МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное автономное образовательное учреждение

высшего образования

**«Национальный исследовательский**

**Нижегородский государственный университет им. Н.И. Лобачевского»**

**Институт информационных технологий, математики и механики**

Направление подготовки «Фундаментальная информатика и информационные технологии»

**ОТЧЕТ**

по лабораторной работе

**Практическое задание для лекции №7**

**Выполнил:** студент группы

381906-2 Углинский Б.С

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Подпись

**Проверил:** Карчков Д.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Подпись

Нижний Новгород

2022

Оглавление

[1. Цели работы. 3](#_Toc120895110)

[2. Постановка задачи. 4](#_Toc120895111)

[3. Решение задачи. 5](#_Toc120895112)

[4. Руководство пользователя. 6](#_Toc120895113)

[Приложение. 8](#_Toc120895114)

# 1. Цели работы.

В процессе выполнения данной лабораторной работы необходимо научится создавать пользовательский интерфейс конфигурируемый состояниям, реализовывать анимированные переходы при смене состояния.

# 2. Постановка задачи.

Для выполнения лабораторной работы были поставлены следующие задачи:

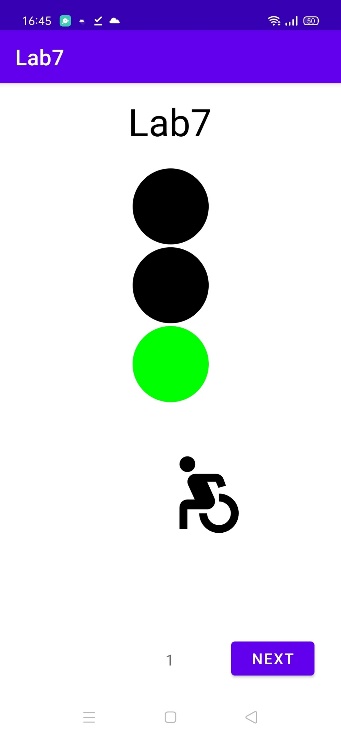
1. Сделать страницу, на которой будет располагаться светофор. Также добавить на страницу иконку человека. Сигналы светофора должны сменяться, также, как и в реальности. Реализовать соответствующее поведение пешехода на сигналы светофора.
2. Создать страницу, содержащую простую строку с текстом. При нажатии на текст он должен плавно перемещаться вниз экрана и поворачиваться на 180 градусов, меняя цвет. При прекращении нажатия, он должен так же плавно вернуться в исходное положение.
3. Сделать страницу, на которой будут располагаться текстовые поля. При нажатии на поле должна создаваться кнопка с таким же названием, как и текстовое поле. При создании кнопки должна быть возможность выбрать её цвет.
4. Создать секундомер, отображающий часы, минуты и секунды. Должна быть возможность поставить секундомер на паузу.
5. Сделать страницу, на которой можно будет добавлять новые вкладки, а также, удалять их. При этом должно подсчитываться количество добавленных и удалённых вкладок.

# 3. Решение задачи.

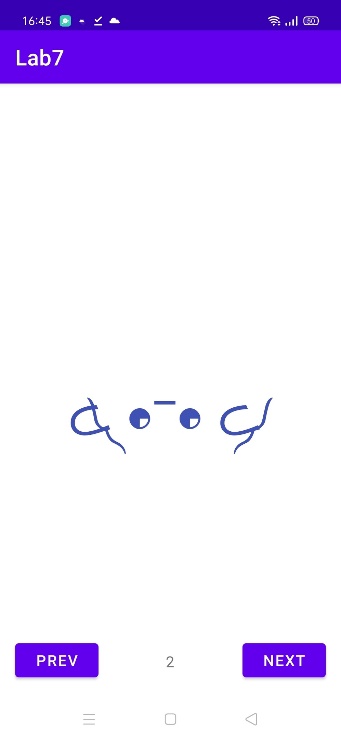
1. Для создания светофора, необходимо прописать алгоритм переключения сигналов, и заменять иконку текущего сигнала на соответствующий, а у остальных двух сигналов необходимо заменять иконки на неактивные. Чтобы при этом приложение по-прежнему было отзывчивым и реагировало на нажатия, необходимо все анимации запускать в отдельном потоке. Анимация человека синхронизируются с анимацией светофора.
2. Чтобы текст плавно перемещался, пока идёт нажатие необходимо добавить тексту обработку OnTouchListener, а не OnClickListener. После чего разграничить поведение при различных типах нажатия(MotionEvent.ACTION\_DOWN и MotionEvent.ACTION\_UP).
3. Для того, чтобы создавать кнопку при нажатии на текстовое поле, необходимо добавить обработчик нажатий для текстового поля, в котором будет создаваться новая кнопка, а также вызываться диалог с выбором цвета, который будет устанавливаться в качестве цвета кнопки.
4. Для реализации секундомера нужно воспользоваться элементом Chronometer, после чего определить в нём обработку изменения времени(setOnChronometerTickListener). Чтобы сделать паузу, нужно запоминать сколько времени прошло в момент нажатия на паузу, после чего сбрасывать состояние секундомера и добавлять полученное время к новым показателям секундомера.
5. Для создания и удаления вкладок нужно создать класс, который будет наследником FragmentStateAdapter. Он будет управлять созданием вкладок. Также необходимо создать класс, наследника Fragment, который будет отвечать за поведение одной вкладки. Дальше нужно добавить двум кнопкам обработку удаления и добавления вкладок, а также создать два счётчика, для удалённых и добавленных вкладок.

# 4. Руководство пользователя.

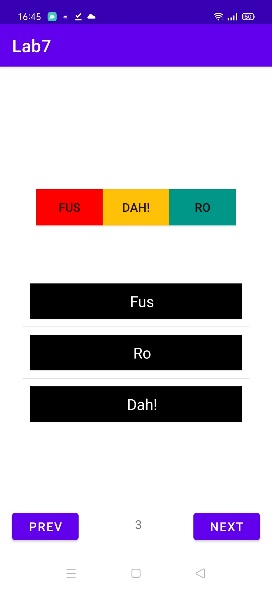
На главном экране пользователь увидит переключающиеся с определённым интервалом сигналы светофора, а также переходящего в нужный момент дорогу пешехода.



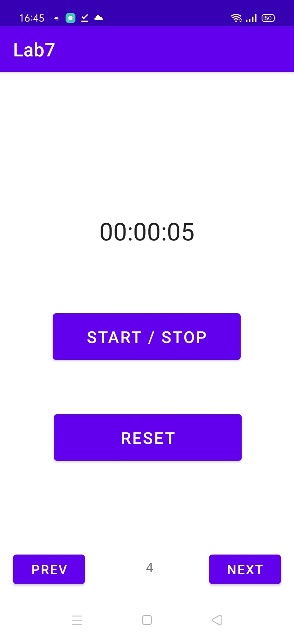
На второй странице располагается текст, при зажатии которого, он переместится вниз, повернётся на 180 градусов и станет синим. При прекращении нажатия, текст ожидают обратные метаморфозы.



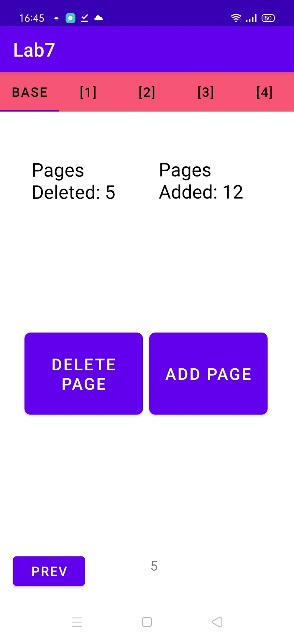
Далее идёт страница с текстовыми полями, при нажатии на которые будут создаваться кнопки, с выбранным цветом.



На следующей странице расположен секундомер с кнопками Старт/Стоп и Сбросить.



На последней странице пользователь может добавлять и удалять вкладки.



# Приложение.

* MainActivity.java

package com.lab7;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.view.animation.Animation;  
import android.view.animation.AnimationUtils;  
import android.widget.ImageView;  
import android.widget.TextView;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
 int light\_id = 2;  
 boolean isGreen = true;  
 boolean flag\_plus = false;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 TextView redLight = findViewById(R.id.RedLight);  
 TextView yellowLight = findViewById(R.id.YellowLight);  
 TextView greenLight = findViewById(R.id.GreenLight);  
 ImageView Human = findViewById(R.id.Human);  
  
 TextView[] lights = {redLight, yellowLight, greenLight};  
 int[] lightsSource = {R.drawable.red\_light, R.drawable.yellow\_light, R.drawable.green\_light};  
 Thread thread = new Thread(() -> {  
 while (true) {  
 if (light\_id == 2) {  
 if (isGreen) {  
 Animation forwardAnim = AnimationUtils.*loadAnimation*(this, R.anim.human\_go\_forward);  
 Human.startAnimation(forwardAnim);  
 } else {  
 Animation backwardAnim = AnimationUtils.*loadAnimation*(this, R.anim.human\_go\_backward);  
 Human.startAnimation(backwardAnim);  
 }  
 flag\_plus = false;  
 } else if (light\_id == 0) {  
 flag\_plus = true;  
 }  
 lights[light\_id].setBackgroundResource(lightsSource[light\_id]);  
 if (light\_id == 0) {  
 if (isGreen) {  
 Human.setRotationY(180);  
 isGreen = false;  
 } else {  
 Human.setRotationY(0);  
 isGreen = true;  
 }  
 }  
 try {  
 Thread.*sleep*(1000);  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
 lights[light\_id].setBackgroundResource(R.drawable.circle);  
  
 if (flag\_plus)  
 light\_id++;  
 else light\_id--;  
 }  
 });  
 thread.start();  
  
 }  
  
 public void NextActivity(View v) {  
 startActivity(new Intent(this, TextMotion.class));  
 }  
}

* activity\_main.xml
* <?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"  
   android:backgroundTint="@null"  
   tools:context=".MainActivity">  
    
   <TextView  
   android:id="@+id/RedLight"  
   android:layout\_width="wrap\_content"  
   android:layout\_height="wrap\_content"  
   android:layout\_margin="3dp"  
   android:background="@drawable/circle"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="0.501"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.155" />  
    
   <TextView  
   android:id="@+id/YellowLight"  
   android:layout\_width="wrap\_content"  
   android:layout\_height="wrap\_content"  
   android:layout\_margin="3dp"  
   android:background="@drawable/circle"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="0.501"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toBottomOf="@+id/RedLight"  
   app:layout\_constraintVertical\_bias="0.0" />  
    
   <TextView  
   android:id="@+id/GreenLight"  
   android:layout\_width="wrap\_content"  
   android:layout\_height="wrap\_content"  
   android:layout\_margin="3dp"  
   android:background="@drawable/circle"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="0.501"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toBottomOf="@+id/YellowLight"  
   app:layout\_constraintVertical\_bias="0.0" />  
    
   <ImageView  
   android:id="@+id/Human"  
   android:layout\_width="wrap\_content"  
   android:layout\_height="wrap\_content"  
   android:background="@drawable/human2"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="0.051"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.7" />  
    
   <TextView  
   android:id="@+id/task\_txt"  
   android:layout\_width="wrap\_content"  
   android:layout\_height="wrap\_content"  
   android:layout\_marginBottom="28dp"  
   android:text="1"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="0.498"  
   app:layout\_constraintStart\_toStartOf="parent" />  
    
   <Button  
   android:id="@+id/NextPage"  
   android:layout\_width="wrap\_content"  
   android:layout\_height="wrap\_content"  
   android:layout\_marginEnd="28dp"  
   android:layout\_marginBottom="16dp"  
   android:onClick="NextActivity"  
   android:text="Next"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent" />  
    
   <TextView  
   android:id="@+id/textView3"  
   android:layout\_width="wrap\_content"  
   android:layout\_height="wrap\_content"  
   android:text="Lab7"  
   android:textColor="@color/black"  
   android:textSize="40dp"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="0.498"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.023" />  
    
  </androidx.constraintlayout.widget.ConstraintLayout>
* TextMotion.java

package com.lab7;  
  
import android.annotation.SuppressLint;  
import android.content.Intent;  
import android.graphics.Color;  
import android.os.Bundle;  
import android.view.MotionEvent;  
import android.view.View;  
import android.view.animation.Animation;  
import android.view.animation.AnimationUtils;  
import android.widget.TextView;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
public class TextMotion extends AppCompatActivity {  
 @SuppressLint("ClickableViewAccessibility")  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_text\_motion);  
 TextView txt = findViewById(R.id.textView);  
 txt.setOnTouchListener(new View.OnTouchListener() {  
 @Override  
 @SuppressLint("ResourceType")  
 public boolean onTouch(View view, MotionEvent event) {  
 switch (event.getAction()) {  
 case MotionEvent.*ACTION\_DOWN*: {  
 txt.setTextColor(Color.*parseColor*(getString(R.color.blue)));  
 Animation descend = AnimationUtils.*loadAnimation*(TextMotion.this, R.anim.text\_descend);  
 txt.startAnimation(descend);  
 break;  
 }  
 case MotionEvent.*ACTION\_UP*: {  
 Animation ascend = AnimationUtils.*loadAnimation*(TextMotion.this, R.anim.text\_ascend);  
 txt.startAnimation(ascend);  
 txt.setTextColor(Color.*parseColor*(getString(R.color.black)));  
 break;  
 }  
 }  
 return true;  
 }  
 });  
  
 }  
  
 public void NextActivity(View v) {  
 startActivity(new Intent(this, Buttons.class));  
 }  
  
 public void PrevActivity(View v) {  
 startActivity(new Intent(this, MainActivity.class));  
 }  
}

* activity\_text\_motion.xml

<?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"  
 tools:context=".TextMotion">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="༼ つ ◕\_◕ ༽つ"  
 android:textColor="@color/black"  
 android:textSize="50dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.5"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.1" />  
  
 <Button  
 android:id="@+id/NextPage"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginEnd="16dp"  
 android:layout\_marginBottom="16dp"  
 android:onClick="NextActivity"  
 android:text="Next"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent" />  
  
 <TextView  
 android:id="@+id/task\_txt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="28dp"  
 android:text="2"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toStartOf="@+id/NextPage"  
 app:layout\_constraintHorizontal\_bias="0.497"  
 app:layout\_constraintStart\_toEndOf="@+id/PreviousPage" />  
  
 <Button  
 android:id="@+id/PreviousPage"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="16dp"  
 android:layout\_marginBottom="16dp"  
 android:onClick="PrevActivity"  
 android:text="prev"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

* Buttons.java

package com.lab7;  
  
import android.content.Context;  
import android.content.Intent;  
import android.graphics.Color;  
import android.os.Bundle;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.AdapterView;  
import android.widget.ArrayAdapter;  
import android.widget.Button;  
import android.widget.LinearLayout;  
import android.widget.ListView;  
import android.widget.TextView;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import com.jaredrummler.android.colorpicker.ColorPickerDialog;  
import com.jaredrummler.android.colorpicker.ColorPickerDialogListener;  
import com.jaredrummler.android.colorpicker.ColorShape;  
  
import java.util.ArrayList;  
  
public class Buttons extends AppCompatActivity implements ColorPickerDialogListener {  
 Button myButton;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_buttons\_begin);  
  
 ListView listView = findViewById(R.id.listView);  
 LinearLayout ll = findViewById(R.id.ButtonsStack);  
  
 ArrayList Actions = new ArrayList<String>();  
 Actions.add("Fus");  
 Actions.add("Ro");  
 Actions.add("Dah!");  
 ListAdapter adapter = new ListAdapter(this, R.layout.list\_items, Actions);  
 listView.setAdapter(adapter);  
  
 listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
 myButton = new Button(Buttons.this);  
 myButton.setText((CharSequence) Actions.get(position));  
 createColorPickerDialog(view.getId());  
 ll.addView(myButton);  
 }  
 });  
  
 }  
  
 private void createColorPickerDialog(int id) {  
 ColorPickerDialog.newBuilder()  
 .setColor(Color.*RED*)  
 .setDialogType(ColorPickerDialog.TYPE\_PRESETS)  
 .setAllowCustom(true)  
 .setAllowPresets(true)  
 .setColorShape(ColorShape.SQUARE)  
 .setDialogId(id)  
 .show(this);  
 }  
  
 @Override  
 public void onColorSelected(int dialogId, int color) {  
 myButton.setBackgroundColor(color);  
 }  
  
 @Override  
 public void onDialogDismissed(int dialogId) {  
   
 }  
  
 public void NextActivity(View v) {  
 startActivity(new Intent(this, Sekundomer.class));  
 }  
  
 public void PrevActivity(View v) {  
 startActivity(new Intent(this, TextMotion.class));  
 }  
  
 public class ListAdapter extends ArrayAdapter {  
  
 private int resourceLayout;  
 private Context mContext;  
 private ArrayList<String> source;  
  
 public ListAdapter(Context context, int resource, ArrayList<String> Notes) {  
 super(context, resource, Notes);  
 this.resourceLayout = resource;  
 this.mContext = context;  
 this.source = Notes;  
 }  
  
 @Override  
 public View getView(int position, View convertView, ViewGroup parent) {  
  
 View v = convertView;  
  
 if (v == null) {  
 LayoutInflater vi;  
 vi = LayoutInflater.*from*(mContext);  
 v = vi.inflate(resourceLayout, null);  
 }  
 TextView item = v.findViewById(R.id.Item);  
 item.setText(source.get(position));  
  
 return v;  
 }  
 }  
}

* activity\_buttons.xml

<?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"  
 tools:context=".Buttons">  
  
 <ListView  
 android:id="@+id/listView"  
 android:layout\_width="300dp"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.622"></ListView>  
  
 <LinearLayout  
 android:id="@+id/ButtonsStack"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="60sp"  
 android:orientation="horizontal"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.501"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.28"></LinearLayout>  
  
 <Button  
 android:id="@+id/NextPage"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginEnd="16dp"  
 android:layout\_marginBottom="16dp"  
 android:onClick="NextActivity"  
 android:text="Next"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent" />  
  
 <TextView  
 android:id="@+id/task\_txt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="32dp"  
 android:text="3"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toStartOf="@+id/NextPage"  
 app:layout\_constraintHorizontal\_bias="0.523"  
 app:layout\_constraintStart\_toEndOf="@+id/PreviousPage" />  
  
 <Button  
 android:id="@+id/PreviousPage"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="16dp"  
 android:layout\_marginBottom="16dp"  
 android:onClick="PrevActivity"  
 android:text="prev"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

* Sekundomer.java

package com.lab7;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.os.SystemClock;  
import android.view.View;  
import android.widget.Chronometer;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
public class Sekundomer extends AppCompatActivity {  
 boolean isStopped = false;  
 boolean isStarted = false;  
 long totalTime = 0;  
 private Chronometer mChronometer;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_sekundomer);  
  
 mChronometer = findViewById(R.id.chronometer);  
 mChronometer.setText("00:00:00");  
 mChronometer.setOnChronometerTickListener(new Chronometer.OnChronometerTickListener() {  
 @Override  
 public void onChronometerTick(Chronometer chronometer) {  
 long time = SystemClock.*elapsedRealtime*() - chronometer.getBase() + totalTime;  
 int h = (int) (time / 3600000);  
 int m = (int) (time - h \* 3600000) / 60000;  
 int s = (int) (time - h \* 3600000 - m \* 60000) / 1000;  
 String t = (h < 10 ? "0" + h : h) + ":" + (m < 10 ? "0" + m : m) + ":" + (s < 10 ? "0" + s : s);  
 chronometer.setText(t);  
 }  
 });  
 }  
  
 public void onStartClick(View view) {  
 if (!isStarted) {  
 mChronometer.setBase(SystemClock.*elapsedRealtime*());  
 mChronometer.start();  
 isStarted = true;  
 } else {  
 if (!isStopped) {  
 mChronometer.stop();  
 totalTime += SystemClock.*elapsedRealtime*() - mChronometer.getBase();  
 isStopped = true;  
 } else {  
 mChronometer.setBase(SystemClock.*elapsedRealtime*());  
 mChronometer.start();  
 isStopped = false;  
 }  
  
  
 }  
 }  
  
 public void onResetClick(View view) {  
 mChronometer.setBase(SystemClock.*elapsedRealtime*());  
 isStopped = false;  
 isStarted = false;  
 totalTime = 0;  
 mChronometer.stop();  
 mChronometer.setText("00:00:00");  
 }  
  
 public void NextActivity(View v) {  
 startActivity(new Intent(this, PageStack.class));  
 }  
  
 public void PrevActivity(View v) {  
 startActivity(new Intent(this, Buttons.class));  
 }  
}

* activity\_sekundomer.xml

<?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"  
 tools:context=".Sekundomer">  
  
  
 <Button  
 android:id="@+id/Start"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:width="200sp"  
 android:height="60sp"  
 android:onClick="onStartClick"  
 android:text="Start / Stop"  
 android:textSize="20dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.497"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.499" />  
  
  
 <Chronometer  
 android:id="@+id/chronometer"  
 android:layout\_width="301dp"  
 android:layout\_height="69dp"  
 android:gravity="center"  
 android:textSize="30dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.276" />  
  
 <Button  
 android:id="@+id/Reset"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:width="200sp"  
 android:height="60sp"  
 android:onClick="onResetClick"  
 android:text="Reset"  
 android:textSize="20dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.507"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/Start"  
 app:layout\_constraintVertical\_bias="0.246" />  
  
 <Button  
 android:id="@+id/NextPage"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginEnd="16dp"  
 android:layout\_marginBottom="16dp"  
 android:onClick="NextActivity"  
 android:text="Next"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent" />  
  
 <TextView  
 android:id="@+id/task\_txt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="32dp"  
 android:text="4"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toStartOf="@+id/NextPage"  
 app:layout\_constraintHorizontal\_bias="0.523"  
 app:layout\_constraintStart\_toEndOf="@+id/PreviousPage" />  
  
 <Button  
 android:id="@+id/PreviousPage"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="16dp"  
 android:layout\_marginBottom="16dp"  
 android:onClick="PrevActivity"  
 android:text="prev"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

* PageStack.java

package com.lab7;  
  
import android.annotation.SuppressLint;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.TextView;  
  
import androidx.annotation.NonNull;  
import androidx.annotation.Nullable;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.fragment.app.Fragment;  
import androidx.fragment.app.FragmentManager;  
import androidx.lifecycle.Lifecycle;  
import androidx.viewpager2.adapter.FragmentStateAdapter;  
import androidx.viewpager2.widget.ViewPager2;  
  
import com.google.android.material.tabs.TabLayout;  
import com.google.android.material.tabs.TabLayoutMediator;  
  
  
public class PageStack extends AppCompatActivity {  
 public static TextView *pageStackExpand*;  
 public static TextView *pageStackReduced*;  
 public static TextView *Number*;  
 static TabsPagerAdapter *adapter*;  
 static int *added* = 0;  
 static int *deleted* = 0;  
 static int *counter* = 0;  
 TabLayout tabs;  
 ViewPager2 viewpager;  
 TextView check;  
  
 @SuppressLint("ResourceType")  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_page\_stack);  
  
 tabs = findViewById(R.id.tabLayout);  
 viewpager = findViewById(R.id.tabs\_viewpager);  
 *adapter* = new TabsPagerAdapter(getSupportFragmentManager(), getLifecycle(), 1);  
 viewpager.setAdapter(*adapter*);  
 viewpager.setUserInputEnabled(true);  
  
 new TabLayoutMediator(tabs, viewpager, (tab, position) -> {  
 switch (position) {  
 case 0:  
 tab.setText("Base");  
 break;  
 default:  
 tab.setText("[" + String.*valueOf*(position) + "]");  
 break;  
 }  
 }  
 ).attach();  
 }  
  
 public void addPage(View view) {  
 TabLayout.Tab tab = tabs.newTab();  
 tab.setText("Tab");  
 tabs.addTab(tab, false);  
 *adapter*.setNumberOfTabs(*adapter*.getItemCount() + 1);  
 *pageStackExpand*.setText("Pages Added: " + Integer.*toString*(++*added*));  
 *adapter*.notifyDataSetChanged();  
 }  
  
 public void deletePage(View view) {  
 if (tabs.getTabCount() > 1) {  
 tabs.removeTab(tabs.getTabAt(1));  
 *adapter*.setNumberOfTabs(*adapter*.getItemCount() - 1);  
 *pageStackReduced*.setText("Pages Deleted: " + Integer.*toString*(++*deleted*));  
 *adapter*.notifyDataSetChanged();  
 }  
 }  
  
 public void PrevActivity(View v) {  
 startActivity(new Intent(this, Sekundomer.class));  
 }  
  
  
 public static class MainFragment extends Fragment {  
 @Override  
 public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {  
 View v = inflater.inflate(R.layout.fragment\_main, container, false);  
 *pageStackExpand* = v.findViewById(R.id.PageStackExpand);  
 *pageStackReduced* = v.findViewById(R.id.PageStackReduced);  
 return v;  
 }  
  
 }  
  
 public static class PageFragment extends Fragment {  
 @Override  
 public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {  
 View v = inflater.inflate(R.layout.blank\_fragment, container, false);  
 *Number* = v.findViewById(R.id.Number);  
 *Number*.setText("༼ つ ◕\_◕ ༽つ");  
 return v;  
 }  
 }  
  
 class TabsPagerAdapter extends FragmentStateAdapter {  
 public int numberOfTabs;  
 FragmentManager fm;  
 Lifecycle lifecycle;  
  
 public TabsPagerAdapter(@NonNull FragmentManager fragmentManager, @NonNull Lifecycle \_lifecycle, int n) {  
 super(fragmentManager, \_lifecycle);  
 fm = fragmentManager;  
 lifecycle = \_lifecycle;  
 numberOfTabs = n;  
 }  
  
 @NonNull  
 @Override  
 public Fragment createFragment(int position) {  
 switch (position) {  
 case 0: {  
 Bundle bundle = new Bundle();  
 bundle.putString("fragmentName", "Music Fragment");  
 MainFragment newFragment = new MainFragment();  
 newFragment.setArguments(bundle);  
 return newFragment;  
 }  
 case 1: {  
 Bundle bundle = new Bundle();  
 bundle.putString("fragmentName", "Movies Fragment");  
 PageFragment newFragment = new PageFragment();  
 newFragment.setArguments(bundle);  
 return newFragment;  
 }  
 default:  
 return new PageFragment();  
 }  
 }  
  
 @Override  
 public int getItemCount() {  
 return numberOfTabs;  
 }  
  
 public void setNumberOfTabs(int numberOfTabs) {  
 this.numberOfTabs = numberOfTabs;  
 }  
 }  
  
}

* activity\_page\_stack.xml

<?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"  
 tools:context=".PageStack"  
 >  
  
 <com.google.android.material.tabs.TabLayout  
 android:id="@+id/tabLayout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:background="#F65576"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.0"  
 app:tabTextColor="@color/black"  
 app:textAllCaps="false">  
  
 </com.google.android.material.tabs.TabLayout>  
  
 <androidx.viewpager2.widget.ViewPager2  
 android:id="@+id/tabs\_viewpager"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_marginTop="75sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.5" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>