



ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ  
ΒΙΝΤΕΟΠΑΙΧΝΙΔΙΟΥ ΣΤΗΝ ΜΗΧΑΝΗ  
UNREAL ENGINE 4

ΖΑΠΑΡΑΣ ΠΑΝΑΓΙΩΤΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΥΠΕΥΘΥΝΟΣ

Δαδαλιάρης Αντώνης  
Επικουρος Καθηγητής

ΣΥΝΕΠΙΒΛΕΠΩΝ

Λουκάκης Γεώργιος  
Λέκτορας Π.Δ. 407





ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ  
ΒΙΝΤΕΟΠΑΙΧΝΙΔΙΟΥ ΣΤΗΝ ΜΗΧΑΝΗ  
UNREAL ENGINE 4

ΖΑΠΑΡΑΣ ΠΑΝΑΓΙΩΤΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΥΠΕΥΘΥΝΟΣ

Δαδαλιάρης Αντώνης  
Επίκουρος Καθηγητής

ΣΥΝΕΠΙΒΛΕΠΩΝ

Λουκάκης Γεώργιος  
Λέκτορας Π.Δ. 407

Λαμία 2020







UNIVERSITY OF  
THESSALY

SCHOOL OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE & TELECOMMUNICATIONS

VIDEO GAME DESIGN AND  
IMPLEMENTATION ON THE UNREAL  
ENGINE 4

ZAPARAS PANAGIOTIS

FINAL THESIS

ADVISOR

Dadaliaris Antonios

Professor

CO ADVISOR

Loukakis Georgios

Lecturer

Lamia 2020

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Ολοκληρώνοντας την διπλωματική μου εργασίας θα ήθελα να ευχαριστήσω θερμά τους επιβλέποντες κ. Δαδαλιάρη Αντώνη και τον κ. Λουκάκη Γεώργιο για την καθοδήγηση τους στο συγκεκριμένο θέμα κατά την εκπόνηση της παρούσας εργασίας. Η εμπιστοσύνη τους, η ενθάρρυνση και οι υποδείξεις τους ήταν για μένα πολύτιμες.

Κλείνοντας, ευχαριστώ εγκάρδια την οικογένειά μου, που με στηρίζει και είναι δίπλα μου σε κάθε προσπάθειά μου εμπυχώνοντάς με ώστε να προσπαθώ για το καλύτερο και ιδιαίτερα τον αδερφό μου που είναι φίλος, συμπαραστάτης και...δοκιμαστής των παιχνιδιών που δημιουργώ.

«Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις <sup>(1)</sup>, που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

1. Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί **χωρίς να τα περικλείω σε εισαγωγικά** και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάσθηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.

2. Δέχομαι ότι η αυτολεξεί **παράθεση χωρίς εισαγωγικά**, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.

3. Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια

4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.

Ημερομηνία: 8/10/2020

Ο Δηλών

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.»



## ΠΕΡΙΛΗΨΗ

---

Το παιχνίδι αποτελούσε πάντα έμφυτη ανάγκη του ανθρώπου. Με το πέρασμα του χρόνου και λόγω της ραγδαίας εξέλιξης της τεχνολογίας, δημιουργήθηκαν τα βιντεοπαιχνίδια και έγιναν η δημοφιλέστερη μορφή παιχνιδιού. Εξαιτίας της πληθώρας και της ποικιλίας τους κατατάσσονται σε διάφορες κατηγορίες, ενώ την ανάπτυξη και την υλοποίησή τους διευκολύνουν τα λογισμικά που ονομάζονται «μηχανές ανάπτυξης βιντεοπαιχνιδιών». Στο πλαίσιο της παρούσας πτυχιακής εργασίας, μετά από μια σύντομη αναφορά σε κάποιες από τις μηχανές αυτές, γίνεται μια ιστορική ανασκόπηση της μηχανής Unreal, ενώ έμφαση δίνεται στην μηχανή Unreal4. Ο αναγνώστης αρχικά πληροφορείται για τα στάδια σχεδιασμού και ανάπτυξης ενός βιντεοπαιχνιδιού και ακολούθως μυείται στην ανάλυση ανάπτυξης του βιντεοπαιχνιδιού The Challenger, το οποίο δημιουργήθηκε με την βοήθεια της μηχανής Unreal 4 για τις ανάγκες της εργασίας αυτής.



## ABSTRACT

---

With the passage of time and technology video games were created and soon became the most popular form of gaming. Due to their wide variety and number they are classified into different categories, while softwares were created to help developers with the development and production. These tools are called game engines. Within the bounds of this paper, after a short reference to some of these game engines and historical overview of unreal engine, we will focus on unreal engine 4. The reader after a short reference to the development stages of video games, is run through the development of the game The Challenger, which was created using unreal engine 4 within the scope of this paper.

## Table of Contents

---

ΠΕΡΙΛΗΨΗ .....	IX
ABSTRACT .....	XI
TABLE OF CONTENTS.....	1
TABLE OF PICTURES.....	5
<b><u>ΚΕΦΑΛΑΙΟ 1 ΕΙΣΑΓΩΓΗ.....</u></b>	<b><u>18</u></b>
1.1 ΠΑΙΧΝΙΔΙΑ .....	18
1.2 ΑΝΑΔΡΟΜΗ ΣΤΗΝ ΙΣΤΟΡΙΑ ΤΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΠΑΙΧΝΙΔΙΩΝ.....	19
1.3 ΕΙΔΗ ΨΗΦΙΑΚΩΝ ΠΑΙΧΝΙΔΙΩΝ .....	20
<b><u>ΚΕΦΑΛΑΙΟ 2 ΜΗΧΑΝΕΣ ΔΗΜΙΟΥΡΓΙΑΣ ΒΙΝΤΕΟΠΑΙΧΝΙΔΙΩΝ .....</u></b>	<b><u>22</u></b>
2.1 ΕΙΣΑΓΩΓΗ.....	22
2.2 UNITY .....	23
2.3 GAMEMAKER .....	23
2.4 TURBULENZGAME.....	23
2.5 GODOT.....	23
<b><u>ΚΕΦΑΛΑΙΟ 3 ΙΣΤΟΡΙΚΗ ΑΝΑΣΚΟΠΗΣΗ ΤΗΣ UNREAL ENGINE.....</u></b>	<b><u>25</u></b>
3.1 UNREAL ENGINE 1.....	25
3.2 UNREAL ENGINE 2 .....	26
3.3 UNREAL ENGINE 3 .....	26
3.4 UNREAL ENGINE 4 .....	26
3.5 ΤΟ ΜΕΛΛΟΝ ΤΗΣ UNREAL ENGINE.....	27
<b><u>ΚΕΦ. 4 ΣΤΑΔΙΑ ΣΧΕΔΙΑΣΜΟΥ ΚΑΙ ΥΛΟΠΟΙΗΣΗΣ ΒΙΝΤΕΟΠΑΙΧΝΙΔΙΩΝ</u></b>	<b><u>29</u></b>



<b>4.1 ΣΧΕΔΙΑΣΜΟΣ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ .....</b>	<b>29</b>
<b>4.2 ΣΤΑΔΙΑ ΑΝΑΠΤΥΞΗΣ ΕΜΠΟΡΙΚΩΝ ΒΙΝΤΕΟΠΑΙΧΝΙΔΙΩΝ.....</b>	<b>29</b>
<b>4.3 ΣΤΑΔΙΑ ΑΝΑΠΤΥΞΗΣ ΤΟΥ «THE CHALLENGER».....</b>	<b>32</b>
<b>4.4 ΓΕΝΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΟΥ THE CHALLENGER .....</b>	<b>32</b>

## **ΚΕΦΑΛΑΙΟ 5 Η ΔΟΜΗ ΤΟΥ THE CHALLENGER..... 35**

<b>5.1 BLUEPRINTS .....</b>	<b>35</b>
ΕΝΟΤΗΤΑ 5.1.1 BP_FUTURESHOT ΚΑΙ ΠΑΙΔΙΑ.....	35
ΕΝΟΤΗΤΑ 5.1.2 BP_BASICLAMP .....	42
ΕΝΟΤΗΤΑ 5.1.3 BP_BLOCKSHIELDBLOCKVISION .....	44
ΕΝΟΤΗΤΑ 5.1.4 BP_BLOCKSHIELDSIMPLE.....	46
ΕΝΟΤΗΤΑ 5.1.5 BP_DOORFIELD.....	49
ΕΝΟΤΗΤΑ 5.1.6 BP_CANDY .....	52
ΕΝΟΤΗΤΑ 5.1.7 BP_CANDYDISIN .....	57
ΕΝΟΤΗΤΑ 5.1.8 BP_DISAPPEARINGPLAT .....	61
ΕΝΟΤΗΤΑ 5.1.9 BP_EVILONE.....	66
ΕΝΟΤΗΤΑ 5.1.10 BP_FORCEWALL.....	69
ΕΝΟΤΗΤΑ 5.1.11 BP_HOOKBEAM.....	73
ΕΝΟΤΗΤΑ 5.1.12 BP_HOOKPOINT .....	75
ΕΝΟΤΗΤΑ 5.1.13 BP_LAUNCHER.....	79
ΕΝΟΤΗΤΑ 5.1.14 BP_LETTEREXPOSAL.....	83
ΕΝΟΤΗΤΑ 5.1.15 BP_PLATFORMMOV .....	88
ΕΝΟΤΗΤΑ 5.1.16 BP_PORTALNOSOUND.....	92
ΕΝΟΤΗΤΑ 5.1.17 BP_TELEPORTINDICATOR.....	96
ΕΝΟΤΗΤΑ 5.1.18 BP_SPIKEMOVE.....	97
ΕΝΟΤΗΤΑ 5.1.19 BP_SPIKE.....	100
ΕΝΟΤΗΤΑ 5.1.20 BP_SPAWNPOINT.....	105
ΕΝΟΤΗΤΑ 5.1.21 BP_WAVESPAWNCONTROLLER.....	114
ΕΝΟΤΗΤΑ 5.1.22 BP_AIBATCHSPAWNER.....	126
<b>5.2 INTERFACE.....</b>	<b>127</b>
<b>5.3 ACTOR COMPONENT .....</b>	<b>128</b>

ENOTHTA 5.3.1 AC_AIUNITDELETIONCONTROLLER.....	128
ENOTHTA 5.3.2 AC_DASHDIRECTIONAL.....	129
ENOTHTA 5.3.3 AC_FORCEWALL.....	134
ENOTHTA 5.3.4 AC_TELEPORTFINAL.....	136
<b>5.4 FIRST PERSON CHARACTER.....</b>	<b>141</b>
ENOTHTA 5.4.1 FIRSTPERSONCHARACTER.....	141
ENOTHTA 5.4.2 FIRSTPERSONGAMEMODE.....	157
ENOTHTA 5.4.3 FIRSTPERSONHUD.....	161
<b>5.5 WIDGET.....</b>	<b>163</b>
ENOTHTA 5.5.1 DEATH_WIDGET.....	163
ENOTHTA 5.5.2 INSTRUCTIONS_WIDGET.....	165
ENOTHTA 5.5.3 LEVELSELECTION_WIDGET.....	166
ENOTHTA 5.5.4 MAINMENU_WIDGET.....	167
ENOTHTA 5.5.5 PASSWORD_WIDGET.....	169
ENOTHTA 5.5.6 PAUSEMENU_WIDGET.....	171
ENOTHTA 5.5.7 STORY_WIDGET.....	172
ENOTHTA 5.5.8 UMG_HUD.....	174
ENOTHTA 5.5.9 WAVE_WIDGET.....	177
<b>5.6 TEXTURES.....</b>	<b>179</b>
<b>5.7 MATERIALS.....</b>	<b>181</b>
ENOTHTA 5.7.1 MAT_SIMPLERED.....	181
ENOTHTA 5.7.2 MAT_CANDY.....	183
ENOTHTA 5.7.3 MAT_SIMPLEDISINEGRATIONE.....	183
ENOTHTA 5.7.4 MAT_METALPLATE.....	184
ENOTHTA 5.7.5 MAT_SIMPLEGLOW.....	185
ENOTHTA 5.7.6 MAT_SHIELDFINAL.....	185
ENOTHTA 5.7.7 MAT_TELEPORT.....	188
ENOTHTA 5.7.8 MAT_ORB.....	188
<b>5.8 AUDIO.....</b>	<b>191</b>
<b>5.9 MAPS.....</b>	<b>192</b>
ENOTHTA 5.9.1 MAINMENU.....	192
ENOTHTA 5.9.2 LEVEL1STORY.....	194

ΕΝΟΤΗΤΑ 5.9.3 LEVEL1.....	196
ΕΝΟΤΗΤΑ 5.9.4 LEVEL2.....	218
ΕΝΟΤΗΤΑ 5.9.5 LEVEL6.....	227
ΕΝΟΤΗΤΑ 5.9.6 LEVEL4.....	238
<b><u>ΚΕΦΑΛΑΙΟ 6 ΣΥΜΠΕΡΑΣΜΑΤΑ.....</u></b>	<b>244</b>
<b><u>ΒΙΒΛΙΟΓΡΑΦΙΑ.....</u></b>	<b>245</b>
<b><u>ΠΑΡΑΡΤΗΜΑΤΑ.....</u></b>	<b>248</b>
ΠΑΡΑΡΤΗΜΑ Α - MATERIALS .....	249
ΠΑΡΑΡΤΗΜΑ Β - GEOMETRY .....	255

## Table of pictures

---

Εικόνα 1 Απεικόνιση σταδίων παραγωγής βιντεοπαιχνιδιού .....	31
Εικόνα 2 Στιγμιότυπο παιχνιδιού στο 1ο επίπεδο .....	33
Εικόνα 3 Στιγμιότυπο παιχνιδιού στο 2ο επίπεδο .....	33
Εικόνα 4 Στιγμιότυπο παιχνιδιού στο 3ο επίπεδο .....	34
Εικόνα 5 Στιγμιότυπο παιχνιδιού στο 4ο επίπεδο .....	34
Εικόνα 6 BP_FutureShot part1 .....	36
Εικόνα 7 BP_FutureShot part 2 .....	37
Εικόνα 8 BP_FutureShot part3 .....	37
Εικόνα 9 BP_FutureShot part 4 .....	38
Εικόνα 10 BP_FutureShot part 5 .....	38
Εικόνα 11 BP_FutureShot part 6 .....	39
Εικόνα 12 BP_FutureShot part7 .....	39
Εικόνα 13 BP_FutureShot part 8 .....	40
Εικόνα 14 BP_FutureShot part 9 .....	40
Εικόνα 15 BP_FutureShot part 10.....	41
Εικόνα 16 BP_AlliedShot part 11 .....	41
Εικόνα 17 BP_EnemyShot V 12.....	41
Εικόνα 18 BP_BasicLamp part 1 .....	42
Εικόνα 19 BP_BasicLamp part 2 .....	42
Εικόνα 20 BP_BasicLamp part 3 .....	43
Εικόνα 21 BP_BlockShieldBlockVision part 1.....	44
Εικόνα 22 BP_BlockShieldBlockVision part 2.....	45
Εικόνα 23 BP_BlockShieldBlockVision part 3.....	46
Εικόνα 24 BP_BlockShieldBlockVision part 4.....	46
Εικόνα 25 BP_BlockShieldSimple part 1.....	47
Εικόνα 26 BP_BlockShieldSimple part 2.....	48
Εικόνα 27 BP_BlockShieldSimple part 3.....	48
Εικόνα 28 BP_DoorField part1 .....	49
Εικόνα 29 BP_DoorField part 2 .....	50
Εικόνα 30 BP_DoorField part 3 .....	50

Εικόνα 31 BP_DoorField part 4 .....	51
Εικόνα 32 BP_DoorField part 5 .....	51
Εικόνα 33 BP_DoorField part 6 .....	52
Εικόνα 34 BP_DoorField part 7 .....	52
Εικόνα 35 BP_Candy part 1 .....	53
Εικόνα 36 BP_Candy part 2.....	54
Εικόνα 37 BP_Candy part 3.....	55
Εικόνα 38 BP_Candy part 4.....	56
Εικόνα 39 BP_Candy part 5.....	56
Εικόνα 40 BP_CandyDisin part 1 .....	57
Εικόνα 41 BP_CandyDisin part 2.....	58
Εικόνα 42 BP_CandyDisin part 3 .....	59
Εικόνα 43 BP_CandyDisin part 4.....	59
Εικόνα 44 BP_CandyDisin part 5 .....	60
Εικόνα 45 BP_CandyDisin part 6 .....	60
Εικόνα 46 BP_DisappearingPlat part 1.....	61
Εικόνα 47 BP_DisappearingPlat part 2.....	62
Εικόνα 48 BP_DisappearingPlat part 3.....	63
Εικόνα 49 BP_DisappearingPlat part 4.....	63
Εικόνα 50 BP_DisappearingPlat part 5.....	64
Εικόνα 51 BP_DisappearingPlat part6.....	64
Εικόνα 52 BP_DisappearingPlat part 7.....	65
Εικόνα 53 BP_DisappearingPlat part 8.....	65
Εικόνα 54 BP_DisappearingPlat part 9.....	65
Εικόνα 55 BP_DisappearingPlat part 10.....	66
Εικόνα 56 BP_EvilOne part1 .....	66
Εικόνα 57 BP_EvilOne part2 .....	67
Εικόνα 58 BP_EvilOne part3 .....	68
Εικόνα 59 BP_EvilOne part4 .....	68
Εικόνα 60 BP_ForceWall part 1.....	69
Εικόνα 61 BP_ForceWall part 2.....	70
Εικόνα 62 BP_ForceWall part 3.....	71

Εικόνα 63 BP_ForceWall part 4.....	71
Εικόνα 64 BP_ForceWall part 5.....	72
Εικόνα 65 BP_ForceWall part 6.....	73
Εικόνα 66 BP_HookBeam part 1 .....	73
Εικόνα 67 BP_HookBeam part 2 .....	74
Εικόνα 68 BP_HookBeam part 3 .....	75
Εικόνα 69 BP_HookPoint part 1 .....	75
Εικόνα 70 BP_HookPoint part 2 .....	76
Εικόνα 71 BP_HookPoint part 3 .....	77
Εικόνα 72 BP_HookPoint part 4 .....	77
Εικόνα 73 BP_HookPoint part 5 .....	78
Εικόνα 74 BP_HookPoint part 6 .....	78
Εικόνα 75 BP_Launcher part 1.....	79
Εικόνα 76 BP_Launcher part 2.....	80
Εικόνα 77 BP_Launcher part 3.....	80
Εικόνα 78 BP_Launcher part 4.....	81
Εικόνα 79 BP_Launcher part 5.....	82
Εικόνα 80 BP_Launcher part 6.....	83
Εικόνα 81 BP_LetterExposal part 1 .....	83
Εικόνα 82 BP_LetterExposal part 2 .....	84
Εικόνα 83 BP_LetterExposal part 3 .....	85
Εικόνα 84 BP_LetterExposal part 4 .....	85
Εικόνα 85 BP_LetterExposal part 5 .....	86
Εικόνα 86 BP_LetterExposal part 6 .....	86
Εικόνα 87 BP_LetterExposal part 7 .....	87
Εικόνα 88 BP_LetterExposal part 8 .....	87
Εικόνα 89 BP_LetterExposal part 9 .....	88
Εικόνα 90 BP_PlatformMov part 1 .....	88
Εικόνα 91 BP_PlatformMov part 2.....	89
Εικόνα 92 BP_PlatformMov part 3.....	90
Εικόνα 93 BP_PlatformMov part 4.....	90
Εικόνα 94 BP_PlatformMov part 5.....	90

Εικόνα 95 BP_PlatformMov part 6.....	91
Εικόνα 96 BP_PlatformMov part 7.....	91
Εικόνα 97 BP_PlatformMov part 8.....	92
Εικόνα 98 BP_PlatformMov part 9.....	92
Εικόνα 99 BP_PortalNoSound part 1.....	93
Εικόνα 100 BP_PortalNoSound part 2.....	93
Εικόνα 101 BP_PortalNoSound part 3.....	94
Εικόνα 102 BP_PortalNoSound part 4.....	95
Εικόνα 103 BP_PortalNoSound part 5.....	95
Εικόνα 104 BP_PortalNoSound part 6.....	96
Εικόνα 105 BP_TeleportIndicator part 1.....	96
Εικόνα 106 BP_TeleportIndicator part 2.....	97
Εικόνα 107 BP_SpikeMove part 1.....	98
Εικόνα 108 BP_SpikeMove part 2.....	98
Εικόνα 109 BP_SpikeMove part 3.....	99
Εικόνα 110 BP_SpikeMove part 4.....	99
Εικόνα 111 BP_SpikeMove part 5.....	100
Εικόνα 112 BP_Spike part 1.....	101
Εικόνα 113 BP_Spike part 2.....	101
Εικόνα 114 BP_Spike part 3.....	102
Εικόνα 115 BP_Spike part 4.....	102
Εικόνα 116 BP_Spike part 5.....	103
Εικόνα 117 BP_SpikeHalf.....	103
Εικόνα 118 BP_SpikeSolo.....	104
Εικόνα 119 BP_SpikeYellow part 1.....	104
Εικόνα 120 BP_SpikeYellow part 2.....	105
Εικόνα 121 BP_SpikeYellowHalf.....	105
Εικόνα 122 BP_SpawnPoint part 1.....	106
Εικόνα 123 BP_SpawnPoint part 2.....	106
Εικόνα 124 BP_SpawnPoint part 3.....	107
Εικόνα 125 BP_SpawnPoint_FixedLocation part 1.....	107
Εικόνα 126 BP_SpawnPoint_FixedLocation part 2.....	108

Εικόνα 127 BP_SpawnPoint_FixedLocation part3.....	108
Εικόνα 128 BP_SpawnPoint_FixedLocation part4.....	108
Εικόνα 129 BP_SpawnPoint_Spawning part 1.....	109
Εικόνα 130 BP_SpawnPoint_Spawning part 2.....	109
Εικόνα 131 BP_SpawnPoint_Spawning part 3.....	110
Εικόνα 132 BP_SpawnPoint_Spawning part 4.....	110
Εικόνα 133 BP_SpawnPoint_Spawning part 5.....	111
Εικόνα 134 BP_SpawnPoint_Spawning part 6.....	111
Εικόνα 135 BP_SpawnPoint_Spawning part 7.....	112
Εικόνα 136 BP_SpawnPoint_Spawning part 8.....	112
Εικόνα 137 BP_SpawnPoint_Spawning part 9.....	113
Εικόνα 138 BP_SpawnPoint_Spawning part 10.....	113
Εικόνα 139 BP_SpawnPoint_Spawning part 11.....	113
Εικόνα 140 BP_WaveSpawnController part 1.....	114
Εικόνα 141 BP_WaveSpawnController part 2.....	115
Εικόνα 142 BP_WaveSpawnController part 3.....	115
Εικόνα 143 BP_WaveSpawnController part 4.....	115
Εικόνα 144 BP_WaveSpawnController part 5.....	116
Εικόνα 145 BP_WaveSpawnController part 6.....	116
Εικόνα 146 BP_WaveSpawnController part 7.....	117
Εικόνα 147 BP_WaveSpawnController part 8.....	117
Εικόνα 148 BP_WaveSpawnController part 9.....	117
Εικόνα 149 BP_WaveSpawnController part 10.....	118
Εικόνα 150 BP_WaveSpawnController part 11.....	118
Εικόνα 151 BP_WaveSpawnController part 12.....	119
Εικόνα 152 BP_WaveSpawnController part 13.....	119
Εικόνα 153 BP_WaveSpawnController part 14.....	119
Εικόνα 154 BP_WaveSpawnController part 15.....	120
Εικόνα 155 BP_WaveSpawnController part 16.....	120
Εικόνα 156 BP_WaveSpawnController part 17.....	121
Εικόνα 157 BP_WaveSpawnController part 18.....	121
Εικόνα 158 BP_WaveSpawnController part 19.....	121



Εικόνα 159 BP_WaveSpawnController part 20.....	122
Εικόνα 160 BP_WaveSpawnController part 21.....	122
Εικόνα 161 BP_WaveSpawnController part 22.....	122
Εικόνα 162 BP_WaveSpawnController part 23.....	123
Εικόνα 163 BP_WaveSpawnController part 24.....	123
Εικόνα 164 BP_WaveSpawnController part 25.....	123
Εικόνα 165 BP_WaveSpawnController part 26.....	124
Εικόνα 166 BP_WaveSpawnController part 27.....	124
Εικόνα 167 BP_WaveSpawnController part28.....	124
Εικόνα 168 BP_WaveSpawnController part 29.....	125
Εικόνα 169 BP_WaveSpawnController part 30.....	125
Εικόνα 170 BP_WaveSpawnController part 31.....	125
Εικόνα 171 BP_AIBatchSpawner part 1 .....	126
Εικόνα 172 BP_AIBatchSpawner part2 .....	127
Εικόνα 173 BPI_WaveCycleStatsController.....	127
Εικόνα 174 BPI_WaveInfoDisplay.....	128
Εικόνα 175 AC_AIUnitDeletionController part 1 .....	128
Εικόνα 176 AC_AIUnitDeletionController part 2 .....	129
Εικόνα 177 AC_DashDirectional part 1.....	129
Εικόνα 178 AC_DashDirectional part 2.....	130
Εικόνα 179 AC_DashDirectional part 3.....	130
Εικόνα 180 AC_DashDirectional part 4.....	131
Εικόνα 181 AC_DashDirectional part 5.....	131
Εικόνα 182 AC_DashDirectional part 6.....	132
Εικόνα 183 AC_DashDirectional part 7.....	132
Εικόνα 184 AC_DashDirectional part 8.....	133
Εικόνα 185 AC_DashDirectional part 9.....	133
Εικόνα 186 AC_ForceWall part 1.....	134
Εικόνα 187 AC_ForceWall part 2.....	135
Εικόνα 188 AC_ForceWall part 3.....	135
Εικόνα 189 AC_ForceWall part 4.....	135
Εικόνα 190 AC_ForceWall part 5.....	136

Εικόνα 191 AC_TeleportFinal part 1.....	136
Εικόνα 192 AC_TeleportFinal part 2.....	137
Εικόνα 193 AC_TeleportFinal part 3.....	137
Εικόνα 194 AC_TeleportFinal part 4.....	138
Εικόνα 195 AC_TeleportFinal part 5.....	138
Εικόνα 196 AC_TeleportFinal part 6.....	139
Εικόνα 197 AC_TeleportFinal part 7.....	139
Εικόνα 198 AC_TeleportFinal part 8.....	140
Εικόνα 199 FirstPersonCharacter part 1 .....	141
Εικόνα 200 FirstPersonCharacter part 2 .....	141
Εικόνα 201 FirstPersonCharacter part 3 .....	142
Εικόνα 202 FirstPersonCharacter part 4 .....	142
Εικόνα 203 FirstPersonCharacter part 5 .....	143
Εικόνα 204 FirstPersonCharacter part 6 .....	144
Εικόνα 205 FirstPersonCharacter part 7 .....	144
Εικόνα 206 FirstPersonCharacter part 8 .....	145
Εικόνα 207 FirstPersonCharacter part 9 .....	146
Εικόνα 208 FirstPersonCharacter part 10 .....	146
Εικόνα 209 FirstPersonCharacter part 11 .....	146
Εικόνα 210 FirstPersonCharacter part 12 .....	148
Εικόνα 211 FirstPersonCharacter part 13 .....	148
Εικόνα 212 FirstPersonCharacter part 14 .....	149
Εικόνα 213 FirstPersonCharacter part 15 .....	149
Εικόνα 214 FirstPersonCharacter part 16 .....	150
Εικόνα 215 FirstPersonCharacter part 17 .....	150
Εικόνα 216 FirstPersonCharacter part 18 .....	151
Εικόνα 217 FirstPersonCharacter part 19 .....	151
Εικόνα 218 FirstPersonCharacter part 20 .....	152
Εικόνα 219 FirstPersonCharacter part 21 .....	152
Εικόνα 220 FirstPersonCharacter part 22 .....	153
Εικόνα 221 FirstPersonCharacter part 23 .....	154
Εικόνα 222 FirstPersonCharacter part 24 .....	154

Εικόνα 223 FirstPersonCharacter part 25 .....	155
Εικόνα 224 FirstPersonCharacter part 26 .....	155
Εικόνα 225 FirstPersonCharacter part 27 .....	156
Εικόνα 226 FirstPersonCharacter part 28 .....	156
Εικόνα 227 FirstPersonCharacter part 29 .....	156
Εικόνα 228 FirstPersonCharacter part 30 .....	157
Εικόνα 229 FirstPersonCharacter part 31 .....	157
Εικόνα 230 FirstPersonGameMode part 1 .....	158
Εικόνα 231 FirstPersonGameMode part 2 .....	158
Εικόνα 232 FirstPersonGameMode part 3 .....	159
Εικόνα 233 FirstPersonGameMode part 4 .....	159
Εικόνα 234 FirstPersonGameMode part 5 .....	160
Εικόνα 235 FirstPersonGameMode part 6 .....	160
Εικόνα 236 FirstPersonGameMode part 7 .....	160
Εικόνα 237 FirstPersonHUD part 1 .....	161
Εικόνα 238 FirstPersonHUD part 2 .....	162
Εικόνα 239 FirstPersonHUD part 3 .....	162
Εικόνα 240 FirstPersonHUD part 4 .....	163
Εικόνα 241 Death_Widget1.....	164
Εικόνα 242 Death_Widget2.....	164
Εικόνα 243 Instructions_Widget1.....	165
Εικόνα 244 Instructions_Widget2.....	165
Εικόνα 245 LevelSelection_Widget1 .....	166
Εικόνα 246 LevelSelection_Widget2 .....	167
Εικόνα 247 MainMenu_Widget1.....	168
Εικόνα 248 MainMenu_Widget3.....	169
Εικόνα 249 Password_Widget1 .....	170
Εικόνα 250 Password_Widget2.....	170
Εικόνα 251 PauseMenu_Widget1 .....	171
Εικόνα 252 PauseMenu_Widget2 .....	172
Εικόνα 253 Story_Widget1.....	172
Εικόνα 254 Story_Widget2.....	173

Εικόνα 255 Story_Widget3.....	173
Εικόνα 256 UMG_HUD1.....	174
Εικόνα 257 UMG_HUD2.....	174
Εικόνα 258 UMG_HUD3.....	175
Εικόνα 259 UMG_HUD4.....	175
Εικόνα 260 UMG_HUD5.....	175
Εικόνα 261 UMG_HUD6.....	176
Εικόνα 262 UMG_HUD7.....	177
Εικόνα 263 Wave_Widget1 .....	177
Εικόνα 264 Wave_Widget2 .....	178
Εικόνα 265 Υφές Candy .....	179
Εικόνα 266 Υφές MetalPlates005 .....	179
Εικόνα 267 Υφές MetalPlates006 .....	180
Εικόνα 268 Υφή T_GridCell .....	180
Εικόνα 269 Υφή T_ScanLineMask.....	181
Εικόνα 270 T_TilingNoise03.....	181
Εικόνα 271 Mat_SimpleRed.....	182
Εικόνα 272 Mat_Candy.....	183
Εικόνα 273 Mat_SimpleDisinegrationE .....	184
Εικόνα 274 Mat_MetalPlate .....	184
Εικόνα 275 Mat_SimpleGlow.....	185
Εικόνα 276 Mat_ShieldFinal part1.....	186
Εικόνα 277 Mat_ShieldFinal part 2.....	186
Εικόνα 278 Mat_ShieldFinal part 3.....	187
Εικόνα 279 Mat_ShieldFinal part 4.....	187
Εικόνα 280 Mat_Teleport.....	188
Εικόνα 281 Mat_Orb part 1 .....	189
Εικόνα 282 Mat_Orb part 2 .....	189
Εικόνα 283 Mat_Orb part 3 .....	190
Εικόνα 284 Mat_Orb part 4 .....	190
Εικόνα 285 Mat_Orb part 5 .....	191
Εικόνα 286 MainMenu World Settings.....	192

Εικόνα 287 MainMenu My Blueprint Tab.....	193
Εικόνα 288 MainMenu Level Blueprint .....	193
Εικόνα 289 Level1Story World Settings.....	194
Εικόνα 290 Level1Story My Blueprint Tab.....	195
Εικόνα 291 Level1Story Level Blueprint.....	195
Εικόνα 292 Level1 World Settings.....	196
Εικόνα 293 Level1 My Blueprint Tab.....	197
Εικόνα 294 Level1 Level Blueprint part 1.....	198
Εικόνα 295 Level1 Level Blueprint part 2.....	199
Εικόνα 296 UnBlur.....	199
Εικόνα 297 Level1 Level Blueprint part 3.....	200
Εικόνα 298 Level1 Level Blueprint part 4.....	200
Εικόνα 299 Level1 DashTrigger .....	200
Εικόνα 300 Level1 Level Blueprint part 5.....	201
Εικόνα 301 Level1 Level Blueprint part 6.....	201
Εικόνα 302 OrbGivingTrigger.....	202
Εικόνα 303 Level1 Level Blueprint part 7.....	202
Εικόνα 304 Level1 Level Blueprint part 8.....	202
Εικόνα 305 Level1 Level Blueprint part 9.....	203
Εικόνα 306 Level1 Level Blueprint part 10.....	203
Εικόνα 307 Level1 Level Blueprint part 11.....	203
Εικόνα 308 SetSpawnSecond.....	204
Εικόνα 309 Level1 Level Blueprint part 12.....	204
Εικόνα 310 HookLandingTarget.....	205
Εικόνα 311 Level1 Level Blueprint part 13.....	205
Εικόνα 312 Level1 Level Blueprint part 14.....	205
Εικόνα 313 Level1 Level Blueprint part 15.....	206
Εικόνα 314 Level1 Level Blueprint part 16.....	207
Εικόνα 315 DeathBoxTrigger.....	207
Εικόνα 316 Level1 Level Blueprint part 17.....	208
Εικόνα 317 PlatformMoveTrigger.....	208
Εικόνα 318 Level1 Level Blueprint part 18.....	208

Εικόνα 319 Room3 Triggers .....	209
Εικόνα 320 Level1 Level Blueprint part 19.....	210
Εικόνα 321 Level1 Level Blueprint part 20.....	210
Εικόνα 322 Remove-AddPurple .....	211
Εικόνα 323 Level1 Level Blueprint part 21.....	211
Εικόνα 324 Level1 Level Blueprint part 22.....	211
Εικόνα 325 Level1 Level Blueprint part 23.....	212
Εικόνα 326 Level1 Level Blueprint part 24.....	212
Εικόνα 327 Level1 Level Blueprint part 25.....	213
Εικόνα 328 Level1 Level Blueprint part 26.....	213
Εικόνα 329 Level1 Level Blueprint part 27.....	214
Εικόνα 330 Level1 Level Blueprint part 28.....	214
Εικόνα 331 Level1 Level Blueprint part 29.....	214
Εικόνα 332 Level1 Level Blueprint part 30.....	215
Εικόνα 333 Level1 Level Blueprint part 31.....	215
Εικόνα 334 Level1 Level Blueprint part 32.....	216
Εικόνα 335 Level1 Level Blueprint part 33.....	216
Εικόνα 336 SetCharacterLocation/Rotation .....	217
Εικόνα 337 MoreHa.....	217
Εικόνα 338 LevelEnd .....	217
Εικόνα 339 Level2 My Blueprint Tab.....	218
Εικόνα 340 Level2 Level Blueprint part 1.....	219
Εικόνα 341 Level2 Level Blueprint part 2.....	220
Εικόνα 342 Level2 Level Blueprint part 3.....	220
Εικόνα 343 HallWay1Trigger .....	221
Εικόνα 344 Level2 Level Blueprint part 4.....	221
Εικόνα 345 Room1Trigger.....	222
Εικόνα 346 Level2 Level Blueprint part 5.....	222
Εικόνα 347 BP_Candy.....	223
Εικόνα 348 Level2 Level Blueprint part 6.....	223
Εικόνα 349 Room2HookPoints.....	224
Εικόνα 350 Level2 Level Blueprint part 7.....	224

Εικόνα 351 Level2 Level Blueprint part 8.....	225
Εικόνα 352 TriggerBox.....	225
Εικόνα 353 Level2 Level Blueprint part 9.....	226
Εικόνα 354 Level2 Level Blueprint part 10.....	226
Εικόνα 355 PassLevelTrigger .....	227
Εικόνα 356 Level6 World Settings.....	227
Εικόνα 357 Level6 My Blueprint Tab.....	228
Εικόνα 358 Level6 Level Blueprint part 1.....	229
Εικόνα 359 Level6 Level Blueprint part 2.....	229
Εικόνα 360 Level6 Level Blueprint part 3.....	230
Εικόνα 361 Level6 Level Blueprint part 4.....	230
Εικόνα 362 BP_Candy.....	231
Εικόνα 363 Level6 Level Blueprint part 5.....	231
Εικόνα 364 Level6 Level Blueprint part 6.....	232
Εικόνα 365 HookPathTrigger and Path .....	232
Εικόνα 366 Level6 Level Blueprint part 7.....	233
Εικόνα 367 Level6 Level Blueprint part 8.....	233
Εικόνα 368 Level6 Level Blueprint part 9.....	234
Εικόνα 369 BP_Candy3/Launcher .....	234
Εικόνα 370 Level6 Level Blueprint part 10.....	234
Εικόνα 371 Level6 Level Blueprint part 11.....	235
Εικόνα 372 Level6 Level Blueprint part 12.....	235
Εικόνα 373 PlatformPathTrigger .....	235
Εικόνα 374 Level6 Level Blueprint part 13.....	236
Εικόνα 375 FinalPathTrigger .....	236
Εικόνα 376 Level6 Level Blueprint part 14.....	237
Εικόνα 377 Level6 Level Blueprint part 15.....	237
Εικόνα 378 Level6 Level Blueprint part 16.....	238
Εικόνα 379 Level4 World Settings.....	238
Εικόνα 380 Level4 My Blueprint .....	239
Εικόνα 381 Level4 Level Blueprint part 1.....	240
Εικόνα 382 Level4 Level Blueprint part 2.....	240

Εικόνα 383 Level4 Level Blueprint part 3.....	241
Εικόνα 384 Level4 Level Blueprint part 4.....	241
Εικόνα 385 Level4 Level Blueprint part 5.....	242
Εικόνα 386 Level4 Level Blueprint part 6.....	242
Εικόνα 387 Level4 Level Blueprint part 7.....	243
Εικόνα 388 Mat_BlockShieldFinal .....	249
Εικόνα 389 MI_BlockTPShieldFinal.....	249
Εικόνα 390 MI_DoorCode .....	250
Εικόνα 391 MI_DisinRed .....	250
Εικόνα 392 MI_SimpleBlueDisinegrationE.....	251
Εικόνα 393 MI_SimpleDisinVioletLE.....	251
Εικόνα 394 M_Vortex .....	252
Εικόνα 395 Mat_Beam .....	252
Εικόνα 396 Mat_CandyDisin .....	253
Εικόνα 397 Mat_MetalPlate1 .....	253
Εικόνα 398 Mat_SimpleGlowRotation.....	254
Εικόνα 399 MI_WallGlowRoomGreen .....	254
Εικόνα 400 1M_Cube .....	255
Εικόνα 401 SM_circle_inv2 .....	255
Εικόνα 402 SM_cylinder_lowPoly_dist .....	256
Εικόνα 403 SM_Cylinder_LP .....	256
Εικόνα 404 SM_lans.....	257
Εικόνα 405 SM_Plane .....	257
Εικόνα 406 SM_PurplePlatform .....	258
Εικόνα 407 SM_quads.....	258
Εικόνα 408 SphereRooms_SM .....	259



# ΚΕΦΑΛΑΙΟ 1 Εισαγωγή

---

## 1.1 Παιχνίδια

---

Η δημιουργία των παιχνιδιών και η ενασχόληση με αυτά ανήκει στα βασικά ένστικτα του Homo Sapiens. Οι αρχαίοι Έλληνες, οι Βίκινγκς, ακόμη και οι άνθρωποι των σπηλαίων είχαν συστήματα παιχνιδιών βασισμένα σε κανόνες (Egenfeldt-Nielsen, Smith, & Tosca, 2020)

Ο Γερμανός φιλόσοφος Ludwig Wittgenstein (1889 – 1951) δεν κατάφερε να βρει έναν ορισμό που θα ισχύει για όλα τα παιχνίδια. Είναι το πόκερ και το Assassin's creed παραδείγματα του ίδιου φαινομένου; Το πόκερ παίζεται με πολλούς παίκτες και απαρτίζεται από κανόνες ενός πραγματικού κόσμου. Αντίθετα το Assassin's creed είναι παιχνίδι μάχης και μυστηρίου που διαδραματίζεται σε έναν φαντασμαγορικό κόσμο με κανόνες που μιμούνται όμως τον πραγματικό και παίζεται με έναν παίκτη. Δύο τόσο διαφορετικά παιχνίδια που όμως ανήκουν στην ίδια κατηγορία: κλασικά παιχνίδια με κοινά σημεία, διότι και στα δύο ο παίκτης αντιμετωπίζει αντιπάλους και οι επιλογές του αξιολογούνται με βάση τους κανόνες του παιχνιδιού (Egenfeldt-Nielsen, Smith, & Tosca, 2020).

Ασφαλώς λοιπόν υπάρχουν πολλές κατηγορίες παιχνιδιών : παραδοσιακά εξωτερικού χώρου όπως κρυφτό, κυνηγητό κ.α, επιτραπέζια όπως τάβλι, σκάκι κ.α., κονσόλας, όπως το metal slug στο playstation και τα ηλεκτρονικά-ψηφιακά παιχνίδια που συνεχώς πληθαίνουν και εμπλουτίζονται. Τα τελευταία, που θα ονομάζονται εφεξής βιντεοπαιχνίδια, κερδίζουν όλο και περισσότερο έδαφος έναντι των παραδοσιακών και αποτελούν εν μέρει το αντικείμενο αυτής της εργασίας.

## 1.2 Αναδρομή στην ιστορία των ηλεκτρονικών παιχνιδιών

---

Κάνοντας μια αναδρομή στην εξέλιξη των βιντεοπαιχνιδιών βλέπουμε τεράστια βήματα ανάπτυξης. Ξεκινώντας από το πρώτο «φλίπερ» του Martin Bromley το 1951, φτάνουμε στις αίθουσες ηλεκτρονικών (arcades) τη δεκαετία του 70, οι οποίες είναι διαθέσιμες στο ευρύ κοινό. Στη συνέχεια το 1962 ο Steve Russell, φοιτητής του MIT, ανέπτυξε τον κώδικα για το παιχνίδι SpaceWar. Η πρώτη οικιακή παιχνιδιομηχανή που κυκλοφόρησε το 1972, περιείχε μόνο ένα παιχνίδι και συνδεόταν στην τηλεόραση. Έκτοτε η ραγδαία ανάπτυξη της τεχνολογίας, των Η/Υ και η διεύρυνση του διαδικτύου συμπαρέσυρε και τα βιντεοπαιχνίδια, τα οποία αποτελούσαν ακριβή ψυχαγωγία για κάποιους και έγιναν σε χρόνο ρεκόρ η πιο δημοφιλή απασχόληση για μικρούς και μεγάλους (Αναγνώστου, 2009).

Σήμερα πλέον ακόμη και όσοι δεν διαθέτουν προσωπικό Η/Υ, αλλά μόνο κινητό τηλέφωνο, έχουν πρόσβαση σε βιντεοπαιχνίδια, που θεωρούνται πλέον μια ακόμη μορφή ψυχαγωγίας. Η δημοτικότητα τους έχει αυξηθεί κατακόρυφα. Λόγω της ευρείας διάδοσης έγιναν αντικείμενο μελέτης ως επιρροή στη σωματική, ψυχική και πνευματική υγείας των παικτών. Παρότι έχει συζητηθεί το θέμα της βίας στα βιντεοπαιχνίδια (κάποια έχουν αποσυρθεί για το λόγο αυτό) και του εθισμού, δεν έχει αποδειχθεί ότι ευθύνονται για άσκηση παραβατικής συμπεριφοράς (Αναγνώστου, 2009). Αντίθετα οι θετικές επιρροές των βιντεοπαιχνιδιών είναι πάρα πολλές. Έτσι πέρα από την ψυχαγωγική πλευρά, ενισχύουν επίσης τον πειραματισμό, την εξερεύνηση και την εξέλιξη φυσικών δεξιοτήτων, όπως για παράδειγμα την αντίληψη του τρισδιάστατου χώρου και την άμεση αντίδραση. Επίσης το αλληλεπιδραστικό τους περιβάλλον προσφέρει άμεση ανάδραση και συνεπώς δυνατότητες για εμπειρική μάθηση (Αναγνώστου, 2009). Επιπλέον, πολλά βιντεοπαιχνίδια σχεδιάστηκαν για να βελτιώσουν την υγεία των παικτών, ακόμη και με χρόνιες παθήσεις, βελτιώνοντας δεξιότητες αυτό-φροντίδας και συμμόρφωσης σε ιατρικές οδηγίες (Griffiths, 2002).

Τα βιντεοπαιχνίδια είναι πολύ δημοφιλή και στη χώρα μας. Σύμφωνα με έρευνες οι προτιμήσεις ηλεκτρονικού παιχνιδιού και η συχνότητα χρήσης εξαρτώνται από το

φύλο, τον τόπο διαμονής και το μορφωτικό επίπεδο των γονέων. Αυτοί εμπλέκονται και στην επιλογή του βιντεοπαιχνιδιού και του χρόνου που αφιερώνεται σ' αυτό, όταν ο παίκτης είναι μικρότερης ηλικίας (Χατζής, 2006).

Τα βιντεοπαιχνίδια όμως δεν είναι μόνο ψυχαγωγία και διασκέδαση, αλλά λόγω του εύρους των χαρακτηριστικών τους όπως αλληλεπίδραση, τήρηση κανόνων, άμεση ανάδραση επιπτώσεων, αναπαράσταση γεγονότων και ιστοριών, προσομοίωση πραγματικού κόσμου, πολλαπλές ευκαιρίες διόρθωσης λαθών, κίνητρο ενασχόλησης, δυνατότητα συμμετοχής ομάδας παικτών έχουν και εκπαιδευτική αξία. Για τον λόγο αυτό τα βιντεοπαιχνίδια βρίσκουν εφαρμογή σε πολλούς τομείς όπως στον τομέα του στρατού, της εκπαίδευσης, του χώρου υγείας (Kato, 2010), των επιχειρήσεων, οπότε χρήζουν και ιδιαίτερης προσοχής (Αναγνώστου, 2009)

### 1.3 Είδη ψηφιακών παιχνιδιών

---

Λόγω της πληθώρας τους, τα βιντεοπαιχνίδια, σύμφωνα με το στυλ και τον τρόπο με τον οποίο παίζονται, αλλά ανεξάρτητα από το θέμα και την ιστορία τους, κατατάσσονται στις εξής γενικές κατηγορίες (Αναγνώστου, 2009):

- ✓ Δράσης (action games), η παλαιότερη κατηγορία με κοινό χαρακτηριστικό την απαίτηση για γρήγορα αντανακλαστικά και καλό συγχρονισμό χειριού-ματιού (π.χ. pac man), με τις παρακάτω υποκατηγορίες:
  - Παιχνίδια βολών 1<sup>ου</sup> προσώπου (first person shooters) π.χ. Halo
  - Παιχνίδια βολών 3<sup>ου</sup> προσώπου (third person shooters) π.χ. Fortnite
  - Παιχνίδια πάλης (fighting games) π.χ. Tekken
  - Παιχνίδια πλατφόρμας (platform games) π.χ. Prince of Persia
- ✓ Περιπέτειας (Adventure games), που εκτός της δράσης απαιτούν σκέψη, εξερεύνηση και λύση γρίφων π.χ. God of War
- ✓ Στρατηγικής (Strategy games), που απαιτούν την διαχείριση πόρων και ανθρώπινου δυναμικού καθώς και τη λήψη αποφάσεων για θέματα πολέμου, εμπορίου ή κατάκτησης εδαφών π.χ. Warcraft

- ✓ Παιχνίδια ρόλων (role-playing games), που αποτελούνται από σύνθετα σενάρια με διαφορετικές αποστολές και έχοντας έντονο το στοιχείο της εξερεύνησης ο παίκτης αναβαθμίζει τον χαρακτήρα προσδίδοντάς του διάφορες ικανότητες/εφόδια π.χ. Star Wars: Knights of Old Republic
- ✓ Παιχνίδια σκέψης (puzzle games), που στηρίζονται στη λύση γρίφων, έχουν απλό χειρισμό, σύντομη διάρκεια, απαιτούν χαμηλή υπολογιστική ισχύ και λόγω αυτού έχουν εξαπλωθεί σε όλες τις πλατφόρμες π.χ. tetris
- ✓ Παιχνίδια προσομοίωσης (simulation games), που αναπαράγουν συνθήκες του πραγματικού κόσμου σε εικονικό και έχουν τις παρακάτω υποκατηγορίες:
  - Παιχνίδια προσομοίωσης οχημάτων (vehicle simulation) π.χ. Need for Speed
  - Αθλητικά παιχνίδια (sports games) π.χ. FIFA, Pro, NFL
  - Παιχνίδια κατασκευής και μάνατζμεντ (construction and management simulations) π.χ. Simcity
  - Μουσικά παιχνίδια π.χ. Guitar Hero

Στις μέρες μας ολόκληρος κλάδος της βιομηχανίας ασχολείται με τα βιντεοπαιχνίδια και πολλές εταιρείες επενδύουν πλέον στον τομέα σχεδίασης και ανάπτυξης παιχνιδιών. Παρά το στάδιο στο οποίο βρίσκονται προς το παρόν τα βιντεοπαιχνίδια, αναμένεται ραγδαία η εξέλιξή τους. Για παράδειγμα, προσεχώς θα κάνουν την εμφάνισή τους παιχνίδια που δύσκολα θα μπορούσαμε να εικάσουμε από τώρα τις δυνατότητές τους. Επομένως, σε σχέση με την μελλοντική τους εξέλιξη, θα λέγαμε ότι βρίσκονται ακόμα σε νηπιακή ηλικία.

Για τις ανάγκες της παρούσας πτυχιακής εργασίας σχεδιάστηκε και υλοποιήθηκε το παιχνίδι «The Challenger» που θα μπορούσε να ενταχθεί στην κατηγορία Puzzle Platform με action στοιχεία, ενσωματώνοντας φυσικά την αίσθηση του χιούμορ του δημιουργού και το μήνυμα που θέλει να περάσει, άρα και στοιχεία της προσωπικότητάς του.

# ΚΕΦΑΛΑΙΟ 2 Μηχανές δημιουργίας βιντεοπαιχνιδιών

---

## 2.1 Εισαγωγή

---

Μηχανή παραγωγής βιντεοπαιχνιδιών (game engine) ονομάζεται μια εργαλειοθήκη-εφαρμογή που προσφέρει ένα σύνολο λειτουργιών για την αυτοματοποίηση του χειρισμού ενός μεγάλου εύρους σειράς διεργασιών (γραφικά, ήχος, φυσική παιχνιδιών, δίκτυα, τεχνητή νοημοσύνη). Ο χειρισμός αντικειμένων σε αυτές τις μηχανές παιχνιδιών γίνεται σύμφωνα με τις αρχές του αντικειμενοστραφούς προγραμματισμού (Charrieras & Ivanova, 2016).

Η μηχανή παραγωγής βιντεοπαιχνιδιών ουσιαστικά αναλαμβάνει την υλοποίηση επαναλαμβανόμενων διαδικασιών δίνοντας στον δημιουργό του παιχνιδιού τον χρόνο και την ευκαιρία να επικεντρωθεί στο σχεδιασμό του παιχνιδιού (Αναγνώστου, 2009)

Υπάρχουν διαθέσιμες μηχανές παραγωγής βιντεοπαιχνιδιών με διαφορετικές δυνατότητες, που διατίθενται στους σχεδιαστές παιχνιδιών:

- με αντάλλαγμα μερίδιο από τα κέρδη του παιχνιδιού όταν αυτό δοθεί στο εμπόριο (π.χ. Unreal Game Engine)
- με πληρωμή της άδειας χρήσης
- ελεύθερες σε περιορισμένη έκδοση
- δωρεάν

Μερικές από τις μηχανές παιχνιδιού (game engine) είναι οι ακόλουθες: GameMaker, JMonkey, Ogre 3D, Shiva Game Engine, Sio 2, Turbulenz, Unity, Unreal Engine's, Godot και πολλές ακόμη (Christopoulou & Xinogalos, 2017). Δεν υπάρχει όμως μία μηχανή παιχνιδιού που να είναι η καλύτερη για κάθε σκοπό, αλλά κάθε χρήστης επιλέγει την καταλληλότερη για τις ανάγκες του παιχνιδιού που θέλει να σχεδιάσει (Christopoulou & Xinogalos, 2017). Κάποιες από τις μηχανές παιχνιδιών αναφέρονται συνοπτικά παρακάτω.

## 2.2 Unity

---

Η μηχανή παιχνιδιού Unity Game ανακοινώθηκε για πρώτη φορά δημόσια στο Worldwide Developers της Apple σε συνέδριο το 2005 και είναι μια από τις γνωστότερες μηχανές παιχνιδιών. Η Unity παρέχει προηγμένη 3D απόδοση γραφικών και δωρεάν εξαγωγή σε έξυπνα κινητά. Επίσης είναι μία από τις δημοφιλέστερες μηχανές παιχνιδιών στη βιομηχανία, ενώ με αυτή έχουν δημιουργηθεί πολλά επιτυχημένα παιχνίδια, όπως Deus Ex: The Fall, Assassin's Creed: Identity (Christopoulou & Xinogalos, 2017).

## 2.3 GameMaker

---

Το GameMaker κυκλοφόρησε για πρώτη φορά με το όνομα Animo το 1991 και ήταν κατάλληλο για δημιουργία 2D παιχνιδιών. Εντούτοις επιτρέπει στον χρήστη να προσθέσει τρισδιάστατα γραφικά και φυσική. Το GameMaker είναι εύκολο στη χρήση για αρχάριους χρήστες, καθώς δεν απαιτεί γνώσεις προγραμματισμού. (Christopoulou & Xinogalos, 2017)

## 2.4 TurbulenzGame

---

Η μηχανή παιχνιδιών Turbulenz δημιουργήθηκε τον Απρίλιο του 2013 και είναι μια μηχανή που επιτρέπει την εξαγωγή παιχνιδιών σε προγράμματα περιήγησης για κινητά, με βάση την πολλά υποσχόμενη προσέγγιση των τεχνολογιών Ιστού. Ταυτόχρονα, η Turbulenz είναι δωρεάν, ανοιχτού κώδικα και υποστηρίζει τρισδιάστατη απόδοση γραφικών (Christopoulou & Xinogalos, 2017).

## 2.5 Godot

---

Με τη μηχανή παιχνιδιών Godot παρέχεται μια επιπλέον προσέγγιση στην ανάπτυξη παιχνιδιών, με εκατοντάδες προγραμματιστικούς κόμβους (Nodes), με ευέλικτο σύστημα σκηνής (Flexible scene system), με οπτικό επεξεργαστή (Visual

editor) και με τη δυνατότητα μόνιμης ζωντανής επεξεργασίας όπου οι αλλαγές δεν χάνονται μετά τη διακοπή του παιχνιδιού. Η μηχανή Godot εκτελείται ακόμη και σε έξυπνες συσκευές (Linietsky & Manzur, 2007-2020).

## ΚΕΦΑΛΑΙΟ 3 ΙΣΤΟΡΙΚΗ ΑΝΑΣΚΟΠΗΣΗ ΤΗΣ UNREAL ENGINE

---

Στο συγκεκριμένο κεφάλαιο θα δούμε αναλυτικότερα την Unreal Engine που χρησιμοποιήθηκε κατά την εκπόνηση της πτυχιακής εργασίας για τη δημιουργία του «The Challenger». Η Game Engine Unreal είναι από τις δημοφιλέστερες μηχανές παραγωγής παιχνιδιών, ενώ έχει καταγραφεί στο βιβλίο των ρεκόρ Γκίνες το 2014 ως η πιο επιτυχημένη μηχανή παραγωγής παιχνιδιών.

Το Unreal Tournament αναπτύχθηκε στην Unreal Engine. Είναι το 1<sup>ο</sup> της παιχνίδι και είναι ανοιχτού κώδικα (Lewis & Jacobson, 2002). Ενσωματώνει τη δική του προγραμματιστική γλώσσα που ονομάζεται UnrealScript και το δικό της επεξεργαστή χαρτών που ονομάζεται UnrealEd (Paul, Goon, & Bhattacharya, 2012)

Αν και αρχικά αναπτύχθηκε για παιχνίδια βολών 1<sup>ου</sup> προσώπου (Unreal first person shooter) αναμφισβήτητα έχει εξελιχθεί σε πολύ ισχυρή μηχανή παιχνιδιού ικανή να υποστηρίξει οποιοδήποτε είδος παιχνιδιού. Μια άλλη διαφορά της Unreal Engine είναι ότι ο πηγαίος κώδικας είναι ανοιχτός. Αυτό προσφέρει στους προγραμματιστές που θέλουν να ερευνήσουν περισσότερο την ευκαιρία να αποκτήσουν μια επιπλέον εμπειρία μάθησης (Andrade, 2015).

### 3.1 Unreal engine 1

---

Σύμφωνα με συνέντευξη του 2017 που έδωσε ο ιδρυτής της Unreal Engine, Tim Sweeney, το 1995 ξεκίνησε να γράφει τον κώδικα για έναν επεξεργαστή-Editor ώστε να βοηθήσει τον James Schmaltz που έγραφε έναν επεξεργαστή σε assembly. Κάποια από τα χαρακτηριστικά του Editor του Sweeney ήταν η ανίχνευση συγκρούσεων, έγχρωμο φωτισμό και μια περιορισμένη έκδοση φιλτραρίσματος υφών. Επίσης υποστήριζε την κατασκευή γεωμετρίας σε πραγματικό χρόνο (Lightbown,



2018). Η αρχική κυκλοφορία της Unreal Engine ήταν το 1998 (Christopoulou & Xinogalos, 2017)

## 3.2 Unreal Engine 2

---

Η Unreal Engine 2 κυκλοφόρησε το 2001, και αναβαθμίστηκε τελευταία φορά το 2005 προτού αντικατασταθεί από την 3. Οι αναβαθμίσεις αφορούσαν Basics, Techniques, Tools, Primitives, Lighting and Textures, Effects, Animation, Physics, Matinee, Scripted Sequences (Epic Games, n.d.)

## 3.3 Unreal Engine 3

---

Η Unreal Engine 3 κυκλοφόρησε το 2007 και υποστήριζε τις κονσόλες Playstation 3, Xbox 360 και φυσικά Η/Υ . Οι αναβαθμίσεις αφορούσαν στα συστήματα Levels , Textures and Materials, Art and Animation, Characters, Weapons and Vehicles, Effects, Cinematics, Gameplay Scripting, Music and Sound (Epic Games, n.d.)

Επιπλέον, σύμφωνα με μια έρευνα η Unreal Engine 3 εκτός από πλατφόρμα για δημιουργία βιντεοπαιχνιδιών, αποτελούσε μια κατάλληλη πλατφόρμα για την ανάπτυξη μιας αποτελεσματικής και ακριβούς προσομοίωσης προβλημάτων όρασης (Lewis, Brown, Cranton, & Mason, 2011).

## 3.4 Unreal Engine 4

---

Η Unreal Engine 4 κυκλοφόρησε για πρώτη φορά τον Μάιο του 2012 και χρησιμοποιήθηκε για να χτίσει συστήματα εικονικής πραγματικότητας και να δημιουργήσει διαδραστικές λειτουργίες (Yuxiang, Xiaowei, & Nongliang, 2020).

Η έκδοση 4.24 που κυκλοφόρησε τον Δεκέμβριο του 2019 είναι αυτή που χρησιμοποιήθηκε κατά τον σχεδιασμό και την υλοποίηση του «The Challenger». Ο χρήστης μπορεί πλέον να δημιουργήσει εσωτερικές και εξωτερικές σκηνές για παιχνίδια και οπτικά με νέα εργαλεία. Επίσης έχει τη δυνατότητα να αναπτύξει παιχνίδια τύπου «ανοιχτού κόσμου» (open world) που περιέχουν δυναμικούς ουράνιους θόλους. Μια άλλη δυνατότητα που του προσφέρεται είναι η δημιουργία κίνησης (animation) με απεικόνιση κινούμενης κόμης (Epic Games, n.d.).

Η 4.25 είναι η πρώτη που υποστηρίζει το Sony PlayStation 5 και το Xbox Series X της Microsoft καθώς και τη χρήση DirectX (Windows, Xbox One), OpenGL (OSX, Linux, PlayStation 4, iOS, Android και Windows XP) και WebGL, για προγράμματα περιήγησης ιστού HTML5 (Andrade, 2015).

Τα προτεινόμενα χαρακτηριστικά των συστημάτων την εταιρείας (Epic Games) για την ανάπτυξη παιχνιδιών με την Unreal Engine 4 είναι τα εξής:

- ✓ Windows 10 64-bit
- ✓ 64GB RAM
- ✓ 256GB SSD (OS Drive)
- ✓ 2TB SSD (Data Drive)
- ✓ NVIDIA GeForce GTX 970
- ✓ Xoreax Incrédibuild (Dev Tools Package)
- ✓ Six-Core Xeon E5-2643 @3.4GHz

### 3.5 Το μέλλον της Unreal Engine

---

Η Epic έχει ανακοινώσει την κυκλοφορία της Unreal Engine 5 το 2021 (Epic Games, 2020). Πρόκειται για τη νέα γενιά της Unreal που θα ενσωματώνει τεχνολογίες όπως:

- ✓ Nanite, εικονική μικροπολυγωνιακή γεωμετρία, που θα δώσει στους σχεδιαστές παιχνιδιών την ελευθερία να σχεδιάσουν με όση γεωμετρική λεπτομέρεια μπορεί το ανθρώπινο μάτι να διακρίνει

- ✓ Lumen. Πρόκειται για ένα πλήρως δυναμικό εργαλείο φωτισμού, που έχει άμεση ανταπόκριση/αντίδραση σε κάθε αλλαγή του φωτός και έτσι οι σχεδιαστές θα μπορούν να δημιουργήσουν πιο δυναμικές σκηνές αλλάζοντας π.χ. την γωνία του ήλιου για την ώρα της ημέρας.

Η Unreal Engine 5 θα υποστηρίζει όλες τις πλατφόρμες.

# ΚΕΦ. 4 ΣΤΑΔΙΑ ΣΧΕΔΙΑΣΜΟΥ ΚΑΙ ΥΛΟΠΟΙΗΣΗΣ ΒΙΝΤΕΟΠΑΙΧΝΙΔΙΩΝ

---

## 4.1 Σχεδιασμός του παιχνιδιού

---

«Ο σχεδιασμός του παιχνιδιού είναι τέχνη και τεχνική μαζί». Ο σχεδιαστής προσδιορίζει τα δομικά στοιχεία του παιχνιδιού, την αρμονική συνύπαρξή τους, καθορίζει την ισορροπία, τις δυναμικές του συστήματος του παιχνιδιού για να είναι λειτουργικό και δίκαιο. Για να τα πετύχει αυτά, με τη φαντασία του δημιουργεί συναρπαστικούς εικονικούς κόσμους και με τις γνώσεις για το μέσο που θα χρησιμοποιήσει για την υλοποίηση προχωρά στην ανάπτυξη του παιχνιδιού, συχνά συνεργαζόμενος με πολλές ειδικότητες (Αναγνώστου, 2009).

Η πλοκή του βιντεοπαιχνιδιού *The Challenger* είναι η εξής:

Σε έναν φανταστικό κόσμο, υπάρχει ένα μυστήριο συγκρότημα κτισμάτων (Feli) όπου επιλέγει να πάει όποιος επιθυμεί να αποκτήσει την δύναμη να πραγματοποιεί οποιαδήποτε ευχή του. Για να γίνει αυτό πρέπει να αποδείξει την αξία του περνώντας ένα σύνολο δοκιμασιών. Στο Feli βρίσκεται ο ήρωας (*The Challenger*) που προσπαθώντας να περάσει τις δοκιμασίες πεθαίνει πολλές φορές. Όμως η φωνή/κόκκινη-μπάλα τον ανασταίνει συνεχώς, αφού ικανοποιείται βλέποντας τον θάνατο του. Έτσι, έρχεται η στιγμή που ο ήρωας αποφασίζει να ξεφύγει από το Feli ακόμα και αν τελικά δεν αποκτήσει την δύναμη να πραγματοποιεί τις ευχές του. Η απόφαση αυτή είναι καταλυτική. Αντιλαμβάνεται ότι η ζωή του είναι πιο σημαντική από την εκπλήρωση όλων των ευχών του.

## 4.2 Στάδια ανάπτυξης εμπορικών βιντεοπαιχνιδιών

---

Η διαδικασία ανάπτυξης παιχνιδιού (game development) απαιτεί πειθαρχία ώστε να επιτευχθεί ο συνδυασμός ήχου, τέχνης, ελέγχου συστημάτων, τεχνητής

νοημοσύνης και ανθρωπίνων παραγόντων και διαφοροποιεί την ανάπτυξη παιχνιδιού από την παραδοσιακή ανάπτυξη λογισμικού (Saiqa, Capretz, & Faheem, 2016).

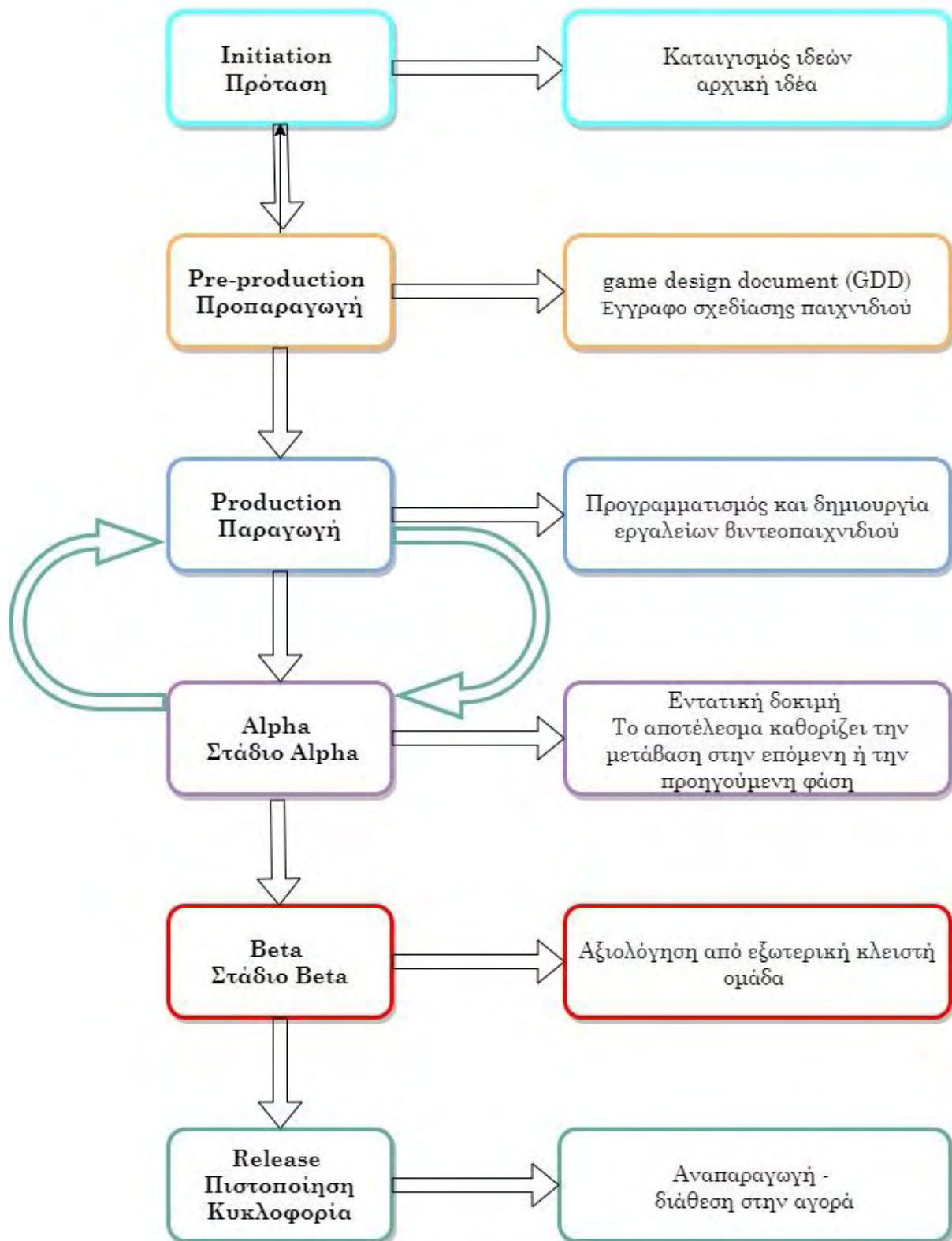
Η ανάπτυξη του παιχνιδιού ξεκινά από το μηδέν και σε γενικές γραμμές ακολουθεί τα επόμενα στάδια:

1. Initiation – Πρόταση , όπου στο στάδιο αυτό γεννιέται η αρχική ιδέα του παιχνιδιού μέσα από καταγισμό ιδεών
2. Pre-production – Προπαραγωγή, όπου περιγράφεται το παιχνίδι, οι χαρακτήρες, η πλοκή, τα συστήματα χειρισμού του παίκτη και τα υπόλοιπα χαρακτηριστικά. Κατά την φάση αυτή καταγράφονται όλα τα παραπάνω στο έγγραφο σχεδίασης παιχνιδιού game design document (GDD). Το GDD είναι εξαιρετικά σημαντικό και παραδοτέο στη φάση αυτή και περιέχει μια συναφή περιγραφή των βασικών στοιχείων, των σχέσεων τους, των κατευθύνσεων και ένα κοινό λεξιλόγιο για να συμβάλλει στην αποτελεσματική ανάπτυξη (Saiqa, Capretz, & Faheem, 2016)
3. Production – Παραγωγή, όπου δίνεται μεγαλύτερη βαρύτητα στον προγραμματισμό και τη δημιουργία των στοιχείων του παιχνιδιού. Η διαδικασία αυτή είναι μεγάλης διάρκειας και το παιχνίδι αρχίζει να παίρνει μορφή.
4. Alpha– Στάδιο Alpha . Στο στάδιο αυτό το παιχνίδι μπορεί να παιχτεί και ελέγχεται από την αντίστοιχη ομάδα αν τα στοιχεία του παιχνιδιού πληρούν τα κριτήρια, οπότε σε περίπτωση σφαλμάτων επιστρέφει στο προηγούμενο στάδιο.
5. Beta – Στάδιο Beta. Αν στο στάδιο Alpha τα αποτελέσματα είναι τα αναμενόμενα, το παιχνίδι πλέον αξιολογείται από εξωτερική κλειστή ομάδα για ανατροφοδότηση.
6. Release – Πιστοποίηση και κυκλοφορία. Το παιχνίδι διατίθεται πλέον στην αγορά. (Ramadan & Widyani, 2013)

Συνοπτικά παρουσιάζονται τα στάδια παραγωγής βιντεοπαιχνιδιού και η περιγραφή τους στην Εικόνα 1.

Στάδια ανάπτυξης βιντεοπαιχνιδιού

Συνοπτική περιγραφή σταδίων



Εικόνα 1 Απεικόνιση σταδίων παραγωγής βιντεοπαιχνιδιού

### 4.3 Στάδια ανάπτυξης του «The Challenger»

---

Για την ανάπτυξη του συγκεκριμένου παιχνιδιού στο πλαίσιο της πτυχιακής εργασίας τα στάδια ανάπτυξης διαφοροποιήθηκαν από τα γενικά στάδια ανάπτυξης εμπορικών παιχνιδιών. Έτσι μετά τον αρχικό σχεδιασμό του παιχνιδιού αποφασίστηκαν τα mechanics - μηχανισμού του παιχνιδιού, όπως η ασπίδα, ο τρόπος κίνησης, το grappling hook, η τηλεμεταφορά, οι μηχανισμοί των πορτών (puzzle), η δυνατότητα πυροβολισμού, το dash και η γενική εμφάνιση του παιχνιδιού. Όλα τα παραπάνω ανήκουν στην φάση της Προπαραγωγής - Pre-production.

Στο στάδιο της παραγωγής υλοποιήθηκαν τα mechanics, τα actors και τα υλικά-materials που χρησιμοποιήθηκαν. Στην παρούσα φάση υλοποιήθηκαν και τα επίπεδα του παιχνιδιού ανάλογα με τους διαθέσιμους πόρους. Το Alpha και Beta στάδιο έτρεχαν παράλληλα με την παραγωγή.

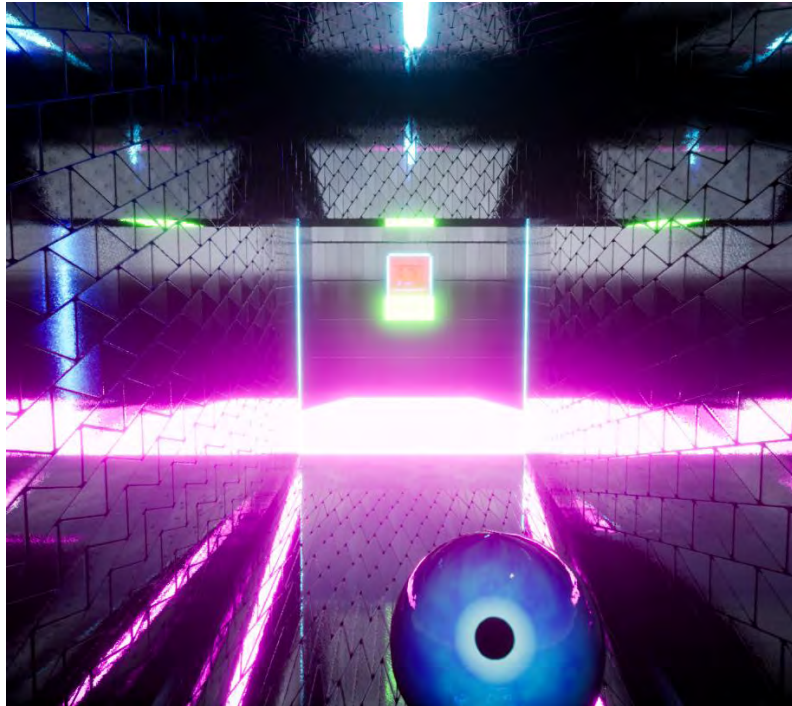
### 4.4 Γενική περιγραφή του The Challenger

---

Το The Challenger είναι παιχνίδι ενός παίκτη (single player) και ανήκει στο είδος Puzzle Platform με action στοιχεία. Ο χαρακτήρας του παίκτη κινείται σε ένα τρισδιάστατο εικονικό περιβάλλον ενός φανταστικού κόσμου προσπαθώντας να φτάσει στο τελευταίο επίπεδο, λύνοντας γρίφους και προσπαθώντας να ανταπεξέλθει στις προκλήσεις κάθε πίστας.

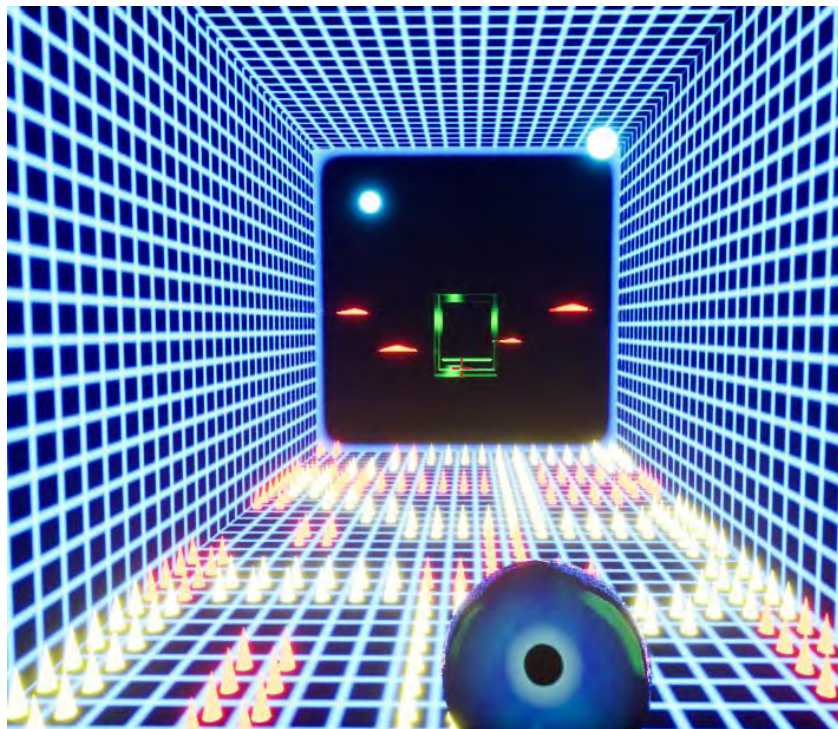
Στο 1<sup>ο</sup> επίπεδο όπως φαίνεται στην Εικόνα 2, ο παίκτης έρχεται σε πρώτη επαφή με τις δεξιότητες του χαρακτήρα.





Εικόνα 2 Στιγμιότυπο παιχνιδιού στο 1ο επίπεδο

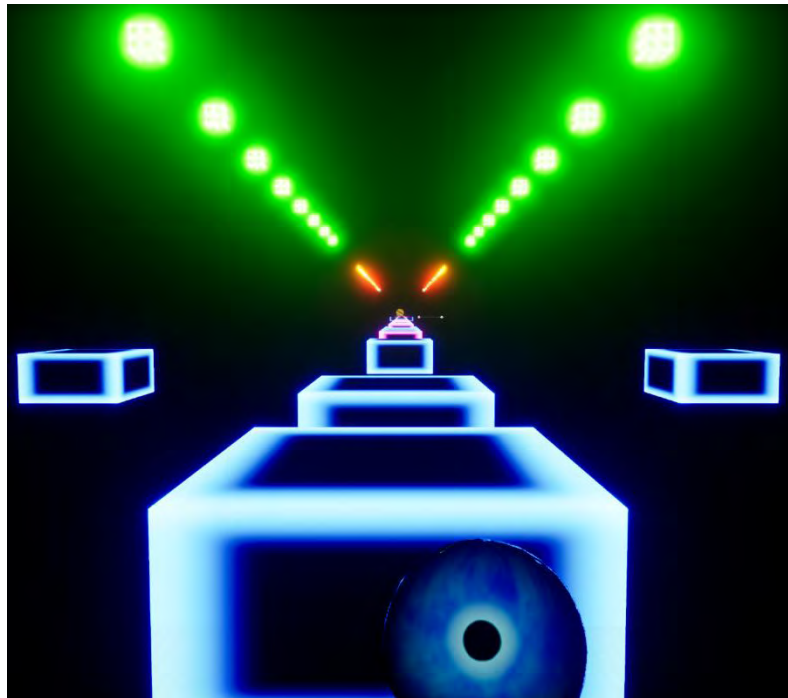
Στο 2<sup>ο</sup> επίπεδο, στιγμιότυπο του οποίου φαίνεται στην Εικόνα 3, ο παίκτης προσπαθεί να συλλέξει ζαχαρωτά, να αποφύγει παγίδες και αφού βρει τον κρυμμένο κωδικό να περάσει στο επόμενο επίπεδο.



Εικόνα 3 Στιγμιότυπο παιχνιδιού στο 2ο επίπεδο

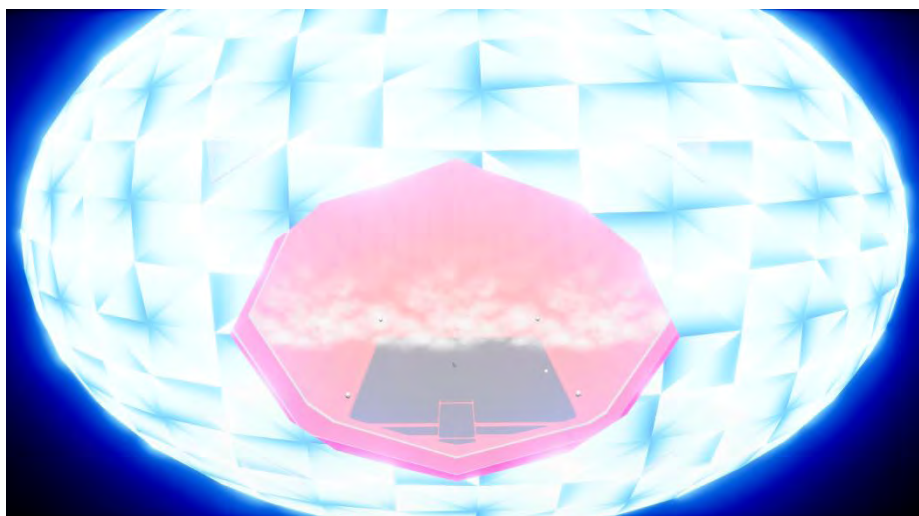


Στο 3<sup>ο</sup> επίπεδο (Εικόνα 4) ο παίκτης θα πρέπει να διεκπεραιώσει ένα σύνολο δοκιμασιών όπως η κίνηση ανάμεσα σε κινούμενες πλατφόρμες κ.α. ώστε να φτάσει στην πύλη εξόδου για το επόμενο επίπεδο.



Εικόνα 4 Στιγμιότυπο παιχνιδιού στο 3ο επίπεδο

Στο 4<sup>ο</sup> επίπεδο (Εικόνα 5) στόχος του παίκτη είναι να επιβιώσει από τους αντιπάλους του και εξουδετερώνοντάς τους να λύσει τον γρίφο για να μάθει τον κωδικό και να καταφέρει να απελευθερωθεί.



Εικόνα 5 Στιγμιότυπο παιχνιδιού στο 4ο επίπεδο

## ΚΕΦΑΛΑΙΟ 5 Η δομή του The Challenger

---

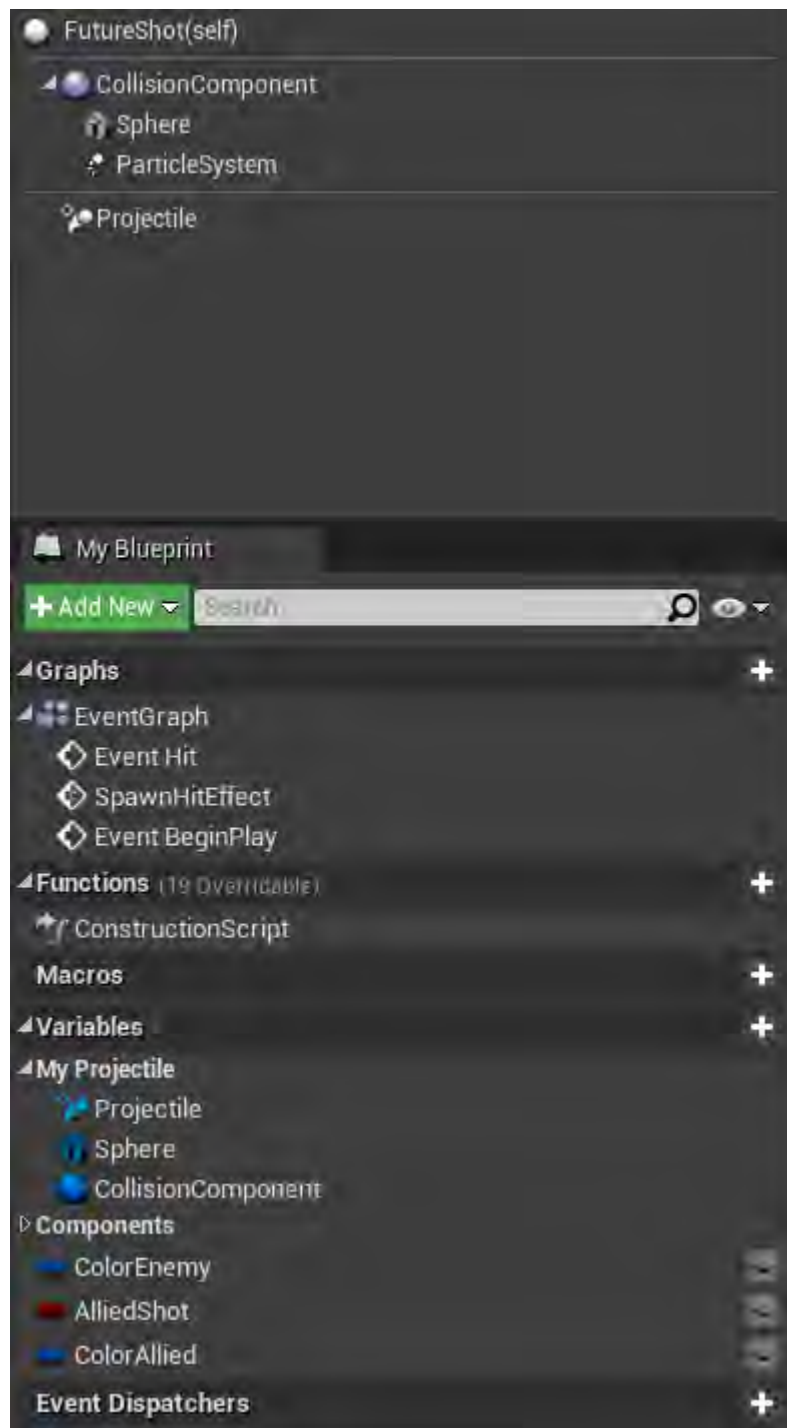
### 5.1 Blueprints

---

#### Ενότητα 5.1.1 BP\_FutureShot και παιδιά

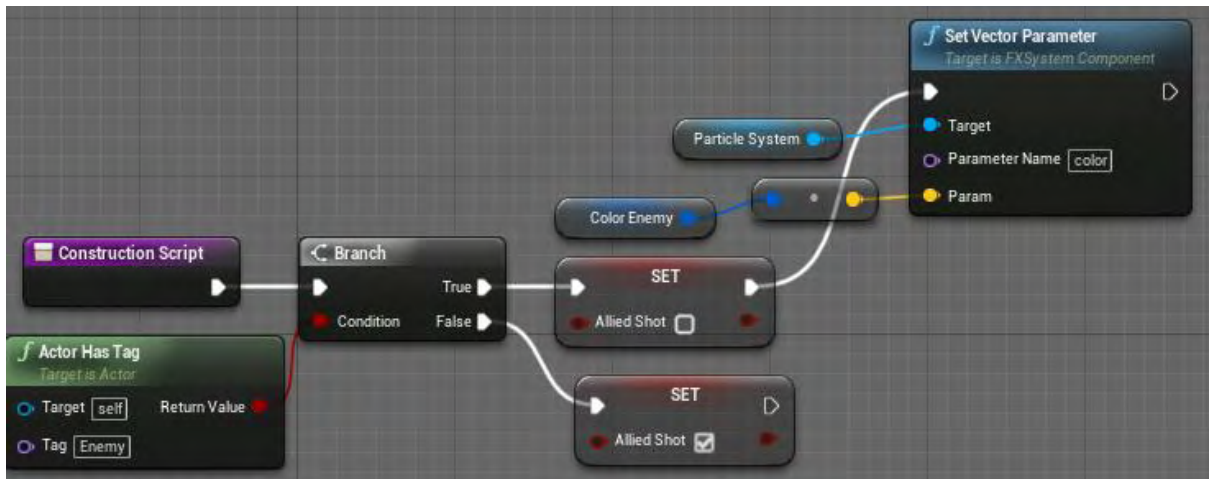
---

Το BP\_FutureShot έχει σαν root component το CollisionComponent με attached components το Static Mesh και το particle system. Επιπλέον έχει το built-in projectile movement component της Unreal. Επίσης περιέχει τις συναρτήσεις Event Hit, SpawnHitEffect και Event BeginPlay. Υπάρχουν οι μεταβλητές ColorEnemy και ColorAllied που είναι τύπου Linear Color και η μεταβλητή AlliedShot που είναι τύπου Boolean (Εικόνα 6).



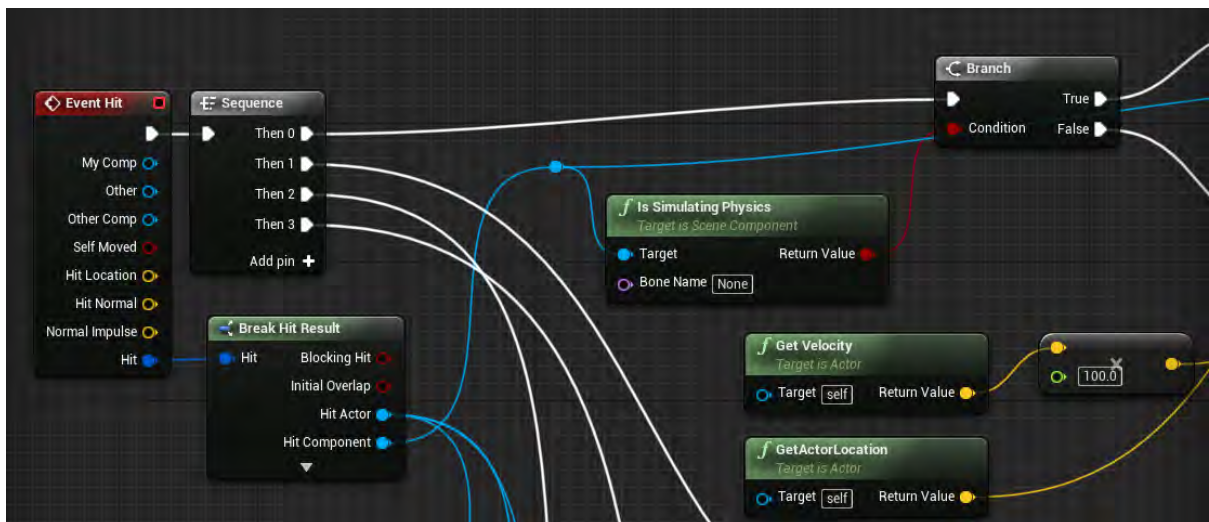
Εικόνα 6 BP\_FutureShot part1

Στο Construction Script του BP\_FutureShot ελέγχεται αν ο actor έχει την ετικέτα Enemy και αναλόγως αλλάζει την τιμή της μεταβλητής AlliedShot και αν δεν είναι AlliedShot τροποποιεί το χρώμα του Particle System του actor BP\_FutureShot (Εικόνα 7).



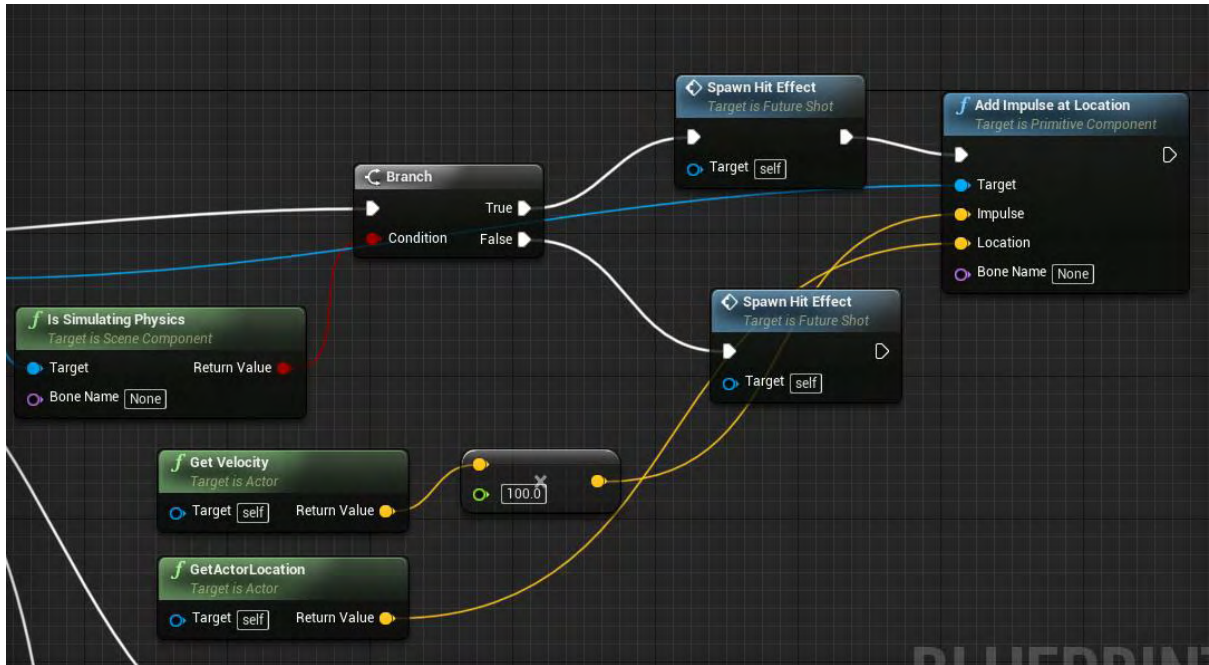
Εικόνα 7 BP\_FutureShot part 2

Για τη βολή δημιουργήθηκε το actor class BP\_FutureShot στο οποίο ορίστηκε να ανιχνεύει τη σύγκρουση και στη συνέχεια να πραγματοποιεί ελέγχους σχετικά με τον εντοπισμό του actor που συγκρούστηκε και να συμπεριφέρεται αναλόγως. Στην Εικόνα 8 BP\_FutureShot φαίνεται η ανίχνευση της σύγκρουσης και η έναρξη των σειριακών ελέγχων, καθώς και ο πρώτος έλεγχος ο οποίος αφορά την ενεργοποίηση της φυσικής του συγκρουόμενου actor.



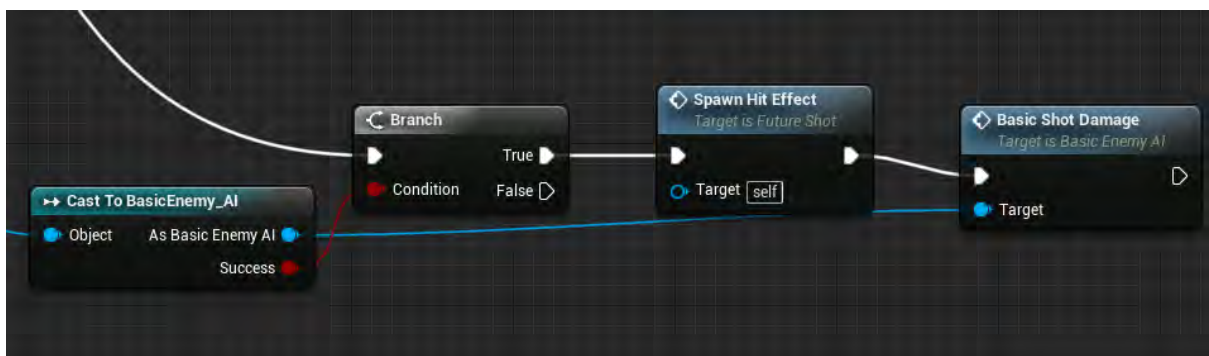
Εικόνα 8 BP\_FutureShot part3

Στην Εικόνα 9 φαίνεται ότι αν ο πρώτος σειριακός έλεγχος είναι αληθής, τότε προστίθεται ώθηση (impulse) στο σημείο σύγκρουσης.



Εικόνα 9 BP\_FutureShot part 4

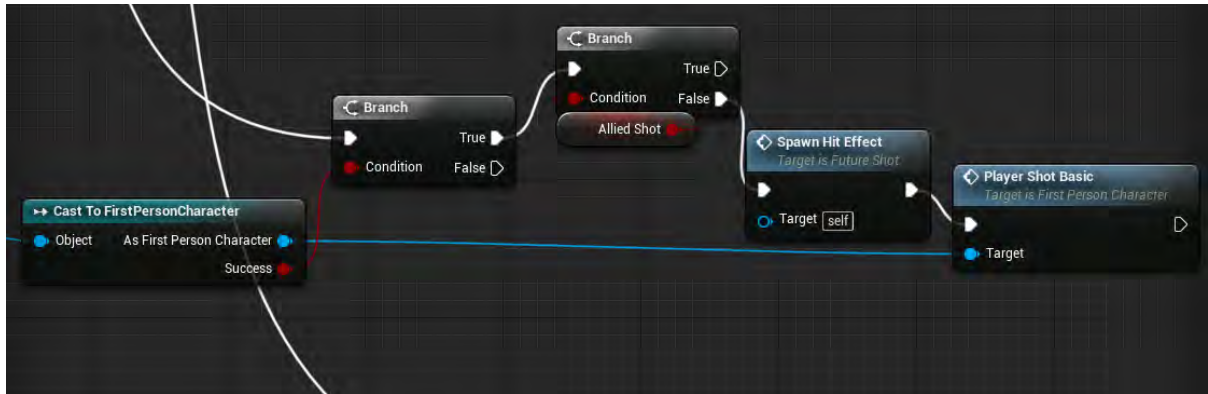
Αν ο actor είναι BasicEnemy\_AI, καλεί τη συνάρτηση SpawnHitEffect και στη συνέχεια τη συνάρτηση Basic Shot Damage του BasicEnemy\_AI. Αυτό απεικονίζεται στην Εικόνα 10.



Εικόνα 10 BP\_FutureShot part 5

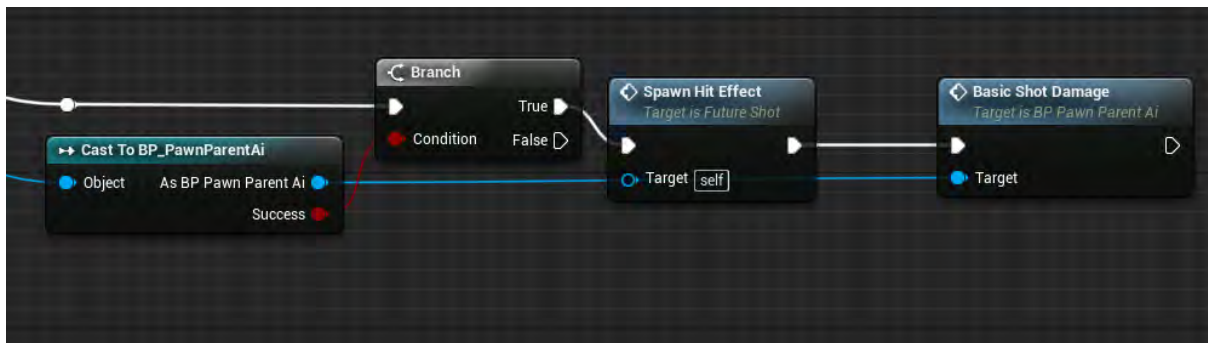


Αν ο actor είναι FirstPersonCharacter καλεί τη συνάρτηση SpawnHitEffect και στη συνέχεια τη συνάρτηση Basic Shot Damage του FirstPersonCharacter, όπως φαίνεται στην Εικόνα 11.



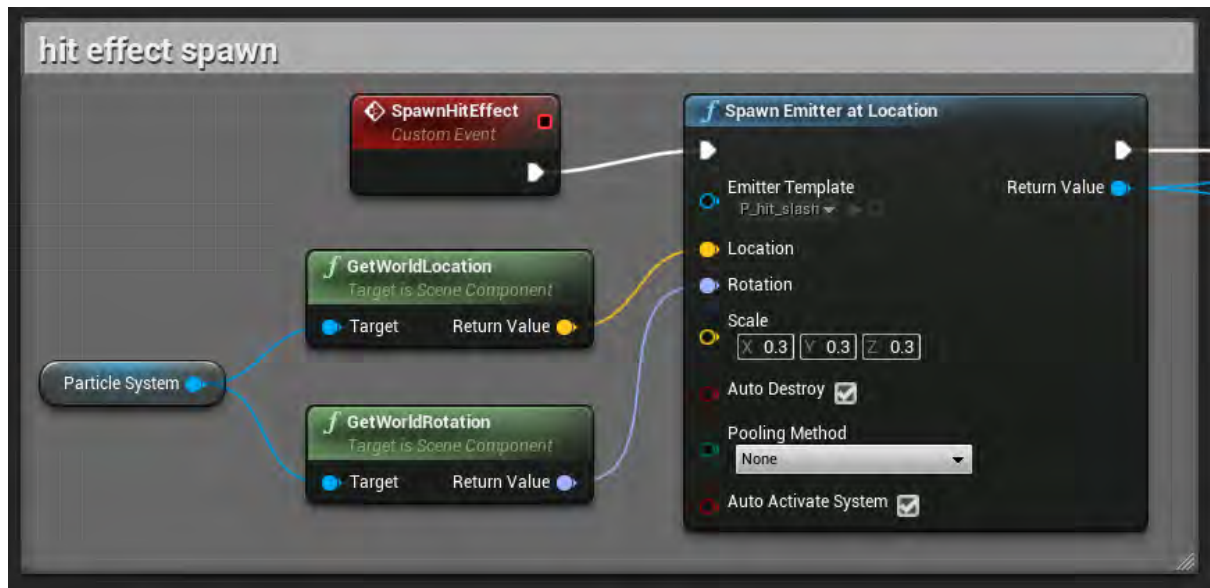
Εικόνα 11 BP\_FutureShot part 6

Αν ο actor είναι BP\_PawnParentAi καλεί τη συνάρτηση SpawnHitEffect και στη συνέχεια τη συνάρτηση Basic Shot Damage του BP\_PawnParentAi, όπως φαίνεται στην Εικόνα 12.



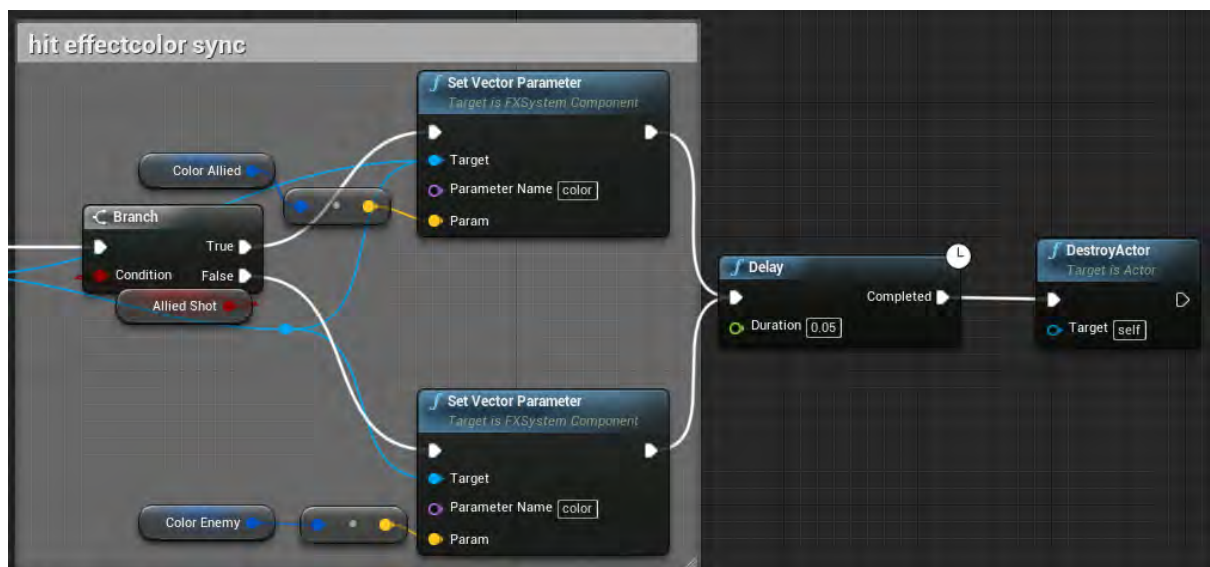
Εικόνα 12 BP\_FutureShot part7

Η συνάρτηση SpawnHitEffect που απεικονίζεται στην Εικόνα 13 εμφανίζει το emitter (P\_hit\_slash) στην τοποθεσία του Particle System με τον προσανατολισμό του.



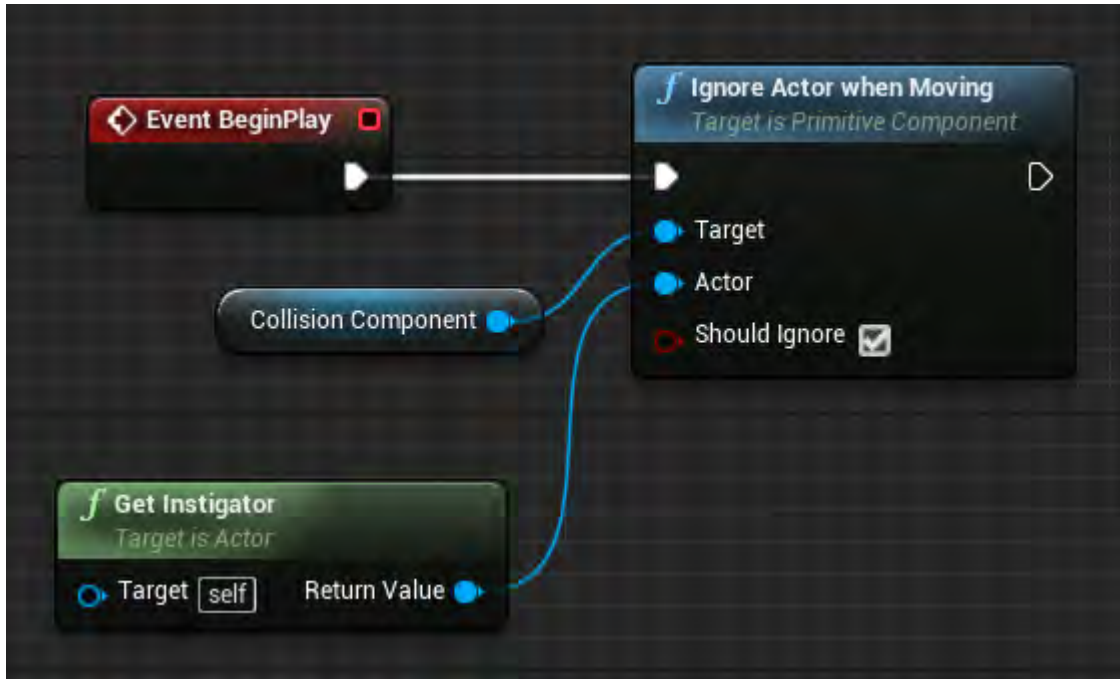
Εικόνα 13 BP\_FutureShot part 8

Γίνεται έλεγχος για να διαπιστωθεί αν η μεταβλητή AlliedShot είναι αληθής και στη συνέχεια τροποποιείται αναλόγως το χρώμα του πεδίου color στον emitter. Μετά από μία μικρή καθυστέρηση για να ολοκληρωθεί το P\_hit\_slash καταστρέφεται ο actor (Εικόνα 14).



Εικόνα 14 BP\_FutureShot part 9

Στο Event BeginPlay αρχικοποιείται ο actor, ώστε να αγνοήσει τη σύγκρουση με τον υποκινητή (instigator) του (Εικόνα 15).



Εικόνα 15 BP\_FutureShot part 10

Το BP\_AlliedShot είναι Child Blueprint Class του BP\_FutureShot και η μεταβλητή AlliedShot είναι true (Εικόνα16).



Εικόνα 16 BP\_AlliedShot part 11

Το BP\_EnemyShot είναι Child Blueprint Class του BP\_FutureShot και η μεταβλητή AlliedShot είναι false (Εικόνα17).

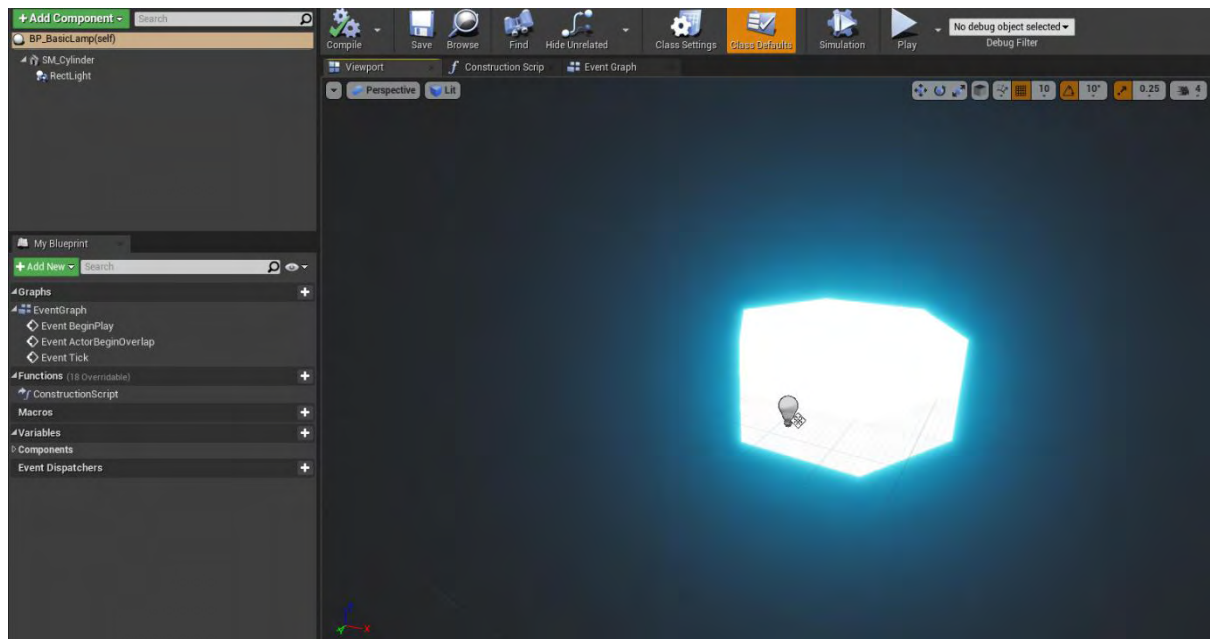


Εικόνα 17 BP\_EnemyShot V 12

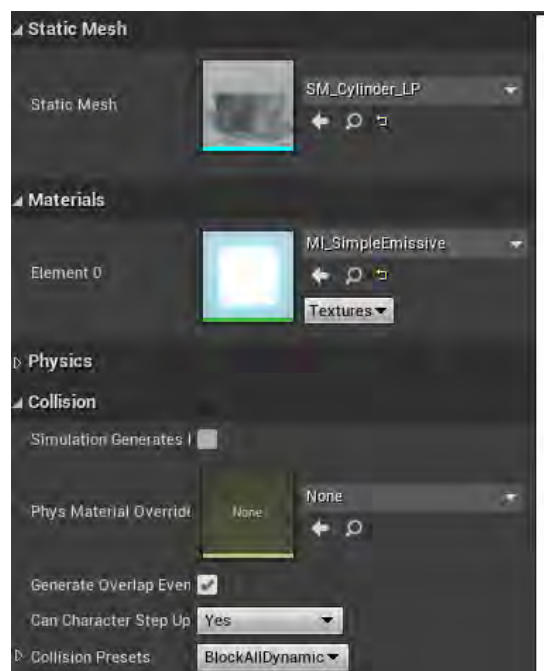


## Ενότητα 5.1.2 BP\_BasicLamp

Το BP\_BasicLamp έχει root component το SM\_Cylinder (Static Mesh Component) και έχει attached component το RectLight (Rect Light Component), όπως φαίνεται στην Εικόνα 18.



Εικόνα 18 BP\_BasicLamp part 1



Εικόνα 19 BP\_BasicLamp part 2

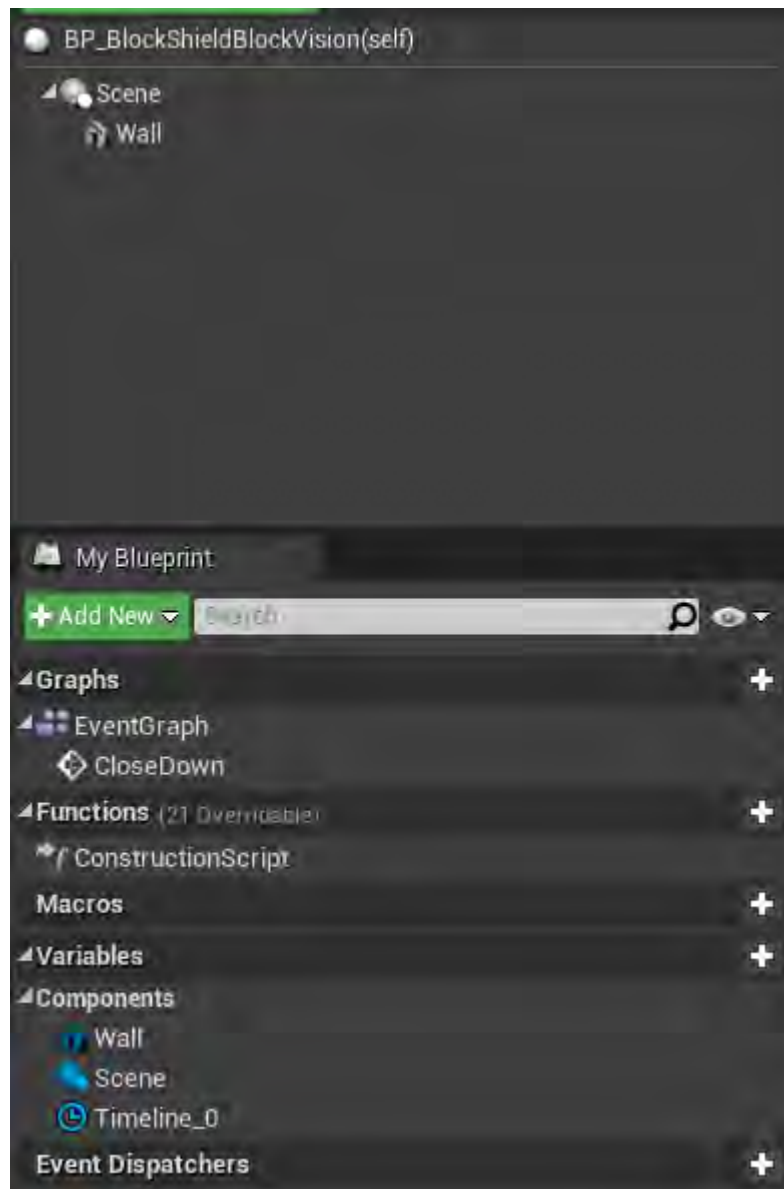


Εικόνα 20 BP\_BasicLamp part 3

### Ενότητα 5.1.3 BP\_BlockShieldBlockVision

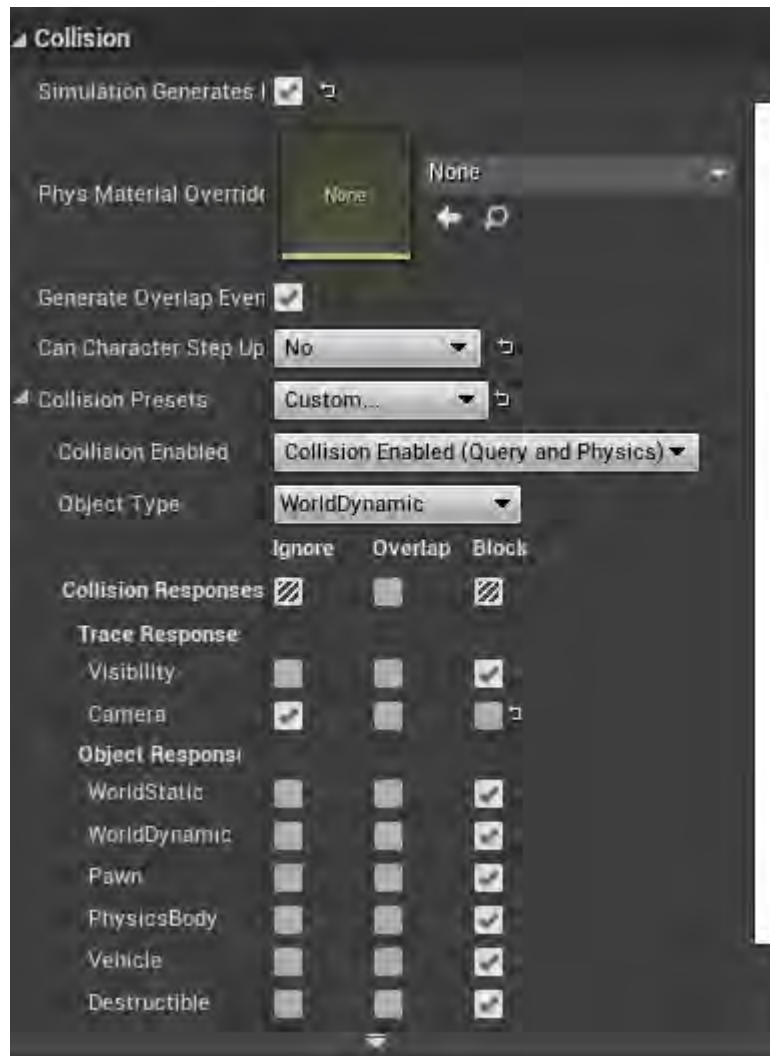
---

Το BP\_BlockShieldBlockVision έχει σαν root component ένα Scene Component με attached component το Wall τύπου Static Mesh Component (Εικόνα 21).



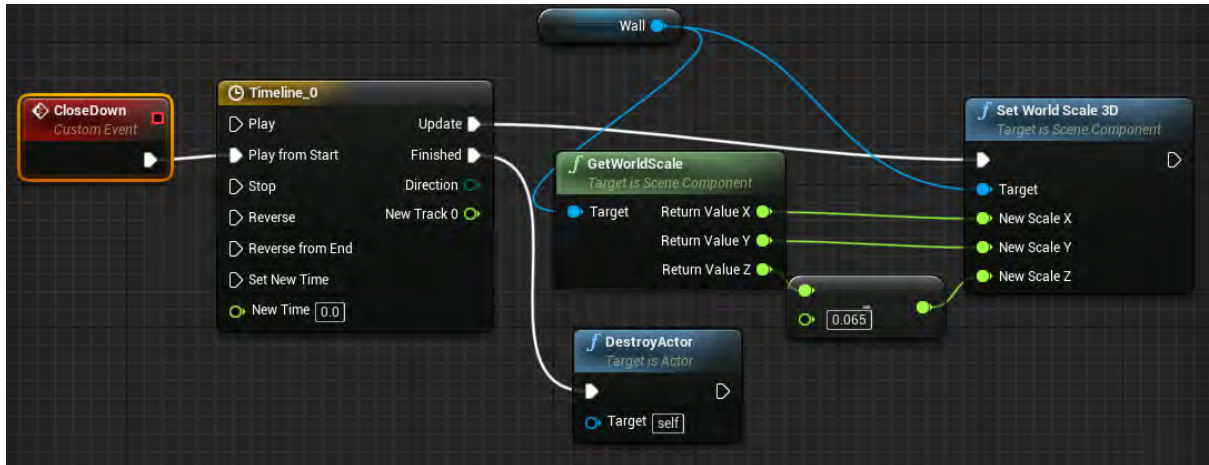
Εικόνα 21 BP\_BlockShieldBlockVision part 1

Οι ρυθμίσεις για το Collision του Wall είναι custom και απεικονίζονται στην Εικόνα 22.



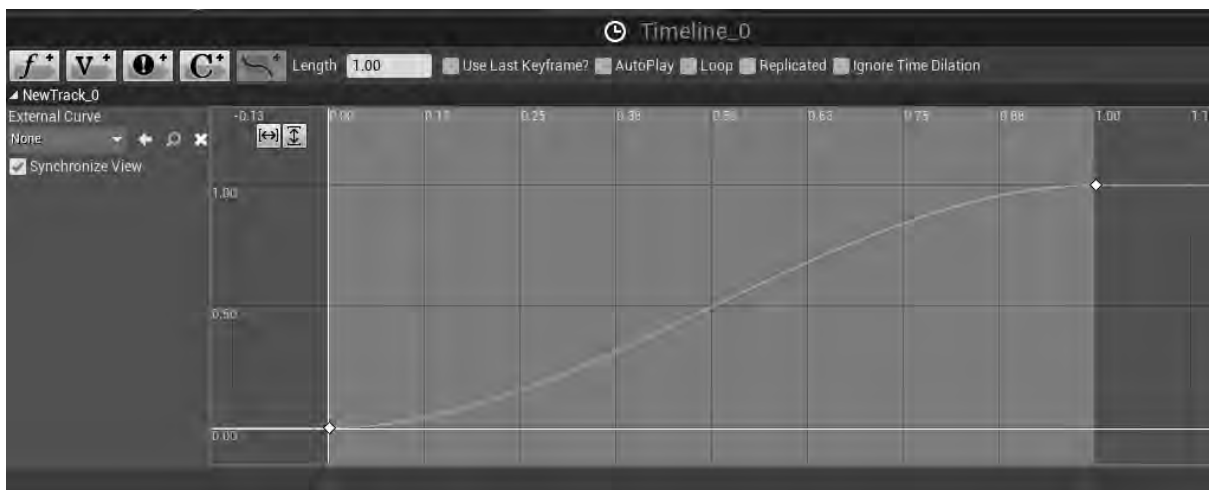
Εικόνα 22 BP\_BlockShieldBlockVision part 2

Η συνάρτηση CloseDown εκτελεί το χρονοδιάγραμμα Timeline\_0 και ενημερώνει καθ' όλη τη διάρκεια του χρονοδιαγράμματος το μέγεθος του Wall (Static Mesh Component) και στο τέλος του χρονοδιαγράμματος καταστρέφει τον actor (Εικόνα 23).



Εικόνα 23 BP\_BlockShieldBlockVision part 3

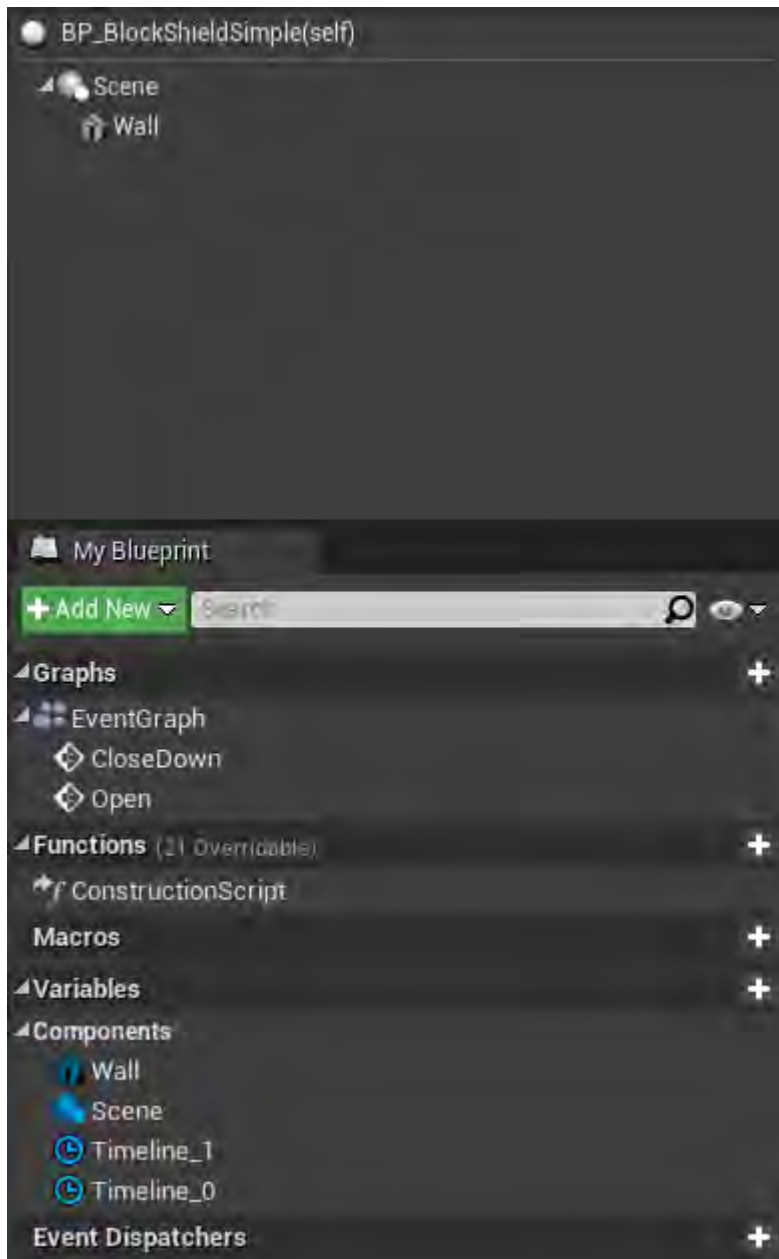
Το Timeline\_0 έχει διάρκεια ενός (1) δευτερολέπτου και ακολουθεί την καμπύλη που φαίνεται παρακάτω (Εικόνα 24).



Εικόνα 24 BP\_BlockShieldBlockVision part 4

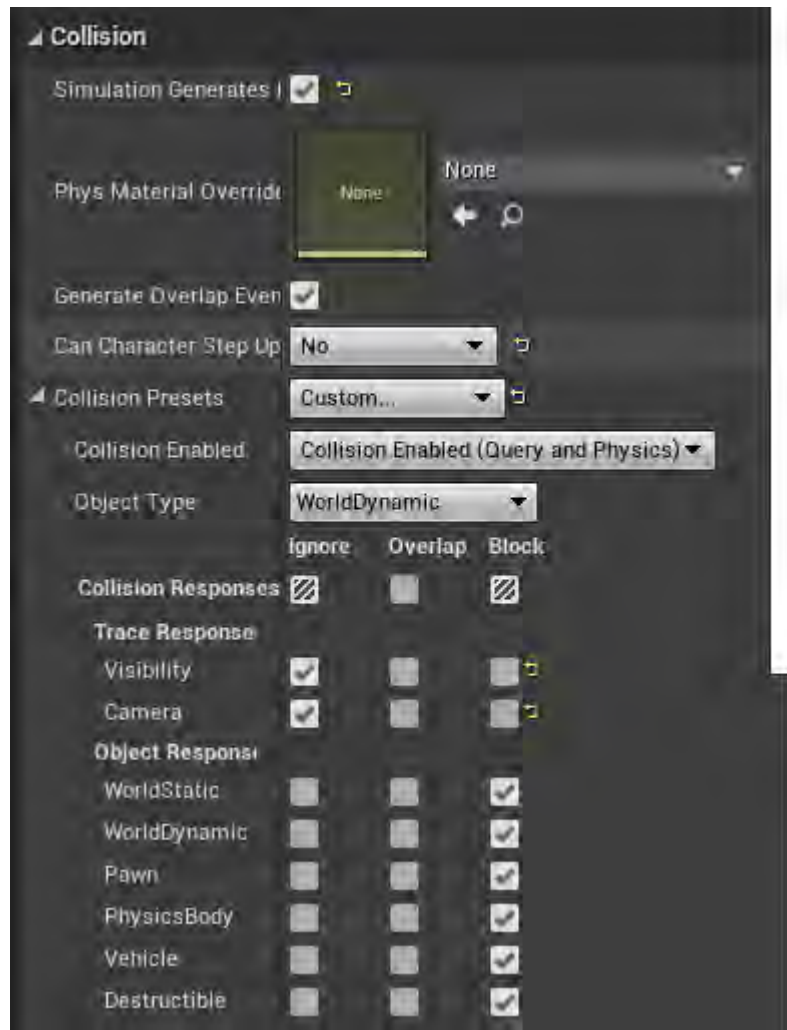
## Ενότητα 5.1.4 BP\_BlockShieldSimple

Το BP\_BlockShieldSimple έχει σαν root component ένα Scene Component με attached component το Wall τύπου Static Mesh Component και υλοποιεί τις συναρτήσεις CloseDown και Open (Εικόνα 25).



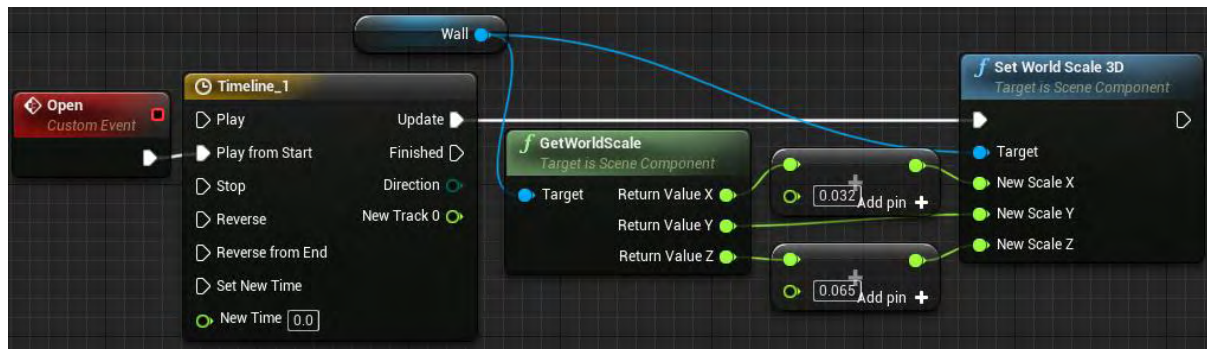
Εικόνα 25 BP\_BlockShieldSimple part 1

Οι ρυθμίσεις για το Collision του Wall είναι custom και απεικονίζονται στην Εικόνα 26.



Εικόνα 26 BP\_BlockShieldSimple part 2

Η συνάρτηση Open τροποποιεί το μέγεθος του Wall σύμφωνα με το Timeline\_1. Το Timeline\_1 είναι ίδιο με το Timeline\_0 και απεικονίζεται στην Εικόνα 27.

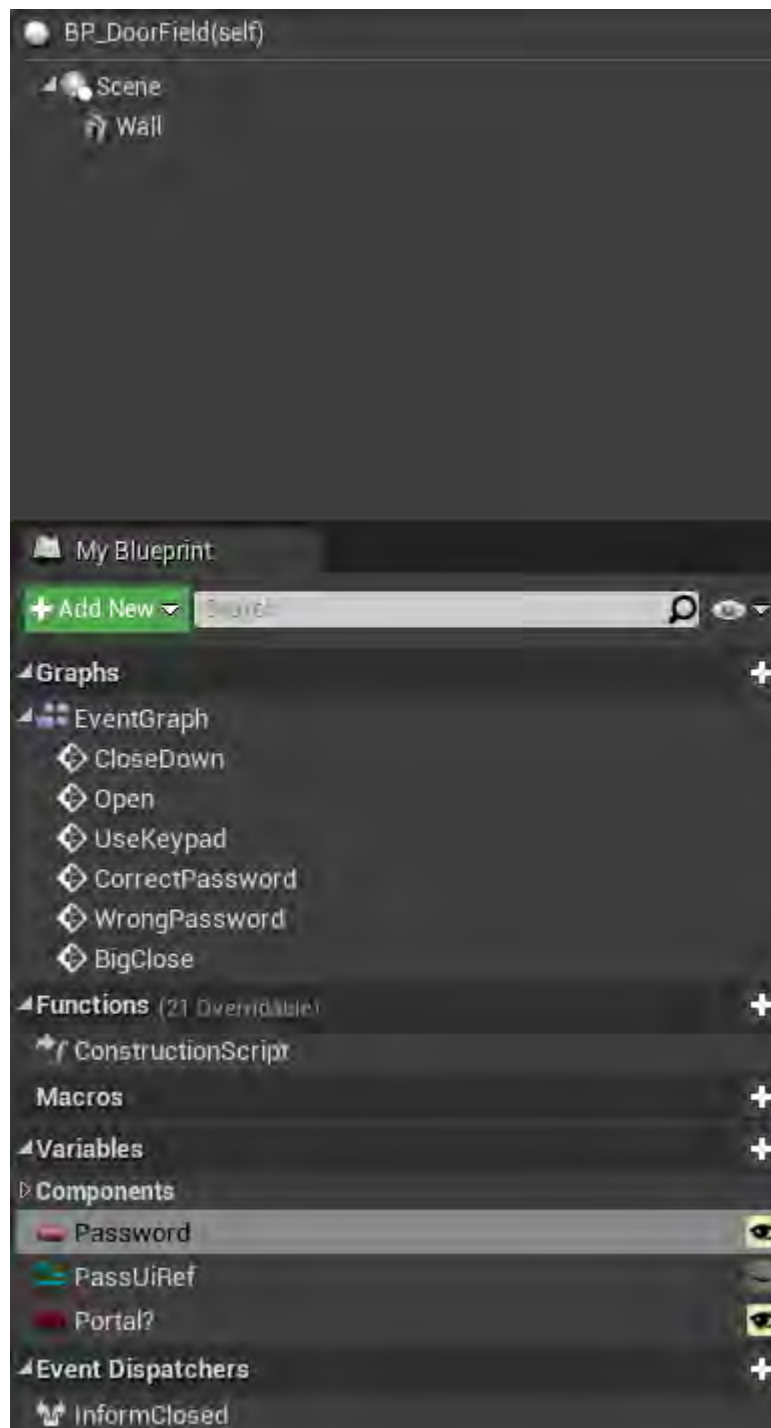


Εικόνα 27 BP\_BlockShieldSimple part 3



## Ενότητα 5.1.5 BP\_DoorField

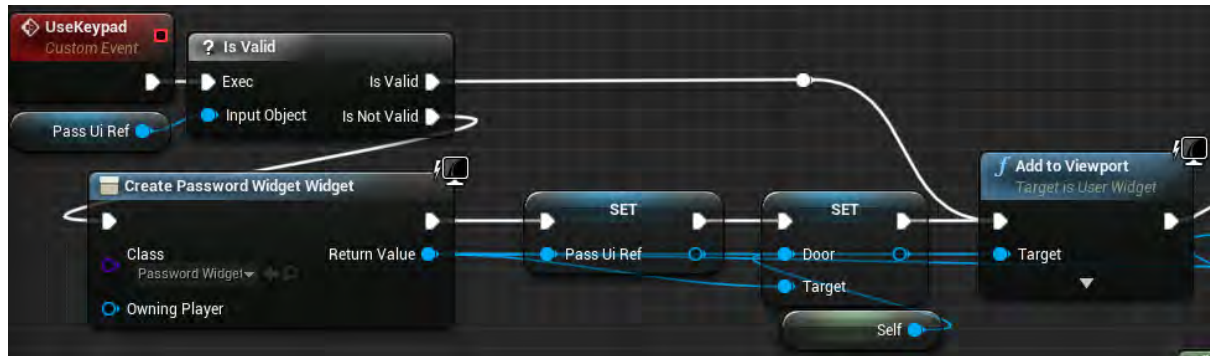
Στην Εικόνα 28 φαίνονται οι συναρτήσεις, οι μεταβλητές, ο Event Dispatcher του actor BP\_Doorfield και η μεταβλητή Wall που είναι τύπου Static Mesh.



Εικόνα 28 BP\_DoorField part1

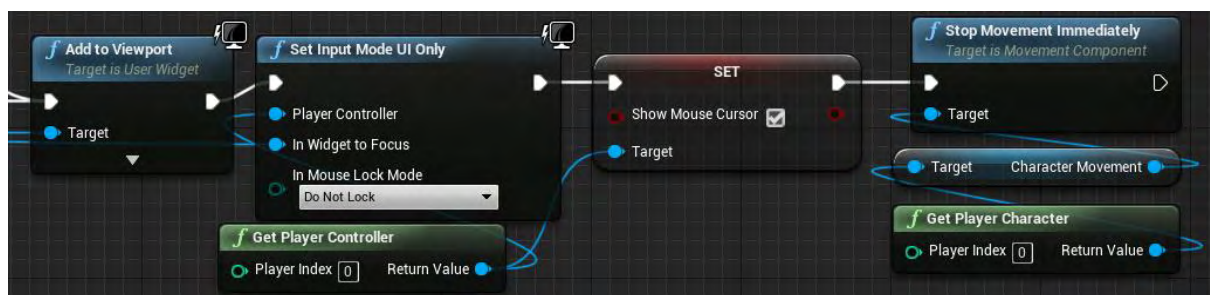


Η συνάρτηση UseKeypad ελέγχει αν υπάρχει ήδη τιμή στη μεταβλητή PassUiRef. Αν δεν υπάρχει, δημιουργεί το Password Widget και αναθέτει στη μεταβλητή το Widget. Στη μεταβλητή Door του Password Widget ανατίθεται ο ίδιος ο actor. Ανεξάρτητα από το αν υπήρχε το Widget, προστίθεται στην οθόνη του χρήστη (Εικόνα 29).



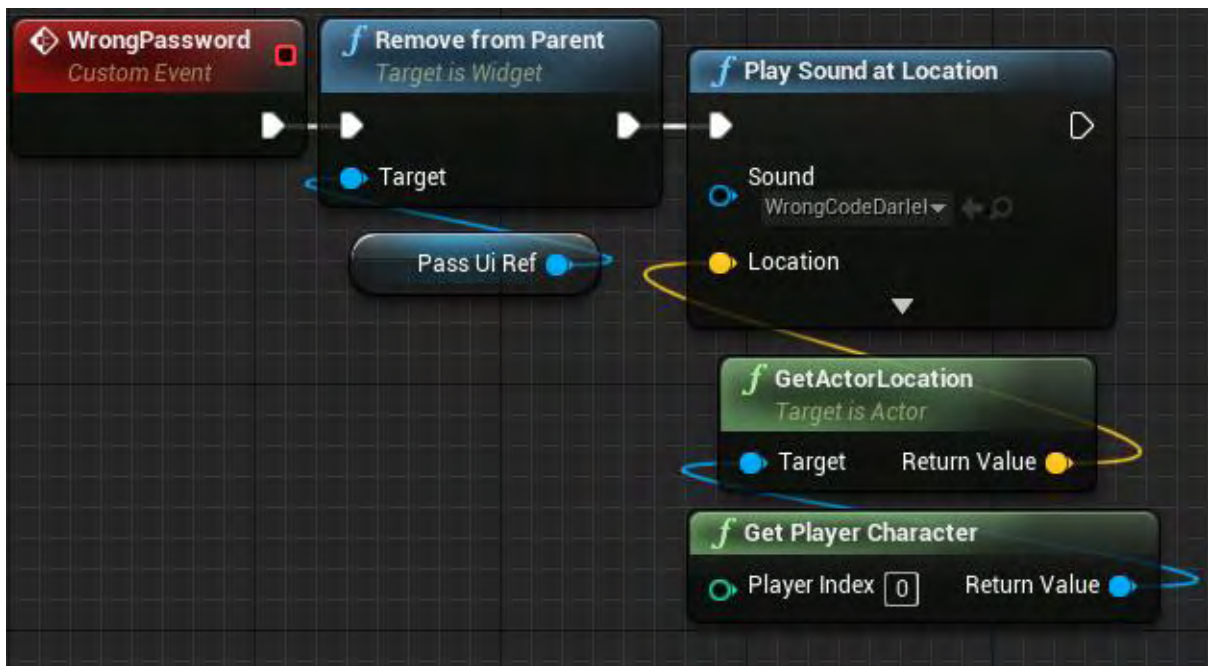
Εικόνα 29 BP\_DoorField part 2

Στη συνέχεια της συνάρτησης UseKeyPad τίθεται το Input Mode του Player Controller μόνο σε UI και επικεντρώνεται στο Password Widget. Εμφανίζεται ο κέρσορας του ποντικιού και παύει η κίνηση του χαρακτήρα (Εικόνα 30).



Εικόνα 30 BP\_DoorField part 3

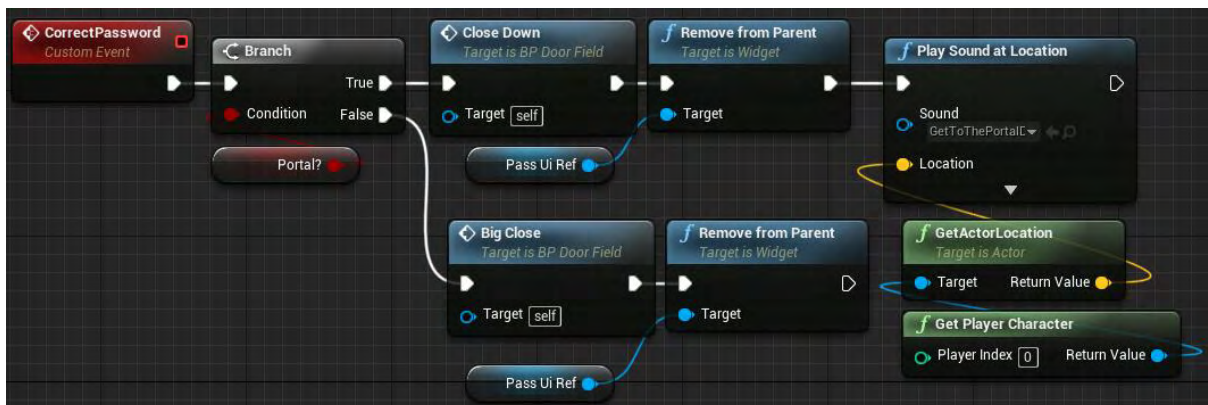
Η συνάρτηση WrongPassword αφαιρεί το Widget της μεταβλητής PassUiRef από την οθόνη του χρήστη και αναπαράγει έναν ήχο στην τοποθεσία του χαρακτήρα (Εικόνα 31).



Εικόνα 31 BP\_DoorField part 4

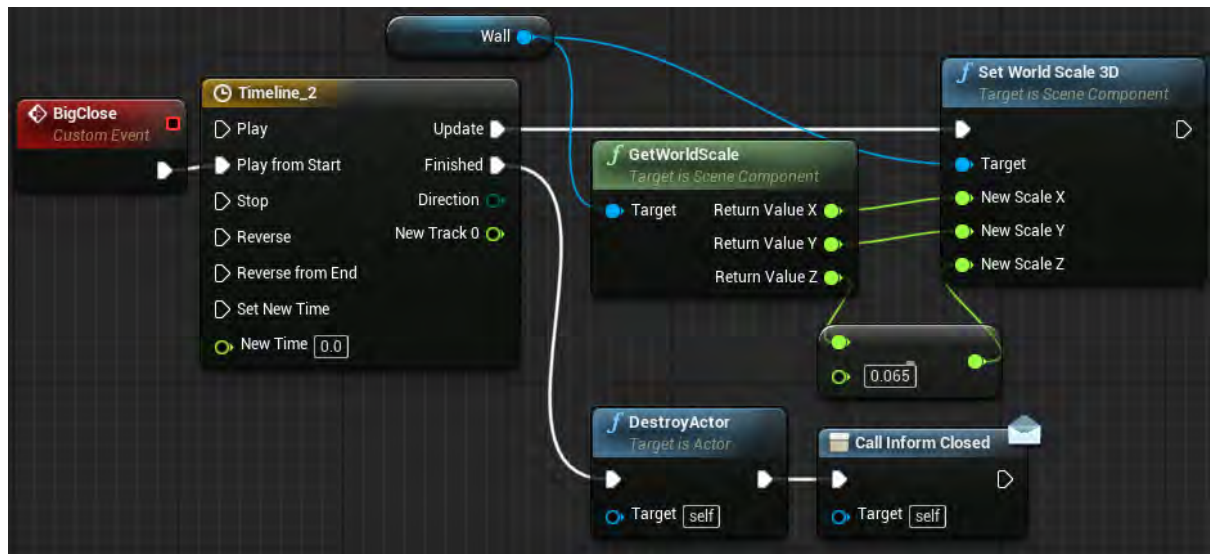
Η συνάρτηση CorrectPassword (Εικόνα 32) ελέγχει την τιμή της μεταβλητής Portal? και αναλόγως καλεί είτε:

- τη συνάρτηση CloseDown, αφαιρεί το Widget της μεταβλητής PassUiRef από την οθόνη του παίκτη και αναπαράγει έναν ήχο.
- τη συνάρτηση BigClose και αφαιρεί το Widget της μεταβλητής PassUiRef από την οθόνη του παίκτη

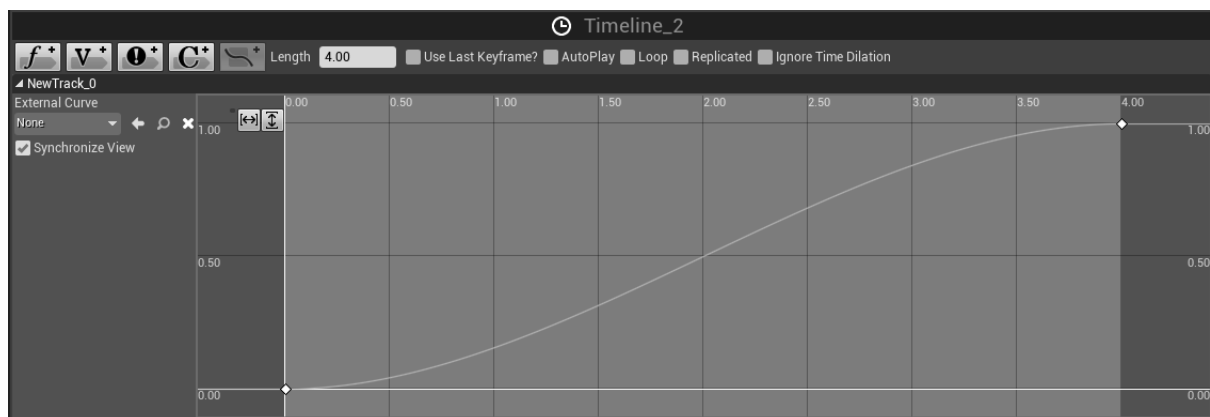


Εικόνα 32 BP\_DoorField part 5

Η συνάρτηση BigClose τροποποιεί το μέγεθος του Wall σύμφωνα με το Timeline\_2 (Εικόνα 33).



Εικόνα 33 BP\_DoorField part 6

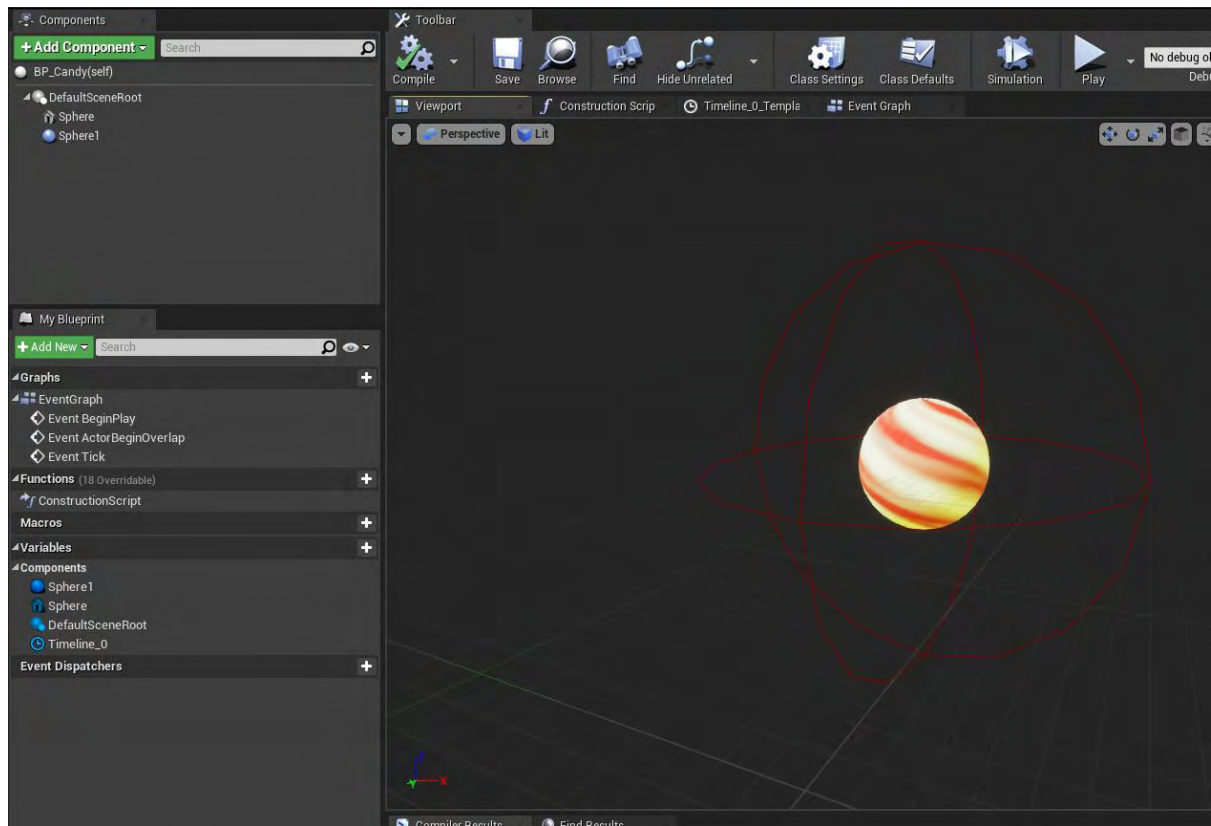


Εικόνα 34 BP\_DoorField part 7

## Ενότητα 5.1.6 BP\_Candy

Όπως φαίνεται στην Εικόνα 35, το BP\_Candy αποτελείται από δύο (2) μέρη:

- το πλέγμα σύγκρουσης (Sphere1) και
- το Static Mesh (Sphere).



Εικόνα 35 BP\_Candy part 1

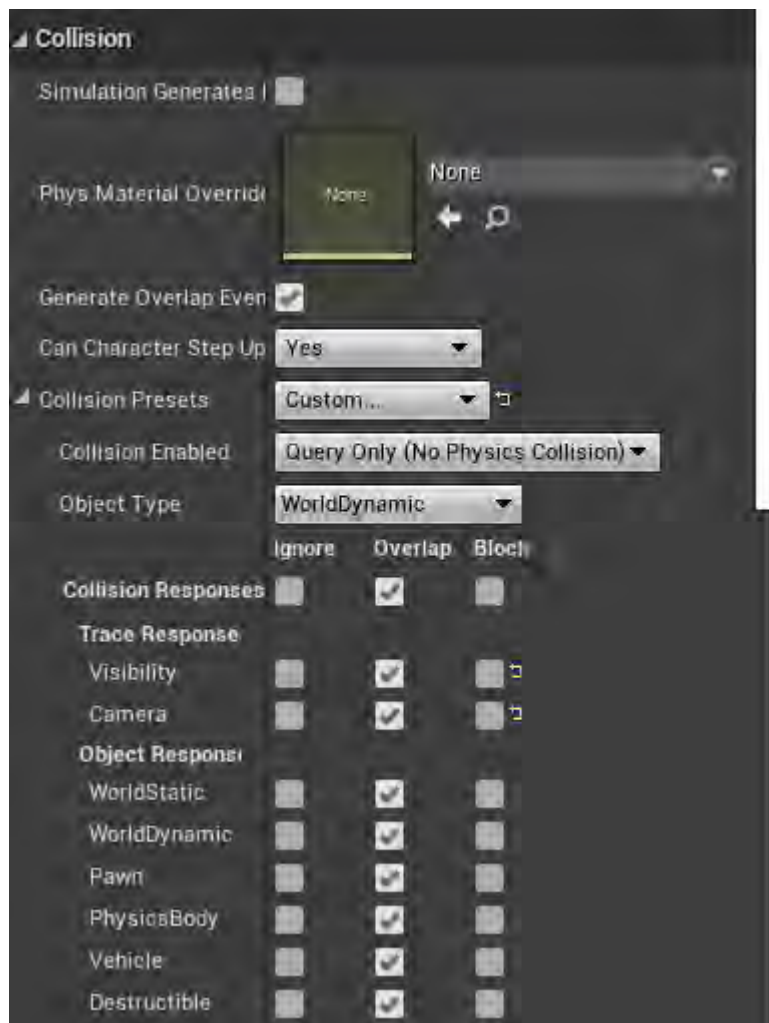
Το Static Mesh είναι από τα βασικά σχήματα Sphere, που περιλαμβάνεται στο Starter Content της Unreal. Το υλικό που χρησιμοποιήθηκε είναι το Mat\_Candy. Οι ρυθμίσεις του πλέγματος συγκρούσεων είναι από τα προκαθορισμένα της Unreal, NoCollision (Εικόνα 36).



Εικόνα 36 BP\_Candy part 2

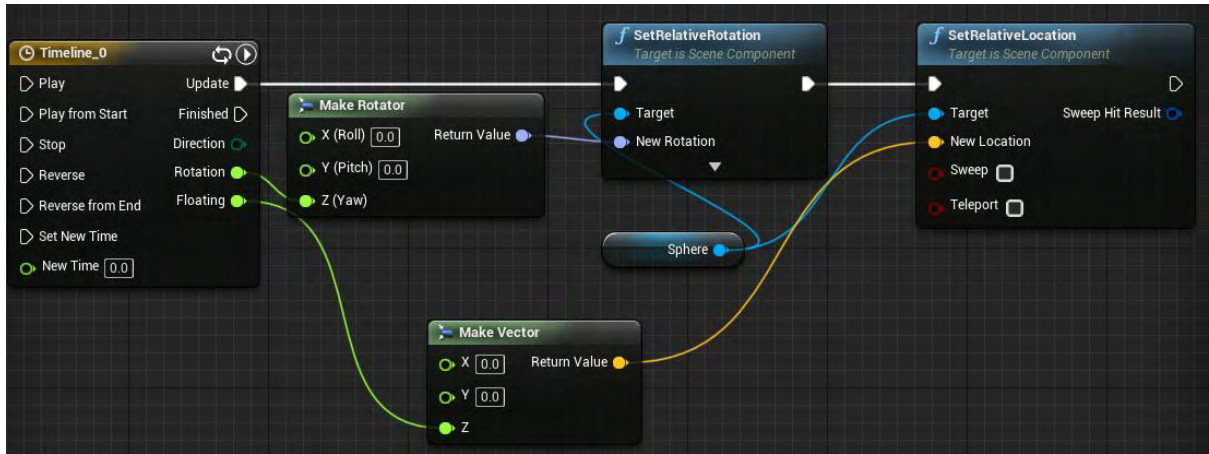
Οι ρυθμίσεις πλέγματος σύγκρουσης του Sphere1 είναι προσαρμοσμένες και φαίνονται στην Εικόνα 37.



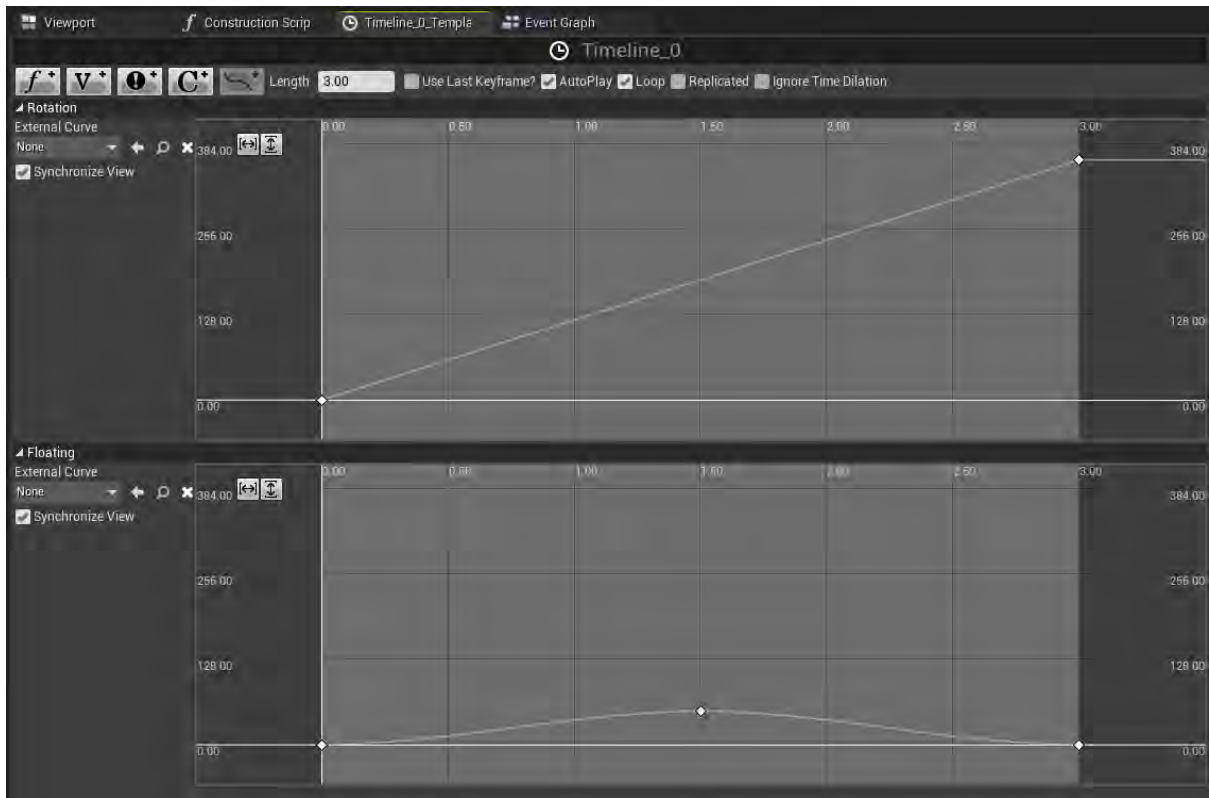


Εικόνα 37 BP\_Candy part 3

Όταν το BP\_Candy δημιουργηθεί στο επίπεδο, το χρονοδιάγραμμα Timeline\_0 ξεκινά να ενημερώνει συνεχώς την σχετική θέση και τον προσανατολισμό του Sphere (Εικόνα 38).



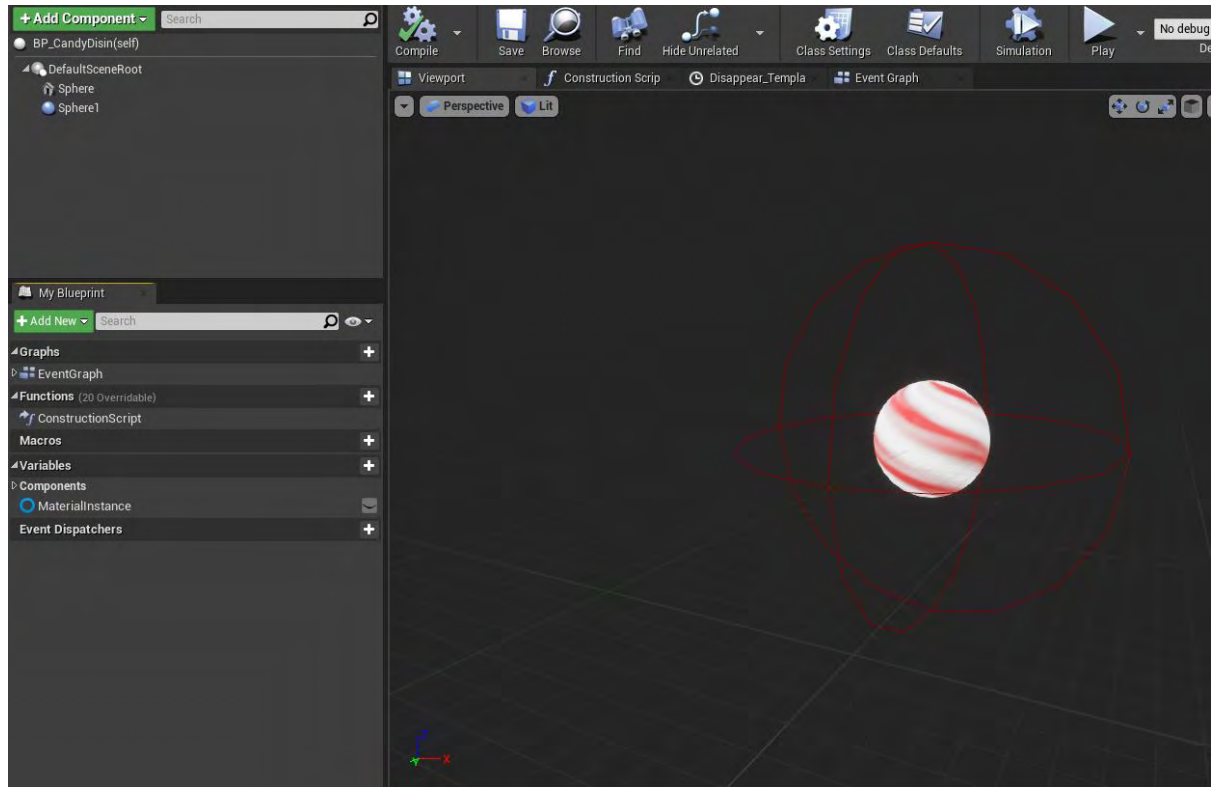
Εικόνα 38 BP\_Candy part 4



Εικόνα 39 BP\_Candy part 5

## Ενότητα 5.1.7 BP\_CandyDisin

Το Static Mesh και το πλέγμα σύγκρουσης του BP\_CandyDisin είναι παρόμοια με του BP\_Candy (Εικόνα 40).



Εικόνα 40 BP\_CandyDisin part 1

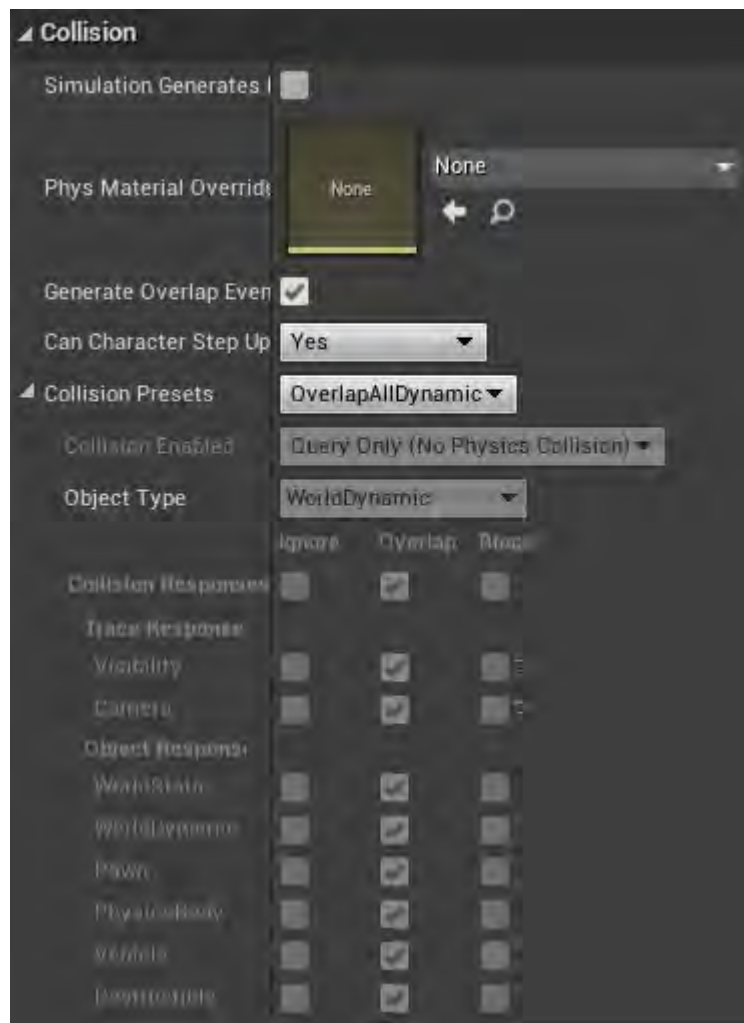
Το υλικό του Static Mesh (Sphere) είναι το Mat\_CandyDisin και είναι η μόνη διαφορά με το Sphere του BP\_Candy (Εικόνα 41).





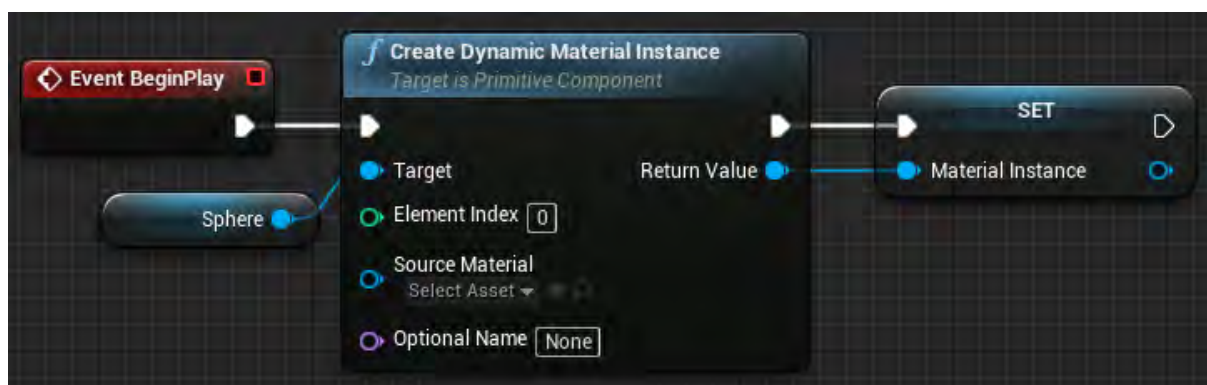
Εικόνα 41 BP\_CandyDisin part 2

Όπως φαίνεται στην Εικόνα 42, οι ρυθμίσεις του πλέγματος σύγκρουσης (Sphere1) είναι από τις προκαθορισμένες της Unreal, OverlapAllDynamic.



Εικόνα 42 BP\_CandyDisin part 3

Όταν το BP\_CandyDisin κατασκευάζεται στο επίπεδο, δημιουργείται μια δυναμική παρουσία υλικού του Sphere και αποθηκεύεται στη μεταβλητή MaterialInstance (Εικόνα 43).

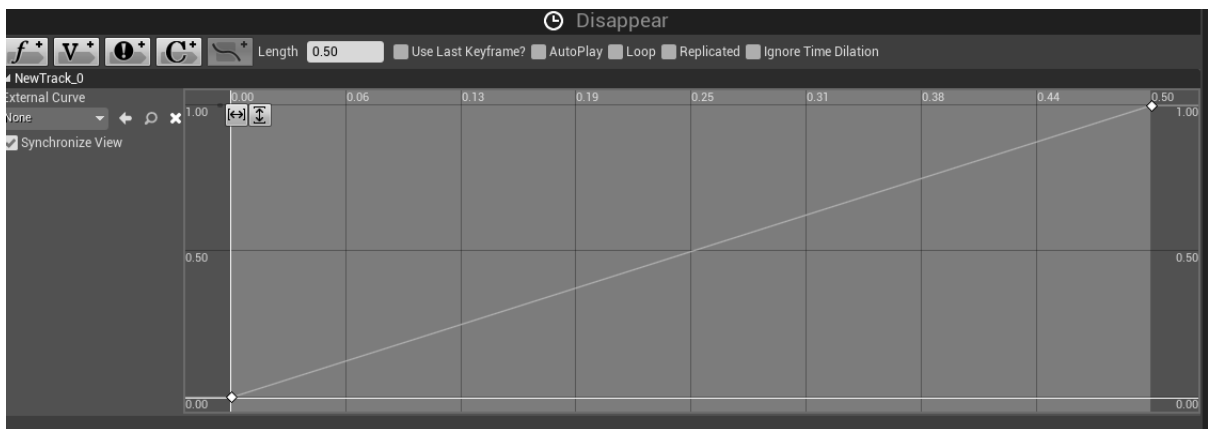


Εικόνα 43 BP\_CandyDisin part 4

Με οποιαδήποτε σύγκρουση με το πλέγμα σύγκρουσης Sphere1 του BP\_CandyDisin, ενεργοποιείται το On Component Begin Overlap (Sphere1). Μόλις συγκρουστεί με το πλέγμα γεωμετρίας του χαρακτήρα του παίκτη, ξεκινά το χρονοδιάγραμμα Disappear (Εικόνα 45) και ενημερώνει την τιμή της μεταβλητής DissolveAmount του MaterialInstance. Όταν ολοκληρωθεί το χρονοδιάγραμμα, καταστρέφεται το BP\_CandyDisin (Εικόνα 44).



Εικόνα 44 BP\_CandyDisin part 5

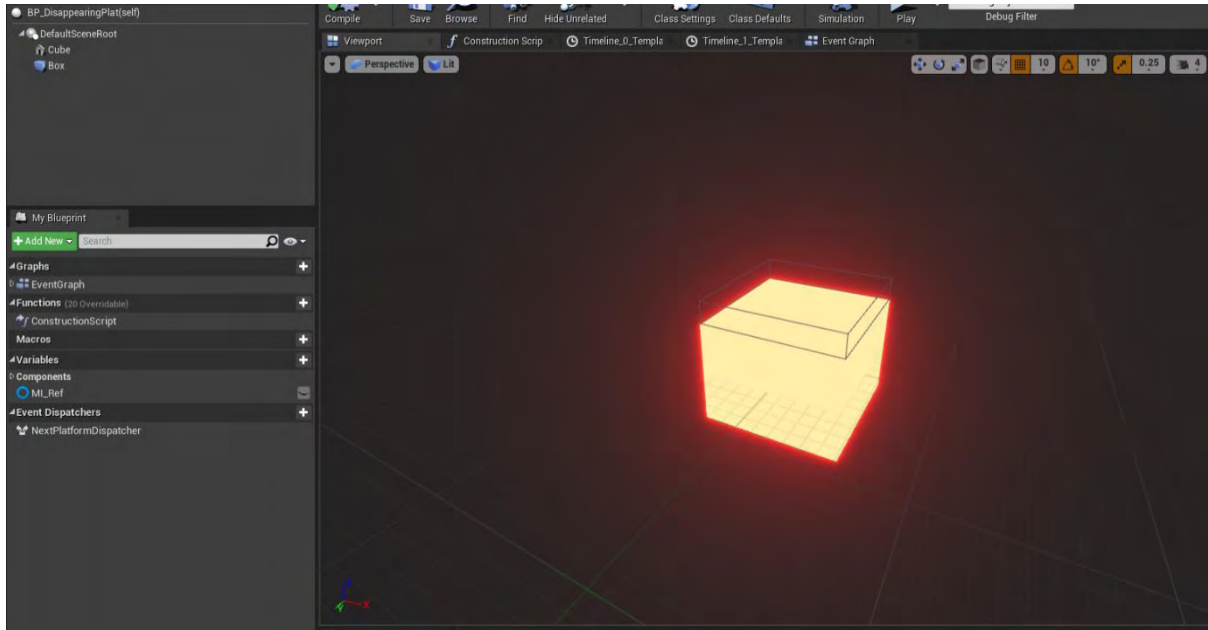


Εικόνα 45 BP\_CandyDisin part 6

## Ενότητα 5.1.8 BP\_DisappearingPlat

Στην Εικόνα 46 φαίνεται το BP\_DisappearingPlat που αποτελείται από δύο (2) μέρη :

- Cube (Static Mesh Component)
- Box (Box Collision)



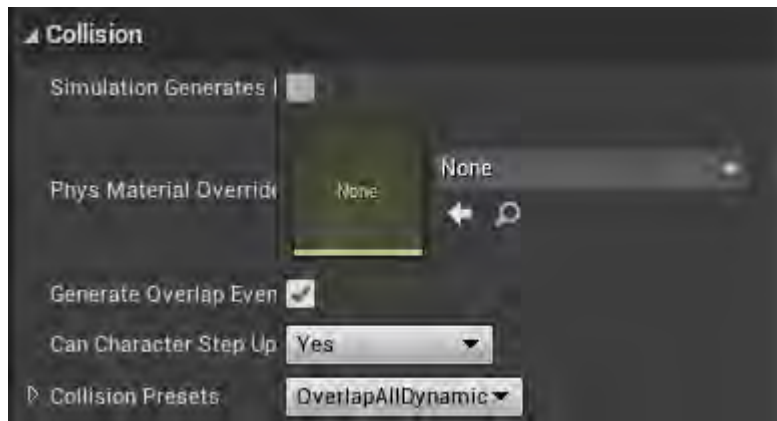
Εικόνα 46 BP\_DisappearingPlat part 1

Το Static Mesh έχει οριστεί ως ένα από τα βασικά σχήματα της Unreal, Cube. Το υλικό του είναι το MI\_DisinRed. Οι ρυθμίσεις του πλέγματος συγκρούσεων είναι από τις προκαθορισμένες της Unreal, η BlockAllDynamic (Εικόνα 47).



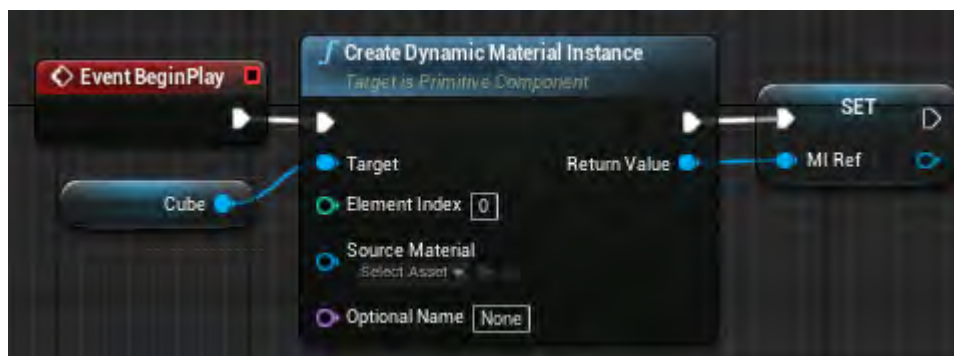
Εικόνα 47 BP\_DisappearingPlat part 2

Το Box (πλέγμα σύγκρουσης) έχει την προκαθορισμένη ρύθμιση OverlapAllDynamic (Εικόνα 48).



Εικόνα 48 BP\_DisappearingPlat part 3

Όταν το BP\_DisappearingPlat κατασκευάζεται στο επίπεδο, δημιουργείται μια δυναμική παρουσία υλικού του Cube και αποθηκεύεται στη μεταβλητή MI Ref (Εικόνα 49).

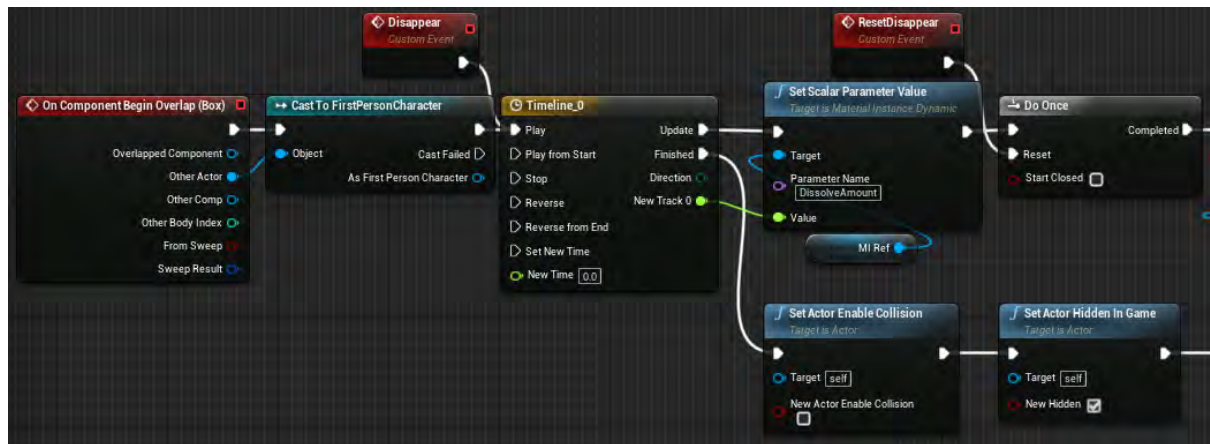


Εικόνα 49 BP\_DisappearingPlat part 4

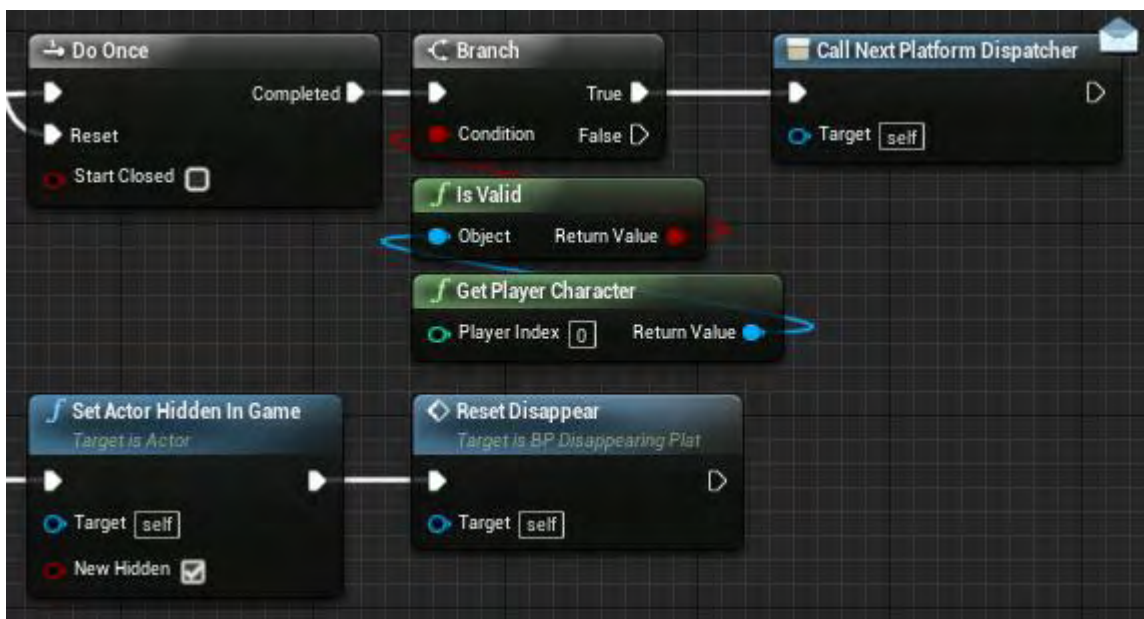
Όπως φαίνεται στην Εικόνα 50 και την Εικόνα 51, το On Component Begin Overlap (Box) ενεργοποιείται σε οποιαδήποτε σύγκρουση του Box. Όταν η σύγκρουση είναι με τον χαρακτήρα του παίκτη, ενεργοποιείται από την αρχή το χρονοδιάγραμμα (Εικόνα 53) Timeline\_0, το οποίο ενημερώνει την τιμή της μεταβλητής MI\_Ref. Κατά τη διάρκεια του χρονοδιαγράμματος εκτελείται μια (1) φορά, λόγω του κόμβου Do Once, ο έλεγχος για την ύπαρξη του χαρακτήρα του παίκτη και αν υπάρχει, καλεί τον Event Dispatcher (NextPlatformDispatcher). Όταν ολοκληρωθεί πλέον το Timeline\_0, απενεργοποιείται η δυνατότητα σύγκρουσης του BP\_DisappearingPlat και γίνεται αόρατο στο παιχνίδι. Η συνάρτηση



ResetDisappear δίνει τη δυνατότητα στον κόμβο Do Once να εκτελεστεί ξανά. Τα παραπάνω μπορεί να εκτελεστούν και καλώντας τη συνάρτηση Disappear.



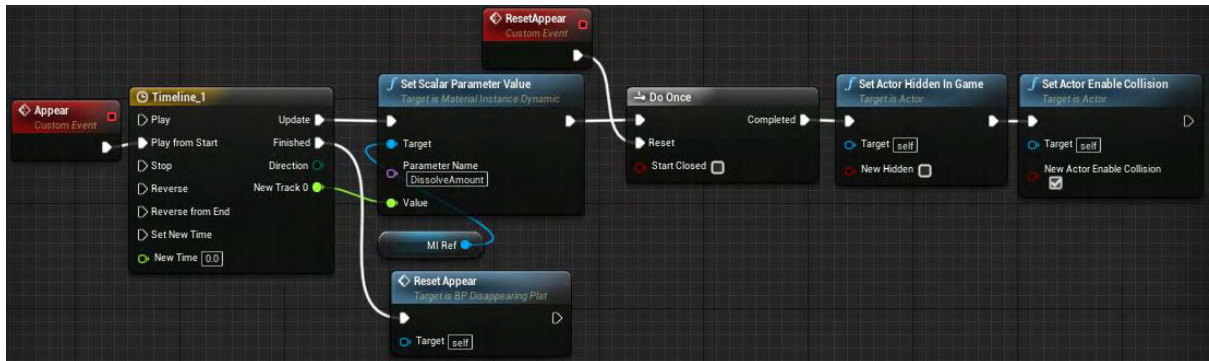
Εικόνα 50 BP\_DisappearingPlat part 5



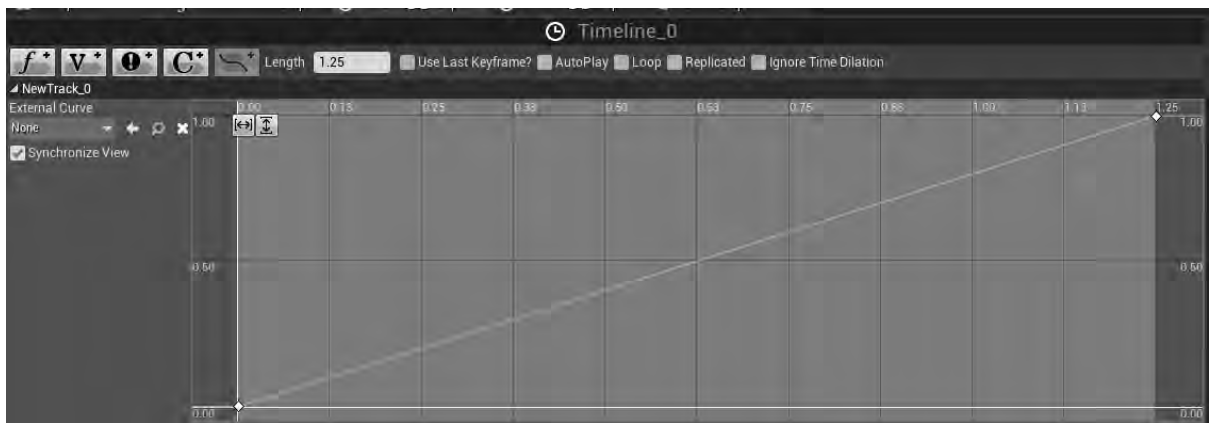
Εικόνα 51 BP\_DisappearingPlat part6

Η συνάρτηση Appear καλεί την εκτέλεση του χρονοδιαγράμματος Timeline\_1 (Εικόνα 54) από την αρχή. Το χρονοδιάγραμμα ενημερώνει την τιμή της μεταβλητής DissolveAmount του MI\_Ref. Κατά τη διάρκεια του χρονοδιαγράμματος εκτελούνται μια (1) φορά, λόγω του Do Once, οι κόμβοι Set Actor Hidden in Game και Set Actor Enable Collision. Ο πρώτος εμφανίζει το BP\_DisappearingPlat και ο δεύτερος ενεργοποιεί την δυνατότητα σύγκρουσης. Αφού ολοκληρωθεί το

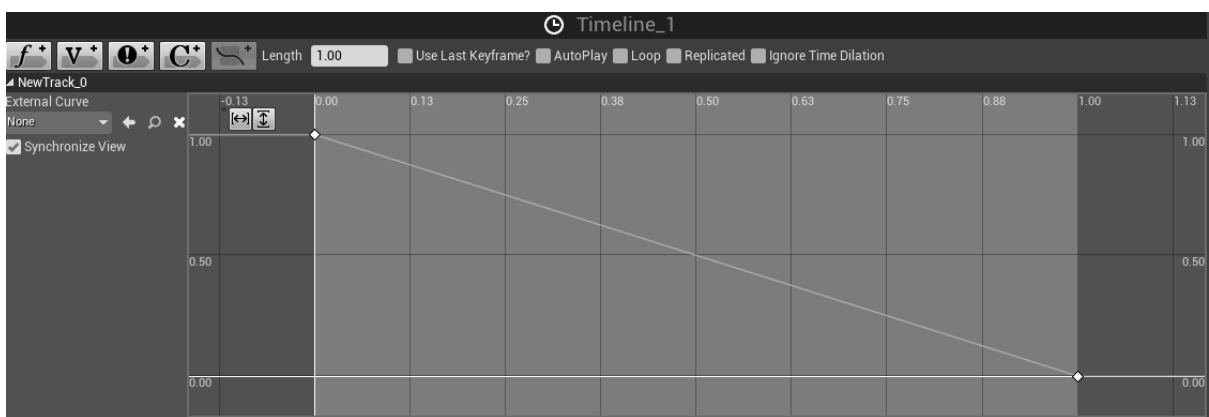
χρονοδιάγραμμα, καλείται η συνάρτηση ResetAppear η οποία δίνει τη δυνατότητα στον κόμβο Do Once να επανεκτελεστεί.



Εικόνα 52 BP\_DisappearingPlat part 7



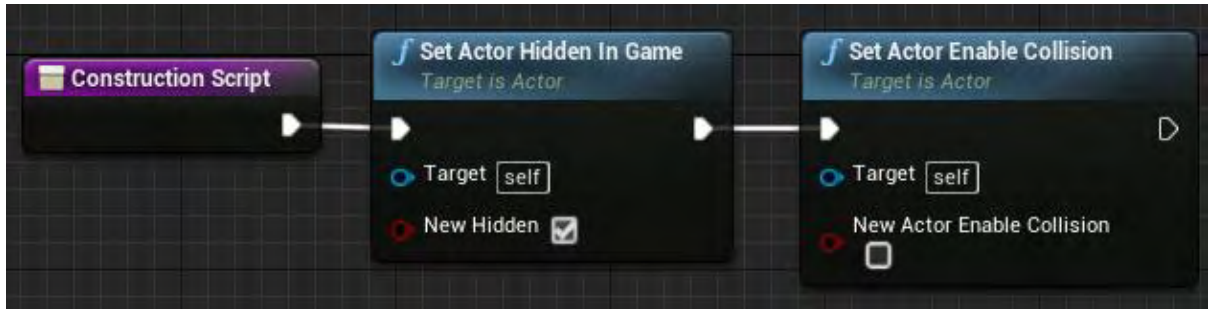
Εικόνα 53 BP\_DisappearingPlat part 8



Εικόνα 54 BP\_DisappearingPlat part 9

Κατά την κατασκευή του, ο BP\_DisappearingPlat γίνεται αόρατος στο παιχνίδι και απενεργοποιείται η δυνατότητα σύγκρουσης (Εικόνα 55).





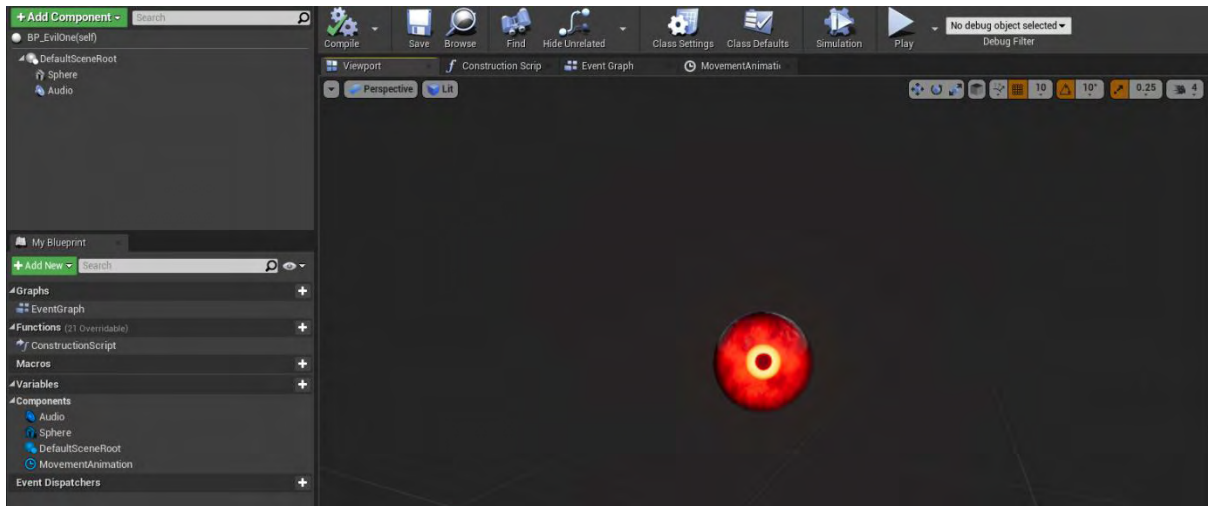
Εικόνα 55 BP\_DisappearingPlat part 10

## Ενότητα 5.1.9 BP\_ EvilOne

Το BP\_EvilOne (Εικόνα 56) αποτελείται από τα εξής μέρη:

- Sphere (Static Mesh Component)
- Audio (Audio Component)

Το Audio δε θα αναλυθεί περαιτέρω εφόσον δεν έχουν αλλάξει οι αρχικές του ρυθμίσεις.



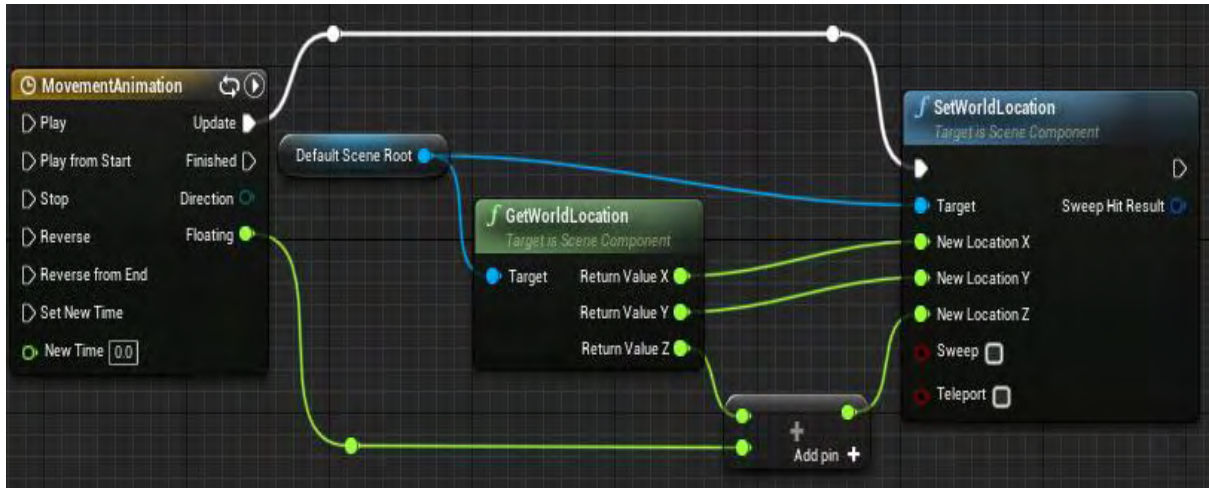
Εικόνα 56 BP\_EvilOne part1

Το Static Mesh Component είναι ένα από τα βασικά προϋπάρχοντα σχήματα. Το υλικό του Sphere είναι το MI\_EvilOrb. Οι ρυθμίσεις του πλέγματος σύγκρουσης είναι από τα προκαθορισμένα, το BlockAllDynamic (Εικόνα 57).

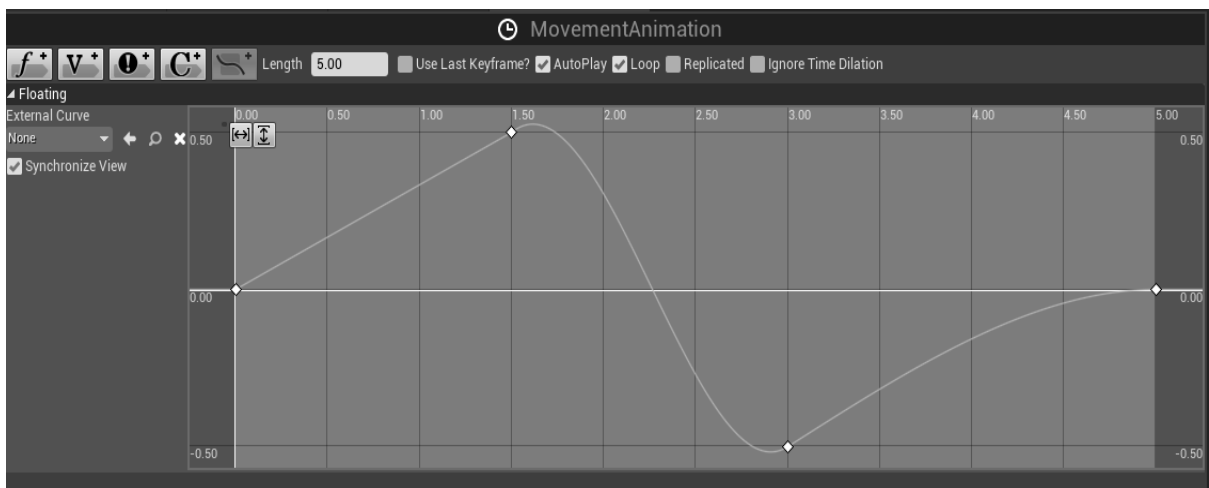


Εικόνα 57 BP\_EvilOne part2

Μόλις δημιουργηθεί το BP\_EvilOne, ενεργοποιείται το χρονοδιάγραμμα MovementAnimation (Εικόνα 59), το οποίο ενημερώνει συνεχώς την τοποθεσία του BP\_EvilOne σε σχέση με την τοποθεσία που βρίσκεται (Εικόνα 58).



Εικόνα 58 BP\_EvilOne part3

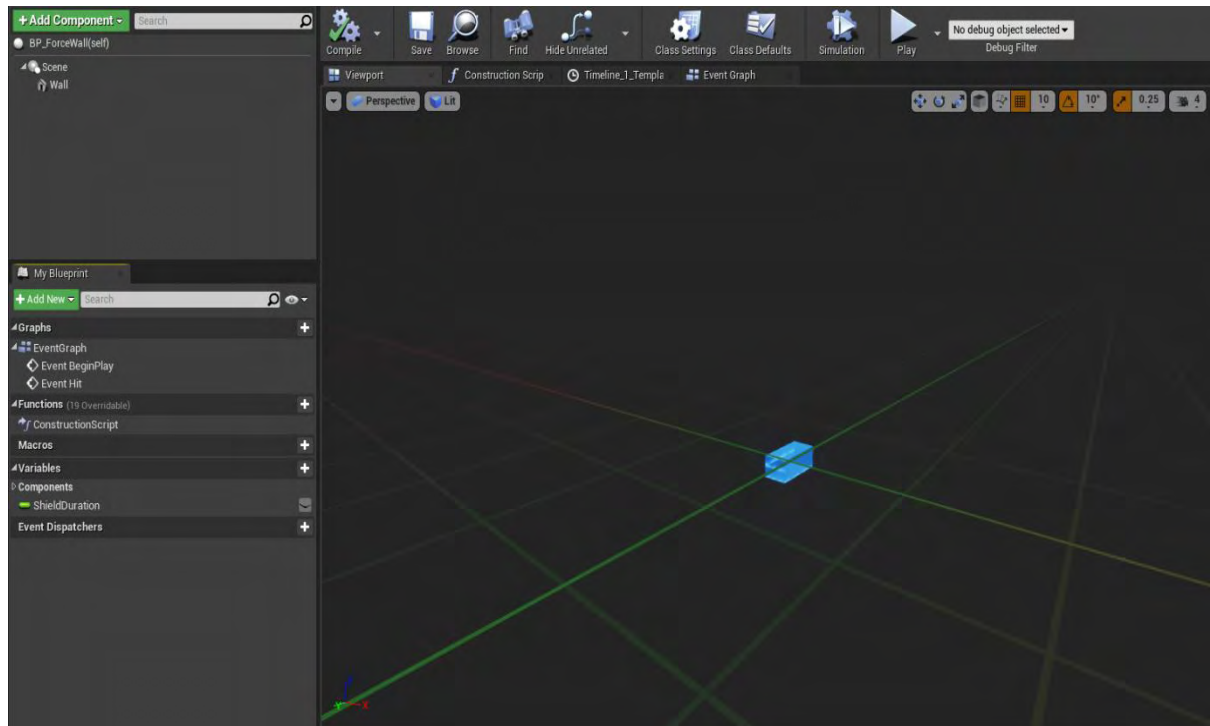


Εικόνα 59 BP\_EvilOne part4

## Ενότητα 5.1.10 BP\_ForceWall

---

Το BP\_ForceWall έχει μόνο το Wall τύπου Static Mesh Component (Εικόνα 60).



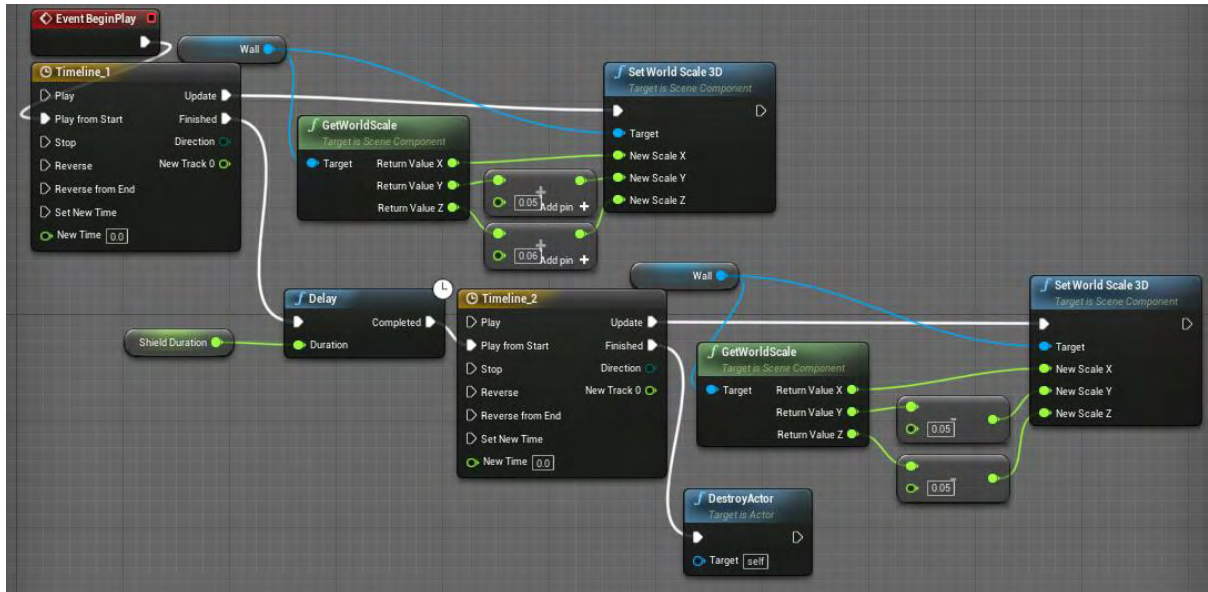
Εικόνα 60 BP\_ForceWall part 1

Το Wall έχει το Static Mesh 1M\_Cube. Το υλικό του είναι το Mat\_ShieldFinal. Οι ρυθμίσεις σύγκρουσης του Wall είναι προσαρμοσμένες όπως φαίνεται στην Εικόνα 61.

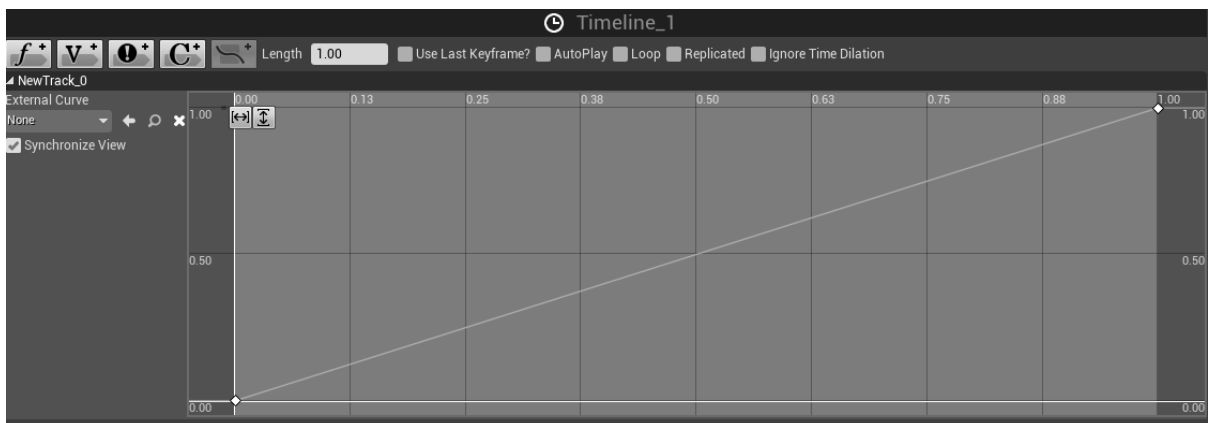


Εικόνα 61 BP\_ForceWall part 2

Μόλις το BP\_ForceWall δημιουργηθεί μέσα σε κάποιο επίπεδο, καλείται το Event BeginPlay και ξεκινά από την αρχή το χρονοδιάγραμμα Timeline\_1 (Εικόνα 63), το οποίο ενημερώνει το μέγεθος του Wall. Όταν ολοκληρωθεί το Timeline\_1, ξεκινά με μια χρονοκαθυστέρηση πέντε δευτερολέπτων από την αρχή το Timeline\_2 (όμοιο με Timeline\_1) που τροποποιεί το μέγεθος του Wall. Στο τέλος του Timeline\_2, το BP\_ForceWall καταστρέφεται (Εικόνα 62).



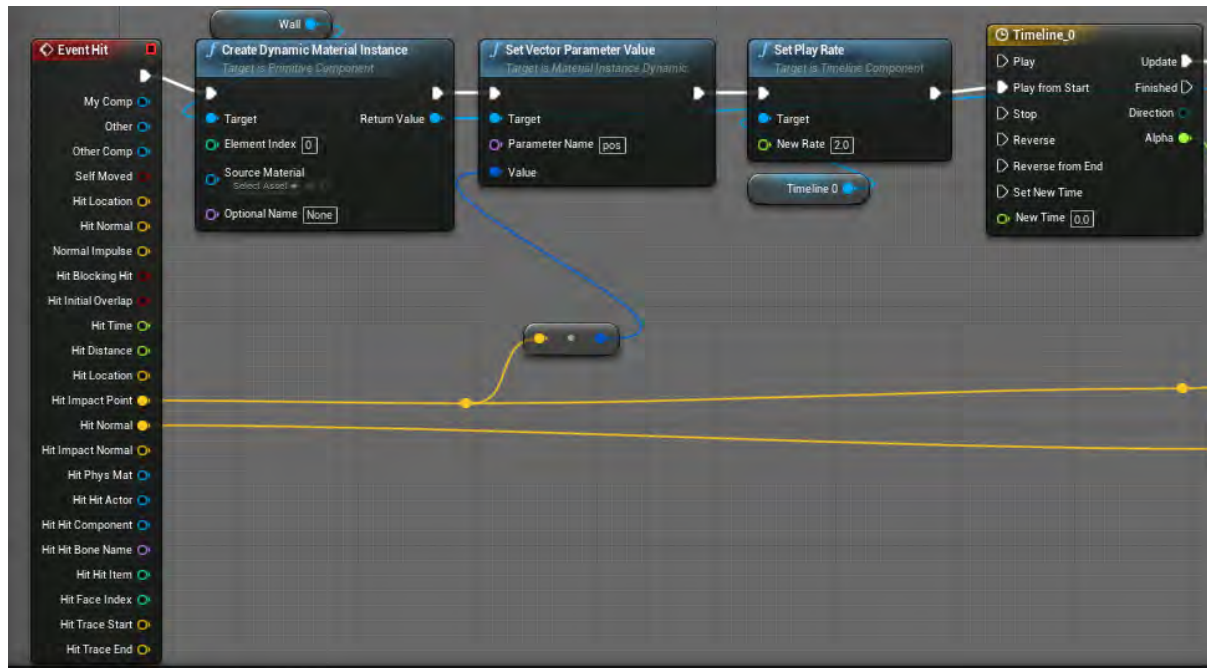
Εικόνα 62 BP\_ForceWall part 3



Εικόνα 63 BP\_ForceWall part 4

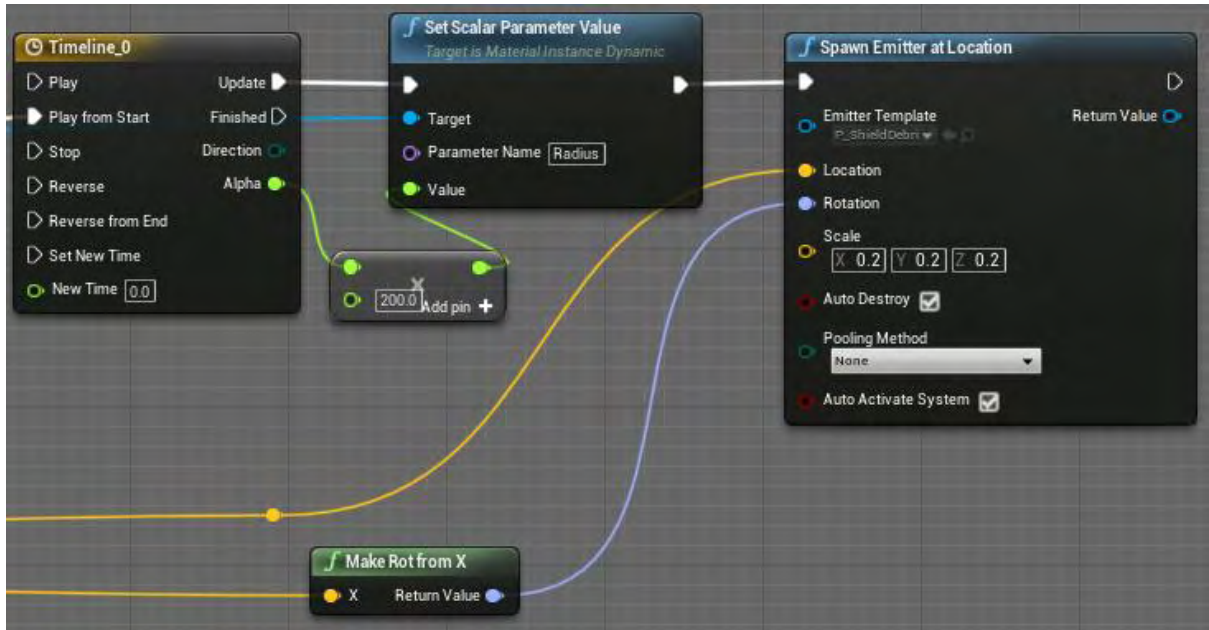


Μόλις στο BP\_ForceWall συγκρουστεί κάποιο actor, ενεργοποιείται το Event Hit. Στη συνέχεια δημιουργείται μια δυναμική παρουσία του υλικού του Wall και τροποποιείται η τιμή της μεταβλητής pos σύμφωνα με την τοποθεσία του σημείου σύγκρουσης. Κατόπιν το χρονοδιάγραμμα Timeline\_0 ξεκινά από την αρχή (Εικόνα 64).



Εικόνα 64 BP\_ForceWall part 5

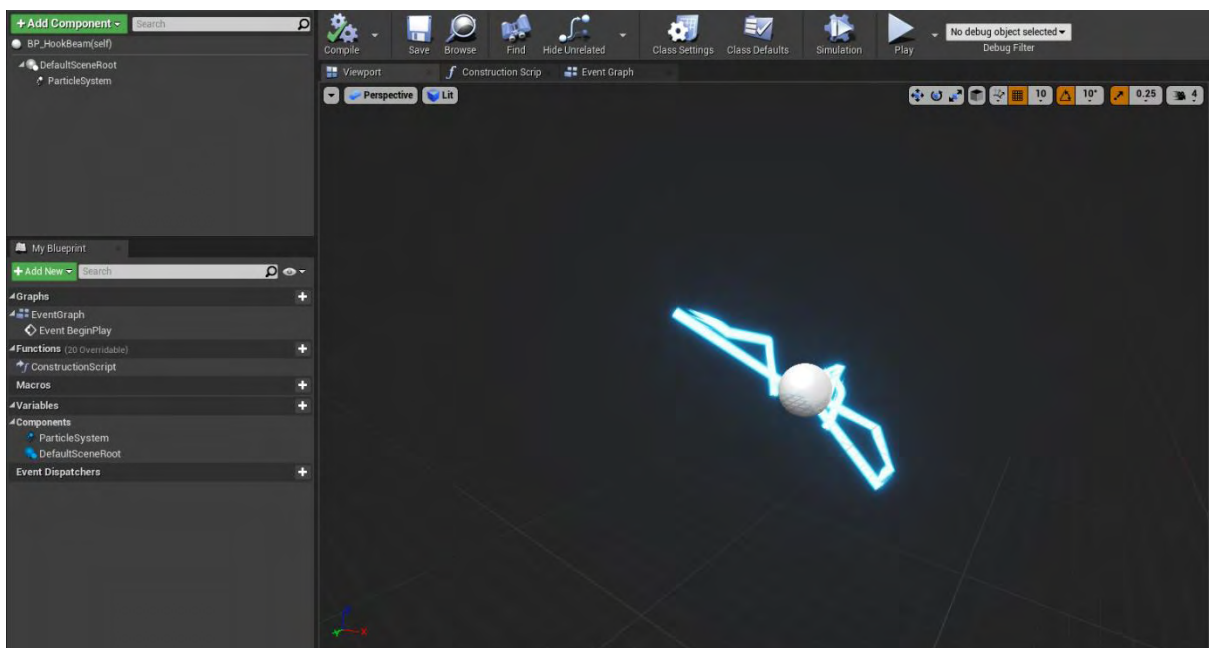
Η τιμή της μεταβλητής Radius της δυναμικής παρουσίας υλικού τροποποιείται σύμφωνα με το χρονοδιάγραμμα Timeline\_0. Ακολούθως παράγεται το σύστημα σωματιδίων P\_ShieldDebri στην τοποθεσία σύγκρουσης με το σωστό προσανατολισμό (Εικόνα 65).



Εικόνα 65 BP\_ForceWall part 6

## Ενότητα 5.1.11 BP\_HookBeam

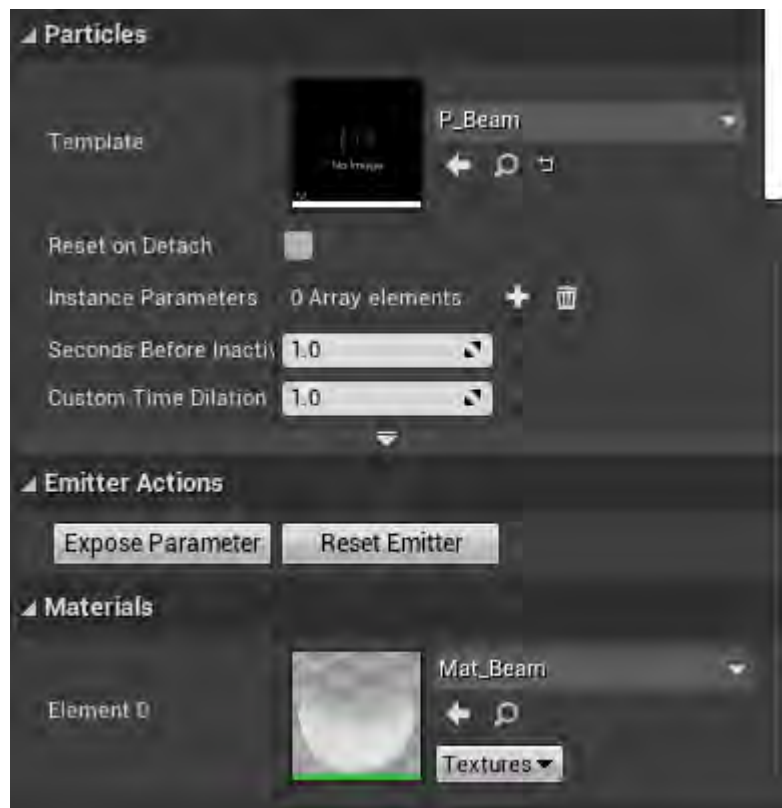
Όπως φαίνεται στην Εικόνα 66, το BP\_HookBeam αποτελείται από το ParticleSystem (Particle System Component).



Εικόνα 66 BP\_HookBeam part 1

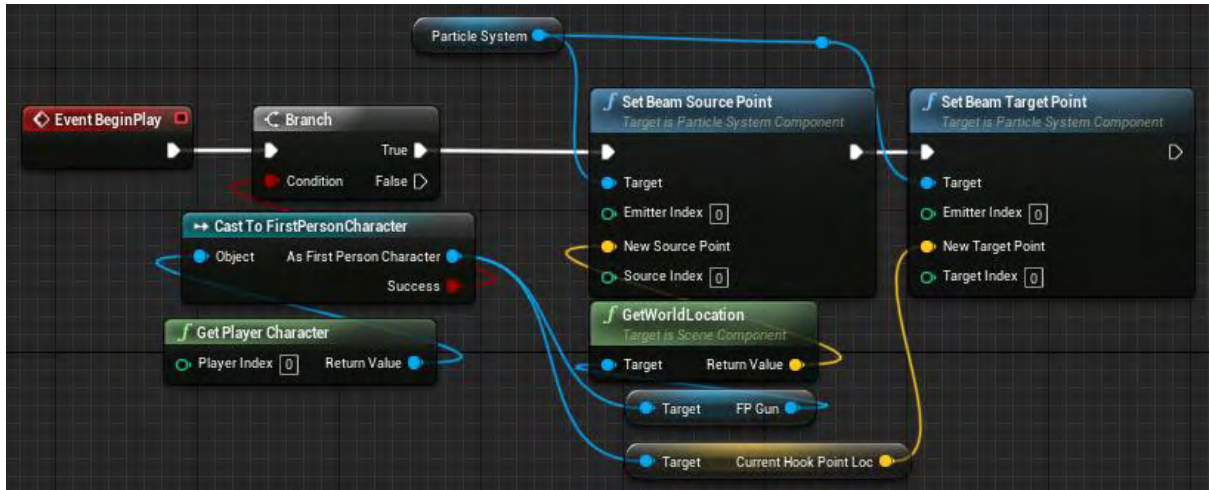


Το ParticleSystem έχει πρότυπο το Particle System P\_Beam. Το Mat\_Beam είναι το υλικό του (Εικόνα 67).



Εικόνα 67 BP\_HookBeam part 2

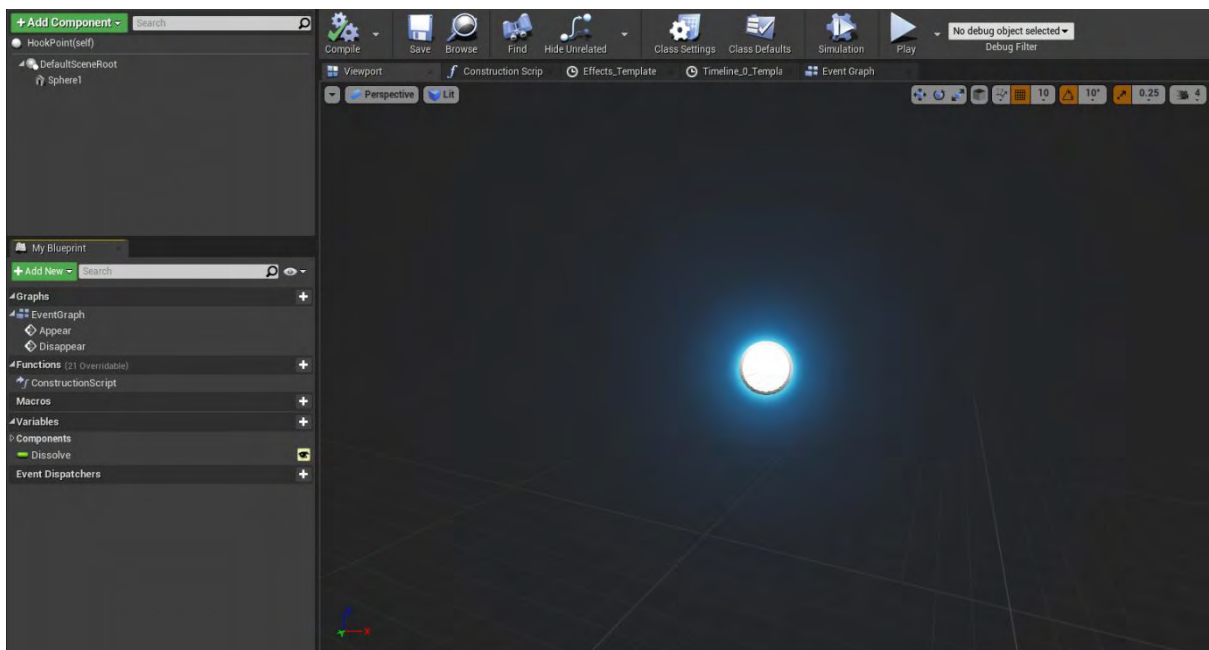
Κατά τη δημιουργία του BP\_HookBeam ελέγχεται η ύπαρξη του χαρακτήρα του παίκτη και στη συνέχεια αν υπάρχει, ορίζεται ως σημείο πηγής του ParticleSystem η τοποθεσία του FP\_Gun. Η τιμή της μεταβλητής CurrentHookPointLoc του FirstPersonCharacter ορίζεται ως σημείο στόχου (Εικόνα 68).



Εικόνα 68 BP\_HookBeam part 3

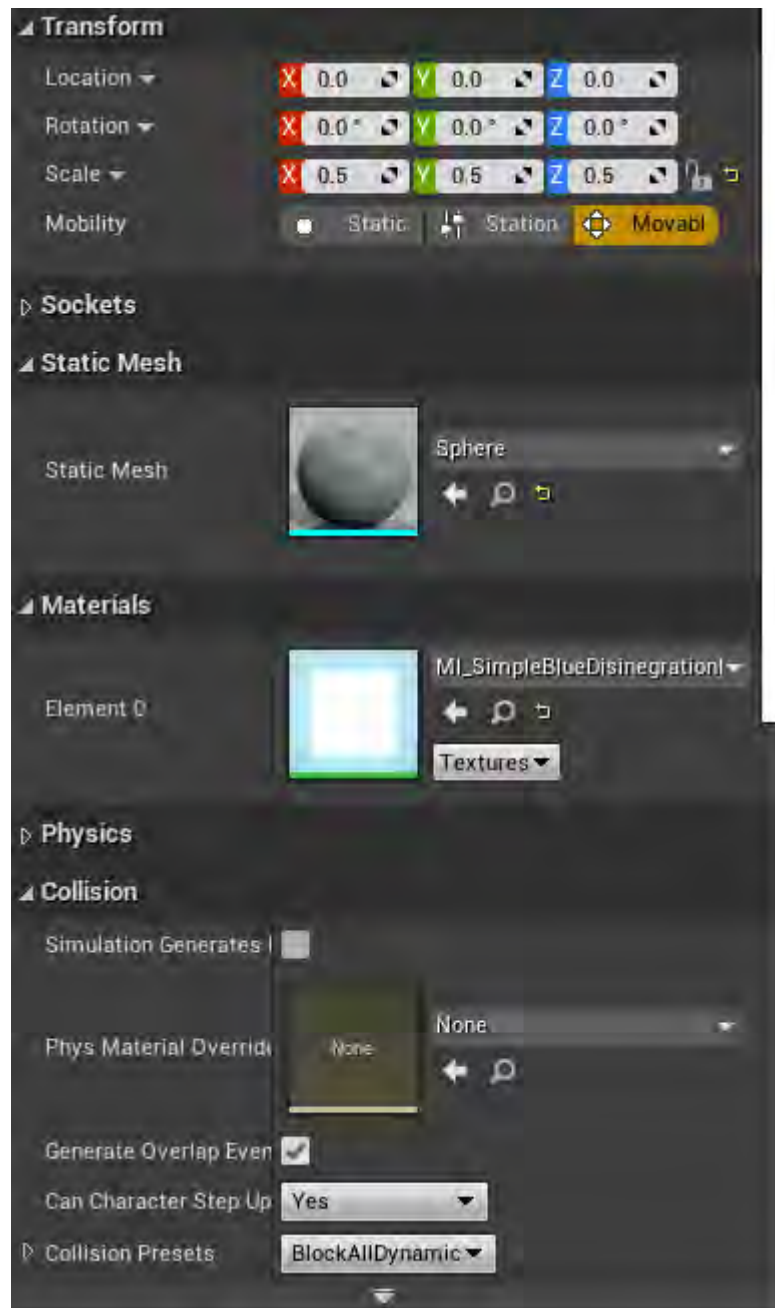
## Ενότητα 5.1.12 BP\_HookPoint

Το BP\_HookPoint αποτελείται από το Sphere1 τύπου Static Mesh Component και υλοποιεί τις συναρτήσεις Appear και Disappear (Εικόνα 69).



Εικόνα 69 BP\_HookPoint part 1

Το Sphere1 έχει το Static Mesh Sphere και υλικό το MI\_SimpleBlueDisintegrationE. Οι ρυθμίσεις σύγκρουσης του Sphere1 είναι από τις προκαθορισμένες (Εικόνα 70).



Εικόνα 70 BP\_HookPoint part 2

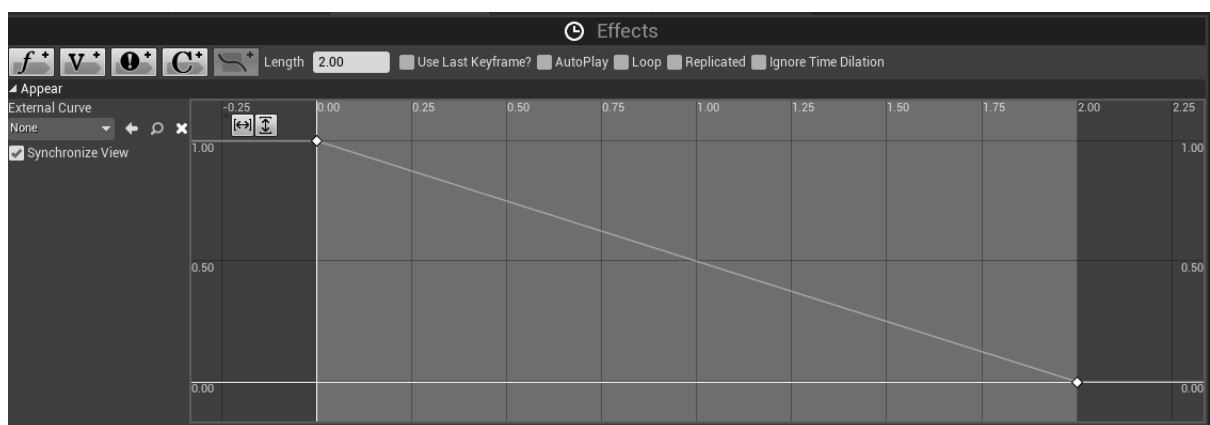
Η συνάρτηση Aappear εμφανίζει το BP\_HookPoint, ενεργοποιεί τη δυνατότητα σύγκρουσης και δημιουργεί μια δυναμική παρουσία του υλικού του Sphere1. Στη

συνέχεια με το χρονοδιάγραμμα Effects (Εικόνα 72) ενημερώνεται η τιμή DissolveAmount της δυναμικής παρουσίας του υλικού.

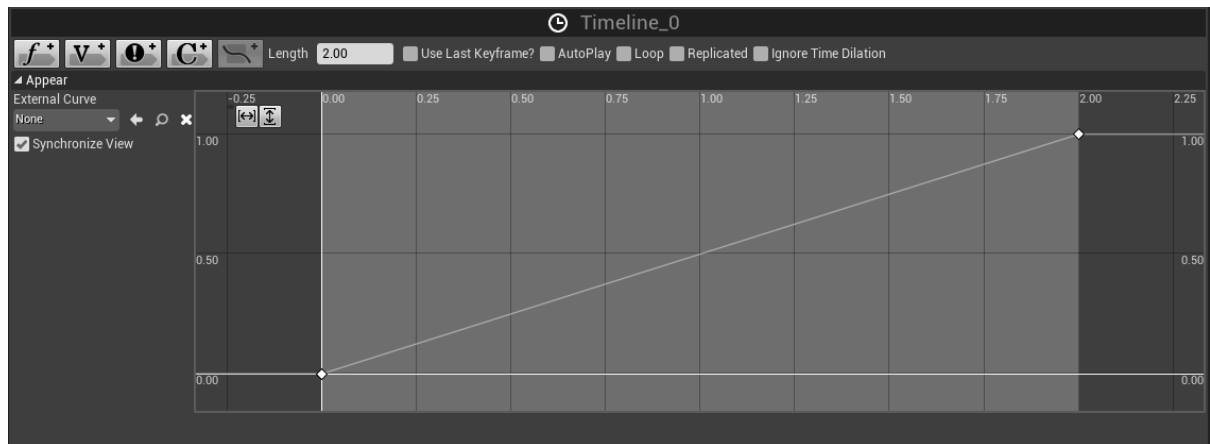
Η συνάρτηση Disappear δημιουργεί μια δυναμική παρουσία του υλικού sphere1. Ακολουθώντας με το χρονοδιάγραμμα Timeline\_0 (Εικόνα 73) ενημερώνεται η τιμή της μεταβλητής DissolveAmount της δυναμικής παρουσίας του υλικού. Όταν ολοκληρωθεί πλέον το Timeline\_0, απενεργοποιείται η δυνατότητα σύγκρουσης του BP\_HookPoint και γίνεται αόρατο στο παιχνίδι (Εικόνα 71).



Εικόνα 71 BP\_HookPoint part 3

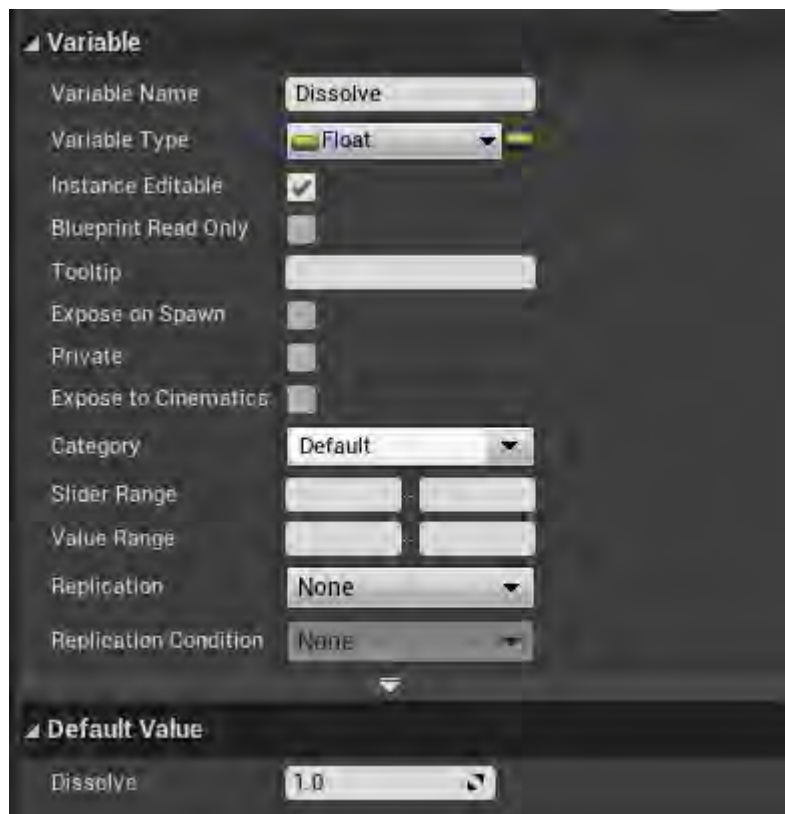


Εικόνα 72 BP\_HookPoint part 4



Εικόνα 73 BP\_HookPoint part 5

Η αρχική τιμή της μεταβλητής Dissolve είναι 1 (Εικόνα 74)



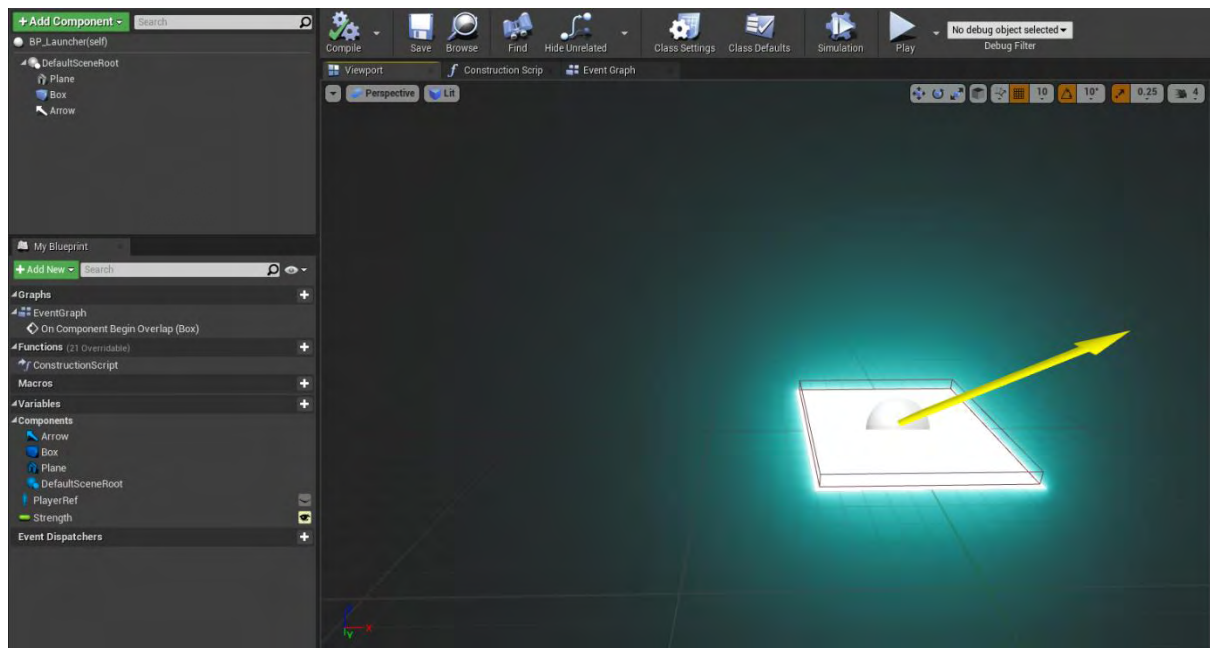
Εικόνα 74 BP\_HookPoint part 6

## Ενότητα 5.1.13 BP\_Launcher

---

Το BP\_Launcher απαρτίζεται από τρία μέρη (Εικόνα 75):

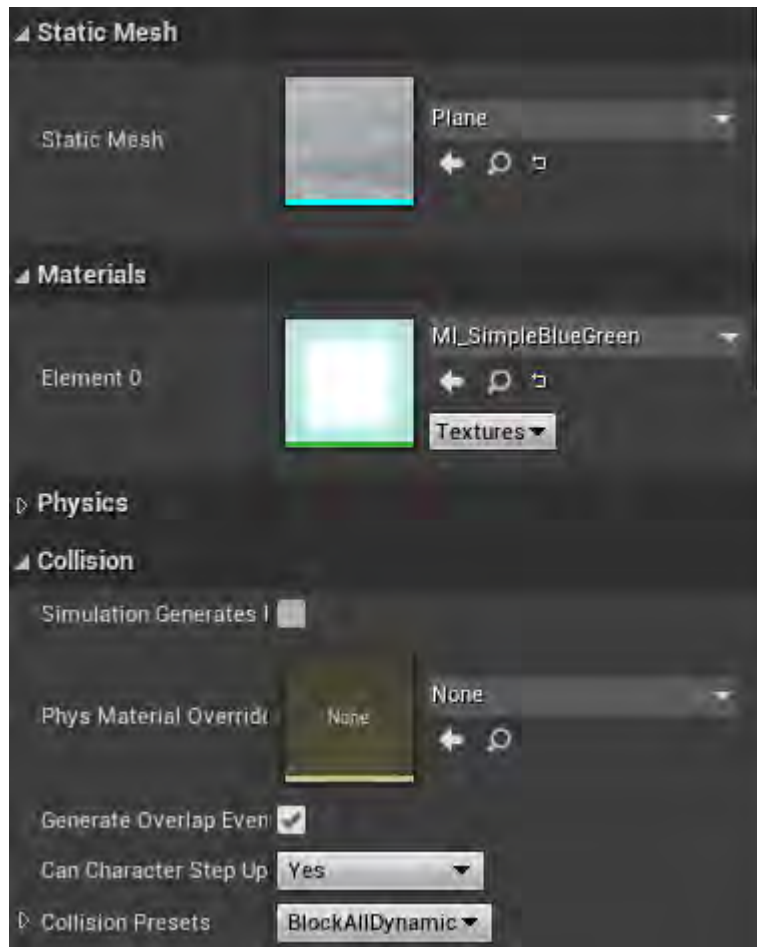
- Plane (Static Mesh Component)
- Box (Box Collision)
- Arrow (Arrow Component)



Εικόνα 75 BP\_Launcher part 1

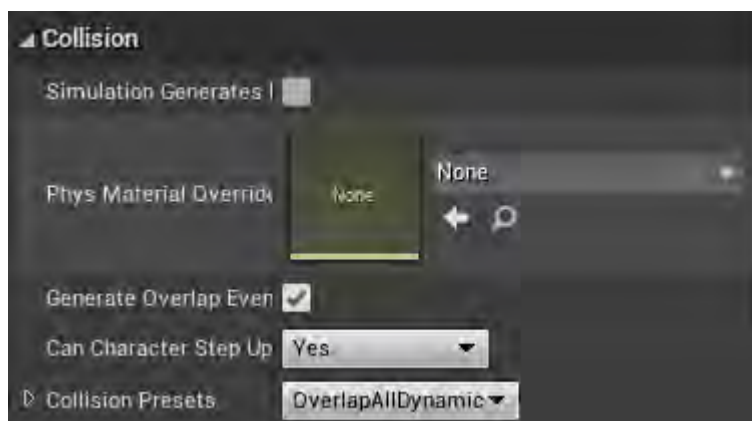
Το Plane χρησιμοποιεί το υλικό MI\_SimpleBlueGreen. Το Plane στις ρυθμίσεις σύγκρουσής του έχει το προκαθορισμένο BlockAllDynamic (Εικόνα 76).





Εικόνα 76 BP\_Launcher part 2

Οι ρυθμίσεις του Box είναι από τις προκαθορισμένες, η OverlapAllDynamic (Εικόνα 77).



Εικόνα 77 BP\_Launcher part 3

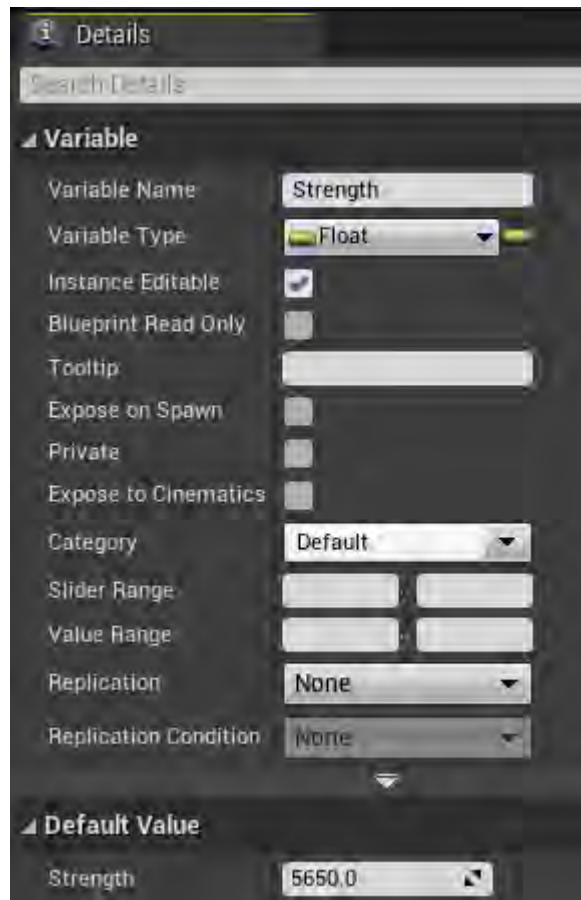
Στο Arrow η μόνη αλλαγή από τις προκαθορισμένες ρυθμίσεις της Unreal είναι το χρώμα του (Εικόνα 78).



Εικόνα 78 BP\_Launcher part 4

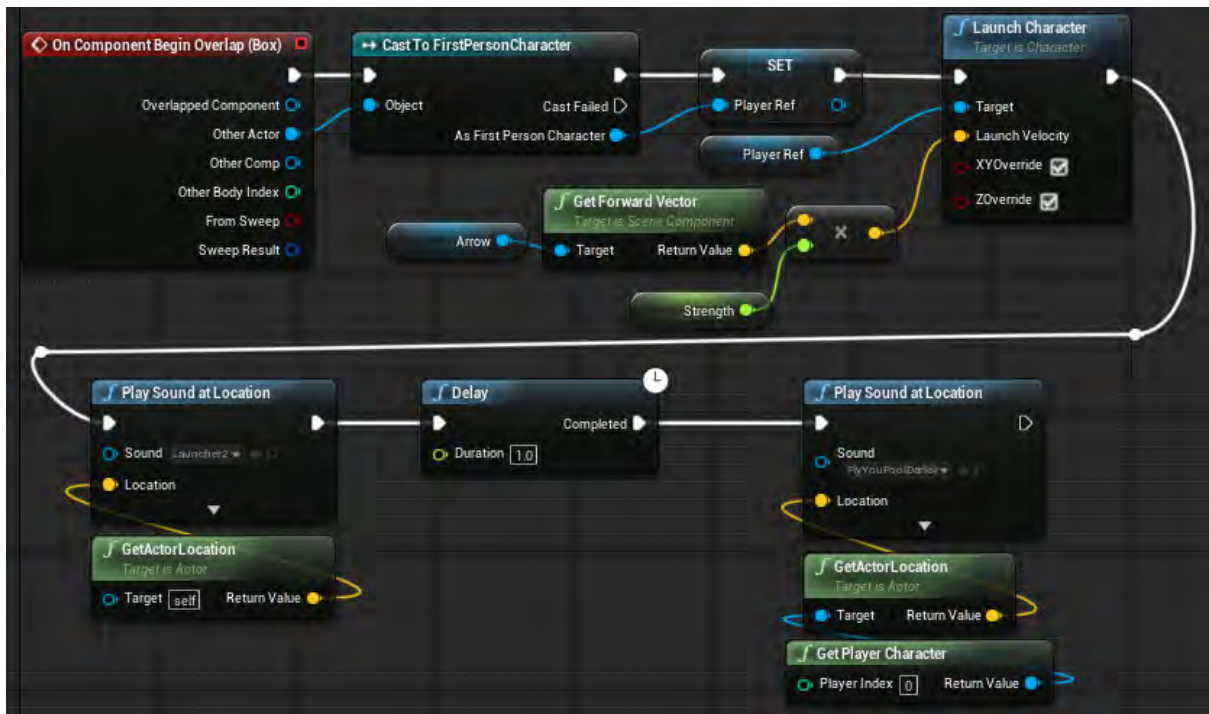


Η μεταβλητή Strength, τύπου Float, χρησιμοποιείται ως πολλαπλασιαστής. Η αρχική της τιμή είναι 5650 και μπορεί να τροποποιηθεί στον Editor (Εικόνα 79).



Εικόνα 79 BP\_Launcher part 5

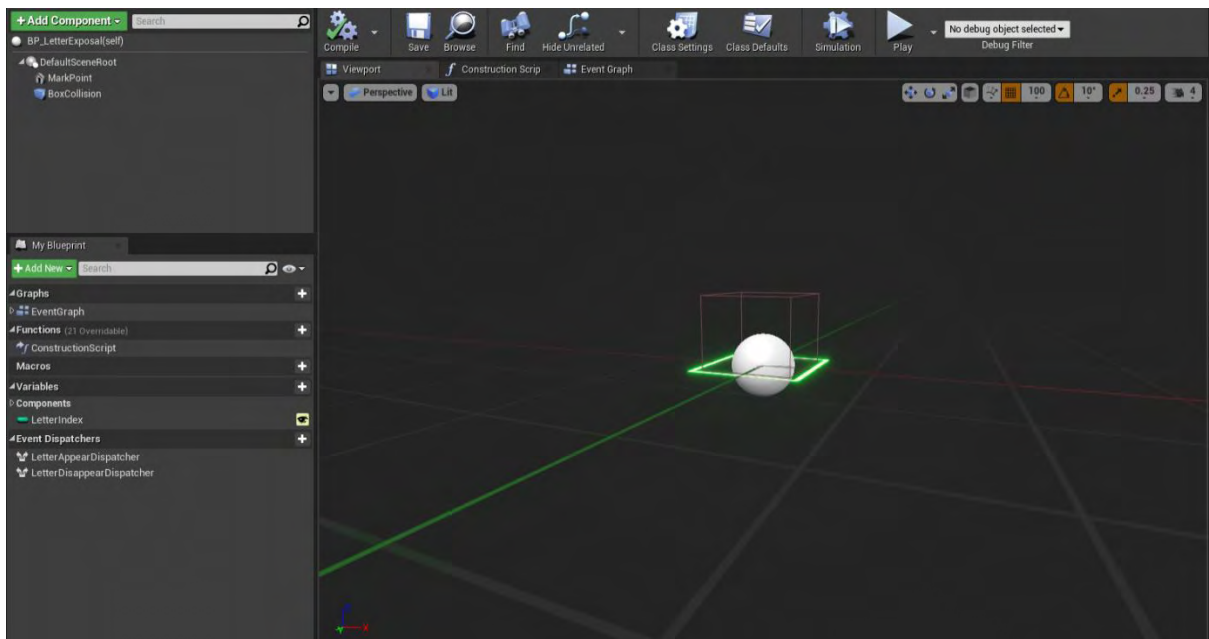
Το On Component Begin Overlap (Box) ενεργοποιείται όταν γίνει σύγκρουση με τον χαρακτήρα του παίκτη και τον εκτοξεύει προς την κατεύθυνση του Arrow με ταχύτητα εκτόξευσης ίση με την τιμή του Strength. Άμεσα αναπαράγεται ο ήχος Launcher2 και με χρονοκαθυστέρηση ενός δευτερολέπτου ακολουθεί ένας ακόμη ήχος (Εικόνα 80).



Εικόνα 80 BP\_Launcher part 6

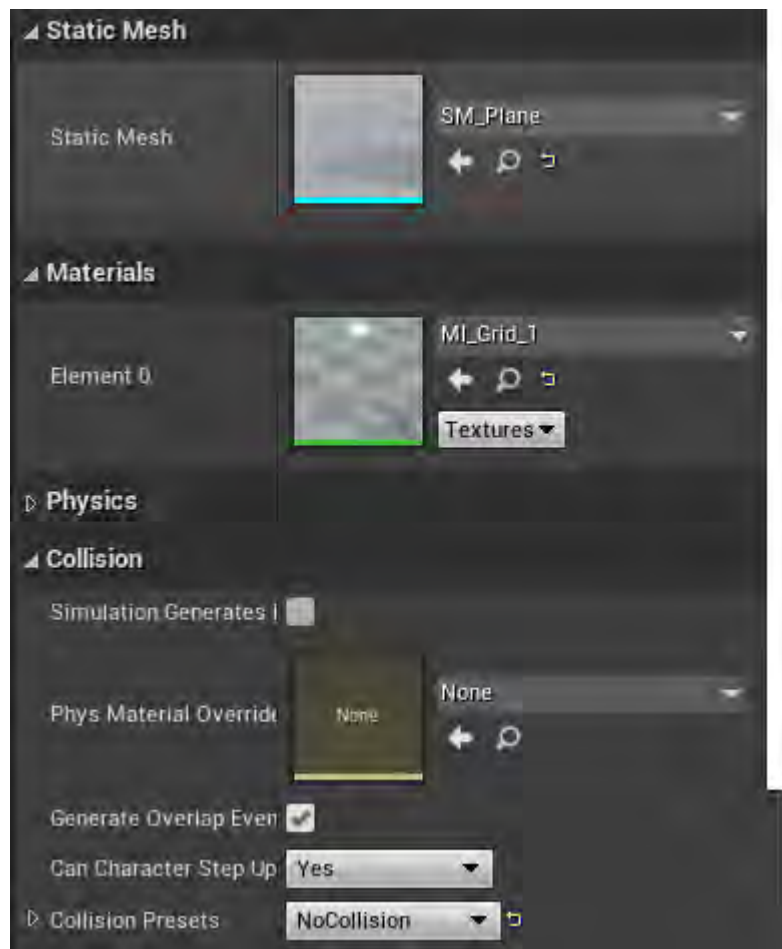
## Ενότητα 5.1.14 BP\_LetterExposal

Το BP\_LetterExposal απαρτίζεται από δύο μέρη: το MarkPoint (Static Mesh Component) και το BoxCollision (Box Collision), όπως φαίνεται στην Εικόνα 81.



Εικόνα 81 BP\_LetterExposal part 1

Το MarkPoint χρησιμοποιεί το Static Mesh SM\_Plane με το υλικό MI\_Grid\_1 (Εικόνα 82).



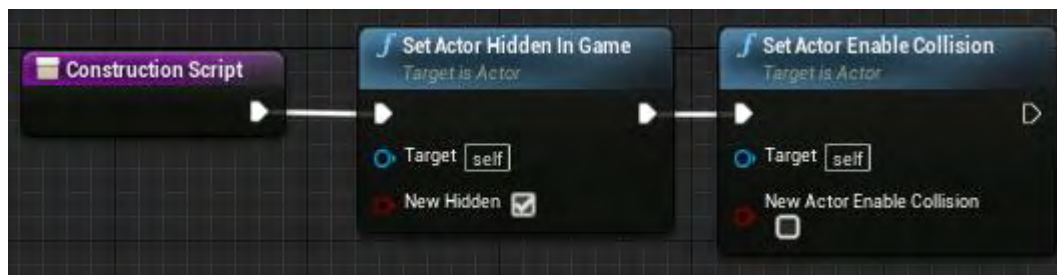
Εικόνα 82 BP\_LetterExposal part 2

Στην Εικόνα 83 φαίνονται οι ιδιότητες σύγκρουσης του BoxCollision.



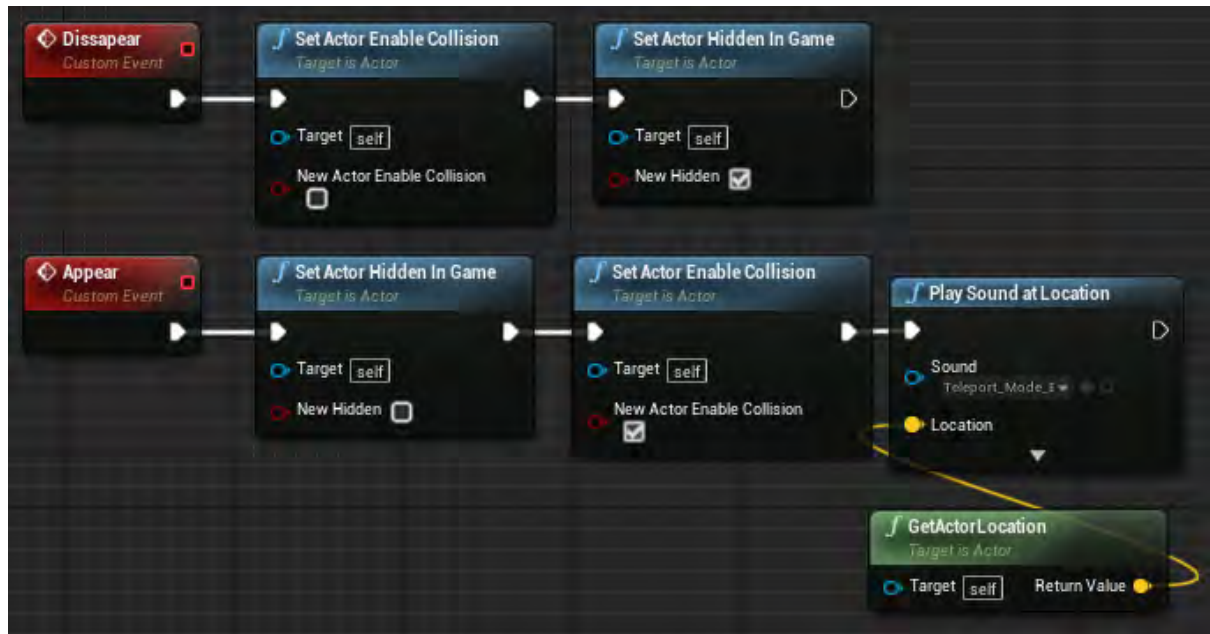
Εικόνα 83 BP\_LetterExposal part 3

Στο Construction Script γίνεται αόρατο το BP\_LetterExposal και απενεργοποιείται η δυνατότητα σύγκρουσης (Εικόνα 84).



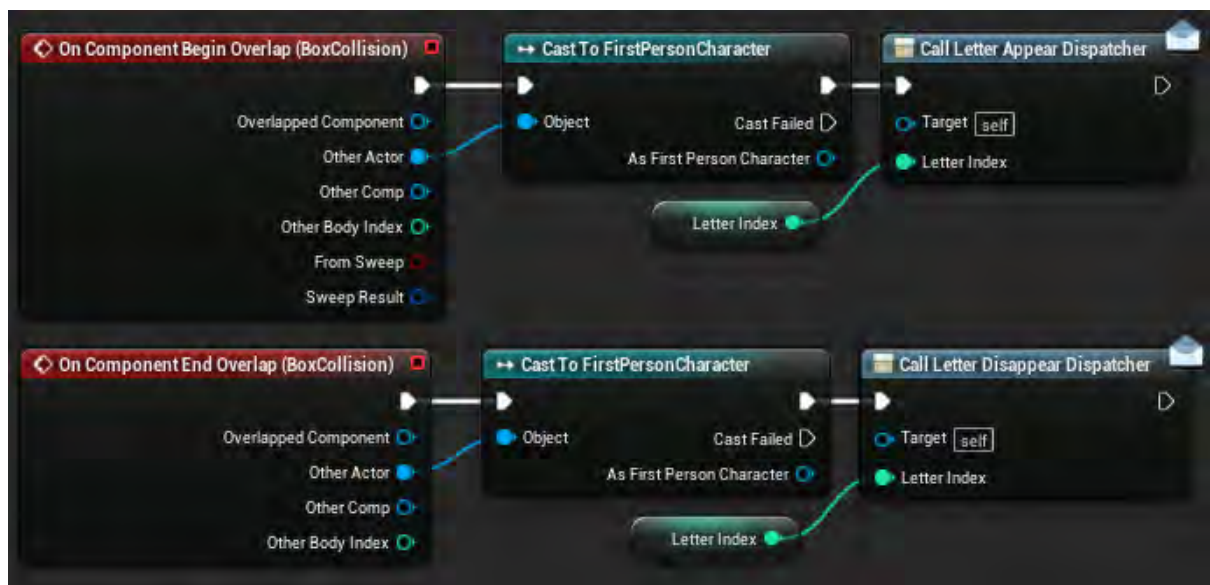
Εικόνα 84 BP\_LetterExposal part 4

Η συνάρτηση Appear εμφανίζει το BP\_LetterExposal, ενεργοποιεί τη δυνατότητα σύγκρουσης και αναπαράγει έναν ήχο. Η συνάρτηση Disappear κάνει αόρατο το BP\_LetterExposal και απενεργοποιεί τη δυνατότητα σύγκρουσης (Εικόνα 85).



Εικόνα 85 BP\_LetterExposal part 5

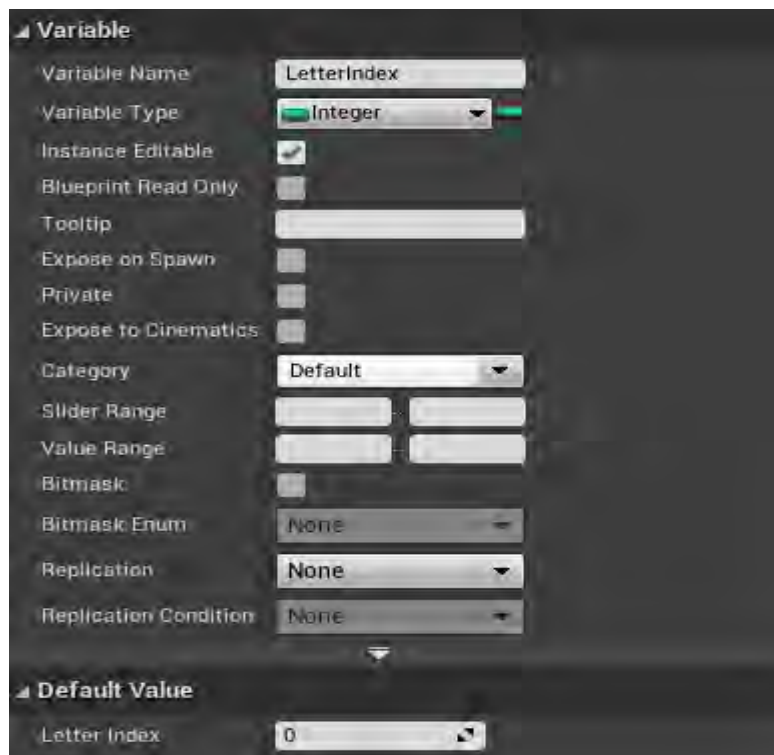
Με την έναρξη σύγκρουσης (On Component Begin Overlap) με τον χαρακτήρα του παίκτη καλείται το Letter Appear Dispatcher. Με τη λήξη της σύγκρουσης (On Component End Overlap) καλείται το Letter Disappear Dispatcher (Εικόνα 86).



Εικόνα 86 BP\_LetterExposal part 6

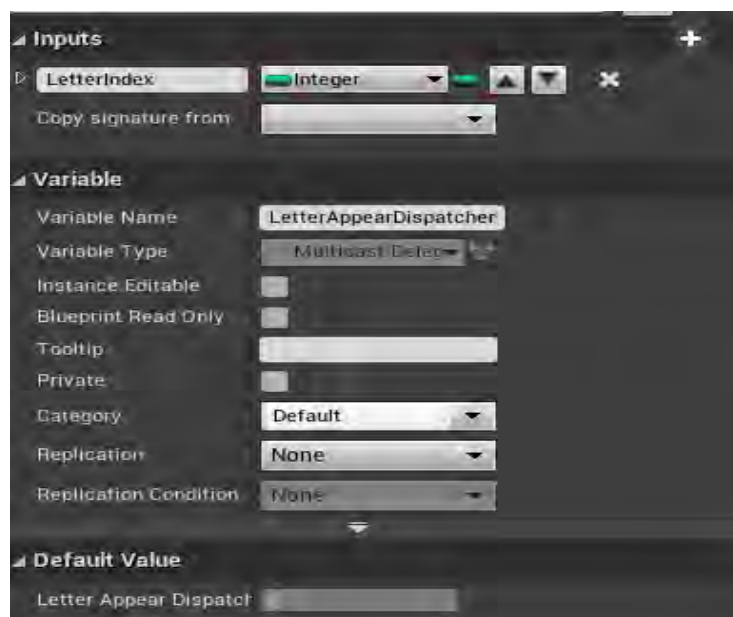


Αξίζει να επισημανθεί ότι η μεταβλητή LetterIndex είναι Instance Editable και η τιμή της μπορεί να διαφέρει από actor σε actor εντός του επιπέδου (Εικόνα 87).

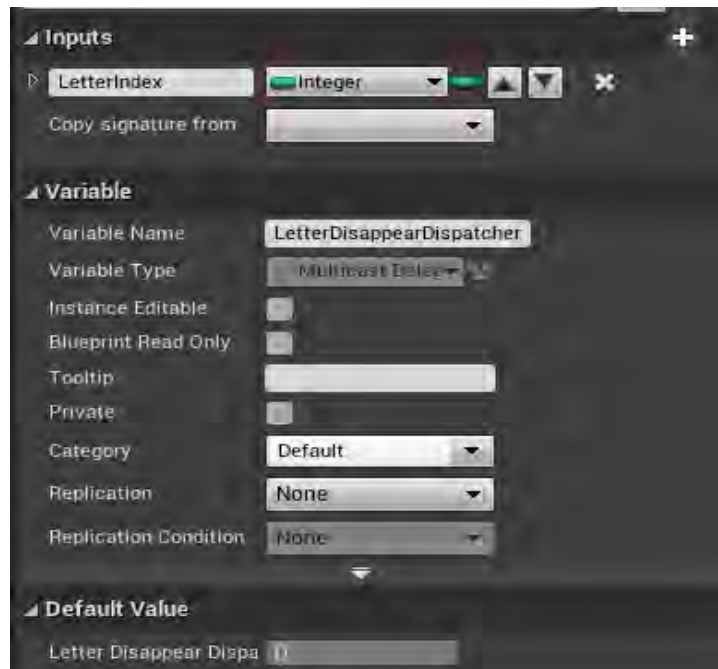


Εικόνα 87 BP\_LetterExposal part 7

Στην Εικόνα 88 και την Εικόνα 89 απεικονίζονται τα Event Dispatcher.



Εικόνα 88 BP\_LetterExposal part 8

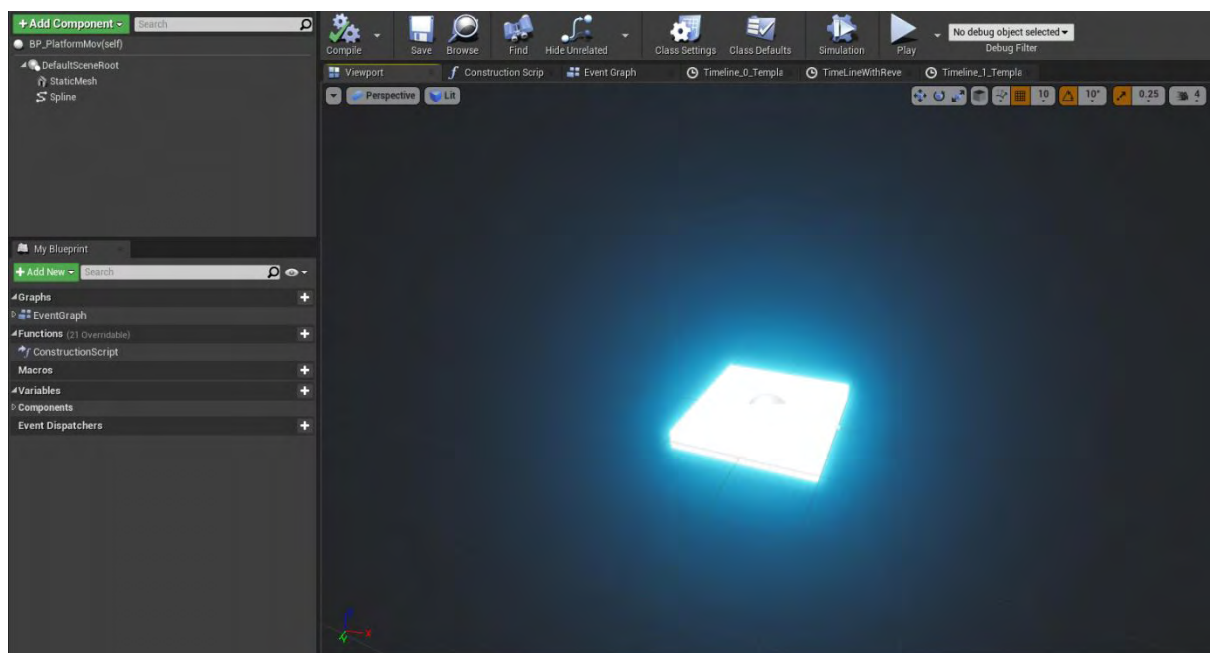


Εικόνα 89 BP\_LetterExposal part 9

## Ενότητα 5.1.15 BP\_ PlatformMov

Το BP\_PlatformMov αποτελείται από τα (Εικόνα 90) :

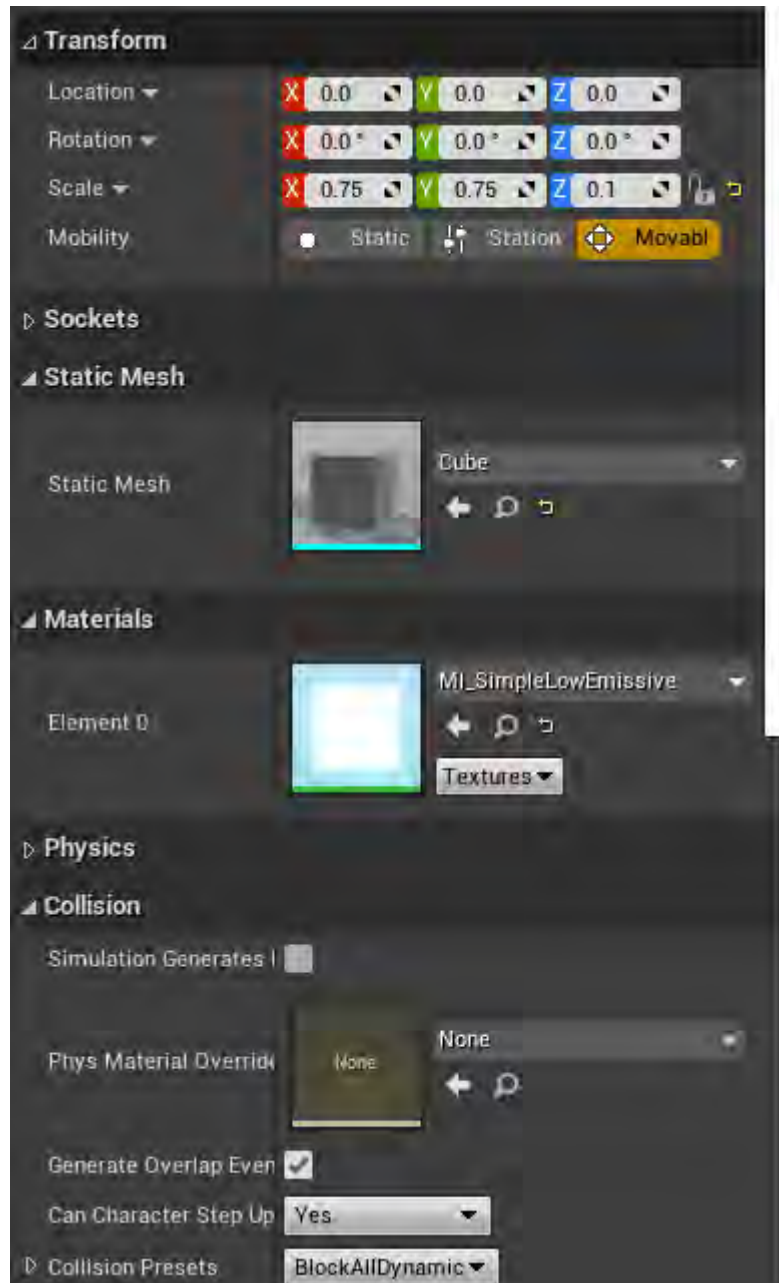
- StaticMesh (Static Mesh Component)
- Spline (Spline Component)



Εικόνα 90 BP\_PlatformMov part 1

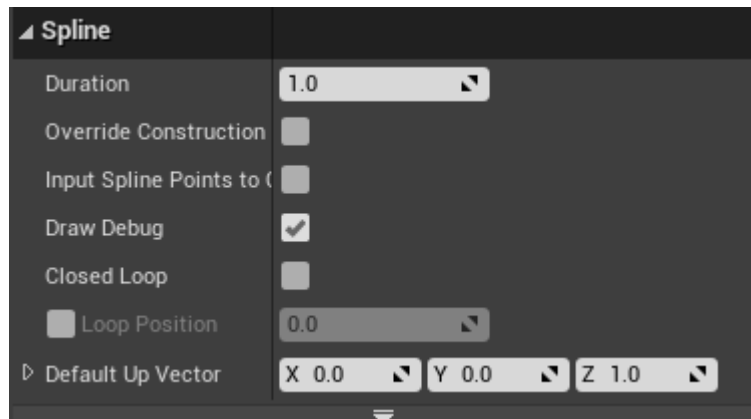


Στην Εικόνα 91 φαίνεται η επιλογή του Cube ως το Static Mesh του StaticMesh. Το υλικό που χρησιμοποιεί (MI\_SimpleLowEmissive) και οι ιδιότητες σύγκρουσης του StaticMesh (Εικόνα 91).



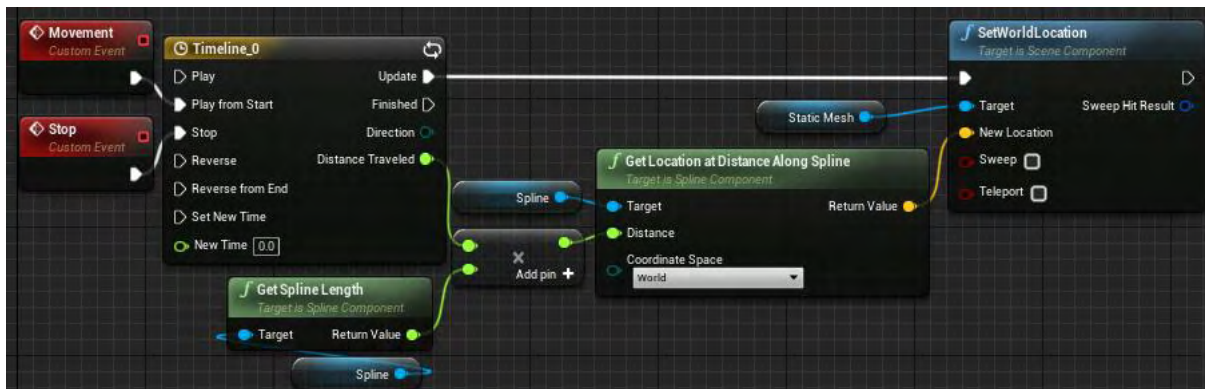
Εικόνα 91 BP\_PlatformMov part 2

Στην Εικόνα 92 φαίνονται οι ρυθμίσεις του Spline, οι οποίες είναι οι αρχικές. (Εικόνα 92).

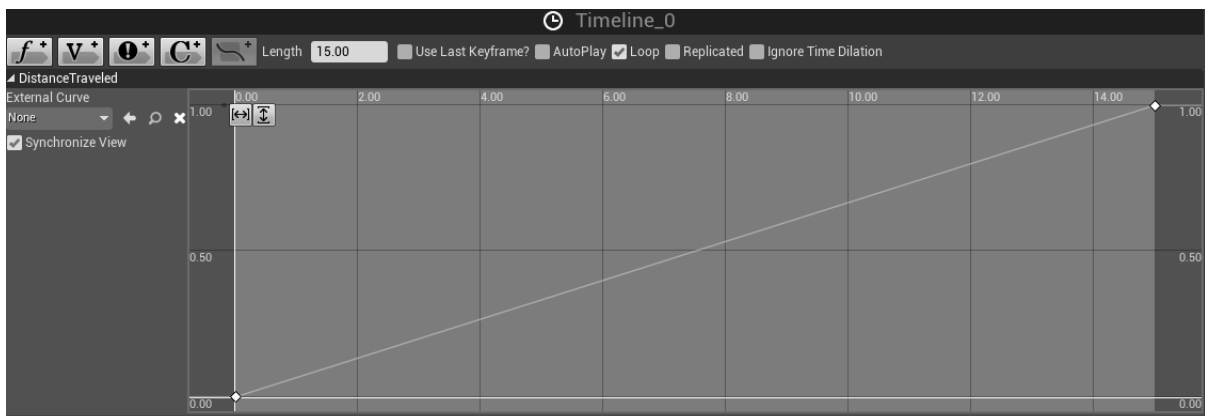


Εικόνα 92 BP\_PlatformMov part 3

Όπως φαίνεται στην Εικόνα 93, η συνάρτηση Movement ξεκινά από την αρχή το χρονοδιάγραμμα Timeline\_0 (Εικόνα 94) σύμφωνα με το οποίο ενημερώνει την τοποθεσία του StaticMesh πάνω στο Spline. Η συνάρτηση Stop σταματά το Timeline\_0.

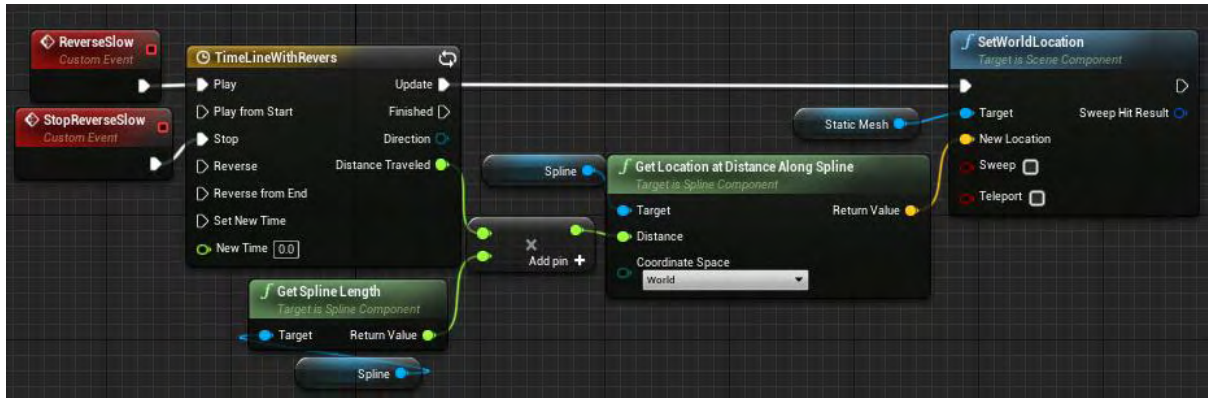


Εικόνα 93 BP\_PlatformMov part 4

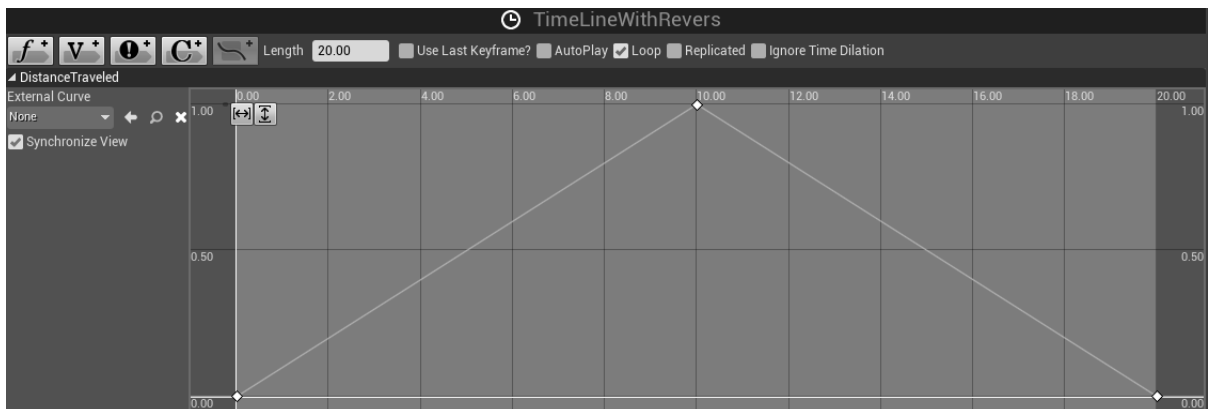


Εικόνα 94 BP\_PlatformMov part 5

Όπως φαίνεται στην Εικόνα 95, η συνάρτηση ReverseSlow ξεκινά το χρονοδιάγραμμα TimeLineWithRevers (Εικόνα 96) σύμφωνα με το οποίο ενημερώνει την τοποθεσία του StaticMesh πάνω στο Spline. Η συνάρτηση StopReverseSlow σταματά το TimeLineWithRevers .

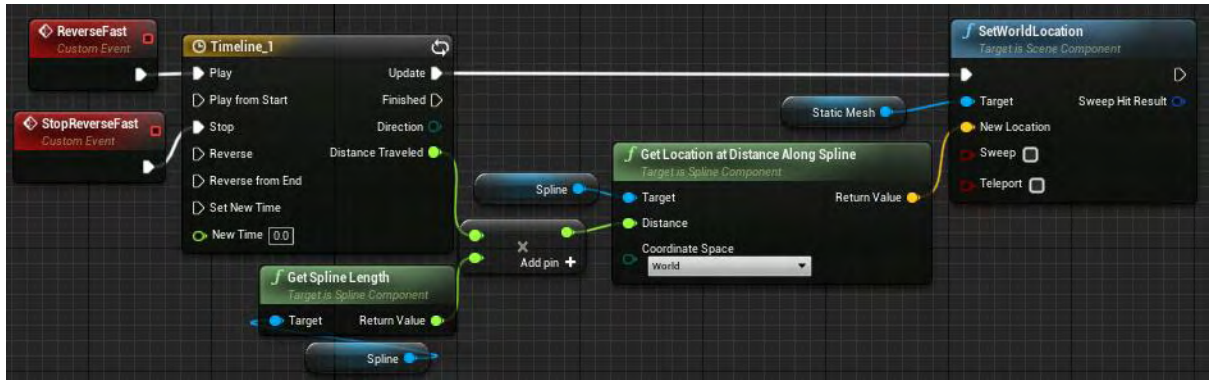


Εικόνα 95 BP\_PlatformMov part 6

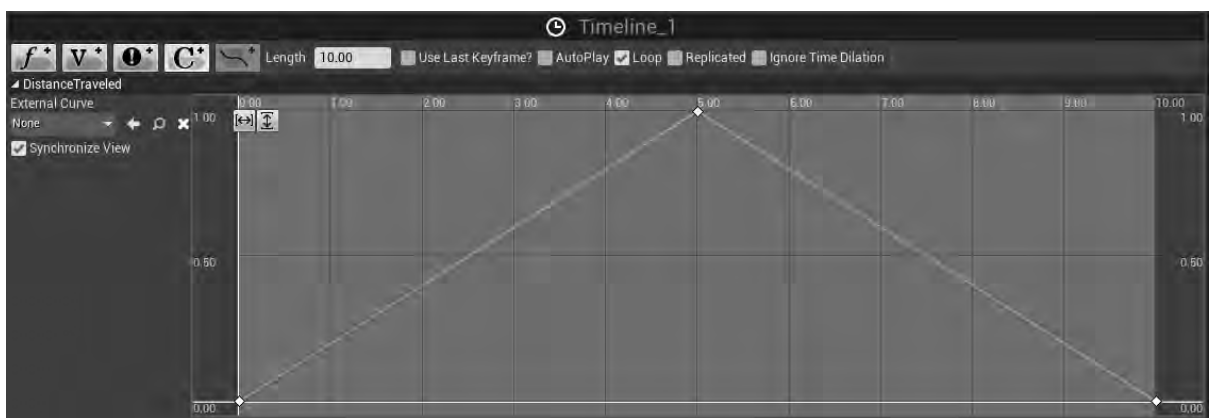


Εικόνα 96 BP\_PlatformMov part 7

Σύμφωνα με την Εικόνα 97, η συνάρτηση ReverseFast ξεκινά το χρονοδιάγραμμα Timeline\_1 (Εικόνα 98) σύμφωνα με το οποίο ενημερώνει την τοποθεσία του StaticMesh πάνω στο Spline. Η συνάρτηση Stop ReverseFast σταματά το Timeline\_1.



Εικόνα 97 BP\_PlatformMov part 8

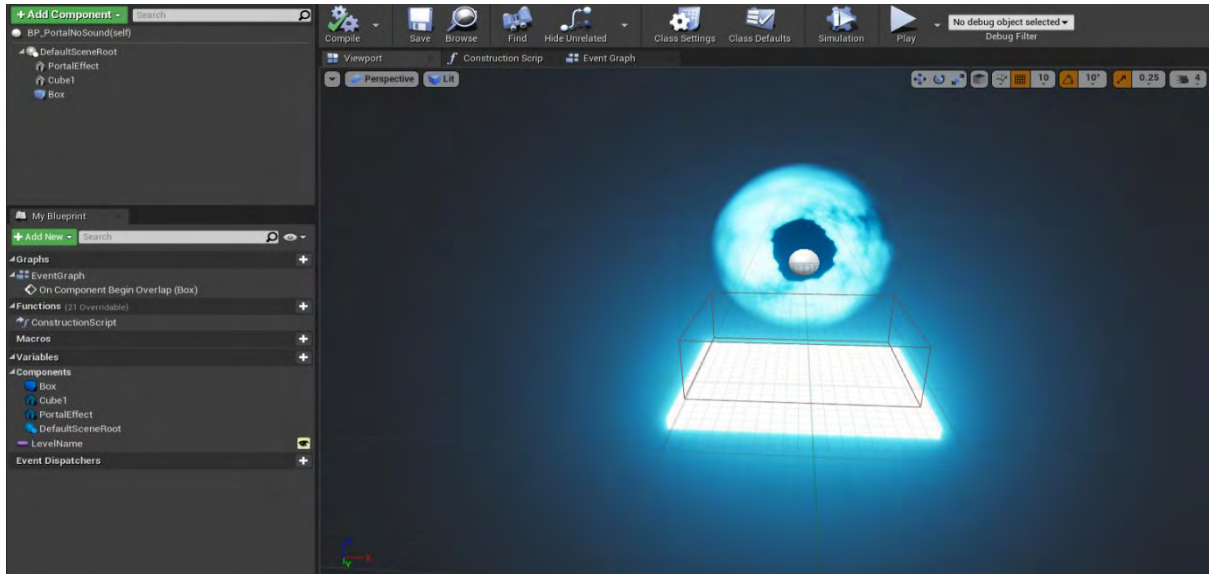


Εικόνα 98 BP\_PlatformMov part 9

## Ενότητα 5.1.16 BP\_PortalNoSound

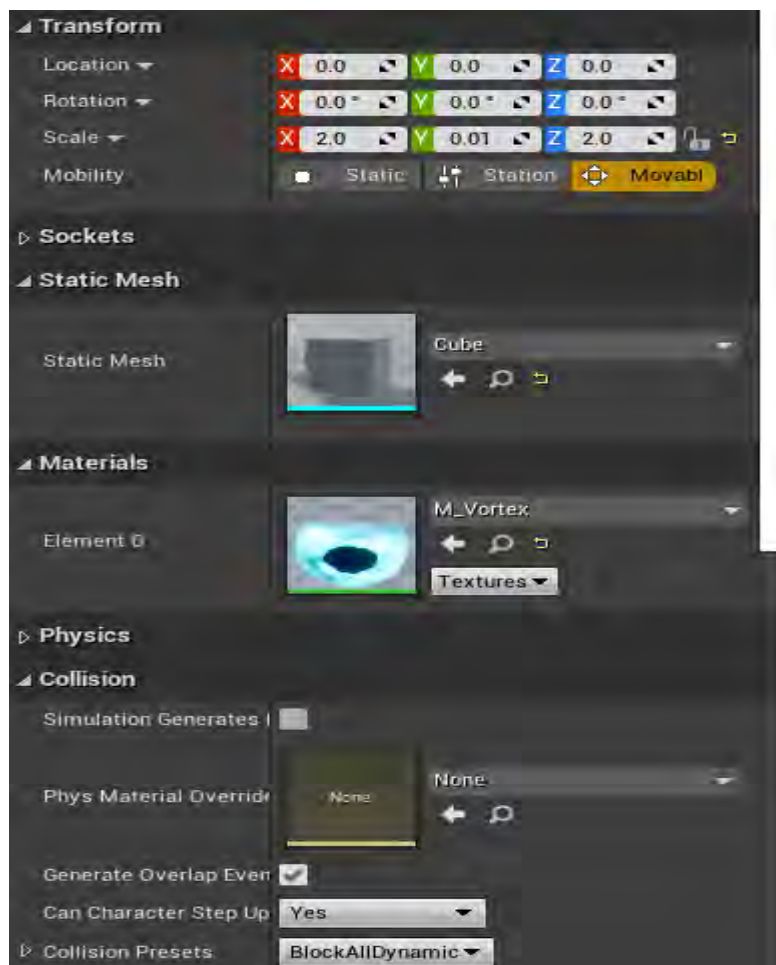
Το BP\_PortalNoSound αποτελείται από τρία μέρη Εικόνα 99:

- PortalEffect (Static Mesh Component)
- Cube1 (Static Mesh Component)
- Box (Box Collision)



Εικόνα 99 BP\_PortalNoSound part 1

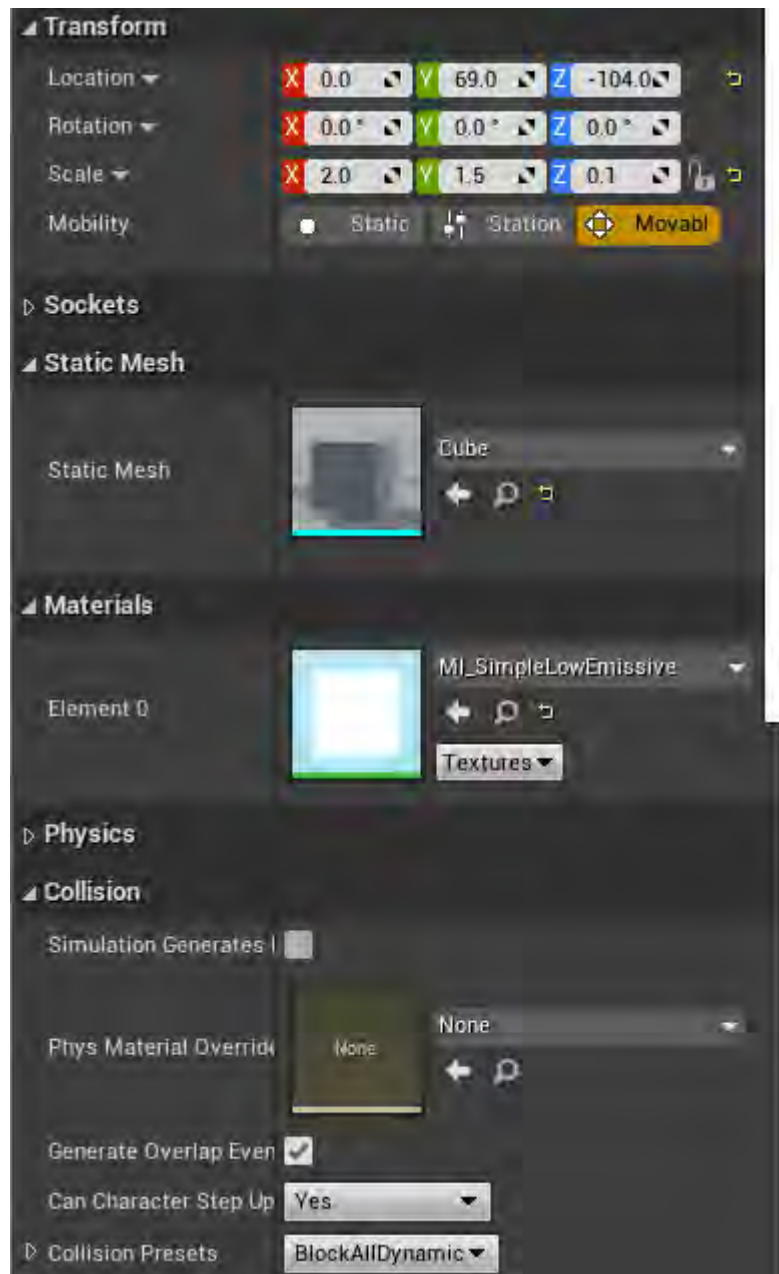
Στην Εικόνα 100 φαίνεται η επιλογή του Cube ως το Static Mesh του PortalEffect, το υλικό που χρησιμοποιεί (M\_Vortex) και οι ιδιότητες σύγκρουσης του.



Εικόνα 100 BP\_PortalNoSound part 2

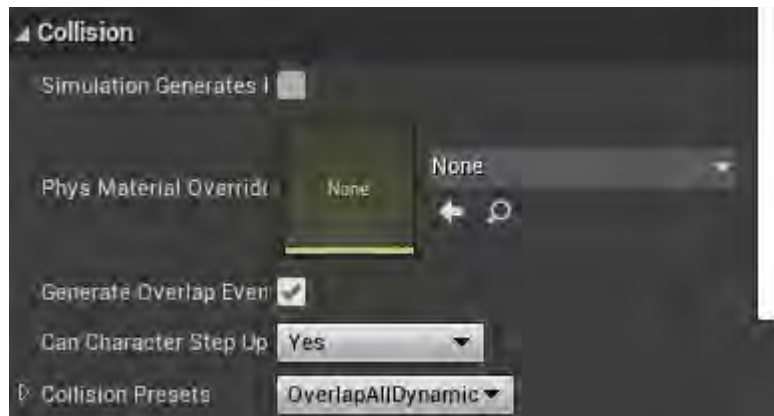


Στην Εικόνα 101 φαίνεται η επιλογή του Cube ως το Static Mesh του Cube1, το υλικό που χρησιμοποιεί (MI\_SimpleLowEmissive) και οι ιδιότητες σύγκρουσης του.



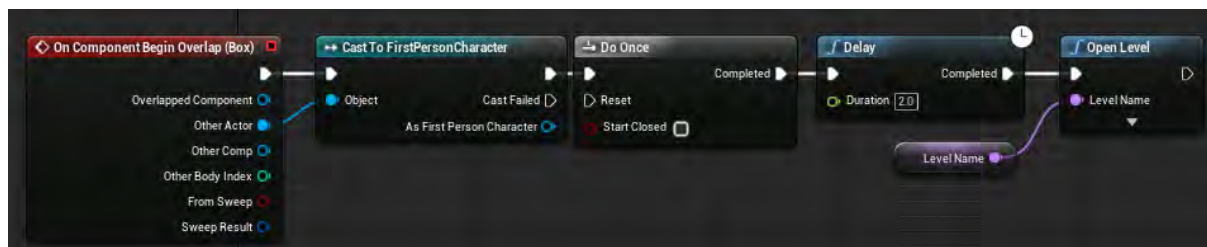
Εικόνα 101 BP\_PortalNoSound part 3

Στην Εικόνα 102 φαίνονται οι ιδιότητες σύγκρουσης του Box.



Εικόνα 102 BP\_PortalNoSound part 4

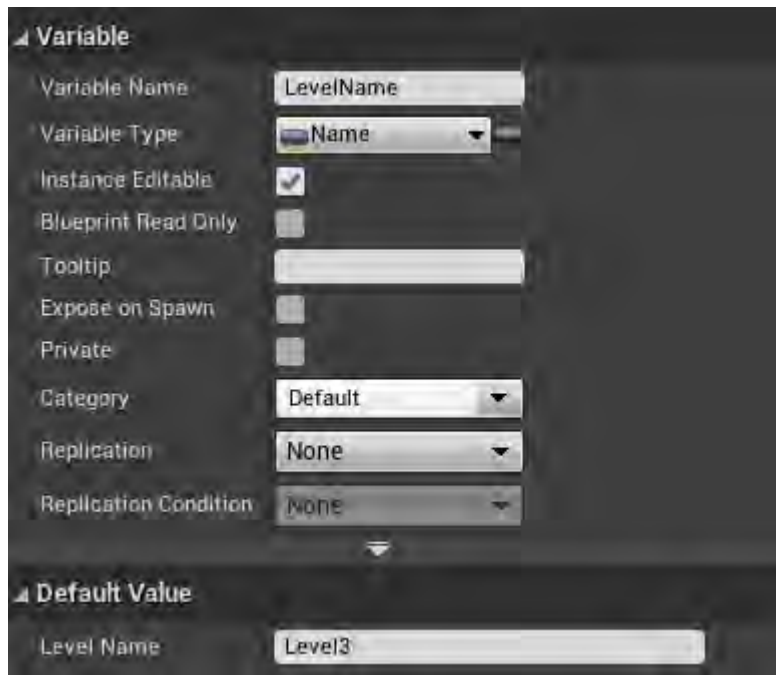
Όταν ξεκινήσει σύγκρουση με το Box ενεργοποιείτε το On Component Begin Overlap και πραγματοποιείται έλεγχος αν η σύγκρουση ήταν με τον χαρακτήρα του παίχτη. Στη συνέχεια μετά από χρονοκαθυστέρηση ενός δευτερολέπτου ανοίγει το επίπεδο που αντιπροσωπεύει η τιμή της μεταβλητής LevelName (Εικόνα 103).



Εικόνα 103 BP\_PortalNoSound part 5

Η μεταβλητή LevelName είναι Instance Editable που σημαίνει ότι έχει την δυνατότητα διαφέρει από actor σε actor (Εικόνα 104).

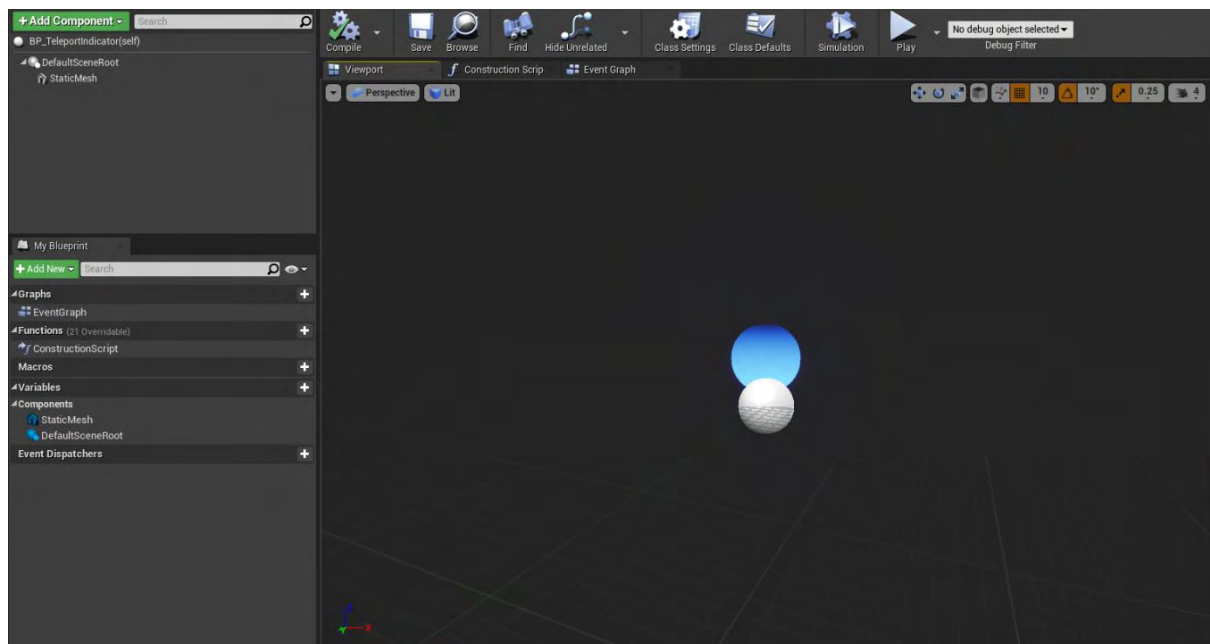




Εικόνα 104 BP\_PortalNoSound part 6

## Ενότητα 5.1.17 BP\_TeleportIndicator

Το BP\_TeleportIndicator έχει μόνο ένα Static Mesh Component το StaticMesh (Εικόνα 105).



Εικόνα 105 BP\_TeleportIndicator part 1

Στην εικόνα 106 φαίνονται οι ιδιότητες του StaticMesh και δεν θα αναλυθούν περαιτέρω.



Εικόνα 106 BP\_TeleportIndicator part 2

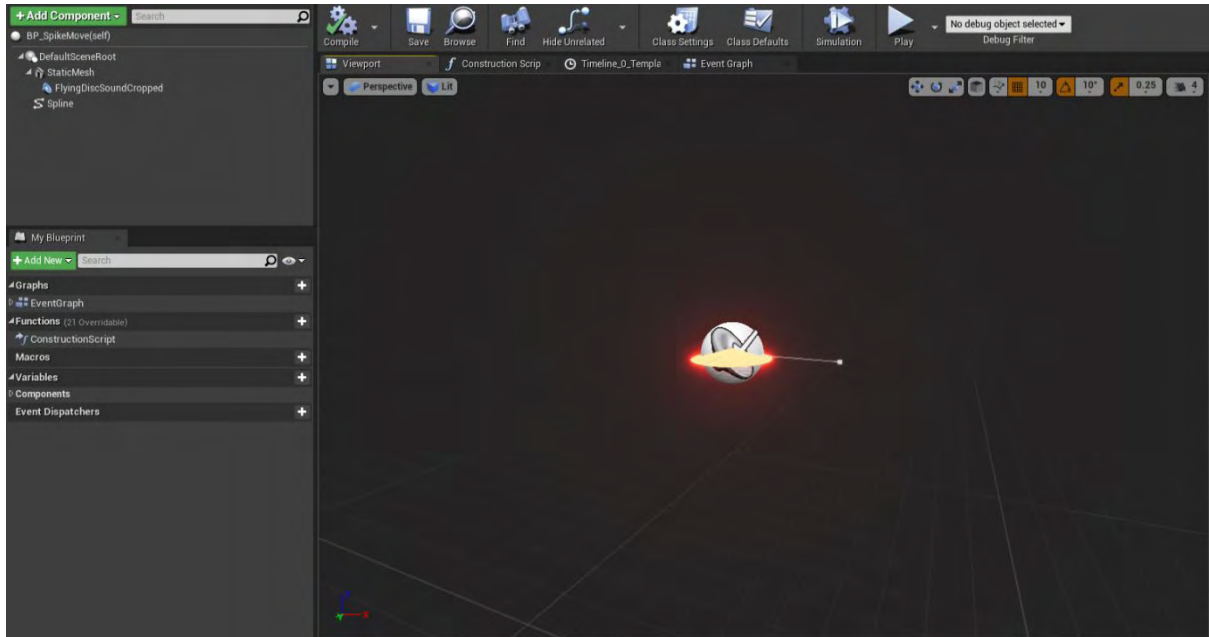
### Ενότητα 5.1.18 BP\_SpikeMove

---

Το BP\_SpikeMove απαρτίζεται από τρία (3) μέρη (Εικόνα 107):

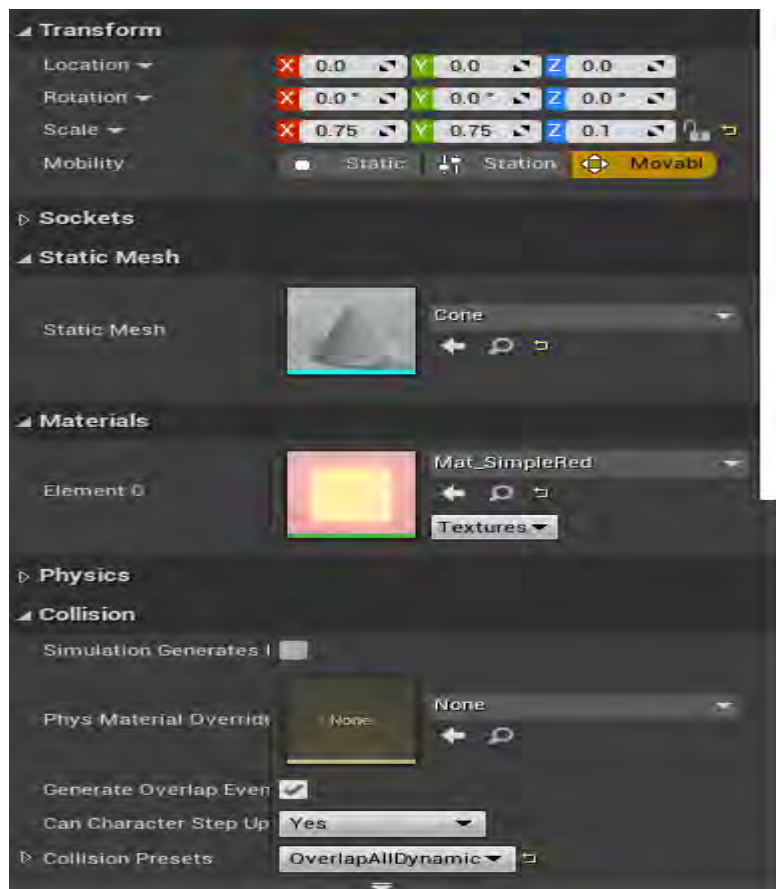
- StaticMesh (Static Mesh Component)
  - ❖ FlyingDiscSoundCropped (Audio Component)
- Spline (Spline Component)

Το Audio Component είναι συνημμένο στο Static Mesh Component.



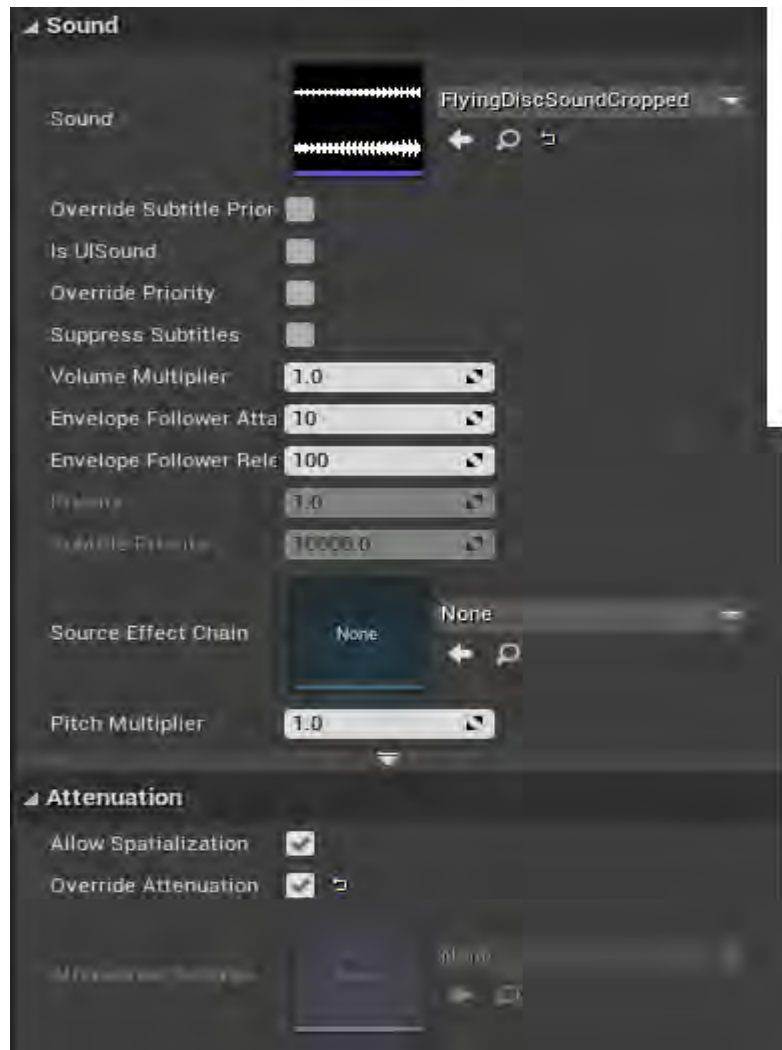
Εικόνα 107 BP\_SpikeMove part 1

Στην εικόνα 108 φαίνονται οι ιδιότητες του Static Mesh Component και δεν θα αναλυθούν περαιτέρω.



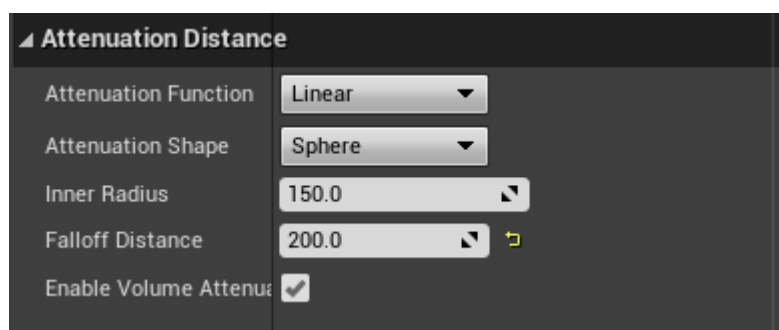
Εικόνα 108 BP\_SpikeMove part 2

Στην εικόνα 109 φαίνονται οι ρυθμίσεις του Audio Component.



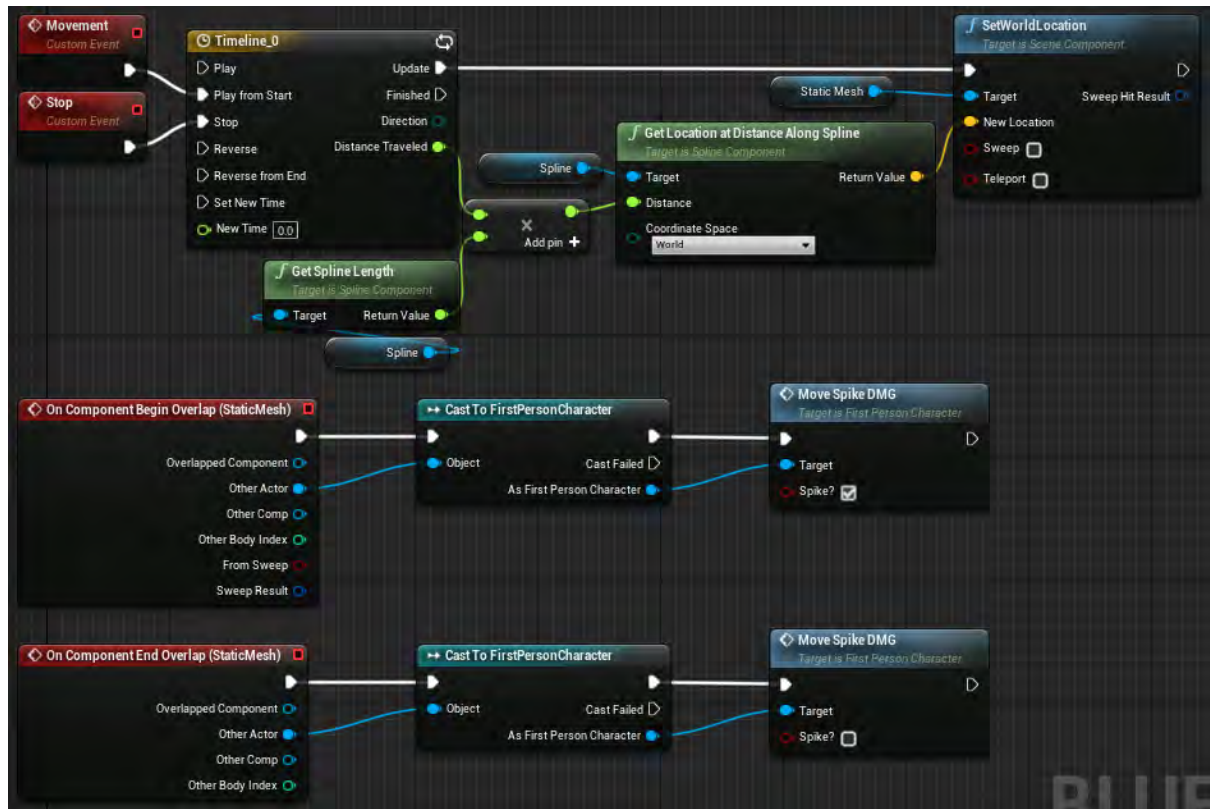
Εικόνα 109 BP\_SpikeMove part 3

Στην εικόνα 110 φαίνονται οι ρυθμίσεις του Audio Component και οι τροποποιήσεις του στην κατηγορία Attenuation Distance.



Εικόνα 110 BP\_SpikeMove part 4

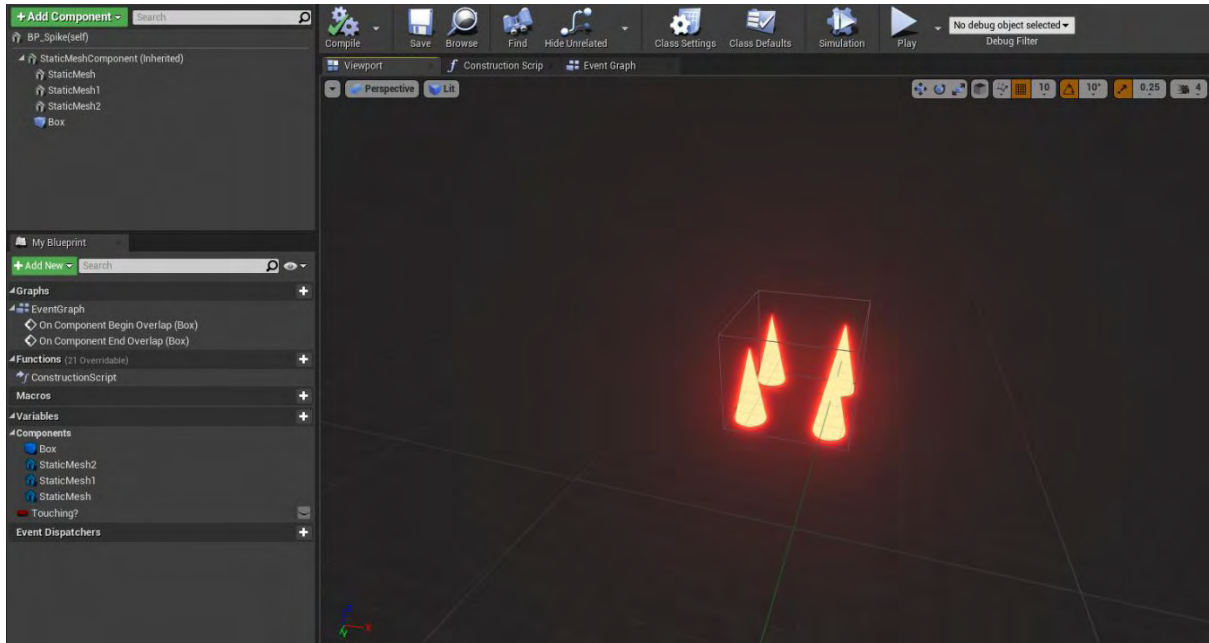
Η συνάρτηση Movement ενεργοποιεί το χρονοδιάγραμμα Timeline\_0 από την αρχή. Αυτό ενημερώνει συνεχώς την τοποθεσία του StaticMesh πάνω στο μονοπάτι του Spline. Με την έναρξη της σύγκρουσης του StaticMesh με τον χαρακτήρα του παίκτη καλείται η συνάρτηση MoveSpikeDMG με την τιμή της μεταβλητής Spike? να είναι αληθής. Με την λήξη της σύγκρουσης καλείται η συνάρτηση MoveSpikeDMG με την τιμή της μεταβλητής Spike? να είναι ψευδής (Εικόνα 111).



Εικόνα 111 BP\_SpikeMove part 5

## Ενότητα 5.1.19 BP\_Spike

Το BP\_Spike αποτελείται από τέσσερα (4) Static Mesh Component και ένα (1) Box Collision (Box), όπως φαίνεται στην Εικόνα 112.



Εικόνα 112 BP\_Spike part 1

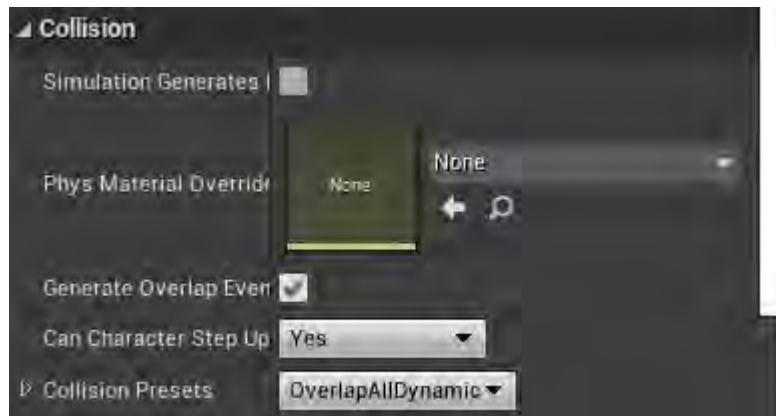
Στην εικόνα 113 φαίνονται οι ιδιότητες όλων των Static Mesh Components και δεν θα αναλυθούν περαιτέρω.



Εικόνα 113 BP\_Spike part 2

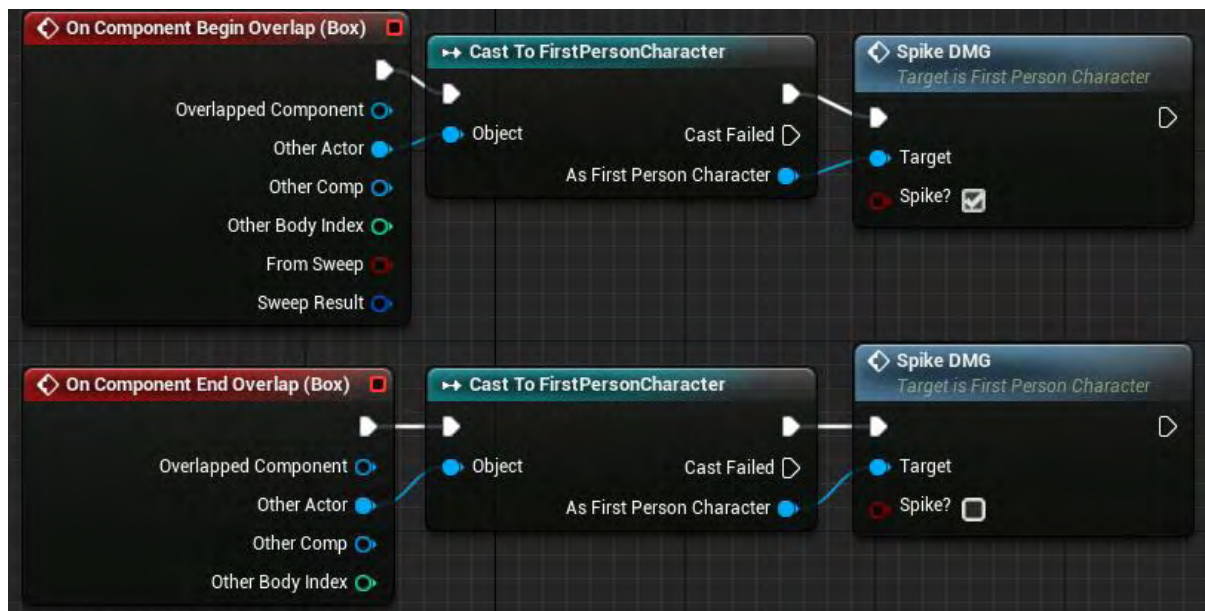


Στην Εικόνα 114 φαίνονται οι ιδιότητες σύγκρουσης του Box.



Εικόνα 114 BP\_Spike part 3

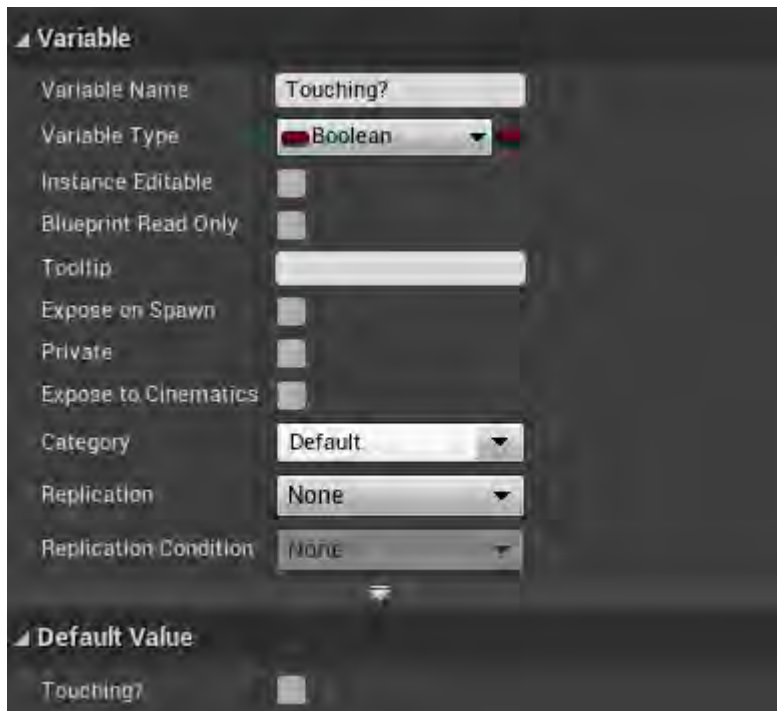
Η έναρξη της σύγκρουσης του Box με τον χαρακτήρα του παίκτη σηματοδοτεί την κλήση της συνάρτησης MoveSpikeDMG με την τιμή της μεταβλητής Spike? να είναι αληθής. Με την λήξη της σύγκρουσης καλείται η συνάρτηση MoveSpikeDMG με την τιμή της μεταβλητής Spike? να είναι ψευδής (Εικόνα 115).



Εικόνα 115 BP\_Spike part 4

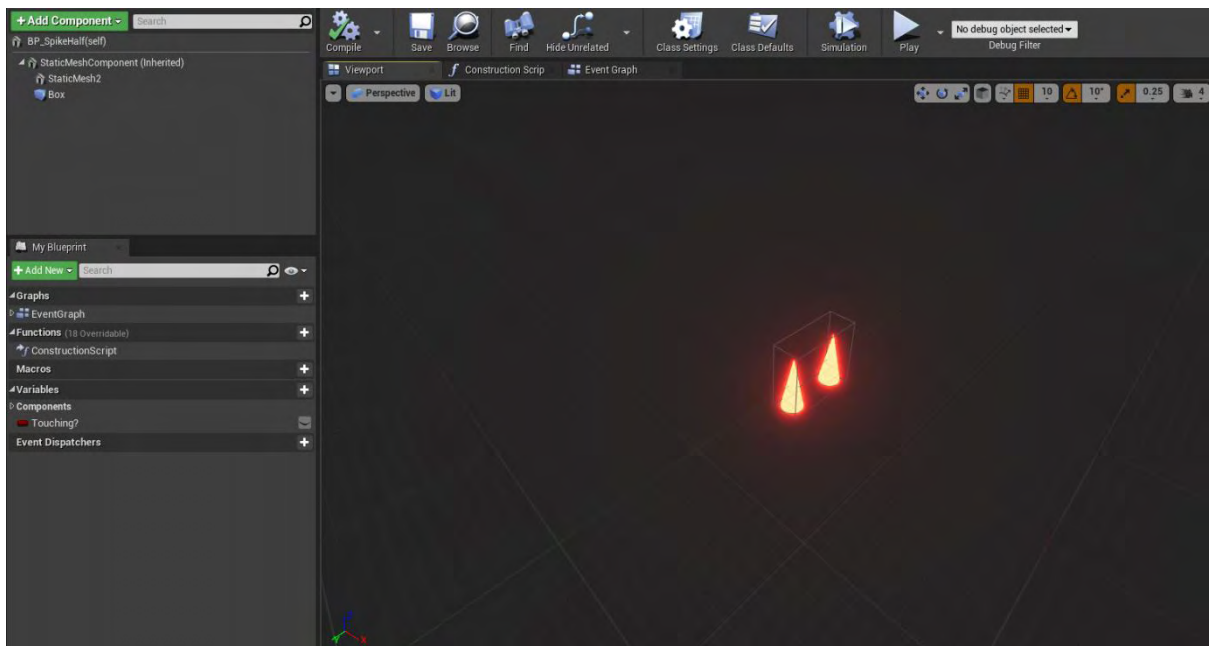
Στην Εικόνα 116 φαίνονται οι ρυθμίσεις της μεταβλητής Touching?.



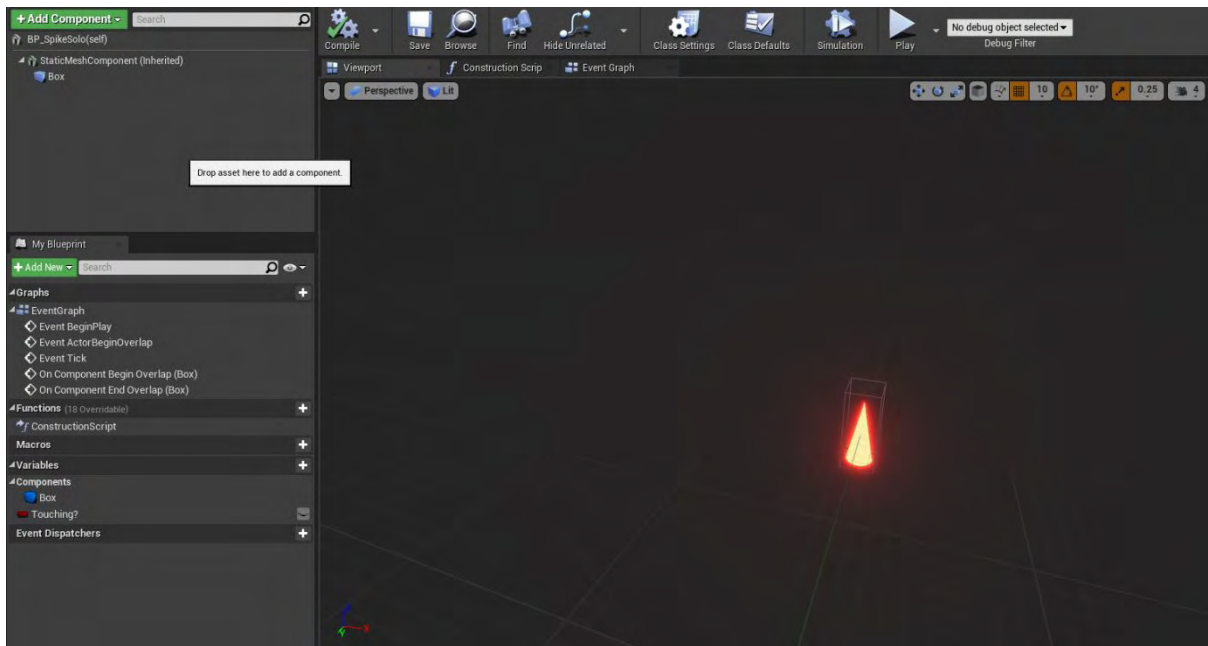


Εικόνα 116 BP\_Spike part 5

Στην Εικόνα 117 και στην Εικόνα 118 απεικονίζονται οι actors BP\_SpikeHalf και BP\_SpikeSolo. Δεν θα αναλυθούν περαιτέρω εφόσον είναι παρόμοια με το BP\_Spike.

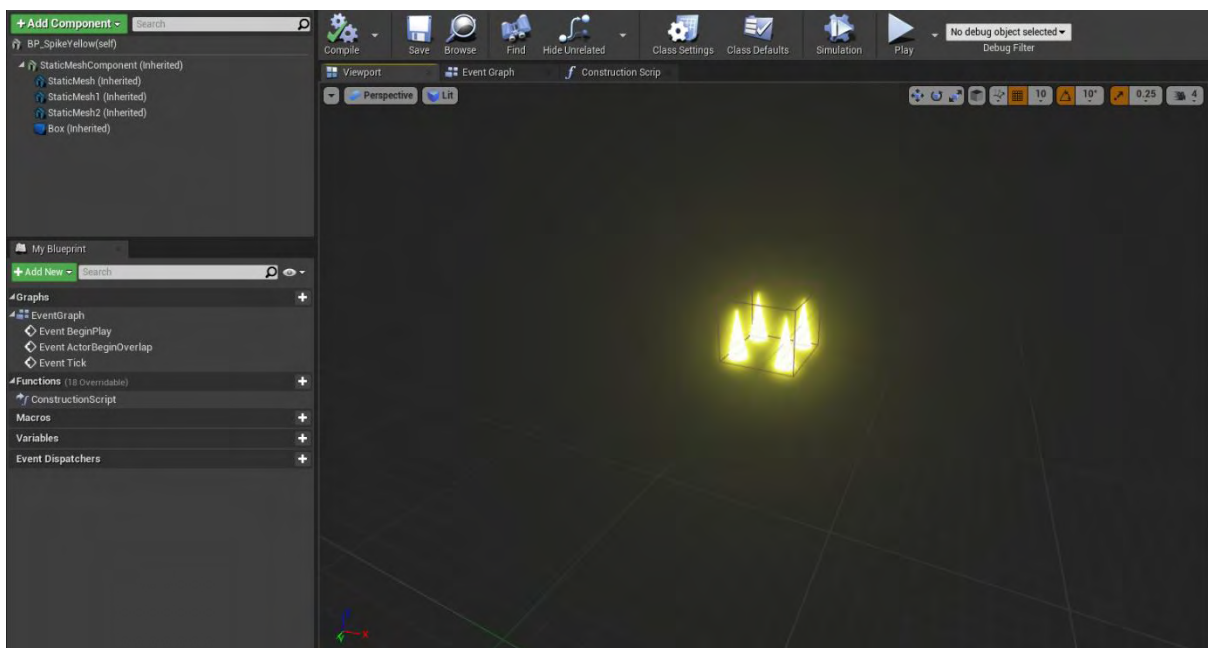


Εικόνα 117 BP\_SpikeHalf



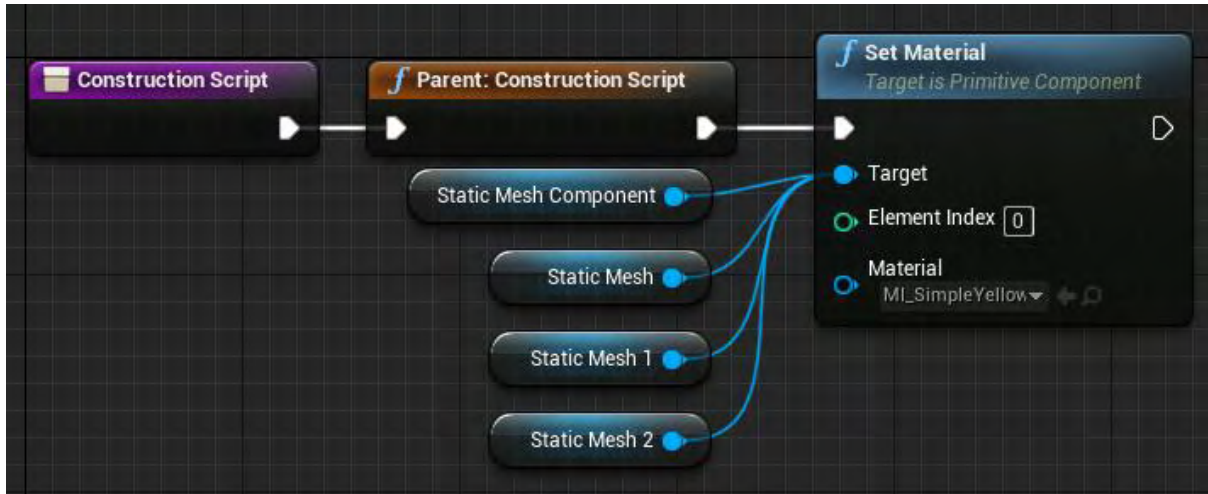
Εικόνα 118 BP\_SpikeSolo

Το BP\_SpikeYellow είναι παιδί του BP\_Spike (Εικόνα 119).



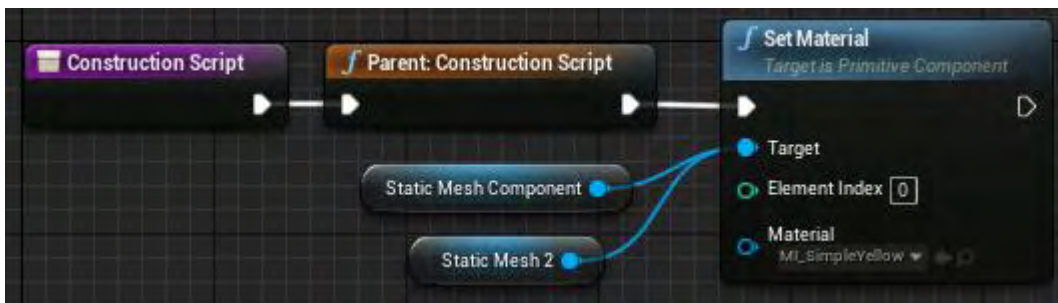
Εικόνα 119 BP\_SpikeYellow part 1

Στην Εικόνα 120 μεταβάλεται το υλικό των Static Mesh Components στο MI\_SimpleYellow.



Εικόνα 120 BP\_SpikeYellow part 2

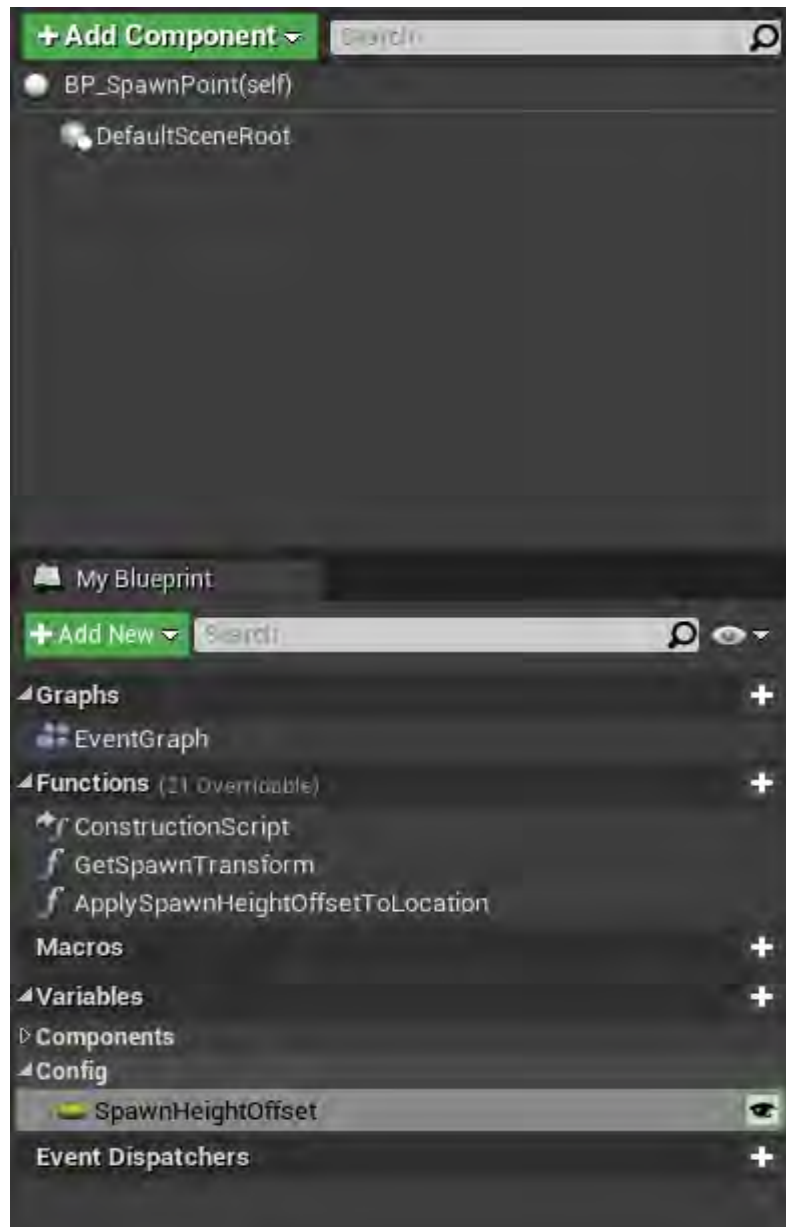
Στην Εικόνα 121 μεταβάλλεται το υλικό των Static Mesh Components του BP\_SpikeYellowHalf στο MI\_SimpleYellow



Εικόνα 121 BP\_SpikeYellowHalf

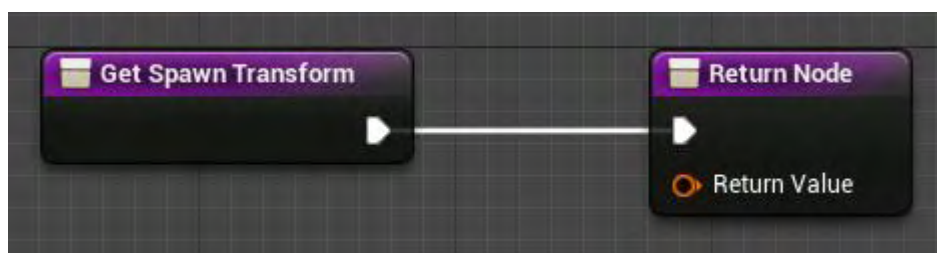
## Ενότητα 5.1.20 BP\_SpawnPoint

Στην ενότητα αυτή αναλύεται το BP\_SpawnPoint και τα παιδιά του. Αρχικά το BP\_SpawnPoint δεν αποτελείται από κάποιο component και υλοποιεί τις βασικές συναρτήσεις `GetSpawnTransform` και `ApplySpawnHeightOffsetToLocation` (Εικόνα 122).



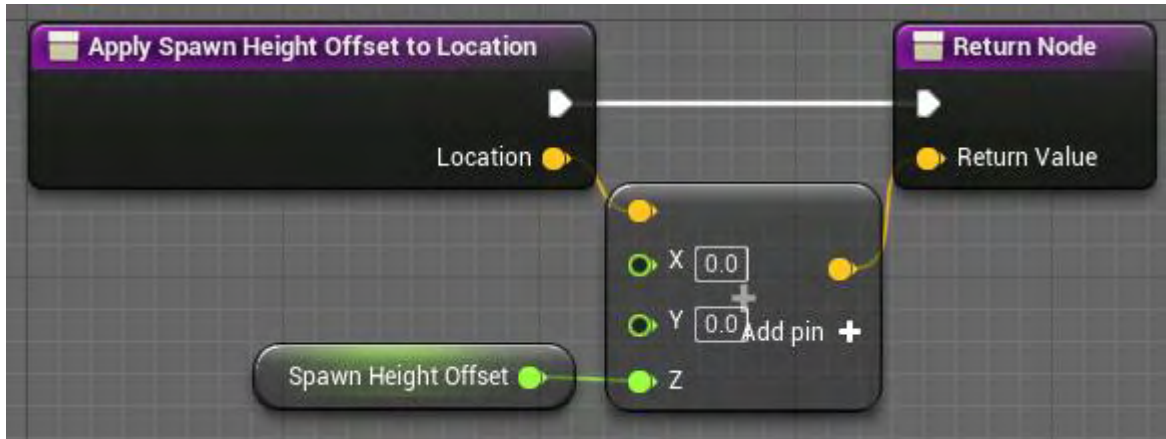
Εικόνα 122 BP\_SpawnPoint part 1

Στην Εικόνα 123 απεικονίζεται η συνάρτηση GetSpawnTransform η οποία επιστρέφει μια τιμή μεταβλητής Transform (τοποθεσία, προσανατολισμό και κλίμακα).



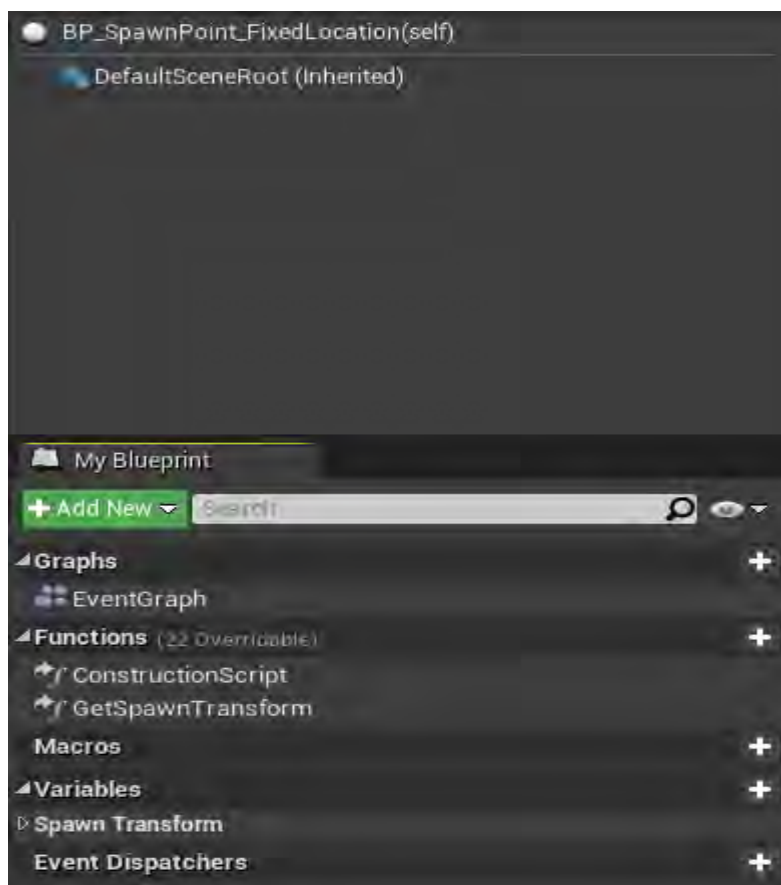
Εικόνα 123 BP\_SpawnPoint part 2

Η συνάρτηση `ApplySpawnHeightOffsetToLocation` επιστρέφει την είσοδο (`Location`) της συνάρτησης τύπου `Vector` τροποποιημένη ανάλογα με την τιμή της μεταβλητής `SpawnHeightOffset` (Εικόνα 124).



Εικόνα 124 BP\_SpawnPoint part 3

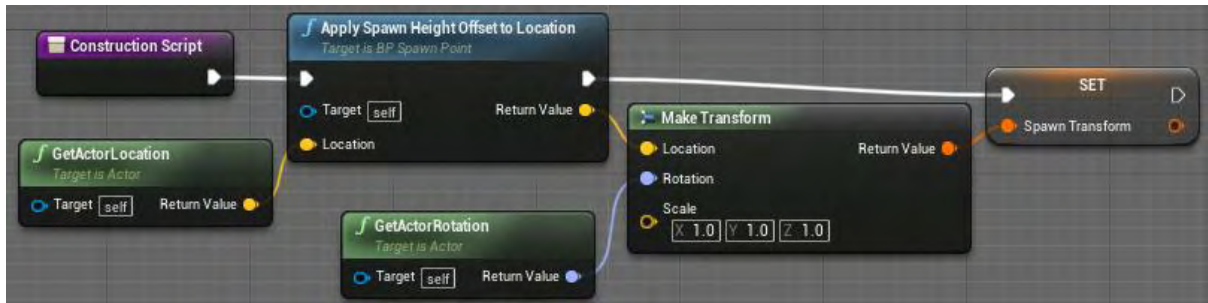
Το παιδί `BP_SpawnPointFixedLocation` φαίνεται στην Εικόνα 125.



Εικόνα 125 BP\_SpawnPoint\_FixedLocation part 1



Στο construction καλείται η συνάρτηση ApplySpawnHeightOffsetToLocation με είσοδο την τοποθεσία του actor. Η νέα τιμή της τοποθεσίας μαζί με τον προσανατολισμό του actor μετατρέπονται σε μεταβλητή τύπου Transform. Αυτή ανατίθεται στη μεταβλητή SpawnTransform (Εικόνα 126).



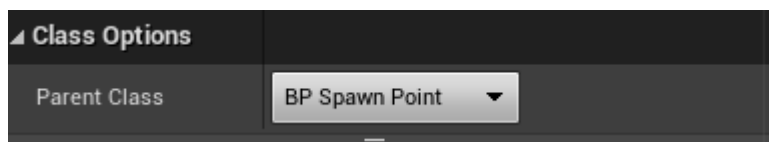
Εικόνα 126 BP\_SpawnPoint\_FixedLocation part 2

Η συνάρτηση GetSpawnTransform επιστρέφει τη μεταβλητή SpawnTransform (Εικόνα 127).



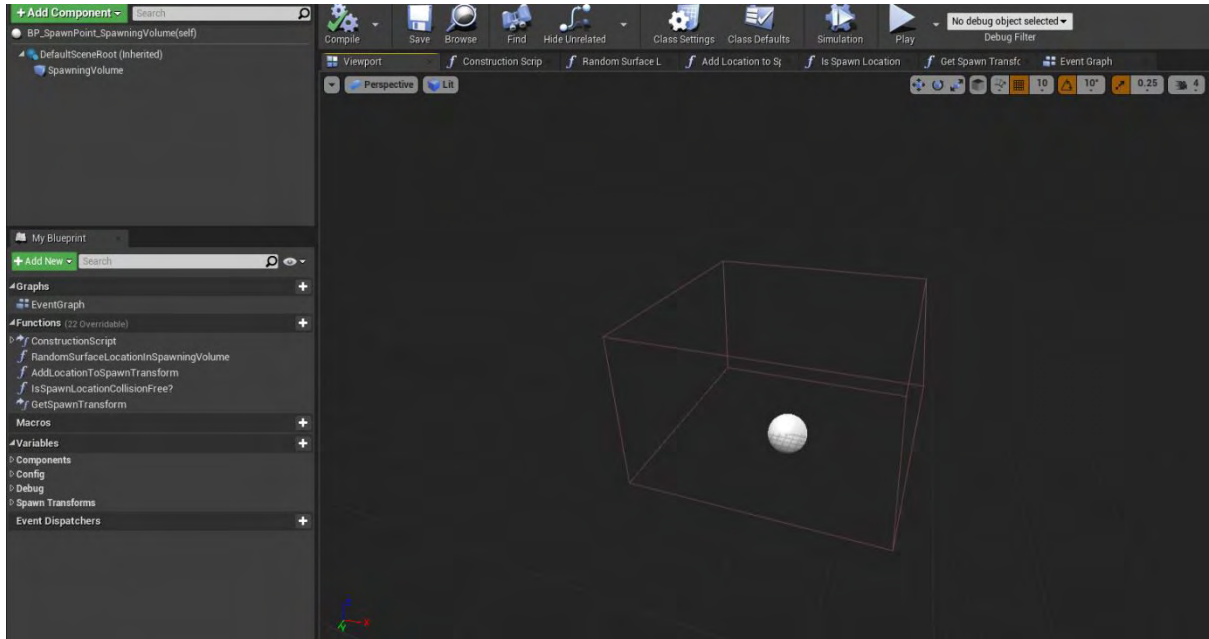
Εικόνα 127 BP\_SpawnPoint\_FixedLocation part3

Το BP\_SpawnPoint\_FixedLocation είναι παιδί της κλάσης BP\_SpawnPoint και για αυτό κληρονομεί και τις συναρτήσεις (Εικόνα 128).



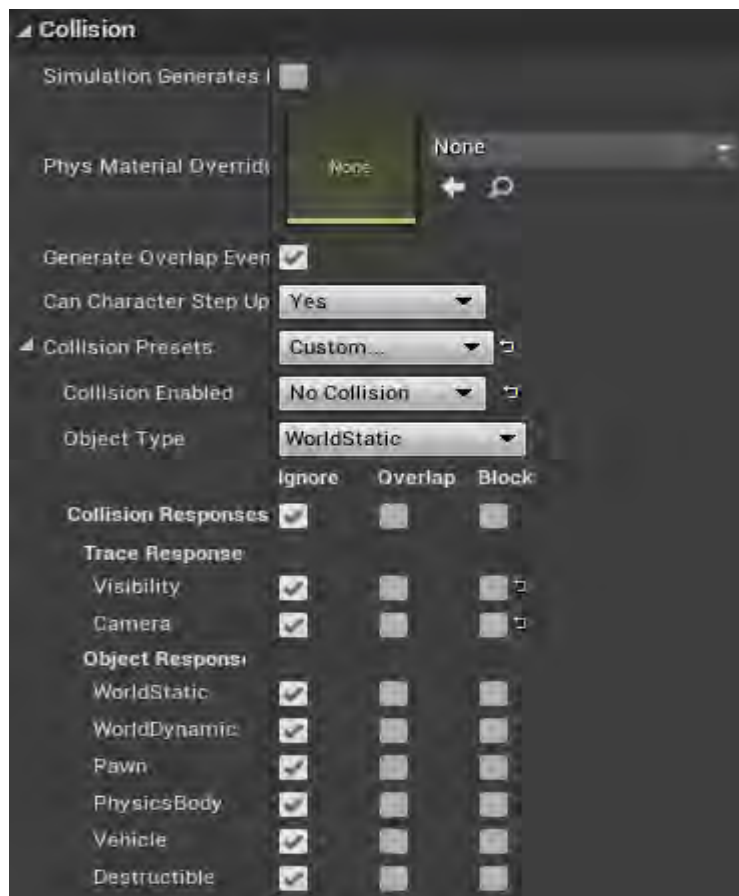
Εικόνα 128 BP\_SpawnPoint\_FixedLocation part4

Το BP\_SpawnPoint\_Spawning αποτελείται από το SpawningVolume τύπου Box Collision (Εικόνα 129).



Εικόνα 129 BP\_SpawnPoint\_Spawning part 1

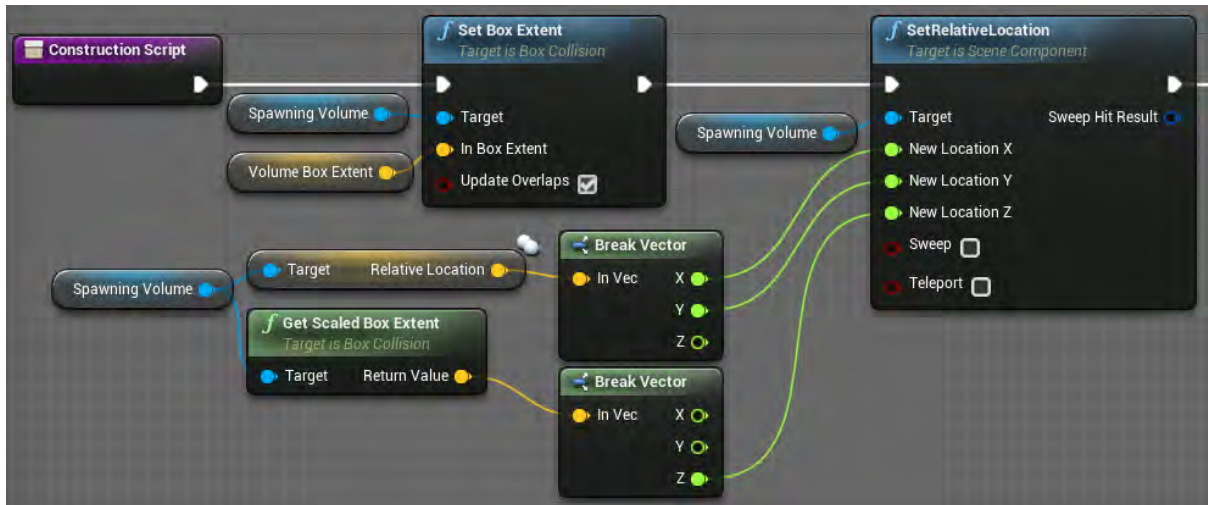
Στην Εικόνα 130 απεικονίζονται οι ρυθμίσεις σύγκρουσης του SpawningVolume.



Εικόνα 130 BP\_SpawnPoint\_Spawning part 2

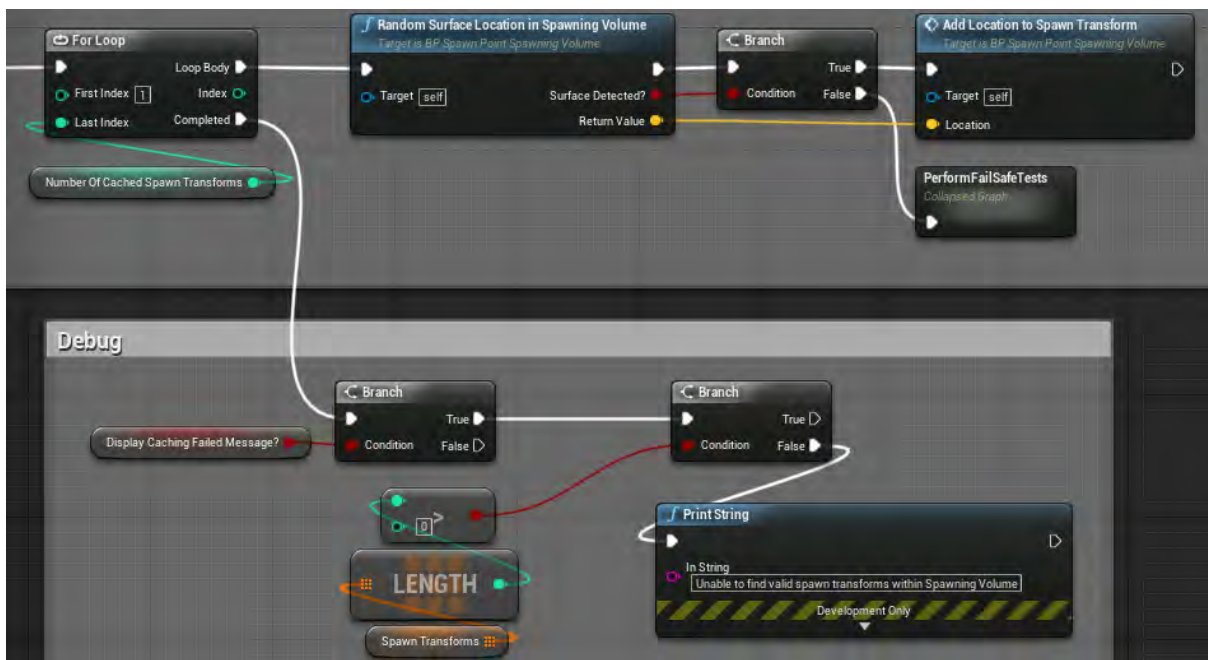


Στο Construction Script αρχικοποιείται το SpawningVolume (Εικόνα 131).



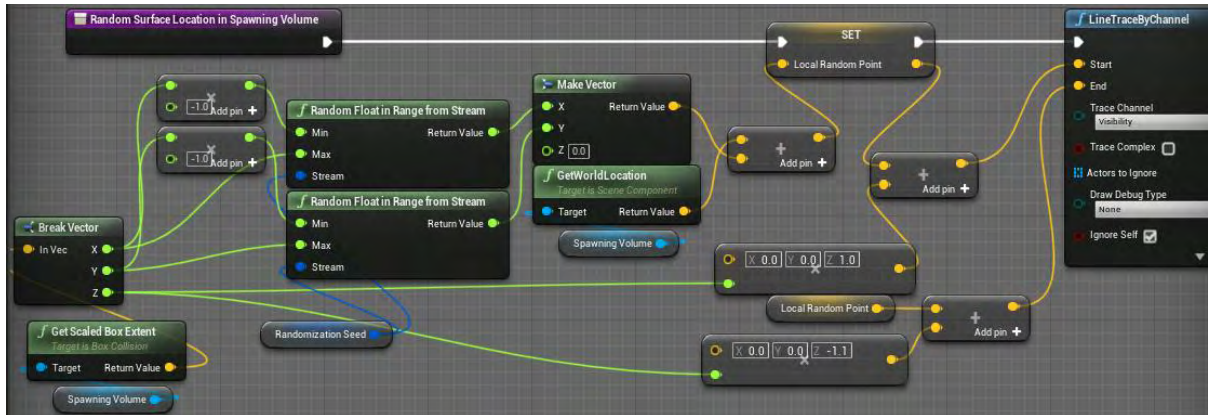
Εικόνα 131 BP\_SpawnPoint\_Spawning part 3

Υπολογίζονται οι τυχαίες τιμές του πίνακα SpawnTransform, χρησιμοποιώντας τις συναρτήσεις RandomSurfaceLocationInSpawningVolume και AddLocationToSpawnTransform (Εικόνα 132).

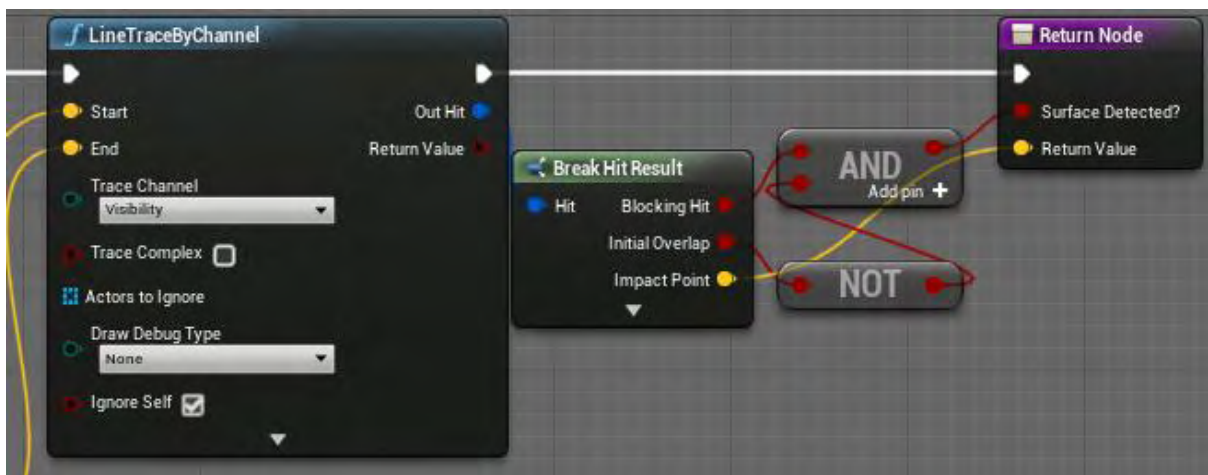


Εικόνα 132 BP\_SpawnPoint\_Spawning part 4

Στην Εικόνα 133 και στην Εικόνα 134 απεικονίζεται η συνάρτηση RandomSurfaceLocationInSpawnVolume, η οποία επιστρέφει τυχαίο σημείο ενός συγκεκριμένου επιπέδου μέσα στα όρια του SpawningVolume.

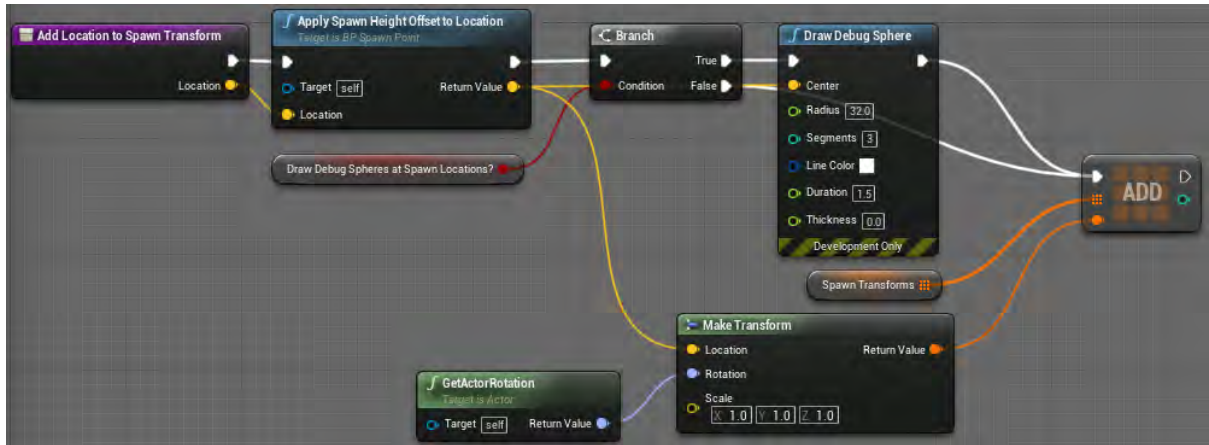


Εικόνα 133 BP\_SpawnPoint\_Spawning part 5



Εικόνα 134 BP\_SpawnPoint\_Spawning part 6

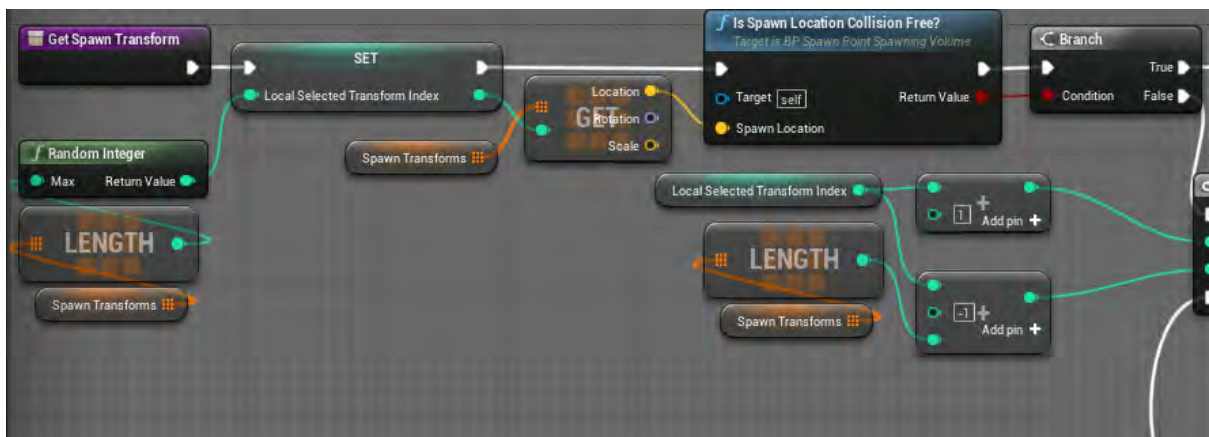
Η συνάρτηση AddLocationToSpawnTransform προσθέτει την τοποθεσία που δέχεται σαν εισοδο στον πίνακα SpawnTransform (Εικόνα 135).



Εικόνα 135 BP\_SpawnPoint\_Spawning part 7

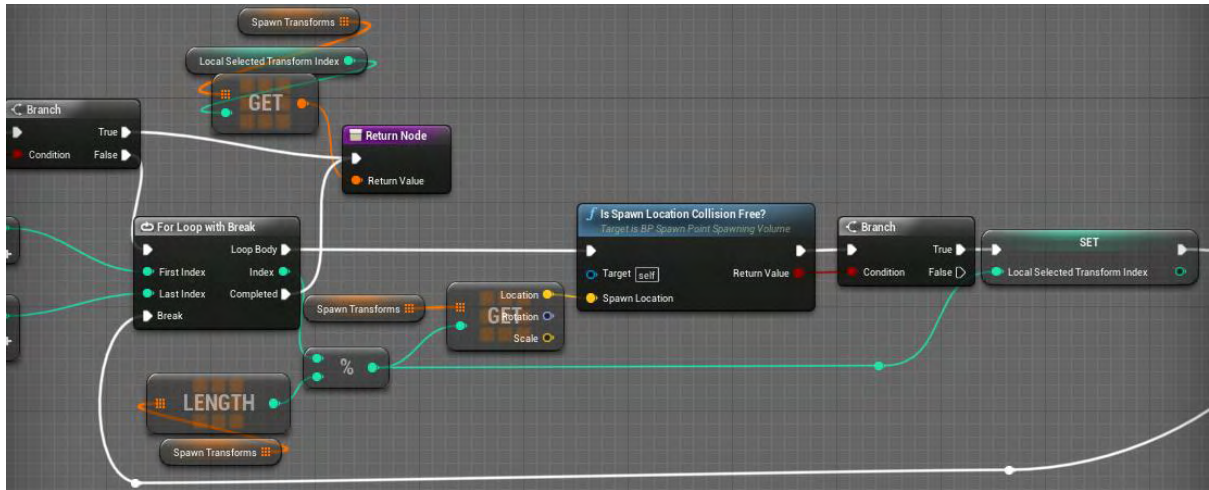
Η συνάρτηση GetSpawnTransform, που απεικονίζεται στην Εικόνα 136 και στην Εικόνα 137, επιστρέφει ένα τυχαίο σημείο από τον πίνακα SpawnTransform. Το σημείο αυτό ελέγχεται με τη συνάρτηση IsSpawnLocationCollisionFree?.

- Αν η τιμή που επιστρέφει η IsSpawnLocationCollisionFree? είναι αληθής τότε η GetSpawnTransform επιστρέφει το σημείο αυτό.
- Αν η τιμή που επιστρέφει η IsSpawnLocationCollisionFree? είναι ψευδής τότε ελέγχονται και άλλα σημεία ώπου η τιμή που θα επιστρέψει η IsSpawnLocationCollisionFree? να είναι αληθής.



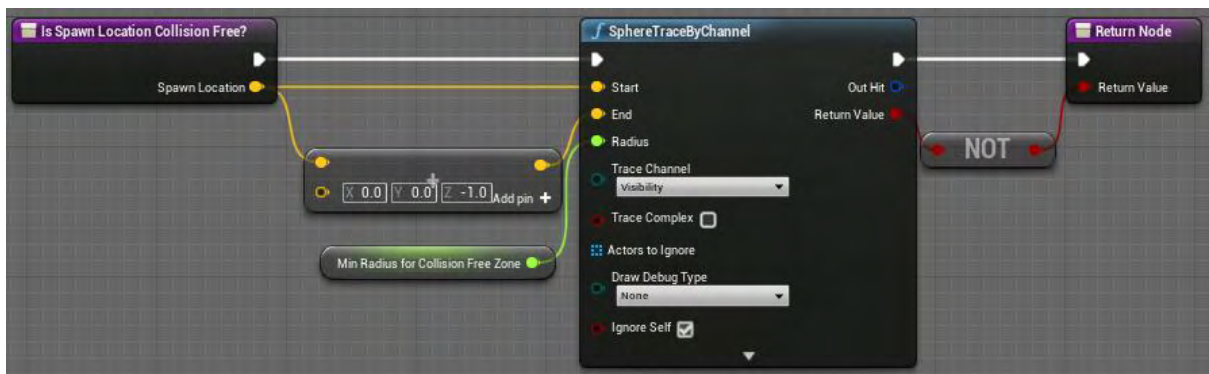
Εικόνα 136 BP\_SpawnPoint\_Spawning part 8





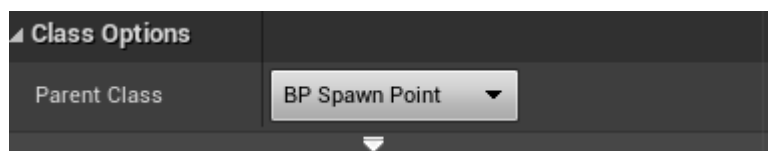
Εικόνα 137 BP\_SpawnPoint\_Spawning part 9

Η συνάρτηση IsSpawnLocationCollisionFree? δέχεται ως είσοδο μια τιμή τύπου Vector. Κατόπιν ελέγχει για συγκρούσεις χρησιμοποιώντας τον κόμβο SphereTraceByChannel και επιστρέφει το αποτέλεσμα του ελέγχου (Εικόνα 138).



Εικόνα 138 BP\_SpawnPoint\_Spawning part 10

Στην Εικόνα 139 φαίνεται ότι το BP\_SpawnPoint\_Spawning είναι παιδί της κλάσης BP\_SpawnPoint.

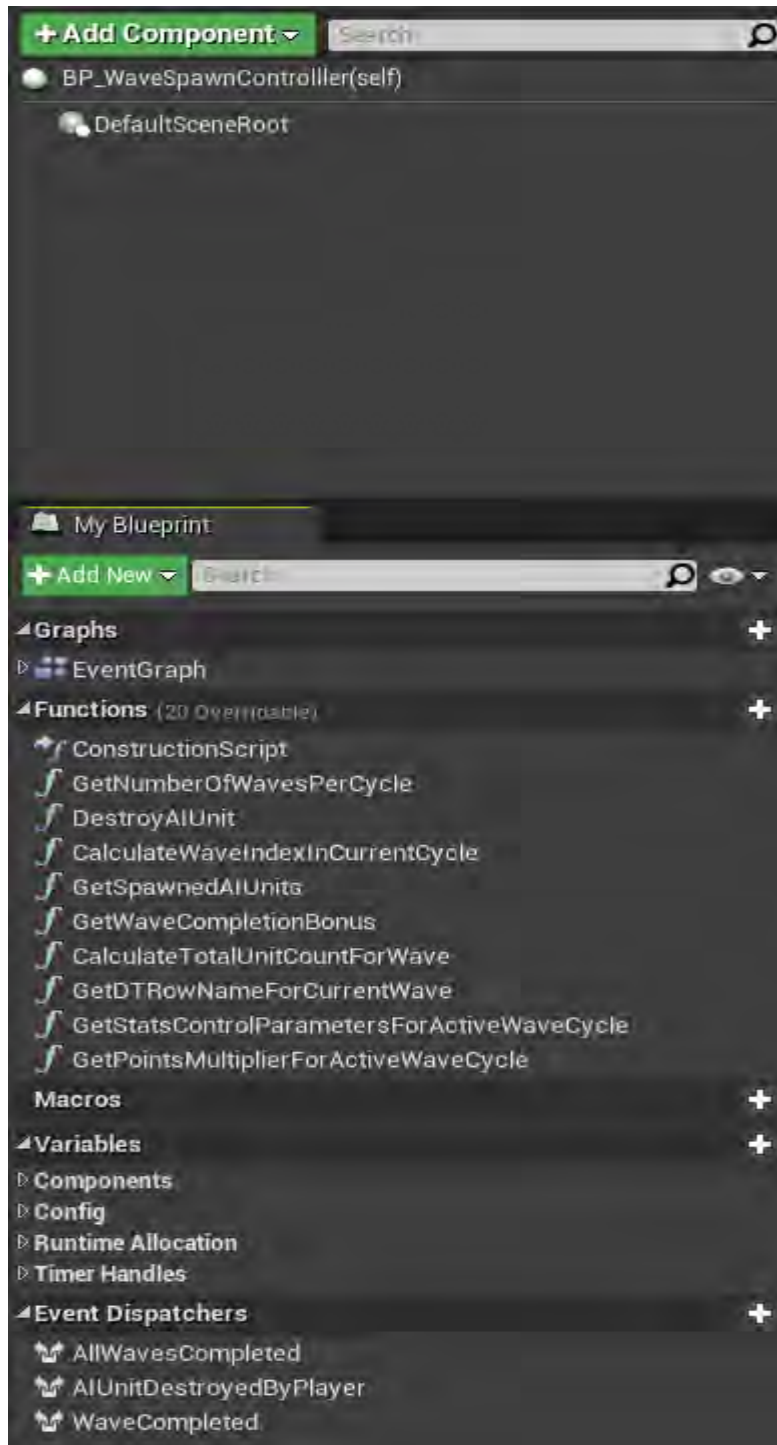


Εικόνα 139 BP\_SpawnPoint\_Spawning part 11

## Ενότητα 5.1.21 BP\_WaveSpawnController

---

Στην Εικόνα 140 απεικονίζονται οι συναρτήσεις που υλοποιούνται στο BP\_WaveSpawnController.



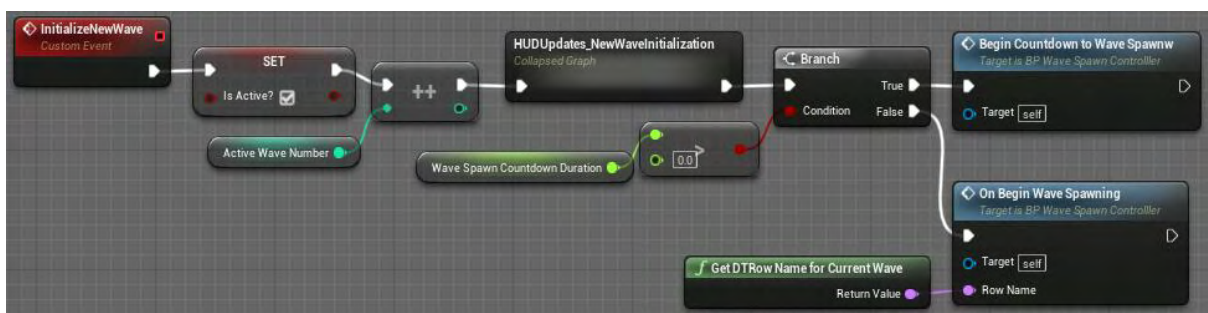
Εικόνα 140 BP\_WaveSpawnController part 1

Μόλις το BP\_WaveSpawnController ενεργοποιηθεί, μετά τους απαραίτητους ελέγχους, καλείται η συνάρτηση InitializeNewWave (Εικόνα 141).



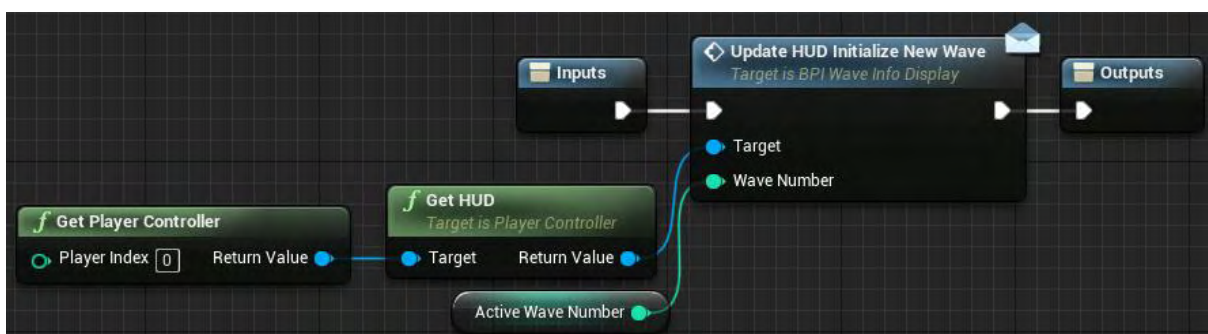
Εικόνα 141 BP\_WaveSpawnController part 2

Η συνάρτηση InitializeNewWave αρχικά ενημερώνει το HUD του χαρακτήρα του παίκτη. Στη συνέχεια, αφού πραγματοποιηθεί ο έλεγχος της τιμής WaveSpawnCountdownDuration, είτε καλεί τη συνάρτηση BeginCountDownToWaveSpawner είτε τη συνάρτηση OnBeginWaveSpawning, οι οποίες θα αναλυθούν παρακάτω (Εικόνα 142).



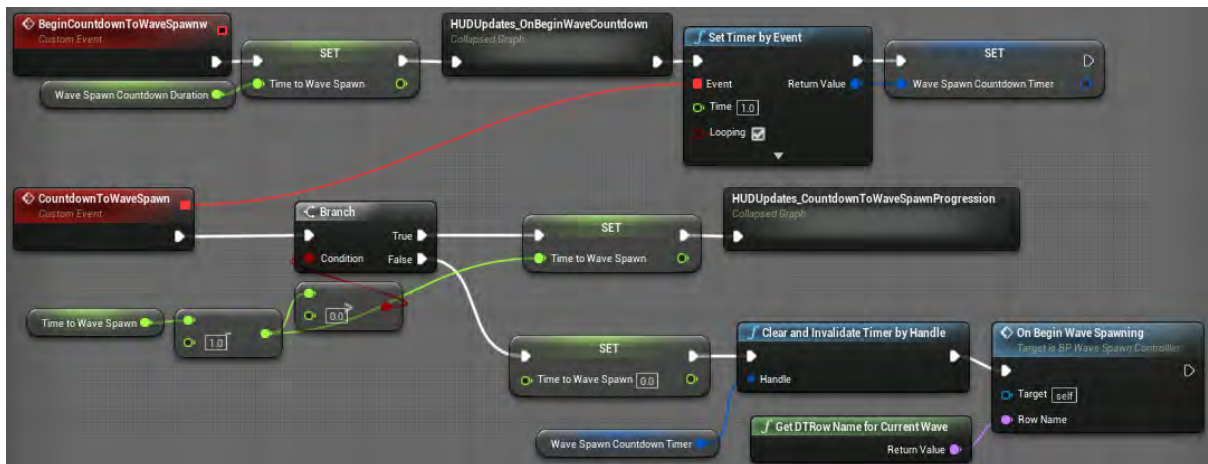
Εικόνα 142 BP\_WaveSpawnController part 3

Στο Collapsed Graph της προηγούμενης εικόνας, καλείται η συνάρτηση UpdateHUDInitializeNewWave της διεπαφής BPI\_WaveInfoDisplay, που θα αναλυθεί παρακάτω, με εισοδο την τιμή της μεταβλητής ActiveWaveNumber (Εικόνα 143).



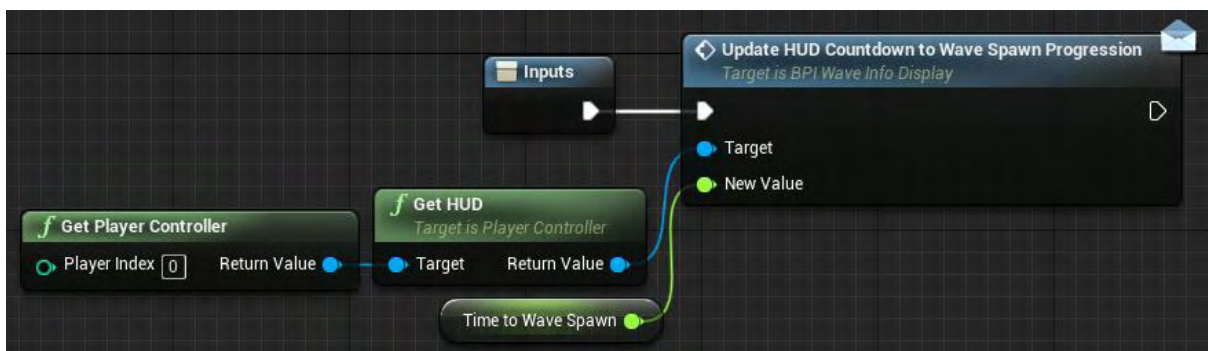
Εικόνα 143 BP\_WaveSpawnController part 4

Η συνάρτηση `BeginCountDownToWaveSpawner` αρχικοποιεί τον μετρητή για το wave spawn και στη συνέχεια δεσμεύει τη συνάρτηση `CountdownToWaveSpawn` ώστε να ενημερώνει την τιμή του μετρητή κάθε δευτερόλεπτο. Όταν η τιμή του μετρητή μηδενιστεί, καλεί τη συνάρτηση `OnBeginWaveSpawning` με είσοδο την έξοδο της συνάρτησης `GetDTRowNameForCurrentWave` (Εικόνα 144).



Εικόνα 144 BP\_WaveSpawnController part 5

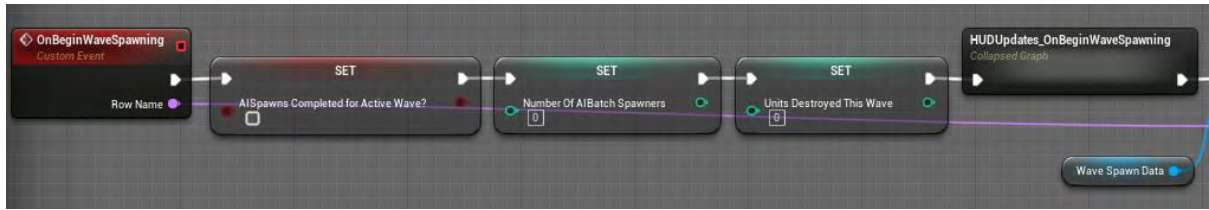
Στο Collapsed Graph της προηγούμενης εικόνας, καλείται, η συνάρτηση `UpdateHUDCountdownToWaveProgression` της διεπαφής `BPI_WaveInfoDisplay`, που θα αναλυθεί παρακάτω, με είσοδο την τιμή της μεταβλητής `TimeToWaveSpawn` (Εικόνα 145).



Εικόνα 145 BP\_WaveSpawnController part 6

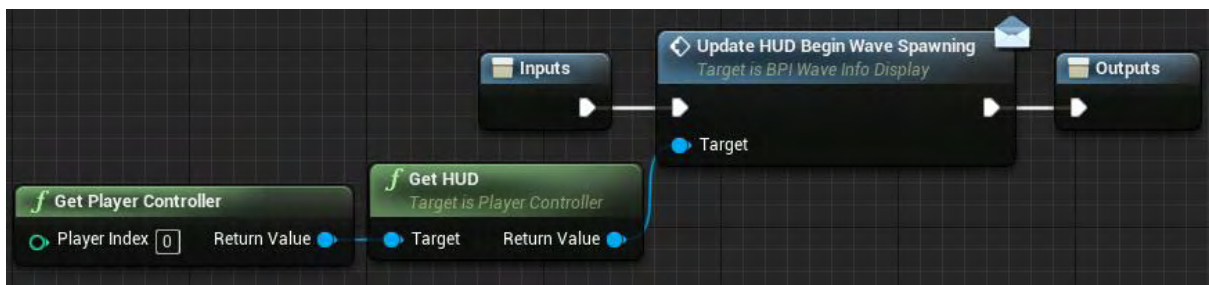
Η συνάρτηση `OnBeginWaveSpawning` αρχικοποιεί τις μεταβλητές `AISpawnsCompletedForActiveWave?`, `NumberOfAIBatchSpawners` και `UnitsDestroyedThisWave` (Εικόνα 146).





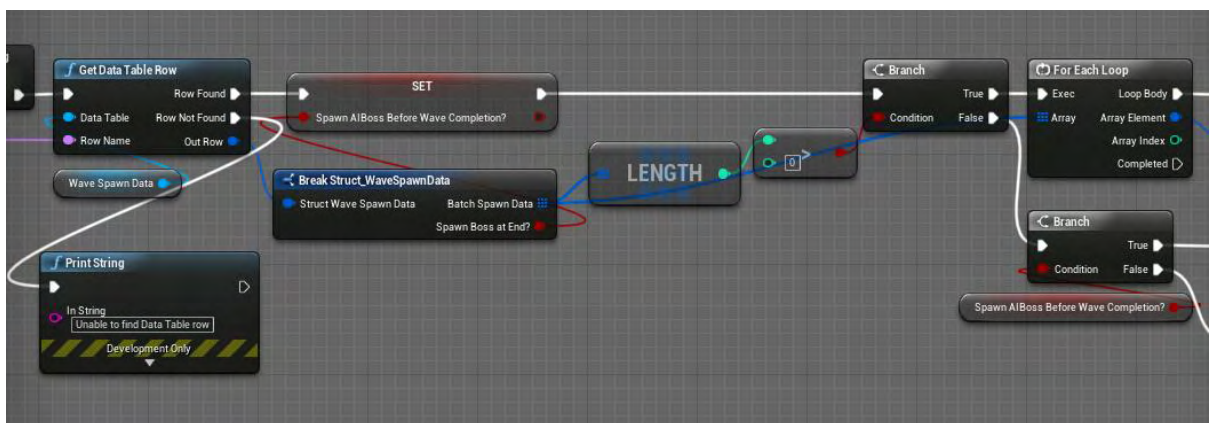
Εικόνα 146 BP\_WaveSpawnController part 7

Στο Collapsed Graph που απεικονίζεται στην παραπάνω εικόνα καλείται η συνάρτηση UpdateHUDBeginWaveSpawning όπως φαίνεται στην Εικόνα 147.

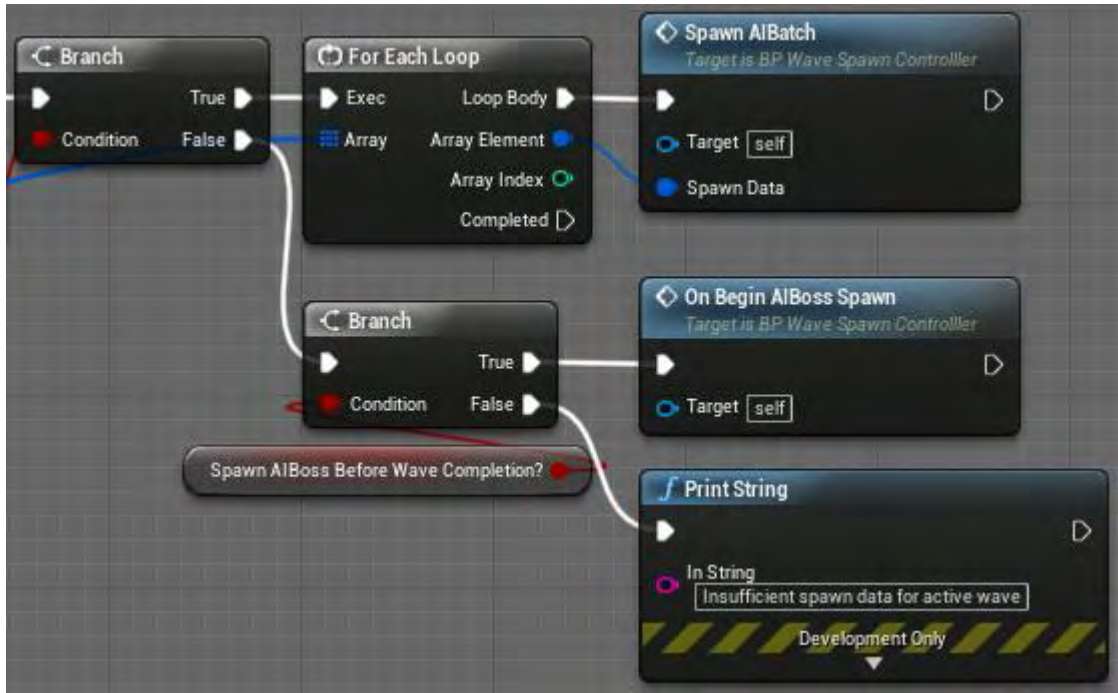


Εικόνα 147 BP\_WaveSpawnController part 8

Στην Εικόνα 148 και στην Εικόνα 149, στις οποίες απεικονίζεται η συνέχεια της συνάρτησης OnBeginWaveSpawning, καλούνται από την παραπάνω συνάρτηση οι SpawnAIBatch (για όλα τα στοιχεία του πίνακα δεδομένων της μεταβλητής WaveSpawnData) και OnBeginAIBossSpawn,

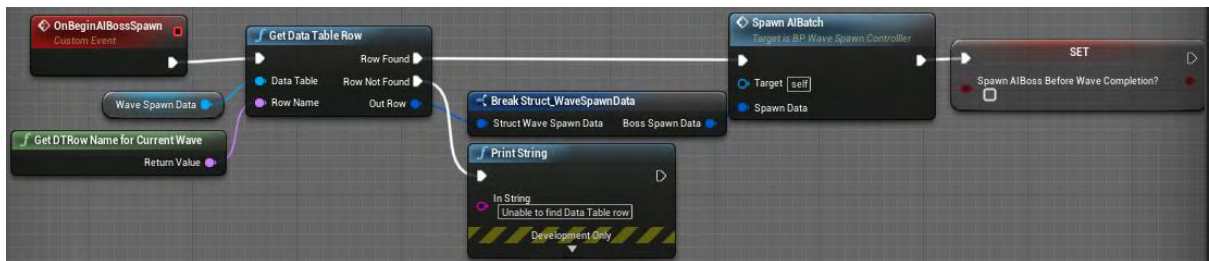


Εικόνα 148 BP\_WaveSpawnController part 9



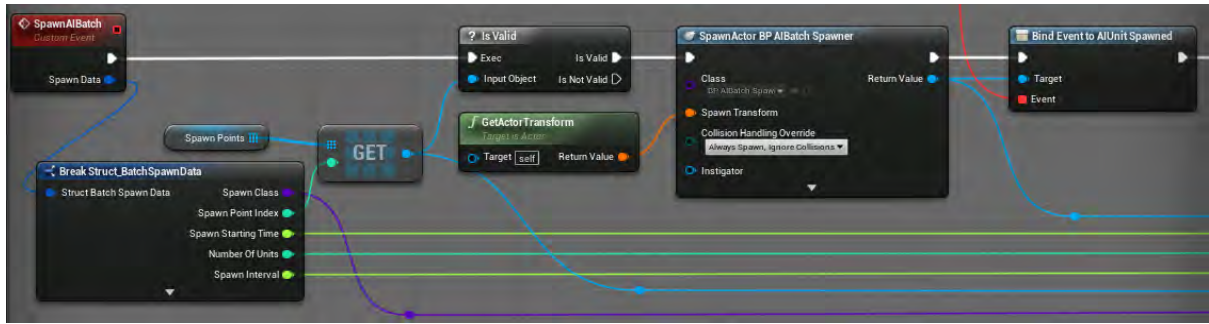
Εικόνα 149 BP\_WaveSpawnController part 10

Η συνάρτηση `OnBeginAIBossSpawn` χρησιμοποιώντας την `GetDTRowNameForeCurrent` αντλεί τα απαραίτητα δεδομένα από την αντιστοιχη γραμμή του πίνακα της μεταβλητής `WaveSpawnData`. Στη συνέχεια καλεί τη συνάρτηση `SpawnAIBatch` με εισοδο τα δεδομένα της γραμμής (Εικόνα 150).



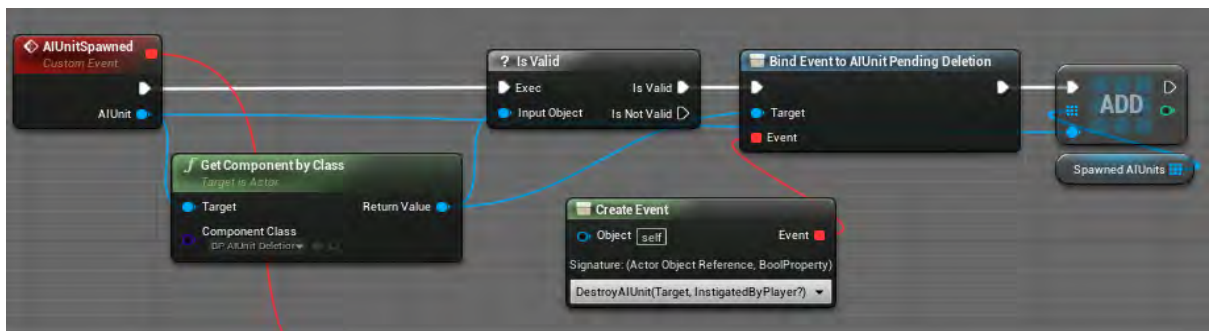
Εικόνα 150 BP\_WaveSpawnController part 11

Από τη συνάρτηση `SpawnAIBatch` πραγματοποιείται ο απαραίτητος έλεγχος για την εγκυρότητα της τιμής της μεταβλητής `SpawnPointIndex`. Ακολούθως δημιουργείται ένας actor τύπου `BP_AIBatchSpawner` με χρήση το `Transform` του αντιστοιχου στοιχείου του πίνακα `SpawnPoints`. Δεσμεύεται η συνάρτηση `AiUnitSpawned` (Εικόνα 151).



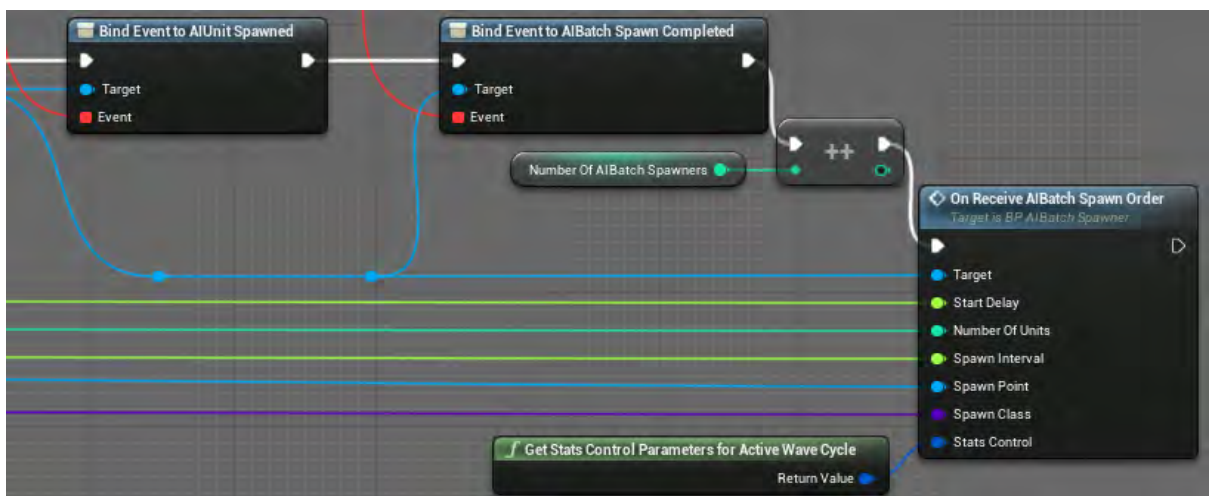
Εικόνα 151 BP\_WaveSpawnController part 12

Η συνάρτηση AiUnitSpawned προσθέτει τον δημιουργημένο actor BP\_AIBatchSpawner στον πίνακα SpawnedAIUnits (Εικόνα 152).



Εικόνα 152 BP\_WaveSpawnController part 13

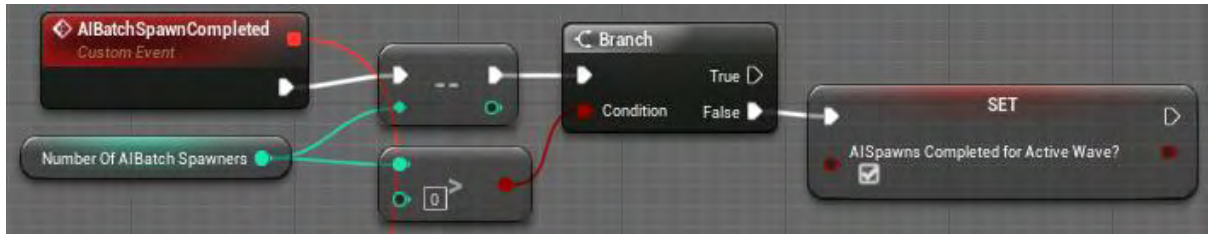
Στη συνέχεια η συνάρτηση SpawnAIBatch δεσμεύει ξανά στον δημιουργημένο actor τη συνάρτηση AIBatchSpawnCompleted, αυξάνει τη μεταβλητή NumberOfAIBatchSpawners και κατόπιν καλεί την OnReceiveAIBatchSpawnOrder του actor BP\_AIBatchSpawner (Εικόνα 153).



Εικόνα 153 BP\_WaveSpawnController part 14

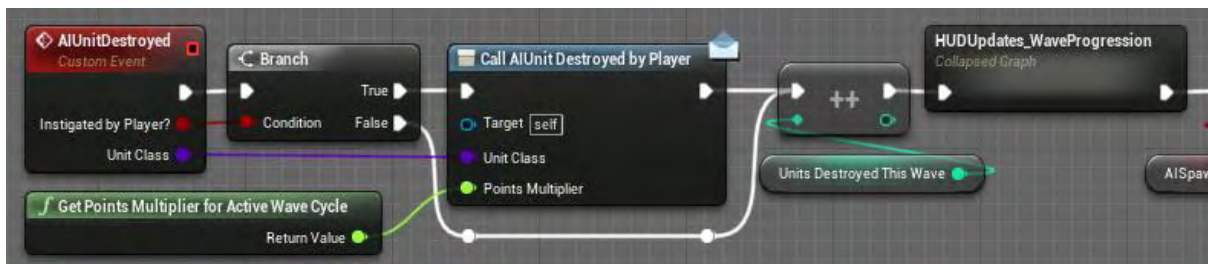


Η συνάρτηση `AIBatchSpawnCompleted` ελέγχει αν έχουν δημιουργηθεί όλοι οι actors τύπου `BP_AIBatchSpawner`. Αν έχουν δημιουργηθεί, τότε ανατίθεται στη μεταβλητή `AISpawnsCompletedForActiveWave?` η τιμή `true` (Εικόνα 154).



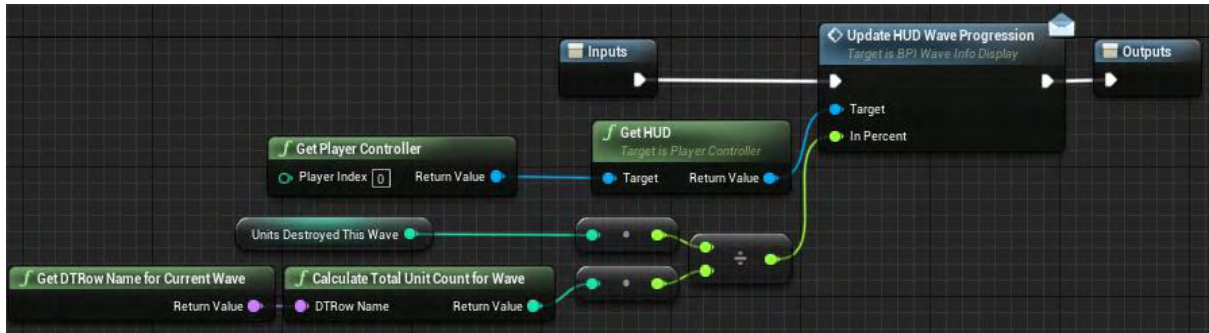
Εικόνα 154 BP\_ WaveSpawnController part 15

Η συνάρτηση `AIUnitDestroyed` καλεί τον Event Dispatcher `AIUnitDestroyedByPlayer`, αν ο actor έχει καταστραφεί από τον χαρακτήρα του παίκτη, και στη συνέχεια ενημερώνει τη μεταβλητή `UnitsDestroyedThisWave` (Εικόνα 155).



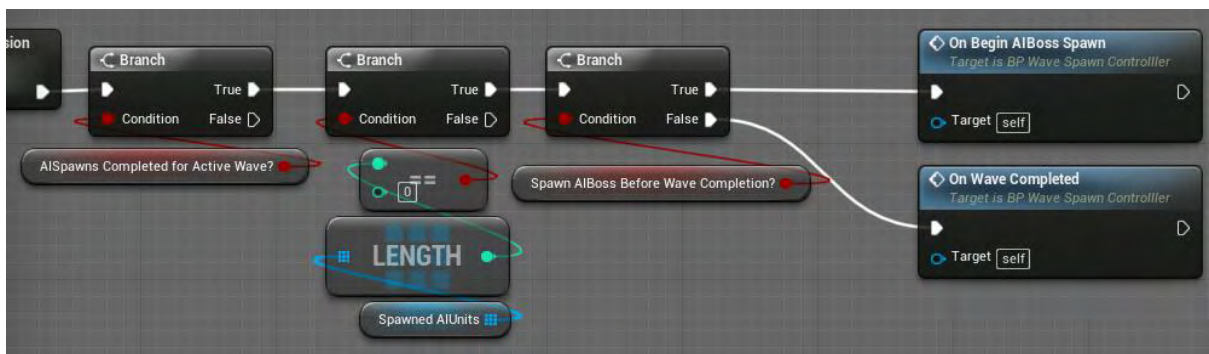
Εικόνα 155 BP\_ WaveSpawnController part 16

Στο Collapsed Graph που απεικονίζεται στην παραπάνω εικόνα, με τη χρήση των συναρτήσεων `GetDTRowNameForCurrentWave`, `CalculateTotalUnitCountForWave` και τη μεταβλητή `UnitsDestroyedThisWave` υπολογίζεται ως ποιο σημείο έχει ολοκληρωθεί το Wave και μετατρέπεται σε ποσοστό. Καλείται η συνάρτηση `UpdateHUDWaveProgression` με είσοδο το ποσοστό αυτό (Εικόνα 156).



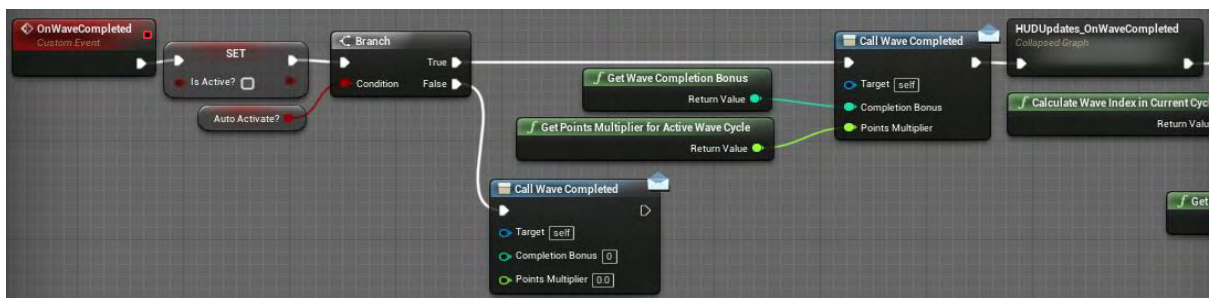
Εικόνα 156 BP\_WaveSpawnController part 17

Κατόπιν η συνάρτηση AIUnitDestroyed ελέγχει αν υπάρχουν και άλλα στοιχεία στον πίνακα SpawnedAIUnits και αναλόγως με το αν ο συγκεκριμένος actor είναι ρυθμισμένος για την εμφάνιση κάποιου Boss unit ή όχι, καλεί τις συναρτήσεις OnBeginAIBossSpawn και OnWaveCompleted αντίστοιχα (Εικόνα 157).



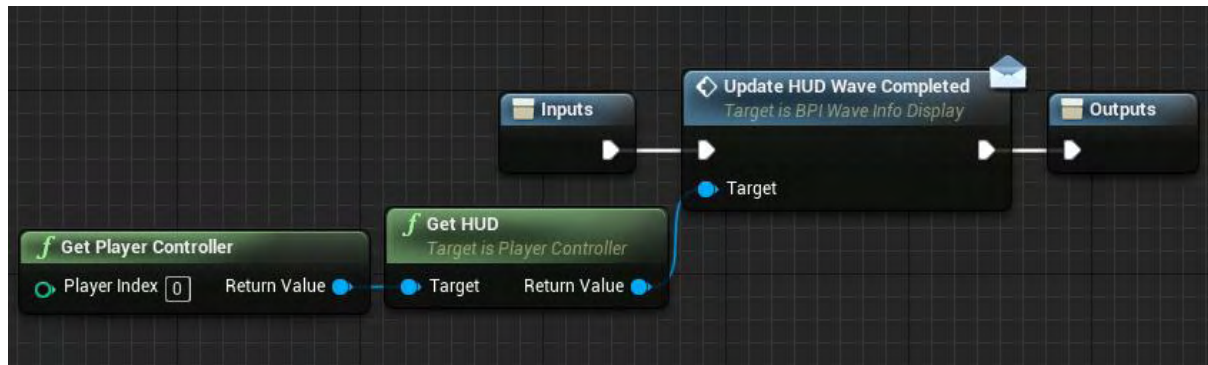
Εικόνα 157 BP\_WaveSpawnController part 18

Η συνάρτηση OnWaveCompleted αναλόγως με την τιμή AutoActivate του actor BP\_WaveSpawnController είτε καλεί απευθείας τον Event Dispatcher Wave Completed είτε καλεί πάλι τον ίδιο Dispatcher χρησιμοποιώντας τις όμως εξόδους των συναρτήσεων GetWaveCompletionBonus και GetPointsMultiplierForActiveWaveCycle για εισοδο του (Εικόνα 158).



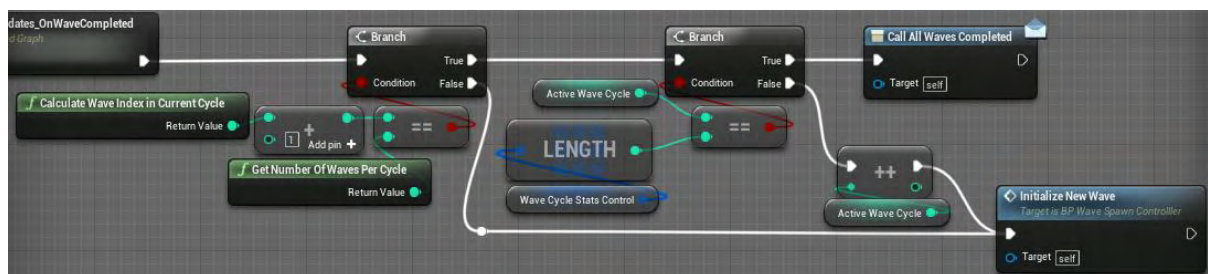
Εικόνα 158 BP\_WaveSpawnController part 19

Το Collapsed Graph της παραπάνω εικόνας είναι η Εικόνα 159 και απεικονίζει το κάλεσμα της συνάρτησης UpdateHUDWaveCompleted του BPI\_WaveInfoDisplay (Εικόνα 159).



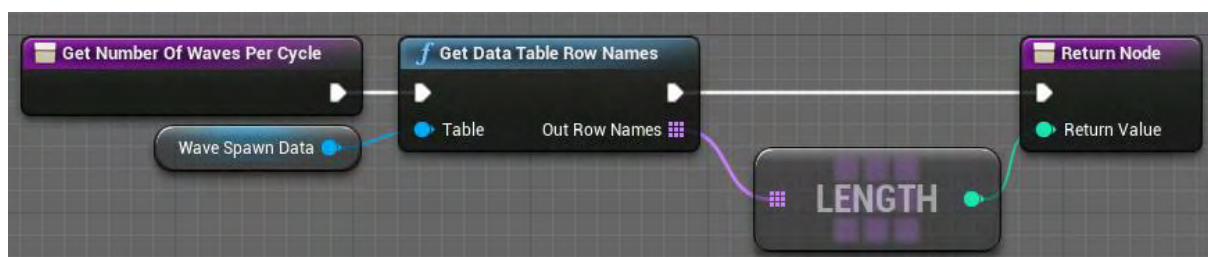
Εικόνα 159 BP\_WaveSpawnController part 20

Η συνέχεια της συνάρτησης OnWaveCompleted καλεί τον Event Dispatcher InitializeNewWave και όταν πλέον δεν υπάρχει επόμενο wave, καλεί τον Event Dispatcher AllWavesCompleted (Εικόνα 160).



Εικόνα 160 BP\_WaveSpawnController part 21

Η συνάρτηση GetNumberOfWavesPerCycle επιστρέφει τον αριθμό των κυμάτων (Εικόνα 161).



Εικόνα 161 BP\_WaveSpawnController part 22

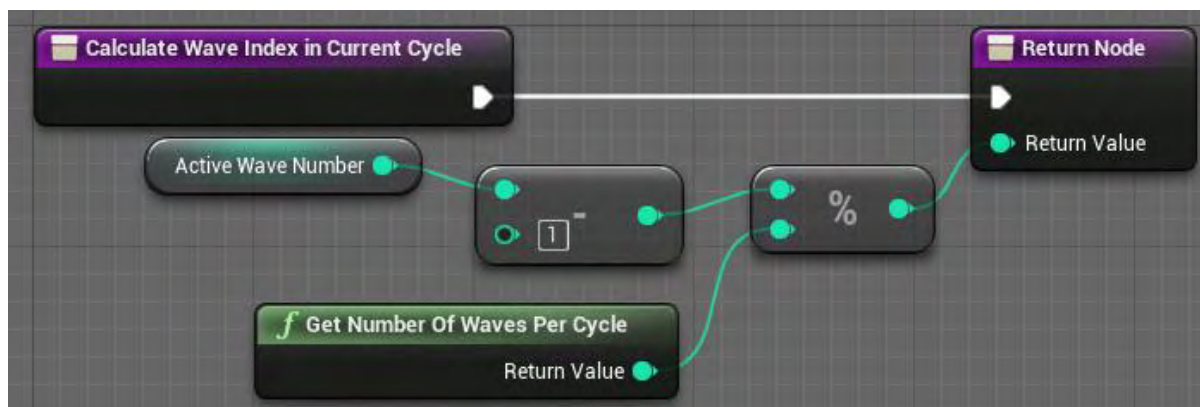


Η συνάρτηση DestroyAIUnit ελέγχει αν ο actor στην είσοδο υπάρχει. Αν ναι, τότε τον αφαιρεί από τον πίνακα SpawnedAIUnits, τον καταστρέφει και καλεί τη συνάρτηση AIUnitDestroyed (Εικόνα 162).



Εικόνα 162 BP\_ WaveSpawnController part 23

Η συνάρτηση CalculateWaveIndexInCurrentCycle χρησιμοποιώντας τη μεταβλητή ActiveWaveNumber και τη συνάρτηση GetNumberOfWavesPerCycle υπολογίζει και επιστρέφει τον δείκτη του τωρινού wave (Εικόνα 163).



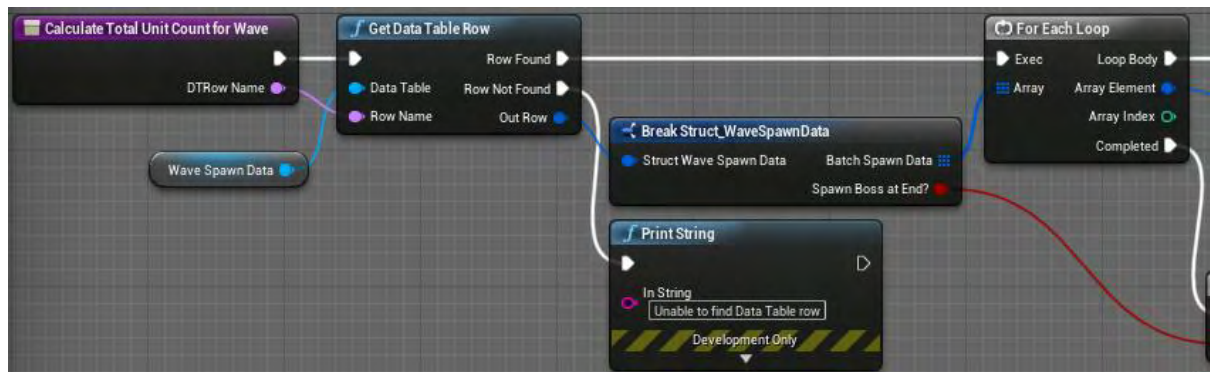
Εικόνα 163 BP\_ WaveSpawnController part 24

Η συνάρτηση GetSpawnedAIUnits επιστρέφει τον πίνακα SpawnedAIUnits, ο οποίος έχει όλα τα actors που έχουν δημιουργηθεί (Εικόνα 164).

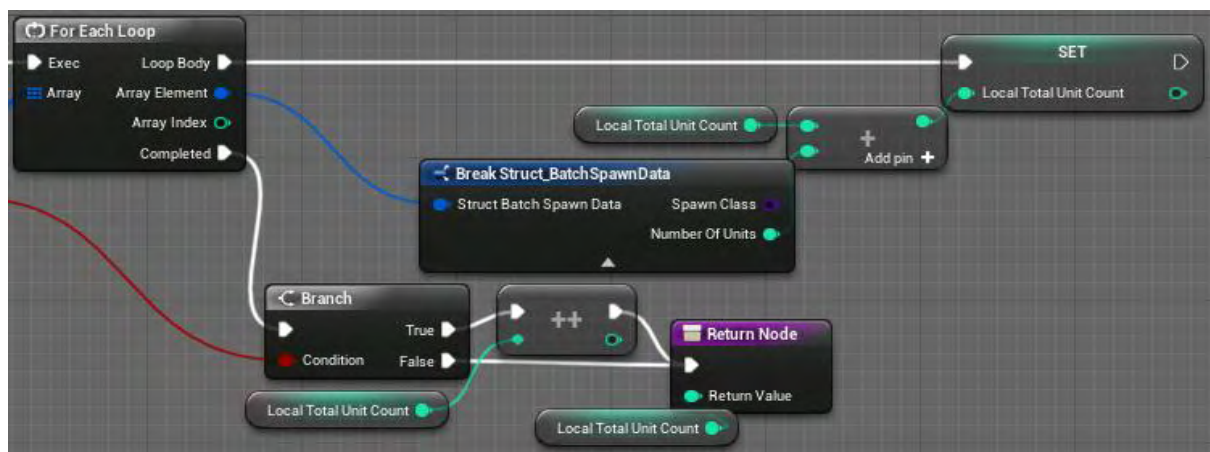


Εικόνα 164 BP\_ WaveSpawnController part 25

Η συνάρτηση CalculateTotalUnitCountForWave στην Εικόνα 165 και στην Εικόνα 166, υπολογίζει τον συνολικό αριθμό των actor που περιέχονται σε μια συγκεκριμένη γραμμή του πίνακα WaveSpawnData. Στη συνέχεια επιστρέφει την τιμή αυτή.



Εικόνα 165 BP\_ WaveSpawnController part 26



Εικόνα 166 BP\_ WaveSpawnController part 27

Η συνάρτηση GetDTRowNameForCurrentWave επιστρέφει το όνομα μιας συγκεκριμένης γραμμής του πίνακα WaveSpawnData (Εικόνα 167).

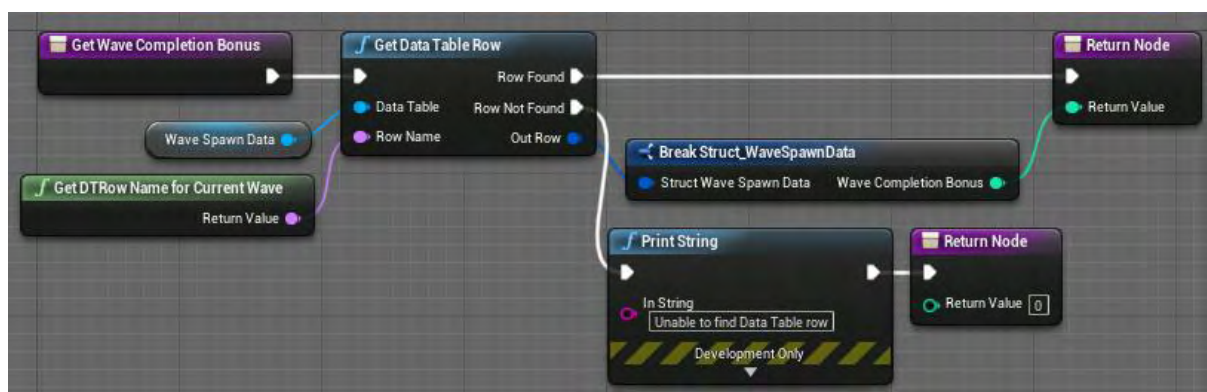


Εικόνα 167 BP\_ WaveSpawnController part28

Οι συναρτήσεις `GetStatsControlParameters` (Εικόνα 168), `GetWaveCompletionBonus` (Εικόνα 169) και `GetPointsMultiplierForActiveWaveCycle` (Εικόνα 170) δεν θα αναλυθούν, εφόσον δημιουργήθηκαν για μελλοντική χρήση.



Εικόνα 168 BP\_ WaveSpawnController part 29



Εικόνα 169 BP\_ WaveSpawnController part 30

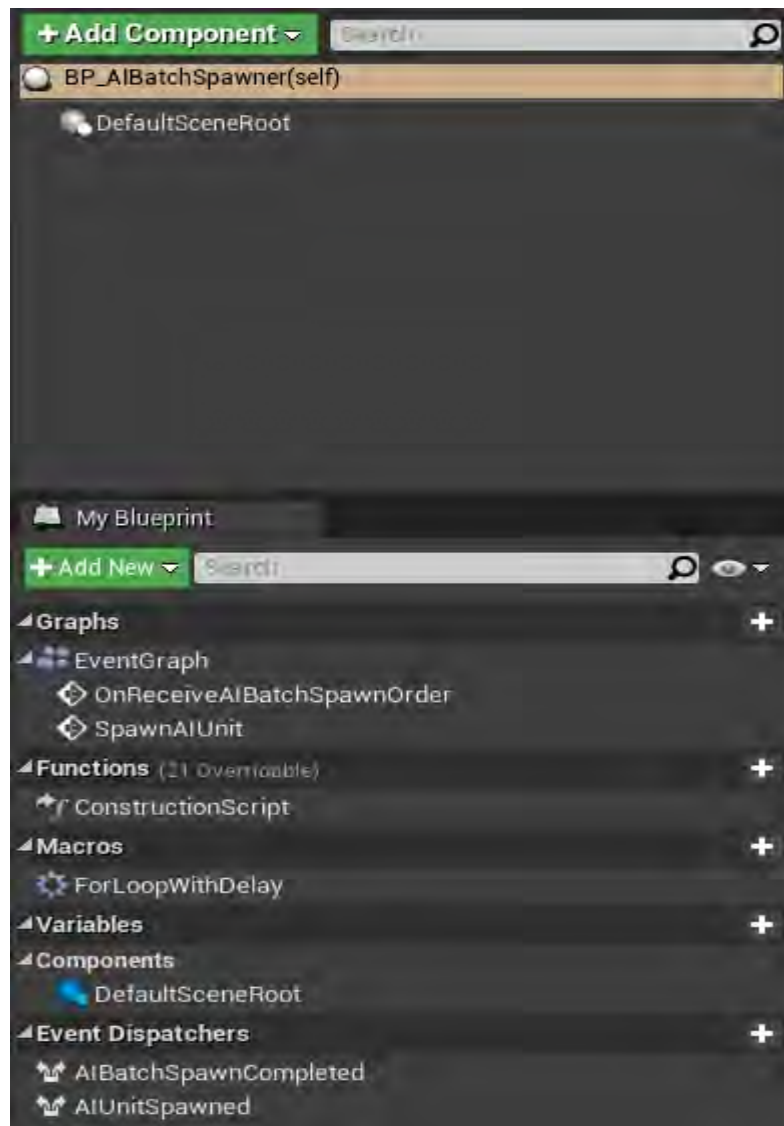


Εικόνα 170 BP\_ WaveSpawnController part 31

## Ενότητα 5.1.22 BP\_AIBatchSpawner

---

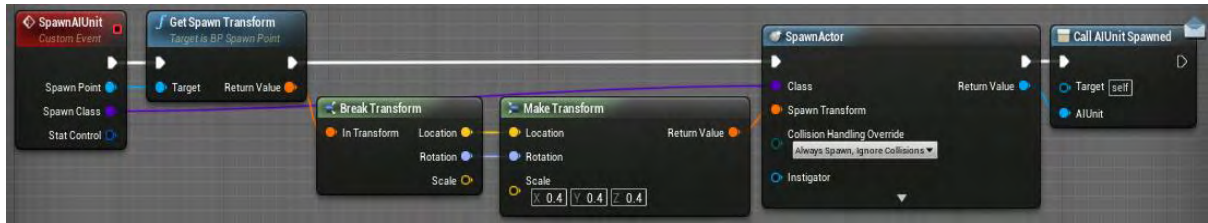
Το BP\_AIBatchSpawner (Εικόνα 171) υλοποιεί τις συναρτήσεις SpawnAIUnit και OnReceiveAIBatchSpawnOrder. Η δεύτερη δεν θα αναλυθεί περαιτέρω, εφόσον ο σκοπός της είναι να καλεί τη συνάρτηση SpawnAIUnit με τις σωστές τιμές στην είσοδο της.



Εικόνα 171 BP\_AIBatchSpawner part 1

Η συνάρτηση SpawnAIUnit δημιουργεί τον actor, που αντιστοιχεί στην τιμή της εισόδου SpawnClass με τον σωστό προσανατολισμό, στην σωστή τοποθεσία και κλίμακα. Στη συνέχεια καλεί τον Event Dispatcher AIUnitSpawned (Εικόνα 172).





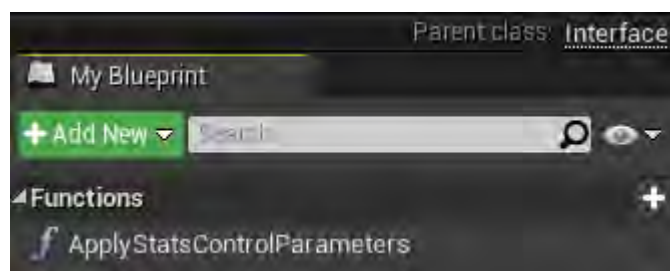
Εικόνα 172 BP\_AIBatchSpawner part2

## 5.2 Interface

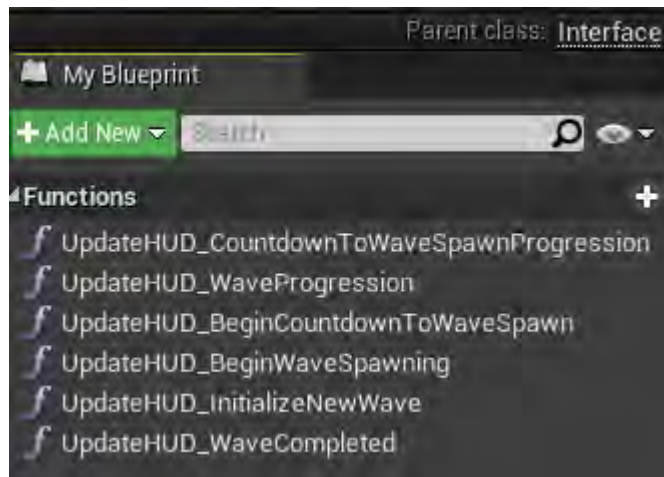
Στην Εικόνα 173 και στην Εικόνα 174 φαίνονται οι διεπαφές (interfaces), που διαθέτουν τις συναρτήσεις:

- ApplyStatsControlParameters
- UpdateHUD\_CountdownToWaveSpawnProgression
- UpdateHUD\_WaveProgression
- UpdateHUD\_BeginCountdownToWaveSpawn
- UpdateHUD\_BeginWaveSpawning
- UpdateHUD\_InitializeNewWave
- UpdateHUD\_WaveCompleted

Ο σκοπός των διεπαφών είναι να διευκολύνουν την πρόσβαση και την τροποποίηση αυτών από τους actors που τις ενσωματώνουν. Επίσης διευκολύνουν την επικοινωνία μεταξύ των actors που έχουν ενσωματωμένη την ίδια διεπαφή.



Εικόνα 173 BPI\_WaveCycleStatsController



Εικόνα 174 BPI\_WaveInfoDisplay

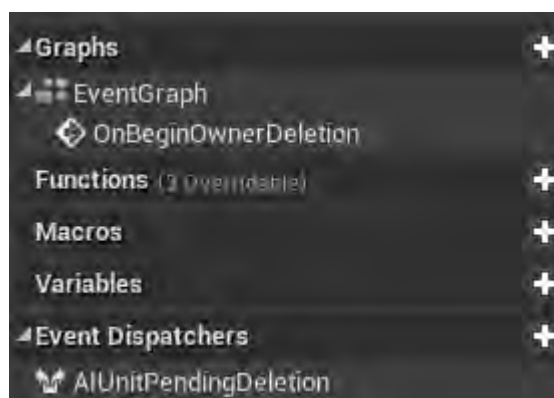
## 5.3 Actor Component

---

### Ενότητα 5.3.1 AC\_AIUnitDeletionController

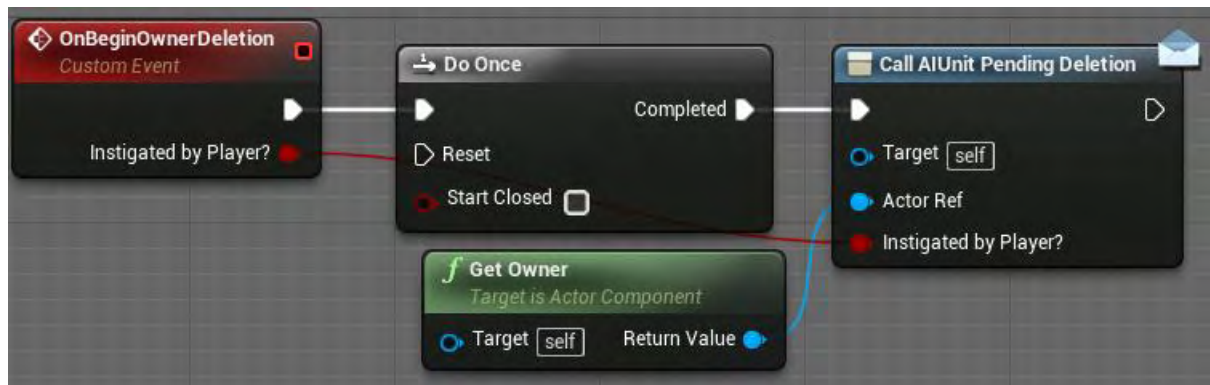
---

Το AC\_AIUnitDeletionController (Εικόνα 175) υλοποιεί τη συνάρτηση OnBeginOwnerDeletion (Εικόνα 176), η οποία καλεί τον Event Dispatcher AIUnitPendingDeletion με τις κατάλληλες τιμές για είσοδο.



Εικόνα 175 AC\_AIUnitDeletionController part 1





Εικόνα 176 AC\_AIUnitDeletionController part 2

## Ενότητα 5.3.2 AC\_DashDirectional

Το AC\_DashDirectional δίνει τη δυνατότητα στους actors τύπου FirstPersonCharacter, που το έχουν ως component, να χρησιμοποιήσουν τη δεξιότητα Dash προς όποια κατεύθυνση επιθυμούν.

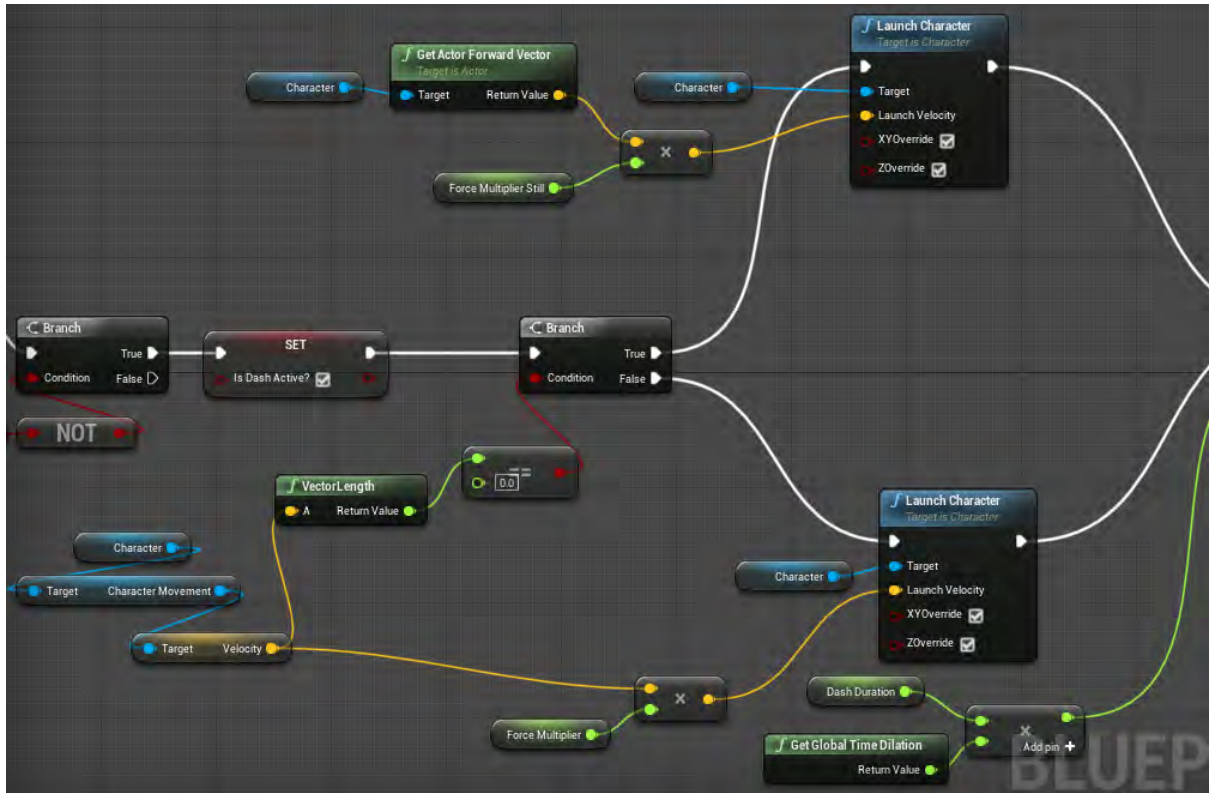
Η συνάρτηση ActivateAbillity μόλις ενεργοποιηθεί, αναθέτει την τιμή false στη μεταβλητή CanDash του χαρακτήρα του παίκτη. Στη συνέχεια ελέγχει την τιμή της μεταβλητής IsDashActive? και αν η τιμή της είναι false, τότε αναθέτει την τιμή true στη μεταβλητή αυτή (Εικόνα 177).



Εικόνα 177 AC\_DashDirectional part 1

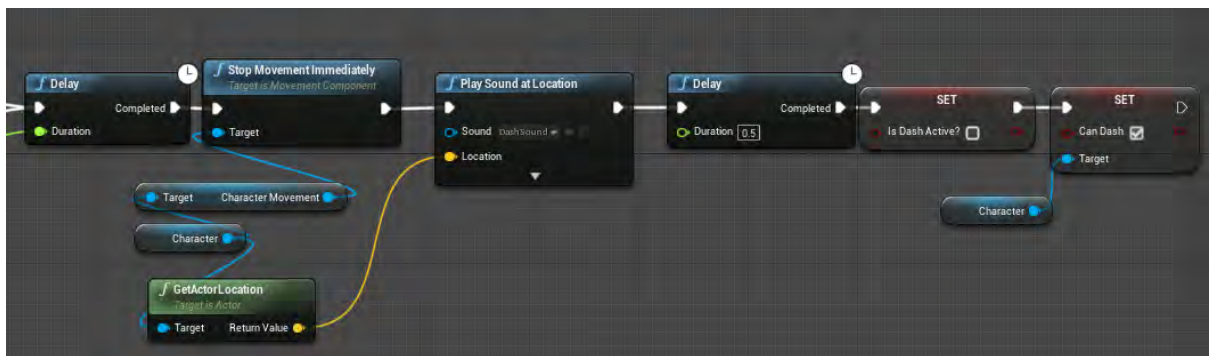
Όπως φαίνεται στην Εικόνα 178, στη συνέχεια της συνάρτησης ActivateAbillity ελέγχεται η ταχύτητα του παίκτη:

- Αν είναι μηδενική, ο χαρακτήρας του παίκτη εκτοξεύεται προς τα μπροστά σύμφωνα με τον δικό του προσανατολισμό.
- Αν δεν είναι μηδενική, ο χαρακτήρας του παίκτη εκτοξεύεται προς την κατεύθυνση που κινείται.



Εικόνα 178 AC\_DashDirectional part 2

Μετά από μια μικρή χρονοκαθυστέρηση σταματά στιγμιαία η κίνηση του χαρακτήρα του παίκτη και αναπαράγεται ένας ήχος. Οι τιμές των μεταβλητών IsDashActive? και CanDash επανέρχονται στις αρχικές τους (Εικόνα 179).

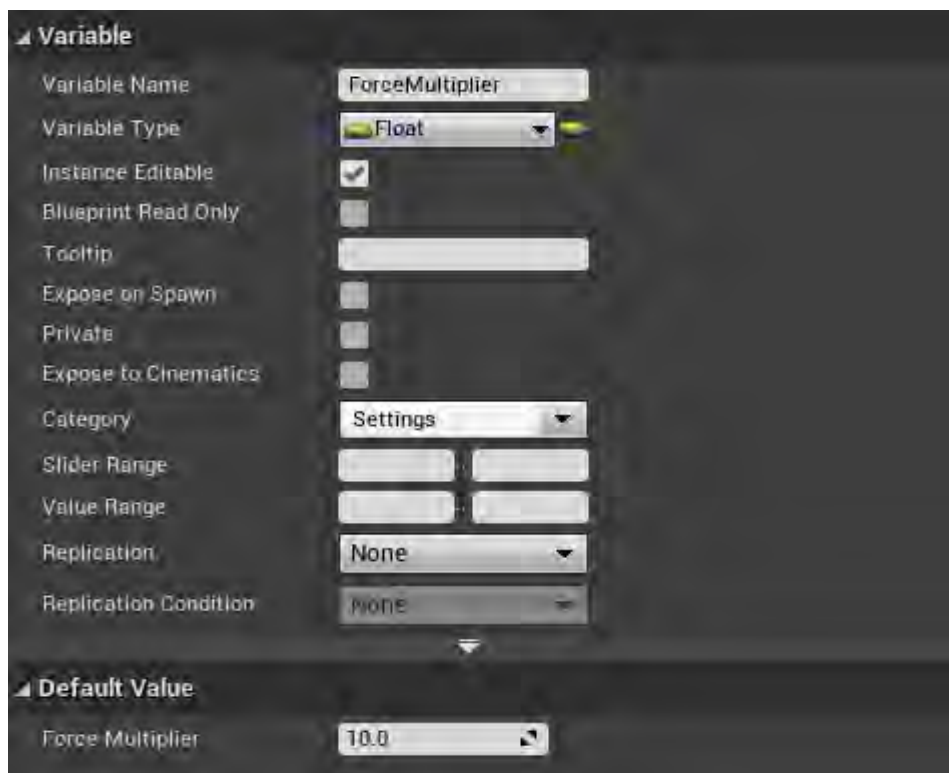


Εικόνα 179 AC\_DashDirectional part 3

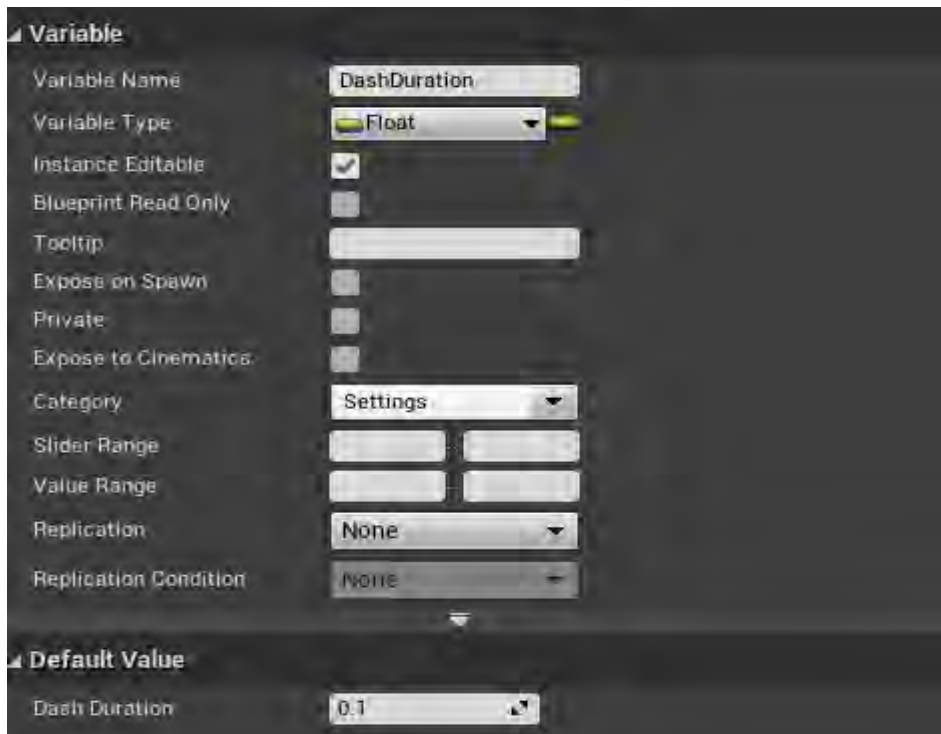
Στην εικόνα 180 απεικονίζονται οι μεταβλητές και η συνάρτηση του AC\_DashDirectional, ενώ στις επόμενες εικόνες (181 – 185) απεικονίζονται οι αρχικές τιμές και οι ρυθμίσεις των μεταβλητών αυτών.



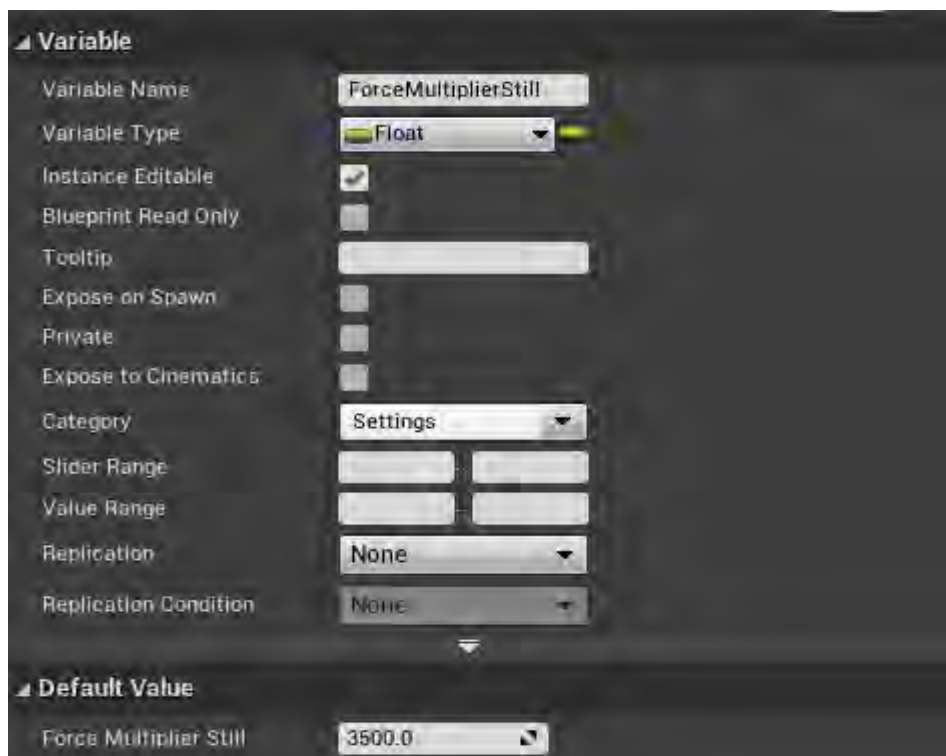
Εικόνα 180 AC\_DashDirectional part 4



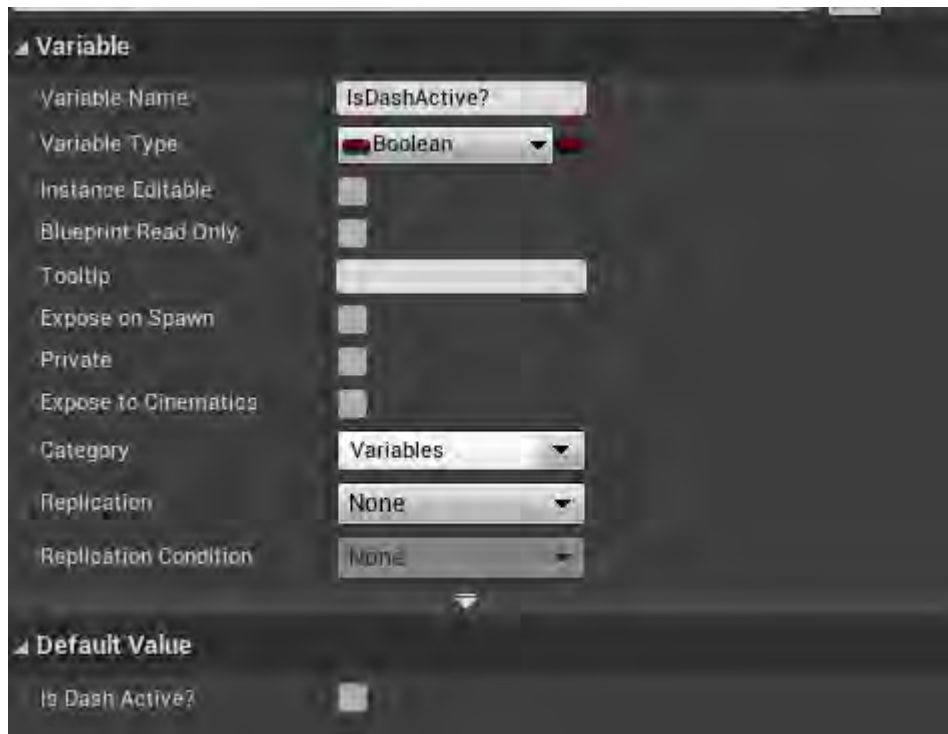
Εικόνα 181 AC\_DashDirectional part 5



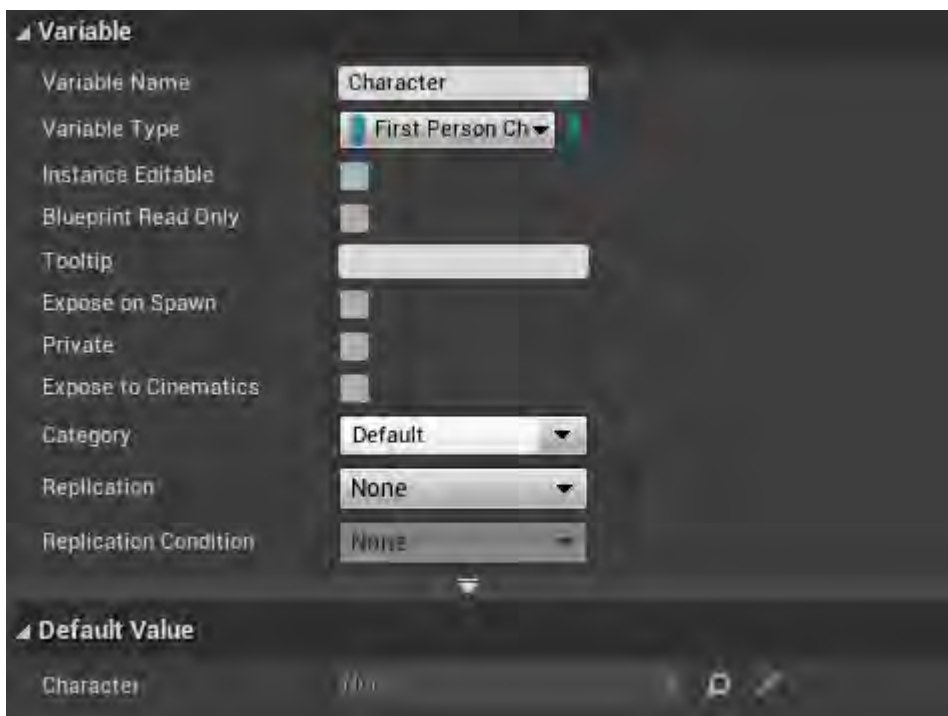
Εικόνα 182 AC\_DashDirectional part 6



Εικόνα 183 AC\_DashDirectional part 7



Εικόνα 184 AC\_DashDirectional part 8



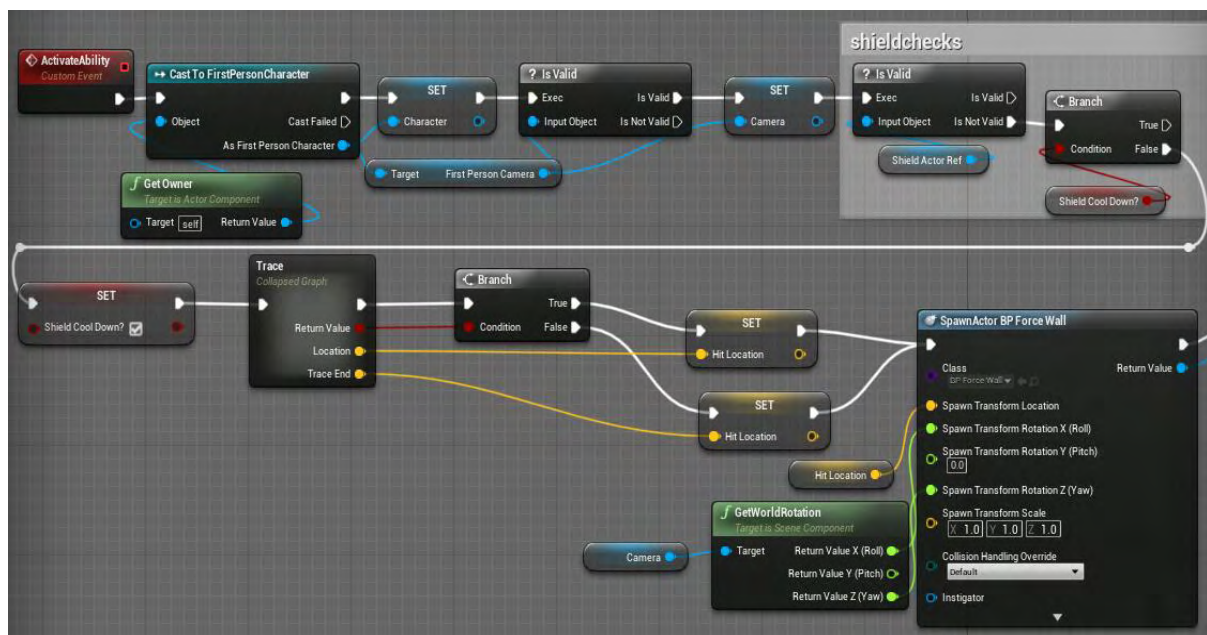
Εικόνα 185 AC\_DashDirectional part 9



### Ενότητα 5.3.3 AC\_ForceWall

Το AC\_ForceWall εμφανίζει το BP\_ForceWall σε συγκεκριμένη τοποθεσία με σκοπό την προστασία του χαρακτήρα του παίκτη από τα εχθρικά βλήματα.

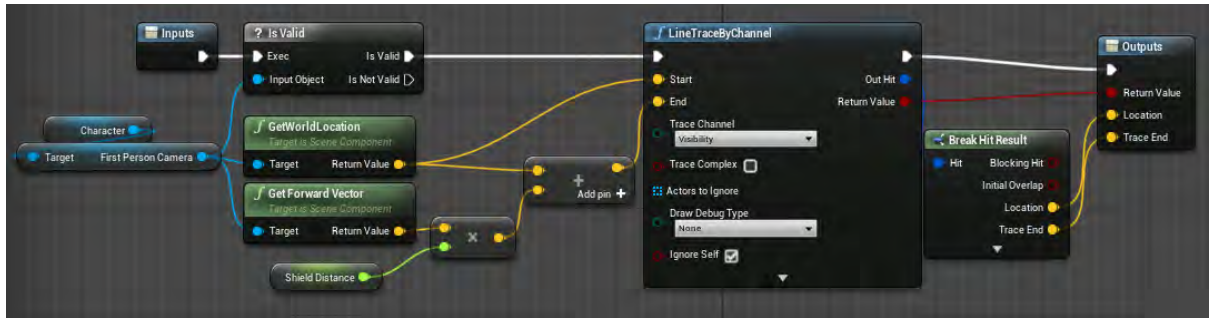
Η συνάρτηση ActivateAbility, έπειτα από τους απαραίτητους ελέγχους, εμφανίζει στην τοποθεσία που υπολογίζεται στο Collapsed Graph (Εικόνα 187) έναν actor τύπου BP\_ForceWall (Εικόνα 186).



Εικόνα 186 AC\_ForceWall part 1

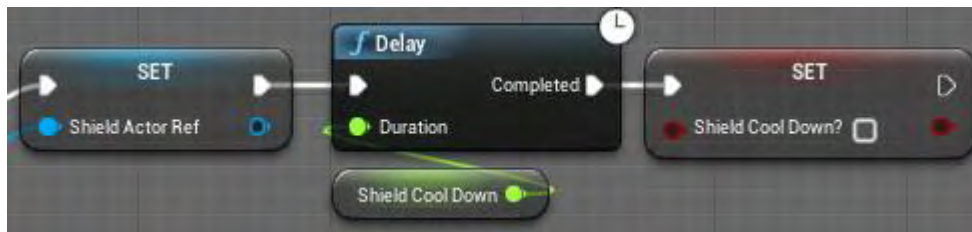
Το Collapsed Graph (Trace) χρησιμοποιώντας τον κόμβο LineTraceByChannel επιστρέφει 2 τιμές τύπου Vector, που αντιστοιχούν σε τοποθεσίες. Στη μεταβλητή Location ανατίθεται το σημείο όπου προέκυψε σύγκρουση του διανύσματος, που δημιουργείται από τον κόμβο, με κάποιο άλλο πλέγμα σύγκρουσης. Στη μεταβλητή TraceEnd ανατίθεται το τελευταίο σημείο του διανύσματος (Εικόνα 187).





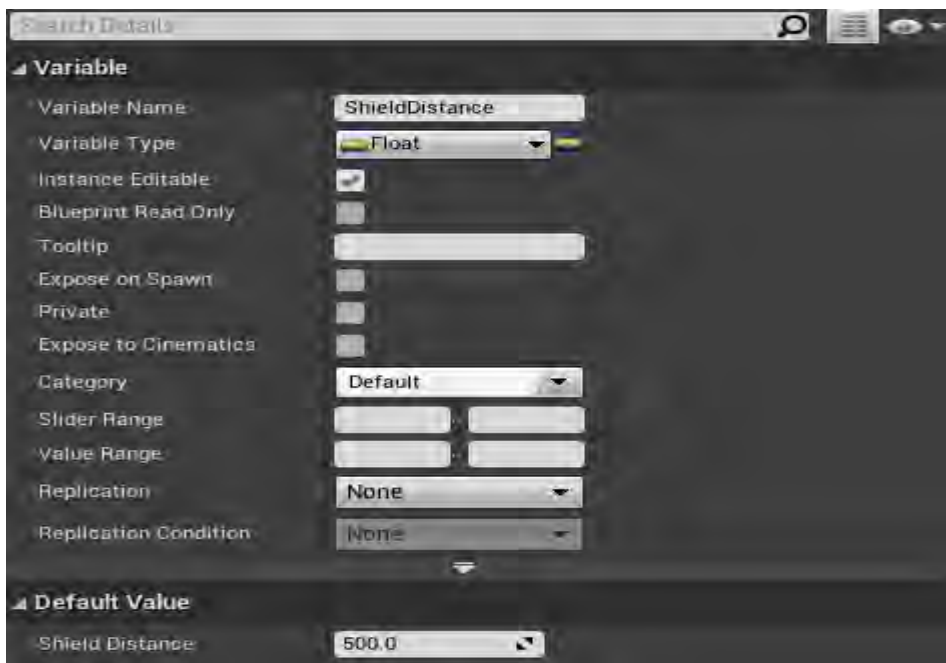
Εικόνα 187 AC\_ForceWall part 2

Στην Εικόνα 188 φαίνεται ότι μετά από μια χρονοκαθυστέρηση προκαθορισμένης διάρκειας ανατίθεται στην τιμή ShieldCoolDown? η αρχική της τιμή.

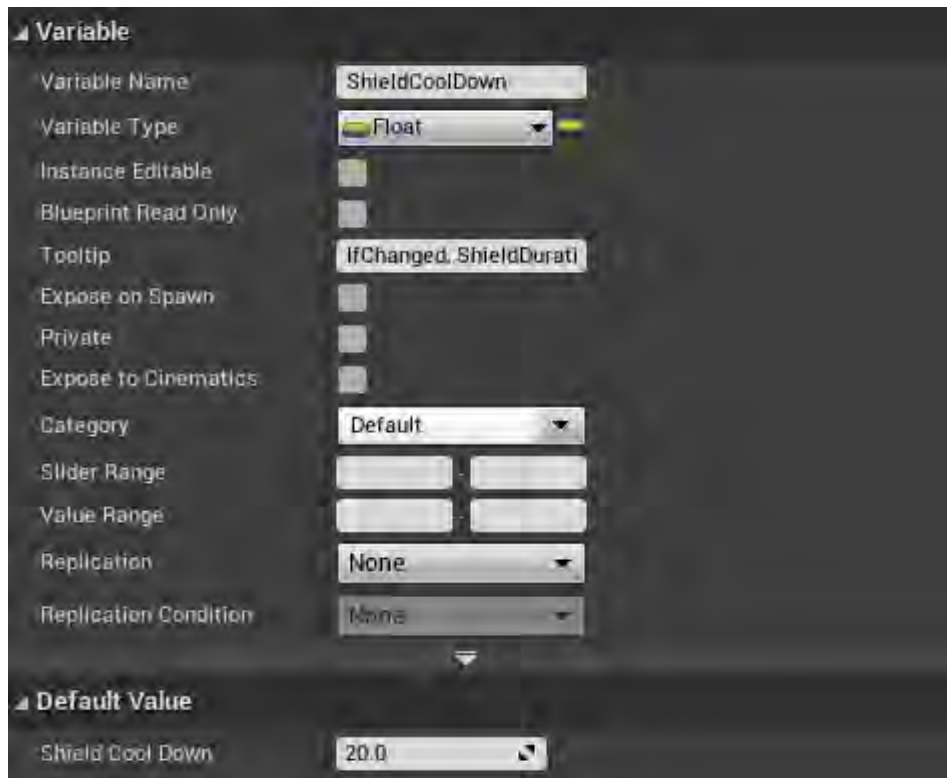


Εικόνα 188 AC\_ForceWall part 3

Στην Εικόνα 189 και στην Εικόνα 190 απεικονίζονται οι αρχικές τιμές των μεταβλητών και οι ρυθμίσεις τους.



Εικόνα 189 AC\_ForceWall part 4

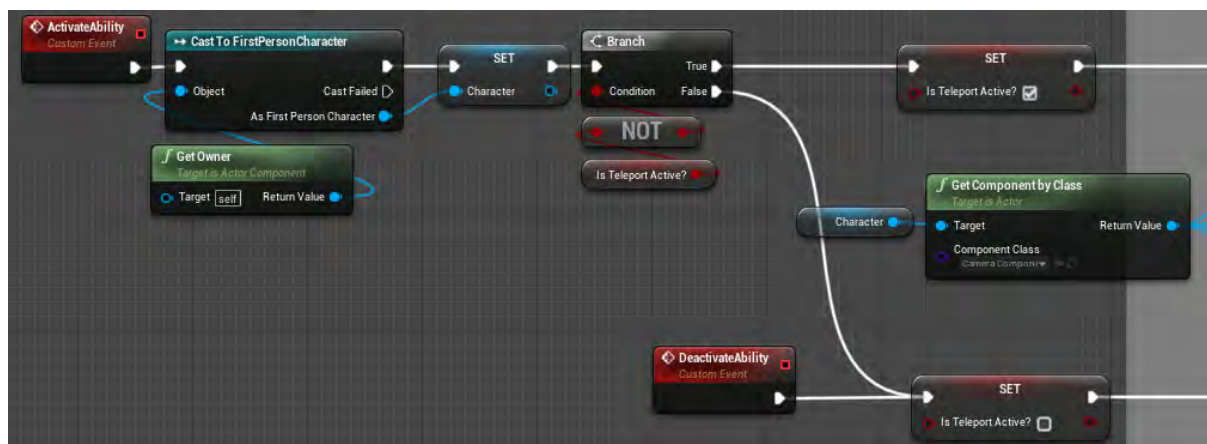


Εικόνα 190 AC\_ForceWall part 5

### Ενότητα 5.3.4 AC\_TeleportFinal

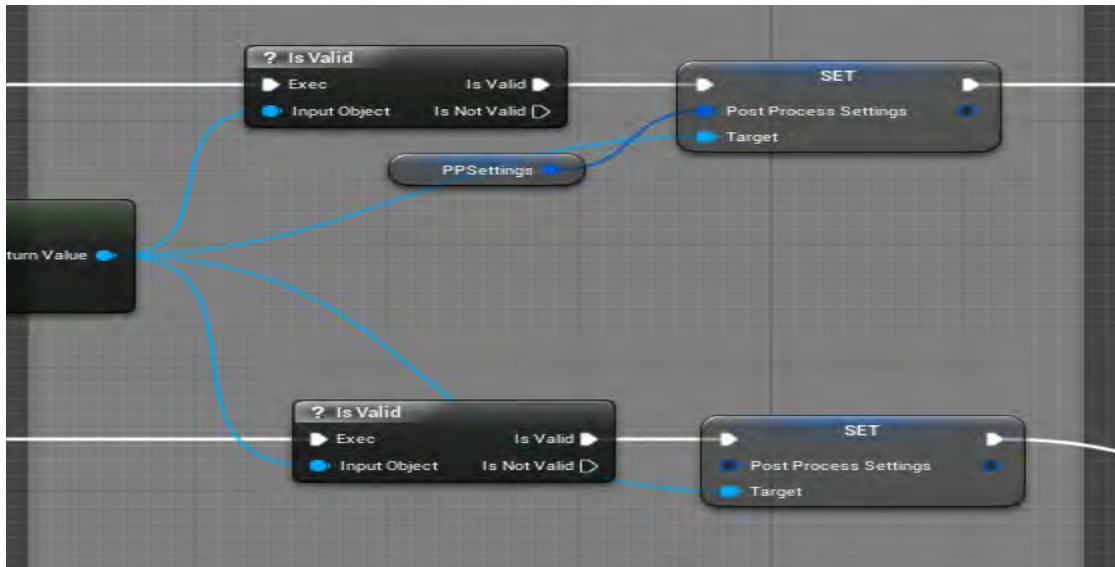
Το AC\_TeleportFinal δίνει την δυνατότητα στους actors τύπου FirstPersonCharacter, που το έχουν ως component, να χρησιμοποιήσουν την δεξιότητα τηλεμεταφοράς.

Η συνάρτηση ActivateAbillity ελέγχει αν είναι ενεργή η δεξιότητα αυτή και αναλόγως τροποποιεί την τιμή της μεταβλητής IsTeleportActive? (Εικόνα 191).



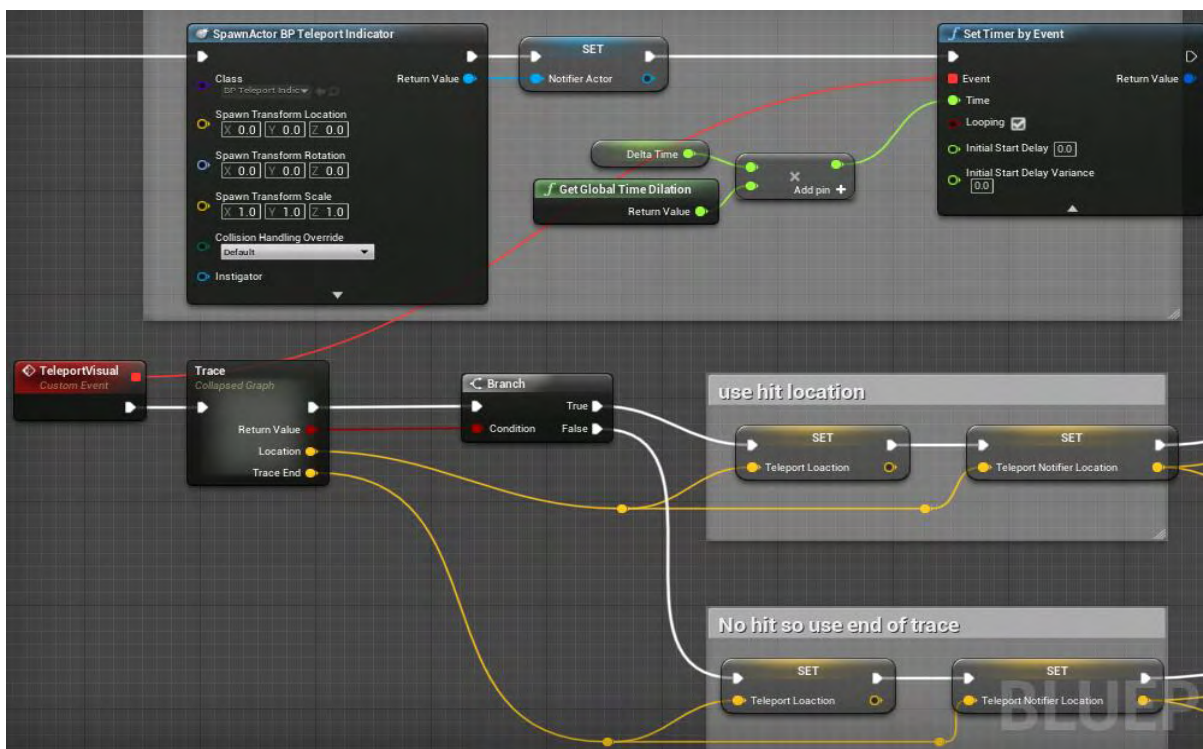
Εικόνα 191 AC\_TeleportFinal part 1

Στη συνέχεια τροποποιεί τις ρυθμίσεις της κάμερας του χαρακτήρα του παίκτη, ανάλογα με την τιμή της μεταβλητής IsTeleportActive? (Εικόνα 192).



Εικόνα 192 AC\_TeleportFinal part 2

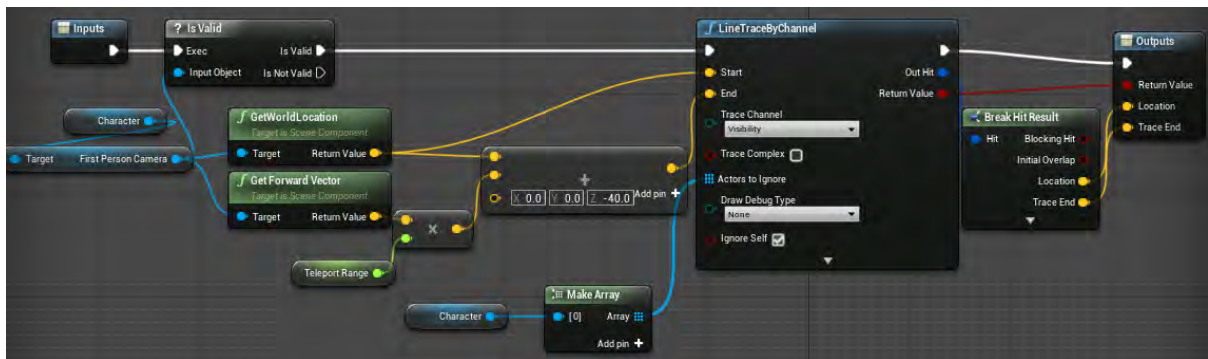
Στην περίπτωση που η τηλεμεταφορά δεν είναι ήδη ενεργή, δημιουργεί έναν actor τύπου BP\_TeleportIndicator. Κατόπιν θέτει μετρητή στη συνάρτηση TeleportVisual, εκτελεί το Collapsed Graph Trace (Εικόνα 194) και ανάλογα με τα αποτελέσματα του Trace, αναθέτει στις μεταβλητές TeleportLocation και TeleportNotifierLocation τις αντίστοιχες τιμές (Εικόνα 193).



Εικόνα 193 AC\_TeleportFinal part 3

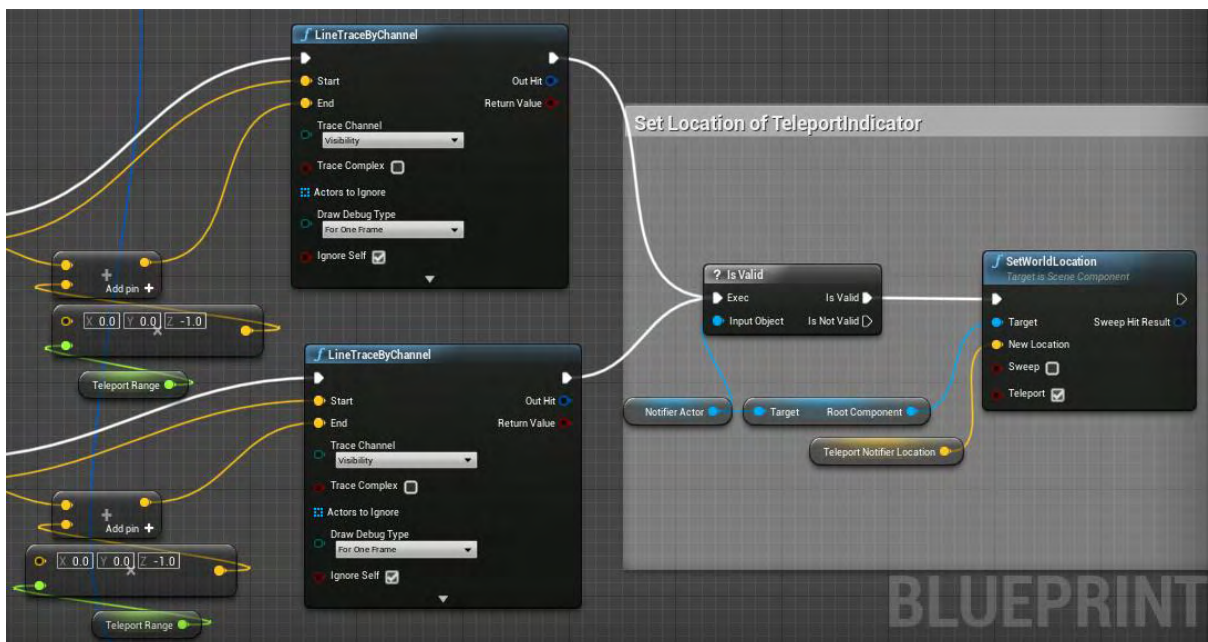


Το Collapsed Graph Trace χρησιμοποιώντας τον κόμβο LineTraceByChannel επιστρέφει 2 τιμές τύπου Vector, που αντιστοιχούν σε τοποθεσίες. Στη μεταβλητή Location ανατίθεται το σημείο όπου προέκυψε σύγκρουση του διανύσματος, που δημιουργείται από τον κόμβο, με κάποιο άλλο πλέγμα σύγκρουσης. Στη μεταβλητή TraceEnd ανατίθεται το τελευταίο σημείο του διανύσματος (Εικόνα 194).



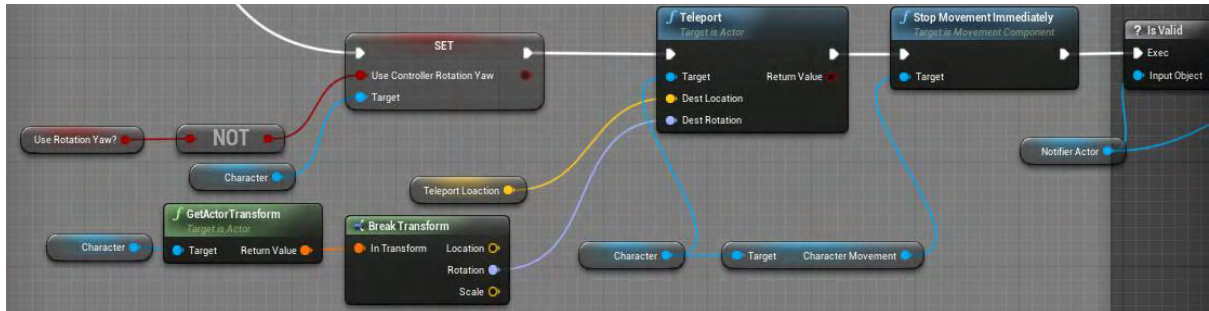
Εικόνα 194 AC\_TeleportFinal part 4

Στη συνέχεια η συνάρτηση TeleportVisual ενημερώνει την τοποθεσία του actor τύπου BP\_TeleportIndicator (Εικόνα 195).



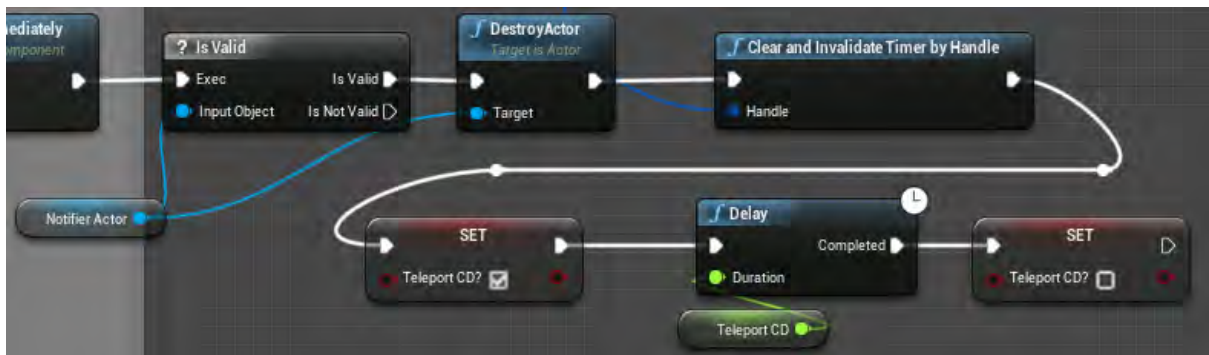
Εικόνα 195 AC\_TeleportFinal part 5

Στην περίπτωση που η τηλεμεταφορά είναι ήδη ενεργή, τηλεμεταφέρει τον χαρακτήρα του παίκτη στην τοποθεσία που αντιπροσωπεύει η τιμή της μεταβλητής TeleportLocation και παύει στιγμιαία την κίνηση του χαρακτήρα (Εικόνα 196).



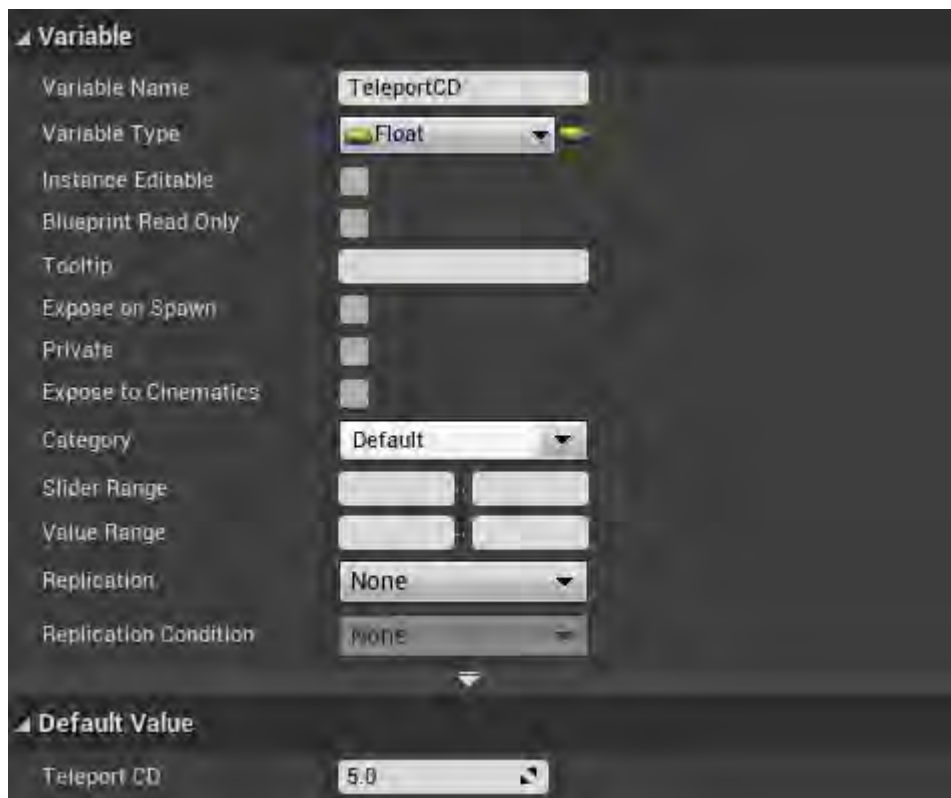
Εικόνα 196 AC\_TeleportFinal part 6

Κατόπιν καταστρέφει τον actor, τύπου BP\_TeleportIndicator, και σταματά τον μετρητή της συνάρτησης TeleportVisual. Ύστερα από μια προκαθορισμένη χρονοκαθυστέρηση ενημερώνει την τιμή της μεταβλητής TeleportCD?, ώστε να μπορεί να χρησιμοποιηθεί και πάλι η τηλεμεταφορά (Εικόνα 197).



Εικόνα 197 AC\_TeleportFinal part 7

Στην Εικόνα 198 απεικονίζεται η αρχική τιμή της μεταβλητής TeleportCD.



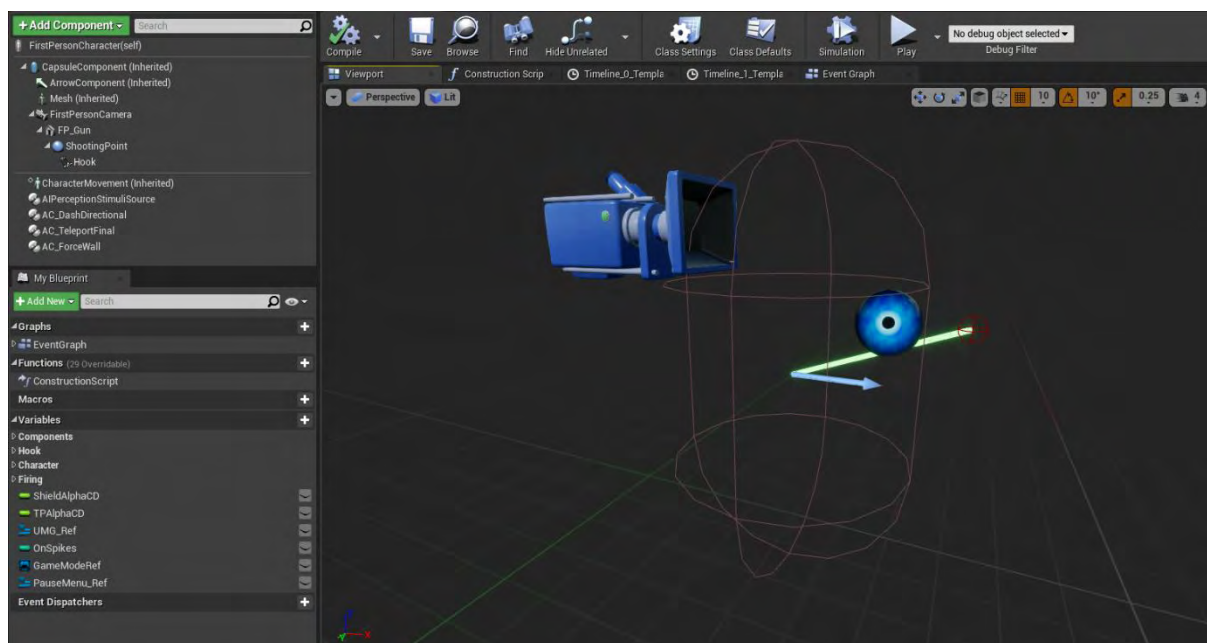
Εικόνα 198 AC\_TeleportFinal part 8



## 5.4 First Person Character

### Ενότητα 5.4.1 FirstPersonCharacter

Το FirstPersonCharacter είναι ο χαρακτήρας του παίκτη του βιντεοπαιχνιδιού του The Challenger. Βασίζεται στο πρότυπο πρώτου προσώπου της Unreal Engine 4. Έχει εμπλουτιστεί με actor components, συναρτήσεις και μεταβλητές, ώστε να έχει περισσότερες δυνατότητες (Εικόνα 199).



Εικόνα 199 FirstPersonCharacter part 1

Στην Εικόνα 200 απεικονίζονται οι ρυθμίσεις σύγκρουσης του CapsuleComponent, το οποίο κληρονομείται από την κλάση Character.



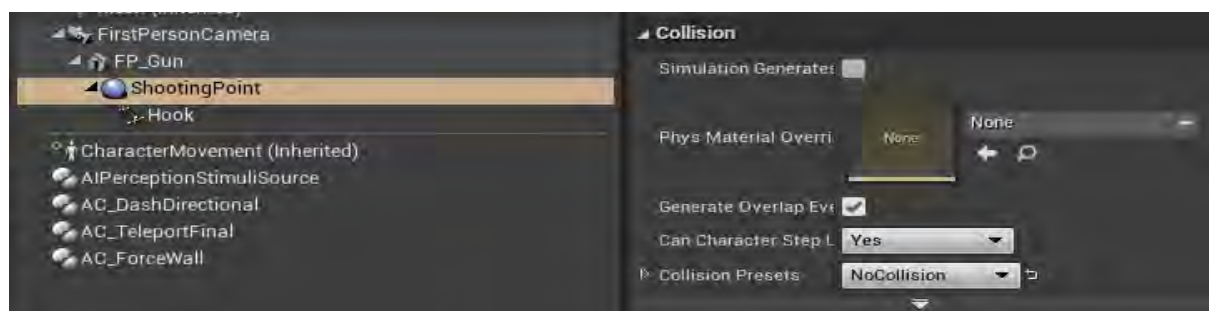
Εικόνα 200 FirstPersonCharacter part 2

Στην Εικόνα 201 φαίνονται οι ρυθμίσεις του FP\_Gun.



Εικόνα 201 FirstPersonCharacter part 3

Οι ρυθμίσεις σύγκρουσης του ShootingPoint απεικονίζονται στην Εικόνα 202.



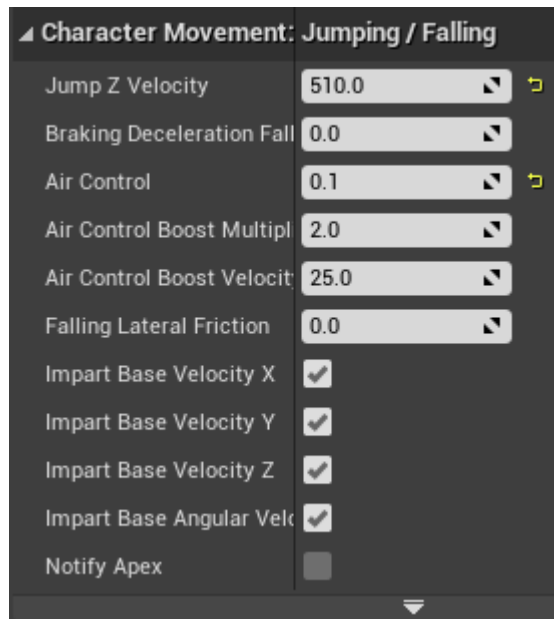
Εικόνα 202 FirstPersonCharacter part 4

Στην Εικόνα 203 φαίνονται οι ρυθμίσεις του Cable Component (Hook).



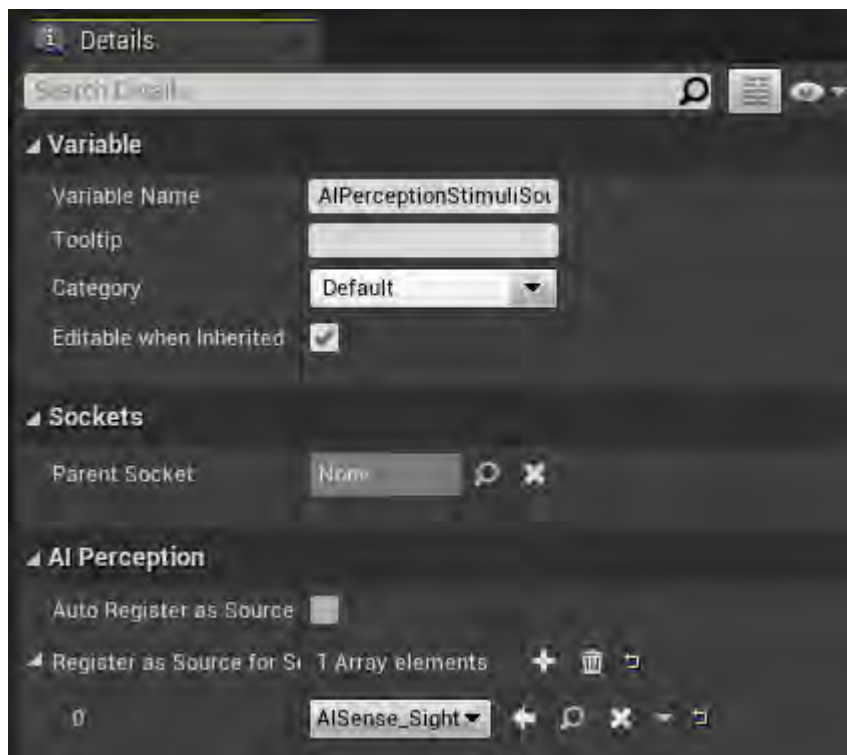
Εικόνα 203 FirstPersonCharacter part 5

Οι ρυθμίσεις του CharacterMovement, που έχουν τροποποιηθεί, απεικονίζονται στην Εικόνα 204.



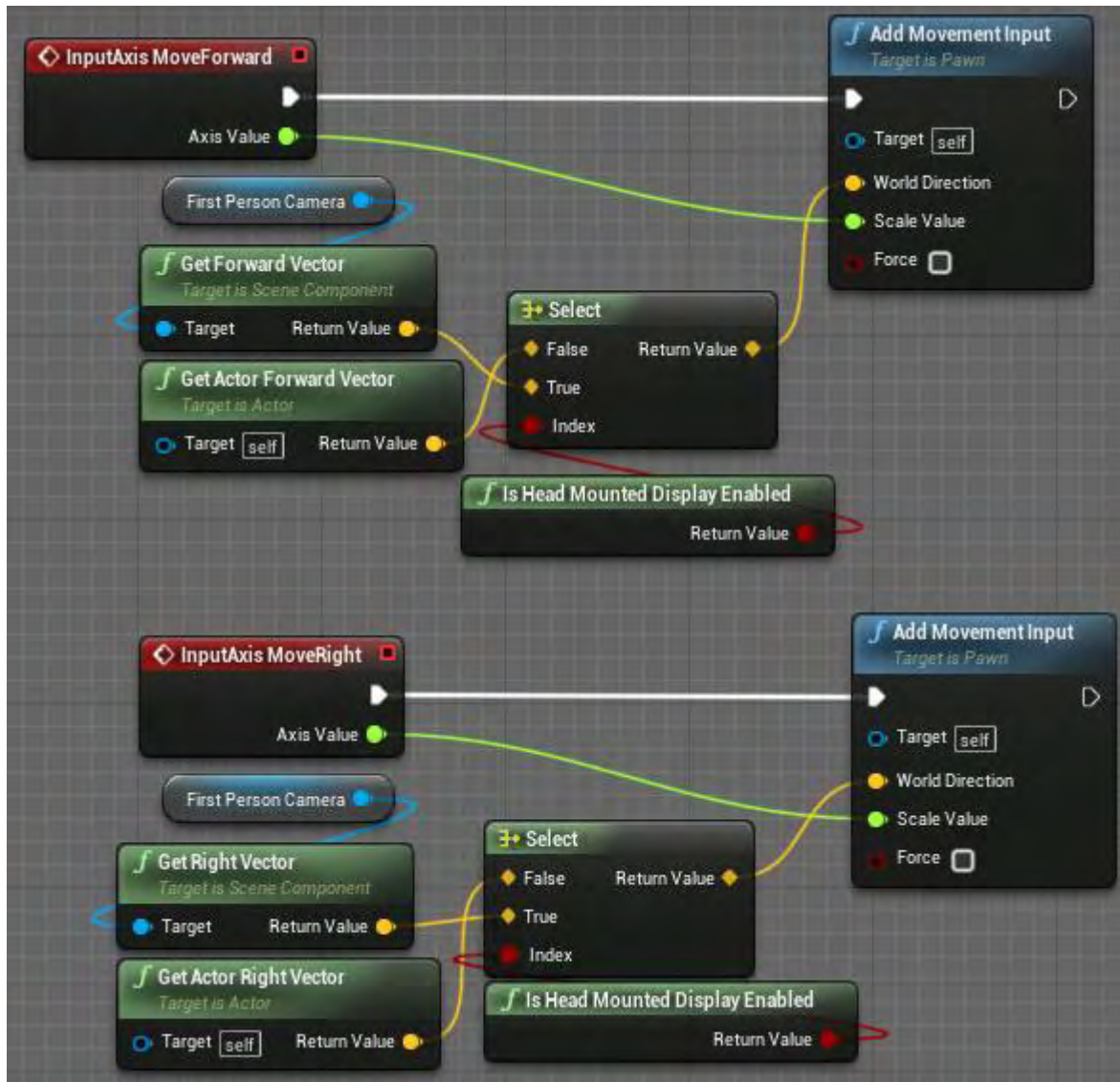
Εικόνα 204 FirstPersonCharacter part 6

Στην Εικόνα 205 φαίνονται οι ρυθμίσεις του AI Perception Stimuli Source.



Εικόνα 205 FirstPersonCharacter part 7

Η συνάρτηση InputAxis MoveForward μετακινεί τον χαρακτήρα του παίκτη προς τα εμπρός και προς τα πίσω. Η συνάρτηση InputAxis MoveRight μετακινεί τον χαρακτήρα του παίκτη προς τα δεξιά και προς τα αριστερά (Εικόνα 206).



Εικόνα 206 FirstPersonCharacter part 8

Η συνάρτηση InputAxis Turn κινεί την κάμερα του χαρακτήρα του παίκτη προς τα δεξιά και προς τα αριστερά, ενώ η InputAxis LookUp προς τα δεξιά και προς τα αριστερά. Η InputAction Jump καλεί τις συναρτήσεις Jump ή StopJumping (Εικόνα 207).





Εικόνα 207 FirstPersonCharacter part 9

Μόλις δημιουργηθεί ο actor, ενεργοποιείται το Event BeginPlay. Κατόπιν αρχικοποιούνται οι μεταβλητές GameModeRef και DefaultWeaponRot. Στη συνέχεια για όλα τα Widgets του τύπου UMG\_HUD (Εικόνα 208) καλούνται οι συναρτήσεις τους UpdateHealth, UpdateShieldCDVisual και UpdateTeleportCooldownVisual. Αρχικοποιείται η μεταβλητή IsPlayerDead (Εικόνα 209).



Εικόνα 208 FirstPersonCharacter part 10



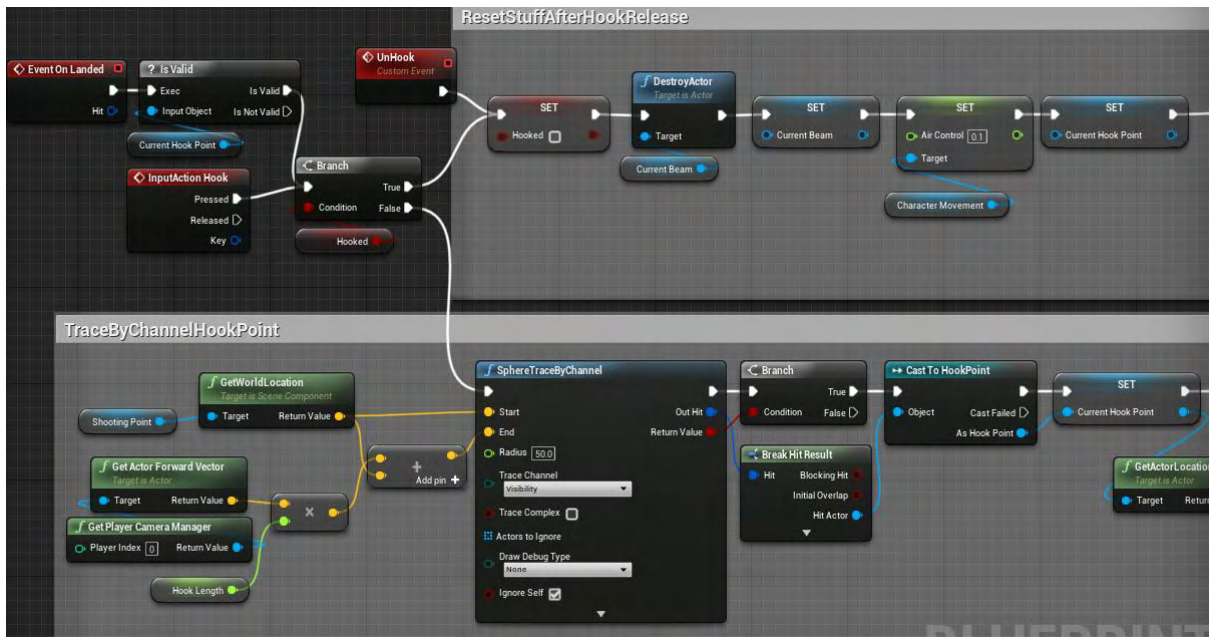
Εικόνα 209 FirstPersonCharacter part 11

Όταν ο παίκτης πατήσει το πλήκτρο E ή R, ενεργοποιείται το InputAction Hook και ελέγχεται η τιμή της μεταβλητής Hooked. Ο έλεγχος αυτός πραγματοποιείται και

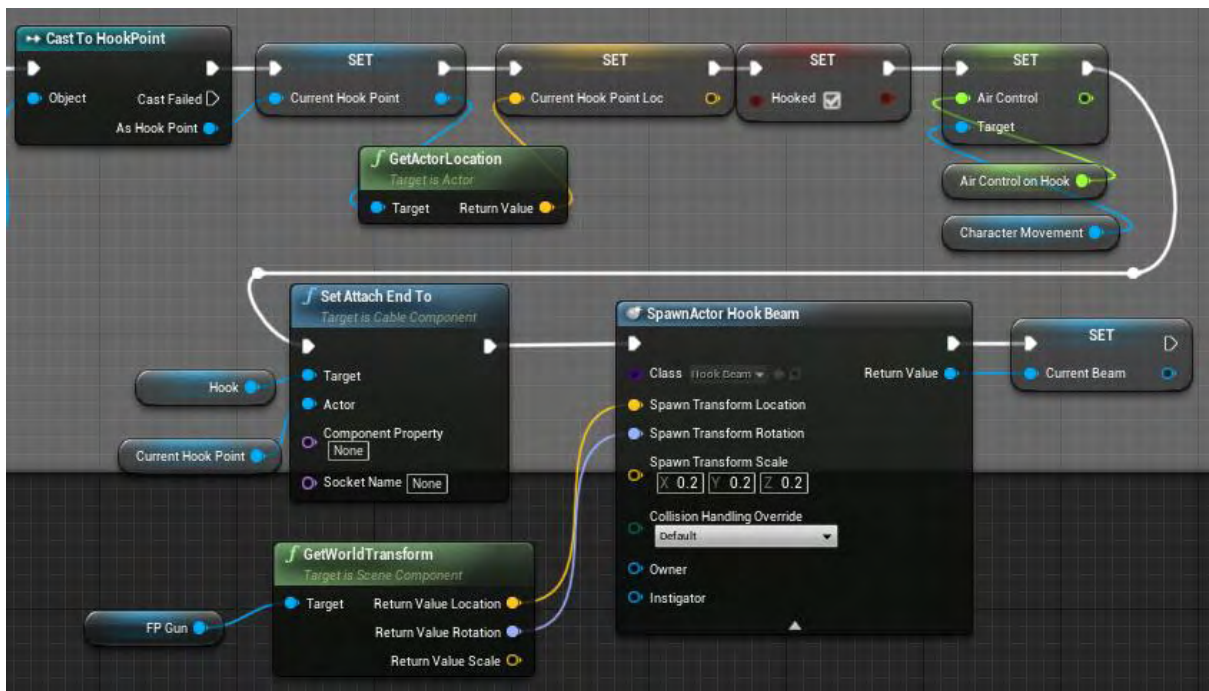


όταν ενεργοποιείται το Event On Landed με την προϋπόθεση να είναι έγκυρη η τιμή της μεταβλητής CurrentHookPoint.

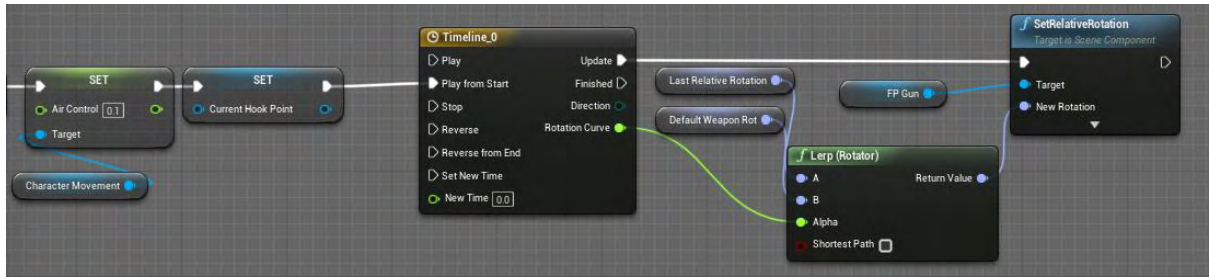
- Όταν η τιμή του Hooked είναι ψευδής, ο παίκτης δεν έχει ενεργοποιημένη την δεξιότητα Hook, η οποία πρέπει να ενεργοποιηθεί αν πληρούνται οι προϋποθέσεις. Για να γίνει αυτό, πραγματοποιείται ο κόμβος SphereTraceByChannel από την τοποθεσία του ShootingPoint προς την κατεύθυνση που κοιτά η κάμερα. Ο κόμβος δημιουργεί μια σφαίρα μήκους 50 μονάδων της Unreal, η οποία επεκτείνεται κατά την τιμή της μεταβλητής HookLength. Αν υπάρξει σύγκρουση και το αντικείμενο της σύγκρουσης είναι τύπου actor BP\_HookPoint, ανατίθεται στη μεταβλητή CurrentHookPoint ο συγκεκριμένος actor (Εικόνα 210). Κατόπιν αποθηκεύεται η τοποθεσία του συγκεκριμένου actor στη μεταβλητή CurrentHookPointLoc. Μετέπειτα ενημερώνεται η μεταβλητή Hooked και τροποποιείται η τιμή του Air Control του CharacterMovement. Στη συνέχεια ορίζεται το τελικό σημείο του Hook, δημιουργείται actor τύπου BP\_HookBeam και ενημερώνεται το CurrentBeam (Εικόνα 211).
- Όταν η τιμή του Hooked είναι αληθής, ο χαρακτήρας του παίκτη χρησιμοποιεί εκείνη τη στιγμή τη δεξιότητα Hook, οπότε πρέπει να πάψει η χρήση της. Για το λόγο αυτό ενημερώνεται η τιμή του Hooked, καταστρέφεται το CurrentBeam και επανέρχεται στην επιλεγμένη τιμή η δυνατότητα του χαρακτήρα του παίκτη να επηρεάζει την κίνησή του στον αέρα. Μετά αρχικοποιείται η τιμή του CurrentHookPoint σε None (Εικόνα 210). Στη συνέχεια ενεργοποιείται το Timeline\_0 και επανέρχεται σύμφωνα με αυτό ο αρχικός σχετικός προσανατολισμός (Εικόνα 212). Όλη η παραπάνω διαδικασία πραγματοποιείται και με τη συνάρτηση UnHook.



Εικόνα 210 FirstPersonCharacter part 12

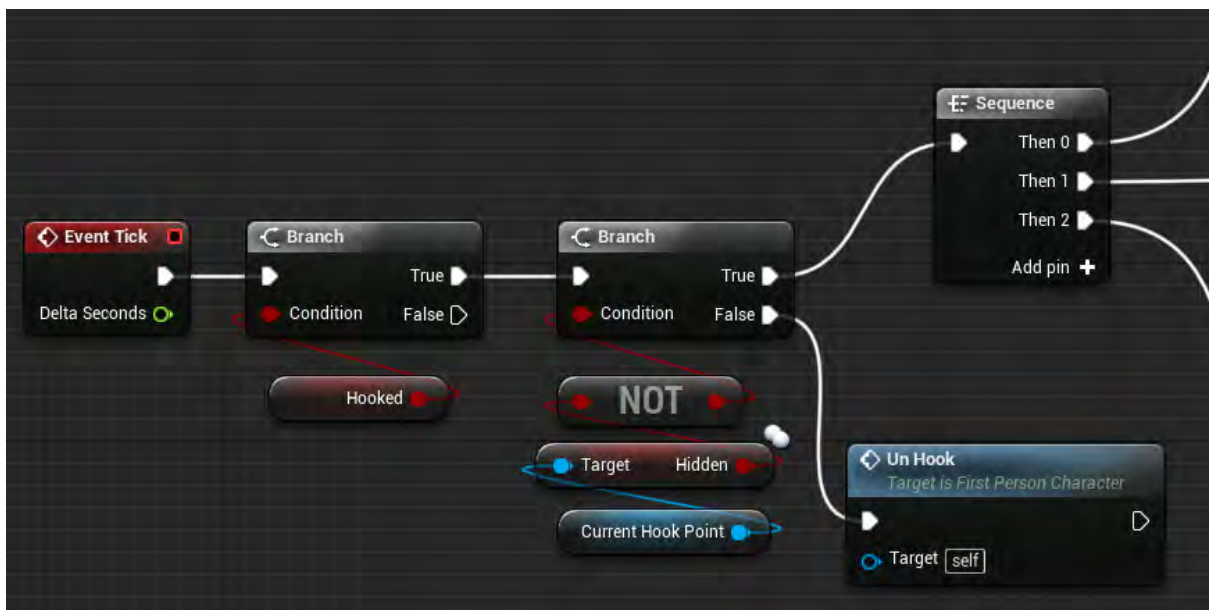


Εικόνα 211 FirstPersonCharacter part 13



Εικόνα 212 FirstPersonCharacter part 14

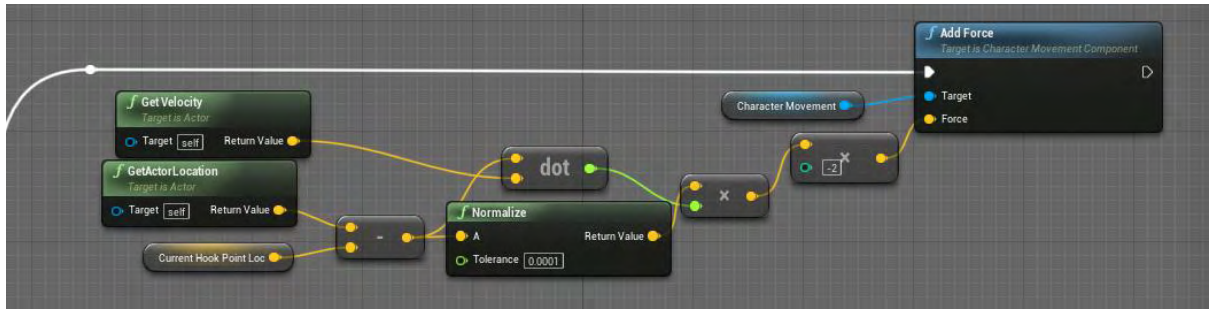
Το γεγονός Event Tick συμβαίνει σε κάθε frame. Ελέγχεται αν ο χαρακτήρας του παίκτη κάνει χρήση της δεξιότητας Hook και αν μπορεί να συνεχίσει να τη χρησιμοποιεί. Είναι εφικτό να τη χρησιμοποιεί όσο το CurrentHookPoint είναι εμφανές. Όσο συμβαίνει αυτό, τρέχουν σειριακά τρεις διαφορετικές ροές κώδικα (Εικόνα 213).



Εικόνα 213 FirstPersonCharacter part 15

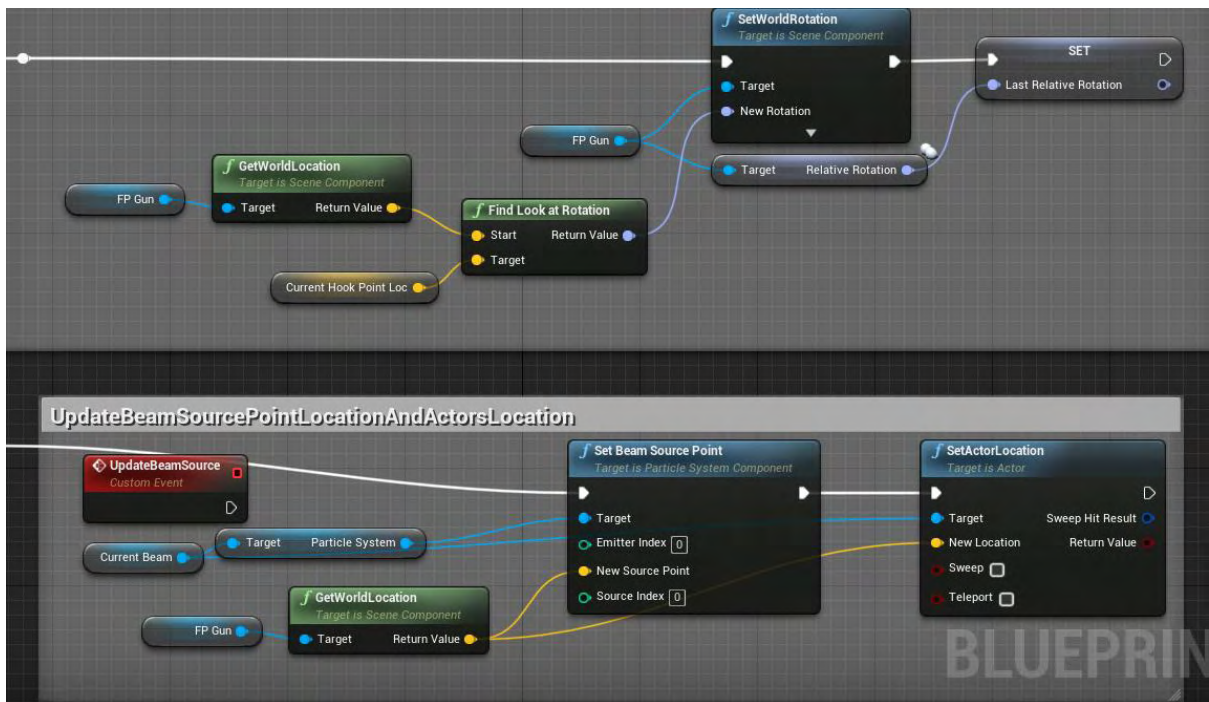
Στην πρώτη ροή κώδικα προστίθεται δύναμη στον χαρακτήρα του παίκτη, η οποία υπολογίζεται με συγκεκριμένες μαθηματικές πράξεις ( Εικόνα 214).





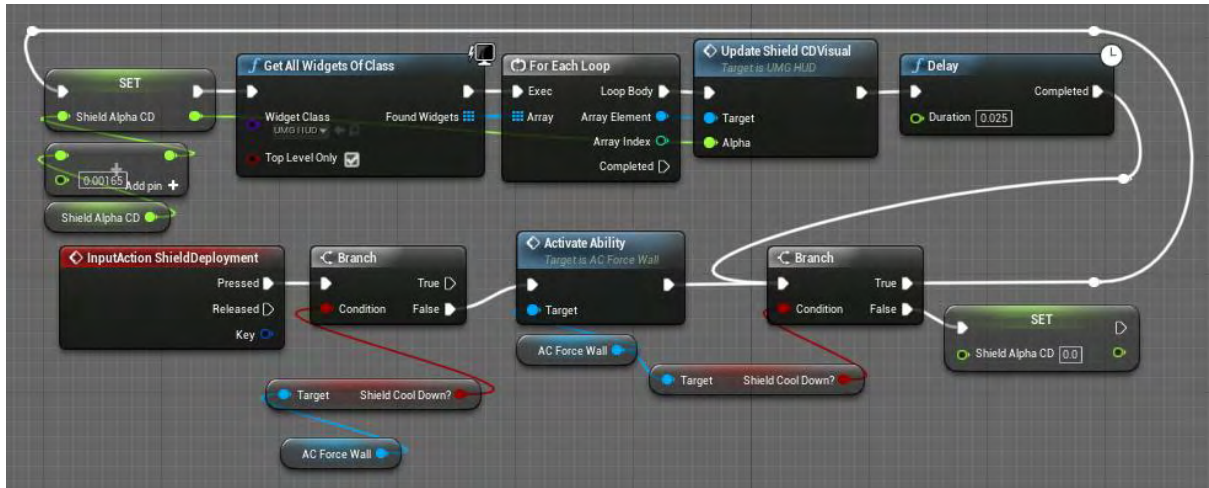
Εικόνα 214 FirstPersonCharacter part 16

Στη δεύτερη ροή κώδικα ενημερώνεται ο προσανατολισμός του FP\_Gun. Στην τρίτη ροή κώδικα ενημερώνεται το σημείο πηγής του ParticleSystem του CurrentBeam και η τοποθεσία του στον τρισδιάστατο κόσμο (Εικόνα 215).



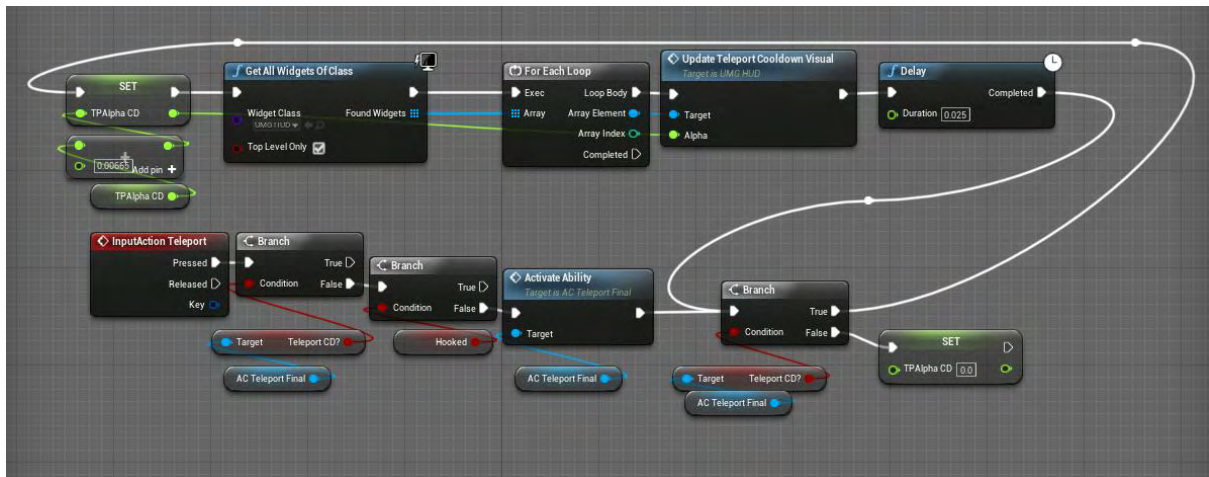
Εικόνα 215 FirstPersonCharacter part 17

Το InputAction ShieldDeployment ενεργοποιείται όταν ο παίκτης πατήσει το πλήκτρο 1 και μπορεί να χρησιμοποιήσει την ασπίδα, αν αυτή είναι διαθέσιμη. Εφόσον χρησιμοποιηθεί η ασπίδα, ξεκινά η ενημέρωση της τιμής της μεταβλητής ShieldAlphaCD και η κλήση της συνάρτησης UpdateShieldCDVisual με είσοδο τη μεταβλητή αυτή (Εικόνα 216). Η συνάρτηση ActivateAbillity του AC\_ForceWall έχει αναλυθεί σε προηγούμενη εικόνα.



Εικόνα 216 FirstPersonCharacter part 18

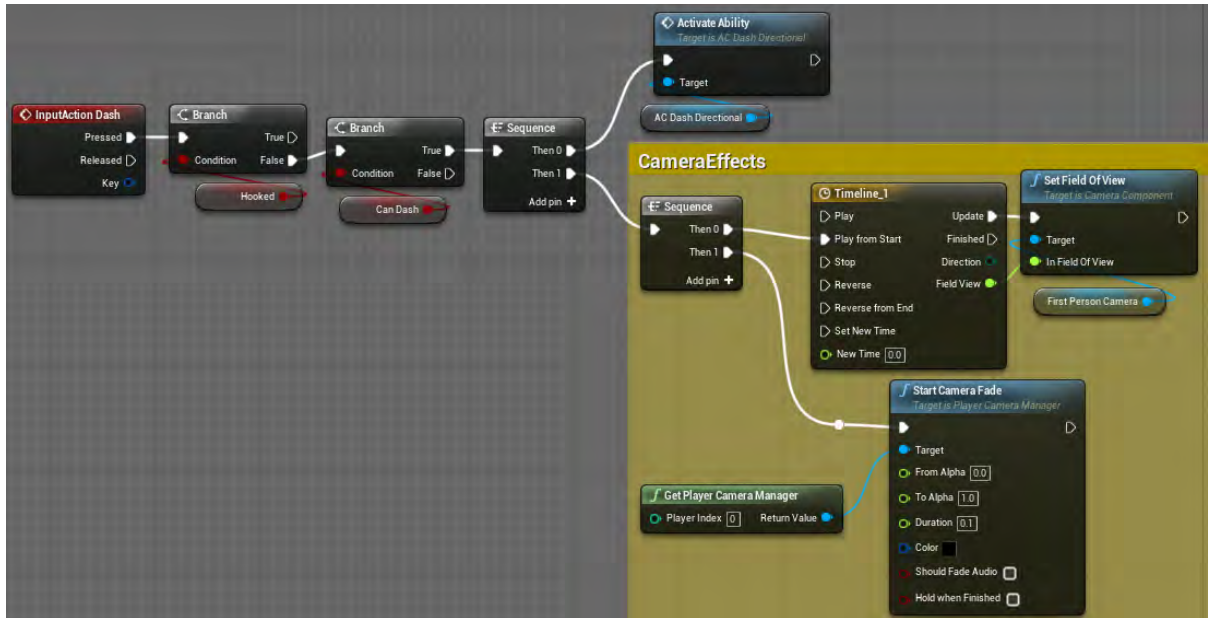
Το InputAction Teleport ενεργοποιείται όταν ο παίκτης πατήσει το δεξί κλικ του ποντικιού και μπορεί να χρησιμοποιήσει την τηλεμεταφορά, αν αυτή είναι διαθέσιμη και δεν κάνει χρήση της δεξιότητας Hook. Εφόσον χρησιμοποιηθεί η τηλεμεταφορά, ξεκινά η ενημέρωση της τιμής της μεταβλητής TPAlphaCD και η κλήση της συνάρτησης UpdateTeleportCooldownVisual με είσοδο τη μεταβλητή αυτή (Εικόνα 217). Η συνάρτηση ActivateAbility του AC\_TeleportFinal έχει αναλυθεί σε προηγούμενη εικόνα.



Εικόνα 217 FirstPersonCharacter part 19

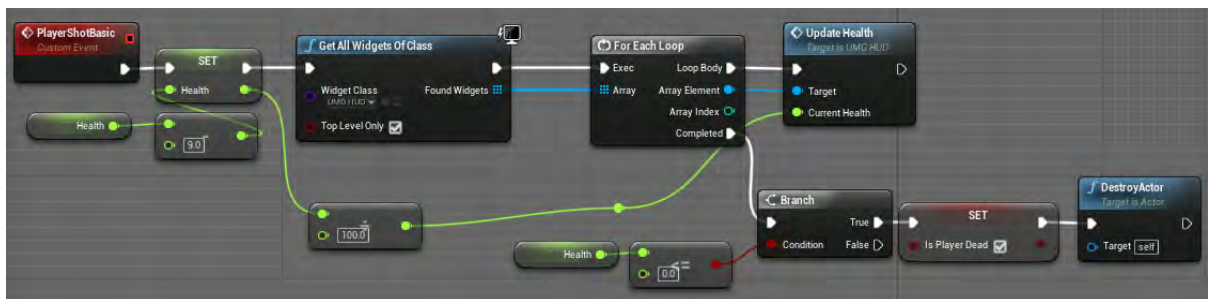
Το InputAction Dash ενεργοποιείται όταν ο παίκτης πατήσει το πλήκτρο Left Shift. Με την προϋπόθεση ότι ο χαρακτήρας του παίκτη δεν κάνει ήδη χρήση της δεξιότητας Hook και Dash, ενεργοποιείται η δεξιότητα Dash. Η συνάρτηση

ActivateAbility του AC\_DashDirectional έχει αναλυθεί σε προηγούμενη εικόνα. Στη συνέχεια δημιουργούνται στιγμιαία οπτικά εφέ τροποποιώντας την κάμερα του χαρακτήρα του παίκτη (Εικόνα 218).



Εικόνα 218 FirstPersonCharacter part 20

Η συνάρτηση BasicShotDamage μειώνει τη ζωή (Health) του χαρακτήρα του παίκτη, καλεί τη συνάρτηση UpdateHealth του UMG\_HUD με είσοδο τη ζωή του παίκτη σε ποσοστό. Στη συνέχεια ελέγχεται αν η ζωή του παίκτη είναι μηδέν ή και λιγότερη. Σε περίπτωση που το αποτέλεσμα του ελέγχου είναι αληθές, τότε καταστρέφεται ο χαρακτήρας (Εικόνα 219).

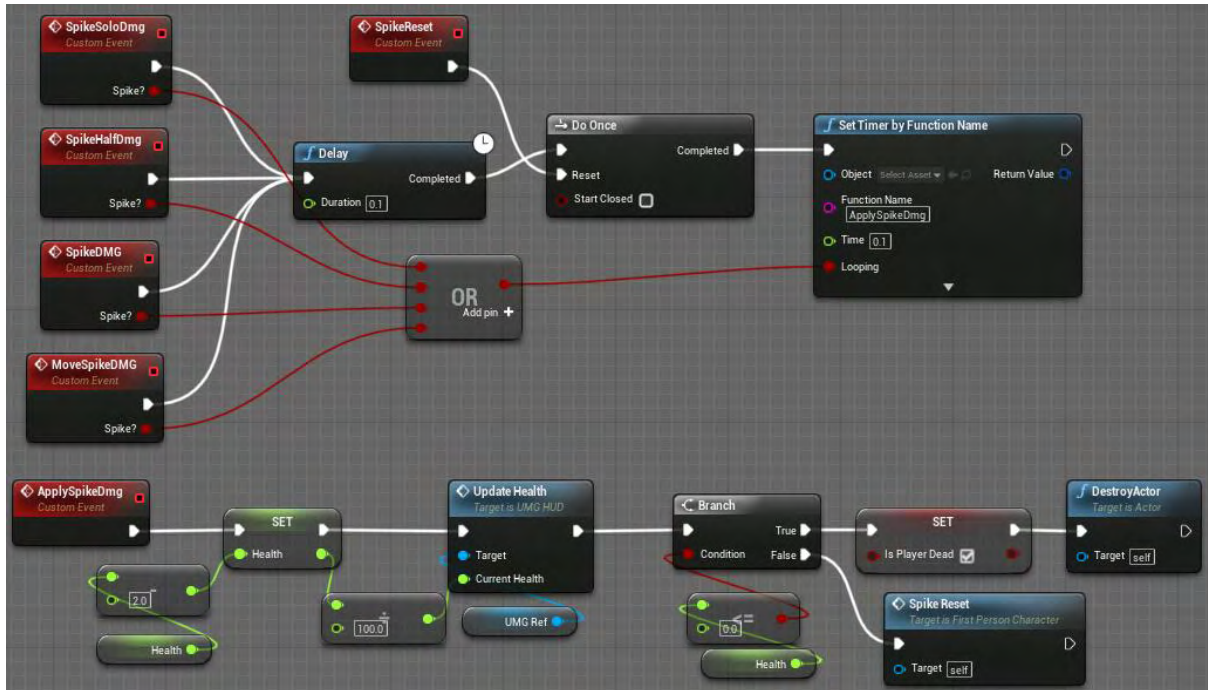


Εικόνα 219 FirstPersonCharacter part 21

Στην Εικόνα 220 απεικονίζεται η λειτουργία των συναρτήσεων που καλούνται από τους actors τύπου Spike. Όσο το πλέγμα σύγκρουσης του χαρακτήρα του παίκτη

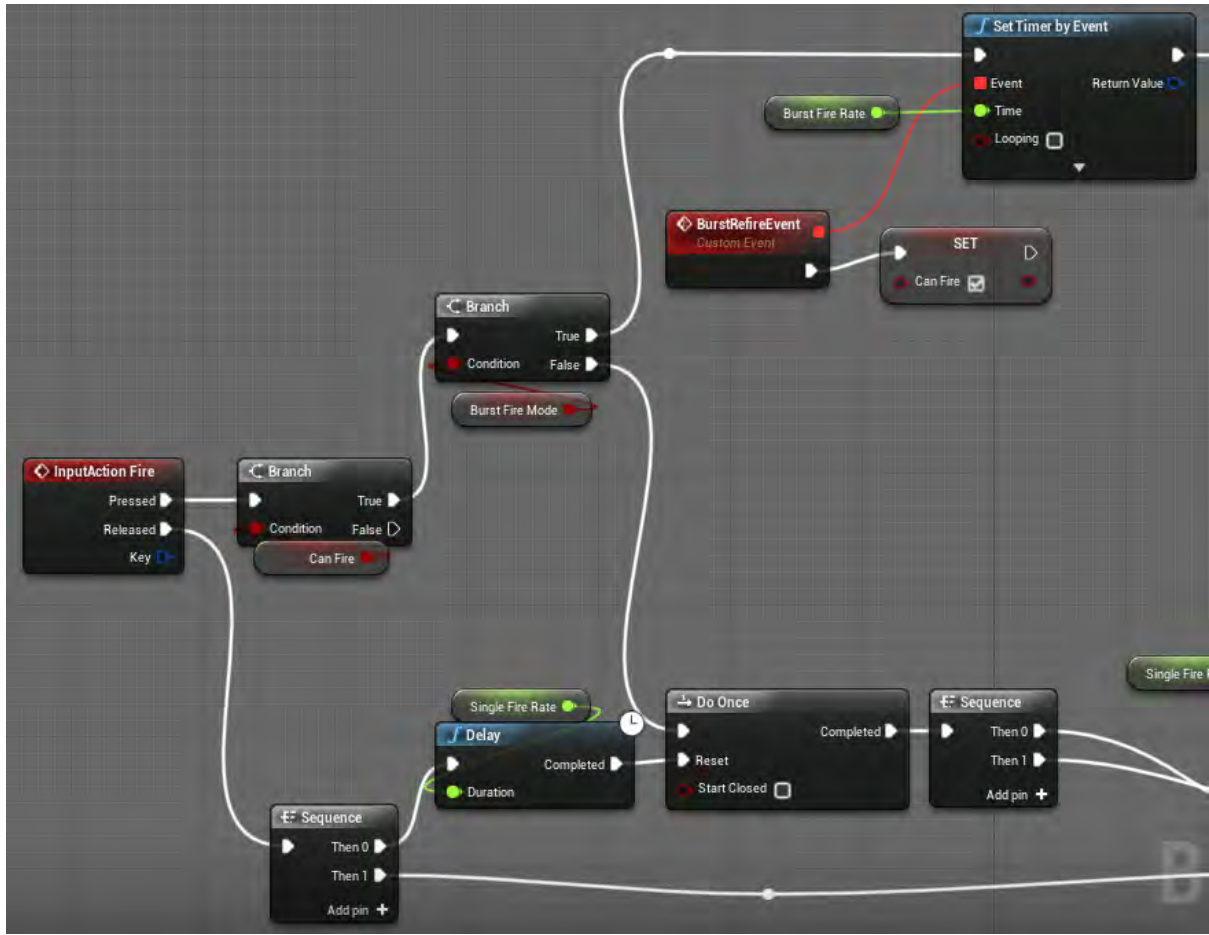


βρίσκεται σε σύγκρουση με κάποιον actor τύπου Spike, η συνάρτηση ApplySpikeDamage καλείται κάθε 1/10 του δευτερολέπτου. Η συνάρτηση αυτή μειώνει τη ζωή του χαρακτήρα, ελέγχει την τιμή της και πράττει αναλόγως.

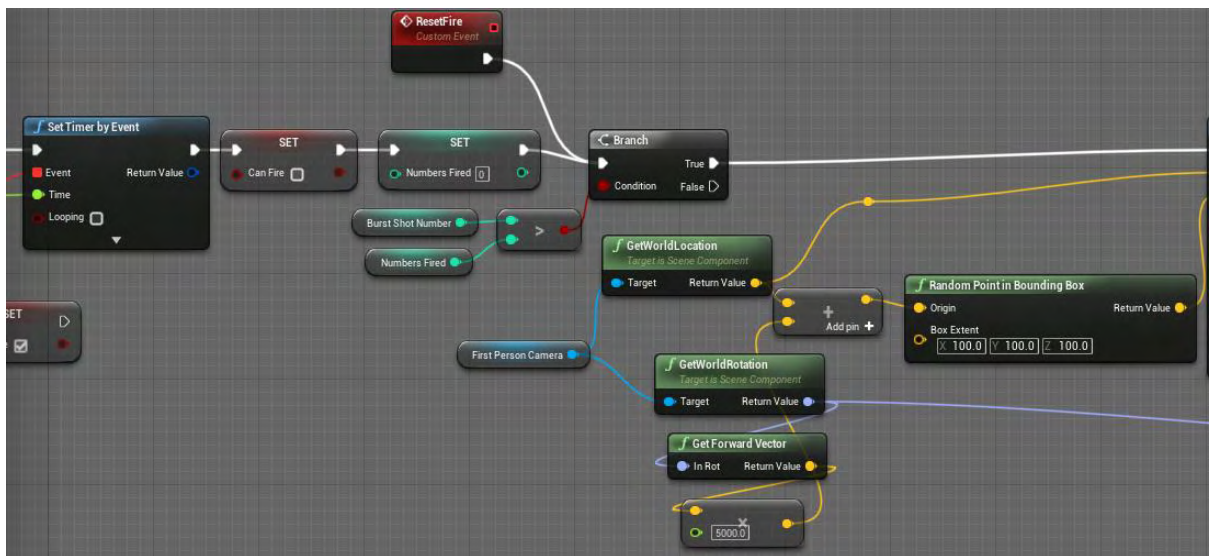


Εικόνα 220 FirstPersonCharacter part 22

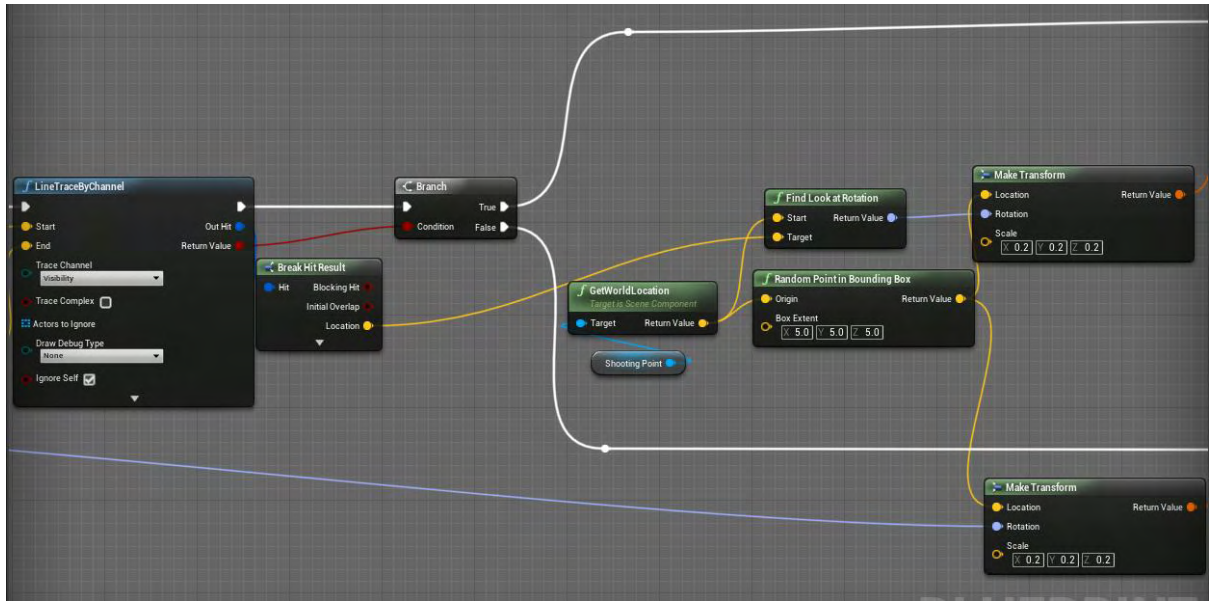
Το InputAction Fire σηματοδοτεί την έναρξη της δυνατότητας ρίψης βλήματος από τον χαρακτήρα του παίκτη και ενεργοποιείται με το αριστερό πλήκτρο του ποντικιού του παίκτη. Στις Εικόνες 221- 227 απεικονίζεται λεπτομερώς ο τρόπος υλοποίησης της δυνατότητας ρίψεων βλημάτων.



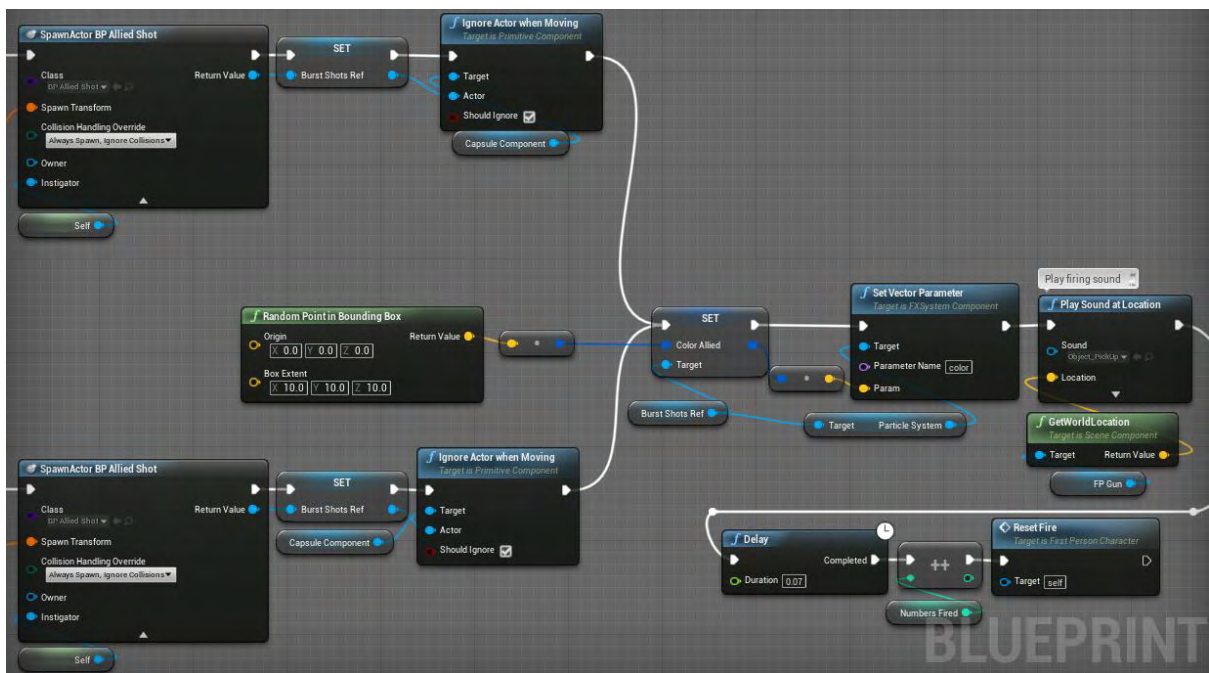
Εικόνα 221 FirstPersonCharacter part 23



Εικόνα 222 FirstPersonCharacter part 24



Εικόνα 223 FirstPersonCharacter part 25

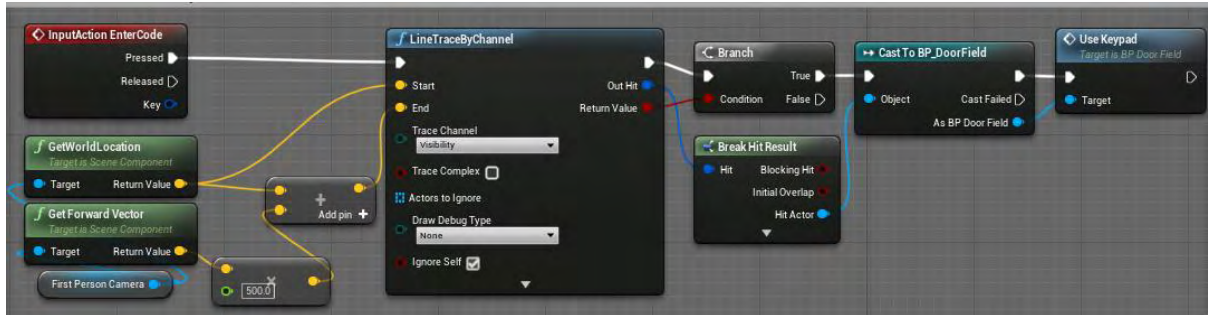


Εικόνα 224 FirstPersonCharacter part 26



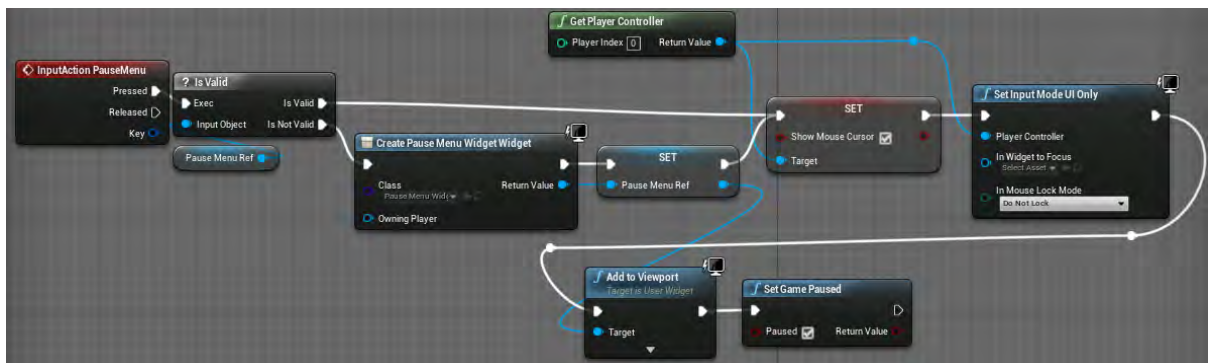


Το InputAction EnterCode ενεργοποιείται όταν ο παίκτης πατήσει το πλήκτρο Q. Εφόσον ο χαρακτήρας του παίκτη βρίσκεται σε κοντινή απόσταση στραμμένος προς έναν actor τύπου BP\_DoorField, καλείται η συνάρτηση του UseKeypad, που έχει αναλυθεί σε προηγούμενη εικόνα (Εικόνα 228).



Εικόνα 228 FirstPersonCharacter part 30

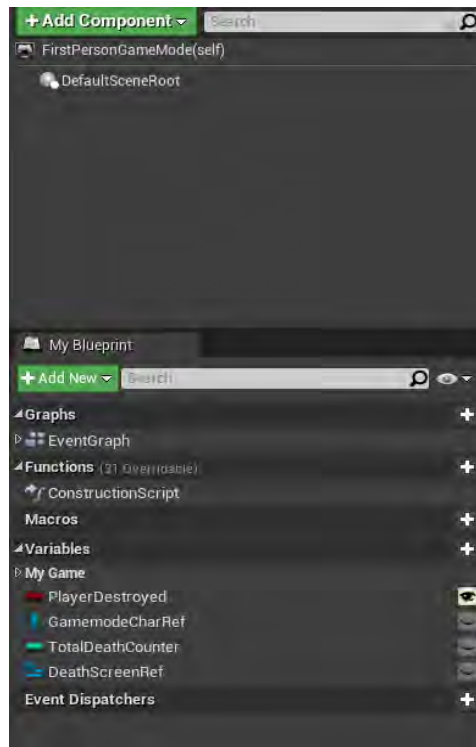
Το InputAction PauseMenu ενεργοποιείται όταν ο παίκτης πατήσει το πλήκτρο Esc ή το P. Έχει ως αποτέλεσμα το πάγωμα του παιχνιδιού και την εμφάνιση του Widget τύπου PauseMenu\_Widget (Εικόνα 229).



Εικόνα 229 FirstPersonCharacter part 31

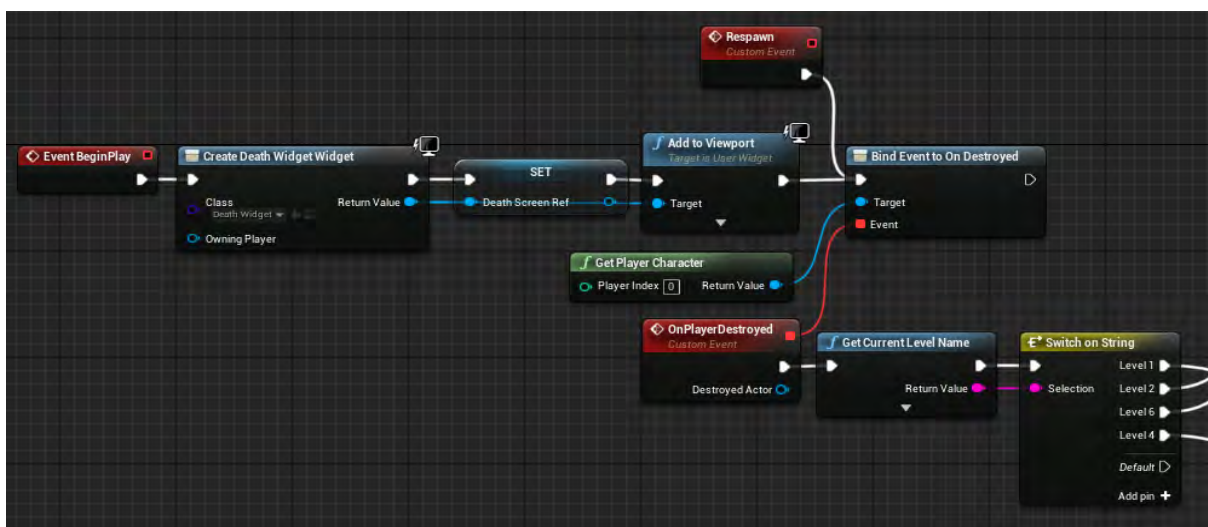
## Ενότητα 5.4.2 FirstPersonGameMode

Το FirstPersonGameMode είναι μεγάλης σημασίας για το παιχνίδι, αφού καθορίζει το Player Controller και το Player Pawn.



Εικόνα 230 FirstPersonGameMode part 1

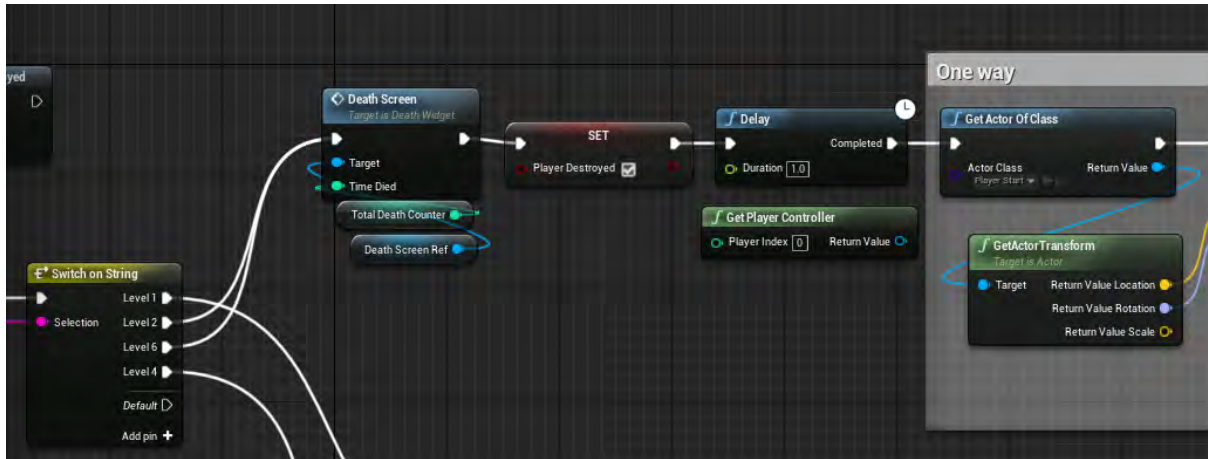
Αρχικά δημιουργείται Widget τύπου Death\_Widget και προστίθεται στο γραφικό περιβάλλον του παίκτη. Δεσμεύεται το OnPlayerDestroyed στο Event το On Destroyed του χαρακτήρα του παίκτη και όταν αυτός καταστραφεί εκτελείται το OnPlayerDestroyed. Ανάλογα με το επίπεδο που βρίσκεται ο χαρακτήρας του παίκτη ακολουθείται συγκεκριμένη ροή πράξεων, η οποία ερμηνεύεται στις επόμενες εικόνες (Εικόνα 231).



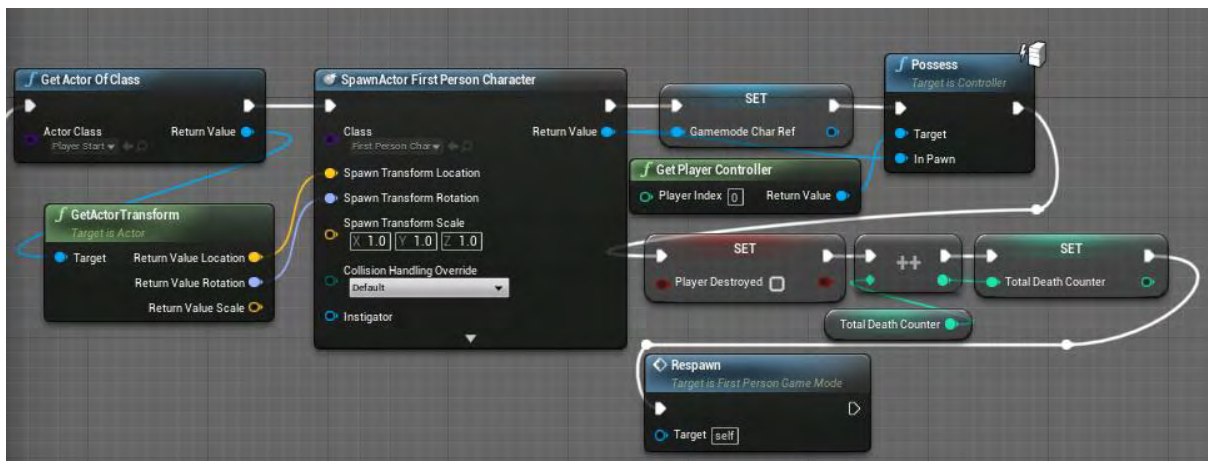
Εικόνα 231 FirstPersonGameMode part 2



Όταν ο χαρακτήρας του παίκτη βρίσκεται είτε στο επίπεδο με την ονομασία Level2 είτε στο επίπεδο με την ονομασία Level6, εκτελείται το γεγονός DeathScreen του Widget με είσοδο τη μεταβλητή TotalDeathCounter. Μετά από μια χρονοκαθυστέρηση ενός δευτερολέπτου (Εικόνα 232) δημιουργείται actor τύπου FirstPersonCharacter και αρχικοποιείται. Κατόπιν αυξάνεται η τιμή της μεταβλητής TotalDeathCounter (Εικόνα 233).

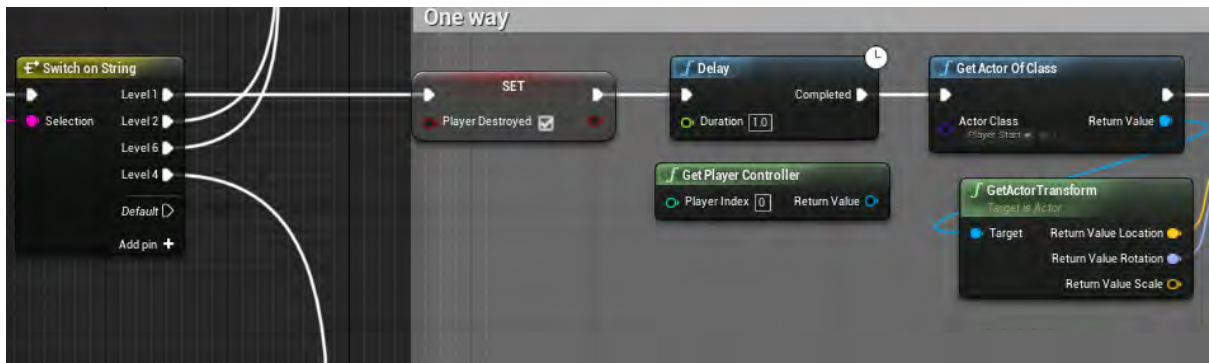


Εικόνα 232 FirstPersonGameMode part 3



Εικόνα 233 FirstPersonGameMode part 4

Όπως φαίνεται στην Εικόνα 234 και στην Εικόνα 235 ,όταν ο χαρακτήρας του παίκτη βρίσκεται στο επίπεδο με την ονομασία Level1, δημιουργείται (μετά από μια χρονοκαθυστέρηση ενός δευτερολέπτου) actor τύπου FirstPersonCharacter και αρχικοποιείται .

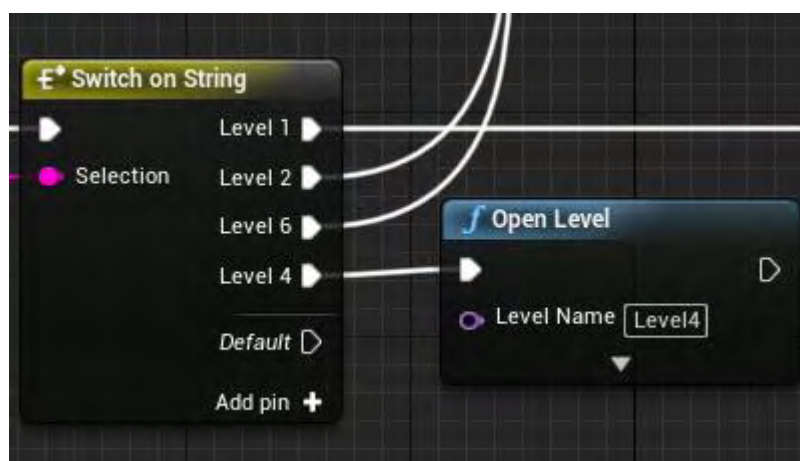


Εικόνα 234 FirstPersonGameMode part 5



Εικόνα 235 FirstPersonGameMode part 6

Στο επίπεδο με την ονομασία Level4 η καταστροφή του χαρακτήρα του παίκτη σηματοδοτεί την επανεκκίνηση του επιπέδου. Αυτό συμβαίνει για να συμβαδίζει με την πλοκή του παιχνιδιού, αφού στο συγκεκριμένο επίπεδο ο χαρακτήρας δεν επανέρχεται στη ζωή από τον master του Feli (Εικόνα 236)

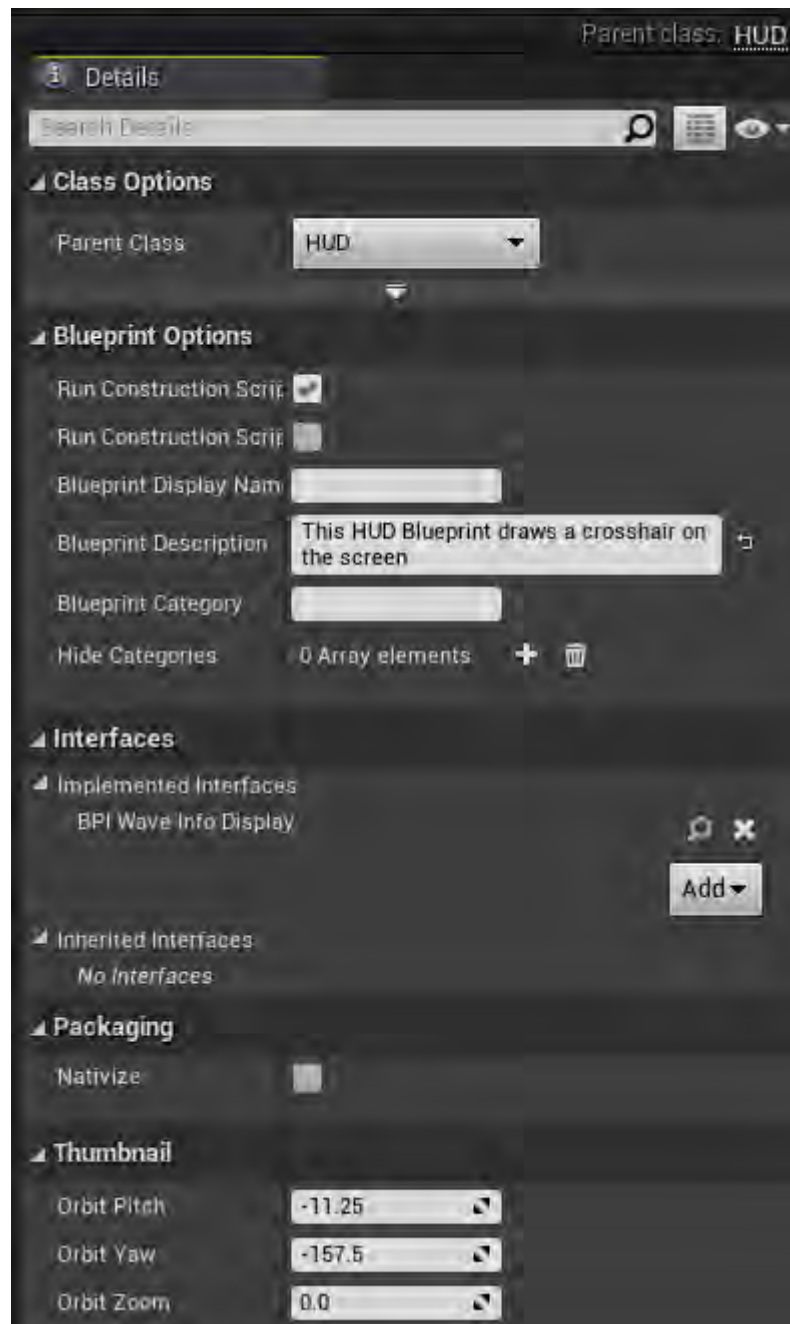


Εικόνα 236 FirstPersonGameMode part 7

## Ενότητα 5.4.3 FirstPersonHUD

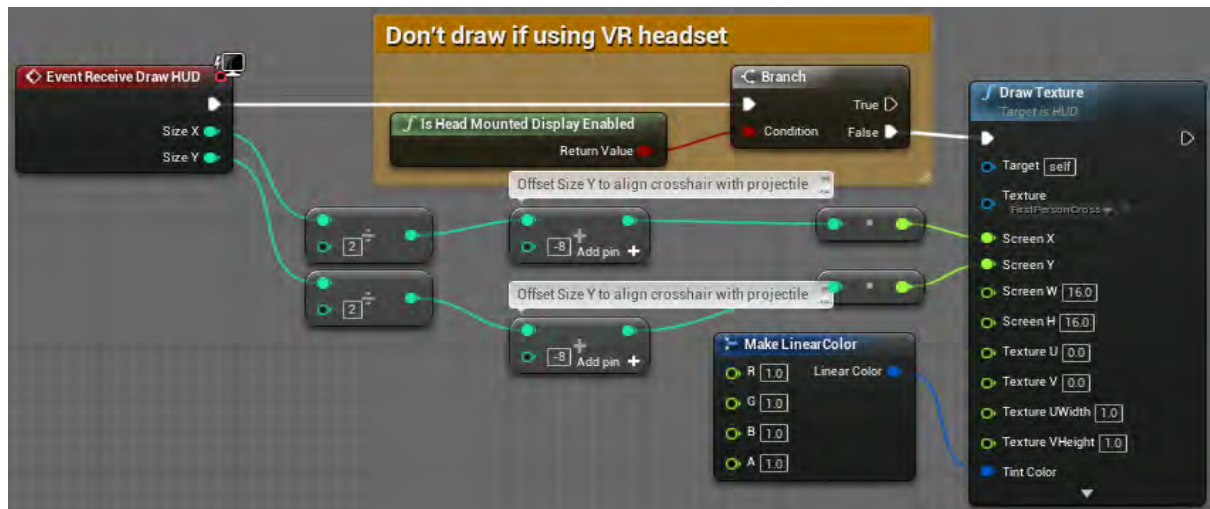
---

Το FirstPersonHUD έχει Parent Class HUD και υλοποιεί τη διεπαφή BPI\_WaveInfoDisplay (Εικόνα 237).



Εικόνα 237 FirstPersonHUD part 1

Στην εικόνα 238 απεικονίζεται η προϋπάρχουσα λειτουργία του FirstPersonHUD, η οποία ζωγραφίζει στο κέντρο της οθόνης ένα κόκκινο σταυρόνημα.



Εικόνα 238 FirstPersonHUD part 2

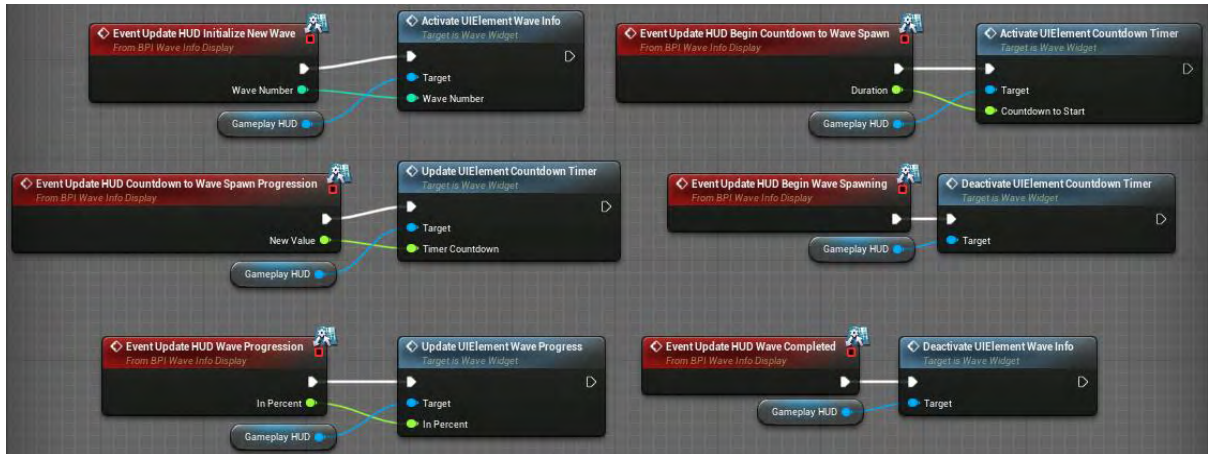
Όταν η ονομασία του επιπέδου είναι Level4, δημιουργείται ένα Widget τύπου Wave\_Widget, αποθηκεύεται στη μεταβλητή GameplayHUD και προστίθεται στην οθόνη του παίκτη (Εικόνα 239).



Εικόνα 239 FirstPersonHUD part 3

Στην εικόνα 240 απεικονίζονται οι λειτουργικότητες των συναρτήσεων του BPI\_WaveInfoDisplay στο FirstPersonHUD. Η υλοποίηση της λειτουργικότητας τους είναι το κάλεσμα των συναρτήσεων του Wave\_Widget με τις σωστές τιμές στις εισόδους τους. Στην ενότητα Widget θα αναλυθούν τα αποτελέσματα των συγκεκριμένων συναρτήσεων (Εικόνα 240).





Εικόνα 240 FirstPersonHUD part 4

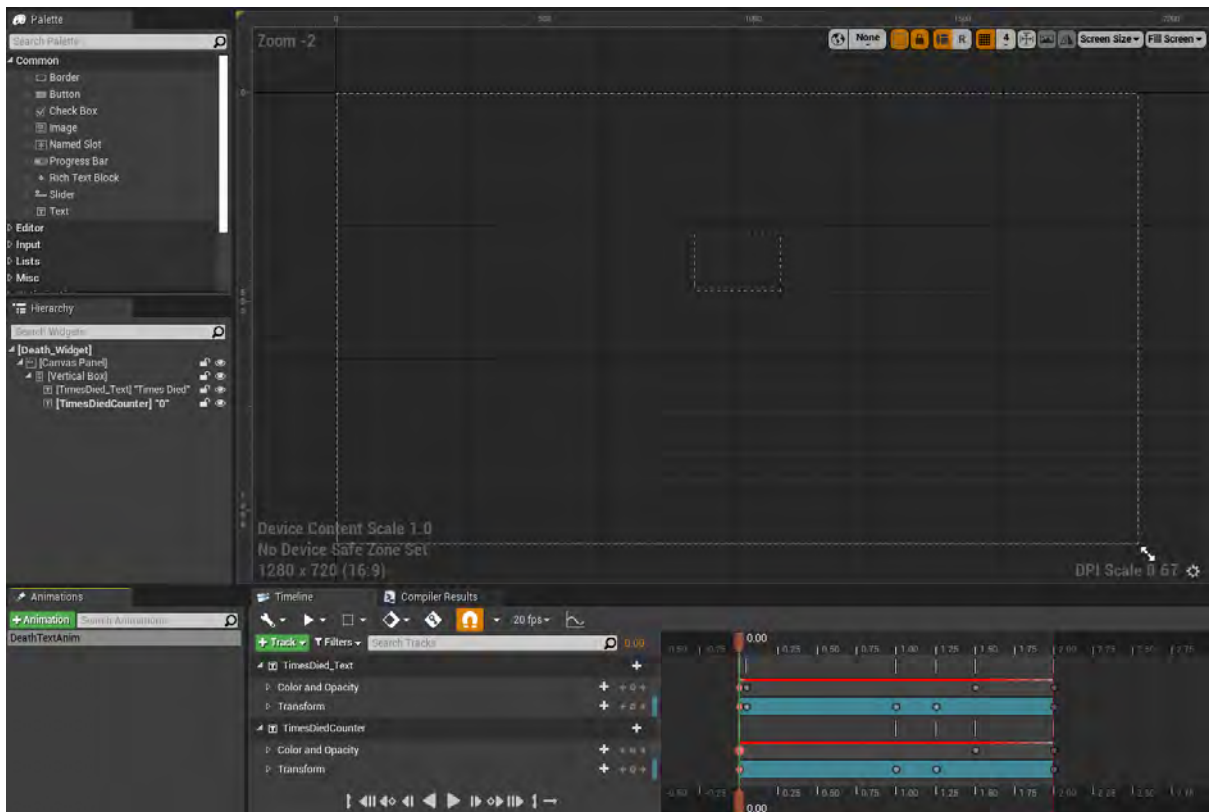
## 5.5 Widget

### Ενότητα 5.5.1 Death\_Widget

Το Death\_Widget δημιουργείται στο Event BeginPlay του FirstPersonGameMode με σκοπό να ενημερώνει τον παίκτη πόσες φορές έχει καταστραφεί ο χαρακτήρας του.

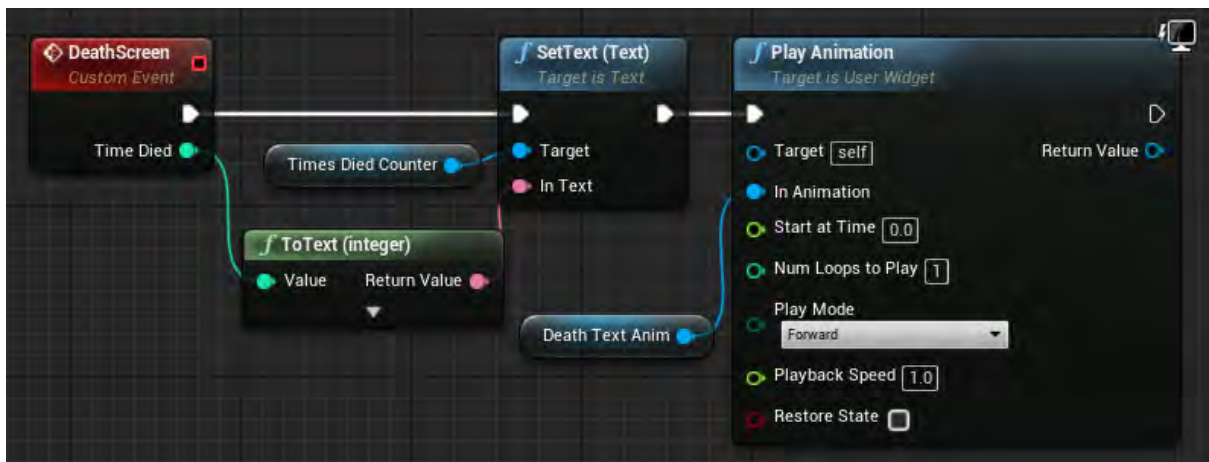
Στην εικόνα 241 απεικονίζεται η καρτέλα Designer του Death\_Widget. Τα κείμενα έχουν αρχικοποιηθεί με μηδενικό opacity, ώστε να είναι ορατά μόνο κατά την διάρκεια του Animation DeathTextAnim. Το Animation αυτό τροποποιεί τις τιμές των κατηγοριών Color and Opacity και Transform των κειμένων.





Εικόνα 241 Death\_Widget1

Στην Εικόνα 242 απεικονίζεται η συνάρτηση DeathScreen, της οποίας η λειτουργικότητα είναι να ενημερώνει το κείμενο TimesDiedCounter που συμβολίζει τον αριθμό των φορών που έχει καταστραφεί ο χαρακτήρας του παίκτη. Στην συνέχεια ενεργοποιείται το DeathTextAnim και ο αριθμός αυτός γίνεται ορατός για την διάρκεια του animation.

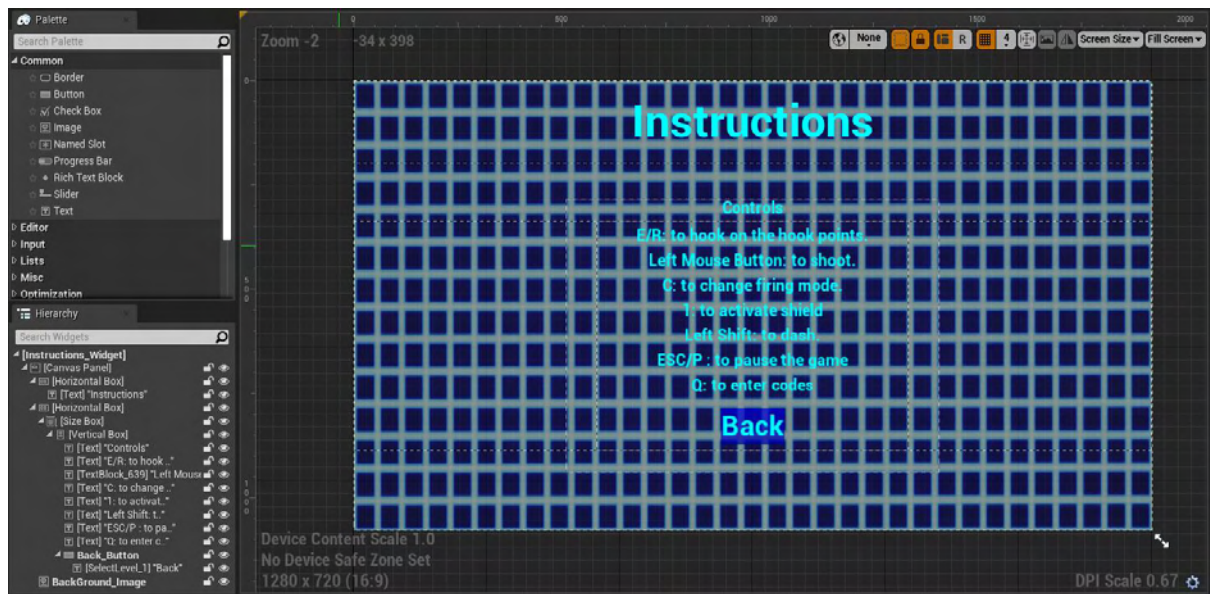


Εικόνα 242 Death\_Widget2

## Ενότητα 5.5.2 Instructions\_Widget

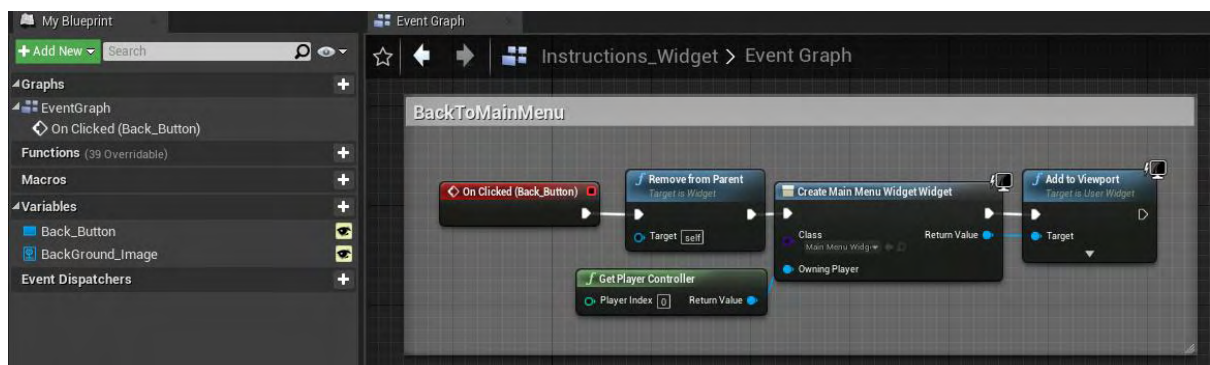
Ο σκοπός του Instructions\_Widget είναι να ενημερώσει τον παίκτη σχετικά με την λειτουργικότητα των πλήκτρων που απαιτούνται στο The Challenger.

Στην Εικόνα 243 απεικονίζεται η καρτέλα Designer του Instructions\_Widget.



Εικόνα 243 Instructions\_Widget1

Η μοναδική αλληλεπίδραση με τον παίκτη στο Widget είναι το κουμπι Back\_Button. Όταν πατηθεί, ενεργοποιείται το On Clicked που επιστρέφει τον παίκτη στο αρχικό μενού (Εικόνα 244).



Εικόνα 244 Instructions\_Widget2

## Ενότητα 5.5.3 LevelSelection\_Widget

Το LevelSelection\_Widget δίνει τη δυνατότητα στον παίκτη να επιλέξει το επίπεδο από το οποίο θα ξεκινήσει το παιχνίδι.

Στην Εικόνα 245 απεικονίζεται η καρτέλα Designer του LevelSelection\_Widget.

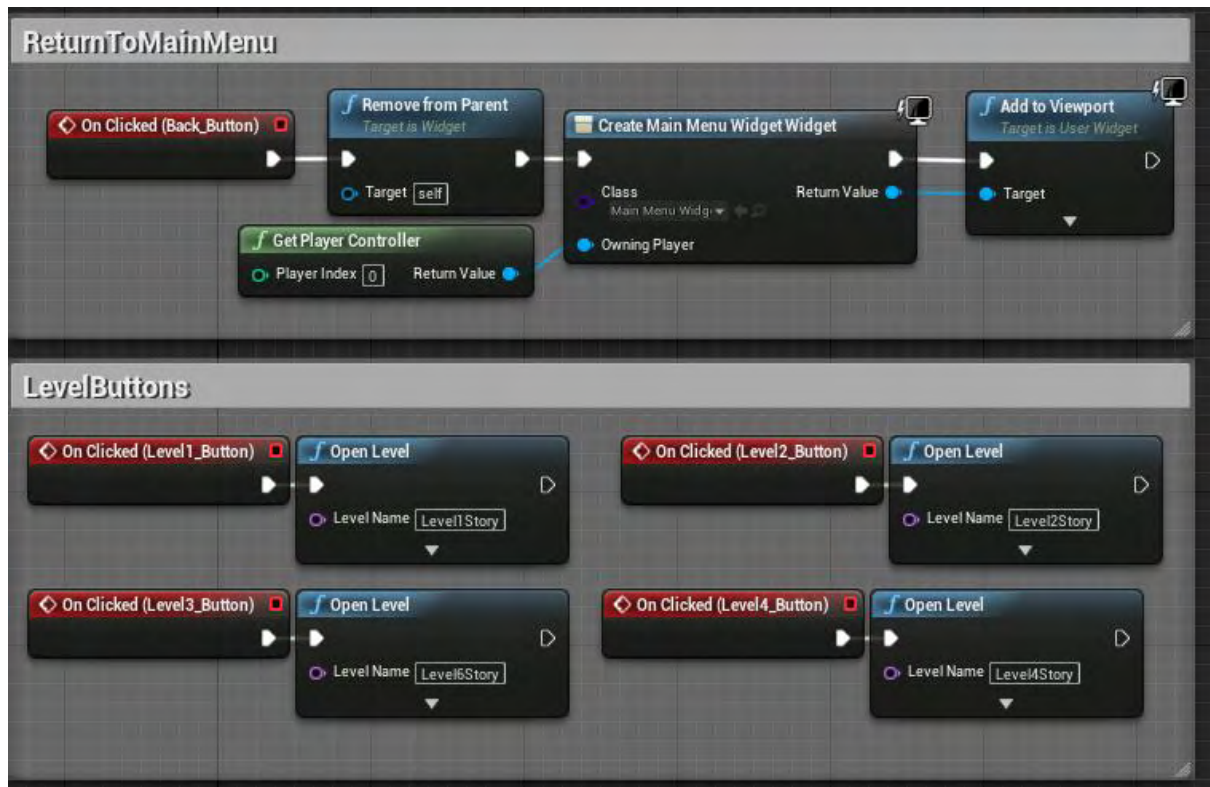


Εικόνα 245 LevelSelection\_Widget1

Στην Εικόνα 246 απεικονίζεται η υλοποίηση της λειτουργικότητας των κουμπιών του συγκεκριμένου Widget. Με το πάτημα του κάθε κουμπιού ενεργοποιείται το αντίστοιχο On Clicked. Συγκεκριμένα:

- Το Back\_Button μεταφέρει τον παίκτη στο αρχικό μενού.
- Το Level1\_Button φορτώνει το επίπεδο με την ονομασία Level1\_Story.
- Το Level2\_Button φορτώνει το επίπεδο με την ονομασία Level2\_Story.
- Το Level3\_Button φορτώνει το επίπεδο με την ονομασία Level6\_Story.
- Το Level4\_Button φορτώνει το επίπεδο με την ονομασία Level4\_Story.





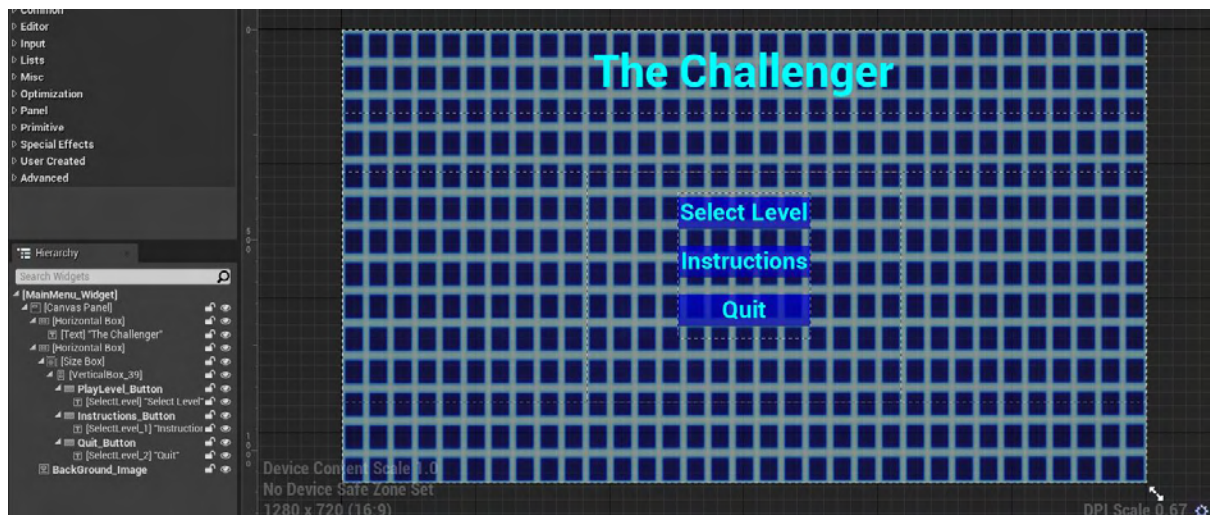
Εικόνα 246 LevelSelection\_Widget2

## Ενότητα 5.5.4 MainMenu\_Widget

Το MainMenu\_Widget παρέχει στον παίκτη τις εξής δυνατότητες:

- εξόδου από το παιχνίδι
- μετάβασης στο Instructions\_Widget, ώστε να δει τις οδηγίες του παιχνιδιού
- μετάβασης στο LevelSelection\_Widget, ώστε να μπορεί να επιλέξει το επίπεδο που επιθυμεί.

Στην Εικόνα 247 απεικονίζεται η καρτέλα Designer του MainMenu\_Widget.

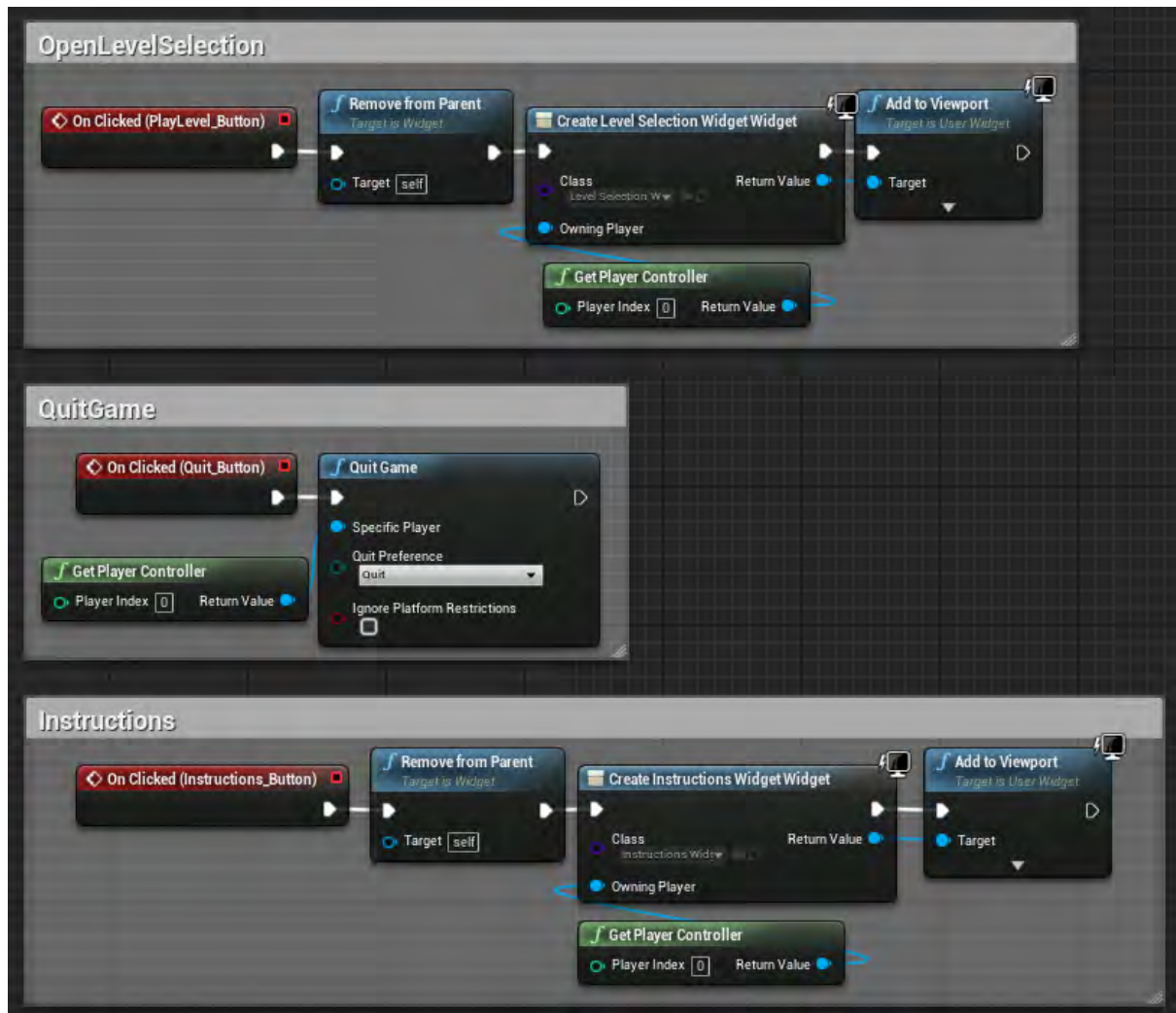


Εικόνα 247 MainMenu\_Widget1

Στην Εικόνα 248 απεικονίζεται η υλοποίηση της λειτουργικότητας των κουμπιών του Widget. Με το πάτημα του κάθε κουμπιού ενεργοποιείται το αντίστοιχο On Clicked. Συγκεκριμένα:

- Το PlayLevel\_Button αφαιρεί από το περιβάλλον του παίκτη το MainMenu\_Widget. Στη συνέχεια δημιουργεί και προσθέτει στο περιβάλλον του παίκτη το LevelSelection\_Widget. Μεταφέρει δηλαδή τον παίκτη στο Widget επιλογής επιπέδου.
- Το Quit\_Button κλείνει το παιχνίδι.
- Το Instructions\_Button αφαιρεί από το περιβάλλον του παίκτη το MainMenu\_Widget. Στη συνέχεια δημιουργεί και προσθέτει στο περιβάλλον του παίκτη το Instructions\_Widget.

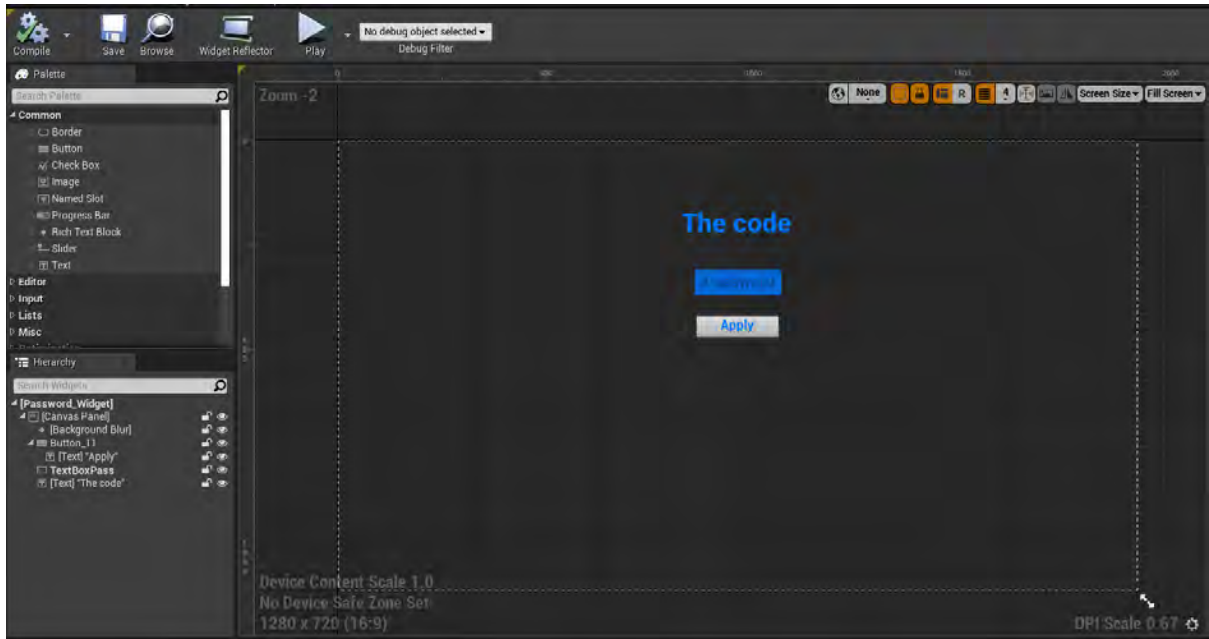




Εικόνα 248 MainMenu\_Widget3

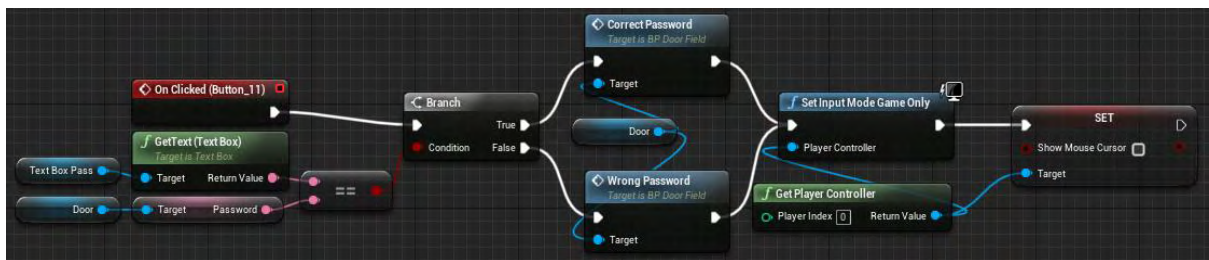
### Ενότητα 5.5.5 Password\_Widget

Το Password\_Widget χρησιμοποιείται για την εισαγωγή κωδικών από τον παίκτη. Στην Εικόνα 249 απεικονίζεται η καρτέλα Designer του Password\_Widget.



Εικόνα 249 Password\_Widget1

Κατά την ενεργοποίηση του On Clicked (Button\_11) πραγματοποιείται έλεγχος του περιεχομένου TextBoxPass. Αν ταυτοποιείται με τον αντίστοιχο κωδικό, καλείται η συνάρτηση CorrectPassword του Door (Reference σε actor τύπου BP\_DoorField), ενώ αν δεν ταυτοποιείται, καλείται η συνάρτηση WrongPassword του Door. Τέλος, τίθεται το Input Mode σε Game Only και αποκρύπτεται ο κέρσορας του ποντικιού (Εικόνα 250) .



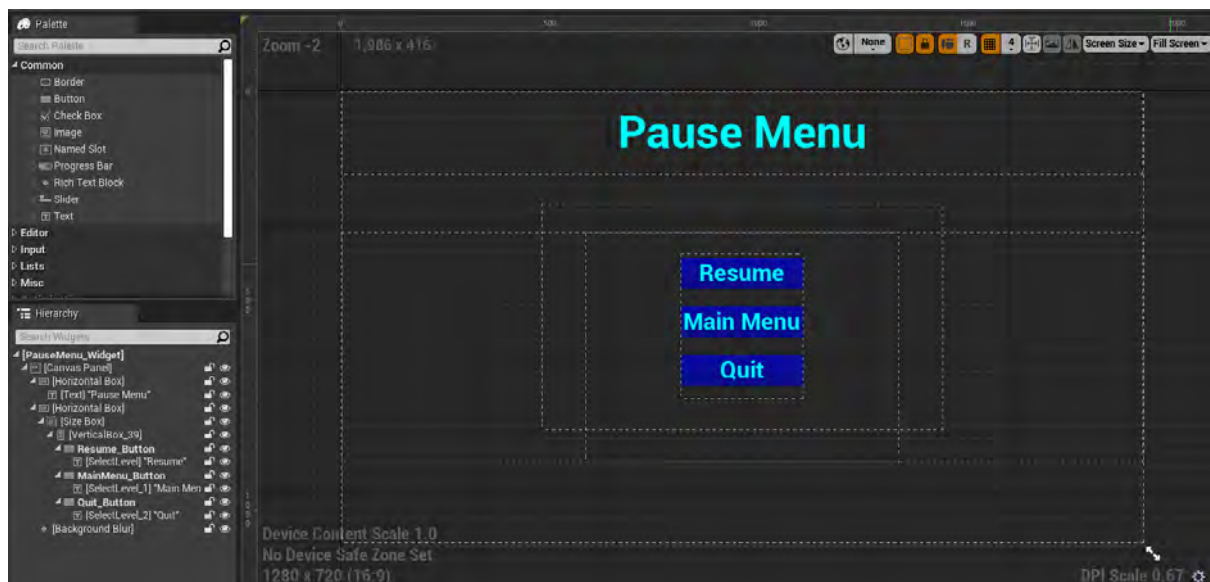
Εικόνα 250 Password\_Widget2

## Ενότητα 5.5.6 PauseMenu\_Widget

Το PauseMenu\_Widget εμφανίζεται στην οθόνη του παίκτη όποτε πατηθεί το αντίστοιχο πλήκτρο. Ο σκοπός του είναι να δώσει την δυνατότητα στον παίκτη να παγώνει την λειτουργία του παιχνιδιού και να του δίνει μια από παρακάτω επιλογές:

- συνέχισης του παιχνιδιού
- επιστροφής στο αρχικό μενού
- εξόδου από το παιχνίδι.

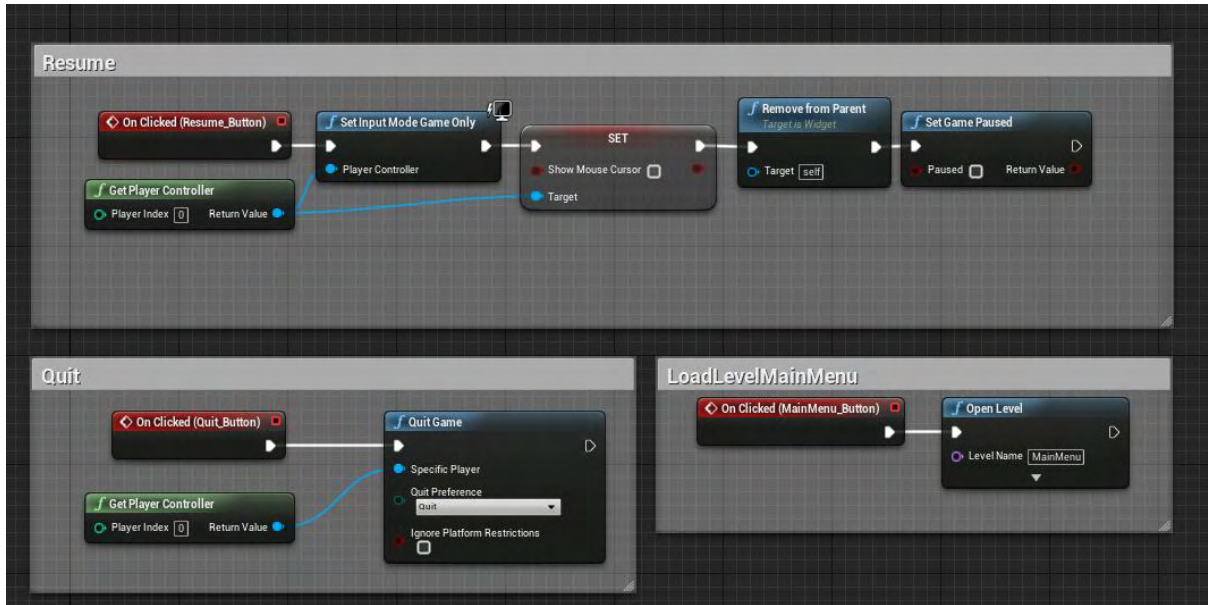
Στην Εικόνα 251 απεικονίζεται η καρτέλα Designer του PauseMenu\_Widget.



Εικόνα 251 PauseMenu\_Widget1

Στην Εικόνα 252 απεικονίζεται η υλοποίηση της λειτουργικότητας των κουμπιών του Widget. Με το πάτημα του κάθε κουμπιού ενεργοποιείται το αντίστοιχο On Clicked. Συγκεκριμένα:

- Το Resume\_Button θέτει το Input Mode σε Game Only, αποκρύπτει τον κέρσορα του ποντικιού, αφαιρεί το Widget από την οθόνη και συνεχίζει το παιχνίδι.
- Το Quit\_Button κλείνει το παιχνίδι.
- Το MainMenu\_Button ανοίγει το επίπεδο με την ονομασία MainMenu.

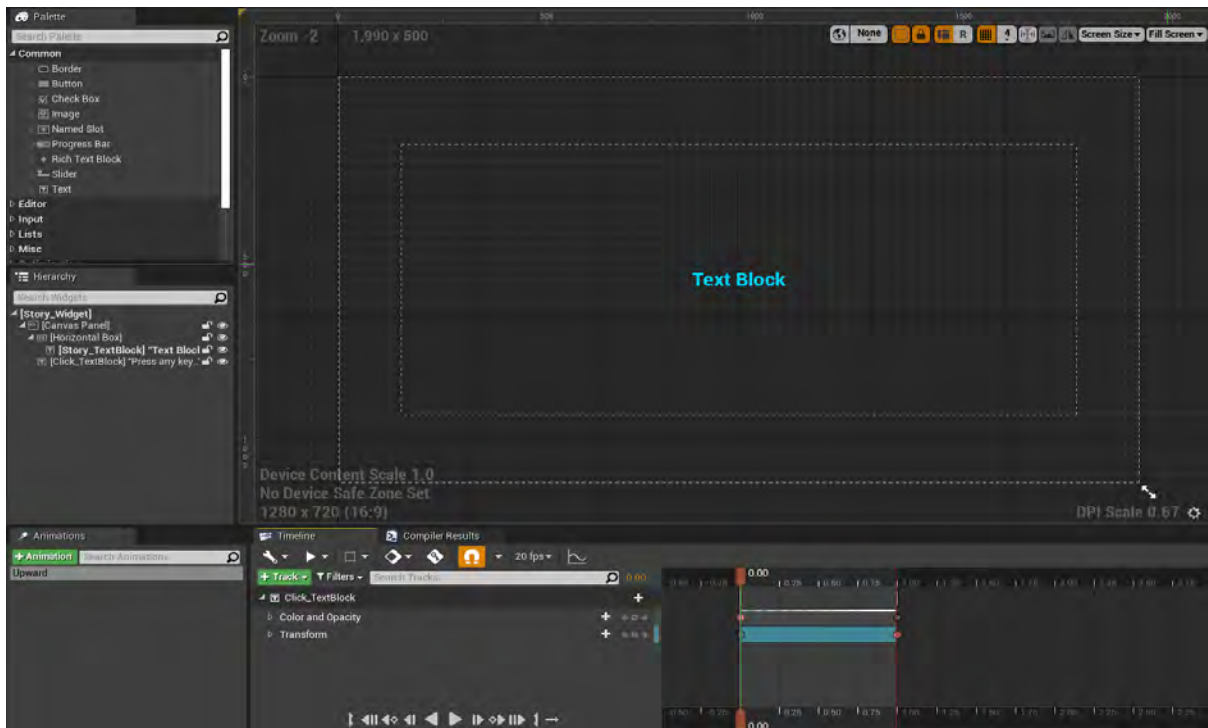


Εικόνα 252 PauseMenu\_Widget2

## Ενότητα 5.5.7 Story\_Widget

Το Story\_Widget έχει σκοπό την αφήγηση της πλοκής του παιχνιδιού χρησιμοποιώντας εφέ γραφομηχανής.

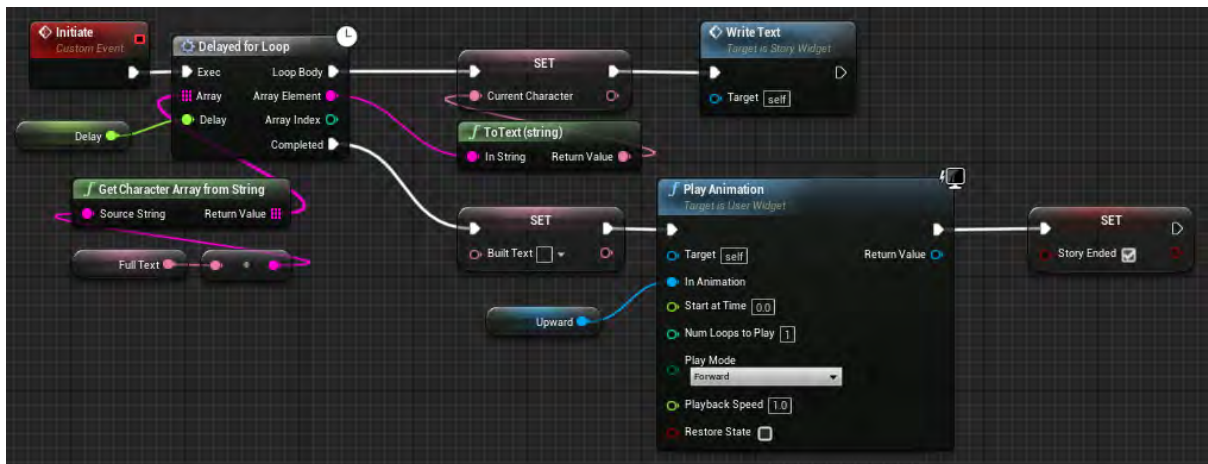
Στην Εικόνα 253 απεικονίζεται η καρτέλα Designer του Story\_Widget.



Εικόνα 253 Story\_Widget1

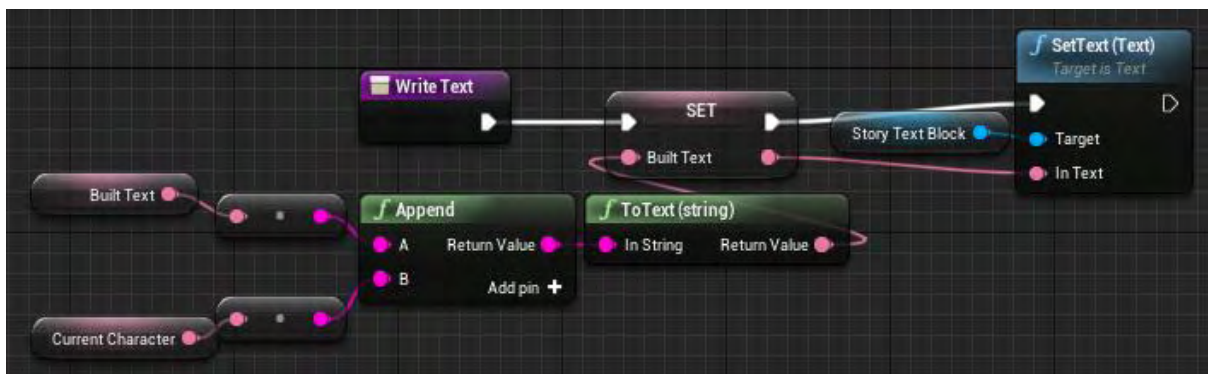


Η συνάρτηση Initialize διατρέχει το κείμενο της μεταβλητής FullText. Καθώς το διατρέχει με μια μικρή χρονοκαθυστέρηση, καλεί τη συνάρτηση WriteText για κάθε χαρακτήρα του κειμένου. Μόλις πραγματοποιηθεί αυτό, αναπαράγεται το Animation Upward και η τιμή της μεταβλητής StoryEnded μετατρέπεται σε True (Εικόνα 254).



Εικόνα 254 Story\_Widget2

Η συνάρτηση WriteText προσθέτει στο τέλος του κειμένου, που υπάρχει ήδη στην οθόνη, τον επόμενο χαρακτήρα του κειμένου (Εικόνα 255).



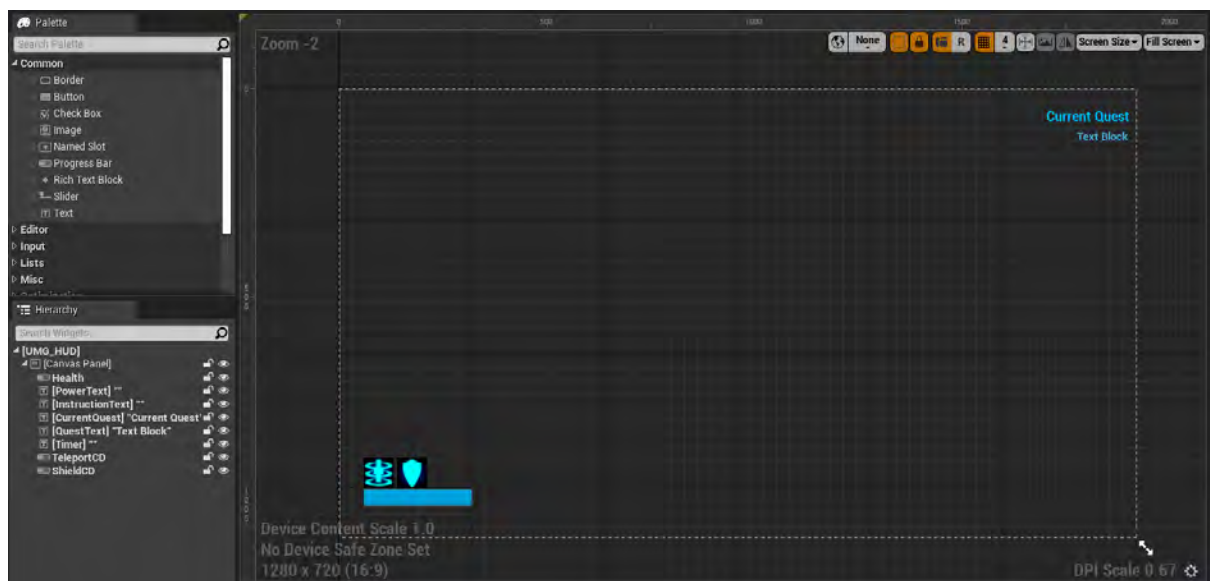
Εικόνα 255 Story\_Widget3



## Ενότητα 5.5.8 UMG\_HUD

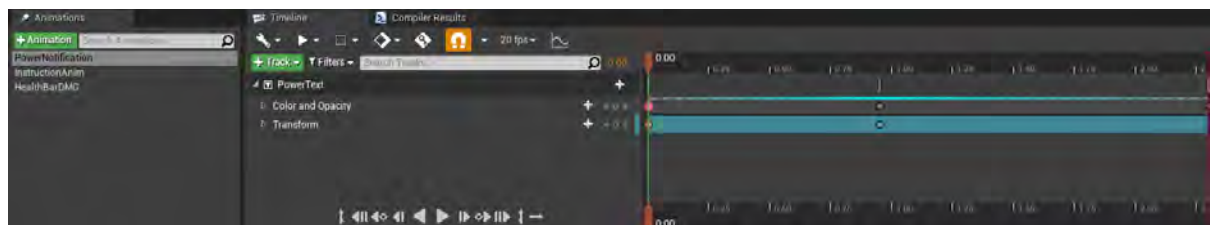
Το UMG\_HUD δημιουργείται σε κάθε επίπεδο και προστίθεται στην οθόνη του παίκτη. Σκοπός του συγκεκριμένου Widget είναι να ενημερώνει τον παίκτη για λειτουργίες του παιχνιδιού, όπως η ζωή του χαρακτήρα και η αποστολή του, κάθε στιγμή του παιχνιδιού.

Στην Εικόνα 256 απεικονίζεται η καρτέλα Designer του UMG\_HUD.

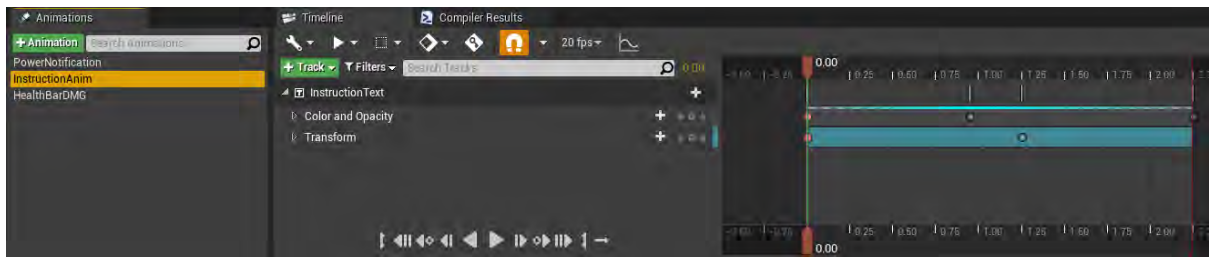


Εικόνα 256 UMG\_HUD1

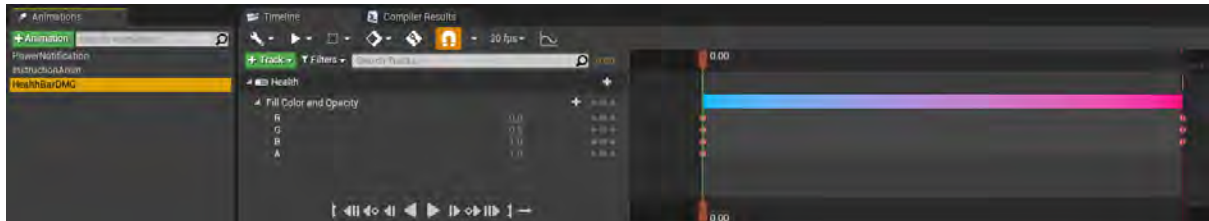
Στην Εικόνα 257, στην Εικόνα 258 και στην Εικόνα 259 απεικονίζονται περισσότερες λεπτομέρειες για τα Animations PowerNotification, InstructionAnim και HealthBarDMG αντίστοιχα.



Εικόνα 257 UMG\_HUD2

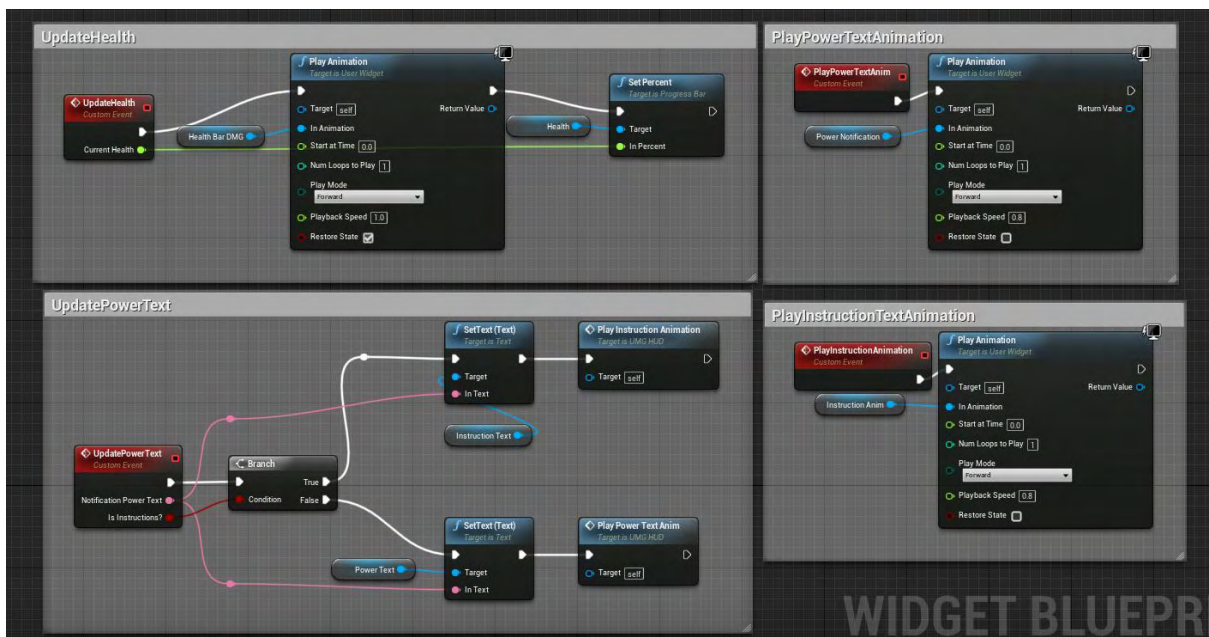


Εικόνα 258 UMG\_HUD3



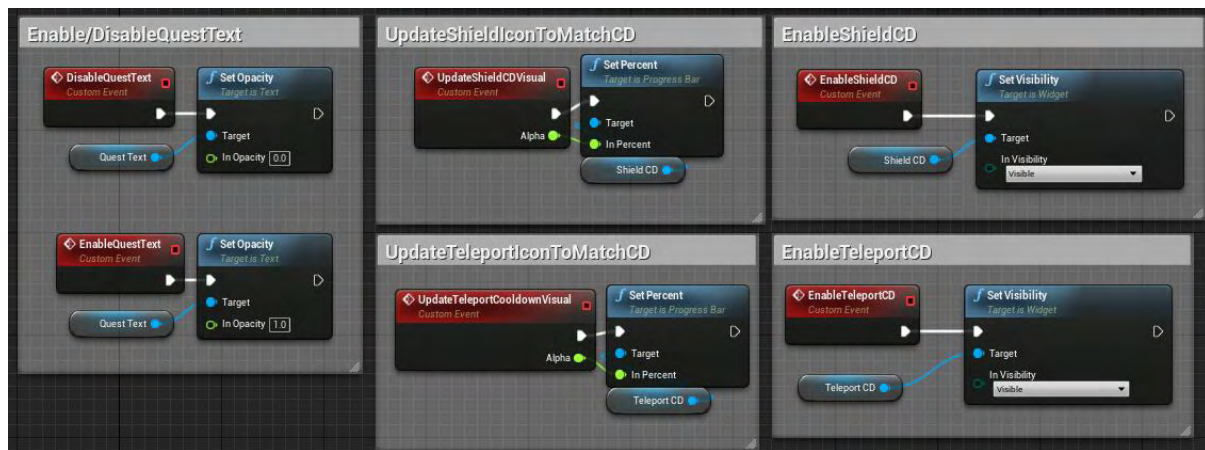
Εικόνα 259 UMG\_HUD4

Η συνάρτηση UpdateHealth δέχεται σαν εισοδο μια τιμή Float, η οποία αντιπροσωπεύει τη ζωή του χαρακτήρα του παίκτη. Αναπαράγει το Animation HealthBarDMG και ενημερώνει το ποσοστό της μπάρας, που αντιπροσωπεύει τη ζωή του χαρακτήρα του παίκτη. Η συνάρτηση UpdatePowerText έχει για παραμέτρους μια μεταβλητή τύπου Boolean και μια μεταβλητή τύπου Text. Ανάλογα με την τιμή της Boolean θέτει το κείμενο είτε στη μεταβλητή PowerText είτε στη μεταβλητή InstructionText και στη συνέχεια αναπαράγει το κατάλληλο Animation (Εικόνα 260).



Εικόνα 260 UMG\_HUD5

Η συνάρτηση DisableQuestText αποκρύπτει το κείμενο της μεταβλητής QuestText κάνοντας το διαφανή. Η συνάρτηση EnableQuestText εμφανίζει το κείμενο της μεταβλητής QuestText κάνοντας το μη διαφανή. Οι συναρτήσεις EnableShieldCD και EnableTeleportCD κάνουν εμφανείς τις μπάρες ShieldCD και TeleportCD, που αντιπροσωπεύουν τις δεξιότητες Shield και Teleport αντίστοιχα. Οι συναρτήσεις UpdateShieldCDVisual και UpdateTeleportCooldownVisual ενημερώνουν τις μπάρες ShieldCD και TeleportCD αντίστοιχα, ώστε να οπτικοποιείται ο εναπομείναντας χρόνος μέχρι ο χαρακτήρας να ξαναποκτήσει τη δυνατότητα να χρησιμοποιήσει πάλι τις δεξιότητες αυτές (Εικόνα 261).



Εικόνα 261 UMG\_HUD6

Η συνάρτηση EnableQuest εμφανίζει τα κείμενα των μεταβλητών QuestText και CurrentQuest κάνοντας τα μη διαφανή. Η συνάρτηση UpdateQuest ενημερώνει το κείμενο της μεταβλητής QuestText. Η συνάρτηση CountdownTimer ενημερώνει το κείμενο της μεταβλητής Timer που αντιπροσωπεύει έναν μετρητή (Εικόνα 262).



Εικόνα 262 UMG\_HUD7

## Ενότητα 5.5.9 Wave\_Widget

Το Wave\_Widget έχει σκοπό την ενημέρωση του παίκτη για τα κύματα εχθρών όπως: ο χρόνος που απομένει για το επόμενο κύμα εχθρών, σε ποιο κύμα βρίσκεται και σε τι ποσοστό έχει αυτό ολοκληρωθεί.

Στην Εικόνα 263 απεικονίζεται η καρτέλα Designer του Wave\_Widget.



Εικόνα 263 Wave\_Widget1



Η συνάρτηση `ActivateUIElement_WaveInfo` κάνει εμφανή τα στοιχεία του Widget αυτού στην οθόνη του παίκτη. Η συνάρτηση `DeactivateUIElement_WaveInfo` αποκρύπτει τα στοιχεία του `BorderStats`, δηλαδή την μπάρα ολοκλήρωσης του κύματος και τον αριθμό του τωρινού κύματος. Η συνάρτηση `ActivateUIElement_CountdownTimer` αρχικοποιεί τον μετρητή για την έναρξη του επόμενου κύματος και εμφανίζει τον μετρητή. Η συνάρτηση `DeactivateUIElement_CountdownTimer` αποκρύπτει τον μετρητή. Οι συνάρτησεις `UpdateUIElement_CountdownTimer` και `UpdateUIElement_WaveProgress` ενημερώνουν τον μετρητή και την μπάρα αντίστοιχα (Εικόνα 264).



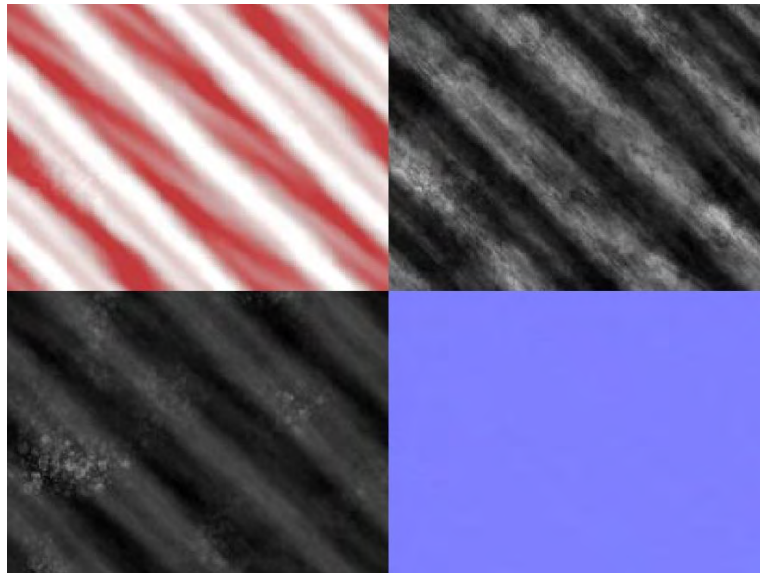
Εικόνα 264 Wave\_Widget2



## 5.6 Textures

---

Οι παρακάτω υφές (Εικόνα 265) χρησιμοποιήθηκαν εντός του παιχνιδιού στο Mat\_Candy και Mat\_CandyDisin.



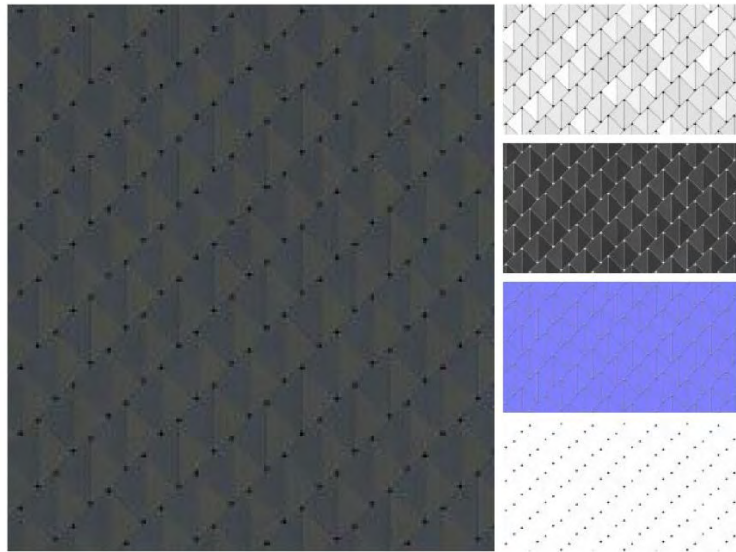
Εικόνα 265 Υφές Candy

Οι υφές στην Εικόνα 266 χρησιμοποιήθηκαν εντός του παιχνιδιού στο Mat\_MetalPlate1.



Εικόνα 266 Υφές MetalPlates005

Οι υφές στην Εικόνα 267 χρησιμοποιήθηκαν εντός του παιχνιδιού στο Mat\_MetalPlate.

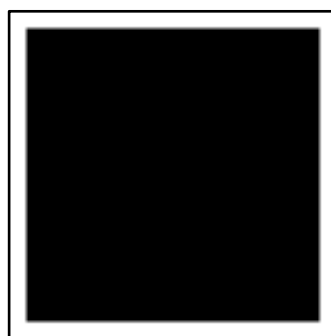


Εικόνα 267 Υφές MetalPlates006

Όλες οι παραπάνω υφές ανακτήθηκαν από τη σελίδα: <https://cc0textures.com/> και διατίθενται με άδεια Creative Commons CC0 License.

Οι υφές: T\_decoLinesB, T\_fire01, T\_flare\_a, T\_gradiation, T\_maskRGB3, T\_maskRGB5, T\_noise13\_sml\_nor, T\_noise5, T\_noise6, και T\_slash01\_4x4 χρησιμοποιήθηκαν για το Particle System P\_stabShot (Kakky, 2017).

Η υφή στην Εικόνα 268 δημιουργήθηκε στο δωρεάν λογισμικό Krita και χρησιμοποιήθηκε στο υλικό M\_Grid.



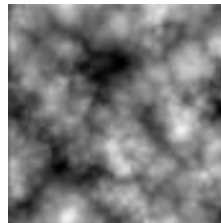
Εικόνα 268 Υφή T\_GridCell

Η υφή στην Εικόνα 269 είναι gradient και δημιουργήθηκε στο δωρεάν λογισμικό Krita και χρησιμοποιήθηκε στο υλικό Mat\_ShieldFinal και τα παρόμοιά του.



Εικόνα 269 Υφή T\_ScanLineMask

Η υφή T\_TilingNoise03 περιλαμβάνεται στο Engine Content της Unreal (Εικόνα 270).



Εικόνα 270 T\_TilingNoise03

## 5.7 Materials

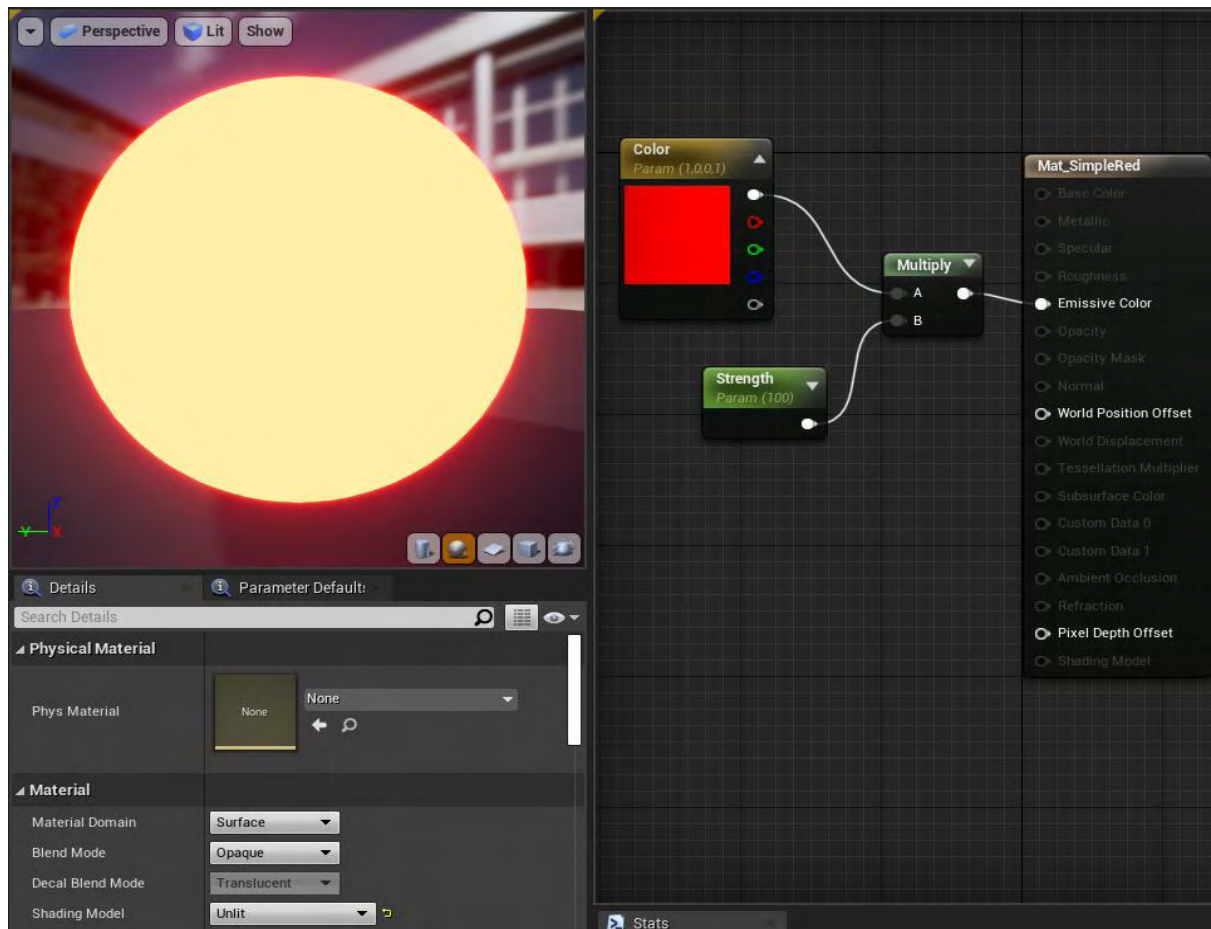
---

### Ενότητα 5.7.1 Mat\_SimpleRed

---

Τα υλικά αποτελούν ένα μεγάλο κομμάτι της οπτικής όψης ενός παιχνιδιού σε συνδυασμό με τα τρισδιάστατα μοντέλα.

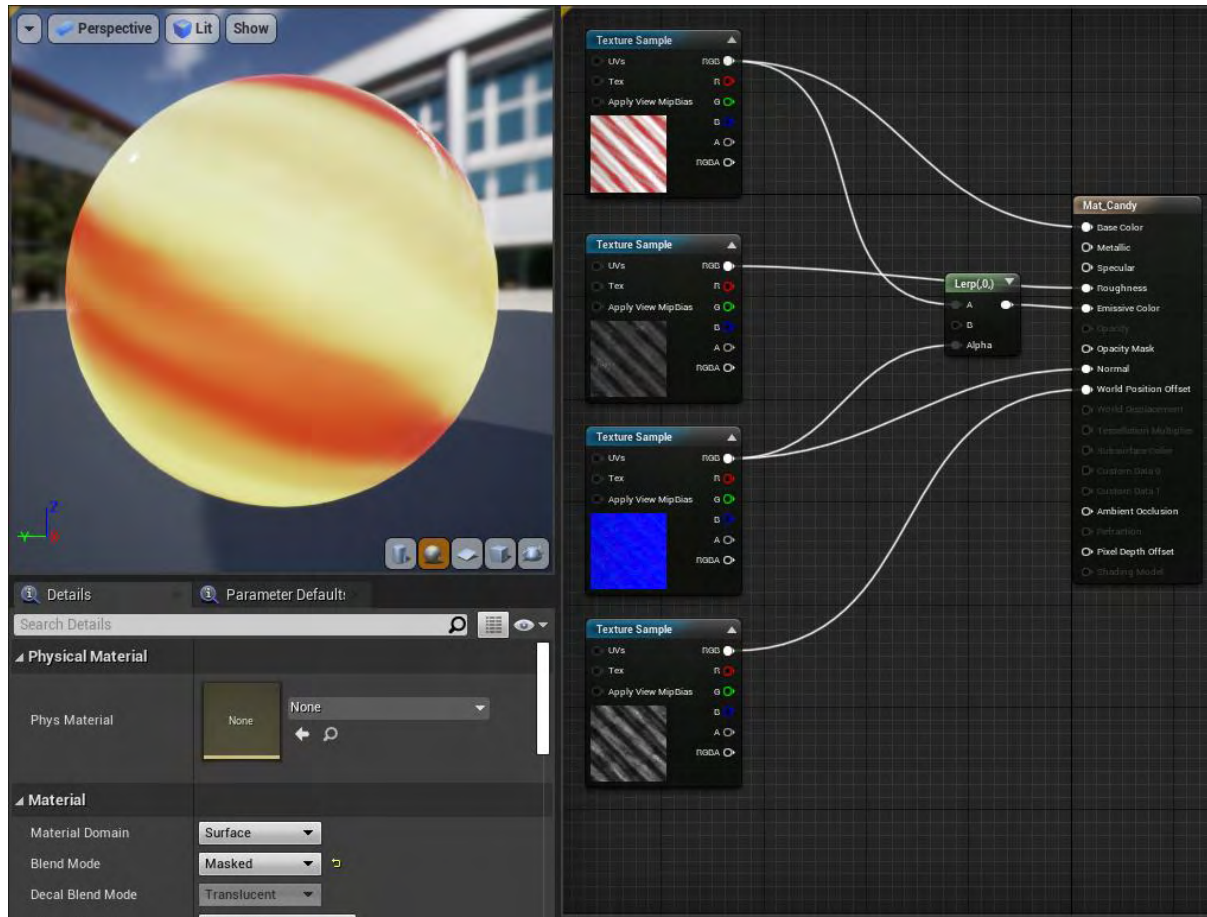
Το υλικό Mat\_SimpleRed εκπέμπει φως συγκεκριμένου χρώματος που καθορίζεται από την παράμετρο Color και έντασης που καθορίζεται από την παράμετρο Strength (Εικόνα 271). Τα υλικά MI\_SimpleGreen, MI\_SimpleEmissive, MI\_SimpleGreenLE, MI\_SimpleLowEmissive, MI\_SimpleVioletLE και MI\_SimpleYellow έχουν γονέα αυτό το υλικό. Κάποια από αυτά απεικονίζονται στο παράρτημα Α.



Εικόνα 271 Mat\_SimpleRed

## Ενότητα 5.7.2 Mat\_Candy

Το υλικό Mat\_Candy ( Εικόνα 272) χρησιμοποιεί τις υφές της Εικόνας 265.

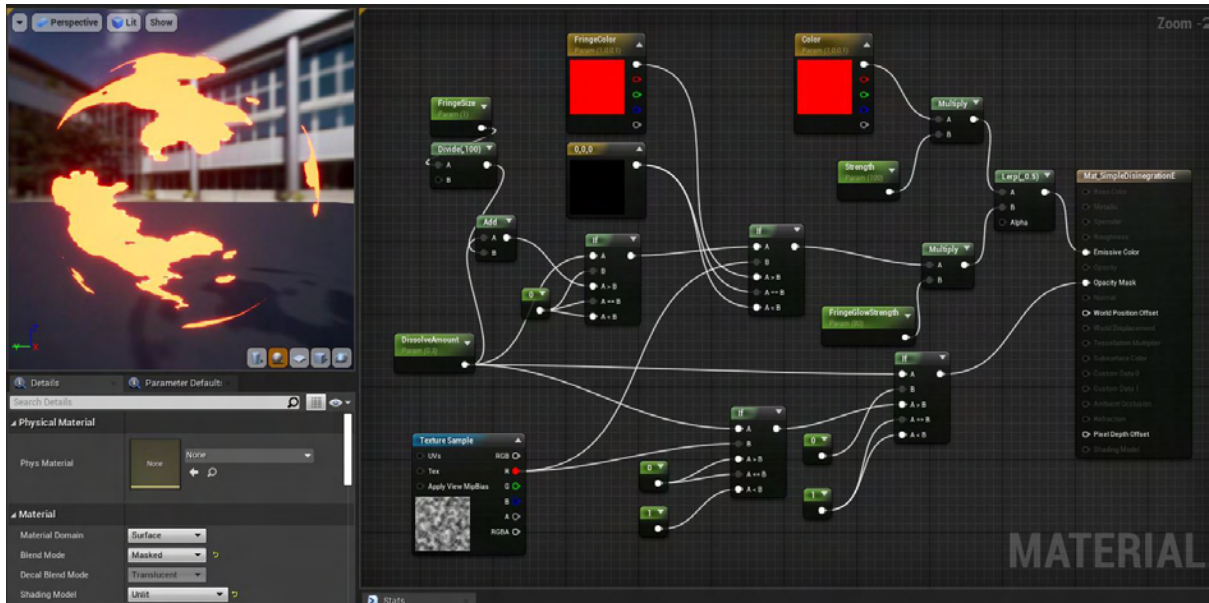


Εικόνα 272 Mat\_Candy

## Ενότητα 5.7.3 Mat\_SimpleDisinegrationE

Το Mat\_SimpleDisinegrationE (Εικόνα 273) είναι ένα υλικό το οποίο προκαλεί την ψευδαισθηση σύνθεσης ή αποσύνθεσης. Το υλικό αυτό κατασκευάστηκε σύμφωνα με τις οδηγίες του διδακτικού υλικού «Disintegrating Baddies: dissolve effect in UE4» (Lamarche, 2015). Τα υλικά MI\_DisinRed, MI\_SimpleBlueDisinegrationE, MI\_SimpleDisinVioletLE έχουν γονέα το υλικό αυτό και κάποια από αυτά απεικονίζονται στο παράρτημα Α.

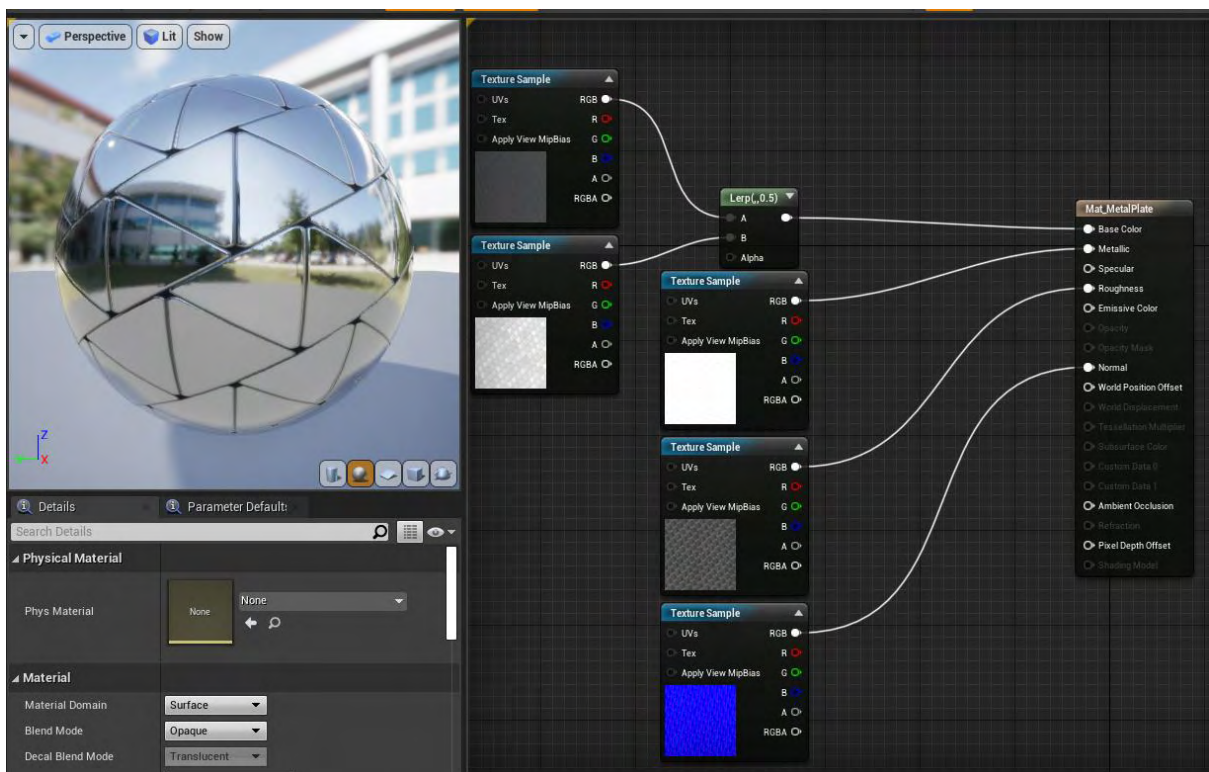




Εικόνα 273 Mat\_SimpleDisinegrationE

## Ενότητα 5.7.4 Mat\_MetalPlate

Το υλικό Mat\_MetalPlate (Εικόνα 274) χρησιμοποιεί τις υφές που απεικονίζονται στην Εικόνα 267.

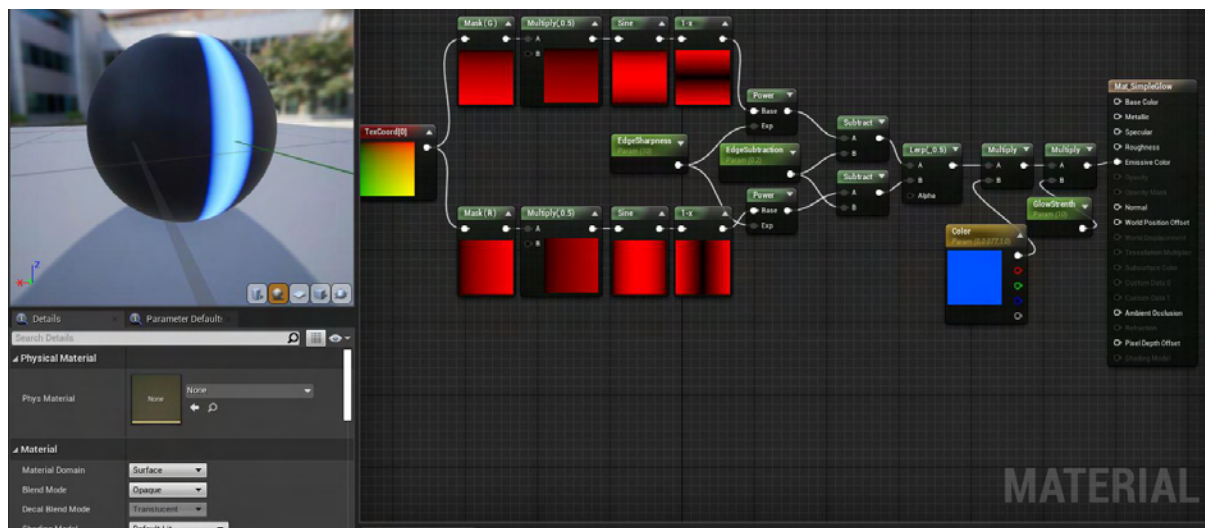


Εικόνα 274 Mat\_MetalPlate

## Ενότητα 5.7.5 Mat\_SimpleGlow

Το Mat\_SimpleGlow (Εικόνα 275) είναι ένα αδιαφανές υλικό επιφάνειας. Δημιουργήθηκε με τη βοήθεια του διδακτικού βίντεο «Unreal 4 Tutorial: Advanced tron material #1» (Liaret, 2015). Τα ακόλουθα υλικά έχουν γονέα το υλικό αυτό:

- MI\_SimpleGlowBlueHigh
- MI\_SimpleGlowGreenHigh
- MI\_SimpleGlowRoomBlue
- MI\_SimpleGlowRoomGreen
- MI\_SimpleGlowRoomOrange
- MI\_SimpleGlowRoomPink
- MI\_SimpleGlowRoomRed
- MI\_SimpleGlowOrangeHigh
- MI\_TileGreen.

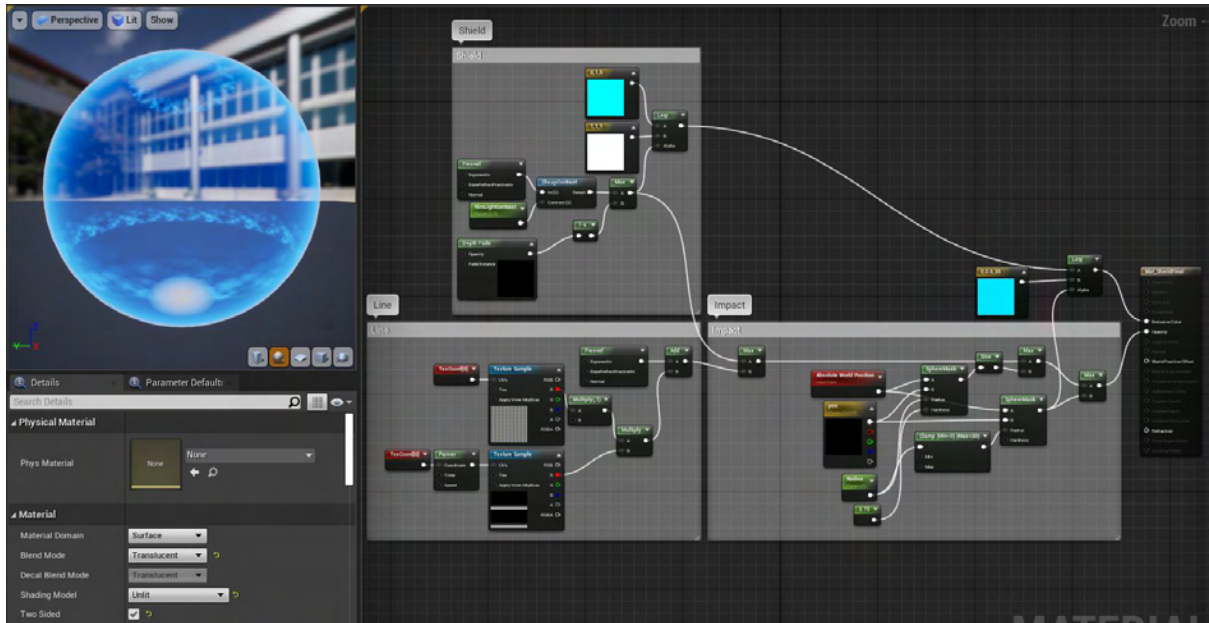


Εικόνα 275 Mat\_SimpleGlow

## Ενότητα 5.7.6 Mat\_ShieldFinal

Το Mat\_ShieldFinal αποτελεί το υλικό της ασπίδας του παίκτη και δημιουργήθηκε με τη βοήθεια του διδακτικού βίντεο «Force Field and Impact Effect - (UE4 Tutorial)» (UnrealCG, 2017). Το υλικό Mat\_BlockShieldFinal είναι παρόμοιο με το υλικό αυτό και τα MI\_BlockTPShieldFinal και MI\_DoorCode είναι παιδιά του. Εικόνες για τα υλικά αυτά υπάρχουν στο παράρτημα Α.

Στην Εικόνα 276 απεικονίζεται καθολικά το υλικό Mat\_ShieldFinal και στις επόμενες εικόνες (Εικόνα 277 – 279) απεικονίζεται κάθε κύριο κομμάτι του υλικού αυτού ξεχωριστά.

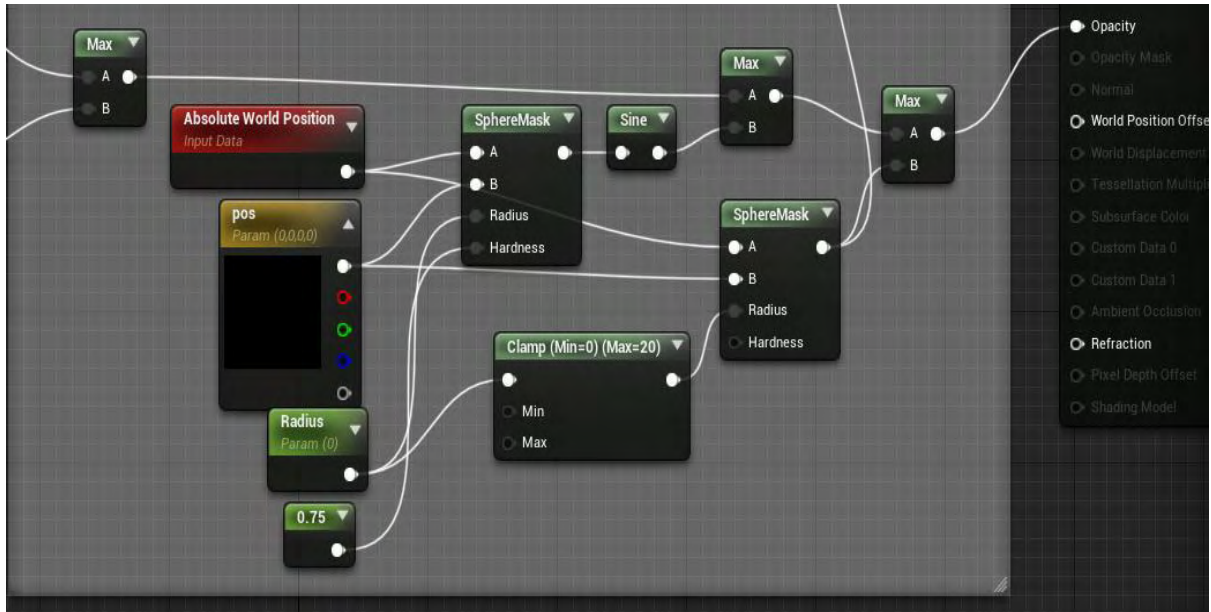


Εικόνα 276 Mat\_ShieldFinal part1

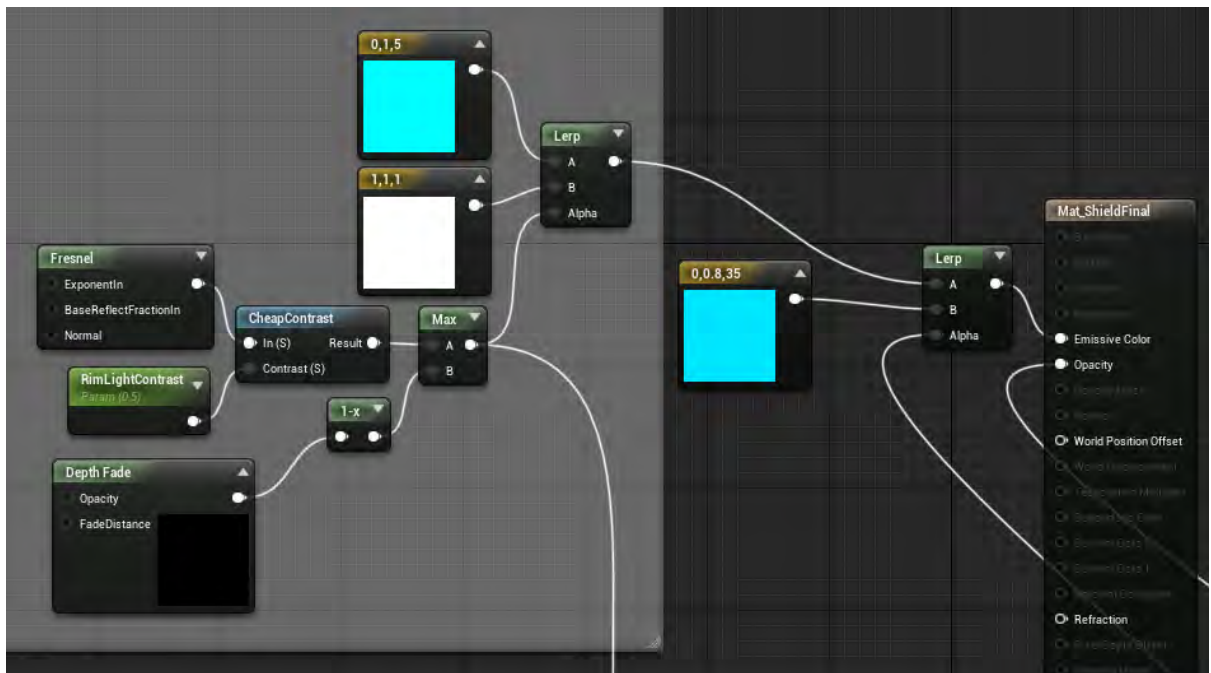


Εικόνα 277 Mat\_ShieldFinal part 2





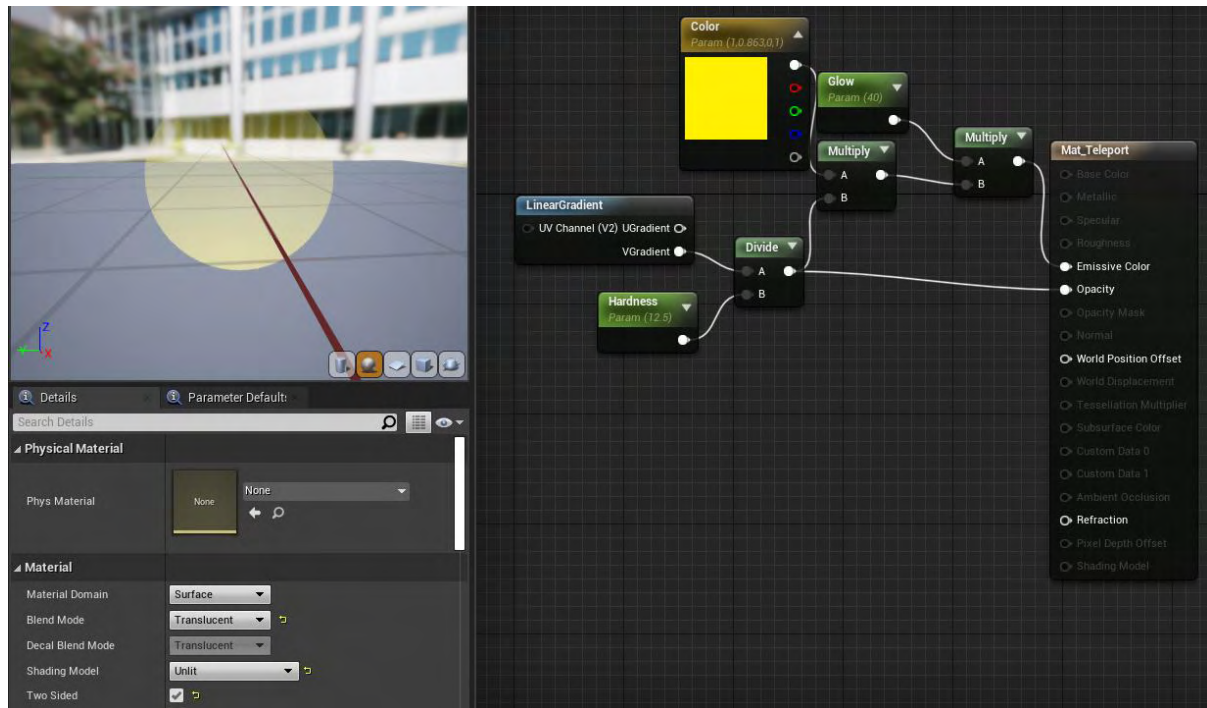
Εικόνα 278 Mat\_ShieldFinal part 3



Εικόνα 279 Mat\_ShieldFinal part 4

## Ενότητα 5.7.7 Mat\_Teleport

Το Mat\_Teleport είναι ένα αυτόφωτο ημιδιαφανή υλικό επιφάνειας. Το υλικό MI\_Teleport είναι παιδί του υλικού αυτού (Εικόνα 280).

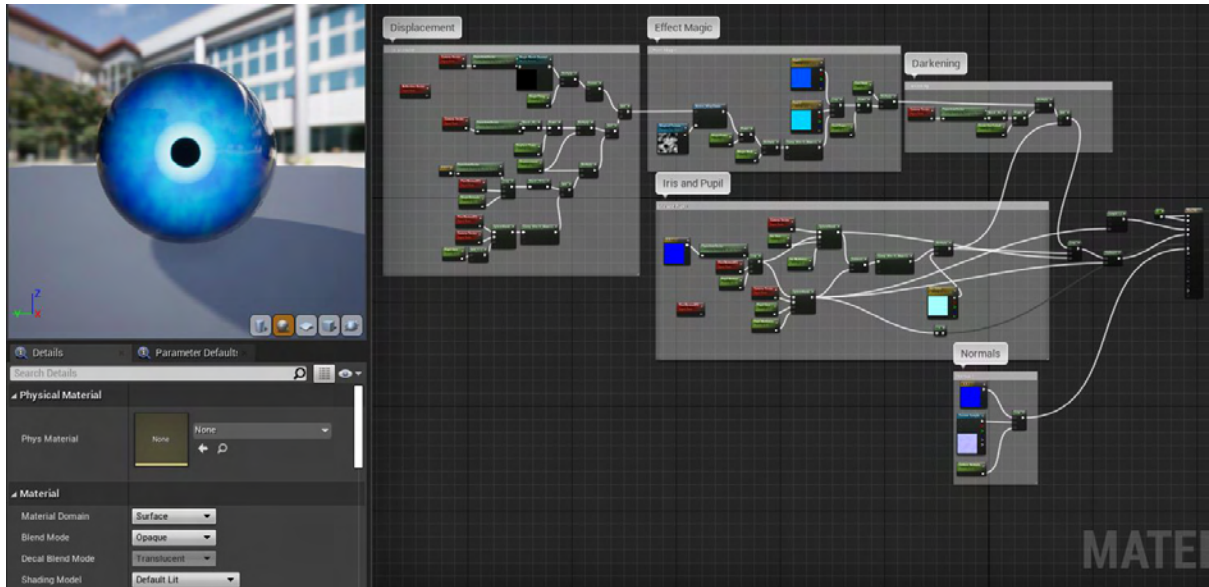


Εικόνα 280 Mat\_Teleport

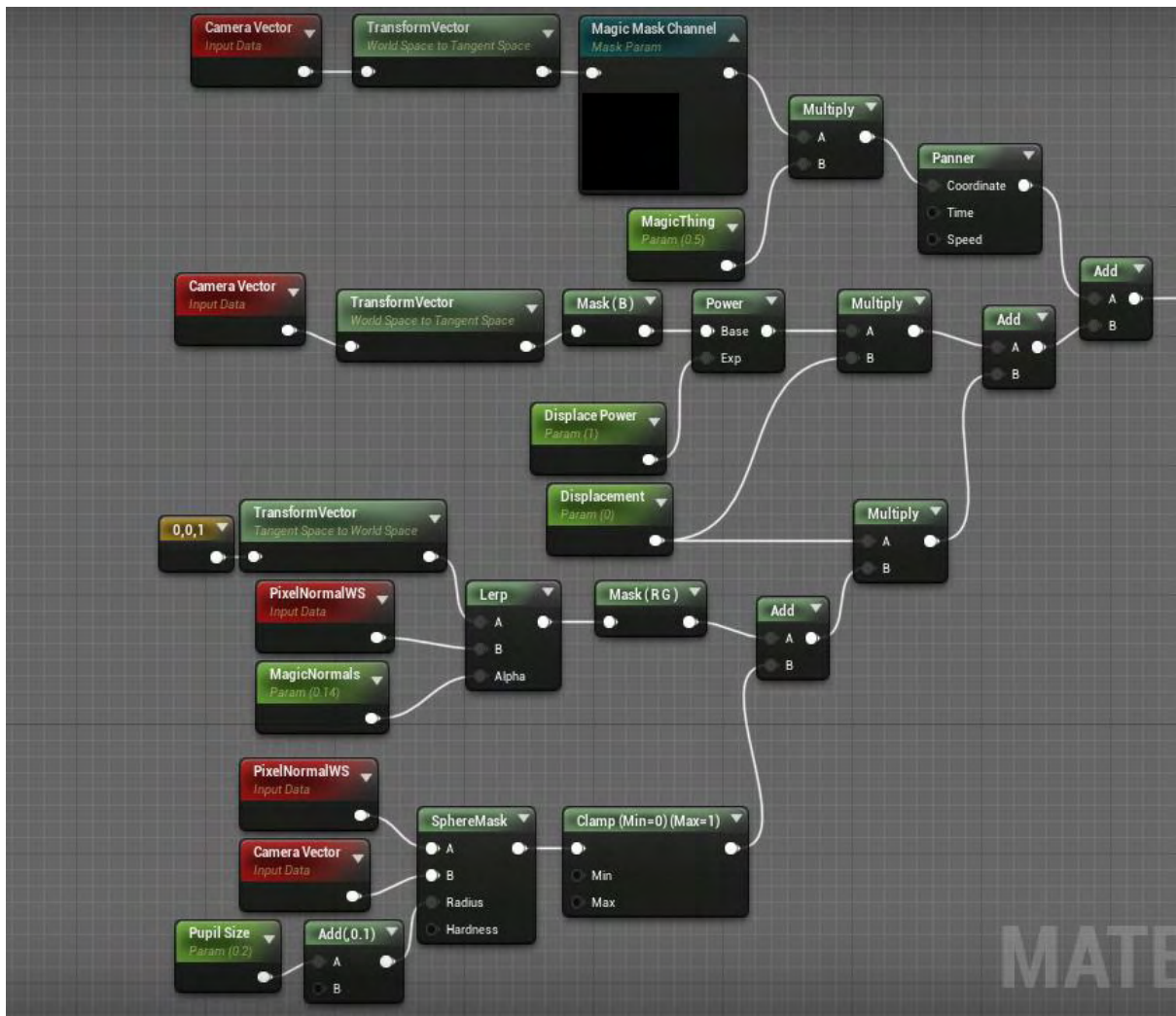
## Ενότητα 5.7.8 Mat\_Orb

Το υλικό Mat\_Orb (Εικόνες 281 – 285) δημιουργήθηκε με τη βοήθεια του διδακτικού βίντεο «Shaders in UE4|Live Training|Unreal Engine Livestream» (Unreal Engine, n.d.). Τα υλικά MI\_EvilOrb και MI\_Orb είναι παιδιά αυτού.

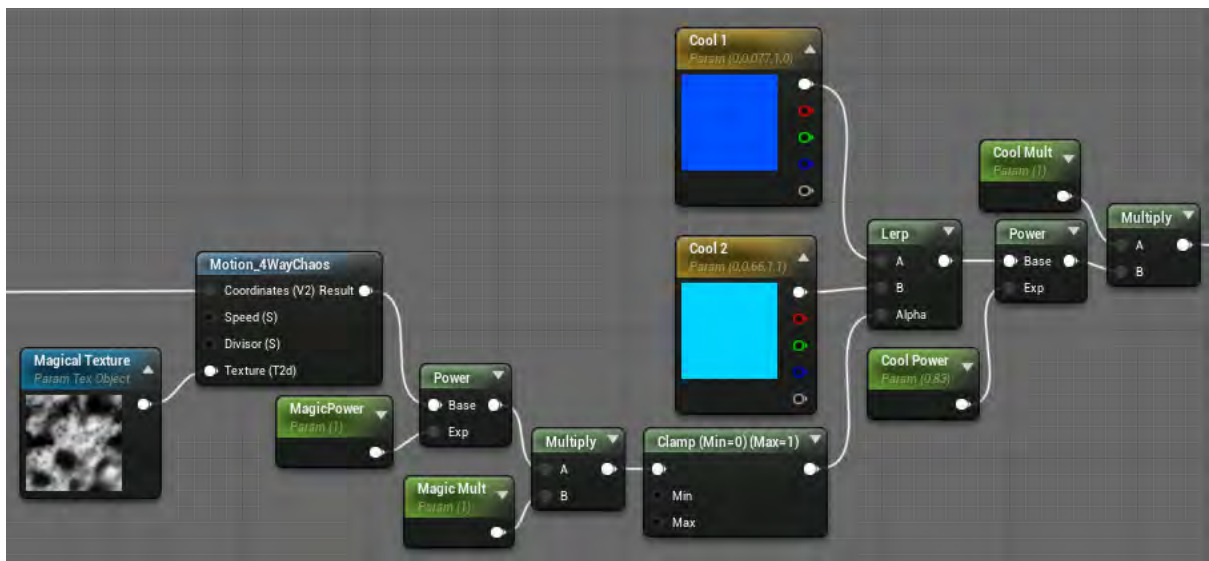




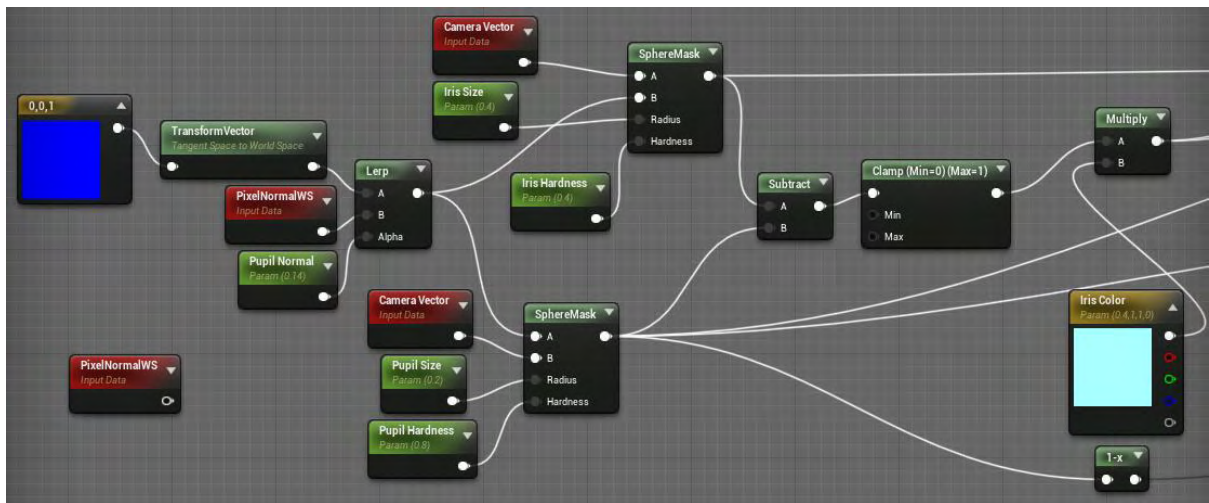
Εικόνα 281 Mat\_Orb part 1



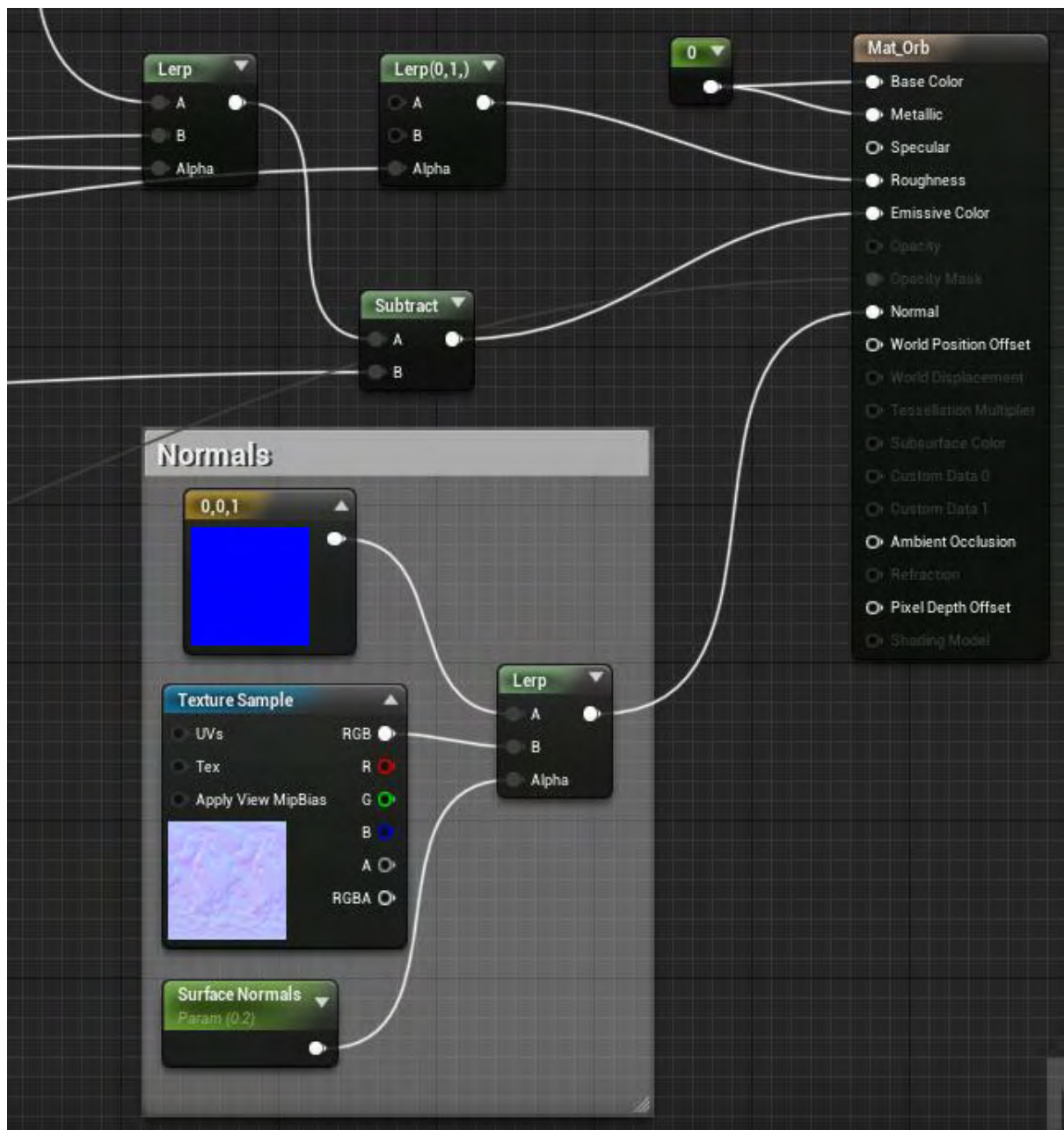
Εικόνα 282 Mat\_Orb part 2



Εικόνα 283 Mat\_Orb part 3



Εικόνα 284 Mat\_Orb part 4



Εικόνα 285 Mat\_Orb part 5

## 5.8 Audio

Οι ήχοι EnemySpawnSoundCropped (aghony, 2019), AlarmSound (Jm\_at, 2018) και DramaticSoundCropped (x86cam, 2013) ανακτήθηκαν από τον ιστότοπο Freesound με άδεια Creative Commons 0 License.



Οι ήχοι FlyingDiscSoundCropped (Development, 2019) και ScifiBackgroundCropped (5pr1n6, 2019) ανακτήθηκαν από τον ιστότοπο OpenGameArt.Org με άδεια Creative Commons 0 License.

Οι φωνές που χρησιμοποιήθηκαν, δημιουργήθηκαν με το δωρεάν online εργαλείο Voice Generator από την ιστοσελίδα voicegenerator.io και τροποποιήθηκαν με το δωρεάν εργαλείο Voice Changer από την ιστοσελίδα voicechanger.io.

Οι υπόλοιποι ήχοι είναι από τους προϋπάρχοντες της Unreal Engine.

## 5.9 Maps

---

### Ενότητα 5.9.1 MainMenu

---

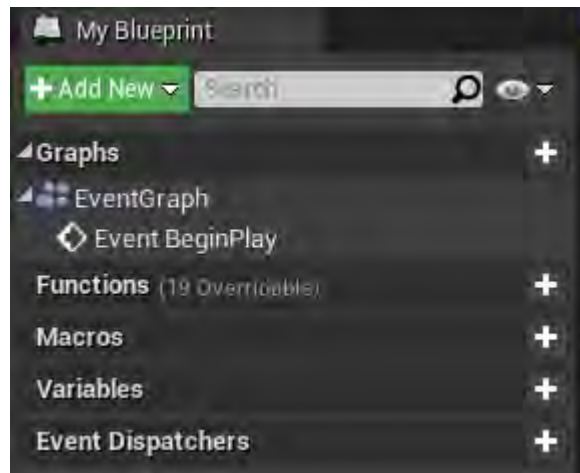
Το επίπεδο με την ονομασία MainMenu είναι το πρώτο επίπεδο του παιχνιδιού με το οποίο έρχεται σε επαφή ο παίκτης. Στο επίπεδο αυτό δίνονται διάφορες επιλογές στον παίκτη, με την πιο κύρια να είναι το ξεκίνημα του παιχνιδιού.

Στην Εικόνα 286 απεικονίζονται οι ρυθμίσεις του επιπέδου στην καρτέλα World Settings.



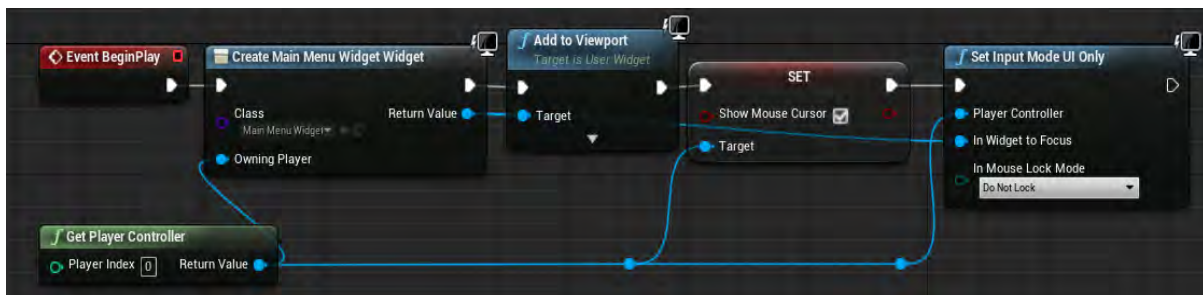
Εικόνα 286 MainMenu World Settings

Στην Εικόνα 286 απεικονίζονται οι συναρτήσεις και οι μεταβλητές που χρησιμοποιήθηκαν στο Level Blueprint του επιπέδου.



Εικόνα 287 MainMenu My Blueprint Tab

Το Event BeginPlay ενεργοποιείται μόλις δημιουργηθεί το επίπεδο και η λειτουργία που υλοποιεί είναι η πρόσθεση του MainMenu\_Widget στην οθόνη του παίκτη. Στη συνέχεια κάνει ορατό τον κέρσορα του ποντικιού και θέτει το Input Mode σε UI Only (Εικόνα 288).



Εικόνα 288 MainMenu Level Blueprint

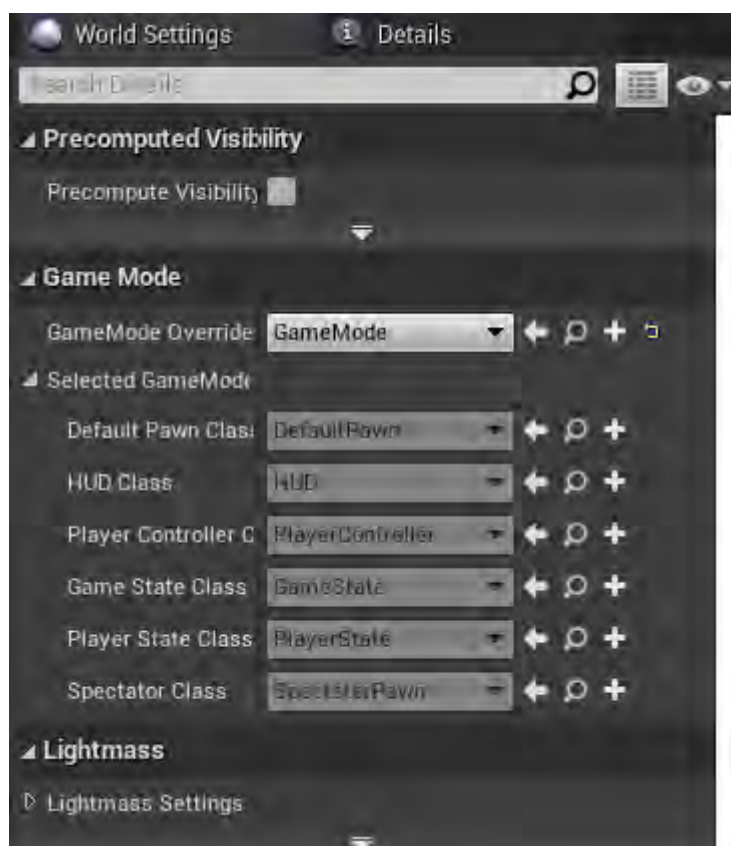


## Ενότητα 5.9.2 Level1Story

---

Σκοπός του επιπέδου με την ονομασία Level1Story, όπως και των υπολοίπων με παρόμοιο τίτλο είναι να προσθέσουν την αφήγηση κατά την διάρκεια του παιχνιδιού.

Στην Εικόνα 289 απεικονίζονται οι ρυθμίσεις του επιπέδου στην καρτέλα World Settings.



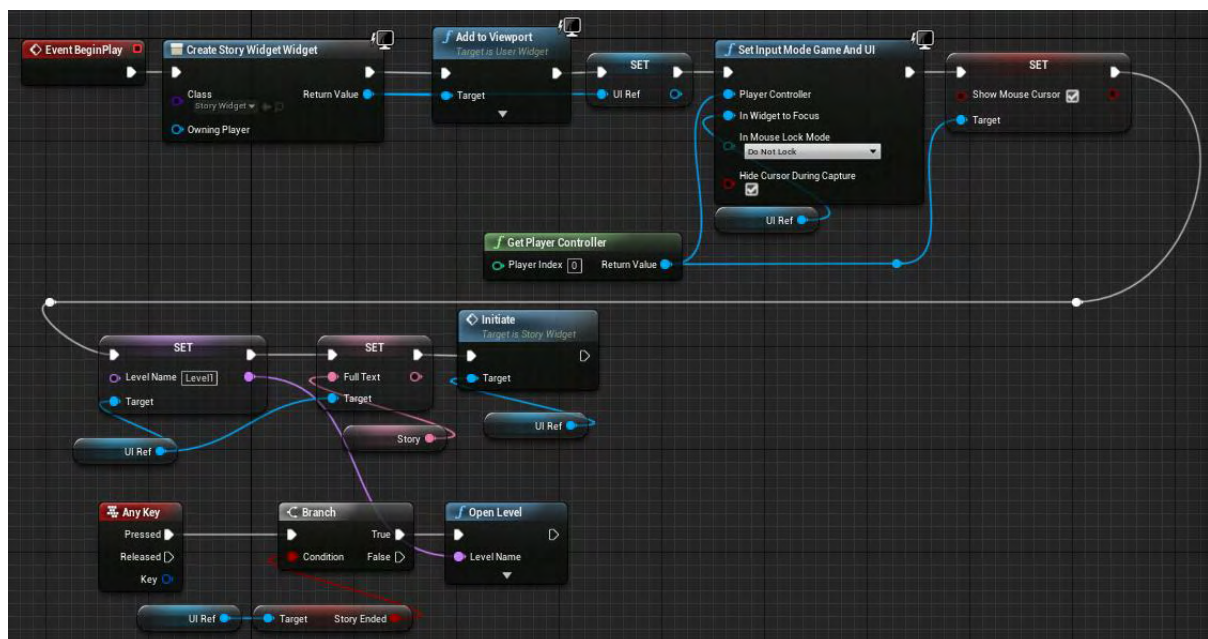
Εικόνα 289 Level1Story World Settings

Στην Εικόνα 290 απεικονίζονται οι συναρτήσεις και οι μεταβλητές που χρησιμοποιήθηκαν στο Level Blueprint του επιπέδου.



Εικόνα 290 Level1Story My Blueprint Tab

Το Event BeginPlay ενεργοποιείται μόλις δημιουργηθεί το επίπεδο και η λειτουργία που υλοποιεί είναι η πρόσθεση του Story\_Widget στην οθόνη του παίκτη. Στη συνέχεια κάνει ορατό τον κέρσορα του ποντικιού και θέτει το Input Mode σε Game and UI. Έπειτα ξεκινάει η αφήγηση της πλοκής. Αφού ολοκληρωθεί, μόλις ο παίκτης πατήσει οποιοδήποτε πλήκτρο, φορτώνεται το επίπεδο με την ονομασία Level1 (Εικόνα 291).



Εικόνα 291 Level1Story Level Blueprint

## Ενότητα 5.9.3 Level1

Το επίπεδο με την ονομασία Level1 έχει σκοπό την επιδείξη των δεξιοτήτων του χαρακτήρα στον παίκτη.

Στην Εικόνα 292 απεικονίζονται οι ρυθμίσεις του επιπέδου στην καρτέλα World Settings. Αξίζει να αναφερθεί ότι οι ρυθμίσεις του Lightmass έχουν τροποποιηθεί με σκοπό το επιθυμητό οπτικό αποτέλεσμα του φωτισμού.



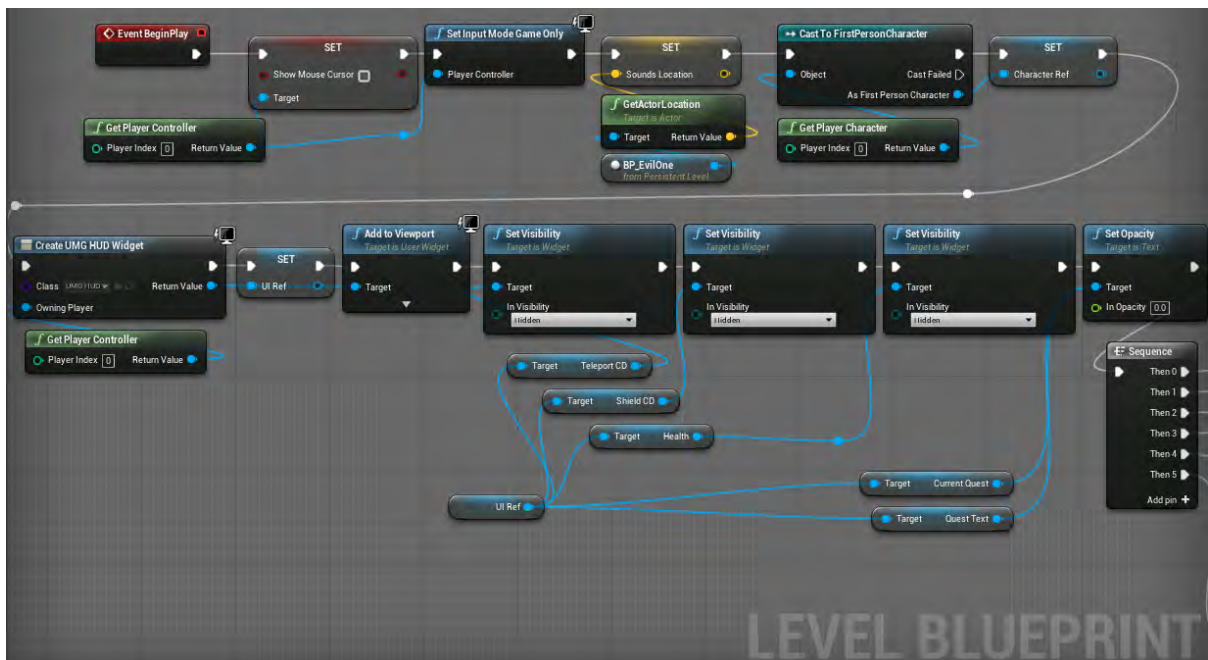
Εικόνα 292 Level1 World Settings

Στην Εικόνα 293 απεικονίζονται οι συναρτήσεις και οι μεταβλητές που χρησιμοποιήθηκαν στο Level Blueprint του επιπέδου.



Εικόνα 293 Level1 My Blueprint Tab

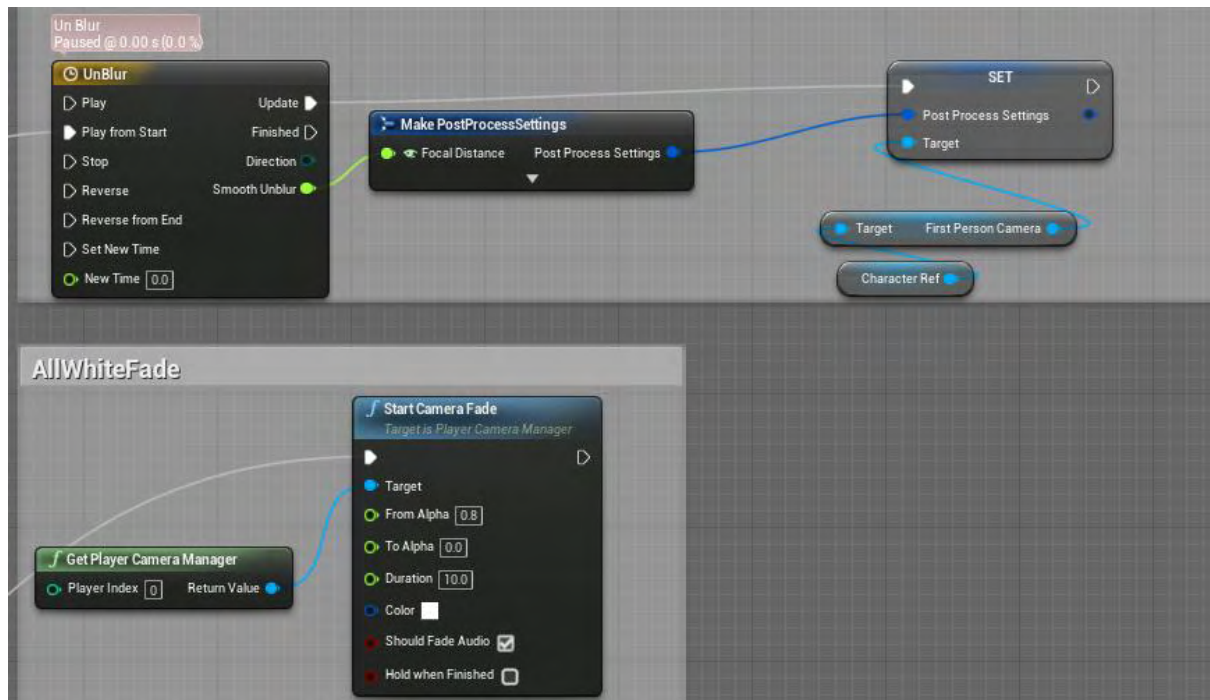
Το Event BeginPlay ενεργοποιείται μόλις δημιουργηθεί το επίπεδο και η λειτουργία που υλοποιεί είναι η απόκρυψη του κέρσορα του ποντικιού και θέτεται το Input Mode σε Game Only. Στη συνέχεια αποθηκεύεται η θέση του actor BP\_EvilOne στη μεταβλητή τύπου Vector SoundsLocation. Ακολούθως, δημιουργείται αναφορά στον χαρακτήρα του παίκτη, που αποθηκεύεται στη μεταβλητή CharacterRef. Κατόπιν δημιουργείται Widget τύπου UMG\_HUD, προστίθεται στην οθόνη του παίκτη και αποκρύπτονται τα στοιχεία TeleportCD, ShieldCD, Health, CurrentQuest και QuestText. Έπειτα ξεκινούν έξι σειριακές ακολουθίες και εκτελούνται (Εικόνα 294).



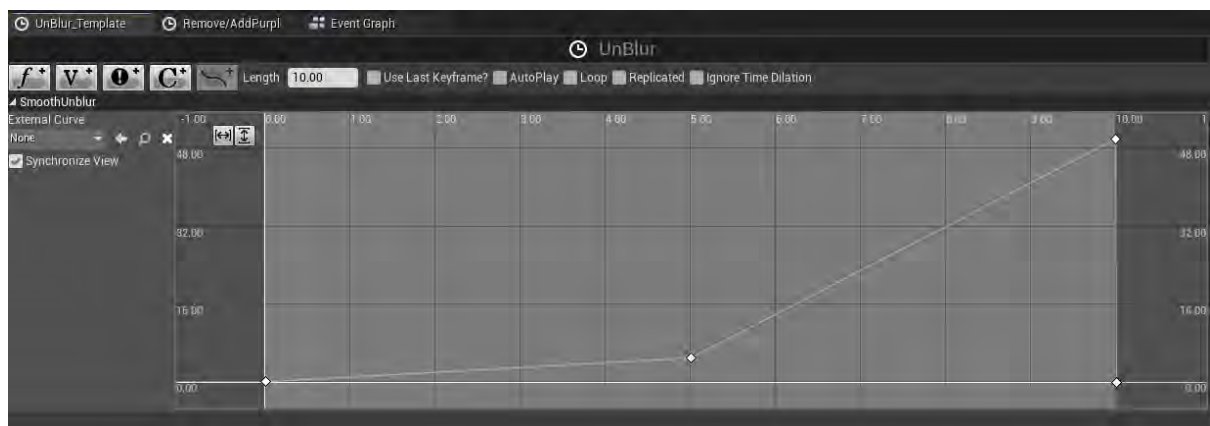
Εικόνα 294 Level1 Level Blueprint part 1

Η πρώτη και η δεύτερη ακολουθία από τα μονοπάτια 0 και 1 δημιουργούν οπτικό εφέ τροποποιώντας τιμές μεταβλητών από την κάμερα του παίκτη (Εικόνα 295).





Εικόνα 295 Level1 Level Blueprint part 2



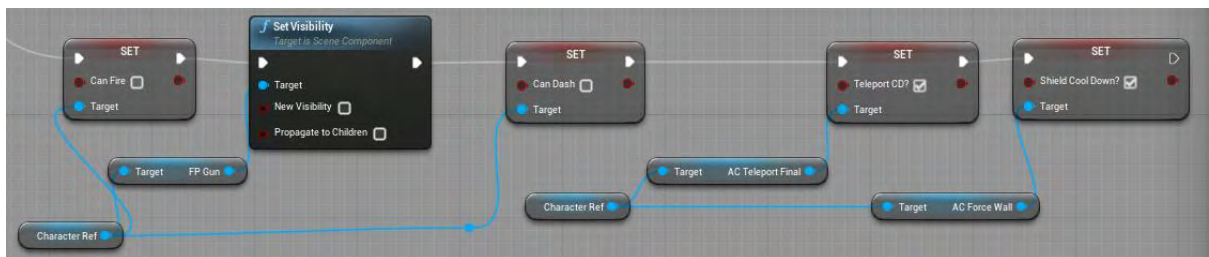
Εικόνα 296 UnBlur

Η τρίτη ακολουθία από το μονοπάτι 2 απενεργοποιεί προσωρινά την ικανότητα χαρακτήρα του παίκτη να κινηθεί. Στη συνέχεια αναπαράγονται διάφοροι ήχοι, ενεργοποιείται η δυνατότητα του παίκτη να κινηθεί. Επίσης γίνεται ορατό και ενημερώνεται το κείμενο της αποστολής (Εικόνα 297).



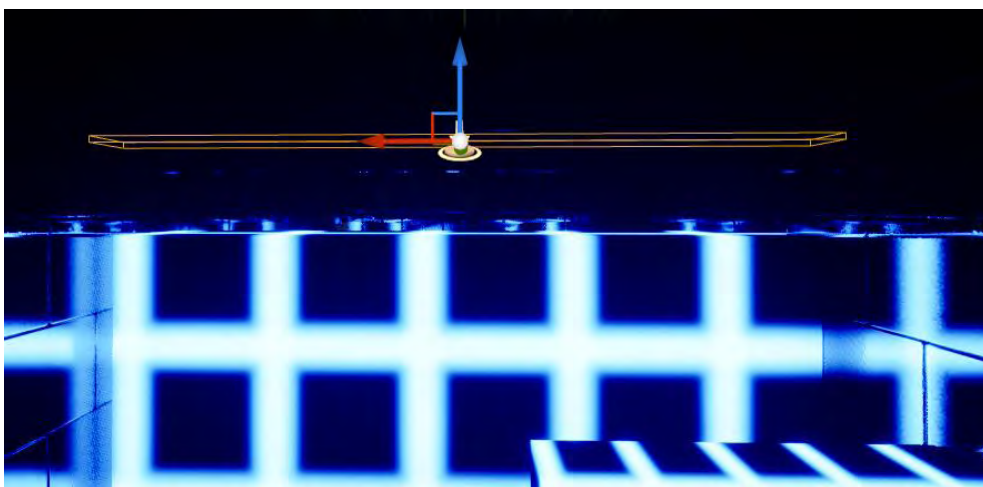
Εικόνα 297 Level1 Level Blueprint part 3

Η τέταρτη ακολουθία από το μονοπάτι 3 απενεργοποιεί όλες τις δεξιότητες του χαρακτήρα του παίκτη (Εικόνα 298).



Εικόνα 298 Level1 Level Blueprint part 4

Στην εικόνα 299 απεικονίζεται ο actor DashTrigger μέσα στο επίπεδο.



Εικόνα 299 Level1 DashTrigger

Στην Εικόνα 300 και στην Εικόνα 301 απεικονίζεται η λειτουργικότητα του actor DashTrigger. Μόλις ο χαρακτήρας του παίκτη συγκρουστεί με το πλέγμα σύγκρουσης DashTrigger, ενεργοποιείται το OnActorBeginOverlap του actor. Στην πρώτη σύγκρουση ενημερώνεται το κείμενο της αποστολής, αναπαράγονται οι κατάλληλοι ήχοι και αυξάνεται η τιμή της μεταβλητής TimesTriedCounter. Στην επόμενη σύγκρουση ενημερώνεται πάλι το κείμενο της αποστολής, αναπαράγεται κατάλληλος ήχος και αυξάνεται η τιμή της μεταβλητής TimesTriedCounter. Στην τρίτη και τελευταία σύγκρουση ενημερώνεται το κείμενο της αποστολής, αναπαράγεται ο επιλεγμένος ήχος και καταστρέφεται ο actor αυτός.



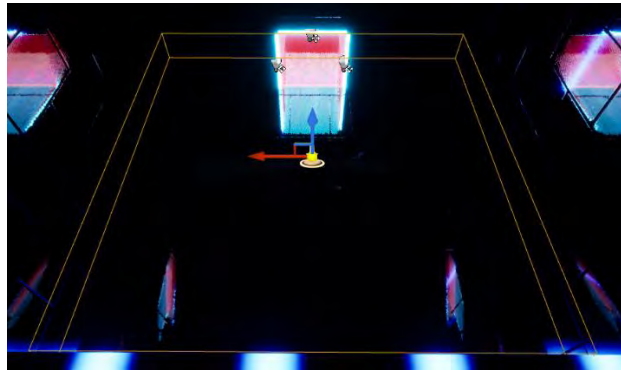
Εικόνα 300 Level1 Level Blueprint part 5



Εικόνα 301 Level1 Level Blueprint part 6



Στην εικόνα 302 απεικονίζεται ο actor OrbGivingTrigger μέσα στο επίπεδο.



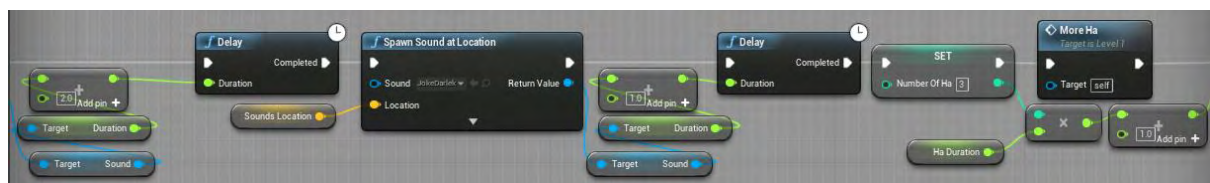
Εικόνα 302 OrbGivingTrigger

Μόλις ο χαρακτήρας του παίκτη συγκρουστεί με τον actor OrbGivingTrigger ενημερώνεται το κείμενο της αποστολής και αναπαράγεται ο επιλεγμένος ήχος (Εικόνα 303).



Εικόνα 303 Level1 Level Blueprint part 7

Στη συνέχεια αναπαράγονται οι επιλεγμένοι ήχοι χρησιμοποιώντας και την συνάρτηση MoreHa (Εικόνα 304).



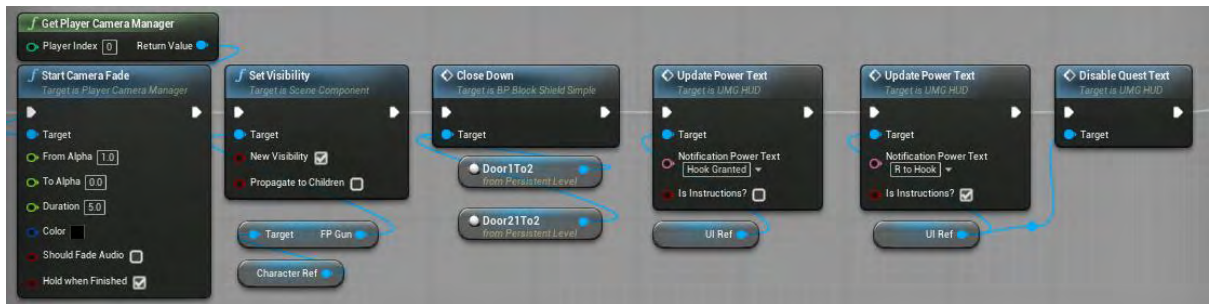
Εικόνα 304 Level1 Level Blueprint part 8

Έπειτα αναπαράγεται πάλι επιλεγμένος ήχος και απενεργοποιείται η δυνατότητα Input για τον παίκτη. Ακολουθεί οπτικό εφέ χρησιμοποιώντας τη συνάρτηση Start Camera Fade της Unreal και μετά από χρονοκαθυστέρηση πέντε δευτερολέπτων ενεργοποιείται η δυνατότητα Input του παίκτη (Εικόνα 305).



Εικόνα 305 Level1 Level Blueprint part 9

Συνεχίζοντας δημιουργείται και άλλο οπτικό εφέ χρησιμοποιώντας τη συνάρτηση Start Camera Fade και γίνεται πλέον εμφανές το όπλο του παίκτη. Χρησιμοποιώντας τη συνάρτηση CloseDown του BP\_BlockShieldSimple εξαφανίζονται οι πόρτες που εμποδίζουν το μονοπάτι και ενημερώνεται το κείμενο της αποστολής (Εικόνα 306).



Εικόνα 306 Level1 Level Blueprint part 10

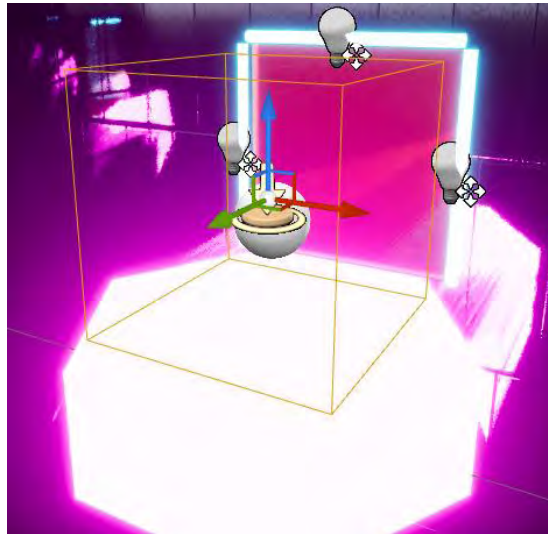
Τέλος, ενημερώνεται η τοποθεσία του actor BP\_EvilOne και η τιμή της μεταβλητής SoundsLocation (Εικόνα 307).



Εικόνα 307 Level1 Level Blueprint part 11

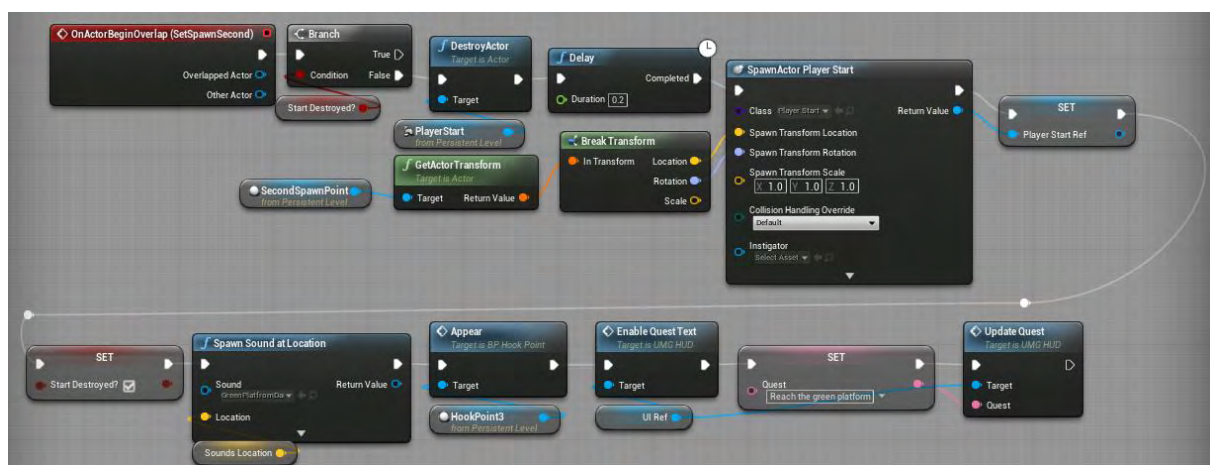


Στην εικόνα 308 απεικονίζεται ο actor SetSpawnSecond μέσα στο επίπεδο.



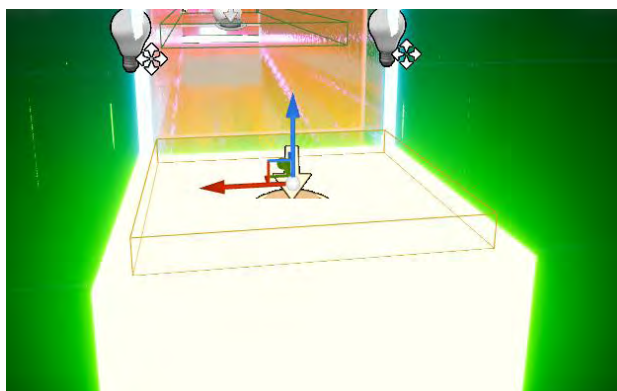
Εικόνα 308 SetSpawnSecond

Μόλις ο χαρακτήρας του παίκτη συγκρουστεί με το πλέγμα σύγκρουσης SetSpawnSecond, ενεργοποιείται το OnActorBeginOverlap του actor. Έτσι σηματοδοτείται η καταστροφή του actor PlayerStart και δημιουργείται νέο PlayerStart στην τοποθεσία του actor SetSecondSpawn, ώστε, όταν ο χαρακτήρας του παίκτη καταστρέφεται, να αναδημιουργείται σε νέα τοποθεσία. Στη συνέχεια εμφανίζεται ο actor HookPoint3 με τη χρήση της συνάρτησης Appear του BP\_HookPoint, στον οποίο θα χρησιμοποιήσει ο παίκτης τη δεξιότητα Hook για να περάσει το εμπόδιο. Ενημερώνεται το κείμενο της αποστολής (Εικόνα 309).



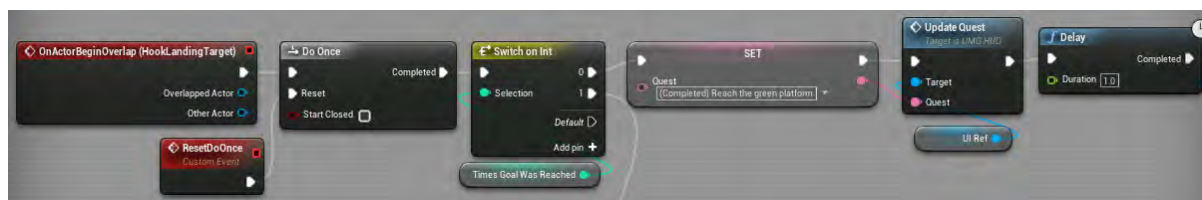
Εικόνα 309 Level1 Level Blueprint part 12

Στην εικόνα 310 απεικονίζεται ο actor HookLandingTarget μέσα στο επίπεδο.



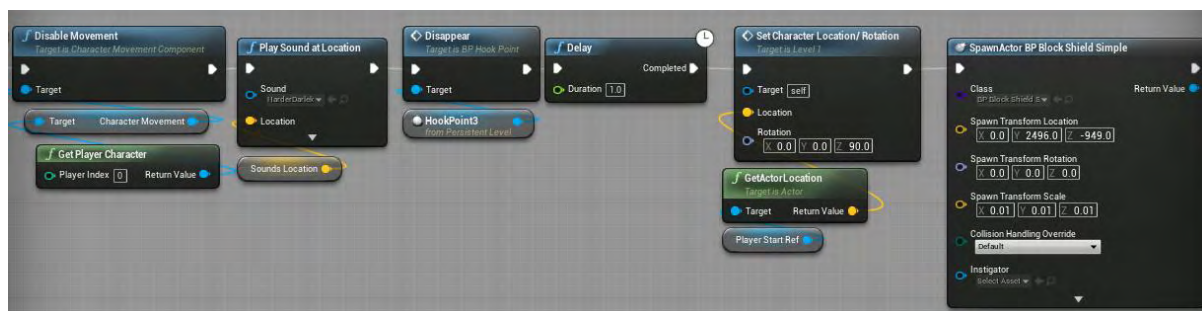
Εικόνα 310 HookLandingTarget

Μόλις ο χαρακτήρας του παίκτη συγκρουστεί την πρώτη φορά με τον actor HookLandingTarget, ενημερώνεται το κείμενο της αποστολής (Εικόνα 311).



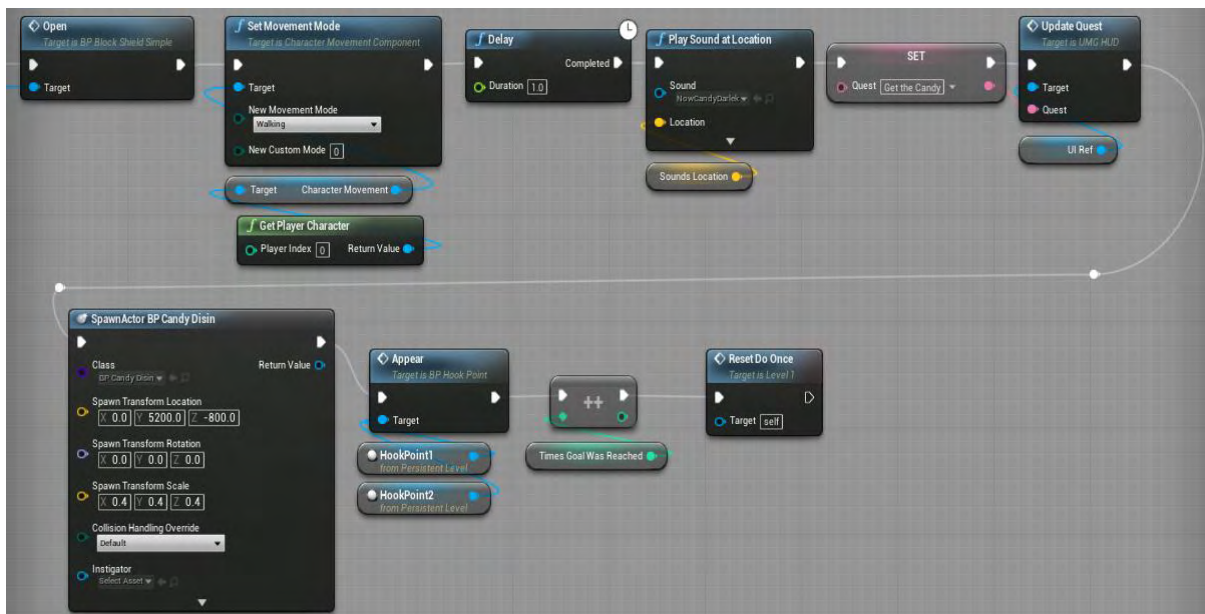
Εικόνα 311 Level1 Level Blueprint part 13

Στην συνέχεια απενεργοποιείται η δυνατότητα κίνησης του χαρακτήρα του παίκτη, αναπαράγεται επιλεγμένος ήχος και εξαφανίζεται ο actor HookPoint3 με τη χρήση της συνάρτησης Disappear του BP\_HookPoint. Κατόπιν μετακινείται ο χαρακτήρας του παίκτη με τη χρήση της συνάρτησης SetCharacterLocation/Rotation και δημιουργείται actor τύπου BP\_BlockShieldSimple (Εικόνα 312).



Εικόνα 312 Level1 Level Blueprint part 14

Έπειτα χρησιμοποιώντας τη συνάρτηση Open του actor BP\_BlockShieldSimple, μπλοκάρετε η διαδρομή για την αρχή της πίστας. Ακολούθως ενεργοποιείται η ικανότητα κίνησης του χαρακτήρα του παίκτη, αναπαράγεται επιλεγμένος ήχος, ενημερώνεται το κείμενο της αποστολής του παίκτη και δημιουργείται actor τύπου BP\_CandyDisin. Τέλος εμφανίζονται χρησιμοποιώντας την συνάρτηση Appear του BP\_HookPoints 2 actors αυτού του τύπου (Εικόνα 313).



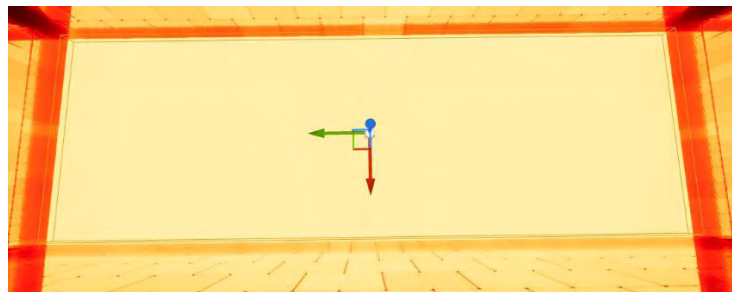
Εικόνα 313 Level1 Level Blueprint part 15

Στη δεύτερη σύγκρουση του χαρακτήρα του παίκτη αναπαράγονται όλοι οι επιλεγμένοι ήχοι, ενημερώνεται η αποστολή του παίκτη και ενεργοποιείται η δεξιότητα teleport του χαρακτήρα. Στη συνέχεια ενημερώνεται η τοποθεσία του actor BP\_EvilOne και η τιμή της μεταβλητής SoundsLocation (Εικόνα 314).



Εικόνα 314 Level1 Level Blueprint part 16

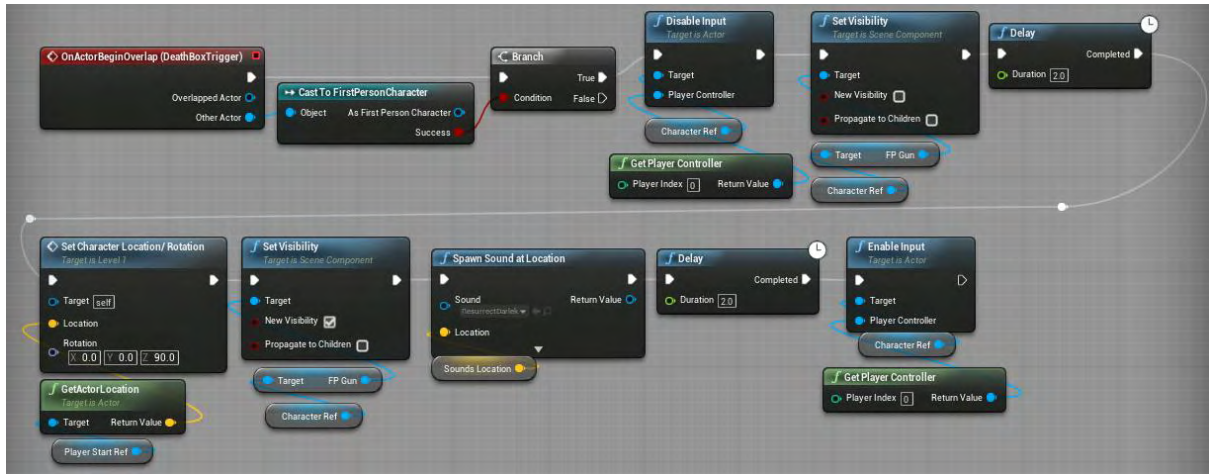
Στην εικόνα 315 απεικονίζεται ο actor DeathBoxTrigger μέσα στο επίπεδο.



Εικόνα 315 DeathBoxTrigger

Μόλις ο χαρακτήρας του παίκτη συγκρουστεί με τον actor DeathBoxTrigger, απενεργοποιείται η δυνατότητα Input του παίκτη, αποκρύπτεται το όπλο του και μετά από χρονοκαθυστέρηση δύο δευτερολέπτων μεταφέρεται στην τοποθεσία της εκκίνησης. Στη συνέχεια γίνεται πάλι ορατό το όπλο του παίκτη, αναπαράγεται συγκεκριμένος ήχος και μετά από προκαθορισμένη χρονοκαθυστέρηση ενεργοποιείται η δυνατότητα Input του παίκτη (Εικόνα 316).





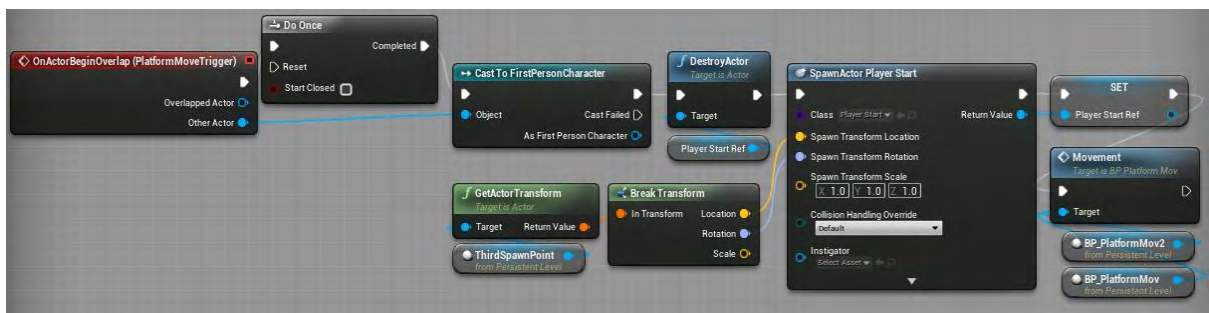
Εικόνα 316 Level1 Level Blueprint part 17

Στην εικόνα 317 απεικονίζεται ο actor PlatformMoveTrigger μέσα στο επίπεδο.



Εικόνα 317 PlatformMoveTrigger

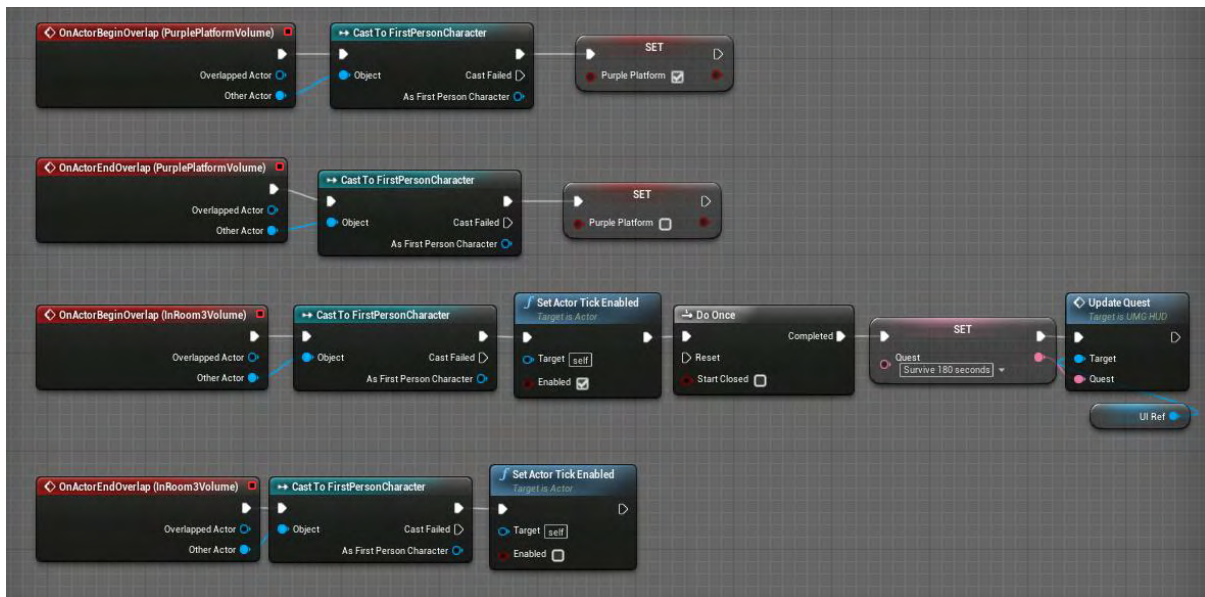
Μόλις ο χαρακτήρας του παίκτη συγκρουστεί με τον actor PlatformMoveTrigger, καταστρέφεται ο actor PlayerStart και δημιουργείται σε άλλη τοποθεσία, ώστε να λειτουργήσει σαν checkpoint. Στη συνέχεια, χρησιμοποιώντας την συνάρτηση Movement του actor τύπου BP\_PlatformMov, ενεργοποιείται η κίνηση των πλατφόρμων (Εικόνα 318).



Εικόνα 318 Level1 Level Blueprint part 18

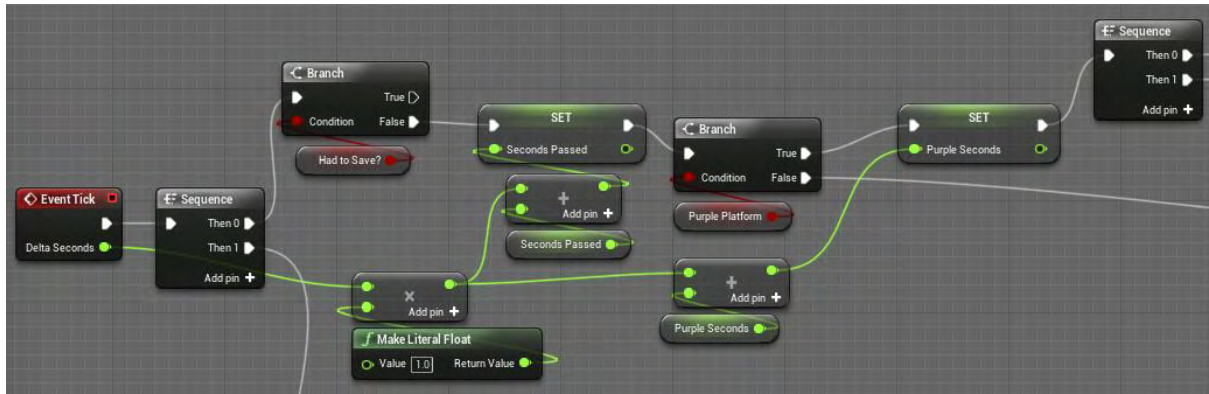


Στην Εικόνα 319 απεικονίζονται οι λειτουργικότητες των trigger Volumes του τρίτου δωματίου. Το PurplePlatformVolume είναι υπεύθυνο για την ενημέρωση της μεταβλητής PurplePlatform, και το InRoom3Volume είναι υπεύθυνο για την ενεργοποίηση και απενεργοποίηση του Event Tick.



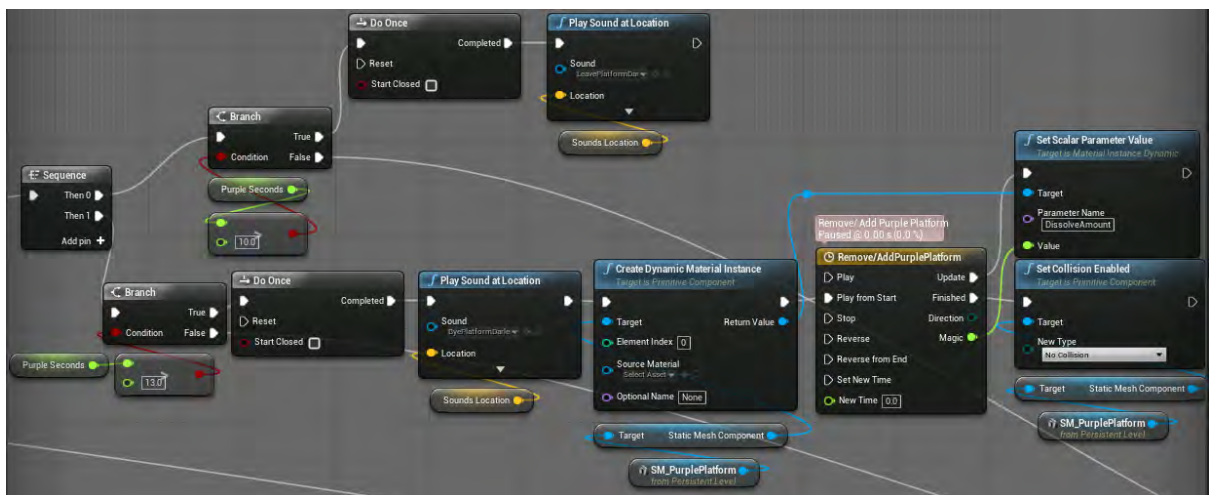
Εικόνα 319 Room3 Triggers

Το Event Tick ενεργοποιείται σε κάθε frame. Υπάρχουν δύο κύριες ακολουθίες που εκτελούνται. Η πρώτη, το μονοπάτι 0, ελέγχει την τιμή της μεταβλητής HadToSave?, η οποία αντιπροσωπεύει αν ο χαρακτήρας του παίκτη έφτασε κοντά στο να καταστραφεί. Όσο αυτό δεν έχει συμβεί, μετρούνται τα συνολικά δευτερόλεπτα που είναι ενεργοποιημένο το Event Tick. Στη συνέχεια ελέγχεται η τιμή της μεταβλητής PurplePlatform που αντιπροσωπεύει αν ο παίκτης στέκεται στην αρχική πλατφόρμα. Αν ο χαρακτήρας του παίκτη βρίσκεται στην αρχική πλατφόρμα, ξεκινούν δύο ακολουθίες ακόμα (Εικόνα 320).

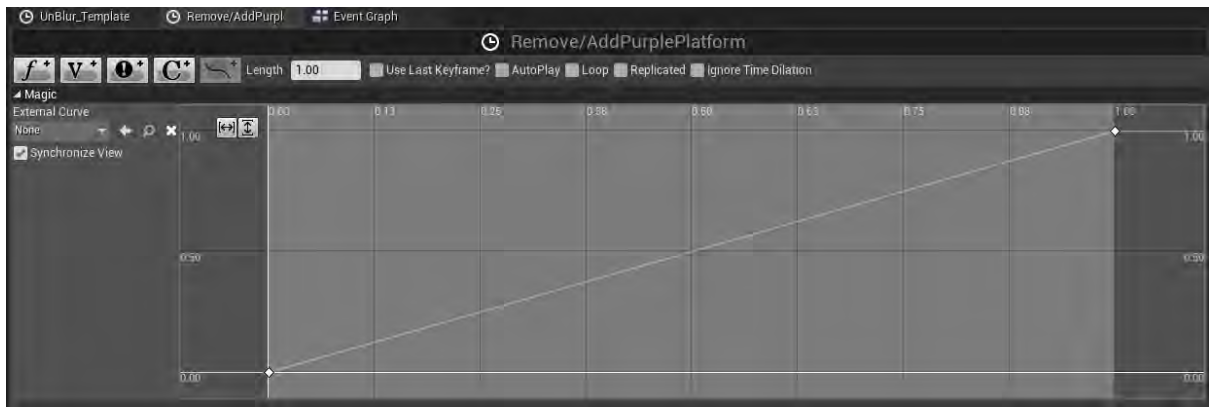


Εικόνα 320 Level1 Level Blueprint part 19

Στην πρώτη ακολουθία (το μονοπάτι 0) ελέγχονται τα δευτερόλεπτα που ο χαρακτήρας του παίκτη βρίσκεται στην αρχική πλατφόρμα. Αν τα δευτερόλεπτα ξεπεράσουν τα 10, ο παίκτης προειδοποιείται να φύγει από την πλατφόρμα. Μόλις τα δευτερόλεπτα ξεπεράσουν τα 13, η πλατφόρμα αποσυντίθεται οπτικά χρησιμοποιώντας το χρονοδιάγραμμα Remove/AddPurplePlatform (Εικόνα 322). Στην συνέχεια απενεργοποιείται η δυνατότητα σύγκρουσης με την πλατφόρμα (Εικόνα 321).

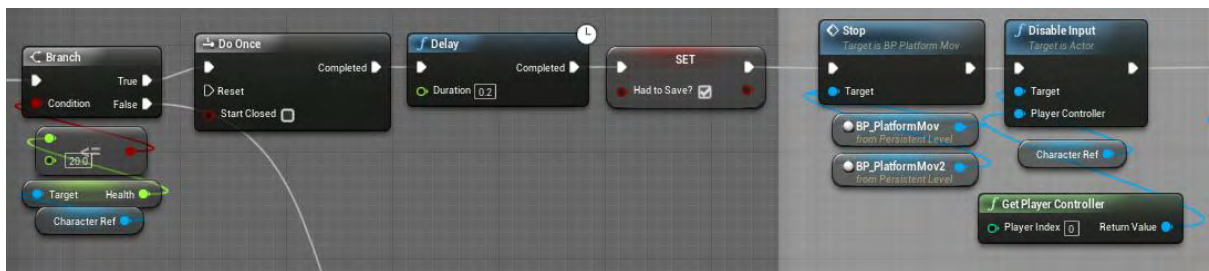


Εικόνα 321 Level1 Level Blueprint part 20



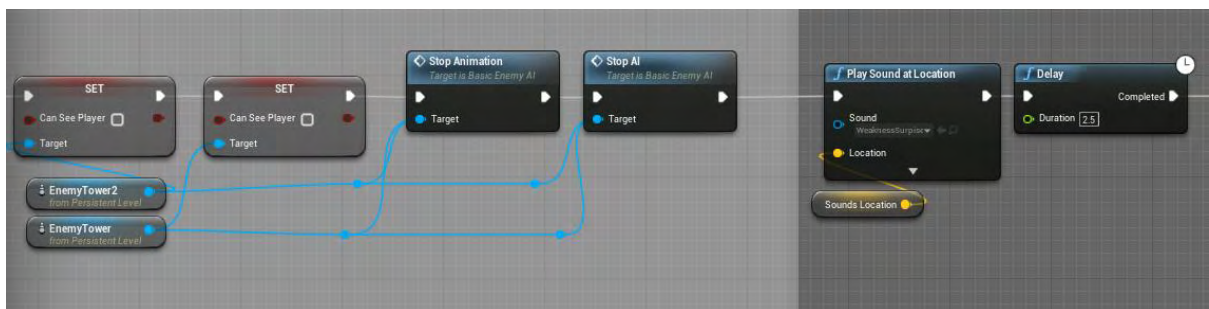
Εικόνα 322 Remove-AddPurple

Αν ο χαρακτήρας του παίκτη δε βρίσκεται στην αρχική πλατφόρμα, ελέγχεται η ζωή του και αν η τιμή της είναι κάτω από 20, ενημερώνεται η τιμή της μεταβλητής HadToSave?. Κατόπιν σταματάει η κίνηση των πλατφόρμων με τη χρήση της συνάρτησης Stop και απενεργοποιείται η δυνατότητα Input του παίκτη (Εικόνα 323).



Εικόνα 323 Level1 Level Blueprint part 21

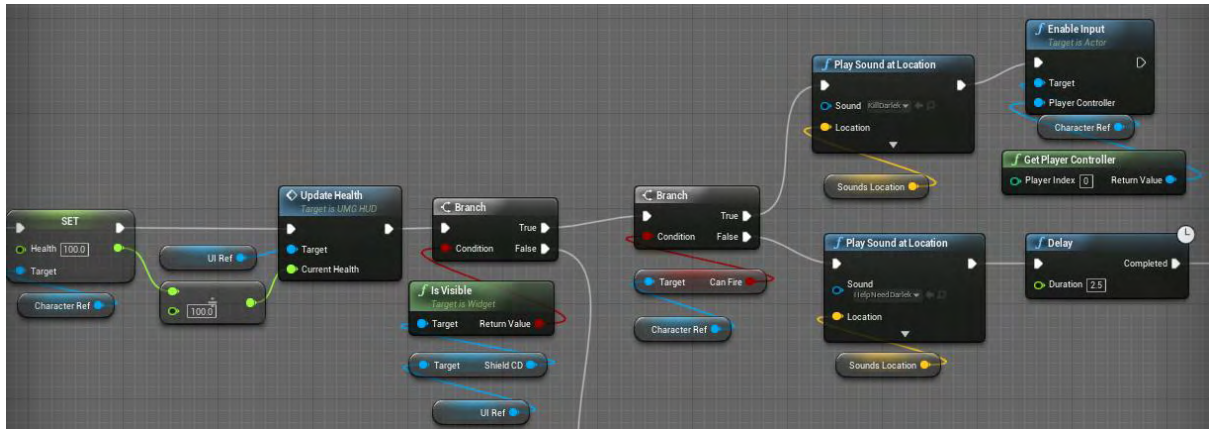
Στη συνέχεια, οι δυο εχθροί του τρίτου δωματίου παύουν το πυρ προς τον χαρακτήρα του παίκτη και αναπαράγεται συγκεκριμένος ήχος (Εικόνα 324).



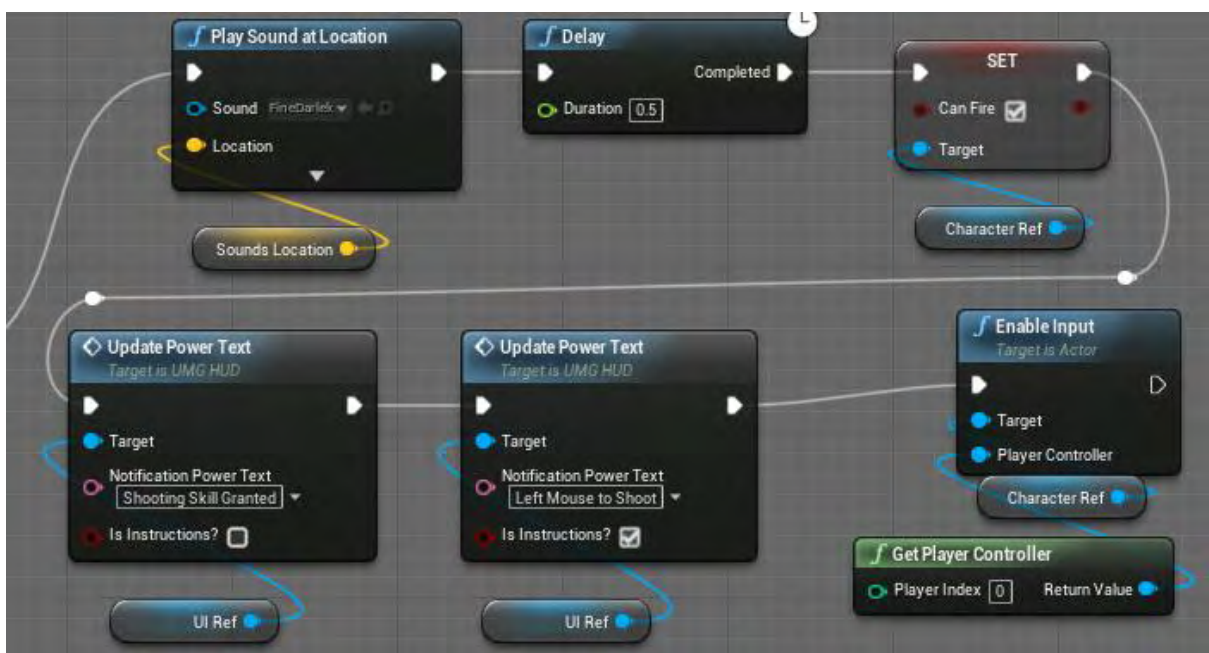
Εικόνα 324 Level1 Level Blueprint part 22



Μετά ενημερώνεται η ζωή του χαρακτήρα και πραγματοποιείται έλεγχος για το ποιες δεξιότητες είναι ήδη ενεργοποιημένες. Στην περίπτωση που είναι ήδη όλες ενεργές, ενεργοποιείται η δυνατότητα Input του παίκτη (Εικόνα 325). Στην περίπτωση που ο παίκτης δεν έχει ήδη τη δεξιότητα να πυροβολήσει, του δίνεται και ενημερώνεται ο παίκτης για αυτό (Εικόνα 326).



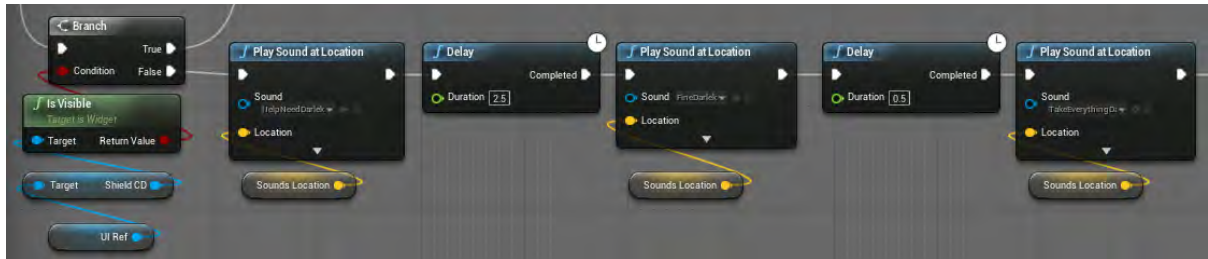
Εικόνα 325 Level1 Level Blueprint part 23



Εικόνα 326 Level1 Level Blueprint part 24

Στην Εικόνα 327 και στην Εικόνα 328 απεικονίζεται η περίπτωση που η δεξιότητα της ασπίδας δεν είναι ενεργοποιημένη, το οποίο σημαίνει ότι ούτε η δεξιότητα του πυροβολισμού είναι ενεργοποιημένη. Οπότε, μετά από μια σειρά ήχων

ενεργοποιούνται και οι δύο δεξιότητες. Ο παίκτης ενημερώνεται για τις νέες του δεξιότητες και ενεργοποιείται η δυνατότητα Input του παίκτη.



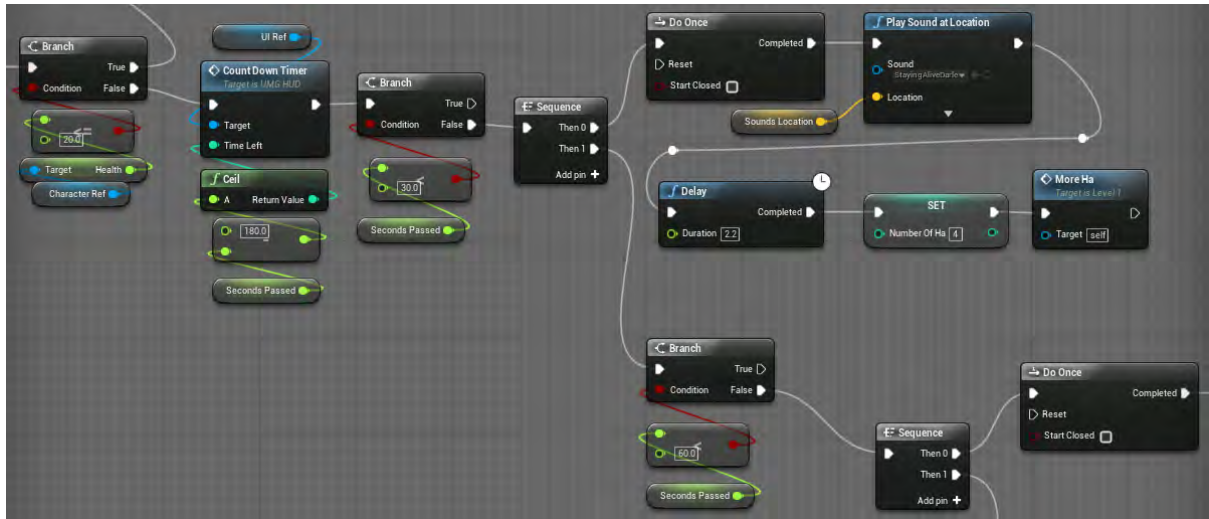
Εικόνα 327 Level1 Level Blueprint part 25



Εικόνα 328 Level1 Level Blueprint part 26

Όσο η ζωή του χαρακτήρα του παίκτη παραμένει πάνω από την αριθμητική τιμή 20, ενημερώνεται συνεχώς ο μετρητής. Στη συνέχεια πραγματοποιείται έλεγχος για τον χρόνο που ο χαρακτήρας του παίκτη βρίσκεται στο τρίτο δωμάτιο. Όταν ο παίκτης επιβιώσει τριάντα δευτερόλεπτα, αναπαράγονται επιλεγμένοι ήχοι. Μόλις ο χρόνος που ο χαρακτήρας του παίκτη βρίσκεται στο τρίτο δωμάτιο ξεπεράσει τα 60 δευτερόλεπτα, ξεκινάει μια νέα ακολουθία με δυο ακόμα μονοπάτια (Εικόνα 329).





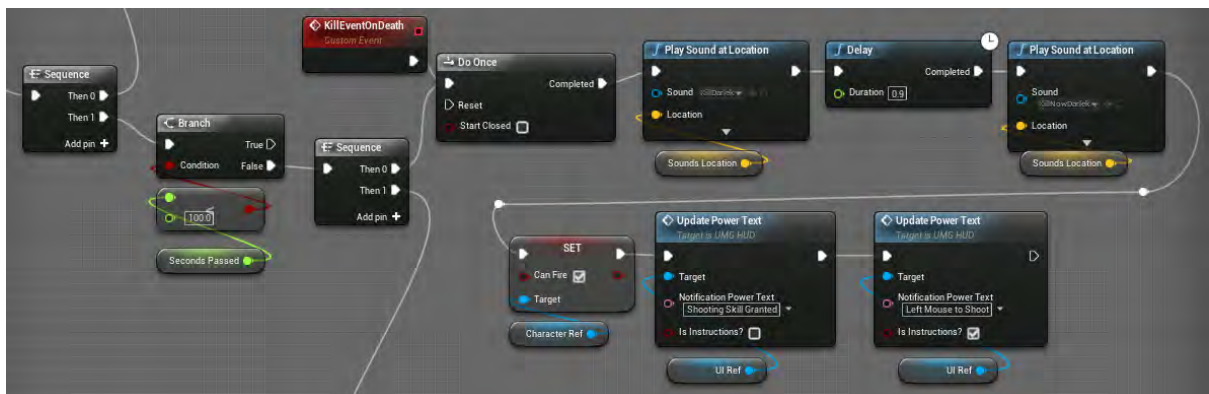
Εικόνα 329 Level1 Level Blueprint part 27

Το πρώτο μονοπάτι ενεργοποιεί τη δεξιότητα της ασπίδας του παίκτη και ενημερώνει τον χρήστη ( Εικόνα 330).



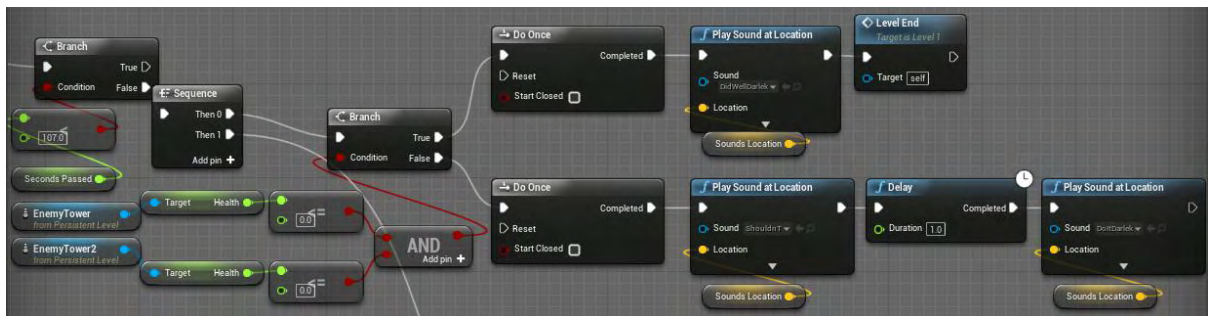
Εικόνα 330 Level1 Level Blueprint part 28

Το δεύτερο μονοπάτι πραγματοποιεί έλεγχο για τον χρόνο. Μόλις ο χρόνος ξεπεράσει τα 100 δευτερόλεπτα, ξεκινά νέα ακολουθία με το πρώτο μονοπάτι να αναπαράγει επιλεγμένους ήχους. Ακολούθως δίνει στον χαρακτήρα του παίκτη τη δεξιότητα να πυροβολήσει και τον ενημερώνει για αυτό. (Εικόνα 332).



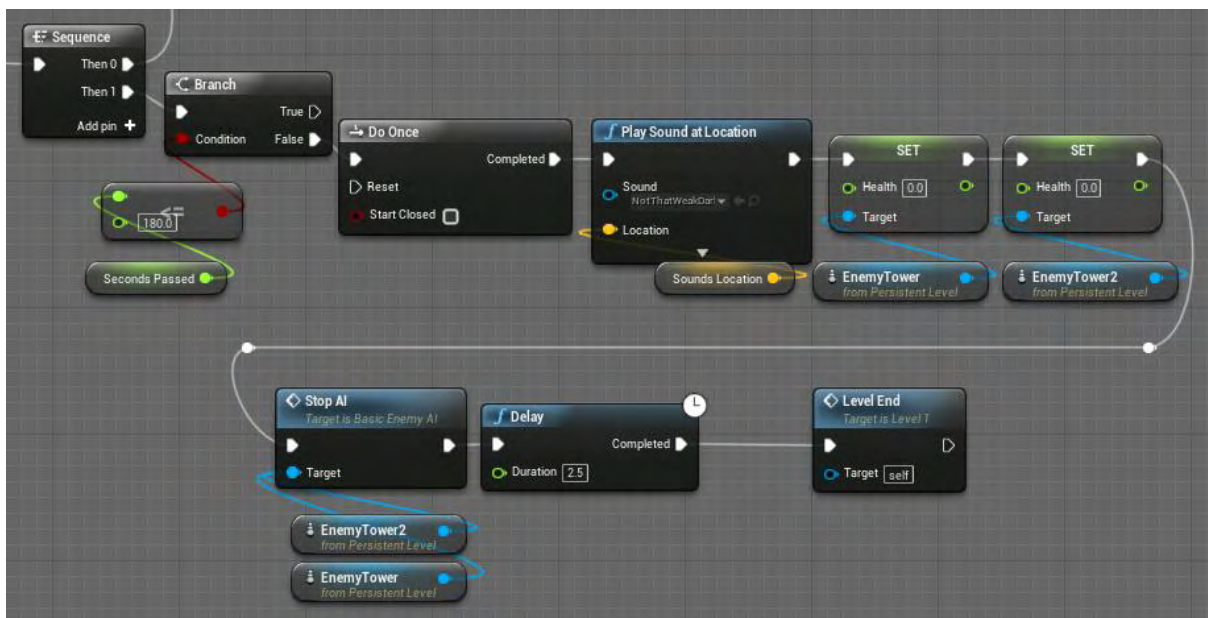
Εικόνα 331 Level1 Level Blueprint part 29

Το δεύτερο μονοπάτι ελέγχει πάλι τον χρόνο. Μόλις ο χρόνος ξεπεράσει τα 107 δευτερόλεπτα, τρέχει μια νέα ακολουθία όπου το πρώτο μονοπάτι ελέγχει τη ζωή των εχθρών στο δωμάτιο και αναλόγως αναπαράγει τους κατάλληλους ήχους. Στην περίπτωση που η ζωή των εχθρών είναι μηδενική, εκτελείται η συνάρτηση LevelEnd (Εικόνα 332).



Εικόνα 332 Level1 Level Blueprint part 30

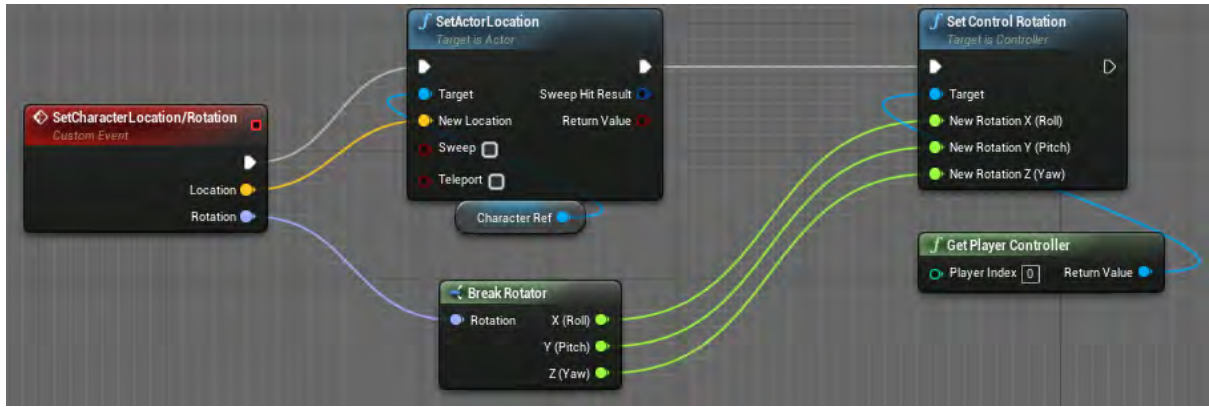
Η ακολουθία στο δεύτερο μονοπάτι ελέγχει τον χρόνο. Μόλις ο χρόνος φτάσει την τιμή των 180 δευτερολέπτων, θέτει μηδέν τη ζωή των εχθρών, συγχαίρει τον παίκτη που επέλεξε να μην τους καταστρέψει μέσω κατάλληλου ήχου και καλεί την συνάρτηση LevelEnd (Εικόνα 333).



Εικόνα 333 Level1 Level Blueprint part 31

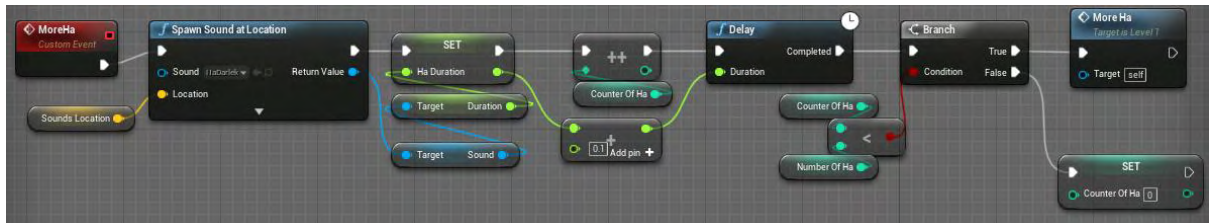






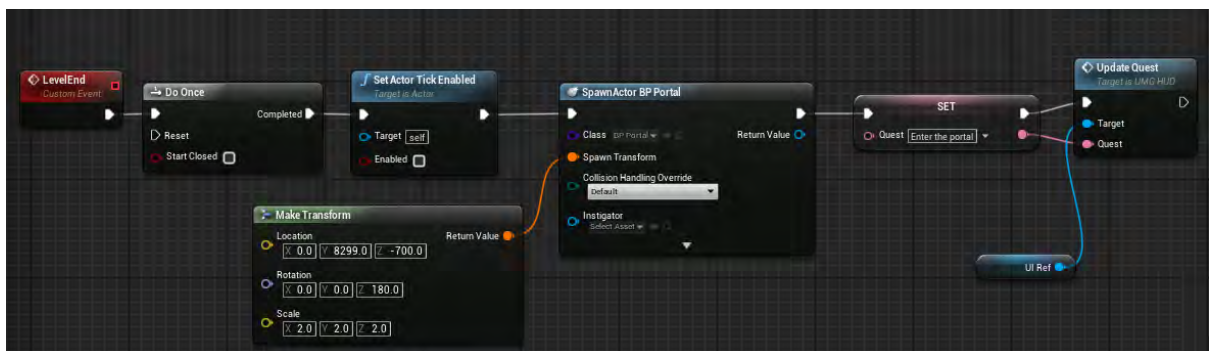
Εικόνα 336 SetCharacterLocation/Rotation

Η συνάρτηση MoreHa αναπαράγει τον ήχο HaDarlek επιλεγμένο αριθμό φορών (Εικόνα 337).



Εικόνα 337 MoreHa

Η συνάρτηση LevelEnd απενεργοποιεί το Event Tick, δημιουργεί actor τύπου BP\_Portal σε επιλεγμένη τοποθεσία και ενημερώνει την αποστολή του παίκτη (Εικόνα 338).



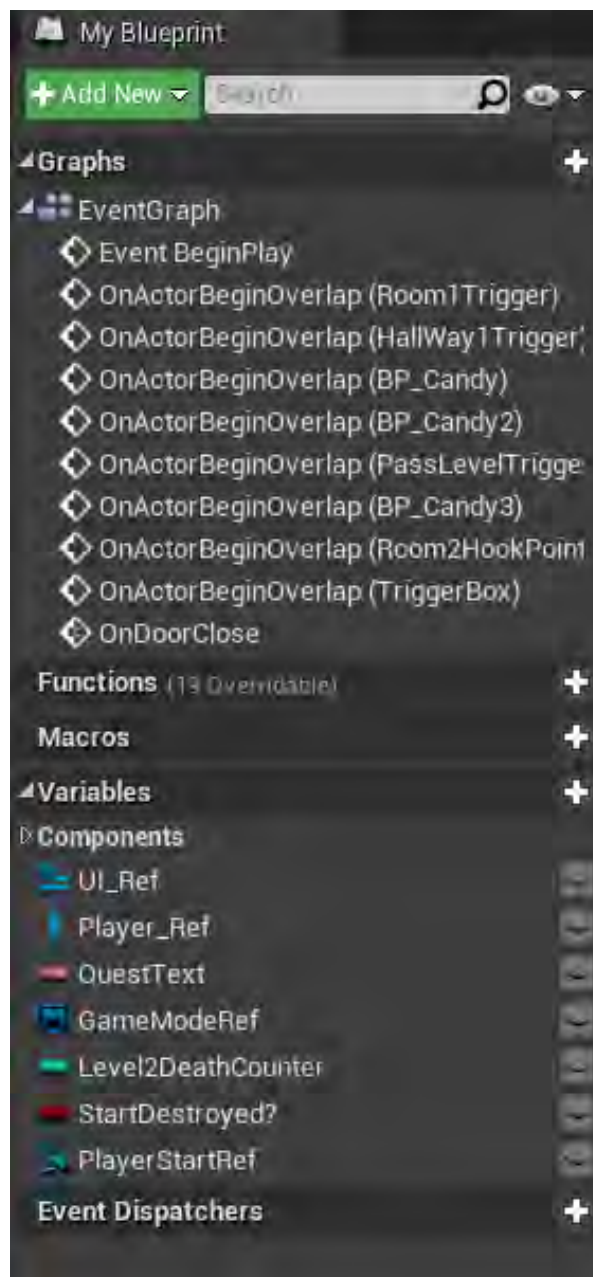
Εικόνα 338 LevelEnd

## Ενότητα 5.9.4 Level2

---

Το επίπεδο με την ονομασία Level2 θέτει τον παίκτη αντιμέτωπο με την πρώτη του δοκιμασία. Ο παίκτης πρέπει να βρει τον κρυμμένο κωδικό που υπάρχει σε δωμάτια για να μπορέσει να συνεχίσει στο επόμενο επίπεδο.

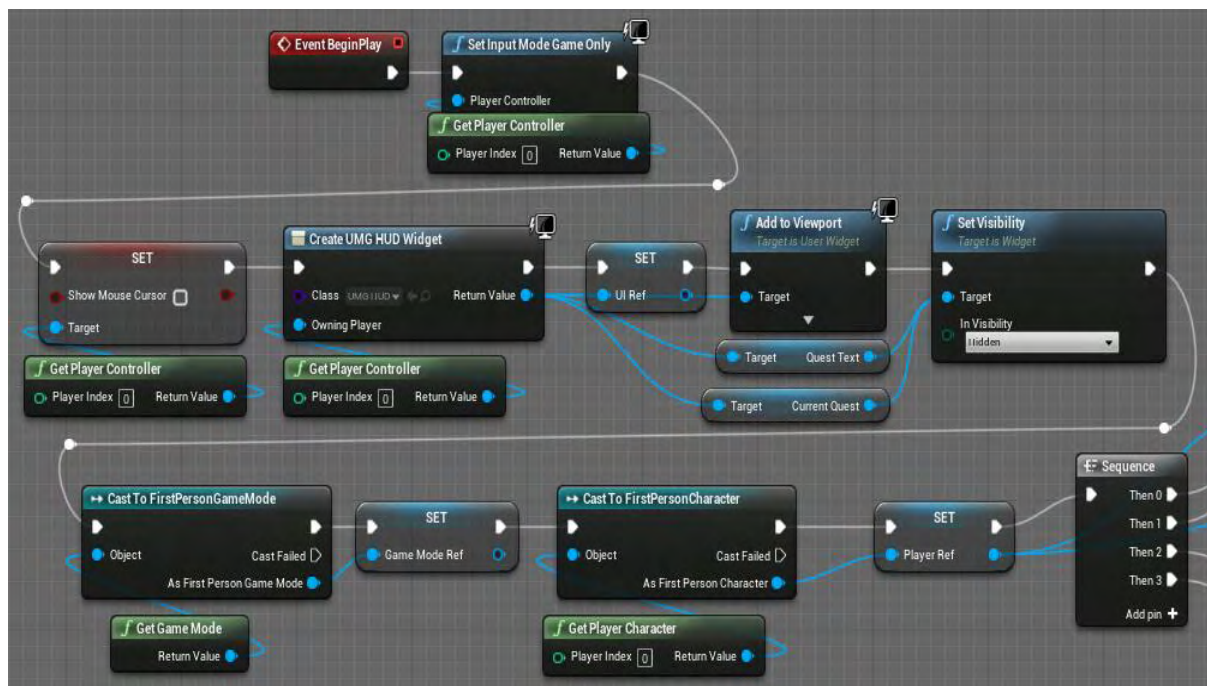
Στην Εικόνα 339 απεικονίζονται οι συναρτήσεις και οι μεταβλητές που χρησιμοποιήθηκαν στο Level Blueprint του επιπέδου.



Εικόνα 339 Level2 My Blueprint Tab

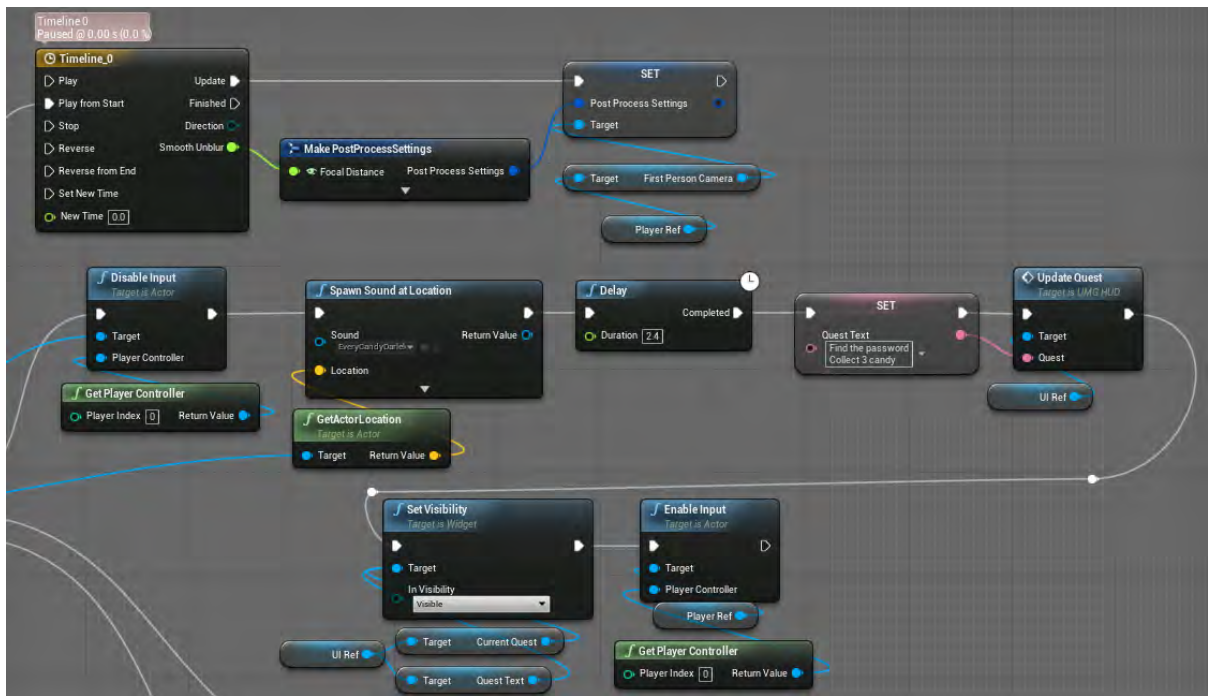


Το Event BeginPlay ενεργοποιείται μόλις δημιουργηθεί το επίπεδο με την ονομασία Level2. Η λειτουργία που υλοποιεί είναι η απόκρυψη του κέρσορα του ποντικιού και θέτεται το Input Mode σε Game Only. Κατόπιν δημιουργείται Widget τύπου UMG\_HUD, προστίθεται στην οθόνη του παίκτη και αποκρύπτονται τα στοιχεία QuestText και Current Quest. Στη συνέχεια δημιουργείται αναφορά στον χαρακτήρα του παίκτη και αποθηκεύεται στη μεταβλητή PlayerRef. Έπειτα ξεκινούν 4 ακολουθίες και εκτελούνται (Εικόνα 340).



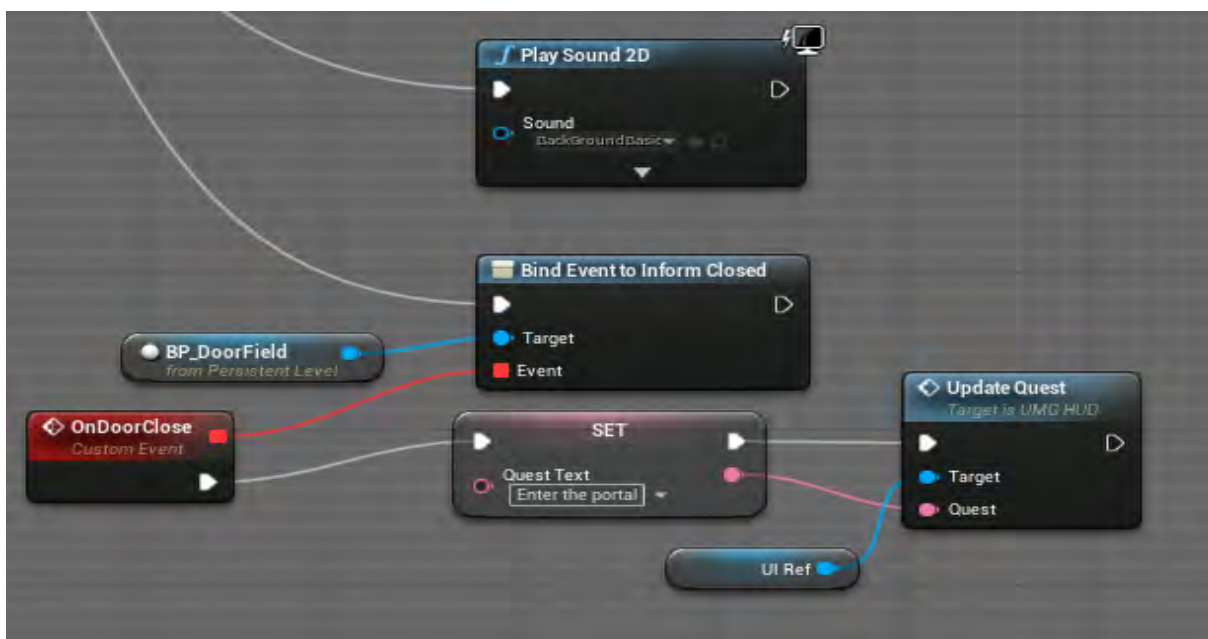
Εικόνα 340 Level2 Level Blueprint part 1

Στο πρώτο μονοπάτι της ακολουθίας γίνεται υλοποίηση του οπτικού εφέ. Στο δεύτερο μονοπάτι απενεργοποιείται η δυνατότητα Input του παίκτη, αναπαράγεται επιλεγμένος ήχος και ενημερώνεται το κείμενο της αποστολής του παίκτη. Τέλος γίνονται ορατά τα στοιχεία CurrentQuest, QuestText και ενεργοποιείται η δυνατότητα Input του παίκτη (Εικόνα 341).



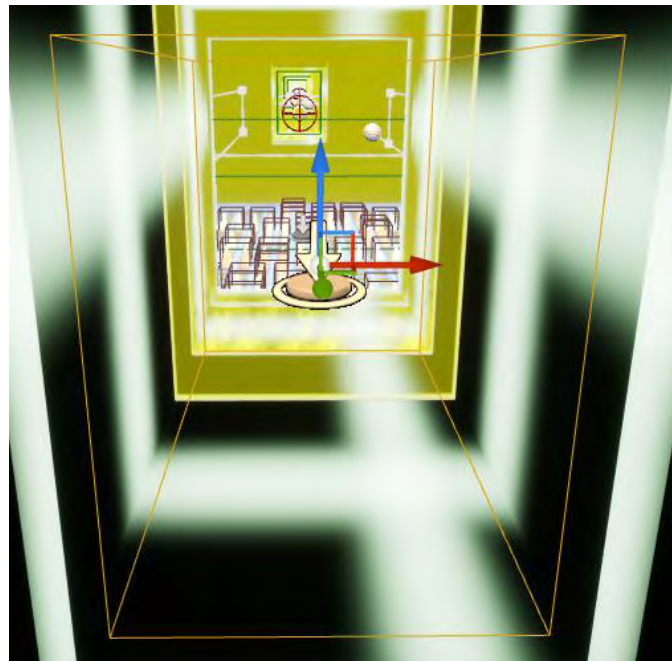
Εικόνα 341 Level2 Level Blueprint part 2

Στο τρίτο μονοπάτι της ακολουθίας ξεκινά η αναπαραγωγή ήχου σαν ήχος υπόβαθρου. Στο τέταρτο μονοπάτι της ακολουθίας δεσμεύεται η συνάρτηση OnDoorClose στο Event Dispatcher InformClosed του actor BP\_DoorField. Η συνάρτηση αυτή ενημερώνει το κείμενο της αποστολής του παίκτη (Εικόνα 342).



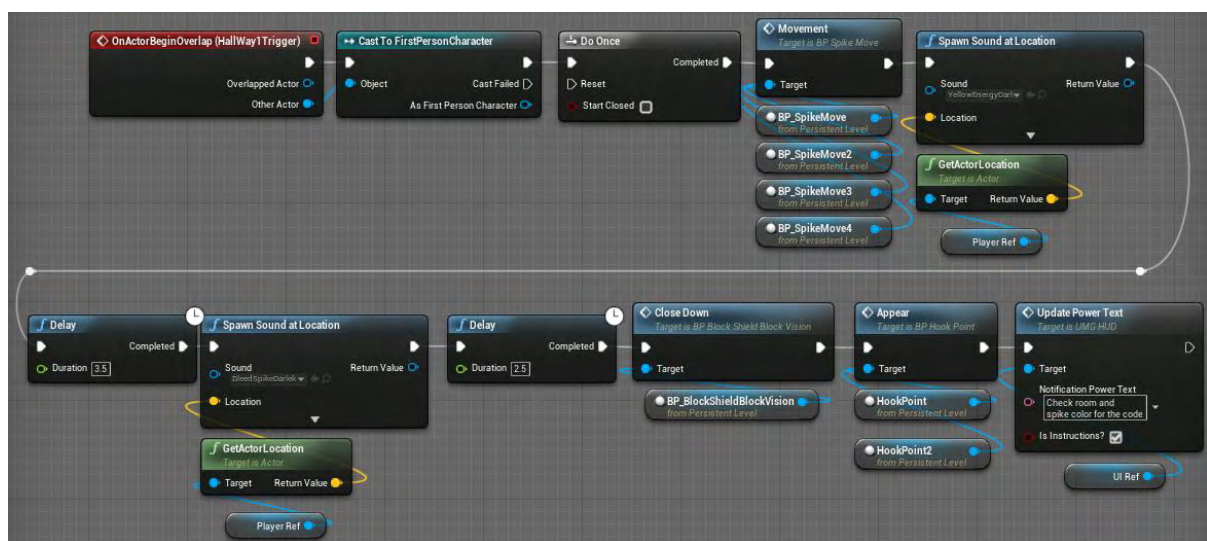
Εικόνα 342 Level2 Level Blueprint part 3

Στην Εικόνα 343 απεικονίζεται ο actor HallWay1Trigger στο επίπεδο.



Εικόνα 343 HallWay1Trigger

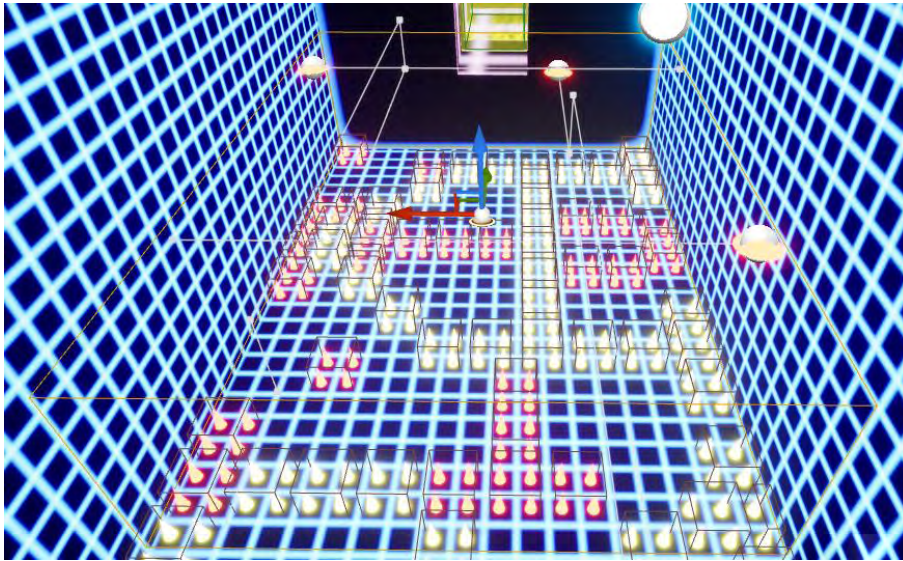
Όταν ο χαρακτήρας του παίκτη συγκρουστεί με το HallWay1Trigger, καλείται η συνάρτηση Movement των actor τύπου BP\_SpikeMove στο πρώτο δωμάτιο. Στη συνέχεια αναπαράγονται επιλεγμένοι ήχοι, καλούνται οι συναρτήσεις CloseDown, Appear του actor BP\_BlockShieldBlockVision και BP\_HookPoint (Εικόνα 344).



Εικόνα 344 Level2 Level Blueprint part 4

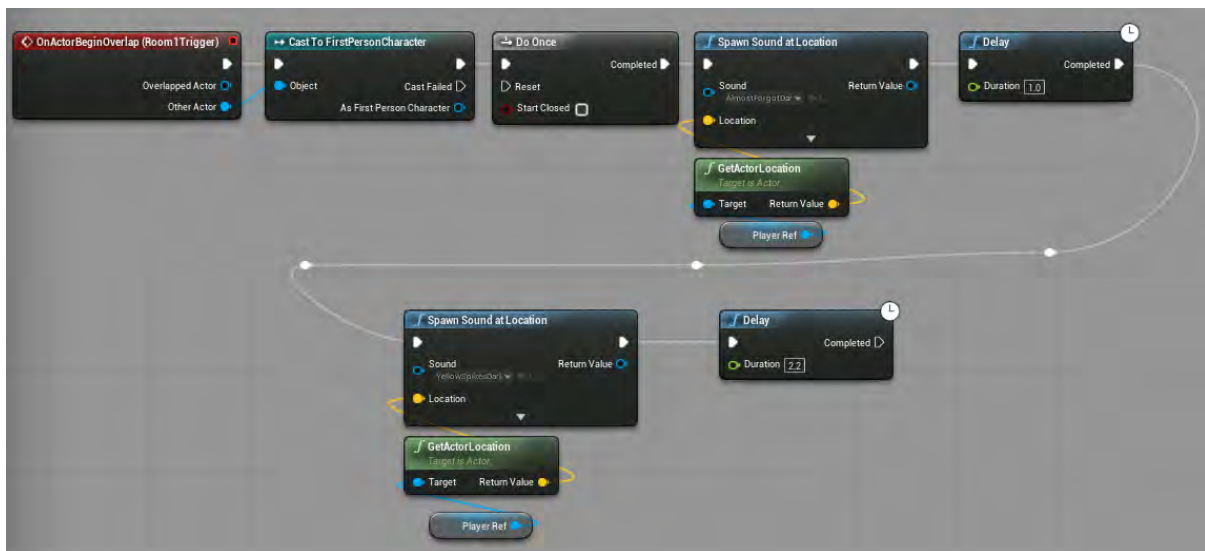


Στην Εικόνα 345 απεικονίζεται ο actor Room1Trigger στο επίπεδο.



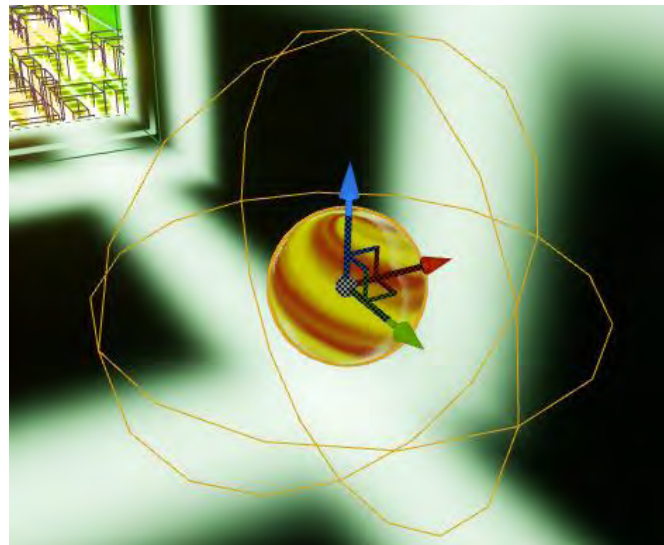
Εικόνα 345 Room1Trigger

Όταν ο χαρακτήρας του παίκτη συγκρουστεί με το Room1Trigger, αναπαράγονται επιλεγμένοι ήχοι (Εικόνα 346).



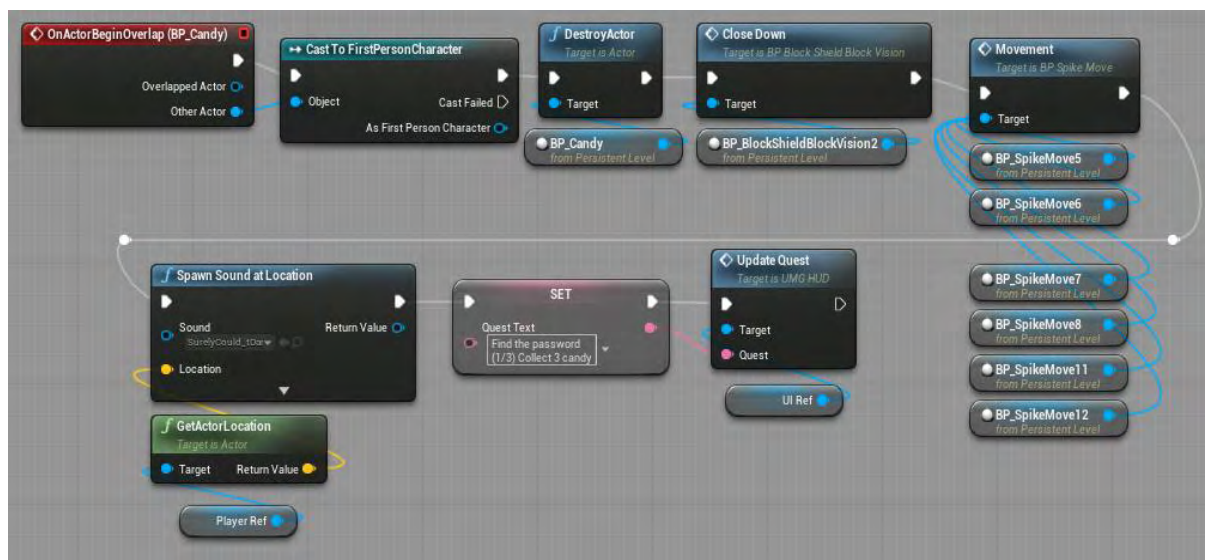
Εικόνα 346 Level2 Level Blueprint part 5

Στην Εικόνα 347 απεικονίζεται ο actor BP\_Candy στο επίπεδο.



Εικόνα 347 BP\_Candy

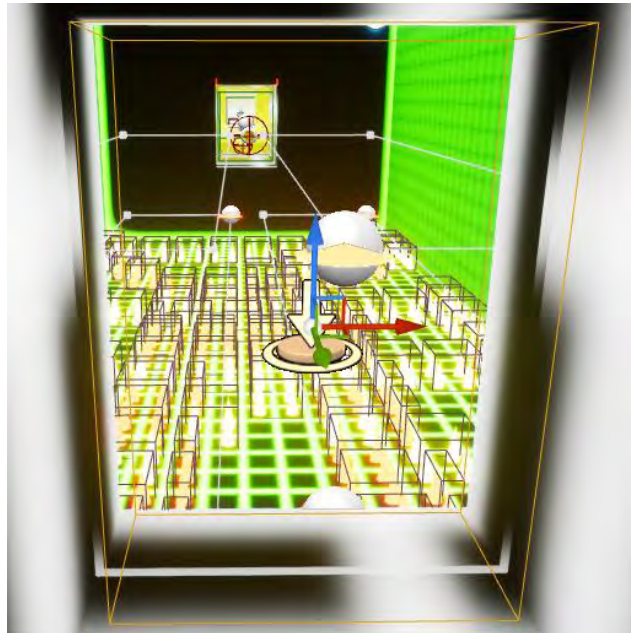
Όταν ο χαρακτήρας του παίκτη συγκρουστεί με το BP\_Candy, καταστρέφεται το BP\_Candy. Στη συνέχεια καλούνται οι συναρτήσεις CloseDown, Movement του actor τύπου BP\_BlockShieldBlockVision και BP\_SpikeMove αντιστοίχα. Τέλος ενημερώνεται το κείμενο αποστολής του παίκτη (Εικόνα 348).



Εικόνα 348 Level2 Level Blueprint part 6



Στην Εικόνα 349 απεικονίζεται ο actor Room2HookPoints στο επίπεδο.



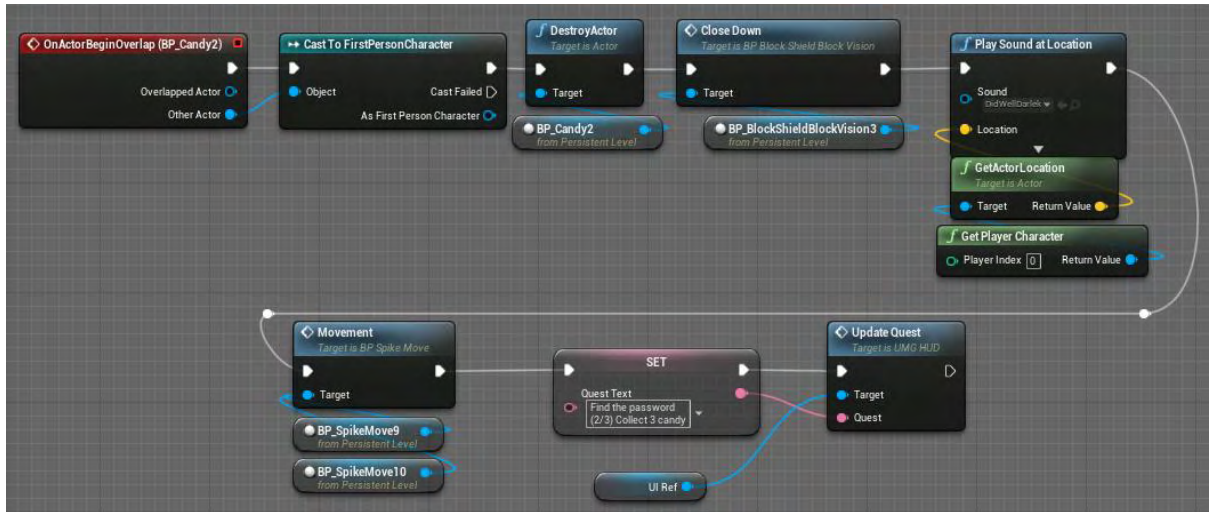
Εικόνα 349 Room2HookPoints

Όταν ο χαρακτήρας του παίκτη συγκρουστεί με το Room2HookPoints καλείται η συνάρτηση Appear του BP\_HookPoint (Εικόνα 350).



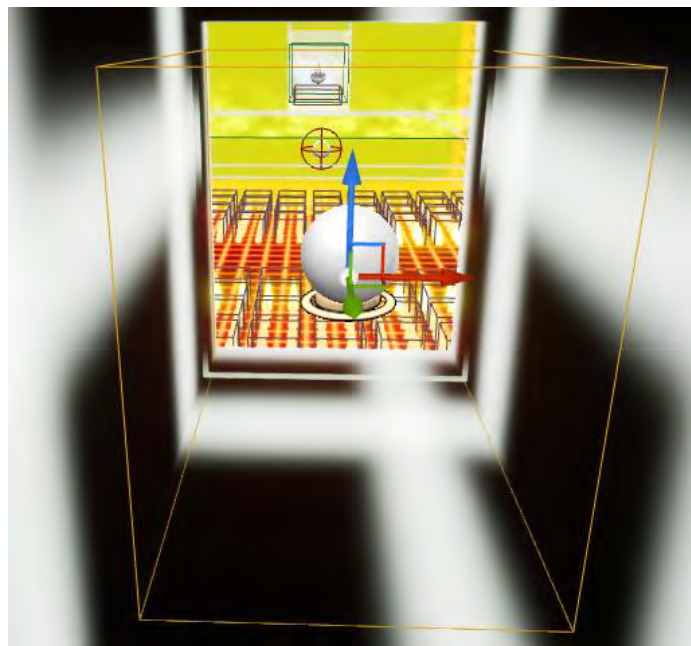
Εικόνα 350 Level2 Level Blueprint part 7

Όταν ο χαρακτήρας του παίκτη συγκρουστεί με το BP\_Candy2 καταστρέφεται το BP\_Candy2. Στη συνέχεια καλούνται οι συναρτήσεις CloseDown, Movement του actor τύπου BP\_BlockShieldBlockVision και BP\_SpikeMove αντίστοιχα. Τέλος ενημερώνεται το κείμενο αποστολής του παίκτη (Εικόνα 351).



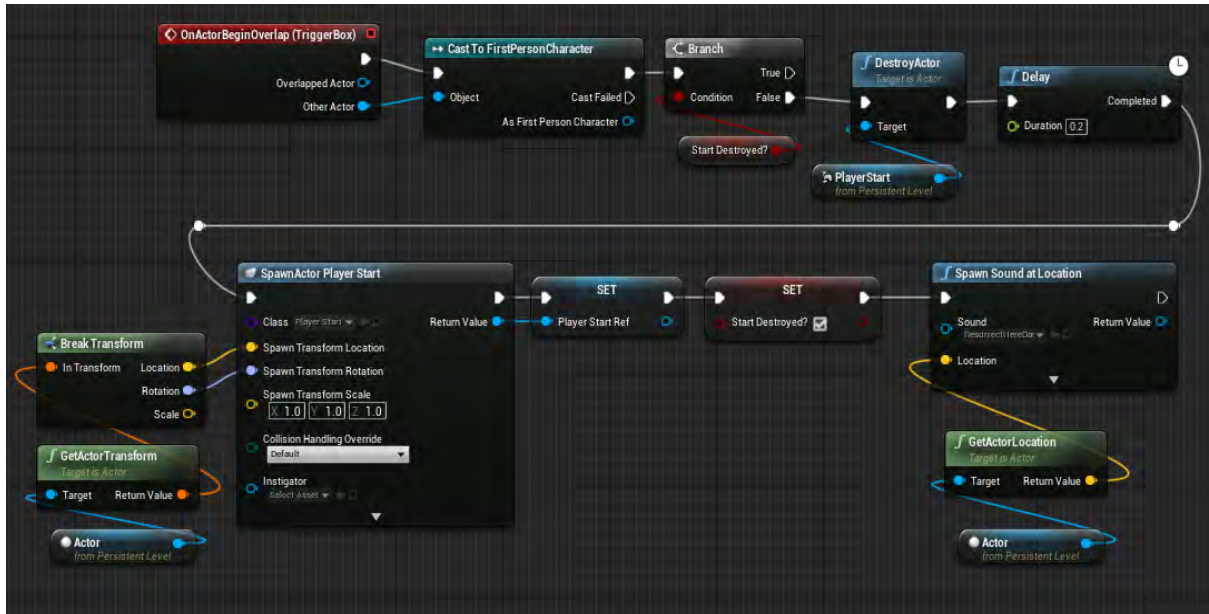
Εικόνα 351 Level2 Level Blueprint part 8

Στην Εικόνα 352 απεικονίζεται ο actor TriggerBox στο επίπεδο.



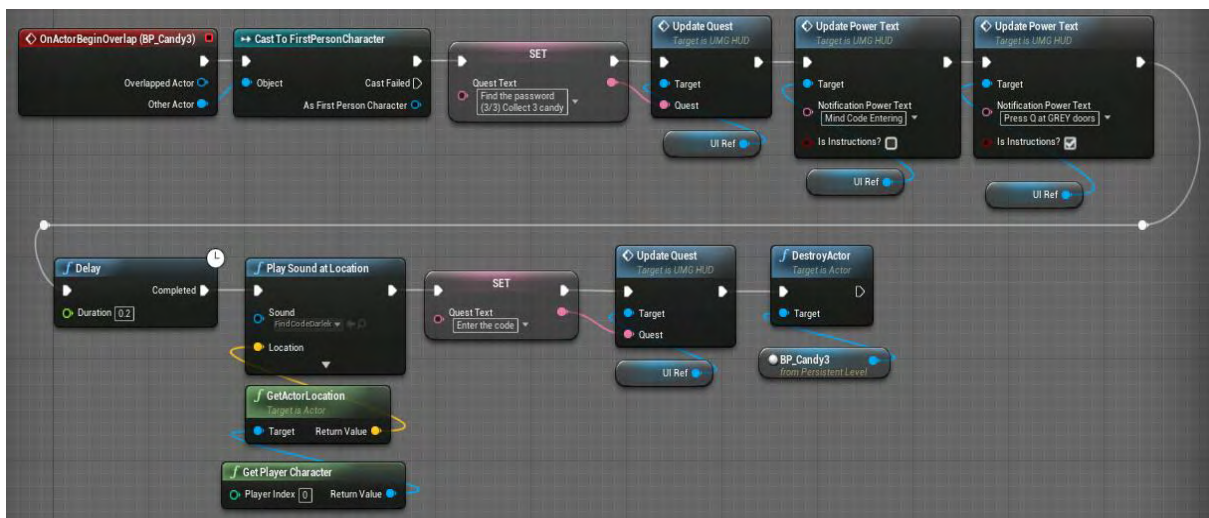
Εικόνα 352 TriggerBox

Όταν ο χαρακτήρας του παίκτη συγκρουστεί με το TriggerBox καταστρέφεται το PlayerStart και δημιουργείται πάλι σε νέα τοποθεσία για να λειτουργήσει σαν checkpoint (Εικόνα 353).



Εικόνα 353 Level2 Level Blueprint part 9

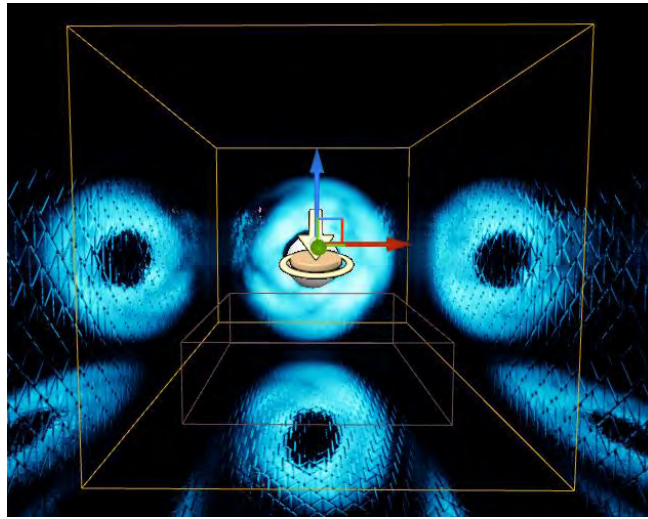
Όταν ο χαρακτήρας του παίκτη συγκρουστεί με το BP\_Candy3, αυτό καταστρέφεται. Στη συνέχεια ενημερώνεται το κείμενο αποστολής του παίκτη και αυτός πληροφορείται για τη νέα του δεξιότητα (Εικόνα 354).



Εικόνα 354 Level2 Level Blueprint part 10



Στην Εικόνα 355 απεικονίζεται στο επίπεδο ο actor PassLevelTrigger, ο οποίος σηματοδοτεί τη φόρτωση του επόμενου επιπέδου.



Εικόνα 355 PassLevelTrigger

## Ενότητα 5.9.5 Level6

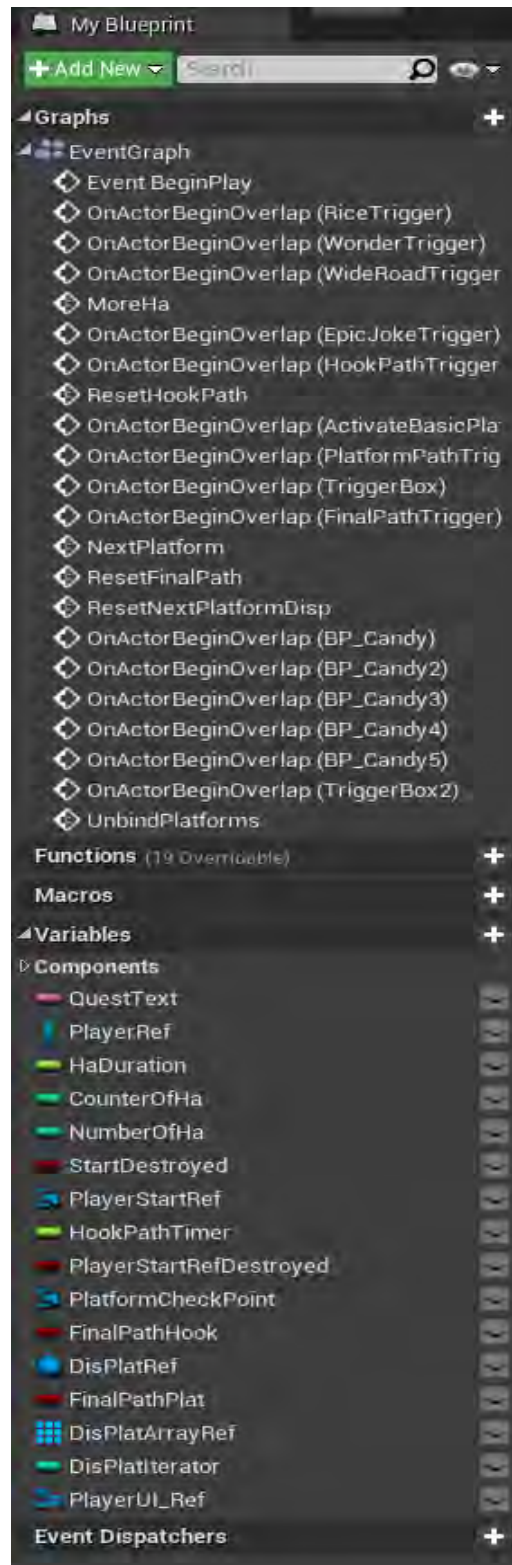
Το επίπεδο με την ονομασία Level6 είναι το τρίτο στην σειρά και αφορά την δεύτερη δοκιμασία που θα αντιμετωπίσει ο παίκτης.

Στην Εικόνα 356 απεικονίζονται οι ρυθμίσεις του επιπέδου στην καρτέλα World Settings.



Εικόνα 356 Level6 World Settings

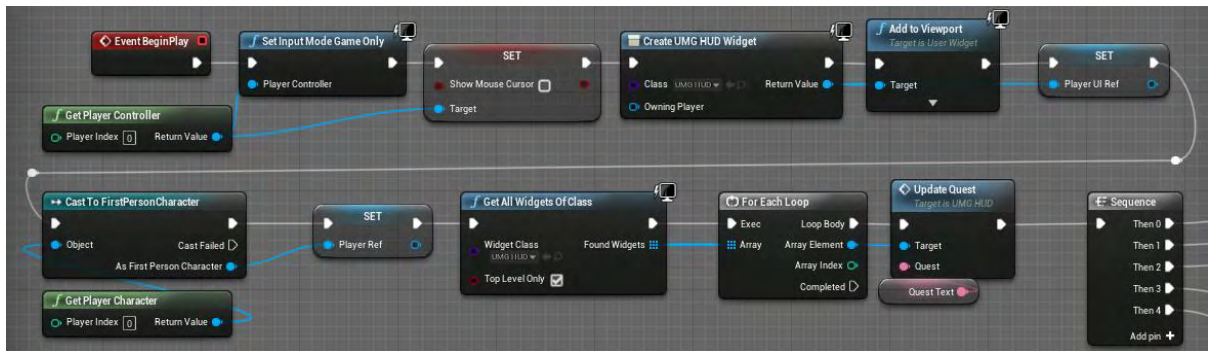
Στην Εικόνα 357 απεικονίζονται οι συναρτήσεις και οι μεταβλητές του επιπέδου στην καρτέλα My Blueprint.



Εικόνα 357 Level6 My Blueprint Tab

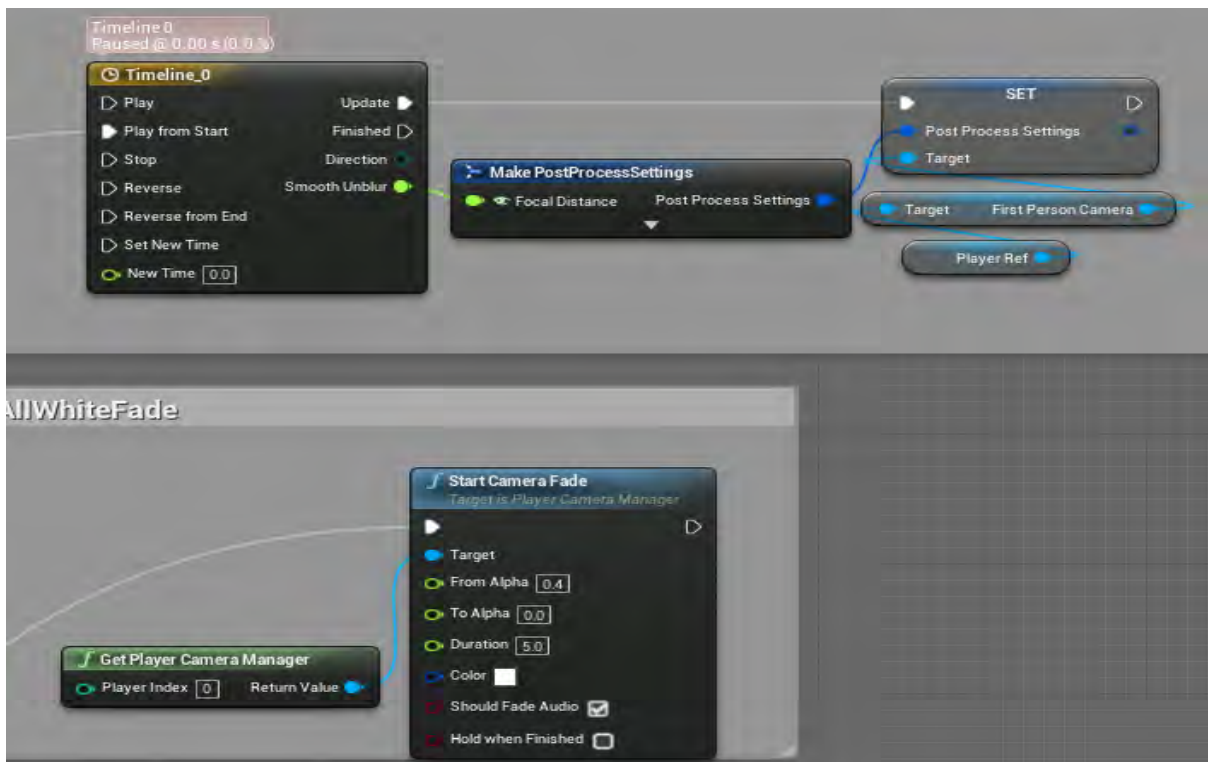


Το Event BeginPlay ενεργοποιείται μόλις δημιουργηθεί το επίπεδο και η λειτουργία που υλοποιεί είναι απόκρυψη του κέρσορα του ποντικιού και θέτεται το Input Mode σε Game Only. Κατόπιν δημιουργείται Widget τύπου UMG\_HUD και προστίθεται στην οθόνη του παίκτη. Στη συνέχεια δημιουργείται αναφορά στον χαρακτήρα του παίκτη και αποθηκεύεται στη μεταβλητή PlayerRef. Κατόπιν ενημερώνεται το κείμενο της αποστολής του παίκτη. Έπειτα ξεκινούν 5 ακολουθίες και εκτελούνται (Εικόνα 358).



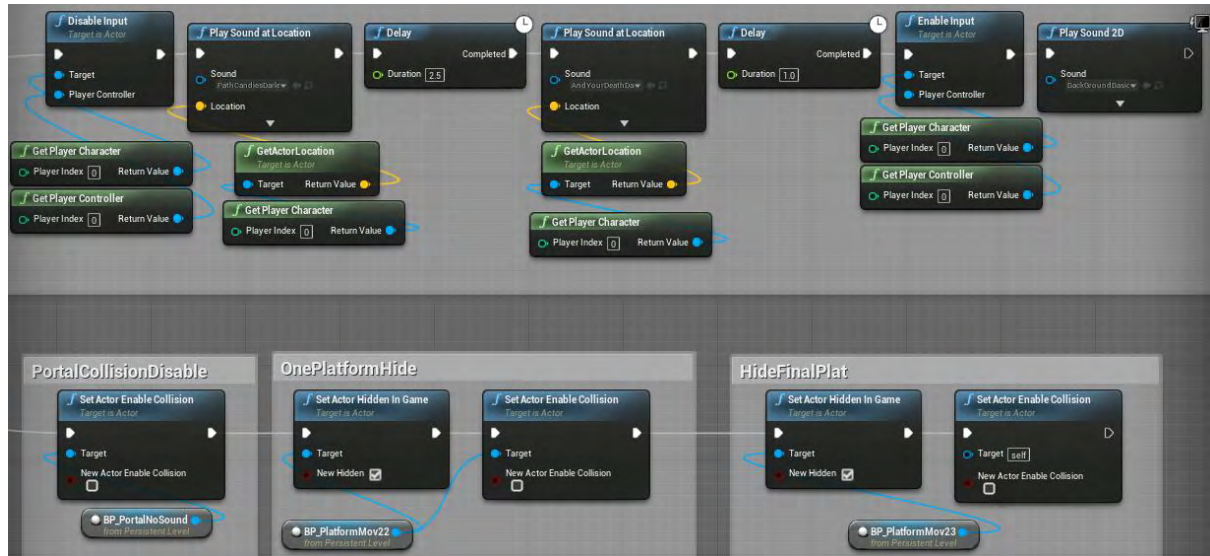
Εικόνα 358 Level6 Level Blueprint part 1

Τα δύο πρώτα μονοπάτια δημιουργούν ένα οπτικό εφέ. Η υλοποίηση φαίνεται στην Εικόνα 359.



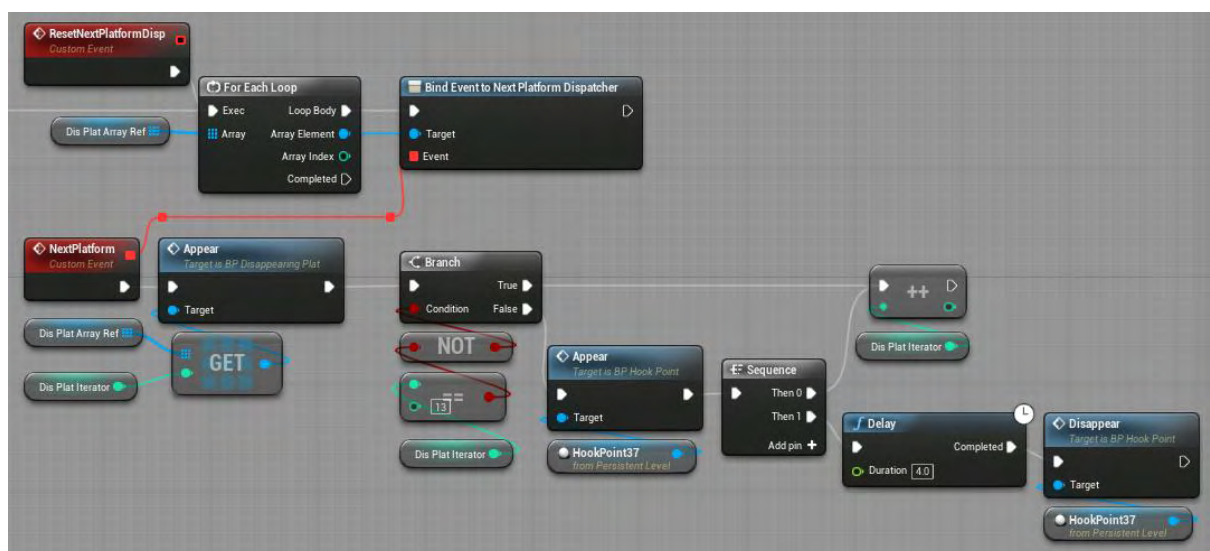
Εικόνα 359 Level6 Level Blueprint part 2

Το τρίτο μονοπάτι απενεργοποιεί το Input του παίκτη, αναπαράγονται επιλεγμένοι ήχοι και ενεργοποιείται πάλι το Input του παίκτη. Το τέταρτο μονοπάτι αποκρύπτει κάποιους actors (Εικόνα 360).



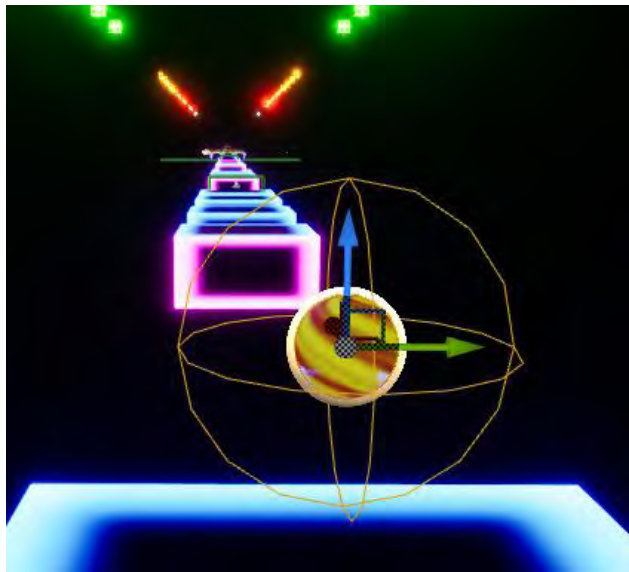
Εικόνα 360 Level6 Level Blueprint part 3

Το πέμπτο μονοπάτι δεσμεύει για όλους τους actors του πίνακα DisPlatArrayRef στο Event Dispatcher (NextPlatformDispatcher) τη συνάρτηση NextPlatform, η οποία εμφανίζει την επόμενη πλατφόρμα από αυτήν που βρίσκεται ο χαρακτήρας του παίκτη. Σχηματίζει δηλαδή το τελικό μονοπάτι του παιχνιδιού (Εικόνα 361).



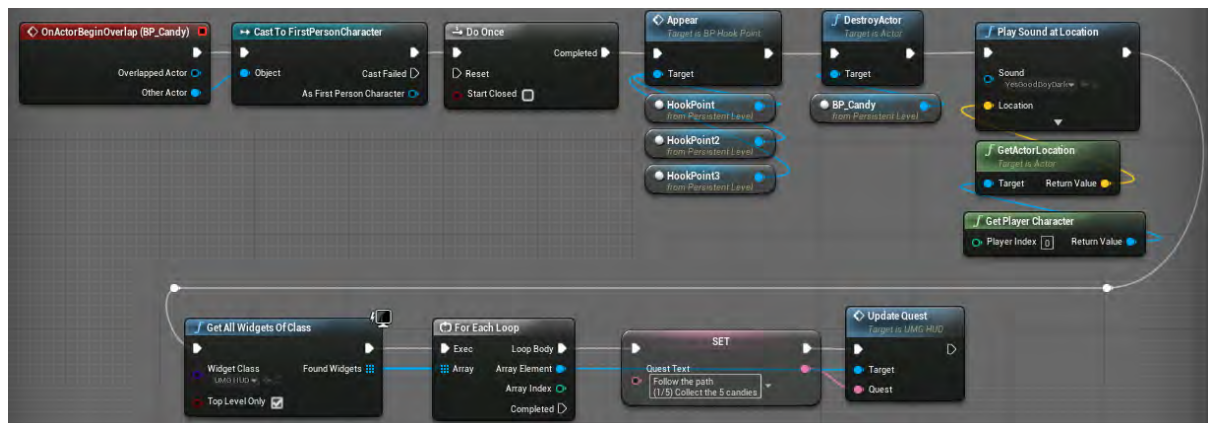
Εικόνα 361 Level6 Level Blueprint part 4

Στην Εικόνα 362 απεικονίζεται ο actor BP\_Candy στο επίπεδο.



Εικόνα 362 BP\_Candy

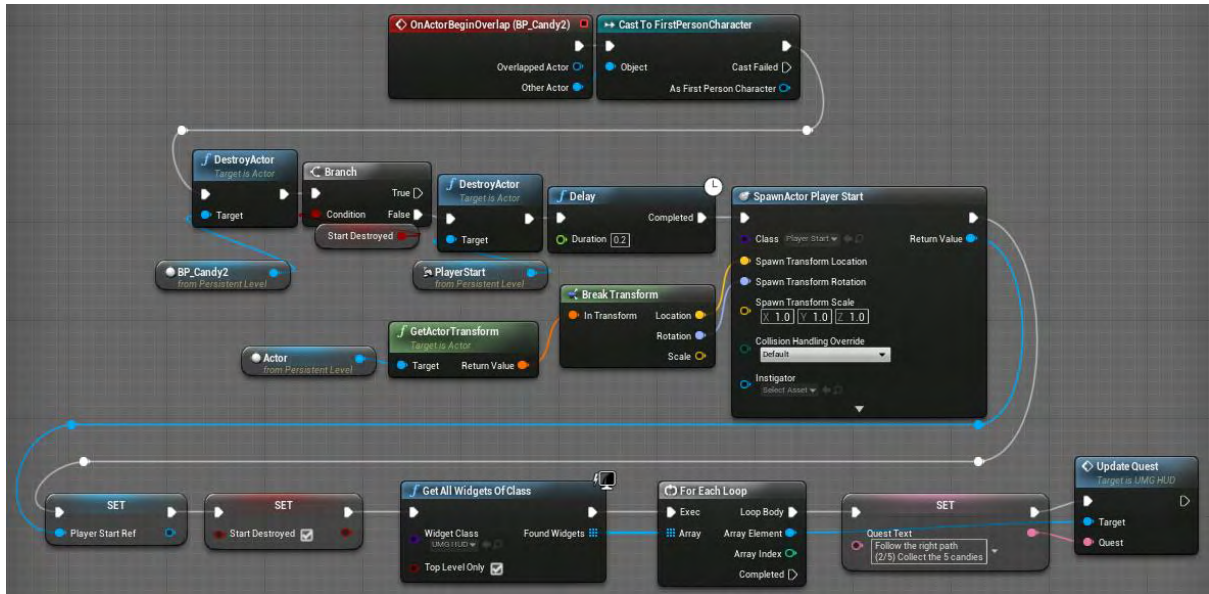
Όταν ο χαρακτήρας του παίκτη συγκρουστεί με το BP\_Candy, αυτό καταστρέφεται. Κατόπιν καλείται η συνάρτηση Arpear και ενημερώνεται το κείμενο αποστολής του παίκτη (Εικόνα 363).



Εικόνα 363 Level6 Level Blueprint part 5

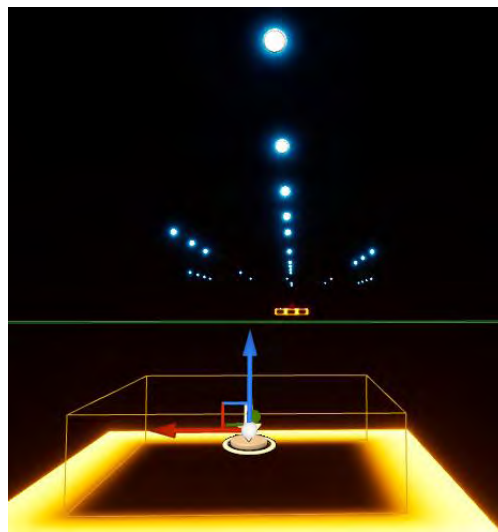
Όταν ο χαρακτήρας του παίκτη συγκρουστεί με το BP\_Candy2, αυτό καταστρέφεται μαζί και ο actor PlayerStart. Στη συνέχεια δημιουργείται καινούριο PlayerStart και ενημερώνεται το κείμενο αποστολής του παίκτη (Εικόνα 364).





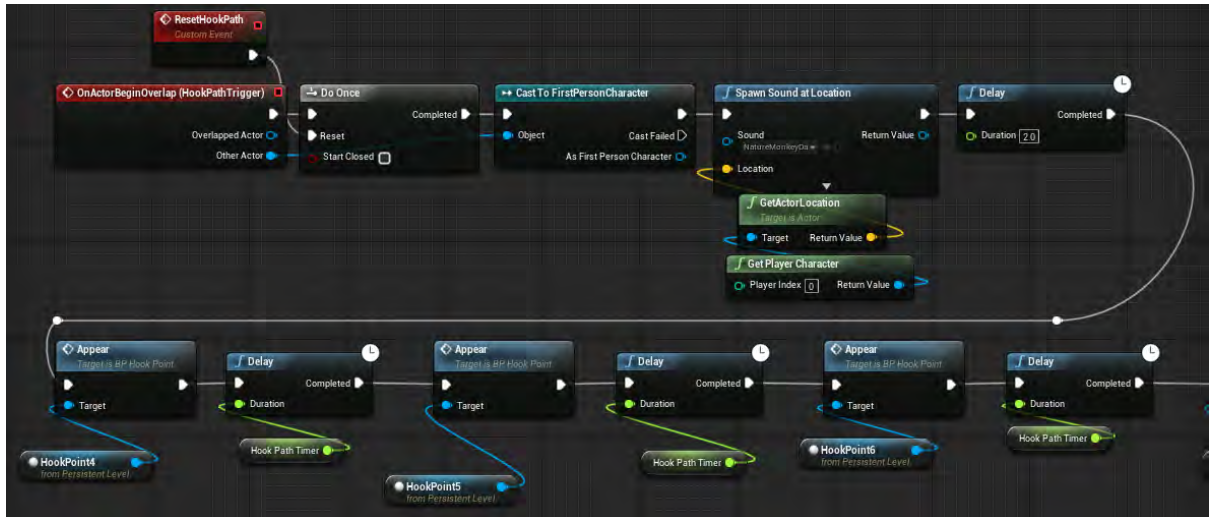
Εικόνα 364 Level6 Level Blueprint part 6

Στην Εικόνα 365 απεικονίζεται ο actor HookPathTrigger και το μονοπάτι από actors τύπου BP\_HookPoints στο επίπεδο.



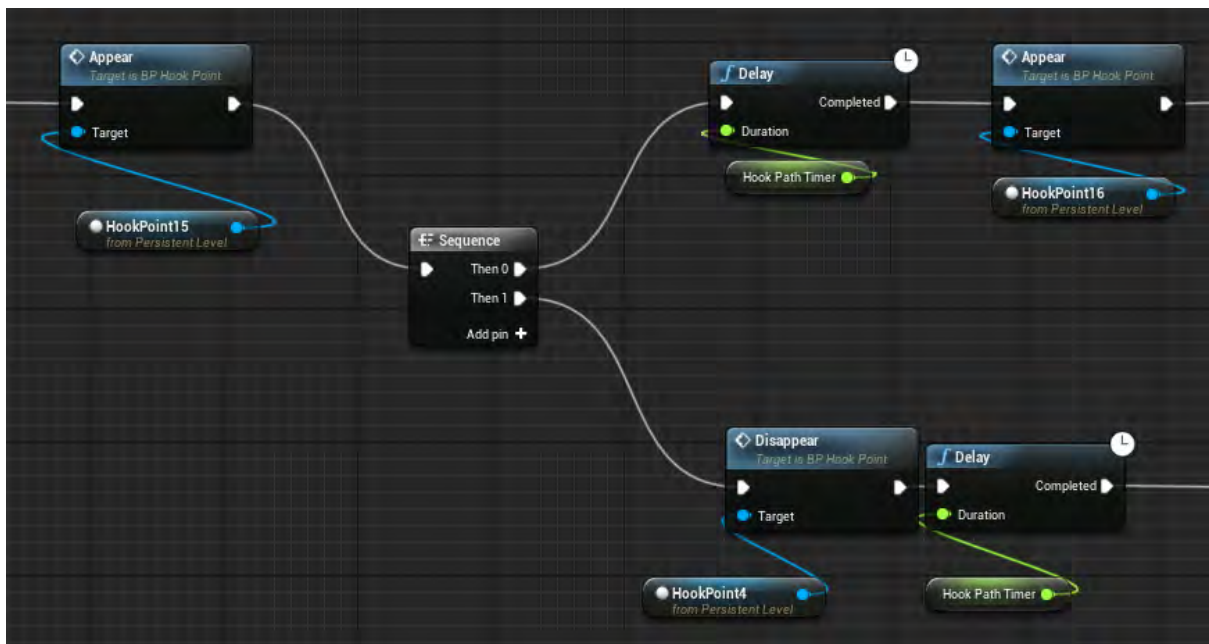
Εικόνα 365 HookPathTrigger and Path

Όταν ο χαρακτήρας του παίκτη συγκρουστεί με το HookPathTrigger, ξεκινά η δημιουργία ενός μονοπατιού χρησιμοποιώντας τη συνάρτηση Appear του BP\_HookPoint (Εικόνα 366).



Εικόνα 366 Level6 Level Blueprint part 7

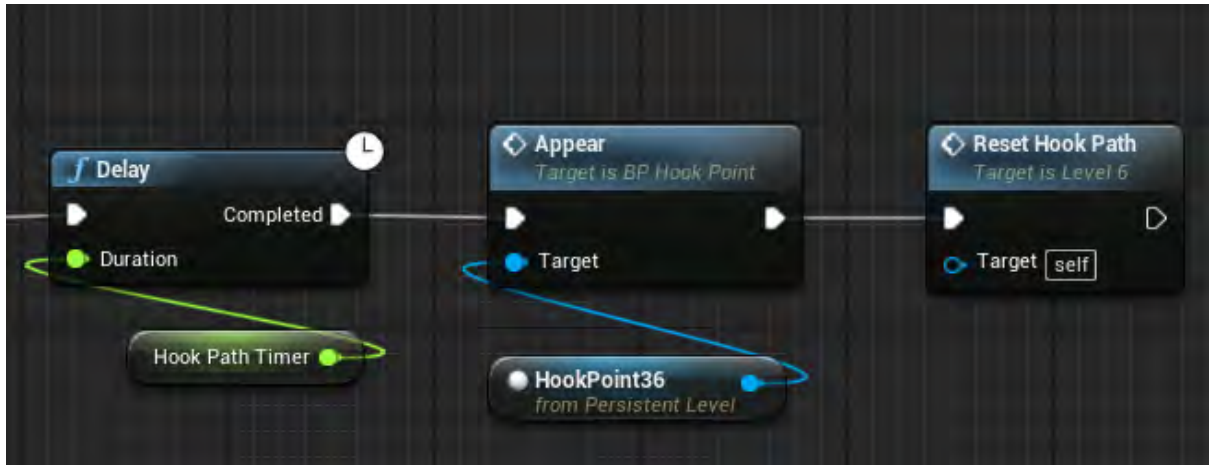
Μετά τη χρήση της συνάρτησης Appear από τον actor HookPoint15 ξεκινά μια επιπλέον διαδρομή στον κώδικα, ο οποίος καλώντας τη συνάρτηση Disappear αποκρύπτει το μονοπάτι από τον παίκτη (Εικόνα 367). Ο παίκτης πρέπει να περάσει το μονοπάτι πριν αυτό προλάβει να εξαφανιστεί.



Εικόνα 367 Level6 Level Blueprint part 8

Αφού πλέον εμφανιστεί το τελευταίο σημείο του μονοπατιού, μπορεί πάλι να ξεκινήσει να εμφανίζεται το μονοπάτι από την αρχή (Εικόνα 368).





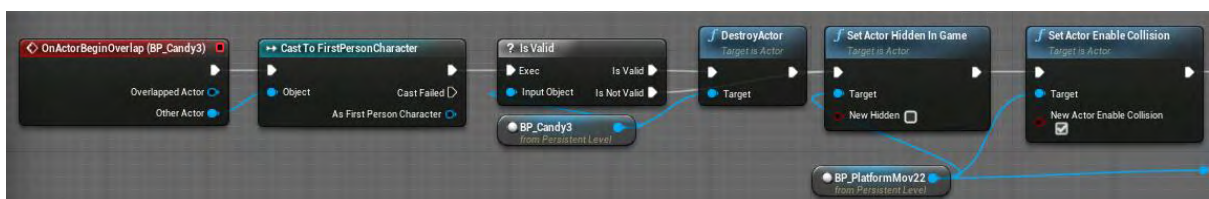
Εικόνα 368 Level6 Level Blueprint part 9

Στην Εικόνα 369 απεικονίζεται ο actor τύπου Launcher που δημιουργείται μετά τη συλλογή του BP\_Candy3.



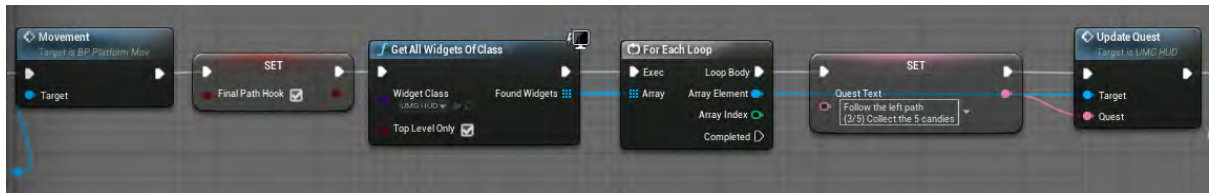
Εικόνα 369 BP\_Candy3/Launcher

Όταν ο χαρακτήρας του παίκτη συγκρουστεί με το BP\_Candy3, αυτό καταστρέφεται και εμφανίζεται η πλατφόρμα, ώστε ο παίκτης να έχει πρόσβαση στο επόμενο μονοπάτι (Εικόνα 370).



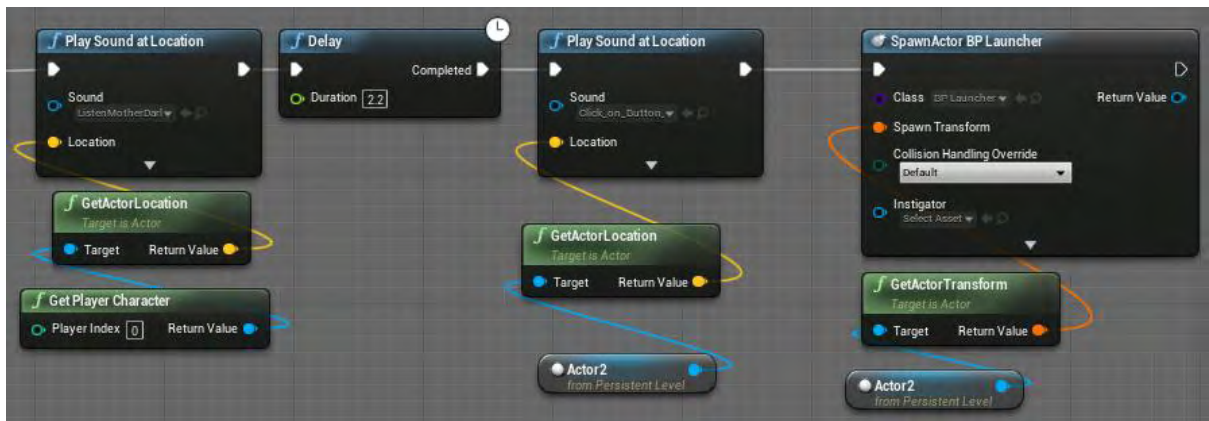
Εικόνα 370 Level6 Level Blueprint part 10

Στη συνέχεια ενημερώνεται η τιμή της μεταβλητής FinalPathHook και το κείμενο αποστολής του παίκτη (Εικόνα 371).



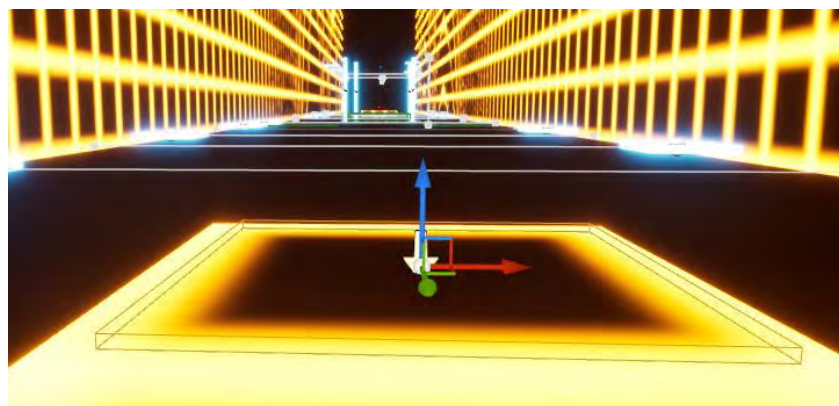
Εικόνα 371 Level6 Level Blueprint part 11

Έπειτα αναπαράγονται οι επιλεγμένοι ήχοι και δημιουργείται ένας actor τύπου BP\_Launcher (Εικόνα 372).



Εικόνα 372 Level6 Level Blueprint part 12

Στην Εικόνα 373 απεικονίζεται το PlatformPathTrigger και το μονοπάτι που θέτει σε κίνηση.



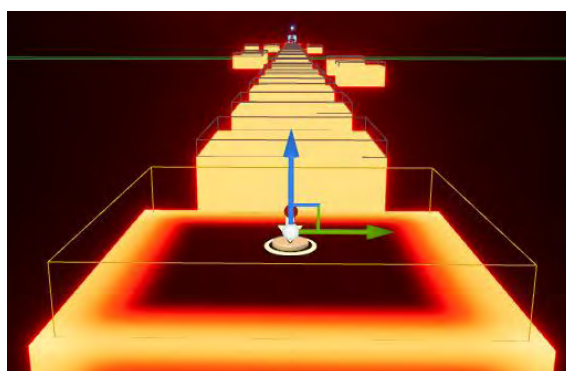
Εικόνα 373 PlatformPathTrigger

Μόλις ο χαρακτήρας του παίκτη συγκρουστεί με το PlatformPathTrigger, τίθεται σε κίνηση το μονοπάτι χρησιμοποιώντας τις συναρτήσεις ReverseSlow και Movement (Εικόνα 374).



Εικόνα 374 Level6 Level Blueprint part 13

Στην Εικόνα 375 απεικονίζεται το FinalPathTrigger.

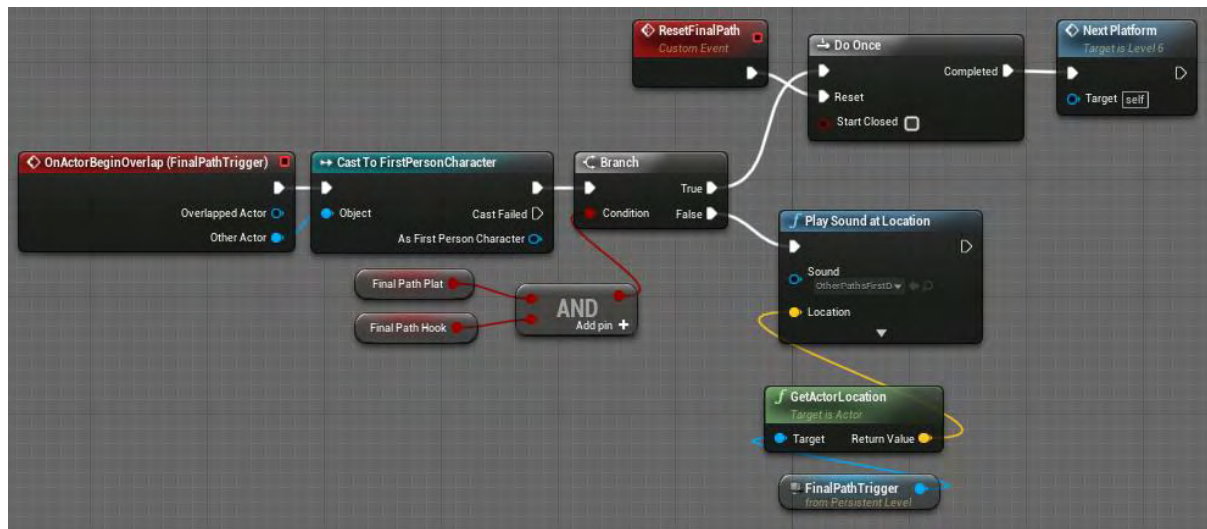


Εικόνα 375 FinalPathTrigger

Μόλις ο χαρακτήρας του παίκτη συγκρουστεί με το FinalPathTrigger, πραγματοποιείται έλεγχος ώστε να διαπιστωθεί αν ο χαρακτήρας του παίκτη έχει περάσει και τα δυο προηγούμενα μονοπάτια. Αν δεν έχει περάσει, αναπαράγεται ένας ήχος που τον ειδοποιεί ότι πρέπει πρώτα να περάσει από τα υπόλοιπα

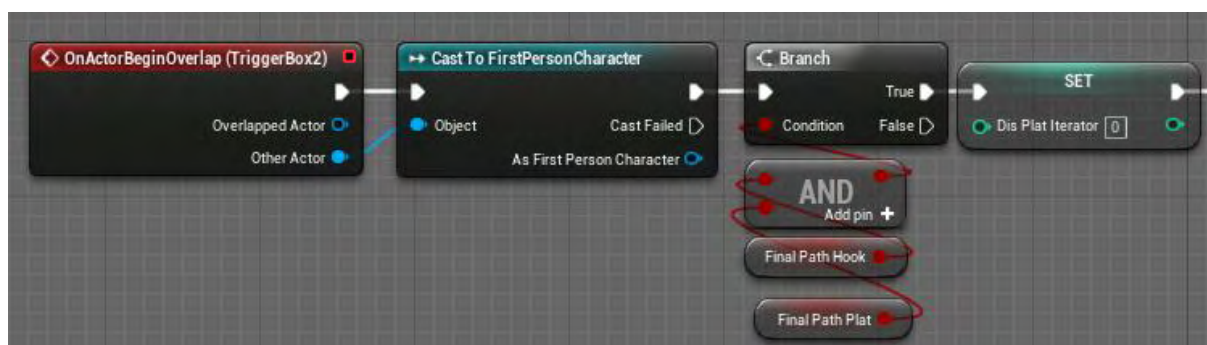


μονοπάτια. Αν έχει περάσει, καλείται η συνάρτηση NextPlatform που εμφανίζει την πρώτη πλατφόρμα (Εικόνα 376).



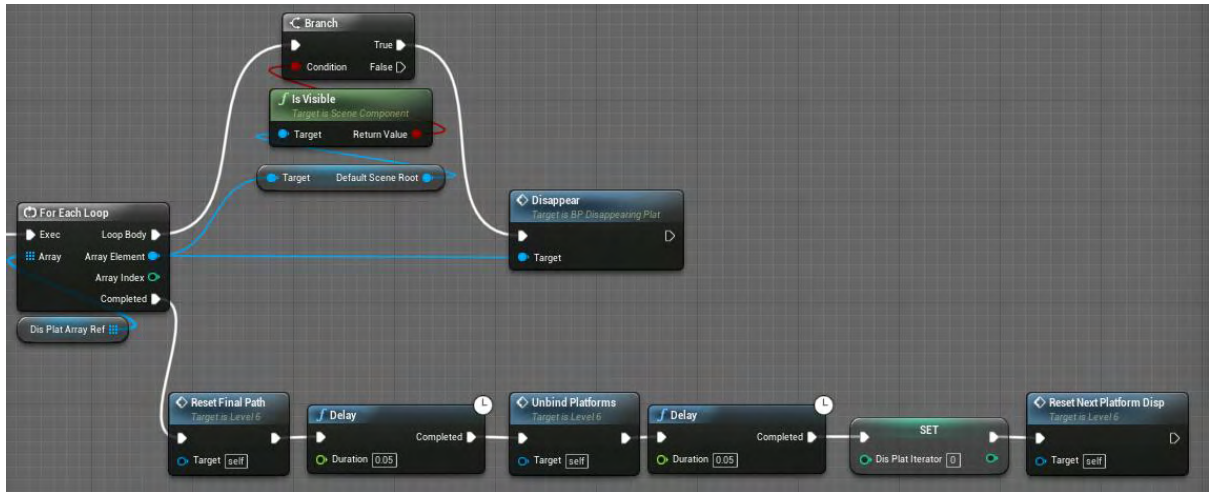
Εικόνα 376 Level6 Level Blueprint part 14

Ο actor TriggerBox2 καταλαμβάνει όλο το επίπεδο αυτό σε ύψος χαμηλότερο από την πίστα. Ο σκοπός του είναι να αρχικοποιεί το τελικό μονοπάτι, όταν ο χαρακτήρας του παίκτη πέφτει και πεθαίνει. Μόλις ο χαρακτήρας του παίκτη συγκρουστεί με το TriggerBox2, πραγματοποιείται έλεγχος των μεταβλητών FinalPathHook και FinalPathPlat. Στη συνέχεια αρχικοποιεί τη μεταβλητή DisPlatIterator (Εικόνα 377).



Εικόνα 377 Level6 Level Blueprint part 15

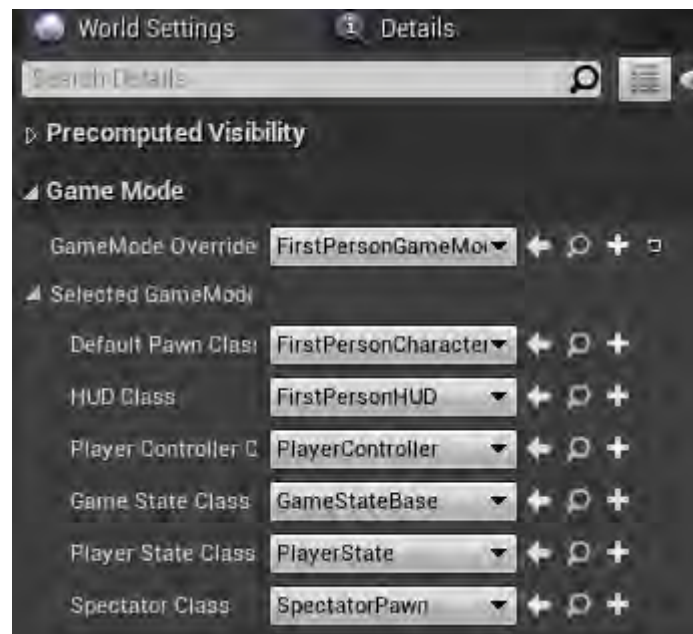
Έπειτα καλείται η συνάρτηση Disappear για τον πίνακα DisPlatArrayRef, ώστε να εξαφανιστούν όλες οι πλατφόρμες. Μόλις διατρεχθεί όλος ο πίνακας, καλούνται οι απαραίτητες συναρτήσεις για να βρεθεί το μονοπάτι στην αρχική του μορφή (Εικόνα 378).



Εικόνα 378 Level6 Level Blueprint part 16

## Ενότητα 5.9.6 Level4

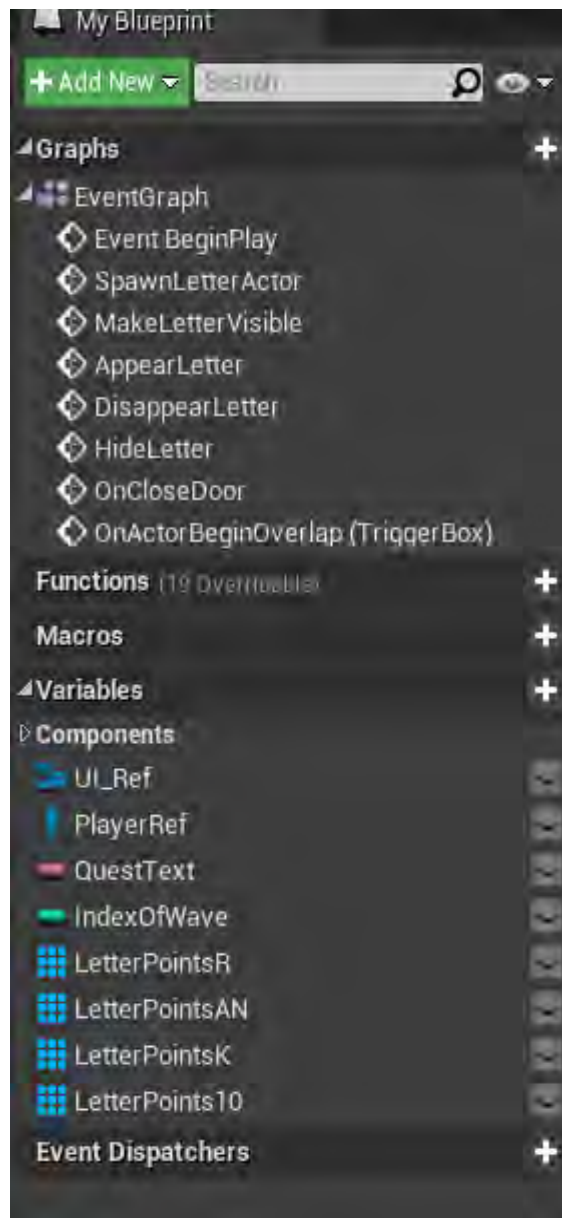
Το επίπεδο με την ονομασία Level4 είναι το τελευταίο επίπεδο του The Challenger. Στην Εικόνα 379 απεικονίζονται οι ρυθμίσεις της καρτέλας World Settings του επιπέδου.



Εικόνα 379 Level4 World Settings

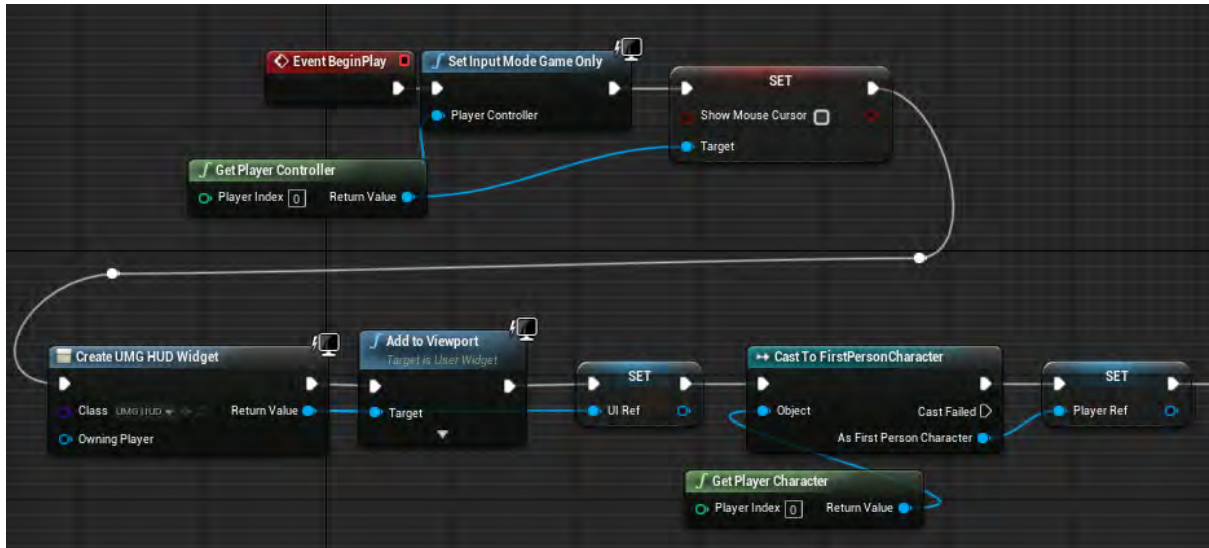
Στην Εικόνα 380 απεικονίζονται οι συναρτήσεις και οι μεταβλητές που χρησιμοποιήθηκαν στο επίπεδο.





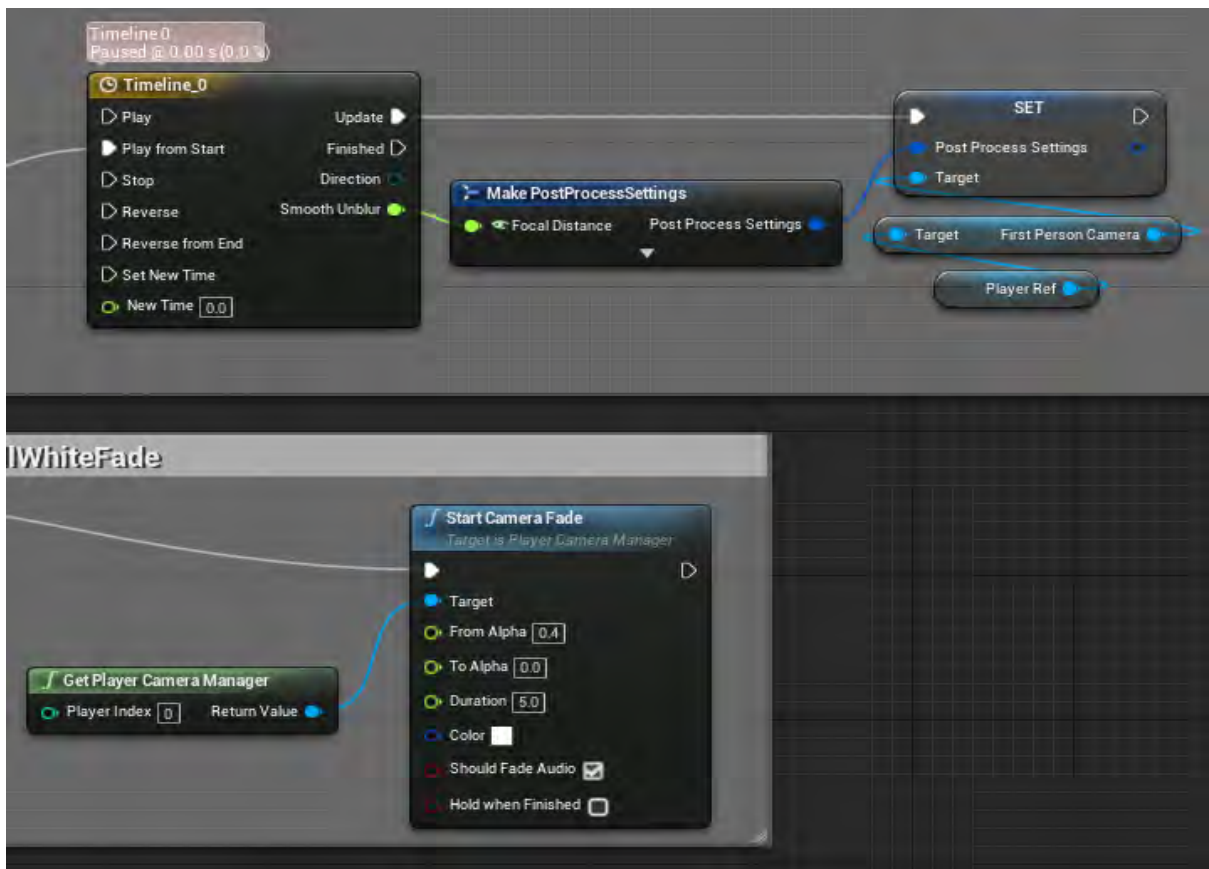
Εικόνα 380 Level4 My Blueprint

Το Event BeginPlay ενεργοποιείται μόλις δημιουργηθεί το επίπεδο και η λειτουργία που υλοποιεί είναι η απόκρυψη του κέρσορα του ποντικιού και θέτεται του Input Mode σε Game Only. Κατόπιν δημιουργείται Widget τύπου UMG\_HUD και προστίθεται στην οθόνη του παίκτη. Στη συνέχεια δημιουργείται αναφορά στον χαρακτήρα του παίκτη και αποθηκεύεται στην μεταβλητή PlayerRef (Εικόνα 381).



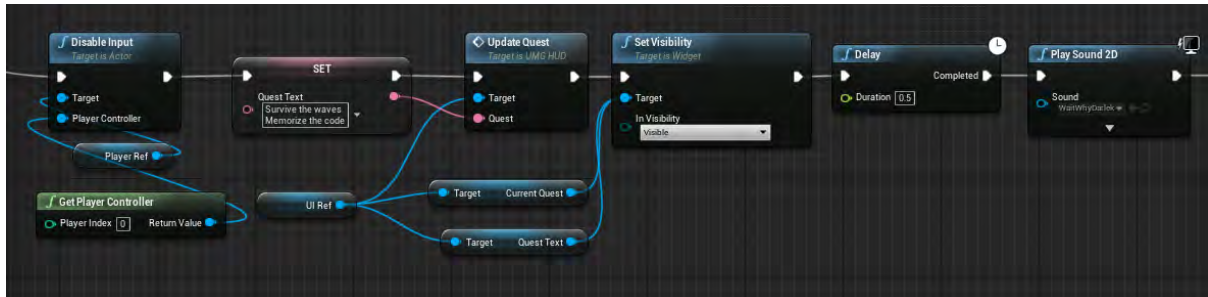
Εικόνα 381 Level4 Level Blueprint part 1

Όπως και στα προηγούμενα επίπεδα, τροποποιώντας παραμέτρους της κάμερας και χρησιμοποιώντας τον κόμβο Start Camera Fade δημιουργείται οπτικό εφέ (Εικόνα 382).



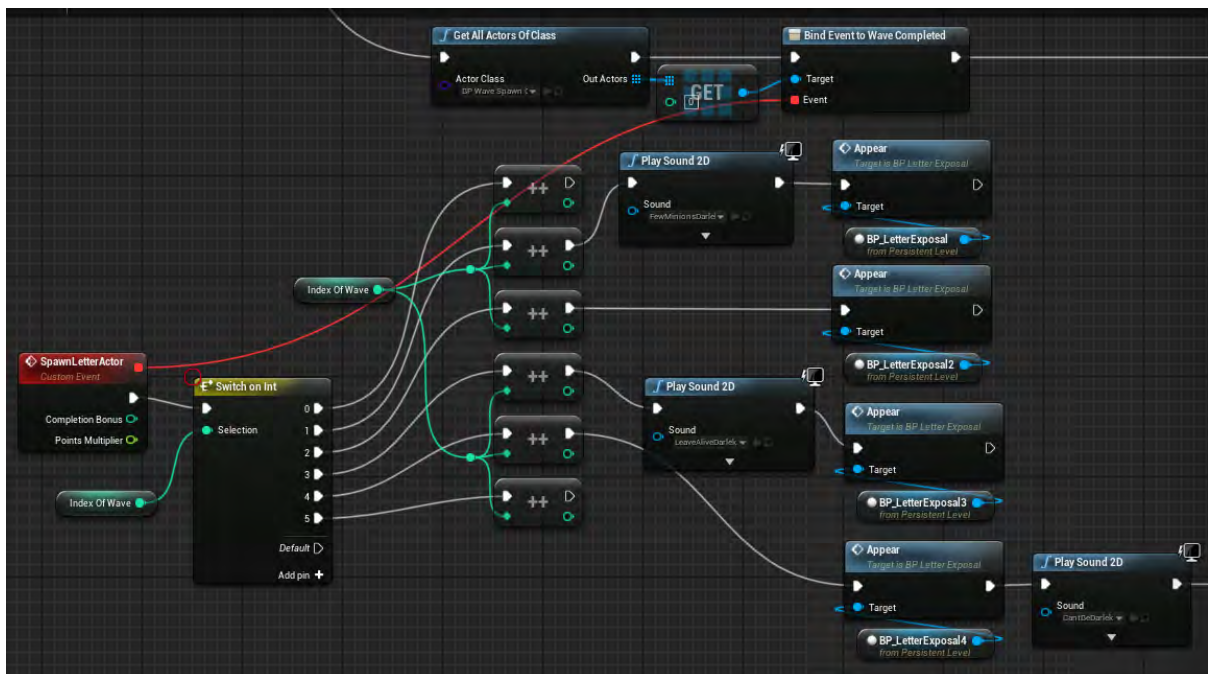
Εικόνα 382 Level4 Level Blueprint part 2

Στη συνέχεια απενεργοποιείται το Input του παίκτη, ενημερώνεται το κείμενο της αποστολής και γίνεται ορατό. Έπειτα αναπαράγονται επιλεγμένοι ήχοι (Εικόνα 383).



Εικόνα 383 Level4 Level Blueprint part 3

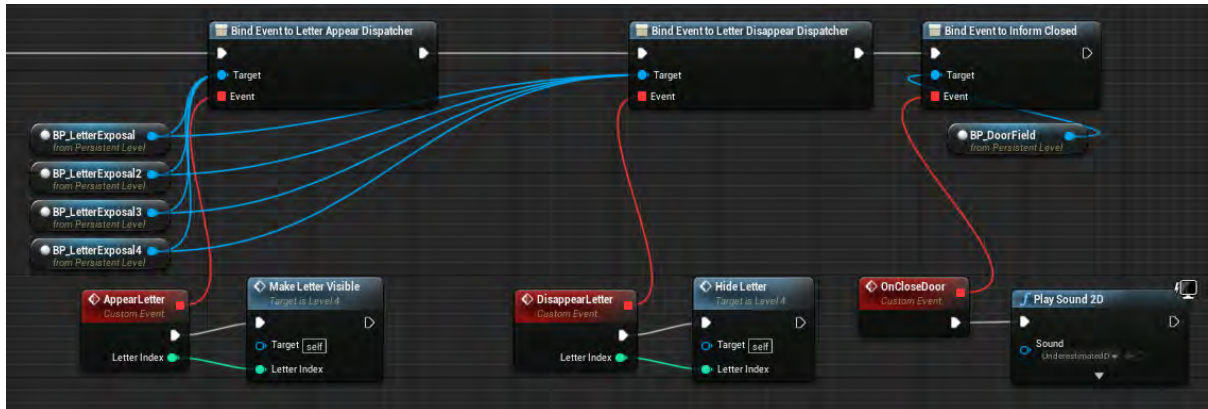
Κατόπιν για τον actor τύπου BP\_WaveSpawnController του επιπέδου δεσμεύεται η συνάρτηση SpawnLetterActor. Η συνάρτηση αυξάνει την τιμή της μεταβλητής IndexOfWave κάθε φορά που καλείται. Αναλόγως με την τιμή της μεταβλητής χρησιμοποιείται η συνάρτηση Appear από τον αντίστοιχο actor τύπου BP\_LetterExposal και εμφανίζεται ο actor (Εικόνα 384).



Εικόνα 384 Level4 Level Blueprint part 4

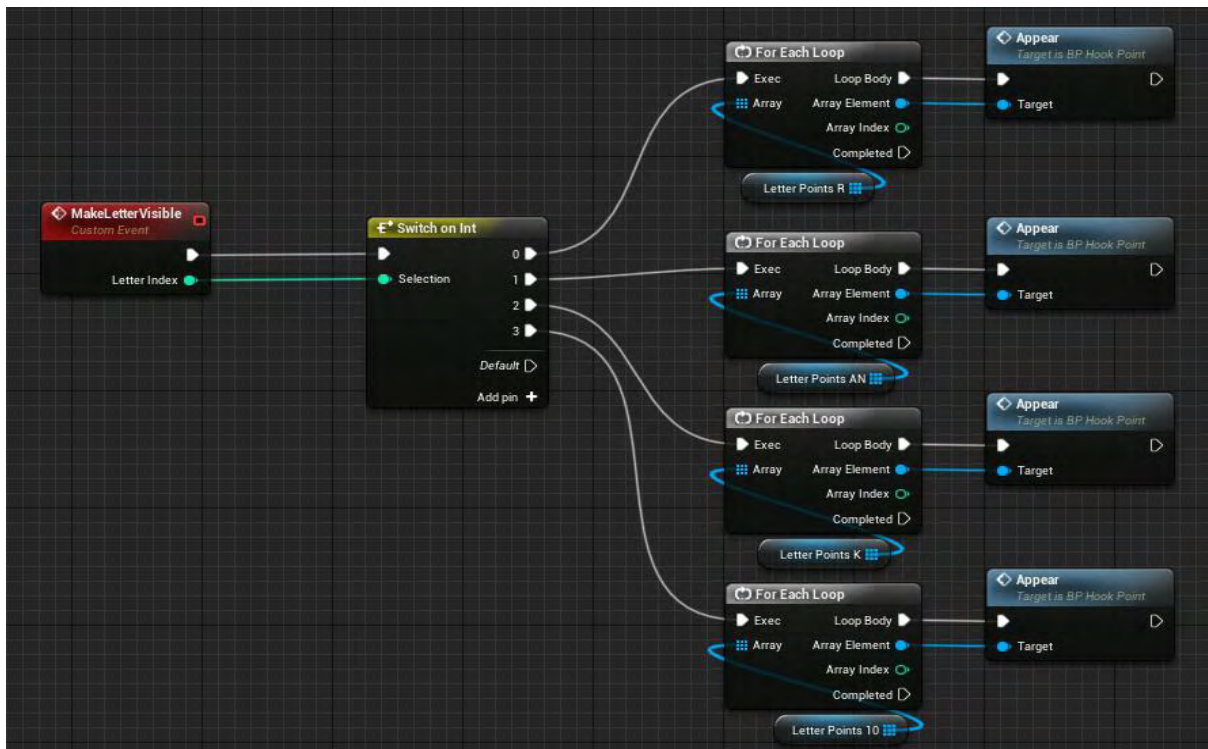


Κατόπιν δεσμεύονται οι συναρτήσεις `AppearLetter`, `DisappearLetter` και `OnCloseDoor` με τους `Event Dispatcher LetterAppearDispatcher`, `LetterDisappearDispatcher` και `InformClosed` αντίστοιχα. Η πρώτη καλεί την συνάρτηση `MakeLetterAppear`, η δεύτερη την συνάρτηση `HideLetter` και η τρίτη αναπαράγει επιλεγμένο ήχο (Εικόνα 385).



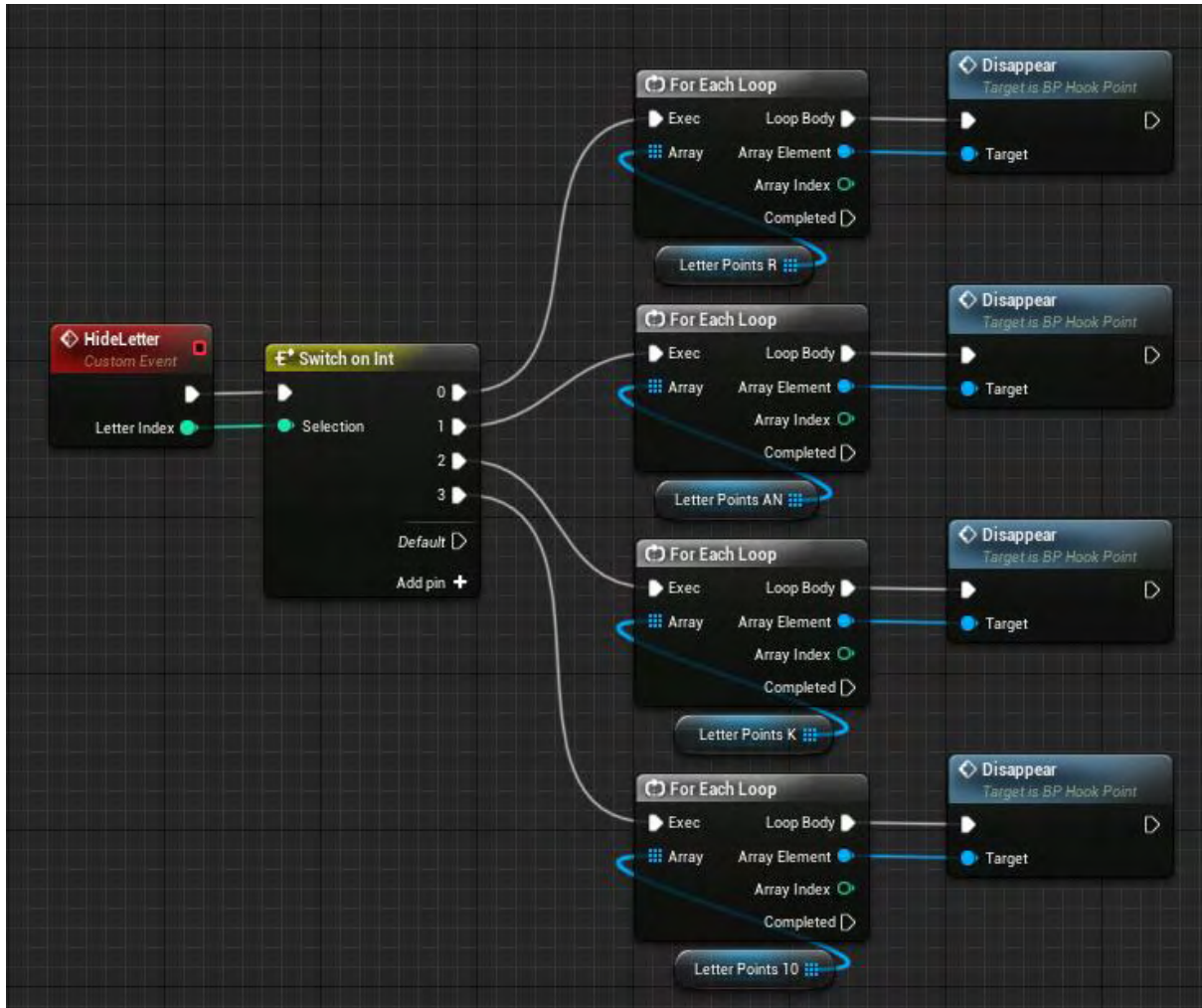
Εικόνα 385 Level4 Level Blueprint part 5

Η `MakeLetterVisible` καλεί τη συνάρτηση `Appear` για όλα τα στοιχεία του κάθε πίνακα, αναλόγως με την τιμή της μεταβλητής `LetterIndex`. Αυτό έχει σαν αποτέλεσμα την εμφάνιση ενός στοιχείου στο επίπεδο (Εικόνα 386).



Εικόνα 386 Level4 Level Blueprint part 6

Η HideLetter καλεί τη συνάρτηση Disappear για όλα τα στοιχεία του κάθε πίνακα αναλόγως με την τιμή της μεταβλητής LetterIndex. Αυτό έχει σαν αποτέλεσμα την απόκρυψη ενός στοιχείου στο επίπεδο (Εικόνα 387).



Εικόνα 387 Level4 Level Blueprint part 7



## ΚΕΦΑΛΑΙΟ 6 Συμπεράσματα

---

Η Unreal Engine με τις αναρίθμητες δυνατότητές της δίνει πλήρη ελευθερία στον σχεδιαστή του παιχνιδιού να αποτυπώσει και να υλοποιήσει οτιδήποτε μπορεί να φανταστεί χρησιμοποιώντας φυσικά και τους αντίστοιχους πόρους/συστήματα. Κάθε νεότερη έκδοση των μηχανών ανάπτυξης βιντεοπαιχνιδιών προσφέρει μεγαλύτερη αποδοτικότητα και περισσότερες νέες δυνατότητες.

Όταν η διαδικασία ανάπτυξης ενός βιντεοπαιχνιδιού υλοποιείται από ένα μόνο άτομο, είναι επακόλουθο να υπολείπεται η λεπτομέρεια σε κάποιους τομείς όπως ηχητικά εφέ, τρισδιάστατα μοντέλα, υφές υλικών κ.α. Για το λόγο αυτό, αφού η διαδικασία ανάπτυξης ενός βιντεοπαιχνιδιού είναι πολύπλοκη και χρονοβόρα, συνήθως συνεργάζονται ομάδες διαφόρων ειδικοτήτων για τη δημιουργία ενός καλαίσθητου και λειτουργικού βιντεοπαιχνιδιού.

Το «The Challenger» έχει μεγάλο περιθώριο για μελλοντικές επεκτάσεις, όπως προσθήκη πολλών ακόμη επιπέδων κλιμακούμενης δυσκολίας και περισσότερων στοιχείων puzzle, που δεν υλοποιήθηκαν στην παρούσα πτυχιακή εργασία λόγω των διαθέσιμων πόρων.

## ΒΙΒΛΙΟΓΡΑΦΙΑ

---

- 5pr1n6. (2019, Sept 14). *OpenGameArt.Org*. Ανάκτηση από [https://opengameart.org/content/sci-fi-background-noise?fbclid=IwAR2XUinvf9qsQZv2hD\\_5oQ\\_0Sj6F8V\\_6-Vcoaff2cA9o59QWmZLdGFeWeeE](https://opengameart.org/content/sci-fi-background-noise?fbclid=IwAR2XUinvf9qsQZv2hD_5oQ_0Sj6F8V_6-Vcoaff2cA9o59QWmZLdGFeWeeE)
- aghony, C. (2019, Jul 30). *FreeSound*. Ανάκτηση από [https://freesound.org/people/Contant\\_aghony/sounds/478179/?fbclid=IwAR2H4sSWyxN7qFGOKqtCPaYDOxG3b5gV0Mx10HMWmu0uaZi4cjMDlnc3Cw](https://freesound.org/people/Contant_aghony/sounds/478179/?fbclid=IwAR2H4sSWyxN7qFGOKqtCPaYDOxG3b5gV0Mx10HMWmu0uaZi4cjMDlnc3Cw)
- Andrade, A. (2015, November 5). *ProQuest*. Ανάκτηση από <https://search.proquest.com/openview/9c6c107bc8ab340f0c796af9f7c2b8d2/1?pq-origsite=gscholar&cbl=4477223>
- Charrieras, D., & Ivanova, N. (2016, April 1). *SAGE Journals*. σσ. 337–356. Ανάκτηση από <https://journals.sagepub.com/doi/abs/10.1177/0539018416642056>
- Christopoulou, E., & Xinogalos, S. (2017, December). *ResearchGate.net*. Ανάκτηση από Overview and Comparative Analysis of Game Engines for Desktop and: [https://www.researchgate.net/publication/322027338\\_Overview\\_and\\_Comparative\\_Analysis\\_of\\_Game\\_Engines\\_for\\_Desktop\\_and\\_Mobile\\_Devices](https://www.researchgate.net/publication/322027338_Overview_and_Comparative_Analysis_of_Game_Engines_for_Desktop_and_Mobile_Devices)
- Development, F. G. (2019, Jan 29). *OpenGameArt.Org*. Ανάκτηση από <https://opengameart.org/content/sci-fi-ambience-sfx?fbclid=IwAR27XsFjqZty2hoqYX8x8omBr7iGICFe-WZgsJYFn2VQMJlgaqMN-s1QoY4>
- Egenfeldt-Nielsen, S., Smith, J. H., & Tosca, S. P. (2020). *Understanding Video Games*. New York: Routledge.
- Epic Games. (2020, Juni 16). *Unreal Engine*. Ανάκτηση από <https://www.unrealengine.com/en-US/blog/a-first-look-at-unreal-engine-5>
- Epic Games. (χ.χ.). *Unreal Engine*. Ανάκτηση από [https://docs.unrealengine.com/en-US/Support/Builds/ReleaseNotes/4\\_24/index.html](https://docs.unrealengine.com/en-US/Support/Builds/ReleaseNotes/4_24/index.html)

- Epic Games. (χ.χ.). *Unrealengine*. Ανάκτηση από <https://docs.unrealengine.com/udk/Two/WebHome.html>
- Epic Games. (χ.χ.). *Unrealengine*. Ανάκτηση από <https://docs.unrealengine.com/udk/Three/SiteMap.html>
- Griffiths, M. (2002). The educational benefits of videogames. *Education and Health*. Ανάκτηση από [http://irep.ntu.ac.uk/id/eprint/15272/1/187769\\_5405%20Griffiths%20Publisher.pdf](http://irep.ntu.ac.uk/id/eprint/15272/1/187769_5405%20Griffiths%20Publisher.pdf)
- Jm\_at. (2018, Marz 22). *Freesound*.
- Kakky. (2017, Mar 29). *Unreal Engine*. Ανάκτηση από <https://www.unrealengine.com/marketplace/en-US/product/advancedmagicfx13?fbclid=IwAR0TpODrChpVZ7ReuPeGIANa2MpOemCz-yKTEImNRTsoCT-CGEZJk0tewQE>
- Kato, P. M. (2010, 6 1). *Sage Journals*. Ανάκτηση από [https://journals.sagepub.com/doi/abs/10.1037/a0019441?fbclid=IwAR0oFdXgg-C6T712jO1qu3g4\\_nPRldgtQg4G8gb\\_Lz3PslF9DOugDYfp0k&](https://journals.sagepub.com/doi/abs/10.1037/a0019441?fbclid=IwAR0oFdXgg-C6T712jO1qu3g4_nPRldgtQg4G8gb_Lz3PslF9DOugDYfp0k&)
- Lamarche, J. (2015, Febr 21). *MARTIANCRAFT*. Ανάκτηση από <https://martiancraft.com/blog/2015/02/disintegrating-baddies/>
- Lewis, J., Brown, D., Cranton, W., & Mason, R. (2011, November). *IEEE Xplore*. Ανάκτηση από <https://ieeexplore.ieee.org/abstract/document/6165430>
- Lewis, M., & Jacobson, J. (2002, January). *University of Nevada, Reno*. Ανάκτηση από <https://www.cse.unr.edu/~sushil/class/gas/papers/GameAIp27-lewis.pdf>
- Liaret. (2015, Febr 28). *YouTube*. Ανάκτηση από <https://www.youtube.com/watch?v=KHIZfy5O1O8>
- Lightbown, D. (2018, September 1). *Gamasutra*. Ανάκτηση από [https://www.gamasutra.com/blogs/DavidLightbown/20180109/309414/Classic\\_Tools\\_Retrospective\\_Tim\\_Sweeney\\_on\\_the\\_first\\_version\\_of\\_the\\_Unreal\\_Editor.php?fbclid=IwAR3spN1Ncfl5J-AknwTegHudBzD00UG4cYD\\_FxRtng7YI6wEOH5D6-0FB1Q](https://www.gamasutra.com/blogs/DavidLightbown/20180109/309414/Classic_Tools_Retrospective_Tim_Sweeney_on_the_first_version_of_the_Unreal_Editor.php?fbclid=IwAR3spN1Ncfl5J-AknwTegHudBzD00UG4cYD_FxRtng7YI6wEOH5D6-0FB1Q)
- Linietsky, J., & Manzur, A. (2007-2020). *Godot*. Ανάκτηση από <https://godotengine.org/>

- Paul, P. S., Goon, S., & Bhattacharya, A. (2012). *researchgate*. Ανάκτηση από [https://www.researchgate.net/profile/Abhishek\\_Bhattacharya4/publication/236590476\\_HISTORY\\_AND\\_COMPARATIVE\\_STUDY\\_OF\\_MODERN\\_GAME\\_ENGINES/links/00b7d518212aec50cb000000/HISTORY-AND-COMPARATIVE-STUDY-OF-MODERN-GAME-ENGINES.pdf](https://www.researchgate.net/profile/Abhishek_Bhattacharya4/publication/236590476_HISTORY_AND_COMPARATIVE_STUDY_OF_MODERN_GAME_ENGINES/links/00b7d518212aec50cb000000/HISTORY-AND-COMPARATIVE-STUDY-OF-MODERN-GAME-ENGINES.pdf)
- Ramadan, R., & Widyani, Y. (2013). *ReserchGate*. Ανάκτηση από [https://www.researchgate.net/profile/Yani\\_Widyani/publication/271548605\\_Game\\_development\\_life\\_cycle\\_guidelines/links/5756483708ae155a87b9d1ae.pdf](https://www.researchgate.net/profile/Yani_Widyani/publication/271548605_Game_development_life_cycle_guidelines/links/5756483708ae155a87b9d1ae.pdf)
- Saiqa, A., Capretz, L. F., & Faheem, A. (2016, November). *Western Libraries*. Ανάκτηση από <https://ir.lib.uwo.ca/cgi/viewcontent.cgi?referer=&httpsredir=1&article=1138&context=electricalpub>
- Unreal Engine. (χ.χ.). *Youtube*. Ανάκτηση από <https://www.youtube.com/watch?v=mig6EF17mR8>
- UnrealCG. (2017, Sept 2). *Youtube*. Ανάκτηση από <https://www.youtube.com/watch?v=hQeJETEUzE4>
- x86cam. (2013, Febr 9). *freesound*. Ανάκτηση από <https://freesound.org/people/x86cam/sounds/177769/?page=1&fbclid=IwAR2T-mRqtv37UO1tG0RHEviu21mj5v2weddbQqzS2sBWhSxr8JSXW4ZS2Lo#comments>
- Yuxiang, Z., Xiaowei, A., & Nongliang, S. (2020, March 31). *ScienceDirect*. Ανάκτηση από <https://www.sciencedirect.com/science/article/pii/S2096579620300553>
- Αναγνώστου, Κ. (2009). *BINTEOΠΑΙΧΝΙΔΙΑ Βιομηχανία και Ανάπτυξη*. Αθήνα: Κλειδάριθμος.
- Χατζής, Τ. Γ. (2006, Μάιος). *ΨΗΦΙΑ*. Ανάκτηση από <https://dspace.lib.uom.gr/handle/2159/1559?fbclid=IwAR3t3inCIWGYzCJJRrHdIFzAMqowp1YX-KXgOvAWvabkbEN7oVGAzR1Hcik>

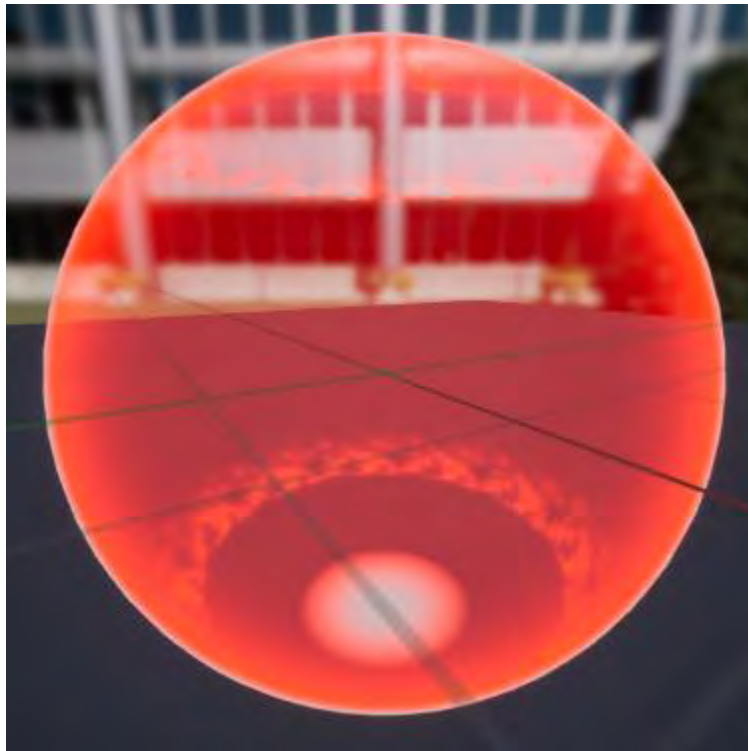
## ΠΑΡΑΡΤΗΜΑΤΑ

---

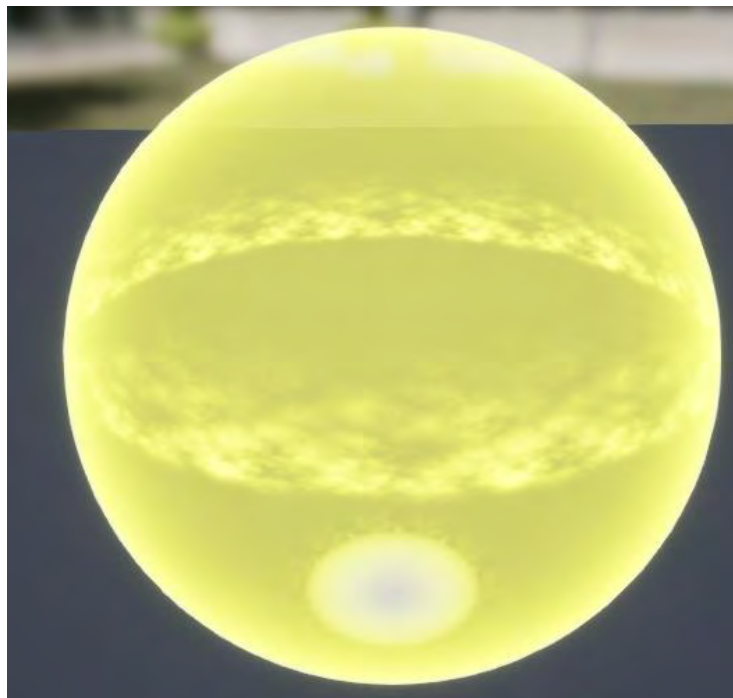


## ΠΑΡΑΡΤΗΜΑ Α - MATERIALS

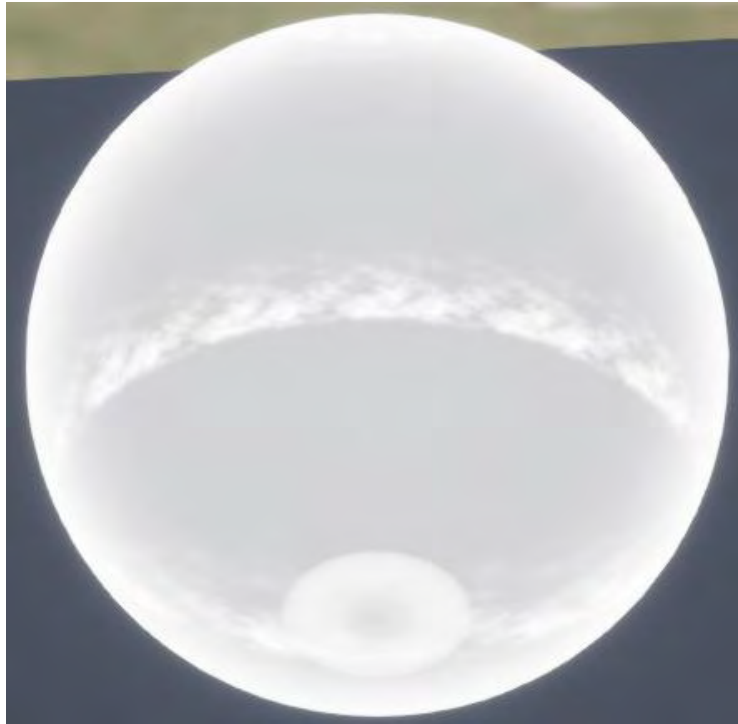
---



Εικόνα 388 Mat\_BlockShieldFinal



Εικόνα 389 MI\_BlockTPShieldFinal



**Εικόνα 390 MI\_DoorCode**



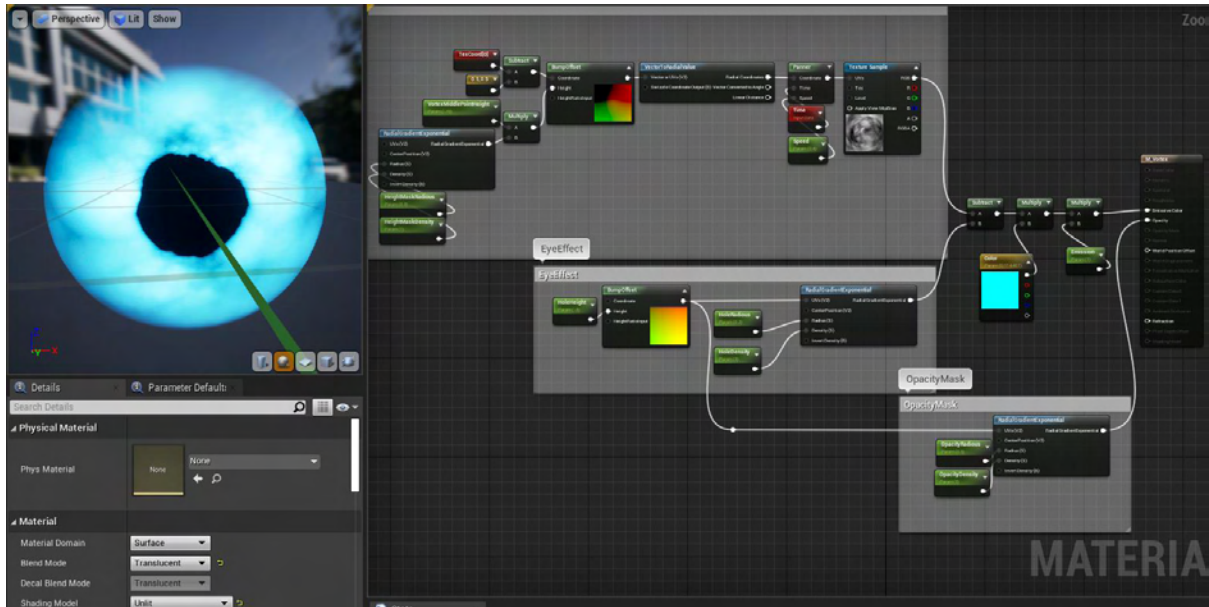
**Εικόνα 391 MI\_DisinRed**



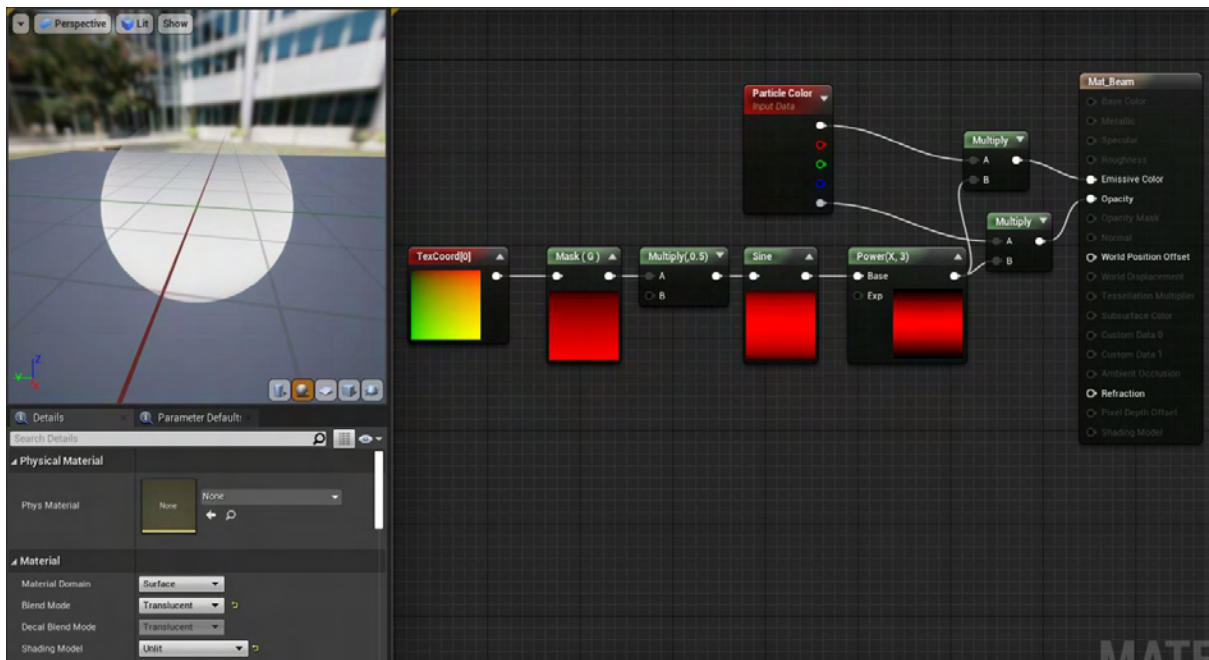
**Εικόνα 392 MI\_SimpleBlueDisinegrationE**



**Εικόνα 393 MI\_SimpleDisinVioletLE**

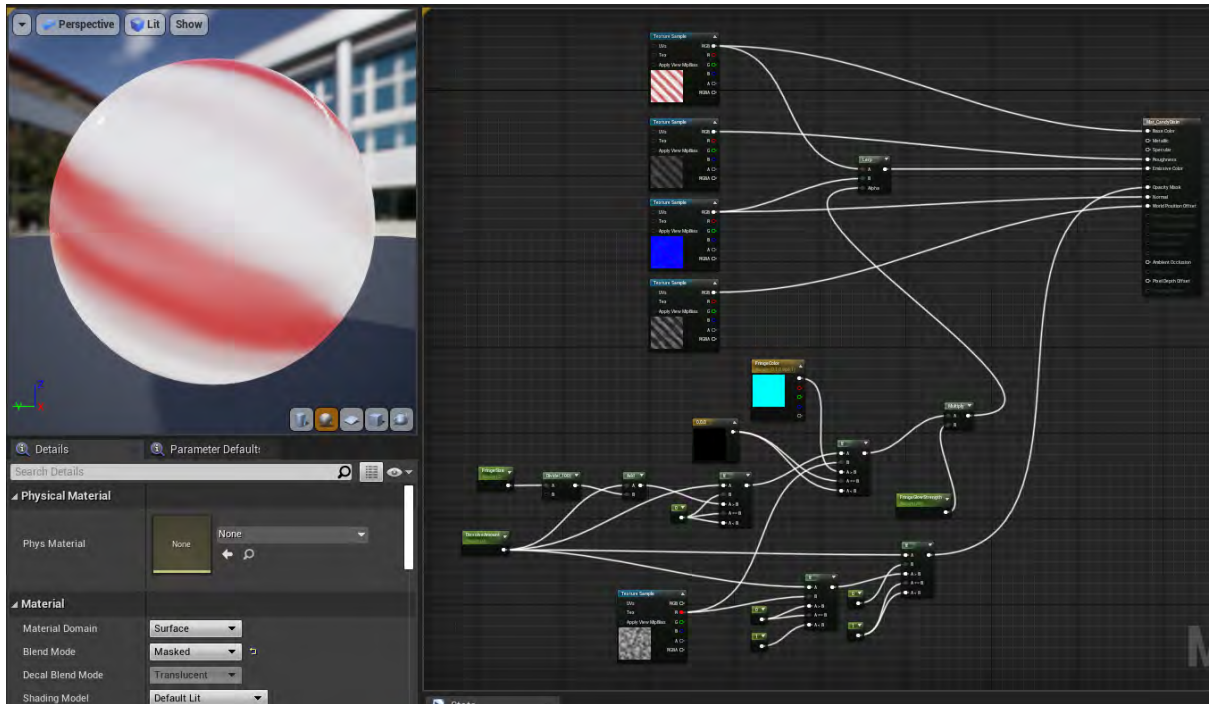


Εικόνα 394 M\_Vortex

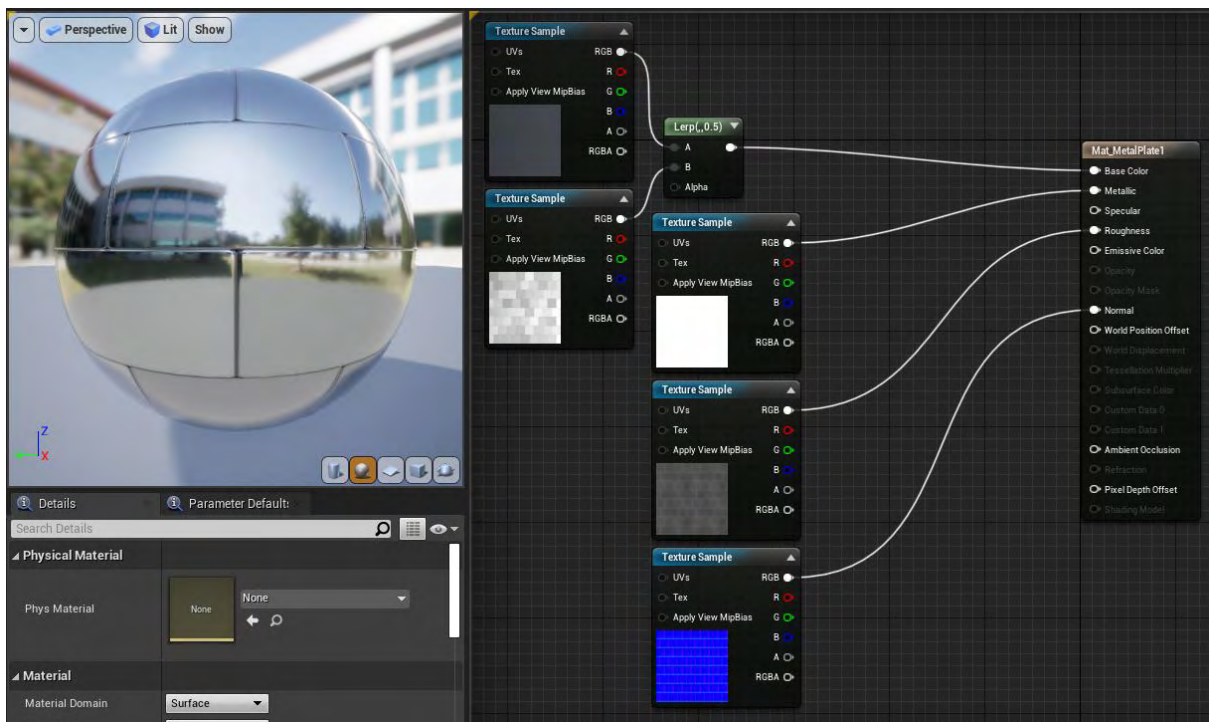


Εικόνα 395 Mat\_Beam



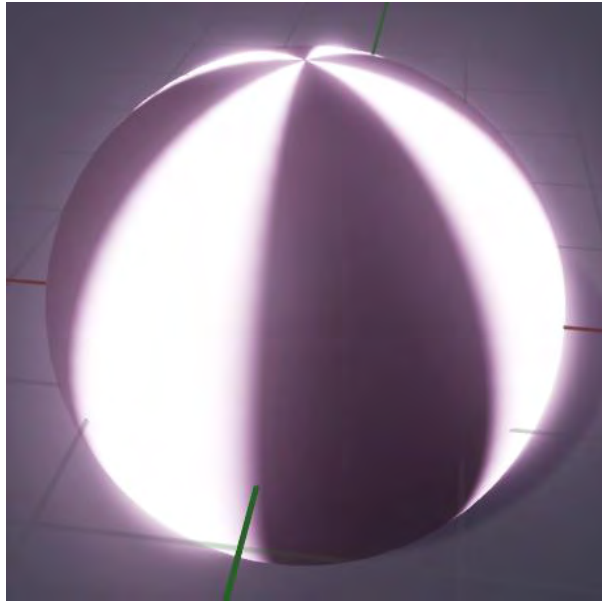


Εικόνα 396 Mat\_CandyDisin



Εικόνα 397 Mat\_MetalPlate1



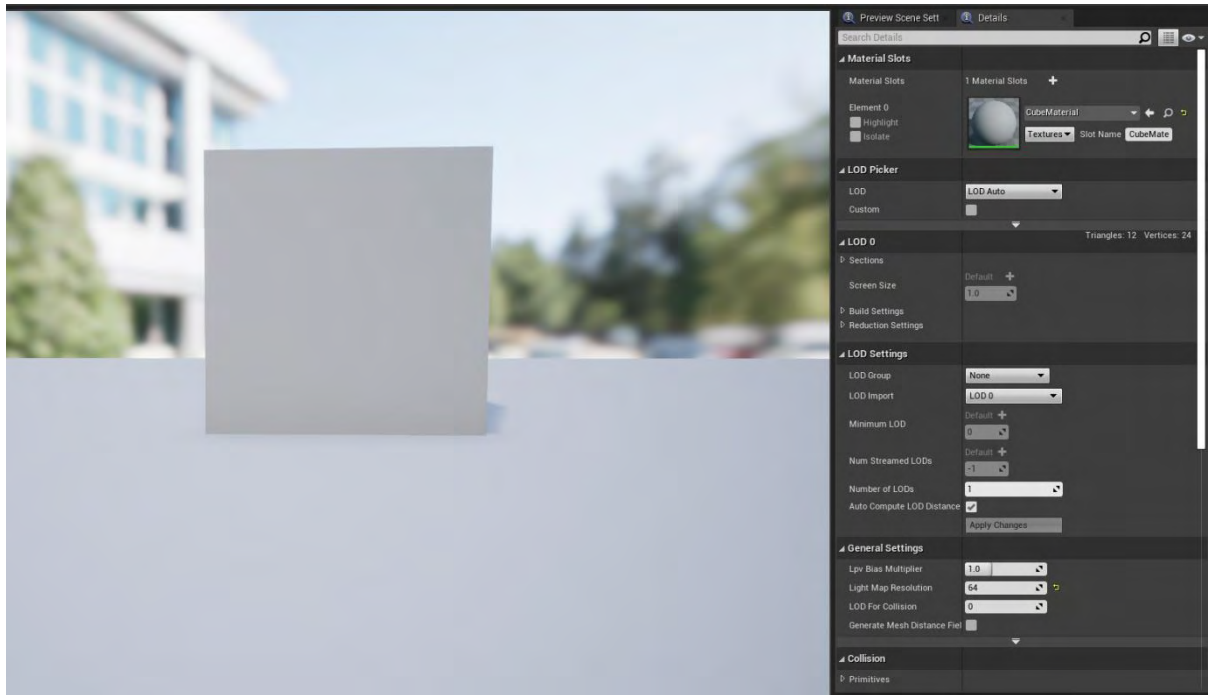


**Εικόνα 398 Mat\_SimpleGlowRotation**

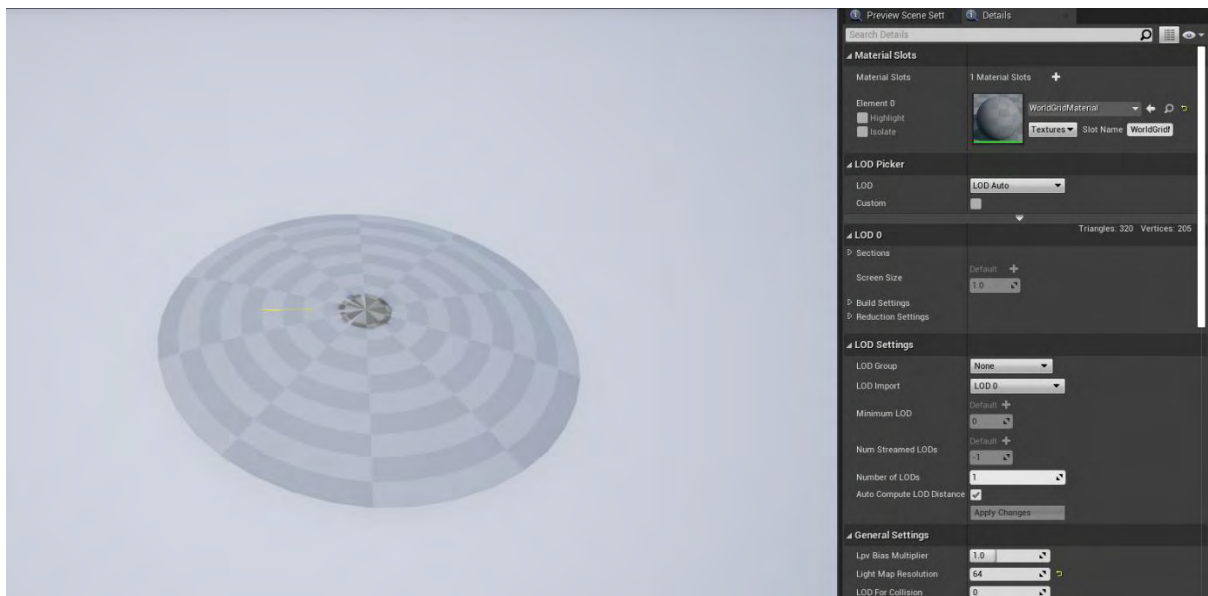


**Εικόνα 399 MI\_WallGlowRoomGreen**

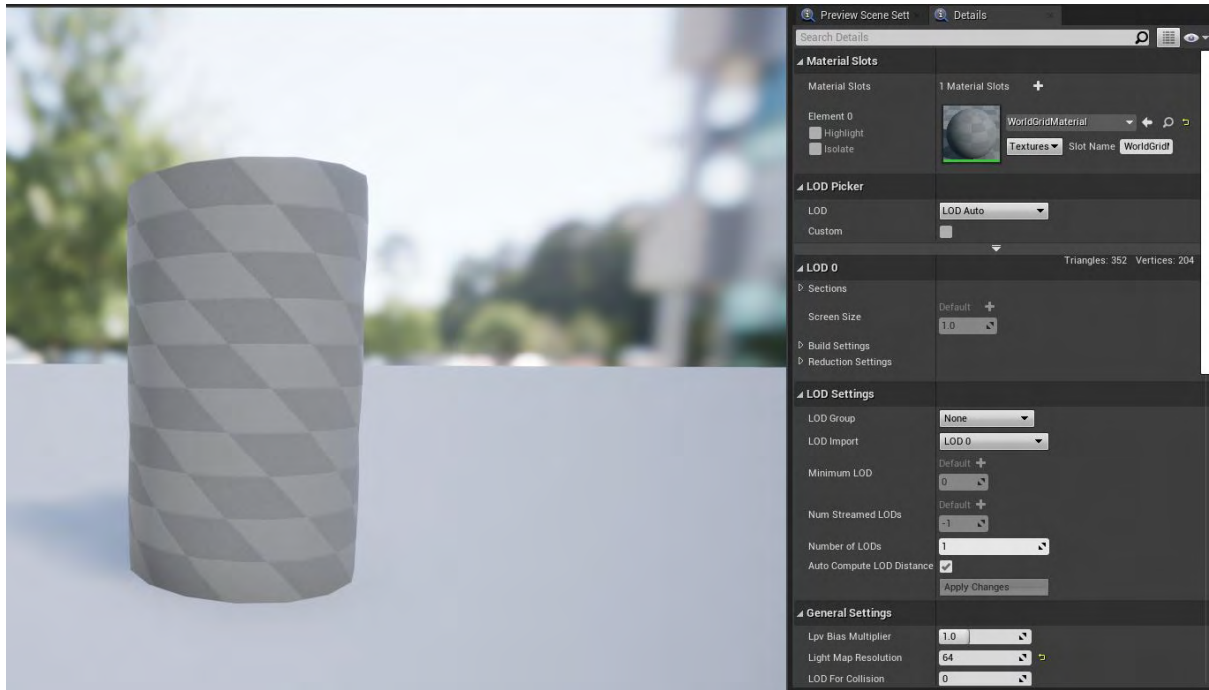
## ΠΑΡΑΡΤΗΜΑ Β - GEOMETRY



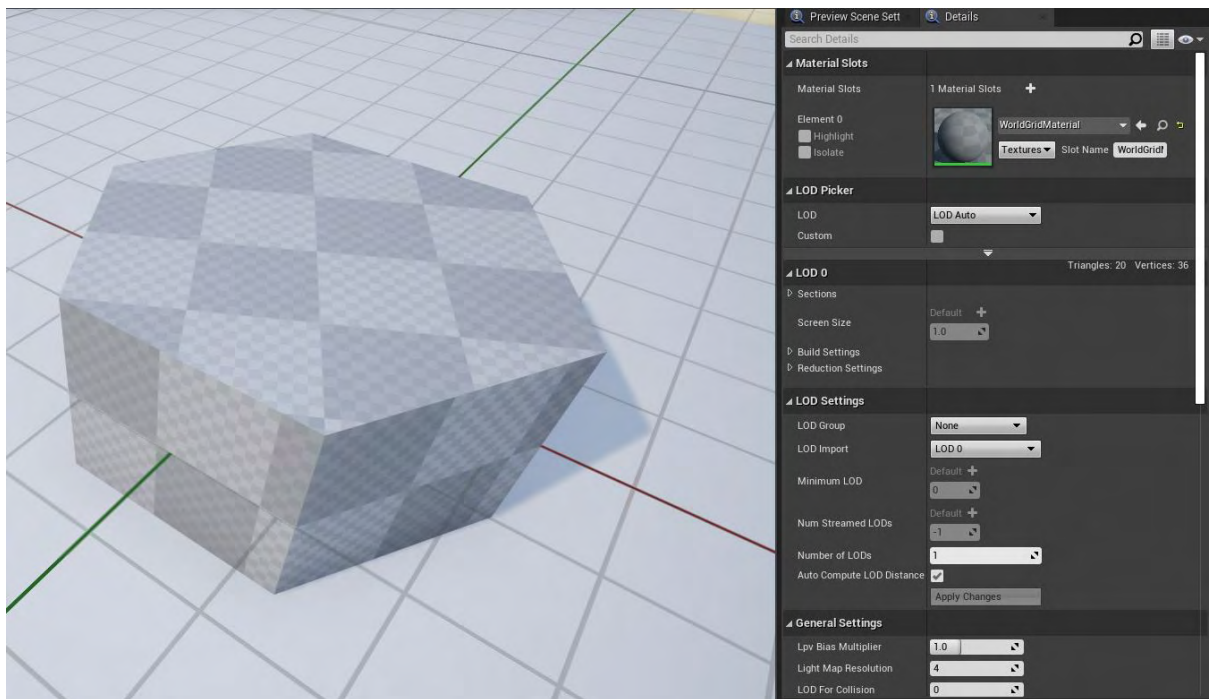
Εικόνα 400 1M\_Cube



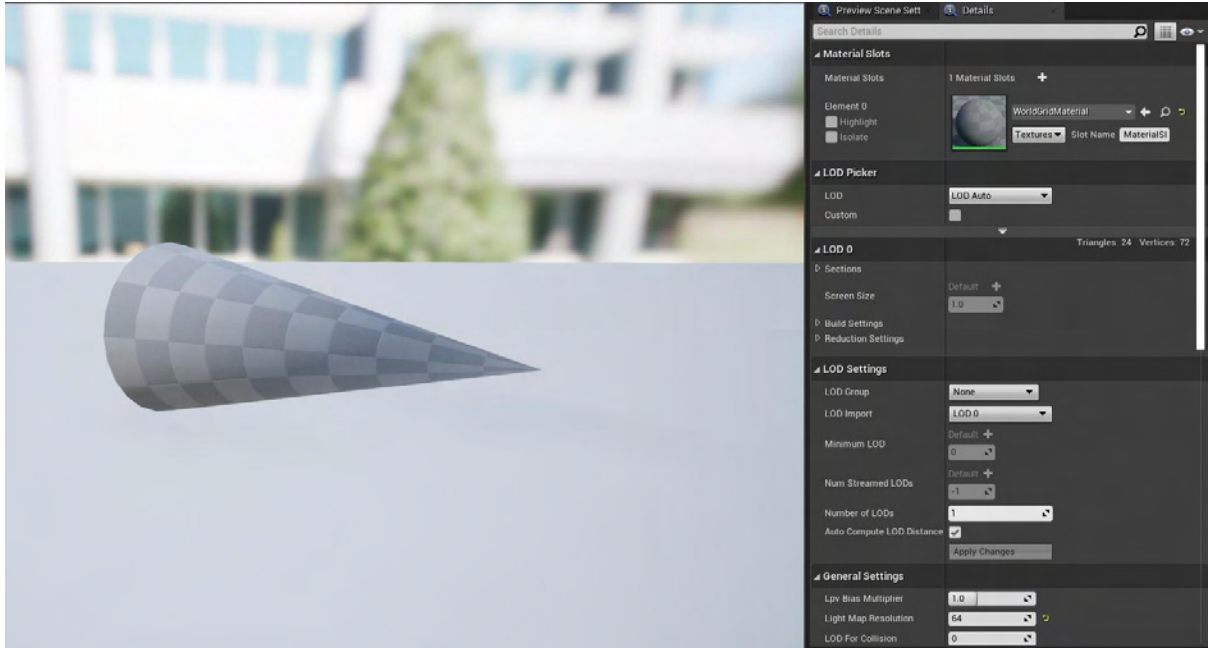
Εικόνα 401 SM\_circle\_inv2



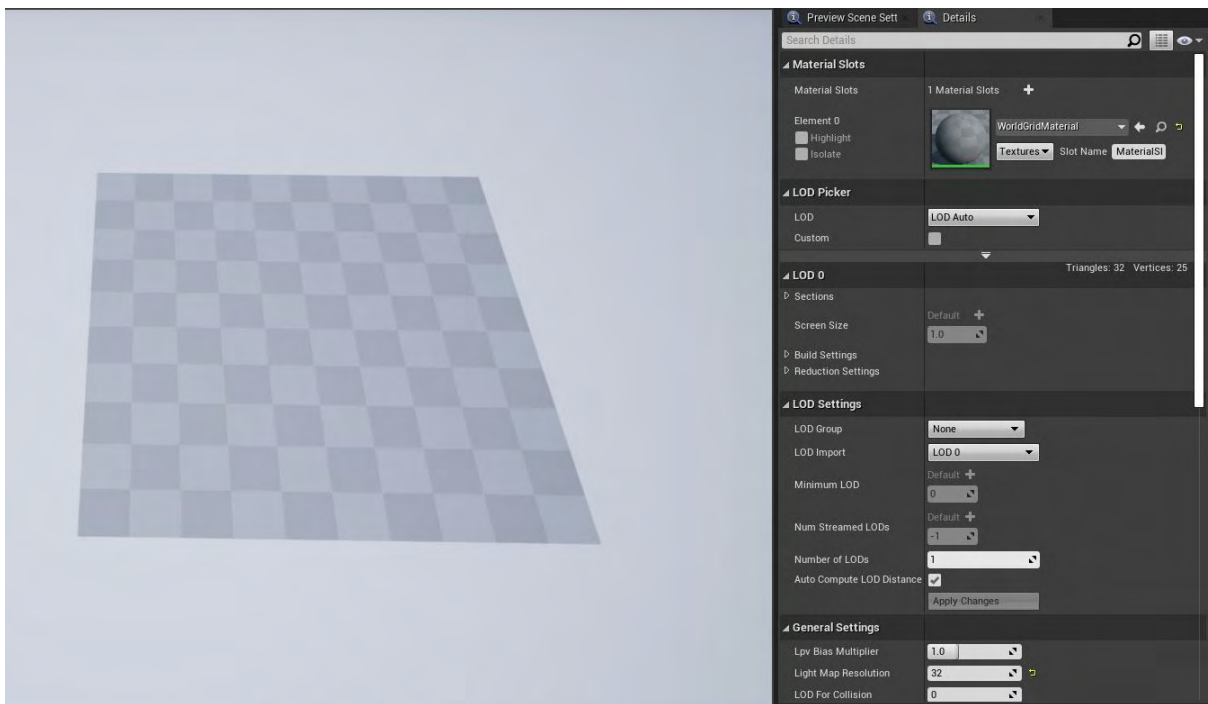
Εικόνα 402 SM\_cylinder\_lowPoly\_dist



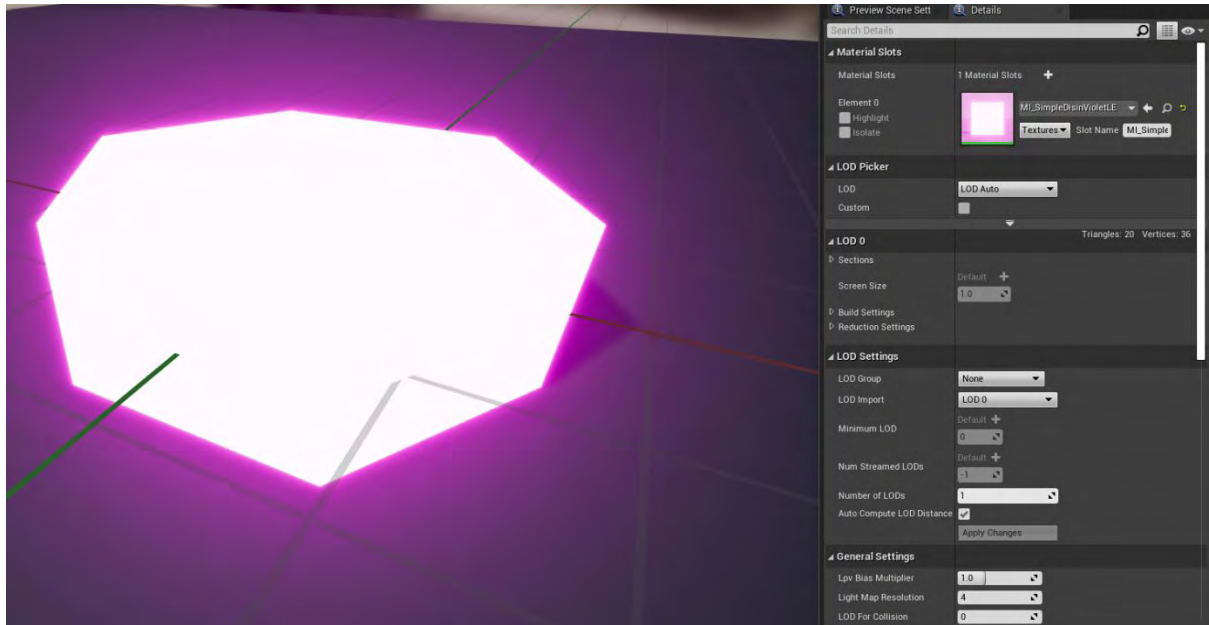
Εικόνα 403 SM\_Cylinder\_LP



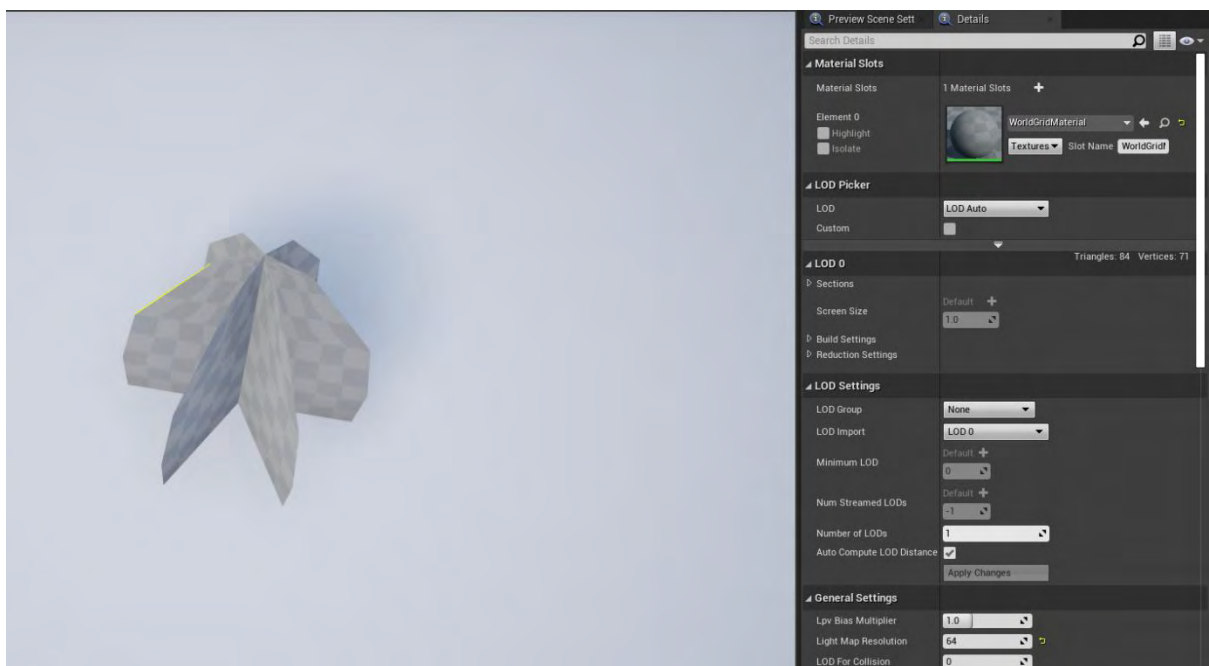
Εικόνα 404 SM\_lans



Εικόνα 405 SM\_Plane

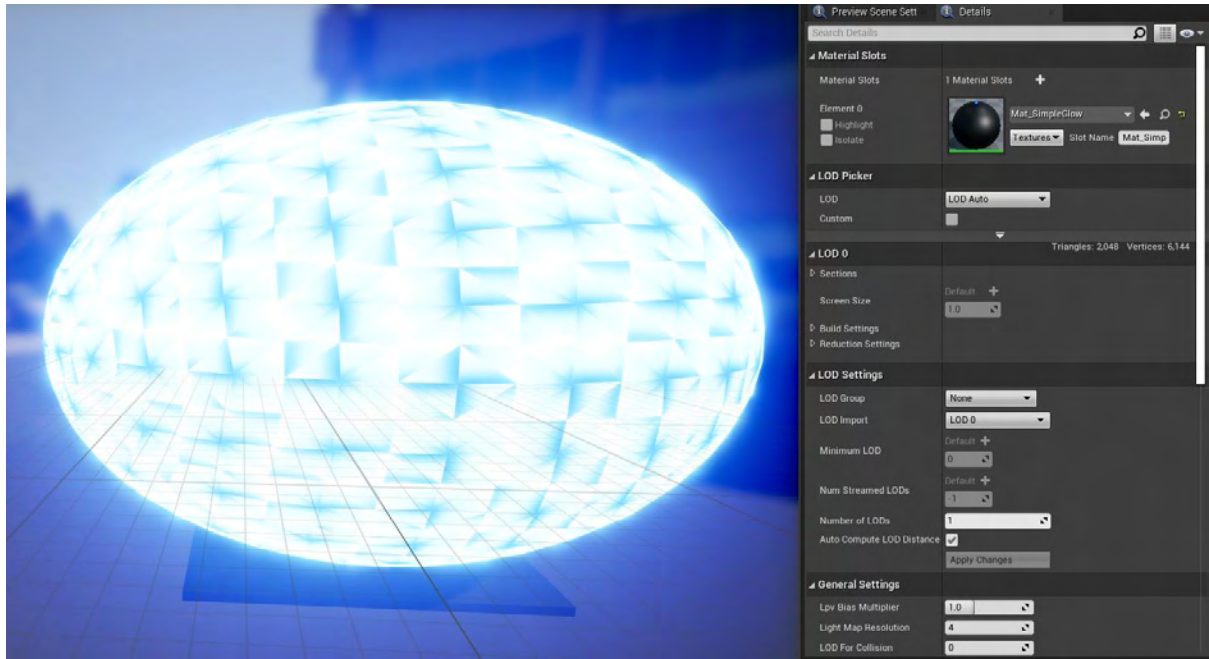


Εικόνα 406 SM\_PurplePlatform



Εικόνα 407 SM\_quads





Εικόνα 408 SphereRooms\_SM