

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
Факультет інформатики та обчислювальної техніки
Кафедра інформатики та програмної інженерії

Лабораторна №5
з курсу «Розробка мобільних застосунків під Android»

Прийняв:
Орленко С. П.

Виконала:
студентка групи ПІ-21
Гордич О. Ю.

Київ 2025

ЛАБОРАТОРНА РОБОТА № 5

ДОСЛІДЖЕННЯ РОБОТИ З ВБУДОВАНИМИ ДАТЧИКАМИ

Мета роботи: ознайомитись з можливостями вбудованих датчиків мобільних пристроїв та дослідити способи їх використання для збору та обробки даних.

ЗАВДАННЯ

БАЗОВЕ (10/20 балів). Написати програму під платформу Андроїд, яка має інтерфейс для виведення даних з обраного вбудованого датчика (тип обирається самостійно, можна відслідковувати зміни значень і з декількох датчиків).

Опис функціоналу програми:

У створеному додатку реалізовано взаємодію з двома типами сенсорів Android-пристрою:

1. **Датчик освітленості (Light Sensor)** – дозволяє вимірювати рівень освітлення в люксах.
2. **Сенсор орієнтації (Orientation Sensor)** – надає значення азимута, нахилу і обертання пристрою у просторі.

Архітектура програми:

- **MainActivity.kt:**
 - Головна активність програми.
 - Використовує ViewPager2 та TabLayout для перемикання між вкладками.
 - Містить два фрагменти:
 - LightFragment (вкладка "Light")
 - OrientationFragment (вкладка "Orientation")
- **LightFragment.kt:**
 - Реалізує обробку даних з датчика освітленості (**Sensor.TYPE_LIGHT**).
 - Виводить назву сенсора та поточний рівень освітлення.

- **OrientationFragment.kt:**

- Взаємодіє з **сенсором орієнтації (Sensor.TYPE_ORIENTATION)**.
- Виводить назву сенсора та поточні значення: азимут, кут нахилу (pitch), обертання (roll).

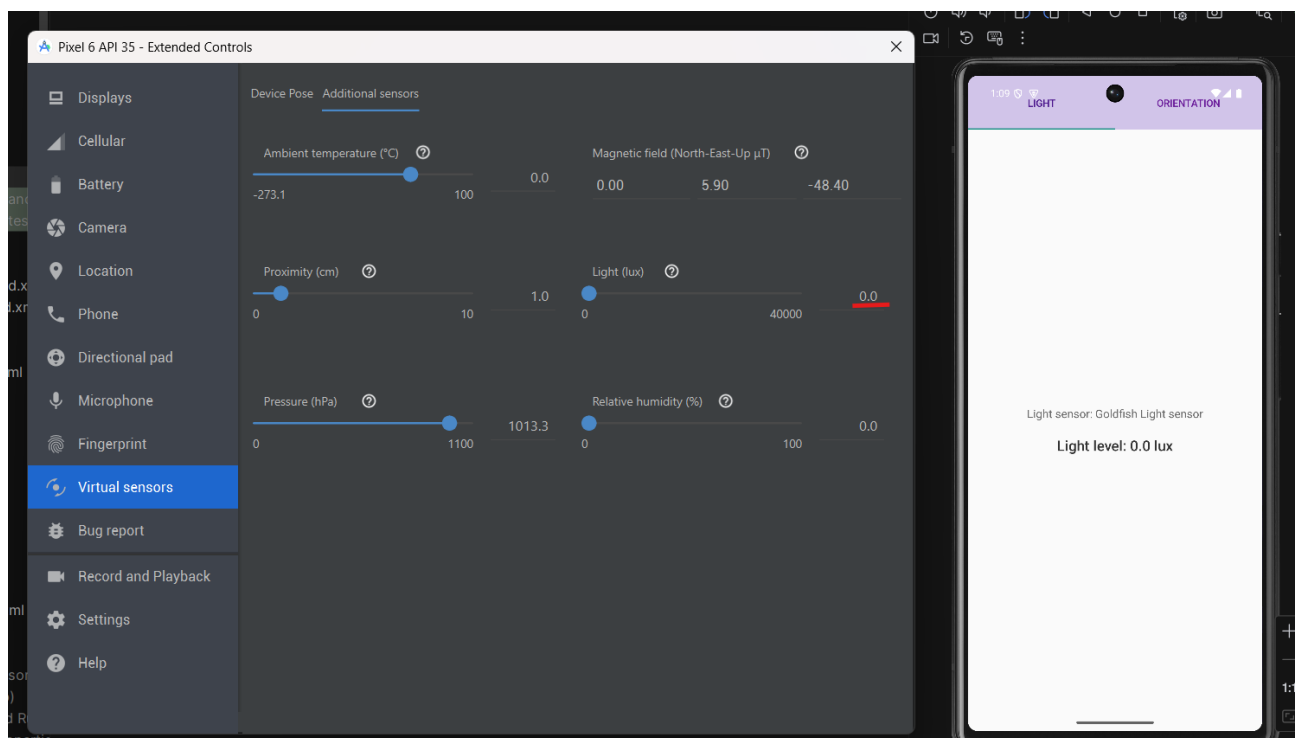
Використані бібліотеки та компоненти Android SDK:

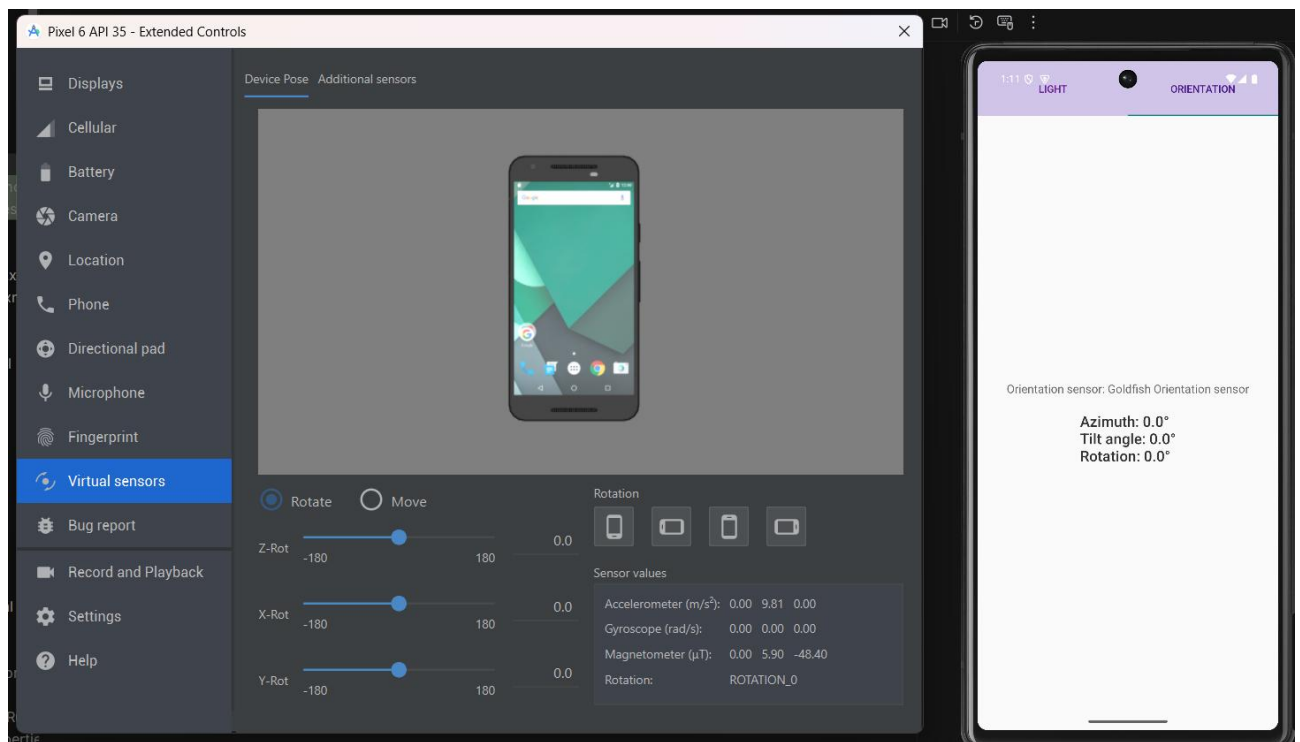
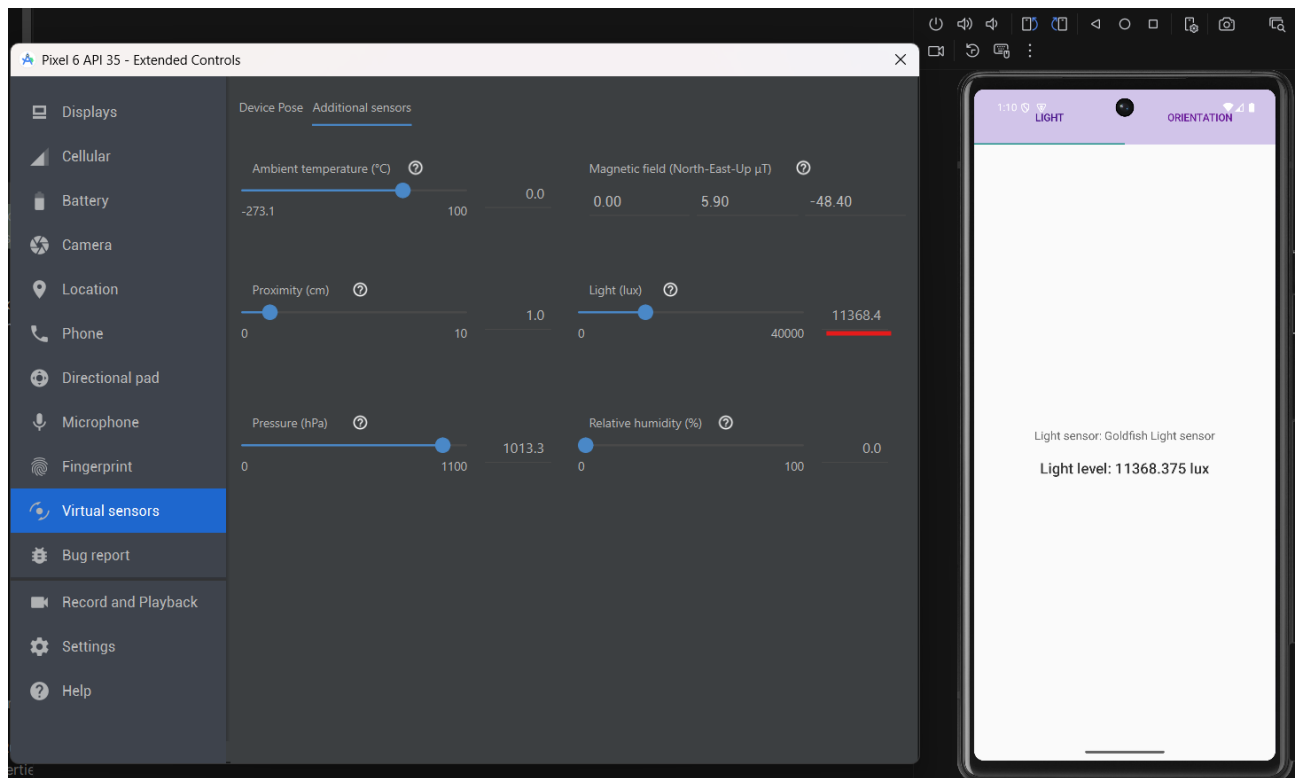
Компонент	Призначення
android.hardware.SensorManager	Клас для доступу до датчиків пристрою
android.hardware.Sensor	Представляє фізичний сенсор
android.hardware.SensorEventListener	Інтерфейс для отримання подій з сенсорів

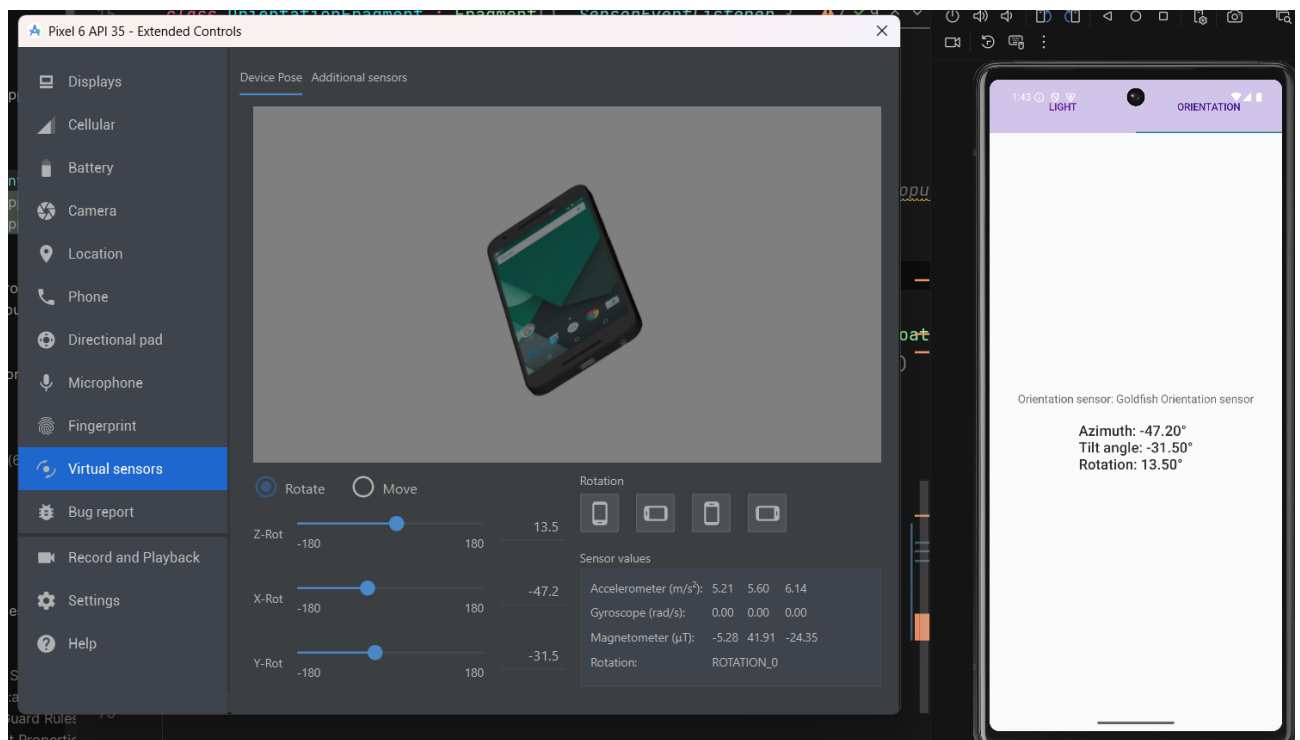
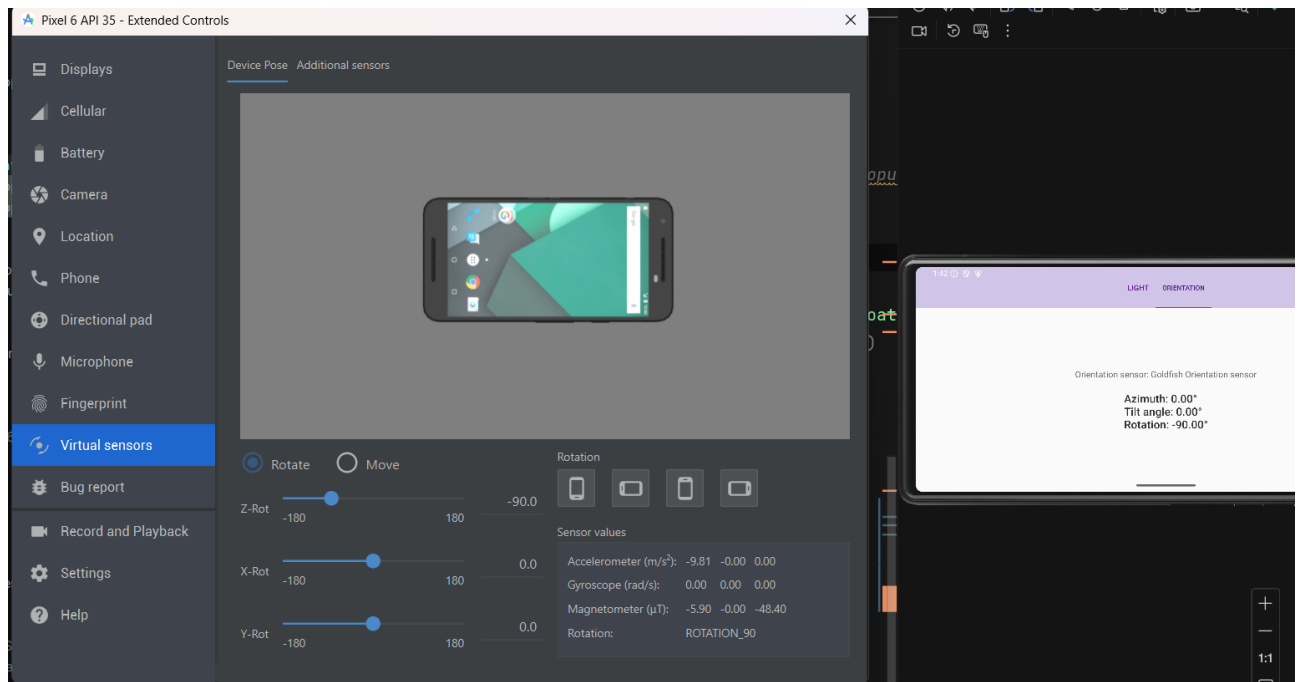
Життєвий цикл роботи з сенсорами:

1. **onViewCreated():** Отримання сенсора через SensorManager та ініціалізація елементів інтерфейсу.
2. **onResume():** Реєстрація слухача подій сенсора.
3. **onSensorChanged():** Обробка нових значень і оновлення інтерфейсу.
4. **onPause():** Відключення слухача при призупиненні активності.

Тестування програми:







Висновок:

Програма демонструє базову взаємодію з вбудованими сенсорами Android-пристрою. За допомогою бібліотек Android SDK реалізовано отримання та виведення значень сенсорів у реальному часі. Архітектура з використанням Fragment та ViewPager2 забезпечує зручну навігацію та масштабованість інтерфейсу.

КОНТРОЛЬНІ ПИТАННЯ

1. Наведіть приклади вбудованих датчиків та величини які з них можна зчитати.

Датчик	Тип (Sensor.TYPE_*)	Величини
Акселерометр	TYPE_ACCELEROMETER	Прискорення по осях X, Y, Z (в м/с ²)
Гіроскоп	TYPE_GYROSCOPE	Кутові швидкості обертання по X, Y, Z (рад/с)
Магнітометр (компас)	TYPE_MAGNETIC_FIELD	Напруженість магнітного поля по X, Y, Z (в мкТл)
Датчик освітлення	TYPE_LIGHT	Рівень освітленості (в люксах)
Орієнтаційний сенсор (deprecated, замінюється на обчислення з акселерометра + магнітометра)	TYPE_ORIENTATION	Азимут, нахил (pitch), обертання (roll)
Датчик тиску	TYPE_PRESSURE	Атмосферний тиск (в гПа)
Датчик температури	TYPE_AMBIENT_TEMPERATURE	Температура навколишнього середовища (в °C)
Датчик наближення	TYPE_PROXIMITY	Відстань до об'єкта (в см)
Датчик положення (Rotation Vector)	TYPE_ROTATION_VECTOR	Орієнтація пристрою у вигляді кватерніонів

2. Наведіть особливості роботи з вбудованими датчиками.

- **Доступ через SensorManager:** Для взаємодії з датчиками необхідно отримати екземпляр системного сервісу SensorManager.
- **Отримання переліку доступних датчиків:** Можна отримати список усіх датчиків певного типу або всіх наявних на пристрої.
- **Реєстрація слухачів (SensorEventListener):** Щоб отримувати дані від датчика, потрібно зареєструвати слухача, який реалізує інтерфейс SensorEventListener та його методи onSensorChanged() (викликається при зміні значення датчика) та onAccuracyChanged() (викликається при зміні точності датчика).

- **Нерегулярність подій:** Події від датчиків надходять асинхронно і нерегулярно, залежно від швидкості зміни фізичної величини та встановленої частоти оновлення.
- **Різна доступність на пристроях:** Набір вбудованих датчиків може суттєво відрізнятися залежно від моделі та виробника пристрою. Перед використанням датчика слід перевіряти його наявність.
- **Енергоспоживання:** Активне використання деяких датчиків може призводити до значного споживання заряду батареї. Важливо скасовувати реєстрацію слухачів у методі `onPause()` Activity або Fragment, коли вони більше не потрібні.
- **Точність та затримка:** Датчики мають різну точність і підтримують різні рівні затримки при отриманні даних (`SENSOR_DELAY_NORMAL`, `SENSOR_DELAY_UI`, `SENSOR_DELAY_GAME`, `SENSOR_DELAY_FASTEST`). Вибір затримки впливає на частоту оновлення даних та енергоспоживання.
- **Системні дозволи (для деяких датчиків):** Хоча більшість поширених датчиків не потребують спеціальних дозволів, деякі (наприклад, датчики серцевого ритму) можуть вимагати оголошення відповідних дозволів у `AndroidManifest.xml` та отримання їх від користувача.
- **Отримання значень:** Дані датчиків передаються у вигляді об'єкта `SensorEvent`, а самі значення зберігаються у масиві `values`. Кількість та значення елементів у цьому масиві залежать від типу датчика.
- **Застарілі датчики:** Деякі датчики, такі як `TYPE_ORIENTATION`, вважаються застарілими, і Google рекомендує використовувати новіші альтернативи для отримання більш точних та стабільних даних.