# Easy Tower Defense Readme

Hello! I wanted to do something midscale and did not want to spend my entire weekend so I limited myself to three and a half hours and tried to go as far as I can with a simple tower defense game. My goal was to add excitement once the game is complete but the time limit did not allowed me to ☺

I have used Unity 2018.2.1f1 for this project. I used Visual Studio 2017 and Resharper which helped a lot with speed. I have included the github repository I used in the email, it is public but if it can’t be reached, I can zip it and send it to you.

I didn’t spend much time to find and use many assets so there aren’t any models or any animations unfortunately. And no particles… The tower defense game has a spawn point, a tower and a base. Enemies which get past the tower hit their damage to the base and then die. The base has a certain amount of configurable health and when it runs out the game ends. I didn’t have time to implement UI but the game breaks and the debug log says “Game Over” ☺

The design is not perfect but I tried to use EntityManager for logic and it is the centralized point. There is an Entity hierarchy tree which has the Turret, Base, Spawner, Enemies and Projectile. They communicate with events to EntityManager and EntityManager handles sending these events to other Entities. This was a very makeshift MVC I tried to impose but I’m not super happy with it.

There are 2 types of enemies but it can be extended by creating more prefabs which contain an Enemy script. The green enemies are faster but they have low health and damage. The red enemies are slower but they have more health and they damage more.

The spawner increases the spawn count and the enemy speed increases after X seconds (configurable in EntityManager).

I chose a 1:1 aspect ratio for simplicity so if the game looks bad and you don’t get that aspect ratio, please change it to a square resolution.

The camera is unfortunately static, I would have used a nice and authored waypoint camera if I had some more time.

The configurable values are not centralized but they are on each entity. EntityManager also controls a lot of game values.

I had fun making this non-intractable game, I hope you have fun watching it ☺

My plan is to use this as a base and make a game based on my pet hedgehog Sonic (yes his name is Sonic ☺ ) I’m sure it will be way more polished than this.

Please let me know if you have any questions.

Cheers,

Utku