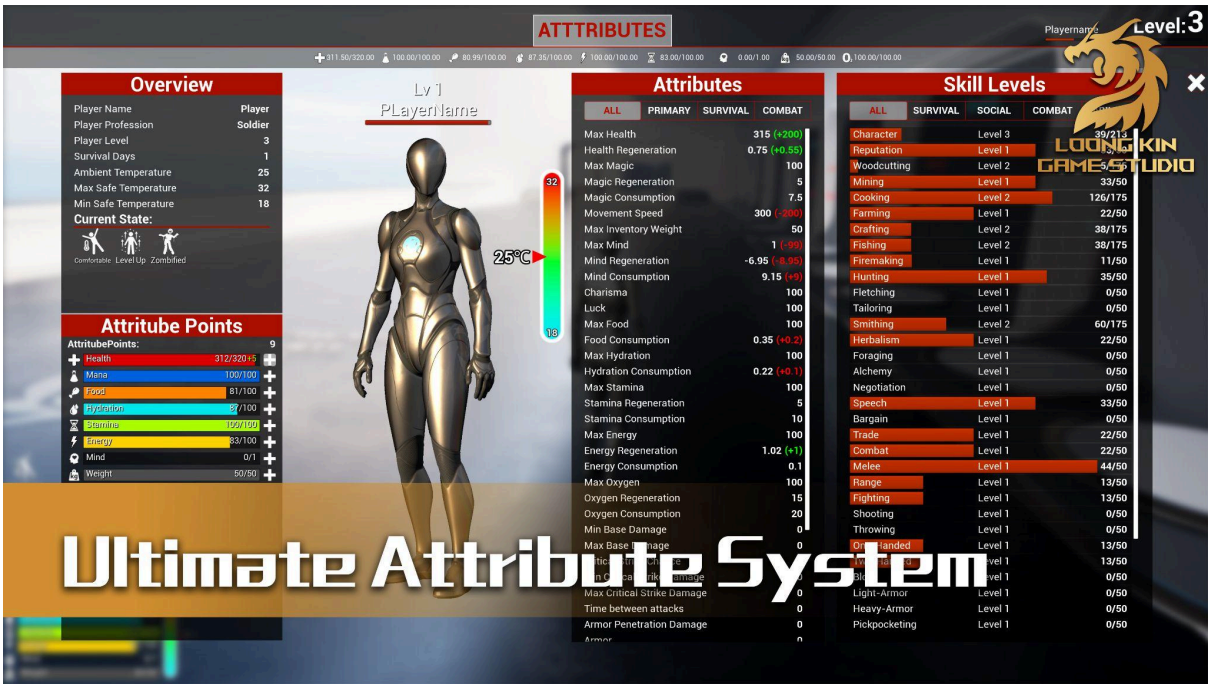
🌡️ Ultimate Attribute System

Ultimate Attribute System

DOCUMENTATION



Produced by Loong Kin Game Studio

Fab link:

Discord:https://discord.gg/3kYpWxv3BT

Youtube :https://youtu.be/nXpCIpVCFDo

📦Introduction

🧠 Ultimate Attribute System – Modular Survival Stats, Emotions & Realism

Survival is more than numbers.
The **Ultimate Attribute System** brings your characters to life through a deeply modular, fully customizable framework that blends survival mechanics, emotional states, and environmental impact — all in one powerful system.

🎮 **Designed for realism, built for extensibility**, this system covers everything from basic health and hunger, to temperature, stamina, hydration, emotional breakdowns, and even complex complications like infections or fractures that evolve over time.

💡 Whether you're building a hardcore survival game, an immersive RPG, or a deep simulation experience, this system gives you the flexibility and control to craft a truly living world.

🔧 Key Features

- 💡 **Five Powerful Modules**
    - **Attribute System** – Manage Health, Food, Hydration, Stamina, Energy, Mind, Weight, Movement Speed with full support for regeneration, consumption, max values, and modifiers.
    - **Leveling System** – Characters and skills both have level progressions. Earn attribute points on level-up and allocate them freely to customize your build.
    - **Temperature System** – Fully customizable temperature simulation, including current temperature, safe thresholds, and environmental effects.
    - **Respawn System** – Select spawn points at game start and choose respawn locations after death.
    - **State System** – 40 built-in status effects including physical, emotional, and environmental states. Each state supports duration, permanent effects, mind impact, and even delayed complications.
- 🎯 **Complication Mechanism**
  Each state can trigger complications after a delay, with a probability system — adding realism and unpredictability to status effects.
- ⚙️ **Highly Filterable UI**
  Built-in filter buttons allow players and designers to quickly sort stats into categories such as *Survival*, *Combat*, *Core*, *Social*, and *Abilities* — even with massive data sets.
- 🔄 **Seamless Ragdoll Integration**
  Fully compatible with [Ragdoll And Recover System] to simulate unconsciousness

and death. Easily configure any state to trigger Ragdoll with a single node call.

- 🧱 **DataTable Driven, Designer Friendly**
  All attributes, states, and complications are data-driven. Designed entirely in Blueprints for full customization and ease of use.

✅ **Seamless integration with *Ragdoll And Recover System***

Effortlessly simulate unconsciousness, death, or any custom state with ragdoll transitions. Perfect for states like *Unconscious* or *Collapsed*. Simply trigger the Ragdoll node — no complex setup required.

Ready to bring emotional depth, physiological realism, and modular stat management into your project?

✅ 100% Blueprint
✅ Easy to integrate
✅ Built for survival, RPGs, and more

🎓 Skill Requirements

Ultimate Attribute System is fully ready to use out of the box — no coding required. It's designed with flexibility in mind, making it suitable for developers of all experience levels, including beginners.

If you plan to extend or heavily modify the system, some **intermediate Blueprint knowledge** is recommended to help you navigate the more advanced customization features.

🤝 About Support

Support is provided **for all features included in this product**. If something isn't working as expected, I'm here to help.

For questions about integrating third-party assets, I'm happy to **discuss possibilities**, but please understand that I may not be able to offer guaranteed support — especially given the wide variety of assets available on Fab.

Also, if you have general questions about Unreal Engine itself (such as basic Blueprint usage or engine behaviors), I recommend checking the **official Unreal Engine Documentation** first, as those topics are beyond the scope of this asset's support.

Thank you for your understanding and support!

🧩Component Explanation

- **BP_AttributeComponent**: Contains all content related to character attributes.
- **BP_RespawnGameModeComponent**: A component used to handle character respawn within the game mode.
- **BP_UIComponent**: A component responsible for handling the UI.

1.AttributeComponent Config

1.1. Use Or Not: Configure which features you want to enable. Set unused features to `false`.

1.2. Fall Damage: Set the fall damage multiplier and minimum height threshold for triggering damage.

1.3. General:

- **Controlled By Player**: Whether the character is controlled by a player.
- **Health Regen Delay After Damage**: Delay before health regeneration starts after taking damage.
- **Attributes Low Threshold**: Dynamic attributes below this value will be considered low.
- **Growth Factor**: Growth factor that determines the required XP for leveling up.
- **Update Timer Interval**: Interval for updating dynamic attributes. Smaller values yield smoother changes but are more performance-intensive; larger values are more performance-friendly but less smooth.
- **Character Class**: Class reference of the player character, used when respawning.

- **Print Debug Text**: Whether to print test/debug messages.

1.4. Point:
Attribute and tech points granted on each level up.

1.5. Check Pawn:
Distance to detect nearby characters for Info UI display, and the relative position of the UI.

1.6. Temperature:

- **Check Cover Distance**: Upward trace distance for detecting shelter (cover).
- **Cover Tag**: Tag used to identify valid shelter objects.
- **Default Temperature**: The default ambient temperature.
- **World Z Temperatures**: Mapping between altitude (Z axis) and temperature.
- **Default Min Safe Temperature**: The default minimum safe temperature.
- **Default Max Safe Temperature**: The default maximum safe temperature.

📊AttributeTypes

To better organize character stats, all attributes are categorized into five types, each serving a distinct purpose:

- **Dynamic Values**
- **Primary Values**
- **Survival Values**
- **Combat Values**
- **Level Values**

You can check the enums prefixed with `E_Attributes` to see which attributes belong to each category.

⚙️State Configuration

All states are configured in the `DT_StateEffects` DataTable.

- **Effect Type**: Defines whether the state is Positive, Negative, or Neutral.
- **Effect State**: The specific type of state being applied.
- **Description**: The text description that appears when the state is active.
- **Icon**: The icon representing the state in the user interface.
- **Assigned Attributes**: The attributes affected by this state (e.g., Health, Stamina).
- **Temporary State**: Whether the state is temporary.
- **Duration**: How long the temporary state lasts (in seconds).
- **Apply Interval Damage**: Whether the state causes damage over time.
- **Interval Damage**: The amount of damage applied per interval.
- **Damage Interval**: Time between each damage tick.
- **Play Sound When Active**: Whether to play a sound when the state becomes active.
- **Active Sound**: The sound that plays upon activation.
- **Play Sound When Remove**: Whether to play a sound when the state is removed.
- **Remove Sound**: The sound that plays upon removal.
- **Play Anim Montage**: Whether to play an animation montage when the state is activated.
- **Anim Montage**: The specific animation montage to play.
- **Has Particle Effect**: Whether a particle effect is shown during the state.
- **Particle Effect Actor**: The particle actor used for visual effects.
- **Particle Actor Socket**: The name of the socket on the character to which the effect is attached.
- **Has Post Process Effect**: Whether a post-process effect is applied.
- **Post Process Effect**: The post-processing settings to use while the state is active.

- **Remove These Effects When Active (Probability)**: A list of other states that may be removed when this state activates, each with a probability.

- **Add These Effects When Active (Probability)**: A list of other states that may be added when this state activates, each with a probability.

- **Add These Effects When Remove (Probability)**: A list of states that may be added when this state is removed, each with a probability.

- **Has Complication**: Whether the state can trigger complications.

- **Complication Data**: The settings for complications, including delay, probability, and the state to be triggered.

🌀State Effect

🟢 Healthy

You are in good condition. No negative effects. All vital stats are stable.

🟣 Poisoned

You are poisoned. Health decreases over time unless treated. If poisoning persists, your mind may be affected, and further complications may arise.

🦴 Fracture

You have a fracture. Movement speed is reduced. Persistent fracture may cause pain, mental strain, or even bleeding.

😫 Fatigued

You are fatigued due to depleted energy. Stamina regeneration is greatly reduced. Sustained fatigue may impair movement or gradually affect your mind.

🍷 Drunk

You are drunk. Your physical faculties are impaired. Staying this state may lead to vomiting, mind disturbances, or even poisoning.

🧬 Infected

You are infected. Health regeneration is reduced. If left untreated, the infection may spread, causing pain, fever, strain on the mind… or something far worse.

🩸 Bleeding

You are bleeding. Health decreases over time. Persistent bleeding may cause weakness, pain, and potentially affect your mind and temperature. It may also attract unwanted attention in the wild.

💧 Wet

You are wet. No direct negative effects, but prolonged wetness may affect your temperature, help avoid catching fire, and possibly weaken your immunity.

🔥 Burning

You are burning. Health rapidly decreases. Being wet may help prevent or extinguish the flames. Burning causes severe damage and strains your mind.

🎒 Overburdened

You are overburdened. Movement speed is reduced, and consumption increases. Carrying too much may cause fatigue and strain your mind.

✨ Level Up

You have leveled up. Attribute points are ready to be assigned. Use them wisely to improve your chances of survival.

🍽️ Hungry

You are hungry. Continued hunger may affect your physical condition and strain your mind.

🚰 Thirsty

You are thirsty. Continued thirst may affect your physical condition and strain your mind.

💀 Critical

You are in critical condition. Further harm may be fatal, and your body and mind are under severe strain.

❄️ Freezing

You are freezing. Your body is shutting down from the cold. Staying cold may affect your movement and disturb your mind.

🧊 Very Cold

You are very cold. Your body struggles to maintain warmth, and your health slowly declines. Staying cold may lower your immunity, further affect your movement, and disturb your mind.

🥶 Cold

You are cold. Your body feels chilled. Staying cold may lower your immunity and mildly affect your movement and mind.

😌 Comfortable

You are at a comfortable temperature. Your body feels stable with no temperature-related effects.

🥵 Hot

You are hot. Your body feels warm and uneasy. Staying hot may cause heat-related issues, mildly increase your needs, and affect your movement.

🌋 Very Hot

You are very hot. Your body struggles to cope with the heat, and your health slowly declines. Staying hot may cause heat-related issues, further affect your movement, increase your needs, and disturb your mind.

🔥 Scorching

You are scorching. Your body is shutting down from the heat. Staying hot may affect your movement, increase your needs, and disturb your mind.

🥴 Fever

You have a fever. Your body temperature is elevated, consumption increases, and immunity decreases. Continued fever may affect your mind.

🤢 Sick

You are sick. Your body feels weak and uneasy. Sickness increases consumption and may cause discomfort, straining your mind.

😖 Pain

You are in pain. Discomfort may affect your movement and strain your mind. Pain can increase consumption and may lead to other complications.

🫁 Hypoxia

You are experiencing low oxygen. Your body struggles to function properly. Hypoxia may reduce movement and strain your mind.

😶‍🌫️ Suffocation

You have no oxygen. Your health rapidly decreases. Suffocation severely impairs your body and may disturb your mind.

😵 Exhausted

Your stamina is completely drained. You struggle to stay active, and continued strain may take a toll on your body and mind.

🧟 Zombified

You have undergone a strange transformation. Your body and mind are altered, pushing you beyond normal limits. This state may be linked to an infection.

🤮 Vomiting

You are vomiting. Your body is under stress, leading to increased consumption and possible discomfort. Vomiting may disturb your mind and worsen other conditions.

💔 Breakdown

Your mind is overwhelmed, causing erratic behavior and impaired decision-making. Survival becomes difficult as control weakens. Immediate recovery is critical.

😰 Anxious

You feel uneasy. Lingering anxiety may gradually strain your mind.

😱 Fearful

You are gripped by fear. It clouds your thoughts, tightens your actions, and may wear down your mind.

😡 Angry

You are filled with anger. You feel driven to act, but this intensity may cloud your judgment and strain your mind.

😄 Joyful

YYou feel uplifted. A clear and stable mind empowers your will to survive. This state may ease or dispel lingering negative emotions.

💃 Lively

You feel full of energy, ready to face the challenges ahead. Your body responds with ease, and your spirit feels light.

⚠️ Starving

You are starving. Your strength fades rapidly, and your body struggles to sustain itself. Extended starvation may weaken your immunity and disturb your mind.

🏜️ Dehydrated

You are dehydrated. Vital functions begin to fail, and your condition worsens with each passing moment. Continued dehydration may lead to collapse and severe mental strain.

🐌 Sluggish

You feel sluggish. Every movement is delayed, as if burdened by unseen weight.

💫 Stunned

You are stunned, briefly losing control of your movements and reactions. This state may cause confusion and strain your mind.

💤 Unconscious

You have lost consciousness and cannot control your body. Energy and resource consumption decrease. Recovery requires time or external aid. This state may offer temporary relief to a strained mind.

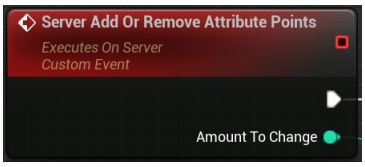
💫 Weakened

You feel weakened. Your body struggles to perform as usual. Persistent weakness may lead to further health issues or affect your mind.

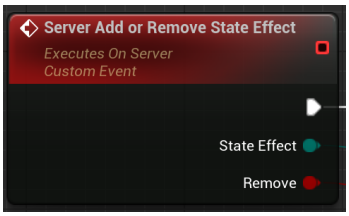
📖How To Use

1. Add Experience Points: Use `Server Add XP`.

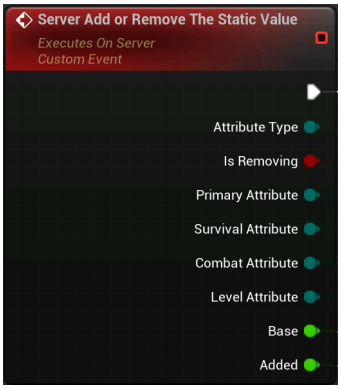
2.Add Attribute Points: Use `Server Add Or Remove Attribute Points`.



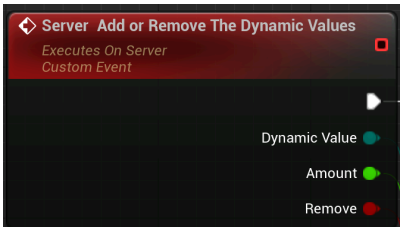
3.Add or Remove a State: Use `Server Add or Remove State Effect`.



4.Add or Remove a Static Attribute Value: Use `Server Add or Remove The Attribute Value`.

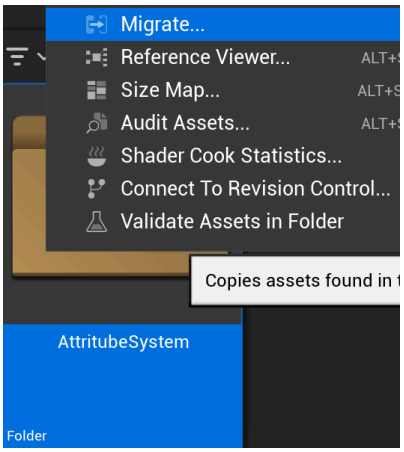


5.Add or Remove a Dynamic Attribute Value: Use `Server Add or Remove The Dynamic Values`.

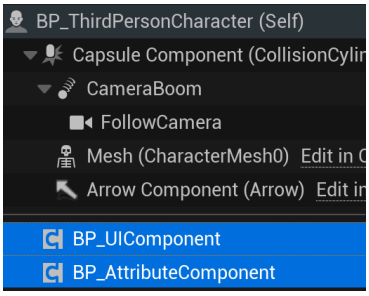


🔗 General Integration

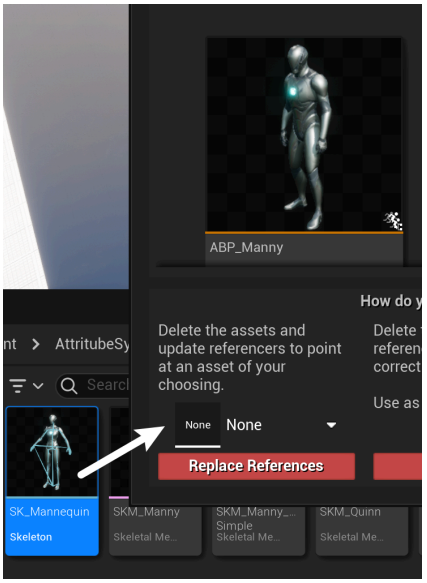
1. Miagrate the **AttributeSystem** folder into the **Content** folder of the target project.



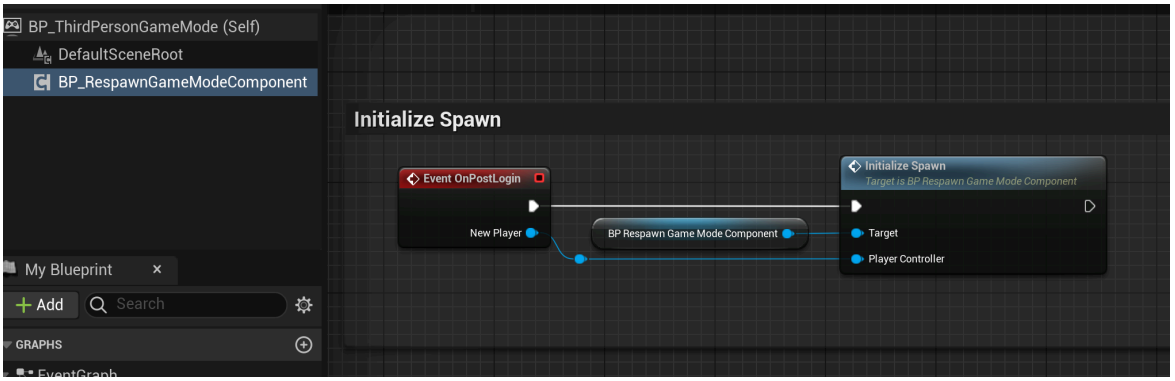
2. Add the two components to your character.



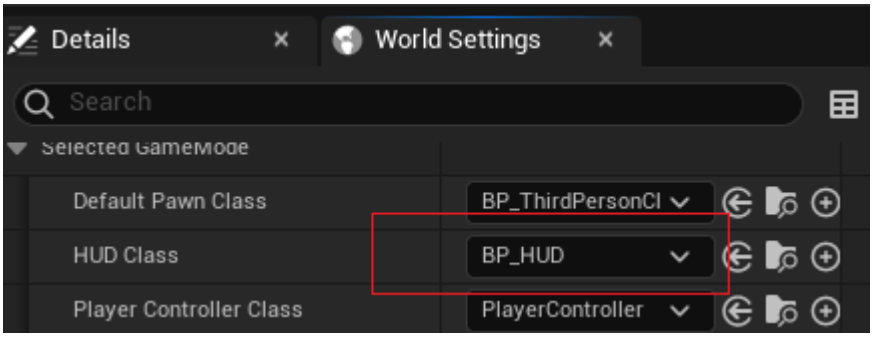
3. Delete the SK_Mannequin skeleton in the Demo folder and replace it with your own UE5 skeleton.



4. Add the **BP_RespawnGameModeComponent** and the **Initialize Spawn** node to your game mode.



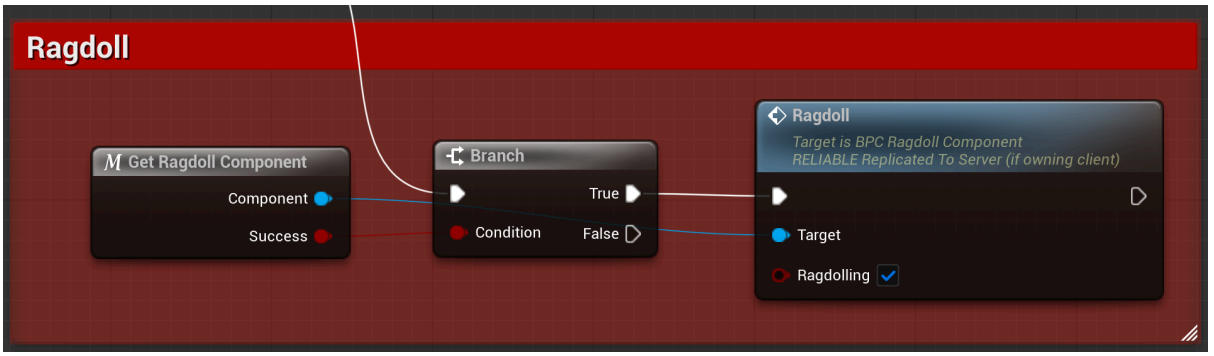
5. Set HUDClass to BP_HUD in WorldSetting.



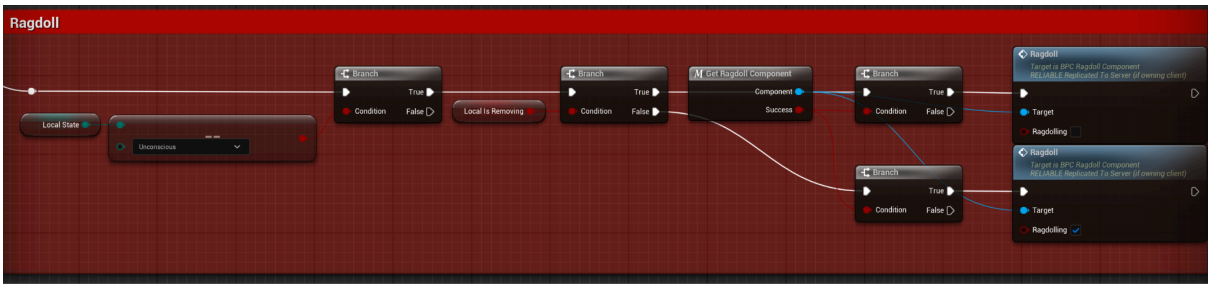
🧬 Ragdoll and Recover Integration

1. First, follow the general integration steps described in the [Ragdoll and Recover documentation].

2. In the `ActorDies` event in the **Damage&Die** graph inside the `AttributeComponent`, add the red-marked nodes..

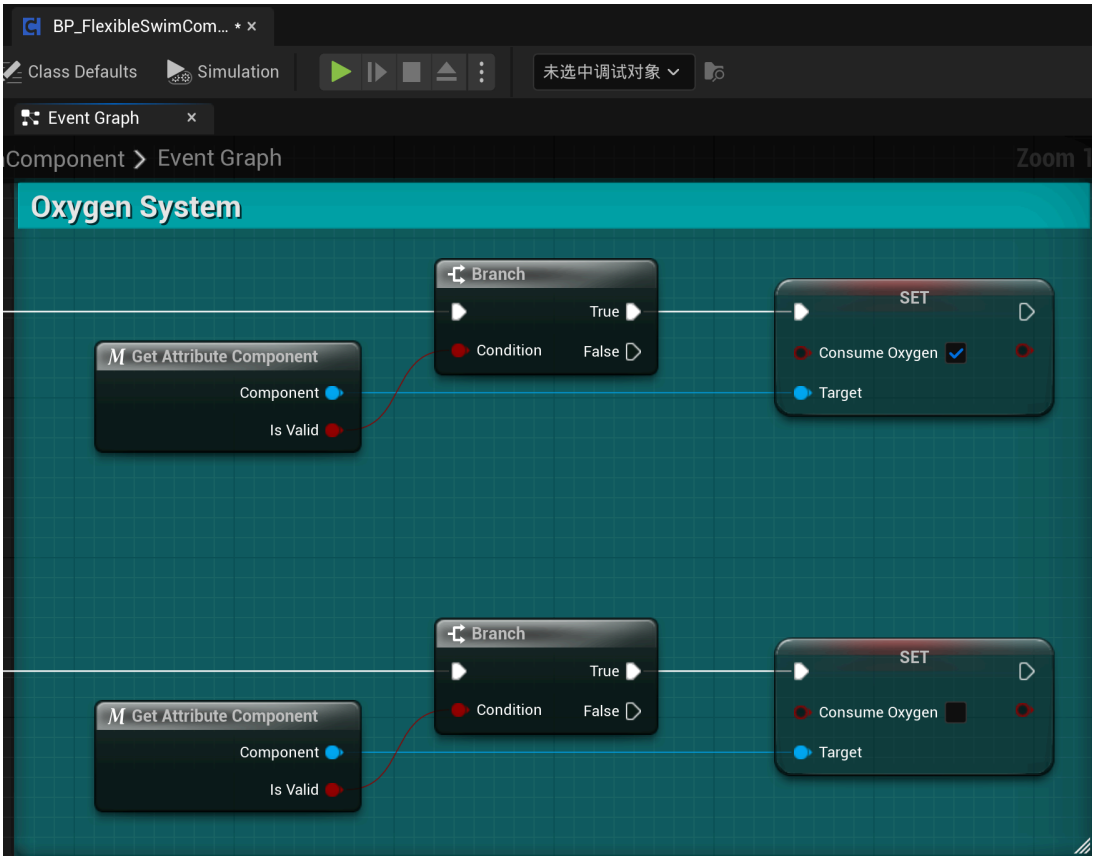


3. In the `Add or Remove State Effect` function of the `AttributeComponent`, also add the red-marked nodes.



🏊 Swim System Integration

1. First, complete the 🔗 General Integration steps to integrate the **Attribute System** into the **Swim System**.
2. In the **EventGrap**h of **BP_FlexibleSwimComponent**, add the Blueprint nodes shown in the image at the reserved spot for the **Oxygen System**.

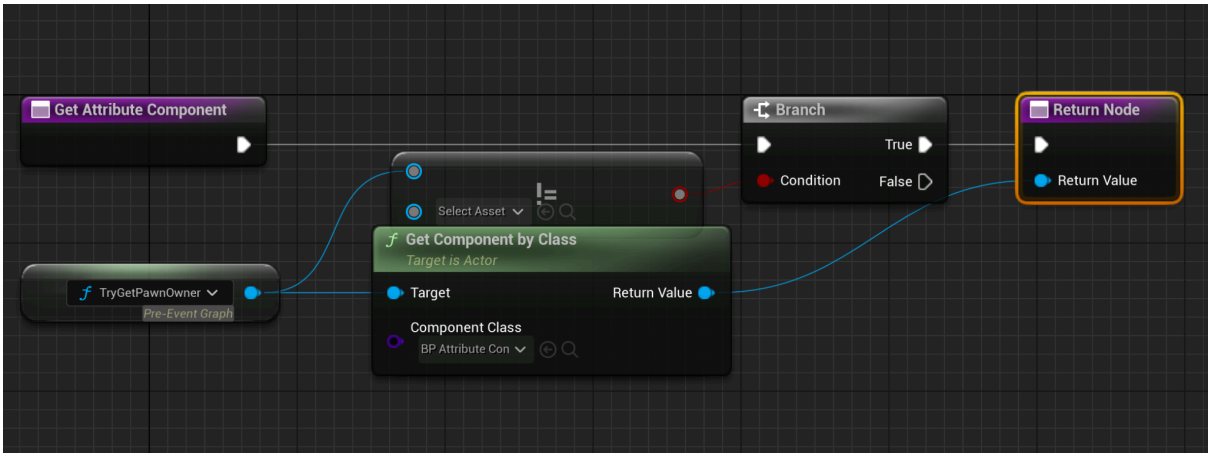


3. In **ABP_Swim**, add the **Get Attribute Component** pure function as shown in the image.

   Note:

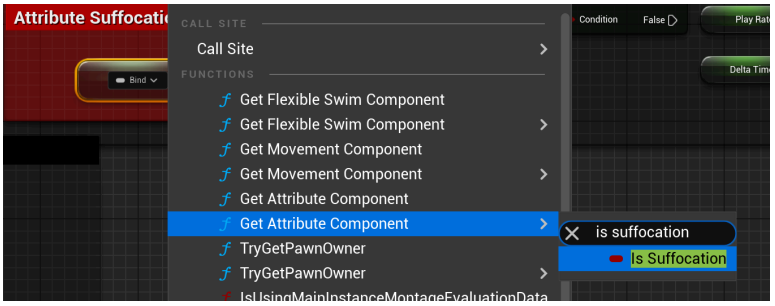
   ①. In **Get Component by Class**, set the **ComponentClass** pin to **BP_AttributeComponent**.

②. Name the output pin as **ReturnValue**, and also set its type to **BP_AttributeComponent**.

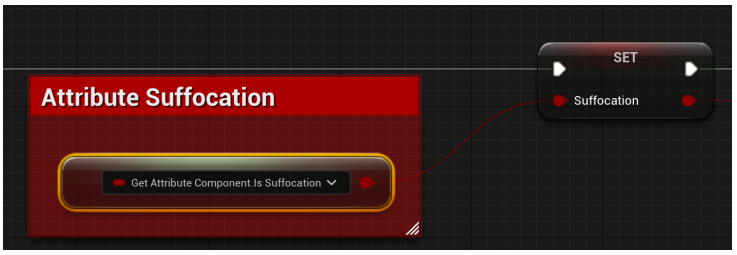


4. Open the **BlueprintThreadSafeUpdateAnimation** function in **ABP_Swim** and locate the **Property Access** node under the red **Attribute Suffocation** comment.

Click the **Bind** pin and, from the pop-up list, select **Get Attribute Component** followed by **Is Suffocation**.

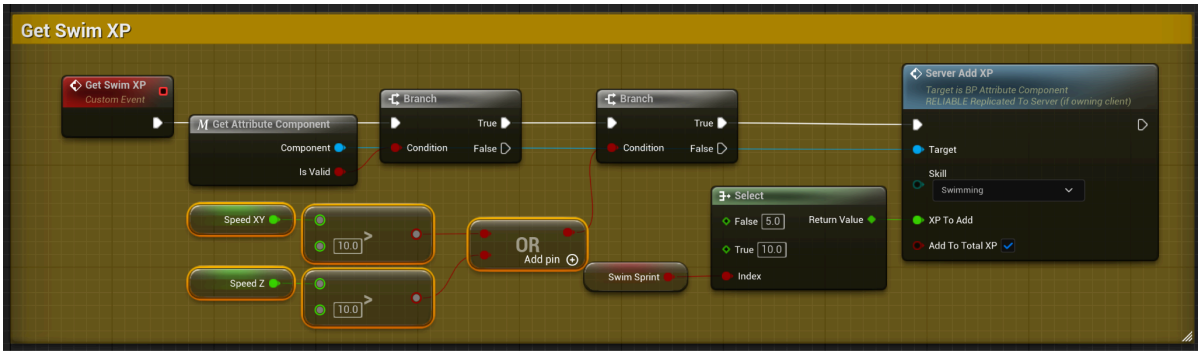


Connect the **Property Access** node to the **Set Suffocation** node.

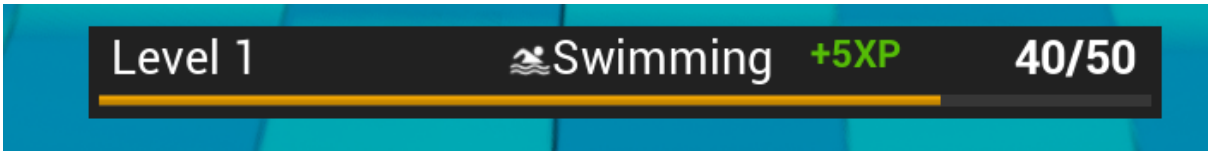


5. You can now consume oxygen while swimming.

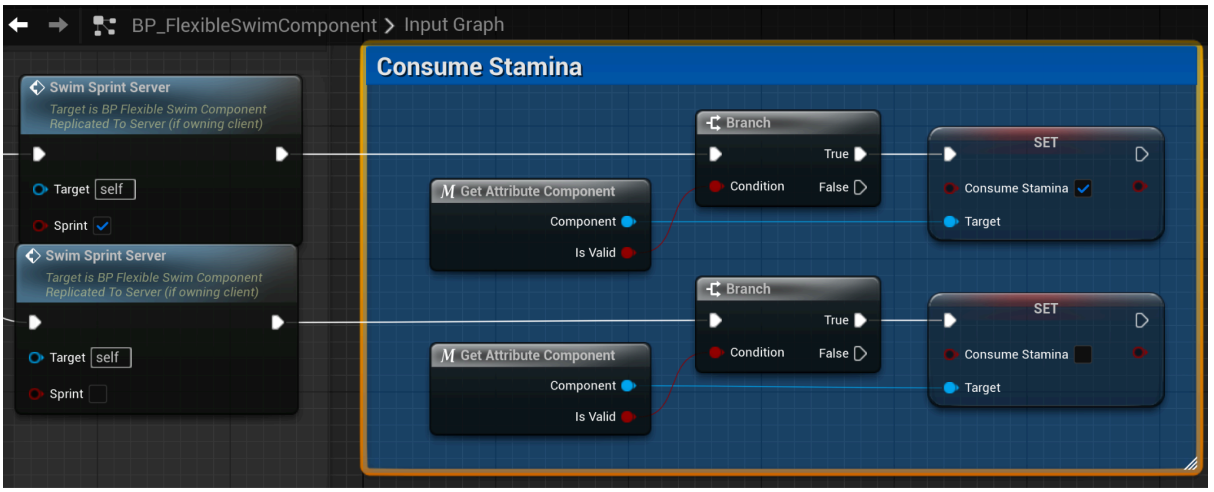
6. In the **EventGraph** of **BP_FlexibleSwimComponent**, add the Blueprint nodes shown in the image at the reserved spot for the **Get Swim XP**.



7. You can now get **Swimming XP** while swimming.



8. Add the nodes shown in the picture after the **SwimSprint** node in the **InputGraph** of **BP_FlexibleSwimComponent.**



9. Now you can **consume stamina** when you swim.

🗒️Release note

Version	Update	Date
V1.0		2025/6/2

Note:
Future updates will be posted in the ChangeLog section of the product page, and will no longer be explained in this document.