



ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ  
ΕΦΑΡΜΟΓΗ ΕΠΑΥΞΗΜΕΝΗΣ  
ΠΡΑΓΜΑΤΙΚΟΤΗΤΑΣ ΜΕ ΑΝΑΠΑΡΑΣΤΑΣΗ  
ΙΣΤΟΡΙΚΩΝ ΜΑΧΩΝ ΑΠΟ ΤΟ ΜΑΘΗΜΑ ΤΗΣ Δ'  
ΔΗΜΟΤΙΚΟΥ



Του φοιτητή  
Βασίλειου Κυνδελέρου  
2019087

Επιβλέπων  
Ευκλείδης Κεραμόπουλος  
Καθηγητής

Θεσσαλονίκη 2024

Τίτλος Δ.Ε. Εφαρμογή επαυξημένης πραγματικότητας με αναπαράσταση ιστορικών μαχών από το μάθημα της Δ' Δημοτικού

Κωδικός Δ.Ε. 23312

Όνοματεπώνυμο φοιτητή/τών Βασίλειος Κυνδελέρος

Όνοματεπώνυμο εισηγητή Ευκλείδης Κεραμόπουλος

Ημερομηνία ανάληψης Δ.Ε. 4/11/2023

Ημερομηνία περάτωσης Δ.Ε. 10/09/2024

*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 του φοιτητή Βασίλειου Κυνδελέρου 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 απαραίτητα και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

*«Στην οικογένειά μου»*



## Πρόλογος

Τα τελευταία χρόνια έχει παρατηρηθεί ραγδαία εξέλιξη της τεχνολογίας. Μια από όλες αυτές είναι της Επαυξημένης Πραγματικότητας, η οποία χρησιμοποιείται πλέον σε πολλές περιπτώσεις εφαρμογών. Έχουν γίνει προτάσεις να χρησιμοποιείται στον τομέα της εκπαίδευσης για να γίνουν πιο ενδιαφέροντα μαθήματα του σχολείου.

Ο λόγος που επέλεξα να υλοποιήσω μια εφαρμογή με την χρήση της Επαυξημένης Πραγματικότητας είναι γιατί παρατηρώ την εξέλιξη που θα έχει μέσα στα επόμενα χρόνια. Επιπλέον, τα παιδιά αυτής της γενιάς είναι εξοικειωμένα με τις κινητές συσκευές, και θα μπορούν να αξιοποιούν την χρήση του κινητού τηλεφώνου, χωρίς να σπαταλάνε τον χρόνο τους μόνο στα μέσα κοινωνικής δικτύωσης. Με αυτόν τον τρόπο, θα μπορεί το παιδί μέσα από ένα διαδραστικό παιχνίδι να βελτιώσει τις γνώσεις τους και να έχει αργότερα μεγαλύτερο κίνητρο να μάθει περισσότερα.

## Περίληψη

Στην συγκεκριμένη διπλωματική εργασία δημιουργήθηκε ένα εκπαιδευτικό παιχνίδι για το μάθημα της Ιστορίας Δ' Δημοτικού. Σε αυτό το παιχνίδι θα γίνει χρήση της Επαυξημένης Πραγματικότητας και θα περιέχει δύο μάχες από την ύλη του μαθήματος. Σε κάθε μάχη θα υπάρχουν τρία μέρη: το πρώτο μέρος είναι οι αναπαραστάσεις που δείχνουν τα πιο σημαντικά στιγμιότυπα, το δεύτερο είναι οι διαδραστικές σκηνές που ο χρήστης θα μετακινεί αντικείμενα με το δάκτυλό του και το τρίτο είναι το κουίζ που θα εμφανιστεί στο τέλος της αναπαραστάσης και θα περιέχει ερωτήσεις από την μάχη που μόλις είδε. Ο σκοπός αυτής της διπλωματικής εργασίας είναι να κάνει πιο ενδιαφέρον και διαδραστικό το μάθημα της Ιστορίας για τους μαθητές της Δ' Δημοτικού.

**Λέξεις-κλειδιά: Επαυξημένη Πραγματικότητα, Storytelling, MR, Unity, Objects, Blender 3D**

# Application of Augmented Reality with representation of historical battles from the 4<sup>th</sup> Grade lesson

Vasileios Kyndeleros

## **Abstract**

In this thesis, an educational game was created for the 4<sup>th</sup> grade History course. This game utilizes Augmented Reality and includes two battles from the course material. Each battle consists of three parts: the first part features representations showing the most important moments, the second one includes interactive scenes where the user can move objects with their finger, and the third part is a quiz that appears at the end of representation and contains questions about the battle that was just witnessed. The purpose of this thesis is to make the History course more engaging and interactive for 4<sup>th</sup> grade students.

**Keywords: Augmented Reality, Storytelling, MR, Unity, Objects, Blender 3D**

## **Ευχαριστίες**

Αρχικά θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή, τον κύριο Ευκλείδη Κεραμόπουλο για την δυνατότητα που με έδωσε την ανάληψη της συγκεκριμένης διπλωματικής εργασίας επειδή η Επαυξημένη Πραγματικότητα είναι μια αναπτυσσόμενη τεχνολογία που την θεωρώ αρκετά ενδιαφέρονσα.

Τέλος, θέλω να ευχαριστήσω και την οικογένειά μου που όταν είχα χάσει αρκετό χρόνο από την υλοποίηση της εργασίας, λόγω διάφορων προβλημάτων που με προέκυψαν, με στήριξαν να την συνεχίσω μέχρι την ολοκλήρωσή της.

# Περιεχόμενα

Πρόλογος.....	v
Περίληψη.....	vi
Abstract .....	vii
Ευχαριστίες .....	viii
Περιεχόμενα .....	ix
Κατάλογος Εικόνων .....	xiii
Κατάλογος Πινάκων.....	xvi
Συνομογραφίες.....	xvii
Κεφάλαιο 1ο: Εισαγωγή.....	1
1.1 Πεδίο Έρευνας.....	1
1.2 Περιγραφή – Σκοπός Εφαρμογής.....	1
1.3 Δομή Εργασίας.....	1
Κεφάλαιο 2ο: Επαυξημένη Πραγματικότητα.....	3
2.1 Εισαγωγή.....	3
2.2 Ορισμός Επαυξημένης Πραγματικότητας.....	3
2.3 Χρήση Επαυξημένης Πραγματικότητας στην Εκπαίδευση .....	4
2.3.1 Επαυξημένη Πραγματικότητα.....	4
2.3.2 Η AR στην Εκπαίδευση .....	5
2.3.3 Πλεονεκτήματα της AR στην Εκπαίδευση.....	5
2.3.4 Προκλήσεις της AR στην Εκπαίδευση.....	6
2.4 Μεθοδολογία της AR .....	6
2.4.1 Απαιτήσεις του Σχεδιασμού .....	7
2.4.2 Φυσικές Απαιτήσεις .....	7
2.4.3 Απαιτήσεις της Διεπαφής.....	7
2.4.4 Απαιτήσεις της Υλοποίησης.....	8
2.5 Επίλογος.....	8
Κεφάλαιο 3ο: Τεχνολογίες – Γλώσσα Προγραμματισμού.....	9
3.1 Εισαγωγή.....	9
3.2 Unity.....	9
3.2.1 Απαιτήσεις Συστήματος.....	9
3.3 Vuforia Engine .....	10
3.4 Visual Studio Code.....	11

3.5	C# Programming Language.....	11
3.6	Blender 3D .....	11
3.7	Επίλογος.....	13
Κεφάλαιο 4ο: Δημιουργία-Κίνηση 3D αντικειμένων στο Blender .....		15
4.1	Εισαγωγή.....	15
4.2	Περιοχές .....	15
4.2.1	Θερμοπύλες.....	15
4.2.2	Σαλαμίνα .....	18
4.3	Χαρακτήρες.....	20
4.3.1	Ελληνικός και Περσικός στρατός.....	20
4.3.2	Λεωνίδας .....	24
4.3.3	Ξέρξης .....	27
4.3.4	Εφιάλτης.....	30
4.3.5	Ευρυβιάδης.....	31
4.3.6	Θεμιστοκλής.....	33
4.4	Αντικείμενα Περσικού στρατού .....	34
4.5	Στόλοι.....	34
4.5.1	Ελληνική Τριήρης .....	34
4.5.2	Περσική Τριήρης.....	35
4.6	Image Textures.....	36
4.7	Επίλογος.....	44
Κεφάλαιο 5ο: Εφαρμογή και Εισαγωγική Σκηνή.....		45
5.1	Εισαγωγή.....	45
5.2	Ονομασία και Εικονίδιο Εφαρμογής.....	45
5.3	Δομή Εφαρμογής.....	46
5.4	Εφαρμογή .....	47
5.5	Εισαγωγική Σκηνή.....	47
5.5.1	Info Scene.....	49
5.5.2	Exit Panel .....	50
5.5.3	Menu Scene .....	52
5.5.4	Start Battle Scene .....	53
5.6	Επίλογος.....	54
Κεφάλαιο 6ο: Δημιουργία AR Scene .....		55
6.1	Εισαγωγή.....	55
6.2	Εισαγωγή Vuforia Engine στο Unity App.....	55

6.3	AR Camera.....	56
6.4	License Key.....	56
6.5	Image Target.....	57
6.6	Εισαγωγή Τρισδιάστατων Αντικειμένων και Panel .....	59
6.7	Lean Touch.....	60
6.8	Επίλογος.....	61
Κεφάλαιο 7ο: Μάχη των Θερμοπυλών .....		63
7.1	Εισαγωγή.....	63
7.2	Ιστορική Αναδρομή.....	63
7.3	Δομή Φακέλου Thermopyles.....	63
7.4	Δημιουργία Σκηνών Αναπαράστασης.....	64
7.4.1	Int1T - Διάταξη στρατευμάτων .....	64
7.4.2	Par1T - Οι Πέρσες φτάνουν στις Θερμοπύλες .....	66
7.4.3	Par2T - Συζήτηση μεταξύ Λεωνίδα και Ξέρξη .....	67
7.4.4	Int2T - Διάταξη περσικού στρατού .....	68
7.4.5	Par3T - Εμφάνιση Εφιάλτη .....	69
7.4.6	Par4T - Τέλος της Μάχης των Θερμοπυλών.....	71
7.4.7	Ερωτήσεις Γνώσεων.....	72
7.5	Επίλογος.....	75
Κεφάλαιο 8ο: Ναυμαχία της Σαλαμίνας .....		77
8.1	Εισαγωγή.....	77
8.2	Ιστορική Αναδρομή.....	77
8.3	Δομή Φακέλου Salamina.....	77
8.4	Δημιουργία Σκηνών Αναπαράστασης.....	78
8.4.1	Par1S - Οι Πέρσες φτάνουν στην Αθήνα .....	78
8.4.2	Par2S - Λογομαχία Ευρυβιάδη και Θεμιστοκλή .....	80
8.4.3	Int1S – Στόλοι .....	81
8.4.4	Par3S - Μετακίνηση Ελληνικού στόλου .....	82
8.4.5	Par4S - Μετακίνηση Περσικού στόλου.....	84
8.4.6	Int2S - Ξέρξης .....	85
8.4.7	Par5S - Τέλος της Ναυμαχίας της Σαλαμίνας .....	87
8.4.8	Ερωτήσεις Γνώσεων.....	88
8.5	Επίλογος.....	90
Κεφάλαιο 9ο: Συμπεράσματα και Προτάσεις Βελτίωσης.....		91
ΒΙΒΛΙΟΓΡΑΦΙΑ.....		92

ΠΑΡΑΡΤΗΜΑ Α : ΚΩΔΙΚΑΣ SCRIPT Home .....	94
ΠΑΡΑΡΤΗΜΑ Β : ΚΩΔΙΚΑΣ Info.....	94
ΠΑΡΑΡΤΗΜΑ C : ΚΩΔΙΚΑΣ SCRIPT Menu.....	95
ΠΑΡΑΡΤΗΜΑ D : ΚΩΔΙΚΑΣ SCRIPT StartT.....	95
ΠΑΡΑΡΤΗΜΑ E : ΚΩΔΙΚΑΣ SCRIPT StartS .....	96
ΠΑΡΑΡΤΗΜΑ F : ΚΩΔΙΚΑΣ SCRIPT Int1T.....	97
ΠΑΡΑΡΤΗΜΑ G : ΚΩΔΙΚΑΣ SCRIPT Int2T.....	97
ΠΑΡΑΡΤΗΜΑ H : ΚΩΔΙΚΑΣ SCRIPT Par1T .....	98
ΠΑΡΑΡΤΗΜΑ I : ΚΩΔΙΚΑΣ SCRIPT Par2T.....	99
ΠΑΡΑΡΤΗΜΑ J : ΚΩΔΙΚΑΣ SCRIPT Par3T .....	100
ΠΑΡΑΡΤΗΜΑ K : ΚΩΔΙΚΑΣ SCRIPT Par4T .....	101
ΠΑΡΑΡΤΗΜΑ L : ΚΩΔΙΚΑΣ SCRIPT QuizT .....	102
ΠΑΡΑΡΤΗΜΑ M : ΚΩΔΙΚΑΣ SCRIPT Int1S .....	107
ΠΑΡΑΡΤΗΜΑ N : ΚΩΔΙΚΑΣ SCRIPT Int2S .....	108
ΠΑΡΑΡΤΗΜΑ O : ΚΩΔΙΚΑΣ SCRIPT Par1S .....	109
ΠΑΡΑΡΤΗΜΑ P : ΚΩΔΙΚΑΣ SCRIPT Par2S.....	110
ΠΑΡΑΡΤΗΜΑ Q : ΚΩΔΙΚΑΣ SCRIPT Par3S .....	110
ΠΑΡΑΡΤΗΜΑ R : ΚΩΔΙΚΑΣ SCRIPT Par4S .....	111
ΠΑΡΑΡΤΗΜΑ S : ΚΩΔΙΚΑΣ SCRIPT Par5S.....	112
ΠΑΡΑΡΤΗΜΑ T : ΚΩΔΙΚΑΣ SCRIPT QuizS .....	113
ΠΑΡΑΡΤΗΜΑ U : ΚΩΔΙΚΑΣ SCRIPT Move1T.....	119
ΠΑΡΑΡΤΗΜΑ V : ΚΩΔΙΚΑΣ SCRIPT Move3T.....	120
ΠΑΡΑΡΤΗΜΑ W : ΚΩΔΙΚΑΣ SCRIPT Move4T .....	122
ΠΑΡΑΡΤΗΜΑ X : ΚΩΔΙΚΑΣ SCRIPT Move1S .....	123
ΠΑΡΑΡΤΗΜΑ Y : ΚΩΔΙΚΑΣ SCRIPT Move3S .....	124
ΠΑΡΑΡΤΗΜΑ Z : ΚΩΔΙΚΑΣ SCRIPT Move4S.....	125
ΠΑΡΑΡΤΗΜΑ AA : ΚΩΔΙΚΑΣ SCRIPT Move 5S .....	127
ΠΑΡΑΡΤΗΜΑ BB : ΚΩΔΙΚΑΣ SCRIPT Object Drag.....	127

## Κατάλογος Εικόνων

Εικόνα 2.1: Ταξονομία Milgram .....	15
Εικόνα 2.2: Pokemon Go .....	5
Εικόνα 2.3: Κύβοι σε Εφαρμογή AR .....	7
Εικόνα 2.4: Scale, Rotate, Locate Αντικειμένων .....	8
Εικόνα 4.1: Θερμοπύλες.....	15
Εικόνα 4.2: Μονοπάτι .....	16
Εικόνα 4.3: Έδαφος.....	16
Εικόνα 4.4: Θάλασσα.....	16
Εικόνα 4.5: Χαρακτηριστικά Landscape Object.....	17
Εικόνα 4.6: Αλλαγή Image Texture .....	17
Εικόνα 4.7: Σαλαμίνα.....	18
Εικόνα 4.8: Join.....	18
Εικόνα 4.9: Τείχη Αρχαίας Αθήνας.....	19
Εικόνα 4.10: Σπίτι στην Αρχαία Αθήνα .....	20
Εικόνα 4.11: Εσωτερικό Μονοπάτι.....	20
Εικόνα 4.12: Έλληνας Στρατιώτης.....	21
Εικόνα 4.13: Πέρσης Στρατιώτης .....	22
Εικόνα 4.14: Animation Επίθεσης Ελλήνων - Περσών .....	23
Εικόνα 4.15: Animation Run Ελλήνων .....	23
Εικόνα 4.16: Animation Death Ελλήνων .....	23
Εικόνα 4.17: Animation Walk Περσών.....	24
Εικόνα 4.18: Λεωνίδα.....	25
Εικόνα 4.19: Animation Talk Λεωνίδα .....	26
Εικόνα 4.20: Animation Attack Λεωνίδα.....	26
Εικόνα 4.21: Animation Death Λεωνίδα.....	27
Εικόνα 4.22: Ξέρξης.....	27
Εικόνα 4.23: Animation Walk Ξέρξη.....	28
Εικόνα 4.24: Animation Talk Ξέρξη.....	29
Εικόνα 4.25: Animation Look Ξέρξη.....	29
Εικόνα 4.26: Εφιάλτης .....	30
Εικόνα 4.27: Animation Pointing Εφιάλτη .....	31
Εικόνα 4.28: Animation Walk Εφιάλτη .....	31
Εικόνα 4.29: Ευρυβιάδη.....	32
Εικόνα 4.30: Animation Ευρυβιάδη.....	32
Εικόνα 4.31: Θεμιστοκλής .....	33
Εικόνα 4.32: Animation Θεμιστοκλή.....	34
Εικόνα 4.33: Αντικείμενα Διάταξης Περσικού Στρατού .....	34
Εικόνα 4.34: Ελληνική Τριήρης.....	35
Εικόνα 4.35: Animation Ελληνικής Τριήρης .....	35
Εικόνα 4.36: Περσική Τριήρης .....	36
Εικόνα 4.37: Animation Περσικής Τριήρης.....	36
Εικόνα 4.38: Grass Texture.....	36
Εικόνα 4.39: Mud Texture .....	37
Εικόνα 4.40: Stone Mountain Texture .....	37

Εικόνα 4.41: Mountain Tree Texture .....	37
Εικόνα 4.42: Pathway Texture .....	38
Εικόνα 4.43: Coast Sand Texture .....	38
Εικόνα 4.44: Cliff Texture .....	38
Εικόνα 4.45: Rock Texture.....	39
Εικόνα 4.46: Stone Wall Texture .....	39
Εικόνα 4.47: Sea Texture .....	39
Εικόνα 4.48: Ship Texture (Greek) .....	40
Εικόνα 4.49: Ship Texture (Persian) .....	40
Εικόνα 4.50: Deck Texture (Greek) .....	40
Εικόνα 4.51: Deck Texture (Greek) .....	41
Εικόνα 4.52: Sails Texture (Greek).....	41
Εικόνα 4.53: Sails Texture (Persian).....	41
Εικόνα 4.54: Metal Texture (Greek) .....	42
Εικόνα 4.55: Metal Texture (Persian) .....	42
Εικόνα 4.56: Paddle Texture .....	42
Εικόνα 4.57: Eye Texture.....	43
Εικόνα 4.58: Grey Plain Texture.....	43
Εικόνα 4.59: Door Texture.....	43
Εικόνα 4.60: Ceramic Roof Texture .....	44
Εικόνα 4.61: Stone Pathway Texture .....	44
Εικόνα 5.1: Icon Εφαρμογής.....	45
Εικόνα 5.2: Tab Player .....	46
Εικόνα 5.3: Δομή Εφαρμογής .....	46
Εικόνα 5.4: Δομή Φακέλου Thermopyles .....	47
Εικόνα 5.5: Home Scene .....	48
Εικόνα 5.6: Texture Type to Sprite .....	49
Εικόνα 5.7: Info Scene .....	50
Εικόνα 5.8: Exit Panel.....	50
Εικόνα 5.9: C# Script - Exit Panel .....	51
Εικόνα 5.10: Home Scene Manager.....	51
Εικόνα 5.11: Button onClick() .....	52
Εικόνα 5.12: Menu Scene.....	52
Εικόνα 5.13: Έναρξη Θερμοπυλών.....	53
Εικόνα 5.14: C# Script - Scene Load Int1T .....	54
Εικόνα 5.15: Έναρξη Σαλαμίνας.....	54
Εικόνα 5.16: C# Script - Scene Load Par1S.....	54
Εικόνα 6.1: Import Vuforia Engine.....	55
Εικόνα 6.2: Επιλογές Σκηνής .....	56
Εικόνα 6.3: AR Camera .....	57
Εικόνα 6.4: Μήνυμα Σφάλματος.....	57
Εικόνα 6.5: Image Target Εφαρμογής.....	58
Εικόνα 6.6: Image Target Inspector .....	58
Εικόνα 6.7: Τοποθέτηση Αντικειμένων στο Image Target .....	59
Εικόνα 6.8: Επιλογή Lean Touch.....	60
Εικόνα 6.9: Εισαγωγή Box Collider.....	61

Εικόνα 7.1: Δομή Φακέλου Thermopyles μαζί με το Quiz.....	63
Εικόνα 7.2: Περιεχόμενο Υποφακέλων Thermopyles .....	64
Εικόνα 7.3: Πρώτη Διαδραστική Σκηνή Θερμοπυλών .....	65
Εικόνα 7.4: Μετακίνηση Αντικειμένων .....	65
Εικόνα 7.5: Πρώτη Σκηνή Αναπαράστασης Θερμοπυλών .....	66
Εικόνα 7.6: Animation και Location Περσών.....	67
Εικόνα 7.7: Δεύτερη Σκηνή Αναπαράστασης Θερμοπυλών.....	67
Εικόνα 7.8: Animation Λεωνίδα - Ξέρξη.....	68
Εικόνα 7.9: Δεύτερη Διαδραστική Σκηνή Θερμοπυλών.....	69
Εικόνα 7.10: Ορισμός Διάταξης Στρατού .....	69
Εικόνα 7.11: Τρίτη Σκηνή Αναπαράστασης Θερμοπυλών .....	70
Εικόνα 7.12: Animation και Location Στρατιωτών - Εφιάλτη.....	70
Εικόνα 7.13: Τέταρτη Σκηνή Αναπαράστασης Θερμοπυλών .....	71
Εικόνα 7.14: Animation και Location Στρατιωτών - Βασιλιάδων.....	72
Εικόνα 7.15: Αρχή του Κουίζ Θερμοπυλών.....	72
Εικόνα 7.16: Ερωτήσεις Μάχης Θερμοπυλών .....	73
Εικόνα 7.17: Σωστή - Λάθος Απάντηση .....	74
Εικόνα 7.18: Εισαγωγή Audio Sources.....	74
Εικόνα 7.19: Panel Αποτελέσματος Κουίζ.....	75
Εικόνα 8.1: Δομή Φακέλου Salamina .....	77
Εικόνα 8.2: Δομή Υποφακέλων Salamina .....	78
Εικόνα 8.3: Πρώτη Σκηνή Αναπαράστασης Σαλαμίνας .....	79
Εικόνα 8.4: Animation και Location Περσών.....	79
Εικόνα 8.5: Δεύτερη Σκηνή Αναπαράστασης Σαλαμίνας.....	80
Εικόνα 8.6: Animation Ευρυβιάδη - Θεμιστοκλή.....	81
Εικόνα 8.7: Πρώτη Διαδραστική Σκηνή Σαλαμίνας.....	81
Εικόνα 8.8: Μετακίνηση Στόλων.....	82
Εικόνα 8.9: Τρίτη Σκηνή Αναπαράστασης Σαλαμίνας .....	83
Εικόνα 8.10: Animation και Location Ελληνικού Στόλου.....	84
Εικόνα 8.11: Τέταρτη Σκηνή Αναπαράστασης Σαλαμίνας.....	84
Εικόνα 8.12: Animation και Location Περσικού Στόλου - Ξέρξη.....	85
Εικόνα 8.13: Δεύτερη Διαδραστική Σκηνή Σαλαμίνας.....	86
Εικόνα 8.14: Μετακίνηση Βασιλιά Ξέρξη.....	86
Εικόνα 8.15: Πέμπτη Σκηνή Αναπαράστασης Σαλαμίνας.....	87
Εικόνα 8.16: Animation και Location Αναχώρησης Περσικού Στόλου .....	88
Εικόνα 8.17: Αρχική Κουίζ Σαλαμίνας.....	89
Εικόνα 8.18: Ερωτήσεις Ναυμαχίας Σαλαμίνας .....	89

## **Κατάλογος Πινάκων**

Πίνακας 2.1: Development Tools and Apps .....	6
Πίνακας 3.1: Απαιτήσεις Unity Editor .....	10
Πίνακας 3.2: Απαιτήσεις Unity Player.....	10
Πίνακας 3.3: Εκδόσεις Blender.....	12
Πίνακας 3.4: Απαιτήσεις Hardware - Blender .....	12

## Συντομογραφίες

AR – Augmented Reality

VR – Virtual Reality

MR – Mixed Reality

LTS – Long Term Support

UI – User Interface

VS – Visual Studio

IDE – Integrated Drive Electronics

OS – Operating System

API – Application Programming Interface

π.χ. – παραδείγματος χάριν

π.Χ. – προ Χριστού

κτλ – και τα λοιπά



## Κεφάλαιο 1ο: Εισαγωγή

### 1.1 Πεδίο Έρευνας

Τα εκπαιδευτικά παιχνίδια είναι αρκετά βοηθητικά για τον χρήστη, καθώς με έναν πιο ψυχαγωγικό τρόπο βοηθάει στην βελτίωση των γνώσεων του, κι έτσι ολοκληρώνεται η εκπαίδευσή του με μεγάλη επιτυχία. Η Επαυξημένη Πραγματικότητα (Ε.Π.) χρησιμοποιείται αρκετά στα εκπαιδευτικά παιχνίδια, καθώς με έναν εναλλακτικό τρόπο μπορεί να βοηθήσει στην εκπαίδευση του χρήστη. Στην συγκεκριμένη μελέτη, θα παρουσιαστεί μια εφαρμογή η οποία θα δείχνει δύο αναπαραστάσεις ιστορικών μαχών και την εξέλιξη που είχε η καθεμία από αυτές, ώστε ο χρήστης-μαθητής να δει και να παρατηρήσει οτιδήποτε έχει συμβεί στις δύο μάχες. Η υλοποίηση της εφαρμογής βασίστηκε από την ύλη του μαθήματος Ιστορίας, για την τάξη της Δ' Δημοτικού, καθώς και από την βιβλιογραφική αναζήτηση. Με βάση αυτών των δύο, η διεπαφή του χρήστη και το περιεχόμενο της εφαρμογής δημιουργήθηκε με αυτόν τον τρόπο ώστε να είναι κατάλληλη για ένα παιδί που είναι μαθητής της Δ' Δημοτικού.

### 1.2 Περιγραφή – Σκοπός Εφαρμογής

Η εφαρμογή που υλοποιήθηκε, χρησιμοποιεί την τεχνολογία AR Storytelling και περιλαμβάνει δύο αναπαραστάσεις ιστορικών μαχών, οι οποίες βρίσκονται στην ύλη της Ιστορίας της Δ' Δημοτικού. Οι μάχες αυτές είναι των Θερμοπυλών και η ναυμαχία της Σαλαμίνας. Ο μαθητής θα μπορεί να διαλέξει μια από τις δύο αναπαραστάσεις και να δει με έναν εικονικό τρόπο την εξέλιξη της μάχης. Κατά την διάρκεια των αναπαραστάσεων, ο μαθητής θα πρέπει να αλληλεπιδράσει με την εφαρμογή, δηλαδή θα πρέπει να μετακινήσει με το δάκτυλό του κάποια αντικείμενα που θα του εμφανιστούν στις σωστές τους θέσεις (interaction). Στο τέλος της αναπαραστάσης, θα χρειαστεί να απαντήσει σε δέκα ερωτήσεις γνώσεων για κάθε μάχη ξεχωριστά που θα εμφανιστούν στην οθόνη του, στα οποία πρέπει να τις απαντήσει όλες για να δει το συνολικό του σκορ που μάζεψε στο επαναληπτικό κουίζ.

Ο σκοπός αυτής της εφαρμογής είναι να κάνει πιο διαδραστικό το μάθημα της Ιστορίας, και έτσι τα παιδιά θα μπορούν να δουν και εικονικά τι ακριβώς έγινε στην διάρκεια της μάχης. Με την βοήθεια των ερωτήσεων γνώσεων και των διαδραστικών σκηνών, θα μπορεί να τραβήξει περισσότερο το ενδιαφέρον του μαθητή και να μάθει περισσότερα πράγματα για τις συγκεκριμένες μάχες, κι αυτό έχει ως αποτέλεσμα, να βελτιώσει αρκετά τις γνώσεις του και κυρίως να δώσει στο παιδί μεγαλύτερο κίνητρο για μάθηση.

### 1.3 Δομή Εργασίας

Η εργασία αυτή αποτελείται από 9 κεφάλαια.

Το **1<sup>ο</sup> κεφάλαιο**, περιέχει την εισαγωγή και αναφέρει περιληπτικά την έρευνα που πραγματοποιήθηκε για την υλοποίηση της εφαρμογής, μια σύντομη περιγραφή της εργασίας και ο σκοπός για τον οποίο πρόκειται να χρησιμοποιηθεί.

Στο **2<sup>ο</sup> κεφάλαιο**, παρουσιάζει γενικές πληροφορίες για την Ε.Π. και τον τρόπο που επιδρά στον τομέα της εκπαίδευσης, αναφέροντας τα πλεονεκτήματα και τις προκλήσεις που προκύπτουν. Τέλος, αναφέρεται η μεθοδολογία της Ε.Π. ώστε να υλοποιηθεί μια εφαρμογή AR.

## Κεφάλαιο 1

Στο **3<sup>ο</sup> κεφάλαιο**, παρουσιάζονται οι τεχνολογίες και τα προγραμματιστικά περιβάλλοντα που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής, τα οποία είναι το Unity, Blender 3D, Vuforia Engine, Visual Studio Code και η γλώσσα προγραμματισμού C#.

Το **4<sup>ο</sup> κεφάλαιο**, δείχνει την δημιουργία των τρισδιάστατων αντικειμένων καθώς και τις κινήσεις αυτών μέσω της χρήσης του Blender 3D.

Στο **5<sup>ο</sup> κεφάλαιο**, παρουσιάζεται η εφαρμογή μαζί με την εισαγωγική σκηνή μέχρι να επιλέξει ο χρήστης μια από τις δύο αναπαράστασεις.

Το **6<sup>ο</sup> κεφάλαιο**, αναφέρεται η λεπτομερής διαδικασία υλοποίησης των AR σκηνών της εφαρμογής με την βοήθεια της χρήσης της τεχνολογίας Vuforia Engine και την απαραίτητη εισαγωγή συγκεκριμένων πακέτων, κλειδιών και εικόνων για την ολοκλήρωση των συγκεκριμένων σκηνών.

Το **7<sup>ο</sup> κεφάλαιο και 8<sup>ο</sup> κεφάλαιο**, εστιάζουν στην υλοποίηση της αναπαράστασης της Μάχης των Θερμοπυλών και της Ναυμαχίας της Σαλαμίνας αντίστοιχα, και για κάθε μάχη ξεχωριστά αναφέρουν μια σύντομη ιστορική αναφορά, την δομή των δύο φακέλων, τα αντικείμενα με τα animation που χρησιμοποιούνται σε κάθε σκηνή. Επίσης, παρουσιάζεται και η σκηνή με τις ερωτήσεις που βρίσκεται στο τέλος της αναπαράστασης. Για όλες τις σκηνές της αναπαράστασης και των διαδραστικών σημείων θα παρουσιαστεί αναλυτικά ο τρόπος που έχουν υλοποιηθεί.

Τέλος, στο **9<sup>ο</sup> κεφάλαιο**, παρατίθενται τα συμπεράσματα και προτεινόμενες μελλοντικές επεκτάσεις.

## Κεφάλαιο 2ο: Επαυξημένη Πραγματικότητα

### 2.1 Εισαγωγή

Στο δεύτερο κεφάλαιο της εργασίας θα δοθεί ο ορισμός της ΕΠ, και θα αναφερθούν μερικές πληροφορίες για την συγκεκριμένη τεχνολογία. Στη συνέχεια θα αναφερθούν πληροφορίες για την χρήση της ΕΠ στην περίπτωση της εκπαίδευσης, όπου θα αναφερθούν όλα τα πλεονεκτήματα που μπορεί να έχει στην εκμάθηση των μαθητών αλλά και για τις προκλήσεις που μπορεί να προκύψουν κατά την διάρκεια υλοποίησης τέτοιας εφαρμογής. Όλο το περιεχόμενο του κεφαλαίου βασίζεται από επιστημονικά άρθρα έπειτα από αρκετή μελέτη.

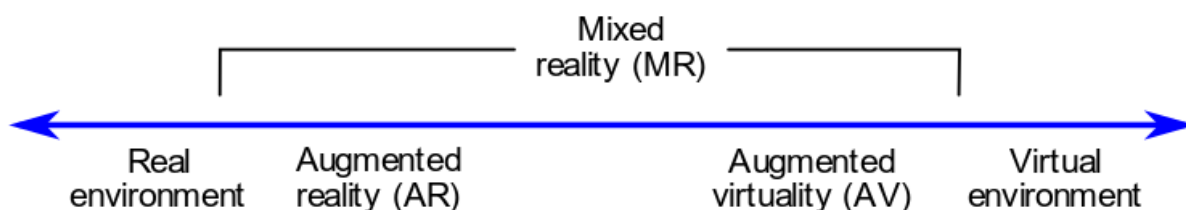
### 2.2 Ορισμός Επαυξημένης Πραγματικότητας

Την τελευταία πενταετία έχει σημειωθεί ραγδαία ανάπτυξη της εικονικής και της επαυξημένης πραγματικότητας. Αν και είμαστε ακόμη στο αρχικό στάδιο, έχουν ήδη προκύψει αρκετές περιπτώσεις για VR και AR, οι οποίες μπορούν να υποστηρίξουν και την διδασκαλία, καθώς και την μάθηση στην εκπαίδευση[1]. Έχουν γίνει πάρα πολλές προσπάθειες για να χρησιμοποιηθεί η συγκεκριμένη τεχνολογία στον τομέα της εκπαίδευσης, με τις περισσότερες από αυτές να χρησιμεύουν σε μεγάλο βαθμό.

Μερικές περιπτώσεις χρήσης είναι στην στρατιωτική εκπαίδευση, στον τομέα της μηχανικής σε εργαστήρια για την εικονική πραγματικότητα, καθώς και σε μαθήματα ιστορίας και αστρονομίας. Η επαυξημένη πραγματικότητα ως εκπαιδευτικό και πειραματικό εργαλείο, έχει το μεγάλο πλεονέκτημα να τραβήξει αρκετά το ενδιαφέρον του χρήστη για οποιαδήποτε σκηνή[2].

Παρότι υπήρχαν εκπαιδευτικές εφαρμογές που είχαν επιτυχή αποτελέσματα, υπήρχαν και άλλες που πέτυχαν μόνο σε ένα μέρος. Συγκεκριμένα στον κλάδο της ιατρικής, στην σχεδίαση τρισδιάστατων αντικειμένων στο μάθημα της ανατομίας, καθώς και στην χρήση της επαυξημένης πραγματικότητας για την μικροχειρουργική. Για να βελτιωθούν περισσότερο οι γνώσεις των μαθητών, θα πρέπει τέτοιου τύπου εφαρμογές να γίνουν πλήρως κατανοητές.

Αρκετοί ερευνητές, εκπαιδευτικοί αλλά και προγραμματιστές έχουν παρουσιάσει, μετά από πολλές μελέτες, διάφορους τρόπους για να κάνουν το μάθημα πιο ενδιαφέρον για τους μαθητές, έχοντας περάσει από τον έλεγχο αξιολόγησης των τριών υπάρχουσών ταξινομήσεων, που είναι του Hughes, του Milgram και της Vergara [3][4][5]. Η επαυξημένη πραγματικότητα κρίνεται αρκετά σημαντική για τους μαθητές για οποιαδήποτε ειδικότητα έχουν. Για αυτούς είναι αρκετά σημαντική η χρήση της στο κοντινό μέλλον, ώστε όχι μόνο να είναι πετυχημένοι, αλλά και να αναπτύξουν περισσότερο τις ικανότητές τους. Φυσικά το ίδιο ισχύει και για τους ανερχόμενους IT specialists.



Εικόνα 2.1: Ταξονομία Milgram

Σύμφωνα με την Εικόνα 2.1, μια «συνέχεια της εικονικότητας» θα μπορούσε να θεωρηθεί ως μια εναλλακτική σειρά από διαφορετικές κατηγορίες αντικειμένων οι οποίες εμφανίζονται σε μια οθόνη [6]. Σε αυτήν την συνέχεια, τα πραγματικά περιβάλλοντα παρουσιάζονται μόνο σε μια οθόνη, ενώ το εικονικό περιβάλλον μπορεί να εμφανιστεί και σε άλλες. Ωστόσο, για να μπορέσουμε να περιγράψουμε με μεγάλη ακρίβεια τα συγκεκριμένα εργαλεία, όπως είναι η λειτουργία και ο σχεδιασμός τους, θα πρέπει να γίνει πιο κατανοητός ο τρόπος με τον οποίο θα ταξινομούνται. Η επαυξημένη πραγματικότητα είναι πιο χρήσιμη στην εκπαίδευση όταν παρέχει τα παρακάτω οφέλη [7]:

- Υλοποίηση περιβάλλοντος με επίκεντρο τον μαθητή
- Παροχή απλής και εύκολης χρήσης της αλληλεπίδρασης.
- Βελτίωση στον τρόπο που διδάσκεται το μάθημα
- Αύξηση των γνώσεων και των ικανοτήτων του μαθητή
- Μείωση του κόστους της εκπαίδευσης

Μια εκπαιδευτική ταξινόμια όπου βασίζεται στη χρήση του VR και AR, μπορεί να βοηθήσει στην διαφοροποίηση καθώς και στην κατηγοριοποίηση των μαθησιακών εμπειριών και να παρουσιάσει τους λόγους που κάποιες από αυτές τις εφαρμογές έχουν επιτυχή αποτελέσματα, ενώ άλλες προκύπτουν ως αποτυχημένες προσπάθειες.

### **2.3 Χρήση Επαυξημένης Πραγματικότητας στην Εκπαίδευση**

Το συγκεκριμένο υποκεφάλαιο περιλαμβάνει τα πλεονεκτήματα της επαυξημένης πραγματικότητας στην εκπαίδευση και τις προκλήσεις που πρέπει να αντιμετωπιστούν[8][9].

#### **2.3.1 Επαυξημένη Πραγματικότητα**

Με την χρήση της AR, ο πραγματικός και ο εικονικός κόσμος μπορούν να συνδυαστούν μεταξύ τους και τα αντικείμενα να δημιουργούνται σε πραγματικό χρόνο χρησιμοποιώντας εικόνες που είναι δημιουργίες του υπολογιστή [9]. Το AR θεωρείται μια τεχνολογία που μπορεί να συνδυάσει τα πραγματικά και εικονικά αντικείμενα σε ένα πραγματικό περιβάλλον, έτσι ώστε να μπορούν να επικοινωνούν μεταξύ τους σε άμεσο χρόνο. Η μεικτή πραγματικότητα (Mixed Reality) του Milgram είναι ένα ηχηρό παράδειγμα για το πώς τα πραγματικά και εικονικά στοιχεία μπορούν να συνδυαστούν[6]. Αυτή η συνέχεια περιλαμβάνει τα πάντα από ένα πλήρως εικονικό περιβάλλον σε ένα πραγματικό περιβάλλον. Η τεχνολογία AR αυξάνει τον πραγματικό κόσμο με διάφορα ψηφιακά στοιχεία με την χρήση της κάμερας του smartphone του χρήστη.

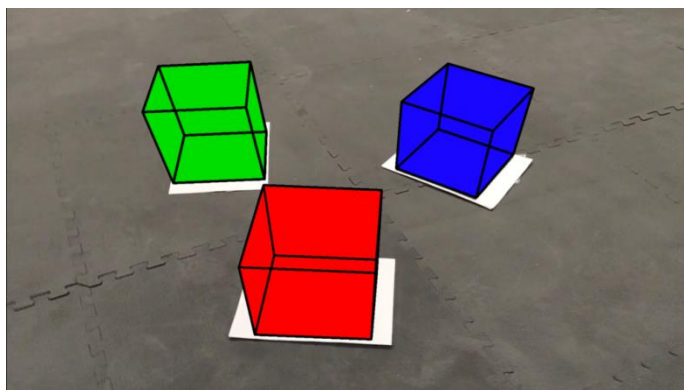
Ένα γνωστό παράδειγμα εφαρμογής AR είναι το Pokemon GO, ένα παιχνίδι που βασίζεται στην τοποθεσία του χρήστη και του επιτρέπει να συλλέγει τα ψηφιακά πλάσματα που βρίσκονται κοντά στο μέρος που βρίσκεται[10]. Με την αύξηση των κινητών συσκευών, αρκετά άτομα πλέον έχουν την δυνατότητα πρόσβασης στην επαυξημένη πραγματικότητα.



Εικόνα 2.2: Pokemon Go

### 2.3.2 Η AR στην Εκπαίδευση

Ένα βασικό στοιχείο στον τρόπο που παρουσιάζεται, αναπτύσσεται και ενσωματώνεται η τεχνολογία AR μέσα σε επίσημα και ανεπίσημα μαθησιακά περιβάλλοντα είναι ο τρόπος με τον οποίο έχουν υλοποιηθεί[11]. Η αποτελεσματικότητα της AR βασίζεται αρκετά στην υποστήριξη και στην διευκόλυνση της ουσιαστικής μάθησης. Η εκπαίδευση θα ήταν πιο παραγωγική με την εξέταση μιας έννοιας AR αντί για μια συγκεκριμένη τεχνολογία[12].



Εικόνα 2.3: Κύβοι σε Εφαρμογή AR

Για να γίνουν οι εφαρμογές AR πιο χρήσιμες στους εκπαιδευτικούς, θα πρέπει οι ίδιοι να συμμετέχουν ενεργά, καθώς η χρήση της επαυξημένης πραγματικότητας σε εφαρμογές στην εκπαίδευση έχει αυξηθεί αρκετά.

### 2.3.3 Πλεονεκτήματα της AR στην Εκπαίδευση

Με την βοήθεια της επαυξημένης πραγματικότητας, οποιοσδήποτε θα μπορεί να αλληλεπιδράσει με αντικείμενα του πραγματικού κόσμου με τρόπους όπου θα ήταν σχεδόν αδύνατο μέσα σε ένα ολοκληρωμένο εικονικό περιβάλλον είτε σε ένα εντελώς αληθινό. Στο συγκεκριμένο υβριδικό περιβάλλον μάθησης, τα πραγματικά αλλά και τα εικονικά στοιχεία θα μπορούν να συνδυαστούν και

να δημιουργήσουν ένα άκρως καθηλωτικό περιβάλλον μάθησης. Δια μέσου της AR, οι μαθητές πλέον θα μπορούν να βιώσουν δραστηριότητες όπου προηγουμένως ήταν μη-προσβάσιμες, όπως είναι για παράδειγμα οι χημικές αντιδράσεις, που στην πραγματικότητα δεν θα μπορούσαν να αλληλεπιδράσουν μαζί[13].

Δίνοντας αυτήν την ευκαιρία στους μαθητές να ζήσουν το συγκεκριμένο είδος μάθησης, θα μπορεί να ενισχυθεί η ικανότητα της εννοιολογικής τους σκέψης και να αντιλαμβάνονται φαινόμενα που θα παρατηρούν, όταν ταυτόχρονα θα διορθώνονται όποιες λανθασμένες αντιλήψεις προκύψουν. Οι μαθητές θα μπορούν να πετύχουν πιο υψηλό επίπεδο γνώσεων και δεξιοτήτων σε μια τεχνολογία όπου αναβαθμίζει σε μεγάλο βαθμό τα μαθησιακά περιβάλλοντα χρησιμοποιώντας τις τεχνολογίες της επαυξημένης πραγματικότητας.

### 2.3.4 Προκλήσεις της AR στην Εκπαίδευση

Είναι αρκετά πιθανό μερικοί μαθητές να θεωρούν δύσκολη και ακατανόητη την χρήση της τεχνολογίας AR εξαιτίας της χρηστικότητας καθώς και των τεχνικών δυσκολιών του. Στις εφαρμογές της επαυξημένης πραγματικότητας η χρηστικότητα είναι μια από τις κυριότερες προκλήσεις, ωστόσο έχει αναφερθεί παραπάνω στα πλεονεκτήματα για το πόσο εύκολη είναι η χρήση της. Επιπλέον, είναι αρκετά πιθανό οι μαθητές όταν χρησιμοποιούν την ίδια στιγμή τα πραγματικά και τα εικονικά αντικείμενα, να χαθούν ανάμεσα στην φαντασία και στην πραγματικότητα[14].

Με την χρήση της AR σε περιβάλλοντα μάθησης, έχει ως αποτέλεσμα οι μαθητές να μπορούν σίγουρα να κάνουν πολλαπλές εργασίες, αλλά και να δουλεύουν με ένα ογκώδες ποσοστό πληροφοριών και πολλαπλών ψηφιακών συσκευών. Αρκετά συχνά οι εφαρμογές της περιέχουν ένα μη ευέλικτο περιεχόμενο, που αυτό περιορίζει τους εκπαιδευτικούς να μην μπορούν να προσαρμόσουν το δικό τους περιεχόμενο μέσα σε αυτό και έτσι να είναι πιθανό να μην μπορέσει να προσαρμοστεί στις ανάγκες των μαθητών[15]. Η τεχνολογία AR για τις κινητές συσκευές μπορεί να αντιμετωπίσει και μια άλλη πρόκληση μέσα σε αυτό, η οποία είναι η μη εγγυημένη σταθερότητά της.

Επίσης, είναι αρκετά πιθανό να υπάρχει δυσκολία στη χρήση της AR, εξαιτίας των πολύ κακών σχεδιασμένων διεπαφών των εφαρμογών και του τρόπου καθοδήγησης στην εφαρμογή, με αποτέλεσμα να κάνει πιο δύσκολη τη χρήση της. Τέλος, οι χρήστες θα πρέπει να αφιερώσουν αρκετό χρόνο για να μπορέσουν να εξοικειωθούν με την τεχνολογία της επαυξημένης πραγματικότητας.

## 2.4 Μεθοδολογία της AR

Υπάρχουν αρκετά SDK διαθέσιμα για AR και VR[16]. Στον παρακάτω πίνακα εμφανίζεται η διαθέσιμη λίστα των εργαλείων ανάπτυξης στην αγορά. Για να μπορέσει κάποιος να διαλέξει άμεσα το σωστό εργαλείο που χρειάζεται, θα πρέπει να είναι εφικτό να συγκριθούν τα κύρια χαρακτηριστικά καθώς και τα χαρακτηριστικά των εργαλείων της επαυξημένης πραγματικότητας. Στον πίνακα αναφέρονται τα AR SDK, οι πλατφόρμες που μπορούν να υποστηρίξουν, για ποιες περιπτώσεις είναι κατάλληλα και το κόστος τους.

Πίνακας 2.1: Development Tools and Apps

AR SDK	Πλατφόρμα Υποστήριξης	Συμβατότητα	Τιμή
Apple ARKit	<ul style="list-style-type: none"><li>iPad Pro</li><li>iPhone SE</li></ul>	Marker-Based-Apps	Δωρεάν

ARToolKit	<ul style="list-style-type: none"> <li>• Android OS</li> <li>• iOS</li> <li>• Linux OS</li> <li>• Mac OS</li> <li>• Smart Glasses</li> </ul>	Location-Based-App	Δωρεάν
Google ARCore	<ul style="list-style-type: none"> <li>• Google Pixel</li> <li>• Samsung Galaxy S Series</li> <li>• Samsung Galaxy Note</li> <li>• Samsung Galaxy Fold</li> </ul>	Marker-Based-Apps	Δωρεάν
Vuforia	<ul style="list-style-type: none"> <li>• Android OS</li> <li>• iOS</li> <li>• Unity 3D</li> </ul>	Marker-Based-Apps	Δωρεάν

Αν ο χρήστης επιλέξει ένα από τα πακέτα του Πίνακα 2.1[18], οι παράγοντες που θα πρέπει να προσέξει κάποιος είναι οι εξής: οι πλατφόρμες που υποστηρίζει, το κόστος, την υποστήριξη του Image tracking, ώστε να αναγνωρίζει την εικόνα που θα εμφανίζεται η εφαρμογή και να υποστηρίζονται το Unity και το GPS. Με την χρήση της συγκεκριμένης μεθόδου η δομή της ταξινόμιας συλλέγεται με κύρια βάση τα τέσσερα ζητήματα που αναφέρονται παρακάτω[9].

#### 2.4.1 Απαιτήσεις του Σχεδιασμού

Για να σχεδιαστεί μια τέτοιου είδους εφαρμογή χρειάζεται πρώτα να γνωρίζουμε:

- από ποιους θα χρησιμοποιηθεί
- σε ποιο μάθημα αναφέρεται
- το επίπεδο μαθητών
- ποια πλατφόρμα θα χρησιμοποιήσουμε

#### 2.4.2 Φυσικές Απαιτήσεις

Για να χρησιμοποιηθεί η εφαρμογή από το χρήστη απαιτεί:

- μία συσκευή εισόδου και εξόδου, ώστε να υπάρχει δυνατότητα επικοινωνίας με το λογισμικό
- Το back-end που δείχνει τον τρόπο που παραδίδονται AR και VR.

#### 2.4.3 Απαιτήσεις της Διεπαφής

Μια καλή διεπαφή χρήστη απαιτείται ώστε να καθορίσει τον τρόπο που το σύστημα θα επικοινωνεί με τον χρήστη[17].

- Η αλληλεπίδραση ανάμεσα στον χρήστη και στην εφαρμογή ορίζεται εδώ

- Η αλληλεπίδραση του συστήματος καθορίζει τον τρόπο με τον οποίο το σύστημα θα μπορεί να επικοινωνεί με τον χρήστη, έτσι ώστε να ολοκληρώσει όποιους εκπαιδευτικούς στόχους έχουν οριστεί.

### 2.4.4 Απαιτήσεις της Υλοποίησης

Τα συστήματα έχουν υλοποιηθεί με τα εξής χαρακτηριστικά.

- Η Τεχνολογία Παραγωγής, που είναι η διαδικασία της παραγωγής και επικεντρώνεται στο κομμάτι της τεχνικής εκτέλεσης.
- Το Gamification, ή αλλιώς στα ελληνικά παιχνιδιοποίηση, εστιάζει σε αυτήν την ενότητα και αναζητά τους τρόπους έτσι ώστε η εφαρμογή να πληροί τους εκπαιδευτικούς στόχους.

Υπάρχουν αρκετές περιπτώσεις μέσα στα Components όπου υλοποιούνται διάφορες λειτουργίες της εφαρμογής. Το κάθε στοιχείο που δημιουργείται στο Unity και στο Unity Library, αποτελείται από όλες τις περιπτώσεις και τα components που χρησιμοποιούνται από το Unity Engine για τη δημιουργία ενός Game Object. Για την υλοποίηση και διαχείριση του περιεχομένου της επαυξημένης πραγματικότητας η πρόσθετη επέκταση του Vuforia Engine είναι αναγκαία για το Unity.

Ο χρήστης σε μια εφαρμογή επαυξημένης πραγματικότητας, όταν επιλέξει ένα αντικείμενο, θα μπορεί να μεγεθύνει, να μικρύνει, να περιστρέφει και να μετακινεί όπως θέλει. Αυτό θα μπορεί να το κάνει μόνο με όσα αντικείμενα εμφανίζονται στην οθόνη του, όπως μπορεί να αποτυπωθεί από την Εικόνα 2.3.



Εικόνα 2.4: Scale, Rotate, Locate Αντικειμένων

## 2.5 Επίλογος

Το αυξημένο ενδιαφέρον την τελευταία πενταετία για τις τεχνολογίες AR και VR ανάμεσα στους εκπαιδευτικούς, οφείλεται κυρίως στη ραγδαία εξέλιξη της τεχνολογίας. Ο αριθμός τέτοιου τύπου εφαρμογών είναι πλέον αρκετά μεγάλος. Για γίνει περισσότερο κατανοητή η τεχνική γλώσσα πίσω από το AR και VR, έχει κριθεί πως είναι αναγκαία να υλοποιηθεί μια νέα ταξινόμηση. Έχει περιγραφεί μια πιο σαφής μεθοδολογία και έχουν οριστεί νέα μέρη της καινούριας ταξινόμησης. Συγκεκριμένα, η επαυξημένη πραγματικότητα είναι μια τεχνολογία όπου μέσω της εικονικής πραγματικότητας, οι χρήστες θα μπορούν να αλληλεπιδρούν με πραγματικές καταστάσεις μέσω ενός ασφαλούς περιβάλλοντος. Σε αυτήν την εργασία η επαυξημένη πραγματικότητα χρησιμοποιήθηκε ως εργαλείο υποστήριξης για το μάθημα της ιστορίας.

## Κεφάλαιο 3ο: Τεχνολογίες – Γλώσσα Προγραμματισμού

### 3.1 Εισαγωγή

Στο παρόν κεφάλαιο θα παρουσιαστούν οι πλατφόρμες που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής. Αναφορικά τα περιβάλλοντα αυτά είναι το Unity, Blender 3D και το Visual Studio Code. Για την υλοποίηση του κώδικα χρησιμοποιήθηκε η γλώσσα προγραμματισμού C# και η δημιουργία των AR σκηνών έγινε μέσω Vuforia Engine. Για όλα τα παραπάνω θα αναφερθούν γενικές πληροφορίες και στα περιβάλλοντα τις απαιτήσεις hardware για να τα ανταπεξέλθει ένας ηλεκτρονικός υπολογιστής, τα οποία βασίζονται από επιστημονικά άρθρα.

### 3.2 Unity

Το Unity είναι ένα cross-platform game engine με ενσωματωμένο IDE και έχει ένα περιβάλλον ανάπτυξης από την Unity Game Engine Technologies, το οποίο είναι φιλικό προς τον χρήστη. Χρησιμοποιείται για να υλοποιήσει βιντεοπαιχνίδια για web-based εφαρμογές, κονσόλες, υπολογιστές και κινητές συσκευές. Η Unity Technologies είναι αυτή που προγραμματίσει το Unity 3D, για το runtime χρησιμοποιήθηκε η γλώσσα C++, ενώ για το Unity Scripting API η C#[18].

Η αρχική κυκλοφορία του Unity 3D έγινε τον Ιούνιο του 2005. Η συγκεκριμένη έκδοση υποστήριζε τις εξής πλατφόρμες: υπολογιστές, κινητά τηλέφωνα, κονσόλες και VR. Επιπλέον, μπορούσε να χρησιμοποιηθεί και για άλλες βιομηχανίες, για ταινίες, την αυτοκινητοβιομηχανία, την μηχανική και για την αρχιτεκτονική[19].

Από το Unity 2018 και στις επόμενες εκδόσεις του περιλαμβάνει τεχνολογίες μηχανικής μάθησης και το Magic Leap support. Επίσης, το 2018 ο κώδικας του Unity C# έγινε διαθέσιμος για reference-only License.

Σύμφωνα με το Unity, το δικό του engine χρησιμοποιείται στις μισές εφαρμογές παιχνιδιών για κινητά, για τις οποίες κάθε μήνα πραγματοποιούνται πάνω από τρία δις downloads και καθημερινά ξεκινάνε πάνω από 1000 νέα έργα. Από το Unity 2020 LTS είναι δυνατή η υποστήριξη των πλατφορμών Google AR core, Apple AR, Playstation VR, Windows MR, Steam VR, Gear VR, Vuforia και Facebook Game[20].

#### 3.2.1 Απαιτήσεις Συστήματος

Στο παρόν υποκεφάλαιο εμφανίζονται δύο πίνακες με τις ελάχιστες απαιτήσεις που πρέπει να έχουν οι ηλεκτρονικές συσκευές για τα Unity Editor και Unity Player, ώστε να μπορούν να ανταπεξέλθουν με το περιβάλλον του Unity. Και οι δύο πίνακες βασίζονται από το επίσημο documentation του Unity. Οι δύο πίνακες αναφέρονται στην έκδοση Unity 2022.3, καθώς με αυτήν την έκδοση υλοποιήθηκε η εφαρμογή[21].

Αρχικά, το Unity μπορεί να κατέβει στα Windows και MacOS, ενώ δεν μπορεί σε Tablet ή στο ChromeOS. Οι απαιτήσεις του hardware για το Unity Editor είναι οι εξής:

Πίνακας 3.1: Απαιτήσεις Unity Editor

OS	Έκδοση OS	CPU	Graphics API
Windows	Windows 7, 10, 11 (64-bit versions)	Αρχιτεκτονική X64 με SSE2 υποστήριξη	DX10,DX11, DX12-capable GPUs
macOS	Mojave 10.14+ (Intel processors) Big Sur 11.0 (Apple silicon-based processors)	Αρχιτεκτονική X64 με SSE2 υποστήριξη (Intel processors) Apple M1 και νεότερη (Apple silicon-based processors)	Meta-capable Intel και AMD GPUs
Linux	Ubuntu 20.4 and Ubuntu 18.4	Αρχιτεκτονική X64 με SSE2 υποστήριξη	OpenGL 3.2+ ή Vulkan-capable NVIDIA και AMD GPUs

Ο δεύτερος πίνακας αναφέρεται στις ελάχιστες απαιτήσεις για να μπορέσει κάποιος να φτιάξει και να τρέξει το Unity Player. Η πραγματική απόδοση και η ποιότητα του rendering βασίζεται κυρίως στην πολυπλοκότητα του project. Στον Πίνακα 3.2 είναι οι απαιτήσεις για να μπορέσει κάποιος να τρέξει το Unity Player.

Πίνακας 3.2: Απαιτήσεις Unity Player

OS	Version	CPU	Graphics API
Android	5.1 (API 22)+	ARMv7 with Neon Support or ARM64	OPENGL ES 3.0+, Vulkan
ChromeOS	R89+	ARMv7, ARM64, x86 and x86_64	OPENGL ES 3.0+, Vulkan
iOS	12+	A7 SoC+	Metal
tvOS	12+	A8 SoC+	Metal

### 3.3 Vuforia Engine

Το Vuforia SDK package είναι ένα δωρεάν λογισμικό και χρησιμοποιείται για την ανάπτυξη Unity 3D εφαρμογών για κινητά τηλέφωνα. Είναι ικανό να υποστηρίξει πολλές πλατφόρμες, περιλαμβάνοντας για συσκευές Android και iOS. Για να μπορέσει ο χρήστης και ο προγραμματιστής να χρησιμοποιήσουν την εφαρμογή, θα πρέπει να κατεβάσουν το license key από την επίσημη ιστοσελίδα της Vuforia. Οι δυνατότητες που έχει το πακέτο Vuforia SDK είναι να μπορεί σε άμεσο χρόνο να αναγνωρίζει εικόνες και να τις παρακολουθεί για όσο εμφανίζονται στην οθόνη του κινητού του χρήστη. Οποιοδήποτε αντικείμενο που είναι στον πραγματικό κόσμο μπορεί να συμπεριληφθεί από μια εικόνα, έτσι όταν η εφαρμογή αναγνωρίσει την συγκεκριμένη εικόνα, τότε πάνω στην εικόνα θα εμφανιστεί το αντικείμενο που έχει συμπεριληφθεί[22]. Το Vuforia SDK μπορεί να υποστηρίξει 2D

και 3D αντικείμενα περιλαμβάνοντας configurations πολλαπλών στόχων, εικόνες με διάφορα σύμβολα και ετικέτες ενός πλαισίου[23].

### 3.4 Visual Studio Code

Το Visual Studio Code είναι ένα IDE που μπορεί να υποστηρίξει πολλαπλές πλατφόρμες, και είναι ικανό να υποστηρίξει αρκετές γλώσσες προγραμματισμού. Μερικές από αυτές είναι HTML, CSS, C#, Java, Python, Javascript, Typescript, PHP κτλ. και διαθέτει μια αρκετά ισχυρή αρχιτεκτονική[24]. Έχει σχεδιαστεί με τέτοιο τρόπο ώστε επιτρέπει στους προγραμματιστές να δημιουργούν και να διαχειρίζονται τον κώδικα τους με μεγάλη ευκολία. Η δυνατότητα λειτουργίας ανάμεσα στις διαφορετικές πλατφόρμες το καθιστά ως μια από τις ιδανικές επιλογές για την ανάπτυξη έργων που απαιτείται η χρήση παραπάνω από μια γλώσσες προγραμματισμού.

Τα τελευταία χρόνια έχει αρχίσει να θεωρείται ως ένας από τους δημοφιλέστερους επεξεργαστές για προγραμματιστές, χάρη στην μεγάλη συμβατότητα, την υψηλή ευελιξία του και το μεγάλο σύστημα επεκτάσεών του. Ο VS Code δίνει την δυνατότητα στον προγραμματιστή να δημιουργήσει την δικιά του επέκταση ή να χρησιμοποιήσει επεκτάσεις άλλων προγραμματιστών, επιτυγχάνοντας με αυτόν τον τρόπο την δημιουργία μιας μεγάλης κοινότητας ανοιχτού κώδικα. Μια επέκταση είναι ένα πρόγραμμα στο VS Code, το οποίο ενεργοποιείται όταν πληρείται μια καθορισμένη συνθήκη. Αυτές οι συνθήκες μπορεί να περιλαμβάνουν είτε το πάτημα συγκεκριμένων πλήκτρων στο πληκτρολόγιο, είτε να πληκτρολογηθεί μια από τις προκαθορισμένες λέξεις-κλειδιά της επέκτασης[25].

Με τη συνεχή ανάπτυξη νέων επεκτάσεων, ο VS Code θα συνεχίζει να εξελίσσεται και θα μπορεί να προσαρμόζεται στις νέες απαιτήσεις που προκύπτουν με την πάροδο του χρόνου, παρέχοντας προηγμένες δυνατότητες και εργαλεία που διευκολύνουν την διαδικασία ανάπτυξης λογισμικού.

### 3.5 C# Programming Language

Η C# (C Sharp) είναι μια γλώσσα προγραμματισμού, η οποία αναπτύχθηκε το έτος 2000 από τον Anders Hejlsberg, που είναι μέχρι και σήμερα εργαζόμενος της Microsoft, και μπορεί να τρέξει στο framework .NET και στο Visual Studio και στην πλατφόρμα open-source. Η C# χρησιμοποιείται πιο πολύ για να υλοποιήσει εφαρμογές που βασίζονται στον ιστό, σε υπολογιστή, κινητά, και σε άλλες πολλές εφαρμογές. Για την σύνδεση του με το Unity, είναι να εισαχθεί ένα C# script στα παράθυρα hierarchy και inspector του Unity 3D IDE [26][27][28].

Η βασική σύνταξη της γλώσσας C# είναι ίδια όπως με τις άλλες γλώσσες, τις C, C++ και Java. Το τέλος μιας command line γίνεται με την εισαγωγή του ελληνικού ερωτηματικού και τα brackets χρησιμοποιούνται κυρίως για να ομαδοποιήσουν δηλώσεις. Το Unity, για τις περιπτώσεις του scripting χρησιμοποιεί όλες τις λειτουργίες της γλώσσας C#, όπως είναι οι μεταβλητές, οι τύποι δεδομένων (integers, strings, arrays), εντολές εισόδου-εξόδου, μεθόδους και return, συναρτήσεις, κλάσεις, εντολές ελέγχου (if-switch) και επανάληψης (for-while). Η C# μπορεί να υποστηρίξει τις κλάσεις με properties, οι οποίες μπορούν να εφαρμοστούν στις συναρτήσεις getter και setter απλές λειτουργίες πρόσβασης με πεδία υποστήριξης[29]. Οι κανόνες και τα χαρακτηριστικά του namespace είναι ίδια με αυτά του πακέτου, και προσφέρει επίσης το ίδιο code separation με το πακέτο της Java και το namespace της C++. Η εισαγωγή namespaces μπορεί να γίνει και συντακτικά.

### 3.6 Blender 3D

Η εφαρμογή Blender σχεδιάστηκε για να παράγει τρισδιάστατα μοντέλα, κινούμενα σχέδια και γραφικά υψηλής ανάλυσης. Το Blender Foundation διατηρεί open-source λογισμικό, το οποίο έχει

αρκετούς χρήστες που το χρησιμοποιούν, και όσο περνάει ο καιρός αυξάνονται όλο και περισσότερο, έχοντας κάθε χρόνο περισσότερες από 3.4 εκατομμύρια λήψεις. Το NeoGeo είναι στούντιο κινουμένων σχεδίων και είναι αυτό που έχει αρχίσει να υλοποιεί το Blender, και η αρχική του έκδοση ήταν στις 2 Ιανουαρίου 1994[30]. Είναι δωρεάν, όπως είναι και όλα τα open-source λογισμικά, και οποιοσδήποτε μπορεί να το κατεβάσει για να αναπτύξει επιχειρηματικά έργα. Το Blender είναι ένα πλήρες πρόγραμμα, που μπορεί να ολοκληρώσει πολλές εργασίες[31]. Μερικά από τα χαρακτηριστικά του Blender είναι τα εξής:

- Έχει την δυνατότητα για rigging και animation, έτσι ώστε τα αντικείμενα να μπορούν να μετακινούνται.
- Το texture tool του Blender, χρησιμοποιείται για να βάψεις επιφάνειες αντικειμένων.
- Διαθέτει και κάμερα φωτισμού για την λήψη animation και φωτογραφιών[31].

Στον παρακάτω πίνακα αναφέρονται οι πιο πρόσφατες εκδόσεις του Blender μέσα στα τελευταία δύο χρόνια, μαζί με την ημερομηνία έκδοσης και τα καινούρια Features[32].

Πίνακας 3.3: Εκδόσεις Blender

Έκδοση	Ημερομηνία Έκδοσης
4.2	16/08/2024
4.1	26/03/2024
4.0	14/11/2023
3.6 LTS	27/06/2023
3.5	29/03/2023

Στον Πίνακα 3.4 αναφέρονται τα ιδανικά εξαρτήματα που πρέπει να έχει ο υπολογιστής για κάθε λειτουργικό σύστημα για να έχει μεγαλύτερη απόδοση. Να σημειωθεί πως αυτά δεν είναι τα ελάχιστα αλλά τα προτεινόμενα hardware για την καλύτερή του λειτουργία[32].

Πίνακας 3.4: Απαιτήσεις Hardware - Blender

OS	CPU	RAM	GPU
Windows 10-11	8 cores	32GB	8GB VRAM
macOS 14 (Sonoma)	Apple Silicon	32GB	-
Linux	8 cores	32GB	8GB VRAM

Η εφαρμογή Blender χρησιμοποιείται για τις εξής περιπτώσεις[33]:

- Σχεδίαση Logo
- VFX
- Δημιουργία ρεαλιστικών σκηνών
- 3D-2D animation
- Motion Graphics
- Ενσωματωμένο πρόγραμμα επεξεργασίας βίντεο

- Αρχιτεκτονικός σχεδιασμός
- Υλοποίηση 3D assets και environments για βιντεοπαιχνίδια
- Sculpting για οπτικοποίηση και Rendering.

### **3.7 Επίλογος**

Σε αυτήν την ενότητα αναφέρθηκαν μερικές πληροφορίες για τις τεχνολογίες (Vuforia Engine), τα προγραμματιστικά περιβάλλοντα (Unity, Blender, VS Code) και η γλώσσα προγραμματισμού (C#) που χρησιμοποιήθηκαν για την δημιουργία της εφαρμογής, καθώς και οι απαιτήσεις της συσκευής ώστε να είναι ικανή να χρησιμοποιήσει τα συγκεκριμένα προγράμματα.



## Κεφάλαιο 4ο: Δημιουργία-Κίνηση 3D αντικειμένων στο Blender

### 4.1 Εισαγωγή

Στο συγκεκριμένο κεφάλαιο θα παρουσιαστεί η δημιουργία των τρισδιάστατων αντικειμένων καθώς και τα animation. Για την υλοποίηση των αντικειμένων χρησιμοποιήθηκε το περιβάλλον Blender. Τα αντικείμενα που υλοποιήθηκαν είναι τα εξής:

- Περιοχές: Σαλαμίνα και Θερμοπύλες.
- Χαρακτήρες: Έλληνες και Πέρσες στρατιώτες, Λεωνίδα, Βασιλιάς Ξέρξης, Εφιάλτης, Θεμιστοκλής και Ευρυβιάδης.
- Αντικείμενα Περσικού στρατού: Αδύναμοι, Δυνατοί, Τοξότες.
- Στόλοι: Ελληνική και Περσική τριήρης.

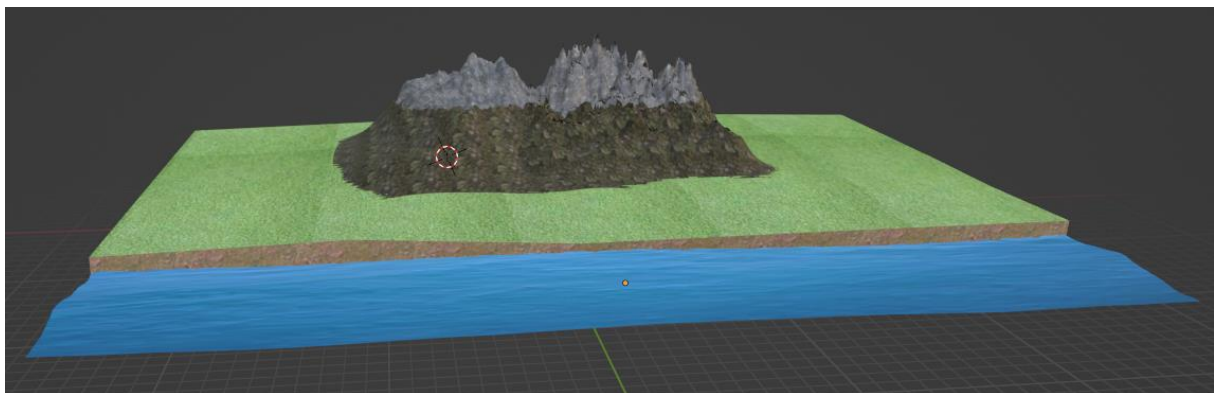
Στα παρακάτω υποκεφάλαια θα παρουσιάζεται ξεχωριστά το κάθε αντικείμενο για τον τρόπο που υλοποιήθηκε αλλά θα αναφέρονται και τα animation που θα κάνουν τα αντικείμενα στην εφαρμογή. Επιπλέον, θα αναφέρεται σε ποια από τις δύο αναπαραστάσεις θα χρησιμοποιούνται τα αντικείμενα και τα animation του καθενός, καθώς κάποιοι από τους χαρακτήρες θα χρησιμοποιηθούν και στις δύο αναπαραστάσεις, αλλά θα εκτελούνται διαφορετικά animation. Στο τέλος επισυνάπτονται και τα image textures που χρησιμοποιήθηκαν σε κάποια από τα αντικείμενα.

### 4.2 Περιοχές

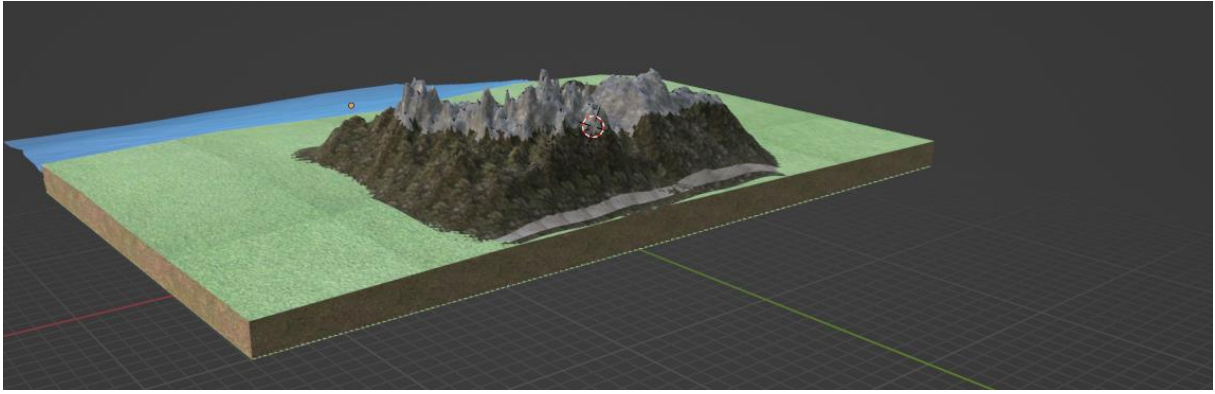
Στο υποκεφάλαιο 4.2 θα παρουσιαστεί αναλυτικά ο τρόπος που υλοποιήθηκαν οι δύο περιοχές που πρόκειται να γίνουν οι μάχες και πως είναι η τελική τους εμφάνιση στην εφαρμογή.

#### 4.2.1 Θερμοπύλες

Για την υλοποίηση της περιοχής των Θερμοπυλών (Εικόνα 4.1) τα μόνα αντικείμενα που χρειάστηκαν να σχεδιαστούν είναι το βουνό που βρίσκεται στην περιοχή, η θάλασσα η οποία βρίσκεται σε αρκετά κοντινή απόσταση από το βουνό, ο διάδρομος που χωρίζει το βουνό και την θάλασσα και είναι το μέρος που βρίσκονταν ο Ελληνικός στρατός στη μάχη, και τέλος το μονοπάτι (Εικόνα 4.2) που βρίσκεται πάνω στο βουνό το οποίο το έδειξε ο Εφιάλτης στους Πέρσες για να περάσουν από την αντίθετη πλευρά και να περικυκλώσουν τους Έλληνες.

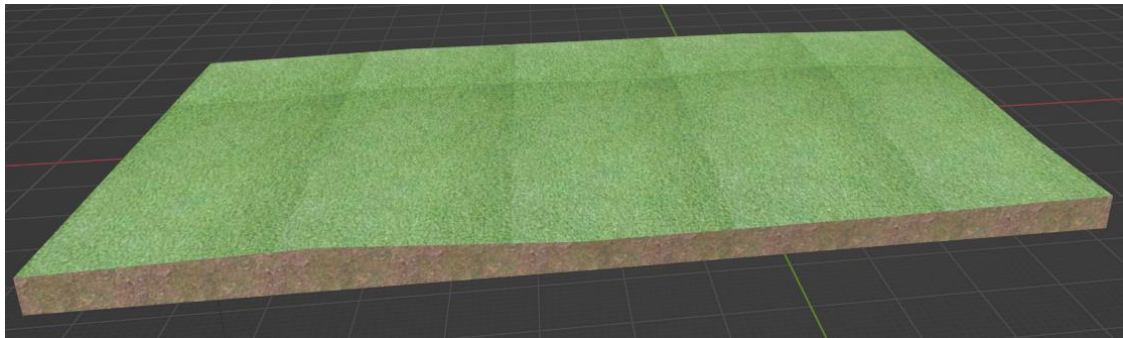


Εικόνα 4.1: Θερμοπύλες



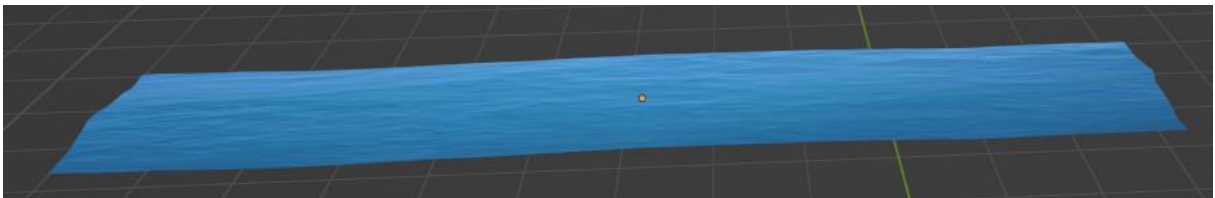
Εικόνα 4.2: Μονοπάτι

Για την υλοποίηση του εδάφους, χρησιμοποιήθηκε ένας κύβος, όπου με το scale πήρε την παραπάνω μορφή και στο τέλος προστέθηκε ένα image texture με μια εικόνα από grass texture ώστε να φαίνεται πως το μέρος είχε βλάστηση, αντί για ξηρασία ή χιόνι, καθώς σύμφωνα με τους ιστορικούς, η μάχη πραγματοποιήθηκε τέλη Αυγούστου με αρχές Σεπτεμβρη. Επιπλέον στις πλάγιες επιφάνειες του κύβου χρησιμοποιήθηκε το mud texture μέσω του Edit Mode που υπάρχει στο Blender και επιλέχθηκαν οι επιφάνειες που είναι κάθετες, όπως φαίνεται στην Εικόνα 4.3.



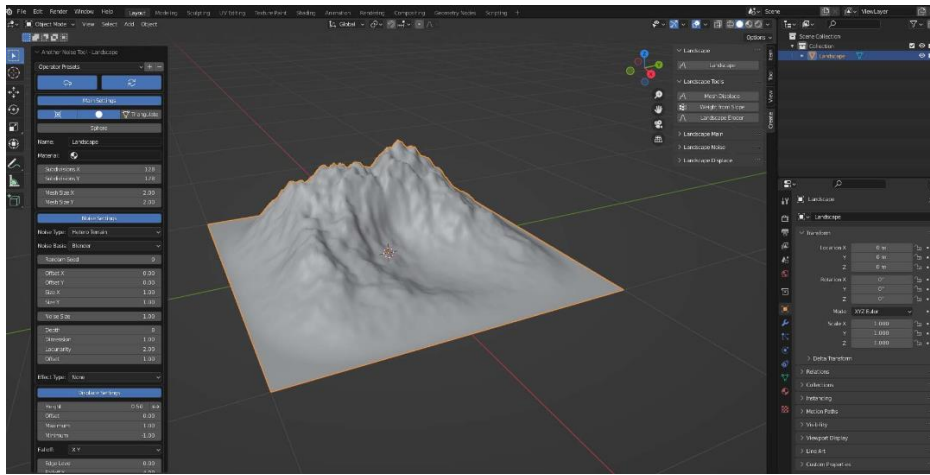
Εικόνα 4.3: Έδαφος

Στην περίπτωση της θάλασσας που παρουσιάζεται στην Εικόνα 4.4, το Blender έχει την δυνατότητα ένα plane object να το παραμορφώσεις για να φαίνεται πως έχει κύματα, ο τρόπος αυτός φαίνεται στην παρακάτω φωτογραφία. Επιπλέον για image texture προστέθηκε ένα sea texture και επεξεργάστηκε κατάλληλα ώστε να μην υπάρχουν ανομοιόμορφα σημεία, εξαιτίας της παραμόρφωσης που αναφέρθηκε πάνω για να δημιουργηθούν τα κύματα.

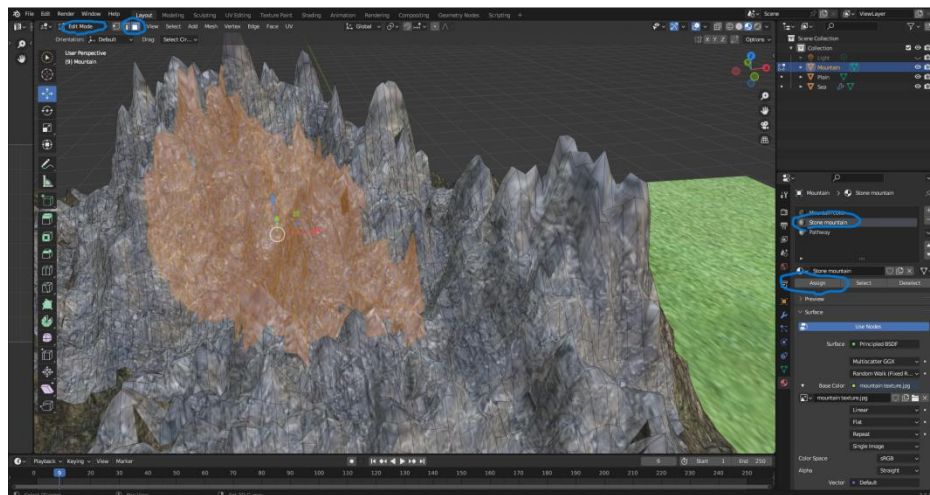


Εικόνα 4.4: Θάλασσα

Τέλος, για την υλοποίηση του βουνού και του μονοπατιού, χρησιμοποιήθηκε το Landscape object, στο οποίο μπορείς να επιλέξεις πως θα φαίνεται το terrain του και στη συνέχεια να αυξήσεις ή να χαμηλώσεις το ύψος των κορυφών. Στο συγκεκριμένο αντικείμενο εισάχθηκαν τα χαρακτηριστικά της Εικόνας 4.5. Και σε αυτό το αντικείμενο χρησιμοποιήθηκε το image texture, αλλά υπάρχουν τρία διαφορετικά, το ένα και το κύριο είναι το mountain texture, στο οποίο φαίνεται η βλάστηση που υπάρχουν πάνω στο βουνό, το stone texture, που δείχνουν τις κορυφές και τέλος το texture της άμμου όπου χρησιμοποιείται για να υλοποιηθεί το μονοπάτι του βουνού. Το αρχικό image texture του αντικειμένου είναι το πρώτο που αναφέρθηκε και τα άλλα δύο προστέθηκαν αργότερα μέσω του Edit mode που υπάρχει στο Blender, όπου μπορείς να επιλέξεις συγκεκριμένα σημεία του αντικειμένου και να επιλέξεις άλλα textures, ακόμη και να τα διαγράψεις. Στην Εικόνα 4.6 δείχνει τον τρόπο που μπορείς να αλλάξεις το texture μόνο σε ορισμένα σημεία του, για το συγκεκριμένο αντικείμενο να σημειωθεί πως λόγω της παραμόρφωσης που είχε ήδη, υπάρχουν πολλά σημεία (points) και επιφάνειες και δεν χρειάστηκε να γίνει η χρήση του “cut” για τη δημιουργία νέων επιφανειών.



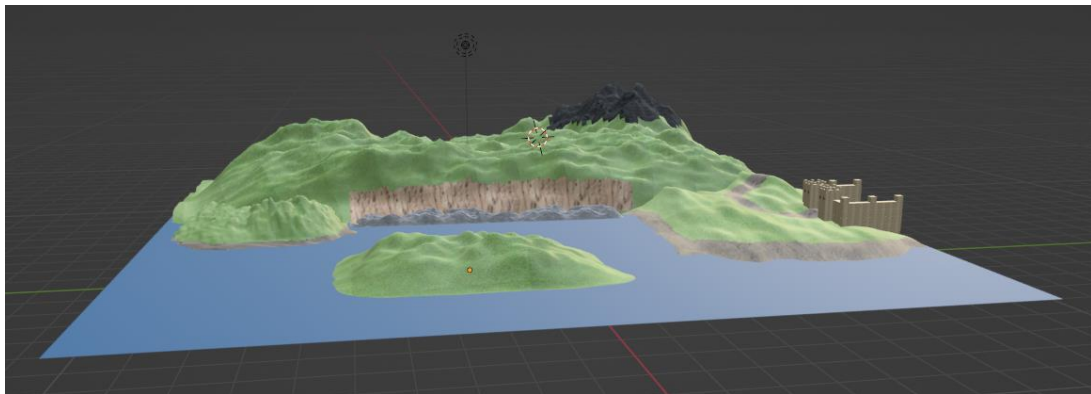
Εικόνα 4.5: Χαρακτηριστικά Landscape Object



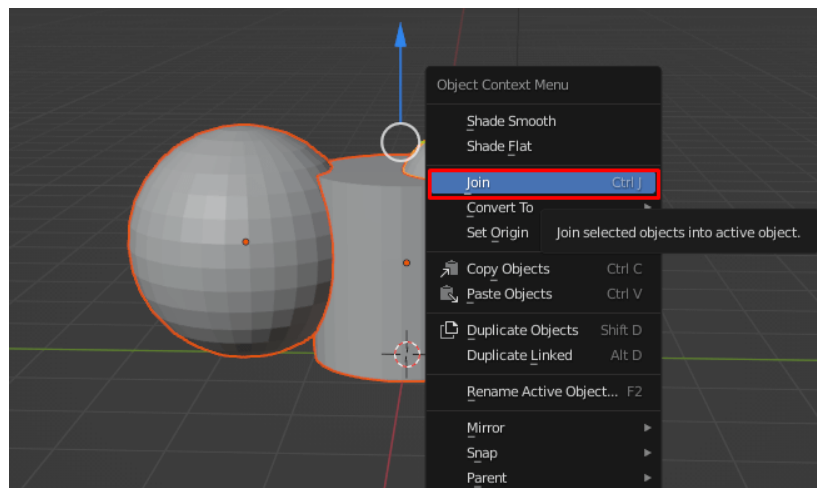
Εικόνα 4.6: Αλλαγή Image Texture

### 4.2.2 Σαλαμίνα

Για την περιοχή της Σαλαμίνας (Εικόνα 4.7), λόγω του τρόπου που είναι διαμορφωμένη η περιοχή χρειάστηκαν παραπάνω αντικείμενα. Για την υλοποίηση του χρησιμοποιήθηκαν τρία διαφορετικά Landscape objects με διαφορετικά terrain και ύψη των κορυφών, στα οποία τα δύο συγχωνεύτηκαν μεταξύ τους επιλέγοντας το “Join” της εικόνας 4.8, αφότου επιλέχθηκαν τα αντικείμενα, ή με τα πλήκτρα CTRL + J, ενώ το τρίτο αντικείμενο χρησιμοποιήθηκε για να δείχνει το νησί Σαλαμίνα που ακριβώς βρίσκεται. Επιπλέον, χρησιμοποιήθηκε ένα plane object για την δημιουργία της θάλασσας και τέλος πάνω στα δύο landscape objects που συγχωνεύτηκαν, προστέθηκαν πολλαπλές πέτρες, σε σημείο που υπάρχει γκρεμός, η πόλη της Αθήνας, για την οποία χρησιμοποιήθηκαν πολλαπλά αντικείμενα και image textures για να γίνει ο τείχος, η πύλη της εισόδου και μερικά από τα σπίτια.



Εικόνα 4.7: Σαλαμίνα



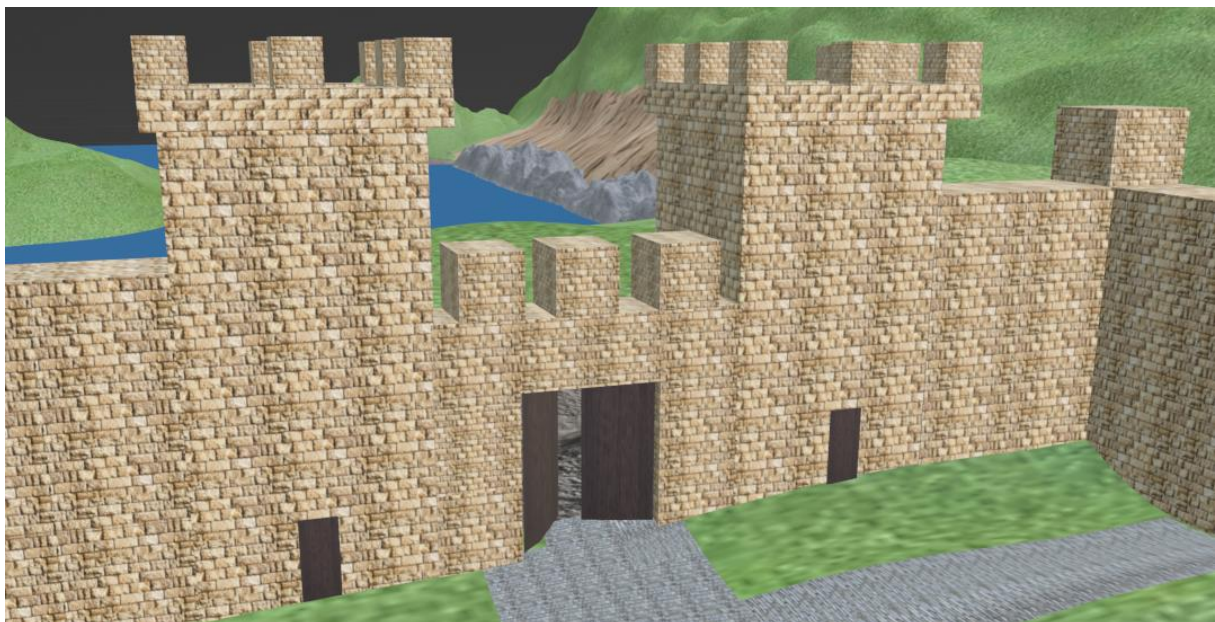
Εικόνα 4.8: Join

Τα landscape αντικείμενα έχουν ένα κοινό μεταξύ τους (τα δύο που ενώθηκαν και το νησί της Σαλαμίνας), είναι το κύριο image texture όπου χρησιμοποιήθηκε το Grass texture και επεξεργάστηκε κατάλληλα στο Edit mode, καθώς λόγω της διαμόρφωσης δεν εμφανιζόταν αρκετά καλά η εικόνα του. Η μόνη διαφορά που έχουν μεταξύ τους αυτά τα δύο, πέρα από τα terrain που χρησιμοποιήθηκαν, είναι ότι στα δύο ενωμένα χρησιμοποιήθηκαν επιπλέον image textures, τα οποία φαίνονται και στην

Εικόνα 4.7. Τα συγκεκριμένα texture χρησιμοποιούνται για το χωμάτινο δρόμο που ακολούθησαν οι Πέρσες μέχρι την Αθήνα, την άμμο που βρίσκεται στα παράλια μέρη της περιοχής (sand texture), τις κορυφές του όρους Αιγάλεω (stone texture) και στο σημείο που υπάρχει γκρεμός έχει προστεθεί το cliff texture και πολλές ενωμένες πέτρες πάνω στην ακτή.

Για την θάλασσα, επειδή το σημείο αυτό είναι κόλπος, η θάλασσα είναι πιο ήρεμη από την περίπτωση των Θερμοπυλών που αναφέρθηκε πιο πάνω και για περισσότερη ευκολία στο animation των δύο στόλων που θα βρίσκονται στην αναπαράσταση, χρησιμοποιήθηκε μόνο το sea texture, χωρίς τα κύματα.

Τέλος, στην περίπτωση της αρχαίας πόλης της Αθήνας, έχουν προστεθεί αρκετοί κύβοι για να φτιαχτούν τα γύρω τείχη μαζί με τον διάδρομο που θα βρίσκονται οι τοξότες και τους πύργους στην είσοδο της πόλης, όπου όλα αυτά στο τέλος συγχωνεύτηκαν, όπως φαίνεται στην Εικόνα 4.9, και χρησιμοποιούν το wall texture. Ανάμεσα στους ψηλούς πύργους προστέθηκαν δύο κύβοι, οι οποίοι μορφοποιήθηκαν ως δύο ψηλές πύλες εισόδου και τοποθετήθηκαν με τρόπο που να φαίνεται πως η είσοδος ήταν ανοιχτή, καθώς είχε ερημωθεί η πόλη και δεν υπήρχε κανένας.



Εικόνα 4.9: Τείχη Αρχαίας Αθήνας

Στο εσωτερικό μέρος της πόλης, προστέθηκαν μερικά σπίτια, παρόμοια της Εικόνας 4.10, τα οποία είναι ίδια διαμορφωμένα, με ένα αντικείμενο που είναι το τοίχος του σπιτιού, ένα για το παράθυρο, ένα για την πόρτα και ένα για την σκεπή, τα οποία συγχωνεύτηκαν για να δημιουργηθεί το σπίτι. Στην Εικόνα 4.11 φαίνεται το μονοπάτι που υπάρχει μέσα στην πόλη και ενώνει την είσοδο της πόλης με τα σπίτια.



Εικόνα 4.10: Σπίτι στην Αρχαία Αθήνα



Εικόνα 4.11: Εσωτερικό Μονοπάτι

### 4.3 Χαρακτήρες

Σε αυτήν υποενότητα περιλαμβάνει όλα τα αντικείμενα-χαρακτήρες της αναπαράστασης. Αναφέρθηκαν πιο πάνω οι χαρακτήρες που πρόκειται να χρησιμοποιηθούν και για κάθε αντικείμενο θα αναφερθεί σε ποια από τις αναπαραστάσεις θα βρίσκονται (κάποια αντικείμενα θα χρησιμοποιηθούν και στις δύο), την ένδυση που χρησιμοποιήθηκε και τα animation που θα έχει το καθένα. Επιπλέον, θα σημειωθεί και σε ποια αναπαράσταση θα εμφανιστεί το κάθε animation.

Τέλος, να σημειωθεί πως σε αντίθεση με τις περιοχές και τους στόλους, ο τρόπος δημιουργίας όλων των χαρακτήρων έγινε με έτοιμα αντικείμενα από ιστοσελίδες που περιέχουν δωρεάν περιεχόμενο τρισδιάστατων αντικειμένων (Sketchfab, Free 3D) και μετά στην εφαρμογή Blender γινόταν συνδυασμός αυτών των αντικειμένων (περικεφαλαίας, σπαθιού, ασπίδας κτλ) για να φτιάξουν την τελική μορφή του χαρακτήρα.

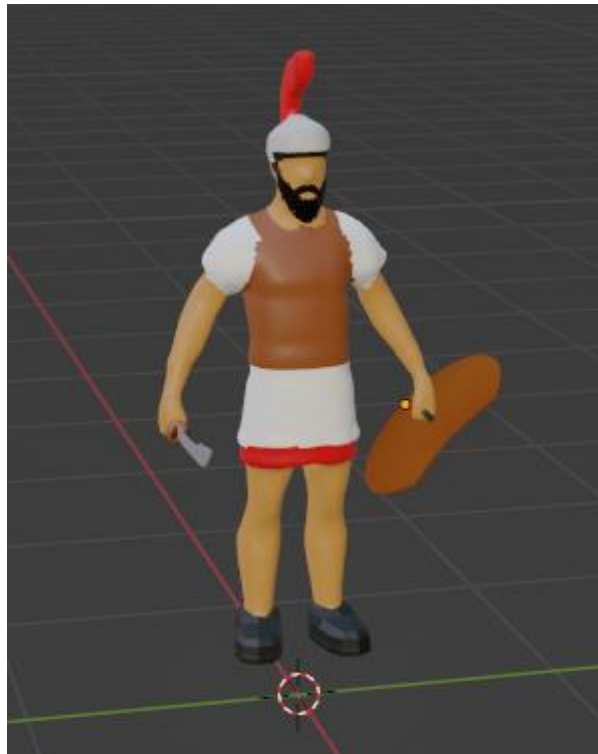
#### 4.3.1 Ελληνικός και Περσικός στρατός

Οι στρατιώτες και των δύο μετώπων έχουν κοινά αντικείμενα, όπως φαίνεται στις δύο παρακάτω εικόνες (4.12 και 4.13). Αυτά τα αντικείμενα είναι το σπαθί, η περικεφαλαία, και η ασπίδα. Τα μόνα

που αλλάζουν σε αυτά είναι η εμφάνισή τους και το χρώμα του δέρματος (οι Πέρσες είναι πιο έγχρωμοι λόγω του ζεστού κλίματος που επικρατεί στην Περσία), για παράδειγμα το σπαθί και η ασπίδα των περσών είναι πιο μεγάλα και η περικεφαλαία των Ελλήνων καλύπτει όλο το πρόσωπο του στρατιώτη και περιέχει και το κόκκινο τρίχωμα στο πάνω μέρος της περικεφαλαίας. Άλλες διαφορές που υπάρχουν είναι στην στολή, καθώς οι Έλληνες έχουν προστατευτικό στα μπράτσα και στην περιοχή της λεκάνης μέχρι τα γόνατα, ενώ οι Πέρσες έχουν μια ολόσωμη στολή που καλύπτει από τους ώμους μέχρι το μηρό.



Εικόνα 4.12: Έλληνας Στρατιώτης



Εικόνα 4.13: Πέρσης Στρατιώτης

Στο animation, το μόνο κοινό που έχουν και τα δύο στρατεύματα είναι την στιγμή που πολεμάνε μεταξύ τους (Εικόνα 4.14), δηλαδή θα φαίνεται όλη η κίνηση της επίθεσης και της μετακίνησης του σπαθιού και της ασπίδας, το οποίο αυτό συμβαίνει στην μάχη των Θερμοπυλών.





Εικόνα 4.14: Animation Επίθεσης Ελλήνων – Περσών

Ο ελληνικός στρατός έχει ακόμη δύο animation (Εικόνα 4.15 και Εικόνα 4.16). Και τα δύο εμφανίζονται στο τέλος της μάχης Θερμοπυλών, όπου δείχνει τους στρατιώτες που τράπηκαν σε φυγή, δηλαδή φαίνονται να τρέχουν, και τους υπόλοιπους που συνέχισαν την μάχη όπου στο τέλος σκοτώθηκαν.



Εικόνα 4.15: Animation Run Ελλήνων



Εικόνα 4.16: Animation Death Ελλήνων

## Κεφάλαιο 4

Ο περσικός στρατός έχει ακόμη ένα animation, το οποίο είναι το περπάτημα (Εικόνα 4.17). Το συγκεκριμένο εμφανίζεται και στις δύο αναπαραστάσεις της εφαρμογής, την στιγμή που φτάνουν στις Θερμοπύλες από την Περσία και στην Αθήνα.



Εικόνα 4.17: Animation Walk Περσών

### 4.3.2 Λεωνίδας

Ο αρχηγός των Ελλήνων στη Μάχη των Θερμοπυλών, Λεωνίδας, εμφανίζεται μόνο στην αναπαράσταση των Θερμοπυλών. Τα αντικείμενα που χρησιμοποιήθηκαν παρουσιάζονται στην Εικόνα 4.18 και είναι παρόμοια με του ελληνικού στρατού, με τις μόνες διαφορές που έχει είναι ότι η στολή είναι κίτρινου χρώματος, έχει προστατευτικό στο στήθος, στους καρπούς των χεριών και στην κνήμη των ποδιών, ενώ δεν έχει στα μπράτσα όπως έχουν οι στρατιώτες, και η περικεφαλαία του αφήνει ακάλυπτο το πρόσωπο του.



Εικόνα 4.18: Λεωνίδας

Ο χαρακτήρας Λεωνίδας έχει τρία animation (Εικόνες 4.19, 4.20, 4.21), τα οποία όλα θα χρησιμοποιηθούν στην ίδια αναπαράσταση, σε αυτήν που αναφέρθηκε πιο πάνω. Το πρώτο που θα εμφανιστεί στην αναπαράσταση είναι η στιγμή που θα μιλάει μαζί με τον Βασιλιά Ξέρξη, πριν ξεκινήσει η μάχη.



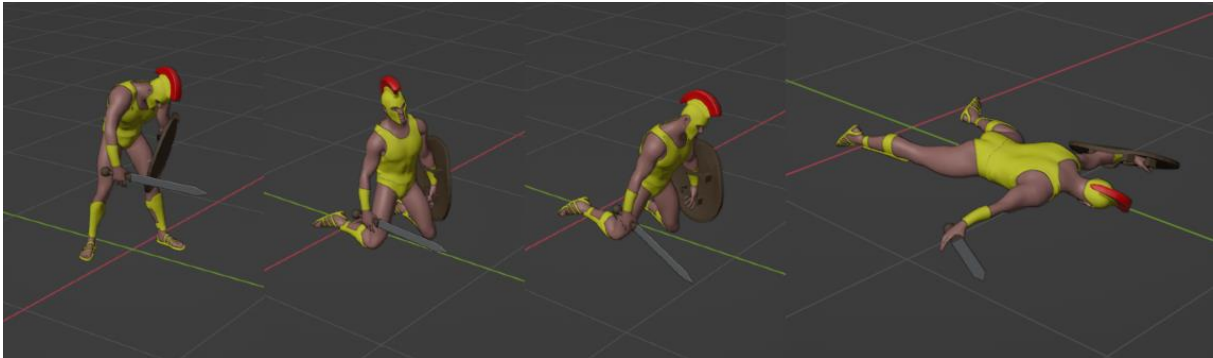
Εικόνα 4.19: Animation Talk Λεωνίδα

Στην διάρκεια της μάχης θα εμφανίζεται ο Λεωνίδας στην πρώτη γραμμή να πολεμάει του Πέρσες στρατιώτες. Η κίνηση θα είναι ίδια όπως αυτή των Περσών και των Ελλήνων στο animation που πολεμάνε.



Εικόνα 4.20: Animation Attack Λεωνίδα

Το τελευταίο animation του Λεωνίδα, είναι το ίδιο με αυτό των Ελλήνων στρατιωτών που είχαν μείνει μέχρι το τέλος της μάχης, δηλαδή θα δείχνει την στιγμή που θα σκοτώνεται και θα πέφτει μαχόμενος.



Εικόνα 4.21: Animation Death Λεωνίδα

### 4.3.3 Ξέρξης

Ο Βασιλιάς των Περσών, Ξέρξης είναι ο δεύτερος χαρακτήρας που θα εμφανιστεί και στις δύο αναπαραστάσεις της εφαρμογής. Για την υλοποίηση του συγκεκριμένου χαρακτήρα, όπως παρουσιάζεται στην Εικόνα 4.22, χρησιμοποιήθηκε μια ρόμπα χρώματος μωβ, μελαχρινά μαλλιά και γένια, σανδάλια και μια χρυσή κορώνα, για να φαίνεται στην αναπαράσταση πως είναι βασιλιάς.



Εικόνα 4.22: Ξέρξης

## Κεφάλαιο 4

Όπως ο Λεωνίδας, και ο Βασιλιάς Ξέρξης έχει τρία animation, ωστόσο είναι διαφορετικά από τον αρχηγό του ελληνικού στρατού. Το πρώτο animation είναι ίδιο με τους Πέρσες στρατιώτες και εμφανίζεται επίσης και στις δύο αναπαραστάσεις, η στιγμή που περπατάει μαζί τους ως τις Θερμοπύλες και έως την πόλη της Αθήνας.



Εικόνα 4.23: Animation Walk Ξέρξη

Το δεύτερο animation είναι ίδιο με του Λεωνίδα και θα χρησιμοποιηθεί μόνο στην αναπαράσταση των Θερμοπυλών, την στιγμή που μιλάνε μεταξύ τους και του ζητάει ο Ξέρξης να παραδοθούν. Συγκρίνοντας την Εικόνα 4.19 και την Εικόνα 4.24 και οι δύο χαρακτήρες κάνουν διαφορετικές κινήσεις την στιγμή που μιλάνε.



Εικόνα 4.24: Animation Talk Ξέρξη

Το τελευταίο animation του Ξέρξη είναι η κίνηση που κάνει για να παρακολουθεί την μάχη. Όπως το πρώτο animation του Ξέρξη (Walk), και αυτό χρησιμοποιείται και στις δύο αναπαραστάσεις. Λόγω της τεράστιας απόστασης που έχει και από τις δύο σκηνές μάχης η κίνηση που φαίνεται να κάνει στην Εικόνα 4.25 είναι να προσπαθεί να δει καλύτερα την εξέλιξη και των δύο μαχών.



Εικόνα 4.25: Animation Look Ξέρξη

#### 4.3.4 Εφιάλτης

Τον Εφιάλτη (Εικόνα 4.26) θα τον συναντήσουμε μόνο στην αναπαράσταση της Μάχης των Θερμοπυλών. Τα μόνα αντικείμενα που χρειάστηκαν για την εμφάνισή του είναι μόνο η άσπρη ρόμπα και τα σανδάλια που φορούσε.



Εικόνα 4.26: Εφιάλτης

Ο Εφιάλτης εμφανίζεται μόνο σε ένα σημείο ολόκληρης της αναπαράστασης, στη διάρκεια της μάχης που δείχνει αρχικά στον Ξέρξη το μονοπάτι που ξέρει μέσα από το βουνό και αμέσως μετά να περπατάει και να οδηγάει τον περσικό στρατό από την άλλη πλευρά. Τα animation που έχει είναι δύο, τα οποία παρουσιάζονται στις Εικόνες 4.27 και 4.28, και είναι αρχικά την επίδειξη του μονοπατιού που υπάρχει στο βουνό και στη συνέχεια την πορεία του ως εκείνο το σημείο.



Εικόνα 4.27: Animation Pointing Εφιάλτη



Εικόνα 4.28: Animation Walk Εφιάλτη

### 4.3.5 Ευρυβιάδης

Ο Ευρυβιάδης είναι ο ένας από τους δύο αρχηγούς του στόλου, και θα εμφανιστεί μόνο στην Ναυμαχία της Σαλαμίνας. Ως αρχηγός από την αρχαία Σπάρτη, η στολή που φοράει (Εικόνα 4.29) είναι ίδια με αυτή του Λεωνίδα (χωρίς τα προστατευτικά στους καρπούς των χεριών), έχοντας επιπλέον μια κόκκινη φουστανέλα μέσα από το προστατευτικό του στήθους.



Εικόνα 4.29: Ευρυβιάδης

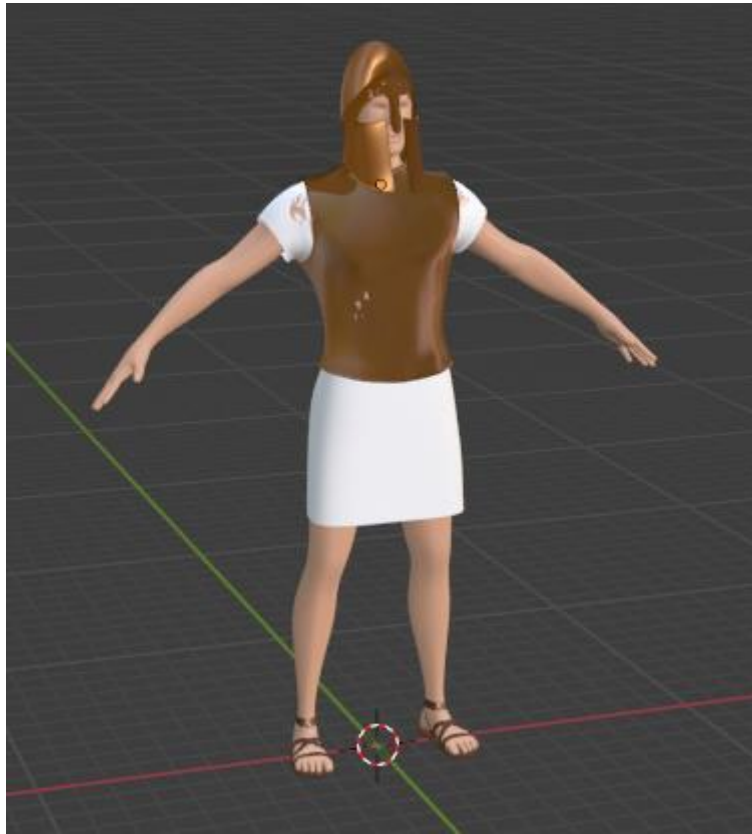
Για τον Ευρυβιάδη, έχει υλοποιηθεί μόνο ένα animation, το οποίο παρουσιάζεται στην παρακάτω εικόνα, καθώς θα εμφανιστεί μόνο σε μια σκηνή της αναπαράστασης, το οποίο εμφανίζεται μόνο στην αναπαράσταση της Σαλαμίνας και είναι την στιγμή που έχει έντονη λογομαχία με τον δεύτερο αρχηγό του στόλου για την τοποθέτηση των πλοίων.



Εικόνα 4.30: Animation Ευρυβιάδη

#### 4.3.6 Θεμιστοκλής

Ο Θεμιστοκλής είναι ο δεύτερος αρχηγός του στόλου και αυτός επίσης θα εμφανιστεί μόνο στην ναυμαχία της Σαλαμίνας. Το ντύσιμό του φαίνεται στην Εικόνα 4.31. Σε σύγκριση με του Ευρυβιάδη, ο Θεμιστοκλής από την Αθήνα έχει πιο απλή ενδυμασία έχοντας μόνο απλά σανδάλια, μια άσπρη φουστανέλα και μια χάλκινη περικεφαλαία χωρίς να έχει πάνω τρίχωμα όπως αυτό των Σπαρτιατών.



Εικόνα 4.31: Θεμιστοκλής

Όπως και στην περίπτωση του Ευρυβιάδη, ο Θεμιστοκλής θα εμφανιστεί μόνο σε μια σκηνή και επίσης έχει μόνο ένα animation στην αναπαράσταση της Σαλαμίνας και είναι στο μέρος που έχει την λογομαχία με τον Ευρυβιάδη. Ωστόσο, το animation του, που φαίνεται στην παρακάτω εικόνα, διαφέρει αρκετά από αυτό του Σπαρτιάτη αρχηγού του στόλου στην Εικόνα 4.30.



Εικόνα 4.32: Animation Θεμιστοκλή

#### 4.4 Αντικείμενα Περσικού στρατού

Τα συγκεκριμένα τρία αντικείμενα θα χρησιμοποιηθούν μόνο για την μια διαδραστική σκηνή της Μάχης των Θερμοπυλών, η οποία αφορά την διάταξη του Περσικού στρατού κατά την εκκίνηση της μάχης. Τα αντικείμενα είναι κατεβασμένα από ιστοσελίδες με έτοιμα και δωρεάν τρισδιάστατα αντικείμενα (Sketchfab, Free 3D) και χρησιμοποιήθηκαν στο Blender για αλλαγή του αρχικού χρώματος και των συντεταγμένων scale. Στην Εικόνα 4.33 εμφανίζονται και τα τρία αντικείμενα μαζί με την εξής σειρά Αδύναμοι, Δυνατοί και Τοξότες. Για τον εύκολο διαχωρισμό των αντικειμένων στην διαδραστική σκηνή, θα πραγματοποιηθεί και χρήση panel.



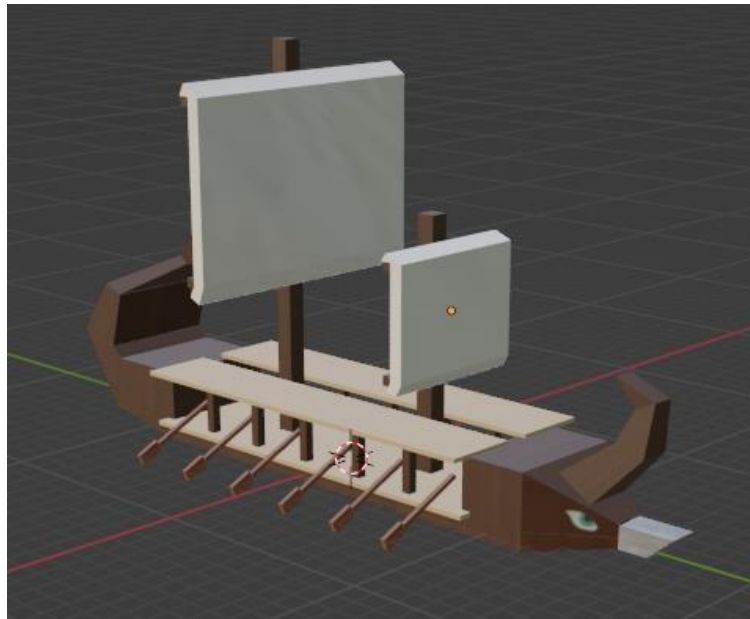
Εικόνα 4.33: Αντικείμενα Διάταξης Περσικού Στρατού

#### 4.5 Στόλοι

##### 4.5.1 Ελληνική Τριήρης

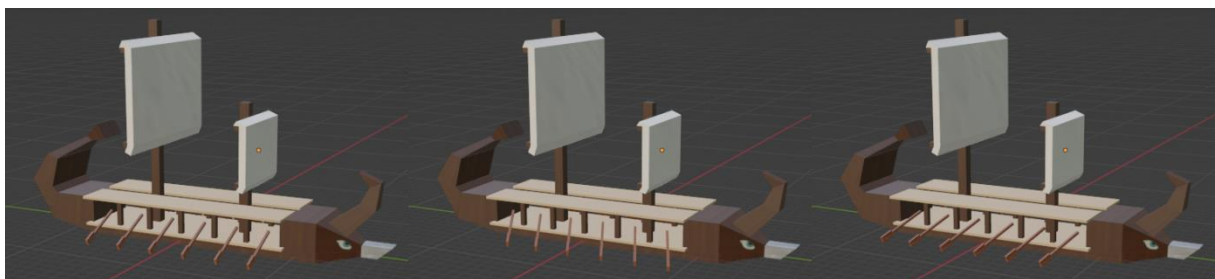
Η ελληνική τριήρης που θα χρησιμοποιηθεί είναι μικρό και ευκίνητο καράβι. Έχει σκελετό χρώματος σκούρο καφέ και κατάστρωμα ανοιχτόχρωμο. Επίσης έχει δύο κατάρτια, άσπρα πανιά και δεκαοχτώ κουπιά, για να κινείται γρήγορα. Στην πλώρη υπάρχει ένα σιδερένιο βέλος για να επιτίθεται στα αντίπαλα πλοία.

Στην πλώρη υπάρχουν σχεδιασμένα δύο μάτια (ένα από κάθε πλευρά) που πιστεύουν ότι φέρνουν τύχη και προστασία. Όλα τα παραπάνω παρουσιάζονται στην Εικόνα 4.34.



Εικόνα 4.34: Ελληνική Τριήρης

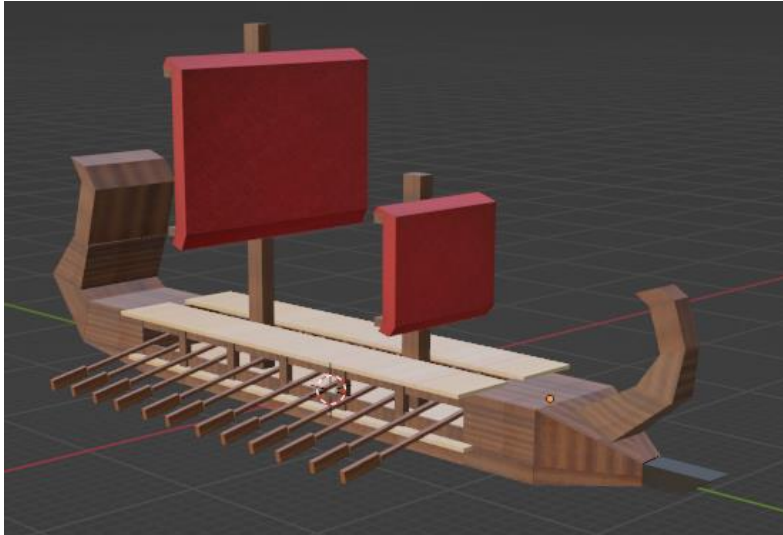
Το μοναδικό animation που έχει η ελληνική τριήρης είναι η κίνηση των κουπιών για όσο μετακινείται στον κόλπο της Σαλαμίνας, το οποίο φαίνεται στην παρακάτω εικόνα, κάνοντας απλές κινήσεις που είναι αρκετές για να μετακινηθεί το πλοίο.



Εικόνα 4.35: Animation Ελληνικής Τριήρης

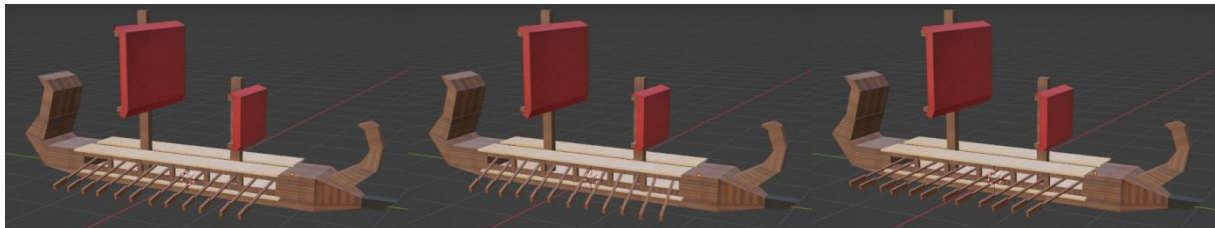
#### 4.5.2 Περσική Τριήρης

Η περσική τριήρης (Εικόνα 4.36) σε αντίθεση με την ελληνική, είναι πολύ πιο μεγάλη σε μέγεθος και έχει αρκετές διαφορές. Αρχικά τα πανιά του είναι κόκκινα και το σκαρί είναι ανοιχτόχρωμο χωρίς να υπάρχουν τα μάτια στην πλώρη του πλοίου. Λόγω του μεγάλου όγκου που έχει, περιέχει περισσότερα κουπιά από αυτά του ελληνικού έχοντας συνολικά είκοσι τέσσερα.



Εικόνα 4.36: Περσική Τριήρης

Το animation της περσικής τριήρης δεν διαφέρει καθόλου από αυτό του ελληνικού, καθώς πραγματοποιούν τις ίδιες ακριβώς κινήσεις για να μπορούν να μετακινούνται μέσα στη θάλασσα. Συγκρίνοντας τις Εικόνες 4.35 και 4.37 η μόνη διαφορά που μπορεί να παρατηρηθεί είναι στον αριθμό των κουπιών που κινούνται.



Εικόνα 4.37: Animation Περσικής Τριήρη

#### 4.6 Image Textures

Η τελευταία υποενότητα του κεφαλαίου περιέχει όλες τις εικόνες που χρησιμοποιήθηκαν στο περιβάλλον Blender, για να εισαχθούν στις περιοχές των Θερμοπυλών και της Σαλαμίνας.

Οι εικόνες που χρησιμοποιήθηκαν για τις περιοχές είναι οι εξής:



Εικόνα 4.38: Grass Texture



Εικόνα 4.39: Mud Texture



Εικόνα 4.40: Stone Mountain Texture



Εικόνα 4.41: Mountain Tree Texture



Εικόνα 4.42: Pathway Texture



Εικόνα 4.43: Coast Sand Texture



Εικόνα 4.44: Cliff Texture



Εικόνα 4.45: Rock Texture



Εικόνα 4.46: Stone Wall Texture



Εικόνα 4.47: Sea Texture

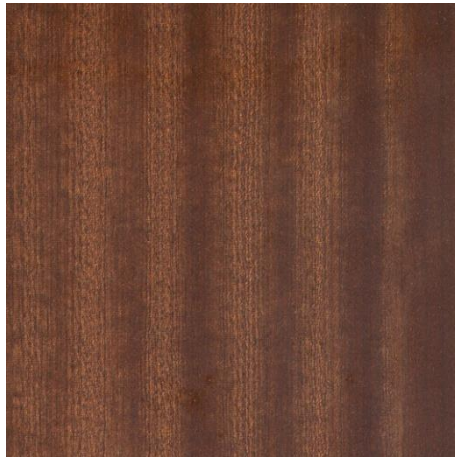
Για την περίπτωση των δύο στόλων χρησιμοποιήθηκαν τέσσερα textures τα οποία διαφέρουν αρκετά ανάμεσα στα δύο πλοία. Ένα κοινό που έχουν, είναι το texture για τα κουπιά, ενώ στο ελληνικό πλοίο προστέθηκε επιπλέον και ένα texture που περιέχει τα μάτια μπροστά στη πλώρη. Για τα μάτια να σημειωθεί πως χρησιμοποιήθηκαν δύο εικόνες του συγκεκριμένου texture, καθώς για την προσθήκη

## Κεφάλαιο 4

του ματιού στην άλλη πλευρά, θα έπρεπε να πραγματοποιηθεί το flip στην εικόνα, για να μπορεί να εισαχθεί σωστά στην άλλη πλευρά της πλώρης.



Εικόνα 4.48: Ship Texture (Greek)



Εικόνα 4.49: Ship Texture (Persian)



Εικόνα 4.50: Deck Texture (Greek)



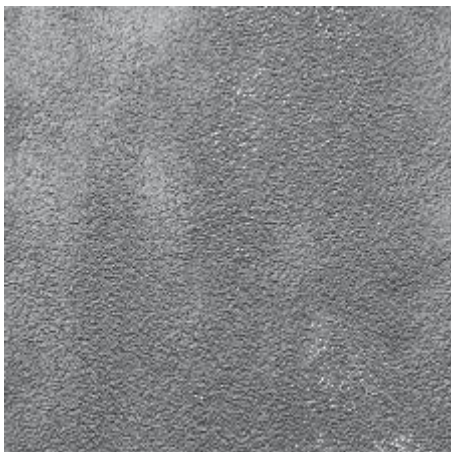
Εικόνα 4.51: Deck Texture (Persian)



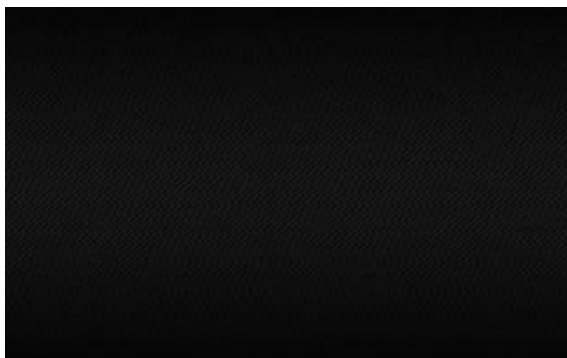
Εικόνα 4.52: Sails Texture (Greek)



Εικόνα 4.53: Sails Texture (Persian)



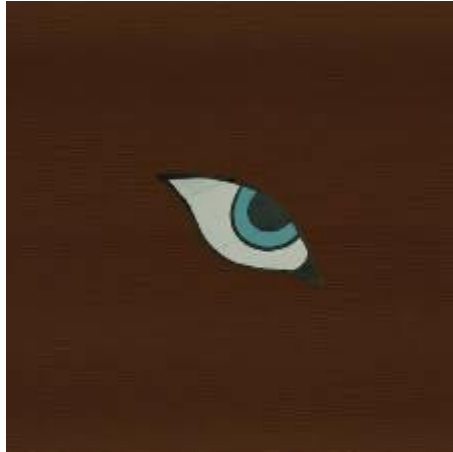
Εικόνα 4.54: Metal Texture (Greek)



Εικόνα 4.55: Metal Texture (Persian)



Εικόνα 4.56: Paddle Texture



Εικόνα 4.57: Eye Texture

Τέλος, για την Αρχαία πόλη της Αθήνας που βρίσκεται στην αναπαράσταση της Σαλαμίνας, δηλαδή στα τείχη και στα σπίτια έχουν χρησιμοποιηθεί τα παρακάτω textures.



Εικόνα 4.58: Grey Plain Texture



Εικόνα 4.59: Door Texture



Εικόνα 4.60: Ceramic Roof Texture



Εικόνα 4.61: Stone Pathway Texture

#### 4.7 Επίλογος

Σε αυτό το κεφάλαιο αναφέρθηκαν όλα τα αντικείμενα και image textures που πρόκειται να χρησιμοποιηθούν στην εφαρμογή Unity, τον τρόπο που έχουν διαμορφωθεί οι περιοχές που θα γίνουν οι δύο μάχες, καθώς και όσα αντικείμενα (χαρακτηριστικά, ενδυμασίες και όπλα) που χρησιμοποιήθηκαν για γίνουν οι τελικές μορφές των χαρακτήρων και στόλων μαζί με τα animation τους.

## Κεφάλαιο 5ο: Εφαρμογή και Εισαγωγική Σκηνή

### 5.1 Εισαγωγή

Σε αυτό κεφάλαιο αναφέρεται η υλοποίηση της αρχικής οθόνης και των εισαγωγικών σκηνών της εφαρμογής μέσα στην εφαρμογή Unity. Επίσης, θα παρουσιαστεί περιληπτικά η δομή της, με το περιεχόμενο όλων των σκηνών και για τις δύο μάχες να αναλύεται περισσότερο στα επόμενα κεφάλαια.

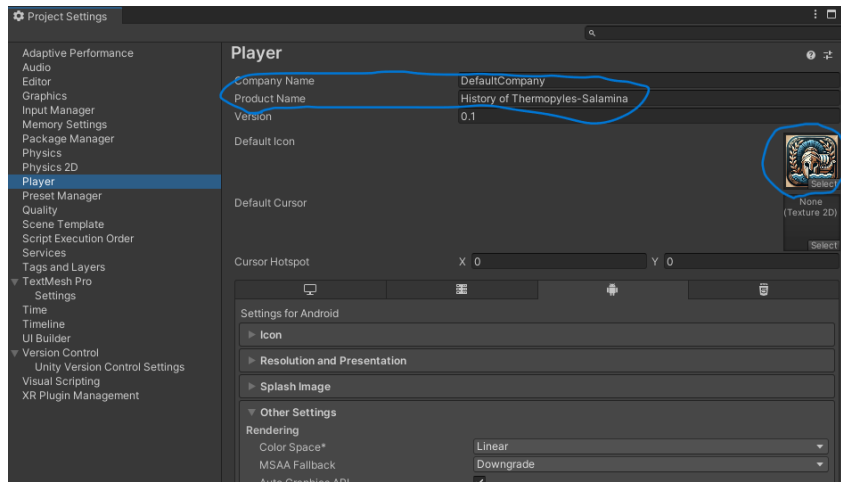
### 5.2 Ονομασία και Εικονίδιο Εφαρμογής

Ο χρήστης το πρώτο πράγμα που βλέπει πριν ανοίξει οποιαδήποτε εφαρμογή είναι το εικονίδιο και την ονομασία. Το icon που χρησιμοποιήθηκε είναι της Εικόνας 5.1, το οποίο κατέβηκε από ιστοσελίδα που δημιουργεί εικόνες.



Εικόνα 5.1: Icon εφαρμογής

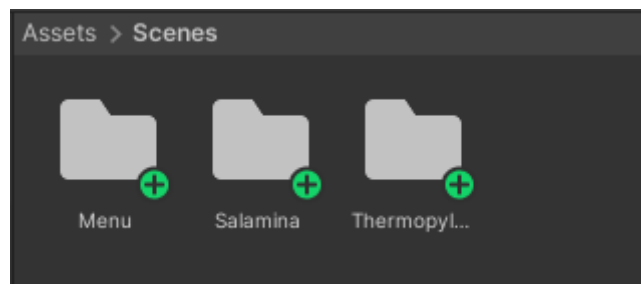
Για την διαδικασία εισαγωγής της εικόνας ως εικονίδιο της εφαρμογής θα έπρεπε αρχικά να ανέβει στα Assets της εφαρμογής στο Unity. Έπειτα πηγαίνοντας στο File που βρίσκεται πάνω αριστερά, υπάρχει η επιλογή “Build settings”, και μέσα στο περιεχόμενο του υπάρχει το “Player Settings”. Όταν επιλεγθεί το συγκεκριμένο υπάρχει το tab “Player” θα εμφανιστεί στην οθόνη η Εικόνα 5.2 και μέσα σε αυτό υπάρχει το “Default icon”, στο οποίο κάνουμε import την εικόνα από τον φάκελο Assets και το “Product name”, όπου ορίζουμε την ονομασία της εφαρμογής. Στην δικιά μας περίπτωση, η εφαρμογή ονομάστηκε “History of Thermopyles - Salamina”, καθώς αφορά μόνο τις δύο συγκεκριμένες μάχες.



Εικόνα 5.2: Tab Player

### 5.3 Δομή Εφαρμογής

Η δομή της εφαρμογής βασίστηκε από δύο κεφάλαια που βρίσκονται στην ύλη του μαθήματος Ιστορίας της τάξης Δ' Δημοτικού. Για την καλύτερη οργάνωση του περιεχομένου έχουν δημιουργηθεί οι τρεις φάκελοι που βρίσκονται στην παρακάτω εικόνα.

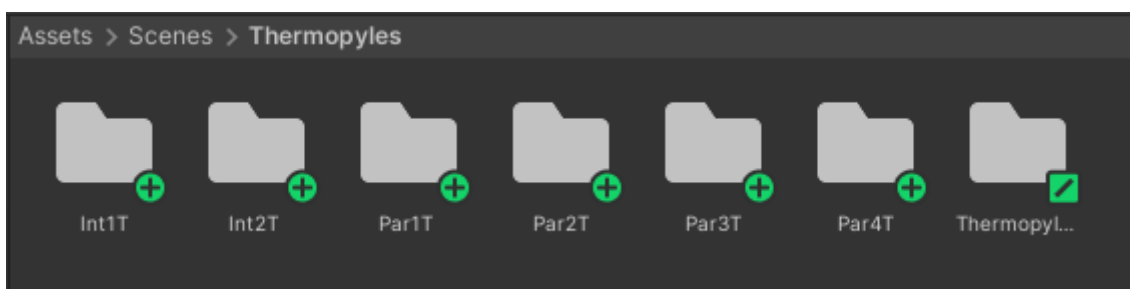


Εικόνα 5.3: Δομή Εφαρμογής

Ο φάκελος “Home” περιέχει την εισαγωγική σκηνή της εφαρμογής, η οποία θα αναφερθεί αναλυτικά στο υποκεφάλαιο 5.5 και οι υπόλοιποι δύο αφορούν τις μάχες που θα παρουσιαστούν ως αναπαραστάσεις. Η κάθε μάχη περιέχει δύο σκηνές όπου θα είναι το interactive mode, άλλες τέσσερις με πέντε σκηνές που θα δείχνουν μια σύντομη ιστορική αναφορά και το animation της σκηνής και στο τέλος υπάρχει ακόμη μια σκηνή, η οποία περιέχει ένα επαναληπτικό κουίζ για ό,τι έγινε μέσα στις δύο μάχες. Το κουίζ θα περιέχει συνολικά δέκα ερωτήσεις όπου στο τέλος όταν ο χρήστης τις απαντήσει όλες θα εμφανιστεί η τελική βαθμολογία του. Ο λόγος που δημιουργήθηκαν πολλές διαφορετικές σκηνές είναι για να ξεχωρίζονται πιο εύκολα όλα τα αντικείμενα, animation, panels και να βρίσκονται μόνο στην σκηνή που πρόκειται να εμφανιστούν. Με αυτόν τον τρόπο, θα μπορεί να γίνει πιο εύκολη η αλλαγή σε κάποιο συγκεκριμένο στιγμιότυπο της αναπαράστασης.

Στην Εικόνα 5.4 φαίνεται ο τρόπος που έχει τοποθετηθεί το περιεχόμενο της Μάχης των Θερμοπυλών. Για κάθε σκηνή έχει δημιουργηθεί ένας φάκελος και μέσα σε αυτόν περιέχεται το αρχείο C# script, τα αντικείμενα και animation, τα οποία θα παρουσιαστούν με περισσότερες

λεπτομέρειες στα επόμενα κεφάλαια. Στις σκηνές που ο χρήστης θα μετακινεί τα αντικείμενα, η ονομασία τους ξεκινάει με το “Int”, ενώ της αναπαράστασης μαζί με τις ερωτήσεις με το “Par”. Στη συνέχεια, εμφανίζεται ένας αριθμός και συμβολίζει την σειρά που θα εμφανιστούν τα “Int” και “Par” στην αναπαράσταση. Ο τρόπος που θα εμφανίζονται οι σκηνές θα βασίζεται ανάλογα με τις απαιτήσεις της εφαρμογής. Για παράδειγμα, μπορεί να ξεκινάει με την σκηνή “Par1T” και “Par2T” και στη συνέχεια να πηγαίνει στην “Int1T”. Το γράμμα “T” στο τέλος της ονομασίας αφορά το πρώτο γράμμα της λέξης “Thermopyles” και βοηθάει στην αποφυγή ορισμένων public μεταβλητών των αρχείων script. Τέλος, να αναφερθεί ότι με τον ίδιο ακριβώς τρόπο είναι δομημένος και ο φάκελος “Salamina”, με την μόνη τους διαφορά είναι το γράμμα “S” αντί του “T” (π.χ. Par1S, Int2S). Ο τελευταίος φάκελος περιέχει μόνο την περιοχή που θα γίνει αναπαράσταση με όλα τα textures και το τρισδιάστατο μοντέλο, όπως έχει διαμορφωθεί πιο πριν από το περιβάλλον Blender 3D.



Εικόνα 5.4: Δομή Φακέλου Thermopyles

## 5.4 Εφαρμογή

Στην εφαρμογή, ο χρήστης θα μπορεί να επιλέξει όποια από τις δύο αναπαραστάσεις θελήσει και μόλις δει ολόκληρη την αναπαράσταση, θα μπορεί να ξαναδεί την ίδια ή να επιλέξει την επόμενη. Για να μπορεί να τις δει, θα πρέπει να σκανάρει το image-target που έχει οριστεί στην εφαρμογή. Όταν το κάνει και για όσο θα σκανάρει την συγκεκριμένη εικόνα, θα έχει την δυνατότητα να δει τα animation της μάχης και να μετακινεί τα αντικείμενα στις σωστές τους θέσεις στις σκηνές που είναι interactive mode. Στο τέλος, θα απαντάει και σε κάποιες ερωτήσεις γνώσεων, όπου θα πρέπει να τις απαντήσει όλες. Όταν θα φτάνει στο τέλος του κουίζ, τότε θα εμφανιστεί το τελικό σκορ του χρήστη και όταν πατήσει το κουμπί που βρίσκεται κάτω θα τον επιστρέφει στο μενού (Εικόνα 5.11).

Για κάθε σκηνή, έχει υλοποιηθεί ξεχωριστός κώδικας, ώστε να υπάρχουν μόνο τα functions που σχετίζονται μόνο για αυτήν. Ο τρόπος που είναι δομημένα τα script είναι σχεδόν ο ίδιος σε όλα τα C# αρχεία, με κύριες μόνο αλλαγές στις ονομασίες των μεταβλητών και στα functions. Επίσης, μερικά από τα τρισδιάστατα αντικείμενα έχουν χρησιμοποιηθεί τουλάχιστον δύο φορές, λόγω των Interactive σκηνών και ορισμένων σημείων της αναπαράστασης όπου κάποιο αντικείμενο δεν έχει κάποιο animation.

## 5.5 Εισαγωγική Σκηνή

Όταν ο χρήστης ανοίξει την εφαρμογή, η πρώτη σκηνή που θα εμφανιστεί στην οθόνη του θα είναι αυτή του Home, η σκηνή είναι η Εικόνα 5.5 που βρίσκεται παρακάτω. Μέσα στο Panel του Home βρίσκεται στο πάνω μέρος ο τίτλος “Ιστορία Δ’ Δημοτικού”, καθώς οι ιστορικές αναπαραστάσεις θα

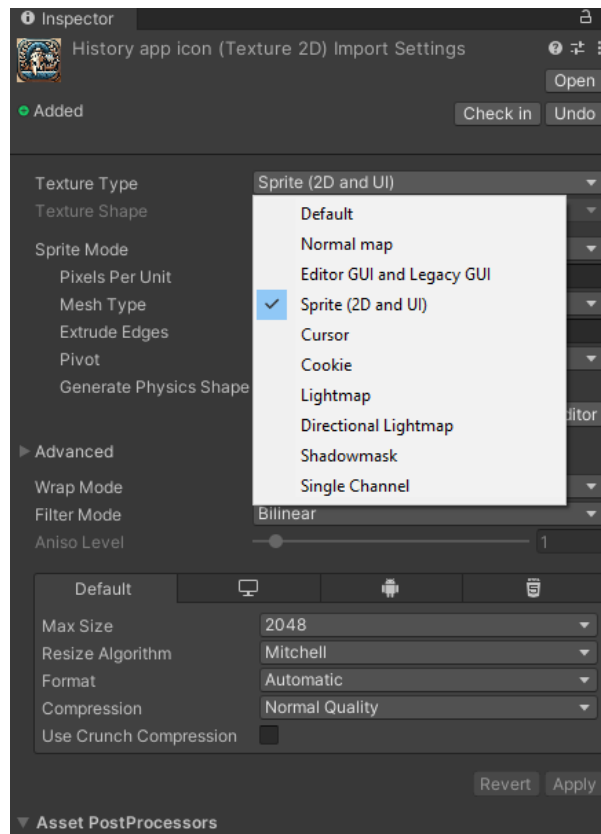
βασίζονται από το περιεχόμενο της ύλης της συγκεκριμένης τάξης, και υπάρχουν τρία Image Buttons, τα οποία έχουν τοποθετηθεί το ένα κάτω από το άλλο.

Οι εικόνες των button έχουν προστεθεί στον φάκελο Assets της εφαρμογής, αφού κατέβηκαν πρώτα από ιστοσελίδες που περιέχουν δωρεάν εικονίδια, και στη συνέχεια προστέθηκαν στα button. Το πρώτο κουμπί είναι για να συνεχίσει στην εφαρμογή και να μεταφερθεί στο Menu για να διαλέξει μια αναπαράσταση. Το δεύτερο τον μεταφέρει στις πληροφορίες της εφαρμογής και το τρίτο είναι για να τερματιστεί η εφαρμογή.



Εικόνα 5.5: Home Scene

Πριν την τοποθέτηση των εικόνων στα κουμπιά, θα πρέπει αρχικά να αλλάξει ο τύπος του texture και να γίνει 2D και συμβατό για το UI. Για να γίνει αυτό θα πρέπει να βρει την εικόνα μέσα στο φάκελο Assets και όταν την βρει να την επιλέξει. Τότε στο δεξιό μέρος της οθόνης θα εμφανιστεί η Εικόνα 5.6, και σε ένα από τα properties που έχει είναι το “Texture Type” με τον τύπο Default, σε αυτό το μέρος επιλέγεις το Sprite αντί για το αρχικό που είχε. Στην εικόνα είναι κυκλωμένο το σημείο που πρέπει να αλλάξει καθώς και τον τύπο που πρέπει επιλεγθεί.

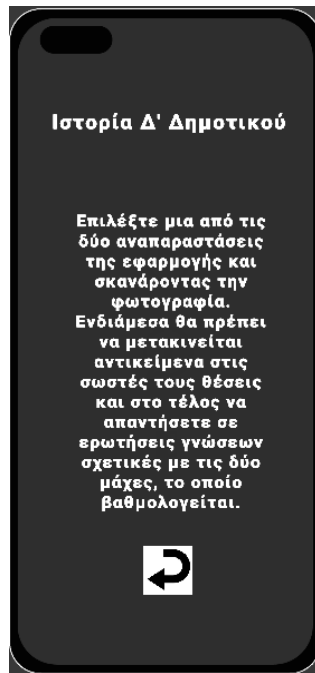


Εικόνα 5.6: Texture Type to Sprite

Για την αλλαγή σκηνών πρέπει στο script αρχείο να γίνει import το πακέτο `UnityEngine.SceneManagement` και μετά στα functions που χρησιμοποιούνται για την αλλαγή σκηνών γίνονται με την εντολή `SceneManager.LoadScene(Όνομα σκηνής);`.

### 5.5.1 Info Scene

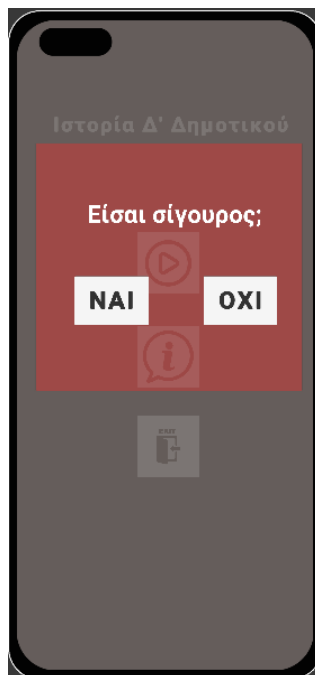
Μόλις ο χρήστης επιλέξει το δεύτερο button της αρχικής οθόνης θα τον μεταφέρει στο “Info Scene”. Σε αυτό το panel περιέχει μερικές πληροφορίες για τις αναπαραστάσεις της εφαρμογής και αυτά που πρέπει να κάνει στη διάρκεια αυτών. Στο κάτω μέρος της εφαρμογής, υπάρχει ένα image button, το οποίο αν το πατήσει ο χρήστης θα επιστρέψει στο Home Scene.



Εικόνα 5.7: Info Scene

### 5.5.2 Exit Panel

Το τρίτο κουμπί της αρχικής οθόνης, δεν μεταφέρει τον χρήστη σε κάποια άλλη σκηνή. Εμφανίζει στην ίδια οθόνη το panel της παρακάτω φωτογραφίας.



Εικόνα 5.8: Exit Panel

Σε αυτό το Panel υπάρχουν δύο κουμπιά και ένα text περιεχόμενο. Το text περιέχει ερώτηση στον χρήστη τον ρωτάει αν όντως θέλει να κλείσει την εφαρμογή και τα δύο κουμπιά περιέχουν τις απαντήσεις “ΝΑΙ” και “ΟΧΙ”. Στην περίπτωση του “ΟΧΙ”, το exit panel εξαφανίζεται και φαίνεται μόνο η αρχική οθόνη, ενώ με το κουμπί “ΝΑΙ”, η εφαρμογή τερματίζεται. Στην Εικόνα 5.9 παρουσιάζονται τα functions που υλοποιήθηκαν για να εμφανίζεται ή να εξαφανίζεται το panel και τον τερματισμό της εφαρμογής.

```

0 references
public void onExit(){
    quitPanel.SetActive(true);
}

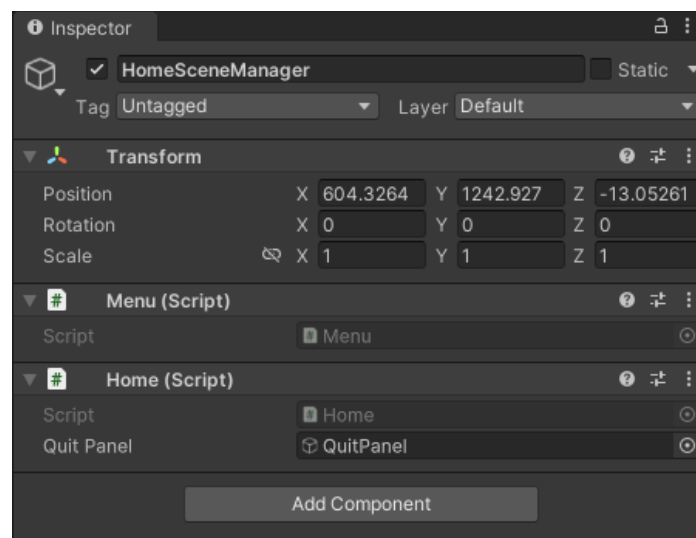
0 references
public void onYes(){
    Application.Quit();
}

0 references
public void onNo(){
    quitPanel.SetActive(false);
}

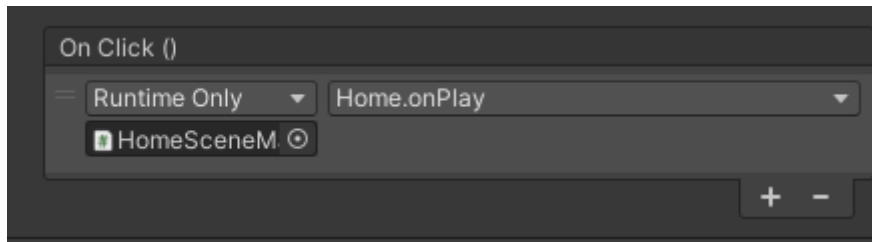
```

Εικόνα 5.9: C# Script – Exit Panel

Για να μπορέσουν να τρέξουν όλα τα παραπάνω, θα πρέπει πρώτα στο empty object “HomeSceneManager”, να προστεθεί το C# script και μετά σε κάθε κουμπί να επιλεγθεί το κάθε function που θα τρέχει όταν πατηθεί από τον χρήστη, επιλέγοντας το αρχείο script και μετά την function, όπως φαίνεται στην παρακάτω Εικόνα 5.11. Επίσης, να σημειωθεί πως και κάθε σκηνή θα έχει ένα empty object με την ονομασία να τελειώνει με το “SceneManager” και στην αρχή να περιέχει το όνομα της σκηνής.



Εικόνα 5.10: Home Scene Manager



Εικόνα 5.11: Button onClick()

### 5.5.3 Menu Scene

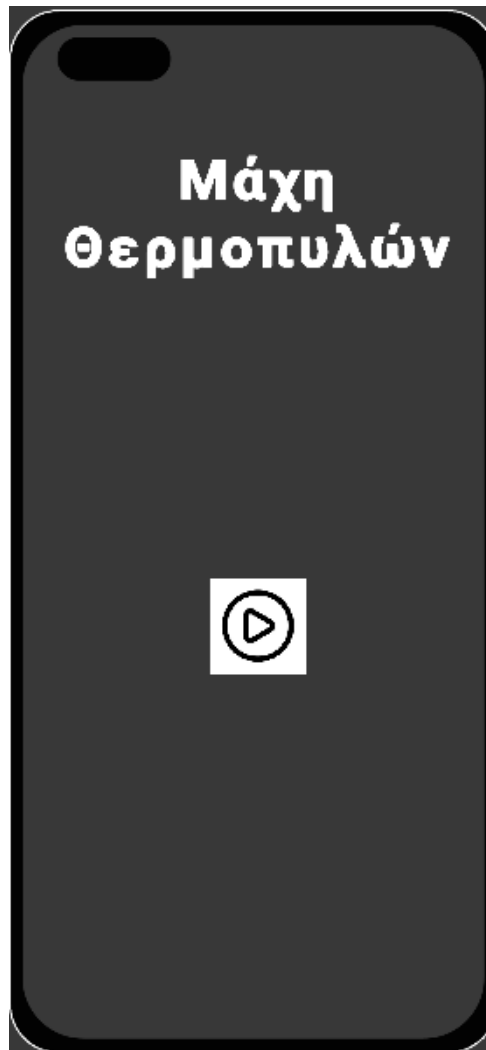
Το πρώτο κουμπί της αρχικής οθόνης θα μεταφέρει τον χρήστη στο Menu της εφαρμογής και θα συνεχίζει να έχει στο πάνω μέρος τον τίτλο “Ιστορία Δ’ Δημοτικού”. Ακριβώς από κάτω θα έχει ένα text περιεχόμενο, που θα λέει στον χρήστη να επιλέξει μια από τις δύο αναπαραστάσεις και στη συνέχεια θα έχει τον τίτλο της κάθε μάχης με ένα image button. Η μία θα είναι κάτω από την άλλη και για εικόνα χρησιμοποιήθηκε το 3D object που υλοποιήθηκε στο περιβάλλον του Blender και σε αυτό θα πραγματοποιηθούν όλες οι σκηνές. Τέλος, στο κάτω μέρος της οθόνης θα υπάρχει ένα ακόμη image button και είναι ίδιο με της σκηνής “Info”, χρησιμοποιείται η ίδια εικόνα και όταν πατηθεί θα επιστρέφει τον χρήστη στην αρχική οθόνη.



Εικόνα 5.12: Menu Scene

#### 5.5.4 Start Battle Scene

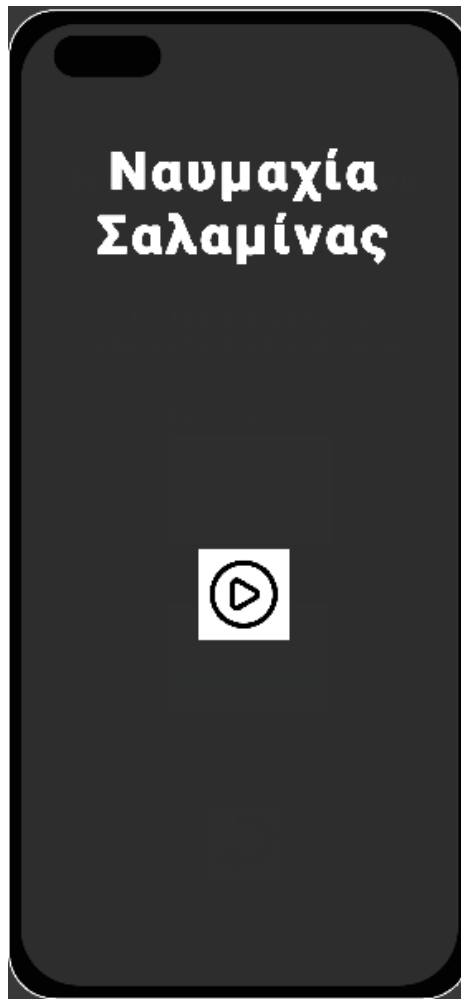
Την στιγμή που ο χρήστης επιλέξει μια από τις δύο μάχες, τότε στην οθόνη του θα εμφανιστεί η Εικόνα 5.13 ή η Εικόνα 5.15, ανάλογα ποια μάχη έχει επιλέξει και θα περιμένει να φορτώσει η αναπαράσταση. Η δομή και των δύο οθόνων είναι ακριβώς ίδια. Δηλαδή και τα δύο έχουν στο πάνω μέρος ένα text με το όνομα της μάχης και από κάτω ένα button που όταν το πατήσει θα τους μεταφέρει στην πρώτη AR σκηνή της εφαρμογής. Η μονή διαφορά είναι το text που αναφέρθηκε έχει το όνομα της κάθε μάχης. Τέλος, ο χρήστης δεν θα έχει την δυνατότητα να γυρίσει στην προηγούμενη οθόνη καθώς δεν εισάχθηκε στην οθόνη κάποιο button που να τον επιτρέπει να γυρίσει πίσω στο Menu.



Εικόνα 5.13: Έναρξη Θερμοπυλών

```
0 references  
public void StartTher(){  
    SceneManager.LoadScene("Int1T");  
}
```

Εικόνα 5.14: C# Script – Scene Load Int1T



Εικόνα 5.15: Έναρξη Σαλαμίνας

```
0 references  
public void StartSal(){  
    SceneManager.LoadScene("Par1S");  
}
```

Εικόνα 5.16: C# Script – Scene Load Par1S

## 5.6 Επίλογος

Σε αυτήν την ενότητα είδαμε το default εικονίδιο και την ονομασία που επιλέχθηκαν για την εφαρμογή και την δομή που έχει για την καλύτερη οργάνωση των πληροφοριών, των c# scripts και των σκηνών των αναπαραστάσεων. Στο τέλος, παρουσιάστηκε το περιεχόμενο της εισαγωγικής σκηνής της εφαρμογής και αναλύθηκε όλη η αρχική οθόνη μέχρι ο χρήστης να επιλέξει μια από τις δύο αναπαραστάσεις.

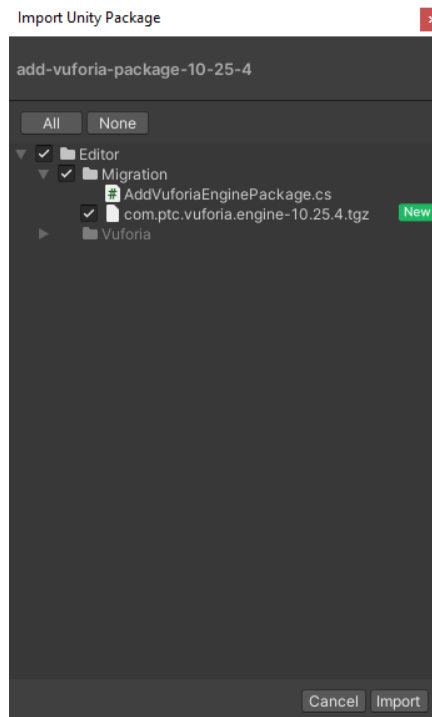
## Κεφάλαιο 6ο: Δημιουργία AR Scene

### 6.1 Εισαγωγή

Στην παρούσα ενότητα θα παρουσιαστεί ο τρόπος υλοποίησης των σκηνών που γίνεται η χρήση της τεχνολογίας AR. Με την χρήση της τεχνολογίας Vuforia Engine η δημιουργία των συγκεκριμένων σκηνών γίνεται πιο γρήγορα και αποτελεσματικά. Στα παρακάτω υποκεφάλαια θα παρουσιαστεί αναλυτικά όλη η διαδικασία. Τα αντικείμενα και τα Panels που χρησιμοποιήθηκαν για κάθε σκηνή θα αναφερθούν με περισσότερες λεπτομέρειες στις επόμενες δύο ενότητες της εργασίας.

### 6.2 Εισαγωγή Vuforia Engine στο Unity App

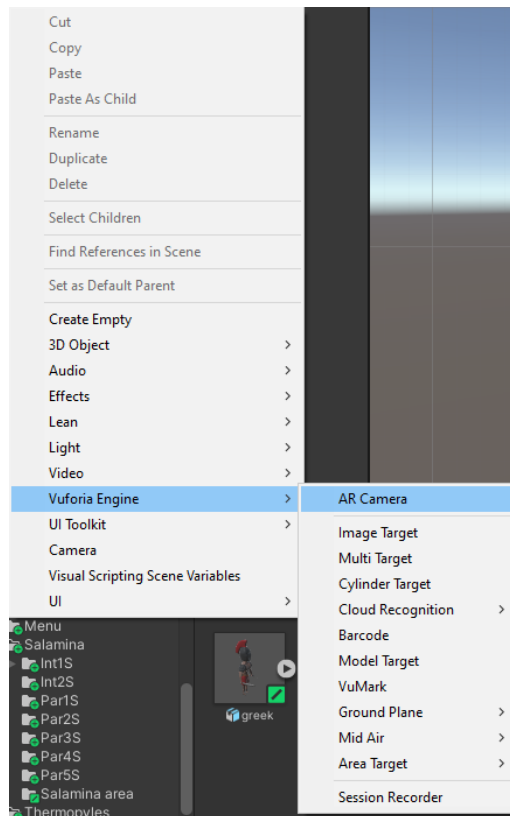
Αρχικά θα χρειαστεί να συνδεθεί κάποιος στην επίσημη ιστοσελίδα της Vuforia και να δημιουργήσει λογαριασμό, αν δεν έχει ήδη. Μόλις συνδεθεί ως χρήστης, τότε μπορεί να πάει στο Downloads -> SDK και επιλέγοντας την πρόσφατη έκδοση του Vuforia Engine (για την εφαρμογή χρησιμοποιήθηκε η έκδοση 10.24), τότε θα επιλέξει να κατεβάσει το αρχείο που αφορά το Unity. Μόλις κατέβει το συγκεκριμένο αρχείο, τότε μέσα στο περιβάλλον του Unity θα το μεταφέρει στο φάκελο Assets. Μόλις το κάνει, θα του εμφανιστεί το παράθυρο της Εικόνα 6.1 στην οθόνη και το μόνο που χρειάζεται κάποιος να κάνει είναι να επιλέξει το κουμπί “Import” και να ολοκληρωθεί η διαδικασία εισαγωγής.



Εικόνα 6.1: Import Vuforia Engine

### 6.3 AR Camera

Μόλις τελειώσει η εισαγωγή του αρχείου, τότε θα μπορεί να πραγματοποιηθεί η χρήση του Vuforia Engine μέσα στις σκηνές. Πατώντας δεξί κλικ στο αριστερό μέρος που βρίσκονται τα Object, στις επιλογές της Εικόνας 6.2 θα εμφανίζεται και η επιλογή “Vuforia Engine”.



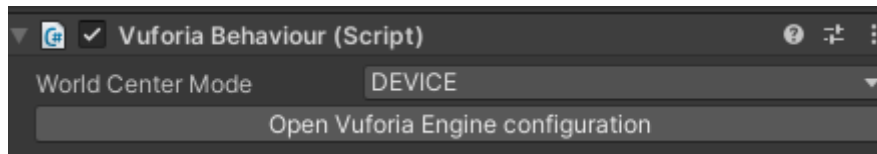
Εικόνα 6.2: Επιλογές Σκηνής

Όταν επιλεγθεί τότε θα εμφανιστούν νέες επιλογές. Μία από αυτές είναι η δημιουργία AR Camera, επιλέγοντας την συγκεκριμένη επιλογή τότε θα εμφανιστεί στα αριστερά με τα υπόλοιπα Objects. Πριν γίνει αυτό, θα πρέπει να διαγραφεί το αντικείμενο “Main Camera” που υπήρχε ήδη, από την δημιουργία του Scene.

### 6.4 License Key

Για να μπορεί να χρησιμοποιηθεί η AR Camera θα πρέπει να εισαχθεί ένα License key, το οποίο δημιουργείται στην επίσημη ιστοσελίδα της Vuforia. Έχοντας συνδεθεί κάποιος ως χρήστης και επιλέγοντας το “My account”, στην οθόνη θα του εμφανιστεί η επιλογή “Licenses”. Όταν το πατήσει θα του εμφανιστεί το κουμπί “Get basic”, και επιλέγοντάς το, θα του ζητηθεί να δώσει όνομα για το συγκεκριμένο κλειδί και να επιλέξει ένα checkbox. Μόλις τα κάνει και τα δύο, τότε θα δημιουργηθεί το License key και θα εμφανιστεί στην οθόνη του.

Στη συνέχεια, θα πρέπει να αντιγραφεί το κλειδί και να προστεθεί στο Unity. Επιλέγοντας το αντικείμενο “AR Camera”, τότε στο δεξιό μέρος της οθόνης θα εμφανιστούν τα Inspector του αντικειμένου της Εικόνας 6.3. Επιλέγοντας το κουμπί του “Vuforia Behavior” της εικόνας θα μεταφερθούμε στο Vuforia Configuration, και επιλέγοντας “Global”, θα εμφανιστεί το License key της εφαρμογής και σε εκείνο κάνουμε επικόλληση το κλειδί από την ιστοσελίδα της Vuforia.



Εικόνα 6.3: AR Camera

## 6.5 Image Target

Κατά την προσθήκη εικόνας θα χρειαστεί ξανά η σύνδεση στην επίσημη ιστοσελίδα της Vuforia. Στην επιλογή “My account”, μαζί με το “Licenses” που αναφέρθηκε στο προηγούμενο υποκεφάλαιο, υπάρχει και η επιλογή “Target Manager”, η οποία αφορά την βάση δεδομένων με όλα τα Targets. Αρχικά, θα δημιουργηθεί μια βάση, όπου χρειάζεται να δοθεί το όνομα της βάσης και τι τύπου θα είναι, στην συγκεκριμένη περίπτωση θα επιλεγθεί το “Device”. Στη συνέχεια, στη βάση αυτή θα εμφανίζεται ένα κουμπί που θα λέει “Add Target”, και εκεί διαλέγεις τι τύπο target θα ανεβάσεις, έχει τέσσερις επιλογές αλλά θα επιλεγθεί ο τύπος “Image”.

Στον στόχο τύπου εικόνας, θα χρειαστεί να ανέβει μια φωτογραφία από τον υπολογιστή, εισάγοντας για width έναν αριθμό που ο χρήστης επιθυμεί, αρκεί το scale να είναι ίσο με το augmented virtual content της εφαρμογής, και το όνομα που συμπληρώνεται αυτόματα με το όνομά της εικόνας που ανέβηκε. Θα πρέπει να δοθεί ιδιαίτερη προσοχή τι μοντέλο εικόνας θα ανέβει στη βάση, καθώς το Vuforia δέχεται μόνο εικόνες τύπου RGB Model. Αν δοθεί κάποιο άλλο μοντέλο, όπως είναι CMYK, τότε θα εμφανιστεί το εξής μήνυμα σφάλματος.

Εικόνα 6.4: Μήνυμα Σφάλματος

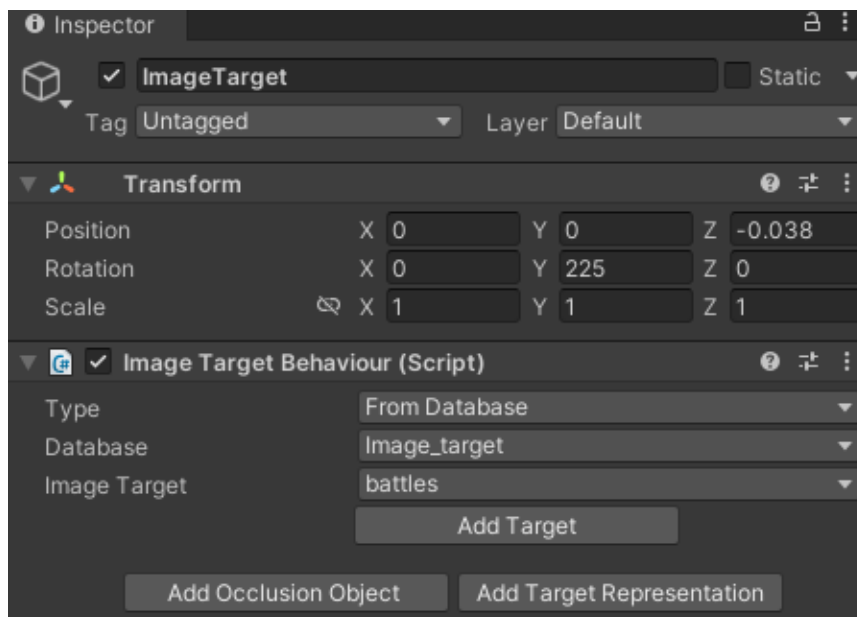
Για την αποφυγή τέτοιων προβλημάτων μπορεί να γίνει χρήση ιστοσελίδων που είναι για image generator, αναφέροντας ότι πρέπει να είναι τύπου RGB Model. Στην εργασία αυτή πραγματοποιήθηκε η χρήση τέτοιας ιστοσελίδας, και υλοποιήθηκε μια φωτογραφία που θα συνδυάζει και τις δύο μάχες της εφαρμογής, η οποία είναι η Εικόνα 6.5.



Εικόνα 6.5: Image Target Εφαρμογής

Όταν ανέβει η εικόνα στην βάση τότε θα πρέπει να εισαχθεί στο περιβάλλον του Unity. Μέσα στην βάση υπάρχει το κουμπί “Download database”, και όταν πατηθεί θα ζητηθεί για ποια πλατφόρμα να χρησιμοποιείται, διαλέγοντας την επιλογή “Unity”. Μόλις κατέβει τότε το αρχείο θα προστεθεί στο φάκελο Assets, και θα εμφανιστεί ένα παρόμοιο παράθυρο όπως της Εικόνας 6.1, με την μόνη διαφορά να είναι το περιεχόμενο, και τότε θα πατηθεί το κουμπί “Import”.

Για να προστεθεί η εικόνα στην σκηνή θα χρειαστεί να πάμε στα αντικείμενα της σκηνής, στο αριστερό μέρος της οθόνης. Όπως αναφέρθηκε παραπάνω, με το δεξί κλικ εμφανίζεται η επιλογή Vuforia και μετά μαζί με την “AR Camera”, υπάρχει και η προσθήκη “Image Target”. Όταν προστεθεί, θα εμφανίζεται στο αριστερό μέρος και όταν πατήσουμε σε αυτό θα φαίνονται τα Inspector στα δεξιά, όπως είναι στην Εικόνα 6.6.



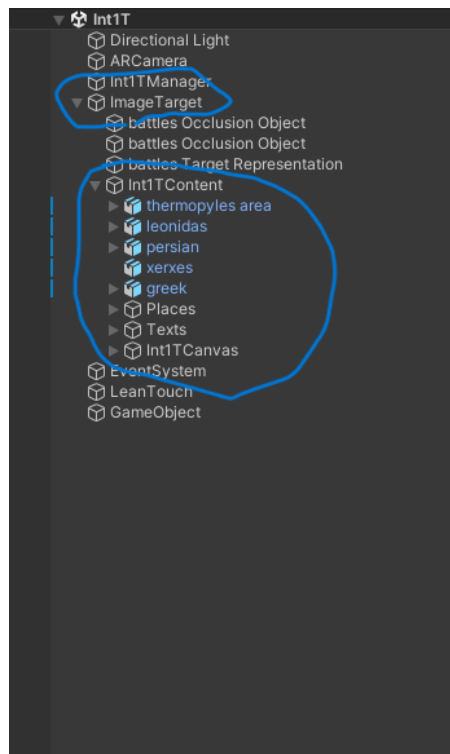
Εικόνα 6.6: Image Target Inspector

Μέσα σε αυτό το περιεχόμενο υπάρχει ένα script με το όνομα “Image Target Behavior”, στο οποίο θα προστεθεί η εικόνα. Για να γίνει το import, θα επιλεγεί για Type η επιλογή “From Database”, για “Database” το όνομα της βάσης που έχει δοθεί και τέλος για το “Image Target” το όνομα της συγκεκριμένης εικόνας που είναι μέσα στην βάση.

## 6.6 Εισαγωγή Τρισδιάστατων Αντικειμένων και Panel

Μόλις ολοκληρωθεί η διαδικασία προσθήκης Image Target, τότε στην σκηνή θα εμφανίζεται η εικόνα που έχουμε ορίσει για στόχο. Για την τελική υλοποίηση των AR σκηνών, το μόνο που θα χρειαστεί να γίνει είναι τα αντικείμενα και τα Panel να προστεθούν στο σημείο που βρίσκεται η εικόνα, ώστε όταν την αναγνωρίζει η εφαρμογή, όλο το περιεχόμενο της σκηνής να εμφανίζεται εκεί πάνω.

Την στιγμή που θα ολοκληρωθεί η προσθήκη όλου του περιεχομένου στην σκηνή, θα χρειαστεί όλα τα αντικείμενα που βρίσκονται στην σκηνή, να μεταφερθούν μέσα στο Object “Image Target”, όπως ακριβώς φαίνεται στην Εικόνα 6.7, για να εμφανίζονται στην τελική εφαρμογή.



Εικόνα 6.7: Τοποθέτηση Αντικειμένων στο Image Target

Όσα από τα αντικείμενα δεν βρίσκονται μέσα στο Image Target τότε αυτά δεν θα εμφανίζονται μέσα στην εφαρμογή, την στιγμή που θα γίνει η αναγνώριση της εικόνας. Το object με το όνομα “Int1TManager” δεν μπαίνει μέσα στο Image Target επειδή δεν περιέχει κάποιο αντικείμενο, παρά μόνο τα C# script αρχεία των σκηνών. Σε όλες τις σκηνές ισχύει το ίδιο με το παραπάνω Object ώστε μετά να χρησιμοποιείται στα αντικείμενα της σκηνής και στον τερματισμό της σκηνής για να συνεχίσει με την επόμενη.

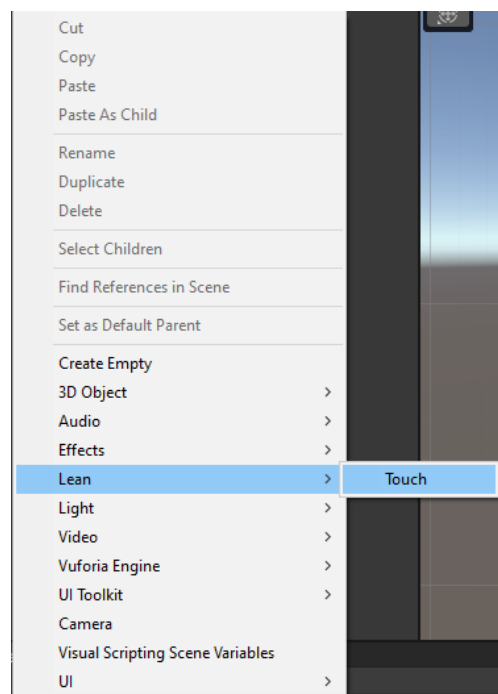
## 6.7 Lean Touch

Το παρόν πακέτο από αυτό το υποκεφάλαιο δεν έχει καμία σχέση με το Vuforia Engine. Ωστόσο, είναι αναγκαίο για τις διαδραστικές σκηνές της εφαρμογής, καθώς παρέχει στην εφαρμογή πολλές δυνατότητες που μπορεί να χρησιμοποιήσει ο χρήστης στα αντικείμενα μιας σκηνής, δηλαδή να κάνει Scale, Rotate και Drag σε όσα αντικείμενα πρόκειται να χρησιμοποιούν το Lean Touch package. Το download είναι πολύ απλό καθώς γίνεται μέσω του Unity Asset Store, αναζητώντας και επιλέγοντας το πακέτο να γίνει import αρχικά στο λογαριασμό και μετά στην εφαρμογή μέσω του Windows -> Package Manager.

Μόλις κατέβει και στην εφαρμογή του Unity, τότε πατώντας δεξί κλικ στο αριστερό μέρος της σκηνής, στο οποίο βρίσκονται όλα τα αντικείμενα, θα εμφανίζεται η επιλογή Lean -> Touch και μετά αυτό αποθηκεύεται μαζί με τα υπόλοιπα objects. Για την χρήση της θα πρέπει να επιλεχθούν τα αντικείμενα που σκοπεύετε να γίνει αλλαγή location, scale και το rotation και μετά επιλέγεται ένα από τα έτοιμα script του.

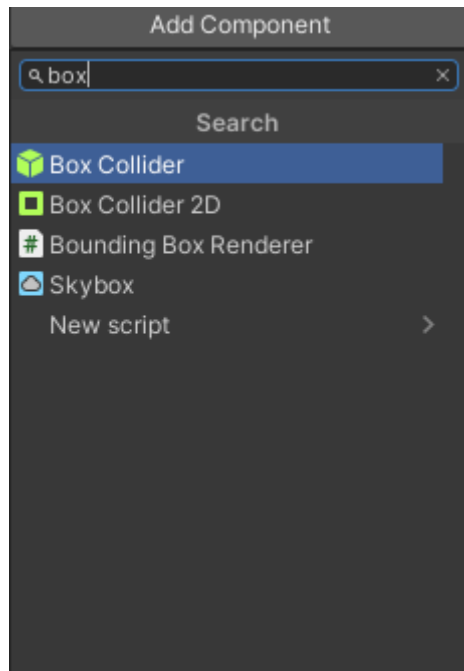
Στη συνέχεια, υλοποιήθηκε το script αρχείο “Object Drag” και χρησιμοποιώντας το Lean Touch μπορούμε να προγραμματίσουμε τα αντικείμενα να μετακινούνται μόνο ένα το καθένα ανάλογα ποιο θα επιλέξει ο χρήστης και όχι όλα μαζί. Μετά αυτό το αρχείο αποθηκεύεται στο Scene Manager της κάθε διαδραστικής σκηνής και στο τέλος εισάγεται το c# script στα Components των συγκεκριμένων αντικειμένων. Ο κώδικας του αρχείου βρίσκεται στο Παράρτημα BB.

Στις Εικόνες 6.8 και 6.9 δείχνει την τρόπο που μπορείς να προσθέσεις το πακέτο Lean Touch στην σκηνή και μετά στα αντικείμενα. Αρχικά στην Εικόνα 6.8, πηγαίνοντας στο αριστερό μέρος της οθόνης που έχει όλα τα αντικείμενα πατώντας δεξί κλικ εμφανίζεται η συγκεκριμένη και στη συνέχεια επιλέγουμε Lean->Touch



Εικόνα 6.8: Επιλογή Lean Touch

Όταν προστεθεί το αντικείμενο Lean Touch μαζί με τα υπόλοιπα της σκηνής, τότε θα πρέπει επιλεγθούν ποια αντικείμενα θα αλλάζουν με την αφή στο κινητό. Για να γίνει αυτό, επιλέγεται το αντικείμενο από το αριστερό μέρος της οθόνης και στο Inspector στα δεξιά, επιλέγουμε το κουμπί “Add Components” και προσθέτουμε αρχικά το script “Drag Object” και μετά πρέπει να προστεθεί και το Component της Εικόνας, που είναι το Box Collider, το οποίο το σύστημα θα αναγνωρίζει το αντικείμενο και μετά θα είναι πιο εύκολο να το επιλέξεις και να το μετακινήσεις. Όταν επιλεγθεί μετά θα πρέπει το λαχανί κουτί που εμφανίζεται γύρω από το αντικείμενο να διαμορφωθεί κατάλληλα για να βρίσκεται μέσα στο box όλο το αντικείμενο.



Εικόνα 6.9: Εισαγωγή Box Collider

## 6.8 Επίλογος

Σε αυτή την ενότητα αναφέρθηκε ολόκληρη η διαδικασία που χρειάζεται για την δημιουργία σκηνών που χρησιμοποιούν την τεχνολογία της ΕΠ και στο τέλος το Lean Touch για την περίπτωση των διαδραστικών σκηνών. Ενδιάμεσα αναφέρθηκε ένα σύνηθες πρόβλημα που μπορεί να προκύψει κατά τη διάρκεια υλοποίησης, που είναι η περίπτωση με το μοντέλο της εικόνας, όπου στο τέλος υπάρχει και μια πρόταση για την αποφυγή του συγκεκριμένου θέματος.



## Κεφάλαιο 7ο: Μάχη των Θερμοπυλών

### 7.1 Εισαγωγή

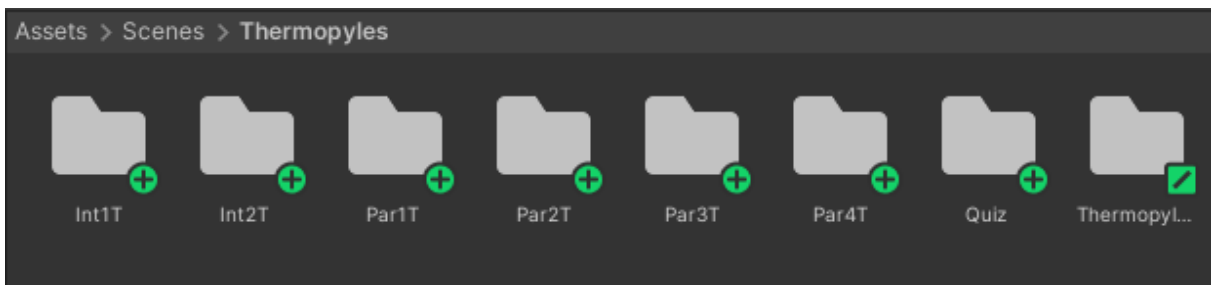
Στο κεφάλαιο επτά θα αναφερθεί μια σύντομη ιστορική αναφορά για την μάχη, η οποία βασίζεται στην ύλη που βρίσκεται στο βιβλίο του μαθήματος Ιστορία Δ' Δημοτικού, ώστε να είναι πλήρως κατανοητή για τα παιδιά της Δ' τάξης, χωρίς να περιέχει παραπάνω πληροφορίες από μεγαλύτερες τάξεις. Επιπλέον, στα επόμενα υποκεφάλαια θα αναφερθεί αναλυτικά όλη η διαδικασία υλοποίησης της κάθε σκηνής, δηλαδή το περιεχόμενο που έχει προστεθεί για την κάθε μία ξεχωριστά (αντικείμενα, animation, panel, location αντικειμένων).

### 7.2 Ιστορική Αναδρομή

Η Μάχη των Θερμοπυλών έγινε το 480 π.Χ. ανάμεσα στους Έλληνες και τους Πέρσες. Ο βασιλιάς Ξέρξης ήρθε με έναν πολυάριθμο στρατό με σκοπό να κατακτήσει την Ελλάδα. Λόγω του μεγάλου αριθμού των Περσών ήταν δύσκολο να αντιμετωπιστούν στις πεδιάδες και για αυτό το λόγο οι Έλληνες με αρχηγό τον Λεωνίδα αποφάσισαν να τους σταματήσουν στα στενά των Θερμοπυλών. Ακόμη και όταν τους είπε ο Ξέρξης να παραδώσουν τα όπλα, η απάντηση που πήρε από τον Λεωνίδα ήταν “Μολών Λαβέ”. Παρόλο που οι Έλληνες ήταν πολύ λιγότεροι κατάφεραν να κρατήσουν γενναία τις θέσεις τους για τρεις μέρες, μέχρι και την εμφάνιση του Εφιάλτη ο οποίος βοήθησε τους Πέρσες να περάσουν από την άλλη πλευρά και να βρεθούν στις πλάτες τους. Όταν έμαθε ο Λεωνίδας πως πρόκειται να περικυκλωθούν, είπε σε όσους ήθελαν να φύγουν από το πεδίο μάχης και έμεινε αυτός με τριακόσιους Σπαρτιάτες και εφτακόσιους Θεσπιείς. Έχοντας επίθεση από δύο μέτωπα, οι Έλληνες πολέμησαν γενναία ώσπου στο τέλος όλοι τους έπεσαν ηρωικά στη μάχη[34].

### 7.3 Δομή Φακέλου Thermopyles

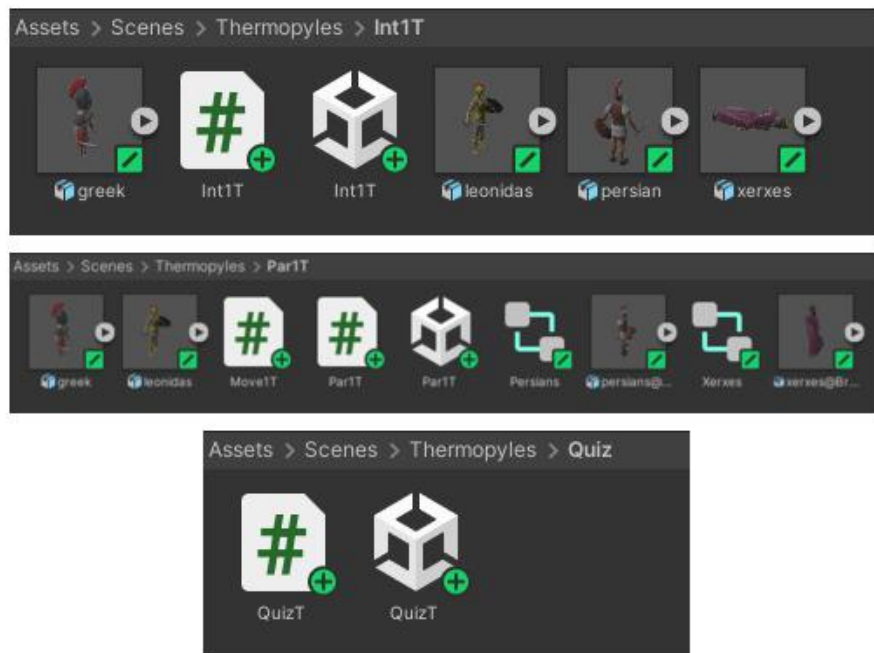
Ο φάκελος Thermopyles περιέχει όλο το περιεχόμενο της αναπαράστασης της Μάχης των Θερμοπυλών. Ο τρόπος που έχει οριστεί η δομή του φαίνεται στην Εικόνα 7.1 και είχε αναφερθεί σαν παράδειγμα στο υποκεφάλαιο 5.2, για κάθε σκηνή έχει δημιουργηθεί ένας φάκελος με το αντίστοιχο όνομα της σκηνής. Τα Int αναφέρονται στις διαδραστικές σκηνές, τα Par για τις αναπαραστάσεις και το Quiz για τις ερωτήσεις στο τέλος. Επίσης το γράμμα T στο τέλος συμβολίζει πως είναι για την μάχη των Θερμοπυλών.



Εικόνα 7.1: Δομή Φακέλου Thermopyles μαζί με το Quiz

Ο κάθε υποφάκελος περιέχει διαφορετικό περιεχόμενο. Όλα εκτός από το κουίζ περιέχουν τρισδιάστατα αντικείμενα που θα χρησιμοποιηθούν στην συγκεκριμένη σκηνή μαζί με τον κώδικα για

την διαχείριση των panels, τα οποία φέρουν το όνομα του φακέλου που βρίσκονται. Σε κάποιες από τις σκηνές με τις αναπαραστάσεις χρησιμοποιείται και script με το όνομα Move μαζί με τον αριθμό υποφακέλου και το γράμμα T και χρησιμοποιείται για το location αντικειμένων καθώς και τα animation που θα εμφανιστούν στην σκηνή και τα animation controllers που αυτά διαχειρίζονται τις κινήσεις του κάθε αντικειμένου. Στην Εικόνα 7.2 φαίνεται το περιεχόμενο των φακέλων Int, Par και Quiz.



Εικόνα 7.2: Περιεχόμενο Υποφακέλων Thermopyles

### 7.4 Δημιουργία Σκηνών Αναπαράστασης

Στην περίπτωση της Μάχης των Θερμοπυλών χρησιμοποιήθηκαν δύο διαδραστικές σκηνές, τέσσερις σκηνές για τα σημαντικά στιγμιότυπα της αναπαράστασης και στο τέλος η σκηνή που περιέχει το τελικό κουίζ που θα πρέπει να απαντήσει ο μαθητής με βάση από αυτά που είδε στην αναπαράσταση. Για την αποφυγή επανάληψης, να σημειωθεί πως οι σκηνές που παρουσιάζουν τα σημαντικά στιγμιότυπα, όλες περιέχουν panel με ιστορική αναφορά και μετά από είκοσι δευτερόλεπτα, θα συνεχίσει αυτόματα στην επόμενη αναπαράσταση.

Στο συγκεκριμένο υποκεφάλαιο, οι σκηνές αυτές θα παρουσιαστούν με την σειρά που θα εμφανίζονται στην τελική εφαρμογή.

#### 7.4.1 Int1T - Διάταξη στρατευμάτων

Την στιγμή που ο χρήστης πατήσει το κουμπί της εκκίνησης, τότε θα μεταφερθεί στην πρώτη διαδραστική σκηνή, η οποία είναι και η πρώτη όλης της αναπαράστασης. Στη σκηνή υπάρχει ένα Panel στο πάνω μέρος και περιέχει ένα text που ζητάει από τον χρήστη να μετακινήσει τα αντικείμενα στις σωστές τους θέσεις. Τα αντικείμενα που πρέπει να μετακινήσει φαίνονται στην Εικόνα 7.3 και είναι οι βασιλιάδες των δύο στρατών και οι στρατιώτες και μπορεί εύκολα να τους ξεχωρίσει ο

χρήστης καθώς ακριβώς κάτω από τα τέσσερα αντικείμενα περιέχει ένα text που αναφέρει ποιος είναι π.χ ο Λεωνίδα, ο Ξέρξης κτλ.



Εικόνα 7.3: Πρώτη Διαδραστική Σκηνή Θερμοπυλών

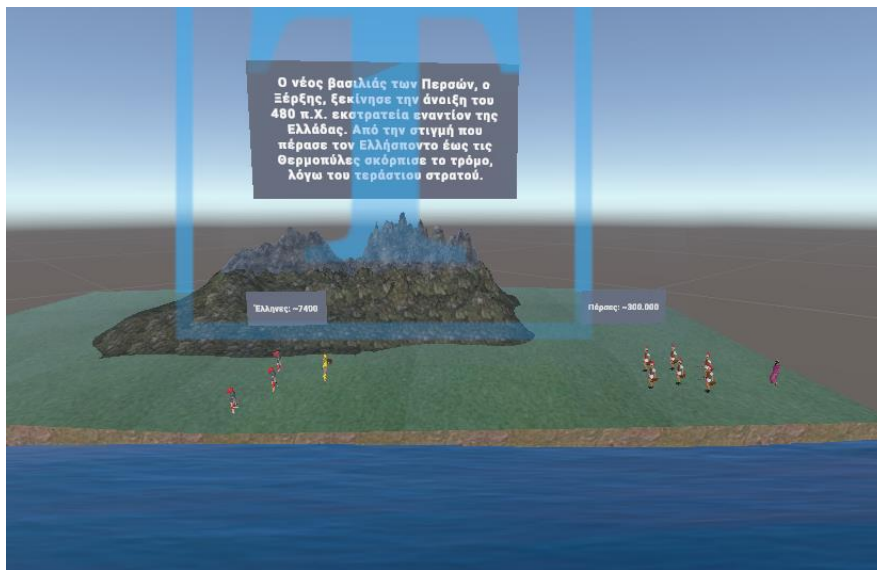
Ο σκοπός της συγκεκριμένης σκηνής είναι να τοποθετηθούν τα αντικείμενα στην θέση που είναι στη διάρκεια της μάχης. Με την χρήση του πακέτου Lean Touch που αναφέρθηκε στην Ενότητα 6, ο χρήστης θα μπορεί να μετακινήσει τα αντικείμενα όπου θέλει, χωρίς να λάβει υπόψη αν τα τοποθέτησε στο σωστό μέρος ή όχι. Έχουν διαμορφωθεί ειδικά panel μέσα στην περιοχή των Θερμοπυλών για την τοποθέτηση των αντικειμένων στα συγκεκριμένα μέρη. Ακόμη και αν ο χρήστης αφήσει τα αντικείμενα σε θέση που είναι μακριά από τα ειδικά panels, πάλι δεν θα το λάβει υπόψη.



Εικόνα 7.4: Μετακίνηση Αντικειμένων

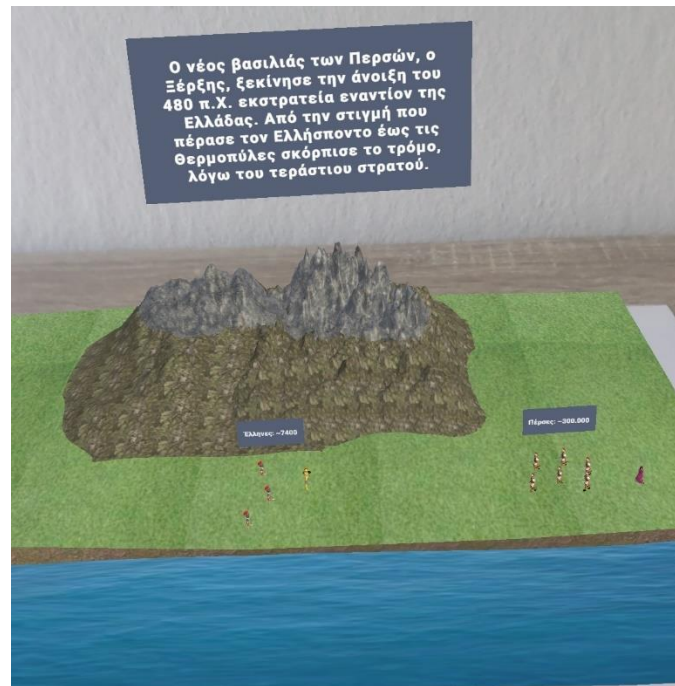
#### 7.4.2 Par1T - Οι Πέρσες φτάνουν στις Θερμοπύλες

Η δεύτερη σκηνή της μάχης είναι η πρώτη αναπαράσταση με τα σημαντικά στιγμιότυπα. Στο περιεχόμενό της έχει τρία panels. Ένα για την ιστορική αναφορά της σκηνής, και άλλα δύο που περιέχουν τον συνολικό αριθμό των ελλήνων και των περσών στρατιωτών, τα οποία θα χρησιμοποιηθούν και στις επόμενες σκηνές. Στην περίπτωση των αντικειμένων, λόγω δυσκολίας εισαγωγής πολλών στρατιωτών όσων ήταν στην πραγματικότητα, τοποθετήθηκαν πολλοί λιγότεροι. Υπάρχουν τρεις Έλληνες στρατιώτες, ο Λεωνίδας, ο Ξέρξης και έξι Πέρσες στρατιώτες.



Εικόνα 7.5: Πρώτη Σκηνή Αναπαράστασης Θερμοπυλών

Στην παρούσα σκηνή το animation που υπάρχει είναι που θα δείχνει τον Περσικό στρατό μαζί με τον βασιλιά Ξέρξη να περπατάνε από την μακρινή Περσία ως τις Θερμοπύλες. Όσο για την περίπτωση του Location, θα φαίνεται η μετακίνηση τους προς το μέρος που βρίσκεται ο ελληνικός στρατός, μέσα στο στενό των Θερμοπυλών. Έχοντας ορίσει positions targets, τα αντικείμενα θα μετακινούνται μέχρι να φτάσουν στο συγκεκριμένο σημείο. Οι Έλληνες δεν θα πραγματοποιούν καμιά κίνηση σε όλη την διάρκεια της σκηνής.

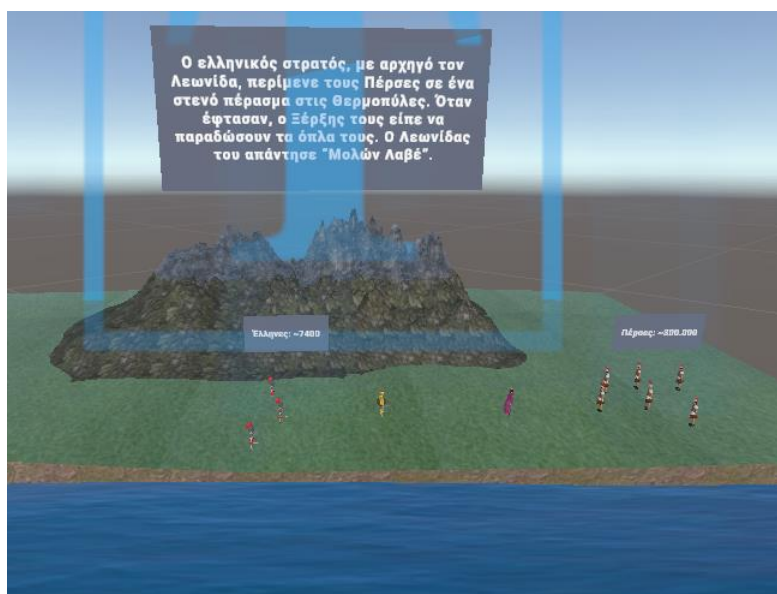


Εικόνα 7.6: Animation και Location Περσών

Θα συνεχίσει να βλέπει την αναπαράσταση μέχρι να περάσουν τα είκοσι δευτερόλεπτα από την στιγμή που ξεκίνησε και τότε θα μεταφερθεί στην επόμενη σκηνή.

#### 7.4.3 Par2T - Συζήτηση μεταξύ Λεωνίδα και Ξέρξη

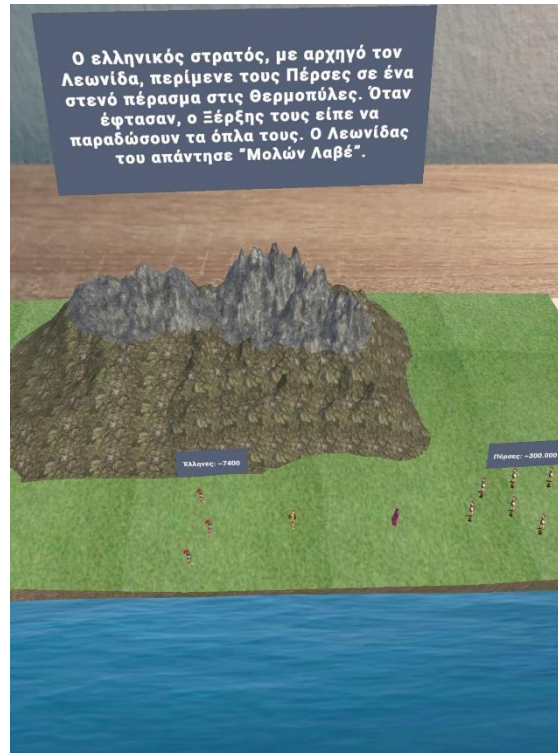
Στην δεύτερη αναπαράσταση θα δείχνει τους δύο βασιλιάδες να μιλάνε μεταξύ τους. Τα panels είναι τα ίδια που θα χρησιμοποιηθούν με το μόνο που θα αλλάζει είναι η ιστορική αναφορά στο κύριο panel. Ακόμη και τα αντικείμενα που θα χρησιμοποιηθούν είναι ακριβώς τα ίδια.



Εικόνα 7.7: Δεύτερη Σκηνή Αναπαράστασης Θερμοπυλών

## Κεφάλαιο 7

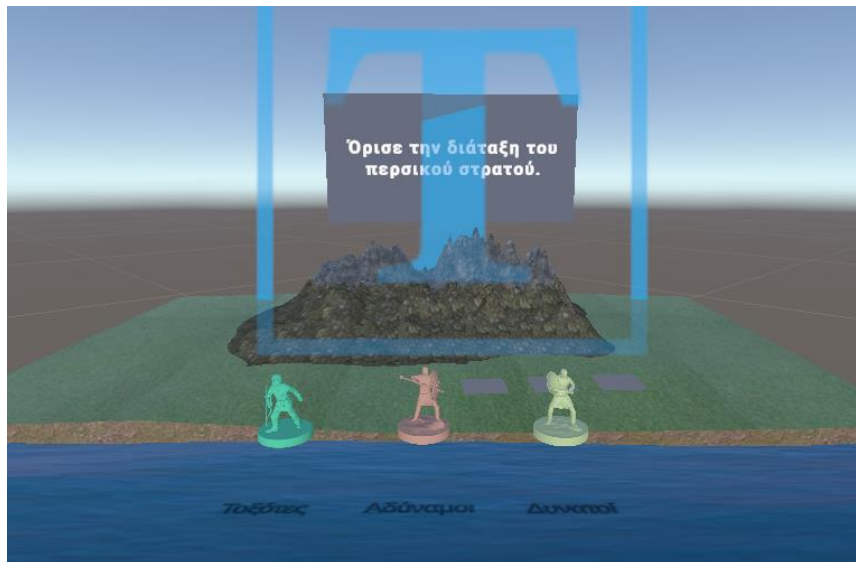
Στην περίπτωση του animation, οι μόνες κινήσεις που θα υπάρχουν θα είναι του Λεωνίδα και του Ξέρξη που μιλάνε μεταξύ τους, οι οποίοι θα πραγματοποιούν διαφορετικές κινήσεις. Όλοι οι στρατιώτες δεν θα κάνουν καμία κίνηση και θα φαίνεται να παρακολουθούν μόνο την συζήτηση των δύο βασιλιάδων. Τα animation και των δύο αντικειμένων περιέχουν ατέρμων βρόχο και θα παίζουν μέχρι να περάσει το χρονικό όριο.



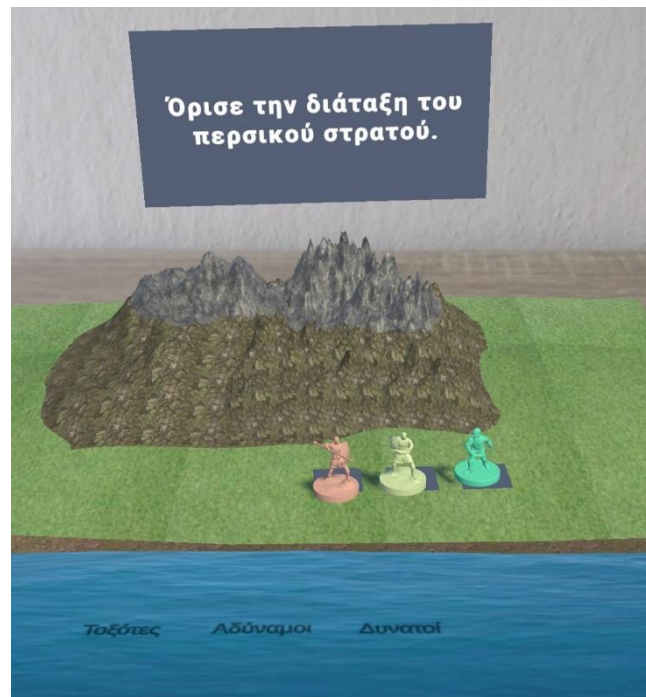
Εικόνα 7.8: Animation Λεωνίδα – Ξέρξη

### 7.4.4 Int2T - Διάταξη περσικού στρατού

Στην δεύτερη διαδραστική σκηνή της Μάχης των Θερμοπυλών, υπάρχουν τρία αντικείμενα, που είναι ένας τοξότης, ο αδύναμος και ο δυνατός στρατιώτης. Το ζητούμενο της συγκεκριμένης σκηνής είναι ο χρήστης να ορίσει τον τρόπο που διατάχθηκε ο Περσικός στρατός μέσα στην μάχη και να τοποθετήσει τα αντικείμενα στα τρία panel που είναι στο μέρος των Θερμοπυλών.



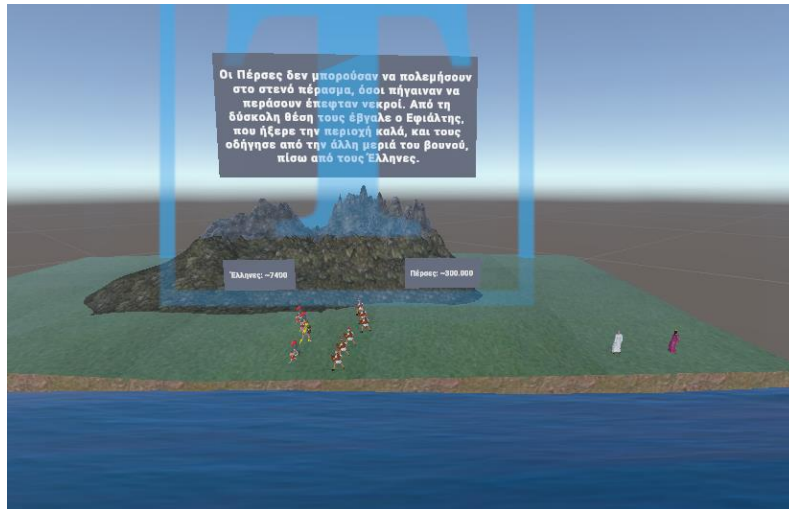
Εικόνα 7.9: Δεύτερη Διαδραστική Σκηνή Θερμοπυλών



Εικόνα 7.10: Ορισμός Διάταξης Στρατού

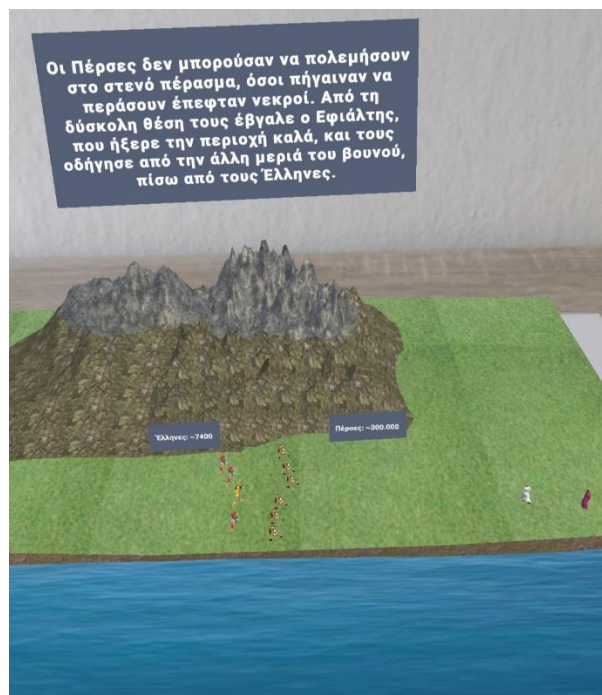
#### 7.4.5 Par3T - Εμφάνιση Εφιάλτη

Όπως έγινε στις προηγούμενες δύο αναπαραστάσεις και σε αυτήν θα χρησιμοποιηθούν τα ίδια ακριβώς αντικείμενα. Στην παρούσα σκηνή θα χρησιμοποιηθεί επιπλέον ο χαρακτήρας Εφιάλτης, ο οποίος εμφανίζεται μόνο σε αυτήν την σκηνή και είναι αυτός που θα δείξει τον δρόμο στους Πέρσες για την άλλη πλευρά. Η τοποθέτηση αλλάζει αρκετά ώστε οι έλληνες στρατιώτες και ο Λεωνίδας να είναι απέναντι από τους Πέρσες και τον Ξέρξη να κάθεται πίσω με τον Εφιάλτη.



Εικόνα 7.11: Τρίτη Σκηνή Αναπαράστασης Θερμοπυλών

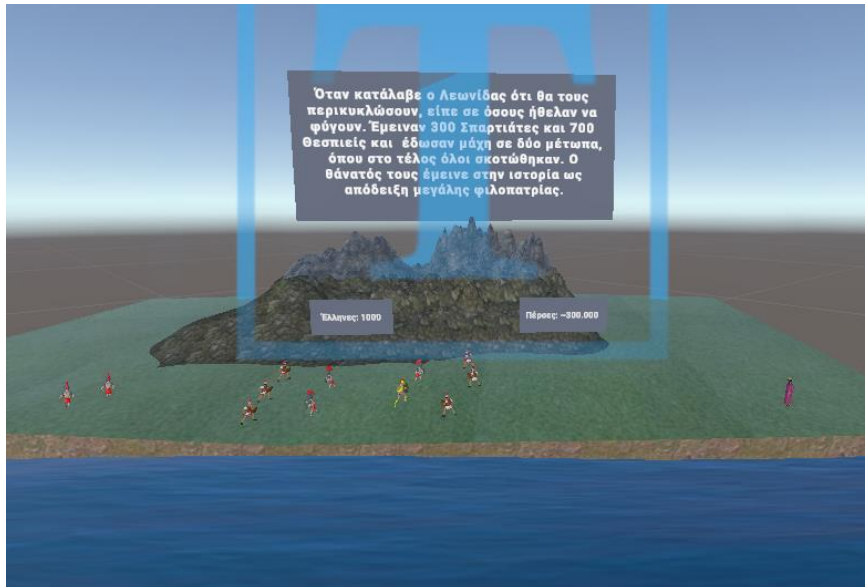
Τα animation στη συγκεκριμένη σκηνή είναι αρκετά. Ο Λεωνίδα, οι Έλληνες και οι Πέρσες στρατιώτες χρησιμοποιούν την κίνηση που πολεμάνε στη διάρκεια της μάχης, ενώ ο Εφιάλτης περιλαμβάνει δύο κινήσεις, οι οποίες εμφανίζονται συνεχόμενα. Η πρώτη δείχνει στον Ξέρξη, ο οποίος κάθεται στάσιμος και τον ακούει, το μονοπάτι που υπάρχει μέσα στο βουνό για να τους βγάλει από την απέναντι πλευρά και το δεύτερο είναι το περπάτημα που κάνει. Έχει οριστεί και ένα position target ώστε όταν ο Εφιάλτης ξεκινήσει το δεύτερο animation, να φαίνεται πως περπατάει ως το σημείο που βρίσκεται το μονοπάτι (έχει αναφερθεί στην ενότητα 4 πως υπάρχει και υλοποιήθηκε αυτό το μονοπάτι) στο βουνό.



Εικόνα 7.12: Animation και Location Στρατιωτών - Εφιάλτη

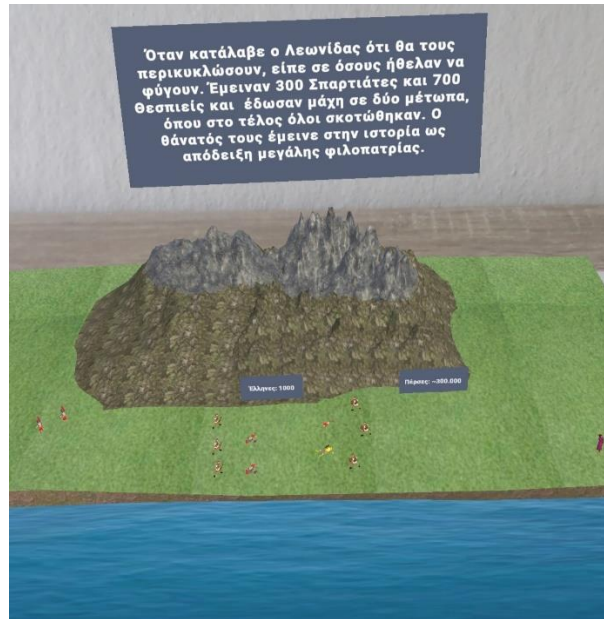
#### 7.4.6 Par4T - Τέλος της Μάχης των Θερμοπυλών

Η τελευταία AR σκηνή της συγκεκριμένης μάχης σχετίζεται με το θάνατο των Ελλήνων που έμειναν και την νίκη των Περσών. Μετά την εμφάνιση του Εφιάλτη που οδήγησε τους Πέρσες από την απέναντι πλευρά, ο Λεωνίδας είπε στους στρατιώτες όποιοι θέλουν να φύγουν πριν περικυκλωθούν από τον περσικό στρατό. Έφυγαν αρκετά άτομα και έμειναν στο τέλος μόνο χίλιοι στρατιώτες (τριακόσιοι Σπαρτιάτες και εφτακόσιοι Θεσπείς) και αναγράφεται στο ανανεωμένο panel των Ελλήνων. Στην περίπτωση των αντικειμένων τώρα, χρησιμοποιήθηκαν επιπλέον δύο έλληνες στρατιώτες που θα συμβολίζουν αυτούς που έφυγαν από την μάχη.



Εικόνα 7.13: Τέταρτη Σκηνή Αναπαράστασης Θερμοπυλών

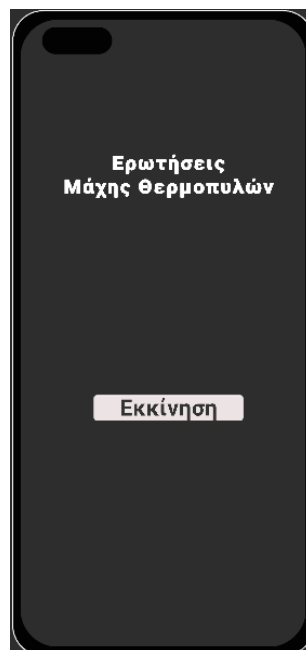
Όσον αφορά τα animation αυτής της σκηνής είναι εξίσου αρκετά στον αριθμό. Αρχικά οι στρατιώτες που βρίσκονται μέσα στην μάχη περιέχουν το ίδιο animation με την προηγούμενη σκηνή, την κίνηση που πολεμάνε μεταξύ τους με μερικούς Πέρσες να βρίσκονται και από την απέναντι πλευρά. Ωστόσο στους Έλληνες έχει προστεθεί μετά από ένα χρονικό διάστημα να σκοτώνονται και να πέφτουν κάτω. Οι υπόλοιπες κινήσεις είναι του Ξέρξη που τον δείχνει να παρακολουθεί από μακριά την εξέλιξη της μάχης και τους δύο Έλληνες στρατιώτες, που προστέθηκαν επιπλέον για αυτή την σκηνή, να τρέχουν μακριά από το πεδίο της μάχης έχοντας για position target, ένα σημείο που βρίσκεται πιο πέρα.



Εικόνα 7.14: Animation και Location Στρατιωτών - Βασιλιάδων

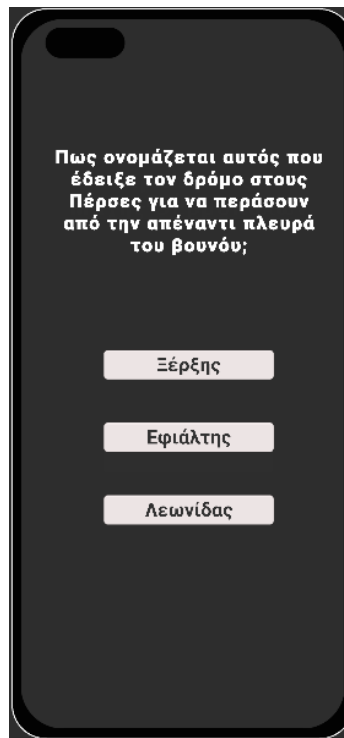
#### 7.4.7 Ερωτήσεις Γνώσεων

Όταν ο χρήστης ολοκληρώσει και με την τελευταία AR σκηνή θα μεταφερθεί στην συγκεκριμένη σκηνή και θα πρέπει να απαντήσει σε ένα κομμάτι δέκα ερωτήσεων για τα όσα συνέβησαν στη μάχη ή για άλλες πληροφορίες που περιείχαν τα panel, όπως αριθμός στρατιωτών κτλ. Πριν ξεκινήσει του εμφανίζεται πρώτα στην οθόνη του η Εικόνα 7.13, η οποία περιέχει ένα text που τον ενημερώνει να πατήσει το κουμπί για να ξεκινήσει το κομμάτι της Μάχης των Θερμοπυλών και ακριβώς από κάτω ένα button.



Εικόνα 7.15: Αρχική του Κομμάτι Θερμοπυλών

Την στιγμή που θα πατηθεί το κουμπί εκκίνησης, τότε θα κρύψει το συγκεκριμένο περιεχόμενο και θα εμφανιστεί ένα text που περιέχει την ερώτηση και τρία buttons με τις πιθανές απαντήσεις της ερώτησης. Από τις απαντήσεις μόνο η μια από τις τρεις θα είναι η σωστή. Ο χρήστης θα πρέπει να απαντήσει και τις δέκα ερωτήσεις για να τελειώσει με την αναπαράσταση. Επιπλέον έχει οριστεί στον κώδικα η μεταβλητή score η οποία θα υπολογίζει τους βαθμούς που κέρδισε συνολικά στο κουίζ.



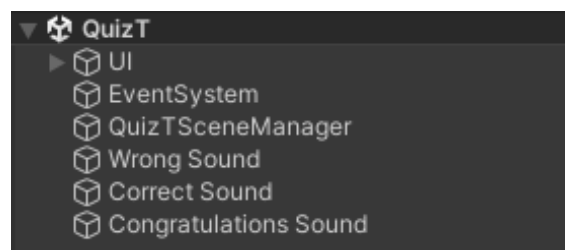
Εικόνα 7.16: Ερωτήσεις Μάχης Θερμοπυλών

Όταν πατηθεί ένα από τα κουμπιά θα γίνεται έλεγχος αν η απάντηση είναι σωστή. Στην περίπτωση της σωστής, θα προστεθεί ένας πόντος στο συνολικό σκορ του χρήστη και θα ακουστεί ανάλογος ήχος και το κουμπί αυτό θα γίνει πράσινο. Αν δοθεί λάθος απάντηση το σκορ παραμένει ίδιο και θα ακουστεί ο ανάλογος ήχος, ενώ το κουμπί που επέλεξε θα γίνει κόκκινο και ταυτόχρονα το κουμπί με την σωστή απάντηση πράσινο. Τέλος σε κάθε απάντηση που δίνει, το πρόγραμμα θα κάνει delay συνολικά τρία δευτερόλεπτα για να συνεχίσει με τις υπόλοιπες ερωτήσεις ή για να τερματιστεί το κουίζ. Μέσα σε αυτά τα τρία δευτερόλεπτα, τα τρία κουμπιά σταματάνε να είναι interactable για να μην μπορεί να τα πατήσει ο χρήστης, και όταν περάσει ο χρόνος και εμφανιστεί η ερώτηση, τότε το χρώμα των κουμπιών γίνεται άσπρο και είναι ξανά interactable.



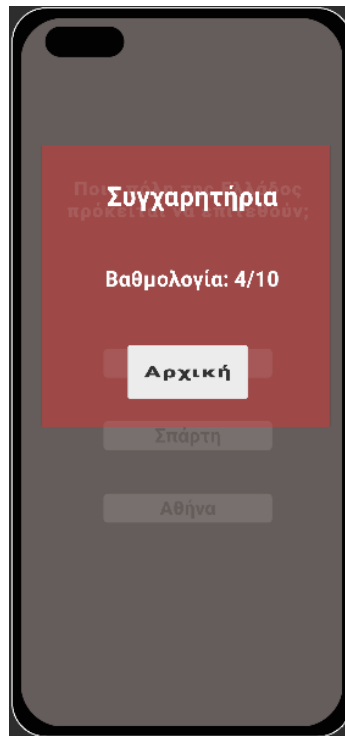
Εικόνα 7.17: Σωστή – Λάθος Απάντηση

Για την προσθήκη των ήχων στο κουίζ, έπρεπε αρχικά να ανέβουν στα Assets του Project, να αποθηκευτούν σε Empty Objects, όπως της Εικόνας 7.17, και μέσα σε αυτά προσθέτουμε από το Add Component “Audio Source” και στο properties “Audio Clip” αποθηκεύεται ο ήχος. Το ίδιο ακριβώς ισχύει και για το κουίζ του επόμενου κεφαλαίου.



Εικόνα 7.18: Εισαγωγή Audio Sources

Στην ολοκλήρωση του κουίζ θα εμφανιστεί ένα panel με το τελικό αποτέλεσμα και ένα ήχο που χρησιμοποιείται σε περιπτώσεις “Congratulations”. Θα εμφανίζεται το τελικό σκορ του κουίζ (ο βαθμός βγαίνει με άριστα το δέκα) και στο πάνω μέρος θα εμφανίζεται το text “Συγχαρητήρια”. Τέλος στο κάτω μέρος του panel υπάρχει ένα button με εικονίδιο που σχετίζεται με την Αρχική, το οποίο μόλις πατηθεί θα μεταφερθεί πίσω στο Menu της εφαρμογής.



Εικόνα 7.19: Panel Αποτελέσματος Κουίζ

## 7.5 Επίλογος

Σε αυτήν την ενότητα αναφέρθηκε μια σύντομη ιστορική αναφορά της μάχης των Θερμοπυλών. Η δομή του φακέλου που βρίσκεται όλο το υλικό της αναπαράστασης και η διαδικασία υλοποίησης της μάχης, δηλαδή όσα αντικείμενα-animation χρησιμοποιήθηκαν στις σκηνές και οι πληροφορίες στη συνέχεια για το περιεχόμενο του κουίζ.



## Κεφάλαιο 8ο: Ναυμαχία της Σαλαμίνας

### 8.1 Εισαγωγή

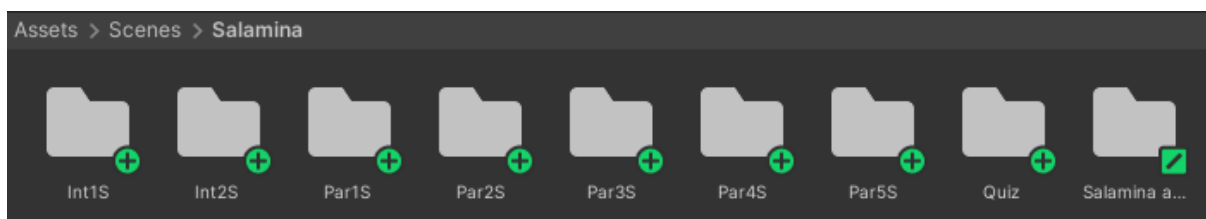
Στο κεφάλαιο οκτώ θα αναφερθεί μια σύντομη ιστορική αναφορά για την ναυμαχία, η οποία βασίζεται επίσης στην ύλη που βρίσκεται στο βιβλίο του μαθήματος Ιστορία Δ' Δημοτικού. Επιπλέον, όπως και στη μάχη των Θερμοπυλών, θα αναφερθεί στα επόμενα υποκεφάλαια αναλυτικά όλη η διαδικασία υλοποίησης της κάθε σκηνής.

### 8.2 Ιστορική Αναδρομή

Η Ναυμαχία της Σαλαμίνας έγινε το 480π.Χ λίγο αργότερα από την Μάχη των Θερμοπυλών. Μετά την νίκη των Περσών στα στενά των Θερμοπυλών ο βασιλιάς Ξέρξης και ο στρατός του συνέχισαν την πορεία τους μέχρι την Αθήνα, όπου εκεί παρατήρησαν πως ήταν έρημη η πόλη, καθώς τα γυναικόπαιδα και οι ηλικιωμένοι μεταφέρθηκαν στα γύρω νησιά και οι άντρες είχαν πάει στα πλοία να πολεμήσουν εναντίον τους για δεύτερη φορά. Το πρόβλημα του ελληνικού στόλου ήταν το σημείο που θα τοποθετηθεί για να αντιμετωπίσει τον περσικό στόλο. Οι δύο αρχηγοί του στόλου, ο Σπαρτιάτης Ευρυβιάδης και ο Αθηναίος Θεμιστοκλής είχαν έντονη λογομαχία για αυτήν την περίπτωση, καθώς ο πρώτος πρότεινε να τοποθετηθεί ο στόλος στον ισθμό της Κορίνθου και ο δεύτερος στο στενό της Σαλαμίνας. Εν τέλει κατέληξαν να συμφωνήσουν με την πρόταση του Θεμιστοκλή, λόγω της δυσκολίας του περσικού στόλου να περάσει μέσα στο στενό. Όταν ξεκίνησε η ναυμαχία, τα ελληνικά πλοία ήταν λιγότερα από τα περσικά, αλλά μπορούσαν πιο εύκολα να κινηθούν μέσα στο στενό, και στο τέλος κατάφεραν να νικήσουν τους Πέρσες. Η νίκη έσωσε την Ελλάδα και ήταν η αρχή του τέλους της περσικής εισβολής[35].

### 8.3 Δομή Φακέλου Salamina

Ο φάκελος Salamina περιέχει όλο το περιεχόμενο της αναπαράστασης της ναυμαχίας. Ο τρόπος που έχει οριστεί η δομή του φαίνεται στην Εικόνα 8.1, και είναι ακριβώς ίδια με την δομή του φακέλου Thermopyles, δηλαδή τα Int αναφέρονται στις διαδραστικές σκηνές, τα Par για τις αναπαραστάσεις και το Quiz για τις ερωτήσεις στο τέλος. Η μόνη τους διαφορά είναι το γράμμα S στο τέλος αντί για το T που έχει στις Θερμοπύλες.



Εικόνα 8.1: Δομή Φακέλου Salamina

Το περιεχόμενο του κάθε υποφακέλου είναι ίδιο με αυτό που αναφέρθηκε στο προηγούμενο κεφάλαιο που σχετίζεται με τις Θερμοπύλες. Ο τρόπος εισαγωγής των τρισδιάστατων αντικειμένων, των `c# script`, των `animation` και των `animation controllers` δεν διαφέρουν καθόλου από τον άλλο φάκελο. Η μόνη τους διαφορά πάλι είναι το γράμμα S που χρησιμοποιείται σε όλες τις ονομασίες. Στην Εικόνα 8.2 παρουσιάζεται το περιεχόμενο των φακέλων Int, Par και Quiz.



Εικόνα 8.2: Δομή υποφακέλων Salamina

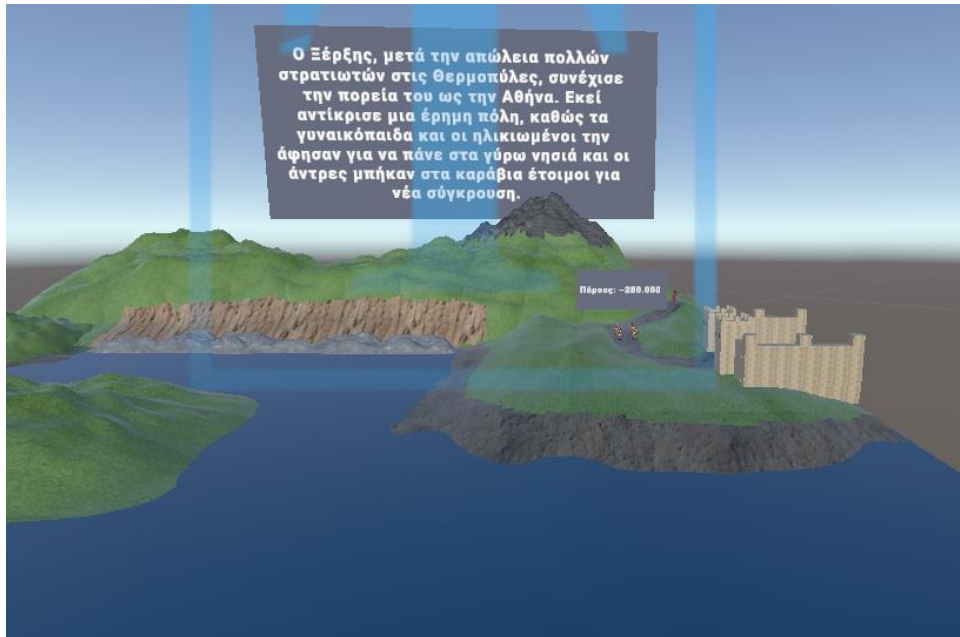
## 8.4 Δημιουργία Σκηνών Αναπαράστασης

Για την Ναυμαχία της Σαλαμίνας χρησιμοποιήθηκαν δύο διαδραστικές σκηνές, πέντε σκηνές για τα σημαντικά στιγμιότυπα της αναπαράστασης και στο τέλος είναι η σκηνή που περιέχει το τελικό κουίζ που θα πρέπει να απαντήσει ο χρήστης με βάση αυτά που είδε στην αναπαράσταση. Όπως ήταν για τις Θερμοπύλες και εδώ υπάρχουν τα panel με ιστορική αναφορά, για την αποφυγή επανάληψης, οι σκηνές αλλάζουν αυτόματα και μετά από είκοσι δευτερόλεπτα συνεχίζει με την επόμενη.

Όπως έγινε και στο κεφάλαιο εφτά, οι σκηνές του Κεφαλαίου 8.4 θα παρουσιαστούν με την σειρά που θα εμφανίζονται στην τελική εφαρμογή.

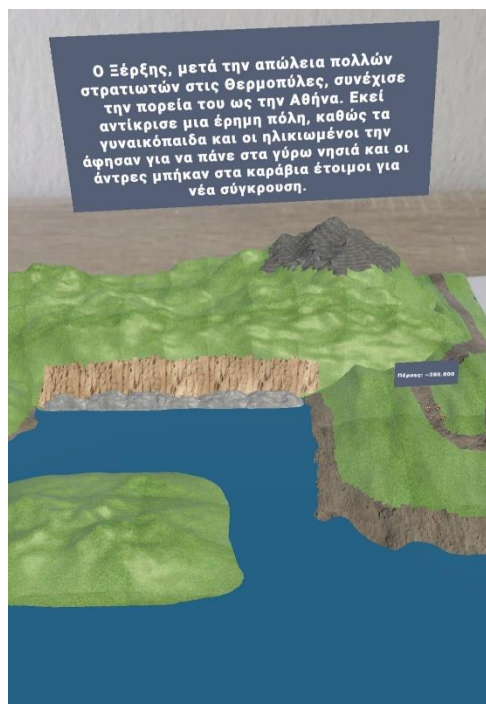
### 8.4.1 Par1S - Οι Πέρσες φτάνουν στην Αθήνα

Η πρώτη σκηνή της ναυμαχίας είναι η αναπαράσταση με τα σημαντικά στιγμιότυπα, σε αντίθεση με αυτή των Θερμοπυλών που ξεκίνησε με διαδραστική σκηνή. Στο περιεχόμενό της έχει για panels ένα για την ιστορική αναφορά της σκηνής, και άλλο ένα που περιέχει τον συνολικό αριθμό των περσών στρατιωτών. Χρησιμοποιείται ο ίδιος που είχε στην προηγούμενη αναπαράσταση, άλλα το περιεχόμενό του έχει διαφορετικό αριθμό, λόγω της μεγάλης απώλειας στρατιωτών που είχε τότε στην μάχη. Τα αντικείμενα που χρησιμοποιήθηκαν στη σκηνή είναι τέσσερις πέρσες στρατιώτες και ο βασιλιάς Ξέρξης.



Εικόνα 8.3: Πρώτη Σκηνή Αναπαράστασης Σαλαμίνας

Στην περίπτωση του animation, υπάρχει μόνο ένα και θα χρησιμοποιηθεί και από τα πέντε αντικείμενα της σκηνής. Το animation είναι το περπάτημα που θα έκαναν μέχρι να φτάσουν στην έρημη πόλη της Αθήνας. Έχουν προστεθεί position targets για κάθε αντικείμενο και να πηγαίνουν προς το συγκεκριμένο σημείο μέχρις ότου περάσει το χρονικό όριο που έχει οριστεί.



Εικόνα 8.4: Animation και Location Περσών

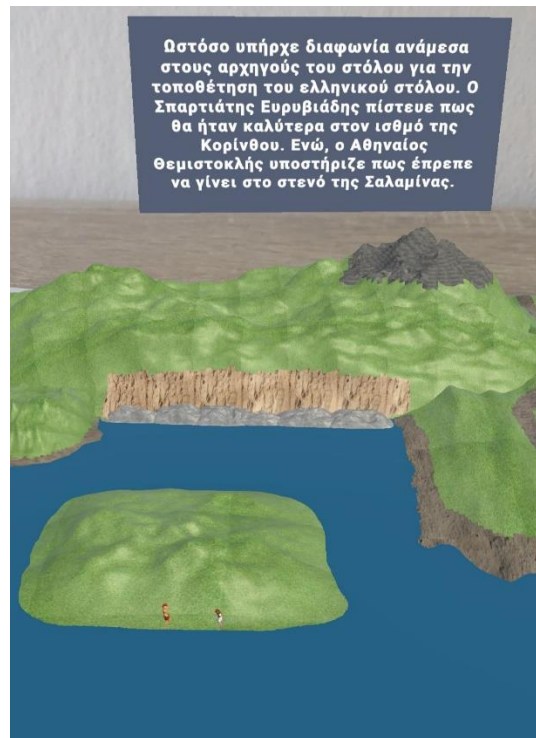
### 8.4.2 Par2S - Λογομαχία Ευρυβιάδη και Θεμιστοκλή

Η δεύτερη αναπαράσταση δείχνει τους δύο αρχηγούς του ελληνικού στόλου, τον Ευρυβιάδη και τον Θεμιστοκλή να έχουν μια έντονη λογομαχία μεταξύ τους. Από τα panels θα χρησιμοποιηθεί μόνο το κύριο panel του οποίου το περιεχόμενο θα διαφέρει από την προηγούμενη σκηνή. Στα τρισδιάστατα αντικείμενα θα χρησιμοποιηθούν δύο καινούργια και μόνο σε αυτήν την σκηνή θα εμφανιστούν, τα οποία είναι οι χαρακτήρες των αρχηγών του ελληνικού στόλου.



Εικόνα 8.5: Δεύτερη Σκηνή Αναπαράστασης Σαλαμίνας

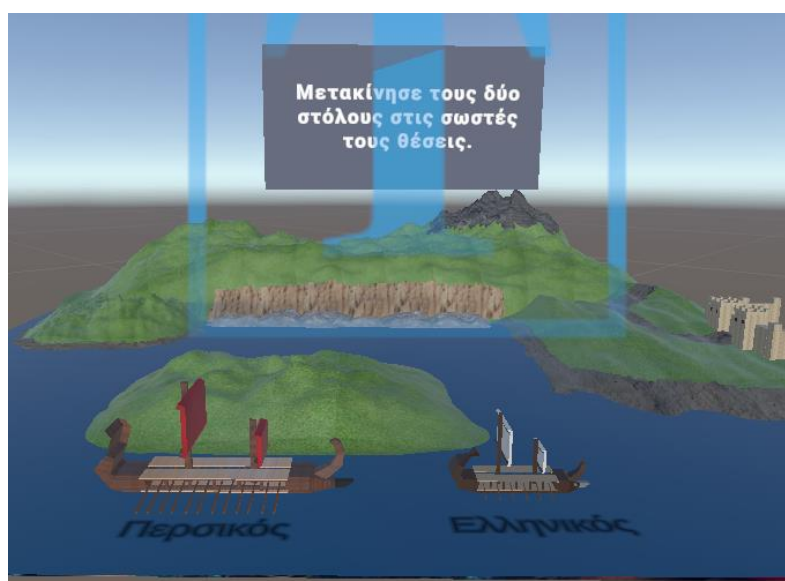
Και τα δύο αντικείμενα έχουν για κίνηση την έντονη λογομαχία που υπάρχει μεταξύ τους. Από τις κινήσεις που κάνουν φαίνεται ότι μιλάνε με πιο έντονο ύφος, επίσης μπορεί να παρατηρηθεί πως ο καθένας κάνει διαφορετικές κινήσεις από τον άλλον. Και τα δύο είναι σε loop time μέχρι να περάσουν τα είκοσι δευτερόλεπτα.



Εικόνα 8.6: Animation Ευρυβιάδη και Θεμιστοκλή

### 8.4.3 Int1S – Στόλοι

Μετά την λογομαχία των δύο αρχηγών του στόλου για την τοποθέτηση του στόλου, στη διαδραστική σκηνή θα πρέπει να ορίσει την θέση των δύο στόλων μέσα στην περιοχή της Σαλαμίνας. Στο Panel αναφέρεται το παραπάνω ζητούμενο και τα αντικείμενα που πρέπει να μετακινήσει φαίνονται στην Εικόνα 8.7 και είναι τα δύο πλοία της εφαρμογής, του ελληνικού και περσικού στόλου, τα οποία μπορεί να τα ξεχωρίσει ο χρήστης, επειδή ακριβώς από κάτω αναγράφεται σε ποιο στόλο ανήκει το κάθε πλοίο.



Εικόνα 8.7: Πρώτη Διαδραστική Σκηνή Σαλαμίνας

Σε αντίθεση με τις διαδραστικές σκηνές της αναπαράστασης των Θερμοπυλών, εδώ δεν υπάρχουν ειδικά panel. Θα πρέπει τα πλοία να μετακινηθούν μέσα στη θαλάσσια περιοχή της αναπαράστασης και για αυτό το λόγο είναι στην επιλογή του καθενός για το πως θα τοποθετήσουν τα πλοία στις θέσεις τους. Θα μπορούν για τα επόμενα είκοσι δευτερόλεπτα να τα αλλάζουν θέσεις μέχρι να τελειώσει ο χρόνος.



Εικόνα 8.8: Μετακίνηση Στόλων

#### 8.4.4 Par3S - Μετακίνηση Ελληνικού στόλου

Ο Θεμιστοκλής έδωσε εντολή τον ελληνικό στόλο να μετακινηθεί προς το στενό της Σαλαμίνας και να περιμένει εκεί τον περσικό. Στην σκηνή θα δείχνει τρία ελληνικά πλοία μαζί με ένα Panel που δείχνει τον αριθμό των πλοίων του στόλου. Η τιμή αυτή βασίστηκε από ιστοσελίδες στο διαδίκτυο καθώς δεν υπήρχε κάποιος ξεκάθαρος αριθμός.



Εικόνα 8.9: Τρίτη Σκηνή Αναπαράστασης Σαλαμίνας

Και τα τρία ελληνικά πλοία θα έχουν για animation την κίνηση των κουπιών μέσα στο νερό και θα λειτουργεί η κίνηση μέχρι να περάσουν τα είκοσι δευτερόλεπτα. Επίσης, έχουν τοποθετηθεί και τρία position targets, που αντιστοιχεί ένα για κάθε πλοίο και θα κινηθούν μέσα στο στενό της Σαλαμίνας.



Εικόνα 8.10: Animation και Location Ελληνικού Στόλου

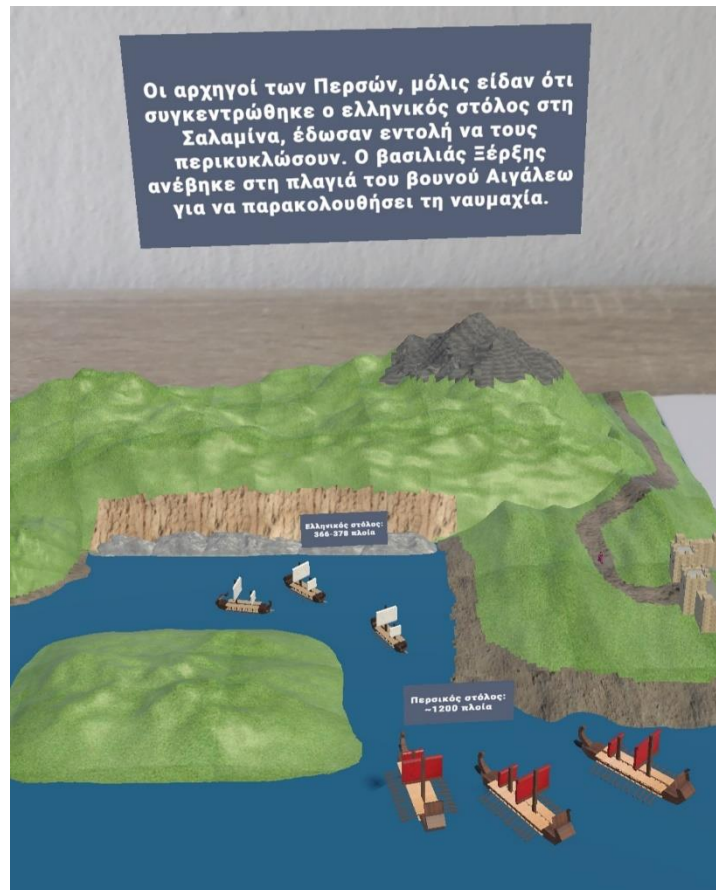
### 8.4.5 Par4S - Μετακίνηση Περσικού στόλου

Μόλις είδαν οι Πέρσες ότι ο ελληνικός στόλος μαζεύτηκε μέσα στο στενό αμέσως δόθηκε η εντολή να πάνε προς το συγκεκριμένο σημείο. Έτσι, στην σκηνή προστέθηκαν επιπλέον και τρία περσικά πλοία μαζί με ένα αντίστοιχο Panel που έχει και ο ελληνικός στόλος, το οποίο αναφέρεται στο συνολικό αριθμό πλοίων του περσικού στόλου. Στην σκηνή βρίσκεται και ο Ξέρξης που ετοιμάζεται να δει την μάχη.



Εικόνα 8.11: Τέταρτη Σκηνή Αναπαράστασης Σαλαμίνας

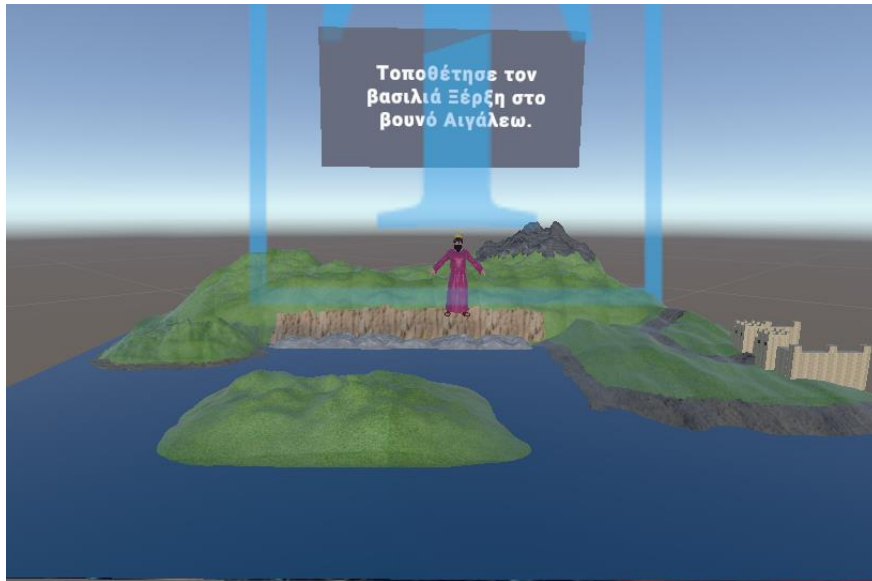
Και τα τρία περσικά πλοία θα έχουν το ίδιο animation που είχαν τα ελληνικά στην προηγούμενη αναπαράσταση, το οποίο είναι η κίνηση των κουπιών μέσα στο νερό και θα λειτουργεί επίσης μέχρι να περάσουν τα είκοσι δευτερόλεπτα. Τα ελληνικά πλοία θα μείνουν σταθερά στο ίδιο σημείο περιμένοντας τα περσικά πλοία να πλησιάσουν. Στην σκηνή υπάρχουν πάλι τρία position targets, που αντιστοιχεί ένα για κάθε πλοίο των Περσών, και τα σημεία αυτά βρίσκονται μπροστά από τον ελληνικό στόλο ώστε να ξεκινήσει η ναυμαχία. Ο βασιλιάς Ξέρξης θα έχει το animation Walk, όπου θα τον δείχνει να πηγαίνει από την έρημη πόλη της Αθήνας προς το βουνό Αιγάλεω.



Εικόνα 8.12: Animation και Location Περσικού Στόλου - Ξέρξης

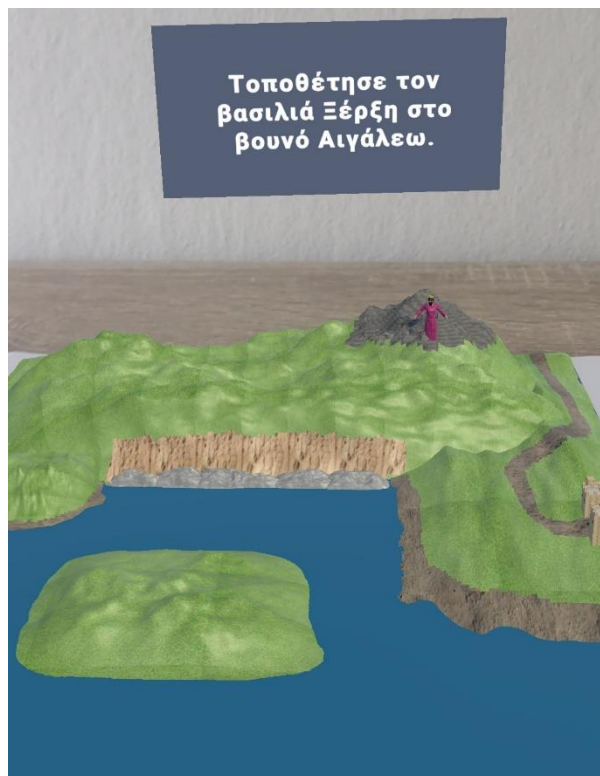
#### 8.4.6 Int2S - Ξέρξης

Πριν ξεκινήσει η ναυμαχία, ο βασιλιάς Ξέρξης ετοιμάζεται να πάει σε κάποιο ύψωμα για να δει από εκεί τη μάχη. Λόγω του μέρους που πρόκειται να πραγματοποιηθεί η μάχη, ο Ξέρξης επέλεξε το βουνό Αιγάλεω για να την παρακολουθήσει από εκεί. Στην δεύτερη και τελευταία διαδραστική σκηνή της ναυμαχίας, ο χρήστης θα πρέπει να μετακινήσει τον Ξέρξη πάνω στο βουνό. Υπάρχει το panel που του ζητάει να κάνει αυτό μέσα στα επόμενα είκοσι δευτερόλεπτα. Σε αυτήν την σκηνή δεν υπάρχει ούτε ειδικό Panel για να τον τοποθετήσει σε αυτό το σημείο, όπως ήταν και στην πρώτη διαδραστική σκηνή, αλλά ούτε κάποιο text που να αναφέρει ποιος είναι, επειδή είναι μόνο ένα αντικείμενο και αναφέρεται στο Panel ποιος είναι, και για αυτό το λόγο θα ήταν περιττό μέσα στη σκηνή.



Εικόνα 8.13: Δεύτερη Διαδραστική Σκηνή Σαλαμίνας

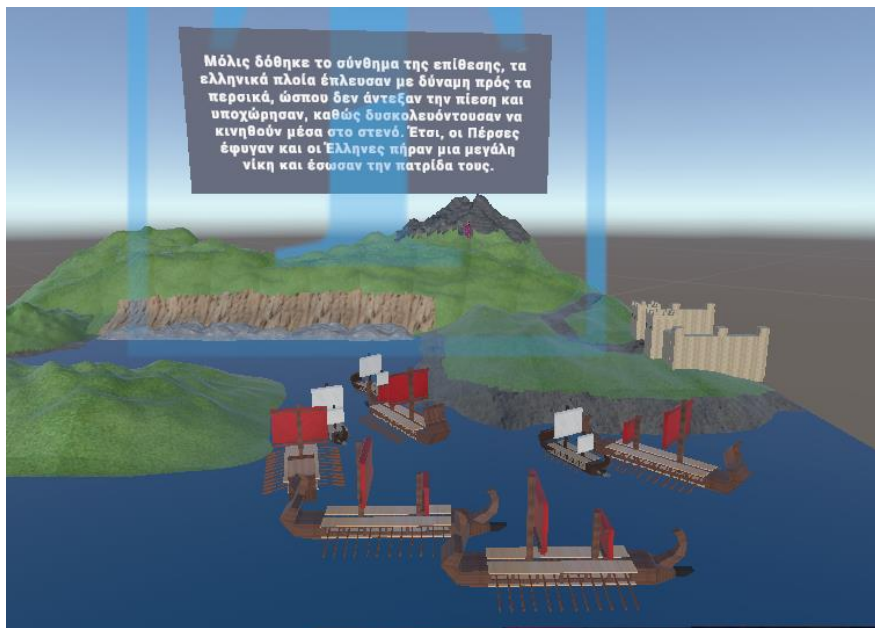
Όπως αναφέρθηκε στην παραπάνω παράγραφο, δεν υπάρχει κάποιο ειδικό panel. Θα μπορεί να τον μεταφέρει ο χρήστης σε οποιοδήποτε σημείο της περιοχής μέχρι να τελειώσει ο διαθέσιμος του χρόνος. Όταν περάσει το χρονικό όριο, τότε θα μεταφερθεί στην τελευταία αναπαράσταση της ναυμαχίας, χωρίς και εδώ να πραγματοποιείται έλεγχος για την σωστή τοποθέτηση του Ξέρξη.



Εικόνα 8.14: Μετακίνηση Βασιλιά Ξέρξη

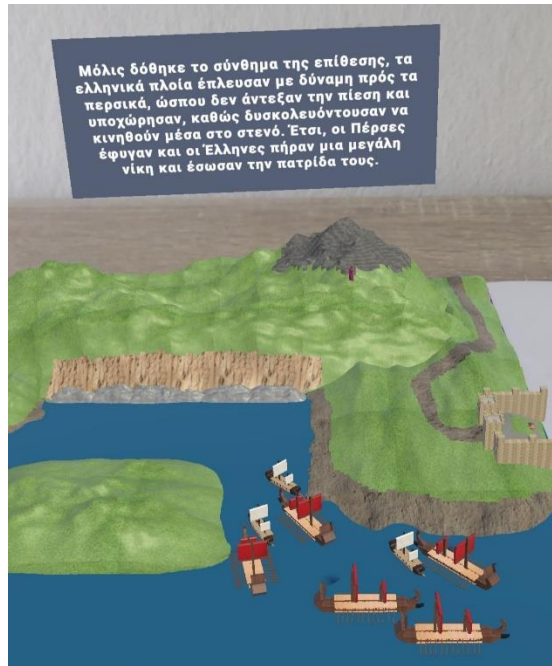
### 8.4.7 Par5S - Τέλος της Ναυμαχίας της Σαλαμίνας

Την στιγμή που δόθηκε το σύνθημα να ξεκινήσει η μάχη, τα πλοία και των δύο στόλων της τέταρτης αναπαράστασης είναι το ένα δίπλα από το άλλο και έχουν ξεκινήσει να πολεμάνε μεταξύ τους. Την ώρα της μάχης είναι και ο βασιλιάς Ξέρξης που κάθεται και παρακολουθεί την μάχη από ψηλά. Τα Panels που αναγράφουν τον αριθμό των δύο στόλων έχουν αφαιρεθεί από την σκηνή για να φαίνεται καλύτερα η μάχη. Στα περσικά πλοία έχουν προστεθεί δύο ακόμη, τα οποία θα είναι αυτά που θα αποχωρήσουν από την μάχη.



Εικόνα 8.15: Πέμπτη Σκηνή Αναπαράστασης Σαλαμίνας

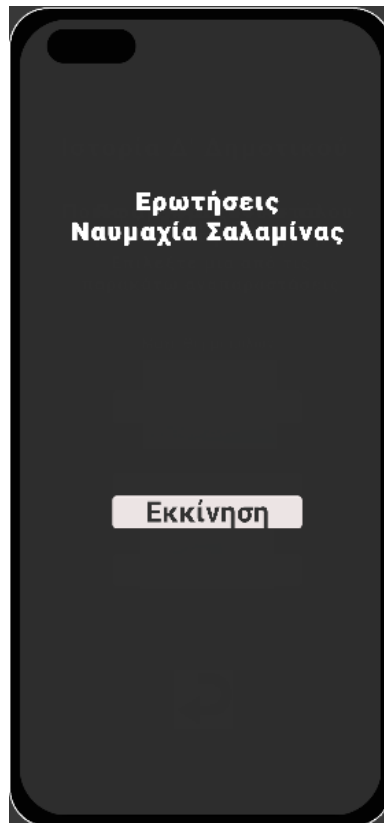
Το animation που έχουν τα δύο περσικά πλοία, θα είναι το ίδιο με τις δύο προηγούμενες σκηνές που είχαν την κίνηση των πλοίων, δηλαδή η αλλαγή θέσης των κουπιών μέσα στη θάλασσα. Τα υπόλοιπα πλοία θα μείνουν σταθερά στο ίδιο σημείο για να φαίνεται στην αναπαράσταση πως πολεμάνε μεταξύ τους. Υπάρχουν επίσης δύο position targets, και τα μεταφέρει μακριά από το πεδίο μάχης μέχρι να τελειώσει η αναπαράσταση.



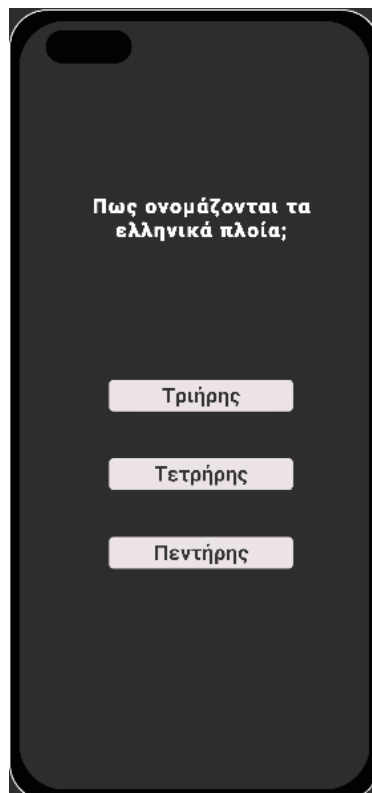
Εικόνα 8.16: Animation και Location Αναχώρησης Περσικού Στόλου

#### 8.4.8 Ερωτήσεις Γνώσεων

Μόλις ολοκληρωθεί και η τελευταία AR σκηνή, ο χρήστης θα μεταφερθεί στην σκηνή που θα πρέπει να απαντήσει σε ένα κουίζ δέκα ερωτήσεων για τα όσα συνέβησαν στη ναυμαχία ή για άλλες πληροφορίες που περιείχαν τα panel, όπως αριθμός πλοίων κτλ. Ο τρόπος που έχει υλοποιηθεί είναι ακριβώς ίδιος με την περίπτωση των Θερμοπυλών στο προηγούμενο κεφάλαιο. Συγκρίνοντας τις Εικόνες 7.14-7.15 με τις Εικόνες 8.17-8.18, ο τρόπος που έχουν τοποθετηθεί στο UI και οι λειτουργίες που έχουν, τα κουμπιά εκκίνησης και η σειρά που εμφανίζονται οι ερωτήσεις, είναι ακριβώς ίδια και στα δύο, ωστόσο αλλάζει το περιεχόμενο της αρχικής οθόνης του κουίζ και των ερωτήσεων.



Εικόνα 8.17: Αρχική Κουίζ Σαλαμίνας



Εικόνα 8.18: Ερωτήσεις Ναυμαχίας Σαλαμίνας

Όπως έγινε με την διεπαφή του κουίζ, και η λειτουργία του υπολογισμού σκορ είναι ακριβώς ίδιος και στην περίπτωση της Σαλαμίνας. Θα μαζεύει δηλαδή τους πόντους και θα πάρει έναν τελικό βαθμό με άριστα το 10. Και στην περίπτωση των απαντήσεων η εμφάνιση των σωστών και λάθος απαντήσεων είναι ίδιος με την Εικόνα 7.16, που για τις σωστές θα αλλάζει το background color του button σε πράσινο και στις λάθος σε κόκκινο.

Τέλος, στην εμφάνιση του τελικού αποτελέσματος, το UI θα είναι ακριβώς ίδιο με την Εικόνα 7.17, και η λειτουργία του button θα μεταφέρει τον χρήστη στο μενού της εφαρμογής.

### **8.5 Επίλογος**

Σε αυτήν την ενότητα αναφέρθηκε η σύντομη ιστορική αναφορά της ναυμαχίας της Σαλαμίνας, η δομή του φακέλου που βρίσκεται όλο το υλικό και η διαδικασία υλοποίησης της ναυμαχίας της Σαλαμίνας. Στο τέλος, μετά την ολοκλήρωση της αναπαράστασης, παρουσιάζεται περιληπτικά το περιεχόμενο του κουίζ.

## Κεφάλαιο 9ο: Συμπεράσματα και Προτάσεις Βελτίωσης

Μετά από όλα αυτά που ειπώθηκαν παραπάνω, συμπεραίνουμε ότι η Ε.Π. είναι ένα αρκετά σημαντικό εργαλείο, και με την ραγδαία εξέλιξή της τα τελευταία χρόνια, θα αλλάξει ριζικά τον τρόπο εκπαίδευσης και εκμάθησης, όχι μόνο των μαθητών, αλλά και των εκπαιδευτικών. Από την στιγμή που έχει θετικό αντίκτυπο από τους χρήστες, θα συνεχίσουν να προκύπτουν αρκετές ακόμη εφαρμογές AR, με πολλές από αυτές να υλοποιούνται με επιτυχία και να χρησιμοποιούνται και ως συμπληρωματικά εργαλεία στο κομμάτι της εκπαίδευσης.

Μπορεί η υλοποίηση μιας τέτοιας εφαρμογής για τις κινητές συσκευές να είναι αρκετά χρονοβόρα και να κοστίζει αρκετά για την δημιουργία της, όμως έχει τόσα πολλά πλεονεκτήματα μέσα στην εκπαίδευση, που θα χρησιμοποιούνται από αρκετούς χρήστες, λόγω της καλής αποδοτικότητας. Ειδικά στη περίπτωση του Storytelling, οι μαθητές επιλέγουν τέτοιου είδους εφαρμογές για το μάθημα της ιστορίας, γιατί προτιμάνε να δούνε μια αναπαράσταση με διάφορα τρισδιάστατα αντικείμενα από το να διαβάζουν απλά ένα κείμενο με το οποίο εύκολα χάνουν το ενδιαφέρον λόγω της μονοτονίας της.

Λαμβάνοντας υπόψη με ό,τι έχει αναφερθεί πιο πάνω, για την εφαρμογή θα μπορούσαν να προστεθούν καινούρια χαρακτηριστικά για να βελτιωθεί η διεπαφή αλλά και η εμπειρία χρήσης της. Το πρώτο και κυριότερο, είναι η υλοποίηση της εφαρμογής και για συσκευές με λειτουργικό iOS, για να αυξηθούν περισσότερο οι χρήστες της εφαρμογής, καθώς αρκετοί χρησιμοποιούν συσκευές με το συγκεκριμένο λειτουργικό. Η βελτίωση της διεπαφής και η προσθήκη progress bar στο αρχικό panel της κάθε αναπαράστασης θα ήταν εξίσου ιδανικά και να μην έχει μόνο απλά texts και buttons. Επιπλέον, στις διαδραστικές σκηνές, όπου ο χρήστης μετακινεί αντικείμενα, θα ήθελα να προσθέσω έλεγχο που να βλέπει τις τοποθετήσεις των αντικειμένων μέσα στην σκηνή και να βλέπει αν είναι σωστές ή όχι για να τις διορθώνει.

Τέλος, στα σχέδια της επέκτασης είναι και η δημιουργία περισσότερων αναπαραστάσεων, από την ύλη του μαθήματος της Ιστορίας Δ' Δημοτικού και να προστεθούν στην εφαρμογή, ώστε ο χρήστης να έχει περισσότερες αναπαραστάσεις να επιλέξει και με αυτόν τον τρόπο να καλυτερεύσει η εμπειρία χρήσης της εφαρμογής.

## **ΒΙΒΛΙΟΓΡΑΦΙΑ**

- [1] J. Motejlek, E. Alpay, “A taxonomy for virtual and augmented reality in education”, 2018
- [2] D. Amin, S. G.-I. J., “Comparative study of augmented reality SDKs”, 2015
- [3] O. Hughes, P. Fuchs, O. Nannipieri, “New Augmented Reality Taxonomy: Technologies and Features of Augmented Environment”, 2011
- [4] D. Vergara, M. P. Rubio, M. Lorenzo, “On the design of virtual reality learning environments in engineering”, 2017
- [5] M. A. Livingston, “Military Applications of Augmented Reality”, 2011
- [6] P. Milgram, “A taxonomy of Mixed Reality Visual displays”, IEICE Transactions on Information Systems, no. 12, 1994 [Online]. Available: [http://vered.rose.utoronto.ca/people/paul\\_dir/IEICE94/ieice.html](http://vered.rose.utoronto.ca/people/paul_dir/IEICE94/ieice.html)
- [7] L. Cen, D. Ruta, L. M. M. S. al Qassem, J. Ng, “Augmented Immersive Reality (AIR) for Improved Learning Performance: A Quantitative Evaluation”, 2020
- [8] J. Radiant, T. A. Majchrzak, J. Fromm, I. Wohlgenannt, “A systematic review of immersive virtual reality applications for higher education: Design elements, lesson learned, and research agenda”, 2019
- [9] J. Motejlek, E. Alpay, “A Taxonomy for Virtual and Augmented Reality in Education”, 2021
- [10] E. K. Layland, G. A. Stone, J. T. Mueller, C. J. Hodge, “Injustice in Mobile Leisure: A Conceptual Exploration of Pokemon GO”, 2018
- [11] J. Miranda, C. Navarrete, “The core components of education 4.0 in higher education: Three case studies in engineering education”, 2021
- [12] R. Kaviyarai, M. Uma, “A survey on Future of Augmented Reality with AI in Education”, 2021
- [13] T. Mullen, “Prototyping augmented reality”, 2021
- [14] A. Binoy, “PUMP FIT: An Augmented Reality App which helps people with their workout more efficiently”, 2020
- [15] S. J. Kim, A. K. Dey “Simulated augmented reality windshield display as cognitive mapping aid for elder driver navigation”, 2011
- [16] V. Rossano, R. Lanzilotti, A. Cazzolla, T. Roselli. “Augmented Reality to Support Geometry Learning”, 2020
- [17] S. Sureshkumar, C. P. Agash, S. Ramya, R. Kaviyaraj, S. Elanchezhivan, “Augmented Reality with Internet of Things”, 2021
- [18] S. Jangra, G. Singh, A. Mantri, S. Angra, B. Sharma, “Interactivity Development Using Unity 3D Software and C# Programming”, 2023.
- [19] A. Juliani, V. P. Berges, A. Cohen, J. Harper, C. Elion, D. Large, “Unity: A general platform for intelligent agents”, 2018

- [20] X. Liull, Y. H. Sohn, D. W Park, “Application development with augmented reality technique using unity 3D and Vuforia”, 2018.
- [21] Unity, “unity3d.com” <https://docs.unity3d.com/Manual/system-requirements.html>
- [22] K. Buse Asena, C. Burakhan, Y. Ugur, “Augmented Reality Application for Preschool Children with Unity 3D Platform”, 2019.
- [23] C. Meenu, S. Gurinder, G. Loveleen, M. Nidhi, K. Shikha, “Leveraging Unity 3D and Vuforia Engine for Augmented Reality Application Development”, Nov. 2023.
- [24] M. Gnana Prakash, A. Ponmalar, S. Deeba, Dr. A. Akilandeswari, S. B. M. Rasool, P. Lavanya, “VSCODE – Code With Voice Using Natural Language Processing (NLP)”, Oct. 2022.
- [25] Hao Bai, “VSC-WebGPU”, Nov. 2023.
- [26] G. Koulaxidis, S. Xinogalos, “Improving Mobile Game Performance with Basic Optimization Techniques in Unity”, 2022.
- [27] A. Parab, N. Rathod, T. Patil, K. Deshpande, N. Deshmukh, “A 3D Storyline Using Unity Game Engine”, 2022.
- [28] A. Chia, “The metaverse, but not the way you think: Game engines and automation beyond game development”, 2022.
- [29] A. Okita, “Learning C# programming with Unity 3D”, 2019.
- [30] S. Lav, K. Amanpreet, S. Aninash, “A Review on Different Versions and Interfaces of Blender Software”, 2023.
- [31] L. Flavell, *Begging Blender Lance Flavell Full Color Inside*. Apress, 2011.
- [32] Blender, “Blender.org” <https://blender.org/download/releases/>
- [33] F. Alecu, “Blender Institoute-the institute for Open 3D Projects”, *Open Source Sci. J.*, vol. 2, no. 1, pp. 36-45, 2010.
- [34] “Ιστορία Δ’ Δημοτικού”, “[//ebooks.edu.gr/ebooks/v/html/8547/2174/Istoria\\_D-Dimotikou\\_html-empl/index3\\_17.html](http://ebooks.edu.gr/ebooks/v/html/8547/2174/Istoria_D-Dimotikou_html-empl/index3_17.html)”
- [35] “Ιστορία Δ’ Δημοτικού”, “[ebooks.edu.gr/ebooks/v/html/8547/2174/Istoria\\_D-Dimotikou\\_html-empl/index3\\_18.html](http://ebooks.edu.gr/ebooks/v/html/8547/2174/Istoria_D-Dimotikou_html-empl/index3_18.html)”

## ΠΑΡΑΡΤΗΜΑ Α : ΚΩΔΙΚΑΣ SCRIPT Home

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class Home : MonoBehaviour{

    [SerializeField] GameObject quitPanel;

    public void onPlay(){
        SceneManager.LoadScene("Menu");
    }

    public void onInfo(){
        SceneManager.LoadScene("Info");
    }

    public void onExit(){
        quitPanel.SetActive(true);
    }

    public void onYes(){
        Application.Quit();
    }

    public void onNo(){
        quitPanel.SetActive(false);
    }
}
```

## ΠΑΡΑΡΤΗΜΑ Β : ΚΩΔΙΚΑΣ SCRIPT Info

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class Info : MonoBehaviour{

    // Start is called before the first frame update
    void Start()
    {

    }
}
```

```

public void onHome(){
    SceneManager.LoadScene("Home");
}
}

```

## **ΠΑΡΑΡΤΗΜΑ C : ΚΩΔΙΚΑΣ SCRIPT Menu**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class Menu : MonoBehaviour{

    public void onThermopyles(){
        SceneManager.LoadScene("StartT");
    }

    public void onSalamina(){
        SceneManager.LoadScene("StartS");
    }

    public void onBack(){
        SceneManager.LoadScene("Home");
    }
}

```

## **ΠΑΡΑΡΤΗΜΑ D : ΚΩΔΙΚΑΣ SCRIPT StartT**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
using UnityEngine.UI;

```

```

public class StartT : MonoBehaviour
{
    public GameObject panel;

    void Start(){
        panel.SetActive(true);
    }

    public void StartTher(){
        SceneManager.LoadScene("Int1T");
    }
}

```

## **ΠΑΡΑΡΤΗΜΑ Ε : ΚΩΔΙΚΑΣ SCRIPT StartS**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
using UnityEngine.UI;

public class StartS : MonoBehaviour
{
    public GameObject panel;

    void Start(){
        panel.SetActive(true);
    }

    public void StartSal(){
        SceneManager.LoadScene("Par1S");
    }
}

```

## ΠΑΡΑΡΤΗΜΑ F : ΚΩΔΙΚΑΣ SCRIPT Int1T

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class Int1T : MonoBehaviour
{
    public GameObject Int1TCanvas, Texts;
    public float timer = 0f;

    void Start(){
        Int1TCanvas.SetActive(true);
        Texts.SetActive(true);
    }

    void Update(){
        timer += Time.deltaTime;

        if(timer >= 20f){
            onPar1T();
        }
    }

    public void onPar1T(){
        SceneManager.LoadScene("Par1T");
    }
}
```

## ΠΑΡΑΡΤΗΜΑ G : ΚΩΔΙΚΑΣ SCRIPT Int2T

```
using System.Collections;
using System.Collections.Generic;
```

```

using UnityEngine;
using UnityEngine.SceneManagement;

public class Int2T : MonoBehaviour
{
    public GameObject Int2TCanvas;
    public float timer = 0f;

    void Start(){
        Int2TCanvas.SetActive(true);
    }

    void Update(){
        timer += Time.deltaTime;

        if(timer >= 20f){
            onPar3T();
        }
    }

    public void onPar3T(){
        SceneManager.LoadScene("Par3T");
    }
}

```

## **ΠΑΡΑΡΤΗΜΑ Η : ΚΩΔΙΚΑΣ SCRIPT Par1T**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
using UnityEngine.UI;

public class Par1T : MonoBehaviour

```

```

{
    public GameObject Par1TCanvas, Panels;

    private float timer = 0f;

    void Start(){
        Par1TCanvas.SetActive(true);
        Panels.SetActive(true);
    }

    void Update(){
        timer += Time.deltaTime;
        if (timer >= 20f){
            onPar2T();
        }
    }

    public void onPar2T(){
        SceneManager.LoadScene("Par2T");
    }
}

```

## **ΠΑΡΑΡΤΗΜΑ Ι : ΚΩΔΙΚΑΣ SCRIPT Par2T**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class Par2T : MonoBehaviour
{
    public GameObject Par2TCanvas, Panels;

    private float timer = 0f;

```

```

void Start(){
    Par2TCanvas.SetActive(true);
    Panels.SetActive(true);
}

void Update(){
    timer += Time.deltaTime;

    if (timer >= 20f){
        onIntTwoT();
    }
}

public void onIntTwoT(){
    SceneManager.LoadScene("Int2T");
}
}

```

## **ΠΑΡΑΡΤΗΜΑ J : ΚΩΔΙΚΑΣ SCRIPT Par3T**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
using UnityEngine.UI;

public class Par3T : MonoBehaviour
{
    public GameObject Par3TCanvas, Panels;
    private float timer = 0f;

    void Start(){
        Par3TCanvas.SetActive(true);

```

```

    Panels.SetActive(true);
}

void Update(){
    timer += Time.deltaTime;
    if (timer >= 20f){
        onPar4T();
    }
}

public void onPar4T(){
    SceneManager.LoadScene("Par4T");
}
}

```

## **ΠΑΡΑΡΤΗΜΑ Κ : ΚΩΔΙΚΑΣ SCRIPT Par4T**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
using UnityEngine.UI;

public class Par4T : MonoBehaviour
{
    public GameObject Par4TCanvas, Panels;
    private float timer = 0f;

    void Start(){
        Par4TCanvas.SetActive(true);
        Panels.SetActive(true);
    }

    void Update(){

```

```

timer += Time.deltaTime;
if (timer >= 20f){
    onQuizThermopyles();
}
}

```

```

public void onQuizThermopyles(){
    SceneManager.LoadScene("QuizT");
}
}

```

## ΠΑΡΑΡΤΗΜΑ L : ΚΩΔΙΚΑΣ SCRIPT QuizT

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.SceneManagement;
using TMPro;
using System;

```

```

public class QuizT : MonoBehaviour{

```

```

    public GameObject startQT, questionsT, resultsT;

```

```

    public TextMeshProUGUI questionTextT, finalScoreTextT;

```

```

    public Button[] answerButtonsT;

```

```

    private int currentQuestionIndex, score, correctIndex;

```

```

    private string[] questions = {

```

```

        "Πως ονομάζεται αυτός που έδειξε τον δρόμο στους Πέρσες για να περάσουν από την απέναντι
        πλευρά του βουνού;",

```

```

"Ποιος ήταν ο αρχηγός των Ελλήνων;",
"Από που ήταν οι 700 στρατιώτες που έμειναν στο τέλος μαζί με τους 300 Σπαρτιάτες;",
"Πόσοι ήταν περίπου οι Πέρσες στρατιώτες;",
"Σε ποιο μέρος των Θερμοπυλών περίμεναν οι Έλληνες τον Περσικό στρατό;",
"Πόσοι ήταν περίπου οι Έλληνες στρατιώτες;",
"Ποιοι νίκησαν την μάχη;",
"Ποια φράση είπε ο Λεωνίδας όταν του είπε ο Ξέρξης να παραδοθούν;",
"Από που θα οδηγούσε ο Εφιάλτης τους Πέρσες στην απέναντι πλευρά;",
"Ποια πόλη της Ελλάδος πρόκειται να επιτεθούν οι Πέρσες;"
};

```

```

public TextMeshProUGUI[] buttonsTexts;
private string[,] answers = {
    { "Ξέρξης", "Εφιάλτης", "Λεωνίδας" },
    { "Λεωνίδας", "Ξέρξης", "Εφιάλτης" },
    { "Θεσπιές", "Αργος", "Κόρινθος" },
    { "320.000", "310.000", "300.000" },
    { "Στην πεδιάδα", "Στο βουνό", "Στο στενό" },
    { "7500", "7400", "7300" },
    { "Πέρσες", "Έλληνες", "Κανένας" },
    { "Νενικήκαμεν", "Μολών Λαβέ", "Θα παραδοθούμε" },
    { "Από το στενό", "Από το βουνό", "Από τη θάλασσα" },
    { "Θεσπιές", "Σπάρτη", "Αθήνα" }
};

```

```

private int[] correctAnswers = { 1, 0, 0, 2, 2, 1, 0, 1, 1, 2 };

```

```

public AudioSource correct, wrong, congrats;

```

```

void Start(){
    startQT.SetActive(true);
    questionsT.SetActive(false);
    resultsT.SetActive(false);
}

```

```
}
```

```
public void onQuestionsT(){  
    score = 0;  
    currentQuestionIndex = 0;  
    startQT.SetActive(false);  
    questionsT.SetActive(true);  
    nextQuestion();  
}
```

```
private void nextQuestion(){  
    if (currentQuestionIndex < questions.Length){  
        questionTextT.text = questions[currentQuestionIndex];  
  
        for(int i = 0; i < buttonsTexts.Length; i++){  
            buttonsTexts[i].text = answers[currentQuestionIndex,i];  
        }  
  
        correctIndex = correctAnswers[currentQuestionIndex];  
        currentQuestionIndex++;  
    }  
    else{  
        endQuiz();  
    }  
}
```

```
public void onAnswerOneT(){  
    if(correctIndex == 0){  
        answerButtonsT[0].image.color = Color.green;  
        score ++;  
        correct.Play();  
    }  
    else{
```

```

        answerButtonsT[0].image.color = Color.red;
        wrong.Play();
        answerButtonsT[correctIndex].image.color = Color.green;
    }
    buttonsEnDis();
    StartCoroutine(Delay3sT(0, correctIndex));
}

```

```

public void onAnswerTwoT(){
    if(correctIndex == 1){
        answerButtonsT[1].image.color = Color.green;
        score ++;
        correct.Play();
    }
    else{
        answerButtonsT[1].image.color = Color.red;
        wrong.Play();
        answerButtonsT[correctIndex].image.color = Color.green;
    }
    buttonsEnDis();
    StartCoroutine(Delay3sT(1, correctIndex));
}

```

```

public void onAnswerThreeT(){
    if(correctIndex == 2){
        answerButtonsT[2].image.color = Color.green;
        score ++;
        correct.Play();
    }
    else{
        answerButtonsT[2].image.color = Color.red;
        wrong.Play();
        answerButtonsT[correctIndex].image.color = Color.green;
    }
}

```

```

    }
    buttonsEnDis();
    StartCoroutine(Delay3sT(2, correctIndex));
}

private Boolean interact = true;
private void buttonsEnDis(){
    if(interact){
        for(int i = 0; i < answerButtonsT.Length; i++){
            answerButtonsT[i].interactable = false;
        }
    }
    else{
        for(int i = 0; i < answerButtonsT.Length; i++){
            answerButtonsT[i].interactable = true;
        }
    }
    interact = !interact;
}

private void endQuiz(){
    resultsT.SetActive(true);
    congrats.Play();
    finalScoreTextT.text = "Βαθμολογία: " + score + "/10";
}

public void onMenuT(){
    SceneManager.LoadScene("Menu");
}

public IEnumerator Delay3sT(int index, int correctIndex){
    yield return new WaitForSeconds(3);
}

```

```

if(index == correctIndex) answerButtonsT[index].image.color = Color.white;
else{
    answerButtonsT[index].image.color = Color.white;
    answerButtonsT[correctIndex].image.color = Color.white;
}
buttonsEnDis();
nextQuestion();
}
}

```

## **ΠΑΡΑΡΤΗΜΑ Μ : ΚΩΔΙΚΑΣ SCRIPT Int1S**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.AI;
using UnityEngine.SceneManagement;

public class Int1S : MonoBehaviour
{
    public GameObject Int1SCanvas, Texts;

    public float timer = 0f;

    void Start(){
        Int1SCanvas.SetActive(true);
        Texts.SetActive(true);
    }

    void Update(){
        timer += Time.deltaTime;

        if(timer >= 20f){
            onPar3S();
        }
    }
}

```

```

    }
}

public void onPar3S(){
    SceneManager.LoadScene("Par3S");
}
}

```

## **ΠΑΡΑΡΤΗΜΑ Ν : ΚΩΔΙΚΑΣ SCRIPT Int2S**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class Int2S : MonoBehaviour
{
    public GameObject Int2SCanvas;

    public float timer = 0f;

    void Start(){
        Int2SCanvas.SetActive(true);
    }

    void Update(){
        timer += Time.deltaTime;

        if(timer >= 20f){
            onPar5S();
        }
    }

    public void onPar5S(){

```

```

        SceneManager.LoadScene("Par5S");
    }
}

```

## ΠΑΡΑΡΤΗΜΑ Ο : ΚΩΔΙΚΑΣ SCRIPT Par1S

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
using UnityEngine.UI;

public class Par1S : MonoBehaviour
{
    public GameObject Par1SCanvas, Persian1SPanel;

    private float timer = 0f;

    void Start(){
        Persian1SPanel.SetActive(true);
        Par1SCanvas.SetActive(true);
    }

    void Update(){
        timer += Time.deltaTime;
        if (timer >= 20f){
            onPar2S();
        }
    }

    public void onPar2S(){
        SceneManager.LoadScene("Par2S");
    }
}

```

## ΠΑΡΑΡΤΗΜΑ Ρ : ΚΩΔΙΚΑΣ SCRIPT Par2S

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
using UnityEngine.UI;

public class Par2S : MonoBehaviour
{
    public GameObject Par2SCanvas;

    private float timer = 0f;

    void Start(){
        Par2SCanvas.SetActive(true);
    }

    void Update(){
        timer += Time.deltaTime;
        if (timer >= 20f){
            onInt1S();
        }
    }

    public void onInt1S(){
        SceneManager.LoadScene("Int1S");
    }
}
```

## ΠΑΡΑΡΤΗΜΑ Q : ΚΩΔΙΚΑΣ SCRIPT Par3S

```
using System.Collections;
using System.Collections.Generic;
```

```

using UnityEngine;
using UnityEngine.SceneManagement;
using UnityEngine.UI;

public class Par3S : MonoBehaviour
{
    public GameObject Par3SCanvas, GreeksPanel;

    private float timer = 0f;

    void Start(){
        Par3SCanvas.SetActive(true);
        GreeksPanel.SetActive(true);
    }

    void Update(){
        timer += Time.deltaTime;
        if (timer >= 20f){
            onPar4S();
        }
    }

    public void onPar4S(){
        SceneManager.LoadScene("Par4S");
    }
}

```

## **ΠΑΡΑΡΤΗΜΑ R : ΚΩΔΙΚΑΣ SCRIPT Par4S**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

```

```

public class Par4S : MonoBehaviour
{
    public GameObject Par4SCanvas, GreekPanel, PersianPanel;

    private float timer = 0f;

    void Start(){
        Par4SCanvas.SetActive(true);
        GreekPanel.SetActive(true);
        PersianPanel.SetActive(true);
    }

    void Update(){
        timer += Time.deltaTime;
        if (timer >= 20f){
            onInt2S();
        }
    }

    public void onInt2S(){
        SceneManager.LoadScene("Int2S");
    }
}.

```

## **ΠΑΡΑΡΤΗΜΑ Σ : ΚΩΔΙΚΑΣ SCRIPT Par5S**

```

using System;
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class Par5S : MonoBehaviour
{

```

```

public GameObject Par5SCanvas;
private float timer = 0f;

void Start(){
    Par5SCanvas.SetActive(true);
}

void Update(){
    timer += Time.deltaTime;
    if (timer >= 20f){
        onQuizSal();
    }
}

public void onQuizSal(){
    SceneManager.LoadScene("QuizS");
}
}.

```

## **ΠΑΡΑΡΤΗΜΑ Τ : ΚΩΔΙΚΑΣ SCRIPT QuizS**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.SceneManagement;
using TMPro;
using System;

public class QuizS : MonoBehaviour
{
    public GameObject startQS, questionsS, resultsS;

    public TextMeshProUGUI questionTextS, finalScoreTextS;

```

```

public Button[] answerButtonsS;

private int currentQuestionIndex, score, correctIndex;

private string[] questions = {
    "Πως ονομάζονται τα ελληνικά πλοία;",
    "Ποιος αρχηγός του στόλου ήταν από την Σπάρτη;",
    "Πόσα περίπου πλοία είχε ο Περσικός στόλος;",
    "Πόσα πλοία είχε ο Ελληνικός στόλος;",
    "Σε ποιανού του βουνού την πλαγιά είδε ο Ξέρξης την μάχη;",
    "Τι αντίκρισαν οι Πέρσες όταν έφτασαν στην Αθήνα;",
    "Που εγκαταστάθηκαν τα γυναικόπαιδα και οι ηλικιώμενοι;",
    "Ποιανού αρχηγού του στόλου έγινε δεκτή η πρόταση να τοποθετηθεί ο στόλος στο στενό της Σαλαμίνας;",
    "Ποιος νίκησε την ναυμαχία;",
    "Που πήγε ο Ξέρξης μετά την ήττα;"
};

public TextMeshProUGUI[] buttonsTexts;

private string[,] answers = {
    { "Τριήρης", "Τετρήρης", "Πεντήρης" },
    { "Θεμιστοκλής", "Ευρυβιάδης", "Κανένας" },
    { "1100", "1200", "1300" },
    { "366-379", "366-387", "366-378" },
    { "Πάρνηθας", "Αιγάλεω", "Υμηττού" },
    { "Ήταν έρημη", "Ελληνικό στρατό", "Δεν έφτασαν" },
    { "Στο Άργος", "Στην Κόρινθο", "Στα γύρω νησιά"},
    { "Ευρυβιάδη", "Θεμιστοκλή", "Κανερός"},
    { "Πέρσες", "Ελληνες", "Κανένας"},
    { "Στις Θερμοπύλες", "Στην Κόρινθο", "Στην Περσία"}
};

private int[] correctAnswers = { 0, 1, 1, 2, 1, 0, 2, 1, 1, 2 };

```

```

public AudioSource correct, wrong, congrats;

void Start(){
    startQS.SetActive(true);
    questionsS.SetActive(false);
    resultsS.SetActive(false);
}

public void onQuestionsS(){
    score = 0;
    currentQuestionIndex = 0;
    startQS.SetActive(false);
    questionsS.SetActive(true);
    nextQuestion();
}

private void nextQuestion(){
    if (currentQuestionIndex < questions.Length){
        questionTextS.text = questions[currentQuestionIndex];

        for(int i = 0; i < buttonsTexts.Length; i++){
            buttonsTexts[i].text = answers[currentQuestionIndex,i];
        }

        correctIndex = correctAnswers[currentQuestionIndex];
        currentQuestionIndex++;
    }
    else{
        endQuiz();
    }
}
}

```

```

public void onAnswerOneS(){
    if(correctIndex == 0){
        answerButtonsS[0].image.color = Color.green;
        score ++;
        correct.Play();
    }
    else{
        answerButtonsS[0].image.color = Color.red;
        wrong.Play();
        answerButtonsS[correctIndex].image.color = Color.green;
    }
    buttonsEnDis();
    StartCoroutine(Delay3sS(0, correctIndex));
}

```

```

public void onAnswerTwoS(){
    if(correctIndex == 1){
        answerButtonsS[1].image.color = Color.green;
        score ++;
        correct.Play();
    }
    else{
        answerButtonsS[1].image.color = Color.red;
        wrong.Play();

        answerButtonsS[correctIndex].image.color = Color.green;
    }
    buttonsEnDis();
    StartCoroutine(Delay3sS(1, correctIndex));
}

```

```

public void onAnswerThreeS(){
    if(correctIndex == 2){

```

```

        answerButtonsS[2].image.color = Color.green;
        score ++;
        correct.Play();
    }
    else{
        answerButtonsS[2].image.color = Color.red;
        wrong.Play();
        answerButtonsS[correctIndex].image.color = Color.green;
    }
    buttonsEnDis();
    StartCoroutine(Delay3sS(2, correctIndex));
}

```

```

private Boolean interact = true;
private void buttonsEnDis(){
    if(interact){
        for(int i = 0; i < answerButtonsS.Length; i++){
            answerButtonsS[i].interactable = false;
        }
    }
    else{
        for(int i = 0; i < answerButtonsS.Length; i++){
            answerButtonsS[i].interactable = true;
        }
    }
    interact = !interact;
}

```

```

private void endQuiz(){
    resultsS.SetActive(true);
    congrats.Play();
    finalScoreTextS.text = "Βαθμολογία: " + score + "/10";
}

```

```
}

public void onMenuS(){
    SceneManager.LoadScene("Menu");
}

public IEnumerator Delay3sS(int index, int correctIndex){
    yield return new WaitForSeconds(3);
    if(index == correctIndex) answerButtonsS[index].image.color = Color.white;
    else{
        answerButtonsS[index].image.color = Color.white;
        answerButtonsS[correctIndex].image.color = Color.white;
    }
    buttonsEnDis();
    nextQuestion();
}
}
```

## ΠΑΡΑΡΤΗΜΑ U : ΚΩΔΙΚΑΣ SCRIPT Move1T

```
using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Move1T : MonoBehaviour

{

    public GameObject persian1T;

    public GameObject persian1T1;

    public GameObject persian1T2;

    public GameObject persian1T3;

    public GameObject persian1T4;

    public GameObject persian1T5;

    public GameObject xerxes1T;

    public GameObject target1T;

    public GameObject target1T1;

    public GameObject target1T2;

    public GameObject target1TXerxes;

    private float speed = 0.005f;
```

```

void Update(){

    var step = speed * Time.deltaTime;

    persian1T.transform.position = Vector3.MoveTowards(persian1T.transform.position,
target1T.transform.position, step);

    persian1T1.transform.position = Vector3.MoveTowards(persian1T1.transform.position,
target1T.transform.position, step);

    persian1T2.transform.position = Vector3.MoveTowards(persian1T2.transform.position,
target1T1.transform.position, step);

    persian1T3.transform.position = Vector3.MoveTowards(persian1T3.transform.position,
target1T1.transform.position, step);

    persian1T4.transform.position = Vector3.MoveTowards(persian1T4.transform.position,
target1T2.transform.position, step);

    persian1T5.transform.position = Vector3.MoveTowards(persian1T5.transform.position,
target1T2.transform.position, step);

    xerxes1T.transform.position = Vector3.MoveTowards(xerxes1T.transform.position,
target1TXerxes.transform.position, step);

}

}

```

## **ΠΑΡΑΡΤΗΜΑ V : ΚΩΔΙΚΑΣ SCRIPT Move3T**

```

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Move3T : MonoBehaviour

```

```

{

public GameObject ephialtes3T;

public GameObject target3T;

private float speed = 0.005f;

private bool isDelayed = false;

void Start(){

    StartCoroutine(delay3T());

}

void Update(){

    if (isDelayed){

        var step = speed * Time.deltaTime;

        ephialtes3T.transform.position = Vector3.MoveTowards(ephialtes3T.transform.position,
target3T.transform.position, step);

    }

}

// Delay until pointing animation is played

public IEnumerator delay3T(){

    yield return new WaitForSeconds(6);

}

```

```
        isDelayed = true;
    }
}
```

## **ΠΑΡΑΡΤΗΜΑ W : ΚΩΔΙΚΑΣ SCRIPT Move4T**

```
using System.Collections;
```

```
using System.Collections.Generic;
```

```
using UnityEngine;
```

```
using UnityEngine.UI;
```

```
public class Move4T : MonoBehaviour{
```

```
    public GameObject greek4T;
```

```
    public GameObject greek4T2;
```

```
    public GameObject target4T;
```

```
    public GameObject target4T2;
```

```
    private float speed = 0.01f;
```

```
    void Update(){
```

```
        var step = speed * Time.deltaTime;
```

```
        greek4T.transform.position = Vector3.MoveTowards(greek4T.transform.position,  
target4T.transform.position, step);
```

```
        greek4T2.transform.position = Vector3.MoveTowards(greek4T2.transform.position,  
target4T2.transform.position, step);
```

```
}  
  
}
```

## **ΠΑΡΑΡΤΗΜΑ X : ΚΩΔΙΚΑΣ SCRIPT Move1S**

```
using System.Collections;  
  
using System.Collections.Generic;  
  
using UnityEngine;  
  
public class MoveS1 : MonoBehaviour  
{  
  
    public GameObject persian1S;  
  
    public GameObject persian1S1;  
  
    public GameObject persian1S2;  
  
    public GameObject persian1S3;  
  
  
    public GameObject xerxes1S;  
  
    public GameObject target1S;  
  
    public GameObject target1S1;  
  
    public GameObject target1S2;  
  
    public GameObject target1S3;  
  
    public GameObject target1SXerxes;  
  
    private float speed = 0.005f;
```

```

void Update(){

    var step = speed * Time.deltaTime;

    persian1S.transform.position = Vector3.MoveTowards(persian1S.transform.position,
target1S.transform.position, step);

    persian1S1.transform.position = Vector3.MoveTowards(persian1S1.transform.position,
target1S1.transform.position, step);

    persian1S2.transform.position = Vector3.MoveTowards(persian1S2.transform.position,
target1S2.transform.position, step);

    persian1S3.transform.position = Vector3.MoveTowards(persian1S3.transform.position,
target1S3.transform.position, step);

    xerxes1S.transform.position = Vector3.MoveTowards(xerxes1S.transform.position,
target1SXerxes.transform.position, step);

}

}

```

## **ΠΑΡΑΡΤΗΜΑ Υ : ΚΩΔΙΚΑΣ SCRIPT Move3S**

```

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Move3S : MonoBehaviour

{

    public GameObject trimere3S;

    public GameObject trimere3S1;

```

```

public GameObject trimere3S2;

public GameObject target3S;

public GameObject target3S1;

public GameObject target3S2;

private float speed = 0.005f;

void Update(){

    var step = speed * Time.deltaTime;

    trimere3S.transform.position = Vector3.MoveTowards(trimere3S.transform.position,
target3S.transform.position, step);

    trimere3S1.transform.position = Vector3.MoveTowards(trimere3S1.transform.position,
target3S1.transform.position, step);

    trimere3S2.transform.position = Vector3.MoveTowards(trimere3S2.transform.position,
target3S2.transform.position, step);

}

}

```

## **ΠΑΡΑΡΤΗΜΑ Ζ : ΚΩΔΙΚΑΣ SCRIPT Move4S**

```

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Move4S : MonoBehaviour

```

```

{

    public GameObject persian4S;

    public GameObject persian4S1;

    public GameObject persian4S2;

    public GameObject xerxes4S;

    public GameObject target4S;

    public GameObject target4S1;

    public GameObject target4S2;

    public GameObject target4SXerxes;

    private float speed = 0.005f;

    void Update(){

        var step = speed * Time.deltaTime;

        persian4S.transform.position = Vector3.MoveTowards(persian4S.transform.position,
target4S.transform.position, step);

        persian4S1.transform.position = Vector3.MoveTowards(persian4S1.transform.position,
target4S1.transform.position, step);

        persian4S2.transform.position = Vector3.MoveTowards(persian4S2.transform.position,
target4S2.transform.position, step);

        xerxes4S.transform.position = Vector3.MoveTowards(xerxes4S.transform.position,
target4SXerxes.transform.position, step);

    }

}

```

## ΠΑΡΑΡΤΗΜΑ ΑΑ : ΚΩΔΙΚΑΣ SCRIPT Move 5S

```
using System.Collections;
```

```
using System.Collections.Generic;
```

```
using UnityEngine;
```

```
public class Move5S : MonoBehaviour
```

```
{
```

```
    public GameObject persian5S;
```

```
    public GameObject persian5S1;
```

```
    public GameObject target5S;
```

```
    public GameObject target5S1;
```

```
    private float speed = 0.005f;
```

```
    void Update(){
```

```
        var step = speed * Time.deltaTime;
```

```
        persian5S.transform.position = Vector3.MoveTowards(persian5S.transform.position,  
target5S.transform.position, step);
```

```
        persian5S1.transform.position = Vector3.MoveTowards(persian5S1.transform.position,  
target5S1.transform.position, step);
```

```
    }
```

```
}
```

## ΠΑΡΑΡΤΗΜΑ ΒΒ : ΚΩΔΙΚΑΣ SCRIPT Object Drag

```

using UnityEngine;
using Lean.Touch;

public class ObjectDrag : MonoBehaviour
{
    private Vector3 offset;
    private bool dragging = false;

    void OnEnable(){
        LeanTouch.OnFingerDown += OnFingerDown;
        LeanTouch.OnFingerUp += OnFingerUp;
        LeanTouch.OnFingerUpdate += OnFingerUpdate;
    }

    void OnDisable(){
        LeanTouch.OnFingerDown -= OnFingerDown;
        LeanTouch.OnFingerUp -= OnFingerUp;
        LeanTouch.OnFingerUpdate -= OnFingerUpdate;
    }

    private void OnFingerDown(LeanFinger finger){
        Ray ray = finger.GetRay();
        RaycastHit hit;

        if (Physics.Raycast(ray, out hit)){
            if (hit.transform == transform){
                offset = transform.position - hit.point;
                dragging = true;
            }
        }
    }

    private void OnFingerUpdate(LeanFinger finger){
        if (dragging){
            Ray ray = finger.GetRay();
            float distance = Vector3.Distance(transform.position, Camera.main.transform.position);
            Vector3 targetPosition = ray.GetPoint(distance) + offset;
            transform.position = new Vector3(targetPosition.x, transform.position.y, targetPosition.z);
        }
    }

    private void OnFingerUp(LeanFinger finger){
        dragging = false;
    }
}

```

