



ΔΙΕΘΝΕΣ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΤΗΣ ΕΛΛΑΔΟΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ  
«ΔΗΜΙΟΥΡΓΙΑ 3D ΓΡΑΦΙΚΩΝ ΚΑΙ  
ΕΜΠΛΟΥΤΙΣΜΟ ΤΟΥΣ ΜΕ ΑΠΤΙΚΑ ΔΕΔΟΜΕΝΑ»



Της φοιτήτριας  
**ΡΑΓΚΑΤΣΗ ΓΕΩΡΓΙΑ**  
Αρ. Μητρώου: 144324

Επιβλέπων  
Όνοματεπώνυμο **ΚΟΚΚΩΝΗΣ**  
**ΓΕΩΡΓΙΟΣ**  
Βαθμίδα **ΕΠΙΚΟΥΡΟΣ**  
**ΚΑΘΗΓΙΤΗΣ**

**Ημερομηνία ΙΑΝΟΥΑΡΙΟΣ 2026**

Τίτλος Δ.Ε. Δημιουργία 3D γραφικών και εμπλουτισμός τους με απτικά δεδομένα

Κωδικός Δ.Ε. 23357

Όνοματεπώνυμο φοιτητή/των Ραγκάτση Γεωργία

Όνοματεπώνυμο εισηγητή Κοκκώνης Γεώργιος

Ημερομηνία ανάληψης Δ.Ε. 21/12/2023

Ημερομηνία περάτωσης Δ.Ε. 23/01/2026

*Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.*

*Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία της φοιτήτριας Ραγκάτση Γεωργίας που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.*

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

## Πρόλογος

Η παρούσα εργασία εκπονήθηκε στο πλαίσιο της σύγχρονης τεχνολογικής εξέλιξης στον τομέα των διαδραστικών συστημάτων και εστιάζει στη διερεύνηση της απτικής ανάδρασης και της αίσθησης της αφής σε εικονικά περιβάλλοντα. Η υλοποίηση βασίζεται στη χρήση σύγχρονων εργαλείων ανάπτυξης και απτικής τεχνολογίας, συνδυάζοντας μια μηχανή γραφικών με μια απτική συσκευή μηχανικού βραχίονα, με στόχο τη δημιουργία ενός λειτουργικού, κατανοητού και επεκτάσιμου πειραματικού συστήματος. Μέσα από την ανάπτυξη και την αξιολόγηση διαδραστικών εφαρμογών, η εργασία επιχειρεί να αναδείξει τον ρόλο της απτικής ανάδρασης ως μέσο ενίσχυσης του ρεαλισμού και της εμπύθισης του χρήστη σε ψηφιακά περιβάλλοντα.

## Περίληψη

Η παρούσα εργασία πραγματεύεται την ανάπτυξη ενός διαδραστικού εικονικού περιβάλλοντος με απτική ανάδραση, με στόχο τη διερεύνηση της αίσθησης της αφής σε εικονικά περιβάλλοντα μέσω σύγχρονων τεχνολογιών προσομοίωσης. Η υλοποίηση πραγματοποιήθηκε με τη χρήση της μηχανής γραφικών Unreal Engine σε συνδυασμό με τον μηχανικό βραχίονα 3D Systems Touch, ο οποίος επιτρέπει στον χρήστη να λαμβάνει απτικά ερεθίσματα κατά την αλληλεπίδραση με εικονικά αντικείμενα.

Στο πλαίσιο της εργασίας σχεδιάστηκε και αναπτύχθηκε ένα σύνολο πειραματικών επιπέδων, καθένα από τα οποία εστιάζει σε διαφορετικές πτυχές της απτικής αλληλεπίδρασης. Παράλληλα, υλοποιήθηκαν μηχανισμοί πλοήγησης, αλληλεπίδρασης και απτικής ανάδρασης, καθώς και σύστημα μενού για τη διαχείριση της ροής της εφαρμογής.

Τέλος, η εφαρμογή αξιολογήθηκε με τη συμμετοχή πραγματικών χρηστών, επιτρέποντας την αποτίμηση της χρηστικότητας, της λειτουργικότητας και της συνολικής εμπειρίας απτικής αλληλεπίδρασης.

«Δημιουργία 3D γραφικών και εμπλουτισμό τους με απτικά  
δεδομένα»

(Synthesize a 3D environment with embedded systems for tactile  
interaction)

«Ραγκάτση Γεωργία»

(Ragkatsi Georgia)

## **Abstract**

This thesis focuses on the development of an interactive virtual environment with haptic feedback, aiming to explore the sense of touch within virtual environments through modern simulation technologies. The implementation was carried out using the Unreal Engine graphics engine in combination with the 3D Systems Touch haptic device, which enables users to receive tactile feedback while interacting with virtual objects.

Within the scope of the thesis, a set of experimental levels was designed and developed, each targeting different aspects of haptic interaction. In parallel, navigation, interaction, and haptic feedback mechanisms were implemented, along with a menu system that supports the overall flow and usability of the application. Particular emphasis was placed on the correct handling of collisions and the consistent integration of haptic responses within the virtual environment.

Finally, the application was evaluated through user testing with real participants, to assess the usability, the functionality, and the overall quality of the haptic interaction experience. The results demonstrate that the proposed system successfully supports meaningful haptic interaction and highlights the potential of combining game engines with haptic technologies for interactive and educational applications.

## **Ευχαριστίες**

Θα ήθελα να ευχαριστήσω θερμά τον καθηγητή μου που μου έδωσε την ευκαιρία να αναλάβω και να παρουσιάσω αυτό το έργο ενώπιον του τμήματος. Όπως επίσης και την οικογένεια και τους φίλους μου που με ενθάρρυναν να συνεχίσω το έργο, όσο μεγάλη και αν ήταν η πρόκληση.

# Περιεχόμενα

Πρόλογος.....	iv
Περίληψη.....	v
Abstract .....	vi
Ευχαριστίες .....	vii
Περιεχόμενα .....	viii
Κατάλογος Εικόνων .....	xii
Κατάλογος Σχημάτων .....	xiii
Κατάλογος Πινάκων.....	xiv
Συντομογραφίες.....	xv
Κεφάλαιο 1ο:     Απτική Τεχνολογία και Ανθρώπινη – Υπολογιστική Αλληλεπίδραση.....	1
1.1    Εισαγωγή.....	1
1.2    Απτική Ανάδραση και Εικονικά Περιβάλλοντα.....	1
1.3    Τεχνολογίες Προσομοίωσης και Ανθρώπινη Αλληλεπίδραση.....	1
1.4    Στόχος, Κίνητρο και Σημασία της Εργασίας.....	2
1.5    Επίλογος.....	2
Κεφάλαιο 2ο:     3D Systems και Unreal Engine 4.....	3
2.1    3D Systems και ο μηχανικός βραχίονας Touch .....	3
2.1.1   Εισαγωγή.....	3
2.1.2   Ο Μηχανικός βραχίονας.....	3
2.1.3   Λειτουργία.....	4
2.1.4   Εφαρμογή της συσκευής .....	5
2.1.5   Σύνδεση με μηχανή γραφικών και Παραδείγματα εφαρμογής της ανάπτυξης συστήματος 5	
2.1.6   Συμπέρασμα.....	6
2.2    Η μηχανή γραφικών Unreal Engine 4.27 .....	6
2.2.1   Εισαγωγή.....	6
2.2.2   Unreal Engine.....	6
2.2.3   Unreal Engine 4.27 .....	6
2.2.4   Τρόποι ανάπτυξης κώδικα.....	7
2.3    Επεκτάσιμα και Πρόσθετα (Plugins).....	8
2.3.1   Εισαγωγή.....	8
2.3.2   Εξατομικευμένες Απτικές Συσκευές και SDK τρίτων κατασκευαστών.....	8

2.3.3	Ζητήματα Απόδοσης κα Πολυπλοκότητας Επεξεργασίας .....	8
2.3.4	Ανάδραση μέσω Απόδοσης και Ιδιοτήτων Υλικών.....	8
2.3.5	Συμπεράσματα.....	9
2.4	Ενσωμάτωση του 3DSystems Touch με το Unreal Engine 4.27.....	9
2.4.1	Εισαγωγή.....	9
2.4.2	Τρόπος σύνδεσης.....	9
2.4.3	Απαιτήσεις Λογισμικού και Ρυθμίσεις.....	10
2.4.4	Συμπεράσματα.....	11
2.5	Περιβάλλον Ανάπτυξης.....	12
2.5.1	Αρχικοποίηση Έργου.....	12
2.5.2	Περιβάλλον Μηχανής (Unreal Engine).....	14
2.5.3	Αντικείμενα του Unreal Engine (Components).....	17
2.5.4	Το Σύστημα Blueprint (Blueprint Editor).....	20
2.5.5	Μενού .....	22
2.5.6	Ρυθμίσεις (Project Settings).....	23
2.6	Επίλογος.....	25
Κεφάλαιο 3ο:	Ανάπτυξη εφαρμογής με Unreal Engine 4.....	26
3.1	Εισαγωγή.....	26
3.2	Περιπτώσεις Χρήσης και Επίπεδα Απτικής Αλληλεπίδρασης.....	26
3.3	Δημιουργία Τρισδιάστατων Χώρων.....	26
3.4	Επίπεδο 1 <sup>ο</sup> : Πειράματα βάση ιδιότητας υλικών .....	28
3.4.1	Εισαγωγή .....	28
3.4.2	BP_HapticTrigger: Components .....	28
3.4.3	BP_HapticTrigger: Construction Script .....	29
3.4.4	BP_HapticTrigger: Physical to Haptic Material Function.....	30
3.4.5	Physical Material .....	32
3.4.6	Δημιουργία .....	33
3.4.7	Ανάθεση Physical Material.....	33
3.4.8	Ανάλυση των υλικών που χρησιμοποιήθηκαν.....	34
3.5	Επίπεδο 2 <sup>ο</sup> : Πειράματα βάση υφής επιφανειών (Texture) .....	36
3.5.1	Εισαγωγή.....	36
3.5.2	Δημιουργία Ειδικών Επιφανειών.....	36
3.5.3	Διαδικασία που ακολουθήθηκε στο Blender 2.79 .....	37
3.5.4	Εισαγωγή (Import) FBX στο Unreal Engine.....	40
3.5.5	Παραμετροποίηση ειδικών επιφανιών.....	40

3.5.6	Τοποθέτηση εντός του εικονικού περιβάλλοντος.....	41
3.6	Επίπεδο 3 <sup>ο</sup> : Αναγνώριση αντικειμένων (Shapes).....	42
3.6.1	Εισαγωγή.....	42
3.6.2	BP_MaterialButtonChange: Components.....	43
3.6.3	Τερματισμός της Εφαρμογής Μέσα από το επίπεδο.....	45
3.6.4	Εσωτερικός Φωτισμός.....	46
3.6.5	Πυροδοτητές Επιπέδου (Trigger Box).....	47
3.7	Παράθυρα οδηγιών και Επικεφαλίδες.....	48
3.8	Κεντρικό Μενού Εφαρμογής.....	49
3.8.1	Εισαγωγή.....	49
3.8.2	Δημιουργία Κεντρικού Μενού.....	49
3.9	Χαρακτήρας (Playable Character).....	51
3.9.1	Εισαγωγή.....	51
3.9.2	Πρόβλημα και Λύση.....	51
3.9.3	Τροποποιήσεις.....	51
3.9.4	Τεχνικές Ρυθμίσεις.....	53
3.9.5	Δημιουργία και Παραμετροποίηση του ThesisHapticDeviceActor Blueprint.....	54
3.9.6	Αρχικοποίηση Θέσης Παίκτη.....	56
3.10	Οι ολοκληρωμένες αίθουσες των πειραμάτων.....	57
3.11	Επίλογος.....	58
Κεφάλαιο 4ο:	Πακετάρισμα.....	59
4.1	Εισαγωγή.....	59
4.2	Project Settings.....	59
4.2.1	Project – Description.....	59
4.2.2	Project – Maps & Modes.....	60
4.2.3	Project – Packaging.....	60
4.2.4	Project – Packaging: Build Configuration.....	62
4.2.5	Project – Supported Platforms & Target Hardware.....	62
4.2.6	Εκτέλεση της διαδικασίας πακεταρίσματος.....	62
4.3	Επίλογος.....	63
Κεφάλαιο 5ο:	Ροή της εφαρμογής.....	64
5.1	Πειράματα βάση ιδιότητας των υλικών.....	65
5.1.1	Πείραμα Σκληρότητας.....	66
5.1.2	Πείραμα Ολισθηρότητας.....	66
5.1.3	Πείραμα Διαπερατότητας.....	67

5.2	Πειράματα βάση υφής επιφανειών (Texture).....	68
5.3	Πειράματα αναγνώρισης αντικειμένων (Shapes).....	69
Κεφάλαιο 6ο:	Αξιολόγηση Εφαρμογής .....	71
6.1	Αξιολόγηση Εφαρμογής με Πραγματικούς χρήστες.....	71
6.1.1	Δημογραφικά στοιχεία δείγματος.....	71
6.2	Πλοήγηση στο Εικονικό Περιβάλλον .....	71
6.3	Ανάλυση Αξιολόγησης ανά Πείραμα.....	71
6.3.1	Αίσθηση Σκληρότητας (Hardness).....	71
6.3.2	Αίσθηση Ολισθηρότητας (Friction).....	72
6.3.3	Αίσθηση Διαπερατότητας (Density).....	73
6.3.4	Αίσθηση Υφής Επιφανειών (Texture).....	73
6.3.5	Αναγνώριση Σχημάτων μέσω της Ψηλάφησης (Shapes) .....	73
6.4	Συμπέρασμα Αξιολόγησης.....	74
Κεφάλαιο 7ο:	Προτάσεις Βελτίωσης και Συμπεράσματα Εργασίας .....	75
7.1	Προτάσεις Εμπλουτισμού και Βελτίωσης.....	75
7.2	Συμπεράσματα εργασίας .....	75
ΒΙΒΛΙΟΓΡΑΦΙΑ.....		76
ΠΑΡΑΡΤΗΜΑ Α : Επιπλέον Σύνδεσμοι.....		81

## Κατάλογος Εικόνων

Εικόνα 2.1: Ο Μηχανικός βραχίονας 3D Systems Touch.....	3
Εικόνα 2.2: Εγκατάσταση Πρόσθετου στη μηχανή Unreal.....	10
Εικόνα 2.3: Ενεργοποίηση Πρόσθετου .....	11
Εικόνα 2.4: Αρχική οθόνη Unreal Engine.....	12
Εικόνα 2.5: Επιλογή πρότυπου .....	13
Εικόνα 2.6: Αρχικές ρυθμίσεις έργου .....	13
Εικόνα 2.7: Περιβάλλον μηχανής Unreal Engine .....	14
Εικόνα 2.8: Παράθυρο Προβολής.....	15
Εικόνα 2.9: World Outliner .....	16
Εικόνα 2.10: Details .....	17
Εικόνα 2.11: World Settings .....	17
Εικόνα 2.12: Components .....	17
Εικόνα 2.13: Φωτισμός .....	19
Εικόνα 2.14: Πυροδοτητές.....	20
Εικόνα 2.15: Blueprint Editor .....	21
Εικόνα 2.16: UMG UI Designer .....	23
Εικόνα 2.17: Project Settings .....	24
Εικόνα 3.1: DefaultSpace_Level.....	27
Εικόνα 3.2: BP_HapticTrigger Viewport.....	28
Εικόνα 3.3α: BP_HapticTrigger Construction Script.....	29
Εικόνα 3.3β: BP_HapticTrigger Construction Script.....	30
Εικόνα 3.4: Physical to Haptic Material Function .....	31
Εικόνα 3.5: BP_HapticTrigger Instance μέσα στο επίπεδο.....	32
Εικόνα 3.6: Physical Material Editor .....	33
Εικόνα 3.7: Material.....	33
Εικόνα 3.8: Material Instance.....	33
Εικόνα 3.9: Πειράματα Σκληρότητας .....	34
Εικόνα 3.10: Πειράματα Ολισθηρότητας.....	35
Εικόνα 3.11: Πειράματα Διαπερατότητας.....	36
Εικόνα 3.12: Πληροφορίες υφής του Material .....	37
Εικόνα 3.13: Hight Map.....	37
Εικόνα 3.14: Object Mode .....	38
Εικόνα 3.15: Edit Mode – Subdivide .....	38
Εικόνα 3.16: Displace Modifier .....	38
Εικόνα 3.17: Texture .....	38
Εικόνα 3.18: Εφαρμογή του Displace Modifier .....	39
Εικόνα 3.19: Export FBX.....	39
Εικόνα 3.20: Import FBX.....	40
Εικόνα 3.21: Custom Collision Mesh .....	41
Εικόνα 3.22: BP_HapticTrigger - Custom Collision Mesh.....	42
Εικόνα 3.23: Αίθουσες Κρυμμένων Αντικειμένων.....	43
Εικόνα 3.24: BP_MaterialButtonChange - Event Graph - Ενέργεια πλήκτρου “T” .....	43
Εικόνα 3.25: Τα αντικείμενα μέσα στο χώρο.....	45

Εικόνα 3.26: BP_Quit – Viewport .....	46
Εικόνα 3.27: Rect Light .....	47
Εικόνα 3.28: Spot Light .....	47
Εικόνα 3.29: Trigger Box.....	48
Εικόνα 3.30: Level Blueprint - Event Graph.....	48
Εικόνα 3.31: Κεντρικό Μενού – Designer .....	50
Εικόνα 3.32: Κεντρικό Μενού – Graph .....	50
Εικόνα 3.33: Αρχείο κεφαλίδας του Character .....	52
Εικόνα 3.34: Δήλωση της κλάσης AHapticDeviceActor .....	52
Εικόνα 3.35: Προσθήκη στοιχείου Κάμερας.....	52
Εικόνα 3.36: Δημιουργία UCapsuleComponent .....	53
Εικόνα 3.37: Δημιουργία UCameraComponent.....	53
Εικόνα 3.38: ThesisHapticDeviceActor Blueprint.....	54
Εικόνα 3.39: ThesisHapticDeviceActor - Event Graph .....	55
Εικόνα 3.40: Χαρακτήρας μέσα στη σκηνή.....	56
Εικόνα 3.41: Χαρακτήρας μέσα στην εφαρμογή .....	56
Εικόνα 3.42: Πειράματα βάση ιδιότητας των υλικών .....	57
Εικόνα 3.43: Πειράματα βάση υφής επιφανειών (Texture) .....	57
Εικόνα 3.44: Πειράματα αναγνώρισης αντικειμένων (Shapes) .....	58
Εικόνα 4.1: Project – Description .....	59
Εικόνα 4.2: Project – Maps & Modes .....	60
Εικόνα 4.3: Project – Packaging .....	61
Εικόνα 4.4: Package Project.....	63
Εικόνα 4.5: Packaging Process.....	63
Εικόνα 5.1: Κεντρικό Μενού Εφαρμογής.....	64
Εικόνα 5.2: Φόρτωση πειράματος.....	64
Εικόνα 5.3: Έξοδος στο κεντρικό μενού.....	65
Εικόνα 5.4: Πειραματικός χώρος - Ιδιότητες Υλικών.....	65
Εικόνα 5.5: Σκληρότητα - Αντικείμενα από χρυσό και σφουγγάρι .....	66
Εικόνα 5.6: Ολισθηρότητα – Αντικείμενα από άσφαλτο και γυαλί.....	67
Εικόνα 5.7: Αντικείμενα με διαφορετική πυκνότητα.....	67
Εικόνα 5.8: Πείραμα αντίληψης της υφής .....	68
Εικόνα 5.9: Διαφορά στη γεωμετρία των αντικειμένων .....	68
Εικόνα 5.10: Ενδεικτικά σχήματα.....	69
Εικόνα 5.11: Αίθουσα πειράματος.....	69
Εικόνα 5.12: Φωτεινό αντικείμενο.....	70
Εικόνα 5.13: Ολοκλήρωση των πειραμάτων.....	70

## Κατάλογος Σχημάτων

Σχήμα 6.1: Αποτελέσματα πειράματος Σκληρότητας.....	72
Σχήμα 6.2: Αποτελέσματα πειράματος Ολίσθησης .....	72
Σχήμα 6.3: Αποτελέσματα πειράματος Διαπερατότητας .....	73
Σχήμα 6.4: Αποτελέσματα πειράματος Αναγνώρισης Σχημάτων .....	74

## Κατάλογος Πινάκων

Πίνακας 2.1: Προδιαγραφές συσκευής Touch .....	4
Πίνακας 2.2: Απαιτήσεις Συστήματος .....	5
Πίνακας 3.1: Πραγματικές τιμές των υλικών του πειράματος Σκληρότητας.....	34
Πίνακας 3.2: Αντίστοιχες τιμές που δόθηκαν μέσα στο Unreal.....	34
Πίνακας 3.3: Πραγματικές τιμές των υλικών του πειράματος Ολισθηρότητας .....	35
Πίνακας 3.4: Αντίστοιχες τιμές που δόθηκαν μέσα στο Unreal.....	35
Πίνακας 3.5: Πραγματικές τιμές των υλικών του πειράματος Διαπερατότητας .....	35
Πίνακας 3.6: Αντίστοιχες τιμές που δόθηκαν μέσα στο Unreal.....	36

## Συντομογραφίες

Δ.Ε.	Διπλωματική Εργασία
ΔΙΠΙΑΕ	Διεθνές Πανεπιστήμιο Ελλάδος
Π.Ε.	Πτυχιακή Εργασία
XR	Extended Reality
SDK	Software Development Kit
LOD	Land of Distance
UMG UI	Unreal Motion Graphics User Interface Designer
HUD	Heads-Up Display
FBX	Filmbox eXchange
VR	Virtual Reality



## **Κεφάλαιο 1ο: Απτική Τεχνολογία και Ανθρώπινη – Υπολογιστική Αλληλεπίδραση**

### **1.1 Εισαγωγή**

Η ραγδαία εξέλιξη των ψηφιακών τεχνολογιών έχει οδηγήσει στη δημιουργία ολοένα και πιο ρεαλιστικών εικονικών περιβαλλόντων, τα οποία αξιοποιούνται σε πληθώρα εφαρμογών, όπως η εκπαίδευση, η προσομοίωση, η ιατρική και η ψυχαγωγία. Παρά την εντυπωσιακή πρόοδο στον τομέα των γραφικών και του ήχου, η απουσία της αίσθησης της αφής περιορίζει σημαντικά τον βαθμό εμπύθισης και τη φυσικότητα της αλληλεπίδρασης του χρήστη με το ψηφιακό περιβάλλον [1], [2].

Η απτική τεχνολογία έρχεται να καλύψει αυτό το κενό, επιτρέποντας στον χρήστη να λαμβάνει φυσικά ερεθίσματα κατά την επαφή με εικονικά αντικείμενα. Μέσω της απτικής ανάδρασης, καθίσταται δυνατή η προσομοίωση ιδιοτήτων όπως η αντίσταση, η επιφάνεια και η φυσική συμπεριφορά των αντικειμένων, ενισχύοντας ουσιαστικά την εμπειρία αλληλεπίδρασης [1], [2].

### **1.2 Απτική Ανάδραση και Εικονικά Περιβάλλοντα**

Η απτική ανάδραση (haptic feedback) αποτελεί έναν από τους βασικότερους μηχανισμούς αλληλεπίδρασης μεταξύ ανθρώπου και ψηφιακών συστημάτων, καθώς επιτρέπει τη μετάδοση αισθητηριακών πληροφοριών που σχετίζονται με την αίσθηση της αφής [1], [2]. Σε αντίθεση με τις κλασικές διεπαφές εισόδου–εξόδου, οι οποίες περιορίζονται κυρίως στην οπτική και ακουστική πληροφορία, τα απτικά συστήματα προσφέρουν τη δυνατότητα φυσικής αλληλεπίδρασης με εικονικά αντικείμενα, ενισχύοντας τον ρεαλισμό και την αίσθηση εμπύθισης [2].

Στο πλαίσιο των εικονικών περιβαλλόντων, η απτική ανάδραση χρησιμοποιείται για την προσομοίωση ιδιοτήτων όπως η αντίσταση, η επαφή, η επιφάνεια και η αλληλεπίδραση με αντικείμενα στον χώρο [1], [2]. Η τεχνολογία αυτή βρίσκει εφαρμογή σε τομείς όπως η εκπαίδευση, η ιατρική προσομοίωση, η αποκατάσταση, η βιομηχανική σχεδίαση και η επιστημονική έρευνα, όπου η κατανόηση της φυσικής συμπεριφοράς των αντικειμένων είναι κρίσιμη [3], [4].

Η ανάπτυξη απτικών εφαρμογών απαιτεί τον συνδυασμό λογισμικού υψηλής ακρίβειας και εξειδικευμένου υλικού, ικανού να μεταφράζει τις ψηφιακές αλληλεπιδράσεις σε απτικά ερεθίσματα [1], [6]. Η πρόκληση έγκειται στην ακριβή αντιστοίχιση των υπολογιστικών μοντέλων με τις ανθρώπινες αισθητηριακές αντιλήψεις, ώστε η εμπειρία να είναι κατανοητή, ρεαλιστική και λειτουργική [2], [7].

### **1.3 Τεχνολογίες Προσομοίωσης και Ανθρώπινη Αλληλεπίδραση**

Οι σύγχρονες τεχνολογίες προσομοίωσης διαδραματίζουν καθοριστικό ρόλο στη διαμόρφωση προηγμένων συστημάτων ανθρώπινης–υπολογιστικής αλληλεπίδρασης. Οι μηχανές γραφικών πραγματικού χρόνου επιτρέπουν τη δημιουργία δυναμικών και πολύπλοκων εικονικών περιβαλλόντων, στα οποία ο χρήστης μπορεί να αλληλεπιδρά με αντικείμενα και συστήματα με φυσικό τρόπο [1], [2].

Η ενσωμάτωση απτικής ανάδρασης σε τέτοιες πλατφόρμες ενισχύει σημαντικά τη βιωματική εμπειρία του χρήστη, καθώς η αλληλεπίδραση δεν περιορίζεται πλέον στην παρατήρηση, αλλά επεκτείνεται στην αίσθηση της αφής [1], [2]. Η προσέγγιση αυτή μετατρέπει τα εικονικά περιβάλλοντα σε εργαλεία μάθησης και πειραματισμού, επιτρέποντας την εξερεύνηση εννοιών που δύσκολα μπορούν να αποδοθούν μόνο μέσω οπτικών μέσων [5].

Ωστόσο, η ανάπτυξη τέτοιων εφαρμογών συνοδεύεται από τεχνικές και σχεδιαστικές προκλήσεις. Απαιτείται προσεκτικός σχεδιασμός της αλληλεπίδρασης, σωστή διαχείριση συγκρούσεων και συγχρονισμός μεταξύ οπτικής και απτικής πληροφορίας [6], [7]. Η επιτυχία ενός απτικού συστήματος εξαρτάται σε μεγάλο βαθμό από τη συνοχή των επιμέρους στοιχείων του και από τον τρόπο με τον οποίο αυτά αντιλαμβάνονται οι τελικοί χρήστες [2], [7].

### 1.4 Στόχος, Κίνητρο και Σημασία της Εργασίας

Στόχος της παρούσας εργασίας είναι η μελέτη και η υλοποίηση ενός διαδραστικού εικονικού περιβάλλοντος που αξιοποιεί απτική ανάδραση, με σκοπό τη διερεύνηση της αίσθησης της αφής σε ψηφιακά περιβάλλοντα [1], [2]. Μέσω της ανάπτυξης πειραματικών διατάξεων και της ενσωμάτωσης απτικής συσκευής μηχανικού βραχίονα, επιδιώκεται η κατανόηση του τρόπου με τον οποίο ο χρήστης αντιλαμβάνεται, ερμηνεύει και διαφοροποιεί απτικά ερεθίσματα κατά την αλληλεπίδρασή του με εικονικά αντικείμενα [2], [7].

Το κίνητρο για την εκπόνηση της εργασίας προκύπτει από την αυξανόμενη ανάγκη ενσωμάτωσης πολυαισθητηριακών μορφών αλληλεπίδρασης σε σύγχρονα εικονικά περιβάλλοντα [2], [7]. Παρότι οι τεχνολογίες απτικής ανάδρασης και οι μηχανές γραφικών έχουν σημειώσει σημαντική εξέλιξη, η αποτελεσματική και ουσιαστική αξιοποίησή τους απαιτεί συστηματικό σχεδιασμό, πειραματισμό και εμπειρική αξιολόγηση [5]. Η απτική αίσθηση, σε αντίθεση με την οπτική και την ακουστική, παρουσιάζει αυξημένη πολυπλοκότητα, γεγονός που καθιστά αναγκαία τη μελέτη της σε ρεαλιστικά και ελεγχόμενα περιβάλλοντα.

Η σημασία της παρούσας εργασίας έγκειται στη σύνδεση της θεωρητικής γνώσης γύρω από την απτική τεχνολογία με την πρακτική υλοποίηση ενός λειτουργικού συστήματος [1], [2], [6]. Μέσα από την ανάπτυξη ενός ολοκληρωμένου εικονικού χώρου και τη δοκιμή του από πραγματικούς χρήστες, η εργασία συμβάλλει στην κατανόηση των δυνατοτήτων και των περιορισμών της απτικής ανάδρασης, καθώς και στη διερεύνηση της συμβολής της στη βελτίωση της εμπειρίας αλληλεπίδρασης σε εκπαιδευτικές και πειραματικές εφαρμογές [2], [3], [7].

### 1.5 Επίλογος

Στο παρόν κεφάλαιο παρουσιάστηκε το θεωρητικό και τεχνολογικό υπόβαθρο της απτικής ανάδρασης σε εικονικά περιβάλλοντα, καθώς και το πλαίσιο μέσα στο οποίο εντάσσεται η παρούσα εργασία. Αναδείχθηκε η σημασία της αίσθησης της αφής στη βελτίωση της ανθρώπινης-υπολογιστικής αλληλεπίδρασης και διατυπώθηκαν οι στόχοι, το κίνητρο και η σημασία της έρευνας. Στο επόμενο κεφάλαιο ακολουθεί η παρουσίαση των εργαλείων και τεχνολογιών που χρησιμοποιήθηκαν για την υλοποίηση του εικονικού απτικού περιβάλλοντος [1], [2].

## Κεφάλαιο 2ο: 3D Systems και Unreal Engine 4

### 2.1 3D Systems και ο μηχανικός βραχίονας Touch

#### 2.1.1 Εισαγωγή

Η 3D Systems είναι μία από τις ιστορικότερες και πλέον καινοτόμες εταιρείες στον χώρο της τρισδιάστατης τεχνολογίας. Ιδρύθηκε το 1986 από τον Chuck Hull, τον εφευρέτη της στερεολιθογραφίας (SLA), της πρώτης μεθόδου τρισδιάστατης εκτύπωσης στον κόσμο. Έκτοτε, η εταιρεία έχει αναπτυχθεί σε παγκόσμιο ηγέτη στον τομέα της προσθετικής κατασκευής, 3D μοντελοποίησης, βιομηχανικού σχεδιασμού, ιατρικών λύσεων και προηγμένων εργαλείων προσομοίωσης και επεξεργασίας πολύπλοκων δεδομένων. Μέσα σε αυτήν την πλούσια τεχνολογική οικογένεια ανήκουν και οι συσκευές απτικής ανάδρασης, με χαρακτηριστικότερο εκπρόσωπο το 3D Systems Touch [8], [9].

#### 2.1.2 Ο Μηχανικός βραχίονας

Το 3D Systems Touch, γνωστό και ως Geomagic Touch, αποτελεί ένα προηγμένο σύστημα απτικής διεπαφής (haptic interface) το οποίο επιτρέπει στον χρήστη να αλληλεπιδρά με εικονικά τρισδιάστατα αντικείμενα μέσω ενός μηχανικού βραχίονα υψηλής ακρίβειας και ενός ειδικά σχεδιασμένου stylus, όπως απεικονίζεται στην Εικόνα 2.1. Η συσκευή ενσωματώνει τεχνολογίες ανάδρασης δύναμης (force feedback) που καταγράφουν με λεπτομέρεια τις κινήσεις του χεριού και αποδίδουν ελεγχόμενες δυνάμεις προς τον χρήστη, προσομοιώνοντας έτσι με ρεαλισμό την αίσθηση επαφής, αντίστασης και υφής των ψηφιακών αντικειμένων. Μέσω αυτής της μηχανικής και απτικής συνέργειας, το Touch καθιστά δυνατή την απτή αναπαράσταση εικονικών επιφανειών και υλικών, επιτρέποντας την εμπειρική εξερεύνηση και επεξεργασία τους σαν να επρόκειτο για πραγματικά φυσικά αντικείμενα [8], [10].



Εικόνα 2.1: Ο Μηχανικός βραχίονας 3D Systems Touch [9]

### 2.1.3 Λειτουργία

Ο μηχανισμός λειτουργίας της απτικής συσκευής Touch™ βασίζεται σε μια αρθρωτή δομή έξι βαθμών ελευθερίας, η οποία επιτρέπει την ακριβή ανίχνευση της θέσης και του προσανατολισμού της γραφίδας σε πραγματικό χρόνο μέσω υψηλής ευκρίνειας αισθητήρων, με ανάλυση που υπερβαίνει τα 450 dpi. Παράλληλα, η συσκευή διαθέτει τρεις βαθμούς ελευθερίας στην ανάδραση δύναμης, αξιοποιώντας ειδικούς μηχανισμούς που ασκούν στο χέρι του χρήστη ελεγχόμενες δυνάμεις έως περίπου 3.3 N, δημιουργώντας την αίσθηση φυσικών αντιστάσεων, σκληρότητας, υφών και ορίων στο εικονικό περιβάλλον. Το εργονομικό στυλέο, εξοπλισμένο με προγραμματιζόμενα κουμπιά, ενισχύει τον χειρισμό και την αλληλεπίδραση, ενώ ο συμπαγής και φιλικός προς το γραφείο σχεδιασμός, σε συνδυασμό με τη σύνδεση μέσω USB, καθιστούν τη συσκευή κατάλληλη για επαγγελματικές και εκπαιδευτικές εφαρμογές στους τομείς της τρισδιάστατης μοντελοποίησης, του σχεδιασμού και της Προσομοίωσης [8], [11].

Πίνακας 2.1: Προδιαγραφές συσκευής Touch [9]

Προδιαγραφές Συσκευής	Τιμές	Σημείωση
Χώρος Εργασίας Ανατροφοδότησης (Force Feedback Workspace)	> 431 W x 348 H x 165 D mm	Μέγιστο τρισδιάστατο πεδίο λειτουργίας στο οποίο μπορεί να κινείται η γραφίδα
Ανάλυση Θέσης (Nominal Position Resolution)	> 450 dpi (~ 0.055 mm)	Διακριτική ικανότητα ανίχνευσης
Μέγιστη δύναμη (Maximum Exertable Force)	3.3 N	Μέγιστη δύναμη απτική ανάδραση στη θέση αναφοράς
Συνεχής δύναμη (Continuous Exertable Force)	> 0.88 N	Δύναμη που μπορεί να διατηρηθεί σε μακρά χρήση
Σκληρότητα (Stiffness)	X axis > 1.26 N/mm Y axis > 2.31 N/mm Z axis > 1.02 N/mm	Σκληρότητα κατά άξονα
Τριβή κίνησης (Backdrive Friction)	< 0.26 N	Εξωτερική μηχανική αντίσταση όταν δεν εφαρμόζεται ενεργά δύναμη
Αδράνεια (Inertia)	~ 45 g	Φαινόμενη μάζα στο άκρο της συσκευής, επηρεάζει την αίσθηση ελαφρότητας/ευκίνησας
Αποτύπωμα (Footprint)	~ 168 W x 203 D mm	Η φυσική περιοχή που καταλαμβάνει η βάση της συσκευής σε μια επιφάνεια
Βάρος (Weight)	~1.42 kg	Το καθαρό βάρος της συσκευής

Πίνακας 2.2: Απαιτήσεις Συστήματος [9]

Απαιτήσεις Συστήματος	
Λειτουργικό σύστημα	64-bit Windows 10 and 11
Επεξεργαστής	Intel i5 / i7, 5th Generation or greater CPU, Minimum 2.5 GHz frequency
RAM	4 GB
Κάρτα γραφικών	Minimum 256 MB VRAM (Install the latest drivers)
Χώρος στο δίσκο	512 MB
Ανάλυση οθόνης	1280 x 800 (Minimum)
Διεπαφή (Interface)	Θύρα USB 2.0 / 3.0 ή διανομέας USB που υποστηρίζει USB 2.0/ 3.0. *Χρησιμοποιήστε μόνο το καλώδιο USB A σε B που παρέχεται από την 3D Systems.

#### 2.1.4 Εφαρμογή της συσκευής

Στην πράξη, το 3D Systems Touch έχει αξιοποιηθεί εκτενώς στον τομέα της ψηφιακής μοντελοποίησης και της καλλιτεχνικής γλυπτικής. Σε συνδυασμό με εξειδικευμένο λογισμικό, η συσκευή επιτρέπει την προσομοίωση υλικών με διαφορετική πλαστικότητα, επιφανειακή σκληρότητα και υφή, διευκολύνοντας έτσι την επεξεργασία οργανικών και μη οργανικών μορφών με υψηλή λεπτομέρεια. Ο χειρισμός που προσφέρει το Touch είναι σημαντικά πιο φυσικός από εκείνον των συμβατικών συσκευών εισόδου, όπως το ποντίκι ή τα γραφιστικά tablets, επιτρέποντας την ανάπτυξη σύνθετων τρισδιάστατων αντικειμένων με αυξημένη αίσθηση ελέγχου [12], [13].

#### 2.1.5 Σύνδεση με μηχανή γραφικών και Παραδείγματα εφαρμογής της ανάπτυξης συστήματος

Η αξία του Touch αναδεικνύεται ακόμη περισσότερο όταν συνδυαστεί με μηχανές γραφικών υψηλής πιστότητας, όπως η Unreal Engine. Η σύζευξη αυτών των δύο τεχνολογιών επιτρέπει τη δημιουργία εικονικών περιβαλλόντων όπου η οπτική πληροφορία συνοδεύεται από απτική ανατροφοδότηση, ανοίγοντας τον δρόμο για εφαρμογές που βασίζονται στην αλληλεπίδραση ανθρώπου-μηχανής σε ένα επίπεδο πολύ πιο κοντά στη φυσική πραγματικότητα [14], [15].

Η σύνδεση αυτή βρίσκει εφαρμογή σε πολλούς τομείς. Για παράδειγμα, στην ιατρική εκπαίδευση το Unreal Engine μπορεί να προσομοιώσει ιστούς, όργανα ή οστά, ενώ το Touch επιτρέπει στους φοιτητές να αισθανθούν την πραγματική αντίσταση που θα συναντούσαν σε μια χειρουργική πράξη. Στη ρομποτική και τη μηχανική, ο συνδυασμός των δύο τεχνολογιών επιτρέπει την πραγματοποίηση προσομοιώσεων συναρμολόγησης, όπου ο χρήστης μπορεί να δοκιμάσει τη θέση και την εφαρμογή εξαρτημάτων με απτική ανατροφοδότηση. Στον χώρο της ψηφιακής γλυπτικής, το Touch επιτρέπει μια φυσική διαδικασία μοντελοποίησης, όπου η πίεση που εφαρμόζει ο χρήστης επηρεάζει την παραμόρφωση του ψηφιακού υλικού μέσα στο περιβάλλον της Unreal Engine [12], [13], [16].

### 2.1.6 Συμπέρασμα

Πέρα από τις επαγγελματικές εφαρμογές, η τεχνολογία αυτή έχει ιδιαίτερη αξία και στην έρευνα της διεπαφής ανθρώπου–μηχανής. Το Unreal Engine παρέχει ένα πλήρως ελεγχόμενο και οπτικά άρτιο περιβάλλον δοκιμών, ενώ το 3D Systems Touch προσφέρει μετρήσιμα και επαναλήψιμα απτικά ερεθίσματα. Αυτό επιτρέπει στους ερευνητές να μελετήσουν την ανθρώπινη αντίληψη της αφής, τη δυναμική των κινήσεων και τη συμπεριφορά σε εικονικά περιβάλλοντα με μεγάλη ακρίβεια [10], [13].

## 2.2 Η μηχανή γραφικών Unreal Engine 4.27

### 2.2.1 Εισαγωγή

Η Epic Games, η εταιρεία πίσω από την ανάπτυξη του Unreal Engine, ιδρύθηκε το 1991 από τον Tim Sweeney και εξελίχθηκε σε έναν από τους σημαντικότερους φορείς τεχνολογικής καινοτομίας στην παγκόσμια βιομηχανία ψηφιακού περιεχομένου. Παρότι έγινε ευρέως γνωστή μέσω δημοφιλών τίτλων βιντεοπαιχνιδιών, η κυριότερη συμβολή της αφορά την ανάπτυξη της μηχανής γραφικών Unreal Engine, η οποία έχει επηρεάσει βαθιά το πεδίο της διαδραστικής απεικόνισης, της εικονικής πραγματικότητας και των οπτικοακουστικών τεχνολογιών [15], [17].

### 2.2.2 Unreal Engine

Το Unreal Engine, από την πρώτη του εμφάνιση το 1998, εξελίχθηκε σε μία από τις πιο ολοκληρωμένες και τεχνολογικά προηγμένες μηχανές ανάπτυξης διαδραστικών εφαρμογών, προσομοιώσεων και ψηφιακών περιβαλλόντων. Με κάθε νέα έκδοση, η μηχανή εμπλουτίστηκε με επιπλέον τεχνολογικές δυνατότητες, ωστόσο η σειρά Unreal Engine 4 αποτέλεσε σημείο καμπής, καθώς εισήγαγε μια νέα αρχιτεκτονική που υποστηρίζει υψηλό επίπεδο γραφικής πιστότητας, προηγμένα συστήματα φυσικής και πολύπλοκες αλληλεπιδράσεις σε πραγματικό χρόνο. Χάρη στα σύνθετα συστήματα φωτισμού, υλικών, animation και προσομοίωσης, το Unreal Engine επιτρέπει τη δημιουργία εικονικών κόσμων με βαθμό ρεαλισμού που προσεγγίζει την πραγματικότητα, τόσο σε επίπεδο οπτικής απόδοσης όσο και σε επίπεδο λειτουργικής συμπεριφοράς. Ο στόχος του είναι να προσφέρει ένα περιβάλλον ανάπτυξης ικανό να υποστηρίξει πλούσιες, διαδραστικές και υψηλής ποιότητας εμπειρίες, ενοποιώντας προηγμένες τεχνολογίες σε ένα συνεκτικό και ευέλικτο σύστημα παραγωγής ψηφιακού περιεχομένου [14].

### 2.2.3 Unreal Engine 4.27

Η έκδοση 4.27 αποτελεί προϊόν μιας μακράς εξελικτικής διαδικασίας που βασίζεται σε αρχιτεκτονική υψηλής απόδοσης και ευελιξίας. Ο πυρήνας της μηχανής έχει σχεδιαστεί ώστε να αξιοποιεί προηγμένες τεχνικές φωτισμού, σκίασης και αναπαράστασης υλικών, επιτρέποντας την παραγωγή εικόνων με υψηλό βαθμό ρεαλισμού. Η υποστήριξη φυσικής προσομοίωσης, κινηματικών συστημάτων, δυναμικών φωτισμών και volumetric rendering καθιστά τη μηχανή κατάλληλη τόσο για γραφικά υψηλής πιστότητας όσο και για συστήματα που απαιτούν ακριβή αναπαράσταση περιβαλλοντικών συμπεριφορών.

Η τρέχουσα έκδοση, χαρακτηρίζεται επίσης από τη σταθερότητα και την ωριμότητά της, οι οποίες είναι κρίσιμες για έργα μεγάλης κλίμακας και για ακαδημαϊκή έρευνα. Η πλήρης τεκμηρίωση, η ενεργή κοινότητα ανάπτυξης και η διαρκής υποστήριξη καθιστούν το Unreal Engine ένα αξιόπιστο εργαλείο για πειραματικές εφαρμογές, ιδίως σε τομείς όπως η ρομποτική, η τεχνητή νοημοσύνη, τα εικονικά εργαστήρια και τα συστήματα αυτόνομης πλοήγησης. Η δυνατότητα αναπαραγωγής σύνθετων φυσικών

φαινομένων και η συμβατότητα με εξωτερικά συστήματα αισθητήρων το καθιστούν ιδιαίτερα χρήσιμο στην ερευνητική δραστηριότητα [15], [17].

#### 2.2.4 Τρόποι ανάπτυξης κώδικα

Στο Unreal Engine 4.27, η ανάπτυξη εφαρμογών βασίζεται σε ένα υβριδικό μοντέλο που συνδυάζει την παραδοσιακή γλώσσα προγραμματισμού C++ με το οπτικό σύστημα δημιουργίας λογικής Blueprints Visual Scripting. Ο συνδυασμός αυτός αποτελεί μία από τις καθοριστικότερες καινοτομίες του Unreal Engine, καθώς επιτρέπει την ταυτόχρονη αξιοποίηση της υψηλής απόδοσης και της εκφραστικότητας της C++ με την ευκολία, την οπτικότητα και την αμεσότητα των Blueprints.

Η C++ λειτουργεί ως το χαμηλού επιπέδου θεμέλιο της μηχανής, επιτρέποντας την υλοποίηση πολύπλοκων αλγορίθμων, την άμεση πρόσβαση στη μνήμη και τη δημιουργία δομών υψηλής αποδοτικότητας. Η χρήση της προσφέρει μεγάλο βαθμό ελέγχου επάνω στη συμπεριφορά της μηχανής και είναι απαραίτητη για εφαρμογές που απαιτούν αυστηρές βελτιστοποιήσεις, προηγμένες τεχνικές τεχνητής νοημοσύνης, προσαρμοσμένα φυσικά μοντέλα ή εξειδικευμένη διαχείριση πόρων. Η δυνατότητα μεταγλώττισης του κώδικα επιτρέπει επίσης τη μέγιστη δυνατή ταχύτητα εκτέλεσης, κάτι που είναι κρίσιμο σε περιβάλλοντα πραγματικού χρόνου όπου ακόμη και μικρές καθυστερήσεις μπορούν να επηρεάσουν την ομαλότητα της εμπειρίας.

Αντίθετα, τα Blueprints συνιστούν μια ανώτερου επιπέδου διεπαφή οπτικής προγραμματιστικής λογικής, στην οποία η ροή των εντολών αποτυπώνεται μέσω διαγραμμάτων και συνδέσεων αντί γραμμών κώδικα. Η προσέγγιση αυτή μειώνει δραστικά το γνωστικό εμπόδιο για τους δημιουργούς που δεν διαθέτουν εκτεταμένη εμπειρία προγραμματισμού, καθώς επιτρέπει τη γρήγορη διαμόρφωση συμπεριφορών, την πειραματική διαδικασία και την άμεση τροποποίηση της λογικής ενός έργου χωρίς την ανάγκη μεταγλώττισης. Τα Blueprints είναι ιδιαίτερα χρήσιμα στον πρωτοτυπικό σχεδιασμό (rapid prototyping), καθώς επιτρέπουν τη δοκιμή ιδεών σε πραγματικό χρόνο και τη γρήγορη αξιολόγηση της λειτουργικότητας χωρίς να απαιτείται σύνθετος κώδικας.

Η συνήθης ροή δόμησης ενός έργου γίνεται με υλοποίηση του πυρήνα της λειτουργικότητας σε C++ και στη συνέχεια η ανάπτυξη από τις λεπτομερείς συμπεριφορές, τις αλληλεπιδράσεις και την παραμετροποίηση του έργου μέσω Blueprints. Το Unreal Engine επιτρέπει επίσης, τη δημιουργία C++ classes με ειδικά ορίσματα που εκθέτουν συγκεκριμένες μεταβλητές, συναρτήσεις και ιδιότητες στα Blueprints, δημιουργώντας ένα διαδραστικό περιβάλλον όπου η λογική σε C++ τροφοδοτεί τα Blueprints χωρίς απώλειες στην απόδοση.

Το υβριδικό αυτό μοντέλο έχει γίνει καθοριστικό στην ευρεία διάδοση της μηχανής, καθώς ενσωματώνει τόσο την ισχύ των παραδοσιακών τεχνικών προγραμματισμού υψηλής απόδοσης όσο και τη χρηστικότητα ενός προσιτού, οπτικού περιβάλλοντος ανάπτυξης. Η ευελιξία αυτή επιτρέπει στην έκδοση 4.27 να εξυπηρετεί ένα εξαιρετικά μεγάλο εύρος εφαρμογών, από επαγγελματικά βιντεοπαιχνίδια και κινηματογραφική παραγωγή έως επιστημονικές προσομοιώσεις, εικονικά εργαστήρια και εκπαιδευτικά εργαλεία[15].

## **2.3 Επεκτάσιμα και Πρόσθετα (Plugins)**

### **2.3.1 Εισαγωγή**

Τα απτικά συστήματα, παρότι δεν υποστηρίζονται εγγενώς από τον βασικό πυρήνα της Unreal Engine, μπορούν να ενσωματωθούν αποτελεσματικά μέσω της αρχιτεκτονικής προσθέτων (plugins), της υποδομής εισόδου σε πραγματικό χρόνο και της επεκτασιμότητας που προσφέρει η C++. Με τον τρόπο αυτό καθίσταται δυνατή η υλοποίηση απτικής αλληλεπίδρασης, επιτρέποντας στους χρήστες να «αγγίζουν» και να χειρίζονται ψηφιακά αντικείμενα μέσα στο εικονικό περιβάλλον, ενισχύοντας ουσιαστικά τον ρεαλισμό και τον βαθμό εμπύθισης της εμπειρίας. Η δημιουργία εξειδικευμένων προσθέτων λειτουργεί ως άμεση γέφυρα μεταξύ της Unreal Engine και των απτικών συσκευών, επιτρέποντας στους προγραμματιστές να επεκτείνουν τα όρια της αισθητηριακής ανάδρασης και να ενσωματώσουν προηγμένες μορφές απτικής πληροφορίας σε διαδραστικά ψηφιακά συστήματα [18].

### **2.3.2 Εξατομικευμένες Απτικές Συσκευές και SDK τρίτων κατασκευαστών**

Η διαδικασία αυτή συνήθως περιλαμβάνει την αλληλεπίδραση με ιδιόκτητα SDK που διαχειρίζονται την επικοινωνία σε χαμηλό επίπεδο με το υλικό, περιλαμβάνοντας την αρχικοποίηση της συσκευής, τη μετάδοση διανυσμάτων δύναμης και την περιοδική ανάγνωση αισθητήρων. Το σύστημα προσθέτων της Unreal επιτρέπει την ενσωμάτωση αυτών των SDK ως αρθρώματα C++ που εντάσσονται στο runtime της μηχανής. Μόλις ενσωματωθούν, μπορούν να δημιουργηθούν κόμβοι Blueprint ή προσαρμοσμένα components που εκθέτουν τη λειτουργικότητα στο περιβάλλον οπτικού προγραμματισμού της Unreal. Η ανίχνευση συγκρούσεων, οι φυσικές ιδιότητες και τα αποτελέσματα προσομοίωσης μπορούν να χρησιμοποιηθούν δυναμικά για την καθοδήγηση της εμπειρίας απτικής ανάδρασης [11], [19].

### **2.3.3 Ζητήματα Απόδοσης και Πολυπλοκότητας Επεξεργασίας**

Η επίτευξη μιας αδιάλειπτης εμπειρίας απτικής ανάδρασης απαιτεί προσεκτική διαχείριση της απόδοσης. Οι δυνατότητες πολυνηματικής επεξεργασίας της Unreal Engine επιτρέπουν τον διαχωρισμό του βρόχου ανάδρασης δύναμης και των λειτουργιών ανάγνωσης συσκευής από τα νήματα απόδοσης και λογικής παιχνιδιού. Αυτό βοηθά στην αποφυγή διακοπών (jitter) και καθυστερήσεων (latency), που είναι ιδιαίτερα επιζήμιες για την απτική αντίληψη. Το σύστημα ανίχνευσης συγκρούσεων της μηχανής μπορεί να επεκταθεί ώστε να περιλαμβάνει παραμέτρους υλικών φυσικής—όπως σκληρότητα ή τραχύτητα—που πληροφορούν τη μηχανή απόδοσης απτικής ανάδρασης για τον τρόπο προσομοίωσης της απόκρισης επιφάνειας. Τα παραπάνω μέτρα διασφαλίζουν απτική ρεαλιστικότητα ενώ διατηρούν την απόκριση σε πραγματικό χρόνο [13].

### **2.3.4 Ανάδραση μέσω Απόδοσης και Ιδιοτήτων Υλικών**

Πέρα από τη φυσική και την είσοδο, οι οπτικές ιδιότητες απόδοσης στην Unreal μπορούν να χρησιμοποιηθούν για να ενημερώσουν τη συμπεριφορά της απτικής ανάδρασης. Υλικά που οπτικά εμφανίζονται μαλακά ή ελαστικά μπορούν να συνδυαστούν με χαμηλή αντίσταση ή διακριτικές δονήσεις, ενώ γυαλιστερές ή άκαμπτες επιφάνειες μπορεί να παράγουν απόκριση έντονης και σταθερής δύναμης. Αυτή η πολυτροπική σύνδεση μεταξύ οπτικής πιστότητας και φυσικής αίσθησης ενισχύει περαιτέρω το αίσθημα παρουσίας και εμπύθισης του χρήστη [13].

### 2.3.5 Συμπεράσματα

Αν και η Unreal Engine 4.27 δεν προσφέρει εγγενή υποστήριξη για προηγμένη απτική ανάδραση πέραν των χαρακτηριστικών που περιλαμβάνονται στα πρότυπα XR, διαθέτει μια ώριμη και εξαιρετικά επεκτάσιμη αρχιτεκτονική που μπορεί να προσαρμοστεί στην ενσωμάτωση απτικής τεχνολογίας.

Είτε πρόκειται για προσομοιώσεις και σχεδίαση εικονικών πρωτοτύπων, είτε για δημιουργία εργαλείων εκπαίδευσης σε φυσική αλληλεπίδραση, η Unreal Engine 4.27 αποτελεί μια ισχυρή και αξιόπιστη μηχανή για την ανάπτυξη συστημάτων απτικής ανάδρασης [15], [13].

## 2.4 Ενσωμάτωση του 3DSystems Touch με το Unreal Engine 4.27

### 2.4.1 Εισαγωγή

Η σύνδεση της Unreal Engine με το 3D Systems Touch αποτελεί ένα χαρακτηριστικό παράδειγμα ολοκλήρωσης μεταξύ ενός σύγχρονου γραφικού engine και μιας συσκευής απτικής ανάδρασης υψηλής ακρίβειας. Ο συνδυασμός αυτός δημιουργεί ένα ολοκληρωμένο σύστημα εικονικής αλληλεπίδρασης, στο οποίο ο χρήστης όχι μόνο βλέπει και κινείται μέσα σε έναν ψηφιακό χώρο, αλλά ταυτόχρονα αισθάνεται την ύπαρξη και τις ιδιότητες των αντικειμένων. Η επικοινωνία μεταξύ των δύο τεχνολογιών επιτυγχάνεται μέσω ειδικού λογισμικού, βιβλιοθηκών και κατάλληλων διεπαφών που επιτρέπουν τον συγχρονισμό των κινήσεων και των απτικών δυνάμεων σε πραγματικό χρόνο [8], [11], [14].

### 2.4.2 Τρόπος σύνδεσης

#### 2.4.2.1 Πρόσθετα (Plugins)

Η τεχνική διασύνδεση ξεκινά από το OpenHaptics Toolkit, το οποίο αποτελεί το βασικό API για την ανάπτυξη απτικής λειτουργικότητας στο 3D Systems Touch. Το OpenHaptics παρέχει πρόσβαση στα δεδομένα αισθητήρων, όπως η θέση, ο προσανατολισμός και η ταχύτητα της γραφίδας, καθώς και εργαλεία για την εφαρμογή δυνάμεων ανάδρασης.

Το Unreal Engine 4.27, ως C++-βασισμένη μηχανή με αρθρωτή αρχιτεκτονική, προσφέρει πλήρη υποστήριξη για ανάπτυξη προσθέτων και διασύνδεση με εξωτερικά SDK όπως το OpenHaptics.

Η ενσωμάτωση των απτικών συσκευών πραγματοποιείται μέσω προσαρμοσμένων προσθέτων σε C++, στην περίπτωση μας το HapticsDirect, τα οποία συνδέουν το SDK της συσκευής με τη μηχανή, εκθέτουν τη λειτουργικότητά της μέσω Blueprints και αξιοποιούν πολυνηματική επεξεργασία για την κάλυψη των υψηλών συχνοτήτων ενημέρωσης που απαιτούνται από απτικά συστήματα (500–1000 Hz). Η προσέγγιση αυτή επιτρέπει τη δημιουργία επαναχρησιμοποιήσιμων Actor Components και επιτυγχάνει υψηλές επιδόσεις.

Το πρόσθετο ενσωματώνεται πλήρως στον Unreal Editor μέσω του παραθύρου Fab, διευκολύνοντας μια ομαλή ροή εργασίας ακόμη και για χρήστες χωρίς εμπειρία σε C++ [11], [18], (βλ. Παράρτημα Α).

#### 2.4.2.2 Λειτουργικότητα

Κατά την αλληλεπίδραση, το Unreal Engine λειτουργεί ως το περιβάλλον που διαχειρίζεται τη γεωμετρία, τα υλικά, τον φωτισμό και τη φυσική συμπεριφορά των αντικειμένων. Όταν η γραφίδα του Touch έρχεται σε επαφή με ένα αντικείμενο, η μηχανή υπολογίζει το σημείο σύγκρουσης, το βάθος διείσδυσης και την αντίδραση του υλικού. Στη συνέχεια, μέσω των λειτουργιών του πρόσθετου (plugin),

στέλνει αυτές τις πληροφορίες πίσω στη συσκευή, η οποία με τη σειρά της ασκεί αντίστοιχη δύναμη στο χέρι του χρήστη. Με τον τρόπο αυτό δημιουργείται ο απαραίτητος βρόχος ανάδρασης (feedback loop), ο οποίος επιτρέπει την απτική ενημέρωση σε πραγματικό χρόνο [10], [11].

### 2.4.2.3 Περιορισμοί

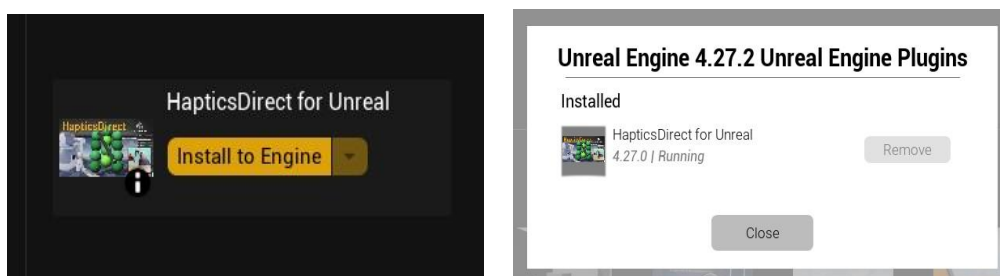
Παρότι το σύστημα προσφέρει σημαντική ευελιξία, παρουσιάζει ορισμένους περιορισμούς, κυρίως την εξάρτησή του από συμβατό απτικό υλικό για δοκιμές και ανάπτυξη, καθώς και το γεγονός ότι η επίσημη υποστήριξη αφορά αποκλειστικά την έκδοση 4.27, κάτι που μπορεί να απαιτήσει χειροκίνητες προσαρμογές για νεότερες ή παλαιότερες εκδόσεις του Unreal Engine. Όπως επίσης, για τη δοκιμή της απτικής απόκρισης απαιτείται η κατάλληλη απτική συσκευή, η οποία θα πρέπει να είναι συνδεδεμένη και σωστά ρυθμισμένη, ενώ ενδέχεται να απαιτούνται επιπλέον οδηγοί ή λογισμικό ενδιάμεσης διασύνδεσης ανάλογα με τον κατασκευαστή [8].

## 2.4.3 Απαιτήσεις Λογισμικού και Ρυθμίσεις

### 2.4.3.1 Διαδικασία εγκατάστασης SDK

Για να μπορεί να γίνει χρήση του μηχανικού βραχίονα θα πρέπει να έχει προηγηθεί η εγκατάσταση του απαραίτητου λογισμικού (Drivers) όπως ορίζεται από την εταιρία διανομέα 3D Systems και να έχουν πραγματοποιηθεί οι απαραίτητοι έλεγχοι μέσω της διεπαφής του διανομέα ώστε ο μηχανικός βραχίονας να καθίσταται πλήρως λειτουργικός. Μετά την ολοκλήρωση της αρχικοποίησης του στο σύστημα θα πρέπει να γίνει η σύνδεση της συσκευής με το Unreal Engine. Η διαδικασία αυτή απαιτεί να υπάρχει εγκατεστημένη η έκδοση Unreal Engine 4.27 που είναι συμβατή με το πρόσθετο (Plugin) της 3D Systems.

Αρχικά θα πρέπει να πραγματοποιηθεί λήψη του OpenHaptics SDK από την πλατφόρμα διανομής περιεχομένου Fab στο τοπικό σύστημα. Το SDK θα πρέπει με την σειρά του να εγκατασταθεί στην μηχανή Unreal, μια αυτόματη διαδικασία που εκτελείται μέσω των συστημάτων της Epic Games και δεν χρειάζεται κάποια ειδική παρέμβαση του χρήστη, όπως φαίνεται στην Εικόνα 2.2.



Εικόνα 2.2: Εγκατάσταση Πρόσθετου στη μηχανή Unreal

Τα αρχεία του πρόσθετου τοποθετούνται στον φάκελο “UE\_4.27\Engine\Plugins\Marketplace”, μέσα στα αρχεία της μηχανής. Με τον τρόπο αυτό η αρχικοποίηση και σύνδεση των δυο συστημάτων έχει ολοκληρωθεί. Παρόλα αυτά, δεν είναι απόλυτα σίγουρο ότι το πρόσθετο θα είναι αναγνωρίσιμο από την μηχανή καθώς η συμβατότητα εξαρτάται αυστηρά από την ακριβή έκδοση της μηχανής. Στην προκειμένη περίπτωση το OpenHaptics SDK έχει διαμορφωθεί ειδικά για την έκδοση 4.27.0 και δεν είναι συμβατό με μεταγενέστερες υποεκδόσεις της σειράς 4.27, όπως η τρέχουσα ενεργή έκδοση 4.27.2 [8], [11].

### 2.4.3.2 Προϋποθέσεις Αναδιαμόρφωσης

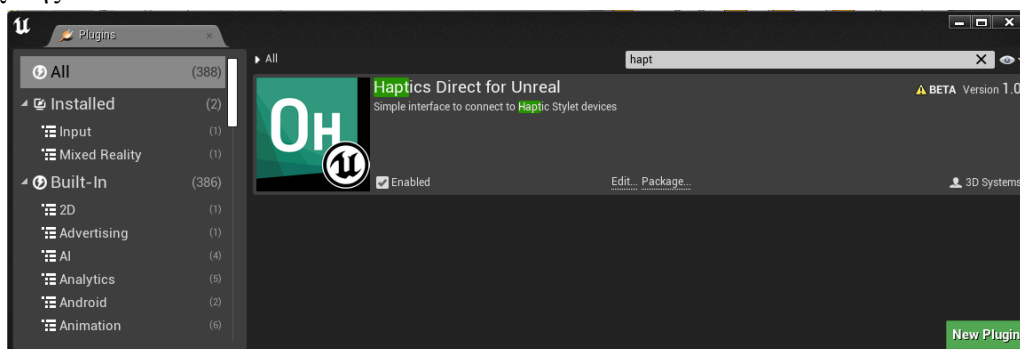
Για την επίλυση του προβλήματος ασυμβατότητας απαιτείται η επαναδιαμόρφωση (reconfiguration) του OpenHaptics SDK ώστε να υποστηρίζεται από την τρέχουσα έκδοση της μηχανής. Για την αναδημιουργία ενός πρόσθετου (plugin) σε προσαρμοσμένη (custom) έκδοση της μηχανής, το πρόσθετο οφείλει να διαθέτει τον δικό του πηγαίο κώδικα - προϋπόθεση που πληροί το πρόσθετο το οποίο επιδιώκουμε να αναδημιουργήσουμε (Re-Build). Όπως επίσης, θα πρέπει να υπάρχει εγκατεστημένη και η έκδοση Visual Studio με την οποία έγινε αρχικά η διαμόρφωσή του. Για το OpenHaptics είναι απαραίτητο να υπάρχει προεγκατεστημένο το Visual Studio 2017 για να μπορέσει να πραγματοποιηθεί η αναδημιουργία.

### 2.4.3.3 Μέθοδος Αναδιαμόρφωσης

Η μέθοδος που ακολουθήθηκε έγινε με την χρήση του Automation Tool και είναι πρακτική κυρίως για πρόσθετα που αναπτύσσονται και διανέμονται από τρίτους κατασκευαστές, όχι επίσημα πρόσθετα της Epic Games, ή για πρόσθετα που δεν διανέμονται σε συσκευασμένη (packaged) μορφή μαζί με τη μηχανή [19]. Μετά την ολοκλήρωση της διαδικασίας δημιουργείται μια νέα, πλήρως αναδομημένη έκδοση του πρόσθετου. Το παραγόμενο πρόσθετο περιλαμβάνει όλα τα απαραίτητα binaries και αρχεία που έχουν αναπτυχθεί ειδικά για την έκδοση της μηχανής που χρησιμοποιήθηκε. Το τελευταίο στάδιο για την ολοκλήρωση της διαδικασίας αναδημιουργίας είναι να αντικατασταθούν τα αρχεία του πρόσθετου από τα νέα αναδομημένα αρχεία στον εσωτερικό φάκελο πρόσθετων της μηχανής. Μόλις ολοκληρωθεί η αντικατάσταση, το πρόσθετο είναι έτοιμο προς χρήση και αναγνωρίζεται κανονικά από τον Unreal Editor.

### 2.4.3.4 Ενεργοποίηση στη Ροή Εργασίας

Μετά την εγκατάσταση, το πρόσθετο πρέπει να ενεργοποιηθεί μέσω του μενού Plugins στην καρτέλα Edit για το τρέχον έργο, όπως φαίνεται στην Εικόνα 2.3 και στη συνέχεια απαιτείται η επανεκκίνηση του για να ολοκληρωθεί η ενσωμάτωση. Μόλις ενεργοποιηθεί, τα στοιχεία του προσθέτου—όπως οι απτικοί actors και οι κόμβοι Blueprint—θα εμφανιστούν στον Content Browser μέσα στα περιεχόμενα της μηχανής.



Εικόνα 2.3: Ενεργοποίηση Πρόσθετου

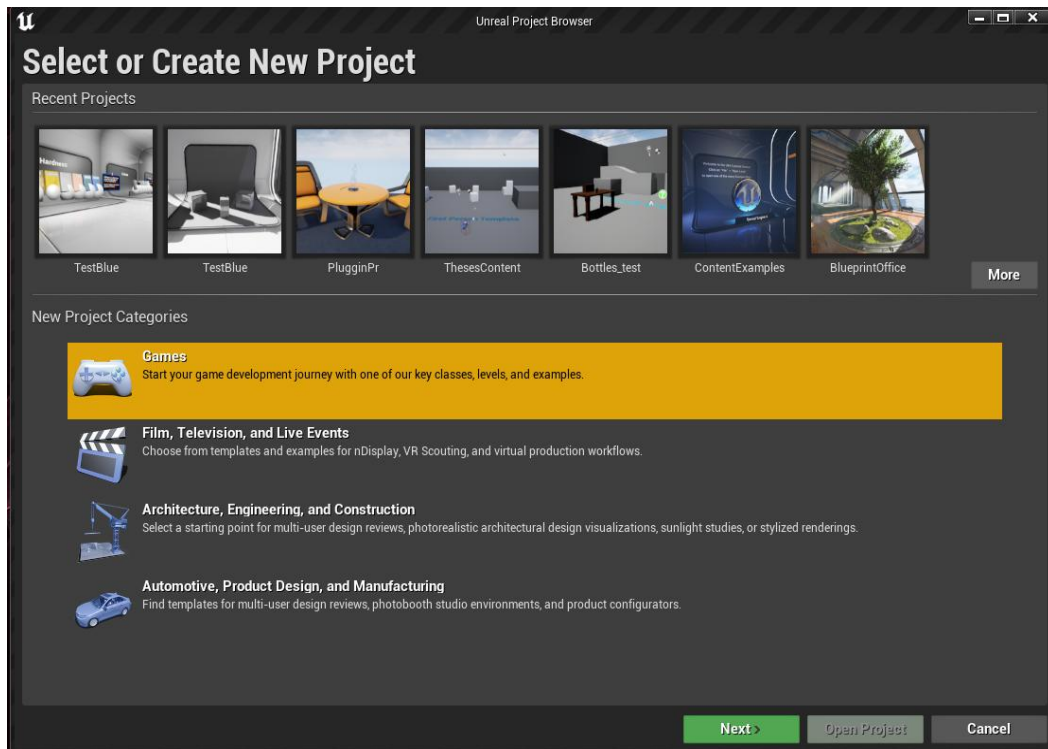
## 2.4.4 Συμπεράσματα

Η ενσωμάτωση του 3D Systems Touch με την Unreal Engine 4.27 δημιουργεί μια έμπιστη πλατφόρμα για απτική αλληλεπίδραση σε εικονικά περιβάλλοντα. Αν και η εγκατάσταση απαιτεί προσαρμοσμένη ανάπτυξη και βελτιστοποίηση απόδοσης, τα οφέλη στις εφαρμογές προσομοίωσης, εκπαίδευσης και σχεδίασης με εμπύθιση δικαιολογούν την τεχνική επένδυση [19].

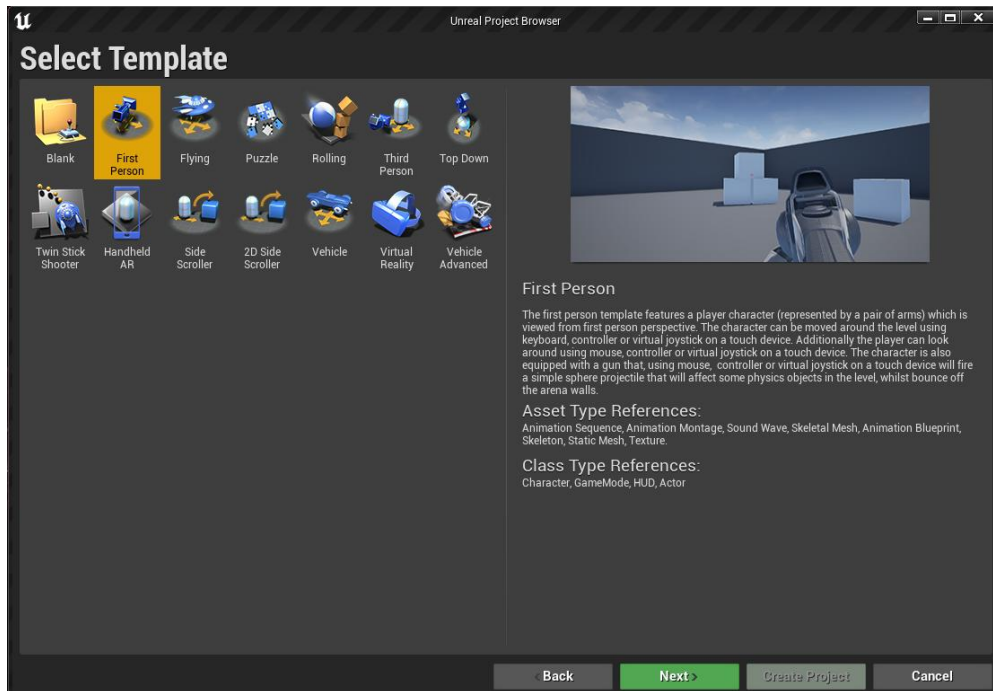
## 2.5 Περιβάλλον Ανάπτυξης

### 2.5.1 Αρχικοποίηση Έργου

Η ανάπτυξη έργου ξεκινά με την επιλογή νέου έργου στην αρχική οθόνη του Unreal Engine επιλέγοντας ένα από τα διαθέσιμα New Project Categories, όπως φαίνεται στην Εικόνα 2.4. Για το άνοιγμα ενός υπάρχοντος έργου, ανοίγουμε την αρχική οθόνη του Unreal Engine και το επιλέγουμε από την λίστα Recent Projects. Στα πλαίσια της τρέχουσας εργασίας επιλέχθηκε η κατηγορία Games για την ανάπτυξη ενός ολοκληρωμένου διαδραστικού περιβάλλοντος και το πρότυπο First person, όπως παρουσιάζεται στην Εικόνα 2.5 [20].

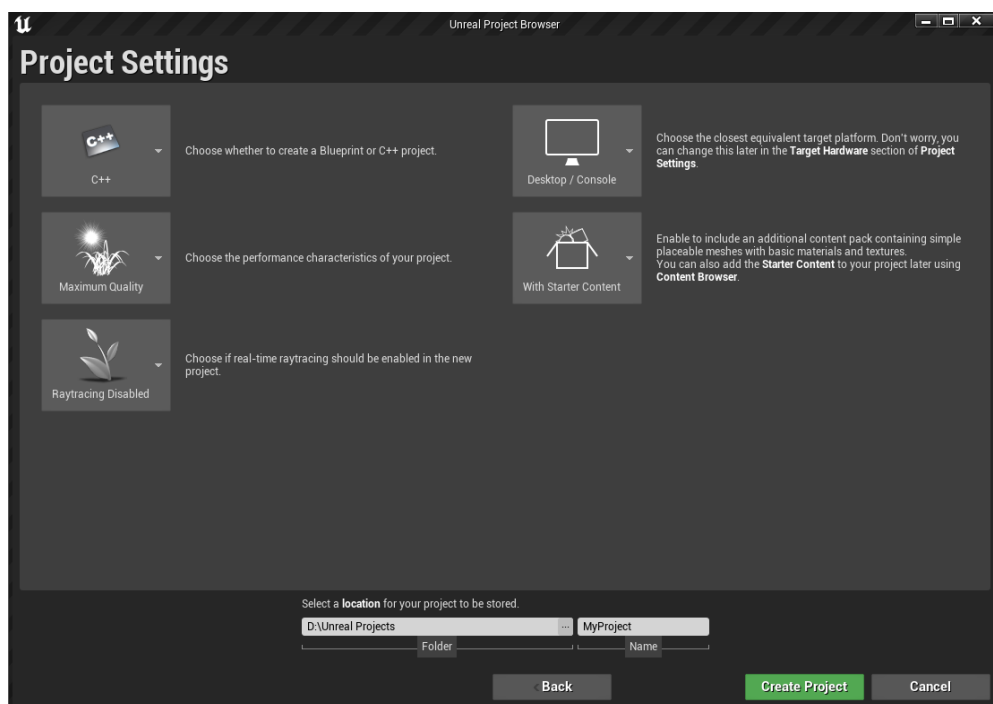


Εικόνα 2.4: Αρχική οθόνη Unreal Engine



Εικόνα 2.5: Επιλογή πρότυπου

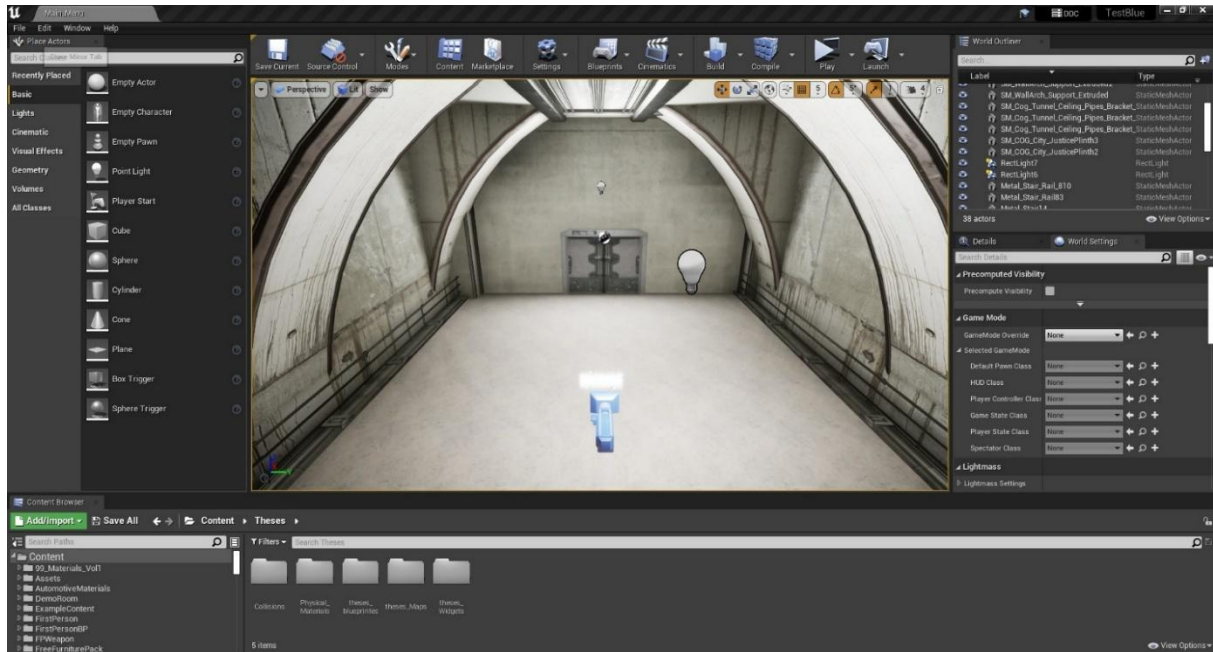
Στη συνέχεια, ορίζονται οι ρυθμίσεις του έργου. Έγινε αλλαγή του τύπου του έργου από Blueprint σε C++ για να γίνει η μεταγλώττιση του πρότυπου σε εκτελέσιμο και να δημιουργηθεί αυτόματα ένα αρχείο Visual Studio για επεξεργασία του κώδικα του έργου, όπως φαίνεται στην Εικόνα 2.6 [20].



Εικόνα 2.6: Αρχικές ρυθμίσεις έργου

## 2.5.2 Περιβάλλον Μηχανής (Unreal Engine)

Μετά την φόρτωση του έργου (project) στη μηχανή, ανοίγει το περιβάλλον ανάπτυξης το οποίο αποτελείται από πέντε βασικές ενότητες, χωρισμένες ανάλογα με την λειτουργικότητα τους, όπως παρουσιάζεται στην Εικόνα 2.7.

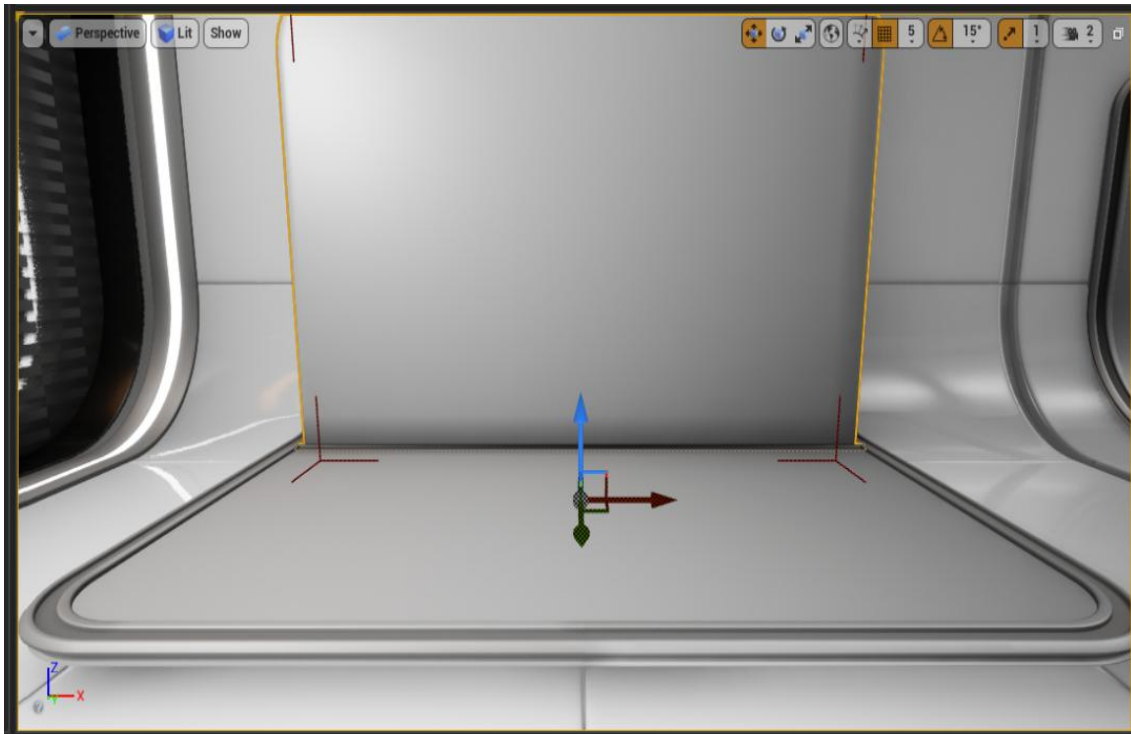


Εικόνα 2.7: Περιβάλλον μηχανής Unreal Engine

Στην αριστερή πλευρά υπάρχει ένα πλαίσιο μενού αντικειμένων (Place Actors) με συχνά χρησιμοποιούμενα αντικείμενα. Στο κέντρο βρίσκεται το παράθυρο προβολής τρισδιάστατων (Viewport) το οποίο εμπεριέχει στην πάνω αριστερή γωνία τέσσερα μενού με ρυθμίσεις, Viewport Options, Perspective, Lit και Show τα οποία καθορίζουν πως θα είναι ορατά τα περιεχόμενα του παραθύρου και στην πάνω δεξιά γωνία διάφορα κουμπιά για το χειρισμό των αντικειμένων και της editor κάμερας στο χώρο. Στην δεξιά πλευρά υπάρχουν τρεις καρτέλες. Η δομική λίστα αντικειμένων του κόσμου (World Outliner) που περιέχει όλα τα αντικείμενα που είναι φορτωμένα στο χώρο, ο περιηγητής ιδιοτήτων του τρέχοντος επιλεγμένου αντικειμένου (Details) και οι ρυθμίσεις του ενεργού επιπέδου (World Settings). Τέλος στο κάτω μέρος βρίσκεται ένα παράθυρο διαχείρισης αρχείων (Content Browser) που έχει πάντοτε ενεργό τον κατάλογο Content, τον κεντρικό κατάλογο περιεχομένου του έργου [20].

### 2.5.2.1 Παράθυρο Προβολής (Viewport)

Το παράθυρο προβολής περιλαμβάνει τον τρισδιάστατο χώρο μέσα στον οποίο αναπτύσσεται και διαμορφώνεται το περιβάλλον του έργου, όπως φαίνεται στην Εικόνα 2.8. Κατά την αρχικοποίηση του, ο χρήστης καλείται να επιλέξει ένα πρότυπο. Την πρώτη φορά που θα ανοίξει το παράθυρο προβολής θα έχει φορτωμένο το αντικείμενο χώρου (Level) που ορίστηκε κατά την αρχικοποίηση του έργου σύμφωνα με το επιλεγμένο πρότυπο. Ο αρχικός χώρος περιέχει κάποιες βασικές ρυθμίσεις φωτισμού και μερικά τρισδιάστατα αντικείμενα, μπορεί όμως να επεκταθεί και να χρησιμοποιηθεί για οποιαδήποτε ανάγκη του έργου [20].



Εικόνα 2.8: Παράθυρο Προβολής

### 2.5.2.2 Μενού Παραθύρου Προβολής

- Μενού επιλογών Παραθύρου Προβολής (Viewport Options) παρέχει βασικές ρυθμίσεις απόδοσης για το παράθυρο προβολής [21].
- Μενού Προοπτικής (Perspective) με τους τρόπους απεικόνισης ως προς την οπτική γωνία.
- Μενού Φωτισμού (Lit) με τους τρόπους απεικόνισης του χώρου ως προς το φωτισμό και τα τρισδιάστατα αντικείμενα.
- Μενού Επιλογών Εμφάνισης (Show) με δυνατότητα προβολής/απόκρυψης στοιχείων μέσα στο viewport. [22]

### 2.5.2.3 Αντικείμενα στο χώρο

Για να προστεθεί υλικό μέσα στον τρισδιάστατο χώρο είτε επιλέγεται μια κλάση από το αριστερό μενού κλάσεων είτε ένα αντικείμενο από το κάτω μενού, στον διαχειριστή αρχείων. Τα αντικείμενα τοποθετούνται σε συντεταγμένες X,Y,Z στον τρισδιάστατο χώρο κυρίως με την μέθοδο drag 'n drop. Επιπλέον, ισχύει η διαγραφή, αντιγραφή και αποκοπή ενός ή περισσότερων αντικειμένων έτσι ώστε ένα περιβάλλον που στήθηκε σε ένα επίπεδο να μπορεί να επαναληφθεί σε άλλο σημείο στο χώρο ή να αντιγραφεί σε ένα καινούργιο επίπεδο. Σε αυτή την περίπτωση το αντικείμενο που αντιγράφηκε κρατάει τις πληροφορίες για Location, Rotation και Scale που είχε στο προηγούμενο επίπεδο [20].

Για την τροποποίηση τους μέσα στον χώρο υπάρχει ένα κανονικό σύστημα αξόνων, το Gizmo. Οι άξονες του Gizmo αλλάζουν σε βέλη για μετακίνηση, καμπύλες για περιστροφή και τετράγωνα για μεγέθυνση/σμίκρυνση των επιλεγμένων αντικειμένων. Κρατώντας πατημένο έναν ή περισσότερους άξονες το αντικείμενο θα εκτελέσει την αντίστοιχη τροποποίηση προς αυτό τον άξονα, π.χ. αν οι άξονες είναι βέλη το αντικείμενο θα μετακινηθεί προς την κατεύθυνση των επιλεγμένων αξόνων.

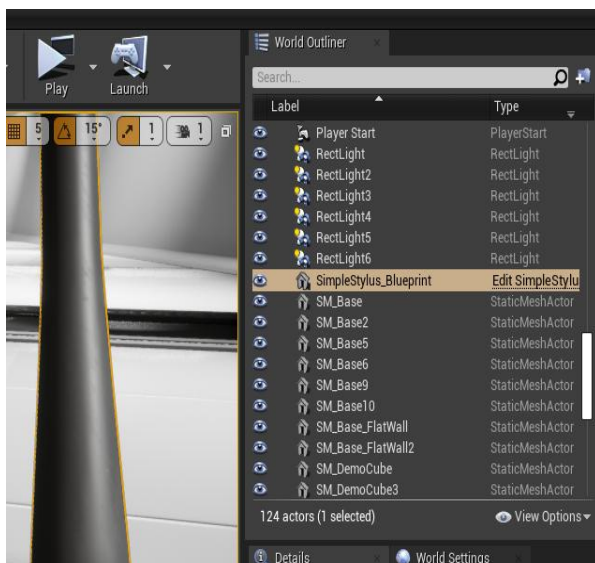
Το πόσο θα μετακινείται το αντικείμενο σε κάθε μετατόπιση ορίζεται από Object Snapping to Grid. Αλλάζοντας το μέγεθος του Grid η μετακίνηση θα πραγματοποιείται κατά μια ή περισσότερες μονάδες

την φορά. Αντίστοιχα υπάρχει το Rotation Grid για την περιστροφή και Scale Grid για την μεγέθυνση/σμίκρυνση. Οι αλλαγές ισχύουν για κάθε αντικείμενο ξεχωριστά καθώς αποτελούν ανεξάρτητα χωρία στα οποία μπορεί να αναφερθεί μεμονωμένα (instances) ο χρήστης. Επίσης, υπάρχει η δυνατότητα ένα αντικείμενο να μετακινηθεί και να περιστραφεί σύμφωνα με το σύστημα αξόνων του χώρου (global), διαφορετικά σύμφωνα με την εκάστοτε περιστροφή του (local) ή σύμφωνα με τις εφαπτόμενες ευθείες που διαπερνούν το κέντρο του αντικειμένου όταν αυτό βρίσκεται σε δεδομένη περιστροφή (normal).

Πολλαπλά αντικείμενα μπορούν να ομαδοποιηθούν (Group) για να είναι πιο εύκολη η τροποποίηση τους στο χώρο. Αν τα αντικείμενα ανήκουν στην ίδια κλάση ή μοιράζονται ιδιότητες, όπως ίδιο μέγεθος σε έναν ή περισσότερους άξονες, αυτές μπορούν να τροποποιηθούν μαζικά όταν είναι επιλεγμένη η ομάδα. Οι ομάδες δεν είναι απόλυτες. Μπορούν να ξεκλειδώσουν (Unlock) για την επιλογή ενός ή περισσότερων αντικειμένων που ανήκουν σε αυτές και να κλειδώσουν (Lock) όταν έχουν ολοκληρωθεί οι αλλαγές. Όπως επίσης να αναδιοργανωθούν (Regroup) για την προσθήκη νέων αντικειμένων ή να διασπαστούν (Ungroup). Επιπλέον, αντικείμενα μπορούν να αφαιρεθούν από μια ομάδα χωρίς να γίνει η διάσπαση της.

### 2.5.2.4 Παράθυρο περιήγησης χώρου (World Outliner)

Το παράθυρο περιηγητή χώρου δείχνει κάθε φορά στο επίπεδο (Level) που είναι φορτωμένος στο viewport και περιλαμβάνει μια λίστα με όλα τα αντικείμενα που έχουν τοποθετηθεί σε αυτόν, όπως φαίνεται στην Εικόνα 2.9 [20]. Για κάθε αντικείμενο της λίστας φαίνεται η ορατότητα του στον χώρο, το όνομα του και ο τύπος του. Όταν επιλεγεί ένα αντικείμενο στο παράθυρο προβολής γίνεται επιλογή



Εικόνα 2.9: World Outliner

του ίδιου αντικειμένου στον περιηγητή χώρου.

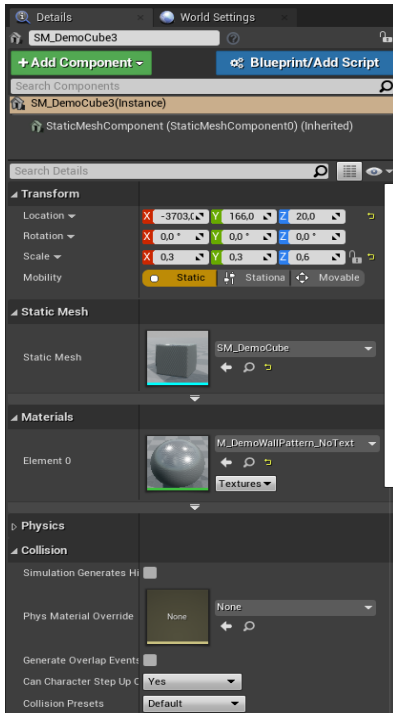
Οι αλλαγές που πραγματοποιούνται στα αντικείμενα της λίστας όπως διαγραφή, αποκοπή, επικόλληση και δημιουργία διπλότυπου ενός αντικειμένου εφαρμόζονται αυτόματα και στα αντικείμενα του παραθύρου προβολής και αντίστροφα.

Επίσης, το παράθυρο αυτό λειτουργεί συνδυαστικά με το παράθυρο λεπτομερειών αντικειμένου, το οποίο δείχνει για κάθε αντικείμενο που έχουμε επιλεγμένο, όλες τις ορατές στη μηχανή, ιδιότητες [20].

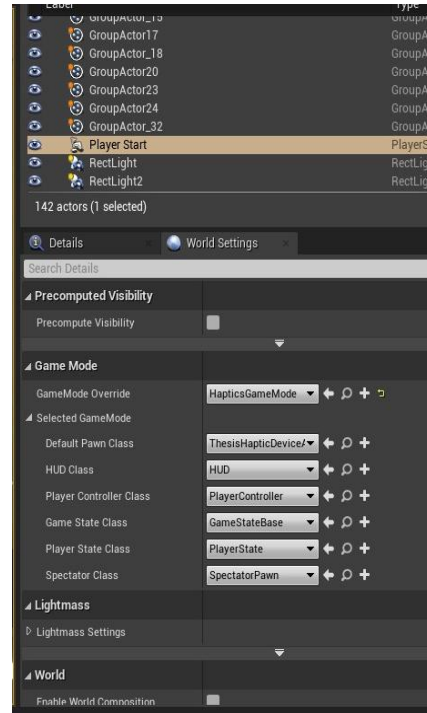
### 2.5.2.5 Παράθυρο λεπτομερειών αντικειμένου (Details)

Σε αυτό το παράθυρο γίνεται η παραμετροποίηση του κάθε αντικειμένου στον τρισδιάστατο χώρο, όπως φαίνεται στην Εικόνα 2.10. Για κάθε αναφορά (instance) ενός αντικειμένου μπορεί να αλλάξει το στατικό μοντέλο του (static mesh), το υλικό του (material), οι πληροφορίες συμπεριφοράς του σε δυνάμεις φυσικής (physics), οι πληροφορίες σύγκρουσης (collision), η δυνατότητα του αντικειμένου να παράγει σκιά, καθώς επίσης αν θα είναι κρυφό ή φανερό εντός του εκτελέσιμου έργου (In-Game

visibility) [20]. Επιπλέον, αλλαγές στις συντεταγμένες, την περιστροφή και το μέγεθος ενός αντικειμένου μπορούν να γίνουν απευθείας από το παράθυρο λεπτομερειών.



Εικόνα 2.10: Details



Εικόνα 2.11: World Settings

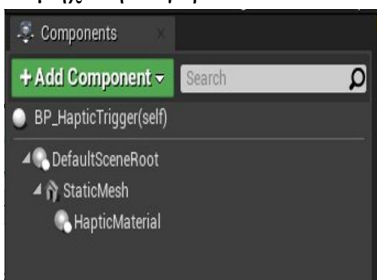
### 2.5.2.6 Παράθυρο ρυθμίσεων επιπέδου (World Settings)

Το παράθυρο αυτό περιλαμβάνει ρυθμίσεις που αφορούν το ενεργό επίπεδο (Level) και είναι χωρισμένες σε κατηγορίες ανάλογα με τις πτυχές του επιπέδου που επηρεάζουν, όπως παρουσιάζεται την Εικόνα 2.11. Ορισμένες από αυτές είναι γενικές, όπως αυτές που επηρεάζουν το Game Mode και την πλοήγηση (navigation) και άλλες, πιο εξειδικευμένες ομάδες ρυθμίσεων, που χρησιμοποιούνται για να ρυθμιστεί ο φωτισμός, ο ήχος και η φυσική. Τα Game Modes καθορίζουν τους κανόνες του εκτελέσιμου έργου για το τρέχον επίπεδο. Η ρύθμιση GameModeOverride επιτρέπει την επιλογή ενός συγκεκριμένου τύπου παιχνιδιού όταν ο παίκτης βρίσκεται εντός αυτού του επιπέδου. Σύμφωνα με το επιλεγμένο Game Mode που έχει φορτωθεί μπορεί να αλλάξει ο Player και η συμπεριφορά του στο χώρο [20].

Στα πλαίσια της τρέχουσας πτυχιακής εργασίας έγινε χρήση του HapticsGameMode που δόθηκε από το πρόσθετο HapticsDirect της 3D Systems.

### 2.5.3 Αντικείμενα του Unreal Engine (Components)

Η μηχανή συγκροτείται από αντικείμενα τα οποία αποτελούν παράγωγα διαφόρων κλάσεων, καθεμία από τις οποίες ενσωματώνει σαφώς ορισμένες ή ευρύτερα προσδιορισμένες λειτουργίες, όπως φαίνεται στην Εικόνα 2.12. Πολλά από τα αντικείμενα αυτά είναι επαναχρησιμοποιήσιμα και αντλούν τα βασικά χαρακτηριστικά και ιδιότητές τους από μια υπερκείμενη κλάση, ενώ παράλληλα δύνανται να επεκταθούν λειτουργικά εντός των ορίων που θέτει η αρχιτεκτονική του συστήματος [23].



Εικόνα 2.12: Components

Κάθε αντικείμενο αποτελεί ένα Component σύμφωνα με την τεχνική ορολογία του Unreal Engine και κληρονομεί παραμέτρους από την κλάση Actor. Από τις πιο συνηθισμένες ιδιότητες των αντικειμένων είναι οι ρυθμίσεις που αφορούν την κινητικότητα τους, καθώς και η συμπεριφορά τους σε φαινόμενα σύγκρουσης και φυσικής αλληλεπίδρασης. Έπειτα χωρίζονται σε μεγάλες κατηγορίες σύμφωνα με την λειτουργικότητα τους όπως οι εξής:

- Στατικά – ακίνητα (Static Mesh), όλα τα αντικείμενα που αποτελούν το συμπαγή χώρο.
- Φώτα – πηγές φωτός, όλα τα αντικείμενα που μπορούν να αλλάξουν τον φωτισμό.
- Χαρακτήρες (Character), όλες οι έμβιες οντότητες εντός του έργου συμπεριλαμβανομένου και του πρωταγωνιστή (Player).

### 2.5.3.1 Ιδιότητες Αντικειμένων – Κινητικότητα

Τα αντικείμενα διακρίνονται σε στατικά (static), με περιορισμένη κινητικότητα (Stationary) και τα κινούμενα (movable) ανάλογα με την κινητικότητά τους.

Στα απολύτως στατικά αντικείμενα (static), η θέση τους παραμένει αμετάβλητη καθ' όλη τη διάρκεια εκτέλεσης της εφαρμογής. Η τοποθέτησή τους στον χώρο πραγματοποιείται αποκλειστικά μέσω του Editor, χωρίς δυνατότητα μεταβολής κατά το runtime. Σε αυτή την κατηγορία ανήκουν συνήθως αρχιτεκτονικά μοντέλα, στοιχεία περιβάλλοντος και αντικείμενα που αναπαριστούν τον μακρινό κόσμο (LOD).

Αντίστοιχα, τα αντικείμενα με περιορισμένη κινητικότητα (Stationary) αποτελούν στατικά στοιχεία στα οποία είναι δυνατή η εφαρμογή δυνάμεων φυσικής, χάρη στη χρήση μιας κάψουλας (Physics Asset) που τα περιβάλλει. Παρότι διαθέτουν δυνατότητα φυσικής αλληλεπίδρασης, παραμένουν σε σταθερή θέση έως ότου δεχθούν εξωτερική επίδραση, όπως πρόσκρουση από άλλο αντικείμενο ή από χαρακτήρα.

Τέλος, τα πλήρως κινούμενα αντικείμενα (movable) έχουν τη δυνατότητα να μεταβάλλουν τόσο τη θέση όσο και τις ιδιότητές τους σε πραγματικό χρόνο, χωρίς ουσιαστικούς περιορισμούς. Υπόκεινται σε δυνάμεις φυσικής και μπορούν να τροποποιούν οποτεδήποτε τις φυσικές παραμέτρους τους, όπως τη μάζα ή τη γεωμετρία της κάψουλας σύγκρουσης (collision) τους. [23]

### 2.5.3.2 Ιδιότητες Αντικειμένων – Συγκρούσεις

Κάθε αντικείμενο συνοδεύεται από γεωμετρία πρόσκρουσης (Collider), η οποία ορίζει τα φυσικά του όρια εντός του εικονικού χώρου. Η επιλογή του τύπου πρόσκρουσης γίνεται σύμφωνα με τις λειτουργικές απαιτήσεις του αντικειμένου. Η ανίχνευση και η επεξεργασία γεγονότων πρόσκρουσης πραγματοποιείται όταν δύο αντικείμενα διαθέτουν κατάλληλα ορισμένη γεωμετρία πρόσκρουσης (Collision Mesh) και, ταυτόχρονα, έχει ενεργοποιηθεί η ρύθμιση που αποτρέπει την επικάλυψη (overlap) μεταξύ τους. [24], [25].

### 2.5.3.3 Κατηγορίες Αντικειμένων

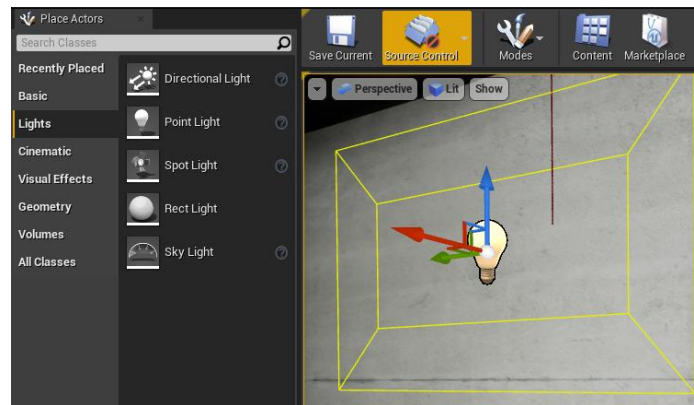
#### 2.5.3.3.1 Actor

Τα περισσότερα αντικείμενα που θα συναντήσουμε εντός ενός τρισδιάστατου χώρου είναι αντικείμενα με γονέα την κλάση Actor. Οι Actors μπορούν, υπό μία έννοια, να θεωρηθούν ως δομές-δοχεία που περιλαμβάνουν ειδικούς τύπους αντικειμένων, γνωστούς ως Components. Κάθε Actor διαθέτει την

παράμετρο RootComponent, η οποία καθορίζει ποιο Component λειτουργεί ως η ρίζα της ιεραρχίας του Actor. Οι Actors αποθηκεύουν δεδομένα μετασχηματισμού και, κατά συνέπεια, διαθέτουν θέση, περιστροφή ή κλίμακα. Ο χωρικός μετασχηματισμός του Actor σε καθολικό επίπεδο γίνεται μέσω του Root Component, ενώ ο τοπικός μετασχηματισμός πραγματοποιείται μέσω της μετακίνησης των εσωτερικών Components. Τα υπόλοιπα συνδεδεμένα Components φέρουν μετασχηματισμό σχετικό ως προς το Component στο οποίο είναι προσαρτημένα, σχηματίζοντας έτσι μια ιεραρχική δομή. Μπορούν να χρησιμοποιηθούν με διαφορετικούς τρόπους, όμως όλα διαθέτουν ορισμένα κοινά χαρακτηριστικά, όπως τη θέση, την περιστροφή και το μέγεθός τους στον χώρο, τις ιδιότητες σύγκρουσης και τον τρόπο κίνησής τους [25].

### 2.5.3.3.2 Φωτισμός

Η μηχανή παρέχει ένα ολοκληρωμένο σύνολο εργαλείων και λειτουργιών για την προσαρμογή του φωτισμού στον εικονικό χώρο, υποστηρίζοντας τόσο προϋπολογισμένο όσο και δυναμικό φωτισμό, όπως φαίνεται στην Εικόνα 2.13. Ο προϋπολογισμένος φωτισμός αξιοποιεί lightmaps και το σύστημα Lightmass για τον υπολογισμό στατικών σκιάσεων και έμμεσου φωτισμού, ενώ ο δυναμικός φωτισμός επιτρέπει μεταβολές σε πραγματικό χρόνο, δίνοντας τη δυνατότητα στα φώτα και τις σκιές να ανταποκρίνονται άμεσα στις αλλαγές της σκηνής.



Εικόνα 2.13: Φωτισμός

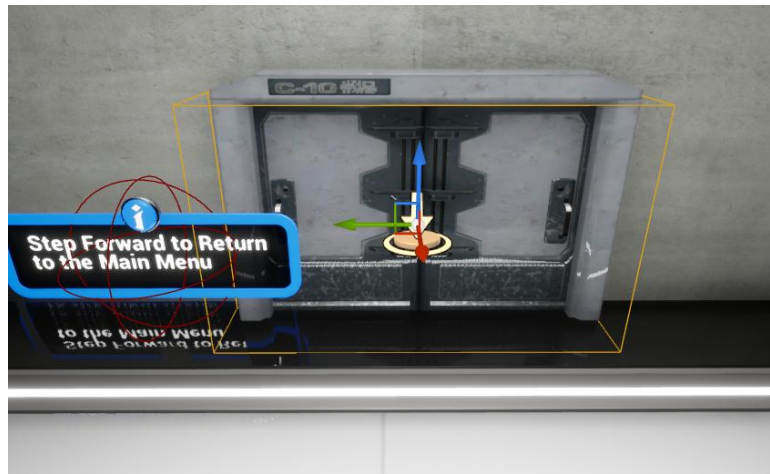
Το σύστημα φωτισμού υποστηρίζει ποικίλους τύπους πηγών φωτός και παρέχει εκτενή έλεγχο παραμέτρων όπως η εμβέλεια, η ένταση και το χρώμα του φωτισμού. Οι φωτεινές πηγές επηρεάζουν τις επιφάνειες που βρίσκονται εντός του εύρους δράσης τους, το οποίο ορίζεται σε εκατοστά, με ένταση φωτισμού που μπορεί να εκφράζεται σε φυσικές μονάδες μέτρησης, όπως τα lumens και οι καντέλες (cd). Επιπλέον, τα φώτα μπορούν να αξιοποιούν συναρτήσεις φωτισμού μέσω υλικών (Material Functions), επιτρέποντας την περαιτέρω παραμετροποίηση της φωτεινής συμπεριφοράς. Τέλος, το σύστημα περιλαμβάνει εργαλεία για τη δημιουργία περιβαλλοντικού φωτισμού μέσω στοιχείων όπως ο ουρανός, η ομίχλη και η ατμόσφαιρα, συμβάλλοντας στη ρεαλιστική και συνεκτική απόδοση του εικονικού περιβάλλοντος [26].

### 2.5.3.3.3 Πυροδοτητές (Triggers)

Οι πυροδοτητές (Triggers) είναι Actors που χρησιμοποιούνται για την ενεργοποίηση γεγονότων όταν αλληλεπιδρούν με κάποιο άλλο αντικείμενο στο επίπεδο (level). Όλοι οι προεπιλεγμένοι πυροδοτητές είναι παρόμοιοι ως προς τη λειτουργικότητά τους και διαφοροποιούνται κυρίως ως προς το σχήμα της περιοχής επιρροής τους.

Τα σχήματα αυτά περιλαμβάνουν το κουτί (box), την κάψουλα (capsule) και τη σφαίρα (sphere). Οι Triggers είναι διάφανοι εντός της μηχανής και διαθέτουν γεωμετρία σύγκρουσης (Collision Mesh),

επιτρέποντας σε στατικά και δυναμικά αντικείμενα του χώρου να τους διαπερνούν χωρίς να παρεμποδίζεται η κίνησή τους, όπως φαίνεται στην Εικόνα 2.14. Συνήθως υλοποιούνται με τη μορφή Blueprint, ώστε η λειτουργία και η συμπεριφορά τους να μπορούν να οπτικοποιηθούν και να τροποποιηθούν άμεσα.



Εικόνα 2.14: Πυροδοτητές

Τα γεγονότα που μπορούν να ενεργοποιηθούν μέσω ενός Trigger είναι ποικίλα και αξιοποιούνται για τον έλεγχο της ροής του έργου. Ενδεικτικά παραδείγματα περιλαμβάνουν την ενεργοποίηση ή απενεργοποίηση αντικειμένων, καθώς και την αλλαγή της κατάστασης της σκηνής, όπως το άνοιγμα θυρών ή η μετάβαση σε επόμενο επίπεδο. Τα περισσότερα από αυτά τα γεγονότα ενεργοποιούνται ως απόκριση σε σύγκρουση ή επικάλυψη με άλλο αντικείμενο, για παράδειγμα όταν ο παίκτης εισέρχεται ή εξέρχεται από την περιοχή επιρροής του Trigger [27].

### 2.5.4 Το Σύστημα Blueprint (Blueprint Editor)

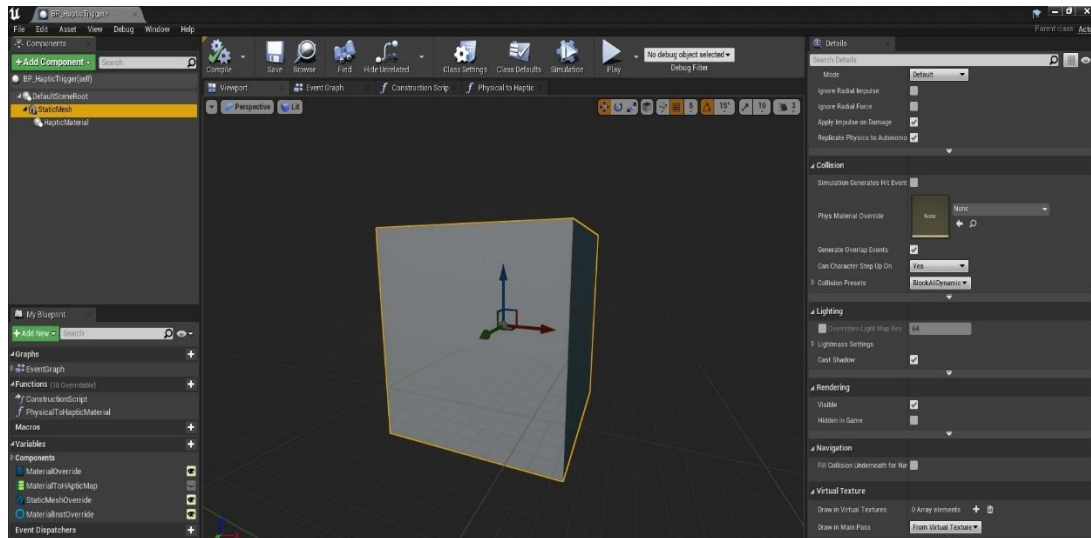
Το Blueprint Visual Scripting System του Unreal Engine αποτελεί ένα ολοκληρωμένο σύστημα οπτικού προγραμματισμού που επιτρέπει τη δημιουργία και τον έλεγχο της λογικής του gameplay χωρίς την άμεση συγγραφή παραδοσιακού πηγαίου κώδικα. Το σύστημα βασίζεται σε γραφήματα κόμβων (node-based graphs), τα οποία συνδέονται μεταξύ τους ώστε να ορίζουν ροές εκτέλεσης, λογικές συνθήκες και λειτουργικές συμπεριφορές. Μέσω των Blueprints είναι δυνατή η επέκταση υπάρχουσών κλάσεων, η τροποποίηση ιδιοτήτων αντικειμένων και η διαχείριση των Components εντός του περιβάλλοντος του Unreal Editor [28], [29].

Το Unreal Engine υποστηρίζει διάφορους τύπους Blueprints, καθένας εκ των οποίων εξυπηρετεί διαφορετικές ανάγκες ανάπτυξης. Στα πλαίσια του τρέχοντος έργου έγινε χρήση δύο τύπων Blueprints:

- Των Blueprint Class τα οποία επιτρέπουν τη δημιουργία νέων κλάσεων που βασίζονται σε υπάρχουσες κλάσεις του συστήματος. Τα αντικείμενα αυτά μπορούν να τοποθετηθούν στον εικονικό κόσμο και να λειτουργούν ως Actors.
- Και των Level Blueprint που αφορούν τη διαχείριση γεγονότων που σχετίζονται με ένα συγκεκριμένο επίπεδο, επιτρέποντας τον έλεγχο καθολικής συμπεριφοράς και αλληλεπίδρασης.

Η ανάπτυξη και επεξεργασία των Blueprints πραγματοποιείται μέσω του Blueprint Editor, όπως φαίνεται στην Εικόνα 2.15, ο οποίος υιοθετεί έναν σχεδιασμό που εξαρτάται από πλαίσιο χρήσης (context-sensitive). Ο σχεδιασμός αυτός διευκολύνει την άμεση πρόσβαση στις κατάλληλες λειτουργίες, ανάλογα με το εκάστοτε αντικείμενο και το στάδιο της ανάπτυξης, ενώ ταυτόχρονα

προσφέρει την απαιτούμενη ευελιξία για την υλοποίηση πιο σύνθετων ή μη τυπικών σεναρίων [28], [29].



Εικόνα 2.15: Blueprint Editor

Το περιβάλλον του Blueprint Editor αποτελείται από ένα σύνολο επιμέρους παραθύρων και εργαλείων για τη δημιουργία και επεξεργασία των Blueprints. Περιλαμβάνει μια βασική γραμμή εργαλείων (Tool Bar) με συχνά χρησιμοποιούμενες λειτουργίες, όπως οι εντολές Compile και Save, καθώς και ένα παράθυρο Components, μέσω του οποίου πραγματοποιείται η εισαγωγή νέων Components και η διαχείριση των ήδη υπαρχόντων. Επιπλέον, διαθέτει μια καρτέλα διαχείρισης των γραφημάτων, συναρτήσεων και μεταβλητών του τρέχοντος Blueprint [30], ένα κεντρικό παράθυρο επεξεργασίας για την ανάπτυξη της λογικής και της διάταξης, καθώς επίσης και ένα παράθυρο λεπτομεριών (Details) στο οποίο εμφανίζονται οι ιδιότητες του επιλεγμένου Component [29], [31].

### 2.5.4.1 Δομή και Λειτουργία ενός Blueprint

#### 2.5.4.1.1 Event Graph

Το Event Graph αποτελεί τον βασικό μηχανισμό ορισμού της δυναμικής συμπεριφοράς ενός Blueprint και χρησιμοποιείται για τον καθορισμό της λογικής που εκτελείται κατά τη διάρκεια του gameplay. Σε αυτό περιλαμβάνονται γεγονότα (events), όπως η έναρξη του παιχνιδιού, η είσοδος ή έξοδος από περιοχές σύγκρουσης και η αλληλεπίδραση με τον παίκτη, καθώς και συναρτήσεις που ενεργοποιούνται ως απόκριση στα γεγονότα αυτά. Λειτουργεί ως ο κεντρικός χώρος σύνδεσης κόμβων και ροών εκτέλεσης, επιτρέποντας τον έλεγχο της συμπεριφοράς των αντικειμένων σε πραγματικό χρόνο. Παράλληλα, η χρήση Functions και Variables διευκολύνει την οργάνωση της λογικής σε επαναχρησιμοποιήσιμες και παραμετροποιήσιμες μονάδες, ενισχύοντας τη δομημένη ανάπτυξη και τη συντηρησιμότητα του έργου. Μέσω αυτής της προσέγγισης, το σύστημα Blueprints υποστηρίζει την ανάπτυξη σύνθετων μηχανισμών αλληλεπίδρασης με σαφή και επεκτάσιμη δομή [29], [32].

#### 2.5.4.1.2 Construction Script

Το Construction Script αποτελεί ένα βασικό τμήμα του Blueprint Editor και χρησιμοποιείται για την αρχικοποίηση και τη ρύθμιση ενός Blueprint κατά τη στιγμή της δημιουργίας ή της τροποποίησής του μέσα στον Editor. Εκτελείται κάθε φορά που ένα instance του Blueprint τοποθετείται ή ενημερώνεται στη σκηνή, καθώς και όταν μεταβάλλονται οι ιδιότητές του, πριν από την έναρξη του gameplay. Μέσω

του Construction Script, είναι δυνατή η δυναμική παραμετροποίηση των Components του Blueprint, όπως η αλλαγή θέσης, κλίμακας ή υλικών, καθώς και η αυτόματη δημιουργία ή προσαρμογή στοιχείων με βάση προκαθορισμένες συνθήκες. Με τον τρόπο αυτό, το Construction Script συμβάλλει στη δημιουργία ευέλικτων και επαναχρησιμοποιήσιμων αντικειμένων, επιτρέποντας την άμεση οπτική ανατροφοδότηση των αλλαγών μέσα στο περιβάλλον του Editor [29], [32].

### 2.5.4.1.3 Functions

Οι Functions στο σύστημα Blueprints χρησιμοποιούνται για την οργάνωση της λογικής ενός Blueprint σε διακριτές και επαναχρησιμοποιήσιμες μονάδες. Κάθε συνάρτηση εκτελεί μια συγκεκριμένη λειτουργία και μπορεί να καλείται από το Event Graph, από άλλες συναρτήσεις ή ακόμη και από εξωτερικά Blueprints, εφόσον αυτό επιτρέπεται από τις ρυθμίσεις πρόσβασης [29], [32].

### 2.5.5 Μενού

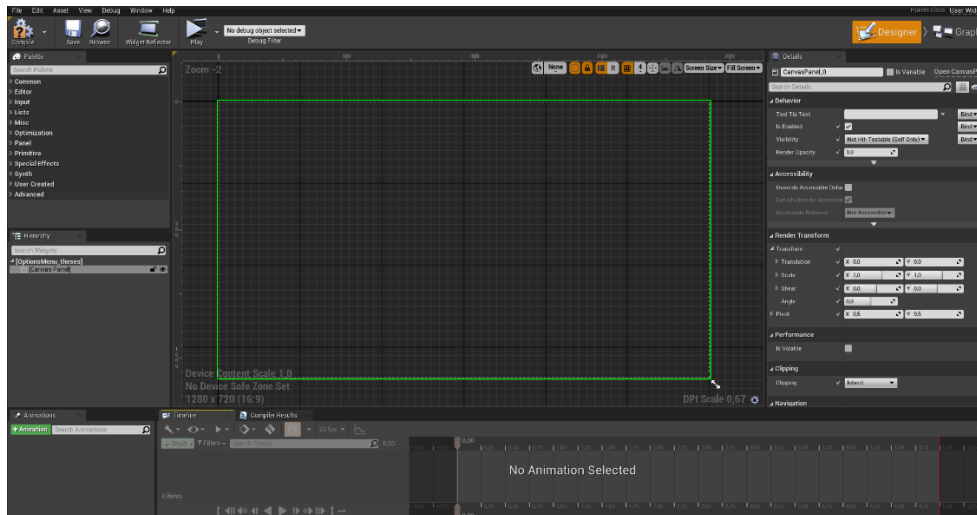
Τα μενού στο Unreal Engine μπορούν να κατασκευαστούν με δύο τρόπους, με ένα αυτόματο εργαλείο που ονομάζεται UMG UI Designer, που χρησιμοποιήθηκε στα πλαίσια του τρέχοντος έργου και με μια διεπαφή προγραμματιστή, το Slate UI Framework.

Το Unreal Motion Graphics UI Designer (UMG) είναι ένα οπτικό εργαλείο δημιουργίας διεπαφών χρήστη (UI), το οποίο μπορεί να χρησιμοποιηθεί για τη σχεδίαση στοιχείων διεπαφής, όπως HUDs (Heads-Up Display) εντός έργου, μενού και άλλα γραφικά στοιχεία που προορίζονται για την αλληλεπίδραση με τον χρήστη [33].

Στον πυρήνα του UMG βρίσκονται τα Widgets, τα οποία αποτελούν προκατασκευασμένα λειτουργικά στοιχεία που μπορούν να χρησιμοποιηθούν για την κατασκευή της διεπαφής, όπως κουμπιά, πλαίσια επιλογής, μπάρες προόδου και παρόμοια στοιχεία. Η επεξεργασία των Widgets πραγματοποιείται μέσα από ένα εξειδικευμένο Widget Blueprint, το οποίο περιλαμβάνει δύο βασικές καρτέλες: η καρτέλα Designer επιτρέπει τη διαμόρφωση της οπτικής διάταξης της διεπαφής και την εφαρμογή βασικών λειτουργιών, ενώ η καρτέλα Graph παρέχει τη λογική και τη λειτουργικότητα που καθορίζουν τη συμπεριφορά των χρησιμοποιούμενων Widgets [33], [34].

#### 2.5.5.1 Καρτέλα Designer

Από προεπιλογή, ο Widget Blueprint Editor ανοίγει στην καρτέλα Designer όταν ανοίγει ένα Widget Blueprint. Η καρτέλα Designer αποτελεί την οπτική αναπαράσταση της διάταξης της διεπαφής και παρέχει μια σαφή εικόνα του τρόπου με τον οποίο θα εμφανίζεται η οθόνη κατά τη διάρκεια του εκτελέσιμου, όπως παρουσιάζεται στην Εικόνα 2.16.



Εικόνα 2.16: UMG UI Designer

Το περιβάλλον του Designer περιέχει έξι παράθυρα χωρισμένα ανάλογα με τη λειτουργία τους.

1. Το βασικό Tool Bar από συχνά χρησιμοποιούμενες λειτουργίες του Blueprint Editor, όπως οι εντολές Compile, Save και Play.
2. Το μενού Palette που περιέχει τη λίστα των διαθέσιμων Widgets που μπορούν να μεταφερθούν με drag-and-drop στον Visual Designer και εμφανίζει όλες τις κλάσεις που κληρονομούν από την κλάση UWidget.
3. Το παράθυρο ιεραρχίας (Hierarchy) στο οποίο εμφανίζει τη δομή ιεραρχίας (γονέα–παιδιού) του Widget του χρήστη και παρέχει τη δυνατότητα μεταφοράς Widgets εντός του παραθύρου για την οργάνωσή τους.
4. Το κεντρικό παράθυρο δημιουργίας (Visual Designer) που αποτελεί την οπτική αναπαράσταση της διάταξης της διεπαφής. Τα Widgets που τοποθετούνται στον Visual Designer μπορούν να τροποποιηθούν και να μετασχηματιστούν απευθείας μέσα στο παράθυρο.
5. Το παράθυρο λεπτομερειών (Details) όπου εμφανίζονται οι ιδιότητες του Widget που είναι επιλεγμένο τη δεδομένη στιγμή.
6. Το παράθυρο Animation το οποίο αποτελεί το σύστημα χρονογραμμής (animation track) του UMG και επιτρέπει τη δημιουργία κινούμενων σχεδίων για τα Widgets μέσω keyframes

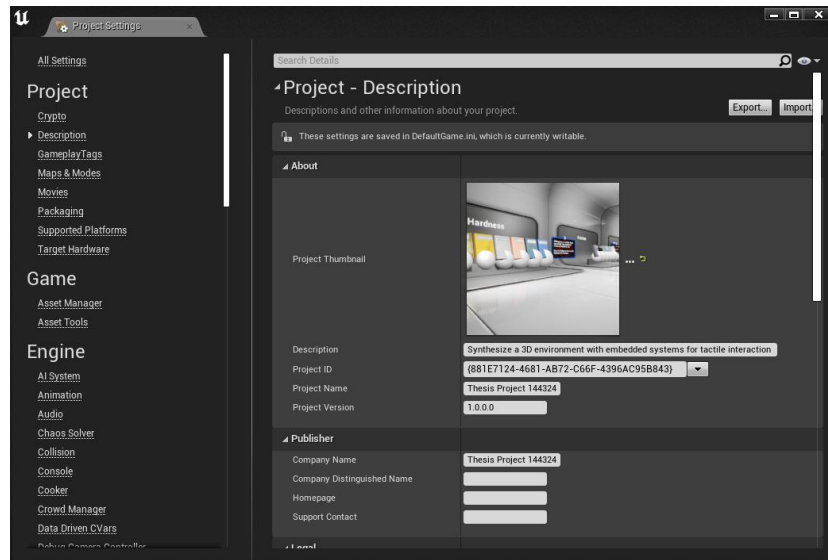
Επιπλέον, στην δεξιά πλευρά του παραθυρικού περιβάλλοντος υπάρχει και το Editor Mode που επιτρέπει την εναλλαγή του UMG Widget Blueprint Editor μεταξύ των λειτουργιών Designer και Graph [34].

### 2.5.5.2 Καρτέλα Graph

Η καρτέλα Graph λειτουργεί με τον ίδιο τρόπο όπως ο προεπιλεγμένος Blueprint Editor. Σε αυτή την καρτέλα δημιουργούνται τα δίκτυα κόμβων και συνδέσεων (nodes και wires) που καθορίζουν τη σεναριακή συμπεριφορά του συστήματος. Οι κόμβοι μπορούν να επιλεγούν εύκολα με ένα κλικ και να μετακινηθούν μέσω drag-and-drop [35].

### 2.5.6 Ρυθμίσεις (Project Settings)

Η πρόσβαση στο περιβάλλον Project Settings πραγματοποιείται μέσω του μενού Edit της κύριας γραμμής εργαλείων του Unreal Editor, όπως φαίνεται στην Εικόνα 2.17.



Εικόνα 2.17: Project Settings

Το περιβάλλον Project Settings αποτελεί ένα κεντρικό συστατικό της μηχανής Unreal Engine 4.27, το οποίο καθιστά δυνατή τη λεπτομερή διαμόρφωση των παραμέτρων λειτουργίας ενός έργου. Ως ένα σύστημα ρυθμίσεων, παρέχει μια οπτική διεπαφή για την τροποποίηση κρίσιμων μεταβλητών που καθορίζουν τόσο τη συμπεριφορά του ίδιου του έργου όσο και το πλαίσιο λειτουργίας των βασικών υποσυστημάτων της μηχανής, όπως το σύστημα γραφικών, η φυσική προσομοίωση και οι μηχανισμοί εισόδου [36].

### 2.5.6.1 Δομή και Οργάνωση

Το περιβάλλον ρυθμίσεων του Unreal Engine οργανώνεται σε έναν ιεραρχικά δομημένο πίνακα κατηγοριών και επιμέρους τμημάτων, όπου κάθε κατηγορία συγκεντρώνει θεματικά συναφείς επιλογές διαμόρφωσης. Η δομή αυτή διευκολύνει τη συστηματική προσέγγιση των ρυθμίσεων και επιτρέπει την αποτελεσματική παραμετροποίηση του έργου. Συνοπτικά, οι κύριες κατηγορίες του παραθύρου Project Settings είναι οι εξής:

- **Project:** Περιλαμβάνει ρυθμίσεις που αφορούν το ίδιο το έργο, όπως γενικές πληροφορίες, προεπιλεγμένοι χάρτες (default maps), καθώς και διαδικασίες πακεταρίσματος και διανομής της εφαρμογής.
- **Engine:** Αφορά τεχνικές ρυθμίσεις που επηρεάζουν τα βασικά υποσυστήματα της μηχανής Unreal Engine, συμπεριλαμβανομένων της φυσικής προσομοίωσης, του συστήματος ήχου, της δικτύωσης και της απόδοσης γραφικών (rendering).
- **Editor:** Περιέχει τις προτιμήσεις που καθορίζουν τη συμπεριφορά και την εμφάνιση του περιβάλλοντος ανάπτυξης, όπως οι μονάδες μέτρησης, οι ρυθμίσεις διεπαφής χρήστη (UI) και άλλες εργονομικές επιλογές.
- **Platforms:** Παρέχει επιλογές προσαρμογής του έργου σε συγκεκριμένες πλατφόρμες στόχου, όπως Android, iOS και Windows, εξασφαλίζοντας τη συμβατότητα και τη βελτιστοποίηση της εφαρμογής για κάθε περιβάλλον εκτέλεσης.
- **Plugins:** Περιλαμβάνει ρυθμίσεις που προέρχονται από εγκατεστημένα πρόσθετα (plugins), τα οποία επεκτείνουν ή τροποποιούν τις βασικές λειτουργικότητες της μηχανής, επιτρέποντας την ενσωμάτωση επιπλέον εργαλείων και δυνατοτήτων στο έργο. [36]

Στο πλαίσιο των ρυθμίσεων Project Settings, πραγματοποιήθηκαν αρχικά τροποποιήσεις στην κατηγορία Engine - Input, όπου προστέθηκαν νέα Action Mappings με σκοπό την υποστήριξη των

λειτουργιών που αναπτύχθηκαν για τις ανάγκες του έργου. Οι συγκεκριμένες ρυθμίσεις διασφαλίζουν τη σωστή αντιστοίχιση των ενεργειών του χρήστη με τις αντίστοιχες λειτουργίες της εφαρμογής και συμβάλλουν στην ομαλή και αποδοτική αλληλεπίδραση με το σύστημα. Παράλληλα, έγιναν αλλαγές σε επιλεγμένες ρυθμίσεις της κατηγορίας Project, οι οποίες θα αναλυθούν και θα επεξηγηθούν εκτενέστερα σε επόμενο κεφάλαιο της παρούσας εργασίας.

## 2.6 Επίλογος

Το Unreal Engine αποτελεί μια πλατφόρμα κατάλληλη για την ανάπτυξη διαδραστικών τρισδιάστατων περιβαλλόντων που απαιτούν αυξημένο βαθμό αλληλεπίδρασης και ρεαλισμού, καθώς παρέχει εκτεταμένες δυνατότητες παραμετροποίησης και ελέγχου της συμπεριφοράς του περιβάλλοντος.

Η οργανωμένη διεπαφή του Unreal Editor, σε συνδυασμό με τη χρήση του συστήματος Blueprints, διευκολύνει τη διαχείριση της λογικής και της αλληλεπίδρασης ακόμη και σε σύνθετα σενάρια. Τα χαρακτηριστικά αυτά καθιστούν τη μηχανή ικανή να συνδεθεί με εξωτερικά συστήματα, όπως το 3D Systems Touch, επιτρέποντας την υλοποίηση εικονικών περιβαλλόντων εμπλουτισμένων με απτικά ερεθίσματα και ενισχύοντας τη συνολική εμπειρία του χρήστη.

## **Κεφάλαιο 3ο: Ανάπτυξη εφαρμογής με Unreal Engine 4**

### **3.1 Εισαγωγή**

Η παρούσα εργασία αναπτύχθηκε με γνώμονα τη δημιουργία ενός πειραματικού χώρου που στοχεύει στην αξιοποίηση των δυνατοτήτων απτικής ανάδρασης του μηχανικού βραχίονα 3D Systems Touch. Κεντρικός στόχος του έργου ήταν ο σχεδιασμός και η υλοποίηση μιας πλατφόρμας η οποία παρέχει στους χρήστες μια ολοκληρωμένη και ρεαλιστική εμπειρία της αίσθησης της αφής, μέσα από την αλληλεπίδραση με ένα τρισδιάστατο εικονικό περιβάλλον [10], [12], [13].

Η διαδικασία σχεδιασμού και υλοποίησης του παρόντος έργου μπορεί να θεωρηθεί ως μια συνθετική πορεία μάθησης, η οποία οδήγησε σε συγκεκριμένα λειτουργικά αποτελέσματα. Η βασική πρόκληση εντοπίστηκε στον ορθό συνδυασμό γνώσεων που αφορούν τόσο την ανάπτυξη εικονικών περιβαλλόντων στο Unreal Engine όσο και την ενσωμάτωση απτικών συστημάτων, με στόχο τη δημιουργία ενός ενιαίου και συνεκτικού συστήματος αλληλεπίδρασης. Η μεθοδολογία που ακολουθήθηκε βασίστηκε σε γενικές αρχές ανάπτυξης διαδραστικών εφαρμογών και εικονικών περιβαλλόντων.

Η διαθεσιμότητα των εργαλείων και των βασικών υποδομών του οικοσυστήματος της Epic Games, σε συνδυασμό με την ενσωμάτωση του 3D Systems Touch, κατέστησε εφικτή την υλοποίηση του έργου στο πλαίσιο ακαδημαϊκής εργασίας και με περιορισμένους πόρους.

### **3.2 Περιπτώσεις Χρήσης και Επίπεδα Απτικής Αλληλεπίδρασης**

Στην παρούσα πτυχιακή εργασία αναπτύχθηκαν περιπτώσεις χρήσης με τρία διαφορετικά επίκεντρα ενδιαφέροντος. Ο πειραματικός χώρος δομήθηκε σε επίπεδα (Levels), καθένα από τα οποία αντιστοιχεί σε διαφορετική απτική εμπειρία για τον χρήστη.

Στο πρώτο επίπεδο ο πειραματισμός επικεντρώνεται στις βασικές ιδιότητες των υλικών, όπως η σκληρότητα (Hardness), η ολισθηρότητα (Friction) και η διαπερατότητα (Density).

Το δεύτερο επίπεδο εστιάζει στην υφή των επιφανειών (Texture) και στο πώς, μέσω ψηλάφησης, ο χρήστης μπορεί να αναγνωρίσει αυλακώσεις, καμπύλες και εξογκώματα.

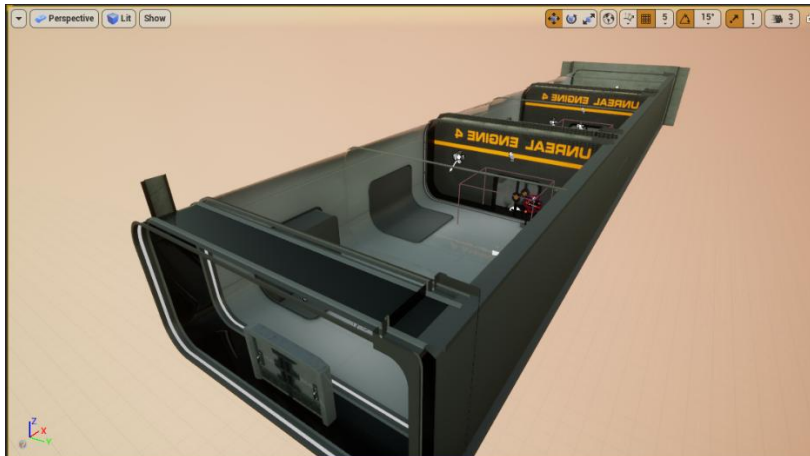
Τέλος, το τρίτο επίπεδο είναι αφιερωμένο στην αναγνώριση αντικειμένων (Shapes) αποκλειστικά μέσω απτικής πληροφορίας (Tactile interaction) [10], [12], [13].

### **3.3 Δημιουργία Τρισδιάστατων Χώρων**

Αφού ολοκληρωθεί η διαδικασία αρχικοποίησης του έργου, το επόμενο βήμα είναι η δημιουργία του αρχικού επιπέδου (Level). Κάθε εικονικό περιβάλλον, ανεξάρτητα από τον χαρακτήρα του ή τα εργαλεία που χρησιμοποιούνται για την κατασκευή του, αποτελείται από έναν τρισδιάστατο χώρο, ο οποίος παρατηρείται μέσω μιας ψηφιακής κάμερας και προβάλλεται στην οθόνη υπό μορφή εικονοστοιχείων.

Για τη δημιουργία του εικονικού χώρου απαιτείται η ύπαρξη ενός αρχείου επιπέδου (Level file). Σύμφωνα με την ορολογία του Unreal Engine, το επίπεδο αποτελεί ένα αρχείο που περιλαμβάνει μια συλλογή αντικειμένων δύο ή τριών διαστάσεων, τοποθετημένων στον χώρο με συγκεκριμένες

συντεταγμένες [20]. Η δημιουργία ενός νέου επιπέδου πραγματοποιείται μέσω του Content Browser, είτε με δεξί κλικ είτε μέσω του κουμπιού Add/Import, επιλέγοντας την αντίστοιχη επιλογή Level.



Εικόνα 3.1: DefaultSpace\_Level

Ο πρώτος χάρτης που δημιουργήθηκε ήταν ο DefaultSpace\_Level, όπως φαίνεται στην Εικόνα 3.1, στον οποίο αναπτύχθηκε το εξωτερικό περιβάλλον των πειραματικών χώρων. Το αρχικό επίπεδο δεν περιείχε αντικείμενα, και για τον λόγο αυτό η διαδικασία ξεκίνησε με την προσθήκη στοιχείων φωτισμού. Συγκεκριμένα, μέσω της λειτουργίας drag-and-drop από το πλευρικό μενού αντικειμένων, προστέθηκαν τρία διαφορετικά αντικείμενα φωτισμού, τα οποία σχετίζονται με την ατμόσφαιρα και τον εξωτερικό φωτισμό, ώστε να είναι ευδιάκριτα τα αντικείμενα που επρόκειτο να τοποθετηθούν στον χώρο.

Το Atmospheric Fog υπολογίζει τη σκέδαση του φωτός (Rayleigh και Mie) μέσα στην ατμόσφαιρα. Επηρεάζει το χρώμα του ουρανού, το φως στον ορίζοντα, καθώς και τον τρόπο με τον οποίο τα μακρινά αντικείμενα φαίνονται πιο ξεθωριασμένα. Με αυτόν τον τρόπο μιμείται τη συμπεριφορά της πραγματικής ατμόσφαιρας της Γης. [37]

Το Sky Atmosphere (ή SkyLight) προσομοιώνει τη σκέδαση του φωτός στην ατμόσφαιρα. Συνδέεται με ένα ή δύο Directional Lights (ήλιο/φεγγάρι) και, ανάλογα με τη θέση και τη γωνία τους, υπολογίζει το χρώμα του ουρανού, την ένταση του φωτός και την ορατότητα των μακρινών αντικειμένων μέσω εναέριας προοπτικής [38].

Το Directional Light προσομοιώνει φως που εκπέμπεται από μια πηγή που βρίσκεται σε άπειρη απόσταση. Αυτό σημαίνει ότι όλες οι σκιές που δημιουργεί είναι παράλληλες, γεγονός που το καθιστά ιδανική επιλογή για την προσομοίωση του ηλιακού φωτός [26], [39].

Ο συνδυασμός των τριών παραπάνω στοιχείων φωτισμού δημιουργεί την αίσθηση ενός εξωτερικού χώρου φωτισμένου από φυσικό ηλιακό φως. Στο επόμενο στάδιο προστέθηκαν σηματοδότες ανατανκλάσεων (Sphere Reflection Capture) καθώς και ένα Lightmass Importance Volume, ώστε ο υπολογισμός του φωτισμού να πραγματοποιείται με μεγαλύτερη ακρίβεια εντός του εσωτερικού χώρου. Τα αντικείμενα αυτά δεν χρησιμοποιήθηκαν σε όλα τα επίπεδα, αλλά προστέθηκαν στον δοκιμαστικό χάρτη με σκοπό τη σαφέστερη απεικόνιση του χώρου και τον ομοιόμορφο φωτισμό του.

Στη συνέχεια πραγματοποιήθηκε η κατασκευή του ενδεικτικού δωματίου που θα φιλοξενούσε τα πειραματικά σενάρια. Τα τρισδιάστατα μοντέλα που χρησιμοποιήθηκαν για την κατασκευή των χώρων προήλθαν από δύο εκπαιδευτικά έργα της Epic Games για την εκμάθηση της μηχανής Unreal, συγκεκριμένα τα *Blueprints* και *Content Examples*, τα οποία διατίθενται μέσω της πλατφόρμας Fab (βλ. Παράρτημα Α). Κατά τη διαδικασία κατασκευής του χώρου δόθηκε ιδιαίτερη έμφαση στη σωστή

συναρμογή των μοντέλων δαπέδου, ώστε να αποφεύγονται κενά που θα επέτρεπαν στον χρήστη να εξέλθει εκτός των ορίων του χώρου. Παράλληλα, ρυθμίστηκε ο εσωτερικός φωτισμός με τρόπο που να διατηρείται η αληθοφάνεια, χωρίς να επιβαρύνεται υπερβολικά η υπολογιστική απόδοση του συστήματος.

Αφού η αρχιτεκτονική του χώρου έφτασε σε ένα ικανοποιητικό επίπεδο, ακολούθησε η δημιουργία των βασικών επιπέδων που απαιτούνται για την ανάπτυξη των λειτουργιών κάθε χάρτη, σύμφωνα με τις περιπτώσεις χρήσης που παρουσιάστηκαν προηγουμένως.

### 3.4 Επίπεδο 1<sup>ο</sup>: Πειράματα βάση ιδιότητας υλικών

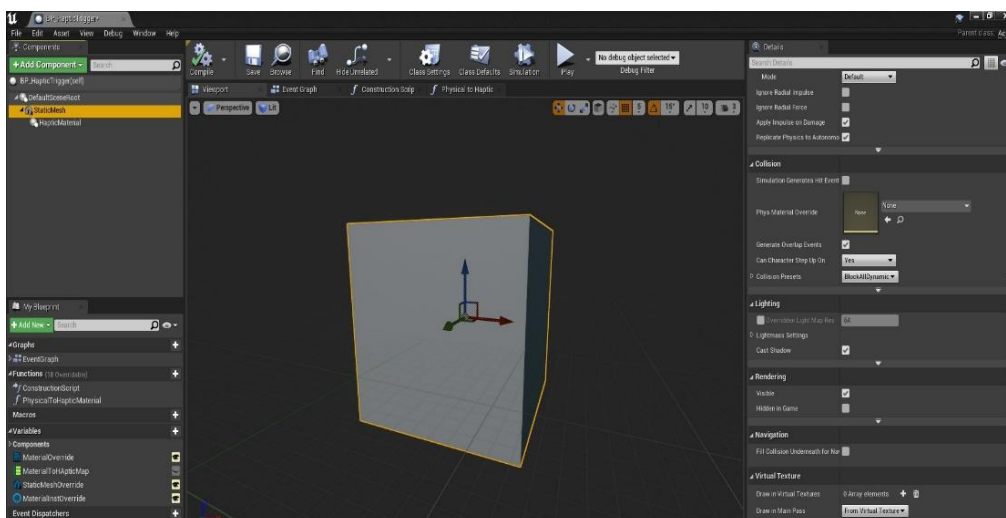
#### 3.4.1 Εισαγωγή

Στο πρώτο επίπεδο του πειραματικού περιβάλλοντος, η έμφαση δίνεται στη διερεύνηση των βασικών φυσικών ιδιοτήτων των υλικών, όπως αυτές γίνονται αντιληπτές μέσω της απτικής ανάδρασης. Στόχος του επιπέδου αυτού είναι να επιτρέψει στον χρήστη να κατανοήσει και να συγκρίνει θεμελιώδη χαρακτηριστικά των επιφανειών μέσα από άμεση απτική αλληλεπίδραση. Στο πλαίσιο αυτό, ο πειραματισμός επικεντρώνεται σε ιδιότητες όπως η σκληρότητα (Hardness), η ολισθηρότητα (Friction) και η διαπερατότητα ή πυκνότητα (Density), οι οποίες επηρεάζουν καθοριστικά την αίσθηση της αφής και τη συμπεριφορά των αντικειμένων κατά την επαφή.

Για την υλοποίηση του παραπάνω πειραματισμού απαιτήθηκε η δημιουργία μιας κατάλληλης τεχνικής υποδομής, η οποία επιτρέπει τη μετάφραση των φυσικών ιδιοτήτων των υλικών σε απτικά ερεθίσματα. Η υποδομή αυτή εξασφαλίζει ότι οι επιλεγμένες ιδιότητες των αντικειμένων μπορούν να αποδοθούν με συνέπεια μέσω του απτικού συστήματος και να γίνουν αντιληπτές από τον χρήστη κατά την αλληλεπίδραση με το εικονικό περιβάλλον.

Για να είναι δυνατή η αλληλεπίδραση του stylus του μηχανικού βραχίονα με τα αντικείμενα του τρισδιάστατου χώρου, τα αντικείμενα αυτά πρέπει να διαθέτουν ως Component το HapticMaterial. Προκειμένου η συγκεκριμένη λειτουργικότητα να είναι επαναχρησιμοποιήσιμη και εύκολα διαχειρίσιμη, αναπτύχθηκε το BP\_HapticTrigger.

#### 3.4.2 BP\_HapticTrigger: Components



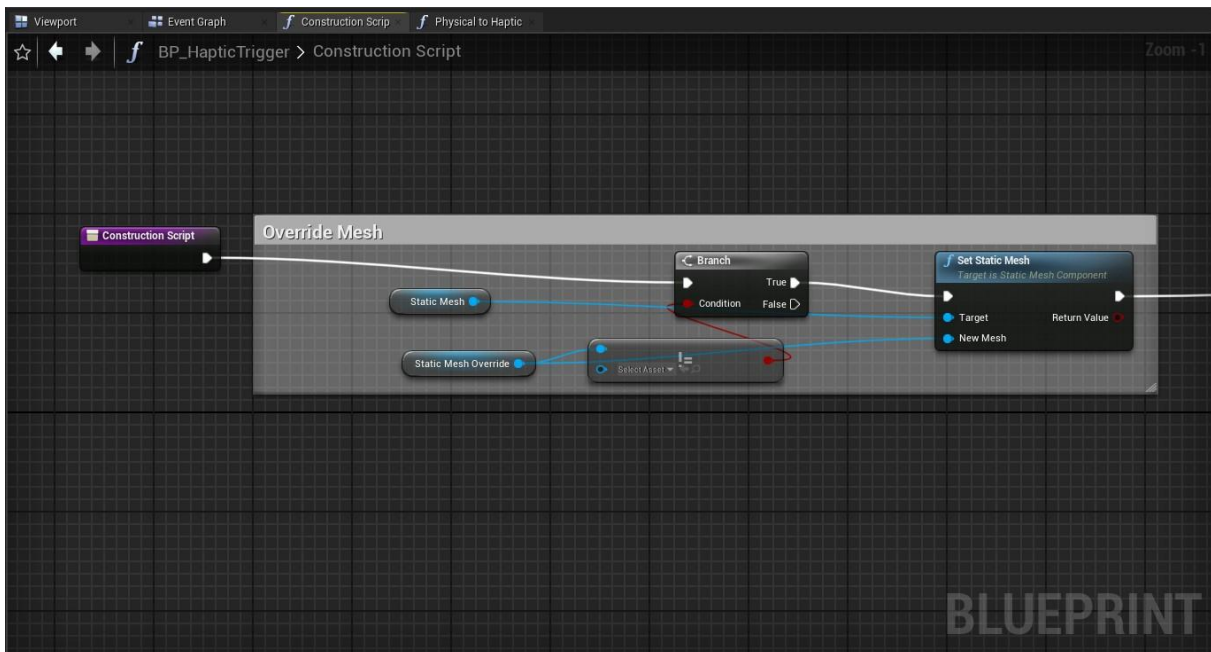
Εικόνα 3.2: BP\_HapticTrigger Viewport

Το συγκεκριμένο Blueprint, όπως φαίνεται στην Εικόνα 3.2, περιλαμβάνει στο DefaultSceneRoot ένα Static Mesh Component, το οποίο καθορίζει τη γεωμετρία του αντικειμένου. Το Static Mesh εμπεριέχει το HapticMaterial Component, επιτρέποντας την επικοινωνία και την απτική αλληλεπίδραση με το stylus του μηχανικού βραχίονα.

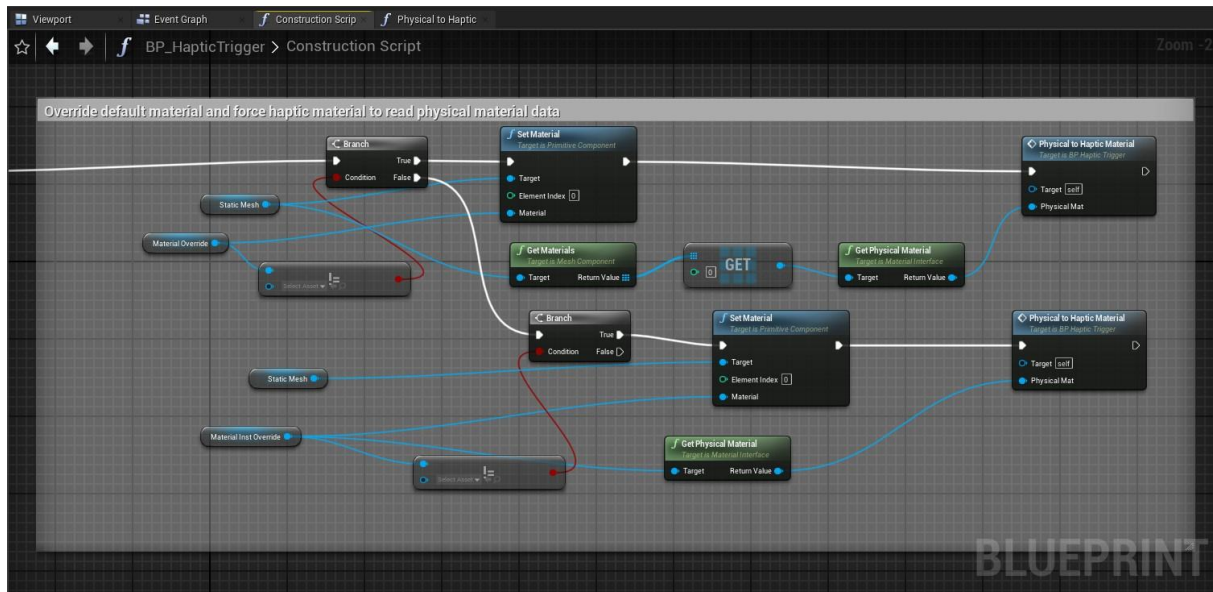
Για να είναι δυνατή η απτική αλληλεπίδραση, το Static Mesh θα πρέπει να έχει οριστεί ως Movable, ώστε να υποστηρίζεται η προσομοίωση φυσικής (Simulate Physics). Παράλληλα, τα Collision Presets πρέπει να έχουν οριστεί στην επιλογή BlockAllDynamic, προκειμένου να εξασφαλίζεται η σωστή ανίχνευση συγκρούσεων με δυναμικά αντικείμενα όπως το stylus. Επιπλέον, το αντικείμενο που χρησιμοποιείται ως Static Mesh θα πρέπει να διαθέτει Simple Collision, καθώς η μηχανή Unreal δεν υποστηρίζει την εφαρμογή φυσικής σε αντικείμενα με Complex Collision. Η συγκεκριμένη ρύθμιση είναι απαραίτητη για την ορθή λειτουργία της προσομοίωσης και τη σταθερή απόδοση της απτικής ανάδρασης [40].

### 3.4.3 BP\_HapticTrigger: Construction Script

Αφού ολοκληρώθηκε η οπτική αναπαράσταση των στοιχείων του Blueprint, ακολούθησε η ανάπτυξη της λειτουργικότητάς του, όπως παρουσιάζεται στην Εικόνα 3.3α και στην Εικόνα 3.3β. Αρχικά υλοποιήθηκε το Construction Script, στο οποίο δημιουργήθηκε μια ροή που επιτρέπει στον χρήστη, κάθε φορά που εισάγει ένα instance του Blueprint στο επίπεδο, να ορίζει δυναμικά το επιθυμητό Static Mesh και Material ή Material Instance.



Εικόνα 3.3α: BP\_HapticTrigger Construction Script



Εικόνα 3.3β: BP\_HapticTrigger Construction Script

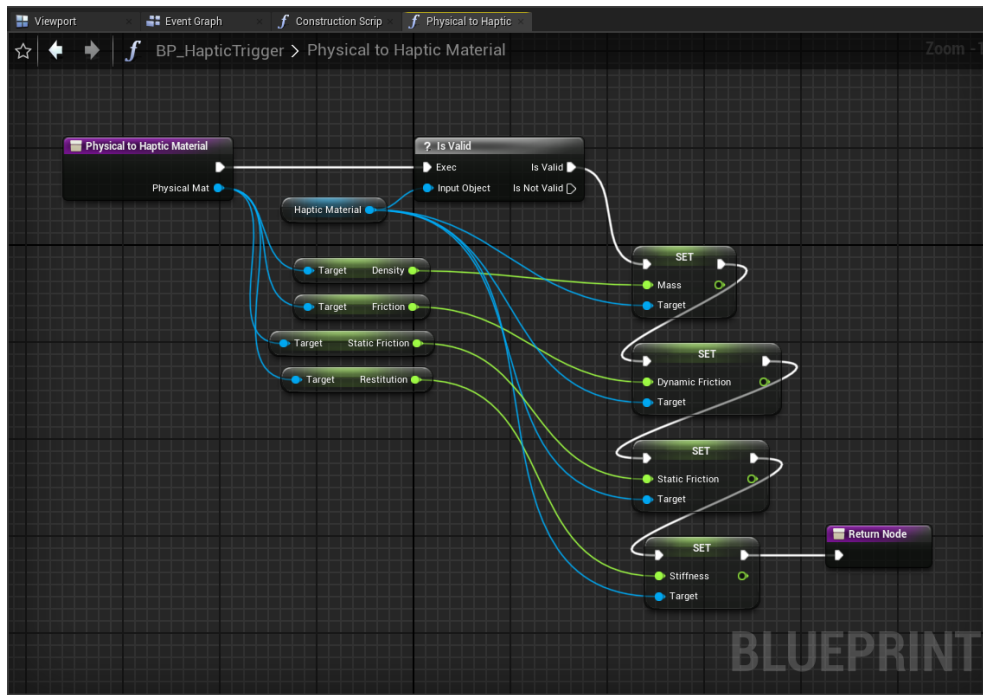
Το Construction Script συνδέεται με έναν κόμβο Branch, ο οποίος ελέγχει αν έχει δοθεί τιμή στη μεταβλητή Static Mesh Override [41]. Στην περίπτωση που η συνθήκη είναι αληθής, καλείται η μέθοδος Set Static Mesh του Static Mesh Component, στην οποία αποδίδεται ως όρισμα το αντικείμενο που έχει οριστεί στη μεταβλητή *Static Mesh Override*.

Η ροή συνεχίζεται με αντίστοιχο τρόπο για τον ορισμό του Material ή του Material Instance. Ένα Static Mesh μπορεί να διαθέτει Material ή Material Instance σε μια θέση Material Element. Στην περίπτωση που δοθούν και τα δύο ως ορίσματα, το τελικό αποτέλεσμα εμφανίζει πάντοτε το Material. Στο συγκεκριμένο Blueprint, το Static Mesh Component περιλαμβάνει μόνο ένα Material Element. Συνεπώς, δεν έχει σημασία αν το αντικείμενο που θα δοθεί ως όρισμα διαθέτει περισσότερα από ένα Material Elements, καθώς ο ορισμός του υλικού καθορίζεται αποκλειστικά από τη ροή του Construction Script. Αν δεν έχει οριστεί κανένα από τα δύο, το αντικείμενο διατηρεί το προκαθορισμένο (default) Material που του έχει ανατεθεί.

Τέλος, από το Material ή το Material Instance αντλείται η ιδιότητα Physics Material, η οποία περιέχει πληροφορίες σχετικά με τις φυσικές ιδιότητες του υλικού. Η ιδιότητα αυτή χρησιμοποιείται ως είσοδος στη συνάρτηση Physical to Haptic Material, της οποίας η λειτουργία αναλύεται στην επόμενη ενότητα.

### 3.4.4 BP\_HapticTrigger: Physical to Haptic Material Function

Για τη διασύνδεση των φυσικών ιδιοτήτων των υλικών με τα απτικά συστήματα του μηχανικού βραχίονα, αναπτύχθηκε μια μέθοδος η οποία λαμβάνει ως όρισμα ένα Physics Material, εξάγει από αυτό τις απαραίτητες παραμέτρους και τις αποδίδει στα αντίστοιχα χαρακτηριστικά του Haptic Material Component, όπως φαίνεται στην Εικόνα 3.4.



Εικόνα 3.4: Physical to Haptic Material Function

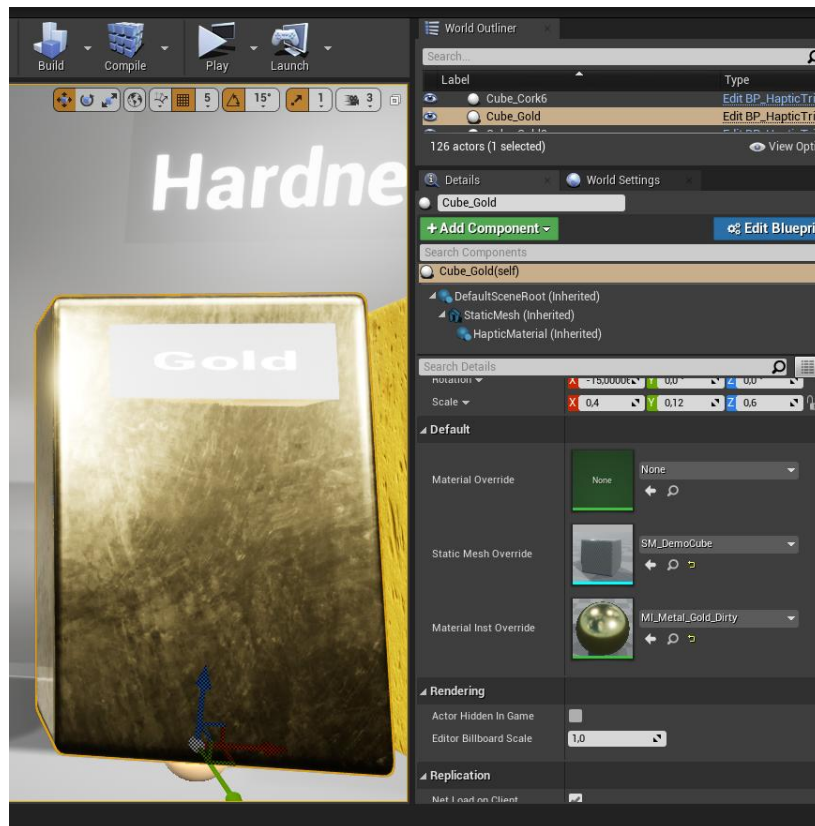
Με την κλήση της μεθόδου, αντλούνται από το Physics Material οι τιμές των πεδίων πυκνότητας–μάζας (Density), δυναμικής τριβής (Friction), στατικής τριβής (Static Friction) και ελαστικότητας (Restitution) [42], [43].

- Πυκνότητα (Density): Χρησιμοποιείται σε συνδυασμό με το σχήμα του αντικείμενου για τον υπολογισμό των ιδιοτήτων μάζας του. Όσο μεγαλύτερη είναι η τιμή της, τόσο βαρύτερο θεωρείται το αντικείμενο.
- Τριβή (Friction): Αποτελεί την τιμή της τριβής της επιφάνειας και καθορίζει το πόσο εύκολα μπορούν τα αντικείμενα να ολισθαίνουν επάνω σε αυτή.
- Στατική Τριβή (Static Friction): Αναφέρεται στην αντίσταση που ασκείται στην έναρξη της κίνησης μεταξύ δύο επιφανειών που βρίσκονται αρχικά σε κατάσταση ηρεμίας. Καθορίζει το πόση δύναμη απαιτείται ώστε ένα αντικείμενο να αρχίσει να ολισθαίνει από την αρχική του θέση.
- Ελαστικότητα (Restitution): Καθορίζει το πόσο «ελαστική» είναι μια επιφάνεια ή πόση ενέργεια διατηρεί κατά τη σύγκρουσή της με μια άλλη επιφάνεια.

Οι παραπάνω τιμές αντιστοιχίζονται και αποδίδονται στις ιδιότητες Mass, Dynamic Friction, Static Friction και Stiffness του Haptic Material Component. Οι ρυθμίσεις αυτές μεταφέρονται στο απτικό

σύστημα και επιστρέφονται στον μηχανικό βραχίονα κάθε φορά που το stylus έρχεται σε επαφή με την επιφάνεια του αντικειμένου, επιτυγχάνοντας την παραγωγή απτικής ανάδρασης.

Το τελικό Blueprint τοποθετείται στο χώρο και μπορεί να πάρει ορίσματα για Static Mesh και Material ή Material Instance, όπως φαίνεται στην Εικόνα 3.5.



Εικόνα 3.5: BP\_HapticTrigger Instance μέσα στο επίπεδο

Πίσω από τα απτικά αντικείμενα του κάθε πειράματος τοποθετήθηκε ένα αντικείμενο το οποίο δεν δέχεται απτικά ερεθίσματα και λειτουργεί αποτρεπτικά σε κάθε μορφή σύγκρουσης. Τα αντικείμενα αυτά χρησιμοποιούνται για την οριοθέτηση της ελάχιστης απόστασης στην οποία μπορεί να προσεγγίσει ο παίκτης το απτικό αντικείμενο, με σκοπό τη διασφάλιση της βέλτιστης εμπειρίας αλληλεπίδρασης και της αποτελεσματικής απτικής ανάδρασης.

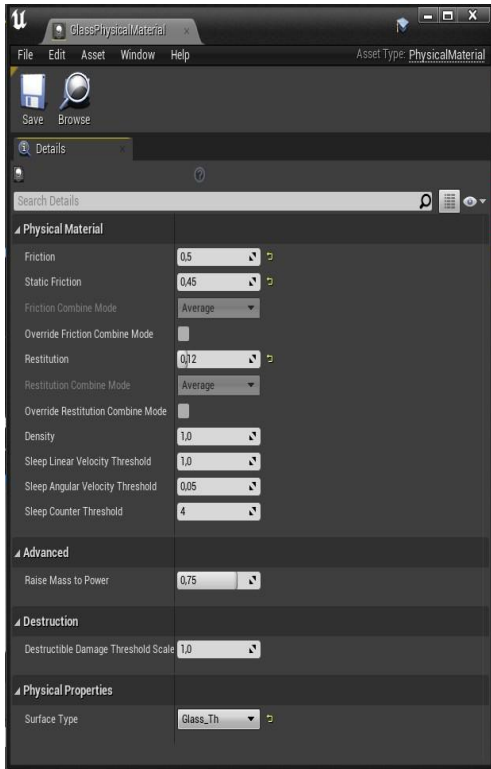
### 3.4.5 Physical Material

Τα Physical Materials (PM) στο Unreal Engine αποτελούν ειδικά δεδομένα που χρησιμοποιούνται για τον καθορισμό της φυσικής συμπεριφοράς των αντικειμένων κατά την αλληλεπίδρασή τους με το περιβάλλον. Μέσω αυτών ορίζονται βασικές φυσικές ιδιότητες, όπως η τριβή, η στατική τριβή, η ελαστικότητα και η πυκνότητα, οι οποίες επηρεάζουν τον τρόπο με τον οποίο τα αντικείμενα κινούνται, συγκρούονται και αντιδρούν σε δυνάμεις [42], [43].

Η χρήση των Physical Materials επιτρέπει τη ρεαλιστική προσομοίωση διαφορετικών τύπων επιφανειών και συμβάλλει στη βελτίωση της συνολικής φυσικής συμπεριφοράς του εικονικού περιβάλλοντος [42], [43]. Στο πλαίσιο εφαρμογών με απτική ανάδραση, τα Physical Materials μπορούν επιπλέον να λειτουργήσουν ως γέφυρα μεταξύ των φυσικών ιδιοτήτων των αντικειμένων και των απτικών παραμέτρων που μεταφέρονται σε εξωτερικά συστήματα, όπως ο μηχανικός βραχίονας 3D Systems Touch, ενισχύοντας τον ρεαλισμό της εμπειρίας χρήστη.

### 3.4.6 Δημιουργία

Η δημιουργία ενός νέου Physical Material πραγματοποιείται μέσω του Content Browser, είτε με δεξί κλικ είτε μέσω του κουμπιού Add/Import, επιλέγοντας την κατηγορία Physics και στη συνέχεια την επιλογή Physical Material. Η ενέργεια αυτή οδηγεί στο άνοιγμα ενός παραθύρου διαλόγου, στο οποίο ορίζεται η κλάση του αντικειμένου, η οποία στην προκειμένη περίπτωση αντιστοιχεί επίσης σε Physical Material [44]. Στην Εικόνα 3.6 παρουσιάζεται το περιβάλλον επεξεργασίας των Physical Material.

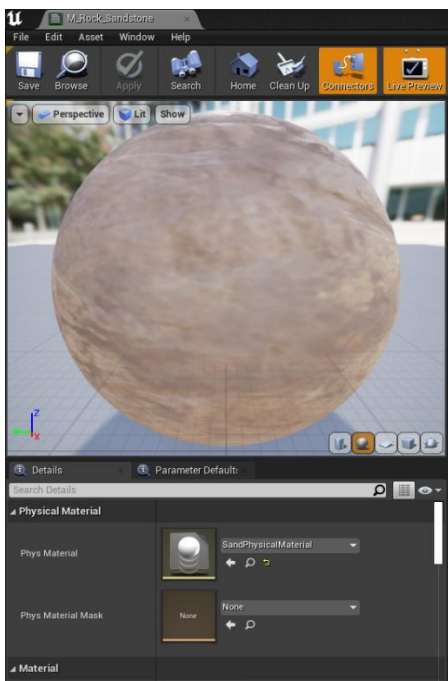


Στο πλαίσιο της παρούσας εργασίας δημιουργήθηκαν συνολικά δεκαοκτώ Physical Materials, τα οποία κατηγοριοποιήθηκαν βάσει της κυρίαρχης φυσικής ιδιότητας που ήταν επιθυμητό να αναδειχθεί. Συγκεκριμένα, δημιουργήθηκαν έξι Physical Materials με έμφαση στην ελαστικότητα, έξι με έμφαση στην τριβή, τέσσερα με έμφαση στην πυκνότητα, καθώς και ένα Physical Material γενικής χρήσης. Το τελευταίο χρησιμοποιήθηκε σε αντικείμενα για τα οποία απαιτείται απτική ανάδραση, χωρίς να δίνεται έμφαση σε κάποια συγκεκριμένη φυσική ιδιότητα του υλικού.

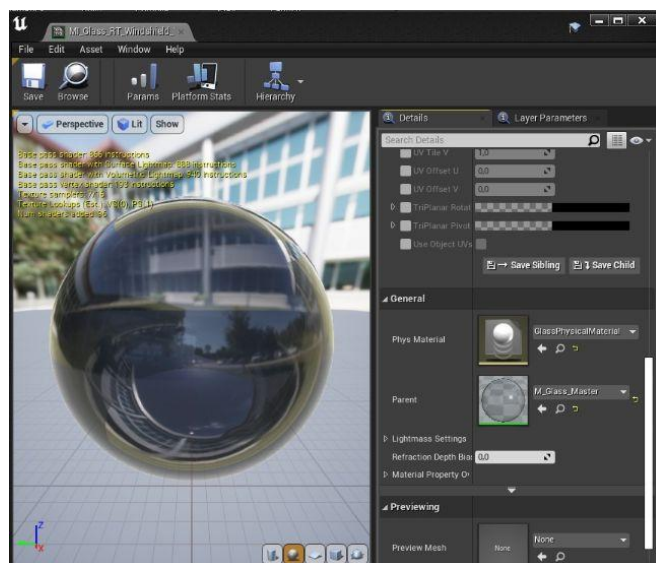
### 3.4.7 Ανάθεση Physical Material

Για την ανάθεση ενός Physical Material σε Material ή Material Instance, αρκεί να ανοιχτεί το αντίστοιχο Material ή Material Instance στον Editor και να οριστεί το επιθυμητό Physical Material στην ιδιότητα Physical Material του παραθύρου Details, όπως φαίνεται στην Εικόνα 3.7 και στην Εικόνα 3.8 αντίστοιχα [45], [46].

Εικόνα 3.6: Physical Material Editor



Εικόνα 3.7: Material



Εικόνα 3.8: Material Instance

Τα Materials που χρησιμοποιήθηκαν για τον εμπλουτισμό των εικονικών χώρων προήλθαν από τρία διαφορετικά έργα, σχεδιασμένα για χρήση στη μηχανή Unreal Engine. Συγκεκριμένα, αξιοποιήθηκαν τα πακέτα *300+ Ultimate PBR Materials Pack*, *99 PBR Materials Vol.01* και *Automotive Materials*, τα οποία διατίθενται μέσω της πλατφόρμας Fab (βλ. Παράρτημα Α).

### 3.4.8 Ανάλυση των υλικών που χρησιμοποιήθηκαν

Οι πραγματικές τιμές των ιδιοτήτων των υλικών δεν αποδίδουν πάντα την επιθυμητή ανάδραση μέσα στο εικονικό περιβάλλον. Για το λόγο αυτό, έπειτα από δοκιμές και αξιολόγηση, πραγματοποιήθηκαν οι απαραίτητες προσαρμογές, ώστε η αλληλεπίδραση του χρήστη με τα πειράματα να παράγει την επιθυμητή ανάδραση και να καθιστά σαφείς τις διαφορές μεταξύ των διαφόρων υλικών σε κάθε πείραμα. Στους παρακάτω πίνακες υπάρχουν οι κανονικοποιημένες τιμές των ιδιοτήτων των υλικών στο διάστημα 0-1 και οι τελικές τιμές που τους αποδόθηκαν μέσα στο Unreal.

Πίνακας 3.1: Πραγματικές τιμές των υλικών του πειράματος Σκληρότητας [62], [63]

Υλικό/Ιδιότητα	Σκληρότητα	Δυναμική Τριβή	Στατική Τριβή	Μάζα
Χρυσός	1.00	0.29	0.41	1.00
Σφουγγάρι	0.00	0.70	0.66	0.00
Στεγνός Πηλός	0.68	0.49	0.50	0.08
Φελλός	0.39	0.60	0.58	0.01
Ύφασμα	0.63	0.49	0.50	0.03
Λάστιχο	0.46	1.00	1.00	0.06

Πίνακας 3.2: Αντίστοιχες τιμές που δόθηκαν μέσα στο Unreal

Υλικό/Ιδιότητα	Σκληρότητα	Δυναμική Τριβή	Στατική Τριβή	Μάζα
Χρυσός (Gold)	1.00	0.50	0.60	1.00
Σφουγγάρι (Sponge)	0.15	0.50	0.60	0.50
Στεγνός Πηλός (GClay)	0.40	0.50	0.60	0.40
Φελλός (Cork)	0.25	0.50	0.60	0.25
Ύφασμα (Denim)	0.40	0.50	0.60	0.50
Λάστιχο (Rubber)	0.30	0.50	0.60	0.60



Εικόνα 3.9: Πειράματα Σκληρότητας

Πίνακας 3.3: Πραγματικές τιμές των υλικών του πειράματος Ολισθηρότητας [62], [63]

Υλικό/Ιδιότητα	Σκληρότητα	Δυναμική Τριβή	Στατική Τριβή	Μάζα
Άσφαλτος	0.80	0.8	0.75	0.12
Γυαλί	0.99	0.29	0.33	0.13
Τσιμέντο	0.93	0.70	0.66	0.12
Πάγος	0.84	0.02	0.03	0.05
Πέτρα	0.98	0.60	0.58	0.14
Μάρμαρο	0.98	0.39	0.41	0.14

Πίνακας 3.4: Αντίστοιχες τιμές που δόθηκαν μέσα στο Unreal

Υλικό/Ιδιότητα	Σκληρότητα	Δυναμική Τριβή	Στατική Τριβή	Μάζα
Άσφαλτος (Asphalt)	1.00	1.00	1.00	0.50
Γυαλί (Glass)	0.12	0.50	0.45	1.00
Τσιμέντο (Concrete)	0.3	0.65	0.75	0.5
Πάγος (Ice)	0.095	0.00	0.00	1.00
Πέτρα (Rock)	0.70	0.70	0.80	0.50
Μάρμαρο (Marble)	0.20	0.80	0.80	1.00



Εικόνα 3.10: Πειράματα Ολισθηρότητας

Πίνακας 3.5: Πραγματικές τιμές των υλικών του πειράματος Διαπερατότητας [62], [63]

Υλικό/Ιδιότητα	Σκληρότητα	Δυναμική Τριβή	Στατική Τριβή	Μάζα
Αέρας	0.00	0.00	0.00	0.00
Άμμος	0.46	0.49	0.50	0.08
Λάσπη	0.17	0.29	0.33	0.09
Υγρός Πηλός	0.51	0.39	0.41	0.10
Βράχος	0.97	0.60	0.58	0.13

Πίνακας 3.6: Αντίστοιχες τιμές που δόθηκαν μέσα στο Unreal

Υλικό/Ιδιότητα	Σκληρότητα	Δυναμική Τριβή	Στατική Τριβή	Μάζα
Αέρας (Air)	0.00	0.10	0.10	0.00
Άμμος (Sand)	0.02	0.50	0.60	0.50
Λάσπη (Mud)	0.10	0.50	0.60	0.50
Υγρός Πηλός (RClay)	0.18	0.50	0.60	0.50
Βράχος	0.85	0.70	0.70	0.60

Για τον βράχο χρησιμοποιήθηκε το Texture Physical Material γενικής χρήσης.



Εικόνα 3.11: Πειράματα Διαπερατότητας

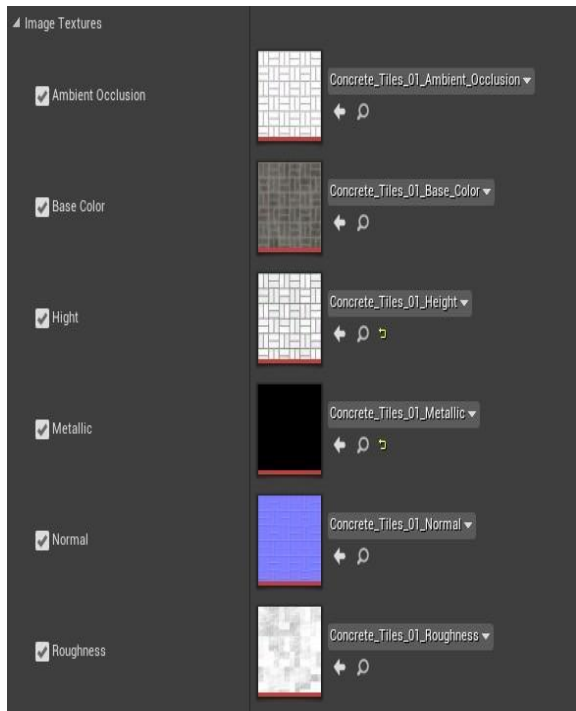
### 3.5 Επίπεδο 2<sup>ο</sup>: Πειράματα βάση υφής επιφανειών (Texture)

#### 3.5.1 Εισαγωγή

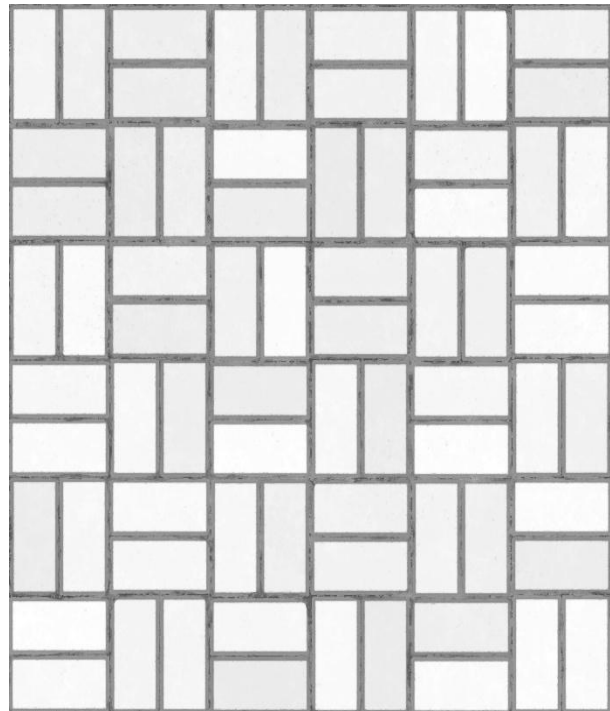
Στο δεύτερο επίπεδο του πειραματικού περιβάλλοντος, η προσοχή μετατοπίζεται από τις βασικές φυσικές ιδιότητες των υλικών στη λεπτομερή απτική αντίληψη της επιφάνειας των αντικειμένων. Το επίπεδο αυτό έχει ως στόχο να διερευνήσει τον τρόπο με τον οποίο η υφή (Texture) επηρεάζει την απτική εμπειρία και πώς, μέσω της ψηλάφησης, ο χρήστης μπορεί να αναγνωρίσει μορφολογικά χαρακτηριστικά όπως αυλακώσεις, καμπύλες και εξογκώματα, αποκλειστικά με βάση την απτική πληροφορία [47], [48].

#### 3.5.2 Δημιουργία Ειδικών Επιφανειών

Δημιουργήθηκαν ειδικά τρισδιάστατα μοντέλα, των οποίων η επιφάνεια ενσωματώνει τα παραπάνω χαρακτηριστικά υφής. Για τον σκοπό αυτό επιλέχθηκαν συγκεκριμένα Materials, τα οποία κρίθηκε ότι μπορούν να προσφέρουν μια διαφοροποιημένη και ενδιαφέρουσα απτική εμπειρία. Από τα Materials αυτά εξήχθη η πληροφορία ύψους (Height Map), όπως φαίνεται στην Εικόνα 3.12 και στην Εικόνα 3.13, βάσει της οποίας δημιουργήθηκε η τελική γεωμετρία των μοντέλων με τη χρήση της μηχανής τρισδιάστατης μοντελοποίησης Blender 2.79.



Εικόνα 3.12: Πληροφορίες υφής του Material



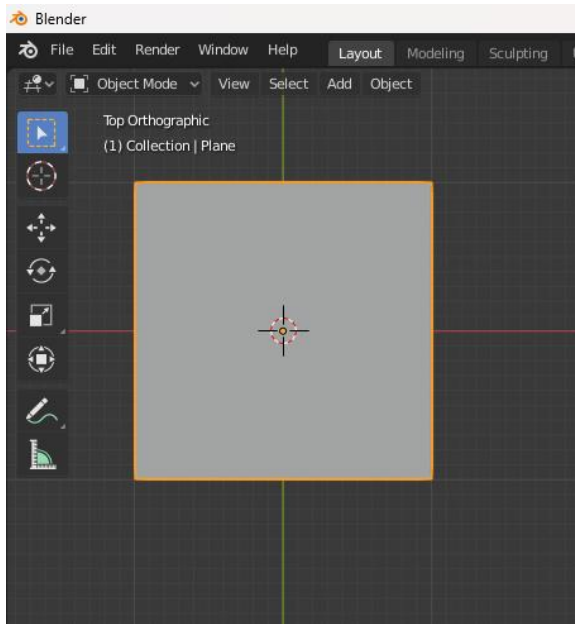
Εικόνα 3.13: Hight Map

### 3.5.3 Διαδικασία που ακολουθήθηκε στο Blender 2.79

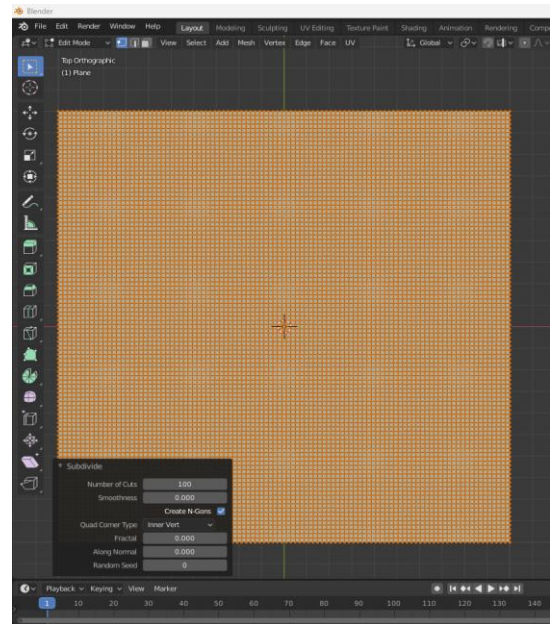
#### 3.5.3.1 Μοντελοποίηση

Στο περιβάλλον του Blender χρησιμοποιήθηκαν δύο βασικές λειτουργικές καταστάσεις (Modes): το Object Mode, όπως φαίνεται στην Εικόνα 3.14, το οποίο περιλαμβάνει ενέργειες που επηρεάζουν το αντικείμενο ως ενιαίο σύνολο, και το Edit Mode, όπως φαίνεται στην Εικόνα 3.15, στο οποίο πραγματοποιείται η επεξεργασία και η παραμόρφωση της γεωμετρίας του αντικειμένου [49].

Σε Object Mode προστέθηκε ένα επίπεδο αντικείμενο (Plane) στη σκηνή του τρισδιάστατου χώρου. Στη συνέχεια, σε Edit Mode, πραγματοποιήθηκε υποδιαίρεση (Subdivide) της επιφάνειας του αντικειμένου, με την παράμετρο *Number of Cuts* να ορίζεται στην τιμή 100, προκειμένου να επιτευχθεί επαρκής γεωμετρική πυκνότητα.



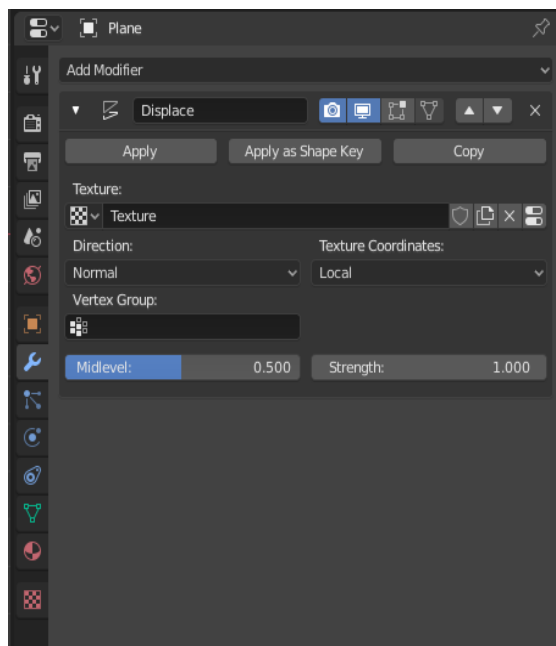
Εικόνα 3.14: Object Mode



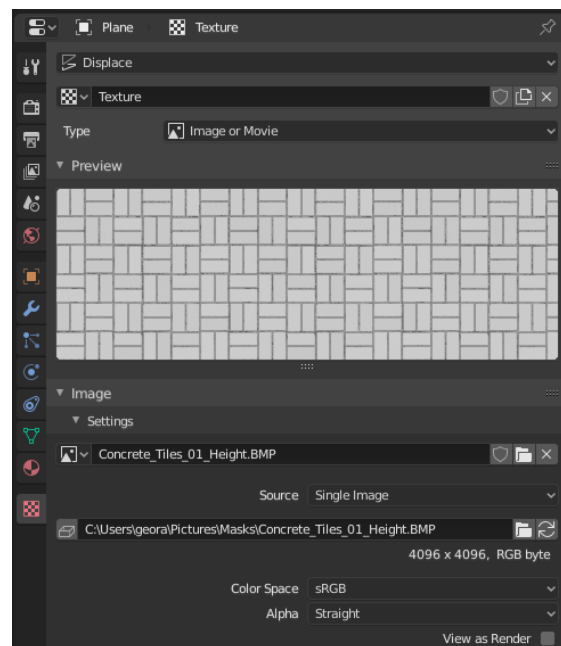
Εικόνα 3.15: Edit Mode – Subdivide

Ακολουθως, επιστρέφοντας σε Object Mode, από την πλευρική καρτέλα εργαλείων προστέθηκε ένας Displace Modifier, όπως φαίνεται στην Εικόνα 3.16, ο οποίος δέχεται ως είσοδο ένα Texture και παραμορφώνει τη γεωμετρία του αντικειμένου βάσει της πληροφορίας που αυτό περιέχει [47], [48].

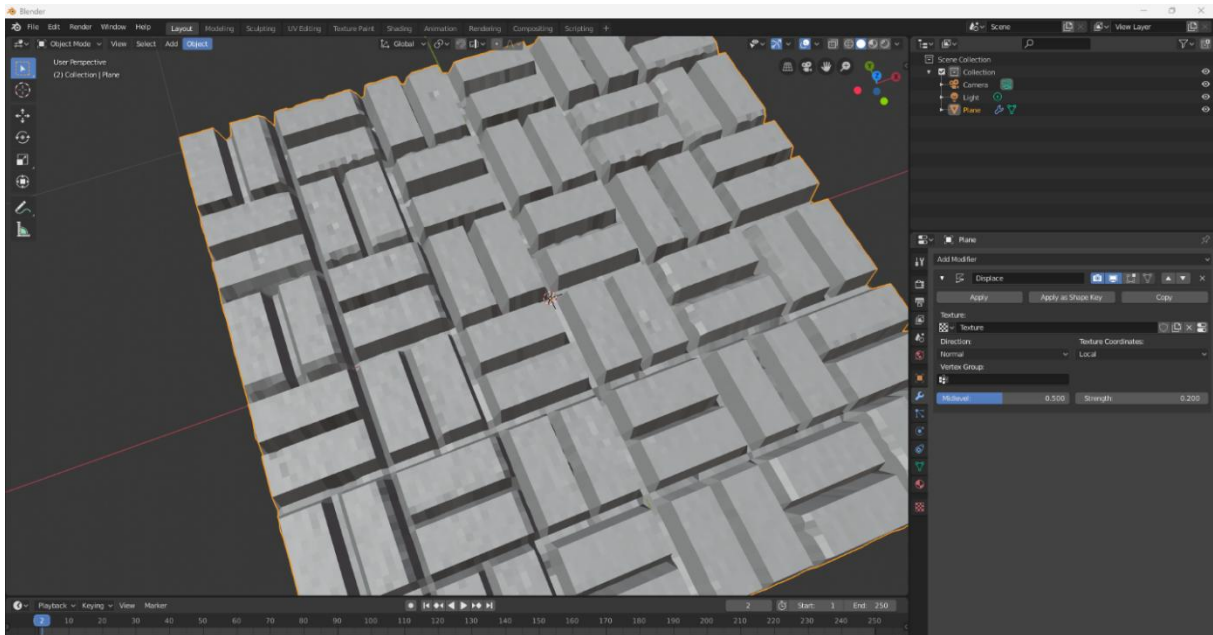
Στην καρτέλα Texture δημιουργήθηκε ένα νέο Texture τύπου εικόνας, στο οποίο ορίστηκε το αντίστοιχο Height Map, όπως φαίνεται στην Εικόνα 3.17. Με την εφαρμογή του Displace Modifier, το Plane παραμορφώθηκε κατάλληλα, ώστε να μεταφέρει την πληροφορία ύψους της διδιάστατης εικόνας Height Map σε τρισδιάστατη γεωμετρία, επιτρέποντας την ακριβή αναπαράσταση της επιφανειακής υφής, όπως φαίνεται στην Εικόνα 3.18.



Εικόνα 3.16: Displace Modifier

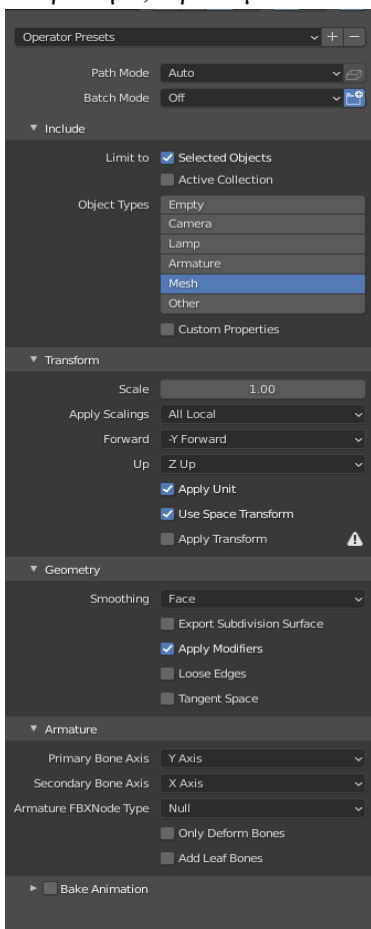


Εικόνα 3.17: Texture



Εικόνα 3.18: Εφαρμογή του Displace Modifier

Στη συνέχεια, ρυθμίστηκε η παράμετρος Strength του Displace Modifier ώστε να επιτευχθεί το επιθυμητό αποτέλεσμα, και ο τροποποιητής εφαρμόστηκε (Apply) [47], [48]. Ακολούθως, σε Edit Mode, πραγματοποιήθηκαν επιμέρους διορθώσεις στη γεωμετρία του αντικειμένου, όπου κρίθηκε απαραίτητο, προκειμένου να αποφευχθούν περιοχές με υπερβολική παραμόρφωση. Πριν από τη διαδικασία εξαγωγής (export) του αντικειμένου, εφαρμόστηκαν όλες οι μετασχηματίσεις (Apply All Transforms), ώστε να οριστικοποιηθούν οι αλλαγές που πραγματοποιήθηκαν στη γεωμετρία.



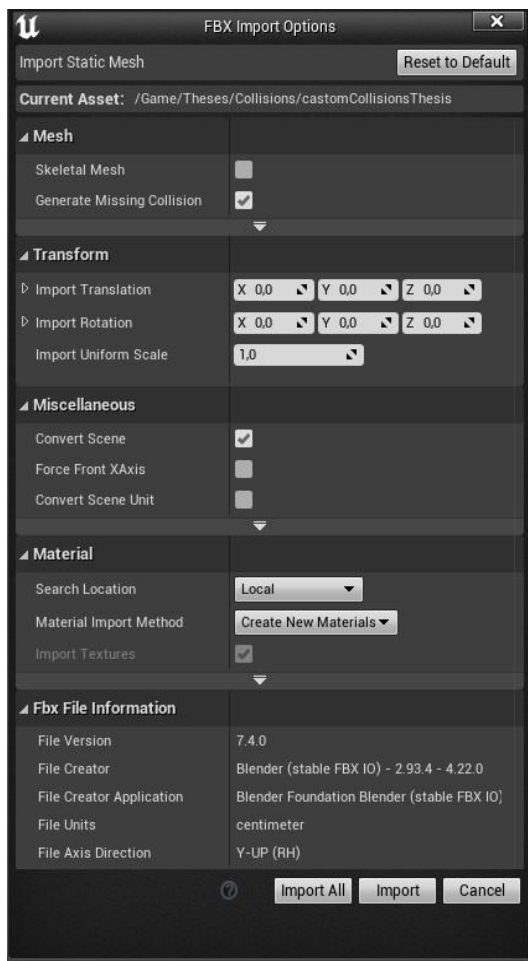
Εικόνα 3.19: Export FBX

### 3.5.3.2 Εξαγωγή (export) FBX από το Blender

Για να είναι δυνατή η εισαγωγή (import) των αρχείων στο Unreal Engine, τα αρχεία πρέπει να έχουν κατάληξη FBX. Στο Blender, η διαδικασία εξαγωγής πραγματοποιείται μέσω του μενού File, επιλέγοντας Export και στη συνέχεια FBX. Στην έκδοση Blender 2.79, η ενέργεια αυτή ανοίγει ένα αναδυόμενο παράθυρο, στο οποίο ορίζονται η τοποθεσία αποθήκευσης, το όνομα του αρχείου, καθώς και οι σχετικές ρυθμίσεις εξαγωγής, όπως φαίνεται στην Εικόνα 3.19 [50], [51]. Για την εξαγωγή των επιθυμητών αντικειμένων, αφού αυτά επιλεγούν εντός της σκηνής, ενεργοποιείται η επιλογή **Selected Objects**, ώστε να εξαχθούν μόνο τα επιλεγμένα στοιχεία. Στη συνέχεια ορίζεται ο τύπος των αντικειμένων προς εξαγωγή, ο οποίος στην παρούσα περίπτωση είναι Mesh. Στη συνέχεια, οι μετασχηματισμοί (Transforms) ρυθμίζονται κατάλληλα ώστε να είναι συμβατοί με τις απαιτήσεις του Unreal Engine. Συγκεκριμένα, η τιμή του Scale ορίζεται στο 1.00, ενώ οι άξονες Forward και Up ρυθμίζονται σε -Y Forward και Z Up, αντίστοιχα. Επιπλέον, ενεργοποιούνται οι επιλογές Apply Unit και Use Space Transform,

ώστε οι μονάδες μέτρησης του Blender να αντιστοιχίζονται σωστά σε εκατοστά στο Unreal Engine και να διασφαλίζεται ο ορθός προσανατολισμός των αξόνων κατά την εισαγωγή των αντικειμένων. Για τη γεωμετρία των αντικειμένων επιλέγεται η ρύθμιση Face Smoothing, ώστε να αποφεύγονται ανεπιθύμητες παραμορφώσεις κατά τη διαδικασία εξαγωγής. Παράλληλα, ενεργοποιείται η επιλογή Apply Modifiers, προκειμένου τυχόν τροποποιητές (Modifiers) που δεν έχουν εφαρμοστεί να ενσωματωθούν στη γεωμετρία πριν την εξαγωγή των αρχείων. Τέλος, οι ρυθμίσεις που αφορούν το Armature και το Bake Animation απενεργοποιούνται, καθώς τα αντικείμενα που πρόκειται να εξαχθούν δεν περιλαμβάνουν σκελετική δομή ή κινούμενα δεδομένα [50], [51].

Η παραπάνω διαδικασία επαναλήφθηκε για τη δημιουργία όλων των διαφορετικών επιφανειών υφής που χρησιμοποιήθηκαν στο συγκεκριμένο επίπεδο (level).



Εικόνα 3.20 : Import FBX

### 3.5.4 Εισαγωγή (Import) FBX στο Unreal Engine

Η εισαγωγή ενός αρχείου FBX πραγματοποιείται μέσω του Content Browser, είτε με δεξί κλικ και επιλογή Import To, είτε μέσω του κουμπιού Add/Import. Η ενέργεια αυτή ανοίγει ένα αναδυόμενο παράθυρο, όπως φαίνεται στην Εικόνα 3.20, στο οποίο γίνεται η αναζήτηση και επιλογή των αρχείων που πρόκειται να εισαχθούν στο έργο. Με την ολοκλήρωση της επιλογής, εμφανίζεται ένα επιπλέον παράθυρο διαλόγου, στο οποίο ορίζονται οι απαραίτητες ρυθμίσεις εισαγωγής [52], [53].

Αφού ολοκληρωθούν οι σχετικές ρυθμίσεις, τα αρχεία φορτώνονται στο έργο και καθίστανται διαθέσιμα μέσω του Content Browser, ώστε να μπορούν να χρησιμοποιηθούν εντός του εικονικού περιβάλλοντος.

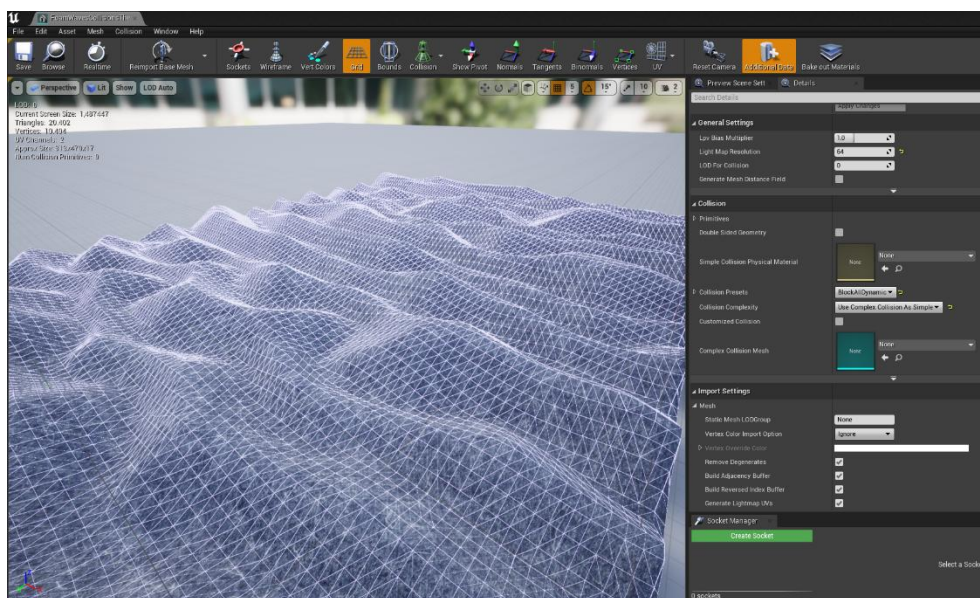
### 3.5.5 Παραμετροποίηση ειδικών επιφανιών

Όλα τα παραπάνω προσαρμοσμένα (custom) αντικείμενα αναγνωρίζονται από το Unreal Engine ως αντικείμενα τύπου Static Mesh και απαιτούν κατάλληλη παραμετροποίηση, ώστε να είναι ανιχνεύσιμα από το stylus του απτικού συστήματος. Σε αντίθεση με τα απλά γεωμετρικά σχήματα που χρησιμοποιήθηκαν στο προηγούμενο επίπεδο, τα συγκεκριμένα αντικείμενα προορίζονται για χρήση ως προσαρμοσμένα collision meshes και χαρακτηρίζονται από ιδιαίτερα πολύπλοκη γεωμετρία.

Ο τρόπος απόδοσης γεωμετρίας σύγκρουσης (collision) σε ένα αντικείμενο διακρίνεται σε δύο βασικές κατηγορίες: τα Simple Collision, τα οποία χρησιμοποιούν απλοποιημένα γεωμετρικά σχήματα που

περικλείουν το αντικείμενο σε μικρή απόσταση από την πραγματική του γεωμετρία, και τα Complex Collision, τα οποία αναπαράγουν με ακρίβεια τη γεωμετρία του αντικειμένου. Το ζήτημα προκύπτει όταν απαιτείται η εφαρμογή φυσικής προσομοίωσης στο αντικείμενο, καθώς στο Unreal Engine η προσομοίωση φυσικής (Simulate Physics) υποστηρίζεται αποκλειστικά από αντικείμενα που διαθέτουν Simple Collision.

Για τον λόγο αυτό, πραγματοποιήθηκαν οι απαραίτητες ρυθμίσεις στις ιδιότητες των εισαγόμενων αντικειμένων, όπως φαίνεται στην Εικόνα 3.21 με στόχο την υποστήριξη της φυσικής προσομοίωσης και της απτικής αλληλεπίδρασης.

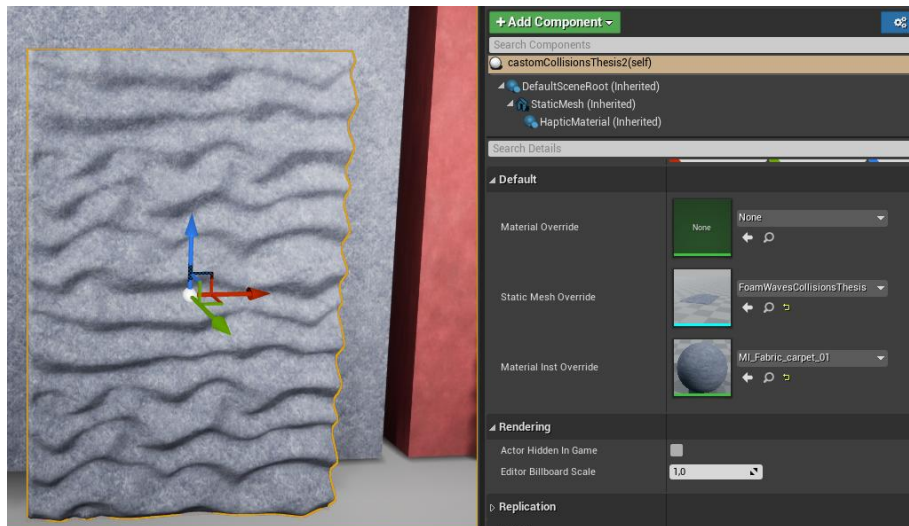


Εικόνα 3.21: Custom Collision Mesh

Στις ιδιότητες Collision κάθε αντικείμενου πραγματοποιήθηκαν τροποποιήσεις στα πεδία Collision Preset και Collision Complexity. Η ρύθμιση του Collision Preset στην επιλογή BlockAllDynamic διασφαλίζει ότι το αντικείμενο ανιχνεύει και αποκρίνεται σε συγκρούσεις με όλα τα δυναμικά αντικείμενα του περιβάλλοντος, συμπεριλαμβανομένου του stylus του μηχανικού βραχίονα, εξασφαλίζοντας την αξιόπιστη ενεργοποίηση της απτικής ανάδρασης. Παράλληλα, η παράμετρος Collision Complexity ορίστηκε στην επιλογή Use Complex Collision As Simple, επιτρέποντας τη χρήση της πραγματικής γεωμετρίας του αντικειμένου για τον υπολογισμό των συγκρούσεων. Ωστόσο, η συγκεκριμένη ρύθμιση δεν υποστηρίζει την εφαρμογή φυσικής προσομοίωσης, γεγονός που την καθιστά κατάλληλη για στατικά ή κινηματικά αντικείμενα που χρησιμοποιούνται αποκλειστικά για απτική ανίχνευση [40].

### 3.5.6 Τοποθέτηση εντός του εικονικού περιβάλλοντος.

Ο εμπλουτισμός του χώρου πραγματοποιήθηκε με τη χρήση του BP\_HapticTrigger Blueprint, στο οποίο αποδόθηκαν ως ορίσματα Static Mesh τα νέα αυτά αντικείμενα, όπως φαίνεται στην Εικόνα 3.22.



Εικόνα 3.22: BP\_HapticTrigger - Custom Collision Mesh

Πίσω από κάθε απτικό αντικείμενο τοποθετήθηκε ένα αντικείμενο το οποίο δεν δέχεται απτικά ερεθίσματα και λειτουργεί αποτρεπτικά σε κάθε μορφή σύγκρουσης. Τα αντικείμενα αυτά χρησιμοποιούνται για την οριοθέτηση της ελάχιστης απόστασης στην οποία μπορεί να προσεγγίσει ο παίκτης το απτικό αντικείμενο, με σκοπό τη διασφάλιση της βέλτιστης εμπειρίας αλληλεπίδρασης και της αποτελεσματικής απτικής ανάδρασης.

### 3.6 Επίπεδο 3<sup>ο</sup>: Αναγνώριση αντικειμένων (Shapes)

#### 3.6.1 Εισαγωγή

Το τρίτο επίπεδο του πειραματικού περιβάλλοντος εστιάζει στη συνολική απτική αντίληψη και στην αναγνώριση αντικειμένων (Shapes) αποκλειστικά μέσω απτικής πληροφορίας. Στόχος του επιπέδου αυτού είναι να διερευνηθεί η ικανότητα του χρήστη να αναγνωρίζει το σχήμα και τη γεωμετρία των αντικειμένων μέσα από την απτική αλληλεπίδραση (tactile interaction), χωρίς την υποστήριξη οπτικών ερεθισμάτων. Η προσέγγιση αυτή επιτρέπει την αξιολόγηση της αποτελεσματικότητας του απτικού συστήματος στη μετάδοση χωρικών και μορφολογικών χαρακτηριστικών, ενισχύοντας την κατανόηση της απτικής αναγνώρισης σε εικονικά περιβάλλοντα [11], [12].

Για την υλοποίηση του παραπάνω πειραματισμού απαιτήθηκε η δημιουργία μιας κατάλληλα διαμορφωμένης σκηνής, όπως φαίνεται στην Εικόνα 3.23, λαμβάνοντας υπόψη τη σωστή θέση του χρήστη και τους λειτουργικούς περιορισμούς του απτικού συστήματος 3D Systems Touch. Ο παίκτης τοποθετείται σε αρχική θέση τέτοια ώστε το αντικείμενο να βρίσκεται σε κατάλληλο ύψος και να υπάρχει επαρκής χώρος, επιτρέποντας στο stylus να κινείται ελεύθερα και να περιηγείται γύρω από το αντικείμενο προς όλες τις κατευθύνσεις.

Σε κάθε αίθουσα τοποθετείται ένα αντικείμενο, το οποίο αναπαρίσταται από ένα απλό γεωμετρικό σχήμα. Καθώς ο χρήστης προχωρά στις διαδοχικές αίθουσες, τα σχήματα γίνονται σταδιακά λιγότερο συνηθισμένα, αυξάνοντας τον βαθμό δυσκολίας της αναγνώρισής τους. Για τη διευκόλυνση της επαλήθευσης του αντικειμένου, παρέχεται στον χρήστη η δυνατότητα, κατά την παραμονή του στην αίθουσα, να ενεργοποιεί ή να απενεργοποιεί τον φωτισμό του αντικειμένου μέσω κουμπιού.

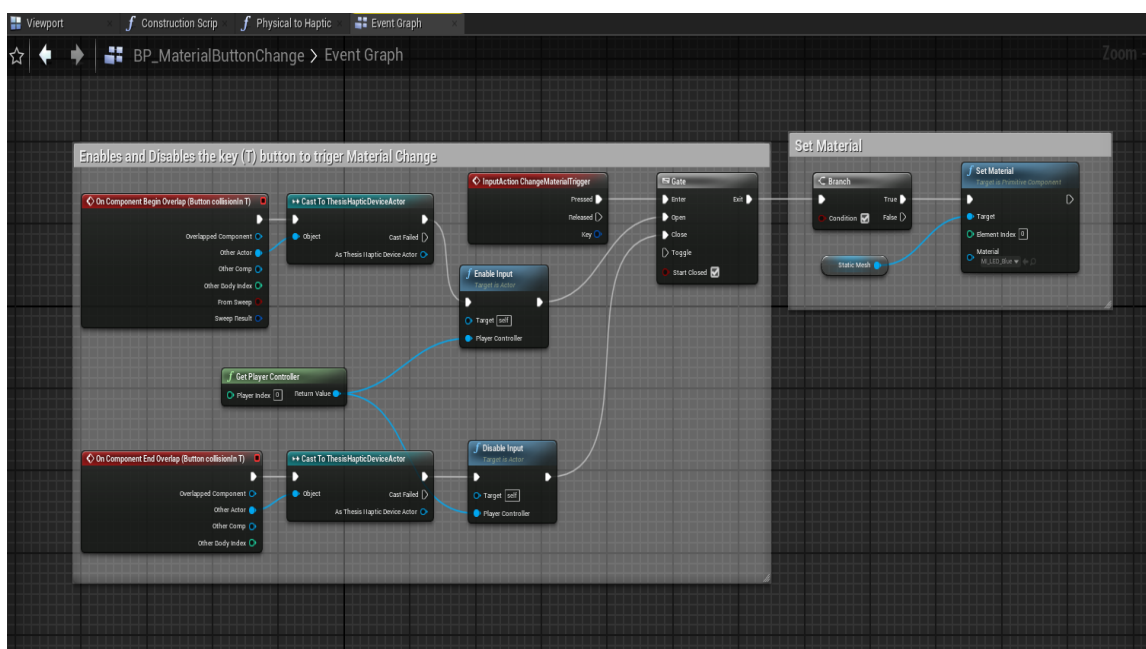


Εικόνα 3.23: Αίθουσες Κρυμμένων Αντικειμένων

Προκειμένου η συγκεκριμένη λειτουργικότητα να είναι επαναχρησιμοποιήσιμη και εύκολα διαχειρίσιμη, αναπτύχθηκε το BP\_MaterialButtonChange.

### 3.6.2 BP\_MaterialButtonChange: Components

Στο Event Graph, όπως φαίνεται στην Εικόνα 3.24, υλοποιήθηκε η λογική που αφορά τη διαχείριση της αλληλεπίδρασης του χρήστη με τα κουμπιά φωτισμού και συσκότισης του αντικειμένου. Συγκεκριμένα, ορίστηκαν γεγονότα επικάλυψης (Overlap Events) τα οποία ενεργοποιούν τις αντίστοιχες αλλαγές κατάστασης.



Εικόνα 3.24: BP\_MaterialButtonChange - Event Graph - Ενέργεια πλήκτρου “T”

#### 3.6.2.1 Ανάλυση των δομικών στοιχείων του Event Graph

Το γεγονός On Component Begin Overlap, το οποίο συνδέεται με το Box Collision Component, ενεργοποιείται όταν ο ενεργός χαρακτήρας (Player), και συγκεκριμένα ο ThesisHapticDeviceActor,

εισέρχεται στην περιοχή επικάλυψης του Box Collision, σηματοδοτώντας την έναρξη της αλληλεπίδρασης. Αντίστοιχα, το γεγονός On Component End Overlap ενεργοποιείται όταν ο χαρακτήρας απομακρυνθεί από την περιοχή αυτή, υποδεικνύοντας τον τερματισμό της επαφής [54], [55].

Η χρήση του Cast To ThesisHapticDeviceActor διασφαλίζει ότι η συγκεκριμένη λειτουργικότητα ενεργοποιείται αποκλειστικά από αυτό τον παίκτη και όχι από οποιοδήποτε άλλο αντικείμενο του περιβάλλοντος [54].

Ο Player Controller είναι υπεύθυνος για τη διαχείριση των εισόδων του χρήστη (πληκτρολόγιο, ποντίκι) και λειτουργεί ως ενδιάμεσος μεταξύ του παίκτη και του ενεργού χαρακτήρα (Player). Μέσω του κόμβου Get Player Controller, το Blueprint αποκτά πρόσβαση στον αντίστοιχο ελεγκτή του παίκτη, επιτρέποντας την αξιοποίηση λειτουργιών που σχετίζονται με τον έλεγχο, όπως η λήψη εισόδου από το πληκτρολόγιο [56].

Με τη χρήση του κόμβου Enable Input, καθίσταται δυνατή η ενεργοποίηση της εισόδου και σε άλλους Actors του επιπέδου, όπως διαδραστικά αντικείμενα, κουμπιά ή μηχανισμοί. Η ενεργοποίηση της εισόδου πραγματοποιείται κατά την είσοδο του παίκτη στην περιοχή αλληλεπίδρασης (Begin Overlap), ενώ αντίστοιχα η είσοδος απενεργοποιείται μέσω του κόμβου Disable Input όταν ο παίκτης απομακρυνθεί από την περιοχή (End Overlap) [57].

Το Input Action αναφέρεται σε έναν μηχανισμό αντιστοίχισης ενεργειών του χρήστη (όπως πάτημα πλήκτρου ή κουμπιού χειριστηρίου) με συγκεκριμένες λειτουργίες του συστήματος.

Τα Input Actions ορίστηκαν αρχικά στις ρυθμίσεις του έργου (Project Settings → Engine → Input), όπου η ενέργεια έλαβε όνομα και συσχετίστηκε με μια είσοδο, συγκριμένα τα πλήκτρα “T” και “R” του πληκτρολογίου. Η ενέργεια αυτή μπορεί πλέον να χρησιμοποιηθεί στα Blueprints μέσω των αντίστοιχων γεγονότων Input Action Events τα οποία ενεργοποιούνται με εκτέλεση ενέργειας σε δύο βασικά σήματα:

- Pressed, το οποίο ενεργοποιείται τη στιγμή που το πλήκτρο πατιέται και
- Released, το οποίο ενεργοποιείται όταν το πλήκτρο αφήνεται.

Στο πλαίσιο διαδραστικών και απτικών εφαρμογών, οι Input Actions χρησιμοποιούνται για τον έλεγχο ενεργειών του χρήστη, όπως η ενεργοποίηση ή απενεργοποίηση φωτισμού, ή έναρξη και τερματισμός αλληλεπίδρασης με αντικείμενα [58].

Ο κόμβος Gate χρησιμοποιείται για τον έλεγχο της ροής εκτέλεσης σε ένα Blueprint, επιτρέποντας ή αποτρέποντας τη διέλευση ενός σήματος ανάλογα με την τρέχουσα κατάστασή του. Λειτουργεί ως μια λογική «πύλη» και διαθέτει τέσσερις βασικές εισόδους:

- Enter: δέχεται το σήμα εκτέλεσης που επιχειρεί να περάσει μέσω της πύλης.
- Open: ανοίγει την πύλη, επιτρέποντας τη διέλευση των επόμενων σημάτων εκτέλεσης.
- Close: κλείνει την πύλη, εμποδίζοντας τη ροή εκτέλεσης.
- Toggle: εναλλάσσει την κατάσταση της πύλης μεταξύ ανοιχτής και κλειστής.

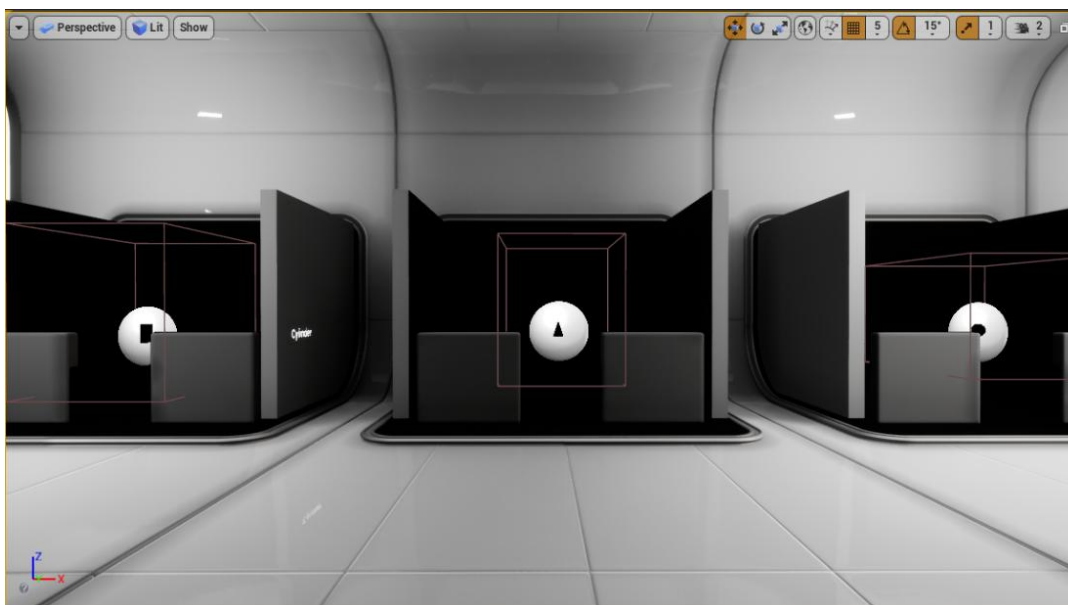
Ο κόμβος Gate χρησιμοποιείται συχνά σε σενάρια όπου απαιτείται έλεγχος της χρονικής ή συνθηματικής εκτέλεσης μιας λειτουργίας. Ενδεικτικό παράδειγμα αποτελεί η ενεργοποίηση εισόδων μόνο όταν ο παίκτης βρίσκεται εντός μιας συγκεκριμένης περιοχής, όπου το γεγονός Begin Overlap

ενεργοποιεί το άνοιγμα της πύλης (Open), ενώ το End Overlap οδηγεί στο κλείσιμό της (Close) [28], [29], [32], [59].

### 3.6.2.2 Ροή του Event Graph

Όταν ο ενεργός χαρακτήρας εισέλθει στην περιοχή που ορίζεται από το Box Collision, ενεργοποιείται το γεγονός On Component Begin Overlap. Το γεγονός αυτό ελέγχει αν ο χαρακτήρας που προκάλεσε την επαφή είναι ο ThesisHapticDeviceActor και, εφόσον αυτό επιβεβαιωθεί, επιτρέπει την ενεργοποίηση (Open) της πύλης (Gate) ώστε να γίνουν αποδεκτές εισόδους (Input Actions) από το πληκτρολόγιο. Η έξοδος της πύλης οδηγεί στην αλλαγή της υφής (Material) του Static Mesh του αντίστοιχου Blueprint.

Στην περίπτωση που δοθεί το Input Action ChangeMaterialTrigger (πλήκτρο “T”), στο Static Mesh εφαρμόζεται ένα Material με ιδιότητα εκπομπής φωτός, μέσω της παραμέτρου Emissive Color. Αντίθετα, όταν δοθεί το Input Action RevertMaterialTrigger (πλήκτρο “R”), το Static Mesh επαναφέρει την αρχική του υφή, η οποία είναι ίδια με αυτή των αντικειμένων που το περιβάλλουν στη σκηνή. Στην Εικόνα 3.25 φαίνεται το ιδικό αντικείμενο μέσα στην αίθουσα και τα όρια του Box Collision που ενεργοποιεί τις εισόδους από το πληκτρολόγιο.



Εικόνα 3.25: Τα αντικείμενα μέσα στο χώρο

### 3.6.3 Τερματισμός της Εφαρμογής Μέσα από το επίπεδο

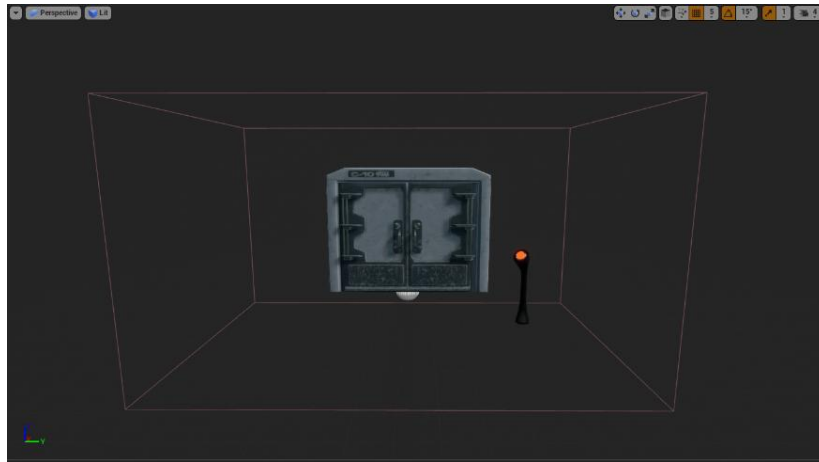
Ο τερματισμός της εφαρμογής μπορεί να πραγματοποιηθεί με δύο τρόπους: είτε μέσω του τελευταίου επιπέδου της εφαρμογής είτε μέσω του κεντρικού μενού. Η διαδικασία τερματισμού μέσω του κεντρικού μενού παρουσιάζεται σε επόμενη ενότητα, ενώ η παρούσα ενότητα εστιάζει στον τερματισμό που πραγματοποιείται εντός του επιπέδου.

Προκειμένου η συγκεκριμένη λειτουργικότητα να είναι επαναχρησιμοποιήσιμη και εύκολα διαχειρίσιμη, αναπτύχθηκε το BP\_Quit.

#### 3.6.3.1 BP\_Quit: Components

Το συγκεκριμένο Blueprint περιλαμβάνει στο DefaultSceneRoot, τέσσερα Static Mesh Components, τα οποία καθορίζουν την οπτική και αισθητική αναπαράσταση του αντικειμένου, όπως φαίνεται στην

Εικόνα 3.26. Επιπλέον, περιλαμβάνει ένα Button Collision Component, το οποίο χρησιμοποιείται για την ανίχνευση αλληλεπίδρασης με τον χρήστη και την ενεργοποίηση της λειτουργίας τερματισμού της εφαρμογής.



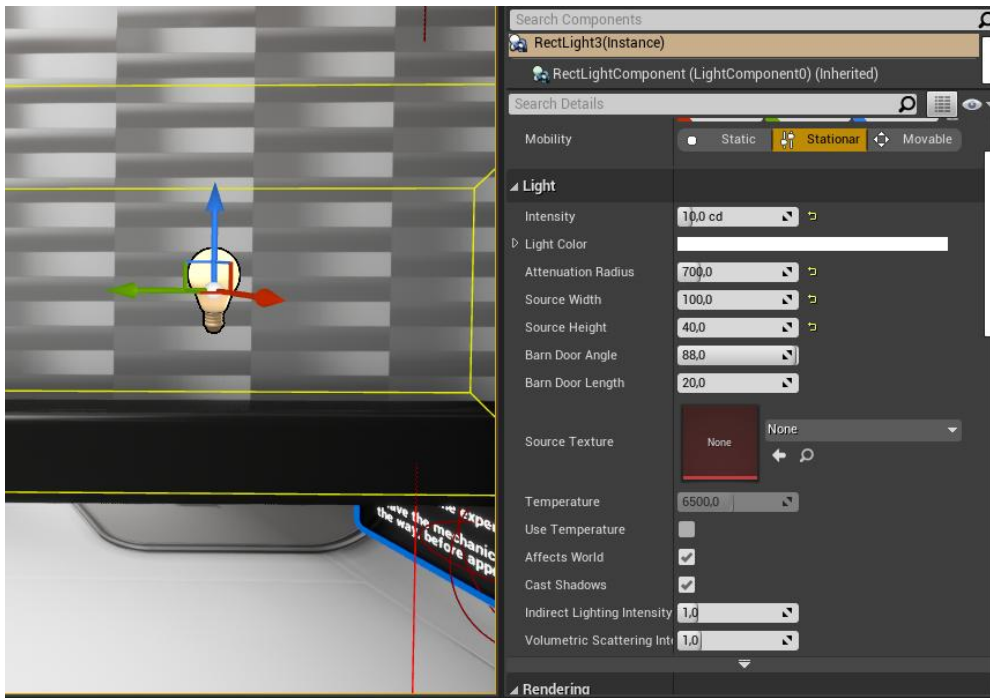
Εικόνα 3.26: BP\_Quit – Viewport

#### 3.6.3.2 BP\_Quit: Event Graph

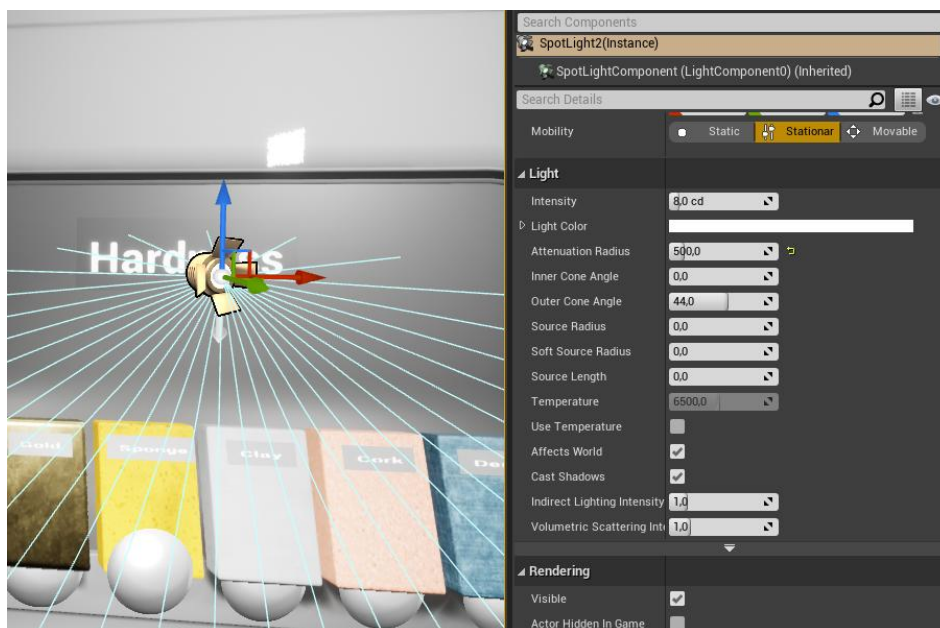
Η λογική υλοποίησης που ακολουθήθηκε είναι αντίστοιχη με εκείνη του Blueprint BP\_MaterialButtonChange. Όταν ο ενεργός χαρακτήρας εισέλθει στην περιοχή που ορίζεται από το Box Collision, ενεργοποιείται το γεγονός On Component Begin Overlap. Το γεγονός αυτό ελέγχει αν ο χαρακτήρας που προκάλεσε την επαφή είναι ο ThesisHapticDeviceActor και, εφόσον αυτό επιβεβαιωθεί, επιτρέπει την ενεργοποίηση (Open) της πύλης (Gate) ώστε να γίνεται αποδεκτή η είσοδος (Input Action Exit Game, πλήκτρο “G”) από το πληκτρολόγιο. Η έξοδος της πύλης οδηγεί στην μέθοδο Quit Game του συστήματος η οποία τερματίζει τη συνεδρία και κλείνει την εφαρμογή.

#### 3.6.4 Εσωτερικός Φωτισμός

Σε όλα τα επίπεδα του έργου χρησιμοποιήθηκαν δύο βασικά είδη φωτισμού, τα Rect Light και τα Spot Light, με στόχο την επίτευξη ισορροπημένου και λειτουργικού φωτισμού στο εικονικό περιβάλλον, όπως φαίνεται στην Εικόνα 3.27 και στην Εικόνα 3.28. Τα Rect Light εφαρμόστηκαν για τον γενικό φωτισμό του χώρου, καθώς προσφέρουν εκτεταμένη και ομοιόμορφη κατανομή φωτός, μειώνοντας τις έντονες σκιές και εξασφαλίζοντας επαρκή ορατότητα. Παράλληλα, τα Spot Light τοποθετήθηκαν σε κατάλληλες θέσεις ώστε να παρέχουν κατευθυνόμενο φωτισμό στα πειράματα, συμβάλλοντας στην ομοιόμορφη φωταγώγησή τους και στη σαφή οπτική εστίαση του χρήστη. Ο συνδυασμός των δύο τύπων φωτισμού επιλέχθηκε ώστε να επιτυγχάνεται τόσο η συνολική φωταγώγηση του χώρου όσο και ο στοχευμένος φωτισμός των πειραμάτων, χωρίς να επηρεάζεται αρνητικά η απόδοση ή η λειτουργικότητα της εφαρμογής [59].



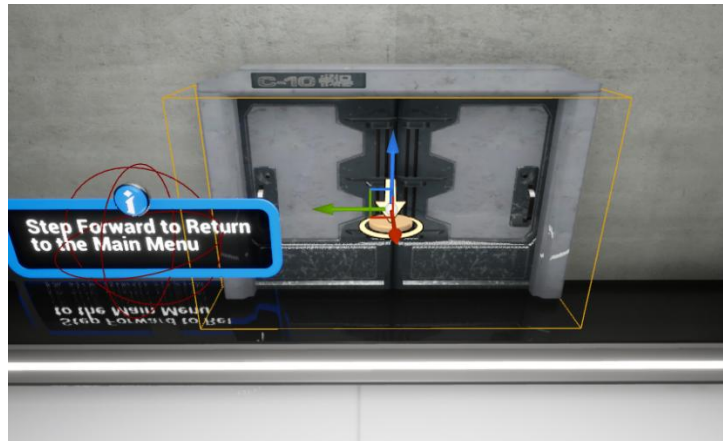
Εικόνα 3.27: Rect Light



Εικόνα 3.28: Spot Light

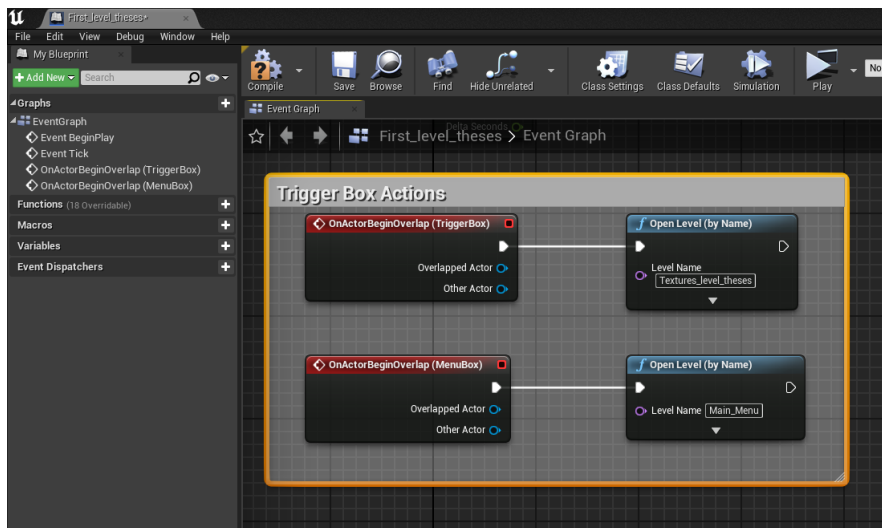
### 3.6.5 Πυροδοτητές Επιπέδου (Trigger Box)

Σε κάθε επίπεδο τοποθετήθηκαν δύο πυροδοτητές (Trigger Box), ένας στην αρχή και ένας στο τέλος του επιπέδου, οι οποίοι ενεργοποιούνται όταν ο ενεργός χαρακτήρας εισέρχεται εντός των ορίων τους, όπως φαίνεται στην Εικόνα 3.29. Ο προγραμματισμός της λειτουργίας τους υλοποιήθηκε στο Event Graph του Level Blueprint κάθε επιπέδου, όπου καθορίζεται η αντίστοιχη συμπεριφορά που θα εκτελείται κατά την ενεργοποίησή τους.



Εικόνα 3.29: Trigger Box

Με την είσοδο του ενεργού χαρακτήρα στην αντίστοιχη περιοχή ενεργοποιείται το γεγονός ActorBeginOverlap, το οποίο καλεί τη συνάρτηση Open Level (By Name) [27]. Η συνάρτηση αυτή δέχεται ως είσοδο το όνομα ενός επιπέδου και πραγματοποιεί τη μετάβαση από το τρέχον επίπεδο στο καθορισμένο, επιτρέποντας την εναλλαγή και τη φόρτωση νέου περιβάλλοντος εντός της εφαρμογής, όπως φαίνεται στην Εικόνα 3.30 [55], [60].



Εικόνα 3.30: Level Blueprint - Event Graph

Ο πυροδοτητής που βρίσκεται στην αρχή κάθε επιπέδου χρησιμοποιείται για τη μεταφορά του χρήστη πίσω στο αρχικό μενού της εφαρμογής. Αντίστοιχα, ο πυροδοτητής που τοποθετήθηκε στο τέλος κάθε επιπέδου οδηγεί τον χρήστη στο επόμενο επίπεδο. Εξάιρεση αποτελεί το τελευταίο επίπεδο της εφαρμογής, όπου ο αντίστοιχος πυροδοτητής κατευθύνει επίσης τον χρήστη πίσω στο αρχικό μενού, ολοκληρώνοντας τον κύκλο πλοήγησης της εφαρμογής.

### 3.7 Παράθυρα οδηγιών και Επικεφαλίδες

Τα παράθυρα οδηγιών που τοποθετήθηκαν στον χώρο προέρχονται από το βασικό πακέτο εκμάθησης Blueprint Office της Epic Games και χρησιμοποιήθηκαν για την παροχή πληροφοριών και οδηγιών προς τον χρήστη κατά τη διάρκεια της αλληλεπίδρασης. Τα συγκεκριμένα στοιχεία συμβάλλουν στη βελτίωση της κατανόησης των λειτουργιών της εφαρμογής και διευκολύνουν τη διαδικασία εξοικείωσης του χρήστη με το περιβάλλον.

Οι επικεφαλίδες αποτελούν μια απλοποιημένη εκδοχή του Blueprint των παραθύρων οδηγίων, από την οποία έχουν αφαιρεθεί τα περιγράμματα καθώς και η λειτουργία αυτόματης αλλαγής προσανατολισμού προς την κατεύθυνση του παίκτη. Επιπλέον, στη λειτουργικότητά τους προστέθηκε η δυνατότητα μεταβολής του μεγέθους της γραμματοσειράς, ώστε να επιτυγχάνεται καλύτερη ορατότητα και προσαρμογή του κειμένου ανάλογα με τις ανάγκες του εκάστοτε χώρου.

### 3.8 Κεντρικό Μενού Εφαρμογής

#### 3.8.1 Εισαγωγή

Μετά την κατασκευή του περιβάλλοντος, το επόμενο βήμα ήταν να υπάρχει ένα κεντρικό μενού μέσω του οποίου να μπορεί να ξεκινήσει το έργο. Το κεντρικό μενού αποτελεί το βασικό σημείο εισόδου του χρήστη στην εφαρμογή και παίζει καθοριστικό ρόλο στη συνολική εμπειρία χρήσης. Μέσω αυτού, ο χρήστης αποκτά πρόσβαση στις κύριες λειτουργίες της εφαρμογής, όπως η έναρξη της αλληλεπίδρασης, η πλοήγηση μεταξύ των επιπέδων και ο τερματισμός της εφαρμογής. Ο σχεδιασμός και η υλοποίηση του κεντρικού μενού στοχεύουν στην παροχή μιας σαφούς, κατανοητής και λειτουργικής διεπαφής, η οποία διευκολύνει την πλοήγηση και μειώνει τη γνωστική επιβάρυνση του χρήστη. Στην παρούσα ενότητα παρουσιάζονται η δομή του κεντρικού μενού, τα επιμέρους στοιχεία του, καθώς και ο τρόπος με τον οποίο υλοποιούνται οι βασικές του λειτουργίες.

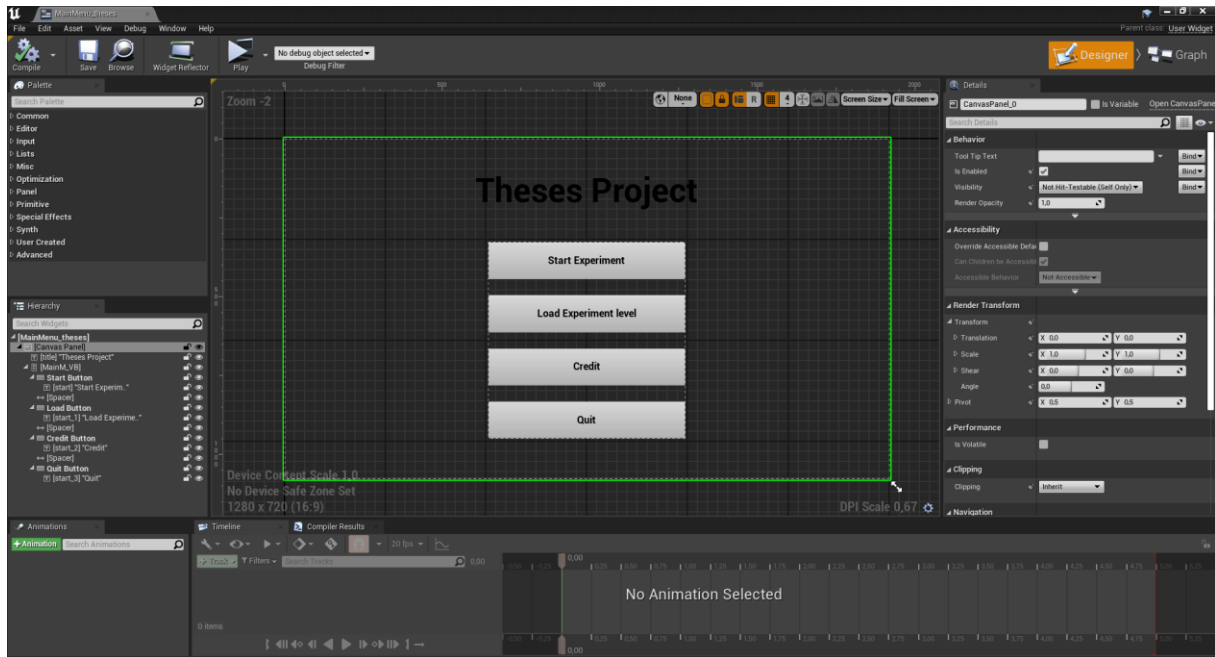
#### 3.8.2 Δημιουργία Κεντρικού Μενού

Η ανάπτυξη της εφαρμογής συνεχίστηκε με τη δημιουργία ενός κεντρικού μενού, το οποίο παρέχει στον χρήστη τη δυνατότητα εκκίνησης και πλοήγησης στο σύστημα. Στο Unreal Engine, το μενού μπορεί να υλοποιηθεί ως ένας ξεχωριστός χάρτης (map), ο οποίος περιλαμβάνει ένα δισδιάστατο φόντο και μία κάμερα προσανατολισμένη προς αυτό. Εναλλακτικά, το φόντο μπορεί να αντικατασταθεί από τρισδιάστατο περιβάλλον, κινούμενη εικόνα, βίντεο ή συνδυασμό εικόνας και τρισδιάστατων οπτικών εφέ.

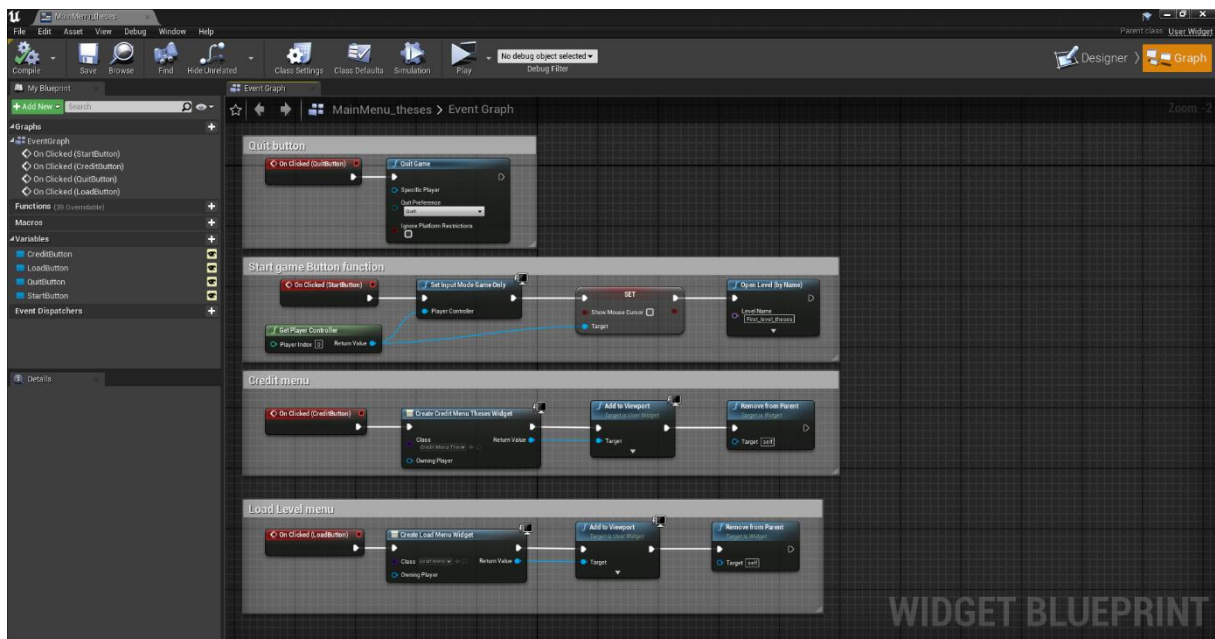
Για τις ανάγκες του παρόντος έργου, τα μενού υλοποιήθηκαν με τη χρήση του συστήματος UMG (Unreal Motion Graphics). Το μενού αποτελείται από στοιχεία διεπαφής χρήστη (Widgets), τα οποία δημιουργούνται μέσω του Content Browser, είτε με δεξί κλικ είτε μέσω της επιλογής Add/Import, επιλέγοντας έναν τύπο Widget Blueprint. Τα Widgets είναι Blueprints τα οποία έχουν ως γονική κλάση το User Widget αντί του Actor, κληρονομώντας ειδικές μεταβλητές και μεθόδους που αφορούν τη διεπαφή χρήστη [33], [34].

##### 3.8.2.1 Δομή και Λειτουργία Κεντρικού Μενού

Στο παράθυρο Viewport του UMG Editor τοποθετήθηκε ένα αντικείμενο Canvas Panel, το οποίο καλύπτει ολόκληρη την επιφάνεια του μενού, όπως φαίνεται στην Εικόνα 3.31. Εντός του Canvas Panel, με ιεραρχική διάταξη, προστέθηκε ένα αντικείμενο κειμένου που εμφανίζει τον τίτλο του έργου, καθώς και ένα Horizontal Box, το οποίο χρησιμοποιείται για την οργάνωση των κουμπιών του κεντρικού μενού. Στο εσωτερικό του Horizontal Box τοποθετήθηκαν τρία αντικείμενα Button, καθένα από τα οποία περιέχει ένα αντικείμενο κειμένου με την αντίστοιχη ονομασία της επιλογής. Η λογική διαχείρισης των αλληλεπιδράσεων υλοποιείται στο γράφημα του Widget Blueprint, όπως παρουσιάζεται στην Εικόνα 3.32.



Εικόνα 3.31: Κεντρικό Μενού – Designer



Εικόνα 3.32: Κεντρικό Μενού - Graph

### 3.8.2.2 Επιλογές Κεντρικού Μενού

- Start Experiment

Η επιλογή Start Experiment εκκινεί το διαδραστικό μέρος της εφαρμογής από την αρχή. Η λειτουργία αυτή υλοποιείται με την εντολή Open Level, με είσοδο τον πρώτο χάρτη της εφαρμογής.

- Load Experiment Level

Η επιλογή Load Experiment Level φορτώνει το πρώτο υπομενού, το οποίο παρέχει στον χρήστη τη δυνατότητα επιλογής του διαδραστικού περιβάλλοντος στο οποίο θα εισέλθει, καθώς και την επιλογή επιστροφής στο κεντρικό μενού. Η μετάβαση επιτυγχάνεται μέσω της εντολής Open Level, με είσοδο τον χάρτη που αντιστοιχεί στο επιλεγμένο πειραματικό περιβάλλον [60].

- Credits

Η επιλογή Credits φορτώνει το δεύτερο υπομενού της εφαρμογής, στο οποίο παρουσιάζονται οι συντελεστές και οι αναφορές που σχετίζονται με την ανάπτυξη του έργου. Το υπομενού περιλαμβάνει επίσης κουμπί επιστροφής στο κεντρικό μενού.

- Quit Game

Η επιλογή Quit Game τερματίζει τη συνεδρία και κλείνει την εφαρμογή. Η λειτουργία αυτή υλοποιείται μέσω της εντολής Blueprint Quit Game, η οποία διακόπτει την εκτέλεση της εφαρμογής.

### 3.9 Χαρακτήρας (Playable Character)

#### 3.9.1 Εισαγωγή

Ο χαρακτήρας του παίκτη αποτελεί το βασικό μέσο αλληλεπίδρασης του χρήστη με το εικονικό περιβάλλον της εφαρμογής. Μέσω αυτού, ο χρήστης μπορεί να πλοηγείται στον χώρο, να αλληλεπιδρά με τα αντικείμενα και να ενεργοποιεί τις λειτουργίες που έχουν υλοποιηθεί στο πλαίσιο του έργου.

Το πρόσθετο Haptics Direct της εταιρείας 3D Systems παρέχει πρόσβαση σε έναν προκαθορισμένο χαρακτήρα, ο οποίος είναι κατάλληλα προγραμματισμένος ώστε να αλληλεπιδρά με τα απτικά αντικείμενα του εικονικού χώρου και να μεταφέρει με ακρίβεια τις πληροφορίες της αλληλεπίδρασης στον μηχανικό βραχίονα 3D Systems Touch.

#### 3.9.2 Πρόβλημα και Λύση

Το βασικό πρόβλημα που προκύπτει με τη χρήση του συγκεκριμένου χαρακτήρα εμφανίζεται κατά την ανάγκη μετακίνησής του εντός του εικονικού χώρου. Συγκεκριμένα, από τον αρχικό του σχεδιασμό ο χαρακτήρας αυτός έχει οριστεί ως τύπος AActor, γεγονός που έχει ως αποτέλεσμα η μηχανή Unreal Engine να τον αντιμετωπίζει ως αντικείμενο του περιβάλλοντος και όχι ως χαρακτήρα που μπορεί να ελεγχθεί άμεσα από τον χρήστη (playable character).

Για την επίλυση του συγκεκριμένου προβλήματος χρειάστηκε να πραγματοποιηθούν ορισμένες τροποποιήσεις στον πηγαίο κώδικα του πρόσθετου.

Η επιλογή της κλάσης Character αντί της κλάσης Actor κρίθηκε απαραίτητη, καθώς η Character παρέχει ενσωματωμένους μηχανισμούς που υποστηρίζουν την κίνηση και τον άμεσο έλεγχο από τον χρήστη. Συγκεκριμένα, η κλάση Character ενσωματώνει το Character Movement Component, το οποίο προσφέρει λειτουργίες όπως μετακίνηση, έλεγχο προσανατολισμού και διαχείριση σύγκρουσης, χωρίς να απαιτείται η υλοποίησή τους από την αρχή. Επιπλέον, η χρήση της κλάσης Character επιτρέπει την εύκολη αντιστοίχιση εισόδων χρήστη (Input Actions) με ενέργειες κίνησης, καθιστώντας τον χαρακτήρα άμεσα ελέγξιμο μέσα στο περιβάλλον της Unreal Engine. Η μετάβαση αυτή διευκολύνει την υλοποίηση της πλοήγησης του χαρακτήρα στον εικονικό χώρο, ενώ παράλληλα διατηρεί τη συμβατότητα με τα απτικά χαρακτηριστικά του πρόσθετου[54].

#### 3.9.3 Τροποποιήσεις

Οι παρεμβάσεις αυτές αφορούσαν αποκλειστικά τον τρόπο με τον οποίο ο χαρακτήρας ορίζεται, ώστε να είναι δυνατή η ορθή διαχείρισή του από τη μηχανή Unreal Engine, με στόχο τη μετατροπή του σε οντότητα που μπορεί να ελεγχθεί από τον χρήστη και να μετακινείται εντός του εικονικού χώρου. Επιπλέον, πραγματοποιήθηκαν ορισμένες τροποποιήσεις στα επιμέρους αντικείμενα από τα οποία αποτελείται ο χαρακτήρας, καθώς και στην ιεραρχία και στη διαδικασία αρχικοποίησής τους.

Οι τροποποιήσεις που πραγματοποιήθηκαν διακρίνονται σε δύο βασικές κατηγορίες. Η πρώτη κατηγορία περιλαμβάνει αλλαγές στο αρχείο `HapticDeviceActor.h`, στο οποίο ορίζονται η κλάση και τα δημόσια μέλη της. Η δεύτερη κατηγορία αφορά αλλαγές στο αρχείο `HapticDeviceActor.cpp`, το οποίο περιέχει την υλοποίηση των λειτουργιών της κλάσης, σύμφωνα με τη συνήθη πρακτική ανάπτυξης λογισμικού σε C++. Τα αρχεία αυτά βρίσκονται στον φάκελο Source του πρόσθετου.

Οι τροποποιήσεις αυτές δεν επηρεάζουν με κανέναν τρόπο τη λειτουργικότητα του πρόσθετου, ούτε αλλοιώνουν τον βασικό τρόπο λειτουργίας του.

### 3.9.3.1 Τροποποιήσεις στο Header file

Αρχικά, στα αρχεία `include` που χρησιμοποιούνται για την αρχικοποίηση των απαραίτητων βιβλιοθηκών και των εξαρτήσεων (*dependencies*), αφαιρέθηκε το αρχείο κεφαλίδας που αφορούσε τον Actor, καθώς δεν χρησιμοποιείται πλέον. Στη θέση του προστέθηκε το κατάλληλο αρχείο κεφαλίδας για τον Character, όπως φαίνεται στην Εικόνα 3.33. Η τροποποίηση αυτή επιτρέπει την κληρονόμηση των λειτουργιών που παρέχονται από την κλάση Character.

```

5  #include "CoreMinimal.h"
6  #include "GameFramework/Character.h"
7  #include "Components/BoxComponent.h"
8  #include "Physics/PhysScene.h"

```

Εικόνα 3.33: Αρχείο κεφαλίδας του Character

Η επόμενη τροποποίηση αφορά τη δήλωση της κλάσης `AHapticDeviceActor`, η οποία πλέον ορίζεται ως υποκλάση της `ACharacter`. Συγκεκριμένα, η κλάση δηλώνεται ως `public ACharacter`, ώστε να κληρονομεί τις λειτουργίες και τα χαρακτηριστικά που παρέχει η κλάση Character, όπως φαίνεται στην Εικόνα 3.34.

```

33
34  UCLASS()
35  class HAPTICDEVICEINTERFACE_API AHapticDeviceActor : public ACharacter
36  {
37      GENERATED_BODY()
38

```

Εικόνα 3.34: Δήλωση της κλάσης AHapticDeviceActor

Τέλος, προστέθηκε ένα στοιχείο κάμερας στον χαρακτήρα, μέσω της δήλωσης ενός Camera Component στην κλάση, όπως φαίνεται στην Εικόνα 3.35. Η συγκεκριμένη δήλωση καθιστά το στοιχείο κάμερας ορατό στο περιβάλλον του Unreal Editor και προσβάσιμο από τα Blueprints μόνο για ανάγνωση, επιτρέποντας την οπτική αναπαράσταση της σκηνής από την οπτική γωνία του χαρακτήρα.

```

120
121  UPROPERTY(VisibleAnywhere, BlueprintReadOnly, Category = Camera)
122      class UCameraComponent* CameraComp;
123

```

Εικόνα 3.35: Προσθήκη στοιχείου Κάμερας

### 3.9.3.2 Τροποποιήσεις στο Source file

Αρχικά, αφαιρέθηκαν οι εντολές που αφορούσαν τη δημιουργία και τον ορισμό του Root Component και στη θέση τους δημιουργήθηκε ένα UCapsuleComponent, όπως φαίνεται στην Εικόνα 3.36. Στην κλάση Character, το UCapsuleComponent χρησιμοποιείται ως κύριο στοιχείο σύγκρουσης και λειτουργεί ως Root Component, επάνω στο οποίο βασίζεται η κίνηση και η φυσική συμπεριφορά του χαρακτήρα. Η γεωμετρία του έχει τη μορφή κάψουλας, γεγονός που επιτρέπει ομαλή και σταθερή αλληλεπίδραση με το περιβάλλον. Έπειτα, πραγματοποιήθηκε έλεγχος ώστε να διασφαλιστεί ότι το συγκεκριμένο στοιχείο έχει δημιουργηθεί σωστά και δεν δείχνει σε κενή θέση μνήμης (null pointer), ενώ παράλληλα αρχικοποιήθηκαν οι διαστάσεις του σύμφωνα με τις απαιτήσεις της εφαρμογής.

```

24
25     // Create Components
26     //RootComponent = CreateDefaultSubobject<USceneComponent>("Haptic Device");
27     //this->SetRootComponent(RootComponent);
28
29     UCapsuleComponent* CapsulComp = GetCapsuleComponent();
30     check(CapsulComp != nullptr);
31     CapsulComp->InitCapsuleSize(23.9f, 90.0f);
32

```

Εικόνα 3.36: Δημιουργία UCapsuleComponent

Στη συνέχεια πραγματοποιήθηκε η δημιουργία και η αρχικοποίηση ενός στοιχείου κάμερας πρώτου προσώπου για τον χαρακτήρα, όπως φαίνεται στην Εικόνα 3.37. Αρχικά, δημιουργήθηκε ένα UCameraComponent μέσω της εντολής CreateDefaultSubobject, το οποίο χρησιμοποιείται για την οπτική αναπαράσταση της σκηνής από την οπτική γωνία του παίκτη. Παράλληλα, πραγματοποιήθηκε έλεγχος ώστε να διασφαλιστεί ότι το στοιχείο κάμερας έχει δημιουργηθεί επιτυχώς και δεν δείχνει σε κενή θέση μνήμης, αποτρέποντας ενδεχόμενα σφάλματα κατά την εκτέλεση. Επιπλέον, ορίστηκε η αρχική θέση της κάμερας σε σχετικές συντεταγμένες ως προς το βασικό σώμα του χαρακτήρα και ενεργοποιήθηκε η δυνατότητα ελέγχου της περιστροφής της μέσω των εισόδων του παίκτη, επιτρέποντας την ελεύθερη μεταβολή της οπτικής γωνίας. Τέλος, το στοιχείο κάμερας προσαρτήθηκε στο UCapsuleComponent του χαρακτήρα, διασφαλίζοντας ότι η κάμερα ακολουθεί τη θέση και την κίνηση του χαρακτήρα εντός του εικονικού περιβάλλοντος.

```

32
33     // Create a first person camera component.
34     CameraComp = CreateDefaultSubobject<UCameraComponent>(TEXT("FirstPersonCamera"));
35     check(CameraComp != nullptr);
36     // Position the camera slightly above the eyes.
37     CameraComp->SetRelativeLocation(FVector(-70.0f, 0.0f, 4.0f));
38     // Enable the pawn to control camera rotation.
39     CameraComp->bUsePawnControlRotation = true;
40     // Attach the camera component to our capsule component.
41     CameraComp->SetupAttachment(CastChecked<USceneComponent, UCapsuleComponent>(CapsulComp));
42

```

Εικόνα 3.37: Δημιουργία UCameraComponent

Τέλος, έγιναν οι απαραίτητες αλλαγές ώστε όλα τα Components να προσαρτηθούν (attach) στο UCapsuleComponent για να κινούνται σαν μια ενιαία οντότητα.

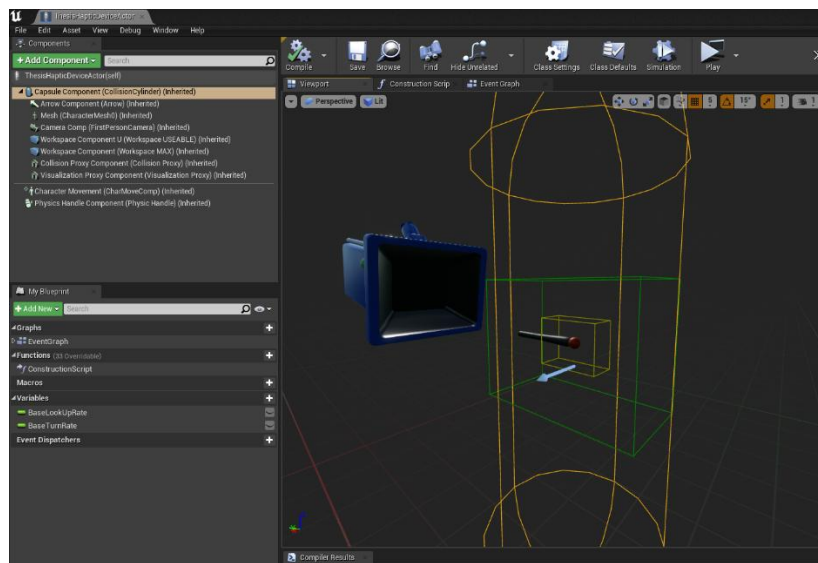
### 3.9.4 Τεχνικές Ρυθμίσεις.

Μετά την ολοκλήρωση και την αποθήκευση των αλλαγών, το πρόσθετο απαιτείται να περάσει εκ νέου από τη διαδικασία αναδιαμόρφωσης (Re-Build), καθώς αποτελεί τμήμα του πυρήνα της μηχανής.

Η συγκεκριμένη διαδικασία δεν είναι απαραίτητη για πρόσθετα που εντάσσονται απευθείας στο εσωτερικό του έργου και δεν επηρεάζουν τον βασικό μηχανισμό λειτουργίας της Unreal Engine.

### 3.9.5 Δημιουργία και Παραμετροποίηση του ThesisHapticDeviceActor Blueprint

Με την ολοκλήρωση της διαδικασίας αναδιαμόρφωσης, ο νέος HapticDeviceActor καθίσταται ορατός στον Unreal Editor ως κλάση C++. Ωστόσο, επειδή το HapticDeviceActor.h αποτελεί κλάση γραμμένη σε C++, για να καταστεί δυνατή η περαιτέρω επεξεργασία και παραμετροποίηση του χαρακτήρα, απαιτείται η έκθεσή του σε Blueprint. Η διαδικασία αυτή πραγματοποιείται επιλέγοντας τον HapticDeviceActor που βρίσκεται στη σκηνή και μέσω του παραθύρου Details να επιλεγεί το Blueprint / Add Script, το οποίο δημιουργεί ένα αντίστοιχο Blueprint βασισμένο στην C++ κλάση. Το Blueprint ονομάστηκε ThesisHapticDeviceActor και ως κλάση γονέας του επιλέχθηκε η HapticDeviceActor, όπως φαίνεται στην Εικόνα 3.38.



Εικόνα 3.38: ThesisHapticDeviceActor Blueprint

Τα Components του Blueprint μπορούν να διακριθούν σε δύο νοητές ομάδες, με βάση την κλάση από την οποία κληρονομούνται. Η πρώτη ομάδα περιλαμβάνει τα στοιχεία που κληρονομούνται από την κλάση Character, συγκεκριμένα τα Capsule, Arrow, Mesh και Camera, τα οποία σχετίζονται με τη βασική δομή, την κίνηση και την οπτική αναπαράσταση του χαρακτήρα. Η δεύτερη ομάδα αποτελείται από τα στοιχεία που κληρονομούνται από την κλάση HapticDeviceActor, και περιλαμβάνει τα Workspace, Collision Proxy και Visualization Proxy, τα οποία είναι υπεύθυνα για τη διαχείριση της απτικής αλληλεπίδρασης.

#### 3.9.5.1 Ανάλυση των Component

- Το Capsule αποτελεί το βασικό στοιχείο σύγκρουσης του χαρακτήρα και ορίζει το φυσικό του περίγραμμα στο εικονικό περιβάλλον. Μέσω αυτού, η μηχανή Unreal Engine διαχειρίζεται την ανίχνευση συγκρούσεων, την απόκριση σε εμπόδια και τη φυσική αλληλεπίδραση του χαρακτήρα με τα αντικείμενα του χώρου.
- Το Arrow χρησιμοποιείται ως βοηθητικό στοιχείο προσανατολισμού και υποδεικνύει τη φορά και την κατεύθυνση του χαρακτήρα στον χώρο. Το συγκεκριμένο στοιχείο δεν συμμετέχει στη φυσική προσομοίωση ή στην αλληλεπίδραση, αλλά διευκολύνει την οπτική κατανόηση.
- Το Mesh αποτελεί την οπτική αναπαράσταση του χαρακτήρα στο εικονικό περιβάλλον. Αν και στην παρούσα υλοποίηση δεν χρησιμοποιείται εκτενώς για την απεικόνιση του παίκτη, παραμένει

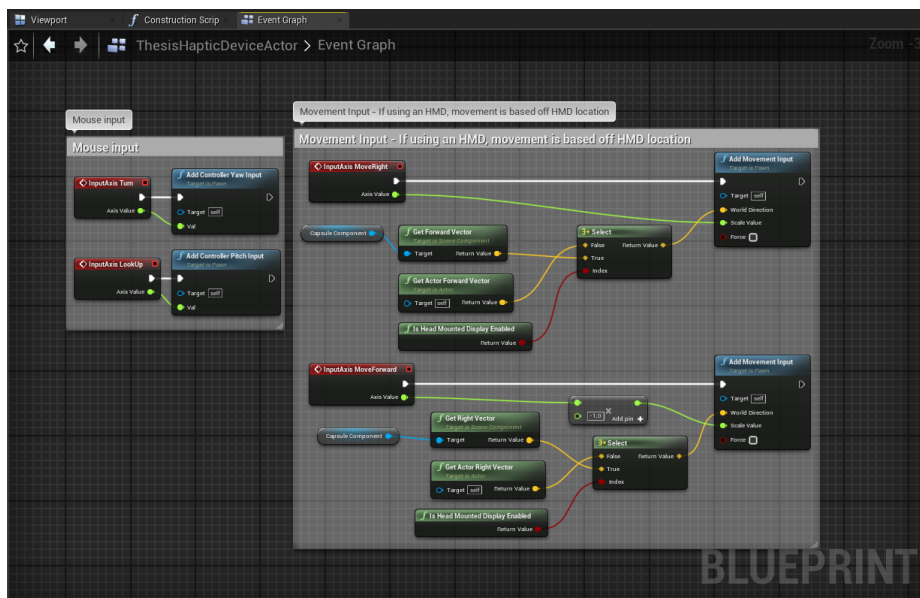
σημαντικό στοιχείο για τη σωστή τοποθέτηση και ευθυγράμμιση των επιμέρους components του χαρακτήρα.

- Το Camera είναι υπεύθυνο για την απεικόνιση της σκηνής από την οπτική γωνία του χρήστη. Στην παρούσα εργασία χρησιμοποιείται κάμερα πρώτου προσώπου, η οποία είναι προσαρτημένη στον χαρακτήρα και ακολουθεί τη θέση και την κίνησή του, επιτρέποντας την άμεση και φυσική αλληλεπίδραση με το εικονικό περιβάλλον.
- Τα Workspace ορίζουν την ελάχιστη και τη μέγιστη απόσταση εντός της οποίας μπορεί να κινείται το Stylus στον τρισδιάστατο χώρο. Τα όρια αυτά έχουν υπολογιστεί από την εταιρεία 3D Systems, ώστε να αντιστοιχούν με ακρίβεια στις πραγματικές αποστάσεις που μπορεί να καλύψει ο μηχανικός βραχίονας της απτικής συσκευής.
- Το Collision Proxy αποτελεί το στοιχείο που έρχεται σε άμεση επαφή με τα υπόλοιπα αντικείμενα του εικονικού περιβάλλοντος και είναι υπεύθυνο για τη λήψη και τη διαχείριση των απτικών ερεθισμάτων κατά την αλληλεπίδραση.
- Το Visualization Proxy είναι ένα Static Mesh που χρησιμοποιείται για την οπτική αναπαράσταση του Stylus στον τρισδιάστατο χώρο, επιτρέποντας στον χρήστη να παρακολουθεί τη θέση και την κίνησή του σε πραγματικό χρόνο.

Για την αποφυγή εσωτερικών συγκρούσεων πραγματοποιήθηκαν ειδικές ρυθμίσεις στο Collision του Capsule, ώστε να αγνοεί τα επιμέρους Components που χρησιμοποιούν Simulate Physics. Με τον τρόπο αυτό αποφεύγονται ανεπιθύμητες αλληλεπιδράσεις μεταξύ των εσωτερικών στοιχείων του χαρακτήρα.

### 3.9.5.2 ThesisHapticDeviceActor: Event Graph

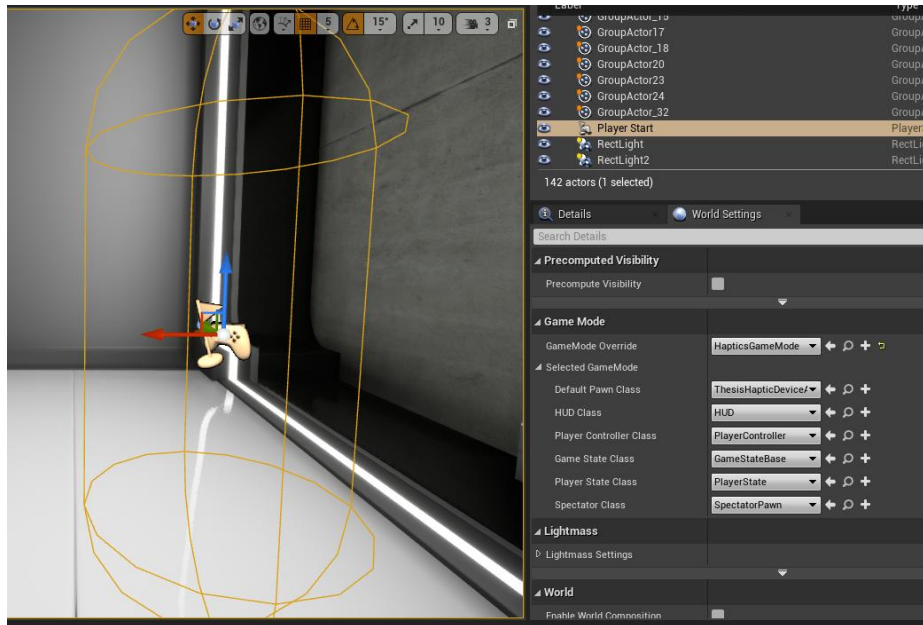
Στο Event Graph, όπως φαίνεται στην Εικόνα 3.39, υλοποιήθηκαν οι βασικές λειτουργίες κίνησης και ελέγχου της κάμερας, αξιοποιώντας τη λογική που παρέχεται από το First Person Character Blueprint της Unreal Engine. Συγκεκριμένα, υλοποιήθηκε ο χειρισμός της περιστροφής της κάμερας μέσω του ποντικιού, καθώς και η μετακίνηση του χαρακτήρα στον χώρο με τη χρήση των πλήκτρων WASD, επιτρέποντας την πλοήγηση του χρήστη στο εικονικό περιβάλλον με τρόπο αντίστοιχο ενός χαρακτήρα πρώτου προσώπου.



Εικόνα 3.39: ThesisHapticDeviceActor - Event Graph

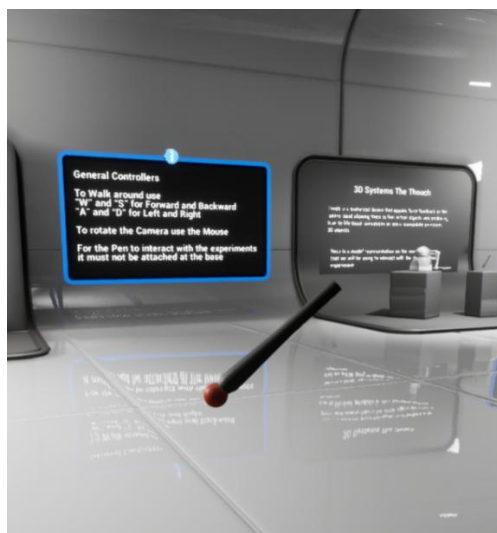
### 3.9.6 Αρχικοποίηση Θέσης Παίκτη

Σε κάθε επίπεδο τοποθετήθηκε ένα αντικείμενο Player Start, όπως φαίνεται στην Εικόνα 3.40, το οποίο χρησιμοποιείται για τον καθορισμό της αρχικής θέσης του παίκτη. Το Player Start λαμβάνει τις απαραίτητες πληροφορίες από το Default Pawn Class που έχει οριστεί στο Game Mode του επιπέδου, όπως αυτό έχει καθοριστεί μέσω των World Settings, εξασφαλίζοντας τη σωστή αρχικοποίηση και τοποθέτηση του χαρακτήρα κατά την έναρξη της εφαρμογής.



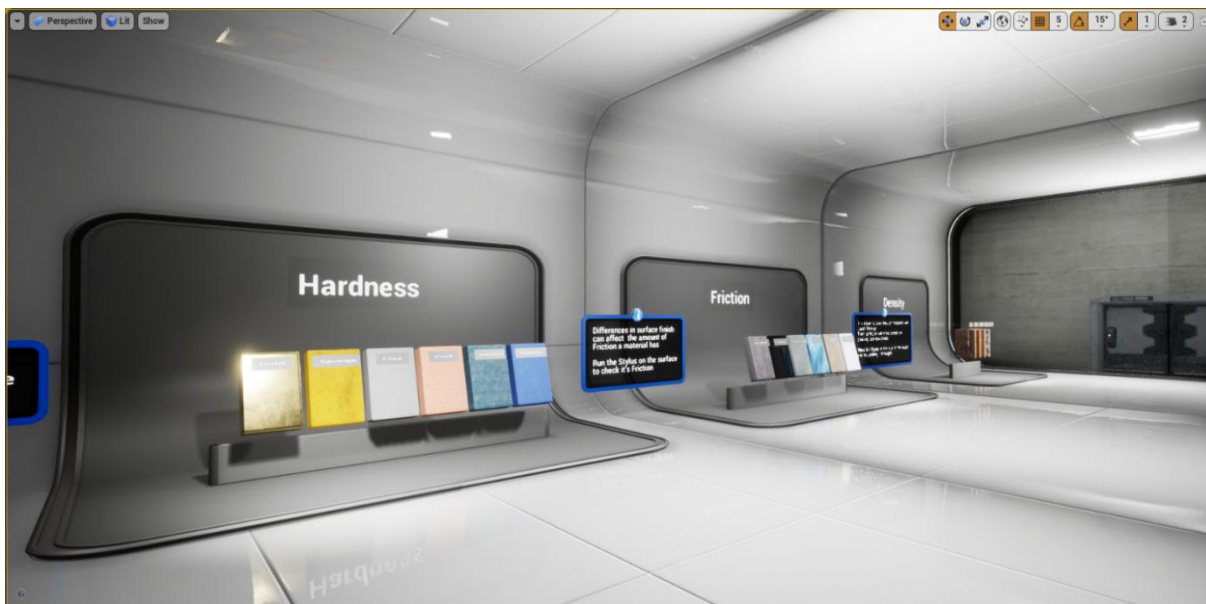
Εικόνα 3.40: Χαρακτήρας μέσα στη σκηνή

Ως Game Mode για όλα τα επίπεδα των πειραμάτων ορίστηκε το HapticGameMode, το οποίο έχει αναπτυχθεί από την εταιρεία 3D Systems. Εξαιρεση αποτελεί το επίπεδο του κεντρικού μενού, στο οποίο χρησιμοποιείται το προεπιλεγμένο (Default) Game Mode του Unreal Engine. Παράλληλα, ως Default Pawn Class επιλέχθηκε η νέα κλάση ThesisHapticDeviceActor που δημιουργήθηκε στο πλαίσιο της παρούσας εργασίας, ώστε να χρησιμοποιείται ως ο βασικός χαρακτήρας του παίκτη κατά την εκτέλεση των επιπέδων. Στην Εικόνα 3.41 φαίνεται ο χαρακτήρας από την οπτική του χρήστη κατά τη διάρκεια εκτέλεσης της εφαρμογής.

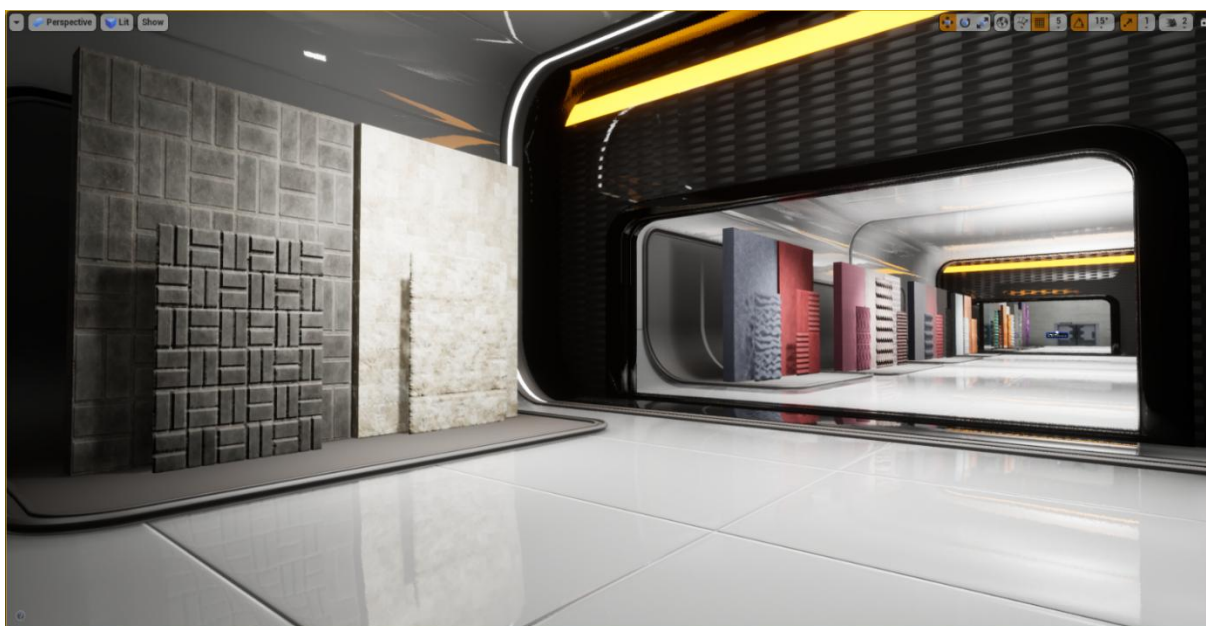


Εικόνα 3.41: Χαρακτήρας μέσα στην εφαρμογή

### 3.10 Οι ολοκληρωμένες αίθουσες των πειραμάτων



Εικόνα 3.42: Πειράματα βάση ιδιότητας των υλικών



Εικόνα 3.43: Πειράματα βάση υφής επιφανειών (Texture)



Εικόνα 3.44: Πειράματα αναγνώρισης αντικειμένων (Shapes)

### 3.11 Επίλογος

Με την ολοκλήρωση της ανάπτυξης των επιπέδων, των πειραματικών διατάξεων και των μηχανισμών αλληλεπίδρασης που παρουσιάστηκαν στις προηγούμενες ενότητες, διαμορφώθηκε το τελικό εικονικό περιβάλλον της εφαρμογής. Ο συνδυασμός των επιμέρους στοιχείων και λειτουργιών οδήγησε στη δημιουργία ενός ολοκληρωμένου και λειτουργικού χώρου, ο οποίος σχεδιάστηκε ώστε να υποστηρίξει με συνέπεια και σαφήνεια τη διεξαγωγή των πειραμάτων απτικής αλληλεπίδρασης [13], [14].

## Κεφάλαιο 4ο: Πακετάρισμα

### 4.1 Εισαγωγή

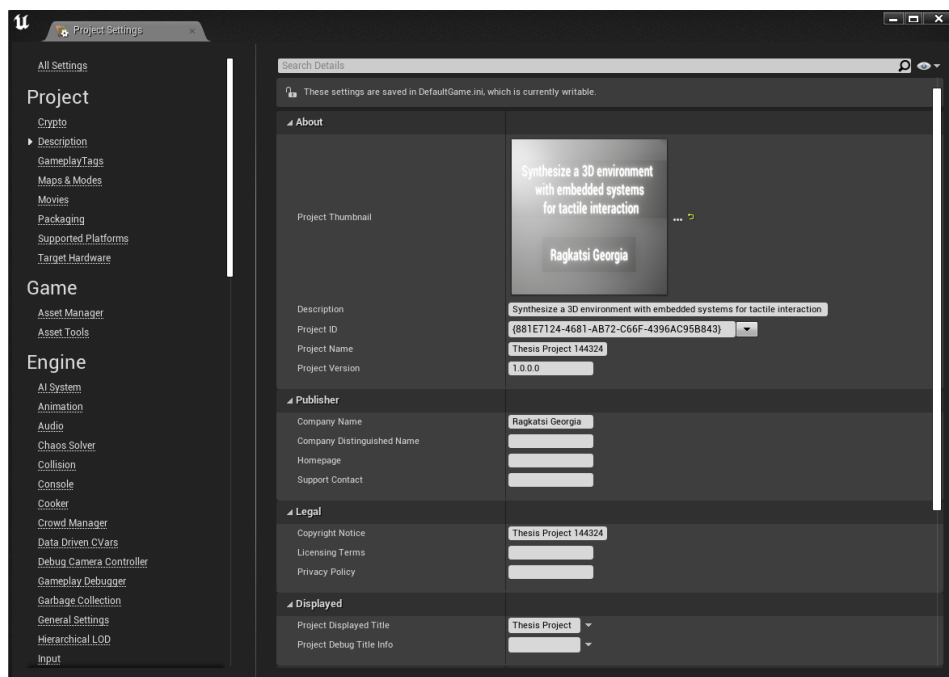
Μετά την ολοκλήρωση όλων των απαραίτητων αλλαγών και αφού διασφαλίστηκε ότι δεν απαιτούνται περαιτέρω διορθώσεις, το έργο θεωρείται έτοιμο για τη διαδικασία πακεταρίσματος και διανομής. Το πακετάρισμα αποτελεί το τελικό στάδιο της ανάπτυξης, κατά το οποίο η εφαρμογή μετατρέπεται σε εκτελέσιμη μορφή, κατάλληλη για χρήση εκτός του περιβάλλοντος ανάπτυξης της Unreal Engine [60]. Στην παρούσα ενότητα περιγράφονται τα βασικά βήματα και οι ρυθμίσεις που χρησιμοποιήθηκαν για την επιτυχή ολοκλήρωση της διαδικασίας.

### 4.2 Project Settings

Για την ορθή εκτέλεση της διαδικασίας πακεταρίσματος, απαιτείται η κατάλληλη παραμετροποίηση του έργου μέσω του παραθύρου Project Settings. Έτσι το έργο δεν θα διανεμηθεί με το σύνολο των αρχείων του, συμπεριλαμβανομένων και των test levels, αλλά θα περιέχει μόνο τα απαραίτητα αρχεία για να εκτελεστεί.

#### 4.2.1 Project – Description

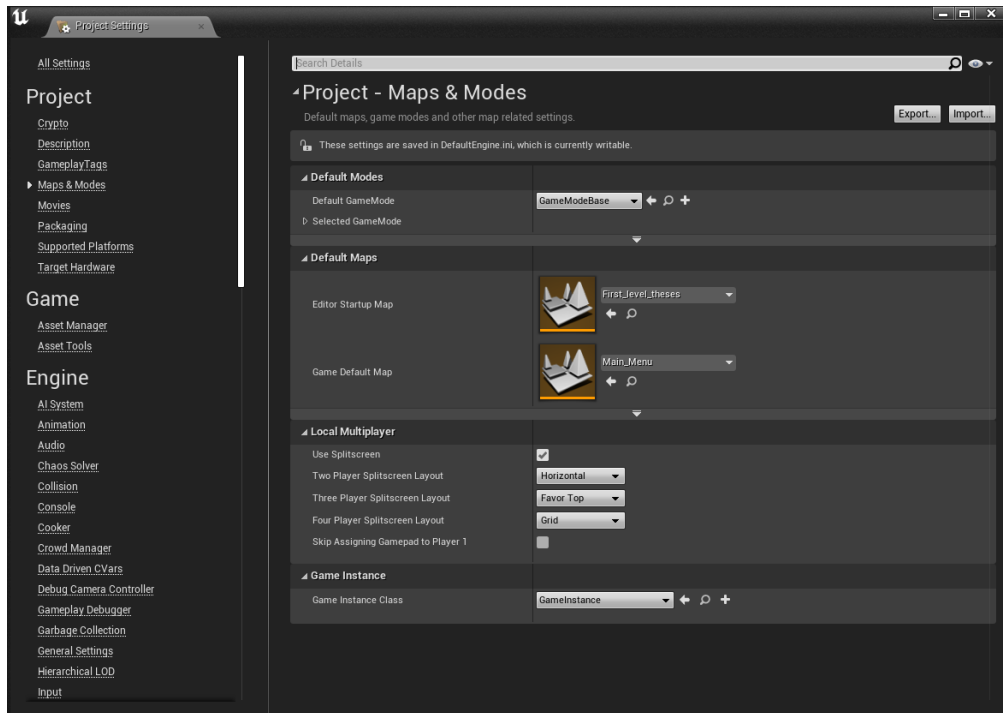
Στην καρτέλα Project – Description, όπως φαίνεται στην Εικόνα 4.1, δηλώνονται βασικές πληροφορίες που αφορούν την ταυτότητα της εφαρμογής [60]. Συγκεκριμένα, περιλαμβάνονται η περιγραφή της εφαρμογής, το thumbnail που συνοδεύει το εκτελέσιμο αρχείο, στοιχεία σχετικά με τον διανομέα, τα πνευματικά δικαιώματα (copyrights), καθώς και το όνομα εμφάνισης (display name) που θα φέρει το εκτελέσιμο αρχείο.



Εικόνα 4.1: Project – Description

### 4.2.2 Project – Maps & Modes

Στην καρτέλα Project – Maps & Modes, όπως φαίνεται στην Εικόνα 4.5, είναι απαραίτητο να οριστεί το Game Default Map, το οποίο καθορίζει ποιος χάρτης (Level) θα φορτώνεται κατά την εκκίνηση της εφαρμογής. Από προεπιλογή, η συγκεκριμένη ρύθμιση λαμβάνει την ίδια τιμή με το Editor Startup Map. Ωστόσο, σε εφαρμογές που περιλαμβάνουν περισσότερα από ένα επίπεδα, συνιστάται ο ρητός ορισμός του Game Default Map, ώστε να διασφαλίζεται η σαφής και ελεγχόμενη ροή εκτέλεσης της εφαρμογής στο αρχικό στάδιο φόρτωσης [36].



Εικόνα 4.2: Project – Maps & Modes

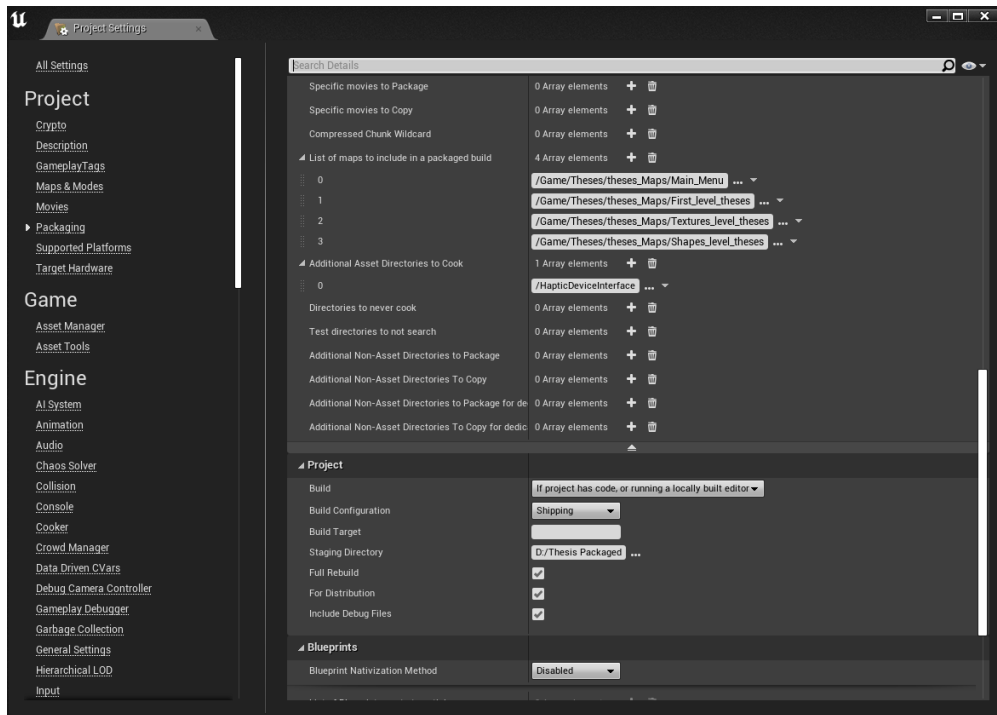
Για την παρούσα εργασία, ως αρχικός χάρτης (Game Default Map) ορίστηκε το επίπεδο Main\_Menu, το οποίο περιλαμβάνει το κεντρικό μενού της εφαρμογής και αποτελεί το σημείο εκκίνησης της αλληλεπίδρασης του χρήστη με το σύστημα.

### 4.2.3 Project – Packaging

Στην καρτέλα Project – Packaging, όπως φαίνεται στην Εικόνα 4.3, πραγματοποιούνται οι απαραίτητες ρυθμίσεις που αφορούν τον τρόπο πακεταρίσματος των αρχείων του έργου. Συγκεκριμένα, καθορίζονται η μέθοδος συμπίεσης των δεδομένων, η μορφή με την οποία θα οργανωθούν τα αρχεία, καθώς και οι παράμετροι που σχετίζονται με τη διαδικασία δημιουργίας (build) του τελικού εκτελέσιμου αρχείου της εφαρμογής [60].

Κατά την ανάπτυξη ενός έργου δημιουργούνται συχνά επίπεδα που χρησιμοποιούνται αποκλειστικά για πειραματισμό και αποσφαλμάτωση των λειτουργιών της εφαρμογής, πέραν των επιπέδων που προορίζονται για την τελική εμπειρία του χρήστη. Τα επίπεδα αυτά εξυπηρετούν την υποστήριξη της ανάπτυξης σε απλουστευμένα περιβάλλοντα και δεν προορίζονται να συμπεριληφθούν στο τελικό πακέτο της εφαρμογής. Για τον λόγο αυτό, στην καρτέλα Project – Packaging είναι σημαντικό να

δηλωθούν ρητά τα επίπεδα που θα συμπεριληφθούν στο τελικό build, μέσω της λίστας List of maps to include in a packaged build.



Εικόνα 4.3: Project – Packaging

Για τις ανάγκες της παρούσας εργασίας, οι χάρτες που περιλαμβάνονται στο τελικό build είναι οι εξής:

- Main\_Menu, το οποίο περιλαμβάνει το κεντρικό μενού της εφαρμογής,
- First\_level\_thesis, το οποίο φιλοξενεί πειράματα βασισμένα στις ιδιότητες των υλικών,
- Textures\_level\_thesis, το οποίο περιλαμβάνει πειράματα που σχετίζονται με την απτική αναγνώριση υφών επιφανειών (*textures*),
- Shapes\_level\_thesis, το οποίο περιλαμβάνει πειράματα αναγνώρισης αντικειμένων (*shapes*) μέσω ψηλάφησης.

Η επιλογή αυτή διασφαλίζει ότι το τελικό πακέτο της εφαρμογής περιλαμβάνει μόνο τα απαραίτητα επίπεδα που σχετίζονται άμεσα με τη λειτουργία και τους στόχους της εφαρμογής.

Επιπλέον, τα πρόσθετα (Plugins) που έχουν ενσωματωθεί στη μηχανή και δεν βρίσκονται εντός του φακέλου Content του έργου, αλλά είναι απαραίτητα για τη λειτουργία της εφαρμογής, πρέπει να δηλωθούν στη λίστα Additional Asset Directories to Cook, ώστε να συμπεριληφθούν σωστά κατά τη διαδικασία του πακεταρίσματος. Στην παρούσα εργασία, το πρόσθετο HapticDeviceInterface της εταιρείας 3D Systems, το οποίο ενσωματώνεται στον πυρήνα της μηχανής Unreal Engine, δηλώθηκε στη συγκεκριμένη λίστα για να διασφαλιστεί η ορθή λειτουργία της απτικής συσκευής στο τελικό εκτελέσιμο.

#### 4.2.4 Project – Packaging: Build Configuration

Στο πλαίσιο της διαδικασίας πακεταρίσματος, ιδιαίτερη σημασία έχει η επιλογή του Build Configuration, η οποία καθορίζει τον τρόπο με τον οποίο θα δημιουργηθεί το τελικό εκτελέσιμο αρχείο της εφαρμογής. Η Unreal Engine παρέχει διαφορετικές επιλογές διαμόρφωσης, όπως Debug, Development και Shipping, καθεμία από τις οποίες εξυπηρετεί διαφορετικό στάδιο της ανάπτυξης.

Για την παρούσα εργασία επιλέχθηκε η διαμόρφωση Shipping, καθώς προορίζεται για την τελική έκδοση της εφαρμογής που θα διανεμηθεί και θα εκτελεστεί εκτός του περιβάλλοντος ανάπτυξης. Η συγκεκριμένη επιλογή απενεργοποιεί τις λειτουργίες αποσφαλμάτωσης, αφαιρεί περιττές πληροφορίες και βελτιστοποιεί την απόδοση της εφαρμογής, εξασφαλίζοντας μικρότερο μέγεθος εκτελέσιμου αρχείου και πιο σταθερή λειτουργία.

#### 4.2.5 Project – Supported Platforms & Target Hardware

Τέλος, πραγματοποιείται έλεγχος στις καρτέλες Project – Supported Platforms και Project – Target Hardware, προκειμένου να διασφαλιστεί ότι η πλατφόρμα για την οποία προορίζεται το πακέταρισμα της εφαρμογής υποστηρίζεται από τη μηχανή Unreal Engine και ότι η εφαρμογή είναι κατάλληλα παραμετροποιημένη (optimized) για το υλικό (hardware) στο οποίο πρόκειται να διανεμηθεί.

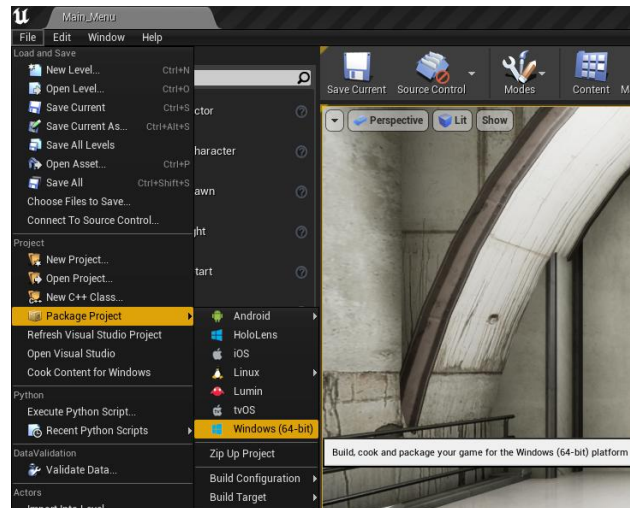
#### 4.2.6 Εκτέλεση της διαδικασίας πακεταρίσματος

Πριν από την εκτέλεση της διαδικασίας πακεταρίσματος, είναι σημαντικό να ληφθεί υπόψη ότι το τελικό πακεταρισμένο εκτελέσιμο αρχείο δεν περιλαμβάνει στοιχεία του πυρήνα της μηχανής Unreal Engine. Κατά συνέπεια, πρόσθετα τα οποία είναι ενσωματωμένα στον πυρήνα της μηχανής και δεν βρίσκονται εντός του καταλόγου του έργου δεν συμπεριλαμβάνονται αυτόματα στο τελικό αποτέλεσμα της διαδικασίας packaging. Στην παρούσα εργασία, το πρόσθετο HapticDeviceInterface της εταιρείας 3D Systems αποτελεί τέτοια περίπτωση, γεγονός που έχει ως αποτέλεσμα, κατά την εκτέλεση της εφαρμογής εκτός του περιβάλλοντος του Editor, να μην είναι δυνατή η φόρτωση του απτικού χαρακτήρα λόγω απουσίας των απαραίτητων αρχείων [61].

Για την αποφυγή του παραπάνω προβλήματος, πριν από την εκτέλεση της διαδικασίας πακεταρίσματος δημιουργήθηκε εντός του έργου ο κατάλογος Plugins, ο οποίος απαιτείται να φέρει επακριβώς τη συγκεκριμένη ονομασία ώστε να αναγνωρίζεται σωστά από τη μηχανή Unreal Engine. Στον κατάλογο αυτό τοποθετήθηκε αντίγραφο του πρόσθετου της 3D Systems, διασφαλίζοντας ότι τα απαραίτητα αρχεία θα συμπεριληφθούν στο τελικό πακεταρισμένο αποτέλεσμα και θα είναι διαθέσιμα κατά την εκτέλεση της εφαρμογής εκτός του περιβάλλοντος ανάπτυξης [61].

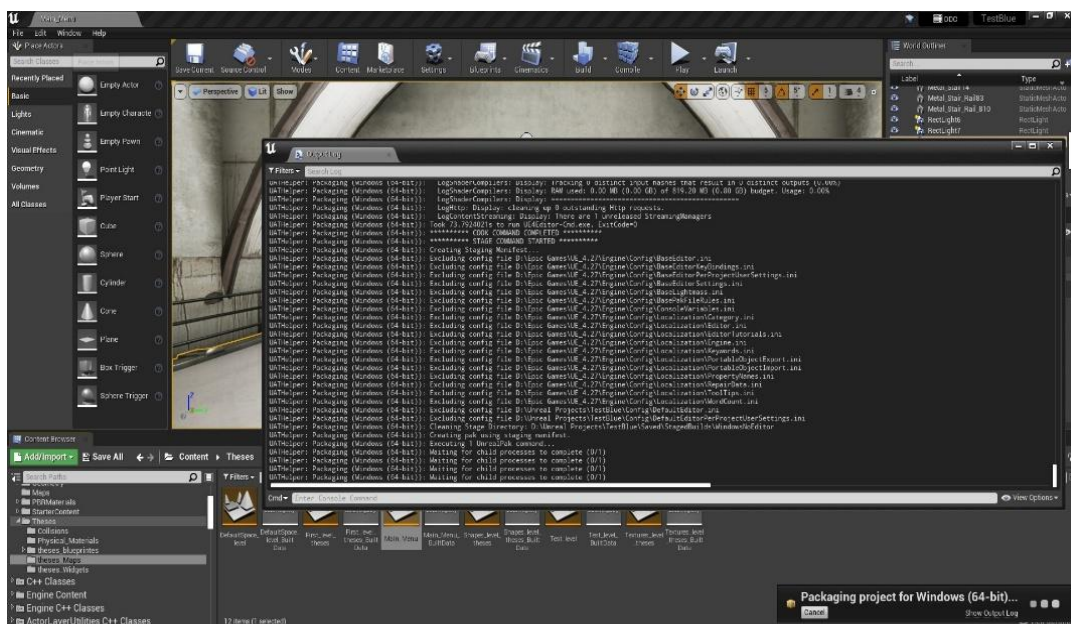
Η παραπάνω διαδικασία δεν απαιτείται για τα πακέτα περιεχομένου που προστέθηκαν στο έργο, καθώς αυτά έχουν ενσωματωθεί απευθείας στον κατάλογο Content και συμπεριλαμβάνονται αυτόματα στο τελικό πακεταρισμένο αποτέλεσμα.

Για την εκτέλεση της διαδικασίας πακεταρίσματος, από το μενού File επιλέγεται η επιλογή Package Project και στη συνέχεια καθορίζεται η πλατφόρμα για την οποία θα πραγματοποιηθεί το πακέταρισμα, όπως φαίνεται στην Εικόνα 4.4. Στη παρούσα εργασία, η επιλεγμένη πλατφόρμα είναι τα Windows.



Εικόνα 4.4: Package Project

Κατά τη διάρκεια της διαδικασίας πακεταρίσματος εμφανίζεται ένα παράθυρο διαλόγου, όπως φαίνεται στην Εικόνα 4.5, στο οποίο καταγράφονται αναλυτικά όλα τα στάδια και οι λειτουργίες που εκτελούνται. Για το συγκεκριμένο μέγεθος έργου, η συνολική διάρκεια της διαδικασίας ήταν περίπου δεκαπέντε λεπτά.



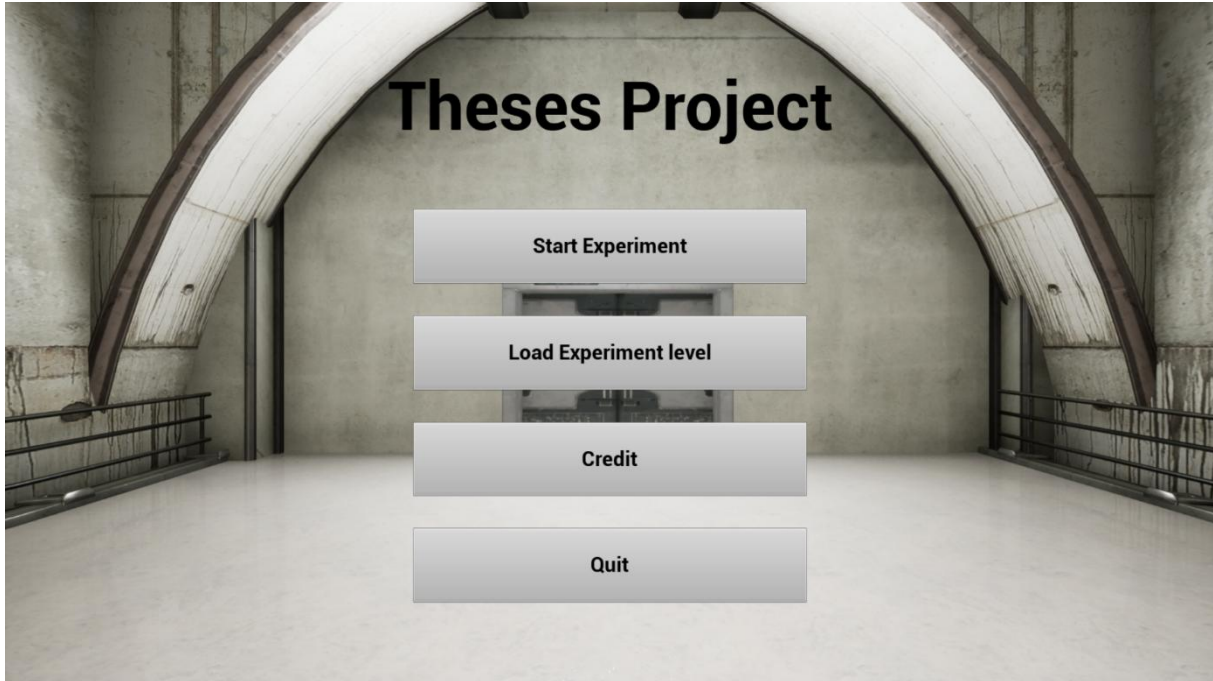
Εικόνα 4.5: Packaging Process

### 4.3 Επίλογος

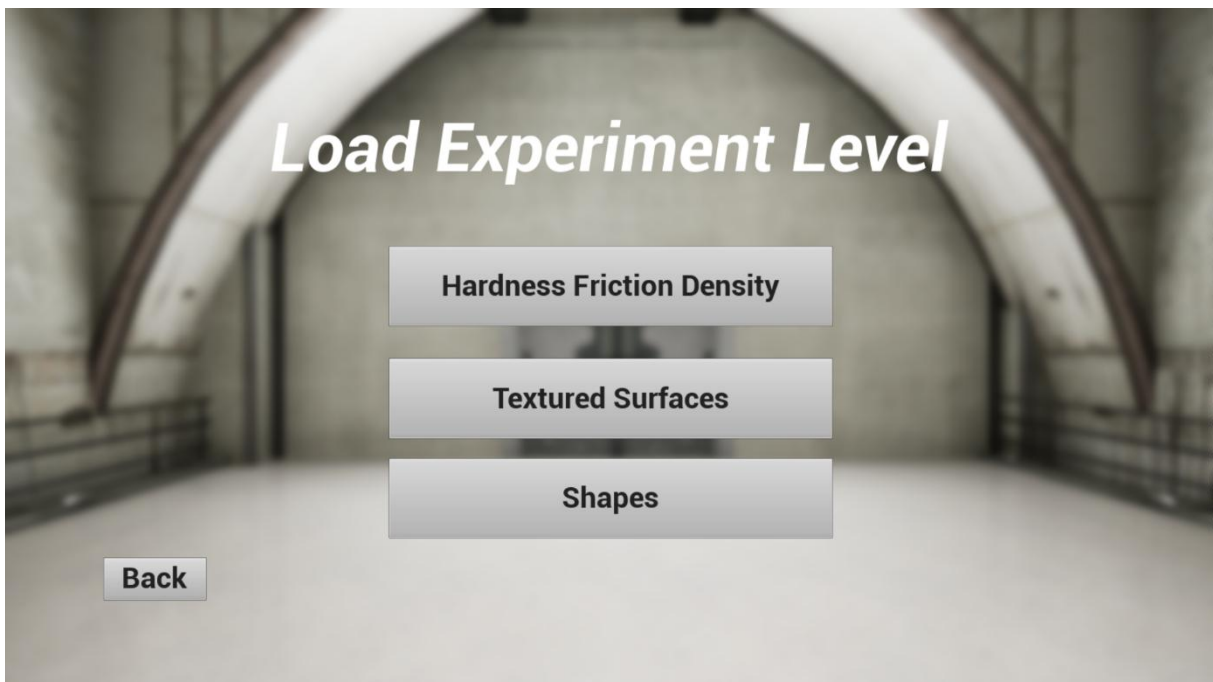
Η ολοκλήρωση της διαδικασίας αυτής σηματοδοτεί τη μετάβαση από το στάδιο της ανάπτυξης στο στάδιο της τελικής διάθεσης της εφαρμογής, καθιστώντας το έργο έτοιμο για εκτέλεση εκτός του περιβάλλοντος ανάπτυξης της Unreal Engine

## Κεφάλαιο 5ο: Ροή της εφαρμογής

Με την ολοκλήρωση της διαδικασίας Packaging, η εφαρμογή είναι έτοιμη προς εκτέλεση. Κατά την εκκίνηση, ο χρήστης εισέρχεται στο κεντρικό μενού, όπως φαίνεται στην Εικόνα 5.1, από όπου μπορεί είτε να ξεκινήσει την εξερεύνηση του πειραματικού χώρου από την αρχή είτε να μεταβεί σε υπομενού που περιλαμβάνει όλα τα διαθέσιμα πειράματα και να επιλέξει κάποιο συγκεκριμένο, όπως φαίνεται στην Εικόνα 5.2.



Εικόνα 5.1: Κεντρικό Μενού Εφαρμογής



Εικόνα 5.2: Φόρτωση πειράματος

Κάθε φορά, ο χρήστης εισέρχεται στην αρχή του αντίστοιχου πειραματικού χώρου. Σε οποιαδήποτε στιγμή ο χρήστης μπορεί να επιστρέψει στο κεντρικό μενού περνώντας μέσα από την αρχική πόρτα του επιπέδου, όπως φαίνεται στην Εικόνα 5.3.



Εικόνα 5.3: Έξοδος στο κεντρικό μενού

## 5.1 Πειράματα βάση ιδιότητας των υλικών

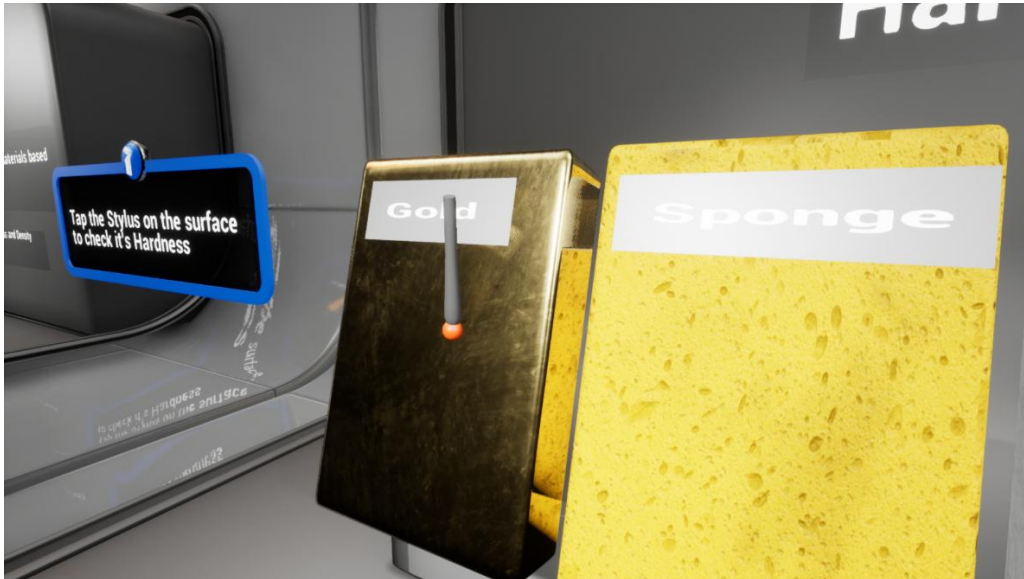
Στο πρώτο επίπεδο, η αρχική αίθουσα περιλαμβάνει βασικές οδηγίες πλοήγησης καθώς και σύντομες πληροφορίες για τον μηχανικό βραχίονα, όπως φαίνεται στην Εικόνα 5.4. Συνεχίζοντας την πλοήγηση, ο χρήστης οδηγείται στην πρώτη αίθουσα πειραμάτων, η οποία εστιάζει στις φυσικές ιδιότητες των υλικών. Τα πειράματα είναι οργανωμένα έτσι ώστε να δίνεται έμφαση στη σκληρότητα, την ολισθηρότητα και τη διαπερατότητα των υλικών.



Εικόνα 5.4: Πειραματικός χώρος - Ιδιότητες Υλικών

### 5.1.1 Πείραμα Σκληρότητας

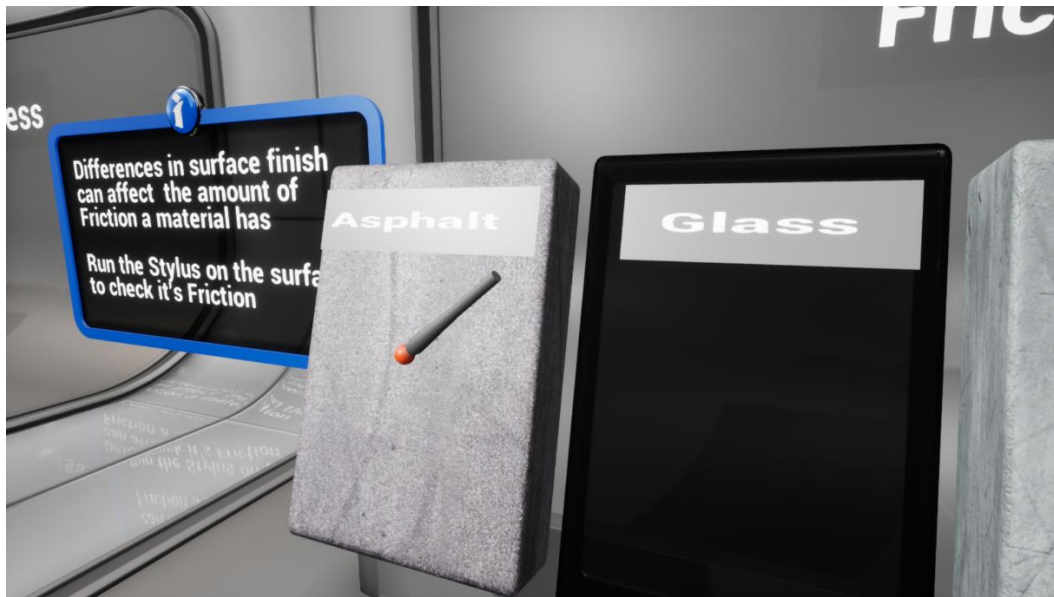
Πλησιάζοντας στο πείραμα σκληρότητας, ο χρήστης μπορεί να χτυπήσει ελαφρά με την πένα (*stylus*) τις επιφάνειες των αντικειμένων, ώστε να αντιληφθεί τη διαφορά μεταξύ σκληρών και μαλακών υλικών, όπως φαίνεται στην Εικόνα 5.5. Ενδεικτικά, τα αντικείμενα από χρυσό και σφουγγάρι παρουσιάζουν έντονη αντίθεση ως προς τη σκληρότητα. Η απτική ανάδραση που παρέχεται μέσω του μηχανικού βραχίονα αντικατοπτρίζει αυτή τη διαφορά. Συνολικά, τοποθετήθηκαν έξι αντικείμενα με διαφορετικά επίπεδα σκληρότητας τα οποία επιτρέπουν στον χρήστη να εξερευνήσει τη διαφορά μεταξύ τους.



Εικόνα 5.5: Σκληρότητα - Αντικείμενα από χρυσό και σφουγγάρι

### 5.1.2 Πείραμα Ολισθηρότητας

Προχωρώντας περνάμε στο πείραμα ολισθηρότητας, όπου ο χρήστης τρίβοντας την πένα πάνω στις επιφάνειες των αντικειμένων προσπαθεί να καταλάβει ποια από αυτές είναι περισσότερο τραχιά ή ολισθηρή, όπως φαίνεται στην Εικόνα 5.6. Ενδεικτικά, τα αντικείμενα από άσφαλτο και πάγο αποτελούν τις ακραίες περιπτώσεις, με την άσφαλτο να παρουσιάζει τη μεγαλύτερη τραχύτητα και τον πάγο τη μικρότερη αντίσταση. Σε αυτό το πείραμα έξι νέα αντικείμενα, με διαφορετικούς βαθμούς ολίσθησης, μέσω της απτικής ανάδρασης δίνουν την δυνατότητα στο χρήστη να καταλάβει διαφορετικά επίπεδα ολισθηρότητας.



Εικόνα 5.6: Ολισθηρότητα – Αντικείμενα από άσφαλτο και γυαλί

### 5.1.3 Πείραμα Διαπερατότητας

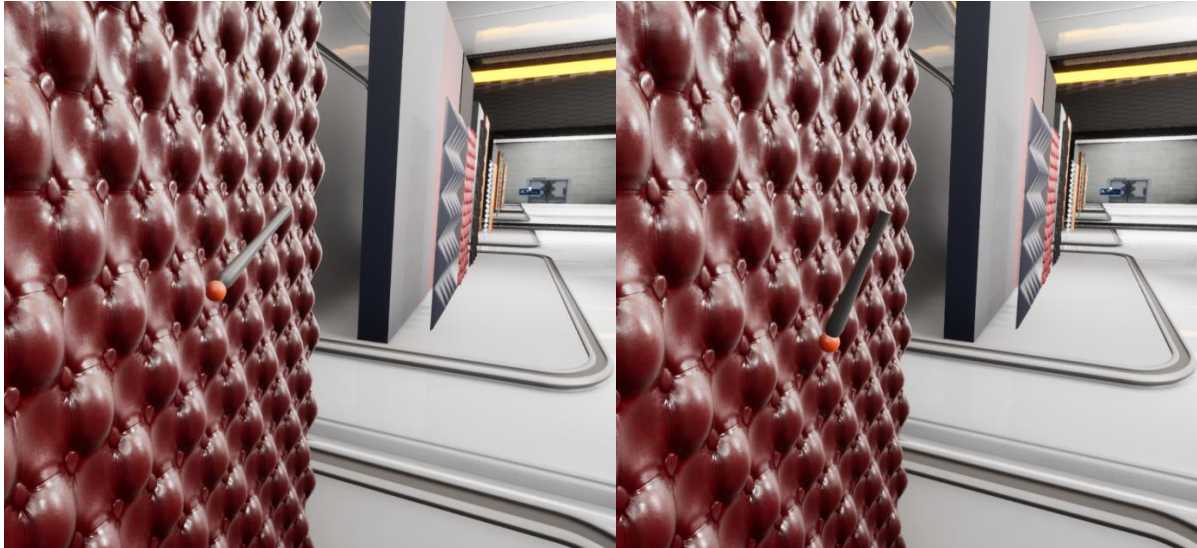
Το τρίτο πείραμα της αίθουσας αφορά τη διαπερατότητα. Ο χρήστης προσπαθεί να περάσει την πένα μέσα από διατεταγμένα πλακίδια με διαφορετική πυκνότητα, όπως φαίνεται στην Εικόνα 5.7. Η αντίσταση αυξάνεται σταδιακά, από τον αέρα, που δεν παρουσιάζει καμία αντίσταση, έως τον βράχο, που παρουσιάζει τη μέγιστη. Αν και στην πραγματικότητα ο βράχος θα ήταν αδιαπέραστος, στο εικονικό περιβάλλον η διέλευση είναι εφικτή, προκειμένου να αποφευχθεί η καταπόνηση του μηχανικού βραχίονα. Με την ολοκλήρωση των πειραμάτων αυτής της αίθουσας, ο χρήστης μπορεί να προχωρήσει στον επόμενο πειραματικό χώρο.



Εικόνα 5.7: Αντικείμενα με διαφορετική πυκνότητα

## 5.2 Πειράματα βάση υφής επιφανειών (Texture)

Ο δεύτερος πειραματικός χώρος εστιάζει στις υφές. Περιλαμβάνει συνολικά δεκαεπτά εκθέματα με διαφορετικές επιφανειακές μορφοποιήσεις το κάθε ένα από τα οποία προσφέρει μια διακριτή εμπειρία αφής. Ο χρήστης μπορεί να κυλίσει απαλά την πένα πάνω στις επιφάνειες, όπως φαίνεται στην Εικόνα 5.8, ενώ ο μηχανικός βραχίονας ακολουθεί την μορφοποίηση της κάθε επιφάνειας επιτρέποντας στην πένα να εισχωρεί σε κοιλότητες και αυλάκια όπου αυτά υπάρχουν και να απωθείται όταν βρίσκεται εξογκώματα και προεξοχές.



Εικόνα 5.8: Πείραμα αντίληψης της υφής

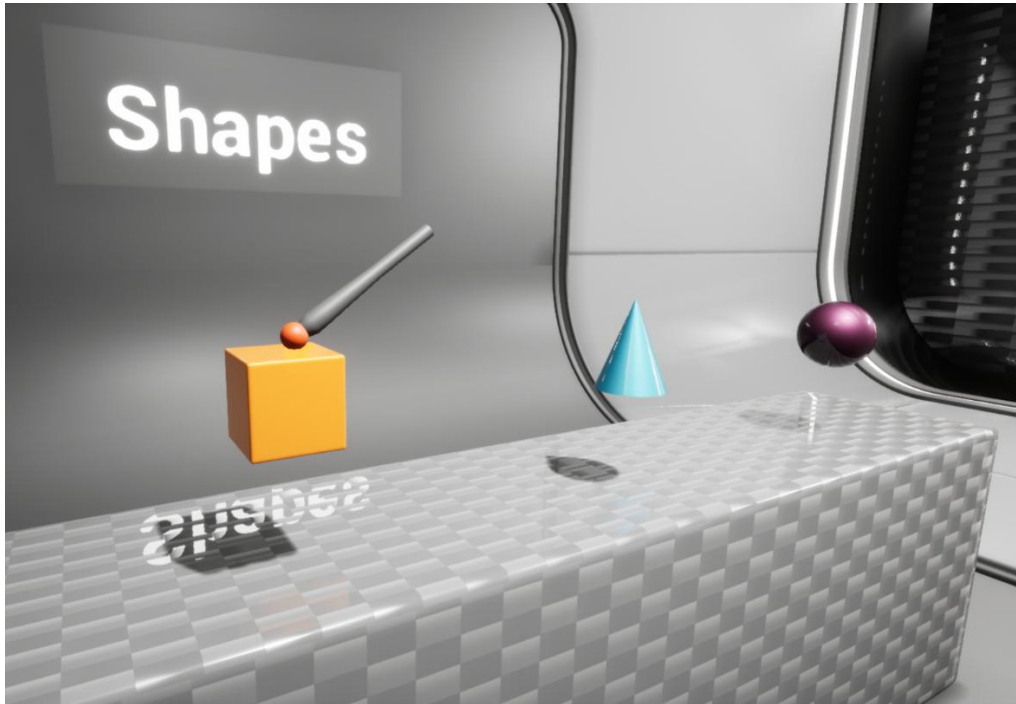
Στον χώρο αυτό, τα υλικά που χρησιμοποιήθηκαν δεν προσδίδουν συγκεκριμένες φυσικές ιδιότητες στα αντικείμενα του χώρου. Σε αντίθεση με τα αντικείμενα του πρώτου πειραματικού χώρου όπου το υλικό των αντικειμένων είναι αυτό που τους δίνει την κάθε ιδιότητα, σε αυτή την αίθουσα η διαφοροποίηση προκύπτει από τη γεωμετρία των αντικειμένων, όπως φαίνεται στην Εικόνα 5.9. Τα υλικά απλά ενισχύουν την αίσθηση της οπτικής αφής.



Εικόνα 5.9: Διαφορά στη γεωμετρία των αντικειμένων

### 5.3 Πειράματα αναγνώρισης αντικειμένων (Shapes)

Ο τρίτος και τελευταίος πειραματικός χώρος επικεντρώνεται στην αναγνώριση σχημάτων. Αρχικά, ο χρήστης έχει τη δυνατότητα να αλληλεπιδράσει με ενδεικτικά σχήματα, ώστε να εξοικειωθεί με το περιεχόμενο των πειραμάτων, όπως φαίνεται στην Εικόνα 5.10. Στη συνέχεια, μπορεί να εισέλθει σε μία από τις επτά σκοτεινές αίθουσες και να προσπαθήσει να αναγνωρίσει το αντικείμενο που βρίσκεται σε καθεμία, όπως φαίνεται στην Εικόνα 5.11.

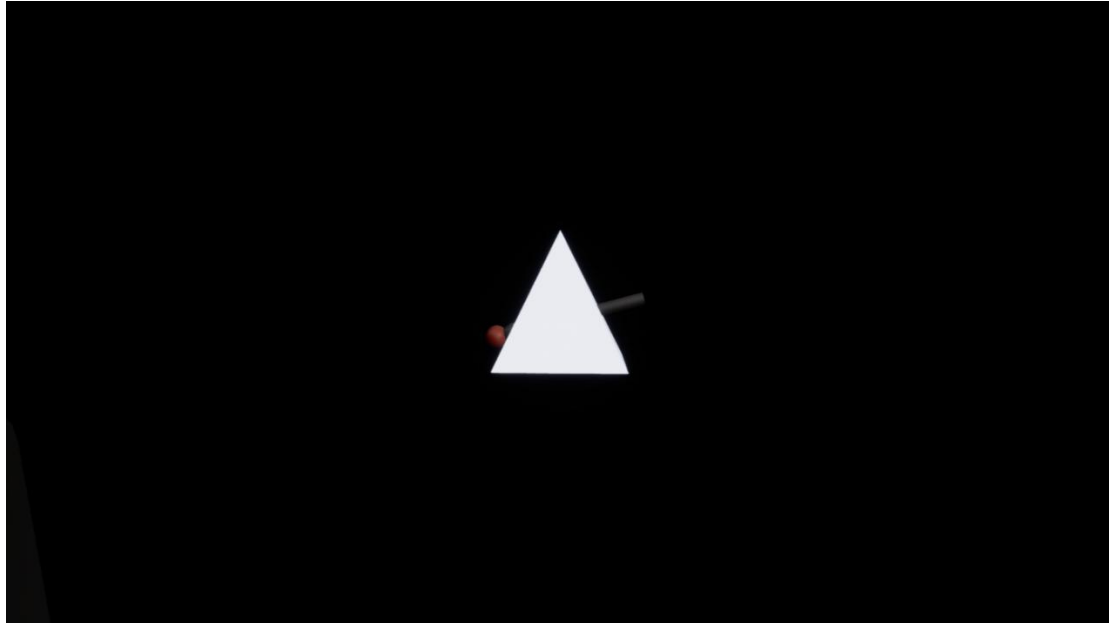


Εικόνα 5.10: Ενδεικτικά σχήματα



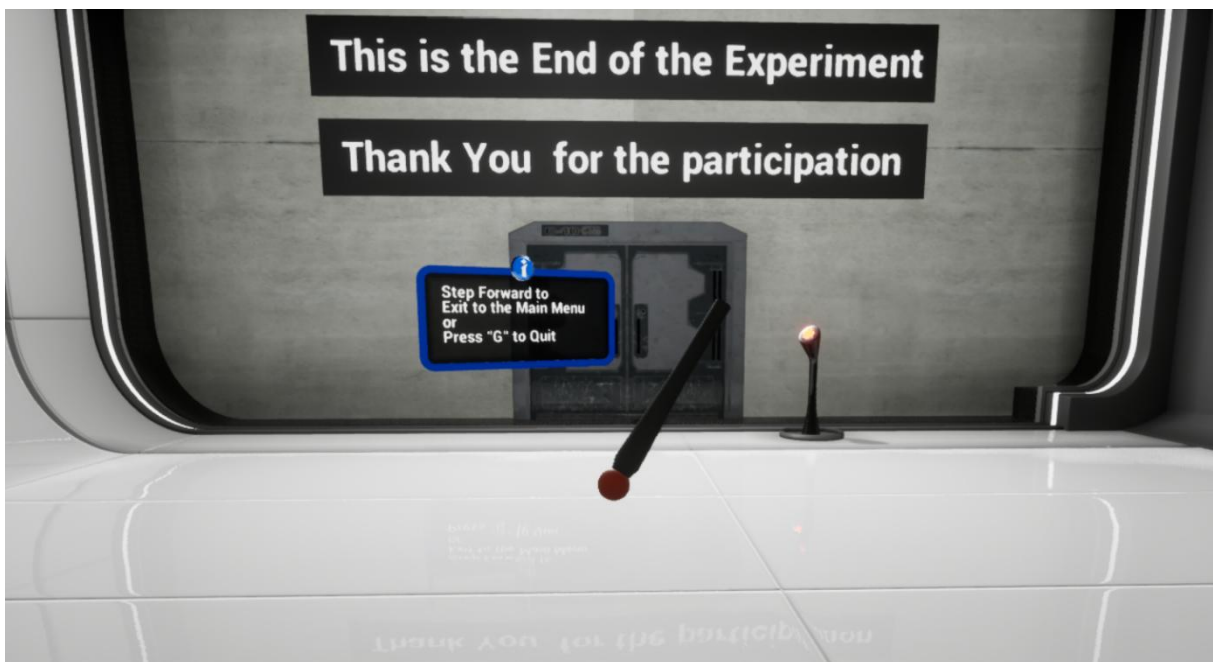
Εικόνα 5.11: Αίθουσα πειράματος

Για την επαλήθευση της απάντησής του ή σε περίπτωση δυσκολίας, ο χρήστης μπορεί να ενεργοποιήσει τον φωτισμό του αντικειμένου μέσω πληκτρολογίου, καθώς και να τον απενεργοποιήσει, εφόσον επιθυμεί να επαναλάβει το πείραμα, όπως φαίνεται στην Εικόνα 5.12.



Εικόνα 5.12: Φωτεινό αντικείμενο

Το πείραμα τελειώνει όταν ο χρήστης φτάσει στο τέλος του τρίτου πειραματικού χώρου όπου έχει την επιλογή να τερματίσει την εφαρμογή ή να επιστρέψει στο κεντρικό μενού, όπως φαίνεται στην Εικόνα 5.13.



Εικόνα 5.13: Ολοκλήρωση των πειραμάτων

## Κεφάλαιο 6ο: Αξιολόγηση Εφαρμογής

### 6.1 Αξιολόγηση Εφαρμογής με Πραγματικούς χρήστες

Η αξιολόγηση της εφαρμογής πραγματοποιήθηκε με τη συμμετοχή 20 ατόμων, με στόχο τον έλεγχο της λειτουργικότητας, της χρηστικότητας και της συνολικής εμπειρίας απτικής αλληλεπίδρασης που προσφέρει το σύστημα. Η διαδικασία αξιολόγησης είχε περιγραφικό χαρακτήρα και βασίστηκε στην παρατήρηση της αλληλεπίδρασης των χρηστών με την εφαρμογή και στην αποτύπωση των εντυπώσεών τους. Η συγκεκριμένη διαδικασία κρίνεται ιδιαίτερα σημαντική, καθώς η αποτελεσματικότητα μιας απτικής εφαρμογής δεν μπορεί να αποτιμηθεί αποκλειστικά σε θεωρητικό ή τεχνικό επίπεδο, αλλά απαιτεί άμεση δοκιμή από τελικούς χρήστες.

Κατά τη διάρκεια της αξιολόγησης, οι χρήστες ήρθαν σε επαφή με τα διαφορετικά πειραματικά επίπεδα της εφαρμογής και αλληλεπίδρασαν με τα απτικά αντικείμενα, εκτελώντας τις προβλεπόμενες λειτουργίες. Μέσω της διαδικασίας αυτής κατέστη δυνατή η παρατήρηση του τρόπου με τον οποίο οι χρήστες αντιλαμβάνονται την απτική ανάδραση, πλοηγούνται στο εικονικό περιβάλλον και ανταποκρίνονται στα ερεθίσματα που παρέχει το σύστημα.

#### 6.1.1 Δημογραφικά στοιχεία δείγματος

Στο δείγμα συμμετείχαν άτομα ηλικίας 25 έως 60 ετών, με μεγαλύτερη συγκέντρωση στις νεότερες ηλικίες, καθώς 15 συμμετέχοντες ανήκαν στο ηλικιακό εύρος 25–30 ετών, ενώ οι υπόλοιποι 5 βρίσκονταν μεταξύ 50–60 ετών. Δεν εφαρμόστηκαν περιοριστικά κριτήρια συμμετοχής, ενώ η πλειονότητα των συμμετεχόντων δεν είχε προηγούμενη εμπειρία με απτικές συσκευές παρόμοιες με τον μηχανικό βραχίονα που χρησιμοποιήθηκε στην εφαρμογή.

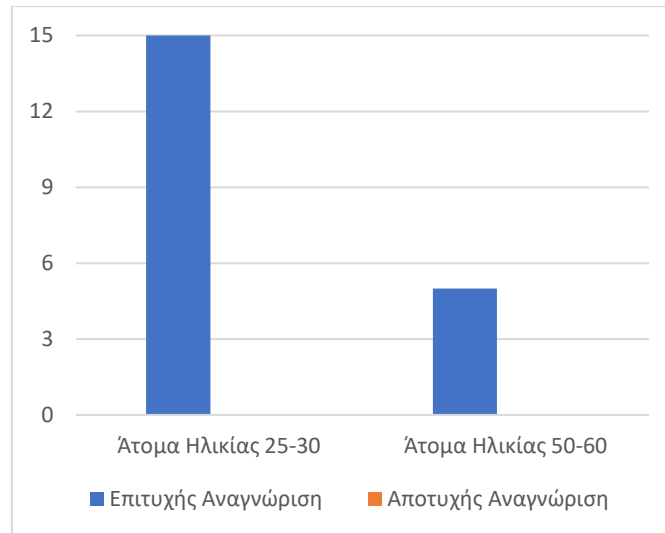
### 6.2 Πλοήγηση στο Εικονικό Περιβάλλον

Η πλοήγηση στον εικονικό χώρο δεν αποτέλεσε ιδιαίτερη πρόκληση για τους συμμετέχοντες. Παρατηρήθηκε γενικά υψηλός βαθμός οικειότητας με το σύστημα, γεγονός που αποδίδεται στην προηγούμενη εμπειρία των χρηστών με εφαρμογές υπολογιστή και βιντεοπαιχνίδια. Η κίνηση και ο έλεγχος της κάμερας κρίθηκαν εύχρηστα και κατανοητά από το σύνολο των συμμετεχόντων.

### 6.3 Ανάλυση Αξιολόγησης ανά Πείραμα

#### 6.3.1 Αίσθηση Σκληρότητας (Hardness)

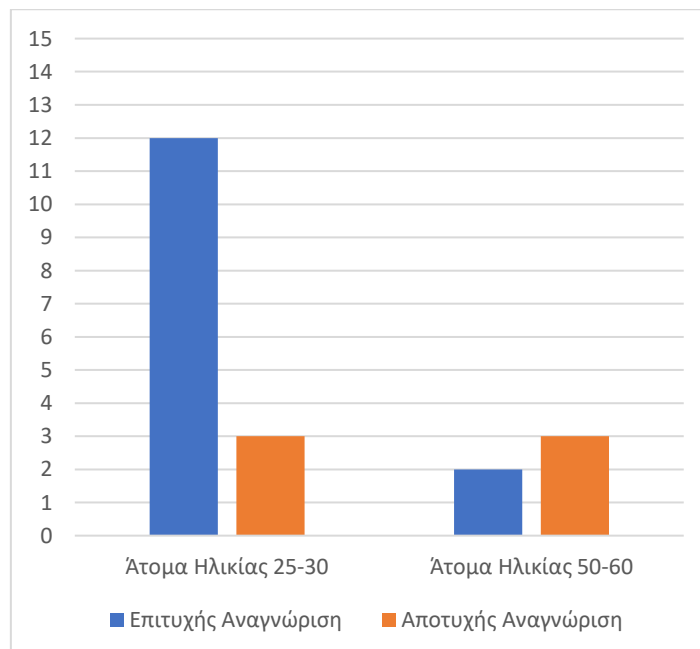
Το πείραμα σκληρότητας χαρακτηρίστηκε ως πλήρως επιτυχημένο, όπως φαίνεται στο Σχήμα 6.1, καθώς όλοι οι συμμετέχοντες κατάφεραν να διακρίνουν με ακρίβεια τα διαφορετικά επίπεδα σκληρότητας και να ταξινομήσουν σωστά τα αντικείμενα από το πιο σκληρό και άκαμπτο στο πιο μαλακό και ελαστικό.



Σχήμα 6.1: Αποτελέσματα πειράματος Σκληρότητας

### 6.3.2 Αίσθηση Ολισθηρότητας (Friction)

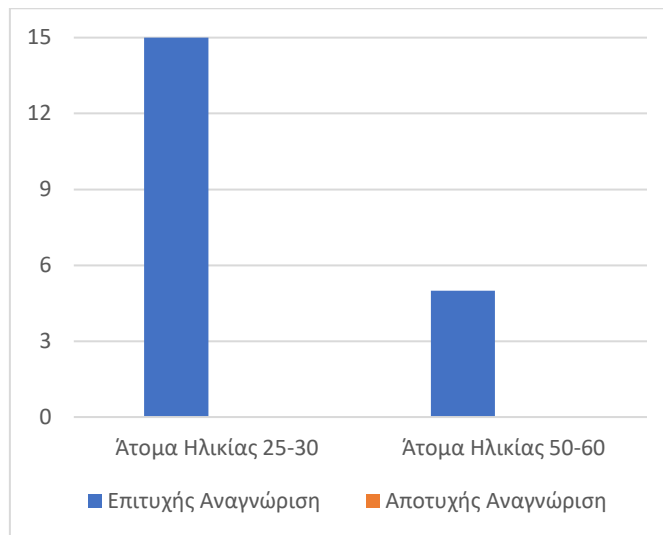
Στο πείραμα ολισθηρότητας παρατηρήθηκαν διαφοροποιήσεις στις επιδόσεις των συμμετεχόντων, όπως φαίνεται στο Σχήμα 6.2. Συγκεκριμένα, 12 από τους 15 νεότερους συμμετέχοντες και 2 από τους 5 μεγαλύτερους κατάφεραν να αναγνωρίσουν σωστά όλα τα αντικείμενα και να τα ταξινομήσουν από το πιο τραχύ στο πιο λείο. Οι υπόλοιποι συμμετέχοντες δυσκολεύτηκαν κυρίως στη διάκριση μεταξύ των δύο πιο τραχιών επιφανειών, ενώ σε ορισμένες περιπτώσεις παρατηρήθηκε δυσκολία και στον διαχωρισμό ενδιάμεσων επιπέδων ολίσθησης.



Σχήμα 6.2: Αποτελέσματα πειράματος Ολίσθησης

### 6.3.3 Αίσθηση Διαπερατότητας (Density)

Το πείραμα διαπερατότητας αποτέλεσε επίσης πλήρη επιτυχία, όπως φαίνεται στο Σχήμα 6.3, καθώς όλοι οι συμμετέχοντες κατάφεραν να αναγνωρίσουν με ακρίβεια τα διαφορετικά επίπεδα πυκνότητας των αντικειμένων. Οι χρήστες μπόρεσαν να κατανοήσουν πόσο εύκολα ή δύσκολα η πένα του μηχανικού βραχίονα διαπέρασε το κάθε αντικείμενο και να τα ταξινομήσουν σωστά από το λιγότερο στο περισσότερο πυκνό.



Σχήμα 6.3: Αποτελέσματα πειράματος Διαπερατότητας

### 6.3.4 Αίσθηση Υφής Επιφανειών (Texture)

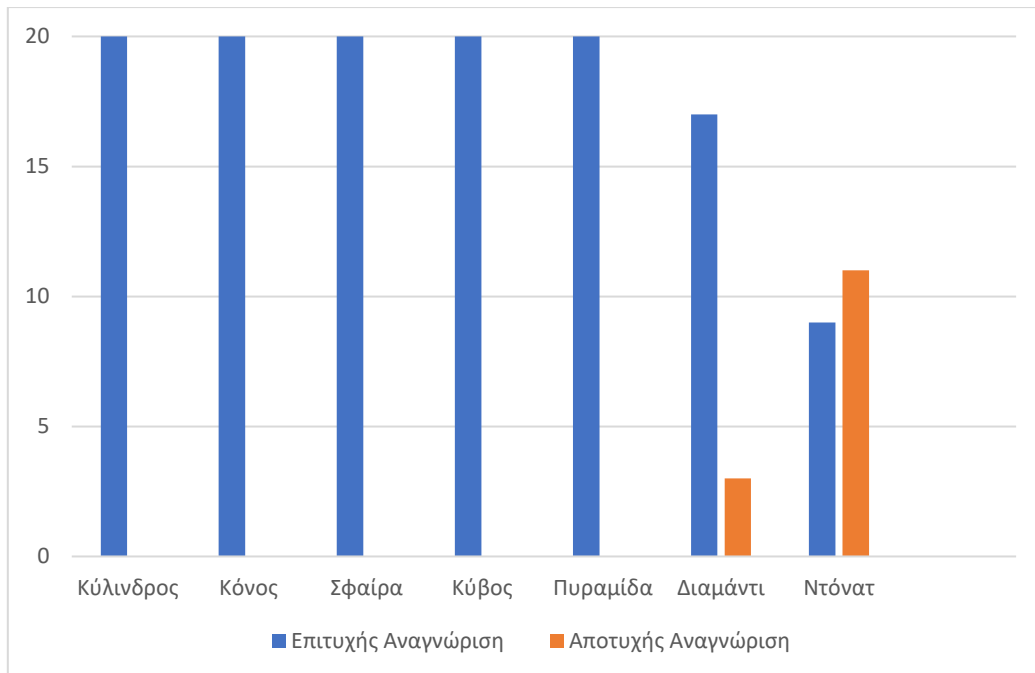
Στο δεύτερο πειραματικό περιβάλλον, οι χρήστες κλήθηκαν να αλληλεπιδράσουν με επιφάνειες διαφορετικής υφής και να αξιολογήσουν κατά πόσο η απτική αίσθηση κύλισης του βραχίονα αντιστοιχούσε στο οπτικό ερέθισμα που παρείχε το αντικείμενο. Η αλληλεπίδραση αυτή προκάλεσε έντονο ενδιαφέρον και ενθουσιασμό στους συμμετέχοντες, οι οποίοι ανέφεραν υψηλό βαθμό αντιστοιχίας μεταξύ οπτικής και απτικής πληροφορίας.

### 6.3.5 Αναγνώριση Σχημάτων μέσω της Ψηλάφησης (Shapes)

Στον τελευταίο πειραματικό χώρο οι χρήστες κλήθηκαν να αναγνωρίσουν ένα κρυμμένο γεωμετρικό σχήμα σε καθεμία από τις οκτώ ειδικά διαμορφωμένες αίθουσες. Τα σχήματα ήταν τοποθετημένα με βαθμό αυξανόμενης δυσκολίας, ανάλογα με το πόσο οικεία ή ασυνήθιστα ήταν για τον χρήστη. Τα αντικείμενα στις πρώτες έξι αίθουσες αναγνωρίστηκαν από όλους τους συμμετέχοντες με σχετική ευκολία, με τον χρόνο αναγνώρισης να κυμαίνεται από μερικά δευτερόλεπτα έως περίπου τρία λεπτά ανά χρήστη.

Τα αντικείμενα στις δύο τελευταίες αίθουσες αντιστοιχούσαν σε ασυνήθιστα γεωμετρικά σχήματα, αυξημένης δυσκολίας. Το αντικείμενο στην προτελευταία αίθουσα αναγνωρίστηκε επιτυχώς από 17 στους 20 συμμετέχοντες, με τον χρόνο αλληλεπίδρασης να κυμαίνεται από ένα έως πέντε λεπτά ανά χρήστη. Αντίθετα, το αντικείμενο της τελευταίας αίθουσας, το οποίο παρουσίαζε την υψηλότερη δυσκολία, αναγνωρίστηκε αρχικά μόνο από 4 άτομα σε διάστημα χρόνου πέντε λεπτών. Συνολικά, το συγκεκριμένο αντικείμενο αναγνωρίστηκε από 9 συμμετέχοντες με τον χρόνο αλληλεπίδρασης να φτάνει έως και τα δεκαπέντε λεπτά ανά χρήστη. Η μειωμένη επιτυχία στην αναγνώριση του τελικού

αντικειμένου θεωρείται αναμενόμενη, καθώς η νοητική οπτικοποίησή του απαιτεί ιδιαίτερα ανεπτυγμένη ικανότητα αντίληψης και αναπαράστασης τρισδιάστατης γεωμετρίας μέσω απτικής πληροφορίας. Τα αποτελέσματα του πειράματος ανά αίθουσα παρουσιάζονται αναλυτικά στο Σχήμα 6.4.



Σχήμα 6.4: Αποτελέσματα πειράματος Αναγνώρισης Σχημάτων

Αξίζει να σημειωθεί ότι κατά τη διάρκεια του πειράματος δεν τέθηκε χρονικός περιορισμός για την αναγνώριση των αντικειμένων και οι χρόνοι που αναφέρονται αποτελούν παρατηρήσεις που προέκυψαν από τη φυσική αλληλεπίδραση των χρηστών με το σύστημα.

#### 6.4 Συμπέρασμα Αξιολόγησης

Κατά τη διάρκεια της αξιολόγησης, οι συμμετέχοντες εξέφρασαν έντονο ενθουσιασμό και περιέργεια για τα πειράματα, αναφέροντας αυξημένη αίσθηση ρεαλισμού κατά την αλληλεπίδραση με τα εικονικά αντικείμενα. Τα αποτελέσματα καταδεικνύουν ότι η εφαρμογή επιτυγχάνει τον βασικό της στόχο, προσφέροντας μια συνεκτική, κατανοητή και εμπυθιστική εμπειρία απτικής αλληλεπίδρασης.

Παράλληλα, η παρατήρηση της συμπεριφοράς των χρηστών και τα σχόλια που συλλέχθηκαν παρείχαν πολύτιμες πληροφορίες για σημεία περαιτέρω βελτίωσης. Η ενσωμάτωση πραγματικών χρηστών στη διαδικασία αξιολόγησης συνέβαλε ουσιαστικά στη συνολική αποτίμηση του έργου, επιβεβαιώνοντας τη λειτουργικότητα της εφαρμογής σε πραγματικές συνθήκες χρήσης και αναδεικνύοντας τη σημασία της εμπειρικής δοκιμής σε συστήματα απτικής τεχνολογίας.

## Κεφάλαιο 7ο: Προτάσεις Βελτίωσης και Συμπεράσματα Εργασίας

### 7.1 Προτάσεις Εμπλουτισμού και Βελτίωσης

Παρότι η εφαρμογή καλύπτει επιτυχώς τους αρχικούς στόχους της παρούσας εργασίας και αξιολογήθηκε θετικά από τους χρήστες, προκύπτουν αρκετές δυνατότητες περαιτέρω εξέλιξης και εμπλουτισμού. Οι προτάσεις που ακολουθούν στοχεύουν τόσο στη διεύρυνση των εκπαιδευτικών και πειραματικών δυνατοτήτων της εφαρμογής όσο και στη βελτίωση της συνολικής εμπειρίας απτικής αλληλεπίδρασης.

- Υποστήριξη συσκευών εικονικής πραγματικότητας (VR), με στόχο την επίτευξη υψηλότερου επιπέδου εμπύθισης του χρήστη στο εικονικό περιβάλλον. Ο συνδυασμός απτικής ανάδρασης και εικονικής πραγματικότητας θα μπορούσε να ενισχύσει σημαντικά την αίσθηση παρουσίας και ρεαλισμού, ιδιαίτερα σε εκπαιδευτικά ή ερευνητικά σενάρια.
- Προσθήκη νέων πειραμάτων που αξιοποιούν τη λειτουργία επιλογής αντικειμένων, με σκοπό τη μελέτη ιδιοτήτων όπως η μάζα και το βάρος, επιτρέποντας στους χρήστες να αντιλαμβάνονται διαφοροποιήσεις μέσω της απτικής ανάδρασης.
- Ανάπτυξη πειραμάτων που σχετίζονται με μαγνητική έλξη και απώθηση, αξιοποιώντας δυναμικές δυνάμεις και μεταβαλλόμενες απτικές αποκρίσεις.
- Εισαγωγή πειραμάτων με πιο σύνθετα γεωμετρικά σχήματα ή μικρά αντικείμενα του πραγματικού κόσμου, αυξάνοντας τον βαθμό δυσκολίας και ενισχύοντας την εκπαίδευση της απτικής αντίληψης.
- Βελτίωση της απτικής ανάδρασης με πιο λεπτομερή παραμετροποίηση των δυνάμεων απτικής ανάδρασης στα επιμέρους πειράματα, με στόχο την ακριβέστερη απόδοση φυσικών ιδιοτήτων.
- Δυνατότητα προσαρμογής της έντασης της απτικής ανάδρασης ανάλογα με το επίπεδο εμπειρίας του χρήστη, ώστε η εφαρμογή να είναι προσβάσιμη τόσο σε αρχάριους όσο και σε πιο έμπειρους χρήστες.

### 7.2 Συμπεράσματα εργασίας

Συνολικά, η σύνδεση της Unreal Engine με το 3D Systems Touch δημιουργεί μία ενιαία πλατφόρμα αλληλεπίδρασης που βασίζεται ταυτόχρονα στην οπτική απεικόνιση και στην απτική ανάδραση. Ο συνδυασμός αυτός ξεπερνά τα όρια των παραδοσιακών εικονικών περιβαλλόντων και επιτρέπει την ανάπτυξη εφαρμογών που προσεγγίζουν τη φυσική πραγματικότητα τόσο σε λειτουργικό όσο και σε αισθητηριακό επίπεδο. Πρόκειται για μια τεχνολογία που επεκτείνει σημαντικά τις δυνατότητες των προσομοιώσεων και αποτελεί βασικό εργαλείο για την επόμενη γενιά εκπαιδευτικών, ερευνητικών και βιομηχανικών συστημάτων.

## ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] K. Salisbury, F. Conti, and F. Barbagli, “Haptic rendering: introductory concepts,” *IEEE Computer Graphics and Applications*, vol. 24, no. 2, pp. 24–32, Mar.–Apr. 2004, doi: 10.1109/MCG.2004.1274058.
- [2] M. A. Srinivasan and C. Basdogan, “Haptics in virtual environments: Taxonomy, research status, and challenges,” *Computers & Graphics*, vol. 21, no. 4, pp. 393–404, Jul. 1997, doi: 10.1016/S0097-8493(97)00030-7.
- [3] T. R. Coles, D. Meglan, and N. W. John, “The role of haptics in medical training simulators: A survey of the state of the art,” *IEEE Transactions on Haptics*, vol. 4, no. 1, pp. 51–66, Jan.–Mar. 2011, doi: 10.1109/TOH.2010.19.
- [4] F. Zara, L. Marechal, R. Chalard, R. Moreau, L. Vadcard, C. Barthod, and A. Lelevé, “Haptic training simulators in medical education,” in *Robot Design: Application to Medical Robotics*, M. A. Laribi and C. Nelson, Eds. Academic Press, 2026, pp. 117–166, doi: 10.1016/B978-0-443-24778-1.00006-0.
- [5] A. Hatira and M. Sarac, “Touch to Learn: A Review of Haptic Technology’s Impact on Skill Development and Enhancing Learning Abilities for Children,” *Advanced Intelligent Systems*, vol. 6, no. 6, Art. no. 2300731, 2024, doi: 10.1002/aisy.202300731.
- [6] M. A. Otaduy and M. C. Lin, “A modular haptic rendering algorithm for stable and transparent 6-DOF manipulation,” *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 751–762, Aug. 2006, doi: 10.1109/TRO.2006.876897.
- [7] B. Knorlein, M. Di Luca, and M. Harders, “Influence of visual and haptic delays on stiffness perception in augmented reality,” in *Proc. 8th IEEE Int. Symp. Mixed and Augmented Reality (ISMAR)*, Orlando, FL, USA, 2009, pp. 49–52, doi: 10.1109/ISMAR.2009.5336501.
- [8] 3D Systems, *Geomagic Touch: Product Specifications*. 3D Systems Inc. [Online]. Available: <https://support.3dsystems.com/>  
Accessed: Feb. 2025.
- [9] 3D Systems, *Touch™ Haptic Device*. [Online]. Available: <https://www.3dsystems.com/haptics-devices/touch>  
Accessed: 2024.
- [10] D. Morris, A. Joshi, and K. Salisbury, “Haptic Feedback Interfaces,” *Communications of the ACM*, vol. 54, no. 1, pp. 66–73, Jan. 2011.
- [11] 3D Systems, *OpenHaptics Toolkit Developer Guide*. 3D Systems Inc. [Online]. Available: <https://hapticapi.3dsystems.com/>  
Accessed: Feb. 2025.
- [12] M. O’Malley, A. Gupta, M. Gen, and Y. Li, “Shared Control in Haptic Systems for Training and Rehabilitation,” *Journal of Human–Robot Interaction*, vol. 3, no. 1, pp. 32–53, 2014.
- [13] C. Basdogan, “Haptics Technologies for Virtual Reality,” *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, pp. 295–317, 2019.
- [14] Epic Games, *Unreal Engine 5 Documentation*. [Online]. Available: <https://docs.unrealengine.com/>  
Accessed: Feb. 2025.

- [15] Epic Games, *Unreal Engine 4.27 Documentation*. [Online]. Available: <https://docs.unrealengine.com/4.27>  
Accessed: Aug. 2021.
- [16] J. J. LaViola Jr., E. Kruijff, R. P. McMahan, D. Bowman, and I. Poupyrev, *3D User Interfaces: Theory and Practice*, 2nd ed. Boston, MA, USA: Addison-Wesley, 2017.
- [17] Epic Games, “Unreal Engine 4.27 Release Highlights,” *Epic Games Blog*. [Online]. Available: <https://www.unrealengine.com/en-US/blog/unreal-engine-4-27-released>  
Accessed: Aug. 2021.
- [18] Epic Games, “Building Plugins in Unreal Engine,” *Epic Developer Community Tutorials*. [Online]. Available: <https://dev.epicgames.com/community/learning/tutorials/qz93/unreal-engine-building-plugins>  
Accessed: Feb. 2025.
- [19] Unity Technologies and IEEE, “Standards and Tools for Haptic Rendering,” *Proc. IEEE Haptics Symposium*, 2020.
- [20] Epic Games, “Unreal Editor Interface,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: <https://docs.unrealengine.com/4.27/en-US/Basics/UI>  
Accessed: Feb. 2025.
- [21] Epic Games, “Viewport Options,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/viewport-options?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/viewport-options?application_version=4.27)  
Accessed: Feb. 2025.
- [22] Epic Games, “Viewport Show Flags,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/viewport-show-flags?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/viewport-show-flags?application_version=4.27)  
Accessed: Feb. 2025.
- [23] Epic Games, “Components,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/components?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/components?application_version=4.27)  
Accessed: Feb. 2025.
- [24] Epic Games, “Collision Overview,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: <https://docs.unrealengine.com/4.27/en-US/InteractiveExperiences/Physics/Collision/Overview/>  
Accessed: Feb. 2025.
- [25] Epic Games, “Actors,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: <https://docs.unrealengine.com/4.27/en-US/ProgrammingAndScripting/ProgrammingWithCPP/UnrealArchitecture/Actors/>  
Accessed: Feb. 2025.
- [26] Epic Games, “Lighting the Environment,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/lighting-the-environment?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/lighting-the-environment?application_version=4.27)  
Accessed: Feb. 2025.
- [27] Epic Games, “Trigger Actors,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/trigger-actors?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/trigger-actors?application_version=4.27)  
Accessed: Feb. 2025.

- [28] Epic Games, “Blueprint Class UI,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/blueprint-class-ui?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/blueprint-class-ui?application_version=4.27)  
Accessed: Feb. 2025.
- [29] Epic Games, “Blueprint Overview,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/blueprint-overview?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/blueprint-overview?application_version=4.27)  
Accessed: Feb. 2025.
- [30] Epic Games, “My Blueprint Panel,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/my-blueprint?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/my-blueprint?application_version=4.27)  
Accessed: Feb. 2025.
- [31] Epic Games, “Blueprint Interface,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/blueprint-interface-ui?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/blueprint-interface-ui?application_version=4.27)  
Accessed: Feb. 2025.
- [32] Epic Games, “Graph Editor Tab,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/graph-editor-tab?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/graph-editor-tab?application_version=4.27)  
Accessed: Feb. 2025.
- [33] Epic Games, “UMG UI Designer Quick Start Guide,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/umg-ui-designer-quick-start-guide?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/umg-ui-designer-quick-start-guide?application_version=4.27)  
Accessed: Feb. 2025.
- [34] Epic Games, “Widget Blueprints,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/widget-blueprints?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/widget-blueprints?application_version=4.27)  
Accessed: Feb. 2025.
- [35] Epic Games, “Graph Editor Tab,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/graph-editor-tab?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/graph-editor-tab?application_version=4.27)  
Accessed: Feb. 2025.
- [36] Epic Games, “Project Settings,” *Unreal Engine Documentation*, version 4.27. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/project-settings?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/project-settings?application_version=4.27)
- [37] Epic Games, “Atmospheric Fog User Guide,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/atmospheric-fog-user-guide?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/atmospheric-fog-user-guide?application_version=4.27)  
Accessed: 2025.
- [38] Epic Games, “Sky Atmosphere,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/sky-atmosphere?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/sky-atmosphere?application_version=4.27)  
Accessed: 2025.
- [39] Epic Games, “Directional Lights,” *Unreal Engine 4.27 Documentation*. [Online]. Available: <https://dev.epicgames.com/documentation/en-us/unreal-engine/directional->

[lights?application\\_version=4.27](#)

Accessed: 2025.

[40] Epic Games, “Simple versus Complex Collision,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/simple-versus-complex-collision?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/simple-versus-complex-collision?application_version=4.27)

Accessed: 2025.

[41] Epic Games, “Flow Control,” *Unreal Engine Documentation*. [Online]. Available: <https://dev.epicgames.com/documentation/en-us/unreal-engine/flow-control-in-unreal-engine>

Accessed: 2025.

[42] Epic Games, “Physical Materials Reference,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/physical-materials-reference?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/physical-materials-reference?application_version=4.27)

Accessed: 2025.

[43] Epic Games, “Physical Materials,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/physical-materials?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/physical-materials?application_version=4.27)

Accessed: 2025.

[44] Epic Games, “Create a Physical Material,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/create-a-physical-material?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/create-a-physical-material?application_version=4.27)

Accessed: 2025.

[45] Epic Games, “Assign a Physical Material to a Material,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/assign-a-physical-material-to-a-material?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/assign-a-physical-material-to-a-material?application_version=4.27)

Accessed: 2025.

[46] Epic Games, “Assign a Physical Material to a Material Instance,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/assign-a-physical-material-to-a-material-instance?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/assign-a-physical-material-to-a-material-instance?application_version=4.27)

Accessed: 2025.

[47] Blender Foundation, “Displace Modifier,” *Blender Manual*. [Online]. Available: <https://docs.blender.org/manual/en/latest/modeling/modifiers/deform/displace.html>

Accessed: 2025.

[48] Blender Foundation, “Modifier Texture Options,” *Blender Manual*. [Online]. Available: [https://docs.blender.org/manual/en/latest/modeling/modifiers/common\\_options.html#texture](https://docs.blender.org/manual/en/latest/modeling/modifiers/common_options.html#texture)

Accessed: 2025.

[49] Blender Foundation, “Object Mode,” *Blender Manual*. [Online]. Available: <https://docs.blender.org/manual/en/latest/editors/3dview/modes.html#object-mode-list>

Accessed: 2025.

[50] Blender Foundation, “FBX Import/Export,” *Blender Manual*. [Online]. Available: [https://docs.blender.org/manual/en/latest/files/import\\_export/fbx.html](https://docs.blender.org/manual/en/latest/files/import_export/fbx.html)

Accessed: 2025.

- [51] Blender Foundation, “FBX Scene Import/Export Add-on,” *Blender Manual*. [Online]. Available: [https://docs.blender.org/manual/en/latest/addons/import\\_export/scene\\_fbx.html](https://docs.blender.org/manual/en/latest/addons/import_export/scene_fbx.html)  
Accessed: 2025.
- [52] Epic Games, “Importing Static Meshes Using FBX,” *Unreal Engine Documentation*. [Online]. Available: <https://dev.epicgames.com/documentation/en-us/unreal-engine/importing-static-meshes-using-fbx-in-unreal-engine>  
Accessed: 2025.
- [53] Epic Games, “FBX Best Practices,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/fbx-best-practices?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/fbx-best-practices?application_version=4.27)  
Accessed: 2025.
- [54] Epic Games, “Blueprints Quick Start Guide,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/blueprints-quick-start-guide?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/blueprints-quick-start-guide?application_version=4.27)  
Accessed: 2025.
- [55] Epic Games, “Blueprint Events,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/blueprint-events?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/blueprint-events?application_version=4.27)  
Accessed: 2025.
- [56] Epic Games, “Get Player Controller,” *Unreal Engine Blueprint API Reference*, Unreal Engine Documentation. [Online]. Available: <https://dev.epicgames.com/documentation/en-us/unreal-engine/BlueprintAPI/Game/GetPlayerController>  
Accessed: 2025.
- [57] Epic Games, “Setting Up Input on an Actor,” *Unreal Engine Documentation*. [Online]. Available: <https://dev.epicgames.com/documentation/en-us/unreal-engine/setting-up-input-on-an-actor-in-unreal-engine>  
Accessed: 2025.
- [58] Epic Games, “Input,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/input?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/input?application_version=4.27)  
Accessed: 2025.
- [59] Epic Games, “Light Components,” *Unreal Engine 4.27 Documentation*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/light-components?application\\_version=4.27](https://dev.epicgames.com/documentation/en-us/unreal-engine/light-components?application_version=4.27)  
Accessed: 2025.
- [60] Epic Games, “Open Level (by Name),” *Unreal Engine Blueprint API Reference*. [Online]. Available: [https://dev.epicgames.com/documentation/en-us/unreal-engine/BlueprintAPI/Game/OpenLevel\\_byName](https://dev.epicgames.com/documentation/en-us/unreal-engine/BlueprintAPI/Game/OpenLevel_byName)  
Accessed: 2025.
- [61] Unreal Engine Forums, “Unable to run community plugins when packaged,” *Unreal Engine Community Forums*. [Online]. Available: <https://forums.unrealengine.com/t/unable-to-run-community-plugins-when-packaged/288877>  
Accessed: Jan. 2025.

[62] The Engineering Toolbox, “Friction Coefficients (Static and Kinetic),” *EngineeringToolbox.com*. [Online]. Available: [https://www.engineeringtoolbox.com/friction-coefficients-d\\_778.html](https://www.engineeringtoolbox.com/friction-coefficients-d_778.html)  
Accessed: 2025.

[63] MatWeb, “Material Property Data,” *MatWeb Material Property Data*. [Online]. Available: <https://www.matweb.com/search/PropertySearch.aspx>  
Accessed: 2025

## **ΠΑΡΑΡΤΗΜΑ Α : Επιπλέον Σύνδεσμοι**

Παρατίθεται ο δημόσιος σύνδεσμος στο Google Drive με την εφαρμογή:  
[https://drive.google.com/file/d/1SGWQpY5vwJtkWDE8xvfrx7k--2a25udJ/view?usp=drive\\_link](https://drive.google.com/file/d/1SGWQpY5vwJtkWDE8xvfrx7k--2a25udJ/view?usp=drive_link)

Παρατίθεται ο δημόσιος σύνδεσμος στο Google Drive με τις αλλαγές που έγιναν στο πρόσθετο:  
[https://drive.google.com/drive/folders/1AmwOPPiI2oZem6SOghrK-7wk8INWjewO?usp=drive\\_link](https://drive.google.com/drive/folders/1AmwOPPiI2oZem6SOghrK-7wk8INWjewO?usp=drive_link)

Από την πλατφόρμα Fab, το πρόσθετο HapticsDirect for Unreal της 3D Systems:  
<https://www.fab.com/listings/45d8740b-dba8-4d6e-9b58-0f329c7092a8>

Από την πλατφόρμα Fab, το πακέτο γραφικών Blueprints: <https://www.fab.com/listings/720af3f1-cd24-40a3-a881-ee695a7c9779>

Από την πλατφόρμα Fab, το πακέτο γραφικών Content Examples:  
<https://www.fab.com/listings/4d251261-d98c-48e2-baee-8f4e47c67091>

Από την πλατφόρμα Fab, το πακέτο γραφικών 300+ Ultimate PBR Materials Pack:  
<https://www.fab.com/listings/6d05888f-bbe7-4f38-8db0-7147ed517fd2>

Από την πλατφόρμα Fab, το πακέτο γραφικών 99 PBR Materials Vol.01:  
<https://www.fab.com/listings/45f3424e-6057-4cbe-9d0a-fb245123a2ab>

Από την πλατφόρμα Fab, το πακέτο γραφικών Automotive Materials:  
<https://www.fab.com/listings/5dd132fe-ee32-4e8c-9cd3-7496547dfb29>