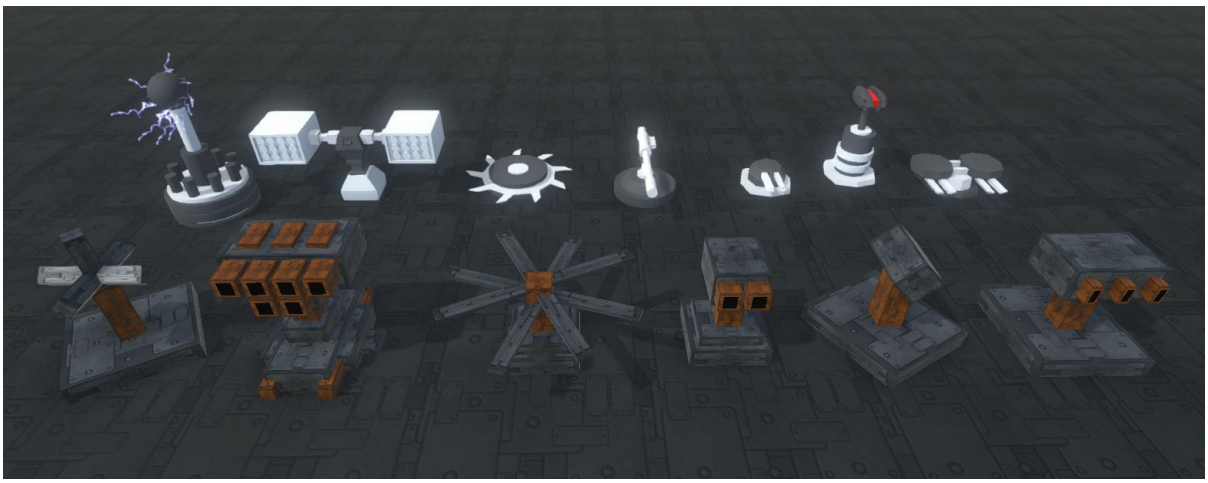
# Turret System



Turret System is a customizable script, allowing developers to easily set up their own custom turrets. It can be used in many cases such as shooter games, tower defense games, tank games as the gun of the tank (AI or player), and others that require turrets.

## Customization Options

The Turret System has a number of available customization options. Here you can see the explanation of all of them. Additionally the script is entirely commented, so you can take a look at it for more details.

**Damage Amount** - The amount of damage the turret deals.
**Rate Of Fire** - How fast is the turret shooting.
**Shot Range** - The range of the bullets, and sensor trigger.
**Turn Speed** - How fast does the turret turn towards the target.
**Error Amount** - Turret precision error. Higher values mean less precision.
**Projectile** - If the turret is a projectile type, assign the projectile prefab that will be fired, here.
**Hit Spark** - The prefab that will be instantiated at hit point when a raycast type turret hits something.
**Muzzle Flash** - Muzzle flash prefab to be instantiated on turret muzzle positions.

**Turret Pivot** - This is strictly a visual part. If the head has a rotating “stall” it’s on, then it will rotate only on the Y axis along with the head.
**Turret Head** - The turret head used for aiming. Raycast obstacle check comes from the center of it.

**Muzzle Positions** - Positions the bullets will be fired from. Used with “Is Raycast”, “Is Projectile” or “Is Melee”.
**Lines** - Used for laser type and shock type turrets, with “Is Line” or “Is Line Continuous”.

**Note: the following 5 options decide the type of the turret. Pick only one. In theory you can mix them, but it’s not intended to be used like that, or tested.**
**Is Raycast** - Raycast type turrets are “instant hit” bullets, and use raycast to deal damage.
**Is Projectile** - Projectile turrets fire physical bullets with their own damage dealing script. More on bullets below.
**Is Melee** - Melee is an experimental saw type turret which inflicts close range melee damage. (it will undergo a lot of refactoring, when animations customization option comes)
**Is Line** - Line is a shock type turret. Uses lines instead of muzzle positions.
**Is Line Continuous** - Line continuous is a laser type turret. Same as shock turret uses lines as visual representation of the attack. But unlike the shock turret it is continuous and deals damage over time, not by hit.

**Top Down Player Control** - Check if the turret is controlled by the player, from a top-down view.
**First Person Player Control** - Check if the turret is controlled by the player, from a first person perspective.
**Player Camera** - The player camera.
**Min Angle** - Minimum tilting angle limit.
**Max Angle** - Maximum tilting angle limit.
**Fire Button** - Button used to fire. Default is Fire1. If you want the turret to fire a secondary weapon (grenade launcher), just parent a smaller turret under it, with no health, turn speed set to 0, and a different button.

**Allow Tilting Forward** - Allows the turret to lean back and forth on the X axis, to aim at targets that could be on higher or lower ground.
**Ignore Raycast** - If checked, the turret will ignore any obstacles that might be between it and the target. If not checked, it will not fire if there is something in the way.
**Shoot Secondary Too** - If checked the turret will fire a target with a secondary tag too. That can be air units with “AirEnemy” tag.

**Obstacles Layer Mask** - Decides what the raycast can hit. Generally everything physical should be on this mask, while triggers such as the turret sensor trigger, on a “RangeCollider” layer, should be unchecked here, so the raycast can shoot through it.

**Enemy Tag** - The primary enemy tag to detect.
**Secondary Enemy Tag** - The secondary enemy tag to detect.
**Check If Audio Ended** - Firing sound plays every time the turret shoots. If this is checked it will wait before the firing sound finished playing, before playing it again. If not checked, and the next shot comes before the audio is done playing, it will interrupt the sound to play again.

**Damage Per Second** - Displays the calculated damage per second in editor, making it easier for you to balance the power of turrets in your game. Note: Projectile turret doesn't work, as the damage value is in the projectile prefab.

## v1.2

**Use Ammo** - Whether the turret will use and need ammo.
**Ammo** - Total ammo amount.
**Clip Size** - Amount of ammo in the clip
**Reload Time** - Time it takes to reload.

**Particles** - Array of all the particle guns on the turret. Place Particle Systems you want to be used as guns here. Remember to check collision, and “send collision messages” in the particle system settings.

**OnTriggerStay Optimization** - Decides how often the code in OnTriggerStay will be run. Higher values mean less ms for Physics.ProcessReports, but bigger delay in switching targets. If you're not experiencing fps issues, don't no need to change this.

**Recoil animators** - All the recoil animators you want to send the 'Shoot' trigger to.
**Reload animators** - All the reload animators you want to send the 'Reload' trigger to.
**OnStart animators** - All the onStart animators you want to send the 'Start' trigger to. The trigger will be sent when a target enters the collider.

**Recoil Trigger** - Trigger string parameter for recoil you want to send to Mecanim.
**Reload Trigger** - Trigger string parameter for recoil you want to send to Mecanim.
**OnStart Trigger** - Trigger string parameter for recoil you want to send to Mecanim.

## v1.3

**Aim Ahead** - Check if you want the projectile turret to estimate, and aim ahead of the moving target.
**Visual Particle** - Check this if the particle will be purely visual. Such as bullet casing shooting off the side.

**Use Camera For Raycast** - If you are making a first person shooter game using the turret script for the weapon, you might wanna check this. It will fire from camera, but muzzle will still use muzzle positions, and everything else works the same.

**Aim Button** - Button used to aim. Default is Fire2.
**Fire On Press** - Shoot on button press .
**Fire On Release** - Shoot on button release.
**Fire On Hold** - Shoot on button hold.
**Can Aim Before Fire** - Can you aim before its time to fire again?
**Aim Position** - Coordinates of the new position to move to when aiming.
**Aim Object** - What object are you moving/aiming with.
**Aim Speed** - Speed of aim transition.
**Aim Error Reduction** - How much to reduce aim error. The aimError is reduced to this number.
**No Target Random Movement Interval** - If set to something other than 0, the turret will move randomly when it doesn't have a target. This is how often it will move. Less is more.
**Custom Script To Send Message** - Script name of the hit object to send a message to. SendMessage is slower than GetComponent, so you're better off with override, or just adding your code to it. Search TakeDamage, to find the places damage is applied.

**Fire On Press Anim** - Animators you wanna send triggers too.
**Fire On Release Anim** - Animators you wanna send triggers too.
**Fire On Hold Anim** - Animators you wanna send triggers too.
**Aim Anim** - Animators you wanna send triggers too.
**On Shutdown Anim** - Animators you wanna send triggers too.

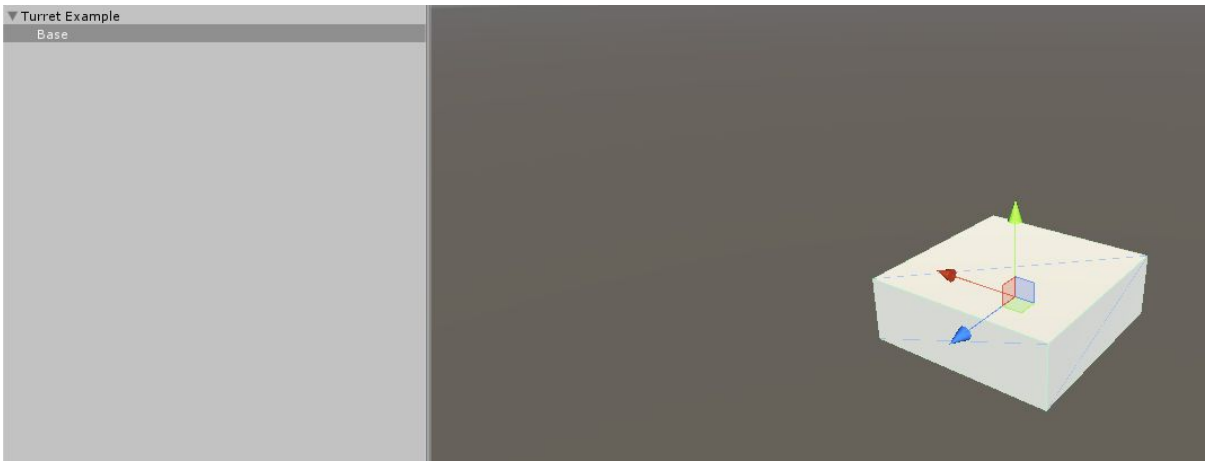
**Fire On Press Trigger** - Trigger string parameter for recoil you want to send to Mecanim.
**Fire On Release Trigger** - Trigger string parameter for recoil you want to send to Mecanim.
**Fire On Hold Trigger** - Trigger string parameter for recoil you want to send to Mecanim.
**Aim Anim Trigger** - Trigger string parameter for recoil you want to send to Mecanim.
**On Shutdown Trigger** - Trigger string parameter for recoil you want to send to Mecanim.

# Basic Setup

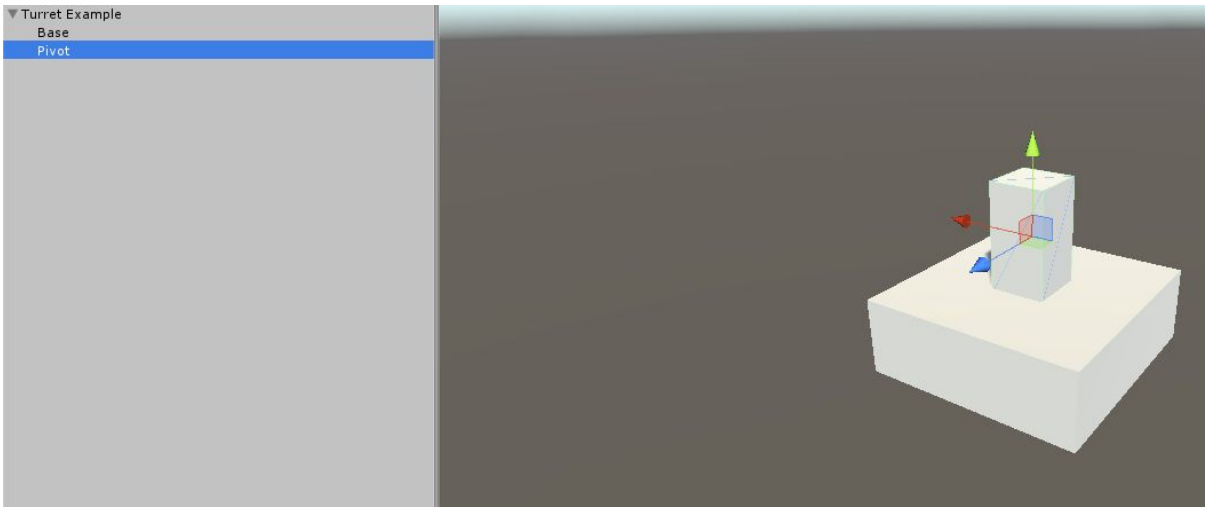
**Note: A really important thing is to make the front of the head the Z axis, and top, Y axis. You can change that either in the modeling software you're using, or parent it under an empty game object, as demonstrated in many of the examples, with names such as "PivotCorrection" and "HeadCorrection".**

Now let's go over setting up a basic step by step raycast turret from scratch, using Unity's primitives. Best thing to do is parent the turret model (or models) under an empty game object, with front going along the Z axis.

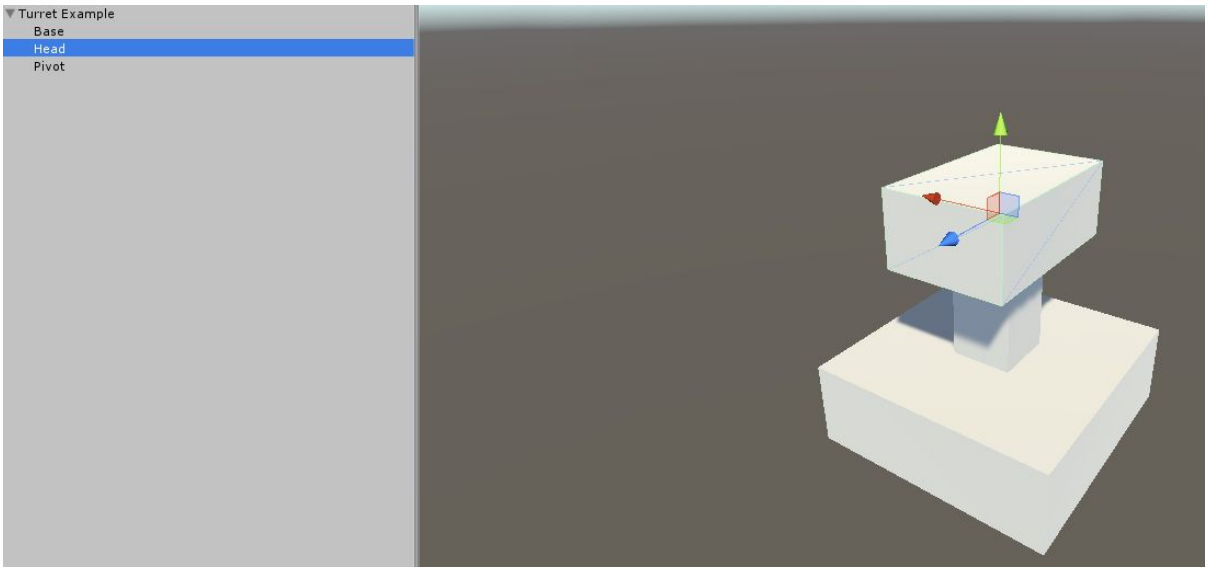
1. Create empty game object, call it "Turret Example". To avoid unwanted artifacts, make sure the scale is (1,1,1).
2. Create a cube, call it "Base", and parent it under the empty game object. Don't forget to center it, at least approximately.
3. Rescale it to look the way you want as a base of the turret. This part is purely visual.



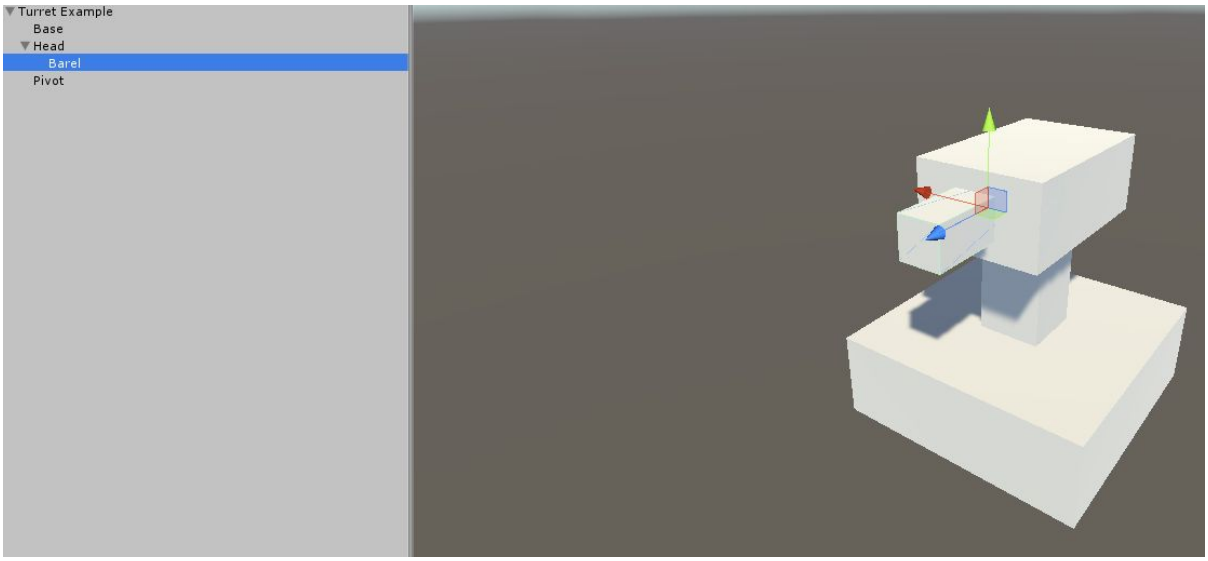

5. Duplicate the base, or create a new cube and call it "Pivot", rescale it to look like a pole or stall of sorts, and parent it under the Turret Example. Make sure the front is going along the Z axis, and Y is the top. Pivot is also purely visual, and it will be rotating on the Y axis.



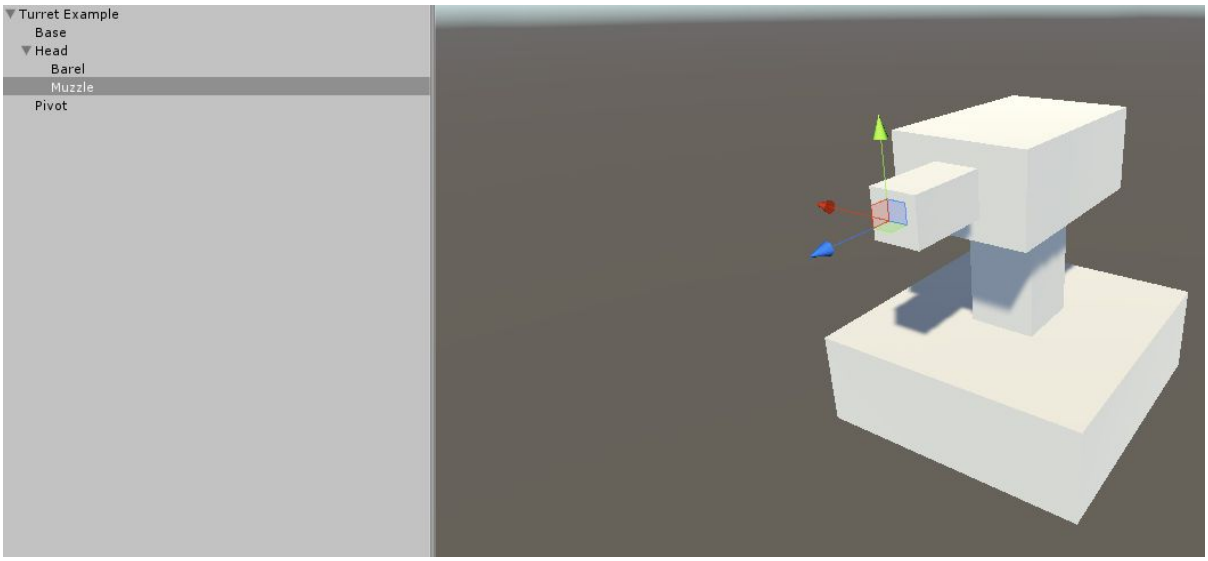

6. Duplicate, or create a new cube and call it “Head”, rescale it to look the way you want, parent it with the rest, and same as with pivot, make sure the front is the Z axis arrow. Head is functional, and where the aiming, and obstacle check, will come from. It is necessary if the turret will be turning to aim.



7. Duplicate, or create a new cube and call it “Barel”. This time parent it under the Head. It's a purely visual part.

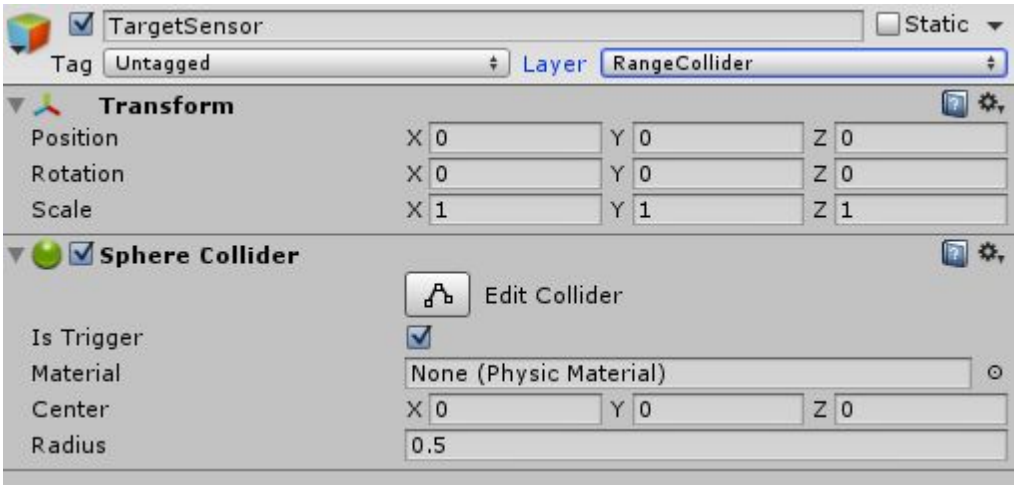



8. Create an empty game object, parent it under the Head, and call it “Muzzle”. This is where the bullets will be coming from, and where the muzzle prefab will be instantiated. Position it to be sort of in front of the barrel.

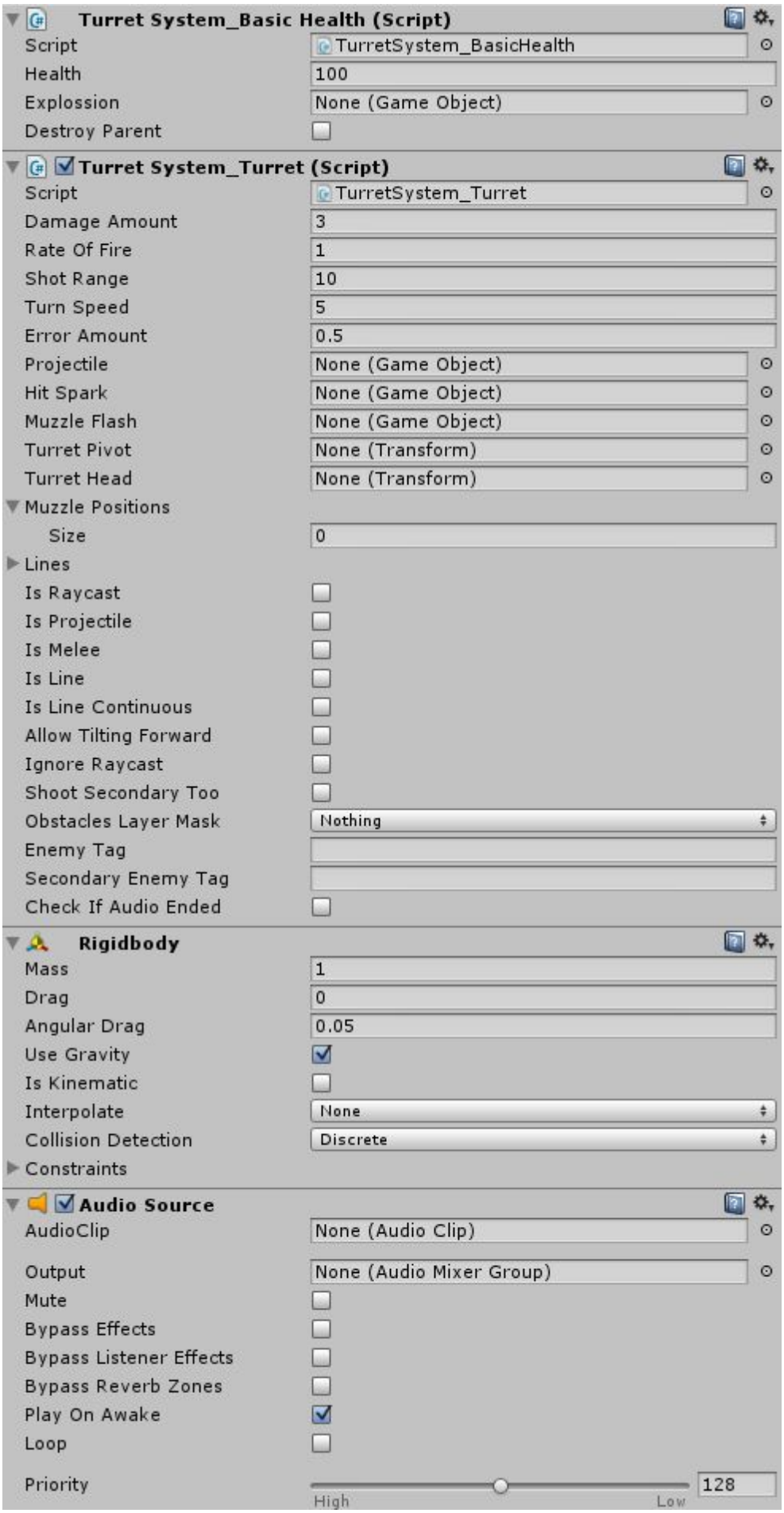



9. And finally create another empty game object, parent it under Turret Example, and place it at the center (0,0,0). Call it “TargetSensor”. Now add a sphere collider to it, and check the trigger box. This is the trigger that will detect the target. The radius doesn’t matter, the script will change it based on the shot range, but you can keep the radius minimal for a clear view.

10. Create a new layer, and assign it to the TargetSensor, and ***only*** to it, not the parent nor any other object. This is so the raycast bullet can shoot through it, because normally raycast treats triggers as solid colliders. In a bit we'll set it up in the obstacle layer mask in the script.




11. Next drag and drop the TurretSystem_Turret.cs script from "Turret System/Turret System Scripts" onto the Turret Example. The script will automatically add the necessary rigidbody, heatlh and audiosource. If you won't be using the health on turrets, you can comment that line out, it is one of the first few lines. You should get something like this.



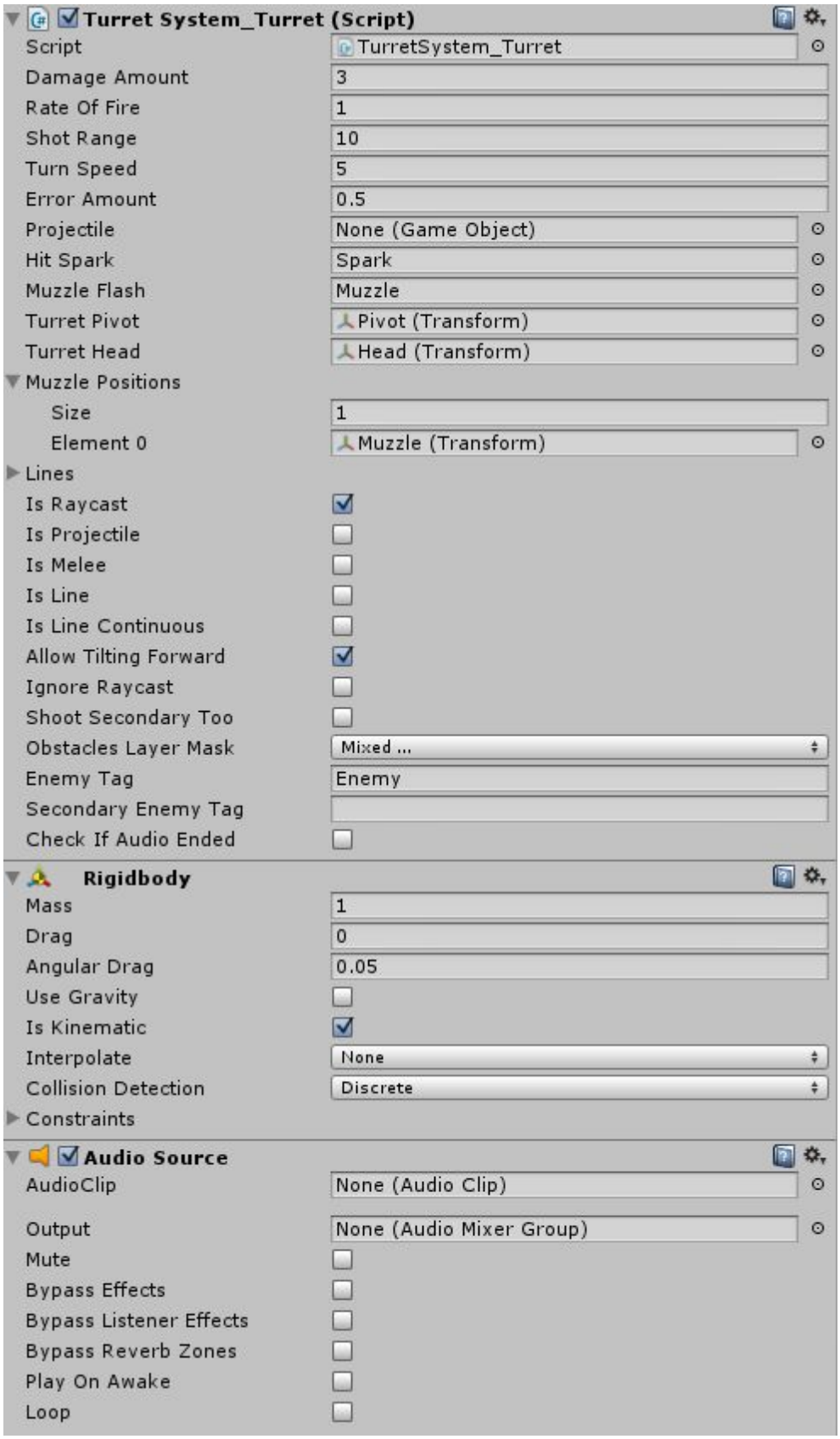
12. Now lets begin setting up the script. First thing we need to do is turn off “Play On Awake” on the audiosource, and on the rigidbody turn off “Use Gravity” and turn on “Is Kinematic”, so our turret can interact with physics, but isn’t moved by them. That’s project specific, but for our example, we’ll set it up like that.

Add the Spark prefab from the “Turret System/Prefabs/Particles” to the “Hit Spark” slot, and the Muzzle to “Muzzle Flash” slot. And simply drag and drop the pivot, head, and muzzle transform, into the appropriate slots.

Check “Is Raycast” and “Allow Tilting Forward”, so the turret is a raycast type, and the head can lean back and forth when aiming.

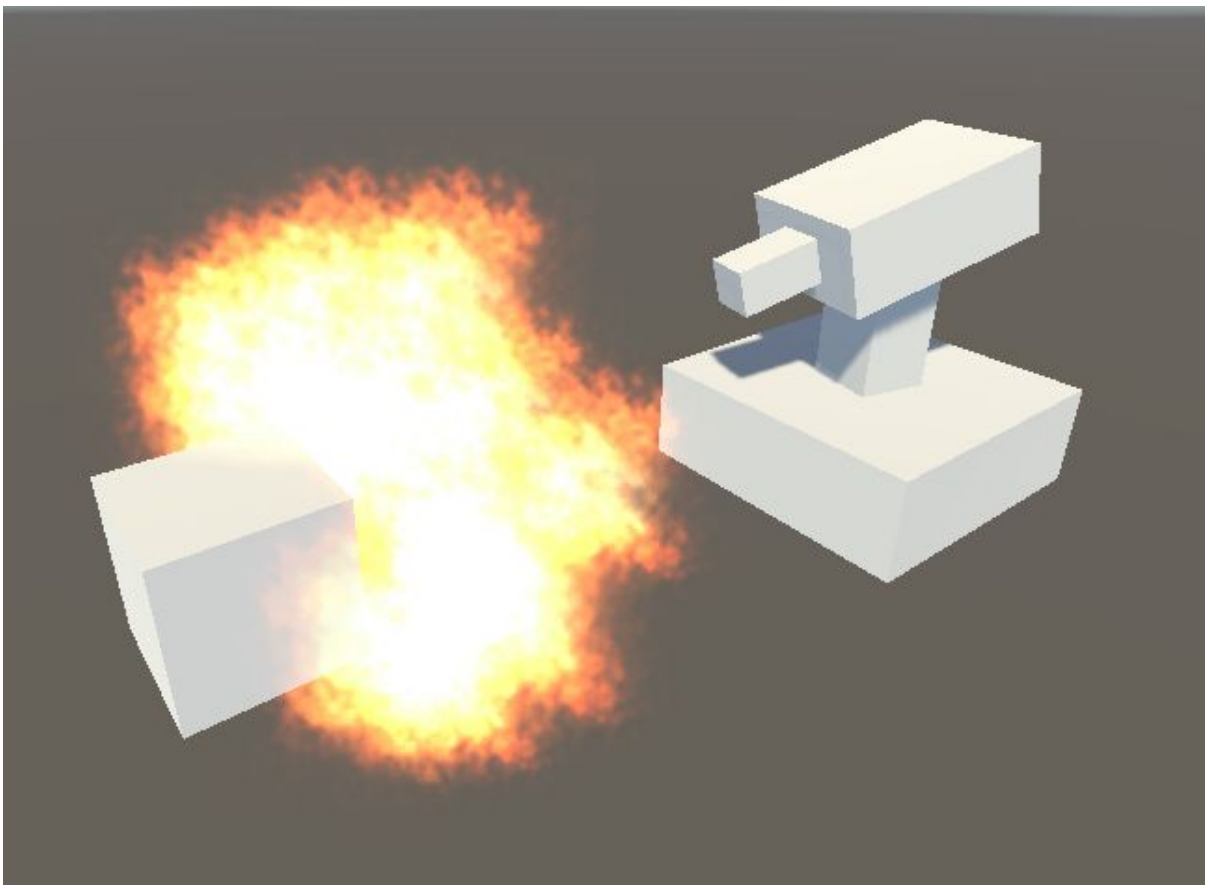
Now the **very important** part is to change the “Obstacle Layer Mask” to Everything, and uncheck the RangeCollider, and any other collider you don’t want the raycast hitting.

Type in “Enemy” in the “Enemy Tag” slot, and the turret is set. You should have something like this.



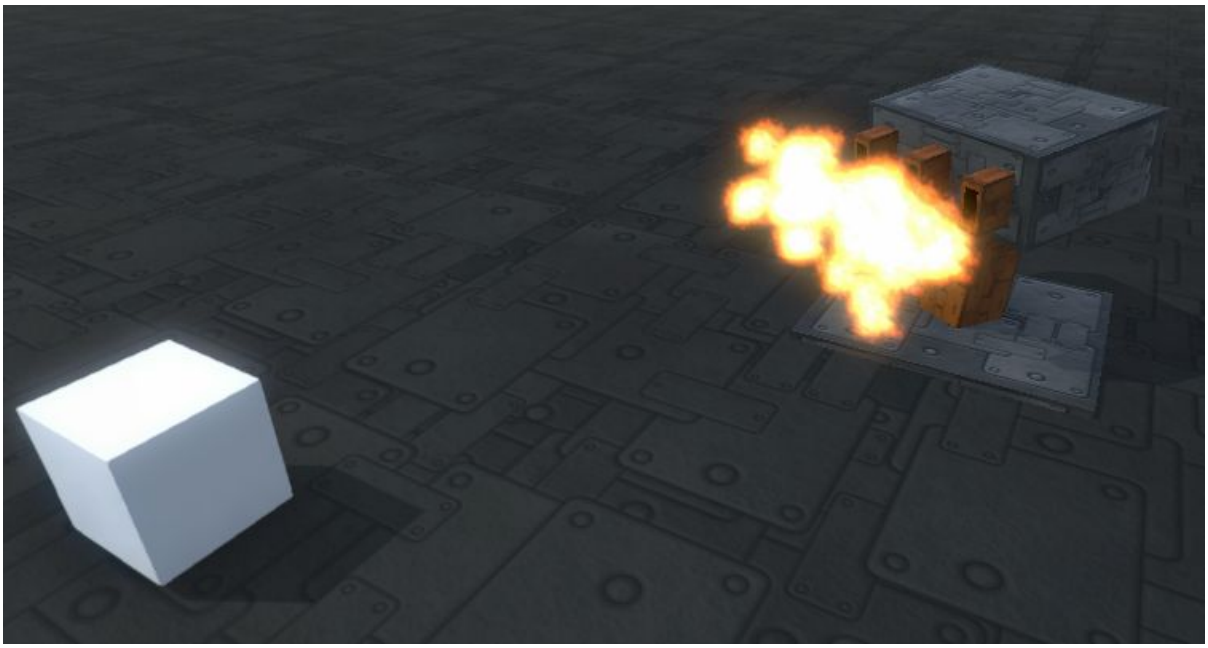
13. To test it, just create a simple cube, place it close to the turret, change its tag to “Enemy” and add the “TurretSystem_BasicHealth” script located in “Turret System/Turret System Scripts/”, and play the scene.

Additionally you can play around the settings. The turret has a high error amount float, so you can reduce it to make it more precise, give it more damage, reduce the fire rate, whatever suits you. :)



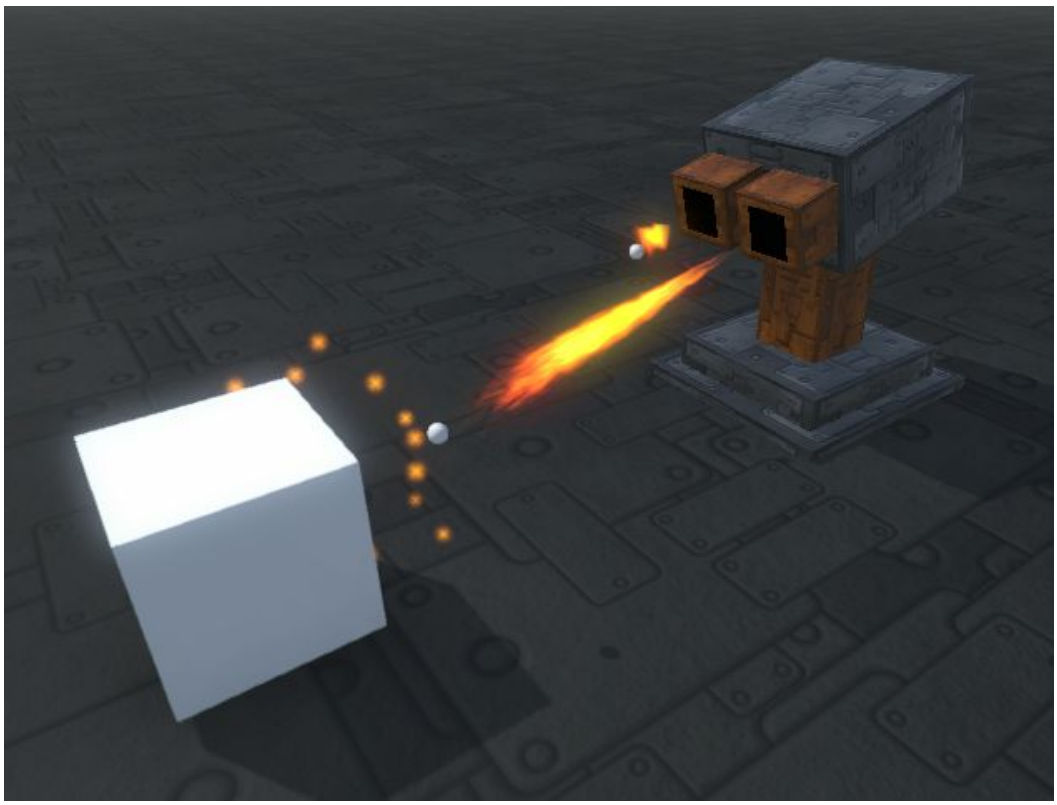
## Raycast Type

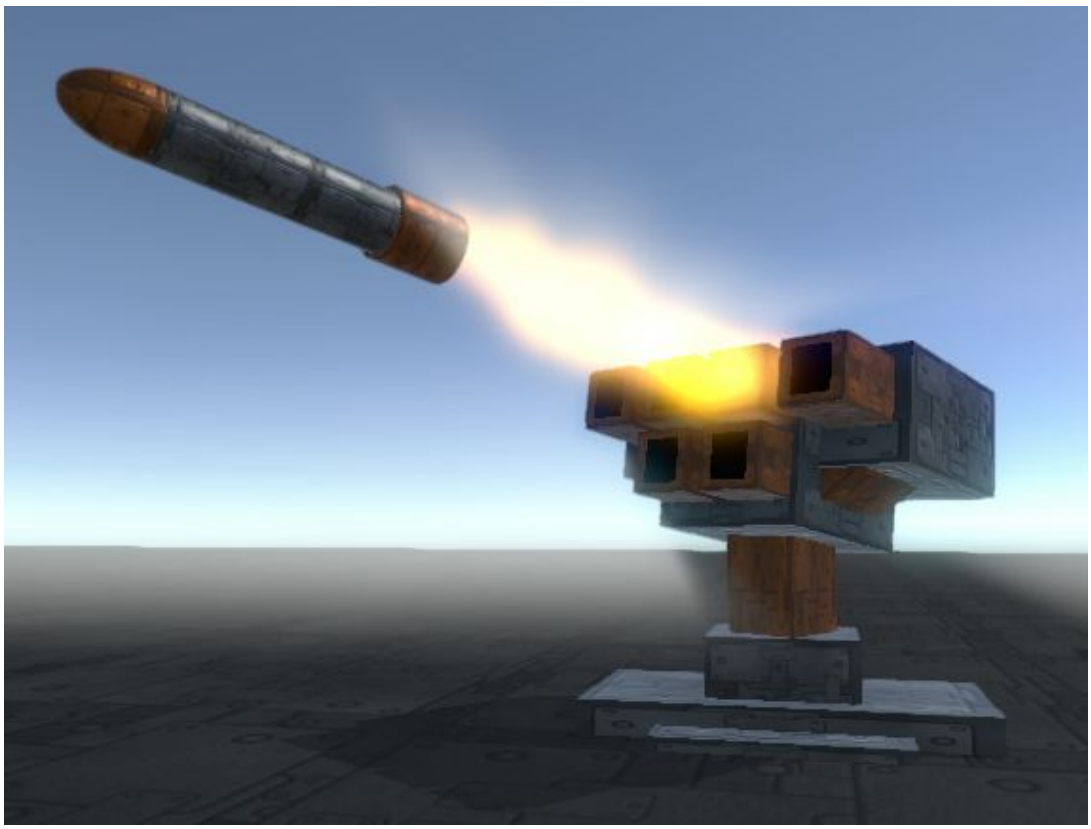
Raycast type turret shoots an “instant bullet” in the form of a raycast. As the one we made in the example tutorial.



## Projectile Type

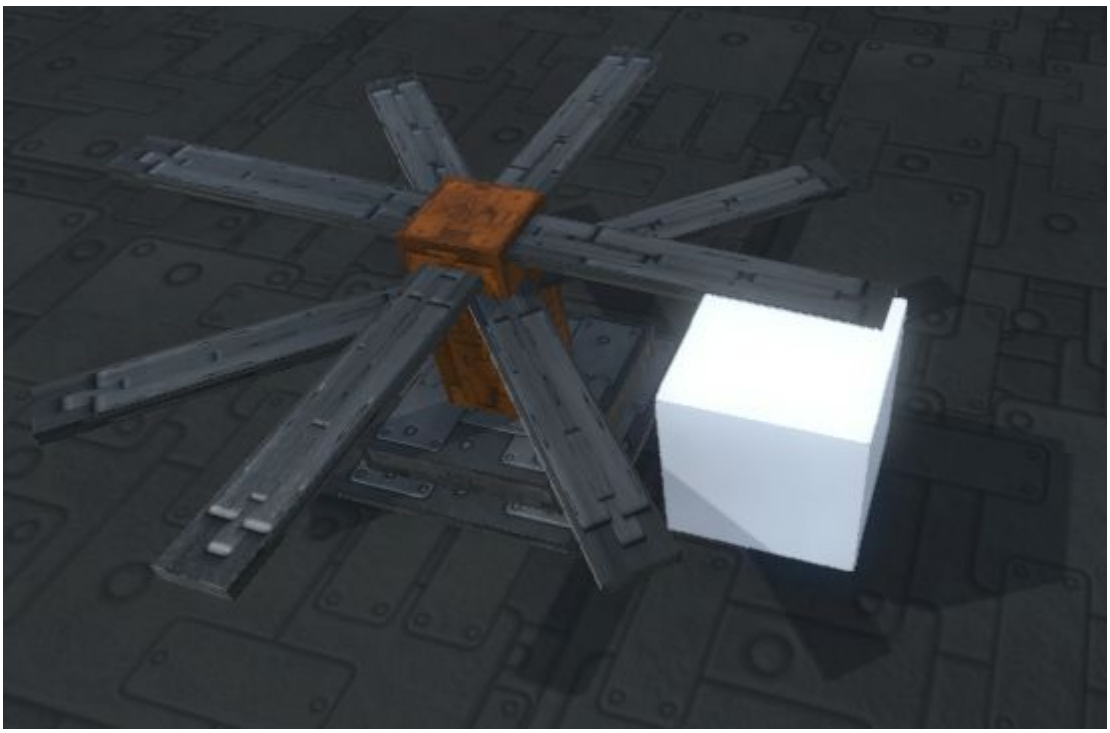
Projectile type shoots a physical projectile, with its own damage dealing script.





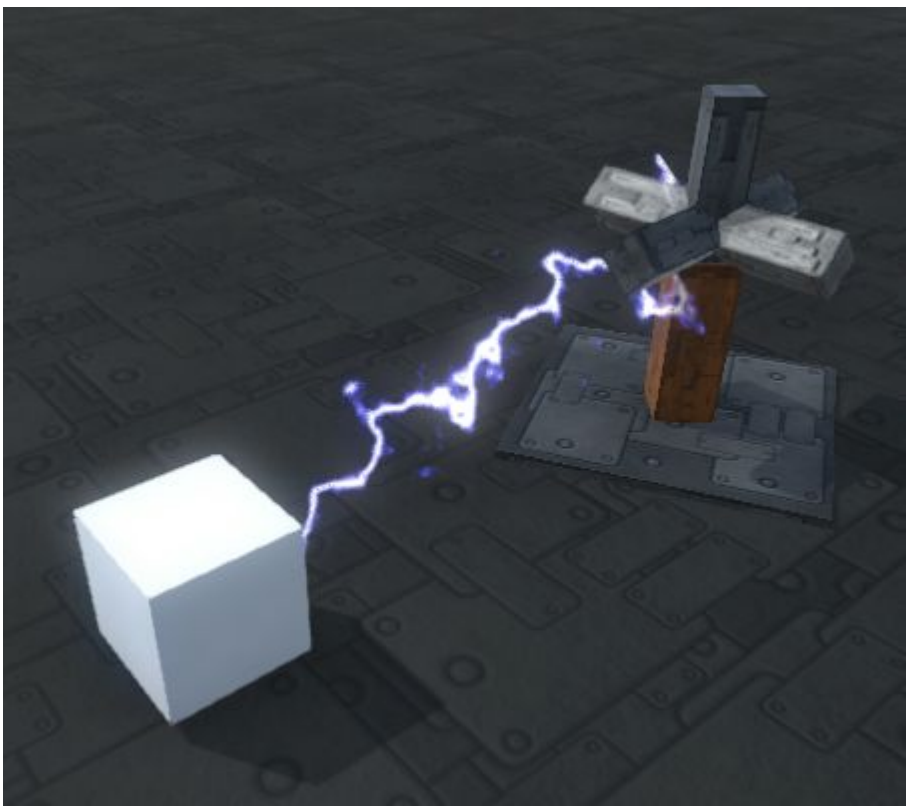
## Melee Type

A saw type close range attack turret. Experimental. Will be refactored in future.



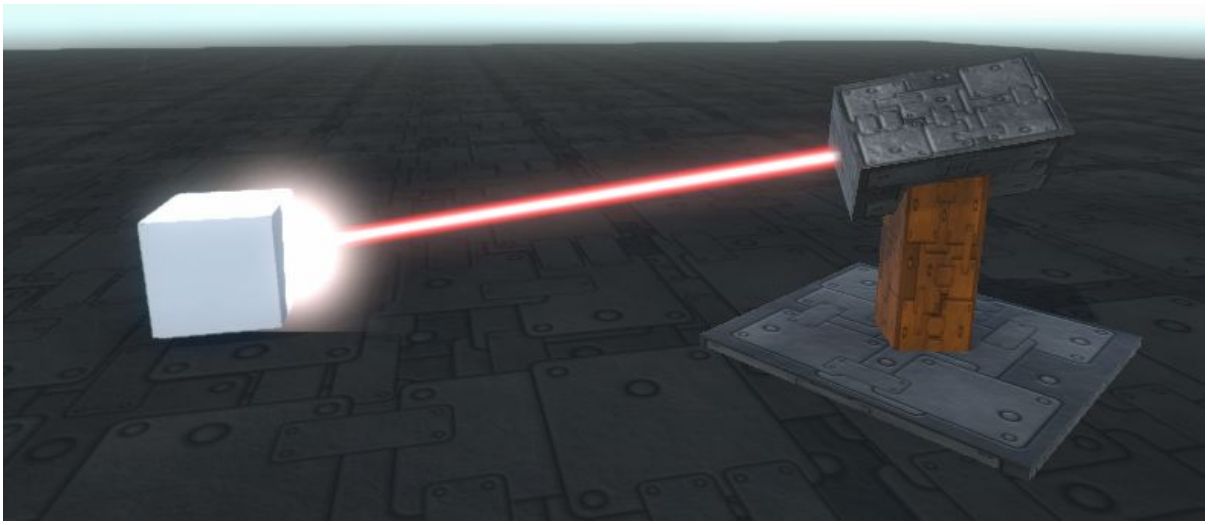
## Line Type

Line type is shock type turret with a lightning type shock attack.



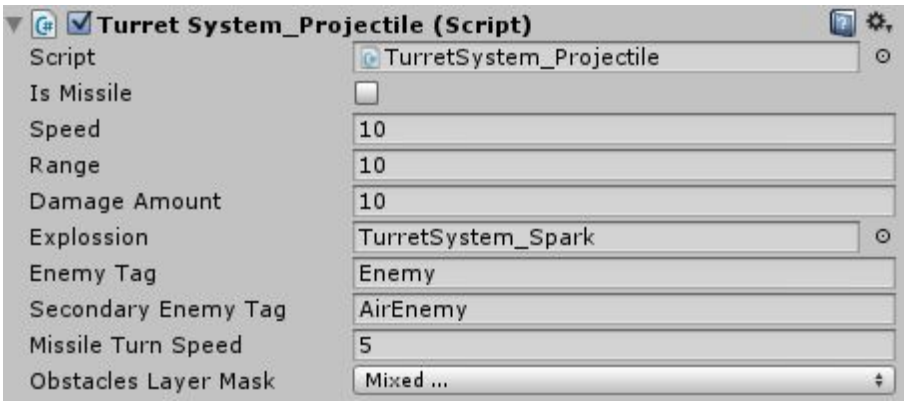
## Line Continuous Type

A laser type turret, that attacks continuously and inflicts damage over time.



## Projectile Script

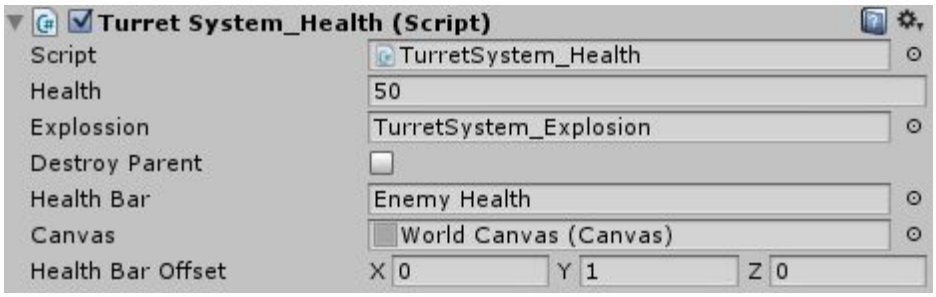
The projectile script has a couple options. It can either be a homing missile, or a simple physical bullet. Both collider and trigger works. The examples are all triggers. Same as with turrets, in the obstacles layer mask, check everything except the “RangeCollider” layer.



## Health Script

Health script you can add to the enemies. Once the health reaches 0 it will call the Explode() function, destroy the object, and instantiate an explosion if there is one.

As of v1.1 there's an optional health bar too. Simply drag a image prefab, and a world canvas into it, adjust the offset, and that's it.



**Update Logs**

Version 1.1 Log
- Prettier inspector
- Tooltips. Hover over any option and you'll get a description of it.
- DPS feature (shows damage per second of your turret)
- Top-Down player controls
- First person player controls
- Health bar for the TurretSystem_Health script
- Simple handy load scene script for buttons (put it on empty GO or canvas, and drag it into buttons)
- Added menu to test out scenes faster
- Added 9 sounds
- Improvements
- Fixes, fixes, fixes :)

Version 1.2. Log
- Optimized (added a frequency setting for OnTriggerStay(), making it almost 70% faster)
- Added DPS for projectile type too
- Added animation support
- added ammo, clip ammo, and reload time
- added turret angle limits for AI too
- added particle turret (2 examples, flamethrower and gattling-laser-gun)

***Damaged Grounds***

**Asset Store**
**Twitter**
**Facebook**
**Youtube**