Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра Информационных систем и программной инженерии

Лабораторная работа №4

«Работа с сенсорами на операционной системе Android»

по дисциплине

«Технологии разработки мобильных приложений»

Выполнил:

Ст. гр. ИСТ-114

Борисов ГМ.

Принял:

Казаев АТ.

Владимир 2017

**Цель работы:** Разработать мобильное приложение, которое будет считывать сигналы сенсоров устройства и представлять их в удобном для восприятия виде.

**Задания**

1. Выполнить пример из лабораторной работы.
2. Выполнить одно из следующих трех заданий:
   1. Используя акселерометр написать приложение, в котором можно перемещать объект на экране с помощью наклона телефона.
   2. Используя магнитный сенсор сделать компас, который бы примерно определял, в каком направлении расположен телефон (север/юг/запад/восток)
   3. Используя датчик освещенности менять яркость экрана в зависимости от уровня света в комнате.

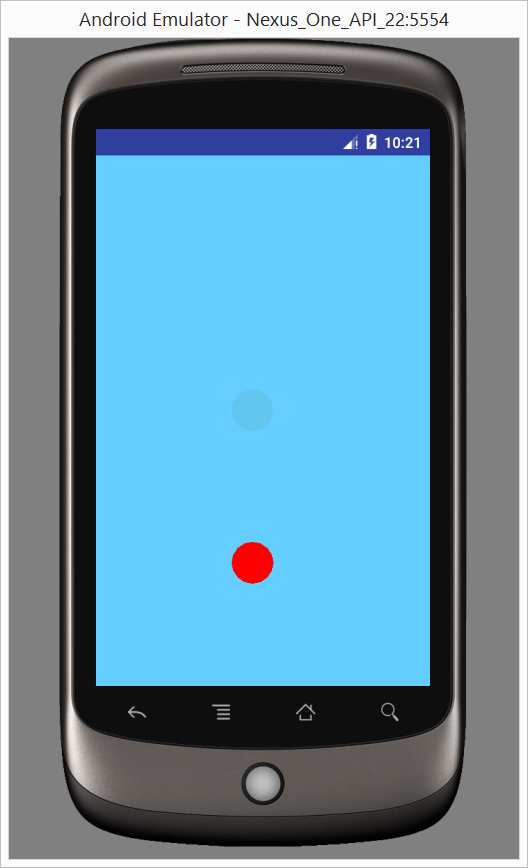
**Ход работы:**

1. Скопировал код из методички и добавил кнопку для перехода к дополнительному заданию. Результат (Рис1).



*Рисунок 1. Виртуальный девайс.*

1. Добавил activity и код который позволяет шарику скатываться в сторону наклона смартфона(Рис2). В Main класс этого активити прописан код, который считывает показатели акселерометра по ОХ и OY, и размеры контейнера, в котором прорисовывается шарик. Шарик прорисовывается в контейнере DrowView при помощи метода canvas.drawCircle. всё это происходит в отдельном потоке класса DrawThread. Листинг метода Main2Activity, для этого активити представлен ниже.



*Рисунок 2. Шарик.*

**Листинг метода Main2Activity**

**public class** Main2Activity **extends** Activity **implements** SensorEventListener {  
 Display **display**;  
 Point **size**;  
 **int MaxWidth**;  
 **int MaxHeight**;  
 **int MinWidth** = 30;  
 **int MinHeight** = 30;  
 Paint **p**;  
 Sensor **aSensor**;  
 **int aX**;  
 **int aY**;  
 SensorManager **sensorManager**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(**new** DrawView(**this**));  
 **sensorManager** = (SensorManager) getSystemService(***SENSOR\_SERVICE***);  
 **aSensor** = **sensorManager**.getDefaultSensor(Sensor.***TYPE\_ACCELEROMETER***);  
 }  
 @Override  
 **public void** onSensorChanged(SensorEvent event) {  
 **if** (event.**sensor**.getType() == Sensor.***TYPE\_ACCELEROMETER***) {  
 **aX** = Integer.*valueOf*((**int**) event.**values**[0]);  
 **aY** = Integer.*valueOf*((**int**) event.**values**[1]);  
 }  
 }  
 @Override  
 **public void** onAccuracyChanged(Sensor sensor, **int** i) {  
 }  
 @Override  
 **public void** onStart() {  
 **super**.onStart();  
 **sensorManager**.registerListener(**this**, **aSensor**, **sensorManager**.***SENSOR\_DELAY\_FASTEST***);  
 }  
 @Override  
 **public void** onStop() {  
 **super**.onStop();  
 **sensorManager**.unregisterListener(**this**, **aSensor**);  
 }  
 **class** DrawView **extends** SurfaceView **implements** SurfaceHolder.Callback {  
 **private** DrawThread **drawThread**;  
  
 **public** DrawView(Context context) {  
 **super**(context);  
 getHolder().addCallback(**this**);  
 **p** = **new** Paint();  
 }  
 @Override  
 **public void** surfaceChanged(SurfaceHolder holder, **int** format, **int** width, **int** height) {  
  
 }  
 @Override  
 **public void** surfaceCreated(SurfaceHolder holder) {  
 **drawThread** = **new** DrawThread(getHolder());  
 **drawThread**.setRunning(**true**);  
 **drawThread**.start();  
 **MaxWidth** = **this**.getMeasuredWidth() - 30;  
 **MaxHeight** = **this**.getMeasuredHeight() - 30;  
 **p** = **new** Paint();  
 }  
 @Override  
 **public void** surfaceDestroyed(SurfaceHolder holder) {  
 **boolean** retry = **true**;  
 **drawThread**.setRunning(**false**);  
 **while** (retry) {  
 **try** {  
 **drawThread**.join();  
 retry = **false**;  
 } **catch** (InterruptedException e) {  
 }  
 }  
 }  
 **class** DrawThread **extends** Thread {  
 **private boolean running** = **false**;  
 **private** SurfaceHolder **surfaceHolder**;  
 **public** DrawThread(SurfaceHolder surfaceHolder) {  
 **this**.**surfaceHolder** = surfaceHolder;  
 }  
 **public void** setRunning(**boolean** running) {  
 **this**.**running** = running;  
 }  
 @Override  
 **public void** run() {  
 Canvas canvas;  
 **while** (**running**) {  
 canvas = **null**;  
 **try** {  
 canvas = **surfaceHolder**.lockCanvas(**null**);  
 **if** (canvas == **null**)  
 **continue**;  
 *// заливка канвы цветом* canvas.drawARGB(80, 102, 204, 255);  
 *// настройка кисти  
 // красный цвет* **p**.setColor(Color.***RED***);  
 *// толщина линии = 10* **p**.setStrokeWidth(10);  
 *// рисуем круг с центром в (x,y), радиус = 30* canvas.drawCircle(**MaxWidth** / 2 + **aX** \* -1 \* **MaxWidth** / 30, **MaxHeight** / 2 + **aY** \* **MaxHeight** / 30, 30, **p**);  
 } **finally** {  
 **if** (canvas != **null**) {  
 **surfaceHolder**.unlockCanvasAndPost(canvas);  
 }  
 }  
 }  
 }  
 }  
 }  
}

1. Кот программы можно посмотреть на github link:

<https://github.com/Ubonasher/lab4-Android>

**Вывод**: в ходе работы я освоил базовые принципы работы с сенсорами в Android studio.