

FISHING GAME TOOL 2D

Fishing System Development Tool for Unity

USER GUIDE

Release 1.1

11.2023

Created by Paweł Mularczyk

This manual, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. The content of this manual is furnished for informational use only, is subject to change without notice and should not be construed as a commitment by its authors. The author assumes no responsibility or liability for any errors or inaccuracies that may appear in this manual.

FISHING GAME TOOL 2D

Table of Contents

Getting Started

Overview.....	3
---------------	---

Fishing Rod 2D

Configure Fishing Rod 2D.....	4
-------------------------------	---

Fishing System 2D

Configure Fishing System 2D.....	8
Configure The Fishing Spot.....	18
Testing.....	21

Additional components

Character Movement.....	23
Character TD Movement.....	24
Camera Control.....	25
Simple UI Manager.....	26

FISHING GAME TOOL 2D

Overview

Fishing Game Tool 2D is a comprehensive tool that makes it easy to integrate a fishing system into a wide range of projects. Whether creating a survival game, a casual production or a complex RPG, this tool provides the necessary framework. The tool is designed for a two-dimensional side view or top down view game.

With its easy customizability, it significantly simplifies and speeds up the integration of a fishing system into an existing project.

FISHING GAME TOOL 2D

Configure Fishing Rod 2D

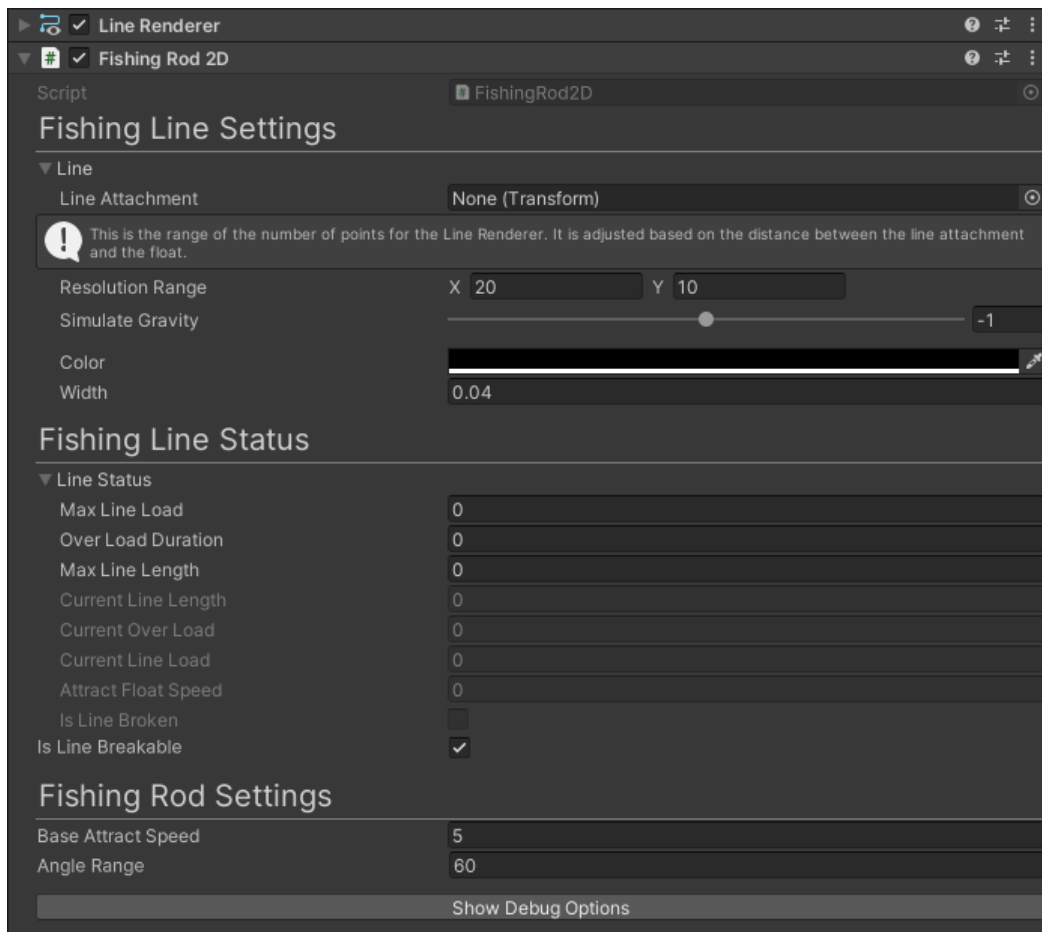
During this phase, our attention will be directed towards configuring the fishing rod.

1. Add Fishing Rod Sprite To The Scene.

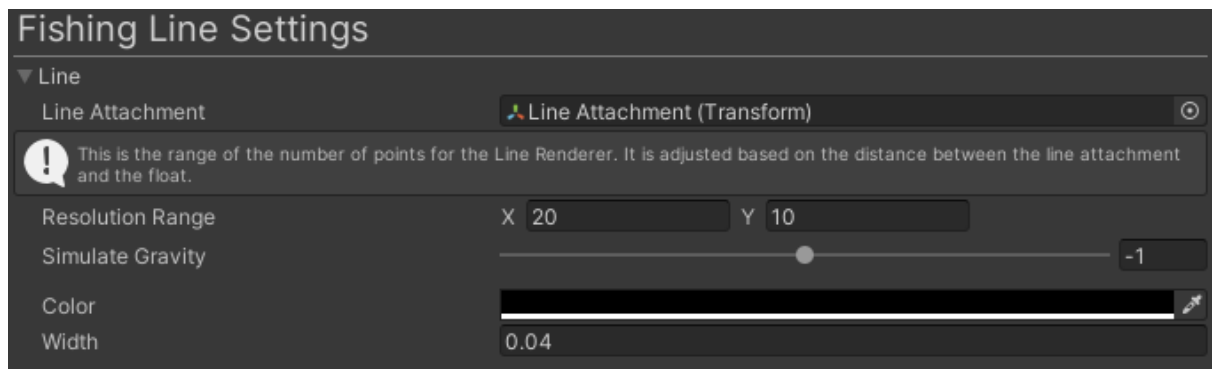
Add the previously prepared fishing rod sprite to the scene.

2. Configure Fishing Rod 2D.

To add the Fishing Rod 2D component, navigate to **Add Component -> Fishing Game Tool 2D -> Fishing Rod 2D**. Simultaneously, the **Line Renderer** component will also be appended. This component is employed for visualizing the fishing line.



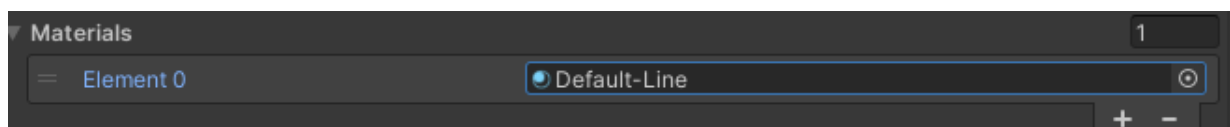
Let's initiate the configuration process from the **Fishing Line Settings** section. Under Line, you'll find elements dedicated to configuring the fishing line. Begin by designating an empty Game Object as the point of origin for the line. Subsequently, employ the created Game Object as the **Line Attachment** within the **Fishing Rod 2D** component.



Resolution Range defines the range of points in the line renderer based on the distance between the fishing float and the fishing rod. Value X determines the number of points close to the rod, and Value Y determines the number of points when the fishing float is distant. The point count changes smoothly based on the distance.

Simulate Gravity determines the force acting on the fishing line. The lower the value, the more the line sags. A value of 0 makes the line taut.

Color defines the color of the fishing line. To make it work, you need to add a material to the Line Renderer. **Width** determines the width of the fishing line.



Fishing Line Status stores information about the durability of the fishing line and its current state. **Max Line Load** determines the maximum load the line can withstand. This value should not

be less than 10. It significantly affects the fishing difficulty. **Over Load Duration** specifies the time after which the line will break if the Max Line Load is exceeded. **Max Line Length** parameter defines the maximum length of the fishing line. When a fish is caught, it will not swim beyond this specified maximum line length. The fishing float on ground will move in the direction of the fishing rod to prevent exceeding this distance. When casting the float, setting this value too low will result in the float stopping prematurely. **Is Line Breakable** indicates whether the line can be broken at all. Deselecting this option will not break the line, but after exceeding the catch limit, the fishing action will still conclude. However, the line won't need to be re-rigged.

Fishing Line Status

▼ Line Status

Max Line Load

0

Over Load Duration

0

Max Line Length

0

Current Line Length

0

Current Over Load

0

Current Line Load

0

Attract Float Speed

0

Is Line Broken

☐

Is Line Breakable

☒

Fishing Rod Settings is the final element of configuration. **Base Attract Speed** determines the basic speed of attracting the fishing float. Boosts are added to the base speed depending on the tension of the line, the angle between the rod and the float, or the loot tier. **Fishing Angle** is utilized in the calculation of line tension and lure retrieval speed. It determines the maximum angle between the rod and the float, which is taken into account during the calculation.

Fishing Rod Settings

Base Attract Speed

5

!

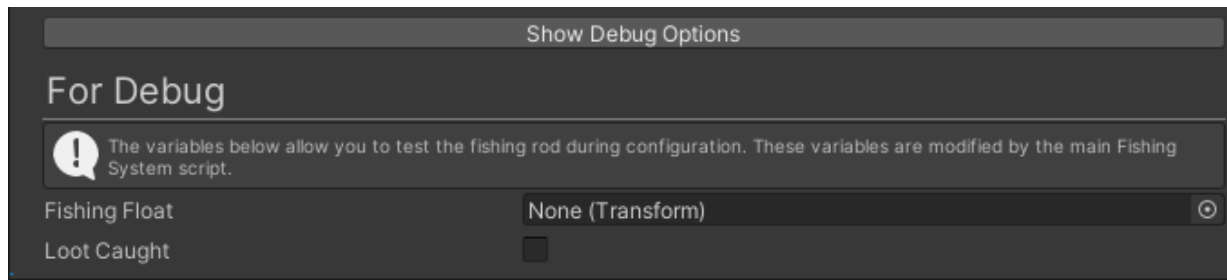
Fishing Angle is utilized in the calculation of line tension and lure retrieval speed. It determines the maximum angle between the rod and the float, which is taken into account during the calculation.

Fishing Angle

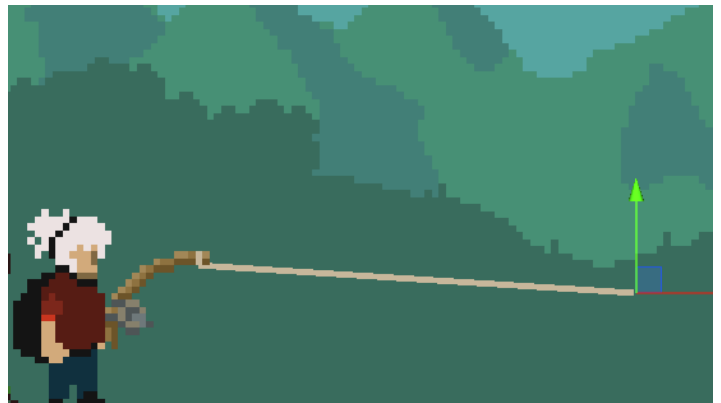
60

The configuration of the fishing rod is now complete. You can proceed to test it. By selecting **Show Debug Options**, you will

gain access to two variables enabling you to assess the rod's behavior independently of the main **Fishing System 2D** component.



For testing purposes, generate a game object on the stage and set it to represent a fishing float. Assign the created object to the **Fishing Float** field. You can now test the operation of the link. By checking **Loot Caught** you can force the tension of the line.



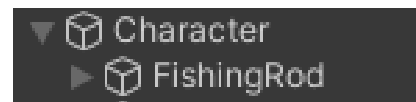
FISHING GAME TOOL 2D

Configure Fishing System 2D

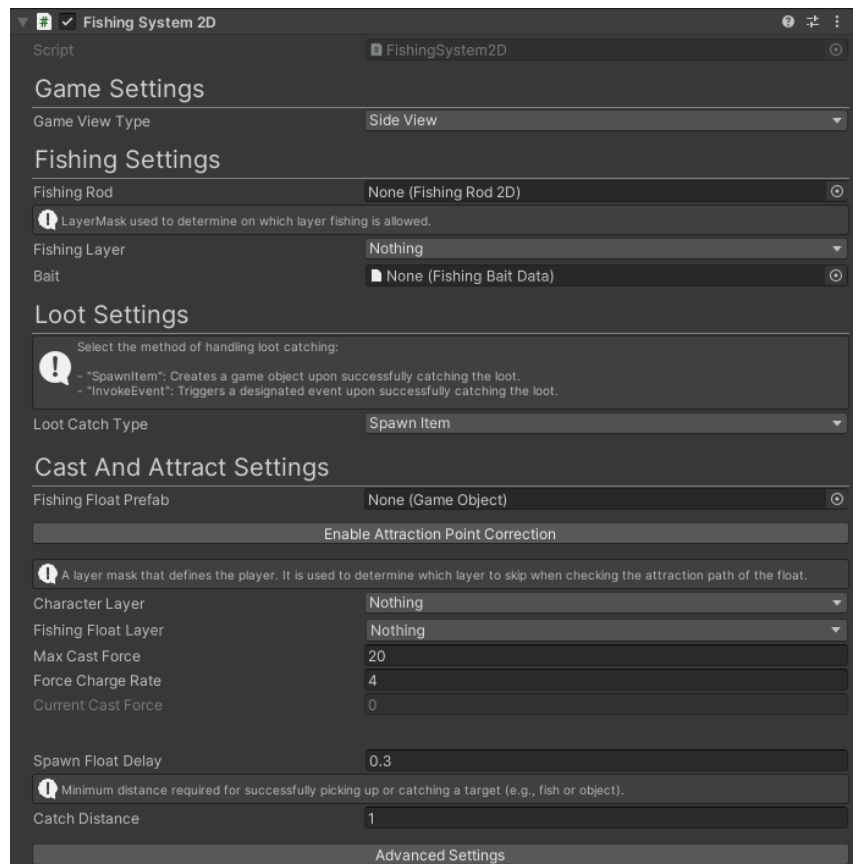
The Fishing System 2D assumes a crucial role as a component engineered to streamline fishing endeavors. Functioning as the core element of your fishing mechanics, this component orchestrates the intricacies of fish-catching activities, encompassing bait management, emulating fish interactions, and bestowing rewards upon successful catches.

1. Adding the Fishing System 2D Component.

Let's start by adding a Fishing System 2D component to our player. For the purpose of this example, I added a sprite representing the character and placed the previously created fishing rod in it.

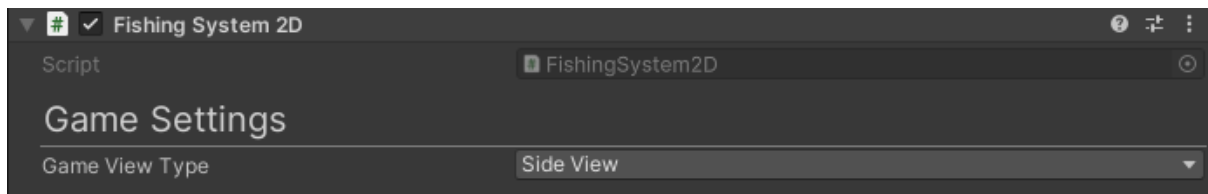


To add the **Fishing System 2D** component, go to **Add Component -> Fishing Game Tool 2D -> Fishing System 2D**.

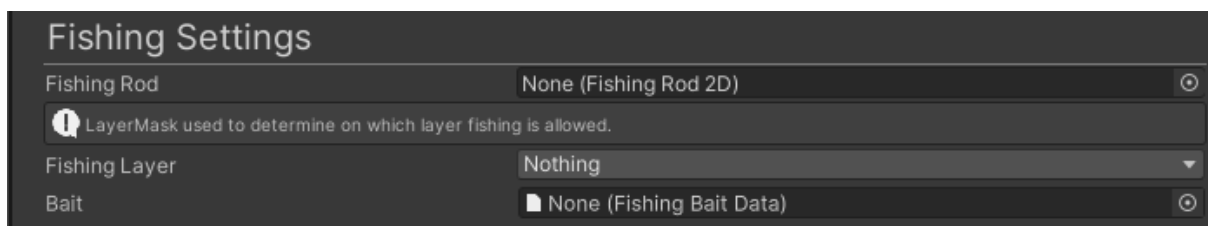


2. Configure Fishing System 2D.

Before starting the configuration, it's essential to choose the game view type. This is a crucial step that should be taken before further configuring the **Fishing System 2D**. The choice between **Side View** and **Top-Down View** affects the functionality of some systems.

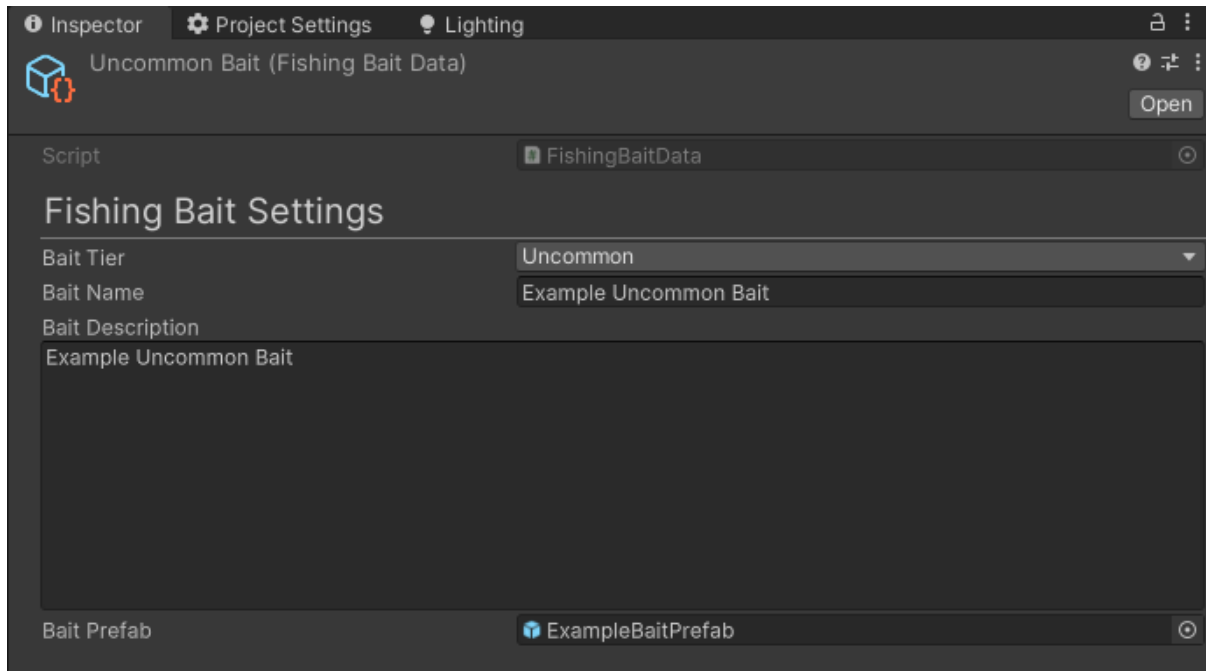


We'll initiate the configuration process from the **Fishing Settings** tab. The **Fishing Rod** parameter accepts values from the currently employed fishing rod. This setup accommodates the inclusion of multiple rods, each with unique parameters.



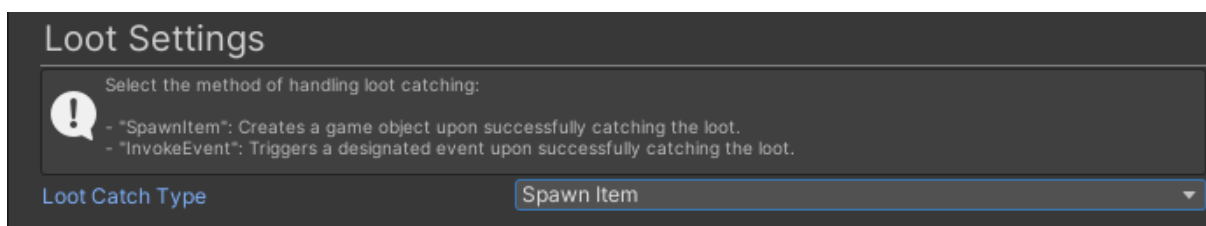
Fishing Layer is used to specify on which layer fishing is allowed. **Bait** parameter takes a scriptable object that specifies the type of bait added. It contains information about the bait's tier, name, description, and bait prefab. The prefab is used when the player wants to add bait after already having added some. The previous bait's prefab spawns in front of the player. The bait's tier determines its quality, which translates into a reduced waiting time for catching a loot on the hook. It also determines the tier of loot that can be caught. For example, using an Epic tier bait allows us to catch loot of a smaller or equal tier, up to Epic tier. Legendary tier loot are excluded.

To add a new bait, you need to click on **Create -> Fishing Game Tool 2D -> New Fishing Bait** in the **Project** window.



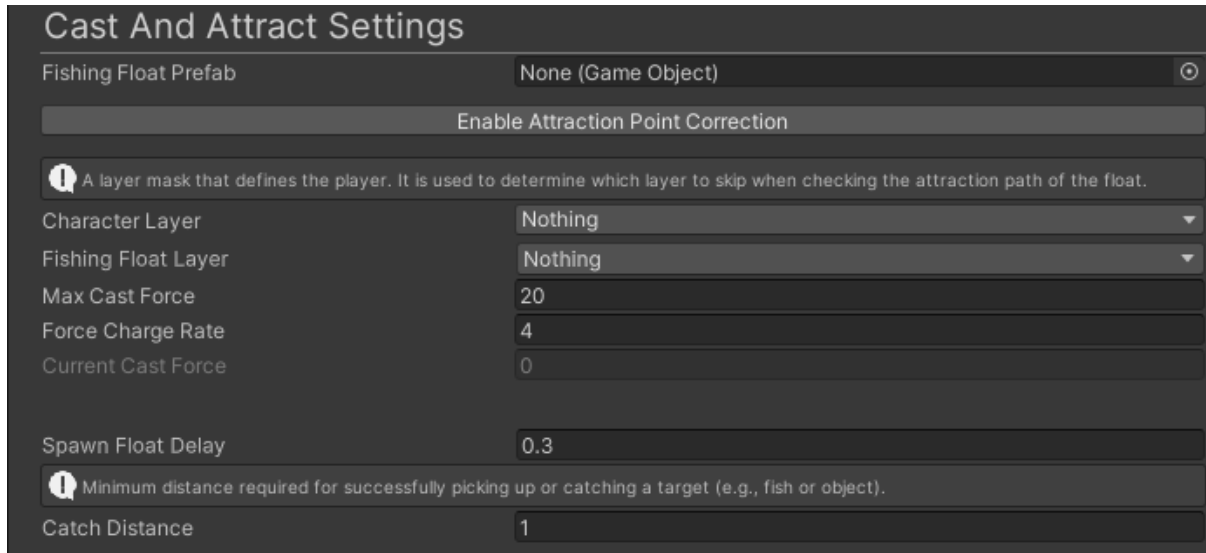
Loot Settings tab is used to select the method of handling the caught loot. **Spawn Item** throws the caught loot onto the shore. **Invoke Event** passes the loot prefab as a GameObject. This means you should add an event that takes a GameObject (public void AddItem(GameObject item)).

The Invoke Event function is in the development phase and is not functional in version 1.0 of the Fishing Game Tool 2D.



Cast and Attract Settings tab is used for configuring the fishing float, as well as its casting and attracting behavior. The options within it vary depending on the selected view.

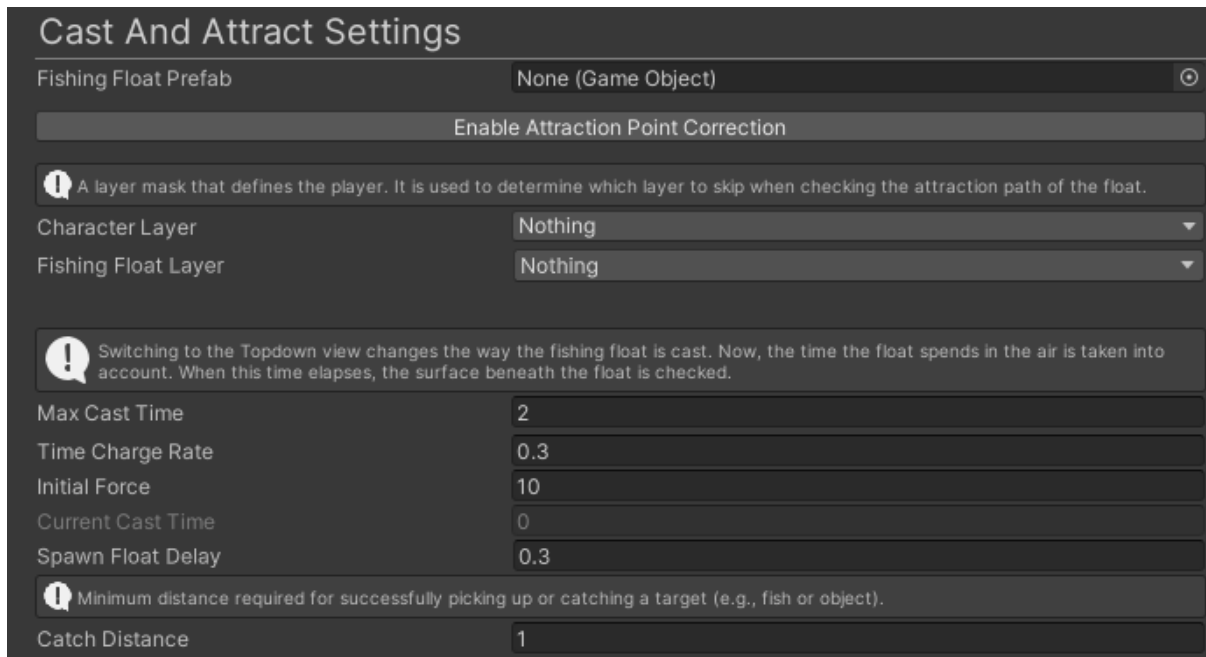
For the Side View:



The screenshot shows the 'Cast And Attract Settings' panel for the Side View. It features a dark grey background with white text. At the top, the title 'Cast And Attract Settings' is displayed. Below it, the 'Fishing Float Prefab' is set to 'None (Game Object)'. A toggle switch for 'Enable Attraction Point Correction' is turned on. A tooltip explains that a layer mask defines the player for attraction path checks. The 'Character Layer' and 'Fishing Float Layer' are both set to 'Nothing'. Numerical inputs for 'Max Cast Force' (20), 'Force Charge Rate' (4), and 'Current Cast Force' (0) are shown. 'Spawn Float Delay' is set to 0.3. Another tooltip explains the 'Catch Distance' (1) as the minimum distance for picking up a target.

Setting	Value
Fishing Float Prefab	None (Game Object)
Enable Attraction Point Correction	Enabled
Character Layer	Nothing
Fishing Float Layer	Nothing
Max Cast Force	20
Force Charge Rate	4
Current Cast Force	0
Spawn Float Delay	0.3
Catch Distance	1

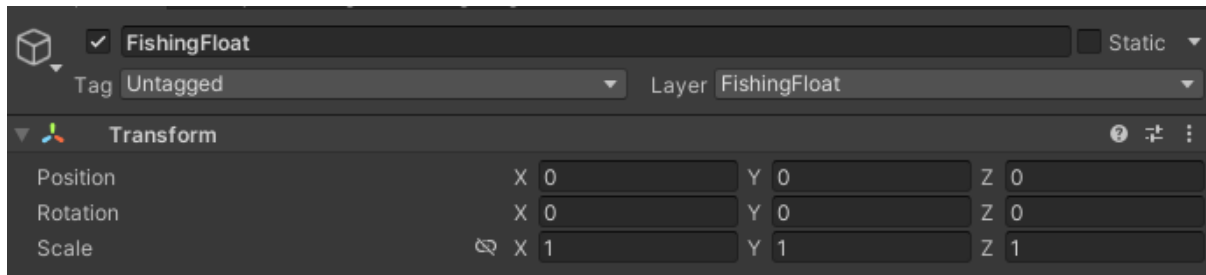
For the Topdown View:



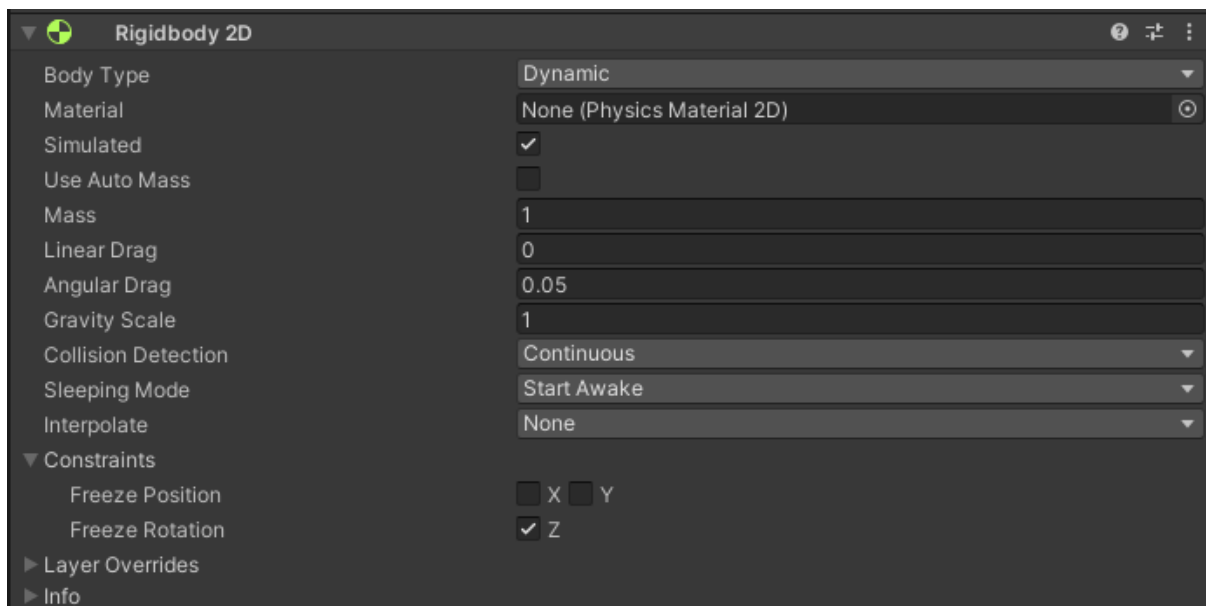
The screenshot shows the 'Cast And Attract Settings' panel for the Topdown View. It has the same layout as the Side View panel but with different settings. The 'Fishing Float Prefab' is 'None (Game Object)'. 'Enable Attraction Point Correction' is on. The same tooltip about layer masks is present. 'Character Layer' and 'Fishing Float Layer' are 'Nothing'. The 'Max Cast Time' is set to 2, 'Time Charge Rate' to 0.3, and 'Initial Force' to 10. 'Current Cast Time' is 0 and 'Spawn Float Delay' is 0.3. The 'Catch Distance' is 1, with a tooltip explaining it as the minimum distance for picking up a target.

Setting	Value
Fishing Float Prefab	None (Game Object)
Enable Attraction Point Correction	Enabled
Character Layer	Nothing
Fishing Float Layer	Nothing
Max Cast Time	2
Time Charge Rate	0.3
Initial Force	10
Current Cast Time	0
Spawn Float Delay	0.3
Catch Distance	1

Fishing Float Prefab stores the fishing float prefab. Let's move on to configuring the fishing float. As an example, I will use a sphere. Set its position to 0,0,0 and scale to 0.1,0.1,0.1. Create a Fishing Float LayerMask and assign it to the float.



Next, add a Rigidbody2D. Set Collision Detection to Continuous and lock rotation for Z axes.

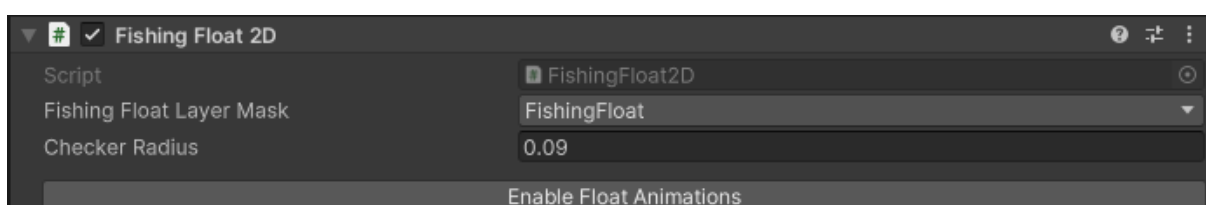
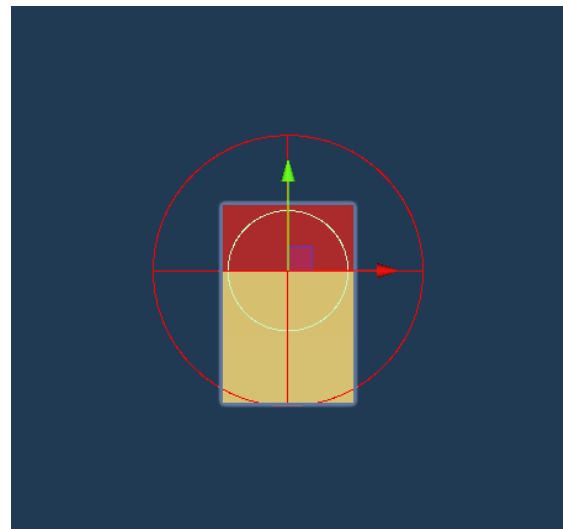


Now, add the **Fishing Float 2D** component. To do this, go to **Add Component -> Fishing Game Tool 2D -> Fishing Float 2D**.

Set the **Fishing Float Layer Mask** to be the same as that of the float.

Set the **Checker Radius** to extend beyond the outline of the fishing float. You can also

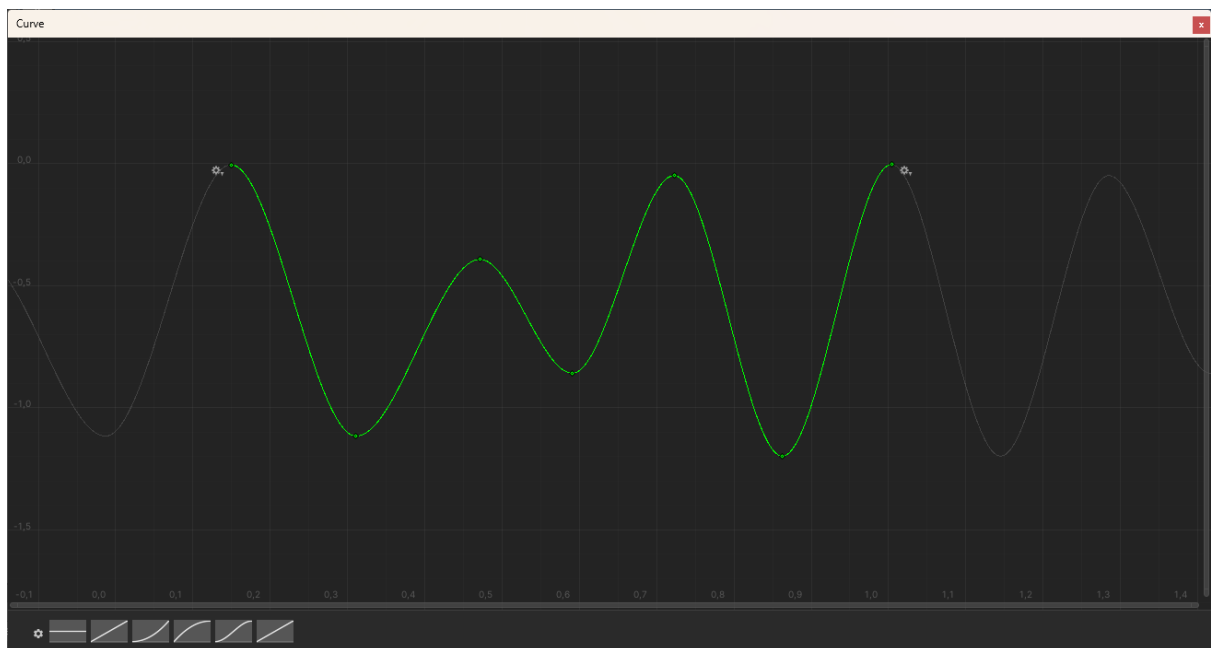
add animations to the float. The animation is based on the animation curve so you can easily modify it. To enable the animations press **Enable Float Animations**.



Float Representation is a sprite representing the float. It must be located in the main object of the float.

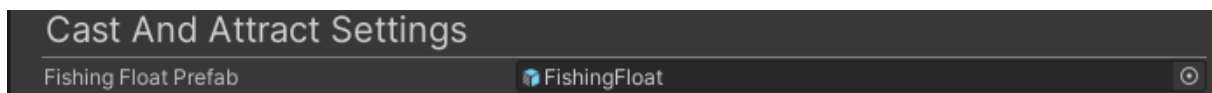


Float Animation Curve allows you to create your float animation. In my case, the curve takes values from 0 to -1. The right side and the left side have ping pong selected.

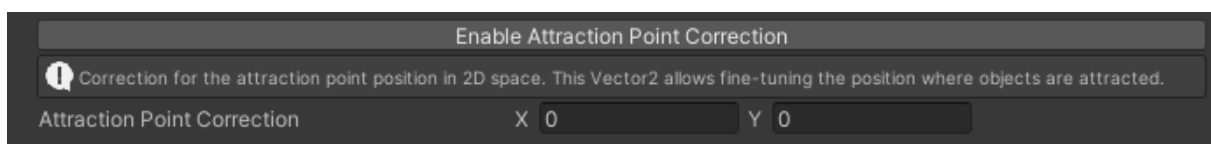


Anim Force determines with what force the animation curve should act on the float. **Anim Speed** determines its speed.

The prepared object should be placed in the Project tab as a prefab. Add the fishing float prefab to the **Fishing Float Prefab** field.



Enable Attraction Point Correction allows you to enable Attraction Point Position Correction. This option can be used when the character's pivot is in the middle of the sprite, for example.



Character Layer specifies the character mask. It is used to exclude it when checking for obstacles when attracting a float. **Fishing Float Layer** is used to specify the layer of the fishing float and exclude it during the path generation.

Character Layer	Character
Fishing Float Layer	FishingFloat

For the Side View:

Max Cast Force is a parameter that determines the maximum force for casting the fishing float. **Force Charge Rate** defines how quickly the current casting force will increase.

Spawn Float Delay determines the time after which the fishing float will be spawned. This allows you to adjust the timing of the float's release to match the casting animation.

Catch Distance defines the distance between the player and the loot at which the loot is successfully caught.

Cast And Attract Settings

Fishing Float Prefab FishingFloat

☐ Enable Attraction Point Correction

Correction for the attraction point position in 2D space. This Vector2 allows fine-tuning the position where objects are attracted.

Attraction Point Correction X Y

A layer mask that defines the player. It is used to determine which layer to skip when checking the attraction path of the float.

Character Layer	Character
Fishing Float Layer	FishingFloat
Max Cast Force	<input type="text" value="8"/>
Force Charge Rate	<input type="text" value="4"/>
Current Cast Force	<input type="text" value="0"/>

Spawn Float Delay

Minimum distance required for successfully picking up or catching a target (e.g., fish or object).

Catch Distance

For the Topdown View:

In the top-down view, the way the fishing float is cast changes. Now, to calculate the casting distance, the time the float spends in the air is utilized. After this time elapses, the collider of the float is enabled, and the surface on which it landed is checked.

Max Cast Time determines the maximum time the fishing float can spend in the air. **TimeChangeRate** determines how quickly the current casting time will increase.

InitialForce determines the basic force with which the fishing float is cast. It affects the casting speed.

Spawn Float Delay determines the time after which the fishing float will be spawned. This allows you to adjust the timing of the float's release to match the casting animation.

Catch Distance defines the distance between the player and the loot at which the loot is successfully caught.

Cast And Attract Settings

Fishing Float Prefab: Fishing Float

Enable Attraction Point Correction

! A layer mask that defines the player. It is used to determine which layer to skip when checking the attraction path of the float.

Character Layer: Character

Fishing Float Layer: FishingFloat

! Switching to the Topdown view changes the way the fishing float is cast. Now, the time the float spends in the air is taken into account. When this time elapses, the surface beneath the float is checked.

Max Cast Time: 2

Time Charge Rate: 1

Initial Force: 4

Current Cast Time: 0

Spawn Float Delay: 0.3

! Minimum distance required for successfully picking up or catching a target (e.g., fish or object).

Catch Distance: 2

The **Advanced Settings** button displays advanced options that can be edited.

Advanced Settings

▼ Advanced

Caught Loot Data: None (Fishing Loot Data)

! Custom Catch Probability Data.

Catch Probability Data: None (Catch Probability Data)

Caught Loot: ☐

Return Speed Without Loot: 2

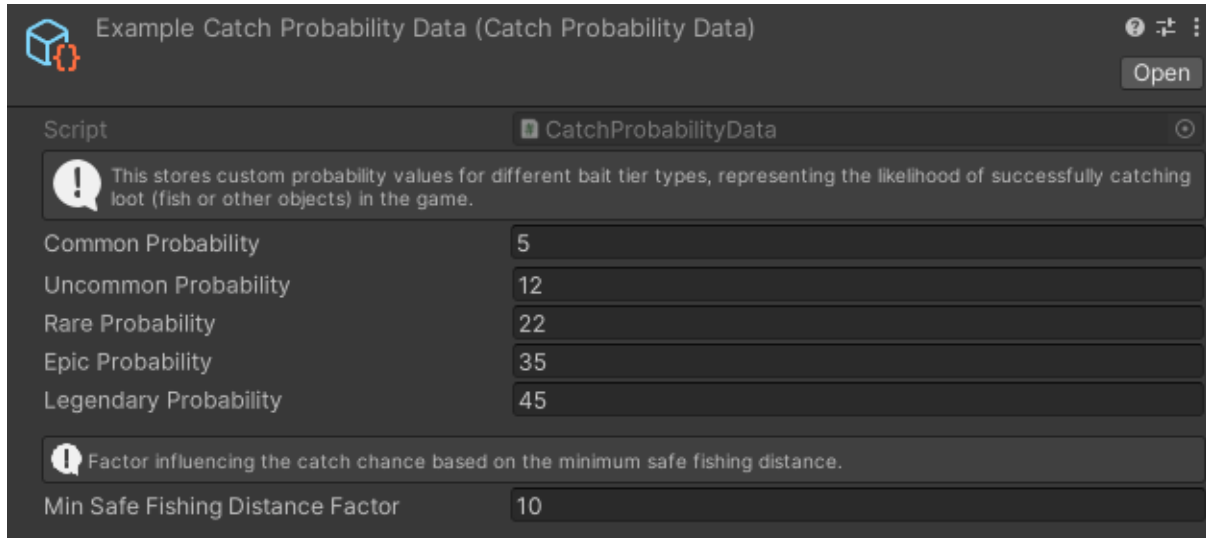
! The time interval in seconds between consecutive checks to determine if the catch has been successfully caught.

Catch Check Interval: 1

Loot Weight: 0

The **Catch Probability Data** allows you to modify pre-defined loot catching probability values with your own.

To add custom probability data, click on **Create -> Fishing Game Tool -> New Catch Probability Data** in the **Project window**. **Min Safe Fishing Distance Factor** determines the minimum distance that does not affect the likelihood of a catch. If the casted float is closer, the probability of catching decreases as the distance decreases.



Bait Tier	Probability
Common	5
Uncommon	12
Rare	22
Epic	35
Legendary	45

Min Safe Fishing Distance Factor: 10

Caught Loot allows you to force the capture of loot. It is used for testing purposes.

Return Speed Without Loot determines the speed at which the fishing float is reeled in when no loot has been caught on the hook.

Catch Check Interval determines how frequently the check is made to see if the loot has been caught on the hook. The values are in seconds.

]

Important!

To correctly determine the direction in which the fishing float will be cast, you need to send the direction from your character movement script to the **Fishing System 2D**.

It is necessary to use the preparation of the enum **CastDir** for this.

```
Odwwołania: 2
public void SetCastDirection(CastDir castDir)
{
    _castDir = castDir;
}
```

```
Odwwołania: 16
public enum CastDir
{
    left,
    right,
    top,
    down
};
```

A sample code that sends information about the direction in which the character is facing:

```
1 odwołanie
private void ControlCastDir()
{
    CastDir castDir = new CastDir();

    if (_saveLookDir == 0)
        castDir = CastDir.down;
    else if (_saveLookDir == 1)
        castDir = CastDir.top;
    else if (_saveLookDir == 2)
        castDir = CastDir.left;
    else if (_saveLookDir == 3)
        castDir = CastDir.right;

    _fishingSystem2D.SetCastDirection(castDir);
}
```

The fishing system has been successfully configured! Now let's move on to configure the rest of the components.

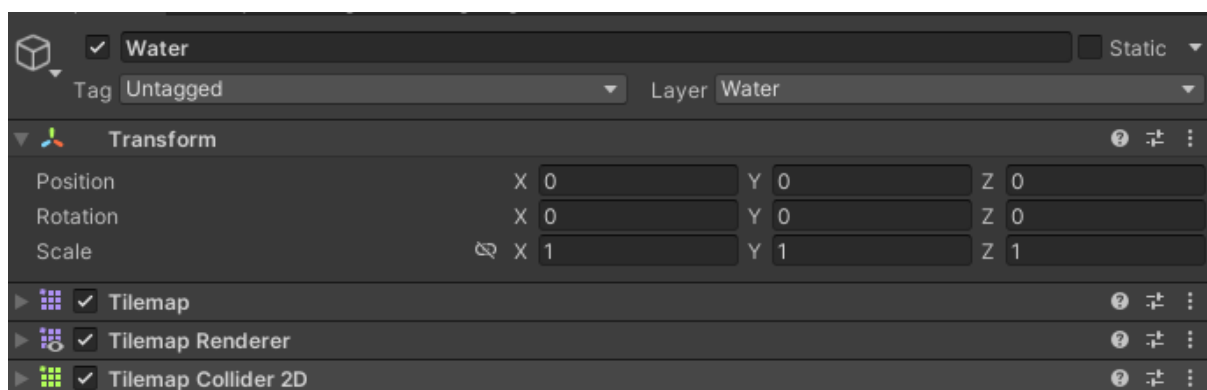
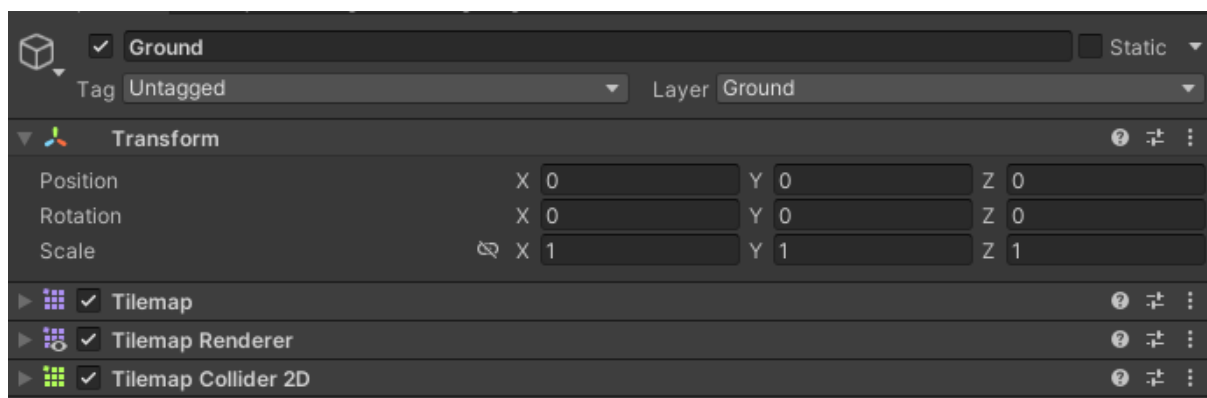
FISHING GAME TOOL 2D

Configure The Fishing Spot

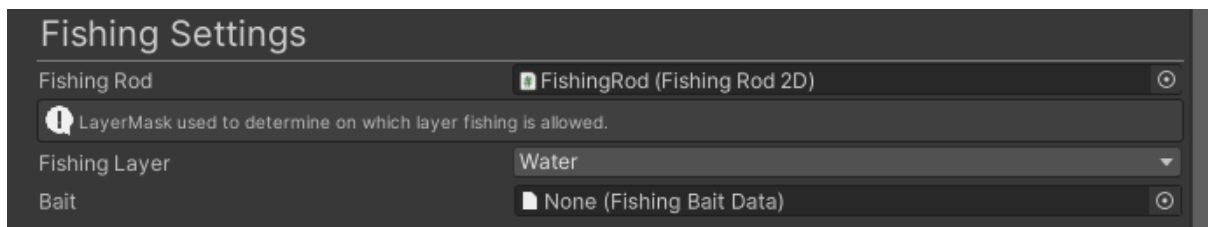
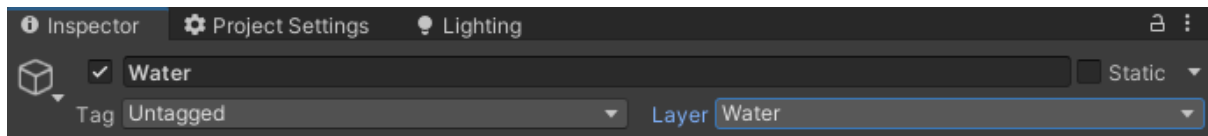
The **Fishing Game Tool 2D** facilitates the setup of multiple distinct fishing spots, each capable of featuring its own exclusive loot for acquisition. The shape or size of the fishing spot does not impose limitations on this capability.

1. Create a fishing surface.

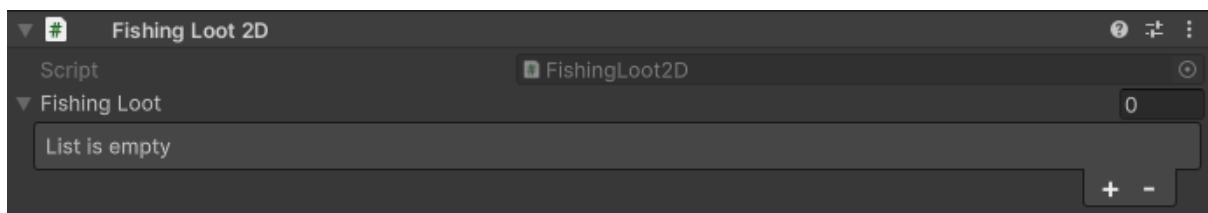
For the resource needs, I prepared a simple world based on tilemap. The land consists of a separate tilemap to which a 2d tilemap collider was added. Water is a separate tilemap to which I added a 2d tilemap collider.



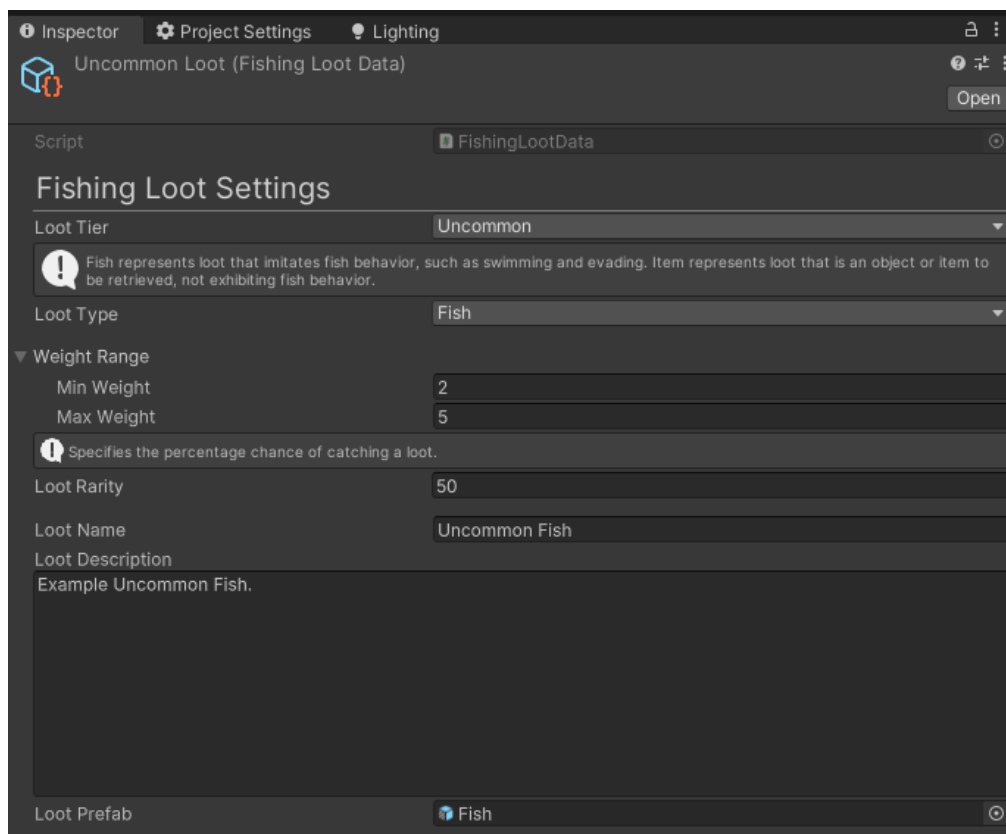
The next step will be setting the Layer for the created water. The Layer must be the same as the one set in the **Fishing Layer** in the **Fishing System 2D**.



Let's add **Fishing Loot 2D** to the water object. To do this, click on **Add Component -> Fishing Game Tool 2D -> Fishing Loot 2D**.



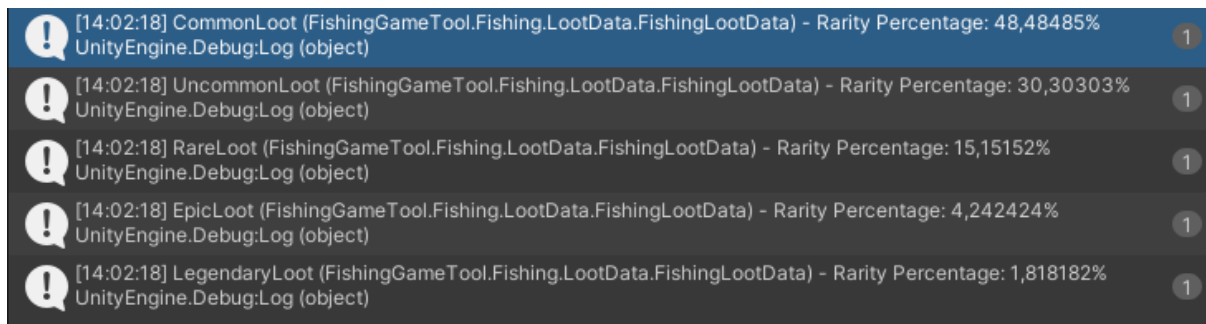
Fishing Loot 2D stores the available loot for a given fishing area. To create loot, click on **Create -> Fishing Game Tool 2D -> New Fishing Loot** in the **Project window**.



Loot Type determines the type of loot, which can be either Fish or Item.

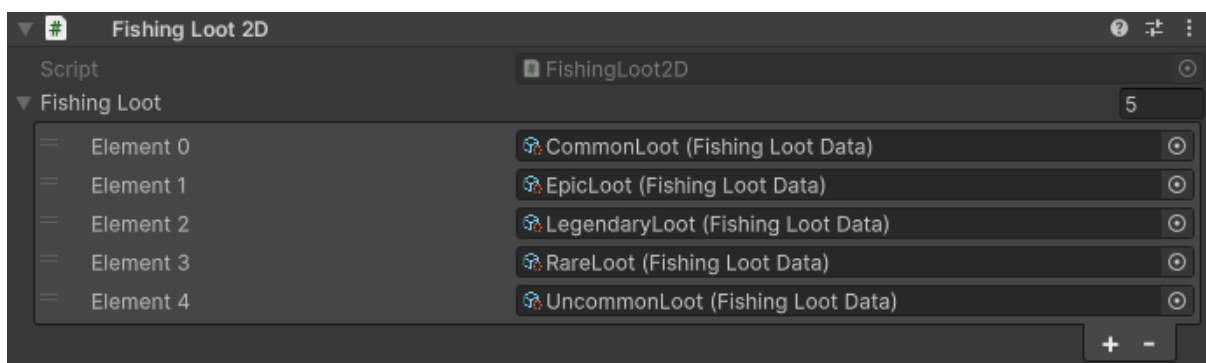
Weight Range stores the weight range for a specific loot. The final weight is randomly generated within the range of the minimum and maximum values of the variable.

Loot Rarity determines the rarity of the loot. The values of all available loot for a specific fishing spot are calculated and scaled to a range of 0-100. There's no need to calculate them manually.



Loot Prefab stores the loot prefab. This prefab will be spawned when the loot is caught.

For testing purposes, I created five different types of loot, ranging from common to legendary. Subsequently, I placed them in the **Fishing Loot 2D** component.



We have just configured the **Fishing Game Tool 2D**. The fishing system is now operational and can be tested. The right mouse button is responsible for reeling in the fishing float, while the left mouse button is used for casting it.

FISHING GAME TOOL 2D

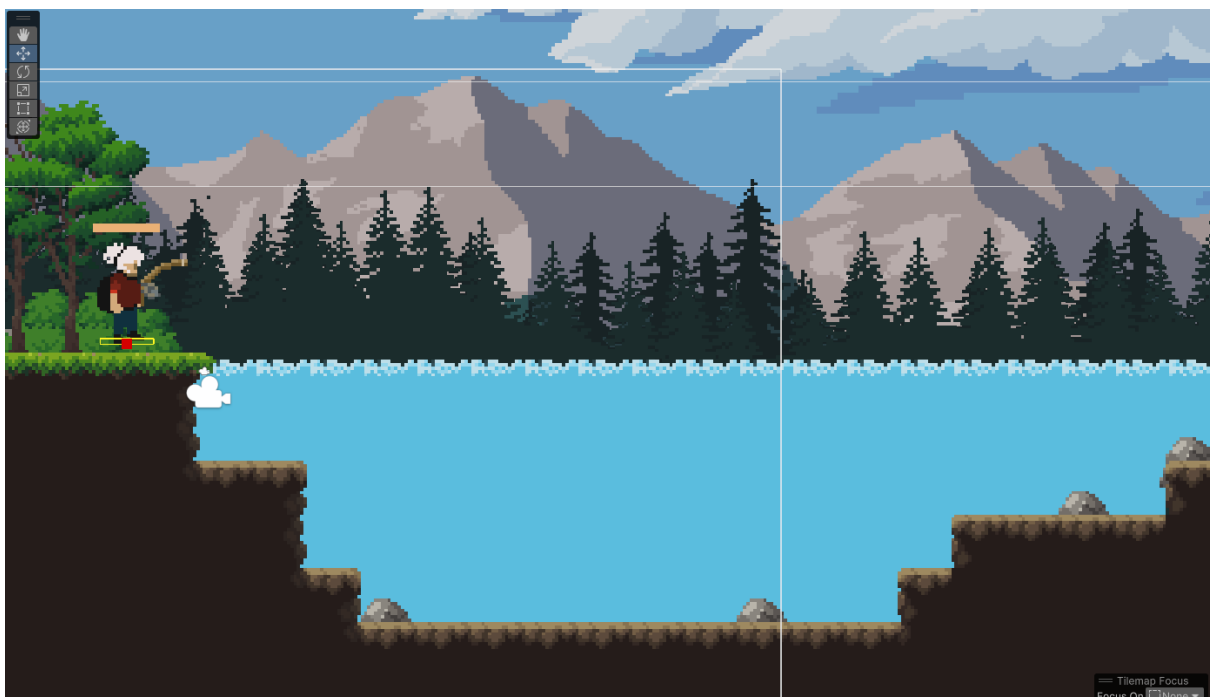
Testing

The configuration of the **Fishing Game Tool 2D** has been completed. You can now advance to the testing phase and evaluate the functionality of all subsystems. This testing process will help ensure that all components interact as expected and contribute to a seamless fishing experience within your project.

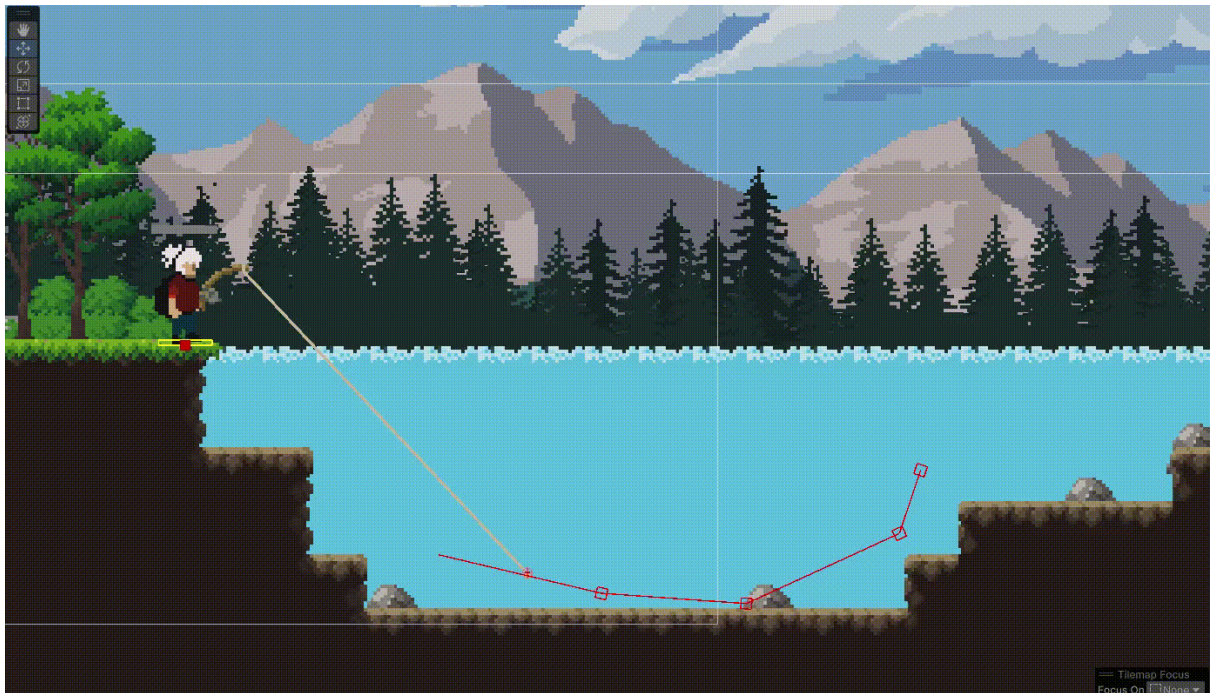
1. Simulation of fish behavior.

The Fishing Game Tool has been augmented with a fish behavior simulation feature. This advancement equips it with the ability to identify shorelines and other obstacles within the water, adapting its navigation to circumvent them. The tool adjusts its direction and speed, infusing the simulation with a heightened sense of realism. This enhancement ensures a more immersive fishing experience within your project.

I prepared a simple water spot. Let's see how the system works.



Press Play!

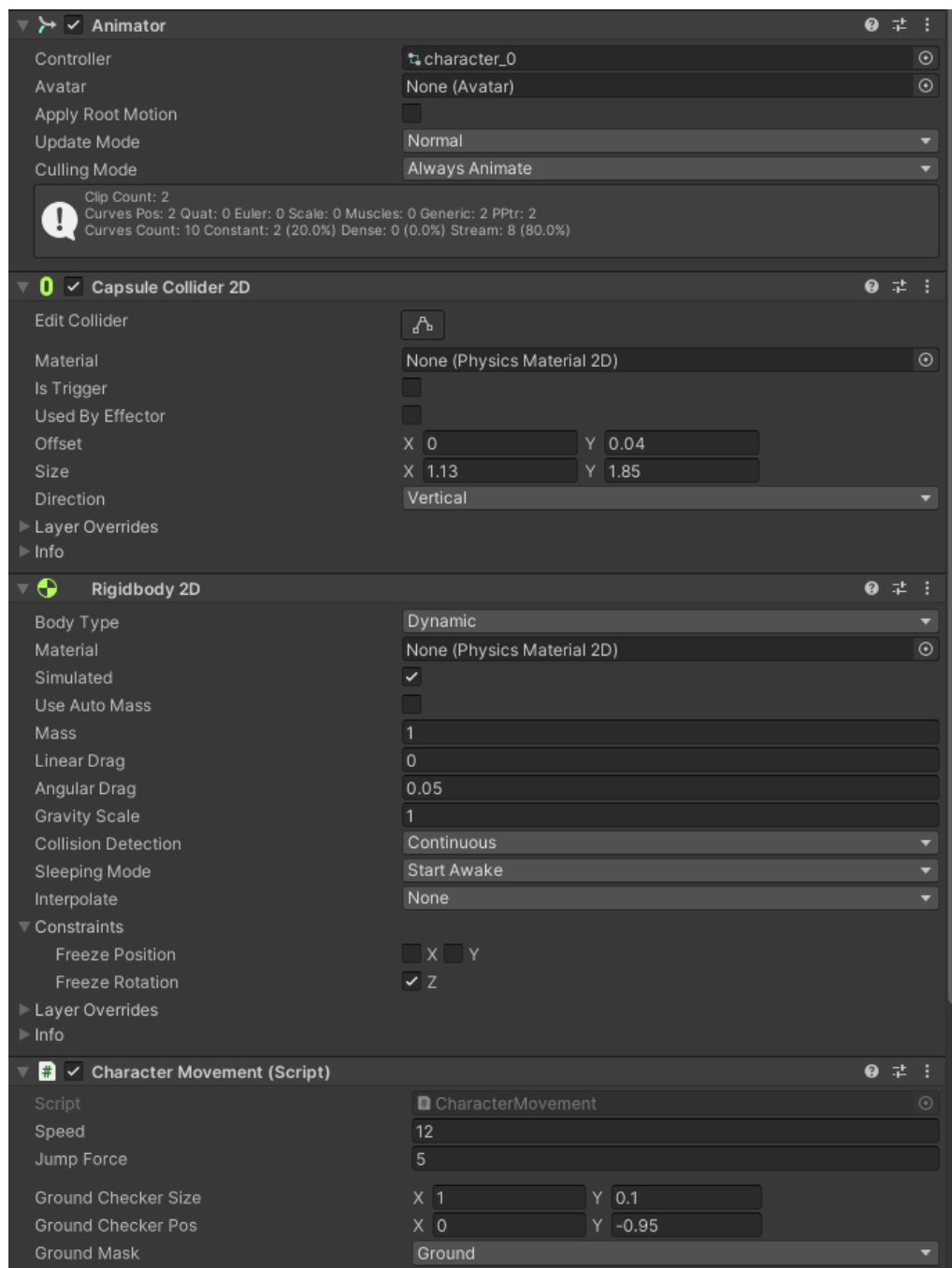


(The gif file in the documentation folder.)

FISHING GAME TOOL 2D

Character Movement

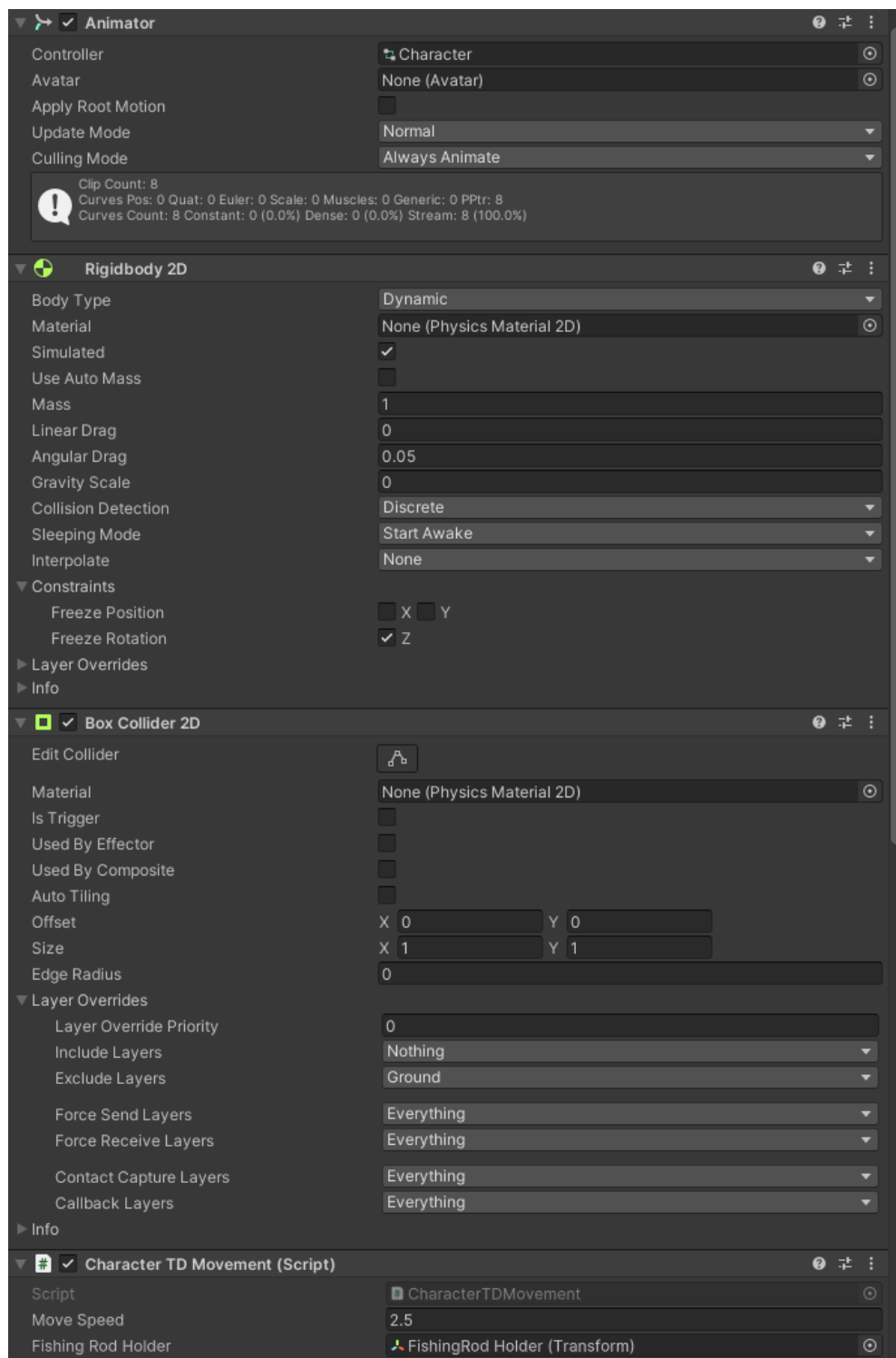
Character Movement is a straightforward script designed to incorporate the capability for character mobility within the project.



FISHING GAME TOOL 2D

Character TD Movement

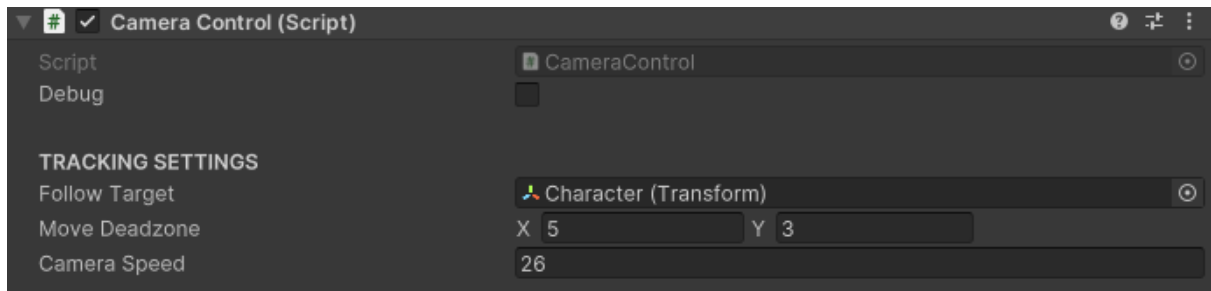
Character TD Movement is a straightforward script designed to incorporate the capability for character mobility within the project.



FISHING GAME TOOL 2D

Camera Control

Camera Control adds camera support to 2D sided view or topdown view game.



FISHING GAME TOOL 2D

Simple UI Manager

Simple UI Manager demonstrates how to construct a user interface (UI) for the established system. It offers the ability to customize colors and fill direction for a progress bar, as well as the option to activate a gradient color effect that corresponds to the progress of the bar. This script exemplifies the process of creating an engaging and interactive UI element within the system.

