

**Міністерство освіти і науки України**

**Національний університет «Львівська політехніка»**

**Кафедра ЕОМ**



**Звіт**

**до лабораторної роботи №1**

**з дисципліни: «Програмні технології мобільних обчислень»**

**на тему: «Дослідження організації функціонування МОС та технології програмної віртуалізації на прикладі ОС Android»**

**Варіант №14**

**Виконав студент групи КІ-44:**

**Ольховик О. С.**

**Перевірив:**

**Бачинський Р.В.**

**Львів 2023**

**Мета:** Засвоїти основні принципи організації функціонування мережних операційних систем та технології програмної віртуалізації.

**Завдання:** В повідомленні, яке виводить тестова програма, вказати своє прізвище та групу.

Хід виконання

## 1. Порядок встановлення та конфігурації середовища розробки програм для ОС Android.

а) Для початку потрібно завантажити і встановити програмне забезпечення JDK, доступне абсолютно безкоштовно на офіційному сайті Java.



Рис.1. Встановлення JDK

б) Розібравшись з інсталяцією і налаштуванням JDK, можна приступати до роботи безпосередньо з Андроїд Студіо(<https://developer.android.com/studio>)

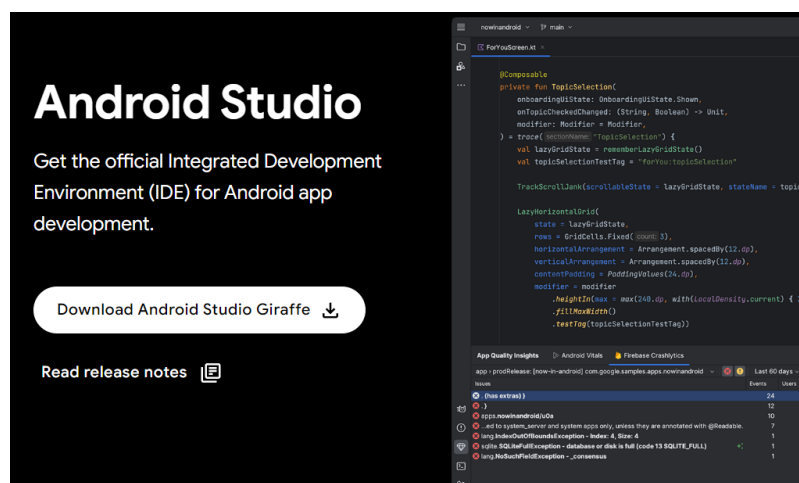


Рис.2. Завантаження Android Studio

с) У вікні «*Choose Components*» необхідно встановити галочку поруч з усіма варіантами і натиснути «*Next*».

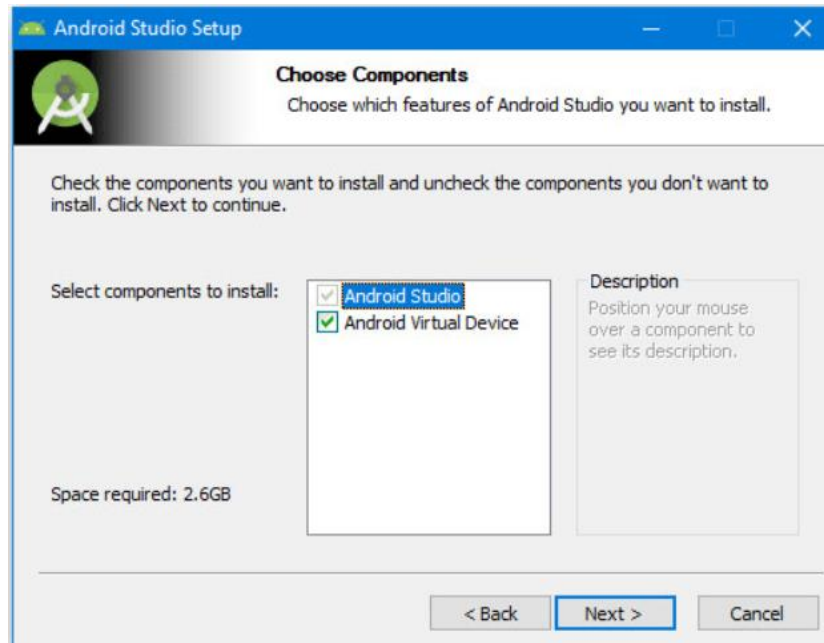


Рис.3. Встановлення Android Studio

d) Для функціонування емулятора потрібно увімкнути **Hyper-V**

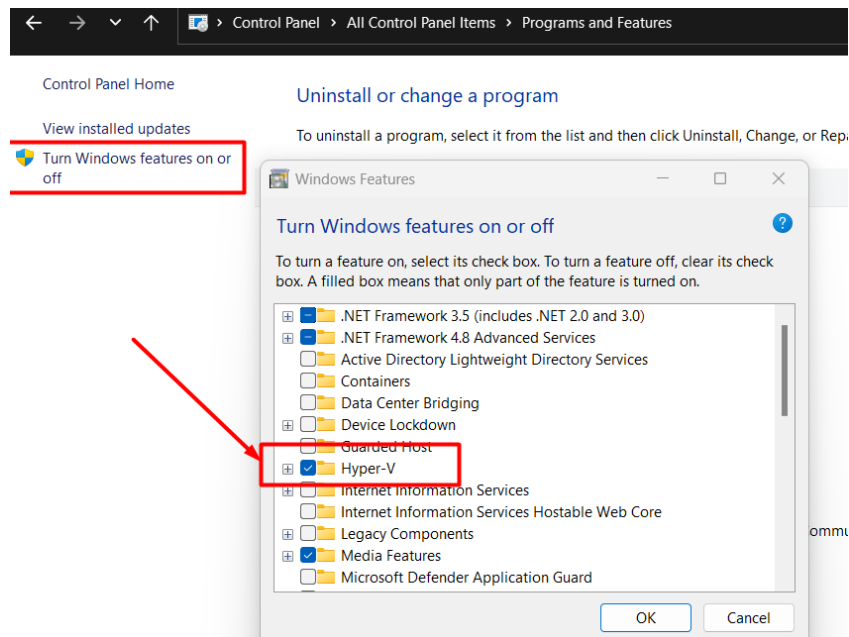


Рис.4. Ввімкнення Hyper-V

е) Для початку необхідно створити проект у Android Studio. На вітальному екрані Create New Project, після цього буде екран із шаблонами пристроїв та Activity (це клас, у якому реалізують логіку екрана програми). Для початку можна вибрати Phone and Tablet (телефони та планшети) та No Activity або Empty Activity. Різниця в тому, що в першому випадку потрібно буде реалізувати першу Activity з нуля, а в другому студія самостійно створить готовий шаблон, що редагується.

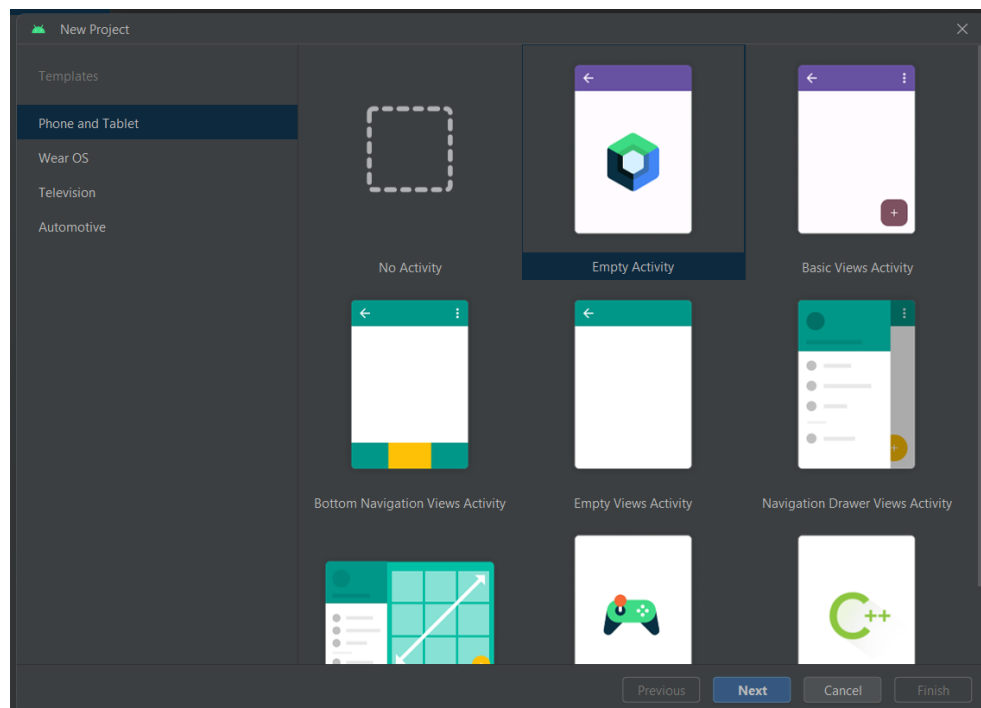


Рис.5. Створення проєкту

ф) А ось ім'я пакета (Package Name) має бути унікальним, якщо потрібно опублікувати програму в Google Play. Що ще можна вибрати: розташування проєкту (Save Location), мову (Language), яку використовувати (Java або Kotlin), та мінімальну підтримувану SDK (Minimum SDK). Мінімальна підтримувана SDK - це рання версія операційної системи Android, в якій програма може працювати.

Студія підказує, скільки пристроїв на сьогоднішній день підтримують ту чи іншу SDK.

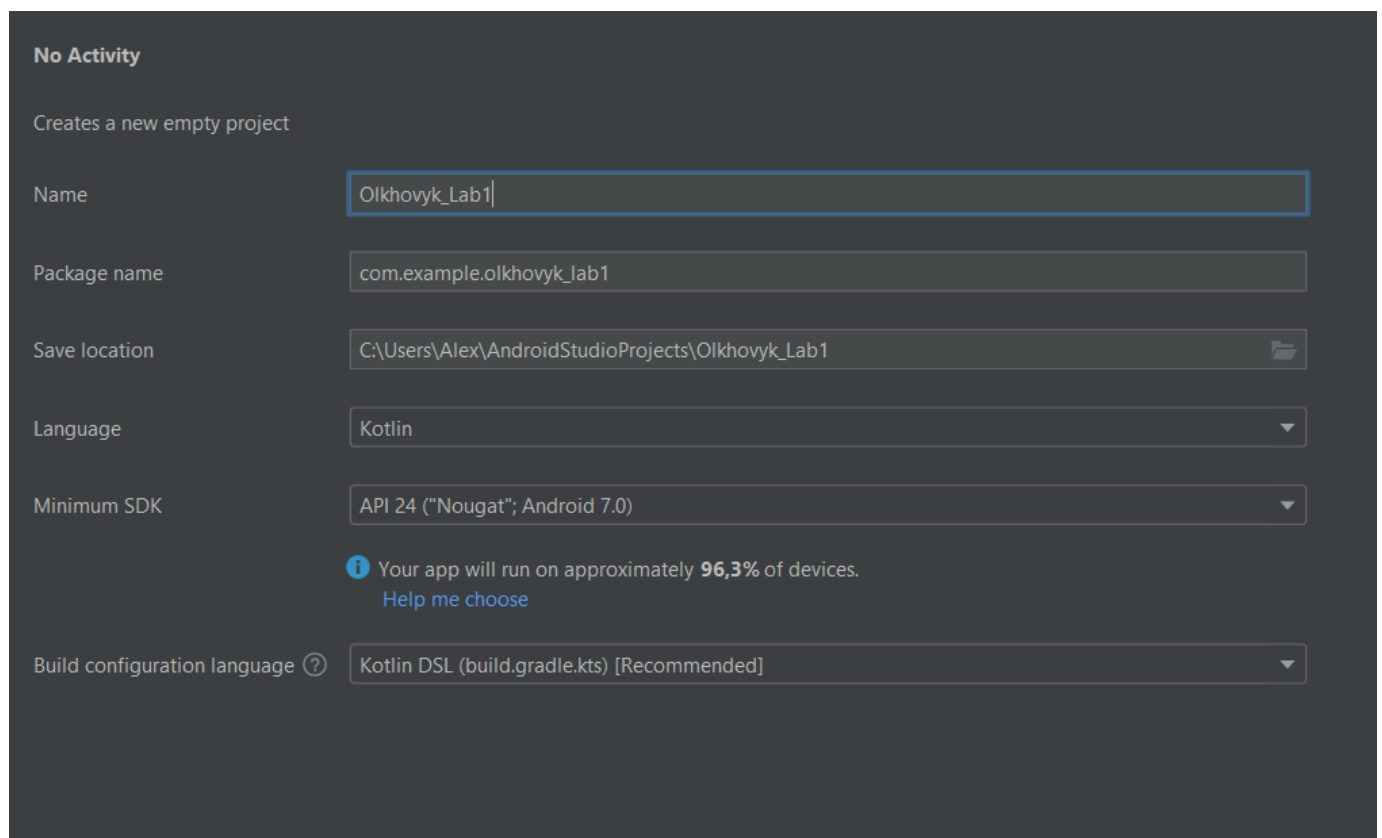


Рис.6. Вибір мови та sdk

## Розберемо структуру проекту Android:

Проект містить 2 кореневі папки: `app` та `Gradle Scripts`. Папка `app` включає 3 підпапки: `manifests` (містить файли конфігурацій програми), `java` (містить вихідний код програми) та `res` (містить файли ресурсів - картинок, стилів, розмірностей тощо).

`AndroidManifest.xml` у папці `manifests` містить інформацію про пакети та компоненти програми. Цей файл важливий для роботи Android, оскільки саме в ньому прописані дозволи програми на використання компонентів системи Android. Також у ньому визначається, як запускатимуться компоненти програми.

У папці `java` вихідні файли проекту можуть бути у різних підпапках. Зазвичай в додатку Android в папці `java` також є певна структура файлів. У цій статті ми не розглядатимемо її, оскільки тема виходить за межі загальної структури проекту.

У папці res є кілька підпапок навіть при створенні тестового базового проекту, і вона сильно розростається при розробці цих додатків. Будьте готові, що файли в цій папці можуть займати багато місця на жорсткому диску.

Розглянемо підпапки, які з'являються під час створення базового проекту:

- Папка drawable містить файли із зображеннями.
- У папці layout потрібно розташовувати XML-файли, які використовуються для створення інтерфейсу програми.
- Папка mipmap зберігає піктограми програми.
- Папка values зберігає xml-файли зі значеннями рядків, кольорів, тем, стилів.
- Папка Gradle Scripts містить скрипти для автоматизації складання проекту.
- Папка .idea не видно на скріншоті, але якщо змінити конфігурацію Android на Project, вона з'явиться. Ця папка потрібна для налаштування метаданих проекту в Android Studio.

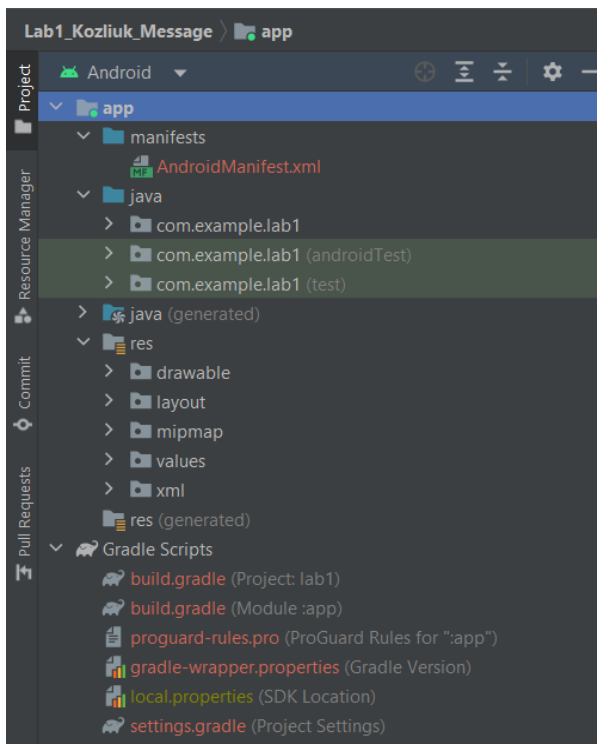


Рис.7. Структура проекту

## 2. Лістинг тестової програми

### MainActivity.java

```
package com.example.lab1;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        TextView textView = findViewById(R.id.text_view);
        String textToDisplay = "Kozliuk Dmytro KI44!";
        textView.setText(textToDisplay);
    }
}
```

### main\_activity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:layout_gravity="center_horizontal"
    android:gravity="center"
    android:orientation="vertical"
    android:padding="16dp"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:textStyle="bold" />
</LinearLayout>
```

### 3. Результати виконання тестової програми



Рис.8. Результат виконання програми

**Висновок:** Під час виконання даної лабораторної роботи, я дослідив організацію функціонування мобільної операційної системи Android та технологію програмної віртуалізації, яка лежить в її основі. Засвоїти основні принципи функціонування



мережних операційних систем та розглянув процес створення та відлагодження тестових програм для ОС Android.