

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

КУРСОВАЯ РАБОТА
по дисциплине «Разработка приложений для мобильных платформ»
Тема: Погодно-гардеробный советчик

Студент гр. 5303

Басин Д.Д.

Студент гр. 5303

Губа Д.А.

Студент гр. 5303

Кадыров Р.Р.

Преподаватель

Заславский М.М.

Санкт-Петербург

2019

ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ

Студенты Басин Д.Д., Губа Д.А., Кадыров Р.Р.

Группа 5303

Тема работы: Погодно-гардеробный советчик

Исходные данные:

Android, Kotlin

Содержание пояснительной записки: Содержание, Введение, Сценарии использования, Пользовательский интерфейс, Модель данных, Разработанное приложение, Последовательность действий для осуществления сценариев использования, Заключение, Список литературы, Приложения.

Предполагаемый объем пояснительной записки:

Не менее 20 страниц.

Дата выдачи задания: 14.02.2019

Дата сдачи реферата: 18.04.2019

Дата защиты реферата: 18.04.2019

Студент гр. 5303

Басин Д.Д.

Студент гр. 5303

Губа Д.А.

Студент гр. 5303

Кадыров Р.Р.

Преподаватель

Заславский М.М.

АННОТАЦИЯ

В курсовой работе реализована игра на базе Android, в которой необходимо определять простое или составное число на экране. После анализа сложности пользовательского интерфейса были предложены возможные улучшения для уменьшения сложности. Также были разработаны материалы для публикации приложения в Play Market.

SUMMARY

In the course work implemented a game on the basis of Android, which is necessary to define the prime or composite number on the screen. After the analysis of complexity of the user interface possible improvements for reduction of complexity were offered. Also materials for the publication applications in Play Market were developed.

Содержание

Введение.....	6
1. Сценарии использования	7
1.1. УС-1. Главное окно приложения.....	7
Основной поток:	7
Альтернативный поток:	7
1.2. УС-2. Выбор местоположения.....	7
Предусловие:	7
Основной поток:	7
Альтернативный поток - 1:	8
Альтернативный поток - 2:	8
1.3. УС-3. Меню.	8
Предусловие:	8
Основной поток:	8
Альтернативный поток:	8
1.4. УС-4. Гардероб.....	8
Предусловие:	8
Основной поток:	9
Альтернативный поток - 1:	9
Альтернативный поток - 2:	9
1.5. УС-5. Новый объект в гардеробе.	9
Предусловие:	9
Основной поток:	9
Альтернативный поток - 1:	10
Альтернативный поток - 2:	10
1.6. УС-6. Настройки.	Error! Bookmark not defined.
Предусловие:	Error! Bookmark not defined.
Основной поток:	Error! Bookmark not defined.
Альтернативный поток - 1:	Error! Bookmark not defined.
Альтернативный поток - 2:	Error! Bookmark not defined.
2. Пользовательский интерфейс.....	11
2.1. Макет интерфейса с графом переходов.....	11
2.2. Целевые устройства, обоснования требования и максимально подробные характеристики	11
3. Модели данных.....	12
3.1. Общая характеристика хранимых данных	12

3.2.	Описание модели данных объектов гардероба	12
4.	Разработанное приложение	13
4.1.	Краткое описание.....	13
4.2.	Схема архитектуры	13
4.3.	Использованные технологии (внешние).....	13
4.4.	Использованные модули/системные библиотеки вашей платформы.....	13
4.5.	Стратегии для обеспечения кросс-платформенности приложения	13
5.	Последовательность действий для осуществления сценариев использования	14
5.1.	Измерение последовательности действий для осуществления сценариев использования	14
5.2.	Варианты улучшения пользовательского интерфейса.....	14
6.	Аналоги приложения.....	15
	Заключение.....	17
	Список использованных источников.....	18
	Приложение А. Макет приложения	19
	Приложение Б. Документация по сборке и развертыванию приложения	20
	Приложение В. Снимки экрана приложения	21

ВВЕДЕНИЕ

Развитие математических способностей важно в жизни людей, потому что они влияют на функции памяти, а также на функции мышления. Поэтому развивая математические способности, развиваются ещё и функции памяти и мышления.

Развивать математические способности можно разными способами, например, решать алгебраические, геометрические и логические задачи, но для этого чаще всего необходимы письменные принадлежности и стол.

Более простой способ развития способностей - это определение простое или составное число. Для усложнения задания добавим дополнительную задачу для составных чисел: найти простые делители этого числа.

Данное решение необходимо реализовать как мобильное приложение, потому что у людей выдаётся свободное время, когда они едут в метро/автобусе, стоят в пробках, ждут очереди в больнице, и они могут его потратить на развитие своих математических способностей в этих местах.

1. СЦЕНАРИИ ИСПОЛЬЗОВАНИЯ

1.1. UC-1. Главное окно приложения.

Основной поток:

1. При открытии приложения пользователь попадает на главный экран приложения, на котором он может выбрать свое местоположение, кликнув на иконку в верхней левой части экрана. (UC-2)
2. После выбора местоположения пользователь может посмотреть погоду на сегодня, а также узнать выбор одежды от приложения.
3. Пользователь, после ознакомления с одеждой, может выбрать любой из видов одежды (Голова, Куртка, Верх, Низ, Обувь) и с помощью стрелок справа и слева менять свою одежду на доступные.

Альтернативный поток:

1. Пользователь имеет возможность открыть “меню” иконкой, располагающейся в правом верхнем углу. (UC-3)

1.2. UC-2. Выбор местоположения.

Предусловие:

1. Пользователь находится на экране выбора своего местоположения.

Основной поток:

1. Пользователь начинает вводить название города.
2. Мобильное приложение предлагает ему варианты, совпадающие с началом введенного пользователем текста.
3. Пользователь выбирает нужный ему вариант.
4. Мобильное приложение возвращается на главный экран. (UC-1)

Альтернативный поток - 1:

1. Пользователь ничего не выбирает, а нажимает на иконку “назад”.
2. Мобильное приложение возвращается на главный экран. (UC-1)

Альтернативный поток - 2:

1. Пользователь имеет возможность открыть “меню” иконкой, располагающейся в правом верхнем углу. (UC-3)

1.3. UC-3. Меню.

Предусловие:

1. Пользователь нажал на иконку “меню” на главном экране.

Основной поток:

2. Мобильное приложение предлагает ему select box, на котором есть два варианта (Гардероб, Настройки).
- Пользователь выбирает вариант “Гардероб”. (UC-4)
 - Пользователь выбирает вариант “Настройки”. (UC-6)

Альтернативный поток:

1. Пользователь ничего не выбирает.
2. Система возвращается на экран, с которого было вызвано “Меню”.

1.4. UC-4. Гардероб.

Предусловие:

1. Пользователь выбрал вариант “Гардероб” из меню приложения.

Основной поток:

1. Пользователь видит созданные им ранее варианты одежды.
2. Пользователь может выбрать, какие варианты одежды, необходимо использовать мобильному приложению при выборе вариантов одежды, поставив или сняв галочку с check box.
3. Пользователь может отредактировать вещь, нажав на иконку справа от объекта. (UC-5).
4. Пользователь может создать новую вещь, нажав на иконку “плюс” внизу экрана. (UC-5).

Альтернативный поток - 1:

1. Пользователь ничего не выбирает, а нажимает на иконку “назад”.
2. Мобильное приложение возвращается на главный экран. (UC-1)

Альтернативный поток - 2:

1. Пользователь имеет возможность открыть “меню” иконкой, располагающейся в правом верхнем углу. (UC-3)

1.5. UC-5. Новый объект в гардеробе.

Предусловие:

1. Пользователь нажал на иконку добавления, либо редактирования объекта в UC-4.

Основной поток:

1. Пользователь пишет название нового объекта.
2. Пользователь выбирает цвет объекта из предустановленных вариантов.

3. Пользователь выбирает тип объекта из select-box'a. (Голова, Куртка, Верх, Низ, Обувь).
4. В зависимости от выбранного варианта в 3 пункте, мобильное приложение выдает нужные типу иконки.
5. Пользователь выбирает нужную иконку.
6. Пользователь вводит температуру, при которой необходимо использовать данный объект.
7. Пользователь нажимает “сохранить” и возвращается в “Гардероб”. (UC-4).

Альтернативный поток - 1:

1. Пользователь ничего не выбирает, а нажимает на иконку “назад”.
2. Мобильное приложение возвращается в "Гардероб". (UC-1)

Альтернативный поток - 2:

3. Пользователь имеет возможность открыть “меню” иконкой, располагающейся в правом верхнем углу. (UC-3)

1. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС

2.1. Макет интерфейса с графом переходов

Макет интерфейса с графом переходов представлен в приложении А. На макете изображены основные экраны и возможные переходы между ними.

2.2. Целевые устройства, обоснования требования и максимально подробные характеристики

Тип устройств: планшеты, смартфоны.

Аппаратная составляющая: все устройства на базе операционной системы Android версии 5.0 и выше.

3. МОДЕЛИ ДАННЫХ

3.1. Общая характеристика хранимых данных

Для пользования приложением необходимо хранить объекты гардероба пользователя между запусками приложения. Для решения задачи хранения объектов гардероба используется мобильная база данных.

Также для использования приложения в офф-лайн режиме сохраняются в кэш данные о последнем местоположении пользователя и последние данные по погоде.

3.2. Описание модели данных объектов гардероба

Объекты гардероба, хранимые в базе, содержат в себе информацию о типе объекта, пользовательской конфигурации объекта (цвет иконки и название), информацию о температурных режимах.

На основе типа объекта гардероба и температурного диапазона алгоритм подбирает все подходящие по погодным условиям вещи. При этом каждый новый запрос собирает комбинацию объектов разного типа в случайном порядке, что позволяет вносить разнообразие пользователю.

4. РАЗРАБОТАННОЕ ПРИЛОЖЕНИЕ

4.1. Краткое описание

Разработанное приложение реализует описанные выше сценарии использования, а также сохраняет результаты игры на устройстве.

В приложении Б представлена документация по сборке и развертыванию приложения.

В приложении В представлена инструкция пользователя.

В приложении Г представлены снимки экрана.

4.2. Схема архитектуры

Общая схема архитектуры полностью отображает паттерн MVC.

4.3. Используемые технологии (внешние)

Реализация не использует внешних библиотек.

4.4. Используемые модули/системные библиотеки вашей платформы

Системные библиотеки и модули нашей платформы:

- `com.android.tools.build:gradle:3.3.0;`
- `org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlin_version.`

4.5. Стратегии для обеспечения кросс-платформенности приложения

Реализация была выполнена на языке Kotlin, которая запускается при помощи JVM, что и обеспечивает кросс-платформенность приложения.

5. ПОСЛЕДОВАТЕЛЬНОСТЬ ДЕЙСТВИЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ СЦЕНАРИЕВ ИСПОЛЬЗОВАНИЯ

5.1. Измерение последовательности действий для осуществления сценариев использования

В табл. 1 представлена оценка сложности пользовательского интерфейса для всех сценариев использования.

Таблица 1 – Оценка сложности пользовательского интерфейса приложения

	Клик	Свайп	Ввод текста	Скролл	Выпадающий список
Выбор местоположения	2	0	1	0	0
Обновление данных о погоде	0	1	0	0	0
Просмотр гардероба	1	0	0	0(5)	0
Добавление (редактирование) объекта гардероба	3	0	1	0	3(0)

Весь функционал приложения осуществляется самым простым действием для пользователя – кликом.

5.2. Варианты улучшения пользовательского интерфейса

Пользовательский интерфейс интуитивно понятен и прост. Направлением к улучшению можно рассматривать сокращение выполняемых сценариев при первом запуске за счет автоматического определения местоположения и предустановки стандартного набора объектов гардероба.

6. АНАЛОГИ ПРИЛОЖЕНИЯ

В Play market существует несколько аналогов:

1. Weatherproof - Weather & Clothes

(<https://play.google.com/store/apps/details?id=com.changemystyle.weatherproof>);

2. Weather Clothes

(<https://play.google.com/store/apps/details?id=com.diapasondev.weatherclothes>);

3. Clothing and Weather

(<https://play.google.com/store/apps/details?id=devcom.clothingandweather>);

4. What 2 Wear?

(<https://play.google.com/store/apps/details?id=com.iloiacomo.what2wear>).

Аналоги 1-3 не позволяют редактировать элементы гардероба, что существенно снижает гибкость приложения и не позволяет сравнивать их с нашей разработкой из-за отсутствия большей части функционала. Аналог 4 имеет максимально близкий функционал. При редактировании вещей в аналоге есть возможность добавления фотографии вещи, однако не позволяет настраивать температурный режим для вещей. (У нас добавить вещь в приложении-аналоге не удалось).

В табл. 2 представлена оценка сложности пользовательского интерфейса для аналогичных сценариев использования в приложении-аналоге 4.

Таблица 2 – Оценка сложности пользовательского интерфейса приложения-аналога

	Клик	Свайп	Ввод текста	Выбор/Создание фото	Выпадающий список
Выбор местоположения	4	0	1	0	0
Обновление данных о погоде	1	0	0	0	0

	Клик	Свайп	Ввод текста	Выбор/Создание фото	Выпадающий список
Просмотр гардероба	0	1	0	0	0
Добавление (редактирование) объекта гардероба	1	0	0	1	2

ЗАКЛЮЧЕНИЕ

Результатом выполнения курсовой работы является разработанное приложения для улучшения математических способностей.

В процессе выполнения были написаны сценарии использования и сделан макет интерфейса, которые в итоге были реализованы в приложении с минимальными изменениями.

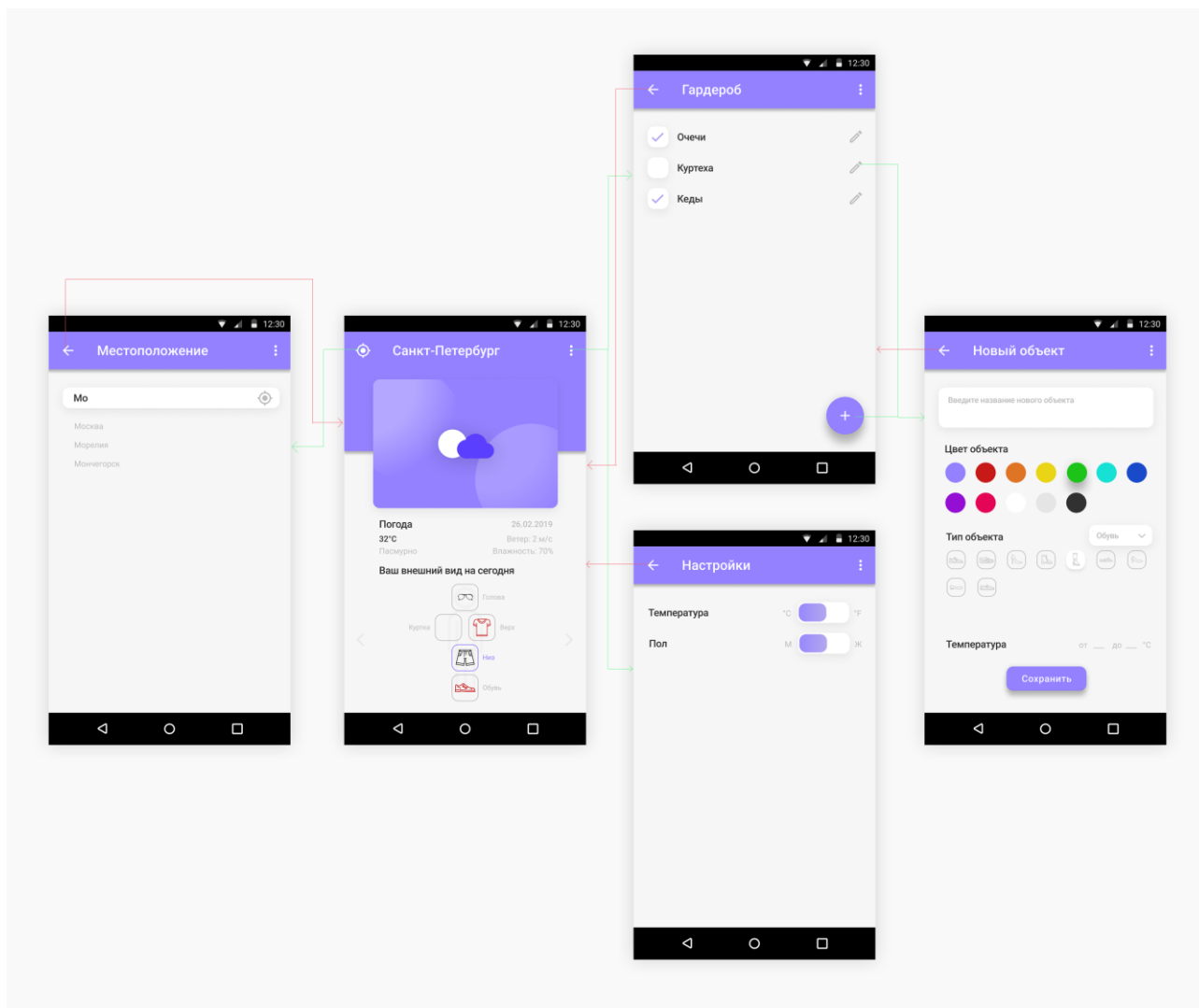
Данная реализация не имеет явных недостатков, но предполагается, что система будет развиваться: добавляться новые режимы и совершенствоваться способы взаимодействия пользователя с приложением.

На данном этапе планируется получить обратную связь от пользователя, учесть все замечания и пожелания и доработать соответствующим образом приложение. Кроме того возможен вариант с добавлением гибкой системы генерации уровней и оценивания результатов пользователя с использованием технологии нейронных сетей.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Макконнелл С. Совершенный код. Мастер-класс; Русская редакция. – 2017.
2. Гамма Э. и др. Приемы объектно-ориентированного проектирования. – "Издательский дом Питер", 2013.
3. Documenting Kotlin Code URL: <https://kotlinlang.org/docs/reference/kotlin-doc.html> (дата обращения: 24.03.2019).
4. Исходный код проекта URL: <https://github.com/moevm/adfmp19-weather-advisor>

ПРИЛОЖЕНИЕ А. МАКЕТ ПРИЛОЖЕНИЯ



ПРИЛОЖЕНИЕ Б. ДОКУМЕНТАЦИЯ ПО СБОРКЕ И РАЗВЕРТЫВАНИЮ ПРИЛОЖЕНИЯ

1. Должен быть установлен gradle 3.3.
2. После этого в терминале или командной строке необходимо ввести следующую команду «gradlew clean build run».

ПРИЛОЖЕНИЕ В. СНИМКИ ЭКРАНА ПРИЛОЖЕНИЯ

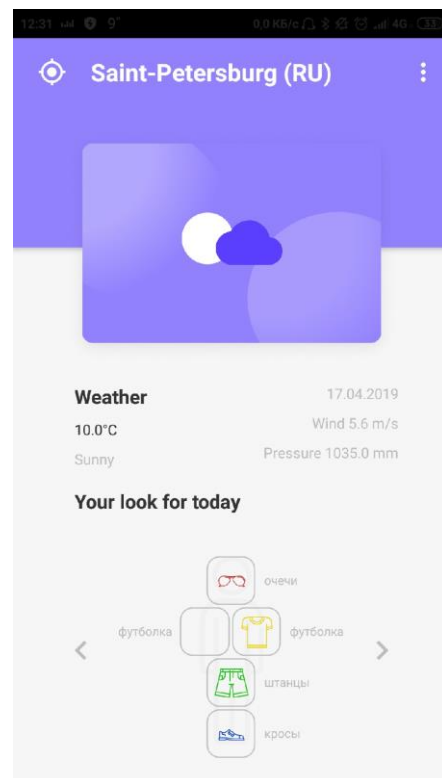
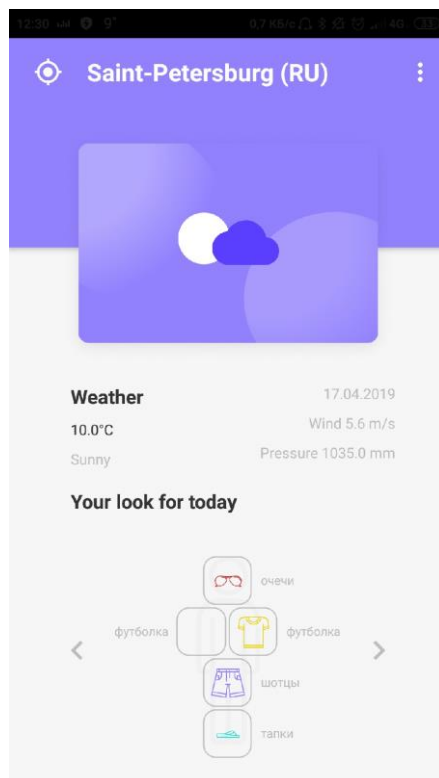
Создание объекта гардероба.

The screenshot shows the 'New Cloth' form. At the top is a purple header with a back arrow and the text 'New Cloth'. Below the header is a text input field containing the word 'вещица'. Underneath is the 'Item color' section with two rows of colored circles: the first row has purple, red, orange, yellow, green, cyan, and blue; the second row has magenta, pink, white, light gray, and black. Below the color selection is the 'Item type' section, which includes a 'Feet' dropdown menu and a grid of shoe icons. At the bottom of the form is the 'Temperature' section with 'From' and 'To' fields, containing the values '-15' and '5' respectively. A purple 'Save' button is located at the very bottom.

Гардероб (список доступных вещей)

The screenshot shows the 'Wardrobe' screen. It has a purple header with a back arrow and the text 'Wardrobe'. Below the header is a list of items, each with a checkmark icon on the left, the item name in the center, and an edit icon (pencil) on the right. The items listed are: 'тишка', 'футболка', 'шотцы', 'штанцы', 'тапки', 'кросы', and 'вещица'. At the bottom right of the screen is a large purple circular button with a white plus sign.

Главный экран



Выбор местоположения

