

Министерство образования Республики Беларусь
Учреждение образования
«Брестский государственный технический университет»
Кафедра ИИТ

Лабораторная работа №8
за 6 семестр

По дисциплине: «Разработка программного обеспечения для мобильных платформ»

Выполнил:
Студент 3 курса
Группы ПО-4(2)
Тупик Д. Л.
Проверил:
Козинский А. А.

Лабораторная работа № 8

Принципы работы с жестами вводимыми пользователями

Цель работы: разработать приложения, помогающего понять принципы работы с жестами вводимыми пользователями.

Задание:

- создать набор жестов
- использовать созданные жесты в приложении

Ход работы:

Интерфейс:

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"
    tools:context=".MainActivity">

    <android.gesture.GestureOverlayView
        android:id="@+id/gestureOverlayView"
        android:layout_width="match_parent"
        android:layout_height="match_parent">

    </android.gesture.GestureOverlayView>

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

MainActivity

```
package com.example.lab08;

import androidx.appcompat.app.AppCompatActivity;

import android.gesture.Gesture;
import android.gesture.GestureLibraries;
import android.gesture.GestureLibrary;
import android.gesture.GestureOverlayView;
import android.gesture.Prediction;
import android.os.Bundle;
import android.widget.TextView;
import android.widget.Toast;

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity implements
    GestureOverlayView.OnGesturePerformedListener {
    private GestureLibrary gLib;
    private GestureOverlayView gestures;

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

        this.gLib = GestureLibraries.fromRawResource(this, R.raw.gestures);
        if (!gLib.load()) {
            finish();
        }

        gestures = (GestureOverlayView) findViewById(R.id.gestureOverlayView);
        gestures.addOnGesturePerformedListener(this);
    }

    @Override
    public void onGesturePerformed(GestureOverlayView gestureOverlayView, Gesture gesture) {
        ArrayList<Prediction> predictions = gLib.recognize(gestures.getGesture());
        if (predictions.size() > 0) {
            Prediction prediction = predictions.get(0);
            if (prediction.score > 1.0) {
```

```

        if (prediction.name.equals("01") || prediction.name.equals("02"))
            Toast.makeText(getApplicationContext(), "0", Toast.LENGTH_SHORT).show();
        else if (prediction.name.equals("11") || prediction.name.equals("12"))
            Toast.makeText(getApplicationContext(), "1", Toast.LENGTH_SHORT).show();
        else if (prediction.name.equals("21") || prediction.name.equals("22"))
            Toast.makeText(getApplicationContext(), "2", Toast.LENGTH_SHORT).show();
        else if (prediction.name.equals("31"))
            Toast.makeText(getApplicationContext(), "3", Toast.LENGTH_SHORT).show();
        else if (prediction.name.equals("41"))
            Toast.makeText(getApplicationContext(), "4", Toast.LENGTH_SHORT).show();
        else if (prediction.name.equals("51"))
            Toast.makeText(getApplicationContext(), "5", Toast.LENGTH_SHORT).show();
        else if (prediction.name.equals("61"))
            Toast.makeText(getApplicationContext(), "6", Toast.LENGTH_SHORT).show();
        else if (prediction.name.equals("71"))
            Toast.makeText(getApplicationContext(), "7", Toast.LENGTH_SHORT).show();
        else if (prediction.name.equals("81"))
            Toast.makeText(getApplicationContext(), "8", Toast.LENGTH_SHORT).show();
        else if (prediction.name.equals("91"))
            Toast.makeText(getApplicationContext(), "9", Toast.LENGTH_SHORT).show();
        else if (prediction.name.equals("stop"))
            this.finishAffinity();
    } else {
        Toast.makeText(getApplicationContext(), "Жест неизвестен", Toast.LENGTH_SHORT).show();
    }
}
}
}

```

Результаты программы:



Вывод: в ходе лабораторной работы, разработал приложение, помогающее понять принципы работы с жестами вводимыми пользователями.