EasyQuest Manual

By Zackary Direen

Zackary Direen

The Academy of Interactive Entertainment Ltd  [Company address]

**Table of Contents**

**About EasyQuest 2**

**Core Components 3-5**

**Player 3**

**NPC 3**

**Quest 3**

**Quest Log 4**

**Leveling System 4**

**InteractInterface 4**

**UI Elements 5**

**Features 6-13**

**Adding/Removing a Quest 5-8**

**Adding/Removing Objectives 9-10**

**Kill Objectives 11**

**Gather Objectives 11**

**Interact Objectives 11**

**Location Objectives 12**

**Adding Prerequisites 13**

**Contact Information 14**

**About EasyQuest**

EasyQuest was developed in 2019 as a response to the lack of Questing systems that are easy to implement within other people role playing games that are in development. EasyQuest allows developers the ability to easily implement a quest system that is not difficult to set up and use of their own personal project. It features the use of 4 different object types and a quest log component that is directly attached to the player to hold the quests the player is currently using. It also features quest markers for developers to allow players to track where a quest can be picked up or completed.

**DISCLAIMER**

Zackary Direen would like to personally thank Virtus Learning Hub for providing the player model that was used for the demonstration scene as well as the inspiration for the skills that are currently within the game, as well as some of the UI elements. I would also like to thank Ryan Laley, who also was able to provide me with assistance though his YouTube series on some of the critical problems I was facing in the development of this project, as well as providing the underlying foundation for some elements of EasyQuest, as well as the esstainals for the systems implemented that were not EasyQuest itself, such as the inventory system and the leveling system.

**Core Components**

There are 6 core components to EasyQuest. These components in its entirety allow for EasyQuest to function. As without these components, everything is unlinked and EasyQuest will not function correctly.

**Player**

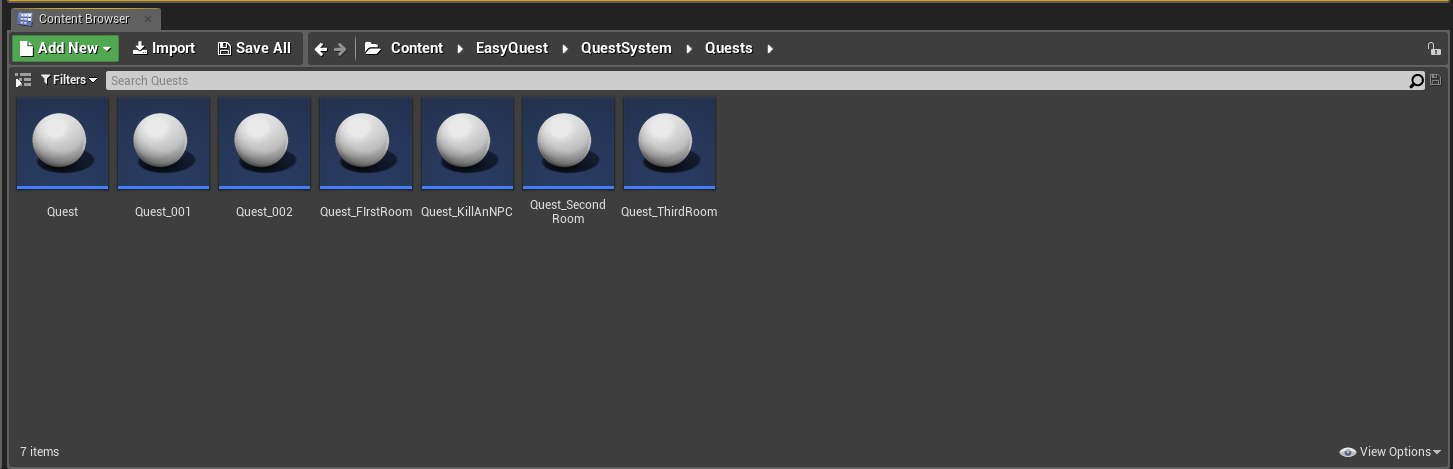
The player is the player character within the game and will be what the player of your game is controlling. They will be responsible for holding the quest log component, as well as the leveling system.

**NPC**

These are Non-Player Controlled characters that are present within your game world. They will be responsible for holding one quest that the player can complete.

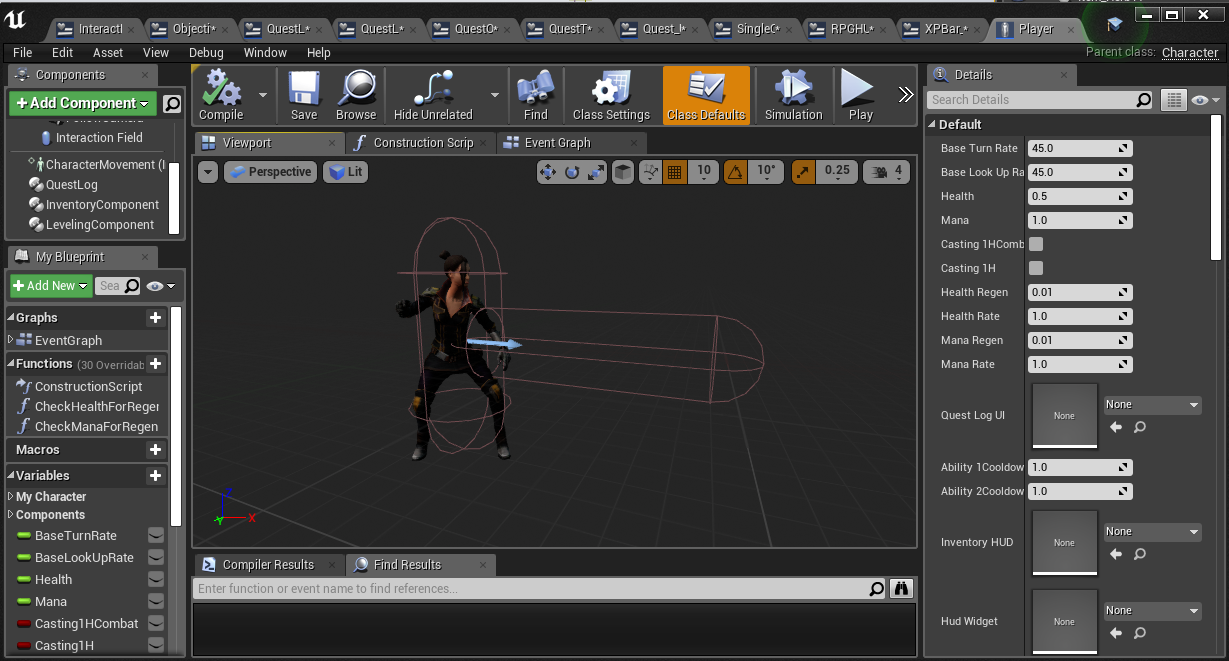
**Quest**

These are an individual set of tasks that the player will be set to complete. Each quest that is created will be derived from the “Quest” parent blueprint class that is within “QuestSystem/Quests”. Any changes that are made to the parent class **WILL** affect every other quest.

****

**Quest Log**

This component will be attached to the player character and will be used to keep track of the quests that the player is currently undertaken. Ensure that a QuestLog component is attached to your player as per the example below.



**Leveling System**

This is responsible for the leveling component for the player, which involves the players level and experience amount. Quests will directly affect this system by adding experience from quests to this system and increase the players level.

**InteractInterface**

This is responsible for the ability for the player to interact with NPC’s in order to accept and complete quests, as well as being able to pick up quest items for the purpose of “collect” objectives within quests.

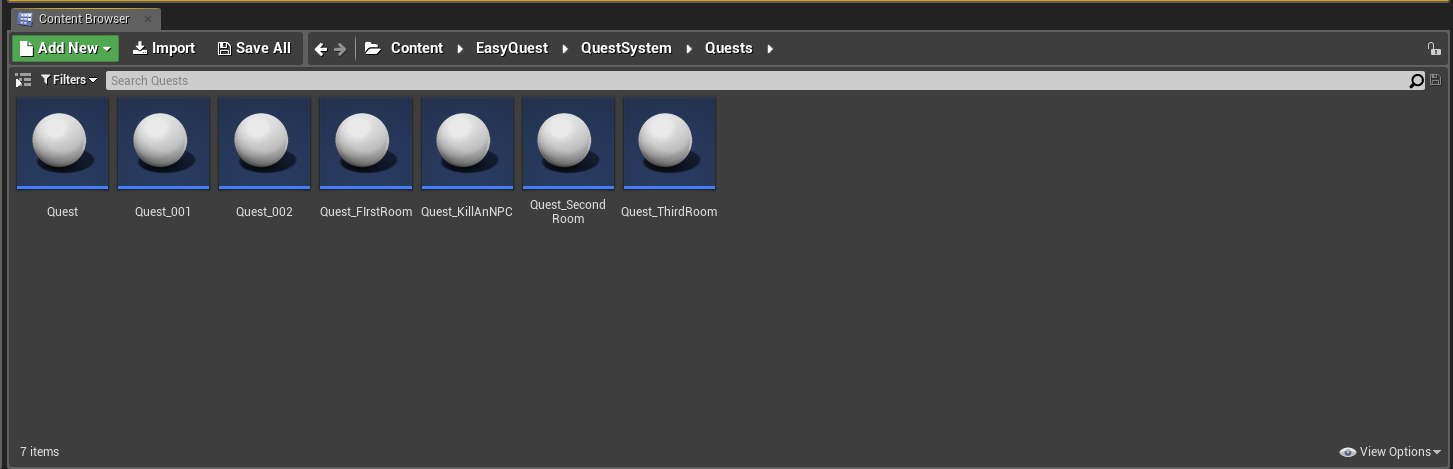
**UI Elements**

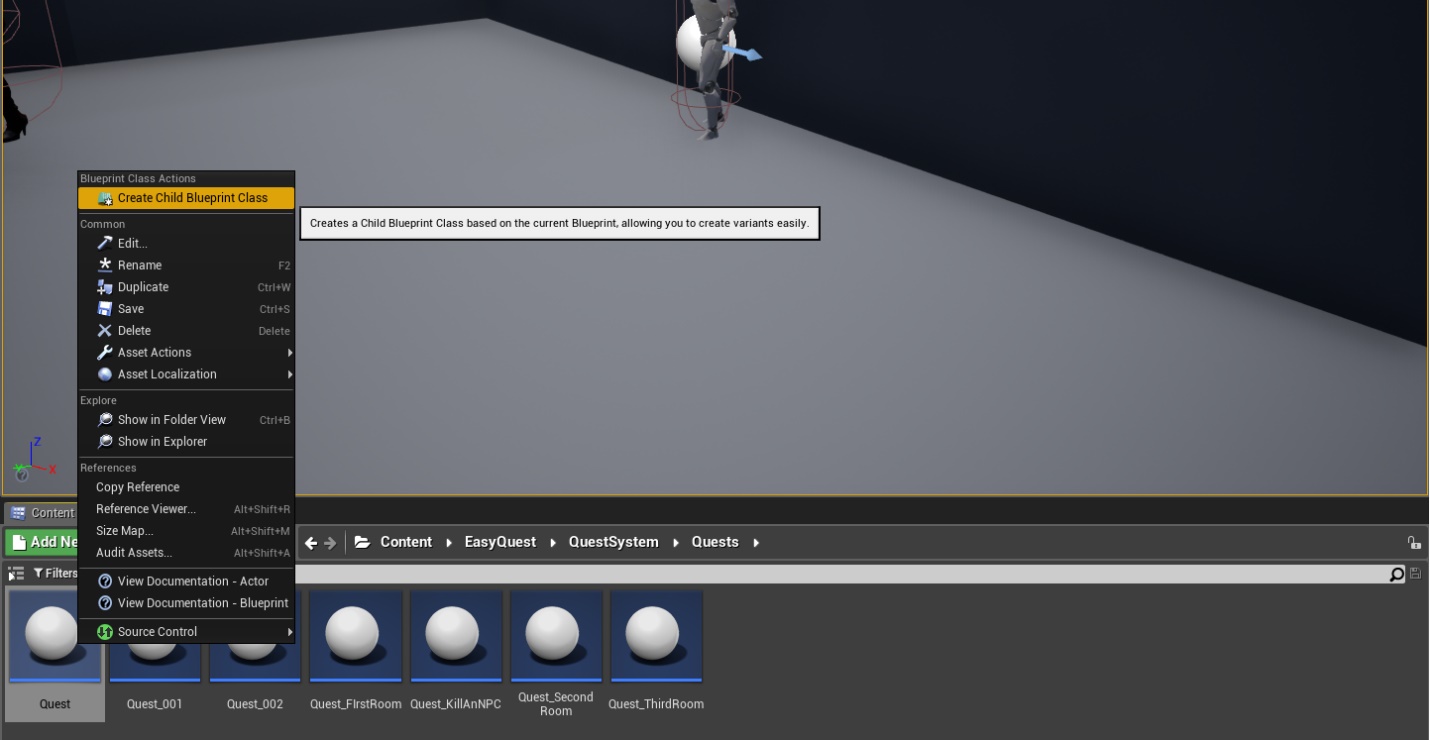
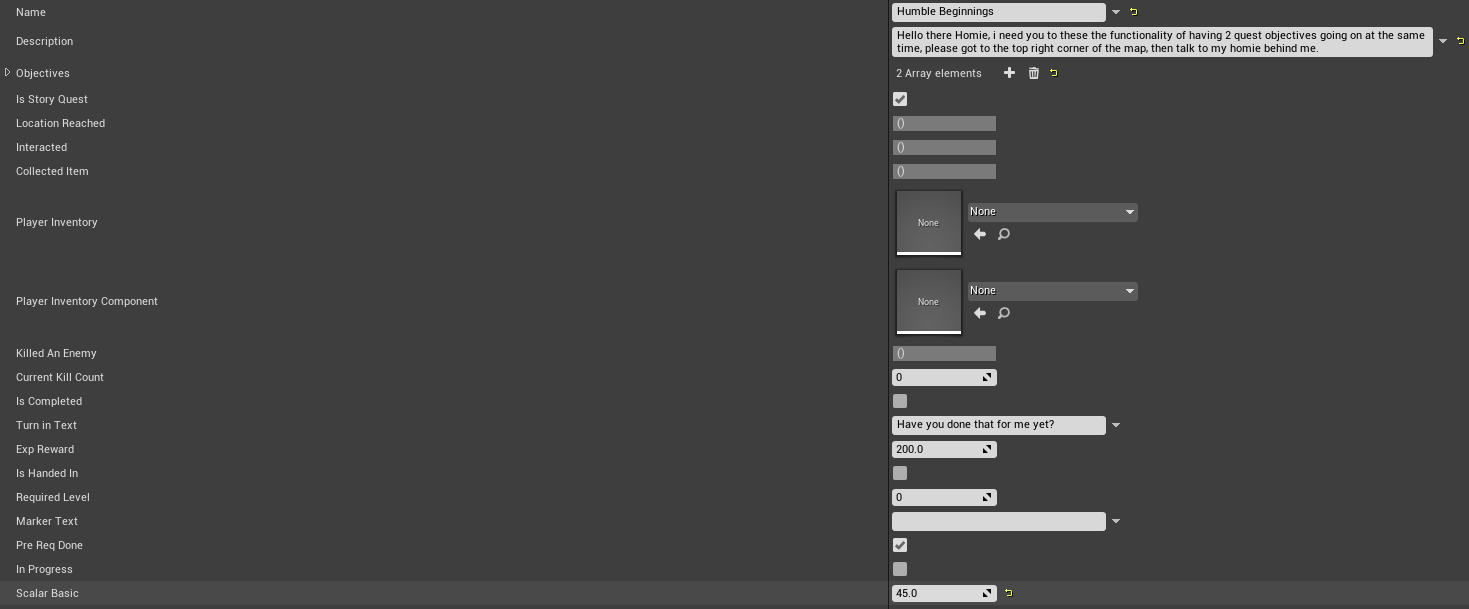
These are responsible for displaying information to the player about progress within quests, as well as displaying the experience amount and level of the player and the health and mana of the player.

**Features**

**Adding/Removing a Quest**

1. Navigate to the folder EasyQuest/QuestSystem/Quests.

****

1. ****Right-click on “Quest” and click ”Create Child Blueprint Class”. Rename the quest in the folder however you wish, as this is not the name that will be present within the game.
2. Double click on your newly created quest.
3. ****Notice the different fields you are able to access. I shall highlight the main one that you should be focusing on.

**Name:** This is the name of the quest that will appear in game and within the quest log component.

**Description:** This is the description that will be given when you interact with an NPC or when you look at the quest log, giving the story behind the quest.

**Objectives:** This uses an enumerator of objective types and are essentially your quest objectives that you will set for the player to complete.

**Is Story Quest:** This will determine if the quest is a part of a story quest or a side quest.

**Turn in Text:** What the NPC will say in the description box when you proceed to hand in a quest.

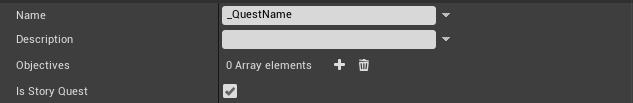
**Exp Reward:** How much experience the player will receive when they complete the quest.

Accessing these will be important for customizing your quest later.

**5.** Should you ever want to delete a quest, right click on a user created quest and select Delete.

**Adding/Removing Objectives**

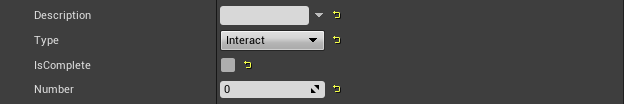
1. In order to add an objective, click on the ‘+’ icon next to “Objectives”.

****

1. You will now see that a blank object has been created. This will display the array index of the amount of objectives (starting with 0) and incrementing for each objective you add. in order to edit your object, click the expand triangle next to the array index.

****

1. You will now see 4 new fields.

****

Here is a description of each field.

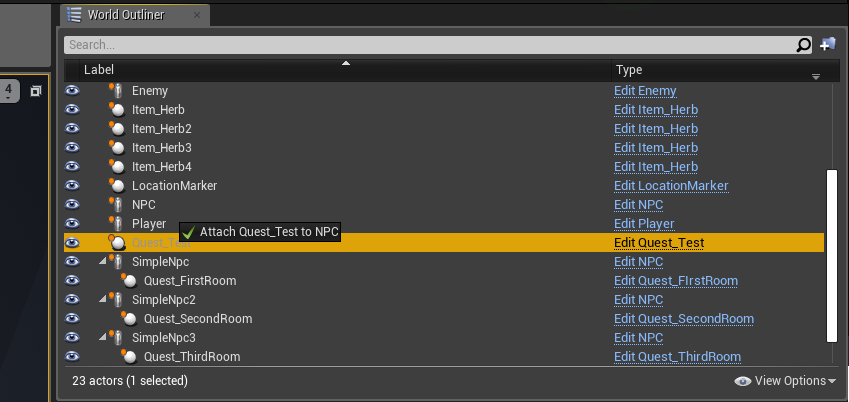
**Description:** This a description of the objective itself.

**Type:** What objective type this objective will be.

**IsComplete:** This value is used to keep track of an objectives progress and should be left unticked.

**Number:** This is used for the tracking of items which require a specific number of items to complete. This value is only relevant to Kill and Collect objectives. Otherwise, leave these values blank.

1. Prefill these fields as much as you can, then exit out of the quest editor and view your level.
2. Drag an NPC into your game world. This will be used for holding a quest. You can find NPC’s within “EasyQuest/Models/Blueprints.”
3. Once the NPC is within your level, drag in the quest you want to attach to the NPC near it.
4. Within the world outliner, find your quest and drag it in the world outliner to your NPC and release the left mouse button, attaching the quest to the NPC. You should see a prompt showing you that it will be attached to the NPC.

****

1. Once the quest is attach, reset the quests location co-ordinates to origin. This will ensure that the placement of the quest marker is correct. Should your NPC model be different, adjust position accordingly.
2. In order to complete the setting up your objectives, follow how to set them up below for each of the different types.

**Kill Objectives**

1. Set type to kill.
2. Set the target to be the enemy that you want the player to kill as a part of the requirement. This should be an enemy that is already present within the level. The enemy should be dragged from the world outliner into the target field.
3. Set the number to be the amount you wish the player to kill.

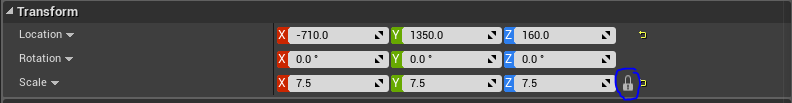
**Gather Objectives**

1. Set type to gather.
2. Set the target to be the item or amount of items that you want the player to collect as a part of the requirement. This should be an item that is already present within the level that the player can interact with, as such, make sure the item is a child of the initial “Item” class. The item should be dragged from the world outliner into the target field.
3. Set the number to be the amount you wish the player to collect.

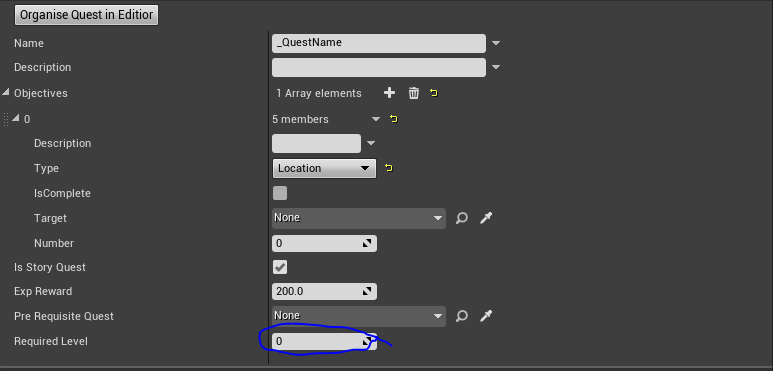
**Interact Objectives**

1. Set type to Interact.
2. Set the target to be the NPC that you want the player to Interact with. This should be of the “NPC” type.

**Location Objectives**

1. Set type to Location.
2. Place a location marker within the level where you wish for the player to explore. This should be under the base “QuestSystem” folder.
3. In order to increase the size of the marker, Increase the scale of the item. Ensuring that all axis of the marker are the same! (Ensure that the little padlock is in the locked position.)****
4. Once the marker has been placed and scaled as desired, set the target as the newly set location marker by dragging the location marker from the world outliner into the target field.

**Adding Prerequisites**

1. If you wish to add a prerequisite based on level, then under the ‘Required Level’ field, set the minimum level the player must be to be able to accept the quest. If this value is 0, then the pre requisite of a level is disabled.****
2. If you wish to add a pre-requisite in terms of a quest that must be completed, then drag a quest from the world outliner into the pre requisite quest field. This will mean that in order to accept this quest, the pre requisite quest **MUST** be completed. If this field is left on none, then this requirement is disabled.

It is worth noting that you can have one, both, or none of these.

**Contact Information**

Should you encounter any issues that are not covered within this manual, you can contact me via email at [zackary.direen@hotmail.com](mailto:zackary.direen@hotmail.com) and I will look into your issue and assist. Should the same issue be addressed multiple times, I shall expand upon this manual.