**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное автономное образовательное учреждение высшего образования

**«Национальный исследовательский Томский политехнический университет»**

Инженерная школа информационных технологий и робототехники

Направление «Информационные системы и технологии»

Отчёт по лабораторной работе №2

**«Вёрстка UI. Сохранение состояния»**

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

Выполнил:

Студент группы 8И6А Козловский В. Е. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Проверил:

Преподаватель ОИТ Погребной А.В. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Томск 2020

# Цель работы

# Целью лабораторной работы является практическое закрепление методов верстки пользовательских интерфейсов при разработке в среде Android Studio, а также изучение принципов управления состояниями элементов.

# Задание

Вариант 4. Custom View содержит вертикальный список записей. Каждая запись состоит из трёх View. Первая – заголовок. Вторая – полоска, отображающая рейтинг (может быть в диапазоне от 0 до 10). Наивысший рейтинг отображается красным цветом, остальные чёрным. Последняя – текстовое представление рейтинга. Сверху над Custom View находится 2 EditText-а и кнопка. При нажатии на кнопку заголовок и оценка получаются из EditText-ов и добавляются в CustomView.

# Ход работы

Для отображения названия записи и её рейтинга в виде числа и полоски progress-bar было реализовано представление Lab2View:

**package** ru.tpu.courses.lab2;  
  
**import** android.content.Context;  
**import** android.util.AttributeSet;  
**import** android.view.View;  
**import** android.widget.TextView;  
**import** android.widget.ProgressBar;  
  
**import** androidx.constraintlayout.widget.ConstraintLayout;  
  
**public class** Lab2View **extends** ConstraintLayout  
{  
 **private** TextView **titleTextView**;  
 **private** ProgressBar **progressBar**;  
 **private** TextView **valueTextView**;  
   
 **public** Lab2View(Context context) {  
 **this**(context, **null**);  
 }  
  
 **public** Lab2View(Context context, AttributeSet attrs) {  
 **this**(context, attrs, 0);  
 }  
  
 **public** Lab2View(Context context, AttributeSet attrs, **int** defStyleAttr) {  
 **super**(context, attrs, defStyleAttr);  
 View mainView = *inflate*(context, R.layout.*lab2\_view*, **null**);  
 **titleTextView** = mainView.findViewById(R.id.*lab2\_titletextview*);  
 **valueTextView** = mainView.findViewById(R.id.*lab2\_valuetextview*);  
 **progressBar** = mainView.findViewById(R.id.*lab2\_progressbar*);  
 addView(mainView);  
 }  
  
 **public void** setTitle(String title){  
 **titleTextView**.setText(String.*valueOf*(title));  
 }  
  
 **public void** setValue(**double** value){  
 **valueTextView**.setText(String.*valueOf*(value));  
 **progressBar**.setProgress(((**int**) Math.*round*(value)));  
  
 }  
}

Для установки значений названия и рейтинга, отображаемых в элементах интерфейса, объявлены публичные методы “setTitle” и “setValue”. Внешний вид activity задается в xml-шаблоне “lab2\_view”:

*<?***xml version="1.0" encoding="utf-8"***?>*<**androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
  
 <**TextView  
 android:id="@+id/lab2\_titletextview"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="56dp"  
 android:textSize="16sp"  
 app:layout\_constraintEnd\_toStartOf="@+id/lab2\_progressbar"  
 app:layout\_constraintHorizontal\_bias="0.246"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
 <**TextView  
 android:id="@+id/lab2\_valuetextview"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="24dp"  
 android:layout\_marginLeft="24dp"  
 android:layout\_marginTop="56dp"  
 android:textSize="16sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toEndOf="@+id/lab2\_progressbar"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
 <**ProgressBar  
 android:id="@+id/lab2\_progressbar"  
 style="?android:attr/progressBarStyleHorizontal"  
 android:layout\_width="174dp"  
 android:layout\_height="22dp"  
 android:layout\_marginTop="56dp"  
 android:max="10"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.763"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
</**androidx.constraintlayout.widget.ConstraintLayout**>

Для сохранения состояния элементов в основном activity модуля значения рейтинга записываются в созданный заранее массив “viewsValues”:

@Override  
**protected void** onSaveInstanceState(@NonNull Bundle outState) {  
 **super**.onSaveInstanceState(outState);  
 outState.putDoubleArray(***STATE\_VIEWS\_VALUES***, **lab2ViewsContainer**.getViewsValues());  
}

Для восстановления состояния был переопределен метод обработки события “onRestoreInstanceState”:

@Override  
**protected void** onRestoreInstanceState(Bundle savedInstanceState) {  
 **super**.onRestoreInstanceState(savedInstanceState);  
 **if** (savedInstanceState != **null**) {  
 **lab2ViewsContainer**.setViewsValues(savedInstanceState.getDoubleArray(***STATE\_VIEWS\_VALUES***));  
 }  
}

# Результат

При переходе к модулю 2 лабораторной работы на экран устройства выводятся начальные записи в количестве, заранее указанном программно, со случайно сгенерированным рейтингом (рис. 1).

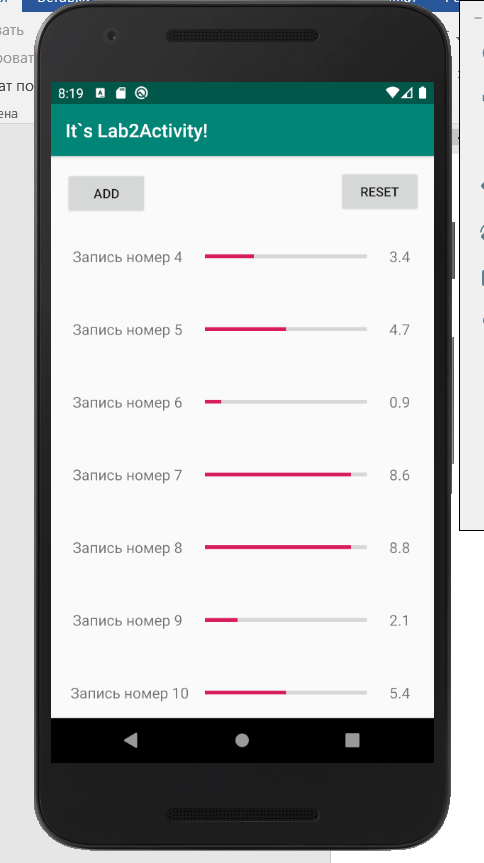
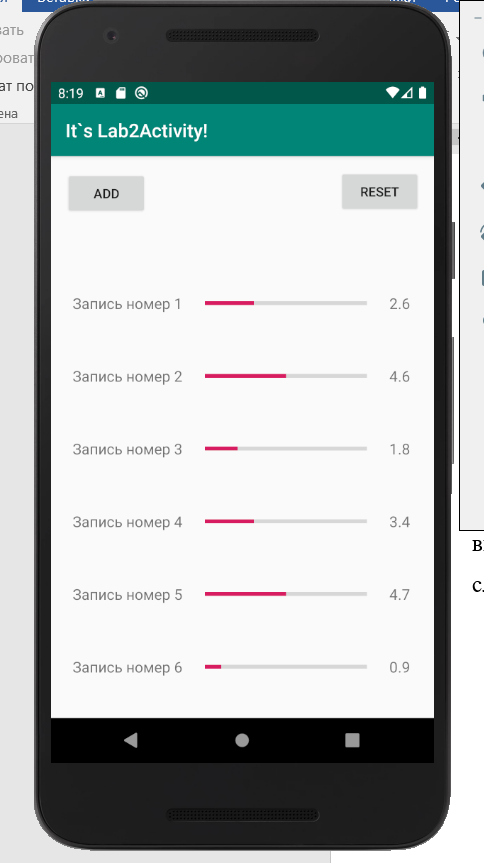


Рисунок 1 – Начальный вид модуля

При нажатии на кнопку “Add” происходит добавление записей со случайно сгенерированным рейтингом (рис. 2).

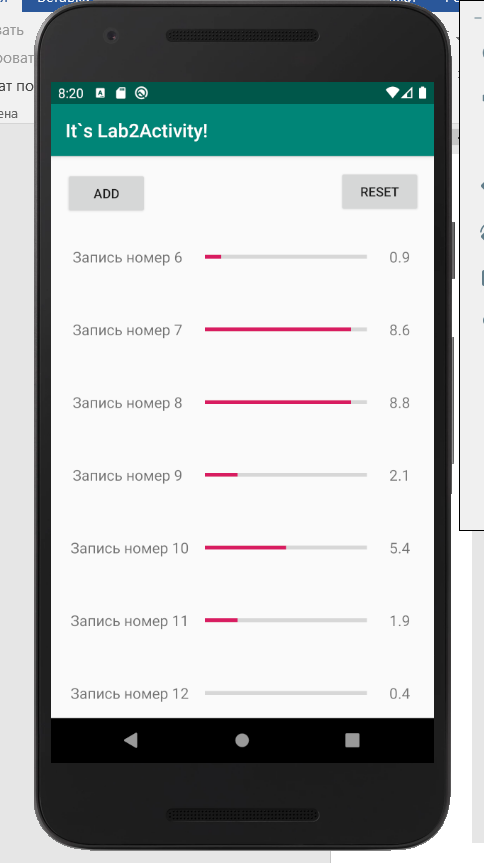


Рисунок 2 – Добавлены две записи

При обновлении состояния (например, изменение ориентации экрана) записи, как и связанные с ними элементы интерфейса сохраняют своё состояние (рис. 3).

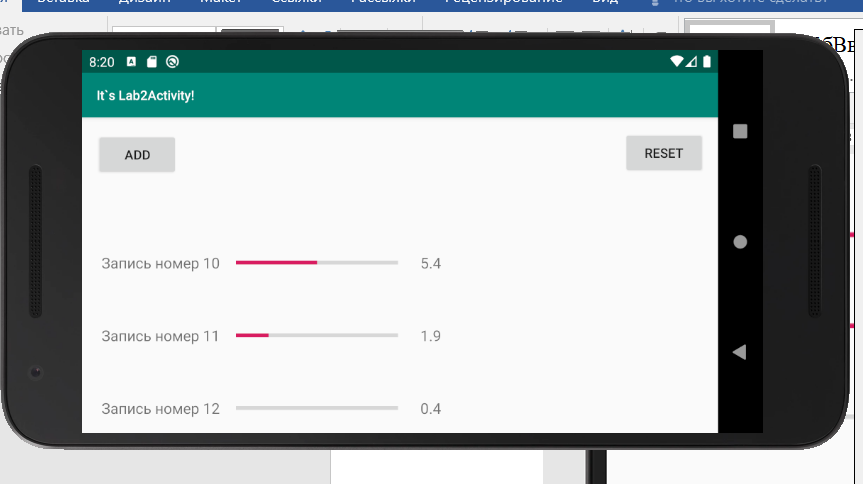


Рисунок 3 – Изменена ориентация экрана

При нажатии на кнопку “Reset” записи в исходном количестве генерируются заново (рис. 4).

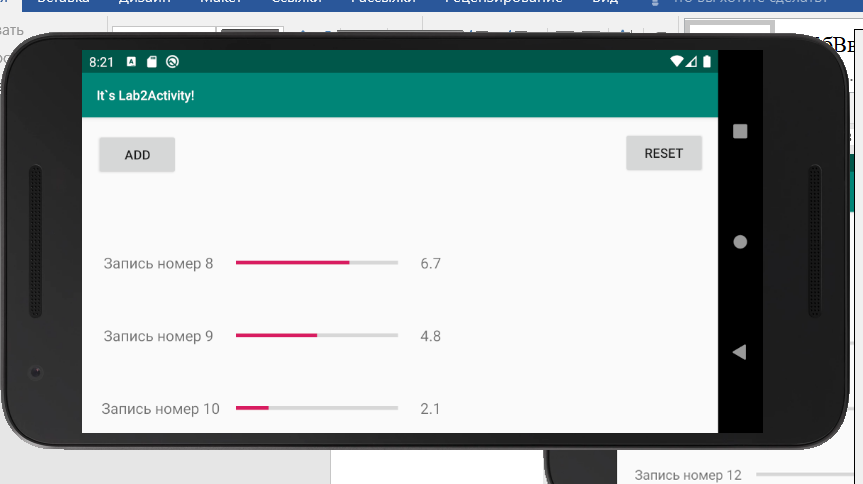


Рисунок 4 – Записи сгенерированы заново

# Вывод

В ходе выполнения лабораторной работы были закреплены на практике знания принципов сохранения состояния элементов окна. Был реализован собственный элемент интерфейса, сформирован макет представления.