

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Институт Институт электронного обучения  
Направление подготовки 09.03.01 Информатика и вычислительная техника  
Кафедра Автоматики и Компьютерных Систем

**БАКАЛАВРСКАЯ РАБОТА**

Тема работы
Мобильное приложение «Путеводитель по Томску»

УДК 004.45191(036)(571.16)

Студент

Группа	ФИО	Подпись	Дата
3-8B2B1	Николаенко Александр Геннадьевич		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Баночкин П. И.			

**КОНСУЛЬТАНТЫ:**

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Рахимов Т.Р.	К.Э.Н		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Инженер	Маланова Н.В.	К.Т.Н		

**ДОПУСТИТЬ К ЗАЩИТЕ:**

И.о. зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
АиКС	Суходоев М.С.	К.Т.Н.		

Томск – 2017 г.

## ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

Код результата	Результат обучения (выпускник должен быть готов)
<i>Профессиональные компетенции</i>	
ПК1	Разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием.
ПК2	Осваивать методики использования программных средств для решения практических задач
ПК3	Разрабатывать интерфейсы "человек - электронно-вычислительная машина".
ПК4	Разрабатывать модели компонентов информационных систем, включая модели баз данных.
ПК5	Разрабатывать компоненты программных комплексов и баз данных, использовать современные инструментальные средства и технологии программирования.
ПК6	Обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности.
ПК7	Готовить презентации, научно-технические отчеты по результатам выполненной работы, оформлять результаты исследований в виде статей и докладов на научно-технических конференциях.
ПК8	Готовить конспекты и проводить занятия по обучению сотрудников применению программно-методических комплексов, используемых на предприятии.
ПК9	Участвовать в настройке и наладке программно-аппаратных комплексов.
ПК10	Сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем.
ПК11	Инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем.

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт Институт электронного обучения

Направление подготовки (специальность) 09.03.01 Информатика и вычислительная техника

Кафедра Автоматики и Компьютерных Систем

УТВЕРЖДАЮ:

Зав. кафедрой

\_\_\_\_\_  
(Подпись)      (Дата)      (Ф.И.О.)

**ЗАДАНИЕ**

**на выполнение выпускной квалификационной работы**

В форме:

бакалаврской работы

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
3-8B2B1	Николаенко Александр Геннадьевич

Тема работы:

Мобильное приложение «Путеводитель по Томску»

Утверждена приказом директора (дата, номер)

18.04.2017, №2773/с

Срок сдачи студентом выполненной работы:

30.05.2017

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

**Исходные данные к работе**

*(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).*

**Объект исследования** – программное приложение для мобильных устройств.

**Предмет исследования** – туристический путеводитель по городу для мобильных устройств.

**Цель исследования** – разработать мобильное приложение-путеводитель.

<p><b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b></p> <p><i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<ol style="list-style-type: none"> <li>1. Проанализировать и определить требования к программному приложению.</li> <li>2. Разработать архитектуру программного обеспечения с учетом требований.</li> <li>3. Выбрать мобильную платформу для реализации.</li> <li>4. Выбрать язык программирования.</li> <li>5. Изучить возможные технологии выбранной платформы.</li> <li>6. Разработка мобильного приложения и графического интерфейса пользователя.</li> </ol>
<p><b>Перечень графического материала</b> <i>(с точным указанием обязательных чертежей)</i></p>	<p>Презентация в формате *.pptx на __ слайдах.</p>
<p><b>Консультанты по разделам выпускной квалификационной работы</b> <i>(с указанием разделов)</i></p>	
<p style="text-align: center;"><b>Раздел</b></p>	<p style="text-align: center;"><b>Консультант</b></p>
<p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>	<p>Рахимов Тимур Рустамович</p>
<p>Социальная ответственность</p>	<p>Маланова Наталья Викторовна</p>
<p><b>Названия разделов, которые должны быть написаны на русском и иностранном языках:</b></p>	
<ul style="list-style-type: none"> <li>• Заключение.</li> </ul>	

<p><b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b></p>	<p>06.02.2017</p>
--	-------------------

**Задание выдал руководитель:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
<p>Ассистент</p>	<p>Баночкин П. И.</p>			

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
<p>3-8B2B1</p>	<p>Николаенко Александр Геннадьевич</p>		



**Министерство образования и науки Российской Федерации**  
 федеральное государственное автономное образовательное учреждение  
 высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
 ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт Институт электронного обучения  
 Направление подготовки (специальность) 09.03.01 Информатика и вычислительная техника  
 Уровень образования бакалавриат  
 Кафедра Автоматики и Компьютерных Систем  
 Период выполнения \_\_\_\_\_ (осенний / весенний семестр 2016/2017 учебного года)

Форма представления работы:

бакалаврская работа
---------------------

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН  
 выполнения выпускной квалификационной работы**

Срок сдачи студентом выполненной работы:	30.05.2017
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
	Выявление и анализ требований	10
	Модель предметной области	15
	Компонентная архитектура	15
	Проектирование и реализация уровня сервера приложений	20
	Проектирование и реализация уровня клиента	20
	Результаты развертывания (внедрения) и тестирования	10
	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	10
	Социальная ответственность	10

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Баночкин П. И.			

**СОГЛАСОВАНО:**

И.о. зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
АиКС	Суходоев М.С.	К.Т.Н.		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И  
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

<b>Группа</b>	<b>ФИО</b>
3-8B2B1	Николаенко Александр Геннадьевич

<b>Институт</b>	Институт электронного обучения	<b>Кафедра</b>	Автоматики и Компьютерных Систем
<b>Уровень образования</b>	Бакалавриат	<b>Направление/специальность</b>	Информатика и вычислительная техника

**Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:**

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	<i>Человеческие ресурсы: 2 человека</i>
2. <i>Нормы и нормативы расходования ресурсов</i>	

**Перечень вопросов, подлежащих исследованию, проектированию и разработке:**

1. <i>Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	<i>Оценка направлений развития проекта (проведение морфологического анализа);</i>
2. <i>Планирование и формирование бюджета научных исследований</i>	<i>Произведен расчет бюджета научных исследований.</i>
3. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	<i>Определена ресурсная, финансовая и экономическая эффективность НИ.</i>

**Перечень графического материала (с точным указанием обязательных чертежей):**

1 <i>Оценка конкурентоспособности технических решений</i>
2 <i>Матрица SWOT</i>
3 <i>Альтернативы проведения НИ</i>
4 <i>График проведения и бюджет НИ</i>
5 <i>Оценка ресурсной, финансовой и экономической эффективности НИ</i>

<b>Дата выдачи задания для раздела по линейному графику</b>	
---	--

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Доцент	Рахимов Т.Р.	К.Э.Н		

**Задание принял к исполнению студент:**

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
3-8B2B1	Николаенко Александр Геннадьевич		

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
3-8B2B1	Николаенко Александр Геннадьевич

Институт	Институт электронного обучения	Кафедра	Автоматики и Компьютерных Систем
Уровень образования	Бакалавриат	Направление/специальность	Информатика и вычислительная техника

### Исходные данные к разделу «Социальная ответственность»:

1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Объектом исследования является программное приложение-путеводитель для мобильных устройств. Результатом работы является мобильное приложения-путеводитель по городу Томску.
--	---

### Перечень вопросов, подлежащих исследованию, проектированию и разработке:

<b>1. Производственная безопасность</b> 1.1. Анализ вредных и опасных факторов, которые может создать объект исследования. 1.2. Анализ вредных и опасных факторов, которые могут возникнуть на производстве при внедрении объекта исследований. 1.3. Обоснование мероприятий по защите исследователя от действия опасных и вредных факторов.	Проводился анализ выявленных вредных факторов при разработке и эксплуатации программной системы Путеводитель по Томску, таких как: – повышенная напряженность электромагнитного поля; – отклонение показателей микроклимата; – недостаток естественного освещения; – недостаточная освещенность рабочей зоны; – повышенный уровень шума.
<b>2. Экологическая безопасность</b> 2.1. Анализ влияния объекта исследования на окружающую среду. 2.2. Анализ влияния процесса эксплуатации объекта на окружающую среду. 2.3. Обоснование мероприятий по защите окружающей среды.	Проведен анализ потребления электроэнергии. и влияния процесса исследования на окружающую среду: – Утилизация неисправной электроники; – Утилизация макулатуры.
<b>3. Безопасность в чрезвычайных ситуациях</b>	Возможные чрезвычайные ситуации на объекте. Защита в чрезвычайных ситуациях: – выбор наиболее типичной ЧС; – Реагирование при возникновении ЧС.
<b>4. Правовые и организационные вопросы обеспечения безопасности</b> 4.1. Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства. 4.2. Организационные мероприятия при компоновке рабочей зоны.	Правовые и организационные вопросы обеспечения безопасности: – Психофизические факторы; – Организационные мероприятия при компоновке рабочей зоны

### Дата выдачи задания для раздела по линейному графику

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Инженер	Маланова Н.В.	к.т.н		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
3-8B2B1	Николаенко Александр Геннадьевич		

## РЕФЕРАТ

Выпускная квалификационная работа 105 с., 32 рис., 26 табл., 53 источников, 1 прил.

Ключевые слова: путеводитель, достопримечательность, мобильное приложение, туристический путеводитель, проектирование.

Объектом исследования является (ются) программное приложение для мобильных устройств.

Цель работы – создание мобильного приложения-путеводитель по городу Томску.

В процессе исследования проводились анализ требований к мобильному приложению, сравнительный анализ мобильных операционных систем, анализ архитектур для реализации программного приложения.

В результате исследования было проектировано и разработано мобильное приложения-путеводитель по городу Томску.

Степень внедрения: готово к внедрению

Область применения: туристический путеводитель по городу Томску

Экономическая эффективность/значимость работы: разработка приложения-путеводитель является эффективной

В будущем планируется расширение функциональности приложения путем добавления авторизации пользователей через социальные сети и возможностью составления собственных маршрутов.

## ОГЛАВЛЕНИЕ

<b>Введение.....</b>	<b>11</b>
<b>1 Выявление и анализ требований .....</b>	<b>13</b>
<b>2 Модель предметной области.....</b>	<b>24</b>
<b>3 Компонентная архитектура .....</b>	<b>27</b>
<b>3.1 Выбор технологий и инструментов реализации.....</b>	<b>31</b>
<b>4 Реализация программной системы .....</b>	<b>36</b>
<b>4.1 Проектирование и реализация уровня сервера приложений</b> <b>36</b>	
<b>4.2 Проектирование и реализация уровня клиента .....</b>	<b>39</b>
<b>4.3 Результаты развертывания (внедрения) и тестирования ..</b>	<b>55</b>
<b>5 Финансовый менеджмент, ресурсоэффективность и</b> <b>ресурсосбережение .....</b>	<b>59</b>
<b>5.1 Цель работы и задачи .....</b>	<b>59</b>
<b>5.2 Техническое задание на разработку приложения .....</b>	<b>60</b>
<b>5.3 Актуальность проекта. ....</b>	<b>60</b>
<b>5.4 Оценка коммерческого потенциала и перспективности</b> <b>проведения научных исследований с позиции ресурсоэффективности и</b> <b>ресурсосбережения.....</b>	<b>61</b>
5.4.1 Потенциальные потребители результатов исследования .....	61
5.4.2 Анализ конкурентных технических решений.....	61
5.4.3 Технология QuaD .....	62
5.4.4 SWOT-анализ .....	63
<b>5.5 Определение возможных альтернатив проведения научных</b> <b>исследований.....</b>	<b>66</b>
<b>5.6 Планирование научно-исследовательских работ .....</b>	<b>67</b>
5.6.1 Структура работ в рамках научного исследования .....	67
5.6.2 Определение трудоемкости выполнения работ .....	68
5.6.3 Разработка графика проведения научного исследования .....	69
<b>5.7 Бюджет научно-технического исследования .....</b>	<b>75</b>
5.7.1 Расчет материальных затрат НИТ .....	75

5.7.2	Расчет основной заработной платы исполнителей системы	76
5.7.3	Отчисления во внебюджетные фонды (страховые отчисления).....	79
5.7.4	Расчет накладных расходов .....	79
5.7.5	Формирование бюджета затрат научно-исследовательского проекта	80
<b>5.8</b>	<b>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования.....</b>	<b>80</b>
<b>6</b>	<b>Социальная ответственность .....</b>	<b>85</b>
	<b>Введение.....</b>	<b>85</b>
<b>6.1</b>	<b>Производственная безопасность .....</b>	<b>85</b>
6.1.1	Повышенная напряженность электромагнитного поля .....	86
6.1.2	Отклонение показателей микроклимата .....	87
6.1.3	Недостаточная освещенность .....	89
6.1.4	Повышенный уровень шума; .....	89
<b>6.2</b>	<b>Экологическая безопасность .....</b>	<b>90</b>
<b>6.3</b>	<b>Безопасность в чрезвычайных ситуациях .....</b>	<b>91</b>
6.3.1	Перечень возможных чрезвычайных ситуаций на объекте..	91
6.3.2	Меры по ликвидации чрезвычайных ситуаций и последствий	93
<b>6.4</b>	<b>Правовые и организационные вопросы обеспечения безопасности.....</b>	<b>94</b>
6.4.1	Психофизиологические факторы .....	94
6.4.2	Организационные рабочего места.....	94
	<b>Заключение .....</b>	<b>96</b>
	<b>Conclusion .....</b>	<b>97</b>
	<b>Список использованных источников .....</b>	<b>98</b>
	<b>Приложение А.....</b>	<b>104</b>

## ВВЕДЕНИЕ

Путеводитель (или гид) – печатный, электронный или аудиовизуальный справочник содержащий сведения о каком-нибудь регионе, городе, историческом месте или туристическом маршруте [1]. Большой энциклопедический словарь определяет путеводитель как «справочное издание, содержащее сведения о стране, городе, туристском маршруте, историко-художественных памятниках и т. п.» [2]. Путеводитель предназначен для знакомства туриста с не известными ему местами, маршрутами, достопримечательностям описываемой местности.

Для города Томска существует много путеводителей в виде бумажных изданий, например, замечательная книга «Томск. Иллюстрированный путеводитель-дайджест». В виде электронных изданий, сайтов, например, «Travel-Tomsk - Туризм в Томске и Томской области / Туристский портал Томска и Томской области» [3].

В современном мире, когда невозможно представить жизнь без мобильных устройств, которые всегда под рукой, возникает необходимость в мобильном путеводителе, который будет содержать самую свежую информацию о ближайших мероприятиях, известных и узнаваемых местах, интересных маршрутах по достопримечательностям города.

С бумажными изданиями возникают сложности в доступности и обновлении информации. У туриста, который приехал на одни-два дня нет времени на поиск бумажного издания. А в брошюрах, которые можно найти на прилавках газетных киосках, может быть недостаточно подробная информация.

У сайтов по сравнению с мобильным приложением есть ряд недостатков [4]:

1. Возможность использования в offline режиме или наличие интернет соединения. Мобильный сайт требует постоянного соединения с сетью, т.к. запускается из браузера. При переходе между частями приложения могут возникать неприятные задержки из-за особенностей мобильного

доступа в интернет. Мобильные приложения могут осуществлять работу без подключения к интернету, выполняя обновление данных при появлении соединения, если это требуется.

2. Интерфейс или интерактивность при взаимодействии с пользователем. Один из основных аргументов в пользу мобильного приложения это наиболее привычный и близкий к ОС интерфейс для пользователя. Мобильный сайт может включать в себя интерактивные компоненты с использованием HTML5, JavaScript, новых API браузеров. Но самые активными пользователя считаются те, кто активно использует последние новшества мобильной ОС.
3. Удобство при регулярном использовании и быстроедействие. Мобильное приложение направлено на конечного пользователя, в нем больше возможностей для доступа к устройству. Мобильный сайт существенно уступает с точки зрения быстрогодействия. Производительность браузеров с каждым годом растет, но пока не может порадовать высокими показателями. Мобильно приложение с излишней анимацией и сложным интерфейсом значительно снижают спрос на приложение.

На рынке мобильных приложений очень мало приложений, которые позиционируют, как подробный, постоянно обновляемый путеводитель по городу Томску для туристов и жителей города. В большинстве случаев, гид по городу Томску предлагается в группе с другими путеводителями и содержит устаревшую информацию о достопримечательностях и мероприятиях города.

Целью данной работы является создание мобильного приложения-путеводитель по городу Томску, позволяющему пользователю иметь всегда под рукой информацию о ближайших мероприятиях, список достопримечательностей города, интересных маршрутах. Мобильное приложение позволит обмениваться впечатлениями и эмоциями о путешествии при помощи комментариев к конкретному месту или маршруту, оценивать на сколько посещённое им место интересно и заслуживает внимание других путешественников.



## 1 Выявление и анализ требований

Для выявления необходимых требований осуществим анализ уже имеющихся на рынке мобильных приложений-путеводителей специальных для города Томска или включающих город Томск в общем списке городов. Были выбраны три приложения, которые получилось найти в магазинах приложений с путеводителем для города Томска. Сравнительный анализ приложений приведен в таблице 1. Все приложения доступны для мобильных платформ Android и iOS, и доступны для загрузки в App Store и Google play.

Таблица 1 – Сравнение мобильных приложений – путеводителей

	TopTripTip	TripAdvisor	Google Trips
Краткое описание	Созданный совместно с Телерадиокомпанией "МИР" – гид для крупных и самых отдаленных уголков России	Автономный доступ к отзывам, фотографиям и картам более 300 городов по всему миру	Приложение для путешественников, которое призвано предоставить всё, что может понадобиться в дороге
Места и достопримечательности	Достопримечательности, природные объекты, популярные туристские места	Достопримечательности, отели, рестораны	Достопримечательности, транспорт, гостиницы, рестораны
Новости и мероприятия	Устаревшие	Нет	Раздел «Надо знать», но не для города Томск
Маршруты	Есть, но не для города Томск	Нет	Раздел «План на день», но не для города Томск
Работа с картой города	Места и достопримечательности на карте	Места и достопримечательности на карте.	Места, достопримечательности, маршруты указываются на карте
Работа в offline режиме	Есть возможность загрузить гид целиком в память устройства	Сохранение карт, отзыв и закладок	Локальное хранение загруженных данных

Продолжение таблицы 1

	TopTripTip	TripAdvisor	Google Trips
Социальное взаимодействие	Пометка «Нравится», комментарии	Возможность оценить место и написать комментарий. Поделиться местом в соц. сетях или по почте.	Показывается рейтинг места и отзывы пользователь оставленные в Google Maps

В результате анализа рассмотренных приложений-путеводителей были выявлены их достоинства и недостатки. Достоинствами данных приложения является наличие офлайн режима и социальное взаимодействие. Недостатками рассмотренных приложений являются устаревшие данные о мероприятиях и отсутствия маршрутов по достопримечательностям города Томска. На основании выявленных достоинств и недостатков был сформирован набор функциональных требований:

- Отображение списка мероприятий;
- Отображение детальной информации о мероприятии;
- Отображение списка достопримечательностей;
- Отображение детальной информации о достопримечательности;
- Отображение списка маршрутов;
- Отображение детальной информации о маршруте;
- Хранение комментариев пользователем;
- Хранение общего рейтинга популярности достопримечательности;
- Хранение общего рейтинга популярности маршрута;
- Хранение избранных мероприятий, достопримечательностей, маршрутов;
- Публикация информации о мероприятии, достопримечательности, маршруте в социальных сетях.

На рисунке 1 представлена обобщенная диаграмма использования.



Рисунок 1 – UML диаграмма использования

На диаграмме представлены следующие варианты использования:

- Просмотреть список мероприятий, достопримечательностей, маршрутов;
- Просмотреть детальную информацию о мероприятии, достопримечательности, маршруте;
- Написать комментарий к достопримечательности или маршруту;
- Оценить достопримечательность или маршрут;
- Добавить/удалить в/из избранное(ого) достопримечательность или маршрут;

- Поделиться достопримечательностью или маршрутом в социальных сетях;
- Просмотреть местоположение достопримечательности или маршрута на карте.

На рисунке 2 представлена уточняющая диаграмма использования «Управление избранными элементами».

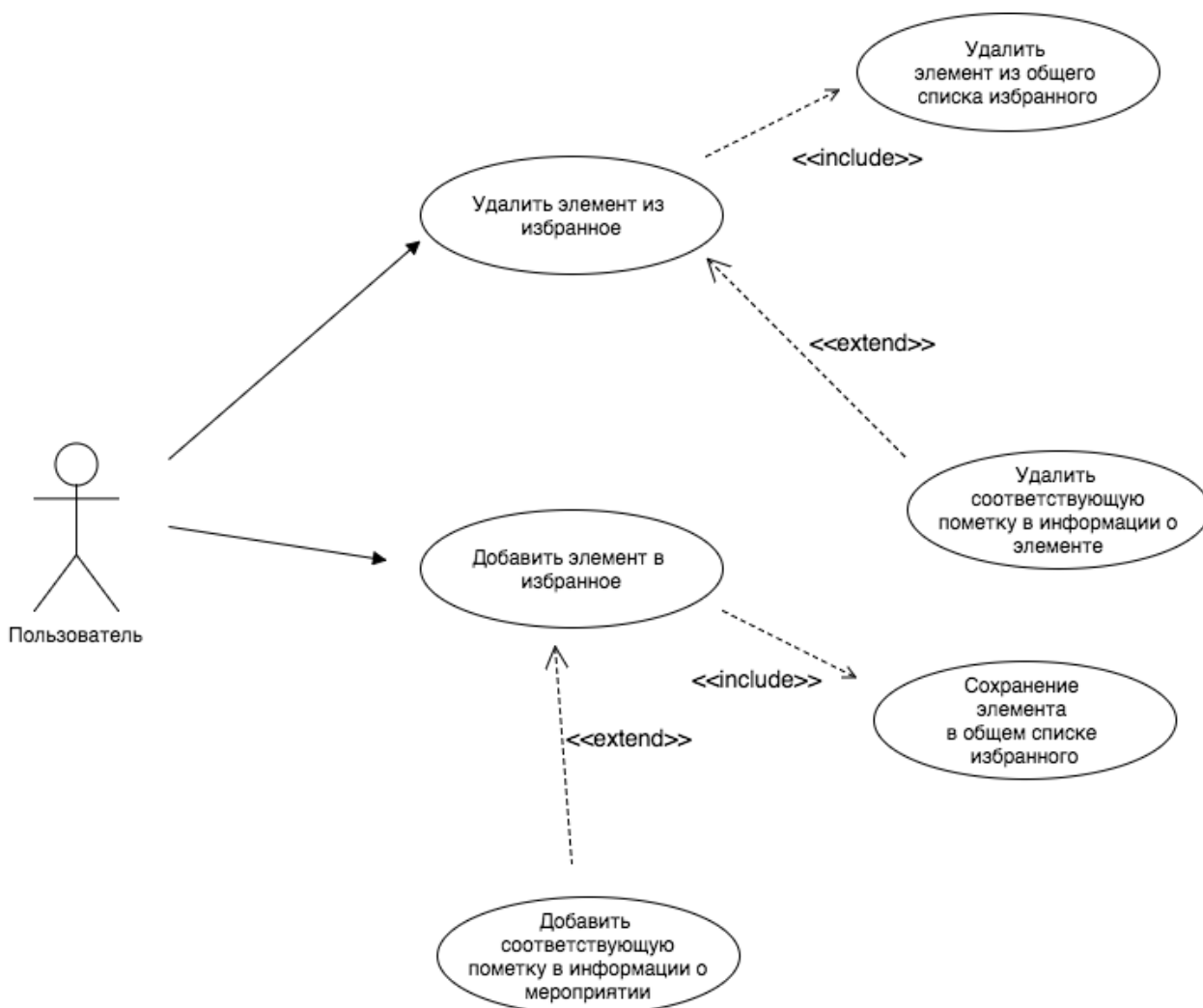


Рисунок 2 – UML диаграмма использования «Управление избранными элементами».

На диаграмме используются следующие типы отношений [5][6]:

- ассоциация между действующим лицом и вариантом использования;
- включение варианта использования;
- расширение варианта использования.

На диаграмме представлены обобщенные варианты использования «Добавить элемент в избранное», «Удалить элемент из избранного».

Вариант использования «Добавить элемент в избранное» включает «Сохранение элемента в общем списке избранного».

Вариант использования «Добавить соответствующую пометку в информации о элементе» расширяет вариант использования «Добавить элемент в избранное».

Вариант использования «Удалить элемент из избранного» включает «Удаление элемента из общего списка избранного».

Вариант использования «Удалить соответствующую пометку в информации о элементе» расширяет вариант использования «Удалить элемент в избранное».

**Детализируется требования варианта использования просмотра мероприятий.** Список требований варианта использования «Просмотреть список достопримечательностей»:

- Список мероприятий состоит из элементов, содержащих фотографию, дату и название мероприятия;
- По нажатию на элемент списка выполняется вариант использования «Просмотр детальной информации о мероприятии»;

Список требований варианта использования «Просмотр детальной информации о мероприятии», должно быть отражено:

- Фотографии мероприятия;
- Название мероприятия;
- Описание мероприятия;
- Дата мероприятия;
- Находиться ли мероприятие в избранном.

На рисунке 3 представлены варианты использования «Просмотр списка мероприятий», «Просмотр детальной информации о мероприятии», «Просмотр местоположения достопримечательности на карте».

**Детализируется требования варианта использования просмотра достопримечательностей.** Список требований варианта использования «Просмотреть список достопримечательностей»:

- Список достопримечательностей состоит из элементов, содержащих фотографию, название и рейтинг достопримечательности;
- По нажатию на элемент списка выполняется вариант использования «Просмотр детальной информации о достопримечательности»;

Список требований варианта использования «Просмотр детальной информации о достопримечательности», должно быть отражено:

- Фотографии достопримечательности;
- Название достопримечательности;
- Описание достопримечательности;
- Рейтинг достопримечательности;
- Комментарии пользователей;
- Возможность просмотреть достопримечательность на карте;
- Возможность поделиться информацией о достопримечательности с соц. сетях;
- Находиться ли достопримечательность в избранном.

На рисунке 4 представлены варианты использования «Просмотр списка достопримечательностей», «Просмотр детальной информации о достопримечательности», «Просмотр местоположения достопримечательности на карте».

**Детализируется требования варианта использования просмотра маршрутов.** Список требований варианта использования «Просмотреть список маршрутов»:

- Список маршрутов состоит из элементов, содержащих фотографию, название и рейтинг достопримечательности;

- По нажатию на элемент списка выполняется вариант использования «Просмотр детальной информации о маршруте»;

Список требований варианта использования «Просмотр детальной информации о маршруте», должно быть отражено:

- Название маршрута;
- Описание маршрута;
- Рейтинг маршрута;
- Комментарии пользователей;
- Возможность просмотреть маршрут на карте;
- Возможность поделиться информацией о маршруте с соц. сетях;
- Находиться ли маршрут в избранном.

На рисунке 5 представлены варианты использования:

- «Просмотр списка достопримечательностей»;
- «Просмотр детальной информации о достопримечательности»;
- «Просмотр местоположения достопримечательности на карте»;
- «Написать комментарий к элементу»;
- «Оценить элемент».

В вариантах использования «Написать комментарий к элементу» и «Оценить элемент» под элементом может выступать достопримечательность или маршрут.

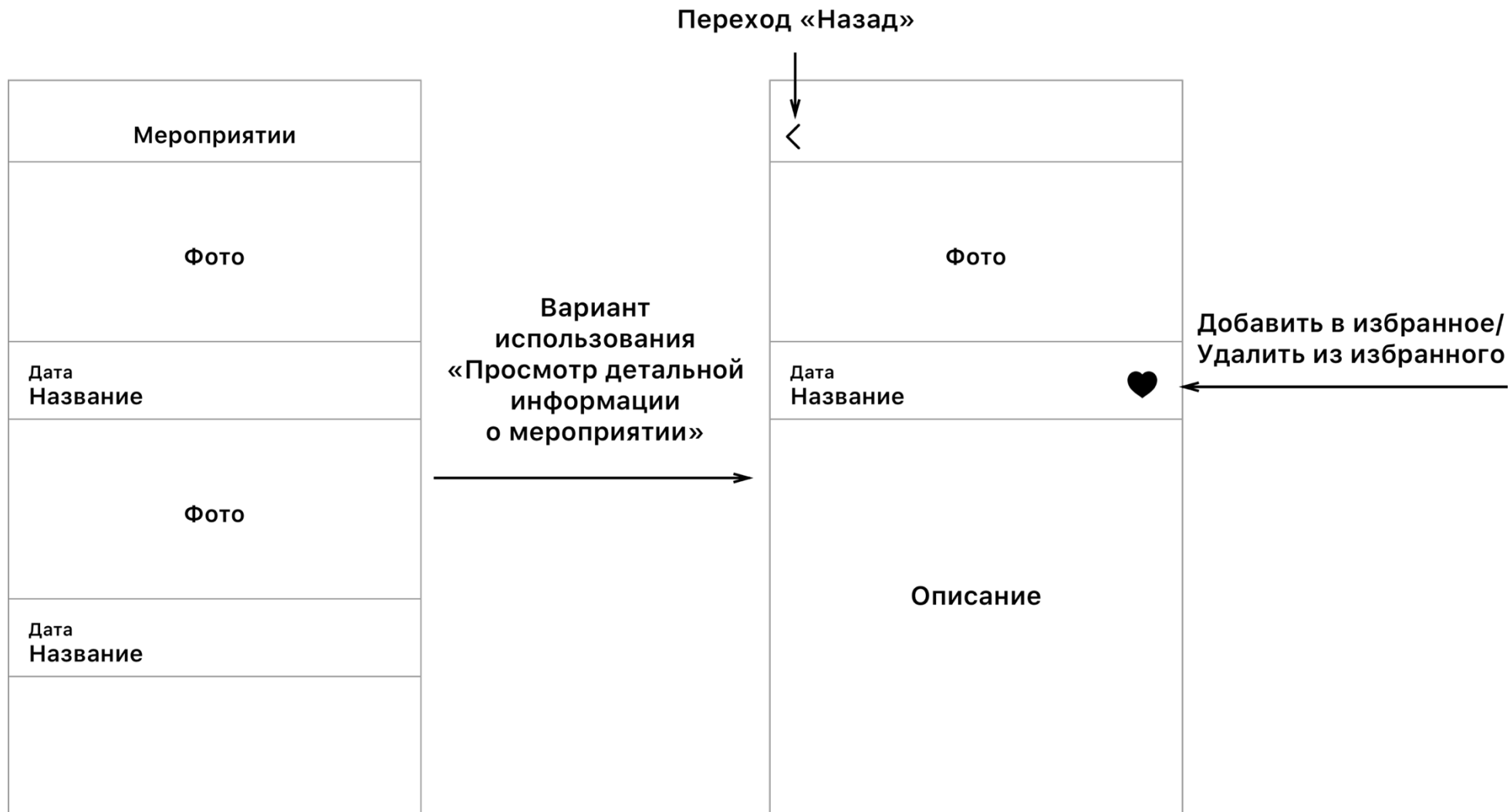


Рисунок 3 – Просмотр мероприятий



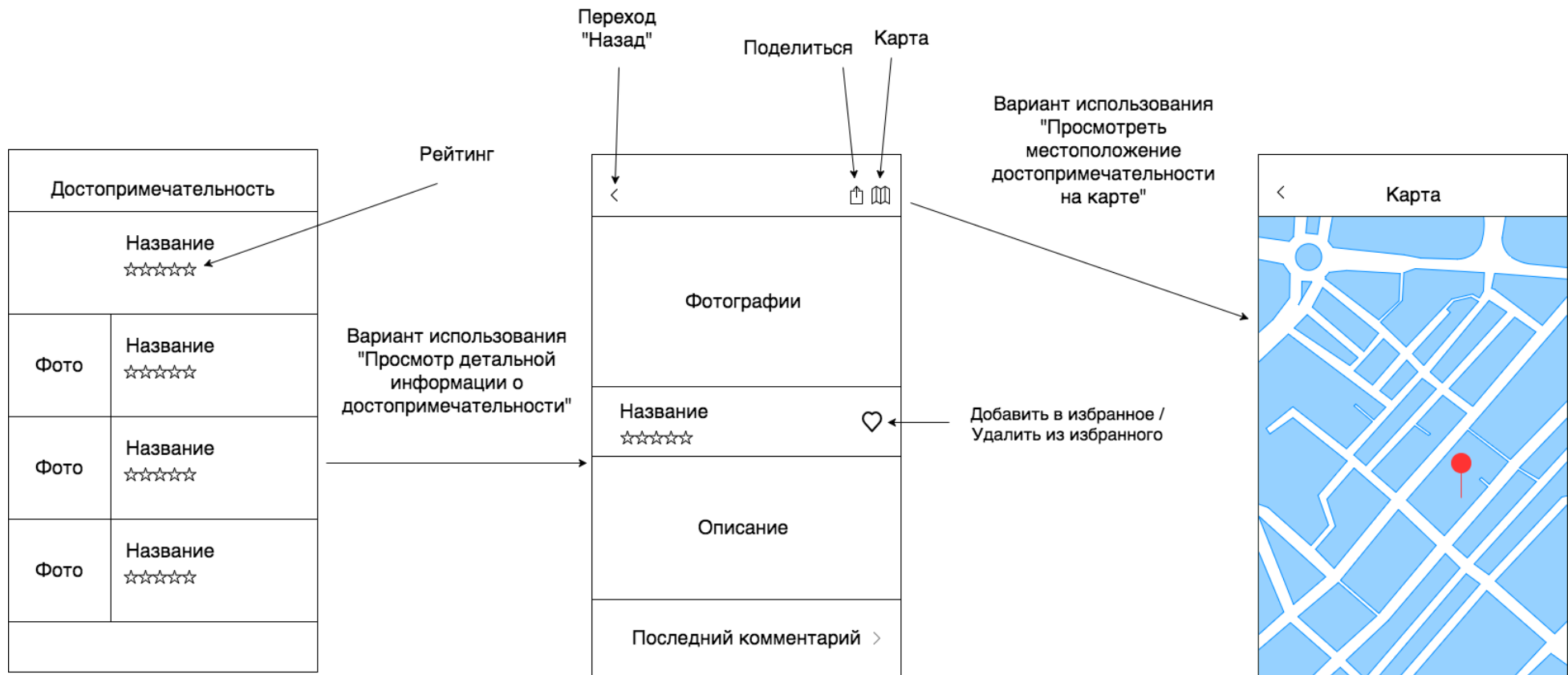


Рисунок 4 – Просмотр достопримечательностей

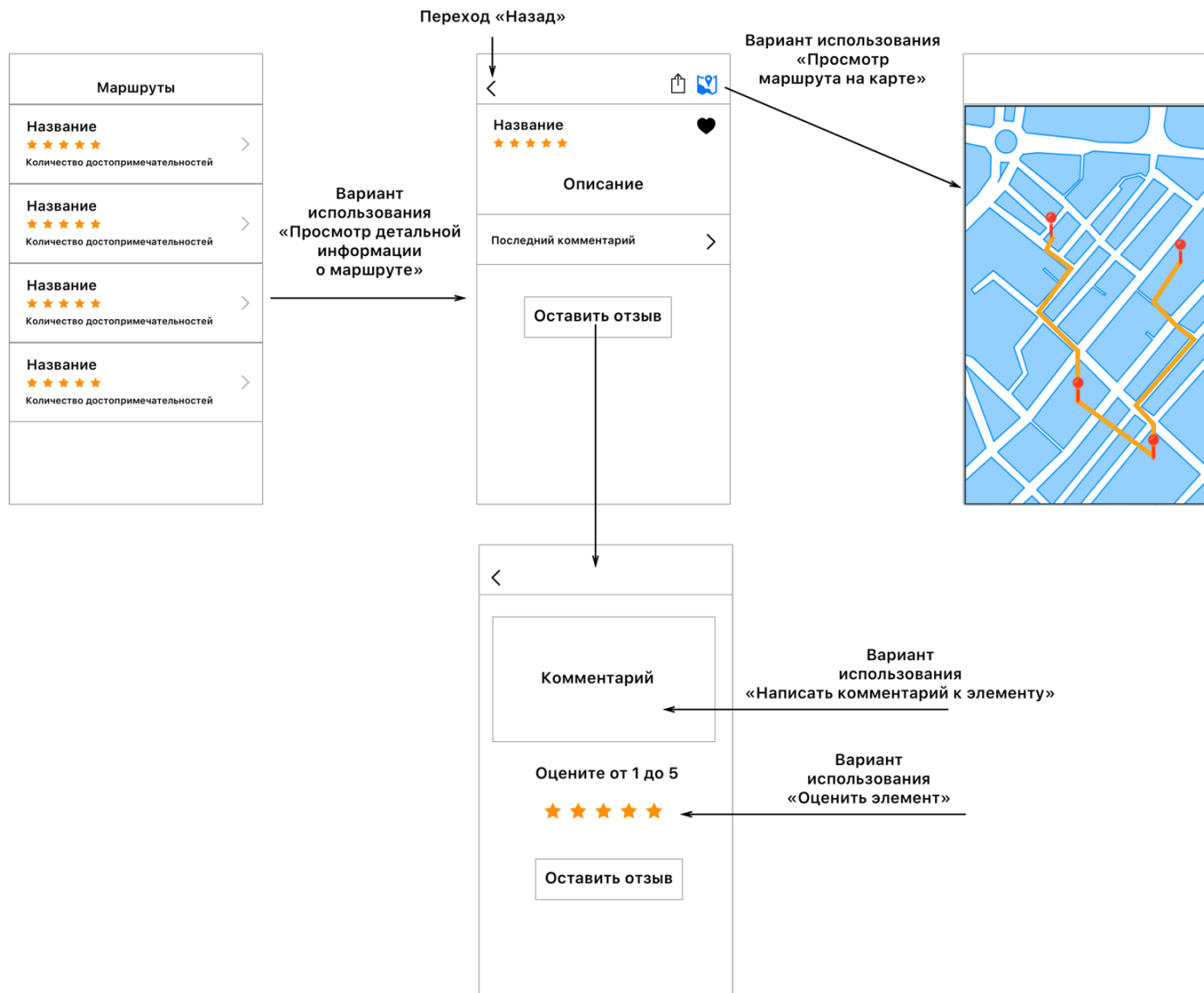


Рисунок 5 – Просмотр маршрутов

Нефункциональные требования – требования, которые задают с соблюдением каких условий система должна что-то делать. Их иногда называют ограничениями или требованиями к качеству. Они могут быть дополнительно классифицированы в зависимости от того, являются ли они требованиями к производительности, совместимости, безопасности, надежности, совместимости и т.д. [7].

На основе проведенного анализа приложений-путеводителей можно выделить следующие основные нефункциональные требования:

- Основные данные должны храниться на сервере в СУБД;
- Клиентское приложение хранит локальную копию данных загруженных с сервера;
- Безопасность передачи данных при загрузке с сервера.

## 2 Модель предметной области

Началом проектирования является создание модели предметной области. Создание модели позволяет выявить ранее неизвестные требования. В процессе проектирования был использован набор правил Domain-Driven Design (DDD) [8].

Domain-Driven Design (DDD) – это подход к разработке программного обеспечения для комплексных потребностей, связывающий реализацию с развивающейся моделью основных бизнес-концепций [9]

Для представления той информации, которую хранит разрабатываемая БД используется концептуальная модель предметной области. Стандартом при моделировании, является семантическая модель Entity-Relationship, «сущность – связь» [10].

Основными понятиями ER-модели являются сущность, связь и атрибут.

Сущность – это реальный или представляемый объект, информация о котором должна сохраняться и быть доступной

Связь – это графически изображаемая ассоциация, устанавливаемая между двумя типами сущностей. Как и сущность, связь – это типовое понятие, все экземпляры обоих связываемых типов сущностей подчиняются устанавливаемым правилам связывания.

Атрибутом сущности является любая деталь, которая служит для уточнения, идентификации, классификации, числовой характеристики или выражения состояния сущности.

На рисунке 6 представлена концептуальная модель предметной области. Для проектирования концептуальной модели использовалась нотация IDEF1X.

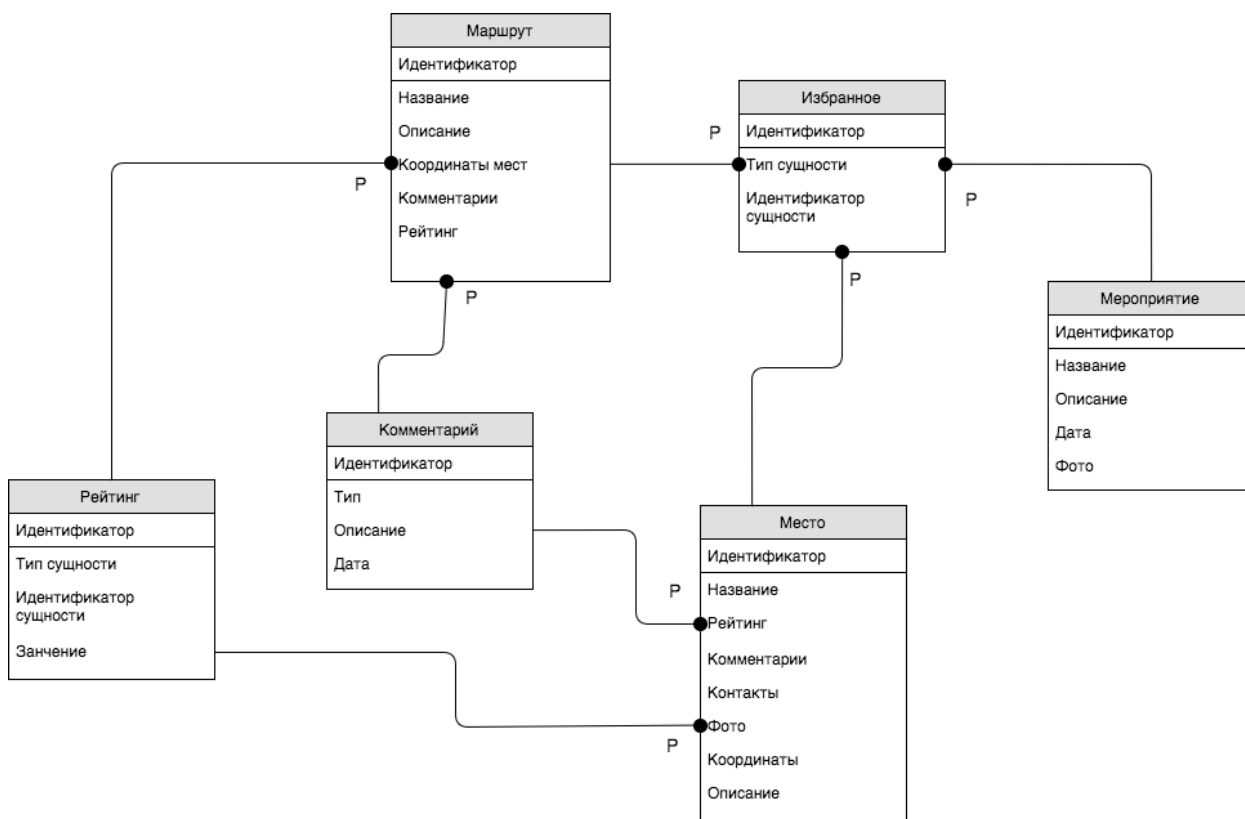


Рисунок 6 – Концептуальная модель предметной области

Сущности модели предметной области:

- Мероприятие – содержит информацию о актуальных мероприятиях города и области;
- Место – содержит информацию о достопримечательностях города.
- Маршрут – содержит сведения о маршрутах построенных на основе достопримечательности;
- Избранное – включает в себя информацию о мероприятиях, достопримечательностях или маршрутах;
- Комментарий – содержит информацию о комментариях пользователей;
- Рейтинг – содержит информацию о рейтинге мест и маршрутов;

Для данных сущностей выбраны следующие связи:

- связь один к многим между сущностями Мероприятие и Избранное;
- связь один ко многим между сущностями Место и Избранное;
- связь один ко многим между сущностями Маршрут и Избранное;
- связь один ко многим между сущностями Комментарий и Место;
- связь один ко многим между сущностями Комментарий и Маршрут;
- связь один ко многим между сущностями Рейтинг и Место;
- связь один ко многим между сущностями Рейтинг и Маршрут;

### **3 Компонентная архитектура**

Компонентная архитектура обеспечивается разделением программного приложения на компоненты. Компонентная архитектура позволяет разбить дизайн на отдельные функциональные компоненты, предоставляющие четко определенные интерфейсы, содержащие методы, события и свойства.

Возможны два подхода к реализации приложения-путеводителя:

- Однокомпонентная архитектура;
- Многокомпонентная архитектура.

#### **Однокомпонентная архитектура**

В однокомпонентной архитектуре приложение устанавливается из магазина приложений с уже загруженным контентом. Вся логика прописана в одном приложении, индивидуально для каждой платформы.

Достоинства данной архитектуры:

- Меньшая трудоемкость разработки;
- Независимость соединения с сетью интернет;
- Безопасность передачи данных.

Недостатки данной архитектуры:

- Длительное время ожидания обновления контента;
- Не позволяет использовать социальное взаимодействие между пользователями;
- Трудности при переходе на новую платформу.

#### **Многокомпонентная архитектура**

Многокомпонентная архитектура или клиент-сервер представляет собой приложение с графическим интерфейсом, которое обменивается данными с сервером базы данных, на котором хранится постоянно обновляемый контент и основная часть бизнес-логики. Обобщенно, данный стиль описывает отношения между клиентом и сервером (или серверами), где со стороны клиента осуществляется последовательность действий, запрос – ожидание ответа – обработка при получении. Сервер в свою очередь, авторизует

пользователя – проводит обработку, необходимую для получения результата – передает результат.

Достоинства данной архитектуры:

- Постоянное обновление контента;
- Социальное взаимодействие между пользователями;
- Хранение данных для разных версий приложения;
- Поддержка несколько типов устройств и платформ.

Недостатки данной архитектуры:

- Создание и поддержка серверного компонента;
- Нужно организовывать безопасную передачу данных между клиентом и сервером.

Сравнения достоинства и недостатки рассмотренных архитектур, был сделан вывод, что наиболее подходящей архитектурой для реализации приложения-путеводитель является многокомпонентная архитектура.

В качестве архитектурного решения используется частный случай многокомпонентной архитектуры – трёхуровневая архитектура.

Данная архитектура позволяет разумно распределять модули обработки данных, которые вынесены на один или несколько отдельных серверов. Данные модули реализуют функции клиента для серверов баз данных и сервера для пользовательских интерфейсов. Одним из явных преимуществ такой архитектуры является то, что сервера могут взаимодействовать друг с другом, что позволит более подробного разделения системы на функциональные блоки с определёнными ролями. Графическое представление данной архитектуры показано на рисунке 7.



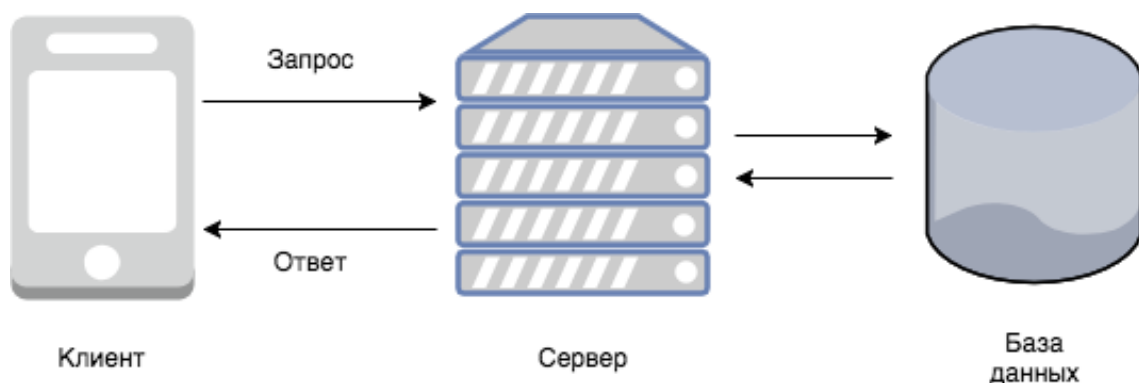


Рисунок 7 – Трёхуровневая компонентная архитектура

Трёхуровневая компонентная архитектура имеет три основных уровня:

- Уровень клиента – представляет собой графический интерфейс пользователя.
- Уровень сервера приложений – является связующим слоем между уровнями клиента и базой данных.
- Уровень базы данных – осуществляет хранение данных, реализуется средствами СУБД.

### **Уровень клиента**

Предоставляет функции ввода и отображение данных. Данный уровень содержит простую бизнес-логику и хранит состояние приложения. В реализуемой системе на этом уровне будет располагаться:

- Бизнес-логика для составления и отправки запросов на сервер, получение и обработки ответов от сервера;
- Бизнес-логика для построения маршрутов и определения геопозиции пользователя;
- Алгоритмы хранения полученных данных;
- Взаимодействие элементов графического интерфейса.

### **Уровень сервера приложений**

Содержится большая часть бизнес-логики. В зависимости от программной системы на данном уровне реализуются рабочие процессы так как координирует перемещение и обработку данных между уровнями клиента

и базы данных, обрабатывает команды, выполняет логические решения и вычисления, выполняет расчеты.

Данный уровень в реализуемой системе будет содержать:

- Бизнес-логику для приема и обработки запросов от клиента;
- Бизнес-логику для взаимодействия с базой данных, составление запросов и обработку результатов выполнения запросов;
- Бизнес-логику для формирования и отправки данных на сторону клиента;

### Уровень базы данных

Данный уровень взаимодействует только с уровнем сервера приложений. Уровень баз данных обеспечивает хранение и обработку данные. Данные отправляются на уровень сервера приложений для обработки и отправки на уровень клиента.

Для представления взаимосвязи между логическими или физическими модулями, из которых состоит система создадим диаграмму компонентов и опишем ее [11,12]. UML диаграмма компонентов представлена на рисунке 8.

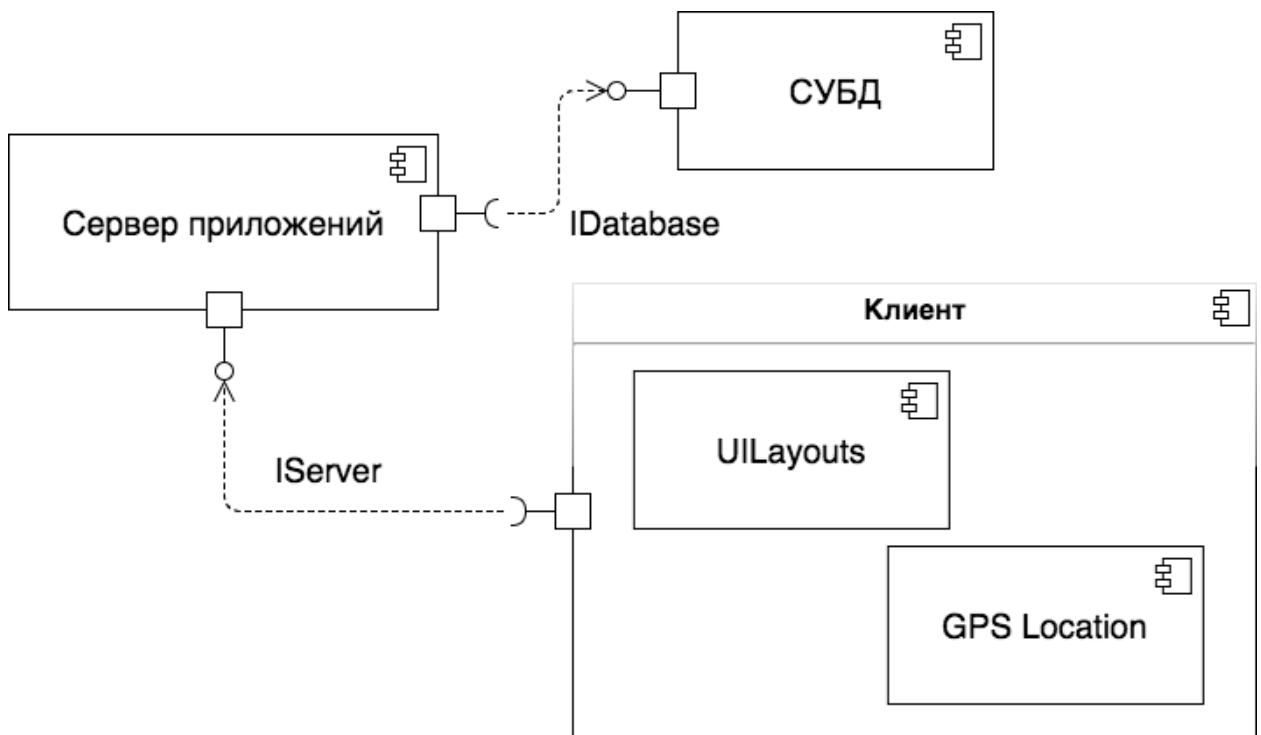


Рисунок 8 – UML диаграмма компонентов

Компонент Сервер приложений – реализует бизнес-логику для приема и обработки запросов клиента, взаимодействия с СУБД, формирования и отправки данных клиенту. Предоставляет интерфейс IServer. Потребляет интерфейс IDatabase.

Компонент СУБД – реализует доступ к хранилище данных. Предоставляет интерфейс IDatabase.

Компонент Клиент – реализуется в виде мобильного приложения. Потребляет интерфейс IServer. Содержит следующие компонентные части:

- UILayouts – компонентная часть, содержащая элементы пользовательского интерфейса. Реализуется в виде библиотеки классов.
- GPS Location – компонентная часть, служит для отображения достопримечательностей и маршрутов на карте. Реализуется в виде библиотеки классов.

### **3.1 Выбор технологий и инструментов реализации**

Трехуровневая архитектура, будет спроектированная с применением сервис-ориентированного архитектурного стиля. Сервис-ориентированная архитектура позволяет создавать распределённое программное обеспечение, состоящее из набора независимых сервисов [13,14].

Данная архитектура обладает рядом преимуществ и одно из них — это способность быстро адаптироваться к меняющимся требованиям и задачам информационного мира.

На данный момент наиболее востребованы SOAP веб сервисы и REST сервисы. Основные различия между подходами SOAP и REST [15]:

- Пакеты запросов и ответов в REST имеют намного меньшие размеры, чем соответствующие им пакеты SOAP.
- SOAP рассматривает уровень передачи данных HTTP, как пассивного наблюдателя, а REST напротив, как активного

участника взаимодействия, используя существующие методы HTTP, такие как GET, POST, PUT, DELETE.

- Модель SOAP поддерживает определенную степень интроспекции, позволяя разработчикам сервиса описывать его API в файле формата WSDL (Web Service Description Language), что является одной из не самых простых задач. В свою очередь модель REST избегает сложностей WSDL и использует более понятный интерфейс, основанный на стандартных методах HTTP.
- REST ограничен операциями CRUD соответствующими методам HTTP. SOAP может содержать практически неограниченное количество методов.

Сравним достоинства и недостатки двух подходов к реализации веб-сервисов, было выявлено, что REST веб-сервисы являются наиболее понятными и простыми в реализации, имеют наиболее меньшие размеры пакетов.

Рассмотрим несколько вариантов, которые позволяют реализовать REST API:

- Типовое решение для C# - *Microsoft MVC Web Application* с модулем Web API [16]. Главное достоинство данного решения — это скорость работы веб-сервисов, а главным недостатком является сложность развертывания из-за наличия большого количества дополнительных компонентов.
- Альтернативой заменой ASP может является использование PHP. PHP является достаточно гибким и мощным. Основные преимущества:
  - Является свободным программным обеспечением, распространяемым под особой лицензией – PHP License.
  - Развитая поддержка баз данных;
  - Огромное количество библиотек и расширений языка;

В итоге для реализации REST API был выбран язык программирования PHP. Т.к. для реализации веб-сервисов на языке PHP не требуется установки дополнительных библиотек, настройки большого количества компонентов или программной среды. Для реализации достаточно установить интерпретатор языка. Как альтернативу языку PHP был рассмотрен язык программирования Python [17].

Установка языка программирования Python осуществляется без подключения дополнительных библиотек, которые необходимы для реализации веб-сервисов. Без дополнительных библиотек Python ориентирован на научные вычисления [18,19].

### **Выбор СУБД**

Необходимые и достаточные требования при выборе СУБД для реализации поставленных целей является:

- Наличие подробной документации;
- Наличие программного интерфейса для доступа к базе данных [20];
- Поддержка различных платформ;
- Поддержка компаниями, которые предоставляют ресурс для размещения информации на сервере (Хостинг);
- Поддержка работы с различными ОС;
- Удобство при развертывании.

Под данные требования подходит свободная, разрабатываемая и поддерживаемая корпорацией Oracle реляционная СУБД MySQL.

Всю серверную часть можно реализовать с помощью готового решения MAMP [21]. MAMP – это акроним, назван по первым буквам входящих в него компонентов:

## Выбор средств реализации уровня клиента

На рисунке 9 представлена диаграмма долей рынка мобильных операционных систем в Российской Федерации на 2017 год [22].

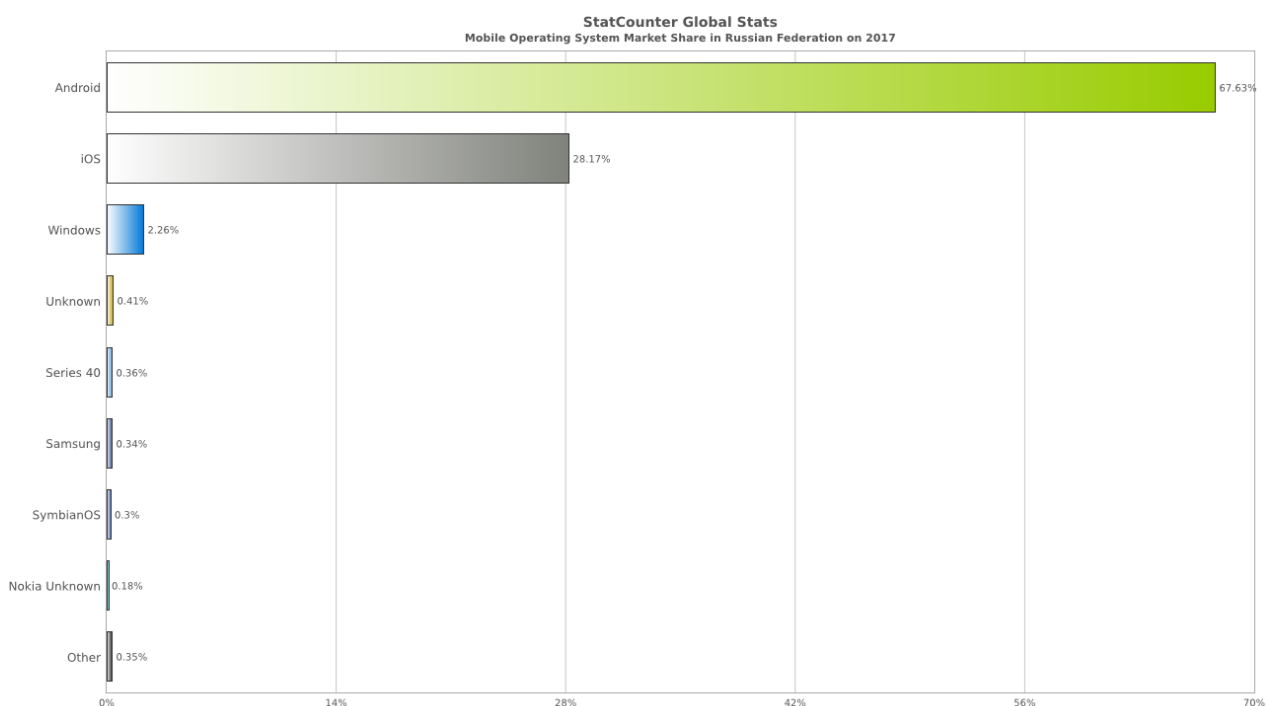


Рисунок 9 – доля рынка мобильных операционных систем в Российской Федерации на 2017 год

Из диаграммы видно, что явными лидерами являются Android и iOS. Проведем сравнительный анализ эти двух операционных систем [23].

IOS более безопасная платформа, чем Android. По данным Pulse Secure's Android-устройства являются основными носителями вредоносного ПО и содержат 97 % всех мобильных вредоносных программ [24].

IOS система имеет более лучшую поддержку конфиденциальности пользователя.

Android является более настраиваемой операционной системой в отличии от жестко контролируемой Apple iOS. В частности, это относится к настройке пользовательского интерфейса.

В итоге выбор сделан в пользу операционной системы Apple iOS, в виду более высоких характеристик безопасности и более стабильная работа симулятора устройств. Она позволит создать безопасное, простое, удобное и приятное в использовании приложение.

Для создания приложения под iOS была выбрана интегрированная среда разработки Xcode, разработанная корпорацией Apple, которая включает в себя значительную и необходимую часть документации разработчика от Apple. Язык программирования – Swift 3.1. Swift создан компанией Apple и является более быстрым, легким для чтения и более устойчив к ошибкам программиста, чем предшествовавший ему Objective-C.

## 4 Реализация программной системы

### 4.1 Проектирование и реализация уровня сервера приложений

Сервер приложений представляет собой набор REST веб-сервисов, реализующих операции чтения, создания и удаления данных, которые соответствуют методам протокола HTTP [25]:

- GET – получение списка мероприятий, мест, маршрутов, избранного, комментариев;
- POST – добавление элемента в избранное, создание комментария, установка рейтинга достопримечательности;
- DELETE – удаление элемента из избранного.

Обязательны заголовки запроса к серверу:

- Accept: application/json;
- Content-Type: application/json;
- Identifier: ИД\_УСТРОЙСТВА;
- Authorization: sha1(ИД\_УСТРОЙСТВА + КЛЮЧ).

Сервисы отображаются в базе данных travel MySQL. На рисунке 10 представлена ER-диаграмма базы данных travel.

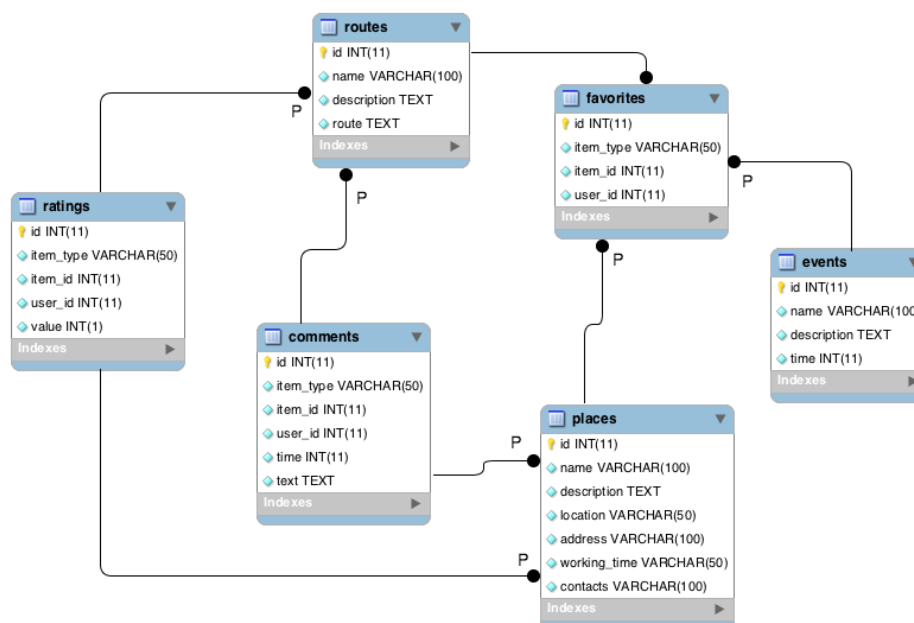


Рисунок 10 – ER-диаграмма базы данных travel



## Описание веб-сервисов

В файле `events.php` реализованы веб-сервисы для работы с мероприятиями:

- GET `/v1/events` – получение списка всех мероприятий, где `v1` – версия API);
- GET `/v1/events/ID_EVENT` – получение информации об одном мероприятии.

В файлах `places.php`, `ratings.php` реализованы веб-сервисы для работы с достопримечательностями:

- GET `/v1/places` – получение списка всех мест
- GET `/v1/places/ID_PLACE` – получение информации о одном месте;
- POST `/v1/places/ID_PLACE/ratings/5` – оценка места.

В файлах `routes.php`, `ratings.php` реализованы веб-сервисы для работы с маршрутами:

- GET `/v1/routes` – получение списка всех маршрутов;
- GET `/v1/routes/ID_ROUTE` – получение конкретного маршрута;
- POST `/v1/routes/ID_ROUTE/ratings/5` – оценка маршрута.

В файле `favorites.php` реализованы веб-сервисы для работы с избранным:

- GET `/v1/favorites` – получение списка избранного;
- POST `/v1/MODULE/ID_MODULE/favorites` – добавление в избранное. В избранное можно добавить для модулей: `events / places / routes`;
- DELETE `/v1/MODULE/ID_MODULE/favorites` – удаление из избранного. Из избранного можно удалить только для модулей `events, places, routes`;

В файле `comments.php` реализованы веб-сервисы для работы с комментариями. Комментарий можно оставить для мест и маршрутов:

- GET /v1/MODULE/ID\_MUDULE/comments – получение всех комментариев для выбранного места или маршрута;
- GET /v1/MODULE/ID\_MUDULE/comments/ID\_COMMENT – получение конкретного комментария для выбранного места или маршрута;
- POST /v1/MODULE/ID\_MUDULE/comments/Комментарий – добавление комментария для выбранного места или маршрута.

Исходный код реализации веб-сервисов для работы с избранным приведен в приложении А.

### **Защита передаваемых данных**

При передачи обновленного контента от сервера клиенту по протоколам HTTP данные могут быть перехвачены сторонними приложениями, используя текст запроса. Для устранения такой возможности перехвата данных нужно обеспечить защиту передаваемых данных.

Защита передачи данных осуществляется с помощью HTTPS, а также идентификации и авторизации клиента на сервере. Для это при выполнении запроса передаются соответствующие заголовки:

- Identifier: ИД\_УСТРОЙСТВА;
- Authorization: sha1(ИД\_УСТРОЙСТВА + КЛЮЧ).

ИД\_УСТРОЙСТВА – это UUID (Universally Unique Identifier) [26] устройства с которого выполняется запрос к серверу. Пример UUID – 98F1B145-56F9-49FE-853B-40603CFAA7EF.

КЛЮЧ – специальный ключ, который храниться на сервере и клиенте. Пример ключа – okdJ6GDF22Gvu2g

Заголовок Identifier содержит ИД\_УСТРОЙСТВА.

Заголовок Authorization содержит хешированную строку ИД\_УСТРОЙСТВА + КЛЮЧ с помощью криптографического алгоритма SHA1 [27]. Пример хешированной строки – 4529cdfa47716fc88d1a56f7e12cd91729066dc6

Сервер принимает значения заголовком, выполняет хеширование ИД\_УСТРОЙСТВА и ключа, который храниться на сервере. Далее сравнивается хеш-сумма [28] полученная от клиента и полученная на сервере.

На рисунке 11 с помощью диаграммы деятельности проиллюстрирована идентификация клиента на сервере.

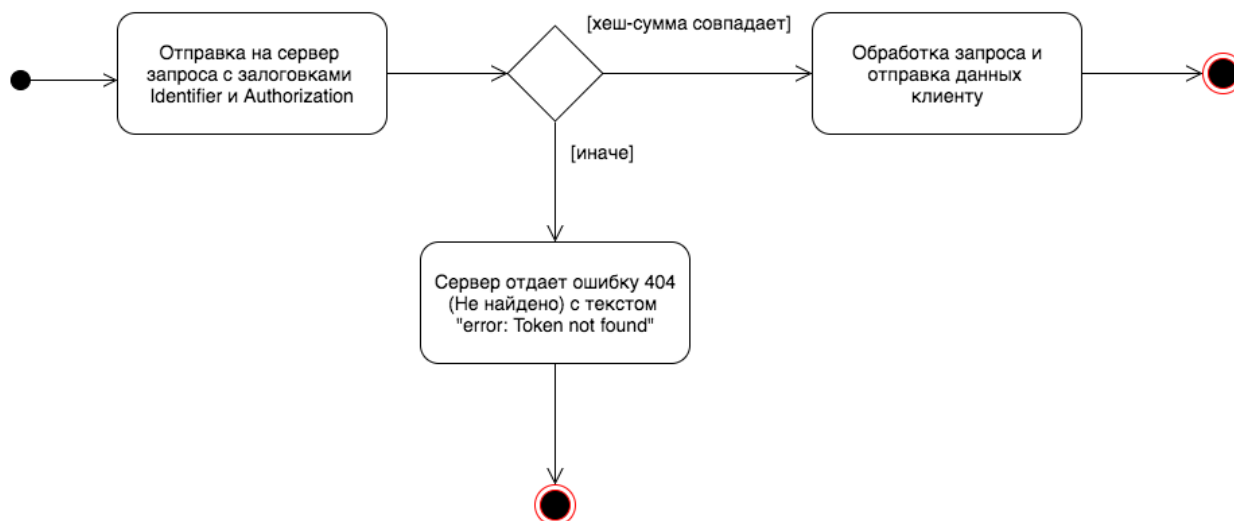


Рисунок 11 – UML диаграмма деятельности идентификации клиента на сервере

## 4.2 Проектирование и реализация уровня клиента

Приложение содержит следующие основные модули:

- UILayouts – интерфейс пользователя;
- Datasets – загрузка и обработка полученных данных с сервера;
- GPS Location – работа с картой и геолокацией.

На рисунке 12 представлена UML диаграмма пакетов модулей уровня клиента.

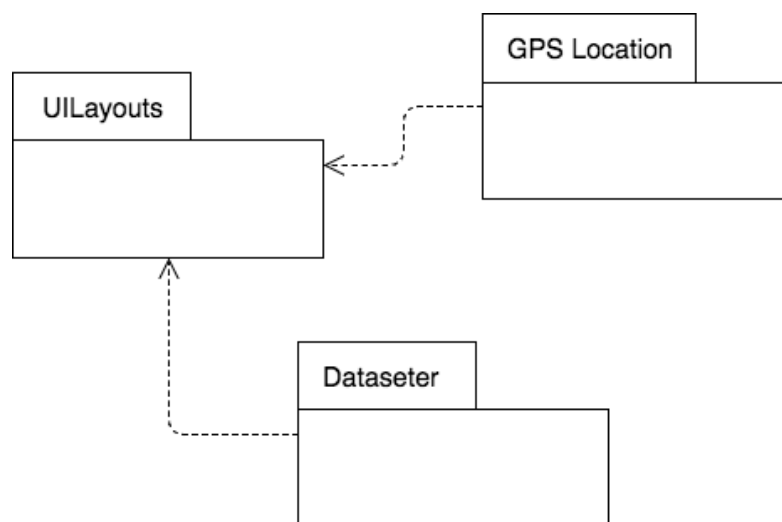


Рисунок 12 – UML диаграмма пакетов модулей уровня клиента

### Реализация интерфейса пользователь модуль UILayouts

Данный модуль проектируется и реализуется по шаблону MVVM (Model – View – ViewModel). Шаблон MVVM позволяет разделить модель приложение и ее представление для изменения их отдельно друг от друга. С помощью данного шаблона связываются элементы представления со свойствами и событиями представления-модели. Соответственно каждый слой MVVM не знает о существовании другого [29].

Шаблон MVVM требует для своей реализации использование инфраструктуры UIKit. Такие элементы View как UIView или UIViewController входят в состав UIKit.

UIKit отвечает за создание и управление пользовательского интерфейса в iOS. UIKit.framework обеспечивает необходимую инфраструктуру необходимую для создания и управления приложениями iOS. UIKit является менеджером для создания представление, управления событиями на действия пользователя, управления основным циклом запуска и взаимодействия с системой [30].

Для реализации приложения был выбран шаблон проекта Xcode – Single View Application. Single View Application – шаблон проекта На рисунке 13 приведен пример создания Single View Application проекта.

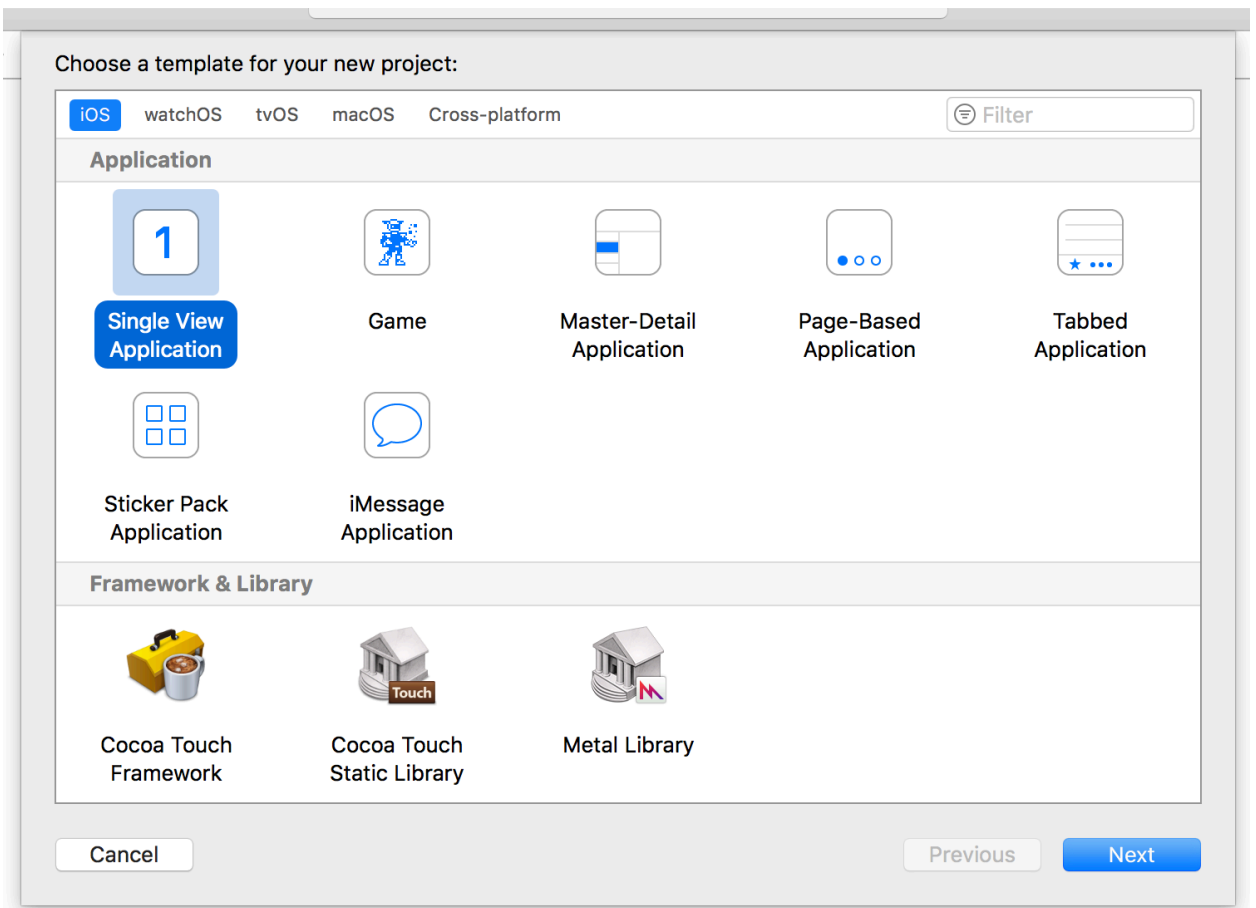


Рисунок 13 – Создание Single View Application проекта

Для перехода между экранами используется `UITabBarController`. `UITabBarController` – это контроллер представлений, который управляет интерфейсом панели вкладок. Вкладки располагаются в нижней части окна, при выборе вкладок показывается соответствующий экран [31]. Панель вкладок с показанными экранами мероприятий и мест показана на рисунке 14.

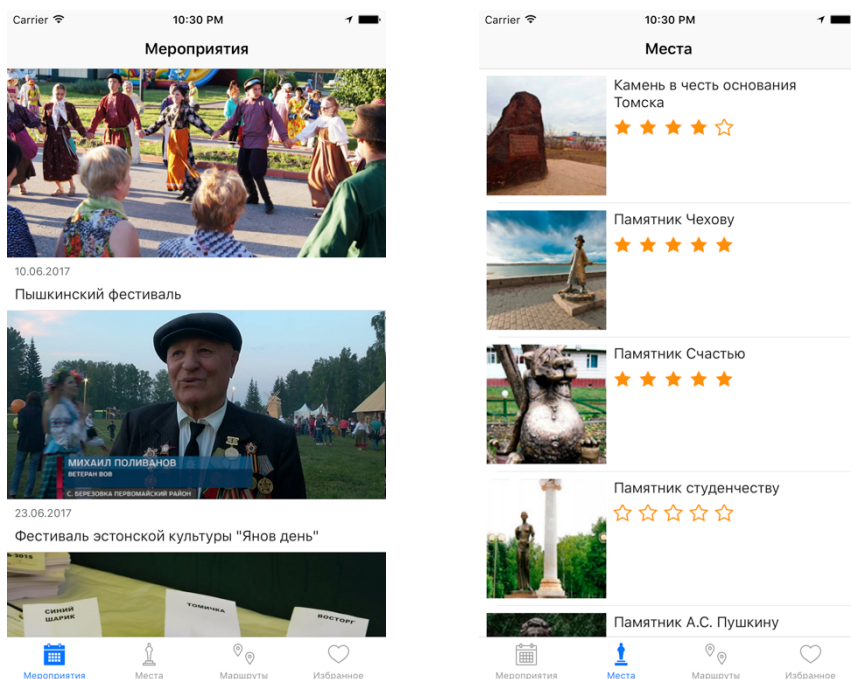


Рисунок 14 – снизу – панель вкладок снизу, слева – главный экран

Экраны со списками мероприятий, достопримечательностей, маршрутов и избранного реализуются с помощью UITableViewController и представляют собой таблицы с ячейками. Макет ячеек таблицы мероприятий, мест и маршрутов представлен на рисунке 15.



Рисунок 15 – Макет ячеек

Добавление комментария и оценка места или маршрута реализуется по схеме, представленной на рисунке 16.

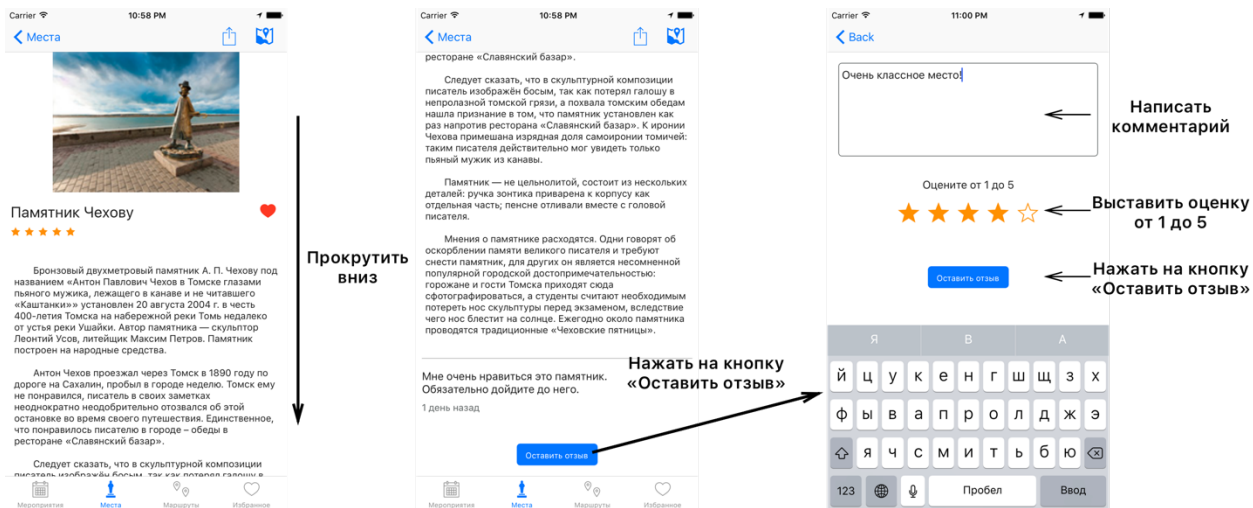


Рисунок 16 – Схема реализации добавления комментария

Просмотр списка всех комментариев реализуется с помощью нажатия на ячейку с комментарием на экране выбранной достопримечательности или маршрута, схема перехода к списку комментариев представлена на рисунке 17.

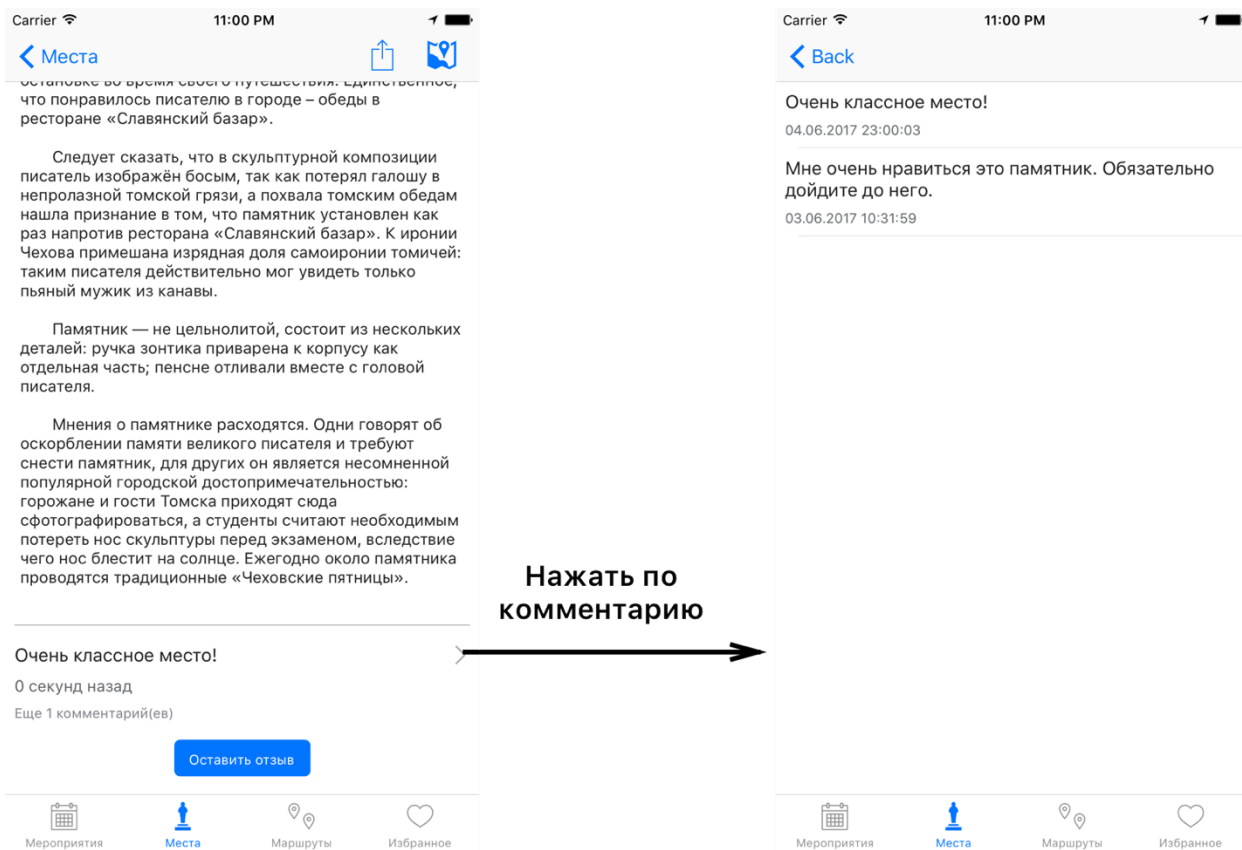


Рисунок 17 – Список всех комментариев

Просмотр достопримечательности или маршрута реализуется с помощью элемента представления MKMapView. Пример просмотра

местоположения достопримечательности и маршрута на карте представлено на рисунке 18.

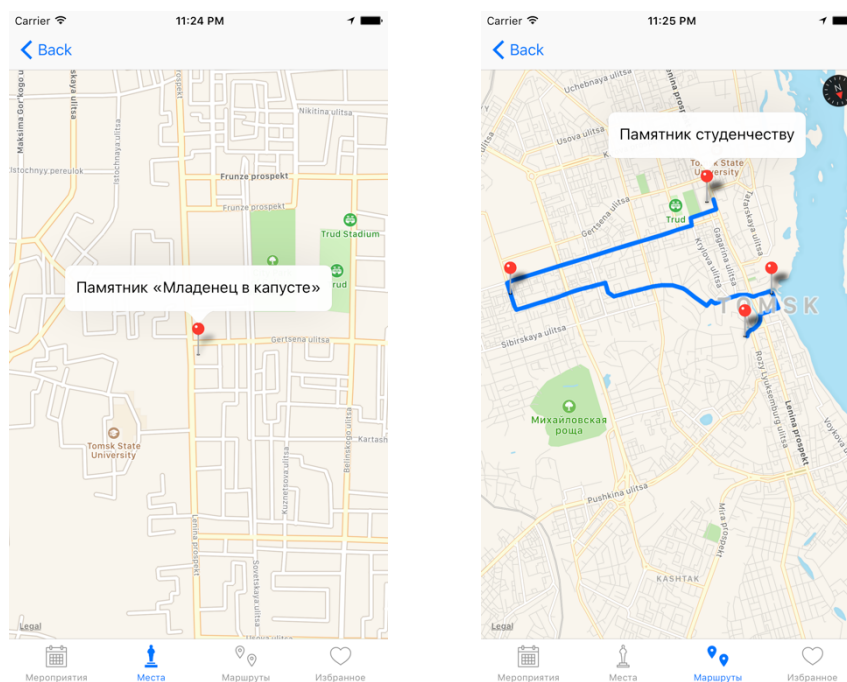


Рисунок 18 – Отображение достопримечательности слева, маршрута справа

### **Реализация загрузки и обработки полученных данных с сервера модуль Dataseter**

Для загрузки и скачивания контента через HTTP Apple предоставляет URLSession, API которая является полным набором сетевых методов [32]. Контент отправляется и получается в JSON формате, для преобразования JSON в объекты операционной системы и наоборот Apple предоставляет класс NSDataSerialization [33].

Для загрузки и обработки контента используется следующий набор инфраструктур:

- Mya – является абстракцией для класса URLSession [34].
- SwiftyJSON – упрощает работу с данными JSON в Swift [35].
- Kingfisher – библиотека для загрузки и кэширования изображения из интернета [36].

Для подключения инфраструктур в приложение использовался менеджер зависимостей Cocoapods



## Реализация взаимодействия с инфраструктурой Moya

Создание запросов к серверу осуществляется через класс `MoyaProvider`, которые реализует протокол `TargetType`. Протокол `TargetType` – служит для определения спецификаций, необходимых для «`MoyaProvider`».

Для указания необходимых спецификаций был создано перечисление (enum) `Guide`, в котором перечислены все варианты запросов с указанием параметров. Перечисление `Guide` определяет следующие спецификации:

- `baseUrl` – базовый URL;
- `path` – добавляется к `baseUrl` для создания полного URL;
- `method` – метод HTTP;
- `parameters` – параметры запроса, которые будут закодированы;
- `parameterEncoding` – указывается тип кодировки параметров запроса;
- `sampleData` – заглушка с данными, используется при тестировании;
- `task` – тип HTTP операции.

Выполнение запроса на получения все комментариев с сервера представлено на рисунке 19 в виде диаграммы деятельности.

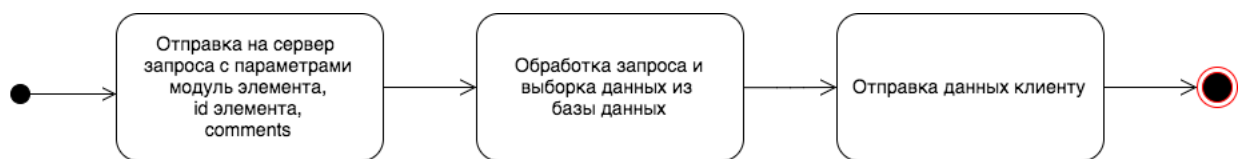


Рисунок 19 – Выполнение запроса на получения всех комментариев

Фрагмент исходного кода запроса к серверу для получения все комментариев для выбранного места:

```
/*
```

```
guideProvider - объект класса MoyaProvider
```

```
.comments - перечисление для формирования запроса получения все комментариев,
```

```
(module: .places, id: place.id) - параметры запроса,
```

```
module - модуль (.places - место),
```

```
id - идентификатор места в базе данных (place.id - идентификатор конкретного
```

места)

```
*/
```

```
guideProvider.request(.comments(module: .places, id: place.id)) { (result) in
```

```
// result - результат выполнения запроса
```

```
switch result {
```

```
case let .success(response):
```

```
// Успешное выполнение запроса
```

```
// Преобразовывает полученные данные с объектов Comment
```

```
let comments = ParseJSON.dataToArray(module: .comments, data: response.data) as?  
s? [Comment] ?? []
```

```
case let .failure(error):
```

```
// Результат выполнен с ошибкой
```

```
// Выводим ошибку в консоль
```

```
Logger.log(error.errorDescription ?? "")
```

```
}
```

```
}
```

## Использование инфраструктур SwiftKeychainWrapper и CryptoSwift

Для указания заголовков запроса класс `MozaProvider` содержит свойство `endpointClosure`. В заголовках запроса указывается идентификатор устройства (UUID) и хешированный набор идентификатор устройства + ключ.

Идентификатор устройства можно получить средствами системы, но на одно устройство их несколько, и при нового сеансе приложения используется новый идентификатор. Поэтому для хранения одного идентификатора используется инфраструктура `SwiftKeychainWrapper` [37], который предоставляет более простую оболочку iOS `Keychain`, чем используется в самой системе.

Получения UUID и сохранение его представлено на рисунке 20 в виде диаграммы деятельности.

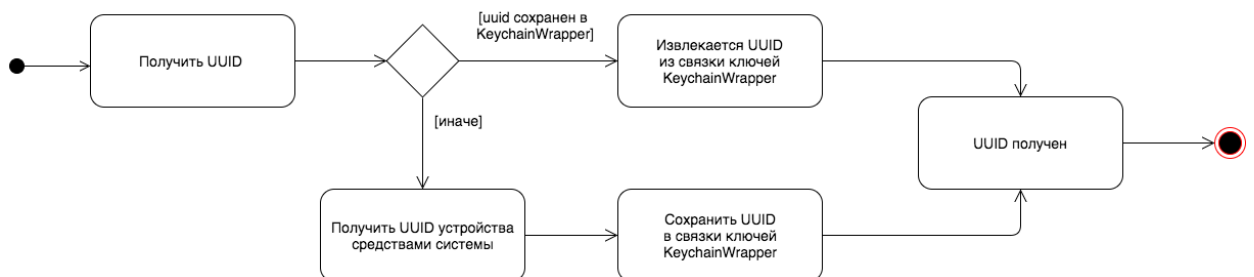


Рисунок 20 – Получения UUID и сохранение его

Фрагмент исходного кода получения UUID и сохранение его:

```
if let uuidString = UIDevice.current.identifierForVendor?.uuidString {
    let saveSuccessful: Bool = KeychainWrapper.standard.set(uuidString, forKey: "uuid")
    print("Save was successful: \(saveSuccessful)")
}
```

Для получения хешированной пары UUID+ключ использовался инфраструктура CryptoSwift [38]. Фрагмент кода получения хешированной строки пары UUID+ключ:

```
let has = (deviceId + ServerParametersValue.sha1Key).sha1()
```

### Взаимодействия с инфраструктурой SwiftyJSON

Для преобразования JSON данных в объекты операционной системы используется класс JSON. Для реализации всех преобразований данных с сервера была создана структура ParseJSON.

Преобразования данных с использованием инфраструктуры SwiftyJSON представлено на рисунке 21 в виде диаграммы деятельности.

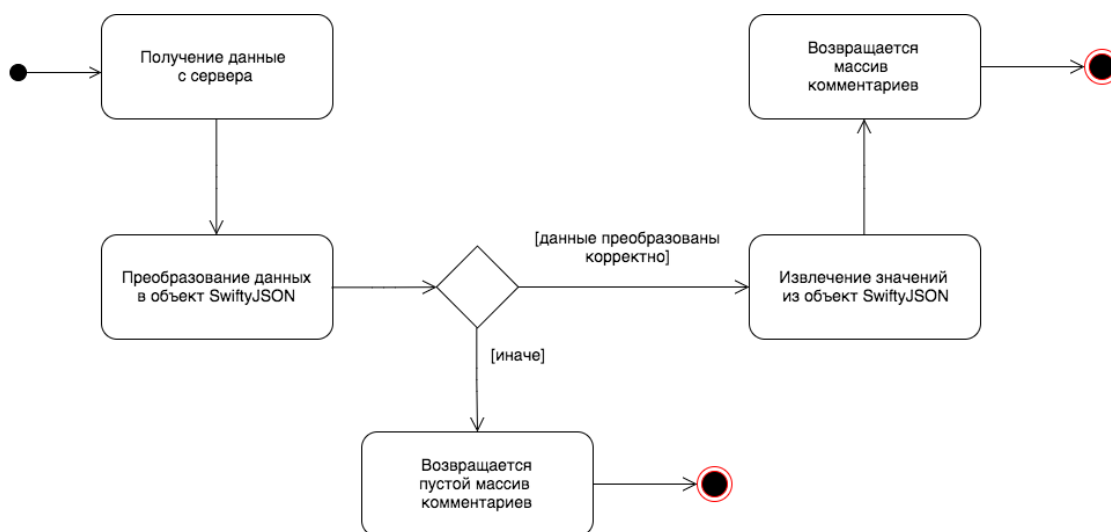


Рисунок 21 – Преобразования данных с использованием инфраструктуры SwiftyJSON

Фрагмент исходного кода для преобразования данных в комментарии:

```
// Преобразование данных с сервера к JSON
let json = JSON(data: data)
var comments: [Comment] = []
// jsonData.arrayObject - получение массива словарей
for data in (jsonData.arrayObject as? [[String: Any]]) ?? [] {
    // Извлечение данных по ключу и создание нового объекта Comment
    let c = Comment(time: (data[CommentKey.time] as? String) ?? "",
```

```

text: (data[CommentKey.text] as? String) ?? ""
    comments.append(c)
}

```

// Возвращается массив комментариев отсортированных по дате

```

return comments.sorted { $0.time > $1.time }

```

Загрузка изображений реализована с помощью класса ImageDownloader инфраструктуры Kingfisher. Изображений для одного события или места может быть несколько, с сервера в результате ответа приходим массив URL адресов на изображения. Фрагмент кода асинхронной загрузки изображений для одного события:

// Перебор массива ссылок на изображения

```

for (index, imgLink) in event.imgLink.enumerated() {

```

// Созадние объекта класса ImageDownloader с указание имени загрузчика

```

    let imageDownloader = ImageDownloader(name: "ImageDownloader.event.images.\(index)")

```

// Запуск загрузки изображения, в качестве параметра передается URL адрес и зображения

```

    imageDownloader.downloadImage(with: imgLink) { (image, error, url, data) in

```

// Результат загрузки, image - загруженное изображение, error - ошибка

// Проверяется, существует ли изображение, при ошибки значение nil

```

    if let img = image {

```

// Изображение сохраняется в массиве изображение события

```

        event.img.append(img)
    }
}
}

```

UML диаграмма классов загрузки и обработки данных с сервера показана на рисунке 22.

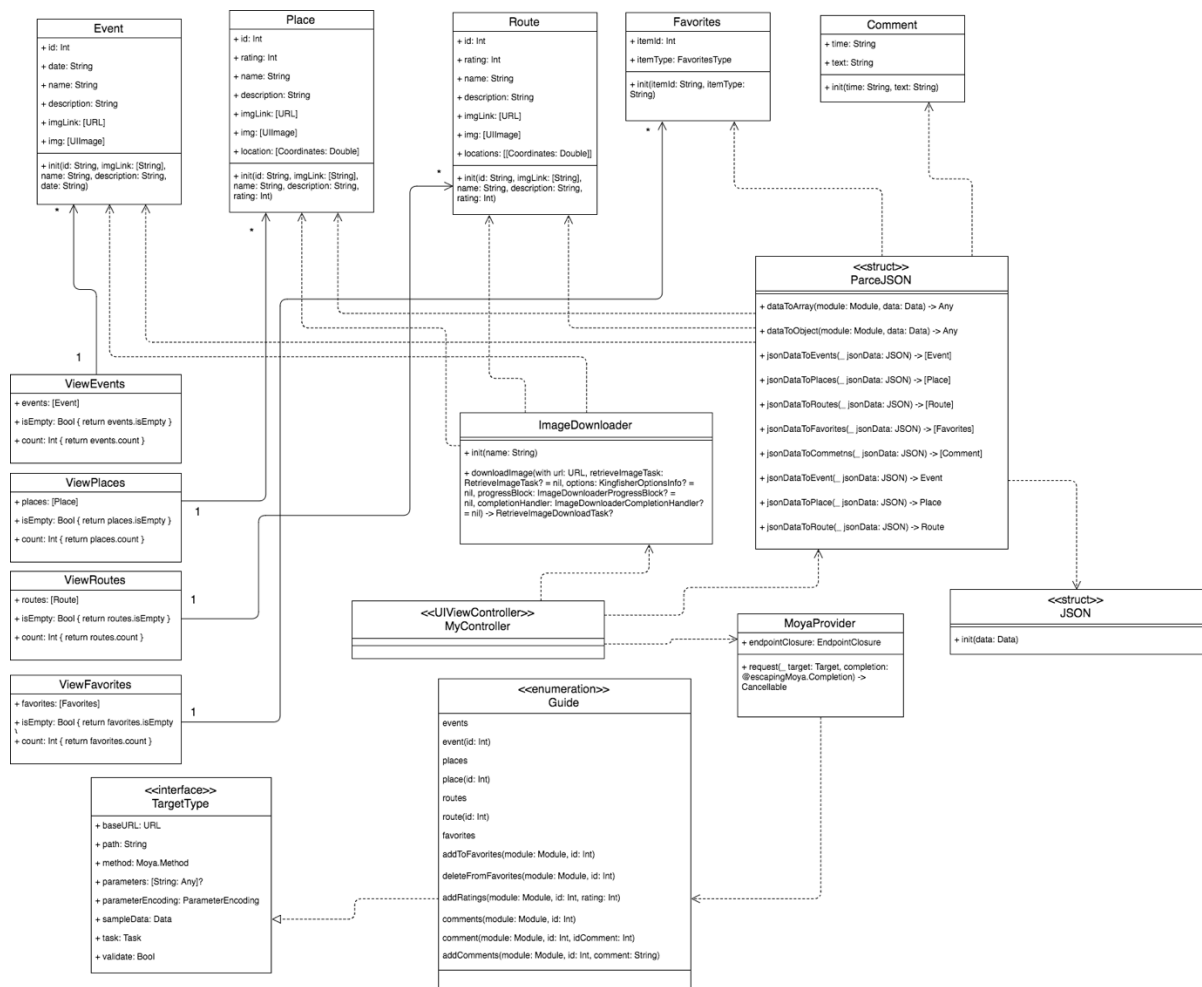


Рисунок 22 – UML диаграмма классов загрузки и обработки данных

## Работа с картой и геолокацией модуль GPS Location

Для работы с картами и геолокацией Apple предоставляет инфраструктура MapKit. MapKit обеспечивает интерфейс для работы с картой непосредственно в нашем приложения, используя собственные представления. Также MapKit позволяет создавать аннотации для указанных мест, добавлять слои для маршрутов и получение информации о месте по указанным координатам [39].

Для получения географического местоположение и ориентацию устройства Apple предоставляет инфраструктура Core Location. Core Location имеет такие возможности, как определение географического местоположения, высоты, ориентации или положения устройства относительно ближайшего iBeacon. Сбор данных осуществляется с использованием всех доступных,

встроенных аппаратных средств устройства, включая Wi-Fi, GPS, Bluetooth, магнитометр, барометр и сотовую аппаратуру [40].

Для отображения места на карте создается представление MapView. Далее весь процесс отображения места на карте и прокладывания маршрутов осуществляется через объект класса MapView и связанным с ним классов. Приведем последовательность действия для построения маршрута на карте.

На карте достопримечательности отображаются в виде аннотаций. Аннотация может содержать метку на карте, название, описание или фото элемента. Пример аннотации приведен на рисунке 23.

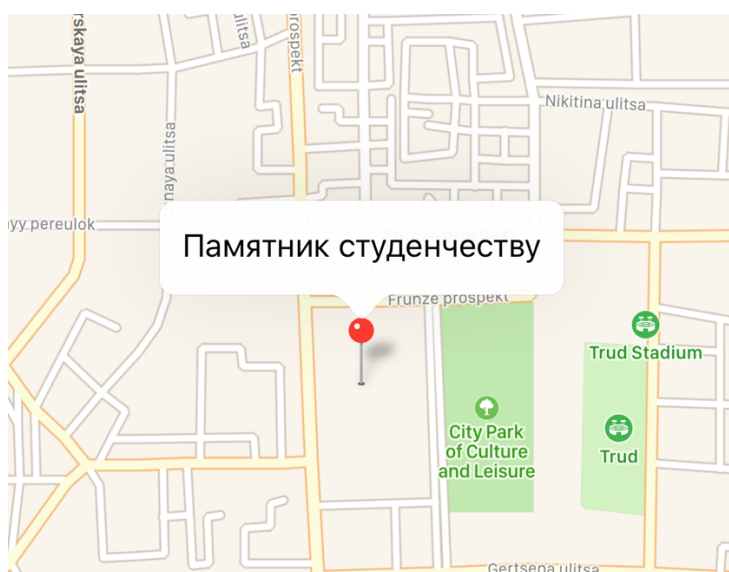


Рисунок 23 – Пример аннотации элемента на карте

На рисунке 24 приведена диаграмма деятельности, которая показывает последовательность действия для построения маршрута на карте.

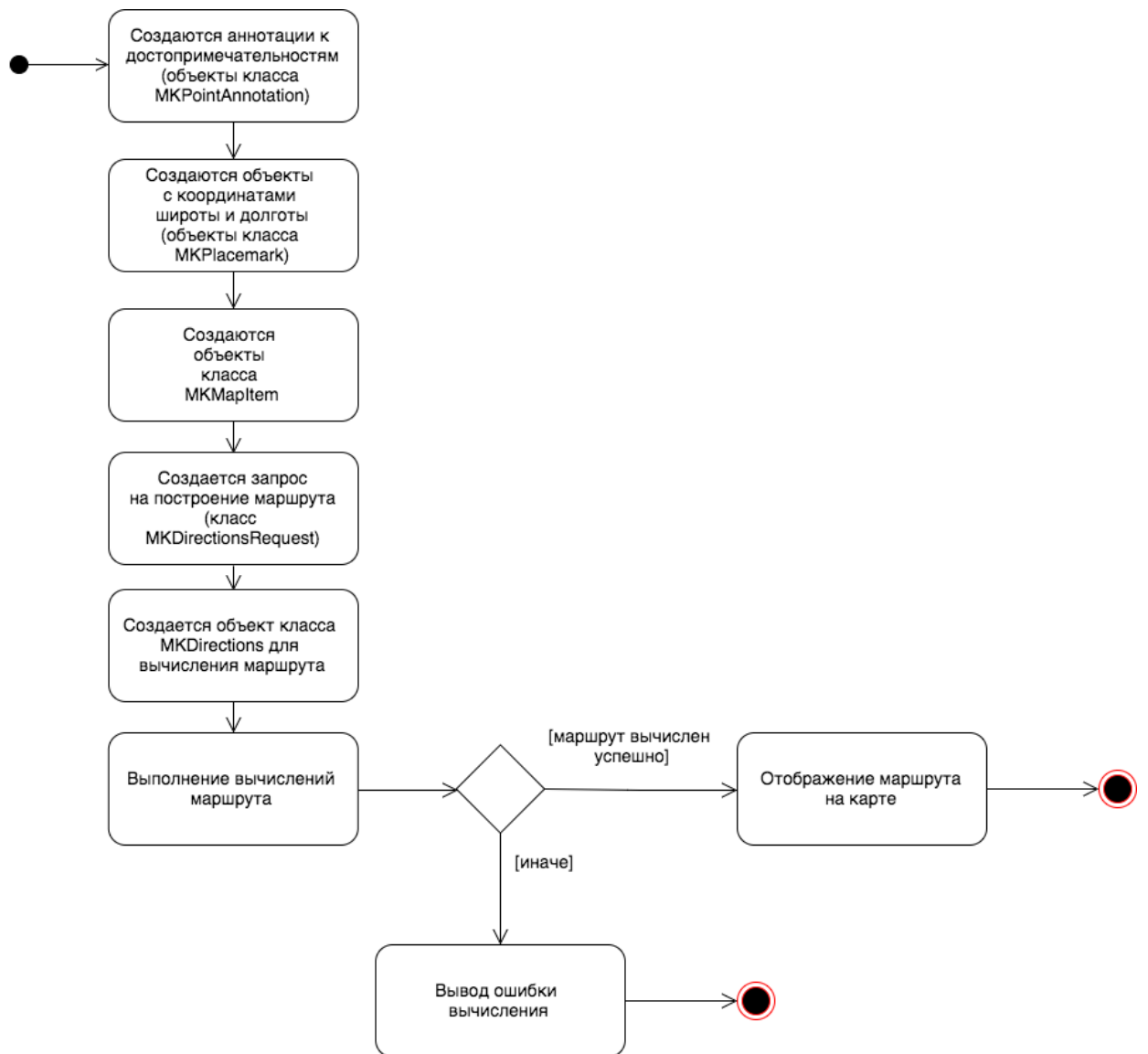


Рисунок 24 – Построения маршрута на карте

Сначала создаются аннотации к местам, которые будут отображены на карте. Это реализуется с помощью класса `MKPointAnnotation`. Данный класс определяет конкретный объект аннотации, расположенный в указанной точке.

Фрагмент кода создания аннотации:

```

let annotation = MKPointAnnotation()
// Указываются координаты места
annotation.coordinate = CLLocationCoordinate2D(latitude: latitude, longitude: longitude)
// Указывается заголовок места
annotation.title = place.name
// Добавление аннотации на карту
mapView.addAnnotation(annotation)
  
```

Далее создаются объекты класса `MKMapItem`, для инкапсуляции информации о конкретной точке на карте. Для создания объекта `MKMapItem`

не обходим объект класс MKPlacemark. Объект класса MKPlacemark хранит метки данные для заданной широты и долготы. Фрагмент кода создания объекта классов MKMapItem и MKPlacemark:

```
// Координаты места. CLLocationCoordinate2D содержит географическую координату с использованием системы координат WGS 84.
let location = CLLocationCoordinate2D(latitude: latitude, longitude: longitude)
// Создание объекта MKPlacemark с указанием координат
let placemark = MKPlacemark(coordinate: location)
// Создание объекта MKMapItem с указанием метки
let mapItem = MKMapItem(placemark: placemark)
```

Далее строится маршрут. Маршрут строится путем вычисления пути между двумя точками на карте. Для создания запроса построения маршрута между двумя точками используется класс MKDirectionsRequest. Для вычисления маршрута с помощью сервера Apple используется класс MKDirections. После успешного вычисления маршрута, он добавляется на карту и вычисляется регион, который будет отображаться на карте. Размер региона и местоположения определяются исходя из координат мест. Фрагмент кода создания запроса MKDirectionsRequest и вычисление маршрута:

```
// Создание запроса MKDirectionsRequest
let directionRequest = MKDirectionsRequest()
// Указывается начальная точка
directionRequest.source = mapItems[indexMapItem]
// Указывается конечная точка
directionRequest.destination = mapItems[indexMapItem + 1]
// Указывается тип передвижения, walking - пешком
directionRequest.transportType = .walking

// Создание маршрута, в виде параметра передается запрос MKDirectionsRequest
let directions = MKDirections(request: directionRequest)

// Вычисление маршрута
directions.calculate {
    (response, error) -> Void in
    // Результат выполнения вычисления
    // Если возникла ошибка, выводится в консоль
    guard let response = response else {
        if let error = error {
            print("Error: \(error)")
        }
    }
}
```



```

    }
    return
  }

  // Добавление маршрута на карту
  let route = response.routes[0]
  self.mapView.add((route.polyline), level: .aboveRoads)

  // Указывается какой регион на карте отображать
  let rect = route.polyline.boundingMapRect
  self.mapView.setRegion(MKCoordinateRegionForMapRect(rect), animated: true)
}

```

Результат выполнения последовательности действий по построению маршрута представлен на рисунке 25.

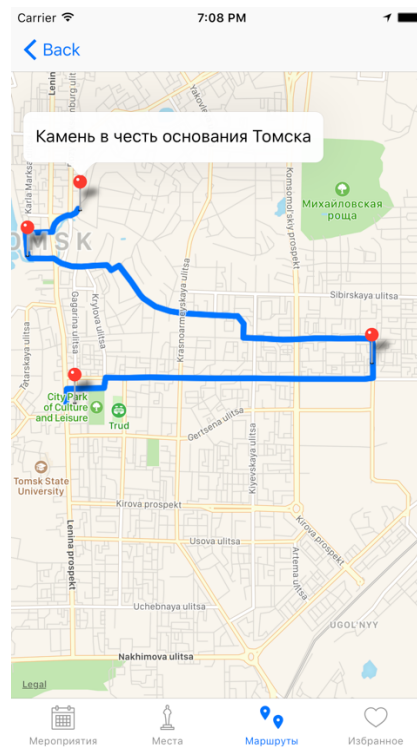


Рисунок 25 – Построенный маршрут на карте

UML диаграмма классов взаимодействия с картой показана на рисунке 26.

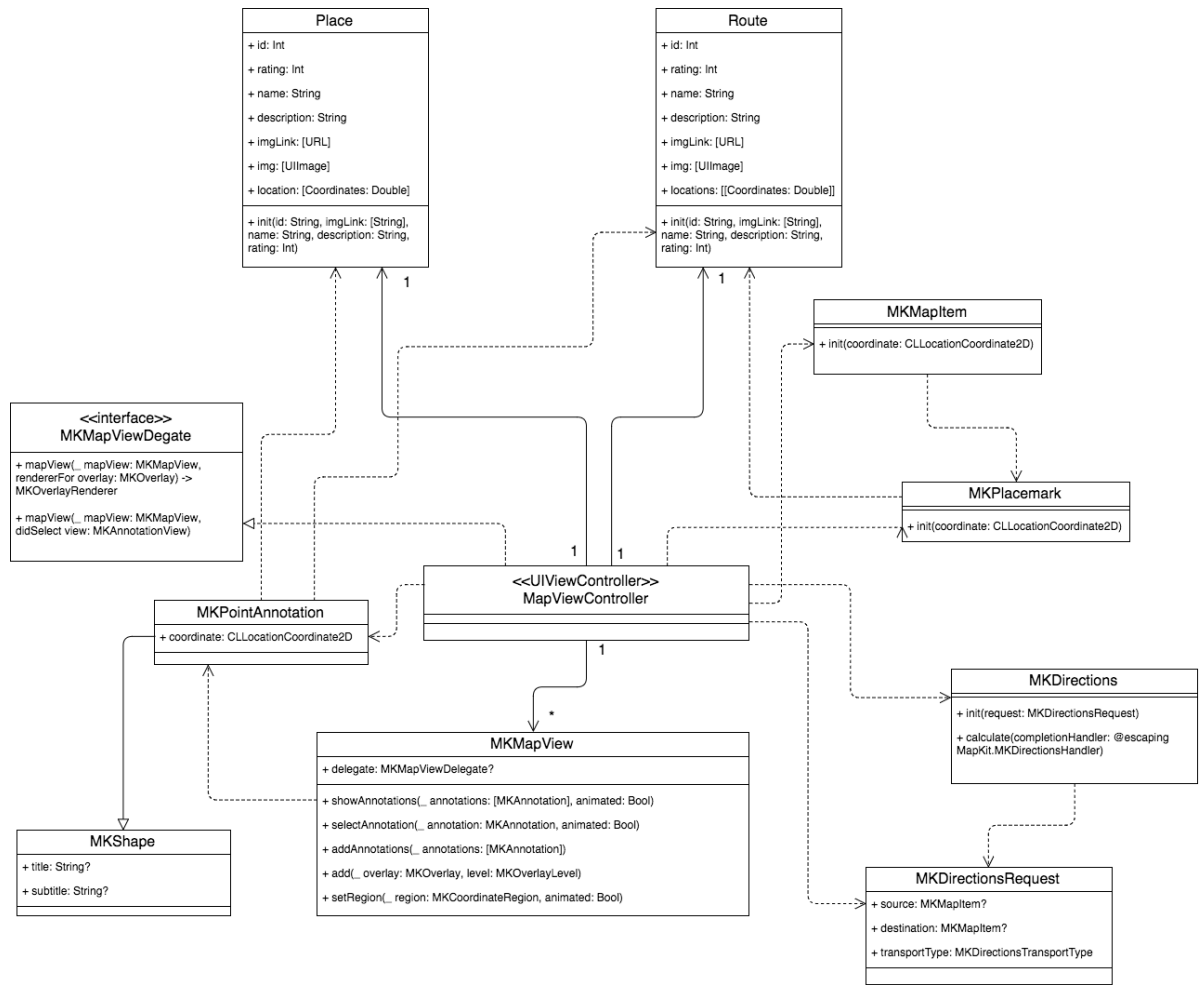


Рисунок 26 – UML диаграмма классов взаимодействия с картой  
**Детализированная диаграмма пакетов**

Визуальное представление зависимостей между частями уровня клиента представлено на UML диаграмме пакетов, рисунок 27.

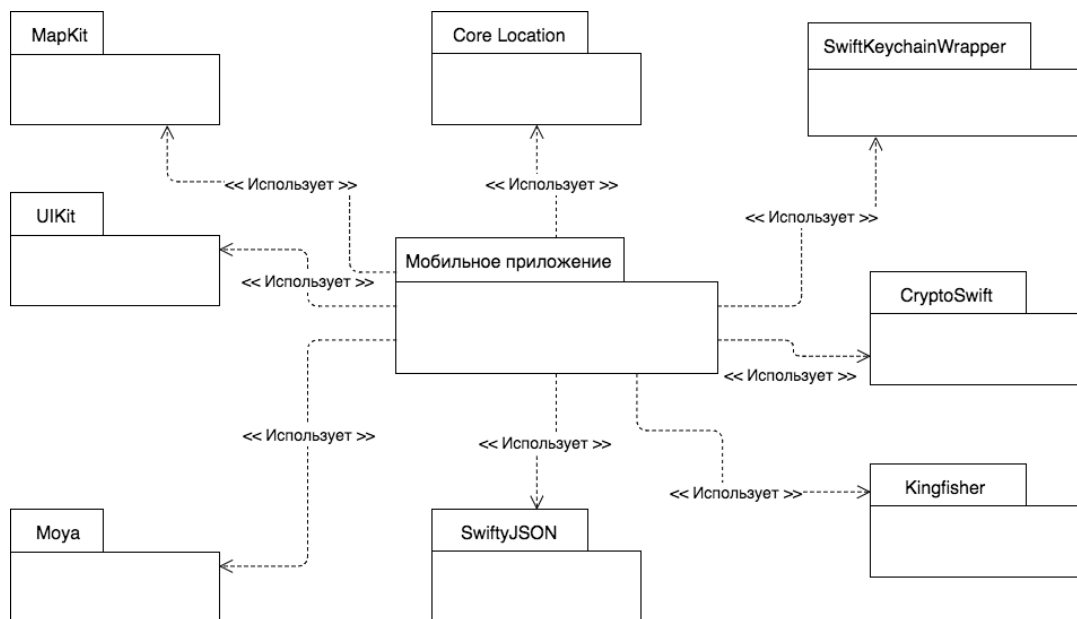


Рисунок 27 – UML детализированная диаграмма пакетов

### 4.3 Результаты развертывания (внедрения) и тестирования

Для отображения физического развертывания системы на вычислительных ресурсах во время выполнения используется диаграмма развертывания (deployment diagram) [41]. Диаграмма развёртывания системы представлено на рисунке 28.

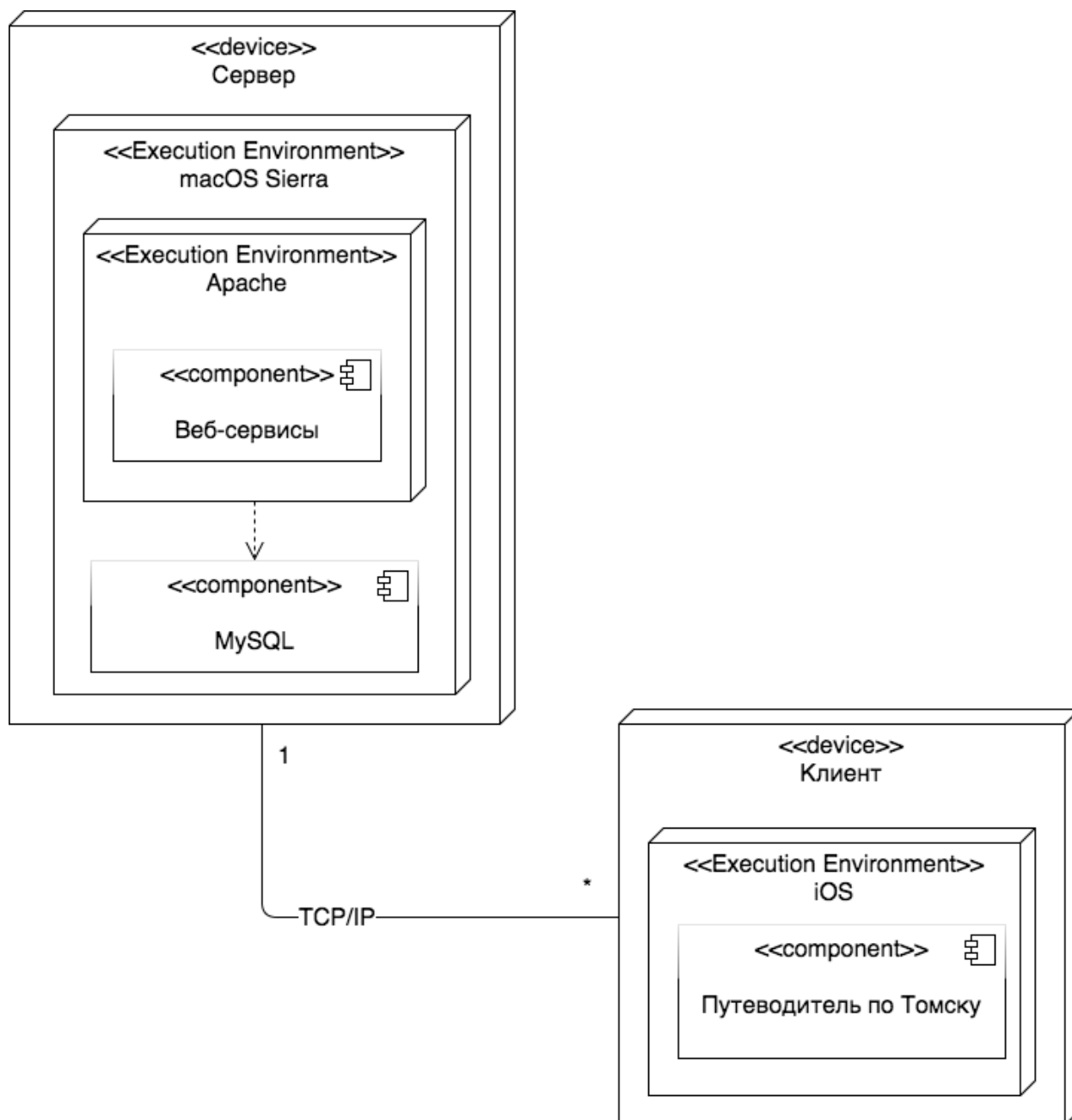


Рисунок 28 – UML диаграмма развертывания

На диаграмме развертывания показаны устройства сервер и клиент. На устройстве Сервер установлена операционная система macOS Sierra, на ней установлен веб-сервер Apache, на веб-сервере реализованы веб-сервисы. На

устройстве Клиент установлена операционная система iOS, на ней установлено приложение «Путеводитель по Томску».

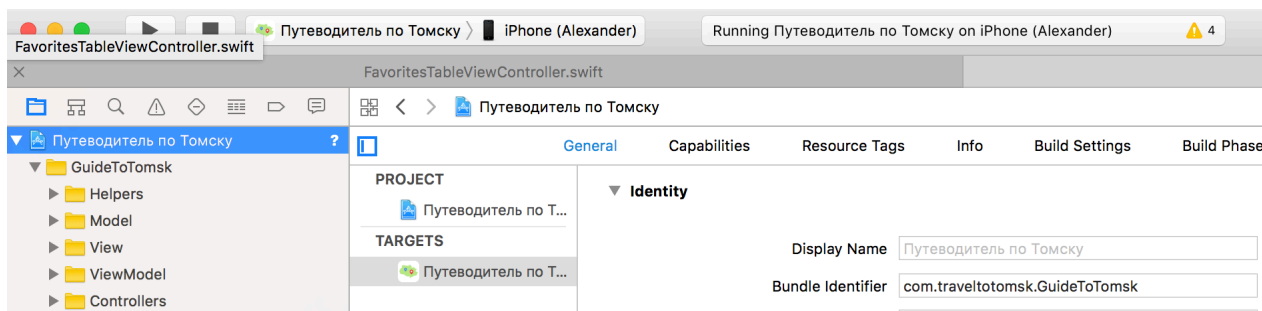
Описание процесса установки приложения в реальных условиях:

- Серверная часть размещается на хостинге;
- В базу данных загружаются данные о мероприятиях, местах и маршрутах;
- Клиентская часть. С помощью среды разработки Xcode, проект архивируется, в результате получается файл приложения с расширением ipa;
- Файл ipa загружается в магазин приложений App Store;
- Пользователь находит и скачивает приложение в магазине приложений App Store.
- Приложение устанавливается на мобильное устройство.

Описание процесса установки приложения в тестовом режиме:

- Сервер устанавливается локально, доступ к нему осуществляется по локальной сети;
- В базу данных загружаются данные о мероприятиях, достопримечательностях и маршрутах;
- Клиентская часть. С помощью среды разработки Xcode, проект собирается на подключенное устройство;

Сборка проекта на устройство и результат работы представлен на рисунке 29.



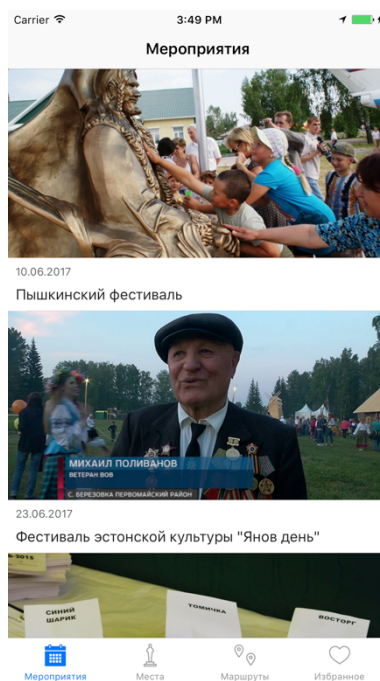
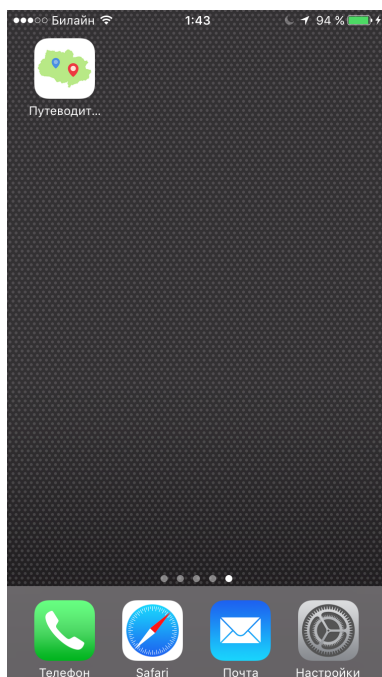


Рисунок 29 – Мобильное приложение собранное запущенное на iPhone

Пример возможности поделиться местом в социальных сетях представлен на рисунке 30.

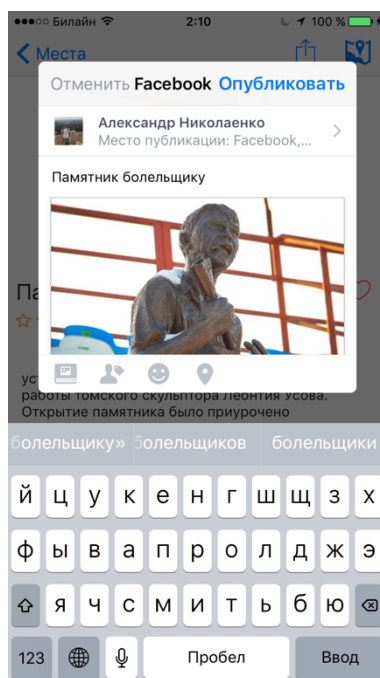
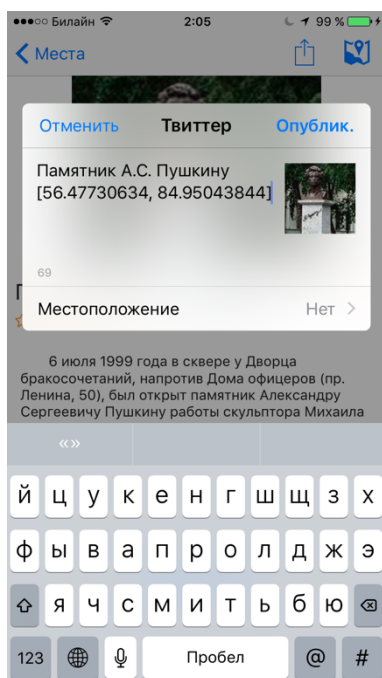


Рисунок 30 – Публикация в социальные сети

Пример возможности добавления комментариев и оценки маршрута показан на рисунке 31.

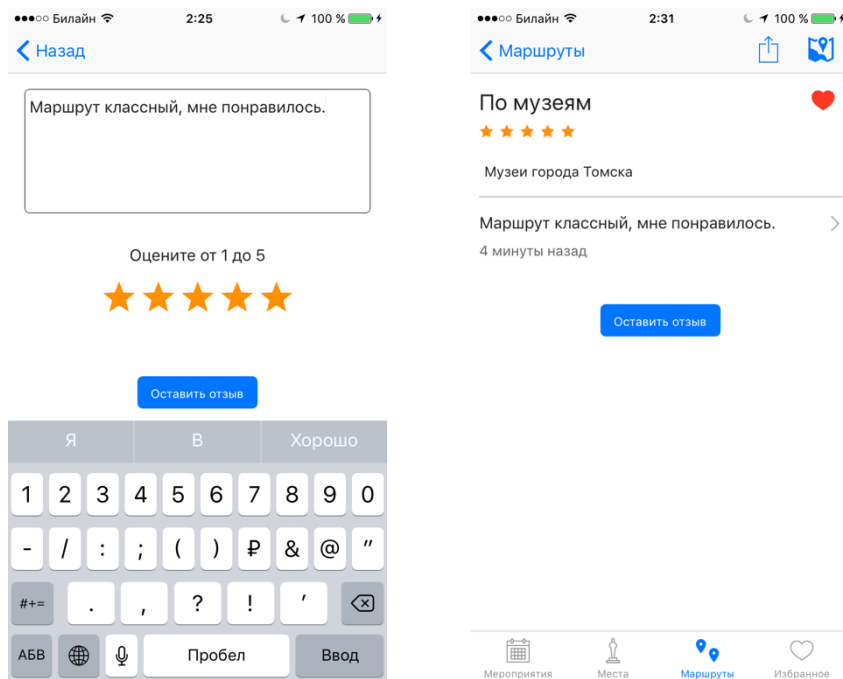


Рисунок 31 – Добавление оценки и комментария к маршруту

Пример построенного туристического маршрута представлен на рисунке 32.

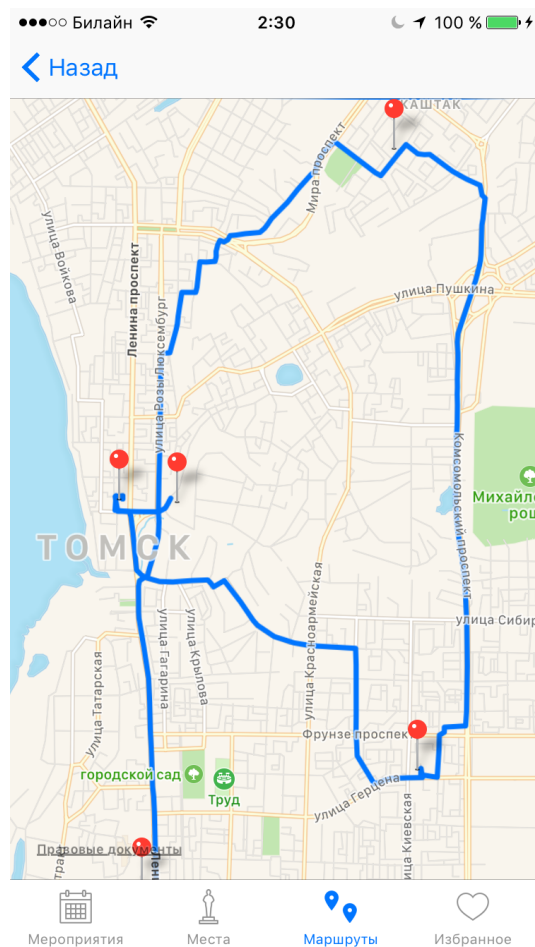


Рисунок 32 – Тестирование отображения выбранного маршрута

## **5 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение**

Процесс разработки программного приложения учитывает коммерческую перспективность с потенциалом, определяющих достаточный бюджет для продвижения создаваемого компонента.

Данный раздел, ресурсоэффективность и ресурсосбережение представляет собой коммерческую оценку программного продукта. Оценка, которая дает возможность увидеть у создаваемого продукта его коммерческую привлекательность на рынке, конкурентоспособность, учесть соответствие продукта современным требованиям в области ресурсосбережения и ресурсоэффективности.

После проведения анализа и оценки программного компонента, будет создан план научно-исследовательской работы, определен сегмент пользователей, произведена оценка эффективности разработанного компонента.

### **5.1 Цель работы и задачи**

Целью данной работы является создание мобильного приложения-путеводитель по городу Томску, позволяющему пользователю иметь всегда под рукой информацию о ближайших мероприятиях, список достопримечательностей города, интересных маршрутах.

Для реализации программного приложения должны быть решены следующие задачи:

- Проанализировать и определить требования к программному приложению.
- Разработать архитектуру программного обеспечения с учетом требований.
- Выбрать мобильную платформу для реализации.
- Выбрать язык программирования.
- Изучить возможные технологии выбранной платформы.

- Разработка мобильного приложения и графического интерфейса пользователя.

## **5.2 Техническое задание на разработку приложения**

Объектом проектирования является мобильное приложение-путеводитель, позволяющее:

- просматривать информацию о мероприятиях, достопримечательностях, маршрутах города Томска;
- просматривать достопримечательности и маршруты на карте города;
- сохранять понравившееся мероприятие, достопримечательность или маршрут в избранное;
- оставлять отзывы к достопримечательностям или маршрутам.

## **5.3 Актуальность проекта.**

На рынке мобильных приложений очень мало приложений, которые позиционируют, как подробный, постоянно обновляемый путеводитель по городу Томску для туристов и жителей города. В большинстве случаев, гид по городу Томску предлагается в группе с другими путеводителями и содержит устаревшую информацию о местах и мероприятиях города.

Разработка мобильного приложения-путеводитель по городу Томску, позволяющему пользователю иметь всегда под рукой информацию о ближайших мероприятиях, список достопримечательностей города, интересных маршрутах является актуальным. Актуальность проекта заключается в его эксклюзивности и ориентированности персонального на город Томск. Перспективность дает возможность развития в самое необходимое приложение-путеводитель с актуальной информацией о городе Томске.



## **5.4 Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения**

### **5.4.1 Потенциальные потребители результатов исследования**

Потенциальным потребителем программного продукта являются пользователи мобильных устройств, туристы или жители Томской области, которым интересны достопримечательности и мероприятия города. Это могут быть жители или туристы города Томска, которым интересны мероприятия, достопримечательности и интересные маршруты города.

### **5.4.2 Анализ конкурентных технических решений**

Анализ аналоговых решений позволяет выявить недостатки своего продукта и повысить тем самым конкурентоспособность. Так же анализ учитывает достоинства своего продукта и недостатки продуктов конкурентных.

Основными конкурентами программного продукта являются мобильные приложения, как «TopTripTip» (K1), «TripAdvisor» (K2) и «Google Trips» (K3). TopTripTip – созданный совместно с Телерадиокомпанией "МИР" – гид для крупных и самых отдаленных уголков России. TripAdvisor – автономный доступ к отзывам, фотографиям и картам более 300 городов по всему миру. Google Trips – приложение для путешественников, которое призвано предоставить всё, что может понадобиться в дороге.

Отличительным преимуществом разрабатываемого программного компонента сравнительно с конкурентными решениями вышеперечисленными, является содержание:

- постоянно обновляемого и актуального контента о мероприятиях и достопримечательностях;
- маршрутов по достопримечательностям города Томска.

Для подробного анализа конкурентных решений, составляется оценочная карта для сравнения конкурентных технических решений, представленная в таблице 2.

Таблица 2 – Оценка качества и перспективности проекта

Критерии оценки	Вес критерия	Баллы				Конкурентоспособность			
		Б <sub>ф</sub>	Б <sub>к1</sub>	Б <sub>к2</sub>	Б <sub>к3</sub>	К <sub>ф</sub>	К <sub>к1</sub>	К <sub>к2</sub>	К <sub>к3</sub>
1	2	3	4	5	6	7	8	9	10
<b>Технические критерии оценки ресурсоэффективности</b>									
1. Повышение производительности труда пользователя	0,2	5	3	3	2	0,6	0,3	0,2	0,2
2. Удобство в эксплуатации (соответствует требованиям потребителей)	0,2	4	4	2	4	1	0,8	0,6	0,8
3. Потребность в ресурсах памяти	0,1	5	3	2	2	0,4	0,8	0,4	0,5
4. Функциональная мощность (предоставляемые возможности)	0,5	3	2	2	2	2,6	1	1	1
	1					4,6	2,9	2,2	2,5
<b>Экономические критерии оценки эффективности</b>									
1. Конкурентоспособность продукта	0,5	4	3	4	4	2,5	1	2	2
2. Уровень проникновения на рынок	0,3	1	2	4	2	0,3	0,6	0,4	0,5
3. Цена	0,2	5	4	1	2	1	1,2	0,8	0,5
	1					3,8	2,8	3,2	3

Показатели критериев конкурентоспособности показывают, что конкурирующие приложения экономическая эффективность выше технической. Основные недостатки конкурирующих решений – это ограниченность функциональных возможностей в сравнении с возможностями программного приложения описанным в данной работе.

### 5.4.3 Технология QuaD

Технология QuaD оценивает рентабельность вложения средств в разрабатываемый продукт, а также дает возможность оценить перспективность и качество выполненной работы. Результат показателей оценки представлен в таблице 3.

Таблица 3 – Оценочная карта для сравнения конкурентных технических решений (разработок)

Критерии оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение (3/4)	Средневзвешенное значение (3x2)
1	2	3	4	5	
<b>Показатели оценки качества разработки</b>					
1. Потребность в ресурсах памяти	0,05	90	100	0,9	0,045
2. Удобство в эксплуатации	0,2	100	100	1	0,2
3. Безопасность системы	0,1	80	100	0,8	0,08
4. Функциональная мощность (предоставляемые возможности)	0,2	100	100	1	0,2
<b>Показатели оценки коммерческого потенциала разработки</b>					
5. Конкурентоспособность продукта	0,2	100	100	1	0,2
6. Уровень проникновения на рынок	0,2	90	100	0,9	0,18
7. Цена продукта	0,05	80	100	0,8	0,04
Итого	1				0,945

После получения среднего значение показателя перспективности и качества равно 0,945, а это значит, что исследуемая разработка считается перспективной.

#### 5.4.4 SWOT-анализ

SWOT- анализ определяет сильные и слабые стороны разработанного программного продукта, определяет угрозы и возможности, возникающие во внешней среде продукта.

Второй этап анализа выявляет соответствия между сильными и слабыми сторонами разработки внешним условиям ее окружающей среды. Это соответствие или несоответствие способствует тому чтобы выявить необходимость в проведения стратегических изменений.

Построенные в рамках второго этапа анализа, интерактивные матрицы проекта представлены в виде таблиц 4 – 8.

Таблица 4 – Интерактивная матрица сильных возможностей и сторон

Сильные стороны проекта					
Направления развития		C1	C2	C3	C4
	B1	+	-	+	+
	B2	-	-	+	+
	B3	+	+	-	-

Таблица 5 – Интерактивная матрица слабых сторон и возможностей

Слабые стороны проекта				
Сдерживающие факторы		Сл1	Сл2	Сл3
	B1	-	-	+
	B2	-	+	-
	B3	-	-	-

Таблица 6 – Интерактивная матрица сильных сторон и угроз

Сильные стороны проекта					
Угрозы развития		C1	C2	C3	C4
	У1	-	-	+	+
	У2	-	+	-	-
	У3	-	-	+	+

Таблица 7 – Интерактивная матрица слабых сторон и угроз

Слабые стороны проекта				
Уязвимости		Сл1	Сл2	Сл3
	У1	-	+	+
	У2	+	-	-
	У3	-	+	-

Итоговый результат SWOT-анализа представлен в таблице 8.

Таблица 8 – Матрица SWOT

	<p><b>Сильные стороны:</b>  С1. Современные технологии разработки.  С2. Уникальность функционала приложения.  С3. Аудитория пользователей широкая и целевая.  С4. Интерес к разработанному приложению.</p>	<p><b>Слабые стороны:</b>  Сл1. Требуется постоянное обновление контента.  Сл2. Необходимость в финансировании.  Сл3. Трудоемкость разработки.</p>
<p><b>Возможности:</b>  В1. Оптимизация работы с ресурсом.  В2. Расширение информационной составляющей ресурса.  В3. Финансирование проекта коммерческими организациями.</p>	<p><b>Направления развития:</b>  В1В2С3С4 – развитие функционала и интерфейса приложения, а также возможность расширения информационной составляющей ресурса.  В3С1С2 – привлечение финансирования коммерческих организаций за счет уникальности продукта.  В1С1 – оптимизация работы с ресурсом за счет использования современных технологий</p>	<p><b>Сдерживающие факторы:</b>  1. В1Сл3 – сложность в оптимизации продукта, за счет трудоемкости разработки.  2. В2Сл2 – сложность в расширении информационной составляющей ресурса за счет необходимости финансирования.</p>

Продолжение таблицы 8

<p><b>Угрозы:</b>          У1. Развитие и появление конкурентов новых, крупных конкурентов.          У2. Закрытие части информационных ресурсов.          У3. Непопулярность продукта на рынке</p>	<p><b>Угрозы развития:</b>          У1С3С4 – появление конкурентов может снизить спрос пользователей на продукт.          У2С2 – закрытие информационных ресурсов скажется на уникальность продукта на рынке.          У3С3С4 – непопулярность продукта на рынке приведет к потерям интереса широкой аудитории пользователей.</p>	<p><b>Уязвимости:</b>          У1Сл2Сл3 – появление сильных конкурентов может привести к потерям финансирования и усложнению разработки.          У2Сл1 – закрытие информационных ресурсов может затруднить обновление контента.          У3Сл2 - непопулярность продукта на рынке значительно может ограничить финансирование.</p>
--	---	---

### 5.5 Определение возможных альтернатив проведения научных исследований

С помощью морфологического подхода можно определить возможные альтернативы в проведении научных исследований. Данный подход и его результат использования представлен в таблице 9.

Таблица 9 – Морфологическая матрица

	1	2	3
А. Язык программирования для уровня клиента	Objective-C	Swift	Java
Б. Среда разработки	Android Studio	AppCode	Xcode
В. Язык программирования для уровня сервера приложений	PHP	Ruby	Python
Г. СУБД	MySQL	SQLite	PostgreSQL
Д. Тип хостинга	Платный	Бесплатный	Свой сервер
Е. Исполнитель	Один студент	Два студента	Программист

Из полученной морфологической матрицы, можно получить как минимум 3 варианта реализации и направления научных исследований при работе над проектом:

- Исполнение 1. А2Б3В1Г1Д1Е1.
- Исполнение 2. А1Б2В3Г3Д1Е2.
- Исполнение 3. А3Б1В2Г2Д2Е3.

В дальнейших расчетах именно эти варианты работы над проектом будут рассматриваться в качестве различных исполнений реализации разработки.

## 5.6 Планирование научно-исследовательских работ

### 5.6.1 Структура работ в рамках научного исследования

В научно-исследовательской работе важной частью планирования является виды запланированных работ и определение рабочей группы, которые выполняются участниками этой группы.

Рабочая группа в рамках описываемой работы состоит из двух участников: научный руководитель (руководитель) и студент.

Перечень этапов работ и распределения исполнителей представлен в таблице 10.

Таблица 10 – Перечень этапов, работ и распределение исполнителей

Основные этапы	№ раб	Содержание работ	Должность исполнителя
Разработка технического задания	1	Составление и утверждение технического задания	Руководитель
Выбор направления исследований	2	Поиск и изучение материалов по теме	Студент(ы)
	3	Анализ источников и представленной в них информации	Студент(ы)
	4	Календарное планирование работ по теме	Руководитель, студент(ы)
Проектирование приложения	5	Проектирование архитектуры	Руководитель, студент(ы)
	6	Выбор технологий разработки	Студент(ы)
	7	Выбор подходов к решению поставленной задачи	Студент(ы)
	8	Выбор средства обеспечения хранения данных	Студент(ы)
Реализация приложения	9	Реализация уровня клиента	Студент(ы)
	10	Реализация уровня сервера приложений	Студент(ы)
	11	Реализация уровня баз данных	Студент(ы)
Тестирование	12	Тестирование и исправление ошибок работы приложения	Студент(ы)
Анализ результатов работы и оформление пояснительной записки	13	Оценка соответствия программного приложения заявленным требованиям	Руководитель, студент(ы)
	14	Оформление пояснительной записки	Руководитель, студент(ы)

### 5.6.2 Определение трудоемкости выполнения работ

Для определения трудоемкости выполнения работ необходимо на основе экспертной ожидаемой трудоемкости выполнения каждой работы рассчитать длительность работ в рабочих и календарных днях для каждого из вариантов исполнения работ последующим формулам:

$$t_{\text{ож}i} = \frac{3t_{\text{min}i} + 2t_{\text{max}i}}{5}$$

Где  $t_{\text{ож}i}$  – ожидаемая трудоемкость выполнения  $i$ -ой работы чел.-дн.;



$t_{\min i}$  – минимально возможная трудоемкость выполнения заданной  $i$ -ой работы (оптимистическая оценка), чел.-дн.;

$t_{\max i}$  – максимально возможная трудоемкость выполнения заданной  $i$ -ой работы (пессимистическая оценка), чел.-дн.

Исходя из ожидаемой трудоемкости работ, определяется продолжительность каждой работы в рабочих днях  $T_p$ , учитывающая параллельность выполнения работ несколькими исполнителями. Такое вычисление необходимо для обоснованного расчета заработной платы, так как удельный вес зарплаты в общей сметной стоимости научных исследований составляет около 65 %.

$$T_{pi} = \frac{t_{ожi}}{Ч_i}$$

Где  $T_{pi}$  – продолжительность одной работы, раб. дн.;

$t_{ожi}$  – ожидаемая трудоемкость выполнения одной работы, чел.-дн.;

$Ч_i$  – численность исполнителей, выполняющих одновременно одну и ту же работу на данном этапе, чел.

Над выполнением дипломного проекта на протяжении всего времени участвовало два студента.

Расчеты продолжительности работ представлены в таблице 11

### 5.6.3 Разработка графика проведения научного исследования

Для сравнительно небольших по объему научных работ, наиболее удобным и наглядным является построение ленточного графика проведения научных работ в форме диаграммы Ганта.

*Диаграмма Ганта* – горизонтальный ленточный график, на котором работы по теме представляются протяженными во времени отрезками, характеризующимися датами начала и окончания выполнения данных работ.

Для удобства построения графика, длительность каждого из этапов работ из рабочих дней следует перевести в календарные дни. Для этого необходимо воспользоваться следующей формулой:

$$T_{ki} = T_{pi} \cdot k_{\text{кал}},$$

где  $T_{ki}$  – продолжительность выполнения  $i$ -й работы в календарных днях;

$T_{pi}$  – продолжительность выполнения  $i$ -й работы в рабочих днях;

$k_{\text{кал}}$  – коэффициент календарности.

Коэффициент календарности определяется по следующей формуле:

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{T_{\text{кал}} - T_{\text{вых}} - T_{\text{пр}}},$$

где  $T_{\text{кал}}$  – количество календарных дней в году;

$T_{\text{вых}}$  – количество выходных дней в году;

$T_{\text{пр}}$  – количество праздничных дней в году.

Рассчитанные значения в календарных днях по каждой работе  $T_{ki}$  необходимо округлить до целого числа.

Для расчета коэффициент календарности подсчитаем количество рабочих и выходных дней в 2017 году. Всего в году 247 рабочих дней и 118 выходных и праздничных дней. Исходя из полученных данных, рассчитывается коэффициент календарности:

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{T_{\text{кал}} - T_{\text{вых}} - T_{\text{пр}}} = \frac{365}{247} = 1,4778$$

На основе таблицы 11 построен календарный план-график проведения работ (таблица 12).

Таблица 11 – Временные показатели проведения научного исследования

Название работы	Исполнители	Трудоёмкость работ									Длительность работ в рабочих днях $T_{pi}$			Длительность работ в календарных днях $T_{ki}$		
		$t_{min}$ , чел-дни			$t_{max}$ , чел-дни			$t_{ожг}$ , чел-дни								
		Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3
Составление и утверждение технического задания	Руководитель	3	4	4	6	8	8	4	6	6	4	6	6	6	8	8
Поиск и изучение материалов по теме	Студент	5	2	7	7	3	9	6	2	8	6	2	8	9	4	12
Анализ источников и представленной в них информации	Студент	2	3	2	4	5	4	3	4	3	3	4	3	4	6	4
Календарное планирование работ по теме	Руководитель, студент	2	2	2	3	3	3	2	2	2	1	2	2	2	4	4
Проектирование архитектуры	Руководитель, студент	4	3	3	5	4	5	4	3	4	2	3	4	3	5	6
Выбор технологий разработки	Студент	4	2	3	8	5	7	6	3	5	6	3	5	8	5	7
Выбор подходов к решению поставленной задачи	Студент	3	2	2	4	3	3	3	2	2	3	2	2	5	4	4
Выбор средства обеспечения хранения данных	Студент	7	3	5	9	5	7	8	4	6	8	4	6	12	6	9
Реализация уровня клиента	Студент	15	11	13	20	15	18	17	13	15	17	13	15	25	19	22

Продолжение таблицы 11

Название работы	Исполнители	Трудоёмкость работ									Длительность работ в рабочих днях $T_{pi}$			Длительность работ в календарных днях $T_{ki}$		
		$t_{min}$ , чел-дни			$t_{max}$ , чел-дни			$t_{ожид}$ , чел-дни								
		Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3
Реализация уровня сервера приложений	Студент	12	10	13	15	13	17	13	11	15	13	11	15	20	17	22
Реализация уровня баз данных	Студент	4	2	3	5	3	4	4	2	3	4	2	3	7	4	5
Тестирование и исправление ошибок работы приложения	Студент	3	2	3	4	3	4	3	2	3	3	2	3	5	4	5
Оценка соответствия программного приложения заявленным требованиям	Руководитель, студент	3	2	2	6	5	5	4	3	3	2	3	3	3	5	5
Оформление пояснительной записки	Руководитель, студент	3	3	4	4	4	5	3	3	4	2	3	4	3	5	7
<b>Итого</b>								<b>82</b>	<b>62</b>	<b>79</b>	<b>75</b>	<b>62</b>	<b>79</b>	<b>111</b>	<b>92</b>	<b>117</b>

Таблица 12 – Календарный план-график проведения работ

№ работ	Вид работ	Исполнители	T <sub>кi</sub> , кал. дн.	Продолжительность выполнения работ											
				февраль			март			апрель			май		
				1	2	3	1	2	3	1	2	3	1	2	3
1	Составление и утверждение технического задания	Руководитель	8	■											
2	Поиск и изучение материалов по теме	Студент	12	■	■	■									
3	Анализ источников и представленной в них информации	Студент	4			■									
4	Календарное планирование работ по теме	Руководитель, студент	4			■									
5	Проектирование архитектуры	Руководитель, студент	6				■			■					
6	Выбор технологий разработки	Студент	7					■							
7	Выбор подходов к решению поставленной задачи	Студент	4						■						
8	Выбор средства обеспечения хранения данных	Студент	9						■	■					
9	Реализация уровня клиента	Студент	22						■	■	■	■			
10	Реализация уровня сервера приложений	Студент	22									■	■	■	■
11	Реализация уровня баз данных	Студент	5									■	■	■	

Продолжение таблицы 12

№ работ	Вид работ	Исполнители	$T_{ki}$ , кал. дн.	Продолжительность выполнения работ											
				февраль			март			апрель			май		
				1	2	3	1	2	3	1	2	3	1	2	3
12	Тестирование и исправление ошибок работы приложения	Студент	5											■	
13	Оценка соответствия программного приложения заявленным требованиям	Руководитель, студент	5											■	
														▨	
14	Оформление пояснительной записки	Руководитель, студент	7												■
															▨

▨ – Руководитель

■ – Студент

## 5.7 Бюджет научно-технического исследования

В состав бюджета выполнения работ по научно-технической работе включает вся себя стоимость всех расходов, необходимых для их выполнения. При формировании бюджета используется группировка затрат по следующим статьям:

- основная заработная плата исполнителей темы;
- дополнительная заработная плата исполнителей темы;
- отчисления во внебюджетные фонды (страховые отчисления);
- накладные расходы.

### 5.7.1 Расчет материальных затрат НИИ

Данная статья включает стоимость всех материалов, используемых при разработке проекта.

Расчет материальных затрат осуществляется по следующей формуле:

$$Z_m = (1 + k_T) \cdot \sum_{i=1}^m \Pi_i \cdot N_{расхи}$$

где  $m$  - количество видов материальных ресурсов, потребляемых при выполнении научного исследования;

$N_{расхи}$  – количество материальных ресурсов  $i$ -го вида, планируемых к использованию при выполнении научного исследования (шт., кг, м, м<sup>2</sup> и т.д.);

$\Pi_i$  – цена приобретения единицы  $i$ -го вида потребляемых материальных ресурсов (руб./шт., руб./кг, руб./м, руб./м<sup>2</sup> и т.д.);

$k_T$  – коэффициент, учитывающий транспортно-заготовительные расходы.

Величина коэффициента ( $k_T$ ), отражающего соотношение затрат по доставке материальных ресурсов и цен на их приобретение, зависит от условий договоров поставки, видов материальных ресурсов, территориальной удаленности поставщиков и т.д. Транспортные расходы принимаются в пределах 15-25% от стоимости материалов. Материальные затраты, необходимые для данной разработки, заносятся в таблицу 13.

Таблица 13 – Материальные затраты

Наименование	Ед. измерения	Количество			Цена за ед., тыс. руб.			Затраты на материалы, (З <sub>м</sub> ), тыс. руб.		
		Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3
Амортизация оборудования	шт.	1	2	1	4,5	4,5	4	4,5	9	4
Среда разработки AppCode	шт.	0	2	0	0	3	0	0	6	0
Хостинг	шт.	1	0	0	0	0	2,3	2,3	0	0
Итого								6,8	15	4

В ходе разработки использовалось имеющееся оборудование, поэтому в материальные расходы внесены затраты на его амортизацию за 4 месяца. Также в материальные расходы занесены затраты на приобретение лицензий для используемого программного обеспечения и аренды хостинга.

### 5.7.2 Расчет основной заработной платы исполнителей системы

В данную статью расходов включается заработная плата научного руководителя и студентов, а также премии и доплаты. Расчет выполняется на основе трудоемкости выполнения каждого этапа и величины почасовой оплаты работы исполнителей.

Основной расчет фонда заработной платы выполняется по формуле:

$$Z_{зп} = Z_{осн} + Z_{доп}$$

где  $Z_{осн}$  – основная заработная плата;

$Z_{доп}$  – дополнительная заработная плата (12-20 % от  $Z_{осн}$ ).

Основная заработная плата ( $Z_{осн}$ ) руководителя (лаборанта, инженера) от предприятия (при наличии руководителя от предприятия) рассчитывается по следующей формуле:

$$Z_{осн} = Z_{дн} \cdot T_p,$$

где  $Z_{осн}$  – основная заработная плата одного работника;



$T_p$  – продолжительность работ, выполняемых научно-техническим работником, раб. часов.

$Z_{дн}$  – среднедневная заработная плата работника, руб.

Таблица 14 – Расчет основной заработной платы

№ п/п	Наименование этапов	Исполнители по категориям	Трудоемкость, чел.-дн.			Зарплата, приходящаяся на один чел.-дн., тыс. руб.			Всего заработная плата по тарифу (окладам), тыс. руб.		
			Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3
1	Составление и утверждение технического задания	Руководитель	4	6	6	1,70	1,70	1,70	6,80	10,20	10,20
2	Поиск и изучение материалов по теме	Студент	6	7	8	0,15	0,30	2,00	0,90	0,60	16,00
3	Анализ источников и представленной в них информации	Студент	3	4	3	0,15	0,30	2,00	0,45	1,20	6,00
4	Календарное планирование работ по теме	Руководитель, студент	2	2	2	1,85	2,00	3,70	3,70	4,00	7,40
5	Проектирование архитектуры	Руководитель, студент	4	3	4	1,85	2,00	3,70	7,40	6,00	14,80
6	Выбор технологий разработки	Студент	6	3	5	0,15	0,30	2,00	0,90	0,90	10,00
7	Выбор подходов к решению поставленной задачи	Студент	3	2	2	0,15	0,30	2,00	0,45	0,60	4,00

Продолжение таблицы 14

№ п/п	Наименование этапов	Исполнители по категориям	Трудоемкость, чел.-дн.			Зарботная плата, приходящаяся на один чел.-дн., тыс. руб.			Всего заработная плата по тарифу (окладам), тыс. руб.		
			Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3
8	Выбор средства обеспечения хранения данных	Студент	8	4	6	0,15	0,30	2,00	1,20	1,20	12,00
9	Реализация уровня клиента	Студент	17	18	15	0,15	0,30	2,00	2,55	3,90	30,00
10	Реализация уровня сервера приложений	Студент	13	12	15	0,15	0,30	2,00	1,95	3,30	30,00
11	Реализация уровня баз данных	Студент	4	4	3	0,15	0,30	2,00	0,60	0,60	6,00
12	Тестирование и исправление ошибок работы приложения	Студент	2	3	3	0,15	0,30	2,00	0,45	0,60	6,00
13	Оценка соответствия программного приложения заявленным требованиям	Руководитель, студент	4	3	3	1,85	2,00	3,70	7,40	6,00	11,10
14	Оформление пояснительной записки	Руководитель, студент	3	3	4	1,85	2,00	3,70	5,55	6,00	14,80
Итого:									<b>40,30</b>	<b>45,10</b>	<b>178,30</b>

### 5.7.3 Отчисления во внебюджетные фонды (страховые отчисления)

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам государственного социального страхования (ФСС), пенсионного фонда (ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников. Величина отчислений во внебюджетные фонды определяется исходя из следующей формулы:

$$З_{внеб} = k_{внеб} \cdot (З_{осн} + З_{доп})$$

где  $k_{внеб}$  – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.).

На 2017 г. в соответствии с Федеральным законом от 24.07.2009 №212-ФЗ установлен размер страховых взносов равный 30%.

Отчисления во внебюджетные фонды представлены в таблице 15.

Таблица 15 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата, руб.		
	Исп.1	Исп.2	Исп.3
Руководитель проекта	4400	4400	4400
Студент-дипломник	35900	40700	-
Программист	-	-	173900
Итого	40300	45100	178300
Коэффициент отчислений во внебюджетные фонды	0,3		
Коэффициент			
Исполнение 1	12090		
Исполнение 2	13530		
Исполнение 3	53490		

### 5.7.4 Расчет накладных расходов

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование

материалов исследования, оплата услуг связи, электроэнергии, размножение материалов и т.д. Их величина определяется по следующей формуле:

$$Z_{\text{накл}} = (\text{сумма статей } 1 \div 7) \cdot k_{\text{нр}}$$

где  $k_{\text{нр}}$  – коэффициент, учитывающий накладные расходы.

Величину коэффициента накладных расходов можно взять в размере 16%. В рассматриваемом случае были использованы лишь 4 статьи, поэтому деление производится на 4.

Для исполнения 1:  $Z_{\text{накл}} = 12090 \cdot 0,16 = 1934,4$

Для исполнения 2:  $Z_{\text{накл}} = 13530 \cdot 0,16 = 2164,8$

Для исполнения 3:  $Z_{\text{накл}} = 53490 \cdot 0,16 = 8558,4$

### 5.7.5 Формирование бюджета затрат научно-исследовательского проекта

Сумма затрат по всем статьям расходов рассчитывается, заносится на данном этапе в таблицу 16.

Таблица 16 – Бюджет затрат научно-исследовательского проекта

Наименование статьи	Сумма, руб.			Примечание
	Исп.1	Исп.2	Исп.3	
1. Материальные затраты НТИ	6,8	15	4	Пункт 5.7.1
2. Затраты по основной заработной плате исполнителей темы	40300	45100	178300	Пункт 5.7.2
3. Отчисления во внебюджетные фонды	12090	13530	53490	Пункт 5.7.3
4. Накладные расходы	1934,4	2164,8	8558,4	16 % от суммы ст. 1-4
5. Бюджет затрат НТИ	54331,2	60809,8	240352,4	Сумма ст. 1- 5

### 5.8 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его

нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}}$$

где  $I_{\text{финр}}^{\text{исп.}i}$  – интегральный финансовый показатель разработки;

$\Phi_{pi}$  – стоимость  $i$ -го варианта исполнения;

$\Phi_{\text{max}}$  – максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги).

Исполнение 1:  $I_{\text{финр}} = 54331,2/240352,4 = 0,226$

Исполнение 2:  $I_{\text{финр}} = 60809,8/240352,4 = 0,253$

Исполнение 3:  $I_{\text{финр}} = 240352,4/240352,4 = 1$

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное удешевление стоимости разработки в размах.

**Интегральный показатель ресурсоэффективности** вариантов исполнения объекта исследования можно определить следующим образом:

$$I_{pi} = \sum a_i \cdot b_i,$$

где  $I_{pi}$  – интегральный показатель ресурсоэффективности для  $i$ -го варианта исполнения разработки;

$a_i$  – весовой коэффициент  $i$ -го варианта исполнения разработки;

$b_i^a, b_i^p$  – бальная оценка  $i$ -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

$n$  – число параметров сравнения.

Расчет интегрального показателя ресурсоэффективности рекомендуется проводить в форме таблицы (таблица 17)

Таблица 17 – Сравнительная оценка характеристик вариантов исполнения проекта

Критерии \ Объект исследования	Весовой коэффициент параметра	Исп.1	Исп.2	Исп.3
1. Функциональная мощность (предоставляемые возможности)	0,21	5	4	4
2. Скорость работы	0,13	3	5	4
3. Потребность в ресурсах памяти	0,12	4	4	4
4. Зависимость от платформы	0,15	4	4	5
5. Удобство эксплуатации	0,21	5	4	4
6. Качество предоставляемого интерфейса	0,18	5	4	4
ИТОГО	1			

$$I_{p-исп1} = 0,21 * 5 + 0,13 * 3 + 0,12 * 4 + 0,15 * 4 + 0,21 * 5 + 0,18 * 5 = 4,47$$

$$I_{p-исп2} = 0,21 * 4 + 0,13 * 5 + 0,12 * 4 + 0,15 * 4 + 0,21 * 4 + 0,18 * 5 = 4,13$$

$$I_{p-исп3} = 0,21 * 4 + 0,13 * 4 + 0,12 * 4 + 0,15 * 5 + 0,21 * 4 + 0,18 * 4 = 4,15$$

**Интегральный показатель эффективности вариантов исполнения разработки** ( $I_{исп.i.}$ ) определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{исп.1} = \frac{I_{p-исп1}}{I_{финр}}$$

$$I_{исп1} = 4,47 / 0,226 = 19,78$$

$$I_{исп2} = 4,13 / 0,253 = 16,32$$

$$I_{исп3} = 4,15 / 1 = 4,15$$

Полученное значение интегрального показателя эффективности исполнения разработки превысило максимальный балл оценки. Таким образом, результат работы можно считать положительным, так как оценка интегрального показателя ресурсоэффективности близка к максимальной, при этом стоимость разработки ниже, чем у ряда аналогов, рассмотренных при анализе конкурентных решений.

Сравнение интегрального показателя эффективности вариантов исполнения разработки позволит определить сравнительную эффективность проекта и выбрать наиболее целесообразный вариант из предложенных.

**Сравнительная эффективность проекта ( $\mathcal{E}_{cp}$ ):**

$$\mathcal{E}_{cp\_i} = \frac{I_{исп.\_i}}{I_{исп.\_min}}$$

Сравнительная эффективность разработки представлена в таблице 21.

Таблица 21 – Сравнительная эффективность разработки

№ п/п	Показатели	Исп. 1	Исп. 2	Исп. 3
1	Интегральный финансовый показатель разработки	0,226	0,253	1
2	Интегральный показатель ресурсоэффективности разработки	4,47	4,13	4,15
3	Интегральный показатель эффективности	19,78	16,32	4,15
4	Сравнительная эффективность вариантов исполнения	4,77	3,93	1

### **Заключение**

В результате проведения научного исследования ресурсоэффективности и ресурсосбережения коммерческого потенциала для работы «Разработка мобильного приложения-путеводитель по городу Томску» были определены потенциальные потребители разработки – пользователи мобильных устройств, туристы или жители Томской области, которым интересны достопримечательности и мероприятия города

Был произведен анализ конкурентных технических решений, результаты которого показали, что разрабатываемое приложение имеет уникальные функциональные возможности, в отличии от аналогов на рынке.

Были определены возможные альтернативы проведения и варианты реализации и направления научных исследований.

Исходя из анализа эффективности и сравнения альтернатив реализации, можно сделать вывод, что первый вариант исполнения является наиболее экономичным и эффективным, его интегральные показатели ресурсоэффективности разработки и эффективности выше, чем у других рассмотренных вариантов. Именно поэтому был выбран первый вариант исполнения.



## **6 Социальная ответственность**

### **ВВЕДЕНИЕ**

Целью данной работы является создание мобильного приложения-путеводитель по городу Томску, позволяющему пользователю иметь всегда под рукой информацию о ближайших мероприятиях, список достопримечательностей города, интересных маршрутах. Мобильное приложение позволит обмениваться впечатлениями и эмоциями о путешествии при помощи комментариев к конкретному месту или маршруту, оценивать на сколько посещённое им место интересно и заслуживает внимание других путешественников.

В настоящем разделе представлено описание вредных и опасных факторов, создаваемых объектом исследования в процессе разработки, а также мероприятий, обеспечивающих производственную безопасность. Опасными или вредными факторами могут являться повышенная напряженность электромагнитного поля, неподходящий микроклимат, неподходящая естественная освещенность, недостаточная освещенность рабочей зоны, повышенный уровень шума в рабочем помещении

При проведении теоретических изысканий, анализа и структурирования полученных данных, проектирования и программирования приложения использовались средства вычислительной техники. Данный фактор накладывает ограничения, вызванные вредностью для человека процесса разработки. В результате у него могут начаться проблемы со здоровьем, что приведет к снижению производительности труда.

#### **6.1 Производственная безопасность**

Разработка мобильного приложения предполагает работу за персональным компьютером. В связи с этим, возникает влияние на человека вредных и опасных факторов, приведенных в 22.

Таблица 22 – Опасные и вредные факторы при выполнении работ с автоматизированной беспроводной системой измерений параметров окружающей среды

Источник фактора, наименование видов работ	Факторы (по ГОСТ 12.0.003-2015)		Нормативные документы
	Вредные	Опасные	
Разработка мобильного приложения - путевоитель	1. Повышенная напряженность электромагнитного поля; 2. Отклонение показателей микроклимата; 3. Недостаток естественного освещения; 4. Недостаточная освещенность рабочей зоны; 5. Повышенный уровень шума;	1. Электрический ток.	Гигиенические требования к персональным электронно-вычислительным машинам и организации работы устанавливаются Сан-ПиН 2.2.2/2.4.1340-03I [42]. Параметры электромагнитного излучения устанавливаются СанПиН 2.2.4.3359-16 [43]. Параметры микроклимата устанавливаются СанПиН 2.2.4.548-96 [44]. Требования к естественному и искусственному освещению устанавливаются СП 52.13330.2011 [45]. Нормируемые параметры и предельно допустимые уровни шума на рабочих местах устанавливаются СН 2.2.4/2.1.8.562-96 [46]. Требования к оценке шума устанавливаются ГОСТ 12.1.003-2014 [47].

### 6.1.1 Повышенная напряженность электромагнитного поля

Источниками повышенной напряженности электромагнитного поля в данном случае является персональный компьютер.

Согласно СанПиН 2.2.4.1191–03 «Электромагнитные поля в производственных условиях», 8 часовой рабочий день для сотрудника на своем рабочем месте, с предельно допустимым уровнем напряженности должен составлять не более 8 кА/м, а уровень магнитной индукции – 10 мТл. Соблюдение данных норм дает возможность избежать негативного воздействия электромагнитных излучений [48].

Для уменьшения уровня электромагнитного поля от персонального компьютера рекомендуется включать в одну розетку не более двух компьютеров, сделать защитное заземление, подключать компьютер к розетке через нейтрализатор электрического поля.

К средствам индивидуальной защиты при работе на компьютере относят спектральные компьютерные очки для улучшения качества изображения, защиты от избыточных энергетических потоков видимого света и для профилактики «компьютерного зрительного синдрома». Очки уменьшают утомляемость глаз на 25-30 %. Их рекомендуется применять всем операторам при работе более 2 ч в день, а при нарушении зрения на 2 диоптрии и более – независимо от продолжительности работы.

### **6.1.2 Отклонение показателей микроклимата**

Микроклимат является одним из важных условий труда. Его основные параметры регулируются СанПиН 2.2.4.548-96 [44] и ГОСТ 12.1.005-88 [49]. Персональные компьютеры и серверное оборудование выделяют большое количество тепловой энергии. В купе с недостатком воздуха в рабочей зоне данный фактор оказывает негативное влияние на человека.

Психоэмоциональная обстановка в коллективе, работающем в помещении может потерять устойчивость и вызывать конфликты. Также велик риск роста болезнетворных бактерий, вызывающих различные заболевания. При недостаточной влажности воздуха слизистые оболочки человека постепенно осушаются. Данный фактор может приводить к повышенному накоплению вредных бактерий и вирусов в дыхательных путях, затруднять работу головного мозга и других органов.

С другой стороны, повышенная влажность способствует активному размножению бактерий, плесени и грибов, что в свою очередь может привести к возникновению аллергических и хронических заболеваний дыхательных путей человека.

Оптимальные величины параметров микроклимата, создающие комфортные условия представлены в таблице 23

Таблица 23 – Оптимальные величины параметров микроклимата на рабочих местах производственных помещений

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	22-24	21-25	60-40	0,1
Теплый	23-25	22-26	60-40	0,1

Допустимые величины параметров микроклимата, создающие комфортные условия представлены в таблице 24.

Таблица 24 – Допустимые величины параметров микроклимата на рабочих местах производственных помещений

Период года	Температура воздуха, °С		Температура поверхностей, °С	Относительная влажность	Скорость движения воздуха, м/с	
	диапазон ниже оптимальных величин	диапазон выше оптимальных величин			для диапазона температур воздуха ниже оптимальных величин, не более	для диапазона температур воздуха выше оптимальных величин, не более
Холодный	20,0-21,9	24,1-25,0	19,0-26,0	15-75	0,1	0,1
Теплый	21,0-22,9	25,1-28,0	20,0-29,0	15-75	0,1	0,2

Исходя из требований, рассматриваемых в данном разделе нормативных документов, в помещении поддерживается температура равная 19–20 С°, при относительной влажности в 55–58%. В помещении функционирует система принудительной вентиляции. Проводится ежедневная влажная уборка и систематическое проветривание помещения. В зимнее время в помещении предусмотрена система водяного отопления со встроенными нагревательными элементами и терморегуляторами.

### **6.1.3 Недостаточная освещенность**

Естественное и искусственное освещение рабочего места оказывает влияние на физическое состояние и на работу сотрудника. Не надлежащего качество освещения ведет к ухудшению зрения работника.

Согласно СанПиН 2.2.2/2.4.1340-03, на поверхности рабочего стола освещенность пользователя ПЭВМ должна быть 300 – 500 лк. При освещении блики должны отсутствовать на поверхности экрана. Поверхность экрана должна быть до 300 лк [42].

Существуют общие требования и рекомендации к организации освещения на рабочем месте:

- Система общего равномерного должно регулировать искусственное освещение в помещениях для эксплуатации ПЭВМ.
- Рабочие места следует размещать таким образом, чтобы естественный свет падал преимущественно слева, а дисплеи монитора были ориентированы боковой стороной к световым проемам.

Соблюдение данных мер позволит сохранить зрение работника или избежать пагубного воздействия на глаза.

### **6.1.4 Повышенный уровень шума;**

Шум – это совокупность звуков, неблагоприятно воздействующих на организм человека и мешающих его работе и отдыху. Источниками звука

являются колебания материальных частиц и тел, передаваемых жидкой, твердой и газообразной средой.

Допустимый уровень шума – это уровень, который не вызывает у человека значительного беспокойства и существенных изменений показателей функционального состояния систем и анализаторов, чувствительных к шуму

Ненормированный показатель шума на рабочих местах оказывает влияние на психологическое состояние работника. У сотрудника на поставленной ему задаче снижаются концентрация и сосредоточенность, а повышается уровень утомляемости и стресса. Повышенный уровень шума может привести к нарушению слуха или являться помехой для коммуникаций между сотрудниками.

Для избегания вышеуказанных последствий воздействия описываемого фактора, необходимо соблюдать следующие требования, обозначенные в СанПиН 2.2.2/2.4.1340–03. В таблице 25 приведены допустимые значения уровней звукового давления в октавных полосах частот и уровня звука, создаваемого ПЭВМ.

В таблице 25 – Допустимые значения уровней звукового давления в октавных полосах частот и уровня звука, создаваемого ПЭВМ.

Уровни звукового давления в октавных полосах со среднегеометрическими частотами									Уровни звука в дБА
31,5 Гц	63 Гц	125 Гц	250 Гц	500 Гц	1000 Гц	2000 Гц	4000 Гц	8000 Гц	
86 дБ	71 дБ	61 дБ	54 дБ	49 дБ	45 дБ	42 дБ	40 дБ	38 дБ	50

Измерение уровня звука и уровней звукового давления проводится на расстоянии 50 см от поверхности оборудования и на высоте расположения источника(ков) звука [42].

## 6.2 Экологическая безопасность

Так как основным объектом исследования данной работы являются электрические приборы, серьезной проблемой является электропотребление.

Это влечет за собой общий рост объема потребляемой электроэнергии. Для удовлетворения потребности в электроэнергии, приходится увеличивать мощность и количество электростанций. Это приводит к нарушению экологической обстановки, так как электростанции в своей деятельности используют различные виды топлива, водные ресурсы, а также являются источником вредных выбросов в атмосферу.

Для защиты окружающей среды от вредного воздействия выбросов необходимо осуществить полный переход к энергосберегающим и малоотходным (в идеале безотходным) производствам. Также на данный момент во многих странах используются альтернативные источники энергии (солнечные батареи, энергия ветра).

В лаборатории не ведется никакого производства. К отходам, производимым в помещении можно отнести, в первую очередь, это бумажные отходы – макулатура, пластиковые отходы, неисправные детали персональных компьютеров и других видов ЭВМ. Бумажные отходы, рекомендуется накапливать и передавать их в пункты приема макулатуры для дальнейшей переработки [50]. Пластиковые бутылки складывать в специально предназначенные контейнеры. Неисправные комплектующие персональных компьютеров и других ЭВМ, а также люминесцентные лампы возможно передавать организациям, занимающимся переработкой отходов [51, 52] или продавать на вторичном рынке.

Дополнительным методом снижения отходов является увеличение доли электронного документооборота.

## **6.3 Безопасность в чрезвычайных ситуациях**

### **6.3.1 Перечень возможных чрезвычайных ситуаций на объекте**

Представим список чрезвычайных ситуаций, которые могут возникнуть при работе в офисе:

В зависимости от намерения.

- Преднамеренные.

- Непреднамеренные.

По характеру происхождения.

- Техногенные: пожары, взрывы, нарушение целостности помещений, повреждение или выход из строя систем жизнеобеспечения. Данные причины также могут возникнуть под воздействием сил природы.
- Экологические – это непредвиденные или аномальные изменения состояния окружающей природной среды (ураганы, ливни, паводки, наводнения и паводки). Нарушения в экологической системе может возникать по антропогенным причинам.
- Биологические – эпидемии различных болезней.
- Комбинированные.

В зависимости от скорости развития.

- Плавные.
- Скоротечные.
- Взрывные.
- Внезапные.

По масштабам последствий.

- Территориальные.
- Местные.
- Локальные.

По возможности предотвращения.

- Предотвращаемые.
- Неизбежные.

Наиболее распространённой формой чрезвычайной ситуации в офисном здании является возникновение пожара. Требования по пожарной безопасности устанавливаются ГОСТ 12.1.004-91 [53].



### 6.3.2 Меры по ликвидации чрезвычайных ситуаций и последствий

Пожар – это неконтролируемый процесс горения за пределами специализированного очага, наносящий вред жизни и здоровью людей, их имуществу, интересам общества и государства.

Основные причины возникновения пожара:

- нарушение требований пожарной безопасности при эксплуатации электроприборов;
- включение электроприбора в неисправную розетку;
- нарушение правил проведения электрогазосварочных и огневых работ;
- случайный или умышленный поджог;
- пренебрежение опасностью, незнание и недооценка возможных последствий пожара;
- курение в помещении.

Противопожарная защита должна достигаться применением одного из следующих способов или их комбинацией:

- применением средств пожаротушения и соответствующих видов пожарной техники;
- применением автоматических установок пожарной сигнализации и пожаротушения;
- применением основных строительных конструкций и материалов, в том числе используемых для облицовок конструкций, с нормированными показателями пожарной опасности;
- применением пропитки конструкций объектов антипиренами и нанесением на их поверхности огнезащитных красок (составов);
- устройствами, обеспечивающими ограничение распространения пожара;
- организацией с помощью технических средств, включая автоматические, своевременного оповещения и эвакуации людей;

- применением средств коллективной и индивидуальной защиты людей от опасных факторов пожара;
- применением средств противодымной защиты.

## **6.4 Правовые и организационные вопросы обеспечения безопасности**

### **6.4.1 Психофизиологические факторы**

К таким факторам относятся чрезмерное умственное напряжение, статические физические перегрузки опорно-двигательной системы человека, вызванные монотонностью труда, а также перегрузки эмоционального характера. Работнику, в течение каждого рабочего дня необходимо предоставлять перерыв для приема пищи и отдыха. Продолжительность перерыва 30-120 минут. Конкретная величина перерыва устанавливается либо правилами трудового распорядка, либо индивидуальной договоренностью разработчика с руководством. Чтобы сохранить на должном уровне производительность труда и избежать преждевременной утомляемости разработчика рекомендуется чередовать типы работ и включать такие типы, в которых персональный компьютер не используется вовсе. Также во время перерывов рекомендуется выполнять комплекс физических упражнений.

### **6.4.2 Организационные рабочего места**

Рабочие места разработчиков, работающих на персональных компьютерах, рекомендуется размещать изолированно друг от друга, так как данный вид работы требует высокой концентрации внимания и значительного умственного напряжения. Расположение экрана монитора должно быть на расстоянии 600 - 700 мм от глаз пользователя. Рекомендуется не приближаться ближе чем 500 мм.

Ввиду сидячего характера работы разработчика важнейшим элементом его рабочего места становится кресло. Его конструкция должна поддерживать рациональную рабочую позу при работе на персональном компьютере, а также позволять изменять позу, чтобы снизить статическое напряжения мышц шейно-плечевой области и спины.

Поверхность рабочего стола должна иметь коэффициент отражения 0,5-0,7.

Немаловажным фактором является учет требований к размещению средств отображения информации, таблица 26.

Таблица 26 – Расположение средств отображения информации

Тип средств отображения информации	Угол, градусы	
	В горизонтальной плоскости	В вертикальной плоскости
Редко используемые	±60	±60
Редко используемые	±30	±30
Очень часто используемые	±15	±15

## ЗАКЛЮЧЕНИЕ

Результатом работы является мобильное приложения-путеводитель по городу Томску.

В процессе выполнения данной работе, был проведён анализ и выявлены необходимые требования, которыми должно обладать мобильное приложение-путеводитель. На основе данных требования было создано мобильное приложение «Путеводитель по Томску» позволяющее иметь под рукой информацию о ближайших мероприятиях, достопримечательностях и интересных маршрутах города Томска.

Данное приложение отличается от конкурирующих решений возможностью обновления туристической информации без необходимости переустановки приложения.

Мобильное приложение позволяет обмениваться впечатлениями и делиться эмоциями оставляя комментарии к конкретному месту или маршруту, оценивать на сколько посещённая достопримечательность или пройденный маршрут интересен. Приложении позволяет делиться информацией с другими пользователями в социальных сетях с помощью возможности опубликовать информацию о выбранной достопримечательности или выбранном маршруте.

Функциональность приложения может быть расширена добавлением авторизации пользователей через социальные сети и возможностью составления собственных маршрутов.

## CONCLUSION

This work resulted in mobile travel guide application for Tomsk city.

During development process as a result of performed analysis necessary requirements for a travel guide application were deducted. Based on those requirements an application “Tomsk travel guide” was created that allows easy access to information about upcoming events, main sights and interesting routes around Tomsk.

Main difference of this application with competitors is opportunity to refresh tourist information without reinstalling application.

This application allows users to share their impressions and emotions with others by leaving comments to specific place or route and to rate visited sights and routes. It also encourages sharing impressions about favorite sights and routes via social networks.

Application functionality can be extended with social networks authorization and ability to construct custom routes.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Абдулхакова Айслу, Бородина Светлана Путеводитель как средство отражения и продвижения туристского объекта. Мир искусств // [Электронный ресурс]: Вестник Международного института антиквариата. 2016. №1 (13). URL: <http://cyberleninka.ru/article/n/putevoditel-kak-sredstvo-otrazheniya-i-prodvizheniya-turistskogo-obekta> (дата обращения: 09.06.2017).
2. ПУТЕВОДИТЕЛЬ ... что такое ПУТЕВОДИТЕЛЬ // [Электронный ресурс]: Большой Энциклопедический Словарь. URL: <https://www.vedu.ru/bigencdic/51424/> (дата обращения: 09.06.2017).
3. Travel-Tomsk - Туризм в Томске и Томской области // [Электронный ресурс]: Туристский портал Томска и Томской области. URL: <http://travel-tomsk.ru/> (дата обращения: 09.06.2017).
4. Мобильный web сайт или мобильное приложение? // [Электронный ресурс]: Хабрахабр. URL: <https://habrahabr.ru/post/168843/> (дата обращения: 09.06.2017).
5. UML-схемы вариантов использования: справочные материалы // [Электронный ресурс]: MSDN. URL: <https://msdn.microsoft.com/ru-ru/library/dd409427.aspx> (дата обращения: 09.06.2017).
6. Моделирование на UML. Диаграммы использования // [Электронный ресурс]: Моделирование на UML. URL: [http://book.uml3.ru/sec\\_2\\_2.html](http://book.uml3.ru/sec_2_2.html) (дата обращения: 09.06.2017).
7. Chapter 1: Software Requirements - SWEБОК // [Электронный ресурс]: SWEБОК. URL: [http://swebokwiki.org/Chapter\\_1:\\_Software\\_Requirements#Functional\\_and\\_Nonfunctional\\_Requirements](http://swebokwiki.org/Chapter_1:_Software_Requirements#Functional_and_Nonfunctional_Requirements) (дата обращения: 09.06.2017).
8. Domain Driven Design Community // [Электронный ресурс]: Domain Driven Design Community. URL: <http://dddcommunity.org> (дата обращения: 09.06.2017).

9. What is Domain-Driven Design? // [Электронный ресурс]: Domain Driven Design Community. URL: [http://dddcommunity.org/learning-ddd/what\\_is\\_ddd/](http://dddcommunity.org/learning-ddd/what_is_ddd/) (дата обращения: 09.06.2017).
10. Peter Pin-Shan Chen «The Entity-Relationship Model — Toward a Unified View of Data» (англ.) // ACM Transactions on Database Systems (TODS): Сб. — Нью-Йорк: ACM, 1976. — Vol. 1. — P. 9-36.
11. Схемы компонентов UML: справочные материалы // [Электронный ресурс]: MSDN. URL: <https://msdn.microsoft.com/ru-ru/library/dd409390.aspx> (дата обращения: 09.06.2017).
12. Моделирование на UML. Общие диаграммы // [Электронный ресурс]: Моделирование на UML. URL: [http://book.uml3.ru/sec\\_1\\_5#p7](http://book.uml3.ru/sec_1_5#p7) (дата обращения: 09.06.2017).
13. OASIS SOA Reference Model TC | OASIS // [Электронный ресурс]: Advancing open standards for the information society. URL: [https://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=soa-rm](https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm) (дата обращения: 09.06.2017).
14. Баночкин П.И. Методы и средства проектирования информационных систем и технологий: учебное пособие / П.И. Баночкин; Томский политехнический университет. — Томск: Изд-во Томского политехнического университета, 2015. — 80 с.
15. Васвани В. Zend Framework: разработка веб-приложений на PHP / В. Васвани; Библиотека программиста, 2011. — 432 с.
16. ASP.NET Web API | Microsoft Docs // [Электронный ресурс]: MSDN. URL: <https://docs.microsoft.com/en-us/aspnet/web-api/> (дата обращения: 09.06.2017).
17. PHP vs Ruby vs Python: The Three Programming Languages in a Nutshell // [Электронный ресурс]: Helping You Build a Better Web. URL: <https://1stwebdesigner.com/php-vs-ruby-vs-python/> (дата обращения: 09.06.2017).

18. Python Advocacy In Scientific Computation - Python Wiki // [Электронный ресурс]: Python Software Foundation Wiki Server. URL: <https://wiki.python.org/moin/PythonAdvocacyInScientificComputation> (дата обращения: 09.06.2017).
19. High Performance Development with Python // [Электронный ресурс]: Informatics, HPC & IT Solutions. URL: <https://www.scientificcomputing.com/article/2008/11/high-performance-development-python> (дата обращения: 09.06.2017).
20. 32/64 ODBC drivers for Linux, Mac OS and Windows to access databases // [Электронный ресурс]: Devart. URL: <https://www.devart.com/odbc/> (дата обращения: 09.06.2017).
21. MAMP & MAMP PRO // [Электронный ресурс]: MAMP. URL: <https://www.mamp.info/en/> (дата обращения: 09.06.2017).
22. Mobile operating system market share in Russian Federation | StatCounter Global Stats // [Электронный ресурс]: StatCounter Global Stats. URL: <http://gs.statcounter.com/os-market-share/mobile/russian-federation/#yearly-2017-2017-bar> (дата обращения: 09.06.2017).
23. iPhone vs Android: 5 reasons to pick the iPhone (and iOS) // [Электронный ресурс]: Macworld UK. URL: <http://www.macworld.co.uk/feature/iphone/iphone-vs-android-5-reasons-pick-iphone-ios-3454817/> (дата обращения: 09.06.2017).
24. 2015 Mobile Threat Report // [Электронный ресурс]: Pulse Secure Mobile Threat Center. URL: <https://www.pulsesecure.net/lp/mobile-threat-report-2014/> (дата обращения: 09.06.2017).
25. Fielding Dissertation: CHAPTER 5: Representational State Transfer (REST) // [Электронный ресурс]: University of California, Irvine. URL: [https://www.ics.uci.edu/~fielding/pubs/dissertation/rest\\_arch\\_style.htm](https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm) (дата обращения: 09.06.2017).



- 26.RFC 4122 - A Universally Unique IDentifier (UUID) URN Namespace // [Электронный ресурс]: IETF Tools. URL: <https://tools.ietf.org/html/rfc4122> (дата обращения: 09.06.2017).
- 27.RFC 3174 - US Secure Hash Algorithm 1 (SHA1) // [Электронный ресурс]: IETF Tools. URL: <https://tools.ietf.org/html/rfc3174> (дата обращения: 09.06.2017).
- 28.CRC Series, Part 1: Additive Checksums// [Электронный ресурс]: Barr Group. URL: <https://barrgroup.com/Embedded-Systems/How-To/Additive-Checksums> (дата обращения: 09.06.2017).
- 29.Introduction to Model/View/ViewModel pattern for building WPF apps &#8211; Tales from the Smart Client // [Электронный ресурс]: MSDN. URL: <https://blogs.msdn.microsoft.com/johngossman/2005/10/08/introduction-to-modelviewviewmodel-pattern-for-building-wpf-apps/> (дата обращения: 09.06.2017).
- 30.UIKit // [Электронный ресурс]: Apple Developer Documentation. URL: <https://developer.apple.com/reference/foundation/uikit> (дата обращения: 09.06.2017).
- 31.UITableViewController – UIKit // [Электронный ресурс]: Apple Developer Documentation. URL: <https://developer.apple.com/reference/foundation/uitableviewController> (дата обращения: 09.06.2017).
- 32.NSURLConnection – Foundation // [Электронный ресурс]: Apple Developer Documentation. URL: <https://developer.apple.com/reference/foundation/nsURLConnection> (дата обращения: 09.06.2017).
- 33.JSONSerialization - Foundation // [Электронный ресурс]: Apple Developer Documentation. URL: <https://developer.apple.com/reference/foundation/jsonserialization> (дата обращения: 09.06.2017).

34. JSONSerialization - Foundation // [Электронный ресурс]: Apple Developer Documentation. URL: <https://developer.apple.com/reference/foundation/jsonserialization> (дата обращения: 09.06.2017).
35. Moya: Network abstraction layer written in Swift // [Электронный ресурс]: GitHub. URL: <https://github.com/Moya/Moya> (дата обращения: 09.06.2017).
36. SwiftyJSON: The better way to deal with JSON data in Swift // [Электронный ресурс]: GitHub. URL: <https://github.com/SwiftyJSON/SwiftyJSON> (дата обращения: 09.06.2017).
37. Kingfisher: A lightweight, pure-Swift library for downloading and caching images from the web // [Электронный ресурс]: GitHub. URL: <https://github.com/onevcat/Kingfisher> (дата обращения: 09.06.2017).
38. SwiftKeychainWrapper: A simple wrapper for the iOS Keychain to allow you to use it in a similar fashion to User Defaults. Written in Swift // [Электронный ресурс]: GitHub. URL: <https://github.com/jrendel/SwiftKeychainWrapper> (дата обращения: 09.06.2017).
39. CryptoSwift: CryptoSwift is a growing collection of standard and secure cryptographic algorithms implemented in Swift // [Электронный ресурс]: GitHub. URL: <https://github.com/krzyzanowskim/CryptoSwift> (дата обращения: 09.06.2017).
40. MapKit // [Электронный ресурс]: Apple Developer Documentation. URL: <https://developer.apple.com/reference/mapkit> (дата обращения: 09.06.2017).
41. Core Location // [Электронный ресурс]: Apple Developer Documentation. URL: <https://developer.apple.com/reference/corelocation> (дата обращения: 09.06.2017).
42. UML 2 Deployment Diagrams: An Agile Introduction // [Электронный ресурс]: Apple Developer Documentation. URL:

- <http://www.agilemodeling.com/artifacts/deploymentDiagram.htm> (дата обращения: 09.06.2017).
43. Постановление от 13 июня 2003 года №118 «О введении в действие санитарно-эпидемиологических правил и нормативов СанПиН 2.2.2/2.4.1340-03» (с изменениями на 21 июня 2016 года). [Электронный ресурс]: Кодексы РФ, СП, ГОСТ, СНИП, Санпин, регламенты. URL: <http://docs.cntd.ru/document/901865498> (дата обращения: 09.06.2017).
  44. Санитарно-эпидемиологические правила и нормативы: СанПиН 2.2.4.3359-16 Санитарно-эпидемиологические требования к физическим факторам на рабочих местах. – М.: Минздрав России, 2016. – 69 с.
  45. Санитарно-эпидемиологические правила и нормативы: СанПиН 2.2.4.548-96 Гигиенические требования к микроклимату производственных помещений. – М.: Минздрав России, 1997. – 20 с.
  46. Свод правил: СП 52.13330.2011 Естественное и искусственное освещение. М.: Минрегион России, 2011. – 74 с.
  47. Санитарные нормы: СН 2.2.4/2.1.8.562-96 Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки. – М.: Минздрав России, 1996. – 8 с.
  48. ГОСТ 12.0.003-2015 ССБТ. Опасные и вредные производственные факторы. Классификация. – М.: Изд-во стандартов, 2016. – 16 с.
  49. СанПиН 2.2.4.1191–03. Электромагнитные поля в производственных условиях.
  50. ГОСТ 12.1.005-88. ССБТ. Общие санитарно-гигиенические требования к воздуху рабочей зоны.
  51. ГОСТ Р 55090-2012 Ресурсосбережение. Обращение с отходами. Рекомендации по утилизации отходов бумаги.
  52. СанПиН 2.1.7.1322-03 Гигиенические требования к размещению и обезвреживанию отходов производства и потребления.
  53. ГОСТ 12.1.004-91 ССБТ. Пожарная безопасность. Общие требования. – М.: Изд-во стандартов, 2006. – 67 с.

## ПРИЛОЖЕНИЕ А

Исходный код реализации веб-сервисов для работы с избранным.

```
<?php
/**
 * Модуль работы с избранным
 */

switch ($method):
case 'GET':
    if ($item_id == 0) {
        $query = $db->query("SELECT
                                `f`.`item_type`,
                                `f`.`item_id`
                                FROM
                                `favorites` AS `f`
                                WHERE
                                `user_id`=" . $user_id . "");
        if ($query->num_rows > 0) {
            while ($row = $query->fetch_assoc()) {
                $result[] = $row;
            }
        } else {
            /* Отдаем 404 */
            http_response_code(404);
        }
    } else {
        http_response_code(400);
    }
    break;
case 'PUT':
    /* Не используется в данном модуле */
    break;
case 'POST':
    // Если передан не поддерживаемый модуль, отдаем 404
    if (!in_array($module, ['events', 'places', 'routes'])) {
        http_response_code(404);
        exit;
    }

    if ($module != 'no_module' && $id != 0)
        $db-
>query("INSERT INTO `favorites` SET `item_type`=" . $module . ", `item_id`=" . $id . ",
```

```

`user_id`="" . $user_id . "");
    else
        http_response_code(400);
    break;
case 'DELETE':
    // Если передан не поддерживаемый модуль, отдаем 404
    if (!in_array($module, ['events', 'places', 'routes'])) {
        http_response_code(404);
        exit;
    }

    if ($module != 'no_module' && $id != 0) {
        $db-
>query("DELETE FROM `favorites` WHERE `item_type`="" . $module . "" AND `item_id`
="" . $id . "" AND `user_id`="" . $user_id . "");
    } else
        http_response_code(400);
    break;
endswitch;

```