



Table of Contents

- 1. Introduction**
- 2. Quick Start / Installation Guide**
- 3. Core Features**
- 4. Customization Options**
 - Fishing Settings
 - Skill Level Settings
 - Inventory Settings
 - Equipment Settings
 - Fish, Loot Items, and Fishing Poles
 - Water and Fish Populations
- 5. Integration / Advanced Usage**
 - Inventory System
 - Equipment System
 - Save / Load System
- 6. Quick Reference Guide**
- 7. Support & Contact**



Introduction

The Fishing System by Unreal Toolkit is a fully modular and highly customizable fishing system designed for smooth integration into any Unreal Engine project. It comes with a basic inventory and equipment system that can be easily swapped for your own custom systems, ensuring smooth implementation into existing gameplay mechanics.

Fishing mechanics are fully customizable, with optional Catch or Release UI, optional skill level progression, customizable settings for fishing duration and reaction time duration, custom fish populations with drop weights and rarity, and more.

The system provides immediate functionality and prototyping capabilities with 10 example fish, 4 example trash/treasure items, and 3 rods with unique stats. The included assets allow you to start using the system right away, while the data tables are designed for easy expansion, making it simple to add custom fish, items, and rods.

With intuitive settings and plug-and-play functionality, the Fishing System provides a complete foundation for fishing mechanics while remaining flexible for customization and expansion. Integration is simple and efficient.



Quick Start Guide

Follow these steps to quickly set up and test the Fishing System in your project:

1 Migrate the Fishing System into Your Project

- In the **Content Browser**, locate the **Fishing** folder included in the asset pack.
- Right-click the folder and select **Migrate**.
- In the pop-up window, ensure all necessary assets are selected (**Uncheck the Demo Folder**), then click **OK**.
- Choose your project's **Content** folder as the migration destination.

2 Add Fishing Functionality to Your Player Character

- Open your **Player Character Blueprint**.
- In the Components panel, click **Add Component** and search for **AC_Fishing**.
- Select **AC_Fishing** to add it to your character.

3 Place a fishable water body in the World

- In the **Content Browser**, locate **BP_Water**.
- Drag **BP_Water** into the level.
- In the **Details Panel**, select the “**Fish Population**” component, and set the fish population for the water by selecting the fish or item from the **DT_FishingItems** data table and setting a drop weight.

4 Place a Fishing Pole in the World

By default the fishing system will equip a standard fishing pole at runtime, but if you disable that you can add a fishing pole to be picked up from the world with the following steps.

- Locate **BP_FishingItem** in the **Content Browser**.
- Drag **BP_FishingItem** into the level.
- In the **Details Panel**, set the **ItemID** to a fishing pole (e.g., “Standard Fishing Pole”).

5 Test the Fishing System

- Press **Play** to start the game.
- If you disabled the option to equip a fishing pole at runtime, pick up the rod from the world and equip the **Fishing Pole** from the inventory by right-clicking it in the inventory and selecting equip.
- Approach the **Water** and **click while facing it to cast**.
- Wait until you see the “Fish Biting” notification, and then click to catch the fish.

Congratulations! The Fishing System is now active in your project. From here, you can customize settings, expand the data tables, and refine the mechanics to fit your game.



Core Features

- ◆ Fully Modular & Data-Driven System
 - Uses data tables for defining Fish data, Fishing Poles, loot drops, skill levels, and fishing mechanics.
 - Easily expandable—add new fish, rods, and loot with minimal effort.
- ◆ Customizable Fishing Mechanics
 - Settings options for various aspects of the fishing mechanics
 - Toggles to use or ignore skill levels
 - Set the duration for fishing and reaction time
- ◆ Skill & Progression System (Optional)
 - Experience gain determined by the fish caught.
 - Skill levels for fishing, which can be enabled or disabled in settings.
 - Fishing poles can have fishing level requirements and bonus yield to encourage progression.
- ◆ Dynamic Loot System
 - Customizable loot drops with rarity-based drop chances.
 - Rarity chance and fishing duration can be adjusted by the fishing pole stats.
 - Customize fish populations and drop weights
 - Use included population volumes to add local populations to certain areas within larger populations, allowing for special fish populations within larger bodies of water.
- ◆ Pre-Configured & Customizable Assets
 - 10 example fish, set up with default data for rarity, value, size, weight, etc.
 - 3 fishing poles, each with adjustable skill level requirements, fishing duration multipliers, and rarity chance multipliers.
 - 4 additional loot items, including driftwood, seaweed, a diamond ring, and a rusty can.
 - Easy to expand with custom assets
- ◆ Visual & Audio Effects for Immersion
 - Niagara FX for fish biting (splashing in the water) and level up effects.
 - Customizable sound effects and animations for casting and reeling, fish biting, etc.
 - All UI widgets are customizable right from the settings panel - making it easy to adjust the aesthetic to fit your project.
- ◆ Easy Integration with Existing Systems
 - Comes with a basic inventory system and equipment that can be easily swapped out for your own.
 - There are clearly marked integration points, and integration helper functions to hook the system up to your own custom game systems and existing logic.



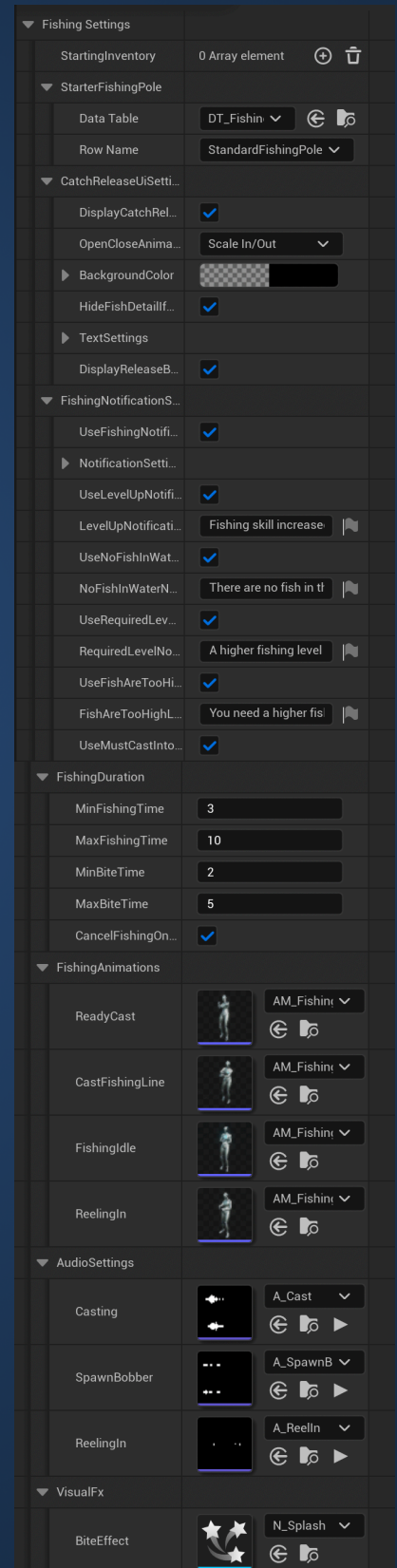
Customization Options

The Fishing System offers extensive customization options, allowing developers to tailor the mechanics, visuals, and overall experience to fit their game. Below is a breakdown of the key settings available in AC_Fishing and DT_FishingItems.

1. Fishing Settings (Fishing Component Settings)

The AC_Fishing component, added to the player, provides settings to customize the fishing system.

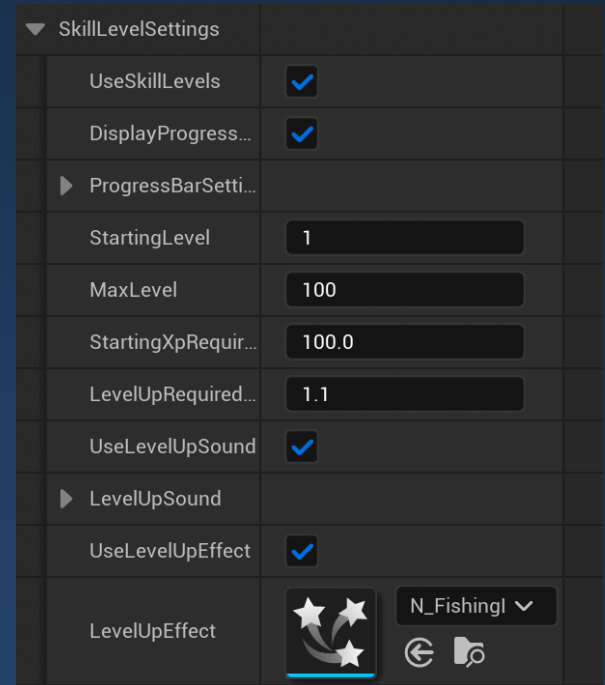
- Set starting inventory and starter fishing pole to determine the items the player should start with, and if they should start with a fishing pole equipped.
- Toggle the use of the skill system, and customize the progress bar appearance and location.
 - Set the starting fishing skill level, amount of XP needed to level up, and multiplier for increasing XP needed as the levels progress.
 - Set level up visual and audio effects
- Toggle the use of the default equipment system
 - Define the bone for the main hand equipment socket, and the local location and rotation for the socket, making it easy to use the equipment system with custom character meshes.
 - Optionally determine socket adjustments for fishing idle and reeling in animations, to accommodate different fishing animations.
- Toggle the use of the Catch or Release widget, giving players the option to keep or release the fish they caught.
 - Customize the appearance of the widget
- Customize system notifications
 - Set duration and adjust notification aesthetics
 - Customize the notification text for the different notification scenarios, and toggle notifications on or off for specific scenarios.
- Set the duration for fishing, and the amount of time players have to react to a bite before the fish gets away.
- Sound & Animation Settings:
 - Optionally customize the fishing animations for ready cast, cast, fishing idle, and reeling in.
 - Optionally customize the audio for casting, spawning the bobber on the water, and reeling in.
- Customize the visual effects for when a fish is biting and when the player levels up the fishing skill.





2. Skill Level Settings (AC_Fishing)

- Toggle whether or not to use the skill level system
- Toggle the use of the progress bar
 - Set progress bar settings to determine location, bar color, text settings, icon, etc.
 - Set the starting level for the fishing skill, as well as the maximum level
 - Set the XP required to level up
 - The experience system works with a multiplier to increase the amount of XP needed to level up with each level gained. Customize the multiplier to adjust how much the amount of XP needed for the next level changes.
 - Toggle the use of a level up sound effect, and select custom sound effects from a drop down menu
 - Toggle the use of the level up visual effect, and select custom visual effects from a drop down menu of niagara effects



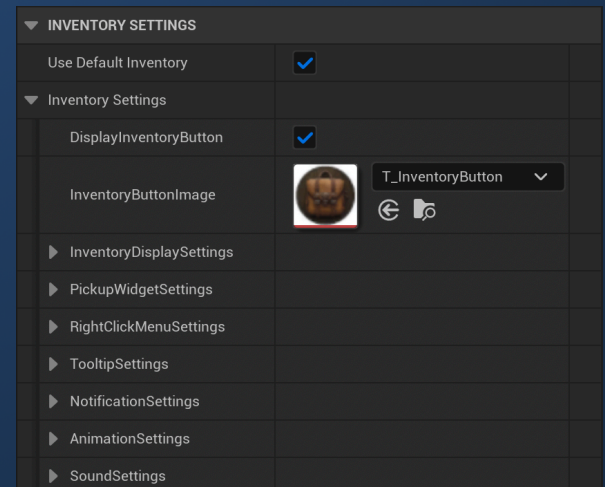
The Progress Bar widget **WBP_FishingSkillBar** updates itself and can be placed anywhere. You can place it in any custom widget and it will update itself.



3. Inventory Settings (AC_Fishing)

The fishing system uses a default inventory system that can be swapped out for your own custom inventory system if desired. The default inventory system is fully customizable.

- Toggle whether or not to use the default inventory system
- Toggle the use of the Inventory Button widget and select an icon for the inventory button.
- Notifications:
 - Customize Pickup Notification Text (e.g., "+1 [ItemName]").
 - Adjust Notification Duration.
- Full customization settings for inventory widget aesthetics
 - Pickup item from world widget
 - Inventory widget
 - right-click menu
- Set inventory notification settings, determine notification text, duration, and aesthetic
- Pickup & Drop Settings:

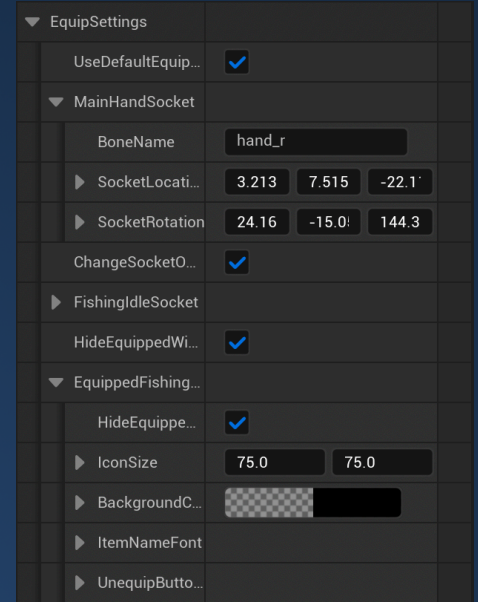




- Enable/disable Pickup Animation when collecting items, and set custom animations from a drop down menu.
- Enable/disable Inventory Sound Effects for adding/removing items, and set custom sound effects from a drop down menu.

4. Equipment Settings (AC_Fishing)

- Toggle whether or not to use the default equipment system
- Set the main hand socket data for your custom character mesh - this is where the fishing pole will visibly appear on your character.
 - Set the bone name
 - Set the local location and rotation for the socket to fine tune the positioning of the equipped fishing pole in your character's hand.
- Toggle whether or not the fishing pole should change sockets, or socket location/rotation during the fishing idle and reeling in animations. This helps to allow the fishing pole to remain positioned correctly while the character mesh uses its main hand to do the reeling motion.
 - Set the bone name, location and rotation for the fishing idle / reeling in socket data
- Toggle whether or not the equipped fishing pole widget should hide itself if no fishing pole is equipped.
 - Fully customize the aesthetics of the widget

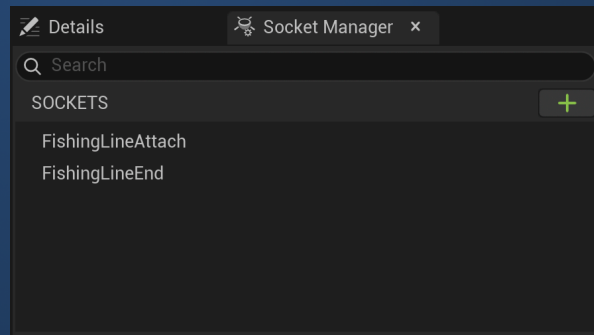


The equipped fishing pole widget **WBP_EquippedFishingPole** updates itself and can be placed anywhere. You can place it in any custom widget and it will update itself and remain functional.

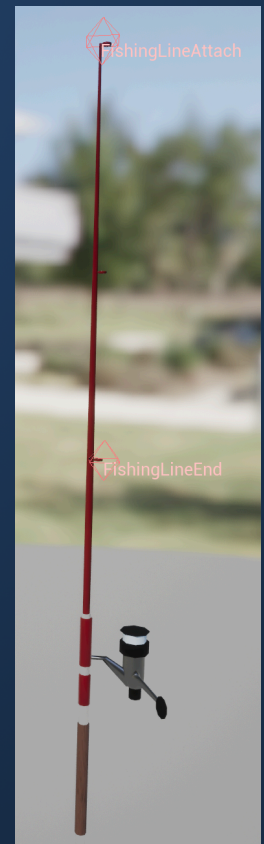


Preparing Custom Fishing Pole Meshes

The fishing system attaches a cable component to the fishing pole to add a dynamic fishing line. To determine where the fishing line should attach to your custom fishing pole mesh you must add 2 sockets to your fishing pole mesh.



- Open your fishing pole mesh
- In the details panel, click the Socket Manager tab and click + to add a socket
 - The name of the sockets **must match exactly** to the socket names shown
 - **FishingLineAttach**
 - **FishingLineEnd**
- Position the sockets where you want them on your mesh. The very tip of the rod, and a loop where you want to "hook the line", as shown in the example.





5. Fish, Loot Items, and Fishing Poles (DT_FishingItem)

The DT_FishingItem data table controls fish and item data, as well as fishing poles and their stats. To add items, **open DT_FishingItems and click +ADD** to add a new row. Fill in the item data to add your item to the system.

Base Item Data (DT_FishingItem)

- ItemID – A reference to the data table row.
- Name - The item's name
- Description - A description of the item. Will appear in item tooltips and other UI.
- Icon - The icon/image for the item
- Static Mesh - The item's mesh
- Weight - The physical weight of the item
- Value - The item's monetary value
- Quantity - The quantity that 1 instance represents
- CanStack? - Toggles if the item can stack
- MaximumStackSize - Sets the maximum stack size
- CanPickupFromWorld - Enables/disables the ability to pick up the item from the world
- CanDestroy - Enables/disables the ability to destroy the item from the inventory
- CanRemoveFromInventory - Enables/disables the ability to remove the item from the inventory by dropping or destroying it.
- PhysicsEnabled? - Toggles physics for the item when it's in the world.

Fish Data (DT_FishingItem)

- Size - The size of the fish/item
- Rarity - The rarity level for the fish/item
- RequiredSkillLevelToCatch - Sets the minimum skill level needed to catch the fish/item
- ExperienceReward - The amount of experience rewarded when the fish/item is caught

Fishing Pole Data (DT_FishingItem)

- IsFishingPole - Toggles whether or not the item is a fishing pole
- RequiredSkillLevelToEquip - Sets the minimum skill level required to equip the fishing pole.
- DurationMultiplier - Affects the duration it takes to get a bite.
- CommonChanceMultiplier - Affects the chances to catch a common fish.
- UncommonChanceMultiplier - Affects the chances to catch an uncommon fish.
- RareChanceMultiplier - Affects the chances to catch a rare fish.

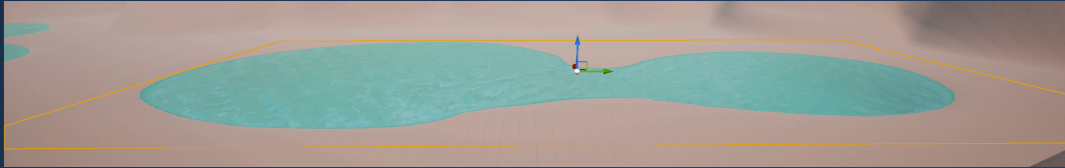
New Row	
BaseItemData	
ItemID	
Data Table	DT_FishingItems
Row Name	None
Name	
Description	
Category	(select)
Icon	None
StaticMesh	None
Weight	0.0
Value	0
Quantity	1
CanStack?	<input checked="" type="checkbox"/>
MaximumStackSize	99
CanPickupFromWorld	<input checked="" type="checkbox"/>
CanDestroy?	<input checked="" type="checkbox"/>
CanRemoveFromInventory	<input checked="" type="checkbox"/>
PhysicsEnabled?	<input checked="" type="checkbox"/>

FishData	
Size	0.0
Rarity	Common
RequiredSkillLevelToCatch	0
ExperienceReward	0.0
FishingPoleData	
IsFishingPole	<input type="checkbox"/>
RequiredSkillLevelToEquip	0
DurationMultiplier	1.0
CommonChanceMultiplier	1.0
UncommonChanceMultiplier	1.0
RareChanceMultiplier	1.0



6. Water and Fish Populations (BP_Water, AC_FishPopulation, BP_PopulationVolume)

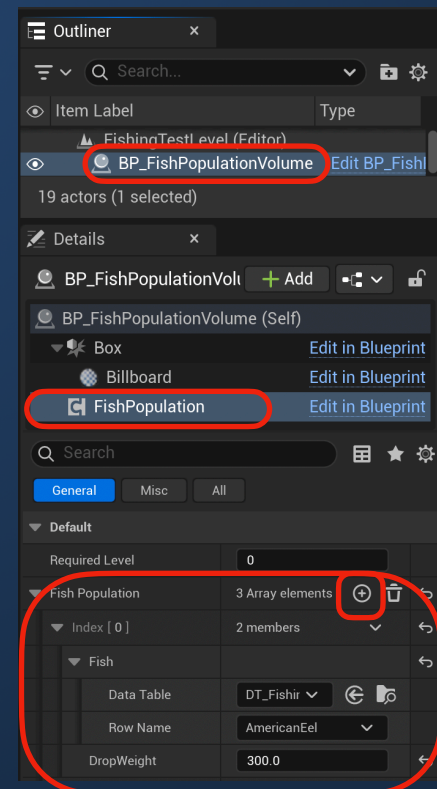
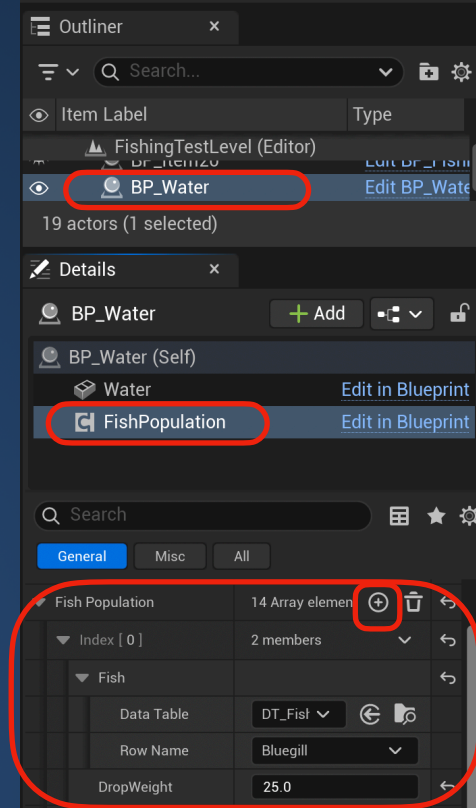
BP_Water is the default water blueprint for the fishing system. It uses **AC_FishPopulation** to make it into fishable water and add a fish population to it. You can use your own water blueprint by simply adding **AC_FishPopulation** to your water blueprint.



- Once **BP_Water** (or your custom water blueprint with **AC_FishPopulation**) is added to the world, select it and in the details panel select **AC_FishPopulation**
 - Here you can click + to add a new fish into the population by selecting the fish **ItemID**, and setting a drop weight.
 - A higher value entered into the drop weight will increase the chances that fish is caught, a lower drop weight makes it rarer to catch.

BP_PopulationVolume uses a collision box to add a fish population to specific areas within larger bodies of water. This allows for rare fishing zones within a larger body of water. For example, it could be used for setting different populations in the deeper parts of the water and along the coast, or give special populations to special areas like rivers that run through caves, or lakes that branch out into smaller lakes, etc.

- BP_PopulationVolume** adds the designated population to the overall water population, so the player can still catch any fish that are in the overall water body population.
 - If a fish is listed in the water body's population, and in a population volume's population then the drop weights are added together increasing the chance that the fish will be caught.
- Drag **BP_PopulationVolume** to the world and resize it as needed to create the area you'd like to add fish to.
- Just like you did with **BP_Water**, select the volume and in the details panel select **AC_FishPopulation**
- Click + to add a new fish into the population by selecting the fish **ItemID**, and setting a drop weight.





Integration / Advanced Usage

The Fishing System is designed to be modular and easy to integrate with other inventory and/or equipment systems. If you are using a custom inventory system, you can override the built-in functions to ensure seamless integration.

Integrating a Custom Inventory System

By default, the fishing system includes AC_DefaultInventory, a basic inventory setup designed for easy testing and quick implementation. However, if you are using your own inventory system, you will need to override a few key functions to replace the default item management logic with your own.

Overriding The Inventory

In AC_Fishing there are the following inventory related functions that you can easily replace with your own Inventory functions. Open each of the functions below in AC_Fishing, and replace the logic with your inventory functions. If you are integrating with your own custom inventory, ensure to uncheck "UseDefaultInventory" found in the Inventory Settings in AC_Fishing. (From your player character blueprint, select AC_Fishing, and find the settings in the details panel.)

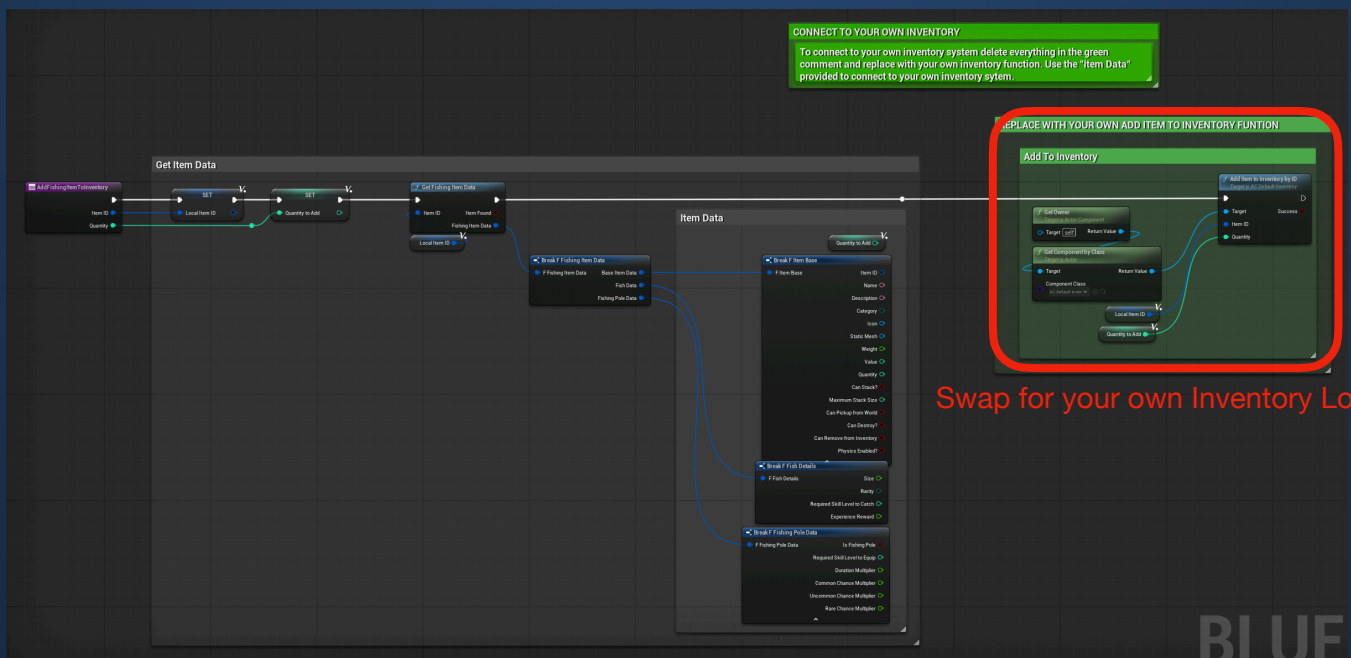


The Inventory functions are set up to expose the item data for easy integration. Replace the green commented section with your own inventory function. Use the exposed Item Data to hook into your inventory function.

Do this for the two inventory functions:

AddFishingItemToInventory

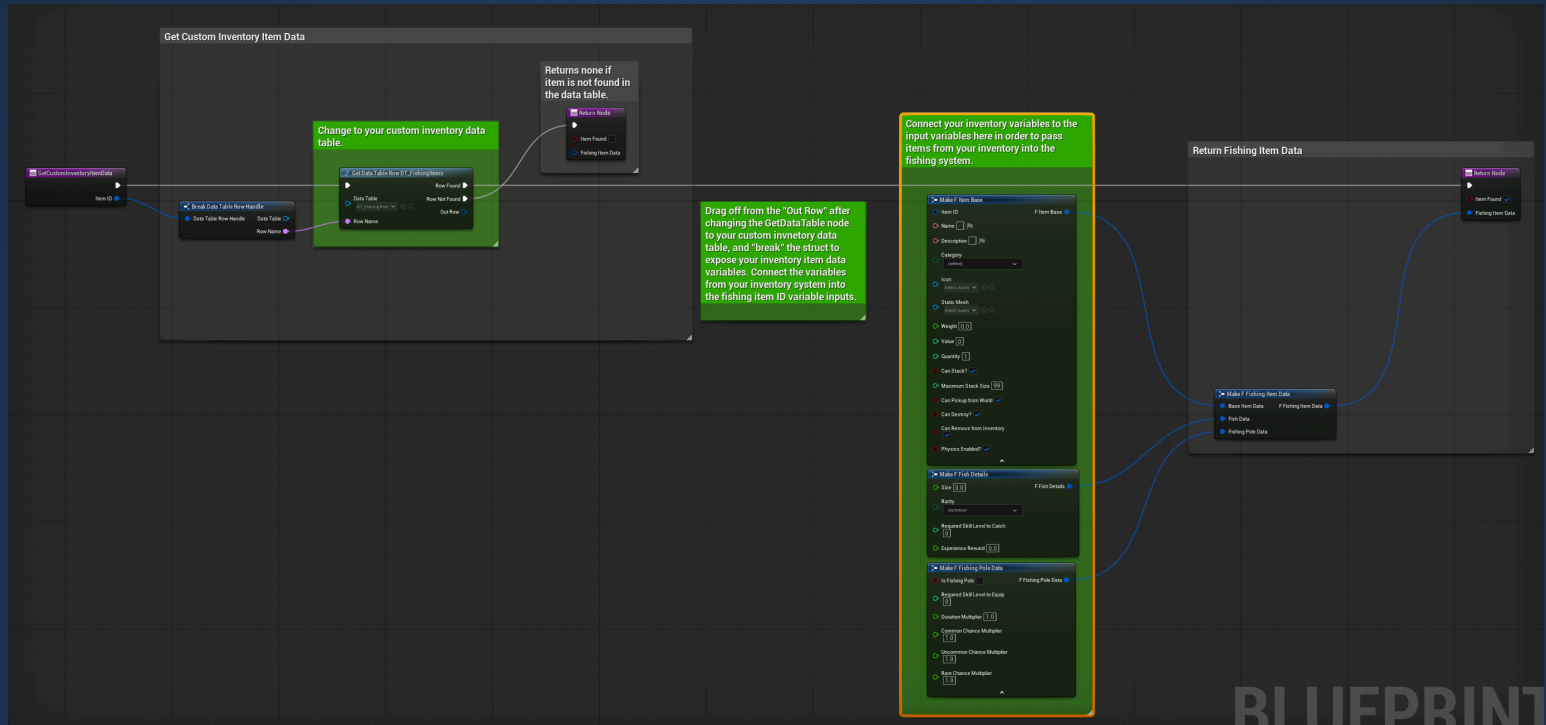
RemoveFishingItemFromInventory





To make the fishing system able to read item data from your custom inventory, you must edit the **GetCustomInventoryItemData** function found in **BPFL_CustomInventoryIntegration**.

- Open **BPFL_CustomInventoryIntegration**
- Select the function **GetCustomInventoryItemData**



- Change the Get data table row node to your custom inventory data table.
- Drag off from the out row and “break” to expose your custom item data.
- Connect your custom item data variables to the corresponding variables in the “MakeFishingItemData” section of the function.

This allows you to set your own custom items as loot items in fish populations. After doing this you will still be able to use DT_FishingItems to add new fish/items/fishing poles to the system.

After editing those **3 inventory functions** the fishing system will work with your inventory by passing item data from DT_FishingItem into your custom inventory system, and item data from your custom inventory system into the fishing system.

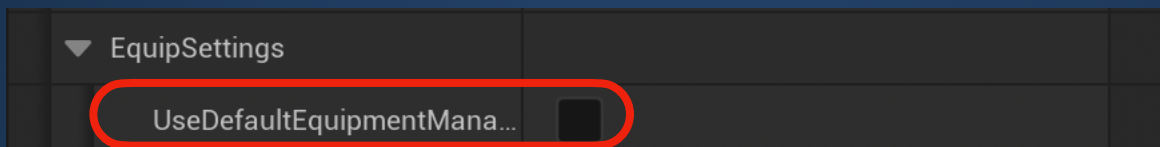


Integrating a Custom Equipment System

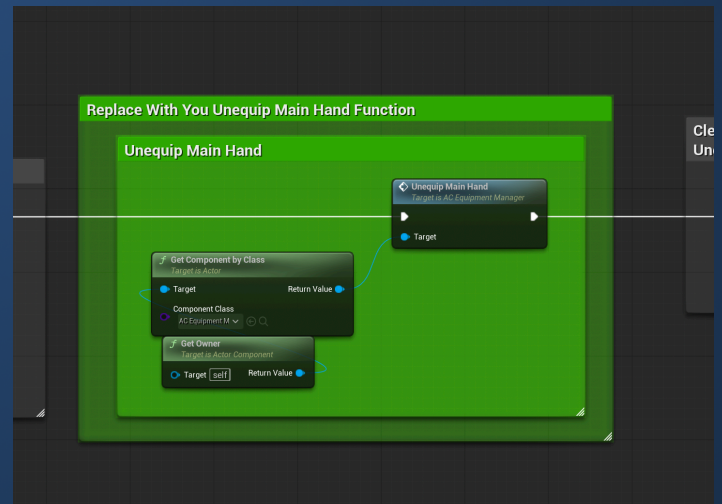
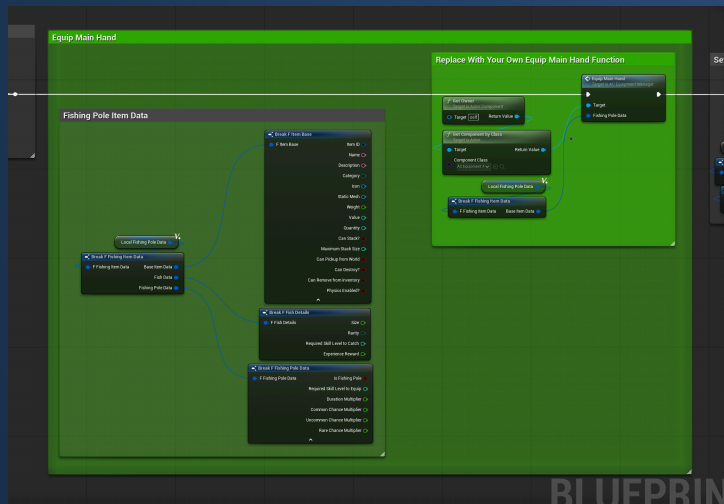
By default, the fishing system includes AC_DefaultEquipmentManager, a basic equipment system designed for easy testing and quick implementation. However, if you are using your own equipment system, you will need to override a few key functions.

Overriding The Equipment System

In AC_Fishing there are the following equipment related functions that you can easily replace with your own equip functions. Open each of the functions below in AC_Fishing, and replace the logic with your equip functions. If you are integrating with your own custom equipment system, ensure to uncheck “UseDefaultEquipmentManager” found in the Equipment Settings in AC_Fishing. (From your player character blueprint, select AC_Fishing, and find the settings in the details panel.)



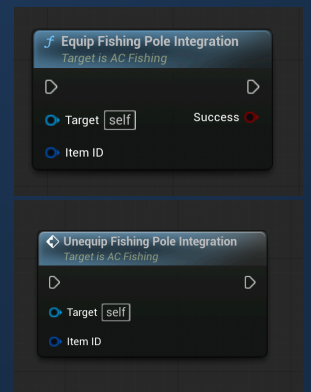
The EquipFishingPole and UnequipFishingPole functions are set up for you to edit. Open AC_Fishing, and edit only the green commented areas of the EquipFishingPole and UnequipFishingPole functions. The fishing system handles the logic for marking “FishingPoleEquipped”, you’re overriding the part of the function that visibly attaches the item mesh to your player, and marking something like “MainHandEquipped” in your custom inventory system. This will allow the fishing system to communicate with your custom equipment system whenever the fishing system calls its equip/unequip functions.



In your own custom equipment system, add the “EquipFishingPole_Integration” function into your equip function. This function will check if the item is a fishing pole, and if so it will mark “FishingPoleEquipped” in the fishing system. You can add this at the end of your custom equip function.

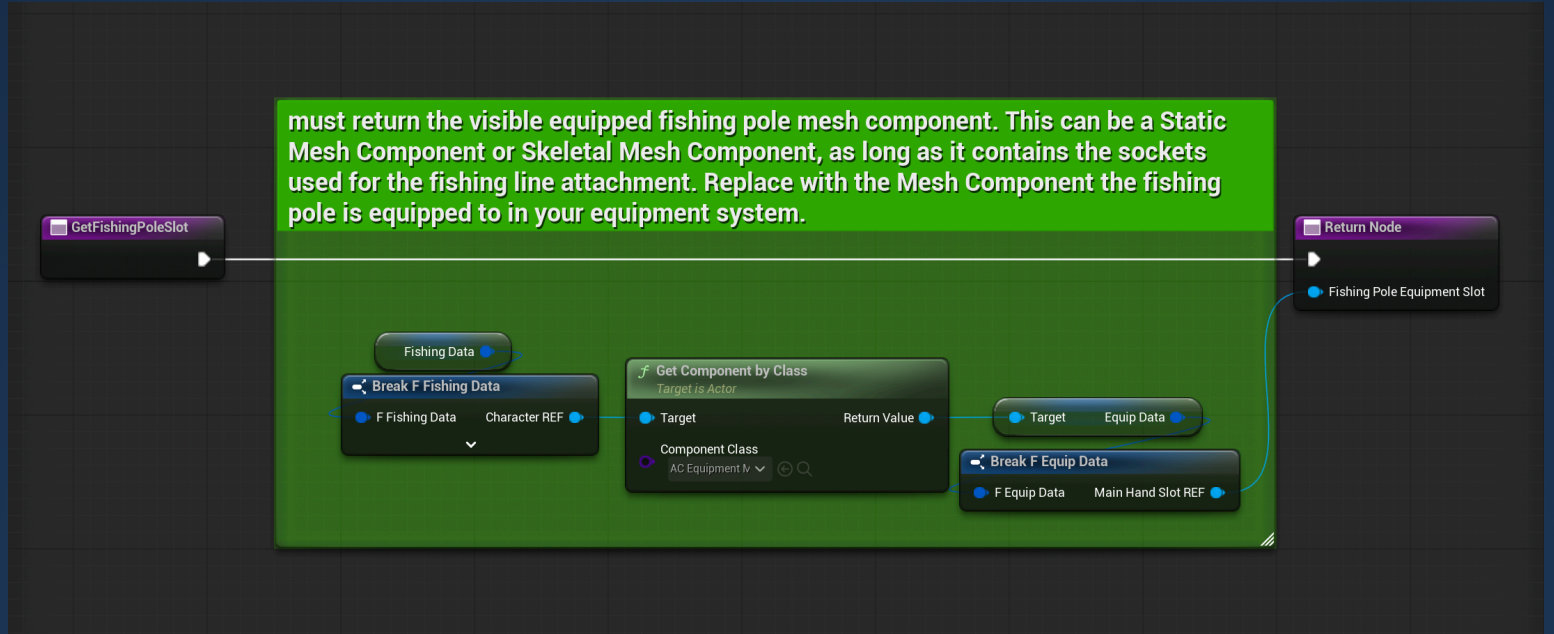
In your unequip function, add the “UnequipFishingPole_Integration”. This function will check if the item being unequipped is a fishing pole, and if so it will uncheck “FishingPoleEquipped” in the fishing system.

For both of these functions, feed in the ItemID (Data table row) of the item being equipped or unequipped.





The fishing system sometimes needs a reference to the fishing pole slot, or main hand slot of the player character. To do so it uses an overridable function “GetFishingPoleSlot”, that you can edit to return a reference to the slot being used from your custom equipment system. This reference can be either a static mesh component or a skeletal mesh component.



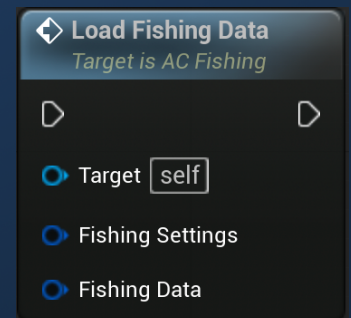
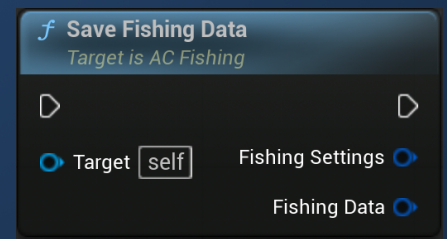
Editing these equipment functions, and adding the integration functions to your custom equipment system will allow you to integrate the fishing system with your own equipment system. Allowing you to use your own equipment system to manage equipped/unequipped states separately from the fishing system, while still allowing the fishing system to communicate with the equipment system.

Integrating Save/Load Functions With Your Save System

The fishing system uses two variables to store all data. FishingSettings, and FishingData. AC_Fishing includes a SaveFishingData and LoadFishingData function to make it easy to set up with your save system.

In your Save Game Blueprint, in your save function, call the SaveFishingData function, and promote the two outputs (FishingSettings, FishingData) to variables within your save game blueprint. Make sure that in the save game function, you are setting these variables from the output of the SaveFishingData function. This will store the data in your save game file.

In your Load Game function, after accessing the save game file, pull off of it and drag the FishingData and FishingSettings variables into the LoadFishingData function. This will ensure that the fishing data and settings are loaded from your saved data.





Quick Reference Guide

Actor Components	
AC_Fishing	Added to your player character blueprint to give the player fishing functionality.
AC_FishPopulation	Can be added to custom water blueprints to make them into fishable water and assign fish populations to them.
Blueprints	
BP_Water	The default water blueprint for the fishing system. Added to the world to add fishable bodies of water, and set fish populations for them.
BP_PopulationVolume	Used to add local fish populations to specific areas of larger bodies of water.
BP_FishingItem	The blueprint that can be added to the world to add items from DT_FishingItems into the game world, allowing the player to pick them up from the world.
Data Table	
DT_FishingItems	The data table that defines the fish, loot items, and fishing poles for the fishing system. Add rows to add custom items.
Structs	
F_FishingSettings	The struct that contains all the fishing system settings options. Editable in the details panel when you select AC_Fishing in your player character blueprint.
Function Libraries	
BPFL_CustomInventoryIntegration	A function library containing a function that allows you to make the fishing system understand how to read item data from your custom inventory system.



Support & Contact

For any questions, issues, or feedback, please contact us at:

support@unreal-toolkit.com

Explore Unreal Toolkit's asset pack library on: Unreal-Toolkit.com