Cottontail Carl’s Continuous Cardio

Nick Rekuch

SDEV 264

I wanted to take a moment to type up a few closing thoughts about my project, and how it relates to the class. I had a lot of fun learning about Android, and a few headaches too.

Looking over my project now, I hate it. There are really two prongs of problems with it. Firstly, it’s just not a good game. The special jumps are almost never useful and the later levels are horribly unfair and the only way to progress is get hit over and over until enough carrots spawn that you can run through the stage. I’m not too worried about that, because this was more of a “learn how to Android” project than a “make a fun game” project. Still, if I was going to expand this into a release it would need to be addressed.

The other, and main, problem is that I built so much of the program before I had any experience with Android at all. In addition, I’ve never used Java before and have been talking an online C# class as a co-rec with this one. There are some true mistakes in the structure of my project that make it hard to work with. One example is that I have the world class in the same file as the game loop! (I’ve been calling it reverse-encapsulation while muttering to myself.)

In hindsight I could transfer a lot of the game state information to series of shared preferences objects- something I’ll need to do in future projects that allow the user a real host of options like changing font size and stuff beyond simply turning sound on and off. These problems are most noticeable in the code surrounding my Fragments that manually assign a ton of variables to them each time they’re created.

The biggest problem my project doesn’t solve is multiple device sizes. I feel like the main two solutions I could try are 1) keeping all the game action on a fragment that is sized in the XML and just letting the size of some kind of border graphic vary by device (with different layout files) or 2) programmatically stretching everything so that it is a fixed percentage of the screen rather than a number of pixels. Even the movement code counts in pixels rather than some standardized unit, which is a mess. I don’t need my game to play on watches, but I don’t want it to get weird if the user uses a tablet. I know the phone games I play all act the same on variously sized devices, so there must be some best practice to address this that I just haven’t learned yet.

ANYWAYS, thanks for being the first tech teacher I’ve had that let me make whatever I wanted to. I learn so much more playing with the tools than by following specific instructions and it’s been a real pleasure to learn from you.