

Tuning

Spread

Original Parameters	Value
Damage On Hit	5
Delay Between Shots	0.25
Velocity	50

I felt that the spread powerup was a little too powerful. When compared to the blaster, the damage output was a bit too much, and after getting multiple upgrades, it is able to destroy many enemies on the screen at the same time because of its spreading shots. I believe that this power-up needs more tuning.

Updated Parameters	Value
Damage On Hit	10
Delay Between Shots	1.25
Velocity	75

I believe the best way to fix this imbalance issue is to increase the delay between shots. However, to make up for this debuff, I think the spread should have its damage increased. I think the spread should act more like a shotgun to encourage a different play style. I decided that I should reward players for risking getting closer to enemies by enabling them to deal more damage up close. I also slightly sped up the velocity of the projectiles to compensate for the delay between shots. After making these changes, the spread powerup feels more unique, and it is not overly oppressive as compared to before.

Blaster

Original Parameter	Value
Damage On Hit	5
Delay Between Shots	0.2
Velocity	50

I felt the blaster overall was fine as it is originally, but I think it would be more interesting to change up the gun.

Updated Parameter	Value
Damage On Hit	3
Delay Between Shots	0.3
Velocity	60

As the default weapon and most likely to drop powerup (based on my game changes), I think it is fair to make the blaster less powerful than the other guns to encourage trying the different powerups more since the blaster is somewhat plain. I purposely lowered the damage and increased the delay to make the blaster somewhat unsatisfying to use. Therefore, it is there if you need it and are unlucky with powerup drops, but the player feels the need to switch to a better weapon type when possible, even if it hurts their damage output temporarily.

Additional Elements

Laser

Parameters	Value
Damage On Hit	0.175
Continuous Damage	1
Delay Between Shots	0
Velocity	90

Here, I created a laser weapon powerup. Essentially, the weapon works as an upgraded blaster by having projectiles continuously fired without delay, which is similar to how a laser would work. Essentially it does very little damage, but the longer the target is shot at, the more this damage stacks up.

Enemy

I have also created an enemy_5 which is based on the enemy_0 prefab. However, what makes this enemy unique is its rarity, characteristics, and score. I created enemy_5 as a rare enemy, and associated with that a score of 1000 when destroyed. Still, enemy_5 is not easy to destroy as it quickly reaches the bottom of the screen. Likewise, it is hard not to notice as it is brightly colored and has a trail behind it. Of course, since it is so rare and quite difficult to destroy, it also guarantees a power-up drop as well.

Feature Implementation

I have decided to add a score feature to the space shmup prototype. Based on the code provided, the enemies have a score attached to them, but in the provided base version of the game, there is nothing done with this score. Therefore, I have decided to add a scoring feature which counts up the total score. In addition to this, whenever an enemy is destroyed their associated score value is displayed momentarily on the screen.

At first, I thought this would be easy, but it was terribly difficult. I am not good at programming and wrapping my head around and reading documentations was unfun to say the least. Most of my problems came from static variables and void methods. I still do not understand all of that stuff and was fortunate enough to stumble upon a working implementation from trial and error. Originally, I had planned on creating text prefabs which would be instantiated when an enemy was destroyed. This worked somewhat, but for some reason the text would not display, and I spent hours trying to troubleshoot why. I am sure there is a way around it, but I was unable to find a working solution.

Instead, I opted for a more simplistic implementation. I chose to create a premade hidden text object, which then has its text and position updated based on the enemy destroyed. Surprisingly, it worked as I envisioned after some work. However, I had a new problem after testing this somewhat crude design. When the enemy generates a powerup the text score goes missing. After more debugging, I found the reason to be a conflict of two object sharing the same position. Therefore, I simply slightly offsetted the x and y position of the score text and that worked in the end.

My next problem was then making the text disappear after sometime. This required a bit of research and I was fortunate to stumble upon the coroutine method. It worked and I was able to hide the text after a set interval. After half a second passes, I set the text to become empty and reset other associated variables.