

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

ДОПУСТИТЬ К ЗАЩИТЕ
Заведующий кафедрой ЭВМ
_____ Д.В. Топольский
«__»_____ 2023 г.

Мобильное приложение для организации питания в туристском походе

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ
ЮУрГУ – 090301.2023.045 ПЗ ВКР

Руководитель работы,
к.т.н., доцент каф. ЭВМ
_____ В.А. Парасич
«__»_____ 2023 г.

Автор работы,
студент группы КЭ-406
_____ В.А. Бобров
«__»_____ 2023 г.

Нормоконтролёр,
ст. преп. каф. ЭВМ
_____ С.В. Сяськов
«__»_____ 2023 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

УТВЕРЖДАЮ
Заведующий кафедрой ЭВМ
_____ Д.В. Топольский
«__» _____ 2023 г.

ЗАДАНИЕ

на выпускную квалификационную работу бакалавра
студенту группы КЭ-406
Боброву Владимиру Александровичу
обучающемуся по направлению
09.03.01 «Информатика и вычислительная техника»

- 1. Тема работы:** «Мобильное приложение для организации питания в туристском походе» утверждена приказом по университету от «__» _____ 2023 г. № _____
- 2. Срок сдачи студентом законченной работы:** 01 июня 2023 г.
- 3. Исходные данные к работе:**
Требования к функционалу разрабатываемого приложения:
 - Возможность составления меню
 - Составление списка продуктов на основании меню
 - Расчёт веса потребляемой в походе пищи
 - Расчёт максимальной нагрузки на туриста
 - Распределение продуктов между туристами для транспортировки
- 4. Перечень подлежащих разработке вопросов:**
Выпускная квалификационная работа должна содержать разбор следующих вопросов:

1. Обзор аналогов разрабатываемого приложения
2. Обзор средств реализации
3. Разработка программного обеспечения
4. Тестирование программного обеспечения

5. Дата выдачи задания: ___ декабря 2022 г.

Руководитель работы _____ / *В.А. Парасич* /

Студент _____ / *В.А. Бобров* /

КАЛЕНДАРНЫЙ ПЛАН

Этап	Срок сдачи	Подпись руководителя
Анализ рынка приложений и обзор аналогов	01.01.2023	
Введение в тематику разработки мобильных приложений и обзор средств реализации	15.01.2023	
Реализация системы	01.02.2023	
Тестирование и отладка	01.05.2023	
Компоновка текста работы и сдача на нормоконтроль	15.05.2023	
Подготовка презентации и доклада	24.05.2023	

Руководитель работы _____ / В.А. Парасич /

Студент _____ / В.А. Бобров /

АННОТАЦИЯ

В.А. Бобров Мобильное приложение для организации питания в туристском походе. – Челябинск: ФГАОУ ВО «ЮУрГУ (НИУ)», ВШ ЭКН; 2023, 48 с., 29 ил., библиогр. список – 6 наим.

В данной выпускной квалификационной работе рассматривается вопрос организации питания туристов во время похода, и реализуется решение в виде мобильного приложения. Процесс выполнения выпускной квалификационной работы включает в себя решение следующих вопросов: обзор аналогов, обзор средств реализации, разработка программного обеспечения, а также тестирование программного обеспечения.

В ходе обзора аналогов реализуемого приложения выявлены и описаны основные направления в разработке туристических мобильных приложений. При рассмотрении программ-аналогов, связанных с организацией питания в походе, отмечались их достоинства и недостатки. Сравнивая туристические приложения между собой, были выявлены требования к функционалу разрабатываемого приложения.

Проведя обзор средств реализации, был составлен перечень программных продуктов и разработок, наиболее подходящих для выполнения задачи выпускной квалификационной работы.

Описывая процесс реализации, был разобран основной функционал разработанного приложения: указаны входные и выходные данные, а также рассмотрен принцип работы.

При тестировании программного обеспечения были использованы современные методики, позволяющие определить неисправности на раннем этапе разработки. Также были приведены примеры функционирования наиболее сложного алгоритма в приложении.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	7
1 ОБЗОР АНАЛОГОВ РАЗРАБАТЫВАЕМОГО ПРИЛОЖЕНИЯ	8
1.1 Классификация туристических приложений	8
1.2 Обзор аналогов среди мобильных приложений	13
1.3 Обзор аналогов среди остальных приложений	16
2 ОБЗОР СРЕДСТВ РЕАЛИЗАЦИИ	21
2.1 Среда разработки.....	21
2.2 Язык программирования	22
2.3 Система управления базой данных	23
3 РАЗРАБОТКА ПРИЛОЖЕНИЯ	25
3.1 Архитектура базы данных.....	25
3.2 Описание входных и выходных данных.....	26
3.3 Описание разработанных алгоритмов	27
4 ТЕСТИРОВАНИЕ ПРИЛОЖЕНИЯ	36
4.1 Виды использованных тестирований.....	36
4.2 Результаты тестирований	36
4.3 Примеры функционирования наиболее важных алгоритмов.....	39
4.4 Примеры работы приложения	42
ЗАКЛЮЧЕНИЕ	47
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	48

ВВЕДЕНИЕ

Поход – это один из наиболее популярных видов активного отдыха. Во время путешествия человеку необходимо регулярно питаться, чтобы поддерживать жизнедеятельность, эту еду приходится постоянно переносить в рюкзаках. А преодолевая большие расстояния, вес переносимого груза ощущается гораздо большим, чем в действительности. Чтобы оптимизировать нагрузку, нужно рассчитать необходимое количество продуктов, которое будет достаточно для похода, но не будет излишним.

Вести подобные расчёты самостоятельно может быть проблематично, тем более что с развитием информационных технологий современному человеку стало доступно огромное количество программ и сервисов, позволяющих упростить решение различных вопросов. Также у абсолютного большинства людей имеются смартфоны, которые по мощности не уступают, а иногда и превосходят стационарные компьютеры. Такие мобильные устройства постоянно находятся рядом с человеком, а приложения, установленные на них, можно использовать в любое время и в любом месте. Таким образом, наиболее удобным и эффективным решением будет использование программы, разработанной для мобильных устройств, автоматизирующей задачу расчёта минимально необходимого количества продуктов в поход.

Для разработки подобного приложения, необходимо решить следующие вопросы.

- Обзор аналогов с целью определения требований к функционалу.
- Обзор средств реализации для выбора наиболее подходящих решений для разработки.
- Реализация приложения.
- Проведение тестирования разработанного приложения с целью выявления и устранения неисправностей.

1 ОБЗОР АНАЛОГОВ РАЗРАБАТЫВАЕМОГО ПРИЛОЖЕНИЯ

Мобильные приложения разрабатываются с целью облегчения деятельности людей, будь то ежедневные потребности или профессиональная задача, практически под каждый запрос есть то или иное приложение, ускоряющее процесс выполнения. Пешие походы не являются исключением. Рынок мобильных приложений обладает большим количеством разработок, направленных на улучшение пользовательского опыта в походах.

Поэтому для начала необходимо произвести обзор основных направлений в разработке туристических приложений.

1.1 Классификация туристических приложений

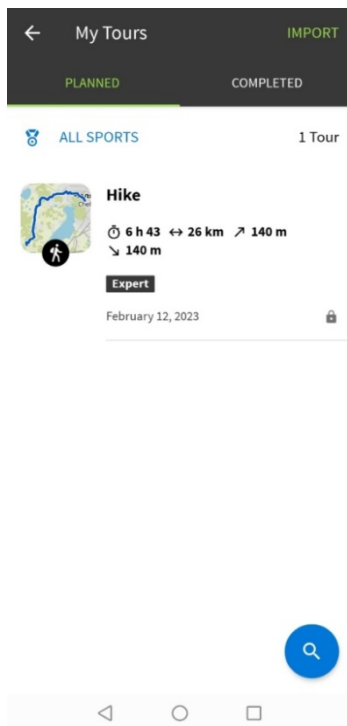
1.1.1 Карты-навигаторы

Данное направление является наиболее популярным среди разработчиков. Во время планирования похода важно правильно составить наиболее комфортный маршрут, который оставит только положительные эмоции от прогулки. Также необходимо постоянно ориентироваться непосредственно во время похода, чтобы не сбиться с пути. Раньше для этих целей использовались бумажные карты, но с развитием технологий у современного человека появилось множество программ, выполняющих подобные функции.

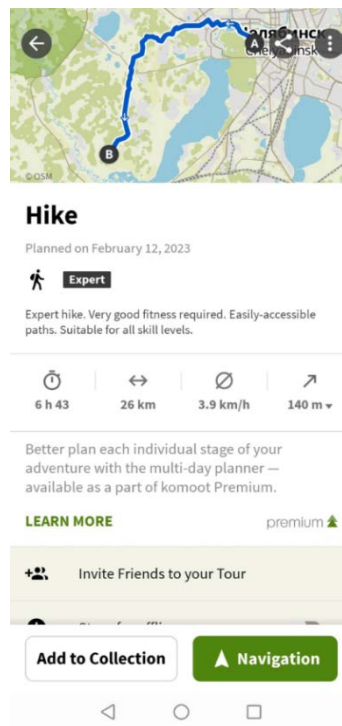
Приложения этой категории предоставляют карты под различные задачи. На них может быть информация о различных пеших или велосипедных маршрутах, наиболее подходящих для похода, удобных местах для лагеря, а также полезная информация о местности. Некоторые приложения представляют собой навигатор с возможностью определения местности через камеру мобильного устройства. Примером разработок этого направления является следующая программа:

Komoot

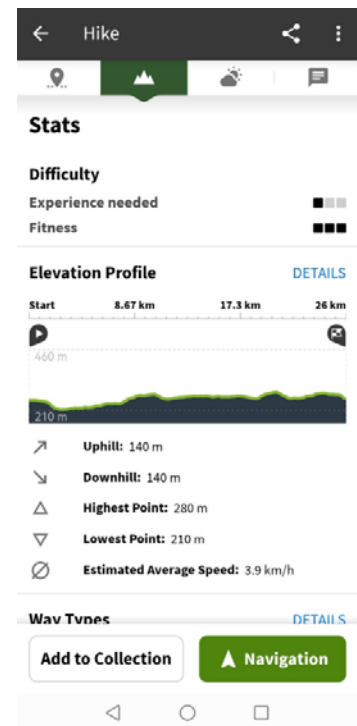
Используется для планирования похода. Приложение позволяет составлять по карте свой маршрут и приводит подробный анализ пути с указанием перепадов высот и средней продолжительностью пути. Также Komoot отмечает интересные места, которые можно посетить во время похода. Примеры работы приложения представлены на рисунках 1 и 2.



а)



б)



в)

Рисунок 1 – Планирование похода в приложении «Komoot»

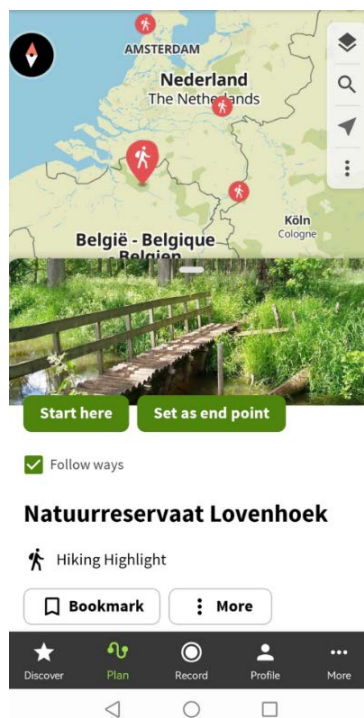


Рисунок 2 – Выбор достопримечательности в приложении «Komoot»

1.1.2 Погода

Для похода всегда необходимо знать, какие погодные условия будут ожидать во время преодоления маршрута. Поэтому такие программы очень полезны для туризма. В этом направлении стоит отметить следующее приложение:

1Weather

Приложение предоставляет множество погодных данных на несколько дней вперёд, что позволяет грамотно планировать этапы движения в походе. Пример работы представлен на рисунке 3.

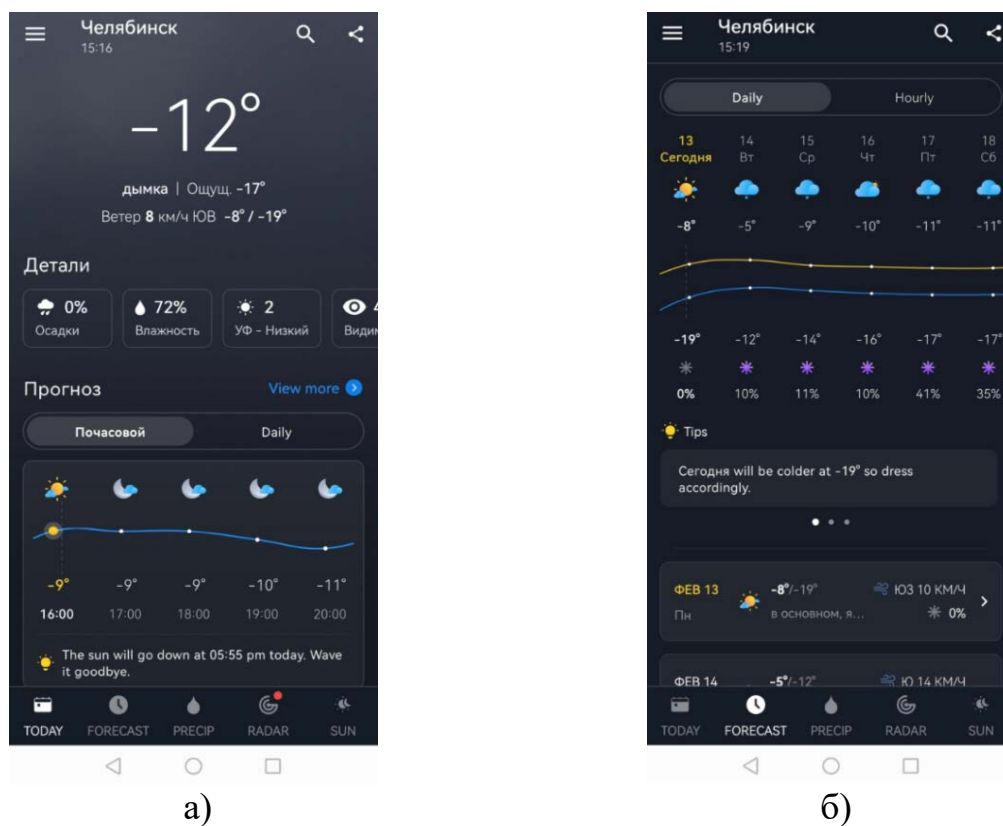


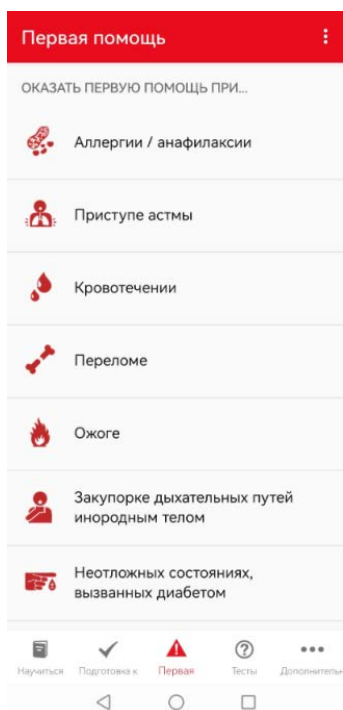
Рисунок 3 – Просмотр погоды в приложении «1Weather»

1.1.3 Туристические справочники

Вдали от цивилизации может возникнуть любая трудность. Однако учесть всё и подготовиться ко всему невозможно. Для этого и существуют карманные справочники, которыми люди пользовались раньше, а сейчас, с развитием технологий, мобильные устройства позволяют хранить в себе огромные объёмы информации, которые можно использовать под справочники на случай экстренных ситуаций. Таким образом, можно подготовиться практически к любой трудности. Наиболее полезными справочниками для туристических походов являются:

Первая помощь

В этом справочнике можно найти полную информацию по оказанию первой помощи и подготовке к чрезвычайным ситуациям. Приложение подстраивается под определённую страну и отображает телефонные номера экстренных служб, которые работают в конкретном регионе. Работа с приложением отображена на рисунках 4 и 5.

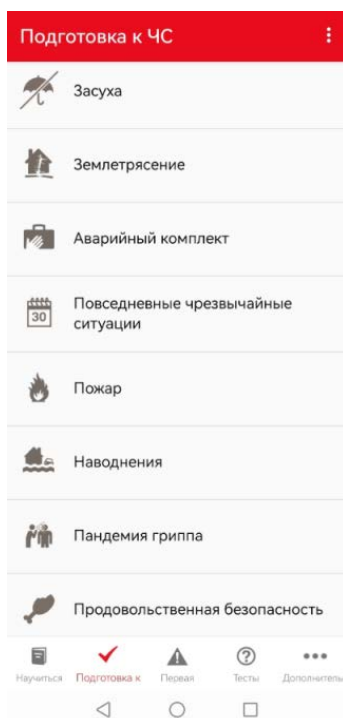


а)

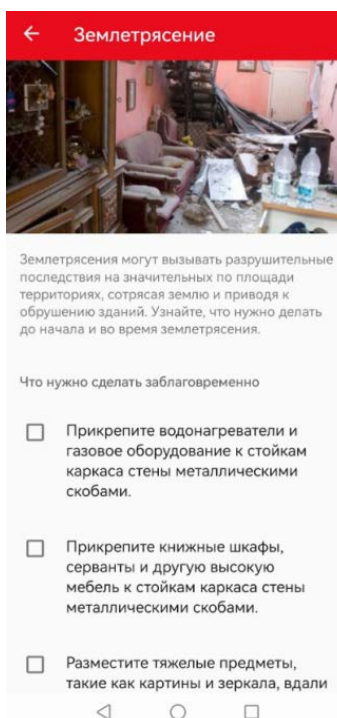


б)

Рисунок 4 – Вкладка первой помощи в приложении «Первая помощь»



а)



б)

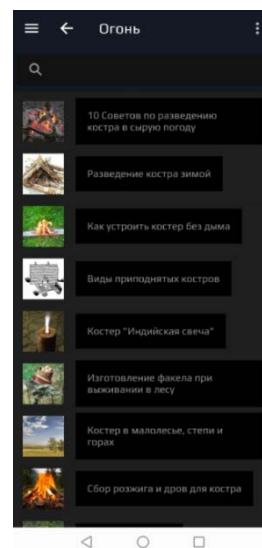
Рисунок 5 – Вкладка подготовки к ЧС в приложении «Первая помощь»

Справочник по выживанию

Это уже более узконаправленное приложение, предназначенное для обеспечения выживания в условиях отсутствия благ цивилизации. В справочнике приводятся небольшие статьи, описывающие порядок действий в непредвиденных ситуациях. Перечень статей представлен на рисунке 6.



а)



б)

Рисунок 6 – Обзор статей в приложении «Справочник по выживанию»

1.1.4 Списки вещей

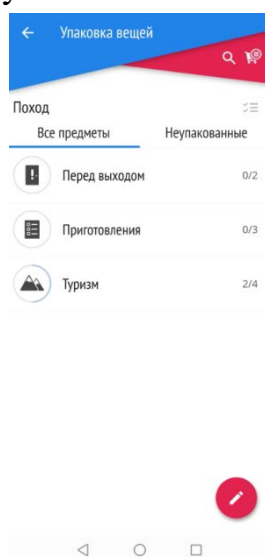
При планировании и во время похода важно помнить, какие вещи и снаряжение были взяты с собой, чтобы не забыть взять что-то важное и ничего не потерять в дороге. Под эти цели были разработаны приложения из данного направления. Они позволяют упростить организацию вещей и делают сборы в поход более комфортными. Наиболее удачным решением для составления списка вещей является следующая программа:

uPackingList

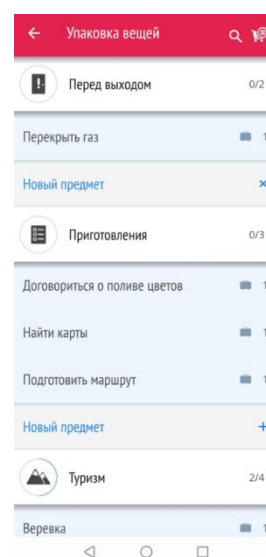
Данное приложение имеет удобный интерфейс и предоставляет большой выбор позиций от средств гигиены до задач, которые необходимо выполнить перед уходом. В случае нехватки разновидности предлагаемых предметов, пользователь может добавить собственную позицию. Пример работы приложения изображен на рисунке 7.



а)



б)



в)

Рисунок 7 – Составление списка в приложении «uPackingList»

Таким образом, произведя обзор основных направлений в разработке программ для туризма, на которые можно разделить имеющиеся приложения, приходим к следующему выводу: несмотря на всё многообразие программ, связанных с тематикой туристических походов, разработок, предназначенных для учёта продуктов и организации приёмов пищи в походе, нет, хотя данное направление не менее важно.

1.2 Обзор аналогов среди мобильных приложений

В качестве дополнительного анализа рынка мобильных приложений необходимо рассмотреть разработки, не имеющие прямого отношения к туризму, но обладающие функционалом, связанным с задачами организации работы с продуктами в походе:

Дневник еда планировщик

Приложение позволяет составлять меню для приёмов пищи на каждый день. Во время составления пользователь сам вводит названия блюд в качестве позиций меню. Пример работы с меню отображен на рисунках 8 и 9.

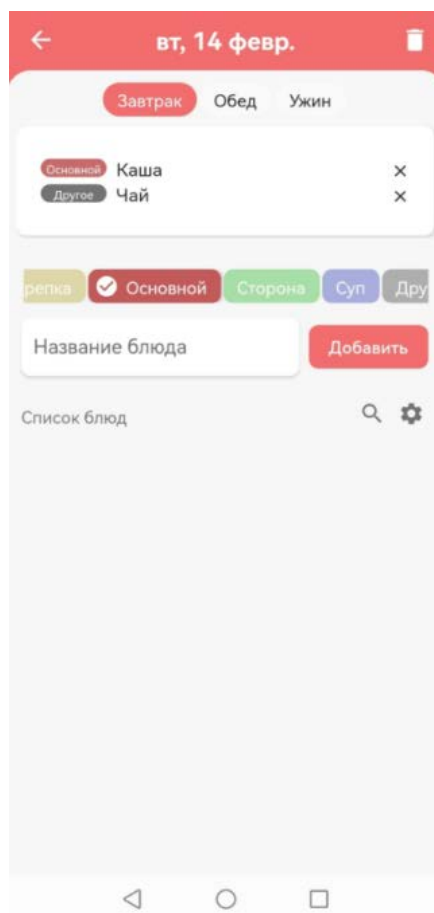


Рисунок 8 – Создание меню в приложении «Дневник еда планировщик»

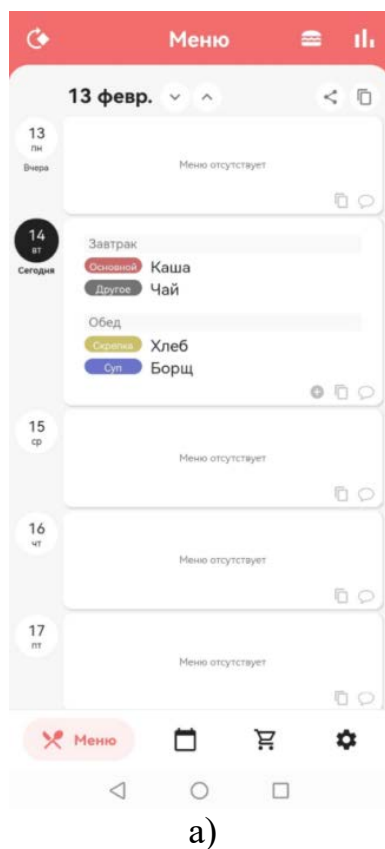


Рисунок 9 – Просмотр меню в приложении «Дневник еда планировщик»

Однако предложенных приложением возможностей недостаточно для полноценной организации питания группы туристов в походе.

Приложение не предоставляет функционал для учёта имеющихся у туристов продуктов, чтобы иметь представление об оставшейся провизии. Также при составлении меню не берётся в учёт количество участников похода и необходимый минимум калорий, который им необходимо употребить для комфортного продолжения пути.

Транспортировка продуктов также является особо важной задачей, т.к. еда – это лишняя нагрузка, которая во время постоянной ходьбы ощущается гораздо тяжелее, чем есть на самом деле. Поэтому необходимо грамотно разделить вес продуктов между участниками похода в равной степени. Такого распределения можно достичь, только если учесть особенности каждого туриста: некоторые, в силу возраста или травм, не могут поднимать тяжести. Подобную функцию данное приложение не предоставляет.

Учёт продуктов

Разработка предназначена для создания списков продуктов. Продукты добавляются пользователем самостоятельно, также указывается срок годности для отслеживания. Имеется возможность добавить собственные места хранения провизии и заполнить их добавленными продуктами. Благодаря подобным функциям данного приложения, можно составлять распределение количества припасов по рюкзакам туристов для переноса. Функционал программы представлен на рисунках 10, 11 и 12.

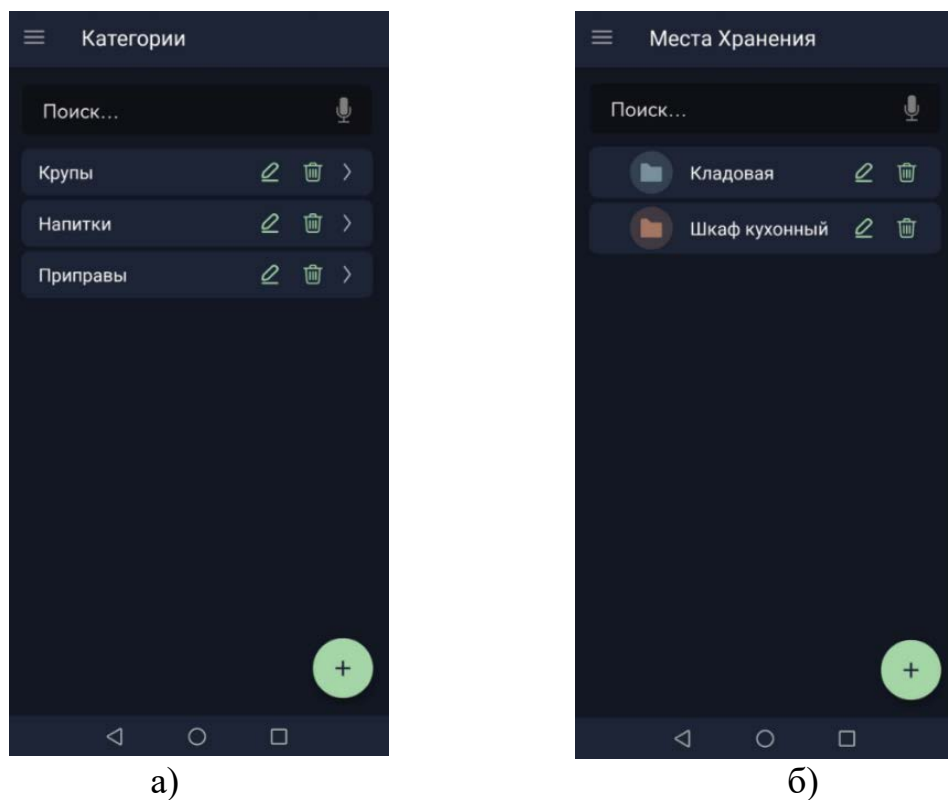


Рисунок 10 – Создание категорий продуктов и мест хранения в приложении «Учёт продуктов»

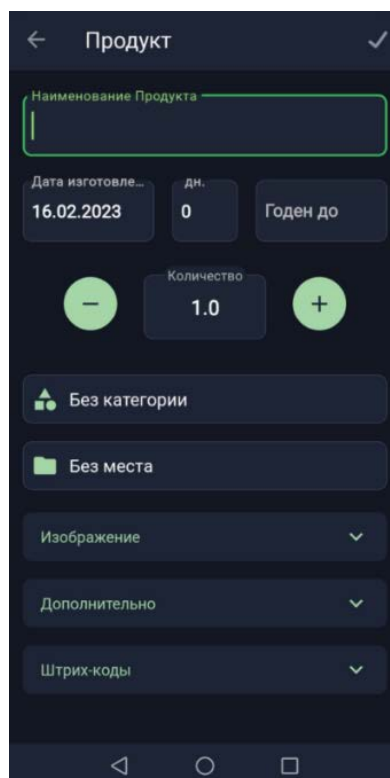


Рисунок 11 – Добавление продукта в приложении «Учёт продуктов»

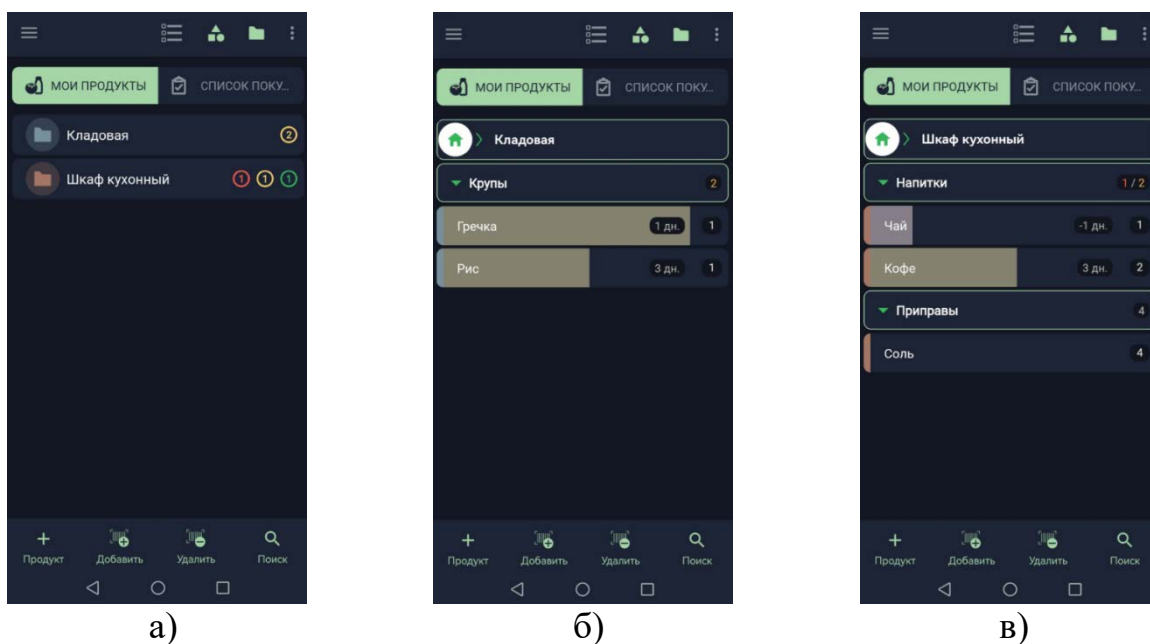


Рисунок 12 – Просмотр списка продуктов в приложении «Учёт продуктов»

Однако, также как и предыдущая разработка, это приложение не подходит для полноценного использования в походе.

В программе не указывается калорийность продуктов, из-за чего не удастся определить минимально необходимое количество еды, чтобы обеспечить всех туристов. Вдобавок распределение продуктов по рюкзакам получается неточным. Так как в программе не указывается вес продуктов, что при распределении общего количества провизии между туристами для транспортировки не позволит оценить нагрузку на человека, которая может быть за пределом его физических возможностей. Подобные расчёты калорийности и веса придётся производить отдельно, что затрудняет планирование похода.

Также, в отличие от прошлого приложения, эта программа не предоставляет функций для создания меню приёмов пищи. Хотя подобная возможность важна при работе с едой в походе, чтобы организовывать группу туристов.

Таким образом, данные разработки, хоть и позволяют производить мониторинг продуктов, но не обладают достаточным функционалом, чтобы учесть все особенности обращения с едой в условиях похода.

1.3 Обзор аналогов среди остальных приложений

Проанализировав мобильные приложения, которые имеют хоть какое-то отношение к тематике туризма или организации питания, можно прийти к выводу, что среди разработок под мобильные устройства нет программ, в полной мере удовлетворяющих требованиям организации питания в туризме.

Следующим этапом необходимо рассмотреть приложения не ориентированные под мобильные устройства:

Раскладка

Данная программа представляет собой веб-приложение, предназначенное для использования в браузере. Функционал этой разработки позволяет создавать записи о походах, в которых указывается количество туристов и количество калорий, которое каждый человек потребляет в день. Также можно составлять меню на каждый день похода, в котором рассчитывается общая калорийность и вес еды, а также калорийность каждого приёма пищи в отдельности. Из дополнительных функций программа предоставляет перечень предлагаемых в поход продуктов, который можно редактировать. Пример работы с программой представлен на рисунках 13, 14 и 15.

Поход 1

Название

Продолжительность похода
13 дней

Первый день похода

февраля 2023							
вс	пн	вт	ср	чт	пт	сб	
4	29	30	31	01	02	03	04
5	05	06	07	08	09	10	11
6	12	13	14	15	16	17	18
7	19	20	21	22	23	24	25
8	26	27	28	01	02	03	04
9	05	06	07	08	09	10	11

Последний день похода

февраля 2023							
вс	пн	вт	ср	чт	пт	сб	
4	29	30	31	01	02	03	04
5	05	06	07	08	09	10	11
6	12	13	14	15	16	17	18
7	19	20	21	22	23	24	25
8	26	27	28	01	02	03	04
9	05	06	07	08	09	10	11

Дальше

НАЗВАНИЕ

КАЛОРИИ

Рисунок 13 – Создание записи похода в приложении «Раскладка»

Поход 1

План калорийности 1

Расход калорий

Количество человек

План калорийности 2

Расход калорий

Количество человек

Удалить

Добавить план

Сохранить

НАЗВАНИЕ

КАЛОРИИ

Среднему мужчине нужно около 2500 ккал в день. Женщине около 2200.

Если вы не знаете какое количество каллорий нужно именно вам, используйте значение по-умолчанию.

Расчёт каллорий можно игнорировать и приложение превратится в список необходимых продуктов.

Рисунок 14 – Учёт потребляемых калорий в приложении «Раскладка»

Поход 1

Отчёт закупок | Текстовый

9 февр. 10 февр. 11 февр. 12 февр. 13 февр. 14 февр.
15 февр. 16 февр. 17 февр. 18 февр. 19 февр. 20 февр.
21 февр.

3 000 г

9 февр. 2023 г.

kcal: 408 + 4 092 → 4 500

1 600 г

Завтрак

33%
136

× Брюква 400 г → 136 >

Обед

25%
100

× Брюссельская капуста вареная 400 г → 100 >

Ужин

42%
172

× Баклажан 400 г → 72 >

× Брюссельская капуста вареная 400 г → 100 >

10 февр. 2023 г.

kcal: 2 588 + 1 912 → 4 500

1 400 г

Завтрак

75%
1936

× Батон белый 800 г → 1 936 >

Продукты

Корзина (0)

Поиск

< СлiВaг	68 г	250
< Абрикос-ядро	100 г	520
< Авокадо	100 г	161
< Антрекоты, прожаренные на жире	120 г	384
< Арахис	100 г	552
< Баклажан	100 г	18
< Баранки, сушки	70 г	190
< Батон белый	100 г	242
< Батончики Эдемский сад	50 г	150
< Брокколи свежие	100 г	25
< Брюква	100 г	34
< Брюссельская капуста вареная	100 г	25
< Брюссельская капуста свежая	100 г	34
< Варенье, джем	50 г	145
< Вафли	80 г	344
< Винегрет	100 г	92
< Вишня сушёная	100 г	292
< Вишня-ядро	100 г	422
< Вода	1 000 г	
< Галеты	70 г	238

Рисунок 15 – Составление меню в приложении «Раскладка»

В качестве недостатков приложения стоит отметить отсутствие списка имеющихся в походе продуктов. Также нет возможности составлять распределение продуктов по рюкзакам туристов для транспортировки.

В дополнение ко всему данное программное обеспечение не подходит для использования на мобильных устройствах. Из-за этого его можно использовать только перед походом во время планирования и невозможно будет в пути свериться с меню.

Outdoor Food Ration Calculator

Это ещё один пример того, каким может быть приложение для организации питания в туристическом походе. Подобно предыдущей программе функционал этой разработки позволяет вести учёт количества туристов, веса и калорийности еды, а также составлять меню на каждый день похода. Также приложение ведёт список имеющихся в походе продуктов и рассчитывает общую калорийность всех припасов вместе с необходимым количеством калорий в день на человека. Приложение предназначено для использования на персональных компьютерах. Функционал приложения представлен на рисунках 16, 17 и 18.

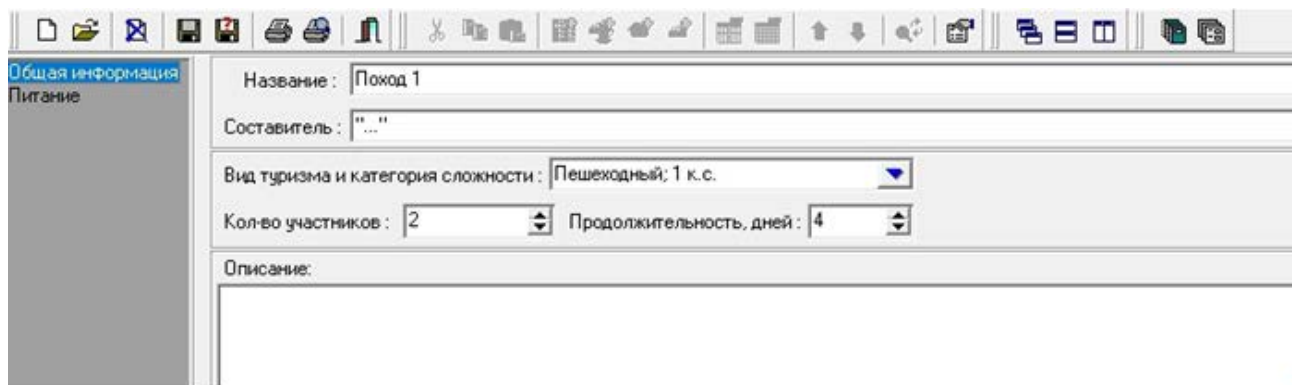


Рисунок 16 – Создание записи о походе в приложении «Outdoor Food Ration Calculator»

Наименование	ККал.	Вес, г. ↓	Белки, г.	Жиры, г.	Углев, г.	Б/Ж/У	Вес на гр, кг
<input checked="" type="checkbox"/> День 1	1133	400	34,60	3,80	233,80	1,0/0,1/6,8	0,800
<input type="checkbox"/> Баранки, сушки	816	300	25,80	1,50	170,40	1,0/0,1/6,6	0,600
<input checked="" type="checkbox"/> Завтрак	317	100	8,80	2,30	63,40	1,0/0,3/7,2	0,200
<input type="checkbox"/> Гречка	317	100	8,80	2,30	63,40	1,0/0,3/7,2	0,200
День 2	0	0	0,00	0,00	0,00	0/0/0	0,000
День 3	0	0	0,00	0,00	0,00	0/0/0	0,000
День 4	0	0	0,00	0,00	0,00	0/0/0	0,000

Рисунок 17 – Составление меню в приложении «Outdoor Food Ration Calculator»

ИТОГО [-203:0:0]	ККал. ↓	Дефицит, ККал.	Вес, кг. ↓	Белки, г.	Жиры, г.	Углев, г.	Б/Ж/У	Сумма, ?
<input checked="" type="checkbox"/> На человека в день	283	2817	0,100	8,65	0,95	58,45		
<input type="checkbox"/> - рекомендуемые значя	3100	не более 930		150,00	180,00	690,00		
<input checked="" type="checkbox"/> На человека за все дни	1133	11267	0,400				1,0/0,1/6,8	
На группу за все дни			0,800					

Рисунок 18 – Просмотр необходимой калорийности в приложении «Outdoor Food Ration Calculator»

Недостатки у этого приложения такие же, как и у предыдущего. Самым большим недостатком является невозможность использования на мобильных устройствах, следовательно, и в походе тоже. Помимо этого у данной разработки неудобный, устаревший интерфейс. Также нет возможности распределения продуктов между туристами для переноса.

Сведём данные проведённого анализа в таблицу и сравним их по определённому ранее списку необходимых функций:

Таблица 1 – Сравнение аналогов по критериям

Приложение Критерий	«Дневник еда планировщик»	«Учёт продуктов»	«Раскладка»	«Outdoor Food Ration Calculator»
Работа на мобильных устройствах	+	+	-	-
Возможность составления меню	+	-	+	+
Возможность составления списка продуктов на основании меню	-	+	-	+
Расчёт веса потребляемой в походе пищи	-	-	+	+
Распределение продуктов для транспортировки	-	+	-	-

Таким образом, проведя анализ мобильных приложений, а также туристических приложений на другие платформы, приходим к выводу о необходимости разработки туристического приложения для работы с продуктами в походе.

Разрабатываемое приложение должно предоставлять полный функционал по мониторингу продуктовых ресурсов от создания списка необходимых в поход продуктов и составления меню на каждый день, до рационального распределения общего количества еды между туристами в соответствии с их физическими возможностями для обеспечения комфортной транспортировки.

Разработку необходимо производить под мобильные устройства, чтобы участники похода имели возможность на любом этапе пути свериться с планом или произвести его корректировку.

2 ОБЗОР СРЕДСТВ РЕАЛИЗАЦИИ

Определив направление разработки и требования к приложению, следующим этапом необходимо выбрать программные средства, наиболее подходящие для реализации проекта.

В качестве мобильной операционной системы, под которую будет разрабатываться приложение, следует использовать операционную систему Android. Подобное решение объясняется тем, что по данным сайта statcounter на апрель 2023 года доля Android среди мобильных операционных систем в мире составляет 68.61%.

Из этих данных можно сделать вывод, что разработанное под наиболее популярную мобильную операционную систему приложение позволит добиться наибольшей распространённости программы среди пользователей.

Таким образом, для реализации проекта остаётся выбрать следующие средства: среда разработки, язык программирования, база данных.

2.1 Среда разработки

Android Studio

Android Studio – это интегрированная среда разработки (IDE), разработанная компанией Google в 2014 году. В Android Studio собраны весь необходимый функционал для быстрого создания приложений на смартфоны с операционной системой Android. Эта среда разработки основанна на мощном редакторе кода IntelliJ IDEA, что позволяет использовать больше возможностей, повышающих производительность при создании приложений для Android, таких как [1]:

- гибкая система сборки на основе Gradle;
- быстрый и многофункциональный эмулятор;
- шаблоны кода и интеграция с GitHub помогающие создавать общие функции приложения и импортировать примеры кода;
- обширные инструменты и фреймворки для тестирования;
- инструменты Lint для определения производительности, удобства использования, совместимости версий и других проблем.

Данное программное обеспечение является официальным инструментом для разработки Android-приложений, поскольку разрабатывалось компанией, являющейся частью Google.

Xamarin

Xamarin – это фреймворк для кроссплатформенной разработки мобильных приложений для платформ iOS, Android и Windows Phone, используемый в среде разработки Visual Studio.

Несмотря на то что код, связанный с бизнес-логикой, доступом к базе данных и сетевыми коммуникациями, может использоваться на всех платформах, Xamarin позволяет создавать слой кода UI под конкретную платформу. Таким образом, кроссплатформенные приложения Xamarin выглядят на 100% нативными на любом устройстве, обеспечивая наилучший пользовательский опыт по сравнению со стандартными гибридными приложениями [2].

Фреймворк Xamarin обладает следующими преимуществами:

- единый стек технологий для разработки на всех платформах;
- производительность близка к нативной;
- полный пакет инструментов разработки.

Также у этого фреймворка есть следующие недостатки:

- большой размер приложений;
- ограниченный доступ к open-source библиотекам;
- малое сообщество программистов.

Использование этого инструмента для разработки позволяет создавать приложения под несколько мобильных платформ, что ускоряет разработку. Однако программисту придётся столкнуться с ограниченностью инструментов для разработки, т.к. необходимая функция может быть не реализована вследствие малого сообщества программистов.

Таким образом, под задачи реализуемого проекта больше подходит Android Studio. Разработанная компанией Google, эта среда разработки предоставляет необходимые инструменты для создания мобильных приложений, соответствующих современным стандартам дизайна и функциональности. Работа в этой среде разработки позволит значительно увеличить быстродействие создаваемого программного продукта.

2.2 Язык программирования

Основными языками программирования для выбранной на предыдущем этапе среды разработки являются Java и Kotlin. Следовательно, выбор будет производиться среди них.

Java

Java – это объектно-ориентированный язык программирования общего назначения. На нем можно писать всё: от серверной части до десктопных и мобильных приложений. Долгое время этот язык программирования был практически единственным средством для разработки качественных программных продуктов под мобильные платформы.

Преимущества:

- платформенно-независим;
- возможность неявных преобразований;
- наличие статических элементов и тернарных операторов;
- большое сообщество и количество документации;
- большой функционал для разработки;
- множество существующих проектов написано на Java.

Работа с этим языком программирования обеспечит многообразие существующих эффективных решений для задач разработчика.

Kotlin

Kotlin – это статически типизированный язык программирования, созданный компанией JetBrains [3]. В 2017 году Google объявил, что язык программирования Kotlin теперь является предпочтительным языком для

разработчиков приложений для Android [4]. Этот факт подтвердил конкурентоспособность языка Kotlin в сфере мобильной разработки. В сравнении с Java, Kotlin имеет следующие преимущества:

- современный, набирающий популярность, язык программирования;
- лаконичность кода позволяющая ускорить работу программы и разработку;
- программы используют фреймворки и библиотеки Java;
- Kotlin компилируется в байткод JVM или в JavaScript;
- безопасность Null-типов;
- совместимость с Java, позволяющая Java код преобразовать в Kotlin.

Однако малое сообщество программистов на этом языке и малое количество документации может ограничить разработчика в поиске решения для реализации необходимой функции в проекте. Но этот недостаток исправляется совместимостью Java и Kotlin.

Из этого следует, что выбор языка Kotlin для реализации проекта позволит использовать современные решения для реализации, отчего разработанное приложение долгое время не будет терять актуальность в качестве быстрогодействия и функциональности.

2.3 Система управления базой данных

Также для правильного функционирования разрабатываемого приложения необходима база данных, чтобы хранить данные о работе с приложением.

Так как разрабатываемое туристическое приложение подразумевает использование в условиях отсутствия Интернет-соединения, необходимо, чтобы система управления базой данных (СУБД) располагалась непосредственно на устройстве пользователя для возможности сохранения данных. Такие СУБД называются встраиваемыми.

Встраиваемая СУБД – СУБД, которая может поставляться как составная часть некоторого программного продукта, не требуя процедуры самостоятельной установки. Встраиваемая СУБД предназначена для локального хранения данных своего приложения и не рассчитана на коллективное использование в сети [5].

Berkeley DB

Berkeley DB – это нереляционная встроенная СУБД, поддерживаемая компанией Oracle.

Преимущества:

- минимальные конфигурирование и установка;
- простота;
- быстрый доступ к данным;
- легкость администрирования;
- легкость установки;
- очень эффективные форумы поддержки.

СУБД типа NoSQL(нереляционные) отличаются скоростью работы с информацией, однако за подобные достоинства придется жертвовать надёжностью работы с данными вследствие неструктурированного хранения информации.

SQLite

SQLite – это наиболее популярная в мобильной разработке реляционная встраиваемая СУБД. В отличие от других систем баз данных для SQLite не требуется сервер базы данных. SQLite представляет встраиваемый движок базы данных, который обращается напрямую к файлу базы данных на диске [6].

Преимущества:

- высокая скорость;
- хранение данных в одном файле;
- надёжность;
- малый размер полностью сконфигурированной базы данных;
- кроссплатформенность;
- автономность системы от стороннего программного обеспечения (ПО), библиотек или фреймворков.

Таким образом, наиболее предпочтительным вариантом в зависимости от уже выбранных средств реализации является SQLite, поскольку выбранная среда разработки Android Studio предоставляет функционал по работе с этой СУБД, что обеспечивает лучшую интеграцию базы данных в проект и улучшает быстродействие.

3 РАЗРАБОТКА ПРИЛОЖЕНИЯ

С использованием выбранных на предыдущем этапе инструментов была осуществлена разработка приложения.

В этой главе рассмотрены следующие вопросы: структура используемой базы данных, входные и выходные данные, задействованные в приложении, а также принцип работы функционала приложения.

3.1 Архитектура базы данных

Для хранения используемых в приложении данных необходимо спроектировать базу данных. Архитектура базы данных представлена на рисунке 19.

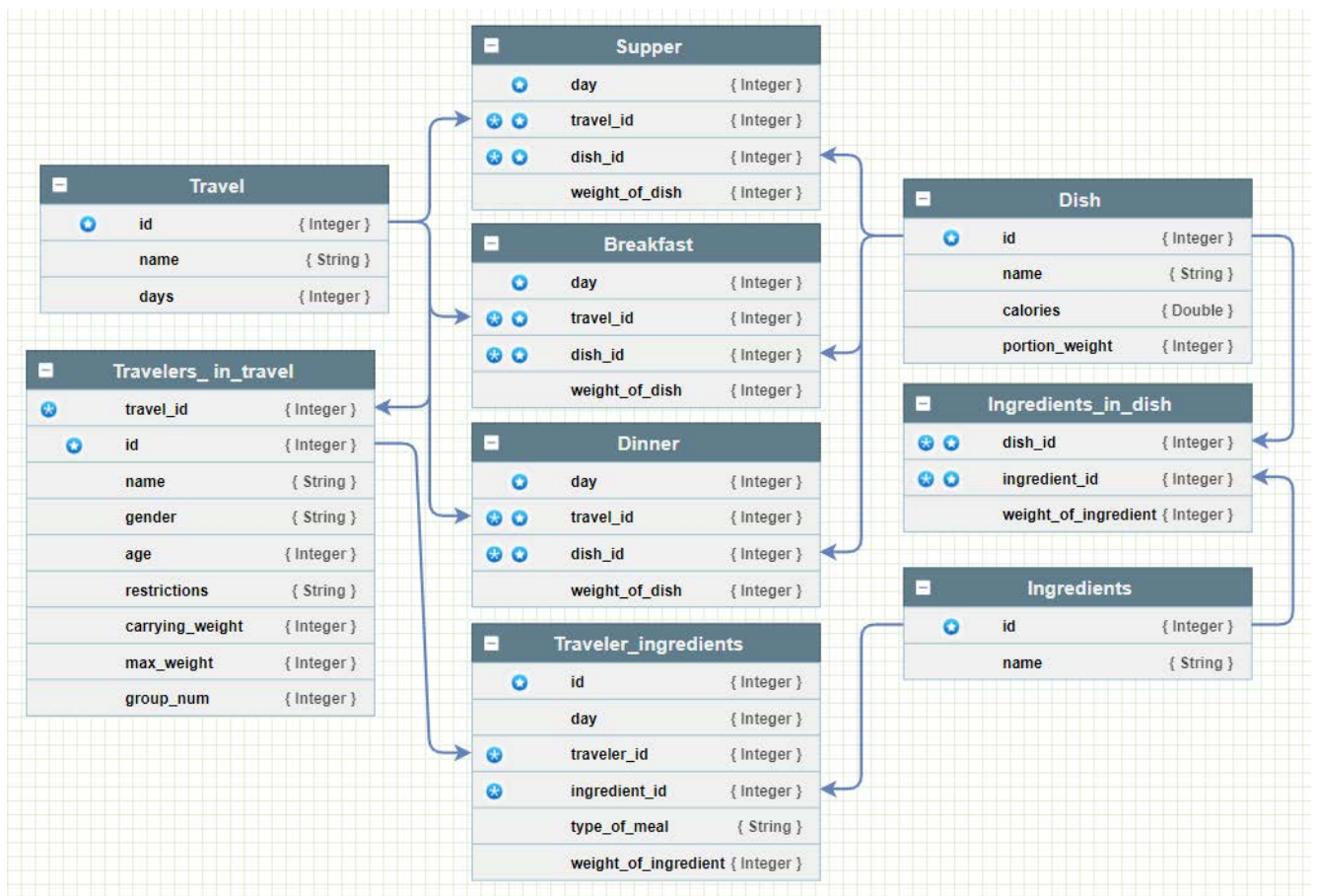


Рисунок 19 – Структура используемой базы данных

Таблица Dish содержит информацию о различных блюдах и предназначена для предоставления информации о блюдах, которые туристы могут использовать в походе.

Таблица Ingredients предназначена для хранения названий ингредиентов для блюд.

Таблица Travel содержит информацию о созданных походах: название и количество дней.

Таблицы Breakfast, Dinner и Supper предназначены для хранения информации о завтраках, обедах и ужинах соответственно. В таблицах есть внешний ключ travel_id, связывающий их с полем id таблицы Travel. А также внешний ключ dish_id, связывающий их с полем id таблицы Dish.

Таблица Ingredients_in_dish предоставляет информацию о том, какие ингредиенты и в каких блюдах находятся. Используется для связи таблиц Dish и Ingredients.

Таблица Travelers_in_travel содержит информацию о туристах, которые участвуют в походе. Связана с таблицей Travel.

Таблица Traveler_ingredients предназначена для предоставления информации о продуктах, которые переносит каждый турист. Таблица используется для связи таблиц Dish и Travelers_in_travel.

3.2 Описание входных и выходных данных

3.2.1 Входные данные

Для создания похода, пользователю необходимо ввести название, чтобы добавить уникальности каждой записи о походе, а также, ввести количество дней, на которое планируется поход.

Для составления списка участников похода требуется следующая информация: имя, возраст, пол, наличие либо отсутствие противопоказаний в виде травм или прочих недугов.

Для составления меню, пользователю предоставляется перечень блюд, записи о которых содержатся в базе данных. Таким образом, входными данными будут:

- выбранная пользователем запись блюда, содержащая название, калорийность и вес порции;
- вес в граммах.

Для расчёта максимальной нагрузки на туриста необходима информация об общем весе имеющихся продуктов.

Для распределения продуктов между туристами используется следующая информация:

- перечень блюд в меню;
- ингредиенты необходимые для блюда и их вес;
- рассчитанная максимальная нагрузка на туристов.

3.2.2 Выходные данные

Под выходными данными в разработанном приложении подразумеваются результаты работ программных алгоритмов, представленные в одном из следующих вариантов: создание записи в базе данных, расчёт конкретного значения.

Результатом работы функции создания похода является составление записи о походе в таблице Travel.

В результате составления списка туристов данные о туристах помещаются в таблицу Travelers_in_travel.

В результате составления меню данные о блюдах, помещаются в таблицы Breakfast, Dinner или Supper, в зависимости от приёма пищи, к которому относится блюдо.

В результате расчёта максимальной нагрузки на туриста получается значение максимального веса, который может переносить участник похода.

Результатом распределения продуктов между туристами является создание записи в таблице Traveler_ingredients, отражающей какой продукт, у какого туриста находится и в каком количестве.

3.3 Описание разработанных алгоритмов

3.3.1 Создание похода

Данная функция используется для ввода данных о походе: название и количество дней. Также активация этой функции предоставляет доступ к остальному функционалу приложения.

3.3.2 Составление списка туристов

С помощью этой функции составляется список участников похода. От характеристики туриста зависит максимальный вес продуктов, который он может переносить.

При добавлении туриста, пользователю предлагается ввести имя, возраст, пол, а также указать наличие медицинских противопоказаний, влияющих на переносимый вес. После завершения ввода данных информация сохраняется в таблицу Traveler_in_travel с добавлением информации о походе, в котором записан турист.

3.3.3 Составление меню

Данная функция позволяет составить план приёмов пищи. Создание меню обеспечивает работу остального функционала приложения.

Во время составления меню пользователь выбирает блюдо из вариантов, предложенных программой. Далее программа рассчитывает необходимый вес блюда для трапезы всех участников похода, и записывает в одну из таблиц

Breakfast, Dinner или Supper, в зависимости от приёма пищи, в которое добавлялось блюдо.

3.3.4 Составление списка продуктов на основании меню

С помощью этой функции приложение составляет список продуктов, которые являются ингредиентами для блюд, указанных в меню.

Программа ищет в базе данных информацию о блюде, добавленном в определённый приём пищи. После чего ищет информацию об ингредиентах, необходимых для одной порции этого блюда и добавляет полученные данные, умноженные на количество туристов, в список продуктов.

3.3.5 Расчёт веса потребляемой в походе пищи

Используя созданный список туристов, рассчитывается максимальный вес всех продуктов, который необходимо будет нести этим туристам.

Общий максимальный вес переносимых продуктов рассчитывается из среднего веса еды, потребляемого человеком за один приём пищи, равного 700 грамм. Это значение умножается на количество приёмов пищи в день, равное 3, а также на количество участников похода и на планируемое количество дней похода.

В дальнейшем, используя данные об общем весе имеющихся продуктов, рассчитывается максимальная нагрузка, приходящаяся на туриста.

3.3.6 Расчёт максимальной нагрузки на туриста

Это значение рассчитывается на основе общей массы блюд, запланированных в походе, и определяет, какой максимальный вес будет переносить участник похода. Этот параметр используется в качестве верхнего предела нагрузки на туриста при распределении общего веса ингредиентов для транспортировки. Участники похода не будут загружены больше, чем указано в их параметре максимальной нагрузки. При сложении показателей предельного переносимого веса всех туристов, полученное значение будет равно общему весу ингредиентов, взятых в поход. Таким образом, все продукты будут распределены.

Для расчёта максимальной нагрузки, общая масса имеющихся продуктов делится на количество участников похода. Полученный результат является средней нагрузкой на одного туриста. Затем происходит уменьшение этого значения в зависимости от характеристик туриста.

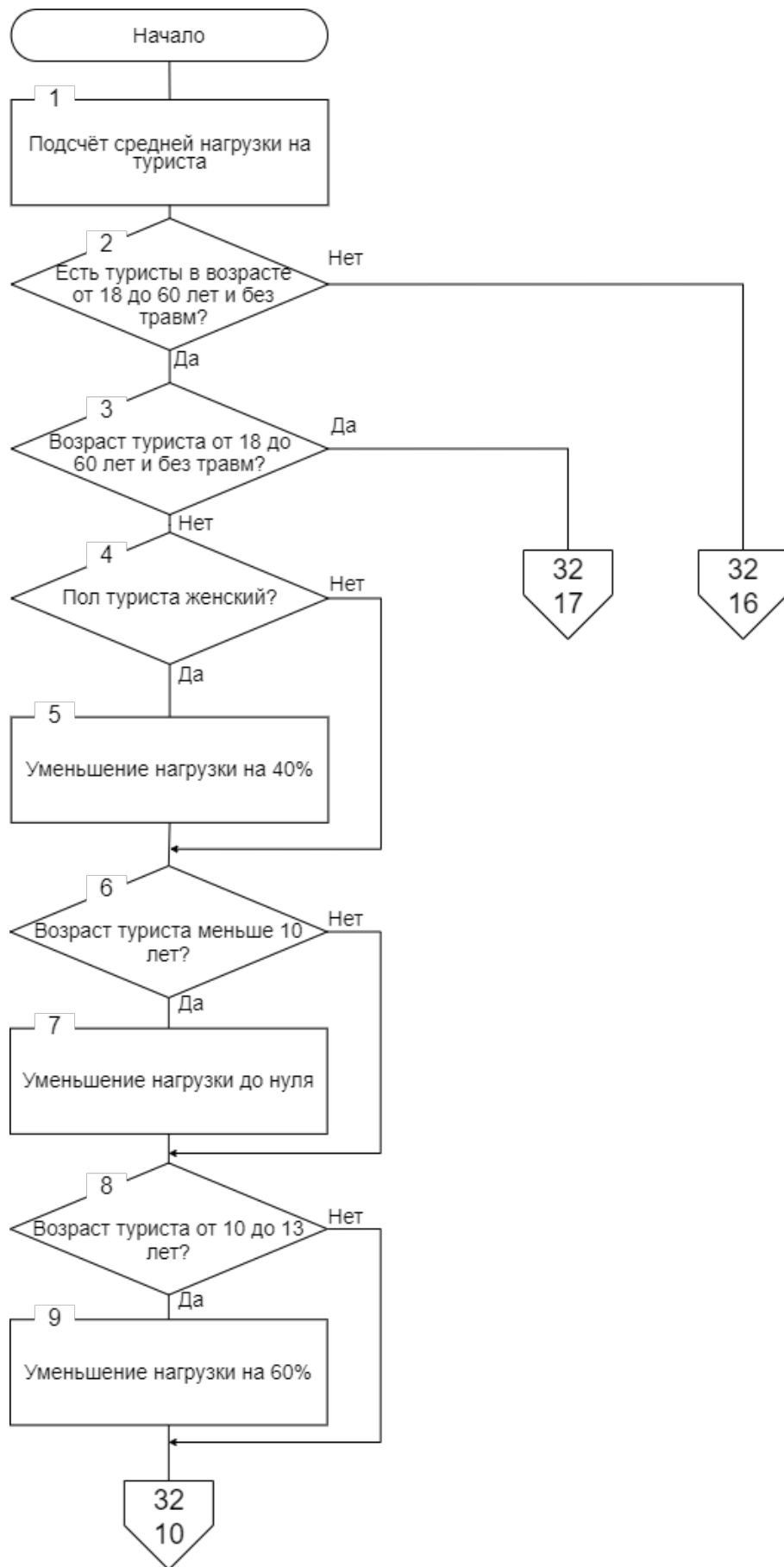
- Если пол туриста женский, то нагрузка уменьшается на 40%.
- Если возраст туриста меньше 10 лет, нагрузка уменьшается до нуля.
- Если возраст туриста от 10 до 13 лет включительно, то нагрузка уменьшается на 60%.
- Если возраст туриста от 14 до 17 лет включительно, то нагрузка уменьшается на 40%.

- Если возраст туриста больше 60 лет, то нагрузка уменьшается на 20%.
- Если у туриста указано наличие медицинских противопоказаний, то нагрузка уменьшается на 60%.

Перечисленные ограничения могут накладываться друг на друга, значительно уменьшая максимальную нагрузку на человека. Вся нагрузка, от которой были освобождены туристы с соответствующими характеристиками, распределяется между туристами без противопоказаний в возрасте от 18 до 60 лет включительно.

В случае если среди участников похода нет туристов без противопоказаний в возрасте от 18 до 60 лет, способных принять уменьшенную нагрузку других туристов, уменьшение показателя максимальной нагрузки не производится, и все участники похода будут иметь одинаковое значение максимальной нагрузки.

Схема алгоритма представлена на рисунке 20.



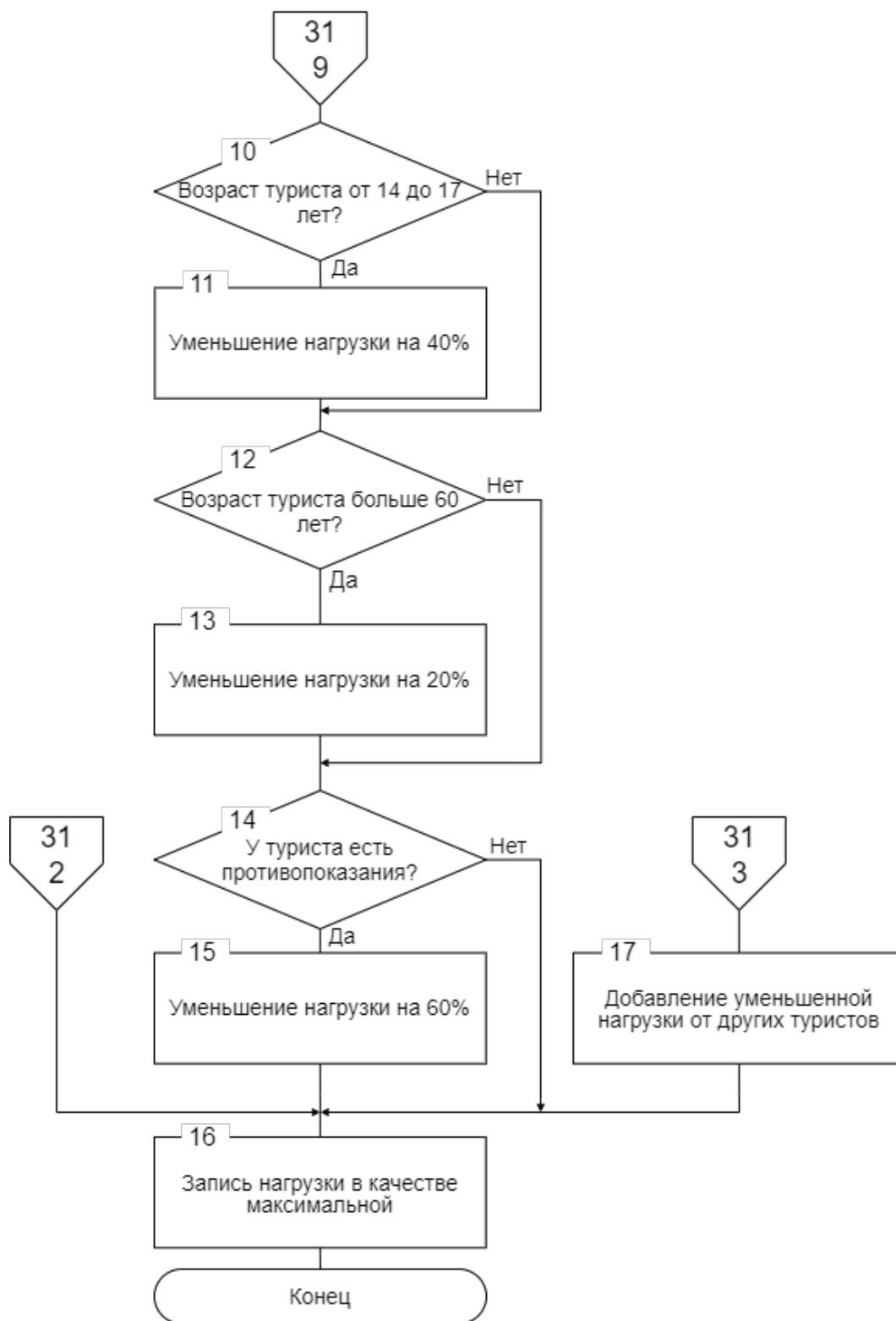


Рисунок 20 – Схема алгоритма расчёта максимальной нагрузки

3.3.7 Распределение продуктов между туристами для транспортировки

Данная функция является одной из основных в приложении, требующей наиболее рационального подхода. Поэтому необходимо рассмотреть существующие решения, связанные с данной задачей.

1) *Транспортная задача*

Это одна из основных задач линейного программирования, направленная на достижение оптимального плана перевозок груза.

Суть решения состоит в том, чтобы обеспечить транспортировку груза из пункта А в пункт Б с минимальными затратами на перевозку.

Однако, применительно к разрабатываемой для туристического приложения функции, данный алгоритм не подходит. Поскольку в решении задачи применяется транспортировка не ограниченная максимальной нагрузкой, чего нельзя сказать о туристах с верхним порогом максимального веса. Также в транспортной задаче ограничением алгоритма является стоимость перевозок, которую требуется сделать минимальной. В разрабатываемой же функции эквивалента данному ограничению нет, поскольку при транспортировке туристами продуктов каких-либо материальных затрат нет.

2) *Задача оптимизации показателя равномерности использования ресурсов*

Задача сводится к достижению равномерности потребления ресурсов путём организации процессов использования этих ресурсов.

В случае внедрения этого алгоритма необходимо будет произвести равномерное распределение самостоятельно путём разделения общей массы продуктов на всех участников похода. В таком случае распределение сводится равномерному использованию продуктов у каждого туриста, чтобы не допустить полной разгрузки одного туриста раньше всех остальных.

Данная задача оптимизации составляет равномерное использование имеющихся ресурсов путём размещения во времени разных процессов, использующих те или иные ресурсы из имеющихся. Каждый процесс тратит фиксированное количество строго определённых ресурсов. Для каждого процесса используемые ресурсы могут отличаться, но они изначально определены.

Подобное решение не подходит под реализацию равномерного потребления продуктов туристами. Алгоритм предполагает автоматическое распределение во времени процессов затраты имеющихся ресурсов, в нашем случае распределение блюд по приёмам пищи в меню. Однако при составлении меню пользователь самостоятельно занимается таким распределением блюд по приёмам пищи, и оно не будет обеспечивать равномерность потребления продуктов. Также для работы алгоритма необходима информация об имеющихся ресурсах, то есть продуктах, Однако список имеющихся в походе продуктов составляется приложением в процессе распределения блюд по приёмам пищи, то есть в тот момент, в который должен использоваться алгоритм решения задачи оптимизации показателя равномерности использования ресурсов. Таким образом,

данное решение не может быть задействовано в реализации функции распределения продуктов между туристами.

Рассмотрев несколько существующих реализаций алгоритмов распределения, приходим к выводу, что они не учитывают все нюансы организации питания для туристов. Таким образом, необходимо самостоятельно разработать алгоритм распределения.

Для реализации алгоритма необходимо использовать меню, составленное пользователем на этапе создания записи о походе. Это меню представляет собой план потребления продуктов туристами, а значит подходит для организации равномерного уменьшения нагрузки между всеми участниками похода.

При проектировании равномерного распределения продуктов необходимо учитывать размер группы туристов, транспортирующих продукты для приёма пищи. Пренебрежение этим параметром приведёт к неэффективности спроектированного алгоритма.

Если выделить по одному человеку на каждый приём пищи для транспортировки ингредиентов, придётся столкнуться с ситуацией, когда один турист полностью разгрузился на первом приёме пищи и более не участвует в переносе продуктов. Такой вариант не соответствует цели приложения по обеспечению равномерной нагрузки на каждого участника похода и не должен рассматриваться.

Обратная же ситуация, когда продукты для каждого приёма пищи переносят все участники похода. При этом вес ингредиентов на одну трапезу равномерно разделяется на всех туристов. Данный подход позволит обеспечить наиболее равномерно снижающуюся нагрузку у всех туристов, к чему и стремится разрабатываемая программа. Однако этот метод является наименее рациональным в плане удобства, а также затрачиваемого времени на распаковку каждым туристом своего рюкзака, поиска необходимых ингредиентов и передаче их повару для готовки.

Таким образом, необходимо выбрать наиболее усреднённый вариант двух описанных методов. В этом варианте все туристы делятся на три группы, соответствующие каждому приёму пищи: завтраку, обеду и ужину; и занимаются транспортировкой ингредиентов для этих приёмов пищи.

- Первая группа занимается транспортировкой продуктов для всех завтраков.
- Вторая группа занимается транспортировкой продуктов для всех обедов.
- Третья группа занимается транспортировкой продуктов для всех ужинов.

При распределении ингредиентов, каждой группе выделяется количество пищи, необходимое для трапез, за которые ответственна группа. То есть первой группе выделяется вес продуктов, необходимых для всех завтраков, второй группе для всех обедов и третьей группе для всех ужинов.

Распределение между группами осуществляется, начиная с последнего дня похода, на который было составлено меню. Подобное решение позволит

обеспечить наиболее равномерную разгрузку туристов на всём протяжении похода, т.к. туристы с меньшей грузоподъёмностью будут загружаться продуктами, которые понадобятся в конце похода, и разгрузятся только в конце, а туристы с большей грузоподъёмностью будут переносить больший вес, но будут разгружаться ежедневно.

После выдачи продуктов, в каждой группе происходит равномерное распределение выделенных ингредиентов на количество участников группы. Во время распределения программа опирается на показатель предельного веса, который будет переносить участник похода, полученный в алгоритме расчёта максимальной нагрузки на туриста. Исходя из этого значения, туристу нельзя будет выдать вес, больший, чем показатель максимальной нагрузки, даже если в соответствии с равномерным распределением выделенных на группу ингредиентов этому туристу необходимо взять больший вес. Таким образом, участник похода будет нагружен в соответствии со своими физиологическими характеристиками. Также подобное решение позволит избежать ситуации, когда в одной группе меньше туристов, чем в остальных и на каждого человека приходится больший вес, в сравнении с другими группами.

Избыточный вес, который не удалось распределить туристу, распределяется другим участникам его группы, нагрузка которых не достигла максимального значения. В случае если переносимый вес всех участников группы соответствует предельному значению, нагрузка распределяется участникам других групп. Среди участников других групп, которым можно распределить ингредиенты, выбирается человек с наименьшим переносимым весом. Подобные действия выполняются до тех пор, пока весь избыточный вес не будет распределён. Таким образом, даже если одной группе будет выделен вес ингредиентов, превышающий пределы нагрузки участников, продукты в любом случае будут распределены между туристами.

Описанный способ распределения позволяет обеспечить ежедневное уменьшение нагрузки у всех участников похода, а также минимизирует затраты времени и сил на сбор всех необходимых ингредиентов для блюд между туристами. Также пользователю предоставляется информация о распределении продуктов между туристами.

Схема алгоритма представлена на рисунке 21.

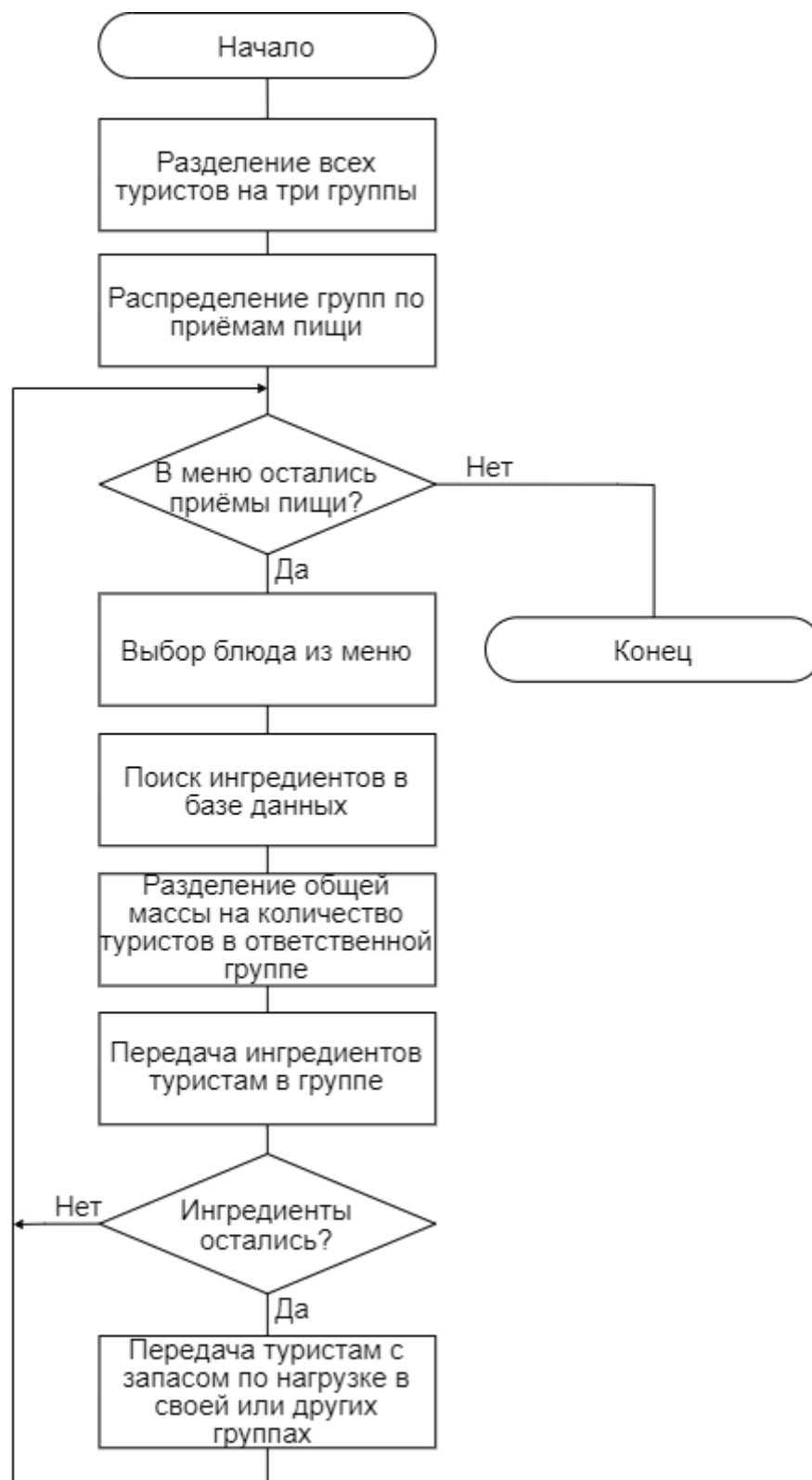


Рисунок 21 – Схема алгоритма распределения продуктов между туристами

4 ТЕСТИРОВАНИЕ ПРИЛОЖЕНИЯ

После реализации необходимо провести тестирование для исправления критических ошибок и проверки соответствия разработанного приложения составленным требованиям.

4.1 Виды использованных тестирований

4.1.1 Модульное тестирование

Модульное тестирование – это вид тестирования, при котором разрабатываемая программа разбивается на отдельные модули. Полученные модули тестируются изолированно от всей программы. В качестве модуля могут выступать: отдельный алгоритм, функция или класс.

Целью модульного тестирования является проверка правильности функционирования отдельных частей программы и выявление неисправностей на ранних этапах. Это позволяет ускорить процесс разработки, экономя время при дальнейших тестированиях.

4.1.2 Интеграционное тестирование

Интеграционное тестирование – это тестирование, при котором разрабатываемые модули программного продукта объединяются для проверки их общей работоспособности. Интеграционное тестирование целесообразно применять после проверки каждого модуля по отдельности.

Данный вид тестирования проводится для поиска неисправностей, пропущенных после модульного тестирования, а также для выявления новых дефектов, возникающих при работе двух и более модулей вместе.

4.1.3 Функциональное тестирование

Функциональное тестирование – это наиболее важный вид тестирования, позволяющий определить, соответствие функционала разработанного программного продукта составленным требованиям к приложению. Функциональное тестирование выполняется после завершения этапа реализации.

Основная цель заключается в поиске ошибок логики работы функционала программы. Для функционального тестирования составляются перечни результатов, ожидаемых от использования той или иной функции разработанного приложения.

4.2 Результаты тестирований

Методы модульного и интеграционного тестирования применялись в процессе реализации программного обеспечения, что позволило улучшить понимание кода проекта и повысить эффективность разработки.

В результате проведения модульного тестирования было исправлено множество дефектов, приводящих к некорректной работе программного обеспечения.

Интеграционное тестирование позволило отследить ошибки, возникающие при взаимодействии модулей программного обеспечения и улучшить их взаимодействие.

Функциональное тестирование проводилось во время разработки, а также после завершения этапа реализации приложения. Это тестирование помогло разработать необходимый функционал, а также проверить его на соответствие заявленным требованиям. Перечень тестов для функций приложения представлен в таблице 2.

Таблица 2 – Перечень тестов для функционального тестирования

№	Тестируемая функция	Описание тестирования	Ожидаемый результат	Полученный результат
1	Создание похода	<ol style="list-style-type: none"> 1. Запустить приложение. 2. Нажать на кнопку добавления похода. 3. Ввести данные о названии похода и количестве дней. 4. Нажать кнопку завершения создания похода. 	<p>В результате нажатия на кнопку добавления похода должна открыться новая страница с полями для ввода.</p> <p>В результате нажатия на кнопку завершения создания похода страница с созданием похода закроется, а запись о походе добавится в таблицу Travel</p>	Соответствует ожидаемому
2	Составление списка туристов	<ol style="list-style-type: none"> 1. Нажать на кнопку добавления туриста на странице создания похода. 2. Ввести данные о туристе. 3. Нажать на кнопку добавления туриста. 	<p>В результате нажатия на кнопку добавления туриста должна открыться новая страница с полями для ввода.</p> <p>В результате нажатия на кнопку добавления турист должен появиться в списке, а запись о нём добавиться в таблицу Travelers_in_travel.</p>	Соответствует ожидаемому

Продолжение таблицы 2

№	Тестируемая функция	Описание тестирования	Ожидаемый результат	Полученный результат
3	Составление меню	<ol style="list-style-type: none"> 1. Нажать на кнопку добавления блюда в меню. 2. Выбрать блюдо из предлагаемых. 3. Нажать кнопку подтверждения. 	<p>В результате нажатия на кнопку добавления должна открыться вкладка со списком блюд. В результате нажатия на кнопку подтверждения блюдо должно появиться в меню, а запись о нём добавиться в одну из таблиц Breakfast, Dinner или Supper.</p>	Соответствует ожидаемому
4	Составление списка продуктов на основе меню	<ol style="list-style-type: none"> 1. Составить меню похода. 2. Завершить создание похода. 3. Нажать кнопку просмотра похода. 4. Нажать на кнопку просмотра списка продуктов. 	<p>В списке продуктов должны появиться продукты, являющиеся ингредиентами только что добавленного блюда.</p>	Соответствует ожидаемому
5	Расчёт максимальной нагрузки на туриста	<ol style="list-style-type: none"> 1. Добавить нового туриста в список. 2. Добавить новое блюдо в меню. 3. Завершить создание похода. 	<p>В результате должна рассчитаться максимальная нагрузка и записаться туристу в таблицу Travelers_in_travel.</p>	Соответствует ожидаемому

Окончание таблицы 2

6	Распределение продуктов между туристами для транспортировки	<ol style="list-style-type: none"> 1. Составить список туристов. 2. Составить меню. 3. Завершить создание похода 	В результате вес продуктов должен равномерно распределиться между туристами и записи о продуктах, переносимых туристами должны добавиться в таблицу <code>Traveler_ingredients</code>	Соответствует ожидаемому
---	---	---	---	--------------------------

4.3 Примеры функционирования наиболее важных алгоритмов

Вследствие высокой значимости и сложности алгоритма равномерного распределения продуктов между туристами необходимо привести примеры его работы для лучшего понимания принципов функционирования, а также для проверки равномерности распределения.

Первый пример представляет наиболее идеальный вариант для работы алгоритма, при котором обеспечивается самое равномерное распределение.

В походе участвуют 6 человек. Длительность похода составляет 3 дня, что соответствует 9 приёмам пищи. Общее количество туристов делится на 3 группы по 2 человека. Первая группа переносит продукты для трёх завтраков, вторая для трёх обедов и третья для трёх ужинов.

Согласно описанному ранее расчёту веса пищи в походе, вес продуктов на одну трапезу будет составлять 700 грамм, умноженные на 6 человек, то есть 4200 грамм еды. В сумме на весь поход понадобится 4200 грамм, умноженные на 9 приёмов пищи, то есть 37800 грамм.

Каждый человек из шести участников похода не имеет медицинских противопоказаний и находится в возрасте от 18 до 60 лет. То есть, при расчёте максимальной нагрузки на туристов, уменьшения пределов нагрузки не будет, и каждый человек должен будет переносить одинаковый вес продуктов, равный 37800 грамм, делённые на 6 человек, 6300 грамм.

Во время распределения ингредиентов на каждую из трёх групп будет выделено по 12600 грамм продуктов (4200 грамм, умноженные на 3 приёма пищи). Этот вес будет разделён каждой группой поровну между своими участниками, то есть каждая группа разделит 12600 грамм на 2 участника и получит 6300 грамм продуктов, которые должен нести один участник группы.

Таким образом, общий вес продуктов, равный 37800 грамм, будет разделён поровну между 6 участниками похода, и каждый человек будет нести по 6300 грамм еды, что равняется показателю их предельной нагрузки. Во время привалов туристы из такой группы будут разгружаться равномерно, т.к. переносят одинаковый вес.

Второй пример демонстрирует обработку ситуации, в которой у туристов из одной группы нельзя распределить выделенные на них продукты и приходится перераспределять избыток между другими группами.

В этот раз в походе участвуют 7 человек. Количество дней похода остаётся равным 3. Группы, на которые делятся туристы, будут иметь следующий состав: первая и вторая группы по 2 человека, а третья 3 человека.

Вес еды на один приём пищи увеличится и будет равен 700 грамм, умноженные на 7 человек, то есть 4900 грамм еды. Следовательно, на весь поход понадобится 4900 грамм, умноженные на 9 приёмов пищи, то есть 44100 грамм.

У всех туристов по-прежнему нет медицинских противопоказаний, а возраст составляет от 18 до 60 лет. Получается, что максимальный вес участников похода, как и в первом примере, будет одинаковым, равным 44100 грамм, делённые на 7 человек, 6300 грамм.

При разделении ингредиентов группы также получают равный вес, т.е. 14700 грамм продуктов (4900 грамм, умноженные на 3 приёма пищи) на группу. Однако, из-за разного количества участников, вес, который будет выделяться на одного участника, в разных группах будет отличаться. В третьей группе он будет составлять 14700 грамм разделить на 3 человека, 4900 грамм. А в первой и второй группах он будет равен 7350 грамм (14700 грамм разделить на 2 человека), что превышает значение максимальной нагрузки в 6300 грамм.

В таком случае в первой и второй группах будет распределена часть продуктов, которая будет равна 6300 грамм, умноженные на 2 человека, 12600 грамм, а избыток в 2100 грамм будет перенаправлен в третью группу, т.к. у участников этой группы переносимый вес меньше максимального значения.

В итоге туристы из первой и второй групп будут загружены по 6300 грамм ингредиентов для приёмов пищи, за которые они ответственны. А в третьей группе у участников будет переносимый вес в 4900 грамм для «своих» приёмов пищи, а также избыток от первой группы, равный 2100 грамм и избыток от второй группы, равный 2100 грамм. В сумме получится, что каждый член третьей группы переносит по 6300 грамм еды.

Также из-за особенности распределения, заключающейся в обработке дней меню с конца, туристы будут разгружаться равномерно, поскольку группы, которые будут помогать другим в переносе продуктов, разгрузят «чужие» продукты на первых привалах и дальше продолжат транспортировать продукты только для «своих» приёмов пищи. И, исходя из того, что участники групп с большим количеством туристов будут разгружаться медленнее, они начнут разгружаться раньше, чем группы с меньшим количеством участников, из-за того, что большие группы чаще будут помогать в транспортировке.

Третий пример иллюстрирует наиболее вероятную ситуацию, когда, помимо неравного количества людей в группах, у туристов ещё неравное значение максимальной нагрузки.

Условия похода остаются прежними: 7 участников, количество дней 3. Первая и вторая группы состоят из двух человек, а третья из 3. Общий вес еды, необходимой на весь поход, равен 44100 грамм.

Отличия в том, что, на этот раз, 6 туристов из 7 не имеют медицинских противопоказаний и их возраст равен от 18 до 60 лет. А один турист является пожилой женщиной с медицинскими противопоказаниями и возрастом более 60 лет. Таким образом, её показатель максимальной нагрузки будет равен 6300 грамм, которые получает каждый турист, умноженные на 0.8, на 0.6 и на 0.4. Что в результате даст 1210 грамм максимальной нагрузки. Подобные уменьшения проводятся из-за возраста туриста (на 20%), пола (на 40%), а также медицинских противопоказаний (на 60%). Весь уменьшенный вес, равный 5090 грамм, распределяется поровну между туристами без медицинских противопоказаний и с возрастом от 18 до 60 лет, в нашем случае таких туристов 6. Поэтому их предельные нагрузки увеличатся с 6300 грамм до $6300 + 849 = 7149$ грамм.

При распределении каждая группа по-прежнему получит по 14700 грамм для «своих» приёмов пищи. Для первой и второй групп на человека будет выделено по 7350 грамм, которые они также не смогут полностью взять и будут передавать другой группе. В третьей группе, в которой находится пожилая женщина, на человека выделяют по 4900 грамм еды. Двое других участников без проблем смогут взять этот вес, однако для пожилой женщины он будет слишком велик. В таком случае она возьмёт из 4900 грамм вес, равный 1210 граммам, а остаток в 3690 грамм отдаст другим участникам третьей группы со свободным местом. Если же у участников группы к этому моменту уже не будет свободного места, то избыток отдастся другим группам.

Таким образом, на двух оставшихся членов третьей группы будет выделена нагрузка в 4900 грамм еды, выделенной всем, 3690 грамм еды, которую не смогла взять пожилая женщина. А также избыток от первой группы, равный 402 грамма ($7350 - 7149 = 201$ грамм, умноженный на 2 человека) и избыток от второй группы, равный 402 грамма.

Из этого следует, что 6 туристов будут переносить вес продуктов, равный 7149 грамм, а пожилая женщина вес, равный 1210 грамм.

В этом примере модель обработки дней меню с конца при распределении также обеспечивает равномерную разгрузку туристов на привалах, поскольку туристы с малым значением максимальной нагрузки, наподобие описанной ранее пожилой женщины, будут полностью загружаться продуктами, необходимыми для последних дней, и будут нести небольшие массы, но до самого конца. В отличие от туристов с большой максимальной нагрузкой, которые будут нести продукты на каждый день, но и разгружаться будут ежедневно.

В итоге были разобраны примеры, отображающие основные модели поведения алгоритма распределения продуктов в различных ситуациях.

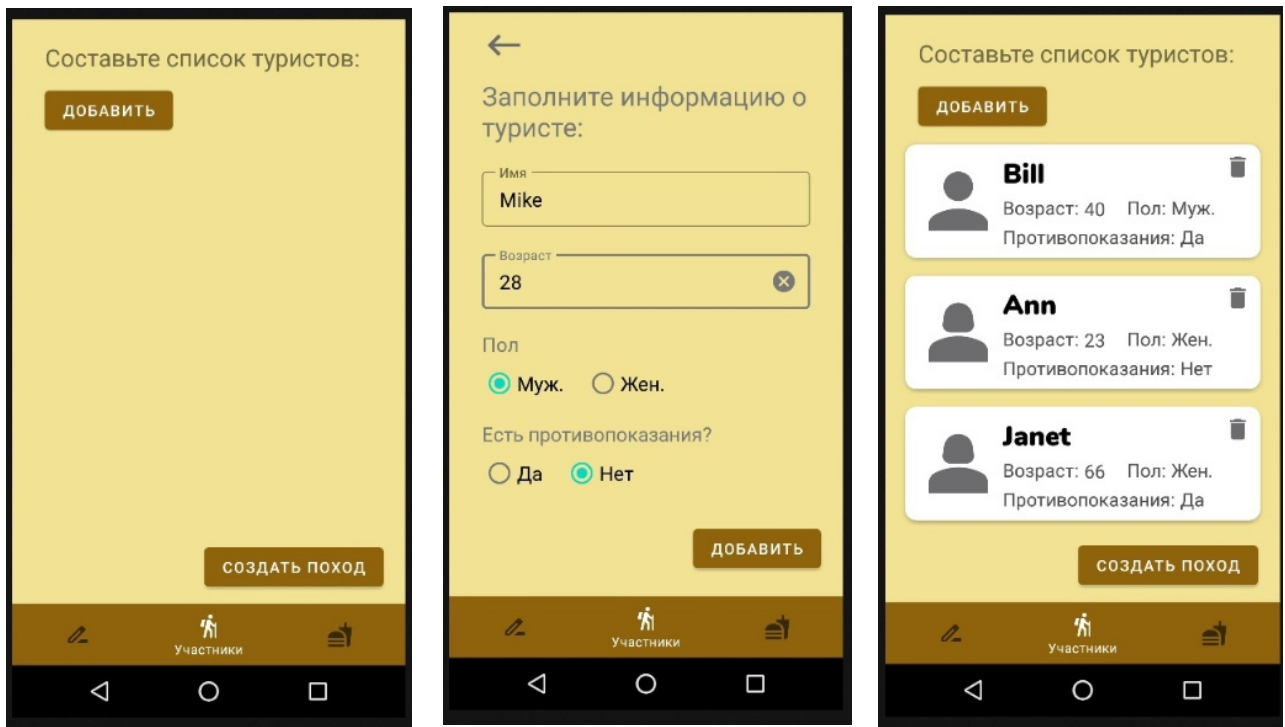
4.4 Примеры работы приложения

На рисунке 22 изображен процесс создания записи похода в приложении.



Рисунок 22 – Создание записи похода

Рисунок 23 отражает функционал по составлению списка туристов.



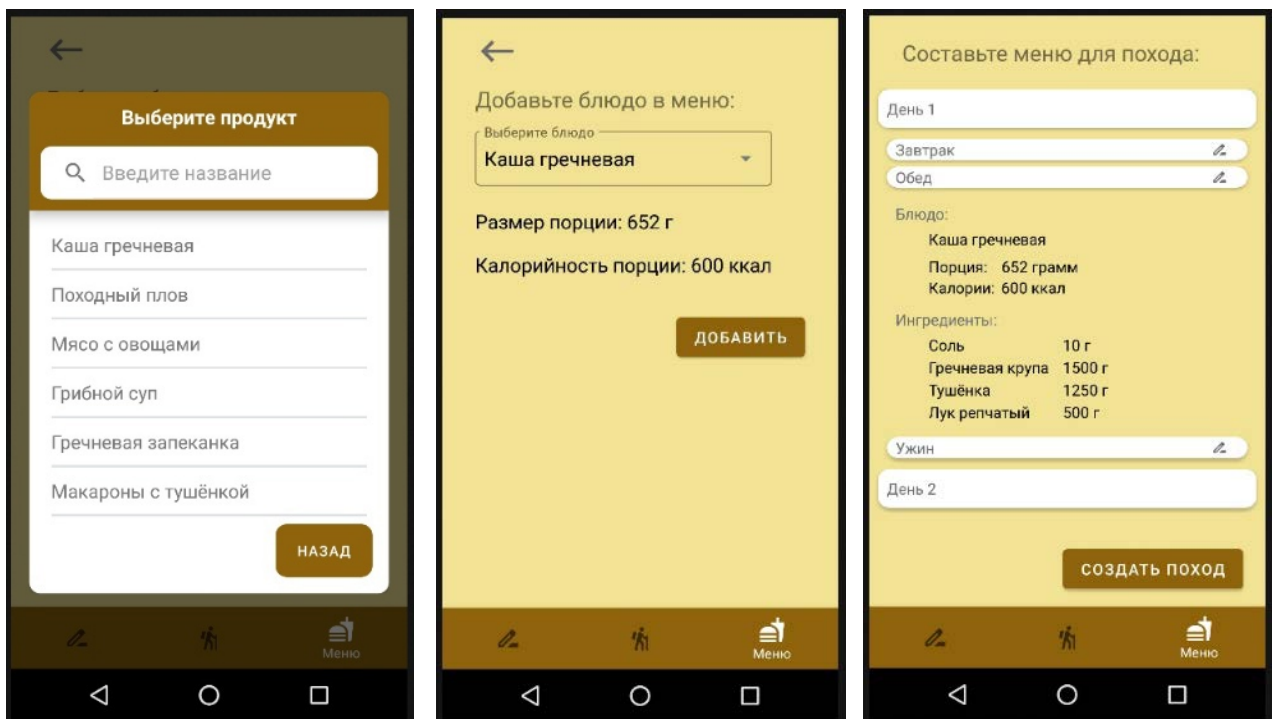
а)

б)

в)

Рисунок 23 – Создание списка участников похода

Функционал для составления меню похода представлен на рисунке 24.



а)

б)

в)

Рисунок 24 – Составление меню похода

Отображение созданной записи похода представлено на рисунке 25.

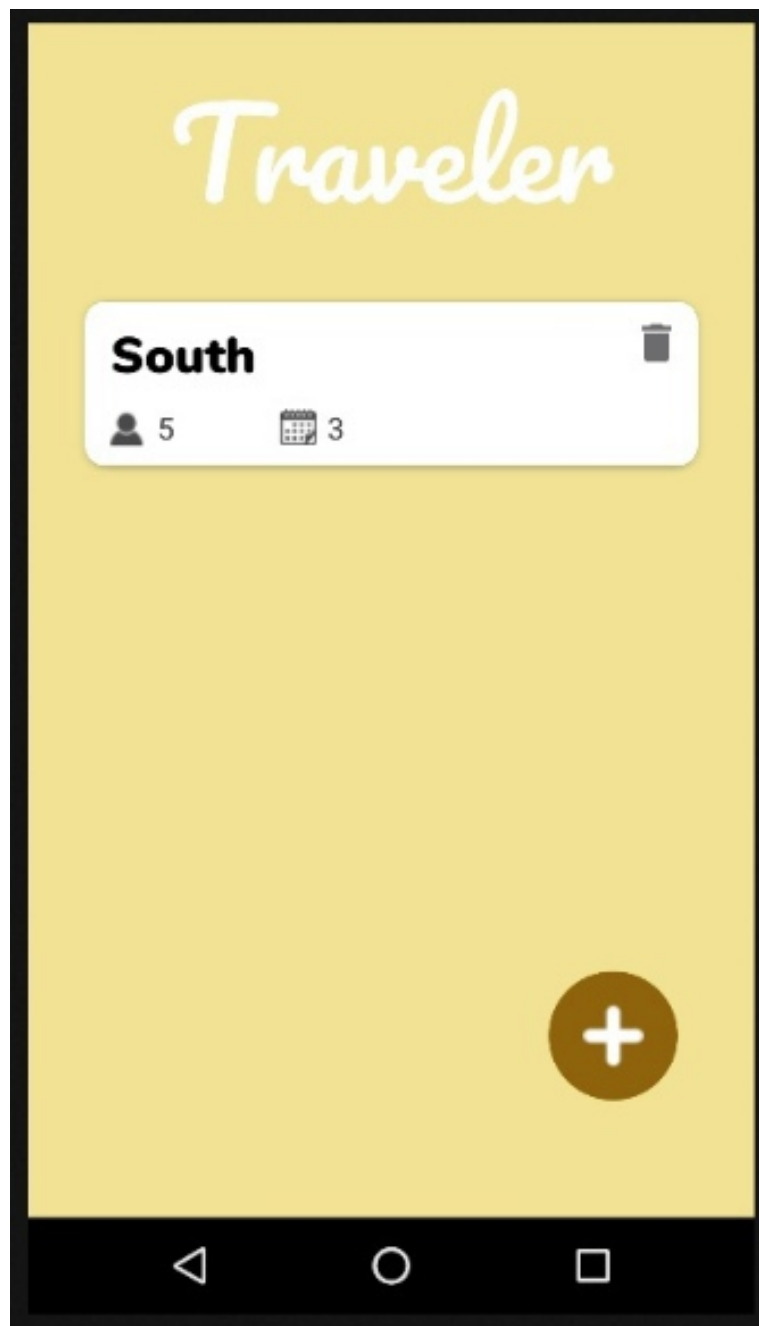
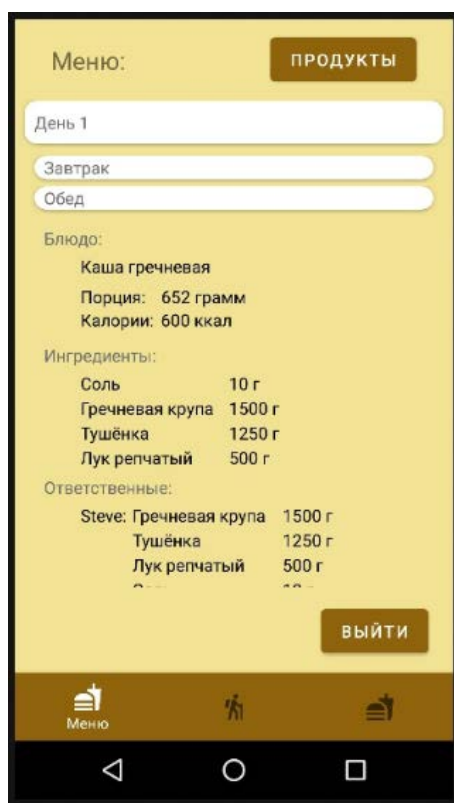


Рисунок 25 – Список созданных записей походов

На рисунках 26, 27, 28 и 29 изображен процесс просмотра созданной записи похода и получение пользователем необходимой информации



а)



б)

Рисунок 26 – Просмотр меню с информацией о распределённых продуктах



Рисунок 27 – Просмотр списка продуктов, взятых в поход

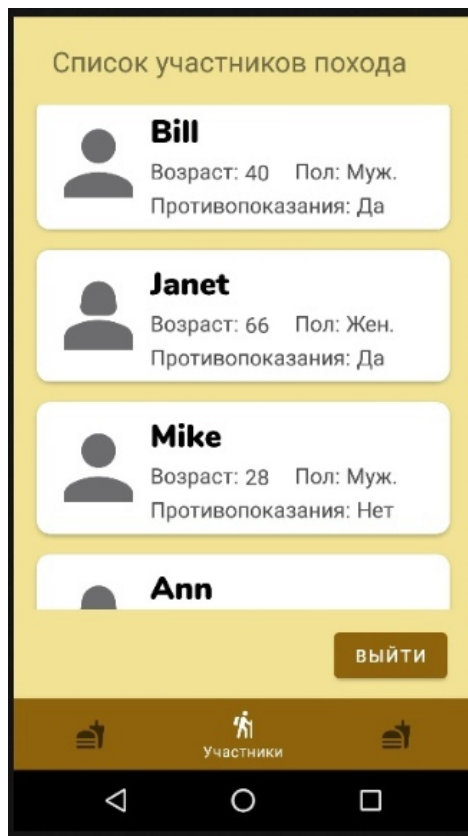


Рисунок 28 – Просмотр списка туристов

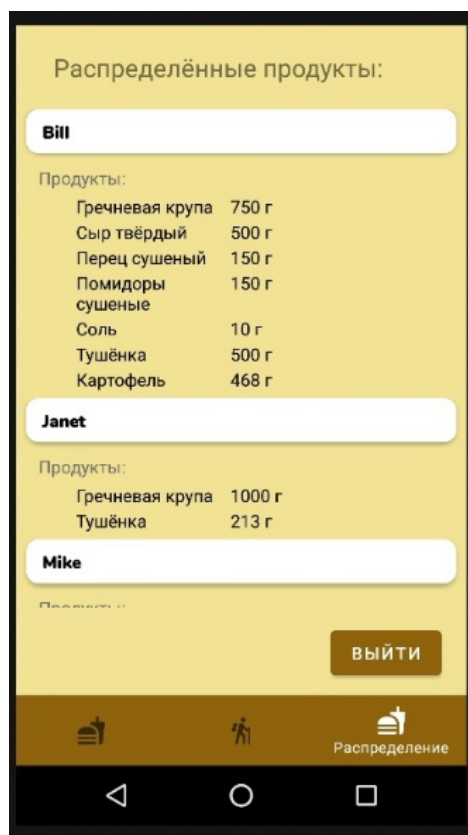


Рисунок 29 – Просмотр распределения продуктов между туристами

ЗАКЛЮЧЕНИЕ

Таким образом, в ходе выполнения выпускной квалификационной работы было разработано мобильное приложение для организации питания в туристическом походе.

Был проведён обзор аналогов разрабатываемого приложения, который позволил получить представление о предметной области и существующих разработках в области туризма. Также были определены требования к разработке.

Был выполнен обзор средств реализации, в ходе которого были обозначены языки программирования, среды разработки и системы управления базами данных. С помощью обзора удалось подобрать наиболее подходящие средства реализации, что позволило повысить эффективность разработки.

Было разработано программное обеспечение. В ходе разработки был реализован функционал, удовлетворяющий требованиям, сформированным в обзоре аналогов.

Было проведено тестирование программного обеспечения, которое позволило определить и устранить критические неисправности в работе приложения.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1 «Meet Android Studio» [Электронный ресурс]. – URL: <https://developer.android.com/studio/intro> (Дата обращения: 16.03.2023).
- 2 «Плюсы и минусы разработки на Xamarin» [Электронный ресурс]. – URL: <https://habr.com/ru/post/343098/> (Дата обращения: 16.03.2023).
- 3 «Kotlin vs Java. В чем отличие и что лучше?» [Электронный ресурс]. – URL: <https://blog.mo-apps.com/razrabotka/kotlin-vs-java-v-chem-otlichie-i-cto-luchshe> (Дата обращения: 16.03.2023).
- 4 «Битва титанов: Java vs Kotlin» [Электронный ресурс]. – URL: <https://tproger.ru/articles/bitva-titanov-java-vs-kotlin/> (Дата обращения: 16.03.2023)
- 5 «Системы управления базами данных» [Электронный ресурс]. – URL: <http://akond0fswat.de/ration> (Дата обращения: 17.03.2023).
- 6 «Введение в SQLite» [Электронный ресурс]. – URL: <https://metanit.com/sql/sqlite/1.1.php> (Дата обращения: 17.03.2023)