МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

імені ІГОРЯ СІКОРСЬКОГО»

ЛАБОРАТОРНА РОБОТА №4

з дисципліни «Розробка мобільних застосувань під Android»

на тему: «Дослідження способів роботи з медіаданими»

|  |  |
| --- | --- |
| Виконав: | Перевірив: |
| Студент групи ІС-22 | асистент, канд. техн. наук |
| Прокопішин Олексій | Орленко С. П. |

Київ 2025

**Зміст**

[**Зміст** 2](#_Toc195534448)

[**Завдання** 3](#_Toc195534449)

[**Лістинг коду** 4](#_Toc195534450)

[activity\_main.xml 4](#_Toc195534451)

[MainActivity.kt 7](#_Toc195534452)

[**Скріншоти виконання коду** 15](#_Toc195534453)

[**Контрольні питання** 21](#_Toc195534454)

# **Завдання**

Написати програму під платформу Андроїд, яка має інтерфейс для запуску аудіо-файлів та відео-файлів. Мінімально інтерфейс має надавати можливість Програвати/Зупиняти/Призупиняти відтворення відео-файлу або аудіо-файлу, який зберігається у внутрішньому сховищі.

Функціональність базового додатку додатково розширюється наступними можливостями:

* надати вибір типу файлу для відтворення (аудіо або відео) з будь-якого сховища на мобільному пристрої;
* надати вибір завантаження файлу з Інтернету;
* використовувати для реалізації обробки медіа-даних спеціалізовані інструменти (особливу увагу приділити програванню відео).

Примітка: конкретних вимог до дизайну та вибору інструментів для виконання лабораторної роботи не передбачено, студент сам формує вигляд програми.

# **Лістинг коду**

## activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout

xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:padding="16dp">

<VideoView

android:id="@+id/video\_view"

android:layout\_width="match\_parent"

android:layout\_height="200dp"

android:visibility="visible"/>

<RadioGroup

android:id="@+id/radioGroup"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:padding="16dp">

<RadioButton

android:id="@+id/radioAudio"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Аудіо"/>

<RadioButton

android:id="@+id/radioVideo"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Відео"

android:checked="true"/>

</RadioGroup>

<Button

android:id="@+id/btnChooseFile"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Вибрати файл з пристрою"/>

<Button

android:id="@+id/btnLoadUrl"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Завантажити файл з Інтернету"/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:gravity="center"

android:padding="16dp">

<Button

android:id="@+id/btnPlay"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Програти"/>

<Button

android:id="@+id/btnPause"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Призупинити"

android:layout\_marginStart="16dp"/>

<Button

android:id="@+id/btnStop"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Зупинити"

android:layout\_marginStart="16dp"/>

</LinearLayout>

</LinearLayout>

## MainActivity.kt

package com.example.lab4

import android.content.Context

import android.media.MediaPlayer

import android.net.Uri

import android.os.Bundle

import android.widget.\*

import androidx.activity.result.ActivityResultLauncher

import androidx.activity.result.contract.ActivityResultContracts

import androidx.appcompat.app.AlertDialog

import androidx.appcompat.app.AppCompatActivity

import androidx.core.net.toUri

interface MediaControllerInterface {

fun play(uri: Uri)

fun pause()

fun stop()

}

class VideoPlayerController(

private val videoView: VideoView,

private val context: Context

) : MediaControllerInterface {

private var currentUri: Uri? = null

override fun play(uri: Uri) {

if (currentUri == null || videoView.duration <= 0) {

currentUri = uri

videoView.setVideoURI(uri)

val mediaController = MediaController(context)

mediaController.setAnchorView(videoView)

videoView.setMediaController(mediaController)

}

videoView.start()

}

override fun pause() {

videoView.pause()

}

override fun stop() {

videoView.stopPlayback()

currentUri = null

}

}

class AudioPlayerController(private val context: Context) : MediaControllerInterface {

private var mediaPlayer: MediaPlayer? = null

private var currentUri: Uri? = null

override fun play(uri: Uri) {

if (mediaPlayer == null) {

currentUri = uri

mediaPlayer = MediaPlayer().apply {

setDataSource(context, uri)

prepare()

start()

}

} else {

try {

mediaPlayer?.start()

} catch (e: IllegalStateException) {

stop()

play(uri)

}

}

}

override fun pause() {

mediaPlayer?.pause()

}

override fun stop() {

mediaPlayer?.let {

try {

it.stop()

} catch (e: IllegalStateException) { }

it.reset()

it.release()

}

mediaPlayer = null

currentUri = null

}

}

class MainActivity : AppCompatActivity() {

private lateinit var videoView: VideoView

private lateinit var btnChooseFile: Button

private lateinit var btnLoadUrl: Button

private lateinit var btnPlay: Button

private lateinit var btnPause: Button

private lateinit var btnStop: Button

private lateinit var radioGroup: RadioGroup

private lateinit var radioAudio: RadioButton

private lateinit var radioVideo: RadioButton

private var currentUri: Uri? = null

private var isVideoFile: Boolean = true

private lateinit var filePickerLauncher: ActivityResultLauncher<String>

private lateinit var videoController: VideoPlayerController

private lateinit var audioController: AudioPlayerController

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

videoView = findViewById(R.id.video\_view)

btnChooseFile = findViewById(R.id.btnChooseFile)

btnLoadUrl = findViewById(R.id.btnLoadUrl)

btnPlay = findViewById(R.id.btnPlay)

btnPause = findViewById(R.id.btnPause)

btnStop = findViewById(R.id.btnStop)

radioGroup = findViewById(R.id.radioGroup)

radioAudio = findViewById(R.id.radioAudio)

radioVideo = findViewById(R.id.radioVideo)

videoController = VideoPlayerController(videoView, this)

audioController = AudioPlayerController(this)

filePickerLauncher = registerForActivityResult(ActivityResultContracts.GetContent()) { uri: Uri? ->

uri?.let {

currentUri = it

Toast.makeText(this, "Файл обрано", Toast.LENGTH\_SHORT).show()

}

}

radioGroup.setOnCheckedChangeListener { \_, checkedId ->

isVideoFile = (checkedId == R.id.radioVideo)

videoView.visibility = if (isVideoFile) VideoView.VISIBLE else VideoView.GONE

if (!isVideoFile && videoView.isPlaying) {

videoController.stop()

}

}

btnChooseFile.setOnClickListener {

val mimeType = if (isVideoFile) "video/\*" else "audio/\*"

filePickerLauncher.launch(mimeType)

}

btnLoadUrl.setOnClickListener {

showUrlInputDialog()

}

btnPlay.setOnClickListener {

currentUri?.let { uri ->

if (isVideoFile) {

if (videoView.duration > 0) {

videoView.start()

} else {

videoController.play(uri)

}

} else {

audioController.play(uri)

}

} ?: Toast.makeText(this, "Файл не обрано", Toast.LENGTH\_SHORT).show()

}

btnPause.setOnClickListener {

if (isVideoFile) {

videoController.pause()

} else {

audioController.pause()

}

}

btnStop.setOnClickListener {

if (isVideoFile) {

if (currentUri != null) {

videoController.stop()

currentUri = null

}

} else {

if (currentUri != null) {

audioController.stop()

currentUri = null

}

}

}

}

private fun showUrlInputDialog() {

val input = EditText(this).apply { hint = "Введіть URL файлу" }

AlertDialog.Builder(this)

.setTitle("Завантаження файлу з Інтернету")

.setView(input)

.setPositiveButton("Завантажити") { dialog, \_ ->

val urlText = input.text.toString()

if (urlText.isNotEmpty()) {

currentUri = urlText.toUri()

if (isVideoFile) {

videoController.play(currentUri!!)

} else {

audioController.play(currentUri!!)

}

} else {

Toast.makeText(this, "URL не може бути порожнім", Toast.LENGTH\_SHORT).show()

}

dialog.dismiss()

}

.setNegativeButton("Скасувати") { dialog, \_ -> dialog.dismiss() }

.show()

}

override fun onDestroy() {

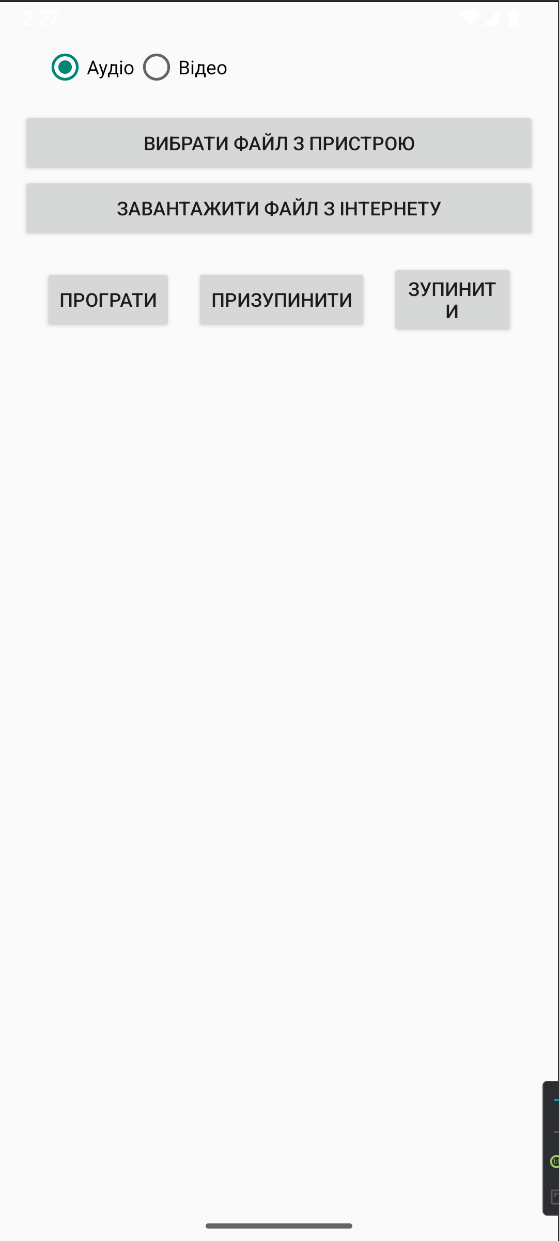
super.onDestroy()

audioController.stop()

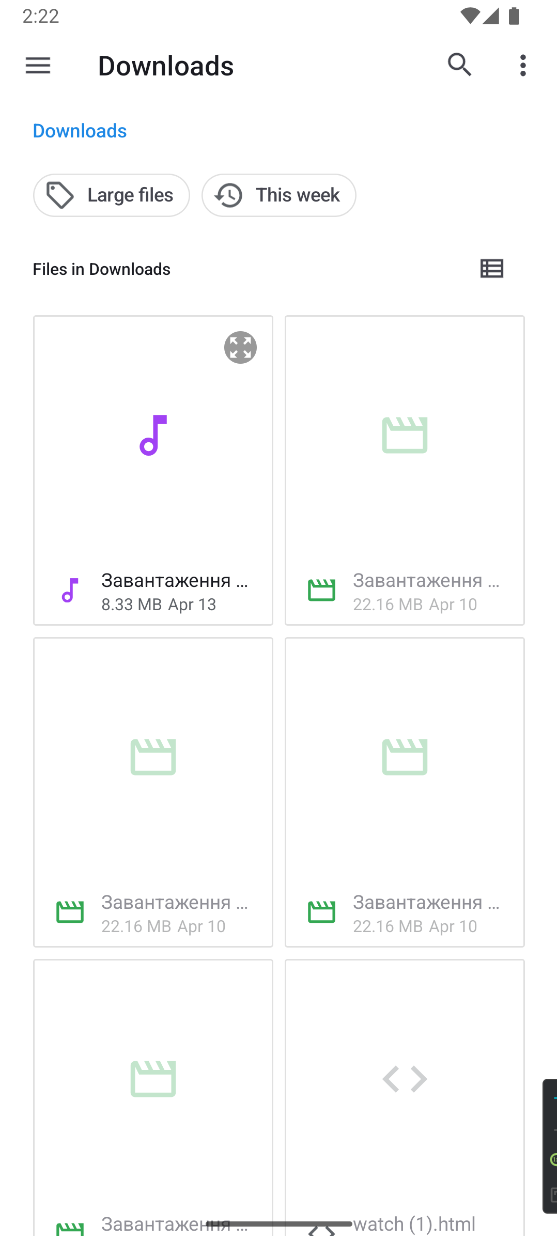
}

}

# **Скріншоти виконання коду**



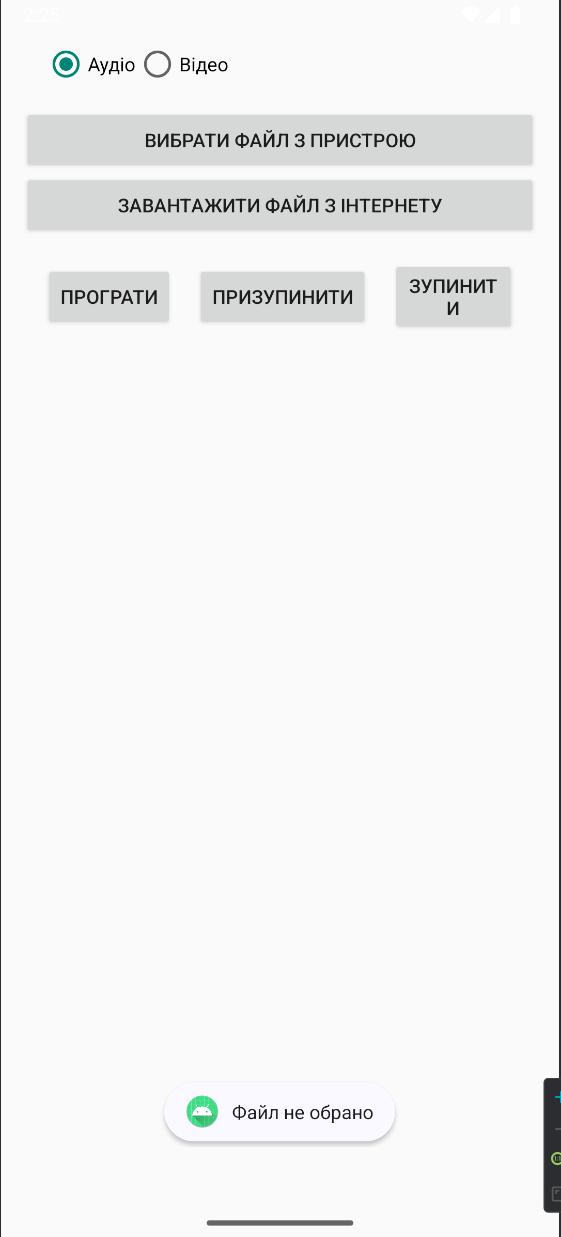
Натискаємо на кнопку Вибрати файл з пристрою:



Обираємо трек, тепер у нас є можливість його програвати, ставити на паузу або взагалі, зупиняти.



Коли після зупинки прогшравання або без обраного файлу ми натискаємо на програвання, то висвічується таке вікно:



Тепер переходимо до відео:



Повторюємо ті ж самі дії та запускаємо відео:



# **Контрольні питання**

1. *Наведіть способи підключення Інтернет ресурсів до мобільного застосунку.*

Інтернет-ресурси до мобільного застосунку підключаються за допомогою HTTP/HTTPS-запитів через бібліотеки, такі як Retrofit або OkHttp, або через WebView для відображення веб-сторінок. Також використовуються WebSockets для обміну даними в реальному часі.

1. *Поясніть різницю між внутрішнім та зовнішнім сховищем.*

Внутрішнє сховище — це захищена пам’ять, доступна лише самому застосунку. Зовнішнє сховище — це спільна пам’ять пристрою або SD-карта, доступна іншим застосункам і користувачу.

1. *Наведіть категорії файлів при збереженні в зовнішньому сховищі.*

Категорії файлів у зовнішньому сховищі включають документи, зображення, відео, музику, завантаження, кеш і власні каталоги застосунків. Вони організовані в системні папки, наприклад, Downloads, Pictures, Movies, Music.

1. *Опишіть властивості спеціалізованих інструментів для відтворення аудіо-файлів.*

Спеціалізовані інструменти для аудіо, такі як MediaPlayer або ExoPlayer, забезпечують підтримку різних форматів, потокове відтворення та керування станами (пауза, зупинка, циклічне програвання). Вони дозволяють обробляти локальні та онлайн-аудіофайли з можливістю контролю гучності та позиції.

1. *Опишіть властивості спеціалізованих інструментів для відтворення відео-файлів.*

Для відео-файлів використовуються такі інструменти, як VideoView, ExoPlayer або Media3, які підтримують апаратне прискорення, адаптивне стрімінг-відтворення, субтитри та взаємодію з візуальними елементами інтерфейсу. Вони забезпечують стабільне програвання з різних джерел та контроль над якістю і буферизацією.