

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ПОЛІТЕХНІКА»
ІНСТИТУТ КОМП'ЮТЕРНИХ СИСТЕМ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ

Лабораторна робота №3

За дисципліною: «Програмування мобільних пристроїв»

На тему: «Робота з декількома Activity. Явні та неявні наміри»

Виконав:

студент групи AI-193

Дмитрієв Ю.Ю.

Перевірили:

Годовиченко М.А.

Смик С.Ю.

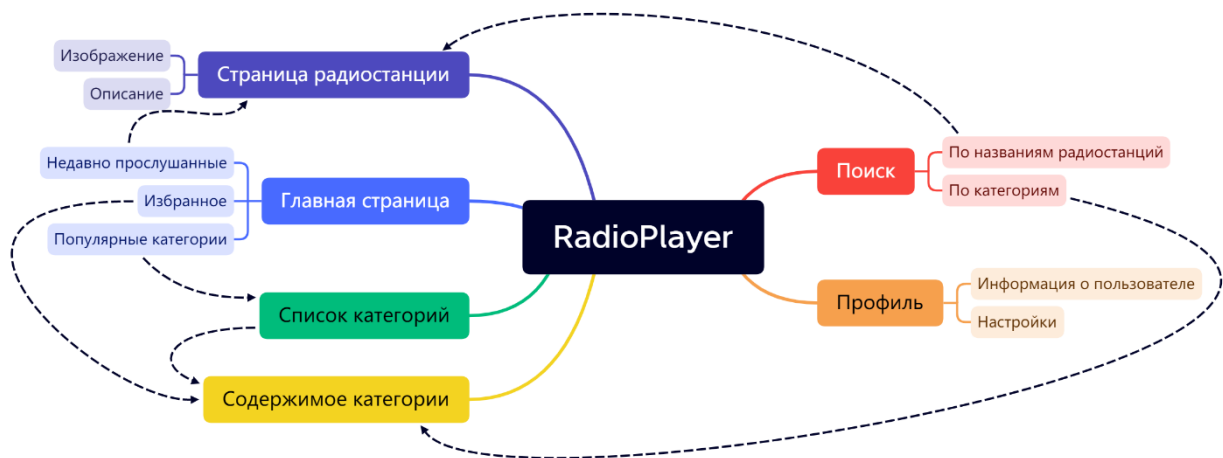
Одеса 2022

Завдання:

1. Сформувати тему для курсової та навести структуру вікон у вигляді схеми.
2. Створити застосунок для відправки селфі на електронну пошту

Хід роботи:

1. У якості курсової роботи планується розробити застосунок програвача інтернет радіостанцій. Планується з віддаленого сервера отримувати каталог та дані радіостанцій. Схему у вигляді Mindmap наведено на рисунку нижче:



2. Створимо новий проект у Android Studio із точкою входу Empty Activity.

Запишемо у клас MainActivity наступний код:

```
package com.yurdm.selfieapp

import android.Manifest
import android.content.Intent
import android.net.Uri
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.ImageView
import android.widget.Toast
import androidx.activity.result.contract.ActivityResultContracts
import androidx.core.content.FileProvider
import java.io.File

class MainActivity : AppCompatActivity() {
    private val emailReceiver = arrayOf("hodovychenko.labs@gmail.com")
    private val emailSubject = "КПП АМ-193 Дмитриев"
    private var emailImage: Uri? = null

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

        val selfieBtn: Button = findViewById(R.id.selfieBtn)
```

```

        val sendBtn: Button = findViewById(R.id.sendBtn)
        val preview: ImageView = findViewById(R.id.preview)
        var image: Uri? = null

        val getCameraImage =
            registerForActivityResult(ActivityResultContracts.TakePicture()) {
success ->
                if (success) {
                    preview.setImageURI(image)
                    emailImage = image
                }
            }

        val permissionLauncher =
            registerForActivityResult(ActivityResultContracts.RequestPermission()) { isGranted ->
                if (isGranted) {
                    image = getTempUri()
                    getCameraImage.launch(image)
                } else {
                    Toast.makeText(this, "Permission Denied",
Toast.LENGTH_SHORT).show()
                }
            }

        selfieBtn.setOnClickListener {
            permissionLauncher.launch(Manifest.permission.CAMERA)
        }

        sendBtn.setOnClickListener {
            if (emailImage == null) {
                Toast.makeText(this, "Take a selfie before sending",
Toast.LENGTH_SHORT).show()
            } else {
                val sendIntent = Intent(Intent.ACTION_SEND)
                sendIntent.type = "application/octet-stream"
                sendIntent.putExtra(Intent.EXTRA_EMAIL, emailReceiver)
                sendIntent.putExtra(Intent.EXTRA_SUBJECT, emailSubject)
                sendIntent.putExtra(Intent.EXTRA_STREAM, emailImage)

                startActivity(sendIntent)
            }
        }
    }

    private fun getTempUri(): Uri {
        val tempFile =
            File.createTempFile("tmp", ".jpg", getExternalFilesDir("tmp_images")).apply {
                createNewFile()
                deleteOnExit()
            }

        return FileProvider.getUriForFile(
            applicationContext,

```

```

        "${BuildConfig.APPLICATION_ID}.provider",
        tempFile
    )
}

```

Перевіримо роботу програми.

Після натискання “Take a selfie” відкривається камера і робить фотографію, яка завантажується у застосунок для попереднього перегляду:

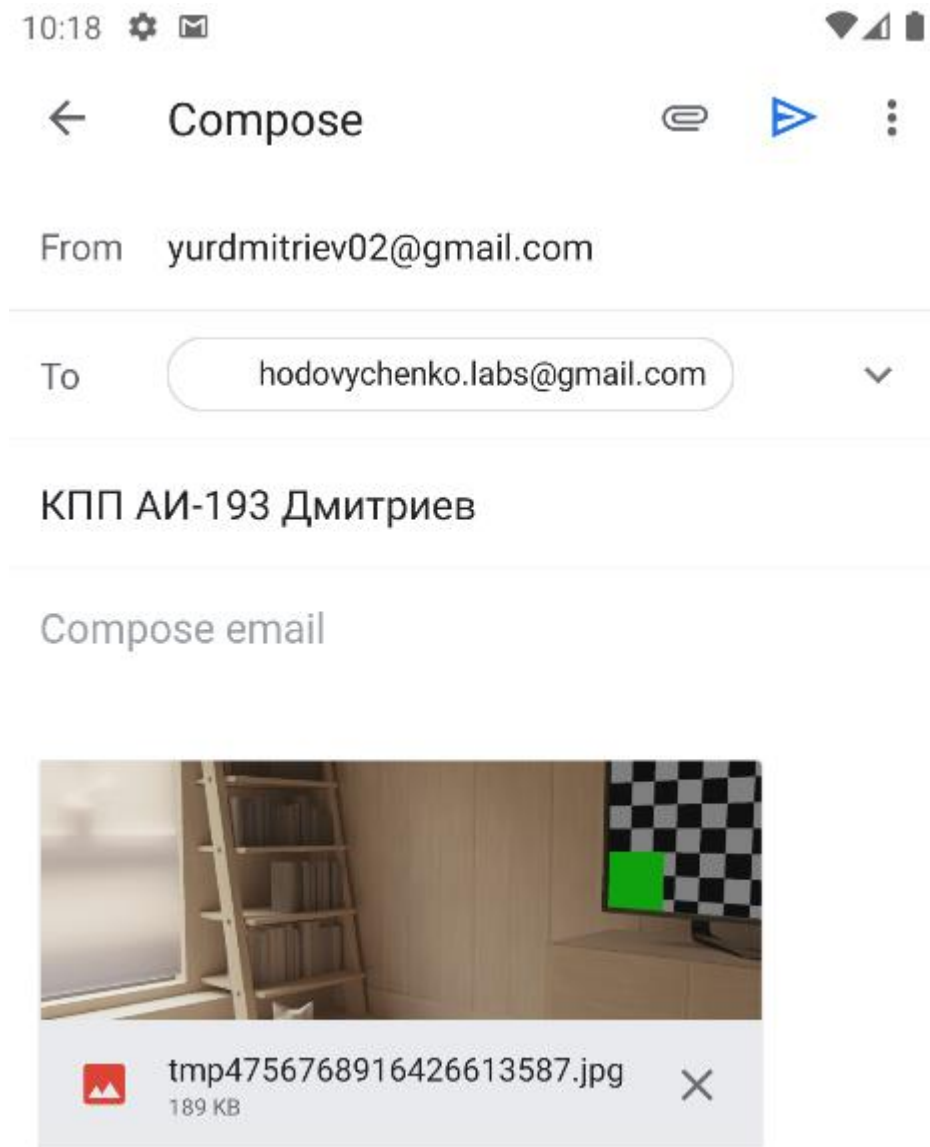


TAKE A SELFIE

SEND A SELFIE



Після натискання «Send a selfie» формується лист і робиться запит на відкриття застосунка, який може надсилати електронні листи:



Висновок: у даній лабораторній роботі набуто навичок роботи із декількома Activity за допомогою неявних намірів. Також у роботі використано Activity Result API.