



ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

«Σχεδιασμός διαδραστικού παιχνιδιού για εκπαίδευση
στα ψηφιακά κυκλώματα»

Της φοιτήτριας
Καούρα Χαρίκλεια
Αρ. Μητρώου: **516043**

Επιβλέπων
Μπάμνιος Γεώργιος
Διδάσκων

Σεπτέμβριος 2023

Τίτλος Δ.Ε. Σχεδιασμός δια δραστικού παιχνιδιού για εκπαίδευση στα ψηφιακά κυκλώματα

Κωδικός Δ.Ε. 22219

Όνοματεπώνυμο φοιτητή: Χαρίκλεια Καούρα

Όνοματεπώνυμο εισηγητή: Μπάμνιος Γεώργιος

Ημερομηνία ανάληψης Δ.Ε.: Μάρτιος του 2022

Ημερομηνία περάτωσης Δ.Ε.: Σεπτέμβριος 2023

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

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

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

Πρόλογος

Το γεγονός ότι όλο και περισσότεροι νέοι ενδιαφέρονται για πιο διασκεδαστικούς και μη συμβατικούς τρόπους μάθησης είναι ο λόγος που δημιουργήθηκε η συγκεκριμένη διπλωματική εργασία. Το εκπαιδευτικό κομμάτι της εργασίας είναι κάτι που θα ήθελα να συνεχίσω και να εξειδικευτώ στη πορεία της καριέρας μου, οπότε τι πιο ενδιαφέρον από την δημιουργία του Quiz Game πάνω στις λογικές πύλες και στην ύλη των ψηφιακών ηλεκτρονικών.

Περίληψη

Σκοπός της παρούσας διπλωματικής εργασίας με τίτλο «Σχεδιασμός διαδραστικού παιχνιδιού για εκπαίδευση στα ψηφιακά κυκλώματα» είναι η ανάλυση, ο σχεδιασμός και η υλοποίηση ενός παιχνιδιού για εκπαιδευτική χρήση στο μάθημα Σχεδίαση Ψηφιακών Κυκλωμάτων. Η εργασία εστιάζεται στα σημαντικά αποτελέσματα που προέκυψαν έπειτα από έρευνα του πεδίου των ηλεκτρονικών παιχνιδιών και ειδικότερα αυτών που χρησιμοποιούνται για εκπαιδευτικούς σκοπούς. Αναλύθηκε η διαδικασία που ακολουθείται για την δημιουργία πολυμεσικών εφαρμογών. Με τη χρήση της παιχνιδομηχανής Unity και του εργαλείου δημιουργίας διεπαφών UI Toolkit, στόχος ήταν η δημιουργία ενός κουίζ όπου σκοπός του παίκτη/φοιτητή θα είναι να επιλέγει το σωστό εικονίδιο σύμφωνα με την ερώτηση που του γίνεται και να το τοποθετήσει στο κουτάκι όπου του έχει ζητηθεί. Κάθε φορά που ο χρήστης δίνει μια σωστή απάντηση, καταφέρνει να συλλέξει ένα σύνολο πόντων και ακούγεται ένας χαρακτηριστικός ήχος, ενώ αντίστοιχα σε μια λάθος απάντηση ενημερώνεται και ακούγεται ο αντίστοιχος ήχος. Με τη χρήση ο παίκτης εξοικειώνεται με τις λογικές πύλες που αποτελούν βασικό πυλώνα στο μάθημα. Παράλληλα, στο μενού δίνονται κάποιες επιπλέον επιλογές όπως την παρακολούθηση βίντεο που αφορούν την ύλη του μαθήματος και ένα ακόμα κουίζ με ερωτήσεις πολλαπλής επιλογής που καλύπτουν ένα καίριο μέρος της ύλης του μαθήματος.

Λέξεις Κλειδιά: Εκπαιδευτικό παιχνίδι, Unity, UI Toolkit, QUIZ

«Designing an interactive game for digital electronics.»

(στην αγγλική γλώσσα)

«Charikleia Kaoura»

Abstract

The purpose of this thesis entitled "Designing an interactive game for training in digital circuits" is the analysis, design and implementation of a game for educational use in the course Designing Digital Circuits. The work focuses on the important results obtained after research in the field of electronic games and in particular those used for educational purposes. The process followed for the creation of multimedia applications was analyzed. Using the Unity game engine and the UI Toolkit, we wanted to create a Quiz where the goal of the player/student is to select the correct icon according to the question asked and drag it into the box where it is asked. Every time the user gives a correct answer, he manages to collect a set of points and a characteristic sound is heard, while correspondingly in the case of a wrong answer, the corresponding sound is heard. With use the player becomes familiar with the logic gates that are a mainstay of the course. At the same time, in the menu options, the user finds some other options such as watching videos related to the course material and another Quiz with multiple choice questions that cover a key part of the course material.

Ευχαριστίες

Θέλω να ευχαριστήσω τους τους κοντινούς μου ανθρώπους, την οικογένεια που με στήριξαν σε ψυχολογικό επίπεδο την περίοδο της συγγραφής της παρούσας διπλωματικής εργασίας, τους καθηγητές που μου έδειξαν εμπιστοσύνη τον κ. Κωτσάκη Ρ. και τον κ. Μπάμνιο Γ. και μου με ενθάρρυναν να συνεχίσω και να φτάσω μέχρι το τέλος.

Περιεχόμενα

Πρόλογος.....	4
Περίληψη.....	5
Abstract	6
Ευχαριστίες	7
Περιεχόμενα	8
Κατάλογος Εικόνων	10
Συντομογραφίες.....	12
1. Εισαγωγή.....	13
1.1 Αντικείμενο και Οργάνωση Εργασίας	13
2. Το Ψηφιακό Παιχνίδι	15
2.1 Ορισμός και βασικά χαρακτηριστικά ηλεκτρονικών παιχνιδιών	15
2.2 Κατηγορίες ηλεκτρονικών παιχνιδιών	16
2.3 Δεξιότητες για την ανάπτυξη ψηφιακών παιχνιδιών και επαγγελματικά.....	20
3. Ηλεκτρονικά παιχνίδια στην μάθηση.....	21
3.1 Σοβαρά παιχνίδια και μάθηση μέσω (ψηφιακού) παιχνιδιού- Game-based Learning	22
3.2 Πλεονεκτήματα χρήσης παιχνιδιών στο εκπαιδευτικό περιβάλλον	24
3.3 Είδη Κουίζ.....	26
4. Στάδια Ανάπτυξης Πολυμεσικής Εφαρμογής	30
4.1 Ανάλυση.....	31
4.1.1 Λογικές Πύλες και Εκπαίδευση	31
4.2.1 Αρχές Χρηστικότητας	34
4.2.2 Σχεδίαση - Πρότυπα Χαμηλής και Υψηλής Πιστότητας	34
5. Περιβάλλον δημιουργίας διάδρασης	37
5.1 Εισαγωγή στο Unity	37
5.2 Πλεονεκτήματα	37
5.3 Απαιτήσεις και δημιουργία Project	38

5.4 Ανάπτυξη σε UI Toolkit.....	39
5.5 Ροή Εργασίας μέσω UI Builder.....	40
6. Ανάπτυξη εφαρμογής.....	44
6.1 Μενού.....	44
6.2.1 Σκηνή Quiz και UI Builder	45
6.2.2 Δημιουργία Ερωτήσεων	48
6.2.3 Extra Panel	49
6.2.4 Quiz πολλαπλής επιλογής	50
6.3 Ολοκλήρωση της Εφαρμογής.....	52
7. Παρατηρήσεις και Βελτιώσεις	54
8. Επίλογος.....	56
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	57
ΙΣΤΟΣΕΛΙΔΕΣ.....	59
ΠΑΡΑΡΤΗΜΑ Α	60

Κατάλογος Εικόνων

Εικόνα 2.1: Παραδείγματα παιχνιδιών διάφορων κατηγοριών

Εικόνα 2.2: Παραδείγματα λειτουργικών συστημάτων ηλεκτρονικών παιχνιδιών

Εικόνα 3.1: Κατηγοριοποίηση ειδών μάθησης και μάθησης μέσω παιχνιδιών, Becker, K. (2021)

Εικόνα 3.2: Παράδειγμα Quiz πολλαπλής επιλογής

Εικόνα 3.3: Παράδειγμα Quiz με εικόνα

Εικόνα 3.4: Παράδειγμα Quiz με ταίριασμα

Εικόνα 3.5: Παράδειγμα Quiz Σωστού/Λάθους ή ΝΑΙ/ΟΧΙ

Εικόνα 3.6: Παράδειγμα Ερευνητικού Quiz

Εικόνα 4.1: Πίνακες Αληθείας Λογικών Πυλών

Εικόνα 4.2: Όψη παιχνιδιού “Circuit Scrable”

Εικόνα 4.3: Μενού παιχνιδιού 7 Gates

Εικόνα 4.4: Πρωτότυπη Οθόνη Χαμηλής Πιστότητας- Διεπαφή Χρήστη (UI)

Εικόνα 4.5: Πρωτότυπη Οθόνη Υψηλής Πιστότητας- Quiz στις λογικές πύλες

Εικόνα 5.1: Επιλογή διαστάσεων

Εικόνα 5.2: Επιλογή UI Builder

Εικόνα 5.3: Παράδειγμα Visual Tree

Εικόνα 5.4: Όψη του περιβάλλοντος εργασίας της μηχανής Unity

*Εικόνα 5.5: Βήματα σχεδίασης σε UI Builder

Εικόνα 6.1: Οπτική της σκηνής του MenuScene

Εικόνα 6.2: UI Builder #Label view

Εικόνα 6.3: UI Builder #QuestionCounter view

Εικόνα 6.4: UI Builder #Score view

Εικόνα 6.5: UI Builder #AnswerBoard view

Εικόνα 6.6: Δημιουργία Questions

Εικόνα 6.7: Εμφάνιση Question στο Inspector

Εικόνα 6.8: Οπτική της σκηνής SampleScene

Εικόνα 6.9: Οπτική του Extra Panel

Εικόνα 6.10: Βίντεο μαθήματος στο Google Drive

Εικόνα 6.11: Εμφάνιση Quiz Manager στον Inspector

Εικόνα 6.12: Ερωτήσεις πολλαπλής επιλογής με απάντηση κείμενο και εικόνα

Εικόνα 6.13: Φάκελος αποθήκευσης

Εικόνα 6.14: Εξαγωγή εφαρμογής από το Unity

Εικόνα 7.1: Διόρθωση Εμφάνισης

Συντομογραφίες

Δ.Ε.	Διπλωματική Εργασία
ΔΠΠΑΕ	Διεθνές Πανεπιστήμιο Ελλάδος
Π.Ε.	Πτυχιακή Εργασία
Σ.Ψ.Κ	Σχεδίαση Ψηφιακών Κυκλωμάτων
VR	Virtual Reality (Εικονικής Πραγματικότητας)
AR	Augmented reality (Επαυξημένη Πραγματικότητα)

1. Εισαγωγή

Όπως είναι γνωστό στα μέσα του 20ού αιώνα άρχισε να γίνεται η χρήση της εξ αποστάσεως εκπαίδευσης αλλά η άνθηση της έγινε κυρίως μέσα στην περίοδο της πανδημίας ως την ανάγκη μια εναλλακτικής μορφής συνέχισης της εκπαίδευσης. Η ενσωμάτωση παιχνιδιών μέσω διαδικτύου ή μέσω εφαρμογών στο κινητό έχει ωφελήσει δασκάλους και μαθητές κάνοντας την διδασκαλία και την μάθηση πιο αποτελεσματική. Μέσα από τα παιχνίδια για την εκπαίδευση δίνεται η δυνατότητα στους μαθητές να αυτό-αξιολογούνται με βάση το αποτέλεσμα ή το σκορ που έχουν πετύχει μετά την ολοκλήρωση της προσπάθειας. Η εκμάθηση όταν μπορεί να συνδυαστεί με την διασκέδαση μπορεί να φέρει πολύ θετική επίδραση στις μαθησιακές δεξιότητες. Με την δημιουργία αυτού του εκπαιδευτικού παιχνιδιού-Quiz θέλουμε να δώσουμε έμφαση στην εξοικείωση με τις βασικές γνώσεις και αρχές στα ψηφιακά ηλεκτρονικά. Τα ψηφιακά ηλεκτρονικά είναι ιδιαίτερα χρήσιμα σε πολλούς τομείς ιδιαίτερα σε τομείς της τεχνολογίας και της ηλεκτρονικής καθώς είναι τα βασικά στοιχεία που χρησιμοποιούνται στον σχεδιασμό και την κατασκευή διαφόρων ηλεκτρονικών κυκλωμάτων. Η γνώση των λογικών πυλών είναι βασική για τον σχεδιασμό και την κατανόηση της λειτουργίας των επεξεργαστών και άλλων συσκευών καθώς επίσης στον χώρο της βιομηχανίας παιχνιδιών, η γνώση των λογικών πυλών είναι βασική για την ανάπτυξη διαφόρων συστημάτων και λογισμικού.

1.1 Αντικείμενο και Οργάνωση Εργασίας

Η αρχική ιδέα ήταν η δημιουργία ενός εκπαιδευτικού λογισμικού διαφόρων ερωτήσεων, που προήλθε από την ανάγκη για περισσότερη διαδραστική εξοικείωση σε διάφορα θεωρητικά κομμάτια της ύλης του μαθήματος Σχεδίαση Ψηφιακών Συστημάτων (ΣΨΣ). Στόχος της εφαρμογής είναι να αφυπνίσει το ενδιαφέρον για τις ψηφιακές πύλες που είναι απαραίτητο για την επιτυχία τους στο μάθημα αλλά και τις βασικές γνώσεις που χρειάζονται για να μπορούν να έχουν ομαλή εξέλιξη και στα επόμενα μαθήματα. Επίσης το παιχνίδι είναι χωρισμένο σε δύο μέρη, στο πρώτο μέρος που είναι καθαρά οι ερωτήσεις πολλαπλών επιλογών και στο δεύτερο μέρος που είναι η επιλογή του σωστού εικονιδίου-πύλη που κρατώντας πατημένο το ποντίκι το μεταφέρει στο κουτάκι απαντήσεων. Παράλληλα, υπάρχει η επιλογή για παρακολούθηση βίντεο-διαλέξεων που έχουν ηχογραφηθεί από τον εισηγητή του μαθήματος και είναι στην διάθεση των φοιτητών για παραπάνω μελέτη. Η παρούσα ΔΕ εκτελέστηκε με εκπαιδευτική στόχευση αλλά και με βαρύτητα στην σωστή βιβλιογραφική επισκόπηση. Αρχικά γίνεται αναφορά στη θεωρία λογισμικών ψυχαγωγίας, δηλαδή στον ορισμό τους και στις κατηγορίες τους όπως επίσης αναφέρεται ο ρόλος τους στην εκπαίδευση. Έπειτα, γίνεται αναφορά στη θεωρία σχεδιασμού και μεθοδολογίας αυτών των λογισμικών

Στο κεφάλαιο 2 επιχειρήθηκε να δοθεί ένας γενικός ορισμός του όρου ηλεκτρονικού παιχνιδιού και ποια είναι τα βασικά χαρακτηριστικά που έχει ένα παιχνίδι. Επίσης, η ανάλυση και αποσαφήνιση όλων των

εννοιών που σχετίζονται με το ψηφιακό παιχνίδι και την κατηγοριοποίηση τους όπως και τις κατάλληλες δεξιότητες ανάπτυξης τους. Τα ηλεκτρονικά παιχνίδια μπορούν να ταξινομηθούν σε πολλές κατηγορίες, υποκατηγορίες και είδη, ανάλογα το περιεχόμενό τους, σύμφωνα για ποιο λειτουργικό σύστημα έχουν δημιουργηθεί και εάν είναι online ποιους στόχους εκπληρώνουν.

Στο κεφάλαιο 3, γίνεται εκτενής αναφορά ως προς τους βασικούς όρους της παιχνιδοποίησης (Gamification), των σοβαρών παιχνιδιών (serious games) και της μάθησης μέσω παιχνιδιού (Game-based Learning) όπως και στα πλεονεκτήματα χρήσης παιχνιδιών στο εκπαιδευτικό περιβάλλον και τα είδη των κουίζ.

Το κεφάλαιο 4 αναφέρεται στα στάδια ανάπτυξης μια πολυμεσικής εφαρμογής. Πιο συγκεκριμένα το πρώτο υποκεφάλαιο ξεκινάει με το στάδιο της ανάλυσης και με παραδείγματα εφαρμογών, λογικών πυλών στην εκπαίδευση. Στο επόμενο υποκεφάλαιο, γίνεται αναφορά στις Αρχές Χρηστικότητας και στην Σχεδίαση Προτύπων Χαμηλής και Υψηλής Πιστότητας.

Στο κεφάλαιο 5 γίνεται αναφορά στο περιβάλλον που επιλέξαμε για την δημιουργία διάδρασης το Unity. Ειδικότερα, γίνεται περιγραφή στα πλεονεκτήματα που έχει σε σύγκριση με άλλες εφαρμογές και περισσότερες λεπτομέρειες σε ένα από τα βασικά εργαλεία που χρησιμοποιήσαμε για την δημιουργία ενός μέρους του παιχνιδιού μας, το UI Toolkit και τη σχεδίαση μέσα από το UI Builder.

Το επόμενο στάδιο που ακολουθεί μετά τον σχεδιασμό είναι το στάδιο της υλοποίησης, σε αυτό περιλαμβάνεται η δημιουργία των σκηνών, η ενοποίηση μεταξύ τους, η ανάπτυξη κώδικα καθώς και η βελτιστοποίηση της αποδοτικότητας της εφαρμογής. Στο κεφάλαιο 6 δηλαδή γίνεται η εξήγηση της ανάπτυξης του παιχνιδιού για κάθε σκηνή και κάθε κουίζ που δημιουργήθηκε.

Το τελευταίο κεφάλαιο, πριν τον επίλογο, είναι το έβδομο και αφορά τις παρατηρήσεις που έγιναν από αυτούς που συμμετείχαν και δοκίμασαν να παίξουν το παιχνίδι και τις βελτιώσεις που ακολούθησαν αργότερα όπως και κάποιες προτάσεις για να γίνει ακόμα καλύτερο το παιχνίδι .

2. Το Ψηφιακό Παιχνίδι

Το παρόν κεφάλαιο περιλαμβάνει το θεωρητικό υπόβαθρο πάνω στο γενικό όρο του ψηφιακού παιχνιδιού. Αρχικά δίνεται ο ορισμός και τα βασικά χαρακτηριστικά, έπειτα η κατηγοριοποίηση με βάση το περιεχόμενο του παιχνιδιού αν είναι για παράδειγμα περιπέτειας ή προσομοίωσης. Τα διαδικτυακά παιχνίδια επίσης, τα λεγόμενα online χωρίζονται και αυτά σε κατηγορίες. Τέλος, στο τελευταίο υποκεφάλαιο έχουμε τις δεξιότητες που χρειάζεται για να δημιουργήσει κάποιος ένα ηλεκτρονικό παιχνίδι.

2.1 Ορισμός και βασικά χαρακτηριστικά ηλεκτρονικών παιχνιδιών

Τα πρώτα παιχνίδια στον υπολογιστή εμφανίστηκαν στη δεκαετία του εξήντα και του εβδομήντα του 20ού αιώνα, ποιος μπορεί να φανταστεί ότι αυτά τα διαδραστικά κλασικά παιχνίδια όπως το σκάκι ή το πινγκ πονγκ θα γίνουν μια από τις μεγαλύτερες βιομηχανίες στον κόσμο; Και ότι αυτή η βιομηχανία θα παράγει περισσότερα ακόμη και από την κινηματογραφική βιομηχανία; [1] Όσον αφορά τον ορισμό του ηλεκτρονικού παιχνιδιού, δεν μπορεί να ειπωθεί πως υπάρχει ένας ξεκάθαρος ορισμός, ούτε μια και μοναδική θεωρία που να μπορεί να εξηγήσει το ρόλο του ηλεκτρονικού παιχνιδιού στη μάθηση και στην ανάπτυξη του παιδιού. Ένα παιχνίδι αποτελεί έναν αγώνα, είτε σωματικό είτε πνευματικό, που υπαγορεύεται από συγκεκριμένους κανόνες, με στόχο την ψυχαγωγία ή την ανταμοιβή των συμμετεχόντων. Αυτός ο ορισμός, σύμφωνα με τον Zyda, ισχύει εξίσου και για τα ηλεκτρονικά παιχνίδια. Τα ηλεκτρονικά παιχνίδια αποτελούν πνευματικούς αγώνες που διεξάγονται με τη βοήθεια υπολογιστή για αυτό και ορίζονται ως ψηφιακά καθώς το μέσο με το οποίο μπορεί να έχει πρόσβαση ο παίκτης είναι ψηφιακό, τηρώντας συγκεκριμένους κανόνες, με στόχο την διασκέδαση, την ψυχαγωγία ή την ενδεχόμενη κερδοσκοπία. Ο όρος "ηλεκτρονικά παιχνίδια" είναι ο συχνότερα χρησιμοποιούμενος όρος όσο αφορά τα παιχνίδια. Συνήθως, υπάρχει μια οθόνη που απεικονίζει το παιχνίδι και αποτελεί την πηγή διασκέδασης του παίκτη. Επιπλέον, ανάλογα με το παιχνίδι, υπάρχουν διάφορες συσκευές εισόδου όπως πληκτρολόγιο, τηλεχειριστήριο ή οθόνη αφής. Το παιχνίδι μπορεί να απαιτεί διαφορετική συσκευή εισόδου ανάλογα με την πλατφόρμα για την οποία δημιουργήθηκε. Διάφοροι επιστήμονες, προσπαθώντας να ερμηνεύσουν τη θέση του ηλεκτρονικού παιχνιδιού στη ζωή μας, ανέπτυξαν διάφορες θεωρίες. [2]

Λόγο της ασάφειας του όρου “παιχνίδι”, σύμφωνα με τον Prensky και Becker, K. θα θεωρήσουμε ότι το ηλεκτρονικό παιχνίδι έχει έξι βασικά χαρακτηριστικά:

- **Αλληλεπίδραση:** Η επίτευξη του στόχου από τους παίκτες εξαρτάται αποκλειστικά από τις επιλογές που θα κάνουν και τις ενέργειες που θα αναλάβουν κατά τη διάρκεια του παιχνιδιού. Το παιχνίδι δεν έχει νόημα χωρίς την ενεργή συμμετοχή του παίκτη.

• **Κανόνες:** Περιλαμβάνει τις συνθήκες και τις πληροφορίες που επηρεάζουν την εξέλιξη του παιχνιδιού. Αυτές περιλαμβάνουν τις δομές δεδομένων που αποθηκεύουν τις πληροφορίες σχετικά με τα αντικείμενα και τους νόμους της φυσικής που διέπουν τον κόσμο του παιχνιδιού. Αυτές οι συνθήκες ποικίλουν ανάλογα με το σενάριο που αναπτύσσεται και τη διαχείριση των καταστάσεων που παρουσιάζονται στον παίκτη. Επιπλέον, στο τμήμα της λογικής συμπεριλαμβάνεται η διαχείριση των διαδικασιών προγραμματισμού και ο μεταφραστής των εντολών του παίκτη που εκτελούνται από τη διασύνδεση σε χαμηλότερο επίπεδο του προγραμματικού κώδικα. Γενικά πρέπει να υπάρχει ένας σαφές σύνολο κανόνων, προκειμένου να διευκολυνθεί η αλληλεπίδραση του χρήστη με το παιχνίδι.

• **Έχει ένα ή και περισσότερους σκοπούς και στόχους:** Αποτελεί και ένα στοιχείο πρόκλησης να επιτευχθεί ο στόχος ή να σημειωθεί μία μεγάλη βαθμολογία.

• **Αποτελέσματα και ανατροφοδότηση:** Το παιχνίδι θα πρέπει να ανταμείβει τις σωστές αποφάσεις και να αναδεικνύει τις λανθασμένες. Με αυτόν τον τρόπο, οι παίκτες θα είναι σε θέση να αναγνωρίζουν τις επιτυχημένες και ανεπιτυχείς ενέργειες, επιδιώκοντας να βελτιώσουν τις δεξιότητές τους.

• **Έχει ένα διακριτό τέλος:** Να αναγνωρίζει ο παίχτης πότε έφτασε στο τέλος.

Στο κομμάτι της εκπαίδευσης σε συνδυασμό με τα ψηφιακά παιχνίδια παρουσιάζονται και κάποια πρόσθετα χαρακτηριστικά όπως:

• **Μαθησιακοί στόχοι:** Τα ψηφιακά παιχνίδια σχεδιάζονται για κάποιο συγκεκριμένο σκοπό και αποβλέπουν στην επίτευξη ενός συγκεκριμένου στόχου.

• **Προϋπάρχουσα γνώση:** Ένα εκπαιδευτικό παιχνίδι συνήθως προϋποθέτει κάποια γνώση πάνω σε ένα τομέα, όπως για παράδειγμα τα μαθηματικά

Σήμερα τα ψηφιακά παιχνίδια γνωρίζουν μια αλματώδη ανάπτυξη. Όπως είναι φανερό αρκετοί ερευνητές εστιάζουν, όχι μόνο στις αλληλεπιδράσεις ανθρώπων και υπολογιστών, αλλά και στον τρόπο με τον οποίο διαμορφώνονται οι διάφορες δραστηριότητες με τα παιχνίδια και το αντίτυπο που έχουν αυτές στη γνώση. Τα περισσότερα από αυτά τα ψυχαγωγικά λογισμικά μπορούν να θεωρηθούν ως κάποιου είδους προσομοιώσεις, είτε κάποιου πραγματικού κόσμου (αθλητικά παιχνίδια, παιχνίδια ανάπτυξης πολιτισμών), είτε κάποιου φανταστικού κόσμου (παιχνίδια περιπέτειας, παιχνίδια φαντασίας), είτε ενός παραδοσιακού παιχνιδιού (τρίλιζα, σταυρόλεξα). Παρακάτω θα δούμε πιο αναλυτικά τις κατηγορίες.

2.2 Κατηγορίες ηλεκτρονικών παιχνιδιών

Σύμφωνα με τον Mark Overmars στην ερώτηση τι είναι ψηφιακό παιχνίδι περιέχονται αρκετά γενικές απαντήσεις και διαφορετικές κατηγορίες. Σύμφωνα με τον Mark οι πιο σημαντικές κατηγορίες παιχνιδιών που υπάρχουν σύμφωνα με το περιεχόμενό τους, είναι η εξής: [3]



Εικόνα 2.1: Παραδείγματα παιχνιδιών διάφορων κατηγοριών

- Παιχνίδια δράσης (Arcade games): Παιχνίδια στα οποία η ταχύτητα αντίδρασης είναι η πιο σημαντική πτυχή του παιχνιδιού. Παιχνίδια σαν αυτά είναι το Pacman όπου συχνά τα συναντάμε σε 2D γραφικά και σχετικά πιο εύκολα στην δημιουργία τους.
- Παιχνίδια γρίφων (puzzle games): Είναι παιχνίδια με γρίφους, στα οποία η γρήγορη και έξυπνη σκέψη, αποτελεί τη πιο σημαντική πτυχή του παιχνιδιού. Επίσης θεωρούνται εύκολα στην κατασκευή τους σχετικά και είναι κατά κύριο λόγο 2D.
- Παιχνίδια ρόλων (Role Playing Games –RPG): Είναι τα παιχνίδια στα οποία υπάρχει κάποιος χαρακτήρας που μετακινείται μέσα σε έναν εικονικό κόσμο. Σ’ αυτή την περίπτωση οι παίκτες αναλαμβάνουν το ρόλο φανταστικών χαρακτήρων και μέσω της συνεργασίας, δημιουργούν ή παρακολουθούν ιστορίες. Οι παίκτες είναι αυτοί που καθορίζουν τις ενέργειες των χαρακτήρων τους, εν μέρει βασιζόμενοι στο σχεδιασμό του χαρακτήρα, και οι ενέργειες πετυχαίνουν ή αποτυγχάνουν σύμφωνα με το σύστημα κανόνων και οδηγιών. Παραδείγματα τέτοιων παιχνιδιών είναι το Assassin’s Creed και το Diablo. Αυτά τα παιχνίδια είναι συνήθως 3D.
- Παιχνίδια στρατηγικής (Strategy games): Τα παιχνίδια αυτά συνήθως πραγματοποιούνται σε πραγματικό χρόνο. Σ’ αυτά τα παιχνίδια ο παίκτης ελέγχει έμμεσα το χαρακτήρα, αλλά ο ίδιος δεν προβλέπει τις στρατηγικές. Αυτού του είδους τα παιχνίδια είναι πιο δύσκολο να υλοποιηθούν, αφού απαιτούν διαφορετικά αντικείμενα, χαρακτήρες και κτίρια σε κάθε φάση, έχουν δηλαδή πολλαπλά σενάρια. Χρησιμοποιούν την ισομετρική θέα, μια τεχνική πιο διάσημη στα πρώτα παιχνίδια της δεκαετίας του ‘90, παράδειγμα αποτελούν το Age of Empires, Caesar κ.α

- Παιχνίδια διαχείρισης (Management games): Σκοπός των παιχνιδιών αυτών είναι η δημιουργία μιας αυτοκρατορίας. Σ' αυτά τα παιχνίδια, ο παίκτης διαχειρίζεται μία πόλη ή μία εταιρία. Αυτά τα παιχνίδια είναι δύσκολο να υλοποιηθούν, γιατί πρέπει να προσομοιώνεται σχεδόν ένα ολόκληρος κόσμος, όπως λόγω χάρη η συμπεριφορά των επισκεπτών σε ένα θεαματικό πάρκο και όχι μόνο. Γνωστά παιχνίδια που ανήκουν σε αυτή την κατηγορία είναι το SimCity, Theme Park, Sims και πολλά άλλα.
- Παιχνίδια περιπέτειας (Adventure games): Είναι τα παιχνίδια στα οποία ο χρήστης κάνει εικονική περιήγηση σ' έναν άγνωστο κόσμο προσπαθώντας να λύσει διάφορους γρίφους. Τα συγκεκριμένα παιχνίδια θέλουν ιδιαίτερη τεχνική στην δημιουργία της "ιστορίας" του παιχνιδιού και να συνδυάζεται με την αντίστοιχη σχεδίαση. Κάποια από αυτά είναι το The Walking Dead game, Spycraft: The Great Game, Grim Fandango κ.α
- Παιχνίδια σκοπευτών (First-person, Third-person shooters): Είναι πάντα σε 3D περιβάλλοντα για να δίνουν την αίσθηση της παρουσίας εκεί και δίνουν περισσότερη έμφαση στην ταχύτητα αντίδρασης και στην ανακάλυψη του εικονικού κόσμου. Κάποια πολύ γνωστά παραδείγματα είναι το Doom και το Tomb Raider.
- Αθλητικά παιχνίδια (Sport games): Τα αθλητικά παιχνίδια παρουσιάζουν εικονικά τα παραδοσιακά αθλήματα, όπως το ποδόσφαιρο. Τα περισσότερα παιχνίδια αυτού του τύπου είναι πολύ γνωστά, γιατί ο χρήστης μπορεί να ελέγξει έναν ή περισσότερους χαρακτήρες σε μια στιγμή και είναι τόσο ρεαλιστικά που πολλές φορές νομίζει κανείς ότι τα παρακολουθείς σε πραγματικό χρόνο.
- Παιχνίδια αγώνων (Racing games): Αυτά τα παιχνίδια αποτελούν μία ειδική κατηγορία των αθλητικών παιχνιδιών. Είναι πάρα πολλά αυτού του είδους των παιχνιδιών για αυτό γίνεται η ένταξη τους σε μία ειδική κατηγορία. Εδώ ανήκουν όλα τα παιχνίδια αγώνων ταχύτητας, όπως για παράδειγμα Formula-1, τα οποία είναι και πολύ ανταγωνιστικά.
- Παιχνίδια προσομοίωσης (Simulators): Τα παιχνίδια αυτού του είδους στοχεύουν στην προσομοίωση ενός μηχανισμού, όσο το δυνατόν πιο ρεαλιστικά. Παραδείγματα τέτοιων παιχνιδιών, είναι οι προσομοιωτές πτήσης.

Προφανώς υπάρχουν και άλλοι τύποι παιχνιδιών και υποκατηγορίες, επίσης να αναφέρουμε ότι πολλά παιχνίδια μπορεί να ανήκουν σε πάνω από μια κατηγορία ταυτόχρονα, αυτό πολλές φορές γίνεται για να είναι πιο ελκυστικά από τους χρήστες. Ο δημιουργός του παιχνιδιού βέβαια είναι βοηθητικό να μπορεί να το ένταξή σε μια από τις παραπάνω κατηγορίες.

Η αμεσότητα που παρείχαν (και εξακολουθούν να παρέχουν) οι προσωπικοί υπολογιστές και οι κονσόλες παιχνιδιών κατά την πρώτη τους εμφάνιση στην αγορά, ήταν αυτή που οδήγησε τους σχεδιαστές παιχνιδιών στο να συνειδητοποιήσουν ότι το ψηφιακό παιχνίδι αποτελεί το μέλλον της βιομηχανίας των παιχνιδιών. Τα κυριότερα λειτουργικά συστήματα που μπορούν να λειτουργήσουν τα ηλεκτρονικά παιχνίδια είναι:

- τα παιχνίδια χειρός (Gameboy, PSP, Nintendo)
- τις οικιακές βιντεοκονσόλες (PlayStation, Xbox)
- τις παιχνιδομηχανές Arcade (Mortal Kombat, Pong)
- τον προσωπικό ηλεκτρονικό υπολογιστή (Laptop, Desktop)



Εικόνα 2.2: Παραδείγματα λειτουργικών συστημάτων ηλεκτρονικών παιχνιδιών

Τα τελευταία χρόνια αξιόλογη απήχηση έχουν τα online παιχνίδια τα οποία μπορούν να χωριστούν στις παρακάτω κατηγορίες. [4]

Παιχνίδια μέσω κάποιου browser: είναι τα παιχνίδια που έχουν ένα απλό χαρακτήρα και παρέχονται συνήθως χωρίς αντίτιμο από ιστοσελίδες που παρέχουν και διαφημίσεις. Για παράδειγμα τέτοια παιχνίδια είναι το Farmville, JellyCrush, dodGames.

Παιχνίδια διαφημιστικού τύπου: είναι παιχνίδια που σχεδιάστηκαν για την προώθηση μιας υπηρεσίας , ενός προϊόντος η μιας ιδεολογίας. Τέτοια παιχνίδια είναι το CoolSpot (7Up), το Zool (Chupa Chups), και το Americas Army (USArmy).

Παιχνίδια δικτύου: είναι τα παιχνίδια που τα εγκαθιστά ο χρήστης σε ένα ηλεκτρονικό υπολογιστή αλλά για να τα παίξει θα χρειαστεί να υπάρχει κάποια σύνδεση στο διαδίκτυο. Παραδείγματα τέτοιων παιχνιδιών είναι το Fortnite (EpicGames), η το League Of Legends (RiotGames).

Παιχνίδια τύπου MMORPG: είναι τα παιχνίδια τα οποία να μεν είναι παιχνίδια δικτύου, αλλά σε κάθε ένα παιχνίδι από αυτά είναι εγγεγραμμένος ένας τεράστιος αριθμός παικτών από όλο τον κόσμο και το παιχνίδι συνεχίζει να προοδεύει και ας είναι εκτός σύνδεσης ο παίκτης. Τέτοια παιχνίδια είναι το World

of Warcraft (Blizzard Entertainment), το Guardians of Divinity (37Games) και το Guild Wars 2 (Arena Net).

2.3 Δεξιότητες για την ανάπτυξη ψηφιακών παιχνιδιών και επαγγελματικά

Απαιτούνται πολλές δεξιότητες και εργαλεία για την ανάπτυξη παιχνιδιών, για αυτό το λόγο ένα επαγγελματικό στούντιο δημιουργίας παιχνιδιών υπάρχουν εργαζόμενοι διαφόρων κλάδων. Αυτοί μπορεί να είναι προγραμματιστές που καθένας είναι υπεύθυνος για ένα συγκεκριμένο μέρος του παιχνιδιού, σχεδιαστές, καλλιτέχνες για το τμήμα γραφικών, video editing, τεχνικοί ήχου και ακόμα και τμήμα διοίκησης. Όταν ένας φοιτητής θέλει να εργαστεί στη βιομηχανία παιχνιδιών πρέπει να σκεφτεί για το ποιο τομέα θα ταιριάζει καλά, θεωρητικά ένας φοιτητής στο κομμάτι της πληροφορικής είναι καλός στο κομμάτι του προγραμματισμού και ένας φοιτητής με γνώσεις Photoshop στο τμήμα ψηφιακή τέχνη και κινούμενα σχέδια. Γενικότερα, υπάρχουν αρκετοί λόγοι για τους οποίους αξίζει κάποιος να ασχοληθεί επαγγελματικά με τον τομέα, όπως:

- Πολυμορφικό επάγγελμα, που πολύ δύσκολα θα το βαρεθείς με τα χρόνια.
- Έχει μεγάλη ζήτηση ως επάγγελμα (στο εξωτερικό κυρίως).
- Έχει πολύ καλές αποδοχές.
- Συνδυάζει άριστα επιστημονικό και καλλιτεχνικό κομμάτι

3. Ηλεκτρονικά παιχνίδια στην μάθηση

Οι νέοι μαθητές ζουν σε έναν ψηφιακό κόσμο, όπου η διαδραστικότητα και η συμμετοχή αποτελούν κομμάτι της καθημερινότητάς τους. Η χρήση ηλεκτρονικών παιχνιδιών στη διαδικασία της μάθησης προσφέρει μια εναλλακτική προσέγγιση στην παραδοσιακή διδασκαλία, επιτρέποντας την αλληλεπίδραση και την ανακάλυψη της γνώσης μέσα από παιχνιδιώδεις διαδικασίες.[5] Παρόλα αυτά, το σχολείο δεν καλύπτει πάντα αυτήν την ανάγκη, καθώς η αλληλεπίδραση που προσφέρει σε πολλές περιπτώσεις φαίνεται παρωχημένη και ανεπαρκής. Αυτό οδηγεί στο γεγονός ότι οι νέοι αναζητούν την ικανοποίηση αυτών των αναγκών μέσω ηλεκτρονικών παιχνιδιών, αν και αυτά δεν προορίζονται για εκπαιδευτικούς σκοπούς.[6] Η εκπαιδευτική κοινότητα στην απόπειρα να αξιοποιήσει αυτήν την ανάγκη, τα τελευταία χρόνια έχει επικεντρωθεί στην μελέτη και την αξιοποίηση των ηλεκτρονικών παιχνιδιών προς όφελος της μαθησιακής διαδικασίας.[7] Ένα σημαντικό μέρος των σύγχρονων ερευνών στον τομέα της μάθησης με βάση το παιχνίδι επικεντρώνεται σε πολύ σύγχρονες μεθόδους όπως στη διερεύνηση των δυνατοτήτων της εικονικής πραγματικότητας (VR) και των τεχνολογιών κινητής τηλεφωνίας. Η χρήση VR και AR επιτρέπει τα εικονικά στοιχεία να είναι επάλληλα με τα πραγματικά αντικείμενα που μπορούν να λειτουργήσουν και μέσω ενημέρωσης και μάθησης. Η μείωση του κόστους αυτών των τεχνολογιών καθιστά εφικτή την χρήση τους ακόμα και σε οικονομικά προγράμματα κατάρτισης σε πολλά εκπαιδευτικά ιδρύματα. Ωστόσο, εξακολουθούν να αποτελούν πρόβλημα για πολλές χώρες, καθώς απαιτούν φορητές συσκευές με την υψηλή υπολογιστική ισχύ και κατάλληλους χώρους τοποθέτησης τους. Πολλοί ερευνητές έχουν προτείνει διάφορες μορφές παιχνιδιοποίησης και ότι μπορεί να στραφεί προς πολλές κατευθύνσεις, ειδικότερα μπορεί να έχει τη μορφή ανταμοιβής για την επίτευξη ενός συγκεκριμένου επιπέδου και επιτυχία σε δοκιμασία υπό όρους για την απόκτηση γνώσεων και μπορεί να χρησιμοποιηθεί κατά τη διάρκεια όλης της διαδικασίας της μάθησης, μετατρέποντας την σε παιχνίδι. Στη δεύτερη περίπτωση, οι μαθητές αναπτύσσουν ένα συγκεκριμένο κίνητρο, να παίξουν καλύτερα και να κατακτήσουν ένα πιο υψηλό επίπεδο, απλά σε αυτή τη περίπτωση απαιτείται η χρήση πιο πολύπλοκου λογισμικού και τη συμμετοχή ενός δασκάλου ή διαχειριστή (game host). Η παραγωγή περιεχομένου για παιχνίδια απαιτεί αρκετό χρόνο, χρήματα και προσπάθεια, όπως επίσης να καλείς τους ειδικούς να το δημιουργούν και να το αναβαθμίζουν. Αυτό επηρεάζει την ταχύτητα με την οποία αναβαθμίζονται τα δεδομένα ενός παιχνιδιού, όμως η διαρκή αλλαγή δεδομένων κρίνεται απαραίτητη για την διατήρηση επαρκούς ελέγχου των γνώσεων των μαθητών, καθώς ανταλλάζουν εύκολα τα επιτεύγματα και τις γνώσεις για να περνάνε τα τεστ. Τέτοια αλληλεπίδραση μέσω κοινωνικών δικτύων κάνει την επαλήθευση των τεστ μη ακριβή εάν το περιεχόμενο δεν ανανεώνεται συχνά για κάθε νέα ομάδα μαθητών.[8]

Ερευνητές έχουν επίσης ασχοληθεί με την εκπαιδευτική αξία των ηλεκτρονικών παιχνιδιών. Οι Mitchel και Savill Smith θεωρούν τα πολύπλοκα παιχνίδια ως εργαλεία που ενθαρρύνουν την αλλαγή στάσεων, την ανάπτυξη κριτικής σκέψης και των δεξιοτήτων απαραίτητων για την επίλυση

προβλημάτων και τη λήψη αποφάσεων. Η συνεργατικότητα είναι ένα ακόμα βασικό στοιχείο που αναπτύσσεται μέσω των "πολλαπλών παικτών παιχνιδιών", όπου οι συμμετέχοντες μαθαίνουν μέσα από τη συμμετοχή σε μια κοινότητα που έχει κοινούς στόχους.[9] Τα παιχνίδια πολλαπλών χρηστών ενισχύουν τη συνεργασία και την αίσθηση της κοινότητας, καθώς οι μαθητές συνεργάζονται για την επίλυση προκλήσεων και την επίτευξη στόχων. Η συνεργασία αυτή ενισχύει την αλληλεπίδραση και την ανταλλαγή απόψεων μεταξύ των μαθητών, προωθώντας την κοινωνική μάθηση. [10]

Ένα ακόμα σημαντικό στοιχείο των ηλεκτρονικών παιχνιδιών είναι η ικανότητα να προσελκύουν το ενδιαφέρον μέσα από ένα περιβάλλον δελεαστικό και εντυπωσιακό που μπορεί να δημιουργηθεί από υψηλής ποιότητας γραφικά. Αυτό σχετίζεται και με την έννοια του κινήτρου, το οποίο αποτελεί τη δύναμη πίσω από τη βουλητική συμπεριφορά. Τα παιχνίδια δημιουργούν κίνητρα μέσω της διασκέδασης, η οποία αποτελεί ουσιώδη μέρος της μαθησιακής διαδικασίας. Το κίνητρο αποτελεί κρίσιμο παράγοντα για την αποτελεσματική μάθηση. Η διαδικασία της μάθησης μέσω παιχνιδιών προσφέρει άμεση ανατροφοδότηση και αμοιβές, διεγείροντας το κίνητρο των μαθητών να επιτύχουν στους εκπαιδευτικούς στόχους. [11]

3.1 Σοβαρά παιχνίδια και μάθηση μέσω (ψηφιακού) παιχνιδιού- Game-based Learning

Το συγκεκριμένο υποκεφάλαιο αναφέρεται γενικότερα στα εκπαιδευτικά ηλεκτρονικά παιχνίδια και εστιάζει πιο συγκεκριμένα στους τρόπους χρήσης του. Θα γίνει αναφορά πώς συνδυάζεται η εκπαίδευση με το παιχνίδι και ποια είναι η διαφορά μεταξύ «serious games», παιχνιδοποίησης και μάθησης μέσω παιχνιδιού.

Ο όρος «serious games» δηλαδή σοβαρά παιχνίδια πρωτοεισήχθη στη βιβλιογραφία από τον Abt αλλά έγινε δημοφιλής από τον Sawyer. [12] Σύμφωνα με τον αρχικό ορισμό του Abt, τα παιχνίδια που ικανοποιούν τον όρο «έχουν ρητό και προσεκτικά μελετημένο εκπαιδευτικό σκοπό και δεν προορίζονται να χρησιμοποιηθούν κυρίως για διασκέδαση. Αυτό δεν σημαίνει ότι δεν είναι, ή δεν θα έπρεπε να είναι, διασκεδαστικά». Είναι σχεδιασμένα για χρήση στον υπολογιστή, για την διδασκαλία διαφόρων εννοιών, όρων και μεθόδων.[13] Τα σοβαρά παιχνίδια τείνουν να είναι στατικά και πιο συγκεκριμενοποιημένα σε ένα μάθημα ή πρόγραμμα ως εκ τούτου δεν μπορούν να μεταφερθούν σε ένα ευρύ φάσμα επιστημονικών κλάδων λόγω του υψηλού κόστους εργασίας και της τεχνικής ανάπτυξης. Έχουν γίνει προσπάθειες να μειωθεί η σχεδιαστική πολυπλοκότητα των “serious games”, παραδείγματα αποτελούν βασισμένα σε σενάρια και ένα άλλο είναι μέσω της χρήσης ενός meta-μοντέλου.[14]

Η βασική διαφοροποίηση της μάθησης μέσω παιχνιδιού (Game-based Learning) με την έννοια του σοβαρού παιχνιδιού έγκειται στο ότι η μάθηση μέσω παιχνιδιού αναφέρεται επί της ουσίας σε μία εκπαιδευτική μεθοδολογία ή για παράδειγμα για την διδασκαλία μια συγκεκριμένης ενότητας. Ενώ ένα σοβαρό παιχνίδι είναι ένα λογισμικό το οποίο μπορεί να αξιοποιηθεί ως μαθησιακό αντικείμενο στο πλαίσιο μιας τέτοιας μεθοδολογίας. Υπάρχουν διαφορετικά είδη ηλεκτρονικών παιχνιδιών που μπορούν

να χρησιμοποιηθούν και να εμπλακούν με διαφορετικούς τρόπους στην εκπαιδευτική πρακτική. Ο ευκολότερος τρόπος είναι να χρησιμοποιούμε το περιεχόμενο του παιχνιδιού σε μορφή τεστ πολλαπλής επιλογής όπου το παιχνίδι είναι επιβράβευση για την μάθηση.

Ο όρος παιχνιδιοποίηση (Gamification) άρχισε να χρησιμοποιείται το μισό του 2010 αναφέρεται “στην πρακτική αξιοποίησης μηχανισμών που τυπικά απαντώνται σε παιχνίδια σε μη-παιγνιώδη περιβάλλοντα και δραστηριότητες”. [15] Ένα τέτοιο περιβάλλον είναι και η εκπαίδευση, εκεί όπου η παιχνιδιοποίηση έχει βρει μεγάλη απήχηση τα τελευταία χρόνια, κυρίως σε ψηφιακά περιβάλλοντα μάθησης. Σε πρακτικό επίπεδο, περιλαμβάνει την υιοθέτηση στοιχείων, όπως επίπεδα, σύστημα πόντων, πίνακες κατάταξης ή/και παράσημα, τα οποία συνδέονται με την ολοκλήρωση δραστηριοτήτων ή/και διδακτικών ενοτήτων, την επίτευξη συγκεκριμένων μαθησιακών στόχων, κτλ. Συνήθως, αυτά τα παιχνίδια δεν έχουν σχεδιαστεί και αναπτυχθεί για εκπαιδευτικούς σκοπούς, αλλά μπορούν να προσαρμοστούν από τους εκπαιδευτικούς αναλόγως τις ανάγκες της διδασκαλίας που χρειάζονται. Παρακάτω βλέπουμε ένα σχεδιάγραμμα που μας δίνει αναλυτικά και διευκρινιστικά όλες τις κατηγορίες και τα είδη παιχνιδιών.

Distinctions between Types of Teaching & Learning using Games						
© K.Becker 2021	Game	Serious Game	Game for Learning (G4L)	Game-Based Learning (GBL)	Game-Based Pedagogy (GBP)	Gamification
Basic Definition	This term includes BOTH Serious Games AND Games for Learning	A game <i>designed</i> for purposes other than or in addition to pure entertainment.	A game <i>designed</i> specifically with some learning goals in mind.	The process and practice of <i>learning</i> using games. [From the <i>learner's</i> point of view]	The process and practice of <i>teaching</i> using games. [From the <i>teacher's</i> point of view]	The use of game elements in a non-game context.
Purpose	Can be for any purpose.	Change in behaviour, attitude, health, understanding, knowledge.	Normally connected with some educational goals.	Not a game - this is an approach to learning.	Not a game - this is an approach to teaching.	Often used to drive motivation, but can also be used to make something more playful and game like.
Primary Driver (why used)	Can be either play or rewards (or both).	To get the message of the game.	To learn something.	To improve learning. To increase learning effectiveness. <i>*Note GBP & GPL are related, but not the same.</i>	To improve teaching practice & effectiveness. <i>*Note GBP & GPL are related. They are like two sides of a single coin.</i>	Depending on how it's implemented, it can tap into extrinsic or intrinsic rewards (or both)
Key Question	Is it fun?	Is the message being received?	Is it effective?	Am I learning what I am supposed / need to be learning?	Is it effective?	Business: Does it improve profits? Education: Is it effective?
Focus	Player Experience (<i>how</i>)	Content / Message (<i>what</i>)	Content / Message (<i>what</i>)	Learning Objectives (<i>what & how</i>)	Learning Objectives (<i>what & how</i>)	User Experience (<i>how</i>)
Budgets	Next to nothing to 100's of millions.	Next to nothing to 100's of thousands.	Next to nothing to 100's of thousands.	Usually part of an institutional budget. Largely irrelevant to the user.	Usually part of an institutional budget. Largely irrelevant to the user.	Next to nothing to 10's of thousands.
Business Model	User Pays	Producer Pays	Varies	Institution Pays	Institution Pays	Producer Pays
Concept Catalyst	Core Amusement.	Message.	Performance or Knowledge Gap	Game is the lesson or is used as a part of the lesson.	Game is the lesson or is used as a part of the lesson.	When used in learning it usually impacts HOW things are taught and administered rather than WHAT is taught.
Fidelity	Self-consistent, otherwise irrelevant.	Faithfulness to message essential	Faithfulness to message essential	Faithfulness to message essential	Faithfulness to message essential	Not Applicable. If a narrative exists, it need have nothing to do with what's being gamified.

Εικόνα 3.1: Κατηγοριοποίηση ειδών μάθησης και μάθησης μέσω παιχνιδιών, Becker, K. (2021)

Πέρα από την παιχνιδιοποίηση και την μάθηση μέσω παιχνιδιού υπάρχει μια γενικότερη στροφή σε άλλες σύγχρονες τεχνικές διδασκαλίας, για παράδειγμα η συν-μάθηση (co-learning) και ηλεκτρονική

μάθηση (e-learning). Η μάθηση με βάση το παιχνίδι έχει δείξει αντικειμενικά καλύτερα αποτελέσματα από τις κλασικές δραστηριότητες της τάξης σε αρκετά σύντομη περίοδο μελέτης. Ωστόσο υπάρχουν νέες μέθοδοι που εισάγονται συνεχώς και αυξάνουν την αποτελεσματικότητα της τάξης. Πολλοί ερευνητές επισημαίνουν την αρνητική επίδραση των βιντεοπαιχνιδιών στις γνωστικές ικανότητες των μαθητών, αλλά δεν υπάρχει κάποια έρευνα δημοσιευμένη πάνω σε αυτό το θέμα. Έτσι, η μακροπρόθεσμη αξιοπιστία των αποτελεσμάτων και η επιρροή είναι καθοριστικής σημασίας για την ευρεία χρήση της ηλεκτρονικής παιχνοποίησης στη μαθησιακή διαδικασία.

3.2 Πλεονεκτήματα χρήσης παιχνιδιών στο εκπαιδευτικό περιβάλλον

Σύμφωνα με τον Norman, οι βασικές προϋποθέσεις που καθιστούν το μαθησιακό περιβάλλον αποτελεσματικό είναι οι εξής [16]:

1. Να παρέχει αλληλεπίδραση και εποικοδομητική ανατροφοδότηση. Η δυνατότητα ανατροφοδότησης ανά πάσα στιγμή για την πρόοδο του φοιτητή και η δυνατότητα πειραματισμού χωρίς να κυριαρχεί ο φόβος του λάθους είναι βασικό χαρακτηριστικό πλεονέκτημα.
2. Να έχει συγκεκριμένους στόχους και εμπεριστατωμένες διαδικασίες. Σε ένα δομημένο σχέδιο μαθήματος είναι πιο σπάνιο να χάσει ο μαθητής το αρχικό του ενδιαφέρον.
3. Να παροτρύνει. Τα ψηφιακά παιχνίδια αποτελούν δίοδο εμφύχωσης των παιδιών που πιθανόν μειονεκτούν σε ενδιαφέρον ή έχουν διάσπαση προσοχής και ενισχύουν πιο πολύ την αυτοπεποίθηση τους.
4. Να παρέχει ένα συνεχόμενο αίσθημα προκλήσεων, οι οποίες δεν είναι ούτε πολύ εύκολες ώστε να προκαλούν ανία, ούτε πολύ δύσκολες ώστε να απελπίζουν τον μαθητευόμενο. Πολλά παιχνίδια, όπως γρίφοι και διαδραστικά παιχνίδια, βελτιώνουν την εφευρετικότητα και την κριτική σκέψη των παιδιών. Ενώ πολύπλοκα παιχνίδια, προκαλούν στα παιδιά νοητική επεξεργασία και ανάπτυξη στρατηγικών ικανοτήτων.
5. Να δίνει την αίσθηση άμεσης δέσμευσης, δημιουργεί την εντύπωση πως η συσχέτιση του ατόμου με την εκάστοτε εργασία που πραγματοποιείται είναι άμεση.
6. Να προσφέρει τα κατάλληλα εργαλεία που ταιριάζουν απόλυτα στη διαδικασία και τον χρήστη και δεν αποσπών από το στόχο. Πολλές φορές τα ψηφιακά παιχνίδια προσφέρουν εξατομικευμένη προσέγγιση της μάθησης, σύμφωνα με τις ανάγκες που ο μαθητής θέλει να καλύψει και δεν διαθέτουν χρονικό όριο περάτωσης τους.
7. Ν αποφεύγουν περισπασμούς και διαταραχές που παρεμβαίνουν και καταστρέφουν την υποκειμενική εμπειρία.

Συνεπώς, η εκμάθηση μέσω της ψηφιακής διασκέδασης αποδεικνύεται αποτελεσματική, διότι υιοθετεί τη μορφή ενός παιχνιδιού. Αυτό το γεγονός την καθιστά πιο ελκυστική, ενώ για την επίτευξή της απαιτείται μια διαδραστική διαδικασία, η οποία προσαρμόζεται ανάλογα με τους εκπαιδευτικούς στόχους. Συνεπώς, τα ψηφιακά παιχνίδια συνδράμουν στην ανάπτυξη και τη βελτίωση των γνωστικών ικανοτήτων των παιδιών, ειδικότερα [17]:

- Στην επίτευξη συγκέντρωσης και προσοχής.
- Στην αύξηση της ικανότητας παρατήρησης.
- Στον συντονισμό των οπτικών και κινητικών δεξιοτήτων.
- Στην ανάπτυξη της δεξιοτεχνίας.
- Στην επέκταση των γενικών γνώσεων.
- Στην ανάπτυξη της ικανότητας επίλυσης προβλημάτων.

Κατά συνέπεια, καταλήγουμε ότι το παιδί που παίζει εκπαιδευτικά παιχνίδια είναι πολλαπλά επωφελούμενο διότι επιδρά ευχάριστα στην ψυχολογία του, το να ολοκληρώνει τις εργασίες του και παράλληλα αναπτύσσει δεξιότητες που διαφορετικά θα ήταν πιο δύσκολο να αναπτύξει. Κάποιες από τις δεξιότητες είναι οι παρακάτω:

- Η εμπάθυνση στην κριτική σκέψη αποτελεί σημαντική πτυχή της εκπαίδευσης μέσω αυτών των παιχνιδιών. Οι εκπαιδευόμενοι αντιμετωπίζουν μεγάλο όγκο πληροφοριών και μαθαίνουν να τις επεξεργάζονται παράλληλα. Αυτός ο τρόπος σκέψης είναι χρήσιμος όχι μόνο στο παιχνίδι, αλλά και στον γενικότερο βίο τους, όπου αντιμετωπίζουν πολλές πληροφορίες ταυτόχρονα. Μαζί με την κριτική σκέψη βελτιώνονται και τα αντανακλαστικά καθώς πρέπει να αντιληφθεί ποιες πληροφορίες είναι χρήσιμες και ποιες όχι και να αποφασίζει την επόμενη κίνηση του μέσα στο χρονικό πλαίσιο του παιχνιδιού.
- Διδάσκουν στα παιδιά την ικανότητα να θέτουν στόχους και να αφιερώνουν την προσοχή τους για να τους επιτύχουν. Αυτό διαμορφώνει όχι μόνο το ψυχολογικό τους κλίμα, αλλά και τον χαρακτήρα και την ηθική τους συμπεριφορά.

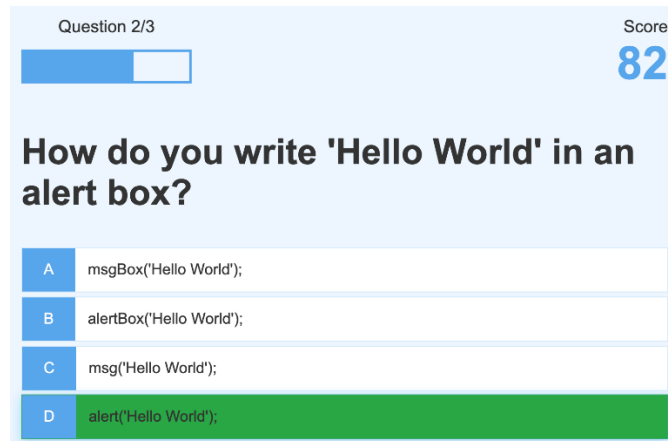
Οι μαθητές έχουν πολύ περισσότερα κίνητρα, επιδεικνύουν μεγαλύτερη συμμετοχή στη διαδικασία και έχουν την πρόθεση να συνεχίσουν να μαθαίνουν τόσο το αντικείμενο αλλά και τη χρήση της μάθησης μέσω παιχνιδιού. Τα αποτελέσματα από δοκιμές που έχουν γίνει έχουν δείξει ότι η ποιότητα της κατάκτησης της γνώσης και η διαδικασία της απομνημόνευσης είναι σχετικά πιο αποτελεσματική από ότι με τις συνηθισμένες εργασίες της τάξης.[8] Ενδιαφέρον προσέγγιση παρουσιάζει η παιγνιδοποιημένη μαθησιακή δραστηριότητα που έκανε χρήση ενός κουίζ με πολλαπλές επιλογές που σχετίζονται με την τεχνολογία της πληροφορικής σε τρία προπτυχιακά μαθήματα οι Cheong C., F και Filippou J. Χρησιμοποίησαν ένα ερωτηματολόγιο για τη συλλογή δεδομένων για την μέτρηση των

επιπέδων μάθησης, δέσμευσης και απόλαυσης. Τα παρακάτω αποτελέσματα αναφέρονται για χρήση που έγινε μέσα σε μικρή χρονικό περίοδο καθώς το συγκεκριμένο κουίζ χρησιμοποιήθηκε για περίοδο τεσσάρων εβδομάδων δύο φορές ανά μάθημα. Η πλειονότητα των συμμετεχόντων σε ποσοστό 77,63% ανέφεραν ότι ήταν αρκετά αφοσιωμένοι ώστε να ολοκληρώσουν το κουίζ και 44,6% ότι ήταν χαρούμενοι καθώς έπαιζαν το κουίζ. Όσον αφορά το μαθησιακό κομμάτι, τα συνολικά αποτελέσματα ήταν αντίστοιχα ενθαρρυντικά καθώς πάνω από τους μισούς μαθητές δήλωσαν ότι ενίσχυε τα μαθησιακά τους αποτελέσματα. Με την ολοκλήρωση του κουίζ εμφανίζονται τα αποτελέσματα στην οθόνη, έπειτα για να υπάρχει μια πιο ολιστική ανατροφοδότηση ο μαθητής μπορεί να συγκρίνει την πρόοδο του έναντι άλλων μαθητών της τάξης μέσα από ένα πίνακα κατάταξης. Ο πίνακας μπορεί να παρουσιάσει ποιόν ήταν το ποσοστό που απάντησαν σωστά και λάθος σε κάθε ερώτηση, καθώς και σε κάθε ερώτηση συγκεκριμένα πόσα άτομα επέλεξαν την κάθε επιλογή είτε αυτή ήταν σωστή είτε λάθος με αυτόν τον τρόπο δίνεται ένα περιθώριο συζήτησης όχι μόνο γιατί απάντησε λανθασμένα κάποιος αλλά τι ήταν αυτό που τον έκανε να επιλέξει ην συγκεκριμένη απάντηση έναντι κάποιας άλλης. Με αυτόν τον τρόπο δημιουργείται μια πιο ολοκληρωμένη άποψη για τον καθηγητή και για το σύνολο της τάξης αλλά και για τον κάθε μαθητή ποια κενά μπορεί να έχει, που πρέπει να εστιάσει περισσότερο την προσοχή του και τι είναι αυτό που γνωρίζει καλά. Μια ακόμα βασική παράμετρος που πρέπει να ληφθεί υπόψιν είναι πως τα κουίζ καλό θα ήταν να παραμένουν μικρά ώστε οι παίκτες να μην βαριούνται, επίσης η συμμετοχή να είναι εθελοντική. Αν θέλουμε οι μαθητές να διατηρήσουν το κίνητρο τους για συμμετοχή, είναι απαραίτητο να τους κάνουμε να νιώσουν πως το κουίζ έχει τη μορφή παιχνιδιού και δεν αποτελεί μέσο αξιολόγησης [14]

3.3 Είδη Κουίζ

Πολλές φορές για έναν δημιουργό παιχνιδιών η δημιουργία ενός κουίζ περιλαμβάνει μια μεγάλη γκάμα είδη κουίζ. Παρακάτω θα δούμε κάποιες από τις μορφές που μπορεί να έχει ένα κουίζ [18]:

- Πολλαπλής επιλογής: Είναι ο πιο διαδομένος τύπος κουίζ στην εκπαίδευση και όπως λέει και ο τίτλος του σκοπός των συμμετεχόντων είναι η επιλογή της σωστής απάντησης μέσα από πολλές επιλογές. Είναι ένας σχετικά γρήγορος και εύκολος τρόπος δημιουργίας κουίζ, δεν απαιτεί πολλή συμβολή από τους παίκτες και οι απαντήσεις μπορούν να αποκαλυφθούν γρήγορα, κρατώντας τους ανθρώπους αφοσιωμένους και συγκεντρωμένους.



Εικόνα 3.2: Παράδειγμα Quiz πολλαπλής επιλογής

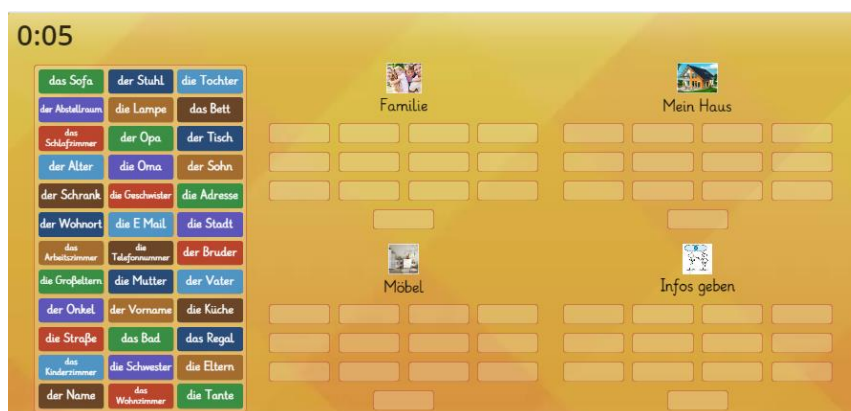
• Ερωτήσεις με εικόνα: Ένας ενδιαφέρον τύπος κουίζ είναι με την χρήση εικόνας, είναι ίσως ο πιο άμεσος τρόπος να κάνεις το κουίζ πιο προσιτό στην συμπλήρωση του. Οι εικόνες τραβούν το ενδιαφέρον, αυξάνουν την συμμετοχή και κάνει πιο εύκολη την επίτευξη του στόχου. Γνωστά παραδείγματα είναι τα κουίζ με ερωτήσεις “Ποια σημαία είναι αυτή;”, “Ποιο είναι το όνομα της παρακάτω δημοσιότητας;”. Άλλες εκδοχές της εικόνας μπορεί να είναι η ύπαρξη βίντεο ή ηχητικό άκουσμα.



Εικόνα 3.3: Παράδειγμα Quiz με εικόνα

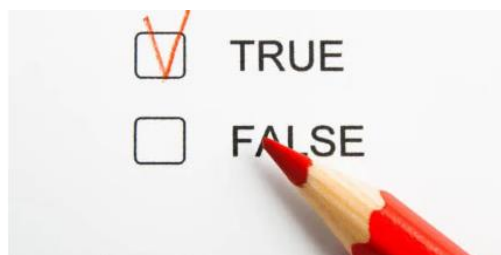
• Κουίζ με ταίριασμα: Πολλές φορές οι παίκτες πρέπει να κοιτάζουν το σύνολο των απαντήσεων και να επιλέξουν μια επιλογή που ταιριάζει στην ερώτηση, στην εικόνα ή στην δήλωση. Τα συγκεκριμένα κουίζ είναι αρκετά διαδραστικά και ευχάριστα, ιδανικά για την χρήση τους στα κοινωνικά δίκτυα. Εάν θες να δώσεις στο κοινό άμεση ανατροφοδότηση μπορείς να εμφανίσεις εάν η απάντηση ήταν σωστή

άμεσα, ενώ για περισσότερο αγωνία μπορείς να το εμφανίσεις στο τέλος. Το κουίζ μπορεί να είναι και ανάμεσα σε ζευγάρια και ιδανικότερα σε μαθήματα Γλώσσας όπου οι μαθητές ταιριάζουν τις ομώνυμες ή τις αντίθετες λέξεις.



Εικόνα 3.4: Παράδειγμα Quiz με ταίριασμα

- Σωστό/Λάθος ή ΝΑΙ/ΟΧΙ: Και αυτό το κουίζ έχει πολύ απλή δομή, δίνει μια πρόταση-δήλωση και περιμένει από τους παίκτες να αποφασίσουν αν είναι σωστή ή λάθος ή αντίστοιχα ναι ή όχι. Είναι από τις πιο διάσημες μορφές κουίζ και είναι και η πιο εύκολες να πετύχεις εάν έχεις γνώσεις ή με τύχη.



Εικόνα 3.5: Παράδειγμα Quiz Σωστού/Λάθους ή ΝΑΙ/ΟΧΙ

- Ερευνητικά κουίζ: Ένα στόχος είναι να μάθεις περισσότερα για το κοινό σου, καθώς τους δίνει και μια διασκεδαστική εμπειρία, μπορείς να δημιουργήσεις ένα ερευνητικό κουίζ. Είναι μια εναλλακτική μορφή της παραδοσιακής μορφής της έρευνας επειδή είναι πιο ελκυστικό, άρα δίνονται περισσότερες απαντήσεις. Είναι ένας χρήσιμος τρόπος για την συλλογή δεδομένων σεβόμενοι τα προσωπικά στοιχεία και τις πολιτικές απορρήτου του κοινού. Συχνά παρατηρούνται σε χρήση για ακαδημαϊκές εργασίες μέσα από το Google Forms όπου οι δημιουργοί θέλουν να συλλέξουν δεδομένα για να τα δημοσιεύσουν.

Art brief creator

Fill in the following fields to generate an art brief that will automatically be shared back to you, so you can include it in the #content-art-direction workflow.

Type 'none' or 'N/A' on any required field you don't know the answer to. If you don't have a Draft URL, just put slack.com in.

This form is automatically collecting email addresses for Slack users. [Change settings](#)

Title *
Short answer text

Writer *
Short answer text

Stakeholders *
Short answer text

Theme *

Εικόνα 3.6: Παράδειγμα Ερευνητικού Quiz.

4. Στάδια Ανάπτυξης Πολυμεσικής Εφαρμογής

Στο παρόν κεφάλαιο θα περιγράψουν τα στάδια ανάπτυξης μιας πολυμεσικής παραγωγής και αναλύθηκαν οι επιμέρους εργασίες που πραγματοποιούνται στην ανάλυση, σχεδίαση και στα επόμενα κεφάλαια θα συνεχίσει η ανάπτυξη και η αξιολόγηση.

Κατά τη διαδικασία ανάπτυξης λογισμικού υπάρχουν τα τέσσερα βασικά στάδια:

1. Προδιαγραφή απαιτήσεων
2. Σχεδιασμός
3. Παραγωγή πηγαίου κώδικα
4. Έλεγχος και διόρθωση σφαλμάτων

Πιο αναλυτικά οι παρακάτω κανόνες, προσδιορίζουν την μεθοδολογία που αναπτύσσεται μια πολυμεσική εφαρμογή [19]:

1. Καταγραφή Απαιτήσεων Ενδιαφερόμενων Μερών, Λογισμικού και Συστήματος (Requirements): Οι μηχανικοί λογισμικού επικοινωνούν με τα ενδιαφερόμενα μέρη, προκειμένου να πάρουν πλήρη περιγραφή του λογισμικού. Αυτή η περιγραφή περιλαμβάνει λειτουργικότητες, αλληλεπιδράσεις με εξωτερικά περιβάλλοντα και αισθητικές προδιαγραφές. Οι προδιαγραφές αυτές ονομάζονται "Requirements" ή "Απαιτήσεις". Επίσης, καταγράφονται οι τεχνικές απαιτήσεις του λογισμικού και του συστήματος γενικότερα.
2. Τυποποίηση Απαιτήσεων και Αφαιρετικός Σχεδιασμός Λογισμικού (Design): Οι μηχανικοί λογισμικού μεταφράζουν τις απαιτήσεις σε διαγράμματα σχέσεων μεταξύ οντοτήτων και ροής εργασιών.
3. Υλοποίηση Λογισμικού (Construction): Ο σχεδιασμός μετατρέπεται σε κώδικα που πληροί τις απαιτήσεις. Αυτή η διαδικασία αποτελεί τον πυρήνα της ανάπτυξης λογισμικού, καθώς με αυτήν η ιδέα γίνεται πράξη.
4. Συνολική Δοκιμή Λογισμικού (Testing): Ο κώδικας υποβάλλεται σε δοκιμές, με την εκτέλεση τεστ που προσομοιώνουν τη λειτουργία του λογισμικού. Κατά τη διαδικασία αυτή ελέγχεται αν όλα λειτουργούν σύμφωνα με τις απαιτήσεις, αλλά και αν υπάρχουν συντακτικά ή λογικά λάθη στον κώδικα.
5. Συνολική Αποσφαλμάτωση Λογισμικού (Debugging): Κατά την διαδικασία αυτή, οι μηχανικοί εντοπίζουν και διορθώνουν τα λάθη που προέκυψαν από τις δοκιμές.

6. Ανάπτυξη Λογισμικού (Deployment): Αφού το λογισμικό ελεγχθεί και λειτουργήσει σωστά, οι μηχανικοί αναλαμβάνουν την εγκατάστασή του, αντικαθιστώντας τα προσωρινά συστήματα με πραγματικά, όπως εξυπηρετητές, βάσεις δεδομένων και δικτυακές συνδέσεις.

7. Συντήρηση Λογισμικού (Maintenance): Λόγω της συνεχούς εξέλιξης, το λογισμικό χρειάζεται συντήρηση για να προσαρμοστεί σε νέες ανάγκες και απαιτήσεις. Αυτή η διαδικασία περιλαμβάνει την ενημέρωση του λογισμικού με νέο κώδικα, συστήματα και λειτουργίες που ανταποκρίνονται στις εξελίξεις.

4.1 Ανάλυση

Στο συγκεκριμένο υποκεφάλαιο πραγματοποιήθηκε η επισκόπηση της θεματολογίας Λογικών Πυλών και η παρουσίαση εκπαιδευτικών λογισμικών με την αντίστοιχη θεματική. Στο στάδιο ανάλυσης μιας πολυμεσικής εφαρμογής, ο αναγνώστης αντιλαμβάνεται τα αρχικά βήματα και τις αποφάσεις που πρέπει να ληφθούν για την έναρξη της παραγωγής της. Ο αναγνώστης θα έχει μια πιο σαφή εικόνα για την ανάγκη δημιουργίας ενός ψηφιακού παιχνιδιού με θέμα τα ψηφιακά ηλεκτρονικά, ενώ θα γνωρίσει και τα βήματα και τις απαιτήσεις για τα επόμενα βήματα.

4.1.1 Λογικές Πύλες και Εκπαίδευση

Οι φοιτητές ή ακόμα και οι μαθητές που παρακολουθούν μαθήματα που συσχετίζονται με τους υπολογιστές ή τα ηλεκτρονικά συστήματα πρέπει να μάθουν και να κατανοούν καλά τα βασικά του ψηφιακού συστήματος. Τα ψηφιακά συστήματα κατασκευάζονται χρησιμοποιώντας λογικές πύλες. Μια λογική πύλη είναι μια απλά συσκευή η οποία “επιστρέφει” κάποιο αριθμό εξόδων. Αυτές οι έξοδοι καθορίζονται από ένα μοτίβο εισόδων και κανόνων που ακολουθεί η πύλη για να δημιουργήσει ένα επιθυμητό αποτέλεσμα. Στον πυρήνα τους, οι πύλες επιστρέφουν είτε ως “αληθής και ψευδής” ή “ανοιχτές και κλειστές”. Οι λογικές πύλες είναι το θεμέλιο όχι μόνο της υπολογιστικής σκέψης και κωδικοποίησης, αλλά ισχύουν και για την ηλεκτρική μηχανική και την αλγεβρική σκέψη. Οι λογικές πύλες συνήθως διδάσκονται στις διαλέξεις, αλλά απαιτούνται πρακτικές δεξιότητες μέσα από περάματα για να ληφθεί το αποτέλεσμα. Μάθημα όπως η “Οργάνωση και Αρχιτεκτονική Υπολογιστών” που υπάρχει σε διάφορους τομείς σπουδών όπως η Πληροφορική, η Τέχνη, ή η Δημιουργική Βιομηχανία οι λογικές πύλες είναι σημαντικές. [20] Στην εικόνα βλέπουμε τις βασικές πύλες και τους πίνακες αληθείας τους. Σύμφωνα με τον ορισμό του Βικιπαίδεια πίνακας αληθείας είναι ένας "λογικός" πίνακας που χρησιμοποιείται για τον υπολογισμό των λογικών πράξεων όπως: AND, OR, NOT, NOR, XOR. Σχεδιαστικά ο πίνακας αληθείας αποτελείται από στήλες που περιέχουν όλες τις μεταβλητές και το αποτέλεσμα ενώ οι γραμμές του περιέχουν όλους τους δυνατούς συνδυασμούς των τιμών των μεταβλητών.

ΟΝΟΜΑ	NOT	AND	NAND	OR	NOR	XOR	XNOR																																																																																																
ΕΚΦΡΑΣΗ	\bar{A}	AB	\overline{AB}	$A+B$	$\overline{A+B}$	$A\oplus B$	$\overline{A\oplus B}$																																																																																																
ΣΥΜΒΟΛΟ																																																																																																							
ΠΙΝΑΚΑΣ ΑΛΗΘΕΙΑΣ	<table border="1"> <thead> <tr> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	X	0	1	1	0	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	B	A	X	0	0	0	0	1	0	1	0	0	1	1	1	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	B	A	X	0	0	1	0	1	1	1	0	1	1	1	0	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	B	A	X	0	0	0	0	1	1	1	0	1	1	1	1	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	B	A	X	0	0	1	0	1	0	1	0	0	1	1	0	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	B	A	X	0	0	0	0	1	1	1	0	1	1	1	0	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	B	A	X	0	0	1	0	1	0	1	0	0	1	1	1
A	X																																																																																																						
0	1																																																																																																						
1	0																																																																																																						
B	A	X																																																																																																					
0	0	0																																																																																																					
0	1	0																																																																																																					
1	0	0																																																																																																					
1	1	1																																																																																																					
B	A	X																																																																																																					
0	0	1																																																																																																					
0	1	1																																																																																																					
1	0	1																																																																																																					
1	1	0																																																																																																					
B	A	X																																																																																																					
0	0	0																																																																																																					
0	1	1																																																																																																					
1	0	1																																																																																																					
1	1	1																																																																																																					
B	A	X																																																																																																					
0	0	1																																																																																																					
0	1	0																																																																																																					
1	0	0																																																																																																					
1	1	0																																																																																																					
B	A	X																																																																																																					
0	0	0																																																																																																					
0	1	1																																																																																																					
1	0	1																																																																																																					
1	1	0																																																																																																					
B	A	X																																																																																																					
0	0	1																																																																																																					
0	1	0																																																																																																					
1	0	0																																																																																																					
1	1	1																																																																																																					

Εικόνα 4.1: Πίνακες Αληθείας Λογικών Πυλών

Για την αποτελεσματικότητα της μάθησης μέσω παιχνιδιού σε θέματα λογικών πυλών παρατηρείτε να έχουν αναφερθεί και η D Dewantara μέσα από την χρήση του παιχνιδιού “Circuit Scrabble ark.” [21]. Η έρευνα διεξάγει στο Τμήμα Φυσικής Εκπαίδευσης, σε είκοσι τρεις μαθητές από τους οποίους συμμετείχαν στο μάθημα Ψηφιακά Ηλεκτρονικά το 2019, η εφαρμογή ήταν της μορφής παρέχει ένα Tutorial δηλαδή μια περίληψη για πληροφορίες διάφορων λογικών πυλών και για να προχωρήσεις στο επόμενο επίπεδο, δηλαδή στην επόμενη πύλη πρέπει να συμπληρώσεις σωστά την πύλη στο επίπεδο που βρίσκεσαι. Τα βήματα που ακολούθησε για την μάθηση μέσω παιχνιδιού ήταν τα εξής:

1. Ο εισηγητής αποφασίζει το είδος και τον τύπο του παιχνιδιού σύμφωνα με το είδος του εκπαιδευτικού αντικειμένου.
2. Ο εισηγητής χωρίζει τις ομάδες του παιχνιδιού
3. Ο εισηγητής παροτρύνει όλους του μαθητές να συμμετέχουν στο παιχνίδι
4. Οι μαθητές εμπλέκονται, ακολουθούν και ελέγχουν το παιχνίδι
5. Οι μαθητές χαίρονται την νίκη τους
6. Οι μαθητές συζητάνε για τις εμπειρίες και την γνώση που κέρδισαν μέσα από το παιχνίδι
7. Ο εισηγητής παρέχει την κατάλληλη ανατροφοδότηση δίνοντας έμφαση στα εκπαιδευτικά αντικείμενα

Η εφαρμογή δεν χρησιμοποιείτε αποκλειστικά τις ώρες του μαθήματος, προσφέρεται αυτή η ευελιξία, ώστε ο μαθητής να μάθει να εκπαιδευτεί και να ανακαλύπτει μόνος του. Μέσω της έρευνας στην οποία συλλέξανε στοιχεία πριν την διεξαγωγή του τεστ και μετά και παρατήρησαν 66,53 αύξηση στην επιτυχία του τεστ, μετά την διεξαγωγή του παιχνιδιού. Οπότε θεωρήθηκε υψηλής αποτελεσματικότητας η μάθηση με αυτό τον τρόπο σε αντικείμενο λογικών πυλών.



Εικόνα 4.2: Όψη παιχνιδιού “Circuit Scramble”

Μια αξιόλογη επίσης αναφορά είναι το 7-Gates (7- Πύλες) ένα εκπαιδευτικό παιχνίδι με φουτουριστική περιπέτεια. Ο μόνος τρόπος για τον παίχτη να σώσει τον ψηφιακό κόσμο είναι μέσα από την χρήση λογικών πυλών, αν δεν γνωρίζει την χρήση τους χάνει την ισχύ του κινητήρα στον ψηφιακό κόσμο. Το παιχνίδι είναι σύνθετου είδους αλλά κατά κύριο λόγο εκπαιδευτικό, το παζλ που περιλαμβάνεται σε αυτό το παιχνίδι σε συνδυασμένο με ένα στυλ πλατφόρμας ο παίκτης πρέπει να συγκεντρώσει όλους τους διακόπτες στον χάρτη της για να περάσει στο επόμενο επίπεδο. Με την ολοκλήρωση κάθε επιπέδου, έχει κατανόησή και απομνημόνευση ο παίκτης τον κάθε τύπο πυλών (NOT, AND, OR, NAND, NOR, XOR και XNOR). Το στοχευμένο κοινό είναι από μαθητές γυμνασίου μέχρι φοιτητές Τριτοβάθμιας Εκπαίδευσης, ανεξαρτήτως φύλου και ο κύριος στόχος είναι μια διαφορετική οπτική της παρουσίας των πυλών. Σύμφωνα με τα σχόλια των παιχτών το παιχνίδι θεωρήθηκε από πολλούς βαρετό με βάση το περιεχόμενο αλλά στα θετικά σχόλια ήταν ότι πετυχαίνει τον στόχο του και μπορεί να θεωρηθεί χρήσιμο για την χρήση σε ακαδημαϊκό επίπεδο.[22]



Εικόνα 4.3: Μενού παιχνιδιού 7 Gates

4.2.1 Αρχές Χρηστικότητα

Τρεις βασικές αρχές που καθορίζουν την χρηστικότητα και τον τρόπο αλληλεπίδρασης σε ένα ψηφιακό παιχνίδι έχουν ήδη αναφερθεί. Στην πραγματικότητα, το ψηφιακό παιχνίδι είναι μια διαδραστική εφαρμογή όπου ο χρήστης προστρέχει σε συγκεκριμένες ενέργειες και αλληλεπιδρά με το εικονικό περιβάλλον. Για να διασφαλιστεί η μέγιστη χρηστικότητα μιας τέτοιας εφαρμογής, εφαρμόζονται συγκεκριμένες αρχές και μεθοδολογίες στη διαδραστική σχεδίαση, με στόχο τη βελτίωση της ευκολίας χρήσης. Αυτή η διαδικασία, γνωστή και ως σχεδίαση διάδρασης ή αλληλεπίδρασης ανθρώπου-υπολογιστή, αποσκοπεί στη βελτίωση της χρηστικότητας (usability) [23].

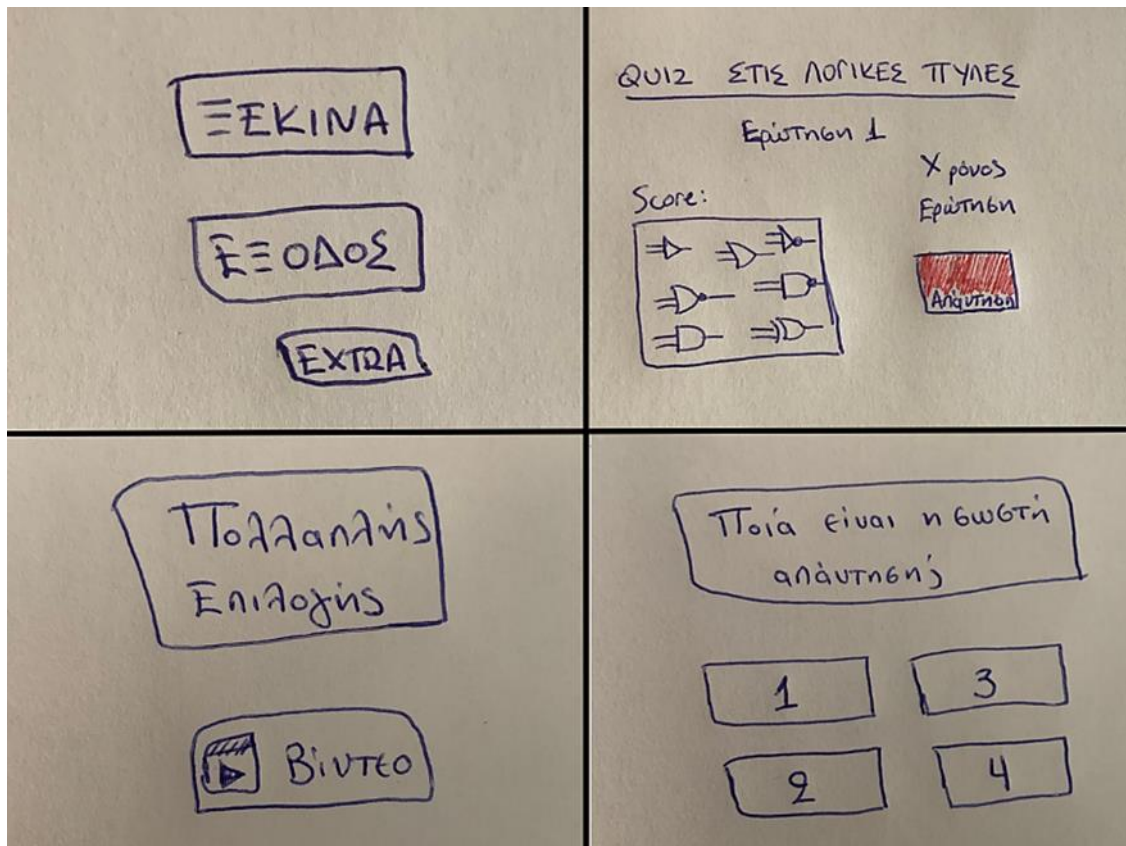
Ο όρος "χρηστικότητα" αποτελείται από πέντε βασικά στοιχεία που βελτιώνουν τον σχεδιασμό μιας ψηφιακής εφαρμογής, καθιστώντας την πιο κατανοητή από τους χρήστες. Συγκεκριμένα, η πρώτη συνιστώσα, γνωστή ως "learnability," αφορά τον βαθμό ευκολίας με τον οποίο ένας χρήστης μαθαίνει να χρησιμοποιεί την εφαρμογή. Η δεύτερη, "efficiency," αναφέρεται στην ταχύτητα με την οποία ένας χρήστης μπορεί να ολοκληρώσει μια εργασία. Η "memorability" περιγράφει τον βαθμό ευκολίας με τον οποίο ένα άτομο μπορεί να θυμηθεί πώς να χρησιμοποιήσει την εφαρμογή αφού περάσει κάποιο χρονικό διάστημα χωρίς χρήση. Ακολουθεί η "διαχείριση σφαλμάτων," που αφορά τον σωστό σχεδιασμό της εφαρμογής για να αποτρέψει λάθη και να διευκολύνει την αντιμετώπιση τους αν συμβούν. Τέλος, η έννοια της "ικανοποίησης" περιγράφει την γενική αίσθηση ικανοποίησης και ευχαρίστησης που προσφέρει η εφαρμογή στο χρήστη [24].

4.2.2 Σχεδίαση - Πρότυπα Χαμηλής και Υψηλής Πιστότητας

Ο σχεδιασμός μιας εφαρμογής περιλαμβάνει τις διαδικασίες οργάνωσης, καθορισμού του περιεχομένου της, καθώς και την ανάλυση και επιλογή των εργαλείων που θα χρησιμοποιηθούν για την ανάπτυξή της. Αυτό το στάδιο ανάπτυξης πολυμεσικής εφαρμογής έχει μεγάλη σημασία, καθώς καθορίζει τον τρόπο με τον οποίο το τελικό προϊόν θα παρουσιαστεί και θα παρέχει υπηρεσίες στον χρήστη.

Ο τίτλος που έχει επιλεγεί για το συγκεκριμένο παιχνίδι είναι "Quiz στις λογικές πύλες", που είναι κυριολεκτικός εν μέρη τίτλος αλλά δεν απευθύνεται όμως για όλο το παιχνίδι. Δεδομένου ότι ο σκοπός του παιχνιδιού είναι εκπαιδευτικός, κρίθηκε ότι ένας όχι άμεσα ξεκάθαρος τίτλος -όσο εμπνευσμένος κι αν ήταν αυτός- όχι μόνο δεν θα προσέφερε κάτι, αλλά -τουναντίον- ενδεχομένως και να «θόλωνε» το εκπεμπόμενο μήνυμα. Όσο αφορά το είδος του παιχνιδιού είναι ένα παιχνίδι για τον προσωπικό ηλεκτρονικό υπολογιστή που προωθεί την επίλυση προβλημάτων υπό την μορφή γρίφων, το κοινό είναι κατά κύριο λόγο μαθητευόμενοι, δηλαδή μαθητές ή φοιτητές που έχουν παρακολουθήσει μαθήματα πάνω στα ψηφιακά ή έχουν γνώσεις.

Όταν αναφερόμαστε στην σχεδίαση ενός παιχνιδιού αναφερόμαστε στην “βάση” του παιχνιδιού. Τα πρότυπα χαμηλής πιστότητας αποτελούν την πρώτη αποτύπωση μια ιδέας με την μορφή γραφικού προτύπου. Αφού δημιουργηθεί η αρχική ιδέα ουσιαστικά δημιουργούμε αυτό το πρότυπο για να γίνει πλήρως κατανοητό πριν αρχίσει η κανονική διαδικασία του σχεδιασμού. Τα αρχικά αυτά πρότυπα επιτρέπουν στην ομάδα σχεδιασμού να καθορίσει με μεγαλύτερη ακρίβεια στην εμφάνιση (και κάποιες φορές τα λειτουργικά χαρακτηριστικά) του συστήματος που είναι υπό ανάπτυξη. Σχεδιάστηκαν σαν σκίτσα σε πολύ πρόχειρο και πρώτο επίπεδο, πριν ακόμα την ενασχόληση μας με το Unity.



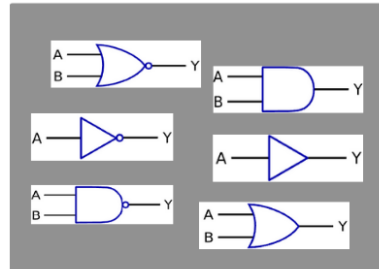
Εικόνα 4.4: Πρωτότυπη Οθόνη Χαμηλής Πιστότητας- Διεπαφή Χρήστη (UI)

Μετά τα πρότυπα χαμηλής πιστότητας, το τελευταίο στάδιο πριν την ολοκλήρωση της σχεδίασης αφορά την δημιουργία προτύπων που αντιπροσωπεύουν τελικά, αναπαραστατικά παραδείγματα πως θα φαίνονται στην οθόνη. Σε αυτό το στάδιο συμπεριλαμβάνονται τα γενικότερα αισθητικά στοιχεία, τα γραφικά με μεγαλύτερη λεπτομέρεια στον σχεδιασμό για αυτό και είναι γνωστά ως πρότυπα υψηλής πιστότητας (High Fidelity Prototypes). Αυτό το στάδιο ουσιαστικά αποτελεί την μετάβαση από τη σκέψη σε χαρτί σε πλήρη ψηφιακή αναπαράσταση. [25]

QUIZ ΣΤΙΣ ΛΟΓΙΚΕΣ ΠΥΛΕΣ

Question 1

Score



Timer

Ερώτηση

Απαντήσεις

Εικόνα 4.5: Πρωτότυπη Οθόνη Υψηλής Πιστότητας- Quiz στις λογικές πύλες

Η πρώτη σκηνή που θέλουμε να εμφανίζεται στο χρήστη είναι το MenuScene, σε αυτό έχει να επιλέξει ανάμεσα σε τρία κουμπιά τα ΞΕΚΙΝΑ, ΕΞΟΔΟΣ και το EXTRA. Το ξεκίνα σηματοδοτεί την εκκίνηση στο Quiz, το ΕΞΟΔΟΣ την διακοπή της λειτουργίας την εφαρμογής και το Extra για να καλύψεις περισσότερο κομμάτι γνώσεων με απάντηση σε ακόμα είκοσι ερωτήσεις πολλαπλής επιλογής και την παρακολούθηση βίντεο για περαιτέρω βοήθεια πάνω στις ερωτήσεις. Η δεύτερη σκηνή είναι η SampleScene που εμφανίζει το παιχνίδι που σχεδιάσαμε που έχει την μορφή “drag and drop” δηλαδή διαλέγω και αφήνω σύμφωνα με την σχεδίαση του UI (User Interface).

5. Περιβάλλον δημιουργίας διάδρασης

Το κεφάλαιο που ακολουθεί παρουσιάζει την παιχνιδομηχανή «Unity», δηλαδή το σύστημα που επιλέχθηκε για τη δημιουργία του πολυμεσικού έργου. Ειδικότερα, γίνεται περιγραφή στα πλεονεκτήματα που έχει σε σύγκριση με άλλες εφαρμογές και περισσότερες λεπτομέρειες σε ένα από τα βασικά εργαλεία που χρησιμοποιήσαμε το UI Toolkit και τη σχεδίαση μέσα από το UI Builder.

5.1 Εισαγωγή στο Unity

Το Unity χρησιμοποιείτε κατά κύριο λόγο ως ένα εργαλείο για την ανάπτυξη ηλεκτρονικών παιχνιδιών 2D και 3D. Προσφέρει γραφικό περιβάλλον, ο χρήστης μπορεί να βλέπει και να ελέγχει σε πραγματικό χρόνο όσα υλοποιεί ενόσω αυτό εκτελείται (run-time editing). Με αυτό τον τρόπο γίνεται πιο εύκολη η δημιουργία του κόσμου του παιχνιδιού, όπως και η σωστή τοποθέτηση των αντικειμένων. Ένα ακόμα μεγάλο του πλεονέκτημα, είναι ότι προσφέρει πληθώρα επιλογών και δυνατοτήτων, ώστε τα αντικείμενα και οι χαρακτήρες να αποκτήσουν συμπεριφορά. Εύκολη ενσωμάτωση scripts γραμμένα σε C# ή Javascript τα οποία μπορούν να αντιστοιχηθούν με συγκεκριμένα αντικείμενα δίνοντας τους ξεχωριστές ιδιότητες. Επίσης η άμεση σύνδεσή του με επεξεργαστές κώδικα (code editors) όπως το Visual Studio ή ο ενσωματωμένος MonoDevelop μια open-source, συμβατή έκδοση που τρέχει και σε Linux περιβάλλοντα, προσφέρει ευκολία στη εγγραφή κώδικα καθώς είναι, απαραίτητη για τη λειτουργία του παιχνιδιού. [26]

5.2 Πλεονεκτήματα

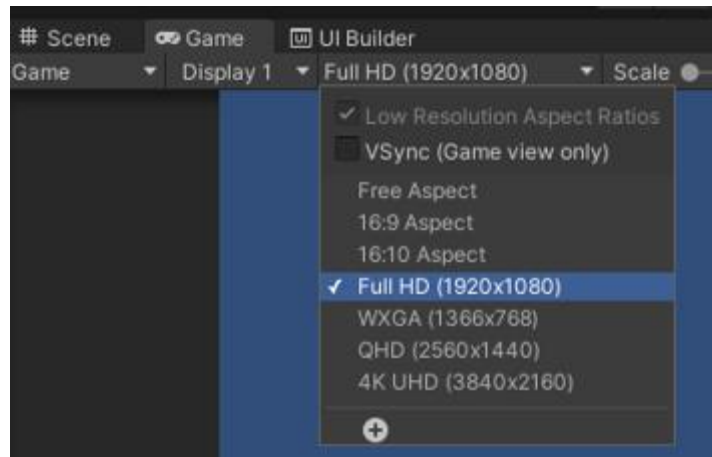
Ένα από τα κυριότερα πλεονεκτήματα που προσφέρει το Unity είναι η δημιουργία για πολλές πλατφόρμες. Μια σημαντική επίσης δυνατότητα που προσφέρει είναι ότι μπορείς να αλλάξεις την πλατφόρμα παιχνιδιού, δηλαδή ακόμα και αν έχεις δημιουργήσει σε μια πλατφόρμα μπορείς να το μεταφέρεις σε μια άλλη χωρίς πολύπλοκες και πολύωρες διαδικασίες. Cross-platform game engine, (με τον όρο αυτό εννοούμε το είδος software που είναι συμβατό με διάφορες πλατφόρμες- συσκευές για παράδειγμα Windows, Mac Os, Android και iOS και σε άλλες 23 ακόμα πλατφόρμες) που κατασκευάστηκε πρώτη φορά από το Unity Technologies, τον Ιούνιο του 2005. Έχει την δυνατότητα να δημιουργήσει και τρισδιάστατα (3D) και δισδιάστατα (2D) παιχνίδια όπως και προσομοιώσεις και διάφορα άλλα. Κάποια από τα πιο γνωστά παιχνίδια που αναπτύχθηκαν στη Unity engine είναι το Pokemon Go, Monument Valley, Call of Duty: Mobile, Assassin's Creed: Identity και πολλά άλλα. Όταν αναφερόμαστε σε Indie Game Development, τρεις είναι οι κύριες επιλογές το Unreal, Unity ή GameMaker. Είναι επίσης ένας εξαιρετικός χώρος για τη δημιουργία ολοκληρωμένων διαδραστικών πρωτοτύπων για UX design ή studies. Ένα ακόμα σημαντικό στοιχείο του Unity είναι η ύπαρξη του Asset Store, απ' όπου μπορεί ο καθένας να κατεβάσει δωρεάν ή επί πληρωμή αντικείμενα, χαρακτήρες, στοιχεία του περιβάλλοντος και άλλα, που θα βοηθήσουν στην υλοποίηση του παιχνιδιού. Εμείς το

επωφεληθήκαμε μόνο για την μουσική όπου χρησιμοποιήσαμε ήχους 2D, αλλά προσφέρεται και 3D ήχος κατά κύριο λόγο σε 3D project όπου ο ήχος έχει την ιδιαιτερότητα ότι μπορεί να αλλάζει ανάλογα με την απόσταση και την κίνηση σύμφωνα με την κάμερα. Αξιόλογη είναι η ύπαρξη μιας μεγάλης ενεργής κοινότητας η οποία προσφέρει επικοινωνία μεταξύ των χρηστών όλων των επιπέδων. Είναι μια πηγή καθοδήγησης ιδιαίτερα για τους νέους χρήστες. Επίσης είναι διαθέσιμα διάφορα tutorials (δηλαδή διάφορα βίντεο που εξηγούν βήμα βήμα πως μπορείς να φτιάξεις διάφορα project) είτε από το επίσημο ιστότοπο <https://learn.unity.com/> είτε από απλούς χρήστες του λογισμικού.

5.3 Απαιτήσεις και δημιουργία Project

Ο μεγαλύτερος ανταγωνιστής στις πλατφόρμες Game Engine του Unity είναι το Unreal engine που είναι εξίσου ευρέως γνωστό στους προγραμματιστές. Και τα δύο αυτά λογισμικά έχουν τα θετικά τους και τα αρνητικά τους. Αρχικά το Unreal Engine χρησιμοποιεί ως γλώσσα προγραμματισμού C++, ενώ το Unity C# που είναι προτιμότερο από τον προγραμματιστικό χώρο. Η πολύ μεγάλη ποικιλία στο Asset store που έχει το Unity επίσης κερδίζει στον ανταγωνισμό την Unreal. Από την άλλη πλευρά το Unreal παρέχει μια καλύτερη ανάλυση αν έχεις απαίτηση σε γραφικά υψηλής ποιότητας. Για την εφαρμογή που δημιουργήθηκε εμείς δεν είχαμε ανάγκη για την κάλυψη τέτοιων αναγκών από άποψη ανάλυσης, άρα επιλέξαμε το unity ώστε να μην επιβαρύνουμε και τον υπολογιστή εργασίας μας παραπάνω.

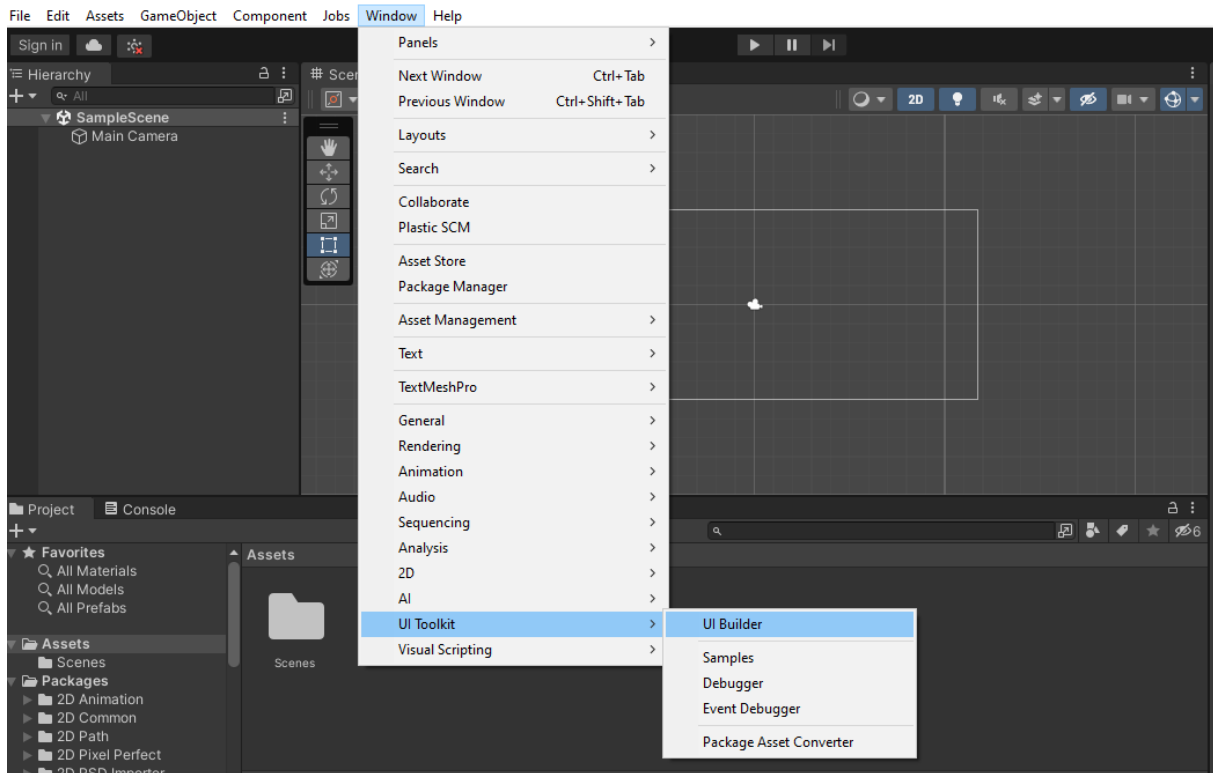
Αρχικά μπαίνοντας στην επίσημη σελίδα <https://unity.com>. Επιλέξαμε να κατεβάσουμε το Unity Personal το οποίο παρέχεται εντελώς δωρεάν, υπάρχει και η Pro έκδοση ως επιλογή που περιέχει κάποια εξτρά χαρακτηριστικά και δημιουργήσαμε ένα Unity ID δηλαδή το δικό μας προσωπικό λογαριασμό που θα μας παρέχει την είσοδο σε οτιδήποτε άλλο παρέχει το Unity ως πλατφόρμα. Το Unity πέρα από την δημιουργία παιχνιδιών παρέχει ένα ευρύ φάσμα δυνατοτήτων όπως Ανάπτυξη Εφαρμογών Επαυξημένης και Εικονικής Πραγματικότητας (AR,VR), χρήση τεχνητής νοημοσύνης (AI) αλλά και εργαλεία κινουμένων σχεδίων. Επιλέξαμε το Windows 10 64-bit μιας και αυτό το λογισμικό που υποστηρίζει ο προσωπικός φορητός υπολογιστής ο οποίος έχει τα παρακάτω τεχνικά χαρακτηριστικά: 2,4 GHz Intel Core i5, 8 GB, 1067 MHz, 15,4” panel with 1440x900 resolution, 256MB Crucial M4 SSD 256GB storage. Αφού έχουμε κατεβάσει το Unity Hub μπορούμε επιτέλους να κατεβάσουμε την version του Unity Editor που επιθυμούμε, στην προκειμένη περίπτωση διαθέσιμη ήταν η 2022.1.21f1. Για την δημιουργία του παιχνιδιού αρχικά δημιουργήσαμε ένα New Project, επιλέξαμε το όνομα και την επιλογή 2D Core και τον φάκελο που θέλουμε να αποθηκευτεί. Έπειτα δημιουργούμε το UI Builder που μας δίνει σαν επιλογή το UI Toolkit που εκεί είναι βασισμένος και ο κύριος κορμός του ενός από τα δύο μέρη του Quiz μας και επιλέγουμε τις διαστάσεις Full HD (1920x1080).



Εικόνα 5.1: Επιλογή διαστάσεων παιχνιδιού

5.4 Ανάπτυξη σε UI Toolkit

Το Unity μας προσφέρει μια πολύ μεγάλη πληθώρα ανάπτυξης Διεπαφή Χρήστη-UI (User Interface) κυρίως μέσα από το UI Toolkit για χρήση σε παιχνίδια και εφαρμογές, η φιλοσοφία του είναι βασισμένη στις γνωστές Standard Web Technologies, όπως δηλαδή είναι η HTML, CSS και JavaScript κάποιες από τις κύριες γλώσσες που χρησιμοποιούνται για την δημιουργία ιστοσελίδων. [27] Μέσα στο UI Toolkit περιέχονται πολλά χρήσιμα εργαλεία τα οποία κάναμε και χρήση στην εργασία και θα δούμε περαιτέρω παρακάτω την χρήση τους. Πάντως, αξίζει να αναφερθεί πως πέρα από την ανάπτυξη UI μέσω UI Toolkit υπάρχουν άλλα δύο συστήματα UI, το Unity UI (uGUI) και το Immediate Mode GUI (IMGUI). Το UI Toolkit προτείνεται κατά βάση καθώς είναι και το πιο πρόσφατα τελειοποιημένο. Τα υπόλοιπα παραμένουν μόνο για συγκεκριμένες χρήσεις και όταν χρειάζονται για υποστήριξη σε παλαιότερες εφαρμογές. Εμείς χρησιμοποιήσαμε την έκδοση 2022.1.21f του Editor. Με αυτό το UI πρώτη φορά μας δίνεται η δυνατότητα διαχωρισμού του UI Toolkit από το υπόλοιπο παιχνίδι (Game). [28] Η αισθητική προσέγγιση ενός λογισμικού διαδραματίζει πολύ σημαντικό ρόλο στην επιτυχία μιας εφαρμογής. Όπως υποδηλώνει το όνομα, τα γραφικά περιβάλλοντα-GUI (Graphic UI) παρουσιάζουν χαρακτηριστικά και λειτουργίες οπτικά. Η αλληλεπίδραση ανθρώπου-ηλεκτρονικού υπολογιστή βασίζεται σε μεγάλο βαθμό στο να βλέπει τα πράγματα, να ερευνά και να αλληλοεπιδρά με τα γραφικά στοιχεία UI. Οι εμπειρικές μελέτες δείχνουν ότι προτιμώνται ζεστά χρώματα (κόκκινο, κίτρινο, πορτοκαλί) σε υπόβαθρα ψυχρών χρωμάτων (μπλε, πράσινο, μωβ) και αντίστροφα. Γενικά, τα χρώματα είναι κατάλληλα για να αυξήσουν την οπτική εμφάνιση ενός προϊόντος λογισμικού επίσης τα ελκυστικά γραφικά περιβάλλοντα διαθέτουν χρώματα που είναι συντονισμένα.

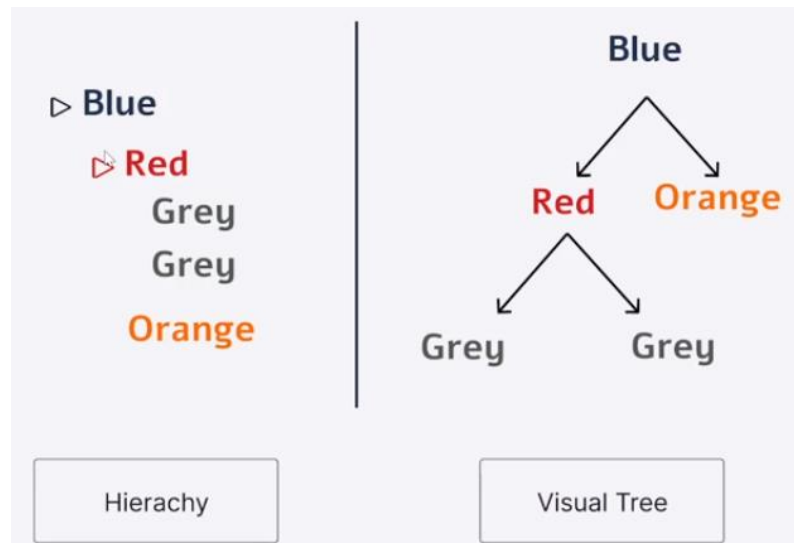


Εικόνα 5.2: Επιλογή UI Builder

5.5 Ροή Εργασίας μέσω UI Builder

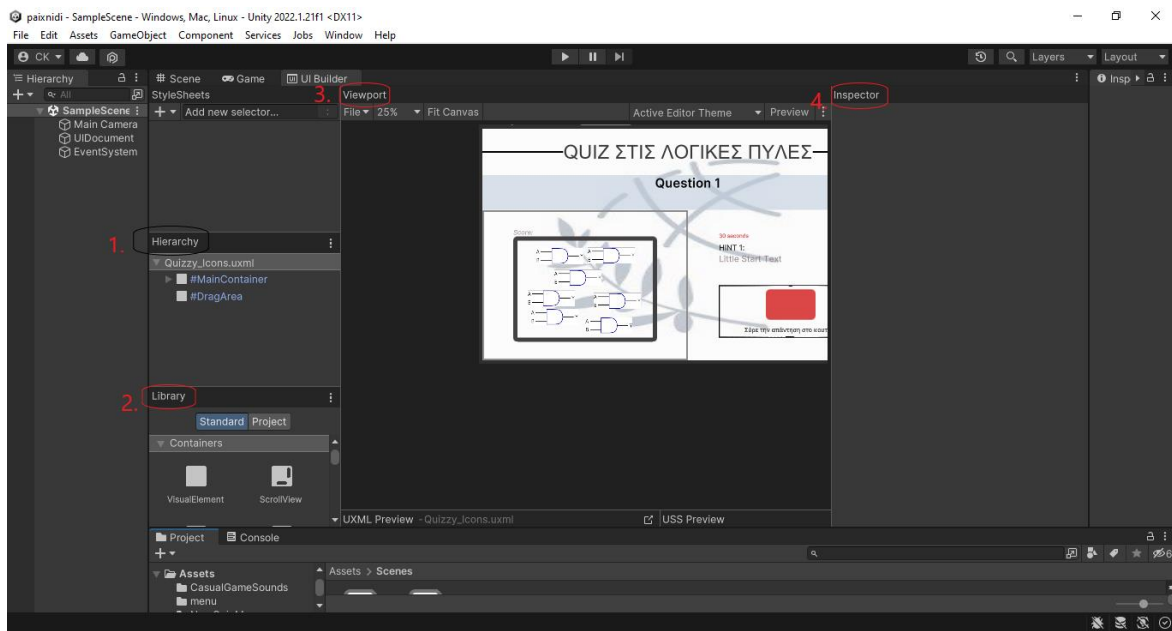
Ο UI Builder είναι ένας νέος editor που προσφέρει το Unity για το UI Toolkit. Προσφέρει αρκετή ευκολία και είναι πιο απλό στην διαχείριση του καθώς επίσης προσφέρει την δυνατότητα να αλλάξεις και να προσαρμόζεις την σχεδίαση, παρατηρώντας απευθείας το τελικό αποτέλεσμα. Έχει αρκετές ομοιότητες με το Unity Game καθώς διαθέτει δικά του ξεχωριστά παράθυρα Hierarchy και Inspector. Με προσθήκη κάποιων ακόμα παραθύρων όμως όπως StyleSheets, Viewport και Library. Στο Viewport (3) παράθυρο μπορείς να αλλάξεις το μέγεθος να ορίσεις εσύ απευθείας ποιες διαστάσεις επιθυμείς ή να επιλέξεις να προσαρμοστεί στο ίδιο μέγεθος με το Gameview δηλαδή με το πως θα φαίνεται στο τέλος όταν θα πατήσεις να ξεκινήσεις το παιχνίδι. Στο Hierarchy (1) είναι το παράθυρο που γίνεται όλη η οργάνωση του UI ουσιαστικά, όπως και το Hierarchy του Unity αντίστοιχα και σε αυτό τα Elements τοποθετούνται σε μια σειρά parent-child με την οποία ο “γονέας” είναι αυτός που περιέχει μέσα τα “παιδιά”. Αυτό το παρατηρούμε στην σχεδίαση και παίζει σημαντικό ρόλο στο να μπορούμε να διαχωρίσουμε τι ιδιότητες έχει κάθε Element και πως επηρεάζεται αυτό σε σχέση με την σχέση που έχει με τα υπόλοιπα Elements. Αυτή η σχέση parent-child την έχουν παρομοιάσει κατά καιρούς σαν Visual Tree δηλαδή σαν δενδρόγραμμα και έχει τα θετικά χαρακτηριστικά της αποτελεσματικότητας, της ευκολίας στην διαμόρφωση καθώς υπάρχει μια σειρά και γίνονται εύκολα

διαχωρισμοί. Όπως βλέπουμε και στο παράδειγμα της παρακάτω εικόνας το Hierachy έχει την αναπαράσταση ενός Visual Tree ή αλλιώς μπορούμε να το δούμε ως ένας γενεαλογικό δέντρο.



Εικόνα 5.3: Παράδειγμα Visual Tree

Ακριβώς από κάτω από το Hierarchy βρίσκεται το παράθυρο Library όπου έχουν πολύ μεγάλη αλληλεπίδραση. Μέσα σε αυτό το παράθυρο υπάρχουν προπαρασκευασμένα Elements όπου προσφέρει το UI Toolkit που είναι τα βασικά Elements που χρησιμοποιούνται σε ένα UI. Περιέχει μια λίστα από Containers (Visual Element κ.α), Controls (Label, Button κ.α), Numeric Fields, Value Fields, Choice Fields, Toolbar και Inspectors αυτά μπορεί να τα χρησιμοποιήσεις όπως είναι, να τα προσαρμόσεις ή να δημιουργήσεις τα δικά σου. Για το δικό μας project η γκάμα επιλογών ήταν αρκετά ικανοποιητική αλλά αν κάποιος χρειάζεται πιο επαγγελματική χρήση ίσως να μην τον καλύψουν μόνο αυτές οι επιλογές. Όταν επιλέγουμε κάποιο στοιχείο από την βιβλιοθήκη μας όπως αναφέραμε παραπάνω και την σύρουμε στο παράθυρο Hierarchy εμφανίζονται στο παράθυρο Inspector (4) όλες τις παραμέτρους που μπορούμε να αλλάξουμε και να επεξεργαστούμε σε αυτό το στοιχείο και στην εμφάνιση του στο Viewport. Ένα ακόμα σημαντικό παράθυρο είναι αυτό που βρίσκεται πάνω από το Hierarchy και ονομάζεται StyleSheets με αυτό δίνουμε μορφή σε όλο το περίγραμμα του UI με ήδη υπάρχοντα USS (User Style Sheets) ή δημιουργώντας νέα.



Εικόνα 5.4: Όψη του περιβάλλοντος εργασίας της μηχανής Unity

Για να έχουμε όμως μια πιο ολοκληρωμένη ιδέα τι ακριβώς είναι το UI Toolkit και πως λειτουργεί πρέπει να το χωρίσουμε σε τρία βασικές ενότητες. Το πρώτο μέρος αφορά τη σχεδίαση του περιγράμματος, το δεύτερο μέρος αφορά την μορφοποίηση (το styling) και το τρίτο μέρος την λειτουργικότητα. Όπως αναφέρθηκε και στην αρχή του κεφαλαίου αυτός ο διαχωρισμός και η συνοχή θυμίζουν τον τρόπο που είναι βασισμένο οι εφαρμογές Web, δηλαδή HTML, CSS και JavaScript αντίστοιχα. Μέσα από το κώδικα UXML μπορούμε να “αποκωδικοποιήσουμε” γνωρίσματα και ιδιότητες ή και περαιτέρω πληροφορίες σχετικά με την θέση θέσει μέσα από το Hierarchy υπό την κατάταξη parent-child και το Inspector. Βεβαίως δεν είναι αναγκαίο να γνωρίζουμε το προγραμματισμό UXML για να μπορούμε να δουλέψουμε πάνω στο UI Toolkit, απλώς ότι βήμα πραγματοποιούμε στο UI Builder καταχωρείτε και αποθηκεύετε και σε μορφή UXML. Επίσης, δημιουργώντας νέο USS file μπορούμε να πραγματοποιούμε ακόμα περισσότερη και πιο στοχευμένη επεξεργασία. Εφαρμόζοντας κάποια Visual Styles καθορίζει ένα σύνολο κανόνων όπου εμείς έχουμε επιλέξει και μας εξοικονομεί χρόνο και κόυραση από το να το εφαρμόζαμε για το κάθε Element ξεχωριστά. Δηλαδή δεν χρειάζεται να προσδιορίζεις για το κάθε ένα ξεχωριστά για παράδειγμα μήκος, πλάτος, χρώμα αλλά τα δηλώνεις ξεχωριστά ως class αυτά τα χαρακτηριστικά. Και έπειτα ορίζεις το class σε ποια Elements το επιθυμείς.



Εικόνα 5.5: Αναπαράσταση βημάτων σχεδίασης σε UI Builder

Έπειτα από το καθαρά σχεδιαστικό κομμάτι, το κομμάτι της λειτουργικότητας είναι αυτό που ακολουθεί μέσω C# Scripts. Είναι τα ίδια τύπου scripts που χρησιμοποιούνται για όλα τα παιχνίδια που δημιουργούνται στο Unity, με την προσθήκη της παραμέτρου “using UnityEngine.UIElements” για να μπορεί να γίνει σύνδεση με την λειτουργικότητα του UI Toolkit. Το επόμενο που ακολουθεί εφόσον έχουμε εξασφαλίσει την διάταξη σχεδίου μέσω UXML file, στυλ μέσω USS file και την λειτουργικότητα μέσω C# Script, είναι πως θα συνδεθούν όλα με το πραγματικό παιχνίδι που εμφανίζεται στο Unity. Για να συμβεί αυτό πρέπει να δημιουργηθεί στο Hierachy του Unity ένα UIDocument που θα συνδέσει το UI που έχουμε δημιουργήσει με τι εμφανίζεται στο Game με την προσθήκη του UXML file και του C# Script στο Inspector. Με αυτόν τον τρόπο διαχωρισμού σχεδίασης υπάρχει μεγαλύτερη διευκόλυνση ειδικότερα σε εργασίες και παιχνίδια πιο μεγάλα και πολύπλοκα.

6. Ανάπτυξη εφαρμογής

Μετά το στάδιο της σχεδίασης, ακολουθεί η ανάπτυξη της εφαρμογής. Για την δημιουργία παιχνιδιού (Game Development) ουσιαστικά υπάρχουν τέσσερις παράγοντες που πρέπει να λάβουμε υπόψιν.

1. Σκηνές και κάμερα: Είναι ουσιαστικά πως παίζεις το παιχνίδι σε 2D ή 3D και πως είναι ο χώρος του παιχνιδιού. Η δική μας εφαρμογή όπως είναι προφανές είναι 2D (δυσδιάστατο). Με την κάμερα, ειδικότερα στα 3D παιχνίδια μπορούμε να έχουμε διάφορα εφέ. Φυσικά ανάλογα με το εφέ δίνουμε και το ανάλογο ύφος. Εμείς μέσα από την Main Camera έχουμε ένα σχετικά φωτεινό εφέ που δείχνει αρκετά ξεκάθαρα την σχεδίαση.

2. Στοιχεία (Assets): Σχεδόν κάθε στοιχείο που υπάρχει στο χώρο του παιχνιδιού, όλα τα γραφικά στοιχεία όπως εικόνες, υφές (textures), 3D models, animations , αρχεία ήχου (audio files), ακόμα και Packages (δηλαδή ένα σύνολο αυτών) αυτά και άλλα πολλά είναι κάποια από τα πολλά είδη assets που μπορούμε να βρούμε στο Asset Store- <https://assetstore.unity.com/> του Unity με πληρωμή και δωρεάν ή να δημιουργήσουμε μόνοι μας. Για την ανάγκη εμπλουτισμού του Quiz, έχουμε επιλέξει δύο αρχεία ήχου από το Asset Store, μέσα από ένα σύνολο αρχείων που κατεβάσαμε που ονομάζεται CasualGameSounds, για να καλύπτει την ανάγκη ανατροφοδότησης λάθος απάντησης και σωστή απάντησης με το αντίστοιχο άκουσμα. Το πλεονέκτημα να παρέχονται έτοιμα κάποια assets είναι ο χρόνος που γλυτώνεις για να τα δημιουργήσεις μόνος σου, ειδικά αν δεν έχεις και την κατάλληλη εξειδίκευση.

3. Κώδικας προγραμματισμού (Code): Με τον κώδικα ουσιαστικά δίνουμε κίνηση και νόημα στα αντικείμενα μέσω εντολών. Άρα θα μπορούσαμε να πούμε ότι είναι ο εγκέφαλος του παιχνιδιού. Κάθε κίνηση μας στο UI φέρνει την αντίστοιχη ανάδραση που είναι προγραμματισμένη.

4. Σύστημα: Ορίζει για ποιο λειτουργικό σύστημα έχει δημιουργηθεί το παιχνίδι (π.χ για κονσόλα ή Η/Υ ή κινητό τηλέφωνο). Η εφαρμογή του Quiz πάνω στις λογικές πύλες δημιουργήθηκε για Windows, Mac, Linux.

Στην αρχή αναφέρεται η παραγωγή περιεχομένου του λογισμικού του Quiz και κατόπιν στην παραγωγή κώδικα. Συνολικά η εφαρμογή αποτελείται από τέσσερις οθόνες οι οποίες αναλύονται και παρακάτω.

6.1 Μενού

Η πρώτη οθόνη την οποία βλέπει ο παίχτης όταν εισέλθει στην εφαρμογή είναι το Μενού. Για την υλοποίηση του, δημιουργήσαμε τη σκηνή MenuScene στην οποία έχουμε ένα Background και μερικά κουμπιά τα οποία μορφοποιήθηκαν κατάλληλα για να ταιριάζουν με το ύφος του υπόλοιπου Quiz. Έτσι με την επιλογή ΞΕΚΙΝΑ, ο χρήστης μεταφέρεται απευθείας στο κύριο QUIZ ΣΤΙΣ

ΛΟΓΙΚΕΣ ΠΥΛΕΣ με την αλλαγή και μεταφορά σε ένα άλλο scene, με την επιλογή ΕΞΟΔΟΣ αποχωρεί από την εφαρμογή και με την επιλογή EXTRA μεταφέρεται σε μια άλλη σελίδα με άλλες δύο επιλογές. Όλη αυτή την διαχείριση την έχουμε επιτύχει μέσω του MainMenu script που ορίζει τη συμπεριφορά του κάθε κουμπιού. (Βλ.ΠΑΡΑΡΤΗΜΑ Α.1)



Εικόνα 6.1: Οπτική της σκηνής του MenuScene

6.2.1 Σκηνή Quiz και UI Builder

Όταν ο χρήστης μεταφερθεί στην σκηνή με το Quiz παρατηρούμε κάποια από τα βασικά στοιχεία το οποίο συνθέτουν την τελική εικόνα. Έχουμε στο επάνω μέρος της εικόνας την επικεφαλίδα η οποία μας αναφέρει πάνω σε ποιο θέμα θα είναι οι ερωτήσεις που θα ακολουθήσουν. Όπως φαίνεται και στην παρακάτω εικόνα παρατηρούμε το πάνω επίπεδο της εικόνας του Quiz μας, το οποίο εμφανίζεται και στο Hierarchy σε σχέση child όπως έχουμε αναφέρει και αναλύσει παραπάνω του #MainContainer και το έχουμε ονομάσει #UpperContainer. Μέσα σε αυτό υπάρχει πάλι ως child το Visual Element #SeperatorLeft, #SeperatorRight και #Label. Το στοιχείο #Label περιέχει και την επικεφαλίδα του Quiz. Τα υπόλοιπα δύο #SeperatorLeft και #SeperatorRight είναι για το διαχωρισμό της αριστερής και την δεξιάς πλευράς.



Εικόνα 6.2: UI Builder #Label view

Ακριβώς από κάτω είναι το μέρος που εμφανίζονται οι ερωτήσεις με την σειρά που εμείς τις έχουμε ορίσει από το πρώτη έως την όγδοη ερώτηση. Στο Hierachy αυτό το έχουμε ορίσει ως #MiddleContainer επίσης child του #MainContainer που βρίσκεται στο μεσαίο επίπεδο και έχει ως child ένα Visual Element. Το Visual Element έχει και αυτό δύο child που είναι ο #QuestionCounter και #AnswerIndicator τα οποία το ένα είναι το μέρος που εμφανίζει σε ποια ερώτηση είμαστε και το άλλο εμφανίζει αν η απάντηση που δώσαμε αν είναι σωστή ή λάθος.



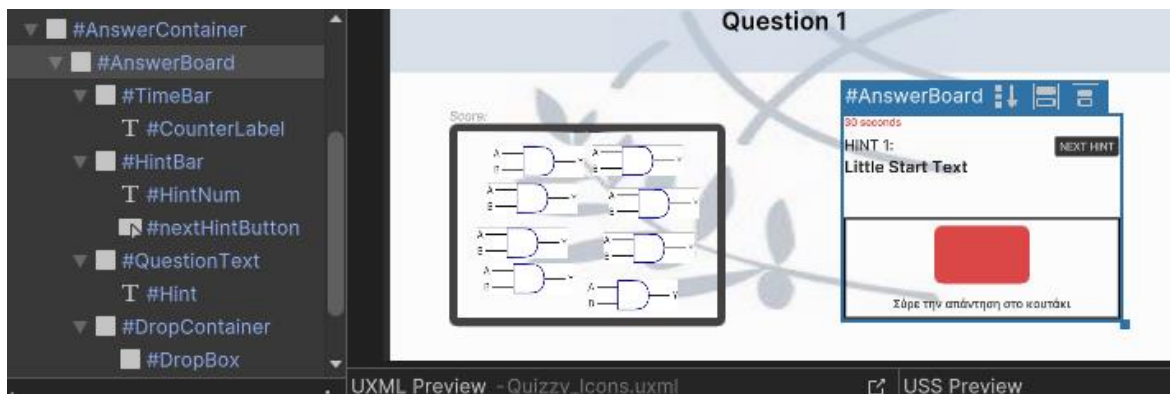
Εικόνα 6.3: UI Builder #QuestionCounter view

Στο σημείο όπως που υπάρχει περισσότερη διαδραστικότητα στο Quiz είναι το μέρος που βρίσκεται στο #LowerContainer και τα έχουμε ονομάσει #IconContainer και #AnswerContainer και αποτελούν πάλι την αντίστοιχη σχέση parent-child που αναφέραμε και παραπάνω. Σε αυτό το σημείο έχουμε δημιουργήσει από την αριστερή πλευρά ένα τετράγωνο χώρο που βρίσκονται όλα τα εικονίδια που απαντούν στις ερωτήσεις που θα θέσουμε. Για αυτό το παιχνίδι, τα εικονίδια είναι όλες οι βασικές πύλες δηλαδή η AND, NAND, NOT, NOR, OR, XOR, XNOR και buffer. Τα εικονίδια είναι μέσα από εικόνες που βρήκαμε στην αναζήτηση του Google και μετά τα επεξεργαστήκαμε μέσω του προγράμματος Canva. Το Canva - <https://www.canva.com/photo-editor/> είναι αρκετά εύχρηστο πρόγραμμα γιατί είναι δωρεάν (έχει και Premium έκδοση) και ταυτόχρονα το χειρίζεσαι μέσω διαδικτύου πολύ εύκολα και γρήγορα. Πιο αναλυτικά, το #IconContainer έχει ως child τα εξής Visual Elements: #Score και #IconBoard. Το IconBoard ως parent είναι ουσιαστικά ο χώρος που περιέχει όλα τα εικονίδια που αναφέραμε σε μορφή Visual Element. Ακριβώς πάνω από το τετράγωνο χώρο υπάρχει το #Score. Το Score είναι ένα σημαντικό χαρακτηριστικό που πρέπει να περιέχει κάθε Quiz. Στην δική μας περίπτωση για κάθε σωστή απάντηση ο παίκτης κερδίζει +10 πόντους που εμφανίζονται δίπλα στο “Score:” και -5 αν απαντήσει λάθος.



Εικόνα 6.4: UI Builder #Score view

Αντίστοιχα στην δεξιά πλευρά έχουμε τον parent #AnswerContainer που μετά διακλαδώνονται πολλαπλά child. Πρώτο το #AnswerBoard που όπως φαίνεται με ευκρίνεια και στην εικόνα είναι το κομμάτι που περιέχει πολύ βασικά στοιχεία του Quiz. Πρώτο και κυριότερο #TimeBar και #CounterLabel είναι τα σημεία που έχουμε σχεδιάσει στο UI Builder για το χρόνο. Από την στιγμή που ο παίκτης πατήσει το πλήκτρο ΞΕΚΙΝΑ ενεργοποιεί και το χρονόμετρο του τριάντα δευτερολέπτων που μετράει αντίστροφα. Αυτός είναι ο χρόνος που δίνεται για να απαντήσει, με άλλα λόγια να σύρει το σωστό εικονίδιο στο κουτάκι. Αν αυτό δεν συμβεί, δηλαδή αν ο παίκτης δεν ενεργοποιηθεί κατάλληλα, ο χρόνος λήγει και συνεχίζει στην επόμενη ερώτηση με την ίδια διάρκεια χρόνου των τριάντα δευτερολέπτων για να απαντήσει. Ακριβώς από κάτω στο ίδιο επίπεδο βρίσκεται το #HintBar και έχει ως child το #HintNum που είναι ένα στοιχείο text και το #nextHintButton που είναι ένα στοιχείο button δηλαδή κουμπί. Χρησιμοποιώντας ένα στοιχείο έτοιμο όπως το Button από τα Controls του Library παίρνουμε ακριβώς αυτό που χρειαζόμαστε για το UI μας. Στην προκειμένη περίπτωση αυτό το κουμπί μας πάει στο επόμενο Hint, δηλαδή αν χρειάζεται ο παίκτης και διατίθεται κάποια ακόμη επιπρόσθετη διευκρίνιση για την απάντηση. Στην συνέχεια, κάτω από το #HintBar υπάρχει το #QuestionText και το child #Hint, δηλαδή σε αυτό τό μέρος του UI που εμφανίζεται η ερώτηση μας. Οι ερωτήσεις που έχουμε θέσει θα το αναλύσουμε και παρακάτω πως τις έχουμε δημιουργήσει και ενσωματώσει με το UI Builder.

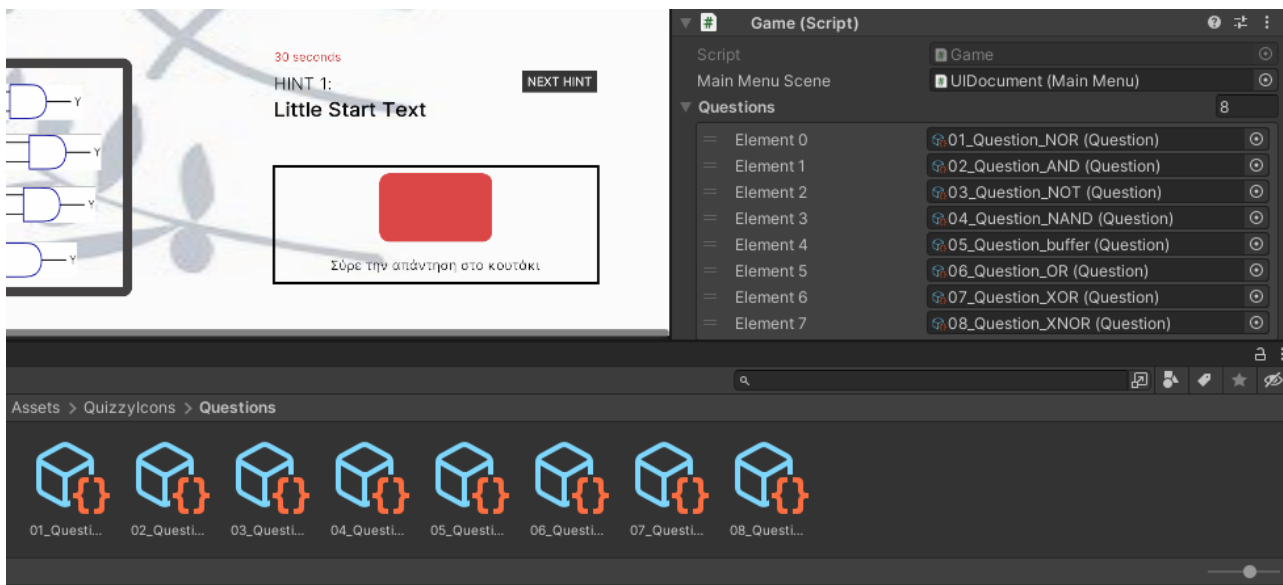


Εικόνα 6.5: UI Builder #AnswerBoard view

Τέλος, ως τελευταίο child του #AnswerBoard έχουμε το #DropContainer που είναι η τετράγωνο που έχουμε οριοθετήσει και μέσα σε αυτό υπάρχει το κόκκινο κουτάκι που έχει θέσει πάλι ως child και το έχουμε ονομάσει #DropBox. Κάτω από το κουτάκι υπάρχει σαν μορφή λεζάντας και η οδηγία που αναφέρει το να σύρεις την απάντηση στο σημείο που ορίσαμε.

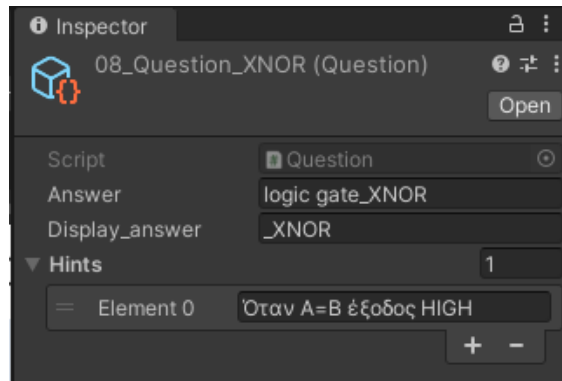
6.2.2 Δημιουργία Ερωτήσεων

Δεδομένου ότι έχουμε ολοκληρώσει την διαδικασία της σχεδίασης του Quiz μας, σε αυτό το σημείο θα εστιάσουμε στις ερωτήσεις που δημιουργήθηκαν. Αρχικά το Hierachy που αφορά το Game (όχι το Hierachy του UI Builder) έχει το UIDocument Visual Element το οποίο δημιουργήθηκε με σκοπό να δώσουμε λογική σε όλο αυτό το UI. Τα αντίστοιχα scripts Game, Main Menu, Controller, UI είναι αυτά που με τις εντολές τους βλέπουμε τα αντίστοιχα αποτελέσματα από την στιγμή που θα τα περάσουμε στο UIDocument. Στην παρακάτω εικόνα φαίνονται οι ερωτήσεις που έχουμε δημιουργήσει πατώντας δεξί κλικ και τις έχουμε περάσει ως Elements κάτω από τα Questions στην κατηγορία Game όπως φαίνεται και στην εικόνα παρακάτω.



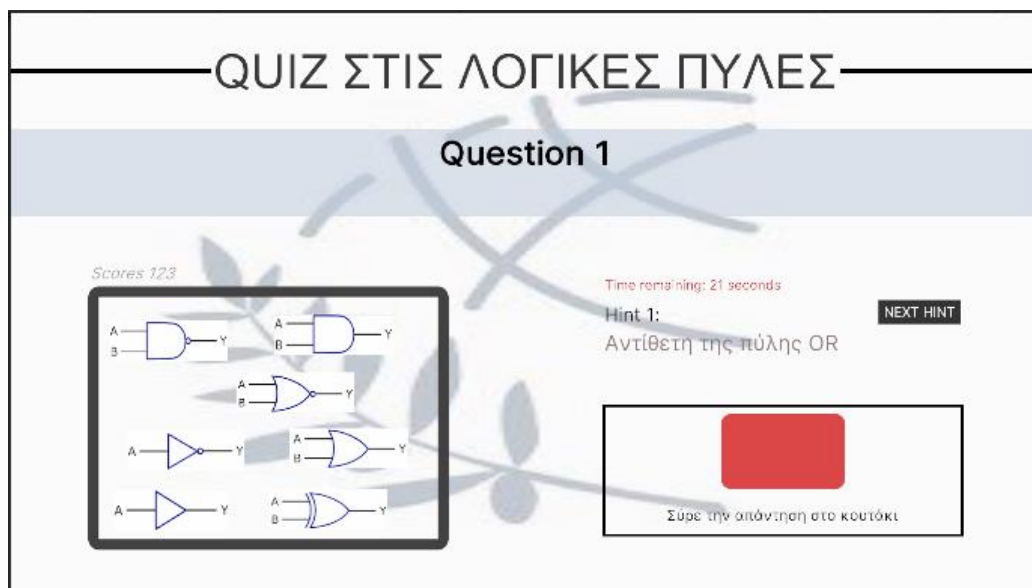
Εικόνα 6.6: Δημιουργία Questions

Κάνοντας διπλό click στις ερωτήσεις βλέπουμε να εμφανίζεται στο Hierachy τι έχουμε ορίσει σαν απάντηση Answer, Display_answer πως θέλουμε να φαίνεται δηλαδή στο console κατά τη διάρκεια του παιχνιδιού και Hints, οι ερωτήσεις δηλαδή που μπορεί να είναι μία ή και παραπάνω. Ως απάντηση δίνουμε το όνομα του εικονιδίου που έχουμε ορίσει ότι είναι απάντηση στην ερώτηση. Κάθε φορά δηλαδή που εμφανίζεται η ερώτηση στο παιχνίδι το σύστημα περιμένει από ένα σύνολο εικονιδίων να δεκτή αυτό με το σωστό όνομα, ώστε να θεωρήσει την απάντηση σωστή.



Εικόνα 6.7: Εμφάνιση Question στο Inspector

Για να δημιουργηθούν οι ερωτήσεις και να ενσωματώσουμε σε κάθε στοιχείο την “συμπεριφορά” που θέλουμε έπρεπε να δώσουμε τις ανάλογες οδηγίες και στον κώδικα. Πιο συγκεκριμένα, το script IconDragger.cs δίνει λειτουργικότητα στα Visual Elements που αναφέραμε παραπάνω dragArea, IconContainer και dropZone και πως να συμπεριφέρονται ανάλογα με το κουμπί του ποντικιού MouseEvent. Το Question.cs εμφανίζει το Hint που έχουμε ορίσει στον Inspector και φροντίζει όταν το εικονίδιο δεν φτάσει στο κουτάκι να επιστρέφει πίσω στην αρχική του θέση. Το UI.cs ορίζει όλα τα χρήσιμα Elements που έχουμε στο Hierachy ορίζει στο να εμφανίζεται το Score και η να δίνεται η σωστή ανατροφοδότηση σύμφωνα με την απάντηση. (Βλ.ΠΑΡΑΡΤΗΜΑ Α. 3,4,5,6,7)



Εικόνα 6.8: Οπτική της σκηνής SampleScene

6.2.3 Extra Panel

Το Extra Panel εξυπηρετεί στο να προσφέρει κάποιες επιπρόσθετες επιλογές στο μενού. Η μια επιλογή που προσφέρει είναι η μετάβαση σε ένα άλλο scene στο #NewScene που είναι ένα ακόμη Quiz κλασσικής μορφής με απαντήσεις πολλαπλής επιλογής. Η άλλη επιλογή είναι το κουτάκι που γράφει

βίντεο και σε μεταφέρει [Videos - Σχεδίαση Ψηφιακών Συστημάτων - Google Drive](#) όπου βρίσκονται βίντεο με παραδείγματα και προβλήματα πάνω στην ύλη του μαθήματος που έχει επιμεληθεί ο εισηγητής Μπάμνιος Γ. Η σύνδεση μεταξύ του URL του Google Drive και του Unity έγινε μέσω του script Hyperlinks (Βλ. ΠΑΡΑΡΤΗΜΑ Α. 2)



Εικόνα 6.9: Οπτική του Extra Panel

Videos - Σχεδίαση Ψη...

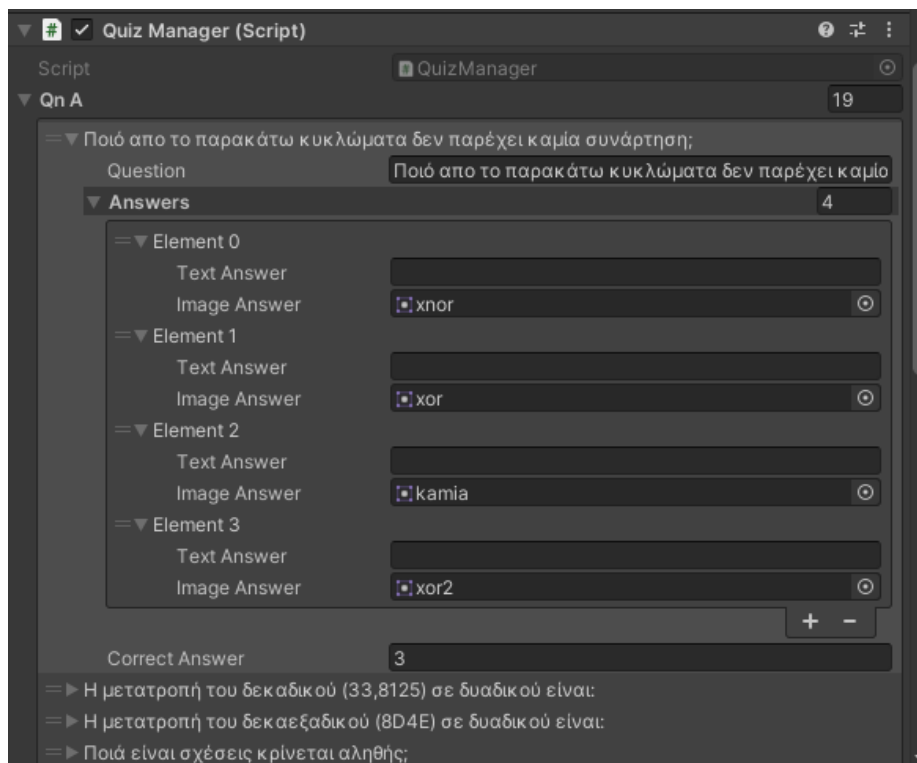
Όνομα	↑
Κεφ. 1 Προβλήματα.mp4	🔊
Κεφ. 3 Προβλήματα.mp4	🔊
Κεφ. 5_1 Προβλήματα Boole και De Morgan.mp4	🔊
Κεφ. 5_1. Προβλήματα Boole και De Morgan part 1.mp4	🔊
Κεφ. 5_1. Προβλήματα Boole και De Morgan part 2.mp4	🔊
Κεφ. 5_2 Παραδείγματα Χάρτες Καρνώ.mp4	🔊
Κεφ. 5_3 Ασκήσεις Χάρτες Καρνώ.mp4	🔊
Κεφ. 6 Προβλήματα XOR, XNOR.mp4	🔊
Κεφ. 7 Προβλήματα.mp4	🔊

Εικόνα 6.10: Βίντεο μαθήματος στο Google Drive

6.2.4 Quiz πολλαπλής επιλογής

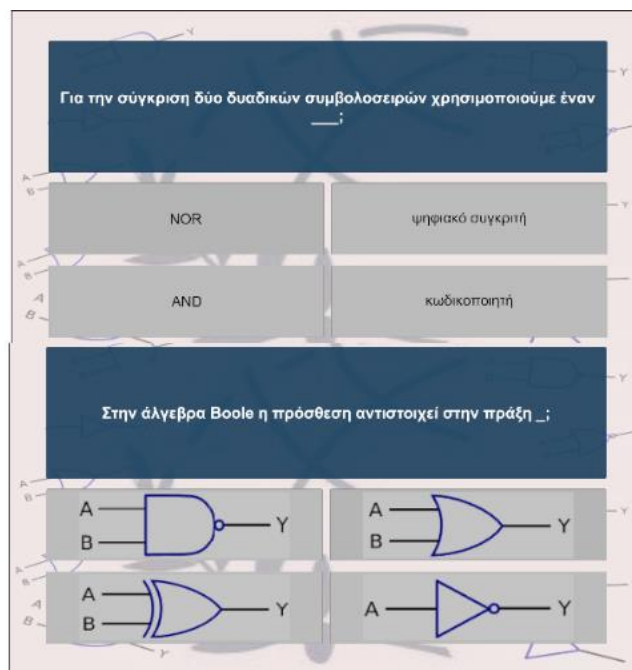
Το Quiz πολλαπλής επιλογής ξεκινάει από την στιγμή που ανοίξει το New Scene στον παίκτη. Η διαφορά με το άλλο Quiz είναι ότι μειώνεται κατά μια έννοια η διαδραστικότητα στο UI καθώς ο παίκτης έχει να απαντήσει ανάμεσα σε δύο ή τέσσερις πιθανές απαντήσεις. Η βασική αιτία που έκρινε απαραίτητη την δημιουργία ενός έξτρα κουίζ ήταν πως θέλαμε να καλύψουμε μεγαλύτερο εύρος της ύλης του μαθήματος. Για την ακρίβεια μέσα από τα κεφάλαια: 1. Συστήματα Αρίθμησης και Κώδικες,

3. Βασικές Λογικές Πύλες, 5. Άλγεβρα Boole, 6. Πύλες XOR και XNOR, 7.Αριθμητικές Πράξεις και Κυκλώματα , 8. Μετατροπές Κώδικα, Πολύ-Αποπολυπλέκτες που βρίσκονται ανεβασμένα στην σελίδα του τμήματος <https://exams-ieee.the.ihu.gr/course/view.php?id=30> και έχουν πρόσβαση όλοι οι εγγεγραμμένοι φοιτητές. Οι είκοσι ερωτήσεις εμπεριέχουν τουλάχιστον δύο ερωτήσεις από κάθε κεφάλαιο και είναι κατά κύριο λόγο θεωρητικές διότι δεν θέλαμε να βάζουμε τον παίκτη στην διαδικασία σχεδίασης. Επίσης, σε αυτό το Quiz δεν υπάρχει η παράμετρος του χρόνου. Με την ολοκλήρωση όλων των ερωτήσεων εμφανίζεται ένα Panel με το τελικό σκορ που έχει συλλέξει ο παίκτης και ένα κουμπί που σε επιστρέφει στο αρχικό μενού. Επιπρόσθετα, η άμεση διαφορά του συγκεκριμένου Quiz με το Quiz πάνω στις λογικές πύλες είναι πως δεν έγινε χρήση του UI Toolkit αλλά αναπτύχθηκε πλήρως μέσα από το Unity. Δημιουργήθηκε το QuizManager στο Hierachy όπου εμφανίζει το QnA που είναι η λίστα με τις ερωτήσεις-απαντήσεις στον Inspector που θέλουμε να δημιουργήσουμε, όπως φαίνεται και στην εικόνα από κάτω. Στις απαντήσεις μας δίνεται η δυνατότητα να είναι ή κείμενο ή εικόνα δηλαδή όπως ορίζεται Element 0,1,2,3 και συνεχίζει Text Answer, Image Answer. Για να δημιουργηθεί όμως ένα κριτήριο για το τι θα εμφανίζει έχουμε ορίσει μέσα από τον κώδικα τις εξής παραμέτρους:



Εικόνα 6.11: Εμφάνιση Quiz Manager στον Inspector

Άρα δεν έχουμε κάποιο περιορισμό σε αυτό, αλλά η ερώτηση είναι πάντα σε μορφή κειμένου. (Βλ.ΠΑΡΑΡΤΗΜΑ Α 9,10,11)



Εικόνα 6.12: Ερωτήσεις πολλαπλής επιλογής με απάντηση κείμενο και εικόνα

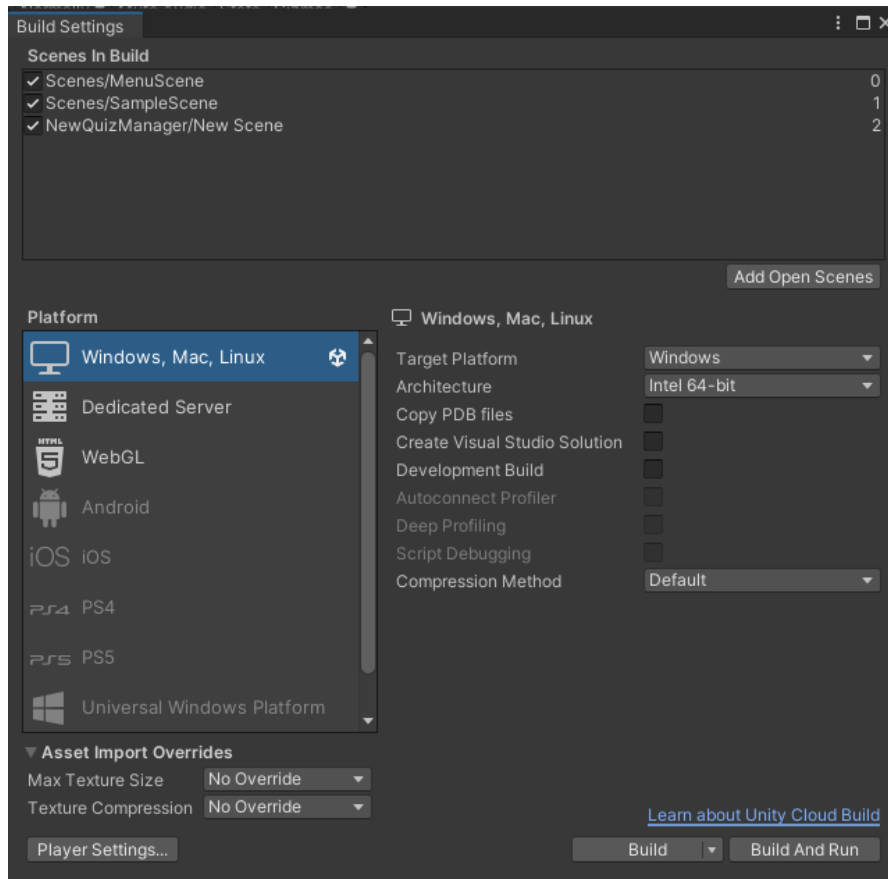
6.3 Ολοκλήρωση της Εφαρμογής

Το τελικό στάδιο της κάθε εφαρμογής είναι η εξαγωγή της από το Unity μέσω του Built Settings. Για να πραγματοποιηθεί αυτό επιλέγουμε ποιες σκηνές θα περιλαμβάνονται, σε ποια πλατφόρμα θέλουμε να είναι συμβατή, όπως αναφέραμε και παραπάνω το Unity παρέχει αρκετές επιλογές. Εμείς έχουμε επιλέξει τα Windows ως πλατφόρμα και συγκεκριμένα Intel 64-bit. Όταν επιθυμούμε να προωθήσουμε την εργασία ή να την κοινοποιήσουμε και σε άλλους μελλοντικούς χρήστες, χρειάζεται να συμπίεσουμε τα αρχεία `paixnidi_Data`, `MonoBleedingEdge`, `UnityPlayer.dll` και `paixnidi.apk` όπως φαίνεται στην Εικόνα 28.

ογιστής > Επιφάνεια εργασίας > THESIS > Build

Όνομα	Ημερομηνία τροποποι...	Τύπος	Μέγεθος
paixnidi_BurstDebugInformation_DoNot...	9/9/2023 11:42 μμ	Φάκελος αρχείων	
paixnidi_Data	6/9/2023 12:51 μμ	Φάκελος αρχείων	
10.09.2023_kaoura	10/9/2023 12:34 πμ	Συμπίεσμένος φά...	124.576 KB
10.9.23_kaoura	10/9/2023 12:32 πμ	Αρχείο 23_KAOURA	41.843 KB
paixnidi	6/9/2023 12:51 μμ	Εφαρμογή	652 KB
THESIS.data	6/9/2023 1:01 μμ	Συμπίεσμένος φά...	138.753 KB
UnityCrashHandler64	6/9/2023 12:51 μμ	Εφαρμογή	1.115 KB
UnityPlayer.dll	6/9/2023 12:51 μμ	Επέκταση εφαρμ...	28.576 KB

Εικόνα 6.13: Φάκελος αποθήκευσης



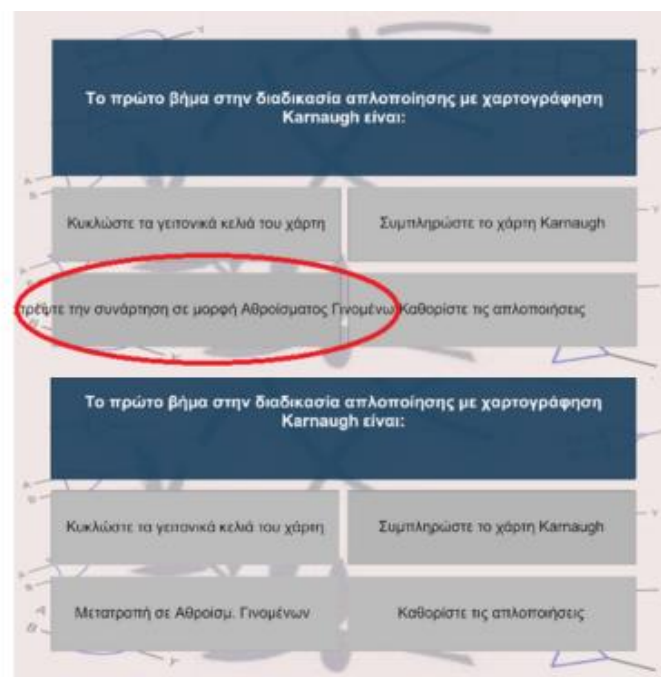
Εικόνα 6.14: Εξαγωγή εφαρμογής από το Unity

7. Παρατηρήσεις και Βελτιώσεις

Στο πλαίσιο ελέγχου λειτουργικότητας, ζητήθηκε από συναδέρφους φοιτητές και απόφοιτους του προγράμματος σπουδών Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε να συμμετέχουν στην διαδικασία όπου προέκυψαν 5 εθελοντές. Για την υλοποίηση μεταφορτώθηκε το συμπιεσμένο αρχείο που αναφέραμε παραπάνω στο Google Drive και στάλθηκε μέσω ηλεκτρονικού ταχυδρομείου. Οι συμμετέχοντες αφού κάνανε εξαγωγή όλων των αρχείων ήταν σε θέση να παίξουν το παιχνίδι και να εντοπίσουν bugs (προβληματικά σημεία) ή/και να προτείνουν βελτιώσεις, προσθήκες ή αλλαγές.

Μέσα από αυτή τη διαδικασία προέκυψαν κάποιες προτάσεις, η οποίες όσες ήταν υλοποιήσιμες άμεσα και στα όρια της παρούσης δεδομένης εργασίας ολοκληρώθηκαν. Η συνεισφορά τους υπήρξε σημαντική, καθώς βοήθησαν να γίνει ακόμα πιο λειτουργικό το παιχνίδι, κάποιες από τις προτάσεις ήταν:

- Στο κουίζ πολλαπλής επιλογής έχουμε την παρατήρηση ότι σε μια ερώτηση στο κουτάκι της απάντησης κάτω αριστερά το κείμενο βγαίνει εκτός περιθωρίου.



Εικόνα 7.1: Διόρθωση Εμφάνισης

Στους εθελοντές παίκτες έγιναν τρεις βασικές ερωτήσεις για το παιχνίδι :

- Πιστεύεις πως το παιχνίδι πέρα από εκπαιδευτικό είναι και διασκεδαστικό; ΝΑΙ ή ΟΧΙ.

Η απάντηση ήταν στο σύνολο θετική. Οι παίκτες συμφώνησαν πως σαν παιχνίδι ως ένα σημείο είναι ευχάριστο και έχει εκπαιδευτικό χαρακτήρα για κάποιον που θέλει να φρεσκάρει ή να δοκιμαστεί στις γνώσεις του στα ψηφιακά ηλεκτρονικά.

- Πιστεύεις ότι πρέπει να δοθεί μια ειδική θέση στην μάθηση μέσω παιχνιδιού και ενσωματωθεί στο μάθημα που αφορά τα ψηφιακά ηλεκτρονικά; ΝΑΙ ή ΟΧΙ.

Η απάντηση ήταν στο σύνολο θετική.

- Μπορούμε να επιτύχουμε την ίδια μάθηση μέσα από το παιχνίδι και τον κλασικό τρόπο διδασκαλίας; ΝΑΙ ή ΟΧΙ;

Σε αυτή την ερώτηση, οι απαντήσεις δίστανται καθώς 2/3 απάντησαν πως όχι. Το αποτέλεσμα αυτή της απάντησης μας δείχνει πως ακόμα για το κοινό, η έννοια του παιχνιδιού δεν μπορεί να συνδυαστεί με την αποτελεσματικότητα της συμβατής διδασκαλίας.

Αναφορικά με την παρούσα έκδοση του εκπαιδευτικού παιχνιδιού, θα μπορούσε να υπάρχει ίσως ένα ακόμα είδος ερωτήσεων το οποίο να είναι της μορφής ΣΩΣΤΟ-ΛΑΘΟΣ ώστε να μην το ενσωματώνουμε με το πολλαπλής επιλογής. Παράλληλα, το σημείο που εμφανίζεται η ερώτηση θα μπορούσε να είναι εικόνα και όχι μόνο κείμενο το ερώτημα. Τέλος, κρίνεται απαραίτητο να δημοσιευτεί αυτό το παιχνίδι για να πετύχουμε περισσότερο «δειγματισμό» του παιχνιδιού στην πράξη και να συλλέξουμε δεδομένα για την πρόοδο των φοιτητών πριν και μετά την ολοκλήρωση του παιχνιδιού. Έτσι θα έχουμε ακόμα πιο ολοκληρωμένη άποψη πόσο επιτυχημένη ήταν η χρήση του και ο σκοπός της μάθησης στο συγκεκριμένο κλάδο επετεύχθη.

8. Επίλογος

Λαμβάνοντας υπόψη την όλη διαδικασία που πραγματοποιήθηκε για τη δημιουργία του εκπαιδευτικού παιχνιδιού της παρούσας πτυχιακής εργασίας πρέπει να σημειωθεί πως ήταν εξαιρετικά εποικοδομητική, λαμβάνοντας υπόψη το πλήθος γνώσεων που λήφθηκαν. Το υπάρχον κομμάτι των γνώσεων στο προγραμματισμό που έχουμε πάρει μέσα από μαθήματα κατά τη διάρκεια φοίτησης στο τμήμα εμπλουτίστηκε με μια πληθώρα νέων γνώσεων και νέων δεξιοτήτων. Ένα σημαντικό στοιχείο αποτέλεσε η διαδικασία της βιβλιογραφικής ανασκόπησης, καθώς παρείχε πρόσθετες γνώσεις και έμπνευση για πιθανές μελλοντικές καινοτομίες.

Ο στόχος αυτής της διπλωματικής εργασίας ήταν να αναπτυχθεί ένα εκπαιδευτικό παιχνίδι για τους φοιτητές και την νεαρή κοινότητα, με τη χρήση του Unity game engine σε γλώσσα προγραμματισμού C#. Μέσα από την εργασία απαντήθηκαν ερωτήματα όπως “Μπορεί ένας φοιτητής του τμήματος Ηλεκτρονικών Μηχανικών να αναπτύξει ένα ψηφιακό παιχνίδι;”. Και απάντηση είναι πως ναι παρόλο που είχε αρκετές απαιτήσεις. Παράλληλα όπως είναι και εμφανές από την εργασία για να θεωρηθεί ένα παιχνίδι ολοκληρωμένο, πρέπει να ακολουθηθούν κάποια συγκεκριμένα βήματα, όπως μια θεωρητική ανασκόπηση για το αντικείμενο και κάποιοι βασικοί ορισμοί. Έπειτα ακολουθεί η ανάλυση της ιδέας, δηλαδή το θέμα που έχουμε επιλέξει και τη σύνδεση με την εκπαίδευση ή άλλα παραδείγματα εφαρμογών με ίδιο περιεχόμενο με το δικό μας. Ακολουθεί η σχεδίαση της εφαρμογής, από την σχεδίαση στο χαρτί μέχρι τα πρότυπα υψηλής πιστότητας και μετά η ανάπτυξη μέσα από τη παιχνιδιομηχανή του Unity. Πέρα από το προγραμματιστικό κομμάτι χρειάζεται και σχεδίαση ή επεξεργασία εικόνας και προσθήκη ήχου. Και τέλος η αξιολόγηση της από την στιγμή που θα το παίζει κάποιος και θα συλλέξουμε τα πρώτα δεδομένα αξιολόγησης.

Η υποστήριξη από φόρουμ του Unity και βίντεο στο YouTube συνέλαβαν σημαντικά στην ανάπτυξη του παιχνιδιού. Κατά την ανάπτυξη, προέκυψαν αρκετά Errors στο Console του Unity με περαιτέρω προσπάθεια και έρευνα όμως το επιθυμητό αποτέλεσμα επετεύχθη. Σαν σύνολο όμως το Unity αποτελεί ένα εύχρηστο αξιόπιστο εργαλείο που σίγουρα κάποια από τι χρήσεις του θα μου φανεί χρήσιμη και στο μέλλον. Με βάση αυτή την εργασία διαπίστωσα πως τα ηλεκτρονικά παιχνίδια είναι ένα εργαλείο για τους εκπαιδευτικούς που μπορούν εύκολα να κάνουν την εκπαιδευτική εμπειρία πιο ευχάριστη.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Παντέχου Μ., Τζήκα Π. (2019). Σχεδίαση και Ανάπτυξη Ψηφιακού Περιβάλλοντος Εικονικής Περιήγησης και Αλληλεπίδρασης (Virtual Game)(Διπλωματική Εργασία), Προηγμένα Συστήματα Υπολογιστών & Επικοινωνιών, Διαπανεπιστημιακό Διατμηματικό Πρόγραμμα Μεταπτυχιακών Σπουδών
- [2] Becker, K. (2021). What's the difference between gamification, serious games, educational games, and game-based learning? Academia Letters, Article 209. Ανακτήθηκε τον Σεπτέμβριο του 2023 από <https://doi.org/10.20935/AL209>.
- [3] Mark Overmars (2007). Game Maker Tutorial: Designing Good Games
- [4] Πετσίνης Α.. (2020). Ανάπτυξη Εφαρμογής – Παιχνιδιού Μέσω του Προγράμματος Unity 3D (Πτυχιακή Εργασία), Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Πάτρα
- [5] Χατζησάββας, Μ., & Ρέππα, Ε. (2016). Η χρήση των ψηφιακών παιχνιδιών στην εκπαίδευση: Μια συστηματική ανασκόπηση. Ψηφιακά Μέσα και Παιδεία.
- [6] Johnson, S. (2005). Everything Bad Is Good for You, Computer.
- [7] Μυσιρλάκη, Σ. & Παρασκευά, Φ. (2009). Το ηλεκτρονικό - εκπαιδευτικό παιχνίδι : κίνητρα μάθησης (Μεταπτυχιακή Εργασία). Τμήμα Ψηφιακών Συστημάτων, Πανεπιστήμιο Πειραιώς.
- [8] Liu, Z.Y., Shaikh, Z. & Gazizova, F. (2020). Using the Concept of Game-Based Learning in Education. International Journal of Emerging Technologies in Learning (iJET), 15(14), 53-64. Kassel, Germany: International Journal of Emerging Technology in Learning. Ανακτήθηκε τον Σεπτέμβριο από <https://www.learntechlib.org/p/217589/>.
- [9] Alice Mitchell and Carol Savill-Smith (2004) The use of computer and video games for learning A review of the literature. Ανακτήθηκε τον Σεπτέμβριο του 2023 από https://dera.ioe.ac.uk/id/eprint/5270/7/041529_Redacted.pdf
- [10] Steinkuehler, C., & Duncan, S. (2008). Scientific habits of mind in virtual worlds. Journal of Science Education and Technology, 17(6), 530-543.
- [11] Malone, T. W. (1981). Toward a theory of intrinsically motivating instruction. Cognitive science, 5(4), 333-369.
- [12] Sawyer, B. (2003). Serious games: improving public policy through game-based learning and simulation. Washington: Woodrow Wilson International Center for Scholars.
- [13] (Abt, 1970: 9). Abt, C. C. (1970). Serious games. New York: Viking

- [14] Cheong, Christopher; Cheong, France; and Filippou, Justin, "Quick Quiz: A Gamified Approach for Enhancing Learning" (2013). PACIS 2013 Proceedings. 206. Ανακτήθηκε τον Σεπτέμβριο του 2023 από <http://aisel.aisnet.org/pacis2013/206>
- [15] Λέτσιου Ζ. (2022). Σχεδιασμός και ανάπτυξη εκπαιδευτικού παιχνιδιού για την υποστήριξη του μαθήματος πληροφορικής του Γυμνασίου με τη χρήση του εκπαιδευτικού περιβάλλοντος AppInventor (Διπλωματική Εργασία), Τμήμα Πληροφορικής, Καστοριά
- [16] Κανίδου Χ. (2019).(Διπλωματική Εργασία), Δημιουργία Διαδραστικού Εκπαιδευτικού παιχνιδιού για Ρεμπέτικο Τραγούδι. Μεταπτυχιακή Εξειδίκευση στα Πληροφοριακά Συστήματα, ΕΑΠ
- [17] Prensky, M. (2001). Digital game-based learning, McGraw-Hill.
- [19] Sommerville, I. (2011). Software Engineering. New York: Pearson Education, Inc.
- [20] N. H. Rahani, A. A. Bilong, M. R. Mat Suruji, and I. Y. Panessai, (Mar. 2020. "Learning Logic Gates Using Augmented Reality", International Journal of Multimedia and Recent Innovation, vol. 2, no. 1, pp. 26-44.
- [21] D Dewantara *, M Wati, M Misbah, S Mahtari, and S Haryandi (2020), The Effectiveness of Game Based Learning on The Logic Gate Topics, Physics Education, Faculty of Teacher Training and Education, Universitas Lambung Mangkurat, Indonesia
- [22] Hanasrullah Halim, Wan Amirah Najwa Wan Idris, Haslina Hassan, Ismail Yusuf Panessai. Learning Logic Gate through 7-Gates. International Journal of Multimedia and Recent Innovation, vol. 2, no. 1, pp. 1-10, March 2020. DOI: 10.36079/lamintang.ijmari-0201.70
- [23] Κετσιακίδης, Γ. Α. (2017). Εφαρμογές επαυξημένης πραγματικότητας και η εκπαιδευτική διαδικασία (Μεταπτυχιακή Διατριβή). Αριστοτέλειο Πανεπιστήμιο, Θεσσαλονίκη. Ανακτήθηκε τον Σεπτέμβριο του 2023 από <http://ikee.lib.auth.gr/record/292830>
- [24] Κουτσαμπάσης, Π. (2015). Αξιολόγηση διαδραστικών συστημάτων με επίκεντρο τον χρήστη. Αθήνα: Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών. Ανακτήθηκε τον Σεπτέμβριο του 2023 από <http://hdl.handle.net/11419/2765>
- [25] Δημούλας Χ. (2017), "Εργαστήρι Πολυμεσικής Παραγωγής", διδακτικές σημειώσεις - διαφάνειες διαλέξεων, ΔΔΠΜΣ Προηγμένα Συστήματα Υπολογιστών και Επικοινωνιών, ΑΠΘ.

ΙΣΤΟΣΕΛΙΔΕΣ

[18] 16 Types of Quizzes Explained. Ανακτήθηκε τον Σεπτέμβριο του 2023 <https://www.opinionstage.com/quiz/types-of-quizzes/>

[26] Βικιπαίδεια, την δωρεάν εγκυκλοπαίδεια. (13 Ιουλίου 2023). Ανακτήθηκε τον Σεπτέμβριο του 2023 [https://en.wikipedia.org/wiki/Unity_\(game_engine\)](https://en.wikipedia.org/wiki/Unity_(game_engine))

[27] The web and web standards. (21 Αυγούστου 2023). Ανακτήθηκε τον Σεπτέμβριο του 2023 https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/The_web_and_web_standards

[28] Unity Documentation, Comparison of UI systems in Unity. Ανακτήθηκε τον Σεπτέμβριο του 2023 <https://docs.unity3d.com/Manual/UI-system-compare.html>

ΠΑΡΑΡΤΗΜΑ Α

Σε αυτό το παράρτημα θα παρουσιαστούν όλα τα Script τα οποία χρησιμοποιήθηκαν για την υλοποίηση του παιχνιδιού. Όλα τα Script δημιουργήθηκαν στο Visual Studio σε γλώσσα C#.

1.Main Menu

```
C: > Users > kleio > paixnidi > Assets > MainMenu.cs
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.SceneManagement;
5
6  public class MainMenu : MonoBehaviour
7  {
8      public void PlayGame()
9      {
10         SceneManager.LoadSceneAsync(1);
11     }
12     public void StartNewGame()
13     {
14         SceneManager.LoadScene("New Scene");
15     }
16     public void QuitGame()
17     {
18         Application.Quit();
19     }
20 }
21 }
```

2.Hyperlinks

```
C: > Users > kleio > paixnidi > Assets > Hyperlinks.cs
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class Hyperlinks : MonoBehaviour
6  {
7      public void OpenURL(string link)
8      {
9         Application.OpenURL(link);
10     }
11 }
```

3.Setup

```
C:\Users > kleio > paixnidi > Assets > Setup.cs
1  using System.Collections;
2  using UnityEngine;
3  using UnityEngine.UIElements;
4  using System.Collections.Generic;
5
6  public class Setup
7  {
8      public static void InitializeDragDrop(VisualElement root, Controller controller)
9      {
10         root.Query<VisualElement>("IconBoard")
11             .Children<VisualElement>()
12             .ForEach((elem) =>
13             {
14                 elem.AddManipulator(new IconDragger(root, controller));
15             });
16     }
17 }
18
19 public static void InitializeIcons(VisualElement root, List<Question> questions)
20 {
21     int currentIconIndex = 0;
22
23     foreach (Question question in questions)
24     {
25         VisualElement questionIcon = root.Query<VisualElement>("IconBoard").Children<VisualElement>().AtIndex(currentIconIndex);
26         questionIcon.style.backgroundImage = Resources.Load<Texture2D>("img/" + question.answer);
27         questionIcon.userData = question;
28
29         currentIconIndex++;
30     }
31 }
```

4.Question

```
C:\Users > kleio > paixnidi > Assets > Question.cs
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  [CreateAssetMenu]
6  public class Question : ScriptableObject
7  {
8      public string answer;
9      public string display_answer;
10
11     public string[] hints = new string[3];
12
13     public string[] GetHints()
14     {
15         if (hints.Length == 0)
16         {
17             Debug.LogError("hints not inized");
18         }
19         return hints;
20     }
21 }
```

5.IconDragger

```
C: > Users > kleio > paixnidi > Assets > IconDragger.cs
1  using UnityEngine;
2  using UnityEngine.UIElements;
3
4  public class IconDragger : MouseManipulator
5  {
6      Controller controller;
7
8      private Vector2 startPos;
9      private Vector2 elemStartPosGlobal;
10     private Vector2 elemStartPosLocal;
11
12     VisualElement dragArea;
13     VisualElement iconContainer;
14     VisualElement dropZone;
15
16     bool isActive;
17
18     public IconDragger(VisualElement root, Controller controller)
19     {
20         this.controller = controller;
21
22         dragArea = root.Q("DragArea");
23         dropZone = root.Q("DropBox");
24
25         isActive = false;
26     }
27
28     protected override void RegisterCallbacksOnTarget()
29     {
30         target.RegisterCallback<MouseDownEvent>(OnMouseDown);
31         target.RegisterCallback<MouseMoveEvent>(OnMouseMove);
```

```

32     target.RegisterCallback<MouseEvent>(OnMouseUp);
33 }
34
35 protected override void UnregisterCallbacksFromTarget()
36 {
37     target.UnregisterCallback<MouseDownEvent>(OnMouseDown);
38     target.UnregisterCallback<MouseMoveEvent>(OnMouseMove);
39     target.UnregisterCallback<MouseEvent>(OnMouseUp);
40 }
41
42 protected void OnMouseDown (MouseDownEvent e)
43 {
44     iconContainer = target.parent;
45     startPos = e.localMousePosition;
46
47     elemStartPosGlobal = target.worldBound.position;
48     elemStartPosLocal = target.layout.position;
49
50     dragArea.style.display = DisplayStyle.Flex;
51     dragArea.Add(target);
52
53     target.style.top = elemStartPosGlobal.y;
54     target.style.left = elemStartPosGlobal.x;
55
56     isActive = true;
57     target.CaptureMouse();
58     e.StopPropagation();
59 }
60

```

```

61 protected void OnMouseMove(MouseMoveEvent e)
62 {
63     if (!isActive || !target.HasMouseCapture())
64         return;
65
66     Vector2 diff = e.localMousePosition - startPos;
67
68     target.style.top = target.layout.y + diff.y;
69     target.style.left = target.layout.x + diff.x;
70
71     e.StopPropagation();
72 }
73 protected void OnMouseUp(MouseEvent e)
74 {
75     if (!isActive || !target.HasMouseCapture())
76         return;
77
78     if (target.worldBound.Overlaps(dropZone.worldBound))
79     {
80         dropZone.Add(target);
81
82         target.style.top = dropZone.contentRect.center.y - target.layout.height / 2;
83         target.style.left = dropZone.contentRect.center.x - target.layout.width / 2;
84
85         Debug.Log("The provided answer is: " + ((Question)target.userData).display_answer);
86
87         controller.CheckAnswer(((Question)target.userData).answer);
88
89     } else
90     {
91         iconContainer.Add(target);

```

```
92
93     target.style.top = elemStartPosLocal.y - iconContainer.contentRect.position.y;
94     target.style.left = elemStartPosLocal.x - iconContainer.contentRect.position.x;
95 }
96
97     isActive = false;
98     target.ReleaseMouse();
99     e.StopPropagation();
100
101     dragArea.style.display = DisplayStyle.None;
102 }
103 }
```

6.UI

```

C: > Users > kleio > paixnidi > Assets > UI.cs
1  using UnityEngine;
2  using UnityEngine.UIElements;
3  using System.Collections;
4
5  public class UI : MonoBehaviour
6  {
7      public Controller controller;
8
9
10     VisualElement root;
11     Label hint;
12     Label hintNumLabel;
13     Label questionNumLabel;
14     Label timeLabel;
15     Label answer_indicator;
16     Label highscoreLabel;
17     Label currentscoreLabel;
18     Button nextHintButton;
19
20     private void OnEnable()
21     {
22     {
23         root = GetComponent<UIDocument>().rootVisualElement;
24
25         hint = root.Q<Label>("Hint");
26         hintNumLabel = root.Q<Label>("HintNum");
27         questionNumLabel = root.Q<Label>("QuestionCounter");
28         nextHintButton = root.Q<Button>("nextHintButton");
29         timeLabel = root.Q<Label>("CounterLabel");
30         answer_indicator = root.Q<Label>("AnswerIndicator");
31         highscoreLabel = root.Q<Label>("Highscore");

```

```

32         currentscoreLabel = root.Q<Label>("Myscore");
33
34         Initialize();
35     }
36     public void UpdateScoreLabel(int newScore)
37     {
38         Debug.Log("Updating score label with score: " + newScore);
39         currentscoreLabel.text = "Score: " + newScore;
40     }
41
42     public void Initialize()
43     {
44         nextHintButton.clicked += () =>
45         {
46             controller.HandleWrongAnswer();
47         };
48
49         Setup.InitializeDragDrop(root, controller);
50         Setup.InitializeIcons(root, controller.getAllQuestions());
51     }
52
53     public void GiveAnswerFeedback(bool correct)
54     {
55         answer_indicator.style.visibility = Visibility.Visible;
56         answer_indicator.text = correct ? "Your answer was correct" : "Your answer was wrong";
57
58         StyleColor colorCorrect = new StyleColor(new Color32(0, 132, 19, 255));
59         StyleColor colorWrong = new StyleColor(new Color32(132, 0, 19, 255));
60

```

```

61     answer_indicator.style.color = correct ? colorCorrect : colorWrong;
62     StartCoroutine(CleanUpQuestion());
63 }
64
65 IEnumerator CleanUpQuestion()
66 {
67     yield return new WaitForSeconds(3);
68     answer_indicator.style.visibility = Visibility.Hidden;
69
70     VisualElement dropZone = root.Q<VisualElement>("DropBox");
71
72     if (dropZone.childCount > 0)
73     {
74         dropZone.RemoveAt(0);
75     }
76 }
77
78 public void SetTimer(string seconds)
79 {
80     timeLabel.text = "Time remaining: " + seconds + " seconds";
81 }
82
83 public void SetHint(string hintText)
84 {
85     hint.text = hintText;
86 }
87
88 public void SetHintNumber(int hintNum)
89 {

```

```

90     hintNumLabel.text = "Hint " + hintNum.ToString() + ":";
91 }
92
93 public void SetQuestionNumber(int questionNum)
94 {
95     questionNumLabel.text = "Question " + questionNum.ToString();
96 }
97 }

```

7.Game

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Game : MonoBehaviour
6 {
7     public MainMenu mainMenuScene;
8     public List<Question> questions = new List<Question>();
9     Question currentQuestion;
10    int questionIndex = 0;
11
12    string currentHint;
13    int hintIndex = 0;
14    public void InitializeGame()
15    {
16        currentQuestion = questions[questionIndex];
17        currentHint = currentQuestion.GetHints()[hintIndex];
18    }
19
20    public bool IsAnswerCorrect(string answer)
21    {
22        return currentQuestion.answer == answer;
23    }
24
25    public void HandleCorrectAnswer()
26    {
27        NextQuestion();
28    }
29
30    public void HandleWrongAnswer()
31    {

```

```

32        if (hintIndex < currentQuestion.GetHints().Length - 1)
33        {
34            hintIndex++;
35            currentHint = currentQuestion.GetHints()[hintIndex];
36        }
37        else
38        {
39            NextQuestion();
40        }
41
42    }
43    public void NextQuestion()
44    {
45        if (questionIndex < questions.Count - 1)
46        {
47            currentQuestion = questions[++questionIndex];
48            hintIndex = 0;
49            currentHint = currentQuestion.GetHints()[hintIndex];
50        }
51        else
52        {
53            mainMenuScene.ReturnToMainMenu();
54        }
55    }
56    public Question getCurrentQuestion()
57    {
58        return currentQuestion;
59    }
60

```

```

61     public int getCurrentQuestionNum()
62     {
63         return questionIndex + 1;
64     }
65
66     public string getCurrentHint()
67     {
68         return currentHint;
69     }
70     public int getCurrentHintNum()
71     {
72         return hintIndex + 1;
73     }
74 }

```

8.Controller

```

C:\> Users > kleio > paixnidi > Assets > UI.cs
1  using UnityEngine;
2  using UnityEngine.UIElements;
3  using System.Collections;
4
5  public class UI : MonoBehaviour
6  {
7      public Controller controller;
8
9
10     VisualElement root;
11     Label hint;
12     Label hintNumLabel;
13     Label questionNumLabel;
14     Label timeLabel;
15     Label answer_indicator;
16     Label highscoreLabel;
17     Label currentscoreLabel;
18     Button nextHintButton;
19
20     private void OnEnable()
21     {
22     {
23         root = GetComponent<UIDocument>().rootVisualElement;
24
25         hint = root.Q<Label>("Hint");
26         hintNumLabel = root.Q<Label>("HintNum");
27         questionNumLabel = root.Q<Label>("QuestionCounter");
28         nextHintButton = root.Q<Button>("nextHintButton");
29         timeLabel = root.Q<Label>("CounterLabel");
30         answer_indicator = root.Q<Label>("AnswerIndicator");
31         highscoreLabel = root.Q<Label>("Highscore");

```

```

32     public void HandleCorrectAnswer()
33     {
34         game.HandleCorrectAnswer();
35         UpdateUI();
36         ResetCounter();
37         this.UpdateScore(true);
38     }
39
40     [SerializeField] private AudioSource correctSoundEffect;
41     [SerializeField] private AudioSource errorSoundEffect;
42
43     public void CheckAnswer(string answer)
44     {
45         bool answerCorrect = game.IsAnswerCorrect(answer);
46
47         if (answerCorrect)
48         {
49             HandleCorrectAnswer();
50             correctSoundEffect.Play();
51             Debug.Log("Answer was correct");
52         }
53         else
54         {
55
56             errorSoundEffect.Play();
57             Debug.Log("Answer was wrong");
58             HandleWrongAnswer();
59         }
60     }

```

```

61
62         ui.GiveAnswerFeedback(answerCorrect);
63     }
64     public void UpdateScore(bool correctAnswer)
65     {
66         Debug.Log("Updating score with correct answer: " + correctAnswer);
67         score += correctAnswer ? 10 : -5;
68         ui.UpdateScoreLabel(score);
69     }
70
71     private int currentCounter;
72
73     public int CurrentCounter
74     {
75         get { return currentCounter; }
76         set
77         {
78             if (value == 0)
79             {
80                 HandleWrongAnswer();
81                 return;
82             }
83             ui.SetTimer(value.ToString());
84             currentCounter = value;
85         }
86     }
87     public void UpdateUI()
88     {
89         ui.SetHint(game.GetCurrentHint());

```

```

90         ui.SetHintNumber(game.GetCurrentHintNum());
91         ui.SetQuestionNumber(game.GetCurrentQuestionNum());
92     }
93
94     public List<Question> GetAllQuestions()
95     {
96         return game.questions;
97     }
98
99     void ResetCounter()
100    {
101        CurrentCounter = 30;
102    }
103
104    IEnumerator UpdateCounter()
105    {
106        while (CurrentCounter > 0)
107        {
108            yield return new WaitForSeconds(1);
109            CurrentCounter--;
110            ui.SetTimer(CurrentCounter.ToString());
111        }
112
113        HandleWrongAnswer();
114    }
115 }

```

9.QuizManager

```

C: > Users > kleio > paixnidi > Assets > NewQuizManager > QuizManager.cs
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.UI;
5  using UnityEngine.SceneManagement;
6
7  public class QuizManager : MonoBehaviour
8  {
9      public List<QuestionandAnswers> QnA;
10     public GameObject[] options;
11     public int currentQuestion;
12
13     public GameObject Quizpanel;
14     public GameObject GoPanel;
15
16     public Text QuestionTxt;
17     public Text ScoreTxt;
18
19     int totalQuestions = 0;
20     public int score;
21
22     [SerializeField] private AudioSource correctSoundEffect;
23     [SerializeField] private AudioSource errorSoundEffect;
24
25     private void Start()
26     {
27         totalQuestions = QnA.Count;
28         GoPanel.SetActive(false);
29         generateQuestion();
30     }

```

```

31     public void retry()
32     {
33         SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex);
34     }
35
36     void GameOver()
37     {
38         Quizpanel.SetActive(false);
39         GoPanel.SetActive(true);
40         ScoreTxt.text = score + "/" + totalQuestions;
41     }
42     public void correct()
43     {
44         score += 1;
45         QnA.RemoveAt(currentQuestion);
46         correctSoundEffect.Play();
47         StartCoroutine(waitForNext());
48     }
49     public void wrong()
50     {
51         QnA.RemoveAt(currentQuestion);
52         errorSoundEffect.Play();
53         StartCoroutine(waitForNext());
54     }
55     IEnumerator waitForNext()
56     {
57         yield return new WaitForSeconds(1);
58         generateQuestion();
59     }
60     void SetAnswers()

```

```

92
93     if(QnA.Count > 0)
94     {
95         currentQuestion = Random.Range(0, QnA.Count);
96
97         QuestionTxt.text = QnA[currentQuestion].Question;
98         SetAnswers();
99     }
100    else
101    {
102        Debug.Log("Out of Questions");
103        GameOver();
104    }
105
106    void ResetButtonColors()
107    {
108        foreach (GameObject button in options)
109        {
110            ColorBlock colors = button.GetComponent<Button>().colors;
111            colors.normalColor = Color.white;
112            button.GetComponent<Button>().colors = colors;
113        }
114    }
115 }
116 }

```

10. AnswerScript

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.UI;
5
6  public class AnswerScript : MonoBehaviour
7  {
8      public bool isCorrect = false;
9      public QuizManager quizManager;
10
11     public Color startColor;
12     private void Start()
13     {
14         startColor = GetComponent<Image>().color;
15     }
16
17     public void Answer()
18     {
19         if(isCorrect)
20         {
21             GetComponent<Image>().color = Color.green;
22             Debug.Log("Correct Answer");
23             quizManager.correct();
24         }
25         else
26         {
27             GetComponent<Image>().color = Color.red;
28             Debug.Log("Wrong Answer");
29             quizManager.wrong();
30         }
31         StartCoroutine(ResetColor());

```

```
32     }
33     IEnumerator ResetColor()
34     {
35         yield return new WaitForSeconds(1f);
36         GetComponent<Image>().color = startColor;
37     }
38 }
```

11.Question and Answers

```
1 using UnityEngine;
2 [System.Serializable]
3 public class QuestionandAnswers
4 {
5     public string Question;
6     public Answer[] Answers;
7     public int CorrectAnswer;
8     [System.Serializable]
9     public class Answer
10    {
11        public string TextAnswer;
12        public Sprite ImageAnswer;
13    }
14 }
```