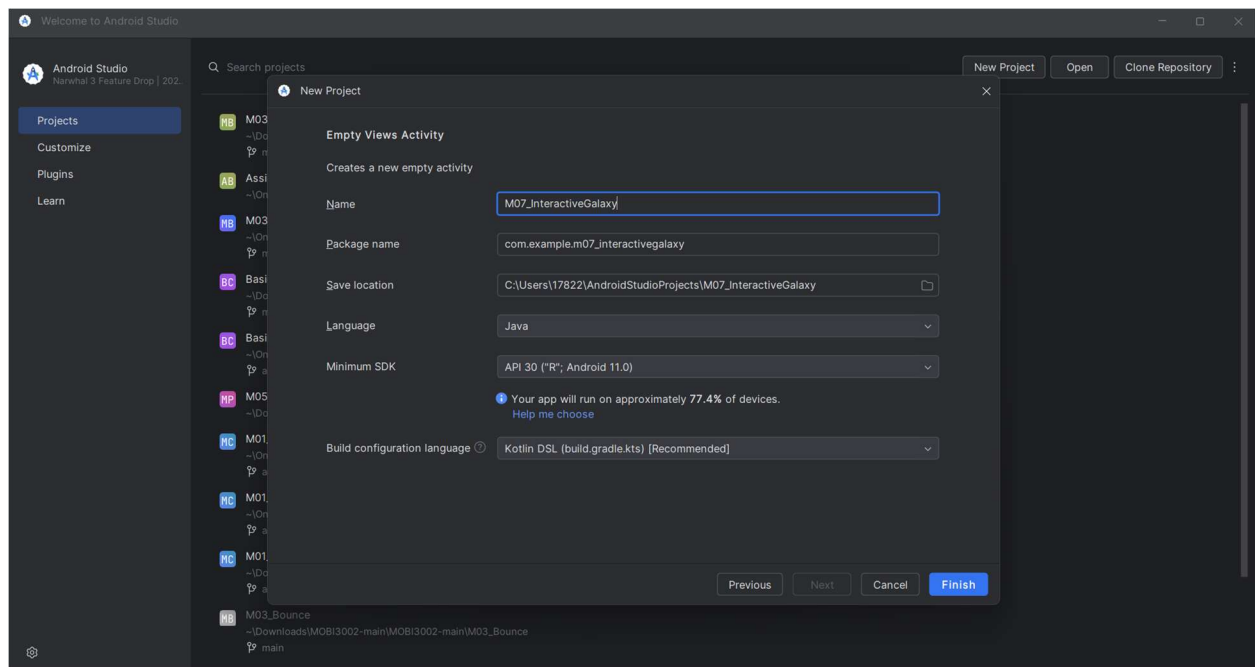


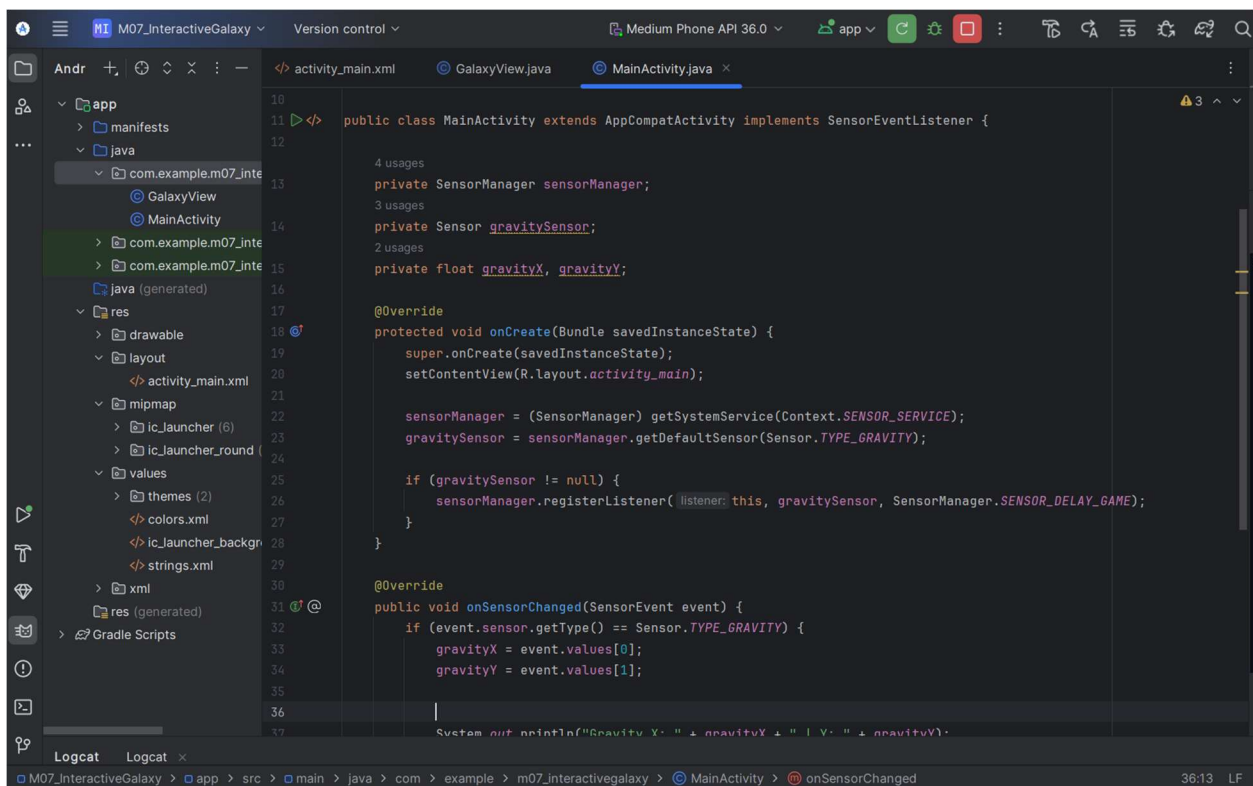
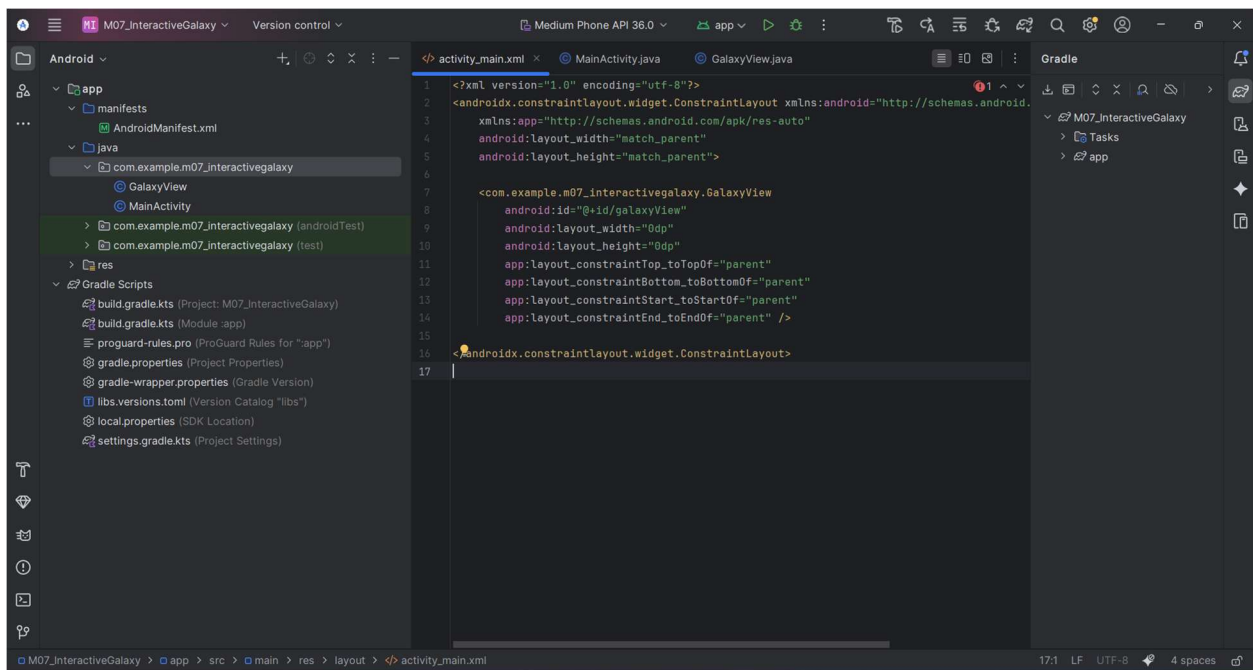
A7 - Sensors

Esraa Abdelmoein

2025-10-20

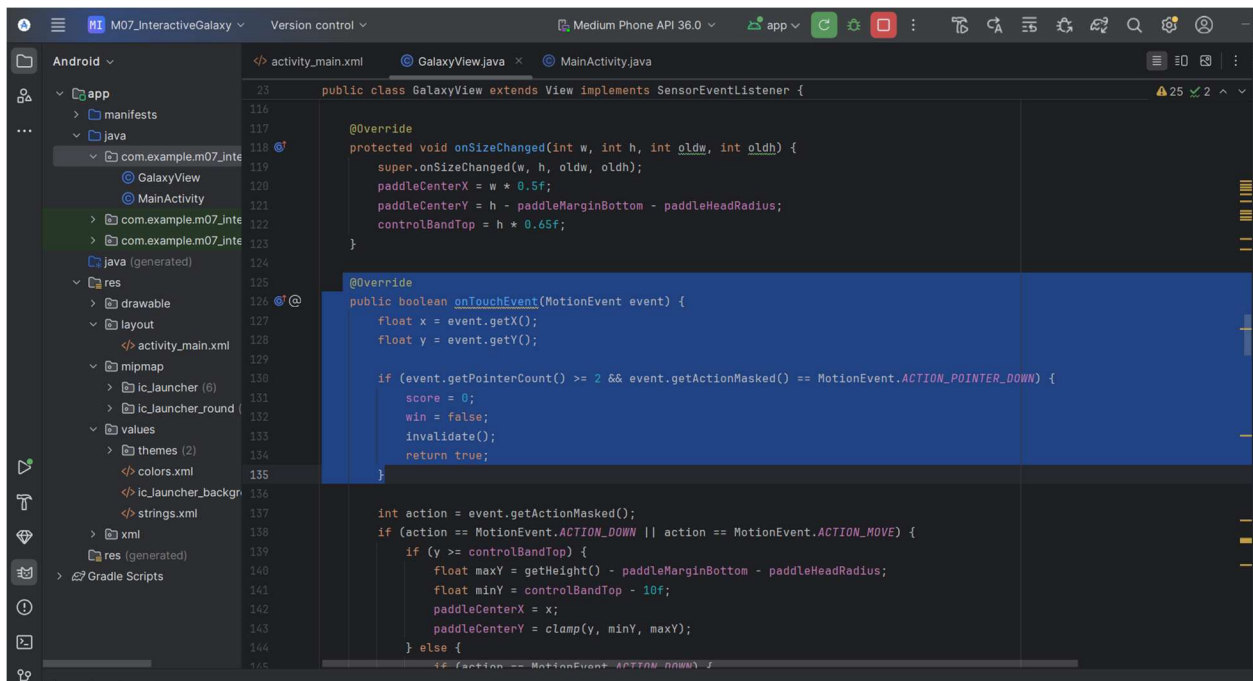
Here is my GitHub URL: https://github.com/EsraaAbdelmoein/Mobile-Application-Android/tree/main/Assignment07_Sensors/M07_InteractiveGalaxy





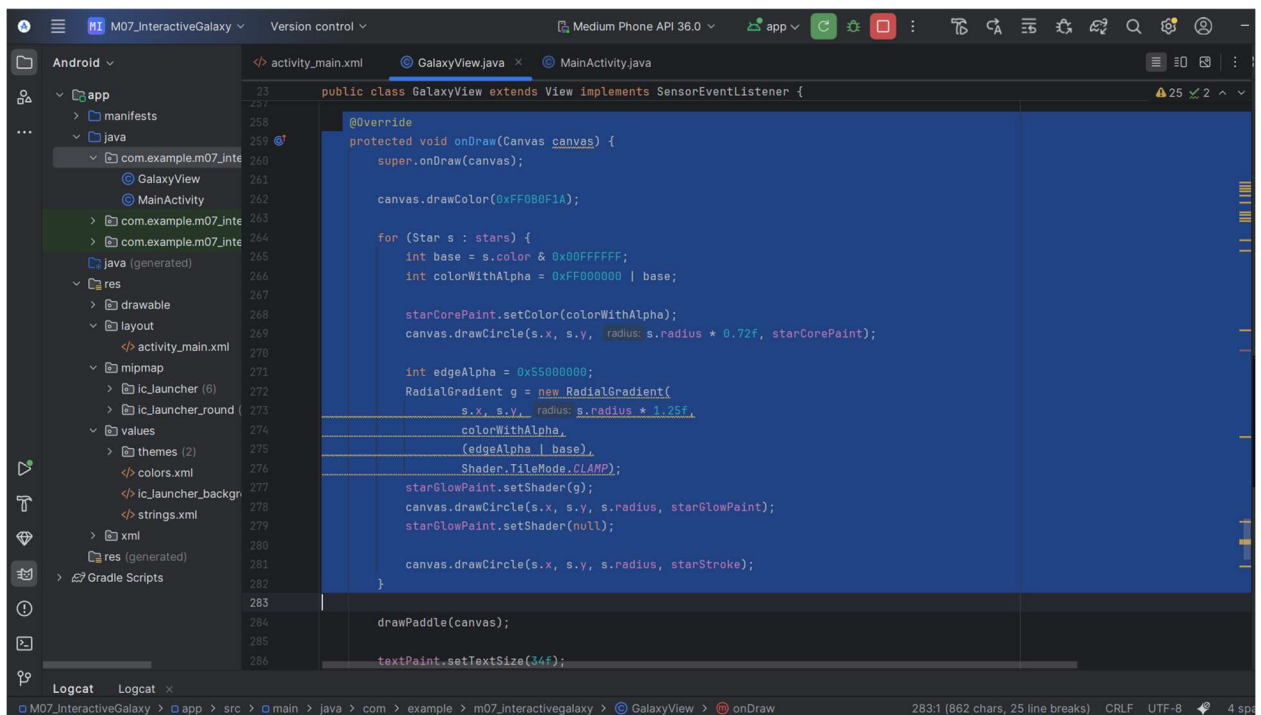
Gravity Sensor Setup (MainActivity.java)

This code registers the gravity sensor and updates the object's X and Y positions in real time. The values are used to simulate tilt movement inside the game



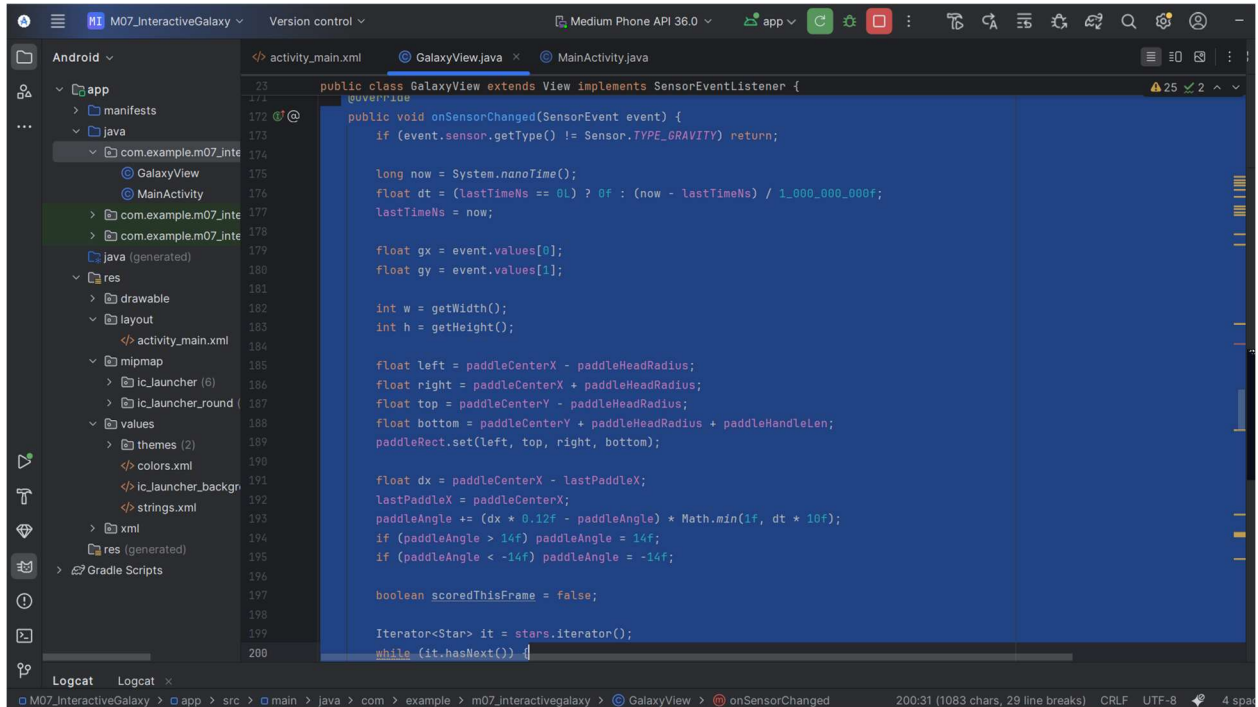
Touch Input Handling (GalaxyView.java)

This part of the code detects when the user touches or drags on the screen. The paddle moves or new balls are created according to the touch actions.



Drawing and Game Updates (GalaxyView.java)

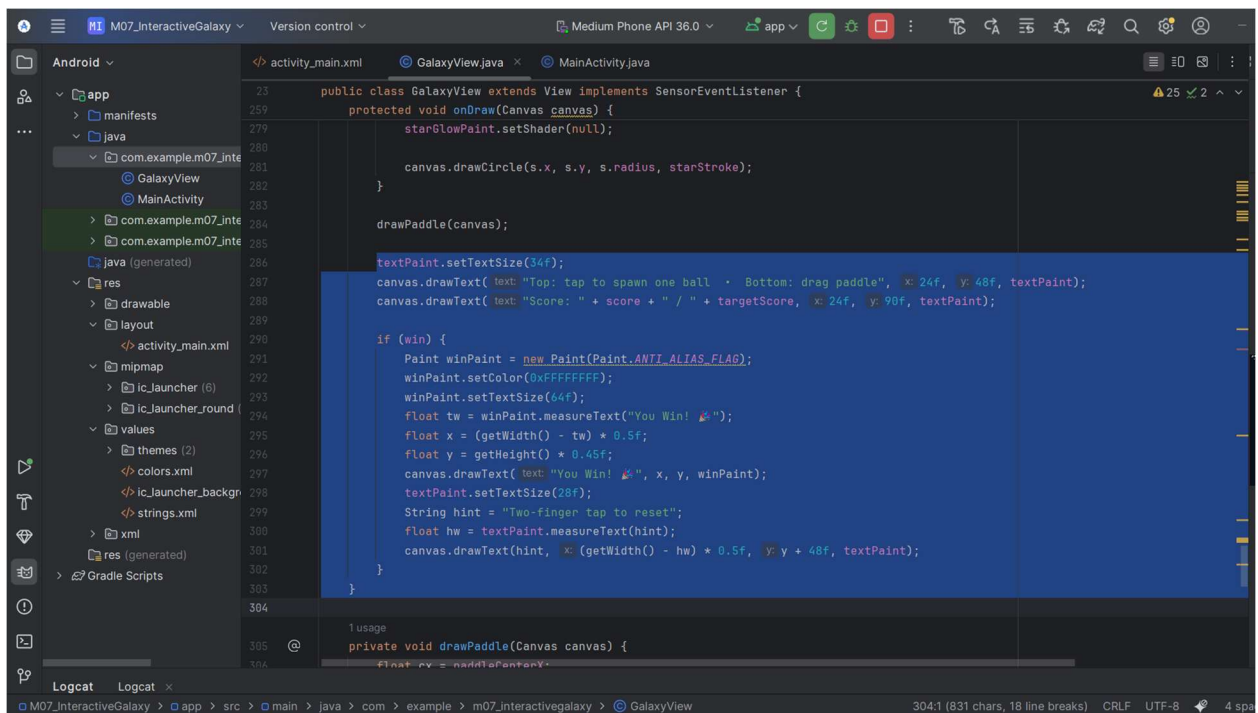
This section draws the paddle, balls, and score on the canvas. It updates the positions continuously to create the animation



```
23 public class GalaxyView extends View implements SensorEventListener {
24     @Override
25     public void onSensorChanged(SensorEvent event) {
26         if (event.sensor.getType() != Sensor.TYPE_GRAVITY) return;
27
28         long now = System.nanoTime();
29         float dt = (lastTimeNs == 0L) ? 0f : (now - lastTimeNs) / 1_000_000_000f;
30         lastTimeNs = now;
31
32         float gx = event.values[0];
33         float gy = event.values[1];
34
35         int w = getWidth();
36         int h = getHeight();
37
38         float left = paddleCenterX - paddleHeadRadius;
39         float right = paddleCenterX + paddleHeadRadius;
40         float top = paddleCenterY - paddleHeadRadius;
41         float bottom = paddleCenterY + paddleHeadRadius + paddleHandleLen;
42         paddleRect.set(left, top, right, bottom);
43
44         float dx = paddleCenterX - lastPaddleX;
45         lastPaddleX = paddleCenterX;
46         paddleAngle += (dx * 0.12f - paddleAngle) * Math.min(1f, dt * 10f);
47         if (paddleAngle > 14f) paddleAngle = 14f;
48         if (paddleAngle < -14f) paddleAngle = -14f;
49
50         boolean scoredThisFrame = false;
51
52         Iterator<Star> it = stars.iterator();
53         while (it.hasNext()) {
54             Star star = it.next();
55             star.move();
56             if (star.isScored()) {
57                 scoredThisFrame = true;
58                 stars.remove(star);
59             }
60         }
61
62         if (scoredThisFrame) score++;
63
64         invalidate();
65     }
66 }
```

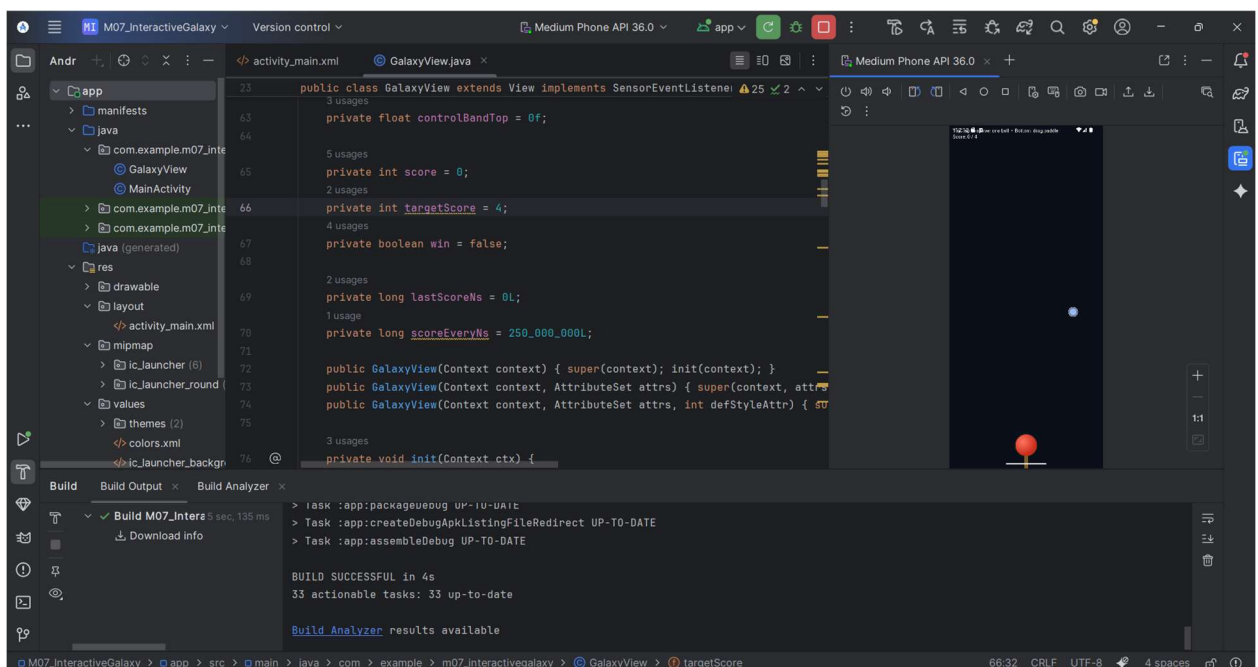
Gravity Sensor Handling (GalaxyView.java)

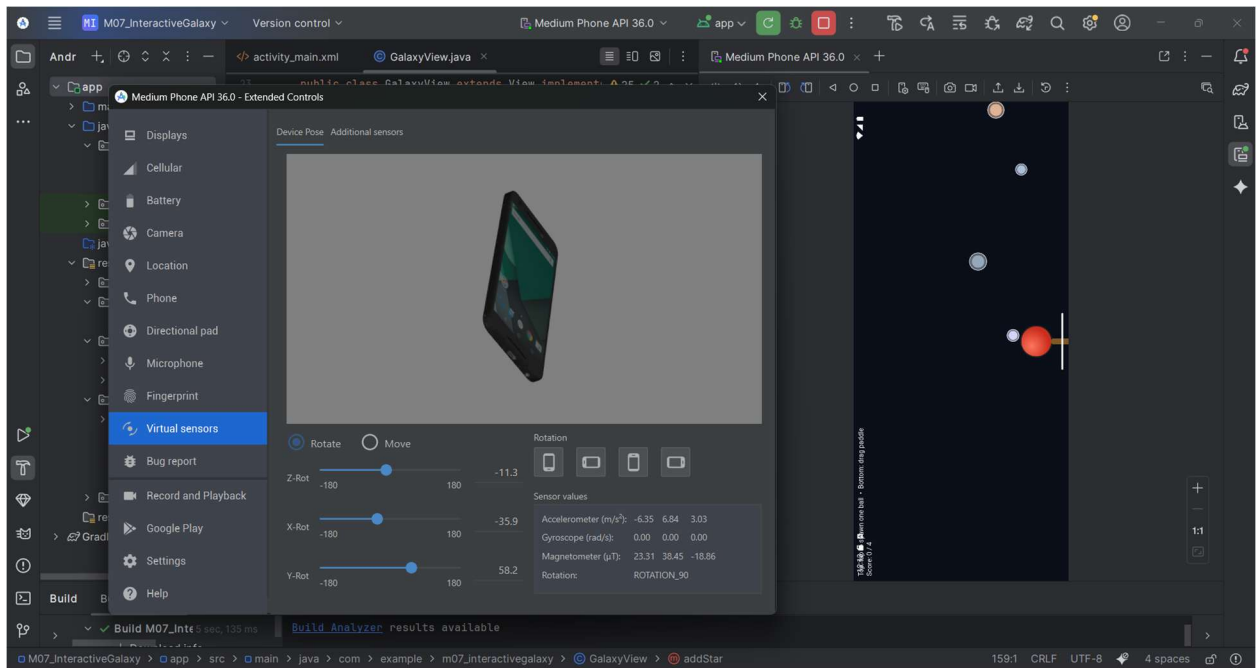
This part receives gravity sensor data and uses it to control the paddle movement in the game

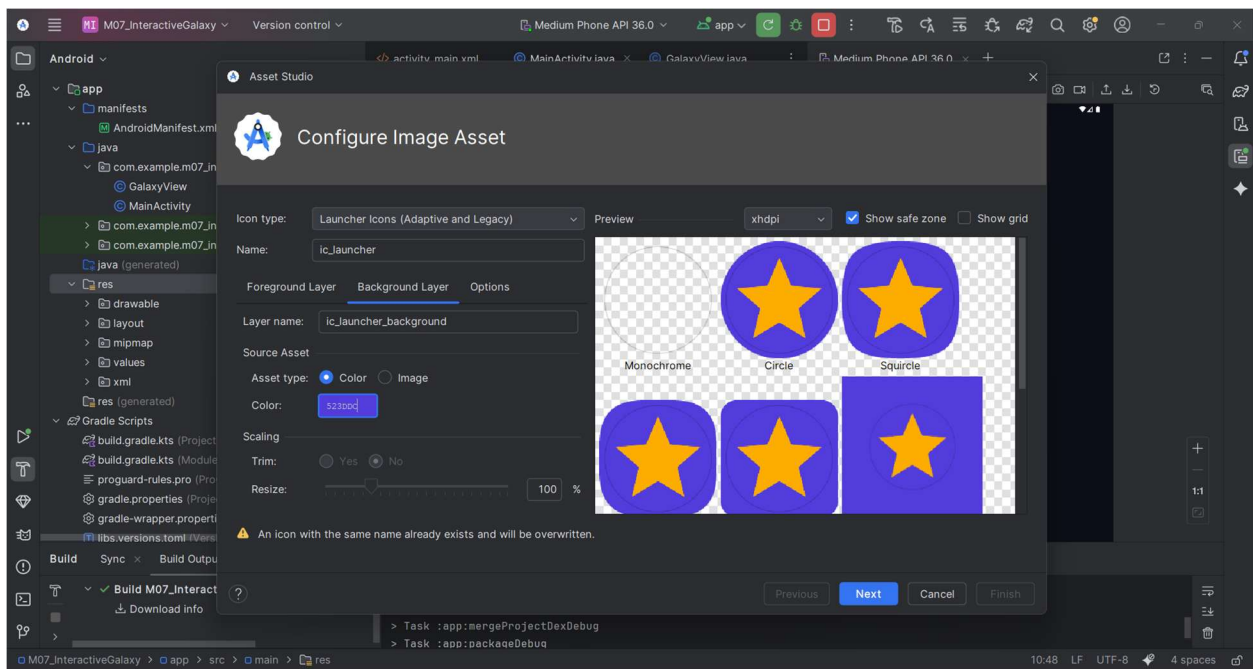


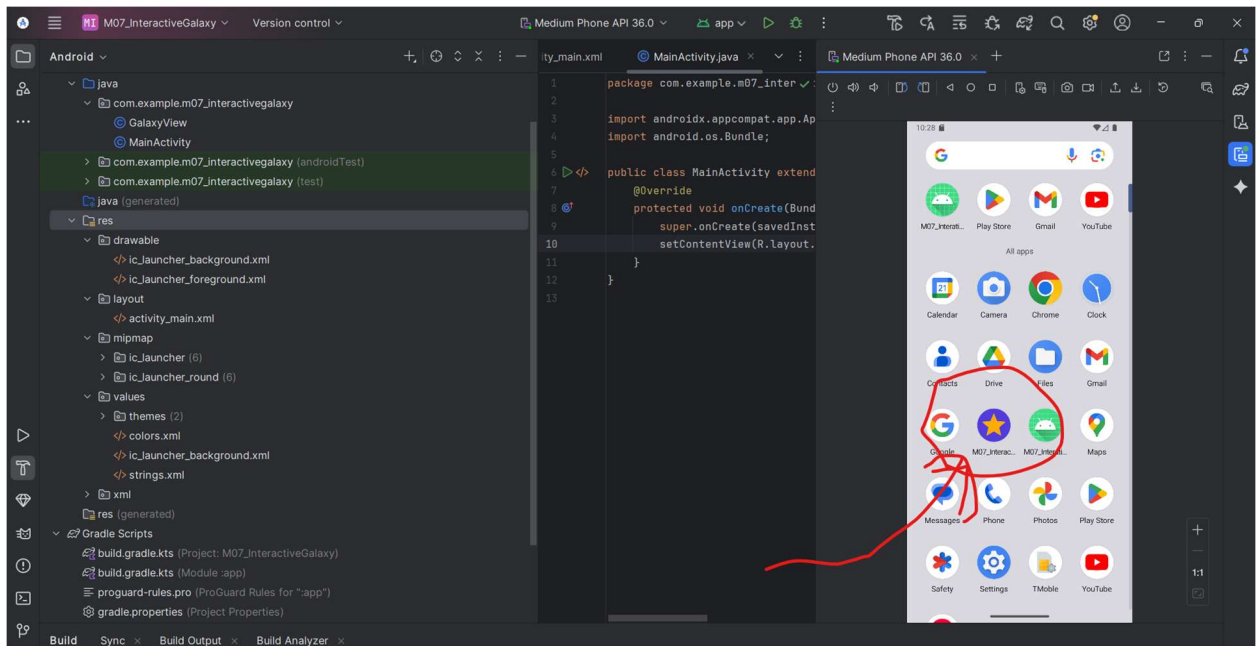
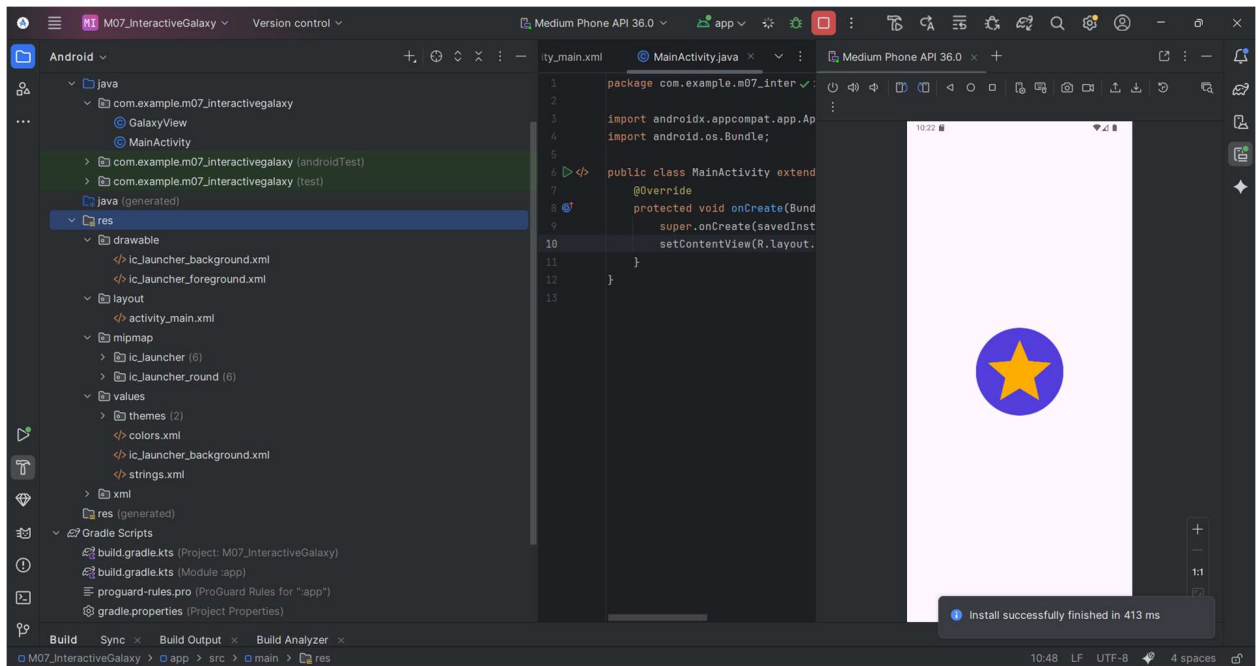
Scoring and Winning Logic (GalaxyView.java)

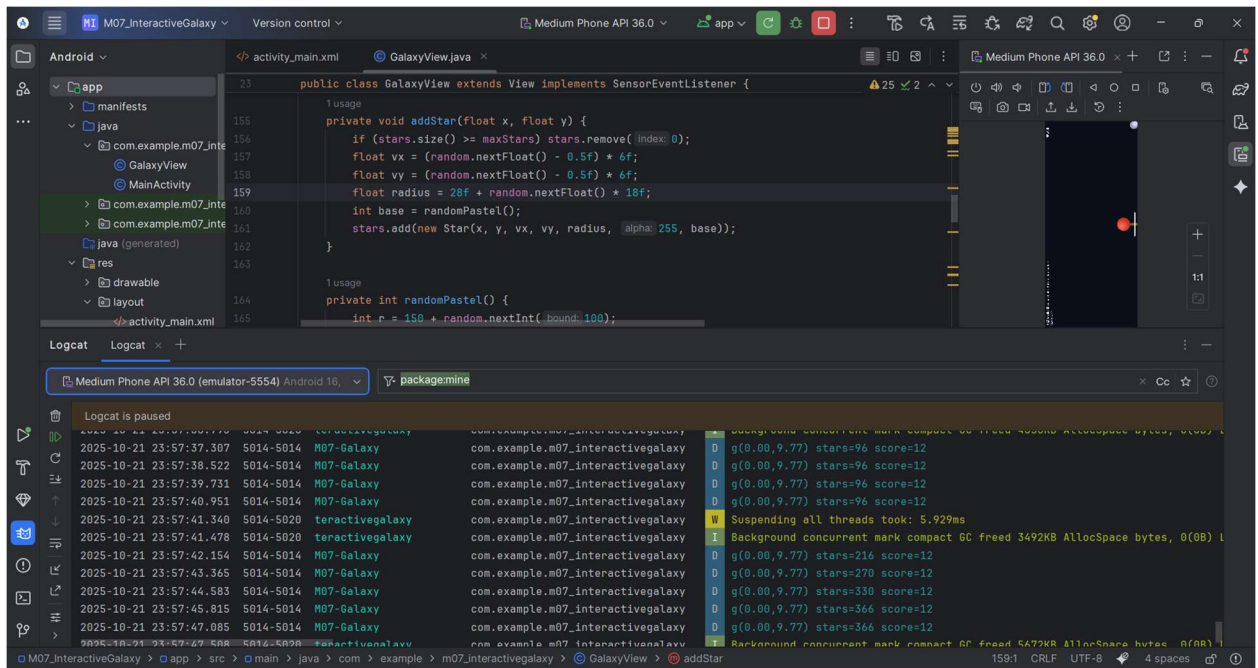
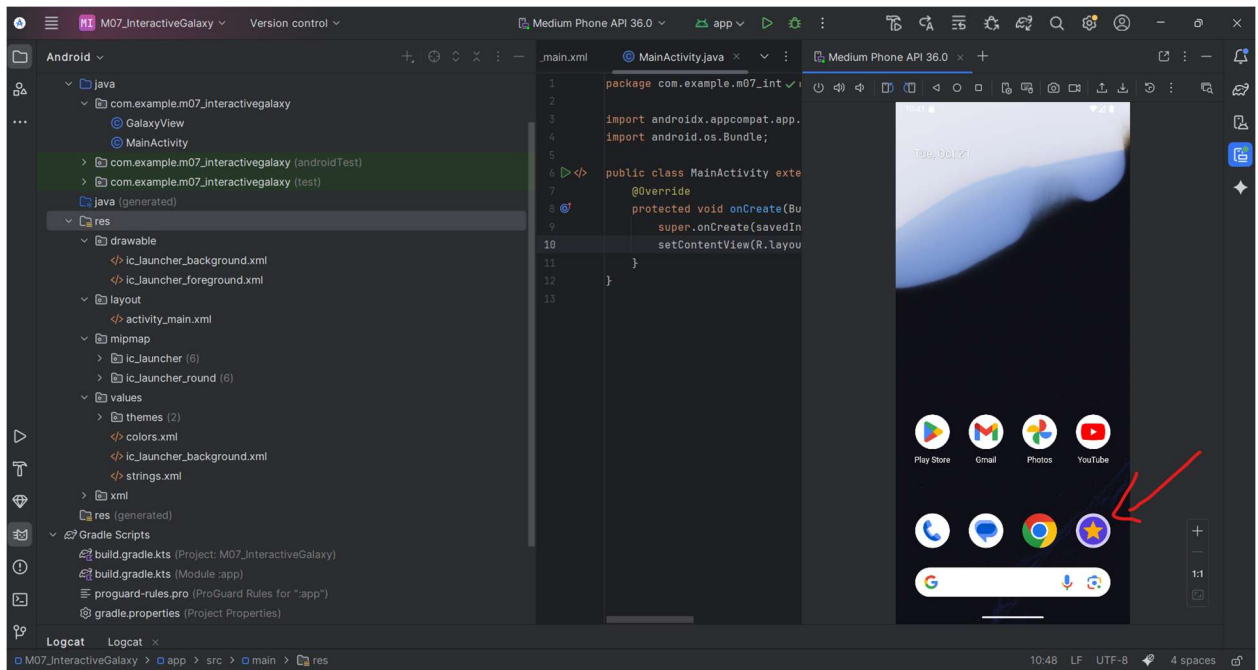
This part of the code increases the score each time the paddle hits a ball and displays a “You Win” message when the player reaches the target score

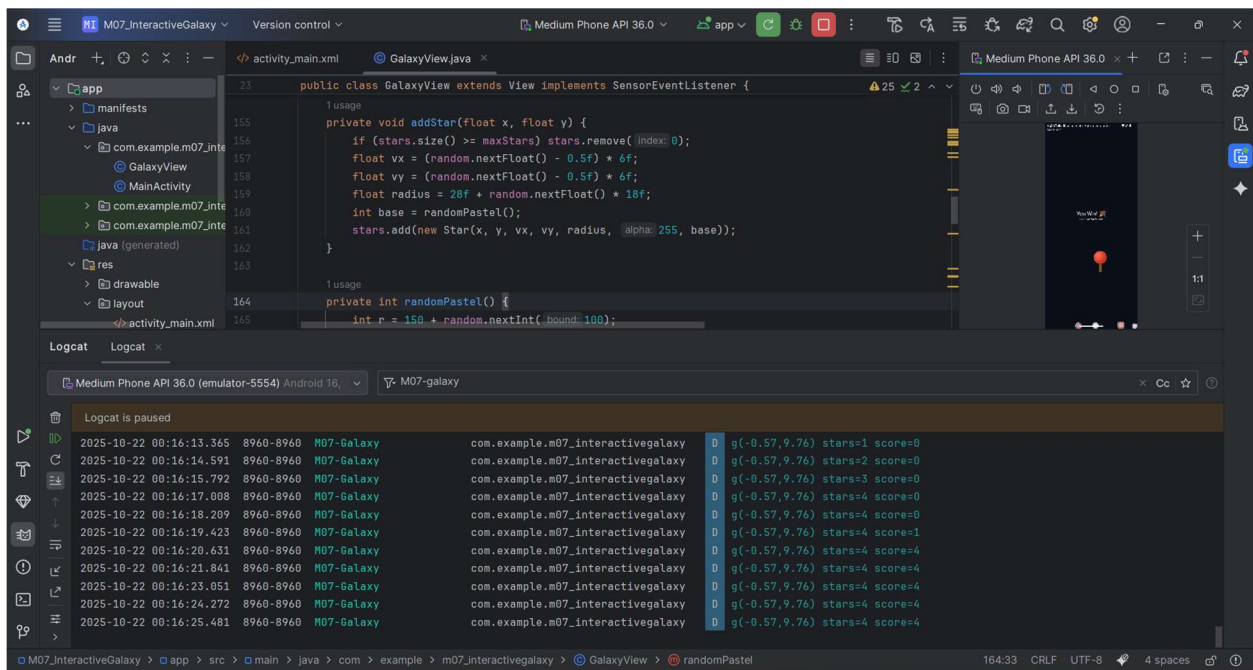
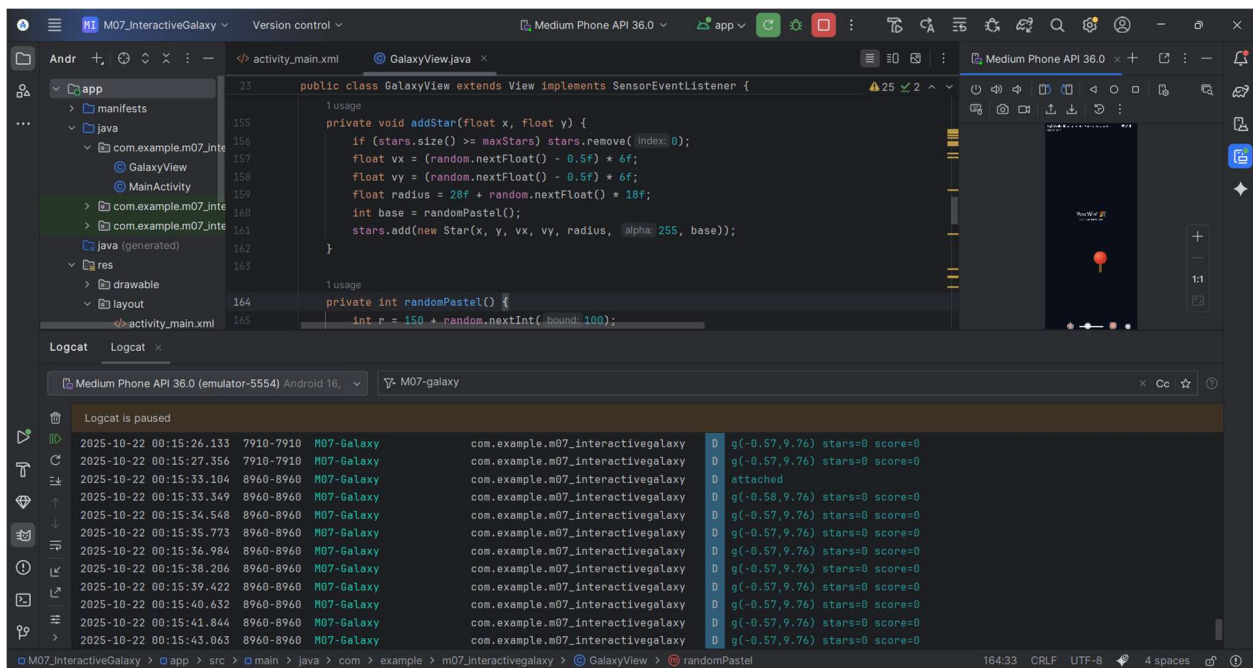












Logcat output confirming that the app runs successfully without any errors