

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Комсомольский-на-Амуре государственный университет»

Факультет компьютерных технологий

Кафедра ПУРИС

Лабораторная работа №10
по дисциплине «Программирование мобильных устройств»

Студент группы 0ВТб-1

Н.Д. Малышев

Преподаватель

В.А. Тихомиров

1 Разработка приложения

По заданию необходимо сделать чтобы, на Activity было два движка: вертикальный и горизонтальный. Они задают положение на экране объекта в Image элементе (например – шарика). Состояние движков сохраняется в файле конфигурации при закрытии программы, и восстанавливается при открытии программы. Кроме того, можно запоминать в файле положение шарика, нажатием кнопки Запомнить. И проиграть все запомненные его состояния кнопкой Проиграть.

В активности MainActivity (листинг 1.1) написан весь основной код, реализующий поставленную задачу.

В листинге 1.2 приведет файл макета activity_main.xml.

Листинг 1.1 – MainActivity

```
package ru.lab_10;

import android.content.Context;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.SeekBar;

import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {
    //инициализация объектов, которые будут использоваться
    private ImageView ball;
    private SeekBar horizontalSeekBar, verticalSeekBar;
    private Button rememberButton, playButton, clearButton;
    private SharedPreferences sharedPreferences;
    private SharedPreferences.Editor editor;
    private int[] savedHorizontalPositions,
savedVerticalPositions;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        ball = findViewById(R.id.ball);
        horizontalSeekBar =
```

```

findViewById(R.id.horizontal_seek_bar);
    verticalSeekBar = findViewById(R.id.vertical_seek_bar);
    rememberButton = findViewById(R.id.remember_button);
    playButton = findViewById(R.id.play_button);
    clearButton = findViewById(R.id.clear_button);
    sharedPreferences =
getSharedPreferences("MyPreferences", Context.MODE_PRIVATE);
    editor = sharedPreferences.edit();
    savedHorizontalPositions = new int[10];
    savedVerticalPositions = new int[10];
    horizontalSeekBar.setProgress(0);
    verticalSeekBar.setProgress(0);

    for (int i = 0; i < 10; i++) {
        savedHorizontalPositions[i] =
sharedPreferences.getInt("horizontalPosition" + i, 0);
        savedVerticalPositions[i] =
sharedPreferences.getInt("verticalPosition" + i, 0);
    }
    //проверка, сохранены ли какие-либо данные
    if (savedInstanceState != null) {
        int horizontalProgress =
savedInstanceState.getInt("horizontalProgress");
        int verticalProgress =
savedInstanceState.getInt("verticalProgress");
        float ballX = savedInstanceState.getFloat("ballX");
        float ballY = savedInstanceState.getFloat("ballY");
        horizontalSeekBar.setProgress(horizontalProgress);
        verticalSeekBar.setProgress(verticalProgress);
        ball.setX(ballX);
        ball.setY(ballY);
    } else {
        // Устанавливаем значения по умолчанию

horizontalSeekBar.setProgress(savedHorizontalPositions[0]);

verticalSeekBar.setProgress(savedVerticalPositions[0]);
    }

    //настройка seekbar для перемещения шарика
    horizontalSeekBar.setOnSeekBarChangeListener(new
SeekBar.OnSeekBarChangeListener() {
        @Override
        public void onProgressChanged(SearchBar seekBar, int
progress, boolean fromUser) {
            ball.setX(progress * 10); // Обновляем положение
шарика по горизонтальной оси
        }
        @Override
        public void onStartTrackingTouch(SearchBar seekBar) {}
        @Override
        public void onStopTrackingTouch(SearchBar seekBar) {}
    });

```

```

        //настройка seekbar для перемещения шарика
        verticalSeekBar.setOnSeekBarChangeListener(new
SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SearchBar seekBar, int
progress, boolean fromUser) {
                ball.setY(progress * 12); // Обновляем положение
шарика по вертикальной оси
            }
            @Override
            public void onStartTrackingTouch(SearchBar seekBar) {}
            @Override
            public void onStopTrackingTouch(SearchBar seekBar) {}
        });
        //обработка нажатия клавиши и сохранение с помощью нее
позиций шарика и seekbar
        rememberButton.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View view) {
                //сохраняет текущее состояние положения двух
SeekBar'ов
                //сдвигает все элементы массивов на одну позицию
вправо,
                // таким образом удаляя самое старое сохраненное
положение,
                // и сохраняет новое положение на первое место
                for (int i = 9; i > 0; i--) {
                    savedHorizontalPositions[i] =
savedHorizontalPositions[i - 1];
                    savedVerticalPositions[i] =
savedVerticalPositions[i - 1];
                }
                savedHorizontalPositions[0] =
horizontalSeekBar.getProgress();
                savedVerticalPositions[0] =
verticalSeekBar.getProgress();

                editor.putInt("horizontalPosition0",
savedHorizontalPositions[0]);
                editor.putInt("verticalPosition0",
savedVerticalPositions[0]);
                for (int i = 1; i < 10; i++) {
                    editor.putInt("horizontalPosition" + i,
savedHorizontalPositions[i]);
                    editor.putInt("verticalPosition" + i,
savedVerticalPositions[i]);
                }
                editor.apply();
            }
        });
        //кнопка play воспроизводит сохраненные позиции в
SeekBar'ax

```

```

        // и перемещает мячик на соответствующие координаты на
экране
        playButton.setOnClickListener(new View.OnClickListener()
        {
            @Override
            public void onClick(View view) {
                for (int i = 0; i < 10; i++) {
                    final int horizontalPosition =
savedHorizontalPositions[i];
                    final int verticalPosition =
savedVerticalPositions[i];
                    // проверка на наличие сохраненных координат
                    if (horizontalPosition != 0 &&
verticalPosition != 0) {
                        new Handler().postDelayed(new Runnable()
                        {
                            @Override
                            public void run() {

horizontalSeekBar.setProgress(horizontalPosition);

verticalSeekBar.setProgress(verticalPosition);

                                ball.setX(horizontalPosition *
10);
                                ball.setY(verticalPosition *
12);
                            }
                        }, i * 1000); // задержка между
воспроизведением состояний
                    }
                }

                new Handler().postDelayed(new Runnable() {
                    @Override
                    public void run() {
                    }
                }, 1000); // время, после которого проигрывание
останавливается
            }
        });
        //кнопка cleare удаляет все сохраненные позиции из
SeekBar'ax' и сбрасывает положение шарика
        clearButton.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View view) {
                editor.clear();
                editor.apply();
                for (int i = 0; i < 10; i++) {
                    savedHorizontalPositions[i] = 0;
                    savedVerticalPositions[i] = 0;
                }
            }
        });
    }
}

```

```

        }
        horizontalSeekBar.setProgress(0);
        verticalSeekBar.setProgress(0);
    }
    });
}
//В этом методе сохраняется состояние текущего положения
seekbar
// и мяча в SharedPreferences, используя editor
@Override
protected void onDestroy() {
    super.onDestroy();
    editor.putInt("horizontalPosition",
horizontalSeekBar.getProgress());
    editor.putInt("verticalPosition",
verticalSeekBar.getProgress());
    editor.apply();
}
// восстанавливается состояние элементов управления и
положение мяча из Bundle,
// который содержит ранее сохраненные значения в
onSaveInstanceState()
// также устанавливается текущее значение индикаторов и
положение мяча,
// чтобы они соответствовали сохраненным значениям
@Override
protected void onRestoreInstanceState(Bundle
savedInstanceState) {
    super.onRestoreInstanceState(savedInstanceState);
    int horizontalProgress =
savedInstanceState.getInt("horizontalProgress");
    int verticalProgress =
savedInstanceState.getInt("verticalProgress");
    float ballX = savedInstanceState.getFloat("ballX");
    float ballY = savedInstanceState.getFloat("ballY");
    horizontalSeekBar.setProgress(horizontalProgress);
    verticalSeekBar.setProgress(verticalProgress);
    ball.setX(horizontalProgress);
    ball.setY(verticalProgress);
}
//сохраняется текущее положение индикаторов и мяча в
SharedPreferences
@Override
protected void onPause() {
    super.onPause();
    editor.putInt("horizontalPosition",
horizontalSeekBar.getProgress());
    editor.putInt("verticalPosition",
verticalSeekBar.getProgress());
    editor.apply();
}

@Override

```

```

        protected void onStop() {
            super.onStop();
            editor.putInt("horizontalPosition",
horizontalSeekBar.getProgress());
            editor.putInt("verticalPosition",
verticalSeekBar.getProgress());
            editor.apply();
        }

        // Здесь восстанавливаются сохраненные ранее значения
индикаторов и положение мяча из SharedPreferences,
        // чтобы восстановить их состояние до того, как Activity
было уничтожено или свернуто
        @Override
        protected void onResume() {
            super.onResume();
            int horizontalPosition =
sharedPreferences.getInt("horizontalPosition", 0);
            int verticalPosition =
sharedPreferences.getInt("verticalPosition", 0);
            horizontalSeekBar.setProgress(horizontalPosition);
            verticalSeekBar.setProgress(verticalPosition);
            ball.setX(horizontalPosition * 10);
            ball.setY(verticalPosition * 12);
        }
    }
}

```

Листинг 1.2 – Activity_main

```

<?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=".MainActivity">

    <ImageView
        android:id="@+id/ball"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:layout_centerInParent="true"
        android:src="@drawable/ball"
        tools:ignore="MissingConstraints"
        tools:layout_editor_absoluteX="181dp"
        tools:layout_editor_absoluteY="310dp" />

    <Button
        android:id="@+id/remember_button"
        android:layout_width="120dp"
        android:layout_height="50dp"

```

```

        android:layout_marginBottom="60dp"
        android:text="Remember"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.384"
        app:layout_constraintStart_toEndOf="@+id/clear_button"
        tools:ignore="MissingConstraints" />

<Button
    android:id="@+id/play_button"
    android:layout_width="120dp"
    android:layout_height="50dp"
    android:layout_marginBottom="60dp"
    android:text="Play"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/clear_button"
    app:layout_constraintHorizontal_bias="0.64"
    app:layout_constraintStart_toStartOf="parent"
    tools:ignore="MissingConstraints" />

<SeekBar
    android:id="@+id/vertical_seek_bar"
    style="@style/Widget.AppCompat.SeekBar.Discrete"
    android:layout_width="441dp"
    android:layout_height="350dp"
    android:layout_marginEnd="156dp"
    android:layout_weight="1"
    android:max="100"
    android:progress="0"
    android:rotation="90"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.22"
    tools:ignore="MissingConstraints" />

<SeekBar
    android:id="@+id/horizontal_seek_bar"
    style="@style/Widget.AppCompat.SeekBar.Discrete"
    android:layout_width="350dp"
    android:layout_height="42dp"
    android:layout_marginBottom="16dp"
    android:max="90"
    android:min="10"
    android:progress="0"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.8"
    app:layout_constraintStart_toStartOf="parent"
    tools:ignore="MissingConstraints" />

<Button
    android:id="@+id/clear_button"

```



```
android:layout_width="120dp"  
android:layout_height="55dp"  
android:layout_marginBottom="56dp"  
android:text="Clear remember"  
app:layout_constraintBottom_toBottomOf="parent"  
app:layout_constraintEnd_toEndOf="parent"  
app:layout_constraintHorizontal_bias="0.498"  
app:layout_constraintStart_toStartOf="parent"  
tools:ignore="MissingConstraints" />
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

Работа приложения представлена на рисунке 1.1 – 1.2.

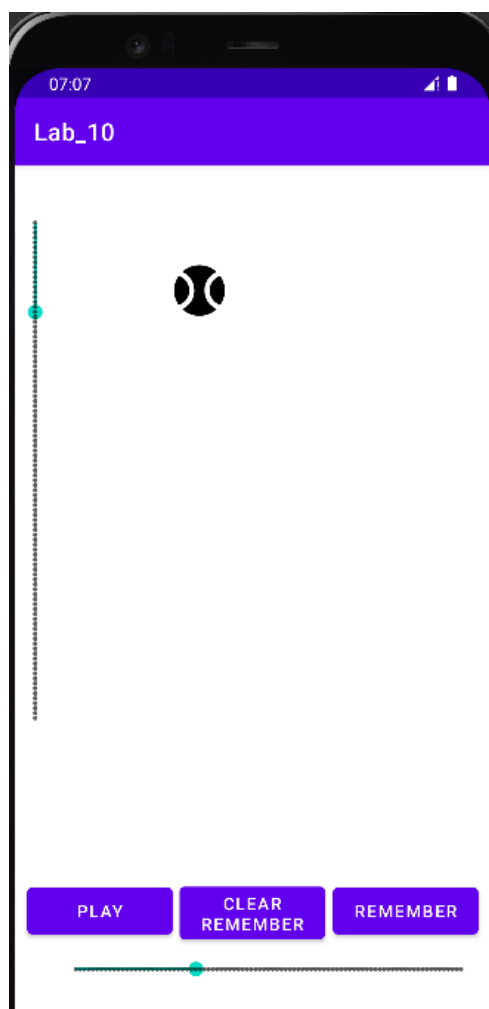


Рисунок 1.1 – Пример работы приложения

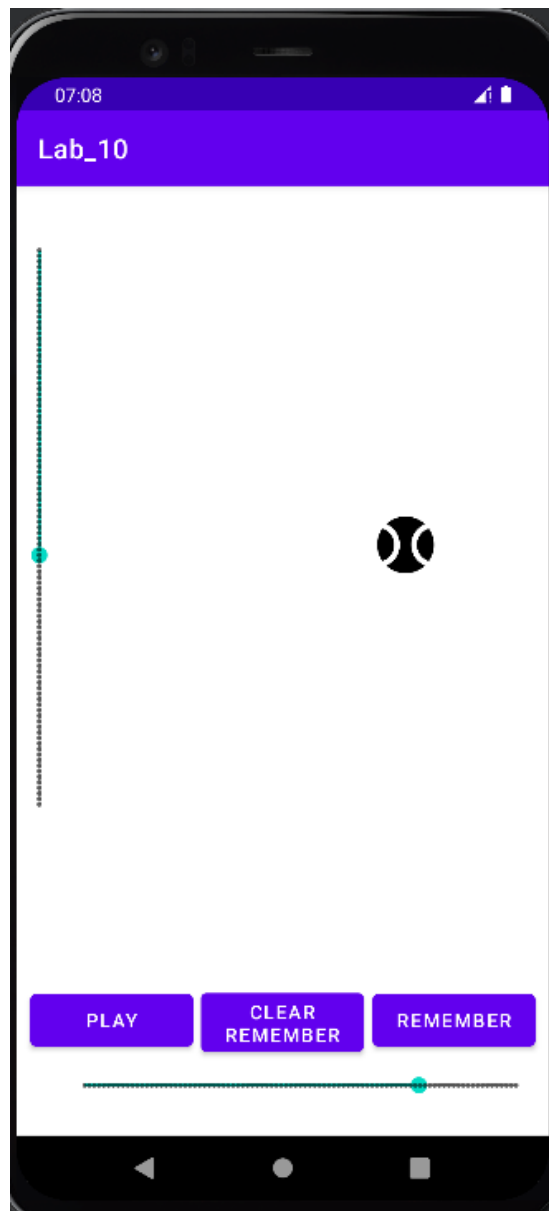


Рисунок 1.2 – Пример работы приложения

Список использованных источников

- 1 РД ФГБОУ ВО «КНАГТУ» 013-2016. Текстовые студенческие работы. Правила оформления. – Введ. 2016-03-10. – Комсомольск-на-Амуре: ФГБОУ ВО «КНАГТУ», 2016. – 55 с.