



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота №4
Розробка мобільних застосунків під Android
«ДОСЛІДЖЕННЯ СПОСОБІВ РОБОТИ З МЕДІАДАНИМИ»
Варіант 6

Виконала:
студентка групи ІА-21
Антропова К.Д.

Київ 2025

Мета роботи: дослідити яким чином платформа Андроїд надає можливість оброблювати аудіо-файли та відео-файли та отримати практичні навички щодо використання інструментів відтворення медіа-даних.

Завдання на лабораторну роботу

БАЗОВЕ (12/20 балів). 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.

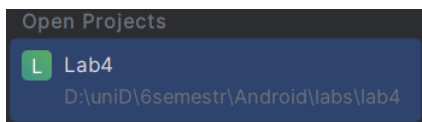
ПОВНЕ (20/20). Функціональність базового додатку додатково розширюється наступними можливостями:

- 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.

Хід роботи

1. Створюємо проект



2. В класі MainActivity пишемо такий код:

```
@file:Suppress("DEPRECATION")

package com.example.lab4

import android.Manifest
import android.content.Intent
import android.content.pm.PackageManager
import android.net.Uri
import android.os.Bundle
import android.widget.Button
import android.widget.Toast
import androidx.activity.result.contract.ActivityResultContracts
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat

class MainActivity : AppCompatActivity() {
    private val STORAGE_PERMISSION_CODE = 101
```

```

        private val pickAudioLauncher =
registerForActivityResult (ActivityResultContracts.GetContent()) { uri: Uri? -
>
        uri?.let {
            val intent = Intent(this, AudioPlayerActivity::class.java)
            intent.putExtra("uri", it.toString())
            startActivity(intent)
        }
    }

    private val pickVideoLauncher =
registerForActivityResult (ActivityResultContracts.GetContent()) { uri: Uri? -
>
        uri?.let {
            val intent = Intent(this, VideoPlayerActivity::class.java)
            intent.putExtra("uri", it.toString())
            startActivity(intent)
        }
    }

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        checkPermissions()

        val playLocalAudioButton: Button =
findViewById(R.id.playLocalAudioButton)
        playLocalAudioButton.setOnClickListener {
            pickAudioLauncher.launch("audio/*")
        }

        val playLocalVideoButton: Button =
findViewById(R.id.playLocalVideoButton)
        playLocalVideoButton.setOnClickListener {
            pickVideoLauncher.launch("video/*")
        }

        val downloadMediaButton: Button =
findViewById(R.id.downloadMediaButton)
        downloadMediaButton.setOnClickListener {
            val intent = Intent(this, DownloadActivity::class.java)
            startActivity(intent)
        }
    }

    private fun checkPermissions() {
        if (ContextCompat.checkSelfPermission(this,
Manifest.permission.READ_EXTERNAL_STORAGE)
            != PackageManager.PERMISSION_GRANTED) {
            ActivityCompat.requestPermissions(
                this,
                arrayOf(Manifest.permission.READ_EXTERNAL_STORAGE),
                STORAGE_PERMISSION_CODE
            )
        }
    }

    override fun onRequestPermissionsResult(
        requestCode: Int,
        permissions: Array<out String>,
        grantResults: IntArray
    ) {
        super.onRequestPermissionsResult(requestCode, permissions,

```

```

grantResults)
    if (requestCode == STORAGE_PERMISSION_CODE) {
        if (grantResults.isNotEmpty() && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
            Toast.makeText(this, "Дозвіл надано",
Toast.LENGTH_SHORT).show()
        } else {
            Toast.makeText(this, "Дозвіл відхилено",
Toast.LENGTH_SHORT).show()
        }
    }
}
}
}

```

3. В класі DownloadActivity пишемо такий код:

```

package com.example.lab4

import android.annotation.SuppressLint
import android.app.DownloadManager
import android.content.BroadcastReceiver
import android.content.Context
import android.content.Intent
import android.content.IntentFilter
import android.net.Uri
import android.os.Bundle
import android.os.Environment
import android.util.Log
import android.widget.Button
import android.widget.EditText
import android.widget.RadioButton
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.core.content.ContextCompat
import java.io.File

class DownloadActivity : AppCompatActivity() {
    private lateinit var urlEditText: EditText
    private lateinit var downloadButton: Button
    private lateinit var audioRadioButton: RadioButton
    private lateinit var videoRadioButton: RadioButton
    private var downloadID: Long = 0

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_download)

        urlEditText = findViewById(R.id.urlEditText)
        downloadButton = findViewById(R.id.downloadButton)
        audioRadioButton = findViewById(R.id.audioRadioButton)
        videoRadioButton = findViewById(R.id.videoRadioButton)

        audioRadioButton.isChecked = true

        downloadButton.setOnClickListener {
            val url = urlEditText.text.toString()
            if (url.isNotEmpty()) {
                Log.d("DownloadActivity", "Starting download for URL: $url")
                downloadFile(url)
            } else {
                Toast.makeText(this, "Будь ласка, введіть URL",
Toast.LENGTH_SHORT).show()
            }
        }
    }
}

```

```

    }

    }

    val filter = IntentFilter(DownloadManager.ACTION_DOWNLOAD_COMPLETE)
    ContextCompat.registerReceiver(
        this,
        onDownloadComplete,
        filter,
        ContextCompat.RECEIVER_EXPORTED
    )
}

private fun downloadFile(url: String) {
    val request = DownloadManager.Request(Uri.parse(url))
        .setTitle("Завантаження медіа")
        .setDescription("Завантаження файлу...")

    .setNotificationVisibility(DownloadManager.Request.VISIBILITY_VISIBLE_NOTIFY_COMPLETED)
        .setDestinationInExternalFilesDir(this,
            Environment.DIRECTORY_DOWNLOADS, "downloaded_media.mp4")
        .setAllowedOverMetered(true)
        .setAllowedOverRoaming(true)

    request.addRequestHeader("User-Agent", "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36")

    val downloadManager = getSystemService(Context.DOWNLOAD_SERVICE) as DownloadManager
    downloadID = downloadManager.enqueue(request)
    Log.d("DownloadActivity", "Download started with ID: $downloadID")
}

private val onDownloadComplete = object : BroadcastReceiver() {
    @SuppressWarnings("Range")
    override fun onReceive(context: Context, intent: Intent) {
        val id = intent.getLongExtra(DownloadManager.EXTRA_DOWNLOAD_ID, -1)

        if (downloadID == id) {
            Log.d("DownloadActivity", "Download completed for ID: $id")
            val downloadManager =
                getSystemService(Context.DOWNLOAD_SERVICE) as DownloadManager
            val query = DownloadManager.Query().setFilterById(downloadID)
            val cursor = downloadManager.query(query)

            if (cursor.moveToFirst()) {
                val columnIndex =
                    cursor.getColumnIndex(DownloadManager.COLUMN_STATUS)
                val status = cursor.getInt(columnIndex)

                if (status == DownloadManager.STATUS_SUCCESSFUL) {
                    val uriString =
                        cursor.getString(cursor.getColumnIndex(DownloadManager.COLUMN_LOCAL_URI))
                    val uri = Uri.parse(uriString)
                    Log.d("DownloadActivity", "Downloaded URI: $uriString")

                    val file = File(uri.path ?: "")
                    if (file.exists()) {
                        val intent = if (audioRadioButton.isChecked) {
                            Intent(this@DownloadActivity,
                                AudioPlayerActivity::class.java)
                        } else {

```

```

Intent(this@DownloadActivity,
VideoPlayerActivity::class.java)
    }
    intent.putExtra("uri", uri.toString())
    startActivity(intent)
    } else {
        Toast.makeText(context, "Файл не знайдено:
$uriString", Toast.LENGTH_LONG).show()
    }
    } else {
        val reasonIndex =
cursor.getColumnIndex(DownloadManager.COLUMN_REASON)
        val reason = cursor.getInt(reasonIndex)
        Log.e("DownloadActivity", "Download failed with
status: $status, reason: $reason")
        Toast.makeText(context, "Завантаження не вдалося.
Причина: $reason", Toast.LENGTH_LONG).show()
    }
    } else {
        Log.e("DownloadActivity", "Cursor is empty, no download
info found")
        Toast.makeText(context, "Завантаження не вдалося:
інформація відсутня", Toast.LENGTH_LONG).show()
    }
    cursor.close()
    }
    }
}

override fun onDestroy() {
    super.onDestroy()
    unregisterReceiver(onDownloadComplete)
}
}

```

4. В класі VideoPlayerActivity пишемо такий код:

```

package com.example.lab4

import android.net.Uri
import android.os.Bundle
import android.widget.Button
import android.widget.MediaController
import android.widget.Toast
import android.widget.VideoView
import androidx.appcompat.app.AppCompatActivity

class VideoPlayerActivity : AppCompatActivity() {
    private lateinit var videoView: VideoView
    private lateinit var playButton: Button
    private lateinit var pauseButton: Button
    private lateinit var stopButton: Button

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_video_player)

        videoView = findViewById(R.id.videoView)
        playButton = findViewById(R.id.playButton)
        pauseButton = findViewById(R.id.pauseButton)
        stopButton = findViewById(R.id.stopButton)
    }
}

```

```

        val uriString = intent.getStringExtra("uri")
        if (uriString == null) {
            Toast.makeText(this, "URI не передано",
                Toast.LENGTH_SHORT).show()
            finish()
            return
        }

        val uri = Uri.parse(uriString)

        videoView.setVideoURI(uri)
        val mediaController = MediaController(this)
        mediaController.setAnchorView(videoView)
        videoView.setMediaController(mediaController)

        playButton.setOnClickListener {
            if (!videoView.isPlaying) {
                videoView.start()
            }
        }

        pauseButton.setOnClickListener {
            if (videoView.isPlaying) {
                videoView.pause()
            }
        }

        stopButton.setOnClickListener {
            if (videoView.isPlaying) {
                videoView.stopPlayback()
                videoView.setVideoURI(uri) // Перезавантаження відео
            }
        }
    }
}

```

5. Додаємо layout файли з потрібним кодом:

```

▼ layout
    </> activity_audio_player.xml
    </> activity_download.xml
    </> activity_main.xml
    </> activity_video_player.xml

```

6. Додаємо необхідний код в файлах в директорії values:

```

▼ values
    </> colors.xml
    </> strings.xml
    </> themes.xml

```

7. В AndroidManifest.xml додаємо:

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.lab4">

    <uses-permission android:name="android.permission.INTERNET" />

```

```

<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"
/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"
android:maxSdkVersion="28" />

<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.Lab4">
    <activity android:name=".MainActivity"
        android:exported="true">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity android:name=".AudioPlayerActivity" />
    <activity android:name=".VideoPlayerActivity" />
    <activity android:name=".DownloadActivity" />
</application>

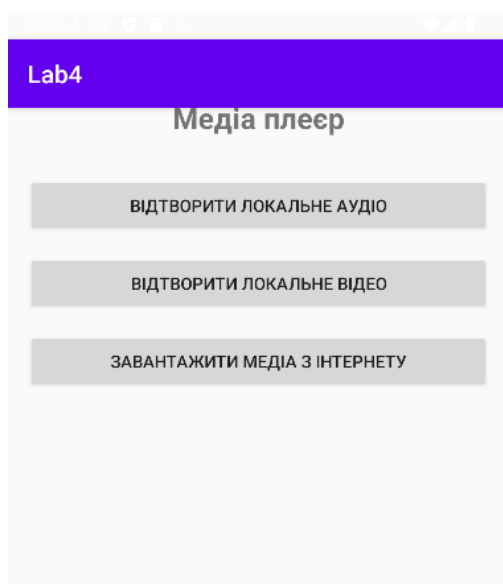
</manifest>

```

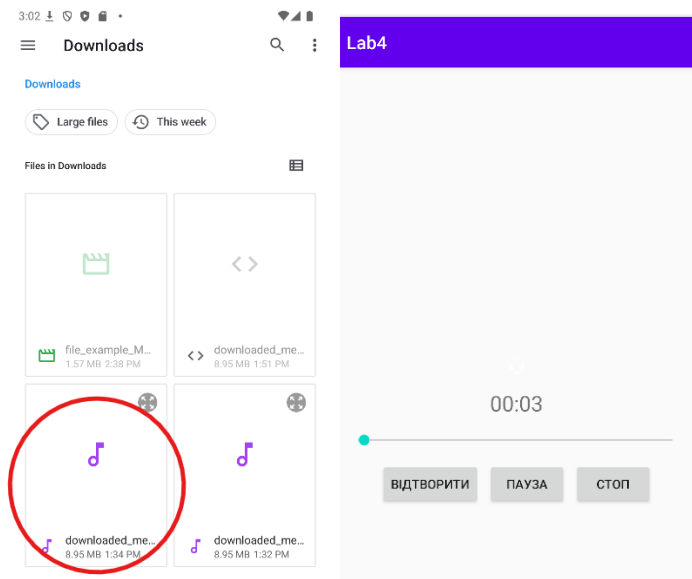
Також вносимо необхідні зміни в build.gradle.kts та gradle.properties.

Результат:

Заходимо в програму і бачимо вказані у завданні компоненти програми, а саме кнопку відтворення локального аудіо, кнопку відтворення локального відео та кнопку завантаження медіа з інтернету:

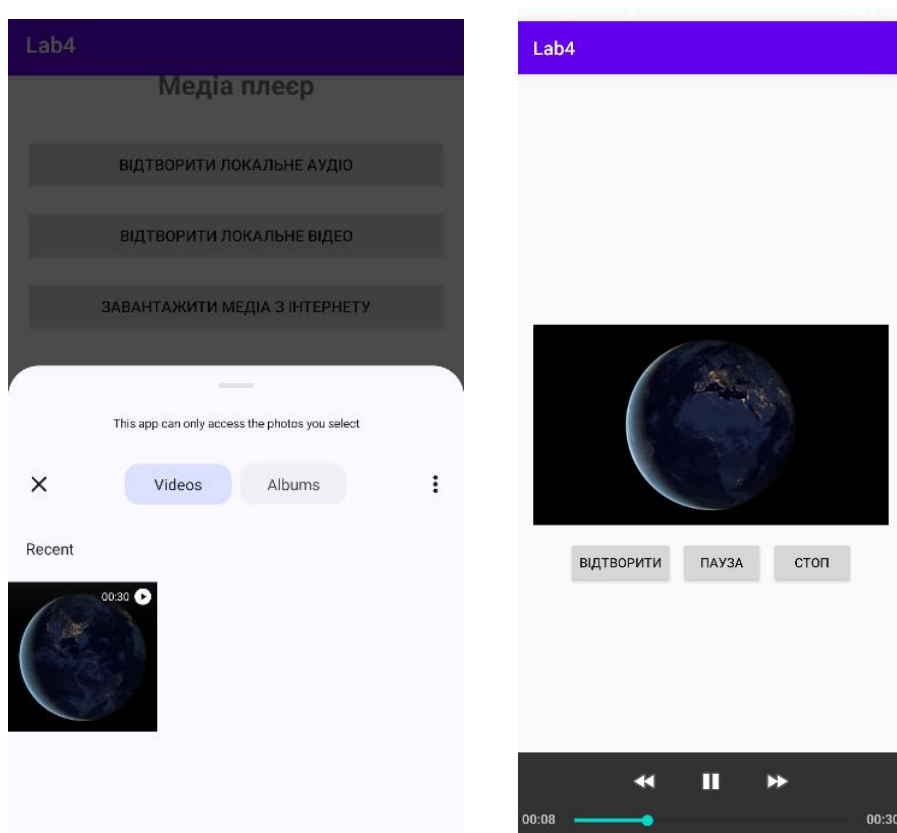


При натисканні кнопки відтворення локального аудіо відкривається каталог локальних файлів, обираємо необхідний, і після цього відкривається плеєр:



При натисненні кнопки «Відтворити» в плеєрі аудіо починає відтворюватись, «Пауза» - зупиняється, «Стоп» - скидається на початок (00:00).

При натисканні кнопки відтворення локального відео відкривається каталог локальних файлів, обираємо необхідний, і після цього відкривається плеєр:

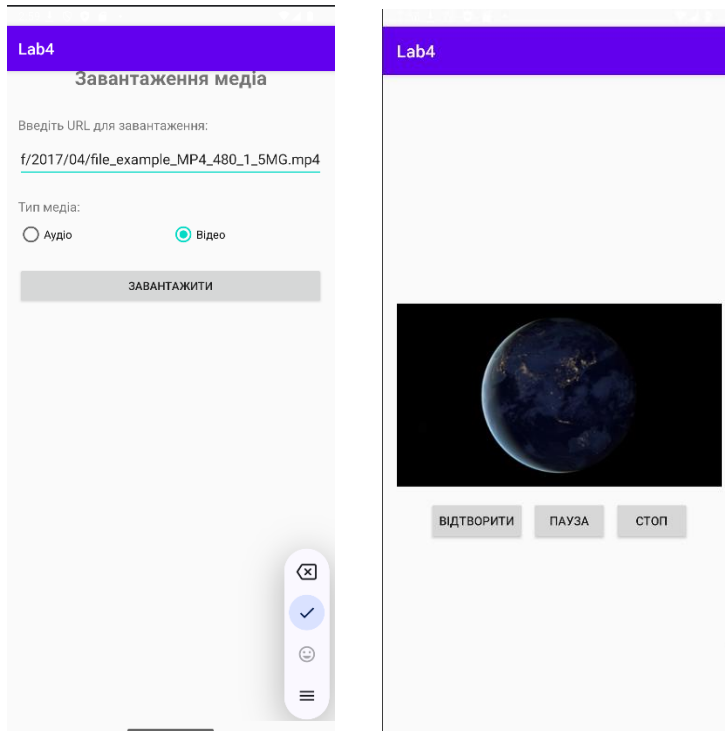


При натисненні кнопки «Відтворити» в плеєрі відео починає відтворюватись, «Пауза» - зупиняється, «Стоп» - скидається на початок (00:00).

При натисканні кнопки завантаження медіа з інтернету відкривається вікно з полем для введення посилання для завантаження, радіо-кнопками для вибору типу завантажуваного файлу та кнопкою завантажити. Після натискання кнопки завантажити оп завершенню завантаження відкривається плеєр.

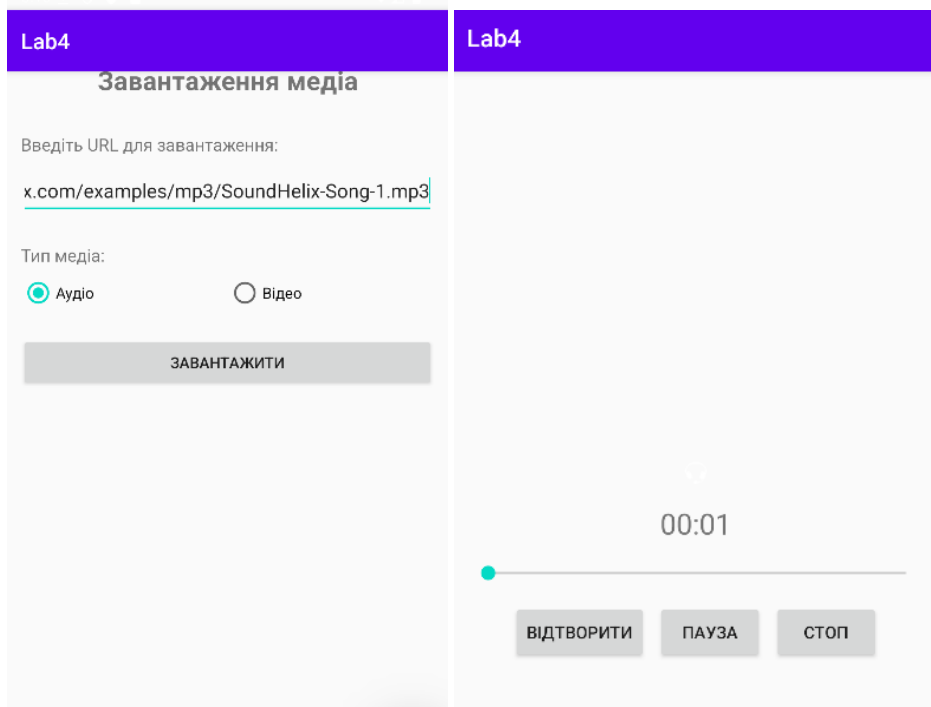
Приклад для відео:

https://file-examples.com/storage/fed00909ee67dc5db96303f/2017/04/file_example_MP4_480_1_5MG.mp4



Приклад для аудіо:

<https://www.soundhelix.com/examples/mp3/SoundHelix-Song-1.mp3>



Отже, програма працює відповідно до вимог завдання.

Висновок : У ході виконання цієї лабораторної роботи я дослідила яким чином платформа Андроїд надає можливість оброблювати аудіо-файли та відео-файли та отримати практичні навички щодо використання інструментів відтворення медіа-даних, створила застосунок відповідно до вимог та отримала відповідні практичні навички.