**­ Министерство образования и науки Российской Федерации**

Федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа информационных технологий и робототехники

Отделение Информационных технологий

Направление Информационные системы и технологии

**ОТЧЕТ**

**по лабораторной работе №5**

**Запросы к сети, многопоточность**

по дисциплине «Программирование мобильных устройств и

встраиваемых систем»

Вариант 4

Выполнил студент гр. 8И6А \_\_\_\_\_\_\_\_\_\_\_ Грачева М.Э.

2020 г.

(Дата сдачи отчета)

Отчет принят:

ассистент ОИТ \_\_\_\_\_\_\_\_\_\_\_\_ Погребной А. В.

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_г.

(дата проверки отчета)

Томск 2020 г

# 1 Задание

Ознакомиться с примером в модуле lab5.

На все фоновые запросы должен отображаться индикатор загрузки (ProgressBar), при ошибке пользователю должна предоставляться возможность переотправить запрос (кнопка «Повтор»). При повороте экрана или переход на другой экран, его состояние не должно теряться, запросы не должны переотправляться.

Реализовать поиск по репозиториям GitHub. В тулбаре расположен SearchView. При вводе строки длиной больше чем 2 символа искать на GitHub репозитории содержащие эти символы и выводить их в списке. Запрос отправлять с задержкой в 500мс (можно использовать метод Handler::postDelayed). Если во время отправки запроса слово для поиска поменялось, прерывать запрос. Каждый элемент списка содержит полное название репозитория, его описание (ограничение 5 строк). Грузить по 20 репозиториев за запрос. При скролле списка вниз, догружать следующие 20 репозиториев. Реализовать обновление списка движением pull-to-refresh (SwipeRefreshLayout).

# 2 Ход работы

На экран добавлены RecyclerView, ProgressBar и Button, причем два последних располагаются по центру родительского контейнера RelativeLayout (Рисунок 1). RecyclerView обернут в SwipeRefreshLayout.

Создан класс Repo, описывающий репозиторий, содержащий поля: fullName и description.

Созданы 3 класса, которые реализуют подписку фонового потока, в котором выполняются запросы к сети на получение данных, на главный поток:

1. Интерфейс Observer – слушатель выполнения задачи. Здесь реализованные 3 метода: onLaoding, onSuccess и onError. Данные методы используются в lab5activity для обмена данными между потоками. Также в данных методах изменяется видимоcть элементов ProgressBar и Button.
2. Абстрактный класс Task – выполняет задачу в фоновом потоке и оповещает главный поток о статусе задачи.
3. Класс SearchTask наследуется от класса Task и реализует методы обращения к сети и получения данных. Метод search (Рисунок 3) выполняет запрос к сети и возвращает тело ответа.

Создание потока, который будет осуществлять запрос к сети, выполняется в методе doSearch с задержкой на 500 мс (Рисунок 3). Данный метод вызывается при вводе символов в строку поиска SearchView (более двух символов) (Рисунок 4), при обновлении списка движением pull-to-refresh, при нажатии на кнопку «повтор», которая появляется при возникновении ошибки, а также при прокрутки списка до конца для добавления следующих 20 элементов в список.

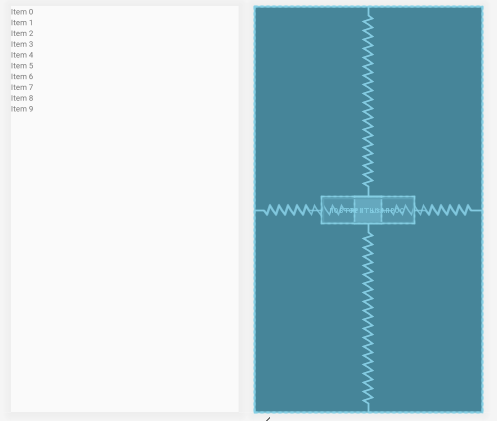


Рисунок 1 – View lab5



Рисунок 2 – Метод search класса SearchTask

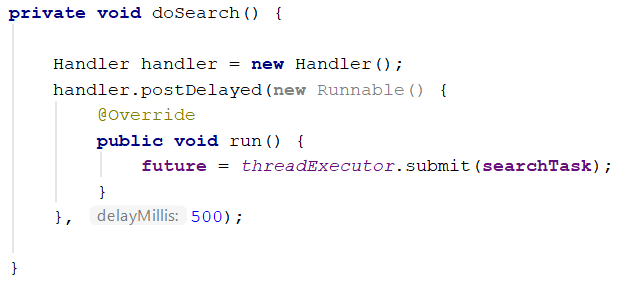


Рисунок 3 – Метод doSerch класса lab5activity

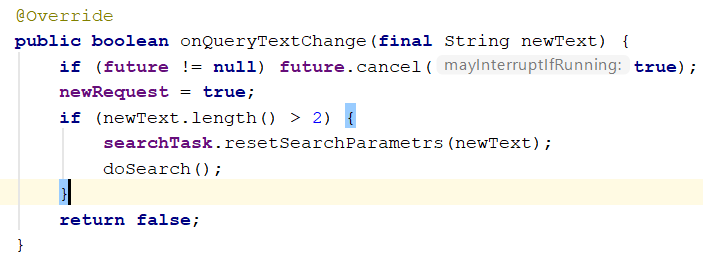


Рисунок 4 – Метод onQueryTextChange класса SearchView

# 3 Результаты работы

В результате выполнения лабораторной работы было дополнено android-приложение с прошлой лр: создан модуль, позволяющий выполнять запросы к сети на поиск репозиториев по введенной строке в SearchView. Полученные 20 репозитории отображаются в RecyclerView (Рисунок 5). При прокрутке списка до конца в RecyclerView подгружаются следующие 20 репозиториев. Все фоновые запросы сопровождаются отображением ProgressBar (Рисунок 6). При возникновении ошибки появляется кнопка для повторения запроса (Рисунок 7).

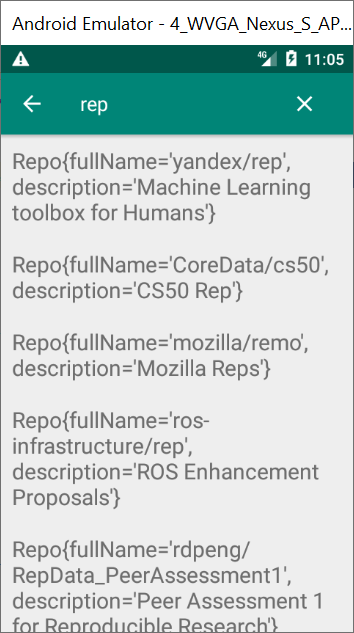


Рисунок 5 – Поиск репозиториев по подстроке «rep»

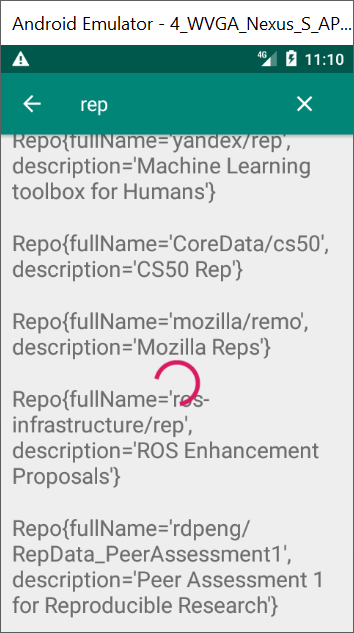


Рисунок 6 – Отображение PgrogressBar



Рисунок 6 – Отображение кнопки «Повторить запрос»

# 4 Вывод

В процессе выполнения данной лабораторной работы получены навыки разработки android-приложений, а именно: выполнение фоновых потоков с задержкой с использованием ExecutorService и Handler, прерывание потоков, выполнение запросов к сети на получение данных с помощью библиотеки Okhttp3, использование SearchView, ProgressBar и SwipeRefreshLayout, упраления видимостью элементов из кода.