

Школа Инженерная школа информационных технологий и робототехники
 Направление подготовки 09.04.02. Информационные системы и технологии
 Отделение школы (НОЦ) Отделение информационных технологий

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы Разработка мобильного приложения для AR-навигации на базе открытых картографических данных

УДК 004.451:656.6:528.8

Студент

Группа	ФИО	Подпись	Дата
8ИМ01	Сергиенко Александр Романович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Демин А.Ю.	К.Т.Н.		

Консультант

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ст. преп. ОИТ	Лоскутов В.В.			

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент ОСГН, ШБИП	Былкова Т.В.	К.Э.Н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор ООД ШБИП	Федоренко О.Ю.	Д.М.Н.		

ДОПУСТИТЬ К ЗАЩИТЕ:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Савельев А.О.	К.Т.Н.		

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

Код компетенции	Наименование компетенции
	Универсальные компетенции
УК(У)-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, выработать стратегию действий
УК(У)-2	Способен управлять проектом на всех этапах его жизненного цикла
УК(У)-3	Способен организовывать и руководить работой команды, выработывая командную стратегию для достижения поставленной цели
УК(У)-4	Способен применять современные коммуникативные технологии, в том числе на иностранном(-ых) языке(-ах), для академического и профессионального взаимодействия
УК(У)-5	Способен анализировать и учитывать разнообразие культур в процессе межкультурного взаимодействия
УК(У)-6	Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки
	Общепрофессиональные компетенции
ОПК(У)-1	Способен самостоятельно приобретать, развивать и применять математические, естественно-научные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте
ОПК(У)-2	Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач
ОПК(У)-3	Способен анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями
ОПК(У)-4	Способен применять на практике новые научные принципы и методы исследований
ОПК(У)-5	Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем
ОПК(У)-6	Способен использовать методы и средства системной инженерии в области получения, передачи, хранения, переработки и представления информации посредством

	информационных технологий
ОПК(У)-7	Способен разрабатывать и применять математические модели процессов и объектов при решении задач анализа и синтеза распределенных информационных систем и систем поддержки принятия решений
ОПК(У)-8	Способен осуществлять эффективное управление разработкой программных средств и проектов
	Профессиональные компетенции
ПК(У)-1	Способен управлять программно-техническими, технологическими и человеческими ресурсами
ПК(У)-2	Способен управлять развитием баз данных
ПК(У)-3	Способен управлять работами по сопровождению и проектами создания (модификации) информационных систем, автоматизирующих задачи организационного управления и бизнес-процессы.
ПК(У)-4	Способен проектировать и организовывать учебный процесс по образовательным программам с использованием современных образовательных технологий.
ПК(У)-5	Способен осуществлять руководство разработкой комплексных проектов на всех стадиях и этапах выполнения работ.

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

Школа Инженерная школа информационных технологий и робототехники
 Направление подготовки 09.04.02. Информационные системы и технологии
 Отделение школы (НОЦ) Отделение информационных технологий

УТВЕРЖДАЮ:
 Руководитель ООП
 _____ Савельев А.О.
 (Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

В форме:

Магистерской диссертации <small>(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)</small>

Студенту:

Группа	ФИО
8ИМ01	Сергиенко Александру Романовичу

Тема работы:

Разработка мобильного приложения для AR-навигации на базе открытых картографических данных	
Утверждена приказом директора (дата, номер)	№103-16/с от 13.04.2022 г.

Срок сдачи студентом выполненной работы:

--

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	<ul style="list-style-type: none"> – План работы – Задание на разработку мобильного приложения
Перечень подлежащих исследованию, проектированию и разработке вопросов	<ul style="list-style-type: none"> – Исследовать предметную область и имеющиеся разработки в данной сфере – Проанализировать и определить требования к программному продукту – Выбрать проектные решения и инструменты для реализации приложения – Разработать архитектуру программного обеспечения с учетом требований – Апробация приложения
Перечень графического материала	<ul style="list-style-type: none"> – Презентация для защиты – Интерфейс мобильного приложения
Консультанты по разделам выпускной квалификационной работы	
Раздел	Консультант
Финансовый менеджмент,	Былкова Т. В.

ресурсоэффективность и энергосбережение	
Социальная ответственность	Федоренко О. Ю.
Раздел на иностранном языке	Пичугова И. Л.
Названия разделов, которые должны быть написаны на иностранном языке:	
Обзор имеющихся навигационных приложений и средств разработки дополненной реальности	

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	
--	--

Задание выдал руководитель / консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Демин А. Ю.	к.т.н.		
Ст. преп. ОИТ	Лоскутов В.В.			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ01	Сергиенко Александр Романович		

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

Студенту:

Группа	ФИО
8ИМ01	Сергиенко Александру Романовичу

Школа	ИШИТР	Отделение школы (НОЦ)	ОИТ
Уровень образования	магистратура	Направление/специальность	09.04.02. Информационные системы и технологии

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 (руководитель и студент-дипломник).</i>
2. <i>Нормы и нормативы расходования ресурсов</i>	<i>Тариф на электроэнергию – 3,85 руб. за 1 кВт·ч. коэффициент накладных расходов – 16%</i>
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	<i>Коэффициент отчислений во внебюджетные фонды – 30 %,</i>

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. <i>Оценка коммерческого и инновационного потенциала НТИ</i>	<i>Произвести предпроектный анализ.</i>
2. <i>Разработка устава научно-технического проекта</i>	<i>Разработать Устав научного проекта магистерской работы</i>
3. <i>Планирование процесса управления НТИ: структура и график проведения, бюджет, риски и организация закупок</i>	<i>Разработать план управления НТИ</i>
4. <i>Определение ресурсной, финансовой, экономической эффективности</i>	<i>Провести оценку экономической эффективности разработки мобильного приложения</i>

Перечень графического материала (с точным указанием обязательных чертежей):

1. *Оценка конкурентоспособности технических решений*
2. *Матрица SWOT*
3. *График проведения и бюджет НТИ*
4. *Оценка ресурсной, финансовой эффективности НТИ*

Дата выдачи задания для раздела по линейному графику	31.01.2022
---	------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент ОСГН, ШБИП	Былкова Татьяна Васильевна	к.э.н.		31.01.2022

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ01	Сергиенко Александр Романович		31.01.2022

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Студенту:

Группа		ФИО	
8ИМ01		Сергиенко Александру Романовичу	
Школа	ИШИТР	Отделение (НОЦ)	ОИТ
Уровень образования	магистратура	Направление/специальность	09.04.02. Информационные системы и технологии

Тема ВКР:

Разработка мобильного приложения для AR-навигации на базе открытых картографических данных

Исходные данные к разделу «Социальная ответственность»:

<p>Введение</p> <ul style="list-style-type: none"> – Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика) и области его применения. – Описание рабочей зоны (рабочего места) при разработке проектного решения 	<p><i>Объект исследования: мобильное приложение для навигации</i> <i>Область применения: навигация в пределах организации</i> <i>Рабочая зона: офис</i> <i>Площадь помещения: 32 м²</i> <i>Описание помещения: офисное отапливаемое помещение с вытяжной системой вентиляции и искусственным освещением, обеспечивающим достаточный уровень освещенности</i> <i>Количество и наименование оборудования рабочей зоны: 6 рабочих мест, каждое из которых включает в себя стул, компьютер с периферийными устройствами, расположенными на столе</i> <i>Рабочие процессы, связанные с объектом исследования, осуществляющиеся в рабочей зоне: разработка приложения в среде Unity с использованием редактора Visual Studio на компьютере, тестирование приложения на мобильных устройствах</i></p>
---	---

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

<p>1. Правовые и организационные вопросы обеспечения безопасности при разработке проектного решения:</p> <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	<ul style="list-style-type: none"> – Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 09.03.2021) – ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования – ГОСТ 12.2.061-81 Система стандартов безопасности труда (ССБТ). Оборудование производственное. Общие требования безопасности к рабочим местам – СП 2.2.3670-20 Санитарно-эпидемиологические требования к условиям труда – ГОСТ 12.0.003-2015 Опасные и вредные производственные факторы. Классификация. Перечень опасных и вредных факторов – СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания – ГОСТ 12.1.012-2004 Система стандартов безопасности труда (ССБТ). Вибрационная
--	--

	<p>безопасность. Общие требования</p> <ul style="list-style-type: none"> – ГОСТ Р 50923-96. Дисплей. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения – СП 52.13330.2016 Естественное и искусственное освещение. Актуализированная редакция СНиП 23-05-95 – ГОСТ Р 50948-2001 Средства отображения информации индивидуального пользователя. Общие эргономические требования безопасности – ГОСТ 12.1.038-82 Система стандартов безопасности труда (ССБТ). Электробезопасность. Предельно допустимые значения напряжений прикосновения и токов – ГОСТ Р 51768-2001 Ресурсосбережение. Обращение с отходами. Методика определения ртути в ртутьсодержащих отходах – ГОСТ Р 22.3.03-94 Безопасность в чрезвычайных ситуациях. Защита населения. Основные положения – ГОСТ 12.1.004-91 Система стандартов безопасности труда (ССБТ). Пожарная безопасность. Общие требования
<p>2. Производственная безопасность при разработке проектного решения:</p> <ul style="list-style-type: none"> – Анализ выявленных вредных и опасных производственных факторов – Расчет уровня опасного или вредного производственного фактора 	<p>Вредные факторы:</p> <ul style="list-style-type: none"> – Производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего. – Повышенный уровень локальной вибрации. – Отсутствие или недостаток искусственного освещения. – Монотонность труда, вызывающая монотонию. – Длительное сосредоточенное наблюдение. <p>Опасные производственные факторы:</p> <ul style="list-style-type: none"> – Производственные факторы, связанные с электрическим током, вызываемым разницей электрических потенциалов, под действие которого попадает работающий. <p>Требуемые средства коллективной и индивидуальной защиты от выявленных факторов: системы вентиляции, источники света, защитные покрытия, защитные заземления, виброизолирующие средства, изолирующие устройства и покрытия.</p>

	Расчет: расчет системы воздухообмена.
3. Экологическая безопасность при разработке проектного решения	Воздействие на литосферу: утилизация отходов, связанных с выходом из строя ПК и периферийных устройств (принтеры, МФУ, веб-камеры, наушники, колонки, телефоны); люминесцентных ламп; утилизация израсходованных канцелярских принадлежностей
4. Безопасность в чрезвычайных ситуациях при разработке проектного решения	Возможные ЧС: пожар, взрыв, землетрясение Наиболее типичная ЧС: пожар
Дата выдачи задания для раздела по линейному графику	

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор ООД ШБИП	Федоренко Ольга Юрьевна	д.м.н.		14.03.2022

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ01	Сергиенко Александр Романович		14.03.2022

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа Инженерная школа информационных технологий и робототехники
 Направление подготовки (специальность) 09.04.02. Информационные системы и технологии
 Уровень образования Магистратура
 Отделение школы (НОЦ) Отделение информационных технологий
 Период выполнения весенний семестр 2021/2022 учебного года

Форма представления работы:

Магистерская диссертация

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

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	
--	--

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
10.02.2022	<i>Составление и утверждение задания</i>	5
25.02.2022	<i>Анализ предметной области</i>	10
02.03.2022	<i>Разработка календарного плана</i>	5
10.03.2022	<i>Выбор программного обеспечения, библиотек, платформы разработки</i>	5
25.03.2022	<i>Разработка интерфейса режима дополненной реальности с добавлением объектов</i>	15
12.04.2022	<i>Разработка картографического режима для навигации</i>	15
22.04.2022	<i>Добавление функционала по определению и записи геокординат создаваемых объектов</i>	5
05.05.2022	<i>Реализация воссоздания ранее созданных объектов по сохраненным геокординатам.</i>	10
15.05.2022	<i>Подключение сервера для хранения, получения новых записей и передачи имеющихся данных</i>	10
22.05.2022	<i>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</i>	5
28.05.2022	<i>Социальная ответственность</i>	5
05.06.2022	<i>Составление пояснительной записки</i>	10

СОСТАВИЛ:

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Демин А.Ю.	к.т.н.		

Консультант

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ст. преп. ОИТ	Лоскутов В.В.			

СОГЛАСОВАНО:**Руководитель ООП**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Савельев А.О.	К.Т.Н.		

РЕФЕРАТ

Выпускная квалификационная работа 155 страниц, 46 рисунков, 30 таблиц, 29 источников, 8 приложений.

Ключевые слова: дополненная реальность, навигация, Android, мобильное приложение, C#.

Объектом исследования является разрабатываемое программное приложение для мобильных устройств.

Цель работы – разработать программное обеспечение (ПО), которое позволит создавать систему навигационных объектов в дополненной реальности для ориентирования в пределах ТПУ.

В процессе исследования был проведен анализ имеющихся мобильных приложений в области навигации, средств для разработки приложений с дополненной реальностью и существующих приложений с возможностью навигации с использованием дополненной реальности. Также были выдвинуты требования к мобильному приложению и были подобраны средства для его разработки.

Функциональные возможности разработанного приложения разделены на два режима. В AR режиме у пользователя имеется возможность создания нового объекта и занесения в него текстовой информации. Также в данном режиме воссоздаются сохраненные объекты по их геокординатам. В картографическом режиме имеющиеся объекты добавляются на карте, отображая сохраненную в них текстовую информацию.

Экономическая эффективность/значимость работы: обусловлена возможностью привлечения большего количества абитуриентов и студентов ТПУ с помощью разрабатываемого приложения.

Функциональность приложения может быть расширена при интеграции мобильного приложения с сервисами ТПУ. Также в дальнейшем планируется дополнить разрабатываемое приложение путем внедрения сервисов по уточнению местоположения пользователя.

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ И НОРМАТИВНЫЕ ССЫЛКИ

AR – Augmented Reality

VR – Virtual Reality

API – Application Programming Interface

YOURS – Yet another OpenStreetMap Route Service

OSRM – Open Source Routing Machine

MVC – Model-View-Controller

SDK – Software Development Kit

JSON – JavaScript Object Notation

REST – Representational State Transfer

ПО – Программное обеспечение

ОС – Операционная система

Содержание

Реферат	12
Определения, обозначения, сокращения и нормативные ссылки	13
Введение	17
1 Обзор имеющихся навигационных приложений и средств разработки дополненной реальности	19
1.1 Анализ существующих навигационных приложений	19
1.2 Сравнение средств разработки AR приложений и их возможностей	29
1.3 Обзор навигационных приложений с возможностями дополненной реальности	35
2 Требования к разрабатываемому приложению и подбор необходимых средств разработки	42
2.1 Общие требования к программному продукту	42
2.2 Выбор средств для разработки приложения	44
3 Проектирование приложения	47
3.1 Архитектура приложения	47
3.2 Алгоритм преобразования географических координат в локальные	49
3.3 Сохраняемые данные навигационных объектов	52
3.4 Серверная часть	53
3.5 Картографический режим	54
3.6 Режим дополненной реальности (AR-режим)	56
3.7 Переход между режимами приложения	59
4 Результаты разработки приложения	61
5 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	64
5.1 Предпроектный анализ	64
5.1.1 Потенциальные потребители результатов исследования	64
5.1.2 Анализ конкурентных технических решений с позиции ресурсоэффективности и ресурсосбережения	66
5.1.3 SWOT–анализ	68
	14

5.1.4	Оценка готовности проекта к коммерциализации	70
5.1.5	Методы коммерциализации результатов научно-технического исследования	72
5.2	Инициация проекта	73
5.2.1	Цели и результат проекта	73
5.2.2	Организационная структура проекта	74
5.3	Планирование управления научно-техническим проектом	74
5.3.1	Структура работ в рамках научного исследования	74
5.3.2	Определение трудоемкости выполнения работ	76
5.4	Бюджет научного исследования	80
5.4.1	Сырье, материалы, покупные изделия и полуфабрикаты (за вычетом отходов)	80
5.4.2	Расчет затрат на электроэнергию	81
5.4.3	Специальное оборудование для научных (экспериментальных) работ	82
5.4.4	Основная заработная плата	83
5.4.5	Отчисления на социальные нужды	86
5.4.6	Накладные расходы	86
5.4.7	Формирование бюджета затрат научно-исследовательского проекта	87
5.5	Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования	88
5.5.1	Оценка сравнительной эффективности исследования	88
5.5.2	Интегральный показатель эффективности разработки	90
6	Социальная ответственность	92
6.1	Правовые и организационные вопросы обеспечения безопасности	92
6.1.1	Специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства	93
6.1.2	Организационные мероприятия при компоновке рабочей зоны	93
6.2	Производственная безопасность при разработке проектного решения	95
6.2.1	Анализ выявленных вредных и опасных производственных факторов	95

6.2.2	Микроклимат рабочего помещения	96
6.2.3	Вибрация	98
6.2.4	Производственное освещение	99
6.2.5	Монотонность труда, вызывающая монотонию	100
6.2.6	Длительное сосредоточенное наблюдение	101
6.2.7	Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека	102
6.2.8	Расчет системы воздухообмена	103
6.3	Экологическая безопасность при разработке проектного решения	104
6.4	Безопасность в чрезвычайных ситуациях при разработке проектного решения	105
6.5	Заключение по разделу	107
	Заключение	108
	Список публикаций студента	109
	Список использованных источников	110
	Приложение А	113
	Приложение Б	115
	Приложение В	118
	Приложение Г	119
	Приложение Д	122
	Приложение Е	130
	Приложение Ж	133
	Приложение И	135

ВВЕДЕНИЕ

На сегодняшний день в век информационных технологий особо ценится своевременность передаваемой информации, частью которой является точное местоположение необходимое для ориентирования людей.

Использование AR технологий в данной области позволяет упростить процесс ориентирования ввиду того, что благодаря дополненной реальности навигация происходит по вспомогательным объектам, созданным поверх реального мира. Тем самым это исключает вероятность движения пользователя в неверном направлении и, в конечном счете, приведет его в нужное место. Также благодаря дополненной реальности появляется возможность передачи пользователю определенной информации, которая может быть как рекламного, так и просветительного характера. Таким образом, при использовании данной возможности можно повысить эффективность сервисов организации путем предоставления имеющейся в них информации пользователю в определенном месте.

Целью работы является разработка программного обеспечения (ПО), которое позволит создавать систему навигационных объектов в дополненной реальности для ориентирования в пределах ТПУ.

Разрабатываемое приложение направлено на улучшение учебного процесса и ориентирования студентов в рамках корпусов университета.

В результате исследования было разработано мобильное приложение, которое предоставляет возможность создания системы навигационных объектов в дополненной реальности для ориентирования в пределах ТПУ. Для реализации данной задачи была разработана система, осуществляющая перерасчет между геолокационными координатами и координатами AR-пространства. На основе данной системы навигационные объекты отображаются как в пространстве реального мира, так и на картографическом режиме, в котором на карте помимо самих навигационных объектов отображается и местоположение пользователя.

В рамках дальнейшего развития проекта возможна интеграция создаваемого ПО с сервисами ТПУ, в частности с расписанием учебных занятий в университете. Таким образом, при дальнейшем объединении с расписанием можно будет создать систему объектов, которые будут находиться напротив каждого кабинета, и в них будет отображаться обновляемая информация о текущем занятии в данной аудитории. Из данной информации легко можно будет узнать, какая дисциплина должна проходить в данной аудитории, кто должен ее проводить и для какой группы. А благодаря использованию картографического режима у студентов появится возможность быстрого нахождения необходимой им аудитории.

Актуальность работы заключается в отсутствии доступного решения, оптимизирующего навигацию абитуриентов и студентов по территории Томского политехнического университета, его корпусов и объектов.

1 ОБЗОР ИМЕЮЩИХСЯ НАВИГАЦИОННЫХ ПРИЛОЖЕНИЙ И СРЕДСТВ РАЗРАБОТКИ ДОПОЛНЕННОЙ РЕАЛЬНОСТИ

1.1 Анализ существующих навигационных приложений

Навигатор – это программное обеспечение позволяющее отображать картографические данные, по которым в режиме реального времени можно добраться из точки А в точку Б.

Все навигаторы преследуют одну цель — помочь пользователю сориентироваться на местности. Однако каждое приложение выполняет эту задачу по-разному — их функционал и возможности могут сильно отличаться. Далее будут представлены наиболее популярные навигационные приложения, отмечены их преимущества и недостатки, а также области их применения.

Яндекс.Навигатор

«Яндекс.Навигатор» — навигационное приложение, разработанное на технологической основе «Яндекс.Карт». Поддерживает схематическое и спутниковое отображение местности. Работает по принципу контекстности: показывает наиболее актуальную для пользователя информацию [1].

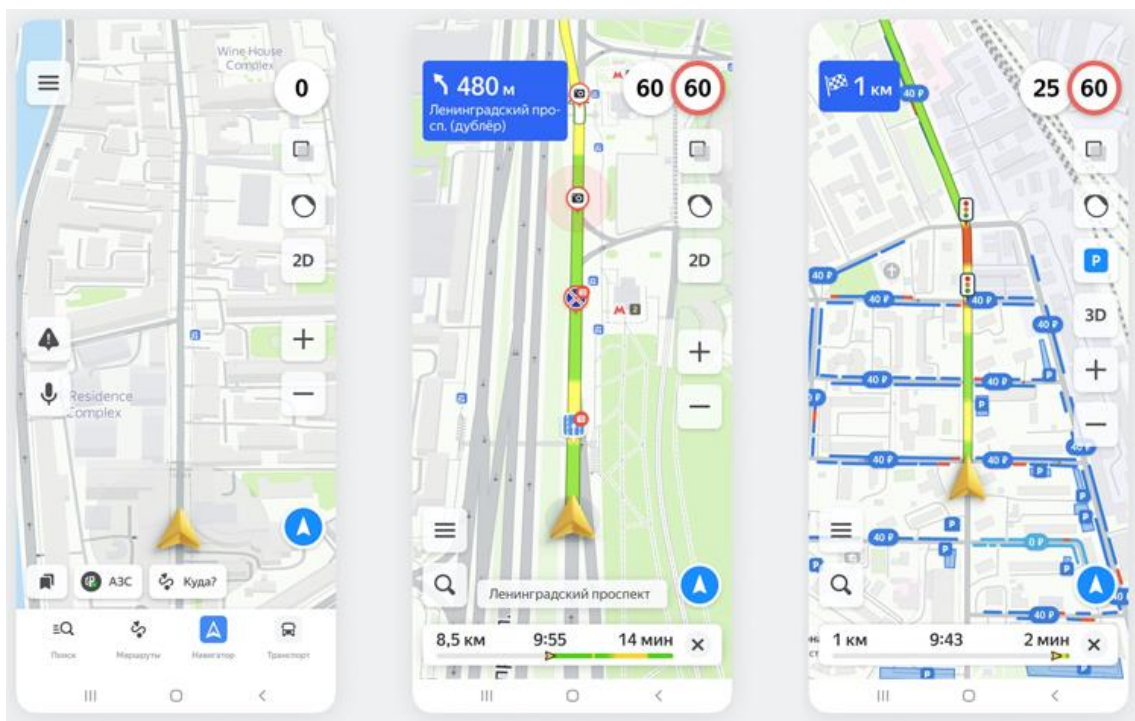


Рисунок 1 – Интерфейс приложения «Яндекс.Навигатор»

- Доступно для устройств на ОС Android и IOS.

- Бесплатное для пользователей, которые не относятся к юридическим лицам.
- Зона покрытия: Россия, Украина, Казахстан, Беларусь, Азербайджан, Армения, Грузия, Абхазия, Молдова, Узбекистан, Киргизия, Таджикистан, Турция.

Преимущества:

- Интуитивно понятный интерфейс, удобный в использовании.
- Построение маршрутов с учетом пробок и ДТП.
- Возможность загрузки маршрутов из истории пользователя.
- Функция голосового поиска для задания маршрута.
- Актуальная, своевременно обновляющаяся информация карт.
- Возможность работы приложения без подключения к сети.
- Навигатор помогает водителю в выборе полосы для маневра, перестроении.

Недостатки:

- Без интернета прокладка маршрутов не работает.
- Отсутствует информация о проездах во дворах.
- Отсутствуют оповещения о превышении скоростного режима.

Где лучше использовать:

Данное приложение оптимально подходит для ориентирования на территории России и стран СНГ, а также для использования на трассах и в крупных городах.

Google Maps

Google Maps являются общедоступным картографическим сервисом компании Google. Данные карты с опцией автомобильной навигации работают сегодня на мобильных устройствах с ОС Android и IOS. Также стоит отметить, что Google-карты могут быть использованы и в мультимедийных автомобильных системах, оснащенных Apple CarPlay и Android Auto. В приложение помимо карт имеются и спутниковые снимки Земли. Позволяет прокладывать сложные маршруты.

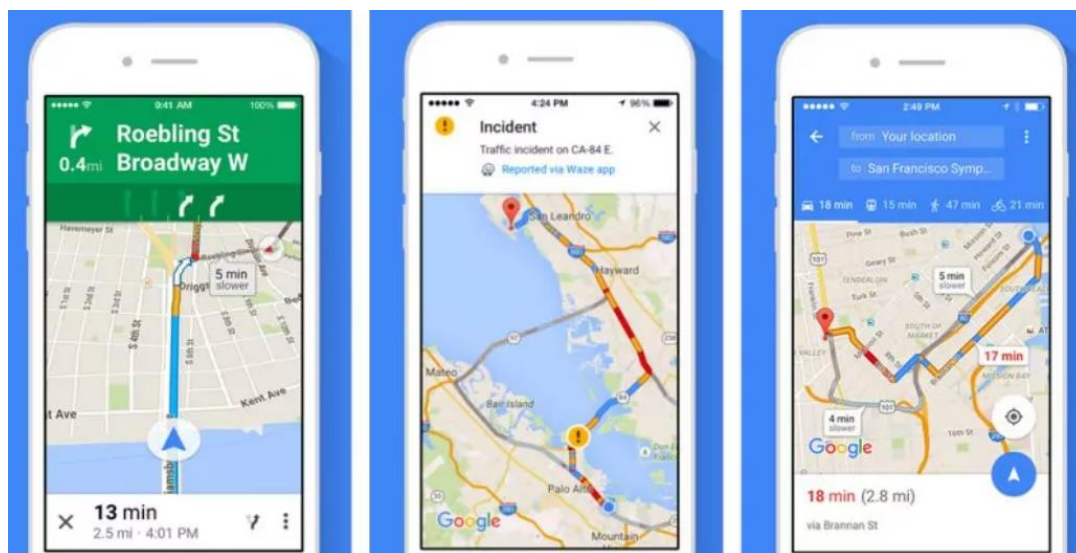


Рисунок 2 – Интерфейс приложения «Google Maps»

- Бесплатное приложение, по умолчанию устанавливается на устройства с ОС Android, также имеется возможность установки на устройства Apple.
- Работает на территории России, Великобритании, части Ирландии и других районов Европы, США, Канады, Японии, Китая, Гонконга.

Преимущества:

- Приложение считывает данные о пробках, ДТП, дорожных работах и своевременно перестраивает маршрут.
- Имеется возможность отображения расположения дорожных камер и предупреждает водителя об ограничениях скорости.
- Имеется встроенный голосовой помощник, который сопровождает движение подсказками.
- Отображается информация об объектах на карте.
- Возможность сохранения карт в память устройства и использования приложения без выхода в интернет.

Недостатки:

- Недостаточная точность и детализация карт и маршрутов на территории России и Европы.

Где лучше использовать:

Приложение оптимально для использования в автомобильных поездках за границу.

Maps.me

В 2010 году белорусский IT-разработчик и предприниматель Юрий Мельничек создал функциональную навигационную систему, способную работать и без доступа в интернет. Сегодня именно Maps.me называют главным конкурентом «Яндексу». Это связано с тем, какие возможности предоставляет Maps.me своему пользователю.

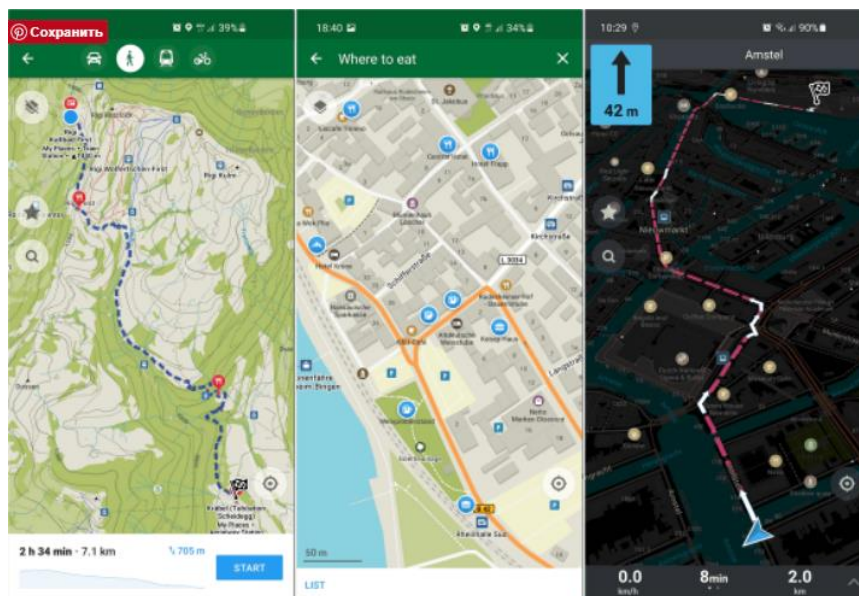


Рисунок 3 – Интерфейс приложения «Maps.me»

Приложение работает на основе свободной географической карты OpenStreetMap. Перед началом работы интересующие пользователя карты загружаются в память устройства и могут использоваться без подключения к интернету.

- Приложение может использоваться на мобильных устройствах с платформами iOS и Android.
- Бесплатное приложение, имеется Pro-версия с дополнительными возможностями.
- Зона покрытия: весь мир.

Преимущества:

- Возможность построения маршрутов для автомобилей и пешеходов в оффлайн-режиме.
- Возможность загрузки только необходимых карт городов и регионов.
- Высокая точность и подробность карт.
- Поиск по объектам, экспорт и импорт данных.
- Имеется возможность использования голосовой навигации.
- Постоянное обновление навигационной информации.
- Возможность передачи данных о своем местоположении другим пользователям (в Pro-версии).

Недостатки:

- Отсутствие информации, обновляемой в режиме реального времени.
- Требуется место в памяти устройства для загрузки карт.
- Максимальный функционал приложения только в платной версии.

Где лучше использовать:

Maps.me лучше всего подходит для заграничных поездок и любых других ситуаций, в которых подключение к интернету невозможно или ограничено.

OSMAND

OsmAnd — навигационное приложение, работающее на основе карт OpenStreetMap, «Яндекс.Карт» и «Яндекс.Пробок», Mapnik, Osmarender, Microsoft Maps, CloudMade и других сервисов. Приложение подразумевает предварительное скачивание карт интересующих регионов в память устройства. В связи с этим OsmAnd способен создать маршруты там, где отсутствует связь с интернетом. Приложение способно восстанавливать прошлые маршруты, выстраивать новые, а также позволяет ориентироваться в соседнем государстве.

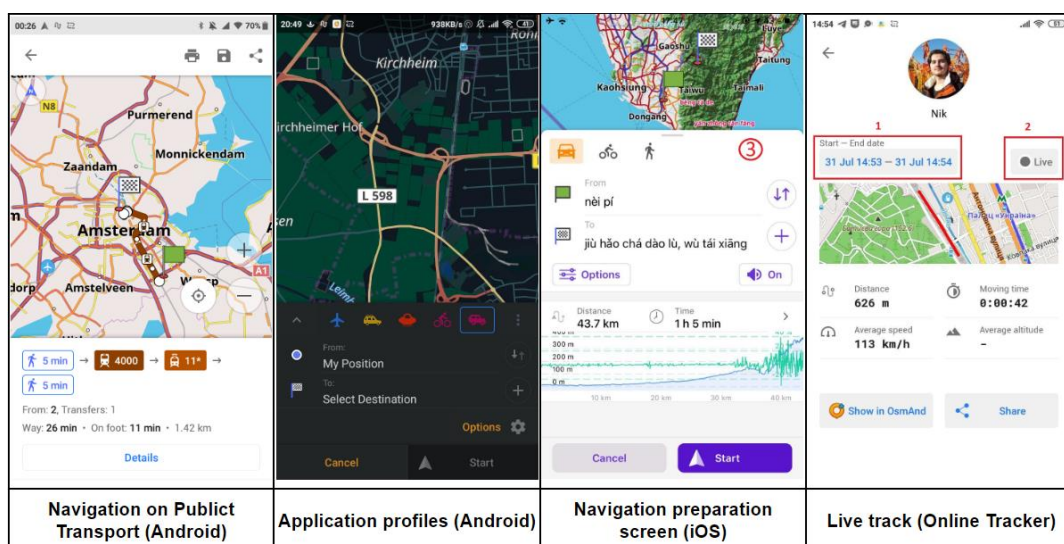


Рисунок 4 – Интерфейс приложения «OSMAND»

В данном приложении поддерживается онлайн-навигация благодаря использованию сервисов YOURS и OSRM. При наличии доступа в интернет маршрут будет перестраиваться с учетом пробок и множества других опций, которые можно задать в настройках. В приложении предусмотрена возможность кеширования данных для экономии интернет-трафика.

- Доступно для скачивания на устройствах Android и IOS.
- Имеется бесплатная и платная версии приложения.
- Зона покрытия: Россия, страны СНГ, дальше зарубежье.

Преимущества:

- Высокая детализация карт.
- Возможность сохранения пользовательского маршрута.
- Имеются голосовые подсказки.
- Автоматическое масштабирование в соответствии со скоростью движения пользователя.
- Наличие различных режимов использования приложения (автомобиль/пешеход/лыжи и т. д.).
- Возможность одновременного использования нескольких онлайн-карт.
- Поддержка дополнительных плагинов.

Недостатки:

- Имеется ограничение количества загрузок и обновлений карт в бесплатной версии.
- Обновление оффлайн-карт происходит только 1 раз в месяц.
- Требуется свободное место в памяти устройства.

Где лучше использовать:

OsmAnd лучше всего подходит для путешественников, использующих различные способы передвижения на территориях со слабым или отсутствующим интернет-соединением.

2ГИС

В 1999 году был создан отечественный проект, который имеет прямое отношение к навигации и картографии: 2ГИС, – навигационная система, в которой имеется справочная информация по городам и организациям в его зданиях. Данное приложение можно назвать электронным справочником, объединенным с картой города, который является бесплатным для пользователей и работает за счет рекламодателей компаний, которые предоставляют информацию о себе, чтобы она была отмечена в приложении.

Картография разработана на основе спутниковых снимков. Компания-разработчик гарантирует 95% точность карт. Приложение может работать в онлайн- и в оффлайн-режиме (посредством предварительной загрузки карт на устройство).

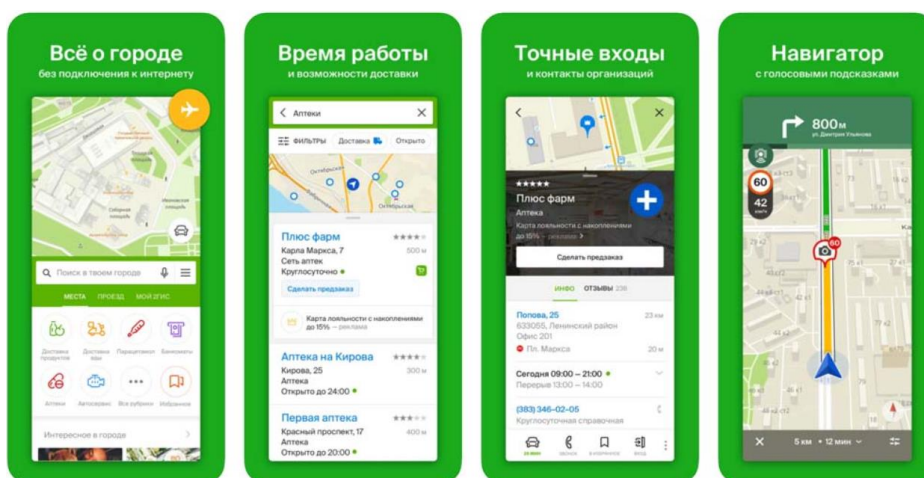


Рисунок 5 – Интерфейс приложения «2ГИС»

- Бесплатное приложение для Android и IOS.
- Зона покрытия: более 390 городов в России, Казахстане, Украине, Узбекистане, Азербайджане, Италии, Чехии, Чили, ОАЭ, Киргизии, на Кипре.

Преимущества:

- Отсутствие платной версии приложения.
- В приложении предусмотрены подсказки для пользователя в ходе его продвижения по маршруту, а также направляющие голосовые указания.
- Предусмотрена возможность автоматической корректировки маршрута (при отклонении от исходного).
- Имеется информация о маршрутах общественного транспорта.
- Простой и понятный интерфейс.
- Высокая скорость работы приложения.

Недостатки:

- Недостаточная детализация карт вне трасс и крупных населенных пунктов.
- Устанавливаемые карты занимают свободное место в памяти устройства.

Где лучше использовать:

Подходит для использования, прежде всего, в крупных городах — в частности, для деловых и рабочих целей.

Waze

Waze — социальное навигационное приложение для мобильных устройств. Данное приложение позволяет отслеживать дорожную ситуацию в режиме реального времени и прокладывать маршруты. Информация на карте обновляется самими пользователями.



Рисунок 6 – Интерфейс приложения «Waze»

- Бесплатное приложение.
- Для мобильных устройств на Android и IOS.

Преимущества:

- Достаточно высокая детализация карт России и ряда других стран.
- Предусмотрена возможность построения оптимального маршрута.
- Имеется информация о радарх скорости.
- В приложении реализована забота о пользователе (предупреждения о плохом покрытии, сложных маневрах и т. д.).
- Имеется возможность добавления своей информации на карте.
- Автоматический подсчет времени, необходимого для проезда пробки.

Недостатки:

- Сложный интерфейс приложения.
- Наличие белых пятен на карте, не лучшая детализация.

Где лучше использовать:

В городах, где приложение используется большим числом людей — для коммуникации с другими пользователями и активного участия в развитии локальной навигационной системы.

Для наглядного сравнения всех вышеперечисленных навигационных приложений по их основным возможностям составлена таблица 1.

Таблица 1 – Сравнение основных параметров приложений

Название приложения	Наличие бесплатной версии	Необходимость доступа в интернет	Зона покрытия	Наличие информации о пробках
Яндекс Навигатор	Есть	Предусмотрена возможность работы приложения как с подключением к интернету, так и без него	Россия, Украина, Казахстан, Беларусь, Азербайджан, Армения, Грузия, Абхазия, Молдова, Узбекистан, Киргизия, Таджикистан, Турция	Есть
Google Maps	Есть	Предусмотрена возможность работы приложения как с подключением к интернету, так и без него	Россия, Великобритания, часть районов Европы, США, Канада, Япония, Китай, Гонконг	Есть
Maps.me	Есть бесплатная и платная версии	После загрузки карт на устройство приложение работает в оффлайн-режиме	Весь мир	Нет
OsmAnd	Есть бесплатная и платная версии	Предусмотрена возможность работы приложения как с подключением к интернету, так и без него	Россия, страны СНГ, дальнее зарубежье	Есть
2ГИС	Есть	Предусмотрена возможность работы приложения как с подключением к интернету, так и без него	Более 390 городов в России, Казахстане, Украине, Узбекистане, Азербайджане, Италии, Чехии, Чили, ОАЭ, Киргизии, на Кипре	Есть
Waze	Есть	Необходимо подключение к интернету	Россия, Израиль, США, Канада, Великобритания, Франция, Германия, Италия, Нидерланды, Бельгия, ЮАР, Эквадор, Чили и Панама	Нет

Обобщив полученную информацию о приведенных выше навигационных приложениях, можно прийти к следующим выводам:

- Любое навигационное приложение базируется на использовании определенных наборов географических данных, которые создаются непосредственно для самого приложения, либо используются общедоступные данные, например, такие как OpenStreetMap.
- В основе любого навигационного приложения лежит функция построения маршрута из точки А в точку Б.
- Набор возможностей навигационных приложений зависит от вида деятельности конечного пользователя, т.е. будет ли он использовать приложение при движении в транспортном средстве, при движении пешком или для туризма.

1.2 Сравнение средств разработки AR приложений и их возможностей

SDK — это своего рода программное обеспечение дополненной реальности, расшифровывается как Software Developer Kit, которые представляет собой набор программ и программного обеспечения, используемых для разработки другого программного обеспечения.

Главное преимущество SDK в том, что они позволяют разработчикам применять в разработке типовые и проверенные временем решения и ярлыки, вместо того, чтобы решать все типовые и типовые проблемы по отдельности и тратить на них время. Далее будут приведены наиболее распространенные из них.

Apple ARKit

ARKit используется для создания приложений как для iPhone и для iPad, в которых используется процессор A9, или выше [2]. ARKit позволяет разработчикам проектировать и разрабатывать приложения, которые расширяют возможности для каждого пользователя с помощью множества функций, таких как:

- Обнаружение и отслеживание 2D-изображений
- Распознавание и размещение 3D-объектов
- Обнаружение горизонтальной и вертикальной плоскостей
- Трекинг лица
- Стабильное и быстрое отслеживание движения
- Расчёт масштаба

Google ARCore

Google ARCore — это один из самых часто используемых SDK при разработке приложений для смартфонов и планшетов. Создаваемые с его помощью AR-приложения поддерживаются как на Android, так и на устройствах на базе iOS, позволяя разрабатывать кроссплатформенные AR-приложения. ARCore базируется на двух элементах: отслеживании позиции и

распознавании объектов. Ниже приведена небольшая часть из его выдающихся особенностей:

- Оценка освещенности в реальном времени
- Точное размещение виртуальных объектов
- Легкое отслеживание для создания реалистичных объектов
- Определение размера и местоположения вертикальных, горизонтальных и наклонных поверхностей
- Отслеживание движения в соответствии с положением телефона

Vuforia

Vuforia входит в число самых популярных SDK для разработки приложений дополненной реальности. Используя SDK Vuforia, разработчики приложений могут демонстрировать широкий спектр возможностей AR-приложений. Благодаря доступности API через Unity, Vuforia можно использовать для разработки собственных приложений под iOS и Android. Она также считается полным SDK с обширным набором функций для приложений AR.

- Идентификация и отслеживание целевых изображений, текстов на английском языке и 3D-объектов в режиме реального времени
- Размещение виртуальных объектов, таких как 3D-модели, в реальной среде
- Многоцелевые 3D-конфигурации
- Vuforia Engine Area Targets вместе с Area Target Generator
- Отсканированные Model Targets
- Расширенные Model Targets — Обнаружение нескольких моделей
- Продолжение работы при приостановке приложений
- Режим симуляции
- Vuforia Engine Tracking Scale

Wikitude

Wikitude используется при разработке мобильных приложений и AR-прототипов. Wikitude SDK позволяет разработчикам реализовывать

возможности геолокации, а также отслеживать изображения и распознавать объекты. Ниже приведены некоторые из его особенностей:

- 3D распознавание и отслеживание
- Распознавание и отслеживание изображений
- Распознавание в облаке
- AR на основе местоположения
- Наложение видео
- Интеграция смарт-очков
- Интеграция с внешними плагинами

MaxST

MaxST — это комплексная платформа, которая обеспечивает быстрый и простой процесс разработки AR-приложений. Он поставляется в виде двух SDK: 2D SDK для распознавания изображений и 3D SDK для распознавания среды. Его многочисленные функции и среды позволяют разработчикам быстро и легко создавать AR-приложения.

- Мгновенное отслеживание
- Идентификация горизонтальных/вертикальных плоскостей
- Одновременное расположение и отображение для создания «виртуальной карты» для отслеживания окружающей среды
- Отслеживание объектов, изображений и отслеживание нескольких маркеров
- Сканирование QR и штрих-кода
- Интеграция плагинов в Unity

EasyAR

EasyAR — это бесплатный SDK для разработки AR, с поддержкой основных мобильных платформ. EasyAR позволяет компаниям и разработчикам расширять возможности погружения в AR с помощью мобильных приложений. Разработчики могут использовать функции EasyAR в зависимости от приобретённых пакетов:

- Easy AR Basic: разработчики могут управлять рабочим процессом, улучшать API, обеспечивать совместимость, воспроизведение видео, сканирование QR-кодов и всесторонняя интеграция.
- EasyAR Pro: разработчики могут реализовать дополнительные функции, такие как отслеживание 3D-объектов, запись с экрана, одновременное обнаружение и отслеживание нескольких типов маркеров.

ARToolKit

ARToolKit — это SDK с открытым исходным кодом, который предлагает множество сложных функций для эффективного и быстрого создания AR-приложений, а также ПО для смарт-очков. Он предлагает скомпилированные SDK для платформ iOS, Android, Windows, Mac OS и Linux. Это один из первых AR SDK с простым в использовании интерфейсом и быстрой работой. Его многочисленные функции включают в себя:

- Интеграция GPS и компаса
- Поддержка Unity3D и OpenSceneGraph
- Распознавание 2D-объектов
- Одновременное отслеживание
- Поддержка как одной, так и двух камер сразу
- Сопоставление дополнительных элементов через OpenGL
- Интеграция со смарт-очками
- Поддерживает несколько языков

Xzing

Xzing — это ПО для обработки изображений и распознавания лиц и объектов на основе дополненной реальности. Имеется несколько продуктов Xzing: решение Augmented Face, Augmented Vision и Magic Face, которые предлагают функции распознавания лиц в режиме реального времени, отслеживания изображений и деформируемого отслеживания лица соответственно. Приложения могут быть разработаны для ПК, мобильного

устройства, или веб-браузера с помощью плагина Unity. Ключевые особенности включают в себя:

- 2D и 3D распознавание
- Распознавание и отслеживание маркеров
- Распознавание лиц в видео
- Отслеживание и замена лица

На рисунке 7 представлено сравнение части предоставляемых возможностей выше рассмотренных SDK.

	ARKit	ARCore	Wikitude	EasyAR	Vuforia	MAXST	ARToolKit	XZIMG
2D recognition	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
3D recognition	Yes	Yes	Yes	No	Yes	Yes	No	Yes
Lighting estimation	Yes	Yes	No	No	No	No	No	No
SLAM	Yes	Yes	Yes	Yes	Yes	Yes	No	No
Cloud recognition	Yes	Yes	Yes	Yes	No	Yes	No	No
Geolocation	Yes	Yes	Yes	No	No	No	No	No
Free version	Yes	Yes	Yes (trial only)	Yes	Yes (trial only)	Yes (trial only)	Yes	Yes (trial only)

Рисунок 7 – Таблица сравнения предоставляемых возможностей AR SDK

Как видно из рисунка 7 далеко не все AR SDK являются полностью бесплатными. Часть из них предоставляет весь спектр возможностей для разработки только при оформлении подписки или лицензии, либо только для тестирования.

Однако все AR SDK тем или иным образом пытаются решить две основные проблемы:

Первая проблема связана со способностью приложений дополненной реальности распознавать объекты в реальном мире и отслеживать их положение. Подходы, основанные на использовании искусственных элементов, таких как маркеры, размещенные в окружающей среде, очень надежны и хорошо работают. Основная сложность 3D-трекинга в реальном времени заключается в сложности сцены и движения целевых объектов, включая степени свободы отдельных объектов и их представление. Отслеживание на

основе зрения направлено на то, чтобы связать целевые местоположения в последовательных видеокдрах, особенно когда объекты движутся быстро относительно частоты кадров.

Вторая проблема заключается в определении точного восприятия глубины [3]. Стереоскопические дисплеи помогают, но дополнительные проблемы, включая конфликты аккомодации и конвергенции или низкое разрешение и тусклые дисплеи, заставляют объект казаться дальше, чем он должен быть. Использование Occlusion позволяет решить некоторые проблемы с глубиной, такие как последовательная регистрация для разных точек зрения [4]. Термин Occlusion подразумевает, что видимые части виртуальных и реальных объектов могут накладываться друг на друга и при получении финального изображения необходимо отображать те части объектов, которые находятся в переднем плане по отношению к пользователю. Таким образом, основная цель Occlusion — сохранить правила прямой видимости при создании AR-сцен. Сложность его применения заключается в том, что для этого необходимо получить границы объектов, в которых начинается перекрытие одного объекта другим. Решение данной задачи в большинстве случаев требует высокой производительности устройства для определения объектов на переднем плане, для чего необходимо сравнивать множество значений точек объектов.

Осенью 2021 года была анонсирована еще одна платформа дополненной реальности для текущего и будущих поколений оборудования дополненной реальности Niantic Lightship [5], которая дает разработчикам возможность создавать широкий спектр приложений, которые будут способствовать развитию технологии.

Их ключевая область разработки, картографирование в реальном времени, предполагает более точную адаптацию виртуальных объектов к реальному миру с помощью камер смартфонов. Lightship стремится к тому, чтобы сделать возможной физику виртуальных объектов. Для реализации

данной возможности Lightship требуются только датчики цвета RGB, которые есть в большинстве камер (в отличие от сканеров LiDAR, доступных только в устройствах высокого класса), поэтому эта функция будет доступна для большого числа мобильных устройств.

На рисунке 8 приведены основные возможности Lightship SDK, которыми могут воспользоваться разработчики мобильных приложений для создания сцен дополненной реальности.

ARDK Feature and Terms	Before May 1, 2022		After May 1, 2022, Pricing Per Application ¹	
	Free	Free		
Depth and Occlusion	Free		Free	
Real-Time Meshing	Free		Free	
Semantic Segmentation	Free		Free	
Development Tools	Free		Free	
Multiplayer ³	Free ²	Monthly Active Users⁴ of Multiplayer Service	Cost per 10K Users per Month	MAU Data Transfer Cap
		1–50K MAU per app	Free	50MB per MAU per month per app
		> 50K MAU per app	\$5.00	50MB per MAU per month per app

Рисунок 8 – Функциональные возможности Lightship SDK

1.3 Обзор навигационных приложений с возможностями дополненной реальности

Сейчас среди лидеров в этой области можно отметить навигационные приложения от Apple, Google и Яндекс. За последние пять лет эти компании добавили к своим навигационным сервисам возможность использования дополненной реальности. Работают они по принципу взаимодействия с их предоставляемыми картами, в которых используется возможность задания точек начала и конца пути для дальнейшего построения маршрута. При этом в режиме дополненной реальности создаются указатели, которые добавляются в местах развилки пути, предварительно получая текстовые данные с карт. Далее будут рассмотрены разработки этих компаний в области AR навигации.

Яндекс.Карты

В 2017 году Яндекс.Карты для мобильных устройств операционной системы iOS 11 получили дополненную реальность. Использовать функцию

можно, если на iPhone или iPad установлено обновление карт до версии 9.6. Совместимы все устройства с процессором от A9 (iPhone 6S/SE и новее) [6].

Дополненная реальность в Яндекс.Картах работает в режиме пешеходной навигации. Под картой появляется синяя кнопка с пиктограммой глаза, по нажатию на которую гаджет включает камеру и прокладывает маршрут с помощью цифровых указателей прямо по реальной местности.

Данная функция может помочь людям, которые плохо работают с простыми картами: теперь для навигации не нужно никаких ориентиров и попыток выставления карты по ходу маршрута. Когда подсказки буквально накладываются на то, что вы видите перед собой, ошибка почти исключена.

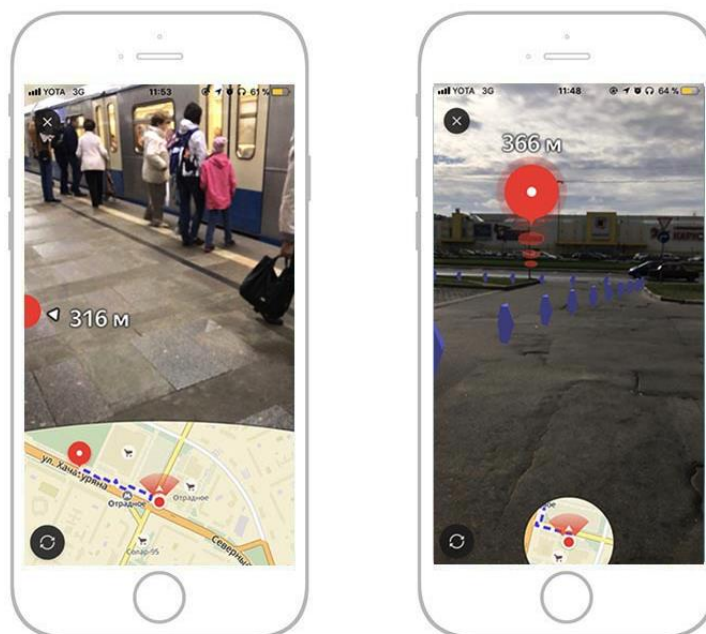


Рисунок 9 – Пример использования AR навигации в Яндекс.Картах

Google Карты

В 2019 году компания Google добавила новые возможности в свое картографическое приложение, которые помогут пользователям быстрее добраться до определенного места. Речь идет об улучшении функции дополненной реальности («AR Liew View») [7].

Когда используется «AR Liew View» и выстроен пеший маршрут, приложение отображает на экране 3D-стрелки и указатели, что заметно облегчает ориентирование на незнакомой местности. Для активации данной

функции, необходимо нажать на кнопку Directions (Направления) и выбрать опцию Walking (Пешком). При использовании AR-режима в нижней части экрана отображается стандартная карта Google с пошаговой навигацией.

«AR Liew View» также можно использовать и при передвижении на общественном транспорте. Указатели подскажут, когда нужно выйти и куда двигаться дальше.

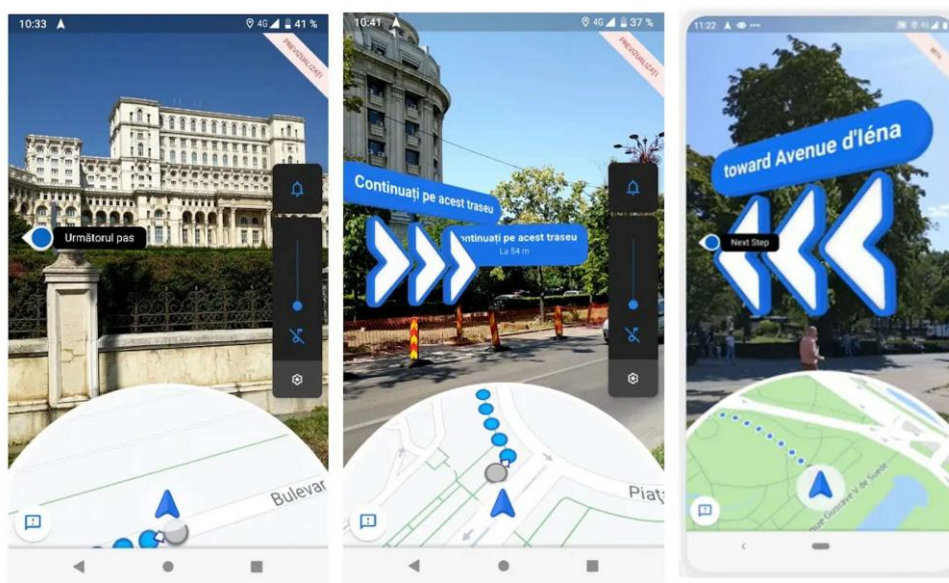


Рисунок 10 – Пример использования AR Liew View

Также Google Maps можно использовать для ориентирования в AR-режиме Indoor Live View в аэропортах, торговых центрах, железнодорожных и автовокзалах. Оказавшись в незнакомом месте, пользователь сможет задать пеший маршрут внутри здания, после чего на экране появятся виртуальные знаки и стрелки, которые укажут путь до ближайшей лестницы, эскалатора или лифта [8].

Чтобы понять, где находится пользователь, система на основе искусственного интеллекта анализирует большое множество изображений из сервиса «Просмотр улиц» (Street View).

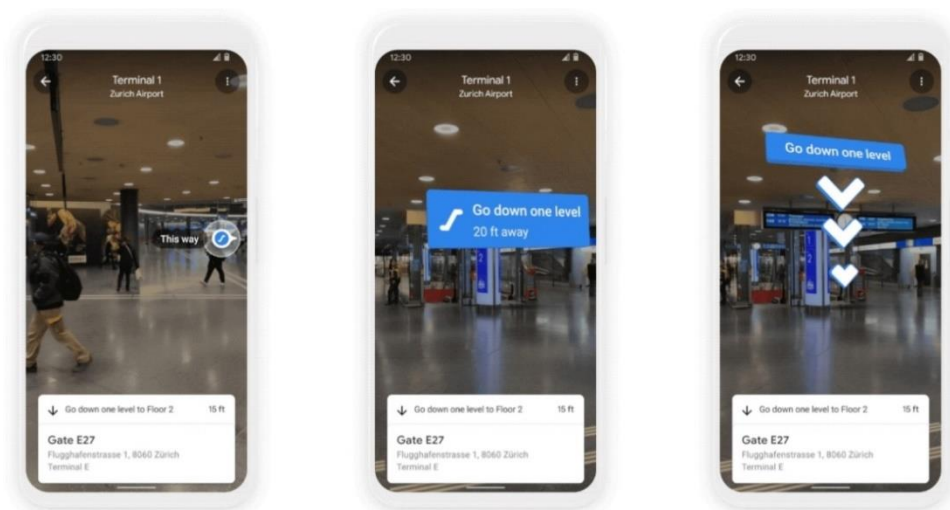


Рисунок 11 – Пример использования Indoor Live View

Однако, пока AR-навигация внутри помещений доступна только в некоторых торговых центрах США (в Чикаго, Лонг-Айленде, Лос-Анджелесе, Ньюарке, Сан-Франциско, Сан-Хосе и Сиэтле). В скором будущем функция также заработает в Токио и Цюрихе, а позднее и в других городах мира.

Apple Maps

27 сентября 2021 в Apple Maps было добавлено обновление, часть которого состоит из иммерсивных пешеходных маршрутов, отображаемых в дополненной реальности [9]. Данные иммерсивные пешеходные маршруты предоставят пользователям пошаговые инструкции по передвижению в дополненной реальности. Для использования данной особенности приложение просит пользователя просканировать область, в которой он находится, для генерации высокоточной позиции при предоставлении подробных указаний, которые можно будет просмотреть в контексте реального мира.

В дополнение к AR навигации, в обновлении Apple Maps была добавлена функция 3D-карт, которые позволяют пользователям перемещаться по городам с 3D-представлением города, которое включает сведения о высоте, новые дорожные метки и множество пользовательских ориентиров.



Рисунок 12 – Навигация в режиме дополненной реальности в Apple Maps

Функция дополненной реальности доступна с конца 2021 года в основных поддерживаемых городах, включая Лондон, Лос-Анджелес, Нью-Йорк, Филадельфию, Сан-Диего, Сан-Франциско и Вашингтон [10].

Также следует отметить, что не только крупные компании ведут разработки в данной области. В качестве примера можно привести приложение AR Indoor Navigation [11], которое разрабатывали для навигации по офисному пространству, в рамках которого гости компании могли в интерактивном режиме познакомиться с офисом клиента и попасть в нужную для них локацию. Также приложение применяется для адаптации новых сотрудников, что значительно упрощает навигацию по офису. Навигация осуществлялась в real-time, в которой пользователь мог строить маршрут из определенных точек пространства к необходимому отделу и видеть путь в виде стрелок в дополненной реальности. На пути гостя появлялся AR-контент, соответствующий выбранному маршруту, а при подходе к необходимой локации появлялась 3D-модель символа компании. Выбирая маршрут, пользователь мог прочитать больше информации о компании и ее отделах, а также прослушать аудио сопровождение. Для разработки данного проекта использовалась SDK Immersal, которая с помощью фотограмметрии позволяла

получать 3D карту помещения, на основе которой навигационные объекты для всех необходимых маршрутов.

Другим примером является Campus Go [12] — проект, созданный за два дня на хакатоне по VR и AR технологиям. Реализован не был, но очень заинтересовал от сотрудников из лаборатории AR/VR Сбербанка. Его идея заключалась в решении проблемы связанной с тем, что кампус ДВФУ — очень большой и запутанный. Приложение Campus Go нужно, чтобы облегчить участникам мероприятий навигацию по ДВФУ: оно объединяет в себе программу лекций, карту кампуса и навигацию по мероприятию. Приложение подберет самый удобный маршрут: скажет, на какой этаж ехать, в какую дверь зайти и куда повернуть.

При запуске приложения пользователь выбирает одно из проходящих в ДВФУ мероприятий. Включается камера и определяет, где находится человек. На полу появляются стрелки, а также плашка с местом назначения и расчетом времени до места. Во время движения появляются новые стрелки и подсветка нужных дверей или кнопок лифта.

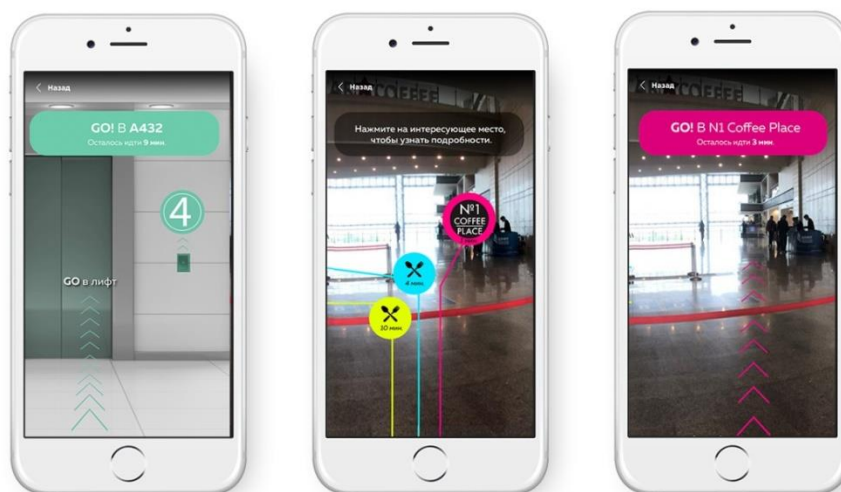


Рисунок 13 – Пример работы приложения Campus Go

Проанализировав представленные в данном разделе приложения можно выделить два основных направления в развитии навигационных приложений с поддержкой дополненной реальности: узконаправленную навигацию по

конкретным организациям и навигацию по общественным местам посредством задания конечных точек маршрута. В первом случае приложения нацелены на то, чтобы ознакомить пользователя с самой организацией и облегчить их ориентирование по крупной запутанной структуре. А во втором для перехода от классического варианта навигации с использованием 2D карт на навигацию по реальному миру посредством привнесения в него дополнительных элементов, исключающих вероятность движения пользователя в неправильном направлении.

Также стоит отметить, что данные приложения прокладывают маршрут следующим образом: анализируется текущее положение пользователя и конечная точка маршрута, после чего выстраивается путь, по которому необходимо будет двигаться, а затем исходя из его протяженности и изменений уровня высоты, определяются точки, в которые будут добавлены навигационные виртуальные объекты.

Исходя из того, что построение таких маршрутов является далеко не тривиальной задачей и требует определённых вычислений. В связи с этим было решено разработать приложение, которое позволит создать систему навигационных объектов по организации путем непосредственного их добавления пользователем приложения в необходимых для него местах взамен использования алгоритмов прокладывания пути. В дополнение к этому у разрабатываемого приложения должен быть и классический вариант навигационного приложения, который позволит убедиться в корректности задаваемых местоположений.

2 ТРЕБОВАНИЯ К РАЗРАБАТЫВАЕМОМУ ПРИЛОЖЕНИЮ И ПОДБОР НЕОБХОДИМЫХ СРЕДСТВ РАЗРАБОТКИ

2.1 Общие требования к программному продукту

Бизнес-требования

- Разрабатываемое приложение должно предоставлять возможность пользователю для создания системы виртуальных навигационных объектов, которая упростит в дальнейшем навигацию по организации, в которой будет использованная полученная система.

Функциональные требования

- Приложение должно иметь два режима использования: картографический и AR-режим.
- Пользователь должен иметь возможность переключения между режимами приложения.
- В картографическом режиме должна отображаться карта местности, которая должна включать в себя дороги, здания и природные объекты.
- В картографическом режиме должна быть возможность отображения местоположений созданных объектов.
- В картографическом режиме должны быть возможности масштабирования и смещения карты относительно текущего положения.
- В картографическом режиме должно отображаться текущее положение пользователя.
- В AR-режиме должна быть предусмотрена возможность добавления новых объектов.
- У пользователя в AR-режиме должна быть возможность изменения расстояния удаления объекта от себя при добавлении объекта.
- В AR-режиме должна быть возможность внесения текстовой информации в панели виртуальных объектов.

Нефункциональные требования

- При переходе в картографический режим карта должна обрисовываться в зависимости от текущего положения пользователя.
- Виртуальные объекты в AR-режиме должны отображаться с использованием Occlusion.

Ограничения

- Приложение должно корректно работать на мобильных устройствах с операционной системой Android.
- Минимальным API level – Android 7.0 (API level 24).
- Максимальный API level – Android 12 (API level 32).
- Ориентация – портретная.
- В качестве платформы для разработки должен быть использован Unity.
- В качестве языка программирования должен быть использован C#.
- Разрабатываемое приложение должно исправно работать с русскоязычной и английской версиями системы.

Требования к данным

- Информация о созданных навигационных объектах во время использования приложения должна храниться в отдельном JSON файле.
- Для хранения информации о созданных навигационных объектах должен быть предусмотрен сервер, обращение к которому для получения этих данных и их записи в JSON файл будет происходить при смене режимов приложения.
- После добавления нового объекта в AR-режиме на сервер должен поступать запрос на добавление новой записи с параметрами данного объекта.

2.2 Выбор средств для разработки приложения

Ключевой частью разрабатываемого приложения будет являться AR-режим. В приложениях с дополненной реальностью имеется два основных подхода при размещении виртуальных объектов: с использованием заранее подготовленных маркеров, либо без их использования. В первом случае виртуальные объекты размещаются поверх тех объектов реального мира, которые были заданы определенным маркером. В данном маркере выделяются ключевые особенности, и при наведении камеры мобильного устройства происходит детектирование данных особенностей, при успехе которого будет размещен сам объект. Во втором случае размещение объектов происходит исходя от положения устройства пользователя. Для разрабатываемого приложения лучше подходит второй способ, т.к. для виртуальных объектов будут сохраняться конкретные местоположения без привязки к конкретным объектам.

Для AR-приложения самым важным является то, насколько качественно будут отображаться создаваемые в нем объекты и то, как они будут взаимодействовать с реальным миром. Однако на самом деле создаваемые объекты создают лишь иллюзию контакта с окружающим миром. В реальности для создания такого эффекта происходит обработка получаемого с камеры мобильного устройства изображения, в ходе которой выделяются границы объектов реального мира, по которым строится виртуальная карта реального мира. Таким образом, для создания эффекта взаимодействия виртуальных объектов с реальным миром, создаваемые объекты взаимодействуют с полученной виртуальной картой мира, результат чего отображается на экране устройства, но отображается на экране лишь только сам объект, без полученной карты. В большинстве случаев реализации данной особенности требуются более дорогостоящие мобильные устройства, которые либо обладают датчиком Lidar, либо поддерживают API глубины. В связи с этим для того, чтобы разрабатываемое приложение могло использовать данную особенность AR

приложений для большинства современных мобильных устройств, в качестве средства для разработки AR части разрабатываемого программного обеспечения был выбран Lightship SDK, который позволяет использовать данную особенность и на устройствах, которые не обладают такими датчиками.

В качестве основной платформы для разработки был использован Unity. В данной среде разработки ПО используется компонентно-ориентированный подход, в рамках которого у разработчика появляется возможность создания объектов и добавления к ним различных компонентов (например, визуальное отображение объекта и способы управления им). Благодаря удобному Drag & Drop интерфейсу и функциональному графическому редактору платформа Unity позволяет рисовать карты и расставлять объекты в реальном времени и сразу же тестировать получившийся результат. Также Unity поддерживает большое количество платформ, технологий, API. Создаваемое в данной среде ПО можно легко портировать между ОС Windows, Linux, OS X, Android, iOS, на консоли семейств PlayStation, Xbox, Nintendo, на VR- и AR-устройства.

В качестве альтернатив платформ для разработки могли быть использованы Unreal Engine и Android Studio. В сравнении с объектно-ориентированными платформами Unity и Unreal Engine, при разработке приложения в Android Studio не будет возможности проводить тестирования части разрабатываемого приложения непосредственно в самой среде разработки. Для проведения тестирования в этом случае необходимо будет производить сборку приложения и переносить его на мобильное устройство. Либо использовать эмулятор мобильного приложения, что будет требовать наличие большого объема оперативной памяти от устройства, на котором будет производиться проектирование приложения. Также в этом случае изображение, которое формируется в эмуляторе, будет поступать с камеры устройства, на котором будет производиться проектирование. В связи с этим данные аспекты заметно увеличат время на разработку приложения. Unreal Engine поддерживает основные SDK для создания AR-приложений – ARCore и ARKit.

Из этого следует, что если разработчику потребуется другой SDK для создания приложения, то для этих целей ему придется использовать другую платформу разработки.

Также необходимо выбрать программные пакеты для Unity, которые позволяют отображать данные OpenStreetMap в сцену для разработки картографического режима. Среди этих средств имеются платные, такие как Mantle, Online Maps, Real World Terrain, и бесплатные – CityGen3D, Tangram, Mapbox Unity SDK, UtyMap [15]. Изначально были рассмотрены бесплатные средства, и среди них для разрабатываемого приложения лучше всего подходит Mapbox Unity SDK [16]. В Mapbox Unity SDK уже имелись необходимые части для создаваемого приложения, такие как возможность масштабирования карты с последующей загрузкой ранее не отображавшихся частей сетки карты, отображение положения пользователя на карте в реальном времени, выбор различных настроек карты (стиль, метки, названия, возможность добавления 3D-зданий, прокладывание маршрутов по дорогам и т.д.), а также обширная документация по взаимодействию с создаваемой картой.

3 ПРОЕКТИРОВАНИЕ ПРИЛОЖЕНИЯ

3.1 Архитектура приложения

В качестве шаблона проектирования для приложения был использован шаблон MVC (Model-View-Controller), который предполагает разделение данных приложения, пользовательского интерфейса и управляющей логики на три отдельных компонента: модель, представление и контроллер – таким образом, что модификация каждого компонента может осуществляться независимо (рис. 16).

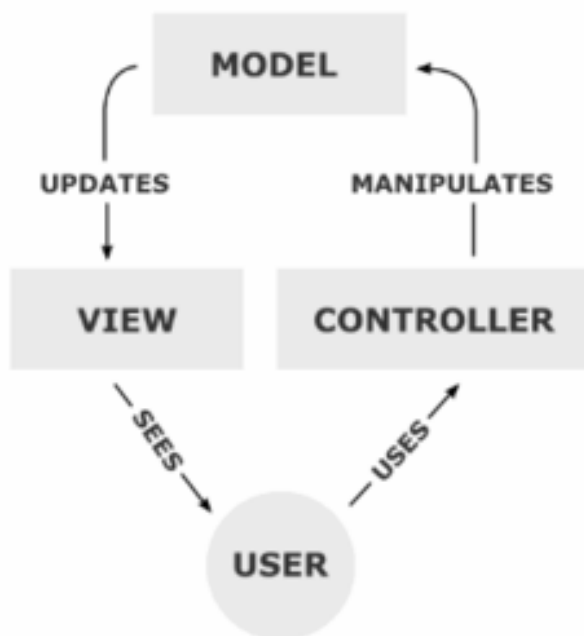


Рисунок 14 – Концептуальная схема шаблона MVC

Так, при создании интерактивных приложений, модульность компонентов имеет огромные преимущества. Максимально возможная изоляция функциональных модулей друг от друга облегчает разработчику приложения понимание и модификацию каждого конкретного модуля [13].

Архитектура приложения приведена на рисунке 17.

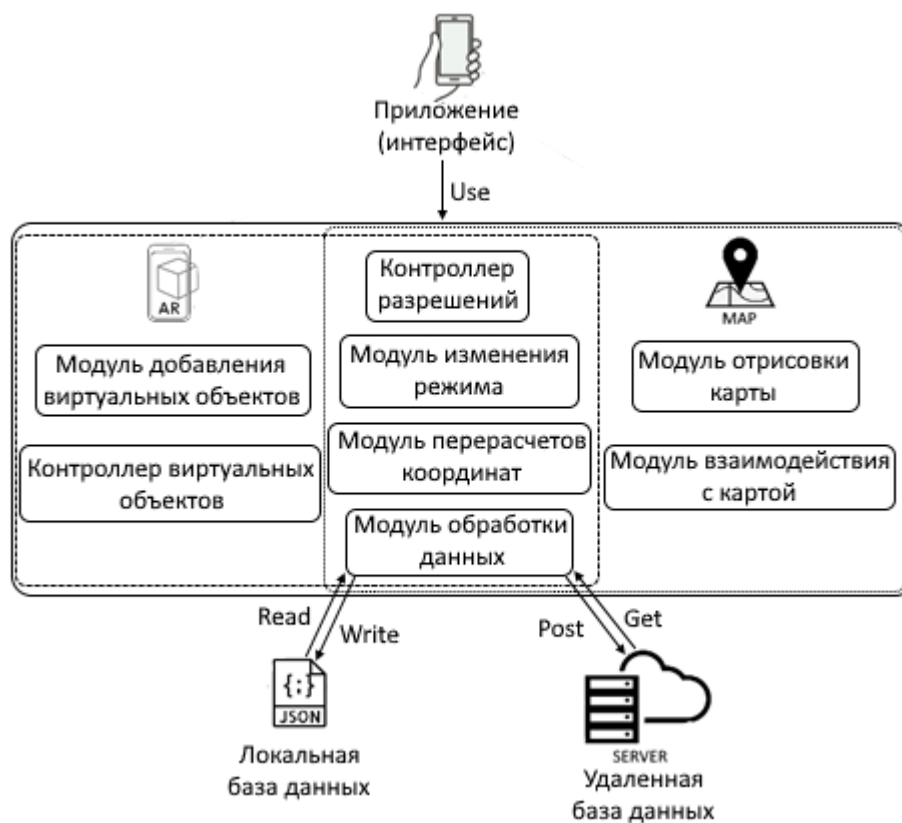


Рисунок 15 – Архитектура приложения

Логика приложения функционально разделена на несколько частей.

Контроллер разрешений отвечает за то, чтобы у приложения были получены разрешения на использование необходимых данных для корректной работы приложения. Данный контроллер будет вызывать запрос на разрешение использования данных о местоположении пользователя при использовании картографического режима. При переключении в AR-режим приложение будет запрашивать разрешение на использование данных камеры устройства. При том, если у приложения будут получены необходимые разрешения, то запросы на их использование производиться не будут.

Модуль изменения режима позволяет пользователю изменять текущий режим приложения на другой.

Модуль перерасчетов координат служит в качестве преобразователя географических координат в трехмерные и обратно. С помощью него реализуется получение необходимых 3D координат для AR-режима и 2D координат объектов для отображения в картографическом режиме.

Модуль обработки данных необходим для получения хранимых данных с сервера и последующей загрузки этих данных в локальную базу данных. Также с помощью данного модуля выполняются запросы на добавление новых данных на сервер и чтение данных с локальной базы данных для дальнейшего их использования в приложении.

Модуль добавления виртуальных объектов создает в указанных трехмерных координатах, преобразованных из локального хранилища данных, необходимые типы объектов. Также данный модуль отвечает за предоставляемую пользователю возможность добавления новых объектов.

Контроллер виртуальных объектов отвечает за взаимодействие пользователя с объектами. Благодаря данному модулю сохраняются текстовые данные в панелях виртуальных объектов.

Модуль отрисовки карты отвечает за отображение тайлов карты в зависимости от того, какие из них попадают в обозреваемую область экрана устройства, а также от текущего уровня масштабирования карты. На самой карте должны отображаться различные компоненты, такие как здания, дороги и природные объекты.

Модуль взаимодействия с картой отвечает за функции масштабирования карты, смещение карты относительно ее текущего положения и отображение местоположений имеющихся навигационных объектов.

3.2 Алгоритм преобразования географических координат в локальные

В приложении для создания нового объекта необходимо задать его координаты в трехмерном пространстве. Но данная система координат не имеет привязки к окружающему миру, а выстраивается относительно ориентации устройства в момент запуска AR-режима. В связи с этим для фиксирования мест расположения создаваемых объектов необходима другая система координат, которая будет зависеть не от положения пользователя, а будет привязана к конкретной точке в пространстве земного шара. Таким образом,

для решения задачи о сохраняемых данных о местоположениях создаваемых объектов необходимо использовать географические координаты.

При этом система координат, которая используется в приложении, каждый раз будет перестраиваться при повторном запуске AR-режима. Она будет располагаться параллельно уровню пола, но ось X будет направлена по направлению исходящему перпендикулярно задней плоскости мобильного устройства, спроецированной на плоскость уровня пола. В связи с этим при перерасчете координат необходимо повернуть полученную систему на север, для совмещения с географической системой координат. Однако повернуть самому систему напрямую не имеется возможности, поэтому при перерасчетах будет использоваться матрица поворота по оси Z.

$$M_z(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad (1)$$

где α – угол поворота вокруг оси Z.

В самом приложении для реализации функций по воссозданию уже созданных объектов и сохранению данных о местоположении добавляемых объектов необходимо выполнять перерасчеты координат из трехмерной системы координат в географическую систему и в обратную сторону. Для этого отдельно рассматриваются значения широты и долготы для проецирования на плоскость и значение высоты над уровнем моря, для задания высоты расположения объекта. Для получения координат в плоскости существует два подхода: получение проекции Меркатора эллипсоида на плоскость и проецирование сферы на плоскость. Исходя из того, что в первом случае приближение эллипсоида к поверхности Земного шара более точное, чем у сферы, то получаемые координаты в этом случае будут более точными. В связи с этим был выбран вариант с проецированием эллипсоида. В этом случае проекция Меркатора эллипсоида на плоскость будет задаваться следующим образом:

$$\begin{aligned} x &= a \cdot \lambda \\ y &= a \cdot \ln\left(\tan\left(\frac{\pi}{4} + \frac{\varphi}{2}\right) \cdot \left(\frac{1-e \cdot \sin \varphi}{1+e \cdot \sin \varphi}\right)^{\frac{e}{2}}\right), \end{aligned} \quad (2)$$

где x и y – прямоугольные координаты, λ – долгота на эллипсоиде в радианах, φ – широта на эллипсоиде в радианах, a – значение большой полуоси эллипсоида, $a = 6378137$ м, e – значение эксцентриситета эллипсоида.

При воссоздании объектов в AR-режиме для получения координат виртуальных объектов в пространстве Unity будут получаться следующим образом. Исходя из значений широты и долготы пользователя и конкретного виртуального объекта, получаются спроецированные координаты на плоскости. На основе полученных спроецированных координат, вычисляется разница между координатами пользователя и объекта. А так как в системе координат Unity пользователь всегда изначально будет находиться в нулевых координатах, то полученный вектор с разницей координат пользователя и объекта и будет являться координатами объекта, в которых его необходимо создать. Значение же высоты для объекта будет вычисляться как сумма текущего значения высоты над уровнем моря пользователя с разницей между этим значением и сохраненным значением высоты объекта.

Таким образом, для получения координат воссоздаваемых объектов используется следующий алгоритм:

1) Перевод координат пользователя и объекта в систему координат ориентированную на север при помощи матрицы поворота и угла на север, который передается с датчиков устройства пользователя

2) Вычисление проекционных координат пользователя и объекта по данным из Input.Location и сохраненным значениям геокординат объекта

3) Получение координат объекта в системе координат, ориентированной на север, путем сложения координат пользователя полученных в данной системе с разницей проекционных координат объекта и пользователя

4) Перевод координат объекта в исходную систему при помощи поворота его координат на обратный угол

Также у пользователя будет иметься возможность удаления объекта на определенное расстояние от себя. В этом случае, когда будут известны значения направляющего вектора от пользователя до объекта, будут получены проекционные координаты объекта путем сложения проекционных координат пользователя с координатами направляющего вектора. Затем из его проекционных координат будут получены необходимые для сохранения значения широты и долготы путем преобразования формулы 2 для получения данных параметров. Сохраняемое значение высоты над уровнем моря для объекта будет получено как сумма текущего значения пользователя с разницей между высотами пользователя и объекта в системе Unity.

3.3 Сохраняемые данные навигационных объектов

Как было выяснено раньше, для воссоздания ранее созданных объектов будут необходимы их географические координаты. Помимо этого у пользователя приложения будет возможность выбора вида создаваемого виртуального объекта, а также предполагается внесение текстовой информации в информационные поля объектов. В связи с этим, для каждого объекта необходимо сохранять следующий набор данных:

- данные о широте и долготе объекта, которые будут браться с получаемых данных от устройства пользователя о его местонахождении с поправкой на расстояние, на которое удалит от себя пользователь виртуальный объект;
- данные о высоте над уровнем моря создаваемого объекта;
- тип созданного объекта, из предлагаемого списка;
- внесенную в поле панели объекта текстовую информацию;
- идентификатор объекта.

В ходе использования приложения информация о всех объектах будет храниться в JSON файле для того, чтобы считывать с него информацию о расположении уже созданных объектов во время взаимодействия с картой и в

AR-режиме. Также во время создания нового объекта в AR-режиме в файл будут заноситься его данные, после чего на сервер поступит запрос о добавлении данных данного объекта.

3.4 Серверная часть

Для разработки данной части было принято решение использовать REST API, который является архитектурным стилем взаимодействия компонентов распределенного приложения в сети. Это было связано с тем, что во время использования приложения на мобильном устройстве все данные об объектах будут храниться в JSON формате, а благодаря данному стилю появится возможность реализовать удаленное хранение данных для их получения всем пользователям в том же формате.

Непосредственно перед запуском удаленного сервера был разработан программный код на JavaScript, который позволит осуществлять запросы на получение и добавление данных на сервер. Его программный код представлен в приложении А.

Для тестирования, используя NodeJS, был развернут локальный сервер, к которому производились запросы с Unity и приложения Postman. На рисунке 16 приведен ответ сервера в приложении Postman, при котором один объект был добавлен с помощью запроса в Unity, а второй при помощи Postman.

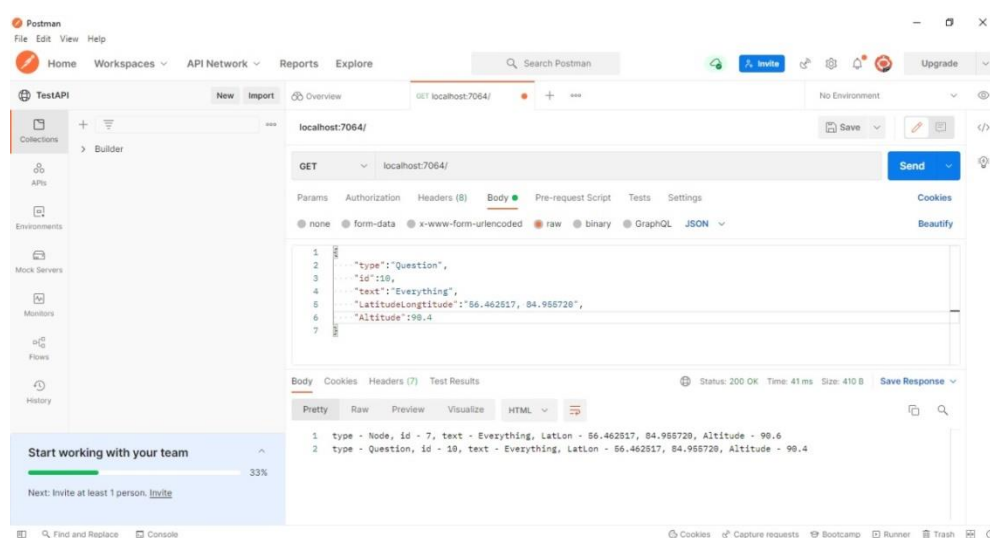


Рисунок 16 – Ответ сервера после добавления нового объекта

На рисунке 17 приведена схема запросов к базе данных с помощью методов GET и POST при использовании веб-сервиса REST.

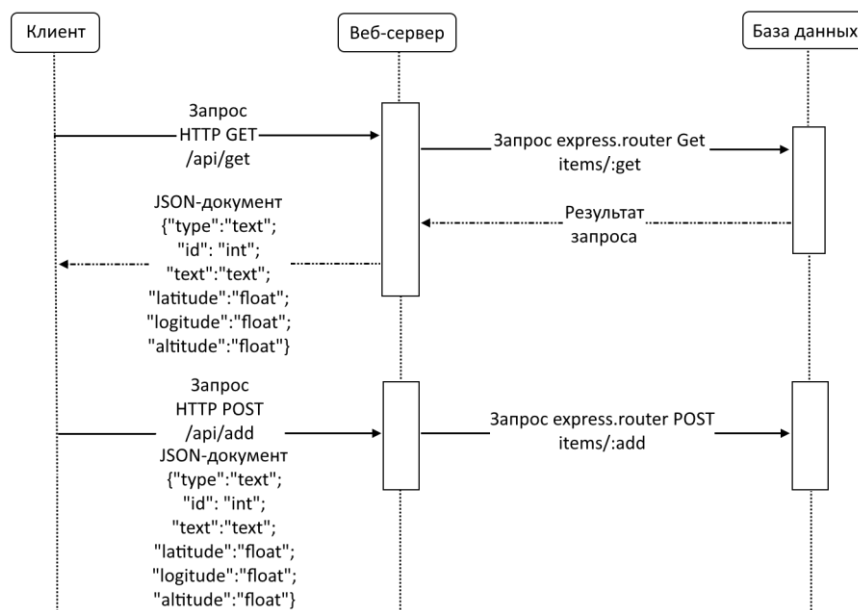


Рисунок 17 – Схема запросов POST и GET с помощью REST

3.5 Картографический режим

В связи с тем, что для разработки картографического режима было решено использовать пакет Mapbox Unity SDK, для приложения был создан стиль отображения объектов на карте. Данный стиль был предварительно настроен в браузере с использованием их веб-сервиса и в дальнейшем был использован в Unity. Пример полученного изображения карты в данном стиле приведен на рисунке 18.

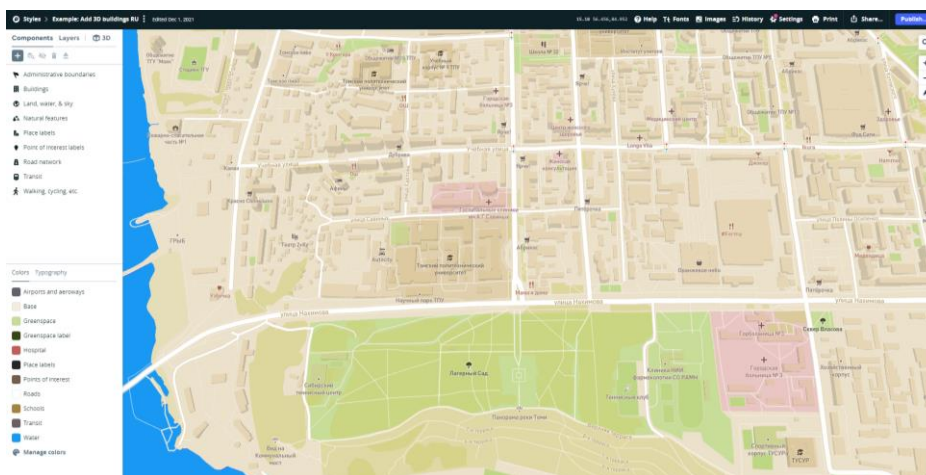


Рисунок 18 – Вид сверху на часть Кировского района города Томска с использованием разработанного стиля

В Unity основным объектом на сцене, который отвечает за то, что будет отображаться на экране, является MainCamera. Таким образом, то, что попадает в область проекции камеры на плоскость с нулевым уровнем высоты, будет отображаться на экране устройства. В связи с этим сама карта располагается на нулевом уровне высоты, и ее размер напрямую зависит от области проекции камеры, как показано на рисунке 19. Исходя из этого, карта будет получать информацию, какие тайлы карты ей необходимо отобразить.

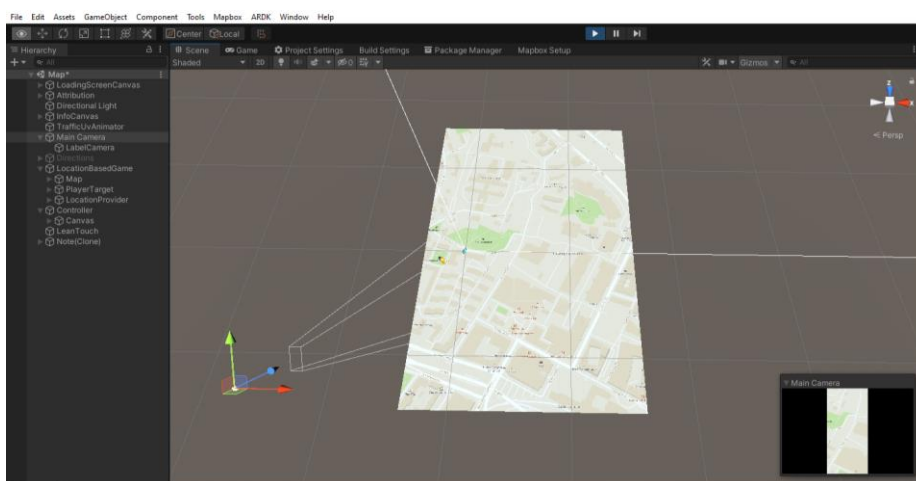


Рисунок 19 – Пример получение необходимой области карты

Для масштабирования и смещения камеры относительно карты используются данные из `Input.GetTouch()`, которые передают информацию о точках касания экрана пользователем. Таким образом, при масштабировании учитывается изменение расстояния между положениями двух касаний относительно друг друга. При увеличении этой разницы будет происходить приближение камеры к карте пропорционально получаемой разнице, а при уменьшении камера будет удаляться от карты. Помимо этого для камеры были заданы граничные значения для положения по высоте во избежание получения отрицательных или бесконечно больших значений высоты камеры. Поэтому, перед как изменить положение камеры по высоте, будет проверяться, входит ли сумма текущей высоты с вычисленным ее приращением в допустимый диапазон, а после получения удовлетворительного ответа будет происходить само масштабирование. Для смещения камеры в плоскости параллельной

плоскости карты используется изменение положения одного касания. Тем самым и будет задаваться направление смещения камеры. Используемый код для этой задачи приведен в приложении Б.

Положение пользователя на карте будет отображаться специальным объектом, координаты которого будут зависеть от получаемых данных устройства о геопозиции с помощью `Input.Location`, предварительно преобразованных в двухмерные координаты карты. Программный код данной механики приведен в приложении В.

3.6 Режим дополненной реальности (AR-режим)

Для данного режима был разработан специальный интерфейс, который будет позволять пользователю добавлять новые виртуальные объекты, а также ориентироваться по ранее созданным объектам. Программный код, отвечающий за реализацию данных функций, приведен в приложении Д.

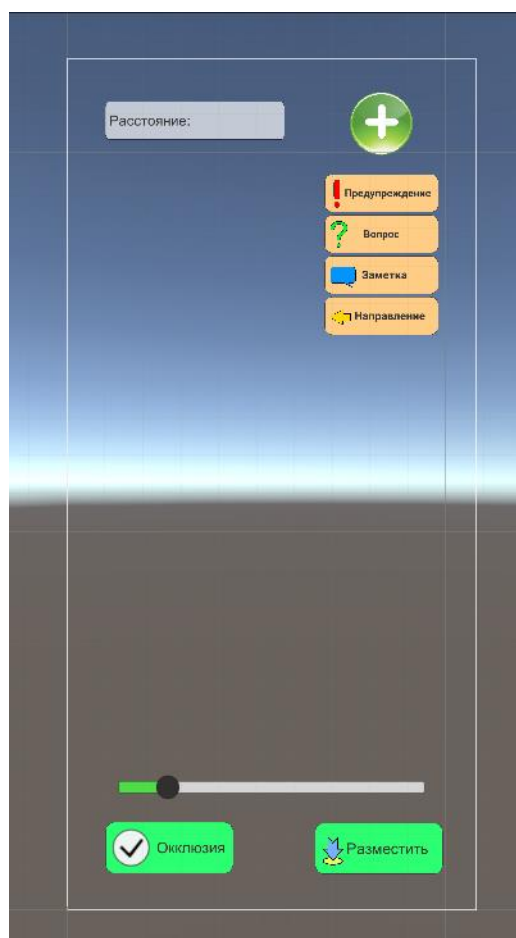


Рисунок 20 – Разработанный интерфейс для AR-режима в Unity

Для создания навигационной системы были созданы прототипы виртуальных объектов с помощью 3D моделирования в Blender, после чего они были импортированы в Unity. Созданные 3D объекты приведены на рисунках с 21-24.

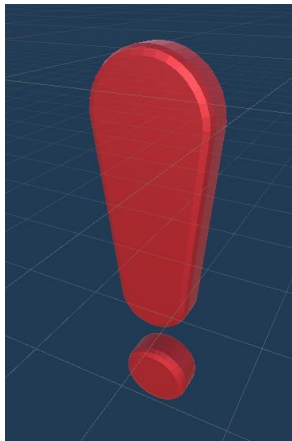


Рисунок 21 – Предупреждающий объект

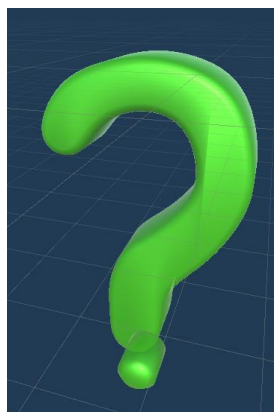


Рисунок 22 – Вопросительный объект

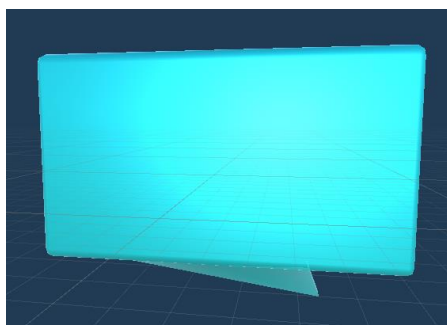


Рисунок 23 – Объект в виде заметки

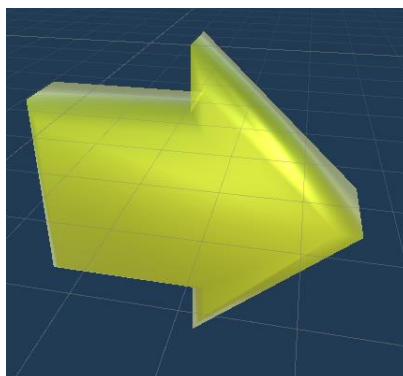


Рисунок 24 – Направляющий объект

Также для данных объектов были сделаны информационные панели, в которые пользователь может занести тактовую информацию. Данная информация сохраняется для каждого объекта, поэтому в дальнейшем при повторном взаимодействии с данным объектом в его панели отобразится внесенная в него информация. Код контроллера создаваемых объектов приведен в приложении Е. Пример информационной панели приведен на рисунке 25.

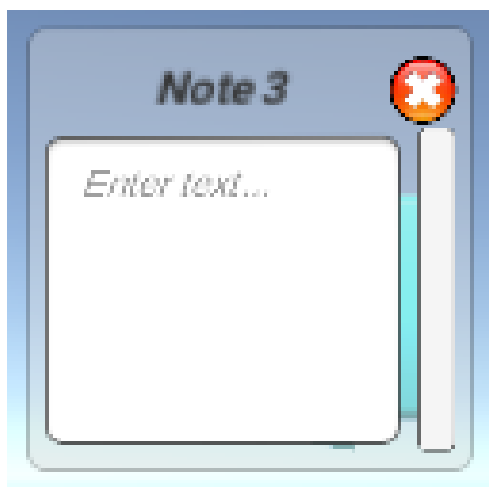


Рисунок 25 – Пример информационной панели объекта

В AR добавление новых объектов происходит в следующей последовательности:

- 1) Нажать на кнопку добавления нового объекта, после чего становится активным список из четырех кнопок для выбора вида создаваемого объекта.

2) Нажать на одну из кнопок выбора объекта, после чего появится соответствующий объект, который будет находиться на заданном по умолчанию расстоянию от камеры, а кнопки выбора объекта скроются.

3) Задать расстояние удаления объекта с помощью ползунка слайдера и нажать на появившуюся кнопку «Разместить» для привязки объекта к текущему местоположению.

4) Нажать на созданный объект, после чего появится информационная панель объекта, в которую нужно ввести текст.

3.7 Переход между режимами приложения

В связи с тем, что у разрабатываемого приложения было заявлено два режима использования, то для каждого из них была создана сцена в Unity, а также переход между сценами при нажатии на соответствующую кнопку. Для плавности перехода между ними было создано две анимации (FadeIn и FadeOut), аниматор с триггером и скрипт (LevelChanger) переключающий этот триггер. Сам аниматор работает по следующему алгоритму:

- При входе в одну из сцен вызывается анимация FadeIn.
- При переключении состояния триггера вызывается анимация FadeOut, в конце которой вызывается функция загрузки следующей сцены.

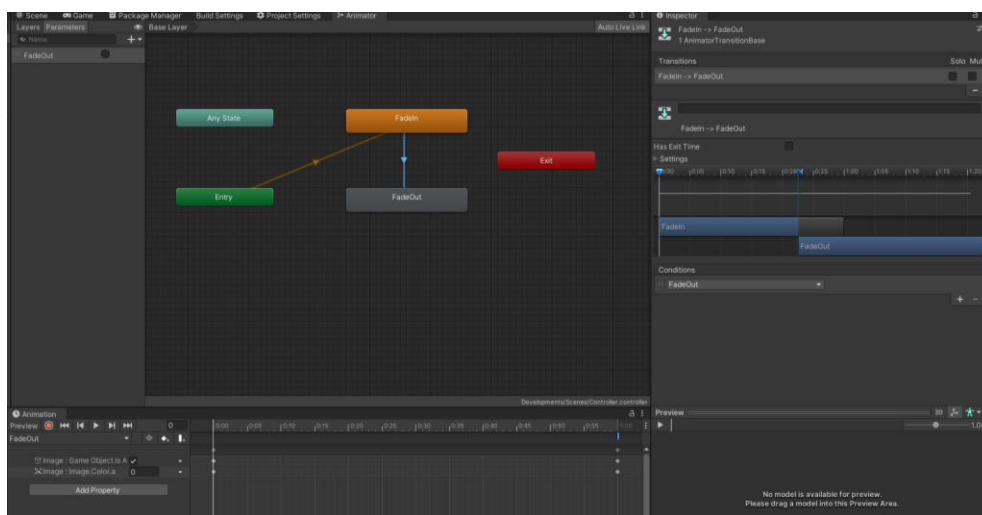


Рисунок 26 – Аниматор для смены сцен

Для изменения состояния активности кнопки перехода между уровнями был скачен ассет LeanTouch и в объекте на сцене LeanTouch были настроены

скрипты таким образом, чтобы при двойном нажатии на экран изменялось состояние кнопки перехода.

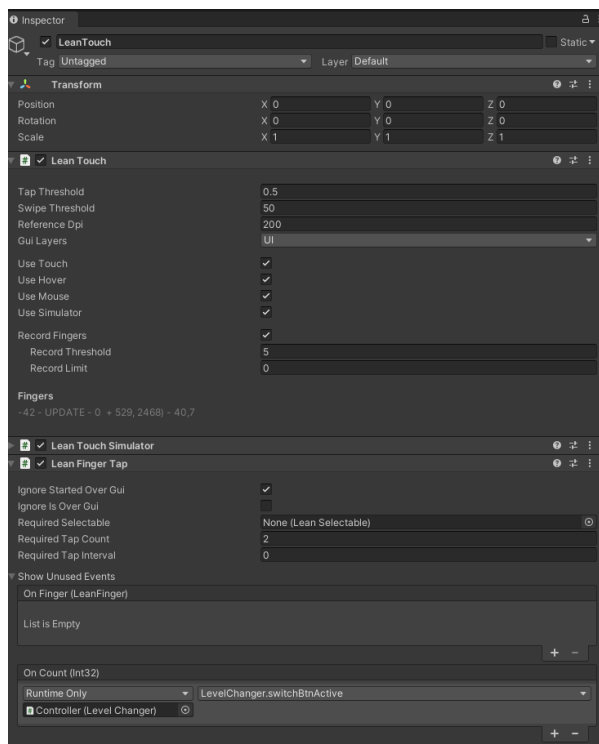


Рисунок 27 – Настройки объекта LeanTouch

4 РЕЗУЛЬТАТЫ РАЗРАБОТКИ ПРИЛОЖЕНИЯ

При первом запуске приложения, при отсутствии необходимых разрешений, приложение будет производить запрос на их использование.



Рисунок 28 – Запрос необходимых разрешений

Пример работы приложения в картографическом режиме приведен на рисунке 29. В данном режиме имеется возможность построения маршрута от пользователя до заданной точки. Также в нем отображаются все созданные объекты по их геокоординатам.

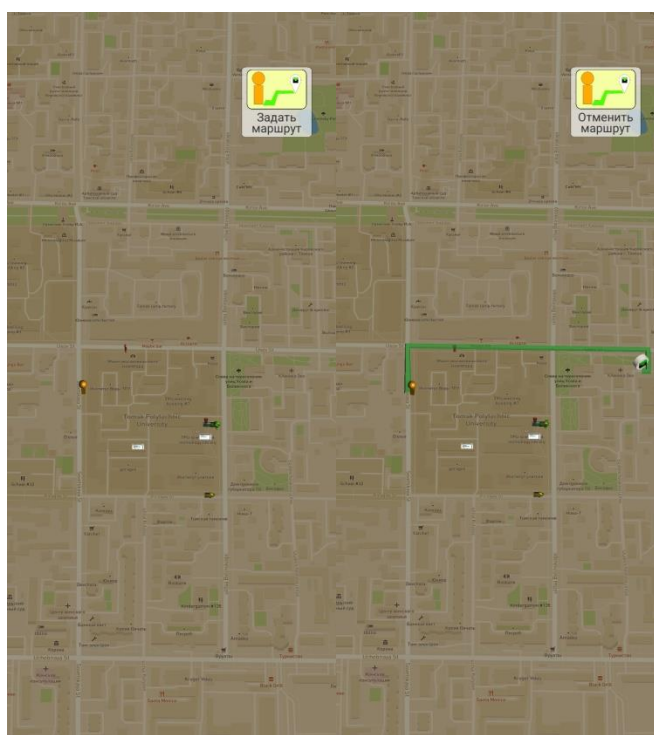


Рисунок 29 – Пример использование картографического режима

В AR режиме при задании маршрута появится вспомогательная стрелка, которая будет указывать на положение заданной точки при построении маршрута.



Рисунок 30 – Пример добавления нового объекта в AR-режиме

На рисунке 31 отражен процесс добавления нового объекта.

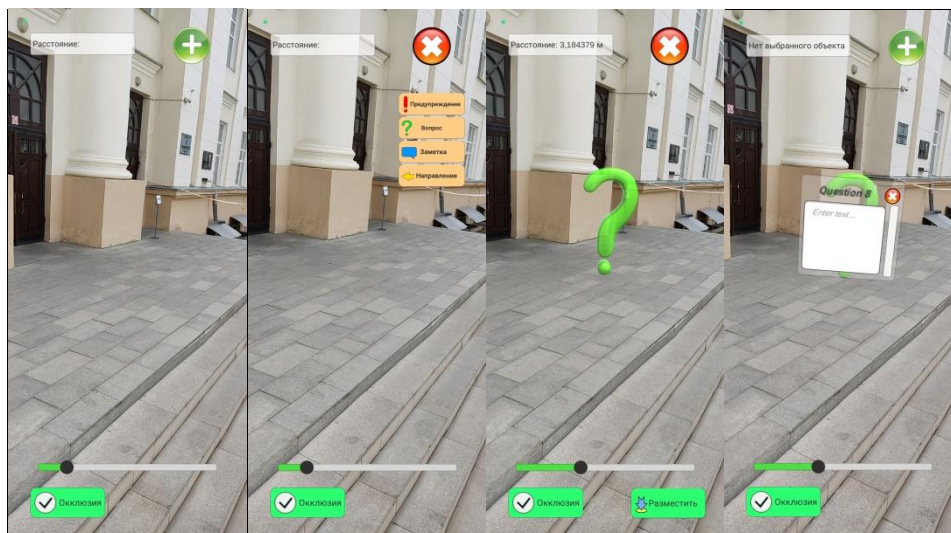


Рисунок 31 – Пример добавления нового объекта в AR-режиме

На рисунке 31 показан пример, в котором часть размещаемого объекта по расстоянию удаления от пользователя находится дальше, чем объект из

реального мира. За счет работы Occlusion та часть объекта, которая находится за реальным объектом, скрывается.

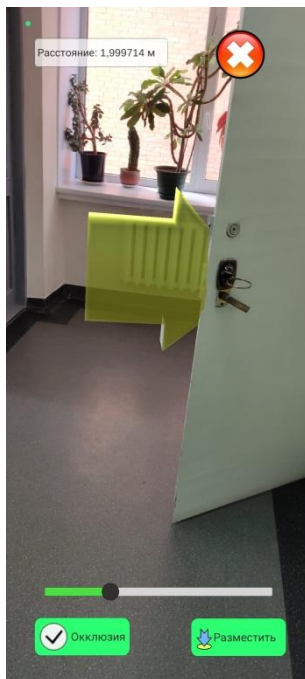


Рисунок 31 – Пример работы Occlusion в AR-режиме

В рамках дальнейшего развития для улучшения навигации студентов планируется добавить навигационные объекты к каждой учебной аудитории.

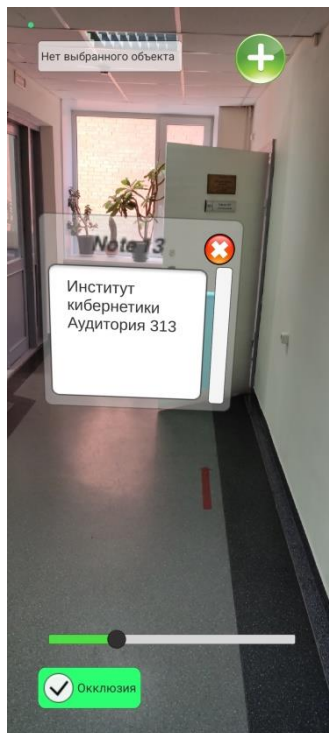


Рисунок 32 – Пример навигационного объекта у учебной аудитории

5 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

Введение

Целью данной работы является разработку программного обеспечения (ПО), которое позволит создавать систему навигационных объектов в дополненной реальности для ориентирования в пределах ТПУ.

Целью раздела является определение эффективности научно-исследовательского проекта.

Данный раздел, предусматривает рассмотрение следующих задач:

- Разработка общей экономической идеи проекта, формирование концепции проекта;
- Организация работ по научно-исследовательскому проекту;
- Определение возможных альтернатив проведения научных исследований;
- Планирование научно-исследовательских работ;
- Оценки коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения;
- Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования.

5.1 Предпроектный анализ

5.1.1 Потенциальные потребители результатов исследования

Выполненная работа направлена на разработку и реализацию мобильного приложения для навигации с использованием технологий дополненной реальности, которое позволит владельцу создать систему виртуальных объектов, которая облегчит пользователям ориентирование в пределах организации владельца.

Исходя из особенностей мобильного приложения с поддержкой дополненной реальности, можно судить о круге лиц, которые потенциально будут заинтересованы в разработке. Целевым рынком нынешней разработки являются компании, имеющие большую организационную структуру и стремящиеся к расширению своих возможностей в направлении современных трендов дополненной реальности, которые могут отразить определенную дополнительную информацию. Сегментировать рынок услуг можно по виду данной информации. Результат сегментирования представлен на таблице 1.

В роли заказчика данного приложения могут выступать организации из различных отраслей. Например, в сфере культуры это могут быть музеи, библиотеки, в образовании – вузы, в транспортной сфере – авиа- и ж/д вокзалы, метро.

Таблица 2 – Карта сегментирования рынка разработки

		Вид информации		
		Дополнительная информация об объектах/окружении	Структура организации	Направляющая информация
Отрасль	Образование			
	Транспорт			
	Культура			

Фирма А – ; Фирма Б – ; Фирма В – ;

Исходя из таблицы 2, можно сделать вывод, что основной вид информации, который необходим большинству заказчиков – направляющая информация, которая поможет пользователям добраться до необходимого им места.

5.1.2 Анализ конкурентных технических решений с позиции ресурсоэффективности и ресурсосбережения

Анализ конкурирующих разработок помогает вносить коррективы в научное исследование, чтобы успешнее противостоять своим конкурентам. Важно реалистично оценить сильные и слабые стороны ближайших аналогов.

В качестве конкурентных решений разрабатываемому приложению можно рассматривать как крупные картографические приложения, такие как Яндекс.Карты, Google Карты и Apple Maps, которые за последние годы расширили набор возможностей своих приложений и добавили в них возможность навигации при помощи дополненной реальности, в которой указатели направления движения добавляются посредством работы алгоритмов, которые обрабатывают заданные пользователем конечные точки маршрута.

Для сравнения разрабатываемого приложения (AN) будут рассмотрены Google Карты (GK), Яндекс.Карты (ЯК) и Apple Maps (AM).

Таблица 3 – Таблица сравнения приложений

	AN	GK	ЯК	AM
Платформы (все, не только с AR)	Android 7.0+	Android 7.0+, iOS 11+, Windows 7+, Linux	iOS 11+, Windows	iOS 11+
Число поддерживаемых языков	2	74	6	34
Территориальные ограничения	нет	Рим, Милан, Амстердам, Бангкок, Барселона, Берлин, Будапешт, Дубай, Стамбул, Куала-Лумпур, Киото, Лондон, Лос-Анджелес, Мадрид, Мюнхен, Нью-Йорк, Осака, Париж, Прага, Сан-Франциско, Сидней, Токио и Вена	нет	Лондон, Лос-Анджелес, Нью-Йорк, Филадельфию, Сан-Диего, Сан-Франциско и Вашингтон
Возможность	нет	Имеется функция	нет	Имеется

уточнения местоположения		сканирования, которая уточняет местоположение пользователя		функция сканирования, которая уточняет местоположение пользователя
Платформа дополненной реальности	Lightship ARDK	AR Core, ARKit	ARKit	ARKit

Для оценки сравнительной эффективности научной разработки с ее конкурентами был проведен анализ с помощью оценочной карты (таблица 4) на основе таблицы 3 и некоторых других введенных критериев для сравнения приложений.

Таблица 4 – Оценочная карта для сравнения конкурентных технических решений (разработок)

Критерий оценки	Вес критерия	Баллы				Конкурентоспособность			
		Б _{АН}	Б _{ГК}	Б _{ЯК}	Б _{АМ}	К _{АН}	К _{ГК}	К _{ЯК}	К _{АМ}
Технические критерии оценки ресурсоэффективности									
Удобство в эксплуатации	0,2	7	9	8	9	1,4	1,8	1,6	1,8
Число языков	0,05	2	10	4	8	0,1	0,5	0,2	0,4
Территориальные ограничения	0,15	10	6	10	5	1,5	0,9	1,5	0,75
Точность определения местоположения	0,15	7	9	7	9	1,05	1,35	1,05	1,35
Платформа дополненной реальности	0,1	9	8	7	7	0,9	0,8	0,7	0,7
Экономические критерии оценки эффективности									
Уровень проникновения на рынок	0,1	0	10	8	7	0	1	0,8	0,7
Финансирование разработки	0,1	1	10	8	10	0,1	1	0,8	1
Сопровождение	0,15	4	9	7	9	0,6	1,35	1,05	1,35
Итого	1					5,65	8,7	7,7	8,05

Исходя из проведенного анализа, можно заключить, что разрабатываемое приложение, сильно уступают по экономическим критериям крупным компаниям, т.к. их финансовые возможности и профессиональный коллектив явно будет лучше, чем у разрабатываемого приложения. Если рассматривать только технические критерии, то можно увидеть направления, в которых стоит в дальнейшем развивать проект.

Таким образом, можно прийти к выводу, что если расширить командный состав, финансирование и функционал, то создаваемое приложение может быть вполне конкурентоспособным.

5.1.3 SWOT–анализ

Следующим этапом является комплексный анализ внешней и внутренней среды проекта с помощью технологии SWOT, который проводится в несколько шагов.

Первый этап заключается в описании сильных и слабых сторон проекта, в выявлении возможностей и угроз для реализации проекта, которые проявились или могут появиться в его внешней среде.

Результаты проведенного первого этапа SWOT-анализа представлены в таблице 5.

Таблица 5 – Сильные и слабые стороны проекта

Возможности:	Угрозы:	Сильные стороны	Слабые стороны
<p>В1. Дальнейшее расширение функционала приложения</p> <p>В2. Расширение платформ приложения (добавить разработку под iOS)</p> <p>В3. Привлечение дополнительных заказчиков из</p>	<p>У1. Отсутствие спроса на продукт на рынке.</p> <p>У2. Развитие и появление аналогов приложения</p>	<p>Сильные стороны научно-исследовательского проекта:</p> <p>С1. Простота разработки основных компонентов приложения</p> <p>С2. Открытый код использованных средств разработки</p> <p>С3. Отображение виртуальных объектов с</p>	<p>Слабые стороны научно-исследовательского проекта:</p> <p>Сл1. Не проработан вопрос, связанный с уточнением местоположения пользователя в зданиях</p> <p>Сл2. Отсутствие подсказок для пользователя по использованию</p>

других сфер		использованием Occlusion для многих устройств	приложения Сл3. Проблема с сохраняемыми текстовыми данными на кириллице из-за кодировок
-------------	--	---	---

После того как сформулированы четыре области SWOT переходят к реализации второго этапа. Необходимо построить интерактивную матрицу проекта. Ее использование помогает разобраться с различными комбинациями взаимосвязей областей матрицы SWOT (таблица 6).

Таблица 6 – Интерактивная матрица проекта

Сильные стороны				
Возможности		С1	С2	С3
	В1	+	+	
	В2	+		+
	В3	+		
Слабые стороны				
Возможности		Сл1	Сл2	Сл3
	В1	+	+	
	В2			
	В3	+	+	+
Сильные стороны				
Угрозы		С1	С2	С3
	У1	+		
	У2			+
Слабые стороны				
Угрозы		Сл1	Сл2	Сл3
	У1	+	+	
	У2			

В рамках третьего этапа составляется итоговая матрица SWOT-анализа, в которую добавляются сильно коррелирующие стороны проекта. Итоговая матрица приведена в таблице 7.

Таблица 7 – Итоговая матрица SWOT

	Сильные стороны научно-	Слабые стороны научно-исследовательского
--	-------------------------	--

	<p>исследовательского проекта:</p> <p>С1. Простота разработки основных компонентов приложения</p> <p>С2. Открытый код использованных средств разработки</p> <p>С3. Отображение виртуальных объектов с использованием Occlusion для многих устройств</p>	<p>проекта:</p> <p>Сл1. Не проработан вопрос, связанный с уточнением местоположения пользователя в зданиях</p> <p>Сл2. Отсутствие подсказок для пользователя по использованию приложения</p> <p>Сл3. Проблема с сохраняемыми текстовыми данными на кириллице из-за кодировок</p>
<p>Возможности:</p> <p>В1. Дальнейшее расширение функционала приложения</p> <p>В2. Расширение платформ приложения (добавить разработку под iOS)</p> <p>В3. Привлечение дополнительных заказчиков из других сфер</p>	<p>С1В1В2В3 Благодаря относительной простоте разработки возможно дальнейшее развитие и расширение проекта. Путем включения функции прокладки маршрута из точки А в точку Б</p>	<p>В3Сл1Сл2Сл3</p> <p>Привлечение других разработчиков позволит разрешить текущие проблемы проекта и освоить большее количество заказов</p>
<p>Угрозы:</p> <p>У1. Отсутствие спроса на продукт на рынке.</p> <p>У2. Развитие и появление аналогов приложения</p>	<p>С1У1 Простота разработки не всегда способствует большому спросу на рынке</p> <p>С3У2 Конкуренты могут воспользоваться теми же средствами для разработки дополненной реальности</p>	<p>У1Сл1Сл2 Доработка пользовательского интерфейса и точности местоположения может поднять спрос на рынке</p>

5.1.4 Оценка готовности проекта к коммерциализации

Для оценки готовности проекта коммерциализации был проведен анализ степени проработанности проекта с позиции коммерциализации и

компетенциям разработчика научного проекта. Перечень вопросов приведен в таблице 8.

Таблица 8 – Бланк оценки степени готовности научного проекта коммерциализации

№ п/п	Наименование	Степень проработанности научного проекта	Уровень имеющихся знаний у разработчика
1	Определен имеющийся научно-технический задел	3	4
2	Определены перспективные направления коммерциализации научно-технического задела	2	3
3	Определены отрасли и технологии (товары и услуги) для предложения на рынке	3	3
4	Определена товарная форма научно-технического задела для представления на рынок	1	1
5	Определены авторы и осуществлена охрана их прав	2	2
6	Проведена оценка стоимости интеллектуальной собственности	1	1
7	Проведены маркетинговые исследования рынка сбыта	2	1
8	Разработан бизнес-план коммерциализации научной разработки	1	1
9	Определены пути продвижения научной разработки на рынок	1	1
10	Разработана стратегия (форма) реализации научной разработки	1	1
11	Проработаны вопросы международного сотрудничества и выхода на зарубежный рынок	1	1
12	Проработаны вопросы использования услуг инфраструктуры поддержки, получения льгот	1	1
13	Проработаны вопросы финансирования	1	2

	коммерциализации научной разработки		
14	Имеется команда для коммерциализации научной разработки	1	1
15	Проработан механизм реализации научного проекта	2	2
	Итого баллов	23	25

Исходя из полученных сумм значений было выявлено, что перспективность проекта ниже среднего.

5.1.5 Методы коммерциализации результатов научно-технического исследования

Продвижение товара на рынок во многом зависит от правильности выбора метода коммерциализации. Всего существует восемь методов коммерциализации: торговля патентными лицензиями, передача ноу-хау, инжиниринг, франчайзинг, организация собственного предприятия, передача интеллектуальной собственности, организация совместного предприятия, организация совместных предприятий с российским производством и зарубежным распространением.

Из представленных методов к разрабатываемому приложению больше всего подойдет инжиниринг в связи с тем, что оно связано с проектированием объекта, разработкой новых технологических процессов на предприятии заказчика и предоставлением ряда инженерно-технических услуг заказчику. На данной стадии проекта это будет лучшим выбором, т.к. имея потенциального заказчика, можно будет доработать приложение по его определенным требованиям, что, несомненно, лучше, чем разработка того, что может в дальнейшем не потребоваться.

5.2 Инициация проекта

5.2.1 Цели и результат проекта

Перед определением целей необходимо перечислить заинтересованные стороны проекта. Информация по заинтересованным сторонам представлена в таблице 9.

Таблица 9 – Потенциально заинтересованные стороны проекта

Заинтересованные стороны проекта	Ожидания заинтересованных сторон
Пользователь приложения	Упрощение навигации в пределах организации
Организация-заказчик	Повышение эффективности процессов внутри организации, а также их оптимизация
Разработчик	Получение заработной платы
Студент	Выполненная выпускная квалификационная работа
Научный руководитель	Выполнение одного из показателей эффективного контракта

Цели и результаты проекта представлены в таблице 10.

Таблица 10 – Цели и результат проекта

Цели проекта:	<ul style="list-style-type: none">• Спроектировать функционал приложения в соответствии с требованиями заказчика• Разработать техническое задание• Разработать прототип приложения• Произвести тестирование приложения• Внедрить разработку и создать систему навигационных объектов в рамках ТПУ
Ожидаемые результаты проекта:	<ul style="list-style-type: none">• Успешно внедренная разработка у организации заказчика• Сдана выпускная квалификационная работа
Критерии приемки проекта:	<ul style="list-style-type: none">• Успешное тестирование функционала приложения в соответствии с техническим заданием
Требования к результату проекта:	Требование:

	<ul style="list-style-type: none"> • Выполнены все пункты технического задания
	<ul style="list-style-type: none"> • Разработанное приложение соответствует требованиям заказчика

5.2.2 Организационная структура проекта

Организационная структура проекта представлена в таблице 11.

Таблица 11 – Рабочая группа проекта

ФИО, основное место работы, должность	Роль в проекте	Функции
Сергиенко Александр Романович, студент	Разработчик, исполнитель	<ul style="list-style-type: none"> • Поиск решения для выполнения поставленной задачи • Проектирование архитектуры • Реализация • Тестирование
Лоскутов Виталий Валерьевич, старший преподаватель ОИТ	Консультант	<ul style="list-style-type: none"> • Постановка задачи • Проверка разработки • Помощь во внедрении
Демин Антон Юрьевич, к.т.н., доцент ОИТ	Руководитель проекта	<ul style="list-style-type: none"> • Составление научных целей и задач • Проверка документации

5.3 Планирование управления научно-техническим проектом

5.3.1 Структура работ в рамках научного исследования

Трудоемкость выпускной квалификационной работы определяется опытным путем в человеко-днях и имеет вероятностный характер, так как зависит от трудно учитываемых факторов. Планирование комплекса предполагаемых работ осуществляется в следующем порядке:

- определение структуры работ в рамках научного исследования;
- определение участников каждой работы;

- установление продолжительности работ;
- построение графика проведения научных исследований.

В качестве исполнителей в проекте представлены студент С и научный руководитель НР. Перечень работ и исполнителей приведен в таблице 12.

Таблица 12 – Основные этапы выполнения ВКР

№	Описание этапа	Исполнители	Загруженность исполнителей
1	Составление и утверждение задания	С, НР	С – 20%, НР – 80%
2	Анализ предметной области	С, НР	С – 50%, НР – 50%
3	Разработка календарного плана	С, НР	С – 10%, НР – 90%
4	Обзор литературы и интернет - источники	С, НР	С – 60%, НР – 40%
5	Выбор программного обеспечения, библиотек, платформы разработки	С, НР	С – 70%, НР – 30%
6	Разработка интерфейса режима дополненной реальности с добавлением объектов	С, НР	С – 90%, НР – 10%
7	Разработка картографического режима для навигации	С, НР	С – 80%, НР – 20%
8	Добавление функционала по определению и записи геоординат создаваемых объектов	С, НР	С – 70%, НР – 30%
9	Реализация воссоздания ранее созданных объектов на тех же местах и отображение их в картографическом режиме	С	С – 100%
10	Подключение сервера для хранения, получения новых записей и передачи имеющихся данных	С, НР	С – 70%, НР – 30%
11	Расчет экономической эффективности научно-технической продукции	С	С – 100%
12	Оценка социальной ответственности проекта	С	С – 100%
13	Написание пояснительной записки	С, НР	С – 90%, НР – 10%

Как видно из представленной выше таблицы, роль НР в ходе выполнения всех видов работ сводится к направляющей и контролирующей

функциям, в то время как С является исполнительным звеном – он выполняет все поставленные задачи для достижения установленной цели.

5.3.2 Определение трудоемкости выполнения работ

Трудовые затраты в большинстве случаев образуют основную часть стоимости разработки, поэтому важным моментом является определение трудоемкости работ каждого из участников научного исследования.

Трудоемкость выполнения научного исследования оценивается экспертным путем в человеко-днях и носит вероятностный характер, т.к. зависит от множества трудно учитываемых факторов. Для определения ожидаемого (среднего) значения трудоемкости $t_{ож}$ используется следующая формула:

$$t_{ож} = \frac{3 \cdot t_{min} + 2 \cdot t_{max}}{5}, \quad (3)$$

где t_{min} – предположительно минимальная продолжительность этапа в рабочих днях, определяемая методом экспертной оценки;

t_{max} – предположительно максимальная продолжительность этапа в рабочих днях, определяемая методом экспертной оценки.

Для построения линейного графика необходимо рассчитать длительность этапов в рабочих днях, а затем перевести ее в календарные дни. Расчет продолжительности выполнения каждого этапа в рабочих днях ($T_{РД}$) ведется по формуле:

$$T_{РД} = t_{ож} \cdot K_{д}, \quad (4)$$

где $t_{ож}$ – продолжительность работы, дн.;

$K_{д}$ – коэффициент, учитывающий дополнительное время на консультации и согласование работ, в данном случае $K_{д} = 1,2$.

Расчет продолжительности этапа в календарных днях ведется по формуле:

$$T_{КД} = T_{РД} \cdot T_{к}, \quad (5)$$

где $T_{\text{КД}}$ – продолжительность выполнения этапа в календарных днях;

$T_{\text{К}}$ – коэффициент календарности, позволяющий перейти от длительности работ в рабочих днях к их аналогам в календарных днях, и рассчитываемый по формуле:

$$T_{\text{К}} = \frac{T_{\text{КАЛ}}}{T_{\text{КАЛ}} - T_{\text{ВД}} - T_{\text{ПД}}}, \quad (6)$$

где $T_{\text{КАЛ}}$ – календарные дни ($T_{\text{КАЛ}} = 365$);

$T_{\text{ВД}}$ – выходные дни ($T_{\text{ВД}} = 52$);

$T_{\text{ПД}}$ – праздничные дни ($T_{\text{ПД}} = 14$).

$$T_{\text{К}} = \frac{365}{365 - 52 - 14} \approx 1,221$$

В таблице 13 приведены продолжительности этапов работ и их трудоемкости по исполнителям, занятым на каждом этапе. В столбцах (3–5) реализован экспертный способ по формуле (3). Столбцы 6 и 7 содержат величины трудоемкости этапа для каждого из участников проекта (научный руководитель, и студент) с учетом коэффициента $K_{\text{Д}} = 1$. Каждое из них в отдельности не может превышать соответствующее значение $t_{\text{ож}} * K_{\text{Д}}$. Столбцы 8 и 9 содержат те же трудоемкости, выраженные в календарных днях путем дополнительного умножения на $T_{\text{К}}$. Итог по столбцу 5 дает общую ожидаемую продолжительность работы над проектом в рабочих днях, итоги по столбцам 6 и 7 – общие трудоемкости для каждого из участников проекта в рабочих днях. Величины трудоемкости этапов по исполнителям $T_{\text{КД}}$ (данные столбцов 8 и 9) далее будут использованы для определения затрат на оплату труда участников и прочие затраты.

Исходя из таблицы 13, была построена диаграмма Ганта (таблица 14), в которой отражен линейный график работ.

Таблица 13 – Трудозатраты на выполнение проекта

Этап	Исполнители	Продолжительность работ, дни			Трудоемкость работ по исполнителям чел.- дн.			
		t_{min}	t_{max}	$t_{ож}$	$T_{РД}$		$T_{КД}$	
					НР	С	НР	С
1	2	3	4	5	6	7	8	9
Составление и утверждение задания	С, НР	2	5	3,2	3,072	0,768	3,751	0,938
Анализ предметной области	С, НР	5	7	5,8	3,48	3,48	4,249	4,249
Разработка календарного плана	С, НР	2	4	2,8	3,024	0,336	3,692	0,41
Обзор литературы и интернет - источников	С, НР	7	10	8,2	3,936	5,904	4,806	7,209
Выбор программного обеспечения, библиотек, платформы разработки	С, НР	3	7	4,6	1,656	3,864	2,022	4,718
Разработка интерфейса режима дополненной реальности с добавлением объектов	С, НР	7	15	10,2	1,224	11,016	1,495	13,451
Разработка картографического режима для навигации	С, НР	10	15	12	2,88	11,52	3,516	14,066
Добавление функционала по определению и записи геокординат создаваемых объектов	С, НР	7	10	8,2	2,952	6,888	3,604	8,41
Реализация воссоздания ранее созданных объектов на тех же местах и отображение их в картографическом режиме	С	12	20	15,2	-	18,24	-	22,271
Подключение сервера для хранения, получения новых записей и передачи имеющихся данных	С, НР	15	25	19	6,84	15,96	8,352	19,487
Расчет экономической эффективности научно-технической продукции	С	5	7	5,8	-	6,96	-	8,498
Оценка социальной ответственности проекта	С	5	7	5,8	-	6,96	-	8,498
Написание пояснительной записки	С, НР	10	14	11,6	1,392	12,528	1,7	15,297
Итого				112,4	42,984	104,424	54,484	127,504

Таблица 14 – Календарный план-график проведения научно-исследовательской работы по теме

Этап	Исполнители	Т _{КД}	Продолжительность выполнения работ													
			февраль		март			апрель			май			июнь		
			2	3	1	2	3	1	2	3	1	2	3	1	2	
1	С, НР	0,938, 3,751	■													
2	С, НР	4,249, 4,249		■												
3	С, НР	0,41, 3,692			■											
4	С, НР	7,209, 4,806				■										
5	С, НР	4,718, 2,022					■									
6	С, НР	13,451, 1,495						■								
7	С, НР	14,066, 3,516							■							
8	С, НР	8,41, 3,604								■						
9	С	22,271														
10	С, НР	19,487, 8,352														
11	С	8,498														
12	С	8,498														
13	С, НР	15,297, 1,7														

НР – ■ С – ■

5.4 Бюджет научного исследования

В данном разделе будет произведен расчет затрат на разработку (*p*) проекта, а также двух вариантов его исполнения. Оба из них будут нацелены на улучшение выходного продукта, т.к. на текущем этапе проекта приложение можно дополнить различными дополнительными функциями. В первом варианте (*a1*) будет рассматриваться привлечение дополнительного инженера-специалиста для доработки приложения. Во втором варианте (*a2*) будет увеличено общее время на разработку приложения.

5.4.1 Сырье, материалы, покупные изделия и полуфабрикаты (за вычетом отходов)

К данной статье расходов относится стоимость материалов, покупных изделий, полуфабрикатов и других материальных ценностей, расходуемых непосредственно в процессе выполнения работ над объектом проектирования.

Расчет материальных затрат осуществляется по следующей формуле:

$$Z_m = (1 + k_T) \cdot \sum_{i=1}^m C_i \cdot N_{рас\ xi}, \quad (7)$$

где m – количество видов материальных ресурсов, потребляемых при выполнении научного исследования; $N_{рас\ xi}$ – количество материальных ресурсов i -го вида, планируемых к использованию при выполнении научного исследования (шт., кг, м, м² и т.д.); C_i – цена приобретения единицы i -го вида потребляемых материальных ресурсов (руб./шт., руб./кг, руб./м, руб./м² и т.д.); k_T – коэффициент, учитывающий транспортно-заготовительные расходы.

Материальные затраты, необходимые для данной разработки, заносятся в таблицу 15.

Таблица 15 – Сырье, материалы, комплектующие изделия и покупные полуфабрикаты

Наименование материалов	Цены за ед., руб.	Кол-во			Сумма (<i>p</i>), руб		
		<i>p</i>	<i>a1</i>	<i>a2</i>	<i>p</i>	<i>a1</i>	<i>a2</i>
Бумага для принтера, А4	597	1 уп.	2 уп.	1 уп.	597	1194	597
Ручка шариковая	50	2 шт.	4 шт.	2 шт.	100	200	100
Картридж для	897	1 шт.	2 шт.	1 шт.	897	1794	897

принтера							
Всего за материалы				1594	3188	1594	
Транспортно-заготовительные расходы (3-5%)				5%			
Итого по статье				1673,7	3347,4	1673,7	

5.4.2 Расчет затрат на электроэнергию

Данный вид расходов включает в себя затраты на электроэнергию, потраченную в ходе выполнения проекта на работу используемого оборудования, рассчитываемые по формуле:

$$C_{\text{эл.об.}} = P_{\text{об.}} \cdot t_{\text{об.}} \cdot \text{Ц}_{\text{э}} \cdot n, \quad (8)$$

где $P_{\text{об}}$ – мощность, потребляемая оборудованием, кВт;

$\text{Ц}_{\text{э}}$ – тариф на 1 кВт·час;

$t_{\text{об}}$ – время работы оборудования, час;

n – количество единиц оборудования

Для ТПУ $\text{Ц}_{\text{э}} = 3,85$ руб./кВт·час.

Время работы оборудования вычисляется на основе итоговых данных таблицы 10 для инженера ($T_{\text{рд}}$) из расчета, что продолжительность рабочего дня равна 8 часов.

$$t_{\text{об}} = T_{\text{рд}} \cdot K_t, \quad (9)$$

где $K_t \leq 1$ – коэффициент использования оборудования по времени, равный отношению времени его работы в процессе выполнения проекта к $T_{\text{рд}}$, в данном случае приравнивается 0,9.

Мощность, потребляемая оборудованием, определяется по формуле:

$$P_{\text{об}} = P_{\text{ном.}} \cdot K_c, \quad (10)$$

где $P_{\text{ном.}}$ – номинальная мощность оборудования, кВт;

$K_c \leq 1$ – коэффициент загрузки, зависящий от средней степени использования номинальной мощности. Для технологического оборудования малой мощности $K_c = 1$.

В таблице 16 приведены расчеты затрат на электроэнергию.

Таблица 16 – Затраты на электроэнергию технологическую

Вариант исполнения	Наименование оборудования	Время работы оборудования $t_{об}$, час	Потребляемая мощность $P_{об}$, кВт	Кол-во п, ед.	Затраты $C_{эл.об.}$, руб.
<i>p</i>	Персональный компьютер	$104,424 \cdot 0,9$	0,282	1	102
<i>a1</i>	Персональный компьютер	$104,424 \cdot 0,9$	0,282	2	204
<i>a2</i>	Персональный компьютер	$134,424 \cdot 0,9$	0,282	1	131

5.4.3 Специальное оборудование для научных (экспериментальных) работ

В данную статью включают все затраты, связанные с приобретением специального оборудования (приборов, контрольно-измерительной аппаратуры, стендов, устройств и механизмов), необходимого для проведения работ по конкретной теме. Определение стоимости спецоборудования производится по действующим прейскурантам, а в ряде случаев по договорной цене. Для реализации проекта было использовано оборудование, затраты на которое приведены в таблице 17.

Таблица 17 – Расчет затрат по статье «Спецоборудование для научных работ»

Наименование оборудования	Количество единиц оборудования			Цена за 1 ед. оборудования, тыс. руб.	Общая стоимость оборудования, тыс. руб.		
	<i>p</i>	<i>a1</i>	<i>a2</i>		<i>p</i>	<i>a1</i>	<i>a2</i>
Персональный компьютер	1	2	1	100	100	200	100
Монитор	2	4	2	15	30	60	30
Клавиатура	1	2	1	2	2	4	2
Компьютерная мышь	1	2	1	1	1	2	1
Итого					133	266	133

5.4.4 Основная заработная плата

Статья включает основную заработную плату работников, непосредственно занятых выполнением НИИ, и дополнительную заработную плату:

$$Z_{\text{зп}} = Z_{\text{осн}} + Z_{\text{доп}}, \quad (11)$$

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

$Z_{\text{доп}}$ – дополнительная заработная плата (15% от основной).

Основная заработная плата руководителя (лаборанта, инженера) от предприятия рассчитывается по следующей формуле:

$$Z_{\text{осн}} = Z_{\text{дн}} \cdot T_p, \quad (12)$$

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

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

T_p – продолжительность работ, выполняемых работником, раб.дн.

Среднедневная зарплата рассчитывается по формуле:

$$Z_{\text{дн}} = \frac{Z_m \cdot M}{F_d}, \quad (13)$$

где Z_m – месячный должностной оклад работника, руб.;

M – количество месяцев работы без отпуска в течение года: при отпуске в 48 раб. дней $M = 10,4$ месяца, 6-дневная неделя;

F_d – действительный годовой фонд рабочего времени научно-технического персонала, раб. дн. (таблица 18).

Таблица 18 – Баланс рабочего времени

Показатели рабочего времени	Руководитель	Инженер-исследователь
Календарное число дней	365	365
Количество нерабочих дней (выходные и праздничные дни)	52 и 14	52 и 14
Потери рабочего времени (отпуск, больничные), дни	48	48
Действительный годовой фонд	251	251

рабочего времени, дни		
-----------------------	--	--

Месячный должностной оклад работника:

$$Z_M = Z_6 \cdot k_p, \quad (14)$$

где Z_6 – базовый оклад, руб.;

k_p – районный коэффициент, равный 1,3 для Томска.

Таким образом месячный должностной оклад руководителя (старшего преподавателя) составляет $28 \cdot 1,3 = 36,4$ тыс. руб, а студента (инженер-исследователь) – $23,8 \cdot 1,3 = 30,94$ тыс.руб.

Среднедневная зарплата руководителя составляет:

$$Z_{\text{дн}} = \frac{36,4 \cdot 10,4}{251} \approx 1,51 \text{ тыс. руб.}$$

Среднедневная зарплата студента составляет:

$$Z_{\text{дн}} = \frac{30,94 \cdot 10,4}{251} \approx 1,28 \text{ тыс. руб.}$$

Исходя из полученных значений среднедневных зарплат исполнителей, был произведен расчет их основных заработных плат (таблица 1).

Таблица 19 – Расчет основной заработной платы

Вариант исполнения	Исполнители по категориям	Общая трудоемкость, чел.-дн	Заработная плата, приходящаяся на один чел.-дн., тыс. руб.	Всего заработная плата по тарифу (окладам), тыс. руб.	Итого, тыс. руб.
р	Руководитель	43	1,51	64,93	198,05
	Инженер-исследователь	104	1,28	133,12	
а1	Руководитель	43	1,51	64,93	331,17
	Инженер-исследователь	104	1,28	133,12	

	Инженер-исследователь	104	1,28	133,12	
a2	Руководитель	43	1,51	64,93	236,45
	Инженер-исследователь	134	1,28	171,52	

Затраты по дополнительной заработной плате исполнителей темы учитывают величину предусмотренных Трудовым кодексом РФ доплат за отклонение от нормальных условий труда, а также выплат, связанных с обеспечением гарантий и компенсаций (при исполнении государственных и общественных обязанностей, при совмещении работы с обучением, при предоставлении ежегодного оплачиваемого отпуска и т.д.).

Расчет дополнительной заработной платы ведется по следующей формуле:

$$Z_{\text{доп}} = k_{\text{доп}} \cdot Z_{\text{осн}}, \quad (15)$$

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

$k_{\text{доп}}$ – коэффициент дополнительной заработной платы на стадии проектирования, принимается равным 15%.

Итоговые значения заработных плат исполнителей по вариантам исполнения приведены в таблице 20.

Таблица 20 – Затраты по основным заработным платам

Вариант исполнения	Исполнитель	Основная заработная плата $Z_{\text{осн}}$, тыс. руб	Дополнительная заработная плата $Z_{\text{доп}}$, тыс. руб	Итоговая заработная плата $Z_{\text{зп}}$, тыс. руб	Итого, тыс. руб.
p	Руководитель	64,93	9,74	74,67	227,76
	Инженер-исследователь	133,12	19,97	153,09	
a1	Руководитель	64,93	9,74	74,67	380,85
	Инженер-исследователь	133,12	19,97	153,09	

	Инженер-исследователь	133,12	19,97	153,09	
a2	Руководитель	64,93	9,74	74,67	271,92
	Инженер-исследователь	171,52	25,73	197,25	

5.4.5 Отчисления на социальные нужды

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органами 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:

$$З_{внеб} = k_{внеб} \cdot З_{ЗП}, \quad (16)$$

где $k_{внеб}$ – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.), $k_{внеб} = 30\%$.

Суммы отчислений на социальные нужды по вариантам исполнения приведены в таблице 21.

Таблица 21 – Отчисления на социальные нужды

Вариант исполнения	Затраты по заработной плате, тыс. руб.	Отчисления на социальные нужды, тыс. руб.
<i>p</i>	227,76	68,33
<i>a1</i>	380,85	114,25
<i>a2</i>	271,92	81,57

5.4.6 Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование материалов исследования, оплата услуг связи, электроэнергии, почтовые и

телеграфные расходы, размножение материалов и т.д. Их величина определяется по следующей формуле:

$$Z_{\text{накл}} = k_{\text{накл}} \cdot Z_{\text{зп}}, \quad (17)$$

где $k_{\text{накл}}$ – коэффициент накладных расходов, 16 %.

Затраты на накладные расходы по вариантам исполнения приведены в таблице 22.

Таблица 22 – Затраты на накладные расходы

Вариант исполнения	Затраты по заработной плате, тыс. руб.	Накладные расходы, тыс. руб.
<i>p</i>	227,76	36,44
<i>a1</i>	380,85	60,93
<i>a2</i>	271,92	43,51

5.4.7 Формирование бюджета затрат научно-исследовательского проекта

Рассчитанная величина затрат научно-исследовательской работы является основой для формирования бюджета затрат проекта, который при формировании договора с заказчиком защищается научной организацией в качестве нижнего предела затрат на разработку научно-технической продукции.

Определение бюджета затрат на научно-исследовательский проект приведено в таблице 23.

Таблица 23 – Расчет бюджета затрат

Наименование статьи	Сумма, тыс. руб.		
	<i>p</i>	<i>a1</i>	<i>a2</i>
Затраты на материалы	1,674	3,347	1,674
Затраты на спецоборудование	133	266	133
Затраты на электроэнергию	0,102	0,204	0,131
Затраты по основной заработной плате	227,76	380,85	271,92
Отчисления на социальные нужды	68,33	114,25	81,57
Накладные расходы	36,44	60,93	43,51
Итого:	467,306	825,58	531,8

5.5 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.

5.5.1 Оценка сравнительной эффективности исследования

Данную оценку получают в ходе оценки бюджета затрат трех (или более) вариантов исполнения научного исследования. Для этого наибольший интегральный показатель реализации технической задачи принимается за базу расчета (как знаменатель), с которым соотносятся финансовые значения по всем вариантам исполнения.

Интегральный финансовый показатель разработки определяется как:

$$I_{\Phi}^p = \frac{\Phi_{pi}}{\Phi_{max}}, \quad (18)$$

где I_{Φ}^p – интегральный финансовый показатель разработки;

Φ_{pi} – стоимость i -го варианта исполнения;

Φ_{max} – максимальная стоимость исполнения научно-исследовательского проекта (в т. ч. аналоги).

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное увеличение бюджета затрат разработки в размах (значение больше единицы), либо соответствующее численное удешевление стоимости разработки в размах (значение меньше единицы, но больше нуля).

Исходя из полученных значений, интегральный финансовый показатель разработки будет равен:

$$I_{\Phi}^p = \frac{\Phi_{pi}}{\Phi_{max}} = \frac{467,306}{825,58} = 0,57$$

$$I_{\Phi}^{a1} = \frac{\Phi_{pi}}{\Phi_{max}} = \frac{825,58}{825,58} = 1$$

$$I_{\Phi}^{a2} = \frac{\Phi_{pi}}{\Phi_{max}} = \frac{531,8}{825,58} = 0,64$$

Таким образом, значение интегрального показателя первого варианта равен единице. Данное значение получено за счет того, что затраты на материалы, электроэнергию и спецоборудование увеличатся вдвое, затраты по основной заработной плате станут равными 331,17 тыс. руб., по дополнительной 49,68 тыс. руб., на социальные нужды 114,25 тыс. руб., на накладные расходы 60,93 тыс. руб. Поэтому и бюджет исполнения увеличился, что составило 825,58 тыс. руб.

Интегральный показатель второго варианта составил 0,65, что выше интегрального показателя разработки на X. Данное превышение произошло за счет увеличения для второго варианта затрат на заработную плату – 236,45 тыс. руб., на дополнительную – 35,47 тыс. руб., на социальные нужды – 81,57 тыс. руб., на накладные расходы – 43,51 тыс. руб., на электроэнергию – 0,131 тыс. руб. Поэтому бюджет второго варианта увеличился и составил 531,8 тыс. руб.

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

$$I_m^a = \sum_{i=1}^n a_i b_i^a, \quad I_m^p = \sum_{i=1}^n a_i b_i^p, \quad (19)$$

где I_m – интегральный показатель ресурсоэффективности вариантов;

a_i – весовой коэффициент i -го параметра;

b_i^a, b_i^p – бальная оценка i -го параметра аналога и разработки, устанавливается экспериментальным путем по выбранной шкале оценивания;

n – число параметров сравнения.

Расчёт интегрального показателя ресурсоэффективности представлен в таблице 24.

Таблица 24 – Сравнительная оценка вариантов исполнения проекта

Объект исследования Критерии	Весовой коэффициент параметра	Текущий проект	Аналог 1	Аналог 2
1. Способствует росту производительности труда пользователя	0,2	4	3	3
2. Удобство в эксплуатации (соответствует требованиям потребителей)	0,2	4	5	5
3. Помехоустойчивость	0,15	3	5	4
4. Надежность	0,2	4	5	4
5. Материалоемкость, фондоемкость	0,15	5	3	4
Итого	1			

$$I_m^p = 0,2 \cdot 4 + 0,2 \cdot 4 + 0,15 \cdot 3 + 0,2 \cdot 4 + 0,15 \cdot 5 = 3,6$$

$$I_m^{a1} = 0,2 \cdot 3 + 0,2 \cdot 5 + 0,15 \cdot 5 + 0,2 \cdot 5 + 0,15 \cdot 3 = 3,8$$

$$I_m^{a2} = 0,2 \cdot 3 + 0,2 \cdot 5 + 0,15 \cdot 4 + 0,2 \cdot 4 + 0,15 \cdot 4 = 3,6$$

5.5.2 Интегральный показатель эффективности разработки

Данный показатель определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{\text{финр}}^p = \frac{I_m^p}{I_{\text{ф}}^p}, \quad I_{\text{финр}}^a = \frac{I_m^a}{I_{\text{ф}}^a} \quad (19)$$

Исходя из полученных значений, показатели эффективности разработки и аналогов будут равны:

$$I_{\text{финр}}^p = \frac{3,6}{0,57} = 6,316$$

$$I_{\text{финр}}^{a1} = \frac{3,8}{1} = 3,8$$

$$I_{\text{финр}}^{a2} = \frac{3,6}{0,64} = 5,625$$

С позиции финансовой эффективности лучшим вариантом исполнения оказался текущий вариант разработки, т.к. для его исполнения требуется меньшее количество финансовых вложений. Со стороны ресурсной эффективности лучшим вариантом получился вариант исполнения с привлечением дополнительного инженера. Это связано с тем, что благодаря его привлечению значительно увеличится объем вопросов и задач, которые могут быть решены за тот же срок, что значительно улучшит финальный вариант приложения. Интегральный показатель эффективности разработки показал, что лучшим является текущий вариант исполнения разработки. Несмотря на то, что в двух других исполнениях улучшится итоговое приложение, прирост затрат для этого превышает прирост показателей по ресурсоэффективности.

6 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

Введение

Объектом дипломной работы является мобильное приложение для навигации с помощью технологий дополненной реальности.

Основной задачей проекта является реализация возможности создания системы виртуальных объектов, которая упростит ориентирование в пределах организации, в которой будет использоваться данная система.

В ходе разработки приложения основными рабочими процессами, связанными с объектом исследования, осуществляющимися в рабочей зоне являлись: разработка приложения в среде Unity с использованием редактора Visual Studio на компьютере, тестирование приложения на мобильных устройствах.

Работа выполнялась в компьютерном классе Кибернетического центра ТПУ (первый этаж) в отделении информационных технологий. Аудитория 105 была помещением выполнения исследования. Данная аудитория представляет собой офисное отапливаемое помещение с вытяжной системой вентиляции и искусственным освещением, обеспечивающим достаточный уровень освещенности и площадью 32 м². В аудитории имеется 6 рабочих мест, каждое из которых включает в себя стул и компьютер с периферийными устройствами, расположенными на столе.

В разделе будут рассмотрены опасные и вредные факторы, оказывающие влияние на рабочую деятельность персонала, рассмотрены воздействия разрабатываемой системы на окружающую среду, правовые и организационные вопросы, а также мероприятия в чрезвычайных ситуациях.

6.1 Правовые и организационные вопросы обеспечения безопасности

Правовые и организационные вопросы задают специфичные требования для выбранного рода деятельности, а именно сфера программирования. Для учета всех требований законодательства по организации работы необходимо

провести анализ организационных мероприятий по компоновке рабочей зоны для обеспечения эффективности работы трудящихся.

6.1.1 Специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства

Работа в офисе относится ко второй категории тяжести труда – работы выполняются при оптимальных условиях внешней производственной среды и при оптимальной величине физической, умственной и нервно-эмоциональной нагрузки. Продолжительность рабочего дня работников не должна превышать 40 часов в неделю. Возможно, сокращение рабочего времени. Для работников, возраст которых меньше 16 лет – не более 24 часа в неделю, от 16 до 18 лет – не более 35 часов, как и для инвалидов I и II группы [16].

По ТК РФ №197-ФЗ [16] работник имеет право на рабочее место, отвечающее требованиям охраны труда, обязательное соц. страхование от несчастных случаев и заболеваний, связанных с производством и профессией, получение информации от работодателя и гос. органов об условиях труда на рабочем месте и возможных рисках повреждения и утраты здоровья, а также о методах защиты и предотвращение опасных производственных факторов.

6.1.2 Организационные мероприятия при компоновке рабочей зоны

В данном разделе рассмотрены основные нормативные документы, выполнение которых необходимо для безопасного исполнения рабочих обязанностей диспетчером, который проводит большую часть времени за компьютером

Рабочее место должно быть организовано с учетом эргономических требований согласно ГОСТ 12.2.032-78 «ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования» [17] и ГОСТ 12.2.061-81 «ССБТ. Оборудование производственное. Общие требования безопасности к рабочим местам» [18]. Конструкцией рабочего места должно быть обеспечено оптимальное положение работающего, которое достигается регулированием:

- Высоты рабочей поверхности, сиденья и пространства для ног.
- Высоты сиденья и подставки для ног (при нерегулируемой высоте рабочей поверхности). В этом случае высоту рабочей поверхности устанавливают по номограмме для работающего ростом 1800 мм. Оптимальная рабочая поза для работающих более низкого роста достигается за счет увеличения высоты рабочего сиденья и подставки для ног на величину, равную разности между высотой рабочей поверхности для работающего ростом 1800 мм и высотой рабочей поверхности, оптимальной для роста данного работающего.

Дисплей на рабочем месте оператора должен располагаться так, чтобы изображение в любой его части было различимо без необходимости поднять или опустить голову. Дисплей на рабочем месте должен быть установлен ниже уровня глаз оператора. Угол наблюдения экрана оператором относительно горизонтальной линии взгляда не должен превышать 60° . Освещенность рабочего места оператора на рабочем столе в горизонтальной плоскости от общего искусственного освещения должна быть от 300 до 500 лк, о чем говорится в ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения» [19].

На рабочем месте, предназначенном для работы в положении сидя в соответствии с СП 2.2.3670-20 «Санитарно-эпидемиологические требования к условиям труда» [20], производственное оборудование и рабочие столы должны иметь пространство для размещения ног высотой не менее 600 мм, глубиной – не менее 450 мм на уровне колен и 600 мм на уровне стоп, шириной не менее 500 мм.

В соответствии с государственными стандартами и правовыми нормами обеспечения безопасности предусмотрена рациональная организация труда в течение смены, которая предусматривает:

- длительность рабочей смены не более 8 часов;

- установление двух регламентируемых перерывов (не менее 20 минут после 1-2 часов работы, не менее 30 минут после 2 часов работы);
- обеденный перерыв не менее 40 минут.

6.2 Производственная безопасность при разработке проектного решения

6.2.1 Анализ выявленных вредных и опасных производственных факторов

Производственные факторы согласно ГОСТ 12.0.003-2015 [21] подразделяются на опасные и вредные. Опасным производственным фактором называется фактор, воздействие которого приводит к травме или резкому ухудшению здоровья. Вредным производственным фактором является фактор, воздействие которого приводит к заболеванию или снижению работоспособности.

На оператора ПЭВМ в течение рабочего дня воздействует множество различных производственных факторов, каждый из которых влияет на производительность, работоспособность и физическое состояние.

Возможные опасные и вредные факторы представлены в таблице 25.

Таблица 25 Возможные опасные и вредные производственные факторы

Производственный фактор	Вредный/опасный (по ГОСТ 12.0.003-2015) [21]	Нормативные документы
Производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего	Вредный	СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания [22]
Повышенный уровень локальной вибрации	Вредный	СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания [22]

		ГОСТ 12.1.012-2004 ССБТ. Вибрационная безопасность. Общие требования [23]
Отсутствие или недостаток искусственного освещения	Вредный	СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания [22] СП 52.13330.2016 Естественное и искусственное освещение [24]
Монотонность труда, вызывающая монотонию	Вредный	Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ [16]
Длительное сосредоточенное наблюдение	Вредный	ГОСТ Р 50948-2001 Средства отображения информации индивидуального пользователя. Общие эргономические требования безопасности [25]
Производственные факторы, связанные с электрическим током, вызываемым разницей электрических потенциалов, под действие которого попадает работающий	Опасный	ГОСТ 12.1.038-82 Система стандартов безопасности труда. Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов [26]

Обоснование мероприятий по защите персонала предприятия от действия опасных и вредных факторов.

6.2.2 Микроклимат рабочего помещения

Микроклимат, оказывает непосредственное воздействие на один из важнейших физиологических процессов — терморегуляцию, имеет огромное значение для поддержания комфортного состояния организма.

Пониженная температура может привести к простудным заболеваниям, высыхание слизистых оболочек. Для отвода избыточного тепла из помещений небольших размеров применяются вентиляционные установки и дефлекторы, а для охлаждения воздуха – кондиционеры. Для обеспечения нормальной температуры в зимнее время используются системы отопления.

В производственных помещениях микроклимат характеризуется температурой, влажностью, скоростью движения воздуха и давлением. Для того чтобы физиологические процессы в организме человека протекали нормально, окружающая атмосфера должна воспринимать тепло, вырабатываемое организмом. Соотношение между вырабатываемым человеком теплом и охлаждающей способностью среды, обеспечивающей сохранение нормального функционального и теплового состояния организма без напряжения терморегуляции и создающей предпосылки для нормальной работоспособности, характеризует комфортные метеорологические условия.

Исходя из СанПиН 1.2.3685-21 [22] значения температуры, влажности и скорости движения воздуха устанавливаются для рабочей зоны производственных помещений в зависимости от категории тяжести выполняемой работы, величины избытков явного тепла, выделяемого в помещении, и периода года.

В таблице 26 приведены допустимые величины показателей микроклимата на рабочих местах производственных помещений для оператора ЭВМ. В данном случае работа относится к категории труда Ib.

Таблица 26 – Допустимые величины параметров микроклимата на рабочих местах в помещениях

Период года	Температура воздуха, °С		Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	
	Диапазон ниже оптимальных величин	Диапазон выше оптимальных величин			Температура воздуха ниже оптимальных величин	Температура воздуха выше оптимальных величин
Холодный	19,0 – 20,9	23,1 – 24,0	18,0 – 25,0	15 – 75	0,1	0,2
Теплый	20,0 – 21,9	24,1 – 28,0	19,0 – 29,0	15 – 75	0,1	0,3

В учебной аудитории 105 поддерживается, согласно неоднократным замерам, проведенным в разные периоды года (теплый и холодный), 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. Данные параметры микроклимата соответствуют требованиям СанПиН 1.2.3685-21 [22].

6.2.3 Вибрация

Основу гигиенического нормирования вибрации составляют критерии здоровья человека при воздействии на него вибрации с учетом напряженности и тяжести труда. Вибрацию разграничивают на опасную и безопасную, научно обоснованные значения параметров, которой составляют гигиенические нормы вибрации.

Основная цель нормирования вибрации на рабочих местах — это установление допустимых значений характеристик вибрации, которые при ежедневном систематическом воздействии в течение всего рабочего дня и в течение многих лет не могут вызвать существенных заболеваний организма человека и не мешают его нормальной трудовой деятельности.

Применение гигиенических норм дает возможность объективно оценивать условия труда на каждом рабочем месте, определять степень виброопасности, производить выбор методов и средств виброзащиты.

Основными документами, регламентирующими уровень вибрации на рабочих местах, являются ГОСТ 12.1.012-2004 «ССБТ. Вибрационная безопасность. Общие требования» [23] и СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания» [22]. Воздействие вибрации на человека классифицируется по способу ее передачи и направлению действия. В зависимости от способа передачи вибрацию подразделяют на общую

(вибрацию рабочих мест), передающуюся через опорные поверхности на тело сидящего или стоящего человека, и локальную — передающуюся через руки (или ноги) человека.

Общая вибрация с частотой менее 0,7 Гц (качка) неприятна, но не вызывает резонансных колебаний. Резонанс человеческого тела, отдельных его органов наступает под действием внешних сил при совпадении собственных частот колебаний внутренних органов с частотами внешних сил. Для всего тела первая собственная частота лежит в диапазоне 4–6 Гц, вторая и третья (менее выраженные) — в диапазонах 10–12 Гц и 20–25 Гц соответственно. Для большинства внутренних органов собственные частоты лежат в диапазонах 6–9 Гц.

Локальная вибрация вызывает спазмы сосудов, начиная с пальцев, распространяется на всю кисть, предплечье, при этом нарушается снабжение конечностей кровью. Одновременно локальные вибрации действуют на нервные окончания, мышечные и костные ткани, что приводит к деформациям и подвижности суставов.

В аудитории, где проходил процесс разработки приложения, отсутствуют источники вибрации, таким образом, нет особой необходимости проводить соответствующие защитные мероприятия.

6.2.4 Производственное освещение

Недостаточная освещенность рабочей зоны является вредным производственным фактором, возникающим при работе с ПЭВМ. Причиной недостаточной освещенности являются недостаточность естественного освещения, недостаточность искусственного освещения, пониженная контрастность.

Работа с компьютером подразумевает постоянный зрительный контакт с дисплеем ПЭВМ и занимает от 80 % рабочего времени. Недостаточность освещения снижает производительность труда, увеличивает утомляемость и

количество допускаемых ошибок, а также может привести к появлению профессиональных болезней зрения.

Искусственное освещение в помещениях эксплуатации компьютеров должно осуществляться системой общего равномерного освещения. Местное освещение не должно создавать бликов на поверхности экрана и увеличивать освещенность экрана более 300 лк согласно СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания» [22].

Разряд зрительных работ программиста и оператора ПЭВМ относится к разряду III и подразряду г (работы высокой точности). В таблице 3 представлены нормативные показатели искусственного освещения при работах заданной точности. Требования к освещению помещений в соответствии с СП 52.13330.2016 «Естественное и искусственное освещение» [24] приведены в таблице 27.

Таблица 27 – Требования к освещению помещений промышленных предприятий для операторов ПЭВМ

Характеристика зрительной работы	Разряд зрительной работы	Подразряд зрительной работы	Контраст объекта с фоном	Характеристика фона	Искусственное освещение		
					Освещенность, лк		
					При системе комбинированного освещения		При системе от общего освещения
					всего	В том числе от общего	
Высокой точности	III	г	Средний, большой	Светлый, средний	400	200	200

6.2.5 Монотонность труда, вызывающая монотонию

Нервно-психические перегрузки – совокупность таких сдвигов в психофизиологическом состоянии организма человека, которые развиваются после совершения работы и приводят к временному снижению эффективности труда. Состояние утомления (усталость) характеризуется определенными объективными показателями и субъективными ощущениями.

Одним из примеров нервно-психических перегрузок является монотонность труда, которая может вызвать монотонию. Также к таким перегрузкам относят умственное перенапряжение, перенапряжение анализаторов и эмоциональные перегрузки.

При первых симптомах психического перенапряжения необходимо:

- дать нервной системе расслабиться;
- рационально чередовать периоды отдыха и работы;
- начать заниматься спортом;
- ложиться спать в одно и то же время;
- в тяжелых случаях обратиться к врачу.

Естественно, что полностью исключить провоцирующие факторы из жизни вряд ли удастся, но можно уменьшить их негативное воздействие, давая нервной системе необходимый отдых, который должен обеспечиваться работнику по ТК РФ №197-ФЗ [16].

6.2.6 Длительное сосредоточенное наблюдение

Работа на ПК сопровождается постоянным и значительным напряжением функций зрительного анализатора. Одной из основных особенностей является иной принцип чтения информации, чем при обычном чтении. Чтобы снизить зрительное напряжение нужно соблюдать визуальные параметры экрана (таблица 28) в соответствии с ГОСТ Р 50948-2001 «Средства отображения информации индивидуального пользователя. Общие эргономические требования безопасности» [25].

Таблица 28 – Допустимые визуальные параметры устройств отображения информации

Параметры		Допустимые значения
Яркость белого поля	Для дисплеев на ЭЛТ	Не менее 35 кд/кв.м
	Для плоских дискретных экранов	Не менее 20 кд/кв.м

Неравномерность яркости рабочего поля экрана		Не более 20%
Контрастность (для монохромного режима)		Не менее 3:1
Амплитуда смещения изображения (пространственная нестабильность изображения - дрожание)		Не более $2 \cdot 10^{-4}L$, где L – проектное расстояние наблюдения, мм
Частота обновления изображения	Для дисплеев на ЭЛТ	Не менее 75 Гц
	Для дисплеев на плоских дискретных экранах	Не менее 60 Гц

6.2.7 Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека

В деятельности организации широко используется электричество для питания компьютерной техники, которая может являться источником опасности. Несоблюдение правил ГОСТ 12.1.038-82 ССБТ. «Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов» [26] может привести к опасным последствиям. Поражение электрическим током может произойти при прикосновении к токоведущим частям, находящимся под напряжением, на которых остался заряд или появилось напряжение.

Электрический ток оказывает на человека термическое, электролитическое, биологическое и механическое воздействие. Действие электрического тока на человека приводит к травмам или гибели людей. Для переменного тока частотой 50 Гц допустимое значение напряжения прикосновения составляет 2 В, а силы тока – 0,3 мА, для тока частотой 400 Гц, соответственно – 2 В и 0,4 мА, для постоянного тока – 8 В и 1 мА.

Мерами защиты от воздействия электрического тока являются оградительные устройства, устройства автоматического контроля и сигнализации, изолирующие устройства и покрытия, устройства защитного

заземления, устройства автоматического отключения, предохранительные устройства.

6.2.8 Расчет системы воздухообмена

В жилых и общественных помещениях постоянным вредным выделением является выдыхаемый людьми углекислый газ (CO₂). Определение необходимого воздухообмена производится по количеству углекислого газа, выделяемого человеком и по его допустимой концентрации. Количество углекислого газа в зависимости от возраста человека и выполняемой работы приведены в таблице 29.

Таблица 29 Количество углекислоты, выделяемой человеком при разной работе

Возраст человека и характер работы	Количество CO ₂	
	g _{co2}	
	л/ч	г/ч
Взрослые: при физической работе	45	68
при легкой работе (в учреждениях)	23	35
в состоянии покоя	23	35
Дети до 12 лет	12	18

Таким образом, количество углекислоты, выделяемой человеком, будет равным 35 г/ч. В помещении в среднем находится 2 человека. Из этого следует, что количество CO₂, выделяемое всеми работниками будет равно:

$$G = N_{\text{людей}} \cdot g_{\text{co2}} = 2 \cdot 35 = 70 \text{ г/ч}$$

Содержание углекислоты в атмосфере для больших городов (свыше 300 тыс. жителей), x_H – 1000 мг/м³. ПДК CO₂ в воздухе рабочей зоны, x_B – 9000 мг/м³. Тогда потребный воздухообмен будет равен:

$$L = \frac{1000 \cdot G}{x_B - x_H} = \frac{1000 \cdot 70}{9000 - 1000} = 8,75 \text{ м}^3/\text{ч}$$

Объем помещения, в котором выполнялась дипломная работа, составляет $V = S \cdot h = 32 \cdot 3 = 96 \text{ м}^3$. Таким образом, кратность воздухообмена будет равна:

$$n = \frac{L}{V} = \frac{8,75}{96} \approx 0,09 \text{ ч}^{-1}$$

Однако в данной аудитории предусмотрено 6 рабочих мест. Из этого следует, что максимально возможные потребный воздухообмен и кратность воздухообмена в ней составляют:

$$L = \frac{1000 \cdot N_{\text{людей}} \cdot g_{\text{CO}_2}}{X_{\text{в}} - X_{\text{н}}} = \frac{1000 \cdot 6 \cdot 35}{9000 - 1000} = 26,25 \text{ м}^3/\text{ч}$$
$$n = \frac{\frac{1000 \cdot N_{\text{людей}} \cdot g_{\text{CO}_2}}{X_{\text{в}} - X_{\text{н}}}}{V} = \frac{\frac{1000 \cdot 6 \cdot 35}{9000 - 1000}}{96} \approx 0,27 \text{ ч}^{-1}$$

В рассматриваемой аудитории отсутствуют окна, однако имеются входная дверь и вытяжная система вентиляции. Т.к. полученное значение кратности воздухообмена $n < 1$, то вентиляция не требуется.

6.3 Экологическая безопасность при разработке проектного решения

В ходе проектной деятельности возможно вредное воздействие на литосферу, которое объясняется из-за утилизации отходов, связанных с выходом из строя ПК и периферийных устройств (принтеры, МФУ, веб-камеры, наушники, колонки, телефоны), а также израсходованных канцелярских принадлежностей.

При утилизации персонального компьютера были выявлены особо вредные выбросы согласно ГОСТ Р 51768-2001 [27]. В случае выхода из строя компьютеров, они списываются и направляются на специальный склад, который при необходимости принимает меры по утилизации списанной техники и комплектующих. В настоящее время в Томской области утилизацией занимаются две компании: городской полигон и ООО НПП «Экотом».

Утилизацией опасных бытовых отходов занимаются компании: ООО «Торем», ООО «СибМеталлГрупп».

Эксплуатация объекта не подразумевает никаких вредных аспектов, которые могли бы неблагоприятно повлиять на окружающую среду.

6.4 Безопасность в чрезвычайных ситуациях при разработке проектного решения

Чрезвычайные ситуации бывают техногенного, природного, биологического, социального или экологического характера.

При работе в кабинете могут возникнуть следующие классификации чрезвычайных ситуаций:

- преднамеренные/непреднамеренные;
- техногенные: взрывы, пожары, обрушение помещений, аварии на системах жизнеобеспечения/природные – связанные с проявлением стихийных сил природы;
- экологические – это аномальные изменения состояния природной среды, такие как загрязнения биосферы, разрушение озонового слоя, кислотные дожди;
- антропогенные – являются следствием ошибочных действий людей;
- биологические – различные эпидемии, эпизоотии, эпифитотии;
- комбинированные.

В качестве анализа мер безопасности при чрезвычайной ситуации была выбрана наиболее распространенная ЧС, а именно пожар. Во время работы организации источником опасности могут стать неправильно подключенные электрические провода и кабели, неисправно работающая оргтехника, наличие легко воспламеняющихся материалов (дерево, бумага).

Выделим несколько основных факторов, по причине которых, возможно возникновение очага возгорания:

- Возникновение короткого замыкания.

- Разряд статического электричества.
- Неисправное электрооборудование.
- Несоблюдение мер пожарной безопасности.

Согласно ГОСТ Р 22.3.03-94 [28] и ГОСТ 12.1.004-91 [29], обеспечение безопасности людей при возникновении чрезвычайной ситуации является обязательной задачей для всех государственных систем и подразделений. Для профилактики возникновения ЧС должен проводиться следующий комплекс мер по предотвращению возгорания:

- Не допускается блокирование или загромождение пожарных выходов.
- Необходимо проводить регулярные проверки первичных средств для тушения пожаров и систем оповещения.
- Во всех служебных помещениях должны быть установлены «Планы эвакуации людей при пожаре и других ЧС».
- Должны проводиться инструктажи по пожарной безопасности и тренировки действий в случае возникновения пожаров.
- Необходимо правильное содержание и эксплуатация электрических приборов.
- Должны соблюдаться установленные в организации правила противопожарной безопасности.
- Помещения с ЭВМ должны быть оборудованы первичными средствами пожаротушения (разрешается использование углекислотных и порошковых огнетушителей).
- Недопустимо использовать для тушения пожара пенные огнетушители или воду, так как они проводят электрический ток.
- Помещения с ЭВМ должны проектироваться согласно I или II степени огнестойкости.

В случае если источник возгорания не может быть ликвидирован самостоятельно необходимо вызвать службы для пожаротушения и организовать эвакуацию персонала в минимально возможные сроки.

Рабочее место располагается в кибернетическом центре ТПУ 105 аудитории. На рисунке 33 представлен план эвакуации первого этажа кибернетического центра.



Рисунок 33 – План эвакуации

6.5 Заключение по разделу

В данном разделе был произведен анализ рабочего помещения, анализ вредных и опасных факторов и методы минимизации их воздействия на человеческое здоровье. Были рассмотрены нормативы микроклимата, освещения, вибрации, электробезопасности помещения. Исходя из полученных данных, можно утверждать, что оно соответствует требованиям законодательства РФ. Также были рассмотрены аспекты экологической, производственной безопасности, безопасности в чрезвычайных ситуациях (на примере пожара).

ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной квалификационной работы было проведено исследование и анализ имеющихся разработок в областях навигации и дополненной реальности.

Было разработано навигационное приложение под платформу Android, реализующее преобразование координат из картографического и AR-режимов для создания системы навигационных объектов, которая отображается в обоих режимах. У каждого объекта данной системы имеется информационная панель, в которой отображаются сохраненные текстовые данные.

Объекты навигационной системы синхронизированы с базой данных на удаленном сервере. При запуске приложения на сервер поступает запрос на получение имеющихся данных объектов, после чего данные записываются в файл на мобильном устройстве. При создании нового объекта на сервер поступает запрос на добавление данных о нем.

Мобильное приложение направлено на улучшение учебного процесса и ориентирование студентов в пределах корпусов ТПУ.

В рамках дальнейшего развития приложения возможна интеграция с другими сервисами ТПУ, расчет оптимального пути до аудиторий, а также сбор аналитики по студентам при использовании приложения.

СПИСОК ПУБЛИКАЦИЙ СТУДЕНТА

1. Использование технологий виртуальной реальности для повышения чистоты экспериментов в области философских проблем выбора / В. В. Видман, А. Д. Дмитриев, А. Р. Сергиенко, Д. И. Мингалеев // Молодежь и современные информационные технологии сборник трудов XVIII Международной научно-практической конференции студентов, аспирантов и молодых учёных — Томск, 2021 . — С. 304-305

2. Проектирование мобильного приложения для навигации с использованием технологий дополненной реальности на базе картографических данных / В. В. Лоскутов, А. Р. Сергиенко // Молодежь и современные информационные технологии сборник трудов XIX Международной научно-практической конференции студентов, аспирантов и молодых учёных — Томск, 2022 . — С. 209-210

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. «Яндекс.Навигатор», Google Maps, Maps.me: какие навигационные системы удобнее и функциональнее [Электронный ресурс] // holographica – 2021 URL: <https://matador.tech/articles/andeksnavigator-google-maps-mapsme-kakie-navigacionnye-sistemy-udobnee-i-funkcionalnee> (дата обращения: 25.04.2022)
2. Топ 18 SDK для работы с AR [Электронный ресурс] // vc.ru – 2020. – URL: <https://vc.ru/design/125017-top-18-sdk-dlya-raboty-s-ar> (дата обращения: 27.04.2022)
3. Van Krevelen, D. W. F.; Poelman, R. A survey of augmented reality technologies, applications and limitations. International Journal of Virtual Reality, 2010, vol. 9, no 2, p. 1
4. Lacueva Pérez, F.J., Brandl, P., Gracia Ban-drés, M.A. Technology Monitoring: Report on Information Needed for Workers in the Smart Factory // ResearchGate: [Электронный ресурс]. – URL: https://www.researchgate.net/publication/293569367_Technology_Monitoring_Report_on_Information_Needed_For_Workers_in_the_Smart_Factory (дата обращения: 20.11.2021)
5. Lightship ARDK [Электронный ресурс] // Niantic Lightship. – 2021. – URL: <https://lightship.dev> (дата обращения: 10.11.21)
6. Карты Яндекса для iOS получили режим дополненной реальности [Электронный ресурс] // holographica. – 2017. – URL: <https://holographica.space/news/yandex-maps-ar-ios-12496/> (дата обращения: 28.04.2022)
7. "Карты Google" включили ориентиры для "Улиц в AR-режиме" [Электронный ресурс] // Вести.Ru – Hi-Tech, 2020. – URL: <https://www.vesti.ru/hitech/article/2466813> (дата обращения: 28.04.2022)

8. В картах Google появилась AR-навигация внутри помещений [Электронный ресурс] // Вести.Ru – Hi-Tech, 2021. – URL: <https://www.vesti.ru/hitech/article/2544129> (дата обращения: 28.04.2022)

9. Apple Maps представляет новые способы изучения крупных городов в 3D в дополненной реальности (AR) [Электронный ресурс] // vc.ru – 2021. – URL: <https://vc.ru/tech/298715-apple-maps-predstavlyayet-novye-sposoby-izucheniya-krupnyh-gorodov-v-3d-v-dopolnennoy-realnosti-ar> (дата обращения: 27.04.2022)

10. iOS 15: How to Get AR Walking Directions in Maps [Электронный ресурс] // MacRumors – 2021. – URL: <https://www.macrumors.com/how-to/ios-ar-walking-directions-maps/> (дата обращения: 27.04.2022)

11. AR INDOOR NAVIGATION [Электронный ресурс] // Phygitalism – 2021. – URL: <https://phygitalism.com/ar-indoor-navigation/> (дата обращения: 28.04.2022)

12. Campus Go [Электронный ресурс] // robin bobina – 2018. – URL: <https://robinbobina.ru/campusgo> (дата обращения: 28.04.2022)

13. Unity – OpenStreetMap [Электронный ресурс] // Wiki. – 2021. – URL: <https://wiki.openstreetmap.org/wiki/Unity> (дата обращения: 18.11.21)

14. Maps for Unity [Электронный ресурс] // Mapbox. – 2010. – URL: <https://www.mapbox.com/unity/> (дата обращения: 19.11.21)

15. Krasner, G.E. and Pope, S.T., A Description of the Model-View-Controller User Interface Paradigm in the Smalltalk-80 system // Journal of Object Oriented Programming, 1988. 1(3): pp. 26-49.

16. Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 09.03.2021).

17. ГОСТ 12.2.032-78 Система стандартов безопасности труда. Рабочее место при выполнении работ сидя. Общие эргономические требования.

18. ГОСТ 12.2.061-81 Система стандартов безопасности труда. Оборудование производственное. Общие требования безопасности к рабочим местам.

19. ГОСТ Р 50923-96 Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения.

20. СП 2.2.3670-20 Санитарно-эпидемиологические требования к условиям труда.

21. ГОСТ 12.0.003-2015 Опасные и вредные производственные факторы. Классификация. Перечень опасных и вредных факторов.

22. СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания.

23. ГОСТ 12.1.012-2004 Система стандартов безопасности труда. Вибрационная безопасность. Общие требования.

24. СП 52.13330.2016 Естественное и искусственное освещение. Актуализированная редакция СНиП 23-05-95

25. ГОСТ Р 50948-2001 Средства отображения информации индивидуального пользователя. Общие эргономические требования безопасности.

26. ГОСТ 12.1.038-82 Система стандартов безопасности труда. Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов.

27. ГОСТ Р 51768-2001 Ресурсосбережение. Обращение с отходами. Методика определения ртути в ртутьсодержащих отходах.

28. ГОСТ Р 22.3.03-94 Безопасность в чрезвычайных ситуациях. Защита населения. Основные положения.

29. ГОСТ 12.1.004-91 Система стандартов безопасности труда. Пожарная безопасность. Общие требования.

ПРИЛОЖЕНИЕ А

(обязательное)

Исходный код реализации сервера с методами GET и POST

```
const express = require("express");
const bodyParser = require("body-parser");
const app = express();

app.use(bodyParser.json());

class LatitudeLongitude {
  x = null
  y = null
  constructor(x, y) {
    this.x = x;
    this.y = y;
  }
  getX() {
    return this.x;
  }
  getY() {
    return this.y;
  }
}

class Item extends LatitudeLongitude {
  next = null;
  constructor(type, id, text, x, y, Altitude) {
    super(x, y);
    this.type = type;
    this.id = id;
    this.text = text;
    this.Altitude = Altitude;
  }
  getX () {
    const x = super.getX()
    if (x === null) {
      return '0.0'
    }
    return x
  }
  getY () {
    const y = super.getY()
    if (y === null) {
      return '0.0'
    }
    return y
  }
}

class LinkedList {
  #head = null
  #tail = null
  append = (type, id, text, x, y, Altitude) => {
    const item = new Item(type, id, text, x, y, Altitude)
    if (!this.#head) {
      this.#head = item
    }
  }
}
```

```

        this.#tail = item
        return
    }
    this.#tail.next = item
    this.#tail = item
}
size = () => {
    let count = 1
    let item = this.#head
    if (!item) return 0
    while ((item = item.next)) {
        count++
    }
    return count
}
find = (id) => {
    let count = 1
    let item = this.#head
    if (!item) return null
    while ((item = item.next)) {
        if (item.id === id) {
            return item
        }
    }
    return null
}
show = () => {
    let item = this.#head
    let str = ``
    if (!item) return null
    str += `type - ${item.type}, id - ${item.id}, text - ${item.text}, LatLon -
${item.getX()}, ${item.getY()}, Altitude - ${item.Altitude} \n`
    while ((item = item.next)) {
        str += `type - ${item.type}, id - ${item.id}, text - ${item.text}, LatLon
- ${item.getX()}, ${item.getY()}, Altitude - ${item.Altitude} \n`
    }
    return str
}
}

const list = new LinkedList()

app.use(bodyParser.json());

app.get("/", (req, res) => {
    res.send(`${list.show()}`);
});

app.post("/add", (req, res) => {
    const { type, id, text, x, y, Altitude } = req.body;
    list.append(type, id, text, x, y, Altitude);
    res.send(`In list ${list.size()} items \n ${list.show()}`);
});

app.listen(7064, () => {
    console.log(`Server is running on port 7064.`);
});

```

ПРИЛОЖЕНИЕ Б

(обязательное)

Исходный код реализации изменения положения камеры в картографическом режиме

```
using UnityEngine;
using UnityEngine.EventSystems;
using Mapbox.Unity.Map;

public class CameraMovement : MonoBehaviour
{
    [SerializeField]
    AbstractMap _map;

    [SerializeField]
    float _panSpeed = 20f;

    [SerializeField]
    float _zoomSpeed = 50f;

    [SerializeField]
    Camera _referenceCamera;

    Quaternion _originalRotation;
    Vector3 _origin;
    Vector3 _delta;
    bool _shouldDrag;

    void HandleTouch()
    {
        float zoomFactor = 0.0f;
        switch (Input.touchCount)
        {
            case 1:
            {
                HandleMouseAndKeyBoard();
            }
            break;
            case 2:
            {
                Touch touchZero = Input.GetTouch(0);
                Touch touchOne = Input.GetTouch(1);

                Vector2 touchZeroPrevPos = touchZero.position -
touchZero.deltaPosition;
                Vector2 touchOnePrevPos = touchOne.position -
touchOne.deltaPosition;

                float prevTouchDeltaMag = (touchZeroPrevPos -
touchOnePrevPos).magnitude;
                float touchDeltaMag = (touchZero.position -
touchOne.position).magnitude;

                zoomFactor = 0.05f * (touchDeltaMag -
prevTouchDeltaMag);
            }
        }
    }
}
```

```

        ZoomMapUsingTouchOrMouse(zoomFactor);
        break;
    default:
        break;
    }
}

void ZoomMapUsingTouchOrMouse(float zoomFactor)
{
    var y = zoomFactor * _zoomSpeed;
    if ((transform.localPosition + transform.forward * y).y > 20 &&
(transform.localPosition + transform.forward * y).y < 2200)
    {
        transform.localPosition += (transform.forward * y);
    }
}

void HandleMouseAndKeyBoard()
{
    if (Input.GetMouseButton(0) &&
!EventSystem.current.IsPointerOverGameObject())
    {
        var mousePosition = Input.mousePosition;
        mousePosition.z = _referenceCamera.transform.localPosition.y;
        _delta = _referenceCamera.ScreenToWorldPoint(mousePosition) -
_referenceCamera.transform.localPosition;
        _delta.y = 0f;
        if (_shouldDrag == false)
        {
            _shouldDrag = true;
            _origin = _referenceCamera.ScreenToWorldPoint(mousePosition);
        }
        Ray ray = _referenceCamera.ScreenPointToRay(Input.mousePosition);
        RaycastHit hitObject;
        if (Physics.Raycast(ray, out hitObject))
        {
            if (hitObject.collider.CompareTag("SpawnedObject"))
            {
                _shouldDrag = false;
            }
        }
    }
    else
    {
        _shouldDrag = false;
    }

    if (_shouldDrag == true)
    {
        var offset = _origin - _delta;
        offset.y = transform.localPosition.y;
        transform.localPosition = offset;
    }
    else
    {
        if (EventSystem.current.IsPointerOverGameObject())
        {
            return;
        }
    }
}

```

```

        var x = Input.GetAxis("Horizontal");
        var z = Input.GetAxis("Vertical");
        var y = Input.GetAxis("Mouse ScrollWheel") * _zoomSpeed;
        if (!(Mathf.Approximately(x, 0) && Mathf.Approximately(y, 0) &&
Mathf.Approximately(z, 0)))
        {
            if ((transform.localPosition + transform.forward * y +
(_originalRotation * new Vector3(x * _panSpeed, 0, z * _panSpeed))).y < 2200 &&
(transform.localPosition + transform.forward * y + (_originalRotation * new Vector3(x *
_panSpeed, 0, z * _panSpeed))).y > 20)
            {
                transform.localPosition += transform.forward * y +
(_originalRotation * new Vector3(x * _panSpeed, 0, z * _panSpeed));
                _map.UpdateMap();
            }
        }
    }
}

void Awake()
{
    _originalRotation = Quaternion.Euler(0, transform.eulerAngles.y, 0);

    if (_referenceCamera == null)
    {
        _referenceCamera = GetComponent<Camera>();
        if (_referenceCamera == null)
        {
            throw new System.Exception("You must have a reference camera
assigned!");
        }
    }

    if (_map == null)
    {
        _map = FindObjectOfType<AbstractMap>();
        if (_map == null)
        {
            throw new System.Exception("You must have a reference map
assigned!");
        }
    }
}

void LateUpdate()
{
    if (Input.touchSupported && Input.touchCount > 0)
    {
        HandleTouch();
    }
    else
    {
        HandleMouseAndKeyBoard();
    }
}
}

```

ПРИЛОЖЕНИЕ В

(обязательное)

Исходный код реализации отображения текущего положения пользователя на карте

```
using Mapbox.Unity.Location;
using Mapbox.Unity.Map;
using UnityEngine;

public class ImmediatePositionWithLocationProvider : MonoBehaviour
{
    bool _isInitialized;

    ILocationProvider _locationProvider;
    ILocationProvider LocationProvider
    {
        get
        {
            if (_locationProvider == null)
            {
                _locationProvider =
                LocationProviderFactory.Instance.DefaultLocationProvider;
            }

            return _locationProvider;
        }
    }

    Vector3 _targetPosition;

    void Start()
    {
        LocationProviderFactory.Instance.mapManager.OnInitialized += () =>
        _isInitialized = true;
    }

    void LateUpdate()
    {
        if (_isInitialized)
        {
            var map = LocationProviderFactory.Instance.mapManager;
            transform.localPosition =
            map.GeoToWorldPosition(LocationProvider.CurrentLocation.LatitudeLongitude);
        }
    }
}
```

ПРИЛОЖЕНИЕ Г

(обязательное)

Исходный код реализации добавления на карту созданных объектов

```
using Mapbox.Unity.Location;
using Mapbox.Unity.Map;
using UnityEngine;

public class MakePointsFromJSON : MonoBehaviour
{
    private JSONController jSON;
    private int ObjInJsonCounter;
    public GameObject[] SpawnedObject;
    private GameObject ObjToSpawn;

    bool _isInitialized;
    bool _isSpawned;

    [SerializeField]
    private Camera _sceneCamera = null;

    ILocationProvider _locationProvider;
    ILocationProvider LocationProvider
    {
        get
        {
            if (_locationProvider == null)
            {
                _locationProvider =
                LocationProviderFactory.Instance.DefaultLocationProvider;
            }

            return _locationProvider;
        }
    }

    void Start()
    {
        jSON = gameObject.GetComponent<JSONController>();
        jSON.LoadField();
        ObjInJsonCounter = jSON.items.Items.Count;
        Debug.Log("In JSON " + ObjInJsonCounter + " items");
        _isSpawned = false;
        LocationProviderFactory.Instance.mapManager.OnInitialized += () =>
        _isInitialized = true;
    }

    void Update()
    {
        if (Input.touchCount > 0)
        {
            int touchCount = Input.touchCount;
            if (touchCount <= 0)
            {
                return;
            }
        }
    }
}
```

```

        Touch touch = Input.GetTouch(0);
        Vector2 touchPosition = touch.position;

        TrySelectObject(touchPosition);
    }
    if (Input.GetMouseButtonDown(0))
    {
        TrySelectObject(Input.mousePosition);
    }
}

void LateUpdate()
{
    if (!_isInitialized)
    {
        if (!_isSpawned)
        {
            var map = LocationProviderFactory.Instance.mapManager;
            for (int i = 0; i < ObjInJsonCounter; i++)
            {
                for (int j = 0; j < SpawnedObject.Length; j++)
                {
                    if (SpawnedObject[j].name == jSON.items.Items[i].type)
ObjToSpawn = SpawnedObject[j];
                }
                GameObject Point = Instantiate(ObjToSpawn,
map.GeoToWorldPosition(jSON.items.Items[i].LatitudeLongitude),
ObjToSpawn.transform.rotation);
                Point.transform.localScale = new Vector3(8, 8, 8);

                SpawnedObject stats = Point.GetComponent<SpawnedObject>();
                stats.Name = jSON.items.Items[i].type + ", id - " +
jSON.items.Items[i].id;
                stats.Id = jSON.items.Items[i].id;
                stats.Description = jSON.items.Items[i].text;
                stats.Object = Point;
                stats.isrotated = false;

                stats.field.text = string.Join(" \r\n ", jSON.items.Items[i].text +
"\r\n" + "Loc - " + Point.transform.position + " \r\n" + "Altitude - " +
jSON.items.Items[i].Altitude);

            }
            _isSpawned = true;
        }
    }
}

public void TrySelectObject(Vector2 pos)
{
    Ray ray = _sceneCamera.ScreenPointToRay(pos);
    RaycastHit hitObject;
    if (Physics.Raycast(ray, out hitObject))
    {
        if (hitObject.collider.CompareTag("SpawnedObject"))
        {
            GameObject viewObject = hitObject.collider.gameObject;
            SpawnedObject stats = viewObject.GetComponentInParent<SpawnedObject>();
            if (stats != null)

```



```
        {
            stats.isrotated = false;
            stats.Panel.SetActive(true);
            stats.ShowPanel = true;
            stats.RotTrigger = false;
            stats.TouchEnter = true;
            stats.CameraRotationY =
_sceneCamera.transform.rotation.eulerAngles.y;
        }
    }
}
```

ПРИЛОЖЕНИЕ Д

(обязательное)

Исходный код реализации контроллера режима дополненной реальности

```
using Niantic.ARDK.Extensions;
using UnityEngine;
using UnityEngine.UI;
using Mapbox.Unity.Location;
using Mapbox.Utils;
using Mapbox.Unity.Utilities;
using System;

public class Manager : MonoBehaviour
{
    [SerializeField] private Slider _depthSlider = null;
    [SerializeField] private Text _depthText = null;
    [SerializeField] private float _maxDepth = 25.0f;
    [SerializeField] private Camera _sceneCamera = null;
    [SerializeField] private Text _pinButtonText = null;
    [SerializeField] private ARDepthManager _arDepthManager = null;
    [SerializeField] private Toggle _toggleUI = null;

    private const float DEGREES_PER_SECOND = 30.0f;
    private const float DEFAULT_SLIDER_VALUE = 0.15f;
    private bool _pinnedToWorldSpace;
    private bool _isShowingDepth;

    [SerializeField] private GameObject BtnAdd;
    [SerializeField] private GameObject BtnClose;
    [SerializeField] private Text _addText;
    [SerializeField] private GameObject BtnPin;
    [SerializeField] private GameObject SpawnObjectScroll;
    private int SpawnObjectIndex = -1;
    private enum ObjectGeneratorState { Default, ActiveSpawn, SpawnReady, Selection }
    private ObjectGeneratorState currentState;
    private GameObject selectedObject;
    private GameObject viewObject;
    public GameObject[] SpawnedObject;
    private int[] SpawnObjectCount;
    private string[] SpawnObjectName;

    public GameObject Exclamation;
    public GameObject Question;
    public GameObject Note;
    public GameObject Direction;
    public GameObject Panel;

    private JSONController jSON;
    private GameObject jsonObj;
    private int ObjInJsonCounter;
    public GameObject Map;

    bool _isInitialized;
    bool _isSpawned;

    public Text PlayerIifo;
    private string ObjectsInfo;
```

```

ILocationProvider _locationProvider;
ILocationProvider LocationProvider
{
    get
    {
        if (_locationProvider == null)
        {
            _locationProvider =
LocationProviderFactory.Instance.DefaultLocationProvider;
        }

        return _locationProvider;
    }
}

void Start()
{
    HideObjects();

    currentState = ObjectGeneratorState.Default;
    SpawnObjectCount = new int[SpawnedObject.Length];
    SpawnObjectName = new string[SpawnedObject.Length];
    for (int i = 0; i < SpawnedObject.Length; i++)
    {
        SpawnObjectCount[i] = 0;
        SpawnObjectName[i] = SpawnedObject[i].name;
    }
    SpawnObjectScroll.SetActive(false);
    BtnPin.SetActive(false);

    jSON = gameObject.GetComponent<JSONController>();
    jSON.LoadField();
    ObjInJsonCounter = jSON.items.Items.Count;
    Debug.Log("In JSON " + ObjInJsonCounter + " items");

    LocationProviderFactory.Instance.mapManager.OnInitialized += () =>
_isInitialized = true;
}
void LateUpdate()
{
    if (_isInitialized)
    {
        if (!_isSpawned)
        {
            for (int i = 0; i < ObjInJsonCounter; i++)
            {
                for (int j = 0; j < SpawnedObject.Length; j++)
                {
                    if (SpawnedObject[j].name == jSON.items.Items[i].type)
                    {
                        jsonObj = SpawnedObject[j];
                        SpawnObjectIndex = j;
                    }
                }
                SpawnObjectCount[SpawnObjectIndex]++;
            }
        }
    }
}

```

```

        Vector3 pos =
LocationProviderFactory.Instance.mapManager.GeoToWorldPosition(jJSON.items.Items[i].LatitudeLongitude);

        Vector2d userGeo =
LocationProvider.CurrentLocation.LatitudeLongitude;
        Vector3 objPos =
RecalcPosForRecreation(/*_sceneCamera.transform.position*/new Vector3(0,0,0),
(float)userGeo.x, (float)userGeo.y, Input.location.lastData.altitude,
jJSON.items.Items[i].LatitudeLongitude, jJSON.items.Items[i].Altitude,
/*Input.compass.magneticHeading*/ LocationProvider.CurrentLocation.UserHeading);

        GameObject Point = Instantiate(jjsonObj, objPos,
jjsonObj.transform.rotation);
        SpawnedObject stats = Point.GetComponent<SpawnedObject>();
        stats.Name = SpawnObjectName[SpawnObjectIndex] + " " +
SpawnObjectCount[SpawnObjectIndex].ToString();
        stats.Description = "Описание объекта:\nИмя:" + stats.Name +
"\nСоздатель: Другой игрок";
        stats.Object = Point;
        stats.isrotated = true;
        stats.Description = jJSON.items.Items[i].text;
        stats.Panel.SetActive(false);
        stats.Id = jJSON.items.Items[i].id;

        PanelController panel =
stats.Panel.GetComponent<PanelController>();
        panel.Title.GetComponent<Text>().text = jJSON.items.Items[i].type +
", id - " + jJSON.items.Items[i].id;
        stats.field.text = string.Join(" \r\n ", jJSON.items.Items[i].text+
\r\n" + "Loc - " + Point.transform.position + " \r\n" + "Altitude - " +
jJSON.items.Items[i].Altitude); //Point.transform.Find("InputField
(TMP)").gameObject.GetComponent<TMP_InputField>().text;

        ObjectsInfo = ObjectsInfo + " \r\n" + stats.Name + " pos - " +
Point.transform.position;
    }
    SpawnObjectIndex = -1;
    _isSpawned = true;
    Map.SetActive(false);
}
}
}

public static Vector3 RotCoordsOnDegr(Vector3 coords, float degr)
{
    float[,] RotMatr = new float[,] { { Mathf.Cos((float)(degr * Math.PI / 180)), -
Mathf.Sin((float)(degr * Math.PI / 180)), 0f }, { Mathf.Sin((float)(degr * Math.PI /
180)), Mathf.Cos((float)(degr * Math.PI / 180)), 0f }, { 0f, 0f, 1f } };
    Vector3 recPos = new Vector3(RotMatr[0, 0] * coords.x + RotMatr[0, 1] *
coords.y + RotMatr[0, 2] * coords.z,
    RotMatr[1, 0] * coords.x + RotMatr[1, 1] * coords.y + RotMatr[1, 2] *
coords.z,
    RotMatr[2, 0] * coords.x + RotMatr[2, 1] * coords.y + RotMatr[2, 2] *
coords.z);
    return recPos;
}

public static Vector2 GeoTo2D(float lat, float lon)

```

```

    {
        double EarthRadius = 6378137;
        double PolarRadius = 6356752.3142;
        double e = Math.Sqrt((1 - (PolarRadius * PolarRadius) / (EarthRadius *
EarthRadius)));
        float x = (float)(EarthRadius * lon * Math.PI / 180);
        float f = (float)(Math.Pow((1 - e * Math.Sin(lat * Math.PI / 180)) / (1 + e *
Math.Sin(lat * Math.PI / 180)), e / 2));
        float ff = (float)(Math.Tan(Math.PI / 4 + lat * Math.PI / 360));
        float fff = (ff * f);
        float y = (float)(EarthRadius * Math.Log(fff));
        Vector2 projCoords = new Vector2(x, y);
        return projCoords;
    }

    public static Vector3 RecalcPosForRecreation(Vector3 userPos, float userLat, float
userLon, float userAlt, Vector2d objGeo, float objAlt, float degr)
    {
        Vector2 userProj = GeoTo2D(userLat, userLon);
        Vector2 objProj = GeoTo2D((float)objGeo.x, (float)objGeo.y);
        Vector3 userPosX = new Vector3(userPos.x, userPos.z, userPos.y);
        Vector3 userWorldPosX = RotCoordsOnDegr(userPosX, degr);

        Vector3 objPosX = new Vector3(userWorldPosX.x + (objProj.x - userProj.x),
userWorldPosX.y + (objProj.y - userProj.y), objAlt - userAlt);
        Vector3 objPosTurned = RotCoordsOnDegr(objPosX, -degr);
        Vector3 returnedObjPos = new Vector3(objPosTurned.x, objPosTurned.z,
objPosTurned.y);

        return returnedObjPos;
    }

    public void TogglePinToWorldSpace()
    {
        _pinnedToWorldSpace = !_pinnedToWorldSpace;
        if (_pinnedToWorldSpace)
        {
            _pinButtonText.text = "Move with Camera";
            _depthText.text = "No selected object";
            _addText.text = "Add new object";
            SpawnObject();
            CloseSelection();
            HideObjects();
            jSON.SaveAllObjects(jSON.items.Items.ToArray());
        }
        else
        {
            _pinButtonText.text = "    Pin To World";
            _depthSlider.value = DEFAULT_SLIDER_VALUE;
        }
    }

    public void ToggleOcclusion()
    {
        if (!_toggleUI.isOn)
            _arDepthManager.DisableFeatures();
        else
            _arDepthManager.EnableFeatures();
    }

```

```

private void Awake()
{
    _depthSlider.onValueChanged.AddListener(AdjustDepth);
}

/// Rotate the object every frame
private void Update()
{
    if (selectedObject != null)
    {
        AdjustDepth(_depthSlider.value);
        selectedObject.transform.Rotate(Vector3.up * Time.deltaTime *
DEGREES_PER_SECOND);
    }

    TextUpdate();

    if (Input.touchCount > 0)
    {
        int touchCount = Input.touchCount;
        if (touchCount <= 0)
        {
            return;
        }

        Touch touch = Input.GetTouch(0);
        Vector2 touchPosition = touch.position;

        //process object movement and rotation
        if (currentState == ObjectGeneratorState.Selection)
        {
            //В дальнейшем можно будет добавить дополнительные функции
            //к примеру, смещение позиции сознного объекта
        }
        // process object selection
        if (currentState == ObjectGeneratorState.Default)
        {
            TrySelectObject(touchPosition);
        }

        // process object creation
        if (currentState == ObjectGeneratorState.ActiveSpawn || currentState ==
ObjectGeneratorState.SpawnReady)
        {
            selectedObject.gameObject.SetActive(true);
        }
    }
    if (Input.GetMouseButtonDown(0))
    {
        TrySelectObject(Input.mousePosition);
    }
}

public void TrySelectObject(Vector2 pos)
{
    Ray ray = _sceneCamera.ScreenPointToRay(pos);
    RaycastHit hitObject;
    Debug.Log("Begin TrySelectObject");
}

```

```

if (Physics.Raycast(ray, out hitObject))
{
    if (hitObject.collider.CompareTag("SpawnedObject"))
    {
        viewObject = hitObject.collider.gameObject;
        SpawnedObject stats = viewObject.GetComponentInParent<SpawnedObject>();
        if (stats != null)
        {
            currentState = ObjectGeneratorState.Selection;
            //stats.isrotated = false;
            //stats.InfoPanel.SetActive(true);
            stats.ShowPanel = true;
            stats.RotTrigger = false;
            stats.TouchEnter = true;
            stats.CameraRotationY =
_sceneCamera.transform.rotation.eulerAngles.y;
        }
    }
}
}
public void SpawnObject()
{
    GameObject spawn = Instantiate(SpawnedObject[SpawnObjectIndex],
selectedObject.transform.position, SpawnedObject[SpawnObjectIndex].transform.rotation);
    currentState = ObjectGeneratorState.Default;

    SpawnObjectCount[SpawnObjectIndex]++;
    SpawnedObject stats = spawn.GetComponent<SpawnedObject>();
    stats.Name = SpawnObjectName[SpawnObjectIndex] + " " +
SpawnObjectCount[SpawnObjectIndex].ToString();
    stats.Description = "Описание объекта:\nИмя:" + stats.Name + "\nСоздатель:
Игрок";
    stats.Object = spawn;
    stats.RotTrigger = true;
    stats.Id = ObjInJsonCounter;
    BtnClose.SetActive(false);
    BtnAdd.SetActive(true);

    jSON.items.Items.Add(new JSONController.Item());
    jSON.items.Items[ObjInJsonCounter].id = ObjInJsonCounter;
    jSON.items.Items[ObjInJsonCounter].type = SpawnedObject[SpawnObjectIndex].name;
    jSON.items.Items[ObjInJsonCounter].text = "";
    jSON.items.Items[ObjInJsonCounter].Altitude = Input.location.lastData.altitude;

    Vector3 userWorldPosX = RotCoordsOnDegr(_sceneCamera.transform.position,
LocationProvider.CurrentLocation.UserHeading);
    Vector3 objWorldPosX = RotCoordsOnDegr(spawn.transform.position,
LocationProvider.CurrentLocation.UserHeading);
    var offset = userWorldPosX - objWorldPosX;
    if (null != LocationProviderFactory.Instance.mapManager)
    {
        float factor =
Conversions.GetTileScaleInMeters(Input.location.lastData.latitude,
LocationProviderFactory.Instance.mapManager.AbsoluteZoom) /
LocationProviderFactory.Instance.mapManager.UnityTileSize;
        var latlongDelta = Conversions.MetersToLatLon(new Vector2d(offset.x *
factor, offset.z * factor));
        var newLatLong = LocationProvider.CurrentLocation.LatitudeLongitude +
latlongDelta;

```

```

        jSON.items.Items[ObjInJsonCounter].LatitudeLongitude = newLatLong;
    }
    ObjectsInfo = ObjectsInfo + " \r\n" + stats.Name + " pos - " +
spawn.transform.position;
    BtnPin.SetActive(false);
    ObjInJsonCounter++;
}

/// Project the object some depth forward from the camera
private void AdjustDepth(float sliderPos)
{
    if (_pinnedToWorldSpace)
        return;

    var convertedDepth = sliderPos * sliderPos * _maxDepth;
    var pos = _sceneCamera.ViewportToWorldPoint(new Vector3(0.5f, 0.5f,
convertedDepth));

    _depthText.text = "Depth: " + convertedDepth + " meters";
    selectedObject.transform.position = pos;
}
public void ToggleShowDepth()
{
    _isShowingDepth = !_isShowingDepth;
    _arDepthManager.ToggleDebugVisualization(_isShowingDepth);
}

public void ShowSelectionScroll()
{
    SpawnObjectScroll.SetActive(true);
    BtnClose.SetActive(true);
    _addText.text = "Cancel";
    BtnAdd.SetActive(false);
    CloseSelection();
    HideObjects();
    _pinButtonText.text = "    Pin To World";
    _depthSlider.value = DEFAULT_SLIDER_VALUE;
}

public void HideSelectionScroll()
{
    SpawnObjectScroll.SetActive(false);
    BtnClose.SetActive(false);
    BtnAdd.SetActive(true);
    _addText.text = "Add new object";
    _pinButtonText.text = "Move with Camera";
    _depthText.text = "No selected object";
    _addText.text = "Add new object";
    BtnPin.SetActive(false);
    CloseSelection();
}

public void ChooseTypeExclamation()
{
    SpawnObjectIndex = 0;
    Exclamation.SetActive(true);
    selectedObject = Exclamation;
    SpawnObjectScroll.SetActive(false);
    BtnPin.SetActive(true);
}

```



```

        currentState = ObjectGeneratorState.ActiveSpawn;
        if (_pinnedToWorldSpace) TogglePinToWorldSpace();
    }
    public void ChooseTypeQuestion()
    {
        SpawnObjectIndex = 1;
        Question.SetActive(true);
        selectedObject = Question;
        SpawnObjectScroll.SetActive(false);
        BtnPin.SetActive(true);
        currentState = ObjectGeneratorState.ActiveSpawn;
        if (_pinnedToWorldSpace) TogglePinToWorldSpace();
    }
    public void ChooseTypeNote()
    {
        SpawnObjectIndex = 2;
        Note.SetActive(true);
        selectedObject = Note;
        SpawnObjectScroll.SetActive(false);
        BtnPin.SetActive(true);
        currentState = ObjectGeneratorState.ActiveSpawn;
        if (_pinnedToWorldSpace) TogglePinToWorldSpace();
    }
    public void ChooseTypeDirection()
    {
        Direction.SetActive(true);
        SpawnObjectIndex = 3;
        selectedObject = Direction;
        SpawnObjectScroll.SetActive(false);
        BtnPin.SetActive(true);
        currentState = ObjectGeneratorState.ActiveSpawn;
        if (_pinnedToWorldSpace) TogglePinToWorldSpace();
    }
    public void CloseSelection()
    {
        currentState = ObjectGeneratorState.Default;
        selectedObject.SetActive(false);
        SpawnObjectIndex = -1;
    }
    public void HideObjects()
    {
        Exclamation.SetActive(false);
        Question.SetActive(false);
        Note.SetActive(false);
        Direction.SetActive(false);
    }

    private void TextUpdate()
    {
        PlayerIifo.text = "Your pos - " + _sceneCamera.transform.position + " \r\n" +
        "Your rot - " + _sceneCamera.transform.rotation.eulerAngles + " \r\n" + ObjectsInfo;
    }
}

```

ПРИЛОЖЕНИЕ Е

(обязательное)

Исходный код реализации контроллера создаваемых объектов

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using TMPro;
using System;

public class SpawnedObject : MonoBehaviour
{
    private string displayName;
    private string description;
    private int id;
    private bool shouldRotate = false;
    private bool rotTrigger = false;
    private bool enterByTouch = false;
    private float mainCameraYRotation = 0;
    private float rotationKoeff = 1f;
    private bool showPanel = false;
    private GameObject body;
    public GameObject Panel;
    private JSONController jSON;
    public TMP_InputField field;
    private bool jsonIsFind = false;

    public GameObject Object
    {
        get { return body; }
        set { body = value; }
    }
    public string Name
    {
        get { return displayName; }
        set { displayName = value; }
    }
    public string Description
    {
        get { return description; }
        set { description = value; }
    }
    public int Id
    {
        get { return id; }
        set { id = value; }
    }
    public bool isrotated
    {
        get { return shouldRotate; }
        set { shouldRotate = value; }
    }
    public bool TouchEnter
    {
        get { return enterByTouch; }
    }
}
```

```

        set { enterByTouch = value; }
    }
    public bool RotTrigger
    {
        get { return rotTrigger; }
        set { rotTrigger = value; }
    }
    public float CameraRotationY
    {
        get { return mainCameraYRotation; }
        set { mainCameraYRotation = value; }
    }
    public bool ShowPanel
    {
        get { return showPanel; }
        set { showPanel = value; }
    }
}

private void Start()
{
    Panel.SetActive(false);
    json = FindObjectOfType<JSONController>();
    if (json != null)
    {
        jsonIsFind = true;
        Debug.Log("JSONController has been find, id - " + id);
    }

    PanelController panel = Panel.GetComponent<PanelController>();
    panel.Title.GetComponent<Text>().text = Name;
    panel.SpObjRef = gameObject.GetComponent<SpawnedObject>();
}

void Update()
{
    var angles = transform.rotation.eulerAngles;
    if (Math.Abs(mainCameraYRotation - angles.y) > 2f)
    {
        rotationKoeff = 3f;
        shouldRotate = true;
    }
    else
    {
        shouldRotate = false;
        if (!rotTrigger && enterByTouch)
        {
            Panel.SetActive(true);
            enterByTouch = true;
        }
    }
    if (rotTrigger)
    {
        shouldRotate = true;
        rotationKoeff = 1f;
    }
    if (shouldRotate)
    {
        angles.y += 0.5f * rotationKoeff;
        transform.rotation = Quaternion.Euler(angles);
    }
}

```

```
}
public void TextChanged()
{
    if (jsonIsFind)
    {
        for (int i = 0; i < jSON.items.Items.Count; i++)
        {
            if (jSON.items.Items[i].id == id)
            {
                jSON.items.Items[i].text = field.text;
            }
        }
        jSON.SaveAllObjects(jSON.items.Items.ToArray());
    }
    else
    {
        Debug.Log("jSON is not found yet");
    }
}
}
```

ПРИЛОЖЕНИЕ Ж

(обязательное)

Исходный код реализации контроллера сохраняемых данных в файле

```
using System.Collections;
using System.Collections.Generic;
using Mapbox.Unity.Location;
using UnityEngine;
using System.IO;
using System;
using System.Text;

public class JSONController : MonoBehaviour
{
    public AllObjects items;

    void Start()
    {
        Debug.Log(Application.persistentDataPath);
        LoadField();
        SaveAllObjects(items.Items.ToArray());
        Debug.Log(items.Items.Count + " items in json");
    }

    [ContextMenu("Load")]
    public void LoadField()
    {
        string path = Application.persistentDataPath + "/JSON.json";

        // if the file path or name does not exist
        if (!Directory.Exists(Path.GetDirectoryName(path)))
        {
            Debug.LogWarning("File or path does not exist! " + path);
            Debug.Log("File was craeted");
            return;
        }

        // load in the save data as byte array
        byte[] jsonDataAsBytes = null;

        try
        {
            jsonDataAsBytes = File.ReadAllBytes(path);
            Debug.Log("<color=green>Loaded all data from: </color>" + path);
        }
        catch (Exception e)
        {
            Debug.LogWarning("Failed to load data from: " + path);
            Debug.LogWarning("Error: " + e.Message);
            return;
        }

        if (jsonDataAsBytes == null)
            return;

        // convert the byte array to json
        string jsonData;
```

```

        // convert the byte array to json
        jsonData = Encoding.ASCII.GetString(jsonDataAsBytes);

        JsonUtility.FromJsonOverwrite(jsonData, items);
    }

    [ContextMenu("Save")]
    public void SaveAllObjects(in Item[] p)
    {
        AllObjects data = new AllObjects();

        foreach(Item obj in p)
        {
            data.Items.Add(new Item());
        }

        File.WriteAllText(Application.persistentDataPath + "/JSON.json",
            JsonUtility.ToJson(items));
    }

    [System.Serializable]
    public class Item
    {
        public string type;
        public int id;
        public string text;
        public Mapbox.Utils.Vector2d LatitudeLongitude;
        public float Altitude;
    }

    [System.Serializable]
    public class AllObjects
    {
        public List<Item> Items = new List<Item>();
    }
}

```

ПРИЛОЖЕНИЕ И

(справочное)

Available navigation applications and augmented reality development tools

Студент:

Группа	ФИО	Подпись	Дата
8ИМ01	Сергиенко Александр Романович		

Консультант школы отделения (НОЦ) ИШИТР, ОИТ :

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Демин Антон Юрьевич	к.т.н.		

Консультант-лингвист отделения иностранных языков ШБИП:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Пичугова Инна Леонидовна			

1 AVAILABLE NAVIGATION APPLICATIONS AND AUGMENTED REALITY DEVELOPMENT TOOLS

1.1 Analysis of available navigation applications

Navigator is kind of software that allows you to display cartographic data, which you can use in real time to get from point A to point B.

All navigators have one goal – to help the user navigate on their location. However, each program copes with this task in different ways – the functionality and capabilities of modern applications can be very different. Further, the most popular navigation applications, their advantages and disadvantages will be presented, as well as their areas of application will be noted.

Yandex.Navigator

Yandex.Navigator is a navigation application by Yandex, developed on the technological basis of Yandex.Maps. This application supports schematic and satellite mapping of the area. It works on the principle of context: it shows the most relevant information for the user [1].

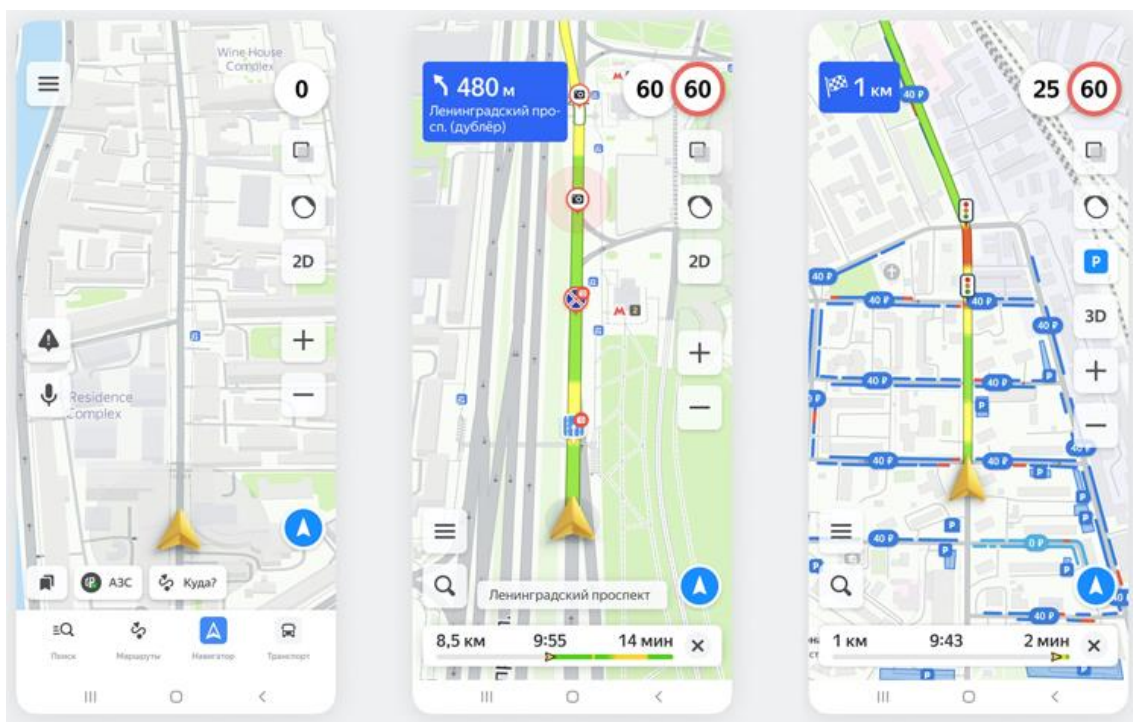


Figure 1 – Interface of the Yandex.Navigator application

- The application is available for Android and IOS devices;
- The application is free for users who do not belong to legal entities;

- Coverage area: Russia, Ukraine, Kazakhstan, Belarus, Azerbaijan, Armenia, Georgia, Abkhazia, Moldova, Uzbekistan, Kyrgyzstan, Tajikistan, Turkey.

Advantages:

- Intuitive interface that is easy to use;
- Building routes taking into account traffic jams and accidents;
- Ability to load routes from the user's history;
- Voice search function for setting a route;
- Up-to-date, timely updated map information;
- Ability to use the application without a network connection;
- The navigator helps the driver in choosing a lane for maneuver.

Disadvantages:

- Routing does not work without the Internet;
- There is no information about the passages in the yards;
- No speeding alerts.

This application is optimally suited for orientation in Russia and the CIS countries, as well as for use on highways and in large cities.

Google Maps

Google Maps is a publicly available mapping service from Google. These maps with the car navigation option work today on mobile devices with Android and IOS operating systems. Google Maps can also be used in car multimedia systems equipped with Apple CarPlay and Android Auto. In addition to maps, the application also contains satellite images of the Earth. The application allows the user to create difficult routes.

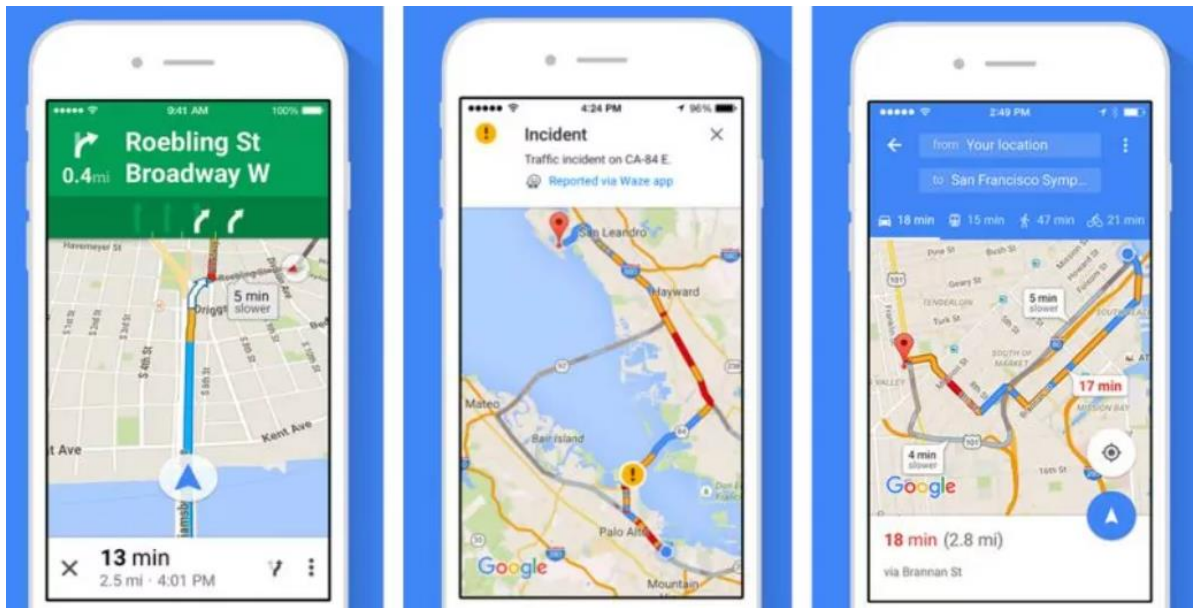


Figure 2 – The interface of the application «Google Maps»

- Free application, installed by default on devices with Android OS, it is also possible to install on Apple devices;
- The application works in Russia, Great Britain, part of Ireland and other parts of Europe, the USA, Canada, Japan, China, Hong Kong.

Advantages:

- The application rebuilds the route in a timely manner by reading data about traffic jams, accidents, roadworks;
- The application provides the ability to display the location of traffic cameras that record the speed of movement;
- There is a built-in voice assistant that accompanies movement with prompts;
- There is information about objects on the map;
- There is the possibility of saving maps to the device memory and using the application without access to the Internet.

Disadvantages:

- Insufficient accuracy and detail of maps and routes in Russia and Europe.

The application is optimal for use in car trips abroad.

Maps.me

In 2010, Belarusian IT developer and entrepreneur Yury Melnichek created a functional navigation system that can work even without Internet access. Today Maps.me is called the main competitor of Yandex. This is due to what opportunities Maps.me provides to its user.

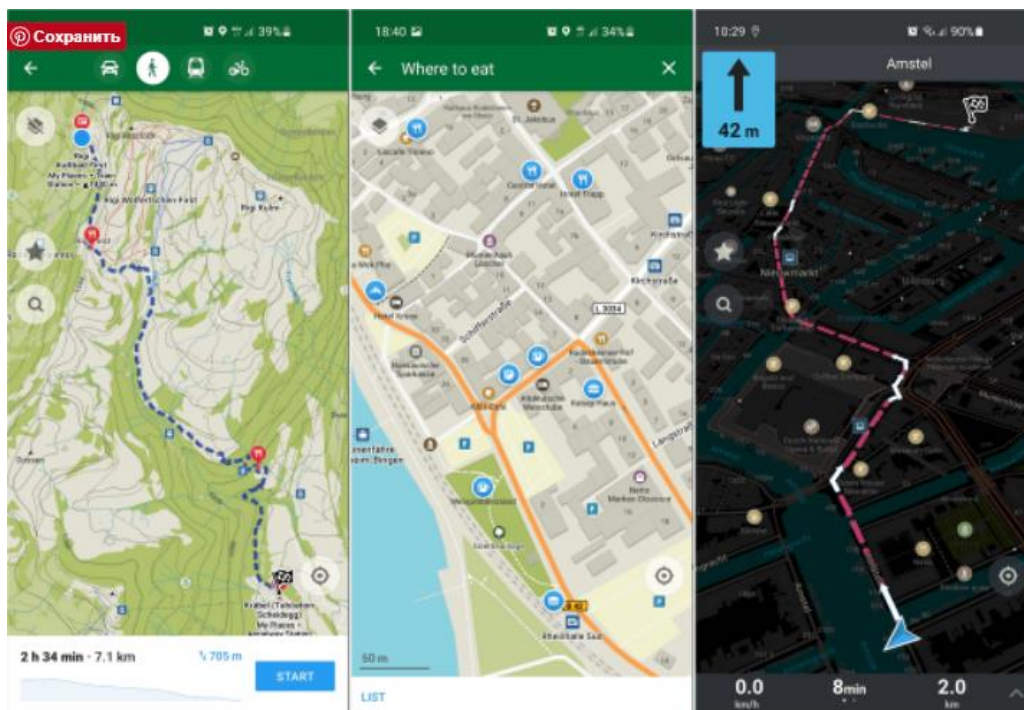


Figure 3 – Maps.me application interface

The application works on the basis of the free geographic map OpenStreetMap. Before starting work, the required maps are loaded into the device's memory and can be used without an Internet connection.

- The application can be used on mobile devices with iOS and Android platforms;
- Free application, there is a Pro version with additional features;
- Coverage area: all over the world.

Advantages:

- The application has the ability to build routes for cars and pedestrians without an Internet connection;
- Ability to download only the necessary maps of cities and regions;
- High accuracy and map detail;
- Search by objects is possible;
- It is possible to use voice navigation;

- Continuous updating of navigation information;
- The ability to transfer data about your location to other users (in the Pro version).

Disadvantages:

- Lack of real-time information;
- Requires space in device memory to load maps;
- The maximum functionality of the application only in the paid version.

Maps.me is best suited for foreign trips and any other situation where internet connectivity is not possible or limited.

OSMAND

OsmAnd is a navigation application based on OpenStreetMap, Yandex.Maps and Yandex.Traffic, Mapnik, Osmarender, Microsoft Maps, CloudMade and other services. The application implies preliminary downloading of maps of the required regions to the device's memory. In this regard, OsmAnd is able to create routes where there is no Internet connection. The application is able to restore past routes, build new ones, and also allows you to navigate in a neighboring state.

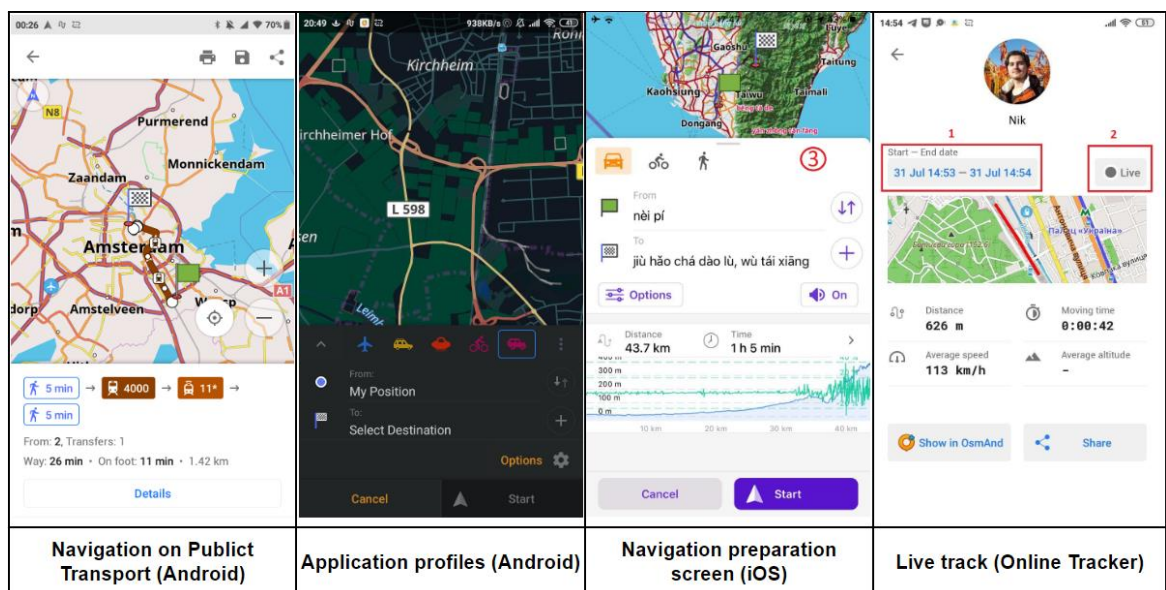


Figure 4 – The interface of the OSMAND application

This application supports online navigation through the use of YOURS and OSRM services. If you have access to the Internet, the route will be rebuilt taking

into account traffic jams and many other options that can be set in the settings. The application provides the ability to cache data to save Internet traffic.

- Available for download on Android and IOS devices;
- There are free and paid versions of the application;
- Coverage area: Russia, CIS countries, non-CIS countries.

Advantages:

- High detail maps;
- Ability to save a custom route;
- There are voice prompts;
- Automatic scaling of the map in accordance with the speed of the user's movement;
- Ability to use the application in different modes (car / pedestrian / ski, etc.);
- Ability to use multiple online maps at the same time;
- The application supports the use of additional plugins.

Disadvantages:

- There is a limit on the number of downloads and map updates in the free version;
- Offline maps are updated only once a month;
- Requires free space on device memory.

OsmAnd is best suited for travelers using various means of transportation in areas with poor or no internet connection.

2GIS

In 1999, a native project was created that is directly related to navigation and cartography: 2GIS, a navigation system that contains reference information on cities and organizations in its buildings. This application can be called an electronic directory combined with a city map, which is free for users and works at the expense of advertisers of companies that provide information about themselves to be marked in the application.

Cartography developed on the basis of satellite images. The developer company guarantees 95% map accuracy. The application can work in online and offline mode (by preloading maps on the device).

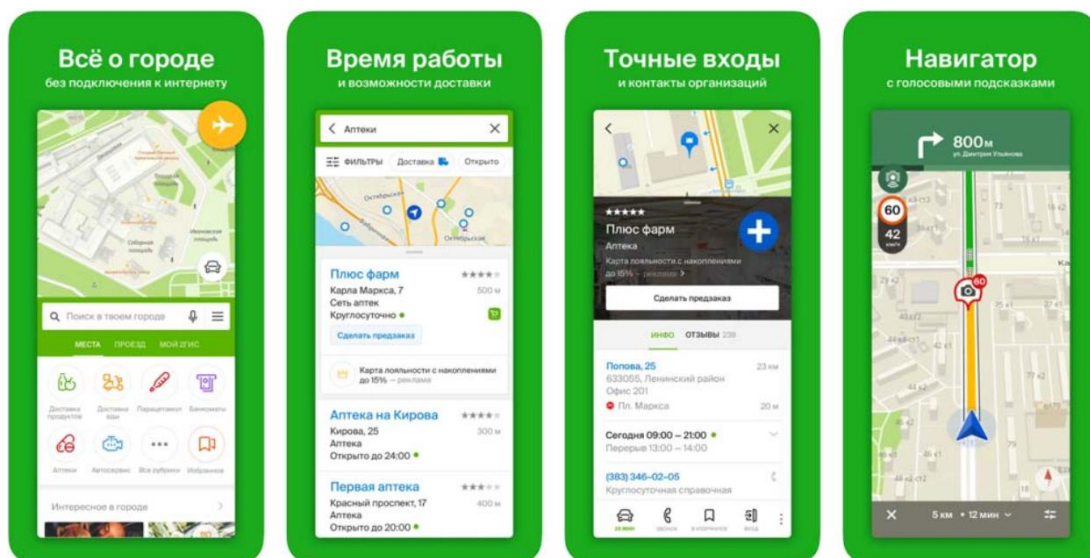


Figure 5 – Interface of the 2GIS application

- The application is free for Android and IOS devices;
- Coverage area: more than 390 cities in Russia, Kazakhstan, Ukraine, Uzbekistan, Azerbaijan, Italy, Czech Republic, Chile, UAE, Kyrgyzstan, Cyprus.

Advantages:

- There is no paid version of the application;
- The application provides prompts for the user during his progress along the route, as well as guiding voice instructions;
- It is possible to automatically correct the route (in case of deviation from the original);
- There is information about public transport routes;
- Simple and clear interface;
- High application speed.

Disadvantages:

- Insufficient detailing of maps outside the tracks and large settlements;
- Installed maps take up free space in the device's memory.

The application is suitable to use primarily in large cities – in particular, for business and work purposes.

Waze

Waze is a social navigation app for mobile devices. This application allows you to track the traffic situation in real time and get directions. Information on the map is updated by the users themselves.



Figure 6 – Waze application interface

- Free application;
- For mobile devices on Android and IOS.

Advantages:

- Sufficiently detailed maps of Russia and a number of other countries;
- It is possible to build the optimal route;
- Availability of speed radar information;
- The application implements user care (warnings about poor coverage, difficult maneuvers, etc.);
- It is possible to add your information on the map;
- Automatic calculation of the time required to pass traffic jams.

Disadvantages:

- Sophisticated application interface;

- The presence of white spots on the map, not the best map detail.

The application is best used in cities where the application is used by a large number of people – to communicate with other users and actively participate in the development of the local navigation system.

Table 1 has been compiled for a visual comparison of all the above navigation applications according to their main capabilities.

Table 1 – Comparison of the main parameters of applications

Application name	Free version	The need for internet access	Coverage area	Availability of traffic information
Yandex.Navigator	Yes	The application can work both with and without an Internet connection.	Russia, Ukraine, Kazakhstan, Belarus, Azerbaijan, Armenia, Georgia, Abkhazia, Moldova, Uzbekistan, Kyrgyzstan, Tajikistan, Turkey	Yes
Google Maps	Yes	The application can work both with and without an Internet connection.	Russia, UK, parts of Europe, the USA, Canada, Japan, China, Hong Kong	Yes
Maps.me	There are free and paid versions	The application works in offline mode after downloading the maps to the device	The whole world	No
OsmAnd	There are free and paid versions	The application can work both with and without an Internet connection.	Russia, CIS countries, non-CIS countries	Yes
2GIS	Yes	The application can work both with and without an Internet connection.	More than 390 cities in Russia, Kazakhstan, Ukraine, Uzbekistan, Azerbaijan, Italy, Czech Republic, Chile, UAE, Kyrgyzstan, Cyprus	Yes
Waze	Yes	Internet connection required	Russia, Israel, the USA, Canada, UK, France, Germany, Italy, Netherlands, Belgium, South Africa, Ecuador, Chile and Panama	No

Summarizing the information received about the above navigation applications, the following conclusions were made:

- Any navigation application is based on the use of certain sets of geographic data that are created directly for the application itself, or public data is used, for example, such as OpenStreetMap.
- At the heart of any navigation application is the function of building a route from point A to point B.

- The feature set of navigation apps depends on the end user's activity, i.e. whether they will use the app while driving, on foot or for tourism.

1.2 Comparison of AR application development tools and their capabilities

SDK is a kind of augmented reality software. It stands for Software Developer Kit, which is a collection of programs and software used to develop other software.

The main advantage of SDKs is that they allow developers to use typical and time-tested solutions and shortcuts in development, instead of solving all typical and typical problems separately and wasting time on them. The most common ones are listed below.

Apple ARKit

ARKit is used to create applications for both iPhone and iPad that use the A9 processor or higher [2]. ARKit allows developers to design and develop applications that enhance the experience for every user with a variety of features such as:

- Detection and tracking of 2D images
- Recognition and placement of 3D objects
- Detection of horizontal and vertical planes
- Face tracking
- Stable and fast motion tracking
- Scale calculation

Google ARCore

Google ARCore is one of the most commonly used SDKs when developing applications for smartphones and tablets. The AR apps it creates are supported on both Android and iOS devices, allowing you to develop cross-platform AR apps. ARCore is based on two elements: position tracking and object recognition. A small part of its outstanding features are listed below:

- Real-time illumination estimation
- Precise placement of virtual objects
- Easy tracking to create realistic objects

- Determination of the size and location of vertical, horizontal and inclined surfaces
- Motion tracking according to phone position

Vuforia

Vuforia is one of the most popular SDKs for developing augmented reality applications. Using the Vuforia SDK, app developers can demonstrate a wide range of AR app experiences. Vuforia can be used to develop native iOS and Android apps thanks to the API being available through Unity. It is also considered a complete SDK with a rich set of features for AR applications. Simulation mode

- Real-time identification and tracking of target images, English texts and 3D objects
- Placement of virtual objects such as 3D models in a real environment
- Multipurpose 3D configurations
- Vuforia Engine Area Targets along with Area Target Generator
- Scanned Model Targets
- Advanced Model Targets – Multiple Model Detection
- Continuing work when the application is paused
- Simulation mode
- Vuforia Engine Tracking Scale

Wikitude

Wikitude is used in the development of mobile applications and AR prototypes. Wikitude SDK allows developers to implement geolocation capabilities as well as image tracking and object recognition. Some of its features are given below:

- 3D recognition and tracking
- Image recognition and tracking
- Cloud recognition
- AR based on location
- Video overlay
- Smart glasses integration

- Integration with external plugins

MaxST

MaxST is a comprehensive platform that provides a fast and easy AR application development process. It comes as two SDKs: 2D SDK for image recognition and 3D SDK for environment recognition. Its many features and environments allow developers to create AR applications quickly and easily.

- Instant tracking
- Identification of horizontal/vertical planes
- Simultaneous positioning and display to create a "virtual map" to track the environment
- Tracking objects, images and tracking multiple markers
- QR and barcode scanning
- Integration of plugins in Unity

EasyAR

EasyAR is a free SDK for AR development, supporting major mobile platforms. EasyAR allows companies and developers to enhance AR immersion with mobile apps. Developers can use EasyAR features depending on purchased packages:

- Easy AR Basic: developers can manage workflow, improve APIs, ensure compatibility, video playback, QR code scanning and comprehensive integration.
- EasyAR Pro: developers can implement additional features such as 3D object tracking, screen recording, simultaneous detection and tracking of multiple types of markers.

ARToolKit

ARToolKit is an open source SDK that offers many sophisticated features for efficient and fast creation of AR apps and smart glasses software. It offers compiled SDKs for iOS, Android, Windows, Mac OS and Linux platforms. It is one of the first AR SDK with easy to use interface and fast operations. Its many features include:

- Integration of GPS and compass
- Support for Unity3D и OpenSceneGraph

- Recognition of 2D objects
- Simultaneous tracking
- Support for both one and two cameras at once
- Matching additional elements via OpenGL
- Integration with smart glasses
- Supports multiple languages

Xzing

Xzing is augmented reality software for image processing, face and object recognition. There are several Xzing products: Augmented Face Solution, Augmented Vision, and Magic Face, which offer real-time face recognition, image tracking, and deformable face tracking, respectively. Applications can be developed for PC, mobile device, or web browser using the Unity plugin. Its key features include:

- 2D and 3D recognition
- Recognition and tracking of markers
- Face recognition in video
- Face tracking and replacement

Figure 7 shows a comparison of some of the features provided by the SDKs discussed above.

	ARKit	ARCore	Wikitude	EasyAR	Vuforia	MAXST	ARToolKit	XZIMG
2D recognition	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
3D recognition	Yes	Yes	Yes	No	Yes	Yes	No	Yes
Lighting estimation	Yes	Yes	No	No	No	No	No	No
SLAM	Yes	Yes	Yes	Yes	Yes	Yes	No	No
Cloud recognition	Yes	Yes	Yes	Yes	No	Yes	No	No
Geolocation	Yes	Yes	Yes	No	No	No	No	No
Free version	Yes	Yes	Yes (trial only)	Yes	Yes (trial only)	Yes (trial only)	Yes	Yes (trial only)

Figure 7 – Table of comparison of the provided features of the AR SDK

As you can see from Figure 7, not all AR SDKs are completely free. Some of them provide the full range of development opportunities only when subscribing or licensed, or only for testing.

However, all AR SDKs in one way or another try to solve two main problems:

The first problem is related to the ability of augmented reality applications to recognize objects in the real world and track their position. Approaches based on the use of artificial elements, such as markers placed in the environment, are very reliable and work well. The main difficulty of real-time 3D tracking lies in the complexity of the scene and movement of the target objects, including the degrees of freedom of individual objects and their representation. Vision-based tracking aims to link target locations across successive video frames, especially when objects are moving fast relative to the frame rate.

The second problem is focused on the exact depth perception [3]. Stereoscopic displays help with this problem, but additional problems, including conflicts of accommodation and convergence, or low resolution and dim displays, make the subject appear further away than it should be. Using Occlusion solves some depth issues like sequential registration for different viewpoints [4]. The term Occlusion implies that the visible parts of virtual and real objects can overlap each other, and when obtaining the final image, it is necessary to display those parts of the objects that are in the foreground in relation to the user. Thus, the main goal of Occlusion is to preserve the rules of line of sight when creating AR scenes. The complexity of its application lies in the fact that for this it is necessary to obtain the boundaries of objects, in which the overlap of one object with another begins. The solution of this problem in most cases requires high performance of the device to determine objects in the foreground, for which it is necessary to compare a set of point values of objects.

In the fall of 2021, another augmented reality platform for current and future generations of augmented reality equipment Niantic Lightship [5] was announced, which gives developers the opportunity to create a wide range of applications that will contribute to the development of technology.

Their key area of development, real-time mapping, aims to better adapt virtual objects to the real world using smartphone cameras. Lightship strives to ensure that virtual objects have physical properties. This Lightship capability only requires RGB color sensors found in most cameras (unlike LiDAR scanners which are only available in high-end devices). In this regard, this function will be available for a large number of mobile devices.

Figure 8 shows the main features of the Lightship SDK, which mobile application developers can use to create augmented reality scenes.

ARDK Feature and Terms	Before May 1, 2022	After May 1, 2022, Pricing Per Application ¹		
Depth and Occlusion	Free	Free		
Real-Time Meshing	Free	Free		
Semantic Segmentation	Free	Free		
Development Tools	Free	Free		
Multiplayer ³	Free ²	Monthly Active Users⁴ of Multiplayer Service	Cost per 10K Users per Month	MAU Data Transfer Cap
		1–50K MAU per app	Free	50MB per MAU per month per app
		> 50K MAU per app	\$5.00	50MB per MAU per month per app

Figure 8 – Lightship SDK functionality

1.3 Overview of augmented reality navigation applications

Nowadays the leaders in this area are navigation applications by Apple, Google and Yandex. These companies have added augmented reality capabilities to their navigation services over the past five years. They work on the principle of interaction with their provided maps, which use the ability to set the start and end points of the path for further route construction. After building the route, signs are created in the augmented reality mode, which are added at the forks in the path, after receiving text data from the maps. Hereunder, developments of these companies in the field of AR navigation will be considered.

Yandex.Maps

In 2017, Yandex.Maps for iOS 11 mobile devices received augmented reality. You can use the function if your iPhone or iPad has a map update to version 9.6. It is compatible with all A9 based devices (iPhone 6S/SE and later) [6].

Augmented reality in Yandex.Maps works in pedestrian navigation mode. A blue button with an eye icon appears under the map, by clicking on which the gadget turns on the camera and plots a route using digital pointers directly on the real terrain.

This feature can help people who do not work well with simple maps: now navigation does not require any landmarks and attempts to set the map along the route. When the hints are literally superimposed on what you see in front of you, the error is almost eliminated.

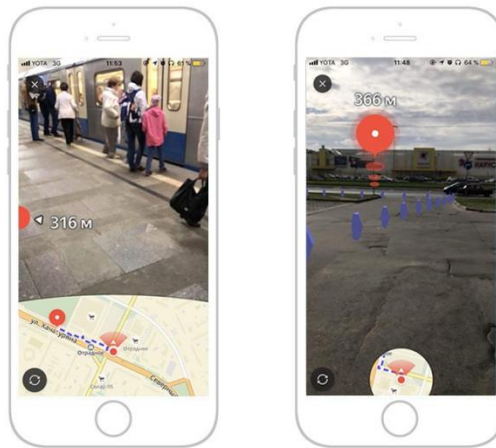


Figure 9 – Example of using AR navigation in Yandex.Maps

Google Maps

In 2019, Google added new features to its mapping app to help users get to a specific location faster. It's about improving the function of augmented reality (AR Live View) [7].

When AR Live View is used and a walking route is built, the application displays 3D arrows and pointers on the screen, which makes it much easier to navigate in unfamiliar terrain. To activate this function, you must press the Directions button and select the Walking option. When using AR mode, a standard Google map with turn-by-turn navigation is displayed at the bottom of the screen.

AR Live View can also be used when traveling by public transport. Signposts will tell you when to get off and where to go next.

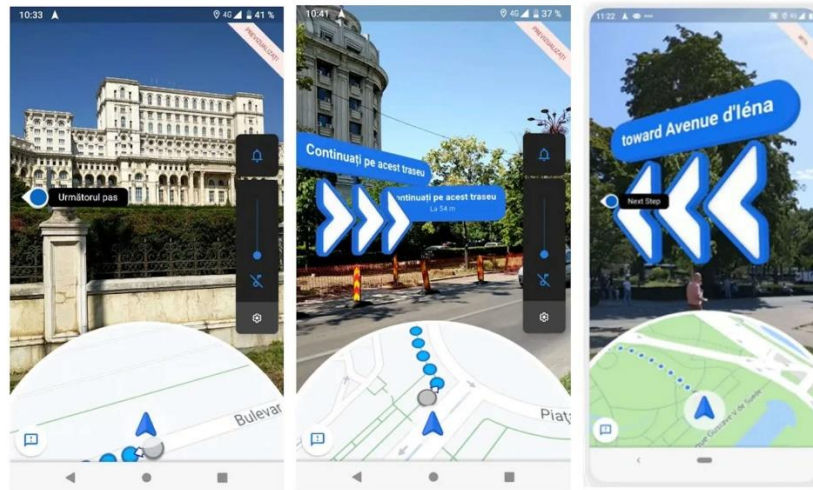


Figure 10 – Example of using AR Live View

Google Maps can also be used for orientation in AR-mode Indoor Live View at airports, shopping centers, railway and bus stations. Once in an unfamiliar place, the user will be able to set a walking route inside the building, after which virtual signs and arrows will appear on the screen that will indicate the path to the nearest stairs, escalator or elevator [8].

To understand where the user is, an artificial intelligence-based system analyzes a large number of images from the Street View service.

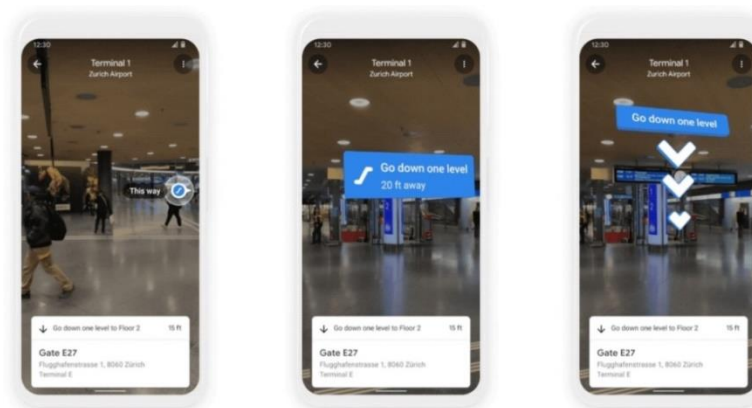


Figure 11 – Example of using Indoor Live View

However, indoor AR navigation is only available in selected malls in the US (Chicago, Long Island, Los Angeles, Newark, San Francisco, San Jose, and Seattle). In the near future, the function will also work in Tokyo and Zurich, and later in other cities around the world.

Apple Maps

On September 27, 2021, an update was added to Apple Maps, part of which consists of immersive walking routes displayed in augmented reality [9]. These immersive walking trails will provide users with turn-by-turn directions in augmented reality. To use this feature, the app asks the user to scan the area they are in to generate a highly accurate position while providing detailed directions that can be viewed in the context of the real world.

In addition to AR navigation, the Apple Maps update has added a 3D maps feature that allows users to navigate cities with a 3D representation of the city that includes elevation information, new road markers, and many custom landmarks.



Figure 12 - Navigation in augmented reality mode in Apple Maps

AR is available from the end of 2021 in major supported cities including London, Los Angeles, New York, Philadelphia, San Diego, San Francisco and Washington [10].

It should also be noted that not only large companies are developing in this area. An example is the AR Indoor Navigation app [11], which was developed for navigating through the office space, within which the company's guests could interactively get to know the client's office and get to the location they need. Also, the application will be used to adapt new employees, which greatly simplifies office navigation. Navigation was carried out in real-time, in which the user could build a route from certain points in space to the required department and see the path in the form of arrows in augmented reality. On the way of the guest, AR content appeared

corresponding to the chosen route, and when approaching the required location, a 3D model of the company symbol was appeared. When choosing a route, the user could read more information about the company and its departments, as well as listen to audio accompaniment. For the development of this project, the Immersal SDK was used, which, using photogrammetry, made it possible to obtain a 3D map of the premises, on the basis of which navigation objects for all necessary routes are.

Another example is Campus Go [12] – a project created in two days at a hackathon on VR and AR technologies. The project has not been implemented, but other companies developing AR / VR applications are very interested in it. The idea was to solve the problem associated with the fact that the FEFU campus is very large and confusing. The Campus Go app is needed to make it easier for event participants to navigate FEFU: it combines the lecture program, campus map and event navigation. The application will select the most convenient route: it will tell you which floor to go to, which door to enter and where to turn.

When launching the application, the user selects one of the events taking place in FEFU. The camera turns on and determines where the person is. Arrows appear on the floor, as well as a plate with the destination and the calculation of the time to the place. While driving, new arrows appear and highlight the desired doors or elevator buttons.

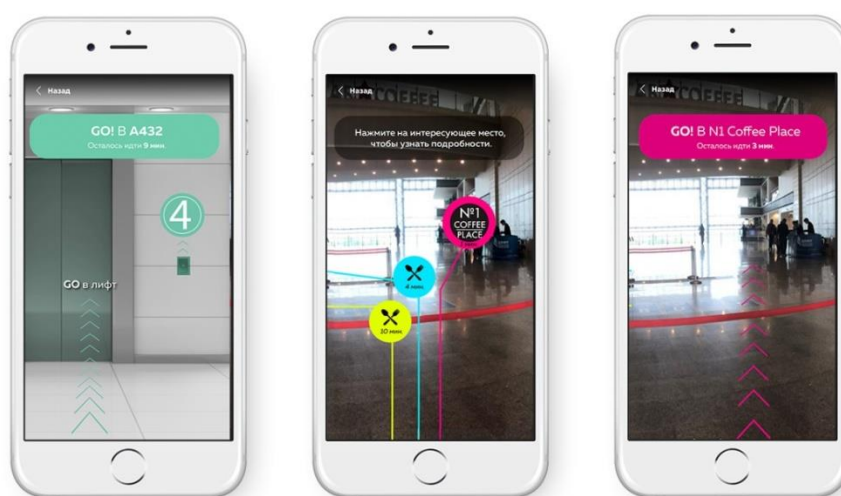


Figure 13 – Example of the Campus Go application

After analyzing the applications presented in this section, two main directions in the development of navigation applications with augmented reality support can be distinguished: narrowly focused navigation for specific organizations and navigation through public places by setting route endpoints. In the first case, applications are aimed at familiarizing the user with the organization itself and facilitating their navigation through a large intricate structure. In the second case, applications are aimed to switch from the classic version of navigation using 2D maps to navigation around the real world by introducing additional elements into it, eliminating the possibility of the user moving in the wrong direction.

It is also worth noting that these applications plot the route as follows: the current position of the user and the end point of the route are analyzed, after which the path is built along which it will be necessary to move, and then, based on its length and changes in the altitude level, the points to which will be added navigation virtual objects.

In fact, the construction of such routes is far from a trivial task and requires certain calculations. In this regard, it was decided to develop an application that will allow creating a navigation object system for TPU. In this case, the objects of this system will be added by the user himself instead of using path-laying algorithms. In addition to this, the application being developed must also have a classic version of the navigation application, which will make sure that the specified locations are correct.