какой-либо сервис. Кроме того, присутствует встроенная система модерации комментариев и легко настраиваемый функционал монетизации за счет показывания рекламных объявлений партнеров Disqus. Данный сервис используется для добавления возможности комментирования отдельных маршрутов пользователей.

### 10.2.3. Amazon RDS

Amazon Relational Database Service (Amazon RDS) - сервис, используемый для развертывания базы данных в облаке. Он позволяет легко настраивать, использовать и масштабировать реляционные базы данных. Сервис обеспечивает экономичное и масштабируемое использование ресурсов при одновременной автоматизации трудоемких задач администрирования, таких как выделение аппаратного обеспечения, настройка базы данных, установка исправлений и резервное копирование. Amazon RDS доступен в виде инстансов базы данных различных типов: оптимизированных по использованию памяти, по производительности или выполнению операций ввода-вывода. В разрабатываемом проекте используется инстанс база данных MySQL.

### 10.2.4. Amazon S3

Amazon S3 – это объектное хранилище, предназначенное для хранения и извлечения любых объемов данных из любых источников: веб-сайтов и мобильных приложений, корпоративных приложений, а также данных с датчиков или устройств IoT. Сервис гарантирует надежность хранения на уровне 99,999999999 % и используется для хранения данных миллионами приложений, применяемых лидерами рынка во всех отраслях. S3 открывает широчайшие возможности для обеспечения безопасности и соответствия согласно самым строгим нормативным требованиям. Сервис дарит клиентам гибкость в управлении данными, обеспечивая оптимизацию затрат, контроль доступа и соответствие требованиям. S3 – единственное облачное хранилище, обладающее функциональными возможностями для выполнения запросов к данным без извлечения,

что позволяет запускать мощную аналитику непосредственно для данных, хранимых в S3. Данный сервис используется для хранения изображений пользователей, включая аватары и фотографии мест, прикрепляемые к маршрутам.

### 10.3. Пропускная способность системы

Для развертывания приложения будет использоваться сервис Amazon EC2, предоставляющий для этого выделенные сервера (“инстансы”) по требованию. Типы инстансов, предлагаемые Amazon, различны для разных приложений, и для данного проекта было решено использовать инстанс типа m4.large, имеющий следующие параметры:

- 2 ЦПУ с частотой 2.4 ГГц
- 8 ГБ оперативной памяти
- Выделенная пропускная способность 450 Мбит/с
- HDD-хранилище объемом в 500 ГБ с пропускной способностью 250 операций ввода-вывода в секунду (или 250 МБ/с)
- около 110 тыс. одновременных входящих TCP-соединений (при этом непосредственно портал будет иметь поддержку не более 10 тыс. одновременных соединений из-за ограничений используемого сервлет-контейнера Apache Tomcat)

### История версий

	Изменения	Дата
1.	Добавлены ключевые особенности системы, определены модель жизненного цикла и методология разработки.	30.09.2017
2.	Добавлены концепция, проектный план. Описаны системные требования.	14.10.2017
3.	Добавлено описание функциональных модулей и use-case сценариев, уточнены требования и ограничения к системе, спроектирована физическая модель базы данных.	28.10.2017
4.	Добавлены use-case диаграмма, диаграмма классов, описание акторов и полосы пропускания сети, а также дополнен формат use-case сценариев.	10.11.2017