Post-Mortem Game Report

Luke Daschko
Morgan Lay
Wednesday, April 8, 2015

What went right for us?

We were pleased with a lot of what we implemented in the game. We were both very happy that the upgrade system was working without issue. The level select screen was something we were concerned about, but it ended up being a lot simpler than we thought. We originally had a full week allocated to the level select screen, but we were able to pound it out in a few hours and produce a sleek design.

Ultimately, the lives system, upgrade system, pausing, graphics and control animations worked great for us. All of the aforementioned mechanics in our game were implemented without problem in generally less than a day. The bulk of our time was spent on what was troubling us. The upgrade system worked perfectly with how we accumulate punk points. The mechanic added the perfect twist that we wanted, that suggestion to play edgy. Trying the game out as we made it, we could feel the upgrades on our character and were pleased with how it reacts.

The collision code we can't take a lot of credit for because Fisica has a built in mechanic that can deal with that. This being said, it worked out well for us and we were pleased with the results. We feel that the strongest thing we have going for us in Bike Messenger has to be the punk points. We are thankful that it turned out like it did. If we had flubbed that design, then our game would be nothing more than any typical endless game. The punk points along with the other features really sets it apart from the competition.

What didn't go our way?

Well there were a lot of ideas we had to throw away because of time constraints. Everything that we implemented into the game pretty much went out way, except for the random level generation. We struggled a lot of the time with finding a viable efficient way of randomly spawning random objects and moving them towards the player. Often we would encounter difficulty with lag, or would find objects spawning unfairly close to the player. This part of our game wasn't able to be implemented until the last week because it took a while for us to find a decent method. Even the current method occasionally spawns things in unfair positions, but we weren't able to do anything about that in the allotted time.

Among other things that didn't go swimmingly for us was a smooth background animation. Oftentimes for whatever reason on foreign computers the background wouldn't run smoothly. On the computer it was coded on, it worked fine though. We couldn't figure out quite why that was, but we managed to minimize the amount of lag that the user experiences. We explored whether it was lag after all, but the fps doesn't dip at all. We were quite frustrated that everything seemed to be fine, but wasn't working as expected.

Switching between game modes was problematic for us at times. The game at one point was very messy, all thrown together in a giant heap of code. When we were switching between levels we found that things weren't where they were supposed to be, or weren't working as intended. It took us a long time before we realized why objects weren't where they were supposed to be. We use millis to record time change, which in turn creates out velocity. When we were switching between levels, the time wasn't resetting, so the game was picking up as if it had just continued at that stage, which wasn't the case. We fixed the issue eventually by declaring millis before and after every function and

subtracting time to 'reset' our millis to 0.

Similar to the above issue, in the early part of our game we struggled to figure out the best way to control the character and to give objects velocity. Controls came shortly after, but we were recommended to use Fisica. Fisica didn't come naturally to us, it took a while before we understood the gist of how it worked. Once we got that library up and running, it was still a bumpy path with getting the mechanics of the pre-set levels to work as we desired. It even became tricky with the random generation too.

There were a few features we wish that we could have implemented. We were aiming for having 5-10 pre-set levels, but we found that a random level would be worth our time. Getting that up and working took the bulk of our time near the end, and when it came down to it, we had two complete fully functioning levels. With more time we wanted to add a map with a system similar to Angry Birds, where you have to get a certain number of punk points to move on. Unfortunately, that was too labour intensive so we ended up with two and a random generating one.

What would we change?

Honestly we would change very little of the mechanics, perhaps with more time we would figure out a better way of implementing them, but aside from that the ideas are great. Maybe we would approach it differently, instead of focusing on the finer details getting the spawning and everything working flawlessly before moving on. There are a number of things that we could do with additional time, namely add more levels and make the game more complete.

Final Product

We were aiming to make a runner based game that had a unique point system that was interactive with the player and engaging. We feel that we delivered a game that satisfied our expectations. The main components that made the game feel complete for us is the: random spawning level, punk points system and the level select screen. We had all the essential foundations of a game, and added extra components to enhance the experience.

Division of Labour

We didn't really follow a strict plan, but rather worked on things sporadically. The following is specifically what each of us did:

Morgan Lay	Luke Daschko
• Shared Title Screen • Movement and Jump Upgrading • Credits • Upgrade Screen • Level Select Screen • Instructions and various splash screens • Wrote Fiction • Shared random spawn • Lives System • Sounds • Bug fixing • Switching between game modes	• Shared Title Screen • Player Movement • Splash screens • Hard-coded map • Shared random spawn • Object images • Background • Animations • Bug fixing • Implemented Fiction • Fisica

The work was divided pretty fairly by the end. There are things we're both forgetting to mention, and we did the work the majority of the time together, so there was collaboration on everything. Nothing was added to the game without the others input.