

## Final Assessment: Sensor Game

**INDIVIDUAL EFFORT** – You may not work with or discuss this assignment with anyone other than the class instructional staff.

## Objectives:

Create a game application from scratch that relies on accelerometer input using what you learned in the second half of this course. For example, here's a screenshot of the sample app that we created.



When this application starts up two images are drawn to the screen. One image represents a bubble. The other image represents an obstruction, which we call the “spike.” Throughout the game, the spike will remain in its initial position. The bubble however may move during the game based on input from the device’s accelerometer. For instance, if the device is held with its top left corner pointed down towards earth the bubble should move towards the top left corner of the device. Notionally, this behavior is meant to simulate the pull of gravity on the bubble. The game continues until the bubble’s movement brings it in contact with the spike. When this happens, the bubble is removed from the screen and a popping sound is played.

More formally:

When the game starts -

1. Place the spike in random location on screen. The spike image should never change location. The spike image's size should be 64dp by 64dp.
2. Place bubble in a random location on screen (that is not in contact with the spike). The bubble image's size should be 128dp by 128dp. The bubble should not move until game play starts.
3. Begin receiving accelerometer readings. No other sensors should be used.

During play -

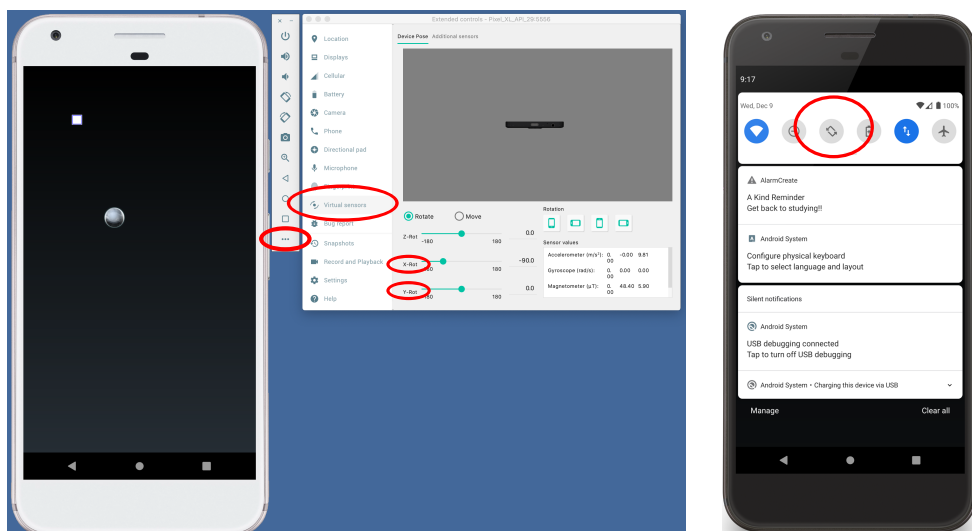
4. When an accelerometer reading is received the bubble should normally be moved based on those readings. The movement should be proportional to the accelerometer readings. The bubble should not move for any other reason.
5. One exception to the previous rule is that no part of the bubble should leave the visible screen. If the bubble reaches the edge of the screen, it should not move past the edge.
6. If the bubble comes into contact with the spike, the bubble should be removed from the screen

Ending play -

7. If the bubble comes into contact with the spike, the bubble image should be removed from the screen and a popping sound should be played.

## Notes

- Your app should be targeted at an API 29, Pixel 3 AVD.
- You can control the sensor input to your emulator opening the emulator's extended control panel and clicking on "Virtual sensors."
- The game should only process the X and Y components of the accelerometer readings. It should ignore the Z component.
- For testing purposes, it may be helpful to turn off "Auto-Rotate" in your emulated device.
- The two bitmaps and sound file are in the project's /res directories



## Rubric

- (20 points) Placing the spike and bubble images in random non-overlapping locations
- (30 points) Moving the bubble according to accelerometer inputs and nothing else
- (20 points) Ensuring that bubble doesn't move outside of the screen, when it reaches the edge
- (15 points) Removing the bubble from the screen if and when it comes into contact with spike
- (15 points) Playing the popping sound if and when it comes into contact with spike

## Submission

You will find this writeup within the FinalAssessment directory of your class repo. Your work is due by 10:00am on Wednesday, December 16th.