# КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ТАРАСА ШЕВЧЕНКА



## ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

## **Кафедра прикладних інформаційних систем**

**Звіт до лабораторної роботи №3**

# з курсу **«Архітектура мобільних операційних систем та пристроїв»**

*Студента 4 курсу*

*Групи ПП-41*

*спеціальності 122*

*«Комп'ютерні науки»*

*ОП«Прикладне програмування»*

*Селецького Віктора Романовича*

## Київ 2023

# Тема

Робота із активностями, кольором та локалізацією в ОС Android.

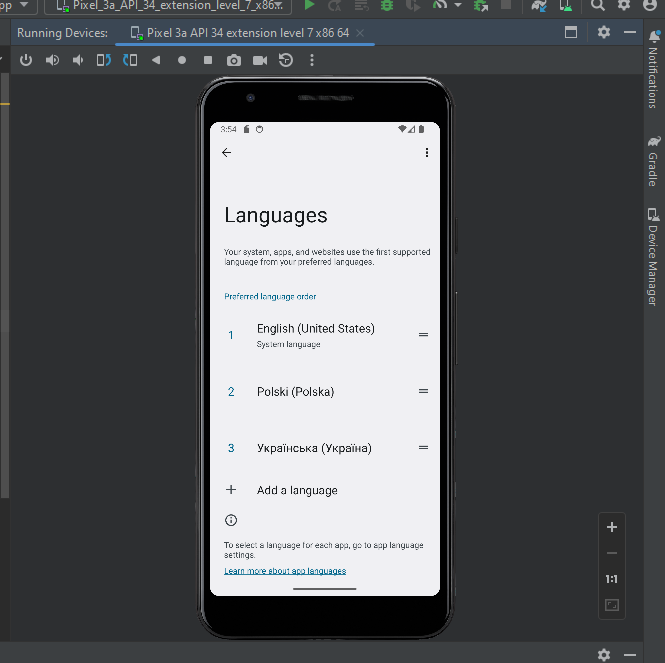
# **Мета**

Отримати досвід роботи із процесами відстеження станів активності, використанням значень рядків і кольорів та зміною локалізації у додатку.

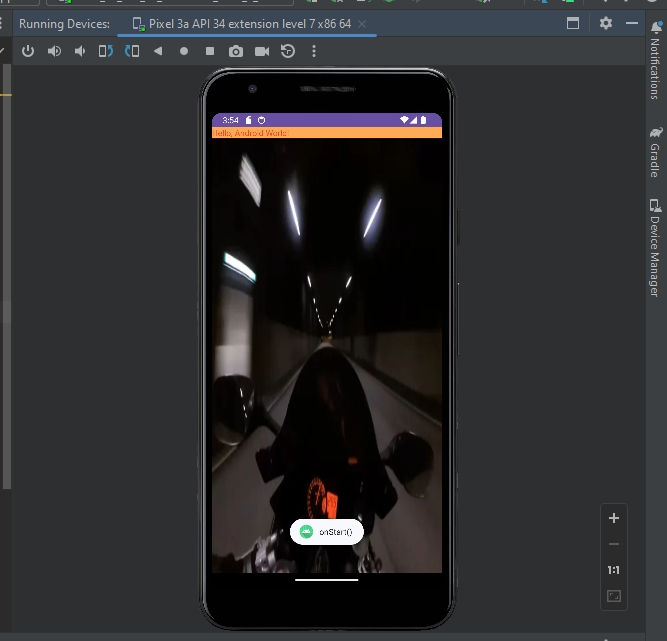
# Завдання

1. Використовуючи програму створену в 2й практичній роботі додайте до неї необхідні елементи для виконання в ній зміни у відповідності до завдань які розглянуті у пунктах теоретичних відомостей поданих вище.

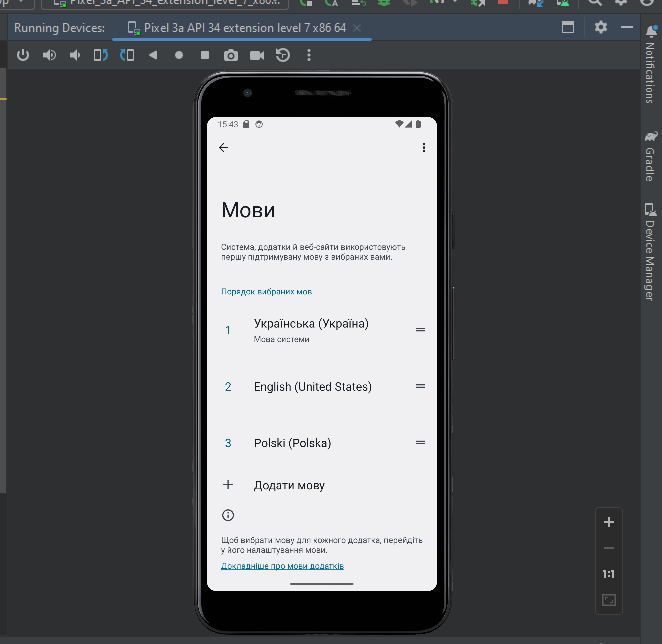
**Виконання**



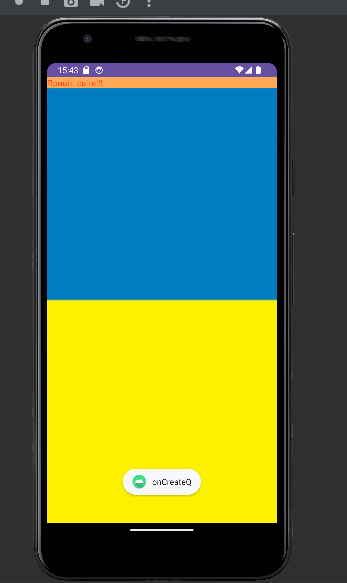
Мова за замовчуванням (англійська)



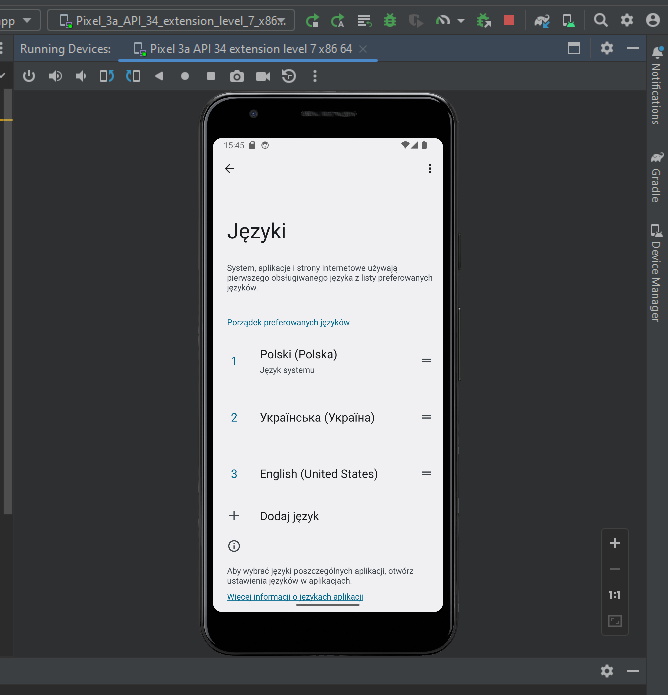
Повідомлення написано англійською, а прапору немає



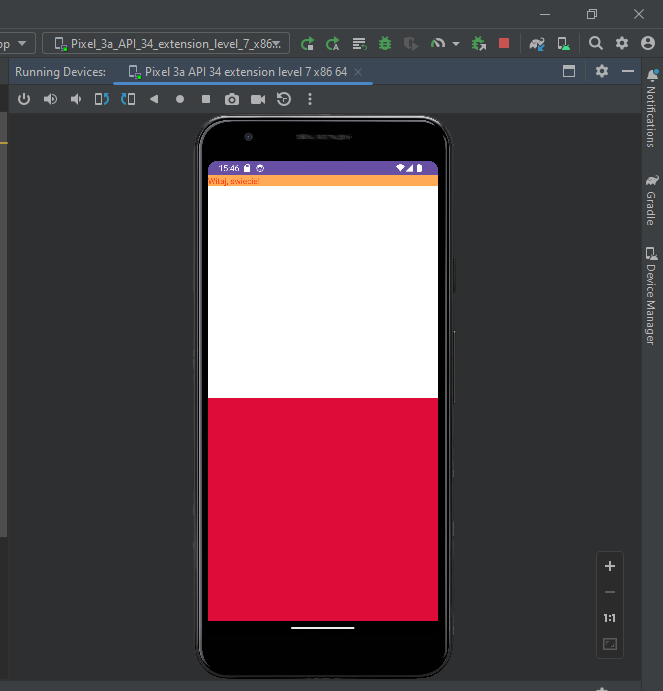
Пріоритетна мова - Українська



Текст і прапор змінено



Пріоритетна мова - Польска



Прапор і мову змінено

**activity\_main.xml:**

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

<LinearLayout

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

android:layout\_width="fill\_parent"

android:layout\_height= "fill\_parent"

android:orientation="vertical"

android:background= "@drawable/background\_image">

<TextView

android:layout\_width="fill\_parent"

android:layout\_height= "wrap\_content"

android:text= "@string/hello"

android:background= "@color/view\_bkg\_color"

android:textColor="@color/text\_color"/>

</LinearLayout>

**colors.xml:**

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

<resources>

<color name="black">#FF215574</color>

<color name="white">#FFF67FFF</color>

<color name="view\_bkg\_color">#FA5</color>

<color name="screen\_bkg\_color">#F28</color>

<color name="text\_color">#99FB0000</color>

</resources>

**strings.xml**

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

<resources>

<string name="hello">Hello, Android World!</string>

<string name="app\_name">HelloAndroidWorld</string>

<string name="image\_path">drawable/flag\_of\_poland\_svg</string>

</resources>

**strings-uk.xml:**

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

<resources>

<string name="hello">Привіт, світе!!!</string>

<string name="app\_name">HelloAndroidWorld</string>

<string name="image\_path">drawable/flag\_of\_poland\_svg</string>

</resources>

**strings-pl.xml:**

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

<resources>

<string name="hello">Witaj, swiecie!</string>

<string name="app\_name">HelloAndroidWorld</string>

<string name="image\_path">drawable/flag\_of\_poland\_svg</string>

</resources>

**Java:**

package com.example.lab3;

import androidx.appcompat.app.AppCompatActivity;

import android.app.Activity;

import android.os.Bundle;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

Toast.makeText(this, "onCreateQ", Toast.LENGTH\_LONG).show();

}

@Override

protected void onPause()

{

Toast.makeText(this, "onPause()", Toast.LENGTH\_LONG).show();

super.onPause();

}

@Override

protected void onRestart() {

super.onRestart();

Toast.makeText(this, "onRestart() ", Toast.LENGTH\_LONG).show();

}

@Override

protected void onStart() {

super.onStart();

Toast.makeText(this, "onStart() ", Toast.LENGTH\_LONG).show();

}

}

**Контрольні питання**

1. Стани активностей та їх відстеження.

Кожна активність - це екран, який додаток може показувати користувачам. Чим складніший додаток, тим більше екранів (активностей) йому потрібно для реалізації функцій. При створенні додатку потрібно, як мінімум, створити початковий екран, який являє собою основу інтерфейсу. За необхідності цей інтерфейс доповнюється другорядними активностями, що призначені для вводу/виводу інформації та додаткових можливостей. Запуск (або повернення з) нової активності до попередньої призводить до «переміщення» між екранами користувацького інтерфейсу.

Більшість активностей проектуються таким чином, щоб використовувати весь екранний простір, але можна також створювати напівпрозорі або плаваючі діалогові вікна для зручності користувача.

Активності можуть знаходитися в одному з чотирьох станів:

- Активна (Active). Активність знаходиться на передньому плані (на вершині стека) і має можливість взаємодіяти з користувачем. Android намагатиметься зберегти її працездатність, при необхідності перериваючи роботу інших активних, що знаходяться на нижчих позиціях в стеку для надання необхідних ресурсів. При виході на передній план іншої активності робота даної активності буде припинена або зупинена.

- Призупинена (Paused). Активність може бути видима на екрані, але не може взаємодіяти з користувачем: в цей момент вона призупинена. Це трапляється, коли на передньому плані знаходяться напівпрозорі або плаваючі (наприклад, діалогові) вікна. Робота призупиненої активності може

бути припинена, якщо ОС необхідно виділити ресурси активності переднього плану. Якщо активність повністю зникає з екрану, вона зупиняється.

- Зупинена (Stopped). Активність невидима, вона знаходиться в пам'яті, зберігаючи інформацію про свій стан. Така активність стає кандидатом на передчасне закриття, якщо системі буде потрібно пам'ять для чогось іншого. При зупинці активності розробнику важливо зберегти дані і поточний стан користувача інтерфейсу (стан полів введення, позицію курсора і т.д.). Якщо активність завершує свою роботу або закривається, вона стає неактивною.

- Неактивна (Inactive). Коли робота активності завершена, і перед тим, як вона буде запущена, дана активність знаходиться в неактивному стані. Такі активності видаляються з стека і повинні бути перезапущені, щоб їх можна було використовувати.

Зміна стану додатка - недетермінований процес, що керується виключно менеджером пам'яті Android. При необхідності Android спочатку закриває додатки, що містять неактивні активності, потім зупинені і, в крайньому випадку, припинені.

Для забезпечення повноцінного інтерфейсу додатку, зміни його стану повинні бути непомітні для користувача. Міняючи свій стан з призупиненого на зупинений або з неактивного на активний, активність не повинна зовні змінюватися. При зупинці або призупиненні роботи активності розробник повинен забезпечити збереження стану активності, щоб її можна було відновити при виході активності на передній план. Для цього в класі Activity міститься обробники подій, перевизначення яких дозволяє розробнику відслідковувати зміну станів активності.

Обробники подій класу Activity дозволяють відслідковувати зміни станів відповідного об'єкта Activity під час усього життєвого циклу, що відображає рисунок поданий нижче.

2. Особливості використання ресурсів у коді програми.

Ресурси меню можуть використовуватися для опису як головного меню активності, так і контекстного, що з'являється при тривалому натисканні на який-небудь елемент користувацького інтерфейсу. Меню, описане у форматі XML, завантажується в додаток з допомогою методу **inflate** системного сервісу **MenuInflater**. Зазвичай це відбувається всередині методу **onCreateOptionsMenu** (для головного, так званого бургерного меню) або **onCreateContextMenu** (для контекстного меню), перевизначених в активності. Кожен екземпляр меню описується в окремому файлі XML в каталозі res/menu. Зазвичай імена файлів (без розширень) стають іменами ресурсів.

Однією з основних переваг використання зовнішніх по відношенню до коду ресурсів - можливість використання механізму автоматичного вибору ресурсів. Користуючись описаним нижче механізмом, можна створювати індивідуальні ресурси для різних апаратних конфігурацій, мов, регіонів і т.д.

Під час виконання додатка система автоматично обирає ресурси.

3. Робота із анімацією в Android.

Android підтримує два види анімації: покрокову анімацію (послідовний вивід на екран зображення з заданою тривалістю) і анімацію, що базується на розрахунку проміжних кадрів (в цьому випадку застосовуються різні перетворення - обертання, розтягування, переміщення і затемнення).

Всі ці трансформації описуються в XML-файлі в каталозі res/anim.

Ресурс, що описує покрокову анімацію, зберігається в каталозі res/drawable.

4. Робота із кольорами та колірні моделі в Android.

Для опису кольорів використовується тег <color>. Значення кольору вказуються в шістнадцятковому вигляді в одному з наступних форматів:

# RGB

#ARGB

# RRGGBB

# AARRGGBB

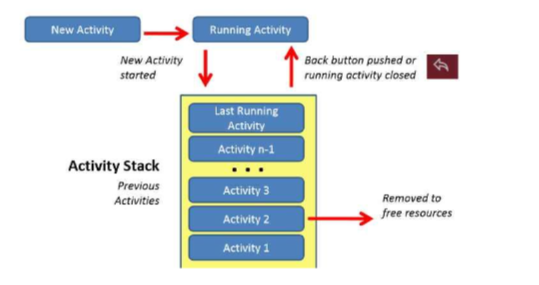
У прикладі показано опис напівпрозорого червоного кольору і непрозорого зеленого:

<color name="transparentred">#77FF0000</color>

<color name="opaquegreen">#0F0</color>

Посилання на розміри найчастіше зустрічаються всередині ресурсів зі стилями і розміткою, наприклад, при вказівці товщини рамки або величини шрифту.

5. Життєвий цикл активності.



**Висновок**

В результаті виконання даної лабораторної роботи я отримав досвід роботи із процесами відстеження станів активності, використанням значень рядків і кольорів та зміною локалізації у додатку. Вважаю дану лабораторну роботу виконаною в повному обсязі.