

Школа Инженерная школа информационных технологий и робототехники  
 Направление подготовки 09.03.01 Информатика и вычислительная техника  
 Отделение школы (НОЦ) Отделение информационных технологий

### БАКАЛАВРСКАЯ РАБОТА

Тема работы
<b>Разработка клиент-серверного приложения для обработки маршрутов движения мобильных устройств</b>

УДК 004.455.1:004.715:656.022

Студент

Группа	ФИО	Подпись	Дата
8В6Б	Евстафиевская Анастасия Эдуардовна		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Стоянов Александр Кириллович	к.т.н.		

Консультант ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ОИТ	Дорофеев Вадим Анатольевич	-		

### КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН	Конотопский Владимир Юрьевич	к.т.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ООТД	Матвиенко Владимир Владиславович	-		

### ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Погребной Александр Владимирович	к.т.н.		

## ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

Код результата	Результат обучения (выпускник должен быть готов)
P1	Применять базовые и специальные естественнонаучные и математические знания в области информатики и вычислительной техники, достаточные для комплексной инженерной деятельности
P2	Применять базовые и специальные знания в области современных информационных технологий для решения инженерных задач
P3	Ставить и решать задачи комплексного анализа, связанные с созданием аппаратно-программных средств информационных и автоматизированных систем, с использованием базовых и специальных знаний, современных аналитических методов и моделей
P4	Разрабатывать аппаратные и программные средства (системы, устройства, блоки, программы, базы данных и т.п.) в соответствии с техническим заданием и с использованием средств автоматизации проектирования
P5	Проводить теоретические и экспериментальные исследования в области создания аппаратных и программных средств информационных и автоматизированных систем, включающие поиск и изучение необходимой научно-технической информации, математическое моделирование, проведение эксперимента, анализ и интерпретацию полученных данных
P6	Внедрять, эксплуатировать и обслуживать современные программно-аппаратные комплексы, обеспечивать их высокую эффективность, соблюдать правила охраны здоровья, безопасность труда, выполнять требования по защите окружающей среды
P7	Использовать базовые и специальные знания в области проектного менеджмента для ведения комплексной инженерной деятельности
P8	Владеть иностранным языком на уровне, позволяющем работать в иноязычной среде, разрабатывать документацию, презентовать и защищать результаты комплексной инженерной деятельности
P9	Эффективно работать индивидуально и в качестве члена группы, состоящей из специалистов различных направлений и квалификаций, демонстрировать ответственность за результаты работы и готовность следовать корпоративной культуре организации
P10	Демонстрировать знания правовых, социальных, экономических и культурных аспектов комплексной инженерной деятельности
P11	Демонстрировать способность к самостоятельному обучению в течение всей жизни и непрерывному самосовершенствованию в инженерной профессии

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

Инженерная школа информационных технологий и робототехники  
 Направление подготовки 09.03.01 Информатика и вычислительная техника  
 Отделение информационных технологий

УТВЕРЖДАЮ:  
 Руководитель ООП  
 09.03.01 Информатика и  
 вычислительная техника  
 \_\_\_\_\_  
 (Ф.И.О.)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

В форме:

Бакалаврской работы
---------------------

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

Студенту:

Группа	ФИО
8В6Б	Евстафиевской Анастасии Эдуардовне

Тема работы:

Разработка клиент-серверного приложения для обработки маршрутов движения мобильных устройств	
Утверждена приказом директора (дата, номер)	28.02.2020, 5.9-501с

Срок сдачи студентом выполненной работы:	04.06.2020
--	------------

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<b>Исходные данные к работе</b>	Литературные данные по разработке по разработке приложений под платформу Android, техническая документация Android SDK. Техническая документация по работе с Google Firebase.
---------------------------------	---

<b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b>	<ol style="list-style-type: none"> <li>1) Исследование принципов работы архитектуры «клиент-сервер»;</li> <li>2) Обзор способов определения местоположения;</li> <li>3) Обзор существующих на рынке приложений для отслеживания мобильных устройств; оценка их популярности и конкурентоспособности;</li> <li>4) Выбор кодовой базы и инструментария для разработки мобильного приложения;</li> <li>5) Разработка мобильного приложения;</li> <li>6) Тестирование мобильного приложения;</li> <li>7) Оценка и анализ готового продукта.</li> </ol>
<b>Перечень графического материала</b>	Скриншоты готового приложения

**Консультанты по разделам выпускной квалификационной работы**

Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Конотопский Владимир Юрьевич
Социальная ответственность	Матвиенко Владимир Владиславович
Техническое задание, Обзор литературы, Реализация модуля	Дорофеев Вадим Анатольевич

<b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b>	01.02.2020
---	------------

**Задание выдал руководитель:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Стоянов А.К.	к.т.н		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8В6Б	Евстафиевской Анастасии Эдуардовне		

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

Инженерная школа информационных технологий и робототехники  
 Направление подготовки 09.03.01 Информатика и вычислительная техника  
 Уровень образования бакалавриат  
 Отделение информационных технологий  
 Период выполнения весенний семестр 2019/2020 учебного года

Форма представления работы:

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

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

### КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	04.06.2020
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
10.02.2020	Сбор и анализ информации для раздела «Обзор литературы»	15
08.03.2020	Поиск инструментов для разработки приложения	25
21.05.2020	Проектирование приложения и его реализация	40
04.06.2020	Оформление и представление ВКР	20

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Стоянов А.К.	к.т.н		

**СОГЛАСОВАНО:**

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Погребной А.В.	к.т.н		

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

Студенту:

<b>Группа</b>	<b>ФИО</b>
8В6Б	Евстафиевская Анастасия Эдуардовна

<b>Школа</b>	<b>ИШИТР</b>	<b>Отделение школы (НОЦ)</b>	<b>ОАР</b>
Уровень образования	Бакалавриат	Направление/специальность	09.03.01 Информатика и вычислительная техника

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

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Использовать действующие ценники и договорные цены на потребленные материальные и информационные ресурсы, а также указанную в МУ величину тарифа на эл. энергию
2. <i>Нормы и нормативы расходования ресурсов</i>	—
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	Действующие ставки единого социального налога и НДС (см. МУ)

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

1. <i>Планирование процесса управления НИИ: структура и график проведения, бюджет, риски и организация закупок</i>	Построение плана-графика выполнения ВКР, составление соответствующей сметы затрат, расчет величины НДС и цены результата ВКР
--	--

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

1. Оценка конкурентоспособности технических решений
2. Матрица SWOT
3. Альтернативы проведения НИ
4. График проведения и бюджет НИ - выполнить  
Оценка ресурсной, финансовой и экономической эффективности НИ- выполнить

**Дата выдачи задания для раздела по линейному графику**

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Доцент	Конотопский Владимир Юрьевич	к. э. н.		28.02.2020 г.

**Задание принял к исполнению студент:**

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
8В6Б	Евстафиевская Анастасия Эдуардовна		28.02.2020 г.

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

<b>Группа</b>	<b>ФИО</b>
8В6Б	Евстафиевской Анастасии Эдуардовне

<b>Школа</b>	<b>ИШИТР</b>	<b>Отделение (НОЦ)</b>	<b>ОИТ</b>
Уровень образования	бакалавриат	Направление/специальность	09.03.01 Информатика и вычислительная техника

Тема ВКР:

Разработка клиент-серверного приложения для обработки маршрутов движения мобильных устройств	
<b>Исходные данные к разделу «Социальная ответственность»:</b>	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	<p>Объект исследования является разработка клиент-серверного приложения для обработки маршрутов движения мобильных устройств.</p> <p>Область применения – сфера бизнеса, курьерские службы и контроль детей.</p> <p>Рабочая зона разработчика – КЦ НИ ТПУ.</p>
<b>Перечень вопросов, подлежащих исследованию, проектированию и разработке:</b>	
<p><b>1. Правовые и организационные вопросы обеспечения безопасности:</b></p> <ul style="list-style-type: none"> <li>– специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства;</li> <li>– организационные мероприятия при компоновке рабочей зоны.</li> </ul>	<ul style="list-style-type: none"> <li>– Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 16.12.2019) [11];</li> <li>– СанПиН 2.2.2/2.4.1340-03 “Гигиенические требования к персональным электронно-вычислительным машинам и организации работы” [12];</li> <li>– ССБТ ГОСТ 12.2.032-78 “Рабочее место при выполнении работ сидя. Общие эргономические требования”[13].</li> </ul>
<p><b>2. Производственная безопасность:</b></p> <p>2.1. Анализ вредных и опасных факторов при проведении исследований и при использовании объекта исследования и обоснование мероприятий по защите исследователя</p>	<p>Возможны следующие вредные и опасные факторы:</p> <ul style="list-style-type: none"> <li>– электромагнитные поля;</li> <li>– электростатические поля;</li> <li>– шум;</li> <li>– нарушения микроклимата в помещении;</li> <li>– нарушения освещенности рабочей зоны;</li> <li>– психофизические факторы.</li> </ul>
<p><b>3. Экологическая безопасность:</b></p>	<p>Влияние на окружающую среду проявляется в:</p> <ul style="list-style-type: none"> <li>– увеличении энергопотребления;</li> <li>– отсутствии утилизации ПЭВМ;</li> <li>–отсутствии утилизации батарей смартфонов;</li> <li>– отсутствии утилизации макулатуры.</li> </ul>
<p><b>4. Безопасность в чрезвычайных ситуациях:</b></p>	<p>Чрезвычайная ситуация техногенного характера для рассматриваемого случая – пожар.</p>

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

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
ассистент ООТД	Матвиенко Владимир Владиславович			28.02.2020 г.

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8В6Б	Евстафиевская Анастасия Эдуардовна		28.02.2020 г.



## Реферат

Выпускная квалификационная работа содержит 91 страницу, 35 рисунков, 12 таблиц, 16 литературных источников и приложение.

Ключевые слова: мобильная разработка, базы данных, отслеживание местоположения, Android Studio, GPS, Google Maps, Firebase.

Цель работы – проектирование и разработка клиент-серверного приложения для обработки маршрутов движения мобильных устройств. Данная цель обусловлена необходимостью точного определения географического местоположения владельцев мобильных устройств для их последующей координации.

Объектом исследования является разработка клиент-серверного приложения для обработки маршрутов движения мобильных устройств.

Область применения – сфера бизнеса, курьерские службы и контроль детей.

Приложение применяется для контроля перемещений частично или полностью недееспособных людей (пенсионеры, инвалиды, дети), а также сотрудников служб логистики и доставки.

## Определения и сокращения

В данной работе используются следующие термины:

- **JAVA:** объектно-ориентированный язык программирования;
- **БД** (база данных): структура взаимосвязанных, хранящихся вместе данных, организованная таким образом, что обеспечивается безызбыточное хранение данных, их комплексное использование и независимость программ обработки от физической структуры;
- **GPS:** спутниковая система навигации, обеспечивающая измерение расстояния, времени и определяющая местоположение во всемирной системе координат WGS 84;
- **ГЛОНАСС:** глобальная навигационная спутниковая система;
- **GSM:** глобальный стандарт цифровой мобильной сотовой связи с разделением каналов по времени и частоте;
- **LBS:** программный сервис, использующий данные о локации для управления какими-либо функциями;
- **XML:** расширяемый язык разметки документов;
- **URL:** единообразный указатель местонахождения ресурса;
- **API:** описание способов (набор классов, процедур, функций, структур или констант), которыми одна компьютерная программа может взаимодействовать с другой программой;
- **ОС:** операционная система.

## Оглавление

Введение .....	14
1 Техническое задание.....	15
1.1 Основные задачи и цели.....	15
1.2 Требования к модулю .....	15
2 Обзор литературы .....	16
2.1 Основные сведения о клиент-серверных приложениях .....	16
2.1.1 Front-end .....	16
2.1.2 Back-end.....	16
2.1.3 Преимущества и недостатки клиент-серверной архитектуры	17
2.2 Способы определения местоположения .....	18
2.2.1 Точное местонахождение по GPS .....	18
2.2.2 Примерное местоположение по вышкам сотовой связи	
GSM/LBS	18
3 Обзор аналогов.....	18
3.1 Где мои дети.....	19
3.2 Zoemob.....	19
3.3 Sygic Family.....	20
4 Реализация модуля.....	21
4.1 Построение архитектуры приложения .....	21
4.2 Выбор инструментария.....	22
4.3 Структура БД .....	22
4.4 Описание работы приложения .....	25
5 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение.....	46
Введение .....	46
5.1 Организация и планирование работ.....	46

5.1.1	Продолжительность этапов работ.....	47
5.2	Расчет сметы затрат на выполнение проекта .....	51
5.2.1	Расчет затрат на материалы.....	51
5.2.2	Расчет заработной платы .....	52
5.2.3	Расчет затрат на социальный налог .....	52
5.2.4	Расчет затрат на электроэнергию.....	53
5.2.5	Расчет амортизационных расходов.....	54
5.2.6	Расчет расходов, учитываемых непосредственно на основе платежных (расчетных) документов (кроме суточных) .....	55
5.2.7	Расчет прочих расходов.....	55
5.2.8	Расчет общей себестоимости разработки .....	56
5.2.9	Расчет прибыли .....	56
5.2.10	Расчет НДС .....	56
5.2.11	Цена разработки НИР .....	56
5.3	Оценка экономической эффективности проекта .....	57
6	Социальная ответственность.....	58
6.1	Правовые и организационные вопросы обеспечения безопасности .....	58
6.1.1	Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства.....	58
6.1.2	Организационные мероприятия при компоновке рабочей зоны. 59	
6.2	Профессиональная социальная безопасность .....	61
6.2.1	Анализ вредных и опасных факторов, которые может создать объект исследования и которые могут возникнуть на рабочем месте при проведении исследования .....	61

6.2.2 Обоснование мероприятий по защите исследователя от действия опасных и вредных факторов.....	67
6.3 Экологическая безопасность.....	68
6.3.1 Анализ влияния объекта и процесса исследования на окружающую среду.....	68
6.3.2 Обоснование мероприятий по защите окружающей среды. .	68
6.4 Безопасность в чрезвычайных ситуациях.....	69
6.4.1 Анализ вероятных ЧС, которые может инициировать объект исследований и которые могут возникнуть на рабочем месте при проведении исследований.....	69
6.4.2 Обоснование мероприятий по предотвращению ЧС и разработка порядка действий в случае возникновения ЧС.....	69
Заключение.....	71
Список использованных источников.....	72
Приложение А.....	74
MainActivity.java.....	74
AdapterUsers.java.....	78
InboxFragment.java.....	80
OutboxFragment.java.....	87

## **Введение**

В современном мире большинство людей придерживаются активного образа жизни, когда в каждом дне много дел и перемещений. В связи с постоянной занятостью работающих граждан им становится тяжело нести ответственность за людей, находящихся под их опекой, без какой-либо помощи. Так, дети остаются предоставлены самим себе и рискуют оказаться не в том месте и не в то время; пенсионеры, страдающие какими-либо заболеваниями, в частности обладающие проблемами с памятью, могут, покинув дом, заблудиться и не вернуться обратно.

Также в настоящее время очень хорошо развита система обслуживания клиентов посредством доставки. Обеспечение людей необходимыми товарами требует наличия свободных курьеров, находящихся максимально близко к нужному адресу. Здесь службы, связанные с логистикой, сталкиваются со сложностями грамотного распределения человеческих, технических и временных ресурсов.

В сфере бизнеса тоже является необходимым контроль сотрудников, в обязанности которых входит проведение различных деловых встреч в заданное время в подходящем месте. Работодателю важно знать, что сотрудник в рабочие часы находится действительно в каком-либо из установленных рабочих мест.

Решением вышеизложенных проблем является создание приложения, позволяющего определять точное местоположение мобильных устройств для последующей координации и контроля действий их владельцев.

# **1 Техническое задание**

## **1.1 Основные задачи и цели**

Целью проекта является разработка клиент-серверного приложения для обработки маршрутов движения мобильных устройств.

Для реализации главной цели необходимо выполнить ряд следующих задач:

- 1) Исследование принципов работы архитектуры «клиент-сервер»;
- 2) Обзор способов определения местоположения;
- 3) Обзор существующих на рынке приложений для отслеживания мобильных устройств; оценка их популярности и конкурентоспособности;
- 4) Выбор кодовой базы и инструментария для разработки мобильного приложения;
- 5) Разработка мобильного приложения;
- 6) Тестирование мобильного приложения;
- 7) Оценка и анализ готового продукта.

## **1.2 Требования к модулю**

В рамках индивидуального задания по разработке проекта выделены следующие пункты:

- 1) Разработка структуры бд;
- 2) Внедрение инструментов бд в проект;
- 3) Обработка всех необходимых данных пользователей;
- 4) Обеспечение режима реального времени.

## **2 Обзор литературы**

### **2.1 Основные сведения о клиент-серверных приложениях**

Написание программного кода любого клиент-серверного приложения делится на FRONT-end программирование и BACK-end программирование [1].

#### **2.1.1 Front-end**

Front-end программирование – это разработка клиентской части приложения, то есть реализация взаимодействия между пользователем и серверной частью приложения. Данная часть отвечает за ввод входной информации от пользователя, а также ее первичную обработку и отправку на сервер по соответствующему API.

Клиентская часть делится на 2 типа: нативная и кроссплатформенная.

Нативные приложения — это приложения, разрабатываемые на «родном» языке платформы (приложения для iOS, Android и др.);

Кроссплатформенные приложения – это приложения, разрабатываемые с помощью различных «фреймворков» (например, Adobe AIR, PhoneGap, IBM Worklight).

#### **2.1.2 Back-end**

Back-end программирование – это разработка серверной части приложения, которая отвечает за передачу данных между пользователями или ресурсами.

Составляющие серверной части:

1. Серверная архитектура.

Проектируется и разрабатывается развернутая серверная архитектура: алгоритмы загрузки данных, методы авторизации, кеширования и многое другое.

2. API.



API (application programming interface) – это интерфейс прикладного программирования, то есть набор запросов к серверу, который последний понимает и может дать корректный ответ.

### 3. Административная панель.

Административная панель – это главный инструмент управления мобильным приложением. Для каждого проекта создаётся отдельный Web-интерфейс управления. Функционал панели разрабатывается исходя из целей и задач проекта. Все изменения, произведённые в административной панели, моментально применяются к мобильному приложению.

### 4. Метрики.

Метрики – это статистические данные по приросту пользователей, активности (лайки, комментарии, сообщения, просмотры, конверсии), ежедневной посещаемости проекта в целом и его отдельных разделов.

#### **2.1.3 Преимущества и недостатки клиент-серверной архитектуры**

Преимуществом модели взаимодействия клиент-сервер является то, что программный код клиентского приложения и серверного разделен [2].

К преимуществам архитектуры клиент-сервер можно отнести пониженные требования к машинам клиентов, так как большая часть вычислительных операций будет производиться на сервере, а также архитектура клиент-сервер довольно гибкая и позволяет администратору сделать локальную сеть более защищенной.

К недостаткам модели взаимодействия клиент-сервер можно отнести то, что стоимость серверного оборудования значительно выше клиентского. Сервер должен обслуживать специально обученный и подготовленный человек. Если в локальной сети возникают сбои на сервере, то и клиенты не смогут работать.

Также стоит отметить, что архитектура «клиент-сервер» не делит машины на «только клиент» или «только сервер», а скорее позволяет распределить нагрузку и разделить функционал между клиентской частью и серверной.

## **2.2 Способы определения местоположения**

Для определения местонахождения человека применяются две основные технологии: GPS и LBS [3].

### **2.2.1 Точное местонахождение по GPS**

Сигналы спутниковой навигации покрывают всю планету и обеспечивают высочайшую точность. Остается лишь передать информацию о местоположении человека наблюдателю. По этому принципу работают как специальные GPS-маячки или GPS-трекеры, так и мобильные приложения для телефонов и планшетов, оборудованных GPS-приемником. Главным достоинством этого способа является высокая точность, поэтому на фоне снижения стоимости устройств и гаджетов, он становится все более популярным.

### **2.2.2 Примерное местоположение по вышкам сотовой связи GSM/LBS**

Вычисление местоположения человека на основе сигналов GSM/LBS тоже широко распространено, однако точность заметно меньше, чем у GPS. Суть технологии в том, чтобы анализировать информацию о тех базовых станциях сотовой связи, которые находятся поблизости. На сегодня этот метод применяется, в основном, как резервный, дополнительный к GPS/ГЛОНАСС. GPS-трекеры и мобильные приложения передают эти данные, и если GPS/ГЛОНАСС сигнал недоступен (в метро, в здании без окон), то определение местонахождения выполняется по базовым станциям.

## **3 Обзор аналогов**

На сегодняшний день уже существуют реализации приложений, обеспечивающих отслеживание мобильных устройств. Для исследования выберем ТОП-3 по оценке пользователей Google Play и App Store [4]: Где мои дети, Zoemob и Sygic Family.

### **3.1 Где мои дети**

Разработчики мобильного локатора «Где мои Дети» [5] предлагают программы для смартфонов или смарт-часов.

Программа предоставляет следующую информацию:

- карту перемещений;
- посещение безопасных зон из списка (например, школы);
- звуки с микрофона;
- анализ разговоров и использования приложений, в том числе подробную статистику.

Также ребенок сможет нажать тревожную кнопку при возникновении сложной ситуации.

Разработчики предлагают три варианта лицензии «Контроль местоположения ребенка онлайн»:

- месячная на 1 устройство за 169 рублей;
- годовая на 3 устройства за 990 рублей;
- вечная на 3 устройства за 1490 рублей.

Прослушивание окружения ребенка также разделено на 3 варианта:

- 30 минут за 59 рублей;
- 180 минут за 279 рублей;
- безлимит на месяц за 990 рублей.

### **3.2 Zoemob**

В отличие от предыдущего варианта, данное приложение не является именно детским локатором, а отображает местоположение и скорость всех членов семьи.

Кроме этого мобильная программа Zoemob [6] предлагает:

- внутренний мессенджер с закрытым каналом для общения;
- семейный календарь, чтобы строить общие планы;
- тревожную кнопку для вызова родственников.

Zoetob получает точные данные GPS-приемника смартфона, но, если сигнала нет или приемник выключен, приложение будет ориентироваться на местоположение ближайшей вышки базовой станции сотового оператора или Wi-Fi роутеры.

Программа доступна для ОС IOS и Android.

### **3.3 Sygic Family**

Приложение Sygic Family [7] позволяет:

- Показывать положение членов семьи на карте;
- Создавать список безопасных и опасных геозон;
- Оповещать о том, что кто-то в них зашел или из них вышел;
- Отправлять сигнал SOS остальным членам семьи;
- Показывать историю перемещения близких;

Премиум версия предлагает следующие платные функции:

- Отслеживание в реальном времени;
- Неограниченное количество геозон;
- Недельную историю маршрутов;
- Отслеживание рейсов с сообщениями о взлете и посадке.

Доступны 2 подписки, — годовая за 586,89 рубля и бесконечная за 990 рублей.

Среди недостатков исследуемых приложений выявлены следующие:

- Отсутствие бесплатной версии, что повлечет за собой денежные затраты;
- Узкая направленность приложений, обеспечивают исключительно контроль внутри семьи.

## 4 Реализация модуля

### 4.1 Построение архитектуры приложения

На основе анализа существующих аналогов приложения определена схема реализации проекта.

Основными функциями приложения определены:

- Регистрация и авторизация пользователя с использованием адреса электронной почты и пароля;
- Определение текущего местоположения вошедшего пользователя с последующим отображением на карте;
- Получение списка всех пользователей, зарегистрированных в системе, с возможностью отправки им заявок на отслеживание;
- Обработка входящих и исходящих заявок на отслеживание;
- Получение списка всех пользователей, разрешивших отслеживание;
- Возможность удаления отслеживаемых пользователей из списка;
- Получение списка всех пользователей, которым вошедший в приложение пользователь разрешил отслеживание;
- Возможность отмены разрешения на отслеживание конкретным пользователям;
- Отображение на карте пользователей, принявших заявку на отслеживание;
- Возможность поиска мест на карте;
- Возможность приближения, отдаления и возврата к текущему местоположению на карте;
- Отображение данных профиля вошедшего пользователя с возможностью их редактирования;
- Возможность выхода из профиля вошедшего пользователя.

## **4.2 Выбор инструментария**

В качестве инструмента для разработки мобильного приложения выбрана среда Android Studio, язык программирования Java. Для определения местоположения используется GPS. Для обработки всех получаемых и вводимых пользователями данных используется Firebase.

Android Studio [8] – это интегрированная среда разработки для работы с платформой Android. Предоставляет возможность работы с сервисами Google и широким спектром графических библиотек, а также позволяет конвертировать проект в готовый установочный APK-файл.

Java [9] – это строго типизированный объектно-ориентированный язык программирования. Является официальным языком для разработки мобильных приложений под Android, поддерживается средой Android Studio и поддерживается компанией Google.

Firebase [10] – это сервис компании Google, предоставляющий услуги БД реального времени и облачного хранилища. Сервис предоставляет разработчикам бесплатное использование услуг для зарегистрированных в системе Firebase проектов при условии, что количество пользователей приложения не превышает установленный лимит.

## **4.3 Структура БД**

Приведенные ниже рисунки отражают структуру базы данных мобильного приложения.

Идентификатор	Поставщики	Время создания	Последний вход	Уникальный идентификатор пользователя ↑
savchsm98@gmail.com	✉	3 июн. 2020 г.	4 июн. 2020 г.	00N85mUh0uSZu0ZgDhELenf8I792
ksu_ukolova_98@mail.ru	✉	3 июн. 2020 г.	4 июн. 2020 г.	CJ6TrXR1ArU9YJ2i5ArVjGvfy3y1
kpv1@tpu.ru	✉	3 июн. 2020 г.	3 июн. 2020 г.	EE2KuvHyAaNjMs60LkEp6huTTEa2
dva@tpu.ru	✉	4 июн. 2020 г.	4 июн. 2020 г.	gvXc9kFFo3NizrFtyvvAWya0YqS2
test@gmail.com	✉	4 июн. 2020 г.	4 июн. 2020 г.	hBnJSI78EocKm1L4jcrIXtmin9v1
q@msn.com	✉	3 июн. 2020 г.	4 июн. 2020 г.	haaoKfi4oQTdQml21TC8XEYyRnU2

Количество строк на странице: 50 ▾ 1-6 из 6 < >

Рисунок 1 – Аутентификация пользователей

При регистрации пользователей в приложении введенные адрес электронной почты и пароль сохраняются во встроенной в Firebase системе аутентификации пользователей для подключенного проекта. Каждому пользователю присваивается уникальный ключ в зашифрованном виде, который будет хранить всю добавляемую о пользователе информацию (программный код метода представлен в классе MainActivity.java Приложения А).

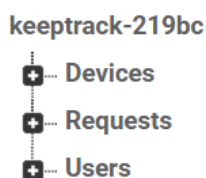


Рисунок 2 – Структура базы данных проекта

База данных проекта включает в себя три таблицы:

1. Users – хранящая данные о всех зарегистрированных пользователях;
2. Requests – хранящая данные о входящих и исходящих заявках пользователей на отслеживание;
3. Devices – хранящая данные о подтвержденных заявках пользователей на отслеживание.

Все таблицы создаются программно при вызове в приложении соответствующих методов. При создании данных поля таблиц окрашиваются в зеленый цвет, при обновлении - в желтый, при удалении - в красный.

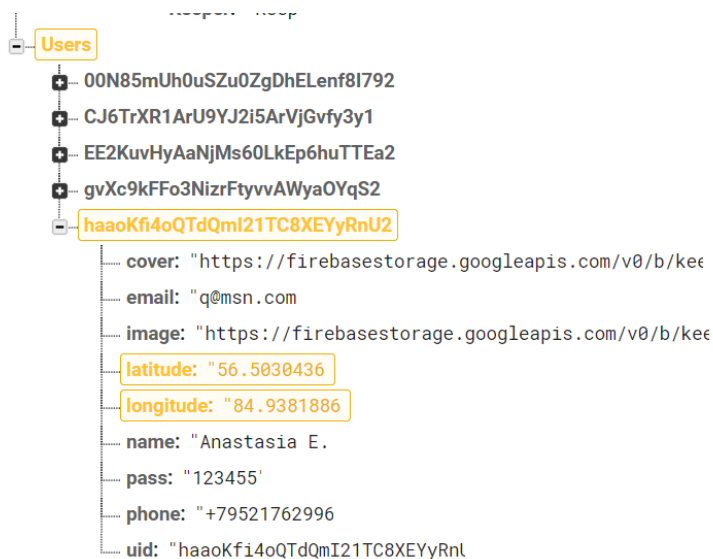


Рисунок 3 – Таблица Users

Данная таблица хранит всю получаемую в процессе работы приложения информацию о каждом конкретном пользователе. Поскольку БД работает в режиме реального времени, данные о местоположении пользователя постоянно обновляются. В БД обновления регистрируются изменением цвета соответствующих полей на желтый.

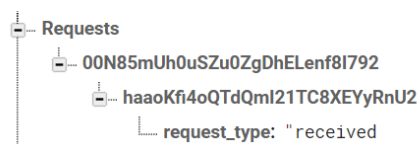


Рисунок 4 – Таблица Requests

Данная таблица создается при отправлении пользователями заявок на отслеживание друг друга. В таблице появляется поле идентификатора пользователя, отправившего заявку, затем поле идентификатора пользователя, получившего заявку и поле “request\_type” у пользователя, получившего заявку, имеющее значение “received” (программный код метода представлен в классе AdapterUsers.java Приложения А).



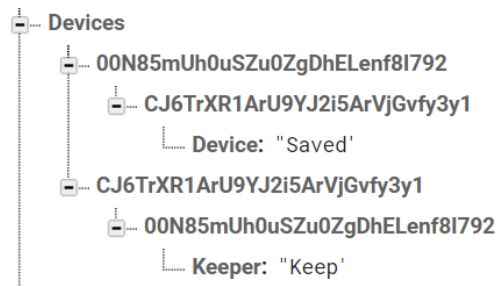


Рисунок 5 – Таблица Devices

В данную таблицу помещаются пользователи, одобрившие заявки на отслеживание. Таблица заполняется симметрично с противоположными значениями. Пользователь, разрешивший отслеживание, получает статус “Saved” в поле “Device” у пользователя, которому одобрил заявку. Тот, кому заявка одобрена, получает статус “Keep” в поле “Keeper” того пользователя, который разрешил его отслеживать. При этом отслеживание может быть односторонним (программный код метода представлен в классах InboxFragment.java и OutBox.java Приложения А).

#### 4.4 Описание работы приложения

Приведенные ниже рисунки демонстрируют функционирование разработанного приложения.



Рисунок 6 – Ярлык приложения на реальном устройстве пользователя

Разработанное приложение носит название KeepTrack. Ярлык приложения представляет собой первую букву названия, помещенную внутрь лупы зеленого цвета. Логотип выполнен на белом фоне.

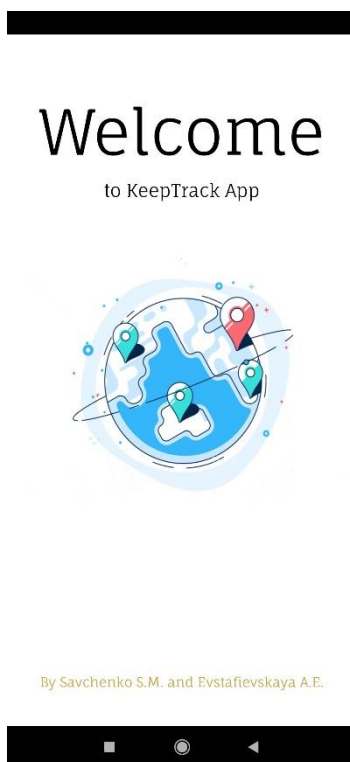


Рисунок 7 – Заставка приложения

После нажатия на ярлык приложения начинается его запуск, во время которого на экране мобильного устройства появляется заставка.



Рисунок 8 – Стартовая страница приложения

На стартовой странице приложения располагаются кнопки авторизации и регистрации.

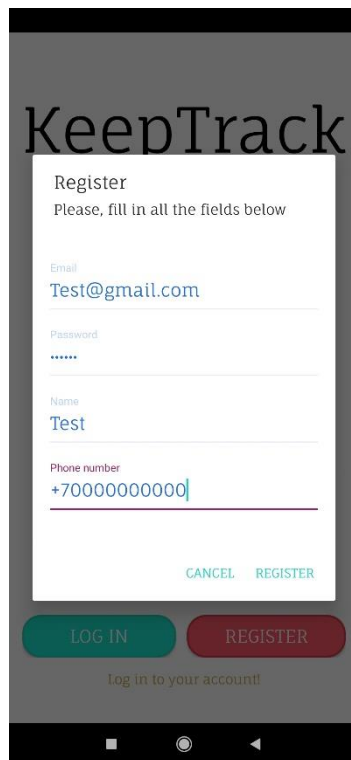


Рисунок 9 – Форма регистрации

Регистрационная форма включает в себя такие поля, как адрес электронной почты, пароль, имя пользователя и номер телефона. На каждое поле установлены ограничения и соответствующая обработка. Имеются кнопки отмены и подтверждения регистрации.

---

# KeepTrack

Authorization Form

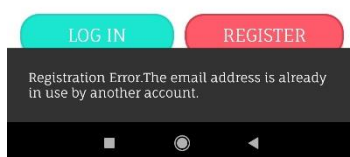


Рисунок 10 – Сообщение об ошибке регистрации

В случае, если пользователь пытается зарегистрировать уже имеющиеся данные, выведется сообщение об ошибке.

---

# KeepTrack

Authorization Form

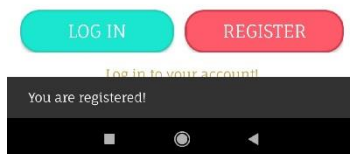


Рисунок 11 – Сообщение об успешной регистрации

Если все поля заполнены и пользователь регистрируется впервые, появится сообщение о том, что он зарегистрирован, а все полученные данные запишутся в БД.

The image shows a mobile application interface for 'KeepTrack'. At the top, the app name 'KeepTrack' is displayed in a large, dark font, with 'Authorization Form' underneath it in a smaller font. Below this is a white rectangular form titled 'Log In'. Inside the form, there is a prompt 'Please, fill in all the fields below'. There are two input fields: 'Email' containing the text 'savchsm98@gmail.com' and 'Password' containing six dots. At the bottom right of the form are two buttons: 'CANCEL' and 'LOG IN'. Below the form, there are two larger buttons: a teal 'LOG IN' button and a dark red 'REGISTER' button. At the very bottom of the screen, there is a small text prompt 'Log in to your account!'. The entire interface is set against a dark grey background.

Рисунок 12 – Форма авторизации

Авторизация пользователя происходит посредством ввода адреса электронной почты и пароля, зарегистрированных в базе.

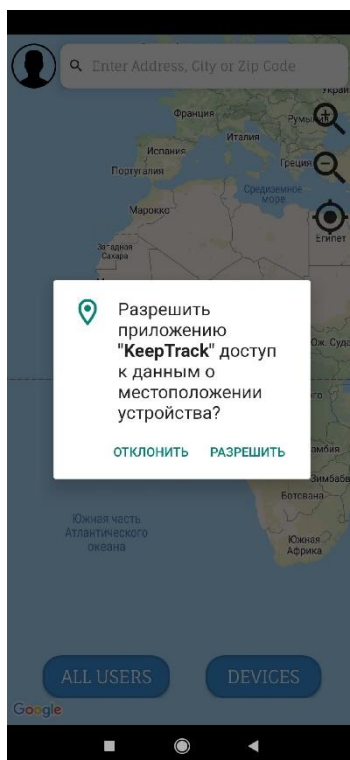


Рисунок 13 – Запрос разрешения на определение местоположения

После авторизации пользователю выводится запрос на разрешение приложению доступа к данным о местоположении устройства.

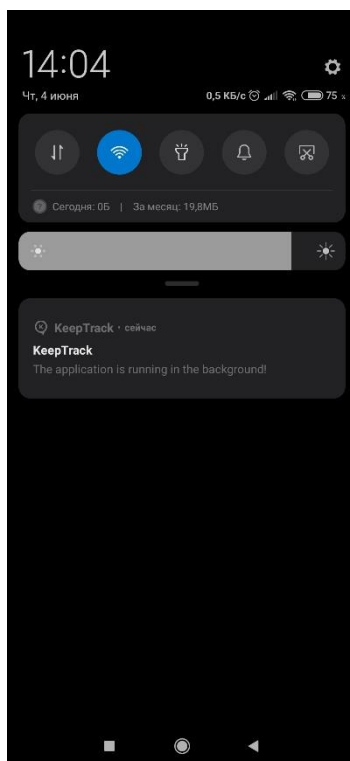


Рисунок 14 – Уведомление о работе приложения

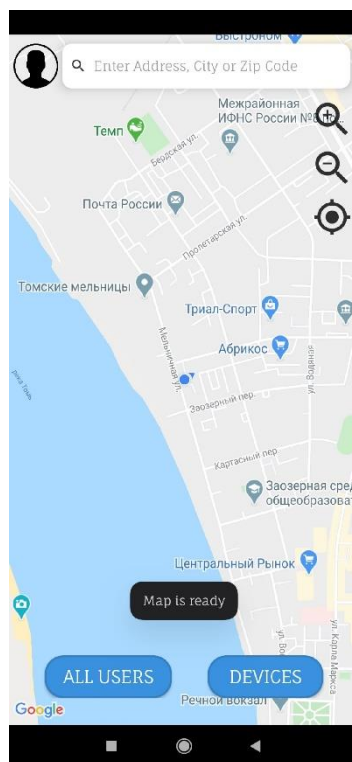


Рисунок 15 – Основная страница приложения

После получения разрешения на определения местоположения устройства текущие координаты пользователя записываются в БД, а на основной странице приложения открывается карта и в панели уведомлений появляется сообщение о работе приложения в фоновом режиме (Рисунок 14).

В месте нахождения пользователя отображается синий маячок. На странице предусмотрены функции приближения, отдаления карты, а также возвращения объектива камеры на текущее местоположение пользователя.



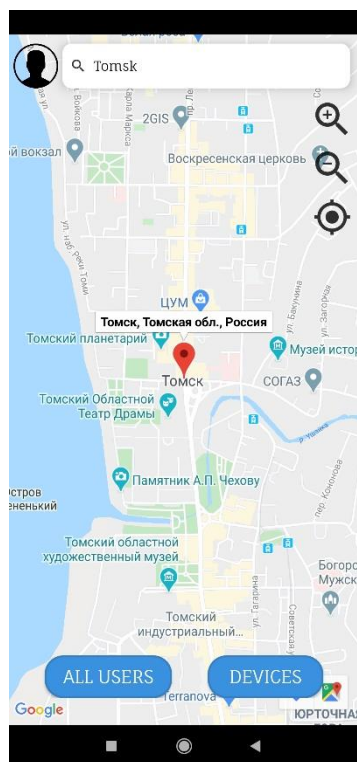


Рисунок 16 – Поиск мест на карте

В верхней части основной страницы находится панель поиска мест. Поиск можно осуществлять как на русском, так и на английском языках.

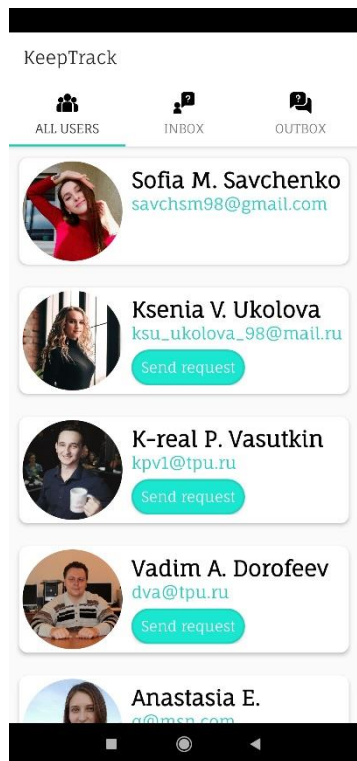


Рисунок 17 – Отображение всех пользователей

При нажатии на кнопку “ALL USERS” в левом нижнем углу открывается новая страница, содержащая три вкладки. Первая отображает всех пользователей, зарегистрированных в системе, с возможностью отправки им заявок на отслеживание.

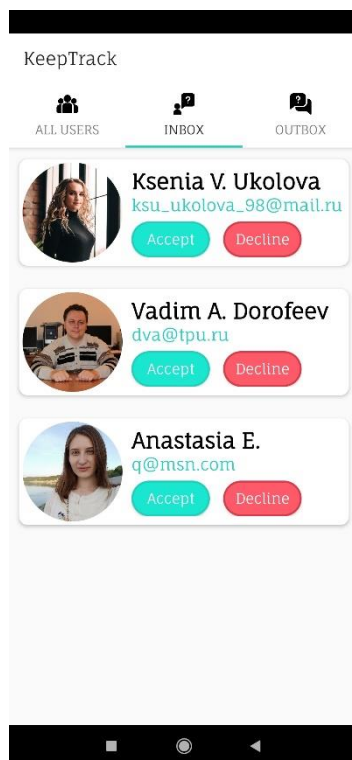


Рисунок 18 – Отображение входящих заявок на отслеживание

Вторая вкладка отображает входящие заявки на отслеживание. Каждую заявку можно принять или отклонить посредством нажатия на соответствующие кнопки.

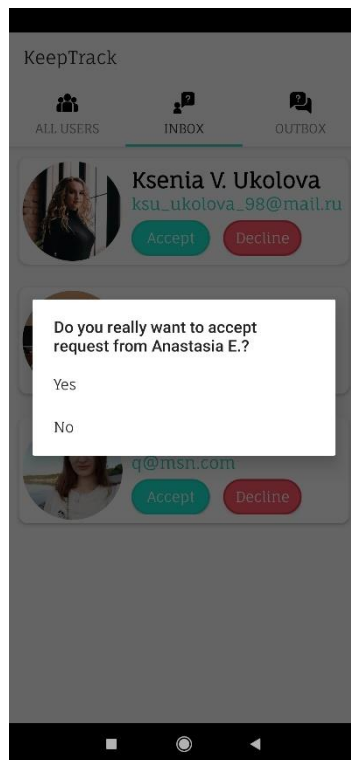


Рисунок 19 – Окно подтверждения входящей заявки на отслеживание

После нажатия на кнопку принятия заявки появляется окно, уточняющее намерения пользователя.

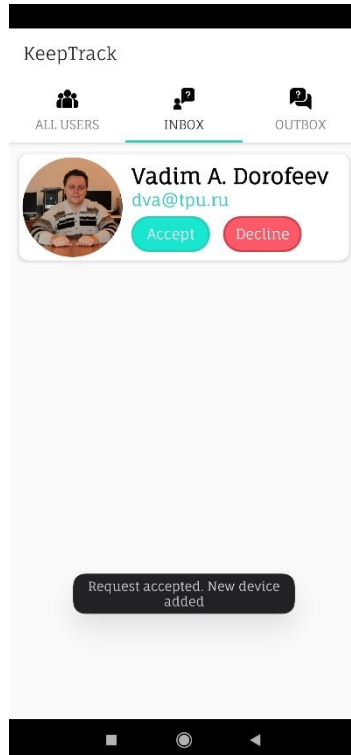


Рисунок 20 – Сообщение о принятии входящей заявки

После подтверждения принятия входящей заявки на отслеживание, карточка пользователя, отправившего запрос, перемещается на другую вкладку (Рисунок 26).

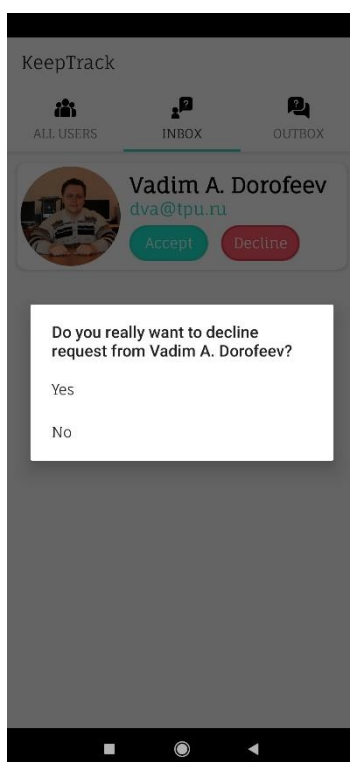


Рисунок 21 – Окно отклонения входящего запроса на отслеживание

После нажатия на кнопку отклонения заявки появляется окно, уточняющее намерения пользователя.

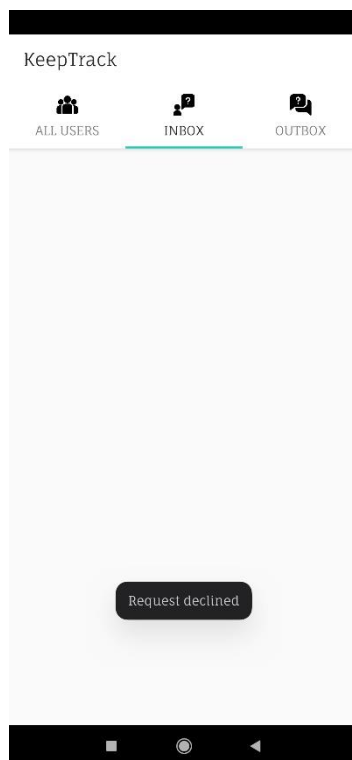


Рисунок 22 – Сообщение об успешном отклонении входящей заявки

В случае, если пользователь подтвердил отклонение входящей заявки, соответствующая карточка исчезает и появляется сообщение о том, что заявка отклонена.

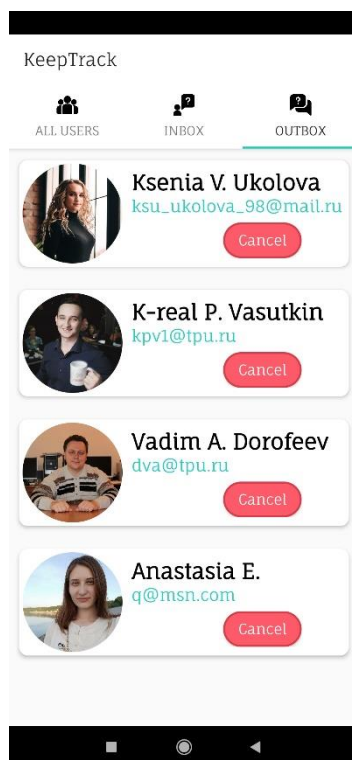


Рисунок 23 – Отображение исходящих заявок на отслеживание

Третья вкладка отображает исходящие заявки пользователя на отслеживание. Каждая карточка содержит кнопку отмены заявки.

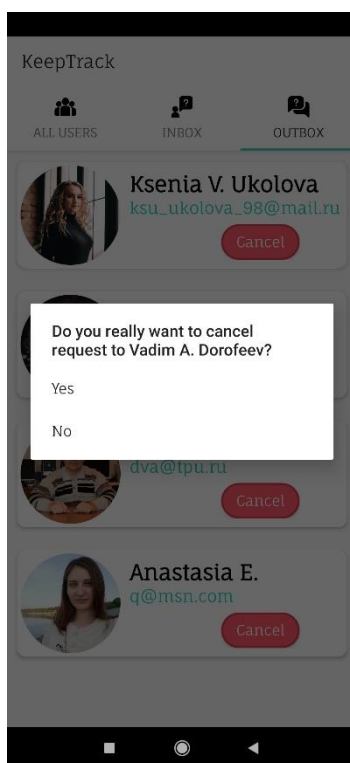


Рисунок 24 – Окно отмены исходящего запроса на отслеживание

После нажатия на кнопку отмены заявки появляется окно, уточняющее намерения пользователя.

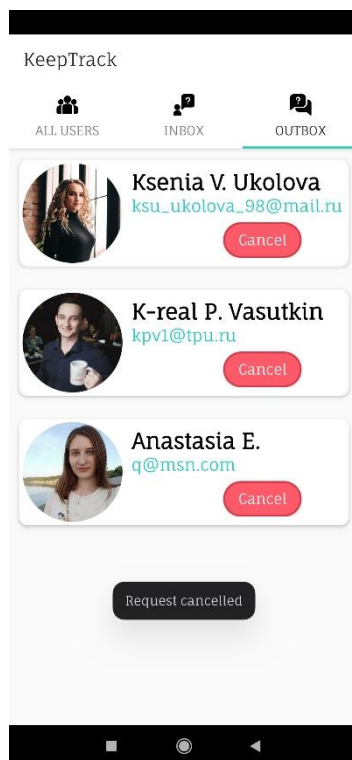


Рисунок 25 – Сообщение об успешной отмене исходящей заявки

В случае, если пользователь подтвердил отмену исходящей заявки, соответствующая карточка исчезает и появляется сообщение о том, что заявка отменена.

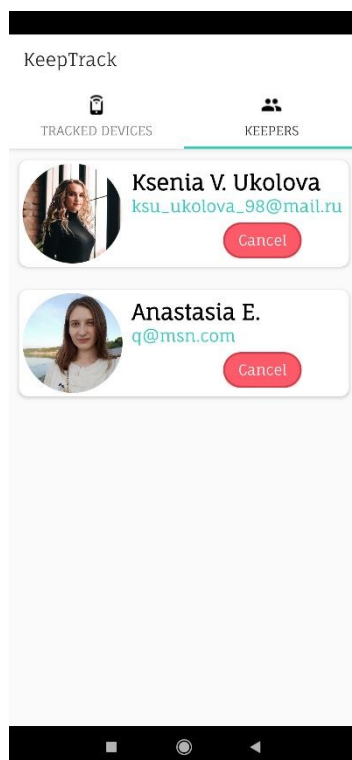


Рисунок 26 – Отображение пользователей, которым одобрена заявка на отслеживание

При нажатии на кнопку “DEVICES” основной страницы (Рисунок 15) открывается новая, содержащая две вкладки.

Вкладка “KEERERS” отображает всех пользователей, кому текущий пользователь разрешил себя отслеживать.

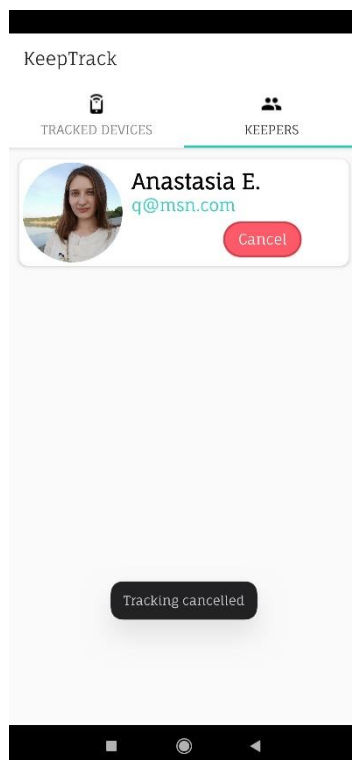


Рисунок 27 – Сообщение об отмене данного разрешения на отслеживание

Для отмены данного разрешения необходимо нажать кнопку “CANCEL”, и выбранный пользователь больше не сможет отслеживать местоположение текущего. Также появится сообщение об этом.



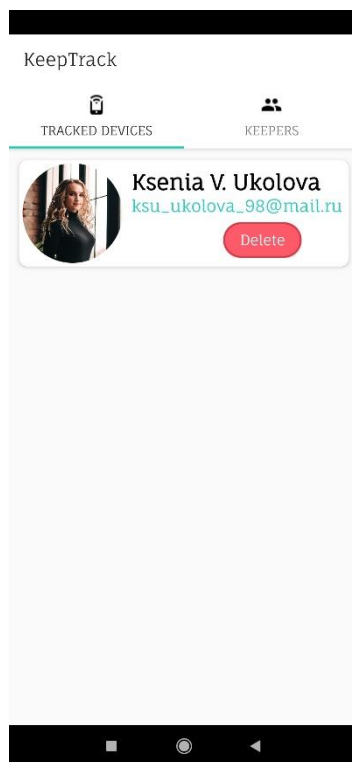


Рисунок 28 – Список отслеживаемых устройств

На вкладке “TRACKED DEVICES” отображаются пользователи, которые разрешили текущему их отслеживать. Удалить можно нажатием на кнопку.

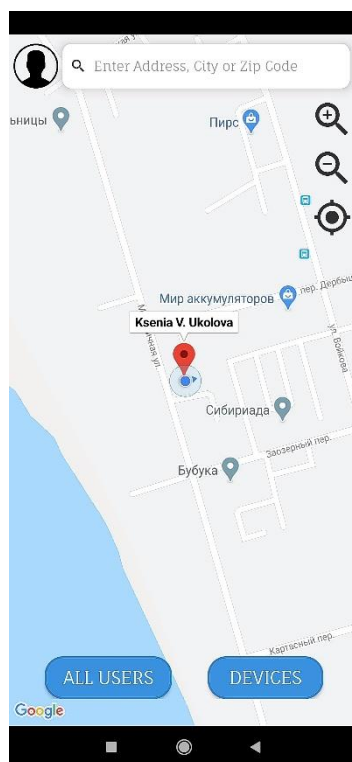


Рисунок 29 – Отображение отслеживаемых пользователей на карте

После того, как текущий пользователь получил разрешение других пользователей на отслеживание, на карте появляется маркер в том месте, где находится пользователь. При нажатии на маркер можно увидеть имя отслеживаемого пользователя.

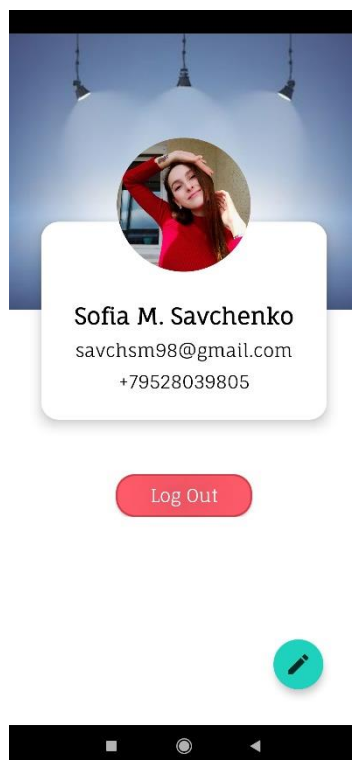


Рисунок 30 – Профиль пользователя

При нажатии на иконку в левом верхнем углу основной страницы открывается страница редактирования профиля текущего пользователя, отображающая фоновую картинку, фото, имя, почту и номер телефона пользователя. Также имеется кнопка выхода в центре страницы и круглая кнопка редактирования профиля в правом нижнем углу.

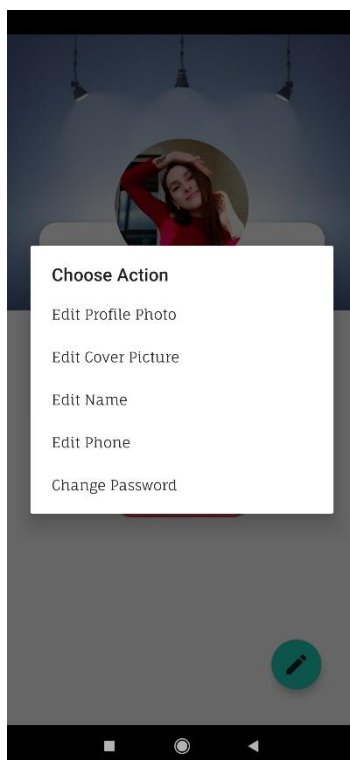


Рисунок 31 – Меню редактирования профиля

При нажатии на круглую кнопку открывается меню редактирования профиля. Здесь пользователь может изменить фоновую картинку, фото профиля, имя, номер телефона или пароль.

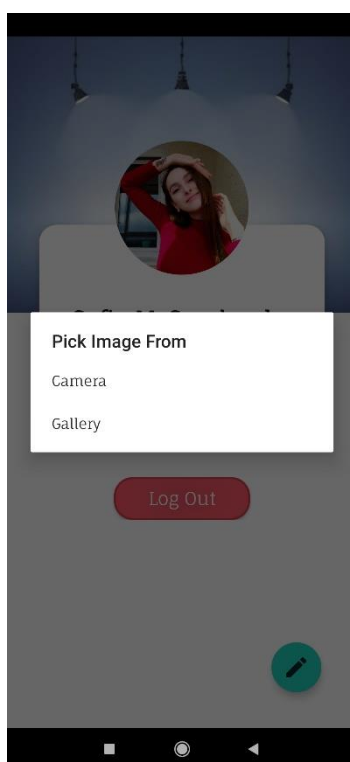


Рисунок 32 – Редактирование фонового изображения или фото пользователя

Загрузка изображения осуществляется с использованием камеры или галереи.

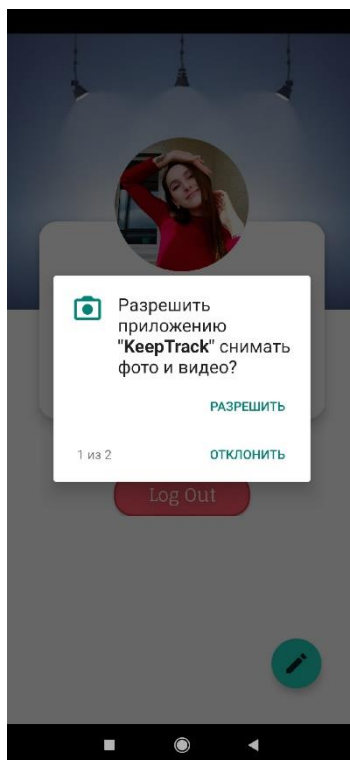


Рисунок 33 – Запрос разрешения на доступ к камере

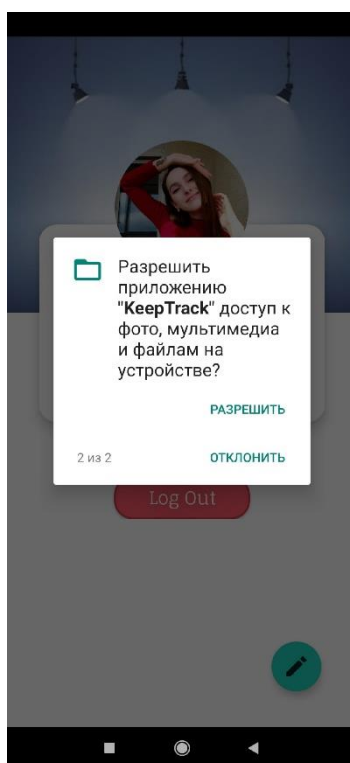


Рисунок 34 – Запрос разрешения на доступ к галерее

При попытке изменить изображение появляется окно, запрашивающее разрешение приложению снимать фото и видео (Рисунок 33), а также разрешение на доступ к галерее (Рисунок 34).

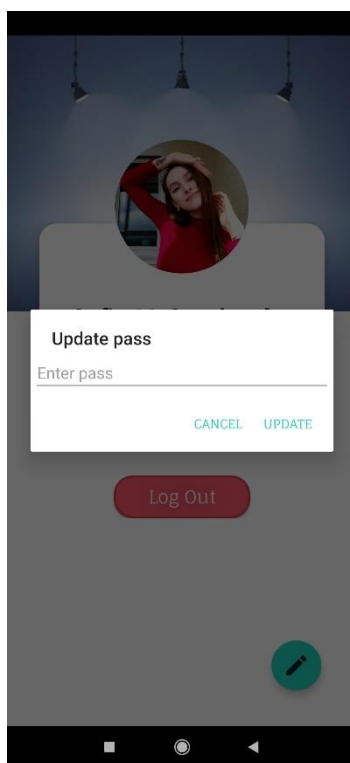


Рисунок 35 – Окно изменения пароля

Для изменения пароля необходимо ввести новую комбинацию. Аналогичным образом осуществляется редактирование имени пользователя и номера телефона.

При выходе из приложения посредством нажатия на кнопку “Log out” выполняется переход на стартовую страницу (Рисунок 8).

## **Выводы**

В ходе выполнения основной части бакалаврской работы были изучены принципы разработки мобильных приложений. Улучшены навыки работы со средой Android Studio и языком программирования Java. Приобретён опыт работы с Google Firebase.

Выполнены все требования к модулю разработки мобильного приложения.

## 5 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

### Введение

В данном разделе производится комплексное описание и анализ финансово-экономических аспектов выполненной работы. Оцениваются полные денежные затраты на проект и дается приближенная экономическая оценка результатов его внедрения.

### 5.1 Организация и планирование работ

В данном пункте составляется полный перечень проводимых работ, определяются их исполнители и рациональная продолжительность. В качестве наглядного результата выступает линейный график реализации проекта. Данные для построения графика приведены в Таблица 1.

Таблица 1. Перечень работ и продолжительность их выполнения

Этапы работы	Исполнители	Загрузка исполнителей
Постановка целей и задач, получение исходных данных	НР	НР – 100%
Разработка и утверждение задания (ТЗ)	НР, И	НР – 100% И – 10%
Подбор и изучение материалов по тематике	НР, И	НР – 30% И – 100%
Разработка календарного плана	НР, И	НР – 100% И – 10%
Проектирование архитектуры приложения	НР, И	НР – 10% И – 100%
Разработка клиентской части приложения	И	И – 100%
Тестирование готового приложения	И	И – 100%
Согласование выполненной работы с научным руководителем	НР, И	НР – 100% И – 100%
Оформление расчетно-пояснительной записки	И	И – 100%
Оформление графического материала	И	И – 100%
Подведение итогов	НР, И	НР – 60% И – 100%

### 5.1.1 Продолжительность этапов работ

Расчет продолжительности этапов работ осуществляется экспертным способом опытно-статистического метода.

Для определения вероятных значений продолжительности работ  $t_{ож}$  применяется следующая формула:

$$t_{ож\ i} = \frac{3t_{min\ i} + 2t_{max\ i}}{5} \quad (1)$$

где:

- $t_{ож\ i}$  – ожидаемая трудоемкость выполнения  $i$ -ой работы чел.-дн.;
- $t_{min\ i}$  – минимально возможная трудоемкость выполнения заданной  $i$ -ой работы (оптимистическая оценка: в предположении наиболее благоприятного стечения обстоятельств), чел.-дн.;
- $t_{max\ i}$  – максимально возможная трудоемкость выполнения заданной  $i$ -ой работы (пессимистическая оценка: в предположении наиболее неблагоприятного стечения обстоятельств), чел.-дн.

Для выполнения перечисленных в таблице 1 работ требуются специалисты:

- инженер – в его роли действует исполнитель НИР (ВКР);
- научный руководитель.

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:

$$T_{РД} = \frac{t_{ож}}{K_{ВН}} \cdot K_{Д} \quad (2)$$

где:

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

- $K_{ВН}$  – коэффициент выполнения работ, учитывающий влияние внешних факторов на соблюдение предварительно определенных длительностей, в частности, возможно  $K_{ВН} = 1$ ;
- $K_{Д}$  – коэффициент, учитывающий дополнительное время на компенсацию непредвиденных задержек и согласование работ ( $K_{Д} = 1-1,2$ ; в этих границах конкретное значение принимает сам исполнитель).

Расчет продолжительности этапа в календарных днях ведется по формуле:

$$T_{КД} = T_{РД} \cdot T_{К} \quad (3)$$

где:

- $T_{КД}$  – продолжительность выполнения этапа в календарных днях;
- $T_{К}$  – коэффициент календарности, позволяющий перейти от длительности работ в рабочих днях к их аналогам в календарных днях, и рассчитываемый по формуле:

$$T_{К} = \frac{T_{КАЛ}}{T_{КАЛ} - T_{ВД} - T_{ПД}} \quad (4)$$

где:

- $T_{КАЛ}$  – календарные дни ( $T_{КАЛ} = 365$ );
- $T_{ВД}$  – выходные дни ( $T_{ВД} = 52$ );
- $T_{ПД}$  – праздничные дни ( $T_{ПД} = 10$ ).

$$T_{К} = \frac{365}{365 - 52 - 10} = 1,205 \quad (5)$$

С помощью указанных расчетных формул вычислим для каждой работы ее трудоемкость и длительность (Таблица 2). По полученным результатам построим график работ (Таблица 3).



Таблица 2. Трудозатраты на выполнение проекта

Этап	Исполнители	Продолжительность работ, дни			Трудоемкость работ по исполнителям чел.-дн.			
					ТРД		ТКД	
		t min	t max	t ож	НР	И	НР	И
1	2	3	4	5	6	7	8	9
Постановка задачи	НР	2	5	3,2	3,84	–	4,63	–
Разработка и утверждение технического задания (ТЗ)	НР, И	2	5	3,2	3,2	3,2	3,8	3,8
Подбор и изучение материалов по тематике	НР, И	12	15	13,2	1,58	15,84	1,9	19,09
Разработка календарного плана	НР, И	2	4	2,8	3,36	0,34	4,05	0,41
Проектирование архитектуры приложения	НР, И	7	14	9,8	1,18	11,76	1,42	14,17
Разработка мобильного приложения	И	20	60	36	–	43,2	–	52,06
Тестирование готового приложения	И	3	7	4,6	–	5,52	–	6,65
Согласование выполненной работы с научным руководителем	НР, И	3	10	5,8	6,96	6,96	8,39	8,39
Оформление расчетно-пояснительной записки	И	6	9	7,2	–	8,64	–	10,41
Оформление графического материала	И	5	6	5,4	–	6,48	–	7,81
Подведение итогов	НР, И	5	8	6,2	4,46	7,44	5,37	8,97
<b>Итого:</b>				<b>97,4</b>	<b>24,58</b>	<b>109,38</b>	<b>29,56</b>	<b>131,76</b>

Таблица 3. Линейный график работ (график Ганта)

Этап	НР	И	Февраль			Март			Апрель			Май			Июнь	
			10	20	30	40	50	60	70	80	90	100	110	120	130	140
1	4,63	–	■													
2	3,8	3,8	■	■												
3	1,9	19,09		■	■	■										
4	4,05	0,41				■										
5	1,42	14,17				■	■	■								
6	–	52,06						■	■	■	■	■	■			
7	–	6,65										■				
8	8,39	8,39										■	■			
9	–	10,41												■	■	
10	–	7,81													■	■
11	5,37	8,97													■	■

## 5.2 Расчет сметы затрат на выполнение проекта

В состав затрат на создание проекта включается величина всех расходов, необходимых для реализации комплекса работ, составляющих содержание данной разработки.

Расчет сметной стоимости ее выполнения производится по следующим статьям затрат:

- материалы и покупные изделия;
- заработная плата;
- социальный налог;
- расходы на электроэнергию (без освещения);
- амортизационные отчисления;
- командировочные расходы;
- оплата услуг связи;
- арендная плата за пользование имуществом;
- прочие услуги (сторонних организаций);
- прочие (накладные расходы) расходы.

### 5.2.1 Расчет затрат на материалы

В данном пункте производится расчет затрат на материальные ресурсы, которые необходимы в процессе выполнения работ над объектом проектирования. Например, специально приобретенное оборудование стоимостью до 40 000 руб. включительно.

Таблица 4. Расчет затрат на материалы

Наименование материалов	Цена за ед., руб.	Кол-во	Сумма, руб.
Бумага для принтера формата А4	190	1 уп.	150
Картридж для принтера	1550	1 шт.	1550
<b>Итого:</b>			<b>4510</b>

## 5.2.2 Расчет заработной платы

Данная статья расходов включает заработную плату научного руководителя и инженера (в его роли выступает исполнитель проекта). Оклад инженера принимается равным окладу инженера кафедры.

Среднедневная тарифная заработная плата ( $ЗП_{\text{дн-т}}$ ) рассчитывается по формуле:

$$ЗП_{\text{дн-т}} = MO/25,083 \quad (6)$$

учитывающей, что в году 301 рабочий день и, следовательно, в месяце в среднем 25,083 рабочих дня (при шестидневной рабочей неделе).

Для учета в составе зарплаты премий, дополнительной зарплаты и районной надбавки используется следующий ряд коэффициентов:  $K_{\text{ГР}} = 1,1$ ;  $K_{\text{доп.ЗП}} = 1,188$ ;  $K_{\text{р}} = 1,3$ . Таким образом, для перехода от тарифной (базовой) суммы заработка исполнителя, связанной с участием в проекте, к соответствующему полному заработку (зарплатной части сметы) необходимо первую умножить на интегральный коэффициент  $K_{\text{и}} = 1,1 * 1,188 * 1,3 = 1,699$ . Вышеуказанное значение  $K_{\text{доп.ЗП}}$  применяется при шестидневной рабочей неделе. Для пятидневной недели  $K_{\text{доп.ЗП}}$  равно 1,62.

Таблица 5. Затраты на заработную плату

Исполнитель	Оклад, руб./мес.	Среднедневная ставка, руб./раб.день	Затраты времени, раб.дни	Коэффициент	Фонд з/платы, руб.
Дорофеев В.А. (старший преподаватель)	24 960	1040	25	1,699	44 174
Савченко С.М. (инженер кафедры)	9 489	431,32	110	1,62	76 861
<b>Итого:</b>					<b>121 035</b>

## 5.2.3 Расчет затрат на социальный налог

Данный пункт направлен на расчет затрат на единый социальный налог (ЕСН), включающий в себя отчисления в пенсионный фонд, на социальное и медицинское страхование, что составляет 30 % от полной заработной платы по проекту, т.е.  $C_{\text{соц.}} = C_{\text{зп}} * 0,3$ .

В нашем случае:  $C_{\text{соц.}} = 121\,035 * 0,3 = 36\,310,5$  руб.

#### 5.2.4 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:

$$C_{\text{эл.об.}} = P_{\text{об.}} \cdot t_{\text{об.}} \cdot \text{ЦЭ} \quad (7)$$

где:

- $P_{\text{ОБ}}$  – мощность, потребляемая оборудованием, кВт;
- $\text{ЦЭ}$  – тариф на 1 кВт·час (для ТПУ  $\text{ЦЭ} = 6,59$  руб./кВт·час (с НДС));
- $t_{\text{об}}$  – время работы оборудования, час.

Время работы оборудования вычисляется на основе итоговых данных таблицы 2 для инженера ( $T_{\text{РД}}$ ) из расчета, что продолжительность рабочего дня равна 8 часов.

$$t_{\text{об}} = T_{\text{РД}} * K_t \quad (8)$$

где  $K_t \leq 1$  – коэффициент использования оборудования по времени, равный отношению времени его работы в процессе выполнения проекта к  $T_{\text{РД}}$ , определяется исполнителем самостоятельно.

В ряде случаев возможно определение  $t_{\text{об}}$  путем прямого учета, особенно при ограниченном использовании соответствующего оборудования.

Мощность, потребляемая оборудованием, определяется по формуле:

$$P_{\text{ОБ}} = P_{\text{ном.}} * K_C \quad (9)$$

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  $K_C = 1$ .

Пример расчета затраты на электроэнергию для технологических целей приведен в таблице 6.

Таблица 6. Затраты на технологическую электроэнергию

Наименование оборудования	Время работы оборудования $t_{\text{ОБ}}$ , час	Потребляемая мощность $P_{\text{ОБ}}$ , кВт	Затраты $\text{Э}_{\text{ОБ}}$ , руб.
Персональный компьютер	110*8*0,6	0,3	1 043,856
Струйный принтер	1	0,1	0,659
<b>Итого:</b>			<b>1 044,515</b>

### 5.2.5 Расчет амортизационных расходов

В статье «Амортизационные отчисления» рассчитывается амортизация используемого оборудования за время выполнения проекта, по формуле:

$$C_{\text{АМ}} = \frac{N_{\text{А}} * C_{\text{ОБ}} * t_{\text{рф}} * n}{F_{\text{Д}}} \quad (10)$$

где:

- $N_{\text{А}}$  – годовая норма амортизации единицы оборудования;
- $C_{\text{ОБ}}$  – балансовая стоимость единицы оборудования с учетом ТЗР. При невозможности получить соответствующие данные из бухгалтерии она может быть заменена действующей ценой, содержащейся в ценниках, прейскурантах и т.п.;
- $F_{\text{Д}}$  – действительный годовой фонд времени работы соответствующего оборудования, берется из специальных справочников или фактического режима его использования в текущем календарном году. При этом второй вариант позволяет получить более объективную оценку  $C_{\text{АМ}}$ ;
- $t_{\text{рф}}$  – фактическое время работы оборудования в ходе выполнения проекта, учитывается исполнителем проекта;
- $n$  – число задействованных однотипных единиц оборудования.

При использовании нескольких типов оборудования расчет по формуле делается соответствующее число раз, затем результаты суммируются.

Согласно постановлению правительства РФ «О классификации основных средств, включенных в амортизационные группы», срок амортизации для ПК составляет  $2 \div 3$  года.

Примем значение  $CA = 2,5$  года.  $N_A$  определяется как величина обратная  $CA$ , в данном случае это  $1: 2,5 = 0,4$ .

Стоимость ПК 65000 руб., время использования 110 (дней) \* 8 (часов) = 880 часов, тогда для него:

$$C_{AM}(ПК) = (0,4*65000*880*1)/2408 = 9\ 501,66 \text{ руб.}$$

Стоимость принтера 12000 руб., его  $F_D = 500$  час.;  $N_A = 0,5$ ;  $t_{pф} = 1$  час., тогда его:

$$C_{AM}(Пр) = (0,5*12000*1*1)/500 = 12 \text{ руб.}$$

Итого начислено амортизации: **9 513,66 руб.**

### **5.2.6 Расчет расходов, учитываемых непосредственно на основе платежных (расчетных) документов (кроме суточных)**

Сюда относятся:

- командировочные расходы, в т.ч. расходы по оплате суточных, транспортные расходы, компенсация стоимости жилья;
- арендная плата за пользование имуществом;
- оплата услуг связи;
- услуги сторонних организаций.

Норма оплаты суточных – **100 руб./день**.

Во время выполнения проекта командировки не осуществлялись, услуги сторонних организаций не требовались.

Итого по данному пункту:  $C_{нр} = 0$  руб.

### **5.2.7 Расчет прочих расходов**

Расходы на выполнение проекта, которые не учтены в предыдущих статьях, следует принять равными 10% от суммы всех предыдущих расходов, т.е.

$$C_{проч.} = (C_{мат} + C_{зп} + C_{соц} + C_{эл.об.} + C_{ам} + C_{нп}) \cdot 0,1$$

Тогда получаем:  $C_{проч.} = (4510 + 121\ 035 + 36\ 310,5 + 1\ 044,515 + 9\ 513,66 + 0) \cdot 0,1 = 17\ 241,37$  руб.

### 5.2.8 Расчет общей себестоимости разработки

Проведя расчет по всем статьям сметы затрат на разработку (Таблица 7), можно определить общую себестоимость проекта.

Таблица 7. Смета затрат на разработку проекта

Статья затрат	Условное обозначение	Сумма, руб.
Материалы и покупные изделия	$C_{\text{мат}}$	4510
Основная заработная плата	$C_{\text{зп}}$	121 035
Отчисления в социальные фонды	$C_{\text{соц}}$	36 310,5
Расходы на электроэнергию	$C_{\text{эл.}}$	1 044,515
Амортизационные отчисления	$C_{\text{ам}}$	9 513,66
Непосредственно учитываемые расходы	$C_{\text{нр}}$	0
Прочие расходы	$C_{\text{проч}}$	17 241,37
<b>Итого:</b>		<b>189 655,045</b>

Таким образом, затраты на разработку составили:  $C = 189\,655,045$  руб.

### 5.2.9 Расчет прибыли

Поскольку исполнитель работы не располагает данными для применения «сложных» методов, прибыль принимается в размере  $5 \div 20\%$  от полной себестоимости проекта, т.е. **37 931,009** руб. (20 %) от расходов на разработку проекта.

### 5.2.10 Расчет НДС

НДС составляет 20% от суммы затрат на разработку и прибыли.

В нашем случае:  $(189\,655,045 + 37\,931,009) * 0,2 = 45\,517,2108$  руб.

### 5.2.11 Цена разработки НИР

Цена равна сумме полной себестоимости, прибыли и НДС.

В нашем случае:  $C_{\text{НИР(КР)}} = 189\,655,045 + 37\,931,009 + 45\,517,2108 = 273\,103,265$  руб.



### **5.3 Оценка экономической эффективности проекта**

Поскольку проект изначально не был ориентирован на экономический результат, приведем только краткое систематизированное описание эффекта, ожидаемого от проекта в случае его реализации.

Созданное приложение позволяет упростить процесс контроля местоположения частично недееспособных лиц (детей; пенсионеров с потерей памяти и т.д.), а также сотрудников различных служб (логистики, сфер бизнеса и других, где важно местонахождение в определенном месте в конкретное время).

Наглядным примером может служить спецавтохозяйство, осуществляющее вывоз различных отходов. При внедрении приложения в деятельность данной организации, руководитель сможет отслеживать передвижение уборочных машин, тем самым предотвращая несвоевременный вывоз мусора и образование несанкционированных свалок.

Разработанное программное обеспечение является универсальным (т.е. подходит одновременно отдельным людям и целым организациям) и предоставляется всем желающим на безвозмездной основе (бесплатно).

В качестве экономического эффекта реализации проекта рассматривается такой вид прибыли, как экономия ресурсов. Использование приложения позволит сократить расходы фирм на ликвидацию свалок, найм сотрудников-сопроводителей. Частным лицам, в свою очередь, предоставляется возможность контролировать перемещение нуждающихся в наблюдении родственников без дополнительных затрат на услуги няни и другого персонала по уходу.

Таким образом, можно сделать вывод, что реализация проекта и выпуск созданного программного продукта являются выгодными с точки зрения финансового менеджмента и ресурсоэффективности.

## **6 Социальная ответственность**

### **Введение**

В данном разделе анализируются вопросы обеспечения безопасности при проектировании объекта исследования и при его последующей эксплуатации. В ходе данной работы осуществляется проектирование серверной части приложения для обработки маршрутов движения мобильных устройств, в связи с чем необходимо изучить аспекты работы с ПЭВМ с точки зрения влияния, оказываемого на здоровье пользователей и окружающую среду.

Пользователями являются разработчики, взаимодействующие с программным кодом, составляемыми базами данных и готовым приложением, а также люди, устанавливающие готовый программный продукт непосредственно на свои устройства – персональные компьютеры и смартфоны.

Разработка производится в г. Томск на домашнем персональном ноутбуке, а также на стационарном компьютере кибернетического центра ТПУ посредством удаленного доступа. Средняя температура воздуха в зимний период года -20 градусов, в летний +21.

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

#### **6.1.1 Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства**

Согласно СанПиН 2.2.2/2.4.1340-03 [12], реализация данного проекта ВКР относится к «группе В» (трудовой деятельности) – творческой работе в режиме диалога с ПЭВМ. Для данной категории установлены следующие нормы для проектируемой рабочей зоны:

- Время непосредственной работы с ПЭВМ не должно превышать 6 часов за смену;
- Суммарное время регламентированных перерывов при 8-часовом рабочем дне составляет 90 минут;
- Продолжительность непрерывной работы за компьютером без регламентированного перерыва не должна превышать 1 час;
- Рекомендуется делать перерывы в работе за ПК продолжительностью 10-15 минут через каждые 45-60 минут работы;
- Во время регламентированных перерывов целесообразно выполнять комплексы упражнений и осуществлять проветривание помещения и влажную уборку;

Для государственного контроля и надзора за соблюдением в организациях существующих правил и норм установлены следующие специализированные органы:

- Федеральная инспекция труда;
- Государственная экспертиза условий труда Федеральной службы по труду и занятости населения;
- Федеральная служба по надзору в сфере защиты прав потребителей и благополучия человека и др.

#### **6.1.2 Организационные мероприятия при компоновке рабочей зоны.**

Для поддержания хорошего здоровья и предотвращения физических перегрузок человека, работающего с ПЭВМ, важно правильно организовать его рабочее место. В соответствии с требованиями стандартов, технических условий и (или) методических указаний по безопасности труда, оно должно удовлетворять следующим требованиям:

- Обеспечивать возможность удобного выполнения работ;
- Учитывать физическую тяжесть работ;

- Учитывать размеры рабочей зоны и необходимость передвижения в ней работающего;
- Учитывать технологические особенности процесса выполнения работ.

Невыполнение правил расположения и компоновки рабочего места создает опасность развития у пользователя профессионального заболевания или получения им производственной травмы. Требования к рабочему месту программиста также регламентируются нормами СанПин 2.2.2/2.4.1340-03.

Действующие правила нацелены на обеспечение максимально комфортных условий для работы в положении сидя за монитором. Оптимальное положение работающего достигается регулированием высоты рабочей поверхности, высоты сидения, оборудованием пространства для размещения ног и высотой подставки для ног.

Между рабочими столами с мониторами должны соблюдаться расстояния не менее 1,2 м – между боковыми поверхностями мониторов и не менее 2,0 м – между боковыми поверхностями мониторов. В случае выполнения творческой работы, требующей значительного умственного напряжения или высокой концентрации внимания, рекомендуется изолировать рабочие места с ПЭВМ друг от друга перегородками высотой 1,5 – 2,0 м.

Оптимальное расположение клавиатуры на поверхности стола установлено расстоянием в 100-300 мм от края, обращенного к пользователю. Экран видеомонитора должен находиться от глаз пользователя на расстоянии 600 - 700 мм, на поверхностях оборудования не должно быть лишних бликов, препятствующих работе.

Конструкция рабочего стула должна позволять изменять позу в процессе работы.

Соблюдение указанных норм позволяет предупредить развитие переутомления и избежать проблем со здоровьем, что, в свою очередь, повышает эффективность и работоспособность пользователей ПЭВМ.

## 6.2 Профессиональная социальная безопасность

### 6.2.1 Анализ вредных и опасных факторов, которые может создать объект исследования и которые могут возникнуть на рабочем месте при проведении исследования

Поскольку проведение исследований и эксплуатация объекта исследования проходят на одном рабочем месте помещения с ПЭВМ, описание вредных и опасных факторов для обоих случаев целесообразно поместить в один пункт.

Вредными называют производственные факторы, воздействие которых на работающего в определенных условиях приводит к заболеванию или снижению работоспособности. При изменении уровня и времени воздействия вредные производственные факторы могут стать опасными.

Опасными считаются производственные факторы, воздействие которых на работающего в конкретных условиях может привести к травмам, а также другим внезапным резким ухудшениям здоровья.

Возможные опасные и вредные факторы для проектируемой производственной зоны представлены в (Таблица 8).

Таблица 8. Возможные опасные и вредные факторы

Факторы (ГОСТ 12.0.003-2015)	Этапы работ		Нормативные документы
	Разработка	Эксплуатация	
1. Электромагнитные поля	+	+	СанПиН 2.2.2/2.4.1340-03 “Гигиенические требования к персональным электронно-вычислительным машинам и организации работы”;
2. Электростатические поля	+	+	ССБТ ГОСТ 12.1.045–84 “Электростатические поля. Допустимые уровни на рабочих местах и требования к проведению контроля”
3. Шум	+		СН 2.2.4/2.1.8.562-96 “Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки”
4. Нарушение	+		СанПиН 2.2.4.548-96 “Гигиенические

микроклимата помещения			требования к микроклимату производственных помещений”, ССБТ ГОСТ 12.1.005-88 “Общие санитарно-гигиенические требования к воздуху рабочей зоны (с Изменением N 1)”
5. Нарушение освещенности рабочей зоны	+		СанПиН 2.2.12.1.1.1278-03 “Гигиенические требования к естественному, искусственному и совмещенному освещению жилых и общественных зданий”
6. Психофизиологические факторы	+		Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 16.12.2019), СанПиН 2.2.2/2.4.1340-03 “Гигиенические требования к персональным электронно-вычислительным машинам и организации работы”

Анализ факторов, представляющих вред для пользователя при работе с ПЭВМ:

1. Электромагнитные поля;

Источниками электромагнитного излучения в данном исследовании являются мониторы и системный блок.

При сильном излучении может нарушаться психическое здоровье человека (возможны психические расстройства), сердечно-сосудистая система.

При небольшом, но постоянном излучении происходит накапливание электромагнитных воздействий, что приводит к снижению иммунитета, частым стрессам и повышенной утомляемости.

Согласно СанПиН 2.2.2/2.4.1340-03, временные допустимые уровни ЭМП, создаваемых ПЭВМ на рабочих местах пользователей следующие (Таблица 9).

Таблица 9. Временные допустимые нормы ЭМП, создаваемых ПЭВМ

Наименование параметров		ВДУ ЭМП
Напряженность электрического поля	В диапазоне частот 5 Гц-2 кГц	25 В/м
	В диапазоне частот 2 кГц-400 кГц	2,5 В/м
Плотность магнитного потока	В диапазоне частот 5 Гц-2 кГц	250 нТл
	В диапазоне частот 2 кГц-400 кГц	25 нТл
Электростатический потенциал экрана видеомонитора		500 В

Для снижения воздействия таких видов излучения рекомендуется применение мониторов с пониженным уровнем излучения (MPR-II, TCO-92, TCO-99), а также установка защитных экранов и соблюдение режимов труда и отдыха.

## 2. Электростатические поля;

Электростатические заряды способны накапливаться на экране компьютера, клавиатуре и компьютерной мыши при включенном питании.

Сильное электростатическое поле способствует нарушению сна, потере аппетита и раздражительности.

Влияние электрического тока искрового разряда статического электричества от ноутбука мало и не может вызвать поражение человека. Однако существующие заряды увеличивают накопление пыли на поверхностях, и при длительном непрерывном взаимодействии с устройством человек подвергается риску плохого самочувствия в связи с попаданием вредных веществ в глаза и дыхательные пути.

Согласно ГОСТ 12.1.045–84 [14], предельно допустимый уровень напряженности электростатических полей устанавливается равным 60 кВ/м в течение 1 часа.

Для защиты от статического электричества применяются следующие методы: заземление оборудования, увлажнение окружающего воздуха, а также применение полов из антистатического материала.

## 3. Шум;

Работа ПЭВМ сопровождается акустическими шумами. Это обусловлено следующими деталями ПЭВМ:

- Кулер на процессоре;
- Кулеры на корпусе;
- Система охлаждения видеокарты;
- Жесткие диски;
- CD-ROM.

Различные шумы от техники являются привычными для современного человека, однако они обладают накопительным эффектом, что может негативно сказываться на здоровье. К таким последствиям относятся головокружение, оглушение, рассеянность.

В соответствии с требованиями санитарных норм СН 2.2.4/2.1.8.562-96 предельно допустимые уровни звука для разных категорий напряженности трудового процесса легкой и средней физической нагрузки представлены в Таблица 10.

Таблица 10. Предельно допустимые уровни звука и эквивалентные уровни звука на рабочих местах для трудовой деятельности разных категорий напряженности в дБА

Категория напряженности трудового процесса	Предельно допустимый уровень звука, дБА
Напряженность легкой степени	80
Напряженность средней степени	70
Напряженный труд 1 степени	60
Напряженный труд 2 степени	50

В качестве средств защиты от шумов предполагается помещение вентиляторов в защитный кожух внутри корпуса ПЭВМ, само помещение (потолки и стены) рекомендуется покрывать звукопоглощающими материалами с максимальными коэффициентами звукопоглощения в области частот 63 - 8000 Гц.

#### 4. Нарушения микроклимата в помещении;

Микроклимат рабочих помещений – метеорологические условия внутренней среды помещений, которые определяются действующими на организм человека сочетаниями температуры, влажности, скорости движения воздуха и теплового излучения; комплекс физических факторов, оказывающих влияние на теплообмен человека с окружающей средой, на тепловое состояние человека и определяющих самочувствие, работоспособность, здоровье и производительность труда.

Нормы оптимальных и допустимых показателей микроклимата при работе с ЭВМ устанавливает СанПиН 2.2.4.548-96 «Гигиенические



требования к микроклимату производственных помещений» и ССБТ ГОСТ 12.1.005-88 «Общие санитарно-гигиенические требования к воздуху рабочей зоны» [15].

Все категории работ разграничиваются на основе интенсивности энергозатрат организма в ккал/ч (Вт). Работа, производимая сидя и сопровождающаяся незначительным физическим напряжением, относится к категории Ia – работа с интенсивностью энергозатрат до 120 ккал/ч (до 139 Вт).

Обеспечение допустимого уровня микроклимата помещения достигается с помощью системы водяного центрального отопления и естественной вентиляции.

Если рабочее помещение не позволяет поддерживать рекомендуемые величины микроклимата, необходимы дополнительные меры по защите работников от возможного перегревания и охлаждения. К таким относятся, например, системы местного кондиционирования воздуха; использование индивидуальных средств защиты от повышенной или пониженной температуры; установление периодов работы и отдыха в соответствии с имеющимися условиями так, чтобы нормализовать тепловое состояние сотрудника; сокращение рабочей смены и др.

Таблица 11. Оптимальные величины показателей микроклимата на рабочих местах производственных помещений

Период года	Категория работ по уровню энергозатрат, Вт	Температура, °С	Относительная влажность, %	Скорость движения воздуха, м/с
Холодный	Ia	(22÷24)	(40÷60)	0,1
Теплый	Ia	(23÷25)	(40÷60)	0,1

Таблица 12. Допустимые величины показателей микроклимата на рабочих местах производственных помещений

Период года	Категория работ по уровню энергозатрат, Вт	Температура, °С		Относительная влажность, %	Скорость движения воздуха, м/с	
		Диапазон ниже	Диапазон выше		Для диапазона	Для диапазона

		оптимальны х величин	оптимальны х величин		температур ниже оптимальны х величин, не более	температур выше оптимальны х величин, не более
Холодный	Ia	(20,0÷21,9)	(24,1÷25,0)	(15÷75)	0,1	0,1
Теплый	Ia	(21,0÷22,9)	(25,1÷28,0)	(15÷75)	0,1	0,2

#### 5. Нарушения освещенности рабочей зоны;

К понятию освещения относятся получение, распределение и использование световой энергии для обеспечения благоприятных условий видения предметов и объектов. От него зависят настроение и самочувствие человека, а также эффективность труда.

Все установки, обеспечивающие освещение, должны соответствовать нормативным требованиям. Согласно СанПиН 2.2.1/2.1.1.1278–03, нормируемый показатель искусственного освещения в помещениях с ПЭВМ равен 300 лк.

Для ликвидации недостаточности освещенности применяется настольная лампа в качестве дополнительного источника света. Также, ноутбук, за которым производится работа, размещается вблизи окна, выходящего на солнечную сторону.

#### 6. Психофизиологические факторы.

К вредным психофизиологическим факторам можно отнести переутомление, стресс, умственное и эмоциональное перенапряжение, монотонность труда.

Утомление и снижение работоспособности неизбежны при неправильном положении в работе. Неграмотно организованное рабочее место вызывает мышечные спазмы и усталость.

Длительная работа за экраном дисплея способствует снижению зрения, головной боли, раздражительности, потери внимания. А отсутствие регламентированных перерывов способно вызывать умственное перенапряжение, что делает неэффективной всю дальнейшую работу,

Режим труда и отдыха работника: при вводе данных, редактировании программ, чтении информации с экрана непрерывная продолжительность работы не должна превышать 4-х часов при 8-часовом рабочем дне. Через каждый час работы необходимо делать перерыв на 5-10 минут, а через два часа – на 15 минут.

Для предотвращения переутомления и напряжения необходимо строго соблюдать регламентируемые перерывы, проводя в это время физическую разминку, то же рекомендуется делать и по окончании рабочего дня.

### **6.2.2 Обоснование мероприятий по защите исследователя от действия опасных и вредных факторов**

Воздействие перечисленных выше факторов уменьшает работоспособность человека и, накапливаясь, может приводить к профессиональному заболеванию.

В связи с этим необходимо соблюдать все установленные государственными стандартами правила по организации рабочего места, труда и отдыха работника, а именно: поддерживать уровень шума, освещенности и микроклимата в допустимых значениях, а также предоставлять работникам возможность своевременного отдыха и разгрузки.

По степени опасности поражения электрическим током помещение относится к категории помещений без повышенной опасности, так как отсутствуют высокая температура, токопроводящая пыль, особая сырость и другие условия, создающие повышенную или особую опасность.

Провода используемых в процессе разработки устройств должны быть тщательно изолированы, а при повреждении должны быть заменены на новые. Электроаппаратура должна быть защищена от воздействия керосина, масла, охлаждающих жидкостей, стружки, пыли и возможного механического повреждения. При подключении к одному источнику сразу нескольких устройств, их суммарная мощность не должна превышать допустимую.

Рабочая зона (комната) полностью соответствует всем перечисленным требованиям, дополнительные мероприятия не нужны.

## **6.3 Экологическая безопасность**

### **6.3.1 Анализ влияния объекта и процесса исследования на окружающую среду**

Работа с любыми современными техническими и промышленными средствами не проходит бесследно для окружающей среды.

Проведение исследования и дальнейшее использование объекта исследования также оказывает свое влияние. Основными источниками разрушения окружающей среды в данном случае являются следующие:

- Увеличение энергопотребления;
- Отсутствие утилизации ПЭВМ;
- Отсутствие утилизации батарей смартфонов;
- Отсутствие утилизации макулатуры.

### **6.3.2 Обоснование мероприятий по защите окружающей среды.**

Разработка исследуемого модуля предполагает постоянное использование ПЭВМ, в результате чего образуется проблема, связанная с энергопотреблением.

Высокая потребность в электроэнергии приводит к необходимости увеличения мощности и количества электростанций, которые кроме потребления многих природных ресурсов осуществляют также и выброс вредных веществ в атмосферу.

При рассмотрении влияния процесса утилизации ПЭВМ были выявлены особо вредные выбросы согласно ГОСТ Р 51768-2001. В случае выхода из строя компьютеров, они списываются и отправляются 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».

Приём электроники ведется в магазинах “Эльдорадо”, “М.Видео”, “Техносила”, а также в специальных пунктах приема организаций: “Ведущая Утилизирующая Компания”, ООО “Утилизация-Томск” и ООО “Кварцит”.

Основной вид мусора – это отходы печати, коробки от техники, использованная бумага. В Томске есть большое количество пунктов приема макулатуры (например, компания “Чистый мир”).

За время разработки ПЭВМ, оргтехника и мобильные телефоны не вышли из строя, соответственно, их утилизация не требуется.

## **6.4 Безопасность в чрезвычайных ситуациях.**

### **6.4.1 Анализ вероятных ЧС, которые может инициировать объект исследований и которые могут возникнуть на рабочем месте при проведении исследований**

Наиболее вероятной ЧС в рассматриваемом случае является пожар в здании. Источниками возгорания могут стать внутренние рабочие элементы ПЭВМ, электропроводка и взрывоопасные предметы в помещении исследователя согласно ГОСТ 12.1.044-89 «Система стандартов безопасности труда. Пожаровзрывоопасность веществ и материалов. Номенклатура показателей и методы их определения».

### **6.4.2 Обоснование мероприятий по предотвращению ЧС и разработка порядка действий в случае возникновения ЧС.**

Для предупреждения пожара используются следующие методы: наличие схем здания с планами эвакуации, обеспечение помещения необходимыми средствами пожаротушения, установка систем звукового и визуального оповещения об опасности, а также инструктаж всех работников по соблюдению требований безопасности.

В случае угрозы возникновения ЧС необходимо отключить электропитание, вызвать по телефону пожарную команду, эвакуировать людей из помещения согласно плану эвакуации. При наличии небольшого очага пламени можно воспользоваться подручными средствами с целью прекращения доступа воздуха к объекту возгорания. В качестве подручных средств можно использовать углекислотные огнетушители ОУ-5 высокого давления с зарядом жидкой двуокиси углерода.

## **Выводы**

В соответствии с приведенными правилами и нормами работы с ПЭВМ и организации рабочего места, осуществлена проверка имеющихся условий на соответствие требованиям и сделал вывод, что исследование и дальнейшая эксплуатация разрабатываемого продукта не приведет к ухудшению здоровья работника и окружающей его среды.

## **Заключение**

В результате выполнения выпускной квалификационной работы бакалавра было разработано клиент-серверное приложение для обработки маршрутов движения мобильных устройств.

В приложении реализованы все утвержденные в техническом задании функции. Тестирование успешно проведено на мобильных устройствах Xiaomi Redmi Note 8 Pro и Redmi Note 8T.

Проведена оценка финансовой ресурсоэффективности, подтвердившая выгодность реализации проекта и выпуска созданного программного продукта.

Посредством анализа проекта с точки зрения социальной ответственности установлено, что исследование и дальнейшая эксплуатация разрабатываемого продукта не причинит вред здоровью трудящегося и окружающей его среды.

## Список использованных источников

1. Структура клиент-серверных приложений [Электронный ресурс] // synset.com URL: <http://fixapp.ru/programmirovanie-front-back-end/> (дата обращения: 05.02.2020).
2. Преимущества и недостатки клиент-серверной архитектуры [Электронный ресурс] // zametkinapolyah.ru URL: <https://zametkinapolyah.ru/servera-i-protokoly/o-modeli-vzaimodejstviya-klient-server-prostymi-slovami-arxitektura-klient-server-s-primerami.html> (дата обращения: 05.02.2020).
3. Определение местоположения человека по GPS/LBS [Электронный ресурс] // gdemoi.ru URL: <https://www.gdemoi.ru/res/gps-mestonahozhdenie-cheloveka/> (дата обращения: 06.02.2020).
4. Как отследить ребенка по мобильному телефону [Электронный ресурс] // КакOperator.ru URL: <https://kakoperator.ru/collect/kak-otsledit-rebenka-po-mobilnomu-telefonu> (дата обращения: 07.02.2020).
5. Где мои дети [Электронный ресурс] // watch.findmykids.org URL: [https://watch.findmykids.org/?utm\\_source=landing&utm\\_medium=seo&utm\\_campaign=ru%2F&utm\\_content=fast\\_delivery\\_B](https://watch.findmykids.org/?utm_source=landing&utm_medium=seo&utm_campaign=ru%2F&utm_content=fast_delivery_B) (дата обращения: 07.02.2020).
6. Zoemob [Электронный ресурс] // zoemob.com URL: <https://www.zoemob.com/> (дата обращения: 07.02.2020).
7. Sigyc Family [Электронный ресурс] // family-locator.com URL: <https://family-locator.com/> (дата обращения: 07.02.2020).
8. Documentation [Электронный ресурс] // developer.android.com URL: <https://developer.android.com/docs> (дата обращения: 12.02.2020).
9. Java Documentation [Электронный ресурс] // docs.oracle.com URL: <https://docs.oracle.com/en/java/> (дата обращения: 20.02.2020).
10. Firebase Documentation [Электронный ресурс] // Firebase URL: <https://firebase.google.com/docs> (дата обращения: 01.03.2020).



11. "Трудовой кодекс Российской Федерации" от 30.12.2001 N 197-ФЗ (ред. от 16.12.2019) [Электронный ресурс] // НПА Нормативно-правовые акты URL: <https://baza.npa.ru/tk/> (дата обращения: 02.03.2020).

12. СанПиН 2.2.2/2.4.1340-03 Гигиенические требования к персональным электронно-вычислительным машинам и организации работы [Электронный ресурс] // Электронный фонд правовой и нормативно-технической документации URL: <http://docs.cntd.ru/document/901865498> (дата обращения: 04.03.2020).

13. ГОСТ 12.2.032-78 Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя. Общие эргономические требования [Электронный ресурс] // Электронный фонд правовой и нормативно-технической документации URL: <http://docs.cntd.ru/document/1200003913> (дата обращения: 07.03.2020).

14. ССБТ ГОСТ 12.1.045–84 “Электростатические поля. Допустимые уровни на рабочих местах и требования к проведению контроля” [Электронный ресурс] // Электронный фонд правовой и нормативно-технической документации URL: <http://docs.cntd.ru/document/9051575> (дата обращения: 08.03.2020).

15. ГОСТ 12.0.003-2015 Система стандартов безопасности труда (ССБТ). Опасные и вредные производственные факторы. Классификация [Электронный ресурс] // Электронный фонд правовой и нормативно-технической документации URL: <http://docs.cntd.ru/document/1200136071> (дата обращения: 07.03.2020).

16. СН 2.2.4/2.1.8.562-96 “Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки. Санитарные нормы” [Электронный ресурс] // Электронный фонд правовой и нормативно-технической документации URL: <http://docs.cntd.ru/document/901703278> (дата обращения: 08.03.2020).

# Приложение А

## MainActivity.java

```
package com.example.keeptrack;

import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.LayoutInflater;
import android.view.View;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.RelativeLayout;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

import com.example.keeptrack.Models.User;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.material.snackbar.Snackbar;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.rengwuxian.materialedittext.MaterialEditText;

public class MainActivity extends AppCompatActivity {

    Button btnSignIn, btnRegister;
    FirebaseAuth auth;//для авторизации
    FirebaseDatabase db;//для подключения к БД
    DatabaseReference users;//для работы с таблицами БД

    RelativeLayout root;

    final int ALPHA = 1;

    TextView tv;
    TextView tv1;
    TextView text_bottom;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        btnSignIn = findViewById(R.id.btnSignIn);
        btnRegister = findViewById(R.id.btnRegister);

        auth = FirebaseAuth.getInstance();
        db = FirebaseDatabase.getInstance();
        users = db.getReference("Users");
```

```

root = findViewById(R.id.root_element);

btnRegister.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        showRegisterWindow();
    }
});

btnSignIn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        showSignInWindow();
    }
});

tv = (TextView) findViewById(R.id.tv);
tv1 = (TextView) findViewById(R.id.tv1);
text_bottom = (TextView) findViewById(R.id.text_bottom);
btnSignIn = (Button) findViewById(R.id.btnSignIn);
btnRegister = (Button) findViewById(R.id.btnRegister);

Animation anim = null;
anim = AnimationUtils.loadAnimation(this, R.anim.alpha_anim);

tv.startAnimation(anim);
tv1.startAnimation(anim);
text_bottom.startAnimation(anim);
btnSignIn.startAnimation(anim);
btnRegister.startAnimation(anim);
}

private void showSignInWindow(){
    AlertDialog.Builder dialog = new AlertDialog.Builder(this);
    dialog.setTitle("Log In");
    dialog.setMessage("Please, fill in all the fields below");

    LayoutInflater inflater = LayoutInflater.from(this);
    View sign_in_window = inflater.inflate(R.layout.sign_in_window,
null);
    dialog.setView(sign_in_window);

    final MaterialEditText email =
sign_in_window.findViewById(R.id.emailField);
    final MaterialEditText pass =
sign_in_window.findViewById(R.id.passField);

    dialog.setNegativeButton("Cancel", new
DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int which) {
            dialogInterface.dismiss();
        }
    });

    //Проверка введенных параметров
    dialog.setPositiveButton("Log in", new
DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int which) {
            if (TextUtils.isEmpty(email.getText().toString())){
                Snackbar.make(root, "Enter your mail",

```

```

Snackbar.LENGTH_SHORT).show();
        return;
    }
    if (pass.getText().toString().length() < 6) {
        Snackbar.make(root, "Password must contain 6 or more
characters", Snackbar.LENGTH_SHORT).show();
        return;
    }
    auth.signInWithEmailAndPassword(email.getText().toString(),
pass.getText().toString())
        .addOnSuccessListener(new
OnSuccessListener<AuthResult>() {
            @Override
            public void onSuccess(AuthResult authResult) {
                startActivity(new Intent(MainActivity.this,
MapActivity.class));
                finish();
            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                Snackbar.make(root, "Authorisation Error." +
e.getMessage(), Snackbar.LENGTH_SHORT).show();
            }
        });
    });
    dialog.show();
}

private void showRegisterWindow() {
    AlertDialog.Builder dialog = new AlertDialog.Builder(this);
    dialog.setTitle("Register");
    dialog.setMessage("Please, fill in all the fields below");

    LayoutInflater inflater = LayoutInflater.from(this);
    View register_window = inflater.inflate(R.layout.register_window,
null);
    dialog.setView(register_window);

    final MaterialEditText email =
register_window.findViewById(R.id.emailField);
    final MaterialEditText pass =
register_window.findViewById(R.id.passField);
    final MaterialEditText name =
register_window.findViewById(R.id.nameField);
    final MaterialEditText phone =
register_window.findViewById(R.id.phoneField);

    dialog.setNegativeButton("Cancel", new
DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int which) {
            dialogInterface.dismiss();
        }
    });

    dialog.setPositiveButton("Register", new
DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int which) {
            if (TextUtils.isEmpty(email.getText().toString())) {

```

```

        Snackbar.make(root, "Enter your email",
Snackbar.LENGTH_SHORT).show();
        return;
    }
    if (pass.getText().toString().length() < 6) {
        Snackbar.make(root, "Password must contain 6 or more
characters", Snackbar.LENGTH_SHORT).show();
        return;
    }
    if (TextUtils.isEmpty(name.getText().toString())) {
        Snackbar.make(root, "Enter your name",
Snackbar.LENGTH_SHORT).show();
        return;
    }
    if (TextUtils.isEmpty(phone.getText().toString())) {
        Snackbar.make(root, "Enter your phone number",
Snackbar.LENGTH_SHORT).show();
        return;
    }

    auth.createUserWithEmailAndPassword(email.getText().toString(),
pass.getText().toString())
        .addOnSuccessListener(new
OnSuccessListener<AuthResult>() {
            @Override
            public void onSuccess(AuthResult authResult) {
                final User user = new User();
                user.setEmail(email.getText().toString());
                user.setPass(pass.getText().toString());
                user.setName(name.getText().toString());
                user.setPhone(phone.getText().toString());
                user.setLatitude("56.4653819");
                user.setLongitude("84.95016091256596");

                FirebaseUser currentUser =
FirebaseAuth.getInstance().getCurrentUser();
                user.setUid(currentUser.getUid());

                users.child(FirebaseAuth.getInstance().getCurrentUser().getUid())
                    .setValue(user)
                    .addOnSuccessListener(new
OnSuccessListener<Void>() {
                        @Override
                        public void onSuccess(Void aVoid)
{
                            Snackbar.make(root, "You are
registered!", Snackbar.LENGTH_SHORT).show();

                            user.setUid(FirebaseAuth.getInstance().getCurrentUser().getUid());
                        }
                    });
            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                Snackbar.make(root, "Registration Error." +
e.getMessage(), Snackbar.LENGTH_SHORT).show();
            }
        });
    }
});
dialog.show();

```

```
}  
}
```

## AdapterUsers.java

```
package com.example.keeptrack;  
  
import android.content.Context;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.Button;  
import android.widget.ImageView;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.recyclerview.widget.RecyclerView;  
  
import com.example.keeptrack.Models.User;  
import com.google.android.gms.tasks.OnCompleteListener;  
import com.google.android.gms.tasks.Task;  
import com.google.firebase.auth.FirebaseAuth;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
import com.squareup.picasso.Picasso;  
  
import java.util.List;  
  
public class AdapterUsers extends  
RecyclerView.Adapter<AdapterUsers.MyHolder>{  
  
    private String currentID =  
FirebaseAuth.getInstance().getCurrentUser().getUid();  
    String Current_State;  
  
    Context context;  
    List<User> userList;  
  
    DatabaseReference RequestsRef;  
    Button SendRequestButton;  
  
    public AdapterUsers(Context context, List<User> userList) {  
        this.context = context;  
        this.userList = userList;  
    }  
  
    @NonNull  
    @Override  
    public MyHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int  
viewType) {  
  
        View view = LayoutInflater.from(context).inflate(R.layout.row_user,  
viewGroup, false);  
        SendRequestButton = view.findViewById(R.id.btn_request_accept);  
        Current_State = "new";  
  
        return new MyHolder(view);  
    }  
  
    @Override  
    public void onBindViewHolder(@NonNull MyHolder myHolder, final int  
position) {
```

```

final String receiverID = userList.get(position).getUId();
String userIc = userList.get(position).getImage();
final String userName = userList.get(position).getName();
final String userEmail = userList.get(position).getEmail();

myHolder.mUserName.setText(userName);
myHolder.mUserEmail.setText(userEmail);

try{

    Picasso.get().load(userIc)
        .placeholder(R.drawable.ic_prof)
        .into(myHolder.mUserIc);

}catch (Exception e){

}

if (!(currentID.equals(receiverID))){

    SendRequestButton.setVisibility(View.VISIBLE);
    SendRequestButton.setMinimumWidth(350);
    SendRequestButton.setText("Send request");

    SendRequestButton.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View v) {

            if (Current_State.equals("new")) {

                RequestsRef =
                FirebaseDatabase.getInstance().getReference().child("Requests");

                RequestsRef.child(currentID).child(receiverID)
                    .child("request_type").setValue("sent")
                    .addOnCompleteListener(new
                OnCompleteListener<Void>() {

                    @Override
                    public void onComplete(@NonNull Task<Void>
                task) {

                            if (task.isSuccessful()) {

                                RequestsRef.child(receiverID).child(currentID)
                                    .child("request_type").setValue("received")
                                        .addOnCompleteListener(new
                OnCompleteListener<Void>() {

                    @Override
                    public void
                onComplete(@NonNull Task<Void> task) {

                            if
                (task.isSuccessful()) {

                                Current_State =
                "request_sent";

                                Toast.makeText(context, "Your request to " + userName + " is successfull!",
                Toast.LENGTH_SHORT).show();

                            }

                    }

                }
            }
        }
    }
}

```

```

        });
    }
}

});
}

else {
    SendRequestButton.setVisibility(View.INVISIBLE);
}

}

@Override
public int getItemCount() {
    return userList.size();
}

class MyHolder extends RecyclerView.ViewHolder{

    ImageView mUserIc;
    TextView mUserName, mUserEmail;

    public MyHolder(@NonNull View itemView) {
        super(itemView);

        mUserIc = itemView.findViewById(R.id.user_ic);
        mUserName = itemView.findViewById(R.id.user_name);
        mUserEmail = itemView.findViewById(R.id.user_email);
    }
}
}

```

## **InboxFragment.java**

```

package com.example.keeptrack;

import android.app.AlertDialog;
import android.content.DialogInterface;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.example.keeptrack.Models.User;
import com.firebase.ui.database.FirebaseRecyclerAdapter;
import com.firebase.ui.database.FirebaseRecyclerOptions;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;

```



```

import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import com.squareup.picasso.Picasso;

import de.hdodenhof.circleimageview.CircleImageView;

public class InboxFragment extends Fragment {

    private View RequestsFragmentView;
    private RecyclerView myRequestsList;

    private DatabaseReference RequestsRef, UserRef, DevicesRef;
    private FirebaseAuth mAuth;
    private String currentUserID;

    public InboxFragment() {
        // Required empty public constructor
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {

        RequestsFragmentView = inflater.inflate(R.layout.fragment_inbox,
            container, false);

        mAuth = FirebaseAuth.getInstance();
        currentUserID = mAuth.getCurrentUser().getUid();

        UserRef =
        FirebaseDatabase.getInstance().getReference().child("Users");
        RequestsRef =
        FirebaseDatabase.getInstance().getReference().child("Requests");
        DevicesRef =
        FirebaseDatabase.getInstance().getReference().child("Devices");

        myRequestsList = (RecyclerView)
        RequestsFragmentView.findViewById(R.id.requests_list);
        myRequestsList.setLayoutManager(new
        LinearLayoutManager(getContext()));

        return RequestsFragmentView;
    }

    @Override
    public void onStart()
    {
        super.onStart();

        FirebaseRecyclerOptions<User> options =
            new FirebaseRecyclerOptions.Builder<User>()
                .setQuery(RequestsRef.child(currentUserID), User.class)
                .build();

        FirebaseRecyclerAdapter<User, RequestsViewHolder> adapter =
            new FirebaseRecyclerAdapter<User,
            RequestsViewHolder>(options) {

```

```

        @Override
        protected void onBindViewHolder(@NonNull final
RequestsViewHolder holder, int position, @NonNull User model) {

            final String list_user_id =
getRef(position).getKey();

            DatabaseReference getTypeRef =
getRef(position).child("request_type").getRef();

            getTypeRef.addValueEventListener(new
ValueEventListener() {
                @Override
                public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {

                    if (dataSnapshot.exists() ){

                        String type =
dataSnapshot.getValue().toString();

                        if (type.equals("received") &&
list_user_id != null) {

holder.itemView.findViewById(R.id.btn_request_accept).setVisibility(View.VISI
BLE);

holder.itemView.findViewById(R.id.btn_request_decline).setVisibility(View.VIS
IBLE);

UserRef.child(list_user_id).addValueEventListener(new ValueEventListener() {
                    @Override
                    public void onDataChange(@NonNull
DataSnapshot dataSnapshot) {

                        if
(dataSnapshot.hasChild("image")) {

                            final String
requestProfileImage = dataSnapshot.child("image").getValue().toString();

Picasso.get().load(requestProfileImage).into(holder.profileImage);

                        }

                            final String
requestUserName = dataSnapshot.child("name").getValue().toString();
                            final String
requestUserEmail = dataSnapshot.child("email").getValue().toString();

holder.userName.setText(requestUserName);

holder.userEmail.setText(requestUserEmail);

holder.AcceptBtn.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View
v) {

```

CharSequence

```

options[] = new CharSequence[]{

                                                                    "Yes",
                                                                    "No"

};

                                                                    AlertDialog.Builder
builder = new AlertDialog.Builder(getContext());
                                                                    builder.setTitle("Do
you really want to accept request from " + requestUserName + "?");

builder.setItems(options, new DialogInterface.OnClickListener() {
                                                                    @Override
                                                                    public void
onClick(DialogInterface dialogInterface, int i) {
                                                                    if(i == 0){

DevicesRef.child(list_user_id).child(currentUserID).child("Device")
.set("Saved").addOnCompleteListener(new OnCompleteListener<Void>() {
                                                                    @Override
                                                                    public void onComplete(@NonNull Task<Void> task) {

                                                                    if (task.isSuccessful()) {

DevicesRef.child(currentUserID).child(list_user_id).child("Keeper")
.set("Keep").addOnCompleteListener(new OnCompleteListener<Void>() {
                                                                    @Override
                                                                    public void onComplete(@NonNull Task<Void> task) {

                                                                    if (task.isSuccessful()) {

RequestsRef.child(currentUserID).child(list_user_id)
.removeValue()
.addOnCompleteListener(new OnCompleteListener<Void>() {
                                                                    @Override
                                                                    public void onComplete(@NonNull Task<Void> task) {

                                                                    if (task.isSuccessful()) {

RequestsRef.child(list_user_id).child(currentUserID)
.removeValue()
.addOnCompleteListener(new OnCompleteListener<Void>() {

```





```

                                                                    }if(i == 1){
dialogInterface.dismiss();
                                                                    }
                                                                    }
                                                                    });
                                                                    builder.show();
                                                                    }
                                                                    });
                                                                    }
                                                                    }
                                                                    @Override
                                                                    public void onCancelled(@NonNull
DatabaseError databaseError) {
                                                                    }
                                                                    });
                                                                    }
                                                                    }
                                                                    }
                                                                    @Override
                                                                    public void onCancelled(@NonNull DatabaseError
databaseError) {
                                                                    }
                                                                    });
                                                                    }
                                                                    @NonNull
                                                                    @Override
                                                                    public RequestsViewHolder onCreateViewHolder(@NonNull
ViewPager viewPager, int viewType) {
                                                                    View view =
LayoutInflater.from(viewGroup.getContext()).inflate(R.layout.row_user,
viewGroup, false);
                                                                    RequestsViewHolder holder = new
RequestsViewHolder(view);
                                                                    return holder;
                                                                    }
                                                                    };
                                                                    myRequestsList.setAdapter(adapter);
                                                                    adapter.startListening();
                                                                    }
                                                                    public static class RequestsViewHolder extends RecyclerView.ViewHolder{
                                                                    TextView userName, userEmail;
                                                                    CircleImageView profileImage;
                                                                    Button AcceptBtn, DeclineBtn;
                                                                    public RequestsViewHolder(@NonNull View itemView) {
                                                                    super(itemView);
                                                                    userName = itemView.findViewById(R.id.user_name);
                                                                    userEmail = itemView.findViewById(R.id.user_email);
                                                                    profileImage = itemView.findViewById(R.id.user_ic);

```

```

        AcceptBtn = itemView.findViewById(R.id.btn_request_accept);
        DeclineBtn = itemView.findViewById(R.id.btn_request_decline);
    }
}
}

```

## OutboxFragment.java

```

package com.example.keeptrack;

import android.app.AlertDialog;
import android.content.DialogInterface;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.example.keeptrack.Models.User;
import com.firebase.ui.database.FirebaseRecyclerAdapter;
import com.firebase.ui.database.FirebaseRecyclerOptions;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import com.squareup.picasso.Picasso;

import de.hdodenhof.circleimageview.CircleImageView;

public class OutboxFragment extends Fragment {

    private View RequestsFragmentView;
    private RecyclerView myRequestsList;

    private DatabaseReference RequestsRef, UserRef, DevicesRef;;
    private FirebaseAuth mAuth;
    private String currentUserID;

    public OutboxFragment() {
        // Required empty public constructor
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        RequestsFragmentView = inflater.inflate(R.layout.fragment_outbox,
            container, false);

        mAuth = FirebaseAuth.getInstance();

```

```

        currentUserID = mAuth.getCurrentUser().getUid();
        UserRef =
        FirebaseDatabase.getInstance().getReference().child("Users");
        RequestsRef =
        FirebaseDatabase.getInstance().getReference().child("Requests");
        DevicesRef =
        FirebaseDatabase.getInstance().getReference().child("Devices");

        myRequestsList = (RecyclerView)
        RequestsFragmentView.findViewById(R.id.requests_list2);
        myRequestsList.setLayoutManager(new
        LinearLayoutManager(getContext()));

        return RequestsFragmentView;
    }

    @Override
    public void onStart()
    {
        super.onStart();

        FirebaseRecyclerOptions<User> options =
            new FirebaseRecyclerOptions.Builder<User>()
                .setQuery(RequestsRef.child(currentUserID),
                User.class)
                .build();

        FirebaseRecyclerAdapter<User, RequestsViewHolder> adapter =
            new FirebaseRecyclerAdapter<User,
            RequestsViewHolder>(options) {
                @Override
                protected void onBindViewHolder(@NonNull final
                RequestsViewHolder holder, int position, @NonNull User model) {

                    holder.itemView.findViewById(R.id.btn_request_decline).setVisibility(View.VIS
                    IBLE);

                    final String list_user_id =
                    getRef(position).getKey();
                    //final String receiverID =
                    list_user_id.get(position).getUid();

                    DatabaseReference getTypeRef =
                    getRef(position).child("request_type").getRef();

                    getTypeRef.addValueEventListener(new
                    ValueEventListener() {
                        @Override
                        public void onDataChange(@NonNull DataSnapshot
                        dataSnapshot) {

                            if (dataSnapshot.exists()){

                                String type =
                                dataSnapshot.getValue().toString();

                                if (type.equals("sent") &&
                                list_user_id!=null){

```



```

UserRef.child(list_user_id).addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull
DataSnapshot dataSnapshot) {

        if
(dataSnapshot.hasChild("image")){

            //final String
requestUserName = dataSnapshot.child("name").getValue().toString();
            //final String
requestUserEmail = dataSnapshot.child("email").getValue().toString();
            final String
requestProfileImage = dataSnapshot.child("image").getValue().toString();

//holder.userName.setText(requestUserName);
//holder.userEmail.setText(requestUserEmail);
Picasso.get().load(requestProfileImage).into(holder.profileImage);
        }

            final String
requestUserName = dataSnapshot.child("name").getValue().toString();
            final String
requestUserEmail = dataSnapshot.child("email").getValue().toString();

holder.userName.setText(requestUserName);
holder.userEmail.setText(requestUserEmail);

holder.CancelRequestButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View
v) {

            CharSequence
options[] = new CharSequence[]{

                "Yes",
                "No"
            };

            AlertDialog.Builder
builder = new AlertDialog.Builder(getContext());
            builder.setTitle("Do
you really want to cancel request to " + requestUserName + "?");

builder.setItems(options, new DialogInterface.OnClickListener() {
    @Override
    public void
onClick(DialogInterface dialogInterface, int i) {

                if(i == 0){

RequestsRef.child(currentUserID).child(list_user_id)
.removeValue()

```



```

        @Override
        public void onCancelled(@NonNull DatabaseError
databaseError) {

            }
        });
    }

    @NonNull
    @Override
    public RequestsViewHolder onCreateViewHolder(@NonNull
ViewGroup viewGroup, int viewType) {

        View view =
LayoutInflater.from(viewGroup.getContext()).inflate(R.layout.row_user,
viewGroup, false);
        RequestsViewHolder holder = new
RequestsViewHolder(view);
        return holder;
    }
};

myRequestsList.setAdapter(adapter);
adapter.startListening();
}

public static class RequestsViewHolder extends RecyclerView.ViewHolder{

    TextView userName, userEmail;
    CircleImageView profileImage;
    Button CancelRequestButton;

    public RequestsViewHolder(@NonNull View itemView) {
        super(itemView);

        userName = itemView.findViewById(R.id.user_name);
        userEmail = itemView.findViewById(R.id.user_email);
        profileImage = itemView.findViewById(R.id.user_ic);

        CancelRequestButton =
itemView.findViewById(R.id.btn_request_decline);
        CancelRequestButton.setVisibility(View.VISIBLE);
        CancelRequestButton.setText("Cancel");
    }
}
}
}

```