



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ**  
**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ**  
**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

**Σχεδίαση και ανάπτυξη διαδραστικού  
εκπαιδευτικού παιχνιδιού με χρήση της  
πλατφόρμας Unity 3D, με θέμα την  
αρχαιοελληνική Μυθολογία και Προϊστορία, για  
μαθητές Πρωτοβάθμιας Εκπαίδευσης**

**ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

της

**ΦΙΤΖΙΟΥ ΑΓΝΗΣ**

( ΑΕΜ: 213 )

**Επιβλέπων:**

**Νικόλαος Δημόκας**

**Επίκουρος Καθηγητής**

**Συνεπιβλέπων:**

**Σπυρίδων Νικολάου**

**Λέκτορας**

Καστοριά Μάρτιος 2023





ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ  
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Σχεδίαση και ανάπτυξη διαδραστικού  
εκπαιδευτικού παιχνιδιού με χρήση της  
πλατφόρμας Unity 3D, με θέμα την  
αρχαιοελληνική Μυθολογία και Προϊστορία, για  
μαθητές Πρωτοβάθμιας Εκπαίδευσης**

**ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

της

**ΦΙΤΖΙΟΥ ΑΓΝΗΣ**

( ΑΕΜ: 213 )

**Επιβλέπων:**

**Νικόλαος Δημόκας**

**Επίκουρος Καθηγητής**

**Συνεπιβλέπων:**

**Σπυρίδων Νικολάου**

**Λέκτορας**

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 4<sup>η</sup> Μαρτίου 2023

Νικόλαος Δημόκας

Ζαχαρούλα Καλογηράτου

Σπυρίδων Νικολάου

Επίκουρος Καθηγητής

Καθηγήτρια

Λέκτορας

Καστοριά Μάρτιος 2023

Copyright © 2023 – ΦΙΤΖΙΟΥ ΑΓΝΗ

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν αποκλειστικά τον συγγραφέα και δεν αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Δυτικής Μακεδονίας.



## Ευχαριστίες

Από τη θέση αυτή θα ήθελα να απευθύνω τις ευχαριστίες μου προς τους ομότιμους καθηγητές του Τμήματος Μαθηματικών της Σχολής Θετικών Επιστημών του Πανεπιστημίου Ιωαννίνων, κ. Σωτήρη Λουκά και κ. Νικόλαο Μαρμαρίδη, τον μεν πρώτο επειδή μου έμαθε να μην αρκούμαι στα ημίμετρα, αλλά να απαιτώ αυτό που αξίζω και τον δεύτερο επειδή πίστεψε σε μένα και μου έδωσε ώθηση να συνεχίζω. Πολλά «ευχαριστώ» οφείλω στην αδερφή μου, συνταξιούχο άριστη εκπαιδευτικό που με βοήθησε στη σύνταξη των ερωτήσεων του παιχνιδιού μου και στον καλό συνάδελφο και φίλο Δαμιανό Κράπη, που με βοήθησε στα πρακτικά ζητήματα της διαδικτυακής σύνδεσης και παρακολούθησης της σελίδας μου. Επίσης, ευχαριστώ θερμά τον επιβλέποντα κ. Νικόλαο Δημόκα και το συνεπιβλέποντά μου κ. Σπυρίδωνα Νικολάου για τη βοήθεια και τη συμπαράστασή τους, την άμεση ανταπόκρισή τους σε κάθε αίτημα και e-mail μου και για την καθοδήγησή τους από την αρχή της συνεργασίας μας. Εξαιρετική μνεία οφείλω στη φίλη Χριστίνα Χρυσόχου που με έπεισε ότι ποτέ δεν είναι αργά... Τελευταίους αλλά πρώτιστης σημασίας, θέλω να ευχαριστήσω τους γονείς μου που με στήριξαν σε όλη τη ζωή μου και τα παιδιά μου για την υπομονή τους, να με βλέπουν να διαβάζω ατελείωτες ώρες στην ηλικία μου!

*Αν δεν μπορείς να το εξηγήσεις με απλά λόγια,  
ούτε εσύ ο ίδιος το έχεις καταλάβει πολύ καλά!*

Albert Einstein

## Περίληψη

Βασιζόμενοι στην εξέλιξη των θεωριών της μάθησης, από το συμπεριφορισμό μέχρι την παιγνιώδη μάθηση και την εξέλιξη στον προγραμματισμό, από τον σειριακό στον αντικειμενοστραφή, οδηγούμαστε στην ανάγκη δημιουργίας ενός εκπαιδευτικού παιχνιδιού, συνδυασμό των παραπάνω, προς όφελος των μαθητών. Η παρούσα μεταπτυχιακή εργασία, πραγματεύεται όλη τη διαδικασία, σχεδιασμού και ανάπτυξης ενός εκπαιδευτικού λογισμικού. Αναλύεται το σενάριο, παρουσιάζονται οι τελικοί χρήστες και οι εκπαιδευτικοί στόχοι. Αποσαφηνίζονται όλες οι επιλογές προγραμματιστικά, σχεδιαστικά και παιδαγωγικά. Καταγράφονται οι τεχνικές προδιαγραφές πολυμέσων, αλλά και οι εκπαιδευτικές προδιαγραφές, με στοιχεία διδακτικής σχεδίασης. Η εκπαιδευτική εφαρμογή περιγράφεται αναλυτικά, σαν παιχνίδι και σαν πρόγραμμα με τη βοήθεια UML διαγραμμάτων. Η ανάπτυξη της έγινε στην παιχνιδομηχανή Unity 3D, στην οποία γίνεται αναφορά καθώς και συγκριτική μελέτη με την επίσης διάσημη πλατφόρμα παραγωγής παιχνιδιών Unreal Engine.

### **Λέξεις Κλειδιά:**

παιγνιώδης μάθηση

αντικειμενοστραφής προγραμματισμός

εκπαιδευτικό λογισμικό

παιχνιδομηχανή Unity 3D

διαγράμματα UML

## **Abstract**

Based on the evolution of learning theories, from behaviorism to playful learning and evolution in programming, from serial to object-oriented, we are led to the need to create an educational game, a combination of the above, for the benefit of students. This postgraduate thesis deals with the entire process of designing and developing an educational software. The scenario is analyzed, the end users and educational objectives are presented. All programming, design and pedagogical options are clarified. Multimedia specifications are recorded, as well as educational requirements, with elements of instructional design. The educational application is described in detail, as a game and as a program with the help of UML diagrams. It was developed in the Unity 3D game engine, which is referenced as well as a comparative study with the also famous Unreal Engine game production platform.

### **Key Words:**

playful learning

object oriented programming

educational software

Unity 3D game engine

UML diagrams

## Πίνακας Περιεχομένων

Εισαγωγή.....	1
<b>1. Παιγνιώδης Μάθηση.....</b>	<b>3</b>
1.1 Παιδαγωγικές Θεωρίες Μάθησης .....	3
1.2 Τι είναι το Παιχνίδι; .....	6
1.3 Παιδαγωγικές Θεωρίες Παιχνιδιού & Είδη.....	7
1.4 Παιχνιδοποίηση (Gamification) .....	9
1.4.1 Τι είναι τελικά η Παιχνιδοποίηση .....	9
1.4.2 Στοιχεία Παιχνιδιών Απαραίτητα στην Παιχνιδοποίηση .....	10
1.4.3 Η Παιχνιδοποίηση στην Εκπαίδευση .....	11
1.4.4 Επιγραμματικά και Διαγραμματικά.....	14
1.4.5 Παιχνιδοποίηση και Εκπαιδευτικά Παιχνίδια .....	15
<b>2. Αντικειμενοστραφής Προγραμματισμός.....</b>	<b>17</b>
2.1 Ιστορική Αναδρομή Γλωσσών Προγραμματισμού .....	17
2.2 Τεχνικές Σχεδίασης Προγραμμάτων .....	19
2.3 Αντικειμενοστραφής Προγραμματισμός – Τα Βασικά.....	20
2.3.1 Αντικείμενα .....	20
2.3.2 Κλάσεις.....	20
2.3.3 Χαρακτηριστικά Αντικειμενοστραφών Εφαρμογών .....	21
2.4 Αντικειμενοστραφής Ανάλυση και Σχεδιασμός .....	22
2.5 Σύγχρονα Προγραμματιστικά Περιβάλλοντα .....	23
2.6 Πλεονεκτήματα του αντικειμενοστραφούς προγραμματισμού.....	24
<b>3. Ησίοδος: το εκπαιδευτικό παιχνίδι.....</b>	<b>25</b>
3.1 Ανάγκη Ύπαρξης .....	26
3.1.1 Σενάριο .....	26
3.1.2 Τελικοί Χρήστες .....	27
3.1.3 Χρονοπρογραμματισμός και Δυσκολίες.....	28

3.1.4 Εκπαιδευτικοί Στόχοι.....	31
σε επίπεδο γνώσεων: .....	31
σε επίπεδο δεξιοτήτων: .....	31
σε επίπεδο στάσεων: .....	31
3.2 Περιγραφή Εφαρμογής.....	31
3.2.1 Προγραμματιστικές Επιλογές.....	32
Τι είναι μια παιχνιδομηχανή .....	32
Unity VS Unreal Engine .....	33
Γιατί Unity; .....	34
Η κοινότητα της Unity .....	35
3.2.2 Τεχνικές Προδιαγραφές Μέσων .....	36
3.2.3 Σχεδιαστικές Επιλογές.....	38
Οι Οκτώ Χρυσοί Κανόνες Σχεδιασμού .....	39
① Συνέπεια – Διαφάνεια.....	39
② Καθολική Χρηστικότητα.....	40
③ Ενημερωτική Ανατροφοδότηση – Απόδοση – Προσανατολισμός .....	40
④ Κλείσιμο Ενοτήτων .....	41
⑤ Αποτροπή Σφαλμάτων .....	41
⑥ Εύκολη Αντιστροφή Ενεργειών .....	42
⑦ Έλεγχος Χρηστών .....	43
⑧ Μείωση Φορτίου της Βραχυπρόθεσμης Μνήμης.....	43
Οι ανατροφοδοτήσεις στα Εκπαιδευτικά Λογισμικά.....	44
Δομή του Λογισμικού και Εκπαιδευτικές Προδιαγραφές .....	44
Επαναληπτικά Μοντέλα Σχεδίασης.....	46
Πρότυπα Σελίδων Εφαρμογής .....	47
Βασικές και Καλά Σχεδιασμένες Οθόνες .....	47
Στρατηγική Διδακτικής Σχεδίασης του Gagne .....	48

Σχεδίαση Ερωτημάτων .....	51
3.2.4 Παιδαγωγικές Επιλογές .....	51
3.2.5 Περιγραφή του Παιχνιδιού .....	54
1 <sup>η</sup> Οθόνη: Έναρξη .....	54
2 <sup>η</sup> Οθόνη: Γνωριμία με τους Φίλους.....	56
3 <sup>η</sup> Οθόνη: Εισαγωγή Στοιχείων Χρήστη.....	56
4 <sup>η</sup> Οθόνη: Επιλογή Ήρωα .....	57
5 <sup>η</sup> Οθόνη: Κεντρικό Μενού - Επιλογή Ενότητας .....	58
6 <sup>η</sup> Οθόνη: Μενού Επιλογής Επιπέδου .....	59
7 <sup>η</sup> Οθόνη: Κεντρική Οθόνη Παιχνιδιού .....	60
8 <sup>η</sup> Οθόνη: Βαθμολογία.....	62
Οθόνες Βοήθειας.....	62
3.3 UML: Μια εικόνα, χίλιες λέξεις.....	64
3.3.1 UML: Τι είναι .....	64
3.3.2 Σημειογραφίες UML.....	65
3.3.3 Το Διάγραμμα UML .....	66
3.3.4 Διαγράμματα Συμπεριφοράς – Behavioral Diagrams .....	68
Use Case Diagram.....	68
Activity Diagram .....	69
Interaction Diagram .....	74
3.3.5 Δομικά Διαγράμματα – Structural Diagrams .....	75
Deployment Diagram.....	75
Component Diagram .....	76
Class Diagram .....	79
Object Diagram.....	80
UML Διαγράμματα: Άμεση Σχέση – Διαφορετικός Ρόλος .....	81
3.4 Αναλυτική Παρουσίαση των Κλάσεων.....	82

3.4.1 Οι υποκλάσεις της MonoBehaviour .....	82
3.4.2 Η ιεραρχία και οι υποκλάσεις της PlayerPrefs .....	87
3.4.3 Κλάσεις και Αντικείμενα στη Unity .....	93
3.4.4 Οι σκηνές της Unity .....	94
<b>4. Διανομή του Παιχνιδιού και Αξιολόγηση.....</b>	<b>96</b>
4.1 Δοκιμές Alpha & Beta.....	98
4.1.1 Ησίοδος: Alpha Testing.....	98
4.1.2 Ησίοδος: Beta Testing .....	105
<b>Συμπεράσματα.....</b>	<b>108</b>
Βιβλιογραφία .....	111
Παράρτημα I - Κώδικας.....	116
Παράρτημα II – Ανάλυση 5βαθμης Κλίμακας .....	152

## Λίστα Εικόνων

---

Εικόνα 1: Εννοιολογικός Χάρτης Ανάπτυξης της Εργασίας.....	1
Εικόνα 2: Πυραμίδα της Μάθησης.....	12
Εικόνα 3: Επίδρασης της Παιχνιδοποίησης στην Εκπαίδευση.....	15
Εικόνα 4: ΗΣΙΟΔΟΣ - Το εικονίδιο της Εφαρμογής.....	25
Εικόνα 5: Το περιβάλλον της Unity.....	35
Εικόνα 6: Οργάνωση Σχεδίασης Εκπαιδευτικού Λογισμικού.....	38
Εικόνα 7: ΗΣΙΟΔΟΣ - Εναρκτήριες Οθόνες.....	55
Εικόνα 8: ΗΣΙΟΔΟΣ - Οθόνη Γνωριμίας με τους Φίλους.....	56
Εικόνα 9: ΗΣΙΟΔΟΣ - Οθόνη Εισαγωγής Στοιχείων.....	57
Εικόνα 10: ΗΣΙΟΔΟΣ - Οθόνη Επιλογής Παίκτη.....	58
Εικόνα 11: ΗΣΙΟΔΟΣ - Οθόνη Κεντρικού Μενού.....	58
Εικόνα 12: ΗΣΙΟΔΟΣ - Οθόνη Επιλογής Επιπέδου (Μυθολογία).....	59
Εικόνα 13: ΗΣΙΟΔΟΣ - Οθόνη Επιλογής Επιπέδου (Προϊστορία).....	60
Εικόνα 14: ΗΣΙΟΔΟΣ - Κυρίως Οθόνη Παιχνιδιού.....	61
Εικόνα 15: ΗΣΙΟΔΟΣ - Οθόνη Τερματισμού.....	62
Εικόνα 16: ΗΣΙΟΔΟΣ - Βοηθητικές Οθόνες.....	63
Εικόνα 17: Σημειογραφία UML για την κλάση αντικειμένων.....	65
Εικόνα 18: Διάγραμμα UML ( Esiodus ).....	68
Εικόνα 19: UML Διάγραμμα Περιπτώσεων Χρήσης.....	69
Εικόνα 20: UML Διάγραμμα Δραστηριότητας.....	70
Εικόνα 21: UML Δραστηριότητας (Μέρος Α΄).....	71
Εικόνα 22: UML Δραστηριότητας (Μέρος Β΄).....	72
Εικόνα 23: UML Διάγραμμα Αλληλεπίδρασης.....	73
Εικόνα 24: UML Διάγραμμα Ανάπτυξης.....	75



Εικόνα 25: Συνιστώσες της Unity.....	77
Εικόνα 26: UML Διάγραμμα Συνιστωσών.....	77
Εικόνα 27: UML Διάγραμμα Συνιστωσών - Επαυξημένο.....	78
Εικόνα 28: UML Ιεραρχικό Διάγραμμα.....	79
Εικόνα 29: Παράδειγμα Διαγράμματος Αντικειμένων.....	80
Εικόνα 30: Το αντικείμενο player σε κάποια στιγμή.....	81
Εικόνα 31: Αντικείμενα και Scripts.....	93
Εικόνα 32: Footer prefab.....	94
Εικόνα 33: Project Esiodus.....	95
Εικόνα 34: Η Σελίδα του Ησίοδου.....	96
Εικόνα 35: Φόρμα Αξιολόγησης Ησίοδου.....	97
Εικόνα 36: Μέσες Τιμές ανά Ερώτηση.....	100
Εικόνα 37: Αποδεκτές Ζώνες Μέσων Τιμών.....	101
Εικόνα 38: Σύγκριση με Άλλες Εφαρμογές.....	104
Εικόνα 39: Μέσες Τιμές και Διαστήματα Εμπιστοσύνης.....	104

## Λίστα Πινάκων

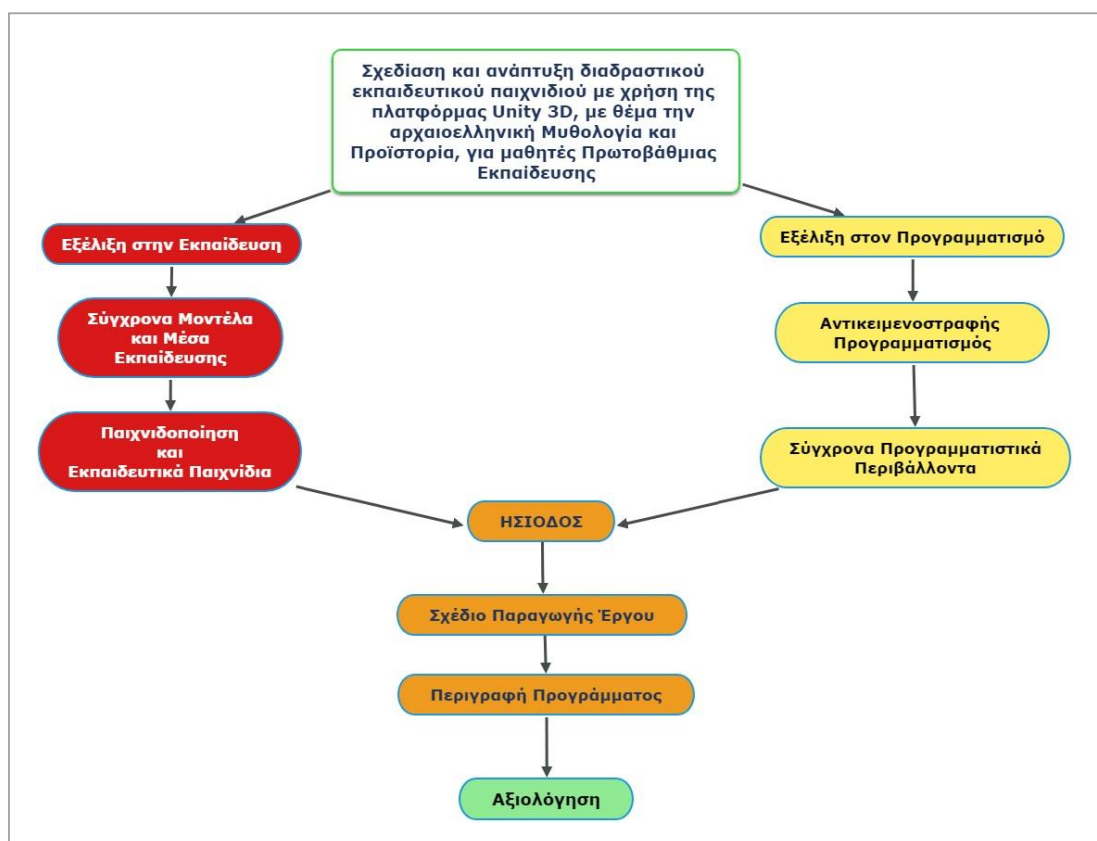
---

Πίνακας 1: Χαρακτηριστικά Παιχνιδιών και Εκπαίδευσης.....	14
Πίνακας 2: Διαφορές Εκπαιδευτικών Παιχνιδιών και Παιχνιδοποίησης .....	16
Πίνακας 3: Χρονοπρογραμματισμός - Διάγραμμα Gantt .....	29
Πίνακας 4: Unity VS Unreal Engine - Συγκριτική Αναφορά .....	33
Πίνακας 5: Τεχνικές Προδιαγραφές Μέσων.....	36
Πίνακας 6: Στατιστικά Αξιολόγησης Ανά Ερώτηση .....	100
Πίνακας 7: Συνολικοί Μέσοι Όροι Αξιολόγησης.....	101
Πίνακας 8: Διαστήματα Εμπιστοσύνης 95% (αναλυτικά).....	102
Πίνακας 9: Διαστήματα Εμπιστοσύνης 95% (ανά ομάδα) .....	102
Πίνακας 10: Δείκτες Alpha (Cronbach) .....	103
Πίνακας 11: Σύγκριση με άλλες Αξιολογημένες Εφαρμογές.....	103

## Εισαγωγή

Από τα αρχαία χρόνια οι άνθρωποι που ενδιαφέρονταν για την παιδεία, αναρωτήθηκαν για το «πώς μαθαίνει ο άνθρωπος». Η αναζήτηση αυτή, οδήγησε σε θεωρίες μάθησης και εκπαιδευτικά μοντέλα που εξελίσσονται και αναπτύσσονται στο πέρασμα του χρόνου, ενώ η παρούσα εργασία είναι μέρος της αναζήτησης αυτής.

Στο πρώτο κεφάλαιο γίνεται βιβλιογραφική ανασκόπηση στις θεωρίες της μάθησης και στις θεωρίες του παιχνιδιού. Δίνονται ποικίλοι ορισμοί του παιχνιδιού και εισάγεται η έννοια της παιχνιδοποίησης στην εκπαίδευση. Παρουσιάζονται η παιχνιδοποίηση και τα εκπαιδευτικά παιχνίδια ως εξέλιξη στη μάθηση. Στο δεύτερο κεφάλαιο με ανάλογο τρόπο σχολιάζεται η εξέλιξη των γλωσσών προγραμματισμού και των τεχνικών σχεδίασης προγραμμάτων, μέχρι τον αντικειμενοστραφή προγραμματισμό. Αποσαφηνίζονται τα βασικά στοιχεία αυτού και αναλύονται σύγχρονα προγραμματιστικά περιβάλλοντα και τα πλεονεκτήματά τους.



Εικόνα 1: Εννοιολογικός Χάρτης Ανάπτυξης της Εργασίας

Όπως φαίνεται και στον εννοιολογικό χάρτη της Εικόνας 1, τις δύο αυτές παράλληλες και ξένες μεταξύ τους κατευθύνσεις, έρχεται να ενώσει στο τρίτο

κεφάλαιο, ο Ησίοδος. Ο Ησίοδος είναι ένα εκπαιδευτικό, διαδραστικό παιχνίδι, που αναπτύχθηκε με χρήση της πλατφόρμας Unity 3D. Πραγματεύεται την αρχαιοελληνική Μυθολογία και Προϊστορία και φτιάχτηκε κυρίως για μαθητές Πρωτοβάθμιας Εκπαίδευσης και συγκεκριμένα της Γ' τάξης του Δημοτικού σχολείου. Στην περιγραφή του καταγράφεται όλη η διαδικασία ανάπτυξης ενός εκπαιδευτικού λογισμικού, με όλα τα διλήμματα και την αιτιολόγηση των αντίστοιχων επιλογών. Η εφαρμογή επεξηγείται ως παιχνίδι και ως πρόγραμμα, με πλήθος φωτογραφιών και σύγχρονες διαγραμματικές τεχνικές, ενώ στο τελευταίο κεφάλαιο παρατίθεται η αξιολόγησή του με alpha και beta users.

## 1. Παιγνιώδης Μάθηση

---

«Ο άνθρωπος ζών πολιτικών» είπε ο Αριστοτέλης στα «Πολιτικά», στο Βιβλίο Α', με συνέπεια η πολιτική, η φιλοσοφία, η κοινωνιολογία, η ψυχολογία, η λογοτεχνία και όλος ο πνευματικός και επιστημονικός χώρος να απασχολείται με την Παιδεία και την Αγωγή. Η αγωγή βρίσκεται σε άμεση συνάφεια με τις κοινωνικοπολιτικές εξελίξεις και ιδεολογίες. Γεγονότα που συμβαίνουν από εποχή σε εποχή, από χώρα σε χώρα, από τόπο σε τόπο της ίδιας χώρας, ακόμα και μέσα στην ίδια κοινωνία, δημιουργούν ένα γενικευμένο πλαίσιο επαναπροσδιορισμού ιδεών, αξιών και αρχών. Στο πλαίσιο αυτό, εκπαιδευτικοί και μαθητές οφείλουν να διαμορφώνουν τους ρόλους τους ανάλογα [1]. Οι στάσεις, οι δεξιότητες, οι στρατηγικές και οι γνώσεις που αποκτούν οι μαθητές θα πρέπει να συμβάλλουν στην επίτευξη του αυτοπροσδιορισμού, της αυτενέργειάς και της ανάπτυξης της κριτικής συνείδησής τους [2]. Διά νόμου έχει αποφασιστεί πως η παιδεία δεν πρέπει να αρκείται μόνο στη μετάδοση τυπικών γνώσεων στους μαθητές, αλλά να φροντίζει και για την ανάπτυξη των νοητικών και ψυχοσωματικών δεξιοτήτων τους: να καλλιεργεί ολοκληρωμένες και υπεύθυνες προσωπικότητες ανεξαρτήτως φύλου και καταγωγής που θα προασπίζονται το κοινό καλό και θα επιδιώκουν τη φιλία και τη συνεργασία με συμπολίτες τους και με άλλους λαούς (Νόμος 1566/85, κεφ. Α', άρθρο 1). Σύμφωνα με την Unesco [3] η εκπαίδευση για την παγκόσμια πολιτότητα, έχει ως στόχο να είναι μετασχηματιστική, να οικοδομήσει γνώσεις, δεξιότητες, αξίες και στάσεις ώστε να είναι σε θέση οι μαθητές να συμβάλουν σε έναν πιο περιεκτικό, απλό και ειρηνικό κόσμο. Η αγωγή, η παιδεία, η μάθηση... «Πώς όμως μαθαίνει κανείς κάτι καινούριο;»

### 1.1 Παιδαγωγικές Θεωρίες Μάθησης

Ο Πλάτων είναι ο πρώτος που προβληματίστηκε για τον τρόπο αυτό. Από την πεποίθησή του ότι οι αισθήσεις οδηγούν στην παγίωση «γνώμης» και όχι «γνώσης», από τη θεώρησή του ότι οι ιδέες αν και έμφυτες είναι δυσδιάκριτες αλλά αληθινές οπότε μόρφωση δεν είναι η διδασκαλία «από έξω προς τα μέσα», αλλά η επαναφορά στο συνειδητό μιας προϋπάρχουσας γνώσης, (σωκρατική μαιευτική και διαλεκτική), μέχρι σήμερα, έχουν διατυπωθεί δεκάδες θεωρίες μάθησης, τακτικές και τεχνικές.

Το φαινόμενο της μάθησης είναι δύσκολο να περιγραφεί ως συγκεκριμένο φυσικό φαινόμενο ή προϊόν εργαστηριακού πειράματος· γίνεται αντιληπτό μόνο από τα αποτελέσματά του [4]. Πέντε βασικές θεωρίες μάθησης είναι οι πιο γνωστές, οι:

- **συμπεριφοριστικές θεωρίες (behaviorism):**
  - έμφαση δίνεται μόνο στα εξωτερικά μετρήσιμα στοιχεία συμπεριφοράς
  - η μάθηση εκλαμβάνεται ως αλλαγή της συμπεριφοράς και το περιβάλλον θεωρείται ο κύριος συντελεστής αυτής της επίδρασης
  - η συνέχεια και η ενίσχυση είναι θεμελιώδεις παράγοντες της μαθησιακής διαδικασίας
  - χαμηλή σημασία στις ιδιαιτερότητες των εκπαιδευομένων
  - παθητική μάθηση, με τυποποιημένα προσδοκώμενα αποτελέσματα
- **γνωστικές θεωρίες (cognitive):**
  - εστιάζουν το ενδιαφέρον τους στη λειτουργία του εγκεφάλου του υποκειμένου
  - η μάθηση χτίζεται σταδιακά, τα απλά στοιχεία αποτελούν τη βάση για να χτιστούν πάνω τους συνθετότερα
  - ο εκπαιδευτής αξιοποιεί προϋπάρχουσα γνώση και εμπειρία των εκπαιδευομένων
  - η μάθηση είναι ενεργητική
- **θεωρίες εποικοδομητισμού (constructivism):**
  - ο εκπαιδευόμενος βασίζεται σε προηγούμενες εμπειρίες για να «κατασκευάσει» μια νέα κατανόηση
  - νόημα κατασκευάζεται μόνο μέσω ενεργής εμπλοκής σε πειράματα ή επίλυση προβλημάτων στον πραγματικό κόσμο
- **ανθρωπιστικές θεωρίες (humanism):**
  - μάθηση υπό την προοπτική της ανάπτυξης της προσωπικότητας
  - εισαγωγή ιεραρχικού συστήματος αναγκών του ατόμου
  - ο εκπαιδευτής δεν είναι δάσκαλος, αλλά καταλύτης
  - η μάθηση ως εσωτερική διεργασία, συντελείται εφόσον το άτομο εμπλακεί ενεργητικά σε αυτή, μέσω αυτενέργειας και πρωτοβουλίας
- **συνδεσιμισμός (connectivism):**
  - έντονη επιρροή από την τεχνολογία

- εντοπίζει και αποκαθιστά τα κενά στη γνώση
- ο εκπαιδευόμενος αντλεί και ενημερώνει πληροφορίες
- η γνώση του πού και πώς θα βρει κανείς τη γνώση εξισώνεται σε σημαντικότητα με τη γνώση

Φυσικά, υπάρχουν κι άλλες ποικίλες θεωρίες μάθησης, όπως οι:

- **θεωρία κοινωνικής μάθησης:**
  - αλλαγή στη συμπεριφορά, που προκαλείται από το εξωτερικό περιβάλλον, ιδιαίτερα από τη συμπεριφορά των άλλων, η οποία λειτουργεί ως πρότυπο προς μίμηση ή αποφυγή
  - το άτομο δεν αντιδρά μηχανιστικά στις εξωτερικές επιδράσεις
  - υπάρχει αλληλεπίδραση μεταξύ ατόμου και εξωτερικών επιδράσεων
- **θεωρία της μετασχηματιστικής μάθησης:**
  - σημαντική ιδιαίτερα για ενήλικες
  - η εμπειρία ζωής και οι γνώσεις συνδυάζονται κριτικά για την αλλαγή της κοσμοθεωρίας των εκπαιδευομένων
- **θεωρία της βιοματικής μάθησης:**
  - δίνει έμφαση στη βιοματική εμπειρία
  - το ζητούμενο είναι να μπορούν οι εκπαιδευόμενοι να εφαρμόσουν τη γνώση σε πραγματικές καταστάσεις

κι οπωσδήποτε κι άλλες και άλλες, αλλά γιατί; Γιατί πρέπει να υπάρχουν τόσες θεωρίες;

Οι θεωρίες αυτές υπάρχουν και αναπτύχθηκαν για να καλύψουν όλα τα διαφορετικά είδη ανθρώπων, όλες τις διαφορετικές προσωπικότητες. Κανένας άνθρωπος δεν είναι ίδιος με κάποιον άλλο, και κανένας εκπαιδευτικός επίσης. Οπότε κάθε ένας επιλέγει τη μέθοδο ή τις μεθόδους που του αρέσουν, που του ταιριάζουν και τον ικανοποιούν και τις εφαρμόζει ανάλογα, για να επιτύχει μια αποτελεσματική τάξη. Το κάνει και για τον εαυτό του και για τους μαθητές του, γιατί και όλοι οι μαθητές δεν είναι ίδιοι, δε μαθαίνουν με τον ίδιο τρόπο και δεν έχουν τις ίδιες ανάγκες.

Σύγχρονες διδακτικές προσεγγίσεις μέσω των οποίων εφαρμόζονται οι παραπάνω θεωρίες είναι:

- επεξεργασία εννοιών/πληροφοριών

- βιωματική
- συνεργατική (ομαδοσυνεργατική)
- σύνθεση έργου (project)
- διαθεματική
- παιχνιδοκεντρική
- ανεστραμμένη τάξη

Στην παρούσα εργασία θα αναλυθεί μία από αυτές: η παιχνιδοκεντρική!

## 1.2 Τι είναι το Παιχνίδι;

Μια λέξη τόσο γνωστή, τόσο οικεία, μια λέξη με την οποία μεγάλωσαν οι περισσότεροι από μας... κι όμως έχουμε αναρωτηθεί ποτέ τι είναι το παιχνίδι; Είναι η γλώσσα των παιδιών, είναι ο τρόπος επικοινωνίας τους ανεξαρτήτως εθνικότητας και καταγωγής, είναι κίνηση, ενέργεια, ζωή, ευτυχία... πώς όμως να «ορισθεί»;! Διάφοροι ερευνητές προσπάθησαν να δώσουν μια ερμηνεία στην πολυσύνθετη αυτή έννοια. Κατά αυτούς, λοιπόν, αντίστοιχα [5] [6] [7] [8] [9] [10] [11] το παιχνίδι είναι,

- μια δραστηριότητα η οποία είναι ελεύθερη επιλογή των παιδιών, κατευθύνεται από εσωτερικά κίνητρα για να παρέχει ευχαρίστηση και ικανοποίηση, καθώς εμπλέκει ενεργά τους παίκτες, είναι αυτο-κατευθυνόμενο και έχει νόημα για το παιδί
- αυτή η συναρπαστική δραστηριότητα στην οποία συμμετέχουν υγιή παιδιά με ενθουσιασμό και ανεμελιά
- μια δραστηριότητα, κατευθυνόμενη από το παιδί, το «νόημα» της οποίας έχει σημασία για το ίδιο και όχι η κατάληξή της
- η δραστηριότητα μέσα από την οποία τα παιδιά καλύπτουν όλες τις αναπτυξιακές τους ανάγκες
- η πραγματοποίηση της μάθησης μέσα από την πράξη
- μία εθελοντική δραστηριότητα ή απασχόληση η οποία πραγματοποιείται μέσα σε κάποια καθορισμένα τοπικά και χρονικά όρια, σύμφωνα με κανόνες ελεύθερα αποδεκτούς, αποτελώντας αυτοσκοπό και συνοδευόμενη από ένα αίσθημα έντασης, χαράς και από τη συνείδηση ότι είναι κάτι «διαφορετικό» από τη «συνήθη ζωή»



- επικοινωνία και έκφραση, συνδυάζει τη σκέψη και τη δράση, προσφέρει ικανοποίηση, είναι ενστικτώδες, εθελοντικό και αυθόρμητο, βοηθάει τα παιδιά να αναπτυχθούν σωματικά, γνωστικά, συναισθηματικά και κοινωνικά, είναι ένα μέσο για να μάθει κάποιος να ζει, και όχι ένας τρόπος για να περνά κάποιος την ώρα του (International Play Association (IPA) for the Child's Right to Play, 2002)

Όλοι οι ορισμοί εμπεριέχουν την ευχαρίστηση, την απόλαυση και την ανεξαρτησία από προκαθορισμένους σκοπούς [12], ενώ μια δραστηριότητα ορίζεται σαν παιχνίδι όταν εμπεριέχει τα παρακάτω στοιχεία [13]:

- την έχουν επιλέξει τα παιδιά και είναι αυτοκατευθυνόμενη
- έχει εσωτερικά κίνητρα, δεν γίνεται για κάποια ανταμοιβή
- κατευθύνεται και υποκινείται από εσωτερικούς κανόνες, αφήνοντας χώρο για δημιουργικότητα
- προϋποθέτει φαντασία
- διατηρεί το παιδί ενεργητικό και σε εγρήγορση χωρίς ένταση και εκνευρισμό

### 1.3 Παιδαγωγικές Θεωρίες Παιχνιδιού & Είδη

Όπως και με τις θεωρίες μάθησης, άλλες τόσες θεωρίες έχουν αναπτυχθεί για το πώς εμπλέκεται το παιδί με το παιχνίδι και πώς το δεύτερο επηρεάζει το πρώτο. Επιγραμματικά οι θεωρίες αυτές είναι:

- πλεονάζουσας ενεργητικότητας
- επιτυχίας και εξάρσεως του «εγώ»
- προγονικής κληρονομικότητας
- βιωματικής μάθησης
- του Piaget
- του Vygotsky
- ψυχαναλυτική (ή θεωρία της κάθαρσης)

Αναφορικά με τα είδη των παιχνιδιών, η διάκριση γίνεται συνήθως, ανάλογα με την ηλικία και το επίπεδο ανάπτυξης του παιδιού. Με κριτήριο αυτό, το παιχνίδι μπορεί να είναι:

- εξερευνητικό, με αντικείμενα ή αισθητηριακό

- κατασκευαστικό
- δραματοποίησης (μοναχική μίμηση)
- αισθησιοκινητικό, λειτουργικό
- συμβολικό
- κοινωνικό δραματοποιημένο ή συντροφικό
- με κανόνες, συνεργατικό
- με επινοημένους κανόνες

ενώ μια δεύτερη κατηγοριοποίηση, βασισμένη στο είδος της αλληλεπίδρασης του παιδιού, διακρίνει το παιχνίδι ως:

- μοναχικό
- παρατήρησης
- παράλληλο
- συσχετιστικό
- συνεργατικό

Όπως κι αν το χαρακτηρίσουμε, ένα παιχνίδι έχει τεράστια σημασία στην ανάπτυξη του παιδιού και κατ' επέκταση του ανθρώπου. Τόσο μάλιστα σημαντική είναι η συμβολή στην υγιή ανάπτυξη, που θεωρείται πια βασικό και αναφαίρετο δικαίωμα του παιδιού και υποχρέωση των ενηλίκων να τους το παρέχουν. Η Διεθνής Σύμβαση για τα Δικαιώματα του Παιδιού, στο άρθρο 31, ορίζει μεταξύ άλλων ότι τα Συμβαλλόμενα Κράτη αναγνωρίζουν στο παιδί το δικαίωμα στην ανάπαυση και στις δραστηριότητες του ελεύθερου χρόνου, στην ενασχόληση με ψυχαγωγικά παιχνίδια και δραστηριότητες που είναι κατάλληλες για την ηλικία του και στην ελεύθερη συμμετοχή του στην πολιτιστική και καλλιτεχνική ζωή.

Δίνεται τόσο μεγάλη σημασία στο παιχνίδι γιατί αυτό συμβάλλει στη σωματική, κινητική αλλά και κοινωνική ανάπτυξη του παιδιού, προάγει τη συναισθηματική ανάπτυξη και τη δημιουργικότητά του και συμπληρωματικά ενισχύει τη γνωστική του ανάπτυξη. Από τη βρεφική ηλικία ωφελεί τις διαδικασίες γνώσης, με μιμητισμό, δραματοποίηση, συμβολικές σκέψεις και αργότερα νοητικές. Το δραματικό παιχνίδι προωθεί τη χρήση των δεξιοτήτων ομιλίας και ακρόασης. Όταν τα παιδιά συμμετέχουν σε αυτό το είδος παιχνιδιού εξασκούνται σε λέξεις που ακούν αλλά και συνειδητοποιούν ότι πρέπει να ακούσουν τι λένε οι άλλοι «παίκτες» για να μπορέσουν να ανταποκριθούν με τον κατάλληλο τρόπο. Ακόμη, τους διδάσκει να επιλέγουν τα

λόγια τους με σύνεση, ώστε οι άλλοι να καταλάβουν τι ακριβώς λένε, άρα προσπαθούν να επικοινωνήσουν διαμορφώνοντας ένα προσωπικό λεξιλόγιο [14].

## 1.4 Παιχνιδοποίηση (Gamification)

Είναι δύσκολο να αρνηθεί κανείς τη συμμετοχή του παιχνιδιού στην εκπαιδευτική διαδικασία και τη μάθηση. Στην πραγματικότητα, το παιχνίδι φέρνει μαθησιακά αποτελέσματα με έναν φυσικό και αβίαστο τρόπο, αφού προωθεί τη συμμετοχή, την αλληλεπίδραση, την ενίσχυση προσωπικών κινήτρων, την απόκτηση εμπειριών και τη δημιουργία νέων στρατηγικών [15]. Ακολουθώντας την ταχύτητα ανάπτυξης των πάντων στη σύγχρονη κοινωνία, τα παιδιά προτιμούν δραστηριότητες που απαιτούν ενεργητική δράση, που είναι γρήγορες και οδηγούν στην απόκτηση πληροφοριών με διάφορους τρόπους [16]. Είναι προφανές ότι τα αποτελέσματα είναι πολύ ευνοϊκότερα όταν οι μαθητές συμμετέχουν πρόσχα σε ένα παιγνιώδες περιβάλλον, με άμεση αύξηση των γνώσεων και της ακαδημαϊκής τους επίδοσης, αλλά ταυτόχρονα των δεξιοτήτων τους και της σωματικής τους υγείας.

Συχνά η κοινωνία αρνείται ή επιφυλάσσεται να αποδεχθεί το προφανές. Ενώ, λοιπόν, η χρήση των παιχνιδιών και των προσομοιώσεων εισέρχεται δυναμικά σε εταιρείες και στρατιωτικά περιβάλλοντα, στην εκπαίδευση προχωράει με μικρά-μικρά βήματα, κυρίως στις μικρές τάξεις του Δημοτικού, ενώ απαξιώνεται από το σύνολο της εκπαιδευτικής κοινότητας (εκπαιδευτικοί, μαθητές και γονείς) στις μεγαλύτερες τάξεις κι ιδιαίτερα στη δευτεροβάθμια εκπαίδευση. Για την ανώτερη εκπαίδευση μάλλον δε γίνεται ούτε λόγος... Ευτυχώς οι εκπαιδευτικοί ειδικής αγωγής είναι οι πρώτοι που άρχισαν να αντιλαμβάνονται την αξία και τη βοήθεια των παιχνιδιών στη δουλειά τους.

### 1.4.1 Τι είναι τελικά η Παιχνιδοποίηση

Είναι η παιχνιδοποίηση ταυτόσημη με το παιχνίδι; Μπορούν να χρησιμοποιηθούν οι δύο όροι με την ίδια έννοια; Σίγουρα παιχνιδοποίηση δε σημαίνει να μετατραπούν τα πάντα σε παιχνίδι. Σίγουρα δεν αποσκοπεί στην κατάργηση της μάθησης, αλλά στη συμπλήρωσή της. Οποσδήποτε πρέπει να μείνει επικεντρωμένη στο στόχο της: τη μάθηση. Η παιχνιδοποίηση έχει σαν στόχο την απόκτηση ικανότητας:

- μάθησης από τα παιχνίδια
- μάθησης από τις επιστήμες που σχετίζονται με τα παιχνίδια

- εξασφάλισης της διασκέδασης και απόλαυσης του παίκτη, αυτού του συστατικού, δηλαδή, που μπορεί να συντηρήσει το κίνητρο ολοκλήρωσης της δραστηριότητας [17].

Με τον όρο παιχνιδοποίηση εννοούμε την υιοθέτηση της τεχνολογίας και των μεθόδων σχεδιασμού παιχνιδιών σε τομείς μη σχετικούς με τη βιομηχανία παιχνιδιών [18]. Η διαφορετικά, το gamification είναι η χρήση στοιχείων παιχνιδιού και τεχνικών σχεδιασμού παιχνιδιού σε περιβάλλοντα εκτός παιχνιδιού [19]. Επομένως παιχνιδοποίηση δεν είναι η άμεση, ξερή χρήση παιχνιδιών αλλά η εκμετάλλευση των στοιχείων που περιλαμβάνουν τα παιχνίδια και των τεχνικών με τις οποίες σχεδιάζονται προκειμένου να επιτευχθεί η επιθυμητή εμπειρία των χρηστών/μαθητών. Ένα εκπαιδευτικό παιχνίδι πρέπει να σχεδιαστεί με βάση τα βιώματα που είναι επιθυμητό να αποκτήσει ο χρήστης, επιλέγοντας τα κατάλληλα στοιχεία και ο σχεδιασμός αυτός έχει αντίκτυπο στο πόσο έχουν επιτευχθεί οι αρχικοί στόχοι [20]. Επιλέγοντας τα κατάλληλα στοιχεία παιχνιδιού, σε οποιαδήποτε παιχνιδοποιημένη εφαρμογή, είναι δυνατή και επιτυχημένη η εμπλοκή των παικτών/εκπαιδευομένων στην εμπειρία που θα διεγείρει το ενδιαφέρον του, ικανοποιώντας θεμελιώδεις ανάγκες του, όπως η ανταμοιβή, ο ανταγωνισμός, η αγωνιστικότητα, κλπ [21].

#### **1.4.2 Στοιχεία Παιχνιδιών Απαραίτητα στην Παιχνιδοποίηση**

Με βάση την Ψυχοπαιδαγωγική και την εξοικείωση των υποψήφιων εκπαιδευομένων, οι παιχνιδοποιημένες εφαρμογές πρέπει να περιλαμβάνουν τα ακόλουθα δεκαπέντε σημαντικά δομικά στοιχεία [22]:

- Επιτεύγματα: απόδοση αμοιβής για επίτευξη στόχων
- Άβαταρ: εικονικό προφίλ ενός παίκτη ή ψηφιακού βοηθού
- Διακριτικά σήματα: οπτικές απεικονίσεις των επιτευγμάτων
- Boss Fights: απαιτητικοί στόχοι προς το τέλος κάποιου επιπέδου
- Συλλογές: διαφόρων εικονικών αντικειμένων ή διακριτικών σημάτων
- Μάχη: καθορισμένη και συνήθως σύντομη αναμέτρηση
- Ξεκλείδωμα περιεχομένου: επίτευγμα ή επιτυχής ολοκλήρωση ενός επιπέδου μπορεί να δώσει στον παίκτη τη δυνατότητα πρόσβασης σε υλικό που θα διευκολύνει το έργο του στη συνέχεια
- Προσφορά: δυνατότητα ένας παίκτης να μοιραστεί πόρους με άλλους παίκτες

- Πίνακες κατάταξης: οπτική απεικόνιση της προόδου ενός παίκτη σε σχέση με την πρόοδο άλλων παικτών
- Επίπεδα: καθορισμένα επίπεδα της προόδου ενός παίκτη
- Πόντοι: αριθμητική απεικόνιση της προόδου ενός παίκτη
- Αποστολές: προκαθορισμένες προκλήσεις με στόχους και αμοιβές
- Κοινωνικά γραφήματα: απεικόνιση των παικτών με τους οποίους ένας παίκτης μπορεί να συνεργαστεί ή να παίζει ενάντια τους
- Ομάδες: ομάδες παικτών που συνεργάζονται για ένα κοινό στόχο
- Εικονικά αγαθά: οι παίκτες διαθέτουν άλλους εικονικούς πόρους που κατέχουν ή ακόμα και πραγματικά χρήματα, προκειμένου να αποκτήσουν κάποια εικονικά αγαθά

Επειδή η παιχνιδοποίηση σχετίζεται με παιχνίδια πολέμου ή προσομοίωσης, σοβαρά παιχνίδια (δηλαδή που αναπτύσσονται όχι για ψυχαγωγία) και παιχνίδια εναλλακτικής πραγματικότητας, μπορεί να μην περιλαμβάνονται κάθε φορά και τα δεκαπέντε στοιχεία που προαναφέρθηκαν. Τα πιο απαραίτητα όλων όμως είναι οι **πόντοι** που συλλέγουν οι παίκτες και απεικονίζουν την πρόδο τους, τα **διακριτικά σήματα**, που ανάλογα με την ιδιοσυγκρασία τους μπορεί να αποτελέσουν ισχυρό κίνητρο δέσμευσης και οι **πίνακες κατάταξης** που ανατροφοδοτούν για τη θέση του παίκτη στο σύνολο της κοινότητας που συμμετέχει.

Μοιάζει παράδοξο αλλά είναι αληθές ότι η παιγνιώδης μάθηση, που είναι τόσο σύγχρονη, ελεύθερη και ελπιδοφόρα, έχει πολλά στοιχεία συμπεριφορισμού! Τόσο οι ανταμοιβές, όσο και η άμεση ανατροφοδότηση που πρέπει να εκλαμβάνει ο χρήστης, πηγάζουν από τη μέθοδο που δείχνει λιγότερο επιθυμητή πια και θεωρείται, από τους περισσότερους, παρωχημένη. Ωστόσο, η παιχνιδοποίηση είναι ένα καλό και σαφές παράδειγμα του πώς οι μέθοδοι, από την πιο παλιά ως την πιο σύγχρονη, μπορούν να συνδυαστούν αποτελεσματικά προς όφελος των εκπαιδευομένων.

#### 1.4.3 Η Παιχνιδοποίηση στην Εκπαίδευση

Η ανάπτυξη των Τεχνολογιών Πληροφορικής & Επικοινωνιών (ΤΠΕ) και η εισαγωγή τους στην εκπαίδευση, μαζί με τη χρήση του διαδικτύου, έφερε τους μαθητές πιο κοντά στην παιγνιώδη μάθηση. Αν τα ηλεκτρονικά παιχνίδια προσελκύουν το ενδιαφέρον, τότε η χρήση των στοιχείων τους που τα κάνει ενδιαφέροντα, θα μπορούσε να κάνει και την εκπαίδευση μια ενδιαφέρουσα διαδικασία [23]. Αυτή είναι η πιο

ουσιαστική ιδέα πίσω από το gamification: χρησιμοποιώντας οικείες ενέργειες, εικόνες και συνήθειες, ακόμη και ορολογία από δραστηριότητες παιχνιδιών, να μπορεί κανείς να διδάξει αβίαστα, χωρίς να νιώθει ο μαθητής ότι διδάσκεται. Υποστηρικτικά, να χρησιμοποιεί κάθε επίτευγμα προς όφελος όχι μόνο της γνωστικής αλλά και της ανθρώπινης γενικότερα υπόστασής του. Παραδείγματος χάριν, αν συνειδητοποιεί ο εκπαιδευόμενος ότι ξεπερνά εμπόδια και είναι ικανός, στο παιχνίδι, νιώθει το ίδιο δυνατός και στη ζωή και γεμίζει συναισθήματα ευχαρίστησης και ενδιαφέροντος για να συνεχίσει [22].



Εικόνα 2: Πυραμίδα της Μάθησης

Στην παραπάνω εικόνα φαίνεται η πυραμίδα της μάθησης, σύμφωνα με την οποία, ένας εκπαιδευόμενος εάν προσέχει συγκρατεί κατά προσέγγιση:

- 5% αυτών που ακούει
- 10% αυτών που διαβάζει
- 20% αυτών που βλέπει και ακούει
- 30% αυτών που καταλαβαίνει παρακολουθώντας επίδειξη
- 50% όσων μαθαίνει συζητώντας ομαδικά
- 75% από εξάσκηση, εφαρμογή στην πράξη όσων έμαθε

- 90% όταν διδάσκει άλλους ή δρα ενεργητικά και σκέφτεται ταυτόχρονα

ενώ οι ειδικοί αποφαινόμενοι ότι η εμπειρία αποτελεί τη βάση κάθε μάθησης [24]. Με την παιγνιώδη μάθηση, γίνεται προσπάθεια πολλαπλής ενεργοποίησης των μαθητών σε όλα τα επίπεδα προκειμένου να συγκρατήσουν όσο δυνατόν περισσότερες πληροφορίες στο μαθησιακό τους θησαυροφυλάκιο, με μια εκπαίδευση αντάξια ή ισάξια με την άμεση πρόσβασή τους σε πηγές γνώσεων και διασκέδασης στο διαδίκτυο.

Προσδίδοντας έναν αγχολυτικό χαρακτήρα [25]

- εξελίσσοντας την εκπαιδευτική διαδικασία σε κλιμακούμενης δυσκολίας επίπεδα,
- αναπτύσσοντας και υιοθετώντας σύστημα βαθμολόγησης των γνώσεων,
- ανατροφοδοτώντας σε πραγματικό χρόνο σχετικά με τη γνώση που αποκτήθηκε,
- καθιερώνοντας γραμμές προόδου πάνω στις οποίες να βασίζεται η αξιολόγηση των εκπαιδευομένων,
- καθιερώνοντας ροές δραστηριοτήτων τις οποίες καλούνται να εκτελέσουν διαδοχικά οι εκπαιδευόμενοι,
- επιτρέποντας το λάθος και την επαναπροσπάθεια,

μπορεί να ενισχυθεί η αποδοτικότητα της εκπαιδευτικής διαδικασίας!

Ο γενικός σκοπός των αναλυτικών προγραμμάτων περιστρέφεται σε τρεις άξονες: τις αξίες, τη γνωστική σφαίρα και την κοινωνικοποίηση, αποβλέποντας στη δημιουργία ελεύθερων και δημοκρατικών πολιτών [26]. Η προσέγγιση της εκπαίδευσης με την παιγνιοποίηση παρουσιάζει το πλεονέκτημα ότι μπορεί να διατρέξει με επιτυχία και τους τρεις επιθυμητούς άξονες και χρησιμοποιώντας χαρακτηριστικά παιχνιδιών να αυξήσει την εμπλοκή των μαθητών. Σε σχέση, μάλιστα με άλλες παραδοσιακές μεθόδους ενισχύει την αυτονομία, αλλά και τη συνεργασία και βοηθάει να ξεδιπλωθούν ακόμη και μαθητές με ηπιότερα προφίλ και χαμηλότερη συμμετοχή στην τάξη.

Μελετώντας τον Πίνακα 1 που ακολουθεί, είναι εμφανείς οι ομοιότητες των ρόλων στις δύο περιπτώσεις [27]: παίκτες στο παιχνίδι, μαθητές στο σχολείο. Αν οι εκπαιδευτές κατάφεραν να εμψυχήσουν την ιδέα στους μαθητές τους, πως η εκπαίδευσή τους μοιάζει με ένα παιχνίδι, τα αποτελέσματα μάλλον θα ήταν θεαματικά.

**Πίνακας 1: Χαρακτηριστικά Παιχνιδιών και Εκπαίδευσης**

	<b>χαρακτηριστικά παιχνιδιών</b>	<b>χαρακτηριστικά εκπαίδευσης</b>
<b>σκοπός</b>	η νίκη	η μάθηση
<b>πρωταγωνιστής</b>	ο παίκτης: αγωνίζεται να πετύχει το σκοπό προσπαθώντας να ξεπεράσει κάποια εμπόδια	ο μαθητής: αγωνίζεται να πετύχει το σκοπό ξεπερνώντας εμπόδια και κατακτώντας γνώση
<b>μετακίνηση</b>	από ένα ευκολότερο επίπεδο σε δυσκολότερο	οι δραστηριότητες έχουν αυξανόμενο βαθμό δυσκολίας
<b>η πρόοδος</b>	η εξέλιξη του παιχνιδιού καθορίζεται από την πρόοδό του και είναι σημαντική	η μαθησιακή του πορεία εξαρτάται από τις γνώσεις που έχει κατακτήσει

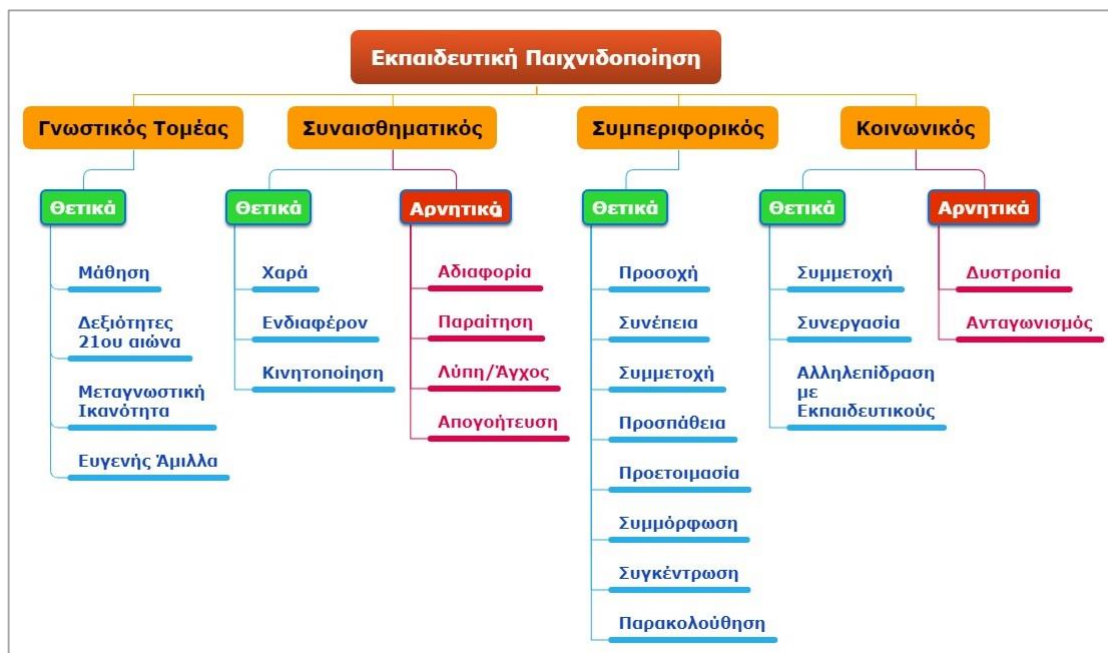
Θα ήταν αστοχία να μην αναφερθούν και κάποιες πιθανές αρνητικές εκβάσεις της παιχνοποίησης. Όπως και στα παιχνίδια, έτσι και στην παιγνιώδη μάθηση, ανάλογα με την ιδιοσυγκρασία των παικτών/μαθητών μπορεί να αναπτυχθεί ανταγωνισμός και δυστροπία στη συμπεριφορά, μπορεί να δημιουργηθούν αισθήματα λύπης ή άγχους σε περιπτώσεις «ήττας», ακόμη και να οδηγηθούν τα παιδιά σε αδιαφορία ή απογοήτευση, είτε λόγω κακής επίδοσης στο γνωστικό μέρος, είτε στο διαδικαστικό, το παιχνοποιημένο.

#### **1.4.4 Επιγραμματικά και Διαγραμματικά**

Εντάσσοντας στοιχεία παιχνοποίησης στην εκπαιδευτική διαδικασία ενισχύονται όλες εκείνες οι δεξιότητες και συμπεριφορές που οδηγούν στην κατάκτηση γνώσεων από τους μαθητές ενώ αυξάνεται η συμμετοχή και το ενδιαφέρον τους σε κάθε στάδιο της διαδικασίας αυτής [28]. Οι αλλαγές, ωστόσο και η αποδοχή νέας νοοτροπίας και φιλοσοφίας γίνονται αργά. Οι ολοκληρωμένοι, ισορροπημένοι, ψυχικά υγιείς και σταθεροί άνθρωποι μπορούν να προχωρήσουν άφοβα, αλλά με σύνεση, στην επιβολή αλλαγών, ανταποκρινόμενοι στις πολύπτυχες απαιτήσεις και προσδοκίες των νέων δεδομένων. Δεν υπάρχουν παιδαγωγικά ιδεώδη ή σκοποί αγωγής με διαχρονικό και απόλυτο κύρος [29], ούτε μέθοδοι χωρίς αρνητικά και πιθανές ανεπιθύμητες



επιπτώσεις. Τα συμπεράσματα για την παιχνιδοποίηση στην εκπαίδευση φαίνονται στον ακόλουθο επαυξημένο εννοιολογικό χάρτη [30]:



Εικόνα 3: Επίδρασης της Παιχνιδοποίησης στην Εκπαίδευση

### 1.4.5 Παιχνιδοποίηση και Εκπαιδευτικά Παιχνίδια

Η παιχνιδοποίηση, όπως αναφέρθηκε επανειλημμένα, χρησιμοποιεί ως όχημα την τεχνολογία, αλλά δεν υπαινίσσεται τη δημιουργία ενός παιχνιδιού (το οποίο, μάλιστα, απαιτεί γνώσεις σχεδιασμού, κόστος και χρόνο)· χρησιμοποιώντας τα «ευχάριστα» στοιχεία των παιχνιδιών, στοχεύει στο να κάνει την εκπαίδευση πιο ευχάριστη και ελκυστική χωρίς όμως να υποσκάπτει την αξιοπιστία της [31].

Με απλά λόγια, τα εκπαιδευτικά παιχνίδια οφείλουν να έχουν τα χαρακτηριστικά της παιχνιδοποίησης, ενώ η παιχνιδοποίηση δεν αντιστοιχεί υποχρεωτικά σε παιχνίδι. Και τα δύο σχετίζονται με τη συμμετοχή σε «παιχνίδι», αλλά είναι σημαντικό να αναγνωρίζονται οι διαφορές τους [32], που συνοψίζονται στον Πίνακα 2.

Το σημαντικό είναι ότι και τα εκπαιδευτικά παιχνίδια και η ένταξη παιχνιδοποίησης συνεισφέρουν καταλυτικά στην επιτυχία της εκπαιδευτικής διαδικασίας, παρέχοντας με τη βοήθεια των ΤΠΕ, εργαλεία ικανά να ερεθίσουν όλες τις αισθήσεις των μαθητών, οδηγώντας στην επίτευξη των μαθησιακών στόχων και μετατρέποντας την εκπαιδευτική διαδικασία, από βαρετή και δυσνόητη, σε συναρπαστική και κατανοητή [27]!

**Πίνακας 2: Διαφορές Εκπαιδευτικών Παιχνιδιών και Παιχνιδοποίησης**

<b>Εκπαιδευτικά Παιχνίδια</b>	<b>Παιχνιδοποίηση στην Εκπαίδευση</b>
η επιβράβευση είναι στοιχείο του παιχνιδιού	το σύστημα επιβράβευσης του εκπαιδευόμενου βασίζεται στην αφομοίωση όσων διδάχτηκε
το κατάλληλο επίπεδο προκλήσεων αποτελεί σκοπό τους	η πρόκληση δεν αποτελεί βασική προτεραιότητα της
χρησιμοποιούν αφήγηση και κοινούς ήρωες/χαρακτήρες	μπορεί να περιλαμβάνουν avatars και ο εκπαιδευόμενος να επιβλέπει τη χρήση τους
εστιάζουν στις εννοιολογικές μεταβολές	εστιάζουν στη μεταβολή συμπεριφοράς του εκπαιδευόμενου
προσομοιώνουν το περιβάλλον παρέχοντας το εκπαιδευτικό πλαίσιο	εφαρμόζονται στο πραγματικό περιβάλλον

*Οι άνθρωποι δεν σταματούν να παίζουνε επειδή γερνάνε.*

*Γερνάνε επειδή σταματούν να παίζουνε.*

Oliver W. Holmes, 1809-1894, Αμερικανός συγγραφέας

## 2. Αντικειμενοστραφής Προγραμματισμός

---

Από τη δημιουργία του πρώτου υπολογιστή μέχρι σήμερα έχουν αλλάξει πάρα πολλά πράγματα. Από τους τεράστιους σε μέγεθος, με πάρα πολύ περιορισμένες δυνατότητες και μικρές ταχύτητες επεξεργασίας, φτάσαμε σε πολύ μικρούς σε μέγεθος υπολογιστές, με τεράστιες δυνατότητες και απίστευτες ταχύτητες επεξεργασίας [33]. Καθώς το υλικό (hardware) των υπολογιστών βελτιώνεται και κατά συνέπεια παρέχει πρωτόγνωρες ευκαιρίες ανάπτυξης εφαρμογών, οι γλώσσες προγραμματισμού εξελίσσονται και συνεχώς εμπλουτίζονται, οδηγώντας και τον προγραμματισμό σε καινούριους δρόμους.

### 2.1 Ιστορική Αναδρομή Γλωσσών Προγραμματισμού

Όλα ξεκίνησαν από την πρώτη προσπάθεια προγραμματισμού, που ήταν η επικοινωνία απευθείας με τη μηχανή. Επομένως θα έπρεπε ο άνθρωπος να μιλήσει τη γλώσσα της.

- **Πρώτη Γενιά - Γλώσσες Μηχανής:** Για να μπορέσει ο υπολογιστής να εκτελέσει μία οποιαδήποτε λειτουργία, έπρεπε να δοθούν κατευθείαν οι κατάλληλες ακολουθίες από 0 και 1, δηλαδή εντολές σε μορφή κατανοητή από τον υπολογιστή. Ο προγραμματισμός σε γλώσσα μηχανής ήταν επίπονος και ελάχιστοι μπορούσαν να τον υλοποιήσουν, αφού απαιτούσε βαθιά γνώση του υλικού και της αρχιτεκτονικής του υπολογιστή. Ο προγραμματιστής έπρεπε να ξέρει ακριβώς τη δυαδική αναπαράσταση των εντολών.
- **Δεύτερη γενιά - Συμβολικές γλώσσες ή γλώσσες χαμηλού επιπέδου (Assembly):** Προκειμένου ο προγραμματιστής να μην χρειάζεται να ξέρει ακριβώς τη δυαδική αναπαράσταση των εντολών, ξεκίνησε η χρήση μιας συμβολικής γλώσσας, η οποία ενώ θα είχε έννοια για τον άνθρωπο, θα μετατρεπόταν εσωτερικά από τους υπολογιστές στις αντίστοιχες ακολουθίες από 0 και 1. Αυτό ήταν δουλειά του συμβολομεταφραστή (assembler). Τώρα ο άνθρωπος χρησιμοποιεί πιο κατανοητούς μνημονικούς κανόνες αλλά οι γλώσσες εξαρτώνται από το hardware.
- **Τρίτη Γενιά - Γλώσσες υψηλού (high-level) επιπέδου:** Οι ανεπάρκειες των συμβολικών γλωσσών και η ανάγκη για καλύτερη επικοινωνία ανθρώπου-

μηχανής, οδήγησαν, στα τέλη της δεκαετίας του 50, στην εμφάνιση των πρώτων γλωσσών προγραμματισμού υψηλού επιπέδου. Τώρα, ο προγραμματιστής γράφει το πρόγραμμά του με εντολές που είναι πιο κοντά στη γλώσσα του και απομνημονεύονται ευκολότερα. Η γλώσσα αυτή είναι κατανοητή από τον ίδιο και καλά δομημένη (source code). Το πρόγραμμα μεταφράζεται από τον ίδιο τον υπολογιστή σε έναν ενδιάμεσο κώδικα (object code) με τη βοήθεια ενός ειδικού προγράμματος, που ονομάζεται μεταγλωττιστής (compiler) και ο ενδιάμεσος κώδικας μετατρέπεται στις ακολουθίες των εντολών της μηχανής (machine code). Η γλώσσα είναι ανεξάρτητη από τον υπολογιστή.

- **Τέταρτη γενιά - Εξειδικευμένες γλώσσες:** Σταδιακά πολλές γλώσσες εφοδιάστηκαν με εργαλεία προγραμματισμού που αποκρύπτουν πολλές λεπτομέρειες από τις τεχνικές υλοποίησης και με αυτά ο χρήστης μπορεί να επιλύει μόνος του μικρά προβλήματα εφαρμογών [33]. Αυτή η αυξανόμενη τάση απόκρυψης της αρχιτεκτονικής του υλικού και της τεχνικής του προγραμματισμού οδήγησε στις γλώσσες 4ης γενιάς. Στις γλώσσες αυτές που έχουν σχεδιαστεί για την επίτευξη ενός συγκεκριμένου στόχου (όπως η ανάπτυξη εμπορικών επιχειρηματικών εφαρμογών) ο χρήστης ενός υπολογιστή έχει τη δυνατότητα, σχετικά εύκολα, να υποβάλει ερωτήσεις σε ένα σύστημα ή να αναπτύξει εφαρμογές που ανακτούν πληροφορίες από βάσεις δεδομένων και να καθορίζει τον ακριβή τρόπο εμφάνισης αυτών των πληροφοριών ή να κάνει άλλες εξειδικευμένες εργασίες.
- **Πέμπτη γενιά - «Φυσικές» γλώσσες:** Ο προγραμματισμός προσπαθεί να μιμηθεί τη φυσική γλώσσα. Οι γλώσσες προγραμματισμού πέμπτης γενιάς επιτρέπουν στους προγραμματιστές να λύσουν προβλήματα καθορίζοντας περιορισμούς αντί να συντάξουν έναν αλγόριθμο. Γι' αυτό μπορούν να χρησιμοποιηθούν για την επίλυση προβλημάτων χωρίς την παρέμβαση προγραμματιστή. Χρησιμοποιούνται στην έρευνα τεχνητής νοημοσύνης και φάνταζαν ως το μέλλον του προγραμματισμού αφού σχεδιάστηκαν ώστε να επιτρέπουν στους υπολογιστές να επιλύουν τα προβλήματα μόνοι τους. Ωστόσο, αυτό το σενάριο δεν υπολόγιζε ότι οι περιορισμοί, ακόμα, έπρεπε να οριστούν από τον άνθρωπο-προγραμματιστή!

Κάθε νέα γενιά προσθέτει κι ένα ακόμα επίπεδο αφαίρεσης, δίνοντας ελευθερία στον προγραμματιστή να κάνει περισσότερα πράγματα, γνωρίζοντας λιγότερα!

## 2.2 Τεχνικές Σχεδίασης Προγραμμάτων

Από το πρώτο εγχείρημα προγραμματισμού σε γλώσσα μηχανής, η προσπάθεια των προγραμματιστών ήταν η δημιουργία απλών, εύκολων στη σύνταξη και την κατανόησή τους, κομψών και φυσικά αποδοτικών προγραμμάτων.

- **Ιεραρχική Σχεδίαση:** Η σχεδίαση αυτή είναι κυρίως γνωστή ως «από πάνω προς τα κάτω» ή top-down design. Η ιδέα είναι η διάσπαση του προβλήματος σε σειρά από υποπροβλήματα που είναι πιο απλά και ευκολότερα διαχειρίσιμα. Λύνοντας αυτά, επιλύεται και το αρχικό. Η διαδικασία αυτής της σχεδίασης υποστηρίζεται από διαγραμματικές τεχνικές.
- **Τμηματικός Προγραμματισμός:** Έχοντας σχεδιάσει και αναπαραστήσει ιεραρχικά ένα πρόβλημα, κάθε υποενότητα του είναι ανεξάρτητη και πρέπει να αντιμετωπιστεί ξεχωριστά. Σε αυτό έγκειται ο τμηματικός προγραμματισμός, ο οποίος διευκολύνει τη δημιουργία του προγράμματος, μειώνει τα λάθη και επιτρέπει την ευκολότερη κατανόηση, παρακολούθηση και συντήρησή του, από τον ίδιο τον προγραμματιστή ή από άλλους.
- **Δομημένος ή Διαδικασιακός Προγραμματισμός:** Όταν οι προγραμματιστές απαρνήθηκαν την «άτακτη» μετατόπιση της ροής των εντολών ενός προγράμματος (με τη χρήση της γνωστής αμφιλεγόμενης εντολής GOTO) αναπτύχθηκε ο δομημένος προγραμματισμός. Δε γεννήθηκε απλά ένα νέο είδος προγραμματισμού αλλά μια φιλοσοφία στη δημιουργία λογισμικού, που θα βοηθούσε στην ανάπτυξη σύνθετων προγραμμάτων με λιγότερα λάθη, καλύτερη συντήρηση, ευκολότερες διορθώσεις και αλλαγές. Χρησιμοποιώντας μόνο τις δομές ακολουθίας, επιλογής και επανάληψης, σε ενότητες προγράμματος με μία είσοδο και μία έξοδο, ευνοεί και εμπεριέχει την ιεραρχική σχεδίαση και τον τμηματικό προγραμματισμό. Οι οργάνωση του κώδικα γίνεται σε διαδικασίες (procedures) και σχεδόν όλες οι σύγχρονες γλώσσες τον υποστηρίζουν.
- **Αντικειμενοστραφής Προγραμματισμός:** Η τελευταία τάση στην αντιμετώπιση προγραμματιστικών αντιλήψεων και δομών, που άλλαξε ριζικά τα σταθερά και γνωστά σημεία αναφοράς των προγραμματιστών είναι ο αντικειμενοστραφής προγραμματισμός. Η αντικειμενοστραφής σχεδίαση ξεκινάει από την απλή ιδέα του να περιγραφούν οι «ενέργειες» που

εφαρμόζονται πάνω σε δεδομένα. Θεωρεί ότι τα δεδομένα είναι αυτά πάνω στα οποία πρέπει να βασίζεται η δομή του προγράμματος, παρά στις ενέργειες και αυτή είναι η διαφορά που ξεχωρίζει τις παραδοσιακές προγραμματιστικές τεχνικές από την αντικειμενοστραφή.

## 2.3 Αντικειμενοστραφής Προγραμματισμός – Τα Βασικά

Ο αντικειμενοστραφής προγραμματισμός αναπτύχθηκε για να αντιμετωπισθούν τα προβλήματα του διαδικασιακού προγραμματισμού. Ο προγραμματισμός που βασίζεται σε διαδικασίες είναι κατάλληλος για μικρά προγράμματα, αλλά σε περιπτώσεις μεγάλων και σύνθετων παρουσιάζουν σοβαρά ζητήματα στη σχεδίαση, την υλοποίηση (ιδιαίτερα όταν δουλεύονται από ομάδες προγραμματιστών), την αποσφαλμάτωση, τη συντήρηση και την τροποποίηση.

Ο αντικειμενοστραφής προγραμματισμός, δίνει προτεραιότητα στα δεδομένα, βάζοντας μαζί τους και τις μεθόδους τους, δηλαδή διαδικασίες που σχετίζονται με αυτά και τα ελέγχουν. Αυτό το επιτυγχάνει χρησιμοποιώντας αντικείμενα και κλάσεις [34].

### 2.3.1 Αντικείμενα

Τα δεδομένα είναι αυτά που αντιπροσωπεύουν την κατάσταση του αντικειμένου. Κάθε αντικείμενο είναι ευδιάκριτο και αυτόρκες, ενώ το σύνολο των χαρακτηριστικών ιδιοτήτων του προσδιορίζουν με λεπτομέρεια τη φυσική του υπόσταση. Κάθε αντικείμενο έχει:

- μια κατάσταση, η οποία ορίζεται από συγκεκριμένα χαρακτηριστικά,
- μια συμπεριφορά, που ορίζεται από συγκεκριμένες ενέργειες που μπορεί να εκτελέσει το αντικείμενο και
- μια ταυτότητα που το ξεχωρίζει από τα υπόλοιπα.

Σε μια εφαρμογή, ένα αντικείμενο είναι ο συνδυασμός κώδικα και δεδομένων, τα οποία μπορούν να χειριστούν ενιαία. Τα δεδομένα (data) αποτελούν τα χαρακτηριστικά και οι ενέργειες καθορίζουν τη συμπεριφορά ενός αντικειμένου.

### 2.3.2 Κλάσεις

Η κλάση είναι μια αφηρημένη περιγραφή ενός αντικειμένου με συγκεκριμένα χαρακτηριστικά και συγκεκριμένη συμπεριφορά. Είναι σαν μια μηχανή παραγωγής αντικειμένων. Αντικείμενα που ανήκουν στην ίδια κλάση είναι στιγμιότυπά της, με

κοινές ιδιότητες και μεθόδους, οι οποίες έχουν συγκεκριμένες τιμές για το καθένα. Οι κλάσεις σχηματίζουν μια ιεραρχία για να μοντελοποιήσουν το σύστημα του πραγματικού κόσμου με το οποίο ασχολείται ο προγραμματιστής. Η ιεραρχία αναπαρίσταται ως κληρονομικότητα και οι κλάσεις μπορούν επίσης να συσχετιστούν με διαφορετικούς τρόπους ανάλογα με την απαίτηση.

### 2.3.3 Χαρακτηριστικά Αντικειμενοστραφών Εφαρμογών

Οι κλάσεις χρησιμοποιούνται για την αναπαράσταση αντικειμένων. Τα αντικείμενα μπορεί να είναι οτιδήποτε έχει ιδιότητες ή χαρακτηριστικά (attributes) και λειτουργίες ή μεθόδους (operations). Τα αντικείμενα είναι οι οντότητες του πραγματικού κόσμου που υπάρχουν στο περιβάλλον και οι βασικές έννοιες όπως η αφαίρεση, η ενθυλάκωση, η κληρονομικότητα και ο πολυμορφισμός είναι τα βασικά χαρακτηριστικά τους.

- **Αφαίρεση (Data abstraction):** Η αφαίρεση αντιπροσωπεύει τη συμπεριφορά μιας οντότητας πραγματικού κόσμου. Κατά την εκτέλεση της εφαρμογής, κάθε αντικείμενο λαμβάνει μόνο τα μηνύματα που το αφορούν αγνοώντας οτιδήποτε άλλο συμβαίνει στο περιβάλλον του. Με τον τρόπο αυτό αποκτάται πλήρης έλεγχος της εφαρμογής αφού κάθε τμήμα του προγράμματος επηρεάζει μόνο τα αντικείμενα που το αφορούν.
- **Ενθυλάκωση (Encapsulation):** Η ενθυλάκωση είναι ο μηχανισμός σύνδεσης των δεδομένων μεταξύ τους και απόκρυψής τους από τον έξω κόσμο. Με την ενθυλάκωση ο χειρισμός κάθε αντικειμένου είναι εύχρηστος και ασφαλής, γιατί τα περιεχόμενά του προστατεύονται μέσα στο εσωτερικό του θυλάκιο (σακουλάκι) και μεταβάλλονται μόνο με τη δράση των μεθόδων που υποστηρίζει.
- **Κληρονομικότητα (Inheritance):** Η κληρονομικότητα είναι ο μηχανισμός δημιουργίας νέων κλάσεων από υπάρχουσες. Αν μια κλάση περιγραφεί, είναι δυνατόν στη συνέχεια να οριστούν μέσω αυτής, νέες κλάσεις (υποκλάσεις – subclasses), οι οποίες θα φέρουν όλα τα χαρακτηριστικά και μεθόδους της υπερκλάσης τους και κάποια καινούρια που θα τις διακρίνουν. Έτσι, χαρακτηριστικά και μέθοδοι μπορούν να είναι κοινά σε διαφορετικές κλάσεις που είναι ιεραρχικά συνδεδεμένες, από προγόνους σε απογόνους.

- **Πολυμορφισμός (Polymorphism):** Ορίζει το μηχανισμό κατά τον οποίο ένα αντικείμενο μπορεί να υπάρχει σε διαφορετικές μορφές, δηλαδή στο επίπεδο εκτέλεσης της εφαρμογής να αναφέρεται σε διαφορετικές κλάσεις. Ο πολυμορφισμός επιτρέπει να δοθεί μια κοινή συμπεριφορά σε κάθε κλάση, η οποία όμως υλοποιείται διαφορετικά για αντικείμενα διαφορετικών κλάσεων απογόνων. Ακόμα, μπορούν να οριστούν πλήρως αφηρημένες κλάσεις, με υποτιθέμενη συμπεριφορά η οποία υλοποιείται σε χαμηλότερες κλάσεις ανάλογα με τις ανάγκες του σχεδιασμού. Με τη χρήση του πολυμορφισμού ο προγραμματιστής απαλλάσσεται από τη σύνταξη και την επανάληψη πολύπλοκων δομών ελέγχου μέσα στην εφαρμογή [33].

## 2.4 Αντικειμενοστραφής Ανάλυση και Σχεδιασμός

Στον αντικειμενοστραφή προγραμματισμό, απαιτείται η ανάλυση και ο σχεδιασμός του κάθε συστήματος. Η αντικειμενοστραφής ανάλυση μπορεί να οριστεί ως έρευνα ή πιο συγκεκριμένα ως διερεύνηση αντικειμένων, ενώ ο σχεδιασμός σημαίνει συνεργασία αντικειμένων που έχουν ταυτοποιηθεί. Ο πιο σημαντικός σκοπός της ανάλυσης είναι να προσδιορίσει τα αντικείμενα του συστήματος που πρόκειται να σχεδιαστεί. Η μοντελοποίηση ενός συστήματος είναι δυνατή μόνο όταν αυτό είναι σε θέση να περιγραφεί με αντικείμενα. Ο προγραμματιστής πρέπει να αρχίσει να σκέφτεται με τρόπο που να αναγνωρίζει αντικείμενα στο σύστημα. Μετά την αναγνώριση των αντικειμένων, εντοπίζονται οι σχέσεις τους και τελικά παράγεται το σχέδιο. Ο σκοπός της ανάλυσης και του αντικειμενοστραφούς σχεδιασμού μπορεί να περιγραφεί ως:

- Αναγνώριση των αντικειμένων ενός συστήματος.
- Αναγνώριση των σχέσεών τους.
- Δημιουργία σχεδίου, το οποίο μπορεί να μετατραπεί σε εκτελέσιμο χρησιμοποιώντας αντικειμενοστραφείς γλώσσες προγραμματισμού.

### **Ανάλυση → Σχεδιασμός → Υλοποίηση με χρήση αντικειμενοστραφών γλωσσών**

Κατά τη διάρκεια της ανάλυσης, ο πιο σημαντικός σκοπός είναι η αναγνώριση αντικειμένων και η περιγραφή τους με τον κατάλληλο τρόπο. Εάν τα αντικείμενα που περιγράφουν ή αναπαριστούν το σύστημα που πρέπει να κωδικοποιηθεί, αναγνωρίζονται αποτελεσματικά, η δουλειά του σχεδιασμού είναι εύκολη.



Συνδυάζονται, στη συνέχεια, με μεθόδους, δηλαδή λειτουργίες που πρέπει να εκτελούν και όταν μπορέσουν οι μέθοδοι αυτές να συνεργαστούν, ο σκοπός του συστήματος εκπληρώνεται.

Η δεύτερη φάση είναι ο σχεδιασμός. Κατά τη φάση αυτή δίνεται έμφαση στις απαιτήσεις και στην εκπλήρωσή τους. Σε αυτό το στάδιο, τα αντικείμενα συνεργάζονται σύμφωνα με τον προβλεπόμενο συσχετισμό τους. Ο σχεδιασμός, λοιπόν, περιλαμβάνει τη διερεύνηση ή τον ορισμό αυτών των συσχετίσεων. Αφού ολοκληρωθεί ο συσχετισμός, ολοκληρώνεται και ο σχεδιασμός.

Η τρίτη φάση είναι η υλοποίηση. Σε αυτή τη φάση, η σχεδίαση υλοποιείται χρησιμοποιώντας αντικειμενοστραφείς γλώσσες όπως Java, C++, C#, Python και άλλες σύγχρονες γλώσσες προγραμματισμού.

## 2.5 Σύγχρονα Προγραμματιστικά Περιβάλλοντα

Η ανάπτυξη του υλικού των υπολογιστών επέτρεψε την ανάπτυξη των γλωσσών προγραμματισμού, που οδήγησαν σε νέες τεχνικές σχεδίασης προγραμμάτων και τελικά στην ανάπτυξη εφαρμογών με μεγαλύτερες απαιτήσεις, συνθετότερες και χρησιμοποιούμενες από πολύ μεγαλύτερη μερίδα χρηστών. Αποτέλεσμα αυτών είναι η έμφαση που δίνεται πια στον τρόπο επικοινωνίας του χρήστη με την εφαρμογή: τη λεγόμενη διεπαφή χρήστη (User Interface). Ενώ λοιπόν, παλαιότερα, ένας προγραμματιστής ασχολούνταν λίγο έως καθόλου με το interface της εφαρμογής του, τώρα πια είναι επιτακτική η ανάγκη να αντιλαμβάνεται την εφαρμογή από τη μεριά του χρήστη.

Όλα τα σύγχρονα προγραμματιστικά εργαλεία, όπως η Visual C++, η Delphi ή Visual Basic προσφέρουν δυνατότητες εμπλουτισμού των εφαρμογών με φιλικότητα και ευελιξία στον τρόπο επικοινωνίας τους με το χρήστη. Αυτό γίνεται μέσα από τη χρήση γραφικών. Παρέχοντας ένα εύχρηστο περιβάλλον εργασίας, αυξάνονται οι ικανότητες των χρηστών και μεγιστοποιούνται τα αποτελέσματα, μειώνοντας το χρόνο και το κόστος εκμάθησής τους σε μια καινούργια εφαρμογή. Με το γραφικό προγραμματιστικό περιβάλλον, αφ' ενός παρουσιάζεται στους χρήστες μια οικεία εφαρμογή στην οποία μπορούν να χρησιμοποιήσουν πλήκτρα εντολής ή επιλογής, λίστες, πλαίσια ελέγχου, κλπ που είναι πλέον κοινότυπα, αφ' ετέρου ο προγραμματιστής διευκολύνεται με έτοιμες εργαλειοθήκες, εργαλεία εκσφαλμάτωσης,

εύχρηστα παράθυρα για να γράψει τον κώδικά του, αυτόματους οδηγούς εκτέλεσης εργασιών (wizards), ακόμα και τη δυνατότητα να εκμεταλλευτεί ήδη υπάρχουσες εφαρμογές μέσα στη δική του εφαρμογή, κερδίζοντας χρόνο, κόστος και ομοιογένεια. Τα σύγχρονα προγραμματιστικά περιβάλλοντα παρέχουν αυτούς τους μηχανισμούς.

Σε ένα σύγχρονο προγραμματιστικό περιβάλλον όπως είναι και αυτά του αντικειμενοστραφούς προγραμματισμού, προσφέρονται νέοι τρόποι υλοποίησης εφαρμογών, καινούρια εργαλεία και ενσωματωμένες τεχνικές καθώς και αυξημένες ευκαιρίες χρήσης γραφικού περιβάλλοντος εργασίας και ανάπτυξης. Αυτό, βέβαια, δε μειώνει την ανάγκη για χρήση των αρχών του τμηματικού και δομημένου προγραμματισμού, κατάλληλα προσαρμοσμένων.

## 2.6 Πλεονεκτήματα του αντικειμενοστραφούς προγραμματισμού

Περιεκτικά και σύντομα, τα πλεονεκτήματα του αντικειμενοστραφούς προγραμματισμού είναι:

- πιο κατανοητός κώδικας, αφού ο αντικειμενοστραφής κώδικας προσπαθεί να μοντελοποιήσει τον πραγματικό κόσμο,
- ευκολότερη επαναχρησιμοποίηση και συνδυασμό δομικών κομματιών που δημιουργούνται με αυτόν, λόγω αυτονομίας,
- ευκολότερη συντήρηση κώδικα, λόγω ενθυλάκωσης
- δυνατότητα δημιουργίας ευχάριστων κι εύχρηστων διεπαφών χρήστη

### 3. ΗΣΙΟΔΟΣ: το εκπαιδευτικό παιχνίδι

---

Ο εκσυγχρονισμός της εκπαίδευσης είναι μια καίρια απαίτηση. Η εισαγωγή της επιστήμης της Πληροφορικής στη ζωή και την εκπαίδευση είναι αληθής και δεδομένη. Η παιχνιδοκεντρική μάθηση μπορεί να συνδυάσει τις δύο αυτές πραγματικότητες με τον καλύτερο τρόπο. Το θέμα είναι πως η θεωρία από την πράξη, πάντα απέχει...

Ακούμε, συνεχώς για την **παιχνιδοκεντρική μάθηση** και τη **διαθεματική διδασκαλία**, αλλά τα περισσότερα προσφερόμενα εργαλεία είναι μάλλον ανεπαρκή και αδιάφορα –ιδίως στα μάτια των μαθητών που γεννιούνται μέσα στην τεχνολογία– αφήνοντας τον κάθε εκπαιδευτικό να κάνει ό,τι μπορεί ο ίδιος και μόνος...

Όσα μητέρα πέντε παιδιών και εκπαιδευτικός ΠΕ03 και ΠΕ86, γεγονός που μου επιτρέπει να εργάζομαι στην Πρωτοβάθμια και τη Δευτεροβάθμια εκπαίδευση και αποκτώντας σχετικές γνώσεις από τα μαθήματα του μεταπτυχιακού προγράμματος που παρακολούθησα, αποφάσισα να κάνω κάτι που δεν υπάρχει... Αποφάσισα να κάνω ένα διαδραστικό παιχνίδι για την Ιστορία της Γ' Δημοτικού [35] που καλύπτει όλη την ύλη της. Είναι υποστηρικτικό και μοιάζει με γνώριμα παιχνίδια των μαθητών.

Το ονόμασα «ΗΣΙΟΔΟΣ». Ο Ησίοδος ήταν αρχαίος Έλληνας ποιητής, ραψωδός, συγγραφέας και μυθογράφος [36]. Υπήρξε ο δεύτερος σπουδαιότερος ποιητής μετά τον Όμηρο. Ένα από τα τρία σημαντικότερα έργα του, που σώθηκαν, είναι η Θεογονία. Η Θεογονία είναι έπος, αποτελούμενο από 1.022 στίχους. Αναφέρεται στις τοπικές λατρείες των Θεών και στην απαρχή του κόσμου, δηλαδή το πρώτο κεφάλαιο της Ιστορίας που πραγματεύεται το παιχνίδι. Είναι μια σημαντικότερη πηγή πληροφοριών για την ελληνική μυθολογία. Αυτός ήταν και ο λόγος που επέλεξα το όνομά του.



Εικόνα 4: Ησίοδος - Το εικονίδιο της Εφαρμογής

Πηγή: <https://www.greek-language.gr/digitalResources/files/image/library/authors/156.jpg>

**...και εγένετο Ησίοδος!**

### 3.1 Ανάγκη Ύπαρξης

Η ανάγκη ανάπτυξης ενός εκπαιδευτικού λογισμικού είναι μάλλον υποκειμενική υπόθεση. Θα μπορούσε να ισχυριστεί κανείς πως δεν υπάρχει καν. Ωστόσο υπάρχουν εκπαιδευτικοί που επιθυμούν να δουλέψουν με πιο σύγχρονους και δημιουργικούς τρόπους, που θέλουν να εφαρμόσουν σύγχρονες θεωρίες μάθησης και διαθεματικές τακτικές και δεν δύνανται λόγω έλλειψης υποδομών. Για να χρησιμοποιήσει ένας δάσκαλος κάτι πιο σύγχρονο, θα πρέπει να είναι ο ίδιος προγραμματιστής ή να έχει ασχοληθεί ακούραστα πολλές ώρες μαθαίνοντας εργαλεία που θα τον βοηθήσουν να εμπλουτίσει τον τρόπο διδασκαλίας του. Πολλά από αυτά έχουν οδηγήσει σε αρκετά όμορφες προσπάθειες, που περιορίζονται πάντα στη χρήση τους από τον δημιουργό τους. Άνθρωποι με όρεξη, ανεξαρτήτως προγραμματιστικών γνώσεων, θα έπρεπε να έχουν την ευχέρεια να ενεργοποιήσουν τους μαθητές τους και να προκαλέσουν το ενδιαφέρον τους, με κάτι οικείο, αλλά διαφορετικό.

Ένα δεύτερο ζήτημα που κάνει επιτακτική την ανάγκη ύπαρξης σοβαρών προσπαθειών ανάπτυξης εκπαιδευτικών λογισμικών είναι η ιλιγγιώδης εξέλιξη της τεχνολογίας. Όταν γίνονται επίσημα κάποιες δειλές, τεχνολογικές προσπάθειες, τα εμπορικά παιχνίδια είναι τόσο πιο προχωρημένα, με εξαιρετικά γραφικά, ήχους και σενάρια που κάνουν τα εκπαιδευτικά να ωχριούν και να δέχονται ειρωνεία και απαξίωση. Η εκπαίδευση δεν πρέπει να αγωνίζεται στα μετόπισθεν· μάλλον οφείλει να είναι στην πρώτη γραμμή, προκειμένου να κερδίσει τους νέους!

#### 3.1.1 Σενάριο

Η Ιστορία της Γ' Δημοτικού είναι το πρώτο μάθημα διαφορετικού ύφους από τη Γλώσσα και τα Μαθηματικά, με το οποίο έρχονται σε επαφή οι μικροί μαθητές, μετά την εισαγωγή τους στην Πρωτοβάθμια εκπαίδευση. Μπορεί να θεωρηθεί ως εισαγωγή στα αντικείμενα που «μαθαίνεις». Αρέσει στα περισσότερα παιδάκια και έχει το μυστήριό της αφού εμπλέκει τους μύθους και την πραγματικότητα. Περνώντας η ύλη από τη μυθολογία στην ιστορία, βοηθάει και τους μικρούς μαθητές να περάσουν από τα παραμύθια στη λήψη νέας γνώσης. Πόσο ωραία θα ήταν να γίνει αυτό ανυπομονώντας οι μικροί μαθητές να παίξουν ένα παιχνιδάκι που θα τους βοηθούσε στο νέο αυτό μάθημα!

Νιώθω ότι είναι απαραίτητο να έχουν τα μικρά παιδιά ένα εργαλείο ευχάριστο κι ενδιαφέρον για να ελέγξουν τις νεοαποκτηθείσες γνώσεις τους και να δουν σε ποιο

επίπεδο βρίσκονται ώστε να προσπαθήσουν όσο περισσότερο μπορούν· και νομίζω πως μπορώ να τους το δώσω.

Η διδακτική σχεδίαση του λογισμικού μου είναι Παιδαγωγική, δηλαδή έχει τις βάσεις της στη μάθηση μέσω παιχνιδιών, αλλά στην πραγματικότητα αυτό που με ενδιαφέρει περισσότερο είναι να φτιάξω ένα λογισμικό που να είναι ελκυστικό στο χρήστη, διαδραστικό αλλά απλό, χωρίς ανάγκη εκπαίδευσης χρήσης. Να μπορεί να το ανοίξει ο μαθητής και να το χρησιμοποιήσει άμεσα. Να δελεάσω το υποψήφιο κοινό χωρίς να αισθάνεται ότι προσπαθώ να το διδάξω. Γι' αυτό προσπάθησα να χρησιμοποιήσω χαρακτηριστικά με τα οποία είναι εξοικειωμένοι οι μικροί μαθητές και αντιλαμβάνονται εύκολα. Επιπρόσθετα το εμπλούτισα με κινήσεις (animations) και γραφικά που να είναι ανταγωνιστικά με τα εμπορικά ηλεκτρονικά παιχνίδια.

Η αξιολόγηση που προσφέρει το παιχνίδι μου είναι μάλλον ποσοτική αφού εμπλέκεται το «σκορ» στη μαθησιακή διαδικασία, αλλά έχει και ποιοτικές προεκτάσεις λόγω της ανατροφοδότησης, της ενθάρρυνσης και της επιβράβευσης κατά τη διάρκειά του. Την τελική αξιολόγηση θα κάνει μόνος ο μαθητής, ελέγχοντας προσωπικά πόσο ευχαριστημένος είναι, ενώ η ύπαρξη της βαθμολογίας λειτουργεί περισσότερο ως κίνητρο επανάληψης και μεγαλύτερης προσπάθειας.

Το λογισμικό, εκτός από τη χρήση του ανά κεφάλαιο και κατά τη διάρκεια της σχολικής-διδακτικής χρονιάς μπορεί να λειτουργήσει και ως επαναληπτικό. Δεν είναι αναγκαίο, λοιπόν, να έχει αυστηρά καθορισμένη δομή. Ο μαθητής-χρήστης μπορεί να επιλέγει όποια ενότητα επιθυμεί, όποτε το επιθυμεί και με οποιαδήποτε σειρά, ενώ για τον ίδιο ακριβώς λόγο μπορεί να επαναλάβει κάθε επίπεδο όσες φορές θέλει.

Σκοπός του παιχνιδιού μου, οπωσδήποτε δεν είναι να τιμωρηθεί ο χρήστης αν αστοχήσει, αλλά να του δοθεί η ευκαιρία να προσπαθήσει ήρεμα κι άλλο, μέχρι να καταφέρει να καλύψει τυχόντα μαθησιακά κενά.

### **3.1.2 Τελικοί Χρήστες**

Όπως ήδη έχει αναφερθεί, τελικοί χρήστες της εφαρμογής θα είναι οι μαθητές της Γ' τάξης του Ελληνικού Δημοτικού σχολείου. Αυτό δεν αποκλείει τη χρήση του από μεγαλύτερα παιδιά του Δημοτικού που έχουν διδαχτεί τη συγκεκριμένη ύλη. Για την ακρίβεια θα μπορούσε να χρησιμοποιηθεί ανεξαρτήτως ηλικίας από όποιον του αρέσει η μυθολογία. Θεωρώ πως είναι πλήρες και ευχάριστο στη χρήση και ενδιαφέρον στην ύλη που περιλαμβάνει.

Θα μπορούσε να χρησιμοποιηθεί από τους ίδιους τους μαθητές και να προταθεί από τους δασκάλους ή τους γονείς τους ή να χρησιμοποιηθεί διαθεματικά σε συνδυασμό με το μάθημα της πληροφορικής. Θεωρώ πως θα μπορούσε να βοηθήσει τους μικρούς μαθητές ακόμα και στη λεπτή κινητικότητα των χεριών τους καθώς και στη χρήση του ηλεκτρονικού υπολογιστή. Μπορεί να αποτελέσει μια ευχάριστη εξάσκηση-διάλειμμα από την «αφή» την οποία διδάσκονται κατά κόρον από τις κινητές συσκευές.

### 3.1.3 Χρονοπρογραμματισμός και Δυσκολίες

Για να ολοκληρωθεί ένα εκπαιδευτικό λογισμικό (όπως και στη γενική περίπτωση ηλεκτρονικών εφαρμογών) θα πρέπει να περάσει από διάφορες φάσεις και να ολοκληρωθούν στάδια, τα οποία πρέπει να διεκπεραιωθούν από την ομάδα έργου. Μέλη της ομάδας έργου, ανάλογα με την ειδικότητά τους αναλαμβάνουν τις υπευθυνότητες που πρέπει να ολοκληρώσουν. Έτσι σε μια πλήρη ομάδα υπάρχουν εκτός από τους γενικούς αναλυτές, σχεδιαστές, γραφίστες, προγραμματιστές και ειδικούς του αντικειμένου που πραγματεύεται το λογισμικό:

- Αναλυτές Συστημάτων,
- Μηχανικοί Εφαρμογών εξειδικευμένοι στην παραμετροποίηση και διαχείριση του λογισμικού υποδομής με γνώσεις στην ανάπτυξη διαδικτυακών εφαρμογών και υπηρεσιών για την ανάπτυξη των εφαρμογών,
- Υπεύθυνος ομάδας των ανωτέρω μηχανικών,
- Μηχανικός Βάσεων Δεδομένων (Database Engineer) για την παραμετροποίηση και διαχείριση της βάσης δεδομένων και τη βελτιστοποίηση των επιδόσεων,
- Σχεδιαστής/Προγραμματιστής Ιστοσελίδων (Web Programmer/Designer),
- Γραφίστας Διαδικτυακών Διεπαφών (Web Graphic Designer),
- Φιλολογοί, Κοινωνιολόγοι, Ψυχολόγοι,

για να μην αναφέρω το εξωτερικό προσωπικό που πρέπει να προσληφθεί, όπως νομικοί, οικονομολόγοι και υπεύθυνοι marketing και επικοινωνίας. Στην περίπτωση του Ησίοδου, η ομάδα αυτή... είμαι εγώ.

Έκανα τον χρονοπρογραμματισμό του λογισμικού μου, όπως φαίνεται στο διάγραμμα Gantt [37] που ακολουθεί, τον οποίο και ακολούθησα με ευλάβεια. Λόγω της μη ύπαρξης ειδικών όλων των τομέων κάποιες εργασίες έγιναν συνοπτικότερα,

όπως είναι φυσιολογικό για έναν μόνο άνθρωπο και ένα περιορισμένο χρονικό διάστημα.

**Πίνακας 3: Χρονοπρογραμματισμός - Διάγραμμα Gantt**

	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12
Διατύπωση Προβλήματος												
Ανάλυση Αναγκών												
Σχεδίαση Προδιαγραφών												
Παραγωγή												
Δοκιμές Alpha & Beta												
Αξιολόγηση & Ανατροφοδότηση												
Έλεγχος Πορείας Ποιότητας Έργου												

Στο διάγραμμα, στην αριστερή στήλη φαίνονται οι εργασίες που έλαβαν χώρα προκειμένου να υλοποιηθεί το εκπαιδευτικό λογισμικό και στο κέντρο, η σειρά με την οποία αυτές εξελίχθηκαν (όπως είχα προϋπολογίσει) κατά τις αντίστοιχες εβδομάδες, που φαίνονται στις στήλες.

Η Διατύπωση του Προβλήματος, όπως και η Ανάλυση των Αναγκών έγιναν ταυτόχρονα, αφού ουσιαστικά δεν υπήρχε η δυνατότητα για αναλυτικές μεθόδους συλλογής πληροφοριών. Βασίστηκα ουσιαστικά στην παρατήρηση των μαθητών μου των αντίστοιχων ηλικιών, όχι μέσω βιντεοσκόπησης, αλλά μέσω προσωπικής εμπειρίας και καθημερινής επικοινωνίας και τριβής μαζί τους.

Συζητήσεις με συναδέλφους που θα μπορούσαν να κατέχουν θέση ημιδομημένων συνεντεύξεων, σχετικά με τα ηλεκτρονικά εργαλεία που παρέχονται σε εκπαιδευτικούς και μαθητές και τις σύγχρονες απαιτήσεις στον τρόπο διδασκαλίας, αλλά και τις πραγματικές ανάγκες των μαθητών μας, αποτέλεσαν, επίσης, πηγή της ανάλυσης των αναγκών, που ενυπήρχε μέσα μου και καταδεικνύει τα εξής:

- **Ανάλυση Αναγκών (Needs Analysis):** Η αρχική κατάσταση γνώσεων των τελικών χρηστών είναι ασαφής. Αυτός είναι και ο σκοπός του λογισμικού: να βοηθήσει στη μαθησιακή διαδικασία και να τους οδηγήσει να εντοπίσουν το επίπεδό τους. Τυπικά η γνώση τους είναι η παρεχόμενη από το Αναλυτικό

Πρόγραμμα Σπουδών για τη Γ' τάξη του Δημοτικού, ενώ η επιθυμητή κατάσταση γνώσεων μετά τη χρήση του λογισμικού είναι η ουσιαστική κατοχή της τυπικά παρεχόμενης γνώσης ή ο εντοπισμός κενών και αδυναμιών.

- **Ανάλυση Χρηστών (User analysis):** Ο κύριος τελικός χρήστης είναι ο μαθητής της Γ' Δημοτικού (το λογισμικό ως επαναληπτικό, υποστηρικτικό και διασκεδαστικό, δεν αποκλείει τη χρήση του από όποιον άλλο επιθυμεί). Μέριμνα για ιδιαίτερες ομάδες χρηστών, όπως δυσλεκτικοί και μαθητές με προβλήματα όρασης ή ακοής δεν έχει ληφθεί ακριβώς, αλλά υπάρχουν οι υποδομές για να υποστηριχθούν οι περισσότερες από τις παραπάνω ομάδες.
- **Ανάλυση Τεχνολογίας (Technology analysis):** Τεχνολογική απαίτηση από το παρόν εκπαιδευτικό λογισμικό είναι η ύπαρξη ενός υπολογιστή (με επεξεργαστή τουλάχιστον i3 και 4MB μνήμη), σταθερού ή φορητού (με ηχεία), μια απλή σύνδεση Internet και ένας φυλλομετρητής που να υποστηρίζει WebGL. Μια τυπική γνώση, από την πλευρά των μαθητών, σχετικά με τη χρήση του υπολογιστή είναι επιθυμητή.
- **Ανάλυση συνθηκών (Situational analysis):** Δεν υπάρχουν ιδιαίτερες συνθήκες που πρέπει να επικρατούν στους χώρους χρήσης του λογισμικού.

Όσον αφορά στη Σχεδίαση των Προδιαγραφών, αυτή σχετίζεται με τις αποφάσεις σχεδίασης του λογισμικού με βάση την ανάλυση και αναφέρονται αναλυτικά σε επόμενο υποκεφάλαιο. Όπως φαίνεται και στον Πίνακα 3, οι σχεδιαστικές επιλογές συνεχίστηκαν παράλληλα με τον προγραμματισμό για αρκετό διάστημα μέχρι να κατασταλάξω, ιδίως, στις παιδαγωγικές μου επιλογές.

Η Παραγωγή: προγραμματισμός, παραγωγή μέσων, κείμενα, γραφικά, κινήσεις και η εκμάθηση της πλατφόρμας Unity από το μηδέν, φυσικά κατέλαβε το μεγαλύτερο χρονικό διάστημα.

Προς την ολοκλήρωση της παραγωγής, ξεκίνησε η πιλοτική εφαρμογή με δοκιμές Alpha και Beta και ταυτόχρονη Αξιολόγηση του λογισμικού. Όπου ήταν δυνατόν έγινε αναθεώρηση κάποιων στοιχείων κατά τα προστάγματά της.

Τώρα μένει ενεργός ο έλεγχος της πορείας αλλά και της ποιότητας του έργου.

Όπως φάνηκε και από τις προηγούμενες αναφορές, είναι εύλογες και ευκόλως κατανοητές οι δυσκολίες που συνάντησα σε όλο το πόνημά μου:



- ο ρόλος μιας ολόκληρης ομάδας που έπαιξα
- οι αποφάσεις που έπρεπε να πάρω μόνη μου
- η παντελής άγνοια της πλατφόρμας που θα χρησιμοποιούσα
- ο περιορισμένος χρόνος για ένα πλήρες έργο

Ωστόσο θεωρώ πως ανταπεξήλθα κι αυτό γιατί είχα στόχο!

### 3.1.4 Εκπαιδευτικοί Στόχοι

Οι εκπαιδευτικοί στόχοι του παιχνιδιού Ησίοδος είναι να αγαπήσουν οι μικροί μαθητές το μάθημα της Ιστορίας, με το οποίο έρχονται αντιμέτωποι για πρώτη φορά, να διατηρήσουν το ενδιαφέρον τους κατά τη διάρκεια διδασκαλίας των διαφόρων κεφαλαίων, να συνδυάσουν την απόκτηση γνώσης με τις ικανότητές τους στην Πληροφορική και γενικότερα... **να μαθαίνουν παίζοντας!** Ειδικότερα θα ανέμενα:

#### σε επίπεδο γνώσεων:

να αναγνωρίζουν τη διαφορά της Μυθολογίας από την Προϊστορία, να συνοψίζουν τα επιμέρους κεφάλαια των ενοτήτων με περιληπτικές αναφορές, να συγκρατούν τα ονόματα, τις ιδιότητες και τα σύμβολα θεών και ηρώων, να ονομάζουν, να περιγράφουν και να καταλαβαίνουν τους πολιτισμούς που αναπτύχθηκαν στον ελλαδικό χώρο

#### σε επίπεδο δεξιοτήτων:

να χειρίζονται έναν ηλεκτρονικό υπολογιστή (την απλή λειτουργία του), να χρησιμοποιούν το εκπαιδευτικό παιχνίδι, να αλλάζουν επιλογές και επίπεδα σε αυτό, να εξοικειωθούν με τη χρήση εκπαιδευτικών λογισμικών

#### σε επίπεδο στάσεων:

να αποδέχονται την παιχνιδιοκεντρική μάθηση, να εκτιμούν τη δυνατότητα διασκεδαστικής μάθησης, να προσαρμόζονται στις επιταγές της σύγχρονης ζωής και εκπαίδευσης, να υιοθετούν, προτιμούν και υποστηρίζουν τη χρήση εκπαιδευτικών λογισμικών.

## 3.2 Περιγραφή Εφαρμογής

Στο παρόν υποκεφάλαιο θα γνωρίσουμε τον Ησίοδο. Από την αρχή μέχρι το τέλος του, από τα πρώτα μέχρι τα τελευταία διλήμματά του και τις αποφάσεις που έπρεπε να ληφθούν από τη δημιουργό του.

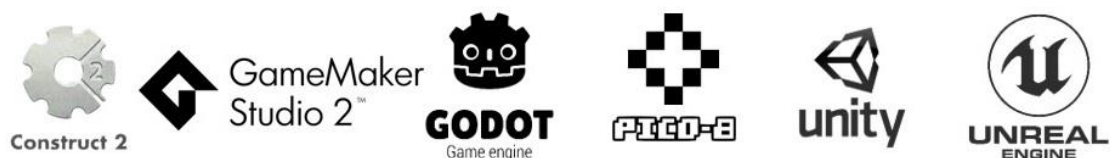
### 3.2.1 Προγραμματιστικές Επιλογές

Ο Ησίοδος είναι ένα εκπαιδευτικό λογισμικό και σαν τέτοιο θα έπρεπε να αναπτυχθεί με κάποιο προγραμματιστικό εργαλείο. Έπρεπε να αποφασίσω σε ποια γλώσσα θα γράψω το πρόγραμμά μου ή σε ποια πλατφόρμα θα το δουλέψω. Η ανάπτυξη παιχνιδιών έχει προχωρήσει πάρα πολύ και η ύπαρξη παιχνιδομηχανών και πλατφορμών ανάπτυξης παιχνιδιών είναι γεγονός.

#### Τι είναι μια παιχνιδομηχανή

Στην πραγματικότητα, τι είναι μια παιχνιδομηχανή; Είναι ένα μεγάλο, έτοιμο τμήμα λογισμικού που επιτρέπει την ανάπτυξη παιχνιδιών χωρίς ο δημιουργός τους να ασχοληθεί με το «βαρετό» κομμάτι αυτής της διαδικασίας. Δεν θα ανησυχήσει, ούτε θα δυσκολευτεί για το πώς θα αποδοθούν τα γραφικά στην οθόνη ή πώς θα συνδεθεί το σύστημα με τα χειριστήρια, πώς θα πάρει θέσεις της οθόνης ή πώς θα φερθεί ένα αντικείμενό του ως φυσικό σώμα: γενικότερα θα μπορεί να αφοσιωθεί στον καθαρό προγραμματισμό. Ένα άλλο πλεονέκτημα των παιχνιδομηχανών είναι ότι διαθέτουν και παρέχουν εύχρηστα εργαλεία για να κάνει κανείς πράγματα που είναι επιθυμητά σ' ένα παιχνίδι, όπως είναι οι κινήσεις (animations) και η διεπαφή χρήστη (user interface): είναι ειδικά σχεδιασμένες γι' αυτό το σκοπό.

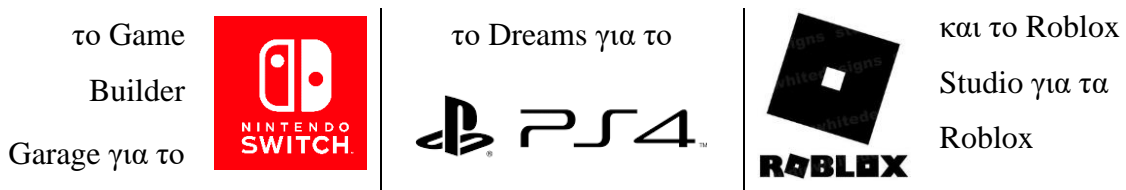
Υπάρχουν πολλές τέτοιες μηχανές (game engines). Οι γνωστότερες από αυτές, αλφαβητικά και με τα λογότυπά τους είναι οι:



ενώ υπάρχουν παιχνιδομηχανές για συγκεκριμένου τύπου παιχνίδια, όπως οι:



Υπάρχουν ακόμα και πλατφόρμες ανάπτυξης παιχνιδιών για συγκεκριμένες κονσόλες βιντεοπαιχνιδιών, όπως:



Κάθε μία από τις παραπάνω παιχνιδομηχανές έχει πλεονεκτήματα και μειονεκτήματα καθώς και φανατικούς οπαδούς και πολέμιους. Κι εγώ έπρεπε να διαλέξω μία –χωρίς να γνωρίζω καμία!

## Unity VS Unreal Engine

Πίνακας 4: Unity VS Unreal Engine - Συγκριτική Αναφορά

	Unity	Unreal Engine
Γλώσσα Προγραμματισμού	C#	C++
Visual Scripting	νέα στο visual scripting - διαθέτει τον playmaker επί πληρωμή	εξελιγμένο, ευέλικτο και δωρεάν visual scripting, το blueprints
Πηγαίος Κώδικας	απαγόρευση στον πηγαίο κώδικα (source code)	διαθέσιμος σε όλους (open source)
Γραφικά	σχετικά φτωχά – δύσκολη διαχείριση του φωτός	εξαιρετικά, με τέλεια αισθητική και ρεαλιστικό χειρισμό φωτός
Απόδοση	με προβλήματα	άριστη
Build Time	γρήγορο	πάρα πολύ αργό
Υποστήριξη 2D Game Development	αντίστοιχη με τη 3D	εγκαταλελειμμένη υποστήριξη
Υποστήριξη όλων των Πλατφορμών	ναι	ναι, ασαφής η υποστήριξη για HTML πλατφόρμα
Κινητές Συσκευές	φτωχότερη απόδοση, μεγάλη κατανάλωση μπαταρίας σε 3D	σταθερή απόδοση όμοια με αυτή του σταθερού υπολογιστή
Assets	τεράστια συλλογή, όλων των ειδών	περιορισμένες επιλογές
Marketplace	τεράστιο, με χαμηλές τιμές για όλα τα είδη	μικρό, με υψηλές τιμές και λίγες επιλογές
Παιχνίδια	όλων των ειδών, κάθε ένα διαφορετικό	κυρίως 1 <sup>st</sup> /3 <sup>rd</sup> person Shooters games, παρόμοια εμφάνιση
Κοινότητα και Υποστήριξη	τεράστια	μέτρια
Ερωτήσεις / Απορίες	σχεδόν όλες οι ερωτήσεις έχουν απαντηθεί	πολλές φορές μπορεί να υπάρξει δυσκολία στην εύρεση απάντησης
Βοηθητικά Βίντεο	αμέτρητα: καλά και κακά	καλά και κακά
Κοινότητα Προγραμματιστών	η μεγαλύτερη	μικρή

Οι Unity και Unreal Engine είναι μάλλον οι πιο γνωστές παιχνιδομηχανές. Έψαξα, διάβασα, έλεγξα... το σίγουρο ήταν πως έπρεπε να διαλέξω την καλύτερη για το σκοπό που ήθελα να εξυπηρετήσει, για το παιχνίδι που ήθελα να προγραμματίσω εγώ. Μια σωστή οδηγία φίλων από το YouTube [38] ήταν να κοιτάξω τα παιχνίδια που φτιάχνονται και στις δύο για να επιλέξω... και ναι, το παιχνίδι μου δεν θα ήταν «πολεμικό». Με βοήθεια από βίντεο και ηλεκτρονικές πηγές [39] συνέταξα μια συγκριτική αναφορά για τις δυο πιο δημοφιλείς παιχνιδομηχανές, που φαίνεται στον Πίνακα 4.

Ο καθένας ανάλογα με τα χαρακτηριστικά που επιθυμεί, κάνει την επιλογή του. Εγώ διάλεξα τη Unity!

### Γιατί Unity;

Παιχνιδομηχανή δεν γνώριζα καμία· ούτε ήξερα τι είναι... Όποια επέλεγα θα τη μάθαινα από την αρχή. Με βάση τις συγκρίσεις που παρακολούθησα και τις ίδιες τις οδηγίες/πληροφορίες μέσα από το Unity Hub και την ιστοσελίδα της Unity [40] που διδάσκει την παιχνιδομηχανή, θεώρησα ότι αυτή κυμαίνεται σε παρόμοια επίπεδα σε ό,τι αφορά στις δυνατότητες που έχει και στην ευκολία εκμάθησης. Μου άρεσε που υπάρχει η δυνατότητα μεταπήδησης από 2D σε 3D άμεσα, το θεώρησα ευελιξία. Καλύφθηκα από το γεγονός ότι υποστηρίζει όλες τις πλατφόρμες και η περιορισμένη απόδοση στις κινητές συσκευές δεν με αφορούσε, αφού το παιχνίδι μου ήθελα να ξεκολλήσει τους μαθητές από αυτές και να τους ελκύσει στον υπολογιστή. Τελευταίος λόγος, αλλά όχι επουσιώδους σημασίας, ήταν η τεράστια δημοφιλία της· αυτή με προσέλκυσε και με κέρδισε.

Η Unity είναι εξαιρετικά δημοφιλής. Όχι μόνο γιατί έχουν δημιουργηθεί μ' αυτή πασίγνωστα παιχνίδια όπως τα Hollow Knight, Cuphead, Fall Guys, Overcooked, Cities: Skylines και Hearthstone (πολλά και δημοφιλή υπάρχουν και στην Unreal)



αλλά κυρίως εξαιτίας της τεράστιας κοινότητας, που είναι έτοιμη να σε βοηθήσει στην κάθε απορία και ερώτηση, από την πιο εύκολη ως την πιο εξειδικευμένη. Αν κατανοήσεις τον τρόπο με τον οποίο πρέπει να κάνεις τις ερωτήσεις, το διαδίκτυο και τα φόρουμ της Unity είναι εκεί να σε βοηθήσουν οποιαδήποτε στιγμή. Εμένα σαν αρχάρια με πλήρη άγνοια, αυτό με ενδιέφερε περισσότερο: η υποστήριξη.

## Η κοινότητα της Unity

Η άγνοιά μου ήταν τόσο μεγάλη που ακόμα και το περιβάλλον της μηχανής ήταν... ακατανόητο και τρομακτικό. Δεν ήξερα τίποτα. Ούτε πού να πατήσω, ούτε πώς να τακτοποιήσω τα παράθυρα, ούτε το σκοπό όλων αυτών των παραθύρων (Εικόνα 5)!

Και τότε, βρήκα το πρώτο βίντεο του Stefan Persson [41], με τίτλο:

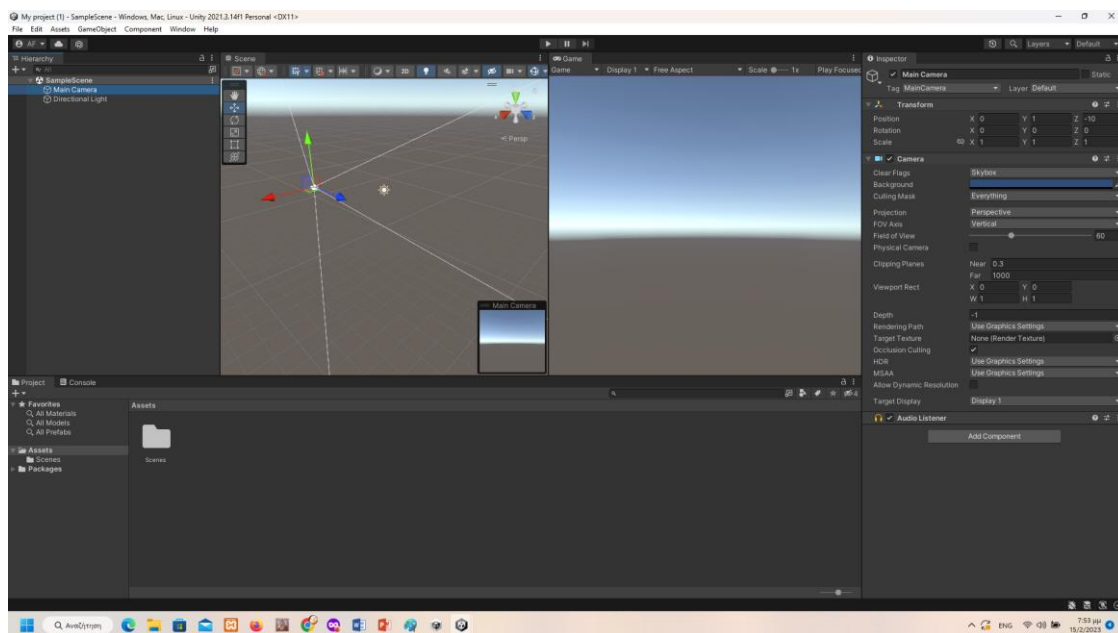
[LEARN UNITY - The Most BASIC TUTORIAL I'll Ever Make](#) και άρχισα να κατακτώ τη γνώση κομμάτι-κομμάτι... Βασικά, άρχισα να καταλαβαίνω τι γίνεται. Ένας άλλος προγραμματιστής, ο Mark Brown, πολύ ορθά, συμβούλεψε: μην προσπαθείς να τα μάθεις όλα από την αρχή! Ακολούθησε τρία βήματα:

**1<sup>ο</sup> βήμα:** Μαθαίνεις τα βασικά και τίποτα περισσότερο

**2<sup>ο</sup> βήμα:** Εξοικειώνεσαι με αυτά τα βασικά, μέσω επαναλήψεων και απλών έργων

**3<sup>ο</sup> βήμα:** Αργά, με το πέρασμα του χρόνου, χτίζεις το σύνολο των εργαλείων που χρειάζεσαι, μαθαίνοντάς τα όταν τα χρειάζεσαι.

Δυστυχώς, όμως, εγώ δεν είχα καιρό για αυτό. Έπρεπε να προλάβω να καταλάβω τι είναι η πλατφόρμα, πώς δουλεύει και ταυτόχρονα να αναπτύξω και το λογισμικό μου.



Εικόνα 5: Το περιβάλλον της Unity

Άσπογος βοηθός μου ήταν η κοινότητα της Unity. Πέρα από τα assets που μου παρείχε, με αποτέλεσμα να μη χρειαστεί να τα παράξω όλα εγώ, ακοίμητοι βοηθοί μου στάθηκαν οι:

- <https://forum.unity.com/>

- <https://answers.unity.com/index.html>
- <https://stackoverflow.com/questions>
- <https://gamedev.stackexchange.com/questions/>

και φυσικά τα manuals της Unity: : <https://docs.unity3d.com/Manual/index.html>.

Μεγάλη βοήθεια έλαβα και από σειρές βίντεο των BRACKEYS, Mark Brown, Lucas από Fire Brain Games και Mike από Comp-3 Interactive, που μπορούν να βοηθήσουν και να ξεδιαλύνουν πράγματα και καταστάσεις στο μυαλό του νέου προγραμματιστή στην πλατφόρμα.

Αρκεί να ξέρεις πώς να το ρωτήσεις και η κοινότητα Unity θα σου απαντήσει!

### 3.2.2 Τεχνικές Προδιαγραφές Μέσων

Για την παιχνιδομηχανή μου χρησιμοποίησα το Unity Hub 3.3.0 και κατέβασα την έκδοση της Unity 2021.3.14f1 LTS για Windows και WebGL. Κατέβασα και την έκδοση 2022.1.23f1 αλλά τελικά προτίμησα να εργαστώ με την **Long Term Support**, για μεγαλύτερη ασφάλεια.

Βέβαια, για να ολοκληρωθεί, οποιοδήποτε λογισμικό, δεν αρκεί μόνο μια πλατφόρμα ανάπτυξης παιχνιδιών. Απαιτούνται κι άλλα βοηθητικά προγράμματα και παραγωγή υλικού, την οποία και θα παρουσιάσω ακολούθως. Θα αναφέρω σελίδες προέλευσης, εργαλεία καθώς και τις τεχνικές μορφές των αρχείων πολυμέσων (και μη) που χρησιμοποίησα.

**Πίνακας 5: Τεχνικές Προδιαγραφές Μέσων**

Είδος Περιεχομένου	Μορφή Αρχείου – Πρόγραμμα Παραγωγής
Κείμενα & Ερωτήσεις	Οι ερωτήσεις και μέρος της εισαγωγής, δημιουργήθηκαν με βάση το σχολικό βιβλίο και το Αναλυτικό Πρόγραμμα Σπουδών του ΥΠΕΘ για τη Γ' Δημοτικού. Δημιουργήθηκαν με LibreOffice 7.4 και αποθηκεύτηκαν ως αρχεία text διαχωριζόμενα με κόμμα, με επέκταση .csv
Εικόνες & Διαγράμματα	Χρησιμοποιήθηκαν εικόνες από το βιβλίο της Ιστορίας της Γ' Δημοτικού. Από τις υπόλοιπες εικόνες, όσες δεν είναι κατασκευασμένες από εμένα, είναι free assets από το <a href="https://assetstore.unity.com/">https://assetstore.unity.com/</a> ή είναι freeware από το διαδίκτυο, από τη σελίδα <a href="https://www.deviantart.com/">https://www.deviantart.com/</a> .

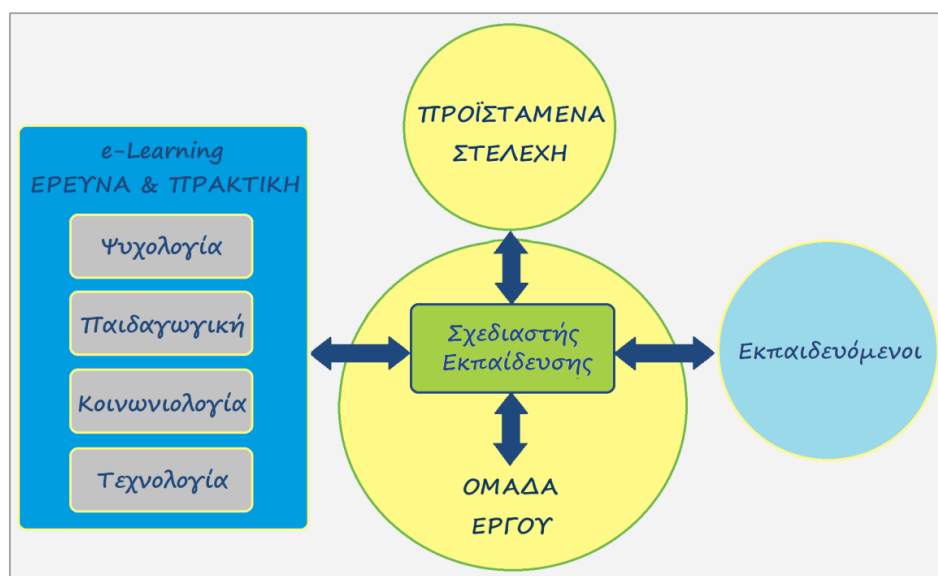
	<p>Επεξεργασία σε εικόνες και η κατασκευή των διαγραμμάτων έγινε με το γνωστό, απλό πρόγραμμα Paint των Windows, με το GIMP 2.10.32 ενώ πολλές επεξεργάστηκα με το δωρεάν on-line εργαλείο <a href="https://www.remove.bg/">https://www.remove.bg/</a>. Τα αρχεία εικόνων είναι τύπου .jpg και .png (όπου απαιτούνταν διαφάνεια).</p>
Ήχοι	<p>Οι ήχοι που χρησιμοποίησα στα animations είναι δωρεάν από το <a href="https://pixabay.com/sound-effects/">https://pixabay.com/sound-effects/</a>, ενώ οι ηχογραφήσεις των φωνών των ηρώων έγιναν με την εφαρμογή Ηχογράφηση των Windows· όλοι οι ήχοι επεξεργάστηκαν με το πρόγραμμα Audacity και τα αρχεία ήχου είναι της μορφής .mp3</p>
Animations	<p>Οι πολυαγαπημένοι ήρωες του παιχνιδιού μου είναι παιδιά της mixamo, εγγόνια της Adobe, που τα προσφέρει δωρεάν στη σελίδα <a href="https://www.mixamo.com/#/">https://www.mixamo.com/#/</a>. Το ίδιο και οι κινήσεις τους. Ο συνδυασμός των κινήσεων ώστε να προκύψει όλο το animation έγινε με τον Animator της Unity, όπως έγιναν και όλα τα animations κουμπιών, εικόνων και της κεντρικής πλατφόρμας ερωτήσεων. Τα αρχεία animation είναι επέκτασης .anim και οι αντίστοιχοι ελεγκτές είναι .controller</p>
Scripts	<p>Όλα τα scripts είναι γραμμένα με τη βοήθεια του Visual Studio 2019, σε C# και αποθηκευμένα ως αρχεία .cs ενώ το χειρισμό των βάσεων δεδομένων τον κάνουν scripts γραμμένα σε php, προφανώς με επέκταση .php που συντάχθηκαν στο Notepad++. Οι δοκιμές σε τοπικό επίπεδο έγιναν μέσω του xampp και phpMyAdmin, δουλεύοντας ως γλώσσα ερωταποκρίσεων την MySQL.</p>
SetUp File Εφαρμογής	<p>Για την εκδοχή του παιχνιδιού που κατεβαίνει τοπικά στον υπολογιστή, χρησιμοποίησα το δωρεάν πρόγραμμα Inno Setup Compiler, με το οποίο έφτιαξα το αρχείο Esiodos-</p>



	SetUpFile.exe το οποίο μπορεί να εγκαταστήσει το παιχνίδι στον υπολογιστή που θα το εκτελέσει.
Σελίδα της Εφαρμογής	Η Σελίδα που έφτιαξα για να υποστηρίξω και να διανείμω, στην ουσία, το παιχνίδι μου είναι φτιαγμένη σε HTML· τα αρχεία της δουλεύτηκαν στο Notepad++ και προφανώς έχουν επέκταση .html εκτός από το βιογραφικό μου που είναι αρχείο .pdf
Ερωτηματολόγιο Αξιολόγησης	Το ερωτηματολόγιο της Αξιολόγησης έγινε με Google Forms. Τα αποτελέσματά του επεξεργάστηκαν με Google Sheets.
Επικοινωνία με το Server	Η επικοινωνία με τον server όπου ανέβασα το παιχνίδι μου γίνεται με το ελεύθερο λογισμικό FTP, FileZilla.

### 3.2.3 Σχεδιαστικές Επιλογές

Ο σχεδιασμός ενός παιχνιδιού και μάλιστα εκπαιδευτικού, δεν είναι καθόλου εύκολη υπόθεση. Για την ακρίβεια, είναι ένα από τα βασικά βήματα ανάπτυξης του και διέπεται από συγκεκριμένες και σαφώς καθορισμένες αρχές. Οι Αρχές Σχεδιασμού Εκπαιδευτικού Λογισμικού έχουν διατυπωθεί με έμφαση στους όρους της Ψυχοπαιδαγωγικής και της Κοινωνιολογίας και απαιτείται ολόκληρη ομάδα για να αναλάβει το πλήρες έργο.



Εικόνα 6: Οργάνωση Σχεδίασης Εκπαιδευτικού Λογισμικού



## Οι Οκτώ Χρυσοί Κανόνες Σχεδιασμού

Ένα εκπαιδευτικό λογισμικό, οφείλει να ακολουθεί τις αρχές γενικού σχεδιασμού που αφορούν στην εκμάθηση, συνοχή, προσαρμοστικότητα και ευρωστία, ενώ προϋποθέτει την ύπαρξη εξειδικευμένων υποομάδων που θα διαμοιραστούν τις ευθύνες και τις υπευθυνότητες. Διεθνείς αναγνωρισμένοι οργανισμοί προτυποποίησης έχουν θέσει πρότυπα ώστε να διασφαλίζεται η συμβατότητα μεταξύ συστημάτων, ενώ κανόνες σχεδίασης έχουν συνταχθεί από τους Smith & Moiser, τη NASA, τον Shneiderman και άλλους. Αν και απαιτείται επικύρωση και ρύθμιση για συγκεκριμένους τομείς σχεδιασμού και καμία λίστα αρχών δε μπορεί να είναι ποτέ πλήρης, οι ακόλουθοι κανόνες σχεδιασμού [42], έχουν γίνει αποδεκτοί ως χρήσιμος οδηγός για τους σχεδιαστές.

### ① Συνέπεια – Διαφάνεια

Είναι αναγκαίο, να απαιτούνται συνεπείς αλληλουχίες ενεργειών σε παρόμοιες καταστάσεις. Η ίδια ορολογία θα πρέπει να χρησιμοποιείται σε προτροπές, μενού και οθόνες βοήθειας. Εξαιρέσεις, όπως η απαιτούμενη επιβεβαίωση της εντολής διαγραφής ή η μη επανάληψη των κωδικών πρόσβασης, είναι χρήσιμο να είναι κατανοητές και περιορισμένες σε αριθμό.

Θα πρέπει να χρησιμοποιούνται παντού: σταθερό χρώμα, διάταξη, κεφαλαία, γραμματοσειρές και ούτω καθεξής, ενώ οι δυνατότητες του συστήματος (χρήση πολυμέσων, πολλαπλών χρωμάτων, εικόνων κλπ.) έχει νόημα να χρησιμοποιούνται με μοναδικό κριτήριο τη διευκόλυνση του μαθητή και όχι την παρουσίαση τεχνικών δυνατοτήτων του συστήματος. Κάθε πρόσθετη πληροφορία «ανταγωνίζεται» την σημαντική πληροφορία στη μνήμη του χρήστη με αποτέλεσμα να την καθιστά λιγότερο «ορατή» και αξιοποιήσιμη.

Η διάταξη των αντικειμένων πρέπει να αντικατοπτρίζει τη λογική ακολουθία των ενεργειών του μαθητή και η αναπαράσταση των στοιχείων να γίνεται με συνεπή και ομοιόμορφο τρόπο ώστε το σύστημα να έχει προβλέψιμη συμπεριφορά.



Όλα αυτά συμβαίνουν με σταθερή ευλάβεια στον Ησίοδο. Συνέπεια υπάρχει σε όλες τις οθόνες, οι βοήθειες είναι παρόμοιες, τα μενού αναπτύσσονται με τον ίδιο τρόπο παντού και το σύστημα έχει απολύτως προβλέψιμη συμπεριφορά, σε σημείο που η αίτηση βοήθειας είναι περιττή.

## ② Καθολική Χρηστικότητα

Οι χρήστες/εκπαιδευόμενοι έχουν διαφορετικές ανάγκες, οι οποίες επιβάλλεται να αναγνωριστούν. Το λογισμικό οφείλει να έχει πλαστικότητα, διευκολύνοντας τον μετασχηματισμό του περιεχομένου και καλύπτοντας διαφορές από αρχάριους έως ειδικούς, διάφορα εύρη ηλικιών, αναπηρίες, διεθνείς παραλλαγές και τεχνολογική ποικιλομορφία.

Παραδείγματος χάριν, η προσθήκη λειτουργιών για αρχάριους (επεξηγήσεις) και λειτουργιών για ειδικούς, (συντομεύσεις και ταχύτερος βηματισμός) εμπλουτίζει τη σχεδίαση της διεπαφής και βελτιώνει την αντιληπτή ποιότητα.

Κάθε λογισμικό, είναι χρήσιμο έως επιβεβλημένο, να συνοδεύεται από πληροφορίες βοήθειας (help pages), που να προσφέρουν στο χρήστη πληροφορίες άμεσα σχετικές με το θέμα που τον απασχολεί, να παρουσιάζει με σαφήνεια τα βήματα που πρέπει να γίνουν και να μην είναι μεγάλες σε έκταση.



Ο Ησίοδος απευθύνεται σε συγκεκριμένο κοινό, χωρίς μεγάλες διαφορές στην αντιληπτική ικανότητα. Συνοδεύεται, βεβαίως, από απλές και σαφείς πληροφορίες βοήθειας, αλλά δεν περιλαμβάνει ειδικές λειτουργίες για αρχάριους ή πιο έμπειρους παίκτες, γιατί δεν είναι τόσες πολλές οι λειτουργίες που θα μπορούσαν να διεκπεραιωθούν με εναλλακτικούς τρόπους.

## ③ Ενημερωτική Ανατροφοδότηση – Απόδοση – Προσανατολισμός

Για κάθε ενέργεια χρήστη, απαιτείται να υπάρχει μια ανατροφοδότηση διεπαφής. Για συχνές και δευτερεύουσες ενέργειες, η ανταπόκριση μπορεί να είναι μέτρια, ενώ για σπάνιες και μεγάλες ενέργειες, η απόκριση θα πρέπει να είναι πιο ουσιαστική. Η οπτική παρουσίαση των αντικειμένων ενδιαφέροντος παρέχει ένα βολικό περιβάλλον για τη ρητή εμφάνιση των αλλαγών.

Η ιεραρχία ενεργειών ελέγχου δεν πρέπει να υπερβαίνει τα τρία επίπεδα ενώ είναι αποτελεσματικό, ο μαθητής να λαμβάνει πληροφορίες που του γνωστοποιούν την τρέχουσα κατάσταση του συστήματος. Έτσι, αν μια αιτηθείσα εργασία αδυνατεί να εκτελεστεί, ο χρήστης είναι αναγκαίο να ενημερώνεται άμεσα. Το ίδιο και όταν καθυστερεί μια εκτύπωση. Όταν ολοκληρώνεται μια εργασία και παράγονται αποτελέσματα, αυτά πρέπει να γνωστοποιούνται στο χρήστη. Όταν επιλέγεται κάποιο στοιχείο μπορεί να τονίζεται, ενώ μπορεί να γίνεται χρήση ειδικού συμβόλου δρομέα για εργασίες που διαρκούν από 1 έως 10 sec.

Είναι ουσιώδες, κάθε στιγμή, το σύστημα να παρέχει πληροφορίες για το πού βρίσκεται ο χρήστης, τι μπορεί να κάνει και πώς να προχωρήσει. Για παράδειγμα, θα πρέπει να υπάρχει επεξηγηματικός τίτλος της οθόνης σε σταθερή θέση. Ακόμη, όπου αυτό χρειάζεται, να γίνεται χρήση γενικού πλάνου εφαρμογής (site map) με ένδειξη της σχετικής θέσης του μαθητή, ενώ η έξοδος από το σύστημα ή την τρέχουσα δραστηριότητα πρέπει να είναι εύκολη από οποιοδήποτε σημείο.



Ανατροφοδότηση διεπαφής υπάρχει σε κάθε βήμα στο εκπαιδευτικό μου παιχνίδι. Όταν ολοκληρώνεται μία ενέργεια, είτε πρόκειται για μια σωστή απάντηση, είτε για μια λανθασμένη, είτε για την ολοκλήρωση του επιπέδου, αυτό γνωστοποιείται στον παίκτη. Το σύστημα παρέχει πληροφορίες για την πρόοδο του χρήστη, τόσο μέσα σε κάθε επίπεδο, όσο και με γενικά πλάνα εφαρμογής ενοτήτων στην οθόνη επιλογής επιπέδου. Τα βοηθητικά μενού βρίσκονται πάντα στο ίδιο σημείο και παρέχουν τη δυνατότητα εξόδου τόσο από την οθόνη που βρίσκεται ο παίκτης όσο και από το παιχνίδι γενικότερα, ακόμα και αν αυτό παίζεται σε λειτουργία παραθύρου. Site maps υπάρχουν στα μενού επιπέδων.

#### ④ Κλείσιμο Ενοτήτων

Οι ακολουθίες ενεργειών, είναι σημαντικό να οργανώνονται σε ομάδες με αρχή, μέση και τέλος. Η ενημερωτική ανατροφοδότηση με την ολοκλήρωση μιας ομάδας ενεργειών δίνει στους χρήστες την ικανοποίηση της ολοκλήρωσης, μια αίσθηση ανακούφισης, ένα μήνυμα για να απορρίψουν τα σχέδια έκτακτης ανάγκης από το μυαλό τους και έναν δείκτη προετοιμασίας για την επόμενη ομάδα.



Φυσικά αυτό συμβαίνει και στον Ησίοδο. Κάθε ενέργεια είναι οργανωμένη ως αυτόνομη ενότητα και για το λόγο αυτό επιλέχθηκε η χρήση διαφόρων οθονών, ώστε να είναι ξεκάθαρη κάθε λειτουργία. Αντίστοιχα, η ικανοποίηση της ολοκλήρωσης κάθε επιπέδου παρέχεται με την ενημέρωση των επιτευγμάτων και την επιβράβευση μέσω αναδυόμενων μηνυμάτων. Το παιχνίδι οδηγεί το χρήστη, εάν αυτός δεν επιλέξει κάτι διαφορετικό μόνος του.

#### ⑤ Αποτροπή Σφαλμάτων

Ο σχεδιασμός της διεπαφής πρέπει να είναι έτσι ώστε οι χρήστες να μην μπορούν να κάνουν σοβαρά λάθη. Παραδείγματος χάριν, να μην επιτρέπεται εισαγωγή αλφαβητικών χαρακτήρων στα αριθμητικά πεδία. Σε περίπτωση σφάλματος, το λογισμικό μπορεί να προσφέρει απλές, εποικοδομητικές και συγκεκριμένες οδηγίες για

ανάκτηση και ανάνηψη από την κατάσταση σφάλματος. Τα μηνύματα σφάλματος, είναι χρήσιμο να είναι σύντομα και διατυπωμένα κατά τρόπο ουδέτερο, χωρίς χρήση προσβλητικής ή σκωπτικής διάθεσης.

Οι εσφαλμένες ενέργειες, είναι βασικό να αφήνουν αμετάβλητη την κατάσταση του λογισμικού, το οποίο υποχρεούται να δίνει οδηγίες για την επαναφορά στην προ λάθους κατάσταση. Για παράδειγμα, οι χρήστες δεν θα πρέπει να πληκτρολογούν ξανά μια ολόκληρη φόρμα ονόματος-διεύθυνσης εάν εισάγουν μη έγκυρο ταχυδρομικό κώδικα, αλλά θα πρέπει να καθοδηγούνται για την επισκευή μόνο του ελαττωματικού τμήματος, ενώ το λάθος δεν θα πρέπει να οδηγεί σε ανώμαλο τερματισμό του λογισμικού, αλλά να το διατηρεί/επαναφέρει στην κατάσταση αναμονής εισόδου του ελλιπούς στοιχείου.



Στον Ησιόδο τα περιθώρια λάθους είναι ελάχιστα. Εισαγωγή από τον παίκτη απαιτείται μόνο αν θέλει να βάλει τα στοιχεία του και σε περίπτωση που δεν τα βάλει σωστά, απλά παίζει σαν «επισκέπτης» χωρίς καμία αρνητική συνέπεια στη λειτουργία του παιχνιδιού. Ελάχιστες εισαγωγές ειδικών χαρακτήρων που θα μπορούσαν να δημιουργήσουν πρόβλημα, διορθώνονται αυτόματα από το ίδιο το λογισμικό. Κάποιο άλλο λάθος δεν έχω καταφέρει να εντοπίσω μέχρι στιγμής.

### © Εύκολη Αντιστροφή Ενεργειών

Όσο είναι δυνατόν, οι ενέργειες πρέπει να είναι αναστρέψιμες. Αυτή η δυνατότητα ανακουφίζει από το άγχος, καθώς οι χρήστες γνωρίζουν ότι τα σφάλματα μπορούν να αναιρεθούν και ενθαρρύνει την εξερεύνηση άγνωστων επιλογών. Γενικά το λογισμικό δεν πρέπει να «τιμωρεί» τον χρήστη. Επιθυμητό είναι να του επιτρέπει να ακυρώσει την επιλογή του και να επιστρέψει στην αρχική του κατάσταση χωρίς να τον «κλειδώνει» σε μια σειρά ενεργειών που ξεκίνησε κατά λάθος.

Για παράδειγμα, μπορούν να προσφέρονται δυνατότητες ακύρωσης (undo) & επαναφοράς (redo) των ενεργειών ή να υπάρχουν ενσωματωμένες δυνατότητες «επιστροφής» του χρήστη στο προηγούμενο βήμα ώστε να μπορεί να μελετήσει μια μαθησιακά ενδιαφέρουσα κατάσταση, ιδιαίτερα στις προσομοιώσεις.



Ακύρωση ή επαναφορά δεν υποστηρίζει ο Ησιόδος. Η σωστή απάντηση είναι σωστή και η λάθος, λάθος. Ωστόσο ο παίκτης παίζει χωρίς άγχος αφού δεν «τελειώνουν» οι ευκαιρίες του, μέχρι να

συμπληρώσει γνωστό αριθμό επιτυχημένων απαντήσεων. Ακόμη μπορεί να παίζει το ίδιο επίπεδο όσες φορές θέλει, ώστε να πετύχει μέγιστη βαθμολογία· στην πραγματικότητα να μάθει καλύτερα τις ερωτήσεις. Σε κάθε οθόνη μπορεί να αποχωρίσει χωρίς ποινές αν μετάνιωσε ή αν την επέλεξε κατά λάθος και γενικότερα δεν εγκλωβίζεται πουθενά, ούτε καν στην επιλογή του παίκτη του για τη συγκεκριμένη συνεδρία: μπορεί να αλλάξει και να δοκιμάσει ήρωες όσες φορές θέλει. Ο χρήστης δεν τιμωρείται από το λογισμικό, ούτε εγκλωβίζεται, ούτε προσβάλλεται. Ίσα-ίσα που η περίπτωση του λάθους αντιμετωπίζεται με χιούμορ.

### ⑦ Έλεγχος Χρηστών

Οι έμπειροι χρήστες επιθυμούν έντονα να έχουν την αίσθηση ότι είναι υπεύθυνοι για τη διεπαφή και ότι η διεπαφή ανταποκρίνεται στις ενέργειές τους. Αλλά και γενικά, ο ρυθμός παρουσίασης των πληροφοριών πρέπει να καθορίζεται από το μαθητή. Μπορούν να διατίθενται πολλαπλά μέσα για την επίτευξη των στόχων του εκπαιδευόμενου. Οι μέθοδοι καταχώρησης στοιχείων, είναι χρήσιμο, να είναι σταθεροί σε όλη την εφαρμογή, ενώ ποτέ ένα στοιχείο δεν πρέπει να ζητείται από το χρήστη περισσότερες από μία φορές.



Πραγματικά, στον ΗΣίοδο, ποτέ ένα στοιχείο δεν ζητείται από το χρήστη πάνω από μία φορά. Ακόμη και ένας αρχάριος παίκτης έχει πλήρη ελευθερία και αυτονομία να ελέγξει τη διεπαφή· να αποφασίσει με ποια σειρά θα παίζει ή θα επιλέξει ενότητα ή επίπεδο, τότε θα αλλάξει οθόνη ή/και αν θα ολοκληρώσει κάποια συνεδρία παιχνιδιού.

### ⑧ Μείωση Φορτίου της Βραχυπρόθεσμης Μνήμης

Η περιορισμένη ικανότητα των ανθρώπων για επεξεργασία πληροφοριών στη βραχυπρόθεσμη μνήμη (εμπειρικός κανόνας: οι άνθρωποι μπορούν να θυμούνται  $7 \pm 2$  κομμάτια πληροφοριών) απαιτεί από τους σχεδιαστές να αποφεύγουν τις διεπαφές στις οποίες οι χρήστες πρέπει να θυμούνται πληροφορίες από μια οθόνη και στη συνέχεια να χρησιμοποιούν αυτές οι πληροφορίες σε άλλη οθόνη.

Δεν πρέπει να επιβαρύνεται η ενεργός μνήμη των χρηστών από την ανάγκη να θυμούνται πάρα πολλά πράγματα για τις λειτουργίες της διεπαφής. Ο σχεδιασμός, είναι αναγκαίο να γίνεται με βάση την αναγνώριση πληροφορίας και όχι την ανάκληση.



Η βραχυπρόθεσμη μνήμη των νεαρών παικτών δεν χρειάζεται να επιβαρυνθεί από τίποτε, αφού δε χρειάζεται να θυμούνται τίποτα απολύτως κατά την περιήγησή τους από μια οθόνη σε μια άλλη. Κάθε ενέργεια γίνεται με βάση την αναγνώριση πληροφορίας και δεν απαιτείται καμία ανάκληση.

### **Οι ανατροφοδοτήσεις στα Εκπαιδευτικά Λογισμικά**

Οι ανατροφοδοτήσεις σε ερωτήσεις, εργασίες και διαδικασίες πρέπει να επιβραβεύουν το χρήστη στην επιτυχία και να μην είναι απαξιωτικές στην αποτυχία αλλά προτρεπτικές, υποστηρικτικές και να ενισχύουν τους μαθητές σε συνέχιση της προσπάθειας. Ακόμα και ένα κακό αποτέλεσμα μπορεί να διατυπωθεί με άρνηση της θετικής, παρά με παράθεση της συνώνυμης αρνητικής έννοιας. Αυτό είναι ιδιαίτερος σημαντικό, ανθρώπινο και παιδαγωγικό, γι' αυτό και δεν θα μπορούσε να είναι διαφορετικά σ' ένα παιχνίδι που αφορά σε μικρά παιδιά που έρχονται πρώτη φορά σε επαφή με ένα καινούριο μάθημα.

### **Δομή του Λογισμικού και Εκπαιδευτικές Προδιαγραφές**

Ο σχεδιαστής, αφού λάβει τις αποφάσεις για την υλοποίηση των παραπάνω αρχών, κατά το δυνατόν και κατά τις απαιτήσεις που έχει το λογισμικό του, βρίσκεται στην καίρια θέση να αποφασίσει και τη δομή του λογισμικού του, αν θα είναι γραμμική, ιεραρχική, δενδροειδής ή δικτυωτή, ανάλογα με το θέμα που διαπραγματεύεται και τους υπάρχοντες μαθησιακούς στόχους και φυσικά να λάβει αποφάσεις σχετικά με την Αρχή Σχεδίασης Στοιχείων Πολυμέσων.

Το ίδιο έκανα και εγώ. Επέλεξα μια μικτή δομή: φαίνεται γραμμική, με την έννοια ότι έχει σαφώς καθορισμένη αρχή, μέση και τέλος, αλλά ο παίκτης έχει τη δυνατότητα να κινηθεί δικτυωτά στο σώμα του παιχνιδιού, παρέχοντας άνεση, ασφάλεια και μείωση άγχους.

Για να πετύχω υψηλή διάδραση και μάθηση ταυτόχρονα θα έπρεπε να οργανώσω κατάλληλα τη δομή του παιχνιδιού και τα στοιχεία πολυμέσων με βάση επίσης συγκεκριμένες αρχές σχεδίασης [43] και να τις αξιοποιήσω κατάλληλα ώστε να πετύχω βέλτιστο –κατά το δυνατόν– αποτέλεσμα.

- **Αρχή Πολυμέσων (Multimedia Principle)**

[αλληλοϋποστηριζόμενα κείμενα και εικόνες που βελτιώνουν τις συνθήκες μάθησης]

Χρησιμοποίησα κατάλληλα στοιχεία πολυμέσων (εικόνες, κίνηση και ήχους) ώστε να αναπτύξω το παιχνίδι όχι μόνον με τη βοήθεια κειμένου. Για παράδειγμα, δεν αρκέστηκα στην ενημέρωση του παίκτη για το σωστό ή λάθος της απάντησής του, αλλά την πλαισίωσα με αστείους ήχους και κατάλληλα animations (ο παίκτης χορεύει ή πέφτει, η λάθος απάντηση φεύγει στο κενό, κ.ο.κ) οι οθόνες βοήθειας αναδύονται σαν μέρος βίντεο και το ίδιο συμβαίνει με τις οθόνες επιβράβευσης, ενώ η εισαγωγή περιλαμβάνει λόγο (ηχητικό) και γραπτά κείμενα που τον συνοδεύουν.

- **Συνεκτικότητα (Coherence)**

[κείμενο και εικόνες που περιέχουν τις βασικές πληροφορίες που είναι απόλυτα σχετικές στη συγκεκριμένη οθόνη]

Σε κάθε οθόνη παρουσιάζεται μόνο η απαραίτητη ανάδραση όποτε χρειάζεται και βασικές πληροφορίες που είναι σχετικές με το στόχο της μάθησης ή/και τη λειτουργία της οθόνης όταν πρόκειται για οθόνες διεργασιών. Πιστεύω πως πέτυχα μεγάλη συνεκτικότητα. Απέφυγα αποπροσανατολιστικές και παραπλανητικές, περιττές αναφορές, όπως απέφυγα και πολλές οδηγίες, επεξηγήσεις και ενημερώσεις (που τα παιδιά τα θεωρούν ιδιαίτερα βαρετά). Οι μαθητές ξέρουν πώς παίζονται τα παιχνίδια και βρίσκουν τον τρόπο!

- **Γειννίαση (Contiguity)**

[στο χώρο: σχετικές πληροφορίες (λέξεις & εικόνες) κοντά μεταξύ τους στην οθόνη ώστε να διευκολύνεται η σύνδεση μεταξύ τους· στο χρόνο: σχετικές πληροφορίες (λέξεις & εικόνες) ταυτόχρονα στην οθόνη· σύνδεση – συσχέτιση: σχεδιαστικά «τεχνάσματα» ώστε να βοηθηθεί ο χρήστης να καταλάβει ποιες πληροφορίες (λέξεις & εικόνες) «συνδέονται» μεταξύ τους]

Σε όλες τις οθόνες, αναφορικά με το χώρο, το κέντρο της οθόνης καταλαμβάνει σταθερά την κυρίως πληροφορία ενώ οποιοδήποτε κείμενο εμφανίζεται, προβάλλεται ακριβώς πάνω στο στοιχείο που περιγράφει. Σχετικά με το χρόνο, ήχος και ενέργειες ή αναδράσεις παρουσιάζονται ταυτόχρονα. Υιοθετήθηκαν συγκεκριμένα χρώματα, δεδομένου να γνωρίζει ο παίκτης το σωστό και το λάθος, το ναι και το όχι (πράσινο – κόκκινο) και αυτά διατηρούνται σε όλη την έκταση του παιχνιδιού.

- **Τροπικότητα (Modality)**



[κατάλληλος συνδυασμός, για παράδειγμα ακουστικής & οπτικής, για υποστήριξη του χρήστη να επεξεργαστεί την πληροφορία μέσω διπλής κωδικοποίησης ή όταν πρόκειται να εξηγηθούν σύνθετα γραφικά στην οθόνη ή να παρουσιαστούν κινούμενα γραφικά]

Γενικά οι τροπικότητες που συνδυάζονται είναι ακουστική και οπτική, συμπεριλαμβανομένων των κινήσεων (animations) ενώ απόλυτος και κυρίαρχος στόχος της χρήσης τους είναι, το παιχνίδι να μην θυμίζει εκπαιδευτικό λογισμικό, παρά παιχνίδια με τα οποία είναι ήδη εξοικειωμένοι οι μικροί μαθητές.

- **Πλεονασμός (Redundancy)**

[δεν είναι οπωσδήποτε ή/και πάντα η καλύτερη παρουσίαση]

Ο πλεονασμός είναι κάτι που απέφυγα, στο μέτρο που ήθελα να το αποφύγω. Διατήρησα τις αρχές περί αριθμού γραμματοσειρών (χρησιμοποιώ μόνο μία), σταθερών χρωμάτων, συνδυασμών και ταυτόχρονης χρήσης πολυμέσων. Ωστόσο επέτρεψα γεμάτα σκηνικά και πολλαπλή τροπικότητα θεωρώντας πως έτσι θα διεγείρω το ενδιαφέρον των μαθητών.

- **Προσωπικό στυλ παρουσίασης (Personalization)**

[παρουσίαση της πληροφορίας με φιλικό προσωπικό λόγο, κείμενο ή αφήγηση σε πρώτο ή δεύτερο πρόσωπο και όχι τυπική απρόσωπη παρουσίαση, χρήση φιλικών χαρακτήρων (agents, πρακτόρων-βοηθών) για παρουσίαση των πληροφοριών, που παρακινούν τους χρήστες αποτελεσματικότερα στο να προσέξουν και να επεξεργαστούν την πληροφορία]

Το προσωπικό στυλ παρουσίασης είναι κάτι που αποπνέεται στο λογισμικό ηθελημένα ή όχι. Είναι σημαντικό, το λογισμικό να «μιλά» τη γλώσσα του χρήστη. Να είναι κατανοητό από αυτόν (λέξεις, φράσεις, έννοιες), χωρίς δύσκολους τεχνικούς όρους. Για το λόγο αυτό, προσπάθησα όλο το παιχνίδι να έχει στυλ φιλικό, όπου απευθύνεται στους μαθητές/παίκτες και χρησιμοποίησα δεύτερο ενικό πρόσωπο, με οικείες εκφράσεις και απλή γλώσσα.

### **Επαναληπτικά Μοντέλα Σχεδίασης**

Τα σχεδιαστικά διλήμματα και οι σχεδιαστικές επιταγές δεν σταματούν εδώ. Οι μεθοδολογίες ανάπτυξης εκπαιδευτικού λογισμικού, που εστιάζει στις ανάγκες των μαθητών και των νοητικών διεργασιών που υποστηρίζονται μέσα από αυτό,



υπηρετώντας τον ανθρωποκεντρικό σχεδιασμό, βασίζονται κυρίως σε επαναληπτικά μοντέλα σχεδίασης. Έτσι, ο Ησίοδος έχει επαναλαμβανόμενες οθόνες χρήσης ή εκμάθησης, ώστε να εξοικειώνεται ο παίκτης με τη λειτουργία του.

### **Πρότυπα Σελίδων Εφαρμογής**

Με τον ορισμό προτύπων (templates) σελίδων ή οθονών μιας εφαρμογής, εξασφαλίζεται η διατήρηση του τρόπου/στυλ παρουσίασης αλλά και η εύκολη διαχείριση μεγάλου αριθμού οθονών, επιτυγχάνοντας ομοιομορφία στην παρουσίαση του περιεχομένου. Γεγονός απολύτως απαραίτητο στην περίπτωση του Ησίοδου, ιδιαίτερα λόγω της φύσης (και της ηλικίας) της ομάδας στόχου στην οποία απευθύνεται το παιχνίδι. Η ύπαρξη ενός προτύπου:

- συνδέει μεταξύ τους τα στοιχεία του περιβάλλοντος εφαρμόζοντας έναν ενοποιημένο τρόπο παρουσίασης
- προσθέτει ενδιαφέρον
- βοηθά στην πλοήγηση
- ενισχύει την κατανόηση του τρόπου που είναι οργανωμένες οι πληροφορίες

Οι βοηθητικές σελίδες έχουν σταθερό ύψος με παρόμοιες εικόνες υποβάθρου (background), ενώ οι κεντρικές του παιχνιδιού ακολουθούν το ίδιο ακριβώς πρότυπο με διαφορετικό background (skyBox), το ίδιο ακριβώς μενού και τις ίδιες βοήθειες και ενημερώσεις.

### **Βασικές και Καλά Σχεδιασμένες Οθόνες**

Η ύπαρξη βασικών οθονών είναι άλλη μια αρχή που είναι απαραίτητη στο σχεδιασμό ενός εκπαιδευτικού παιχνιδιού. Ο Ησίοδος, όπως οφείλει κάθε εκπαιδευτικό λογισμικό, έχει τουλάχιστο:

- εισαγωγική οθόνη
- κεντρική οθόνη
- οθόνες διεργασιών

ώστε να εξυπηρετείται η καλή πλοήγηση των εκπαιδευομένων στο λογισμικό και η δημιουργία θετικής στάσης απέναντί του. Με βάση αυτή την αρχή, σχεδίασα δύο με τρεις εισαγωγικές οθόνες: την εναρκτήρια, μία για την πρώτη φορά που θα παιχτεί το παιχνίδι (τις επόμενες μπορεί να παραλείπεται) και μία οθόνη συλλογής στοιχείων του παίκτη. Η κεντρική οθόνη είναι μία και βασική για όλα τα επίπεδα, ενώ υπάρχουν

οθόνες διεργασιών για την επιλογή παίκτη/πρωταγωνιστή, για την επιλογή ενότητας και την επιλογή επιπέδου, καθώς και οθόνη καλύτερων βαθμολογιών πριν την έξοδο.

Οι οθόνες για να είναι καλά σχεδιασμένες θα πρέπει να είναι

- οπτικά ενδιαφέρουσες
- εύκολες στην ανάγνωσή τους (ακολουθία της σχεδίασης «Z»)
- χωρίς ενοχλητικά ή αποπροσανατολιστικά στοιχεία
- να υποστηρίζουν το χρήστη-εκπαιδευόμενο ως προς
  - την προσοχή και το ενδιαφέρον,
  - την εμπλοκή και την επεξεργασία και
  - την πλοήγηση και την οργάνωση.

Αυτό προσπάθησα να κάνω. Διάλεξα ενδιαφέρουσες εικόνες ως υπόβαθρα, σχέδια και μενού που θυμίζουν παιχνίδια περιπέτειας και προσπάθησα να κινήσω το ενδιαφέρον με κινήσεις (animations) των ηρώων και του σκηνικού. Φυσικά, η επιλογή είναι υποκειμενική και το αποτέλεσμα θα κριθεί από την αξιολόγηση του έργου. Θεωρώ ότι δεν περιέλαβα περιττά στοιχεία και οι βασικές οθόνες ακολουθούν το Z-σχεδιασμό. Παραδείγματος χάριν, στην κεντρική οθόνη του παιχνιδιού, όπου ο παίκτης περνάει και τον περισσότερο χρόνο, οι πληροφορίες για την πρόοδο του παιχνιδιού είναι πάνω, από αριστερά προς τα δεξιά, το κυρίως σκηνικό βρίσκεται και παίζεται στη μέση και οι πληροφορίες εξέλιξης του επιπέδου, όπως και το σταθερό μενού, βρίσκονται κάτω, επίσης από αριστερά προς τα δεξιά, όπως φαίνεται στις Εικόνες 14 της σελίδας 61.

### Στρατηγική Διδακτικής Σχεδίασης του Gagne

Από το 1985, διατυπώθηκε από τον Gagne μια λίστα από εννέα διδακτικά γεγονότα στα οποία μπορεί να στηριχθεί ένας εκπαιδευτικός για να σχεδιάσει ένα μάθημα [44]. Ο τρόπος επίτευξης αυτών κατά το σχεδιασμό εκπαιδευτικού λογισμικού, καθοδηγεί και αποτελεί προσφιλή τακτική των σχεδιαστών, ακόμα και σήμερα. Κατ' αυτήν το εκπαιδευτικό λογισμικό πρέπει να:

- προσελκύει την προσοχή του χρήστη και παρέχει κίνητρα χρήσης,

Για να εστιαστεί η «προσοχή» του χρήστη στο αντικείμενο της διδασκαλίας επιλέχτηκε η χρήση διαδραστικού πρωταγωνιστή βοηθού και μάλιστα όχι ενός αλλά με δυνατότητα επιλογής από πέντε επώνυμους φίλους. Ο Θανάσης, η Αλεξάνδρα, ο Πόλυς, ο Γιώργος και η Σούζαν εμφανίζονται από τις πρώτες

οθόνες, δημιουργώντας αναμονή και ενδιαφέρον, ενώ όλη η επαναληπτική δραστηριότητα προβάλλεται ως ένα παιχνίδι με συγκέντρωση πόντων, ολοκλήρωσης επιπέδων και κατάκτηση των αντίστοιχων σημάτων τους.

- πληροφορεί το χρήστη για τους στόχους του μαθήματος,

Οι στόχοι του μαθήματος αναφέρονται στην εισαγωγική οθόνη. Αναλυτικότερα δεν αναφέρονται «**εκπαιδευτικοί στόχοι**», ούτε γίνεται σύνδεση με προηγούμενες γνώσεις του εκπαιδευόμενου, ούτε προαπαιτείται συγκεκριμένη σειρά ολοκλήρωσης των επιπέδων, επειδή το λογισμικό μπορεί να λειτουργήσει τόσο για εκμάθηση, όσο και επαναληπτικά και ευελπιστεί να είναι μαθητοκεντρικό: ο μαθητής θα μπορεί να επιλέγει ενότητα, επίπεδο και σειρά ολοκλήρωσης. Άλλωστε το ζητούμενο είναι να μην έχει στο μυαλό του ο μικρός μαθητής ότι... μαθαίνει αλλά πως παίζει κάτι που τον ευχαριστεί.

- προκαλεί ανάκληση της αποκτηθείσας γνώσης,

Η «**ανάκληση**» πρότερης γνώσης γίνεται άμεσα με την εισαγωγή στο κεντρικό παιχνίδι, όπου ο μαθητής/παίκτης ενεργοποιείται προκειμένου να απαντήσει σε ερωτήσεις που έχει ήδη διδαχθεί.

- παρουσιάζει αποτελεσματικά το περιεχόμενο,

Για την «**παρουσίαση υλικού**» επιλέχτηκε η μέθοδος των ερωτήσεων πολλαπλής επιλογής, η οποία γίνεται διαδραστικά: ο παίκτης πηδάει ψηλά, στην ερώτηση που θεωρεί σωστή και το σενάριο συνεχίζεται ανάλογα με το αν είναι όντως σωστή ή όχι.

- καθοδηγεί το χρήστη,

Δυνατότητα για «**υποστήριξη**» παρέχεται καθ' όλη τη διάρκεια του παιχνιδιού. Η πλοήγηση από οθόνη σε οθόνη είναι τόσο εύκολη έως προφανής, ενώ βοήθεια και καθοδήγηση είναι προσφερόμενη καθολικά στο λογισμικό και ξεχωριστά σε κάθε οθόνη, καθώς και όπου θα ήταν αναγκαίο, ακόμα και αν δεν τη ζητήσει ο παίκτης. Παραδείγματος χάριν, σε κάποιο σημείο που απαιτείται η χρήση πλήκτρου για συνέχεια, δίνεται η οδηγία ούτως ή άλλως μέχρι να μάθει τη λειτουργία ο παίκτης.

- παρέχει ανατροφοδότηση,

Μετά από κάθε απάντηση του μαθητή υπάρχει άμεση ενημέρωση της επιτυχίας ή της αστοχίας του, με ηχητικό μήνυμα (εάν έχει ανοιχτό τον ήχο) και με γραφικά, παρέχοντας γρήγορη «**ανάδραση**». Ο ήρώας του, π.χ. χορεύει, η απάντησή του γίνεται πράσινη, και το σκηνικό όλο «ανεβαίνει» ακόμα πιο ψηλά στον ουρανό, μέχρι να φανερωθεί όλη η εικόνα που βρίσκεται στο υπόβαθρο.

- βοηθά στην εξαγωγή αποτελεσμάτων και συμπερασμάτων,

Με τις συνεχείς ερωτήσεις στις οποίες υποβάλλεται ο παίκτης επιτυγχάνεται η συνεχής «**δράση**» του ενώ η εξαγωγή των αποτελεσμάτων είναι διαρκής. Με κάθε απάντηση ενημερώνεται για το πόσες απαντήσεις έχει πετύχει σωστά στο σύνολο των ερωτήσεων, βλέπει πόσες ακόμα πρέπει να ολοκληρώσει προκειμένου να κερδίσει το επίπεδο και στο τέλος βγάζει τα συμπεράσματά του με την ενημέρωση της βαθμολογίας του και τη θέση του σε μια συνολική κατάταξη.

- βοηθά στην αξιολόγηση αποτελεσμάτων και συμπερασμάτων,

Η «**αξιολόγησή**» του αφήνεται στον ίδιο, ενώ η βαθμολογία που θα συγκεντρώσει είναι, ενδεικτική και μεταβλητή, αφού μπορεί να παίζει και να ξαναπαίζει το ίδιο επίπεδο όσες φορές θέλει, αλλάζοντας τη βαθμολογία του. Τα παιδιά είναι ειλικρινείς κριτές του εαυτού τους και η συνολική κατάταξη αποτελεί κίνητρο και μέσο ελέγχου. Εξάλλου κύριος στόχος είναι η μάθηση κι όχι ο χαρακτηρισμός του παίκτη ως καλού ή μέτριου.

- αναπτύσσει τη μνήμη και μεταφέρει γνώση.

Ως προς την «**ενίσχυση**» της γνώσης, δίνεται η δυνατότητα επανάληψης των επιπέδων του παιχνιδιού όσες φορές επιθυμεί ο χρήστης. Σημαντική στον τομέα αυτό, είναι και η επαναχρησιμοποίηση των ερωτήσεων στις οποίες αστοχεί ο μαθητής. Αν επιλέξει λάθος απάντηση, αντί να υπάρχει ανατροφοδότηση με τη σωστή απάντηση, η ερώτηση ξαναμπαίνει στη λίστα των υποψήφιων επερχόμενων ερωτήσεων. Με αυτόν τον τρόπο θα έχει τη δυνατότητα να ξανασυναντήσει την ίδια ερώτηση γνωρίζοντας ποια επιλογή ήταν άστοχη, εκμαιεύοντας έτσι τη σωστή.

Η συνολική σχεδίαση και οργάνωση του περιεχομένου του Ησιόδου ανταποκρίνεται στα 9 στάδια του μοντέλου του Gagne.

### Σχεδίαση Ερωτημάτων

Για τη σχεδίαση των ερωτημάτων ελήφθησαν υπόψη μία σειρά σχεδιαστικών προβληματισμών, και κατέληξα τελικά στα εξής:

- Οι ερωτήσεις κάθε επιπέδου είναι περισσότερες από αυτές που οφείλει να απαντήσει ο παίκτης και η επιλογή τους γίνεται τυχαία. Αποτέλεσμα είναι να μην λαμβάνει τις ίδιες ερωτήσεις κάθε φορά που θα ξαναπαίξει το ίδιο επίπεδο και οπωσδήποτε όχι με την ίδια σειρά.
- Οι ερωτήσεις είναι πολλαπλών επιλογών με τρεις πιθανές απαντήσεις.
- Η μία από τις παραπλανητικές επιλογές είναι αληθοφανής, ώστε να μην επιλέγει ο μαθητής το «προφανές».
- Η δεύτερη παραπλανητική επιλογή, συνήθως, είναι κάτι αστείο, και απορρίπτεται ευκολότερα, αφενός μεν γιατί το υποψήφιο κοινό είναι μικρής ηλικίας και χρειάζεται βοήθεια και αφετέρου γιατί δεν υπάρχει πιο όμορφο πράγμα από το να βλέπεις μικρά παιδάκια να χαμογελούν..!
- Η σωστή επιλογή, εννοείται πως, είναι πάντα μία από τις προτεινόμενες.
- Για κάθε ερώτηση, οι πιθανές απαντήσεις επίσης δίνονται τυχαία και όχι με την ίδια σειρά, οπότε δε μπορεί κάποιος να αποστηθίσει ότι «το σωστό είναι το πρώτο».

Και οι αποφάσεις δεν τελειώνουν ούτε εδώ, αφού για τις αρχές και επιλογές σχεδίασης, όπως αναφέρουν οι Shneiderman, Plaisant, Cohen, Jacobs, και Elmqvist για το Designing the User Interface [42] «*No list such as this can be complete*».

#### 3.2.4 Παιδαγωγικές Επιλογές

Πραγματικά, οι προβληματισμοί δεν σταματούν να εμφανίζονται σε κάθε βήμα τόσο του σχεδιασμού, όσο και της υλοποίησης ενός εκπαιδευτικού παιχνιδιού. Ακόμα κι όταν κανείς έχει κατασταλάξει στα περισσότερα, θα εμφανιστούν καινούρια διλήμματα, ιδιαιτέρως όταν το εκπαιδευτικό λογισμικό έχει τη μορφή παιχνιδιού. Πρώτο και σημαντικό σε ένα διαδραστικό πρόγραμμα είναι το δίλημμα:

μαθησιακή ικανότητα ↔ ικανότητα χειρισμού υπολογιστή

Ο μαθητής πρέπει να συνδυάσει την ικανότητα στο ηλεκτρονικό παιχνίδι και την ευχέρεια στο χειρισμό ενός ηλεκτρονικού υπολογιστή, με τη γνώση και τη μαθησιακή

του ικανότητα; Για να είναι ρεαλιστικό το παιχνίδι θα πρέπει να έχει απαιτήσεις «καλής χρήσης». Όμως δεν είναι κρίμα, να γνωρίζει ένας μαθητής την απάντηση και να τη χάσει λόγω αδυναμίας χειρισμού; Μήπως θα έπρεπε να πηγαίνει «σταθερά» πάνω στις απαντήσεις/στόχο με απλή επιλογή δεξιά-κέντρο-αριστερά;

Αποφάσισα να τα κάνω και τα δύο: να αφήσω το ενδιαφέρον του διαδραστικού παιχνιδιού που το ξεχωρίζει από τα απλά και συνηθισμένα κουίζ και να υποβοηθήσω λίγο (προγραμματιστικά) στα ανεβάσματα στα σκαλιά των απαντήσεων. Τα παιδάκια μαθαίνουν εύκολα και σύντομα θα τα καταφέρουν.

Κι αν ο καλός χειρισμός μέχρι να επιτευχθεί οδηγήσει σε πλήρη αστοχία επιλογής ενός σκαλοπατιού; Αν ο παίκτης ενώ έχει ανεβεί κάπως ψηλά, πέσει στο έδαφος; Θα τον τιμωρήσω; Να χάσει το επίπεδο, να πρέπει να το ξαναπαίξει; Όχι. Αποφάσισα, όσο ψηλά ανεβαίνει η πλατφόρμα να δημιουργούνται και βοηθητικά σκαλοπατάκια στο πλάι ώστε να έχει τη δυνατότητα ο παίκτης να ανεβεί στη θέση του –αν τυχόν και πέσει.

Τα μικρά παιδιά σφύζουν από περιέργεια. Αν κάποιο παιδάκι αφήσει την κυρίως πλατφόρμα και πηγαίνει συνέχεια δεξιά ή μόνο αριστερά για να δει «τι υπάρχει πιο πέρα», θα το αφήσω να χάνει το χρόνο του; Όχι, έβαλα απαγορευτικά σήματα στις δύο μεριές ώστε να το επαναφέρει σύντομα στο παιχνίδι.

Πολύ δύσκολη απόφαση, ίσως από τις δυσκολότερες, ήταν η σχετική με την ολοκλήρωση ενός επιπέδου: πότε θα πρέπει να τελειώσει ένα επίπεδο; Να τελειώσει στις 10 ή 12 ερωτήσεις με ό,τι βαθμολογία συγκεντρώσει ο παίκτης; Να τελειώσει στις 10 ή 12 σωστές απαντήσεις; Μήπως όταν τελειώσουν όλες οι ερωτήσεις με ό,τι βαθμολογία συγκεντρώσει συνολικά; Οι ερωτήσεις που δεν απαντώνται σωστά, να χάνονται; Μήπως να δίνεται η σωστή απάντηση ως ανάδραση; Να υπάρχει αρνητική βαθμολογία στο λάθος; Πόσο να βαθμολογείται η σωστή απάντηση;

Μετά από πολλή-πολλή σκέψη κατέληξα στα εξής: δεν υπάρχει αρνητική βαθμολογία, αλλά ούτε και δίνεται η σωστή απάντηση. Προτίμησα να ξαναχρησιμοποιείται η μη ορθά απαντημένη ερώτηση για να οδηγήσει στη μάθηση και να αναπτύξει τη μνήμη. Θεώρησα αγχώδες το να έχεις μόνο 10-12 ερωτήσεις στη διάθεσή σου και να απαντήσεις μόνο όσες ξέρεις, οπότε άφησα ελεύθερο τον παίκτη να προσπαθεί μέχρι να συμπληρώσει 12 ή 18 (στη Μυθολογία και την Προϊστορία αντίστοιχα) σωστά απαντημένες ερωτήσεις. Από τη στιγμή που επέλεξα αυτόν τον τρόπο ολοκλήρωσης επιπέδου εξυπακούεται ότι δεν θα βαθμολογούσα μόνο τις σωστές

απαντήσεις γιατί έτσι θα είχαν όλοι οι παίκτες την ίδια βαθμολογία. Οπότε κάθε σωστή ερώτηση δεν παίρνει ένα σταθερό βαθμό, αλλά υπολογίζεται το ποσοστό σωστών απαντήσεων προς το σύνολο των ερωτήσεων που τέθηκαν. Αυτό πολλαπλασιάζεται με 10.000 και αυξάνεται κατά το ήμισυ στα επίπεδα της προϊστορίας, αφού σε αυτά θα πρέπει να απαντηθούν σωστά 50% περισσότερες ερωτήσεις για να ολοκληρωθούν.

Το τι θα αντιπροσώπευε κάθε επίπεδο ήταν επίσης ένα σοβαρό θέμα. Δεν ήθελα να είναι κάθε μάθημα και ένα επίπεδο γιατί οι ερωτήσεις θα ήταν περιορισμένες και σταθερές. Έτσι χώρισα το βιβλίο σε μεγάλες ενότητες οι περισσότερες ομαδοποιημένες, που ακολουθούν την ύλη του βιβλίου αλλά δίνουν την εντύπωση γενικότερων γνώσεων· γι' αυτό άνετα μπορούν να χρησιμοποιηθούν και επαναληπτικά. Αυτό δίνει τη δυνατότητα σ' ένα παιδί, να δει πιο σφαιρικά το αντικείμενο της Ιστορίας ως γνώση και όχι αυστηρά και περιορισμένα όπως πηγαίνει το ημερήσιο μάθημα.

Οπωσδήποτε θα μπορεί να παίζει το ίδιο επίπεδο ξανά. Επανάληψη μήτηρ πάσης μαθήσεως! Αλλά τι θα γίνεται με τη βαθμολογία; Θα χάνεται; Θα διατηρείται; Θα αλλάζει; Είναι σημαντικό στο εκπαιδευτικό λογισμικό να επιτρέπουμε στο χρήστη να επαναλαμβάνει κάποια δοκιμασία αφού έτσι οδηγείται στη γνώση. Αλλά για να μείνει το ρίσκο του παιχνιδιού, η βαθμολογία του θα ξαναυπολογιστεί. Πρόκειται ξεκάθαρα για μια υποκειμενική επιλογή, που θεώρησα εγώ σωστή ώστε να μην εφησυχάζεται ποτέ κανείς. Θα μπορούσα να αλλάζω το σκορ μόνο αν το πιο πρόσφατο υπερέβαινε το προηγούμενο αλλά τότε όλα θα ήταν εκ του ασφαλούς. Ελπίζω να κάνει πιο προσεκτικούς τους παίκτες το γεγονός ότι παίζοντας μπορείς να σημειώσεις μικρότερη βαθμολογία από ό,τι είχες...!

Και αν είχε παίζει χτες ένα επίπεδο, σήμερα πρέπει να το ξαναπαίξει; Ένα πολύ σημαντικό χαρακτηριστικό που έχει το παιχνίδι μου και που αποφάσισα μετά από προβληματισμό, είναι η μνήμη του! Λόγω του όγκου και της έκτασής του, δεν είναι λογικό να αναμένει κανείς να τελειώσουν όλα τα επίπεδα σε μία φορά. Ούτε είναι σωστό να μην ξέρει κανείς ποια ολοκλήρωσε και ποια είναι πρόοδός του. Οπότε χάρισα στον Ησίοδο «μνήμη». Αν ένας χρήστης έχει παίζει με το όνομά του έχοντας έναν υποτυπώδη λογαριασμό, το παιχνίδι τον θυμάται, ξέρει την πρόοδό του και του τη δείχνει από την είσοδό του και στη συνέχεια σε κάθε επίπεδο.

Τέλος, επέτρεψα την επιλογή ενοτήτων και επιπέδων κατά βούληση. Δεν υπάρχουν προαπαιτούμενα ούτε υποχρεωτική σειρά ολοκλήρωσης επιπέδων. Ο παίκτης μπορεί

να επιλέξει όποιο επίπεδο θέλει και όποτε θέλει. Έτσι του δίνεται η δυνατότητα να επιχειρήσει να μάθει από το παιχνίδι και όχι μόνο να ελέγχει τις γνώσεις του.

Αυτά και άλλα διλήμματα αντιμετώπισα καθ' όλη τη διάρκεια της ανάπτυξης του παιχνιδιού μου και σε όλες τις φάσεις, από το σχεδιασμό μέχρι την αξιολόγησή του, συμπεριλαμβανομένου του αν θα είναι διαδικτυακό ή αν θα παίζεται τοπικά στον υπολογιστή. Επέλεξα να υλοποιήσω και τις δύο εκδοχές!

Μετά από όλο αυτό τον αγώνα έφτασα σε σημείο να μπορώ να δηλώσω πως: το λογισμικό μου δε βασίζεται στο κλασικό διδακτικό μοντέλο, που περιλαμβάνει

- μελέτη συγκεκριμένου περιεχομένου και
- άσκηση για την εκμάθηση του περιεχομένου

αλλά σε μορφή παιχνιδιού με υψηλή διάδραση και μάθηση που μπορεί να επέλθει με τη συνεχόμενη διασκεδαστική επανάληψη ή και τον πειραματισμό.

### **3.2.5 Περιγραφή του Παιχνιδιού**

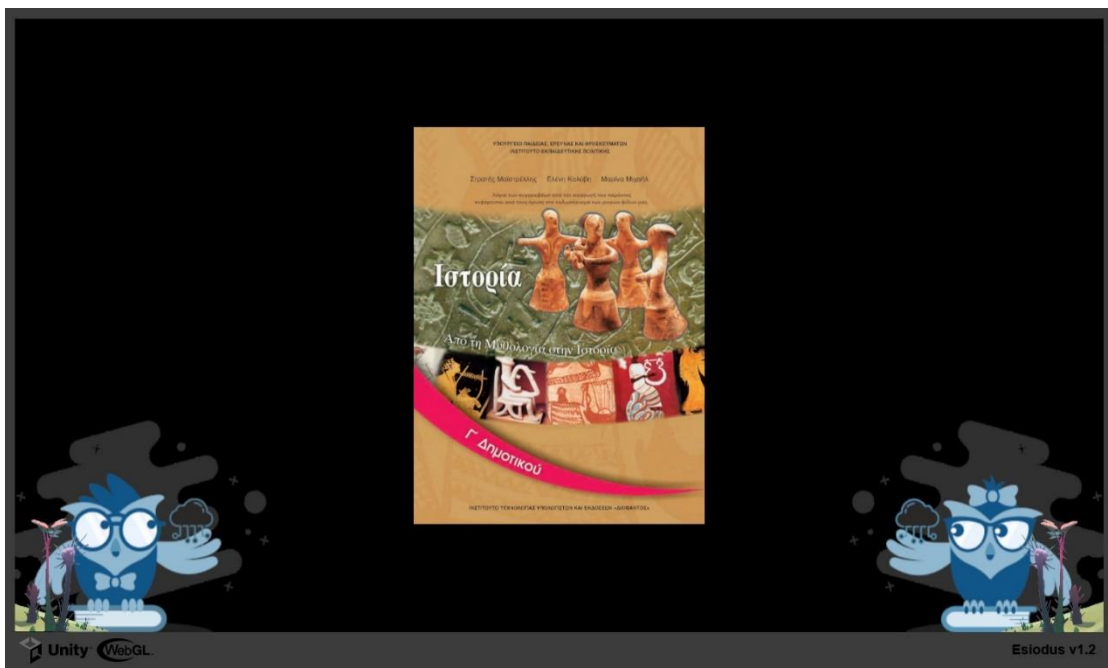
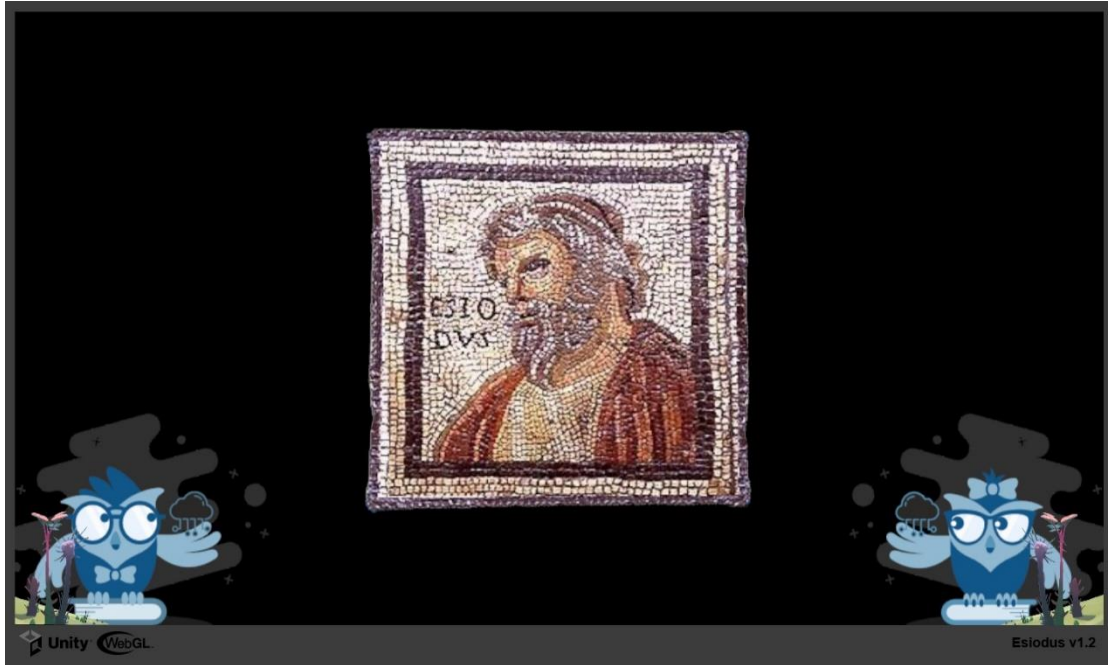
Μετά από όλες τις επιλογές και αποφάσεις, ο Ησίοδος έγινε πραγματικότητα. Αποτελείται από οχτώ οθόνες οι οποίες, στην πραγματικότητα, υπάρχουν για καλύτερη και απλούστερη υλοποίηση. Κάποιες θα μπορούσαν να συμπτυχθούν χρησιμοποιώντας επίπεδα ή την ορατότητα κάποιων καμβάδων –όπως συμβαίνει στην οθόνη επιλογής επιπέδου. Προτίμησα, όμως, αυτή τη μορφή ώστε να μπορεί ευκολότερα να παρακολουθηθεί το πρόγραμμά μου από κάποιον τρίτο και να παρουσιάζει τα χαρακτηριστικά του δομημένου προγραμματισμού. Έτσι, οι οθόνες ακολουθούν σε γενικές γραμμές τη λογική με την οποία θα ήταν ορατές ούτως ή άλλως.

#### **1<sup>η</sup> Οθόνη: Έναρξη**

Με την έναρξη του παιχνιδιού η Unity έχει μια εισαγωγή όπου αναφέρεται ότι το παρόν πρόγραμμα έγινε με αυτήν. Από το σημείο αυτό που αναλαμβάνω εγώ, εμφανίζεται αναδυόμενη, η εικόνα του Ησίοδου, το όνομα του οποίου φέρει το παιχνίδι και χρησιμοποιείται και ως εικονίδιο της τοπικής αυτόνομης εκδοχής του. Ακολουθεί το επίσημο εγχειρίδιο της Ιστορίας της Γ' Τάξης Δημοτικού του ελληνικού σχολείου, για να υπάρξει η άμεση σύνδεση του παιχνιδιού με το τι πραγματεύεται. Στο εξώφυλλο του εγχειριδίου έχω κάνει μια μικρή παρέμβαση για να αναφέρω ότι χρησιμοποιώ κάποια από τα λόγια των συγγραφέων του από την εισαγωγή του βιβλίου ως λόγια των ηρώων της επόμενης οθόνης (Εικόνα 7).



Επειδή για το διαδικτυακό παιχνίδι υπάρχει ολόκληρη ιστοσελίδα, στόλισα την τοπική έκδοση, με μία επιπλέον εικόνα κατά το άνοιγμα. Είναι οι ήρωες της εφαρμογής, που παρουσιάζονται μαζί με την υπογραφή της παιχνιδομηχανής.



Εικόνα 7: Ησίοδος - Εναρκτήριες Οθόνες

## 2<sup>η</sup> Οθόνη: Γνωριμία με τους Φίλους



Εικόνα 8: Ησίοδος - Οθόνη Γνωριμίας με τους Φίλους

Ακολουθεί η οθόνη γνωριμίας με τους ήρωες που μπορούν να βοηθήσουν τον παίκτη, παίζοντας για αυτόν το παιχνίδι. Οι ήρωες εξηγούν στα παιδιά ποιος ήταν ο Ησίοδος και χρησιμοποιώντας και λόγια των συγγραφέων από την εισαγωγή του βιβλίου της Ιστορίας, προλογίζουν ή αναγγέλλουν το περιεχόμενο του παιχνιδιού και το σκοπό του. Ταυτόχρονα εμφανίζονται τα ονόματά τους για να επέλθει η πρώτη γνωριμία με τον παίκτη.

Τηρώντας τις αρχές που πρέπει να διέπουν το σχεδιασμό και την υλοποίηση των εκπαιδευτικών λογισμικών, από αυτήν κιόλας την οθόνη, υπάρχει η δυνατότητα εξόδου από το παιχνίδι καθώς και η παράλειψη της εισαγωγικής αυτής οθόνης. Μετά την πρώτη φορά που θα την παρακολουθήσει κάποιος, δεν είναι υποχρεωτικό να ακούει ή να διαβάζει όλη την εισαγωγή από την αρχή μέχρι το τέλος. Φυσικά υπάρχει η δυνατότητα να επιτρέψει ή όχι τον ήχο, ανοιγοκλείνοντας τη φωνή, όπως γίνεται στα περισσότερα παιχνίδια.

## 3<sup>η</sup> Οθόνη: Εισαγωγή Στοιχείων Χρήστη

Τρίτη στη σειρά, έρχεται η οθόνη που αφορά στην εισαγωγή στοιχείων του παίκτη (Εικόνα 9). Το αν όντως θα βάλει τα στοιχεία του ή όχι είναι επιλογή του. Πρόκειται για μια ακόμη ευελιξία του παιχνιδιού: ο χρήστης δεν είναι υποχρεωτικό να έχει λογαριασμό, ούτε να παίζει πάντα επώνυμα. Γενικότερα υπάρχουν οι δυνατότητες:



**Εικόνα 9: ΗΣίοδος - Οθόνη Εισαγωγής Στοιχείων**

- μπορεί να πατήσει κατευθείαν το κουμπί «αποθήκευση στοιχείων», παίζοντας ως ανώνυμος παίκτης, βλέποντας μόνο τη βαθμολογία της συγκεκριμένης συνεδρίας (session), χωρίς αποθήκευση της προόδου του
- αν είναι νέος παίκτης, δηλαδή παίζει πρώτη φορά, μπορεί να βάλει το όνομά του στο διαδικτυακό παιχνίδι και αυτό θα του δώσει έναν αριθμητικό κωδικό με τον οποίο θα μπορεί να παίζει κάθε φορά, διατηρώντας και παρακολουθώντας την πρόοδο και τη βαθμολογία του
- αν έχει ξαναπαίξει επώνυμα και βάλει σωστά τα στοιχεία του, θα συνεχίσει από το σημείο που είχε φτάσει την προηγούμενη φορά βλέποντας την πρόδό του και μάλιστα θα ενημερωθεί ήδη από αυτή την οθόνη για την προηγούμενη βαθμολογία του.

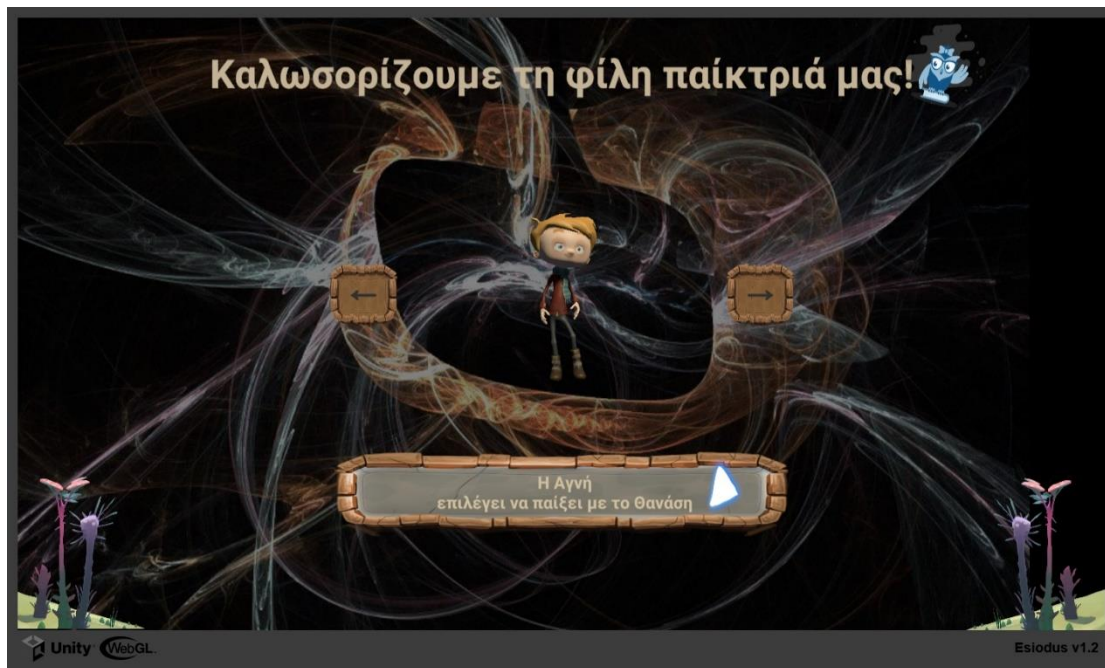
Αντίστοιχες δυνατότητες υπάρχουν στην τοπική έκδοση, με τη διαφορά ότι ο παίκτης μόνος του αποφασίζει και για τον κωδικό με τον οποίο θα παίζει στον δικό του υπολογιστή.

#### **4<sup>η</sup> Οθόνη: Επιλογή Ήρωα**

Στην τέταρτη κατά σειρά οθόνη, ο παίκτης επιλέγει με ποιον φίλο θα παίζει το παιχνίδι, ποιος θα τον αντιπροσωπεύσει... Με τα βελάκια περιηγείται στους πέντε διαθέσιμους παίκτες, οι οποίοι τον καλούν να τους επιλέξει με έναν ξεχωριστό χορό ο

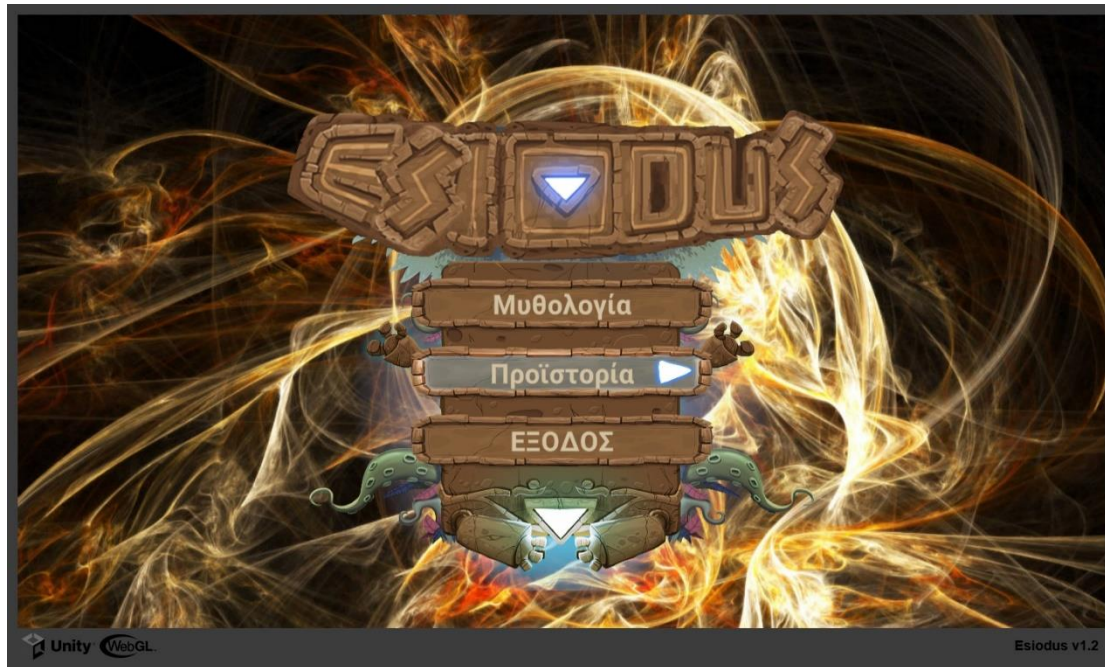


καθένας τους (Εικόνα 10). Η επιλογή γίνεται απλά με την επιβεβαίωση – πάτημα του κουμπιού «[Ο Παίκτης] επιλέγει να παίξει με [τον Φίλο]».



Εικόνα 10: ΗΣΙΟΔΟΣ - Οθόνη Επιλογής Παίκτη

### 5<sup>η</sup> Οθόνη: Κεντρικό Μενού - Επιλογή Ενότητας



Εικόνα 11: ΗΣΙΟΔΟΣ - Οθόνη Κεντρικού Μενού

Στο σημείο αυτό, ο παίκτης έχει τη δυνατότητα να επιλέξει μία από τις δύο μεγάλες ενότητες του μαθήματος της Ιστορίας, τη Μυθολογία ή την Προϊστορία. Ο διαχωρισμός δεν είναι ουσιαστικός προγραμματιστικά, ωστόσο υπάρχει περισσότερο

για να διευκολύνει τους μικρούς υποψήφιους παίκτες και να δείχνει το παιχνίδι διακριτό και ξεκάθαρο στα μάτια τους. Στην πραγματικότητα οδηγεί στην ίδια οθόνη, την 6<sup>η</sup>, η οποία ανάλογα με την επιλογή του παίκτη σε αυτό το σημείο έχει απλά άλλη εμφάνιση.

### 6<sup>η</sup> Οθόνη: Μενού Επιλογής Επιπέδου



Εικόνα 12: Ησιόδος - Οθόνη Επιλογής Επιπέδου (Μυθολογία)

Εάν ο παίκτης επέλεξε να παίξει με τη μυθολογία, θα αντικρύσει μπροστά του την οθόνη της παραπάνω εικόνας. Ο αραχνοειδής χάρτης αντιπροσωπεύει τα διάφορα επίπεδα-κεφάλαια της μυθολογίας. Είναι μια αναπαράσταση επιπέδων που είναι οικεία στους μικρούς μαθητές από άλλα γνωστά σε αυτούς διαδικτυακά παιχνίδια: δεν επιβάλλει ολοκλήρωση προηγούμενων επιπέδων για την κατάκτηση νέων. Για το λόγο αυτό δεν είναι σειριακή η απεικόνισή τους και όλα τα επίπεδα είναι ενεργά. Αν μάλιστα ο παίκτης έχει κατακτήσει κάποια, αυτά έχουν ένα ιδιαίτερο μπορντώ/καφέ περίγραμμα ώστε να ξέρει ποια έχει παίξει και ποια του μένουν ακόμα. Με το πέρασμα του ποντικιού πάνω από τον κάθε κόμβο, εντείνεται το χρώμα της εικόνας που ετοιμάζεται να ενεργοποιηθεί, ενώ εμφανίζεται ως υπόδειξη (hint) το όνομα του επιπέδου που αντιπροσωπεύει το συγκεκριμένο εικονίδιο.

Εάν ο παίκτης επέλεξε να παίξει με την προϊστορία, τότε θα δει μπροστά του την οθόνη της Εικόνας 13, με όλα τα χαρακτηριστικά της προηγούμενης εκδοχής. Τα

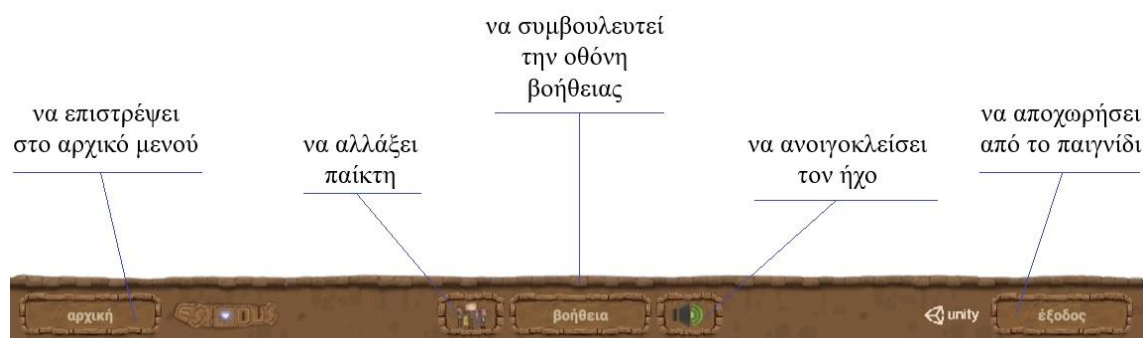


επίπεδα εδώ είναι λιγότερα αλλά περιλαμβάνουν περισσότερες ερωτήσεις, με ανάλογη επίδραση στην τελική βαθμολογία του παίκτη.



Εικόνα 13: Ησίοδος - Οθόνη Επιλογής Επιπέδου (Προϊστορία)

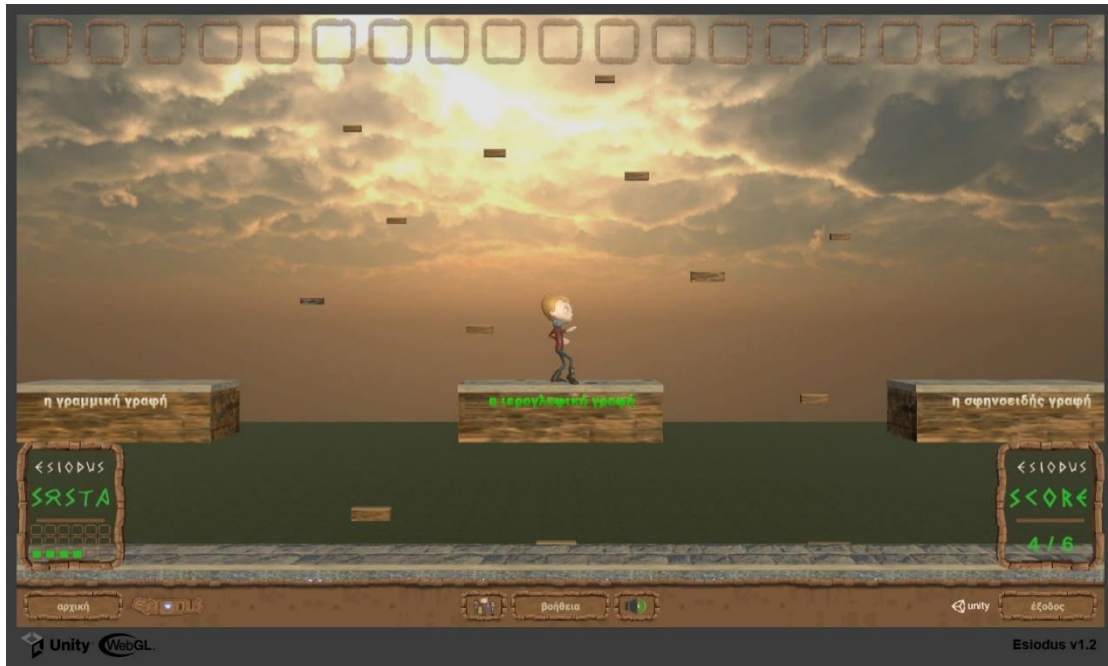
Στο σημείο αυτό, όπως και σε κάθε επόμενο, ο παίκτης χρησιμοποιώντας το μενού που βρίσκεται στο κάτω μέρος της οθόνης μπορεί:



### 7<sup>η</sup> Οθόνη: Κεντρική Οθόνη Παιχνιδιού

Στην 7<sup>η</sup> οθόνη παίζονται όλα τα επίπεδα του παιχνιδιού. Κάθε επίπεδο έχει την ίδια ακριβώς μορφή, αλλά το δικό της διαφορετικό υπόβαθρο (background) ώστε να φαίνεται όντως διαφορετικό και να διακρίνεται από τα υπόλοιπα. Έχει το ίδιο μενού επιλογών στο κάτω μέρος και πληροφορίες για την πρόοδο του παίκτη στο επίπεδο, αλλά και στο παιχνίδι. Κάτω δεξιά φαίνονται οι απαντήσεις που έχει δώσει ο παίκτης στο σύνολο των ερωτήσεων που έχουν παιχτεί στο επίπεδο, ενώ κάτω αριστερά φαίνονται κουτάκια που αντιστοιχούν στις σωστές απαντήσεις που πρέπει να

απαντηθούν. Μόλις δίνεται μια σωστή απάντηση, ένα κουτάκι γίνεται πράσινο, οπότε ο παίκτης βλέπει πόσες ακόμα πρέπει να απαντήσει προκειμένου να ολοκληρώσει το επίπεδο (Εικόνα 14).

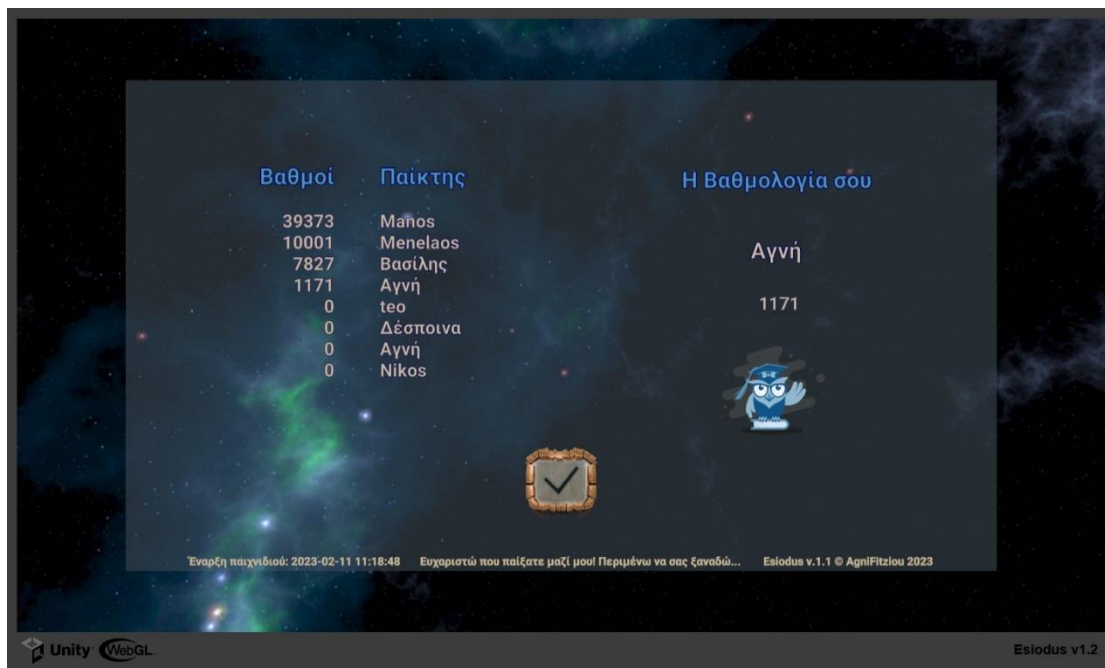


Εικόνα 14: Ησίοδος - Κυρίως Οθόνη Παιχνιδιού

Ψηλά στην οθόνη φαίνεται αν έχει κερδίσει κάποια από τα επίπεδα αποκτώντας τα σήματά τους ως επιβράβευση. Μόλις ολοκληρώνεται ένα επίπεδο, ο παίκτης οδηγείται

πίσω στο μενού επιλογής επιπέδου της ενότητας που είχε επιλέξει πριν, προκειμένου να συνεχίσει.

### 8<sup>η</sup> Οθόνη: Βαθμολογία



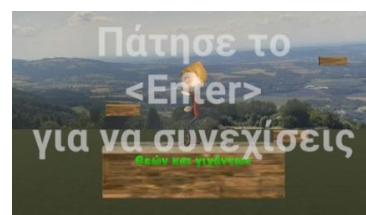
Εικόνα 15: Ησιόδος - Οθόνη Τερματισμού

Όταν ο παίκτης επιλέξει να αποχωρήσει από το παιχνίδι, οδηγείται στην τελευταία οθόνη. Εδώ ενημερώνεται για τη βαθμολογία που έχει επιτύχει ο ίδιος και αν είναι στους δέκα καλύτερους παίκτες, το όνομά του γράφεται στον πίνακα των καλύτερων παικτών. Προϋπόθεση, βέβαια, για να γραφτεί κανείς στον πίνακα των καλύτερων είναι να παίζει επώνυμα, διαφορετικά απλά βλέπει το επίτευγμά του χωρίς να αποθηκεύεται κάπως η πρόοδός του.

### Οθόνες Βοήθειας

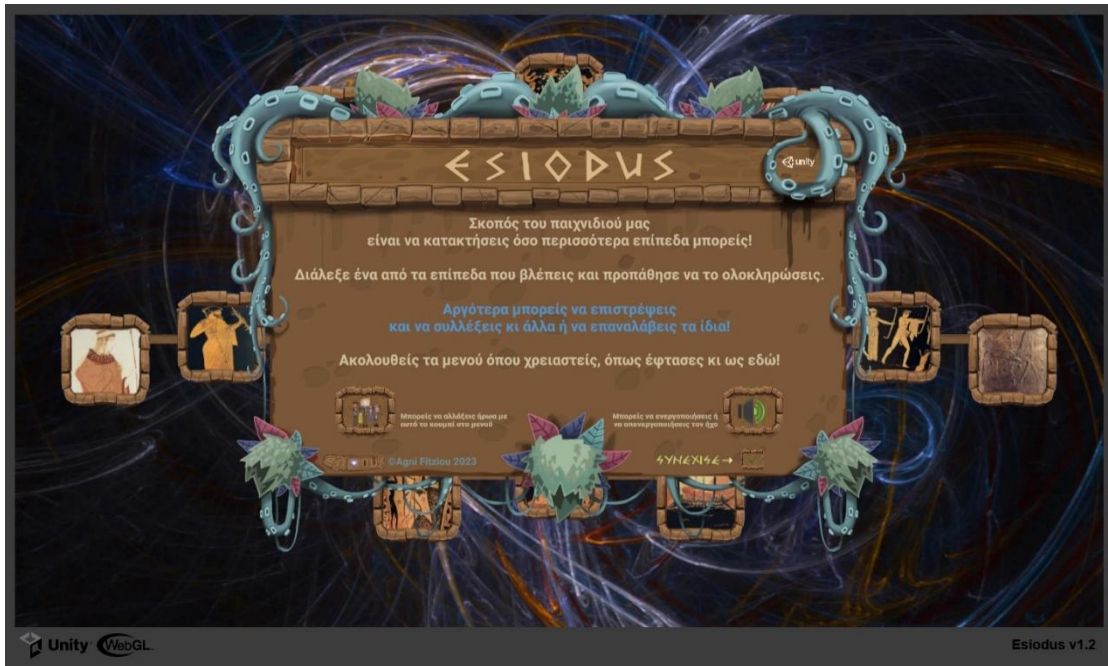
Αν ο παίκτης χρειαστεί βοήθεια οποιαδήποτε στιγμή, μπορεί να πάρει σύντομη βοήθεια πατώντας το κουμπί του μενού «Βοήθεια» (Εικόνα 16). Οι οθόνες βοήθειας περιλαμβάνουν οδηγίες αλλά και animation των πλήκτρων και των κουμπιών στα οποία αναφέρονται. Αναλυτικότερες οδηγίες για το πώς παίζεται το παιχνίδι βρίσκονται στην αρχική σελίδα του: [esiodus.eu](http://esiodus.eu), μέσα από την οποία παίζεται.

Σε ένα σημείο που απαιτείται η χρήση του πλήκτρου Enter, η οδηγία εμφανίζεται αναδυόμενη, δύο φορές σε κάθε επίπεδο, γιατί είναι πολύ πιθανό να μην ψάξει ο παίκτης τι πρέπει να κάνει και να θεωρήσει ότι πρόκειται





για αστοχία του παιχνιδιού.



Εικόνα 16: ΗΣΙΟΔΟΣ - Βοηθητικές Οθόνες

### 3.3 UML: Μια εικόνα, χίλιες λέξεις

Αφού παρουσιάστηκε η διαδικασία ανάπτυξης του Ησίοδου, από τους σχεδιαστικούς προβληματισμούς μέχρι την υλοποίησή του ως παιχνίδι, ήρθε η ώρα να περιγραφεί και ως πρόγραμμα. Ακολουθεί, λοιπόν, παρουσίαση του προγράμματος του παιχνιδιού Ησίοδος, τόσο με διαγράμματα, όσο και με περιγραφή των βασικών λειτουργιών του κώδικά του, μέσω κλάσεων. Αρχικά ας γνωρίσουμε τη UML.

#### 3.3.1 UML: Τι είναι

Η UML (Unified Modeling Language) είναι μια τυπική γλώσσα για τον προσδιορισμό, την απεικόνιση, την κατασκευή και την τεκμηρίωση των τεχνουργημάτων των συστημάτων λογισμικού [45]. Προτάθηκε στην OMG ( Object Management Group ) τον Ιανουάριο του 1997, ενώ καταβάλλοντας συνεχώς προσπάθειες για τη δημιουργία ενός πραγματικά βιομηχανικού προτύπου, μετά από κάποια τυποποίηση, κατάφερε να γίνει ένα πρότυπο OMG. Δεν είναι προγραμματιστική γλώσσα όπως η C++, η C#, η Python ή η Java αλλά είναι μια περιγραφική γλώσσα λογισμικού. Προσφέρει τη δυνατότητα οπτικής μοντελοποίησης λογισμικού, μέσω σχεδιαγραμμάτων, για τον προσδιορισμό και την τεκμηρίωσή του, πιθανά και την ίδια την κατασκευή του.

Η UML έχει άμεση σχέση με την αντικειμενοστραφή ανάλυση και περιγραφή, οπότε δεν θα μπορούσε να απουσιάζει από την περιγραφή ενός σύγχρονου παιχνιδιού, επειδή:

- είναι φτιαγμένη όχι μόνο για προγραμματιστές αλλά και για οποιονδήποτε ενδιαφέρεται να κατανοήσει ένα σύστημα
- βοηθά, από απλούς χρήστες έως επαγγελματίες, στην κατανόηση των οντοτήτων του λογισμικού στον πραγματικό κόσμο και στο πώς αυτές αλληλεπιδρούν μεταξύ τους.

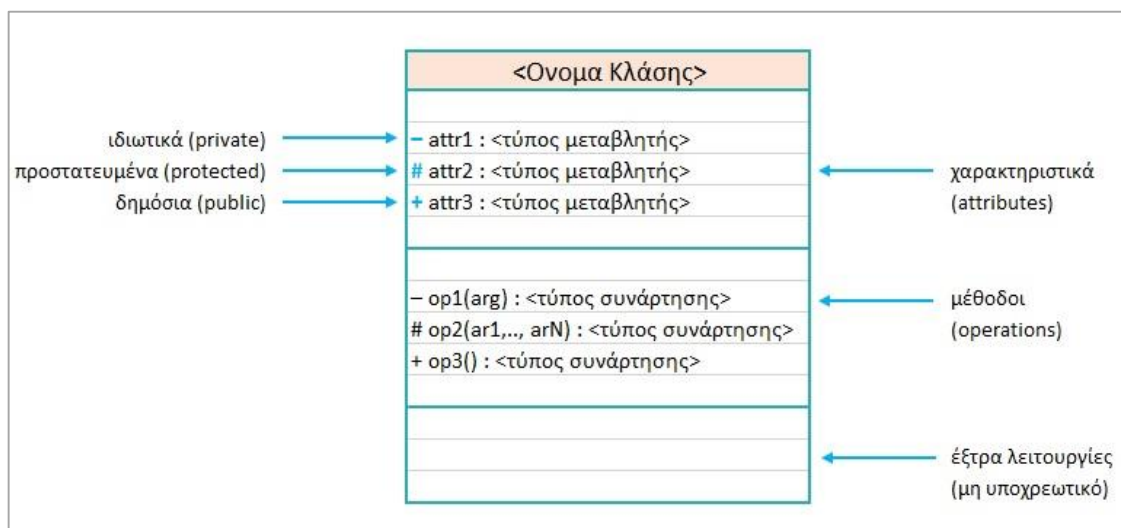
Ο πιο σημαντικός στόχος της UML είναι η παρουσίαση του κάθε συστήματος λογισμικού με τρόπο που να είναι εύχρηστος, απλός και κατανοητός από όλους τους μοντελιστές και τους ενδιαφερόμενους· και αυτός ο τρόπος δεν θα μπορούσε να είναι άλλος από το διαγραμματικό. Η UML είναι εδώ, να ταιριάζει απόλυτα την περιγραφή της με το απόφθεγμα «μια εικόνα αξίζει όσο χίλιες λέξεις»!

### 3.3.2 Σημειογραφίες UML

Οι συμβολισμοί που χρησιμοποιεί η UML είναι τα πιο σημαντικά στοιχεία της στη μοντελοποίηση. Αν δεν υπάρχει αποτελεσματική και κατάλληλη χρήση σημειογραφιών δεν είναι δυνατή η δημιουργία ενός πλήρους και ουσιαστικού μοντέλου. Για να είναι το μοντέλο χρήσιμο, πρέπει να απεικονίζεται σωστά και ξεκάθαρα.

Στην περιγραφή που ακολουθεί θα χρησιμοποιήσω όλη τη βασική σημειογραφία της UML, χρησιμοποιώντας λίγο πιο γραφικά στοιχεία στις αναπαραστάσεις των παικτών και των υλικών του συστήματος, επειδή δεν αλλάζουν κάτι στην ουσιαστική μοντελοποίηση.

- Η κλάση, όπως φαίνεται στην Εικόνα 17, περιγράφεται με ένα παραλληλόγραμμο με 3 ή (προαιρετικά) 4 μέρη. Το επάνω μέρος περιέχει το όνομα της κλάσης, το αμέσως επόμενο, τα χαρακτηριστικά της, με κατάλληλη σήμανση της ορατότητας - προσβασιμότητάς τους από άλλους (visibility) και το τρίτο μέρος περιλαμβάνει τις μεθόδους της κλάσης με αντίστοιχη σήμανση. Το προαιρετικό τελευταίο μέρος χρησιμοποιείται για την περιγραφή πρόσθετων λειτουργιών που αναλαμβάνει η κλάση.



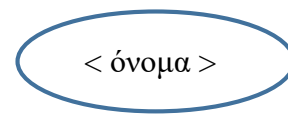
Εικόνα 17: Σημειογραφία UML για την κλάση αντικειμένων

- Τα αντικείμενα (ως στιγμιότυπά τους) περιγράφονται ακριβώς όπως και οι κλάσεις.

- Οι υποκλάσεις δηλώνονται με ένα βέλος με φαρδιά μύτη, στο οποίο η μύτη δείχνει την υπερκλάση.
- Η εξάρτηση μιας οντότητας από μία άλλη συμβολίζεται με διακεκομμένο βέλος, η μύτη του οποίου δείχνει την ανεξάρτητη οντότητα, ενώ πολλές φορές πάνω στο βέλος μπορεί να φαίνεται το στοιχείο της εξάρτησης.
- Οι οντότητες περιγράφονται με ελλείψεις: στο κέντρο υπάρχει το όνομά τους.
- Κάθε εσωτερική ή εξωτερική οντότητα που αλληλεπιδρά με το σύστημα συμβολίζεται με ένα ανθρωπάκι.
- Τέλος, οι κόμβοι που αναπαριστούν φυσικό μέρος ενός συστήματος όπως ένας διακομιστής, το δίκτυο κλπ, τυπικά, παρουσιάζονται ως τετράγωνα κουτιά.

υποκλάση  υπερκλάση

εξαρτημένη  ανεξάρτητη



### 3.3.3 Το Διάγραμμα UML

Όλη η παραπάνω σημειογραφία και οι διάφορες συνιστώσες της UML έχουν ως στόχο την παρουσίαση μιας ολοκληρωμένης εικόνας του συστήματος, γνωστής ως «διάγραμμα». Μπορούν να συνδυαστούν με διάφορους τρόπους και να παρουσιάσουν επιμέρους πτυχές του περιγραφόμενου συστήματος. Στην πραγματικότητα, το UML Διάγραμμα απαρτίζεται από διάφορα άλλα διαγράμματα που επεξηγούν το μοντέλο.

Αρχικά τα διαγράμματα διακρίνονται σε δομικά (Structural) και διαγράμματα συμπεριφοράς (Behavioral).

Τα δομικά διαγράμματα, όπως καταμαρτυρεί και το όνομά τους περιγράφουν τη δομή του συστήματος, δηλαδή τη στατική πτυχή του. Κλάσεις, διεπαφές, αντικείμενα,

συνιστώσες και κόμβοι που είναι σταθερά, αποτελούν τις στατικές όψεις που περιγράφονται με τα αντίστοιχα διαγράμματα συνθέτοντας την κύρια δομή. Αυτά είναι τα διαγράμματα:

- Κλάσης
- Αντικειμένου
- Συνιστωσών
- Ανάπτυξης

Ένα σύστημα, εκτός από τη στατική του δομή έχει και τη δυναμική του πτυχή. Υπεύθυνα για την περιγραφή των μεταβαλλόμενων μερών του συστήματος είναι τα διαγράμματα συμπεριφοράς, δηλαδή τα διαγράμματα:

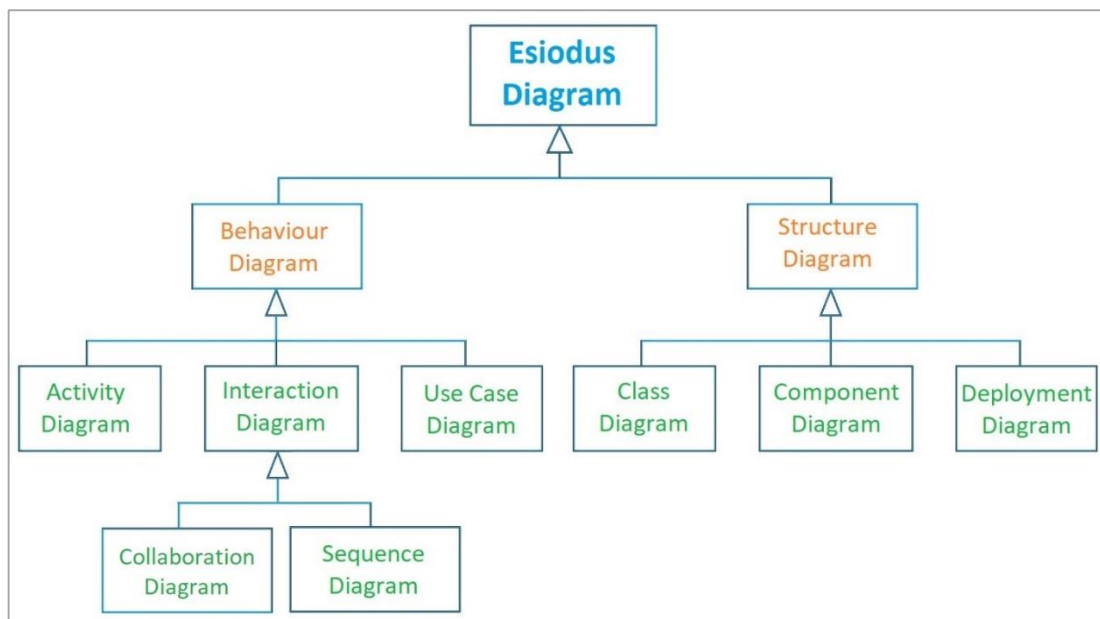
- Περιπτώσεων Χρήσης
- Ακολουθίας
- Συνεργασίας
- Καταστάσεων
- Δραστηριότητας

Όταν καλυφθούν και οι δύο όψεις, στατική και δυναμική, το μοντέλο μπορεί να θεωρηθεί ολοκληρωμένο.

Δεδομένου ότι

- το διάγραμμα Συνεργασίας (Collaboration Diagram) είναι παρόμοιο με αυτό της Ακολουθίας· και
- στα διαγράμματα Καταστάσεων (Statechart Diagrams) δίνεται έμφαση στις αλλαγές κατάστασης σε ορισμένα εσωτερικά ή εξωτερικά γεγονότα και στην περίπτωση του ΗΣίοδου, θα καλυφθεί επίσης από το διάγραμμα Δραστηριοτήτων,

το λογισμικό σύστημα του ΗΣίοδου θα περιγραφεί με τα ακόλουθα διαγράμματα:



Εικόνα 18: Διάγραμμα UML ( Esiodus )

### 3.3.4 Διαγράμματα Συμπεριφοράς – Behavioral Diagrams

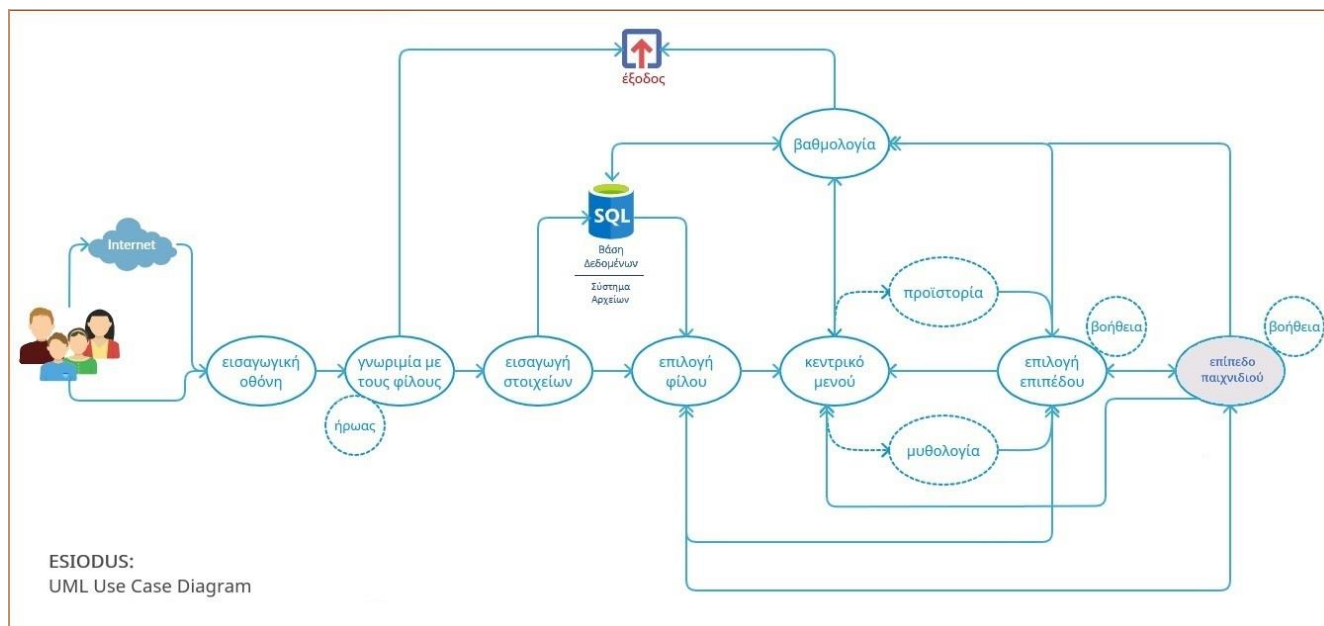
Θα ξεκινήσω την περιγραφή μου από τα γενικότερα διαγράμματα και θα προχωρήσω προς την ανάλυση των συστατικών, κάνοντας μια παρουσίαση από την κορυφή προς τη βάση, καταλήγοντας έτσι σιγά-σιγά στον κώδικα του λογισμικού μου.

#### Use Case Diagram

Τα Διαγράμματα Περιπτώσεων Χρήσης προβάλλουν τη χρήση του συστήματος. Περιλαμβάνουν το σύνολο των περιπτώσεων χρήσης, τους ελεγκτές των εσωτερικών και εξωτερικών λειτουργιών, που αναφέρονται ως ηθοποιοί/δράστες (actors) και των σχέσεών τους. Κάθε συγκεκριμένη λειτουργικότητα του συστήματος παρουσιάζεται σαν ένα μαύρο κουτί με γνωστή μόνο την είσοδο και την έξοδο, χωρίς να περιγράφεται πώς υλοποιείται.

Στην Εικόνα 19 βλέπουμε το Διάγραμμα Περιπτώσεων Χρήσης του ΗΣίοδου. Ο actor του ΗΣίοδου είναι φυσικά ο χρήστης/παίκτης. Αυτός αποφασίζει αρχικά, αν θα παίξει το παιχνίδι διαδικτυακά ή τοπικά στον υπολογιστή του. Σε κάθε περίπτωση, πρώτη είναι η Εισαγωγική Οθόνη του προγράμματος, που οδηγεί άμεσα στην οθόνη Γνωριμίας με τους Φίλους. Στο σημείο αυτό μπορεί να εγκαταλείψει το παιχνίδι ή να συνεχίσει. Επόμενη περίπτωση χρήσης είναι η Εισαγωγή των Στοιχείων του.





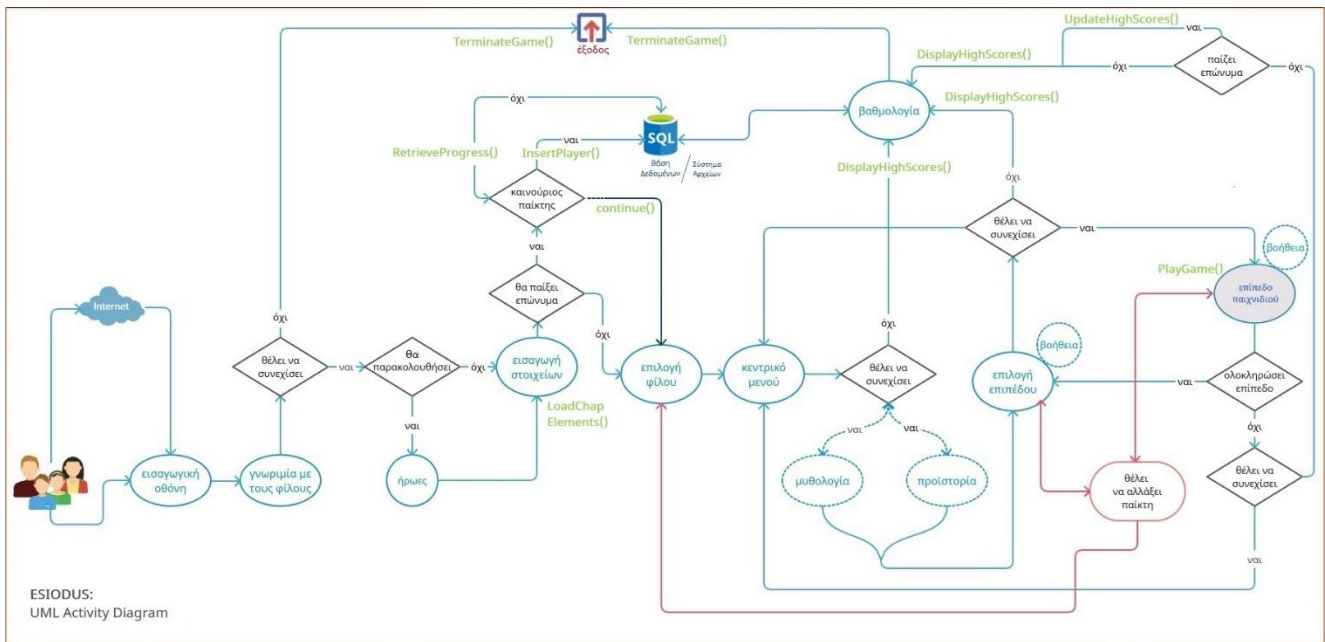
**Εικόνα 19: UML Διάγραμμα Περιπτώσεων Χρήσης**

Συνδεδεμένο το σύστημα με τη βάση δεδομένων του ή όχι, ο παίκτης προχωράει στην Επιλογή του Φίλου με τον οποίο θα παίζει, ενώ μετά από αυτή την απόφαση φτάνει στο Κεντρικό Μενού υποχρεωτικά. Από το σημείο αυτό ο actor ελέγχει πλήρως το παιχνίδι αποφασίζοντας αν θα παίζει, ποιο κεφάλαιο θα επιλέξει και ποιο επίπεδο σ' αυτό, αν θα ξαναπαίξει, αν θα αλλάξει παίκτη ή αν θα αποχωρήσει σε οποιοδήποτε βήμα από τα παραπάνω. Στην Έξοδο φτάνει αφού δει τις δέκα υψηλότερες βαθμολογίες και τη δική του.

### Activity Diagram

Η αποτύπωση της δυναμικής ροής ενός συστήματος δεν είναι εύκολο να αποτυπωθεί. Το Διάγραμμα Δραστηριότητας έχει αυτό το ρόλο προσπαθώντας να περιγράψει τη ροή του ελέγχου στο σύστημα. Αποτελείται από τις λειτουργίες του συστήματος, τις λεγόμενες δραστηριότητες και τους συνδέσμους που τις συσχετίζουν. Η ροή ως δυναμική, μπορεί να είναι διαδοχική, ταυτόχρονη ή διακλαδισμένη.

Η οπτικοποίηση της ροής των στοιχείων ελέγχου σε ένα σύστημα γίνεται με σκοπό να υπάρχει μια ιδέα για το πώς θα λειτουργεί το σύστημα όταν εκτελεστεί και στην Εικόνα 20 βλέπουμε πώς θα λειτουργήσει ο Ησίοδος στην πραγματικότητα.



**Εικόνα 20: UML Διάγραμμα Δραστηριότητας**

Όπως είδαμε και στο διάγραμμα περιπτώσεων χρήσης, ο παίκτης είναι αυτός που αποφασίζει αν θα παίξει διαδικτυακά ή τοπικά. Αυτή είναι μια απόφαση που επηρεάζει κάποιες από τις μετέπειτα ενέργειες, κυρίως στο αν το σύστημα θα συνδεθεί με Server για την ανάκτηση στοιχείων της βάσης του Ησιόδου ή αν οι ενέργειες που θα γίνουν, έχουν να κάνουν με το τοπικό σύστημα αρχείων στον υπολογιστή του χρήστη.

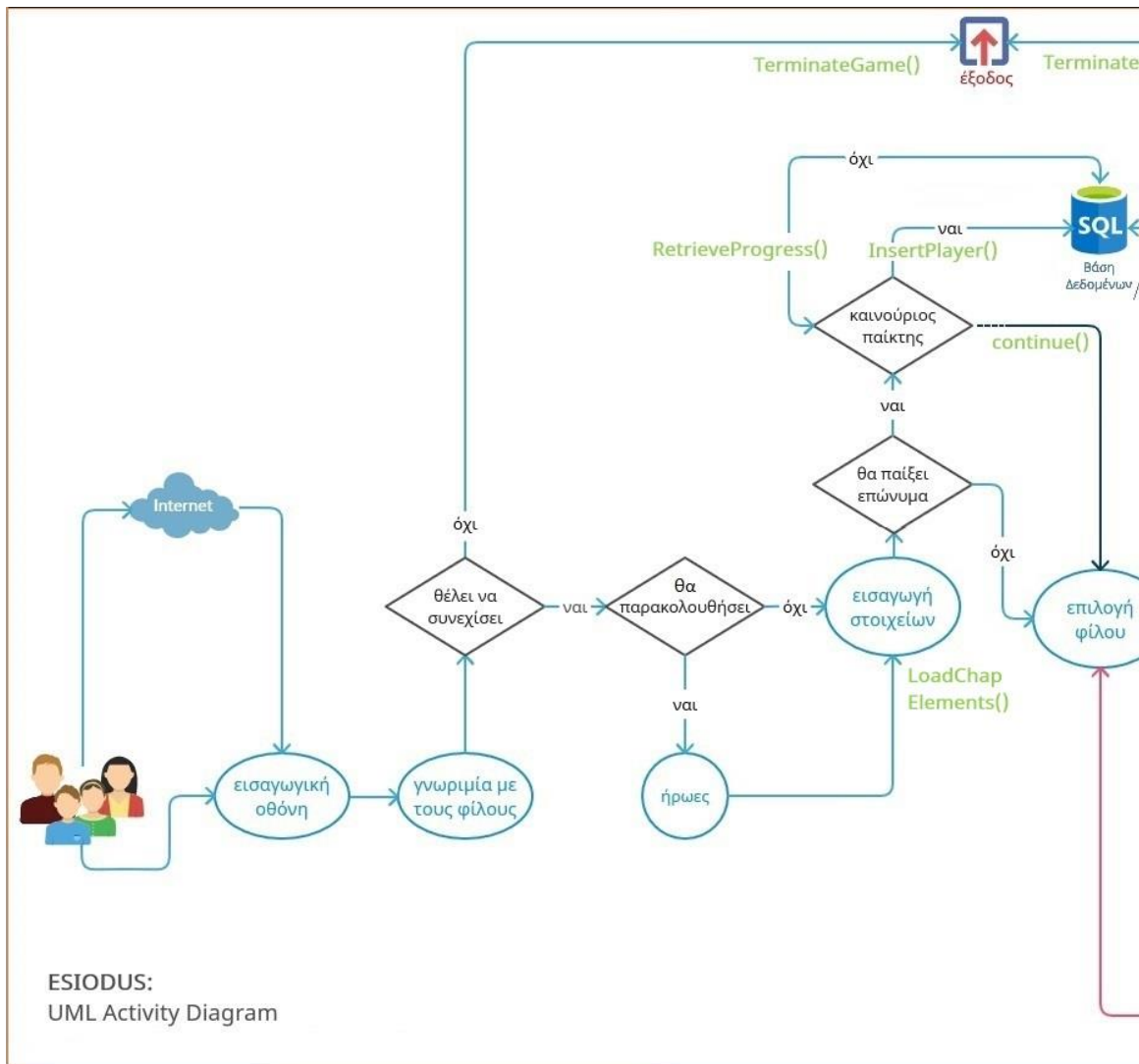
Πρώτη εμφανίζεται στον παίκτη η Εισαγωγική Οθόνη του προγράμματος, μια αναφορά μέσω εικόνων του Ησιόδου και του βιβλίου της Ιστορίας που πραγματεύεται το παιχνίδι. Η οθόνη Γνωριμίας με τους Φίλους έρχεται αμέσως μετά. Ο παίκτης έχει τη δυνατότητα, ήδη από αυτή την οθόνη, να φύγει από το παιχνίδι επιλέγοντας την Έξοδο, βάζοντας σε ενέργεια τη διαδικασία `TerminateGame()`, όπως φαίνεται καθαρότερα στην Εικόνα 21. Αν δεν το πράξει, μπορεί να παρακολουθήσει την εισαγωγή όπου μιλάνε οι πέντε φίλοι και να τους γνωρίσει ή να παραλείψει αυτό το βήμα και να προχωρήσει στην εισαγωγή των στοιχείων του.

Επιλογή του είναι αν θα παίξει ανώνυμα, οπότε προχωράει στην Επιλογή του Φίλου που θα τον συντροφεύσει ή αν θα παίξει επώνυμα. Το πώς θα παίξει επηρεάζει τη λειτουργία του παιχνιδιού:

Αν είναι νέος παίκτης, θα γίνει η εισαγωγή του στη βάση δεδομένων του Ησιόδου ενεργοποιώντας μια λειτουργία της μορφής `InsertPlayer()`. Αν έχει ξαναπαίξει και



βάλει τα στοιχεία του σωστά (όνομα και κωδικό), ο Ησίοδος θα του επιστρέψει την μέχρι τώρα βαθμολογία του με μια λειτουργία τύπου RetrieveProgress().

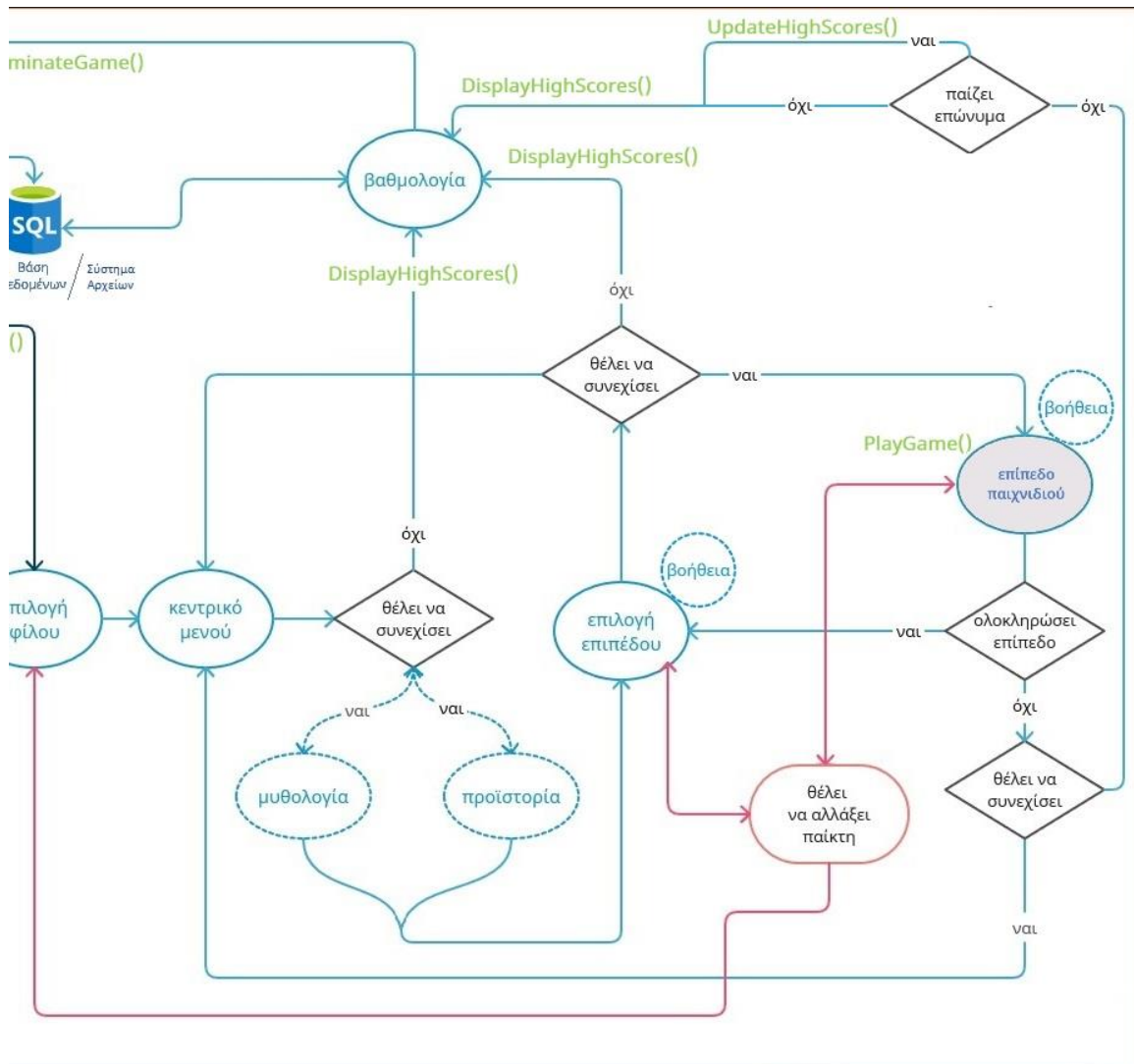


Εικόνα 21: UML Δραστηριότητας (Μέρος Α')

Ο παίκτης ενώ επηρεάζει με την ενέργειά του το σύστημα, δεν επηρεάζεται από αυτή. Σε κάθε περίπτωση το σύστημα φορτώνει στοιχεία για τα επίπεδα/κεφάλαια που ενδέχεται να παιχθούν και αυτός συνεχίζει (continue()) και διαλέγει τον ήρωά του.

Οδηγείται στο κεντρικό μενού. Στο κεντρικό μενού καλείται να αποφασίσει αν θα συνεχίσει ή όχι (Εικόνα 22). Αν μετάνιωσε και θέλει να αποχωρήσει, μπορεί, οδηγούμενος στη Βαθμολογία των δέκα καλύτερων παικτών. Στο σημείο αυτό, από όπου κι αν φτάσει ο παίκτης, ενεργοποιείται η λειτουργία DisplayHighScores(), ενώ με μία ακόμη παρέμβαση του οδηγείται στην έξοδο, με την TerminateGame().

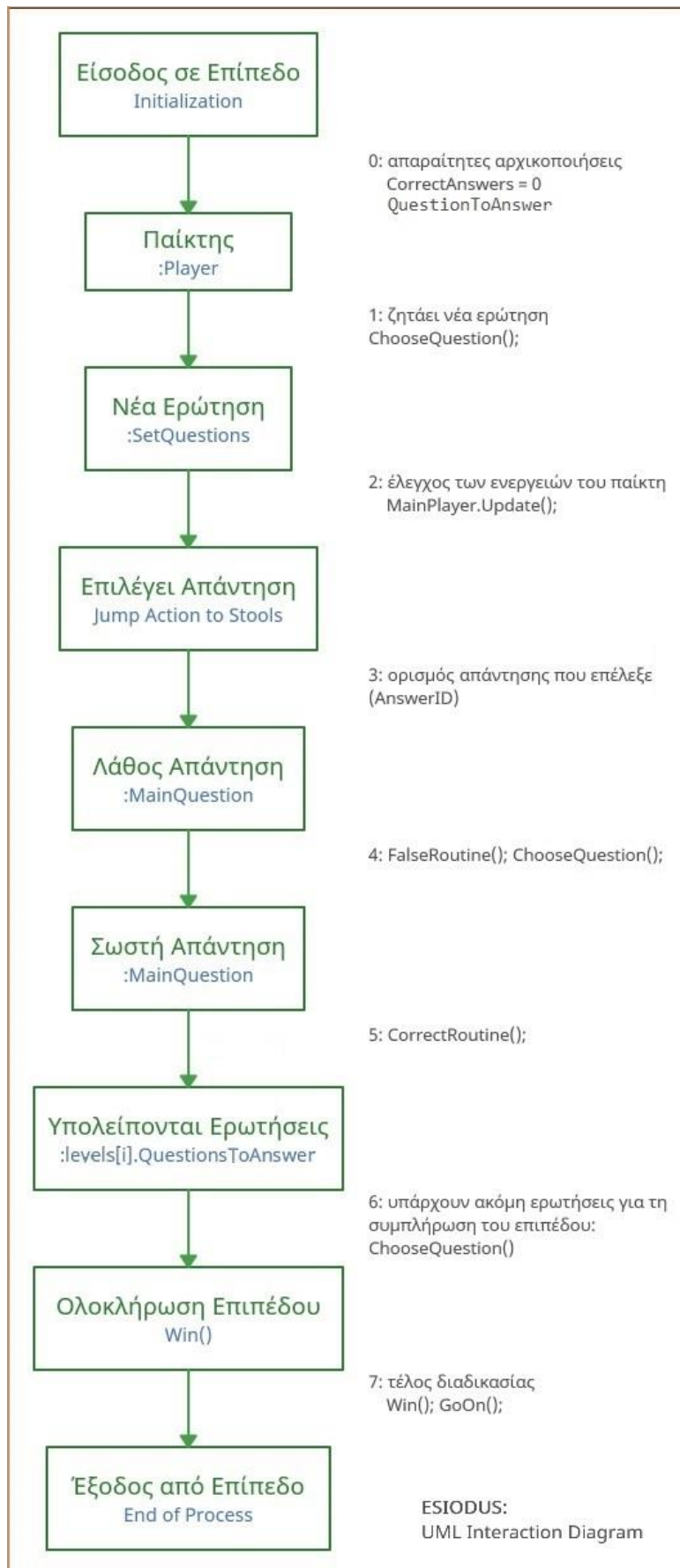
Αν δεν αποχωρήσει, μπορεί να επιλέξει ποιο από τα δύο μέρη της Ιστορίας επιθυμεί για να συνεχίσει και ποιο επίπεδο αυτών. Τα δύο μέρη, της Μυθολογίας και της



**Εικόνα 22: UML Δραστηριότητας (Μέρος Β')**

Προϊστορίας είναι εικονικά γι' αυτό και παρουσιάζονται με διακεκομμένες ελλείψεις: στην πραγματικότητα δεν υπάρχουν (όπως και οι οθόνες βοήθειας όπου εμφανίζονται), είναι θέμα της διεπαφής. Επίπεδο διαλέγει μόνο στην πραγματικότητα και παίζει το παιχνίδι: PlayGame().

Όπως, επίσης, φαίνεται και στην Εικόνα 22, ο παίκτης μπορεί να αποχωρήσει σε κάθε βήμα τόσο στην επιλογή επιπέδου όσο και όταν παίζει το επίπεδο που διάλεξε, είτε το έχει ολοκληρώσει, είτε όχι με αντίστοιχη επιστροφή στην επιλογή επιπέδου ή στο κεντρικό μενού ή να αλλάξει παίκτη και στις δύο καταστάσεις, επιστρέφοντας στο σημείο ακριβώς που ήταν. Υπάρχει μια τελευταία διαφοροποίηση κατά την έξοδο από το παιχνίδι που εξαρτάται από την αρχική επιλογή του χρήστη να παίζει επώνυμα ή ανώνυμα: στον επώνυμο παίκτη ενημερώνονται τα στοιχεία του, με τη λειτουργία UpdateHighScores().



Εικόνα 23: UML Διάγραμμα Αλληλεπίδρασης

### Interaction Diagram

Τα διαγράμματα Ακολουθίας (Sequence Diagrams) και αυτά της Συνεργασίας (Collaboration Diagrams) έχουν ως κοινό σκοπό την παρουσίαση της δυναμικής ροής του συστήματος. Το πρώτο καταγράφει τη χρονική ακολουθία της ενδοεπικοινωνίας από το ένα αντικείμενο στο άλλο και το δεύτερο περιγράφει την οργάνωση των αντικειμένων που συμμετέχει σ' αυτή τη ροή. Το Ακολουθίας δείχνει τις κλήσεις μεθόδων από ένα αντικείμενο στο άλλο και παρουσιάζει το πραγματικό σενάριο του συστήματος όταν εκτελείται. Το Συνεργασίας παρουσιάζει την ακολουθία κλήσεων μεθόδων χρησιμοποιώντας μια απλή τεχνική αρίθμησης. Ο αριθμός υποδεικνύει απλώς με ποια σειρά καλούνται οι μέθοδοι η μία μετά την άλλη.

Τα δύο μαζί, παρόμοια αλλά αλληλοσυμπληρούμενα αποτελούν το Διάγραμμα Αλληλεπίδρασης. Επειδή αυτό χρησιμοποιείται όταν θέλουμε να κατανοήσουμε την ακολουθία της ροής ελέγχου από το ένα αντικείμενο στο άλλο και την οπτική οργάνωση των στοιχείων ενός συστήματος, επέλεξα να παρουσιάσω με Διάγραμμα Αλληλεπίδρασης το κυρίως μέρος του συστήματος, δηλαδή το παιχνίδι.

Όπως φαίνεται στην Εικόνα 23 θεωρώ σαν αρχικοποίηση την Είσοδο σε ένα επίπεδο και σαν μηδενικής σειράς ενέργειες τις αρχικοποιήσεις των απαραίτητων μεταβλητών CorrectAnswers που δηλώνει τις σωστές απαντήσεις του παίκτη και TotalAnswers που είναι οι απαντήσεις που πρέπει να απαντηθούν ώστε να ολοκληρωθεί το επίπεδο.

1. Το αντικείμενο player ζητάει μία ερώτηση καλώντας την ChooseQuestion()
2. Οι κινήσεις του παίκτη ελέγχονται από την MainPlayer.Update()
3. ο οποίος επιλέγει μια απάντηση πηδώντας πάνω σε ένα σκαλί ορίζοντας το AnswerID
4. Αν το AnswerID δεν ταιριάζει με τη σωστή απάντηση της MainQuestion καλείται η ρουτίνα του «λάθους» και στη συνέχεια η ChooseQuestion() ξανά
5. Αν το AnswerID ταιριάζει με τη σωστή απάντηση της MainQuestion καλείται η ρουτίνα του «σωστού»
6. Αν υπάρχουν ακόμη απαντήσεις που πρέπει να απαντηθούν καλείται ξανά η ChooseQuestion()

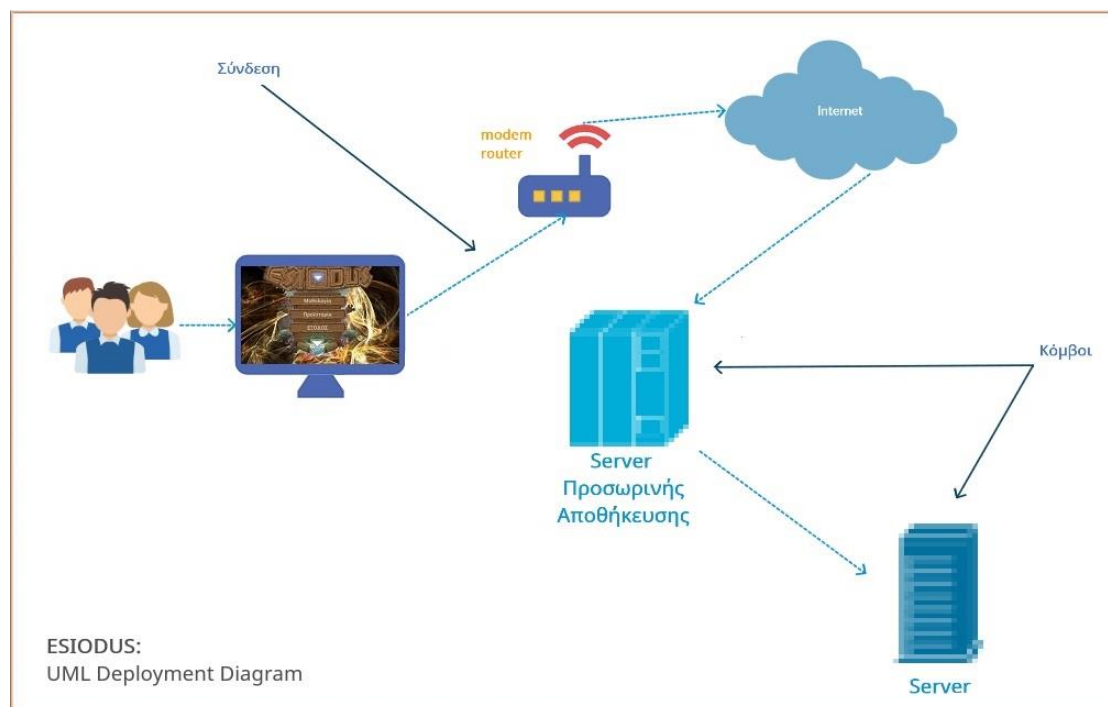
7. Αν οι CorrectAnswers είναι όσες και οι QuestionToAnswer του επιπέδου καλείται η ρουτίνα της «νίκης» και της «συνέχειας» που οδηγεί στην έξοδο από το επίπεδο.

Προφανώς και η παραπάνω περιγραφή είναι γενική και αφορά στο ουσιαστικό μέρος του παιχνιδιού. Δεν αναφέρονται όλες οι λεπτομέρειες των γραφικών, των κινουμένων σχεδίων και των κλήσεων των ήχων, που στολίζουν το παιχνίδι. Στην πραγματικότητα κάθε αριθμός της παραπάνω ακολουθίας είναι ένα στιγμιότυπο που θα μπορούσε να περιγραφεί με το δικό του ξεχωριστό διάγραμμα αλληλεπίδρασης. Ωστόσο αυτό θα διέφευγε του σκοπού της περιγραφής του παιχνιδιού. Τα διαγράμματα που επέλεξα να παρουσιάσω είναι αντιπροσωπευτικά της ουσίας του συστήματός μου, ενώ λεπτομερέστερη περιγραφή θα ακολουθήσει στο σχολιασμό των κλάσεων.

### 3.3.5 Δομικά Διαγράμματα – Structural Diagrams

Ολοκληρώνοντας τα διαγράμματα συμπεριφοράς, η περιγραφή περνάει στα διαγράμματα της δομής. Προχωράμε, λοιπόν, σε στοιχεία, που αποτελούν την κύρια δομή του προγράμματος επομένως είναι σταθερά.

#### Deployment Diagram



Εικόνα 24: UML Διάγραμμα Ανάπτυξης

Από τα πιο στατικά μέρη ενός συστήματος είναι το σύνολο των κόμβων, δηλαδή των φυσικών οντοτήτων στις οποίες αναπτύσσονται οι συνιστώσες του συστήματος και περιγράφεται από το Διάγραμμα Ανάπτυξης. Στην Εικόνα 24 βλέπουμε την οπτικοποίηση της ανάπτυξης του συστήματος του Ησιόδου. Οι χρήστες που θα καθίσουν μπροστά στον υπολογιστή τους, ακόμα και αν κατεβάσουν το παιχνίδι τους τοπικά, θα πρέπει να χρησιμοποιήσουν το διαδίκτυο για την ενέργεια αυτή. Οπότε, ο υπολογιστής τους θα πρέπει να συνδεθεί με το Internet μέσω ενός δρομολογητή και από κει, περνώντας από τον εξυπηρετητή της προσωρινής αποθήκευσης θα συνδεθεί με τον εξυπηρετητή που «σερβίρει» τον Ησιόδο. Αφού επιτευχθεί η σύνδεση, οι χρήστες θα έχουν τον Ησιόδο στη διάθεσή τους.

### **Component Diagram**

Ως συνιστώσες ενός συστήματος εννοούνται οι κλάσεις, οι διεπαφές και οι συνεργασίες. Τα Διαγράμματα Συνιστωσών αντιπροσωπεύουν το σύνολο αυτών των στοιχείων και τις σχέσεις τους. Αντιπροσωπεύουν την υλοποίηση του συστήματος.

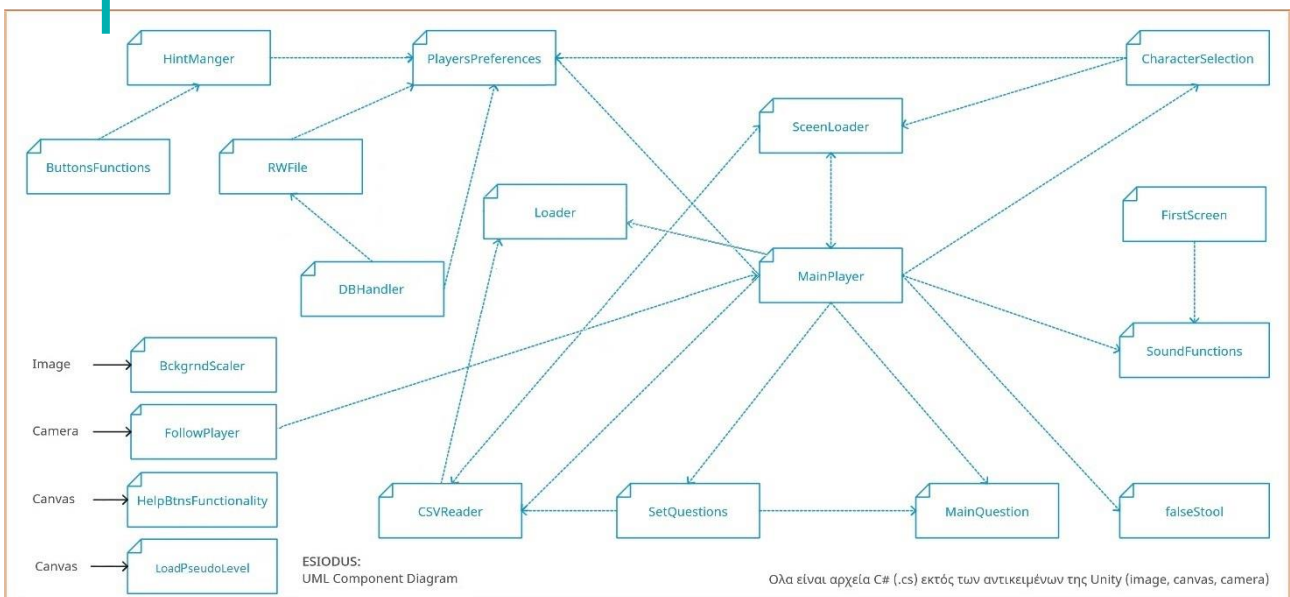
Στην Εικόνα 26 βλέπουμε το διάγραμμα Συνιστωσών για τον Ησιόδο, όπου φαίνονται όλες οι κλάσεις που όρισα εγώ προγραμματιστικά και οι μεταξύ τους εξαρτήσεις. Αντικείμενα της Unity εξαρτώνται από κάποιες κλάσεις μου, όπως κάποιες εικόνες, καμβάδες και η κάμερα του κυρίως παιχνιδιού που ακολουθεί τον παίκτη μου. Πρέπει να τονίσω ότι όλες οι συνιστώσες μου είναι αρχεία C# (.cs) και όπως είναι λογικό εξαρτώνται από βιβλιοθήκες της Unity και τις C#. Στο σύνολό τους έχω χρησιμοποιήσει τις 11 που φαίνονται στην Εικόνα 25. Επέλεξα να τις παρουσιάσω με αυτόν τον τρόπο για να μη γίνει πιο πολύπλοκο το διάγραμμα.

Επειδή δε, το διάγραμμα συνιστωσών χρησιμοποιείται για την οπτικοποίηση της υλοποίησης επέλεξα να παρουσιάσω το ίδιο διάγραμμα επαυξημένο ώστε να φαίνονται οι συγκεκριμένες εξαρτήσεις, δηλαδή τα διαμοιραζόμενα ή κληρονομούμενα χαρακτηριστικά και μέθοδοι. Τα στοιχεία βρίσκονται κοντά στην εξαρτημένη κλάση που τα χρησιμοποιεί (ή την υποκλάση), ενώ έχουν το χρώμα της κλάσης στην οποία ανήκουν. Για παράδειγμα, η κλάση HintManager εξαρτάται από την PlayersPreferences αφού χρησιμοποιεί τα στοιχεία levels[i]. Το levels[i] είναι γραμμένο κοντά στη HintManager ενώ έχει χρώμα πορτοκαλί, αυτό της κλάσης της, της PlayersPreferences. Ακόμα, ένα αντικείμενο της Unity, η camera, έχει κοντά της γραμμένο τον κόκκινο :player που είναι στιγμιότυπο της κόκκινης MainPlayer, κ.ο.κ. Στην Εικόνα 27 φαίνονται όλες οι εξαρτήσεις μεταξύ συνιστωσών, αναλυτικά.



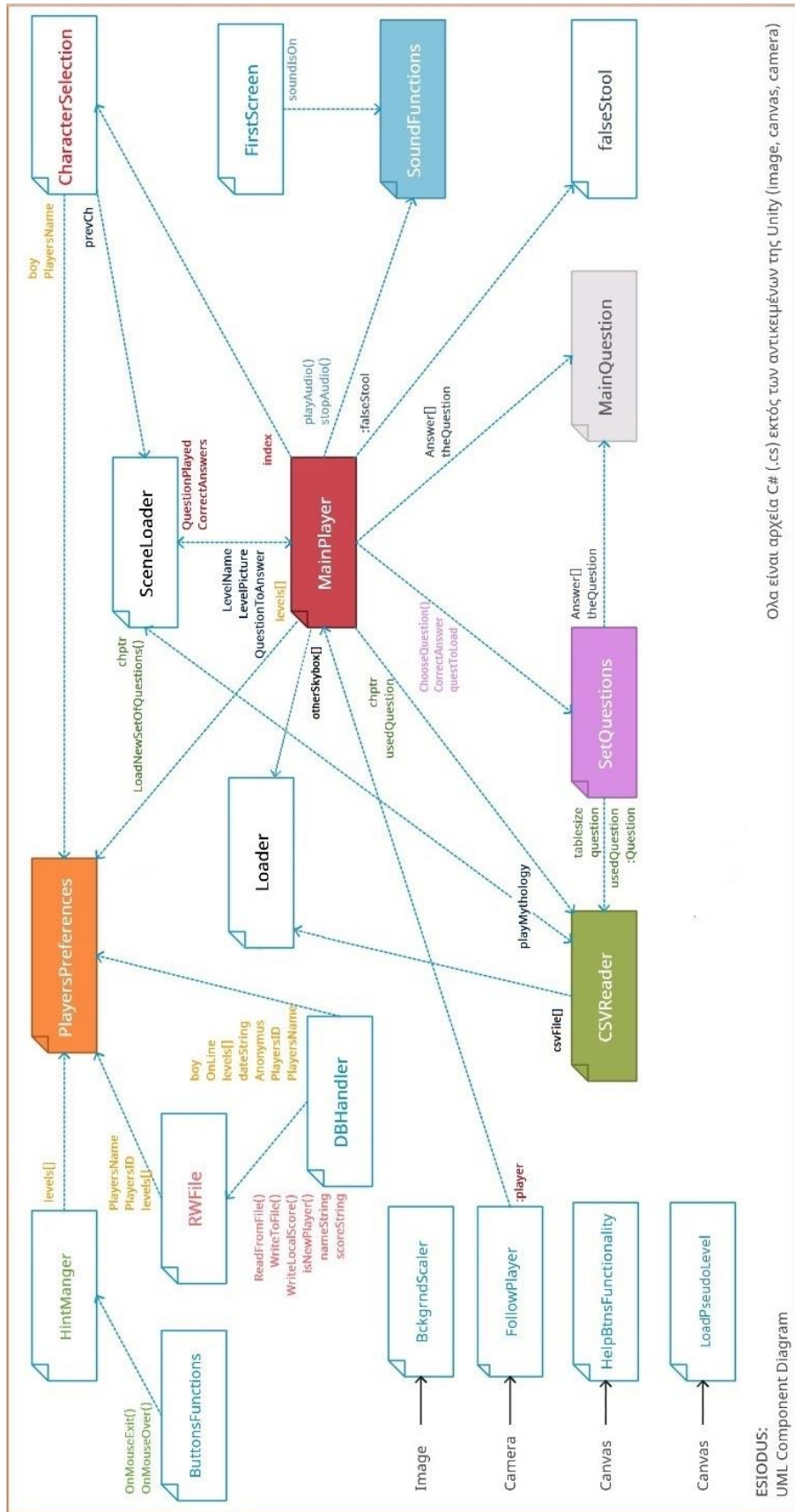
```
UnityEngine
UnityEngine.UI
UnityEngine.Networking
UnityEngine.EventSystems
UnityEngine.SceneManagement
System
System.IO
System.Collections
System.Collections.Generic
System.Text.RegularExpressions
TMPo
```

Εικόνα 25: Συνιστώσες της Unity



Εικόνα 26: UML Διάγραμμα Συνιστωσών

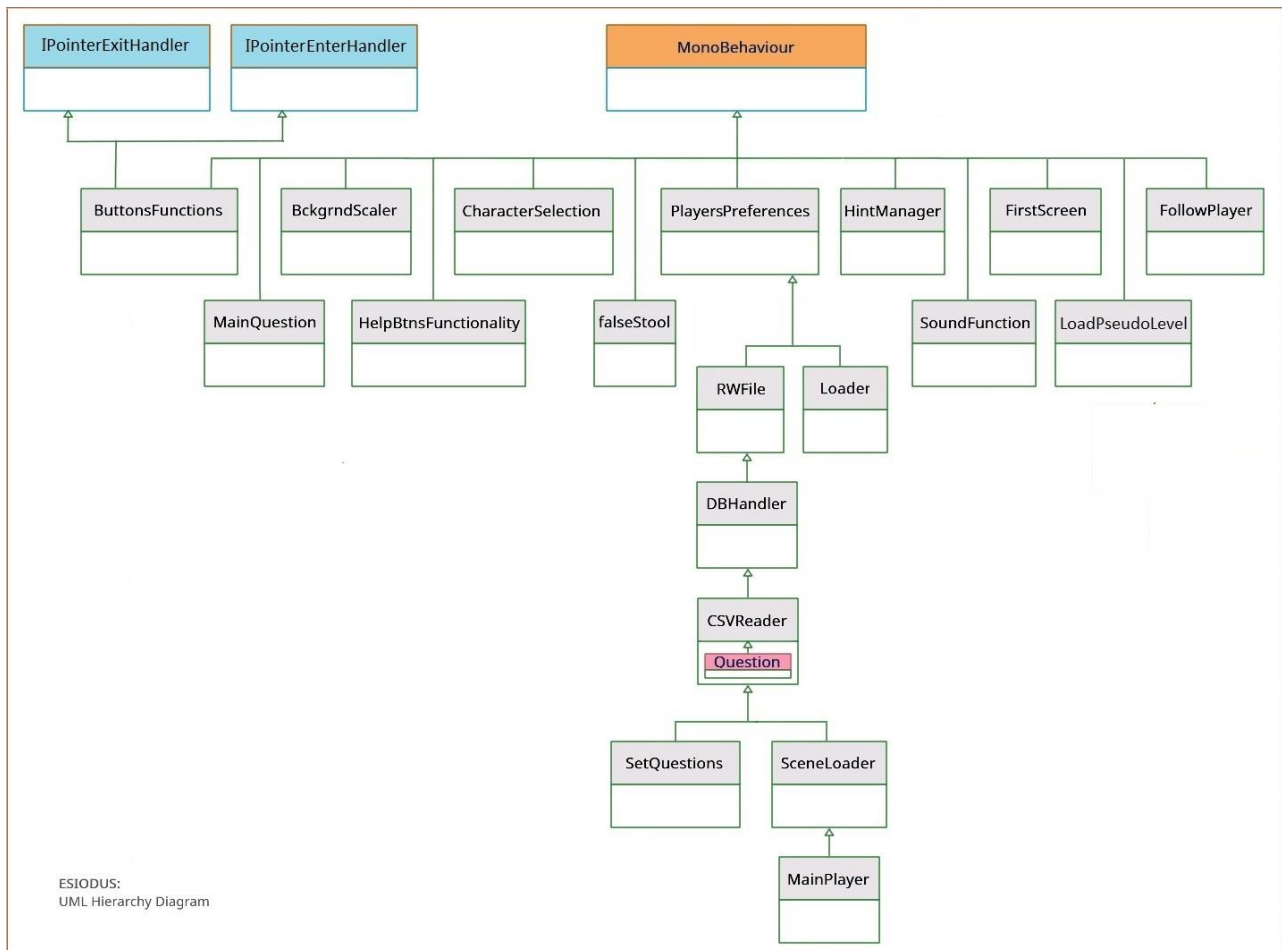




Εικόνα 27: UML Διάγραμμα Συνιστωσών - Επαυξημένο

## Class Diagram

Φτάνοντας στα διαγράμματα κλάσεων, πρέπει να σημειωθεί ότι αυτά είναι τα πιο κοινά διαγράμματα της UML. Αποτελούνται από κλάσεις, διεπαφές, συσχετίσεις και συνεργασίες. Ουσιαστικά, αντιπροσωπεύουν τον αντικειμενοστραφή χαρακτήρα ενός συστήματος. Στον Ησίοδο, το διάγραμμα Ιεραρχίας των κλάσεων του φαίνεται στην Εικόνα 28 που ακολουθεί:

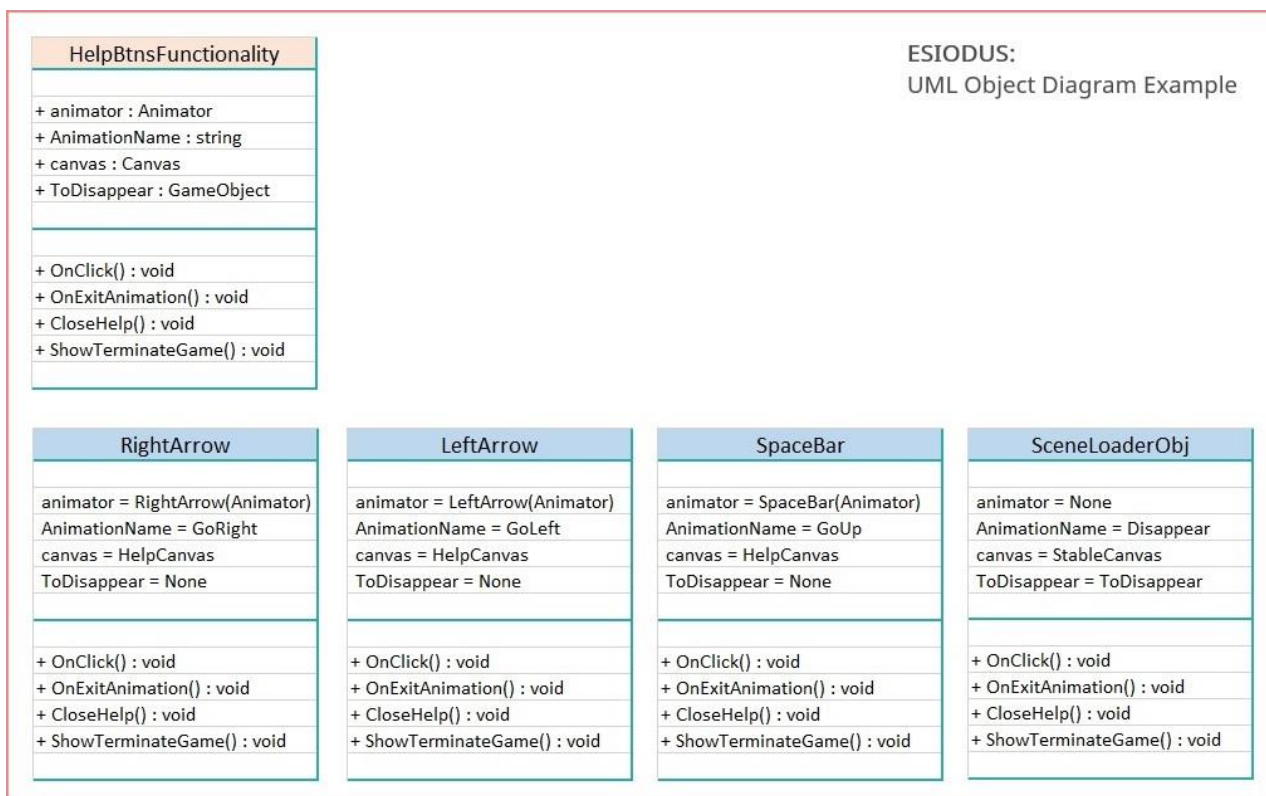


**Εικόνα 28: UML Ιεραρχικό Διάγραμμα**

Από τις 20 κλάσεις που δημιούργησα, οι 12 προέρχονται από την MonoBehaviour, δηλαδή είναι υποκλάσεις της. Η MonoBehaviour είναι η βασική κλάση από την οποία μπορεί να προέρχεται κάθε σενάριο (script) της Unity σε C# [46]. Όρισα μία υποκλάση μέσα στην CSVReader, την Question, ενώ μία εκ των κλάσεών μου, η ButtonsFunctions, εκτός από τη MonoBehaviour κληρονομεί και από τις διεπαφές (interfaces) IPointerExitHandler και IPointerEnterHandler που με τη σειρά τους ανήκουν στην EventSystem. Τέλος, 7 κλάσεις κληρονομούν η μία από την άλλη, μεταφέροντας χαρακτηριστικά και μεθόδους από τις υπερκλάσεις, απλοποιώντας τις εξαρτήσεις των συνιστωσών.

## Object Diagram

Πριν την αναλυτική περιγραφή των κλάσεων, θα κάνω μια μικρή αναφορά και στα διαγράμματα Αντικειμένων. Αυτά μοιάζουν με ένα «παράδειγμα» κλάσης. Είναι δηλαδή στιγμιότυπα της κλάσης σε κάποια δεδομένη στιγμή, άρα και πιο κοντά στην πραγματική εφαρμογή του συστήματος. Η χρήση τους είναι παρόμοια με αυτή των κλάσεων, δηλαδή χρησιμοποιούνται για την κατασκευή πρωτοτύπου ενός συστήματος από πρακτική άποψη. Ένα παράδειγμα αντικειμένων παρουσιάζεται στην επόμενη εικόνα:



**Εικόνα 29: Παράδειγμα Διαγράμματος Αντικειμένων**

Βλέπουμε την κλάση HelpBtnsFunctionality η οποία είναι υπεύθυνη για τη λειτουργία των κουμπιών της οθόνης βοήθειας, συμπεριλαμβανομένης και της κίνησής τους (animation). Από κάτω έχουμε τέσσερα αντικείμενα που ανήκουν στην κλάση αυτή, όπου τα γενικά χαρακτηριστικά της κλάσης παίρνουν συγκεκριμένα ονόματα, διατηρώντας όλες τις λειτουργίες της. Έτσι, το RightArrow ανήκει στην κλάση HelpBtnsFunctionality και έχει χαρακτηριστικά

- RightArrow.animator = RightArrow,
- RightArrow.AnimationName = GoRight,

- `RightArrow.canvas = HelpCanvas` και
- `RightArrow.ToDisappear = null`

ενώ έχει όλες τις μεθόδους της κλάσης. Αντίστοιχα ισχύουν και για τα επόμενα τρία αντικείμενα. Τέλος ακολουθεί κι ένα ακόμα παράδειγμα αντικειμένου, αυτό «του παίκτη», όπου παρουσιάζονται και τα κληρονομούμενα χαρακτηριστικά του:

:player
<code>PlayerPreferences.PlayersName = "Agni"</code>
<code>PlayerPreferences.PlayersID = 10</code>
<code>PlayerPreferences.boy = false</code>
<code>PlayerPreferences.Anonymus = false</code>
<code>PlayerPreferences.dateString = "2023-02-08 20:10:32"</code>
<code>PlayerPreferences.OnLine = false</code>
<code>PlayerPreferences.levels = [9281, 0, 0, ... , 0]</code>
<code>RWFile.scoreFile = "Agni10.txt"</code>
<code>RWFile.scoreString = ""</code>
<code>RWFile.nameString = ""</code>
<code>CSVReader.chptr = 2</code>
<code>CSVReader.tablesize = 23</code>
<code>CSVReader.quetion = ["Ποια θεά παντρεύτηκε ο Δίας", "Τη</code>
<code>CSVReader.usedQuestion = []</code>
<code>SceneLoader.prevCh = 2</code>
<code>SceneLoader.QuestionToAnswer = 12</code>
<code>SceneLoader.HelpCanvas = "HelpCanvas"</code>
<code>SceneLoader.LevelName = "Οι θεοί του Ολύμπου"</code>
<code>SceneLoader.LevelPicture = kef112.jpg</code>
<code>SceneLoader.PlayMythology = true</code>
<code>Avatar = [Alexandra, Pollys, Giwrgos, Suzan, Thanassis]</code>
<code>jumpWasPressed = false</code>
<code>mistakeWasMade = false</code>
<code>firstTimeOnStool = true</code>
<code>:</code>
<code>isGrounded = true</code>
<code>playerAnimator = Thanassis(Animator)</code>
<code>CorrectAnswers = 3</code>
<code>QuestionsPlayed = 3</code>
<code>all MainPlayer.operations()</code>

Εικόνα 30: Το αντικείμενο `player` σε κάποια στιγμή

### UML Διαγράμματα: Άμεση Σχέση – Διαφορετικός Ρόλος

Παρατηρώντας προσεκτικά όλα τα παραπάνω διαγράμματα και μελετώντας τις περιγραφές και τις χρήσεις των συνιστωσών σε αυτά, γίνεται σαφές ότι όλα μεταξύ

τους έχουν κάποια σχέση. Τα διαγράμματα συνιστώσων εξαρτώνται από τις κλάσεις, τις διεπαφές κλπ. που αποτελούν μέρος των διαγραμμάτων κλάσεων ή αντικειμένων, ενώ το διάγραμμα ανάπτυξης εξαρτάται πάλι από αυτές τις ίδιες συνιστώσες. Η σχέση τους είναι απολύτως άμεση. Κάθε ένα από αυτά ωστόσο ενδιαφέρει άλλη ομάδα κατά τη διαδικασία παραγωγής ενός συστήματος λογισμικού. Έτσι, τα διαγράμματα κλάσης χρησιμοποιούνται για γενικούς σκοπούς ανάπτυξης κατά τη στιγμή της κατασκευής του συστήματος, το ίδιο και τα διαγράμματα αντικειμένων. Το διάγραμμα ανάπτυξης, όπως φανερώνει και το όνομά του, ενδιαφέρει την ομάδα ανάπτυξης, ενώ το διάγραμμα συνιστώσων είναι χρήσιμο κατά τη φάση του σχεδιασμού.

Η UML είναι μια γλώσσα περιγραφική που χρησιμοποιείται για τη μοντελοποίηση συστημάτων λογισμικού και μη, με έμφαση στον αντικειμενοστραφή προγραμματισμό. Οπότε αν και τα περισσότερα από τα διαγράμματα UML χρησιμοποιούνται για τη μοντελοποίηση στατικών ή δυναμικών πτυχών, τα τεχνουργήματά τους δεν παύουν να είναι αντικείμενα.

### 3.4 Αναλυτική Παρουσίαση των Κλάσεων

Ως τελικό στάδιο της περιγραφής του παιχνιδιού μου θα παρουσιάσω, με τη σημειολογία της UML, αναλυτικά τις κλάσεις που σχεδίασα, εξηγώντας και τη χρήση τους, με αντίστοιχη σειρά παρουσίασής του κώδικα, στο Παράρτημα Α. Όπως ενδείκνυται, στα χαρακτηριστικά αναφέρονται αυτά που είναι σημαντικά για τον κορμό της λειτουργίας τους και όχι βοηθητικά και δευτερεύοντα.

#### 3.4.1 Οι υποκλάσεις της MonoBehaviour

Παρακολουθώντας το ιεραρχικό διάγραμμα UML στην Εικόνα 28, οι κλάσεις που συναντάμε, υποκλάσεις της MonoBehaviour, ακολουθούν αλφαβητικά.

BckgrndScaler	Κλάση υπεύθυνη για την κλιμάκωση της εικόνας του υποβάθρου σε μια σκηνή. Η εικόνα που θα κλιμακωθεί είναι η backgroundImage, ενώ ratio είναι το ποσοστό της κλιμάκωσης, το οποίο καθορίζεται στη Start(), ενώ η Update() φροντίζει να επαναυπολογίζει το μέγεθος της backgroundImage ανάλογα με τις διαστάσεις της οθόνης κάθε στιγμή.
– ratio : float	
+ backgroundImage : Image	
– Start() : void	
– Update() : void	

ButtonsFunctions
– timeToWait : float + hintText : Text
– StartTimer() : IEnumerator – ShowMessage() : void + OnPointerEnter(eventData) : void + OnPointerExit(eventData) : void

Η ButtonsFunctions σε συνδυασμό με την HintManger που περιγράφεται παρακάτω, φροντίζουν ώστε να εμφανίζονται οι υποδείξεις (hints) με τα ονόματα των επιπέδων, όταν το ποντίκι περάσει πάνω από το αντίστοιχο εικονίδιο, στο μενού επιλογής επιπέδων. hintText είναι το μήνυμα της υπόδειξης, ενώ timeToWait είναι η μικρή χρονοκαθυστέρηση που ξεκινάει με τη μέθοδο:

StartTimer(), μέχρι να εμφανιστεί το μήνυμα, με τη ShowMessage(). Όταν το ποντίκι εισέλθει πάνω από την εικόνα η OnPointerEnter(eventData) εκτελείται αντίστοιχα όταν εξέρχεται, εκτελείται η OnPointerExit(eventData). Και οι δύο είναι συναρτήσεις των διεπαφών IPointerEnterHandler και IPointerExitHandler αντίστοιχα. Η ButtonsFunctions θα μπορούσε να ήταν και υποκλάση της HintManger, αλλά θα κληρονομούσε πολλά χαρακτηριστικά από αυτήν που ουσιαστικά είναι αχρείαστα στα αντικείμενά της. Έτσι αρκέστηκα στη συνεργασία τους.

CharacterSelection
– CharacterList : GameObject[] + index : int
– Start() : void + ToggleLeft() : void + ToggleRight() : void + ConfirmChoice() : void

Μέσω της παρούσας κλάσης γίνεται η επιλογή ήρωα. CharacterList είναι η λίστα με τους 5 φίλους και index είναι η θέση του επιλεγμένου ήρωα στη λίστα.

Με τη Start() αρχικοποιούνται όσες τιμές είναι απαραίτητο και απενεργοποιούνται οι πέντε χαρακτήρες· ενεργοποιείται αυτός που ήταν ήδη επιλεγμένος με τον index. ToggleLeft() και ToggleRight() είναι μέθοδοι κατά τις οποίες ενεργοποιείται ο προηγούμενος ή ο επόμενος αντίστοιχα ήρωας στη λίστα. Η ConfirmChoice(), φυσικά, επιβεβαιώνει την επιλογή, οδηγώντας στην στην επόμενη οθόνη ή στην οθόνη όπου βρισκόταν ο παίκτης προηγούμενος.

falseStool
+ rbStool : Rigidbody

Στη falseStool ανήκουν τα αντικείμενα που αντιστοιχούν στις λανθασμένες απαντήσεις.



	Κάθε φορά που δίνεται μια λανθασμένη απάντηση, δημιουργείται ένα νέο στιγμιότυπο της falseStool για να πέσει μακριά. rbStool είναι το Rigidbody του αντικειμένου, που εκλαμβάνεται στη Start(), ενώ στην OnCollisionEnter(collision) ακινητοποιείται το τουβλάκι μόλις έρθει σε επαφή με το γκαζόν.
– Start() : void	
– OnCollisionEnter(collision) : void	

FirstScreen	<p>Η FirstScreen ελέγχει τη ροή της γνωριμίας με τους φίλους στη 2η οθόνη. Στα ιδιωτικά χαρακτηριστικά της ανήκουν τα EnterWasPressed, που ελέγχει πόσες φορές θα πατηθεί το πλήκτρο Enter, ώστε να ενεργοποιεί τον αντίστοιχο ήρωα και το Voice που κρατάει κάθε φορά την ενεργή ομιλία. Δεδομένου ότι μπορεί να πρέπει ο χρήστης να περιμένει μέχρι να ολοκληρωθεί ένα animation, το αν επιτρέπεται το πάτημα πλήκτρων ελέγχεται από την getKeys.</p> <p>Με την Awake() ενεργοποιούνται οι πρώτες λειτουργίες που απαιτούνται στην έναρξη της οθόνης, όπως ο ορισμός του ήχου σε «On», η ενεργοποίηση των πρώτων animations και όποια</p>
– EnterWasPressed : int	
– Voice : AudioSource	
+ getKeys : bool	
– Awake() : void	
– Update() : void	
– FixedUpdate() : void	
– AnimatorIsPlaying(Animator, stateName) : bool	
– GiwrgosAction() : void	
– SuzanAction : void	
– ThanassisFirstAction() : void	
+ NextActionOnClick() : void	
+ NextAction() : void	

άλλη αρχικοποίηση είναι απαραίτητη. Η Update() φροντίζει να εφαρμόζει την επιλογή του χρήστη για τον ήχο και μετράει τα Enter που αυτός πατάει ώστε να προχωράει η εξέλιξη του σεναρίου. Η FixedUpdate() φροντίζει για την επόμενη ενέργεια που θα λάβει χώρα με το Enter του παίκτη καλώντας την NextAction() ενώ η NextActionOnClick() φροντίζει να υπάρχει το ίδιο αποτέλεσμα με το Enter του πληκτρολογίου, με το αν γίνει κλικ με το ποντίκι στο κουμπί που δίνει τις οδηγίες.

FollowPlayer	Με offset, όσο δηλώνει το χαρακτηριστικό offset, η κάμερα της κλάσης FollowPlayer, ακολουθεί τον παίκτη player.
– offset : Vector3	
+ player : Transform	

Το offset ορίζεται στη Start() ενώ στην Update() υπολογίζεται κάθε φορά η θέση της κάμερας να



<code>- Update() : void</code>	είναι η θέση του παίκτη, με offset το προκαθορισμένο.
--------------------------------	---

HelpBtnsFunctionality	
<code>+ animator : Animator</code>	Η HelpBtnsFunctionality είναι υπεύθυνη για τα animations των πλήκτρων στις οθόνες βοήθειας. animator και AnimationName είναι ο ελεγκτής της κίνησης και το όνομα του animation που θέλουμε να παίξει, στον καμβά canvas, ενώ ToDisappear είναι ένα αντικείμενο της Unity που πρέπει να του στερηθεί η ορατότητα σε κάποιο σημείο του προγράμματος, προκειμένου η εφαρμογή να οδηγηθεί στην έξοδο,
<code>+ AnimationName : string</code>	
<code>+ canvas : Canvas</code>	
<code>+ ToDisappear : GameObject</code>	
<code>+ onClick() : void</code>	με την ShowTerminateGame(), που οδηγεί κατευθείαν στον τερματισμό. Με την onClick() παίζει το animation ενώ μόλις αυτό τελειώσει εκτελείται η OnExitAnimation(). Η CloseHelp() κλείνει την οθόνη. Χαρακτηριστικά και μέθοδοι της κλάσης είναι όλα δημόσια για να είναι ορατά από τα αντικείμενα της Unity και να μπορούν να δημιουργούν κάθε φορά αντίστοιχα στιγμιότυπα.
<code>+ OnExitAnimation() : void</code>	
<code>+ CloseHelp() : void</code>	
<code>+ ShowTerminateGame() : void</code>	

HintManager	
<code>+ DisplayedHint : Text</code>	Η HintManager είναι, όπως προαναφέρθηκε, η κλάση που σε συνδυασμό με την ButtonsFunctions δείχνουν τα hints των επιπέδων. DisplayedHint είναι το μήνυμα που θα εμφανιστεί, στο κατάλληλο παράθυρο HintWindow, διαμορφώνοντας τις ενέργειες OnMouseHover, OnMouseExit.
<code>+ HintWindow : RectTransform</code>	
<code>+ OnMouseHover : Action&lt;string, Vector2&gt;</code>	
<code>+ OnMouseExit : Action</code>	
<code>- Start() : void</code>	Στη Start() «κρύβεται» το παράθυρο των υποδείξεων. Όταν ενεργοποιείται ή απενεργοποιείται ένα αντικείμενο της κλάσης, οι ενέργειες χειρίζονται ανάλογα, δείχνοντας την υπόδειξη με τη ShowHint(hint, mousePos) ή κρύβοντάς την με τη HideHint(). Οι μέθοδοι
<code>- OnEnable() : void</code>	
<code>- OnDisable() : void</code>	
<code>- ShowHint(hint, mousePos) : void</code>	
<code>- HideHint() : void</code>	

αυτοί είναι ιδιωτικές αφού καλούνται μέσα από τη συνεργασία της κλάσης με τη ButtonsFunctions χωρίς να υπάρχει ανάγκη άμεσης εκτέλεσης και της αντίστοιχης ορατότητας.

LoadPseudoLevel
- fromPart : int
- toPart : int
+ MythCanvas : Canvas
+ MythBgrCanvas : Canvas
+ PreHiCanvas : Canvas
+ PreHiBgrCanvas : Canvas
- Start() : void

Η συγκεκριμένη κλάση είναι υπεύθυνη για την εικονική φόρτωση των επιμέρους ενοτήτων. Αν ο παίκτης έχει επιλέξει τη μυθολογία, ενεργοποιείται, δηλαδή γίνεται ορατός, ο καμβάς της μυθολογίας (MythCanvas) και το αντίστοιχο υπόβαθρο (MythBgrCanvas), ενώ διαφορετικά θα εμφανιστούν στο χρήστη οι PreHiCanvas και PreHiBgrCanvas που αντιστοιχούν στην προϊστορία. Τα κεφάλαια της κάθε ενότητας ξεκινούν από τον αριθμό fromPart και φτάνουν ως τον αριθμό toPart.

Οι παραπάνω ενέργειες λαμβάνουν όλες χώρα στη Start(), η οποία «βάφει» και τα ολοκληρωμένα επίπεδα με ένα σκούρο χρώμα για να φαίνεται ποια έχουν ολοκληρωθεί και ποια όχι.

MainQuestion
+ theQuestion : Text
+ Answer : Text[3]
- Start() : void

Η MainQuestion είναι η κλάση που ορίζει τα δημόσια χαρακτηριστικά theQuestion, δηλαδή την τρέχουσα ερώτηση και τον πίνακα Answer[3] με τις τρεις πιθανές απαντήσεις. Για την ακρίβεια, τα χαρακτηριστικά αυτά αντιστοιχούν στα αντικείμενα της Unity τα οποία θα περιλαμβάνουν την ερώτηση και τις απαντήσεις

και η αντιστοιχία γίνεται στη Start().

SoundFunction
+ soundIsOn : bool
+ eventSystem : EventSystem
- Start() : void

Όπως υποδηλώνει το όνομα της κλάσης, είναι υπεύθυνη για το χειρισμό του ήχου. Το soundIsOn είναι αληθές όταν ο παίκτης θέλει να υφίσταται ο ήχος και ψευδές στην αντίθετη περίπτωση. eventSystem είναι το σύστημα που ελέγχει τις ενέργειες κουμπιών.

+ ControlSound() : void	Στην έναρξη Start() και ανάλογα με τη μέχρι τώρα επιλογή του χρήστη, το κουμπί του ήχου δείχνει το ανάλογο εικονίδιο. ControlSound() φροντίζει την ίδια ενέργεια και τον ορισμό της soundIsOn. Η PlayAudio() παίζει έναν ήχο –αν αυτός είναι ανοιχτός και η StopAudio() τερματίζει αυτό το παίξιμο. Με την DisableBtns() απενεργοποιούνται τα κουμπιά του eventSystem ώστε να μην ισχύουν οι προκαθορισμένες λειτουργίες και να υπάρξει έλεγχος χρήσης των πλήκτρων από τον προγραμματιστή.
+ PlayAudio() : void	
+ StopAudio() : void	
+ DisableBtns() : void	

PlayersPreferences	Εκτός σειράς, τελευταία και σημαντική κλάση της MonoBehaviour είναι η PlayersPreferences. Ιδιωτικά της χαρακτηριστικά είναι τα πεδία InputName και InputID που περιλαμβάνουν το όνομα και τον κωδικό του παίκτη, τα οποία και αποθηκεύονται αντίστοιχα στα δημόσια PlayersName και PlayersID. levels[19] είναι οι βαθμολογίες που έχει πετύχει κάθε παίκτης στα 19 επίπεδα του παιχνιδιού, datestring είναι η ημερομηνία και ώρα εισόδου στο παιχνίδι ώστε να ξέρει ο παίκτης πόση ώρα ασχολήθηκε με αυτό. Το boy περιγράφει το φύλο του παίκτη, ώστε να ρυθμιστούν τα άρθρα στις εκφωνήσεις του παίκτη και να του αντιστοιχηθεί κατάλληλο εικονίδιο, το Anonymus γίνεται αληθές όταν ο παίκτης παίζει χωρίς να έχει βάλει τα στοιχεία του και η OnLine καθορίζει την εκδοχή του παιχνιδιού ως τοπικό ή διαδικτυακό, με όσες διαφορές αυτό περιλαμβάνει.
- InputName : Text	
- InputID : Text	
+ PlayersName : string	
+ PlayersID : int	
+ levels : int[]	
+ datestring : string	
+ boy : bool	
+ Anonymus : bool	
+ OnLine : bool	
- Start() : void	
+ StoreName() : void	

Όλες οι αρχικοποιήσεις γίνονται, όπως πάντα στη Start() ενώ η αποθήκευση και οι λοιπές επιλογές τακτοποιούνται στην StoreName().

### 3.4.2 Η ιεραρχία και οι υποκλάσεις της PlayersPreferences

Οι υπόλοιπες κλάσεις που θα περιγραφούν είναι όλες υποκλάσεις της PlayersPreferences, κληρονομώντας όλα τα χαρακτηριστικά που μόλις περιγράψαμε. Άρα «βλέπουν» η κάθε επόμενη τα δημόσια στοιχεία της

προηγούμενης, ώστε να τα γνωρίζουν και να μη χρειάζεται να έχουν πρόσβαση σε αυτά μέσω συνδέσεων και αναφορών.

Loader	
+ Skybox : Material[]	Η Loader είναι μια υποκλάση της PlayersPreferences χωρίς απογόνους. Όπως φανερώνει το όνομά της, φορτώνει όλα τα διαφορετικά υλικά Skybox[] που χρησιμοποιούνται για την αντίστοιχη ιδιότητα των σκηνών της Unity και στη συνέχεια τα διανέμει δημόσια μέσω της otherSkybox[]. Αντίστοιχα πράττει με τα αρχεία ερωτήσεων getFile[], csvFile[].
+ otherSkybox : Material[]	
+ getFile : TextAsset[]	
+ csvFile : TextAsset[]	
- Start() : void	

Η αντιστοίχιση γίνεται μέσα στη Start().

RWFile	
+ basicPath : string	Επόμενη υποκλάση της PlayersPreferences είναι η RWFile, υπεύθυνη για τη διαχείριση των αρχείων των παικτών και του αρχείου Βαθμολογίας (scoreFile) στην τοπική έκδοση του παιχνιδιού. Οπότε, το basicPath αναφέρεται στο βασικό κατάλογο όπου δημιουργούνται και αποθηκεύονται τα αρχεία αυτά και το path αποθηκεύει τη διαδρομή μέχρι το αρχείο του συγκεκριμένου παίκτη. Τα scoreString και nameString είναι χαρακτηριστικά που παίρνουν μέρος στην εμφάνιση των καλύτερων βαθμολογιών.
+ path : string	
+ scoreFile : string	
+ scoreString : string	
+ nameString : bool	
- Start() : void	Προφανώς οι αρχικοποιήσεις αυτών των μεταβλητών γίνονται στη Start(), στην οποία δημιουργείται και το αρχείο βαθμολογίας, αν δεν υπάρχει ως τώρα (δηλαδή την πρώτη φορά). Οι υπόλοιπες μέθοδοι έχουν τις προφανείς λειτουργίες που υποδηλώνουν τα ονόματά τους: Η WriteToFile() γράφει σε αρχείο το χαρακτηριστικό levels[] με τις βαθμολογίες του παίκτη σε κάθε επίπεδο του παιχνιδιού, που κληρονομεί από την υπερκλάση της, η ReadFromFile() τις διαβάζει από το αρχείο του παίκτη και τις αναθέτει στην levels[], ενώ η isNewPlayer() αποφαινεται αν ο παίκτης που κάνει εισαγωγή με συγκεκριμένα
+ WriteToFile() : void	
+ ReadFromFile() : void	
+ WriteLocalScore (newScore, newName) : void	
+ isNewPlayer() : bool	

Προφανώς οι αρχικοποιήσεις αυτών των μεταβλητών γίνονται στη Start(), στην οποία δημιουργείται και το αρχείο βαθμολογίας, αν δεν υπάρχει ως τώρα (δηλαδή την πρώτη φορά). Οι υπόλοιπες μέθοδοι έχουν τις προφανείς λειτουργίες που υποδηλώνουν τα ονόματά τους: Η WriteToFile() γράφει σε αρχείο το χαρακτηριστικό levels[] με τις βαθμολογίες του παίκτη σε κάθε επίπεδο του παιχνιδιού, που κληρονομεί από την υπερκλάση της, η ReadFromFile() τις διαβάζει από το αρχείο του παίκτη και τις αναθέτει στην levels[], ενώ η isNewPlayer() αποφαινεται αν ο παίκτης που κάνει εισαγωγή με συγκεκριμένα

στοιχεία έχει ξαναπαίξει ή είναι καινούριος παίκτης. Τέλος, η `WriteLocalScore(newScore, newName)`, ελέγχει αν η βαθμολογία του παίκτη είναι μέσα στις 10 καλύτερες και αν ναι την τοποθετεί στην κατάλληλη σειρά της, ετοιμάζοντας και τα `scoreString` και `nameString` για την εμφάνιση την Βαθμολογίας κατά την έξοδο από το παιχνίδι.

DBHandler	
– <code>firstLogin</code> : bool	Υποκλάση της <code>RWFile</code> αποτελεί η <code>DHandler</code> , που διαχειρίζεται οποιασδήποτε μορφής βάση δεδομένων του παιχνιδιού, είτε της τοπικής είτε την διαδικτυακής έκδοσης, οπότε έχει πολλή δουλειά να κάνει. Το χαρακτηριστικό <code>firstLogin</code> διακρίνει τη χρήση της κλάσης στην έναρξη ή στη λήξη του παιχνιδιού και το <code>GenScore</code> αποθηκεύει το συνολικό σκορ του παίκτη. Τα υπόλοιπα χαρακτηριστικά <code>loginURL</code> , <code>highscoreURL</code> , <code>updatehighscoreURL</code> , περιλαμβάνουν τις διευθύνσεις των αρχείων <code>.php</code> που διαχειρίζονται τη βάση δεδομένων των διαδικτυακών παικτών για την εμφάνιση του προηγούμενου σκορ του, για την ενημέρωση των στοιχείων του παίκτη και
– <code>GenScore</code> : int	
– <code>loginURL</code> : string	
– <code>updatehighscoreURL</code> : string	
– <code>highscoreURL</code> : string	
– <code>Start()</code> : void	
– <code>TryToLogin()</code> : void	
– <code>Login()</code> : IEnumerator	
– <code>UpdateHighScores()</code> : IEnumerator	
– <code>DisplayHighScores()</code> : IEnumerator	

την εμφάνιση των καλύτερων βαθμολογιών αντίστοιχα.

Η `Start()`, λοιπόν, αρχικοποιεί όσες μεταβλητές χρειάζεται, υπολογίζει τη συνολική βαθμολογία ως άθροισμα των επιμέρους βαθμολογιών των επιπέδων και ανάλογα αν το παιχνίδι είναι διαδικτυακό ή τοπικό κάνει την κατάλληλη διαχείριση ως προς την παρουσίαση των τελικών αποτελεσμάτων, όταν η `firstLogin` είναι ψευδής. Για το λόγο αυτό είναι και υποκλάση της `RWFile`, ώστε να έχει πρόσβαση στις λειτουργίες της. Η `TryToLogin()` χρησιμοποιείται στην περίπτωση του `firstLogin` για να κατευθύνει σωστά ποιες μέθοδοι θα χρησιμοποιηθούν για την είσοδο στο παιχνίδι, ανάλογα με την εκδοχή του. Η `Login()` κάνει τη διαδικτυακή είσοδο και κατεβάζει το προηγούμενο σκορ του παίκτη. `UpdateHighScores()` ενημερώνει τη βάση δεδομένων για τη βαθμολογία που πέτυχε ο παίκτης, ενώ η `DisplayHighScores()` ετοιμάζει τα `scoreString` και `nameString` στη διαδικτυακή περίπτωση.

CSVReader
+ chptr : int
+ tableSize : int
<b>+ Question</b>
+ Q : string
+ A1 : string
+ A2 : string
+ A3 : string
+ question : List<Question>
+ usedQuestion : List<Question>
- ReadFromCSV(csvFile) : void
+ LoadNewSetOfQuestions() : void

Ακολουθεί η κλάση CSVReader. Κατά το όνομα της είναι ο αναγνώστης των αρχείων των ερωτήσεων, με chptr το κεφάλαιο που επέλεξε να παίξει ο χρήστης και tableSize το σύνολο των ερωτήσεων που περιλαμβάνει αυτό. Στην κλάση αυτή ορίζεται μια μικρή υποκλάση η Question, ως νέος τύπος δεδομένων με τέσσερα πεδία. Q: είναι η ερώτηση, A1: η πρώτη πιθανή απάντηση, A2: η δεύτερη πιθανή απάντηση και A3: η τρίτη πιθανή απάντηση. Όρισα αυτόν τον τύπο, σε αντιστοιχία με την κλάση αντικειμένων MainQuestion της Unity για να αποθηκεύω τις ερωτήσεις. Σε συνέχεια οι question και usedQuestion είναι λίστες τύπου Question που φυλάγουν αντίστοιχα: τις ερωτήσεις που είναι υποψήφιες να επιλεγούν

και τις ερωτήσεις που έχουν απαντηθεί σωστά και δεν θα είναι ξανά υποψήφιες σε αυτή τη συνεδρία.

Η ReadFromCSV(csvFile) θα διαβάσει από το αρχείο csvFile τις ερωτήσεις του επιπέδου φτιάχνοντας τη λίστα question και αρχικοποιώντας τη usedQuestion. Η LoadNewSetOfQuestions() πραγματοποιεί ακριβώς την ίδια λειτουργία καλώντας την ReadFromCSV(csvFile), υπό την έννοια της ενθουλάκωσης, προκειμένου να μπορεί κάποιος να χρησιμοποιήσει την προηγούμενη μέθοδο χωρίς να έχει άμεση πρόσβαση σε αυτή.

SetQuestions
+ questToLoad : CSVReader.Question
+ CorrectAnswer : int
+ ChooseQuestion() : void

Έπεται η υποκλάση SetQuestion, δηλαδή αυτή που θα φορτώσει την επόμενη υποψήφια ερώτηση questToLoad από την question στο αντικείμενο MainQuestion της Unity και θα αποθηκεύσει τη σωστή απάντηση 1, 2 ή 3 στην ακέραιη CorrectAnswer.

Μοναδική μέθοδος η ChooseQuestion(), η οποία είναι υπεύθυνη για την κυρίως λειτουργία του παιχνιδιού. Φορτώνει στην τύχη (Random) μια ερώτηση από τη λίστα question των υποψήφιων ερωτήσεων στην

questToLoad αφαιρώντας την ταυτόχρονα από αυτήν. Φορτώνει στην MainQuestion.theQuestion την ερώτηση και αναθέτει τις πιθανές απαντήσεις στα MainQuestion.Answer[1][2][3], επίσης τυχαία, κρατώντας στην CorrectAnswer, σε ποια από τις τρεις πέτυχε η σωστή απάντηση. Αν αδειάσει η λίστα των ερωτήσεων, ανταλλάσσει τη θέση της με τις ήδη χρησιμοποιημένες και η διαδικασία συνεχίζεται όπως πριν. Στην περίπτωση των επιπέδων μου, αυτό δε μπορεί να συμβεί, αλλά υπάρχει για λόγους ευελιξίας σε αρχεία άλλων ερωτήσεων, διαφορετικών μεγεθών.

Η SetQuestion δεν είναι υπερκλάση καμίας άλλης. Στη Unity δεν υποστηρίζεται διπλή κληρονομικότητα. Διαφορετικά θα έφτιαχνα τη SetQuestion υποκλάση και της MainQuestion, ενώ η MainPlayer θα ήταν υποκλάση και της SetQuestion.

SceneLoader	
+ LevelName : string	Από τις βασικές συνιστώσες είναι η κλάση SceneLoader, η οποία όπως φανερώνει και το όνομά της είναι υπεύθυνη για το φόρτωμα των σκηνών της Unity που απαρτίζουν το παιχνίδι. Δημόσια χαρακτηριστικά της είναι το όνομα του επιπέδου που παίζεται (LevelName) και το σήμα της, δηλαδή το εικονίδιο που το αντιπροσωπεύει (LevelPicture), το prevCh που δείχνει σε ποια οθόνη θα οδηγείται ο παίκτης μετά από την επιλογή του ήρωα καθώς και οι ερωτήσεις που θα πρέπει να απαντηθούν (QuestionsToAnswer) σωστά σε κάθε κεφάλαιο ώστε αυτό να ολοκληρώνεται. Η playMythology αποθηκεύει την επιλογή ή όχι (true ή false) του χρήστη να παίξει με τη μυθολογία.
+ LevelPicture : Sprite	
+ prevCh : int	
+ QuestionsToAnswer : int	
+ playMythology : bool	
+ TerminateGame() : void	Τα ονόματα των μεθόδων είναι απολύτως προφανή. Η Μυθολογία ξεκινάει με τη StartUnit1(ch) και τον ακέραιο ch να δηλώνει το συγκεκριμένο κεφάλαιο/επίπεδο. Αντίστοιχα η StartUnit2(ch) για την προϊστορία.
+ QuitGame() : void	
+ ImportYourName() : void	
+ ChooseYourAvatar() : void	
+ GoToHomePage() : void	
+ StartMythology() : void	
+ StartPrehistory() : void	
+ StartUnit1(ch) : void	
+ StartUnit2(ch) : void	
+ ShowHelp() : void	
+ StartPrevious() : void	

MainPlayer	
- Avatar : Animator[]	Τελευταία και κατώτερη στην ιεραρχία, βρίσκεται η MainPlayer, η βασική κλάση του



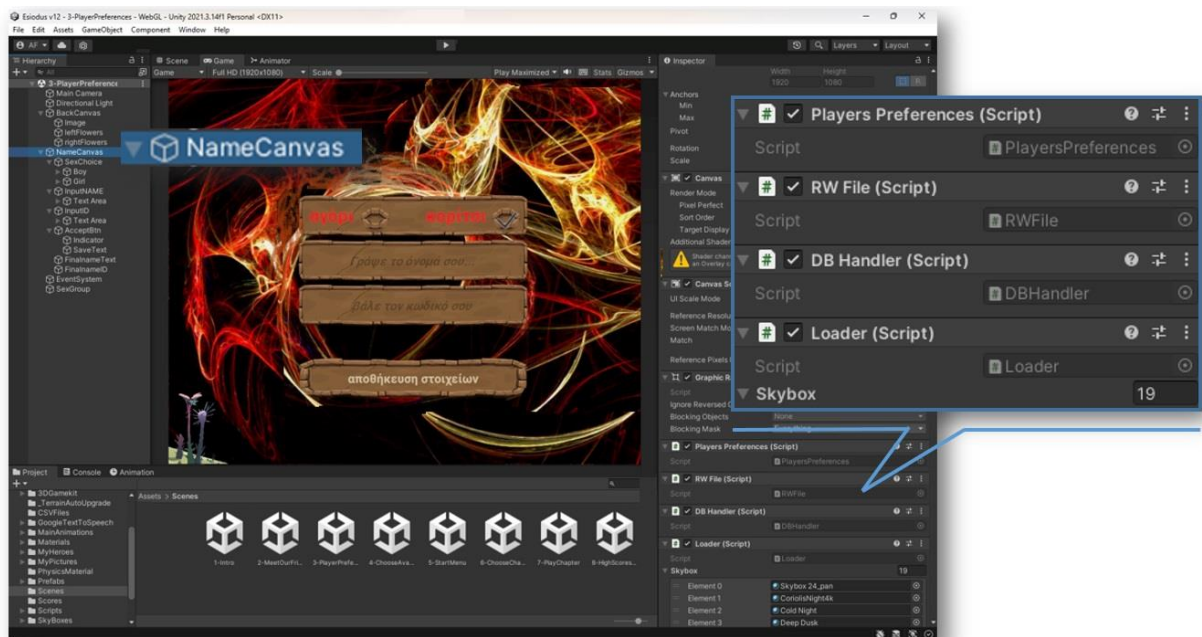
+ playerAnimator : Animator	<p>παιχνιδιού. Κληρονομεί όλα τα χαρακτηριστικά των προγόνων υπερκλάσεων της. Το CorrectAnswers είναι ο αριθμός των σωστών απαντήσεων που έχει πετύχει ο παίκτης στο επίπεδο, QuestionsPlayed είναι το σύνολο των ερωτήσεων που έχουν παιχθεί, Avatar[] είναι ο πίνακας των πέντε animators που αντιστοιχούν στους ήρωες, ενώ playerAnimator είναι αυτός που τελικά έχει επιλέξει ο παίκτης. Φυσικά υπάρχουν πολλά ακόμη χαρακτηριστικά, βοηθητικά για την εξέλιξη της εμφάνισης του παιχνιδιού, αλλά δεν έχει νόημα η περιγραφή τους στην παρουσίαση της κλάσης, όπου παίζουν ρόλο τα χαρακτηριστικά που την επηρεάζουν.</p> <p>Όσο για τις μεθόδους, αυτές αναπτύσσουν όλο το παιχνίδι. Εξελίσσουν όλο το σενάριο. Από τη</p>
+ CorrectAnswers : int	
+ QuestionsPlayed : int	
– Start() : void	
– Update() : void	
– FixedUpdate() : void	
– OnCollisionEnter(collision) : void	
– OnCollisionExit(collision) : void	
– DestroyStool(stool) : void	
– PlayerWillFall() : void	
– ShowStool(stool) : void	
– PlayWinningScenario (NewPosition) : void	
– ClimpUp() : void	
– Win() : void	
– UpdateScore() : void	

Start(), που αρχικοποιεί την οθόνη κάθε επιπέδου, ανάλογα με την πρόοδο του χρήστη, τον ήρωα που επιλέγει αυτός και το επίπεδο το οποίο θα παίζει, μέχρι την UpdateScore() που ενημερώνει την οθόνη για τον αριθμό των ερωτήσεων που απαντήθηκαν σωστά στο σύνολο αυτών που παίχτηκαν. Η Update() είναι υπεύθυνη για την αναγνώριση των πλήκτρων που πατάει ο παίκτης. Η FixedUpdate() εκμεταλλεύεται τα αποτελέσματα της προηγούμενης μεθόδου για να ορίσει τις κινήσεις του παίκτη, ανάλογα με τα πλήκτρα που πατήθηκαν, δηλαδή: αν θα πάει αριστερά ή δεξιά ή ποτέ μπορεί να πηδήξει, αν θα προχωρήσει πέρα από κάποια όρια ή όχι. Η OnCollisionEnter(collision) περιγράφει τι συμβαίνει στον ήρωα ανάλογα με το τι (collision) έρχεται σε σύγκρουση. Στην πραγματικότητα ελέγχει το παιχνίδι: αν ο ήρωας «συγκρουστεί» με το τουβλάκι της σωστής απάντησης, τότε κερδίζει μια ερώτηση, αν συγκρουστεί με ένα λάθος, χάνει την απάντηση. Όταν πέσει πάνω στην κεντρική πλατφόρμα για πρώτη φορά μετά από μια επιλογή, παίρνει μια καινούρια ερώτηση. Αντίστοιχα, η OnCollisionExit(collision) καθορίζει τι συμβαίνει όταν ο παίκτης θα βγει από μια σύγκρουση. Η DestroyStool() φροντίζει να δημιουργηθεί ένα αντικείμενο της κλάσης falseStool προκειμένου να πεταχτεί ως «χαμένο» και καλεί την PlayerWillFall() που φροντίζει να παιχτεί το animation της πτώσης του

ήρωα, με τους κατάλληλους ήχους ενώ τον ετοιμάζει για την επόμενη κίνησή του. Η ShowStool() δείχνει το σκαλοπάτι που έπρεπε να είναι στη θέση ενός λανθασμένου, όταν το τελευταίο πετάχτηκε. Με την PlayWinningSenario() γίνονται οι κινήσεις του ήρωα όταν κερδίζει μια ερώτηση. Ήχοι, κινήσεις, οδηγίες, όλα σχεδόν ελέγχονται με αυτή τη μέθοδο, σε συνδυασμό με την Win(), που συμπληρώνει τις ενέργειες νίκης. Επιπρόσθετα ελέγχει αν ολοκληρώθηκε το επίπεδο, προκειμένου να παρουσιαστεί το αναδυόμενο παράθυρο επιβράβευσης και παρουσίασης της βαθμολογίας που κατάφερε να πετύχει ο παίκτης και η οποία επίσης υπολογίζεται εδώ. Η ClimpUp(), ανεβάζει όλη την πλατφόρμα ψηλά κάθε φορά που δίνεται σωστή απάντηση.

### 3.4.3 Κλάσεις και Αντικείμενα στη Unity

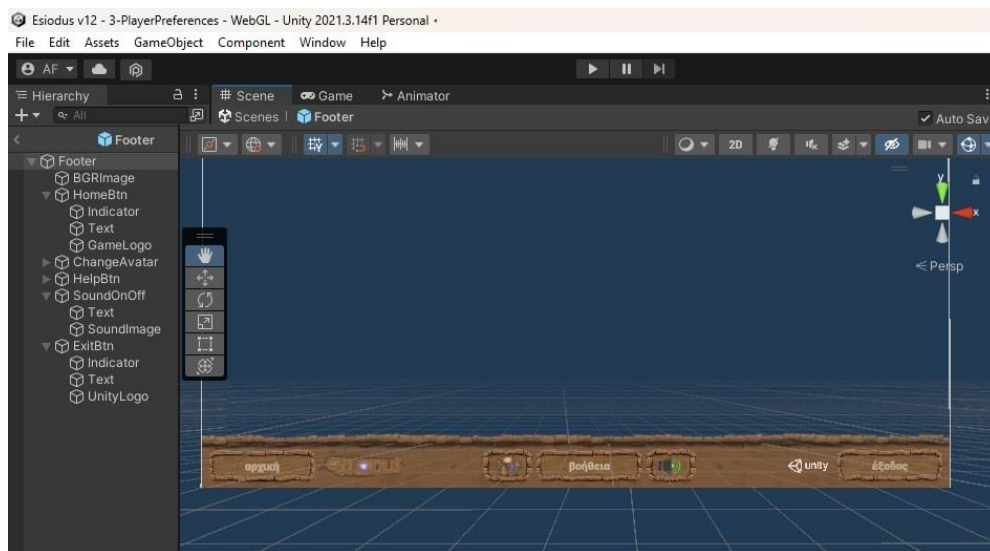
Η Unity δουλεύει ολόκληρη βασισμένη στα αντικείμενα. Τα αντικείμενα είναι σαν αόρατα δοχεία που μπορούν να γεμίσουν συνιστώσες για να κάνουν πράγματα. Κατέχουν μια θέση στο χώρο (position), μπορούν να περιστραφούν (rotation) ή να κλιμακωθούν (scale). Ανάλογα με το τι θα αντιπροσωπεύουν, μπορεί να έχουν μια συνιστώσα για να περιλαμβάνουν εικόνα (sprite renderer), ένα άκαμπτο σώμα (rigidbody) για να αποκτήσουν φυσικές ιδιότητες ή μια συνιστώσα για δυνατότητα συγκρούσεων (collider). Έχουν δικά τους χαρακτηριστικά και μεθόδους ανάλογα με τις συνιστώσες με τις οποίες είναι ενισχυμένα ενώ μπορούν να αποκτήσουν μεθόδους (scripts) για να εκτελούν συγκεκριμένες εργασίες.



Εικόνα 31: Αντικείμενα και Scripts

Οι κλάσεις που περιγράφηκαν στην υποενότητα 3.4.2 είναι αυτές που δημιούργησα εγώ προγραμματιστικά. Τα αντικείμενα μπορούν να ανήκουν σε αυτές χρησιμοποιώντας τις μεθόδους τους, όπως φαίνεται στην Εικόνα 31, όπου το αντικείμενο NameCanvas φαίνεται να χρησιμοποιεί τις μεθόδους των PlayersPreferences, RWFile, DBHandler και Loader.

Θα μπορούσε κάποιος να δει κλάσεις και στον editor της Unity. Η Unity σε κάθε σκηνή έχει ένα σύνολο αντικειμένων, η σειρά των οποίων καθορίζει την ιεραρχία τους και τη σειρά εμφάνισής τους. Μπορεί να έχουν παιδιά και τα παιδιά τους άλλα παιδιά και ούτω καθεξής. Μεταξύ αυτών μπορεί κανείς να φτιάξει αντικείμενα προκατασκευασμένα (prefabs) τα οποία να χρησιμοποιεί έτοιμα σε όποια οθόνη χρειαστεί μετά. Θα μπορούσε, λοιπόν, να δει κανείς αυτά τα prefabs ως κλάσεις, στιγμιότυπα των οποίων χρησιμοποιεί ο προγραμματιστής οπουδήποτε θελήσει. Σε αυτά μπορεί να ορίσει κάποια χαρακτηριστικά χωρίς να μπορεί να επέμβει στη βασική δομή, όπως ακριβώς γίνεται και στις κλάσεις. Τα prefabs φαίνονται από το διαφορετικό χρώμα που έχουν στην ιεραρχία του editor της Unity (Εικόνα 33). Υπάρχουν πολλά prefabs (τα βοηθητικά σκαλοπάτια, το σκαλοπάτι που δημιουργείται για τη λάθος απάντηση, οι καμβάδες της βοήθειας, κ.α.) ενώ από τα πιο προφανή prefabs του Ησίοδου είναι το μενού, όπως φαίνεται στην ακόλουθη εικόνα.

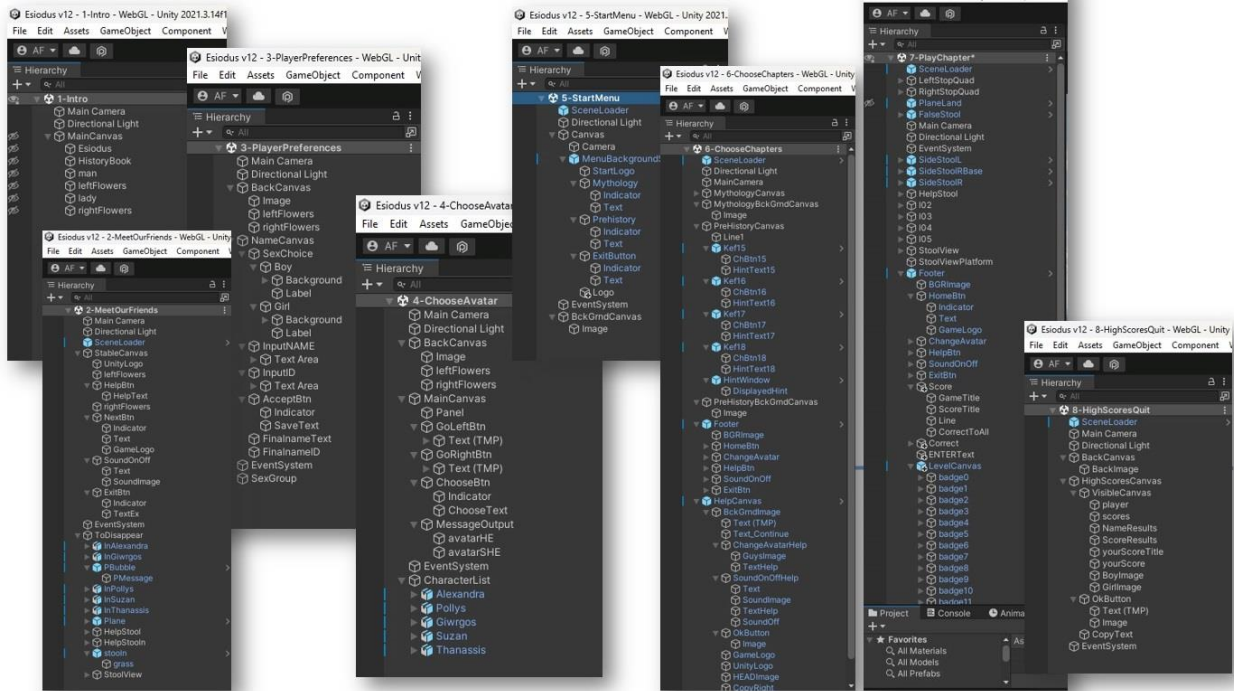


Εικόνα 32: Footer prefab

### 3.4.4 Οι σκηνές της Unity

Η ιεραρχία των αντικειμένων στην παιχνιδομηχανή, φαίνεται σε όλες τις σκηνές της Unity. Η Εικόνα 33 είναι απλά ενδεικτική του πώς εξελίσσεται ένα έργο στην

παιχνιδομηχανή! Χρειάζεται υπομονή, μελέτη, πολλή δουλειά και πολλή-πολλή όρεξη για να τα καταφέρει κανείς. Ο χρόνος πειραματισμού είναι κάτι πολύτιμο για όποιον μπορεί να τον διαθέσει. Τώρα που ολοκλήρωσα το έργο μου, μπορώ να πω πως νιώθω μια ασφάλεια και καταλαβαίνω τι γίνεται. Είμαι έτοιμη για ένα νέο έργο. Αν γνώριζα από την αρχή πώς δουλεύει η μηχανή... όλα θα ήταν ευκολότερα!

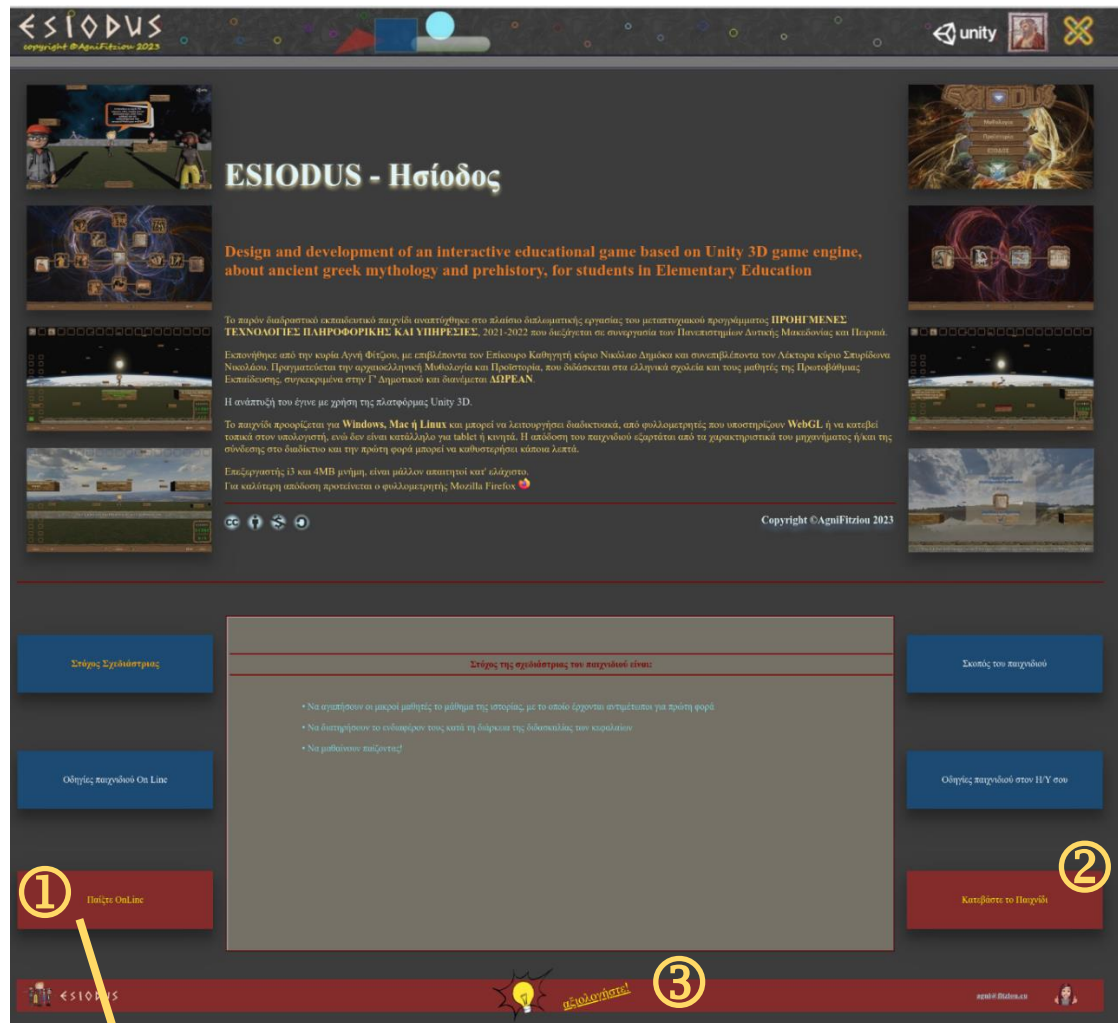


Εικόνα 33: Project Esiodus



## 4. Διανομή του Παιχνιδιού και Αξιολόγηση

Κάθε λογισμικό, εκπαιδευτικό ή όχι πρέπει να έχει τον τρόπο για να διανεμηθεί. Εγώ επέλεξα να φτιάξω μια σελίδα για το σκοπό αυτό, την <http://esiodus.eu>:



Εικόνα 34: Η Σελίδα του Ησιόδου



Από την επίσημη σελίδα του Ησιόδου, το παιχνίδι μπορεί

1. να παιχθεί online,
2. να κατεβεί τοπικά στον υπολογιστή του αιτούντος ή/και
3. να αξιολογηθεί

ΕΣΙΟΔΟΣ  
copyright © Agnifitziou 2023
unity

3

### Esiodus

Αξιολογήστε το παιχνίδι που παίξατε. σε λίγα μόνο δευτερόλεπτα! Μαςι μπορούμε να φτιάξουμε μια καλύτερη εκπαίδευση!

[Sign in to Google](#) to save your progress. [Learn more](#)

\* Required


Θέλω να πω τη γνώμη μου ως: \*

Εκπαιδευτικός

Μαθητής

Other: \_\_\_\_\_

Το παιχνίδι που χρησιμοποιήσα θεωρώ ότι είναι:



1 2 3 4 5

περιοριστικό      υποστηρικτικό

1 2 3 4 5

περίπλοκο      εύκολο

1 2 3 4 5

ασεπάρτες      ικανοποιητικό

1 2 3 4 5

μπερδισμένο      σαφές

1 2 3 4 5

βαρετό      συναρπαστικό

1 2 3 4 5

αδιάφορο      ενδιαφέρον


1 2 3 4 5

σημιατικό      εφιαρπητικό

1 2 3 4 5

συνθηλαμένο      πρωτοπαόρο

Ευχαριστώ πολύ για τη βοήθεια και το χρόνο σας!



Submit
Cancel

Google Forms - This content is neither created nor endorsed by Google.

Παιζει Σινα
Αρρηκη Σιλινα

ΕΣΙΟΔΟΣ
αποστολαιοιαν

Εικόνα 35: Φόρμα Αξιολόγησης Ησίοδου

## 4.1 Δοκιμές Alpha & Beta

Δοκιμές Alpha και Beta είναι οι μεθοδολογίες επικύρωσης πελατών (τύποι δοκιμών αποδοχής) που βοηθούν στην οικοδόμηση εμπιστοσύνης για την κυκλοφορία ενός προϊόντος, και συνεπώς οδηγεί στην επιτυχία του προϊόντος στην αγορά [47].

Παρόλο που και οι δύο βασίζονται σε πραγματικούς χρήστες και σε διαφορετικά σχόλια της κάθε ομάδας, καθοδηγούνται από διαφορετικές διαδικασίες, στρατηγικές και στόχους. Αυτοί οι δύο τύποι δοκιμών μαζί αυξάνουν την επιτυχία και τη διάρκεια ζωής ενός προϊόντος στην αγορά, με τις περισσότερες από τις μικρές δυσλειτουργίες να επιδιορθώνονται πριν από την τελική κυκλοφορία.

Οι φάσεις δοκιμών Alpha και Beta επικεντρώνονται κυρίως στην ανακάλυψη των σφαλμάτων από ένα ήδη δοκιμασμένο προϊόν και δίνουν μια σαφή εικόνα για το πώς το προϊόν χρησιμοποιείται πραγματικά από τους χρήστες σε πραγματικό χρόνο. Βοηθούν επίσης στην απόκτηση εμπειρίας με το προϊόν πριν από την κυκλοφορία του και τα πολύτιμα σχόλια εφαρμόζονται αποτελεσματικά για να αυξήσουν τη χρηστικότητα του προϊόντος.

### 4.1.1 Ησίοδος: Alpha Testing

Η δοκιμή Alpha είναι η τελευταία δοκιμή που γίνεται από τις ομάδες δοκιμών στον ιστότοπο ανάπτυξης πριν από την κυκλοφορία του λογισμικού για τη δοκιμή Beta. Στόχοι της δοκιμής Alpha είναι:

- η αξιολόγηση της ποιότητας του προϊόντος
- η εύρεση πιθανών σφαλμάτων
- ο έλεγχος αν όντως λειτουργεί το προϊόν και
- η διασφάλιση ετοιμότητας για τη δοκιμή Beta

Ο Ησίοδος, δεδομένου ότι είμαι εκπαιδευτικός και όχι εταιρεία ή ομάδα παραγωγής, δοκιμάστηκε σε τέσσερις ομάδες εφήβων περίπου των 25 ατόμων, μαθητών Α' και Β' Λυκείου, στο εργαστήριο Πληροφορικής του 2<sup>ου</sup> Γενικού Λυκείου Καστοριάς. Πέρα από τον ενθουσιασμό που έδειξαν οι testers μου, την επιθυμία που εκδήλωσαν να μετατραπεί το παιχνίδι σε κάποιο θέμα ενδιαφέροντος της ηλικίας τους και τις φιλότιμες προσπάθειές τους να βρουν υποψήφια λάθη για να τα διορθώσουμε, αξιολόγησαν το παιχνίδι μου ως ακολούθως:



Απάντησαν 96 άτομα, στο σύντομο ερωτηματολόγιο της σελίδας του Ησίοδου (Εικόνα 35), για το πόσο ικανοποιημένοι ήταν από τη χρήση της εφαρμογής. Πρόκειται για ένα ερωτηματολόγιο UEQ (User Experience Questionnaire). Οι πρώτες τέσσερις ερωτήσεις αφορούν στην πραγματική ποιότητα (Pragmatic Quality) και οι επόμενες τέσσερις αφορούν στην ηδονική ποιότητα (Hedonic Quality). Ενώ οι πραγματικές ιδιότητες αναφέρονται στην αντιληπτή χρησιμότητα, αποτελεσματικότητα και ευκολία χρήσης (τις λεγόμενες πτυχές χρησιμότητας και χρηστικότητας), οι ηδονικές ιδιότητες λαμβάνουν υπόψη τη «χαρά της χρήσης» και τονίζουν τη διέγερση, την ταύτιση και την επίκληση που δημιουργείται από τη χρήση ενός συστήματος ή ενός προϊόντος [48]. Μελετάται κάθε κατηγορία ξεχωριστά αλλά και το σύνολο των ιδιοτήτων (Overall).

Η κλίμακα του ερωτηματολογίου ήταν 5βάθμια, γιατί θεώρησα ότι για παιδιά θα ήταν δυσκολότερο να χειριστούν μια 7βάθμια. Η ανάλυση των αποτελεσμάτων έγινε, με τη βοήθεια συναρτήσεων των υπολογιστικών φύλλων (Παράρτημα II). Ωστόσο, επέλεξα να κάνω την ανάλυση και με το εργαλείο UEQ Data Analysis Tool του Dr. Martin Schrepp, επειδή διαθέτει μεγάλη βάση δεδομένων από 7βάθμια αντίστοιχα ερωτηματολόγια ικανοποίησης. Για το λόγο αυτό μετέτρεψα τις 5βάθμιες τιμές σε αντίστοιχες 7βαθμες [49], σύμφωνα με τον τύπο:

$$x_7 = (x_5 - 1)(6/4) + 1$$

ώστε να μπορεί να συγκριθεί η εφαρμογή με άλλες της βάσης. Μετά τη μετατροπή της κλίμακας, οι τιμές των απαντήσεων μετασχηματίστηκαν ανά στοιχείο και ανά συμμετέχοντα, αντί από 1 έως 7, από -3 έως 3. Το +3 αντιπροσωπεύει την πιο θετική τιμή και το -3 την πιο αρνητική τιμή.

Στον Πίνακα 6 φαίνονται διακυμάνσεις, τυπικές αποκλίσεις και μέσοι όροι, για την πραγματική και την ηδονική ποιότητα, οι οποίοι είναι ιδιαίτερα υψηλοί. Μπορούμε να ερμηνεύσουμε τα αποτελέσματα έχοντας υπόψιν μας ότι:

- οι τιμές μεταξύ -0,8 και 0,8 αντιπροσωπεύουν μια μέση αξιολόγηση της αντίστοιχης κλίμακας,
- οι τιμές > 0,8 αντιπροσωπεύουν μια θετική αξιολόγηση και
- οι τιμές < -0,8 αντιπροσωπεύουν μια αρνητική αξιολόγηση.

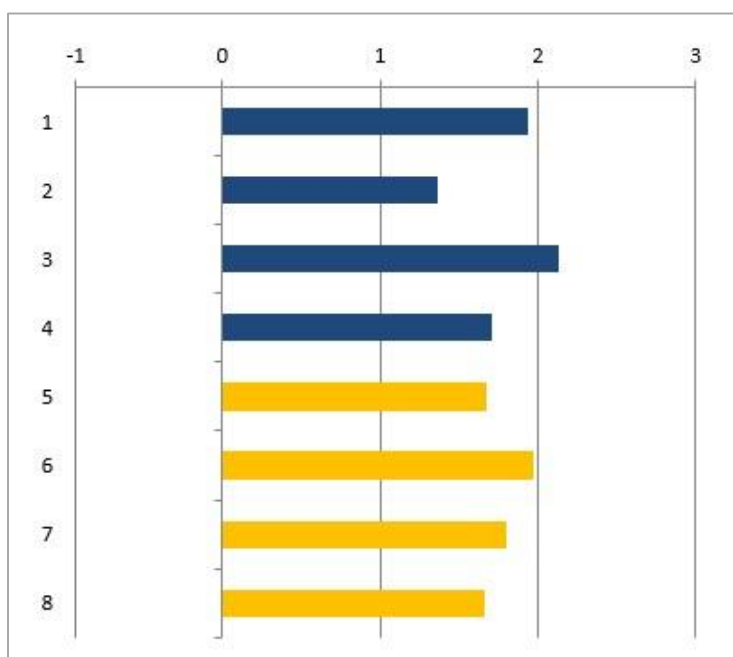
Ακόμη και σε 7βάθμιες κλίμακες με εύρος από -3 (φρικτά κακή) έως +3 (εξαιρετικά καλή) παρατηρούνται τιμές μόνο σε περιορισμένο εύρος. Αυτό οφείλεται στον

υπολογισμό των μέσων για μια σειρά διαφορετικών προσώπων με διαφορετικές απόψεις και τάσεις απαντήσεων.

**Πίνακας 6: Στατιστικά Αξιολόγησης Ανά Ερώτηση**

Ερ:	Μέση Τιμή	Διακύμανση	Τοπική Απόκλιση	Αρνητικό	Θετικό	Επιμέρους Διάκριση
π1	↑ 1,9	1,6	1,2	παρελκυστικό	υποστηρικτικό	πραγματική ποιότητα
π2	↑ 1,4	2,0	1,4	περίπλοκο	εύκολο	πραγματική ποιότητα
π3	↑ 2,1	1,3	1,1	ανεπαρκές	επαρκές	πραγματική ποιότητα
π4	↑ 1,7	2,2	1,5	μπερδεμένο	σαφές	πραγματική ποιότητα
η1	↑ 1,7	1,7	1,3	βαρετό	συναρπαστικό	ηδονική ποιότητα
η2	↑ 2,0	1,4	1,2	αδιάφορο	ενδιαφέρον	ηδονική ποιότητα
η3	↑ 1,8	1,6	1,3	συμβατικό	εφευρετικό	ηδονική ποιότητα
η4	↑ 1,7	1,8	1,4	συνηθισμένο	πρωτοπόρο	ηδονική ποιότητα

Είναι εξαιρετικά απίθανο να παρατηρηθούν τιμές πάνω από +2 ή κάτω από -2. Οπότε ακόμη και μια αρκετά καλή τιμή +1,5 για μια κλίμακα φαίνεται (από καθαρά οπτική άποψη στην κλίμακα από -3 έως +3) όχι τόσο θετική όσο είναι στην πραγματικότητα. Τα αποτελέσματα, δηλαδή είναι πολύ καλύτερα από ό,τι δείχνουν στην Εικόνα 37!



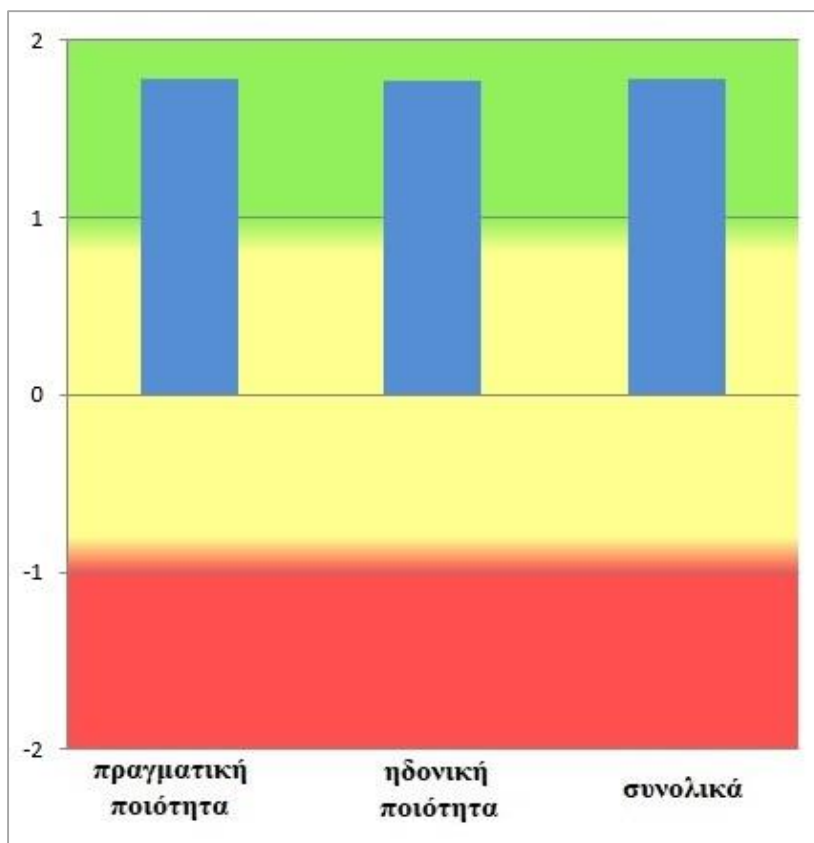
**Εικόνα 36: Μέσες Τιμές ανά Ερώτηση**

Οι μέσες τιμές ανά ερώτηση φαίνονται και στο διάγραμμα της Εικόνας 36, με ευνόητη αντιστοιχία των χρωμάτων πίνακα-εικόνας, ενώ στον αμέσως επόμενο πίνακα φαίνονται οι μέσοι όροι ανά κατηγορία και συνολικά:

**Πίνακας 7: Συνολικοί Μέσοι Όροι Αξιολόγησης**

<b>Ομαδοποιημένοι Μέσοι Όροι</b>	
πραγματική ποιότητα	↑ 1,781
ηδονική ποιότητα	↑ 1,773
για όλες τις ερωτήσεις	↑ 1,777

Μια πολύ ωραία αναπαράσταση του παραπάνω πίνακα, φαίνεται στην Εικόνα 37, όπου διακρίνεται ότι οι μέσες τιμές κυμαίνονται ξεκάθαρα στις επιθυμητές περιοχές, με απαγορευτικές τις κόκκινες, κάτω από -0.8. Οι μέσες τιμές του Ησίοδου υπερβαίνουν κατά πολύ το αποδεκτό 0.8 καταδεικνύοντας θετική αξιολόγηση.



**Εικόνα 37: Αποδεκτές Ζώνες Μέσων Τιμών**

Τα διαστήματα εμπιστοσύνης 95% για τις παραπάνω μέσες τιμές ανά ερώτηση και συγκεντρωτικά, ακολουθούν, στους Πίνακες 8 και 9 αντίστοιχα:

**Πίνακας 8: Διαστήματα Εμπιστοσύνης 95% (αναλυτικά)**

Διαστήματα Εμπιστοσύνης ( $p=0.05$ ) ανά Ερώτηση						
Ερ:	Μέση Τιμή	Τυπική Απόκλιση	N	Εμπιστοσύνη	Διάστημα Εμπιστοσύνης	
π1	1,938	1,249	96	0,250	1,688	2,187
π2	1,359	1,412	96	0,282	1,077	1,642
π3	2,125	1,145	96	0,229	1,896	2,354
π4	1,703	1,470	96	0,294	1,409	1,997
η1	1,672	1,322	96	0,264	1,408	1,936
η2	1,969	1,178	96	0,236	1,733	2,204
η3	1,797	1,262	96	0,252	1,544	2,049
η4	1,656	1,350	96	0,270	1,386	1,926

**Πίνακας 9: Διαστήματα Εμπιστοσύνης 95% (ανά ομάδα)**

Διαστήματα Εμπιστοσύνης ( $p=0.05$ ) ανά Ομάδα						
Ομάδα	Μέση Τιμή	Τυπική Απόκλιση	N	Εμπιστοσύνη	Διάστημα Εμπιστοσύνης	
πραγματική ποιότητα	1,781	1,027	96	0,205	1,576	1,987
ηδονική ποιότητα	1,773	1,013	96	0,203	1,571	1,976
για όλες τις ερωτήσεις	1,777	0,854	96	0,171	1,607	1,948

Το διάστημα εμπιστοσύνης είναι ένα μέτρο για την ακρίβεια της εκτίμησης του μέσου όρου. Όσο μικρότερο είναι το διάστημα εμπιστοσύνης τόσο μεγαλύτερη είναι η ακρίβεια της εκτίμησης. Το εύρος του διαστήματος εμπιστοσύνης εξαρτάται από τον αριθμό των διαθέσιμων δεδομένων και από το πόσο με συνέπεια τα άτομα έκριναν το αξιολογούμενο προϊόν. Όσο πιο συνεπής είναι η γνώμη τους τόσο μικρότερο είναι το διάστημα εμπιστοσύνης. Τα παραπάνω διαστήματα έχουν εύρος από 0,588 στην π4 το μεγαλύτερο, μέχρι 0,342 για τη μέση τιμή όλων των ερωτήσεων στο σύνολό τους!

Για να ολοκληρωθεί η συγκεκριμένη έρευνα, μπορούν να αναφερθούν και οι συσχετίσεις των στοιχείων ανά ομάδα-κλίμακα. Τα στοιχεία που ανήκουν στην ίδια ομάδα, δηλαδή οι τέσσερις πρώτες ερωτήσεις και οι τέσσερες επόμενες μεταξύ τους, θα πρέπει να παρουσιάζουν γενικά υψηλή συσχέτιση, για να θεωρείται αξιόπιστο το ερωτηματολόγιο. Ο συντελεστής Alpha (Cronbach, 1951) είναι ένα μέτρο για τη συνέπεια μιας κλίμακας. Αν και δεν υπάρχει γενικά αποδεκτός κανόνας για το πόση πρέπει να είναι η τιμή του συγκεκριμένου συντελεστή, θεωρείται αποδεκτό από τους πιο πολλούς συγγραφείς ότι μια ομάδα ερωτήσεων πρέπει να εμφανίζει τιμή

$Alpha > 0,7$  για να θεωρείται επαρκώς αξιόπιστη. Στην παρούσα έρευνα οι ερωτήσεις καλύπτουν και αυτήν την απαίτηση αξιοπιστίας, όπως φαίνεται στους τελευταίους πίνακες που ακολουθούν.

**Πίνακες 10: Δείκτες Alpha (Cronbach)**

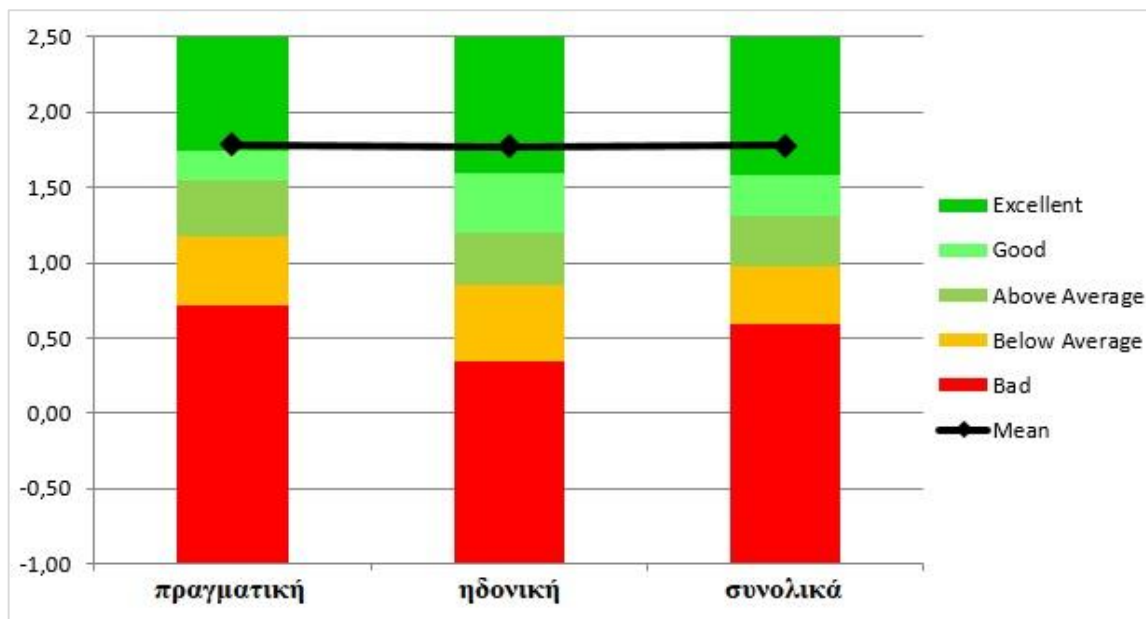
πραγματική ποιότητα		ηδονική ποιότητα	
στοιχεία	συσχέτιση	στοιχεία	συσχέτιση
1,2	0,30	5,6	0,66
1,3	0,57	5,7	0,48
1,4	0,49	5,8	0,44
2,3	0,36	6,7	0,43
2,4	0,48	6,8	0,43
3,4	0,64	7,8	0,58
<b>M.O.</b>	0,48	<b>M.O.</b>	0,50
<b>Alpha</b>	<b>0,78</b>	<b>Alpha</b>	<b>0,80</b>

Όπως φαίνεται από τη στατιστική ανάλυση των αποτελεσμάτων των ερωτήσεων των δοκιμαστών βοηθών μου, ο Ησίοδος, τόσο σε πραγματικό/ρεαλιστικό επίπεδο που έχει σχέση με τη λειτουργία και τη χρήση του, όσο και στο επίπεδο της ευχαρίστησης και της διασκέδασης αξιολογήθηκε ιδιαίτερα θετικά. Συγκρίνοντας, μάλιστα, τον Ησίοδο με άλλες εφαρμογές που αναλύθηκαν με το ίδιο εργαλείο, παίρνουμε τα θεαματικά αποτελέσματα του Πίνακα 11:

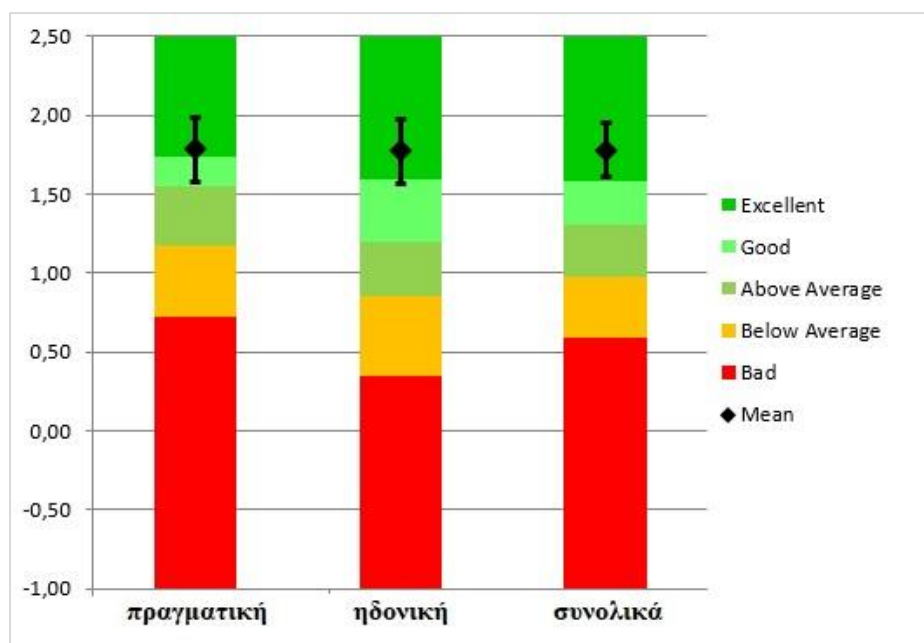
**Πίνακας 11: Σύγκριση με άλλες Αξιολογημένες Εφαρμογές**

Ιδιότητες	Μέση Τιμή	Σύγκριση με Σημείο Αναφοράς	Ερμηνεία
ρεαλιστική ποιότητα	1,781	<b>Excellent</b>	Στο 10% καλύτερων αποτελεσμάτων
ηδονική ποιότητα	1,773	<b>Excellent</b>	Στο 10% καλύτερων αποτελεσμάτων
όλες οι ερωτήσεις	1,777	<b>Excellent</b>	Στο 10% καλύτερων αποτελεσμάτων

Ο Ησίοδος κατατάσσεται στο 10% των καλύτερων εφαρμογών που αξιολογήθηκαν με αυτόν τον τρόπο. Αυτό φαίνεται παραστατικά στην Εικόνα 38, όπου με «μαύρο» παρουσιάζεται ο μέσος όρος, ο οποίος βρίσκεται στις περιοχές των «εξαιρετικά» αξιολογημένων εφαρμογών, ενώ στην Εικόνα 39 φαίνονται, πέρα από τη μέση τιμή και τα διαστήματα εμπιστοσύνης στα οποία αυτή εκτιμάται.



Εικόνα 38: Σύγκριση με Άλλες Εφαρμογές



Εικόνα 39: Μέσες Τιμές και Διαστήματα Εμπιστοσύνης

Οι υπολογισμένες μέσες τιμές ορίζονται σε σχέση με ένα δείκτη αναφοράς. Ο δείκτης αυτός καθορίζεται από τις τιμές ενός συνόλου δεδομένων αναφοράς, που περιέχει δεδομένα από 21.175 άτομα 468 μελετών που αφορούν διαφορετικά προϊόντα.

Η σύγκριση των αποτελεσμάτων του αξιολογούμενου προϊόντος με τα δεδομένα του δείκτη αναφοράς επιτρέπει συμπεράσματα για τη σχετική ποιότητα του πρώτου σε σύγκριση με άλλα προϊόντα.

#### 4.1.2 Ησίοδος: Beta Testing

Πρόκειται για την τελική φάση δοκιμών κατά την οποία το παιχνίδι μου διανεμήθηκε σε πραγματικές ομάδες χρηστών, σε δημοτικά σχολεία. Αφορά, δηλαδή σε δοκιμές που πραγματοποιήθηκαν από πραγματικούς χρήστες σε πραγματικό περιβάλλον. Στόχοι της δοκιμής Beta είναι:

- να αξιολογηθεί η ικανοποίηση των «πελατών»
- να εξασφαλιστεί η ετοιμότητα της κυκλοφορίας (για την παραγωγή)
- η εστίαση στη συλλογή προτάσεων/σχολίων και η αποτελεσματική τους αξιολόγηση και
- ο εντοπισμός της συμπάθειας του προϊόντος από τους «πελάτες»

Οι δοκιμές πραγματοποιήθηκαν στα σχολεία:

	3 <sup>ο</sup> Δημοτικό Σχολείο Άργους Ορεστικού, Νομός Καστοριάς	15 <sup>ο</sup> Δημοτικό Σχολείο Καβάλας
<b>Εκπαιδευτικός:</b>	Νικολαΐδης Πέτρος	Ανδρεάδης Ναούμ
<b>Τάξη:</b>	Γ'	Γ'
<b>Δύναμη Τάξης:</b>	15	19
<b>Σχόλια Δασκάλου:</b>	<p>Στα παιδιά άρεσαν οι ήρωες του παιχνιδιού, τους θύμιζαν παιδιά Γυμνασίου.</p> <p>Ο τρόπος του παιχνιδιού τούς ήταν οικείος γιατί μοιάζει με παιχνίδια που παίζουν στο σπίτι τους.</p> <p>Οι ερωτήσεις ήταν κατανοητές και σύμφωνες με τις γνώσεις τους.</p> <p>Μου ζήτησαν να το ξαναπαιξουν για να πετύχουν ένα καλύτερο βαθμό επιτυχίας, ώστε να ανέβουν στη γενική κατάταξη.</p> <p>Είναι ένα παιχνίδι που συνδυάζει μάθηση, ανταγωνισμό και διασκέδαση.</p>	<p>Το παιχνίδι είναι ευχάριστο και προκαλεί το ενδιαφέρον των παιδιών που βρίσκουν έναν παιγνιώδη τρόπο να εκφράσουν ό,τι έχουν μάθει στο μάθημα της ιστορίας.</p> <p>Υπάρχει καθολικό ενδιαφέρον για την εφαρμογή αυτή, ακόμα και από τους πιο αδύναμους μαθητές.</p> <p>Για το δάσκαλο είναι ένας διαφορετικός τρόπος αξιολόγησης και διεξαγωγής του μαθήματος, κάτι που δίνει ποικιλία στη διδασκαλία και την κάνει πιο ευχάριστη και αγαπητή στα παιδιά.</p>



Εκτός από την αξιολόγηση των δασκάλων που δούλεψαν τον Ησίοδο στην τάξη τους, τον μελέτησαν και αποφάνθηκαν γι' αυτόν, αξιόλογοι, πολύπειροι, συνταξιούχοι εκπαιδευτικοί, όπως φαίνεται ακολούθως. Τους ευχαριστώ όλους για τη συνδρομή!

---

## **ΑΓΓΕΛΟΥ ΘΕΟΦΑΝΗΣ**

**ΣΥΝΤΑΞΙΟΥΧΟΣ ΕΚΠΑΙΔΕΥΤΙΚΟΣ στο 16ο ΔΗΜΟΤΙΚΟ ΘΕΣΣΑΛΟΝΙΚΗΣ**

Τα σχόλια που έχω να κάνω για το συγκεκριμένο εκπαιδευτικό παιχνίδι είναι μόνο θετικά:

- Είναι ένα πολύ καλό εργαλείο εμπέδωσης της ύλης της ιστορίας της Γ' δημοτικού το οποίο θα αγαπήσουν τα παιδιά γιατί μαθαίνουν παίζοντας.
- Το περιβάλλον του παιχνιδιού είναι ελκυστικό, γνώριμο και εύκολα διαχειρίσιμο για τους μικρούς μαθητές .
- Η ύλη είναι χωρισμένη σε ενότητες και οι ερωτήσεις είναι απλές και κατανοητές.
- Μπορεί να παιχτεί από μικρότερους και μεγαλύτερους μαθητές είτε για μάθηση είτε για επανάληψη των γνώσεων.
- Οι χαρακτήρες που καλείται να επιλέξει ο μαθητής για να παίξει το παιχνίδι είναι φιλικοί.
- Ο στόχος του παιχνιδιού, που είναι να αγαπήσουν οι μαθητές το μάθημα της Ιστορίας και να μαθαίνουν παίζοντας, εξυπηρετείται απόλυτα.
- Τέλος, συνδυάζει το μάθημα της Ιστορίας με το μάθημα της Πληροφορικής κάνοντάς το ακόμη πιο ενδιαφέρον εμπλέκοντας και τον καθηγητή της Πληροφορικής για να δώσει απαντήσεις στις ερωτήσεις που θα γεννηθούν στα παιδιά σε σχέση με τη δημιουργία τέτοιων παιχνιδιών και άλλων εφαρμογών.
- Με λίγα λόγια το εκπαιδευτικό παιχνίδι της Αγνής Φίτζιου είναι μια πάρα πολύ αξιόλογη προσπάθεια που θα βοηθήσει τους μαθητές στην κατάκτηση της ιστορικής γνώσης παίζοντας, αλλά και για τους εκπαιδευτικούς και γονείς ένα πολύ καλό εργαλείο στην φαρέτρα τους.

Αγνή σε ευχαριστούμε και ευχόμαστε να υπάρξει και συνέχεια!!!

---

### **Ζαφείρης Ζαπρούδης**

**Συνταξιούχος, πρώην διευθυντής στο Δημοτικό Σχολείο Δημητρίτσιου Σερρών**

- Εύκολο στη χρήση του, διασκεδαστικό,
  - Ενδιαφέρον και αξιόλογο.
  - Λειτουργεί ως εκπαιδευτικό υποστηρικτικό υλικό για ενότητες που έχουν διδαχθεί.
  - Στην OnLine έκδοση αργεί η φόρτωση και η ταχύτητα επεξεργασίας εάν ο υπολογιστής δεν πληροί τις απαιτούμενες προδιαγραφές.
- 

### **Θωμάς Τζιβνίκος**

**Συνταξιούχος, πρώην διευθυντής του 16<sup>ου</sup> Δημοτικού Σχολείου**

- Το παιχνίδι το βρήκα έξυπνο και ευρηματικό. Επίσης δίνει την δυνατότητα στα παιδιά να ενσωματώσουν γνώσεις που με άλλον τρόπο είναι δυσκολότερα και όχι ελκυστικά όπως είναι στο παιχνίδι.
  - Είναι μια εφαρμογή που δείχνει στο μέλλον· πολύ ενδιαφέρουσα και αποτελεσματική.
  - Εκμεταλλεύεται την τάση της εποχής και την αγάπη των μαθητών για την πληροφορική, παράγοντας σημαντικά μορφωτικά αποτελέσματα.
- 

Κατά το διάστημα 03 Φεβρουαρίου 2023 έως 25 Φεβρουαρίου 2023, η σελίδα του Ησίοδου είχε 184 επισκέψεις, παίχτηκαν 101 OnLine παιχνίδια ενώ η έκδοση για το τοπικό παιχνίδι κατέβηκε σε υπολογιστές 57 φορές· νούμερα που δείχνουν συμπάθεια και ιδιαίτερα αυξημένη απήχηση σε όποιον ήρθε σε επαφή με τον Ησίοδο.

## Συμπεράσματα

---

Η αλματώδης ανάπτυξη της τεχνολογίας και των υπολογιστών, η αύξηση της υπολογιστικής δύναμης και η μεγάλη ευκολία στη χρήση του διαδικτύου, έχουν φέρει αλλαγές σε όλους της τομείς της ζωής, από τα εργαστήρια των ηλεκτρονικών υπολογιστών μέχρι τα σπίτια μας, από τις εταιρείες και τις επιχειρήσεις μέχρι τα σχολεία μας.

Οι μέθοδοι στην εκπαίδευση εξελίχθηκαν, εμπλουτίστηκαν και προσπαθούν να σταθούν αντάξιες της σύγχρονης ζωής. Η παιχνιδοποίηση και η μάθηση μέσω παιχνιδιών κερδίζει έδαφος σε μία κοινότητα, την εκπαιδευτική, που ακόμα αμφισβητεί την αξία της. Το παιχνίδι φέρνει μαθησιακά αποτελέσματα με έναν φυσικό και αβίαστο τρόπο, αφού προωθεί τη συμμετοχή, την αλληλεπίδραση, την ενίσχυση προσωπικών κινήτρων, την απόκτηση εμπειριών και τη δημιουργία νέων στρατηγικών. Η παιχνιδοποίηση στη μάθηση εκμεταλλεύεται τις μεθόδους σχεδιασμού παιχνιδιών προκειμένου να πετύχει ενεργοποίηση των μαθητών. Χρησιμοποιεί ως όχημα την τεχνολογία χωρίς να υπαινίσσεται τη δημιουργία ενός παιχνιδιού, ενώ τα εκπαιδευτικά παιχνίδια οφείλουν να έχουν τα χαρακτηριστικά της παιχνιδοποίησης. Το σημαντικό είναι ότι, εκπαιδευτικά παιχνίδια και παιχνιδοποίηση συνεισφέρουν καταλυτικά στην επιτυχία της εκπαιδευτικής διαδικασίας, παρέχοντας κατάλληλα εργαλεία για να ενεργοποιήσουν όλες τις αισθήσεις των μαθητών, επιτυγχάνοντας στο γνωστικό έργο και μετατρέποντας την εκπαιδευτική διαδικασία, από δυσνόητη ή βαρετή, σε κατανοητή και συναρπαστική.

Το ίδιο συναρπαστική έγινε και η σχεδίαση πολυσύνθετων προγραμμάτων μέσω του αντικειμενοστραφούς προγραμματισμού. Η αντικειμενοστραφής σχεδίαση ξεκινάει από την απλή ιδέα του να περιγραφούν οι ενέργειες που εφαρμόζονται πάνω σε δεδομένα, καταδεικνύοντας τα αντικείμενα και τις κλάσεις. Αντικείμενα είναι οι οντότητες του πραγματικού κόσμου που υπάρχουν γύρω μας, οτιδήποτε έχει χαρακτηριστικά και μεθόδους, ενώ οι κλάσεις χρησιμοποιούνται για την αναπαράστασή τους. Βασικές έννοιες όπως η αφαίρεση, η ενθυλάκωση, η κληρονομικότητα και ο πολυμορφισμός επιτρέπουν στον προγραμματιστή να γράψει πιο κατανοητό κώδικα, προσφέρουν ευκολότερη επαναχρησιμοποίηση λόγω αυτονομίας, ευκολότερη συντήρηση λόγω ενθυλάκωσης και δυνατότητα δημιουργίας ευχάριστων κ εύχρηστων διεπαφών χρήστη.

Μέσω της ευχάριστης διεπαφής χρήστη που δημιουργήθηκε με την αντικειμενοστραφή πλατφόρμα Unity 3D και βασισμένος στις αρχές και τις προταγές της μάθησης μέσω παιχνίων, γεννήθηκε ο Ησίοδος, ένα εκπαιδευτικό παιχνίδι για τους μαθητές της πρωτοβάθμιας εκπαίδευσης. Η ανάπτυξη ενός τέτοιου λογισμικού δεν ήταν καθόλου εύκολη υπόθεση αφού έπρεπε να ληφθούν αμέτρητες αποφάσεις, σχεδιαστικές και προγραμματιστικές και παιδαγωγικές, καλαισθητικές, τεχνικές, προωθητικές και ένα σωρό άλλες, που σε πραγματικές συνθήκες παραγωγής θα έπρεπε να αποσαφηνισθούν από αντίστοιχους ειδικούς επαγγελματίες. Η αξιολόγηση του παιχνιδιού έδειξε ότι είναι ένα αξιόλογο εργαλείο που πληροί τις σύγχρονες απαιτήσεις και θα μπορέσει, όπως επιβάλλεται, να διασκεδάσει τα παιδιά, μορφώνοντάς τα ή... να τα μορφώσει διασκεδάζοντάς τα!

Ο Ησίοδος είναι μία τεράστια προσπάθεια να παρουσιαστεί ένα πλήρες έργο, από όλες τις απόψεις. Να περιλαμβάνει όσες ερωτήσεις θα είχε κανονικά κάθε επίπεδο· να μην είναι παράδειγμα. Να περιλαμβάνει όλη την ύλη· να μην είναι ενδεικτικό. Να έχει έκδοση για online παιχνίδι αλλά και για τοπικό. Μπορεί να κατάφερα όλα τα παραπάνω, όμως, ο Προγραμματισμός δεν έχει τέλος αν είσαι προγραμματιστής! Πριν ολοκληρωθεί το έργο είχα ήδη πάμπολλες ιδέες στο μυαλό μου για επεκτάσεις.

Το ίδιο το παιχνίδι, θα μπορούσε να εμπλουτιστεί με απλές λειτουργίες εμφάνισης όλων των επιμέρους βαθμολογιών ανά επίπεδο. Θα μπορούσε να προστεθεί bonus επίπεδο με εικόνες και πηγές, τόσο στη μυθολογία όσο και στην προϊστορία. Δηλαδή οι απαντήσεις να δίνονται πάνω σε ερώτηση που βασίζεται σε εικόνα. Αν κάποιος επιθυμεί ως ανάδραση να δίνει τη σωστή απάντηση στο λάθος, να το κάνει. Αν προτιμάει επίσης κάποιος να συμπτύξει τις πρώτες σκηές σε μία, γίνεται. Αν και εγώ το απέφυγα λόγω ηλικίας του υποψήφιου κοινού μου, οι ενέργειες δημιουργίας λογαριασμού μπορούν να γίνουν με πιο αυστηρές προδιαγραφές δημιουργίας κωδικού πρόσβασης, που θα δίνεται από το χρήστη.

Για μεγαλύτερη αίσθηση 3D παιχνιδιού, θα μπορούσε κάποιος να βάλει όλα τα επίπεδα μέσα στο χώρο. Να δώσει τη δυνατότητα στον ήρωα να πηγαίνει και μπροστά και πίσω ψάχνοντας για το επόμενο επίπεδο, προσθέτοντας ένα πλάνο (site map) σε κάποιο σημείο της οθόνης, για να ξέρει πού να ψάξει (πιθανότατα για παιδιά μεγαλύτερης ηλικίας).

Υπόβαθρο για χρήση ήχου υπάρχει. Όποιος έχει ειδικές γνώσεις θα μπορούσε να εξελίξει το παιχνίδι για παιδιά με προβλήματα στην όραση, με χρήση αναγνώστη οθόνης ή να χρησιμοποιήσει γραμματοσειρές κατάλληλες για τη δυσλεξία, να αλλάξει τα background και τα χρώματα για να καλύψει ομάδες κοινού με ειδικές ανάγκες και ικανότητες.

Τέλος, το παιχνίδι μου είναι έτσι φτιαγμένο που να μην εξαρτάται σε πολλά πράγματα από το θέμα που πραγματεύεται. Θα ήταν λοιπόν εύκολα μεταφέρσιμο και μετατρέψιμο αν κάποιος ήθελε να ασχοληθεί με τη δυνατότητα διαχείρισής του. Να φτιάξει ένα βοηθητικό πρόγραμμα κατά το οποίο ο έχων τους κατάλληλους κωδικούς διαχειριστή να μπορεί να σβήνει ή να δημιουργεί αρχεία ερωτήσεων απελευθερωμένος από την προγραμματιστική τους μορφή. Κάθε εκπαιδευτικός, λοιπόν, θα μπορούσε να στήσει το δικό του παιχνίδι, με τις δικές του ερωτήσεις για το μάθημα που τον ενδιαφέρει.

Ήταν χαρά μου να βαφτίσω, να σχεδιάσω και να δημιουργήσω τον ΗΣΙΟΔΟ από το μηδέν μέχρι το πολύ αξιόλογο σημείο στο οποίο βρίσκεται!

## Βιβλιογραφία

---

- [1] I. Πυργιωτάκης, Εισαγωγή στην Παιδαγωγική Επιστήμη, Αθήνα: Ελληνικά Γράμματα, 2010.
- [2] I. Φύκαρης, Σύγχρονες διαστάσεις του διδακτικού έργου και ρόλου του εκπαιδευτικού, Θεσσαλονίκη: Αφοί Κυριακίδη, 2010.
- [3] UNESCO, «Global citizenship education: topics and learning objectives, Paris : UNESCO, 2015,» 2015. [Ηλεκτρονικό]. Available: [https://unesdoc.unesco.org/in/documentViewer.xhtml?v=2.1.196&id=p::usmarc\\_def\\_0000232993&file=/in/rest/annotationSVC/DownloadWatermarkedAttachment/attach\\_import\\_59f36c73-b9c9-4898-aea4-f93623d8a898%3F\\_%3D232993eng.pdf&locale=en&multi=true&ark=/ark:/48223](https://unesdoc.unesco.org/in/documentViewer.xhtml?v=2.1.196&id=p::usmarc_def_0000232993&file=/in/rest/annotationSVC/DownloadWatermarkedAttachment/attach_import_59f36c73-b9c9-4898-aea4-f93623d8a898%3F_%3D232993eng.pdf&locale=en&multi=true&ark=/ark:/48223).
- [4] ΕΟΠΠΕΠ, Εκπαίδευση Εκπαιδευτών Ενηλίκων - Εκπαιδευτικό Υλικό, Καστοριά: ΚΕΚ "Ορεστιάς" ΚΕΔΙΒΙΜ επιπέδου 2, Βιβλίο 6.
- [5] A. Meckley, «Observing children's play: Mindful methods. Paper presented to the International Toy Research Association,» London, 12 August 2002.
- [6] B. Scales, M. Almy, A. Nicolopoulou και S. Ervin-Tripp, Play and the social context of development in early care and education., New York: Teachers College Press, 1991.
- [7] M. Kostelnik, A. Soderman και . A. Whiren, Developmentally appropriate programs in early childhood education., New York: Merrill, 1993.
- [8] G. Maxim, The very young child (3rd ed.), Englewood Cliffs, NJ: Merrill/Prentice Hall, 1989.
- [9] S. Feeney, D. Christensen και E. Moravcik, Who am I in the lives of children? Englewood Cliffs, NJ.: Merrill, 1996.
- [10] J. Huizinga, Ο άνθρωπος και το παιχνίδι., Αθήνα: Γνώση, 2010.

- [11] Κ. Μποτσόγλου, *Υπαίθριοι χώροι παιχνιδιού και παιδί.*, Αθήνα: Gutenberg, 2010
- [12] Γ. Κιτσαράς, *Προσχολική Παιδαγωγική*, Αθήνα, 2001.
- [13] P. Gray, *Free to Learn: Why Unleashing the Instinct to Play Will Make Our Children Happier, More Self-Reliant, and Better Students for Life*. Basic Books 213.
- [14] M. Cecchini, «How Dramatic Play Can Enhance Learning,» 2015. [Ηλεκτρονικό] Available: <http://cpbucket.fiu.edu/1188-fiu01-red-4100-secrnc80381%2Fbenefits-of-dramatic-play.pdf>. [Πρόσβαση 20 02 2023].
- [15] Α. Μουστάκα, *Το παιχνίδι στην εκπαιδευτική διαδικασία κατά την προσχολική ηλικία*. Πτυχιακή εργασία. Παιδαγωγικό Τμήμα Προσχολικής Εκπαίδευσης Πανεπιστήμιο Θεσσαλίας, Βόλος, 2017.
- [16] Π. Ανδρεοπούλου, *Ο ρόλος του παιχνιδιού στην εκπαιδευτική διαδικασία*. Διπλωματική εργασία. ΠΜΣ «Νέες Μορφές Εκπαίδευσης και Μάθησης», Σχολή Ανθρωπιστικών Επιστημών, Πανεπιστήμιο Αιγαίου, Ρόδος, 2018.
- [17] Ε. Τζιάσιου, *Σενάριο διδασκαλίας κεφαλαίου Μαθηματικών Γ' Δημοτικού βασισμένο στις αρχές του Gamification, σε συνδυασμό με WebII εργαλεία*. Διπλωματική Εργασία, Πρόγραμμα μεταπτυχιακών σπουδών στην Οικονομική της Εκπαίδευσης και Διαχείριση Εκπαιδευτικών Μονάδων., Τμήμα Οικονομικών Επιστημών Πειραιά, Πειραιάς, 2021.
- [18] D. Helgason, Trends, Unity Technologies Blog, 2010.
- [19] S. Deterding, R. Dixon, R. Khaled και L. Nacke, «From Game Design Element to Gamefulness: Defining “Gamification”..» σε *15th International Academic MindTrek Conference: Envisioning Future Media Environments*, MindTrek '11 pp. 9-15, 2011.
- [20] Ε. Αμπατζίδου, *Τεχνολογίες Παιχνιδοποίησης (Gamification) στην κατάρτιση - εκπαίδευση προσωπικού επιχειρήσεων και οργανισμών*. Διπλωματική Εργασία ΔΠΜΣ Τμήματος Εφαρμοσμένης Πληροφορικής, Πανεπιστήμιο Μακεδονίας Θεσσαλονίκη, 2020.



- [21] Σ. Δημητριάδου, Παιχνιδοποίηση στην ηλεκτρονική μάθηση: Σχεδιασμός συστήματος Παιχνιδοποίησης και εφαρμογή του στην ηλεκτρονική πλατφόρμα Moodle. Μεταπτυχιακή Εργασία, Τμήμα Ψηφιακών Συστημάτων, Πανεπιστήμιο Πειραιά, Πειραιάς, 2016.
- [22] K. Werbach και D. Hunter, For the win: How game thinking can revolutionize your business., Wharton Digital Press, 2012.
- [23] Π. Γιοβανόπουλος και Α. Κατσιούλας,, Ενίσχυση της μαθησιακής αποτελεσματικότητας των Στρατηγικών Συνεργατικής Μάθησης, (Collaborative Strategies), με Παιχνιδοποίηση (Gamification) και Web2, στην Πρωτοβάθμια Εκπαίδευση. Διπλωματική Εργασία, Πανεπιστήμιο Αιγαίου, Σάμος, 2018.
- [24] B. Rogers και W. Rogers, Classroom Behaviour: A Practical Guide to Effective Teaching, Behaviour Management and Colleague Support, SAGE.
- [25] C. Pappas, «The Top Gamification Statistics And Facts For 2015 You Need To Know,» elearningindustry.com, 2015. [Ηλεκτρονικό]. Available <https://elearningindustry.com/top-gamification-statistics-and-facts-for-2015>. [Πρόσβαση 2023 01 31].
- [26] Γ. Φλουρής, Αναλυτικά Προγράμματα για μια νέα εποχή στην εκπαίδευση (4η εκδ.), Αθήνα: Γρηγόρη, 1995.
- [27] Ε. Ανδρεάδου, Ν. Νταφόπουλος και Ι. Σταθοπούλου, Η Παιχνιοποίηση στην Εκπαίδευση, ΕΕΠΕΚ: Επιμορφωτικό Υλικό Σεμιναρίου, 2022-2023.
- [28] K. M. Kapp, The gamification of learning and instruction: game-based methods and strategies for training and education., John Wiley & Sons, 2012.
- [29] Π. Ξωχέλης, Εισαγωγή στην Παιδαγωγική - Θεμελιώδη προβλήματα της Παιδαγωγικής Επιστήμης (6η εκδ.), Θεσσαλονίκη: Αφοί Κυριακίδη, 2005.
- [30] Θ. Γιαννοπούλου, Η Επίδραση της Παιχνιδοποίησης στην Εμπλοκή των Εκπαιδευομένων: Μια Συστηματική Βιβλιογραφική Επισκόπηση. Διπλωματική εργασία, Μεταπτυχιακό Πρόγραμμα Σπουδών στις Σπουδές στην Εκπαίδευση Σχολή Ανθρωπιστικών Σπουδών, Ελληνικό Ανοικτό Πανεπιστήμιο, Πάτρα, 2020

- [31] C. C. I. Muntean, Raising engagement in e-learning through gamification. The 6th International Conference on Virtual Learning ICVL 2011, 1, 323–329., 2011.
- [32] K. E. DiCerbo, Integrating Data Across Digital Activities. *Journal of Learning Media, and Technology*, 41, 233-251, 2016.
- [33] Α. ΒΑΚΑΛΗ, Η. ΓΙΑΝΝΟΠΟΥΛΟΣ, Ν. ΙΩΑΝΝΙΔΗΣ, Χ. ΚΟΙΛΙΑΣ, Κ. ΜΑΛΑΜΑΣ, Ι. ΜΑΝΩΛΟΠΟΥΛΟΣ και Π. ΠΟΛΙΤΗΣ, Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον - Γ' Γενικού Λυκείου, Ινστιτούτο Τεχνολογίας Υπολογιστών και Εκδόσεων "Διόφαντος", 2010.
- [34] B. Liskov και J. Guttag, *Program Development in Java: Abstraction, Specification and Object-Oriented Design*, Addison-Wesley Professional, 2000.
- [35] Σ. Μαϊστρέλλης, Ε. Καλύβη και Μ. Μιχαήλ, *Ιστορία - Από τη Μυθολογία στη Ιστορία Γ' Δημοτικού*, Ινστιτούτο Τεχνολογίας Υπολογιστών και Εκδόσεων "Διόφαντος", 2014.
- [36] wikipedia, «Ησίοδος,» [Ηλεκτρονικό]. Available <https://el.wikipedia.org/wiki/Ησίοδος>. [Πρόσβαση 15 11 2022].
- [37] R. Duke, «Gantt.com,» [Ηλεκτρονικό]. Available: <https://www.gantt.com/> [Πρόσβαση 30 11 2022].
- [38] TairaGames, «Unity vs Unreal,» [Ηλεκτρονικό]. Available <https://www.youtube.com/watch?v=baAPN0TCwkc>. [Πρόσβαση 28 10 2022].
- [39] Perforce, «What Are the Best Game Engines?,» [Ηλεκτρονικό]. Available <https://www.perforce.com/blog/vcs/most-popular-game-engines>. [Πρόσβαση 2-01 2023].
- [40] Unity, «Unity Learn,» [Ηλεκτρονικό]. Available: <https://learn.unity.com/> [Πρόσβαση 30 10 2022].
- [41] S. Persson, «LEARN UNITY - The Most BASIC TUTORIAL I'll Ever Make,» 2-07 2020. [Ηλεκτρονικό]. Available <https://www.youtube.com/watch?v=pwZpJzpE2lQ>. [Πρόσβαση 11 11 2022].

- [42] B. Shneiderman, C. Plaisant, M. Cohen, S. Jacobs και N. Elmqvist, *Designing the User Interface: Strategies for Effective Human-Computer Interaction: Sixth Edition*, Pearson, 2016.
- [43] R. Mayer, *The Cambridge Handbook of Multimedia Learning*, Cambridge University Press, 2005.
- [44] R. Gagne, L. Briggs και W. Wager, *Principles of Instructional Design (4th Ed.)* Fort Worth: TX: HBJ College Publishers, 1992.
- [45] tutorialspoint, «Learn UML,» tutorialspoint, [Ηλεκτρονικό]. Available <https://www.tutorialspoint.com/uml/index.htm>. [Πρόσβαση 02 2023].
- [46] UnityDocumentation, «Unity Manual Version: 2021.3,» [Ηλεκτρονικό] Available: <https://docs.unity3d.com/Manual/index.html>. [Πρόσβαση 30 11 2022]
- [47] myservername.com, «myservername.com,» [Ηλεκτρονικό]. Available <https://el.myservername.com/alpha-testing-beta-testing>. [Πρόσβαση 01 02 2023]
- [48] T. Merčun και M. Žumer, «Exploring the influences on pragmatic and hedonic aspects of user experience. *Information Research*, 22(1), CoLIS paper 1621,» 2017. [Ηλεκτρονικό]. Available: <http://InformationR.net/ir/22/1/colis/colis1621.html>. [Πρόσβαση 2023 02 14].
- [49] J. Lewis και J. Sauro, «How to Convert Between Five- and Seven-Point Scales,» *measuringu.com*, 17 06 2020. [Ηλεκτρονικό]. Available <https://measuringu.com/convert-point-scales/>. [Πρόσβαση 14 02 2023].
- [50] R. Mucchielli, *Les méthodes actives: Dans la pédagogie des adultes*, ESF, 2016.

## Παράρτημα I - Κώδικας

---

Στο παρόν παράρτημα παρουσιάζονται τα C# scripts που αντιστοιχούν στις κλάσεις που παρουσιάστηκαν στο υποκεφάλαιο **3.4 Αναλυτική Παρουσίαση των Κλάσεων**, με την ίδια σειρά.

### public class BckgrndScaler

---

```
1 using UnityEngine;
2 using UnityEngine.UI;
3
4 public class BckgrndScaler : MonoBehaviour
5 {
6     public Image backgroundImage;
7     RectTransform rt;
8     float ratio;
9
10    void Start()
11    {
12        rt = backgroundImage.rectTransform;
13        ratio = backgroundImage.sprite.bounds.size.x /
14              backgroundImage.sprite.bounds.size.y;
15
16    void Update()
17    {
18        if (!rt)
19            return;
20
21        if (Screen.height * ratio >= Screen.width)
22        {
23            rt.sizeDelta = new Vector2(Screen.height * ratio, Screen.height);
24        }
25        else
26        {
27            rt.sizeDelta = new Vector2(Screen.width, Screen.width / ratio);
28        }
29    }
30 }
```

## public class ButtonsFunctions

---

```
1 using System.Collections;
2 using UnityEngine;
3 using TMPro;
4 using UnityEngine.EventSystems;
5
6 public class ButtonsFunctions : MonoBehaviour, IPointerEnterHandler, IPointerExitHandler
7 {
8     public TextMeshProUGUI HintText;
9     private float timeToWait = 0.5f;
10
11     public void OnPointerEnter(PointerEventData eventData)
12     {
13         StopAllCoroutines();
14         StartCoroutine(StartTimer());
15     }
16
17     public void OnPointerExit(PointerEventData eventData)
18     {
19         StopAllCoroutines();
20         HintManager.OnMouseExit();
21     }
22
23     private void ShowMessage()
24     {
25         HintManager.OnMouseHover(HintText.text, Input.mousePosition);
26     }
27
28     private IEnumerator StartTimer()
29     {
30         yield return new WaitForSeconds(timeToWait);
31         ShowMessage();
32     }
33 }
```

## public class CharacterSelection

---

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.UI;
5 using TMPro;
6
7 public class CharacterSelection : MonoBehaviour
8 {
9     private GameObject[] CharacterList;
10    private string[] CharacterName = { "την Αλεξάνδρα", "τον Πόλυ", "το
        Γιώργο", "τη Σούζαν", "το Θανάση" };
11    private string tmpBtnWithName;
12    public TextMeshProUGUI BtnWithName, Message;
13    public Image HE, SHE;
14    public static int index = 4;
15
16    void Start()
17    {
18        Time.timeScale = 1.0f;
19
20        if (PlayersPreferences.boy)
21        {
22            tmpBtnWithName = "Ο " + PlayersPreferences.PlayersName + "\n" +
                BtnWithName.text;
23            Message.text = "Καλωσορίζουμε τον φίλο και παίκτη μας!";
24            HE.enabled = true;
25        }
26        else
27        {
28            tmpBtnWithName = "Η " + PlayersPreferences.PlayersName + "\n" +
                BtnWithName.text;
29            Message.text = "Καλωσορίζουμε τη φίλη παίκτριά μας!";
30            SHE.enabled = true;
31        }
32        BtnWithName.text = tmpBtnWithName + CharacterName[index];
33
34        CharacterList = new GameObject[transform.childCount];
35
36        for (int i = 0; i < transform.childCount; i++)
37            CharacterList[i] = transform.GetChild(i).gameObject;
38
39        foreach (GameObject avatar in CharacterList)
40            avatar.SetActive(false);
41
42        if (CharacterList[index])
43            CharacterList[index].SetActive(true);
44    }
45
46    public void ToggleLeft()
47    {
48        CharacterList[index--].SetActive(false);
49
50        if (index < 0)
51            index = CharacterList.Length - 1;
52
53        CharacterList[index].SetActive(true);

```

```
54     BtnWithName.text = tmpBtnWithName + CharacterName[index];
55 }
56
57 public void ToggleRight()
58 {
59     CharacterList[index++].SetActive(false);
60
61     if (index == CharacterList.Length)
62         index = 0;
63
64     CharacterList[index].SetActive(true);
65     BtnWithName.text = tmpBtnWithName + CharacterName[index];
66 }
67
68 public void ConfirmChoice()
69 {
70     UnityEngine.SceneManagement.SceneManager.LoadScene(SceneLoader.prevCh);
71 }
72 }
```



## public class falseStool

---

```
1 using UnityEngine;
2
3 public class falseStool : MonoBehaviour
4 {
5     public static Rigidbody rbStool;
6     public static Transform StoolPosition;
7     public static Renderer rnder;
8
9     void Start()
10    {
11        rbStool = GetComponent<Rigidbody>();
12        rnder= GetComponent<Renderer>();
13        StoolPosition = GetComponent<Transform>();
14    }
15
16    private void OnCollisionEnter(Collision collision)
17    {
18        if (collision.gameObject.name == "PlaneLand")
19        {
20            rbStool.isKinematic = true;
21            rbStool.constraints = RigidbodyConstraints.FreezeAll;
22        }
23    }
24 }
```

## public class FirstScreen

---

```

1  using UnityEngine;
2  using TMPro;
3
4  public class FirstScreen : MonoBehaviour
5  {
6      public Animator Giwrgos, Suzan, Thanassis;
7      public AudioSource AlexandrasVoice, PollysVoice, GiwrgosVoice,           ↗
8          SuzansVoice, ThanassisVoice, WalkOnMud;
9      private AudioSource Voice;
10     public GameObject PollysBubble, PollysText;
11     public TextMeshProUGUI GoOnBtn, MessageBtn;
12     private int EnterWasPressed;
13     private bool playedPollys, playedGiwrgos = true, playedSuzan = true,     ↗
14         playedThanassis = true;
15     public static bool getKeys = true;
16
17     void Awake()
18     {
19         SoundFunction.soundIsOn = true;
20         Giwrgos.enabled = false;
21         Suzan.enabled = false;
22         Thanassis.enabled = false;
23         Voice = AlexandrasVoice;
24         AlexandrasVoice.Play();
25     }
26
27     private void Update()
28     {
29         if (SoundFunction.soundIsOn) Voice.volume = 0.5f;
30         else Voice.volume = 0f;
31
32         if (!getKeys)
33         {
34             return;
35         }
36
37         if (Input.GetKeyDown(KeyCode.Return))
38         {
39             EnterWasPressed++;
40         }
41     }
42
43     private void FixedUpdate()
44     {
45         NextAction();
46     }
47
48     public void NextActionOnClick()
49     {
50         if (!getKeys)
51         {
52             return;
53         }
54         EnterWasPressed++;

```

```

55     NextAction();
56 }
57
58 void NextAction()
59 {
60     if (EnterWasPressed == 1 && !playedPollys)
61     {
62         Destroy(GameObject.Find("ABubble"));
63         PollysText.GetComponent<Renderer>().enabled = true;
64         PollysBubble.GetComponent<Renderer>().enabled = true;
65         MessageBtn.text = "Πίεσε <Enter> και περίμενε·\nθα μιλήσει κι ο  ⤴
        Γιώργος.";
66         AlexandrasVoice.Pause();
67         Voice = PollysVoice;
68         PollysVoice.Play();
69         MessageBtn.color = new Color(0.8f, 1.0f, 0.8f, 1.0f); // ⤴
        (0.126f, 0.151f, 0.119f, 1.0f);
70         playedPollys = true;
71         playedGiwrgos = false;
72         return;
73     }
74
75     if (EnterWasPressed == 2 && !playedGiwrgos)
76     {
77         GiwrgosAction();
78     }
79
80     if (AnimatorIsPlaying(Giwrgos, "Idle") && EnterWasPressed == 2)
81     {
82         getKeys = true;
83         MessageBtn.text = "Πίεσε <Enter> και περίμενε·\nαργά - αργά... ⤴
        έρχεται η Σούζαν.";
84         MessageBtn.color = new Color(1.0f, 1.0f, 0.4f, 1.0f); // ⤴
        (0.736f, 0.646f, 0.044f, 1.0f);
85         playedSuzan = false;
86     }
87
88     if (EnterWasPressed == 3 && !playedSuzan)
89     {
90         SuzanAction();
91     }
92
93     if (AnimatorIsPlaying(Suzan, "Idle") && EnterWasPressed == 3)
94     {
95         getKeys = true;
96         MessageBtn.text = "Πίεσε <Enter> και περίμενε·\nτελευταίος έρχεται ⤴
        ο Θανάσης.";
97         MessageBtn.color = new Color(0.0f, 1.0f, 1.0f, 1.0f); // ⤴
        (0.055f, 0.220f, 0.341f, 1.0f);
98         playedThanassis = false;
99     }
100
101     if (EnterWasPressed == 4 && !playedThanassis)
102     {
103         ThanassisFirstAction();
104     }

```

```

105
106     if (AnimatorIsPlaying(Thanassis, "Left Turn 90") && EnterWasPressed == 5)
107     {
108         WalkOnMud.Pause();
109         Voice = ThanassisVoice;
110         ThanassisVoice.Play();
111         EnterWasPressed++;
112     }
113
114     if (AnimatorIsPlaying(Thanassis, "Thanassis is still Talking") && EnterWasPressed == 6)
115     {
116         MessageBtn.text = "Για να συνεχίσεις μετά το θάναση,\ηπάτησε το κουμπι <συνέχεια>.";
117         MessageBtn.color = new Color(1.0f, 1.0f, 0.8f, 1.0f); // (0.831f, 0.765f, 0.639f, 1.0f);
118         playedThanassis = true;
119         GoOnBtn.text = "συνέχεια";
120         GoOnBtn.color = Color.green;
121         playedThanassis = true;
122         return;
123     }
124 }
125
126 bool AnimatorIsPlaying(Animator animator, string stateName)
127 {
128     return animator.GetCurrentAnimatorStateInfo(0).IsName(stateName);
129 }
130
131 void GiwrgosAction()
132 {
133     PollysVoice.Pause();
134     Voice = GiwrgosVoice;
135     GiwrgosVoice.Play();
136     Destroy(GameObject.Find("PBubble"));
137     getKeys = false;
138     Giwrgos.enabled = true;
139     playedGiwrgos = true;
140 }
141
142 void SuzanAction()
143 {
144     GiwrgosVoice.Pause();
145     Voice = SuzansVoice;
146     SuzansVoice.Play();
147     Destroy(GameObject.Find("GBubble"));
148     getKeys = false;
149     Suzan.enabled = true;
150     playedSuzan = true;
151 }
152
153 void ThanassisFirstAction()
154 {
155     SuzansVoice.Pause();
156     Voice = WalkOnMud;

```

```
157     WalkOnMud.Play();
158     Destroy(GameObject.Find("SBubble"));
159     getKeys = false;
160     Thanassis.enabled = true;
161     EnterWasPressed++;
162 }
163 }
```

## public class FollowPlayer

---

```
1 using UnityEngine;
2
3 public class FollowPlayer : MonoBehaviour
4 {
5     public Transform player;
6     private Vector3 offset;
7
8     private void Start()
9     {
10        offset = new Vector3(0, 1, -23);
11    }
12
13    void Update()
14    {
15        transform.position = player.position + offset;
16    }
17 }
```

## public class HelpBtnsFunctionality

---

```
1 using UnityEngine;
2
3 public class HelpBtnsFunctionality : MonoBehaviour
4 {
5     public Animator animator;
6     public string AnimationName;
7     public Canvas canvas;
8     public GameObject ToDisappear;
9
10    public void OnClick()
11    {
12        animator.SetTrigger("startAnimation");
13        animator.Play(AnimationName, 0, 0.0f);
14    }
15
16    public void OnExitAnimation()
17    {
18        animator.ResetTrigger("startAnimation");
19    }
20
21    public void CloseHelp()
22    {
23        canvas.enabled = false;
24        GameObject.Find("Footer").GetComponent<Canvas>().enabled = true;
25        Time.timeScale = 1.0f;
26    }
27
28    public void ShowTerminateGame()
29    {
30        canvas.enabled = false;
31        if (AnimationName == "Disappear")
32        {
33            Vector3 Help = ToDisappear.transform.position;
34            Help.y = -200;
35            ToDisappear.transform.position = Help;
36        }
37    }
38 }
```



## public class HintManager

---

```
1 using System;
2 using UnityEngine;
3 using UnityEngine.UI;
4 using TMPro;
5
6 public class HintManager : MonoBehaviour
7 {
8     public TextMeshProUGUI DisplayedHint;
9     public RectTransform HintWindow;
10    public static Action<string, Vector2> OnMouseHover;
11    public static Action OnMouseExit;
12
13    void Start()
14    {
15        HideHint();
16    }
17
18    private void OnEnable()
19    {
20        OnMouseHover += ShowHint;
21        OnMouseExit += HideHint;
22    }
23
24    private void OnDisable()
25    {
26        OnMouseHover -= ShowHint;
27        OnMouseExit -= HideHint;
28    }
29
30    private void ShowHint(string hint, Vector2 mousePos)
31    {
32        DisplayedHint.text = hint;
33        HintWindow.sizeDelta = new Vector2(DisplayedHint.preferredWidth > 300 ?
34        300 : DisplayedHint.preferredWidth, DisplayedHint.preferredHeight);
35        HintWindow.gameObject.SetActive(true);
36        HintWindow.transform.position = new Vector2(mousePos.x +
37        HintWindow.sizeDelta.x / 2, mousePos.y);
38    }
39
40    private void HideHint()
41    {
42        DisplayedHint.text = default;
43        HintWindow.gameObject.SetActive(false);
44    }
45 }
```

## public class LoadPseudoLevel

---

```
1 using UnityEngine;
2 using UnityEngine.UI;
3
4 public class LoadPseudoLevel : MonoBehaviour
5 {
6     public Canvas MythCanvas, MythBgrCanvas, PreHiCanvas, PreHiBgrCanvas;
7     private int fromPart, toPart;
8
9     void Start()
10    {
11        if (SceneLoader.playMythology)
12        {
13            PreHiCanvas.enabled = false;
14            PreHiBgrCanvas.enabled = false;
15            MythCanvas.enabled = true;
16            MythBgrCanvas.enabled = true;
17            fromPart = 0;
18            toPart = 15;
19        }
20        else
21        {
22            MythCanvas.enabled = false;
23            MythBgrCanvas.enabled = false;
24            PreHiCanvas.enabled = true;
25            PreHiBgrCanvas.enabled = true; ;
26            fromPart = 15;
27            toPart = 19;
28        }
29
30        for (int i = fromPart; i < toPart; i++)
31            if (PlayersPreferences.levels[i] > 0)
32            {
33                GameObject.Find("ChBtn" + i).GetComponent<Image>().color = new Color32(233, 153, 116, 255);
34            }
35    }
36 }
```

## public class MainQuestion

---

```
1 using UnityEngine;
2 using TMPro;
3
4 public class MainQuestion : MonoBehaviour
5 {
6     public static TMP_Text theQuestion;
7     public static TMP_Text[] Answer = new TMP_Text[3];
8
9     void Start()
10    {
11        theQuestion = GetComponent<TMP_Text>();
12        Answer[0] = GameObject.Find("Answer0").GetComponent<TMP_Text>();
13        Answer[1] = GameObject.Find("Answer1").GetComponent<TMP_Text>();
14        Answer[2] = GameObject.Find("Answer2").GetComponent<TMP_Text>();
15    }
16 }
```

## public class SoundFunction

---

```
1 using UnityEngine;
2 using UnityEngine.UI;
3 using UnityEngine.EventSystems;
4
5 public class SoundFunction : MonoBehaviour
6 {
7     public Sprite soundOn, soundOff;
8     public Image soundImage;
9     public EventSystem eventSystem;
10    public static bool soundIsOn = true;
11
12    private void Start()
13    {
14        if (!soundIsOn)
15            soundImage.sprite = soundOff;
16    }
17
18    public void DisableBtns()
19    {
20        eventSystem.SetSelectedGameObject(null);
21    }
22
23    public void ControlSound()
24    {
25        if (soundIsOn)
26        {
27            soundIsOn = false;
28            soundImage.sprite = soundOff;
29        }
30        else
31        {
32            soundIsOn = true;
33            soundImage.sprite = soundOn;
34        }
35        eventSystem.SetSelectedGameObject(null);
36    }
37
38    public static void PlayAudio(AudioSource Sound)
39    {
40        if (soundIsOn)
41            Sound.Play();
42    }
43
44    public static void StopAudio(AudioSource Sound)
45    {
46        Sound.Stop();
47    }
48 }
```

## public class PlayersPreferences

---

```

1 using UnityEngine;
2 using UnityEngine.UI;
3 using TMPro;
4 using System;
5
6 public class PlayersPreferences : MonoBehaviour
7 {
8     public static string PlayersName, dateString;
9     public static int PlayersID;
10    public static bool boy, Anonymus;
11    public static bool OnLine = true;
12    public static int[] levels = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };
13    private TextMeshProUGUI InputName;
14    private TextMeshProUGUI InputID;
15    private TextMeshProUGUI FInputName;
16    private TextMeshProUGUI FInputID;
17    private Toggle Boy;
18    private int next = 0;
19
20    void Start()
21    {
22        DateTime dt = DateTime.Now;
23        dateString = dt.ToString("yyyy-MM-dd HH:mm:ss");
24    }
25
26    public void StoreName()
27    {
28        string article = " η ";
29        if ( next == 1)
30        {
31            UnityEngine.SceneManagement.SceneManager.LoadScene(3);
32            return;
33        }
34
35        InputName = GameObject.Find("nameText").GetComponent<TextMeshProUGUI>
36        ();
37        InputID = GameObject.Find("idText").GetComponent<TextMeshProUGUI>();
38        InputName.text = "";
39        InputID.text = "";
40
41        FInputName = GameObject.Find
42        ("FinalnameText").GetComponent<TextMeshProUGUI>();
43        FInputID = GameObject.Find("FinalnameID").GetComponent<TextMeshProUGUI>
44        ();
45
46        Boy = GameObject.Find("Boy").GetComponent<Toggle>();
47
48        if (Boy.isOn)
49        {
50            boy = true;
51            article = " ο ";
52        }
53
54        if (InputName.text.Length > 1)
55        {

```

```

53     Anonymus = false;
54     PlayersName = InputName.text.Replace('_', '-');
55     PlayersName = PlayersName.Replace(' ', '.');
56
57     string tmpId = InputID.text.Substring(0, InputID.text.Length - 1);
58     if (InputID.text.Length > 1 && int.TryParse(tmpId, out int tmpInt))
59         PlayersID = Convert.ToInt32(tmpId);
60 }
61 else
62 {
63     Anonymus = true;
64     PlayersID = 0;
65
66     if (boy )
67         PlayersName = "Παίκτης";
68     else
69         PlayersName = "Παίκτηρια";
70 }
71
72 GameObject.Find("PlaceholderID").GetComponent<TextMeshProUGUI>().text =
73     "";
74
75 GameObject.Find("Placeholder").GetComponent<TextMeshProUGUI>().text =
76     "";
77
78 InputName.enabled = false;
79 InputID.enabled = false;
80 FInputName.text = "παίξει" + article + PlayersName;
81 FInputID.text = "με κωδικό: " + PlayersID;
82 FInputName.enabled = true;
83 FInputID.enabled = true;
84
85 next++;
86 GameObject.Find("SaveText").GetComponent<TextMeshProUGUI>().text =
87     "συνέχεια";
88 }
89 }

```

## public class Loader

---

```
1 using UnityEngine;
2
3 public class Loader : PlayersPreferences
4 {
5     public Material[] Skybox;
6     public static Material[] otherSkybox = new Material[19];
7     public TextAsset[] getFile = new TextAsset[19];
8     public static TextAsset[] csvFile = new TextAsset[19];
9
10    void Start()
11    {
12        for(int i=0; i<19; i++)
13        {
14            otherSkybox[i] = Skybox[i];
15            csvFile[i] = getFile[i];
16        }
17    }
18 }
```



## public class RWFile

---

```

1 using UnityEngine;
2 using System.IO;
3
4 public class RWFile : PlayersPreferences
5 {
6     public static string basicPath, path;
7     public static string scoreFile;
8     public static string scoreString = "", nameString = "";
9
10    void Start()
11    {
12        basicPath = Application.persistentDataPath + "/Scores";
13        if (!Directory.Exists(basicPath))
14        {
15            Directory.CreateDirectory(basicPath);
16
17            StreamWriter writer = new StreamWriter(basicPath + "/"
18                LocalScores.csv", false);
19            for (int i = 0; i < 10; i++)
20            {
21                writer.WriteLine("0");
22                writer.WriteLine("Player " + i);
23            }
24            writer.Close();
25        }
26        scoreFile = basicPath + "/LocalScores.csv";
27        path = basicPath + "/" + PlayersName + +PlayersID + ".csv";
28    }
29
30    public static bool isNewPlayer()
31    {
32        scoreFile = basicPath + "/LocalScores.csv";
33        path = basicPath + "/" + PlayersName + +PlayersID + ".csv";
34
35        return !File.Exists(path);
36    }
37
38    public static void WriteToFile()
39    {
40        StreamWriter writer = new StreamWriter(path, false);
41        for (int i = 0; i < 19; i++)
42        {
43            writer.WriteLine(levels[i]);
44        }
45        writer.Close();
46    }
47
48    public static void ReadFromFile()
49    {
50        int sum = 0;
51        StreamReader reader = new StreamReader(path);
52        for (int i = 0; i < 19; i++)
53        {
54            levels[i] = System.Convert.ToInt32( reader.ReadLine() );
55            sum += levels[i];
56        }
57    }
58
59 }

```

```

56     reader.Close();
57     scoreString = sum.ToString();
58 }
59
60 public static void WriteLocalScore(int newScore, string newName)
61 {
62     int[] scores = new int[10];
63     string[] names = { " ", " ", " ", " ", " ", " ", " ", " ", " ", " ", " ", " " };
64     scoreString = "";
65
66     StreamReader reader = new StreamReader(scoreFile);
67     for (int k = 0; k < 10; k++)
68     {
69         scores[k] = System.Convert.ToInt32(reader.ReadLine());
70         names[k] = reader.ReadLine();
71     }
72     reader.Close();
73
74     int i = 0;
75     int end = 10;
76     bool notInsert = true;
77     StreamWriter writer = new StreamWriter(scoreFile, false);
78     while (i < end && notInsert)
79     {
80         if (scores[i] < newScore)
81         {
82             writer.WriteLine(newScore);
83             writer.WriteLine(newName);
84             scoreString += newScore + "\n";
85             nameString += newName + "\n";
86             end--;
87             for (int j=i; j<end; j++ )
88             {
89                 writer.WriteLine(scores[j]);
90                 writer.WriteLine(names[j]);
91                 scoreString += scores[j] + "\n";
92                 nameString += names[j] + "\n";
93             }
94             notInsert = false;
95         }
96         else
97         {
98             writer.WriteLine(scores[i]);
99             writer.WriteLine(names[i]);
100            scoreString += scores[i] + "\n";
101            nameString += names[i] + "\n";
102        }
103        i++;
104    }
105    writer.Close();
106 }
107 }

```

## public class DBHandler

---

```

1 using System.Collections;
2 using System.Text.RegularExpressions;
3 using UnityEngine;
4 using UnityEngine.UI;
5 using UnityEngine.Networking;
6 using TMPro;
7
8 public class DBHandler : RWFile
9 {
10     private string getURL = "http://localhost/HighScoreGame/GetDate.php";
11     private string loginURL = "http://localhost/HighScoreGame/Login.php";
12     private string updatehighscoreURL = "http://localhost/HighScoreGame/
13         UpdateHighScores.php";
14     private string highscoreURL = "http://localhost/HighScoreGame/
15         LoadUsers.php";
16     private TextMeshProUGUI nameResultText;
17     private TextMeshProUGUI scoreResultText;
18     private TextMeshProUGUI copyText, yourScore;
19     private Image BoyImage, GirlImage;
20     private bool firstLogin = true;
21     private int GenScore;
22
23     void Start()
24     {
25         var item = GameObject.Find("HighScoresCanvas");
26
27         if (item != null)
28         {
29             nameResultText = GameObject.Find
30                 ("NameResults").GetComponent<TextMeshProUGUI>();
31             scoreResultText = GameObject.Find
32                 ("ScoreResults").GetComponent<TextMeshProUGUI>();
33             copyText = GameObject.Find
34                 ("CopyText").GetComponent<TextMeshProUGUI>();
35             yourScore = GameObject.Find
36                 ("yourScore").GetComponent<TextMeshProUGUI>();
37             BoyImage = GameObject.Find("BoyImage").GetComponent<Image>();
38             GirlImage = GameObject.Find("GirlImage").GetComponent<Image>();
39
40             for (int i = 0; i < 19; i++)
41                 GenScore += levels[i];
42
43             if (boy)
44                 GirlImage.enabled = false;
45             else
46                 BoyImage.enabled = false;
47
48             if (OnLine)
49             {
50                 if (!Anonymus)
51                     StartCoroutine(UpdateHighScores());
52                 else
53                     StartCoroutine(DisplayHighScores());
54             }
55         }
56     }
57 }

```

```

51         if (!Anonymus)
52         {
53             WriteToFile();
54             WriteLocalScore(GenScore, PlayersName);
55         }
56         else
57             WriteLocalScore(0, "");
58
59         nameResultText.text = nameString;
60         scoreResultText.text = scoreString;
61     }
62
63     copyText.text = "Εναρξη παιχνιδιού: " + dateString +
64     copyText.text;
65     yourScore.text = PlayersName + "\n\n" + GenScore;
66 }
67 else
68     nameResultText = GameObject.Find
69     ("FinalnameID").GetComponent<TextMeshProUGUI>();
70 }
71
72 public void TryToLogin()
73 {
74     if (firstLogin && !OnLine)
75     {
76         if (!Anonymus)
77         {
78             if (isNewPlayer())
79             {
80                 WriteToFile();
81             }
82             else
83             {
84                 ReadFromFile();
85                 nameResultText.text = "προηγούμενο σκορ: " + scoreString;
86             }
87         }
88         firstLogin = false;
89         return;
90     }
91
92     if (firstLogin && !Anonymus)
93     {
94         StartCoroutine(Login());
95         firstLogin = false;
96     }
97 }
98
99 IEnumerator Login()
100 {
101     WWWForm form = new WWWForm();
102     form.AddField("loginName", PlayersName);
103     form.AddField("loginId", PlayersID);
104     for (int i = 0; i < 19; i++)
105         form.AddField("L" + i, levels[i]);

```

```

105     using (UnityWebRequest www = UnityWebRequest.Post(loginURL, form))
106     {
107         yield return www.SendWebRequest();
108
109         if (www.result != UnityWebRequest.Result.Success)
110         {
111             nameResultText.text = www.error;
112         }
113         else
114         {
115             if (www.downloadHandler.text[0] == '_')
116             {
117                 PlayersID = System.Convert.ToInt32
118                     (www.downloadHandler.text.Substring(1,
119                     www.downloadHandler.text.Length - 1));
120                 nameResultText.text = "ο κωδικός σου: " + PlayersID;
121             }
122             else
123             {
124                 string dataText = www.downloadHandler.text;
125
126                 MatchCollection mc = Regex.Matches(dataText, @"_");
127                 if (mc.Count > 0)
128                 {
129                     int i;
130                     string[] splitData = Regex.Split(dataText, @"_");
131
132                     for (i = 0; i < mc.Count-1; i++)
133                     {
134                         levels[i] = System.Convert.ToInt32(splitData[i]);
135                     }
136
137                     nameResultText.text = "προηγούμενο σκορ: " + splitData
138                         [i];
139                 }
140             }
141         }
142     }
143
144     IEnumerator UpdateHighScores()
145     {
146         WWWForm form = new WWWForm();
147         form.AddField("loginName", PlayersName);
148         form.AddField("loginId", PlayersID);
149         for (int i = 0; i < 19; i++)
150             form.AddField("L" + i, levels[i]);
151
152         using (UnityWebRequest www = UnityWebRequest.Post(updatehighscoreURL,
153             form))
154         {
155             yield return www.SendWebRequest();
156
157             if (www.result != UnityWebRequest.Result.Success)
158             {

```



```
156         nameResultText.text = www.error;
157     }
158     else
159     {
160         StartCoroutine(DisplayHighScores());
161     }
162 }
163 }
164
165 IEnumerator DisplayHighScores()
166 {
167     nameResultText.text = "";
168     scoreResultText.text = "";
169
170     UnityWebRequest hs_get = UnityWebRequest.Get(highscoreURL);
171     yield return hs_get.SendWebRequest();
172     if (hs_get.error != null)
173         nameResultText.text = "There was an error getting the high score: ↗"
174         " + hs_get.error;
175     else
176     {
177         string dataText = hs_get.downloadHandler.text;
178         MatchCollection mc = Regex.Matches(dataText, @"_");
179         if (mc.Count > 0)
180         {
181             string[] splitData = Regex.Split(dataText, @"_");
182             for (int i = 0; i < mc.Count; i++)
183             {
184                 nameResultText.text += splitData[i++] + "\n";
185                 scoreResultText.text += splitData[i] + "\n";
186             }
187         }
188     }
189 }
```

## public class CSVReader

---

```

1 using System.Collections.Generic;
2 using UnityEngine;
3 using System;
4
5 public class CSVReader : DBHandler
6 {
7
8     [System.Serializable]
9     public class Question
10    {
11        public string Q { get; set; }
12        public string A1 { get; set; }
13        public string A2 { get; set; }
14        public string A3 { get; set; }
15    }
16
17    public static int chptr = 0;
18    public static int tableSize = 0;
19    public static List<Question> question;
20    public static List<Question> usedQuestion;
21
22    public static void ReadFromCSV( TextAsset questionFile )
23    {
24        string[] data = questionFile.text.Split(new string[] { ",", "\n" },
25        StringSplitOptions.None );
26        Question tmpQ;
27        tableSize = data.Length / 4;
28        question = new List<Question>{
29            new Question(){ Q = "Ερώτηση", A1="Πιθανή Απάντηση 1",
30            A2="Πιθανή Απάντηση 2", A3="Πιθανή Απάντηση 3" }
31        }
32        usedQuestion = new List<Question>{
33            new Question(){ Q = "Ερώτηση", A1="Πιθανή Απάντηση 1",
34            A2="Πιθανή Απάντηση 2", A3="Πιθανή Απάντηση 3" }
35        }
36
37        for (var i = 0; i < tableSize; i++)
38        {
39            tmpQ = new Question() { Q = data[4 * i], A1 = data[4 * i + 1], A2 =
40            data[4 * i + 2], A3 = data[4 * i + 3] };
41            question.Add(tmpQ);
42        }
43        tableSize += 1;
44    }
45
46    public static void LoadNewSetOfQuestions()
47    {
48        ReadFromCSV(Loader.csvFile[chptr]);
49    }
50 }

```



## public class SetQuestions

---

```

1 using System.Collections.Generic;
2 using UnityEngine;
3 using System;
4
5 public class SetQuestions : CSVReader
6 {
7     public static Question questToLoad;
8     public static int CorrectAnswer;
9
10    public static void ChooseQuestion()
11    {
12        int i;
13        var rnd = new System.Random(DateTime.Now.Millisecond);
14
15        if ( tableSize < 2)
16        {
17            tableSize = usedQuestion.Count;
18            question = usedQuestion;
19            usedQuestion = null;
20            usedQuestion = new List<Question>{
21                new Question(){ Q = "Ερώτηση", A1="Πιθανή Απάντηση 1",
22                    A2="Πιθανή Απάντηση 2", A3="Πιθανή Απάντηση 3" }
23            };
24            ChooseQuestion();
25        }
26        else
27        {
28            i = rnd.Next(1, tableSize--);
29            questToLoad = question[i];
30            question.RemoveAt(i);
31
32            MainQuestion.theQuestion.text = questToLoad.Q;
33            List<int> indexes = new List<int> { 0, 1, 2 };
34
35            i = rnd.Next(0, 3);
36            CorrectAnswer = i;
37            MainQuestion.Answer[i].text = questToLoad.A1;
38            indexes.Remove(i);
39
40            i = rnd.Next(0, 2);
41            MainQuestion.Answer[indexes[i]].text = questToLoad.A2;
42            indexes.Remove(indexes[i]);
43
44            MainQuestion.Answer[indexes[0]].text = questToLoad.A3;
45        }
46    }
47 }

```

## public class SceneLoader

---

```
1 using UnityEngine;
2 using UnityEngine.UI;
3 using UnityEngine.SceneManagement;
4 using TMPro;
5
6 public class SceneLoader : CSVReader
7 {
8     public static int QuestionsToAnswer = 12, prevCh = 4;
9     public static string LevelName;
10    public static Sprite LevelPicture;
11    public static bool playMythology = true;
12
13    public void TerminateGame()
14    {
15        Application.Quit();
16        UnityEditor.EditorApplication.isPlaying = false;
17    }
18
19    public void QuitGame()
20    {
21        SceneManager.LoadScene(7, LoadSceneMode.Single);
22    }
23
24    public void ImportYourName()
25    {
26        SceneManager.LoadScene(2, LoadSceneMode.Single);
27    }
28
29    public void ChooseYourAvatar()
30    {
31        prevCh = SceneManager.GetActiveScene().buildIndex;
32        SceneManager.LoadScene(3, LoadSceneMode.Single);
33    }
34
35    public void GoToHomePage()
36    {
37        SceneManager.LoadScene(4, LoadSceneMode.Single);
38    }
39
40    public void StartMythology()
41    {
42        playMythology = true;
43        SceneManager.LoadScene(5, LoadSceneMode.Single);
44    }
45
46    public void StartPrehistory()
47    {
48        playMythology = false;
49        SceneManager.LoadScene(5, LoadSceneMode.Single);
50    }
51
52    public void StartUnit1(int ch)
53    {
54        QuestionsToAnswer = 12;
55        LevelName = GameObject.Find("HintText" +
56            ch).GetComponent<TextMeshProUGUI>().text;
```

```
56     LevelPicture = GameObject.Find("Kef" + ch).GetComponent<Image>
      (.sprite;
57     chptr = ch;
58     LoadNewSetOfQuestions();
59     MainPlayer.QuestionsPlayed = 0;
60     MainPlayer.CorrectAnswers = 0;
61     SceneManager.LoadScene(6, LoadSceneMode.Single);
62 }
63
64 public void StartUnit2(int ch)
65 {
66     QuestionsToAnswer = 18;
67     LevelName = GameObject.Find("HintText" +
      ch).GetComponent<TextMeshProUGUI>().text;
68     LevelPicture = GameObject.Find("Kef" + ch).GetComponent<Image>
      (.sprite;
69     chptr = ch;
70     LoadNewSetOfQuestions();
71     MainPlayer.QuestionsPlayed = 0;
72     MainPlayer.CorrectAnswers = 0;
73     SceneManager.LoadScene(6, LoadSceneMode.Single);
74 }
75
76 public void ShowHelp()
77 {
78     Time.timeScale = 0.0f;
79     GameObject.Find("Footer").GetComponent<Canvas>().enabled = false;
80     GameObject.Find("HelpCanvas").GetComponent<Canvas>().enabled = true;
81 }
82
83 public void StartPrevius()
84 {
85     Time.timeScale = 1.0f;
86     SceneManager.LoadScene(5, LoadSceneMode.Single);
87 }
88 }
```

## public class MainPlayer

```

1 using UnityEngine.UI;
2 using UnityEngine;
3 using TMPro;
4
5 public class MainPlayer : SceneLoader
6 {
7     private Rigidbody rb;
8     private bool jumpWasPressed, enterWasPressed, mistakeWasMade,           ↗
9         answerWasFound, firstTimeOnStool = true, EnableGrassCollisions;
10    private float horizontalInput, RunLeft = 1, RunRight = 1;
11    public static int QuestionsPlayed = 0, CorrectAnswers = 0;
12    private int ThrowAway = 12000;
13    private falseStool clone;
14    private GameObject StoolToHide;
15    private Vector3 HiddenPosition = new (0, -1000, 0), CurrentPosition,     ↗
16        TurnLeft = new (0, -90, 0), TurnRight = new (0, 90, 0), TurnInFront = ↗
17        new (0, 180, 0);
18    private Quaternion CurrentRotation;
19    public static bool isGrounded = true;
20    public Animator PlayerAnimator, SceneAnimator, Instruction;
21    public falseStool StoolToClone;
22    public GameObject SetOfStools, AllScene, stool0, stool1, stool2, cherry;
23    public Canvas WinCanvas;
24    public AudioSource Correct, Wrong, Gongrats, Down, RunSteps;
25    public Animator[] Avatar = new Animator[5];
26    public GameObject[] playerObject = new GameObject[5];
27    public static Animator playerAnimator;
28
29    void Start()
30    {
31        RenderSettings.skybox = Loader.otherSkybox[chptr];
32        rb = GetComponent<Rigidbody>();
33
34        for (int i = 0; i < 5; i++)
35            playerObject[i].SetActive(false);
36
37        playerObject[CharacterSelection.index].SetActive(true);
38        PlayerAnimator = Avatar[CharacterSelection.index];
39        Time.timeScale = 1.0f;
40
41        for (int i = 1; i <= QuestionsToAnswer; i++)
42            GameObject.Find("Plaisio" + i).GetComponent<Image>().enabled =     ↗
43                true;
44
45        for (int i = 1; i <= CorrectAnswers; i++)
46            GameObject.Find("CAAnswer" + i).GetComponent<Image>().enabled =   ↗
47                true;
48
49        UpdateScore();
50
51        for (int i = 0; i <= 18; i++)
52            if (levels[i] != 0)
53                GameObject.Find("badge" + i).GetComponent<Image>().enabled = ↗
54                    true;
55
56        for(int i = 1; i <= CorrectAnswers; i++)

```

```
51         SceneAnimator.SetBool("Winning" + i.ToString("D2"), true);
52     }
53
54     private void SetHight()
55     {
56         for( int i=1; i<=CorrectAnswers; i++)
57             SceneAnimator.SetBool("Winning" + i.ToString("D2"), true);
58     }
59
60     void Update()
61     {
62         if (Input.GetKeyDown(KeyCode.Return))
63         {
64             enterWasPressed = true;
65         }
66
67         if (Input.GetKeyDown(KeyCode.Space))
68         {
69             jumpWasPressed = true;
70         }
71
72         if (Input.GetKeyDown(KeyCode.LeftArrow))
73         {
74             RunLeft++;
75         }
76
77         if (Input.GetKeyDown(KeyCode.RightArrow))
78         {
79             RunRight++;
80         }
81
82         horizontalInput = Input.GetAxis("Horizontal");
83     }
84
85
86     private void FixedUpdate()
87     {
88         if (transform.position.x < -100)
89         {
90             Vector3 returnPosition = transform.position;
91             returnPosition.x += 1;
92             transform.position = returnPosition;
93             SoundFunction.PlayAudio(Wrong);
94         }
95
96         if( transform.position.x > 165)
97         {
98             Vector3 returnPosition = transform.position;
99             returnPosition.x -= 1;
100             transform.position = returnPosition;
101             SoundFunction.PlayAudio(Wrong);
102         }
103
104         if (PlayerAnimator.GetBool("IsFalling"))
105         {
106             return;
```



```

107     }
108
109     if (horizontalInput == 0)
110     {
111         SoundFunction.StopAudio(RunSteps);
112         PlayerAnimator.SetBool("IsMoving", false);
113         RunLeft = RunRight = 1;
114     }
115
116     if (horizontalInput < 0 && RunRight > 0)
117     {
118         RunLeft = 1;
119         transform.localRotation = Quaternion.Euler(TurnLeft);
120         RunRight = 0;
121         PlayerAnimator.SetBool("IsMoving", true);
122         if (!PlayerAnimator.GetBool("Winning")) SoundFunction.PlayAudio (RunSteps);
123     }
124     if (horizontalInput > 0 && RunLeft > 0)
125     {
126         RunRight = 1;
127         transform.localRotation = Quaternion.Euler(TurnRight);
128         RunLeft = 0;
129         PlayerAnimator.SetBool("IsMoving", true);
130         if (!PlayerAnimator.GetBool("Winning")) SoundFunction.PlayAudio (RunSteps);
131     }
132
133     rb.velocity = new Vector3(horizontalInput * 500 * Time.deltaTime,
134                               rb.velocity.y, 0);
135
136     if (enterWasPressed && answerWasFound)
137     {
138         enterWasPressed = false;
139         ClimbUp();
140     }
141
142     if ( !isGrounded)
143     {
144         SoundFunction.StopAudio(RunSteps);
145         return;
146     }
147
148     if( jumpWasPressed )
149     {
150         rb.AddForce(600 * Time.deltaTime * Vector2.up,
151                    ForceMode.VelocityChange);
152         jumpWasPressed = false;
153     }
154 }
155
156 private void OnCollisionEnter(Collision collision)
157 {
158     isGrounded = true;

```

```

159
160     SoundFunction.PlayAudio(Down);
161     SoundFunction.PlayAudio(RunSteps);
162
163     if (collision.gameObject.name == "LeftStopQuad" ||
164         collision.gameObject.name == "RightStopQuad")
165     {
166         SoundFunction.PlayAudio(Wrong);
167         PlayerAnimator.SetBool("IsFalling", true);
168         RunLeft = RunRight = 1;
169         Vector3 tmp = transform.position;
170         tmp.y += 1.8f;
171         transform.position = tmp;
172     }
173
174     if (collision.gameObject.tag is "cherryHelp")
175     {
176         cherry.SetActive(false);
177         isGrounded = false;
178     }
179     else
180     {
181         cherry.SetActive(true);
182     }
183
184     if ( collision.gameObject.tag is "ignore" )
185     {
186         return;
187     }
188
189     if ( collision.gameObject.name == "mainGrass" )
190     {
191         if ( firstTimeOnStool )
192         {
193             if (mistakeWasMade)
194             {
195                 PlayerAnimator.SetBool("IsFalling", false);
196                 mistakeWasMade = false;
197                 ShowStool(StoolToHide);
198             }
199
200             EnableGrassCollisions = true;
201             PlayerAnimator.SetBool("IsMoving", true);
202             firstTimeOnStool = false;
203             SetQuestions.ChooseQuestion();
204             QuestionsPlayed++;
205         }
206         return;
207     }
208
209     if (!EnableGrassCollisions)
210     {
211         return;
212     }
213
214     if (collision.gameObject.name == "Grass" + SetQuestions.CorrectAnswer)

```



```

214     {
215         PlayWinningSenario(collision.gameObject.transform.position);
216     }
217     else
218     { if (collision.gameObject.name == "Grass0" ||
collision.gameObject.name == "Grass1" || collision.gameObject.name
== "Grass2")
219         {
220             Debug.Log(collision.gameObject.name);
221             if (collision.gameObject.name[5] - '0' !=
SetQuestions.CorrectAnswer)
222                 {
223                     StoolToHide = (collision.gameObject.name[5] - '0') switch
224                     {
225                         0 => stool0,
226                         1 => stool1,
227                         _ => stool2,
228                     };
229                     usedQuestion.Insert(1, SetQuestions.questToLoad);
230                     SoundFunction.StopAudio(RunSteps);
231                     DestroyStool(StoolToHide);
232                     UpdateScore();
233                     return;
234                 }
235             }
236     }
237 }
238
239
240
241 private void OnCollisionExit(Collision collision)
242 {
243     isGrounded = false;
244 }
245
246
247
248 private void PlayWinningSenario(Vector3 NewPosition)
249 {
250     answerWasFound = true;
251     mistakeWasMade = false;
252     firstTimeOnStool = true;
253     enterWasPressed = false;
254
255     PlayerAnimator.SetBool("IsMoving", false);
256     PlayerAnimator.SetBool("Winning", true);
257
258     transform.localRotation = Quaternion.Euler(TurnInFront);
259     RunLeft = RunRight = 1;
260
261     CorrectAnswers++;
262     Debug.Log("Correct! " + CorrectAnswers + " / " + QuestionsPlayed);
263     Win();
264
265     NewPosition.y += 1.8f;
266     transform.position = NewPosition;

```

```

267
268     rb.constraints = RigidbodyConstraints.FreezeAll;
269     EnableGrassCollisions = false;
270     TextMeshPro TextToPaint = GameObject.Find("Answer" +
271         SetQuestions.CorrectAnswer).GetComponent<TextMeshPro>();
272     TextToPaint.color = Color.green;
273     SceneAnimator.SetBool("Winning" + CorrectAnswers.ToString("D2"),
274         true);
275
276
277
278     private void PlayerWillFall()
279     {
280         SoundFunction.StopAudio(RunSteps);
281         SoundFunction.PlayAudio(Wrong);
282         PlayerAnimator.SetBool("IsFalling", true);
283         RunLeft = RunRight = 1;
284     }
285
286
287
288     private void ClimbUp()
289     {
290         for (int i=0; i<3; i++)
291         {
292             MainQuestion.Answer[i].color = new Color(1, 1, 1, 1);
293             MainQuestion.Answer[i].text = "";
294         }
295         MainQuestion.theQuestion.text = "";
296
297         PlayerAnimator.SetBool("Winning", false);
298         rb.constraints = RigidbodyConstraints.FreezeRotation |
299             RigidbodyConstraints.FreezePositionZ;
300         PlayerAnimator.SetBool("IsMoving", true);
301
302         Vector3 NewPosition = SetOfStools.transform.position; ;
303         NewPosition.y += 10 * CorrectAnswers;
304
305         GameObject NextHelpingStools = Instantiate(SetOfStools, NewPosition,
306             SetOfStools.transform.rotation);
307
308         firstTimeOnStool = true;
309         answerWasFound = false;
310     }
311
312     public void DestroyStool(GameObject stool)
313     {
314         mistakeWasMade = true;
315         firstTimeOnStool = true;
316
317         CurrentPosition = stool.transform.position;
318         CurrentRotation = stool.transform.rotation;

```

```

319         stool.transform.position = HiddenPosition;
320
321         clone = Instantiate(StoolToClone, CurrentPosition, CurrentRotation);
322         clone.transform.Rotate(0, 0, 1 * 10 * Time.deltaTime);
323         clone.GetComponent<Rigidbody>().AddForce(ThrowAway * Time.deltaTime *
324             Vector3.forward);
325         ThrowAway += 10000;
326     PlayerWillFall();
327 }
328
329
330
331 public void ShowStool(GameObject stool)
332 {
333     stool.transform.position = CurrentPosition;
334 }
335
336
337
338 void Win()
339 {
340     RunSteps.Pause();
341
342     if (CorrectAnswers > QuestionsToAnswer)
343         return;
344
345     if ( CorrectAnswers < 3 )
346         Instruction.Play("EnterInstruction", 0, 0.0f);
347
348     GameObject.Find("CAnswer" + CorrectAnswers).GetComponent<Image>
349         ().enabled = true;
350
351     UpdateScore();
352
353     if (CorrectAnswers == QuestionsToAnswer)
354     {
355         for (int i = 0; i < 3; i++)
356         {
357             MainQuestion.Answer[i].color = new Color(1, 1, 1, 1);
358             MainQuestion.Answer[i].text = "";
359         }
360
361         SoundFunction.PlayAudio(Gongrats);
362         RunSteps.volume = 0;
363
364         WinCanvas.enabled = true;
365         GameObject.Find("Footer").GetComponent<Canvas>().enabled = false;
366
367         Time.timeScale = 0.0f;
368         GameObject.Find("Level").GetComponent<TextMeshProUGUI>().text =
369             LevelName;
370         GameObject.Find("Badge").GetComponent<Image>().sprite =
371             LevelPicture;
372
373         levels[chptr] = ((int)((CorrectAnswers * 1.0 / QuestionsPlayed) *

```

```

    10000 ) + 1);
371     levels[chptr] = ((int)(levels[chptr] * (QuestionsToAnswer /
    12.0)); Debug.Log("Teliko "+levels[chptr]));
372     GameObject.Find("GetTheBadge").GetComponent<TextMeshProUGUI>
    (.text += levels[chptr].ToString());
373
374     GameObject.Find("CAAnswer" + CorrectAnswers).GetComponent<Image>
    (.enabled = true;
375     GameObject.Find("badge" + chptr).GetComponent<Image>().enabled =
    true;
376 }
377 else
378     SoundFunction.PlayAudio(Correct);
379 }
380
381
382
383 private void UpdateScore()
384 {
385     GameObject.Find("CorrectToAll").GetComponent<TextMeshProUGUI>().text =
    CorrectAnswers + " / " + QuestionsPlayed;
386 }
387 }
```

## Παράρτημα II – Ανάλυση 5βαθμης Κλίμακας

Η ανάλυση των αποτελεσμάτων του ερωτηματολογίου αξιολόγησης, όπως ακριβώς δόθηκαν οι απαντήσεις σε 5βαθμη κλίμακα, είναι αυτά που ακολουθούν. Αναλυτικά ανά ερώτηση:

Ερ:	Μέση Τιμή	Διακύμανση	Τυπική Απόκλιση	Αρνητικό	Θετικό	Επιμέρους Διάκριση
π1	↑ 1,29	0,7	0,8	παρελκυστικό	υποστηρικτικό	πραγματική ποιότητα
π2	↑ 0,91	0,9	0,9	περίπλοκο	εύκολο	πραγματική ποιότητα
π3	↑ 1,42	0,6	0,8	ανεπαρκές	επαρκές	πραγματική ποιότητα
π4	↑ 1,14	1,0	1,0	μπερδεμένο	σαφές	πραγματική ποιότητα
η1	↑ 1,11	0,8	0,9	βαρετό	συναρπαστικό	ηδονική ποιότητα
η2	↑ 1,31	0,6	0,8	αδιάφορο	ενδιαφέρον	ηδονική ποιότητα
η3	↑ 1,20	0,7	0,8	συμβατικό	εφευρετικό	ηδονική ποιότητα
η4	↑ 1,10	0,8	0,9	συνηθισμένο	πρωτοπόρο	ηδονική ποιότητα

και συγκεντρωτικά:

Ομαδοποιημένοι Μέσοι Όροι	
πραγματική ποιότητα	↑ 1,188
ηδονική ποιότητα	↑ 1,182
για όλες τις ερωτήσεις	↑ 1,185

με αντίστοιχα διαστήματα εμπιστοσύνης, ανά ερώτηση

Διαστήματα Εμπιστοσύνης (p=0.05) ανά Ερώτηση						
Ερ:	Μέση Τιμή	Τυπική Απόκλιση	N	Εμπιστοσύνη	Διάστημα Εμπιστοσύνης	
π1	1,292	0,832	96	0,167	1,125	1,458
π2	0,906	0,941	96	0,188	0,718	1,095
π3	1,417	0,763	96	0,153	1,264	1,569
π4	1,135	0,980	96	0,196	0,939	1,331
η1	1,115	0,881	96	0,176	0,938	1,291
η2	1,313	0,786	96	0,157	1,155	1,470
η3	1,198	0,841	96	0,168	1,030	1,366
η4	1,104	0,900	96	0,180	0,924	1,284

και συγκεντρωτικά ανά ομάδα:

Διαστήματα Εμπιστοσύνης ( $p=0.05$ ) ανά Ομάδα						
Ομάδα	Μέση Τιμή	Τυπική Απόκλιση	N	Εμπιστοσύνη	Διάστημα Εμπιστοσύνης	
πραγματική ποιότητα	1,188	0,684	96	0,137	1,051	1,324
ηδονική ποιότητα	1,182	0,675	96	0,135	1,047	1,317
για όλες τις ερωτήσεις	1,185	0,569	96	0,114	1,071	1,299

Τα 95% διαστήματα εμπιστοσύνης ως προς την εκτίμηση της μέσης τιμής, έχουν εύρη, από 0,392 (το μεγαλύτερο) στην π4, μέχρι 0,228 στο σύνολο των ερωτήσεων.

Η εκτίμηση των μέσων τιμών στην 5βαθμη κλίμακα, από -2 μέχρι 2, υποδεικνύει τα ίδια αποτελέσματα με την 7βαθμη και είναι πολύ ακριβέστερη.

*Άρχισε κάνοντας αυτό που είναι αναγκαίο, μετά αυτό που είναι εφικτό και ξαφνικά  
θα βρεθείς να κάνεις το αδύνατο.*

Άγιος Φραγκίσκος της Ασίζης, 1181-1226, Ιταλός Άγιος