МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Лабораторная работа № 14

По дисциплине: «Разработка программного обеспечения для мобильных платформ»

Тема: «Работа с файловой системой. Разрешения.»

Выполнила: Велесевич В. А.

Группа: ПО-10

Проверил: Козинский А. А.

Брест 2025

**Цель:** реализовать приложение работающее с файлами внешнего хранилища.

**Постановка задачи:**

Создать приложение, обеспечивающее выбор файла во внешнем хранилище с возможностью дальнейшей его обработки в зависимости от расширения:

- графический файл отобразить с использованием элемента. ImageView

- аудиофайл воспроизвести с использованием элемента MediaPlayer;

- видеофайл воспроизвести с использованием элемента VideoView.

2. Загрузить заранее набор медиафайлов (изображения, аудио, видео) для тестирования.

3. Создать новый проект.

4. Добавить необходимые элементы интерфейса для реализации всех функций, перечисленных в задании. Для реализации различных функций можно использовать как дополнительные разметки, так и дополнительные активности. Простейший вариант оформления интерфейса приложения приведен в описании работы.

**Графическое изображение интерфейса:**

|  |  |
| --- | --- |
| Рис. 1 - Отображение списка элементов с помощью собственного адаптера | Рис. 2 - Отображение разрешений для доступа к музыке, аудио и изображениям |

**Код программы:**

**MainActivity**

package com.example.lab14;  
import android.Manifest;  
import android.content.Intent;  
import android.content.pm.PackageManager;  
import android.media.MediaPlayer;  
import android.net.Uri;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.ImageView;  
import android.widget.Toast;  
import android.widget.VideoView;  
import android.os.Build;  
import androidx.annotation.NonNull;  
import androidx.annotation.Nullable;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
import androidx.core.content.ContextCompat;  
  
public class MainActivity extends AppCompatActivity {  
  
 private static final int PICK\_MUSIC\_REQUEST = 1;  
 private static final int PICK\_VIDEO\_REQUEST = 2;  
 private static final int PICK\_IMAGE\_REQUEST = 3;  
 private static final int REQUEST\_CODE\_PERMISSIONS = 100;  
  
 private ImageView imageView;  
 private VideoView videoView;  
 private Button btnPlay, btnPause, btnStop;  
 private MediaPlayer mediaPlayer;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 imageView = findViewById(R.id.imageView);  
 videoView = findViewById(R.id.videoView);  
 btnPlay = findViewById(R.id.btnPlay);  
 btnPause = findViewById(R.id.btnPause);  
 btnStop = findViewById(R.id.btnStop);  
  
 Button btnChooseMusic = findViewById(R.id.btnChooseMusic);  
 Button btnChooseVideo = findViewById(R.id.btnChooseVideo);  
 Button btnChooseImage = findViewById(R.id.btnChooseImage);  
  
 btnChooseMusic.setOnClickListener(v -> openFileChooser("audio/\*", PICK\_MUSIC\_REQUEST));  
 btnChooseVideo.setOnClickListener(v -> openFileChooser("video/\*", PICK\_VIDEO\_REQUEST));  
 btnChooseImage.setOnClickListener(v -> openFileChooser("image/\*", PICK\_IMAGE\_REQUEST));  
  
 // Check for permissions  
 checkAndRequestPermissions();  
 }  
  
 private void checkAndRequestPermissions() {  
 if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.TIRAMISU) {  
 ActivityCompat.requestPermissions(this, new String[]{  
 Manifest.permission.READ\_MEDIA\_IMAGES,  
 Manifest.permission.READ\_MEDIA\_VIDEO,  
 Manifest.permission.READ\_MEDIA\_AUDIO  
 }, REQUEST\_CODE\_PERMISSIONS);  
 } else {  
 ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.READ\_EXTERNAL\_STORAGE}, REQUEST\_CODE\_PERMISSIONS);  
 }  
 }  
  
 @Override  
 public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {  
 super.onRequestPermissionsResult(requestCode, permissions, grantResults);  
 // Handle permissions results if needed  
 }  
  
 private void openFileChooser(String mimeType, int requestCode) {  
 if (requestCode == PICK\_MUSIC\_REQUEST && !checkAudioPermission()) {  
 Toast.makeText(this, "Audio permission not granted. Please allow access to audio.", Toast.LENGTH\_SHORT).show();  
 return;  
 }  
 if (requestCode == PICK\_VIDEO\_REQUEST && !checkVideoPermission()) {  
 Toast.makeText(this, "Video permission not granted. Please allow access to video.", Toast.LENGTH\_SHORT).show();  
 return;  
 }  
 if (requestCode == PICK\_IMAGE\_REQUEST && !checkImagePermission()) {  
 Toast.makeText(this, "Image permission not granted. Please allow access to images.", Toast.LENGTH\_SHORT).show();  
 return;  
 }  
  
 Intent intent = new Intent(Intent.ACTION\_GET\_CONTENT);  
 intent.setType(mimeType);  
 startActivityForResult(intent, requestCode);  
 }  
  
 private boolean checkAudioPermission() {  
 if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.TIRAMISU) {  
 return ContextCompat.checkSelfPermission(this, Manifest.permission.READ\_MEDIA\_AUDIO) == PackageManager.PERMISSION\_GRANTED;  
 }  
 return ContextCompat.checkSelfPermission(this, Manifest.permission.READ\_EXTERNAL\_STORAGE) == PackageManager.PERMISSION\_GRANTED;  
 }  
  
 private boolean checkVideoPermission() {  
 if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.TIRAMISU) {  
 return ContextCompat.checkSelfPermission(this, Manifest.permission.READ\_MEDIA\_VIDEO) == PackageManager.PERMISSION\_GRANTED;  
 }  
 return ContextCompat.checkSelfPermission(this, Manifest.permission.READ\_EXTERNAL\_STORAGE) == PackageManager.PERMISSION\_GRANTED;  
 }  
  
 private boolean checkImagePermission() {  
 if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.TIRAMISU) {  
 return ContextCompat.checkSelfPermission(this, Manifest.permission.READ\_MEDIA\_IMAGES) == PackageManager.PERMISSION\_GRANTED;  
 }  
 return ContextCompat.checkSelfPermission(this, Manifest.permission.READ\_EXTERNAL\_STORAGE) == PackageManager.PERMISSION\_GRANTED;  
 }  
  
 @Override  
 protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {  
 super.onActivityResult(requestCode, resultCode, data);  
 if (resultCode == RESULT\_OK && data != null) {  
 Uri uri = data.getData();  
 if (requestCode == PICK\_MUSIC\_REQUEST) {  
 playAudio(uri);  
 } else if (requestCode == PICK\_VIDEO\_REQUEST) {  
 playVideo(uri);  
 } else if (requestCode == PICK\_IMAGE\_REQUEST) {  
 displayImage(uri);  
 }  
 }  
 }  
  
 private void displayImage(Uri uri) {  
 imageView.setVisibility(View.VISIBLE);  
 videoView.setVisibility(View.GONE);  
 btnPlay.setVisibility(View.VISIBLE);  
 btnPause.setVisibility(View.VISIBLE);  
 btnStop.setVisibility(View.VISIBLE);  
  
 Intent intent = new Intent(MainActivity.this, ImageDisplayActivity.class);  
 intent.putExtra("imageUri", uri);  
 startActivity(intent);  
 }  
  
 private void playAudio(Uri uri) {  
 imageView.setVisibility(View.GONE);  
 videoView.setVisibility(View.GONE);  
 btnPlay.setVisibility(View.VISIBLE);  
 btnPause.setVisibility(View.VISIBLE);  
 btnStop.setVisibility(View.VISIBLE);  
  
 if (mediaPlayer != null) {  
 mediaPlayer.release();  
 }  
 mediaPlayer = MediaPlayer.create(this, uri);  
  
 btnPlay.setOnClickListener(v -> mediaPlayer.start());  
 btnPause.setOnClickListener(v -> mediaPlayer.pause());  
 btnStop.setOnClickListener(v -> {  
 mediaPlayer.stop();  
 mediaPlayer.release();  
 mediaPlayer = null;  
 });  
 }  
  
 private void playVideo(Uri uri) {  
 imageView.setVisibility(View.GONE);  
 videoView.setVisibility(View.VISIBLE);  
 btnPlay.setVisibility(View.VISIBLE);  
 btnPause.setVisibility(View.VISIBLE);  
 btnStop.setVisibility(View.VISIBLE);  
  
 videoView.setVideoURI(uri);  
  
 btnPlay.setOnClickListener(v -> videoView.start());  
 btnPause.setOnClickListener(v -> videoView.pause());  
 btnStop.setOnClickListener(v -> videoView.stopPlayback());  
 }  
}

**ImageDisplayActivity**

package com.example.lab14;  
  
import android.net.Uri;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.ImageView;  
import androidx.annotation.Nullable;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class ImageDisplayActivity extends AppCompatActivity {  
 private ImageView imageView;  
 private Button btnBackToMain;  
  
 @Override  
 protected void onCreate(@Nullable Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_image\_display); // Убедитесь, что используется правильный layout  
  
 imageView = findViewById(R.id.imageViewFull);  
 btnBackToMain = findViewById(R.id.btnBackToMain); // Находим кнопку  
  
 // Получаем URI изображения из Intent  
 Uri imageUri = getIntent().getParcelableExtra("imageUri");  
 if (imageUri != null) {  
 imageView.setImageURI(imageUri);  
 }  
  
 // Обработчик кнопки "Вернуться в главное меню"  
 btnBackToMain.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 // Закрываем текущую активность и возвращаемся в MainActivity  
 finish();  
 }  
 });  
 }  
}

**Вывод:** реализовала интерфейс приложения для отображения списка элементов.