

# Full Guide: Setting up an Event for Cozy Tales

In diesem Guide wird von Anfang bis Ende ein Beispiel durchlaufen, wie man ein Event für Cozy Tales einrichtet.

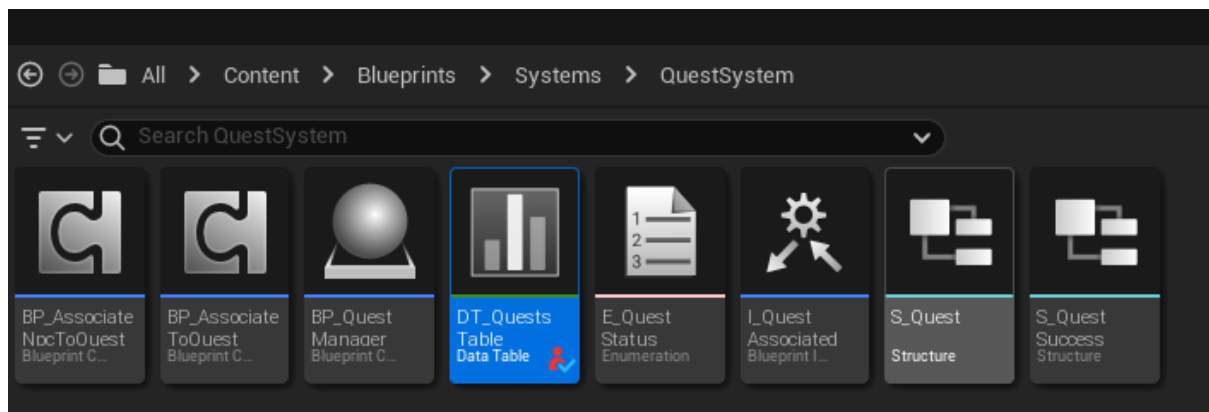
Dazu gehört...

- das **hinzufügen** eines **neuen Events** in die **Datentabelle aller Events**
- das einstellen, **wo der Quest Marker vom Spieler auf der Player Map platziert werden muss**, um das Event erfolgreich am Quest Board anzubringen
- das Einrichten der **Welt NPCs**, die mit ihrem **aktiven Dialog** dem Spieler die nötigen Infos für ein Event geben und das **Event überhaupt erst triggern**
- das **Einrichten des Event-Orts** im Level
  - welche Objekte und Npcs dort spawnen, sobald die Quest getriggert wurde
  - welche Objekte und Npcs despawnen sobald die Quest abgeschlossen ist
  - welche gespawnten Objekte und Npcs erhalten bleiben sollen, nachdem die Quest abgeschlossen ist
  - welche Objekte und Npcs erst gespawnt werden sollen SOBALD die Quest abgeschlossen ist
- das **globale** hinzufügen und entfernen **passiver Dialoge und Monologe** zu und von **allen NPCs von einer bestimmten Art** nach Abschluss einer Quest oder Fehlschlag
- das hinzufügen und entfernen **passiver Dialoge und Monologe sowie eines aktiven Dialogs** zu und von **SPEZIFISCHEN Welt NPCs** nach Abschluss einer Quest oder Fehlschlag

## STEP 1: Ein neues Event zur Datentabelle aller Events hinzufügen

Um ein Event mit dem Rest der Spielwelt kommunizieren lassen zu können müssen wir eine Referenz auf nötige Infos über das Event hinterlegen. Das tun wir in der Datentabelle, die an folgender Stelle zu finden ist:

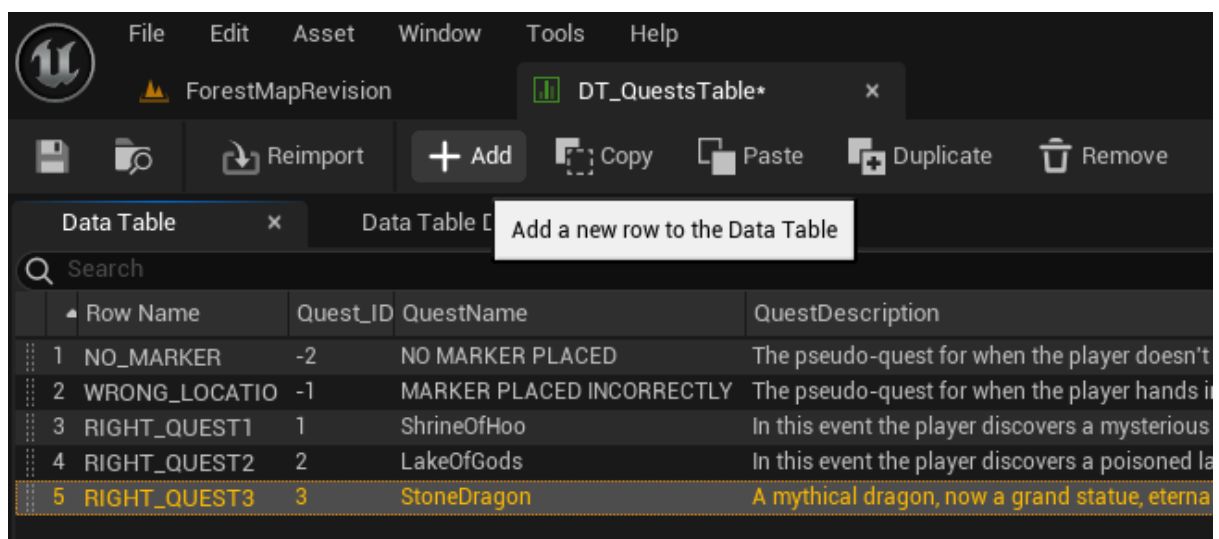
**Content/Blueprints/Systems/QuestSystem/DT\_QuestsTable**



Wenn wir die Datentabelle öffnen erhalten wir eine Liste mit allen bisher definierten Events (sowie die beiden Fail-States mit Index -2 und -1)

Jedes Event benötigt eine **EINZIGARTIGE** ID, die **größer als 0** ist. (Bsp. Shrine of Hoo: 1, Lake of Gods: 2 usw.)

Um ein neues Event zu erstellen klicken wir über den Zeilen und Tabs auf "+ Add".

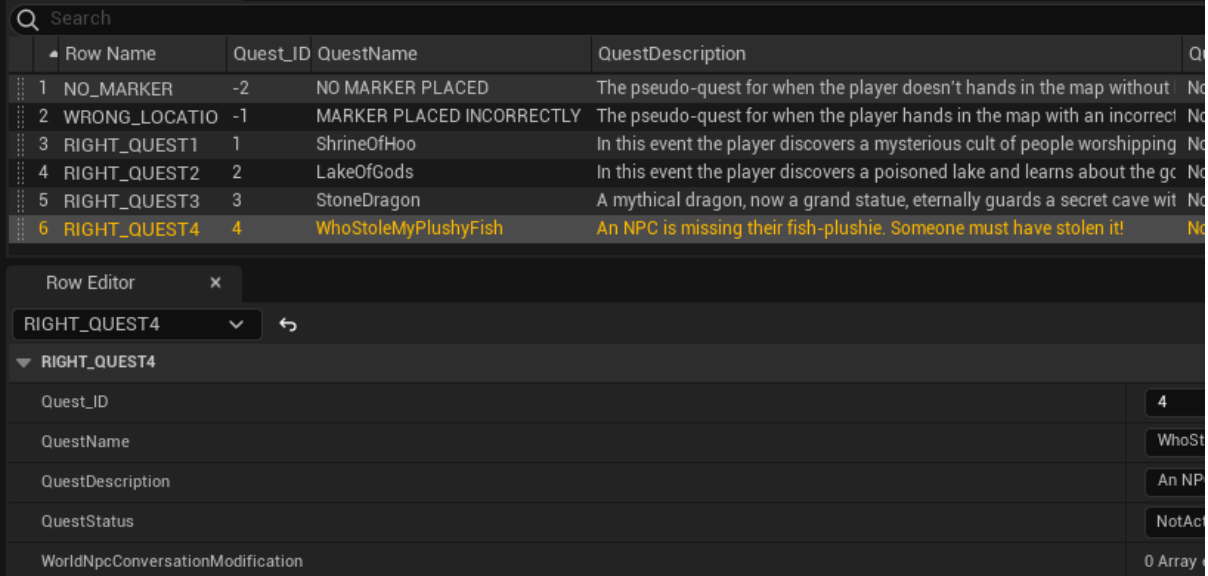


Es wird eine neue Zeile erstellt, in der wir die Infos für das neue Event eintragen können.

“Row Name”, “Quest Name” und “Quest Description” haben keinen Einfluss auf Mechaniken im Spiel und dienen nur der eigenen Übersicht für die Datentabelle.

Je nach Konvention, können hier dann Namen und Beschreibungen für diese drei Textfelder eingetragen werden. Für den “Row Name” in der Tabellenzeile auf “NewRow” Doppelklicken und für die restlichen Werte den Row Editor am unteren Ende des Fensters benutzen.

Danach nicht vergessen, dem neuen Event eine ID zuzuweisen. Präferabel die nächstgrößere ganze Zahl zur ID des vorherigen Events.



The screenshot shows a quest management interface. At the top is a search bar. Below it is a table with columns: Row Name, Quest\_ID, QuestName, and QuestDescription. The table contains six rows of quest data. The sixth row, 'RIGHT\_QUEST4', is highlighted in yellow. Below the table is a 'Row Editor' window for the selected quest. It has a dropdown menu showing 'RIGHT\_QUEST4' and a back arrow. Below this, the quest details are shown in a form with fields for Quest\_ID, QuestName, QuestDescription, QuestStatus, and WorldNPCConversationModification.

Row Name	Quest_ID	QuestName	QuestDescription
1 NO_MARKER	-2	NO MARKER PLACED	The pseudo-quest for when the player doesn't hands in the map without
2 WRONG_LOCATIO	-1	MARKER PLACED INCORRECTLY	The pseudo-quest for when the player hands in the map with an incorrect
3 RIGHT_QUEST1	1	ShrineOfHoo	In this event the player discovers a mysterious cult of people worshipping
4 RIGHT_QUEST2	2	LakeOfGods	In this event the player discovers a poisoned lake and learns about the gc
5 RIGHT_QUEST3	3	StoneDragon	A mythical dragon, now a grand statue, eternally guards a secret cave wit
6 RIGHT_QUEST4	4	WhoStoleMyPlushyFish	An NPC is missing their fish-plushie. Someone must have stolen it!

Row Editor	
RIGHT_QUEST4	↩
▼ RIGHT_QUEST4	
Quest_ID	4
QuestName	WhoSt
QuestDescription	An NP
QuestStatus	NotAc
WorldNPCConversationModification	0 Array

Um die restlichen Werte der Tabellenzeile kümmern wir uns später.

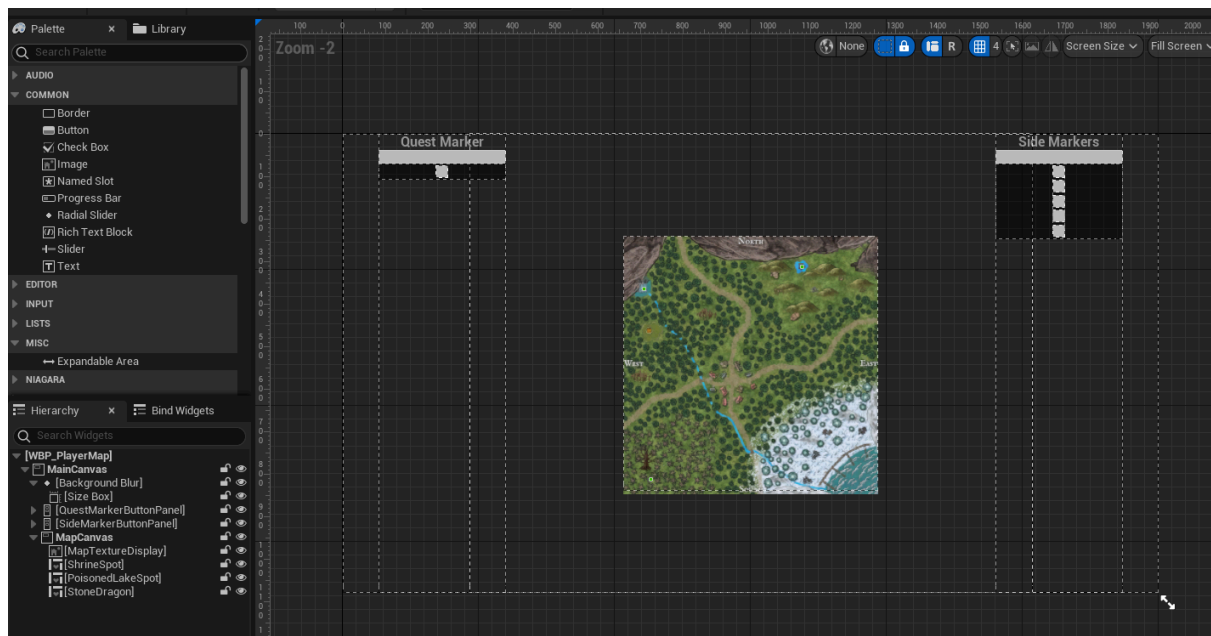
Damit ist fürs erste die Eventbasis eingerichtet.

## **STEP 2: Einrichten der Koordinaten des Events auf der UI-Map**

Um einzustellen, wo der Spieler den Questmarker hinsetzen muss, um eine valide Quest an das Questboard zu hängen öffnen wir das Playermap-Widget. Dieses ist zu finden unter...

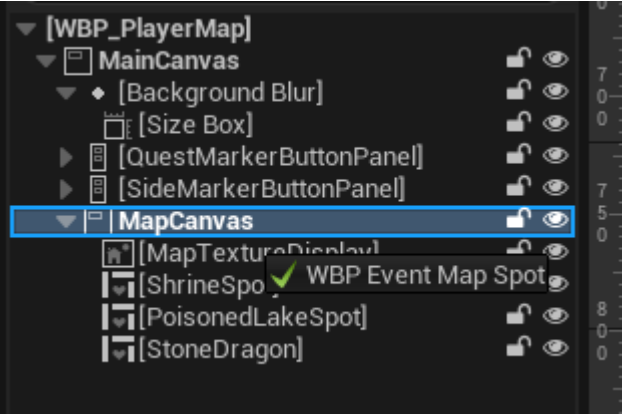
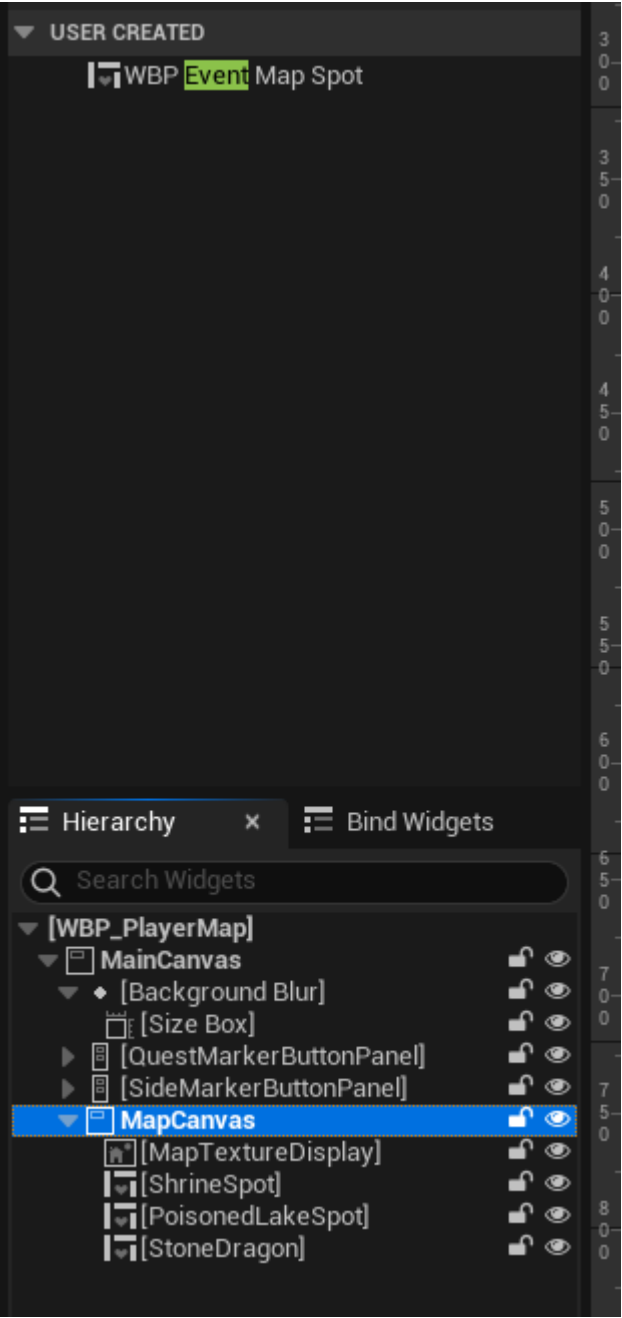
**Content/UI/Widgets/PlayerMap/WBP\_PlayerMap**

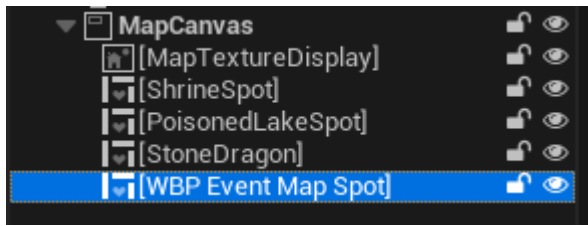
Der neue Tab sollte so aussehen:



Um die Event Location auf der Karte zu platzieren können wir in der Suchleiste der Palette (oben links im Bild) nach “Event” suchen und es sollte ein Suchergebnis namens “WBP Event Map Spot” auftauchen.

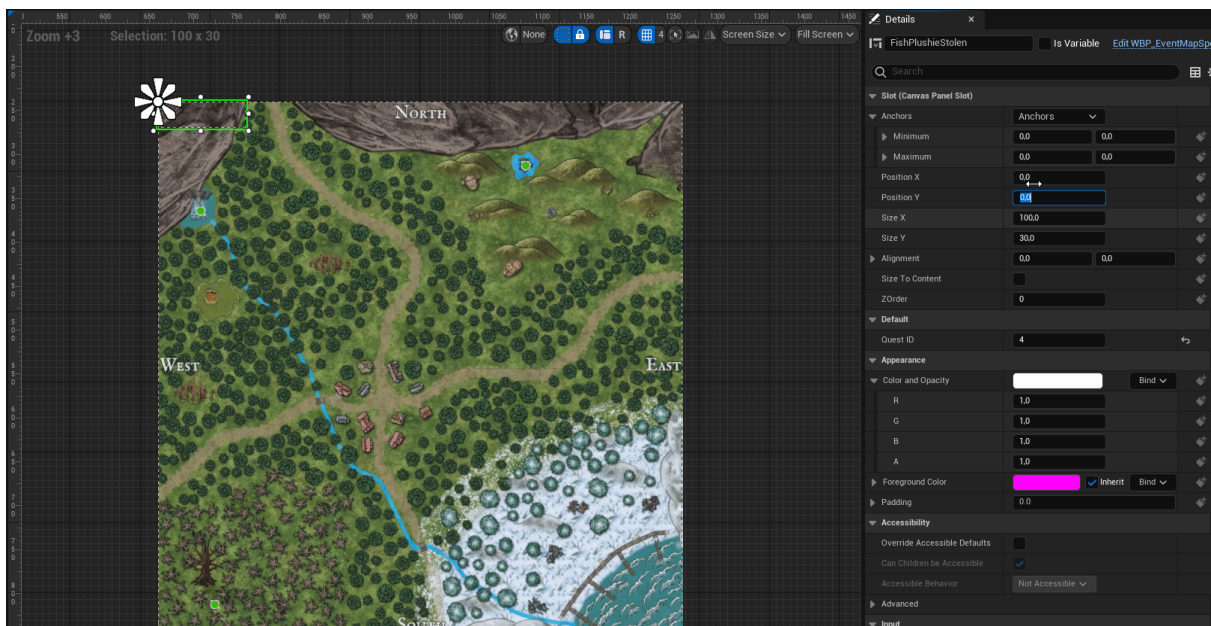
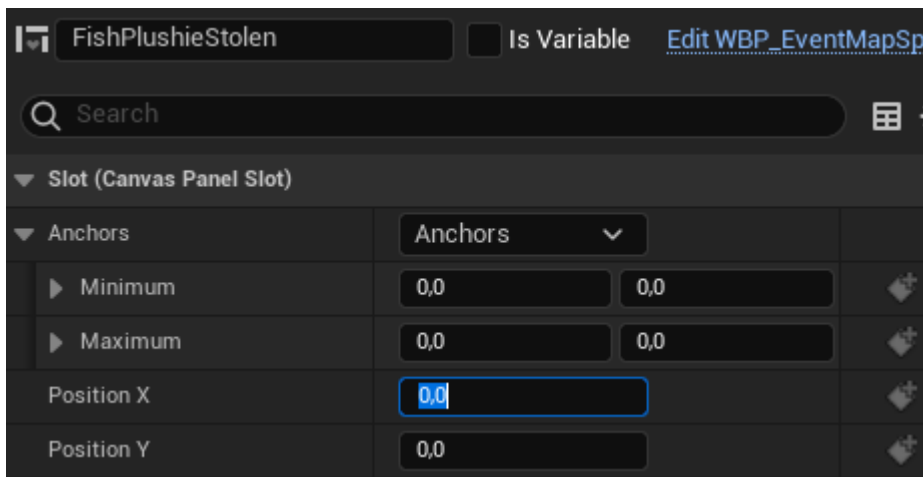
Unten in der Widget-Hierarchie gibt es ein Canvas Panel namens “MapCanvas”. In diesem platzieren wir per Drag & Drop aus dem Suchergebnis den Map Spot in der Karte.





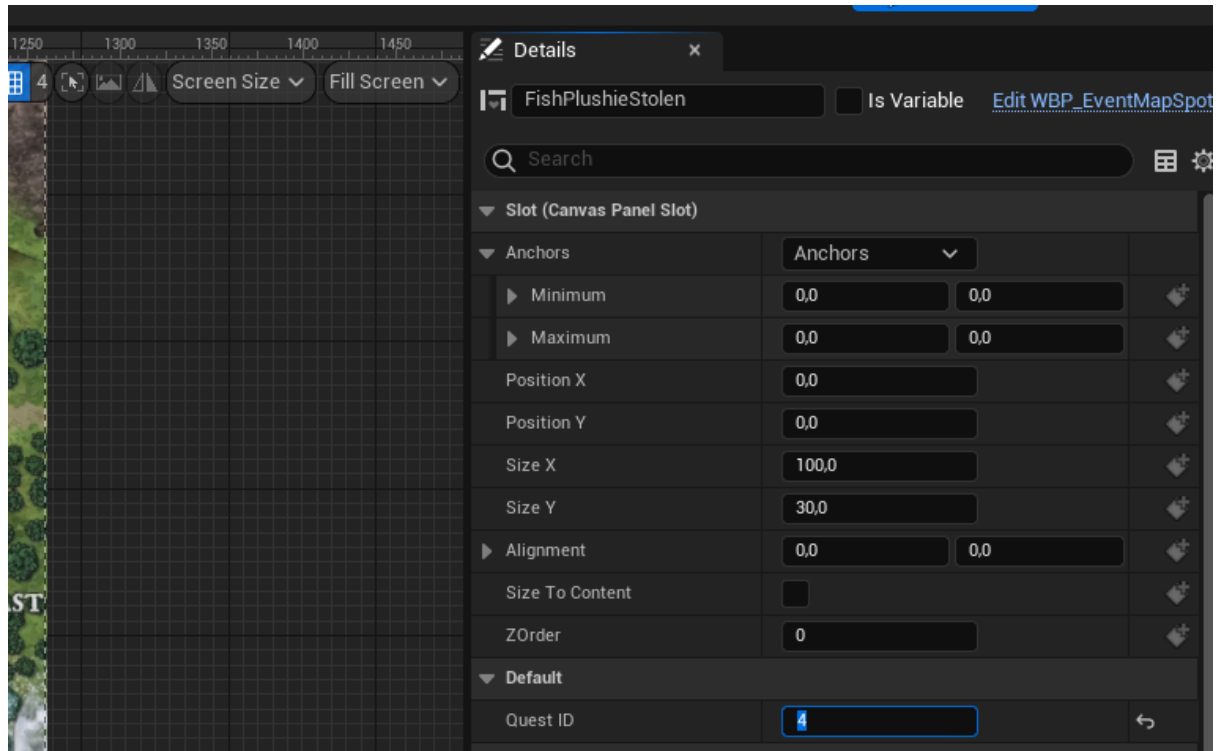
Diesen nennen wir dann (ähnlich wie die anderen Event Spots) so um, dass man es mit der betreffenden Quest in Verbindung setzen kann. Während der Name sich nicht auf die Mechanik auswirkt, tut es die Quest-ID, die diesem Event Map Spot zugewiesen ist.

Weil es schwierig ist, den Spot mit der Maus auszuwählen, um ihn zu verschieben, kann man die "Position X" und "Position Y" Slider des Inspektors des Spots benutzen, um diesen richtig zu positionieren.



Nicht vom grünen Rechteck verwirren lassen. Was zählt, ist wo der grüne Punkt in der oberen linken Ecke des Rechtecks landet.

Um diesen Event Map Spot auf die zuvor in der Datentabelle erstellten Quest zu beziehen müssen wir nur im Inspektor des Event Map Spots die "Quest ID" Variable auf den selben Wert setzen, wie die Quest ID des Events. (In diesem Beispiel: 4)



Damit ist die Event-Koordinate auf der Playermap definiert und das WBP\_PlayerMap Fenster kann geschlossen werden.

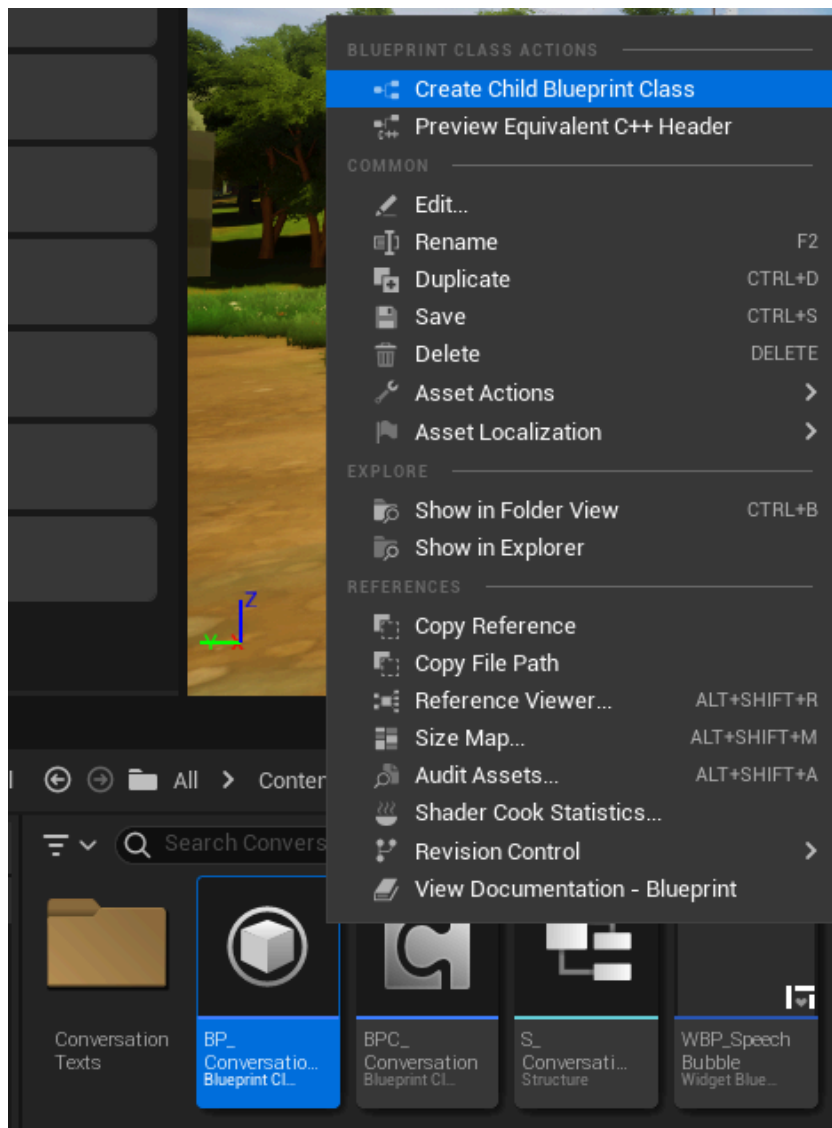
## **STEP 3: Erstellen einer Konversation**

Da alle Events by default zum Start des Spiels gelockt sind und für den Spieler nicht wirklich auffindbar, markierbar oder irgendwie interagierbar, müssen wir dafür sorgen, dass der Spieler überhaupt erstmal von diesen Events erfährt (was auch gleichzeitig dazu führt, dass Events aktiviert werden). Dafür benutzen wir die NPCs des Spiels, denen der Spieler beim reden zuhören kann.

Dafür legen wir erstmal ein Objekt an, in dem die zu sprechenden Zeilen einer NPC-zu-NPC Konversation enthalten sind.

Dafür erstellen wir uns ein Child-Blueprint vom Base-Conversation Objekt. Dieses ist auffindbar unter...

**Content/Blueprints/Systems/AI/Conversation/BP\_Conversation\_P**





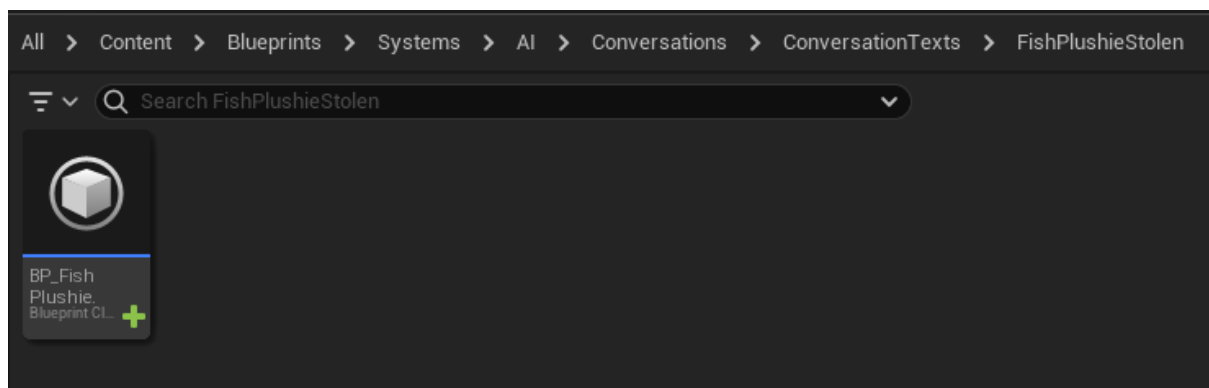
Das neu erstellte Objekt benennen wir so um, damit wir später wissen, auf welche Konversation es sich bezieht.

Hier nennen wir es BP\_FishPlushieStolen1

Zuerst ziehen wir also zwei beliebige NPCs in die Welt, an einen Ort, wo der Spieler sie finden kann.

Im “ConversationTexts” Ordner legen wir dann einen Unterordner für das Event an. Hier: “FishPlushieStolen”.

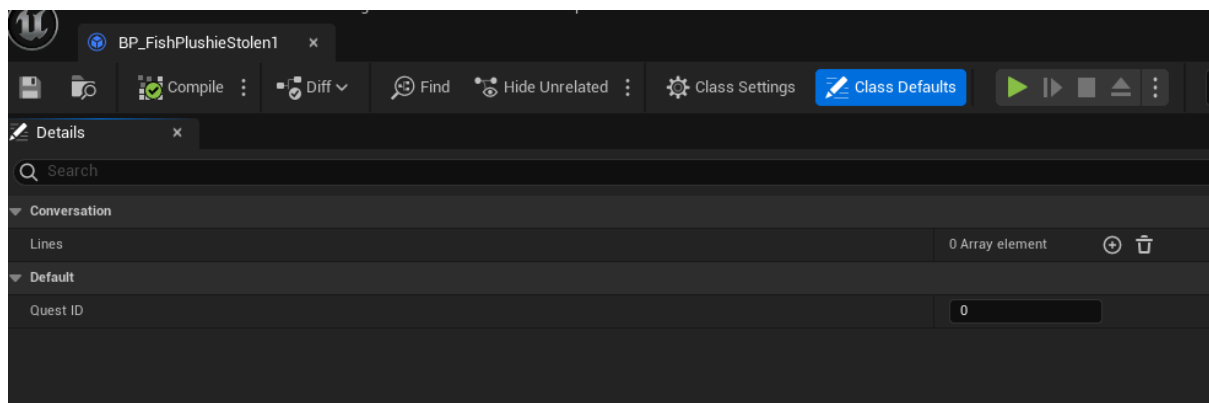
In diesem Ordner legen wir das neu erstellte Objekt ab.



Alle zukünftig erstellten Konversationen, die mit diesem Event zu tun haben legen wir in diesem Ordner ab.

Zum einstellen der Textzeilen öffnen wir das neu erstellte Blueprint.

In dem sich neu öffnenden Fenster können alle Tabs, bis auf “Details” geschlossen werden. Das sollte dann so aussehen.



Um der Konversation die erste Zeile hinzuzufügen, klicken wir auf das ‘+’ in der “Lines” Zeile.

Conversation

Lines 0 Array element

Default

Quest ID 0

Add Element

Lines 1 Array element

Index [ 0 ] 3 members

ConversationLine

AudioFile None

SpeakerIndex 0

Das machen wir für jede Zeile, die gesprochen werden soll. D.h. die Konversation wird so oft eine neue Sprechblase mit neuem Text anzeigen, wie wir dem Lines-Array neue Elemente hinzufügen. Diese werden dann während der Konversation chronologisch von oben nach unten durchlaufen.

Unter "ConversationLine" tragen wir den Text ein, der in einer Sprechblase angezeigt werden soll.

### **HIER LINK FÜR TEXTFORMATIERUNG EINFÜGEN**

Unter "AudioFile" kann eine Sound Wave Datei eingefügt werden, die da z.B. die hörbare Voiceline des gesprochenen Textes abspielen soll, wenn diese Zeile durchlaufen wird. Es kann natürlich jeder beliebige Sound eingefügt werden, der zum Zeitpunkt dieses Satzes abgespielt werden soll.












Dieser Slot kann leer gelassen werden, wenn kein Sound abgespielt werden soll.

Mit dem "SpeakerIndex" kann bestimmt werden, welcher NPC eine Zeile spricht (z.B. wenn man möchte, dass ein NPC mehrere Sätze hintereinander sagt).

Der Index '1' ist hierbei der NPC, der das Gespräch einleitet und Index '2' der NPC der "angesprochen" wird.




Lässt man den SpeakerIndex standardmäßig bei allen Elementen auf 0, werden sich die NPCs mit den Sätzen einfach abwechseln.

Das kann dann so aussehen:

3 Array elements	 
3 members	▼
Oh my god! Where is it?! Where is it?! It can't be gone!!! 	
None	<div>None ▼</div> <div> </div>
0	
3 members	▼
What? What happened? Did you lose something? 	
None	<div>None ▼</div> <div> </div>
0	
3 members	▼
My fish plushie is gone!! I took a walk around the north of the village and spaced out! 	
None	<div>None ▼</div> <div> </div>
0	

Damit ist die Konversation, die das Event aktivieren soll ausgeschrieben.  
Damit sich die Konversation aber auch auf das Event bezieht, muss ganz unten noch die entsprechende Quest ID eingestellt werden (hier: 4)

(Lässt man die ID auf 0 triggert die Konversation keine Quest und dient nur dem zuhören)

AudioFile	None  
SpeakerIndex	0
▼ Default	
Quest ID	4 

Damit ist das Konversationsobjekt fertiggestellt.

## **STEP 4: Einrichten der Welt NPCs zum triggern eines Events**

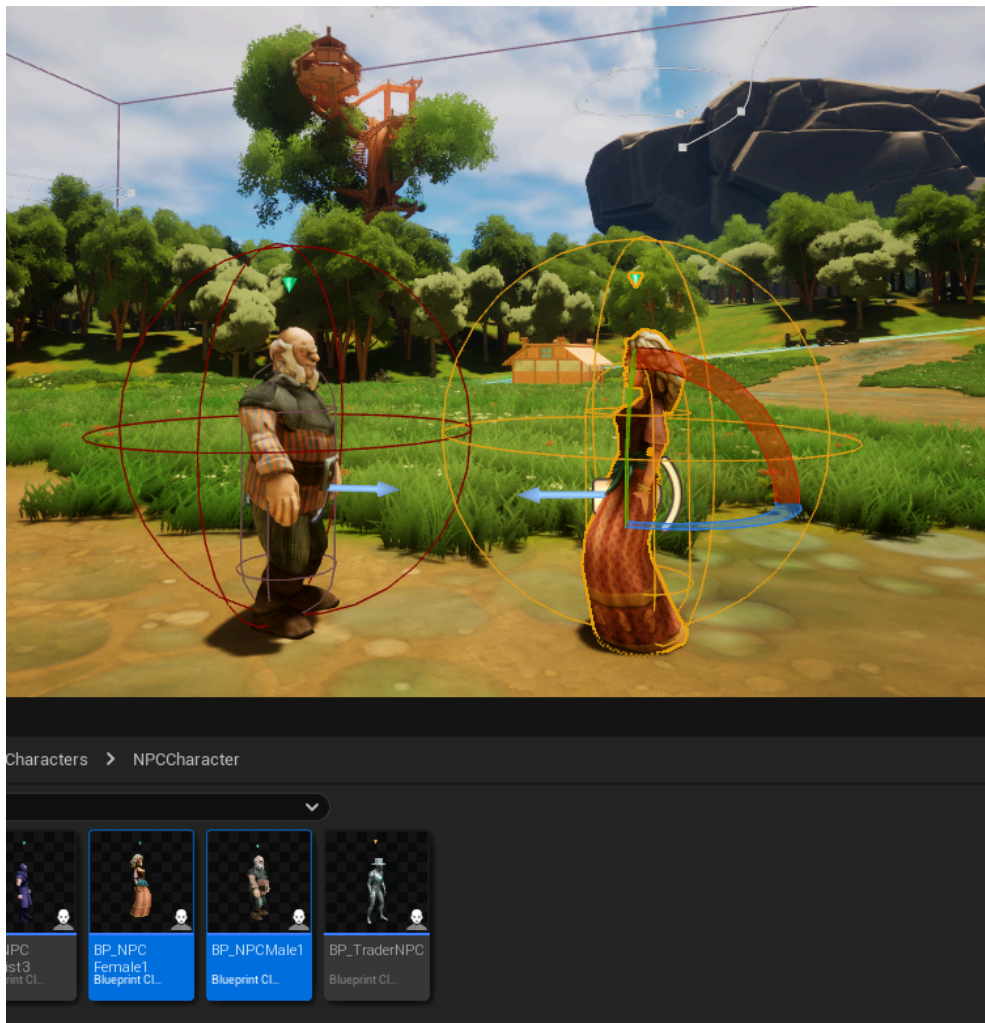
Jetzt können zwei NPCs in die Welt gezogen werden, die vom Spieler möglichst gefunden werden können und mit denen dieser dann interagieren kann.

NPCs sind zu finden unter...

### **Content/Blueprints/Characters/NPCCharacter**

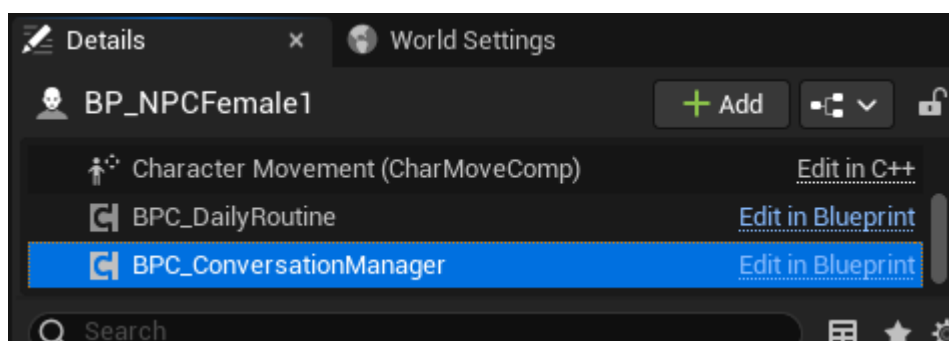
Am wahrscheinlichsten werden wir Dorfbewohner benutzen, um dem Spieler die Infos zu geben, die gebraucht werden.

Also ziehen wir für diese Beispiel einen BP\_NPCMale1 und eine BP\_NPCFemale1 nebeneinander.

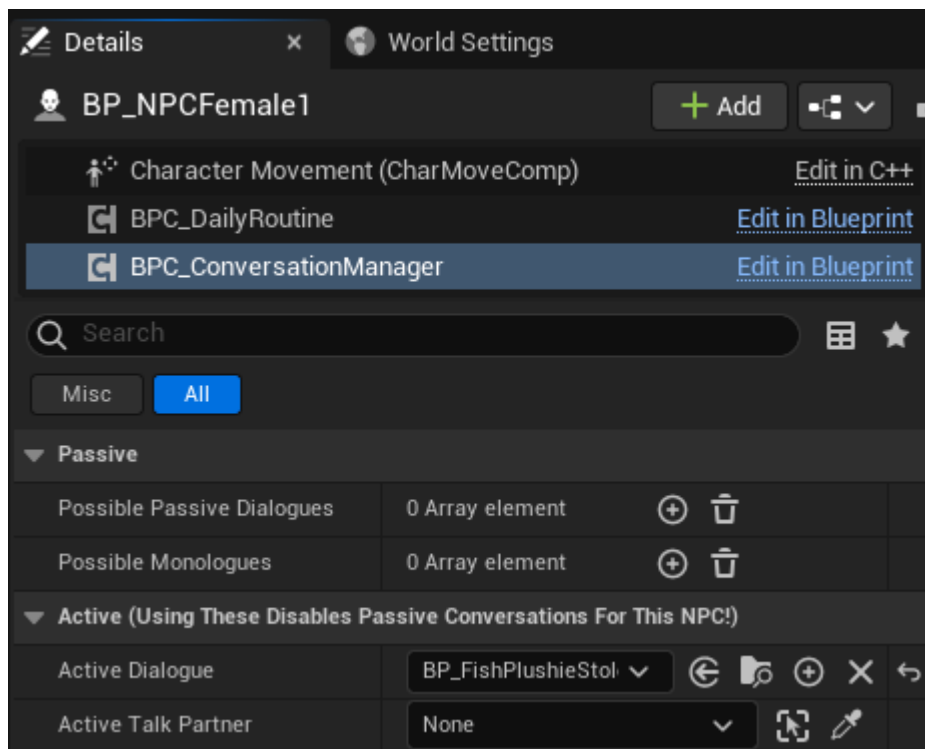


Um diese NPCs (sobald der Spieler mit ihnen interagiert) zum miteinander reden zu bringen, weisen wir einem von ihnen einen aktiven Dialog zu und setzen den anderen NPC als Gesprächspartner für diesen aktiven Dialog. Dafür nutzen wir die BPC\_ConversationManager-Komponente, die in jedem NPC vorhanden ist. Diese kann im Inspektor des NPCs ausgewählt werden.

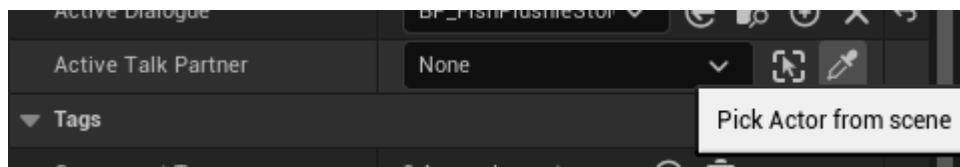
In diesem Beispiel weisen wir der young female NPC eine aktive Konversation zu, in dem wir sie im Viewport auswählen und die ConversationManager Komponente in der Blueprint-Hierarchie auswählen.



In den Eigenschaften klicken wir auf die Kategorie “All”.  
Dort sollte dann unter Anderem ein Drop-Down Menü für “Active Dialogue” auftauchen. In diesem können wir die zuvor geschriebene Konversation zuweisen.



Den “Active Talk Partner” können wir einfach mit der Pipette im Viewport auswählen.





Damit stehen die NPCs zueinander in Relation und werden ihr Gespräch beginnen, unabhängig davon mit welchem der beiden der Spieler interagiert.

Und sobald die Konversation im Spiel begonnen wird, aktiviert sich durch die im Konversationsobjekt eingetragene Quest ID auch das zugehörige Event.

## **STEP 5: Den Event-Ort einrichten**

Da Event-Orte zum Zeitpunkt, an dem sie noch nicht aktiviert sind, leerstehen sollen, müssen die Props und NPCs, die beim Event platziert werden mit bestimmten Komponenten versehen werden.

Dabei unterscheiden wir zwischen zwei Komponenten.

### **BP\_AssociateToQuest:**

Kümmert sich um erscheinen und verstecken von Actors, zum Zeitpunkt des triggern, sowie abschließen eines Events mit einstellbarer ID.

### **BP\_AssociateNPCToQuest:**

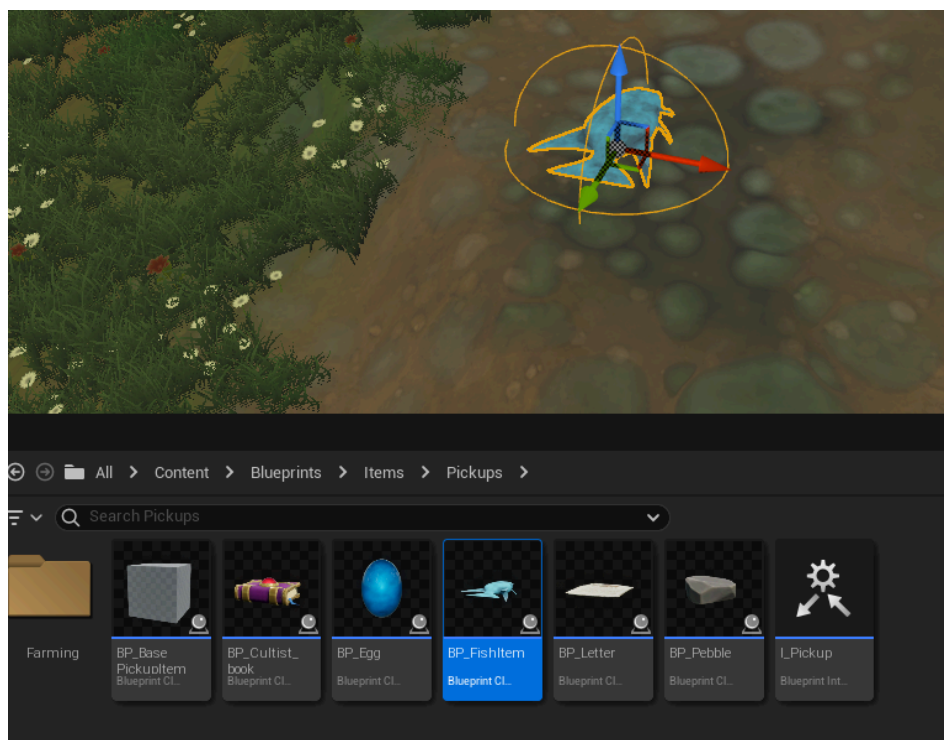
Kümmert sich um das zuweisen und entfernen bestimmter Konversationen von NPCs, wenn ein Event mit einstellbarer ID aktiviert oder abgeschlossen wird.

Beide Komponente sind zu finden unter...

**Content/Blueprints/Systems/QuestSystem/BP\_AssociateToQuest**  
**Content/Blueprints/Systems/QuestSystem/BP\_AssociateNPCToQuest**

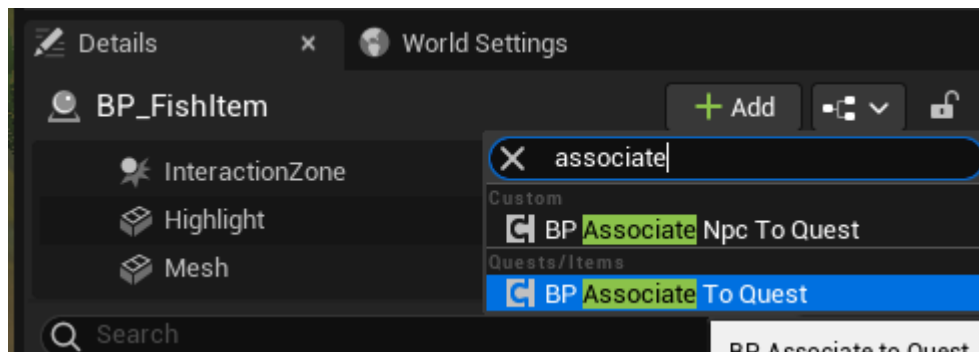
In unserem Beispiel möchten wir, dass das Fisch-Plüschtier im Nordausgang des Dorfes erscheint, sobald das Event aktiviert ist.

Dafür nehmen wir uns das Fisch-Item-Blueprint und platzieren es erstmal an einem beliebigen Ort im Level.





Im Inspektor können wir dem Actor eine neue Komponente hinzufügen (mit dem Add-Button). Als neue Komponente wählen wir **BP\_AssociateToQuest**.

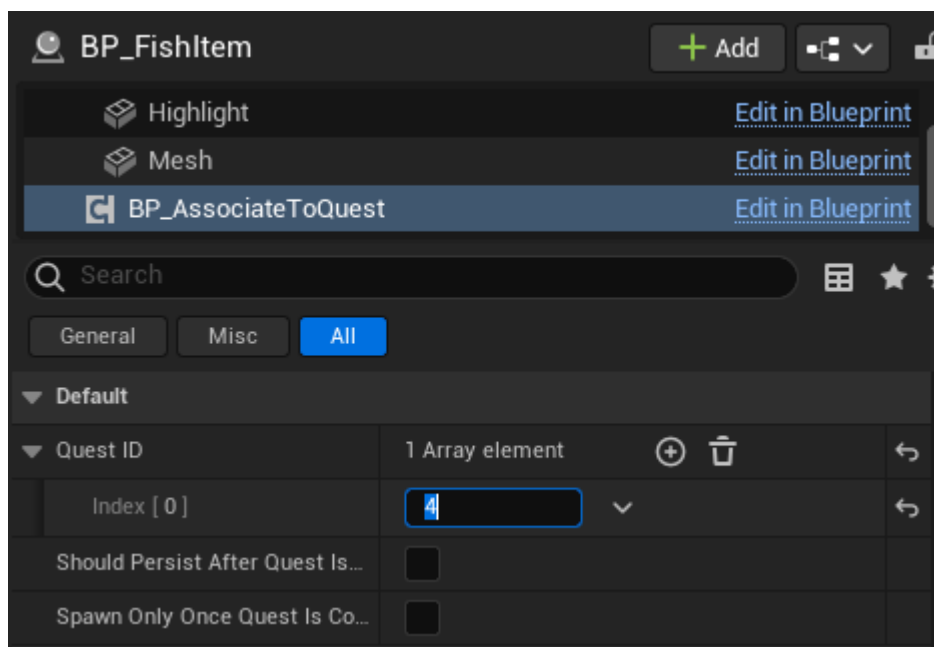


In der Komponente sind drei einstellbare Werte.

- Die zugehörige Quest-ID, auf die das Objekt reagieren soll
- Ob das Objekt nach Abschluss des zugehörigen Events sichtbar bleiben soll
- Ob das Objekt erst sichtbar gemacht werden soll, wenn das zugehörige Event abgeschlossen wurde

Stellt man nur eine Quest-ID ein, ist das **default-Verhalten**, dass der Actor **beim triggern des Events sichtbar** wird und **nach Abschluss des Events wieder verschwindet**.

Da das genau das Verhalten ist, das wir uns für das Fisch-Event wünschen, stellen wir nur die zugehörige Quest-ID für den Actor ein.



D.h. sobald der Spieler sich die Konversation der zuvor eingestellten Npcs angehört hat, wird der Fisch auftauchen und sobald der Spieler den Fisch auf der Karte markiert hat und die Karte abgibt, wird dieser beim erfolgreichen Test, ob der Marker richtig platziert wurde, verschwinden.

Diese Komponente kann für alle statischen Actors am Event-Ort, als auch für Npcs benutzt werden!! (z.B. Kultisten für das Shrine- und Lake-Event).

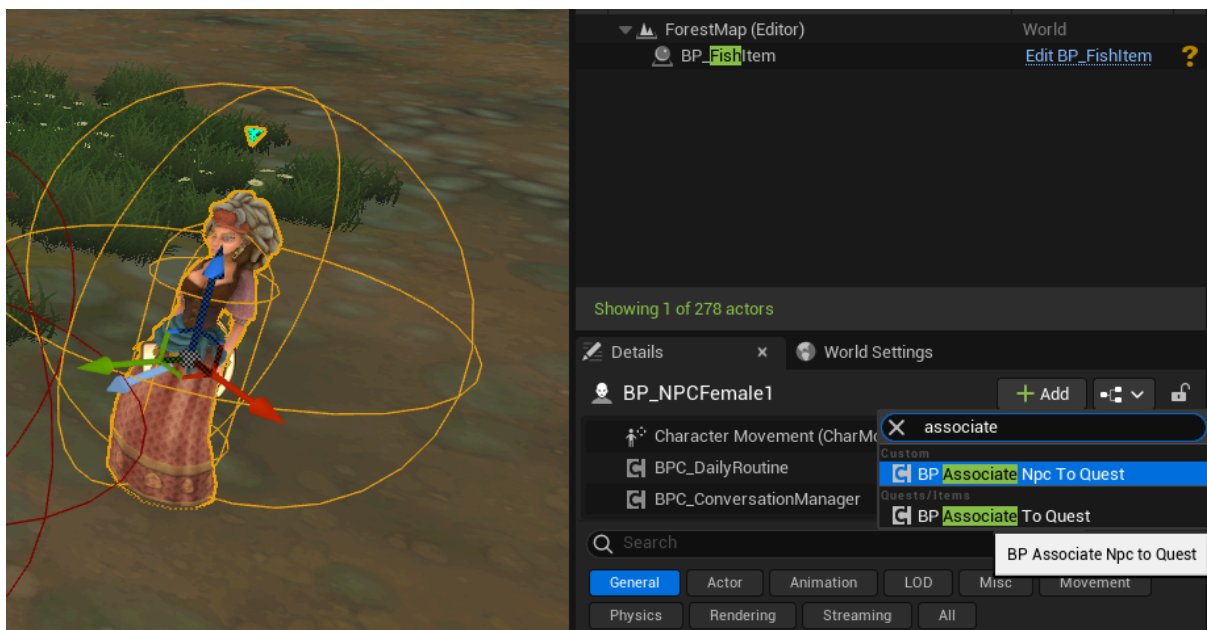
## STEP 6: Einrichten NPC-Reaktionen auf abgeschlossene Events

Es gibt zwei Möglichkeiten, Npcs auf Event-Abschlüsse reagieren zu lassen.

- 1.) **Bestimmte Npcs im Level** einzeln Konversationen zuweisen und entfernen, durch das **hinzufügen einer Komponente**, auf die gleich näher eingegangen wird.
- 2.) **Alle Npcs, die dem selben Blueprint entspringen** auf einmal Konversationen zuweisen und entfernen.

### Zu 1.):

Damit den Npcs die Konversation über den Fisch nach Abschluss des Events nicht mehr führen und stattdessen darüber reden, dass der Fisch gefunden wurde, müssen wir dem Npc, von dem aus die aktive Konversation eingestellt ist, eine **BP\_AssociateNPCToQuest** Komponente hinzufügen. Im Beispiel ist das hier die Peasant Girl.



Auch diese Komponente hat eine Quest-ID, die für das Event, auf das es sich bezieht einzustellen ist.

Die anderen einzustellenden Werte sind vom Namen her relativ selbsterklärend.

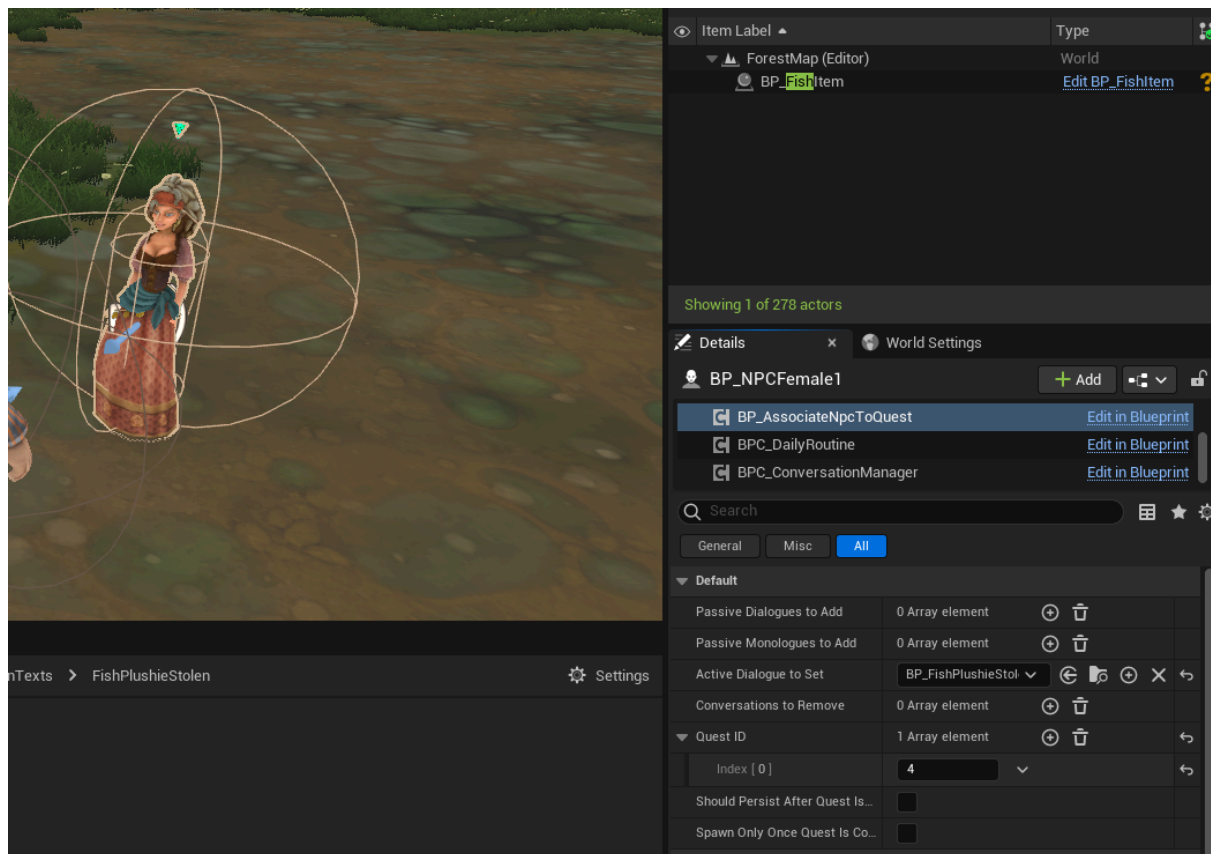
Es gilt weiterhin die gleiche Regel wie zuvor:

Npcs, denen aktive Dialoge zugewiesen werden, können zwar auch passive Dialoge zugewiesen bekommen, werden diese aber nie nutzen.

In diesem Beispiel muss dem Npcs nur ein neuer, aktiver Dialog zugewiesen werden, da ein Npc sowieso nur einen aktiven Dialog haben kann, und der alte damit einfach überschrieben wird.

Möchte man einem Npc mit passiven Konversationen solche bestimmten bei Abschluss eines Events wegnehmen, kann man das mit "Conversations To Remove" tun. Wir schreiben ein neues Konversations-Objekt BP\_FishPlushieStolenSUCCESS und weisen es BP\_NPCFemale1 mithilfe der Komponente zu.

Dadurch wird ihr alter, aktiver Dialog bei Abgabe der Map überschrieben.



## Zu 2.):

Möchte man allen Npcs, die aus dem selben Blueprint stammen, nach Abschluss eines Events Konversationen zuweisen oder entfernen, kann man dies in der **Quest-Datentabelle** tun.

Dafür wählen wir in der Datentabelle die zugehörige Quest aus und können dann unten im Row Editor unter "WorldNpcConversationModification" ein neues Element hinzufügen.

In dem neuen Element kann man unter "AffectedNpcs" eine Sorte von Npcs auswählen, denen die danach eingestellten Passiven Dialoge und Monologe hinzugefügt oder entfernt werden sollen.

Auf diese Weise sind **ALLE** Npcs der ausgewählten Sorte betroffen. Auch die, denen man die "BP\_AssociateNPCToQuest"-Komponente hinzugefügt hat.

Auf diese Weise kann man global allen Npcs Konversationen abnehmen, die mit dem Event zu tun hatten, so dass der Spieler keine redundanten Informationen zu Events mehr erhält, die bereits abgeschlossen wurden.

**Sobald all diese Sachen eingestellt sind, ist das Event fertig eingerichtet und kann im Level bespielt werden.**