

Санкт-Петербургский государственный университет

Математическое обеспечение и администрирование информационных систем

Фомина Виктория Викторовна

Кулинарное Android-приложение

нижняя навигация, поиск ингредиентов, добавление рецептов в избранное

Отчет по учебной практике

Научный руководитель:
к. т. н., доцент Литвинов Ю.В

Санкт-Петербург
2019

Содержание

1	Введение	2
2	Задачи команды	2
3	Обзор существующих решений	3
3.1	Приложение «Что готовим»	3
3.2	Приложение «Подбери рецепт»	3
4	Задачи автора	4
5	Общий обзор решения	4
6	Архитектура решения	4
7	Обзор используемых решений	5
8	Реализация	6
8.1	Меню «Нижняя навигация»	6
8.2	Фрагмент «Добавление ингредиентов»	6
8.3	Фрагмент «Избранное»	7
9	Общие итоги	8
10	Апробация	9
11	Итоги работы, проделанной автором	9
12	Ссылки на источники	11

1 Введение

В повседневной жизни практически каждый человек тратит довольно много времени на приготовление пищи. Для поиска новых рецептов люди часто пользуются кулинарными сайтами или соответствующими мобильными приложениями. Искать рецепт исключительно по названию не всегда удобно. Могут возникнуть трудности, связанные с нехваткой некоторых ингредиентов. Приложение, которое могло бы формировать список подходящих рецептов по ингредиентам, избежало бы этих трудностей. В Play Market существует несколько русскоязычных приложений, обладающих такой функциональностью, но, к сожалению, эти приложения обладают рядом недостатков: интерфейс, который слишком сложен для пользователя; отсутствие информации к рецепту о количестве порций.

Целью команды стало разработать приложение, ключевая функциональность которого заключалась бы в поиске рецептов по ингредиентам. Интерфейс приложения необходимо было сделать как можно более простым и понятным; к каждому рецепту необходимо было добавить количество порций.

2 Задачи команды

- Реализовать приложение, поддерживающее поиск рецептов по ингредиентам.
- Добавить в приложение поиск рецептов по названию.
- Добавить в приложение различные фильтры для поиска рецептов (выбор кухни, типа блюда, времени приготовления).
- Добавить раздел «избранные» рецепты.
- Заполнить базу рецептами с информацией о количестве порций.
- Сделать интерфейс приложения интуитивно-понятным.

3 Обзор существующих решений

3.1 Приложение «Что готовим»

«Что готовим» [1] обладает такой функциональностью как поиск рецептов по ингредиентам. Приложение позволяет осуществить поиск рецептов по названию, добавить свой рецепт, добавить продукты в черный список, дает возможность добавить рецепт в избранное. Несмотря на широкий спектр возможностей для пользователя, оно обладает существенным недостатком: сложный, интуитивно непонятный интерфейс (подтверждено большим количеством негативных отзывов в Play Market).

3.2 Приложение «Подбери рецепт»

Приложение «Подбери рецепт» [2] предоставляет пользователю большое количество возможностей. Оно не уступает по функциональности приложению «Что готовим». Помимо этого, оно содержит таблицу мер и весов. Однако и это приложение сложно назвать совершенным. Например, в рецептах этого приложения отсутствует информация о количестве порций. Часто это может стать причиной того, что будет приготовлено слишком мало или слишком много пищи.

4 Задачи автора

- Реализовать интуитивно-понятный интерфейс.
- Реализовать добавление ингредиентов.
- Реализовать добавление рецепта в избранное.

5 Общий обзор решения

Было разработано приложение, основная функциональность которого заключается в поиске рецептов по ингредиентам. Поиск рецептов по названию, а также расширенный поиск с применением таких фильтров как выбор кухни, типа блюда и времени приготовления приложение поддерживает тоже. Приложение предоставляет возможность добавлять наиболее понравившиеся рецепты в избранное.

В отчете будет рассказано о тех деталях, которые были реализованы непосредственно автором.

6 Архитектура решения

Основными составляющими решения являются:

1. База данных и класс DBProvider, представляющий собой обёртку над базой данных. С помощью класса DBProvider были реализованы поиск рецептов по ингредиентам, названию, добавление рецептов в избранное.
2. Фрагменты, отвечающие за взаимодействие с пользователем. Приложение можно представлять множеством фрагментов, где каждый фрагмент отвечает за отдельное окно.

7 Обзор используемых решений

- Для представления каждого из окон приложения использовался фрагмент [3] – элемент Android API. Фрагмент отвечает за отображение пользовательского интерфейса и обработку пользовательских действий. В коде это реализуется так: класс MyFragment наследуется от стандартного в Android класса Fragment. Отображение макета с пользовательским интерфейсом происходит при создании фрагмента в переопределенном методе onCreate(). Перемещение между фрагментами осуществляется посредством экземпляра класса FragmentManager [5]. Это стандартный класс в Android, который используется для навигации между фрагментами.
- Для отображения больших списков использовался класс RecyclerView [4] – элемент библиотеки v7 Support Library. RecyclerView создает представления для элементов и заменяет содержимое некоторых представлений новыми данными, когда исходный элемент перестает отображаться. Для привязки набора данных к представлениям, отображаемым RecyclerView, был использован наследник класса RecyclerView.Adapter, который переопределял три метода:
 - onCreateViewHolder (возвращает объект ViewHolder, который хранит данные об одном элементе RecyclerView);
 - onBindViewHolder (заменяет содержимое представлений, больше не отображаемых на экране);
 - getItemCount (возвращает длину набора данных).

8 Реализация

8.1 Меню «Нижняя навигация»

Меню «нижняя навигация» (Рис. 1) представляет собой панель из нескольких кнопок в нижней части экрана. «Нижняя навигация» позволяет сделать достаточно простым перемещение между основными окнами: лента рецептов, поиск рецептов, избранное и настройки.

Например, при нажатии на кнопку «избранное» происходит переход из фрагмента, в котором находился пользователь, во фрагмент «избранное».

В каждый момент времени выбрана ровно одна кнопка меню. Выбранная кнопка оснащена световой индикацией и подписью («лента», «настройки» и т.д.).

«Нижняя навигация» – это объект стандартного в Android класса `BottomNavigationView`. Макет объекта содержит в себе описание иконок меню. В приложении использованы стандартные иконки Android Studio.



Рис. 1: Панель кнопок «нижняя навигация»

8.2 Фрагмент «Добавление ингредиентов»

Макет фрагмента «Добавление ингредиентов» содержит в себе строку поиска и кнопку, нажатие на которую возвращает в предыдущий фрагмент: фрагмент поиска рецепта.

Ввод названия ингредиента в строку поиска сопровождается подсказками. К строке поиска прикреплен обработчик событий, который следит за каждым изменением текста. Обработчик обращается к методу `getIngredients(String)` класса `DBProvider`, который возвращает список ингредиентов, содержащих строку, введенную пользователем

в качестве подстроки. «Подходящие» ингредиенты добавляются в RecyclerView и отображаются на экране в виде элементов, состоящих из названия ингредиента и кнопки. Если ингредиент выбран пользователем (то есть произошло однократное нажатие на него), то на кнопке появляется «галочка» – индикатор того, что ингредиент выбран. Повторное нажатие убирает «галочку».

Поддерживается множественный выбор ингредиентов (Рис. 2).

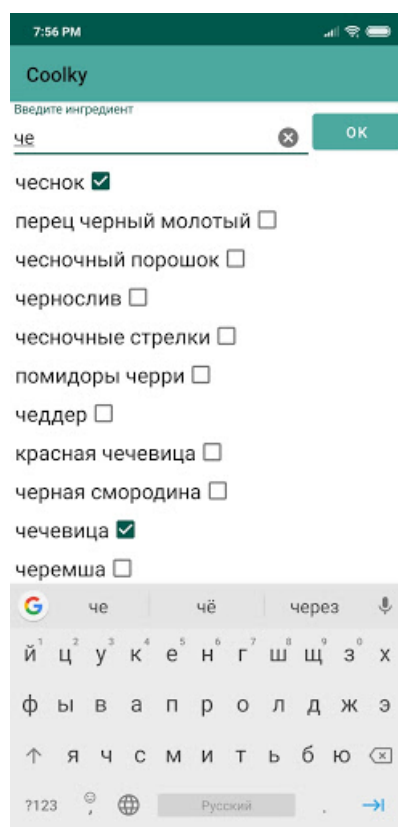


Рис. 2: Графический интерфейс фрагмента «поиск ингредиента»

8.3 Фрагмент «Избранное»

В приложении была реализована возможность добавления наиболее понравившихся рецептов в избранное (Рис. 3).

Пользователь заполняет различные фильтры (ингредиенты, кухня,

тип блюда и время приготовления), нажимает на кнопку «поиск». На экране отображается лента с результатами поиска – список с краткой информацией о рецептах.

Элемент с краткой информацией о рецепте содержит кнопку «звездочка», уведомляющую о том, добавлен ли рецепт в избранное. Реализовано это следующим образом: переход во фрагмент отображения результатов поиска сопровождается вызовом метода `isFavourite(recipeId)` класса `DBProvider` для каждого из отображаемых элементов с кратким описанием рецепта. Если рецепт является «избранным», отображаем кнопку «звездочка» в желтом цвете.

Добавить рецепт в избранное можно и при просмотре полной информации о нем.

Меню «нижняя навигация» содержит кнопку, нажатие на которую позволяет открыть фрагмент «избранное». При открытии этого фрагмента вызывается метод `getFavourites()` класса `DBProvider`. Этот метод возвращает список объектов класса `Favourite`. Объект класса `Favourite` содержит свойство `recipeId`. Id рецепта позволяет легко удалить рецепт из избранного или добавить в избранное, а также отобразить данные рецепта в разделе «избранное».

9 Общие итоги

Было разработано кулинарное приложение, которое по входным данным – ингредиентам формирует выходные – список подходящих пользователю рецептов. Для поиска рецептов по ингредиентам можно применять такие фильтры как выбор кухни, типа блюда, времени приготовления. Интерфейс приложения интуитивно понятен, не создает никаких трудностей для взаимодействия с пользователем. Поддерживается поиск рецептов по названию. База приложения заполнена рецептами. Каждый из рецептов содержит информацию о количестве порций.

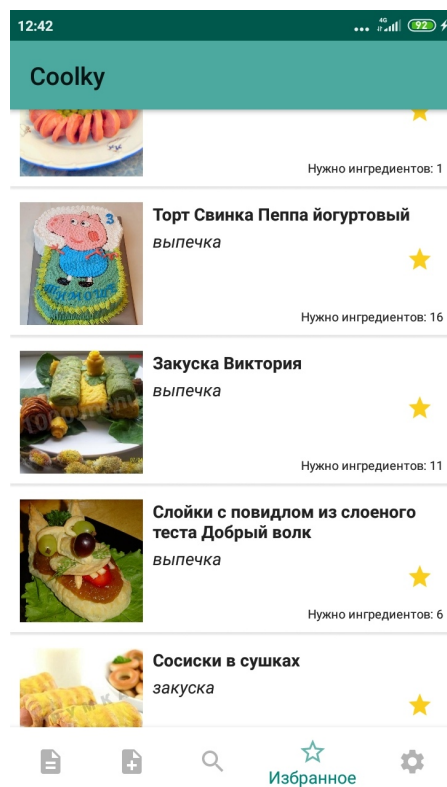


Рис. 3: Графический интерфейс фрагмента «избранное»

10 Апробация

Командой была проведена апробация приложения на нескольких смартфонах с различными версиями ОС Android. Кулинарное приложение работало корректно на всех тестовых сценариях.

Апробация проходила на небольшом количестве участников. В целом были получены положительные отзывы. Участники апробация в большинстве своем нашли приложение полезным в повседневной жизни.

11 Итоги работы, проделанной автором

- Реализовано меню «нижняя навигация».

- Реализовано добавление ингредиента.
- Реализовано добавление рецепта в избранное.
- Проведена апробация.

12 Ссылки на источники

- [1] GooglePlay: Что готовим? – Режим доступа: <https://play.google.com/store/apps/details?id=ru.gamespace.myfridgehl=ru> (дата обращения: 25.12.2019)
- [2] GooglePlay: Подбери рецепт. – Режим доступа: <https://play.google.com/store/apps/details?id=com.ggl.jr.cookbooksearchbyingredientshl=ru> (дата обращения: 25.12.2019)
- [3] Fragments. – Режим доступа: <https://developer.android.com/guide/components/fragment> (дата обращения: 25.12.2019)
- [4] RecyclerView. – Режим доступа: <https://developer.android.com/reference/androidx/recyclerview/widget/RecyclerView.html> (дата обращения: 25.12.2019)
- [5] FragmentManager. – Режим доступа: <https://developer.android.com/reference/android/app/FragmentManager> (дата обращения: 25.12.2019)
- [6] FragmentTransaction. – Режим доступа: <https://developer.android.com/reference/android/app/FragmentTransaction.html> (дата обращения: 25.12.2019)