Too Many Eyes / Resource System

System Design Document

# Changes

## V 1.0

Editor: Perrin Peterson

12/08/2022

* Document Created, and initial version

# Introduction

This document details the design and purpose of a locomotion system designed for Too Many Eyes. Designed for the Blight Brew Game, the document will cover;

* Usage for the designers.
* UML for the Programmers.

# Design Goals

The goal for this system is to give the player a way of acquiring resources for crafting. This system Will likely take the form of many different items, such as a “Harvest Spot” that the player can gather from.

# Behaviour

The system is mainly focused around supplying the player with resources. The system will be smart enough to allow for rarity of both drop amounts, and resources. For example;

***A common drop***- 1x grass

***An uncommon drop***- 3x grass  
- 2x Insect

And both drops can come from the same source, depending on the luck of the player. A lucky player will get multiple pulls from the loot pool, an unlucky one will only get a few. This is, of course, customisable.

# High Level Design

Graphical user interface

Description automatically generated with medium confidence

* Player system - the system is our core system and is what the player interacts with to use all our other systems.
* Locomotion system - an advanced movement system we’re using to give us more control over the players movement. This gives us variables we can control, such as stamina, and additional functionality, such as climbing and vaulting.
* Potion system – a system meant to be easy to use for designers. The system allows for the designers as much freedom as possible, while giving the coders a minimal amount of updating to do. This is the main system the player will use to complete puzzles and generate income.
* Inventory system – a system meant to be easy to use for designers. The system allows for the designers as much freedom as possible, while giving the coders a minimal amount of updating to do. The system works as a container to hold items that the player collects. This includes the players backpack, as well as storage containers around the world.
* Resource system – a system meant to be the first of the steps for the player to generate income. This system gives the player ways to harvest materials and shows off behavioural logic for when the system is used.
* Crafting System – the system in between the Resource system, and the Shop/Potion systems, in terms of actual gameplay. This systems job is to refine the resources the player gathers into other items or potions for use in one of the other systems.
* Shop system (Not yet implemented) – a system to generate income, and allow for the player to improve. The system interfaces with AI, the player, and Storage objects, to give the player a way of selling items to NPC’s. This can have numerous effects on the AI and is the primary source of income, a necessary resource for improving the players arsenal, and serves to break the monotony of just grinding for resources.
* AI System - The system dealing with the NPC’s in the world, including fauna. The system interfaces with the shop system, allowing for NPC’s to be customers, as well as gives the AI their logic for movement, interaction, and anything else.

# Mid Level View

Graphical user interface, website

Description automatically generated

*For more detailed information about ABBBaseItem, please see the Inventory System MDD.*

* ABBHarvestSpot is an actor that exists in the world and gives the player items based on the internal list of items. It has lots of logic for making items rare and mixing up what the player gets, such like giving the player more, less, or different items every time they interact with it.
* FBBHarvestableItemData is a struct containing the information that the HarvestSpot uses to give the player items, like the item, and its rarity.

# Logical View

Text

Description automatically generatedA picture containing table

Description automatically generated

# Process View

## Player interacts with a HarvestSpot

Diagram

Description automatically generated

1. Player Interacts with the HarvestSpot. (Pressing the interact button while looking at it).
2. Cast the InstagatingActor into a ABBPlayerCharacter.
3. If the cast is successful, execute the rest of the code.
4. Call GetInventoryComponent() on the player.
5. Store the return value in a local variable.
6. Call GenerateLinearAmountPullList().
7. Pull a random index from the list, storing in a local variable.
8. Loop until we have that amount of items pulled.
   1. Pull a random item.
   2. Call AddToInventory() on the InventoryComponent, passing in the random item, and 1.
9. Set the Static Mesh of the MeshComponent to the AfterHarvestMesh.
10. Set IsInteractable to false.
11. Add the prefix “(Harvested) ” to the FocusText.
12. Play the harvest sound at the location of the HarvestSpot.
13. Deactiate the particleSystemComponent.
14. Set a timer to ReActivate the HarvestSpot.

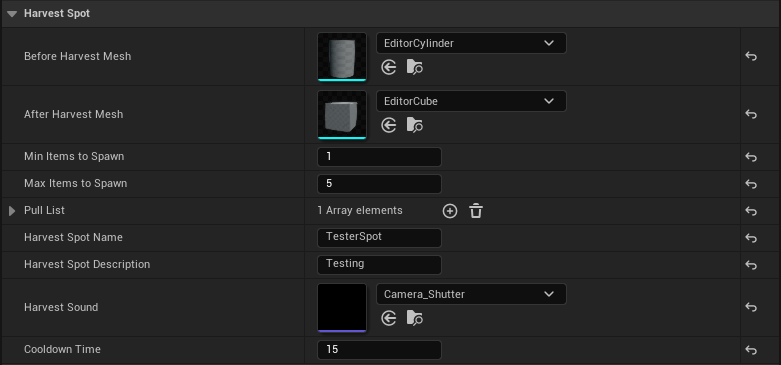
# Use Case View

## Harvest Spot

### Set a before and after mesh

The harvest spot supports a changing mesh that changes once its interacted with.

1. Navigate to the Harvest Spot section of the blueprint settings.

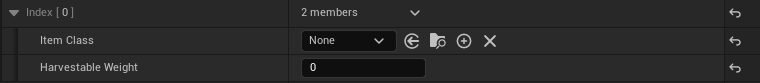


1. Two variables here are specifically for this, the Before and After Harvest Mesh. The before is what it looks like when the player can harvest it, and after is what it looks like when the player harvests it.  
   Graphical user interface, application, website

   Description automatically generated

### Add an item to the harvest spot

1. Navigate to the Harvest Spot section of the blueprint settings.  
   A screenshot of a computer

   Description automatically generated with medium confidence
2. The Pull List is what you’re looking for, add a new element.  
   
3. Expand the new index of the array.  
   
   1. Set the Item class to the BP of the desired item.
   2. Set the Harvestable Weight, the weight is easier to understand of you think of it as the “chances” you have to get this item, out of ALL the items in this list. So if you have 5 in here, you have 5 chances to get the item.