**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»

Отчёт по лабораторной работе № 3

по дисциплине

**«Программирование мобильных устройств и встраиваемых систем»**

вариант № 4

Выполнил:

Студент группы 8И6А Кагарманов М.А. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

Томск, 2020 г.

**Задание**

Ознакомиться с примером в модуле lab3.

Над кнопкой добавления студента добавить еще один FloatingActionButton, который открывает диалог (AlertDialog) с добавлением группы (название группы и кнопка «ок»). На экране добавления студента добавить Spinner, который позволяет выбрать, к какой группе принадлежит студент. При нажатии на студента в списке, открывать AddStudentActivity с заполненными параметрами студента. Выводить список студентов, сгруппированный по группам. Т.е. сначала выводится группа, потом список студентов в группе.

**Ход работы**

Ссылка на репозиторий: <https://github.com/maximusmeridimus/tpu-android-courses/tree/lab-2-3>

Для работы с группами был добавлен класс Group. В нем описывается название группы.

Был добавлен кэш групп на основе паттерна «синглтон» по аналогии с кэшем студентов.

Для создания новых групп был использован AlertDialog. Диалог имеет кастомную view, описываемую в layout/lab3\_activity\_add\_group. При сохранении группы группа заносится в кэш.

При нажатии на студента открывается AddStudentActivity с его параметрами (имя, отчество, фамилия, группа). Для этого был определен интерфейс OnClickListenner в StudentAdapter`e и в StudentHolder`e он вызывается при назначении уже стандартного listener`a от view.

Группировка происходит следующим образом. При старте приложения создаются сначала группы с данными. Затем для каждой учебной группы сначала вставляем в RecyclerView один GroupItem и StudentItem`ы для каждого студента. Таким образом число элементов в RecyclerView = числу групп + числу студентов.

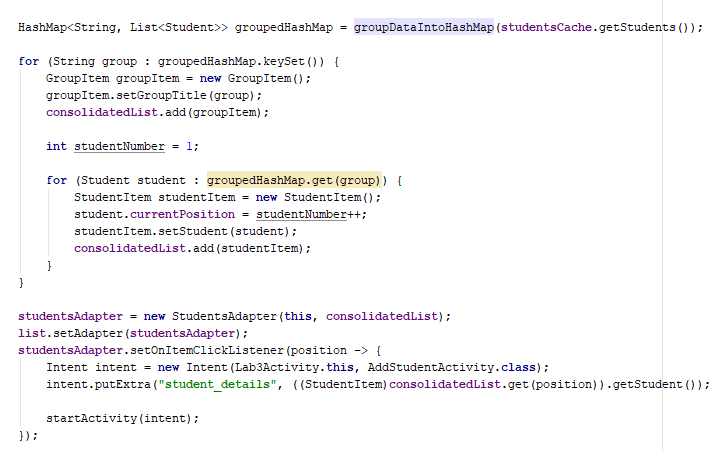


Рисунок – Инициализация адаптера для RecyclerView

При добавлении группы она просто добавляется в конец RecyclerView.

При добавлении студента в группу необходимо было высчитать номер его позиции в списке элементов RecyclerView. Для этого высчитывается индекс View с названием группы и к нему добавляется число студентов в этой группе, так как студент вставляется в конец группы.

Подробнее можно посмотреть код в репозитории.

**Результаты работы программы**

Главный экран модуля 3. Нижняя floating action button для добавления студента, верхняя – для добавления группы. Студенты сгруппированы по группам: сначала выводится номер группы, затем список студентов этой группы, причем, пронумерованный.

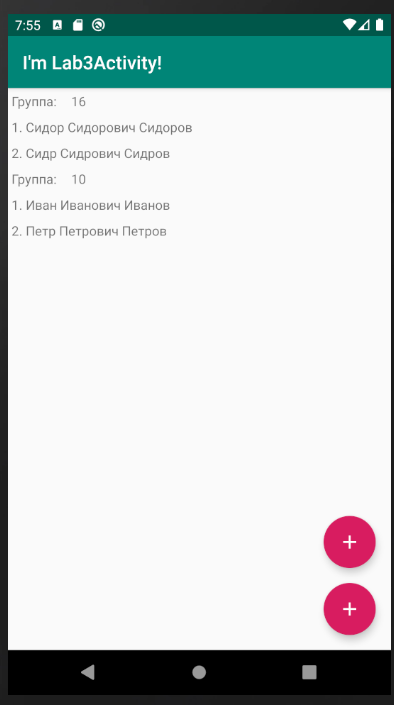


Рисунок – Главный экран модуля 3

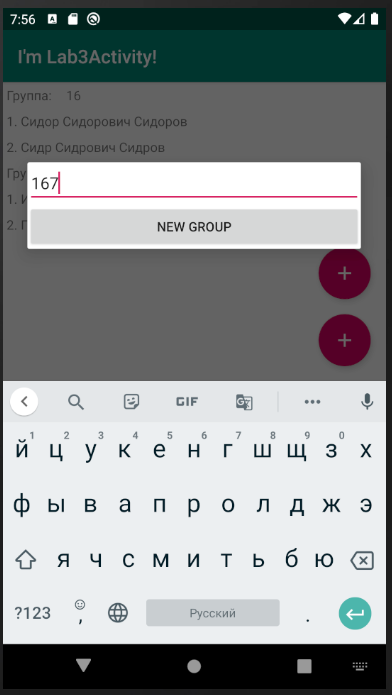


Рисунок – Добавление группы

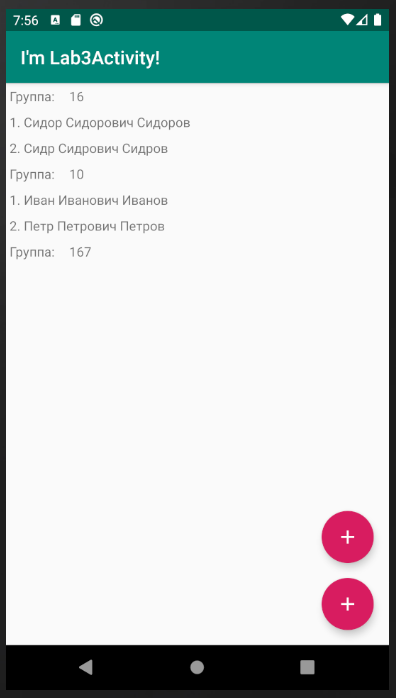


Рисунок – Добавлена новая группа

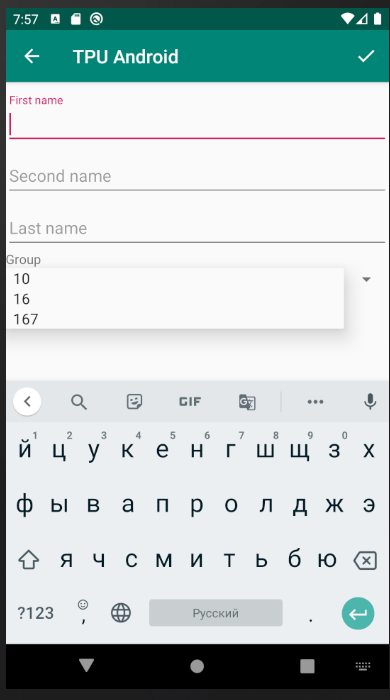


Рисунок – Выбор группы при добавлении студента

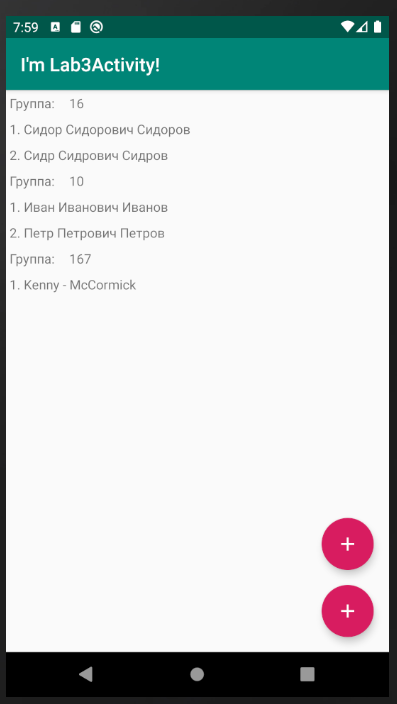


Рисунок – В новую группу добавлен новый студент

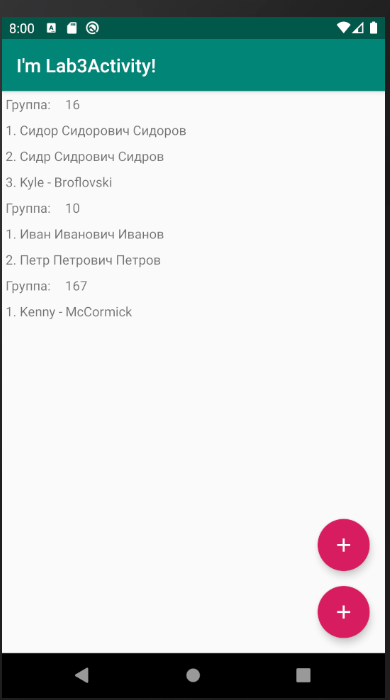


Рисунок – В старую группу добавлен новый студент

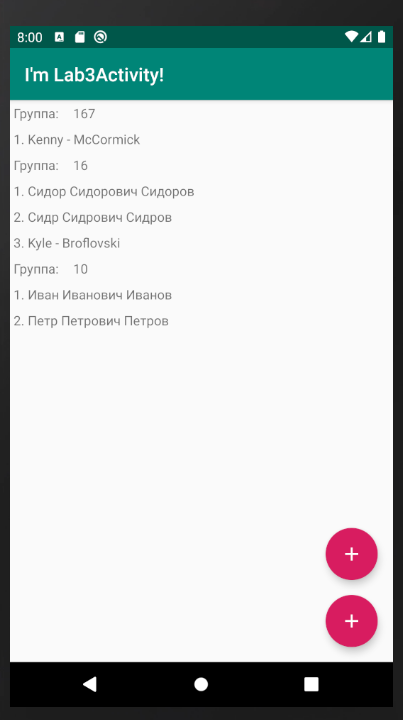


Рисунок – После перезахода в модуль данные сохраняются

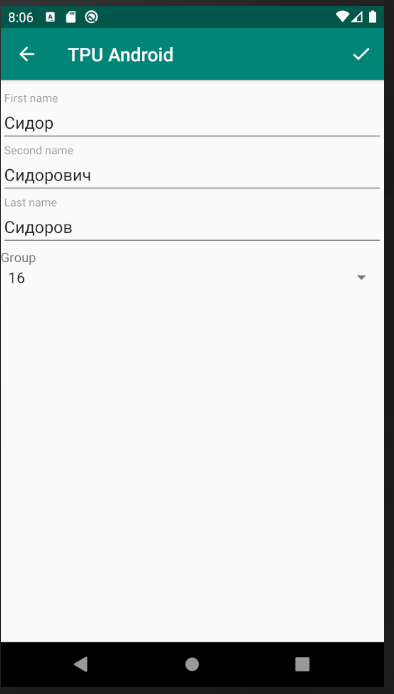


Рисунок – Открытие AddStudentActivity для выбранного студента

**Вывод**

В результате лабораторной работы было выполнено поставленное задание: добавлены группы для студентов, открытие деталей студента при клике, студенты сгруппированы в группы.