



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΕΚΠΑΙΔΕΥΤΙΚΟ ΠΑΙΧΝΙΔΙ ΕΚΜΑΘΗΣΗΣ ΑΛΓΟΡΙΘΜΩΝ ΜΕ ΤΗΝ
ΧΡΗΣΗ ΨΗΦΙΑΚΟΥ ΒΟΗΘΟΥ**

ΔΗΜΗΤΡΙΟΣ ΓΚΟΦΑΣ

A.M. 71347123

ΕΠΙΒΛΕΠΩΝ:

ΕΠ. ΚΑΘΗΓΗΤΗΣ ΧΡΗΣΤΟΣ ΤΡΟΥΣΣΑΣ

Διπλωματική Εργασία υποβληθείσα στο Τμήμα Μηχανικών Πληροφορικής και
Υπολογιστών του Πανεπιστημίου Δυτικής Αττικής

Αθήνα, 2023

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
ΕΚΠΑΙΔΕΥΤΙΚΟ ΠΑΙΧΝΙΔΙ ΕΚΜΑΘΗΣΗΣ ΑΛΓΟΡΙΘΜΩΝ ΜΕ ΤΗΝ
ΒΟΗΘΕΙΑ ΨΗΦΙΑΚΟΥ ΒΟΗΘΟΥ

Δημήτριος Γκόφας

A.M. 71347123

Εισηγητής:

Επ. Καθηγητής Χρήστος Τρούσσας

Εξεταστική Επιτροπή:

Παναγιώτα Τσελέντη, ΕΔΙΠ

Ακριβή Κρούσκα, ΕΔΙΠ

Ημερομηνία εξέτασης: 6/10/2023

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΕΥΧΑΡΙΣΤΙΕΣ	10
ΠΕΡΙΛΗΨΗ ΕΡΓΑΣΙΑΣ	11
ABSTRACT	12
ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ	13
ΚΑΤΑΛΟΓΟΣ ΔΙΑΓΡΑΜΜΑΤΩΝ	15
ΚΕΦΑΛΑΙΟ ΠΡΟΤΩ - ΕΙΣΑΓΩΓΗ	
1.1 Η ΣΗΜΑΣΙΑ ΤΗΣ ΤΕΧΝΟΛΟΓΙΑΣ ΣΤΗΝ ΕΚΠΑΙΔΕΥΣΗ	16
1.2 Η ΣΗΜΑΣΙΑ ΤΩΝ ΨΗΦΙΑΚΩΝ ΕΚΠΑΙΔΕΥΤΙΚΩΝ ΠΑΙΧΝΙΔΙΩΝ ΚΑΙ ΤΩΝ ΕΥΦΥΩΝ ΨΗΦΙΑΚΩΝ ΠΑΙΧΝΙΔΙΩΝ ΣΕ ΨΗΦΙΑΚΑ ΠΕΡΙΒΑΛΛΟΝΤΑ ΜΑΘΗΣΗΣ	17
1.3 ΔΙΑΡΘΡΩΣΗ ΚΕΦΑΛΑΙΩΝ	18
ΚΕΦΑΛΑΙΟ ΔΕΥΤΕΡΟ - ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ ΚΑΙ ΑΝΑΣΚΟΠΗΣΗ ΒΙΒΛΙΟΓΡΑΦΙΑΣ	
2.1 ΕΙΣΑΓΩΓΗ ΣΤΟ ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ	19
2.2 ΣΤΟΧΟΙ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ ΜΕ ΜΙΑ ΜΑΤΙΑ ΣΤΗ ΛΟΓΙΚΗ ΤΟΥ	20
2.3 ΨΗΦΙΑΚΑ ΠΟΛΥΜΕΣΑ	20
2.4 ΨΗΦΙΑΚΑ ΠΑΙΧΝΙΔΙΑ	21
2.5 ΕΚΠΑΙΔΕΥΤΙΚ ΠΑΙΧΝΙΔΙΑ	22
2.6 ΕΥΦΥΗ ΕΚΠΑΙΔΕΥΤΙΚΑ ΠΑΙΧΝΙΔΙΑ ΜΕ ΧΡΗΣΗ ΨΗΦΙΑΚΟΥ ΒΟΗΘΟΥ	23
2.7 ΤΙ ΕΙΝΑΙ ΕΝΑΣ ΨΗΦΙΑΚΟΣ ΒΟΗΘΟΣ ΚΑΙ ΟΙ ΠΙΟ ΣΗΜΑΝΤΙΚΕΣ ΚΑΤΗΓΟΡΙΕΣ	24
2.8 ΤΙ ΕΙΝΑΙ ΕΝΑ ΜΟΝΤΕΛΟ ΠΡΟΚΑΘΟΡΙΣΜΕΝΗΣ ΓΝΩΣΗΣ	24
2.9 ΤΙ ΕΙΝΑΙ Η ΤΕΧΝΟΛΟΓΙΑ NLP	25
2.10 ΟΡΙΣΜΟΣ ΑΛΓΟΡΙΘΜΩΝ ΚΑΙ ΑΝΑΣΚΟΠΗΣΗ ΒΑΣΙΚΩΝ ΑΛΓΟΡΙΘΜΩΝ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ	25
2.10.1 ΠΑΡΑΔΕΙΓΜΑ ΑΡΙΘΜΗΤΙΚΟΥ ΑΛΓΟΡΙΘΜΟΥ	25

2.10.2 ΠΑΡΑΔΕΙΓΜΑ ΑΛΓΟΡΙΘΜΟΥ ΜΙΝ / ΜΑΧ	26
2.10.3 ΠΑΡΑΔΕΙΓΜΑ ΑΛΓΟΡΙΘΜΟΥ ΤΑΞΙΝΟΜΗΣΗΣ	26
2.10.4 ΠΑΡΑΔΕΙΓΜΑ ΑΛΓΟΡΙΘΜΟΥ ΕΠΕΞΕΡΓΑΣΙΑΣ ΣΥΜΒΟΛΟΣΕΙΡΑΣ	26
2.11 ΤΙ ΕΙΝΑΙ ΕΝΑ ΠΑΙΧΝΙΔΙ ΠΕΡΙΠΕΤΕΙΑΣ	26
2.12 ΠΑΙΧΝΙΔΙ ΠΑΖΛ	27
2.13 ΚΑΜΕΡΑ ΣΕ ΤΡΙΤΟ ΠΡΟΣΩΠΟ	27
2.14 ΑΝΑΣΚΟΠΗΣΗ ΒΙΒΛΙΟΓΡΑΦΙΑΣ ΓΙΑ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΠΑΙΧΝΙΔΙΟΥ ΜΕ ΤΗΝ ΧΡΗΣΗ ΨΗΦΙΑΚΟΥ ΒΟΗΘΟΥ	28
2.15 ΤΑ ΠΑΙΧΝΙΔΙΑ ΑΛΓΟΡΙΘΜΩΝ	29
2.16 ΤΟ NLP ΩΣ ΠΗΓΗ ΓΝΩΣΗΣ ΤΗΝ ΣΗΜΕΡΙΝΗ ΕΠΟΧΗ	30
ΚΕΦΑΛΑΙΟ ΤΡΙΤΟ - ΜΕΘΟΔΟΛΟΓΙΑ	
3.1 ΕΙΣΑΓΩΓΗ ΜΕΘΟΔΟΛΟΓΙΑΣ	33
3.2 ΣΧΕΔΙΑΣΜΟΣ ΠΑΙΧΝΙΔΟΥ ΚΑΙ ΚΑΤΑΣΚΕΥΗ ΤΟΥ	33
3.3 ΣΥΝΟΨΗ ΜΕΘΟΔΟΛΟΓΙΑΣ ΚΑΙ ΑΞΙΟΛΟΓΙΣΗΣ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ	35
3.4 ΣΥΝΟΨΗ ΚΕΦΑΛΑΙΟΥ	35
ΚΕΦΑΛΑΙΟ ΤΕΤΑΡΤΟ - ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΚΑΙ ΕΠΙΣΚΟΠΗΣΗ	
4.1 ΕΙΣΑΓΩΓΗ	36
4.2 ΙΣΤΟΡΙΑ	36
4.3 ΛΟΓΙΚΗ	37
4.4 ΕΙΚΟΝΑ	38
4.5 ΠΡΟΕΡΓΑΣΙΑ ΚΑΙ ΔΟΜΙΚΑ ΣΤΟΙΧΕΙΑ ΠΟΛΥΜΕΣΩΝ	39
4.5.1 ΚΕΙΜΕΝΟ	39
4.5.2 ΕΙΚΟΝΑ	40
4.5.3 ΗΧΟΣ	40
4.5.4 ΚΙΝΟΥΜΕΝΗ ΕΙΚΟΝΑ (ANIMATION)	41
4.5.5 ΑΛΛΗΛΕΠΙΔΡΑΣΤΙΚΟΤΗΤΑ (INTERACTIVITY)	41
4.5.6 ΤΡΟΠΟΙ ΑΠΟΘΗΚΕΥΣΗΣ ΣΤΟ ΠΑΙΧΝΙΔΙ	42

4.5.7 ΤΙ ΕΙΝΑΙ ΕΝΑ MODULE ΣΤΗΝ PYTHON	43
4.5.8 ΤΙ ΕΙΝΑΙ ΤΟ PYGAME MODULE	43
4.5.8.1 ΒΑΣΙΚΗ ΔΟΜΗ ΠΑΡΑΘΥΡΟΥ ΣΤΟ PYGAME	43
4.5.8.2 EVENTS ΣΤΟ PYGAME	45
4.5.8.3 ΕΠΙΦΑΝΙΕΣ ΣΤΟ PYGAME	45
4.5.8.4 ΚΕΙΜΕΝΟ ΣΤΟ PYGAME	47
4.5.8.5 ΗΧΟΙ ΣΤΟ PYGAME	47
4.6 ΤΟ ΚΥΡΙΩΣ ΜΕΝΟΥ	48
4.6.1 ΕΜΦΑΝΙΣΗ	49
4.6.2 ΛΕΙΤΟΥΡΓΙΑ	49
4.6.3 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ	50
4.7 ΕΜΦΑΝΙΣΗ ΤΟΥ ΜΕΝΟΥ ΕΠΙΠΕΔΩΝ	51
4.7.1 ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΜΕΝΟΥ ΕΠΙΠΕΔΩΝ	51
4.7.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΜΕΝΟΥ ΕΠΙΠΕΔΩΝ	51
4.8 ΕΜΦΑΝΙΣΗ ΤΟΥ ΜΕΝΟΥ ΕΠΙΛΟΓΗΣ ΠΛΗΚΤΡΩΝ	52
4.8.1 ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΜΕΝΟΥ ΕΠΙΛΟΓΗΣ ΠΛΗΚΤΡΩΝ	53
4.8.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΜΕΝΟΥ ΕΠΙΛΟΓΗΣ ΠΛΗΚΤΩΝ	53
4.9 ΕΜΦΑΝΙΣΗ ΤΟΥ ΜΕΝΟΥ ΕΝΤΑΣΗΣ ΤΟΥ ΗΧΟΥ	54
4.9.1 ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΜΕΝΟΥ ΕΝΤΑΣΗΣ ΤΟΥ ΗΧΟΥ	54
4.9.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΜΕΝΟΥ ΕΝΤΑΣΗΣ ΤΟΥ ΗΧΟΥ	55
4.10 ΕΜΦΑΝΙΣΗ ΤΟΥ ΜΕΝΟΥ ΒΟΗΘΕΙΑΣ (HELP MENU)	56
4.11 ΤΟ ΕΠΙΠΕΔΟ	57
4.11.1 ΕΜΦΑΝΙΣΗ	58
4.11.2 ΛΕΙΤΟΥΡΓΙΑ	58
4.11.3 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ	59
4.12 ΕΜΦΑΝΙΣΗ ΤΟΥ ΠΑΙΚΤΗ (ΜΑΓΟΥ)	61
4.12.1 ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΠΑΙΚΤΗ (ΜΑΓΟΥ)	61
4.12.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΠΑΙΚΤΗ (ΜΑΓΟΥ)	62

4.13 ΕΜΦΑΝΙΣΗ ΤΗΣ ΦΩΤΙΑΣ	67
4.13.1 ΛΕΙΤΟΥΡΓΙΑ ΤΗΣ ΦΩΤΙΑΣ	68
4.13.1 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΗΣ ΦΩΤΙΑΣ	68
4.14 ΕΜΦΑΝΙΣΗ ΤΩΝ ΤΕΡΑΤΩΝ	70
4.14.1 ΛΕΙΤΟΥΡΓΙΑ ΤΩΝ ΤΕΡΑΤΩΝ	70
4.14.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΩΝ ΤΕΡΑΤΩΝ	70
4.15 ΕΜΦΑΝΙΣΗ ΤΟΥ ΜΑΓΙΚΟΥ ΚΛΕΙΔΙΟΥ	77
4.15.1 ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΜΑΓΙΚΟΥ ΚΛΕΙΔΙΟΥ	77
4.15.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΜΑΓΙΚΟΥ ΚΛΕΙΔΙΟΥ	77
4.16 ΕΜΦΑΝΙΣΗ ΤΩΝ ΚΕΡΜΑΤΩΝ	78
4.16.1 ΛΕΙΤΟΥΡΓΙΑ ΤΩΝ ΚΕΡΜΑΤΩΝ	78
4.16.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΩΝ ΚΕΡΜΑΤΩΝ	78
4.17 ΕΜΦΑΝΙΣΗ ΤΩΝ ΚΟΥΤΙΩΝ ΜΕ ΤΙΣ ΠΛΗΡΟΦΟΡΙΕΣ	79
4.17.1 ΛΕΙΤΟΥΡΓΙΑ ΤΩΝ ΚΟΥΤΙΩΝ ΜΕ ΤΙΣ ΠΛΗΡΟΦΟΡΙΕΣ	79
4.17.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΩΝ ΚΟΥΤΙΩΝ ΜΕ ΤΙΣ ΠΛΗΡΟΦΟΡΙΕΣ	79
4.18 ΕΜΦΑΝΙΣΗ ΤΗΣ ΤΕΛΙΚΗΣ ΠΥΛΗΣ	80
4.18.1 ΛΕΙΤΟΥΡΓΙΑ ΤΗΣ ΤΕΛΙΚΗΣ ΠΥΛΗΣ	81
4.18.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΗΣ ΤΕΛΙΚΗΣ ΠΥΛΗΣ	81
4.19 ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΜΑΓΙΚΟΥ ΒΙΒΛΙΟΥ ΣΥΝΟΠΤΙΚΑ	85
4.20 ΛΕΙΤΟΥΡΓΙΑ ΜΗΧΑΝΗΣ ΓΡΑΦΙΚΩΝ ΣΥΝΟΠΤΙΚΑ	86
4.21 ΛΕΙΤΟΥΡΓΙΑ ΤΗΣ ΜΗΧΑΝΗΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΚΩΔΙΚΑ ΣΥΝΟΠΤΙΚΑ	87
4.22 ΛΕΙΤΟΥΡΓΙΑ ΤΗΣ ΑΠΟΘΗΚΕΥΣΗΣ ΠΑΙΚΤΩΝ ΣΥΝΟΠΤΙΚΑ	87

ΚΕΦΑΛΑΙΟ ΠΕΜΠΤΟ - ΑΞΙΟΛΟΓΙΣΗ

5.1 ΕΙΣΑΓΩΓΗ	88
5.2 ΟΙ ΕΡΩΤΗΣΕΙΣ ΑΞΙΟΛΟΓΗΣΗΣ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ ΑΠΟ ΤΟΥΣ ΠΑΙΚΤΕΣ ΜΑΘΗΤΕΣ	88

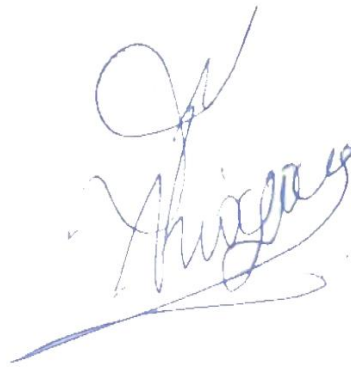
5.3 ΑΠΟΤΕΛΕΣΜΑΤΑ ΑΞΙΟΛΟΓΗΣΗΣ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ ΑΠΟ ΤΟΥΣ ΠΑΙΚΤΕΣ (ΜΑΘΗΤΕΣ)	89
5.4 ΤΕΛΙΚΟ ΠΟΡΙΣΜΑ ΑΞΙΟΛΟΓΗΣΗΣ	101
ΚΕΦΑΛΑΙΟ ΕΚΤΟ - ΣΥΜΠΕΡΑΣΜΑΤΑ	
6.1 ΕΙΣΑΓΩΓΗ ΣΥΜΠΕΡΑΣΜΑΤΩΝ	102
6.2 ΣΥΜΠΕΡΑΣΜΑΤΑ ΒΙΒΛΙΟΓΡΑΦΙΑΣ ΚΑΙ ΘΕΩΡΗΤΙΚΟΥ ΥΠΟΒΑΘΡΟΥ	102
6.3 ΣΥΜΠΕΡΑΣΜΑΤΑ ΑΞΙΟΛΟΓΙΣΗΣ	103
6.4 ΠΡΟΤΑΣΗ ΓΙΑ ΠΕΡΑΙΤΕΡΩ ΕΡΑΥΝΑ	103
ΒΙΒΛΙΟΓΡΑΦΙΚΕΣ ΑΝΑΦΟΡΕΣ	104
ΕΛΛΗΝΙΚΗ ΒΙΒΛΙΟΓΡΑΦΙΑ	104
ΞΕΝΟΓΛΩΣΣΗ ΒΙΒΛΙΟΓΡΑΦΙΑ	104
ΗΛΕΚΤΡΟΝΙΚΕΣ ΠΗΓΕΣ	105
ΠΑΡΑΡΤΗΜΑ	106

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΡΟΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ο/η κάτωθι υπογεγραμμένος Δημήτριος Γκόφας του Χρήστου, με αριθμό μητρώου 71347123 φοιτητής του Τμήματος Μηχανικών Πληροφορικής και Υπολογιστών της Σχολής Μηχανικών του Πανεπιστημίου Δυτικής Αττικής, δηλώνω ότι:

«Βεβαιώνω ότι είμαι συγγραφέας αυτής της Διπλωματικής εργασίας και κάθε βοήθεια την οποία είχα για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος. Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου».

Ο Δηλών:

A handwritten signature in blue ink, appearing to read 'Dimitrios Gkoffas', with a long horizontal flourish underneath.

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου τον κύριο Τρούσσα Χρήστο για την καθοδήγηση που μου παρείχε, για τον χρόνο που αφιέρωσε και για την συνεργασία κατά τη διεκπεραίωση της Διπλωματικής μου εργασίας.

Επίσης, θα ήθελα να ευχαριστήσω την οικογένειά μου.

ΠΕΡΙΛΗΨΗ ΕΡΓΑΣΙΑΣ

Η παρούσα εργασία έχει ως στόχο την δημιουργία ενός εκπαιδευτικού ψηφιακού 2D παιχνιδιού εκμάθησης βασικών αλγορίθμων με την χρήση ψηφιακού βοηθού (chatbot). Αναλύονται ευρύτερα οι έννοιες των εκπαιδευτικών παιχνιδιών, των αλγορίθμων, των παιχνιδιών περιπέτειας και παζλ καθώς και της χρήσης τεχνητής νοημοσύνης για την βελτίωση της εκπαιδευτικής διαδικασίας μέσω παιχνιδιών. Το εν λόγω παιχνίδι σχεδιάστηκε και δημιουργήθηκε μέσω της PyGame της Python. Η ιστορία του παιχνιδιού διαδραματίζεται γύρω από έναν χαρακτήρα ενός εκπαιδευόμενου (wizard) ο οποίος πρέπει να μάθει και τα τέσσερα ξόρκια αποφεύγοντας τα τέρατα που τον κυνηγούν για να γίνει ένας ολοκληρωμένος μάγος. Και τα τέσσερα ξόρκια περιέχουν έναν βασικό αλγόριθμο της Python, τον οποίο μαθαίνει ο παίκτης απομνημονεύοντας την λειτουργία του και χρησιμοποιώντας τον σε μορφή ξορκιού για να περάσει την πίστα ώστε να πάει στην επόμενη μέχρι να καταφέρει να τελειώσει το παιχνίδι. Συνεπώς το παιχνίδι είναι ένα ψηφιακό παιχνίδι εκμάθησης αλγορίθμων για την επίλυση προκλήσεων, που πληρεί τους παράγοντες της ψυχαγωγίας, αλληλεπιδραστικότητας και εκπαίδευσης. Διεγείρει την σκέψη των μαθητών στους οποίους και απευθύνεται, ψυχαγωγώντας τους παράλληλα και χρησιμοποιώντας την κριτική τους ικανότητα για την επίλυση των παζλ. Ύστερα από την δημιουργία του το παιχνίδι αξιολογήθηκε από ένα μικρό δείγμα μαθητών. Τα πορίσματα της αξιολόγησης έδειξαν ότι το παιχνίδι τους άφησε μία ευχάριστη εντύπωση καθώς χαρακτηρίστηκε ως διασκεδαστικό και ως εξαιρετικά βοηθητικό στην κατανόηση των αλγορίθμων ειδικά μέσω της χρήσης του ψηφιακού βοηθού (chatbot) που απαντούσε στις απορίες τους και τους διευκόλυνε στον τερματισμό του παιχνιδιού. Εν κατακλείδι το παιχνίδι κατάφερε να επιτύχει τους βασικούς του στόχους: την εκμάθηση των αλγορίθμων αλλά και να καταστήσει την εκπαιδευτική διαδικασία πιο βιωματική και ψυχαγωγική για τους μαθητές.

Λέξεις Κλειδιά: Αλγόριθμος, εκπαιδευτικό ψηφιακό παιχνίδι, ψηφιακός βοηθός(chatbot), Python, PyGame.

ABSTRACT

The present work aims to create an educational digital 2D game for learning basic algorithms using a digital assistant (chatbot). It broadly analyzes the concepts of educational games, algorithms, adventure games, and puzzles, as well as the use of artificial intelligence to enhance the educational process through games. The game in question was designed and developed using Python's PyGame framework. The game's storyline revolves around a character of a learner (wizard) who must learn four spells while avoiding monsters pursuing him to become a complete wizard. Each of the four spells contains a basic Python algorithm that the player learns by memorizing its function and using it in spell form to pass the level and progress through the game. Therefore, the game is a digital algorithmic learning game that meets the factors of entertainment, interactivity, and education. It stimulates the thinking of the targeted students, entertaining them while also using their critical thinking skills to solve puzzles. After the game's creation, it was evaluated by a small sample of students. The evaluation results showed that the game left a pleasant impression on them, as it was considered fun and extremely helpful in understanding algorithms, especially through the use of the digital assistant (chatbot) that answered their questions and facilitated their progress in the game. In conclusion, the game successfully achieved its primary objectives: the learning of algorithms and making the educational process more experiential and enjoyable for the students.

Keywords: Algorithm, educational digital game, digital assistant (chatbot), Python, PyGame.

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα1: (Reasoning types)

Εικόνα2: The Legend of Zelda illustration

Εικόνα3: Main Loop in a PyGame window

Εικόνα4: Επικάλυψη επιπέδων (RGBA) εικόνων (bitmap)

Εικόνα5: Little Endian RGBA32 bit

Εικόνα6: The Legend of Zelda: Breath of the Wild (Menu)

Εικόνα7: Κυρίως μενού παιχνιδιού

Εικόνα8: Μενού επιπέδων

Εικόνα9: Υπόδειξη κλειδωμένου επιπέδου

Εικόνα10: Μενού εναλλαγής πλήκτρων για την κίνηση του παίκτη

Εικόνα11: Εναλλαγή του εικονιδίου στα πλήκτρα (W, A, S, D)

Εικόνα12: Εναλλαγή του εικονιδίου σε βέλη

Εικόνα13: Μενού έντασης του ήχου

Εικόνα14: Επιλογή έντασης ήχου με τον κέρσορα

Εικόνα15: Μενού βοήθειας του παίχτη

Εικόνα16: Δομή της κλάσης με τα κύρια αντικείμενα που περιέχει

Εικόνα17: Τυπική όψη του επιπέδου με την κάμερα σε τρίτο πρόσωπο

Εικόνα18: Δημιουργία βασικών αντικειμένων στον κατασκευαστή της κλάσης (Level)

Εικόνα19: Δημιουργία του επιπ. 1 στην μεταβλητή (selected_level)

Εικόνα20: Βασικό κομμάτι λειτουργίας της μεθόδου (run()) του αντικειμένου (Level)

Εικόνα21: Το επίπεδο τρέχει μέσα στο κυρίως μενού όσο αυτό είναι κλειστό

Εικόνα22: Όψη από τη λειτουργία των βοηθητικών αντικειμένων (assets) του μάγου

Εικόνα23: Δημιουργία βασικών αντικειμένων στον κατασκευαστή της κλάσης (Player)

Εικόνα24: Άλλα βασικά αντικείμενα στον κατασκευαστή της κλάσης (Player)

Εικόνα25: Δομή της μεθόδου (run()) για το αντικείμενο (Player)

Εικόνα26: Όψη κατεύθυνσης κίνησης του παίκτη

Εικόνα27: Μέθοδος της κίνησης με βάση τα πλήκτρα και το περιβάλλον του μάγου

Εικόνα28: Κομμάτι μεθόδου της πρόσθεσης φλόγας

Εικόνα29: Μέθοδος της απεικόνισης της ζωής, της μαγείας και του χρόνου

Εικόνα30: Μέθοδος απόκτησης κέρματος της κλάσης (Player)

Εικόνα31: Εικόνες αντικειμένων φωτιάς και οι εκρήξεις τους

Εικόνα32: Κατασκευαστής της κλάσης της φωτιάς (Fier)

Εικόνα33: Εικόνα της μεθόδου (run()) της κλάσης της φωτιάς (Fier)

Εικόνα34: Κομμάτι κώδικα της μεθόδου (run()) της κλάσης τύπου (Level)

Εικόνα35: Μέθοδος δημιουργίας τέρατος από την κλάση (Level)

Εικόνα36: Σημαντικό κομμάτι από τον κατασκευαστή της κλάσης του τέρατος

Εικόνα37: Η εκτέλεση της μεθόδου (run()) όλων των τεράτων

Εικόνα38: Μέθοδος (run()) της κλάσης του τέρατος (Monster)

Εικόνα39: Μέθοδος κνηγητού του παίκτη από το τέρας της κλάσης τύπου τέρας

Εικόνα40: Όψη μέγιστης ακτίνας ορατότητας τέρατος 8 μονάδων

Εικόνα41: Παράδειγμα αποτυχίας ορατότητας του παίκτη σε μικρή απόσταση

Εικόνα42: (Simple Ray Casting Example Image)

Εικόνα43: Μέθοδος του τέρατος (Monster) για την μείωση ζωής του παίκτη (Player)

Εικόνα44: Μέθοδος θανάτου του τέρατος της κλάσης (Monster)

Εικόνα45: Μέθοδος (give_key()) της κλάσης (Monster) για την δημιουργία του κλειδιού

Εικόνα46: Εικόνα του μαγικού κλειδιού

Εικόνα47: Μέθοδος (get_key()) της κλάσης τύπου (Player)

Εικόνα48: Μέθοδος της κλάσης (Player) που ανοίγει με την σειρά τα κουτιά

Εικόνα49: Εικόνα δημιουργία αντικειμένου τύπου (InfoBox) με σειρά ξεκλειδώματος(1)

Εικόνα50: Εικόνα πύλης κατά την διάρκεια του παιχνιδιού

Εικόνα51: Άνοιγμα του πάπυρου και έλεγχος τερματισμού του επιπέδου

Εικόνα52: Μέθοδος σύγκρισης αλγορίθμων της κλάσης (Riddle)

Εικόνα53: Τελική εικόνα με αποτυχία της εκπαίδευσης

Εικόνα54: Τελική εικόνα με επιτυχία της εκπαίδευσης

Εικόνα55: Εικόνα για συγγραφή των ζορκιών σε γλώσσα (Python)

Εικόνα56: Εικόνα βιβλίου με παράδειγμα χρήσης του (Chat_Gpt_3.5_turbo)

Εικόνα57: Αρχική συζήτηση με το (API)

Εικόνα58: Ερωτήσεις και απάντησης στο (API) του (Chat) με την βοήθεια του (openai)

ΚΑΤΑΛΟΓΟΣ ΔΙΑΓΡΑΜΜΑΤΩΝ

- Διάγραμμα1: Επάρκεια επιπέδων
- Διάγραμμα2: Μέγεθος και σχήμα επιπέδων
- Διάγραμμα3: Επάρκεια αντικειμένων αλληλεπίδρασης παίκτη
- Διάγραμμα4: Επάρκεια λειτουργιών παιχνιδιού (gameplay)
- Διάγραμμα5: Ποσότητα κουτιών και περιεχομένου τους
- Διάγραμμα6: Κατανόηση κειμένων κουτιών
- Διάγραμμα7: Κατανόηση αλγορίθμων
- Διάγραμμα8: Ευκολία κατανόησης και χρήσης της γλώσσας (Python)
- Διάγραμμα9: Επάρκεια χρόνου για την κατανόηση του αλγορίθμου
- Διάγραμμα10: Ευχρηστία του βιβλίου βοήθειας του παίκτη (μάγου)
- Διάγραμμα11: Ευκολία επικοινωνίας μέσω γραψίματος με το βιβλίο βοηθό
- Διάγραμμα12: Απαντήσεις εντός θέματος με το βιβλίο βοηθό
- Διάγραμμα13: Συζητήσεις εντός θέματος με το βιβλίο βοηθό
- Διάγραμμα14: Ταχύτητα απαντήσεων βιβλίου βοηθού
- Διάγραμμα15: Επιλογή αλλαγής παίκτη (μάγου)
- Διάγραμμα16: Ποιότητα απεικόνισης λίστας επιτυχόντων παικτών
- Διάγραμμα17: Όψη του αρχικού μενού
- Διάγραμμα18: Όψη επιμέρους μενού
- Διάγραμμα19: Ευκολία χρήσης του μενού
- Διάγραμμα20: Ποιότητα γραφικών παιχνιδιού
- Διάγραμμα21: Ανάλυση εικόνας παιχνιδιού σε (pixels)
- Διάγραμμα22: Ταχύτητα αναπαραγωγής γραφικών
- Διάγραμμα23: Επιλογή κωδικού για κάθε παίκτη
- Διάγραμμα24: Διάρκεια αλληλεπίδρασης παίκτη με το εκπαιδευτικό παιχνίδι

ΚΕΦΑΛΑΙΟ ΠΡΩΤΟ

ΕΙΣΑΓΩΓΗ

1.1 Η ΣΗΜΑΣΙΑ ΤΗΣ ΤΕΧΝΟΛΟΓΙΑΣ ΣΤΗΝ ΕΚΠΑΙΔΕΥΣΗ

Από τα πολλά επιτεύγματα της τεχνολογικής εξέλιξης ίσως το πιο καίριο να είναι η εδραίωση του ηλεκτρονικού υπολογιστή στην καθημερινότητα μας. Αυτή η ανακάλυψη του νέου εργαλείου κατέστησε αλυσιδωτές αλλαγές στην συλλογιστική πορεία και στον τρόπο διευθέτησης των προβλημάτων των ανθρώπων. Ριζικές αλλαγές σημειώθηκαν ακόμη και στον τρόπο επικοινωνίας, στην μετάδοση της γνώσης και των δεξιοτήτων αλλά και ευρύτερα σε κάθε πτυχή του κοινωνικοπολιτικού μας βίου. (“Μάθηση και διδασκαλία στην εποχή της πληροφορικής, Α.Ράπτης-Α.Ράπτη, Α τόμος 2017). Ακλόνητος δεν θα έμενε και ο τομέας της εκπαίδευσης καθώς είναι ο κατεξοχήν τομέας εκμάθησης δεξιοτήτων και γνώσεων. Κάθε μεταβολή δεδομένων στην κοινωνία θα επηρέαζε εύλογα και την διδασκαλία τους. Συνεπώς, θα μπορούσαμε να πούμε πως η εισβολή των νέων ψηφιακών μέσων στην καθημερινότητα μας επακόλουθα σήμανε την εισαγωγή των τεχνολογικών ψηφιακών μέσων και των ψηφιακών περιβαλλόντων τους στην εκπαιδευτική διαδικασία. Πιο συγκεκριμένα οι νέες τεχνολογίες στην εκπαίδευση και κυρίως των εκπαιδευτικών παιχνιδιών 2D ή 3D προσφέρουν μία πληθώρα οφελών. Αρχικά, ενισχύουν την κριτική σκέψη και αντίληψη των μαθητών, την εναλλακτική σκέψη, την ομαδικότητα ανάμεσα στους μαθητές και συνεπώς την συνεργατική μάθηση. Άλλα οφέλη μεταξύ των οποίων συμπεριλαμβάνονται η αυτορρυθμιζόμενη μάθηση, η δυνατότητα ατομικής ή συνεργατικής εργασίας και η ανάπτυξη αυτομάθησης. Οι μαθητές ασχολούνται με κομμάτια της διδακτέας ύλης πιο διαδραστικά μέσω εικόνας, ήχου και άλλων αλληλεπιδραστικών εργαλείων. Αυτόματα η εκπαιδευτική οδός της γνώσης καθίσταται πιο διασκεδαστική και πιο βιωματική με αποτέλεσμα οι μαθητές να εμπλέκονται πιο ενεργά στην μάθηση και όπως αργότερα θα δούμε με την μετέπειτα εξέλιξη των εκπαιδευτικών παιχνιδιών σε ευφυή εκπαιδευτικά παιχνίδια που χρησιμοποιούν τεχνητή νοημοσύνη όπως ψηφιακούς βοηθούς, στους μαθητές παρέχεται η βέλτιστη βοήθεια στην κατανόηση ακόμη και των πιο περίπλοκων εννοιών. Πέρα από την διδακτέα ύλη οι μαθητές μέσω της χρήσης ψηφιακών μέσων και ψηφιακών παιχνιδιών μαθαίνουν πως να διαχειρίζονται τους διαθέσιμους πόρους τους, να επικοινωνούν και να συνεργάζονται έχοντας ανοχή στην διαφορετικότητα, να αναζητά και να στοιχειοθετεί τις πληροφορίες, να συμμετέχει ενεργά σε ψηφιακά περιβάλλοντα και να χειρίζεται άριστα ψηφιακά μέσα. Το εκπαιδευτικό ψηφιακό παιχνίδι λοιπόν αποτελεί ένα εξαιρετικά παιδαγωγικό μέσο διδασκαλίας και γνωσιακής μετάδοσης καθώς στον ψηφιακό του χώρο υπάρχουν πολλές λειτουργίες με τις οποίες αλληλεπιδρά ο μαθητής και ο εκπαιδευτικός ως σχεδιαστής και καθοδηγητής του μαθητή. Ο μαθητής μέσω τους φτάνει στην κατάκτηση ανώτερης γνώσης και επικοινωνίας με την ταυτόχρονη πραγμάτωση πολλών μοντέρνων διδακτικών αρχών είτε μεμονωμένα σε έναν διδασκόμενο είτε παράλληλα σε πολλούς, κάτι που δεν πραγματοποιείται συνήθως εύρυθμα εντός της σχολικής τάξης. Τα ψηφιακά παιχνίδια

αποτελούν σημάδι της εποχής τους δηλαδή της λεγόμενης “Κοινωνίας της γνώσης” (Μάθηση και Διδασκαλία στην εποχή της πληροφορικής, Α.Ράπτης-Α.Ράπτη Α τόμος, 2017) και συνάδουν επακριβώς με τις απαιτούμενες προϋποθέσεις γνώσης καλής πληροφορικής, διευρυμένης μάθησης, κριτικής σκέψης και χειρισμού τεχνολογικών μέσων που καλούνται να καλύψουν οι μαθητές. Τα ψηφιακά μέσα τεχνολογίας πληροφορικής και επικοινωνίας πρέπει να λαμβάνονται υπόψην οπωσδήποτε στον εκπαιδευτικό σχεδιασμό: “Όχι τόσο για την επίτευξη ενός αμφίβολου αξίας τεχνολογικού εκσυγχρονισμού, αλλά για την αξιοποίηση των δυνατοτήτων αυτών στην υπηρεσία του ανθρώπου, της κοινωνίας, του πολιτισμού και της μάθησης. (Α.Ράπτης-Α.Ράπτη, Α τόμος “Μάθηση και Διδασκαλία στην εποχή της πληροφορικής, 2.1.1,2017).

1.2 Η ΣΗΜΑΣΙΑ ΤΩΝ ΨΗΦΙΑΚΩΝ ΕΚΠΑΙΔΕΥΤΙΚΩΝ ΠΑΙΧΝΙΔΙΩΝ ΚΑΙ ΤΩΝ ΕΥΦΥΩΝ ΨΗΦΙΑΚΩΝ ΠΑΙΧΝΙΔΙΩΝ ΣΕ ΨΗΦΙΑΚΑ ΠΕΡΙΒΑΛΛΟΝΤΑ ΜΑΘΗΣΗΣ

Όλα τα μαθήματα μπορούν να ευνοηθούν από εκπαιδευτικά παιχνίδια σε ψηφιακά περιβάλλοντα από τα μαθηματικά και την γλώσσα έως επιστήμες, τέχνες και ξένες γλώσσες. Το ψηφιακό περιβάλλον δεν θα αντικαταστήσει σε καμία περίπτωση την τάξη ή το σύνηθες μαθησιακό περιβάλλον στην διδακτική διαδικασία, ο ρόλος του είναι βοηθητικός δηλαδή εκείνος του ενισχυτή της γνώσης που θέτει την κατάκτηση της σε άλλο επίπεδο. Ο μαθητής θα μπορεί παράλληλα να συνομιλεί με τους υπόλοιπους συμμαθητές του και με τον διδάσκοντα, να παρακολουθεί τις απορίες του να επιλύονται μέσω συζήτησης με ψηφιακούς βοηθούς, να παρακολουθεί την πρόοδο του και να καταγράφονται οι αδυναμίες του μέσω υψηλού και χαμηλού σκορ. Ακόμη ο διδάσκοντας θα μπορεί να ενημερώνεται για την πρόοδο των μαθητών του άμεσα και γρήγορα και να κεντρίζει το ενδιαφέρον των μαθητών του κάνοντας το μάθημα πιο διασκεδαστικό άρα και πιο κατανοητό και μεταδοτικό. Συνεπώς το παιχνίδι σαν εφαρμογή επιδρά ως συντονιστής της ύλης και μέσο διεπικοινωνίας ανάμεσα στον διδάσκοντα και τους διδασκόμενους. Ξεπερνά τα όρια ενός απλού εγχειριδίου γνώσης ή ενός επιστημονικού βιβλίου που θα ενίσχυε την πιο μηχανική αποστήθιση γνώσης και μετατρέπεται σε ένα βοηθητικό πολυεργαλείο μάθησης που με γλαφυρά γραφικά, ζωντανή εικόνα και έντονους ήχους αιχμαλωτίζει το ενδιαφέρον του μαθητή και εξηγεί με παραδείγματα πολύτιμη γνώση οδηγώντας σε μία ομαδοσυνεργατική και βιωματική αφομοίωση των περίπλοκων εννοιών. Με τα εκπαιδευτικά παιχνίδια και τα περιβάλλοντα τους οι χρήστες αφομοιώνουν ενεργά την γνώση, βρίσκουν εποικοδομητικές λύσεις στα προβλήματα και βελτιώνουν την επικοινωνία μέσα στην τάξη. Αυτά τα περιβάλλοντα είτε σύγχρονα είτε ασύγχρονα δρουν καταλυτικά στην εξασφάλιση της γνώσης. Από την άλλη πλευρά τα ευφυή ψηφιακά εκπαιδευτικά παιχνίδια ή αλλιώς τα παιχνίδια στα οποία χρησιμοποιείται τεχνητή νοημοσύνη όπως ψηφιακοί βοηθοί διευκολύνουν ακόμη περισσότερο τον μαθητή στην κατανόηση του μαθήματος, καθώς οι ψηφιακοί βοηθοί βρίσκονται εκεί μέσω συζήτησης να απαντούν στα διάφορα ερωτήματα του μαθητή, να διαχειρίζονται την γνώση ώστε να την διαβαθμίζουν ανάλογα με τις μαθησιακές ανάγκες του

κάθε χρήστη στις διάφορες πίστεις, να επαναλαμβάνουν κομμάτια ύλης, να του υπενθυμίζουν γενικούς κανόνες και διορθώνουν τα λάθη του επεξηγηματικά. Γι' αυτό και στην παρούσα εργασία επιλέχθηκε η δημιουργία ενός ευφυούς εκπαιδευτικού παιχνιδιού προγραμματισμού με την χρήση ψηφιακού βοηθού ώστε να μεγιστοποιηθούν τα οφέλη του παιχνιδιού στην διδασκαλία. Αναλυτικότερα, στόχος μου ήταν ο σχεδιασμός και η δημιουργία ενός εκπαιδευτικού ψηφιακού παιχνιδιού 2D σε γλώσσα Python μαζί με έναν ψηφιακό βοηθό chatbot που θα βοηθούσε τους μαθητές στην αφομοίωση και την μετέπειτα χρήση βασικών κωδικών ψευδόγλωσσας προγραμματισμού. Το παιχνίδι αναπτύχθηκε με την χρήση της Python του PyGame και του ChatGpt API.

1.3 ΔΙΑΡΘΡΩΣΗ ΚΕΦΑΛΑΙΩΝ

Τα κεφάλαια που ακολουθούν θα αναλύσουν τις διάφορες θεωρίες πάνω στις οποίες στηρίχθηκε η δημιουργία της εργασίας καθώς και τους τρόπους με τους οποίους προσεγγίστηκε το υλικό και διαμορφώθηκε η καταληκτική δομή του παιχνιδιού. Αναλυτικότερα, στο **δεύτερο κεφάλαιο** θα εξεταστεί το θεωρητικό υπόβαθρο και η ανασκόπηση της βιβλιογραφίας πάνω στο οποίο έγκειται η δομή και η λειτουργία του παιχνιδιού. Ύστερα, στο **τρίτο κεφάλαιο** θα παρουσιαστεί η μεθοδολογία της δημιουργίας του. Στο **τέταρτο κεφάλαιο** θα εξηγηθεί η αρχιτεκτονική καθώς και η επισκόπηση της εφαρμογής. Στο **πέμπτο κεφάλαιο** θα αναφερθεί η συνολική αξιολόγηση του παιχνιδιού και τέλος στο **έκτο κεφάλαιο** αναλυθούν τα αποτελέσματα της εργασίας.

ΚΕΦΑΛΑΙΟ ΔΕΥΤΕΡΟ

ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ ΚΑΙ ΑΝΑΣΚΟΠΗΣΗ ΤΗΣ ΒΙΒΛΙΟΓΡΑΦΙΑΣ

2.1 ΕΙΣΑΓΩΓΗ ΣΤΟ ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

Σήμερα παρατηρείται ολοένα και περισσότερο το φαινόμενο χρήσης ψηφιακών παιχνιδιών στην εκπαίδευση χάρη στην πληθώρα εκπαιδευτικών λειτουργιών που μπορούν να προσφέρουν στην διδακτική διαδικασία. Η παράδοση του μαθήματος γίνεται με την χρήση τους πιο διασκεδαστική, διαδραστική και αποφεύγεται ο μηχανικός τρόπος αποστήθισης. Κάνοντας χρήση τέτοιων παιχνιδιών μαθήματα όπως ιστορία, μαθηματικά, ζωγραφική ακόμα και αρχαία μπορούν να διδαχθούν εξ αποστάσεως ταυτόχρονα σε έναν τεράστιο όγκο μαθητών. Ο συνεχώς εξελισσόμενος τομέας του προγραμματισμού είναι απολύτως συνυφασμένος με την ραγδαία εξέλιξη της τεχνολογίας και επακολούθως συνυφασμένος με την εξέλιξη της δημιουργίας των αρχικών παιχνιδιών. Μία σημαντική υποκατηγορία των εκπαιδευτικών παιχνιδιών που απορρέει από αυτή την ευρύτερη ανάπτυξη είναι τα ευφυή εκπαιδευτικά παιχνίδια που χρησιμοποιούν ψηφιακούς βοηθούς και άλλα μέσα τεχνητής νοημοσύνης για να μεταδοθεί η γνώση πιο εύκολα στους μαθητές. Στην ζωή μας έχει εξάλλου εδραιωθεί πια η τεχνητή νοημοσύνη. Έννοιες όπως ψηφιακοί βοηθοί, ευφυή γραφικά περιβάλλοντα επαφής με τον χρήστη, ψηφιακά παιχνίδια με χρήση τεχνητής νοημοσύνης είναι πλέον κομμάτι της καθημερινότητας. Ακόμα και στον τομέα της εκπαίδευσης ο όρος τεχνητή νοημοσύνη, δεν αποτελεί κάτι το πρωτοφανές. Τα παιδιά από πολύ νεαρή ηλικία μαθαίνουν να αλληλεπιδρούν με αυτές τις τεχνολογίες. Στα σχολεία εγκαθίστανται έξυπνοι πίνακες που βοηθούν στην ομαλή εξαγωγή του μαθήματος, βοηθώντας καθηγητές και μαθητές στο έργο τους. Στο σπίτι τα παιδιά εφοδιάζονται με ηλεκτρονικές συσκευές όπως (tablet) που τα βοηθούν στα μαθήματα τους. Νέες εφαρμογές δημιουργούνται καθημερινά για να καλύψουν τις ανάγκες αυτού του νέου τρόπου διδασκαλίας. Τα παιδιά φαίνεται να μαθαίνουν πιο γρήγορα και αποτελεσματικά περίπλοκες έννοιες μέσω παιχνιδιών. Φυσικά πάντοτε υπάρχουν περιθώρια περαιτέρω ανάπτυξης και εξέλιξης δίχως τα παιχνίδια να αποτελούν εξαίρεση. Ίσως στο μέλλον να υπάρξουν ακόμη πιο εξελιγμένα παιχνίδια που κάνουν χρήση της τεχνητής νοημοσύνης σε άλλα επίπεδα.

2.2 ΣΤΟΧΟΙ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ ΜΕ ΜΙΑ ΜΑΤΙΑ ΣΤΗ ΛΟΓΙΚΗ ΤΟΥ

Κυρίαρχος στόχος της εργασίας μου είναι η δημιουργία ενός εκπαιδευτικού 2-D παιχνιδιού όπου οι μαθητές θα μπορούν με την χρήση ψηφιακού βοηθού να διδαχθούν και να εξοικειωθούν με βασικές έννοιες προγραμματισμού. Πιο συγκεκριμένα, το παρόν παιχνίδι αφορά ένα μάγο ο οποίος βρίσκεται σε έναν χώρο από τον οποίο πρέπει να δραπετεύσει. Για να το κάνει αυτό θα πρέπει να διαβάσει κάποιους πάπυρους οι οποίοι περιέχουν ψευδοκώδικα με την μορφή ξορκιών. Διαβάζοντας τα ξόρκια λαμβάνει γνώση για να γράψει τον τελικό αλγόριθμο (ξόρκι) που θα τον βοηθήσει να δραπετεύσει από τον χώρο αυτόν. Ο ψηφιακός βοηθός θα είναι σε μορφή μαγικού βιβλίου που ο παίκτης θα γράφει τις απορίες του και θα τροφοδοτείται με απαντήσεις σε μορφή κειμένου ή βοηθητικού ψευδοκώδικα για να μπορέσει να ολοκληρώσει την διαδικασία. Τέλος θα διατηρείται το σκορ του καλύτερου παίκτη που θα περιέχει την διάρκεια ολοκλήρωσης της διαδικασίας, την ζωή που του απέμεινε και την μαγεία που του απέμεινε. Για την κατασκευή του παιχνιδιού μελετήθηκε δομές των παιχνιδιών (2-D) με την βοήθεια της βιβλιοθήκης PyGame, ευρύτερα θεωρίες ψηφιακών παιχνιδιών, έννοιες και δομές παιχνιδιών περιπέτειας καθώς και το θεωρητικό υπόβαθρο του ψηφιακού βοηθού (ChatBot). Παράλληλα και οι αλγόριθμοι οι οποίοι χρησιμοποιούνταν σε παλιά παιχνίδια για απεικόνιση καθώς και οι αλγόριθμοι διδασκαλίας. Επίσης ερευνήθηκε η βιβλιογραφία από την οποία αντλήθηκε υλικό για την βέλτιστη δημιουργία του παιχνιδιού, την δημιουργία δηλαδή της αρχιτεκτονικής της εφαρμογής του και την μεθοδολογία πίσω από το παιχνίδι. Τέλος το παιχνίδι αξιολογήθηκε από παίκτες. Ύστερα από την αξιολόγηση τα αποτελέσματα που δόθηκαν ήταν ικανοποιητικά και πληρούσαν τις προϋποθέσεις ενός εκπαιδευτικού παιχνιδιού προγραμματισμού με ψηφιακό βοηθό.

2.3 ΨΗΦΙΑΚΑ ΠΟΛΥΜΕΣΑ

Με τον όρο πολυμέσα αναφερόμαστε σε οποιονδήποτε συνδυασμό απλού ή μορφοποιημένου κειμένου, γραφικών, εικόνων, κινούμενης εικόνας (animation) και βίντεο που χρησιμοποιείται για την παρουσίαση πληροφοριών μέσω οποιουδήποτε ηλεκτρονικού μέσου. (Τεχνολογίες Πολυμέσων: Θεωρία, Υλικό, Λογισμικό, Φώτης Λαζαρίνης, 2007, εκδόσεις Κλειδάριθμος). Σε αυτούς τους ψηφιακούς χώρους έχουν την δυνατότητα αλληλεπίδρασης, ανατροφοδότησης και εκμάθησης δεξιοτήτων. Κατασκευάζονται ανάλογα με τους στόχους του δημιουργού τους και χρησιμοποιούνται σε πολλούς τομείς εργασίας, σπουδών ακόμη και τέχνης. Οι κατηγορίες των ψηφιακών περιβαλλόντων ποικίλλουν ανάλογα με τον σχετικό τους τομέα. Ανάμεσα στις κυριότερες κατηγορίες τους περιλαμβάνονται: Τα μέσα κοινωνικής δικτύωσης (πχ. Instagram, Facebook) όπου οι χρήστες έχουν την ελευθερία δημιουργίας κοινωνικού περιεχομένου και αλληλεπίδρασης μέσω αντιδράσεων. Ελεγχόμενα Συστήματα στα οποία ανήκουν εφαρμογές εξυπηρέτησης και οργάνωσης οικιακού χώρου, γραφείου ή συσκευών. Τα συνεργατικά περιβάλλοντα εργασίας (πχ. Skype, Webex, Zoom)

όπου ανήκουν εφαρμογές και πλατφόρμες διαδικτυακών συνεδρίων και βιντεοκλήσεων που επιτρέπουν σε έναν μεγάλο αριθμό συνεργάτων να εργάζονται από κοινού για ένα πρότζεκτ. Η εικονική πραγματικότητα και η επαυξημένη πραγματικότητα στις οποίες ανήκουν εφαρμογές παιχνιδιών ή ευρύτερα ψηφιακά περιβάλλοντα αλληλεπίδρασης και τέλος η κατηγορία που αφορά την παρούσα εργασία: “τα εκπαιδευτικά παιχνίδια που δημιουργούνται σε ψηφιακά περιβάλλοντα”.

2.4 ΨΗΦΙΑΚΑ ΠΑΙΧΝΙΔΙΑ

Τα ψηφιακά παιχνίδια είναι παιχνίδια που εξελίσσονται σε έναν ψηφιακό χώρο και μπορεί κανείς να τα παίξει μόνο μέσω ψηφιακών συσκευών. (πχ. Ηλεκτρονικός υπολογιστής, λάπτοπ, τάμπλετ, παιχνιδοκονσόλα, κινητό κτλπ.). (Τεχνολογίες Πολυμέσων: Θεωρία, Υλικό, Λογισμικό, Φώτης Λαζαρίνης,2017). Τα ψηφιακά παιχνίδια χωρίζονται σε 2D ή 3D. 2D παιχνίδια αποκαλούνται τα παιχνίδια που η προσομοίωση κίνησης τους γίνεται σε δύο επίπεδα. Η αλλαγή θέσης του ίδιου αντικειμένου και η εμφάνιση παρόμοιων διαδοχικών εικόνων είναι οι δύο μορφές που συναντάμε στα 2D παιχνίδια. (Τεχνολογίες Πολυμέσων: Θεωρία, Υλικό, Λογισμικό, Φώτης Λαζαρίνης,2017). Το παιχνίδι με το οποίο ασχολείται η εργασία ανήκει στα 2D παιχνίδια. Αναλύοντας περαιτέρω τις λειτουργίες που προσφέρει το ψηφιακό παιχνίδι, δίνεται η δυνατότητα στους χρήστες να τα παίξουν είτε κατά μόνους είτε παράλληλα με άλλους χρήστες και στο διαδίκτυο και δίχως διαδίκτυο. Τα ψηφιακά παιχνίδια με την βοήθεια μουσικής και γραφικών μπορούν να λάβουν πολλές μορφές και είναι πολύ δημοφιλή μέσα ψυχαγωγίας. Μολονότι ψυχαγωγούν, τα οφέλη τους δεν σταματούν στην διασκέδαση. Μπορούν να χρησιμοποιηθούν και ως εξαιρετικά εργαλεία εκμάθησης δεξιοτήτων και ευρύτερα ως συμπληρωματικά εργαλεία διδασκαλίας. Ανάλογα με τον τρόπο δημιουργίας τους, την εφαρμογή στην οποία δημιουργήθηκαν και τον σκοπό τους χωρίζονται σε ορισμένες βασικές κατηγορίες. Πρώτον στα παιχνίδια Ρόλων (Role Playing Games-RPGs). Σε αυτή την κατηγορία ανήκουν όσα παιχνίδια επιτρέπουν στον παίκτη να κατασκευάσει έναν κύριο παίκτη (avatar) και να αλληλοεπιδρά μέσω του στις διάφορες πίστες και επιλογές προσποιούμενος τον χαρακτήρα αυτόν. Δεύτερη κατηγορία ψηφιακών παιχνιδιών είναι τα παιχνίδια αθλητισμού. Σε αυτά τα παιχνίδια αθλήματα απεικονίζονται στην οθόνη των ψηφιακών μέσων και ο χρήστης μέσω παιχνιδοκονσόλας μπορεί να “παίξει” ποδόσφαιρο, μπάσκετ, τένις και άλλα πολλά. Στην τρίτη κατηγορία ανήκουν τα παιχνίδια στρατηγικής. Σε αυτά οι χρήστες καταστρώνουν κάποιο σχέδιο στρατηγικής για να ολοκληρώσουν ορισμένες λειτουργίες του παιχνιδιού. Στην επόμενη κατηγορία ανήκουν τα παιχνίδια παζλ όπου κλασικά τοποθετείται κάποιο αίνιγμα ή κάποιο παζλ το οποίο ο χρήστης καλείται να φέρει εις πέρας ώστε να περάσει το επίπεδο της πίστας. Άλλες δύο κατηγορίες είναι τα παιχνίδια δράσης και περιπέτειας. Αν και ακούγονται το ίδιο η διαφορά τους έγκειται στο ότι στα παιχνίδια δράσης μέσω ενός χαρακτήρα ο χρήστης προσπαθεί να περάσει τις πίστες επιβιώνοντας από ακραίες σκηνές δράσης όπως κυνηγητό ή πυροβολισμούς. Ενώ στα παιχνίδια περιπέτειας ο χρήστης από το τι επιλέγει για τον χαρακτήρα του αλλάζει την πορεία της ιστορίας και επλύνει αινίγματα, γρίφους ή και άλλες

λειτουργίες ώστε να περάσει στην επόμενη πίστα. Τα παιχνίδια κοινωνικής δικτύωσης αποτελούν μια ιδιαίτερα ενδιαφέρουσα κατηγορία όπου οι χρήστες μέσω της πλατφόρμας του κοινωνικού δικτύου στο οποίο συνδέονται και παίζουν μπορούν διαδραστικά να επικοινωνούν μεταξύ τους και να ανατροφοδοτούν την ολόκληρη διαδικασία του παιχνιδιού. Τέλος, φτάσαμε στη κατηγορία των ψηφιακών παιχνιδιών που μας ενδιαφέρει περισσότερο: τα εκπαιδευτικά ψηφιακά παιχνίδια. Τα εκπαιδευτικά παιχνίδια αποσκοπούν κυρίως στην εκμάθηση κάποιων δεξιοτήτων και στην καλλιέργεια της λογικής, αντίληψης των μαθητών. Απώτερος στόχος τους είναι η αφομοίωση των γνώσεων του εκάστοτε μαθήματος από τον μαθητή.

Ορισμένα παραδείγματα ψηφιακών παιχνιδιών είναι:

- Fortnite <https://www.fortnite.com/>
- Minecraft <https://www.minecraft.net/en-us>
- Roblox <https://www.roblox.com/>
- Overwatch <https://overwatch.blizzard.com/en-us/>

2.5 ΕΚΠΑΙΔΕΥΤΙΚΑ ΠΑΙΧΝΙΔΙΑ

Τα εκπαιδευτικά παιχνίδια είναι παιχνίδια που δημιουργήθηκαν με σκοπό την εκμάθηση δεξιοτήτων ή ευρύτερα την καλλιέργεια γνώσης. Οι στόχοι τους είναι καθαρά εκπαιδευτικοί και συνδυάζοντας μουσική, γραφικά και λειτουργίες επιτέλεσης προσφέρουν έναν πιο διαδραστικό και διασκεδαστικό τρόπο μάθησης. (Μάθηση και Διδασκαλία στην εποχή της Πληροφορίας, Α.Ράπτης-Α.Ράπτη, 2017, Α τόμος). Βασικότερα χαρακτηριστικά τους είναι η δυνατότητα επικοινωνίας και αλληλεπίδρασης μεταξύ των χρηστών, η ενθάρρυνση μέσω αμοιβών όταν οι χρήστες επιτελούν με επιτυχία μία λειτουργία, δυνατότητα ομαδικότητας ανάμεσα στους παίκτες, οι επιβλέποντες γονείς και οι καθηγητές να ελέγχουν την κατακτημένη γνώση των μαθητών και τέλος οι μαθητές να κατακτούν παράλληλα παραπάνω από έναν γνωστικό τομέα. Τα εκπαιδευτικά παιχνίδια χωρίζονται σε διάφορες κατηγορίες ανάλογα με τα διαφορετικά γνωστικά αντικείμενα με τα οποία συνάδουν. Ανάμεσα στις πιο σημαντικές κατηγορίες των εκπαιδευτικών παιχνιδιών περιλαμβάνονται: Αρχικά, τα παιχνίδια γλωσσικής διδασκαλίας όπου οι μαθητές καλούνται μέσω ανάλογων παιχνιδιών να επιλύσουν γλωσσικές ασκήσεις, να λύσουν σταυρόλεξα, να κάνουν αντιστοιχίσεις και γενικότερα να εξασκηθούν σε ορθογραφία και στην εκμάθηση γλωσσικών λέξεων. Άλλη μία κατηγορία είναι τα παιχνίδια μαθηματικών όπου βασικός γνώμονας είναι η απόκτηση μαθηματικών ικανοτήτων όπως διαίρεση, πολλαπλασιασμός κτλ αλλά και όσο αυξάνεται το διδακτικό επίπεδο σύνθετων μαθηματικών εξισώσεων. Ύστερα, έχουμε παιχνίδια Ιστορίας όπου ο μαθητής μαθαίνει σημαντικές ιστορικές εξελίξεις και παιχνίδια επιστημών όπως η βιολογία, η φυσική και η χημεία. Σε τέτοιου είδους παιχνίδια προσφέρεται μία πληθώρα ασκήσεων όπου συνδυάζεται θεωρία με γραφικές αναπαραστάσεις για την αποστήθιση και εξάσκηση τύπων. Άλλες κατηγορίες είναι τα παιχνίδια υγείας και αθλητισμού όπου οι μαθητές μαθαίνουν πως να διαχειρίζονται την διατροφή τους και να αθλούνται. Τέλος, η

κατηγορία με την οποία ασχολείται η παρούσα εργασία είναι τα παιχνίδια προγραμματισμού όπου ο μαθητής μαθαίνει γλώσσες προγραμματισμού και γενικότερα ψηφιακές δεξιότητες.

Τα παιχνίδια δεν χωρίζονται όμως μόνο με βάση το γνωστικό αντικείμενο αλλά και με το εάν συμπεριλαμβάνουν τεχνητή νοημοσύνη ή όχι.

Ορισμένα παραδείγματα εκπαιδευτικών παιχνιδιών είναι:

- Scratch <https://scratch.mit.edu/>
- ABCMouse <https://www.abcmouse.com/abc/8a08850bc2=T1840430826.1694718098.138>
- NationalGeographic Kids <https://kids.nationalgeographic.com/games/>

2.6 ΕΥΦΥΗ ΕΚΠΑΙΔΕΥΤΙΚΑ ΠΑΙΧΝΙΔΙΑ ΜΕ ΧΡΗΣΗ ΨΗΦΙΑΚΟΥ ΒΟΗΘΟΥ

Τα ευφυή εκπαιδευτικά παιχνίδια αποτελούν υποκατηγορία των εκπαιδευτικών παιχνιδιών. Αποσκοπούν και αυτά στην εκμάθηση δεξιοτήτων και στην καλλιέργεια γνώσης ανάλογης με το γνωστικό αντικείμενο με το οποίο συνάδουν, με μοναδική διαφορά την ύπαρξη τεχνητής νοημοσύνης σε αυτά. (Μάθηση και Διδασκαλία στην εποχή της πληροφορίας, Α.Ράπτης-Α.Ράπτη, Α τόμος, 2017). Η τεχνητή νοημοσύνη (artificial intelligence-a.i) μπορεί να λάβει ποικίλες μορφές σε αυτά τα εκπαιδευτικά παιχνίδια και να διευκολύνει την εκπαιδευτική διαδικασία. Υπάρχουν παιχνίδια μαθηματικών που χρησιμοποιούν ψηφιακούς βοηθούς για να παρατείνουν την λογική σκέψη των χρηστών με γρίφους, αινίγματα ακόμη και με την επίλυση μαθηματικών προβλημάτων. Παιχνίδια επιστήμης με προσομοιώσεις πειραμάτων και γλαφυρές αναπαραστάσεις καθώς και παιχνίδια γλωσσικών ασκήσεων όπου ο χρήστης με την επίλυση διαδραστικών προβλημάτων μαθαίνει γλωσσικές δομές και λαμβάνει ανατροφοδότηση παράλληλα από τον ψηφιακό βοηθό σχετικά με τα κομμάτια ύλης που δεν κατανόησε. Επίσης, υπάρχουν ευφυή εκπαιδευτικά παιχνίδια προγραμματισμού όπου βοηθούν τους χρήστες να κατανοήσουν πιο εύκολα βασικές έννοιες προγραμματισμού και να εξασκηθούν πάνω σε διάφορες εντολές με έναν πιο ζωντανό και αλληλεπιδραστικό τρόπο. Σε αυτή την κατηγορία ανήκει και το ευφυές παιχνίδι προγραμματισμού το οποίο αναλύει η παρούσα εργασία και τα οφέλη του στην εκμάθηση εντολών προγραμματισμού με την χρήση ψηφιακού βοηθού.

Ορισμένα παραδείγματα ευφυών εκπαιδευτικών παιχνιδιών είναι:

- Minecraft Education <https://education.minecraft.net/en-us>
- Code.org <https://code.org/>

2.7 ΤΙ ΕΙΝΑΙ ΕΝΑΣ ΨΗΦΙΑΚΟΣ ΒΟΗΘΟΣ ΚΑΙ ΟΙ ΠΙΟ ΣΗΜΑΝΤΙΚΕΣ ΚΑΤΗΓΟΡΙΕΣ

Ο ψηφιακός βοηθός είναι μοντέλο AI που βοηθά τον χρήστη μέσω κειμένου ή μέσω γραπτής ή φωνητικής συνομιλίας να αναζητήσει τις πληροφορίες που χρειάζεται ή να επιτελέσει την οποιαδήποτε επιθυμητή στην χρήστη ενέργεια. Υπάρχουν διάφορες κατηγορίες ψηφιακών βοηθών που διαθέτουν μοντέλα προκαθορισμένης γνώσης (pre-trained models) καθώς και τεχνολογία επεξεργασίας φυσικής γλώσσας (NLP). (Natural Language Processing in LISP, “An Introduction to Computational Linguistics, Gerald Gazdar, Chris Mellish, 1989.)

Βασικότερες κατηγορίες είναι:

Οι υπηρεσίες προσωπικών βοηθών (personal assistant) όπου οι ψηφιακοί βοηθοί απαντούν σε ερωτήσεις και διατελούν λειτουργίες εκ μέρους του χρήστη. (πχ. Siri, Amazon Alexa κ.α). Άλλη μία κατηγορία είναι οι υπηρεσίες πελατών. Στις υπηρεσίες πελατών (customer services) μέσω chatbots απαντώνται ερωτήσεις και εξυπηρετούνται πελάτες. Υπάρχουν επίσης ιατρικοί βοηθοί (medical assistants) που είναι ικανοί να διαγνώσουν ασθένειες και να παρέχουν ιατρικά δεδομένα. (πχ. IBM Watson for Healthcare). Άλλες κατηγορίες είναι βοηθοί διαχείρισης πληροφοριών (Information Management, πχ. Evernote assistant) που αναλύει και αναζητά πληροφορίες, βοηθοί επιστημονικής έρευνας και ανάλυσης (Scientific Research and Analysis) που αναλύουν επιστημονικά πορίσματα και πληροφορίες και επαγγελματικοί βοηθοί (Professional Assistance) όπως οι Legal AI Assistants που συμβουλεύουν για θέματα νομικής ή οικονομικής φύσεως και βοηθοί ψυχολογικής υποστήριξης για την διάγνωση και διαχείριση προβλημάτων ψυχολογικής φύσεως όπως το Woebot. Τέλος, η κατηγορία με την οποία ασχολείται η παρούσα εργασία: οι εκπαιδευτικοί βοηθοί (Educational Assistants) όπως είναι τα Duolingo chatbots που παρέχουν απαντήσεις στις ερωτήσεις των μαθητών και παρέχοντας ευρύτερα πληροφορίες βοηθούν στην εκπαιδευτική διαδικασία εκμάθησης ικανοτήτων.

2.8 ΤΙ ΕΙΝΑΙ ΕΝΑ ΜΟΝΤΕΛΟ ΠΡΟΚΑΘΟΡΙΣΜΕΝΗΣ ΓΝΩΣΗΣ

Ένα μοντέλο προκαθορισμένης γνώσης είναι ένα υπολογιστικό μοντέλο που έχει εκπαιδευτεί πρώτα σε έναν μεγάλο όγκο δεδομένων και ύστερα χρησιμοποιείται. Αυτή η προπαίδευση τα καταστά ικανά να κατέχουν γνώσεις σε διάφορα γνωστικά αντικείμενα ανάλογα με τον στόχο της εφαρμογής στην οποία θα χρησιμοποιηθούν. Τα δεδομένα στα οποία εκπαιδεύονται μπορεί να είναι γλωσσικά και σημασιολογικά που επιτρέπουν την μετέπειτα επικοινωνία με τους χρήστες. Θα μπορούσαμε ακόμη να πούμε πως τα προκαθορισμένα μοντέλα έχουν προκαλέσει την εξέλιξη της τεχνητής νοημοσύνης. (Natural Language Processing in LISP, “An Introduction to Computational Linguistics, Gerald Gazdar, Chris Mellish, 1989)

2.9 ΤΙ ΕΙΝΑΙ Η ΤΕΧΝΟΛΟΓΙΑ NLP

Ο όρος AI (Artificial Intelligence) αφορά την τεχνητή νοημοσύνη και εφαρμόζεται σε συστήματα που επιτελούν λειτουργίες που κανονικά απαιτούν ανθρώπινη νοημοσύνη. Η τεχνητή νοημοσύνη σήμερα χρησιμοποιείται σε διάφορους τομείς της ζωής μας όπου παραδοσιακά χρειαζόταν ανθρώπινη ενέργεια. Η τεχνολογία NLP είναι η επεξεργασία της φυσικής γλώσσας και αποτελεί υποκατηγορία της τεχνητής νοημοσύνης. Βασικός γνώμονας αυτής της τεχνολογίας είναι τα ψηφιακά μέσα να διαθέτουν παρόμοια γλωσσική ικανότητα με τους ανθρώπους ώστε να μπορούν να επικοινωνούν μαζί τους. Ο συγκεκριμένος ψηφιακός βοηθός που χρησιμοποιείται εδώ στηρίζεται σε μοντέλο της Open AI που εφαρμόζει την NLP με μία μεγάλη γκάμα αλγόριθμων για να απαντά στις ερωτήσεις των χρηστών.

2.10 ΟΡΙΣΜΟΣ ΑΛΓΟΡΙΘΜΟΥ ΚΑΙ ΑΝΑΣΚΟΠΗΣΗ ΒΑΣΙΚΩΝ ΑΛΓΟΡΙΘΜΩΝ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ

Αν και δεν υπάρχει κάποια κοινά αποδεκτή φρασεολογία για να περιγράψουμε τη συγκεκριμένη έννοια του αλγορίθμου υπάρχει εν τούτοις γενική συμφωνία για το τι σημαίνει αυτή. Ένας αλγόριθμος είναι μία ακολουθία ξεκάθαρων, ρητών εντολών για την επίλυση ενός προβλήματος, δηλαδή για την παραγωγή της απαιτούμενης εξόδου για κάθε αποδεκτή είσοδο σε πεπερασμένο χρόνο. (Εισαγωγή στην ανάλυση και Σχεδίαση Αλγορίθμων, Anany Levitin, εκδόσεις Τζιόλα,2008). Το παιχνίδι μου σχεδιάστηκε με στόχο οι μαθητές να κατανοήσουν αυτούς τους βασικούς αλγόριθμους που θα περιγραφθούν στην συνέχεια. Σκοπός των παικτών είναι να μάθουν τέσσερα είδη βασικών αλγορίθμων:

- Αλγόριθμος για αριθμητικά προβλήματα
- Αλγόριθμος ταξινόμησης
- Αλγόριθμος αναζήτησης και
- Αλγόριθμος επεξεργασίας συμβολοσειρών

2.10.1 ΠΑΡΑΔΕΙΓΜΑ ΑΡΙΘΜΗΤΙΚΟΥ ΑΛΓΟΡΙΘΜΟΥ

Αρχικά ο αλγόριθμος για τα αριθμητικά προβλήματα (numerical problems) αφορά προβλήματα που εμπλέκουν μαθηματικά αντικείμενα συνεχούς συνήθως φύσης. Συγκεκριμένα στο παιχνίδι οι μαθητές εξοικειώνονται με την επίλυση εξίσωσης. Αναλυτικότερα καλεί τον μαθητή να γράψει κώδικα που θα επιλύει μία πολυωνμική εξίσωση πρώτου βαθμού.

2.10.2 ΠΑΡΑΔΕΙΓΜΑ ΑΛΓΟΡΙΘΜΟΥ MAX / MIN

Ο αλγόριθμος (max / min) είναι ένας πολύ βασικός αλγόριθμος ο οποίος έχει σκοπό να βρεί σε έναν πίνακα αριθμών τον μεγαλύτερο ή τον μικρότερο αριθμό μέσα σε αυτή.

2.10.3 ΠΑΡΑΔΕΙΓΜΑ ΑΛΓΟΡΙΘΜΟΥ ΤΑΞΙΝΟΜΗΣΗΣ

Ο αλγόριθμος ταξινόμησης αναφέρεται στην αναδιάταξη των στοιχείων μίας δεδομένης λίστας ώστε αυτά να βρεθούν σε αύξουσα σειρά. Πιο συγκεκριμένα ο αλγόριθμος με τον οποίο εξοικειώνεται εδώ ο μαθητής αποκαλείται bubblesort.

2.10.4 ΠΑΡΑΔΕΙΓΜΑ ΑΛΓΟΡΙΘΜΟΥ ΕΠΕΞΕΡΓΑΣΙΑΣ ΣΥΜΒΟΛΟΣΕΙΡΑΣ

Ο αλγόριθμος επεξεργασίας συμβολοσειρών αναφέρεται στην αναζήτηση μίας δεδομένης λέξης εντός ενός κειμένου. Πιο συγκεκριμένα εδώ συναντάμε τον αλγόριθμο εύρεσης λέξης μέσα σε μία πρόταση ή αλλιώς ταίριασμα συμβολοσειρών. (string matching)

(Εισαγωγή στην ανάλυση και Σχεδίαση Αλγορίθμων, Anany Levitin, εκδόσεις Τζιόλα,2008)

2.11 ΤΙ ΕΙΝΑΙ ΕΝΑ ΠΑΙΧΝΙΔΙ ΠΕΡΙΠΕΤΕΙΑΣ

Το θεωρητικό υπόβαθρο δεν θα μπορούσε να ολοκληρωθεί χωρίς να δοθεί ορισμός στον όρο παιχνίδι περιπέτειας (adventure game). Πιο συγκεκριμένα, το παιχνίδι περιπέτειας είναι ένα είδος βιντεοπαιχνιδιού στο οποίο ο παίκτης παίρνει τον ρόλο του πρωταγωνιστή της ιστορίας και εξερευνά πίστες ή επιλύει παζλ με θέση κάμερας σε τρίτο πρόσωπο.

2.12 ΠΑΙΧΝΙΔΙ ΠΑΖΛ

“Τα βιντεοπαιχνίδια παζλ αποτελούν ένα ευρύ είδος βιντεοπαιχνιδιών που δίνουν έμφαση στην επίλυση γρίφων. Οι τύποι παζλ μπορούν να δοκιμάσουν τις δεξιότητες επίλυσης προβλημάτων, συμπεριλαμβανομένης της λογικής, της αναγνώρισης προτύπων, της επίλυσης ακολουθιών, της χωρικής αναγνώρισης και της συμπλήρωσης λέξεων. Πολλά παιχνίδια παζλ περιλαμβάνουν ένα στοιχείο σε πραγματικό χρόνο και απαιτούν γρήγορη σκέψη, όπως το (Tetris 1985) και το (Lemmings 1991).”

(https://en.wikipedia.org/wiki/Puzzle_video_game)

2.13 ΚΑΜΕΡΑ ΣΕ ΤΡΙΤΟ ΠΡΟΣΩΠΟ

Τέλος η κάμερα στο παιχνίδι ανάγεται σε τρίτο πρόσωπο. “Οι κάμερες παρακολούθησης σε τρίτο πρόσωπο παρακολουθούν τους χαρακτήρες από πίσω. Ο παίκτης δεν ελέγχει την κάμερα με κανέναν τρόπο – δεν μπορεί για παράδειγμα να την περιστρέψει ή να τη μετακινήσει σε διαφορετική θέση. Αυτός ο τύπος συστήματος κάμερας ήταν πολύ κοινός στα πρώτα παιχνίδια 3D όπως το (Crash Bandicoot) ή το (Tomb Raider), καθώς είναι πολύ απλό στην εφαρμογή του. Ωστόσο, υπάρχει μια σειρά από ζητήματα με αυτό. Ειδικότερα, εάν η τρέχουσα προβολή δεν είναι κατάλληλη (είτε επειδή αποφράσσεται από κάποιο αντικείμενο, είτε επειδή δεν δείχνει αυτό που ενδιαφέρει τον παίκτη), δεν μπορεί να αλλάξει αφού η συσκευή αναπαραγωγής δεν ελέγχει την κάμερα. Μερικές φορές αυτή η άποψη προκαλεί δυσκολία όταν ένας χαρακτήρας στρίβει ή στέκεται μπροστά σε έναν τοίχο. Η κάμερα μπορεί να τραντάξει ή να καταλήξει σε άβολες θέσεις.”

(https://en.wikipedia.org/wiki/Virtual_camera_system#Third-person_view)

ΑΝΑΣΚΟΠΗΣΗ ΒΙΒΛΙΟΓΡΑΦΙΑΣ

Θα αναλυθεί η έρευνα γύρω από την δημιουργία παιχνιδιών πάζλ. Ευρύτερα, την χρήση των γραφικών, των χρωμάτων, των ήχων και της θεματολογίας που χρησιμοποιείται σε αυτά ώστε να είναι πιο ελκυστικά ως προς τους παίκτες. Επίσης, θα αναφερθούν παραδείγματα παιχνιδιών πάζλ από τα οποία αντλήθηκε έμπνευση για την δημιουργία του παρόντος παιχνιδιού.

2.14 ΑΝΑΣΚΟΠΗΣΗ ΒΙΒΛΙΟΓΡΑΦΙΑΣ ΓΙΑ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΕΝΟΣ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΠΑΙΧΝΙΔΙΟΥ ΜΕ ΧΡΗΣΗ ΨΗΦΙΑΚΟΥ ΠΑΙΧΝΙΔΙΟΥ

Τα βιντεοπαιχνίδια αποτελούν μία πολύ δημοφιλή κατηγορία παιχνιδιών στις μέρες μας. Είναι τόσο δημοφιλή γιατί προσφέρουν μία ευρεία γκάμα επιλογών στον χρήστη όπου μπορεί να αλληλεπιδρά (μέσω του ψηφιακού περιβάλλοντος του) λαμβάνοντας αποφάσεις για την εξέλιξη του. Αυτή η δυνατότητα σε συνδυασμό με τα γλαφυρά γραφικά, τα έντονα χρώματα και την μουσική κεντρίζουν το ενδιαφέρον των παικτών και τους ψυχαγωγούν. Πλέον ο κλάδος της εκπαίδευσης χρησιμοποιεί και εκείνος τα ψηφιακά παιχνίδια για να καταστήσει το έργο του πιο αποτελεσματικό και πιο βιωματικό. Οι διδασκόμενοι μπορούν να εκμάθουν μέσω ενός μόνο παιχνιδιού πολλαπλές δεξιότητες που αφορούν πολλαπλά γνωστικά αντικείμενα. Αυτή η διαδραστικότητα και η δυνατότητα επιλογής που προσφέρουν στον μαθητή, μετουσιώνουν την εκπαιδευτική διαδικασία σε κάτι διασκεδαστικό και απομακρύνουν από την τυποποιημένη και μη πρακτική αποστήθιση αποσπασμάτων θεωρίας. (Α. Ράπτης-Α.Ράπτη, Μάθηση και Διδασκαλία στην εποχή της πληροφορίας, εκδόσεις Openline,2017). Πιο συγκεκριμένα, ευρήματα ερευνών μας παρουσιάζουν τα πολλαπλά οφέλη που μας παρέχουν τέτοιου είδους βιντεοπαιχνίδια. Μπορούν να αναπτύξουν την λειτουργικότητα του εγκεφάλου μας στην επιτέλεση απλών αλλά και σύνθετων λειτουργιών. Όπως αναφέρει και ο Richard Mayer, ως εκπαιδευτικός ψυχολόγος στο πανεπιστήμιο της Καλιφόρνιας στη Σάντα Μπάρμπαρα : «Υπάρχουν πολλά στοιχεία που δείχνουν ότι οι άνθρωποι - και όχι μόνο οι νέοι - περνούν πολύ χρόνο παίζοντας παιχνίδια στις οθόνες τους. Αν μπορούσαμε να το μετατρέψουμε αυτό σε κάτι πιο παραγωγικό, αυτό θα ήταν μια αξιόλογη προσπάθεια.» (<https://knowablemagazine.org/article/mind/2019/video-games-educational-benefits>). Υπάρχουν τεράστια πλεονεκτήματα στην εκπαίδευση μέσω των ψηφιακών παιχνιδιών από την εκμάθηση μίας ξένης γλώσσας έως την επίτευξη κατανόησης περίπλοκων επιστημονικών εννοιών. Η παντοδυναμία συνδυασμού ψυχαγωγίας και εκμάθησης επιφυλάσσει ένα μακρόπνοο μέλλον στην εκπαίδευση. Οι μαθητές πια μπορούν να μαθαίνουν μέσω από μία εφαρμογή τόσο εύκολα και γρήγορα δίχως να δυσανασχετούν. Παράλληλα, οξύνονται οι ικανότητες επίλυσης προβλημάτων, η κριτική σκέψη και η

αντιληπτική ικανότητα των παικτών από την επίλυση παράλληλων διαδικασιών. Ένας άλλος τομέας που έχει διευρυνθεί στην εποχή μας είναι η χρήση τεχνητής νοημοσύνης για την περαιτέρω βοήθεια του μαθητή στην κατανόηση του υλικού που πρέπει να αφομοιώσει και γενικότερα στην διευκόλυνση του χειρισμού του παιχνιδιού, εν ολίγοις την διευκόλυνση της εμπειρίας του. Στο παιχνίδι χρησιμοποιείται ένας ψηφιακός βοηθός με την μορφή chatbot, ο οποίος καλείται να απαντά στις διάφορες απορίες των μαθητών για να εξομαλύνει την έκβαση του παιχνιδιού.

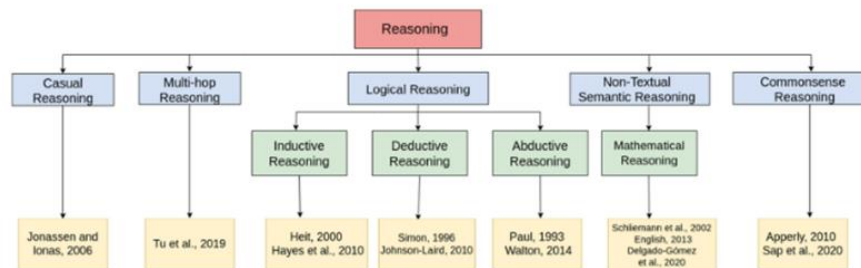
2.15 ΤΑ ΠΑΙΧΝΙΔΙΑ ΑΛΓΟΡΙΘΜΩΝ

Τα παιχνίδια αλγορίθμων αποτελούν κυρίως εκπαιδευτικά παιχνίδια που απευθύνονται στον μαθητικό πληθυσμό. Οι μαθητές που ζουν πια σε μία συνεχώς εξελισσόμενη κοινωνία πρέπει να εξοπλίζονται με τις κατάλληλες ικανότητες ώστε να μπορούν να ανταπεξέλθουν στις προϋποθέσεις που τους ζητούνται. Τα παιχνίδια αυτά εξασφαλίζουν την ομαλή αφομοίωση υλικού πολλαπλών γνωστικών τομέων δίχως κοπιώδη διεργασία και δίχως η γνώση να παραμείνει στην σφαίρα της θεωρίας. Μπορούν χρησιμοποιώντας τέτοια παιχνίδια και ειδικά παιχνίδια προγραμματισμού, να κατανοήσουν πολύπλοκα μαθηματικά προβλήματα, μαθηματικές έννοιες και αλγορίθμους προγραμματισμού. Για την εκπόνηση του παιχνιδιού μου εμπνεύστηκα από την εκπαιδευτική πλατφόρμα παιχνιδιού του (Scratch), ένα παιχνίδι εκμάθησης ψευδό γλώσσας με την βοήθεια (blocks). Το (Scratch) βοηθά τους μαθητές να “να εξερευνήσουν και να μάθουν σημαντικές δεξιότητες όπως αλγοριθμική και κριτική σκέψη, με τρόπο διασκεδαστικό μέσω της δημιουργίας έργων. Αυτό το καθιστά όχι μόνο ένα πρόγραμμα μονομερές, αλλά ένα πρόγραμμα με μεγάλη ποικιλία, φιλόξενο στη διαδικτυακή κοινότητα που προάγει τον προγραμματισμό και προκαλεί ενδιαφέρον ακόμη και στον επιστημονικό τομέα της νευροεκπαίδευσης.” (Analysis of Scratch Software in Scientific Production for 20 Years: Programming in Education to Develop Computational Thinking and STEAM Disciplines, by Pablo Duo Terron). Καθώς όμως εγώ δεν είχα πειρα στην κατασκευή ψευδογλώσσας, πόσο μάλλον την κατασκευή κάποιας μορφής ευπεπτης ψευδογλώσσας προγραμματισμού. Κατευθύνθηκα στην καλύτερη κατά την γνώμη μου γλώσσα προγραμματισμού, που πλέον έχει αρχήσει να μπαίνει και στα σχολεία, τη γλώσσα (Python). Στην γλώσσα που έχει γραφτεί και ο γνήσιος κώδικας του παιχνιδιού! Όπως χαρακτηρίζει και ο Hans Petter Langtangen στο άρθρο του με όνομα (A Primer on Scientific Programming with Python) η μέθοδος (eval()) και η μέθοδος (exec()) στην γλώσσα (Python) χαρακτηρίζονται ως μαγικές συναστίσεις, καθώς η πρώτη μετατρέπει ένα απλό κείμενο εισόδου (input string) σε κατάλληλης μορφής κώδικα και η δεύτερη τρέχει αυτόν τον κώδικα! Παρόλλα τα παραπάνω τελικά χρησιμοποιήθηκε μόνο η μέθοδος (exec()) στο τελικό παιχνίδι. Καθώς αφήνει τα παιδιά ελεύθερα να χρησιμοποιήσουν το μυαλό τους να τους κατευθύνει στην εύρεση του λόγου μη λειτουργίας του κώδικα σε περίπτωση λάθους! Η παραπάνω λογική παιχνιδιού μπορεί να περιγραφεί καλύτερα ως (TC) ή (computational thinking). Οι πρώτες προσεγγίσεις στον όρο (TC) εμφανίστηκαν στις δεκαετίες του 1960 και του 1970, όταν ο (Papert) τον περιέγραψε “Ως διαδικαστική σκέψη στη χρήση και ανάπτυξη δεξιοτήτων προγραμματισμού και αλγοριθμικών σχεδίων” . Σύμφωνα με τον (Wing, J.

Computational Thinking: What and Why?). Το πρώτο άτομο που εισήγαγε τον όρο (CT) στο επιστημονικό πεδίο το 2006. “Η (CT) είναι η ανάπτυξη και η γνώση που αποκτούν οι άνθρωποι σκεπτόμενοι σαν προγραμματιστές υπολογιστών.” Η αξονική τομογραφία έδειξε ότι το (TC) είναι μια θεμελιώδης και αναλυτική δεξιότητα που πρέπει να αναπτύξουν τα παιδιά του 21ου αιώνα, επειδή επιτρέπει στους μαθητές να αναλύσουν μια κατάσταση επίλυσης προβλημάτων και να την αναλύσουν σε απλούστερες μέχρι να βρεθεί μια λύση. Στη σημερινή αγορά εργασίας, απαιτούν τέτοιου ίδους δεξιότητες.

2.16 ΤΟ NLP ΩΣ ΠΗΓΗ ΓΝΩΣΗΣ ΤΗΝ ΣΗΜΕΡΙΝΗ ΕΠΟΧΗ

Το (NLP) ή (Natural Language Processing) είναι ο κύριος βοηθός του ατόμου στην αναζήτηση μιας πληροφορίας στο διαδίκτυο και γενικότερα σε ένα σύστημα βιβλιοθήκης. Τι ακριβώς όμως είναι το (NLP); Φανταστείτε το σενάριο στο οποίο ένα άτομο βρίσκεται σε μια ηλεκτρονική βιβλιοθήκη. Αντί να δίνει μόνο τον τίτλο του βιβλίου που ψάχνει, μπορεί να δώσει και μια περιγραφή αυτού, καθώς και να επικοινωνήσει με το σύστημα σε μορφή διαλόγου (chat), για να κατευθυνθεί ανάλογα στο βιβλίο που επιθυμεί. Ακόμα μπορεί να αποσπάσει περαιτέρω γνώση για αυτό “από μια δεύτερη οπτική γωνία”. Παρακάτω θα αναλυθούν συνοπτικά οι βασικές κατηγορίες λογισμού του (NLP) όπου και φαίνεται ο λόγος που επιλέχθηκε στο παρόν παιχνίδι ως η καλύτερη λύση στην επικοινωνία του μαθητή με το πρόγραμμα για την εξαγωγή γνώσης. Μέσα από την επισκόπηση του επιστημονικού περιοδικού με όνομα (Natural Language Processing Journal) και όνομα άρθρου (GPT-3.5, GPT-4, or BARD? Evaluating LLMs reasoning ability in zero-shot setting and performance boosting through prompts)



(Reasoning types)

Εικόνα 1

Φαίνεται ότι το (NLP) στηρίζεται στις παρακάτω μεθόδους λογικής για να προσφέρει την καλύτερη δυνατή εύρεση απαντήσεων από τον χρήστη, όχι μόνο μέσω δεδομένων αλλά και μέσω, δομημένου τρόπου σκέψης που θα κατευθύνει τον μαθητή να βρει τελικά την απάντηση αν έχει βρεθεί σε αδιέξοδο. Παρακάτω εμπεριέχονται αυτοί οι τρόποι συνοπτικά:

1. Συνήθης συλλογισμός ή (Casual reasoning)
 Με βάση τους την εφημερίδα με όνομα (Apperly, I., 2010. Mindreaders: The Cognitive Basis Of “Theory of Mind”. Psychology Press) “Η κοινή λογική είναι μια θεμελιώδης πτυχή της ανθρώπινης γνώσης και μας επιτρέπει να περιηγηθούμε στον κόσμο γύρω μας με τεκμηριωμένες αποφάσεις και κρίσεις”, κάτι που όλοι οι άνθρωποι διαθέτουμε και είναι βασικό στοιχείο στις συζητήσεις οποιουδήποτε με τον οποιονδήποτε για την εξαγωγή συμπερασμάτων. Ποσο μάλλον των παιδιών με ένα ψηφιακό βοηθό.
2. Συλλογισμός πολλαπλών βημάτων ή (Multi-hop reasoning)
 Με βάση τα δεδομένα από το (HotpotQA). “Η έννοια του συλλογισμού πολλαπλών βημάτων αναφέρεται στη διαδικασία με την οποία ένα σύστημα ή μοντέλο συνθέτει πληροφορίες από πολλαπλές πηγές ή κομμάτια δεδομένων για να εξαχθεί ένα συμπέρασμα ή μια απάντηση. Αυτό απαιτεί από το σύστημα να συνδέσει και να ενσωματώσει διάφορες πληροφορίες, συχνά μέσω πολλών λογικών βημάτων, προκειμένου να καταλήξει σε μια καλά υποστηριζόμενη απόκριση.” Πράγμα το οποίο θα βοηθήσει τον μαθητή να δει το πρόβλημα με άλλη οπτική γωνία ανάλογα την παντηση του ψηφιακού βοηθού!
3. Λογικός συλλογισμός ή (Logical reasoning) έχει 3 κατηγορίες:
 - a. Επαγωγικός συλλογισμός πρώτου τύπου ή (Inductive reasoning)
 Σύμφωνα με το (CLUTRR: Διαγνωστικό σημείο αναφοράς για επαγωγικό συλλογισμό από κείμενο. Στα: Πρακτικά του 2019 Συνέδριο για τις Εμπειρικές Μεθόδους στην Επεξεργασία Φυσικής Γλώσσας και το 9ο Διεθνές Κοινό Συνέδριο για την Επεξεργασία Φυσικής Γλώσσας. EMNLP- IJCNLP.) “Επαγωγικός συλλογισμός πρώτου τύπου είναι η κατανόηση συνθετικής γλώσσας με σχεσιακή βάση κειμένου και αιτιολόγησης.”
 - b. Επαγωγικός συλλογισμός δευτέρου τύπου ή (Deductive reasoning)
 Σύμφωνα με το άρθρο του (Dalvi et al., 2021). “Ο επαγωγικός συλλογισμός δευτέρου τύπου, είναι μια συνεπαγωγή πολλαπλών βημάτων σύνολο δεδομένων που έχει σχεδιαστεί για να δημιουργεί επεξηγήσεις με τη μορφή συνεπακόλουθης δέντρα. Αποτελείται από επτά πεδία: (ID), πλαίσιο, ερώτηση, απάντηση, υπόθεση, απόδειξη, μετα. Επιπλέον, οι συγγραφείς έχουν ορίσει τρεις εργασίες που σταδιακά αυξάνουν τη δυσκολία και απλοποιούν τα εγγενή προβλήματα του έργου.”
 - c. Παραγωγικός συλλογισμός ή (Abductive reasoning)
 Βασίζεται στην δημιουργία γνώσης από ήδη υπάρχουσα γνώση. Η βασική δομή σκέψης για τον μαθητή!
4. Μαθηματικός συλλογισμός ή (Mathematical reasoning)
 Ο μαθηματικός λογισμός είναι κυρίως για επήληση μόνο μαθηματικών προβλημάτων. Οι μαθητές μπορούν να δουν λύσεις μαθηματικών με σωστή λογική σειρά.
5. Συλλογισμός κοινής λογικής (Common sense reasoning)

Τέλος ο συλλογισμός κοινής λογικής, μοιάζει με τον συνήθη συλλογισμό και βοηθά τον μαθητή να επικοινωνεί, ακόμα καλύτερα και πιο ανθρώπινα με το μηχάνημα (API).

Με αυτόν τον τρόπο ο μαθητής μπορεί να ενημερωθεί πλήρως για το αντικείμενο του και να πράξει στην συνέχεια αναλόγως με την νέα γνώση και την περαιτέρω καθοδήγηση μιας μηχανής όπως το (API) του (Chat_Gpt_3.5_turbo). Τα παραπάνω καθιστούν αυτή την επιλογή του ψηφιακού βοηθού όχι την καλύτερη μόνο ως πηγή έμπνευσης, αλλά και την καλύτερη επιλογή ενσωμάτωσης της λογικής ψηφιακού βοηθού μέσα στο παιχνίδι!

ΚΕΦΑΛΑΙΟ ΤΡΙΤΟ

ΜΕΘΟΔΟΛΟΓΙΑ ΠΑΙΧΝΙΔΙΟΥ

3.1 ΕΙΣΑΓΩΓΗ ΜΕΘΟΔΟΛΟΓΙΑΣ

Το θέμα που επέλεξα για την συγγραφή της εργασίας μου ήταν η δημιουργία ενός (2D) ευφυούς εκπαιδευτικού παιχνιδιού εκμάθησης αλγορίθμων σε (Python) που θα βοηθούσε τους μαθητές να αντιληφθούν καλύτερα την περιπλοκότητα των αλγορίθμων αλλά και να εξασκηθούν σε βασικές λειτουργίες προγραμματισμού. Για το ευφύς κομμάτι επέλεξα να ενσωματώσω στο παιχνίδι μου έναν ψηφιακό βοηθό (AI chatbot). Σε αυτό το κεφάλαιο θα αναλύσω την μεθοδολογία πίσω από την δημιουργία και τον σχεδιασμό του παιχνιδιού καθώς τα μέσα και τους τρόπους που χρησιμοποιήθηκαν για την πραγματοποίησή του.

3.2 ΣΧΕΔΙΑΣΜΟΣ ΠΑΙΧΝΙΔΙΟΥ ΚΑΙ ΚΑΤΑΣΚΕΥΗ ΤΟΥ

Αρχικά, όλα ξεκίνησαν με την εύρεση της ιστορίας του παιχνιδιού, τον σχεδιασμό των πιστών και των αντικειμένων καθώς και ευρύτερα με τις λειτουργίες που θα επιτελεί το παιχνίδι ώστε να επιτευχθεί ο σκοπός της διδασκαλίας των αλγορίθμων. Ύστερα από αρκετό συλλογισμό κατέληξα στην επιλογή των τεσσάρων προαναφερθέντων αλγορίθμων καθώς θεώρησα πως ήταν οι πιο σημαντικοί και οι πιο χρήσιμοι στην χρήση μίας γλώσσας προγραμματισμού που μετέπειτα θα βοηθούσε τους μαθητές να συλλάβουν την λογική πίσω από την χρήση της. Αφότου λοιπόν σκέφτηκα τους αλγόριθμους, ξεκίνησα να σχεδιάζω την πλοκή του παιχνιδιού. Εφόσον το παιχνίδι είναι ένα παιχνίδι περιπέτειας 2d έκρινα πως πιο διασκεδαστικό μα συνάμα εκπαιδευτικό θα ήταν ο παίκτης να λαμβάνει τον χειρισμό ενός μάγου σε τρίτο πρόσωπο, που προσπαθεί να μάθει ξόρκια για να επιλύσει γρίφους και να περάσει τα τέσσερα επίπεδα διαβαθμισμένης δυσκολίας. (όσο ανεβαίνει επίπεδο τόσο περισσότερο δυσκολεύει ο γρίφος και ο αλγόριθμος εκμάθησης που χρειάζεται να χρησιμοποιήσει ο μαθητής για να περάσει το επίπεδο). Χρησιμοποιήθηκε λοιπόν η (Python) ως γλώσσα προγραμματισμού και η (Pygame) ως μηχανή γραφικών για την δημιουργία του παιχνιδιού. Αρχικά, προσπάθησα να κάνω τα γραφικά του παιχνιδιού σε (OpenGL) αλλά υπήρξαν προβλήματα με την μνήμη της (GPU). Γι'αυτό χρησιμοποίησα την (Pygame) γιατί είχε ένα σχετικά απλό πακέτο εντολών, ήταν εύκολη στον χειρισμό και με βοήθησε να φτιάξω εύκολα και γρήγορα τα γραφικά δίχως να χρειαστώ (unity) ή (unreal). Το εν λόγω παιχνίδι είναι ένα εκπαιδευτικό και ψυχαγωγικό παιχνίδι που αποτελείται από τέσσερις πίστες. Σε κάθε πίστα ο παίκτης πρέπει να μάθει έναν αλγόριθμο και να επιλύσει μία συγκεκριμένη παραλλαγή του αλγορίθμου αυτού υπό την μορφή ξορκιού-γρίφου σε γλώσσα (Python)! Τα γραφικά και η επιλογή των αντικειμένων από το πρόγραμμα (Ζωγραφική (3D) των (Windows)) μετατράπηκαν σε εικόνες (PNG), διαφανείς). Με τον όρο διαφανείς εικόνες

εννοούνται εικόνες που υποστηρίζουν και κανάλι α διαφάνειας. Ο παίκτης χειρίζεται τον μάγο (avatar) σε τρίτο πρόσωπο ώστε να εστιάζει περισσότερο στην εκμάθηση του αλγορίθμου και στην επίλυση του γρίφου και όχι τόσο στην πρωτοπρόσωπη αφήγηση της ιστορίας. Επιλύοντας όλους τους γρίφους-ξόρκια όλων των πιστών με την χρήση των αλγορίθμων ο παίκτης τερματίζει το παιχνίδι. Όσον αφορά φυσικά την επιλογή των δεξιοτήτων του χαρακτήρα του παιχνιδιού υπό την ονομασία (wizard), πρόσεξα να συνάδουν απόλυτα με τον σκοπό του παιχνιδιού που δεν είναι κανένας άλλος παρά η εκμάθηση και η διδασκαλία αλγορίθμων. Ο παίκτης χρησιμοποιώντας τον χαρακτήρα μπορεί να σκοτώνει τέρατα με μαγεία, να μαζεύει νομίσματα για ζωή, να ξεκλειδώνει κουτιά με πληροφορίες και να επιλύει γρίφους γράφοντας σε πάπυρο. Για την εύρεση βοήθειας μπορεί να συμβουλευτεί το μαγικό βιβλίο που έχει την μορφή ψηφιακού βοηθού (chatbot). Ο παίκτης μπορεί να γράφει στο μαγικό βιβλίο τις απορίες του και άλλες σχετικές ερωτήσεις στις οποίες βρίσκει απαντήσεις αν δεν καταλαβαίνει το περιεχόμενο των κουτιών. Το μαγικό βιβλίο συνδέεται με (Chat_gpt_3.5_turbo API) και μπορεί να απαντήσει σε κάθε σχετική ερώτηση με το παιχνίδι γρήγορα και αποτελεσματικά διευκολύνοντας την εμπειρία του παίκτη με έναν ευφυή τρόπο. Επίσης, για να γίνει το παιχνίδι πιο ενδιαφέρον ο παίκτης διαθέτει από την αρχή του παιχνιδιού ζωή, μαγεία και υπολοιπούμενο χρόνο για να ολοκληρώσει την κάθε πίστα. Όταν επιλύει τον γρίφο-ξόρκι της μίας πίστας ο παίκτης περνά αμέσως στην επόμενη. Φυσικά όσο μεγαλώνει ο αριθμός της πίστας μεγαλώνει και η δυσκολία καθώς και οι απαιτήσεις του παιχνιδιού. Για να επιλύσει κάθε ξόρκι χρειάζεται να κατανοήσει καλά την λειτουργία του αλγορίθμου και να μπορέσει να αναπαράγει έναν παρόμοιο αλγόριθμο με βάση αυτό που του ζητείται. Συνεπώς για να μπορέσει ο παίκτης να ολοκληρώσει το παιχνίδι χρειάζεται να προσέχει τις οδηγίες των κουτιών, να απομνημονεύει και να κατανοεί τις πληροφορίες που του δίνονται χρησιμοποιώντας τις όλες συνδυαστικά για την επίλυση των γρίφων. Τέλος το μενού που αποτελεί την βάση όλου του παιχνιδιού, δίνει την δυνατότητα στον παίκτη να καταγράψει το όνομα του για να αποθηκευτεί το σκορ του σε κάθε πίστα που ολοκληρώνει. Επίσης το μενού προσφέρει έλεγχο της μουσικής του παιχνιδιού και των ήχων στα ειδικά εφέ (εκρήξεις, ήχοι τεράτων κλπ.), αλλαγής των πλήκτρων κίνησης του παίκτη και τέλος ένα κουμπί βοήθειας που επεξηγεί κάποιες κρυμμένες λειτουργίες. Όλες αυτές οι λειτουργίες του παιχνιδιού όπως η κίνηση του παίκτη, τις συγκρούσεις (collisions) των αντικειμένων και ευρύτερα ο τρόπος εμφάνισης του παιχνιδιού είναι καθαρά δικά μου δημιουργήματα βασισμένα σε θεωρίες παιχνιδιών που ακολουθούν παρακάτω και σε βιβλιογραφία του (Pygame).

3.3 ΣΥΝΟΨΗ ΜΕΘΟΔΟΛΟΓΙΑΣ ΚΑΙ ΑΞΙΟΛΟΓΗΣΗΣ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ

Το παιχνίδι δοκιμάστηκε από οχτώ μαθητές, τρεις μαθητές Α Γυμνασίου, τρεις μαθητές Α Λυκείου και δύο μαθητές Γ Λυκείου ώστε να αξιολογηθεί καλύτερα ως προς τον σκοπό του εφόσον είναι κατεξοχήν ένα εκπαιδευτικό παιχνίδι. Η αξιολόγηση έγινε μέσω ερωτηματολογίου, το οποίο συμπληρώθηκε μεμονωμένα για τον κάθε μαθητή και σε διαφορετικούς χρόνους ώστε να αποφευχθεί το σφάλμα της μεροληψίας. Ύστερα ζητήθηκε από τους μαθητές να σχολιάσουν προφορικά τις απαντήσεις τους με βάση τις δικές τους απόψεις σε μία μικρή “συνέντευξη” που διήρκεσε λίγα λεπτά. Θεώρησα πως ήταν καλό να δοκιμαστεί σε μαθητές διότι μέσω της ανατροφοδότησης από τα σχόλια τους θα μπορούσα να αξιολογήσω πιο αντικειμενικά εάν η αρχική ιδέα, οι αρχικοί στόχοι και σκοποί του παιχνιδιού είχαν υλοποιηθεί. (feedback) Ανάμεσα στις απαντήσεις που έλαβα προσμετρήθηκαν πέρα από τον πιο καταλυτικό παράγοντα δηλαδή τον βαθμό στον οποίο είχαν κατανοήσει τις βασικές έννοιες των αλγορίθμων, ο βαθμός ψυχαγωγίας και ο βαθμός του ενδιαφέροντος που εκδήλωσαν. Ευρύτερα το παιχνίδι έλαβε θετικές κριτικές. Η αξιολόγηση του παιχνιδιού θα εξεταστεί περαιτέρω στο πέμπτο κεφάλαιο. Οι μαθητές ενώ αρχικά δεν ήταν τόσο εξοικειωμένοι με τους αλγορίθμους ούτε με κάποια γλώσσα προγραμματισμού δήλωσαν ότι στο τέλος του παιχνιδιού είχαν πια μία διαφορετική αντίληψη γύρω από αυτές τις έννοιες. Από την άλλη κρίθηκαν τα γραφικά του παιχνιδιού, τα οποία θα προτιμούσαν να είναι πιο ζωντανά. Τους άρεσε η μουσική και η χρήση του ψηφιακού βοηθού διευκόλυνε την διαδικασία του παιχνιδιού. Γενικότερα, φάνηκαν ψυχαγωγημένοι και προσηλωμένοι στην επίλυση των ξορκιών καθώς και των διάφορων εμποδίων μέσα κατά την διάρκεια του παιχνιδιού που απαιτούσε την προσήλωση τους, την δεξιότητα απομνημόνευσης αλλά και την ικανότητα αντίληψης και συνδυασμού των αλγορίθμων αλλά και μαθηματικών πράξεων. Συνολικά, ήταν αρκετά ικανοποιημένοι με την εμπειρία τους αν και φάνηκε πως η καλή έως άριστη χρήση αγγλικών ήταν σημαντική προϋπόθεση καθώς και τα γραφικά θα μπορούσαν να ήταν πιο έντονα.

3.4 ΣΥΝΟΨΗ ΚΕΦΑΛΑΙΟΥ

Για την παρούσα εργασία επιλέχθηκε η κατασκευή ενός 2d εκπαιδευτικού ψηφιακού παιχνιδιού με την χρήση ενός ψηφιακού βοηθού. Στο κεφάλαιο αυτό αναλύθηκε η μεθοδολογία πίσω από την κατασκευή του παιχνιδιού. Δηλαδή η λογική του παιχνιδιού, τα μέσα και ο τρόπος υλοποίησής του. Συλλήβδην, επεξηγήθηκε ο τρόπος διεξαγωγής της αξιολόγησης του παιχνιδιού από τους μαθητές.

ΚΕΦΑΛΑΙΟ ΤΕΤΑΡΤΟ

Η ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΚΑΙ ΕΠΙΣΚΟΠΗΣΗ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ

4.1 ΕΙΣΑΓΩΓΗ

Στο κεφάλαιο της αρχιτεκτονικής θα αναλυθεί η ιστορία, η λογική του παιχνιδιού και ο σκοπός του παίχτη. Τέλος θα αναλυθεί πώς υλοποιήθηκε το παιχνίδι μέσω αυτών των κεντρικών ιδεών. Πιο συγκεκριμένα ο κάθε παίχτης έχει τον δικό του μάγο και πρέπει να ολοκληρώσει και τα 4 επίπεδα για να κατακτήσει την απόλυτη γνώση άλλων των ξορκιών! Αναλύεται η πορεία του παίχτη από την αρχή του κάθε επίπεδου (level), μέχρι και την ολοκλήρωσή του. Τέρατα, το μαγικό κλειδί, κλειδωμένα κουτιά με γνώση των ξορκιών είναι μερικά από τα εμπόδια που έχει να αντιμετωπίσει ο παίχτης για να φτάσει στην τελική πύλη του επιπέδου, δεν είναι όμως μόνος καθώς έχει και το μαγικό του βιβλίο! Αρχικά όμως θα αναλυθούν το μενού (menu) του παιχνιδιού καθώς και επιπρόσθετες λειτουργίες, όπως ο ήχος και η εικόνα του παιχνιδιού τα οποία βοηθούν με την σειρά τους για να επιτευχθεί το τελικό αποτέλεσμα.

4.2 ΙΣΤΟΡΙΑ

Η ιστορία ξεκινάει με τον παίχτη να βρίσκεται στην μέση μιας αρένας χωρίς να ξέρει η να θυμάται τίποτα. Ξαφνικά μέσα από το έδαφος μαγικά τέρατα ξεπροβάλλουν και αρχίζουν να τον κυνηγούν! Ύστερα από λίγο κυνηγείτο καταλαβαίνει ότι έχει την δύναμη να τα εξοντώνει πετώντας ξόρκια σας φωτιές από τα χέρια του. Ένα τέρας από αυτά του φανερώνει ένα μαγικό κλειδί. Το κλειδί αυτό ανοίγει τα μαγικά κουτιά μέσα στην αρένα και αυτά τον βοηθούν να μάθει να κάνει ξόρκια! Όμως δεν ξέρει ότι τα κουτιά ανοίγουν με την σειρά. Το τελευταίο μαγικό κουτί του λέει να βρει την πύλη που θα σημάνει την λήξη της εκπαίδευσής του! Επιλύοντας τα ξόρκια θα φτάσει στο τελικό επίπεδο (final level) όπου και θα καταλάβει το πραγματικό νόημα της εκπαίδευσής. Αυτός ο κόσμος του παιχνιδιού δεν είναι πάρα ένα συνονθύλευμα πολύ δυνατών ξορκιών, πολύ δυνατώτερων από ότι είχε ποτέ φανταστεί και είναι επιλογή του αν θα συνεχίσει την εξερεύνηση αυτού του μαγικού κόσμου ή όχι!

4.3 ΛΟΓΙΚΗ

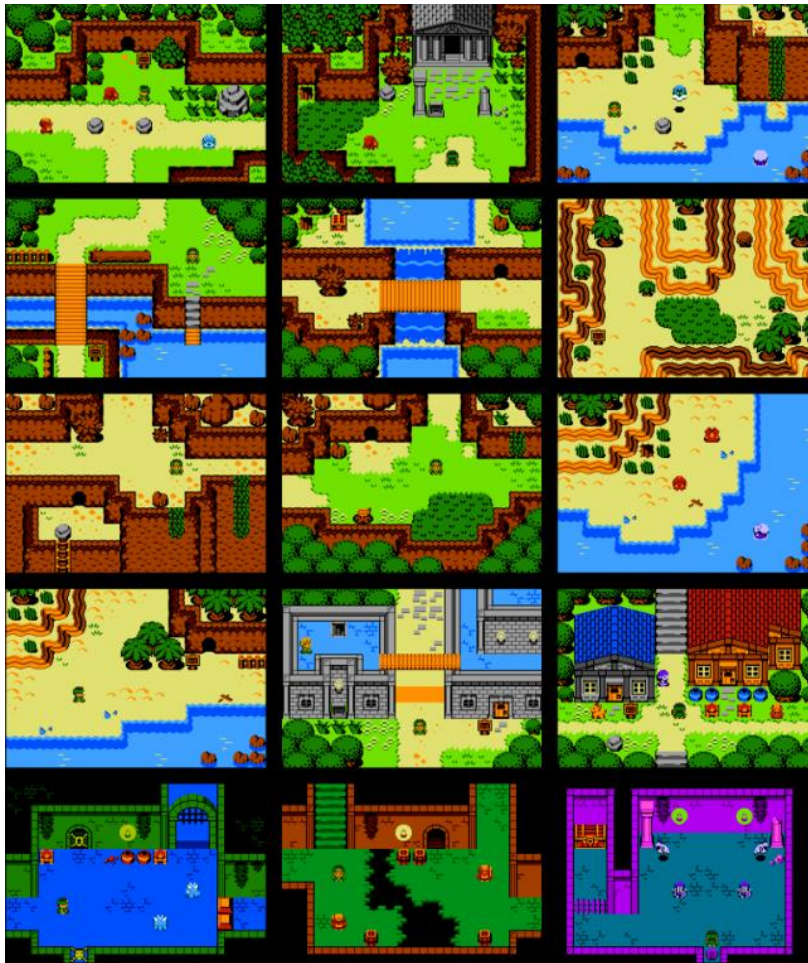
Η λογική του παιχνιδιού χωρίζεται στο κυρίως μενού στα επίπεδα και στο μαγικό βιβλίο βοηθό (assistant).

Στο μενού ο παίκτης επιλέγει το όνομα του μάγου του και ρυθμίζει τα πλήκτρα που θα κινεί τον μάγο κατά την διάρκεια του παιχνιδιού. Επίσης, μπορεί να ρυθμίσει την ένταση της μουσικής του παιχνιδιού και την ένταση των εφέ ήχου (sfx). Ακόμη, υπάρχει ένα προαιρετικό κουμπί (help) που προσφέρει στον παίκτη κάποιες επιπρόσθετες πληροφορίες. Στα επίπεδα ο παίκτης στοχεύει στο να μάθει να χρησιμοποιεί τέσσερις βασικούς αλγόριθμους σε (Python) για να ολοκληρώσει και τις τέσσερις πίστες. Παράλληλα, πρέπει να ξεφύγει από τα τέρατα που τον κυνηγούν ώστε να μην χάσει ζωή, να μαζέψει νομίσματα που του δίνουν έξτρα ζωή και να προλάβει να κάνει το ξόρκι εντός του χρόνου. Τέλος το μαγικό βιβλίο βοηθό (assistant), μπορεί να το συμβουλευτεί οποτεδήποτε θέλει ανεξάρτητα του σημείου στο οποίο βρίσκεται ο παίχτης μέσα στο παιχνίδι!

4.4 ΕΙΚΟΝΑ

Η εικόνα ως το πιο βασικό κομμάτι σε ένα παιχνίδι επιλέχθηκε να ακολουθεί το θέμα (theme) ενός διάσημου (2D platformer game), το παιχνίδι με όνομα “*The Legend of Zelda*”.

Η έμπνευση αυτή προήλθε από το παιχνίδι (THE LEGENT OF ZELDA), το οποίο έχει πολλές παρομοιώσεις με το συγκεκριμένο παιχνίδι.(βλέπε *Εικόνα 2*)



The Legend of Zelda illustration.

Εικόνα 2

Το μενού (menu), τα επίπεδα (levels), ακόμα και το βιβλίο βοηθός είναι εμπνευσμένα από αυτή την εποχή παιχνιδιών και πιο συγκεκριμένα από αυτό το παιχνίδι!

(Ποιο πολλά για το παιχνίδι στα παρακάτω links:

https://en.wikipedia.org/wiki/The_Legend_of_Zelda ,

<https://opengameart.org/content/zoria-tileset>)

4.5 ΠΡΟΕΡΓΑΣΙΑ ΚΑΙ ΔΟΜΙΚΑ ΣΤΟΙΧΕΙΑ ΠΟΛΥΜΕΣΩΝ

Τα πολυμέσα αποτελούν συνδυασμούς κειμένου , ήχου εικόνων, βίντεο και κινούμενης εικόνας (animation). Οι διαφορετικές μορφές πληροφοριών ονομάζονται **μέσα** (Τεχνολογίες Πολυμέσων: Θεωρία, Υλικό, Λογισμικό, Φώτης Λαζαρίνης,2017). Παρακάτω αναλύονται οι κατηγορίες που χρησιμοποιήθηκαν για το παιχνίδι, καθώς και η αλληλεπιδραστικότητα (interactivity) με τον χρήστη. Όλη η θεωρία είναι από το παραπάνω βιβλίο.

- Κείμενο
- Εικόνα
- Ήχος
- Κινούμενη εικόνα (animation)
- Αλληλεπιδραστικότητα (interactivity)

4.5.1 ΚΕΙΜΕΝΟ

Το κείμενο (text) αποτελεί το βασικότερο μέσο παρουσίασης πληροφοριών σε οποιαδήποτε εφαρμογή υπολογιστών. Το κείμενο αυτόνομο ή σε πλήκτρα εντολών ή χάρτες ή σε υπέρθεση σε βίντεο παίζει τον σημαντικότερο ρόλο στις εφαρμογές πολυμέσων. Η μορφή του κειμένου, η ποσότητα και η θέση του αποτελούν καθοριστικούς παράγοντες επιτυχίας και αποδοχής ενός τίτλου πολυμέσων. Μερικά από τα χαρακτηριστικά του κειμένου είναι το μέγεθος, το χρώμα, η γραμματοσειρά (font) και το στυλ. Το μέγεθος μετρείται σε στιγμές. Μια στιγμή κειμένου ισούται με το 1/72 της ίντσας. Τυπικές γραμματοσειρές είναι οι Times, Courier, Arial, Helvetica ενώ κάποια από τα στυλ είναι τα έντονα, τα πλάγια και τα υπογραμμισμένα. Οι γραμματοσειρές χωρίζονται σε δύο μεγάλες κατηγορίες τις **ψηφιογραφικές** ή **χαρτογραφικές** (bitmap) και τις **διανυσματικές** (vector). Στη πρώτη κατηγορία ανήκουν οι γραμματοσειρές System των Windows ενώ στην δεύτερη κατηγορία οι γραμματοσειρές TrueType που χρησιμοποιούνται στα Windows και στο Mac OS. (λειτουργικό των Macintosh). Αν και υπάρχει πληθώρα έτοιμων και δωρεάν γραμματοσειρών στο διαδίκτυο (πχ. <http://www.fontfoundry.com>, <http://www.larabiefonts.com>) είναι δυνατή και η κατασκευή νέων που θα ικανοποιούν τον δημιουργό τους καθώς και τις σχεδιαστικές απαιτήσεις κάποιας πολυμεσικής εφαρμογής. Για την δημιουργία νέων γραμματοσειρών υπάρχουν προγράμματα όπως ο ResEdit (Apple) και ο Fontographer (Macromedia).

4.5.2 ΕΙΚΟΝΑ

Από το βιβλίο (Τεχνολογίες Πολυμέσων: Θεωρία, Υλικό, Λογισμικό, Φώτης Λαζαρίνης,2017). Οι εικόνες συνυπάρχουν αρμονικά και απαραίτητα με το κείμενο είτε ως εναλλακτικός τρόπος παρουσίασης πληροφοριών είτε ως απτή αναπαράσταση κάποιου αντικειμένου η προγράμματος. Οι εικόνες μπορεί να είναι γραφικά (graphics), δηλαδή σχέδια που έχου δημιουργηθεί με τη βοήθεια κάποιου σχεδιαστικού προγράμματος ή φωτογραφίες που έχου εισαχθεί στον υπολογιστή με τη βοήθεια ενός σαρωτή ή μιας ψηφιακής φωτογραφικής μηχανής.

Τα γραφικά υπολογιστών χωρίζονται σε δύο κατηγορίες: τα **ψηφιογραφικά** ή **χαρτογραφικά** γραφικά (bitmap graphics) και τα **διανυσματικά** (vector graphics). Η πρώτη κατηγορία γραφικών αποτελείται από ένα σύνολο κουκκίδων που ονομάζονται **εικονοστοιχεία** (pixels), ενώ τα διανυσματικά γραφικά είναι συνδυασμοί γεωμετρικών σχημάτων τα οποία περιγράφονται με μαθηματικές συναρτήσεις. Οι εικόνες που εισάγονται στον υπολογιστή με τη βοήθεια κάποιας εξωτερικής συσκευής λήψης αποθηκεύονται ως γραφικά (bitmap), ενώ τα σχέδια που δημιουργούνται με τον υπολογιστή μπορούν να αποθηκεύονται ως γραφικά (bitmap) και να μπορούν να περιγράψουν με μαθηματικό τρόπο, τότε μπορούν και να αποθηκεύονται και ως διανυσματικά γραφικά.

4.5.3 ΗΧΟΣ

Ο ήχος (sound) προστίθεται στις εφαρμογές πολυμέσων για να προσθέσει έναν ευχάριστο μουσικό τόνο καθιστώντας τη χρήση της εφαρμογής πιο ευχάριστη ή για την παρουσίαση πληροφοριών σχετικά με τα παρουσιαζόμενα θέματα. Η ύπαρξη ηχητικών εφέ προσδίδει ρεαλισμό στα πολυμέσα και βοηθά στην εξοικείωση του χρήστη με το πρόγραμμα. Κάποιες φορές οι οδηγίες της εφαρμογής εκφωνούνται αντί μόνο γραπτής προβολής τους. Επιπλέον ο ήχος συνυπάρχει με τις εικόνες στη κινούμενη εικόνα (animation).

Συναντώνται δύο μορφές ήχου στον υπολογιστή: ο ήχος **κυματομορφών** ή **ψηφιακός** (wave ή digital) και το πρότυπο **Midi**. Η πρώτη μορφή χρησιμοποιείται για την αποθήκευση κάθε τύπου ήχου, π.χ. ανθρώπινης ομιλίας, μουσικής, ήχων περιβάλλοντος, ενώ στα αρχεία (midi) αποθηκεύεται μουσική που έχει δημιουργηθεί από μουσικά όργανα και έχει εισαχθεί στον υπολογιστή. Ο ψηφιακός ήχος εισάγεται στον υπολογιστή συνήθως μέσω μικροφώνου, το οποίο μετατρέπει τον ήχο από αναλογικό σε ψηφιακό.

4.5.4 ΚΙΝΟΥΜΕΝΗ ΕΙΚΟΝΑ (ANIMATION)

Οι κινούμενες εικόνες ή προσομοιώσεις κίνησης, όπως καλούνται συχνά συντίθενται με τη βοήθεια προγραμμάτων λογισμικού στον υπολογιστή. Ουσιαστικά πρόκειται για διαδοχικές παρόμοιες στατικές εικόνες 2 ή 3 διαστάσεων, που όταν προβληθούν η μία μετά την άλλη, εμφανίζονται με κίνηση. Για παράδειγμα αν οι διαδοχικές στατικές εικόνες που συνθέτουν μία κινούμενη εικόνα προβληθούν με την σειρά τότε το ανθρώπινο μάτι θεωρεί ότι βλέπει μία κίνηση ενός αντικειμένου να προχωρά και να επιστρέφει στο τέλος στην θέση που είχε αρχικά. Το ίδιο συμβαίνει ακριβώς και στην περίπτωση του βίντεο μόνο που εκεί διαδοχικές εικόνες είναι οι φωτογραφίες από τον πραγματικό κόσμο. Η τεχνική δημιουργίας κινούμενων εικόνων με την χρήση πολλών παρόμοιων διαδοχικών **καρέ** (frame) ονομάζεται cel animation και προήλθε από τα κινούμενα σχέδια της Disney. Με την χρήση των υπολογιστών αναπτύχθηκε μία νέα τεχνική η οποία καλείται **path animation**. Με την δεύτερη τεχνική ο σχεδιαστής δημιουργεί το αρχικό καρέ και μία γραμμή (ευθεία, καμπύλη ή τεθλασμένη) πάνω στην οποία θα γίνει η κίνηση. Το πρόγραμμα επεξεργασίας δημιουργεί τα επόμενα καρέ. Υπάρχουν πολλοί τύποι αρχείων για την αποθήκευση animation. Ένα από τα πιο γνωστά προγράμματα επεξεργασίας κινούμενης εικόνας είναι το 3D Studio Max, του οποίου τα αρχεία έχουν την κατάληξη max.

4.5.5 ΑΛΛΗΛΕΠΙΔΡΑΣΤΙΚΟΤΗΤΑ (INTERACTIVITY)

Αλληλεπιδραστικότητα ή διαδραστικότητα (interactivity) είναι η δυνατότητα που παρέχει μια εφαρμογή στον χρήστη να καθορίσει την εξέλιξη της. Είναι δηλαδή η δυνατότητα καθορισμού της ροής ενός προγράμματος, του τρόπου παρουσίασης της πληροφορίας, της επιλογής πληροφοριών για αναλυτικότερη παρουσίαση, της σειράς και της ταχύτητας εμφάνισης των θεμάτων που πραγματεύεται η εφαρμογή και την απάντηση σε ερωτήματα που θέτει η εφαρμογή προς τον χρήστη. Ουσιαστικά, είναι η δυνατότητα διαλόγου μεταξύ του χρήστη και της εφαρμογής. Γι' αυτό άλλωστε ο όρος interactivity μεταφράζεται συχνά και ως διαλογικότητα.

Τα παιχνίδια είναι αλληλεπιδραστικές εφαρμογές!

Οι παραπάνω κατηγορίες είναι αυτολεξείς από το βιβλίο (Τεχνολογίες Πολυμέσων: Θεωρία, Υλικό, Λογισμικό, Φώτης Λαζαρίνης, 2017) οι τρόποι αποθήκευσης εικόνων κειμένου και ήχων.

Με βάση αυτή την δομή, η λογική του παιχνιδιού χρησιμοποιεί τα παραπάνω με την βοήθεια του (Pygame module), για να δώσει ζωή στο παιχνίδι. Πιο συγκεκριμένα στο παιχνίδι χρησιμοποιήθηκε κείμενο και εικόνα με ψηφιογραφική μέθοδο., (bitmap), ψηφιακός ήχος (wav ή mp3) και κινούμενη εικόνα (cel animation).

4.5.6 ΤΡΟΠΟΙ ΑΠΟΘΗΚΕΥΣΗΣ ΣΤΟ ΠΑΙΧΝΙΔΙ

Για την αποθήκευση χρήσιμων πληροφοριών χρησιμοποιήθηκαν:

- Αρχεία εικόνας PNG.

Οι εικόνες PNG εικόνες που δημιουργούνται με το πρότυπο (Portable Network Graphics), είναι εναλλακτική κωδικοποίηση των εικόνων GIF και δημιουργήθηκε για το διαδίκτυο. Δεν χρησιμοποιείται πολύ, καθώς έχει κακό αλγόριθμο συμπίεσης. Επιλέχθηκαν αρχεία για το παιχνίδι αυτής της μορφής, καθαρά επειδή είναι πολύ διαδεδομένη μορφή και επίσης υποστηρίζει το κανάλι διαφάνειας για τα επάλληλα επίπεδα που έχουν τα site ή στην συγκεκριμένη περίπτωση το παιχνίδι.

(<http://www.libpng.org/pub/png/book/chapter08.html#png.ch08.div.5.8>)

- Αρχεία ήχου WAV και MP3

Τα αρχεία WAV και τα αρχεία MP3 είναι αρχεία ήχου. Τα WAV χρησιμοποιούνται από τα Windows κυρίως και έχουν καλούς ρυθμούς δειγματοληψίας. Τα MP3 είναι συμπιεσμένα αρχεία ήχου για μεγάλη διάρκεια ήχου και υποστηρίζουν (streaming). Για αυτό το λόγο στο παιχνίδι χρησιμοποιείται το MP3 για την μουσική και το WAV για ήχους μικρής διάρκειας. Η μηχανή πολυμέσων που χρησιμοποιήθηκε υποστηρίζει και τους δύο τύπους αρχείων, για αναπαραγωγή ήχου.

(Τεχνολογίες Πολυμέσων: Θεωρία, Υλικό, Λογισμικό, Φώτης Λαζαρίνης,2017)

- Αρχεία JSON

Όπως ορίζει ο παρακάτω σύνδεσμος:

“ Το JSON (JavaScript Object Notation) είναι μια ανοιχτή τυπική μορφή αρχείου για κοινή χρήση δεδομένων που χρησιμοποιεί κείμενο αναγνώσιμο από τον άνθρωπο για την αποθήκευση και τη μετάδοση δεδομένων. Τα αρχεία (JSON) αποθηκεύονται με την επέκταση .json. Το (JSON) απαιτεί λιγότερη μορφοποίηση και είναι μια καλή εναλλακτική για (XML). Το (JSON) προέρχεται από (JavaScript) αλλά είναι μια μορφή δεδομένων ανεξάρτητη από τη γλώσσα. Η δημιουργία και η ανάλυση του (JSON) υποστηρίζεται από πολλές σύγχρονες γλώσσες προγραμματισμού. (application/json) είναι ο τύπος μέσου που χρησιμοποιείται για το (JSON).” Ήταν λοιπόν επόμενο να επιλεγθεί από την (Python) για το παρών παιχνίδι καθώς έχει πολύ μικρό κώδικα διεπαφής, για εγγραφή παικτών και λειτουργίες, όπως εύρεση σε λίστα του (max score) του κάθε παίχτη χωρίς ιδική διαχείριση των δεδομένων.

(<https://docs.fileformat.com/web/json>)

4.5.7 ΤΙ ΕΙΝΑΙ ΕΝΑ MODULE ΣΤΗΝ PYTHON

Όπως ορίζει ο παρακάτω σύνδεσμος:

“ Μια λειτουργική μονάδα Python ή αλλιώς (module), είναι ένα αρχείο που περιέχει ορισμούς και δηλώσεις (Python). Μια ενότητα μπορεί να ορίσει συναρτήσεις, κλάσεις και μεταβλητές. Μια ενότητα μπορεί επίσης να περιλαμβάνει κώδικα με δυνατότητα εκτέλεσης. Η ομαδοποίηση σχετικού κώδικα σε μια ενότητα καθιστά τον κώδικα πιο κατανοητό και πιο εύκολο στη χρήση. Κάνει επίσης τον κώδικα λογικά οργανωμένο.” Απο εδώ και στο εξής για την καλύτερη κατανόηση των δομών του παιχνιδιού θα αναφερόμαστε τα (modules) όχι μόνο σαν λειτουργικές μονάδες κώδικα αλλά και σαν βιβλιοθήκες ή κομμάτια αυτών, διατηρώντας όμως το ορισμό.

(<https://www.geeksforgeeks.org/python-modules/>)

4.5.8 ΤΙ ΕΙΝΑΙ ΤΟ PYGAME MODULE

Όπως ορίζει ο παρακάτω σύνδεσμος:

“Το (Pygame) είναι ένα σύνολο λειτουργικών μονάδων (Python) που έχουν σχεδιαστεί για τη σύνταξη βιντεοπαιχνιδιών. Το (Pygame) προσθέτει λειτουργικότητα πάνω από την εξαιρετική βιβλιοθήκη (DirectMedia Layer **SDL**).”

(<https://www.pygame.org/wiki/about>)

(<https://www.libsdl.org/>)

4.5.8.1 ΒΑΣΙΚΗ ΔΟΜΗ ΠΑΡΑΘΥΡΟΥ ΓΡΑΦΙΚΩΝ ΣΤΟ PYGAME

Όπως ορίζει ο παρακάτω σύνδεσμος:

Για να φτιάξω αυτό το παιχνίδι 2D γραφικών διάλεξα την βιβλιοθήκη (Pygame) της (Python), η οποία είναι μια απλή βιβλιοθήκη πολυμέσων.

Για να μπορέσει ένα παιχνίδι να τρέξει μέσα στην (Python) με την βοήθεια του (Pygame), θα πρέπει να ακολουθηθεί η βασική ροή λειτουργίας για αυτό το (module). Αρχικά πρέπει να οριστεί ένα αντικείμενο τύπου (display) με αρχικές διαστάσεις στο οποίο θα τρέξει το παιχνίδι. Στην συνέχεια πρέπει να οριστεί το όνομα του παραθύρου του παιχνιδιού, καθώς και η τιμή για την ταχύτητα αναπαραγωγής του (frame rate). Μετά είναι δουλειά του προγραμματιστή να ορίσει αντικείμενα του (Pygame) για το παιχνίδι του. Τέλος χρειάζεται

έναν επαναληπτικό βρόχο που θα στεγάζει όλη την λειτουργία των αντικειμένων για την εκτέλεση του παιχνιδιού, καθώς και μια μεταβλητή που θα τερματίζει αυτόν τον βρόχο και θα κλείνει το παιχνίδι.

(<https://www.geeksforgeeks.org/how-to-set-up-the-game-loop-in-pygame/>)

```
import pygame

# Initialize Pygame
pygame.init()

# Define the width and height of the window
width, height = 800, 600

# Create a Pygame window with a title
window = pygame.display.set_mode((width, height))
pygame.display.set_caption("My Pygame Window")

# Game loop
running = True
while running:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = False

    # Clear the screen with a white background
    window.fill((255, 255, 255))

    # Add your game logic and drawing code here

    # Update the display
    pygame.display.flip()

# Quit Pygame
pygame.quit()
```

Main Loop in a PyGame window

Εικόνα 3

4.5.8.2 EVENTS ΣΤΟ PYGAME

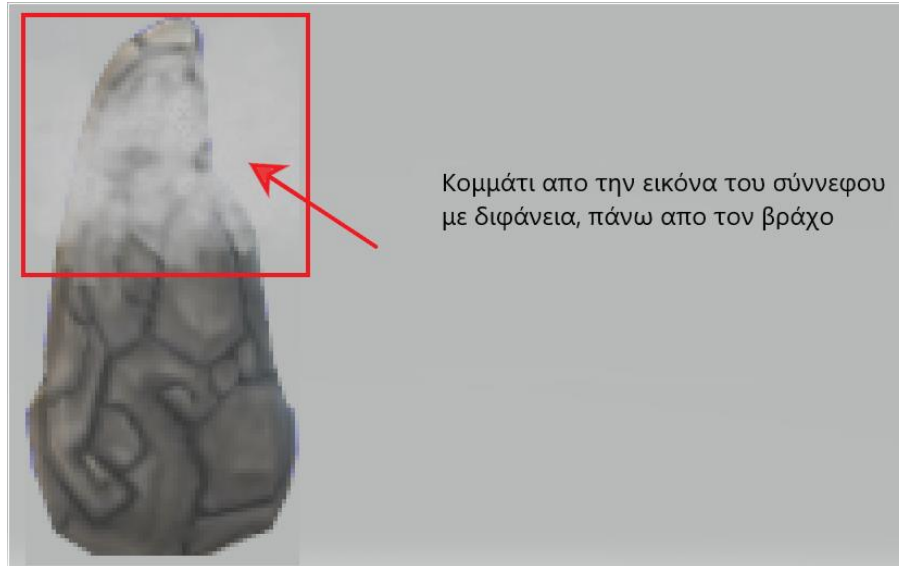
Στην πραγματικότητα μόνο ένα (event) είναι σωστό να τερματίζει ένα παιχνίδι η μία εφαρμογή διεπαφής. Έτσι και εδώ χρησιμοποιείται μέθοδος του (Pygame), (pygame.event.get()), για χειρισμό των πλήκτρων του ποντικιού και γενικά οποιασδήποτε άλλης περιφερικής εισόδου. Αυτό θα επιτρέψει τον τερματισμό στην προκειμένη περίπτωση του παιχνιδιού αλλά και την δημιουργία μεταβλητής που θα αποθηκεύει τα (events) του παραθύρου για μετέπειτα χρήση τους. Είναι σημαντικό να σημειωθεί, ότι τα (events) σε ένα παράθυρο είναι ορατά μέχρι και την στιγμή που καθαρίζει το (display), (pygame.display.flip()). (βλέπε *Εικόνα 3*).

(Χωρίς τα (events) ένα παιχνίδι δεν είναι διαδραστικό, καθώς ο παίχτης δεν έχει κανέναν έλεγχο του παραθύρου!)

4.5.8.3 ΕΠΙΦΑΝΕΙΕΣ ΣΤΟ PYGAME

Δεν θα είχε πολύ νόημα να φτιάξουμε ένα παράθυρο γραφικών για ένα παιχνίδι χωρίς να υπάρχουν γραφικά.

Στο (Pygame) οι εικόνες και τα (animations) γίνονται με τη μέθοδο της επιφάνειας (surface). Με την βοήθεια της κλάσης ορθογωνίου (Rect), ένα πεδίο ορίζεται μέσα στο χώρο της οθόνης με αρχική θέση, τη πάνω αριστερά κρυφή του του ορθογωνίου αυτού και διαστάσεις πλάτους και ύψους. Μέσα εκεί μία επιφάνεια μπορεί να υπάρχει. Αυτή η επιφάνεια μπορεί να φορτώνει εικόνες και να αλλάζει τις ιδιότητες της με χρήση άλλων μεθόδων όπως της μεγέθυνσης ή της αλλαγής χρώματος. Μπορεί ακόμα και να αντικατασταθεί με μία άλλη. Ίσως το πιο σημαντικό από όλα είναι η δυνατότητα διαφάνειας! Το κανάλι άλφα (alpha channel) μαζί με τα υπάρχοντα κανάλια της εικόνας (Red, Green, Blue), δημιουργούν το εφέ της διαφάνειας. Τώρα αν ο προγραμματιστής επιθυμεί να έχει ένα σύννεφο πάνω από μία εικόνα βράχου, μπορεί να θέσει το κανάλι άλφα (alpha channel), σε μια τιμή που είναι κατάλληλη έτσι ώστε να δίνεται η εντύπωση πως το σύννεφο είναι διαφανές και περνάει πάνω από τον βράχο! (βλέπε *Εικόνα 4*)



Επικάλυψη επίπεδων (RGBA) εικόνων (bitmap)

Εικόνα 4

Οι εικόνες που πρέπει να φορτωθούν αρχικά είναι PNG με την μέθοδο `pygame.image.load()` και εναποθέτονται τελικά πάνω στην βασική επιφάνεια του (`display`) με την μέθοδο (`image.blit()`). Η μέθοδος αυτή λειτουργεί για όλες τις επιφάνειες προς όλες τις επιφάνειες!

ΤΙ ΕΙΝΑΙ ΤΟ RGB ΠΡΟΤΥΠΟ ΚΑΙ ΤΟ ΚΑΝΑΛΙ ΑΛΦΑ (*ALPHA CHANNEL*)

Τα τέσσερα κανάλια στην κωδικοποίηση **RGBA32** Τα κανάλια είναι διατεταγμένα στη μνήμη με τέτοιο τρόπο ώστε ένας απλός ακέραιος 32-bit χωρίς πρόσημο να έχει το δείγμα του κόκκινου καναλιού στην αρχή, μετά το κανάλι του πράσινου καναλιού, στα το κανάλι του μπλέ καναλιού και τέλος το κανάλι αλφα με μεγέθη καναλιων στα (8 bit). Όπως φαίνεται και παρακάτω (βλέπε *Εικόνα 5*).

Sample Length:	8								8								8								8							
Channel Membership:	Red								Green								Blue								Alpha							
Bit Number:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Little Endian RGBA32 bit

Εικόνα 5

(https://en.wikipedia.org/wiki/RGBA_color_model)

Η χρήση του όρου άλφα εξηγείται από τον Smith ως εξής:

Το ονομάσαμε έτσι λόγω του κλασικού τύπου γραμμικής παρεμβολής

$$\text{Classic linear interpolation formula } \alpha A + (1-\alpha)B$$

που χρησιμοποιεί το ελληνικό γράμμα α (άλφα) για τον έλεγχο του ποσού της παρεμβολής μεταξύ, σε αυτήν την περίπτωση, δύο εικόνων A και B. Δηλαδή, κατά τη σύνθεση της εικόνας A στην κορυφή της εικόνας B, η τιμή του. Το α στον τύπο λαμβάνεται απευθείας από το κανάλι άλφα του A.

(https://en.wikipedia.org/wiki/Alpha_compositing)

Το μοντέλο RGBA32 βασίζεται σε ένα παλιότερο χρωματικό μοντέλο εικόνας με όνομα RGB ή αλλιώς μοντέλο κόκκινου πράσινου και μπλε χρώματος (Red, Green, Blue model), 24 (bit)! (Τεχνολογίες Πολυμέσων: Θεωρία, Υλικό, Λογισμικό, Φώτης Λαζαρίνης, 2017).

4.5.8.4 ΚΕΙΜΕΝΟ ΣΤΟ PYGMAE (*PYGAME FONTS*)

Από ένα παιχνίδι όπως και από κάθε εφαρμογή πολυμέσων δεν μπορεί να λείπει το κείμενο, καθώς είναι η κύρια μορφή επικοινωνία πληροφορίας με τον παίχτη!

Η βιβλιοθήκη του (Pygame) παρέχει μεθόδους για την απεικόνιση κειμένου σε διαφορετικά (fonts) μεγέθη και χρώματα. Η κλάση που περιλαμβάνει όλες αυτές τις λειτουργίες είναι η κλάση (Font). Με την εντολή (pygame.font.Font()), δημιουργείται ένα (font) με τις κατάλληλες προδιαγραφές. Μπορεί ακόμα και να φορτωθούν μη υπάρχοντα (fonts) με την κατάληξη (.tff). Δεν χρησιμοποιήθηκαν στο συγκεκριμένο παιχνίδι. Τέλος με την εντολή (font.render()) το κείμενο με το συγκεκριμένο φόντο (font), εναποθέτεται πάνω σε μια επιφάνεια γραφικών. Προσοχή, η εικόνα είναι πλέον αλλαγμένη και περιέχει το κείμενο!

4.5.8.5 ΗΧΟΙ ΣΤΟ PYGAME

Τέλος όπως σε κάθε βιβλιοθήκη πολυμέσων έτσι και εδώ οι ήχοι καλύπτονται εξολοκλήρου από το (Pygame). Ο προγραμματιστής το μόνο που έχει να κάνει είναι να χρησιμοποιήσει άλλη μια κλάση της βιβλιοθήκης, την κλάση (pygame.mixer.Sound()), για να φορτώσει το αρχείο ήχου και να το επεξεργαστεί μετέπειτα. Στο παιχνίδι οι μόνες λειτουργίες ήχου είναι το ξεκίνημα του ήχου, η επανάληψη του, ο ορισμός της έντασης και η διακοπή του. Οι μέθοδοι για τα παρακάτω είναι οι εξής: (Εστω υποθετικό αντικείμενο ήχου (sound = pygame.mixer.Sound(sound_path))

- Εκκίνηση ήχου : (sound.play())
- Ορισμός του ήχου σε επανάληψη : (sound.play(loops = loops))
- Ορισμός έντασης : (sound.set_volume(value)), με το (value) σε τιμές από 0 έως 255.
- Διακοπή ήχου : (sound.stop())

(Παραπάνω δεν αναφέρονται, πάρα μόνοι οι βασικές μέθοδοι για την αναπαραγωγή ήχου στη βιβλιοθήκη του (Pygame))

Με αυτόν τον τρόπο πολλοί ήχοι μπορούν να παίζουν και να σταματάνε παράλληλα χωρίς να χρειάζεται κάποια άλλη ιδιαίτερη μεταχείριση, καθώς το (Pygame) κανονίζει για τις εσωτερικές λειτουργίες στην κάρτα ήχου του υπολογιστή.

(Βοηθητικές κλάσεις και ευρετήρια καθώς και ονόματα αρχείων για την λειτουργία του παιχνιδιού και την αποθήκευση χρήσιμων πληροφοριών, ακόμα και βασικών λειτουργιών όπως απεικόνιση πολλών επιπέδων, εκτέλεση πολλαπλών ήχων και επιλογή αντικειμένου με τον κέρσορα, θα δοθούν στο τέλος της αρχιτεκτονικής)

(Περισσότερα για την λειτουργία της Python και του Pygame στα παρακάτω links:

<https://www.python.org/about/gettingstarted> , <https://www.pygame.org>)

4.6 ΤΟ ΚΥΡΙΩΣ ΜΕΝΟΥ

Το κυρίως μενού για να ταιριάζει με το θέμα (theme) του παιχνιδιού σχεδιάστηκε με την λογική των παλιών παιχνιδιών που χρησιμοποιούσαν κυρίως την λογική του γραμμικού μενού (linear menu). Αυτό γινόταν για λόγους επίδοσης, καθώς ήταν ο πιο ευθύς τρόπος να αναπαρασταθούν οι επιλογές του παίχτη. Έτσι ο παίχτης περιοριζόταν μόνο σε ότι τον ενδιέφερε για να ξεκινήσει γρήγορα ένα νέο παιχνίδι. Αυτό δεν κούραζε συνεπώς ούτε τους προγραμματιστές στο έργο τους, αλλά ούτε και τα μηχανήματα με άσκοπες λειτουργίες. Σήμερα δεν υπάρχει τέτοιο πρόβλημα, όμως κάποιες εταιρίες θέλουν να διατηρούν το θέμα (theme) του παιχνιδιού τους ίδιο με τις αρχικές εκδόσεις, για να κρατάνε την ποιότητα τους σταθερή, προσελκύοντας νέους αγοραστές αλλά διατηρώντας και τους παλιούς.



The Legend of Zelda: Breath of the Wild (Menu)

Εικόνα 6

(https://en.wikipedia.org/wiki/The_Legend_of_Zelda:_Breath_of_the_Wild)

Στο παιχνίδι (THE LEGEND OF ZELDA: Breath of the Wild), (βλέπε Εικόνα 6) φαίνεται να ακολουθείται αυτή η στρατηγική. Με βάση λοιπόν αυτήν την λογική έχει δημιουργηθεί και το μενού του συγκεκριμένου παιχνιδιού. Όπως φαίνεται και παρακάτω (βλέπε Εικόνα 7).



Κυρίως μενού παιχνιδιού

Εικόνα 7

4.6.1 ΕΜΦΑΝΙΣΗ

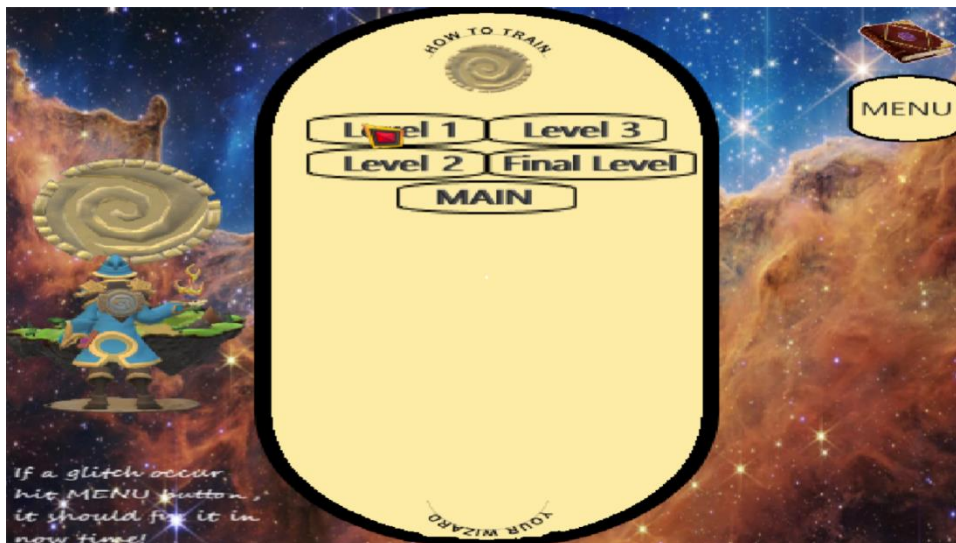
Εμφανίζεται στο κέντρο της οθόνης και εμπεριέχει μέσα του τα κουμπιά των επίπεδων των πλήκτρων του παίχτη του ήχου του παιχνιδιού καθώς και το κουμπί βοήθειας. Τέλος μια κινούμενη λίστα στο κάτω μέρος του μενού δείχνει συνεχώς τους παίχτες και τα επίπεδα που έχουν ολοκληρώσει. Πάνω δεξιά βρίσκεται το κουμπί για την επιστροφή στο μενού από οποιοδήποτε επίπεδο.

4.6.2 ΛΕΙΤΟΥΡΓΙΑ

Όταν ο παίχτης επιλέξει το πεδίο (YOUR WIZARD'S NAME) θα μπορέσει να δώσει το όνομά του στον μάγο για να ξέρει το παιχνίδι την εξέλιξη αυτού του παίχτη 'μέσα στο παιχνίδι(Οι νέοι παίχτες ξεκινούν από το επίπεδο 1!). Στην συνέχεια ο παίχτης μπορεί να πατήσει το κουμπί (LEVELS) για να κατευθυνθεί στο μενού με τα επίπεδα του παιχνιδιού. Αν θέλει να αλλάξει τον τρόπο με τον οποίο θα χειρίζεται τον παίχτη μπορεί να επιλέξει τον κουμπί (KEYS) που θα τον κατεύθυνση στο μενού για αλλαγή κουμπιών κίνησης του μάγου. Αν τώρα ο παίχτης θέλει να επιλέξει, αν θα ακούγονται ή όχι οι ήχοι του παιχνιδιού μπορεί να πατήσει το κουμπί (SOUND) και να κατευθυνθεί στο μενού έντασης μουσικής και των ιδικών εφέ ήχου του παιχνιδιού. Τέλος για ότι παραπάνω απορίες μπορεί να έχει θα κατευθυνθεί στο μενού βοήθειας μέσω του κουμπιού (HELP). Το κουμπί (MENU) είναι ενεργό μόνο όταν ο παίχτης παίζει μέσα στο επίπεδο για επιστροφή στο κυρίως μενού!

4.6.3 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ

Αποτελείται από την κύρια κλάση (Menu) η οποία είναι υπεύθυνη για τον χειρισμό του μενού της εναλλαγής των πλήκτρων για την κίνηση του παίχτη, του μενού της εκκίνησης των επιπέδων και του μενού για τον έλεγχο του ήχου όλου του παιχνιδιού. Αρχικά το μενού δημιουργεί αντικείμενα της βοηθητικής κλάσης τύπου (Object) για τα κουμπιά και τα περιγράμματα του κυρίως μενού, άλλα και για όλα τα δευτερεύοντα μενού που έχουν προαναφερθεί, ακόμα και για την κινούμενη λίστα με τους παίχτες. Καθώς και για το κουμπί επιστροφής από το επίπεδο πίσω στο κυρίως μενού. Όταν ο παίχτης επιλέξει το κουμπί επιπέδων, τότε απλά εμφανίζονται όλα τα προϋπάρχοντα αντικείμενα που αφορούν αυτό το μενού.



Μενού επιπέδων

Εικόνα 8

4.7 ΕΜΦΑΝΙΣΗ ΤΟ ΜΕΝΟΥ ΕΠΙΠΕΔΩΝ

Το μενού επιπέδων έχει παρόμοια όψη με το κυρίως μενού(βλέπε *Εικόνα 8*), με την μόνη διαφορά ότι λείπει η λίστα με τους παίκτες και στην θέση της δεν υπάρχει τίποτα. Τα πλήκτρα πλέων γράφουν (LEVEL 1, LEVEL 2, LEVEL 3, FINAL LEVEL) και υπάρχει το πλήκτρο επιστροφής στο κύριο μενού (MAIN).

4.7.1 ΛΕΙΤΟΥΡΓΕΙΑ ΤΟΥ ΜΕΝΟΥ ΤΩΝ ΕΠΙΠΕΔΩΝ

Αν ο παίχτης διαλέξει ένα επίπεδο π.χ. το επίπεδο 3 και δεν έχει φτάσει με την πρόοδο του μέσα στο παιχνίδι σε αυτό. Τότε θα εμφανιστεί μήνυμα στον κέρσορα του ότι αυτό το επίπεδο είναι κλειδωμένο(βλέπε *Εικόνα 9*). Αν πάλι επιλέξει ένα επίπεδο που είναι ανοιχτό τότε θα ξεκινήσει το επίπεδο αυτό από την αρχή. Τέλος αν επιλέξει το πλήκτρο (MAIN) θα επιστρέψει πίσω στο αρχικό μενού.



Υπόδειξη κλειδωμένου επιπέδου

Εικόνα 9

4.7.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΕΙΑΣ ΤΟΥ ΜΕΝΟΥ ΤΩΝ ΕΠΙΠΕΔΩΝ

Η κλάση του κυρίως μενού έχει το αντικείμενο (`selected_level = None`), όταν ο παίχτης επιλέξει ένα επίπεδο το μενού είναι υπεύθυνο να καλέσει το βοηθητικό αντικείμενο το οποίο είναι υπεύθυνο για την καταγραφή όλων των παικτών, τύπου (`ExampleClass`) και να το ρωτήσει αν ο παίχτης έχει ξεκλειδώσει με επιτυχία το επίπεδο. Αυτό γίνεται μέσω μιας μεταβλητής (`max_level`) η οποία παίρνει τον μέγιστο αριθμό επιπέδου που βρίσκεται αυτός ο παίχτης αυτή την χρονική στιγμή. Αν το (`max_level`) είναι μεγαλύτερο η ίσο με την πίστα που επέλεξε ο χτίστης τότε, το (`selected_level`) φορτώνει το επίπεδο επιλογής π.χ. 3 με την βοήθεια της κλάσης (`Level`). Αν τώρα αποτύχει η ερώτηση το μενού ενημερώνει το αντικείμενο του κέρσορα το οποίο είναι και αυτό τύπου (`Object`) να αλλάξει την εικόνα του στην εικόνα που δηλώνει ότι το επίπεδο είναι κλειδωμένο. Η εικόνα του κέρσορα επαναφέρεται στην αρχική με την επιλογή οποιουδήποτε άλλου κουμπιού από το μενού.

4.8 ΕΜΦΑΝΙΣΗ ΤΟΥ ΜΕΝΟΥ ΕΠΙΛΟΓΗΣ ΠΛΗΚΤΡΩΝ

Το μενού επιλογής των πλήκτρων δεν μοιάζει με κανένα άλλο μενού λόγω της απλότητας του (βλέπε *Εικόνα 9*). Τα μόνα κουμπιά που υπάρχουν είναι, το κουμπί εναλλαγής για την κίνηση του παίχτη και το πλήκτρο επιστροφής στο κύριο μενού (MAIN). Είναι σημαντικό το γεγονός ότι το κουμπί εναλλαγής δείχνει τα επιλεγμένα πλήκτρα κίνησης του παίχτη με ένα μόνο πάτημα σε αυτό!(βλέπε *Εικόνες 10, 11*)



Μενού εναλλαγής πλήκτρων για την κίνηση του παίκτη

Εικόνα 10

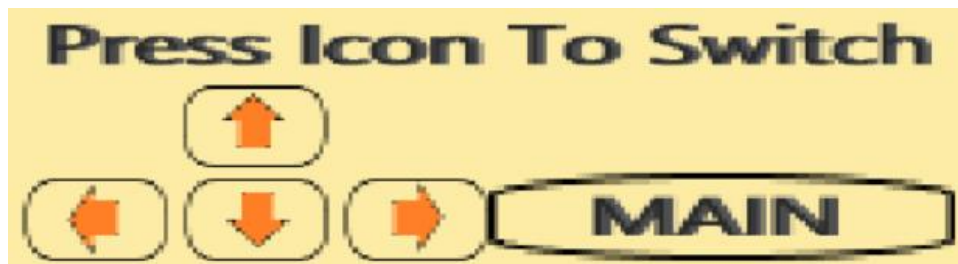
4.8.1 ΛΕΙΤΟΥΡΓΕΙΑ ΤΟΥ ΜΕΝΟΥ ΕΠΙΛΟΓΗΣ ΠΛΗΚΤΡΩΝ

Ο παίχτης με όταν θα επιλέξει το εικονίδιο με τα πλήκτρα, αυτό θα αλλάξει από την επιλογή χειρισμού του παίκτη με τα πλήκτρα (W, A, S, D), στην επιλογή χειρισμού του παίκτη με τα βέλη. (βλέπε Εικόνες 11, 12)



Εναλλαγή του εικονιδίου στα πλήκτρα (W, A, S, D)

Εικόνα 11



Εναλλαγή του εικονιδίου σε βέλη

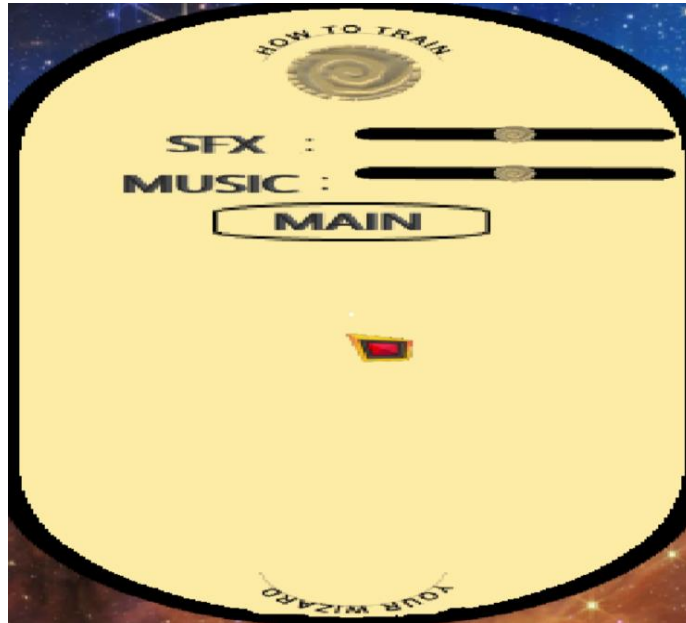
Εικόνα 12

4.8.2 ΔΟΜΗ ΤΗΣ ΛΕΙΤΟΥΡΓΕΙΑ ΤΟΥ ΜΕΝΟΥ ΕΠΙΛΟΓΗΣ ΠΛΗΚΤΡΩΝ

Η κλάση του κυρίως μενού έχει την μεταβλητή (bool_wasd_arrows) όπου και αποθηκεύεται η κατάσταση των πλήκτρων σε αληθής ή ψευδής (True / False). Η μεταβλητή αυτή θα χρησιμοποιηθεί αργότερα από την κλάση του παίκτη για να δει αν θα κινηθεί με με τα κουμπιά αυτά η με τα βέλη. Η αρχική κατάσταση είναι πάντα η επιλογή των πλήκτρων (W, A, S, D) και αυτή η πληροφορία δεν αποθηκεύεται με το κλείσιμο το παιχνιδιού για τον κάθε παίκτη.

4.9 ΕΜΦΑΝΙΣΗ ΤΟΥ ΜΕΝΟΥ ΕΝΤΑΣΗΣ ΤΟΥ ΗΧΟΥ

Το μενού έντασης ήχου έχει δύο επιλογείς για τις τιμές ήχου. Ο πρώτος επιλογέας είναι για την προσαρμογή της έντασης στα εφέ ήχου του παιχνιδιού και γενικότερα όποιου ήχου δεν είναι η μουσική του παιχνιδιού. Ο δεύτερος επιλογέας είναι καθαρά και μόνο για την προσαρμογή της έντασης της μουσικής του παιχνιδιού. Τέλος το μενού διαθέτει και αυτό πλήκτρο για επιστροφή στο κυρίως μενού.(βλέπε *Εικόνα 13*)

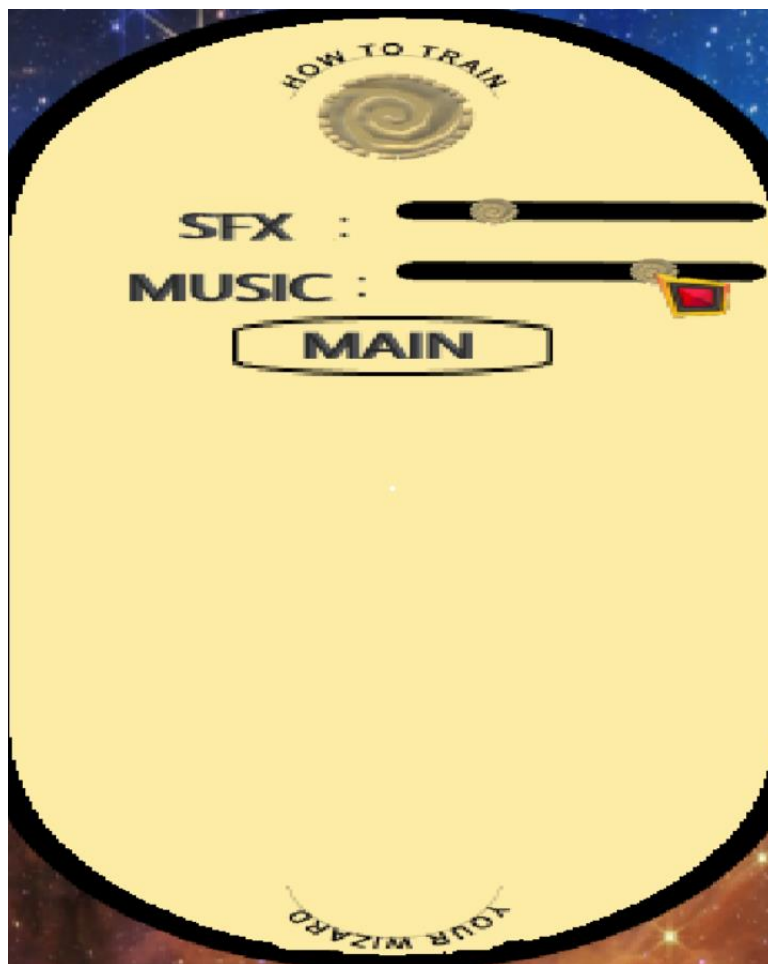


Μενού έντασης του ήχου

Εικόνα 13

4.9.1 ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΜΕΝΟΥ ΕΝΤΑΣΗΣ ΤΟΥ ΗΧΟΥ

Ο παίχτης μπορεί να προσαρμόσει την ένταση του ήχου στα ειδικά εφέ και την ένταση του ήχου της μουσικής του παιχνιδιού επιλέγοντας την θέση του κέρματος με τον κέρσορα που σε τιμές έντασης μεταξύ 0 και 100 %.(βλέπε *Εικόνα 14*)



Επιλογή έντασης ήχου με τον κέρσορα

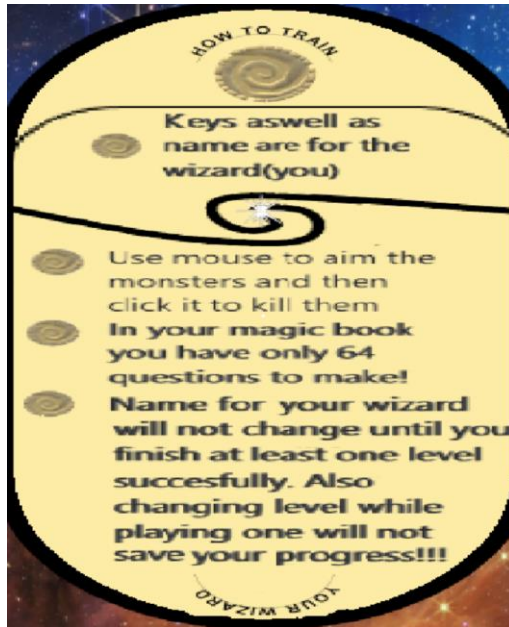
Εικόνα 14

4.9.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΜΕΝΟΥ ΕΝΤΑΣΗΣ ΤΟΥ ΗΧΟΥ

Με κάθε επιλογή που κάνει ο παίκτης για την νέα ένταση του ήχου μέσω του κέρσορα δύο μεταβλητές της κλάσης του κυρίως μενού με ονόματα (`sfx_volume`) και (`music_volume`) παίρνουν τιμές από 0 έως 255, έτσι ώστε να θέσουν και αυτές με την σειρά τους τιμή ήχου στα βοηθητικά αντικείμενα της κλάσης (`SoundPlayer`). Αυτά τα αντικείμενα είναι υπεύθυνα για τον ήχο του παιχνιδιού και έχουν παρόμοια δομή μεθόδων με τις παραπάνω μεθόδους για τον χειρισμό του ήχου στην βιβλιοθήκη του (`Pygame`).

4.10 ΕΜΦΑΝΙΣΗ ΤΟΥ ΜΕΝΟΥ ΒΟΗΘΕΙΑΣ (HELP MENU)

Το τελευταίο μενού του παιχνιδιού είναι το μενού την βοήθειας. Μέσα του υπάρχει μια μεγάλη εικόνα με σημειώσεις που βοηθούν τον παίκτη να καταλάβει κάποια περεταίρω λειτουργικά κομμάτια του παιχνιδιού. Το κουμπί επιστροφής στι κυρίως μενού δεν υπάρχει εδώ, καθώς όταν ο παίκτης έχει ενημερωθεί μπορεί να πατήσει το ίδιο το κείμενο της βοήθειας και να επιστρέψει στο κυρίως μενού.(βλέπε *Εικόνα 15*)



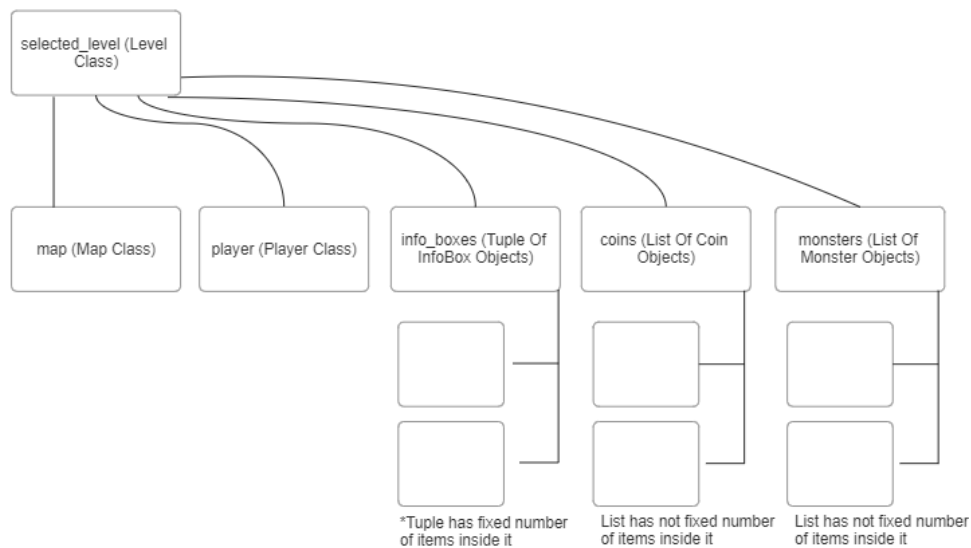
Μενού βοήθειας του παίκτη

Εικόνα 15

(Για το παραπάνω μενού δεν υπάρχει κάποια ιδιαίτερη λειτουργία. Επίσης η κλάση (Menu) δεν συμπεριλαμβάνεται αυτούσια, καθώς έχει πάρα πολλές λειτουργίες και αντικείμενα, πράγμα περιττό για το κεφάλαιο της αρχιτεκτονικής. Παρόλα αυτά βασικές λειτουργίες της κλάσης θα συμπεριληφθούν παρακάτω.)

4.11 ΤΟ ΕΠΙΠΕΔΟ

Το επίπεδο είναι το κύριο κομμάτι του παιχνιδιού. Στεγάζει τα περισσότερα αντικείμενα του παιχνιδιού και τις περισσότερες λειτουργίες σε ένα μόνο αντικείμενο τύπου (Level)! Επιλέχτηκε αυτή η λογική, καθώς σε αντίθεση με του κυρίως μενού και την κλάση (Menu) όπου και στεγάζεται η κλάση (Level) μέσω του αντικειμένου (selected_level), η επικοινωνία με τα εσωτερικά στοιχεία της κλάσης θα είναι πολύ πιο συχνή και τα αντικείμενα θα είναι καλύτερα προσβάσιμα όταν υπάρχουν μέσα στην ίδια την κλάση μαζί με τις λειτουργίες τους. Παρακάτω ακολουθεί το διάγραμμα κλάσεων(βλέπε *Εικόνα 16*) που στεγάζονται από το αντικείμενο (Level) και κάτω.



Δομή της κλάσης με τα κύρια αντικείμενα που περιέχει

Εικόνα 16

(Το παρών δενδρόγραμμα δημιουργήθηκε με την βοήθεια του παρακάτω link:

<https://app.smartdraw.com/index.aspx?nsu=1>)

4.11.1 ΕΜΦΑΝΙΣΗ

Το επίπεδο εμφανίζεται σε όλη την οθόνη (display), με κέντρο το μάγο μέσα στο χάρτη (map), να περιτριγυρίζεται από διάφορα αντικείμενα, τα αντικείμενα του ίδιου του χάρτη που δεν αλληλοεπιδρά άμεσα και αντικείμενα που μπορεί να αλληλοεπιδράσει άμεσα, όπως τέρατα κέρματα κουτιά.(βλέπε Εικόνα 17)



Τυπική όψη του επιπέδου με την κάμερα σε τρίτο πρόσωπο

Εικόνα 17

4.11.2 ΛΕΙΤΟΥΡΓΙΑ

Όταν ο παίκτης επιλέξει ένα επίπεδο αυτό φορτώνει και όταν ανοίξει ο παίκτης μπορεί να χειριστεί τον μάγο του με τα πλήκτρα που έχει επιλέξει. Σκοπός του όπως έχει προαναφερθεί, είναι να σκοτώσει τα τέρατα, να βρει το μαγικό κλειδί και να ανοίξει με την σωστή σειρά τα κουτιά. Τέλος αφού διαβάσει το γρίφο πρέπει να βρει την μαγική πύλη και να λύσει το γρίφο με ένα μαγικό ξόρκι στην γλώσσα (Python). Όταν ολοκληρωθεί το επίπεδο ένα μήνυμα νικητή εμφανίζεται στην οθόνη και επιστρέφει στο κυρίως μενού. Αν για κακή του τύχη ο παίκτης πεθάνει στην προπόνηση ένα άλλο μήνυμα ήττας θα εμφανιστεί στην οθόνη.

4.11.3 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ

Όταν ο παίκτης επιλέξει πίστα τότε το αντικείμενο από το κυρίως μενού με όνομα (selected_level) αρχίζει και δημιουργείται. Εκείνη την στιγμή αρχικοποιούνται τα παραπάνω αντικείμενα που περιέχει το επίπεδο είτε σε λίστες είτε μόνα τους όπως ο παίκτης. Το μενού είναι υπεύθυνο για να δώσει αρχικά χαρακτηριστικά στο επίπεδο με βάση τον αριθμό του. Αν το επίπεδο πάρει την τιμή (selected_level_num = 1) της παραβλητής της κλάσης (Menu), τότε γνωρίζει ποιο χάρτη να φορτώσει ποιος θα είναι ο αριθμός του μέγιστου πλήθους τεράτων (max_monster_num), καθώς και ποιος θα είναι ο αρχικός αμετάβλητος πίνακας κουτιών (info_boxes) με τα κουτιά ορισμένα σε σωστές θέσεις και με σωστό περιεχόμενο με βάση το επίπεδο. Τέλος το επίπεδο θα καθορίσει την αρχική μορφή της λίστας των νομισμάτων (coins) καθώς και την ακριβή τους θέση. Είναι σημαντικό να σημειωθεί ότι στην αρχικοποίηση του επιπέδου ο παίκτης είναι πάντα στην ίδια θέση όπως και το τελικό κουτί για τον γρίφο. Καθώς αυτή η λογική συνάδει με την εικόνα που υπάρχει στο έδαφος του παιχνιδιού. (Παρακάτω ακολουθεί εικόνα με τα βασικά αντικείμενα του κατασκευαστή της κλάσης (Level), καθώς και ο τρόπος δημιουργίας του από το αρχικό μενού, βλέπε Εικόνες 18, 19)

```
def __init__(self, name, level_num, init_camera_pos, menu, all_objects):
    self.menu = menu

    self.name = name

    self.level_num = level_num - 1

    self.old_menu_state = self.menu.background.get_active_state()

    self.map = mp.Map(self.name, self.level_num, self.menu, all_objects)
    self.camera_pos = init_camera_pos

    self.player = pl.Player(self.menu, self, self.map, 0.1, (4, 13), all_objects)

    self.max_monster_num = 4 * level_num
    self.monsters = []

    self.magic_key = True
    self.MK = None

    self.info_boxes = None
    if level_num == 1:
        self.info_boxes = (
            i_b.InfoBox(self.menu, self.map, self.player, 1, '11b0', (1, 4), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 2, '11b1', (32 - 4, 4), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 3, '11b2', (32 - 4, 8), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 4, '11b3', (32 - 4, 12), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 5, '11b4', (32 - 4, 16), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 6, '11b5', (32 - 4, 20), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 7, '11b6', (32 - 8, 20), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 8, '11b7', (32 - 12, 20), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 9, '11b8', (32 - 12, 24), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 10, '11b9', (32 - 12, 28), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 11, '11b10', (32 - 8, 28), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 12, '11b11', (32 - 4, 28), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 13, '11b12', (32 - 4, 30), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 14, '11b13', (32 - 8, 30), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 15, '11b14', (32 - 12, 30), all_objects),
            i_b.InfoBox(self.menu, self.map, self.player, 16, '11riddle', (1, 32 - 4), all_objects)
        )

        i_b.InfoBox(self.menu, self.map, self.player, 17, 'genera_box', (13, 1), all_objects)

    self.magic_gate_riddle = g_r.Riddle(self.menu, self, self.map, self.player, init_pos = (14, 0.5), riddle_string = riddle_level_1, all_objects = all_objects)
    self.coins = [
        cn.Coin(self.menu, self.map, (20, 4), all_objects = all_objects),
        cn.Coin(self.menu, self.map, (20, 8), all_objects = all_objects),
        cn.Coin(self.menu, self.map, (20, 12), all_objects = all_objects),
        cn.Coin(self.menu, self.map, (20, 16), all_objects = all_objects)
    ]
```

Δημιουργία του αντικειμένου του χάρτη (map)

Δημιουργία του παίκτη (player)

Δημιουργία της λίστας των τεράτων (monsters) και αρχικοποίηση του μέγιστου αριθμού τους

Δημιουργία του (tuple) με όλες τις δημιουργίες των αντικειμένων των κουτιών για την πίστα 1 και όλα τα απαραίτητα αρχικά δεδομένα

Δημιουργία λίστας νομισμάτων με όλα τα νομίσματα αρχικά δημιουργημένα

Δημιουργία βασικών αντικειμένων στον κατασκευαστή της κλάσης (Level)

Εικόνα 18

```

elif self.menu == 'levels':
    self.focus = False
    if coll_obj == self.level_1:
        self.background.play_sound()

        mouse_image.set_cur_frame_num(0)

        selected_level_num = 0
        if not (self.selected_level is None):
            selected_level_num = self.selected_level.get_num()
        if self.selected_level is None or selected_level_num != 1:
            if not (self.selected_level is None):
                self.selected_level.remove_from_all_objects()
            self.selected_level = lv.Level('level_1', 1, (16, 16), self, self.background.all_objects)
        self.close()
        self.selected_level.change_active_state()

```

An to μενού τώρα έχει όνομα 'levels', δηλαδή βρίσκεται ο χρήστης στο μενού των επιπέδων και πατήσει με τον κέρσορα το αντικείμενο 'κουμπί επιπέδου 1'
 An δεν υπάρχει φορτωμένο επίπεδο 'is None' ή αν υπάρχει άλλο επίπεδο φορτωμένο π.χ. 2.
 Φότρώσε ή δημιούργησε το επίπεδο 1
 Ορισμός αριθμού επιπέδου (level_num) της κλάσης (Level)

Δημιουργία του επιπ. 1 στην μεταβλητή (selected_level)

Εικόνα 19

Αρχικά δεν υπάρχουν τέρατα μέσα στο επίπεδο αλλά πετάγονται μέσα από τις τρύπες του χάρτη σε τυχαίες στιγμές και σε τυχαία σειρά. Υπεύθυνη και βασική συνάρτηση για αυτή την κλάση και τις εσωτερικές κλάσεις της είναι η μέθοδος (run()) που ακολουθεί σε εικόνα παρακάτω (βλέπε Εικόνες 20, 21). Η μέθοδος (run()) της κλάσης (Level) τρέχει μόνο στο κυρίως μενού όταν αυτό απενεργοποιηθεί και το επιτρέπει. Δηλαδή το παιχνίδι τρέχει όταν κλείνει το μενού.

```

def run(self, mouse_image, game_events):
    if not self.menu.background.get_active_state():
        if not self.magic_gate_riddle.if_close_and_unlocked_to_see():
            self.map.run(self.camera_pos)

            dims = self.player.image.render_obj.get_cur_frame().get_dims()
            rel_pos = self.player.image.get_map_rel_pos()
            self.set_level_camera_pos((-rel_pos[0] * 32 + 128 - dims[0] * 0.5, -rel_pos[1] * 16 + 169 + dims[1] * 0.5))
            # player's motion by level but it must be in player's class
            # self.player.image.change_map_rel_pos(self.player.set_dir_from_map_col((random.random() * 0.2 - 0.1, random.random() * 0.2 - 0.1)))

            if rd.random() > 0.95:
                self.add_monster()

```

Ορισμός θέσης μάγου και θέσης κάμερας
 Πιθανότητα να προστεθεί νέα δημιουργία τέρατος στην λίστα των τεράτων

Βασικό κομμάτι λειτουργίας της μεθόδου (run()) του αντικειμένου (Level)

Εικόνα 20

```

else:
    for event in game_events:
        if event.type == pygame.MOUSEBUTTONDOWN:
            # Get the mouse position when a mouse button is pressed
            x, y = mouse_image.get_pos()
            # print(f"Mouse X: {x}, Mouse Y: {y}")
            coll_obj = self.background.all_objects.check_all((x, y))
            ...

            # This was the original P for puse action but is now the new muse action on small menu icon
            for event in game_events:
                if event.type == pygame.KEYDOWN:
                    if event.key == pygame.K_p:
                        ...

            if coll_obj == self.small_menu:
                self.focus = False
                self.open()
                self.selected_level.change_active_state()

                print('All objects num. in all object list:')
                print(len(self.background.all_objects.list_used) + len(self.background.all_objects.list_not_used))

            self.selected_level.run(mouse_image, game_events)

```

Το μενού όταν είναι κλειστό το ποντίκι για το αν θα ξαναανοίξει πατώντας το εικονίδιο (MENU)
 Το επίπεδο τρέχει μέσα στο κυρίως μενού όσο αυτό είναι κλειστό

Το επίπεδο τρέχει μέσα στο κυρίως μενού όσο αυτό είναι κλειστό

Εικόνα 21

Η μέθοδος (run()) στεγάζει όλες τις μεθόδους (run()) των κλάσεων που εμπεριέχει τις λειτουργίες όλων των αντικειμένων στο επίπεδο και είναι υπεύθυνη για την εκτέλεση της ροής του παιχνιδιού όταν το κυρίως μενού είναι ανενεργό. Μέχρι τώρα έχει αναλυθεί μόνο ανά κομμάτι αυτής. Αφού οριστούν όλα τα αντικείμενα και οι λειτουργίες τους μέσα στο επίπεδο θα αναφερθεί πάλι η μέθοδος (run()) διεξοδικά. Καθώς είναι η βασική μέθοδος και για τον τερματισμό ενός επιπέδου!

4.12 ΕΜΦΑΝΙΣΗ ΤΟΥ ΠΑΙΚΤΗ (ΜΑΓΟΥ)

Ο παίκτης έχει την εμφάνιση του ενός μάγου ντυμένο στα μπλέ. Μέσα στο παιχνίδι ο παίκτης που τον κινεί μπορεί δει όλη την όψη του. Ο μάγος εμφανίζεται να πετάει φωτιές από τα χέρια και να κινείται ελεύθερα μέσα στο χάρτη. Πάνω αριστερά στην οθόνη υπάρχει η ζωή τη μαγεία και ο χρόνος του μάγου.(βλέπε Εικόνα 22)



Όψη απο τη λειτουργία των βοηθητικών αντικειμένων (assets) του μάγου

Εικόνα 22

4.12.1 ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΠΑΙΚΤΗ (ΜΑΓΟΥ)

Ο μάγος μπορεί να κινείται πάνω κάτω δεξιά αριστερά και διαγώνια με βάση τα πλήκτρα που πατάει ο παίκτης, τα οποία έχει ορίσει ο ίδιος στο κυρίως μενού. Με τον κέρσορα μπορεί να δείξει την κατεύθυνση στην οποία θέλει να πετάξει τις μαγικές φωτιές, το μόνο που χρειάζεται να κάνει ο παίκτης για να ρίξει είναι να πατήσει το κέρσορα. Ο μάγος αν συναντηθεί με ένα τέρας πρέπει να το αποφύγει πριν να είναι πολύ κοντά, γιατί αλλιώς θα τον σκοτώσει! Τα κέρματα είναι το μόνο πράγμα που μπορεί να του δώσει ζωή! Δεν είναι όμως το μόνο που πρέπει να φοβάται. Αν ο χρόνος τελειώσει πάλι αποτυγχάνει η εκπαίδευση. Καλό θα ήταν να προσέχει και πόση μαγεία του έχει απομείνει.

4.12.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΠΑΙΚΤΗ (ΜΑΓΟΥ)

Ο μάγος δημιουργείται με την δημιουργία του αντικειμένου (Level) και ως το αντικείμενο (player). Απρκα από το βοηθητικό αρχείο (all_payer) φορτώνονται άλλες οι εικόνες για το παίκτη, και ορίζονται μεταβλητές και αντικείμενα όπως ήχοι του παίκτη, η αρχική ζωή, η μαγεία ο χρόνος που υπολείπεται για την εξάσκηση στο επίπεδο καθώς και τα (animation) του μάγου και η ταχύτητα αναπαραγωγής τους. Είναι πολύ σημαντικό να αναφερθεί η μονάδα χρόνου για όλα τα (animation) μέσα στο παιχνίδι του 1ος (frame), όλοι οι χρόνοι (animation) από εδώ και πέρα θα είναι ακέραια πολλαπλάσια αυτής της μονάδας! Βασική δομή μέσα στον παίκτη είναι η λίστα με αντικείμενα φωτιάς (Fier) με τα οποία μπορεί να σκοτώνει τα τέρατα! Με την δική του μέθοδο (run()) ο μάγος μπορεί να κινείται και να χτυπάει τα τέρατα. Παρακάτω ακολουθεί το βασικό κομμάτι του κατασκευαστή για τον μάγο.(βλέπεΕικόνες 23, 24)

```
def __init__(self, menu, level, map, speed, init_pos, all_objects):
    self.master_level = level

    self.image = dy_obj.DynamicObject(menu, map, 'player', init_pos, 0, order_frames = (
        'ri_mo',
        'ri_up_mo_0',
        'ri_up_mo_1',
        'up_mo_0',
        'up_mo_1',
        'up_le_mo_0',
        'up_le_mo_1',
        'le_mo',
        'le_do_mo_0',
        'le_do_mo_1',
        'do_mo_0',
        'do_mo_1',
        'do_ri_mo_0',
        'do_ri_mo_1',
        'ri_st',
        'ri_up_st',
        'up_st',
        'up_le_st',
        'le_st',
        'le_do_st',
        'do_st',
        'do_ri_st'
    ),
    order_animations = (
        (14, 0, 14, 0),
        (21, 12, 21, 13),
        (20, 10, 20, 11),
        (19, 8, 19, 9),
        (18, 7, 18, 7),
        (17, 5, 17, 6),
        (16, 3, 16, 4),
        (15, 1, 15, 2)
    ), order_sounds = ('steps', 'hit'), all_objects = all_objects)

    self.image.set_frame_time(4)
    self.sound cont_playing = self.sound cont_playing_old = False
    self.fiers = []
```

Οι εικόνες του παίκτη για το (animation) ορίζονται κατευθείαν μέσα στο βοηθητικό αντικείμενο (image)ως η εικόνα του. Απο κάτω είναι πίνακας με πίνακες (tuples) με αριθμούς για την σειρά των εικόνων (animation)

Ορισμός μεταβλητής για την ταχύτητα αναπαραγωγής του (animation)

Δημιουργία άδειας λίσταςγιανακρατήσει αντικείμενα φωτιάς

Δημιουργία βασικών αντικειμένων στον κατασκευαστή της κλάσης (Player)

Εικόνα 23

```

self.life = 100 # life gets up with coins only!
self.magic = 100 # magic goes up by itself
self.time_remaining = 100

self.show_life = obj.Object()
self.show_life_container = obj.Object(name = 'life', init_pos = (0, 0), init_pos_z = 0.98, order_frames = ('life'), all_objects = all_objects)
self.show_life_container.change_active_state()
self.show_life = obj.Object(name = 'life', init_pos = (11, 0), init_pos_z = 0.98, order_frames = (
    'blank',
    'coin 0'
), all_objects = all_objects)
self.show_life.get_cur_frame().scale((11, 2))
self.show_life.change_active_state()

# self.show_magic = obj.Object()
self.show_magic_container = obj.Object(name = 'magic', init_pos = (0, 48), init_pos_z = 0.98, order_frames = ('magic'), all_objects = all_objects)
self.show_magic_container.change_active_state()
self.show_magic = obj.Object(name = 'magic', init_pos = (11, 48), init_pos_z = 0.98, order_frames = (
    'blank',
    'elixir'
), all_objects = all_objects)
self.show_magic.get_cur_frame().scale((11, 2))
self.show_magic.change_active_state()

# self.show_time = obj.Object()
self.show_time_container = obj.Object(name = 'time', init_pos = (0, 96), init_pos_z = 0.98, order_frames = ('time'), all_objects = all_objects)
self.show_time_container.change_active_state()
self.show_time = obj.Object(name = 'time', init_pos = (11, 96), init_pos_z = 0.98, order_frames = (
    'blank',
    'clock'
), all_objects = all_objects)
self.show_time.get_cur_frame().scale((11, 2))
self.show_time.change_active_state()

Initialize a dictionary to keep track of key states
self.key_states = {
    pygame.K_d: False,
    pygame.K_s: False,
    pygame.K_a: False,
    pygame.K_w: False,
    pygame.K_RIGHT: False,
    pygame.K_DOWN: False,
    pygame.K_LEFT: False,
    pygame.K_UP: False
}

```

← Αρχικοποίηση ζωής, μαγίας και υπολειπόμενου χρόνου στο 100 %!

Δημιουργία βοηθητικών αντικειμένων για την απεικόνιση ζωής, μαγίας και χρόνου

Δημιουργία και αρχικοποίηση πίνακα με πλήκτρα του (Pygame) για την ταυτόχρονη επιλογή πλήκτρων κατά την διαγωνία κίνηση του παίκτη

Άλλα βασικά αντικείμενα στον κατασκευαστή της κλάσης (Player)

Εικόνα 24

Παρακάτω ακολουθεί εικόνα από την μέθοδο (run()) του μάγου καθώς και επεξήγηση των εσωτερικών βασικών συναρτήσεων.(βλέπε Εικόνα 25)

```

def run(self, mouse_image, game_events):
    self.image.run()
    self.move_on_key_input(game_events)
    self.add_tier(mouse_image, game_events)
    for fier in self.fiers:
        fier.run()
    self.blit_all()
    # self.show_life.run()
    self.get_key()
    self.unlock()
    self.read_box_info()
    self.get_coin()
    self.maximize_magic()
    self.minimize_time()

```

Κίνηση παίκτη και άλλα

Δημιουργήσε μια φωτιά

Τρέξε όλες τις φωτιές του μάγου

Απεικόνιση της ζωής, της μαγίας και του χρόνου

Πάρε το κέρμα

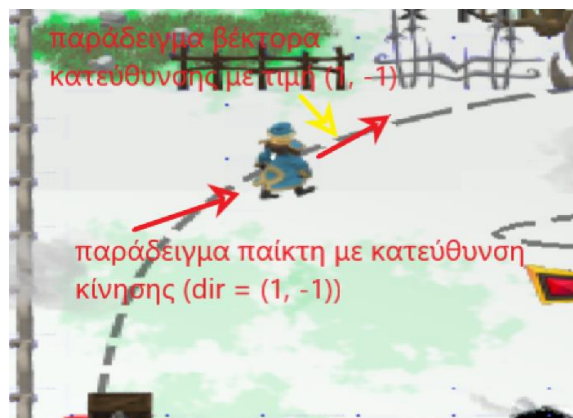
Αύξησε την μαγεία

Μείωσε το χρόνο!

Δομή της μεθόδου (run()) για το αντικείμενο (Player)

Εικόνα 25

Αρχικά η μέθοδος (`move_on_key_input()`) θέτει τις μεταβλητές που βρίσκονται μέσα στον πίνακα (`key_states`), σε αληθή ή ψευδή ακούγοντας στο ίδιο αντικείμενο που ακούει ακόμα και το κυρίως μενού για την εκτέλεση εντολών με τον κέρσορα ή το πληκτρολόγιο το αντικείμενο (`game_events`) τύπου (`event`), το συγκεκριμένο είναι αντικείμενο της βιβλιοθήκης (`Pygame`) και έχει αναφερθεί ξανά στην προεργασία με την μορφή (`pygame.event.get()`). Όταν καταγραφούν τα πατημένα κουμπιά ένας μικρός πίνακας τύπου δύο τιμών κατεύθυνσης ($x = 0, y = 0$) δημιουργείται. Ονομάζεται (`dir`) και είναι βοηθητική μεταβλητή. Μετά γίνεται μία πράξη πρόσθεσης που υπολογίζει την τελική κατεύθυνση ανάλογα με το πόσα κουμπιά έχουν πατηθεί. Επίσης λαμβάνει υπόψη την μεταβλητή από το κυρίως μενού, αν δηλαδή πατιόνται τα πλήκτρα (W, A, S, D) ή τα πλήκτρα για να από φασίσει με ποια από τις δύο κατηγορίες θα κάνει την πράξη της κατεύθυνσης. Τέλος και ίσως το πιο σημαντικό είναι η πράξη της μεθόδου από τη βοηθητικό αντικείμενο (`image.change_map_rel_pos()`) όπου υπολογίζει τις συγκρούσεις με τα στατικά αντικείμενα του χάρτη. (Ανάλυση του τρόπου απεικόνισης των στατικών και δυναμικών αντικειμένων, επεξεργασία ήχου, καθώς και των συγκρούσεων και της δομής βασικών κλάσεων, υπάρχει στο τέλος της αρχιτεκτονικής, μαζί με επεξήγηση στο τι σημαίνει στατικό αντικείμενο και δυναμικό στο παρών παιχνίδι) Μετά από αυτή την διεργασία, αν δεν είναι μηδενική η κατεύθυνση παίζει τον ήχο περπατήματος το παίκτη. Για να μην αναπαράγει τον ήχο του περπατήματος σε κάθε (`frame`) και υπερφορτωθούν τα βοηθητικά αντικείμενα του ήχου (`SoundPlayer`) που εμπεριέχονται στο αντικείμενο (`image`). Η εκκίνηση και το σταμάτημα του ήχου γίνονται μόνο στην αλλαγή της κατεύθυνσης από (0, 0) σε οτιδήποτε άλλο, το (`animation`) του παίκτη πάλι είναι ενεργό μόνο όταν η κατεύθυνση του παίκτη δεν είναι μηδενική. Παρακάτω βρίσκεται ο σύνδεσμος για την εύρεση κατεύθυνσης με πολλαπλή επιλογή πλήκτρων. (βλέπε Εικόνες 26, 27)



Όψη κατεύθυνσης κίνησης του παίκτη

Εικόνα 26


```

def move on key input(self, game_events):
    for event in game_events:
        if event.type == pygame.KEYDOWN:
            if event.key in self.key_states:
                self.key_states[event.key] = True
            elif event.type == pygame.KEYUP:
                if event.key in self.key_states:
                    self.key_states[event.key] = False
        # Calculate the direction based on key states
        dir = (0, 0)
        if self.menu.bool_wasd_arrows:
            if self.key_states[pygame.K_d]:
                dir = (dir[0] + 1, dir[1])
            if self.key_states[pygame.K_s]:
                dir = (dir[0], dir[1] + 1)
            if self.key_states[pygame.K_a]:
                dir = (dir[0] - 1, dir[1])
            if self.key_states[pygame.K_w]:
                dir = (dir[0], dir[1] - 1)
        else: # Use arrow keys instead of WASD
            if self.key_states[pygame.K_RIGHT]:
                dir = (dir[0] + 1, dir[1])
            if self.key_states[pygame.K_DOWN]:
                dir = (dir[0], dir[1] + 1)
            if self.key_states[pygame.K_LEFT]:
                dir = (dir[0] - 1, dir[1])
            if self.key_states[pygame.K_UP]:
                dir = (dir[0], dir[1] - 1)

self.image.change_map_rel_pos(self.image.set_dir_from_map_col((dir[0] * self.speed, dir[1] * self.speed)))
# Play sound and stop it, without overlay
# Reset keys to False if there is a direction without key_pressed
if dir[0] != 0 or dir[1] != 0:
    self.sound_cont_playing = True
    # self.old_dir = dir
else:
    self.sound_cont_playing = False

if self.sound_cont_playing != self.sound_cont_playing_old:
    if self.sound_cont_playing == True:
        self.image.render_obj.play_sound()
    else:
        self.image.render_obj.stop_sound()

self.sound_cont_playing_old = self.sound_cont_playing

self.animation on key input(dir)

```

Ενεργοποίηση πλήκτρων κίνησης ακούγοντας τα (events) απο την μεταβλητή με (events) πληκτρολογίου και κέρσορα, του αντικειμένου (game_events)

Υπολογισμός της κατεύθυνσης πριν απο τις συγκρούσεις με την βοηθητική μεταβλητή (dir) τυπου (tuple) πίνακα Τελικός υπολογισμός μαζί με τις συγκρούσεις απο τα στατικά αντικείμενα του αντικειμένου (map) χάρτη μέσω μεθόδου του αντικειμένου (image)

Επιλογή για την αναπαραγωγή ήχου με κίνηση ή χωρίς

Αναπαραγωγή των (animation) του μαγού με βλαση την κατεύθυνση

Μέθοδος της κίνησης με βάση τα πλήκτρα και το περιβάλλον του μάγου

Εικόνα 27

(<https://www.techwithtim.net/tutorials/game-development-with-python/pygame-tutorial/pygame-tutorial-movement>)

Για να μπορέσει ο μάγος να προσθέσει μία φωτιά καλείται η μέθοδος του με όνομα (add_fier()), αυτή προσθέτει στην ουσία ένα νέο αντικείμενο τύπου φωτιάς (Fier) στην λίστα του, με αρχική κατεύθυνση την κατεύθυνση που δείχνει ο κέρσορας. Ο Υπολογισμός γίνεται εξάγοντας το ημίτονο και το συνημίτονο με γεωμετρικό τρόπο από την διαφορά απόστασης μάγου κέρσορα και στογκυλοποιώντας για να έχει και η φωτιά απλή κατεύθυνση της μορφής που έχει και ο παίκτης π.χ.(1, 1) διαγώνια κάτω.(βλέπε Εικόνα 26) Τέλος η αρχική θέση αυτής της νέας φλόγας είναι η θέση του μάγου αυτή την χρονική στιγμή που δημιουργήθηκε και μειώνεται η μαγεία κατά 10 μονάδες.(βλέπε Εικόνα 28)

```

# This was the original F for player's fier but it is better with mouse klick!
if event.type == pygame.KEYDOWN:
    if event.key == pygame.K_f:
        ...
        mx, my = mouse_image.get_pos()
        # because the player is at the center of screen it could be done with p_x, p_y = player.image.render_obj.get_pos(). It is almost equal with
        dx = mx - 256
        dy = my - 256
        radius = (dx * dx + dy * dy) ** 0.5
        mouse_dir = (round(dx / abs(radius)), round(dy / abs(radius)))
        self.fiers.append(fier.Fier(self.menu, self.map, self, mouse_dir, self.image.rel_pos, self.image.render_obj.all_objects))
        self.fiers[-1].change_active_state()
        self.fiers[-1].image.render_obj.play_sound()
        self.minimize_magic()

```

Υπολογισμός αρχικής κατεύθυνσης φλόγας

Ορισμός αρχικής κατεύθυνσης φλόγας με την μεταβλητή (mouse_dir)

Αφαίρεση 10 μονάδων από την μαγεία του μάγου

Κομμάτι μεθόδου της πρόσθεσης φλόγας

Εικόνα 28

Παρακάτω στην (run()) του μάγου(βλέπε Εικόνα 25), υπάρχει μία δομή επανάληψης που τρέχει την μέθοδο (run()), από όλες τις φλόγες. Η συγκεκριμένη μέθοδος θα αναλυθεί στο αντικείμενο (Fier). Ακολουθεί η μέθοδος (blit_all) η οποία σχεδιάζει σε κάθε (frame) τα βοηθητικά αντικείμενα της ζωής της μαγείας και του χρόνου. Παρακάτω φαίνεται πώς ακριβώς επιτυγχάνεται αυτό.(βλέπε Εικόνα 29)

```

def blit_all(self):
    # life Χρήση βοηθητικής μεθόδου (blit)
    self.show_life.get_cur_frame().clear_texture()
    for counter in range(int(self.life / 10)):
        self.show_life.blit((1, 0), (counter * 8, 0))

    # magic της κλάσης τύπου (DynamicObject)
    self.show_magic.get_cur_frame().clear_texture()
    for counter in range(int(self.magic / 10)):
        self.show_magic.blit((1, 0), (counter * 8, 0))

    # time χρόνο καθώς και των περιγραμμάτων τους
    self.show_time.get_cur_frame().clear_texture()
    for counter in range(int(self.time_remaining / 10)):
        self.show_time.blit((1, 0), (counter * 8, 0))

```

Μέθοδος της απεικόνισης της ζωής, της μαγείας και του χρόνου

Εικόνα 29

Τέλος η μέθοδος που θα αναλυθεί διεξοδικά στο αντικείμενο (Player) είναι η μέθοδος (get_coin()). Η συγκεκριμένη μέθοδος αφαιρεί από την λίστα των κερμάτων της κλάσης (Menu), το κέρμα το οποίο βρίσκεται σε επαφή με τον παίκτη. Αν η ζωή του παίκτη είναι μικρότερη από 100 μονάδες τότε το κέρμα αφαιρείται από την λίστα και η ζωή του παίκτη αυξάνεται κατά 10 μονάδες, παίζοντας τον χαρακτηριστικό ήχο του κέρματος. Παρακάτω ακολουθεί αναλυτική επεξήγηση.(βλέπε Εικόνα 30)

```
def get_coin(self):
    for coin in self.master_level.coins:
        if self.image.get_dynamic_dist(coin.image, (True)) < 1 and self.life < 100:
            coin.image.render_obj.play_sound()
            coin.image.remove_from_all_objects()
            self.master_level.coins.remove(coin)

            self.maximize_life()
            break
```

Η παραπάνω μέθοδος σαρώνει όλα τα κέρματα από την λίστα των κερμάτων της κλάσης (Menu). Αν ένα κέρμα είναι αρκετά κοντά σύμφωνα με την μέθοδο της βοηθητικής κλάσης (image) και αν η ζωή του παίκτη έχει μειωθεί κάτω των 100 μονάδων. Τότε αφαιρείται το κέρμα από αυτή την λίστα αφού πρώτα παίζει ο χαρακτηριστικός του ήχος. Ο παίκτης καρδίζει έτσι 10 πόντους ζωή με την βοήθεια από την μέθοδο της κλάσης του (maximize_life()!

Μέθοδος απόκτησης κέρματος της κλάσης (Player)

Εικόνα 30

Περιληπτικά οι μέθοδοι της κλάσης (Player):

- maximize_magic(), αυξάνει συνεχώς την μαγεία με πολύ αργό ρυθμό.
- minimize_time(), μειώνει συνεχώς τον χρόνο με πολύ αργό ρυθμό.

(Οι μέθοδοι get_key(), unlock() και read_box_info() θα αναφερθούν σε άλλες κλάσεις που σχετίζονται με αυτές)

4.13 ΕΜΦΑΝΙΣΗ ΤΗΣ ΦΩΤΙΑΣ

Ο παίκτης βλέπει τον μάγο να πετάει λοιπόν φωτιές σε διάφορε κατευθύνσεις που θυμίζουν την κίνηση του παίκτη. Η φωτιά φεύγει από αυτόν με μεγάλη ταχύτητα σκοτώνοντας όσα τέρατα είναι στον δρόμο της τέλος εκρήγνυται όταν συγκρουστεί με κάποιο στατικό αντικείμενο σε ένα (animation) έκρηξης με ήχο!(βλέπε Εικόνα 31)



Εικόνες αντικειμένων φωτιάς και οι εκρήξεις τους

Εικόνα 31

4.13.1 ΛΕΙΤΟΥΡΓΙΑ ΤΗΣ ΦΩΤΙΑΣ

Η λειτουργία της φλόγας είναι πολύ απλή. Αφού δημιουργηθεί από τον παίκτη στην διεύθυνση που επιθυμεί ο μάγος του να ρίξει το ξόρκι, έχει μια και μόνο δουλειά να τρέξει γρήγορα προς αυτή την κατεύθυνση και όταν φτάσει σε κάποιο αντικείμενο το χάρτη να εκραγεί. Όσο και αν φαίνεται περίεργο στο παρών παιχνίδι δεν σκοτώνει η φλόγα τα τέρατα αλλά τα ίδια σκοτώνονται όταν η φλόγα φτάσει κοντά! Θα μπορούσε να είχε γίνει και διαφορετικά, αλλά επιλέχθηκε αυτός ο τρόπος λόγο της απλότητας του.

4.13.2 ΔΟΜΗ ΤΗΣ ΛΕΙΤΟΥΡΓΙΑΣ ΤΗΣ ΦΩΤΙΑΣ

Όπως προαναφέρθηκε και στην ανάλυση της κλάσης του μάγου (Player), όταν ένα αντικείμενο τύπου φλόγας δημιουργηθεί μπαίνει στην λίστα του παίκτη με τις φωτιές του ένα νέο αντικείμενο φωτιάς το οποίο έχει αρχική κατεύθυνση τα δικά του (animation) και εικόνες, τους δικούς του ήχους καθώς και την δική του μέθοδο (run()) η οποία τρέχει από τον παίκτη. Εδώ θα αναλυθούν τα πιο βασικά στοιχεία της λειτουργίας του κατασκευαστή αυτής της κλάσης καθώς και την μεθόδου (run()). Ο κατασκευαστής έχει πολλές ομοιότητες με τον κατασκευαστή του παίκτη διότι όλα τα αντικείμενα της κλάσης (Level), μοιάζουν αρκετά μεταξύ τους με πολλές κοινές μεθόδους και δομή. Η φωτιά φορτώνει αρχικά ένα αντικείμενο (image) που θα τελέσει την ίδια λειτουργία όπως στην κλάση του παίκτη, για τη θέση της φλόγας και άλλα στοιχεία της όπως εικόνα και ήχος. Μετά η κατεύθυνση της φωτιάς καθορίζεται μία φορά στην αρχή με την βοήθεια της μεταβλητής της κλάσης (init_dir) και επιλέγεται το αρχικό (animation) κίνησης με βάση αυτή την κατεύθυνση. Τέλος ορίζονται κάποιες μεταβλητές για την ταχύτητα αναπαραγωγής του (animation) της κίνησης και του χρόνου έκρηξης. Είναι σημαντικό να αναφερθεί ότι και ο παίκτης έχει τέτοιου είδους μέθοδο για την επιλογή κατάλληλου (animation) ανάλογα με την κατεύθυνση, αλλά αναλύεται εδώ για οικονομία χώρου, παρακάτω φαίνεται ο κατασκευαστής αυτής της κλάσης. (βλέπε Εικόνα 32)

```

def __init__(self, menu, map, master, init_dir, init_pos, all_objects):
    self.master = master

    self.image = dy_obj.DynamicObject(menu, map, "player_fier", init_pos, 0, order_frames = (
        'f_r_0',
        'f_r_1',
        'f_r_u_0',
        'f_r_u_1',
        'f_u_0',
        'f_u_1',
        'f_u_l_0',
        'f_u_l_1',
        'f_l_0',
        'f_l_1',
        'f_l_d_0',
        'f_l_d_1',
        'f_d_0',
        'f_d_1',
        'f_d_r_0',
        'f_d_r_1',
        'fier_expl_small'
    ), order_animations = ((0, 1), (2, 3), (4, 5), (6, 7), (8, 9), (10, 11), (12, 13), (14, 15)), order_sounds = ('fier', 'explosion'), all_objects = all_objects)

    if init_dir == (1, 0):
        self.image.set_cur_animation(0)
    elif init_dir == (1, -1):
        self.image.set_cur_animation(1)
    elif init_dir == (0, -1):
        self.image.set_cur_animation(2)
    elif init_dir == (-1, -1):
        self.image.set_cur_animation(3)
    elif init_dir == (-1, 0):
        self.image.set_cur_animation(4)
    elif init_dir == (-1, 1):
        self.image.set_cur_animation(5)
    elif init_dir == (0, 1):
        self.image.set_cur_animation(6)
    elif init_dir == (1, 1):
        self.image.set_cur_animation(7)

    self.image.set_frame_time(2)

    self.speed = master.speed * 2

    self.init_dir = init_dir

    self.not_expl = True
    self.explosion_time = 0
    self.max_expl_time = 16

```

Δημιουργία βοηθητικού αντικειμένου (image) τύπου (DynamicObject), το οποίο όπως και στον παίκτη κρατάει άλλες τις λειτουργίες απεικόνισης της φωτιάς του (animation) και των ήχων.

Εδώ το αντικείμενο της φωτιάς επιλέγει ποιά (animation) θα τρέξει ανάλογα την αρχική και μόνη κατευθυνση του (init_dir)

Κατασκευαστής της κλάσης της φωτιάς (Fier)

Εικόνα 32

Η μέθοδος (run()) ελέγχει αν δεν έχει εκραγεί η φωτιά για να συνεχίζει να παίζει το (animation) της και να την κινεί μπροστά. Αν τώρα η μεταβλητή της κλάσης με όνομα (not_expl) γίνει ψευδής, παίζει ο ήχος της έκρηξης μια φορά και μετά για όσο χρόνο ορίζει η μεταβλητή μεγίστου χρόνου έκρηξης της (max_expl_time). Τέλος το αντικείμενο βγαίνει από την λίστα με τις φωτιές του παίκτη και τελικά καταστρέφεται από την (Python). (βλέπε Εικόνα 33)

```

def run(self):
    self.image.run()

    if self.not_expl:
        if self.image.set_dir_from_map_col(self.init_dir) != self.init_dir:
            self.not_expl = False
            self.image.render_obj.set_cur_frame_num(16)
            self.image.render_obj.set_cur_sound_num(1)
            self.image.render_obj.play_sound()
            # self.image.remove_from_all_objects()
            # self.master.fiers.remove(self)
        else:
            self.image.play_cur_animation()
            self.image.change_map_rel_pos(self.image.set_dir_from_map_col((self.init_dir[0] * self.speed, self.init_dir[1] * self.speed)))
    else:
        if self.explosion_time == self.max_expl_time:
            self.image.remove_from_all_objects()
            self.master.fiers.remove(self)
            self.image.render_obj.set_alpha(int(256 * (1 - self.explosion_time / self.max_expl_time)))
            self.explosion_time += 1

```

Αν η φωτιά δεν έχει εκραγεί, δηλαδή δεν έχει συγκρουστεί με αντικείμενο του χάρτη Εκκίνησε τον ήχο έκρηξης και θέσε την μεταβλητή (not_expl) σε Ψευδή

Τέλος παίζει το (animation) της έκρηξης για όσα (frame) ορίζει η μεταβλητή (max_expl_time), και μετά αφαίρεσε το αντικείμενο απο την λίστα του παίκτη με τις φωτιές

Εικόνα της μεθόδου (run()) της κλάσης της φωτιάς (Fier)

Εικόνα 33

4.14. ΕΜΦΑΝΙΣΗ ΤΩΝ ΤΕΡΑΤΩΝ

Στην αρχή του επιπέδου (Level) ο παίκτης αντικρίζει μόνο κάποιες σχισμές (ruptures) πάνω στον χάρτη (map) , δεν αργεί να καταλάβει ότι αυτές οι σχισμές είναι το σπίτι των μαγικών τεράτων που τον καταδιώκουν. Μαγικά πλάσματα που μοιάζουν με σκιάχτρα που αιωρούνται. Ο παίκτης μπορεί να τα δει από όλες τις πλευρές και να καταλάβει ότι μάλλον μια μαγική δύναμη τα κινεί, καθώς και το μυστικό που του επιφυλάζουν.

4.14.1 ΛΕΙΤΟΥΡΓΙΑ ΤΩΝ ΤΕΡΑΤΩΝ

Τα τέρατα είναι τα αντικείμενα του παιχνιδιού που κυνηγούν τον παίκτη. Ένα τέρας μόλις ξεπηδήσει από μια σχισμή αρχίζει και κυνηγάει τον μάγο αν τον βλέπει και δεν είναι τίποτα ανάμεσα τους. Ο παίκτης μάγος έχει μόνο έναν τρόπο να αλληλοεπιδράσει μαζί του. Να το σκοτώσει! Όταν τα τέρατα είναι σε μικρή απόσταση από την φοιτά του παίκτη αναφλέγονται και πεθαίνουν ακαριαία. Μόνο ένα από αυτά συνεισφέρει στην πλοκή του παιχνιδιού καθώς κουβαλάει το μαγικό κλειδί που θα ξεκλειδώσει όλα τα κουτιά πληροφοριών. Πιο πολλά για τα κουτιά στην συνέχεια.

4.14.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΩΝ ΤΕΡΑΤΩΝ

Τα τέρατα κρατούνται από την λίστα (monsters) της κλάσης (Level). Όταν ενεργοποιηθεί η συνθήκη της πιθανότητας εμφάνισης ενός τέρατος που βρίσκεται στην κλάση αυτή(βλέπε εικόνα 34)

```
if rd.random() > 0.95:  
    self.add_monster()
```

Κομμάτι κώδικα της μεθόδου (run()) της κλάσης τύπου (Level)

Εικόνα 34

Ένα νέο αντικείμενο τέρατος προστίθεται στην λίστα των τεράτων με βάση την βοηθητική συνάρτηση (add_monster()). Παρακάτω φαίνεται η λειτουργία της συνοπτικά μέσα στην κλάση (Level).(βλέπε Εικόνα 35)

```
def add_monster(self):
    if self.max_monster_num > len(self.monsters):
        self.monsters.append(mon.Monster(self.menu, self, self.map, self.get_level_num(), self.player, self.map.floor.all_objects))
        self.monsters[-1].change_active_state()

    if rd.random() > 1- 1/16 and self.have_key():
        self.give_key()
        self.monsters[-1].get_key()
```

Αν υπάρχει περιθώριο για νέο τέρας με βάση τον μέγιστο αριθμό τεράτων (max_monsters_num), φτιάξε ένα νέο τέρας
 Αν είναι το τέρας τυχερό δώστο το κλειδί και μην ξαναδώσεις σε άλλο τέρας κλειδί

Μέθοδος δημιουργίας τέρατος από την κλάση (Level)

Εικόνα 35

Το τέρας επιλέγει την θέση που θα εμφανιστεί μέσα στον δικό του κατασκευαστή, σαρώνοντας όλες τις θέσεις του (map) και ρωτώντας συνέχεια αν η συγκεκριμένη θέση έχει τον κωδικό κλειδί ("R42") που υποδηλώνει ότι υπάρχει σημείο σχισμής στο χάρτη. Αν βρει την θέση του δημιουργείται σε αυτή ανάλογα αν είναι η πρώτη σχισμή που πρέπει να εμφανιστεί η δεύτερη και λοιπά. Η βοηθητική μεταβλητή που ορίζει σε ποια τρυπα θα εμφανιστεί συγκρατείται από την βοηθητική μεταβλητή (spawn_rupture_num). Πάλι το αντικείμενο έχει βοηθητικό αντικείμενο (image) για τις γνωστές λειτουργίες. Όμως έχει και την μεταβλητή του κλειδιού που καθορίζει την στιγμή που θα πεθάνει αν θα έχει να δώσει το κλειδί στο παίκτη! Το όνομα της μεταβλητής αυτής είναι (magic_key). Όπως είδαμε και προηγουμένως το τέρας δεν γνωρίζει αν έχει ή όχι το κλειδί. Η κλάση (Level) το καθορίζει αυτό όταν λέει στο τέρας να πάρει αυτό το κλειδί τυχαία!(βλέπε Εικόνα 36)

```
self.magic_key = False
self.dir = (rd.randint(-1, 1), rd.randint(-1, 1))
# random init spawn position based on spon map static objects

# find number of ruptures in this map
ruptures = 0
for layer_counter, line in enumerate(all_mp.maps[self.map_num]):
    for counter_x_32, key in enumerate(line):
        if key == 'R42':
            ruptures += 1
# chose a random one of those to be spawned from
spawn_rupture_num = rd.randint(1, ruptures)

# actual spawn in that particular paper rupture
already_spawned = False
ruptures_counter = 0
for layer_counter, line in enumerate(all_mp.maps[self.map_num]):
    for counter_x_32, key in enumerate(line):
        if key == 'R42':
            ruptures_counter +=1
            if spawn_rupture_num == ruptures_counter and not already_spawned:
                self.image.set_map_rel_pos((counter_x_32, layer_counter))
                already_spawned = True
```

Αρχικά το τέρας δεν έχει το κλειδί!
 Αρχικής κατεύθυνσης
 Κώδικας επιλογής σχισμής

Σημαντικό κομμάτι από τον κατασκευαστή της κλάσης του τέρατος

Εικόνα 36

Τώρα το τέρας είναι έτοιμο να κυνηγήσει και να φάει τον παίκτη η ακόμα και να πεθάνει και να του δώσει το κλειδί για την συνέχεια του παιχνιδιού. Αυτό γίνεται μέσω της μεθόδου του (run()) που αναλύεται στην συνέχεια.

Η μέθοδος (run()) τρέχει μέσα στην μέθοδο (run()) του επιπέδου όπως προαναφέρθηκε για όλα τα αντικείμενα του. Αυτή με την σειρά της τρέχει τις παρακάτω μεθόδους της κλάσης του τέρατος (Monster). Την μέθοδο (chase_player()) την μέθοδο (kill()) και την μέθοδο (die()). Όπως σε όλα τα προηγούμενα αντικείμενα έτσι και εδώ, οι μέθοδοι από την βοηθητικό αντικείμενο (image) δεν θα αναλυθούν, παρόλα αυτά όπως και προηγουμένως οι μέθοδοι που είναι υπογραμμισμένες θα αναλυθούν κανονικά! Παρακάτω φαίνεται η εικόνα από την μέθοδο (run()) καθώς και η εικόνα με κομμάτι της μεθόδου (run()) στην κλάση (Level) όπου και αυτή εμπεριέχεται.(βλέπε Εικόνες 37, 38)

```
for monster in self.monsters:  
    monster.run()
```

Η εκτέλεση της μεθόδου (run()) όλων των τεράτων

Εικόνα 37

```
def run(self):  
    self.image.run()  
    # self.dir = (rd.randint(-1, 1), rd.randint(-1, 1))  
    self.chase_player() ← κυνηγητό με τον παίκτη  
    self.kill() ← αν είναι κοντά επίθεση στον παίκτη και αφαίρεση ζωής απο αυτόν  
    self.play_random_spooky_sound()  
    self.image.change_map_rel_pos(self.image.set_dir_from_map_col((self.dir[0] * self.speed, self.dir[1] * self.speed)))  
    # use of level to dispone the monsters from it's list 'levels.monsters'  
    # self.image.remove_from_all_objects()  
    # self.master_level.monsters.remove(self)  
    self.die() ← το τέρας πεθαίνει αν είναι κοντά σε μία φωτιά!
```

Μέθοδος (run()) της κλάσης του τέρατος (Monster)

Εικόνα 38

Αρχικά η μέθοδος (chase_player()) η οποία είναι υπεύθυνη για το κυνηγίτο παίκτη και τέρατος. Ανά τακτά χρονικά διαστήματα που καθορίζονται από τη μεταβλητή της κλάσης με όνομα (frame_time()) επιλέγει το τέρας με την βοήθεια της μεθόδου του αντικειμένου (image) βρίσκεται η απόσταση μεταξύ τέρατος και μάγου. Αν η απόσταση είναι μεγαλύτερη από 8 μονάδες σε ευκλείδεια απόσταση, τότε το τέρας δεν βλέπει τον παίκτη. Αλλιώς βλέπει τον παίκτη και πάει κατά πάνω του.(βλέπε Εικόνες 39, 40)


```

def chase_player(self): # chases player by using image.time to give trigger on when to set new chase direction
if self.image.time % self.image.frame_time == 0:
    radius = self.image.get_dynamic_dist(self.player.image, True)
    if radius < 8: # απόσταση μικρότερη από 8 μονάδες!
        if self.see_player(radius):
            dx, dy = self.image.get_dynamic_dist(self.player.image, False)
            rad = self.image.get_dynamic_dist(self.player.image, True)
            self.dir = (round(dx / abs(rad)), round(dy / abs(rad)))
        else:
            self.dir = (rd.randint(-1, 1), rd.randint(-1, 1))
    else:
        self.dir = (rd.randint(-1, 1), rd.randint(-1, 1))

self.chose_frame_from_dir()
self.image.time += 1

```

Επιλογή κίνησης τυχαία η προς τον παίκτη αναλογα την αποσταση τους

Αυτη η επιλογή κατευθισης γινεται κάθε πχ 10(frames) ανάλογα με την τιμή μεταβλητης του αντικειμένου με όνομα (frame_time)

Μέθοδος κνηγητού του παίκτη από το τέρας της κλάσης τύπου τέρας

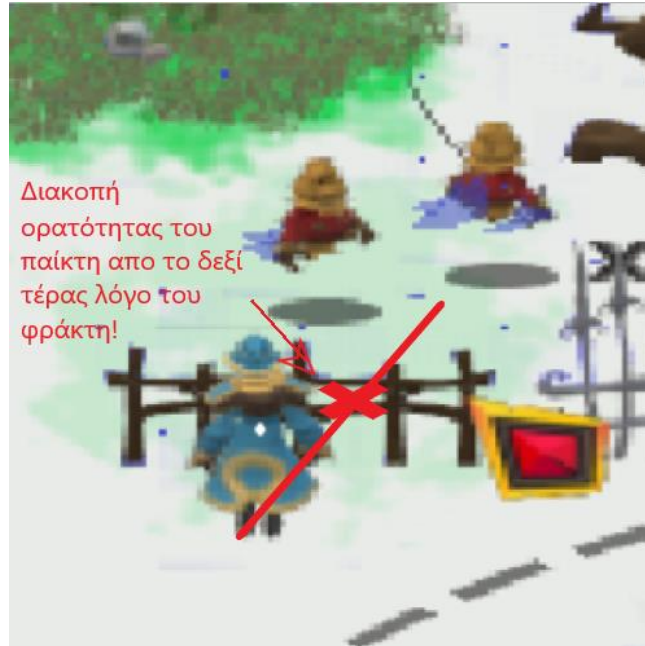
Εικόνα 39



Όψη μέγιστης ακτίνας ορατότητας τέρατος 8 μονάδων

Εικόνα 40

Σημαντική υποσημείωση, ακόμα και αν είναι κοντύτερα από ο μονάδες μπορεί να μην δει τον παίκτη το τέρας και αυτό συμβαίνει λόγω της μεθόδου (see_player()) η οποία στηρίζεται στην λογική του (Ray Casting) από το τέρας στον παίκτη. Αυτή η μέθοδος βρίσκει αν παρεμβάλλεται κάποιο αντικείμενο του χάρτη ανάμεσα στον παίκτη και το τέρας με αποτέλεσμα το τέρας να μην βλέπει τον παίκτη μάγο!(βλέπε Εικόνα 41)

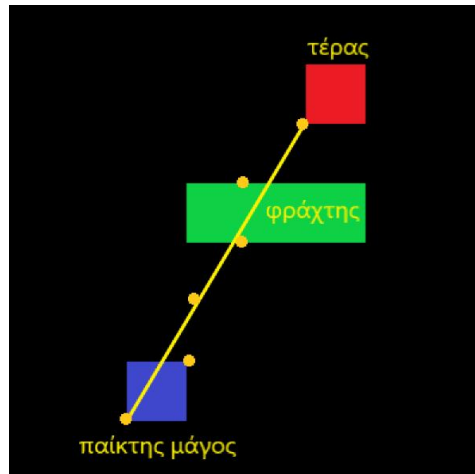


Παράδειγμα αποτυχίας ορατότητας του παίκτη σε μικρή απόσταση

Εικόνα 41

Σύμφωνα με τον Dr.M.Mohamed Sathik και την τρισδιάστατη εξαγωγή ακτίνας, σε δισδιάστατη μορφή μιας ακτίνας για τον απλουστευμένο αλγόριθμο (Ray Casting). Χρειάζονται τρία πράγματα η αρχή της ακτίνας, το επίπεδο δειγματοληψίας και το τελικό σημείο ο παίκτης μάγος. Η ακτίνα που επιτρέπει το τέρας να δει τον παίκτη κινείται με την κατεύθυνση που δίνεται από την διαφορά σημείων τέρατος και παίκτη. Η συχνότητα επιπέδων δειγματοληψίας έχει απόσταση ακριβώς το μέγεθος ενός τετραγώνου του χάρτη (η μικρότερη δυνατή του ενός (voxel)) συγκρούσεων από την κλάση (Map). Το σημείο που είναι υπεύθυνο για την διεξαγωγή συμπεράσματος για το αν υπάρχει τετράγωνο μέσα στο χάρτη που να κρύβει τον παίκτη από το τέρας, πρέπει να κάνει όλη την διαδρομή από το τέρας ως το παίκτη σε διακριτά βήματα. Αν σε κανένα βήμα δεν συναντήσει εμπόδιο, τότε το πεδίο είναι καθαρό και το τέρας βλέπει τον παίκτη. Παρακάτω ακολουθεί εικόνα επεξήγησης της διαδικασίας.

Η κίτρινη γραμμή είναι η ακτίνα, οι κίτρινες κουκίδες είναι οι στρογγυλοποιημένες δειγματοληψίες από αυτή την γραμμή αν έστω μια από αυτές μεταξύ του παίκτη και του τέρατος έχει κωδικό κλειδί από κάποιο αντικείμενο του χάρτη π.χ (το σημείο με συντεταγμένες στον χάρτη (5, 6) έχει την τιμή κλειδί = "t42"), δηλαδή δέντρο τότε ξέρει το τέρας ότι μπροστά από τον παίκτη έχει ένα δέντρο και δεν μπορεί να τον πλησιάσει!(βλέπε *Εικόνα 41*)



(Simple Ray Casting Example Image)

Εικόνα 42

Η κίτρινη γραμμή είναι η ακτίνα, (βλέπε Εικόνα 42) οι κίτρινες κουκίδες είναι οι στρογγυλοποιημένες δειγματοληψίες από αυτή την γραμμή αν έστω μια από αυτές μεταξύ του παίκτη και του τέρατος έχει κωδικό κλειδί από κάποιο αντικείμενο του χάρτη π.χ(το σημείο με συντεταγμένες στον χάρτη (5, 6) έχει την τιμή κλειδί = “42”), δηλαδή δέντρο τότε ξέρει το τέρας ότι μπροστά από τον παίκτη έχει ένα δέντρο και δεν μπορεί να τον πλησιάσει!

(Η κατανόηση του αλγορίθμου οφείλεται στο παρακάτω link:

<http://ijiet.com/wp-content/uploads/2015/02/17.pdf>)

Συνεχίζοντας η μέθοδος του τέρατος (kill()) φροντίζει αν βρεθεί πολύ κοντά στον παίκτη να του αφαιρέσει 10 μονάδες ζωής χρησιμοποιώντας την μέθοδο από την κλάση του παίκτη με όνομα (minimize_life()).(βλέπε Εικόνα 43) Η μέθοδος του παίκτη για αυτό τον σκοπό μοιάζει πολύ με την μέθοδο αφαίρεσης μαγείας.

```
def kill(self): # remove from levels list and from all_objects!
    if self.image.get_dynamic_dist(self.player.image, True) < 1:
        self.player.minimize_life()
        # print(self.player.life)
```

Μέθοδος του τέρατος (Monster) για την μείωση ζωής του παίκτη (Player)

Εικόνα 43

Τέλος το μέσω την μεθόδου (die()) το τέρας πεθαίνει και αφαιρείται από την λίστα του επιπέδου (Level), που περιέχει τα τέρατα (monsters) παίζοντας για μια τελευταία φορά τον ήχο του. Αν το τέρας αυτό είχε το κλειδί, δηλαδή την μεταβλητή (magic_key) σε αληθή τιμή τότε με την μέθοδο του (give_key()) πετάει το κλειδί στον χάρτη στο σημείο που ήταν στο τελευταίο (frame) της ζωής του. Παρακάτω φαίνεται με λεπτομέρεια αυτή η διαδικασία.(βλέπε Εικόνα 44)

```

def die(self):
    for fier in self.player.fiers:
        if self.image.get_dynamic_dist(fier.image, True) < 1:
            self.image.render_obj.play_sound()
            self.image.remove_from_all_objects()
            self.master_level.monsters.remove(self)
            if self.have_key():
                self.give_key()
            break

```

Σκανάρισμα όλων των αντικείμενων φωτιάς της λίστας του αντικειμένου του παίκτη, αν έστω και μια φωτιά είναι κοντά παίξει τον ήχο του τέρατος και αφαιρεσεται απο την λιστατεράτων του αντικειμένου (Level). Αν το τέρας έχει το κλειδί να το πετάξει πριν πεθάνει!

Μέθοδος θανάτου του τέρατος της κλάσης (Monster)

Εικόνα 44

Περαιτέρω σχολιασμός της μεθόδου (give_key()) της κλάσης του τέρατος.(βλέπε *Εικόνα 45*)

```

def give_key(self):
    self.master_level.MK = m_k.MagicKey(self.image.menu, self.image.map, self.player, self.image.get_map_rel_pos(), self.image.render_obj.all_objects)
    self.master_level.MK.change_active_state()
    self.master_level.MK.image.set_cur_animation(0)
    self.master_level.MK.image.render_obj.play_sound()
    self.magic_key = False

```

Παραπάνω φαίνεται η δημιουργία μέσα στο επίπεδο του νέου αντικειμένου κλειδί (MK = m_k.MagicKey()) μέσω της κλάσης (MagicKey) και ο ήχος το κλειδιού παίζει στην συνέχεια.

Μέθοδος (give_key()) της κλάσης (Monster) για την δημιουργία του κλειδιού

Εικόνα 45

Με την βοήθεια της κλάσης τύπου (MagicKey),(βλέπε *Εικόνα 45*) το κλειδί του επιπέδου με όνομα (MK) που προηγουμένως είχε τιμή (None), δηλαδή κανένα κλειδί, τώρα έχει ένα κλειδί στην θέση του τέρατος που καταστράφηκε και θεωρητικά το εμπεριείχε. Μετά ο ήχος του κλειδιού παίζει μια φορά για να σημάνει στον παίκτη μάγο ότι το κλειδί υπάρχει πάνω στο χάρτη ακόμα και αν δεν το βλέπει αυτή την στιγμή!

4.15 ΕΜΦΑΝΗΣΗ ΤΟΥ ΜΑΓΙΚΟΥ ΚΛΕΙΔΙΟΥ

Το κλειδί εμφανίζεται στον παίκτη στο σημείο που πεθαίνει ένα τέρας. Χρυσό σε χρώμα και καθώς αιωρείται γύρω από τον εαυτό του έχει μόνο ένα σκοπό, να το βρει και να το πάρει ο παίκτης!(βλέπε Εικόνα 46)



Εικόνα του μαγικού κλειδιού

Εικόνα 46

4.15.1 ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΜΑΓΙΚΟΥ ΚΛΕΙΔΙΟΥ

Το κλειδί δημιουργείται από ένα σκοτωμένο τέρας και στην ουσία η μόνη λειτουργία του είναι να καταστραφεί όταν το πλησιάσει και το πάρει ο παίκτης.

4.15.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΜΑΓΙΚΟΥ ΚΛΕΙΔΙΟΥ

Όπως προαναφέρθηκε το στο επίπεδο έχει δημιουργηθεί το κλειδί με την βοήθεια του κατασκευαστή της κλάσης (MagicKey) η αλήθεια είναι ότι μετά την δημιουργία του δεν εμπεριέχει τίποτα το ιδιαίτερο στην δική του μέθοδο (run()) παράς μόνο την απεικόνιση της περιστροφής του με την βοήθεια του βοηθητικού αντικειμένου (image). Ο παίκτης είναι υπεύθυνος για την καταστροφή του και την εξαφάνιση του από το χάρτη. Μαζί με την ανάθεση της δικής του μεταβλητής (magic_key) της κλάσης (Player) σε αληθή, δείχνοντας ότι τώρα ο παίκτης έχει το κλειδί. Παρακάτω φαίνεται η δομή της λειτουργίας αυτής της

μεθόδου στην κλάση αυτή.(βλέπε *Εικόνα 47*) Από την μέθοδο (run()) του παίκτη (Player), που περιέχει την μέθοδο (get_key()).

```
def get_key(self):
    if not (self.master_level.MK is None):
        if self.image.get_dynamic_dist(self.master_level.MK.image, True) < 1:
            self.master_level.MK.image.render_obj.set_cur_sound_num(1)
            self.master_level.MK.image.render_obj.play_sound()
            self.master_level.MK.image.remove_from_all_objects()
            self.master_level.MK = None
            self.magic_key = True
```

Αν η απόσταση από το κλειδί είναι κοντινή θέσε την μεταβλητή (magic_key) σε αληθή και θέσε την μεταβλητή του επιπέδου (Level) σε (None)

Μέθοδος (get_key()) της κλάσης τύπου (Player)

Εικόνα 47

4.16 ΕΜΦΑΝΗΣΗ ΤΩΝ ΚΕΡΜΑΤΩΝ

Τα κέρματα υπάρχουν σε συγκεκριμένες θέσεις από την αρχή του επιπέδου. Εξαφανίζονται μόνο αν τα χρειαστεί ο παίκτης. Χρυσά και αυτά σε χρώμα αιωρούνται και περιστρέφονται γύρω από τον εαυτό τους. Είναι εκεί μέχρι να τα χρειαστεί και αυτά ο παίκτης!

4.16.1 ΛΕΙΤΟΥΡΓΙΑ ΤΩΝ ΚΕΡΜΑΤΩΝ

Όπως και το μαγικό κλειδί τα κέρματα μόνο περιστρέφονται με ένα απλό (animation) με την βοήθεια πάλι του αντικειμένου (image) και εξαφανίζονται όταν έρθει ο παίκτης πολύ κοντά!

4.16.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΩΝ ΚΕΡΜΑΤΩΝ

Τα κέρματα όπως προαναφέρθηκε έχουν την ίδια λειτουργία περιστροφής όπως το μαγικό κλειδί, πάλι το αντικείμενο του παίκτη στο επίπεδο είναι υπεύθυνος για την αφαίρεση τους από την λίστα με την χρήση της μεθόδου του με όνομα (get_coin()), που έχει προαναφερθεί μέσα στην δομή της λειτουργίας του παίκτη παραπάνω.

4.17 ΕΜΦΑΝΗΣΗ ΤΩΝ ΚΟΥΤΙΩΝ ΜΕ ΤΙΣ ΠΛΗΡΟΦΟΡΙΕΣ

Τα κουτιά είναι και αυτά όπως τα κέρματα σε συγκεκριμένες θέσεις μέσα στο χάρτη του επίπεδου. Έχουν πάνω ένα λουκέτο και υποψιάζου τον παίκτη στην ιδέα να τα ξεκλειδώσει. Όμως αυτή την φορά δεν είναι όσο απλό φαίνεται!

4.17.1 ΛΕΙΤΟΥΡΓΙΑ ΤΩΝ ΚΟΥΤΙΩΝ ΜΕ ΤΙΣ ΠΛΗΡΟΦΟΡΙΕΣ

Τα κουτιά ανοίγουν μόνο με το κλειδί από τα τέρατα και αλλά και μια συγκεκριμένη σειρά που είναι άγνωστη στον παίκτη! Όταν διαβαστούν όλα τα κουτιά είναι πλέον ώρα για την επίλυση του γρίφου με έναν χαρακτηριστικό ήχο νίκης, καθώς ο παίκτης βρήκε την σειρά για όλα τα κουτιά!

4.17.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΩΝ ΚΟΥΤΙΩΝ ΜΕ ΤΙΣ ΠΛΗΡΟΦΟΡΙΕΣ

Τα κουτιά είναι δημιουργημένα ήδη από την αρχή του επιπέδου μέσα στην στατική δομή πίνακα (tuple) του αντικειμένου του επιπέδου τύπου (Level). Τα κουτιά δεν καταστρέφονται λοιπόν και δεν δημιουργούνται. Επίσης είναι σχετικά στατικά αντικείμενα με την μόνη λειτουργία στην μέθοδο (run()) να αλλάζουν την εικόνα τους στο κείμενο που περιέχουν μόνο αν έχουν κλειδωθεί με την τιμή της μεταβλητής (locked) να έχει τεθεί σε αληθής με την βοήθεια πάλι της μεθόδου του αντικειμένου του παίκτη (unlock()) που την αλλάζει.

Παρακάτω ακολουθεί η εικόνα της δομής αυτής της μεθόδου μαζί με μικρή επεξήγηση για την λειτουργία της.(βλέπε *Εικόνα 48*)

```
def unlock(self):
    for box in self.master_level.info_boxes:
        if self.image.get_dynamic_dist(box.image, True) < 1 and self.have_key() and box.is_locked() and box.get_unlock_num() == self.get_unlock_num():
            if self.get_unlock_num() == len(self.master_level.info_boxes) - 1:
                box.image.render_obj.set_cur_sound_num(2)
                box.image.render_obj.play_sound()
                box.image.render_obj.set_cur_sound_num(0)
            else:
                box.image.render_obj.play_sound()
        box.unlock()
        self.change_unlock_num()
```

← Έχει ξεκλειδωμάτος και εμφάνισης των πληροφοριών μόνο αν η μεταβλητή σειράς ανοίγματος του κουτιού είναι ίδια με τον αριθμό κουτιού της κλάσης του κουτιού. Ήχος νίκης αν ανοίξει και το τελευταίο κουτί

Εδώ γίνεται όντως το ξεκλείδωμα του

← Αλλαγή μεταβλητής σειράς ανοίγματος κουτιού, για να ανοίξει το επόμενο σε σωστή σειρά κουτί

Μέθοδος της κλάσης (Player) που ανοίγει με την σειρά τα κουτιά

Εικόνα 48

```
i_b.InfoBox(self.menu, self.map, self.player, 1, '11b0', (1, 4), all_objects),
```

Εικόνα δημιουργία αντικειμένου τύπου (InfoBox) με σειρά ξεκλειδώματος(1)

Εικόνα 49

Στην παραπάνω εικόνα(βλέπε *Εικόνα 49*) φαίνεται η δημιουργία του πρώτου αντικειμένου κουτιού με τονισμένο τον αριθμό του, δηλαδή τον αριθμό ξεκλειδώματος του από τον παίκτη. Το συγκεκριμένο κουτί θα ξεκλειδωθεί πρώτο όταν το βρει ο παίκτης γιατί έχει τον αριθμό 1!

4.18 ΕΜΦΑΝΗΣΗ ΤΗΣ ΤΕΛΙΚΗΣ ΠΥΛΗΣ

Η τελική πύλη αποτελείται από τον πάπυρο που ανοίγει στο τέλος του επιπέδου και από το (animation) που δείχνει την πύλη κλειστή.(βλέπε *Εικόνα 50*)



Εικόνα πύλης κατά την διάρκεια του παιχνιδιού

Εικόνα 50

4.18.1 ΛΕΙΤΟΥΡΓΙΑ ΤΗΣ ΤΕΛΙΚΗΣ ΠΥΛΗΣ

Όταν ο παίκτης τελειώσει με τα κουτιά του μένει μόνο ο γρίφος και το ξόρκι που έχει να λύσει πηγαίνοντας κοντά στην τελική πύλη του επιπέδου. Όταν χτυπήσει την πύλη με ένα ξόρκι ανοίγει μπροστά του ο μαγικός πάπυρος που του ζητάει να λύσει το ξόρκι για να τελειώσει την εκπαίδευση του. Ο παίκτης γραφεί σε γλώσσα (Python) και όταν τελειώσει απομακρύνεται από την πύλη και ενεργοποιεί το ξόρκι! Αν το ξόρκι πετύχει η πύλη ανοίγει και τελειώνει το επίπεδο με ήχο νίκης (βλέπε Εικόνα 54) και καταγράφεται το όνομα του στην λίστα με τους παίκτες. Αλλιώς ξανά προσπαθεί μέχρι να τελειώσει ο χρόνος!

4.18.2 ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΤΗΣ ΤΕΛΙΚΗΣ ΠΥΛΗΣ

Ο κατασκευαστής του αντικειμένου τύπου (Riddle) της πύλης εκτός από τις βασικές λειτουργίες (animation) εικόνων και ήχου, έχει και τον βασικό ρόλο να θέτει το ξόρκι ως προς τις εισόδους του αλλά και ως προς τον κώδικα του ανάλογα με το επίπεδο! Στην συνέχεια υπάρχει βοηθητικό αντικείμενο τύπου (SpellRun) το οποίο τρέχει και τις δύο εκδόσεις του κώδικα. Μια του παίκτη μάγου και μία του προδιαγεγραμμένου κώδικα από το βοηθητικό (module) με όνομα (riddle) το οποίο έχει τους γνήσιους κώδικες για της λύση κάθε επιπέδου! Αν το αποτέλεσμα από τον παίκτη ταιριάζει με αυτό του γνήσιου κώδικα κλείνει το επίπεδο με επιτυχία. Αλλιώς ο παίκτης ξανά προσπαθεί. Την παραπάνω λειτουργία την αναλαμβάνει η μέθοδος της κλάσης (Riddle) με όνομα (close_level_successfully()). Αν τώρα ο παίκτης θέλει να λύσει τον γρύφο χωρίς να έχει ανοίξει όλα τα κουτιά δεν μπορεί να το κάνει καθώς δεν του εμφανίζεται ο πάπυρος, αυτό γίνεται λόγω της βοηθητικής μεθόδου της κλάσης αυτή με όνομα (is_player_close_on_click()). Οι προϋποθέσεις είναι δύο να είναι ξεκλειδωμένη η πύλη, να είναι κοντά ο παίκτης και να χτυπήσει την πύλη με μία φωτιά!(βλέπε Εικόνα 51)

```
def is_player_close_on_click(self, mouse_image, game_events):
    for event in game_events:
        x, y = mouse_image.get_pos()
        if event.type == pygame.MOUSEBUTTONDOWN and not self.is_locked() and self.image.get_dynamic_dist(self.player.image, (True)) < 2 and x < 384:
            self.see = not self.see
            self.papyrus.change_active_state()
            self.papyrus.play_sound()
            # reset also players motion to not glitch the game
            self.player.key_states = {
                pygame.K_d: False,
                pygame.K_s: False,
                pygame.K_a: False,
                pygame.K_w: False,
                pygame.K_RIGHT: False,
                pygame.K_DOWN: False,
                pygame.K_LEFT: False,
                pygame.K_UP: False
            }
        elif event.type == pygame.KEYDOWN and not self.is_locked() and self.image.get_dynamic_dist(self.player.image, (True)) > 2 and x < 384:
            if event.key == pygame.K_e:
                self.close_level_successfully()
```

Αν όλες οι προϋποθέσεις είναι αληθείς ανοίγει τον πάπυρο με τον χαρακτηριστικό ήχο του.

Τέλος με επιτυχία αν είναι ίδιες οι εξόδοι (print()) από τους κώδικες παίκτη και μηχανής

Άνοιγμα του πάπυρου και έλεγχος τερματισμού του επιπέδου

Εικόνα 51

(Παρακάτω ακολουθεί συνοπτικά η πιο σημαντική μέθοδος της σύγκρισης των αλγορίθμων ουσιαστικά του παίκτη και της μηχανής από τα αποτελέσματά τους! Βλέπε Εικόνα 52)

```

def close_level_successfully(self):

    print(self.inputs + self.papyrus_text.text)

    self.my_exe.restricted_exec_run(self.inputs + self.papyrus_text.text)
    player_code_output_list = self.my_exe.get_printed_outputs()

    print(player_code_output_list)

    if self.test_code_output_list == self.my_exe.get_printed_outputs():
        if not (self.menu.selected_level is None):
            if not self.menu.selected_level.exit:
                self.menu.focus = False
                self.menu.open()
                self.menu.selected_level.change_active_state()
                # show loser_image first
                self.menu.winner_image.change_active_state()
                self.menu.winner_image.play_sound() # testing if this line will play the loser sound

                self.menu.selected_level.remove_from_all_objects()

                self.menu.selected_level.exit = True
            else:
                self.papyrus_text.text = '# Wrong spell!'
    # End of game's level if the code is successfull!

```

← Αν οι εξόδοι απο τους αλγόριθμους είναι ίδιες τερματισσε με επιτυχία και ανοιξε το κυρίως μενού με την εικόνα νίκης!
 ← Αλλιώς γράψε στον πάπυρο να ξαπαπροσπαθήσει ο παίκτης!

Μέθοδος σύγκρισης αλγορίθμων της κλάσης (Riddle)

Εικόνα 52

Τελευταίες παρατηρήσεις πριν την επεξήγηση του μοντέλου (chat_bot API) που χρησιμοποιήθηκε ,της μηχανής γραφικών που βασίζεται πάνω στις δομές και τις μεθόδους του (Pygame) και την ιδέα της κλάσης για την επεξεργασία του κώδικα με την βοήθεια της μεθόδου exec() της (Python). Το κυρίως μενού τελικά σώζει τον παίκτη στο αρχείο παικτών με όνομα (players_data.json) με την βοηθητική κλάση τύπου (ExampleClass), η οποία θα εξηγηθεί και αυτή συνοπτικά παρακάτω. Ο παίκτης τελειώνει το παιχνίδι ανεπιτυχώς με ήχο αποδοκιμασίας και εικόνα που του λέει ότι απέτυχε στην εκπαίδευση του(βλέπε *Εικόνα 53*). Τέλος το μενού ενώνεται με το βιβλίο στο βασικό κομμάτι κώδικα με όνομα (game.py). Η λειτουργία των κειμενογράφων στον πάπυρο και στο βιβλίο καθώς και ο μηχανισμός παγώματος της ροής του παιχνιδιού για τις ερωτήσεις και την συγγραφή των ξορκιών δεν επεξηγείται παρακάτω, όμως η πηγή για την δημιουργία τους αντλήθηκε από το παρακάτω σύνδεσμο (link):

(https://pygame.readthedocs.io/en/latest/4_text/text.html)

(Παρακάτω ακολουθούν η εικόνες τέλους του παιχνιδιού, καθώς και εικόνες από τον πάπυρο και το βιβλίο που δεν συμπεριλήφθηκαν παραπάνω!)



Τελική εικόνα με αποτυχία της εκπαίδευσης

Εικόνα 53



Τελική εικόνα με επιτυχία της εκπαίδευσης

Εικόνα 54



Εικόνα για συγγραφή των ξορκιών σε γλώσσα (Python)

Εικόνα 55



Εικόνα βιβλίου με παράδειγμα χρήσης του (Chat_Gpt_3.5_turbo)

Εικόνα 56

4.19 ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΜΑΓΙΚΟΥ ΒΙΒΛΙΟΥ ΣΥΝΟΠΤΙΚΑ

Το βιβλίο με τα ξόρκια βρίσκεται την πάνω δεξιά μεριά του παιχνιδιού και όπως προαναφέρθηκε μπορεί να λειτουργεί συνεχώς και ανεξάρτητα με το κυρίως μενού ή τα επίπεδα. Την στιγμή που ο παίκτης επιλέξει να διαβάσει το βιβλίο πατώντας το χαρακτηριστικό εικονίδιο τότε η λειτουργία του παιχνιδιού σταματάει και η λειτουργία του βιβλίου ξεκινάει, μόνο αν υπάρχει σύνδεση στο διαδίκτυο. Γίνεται μια ερώτηση με βάση τον κειμενογράφο του βιβλίου και όταν πατηθεί το κουμπί (Enter) τότε ένα ερώτημα γίνεται στο (API) του (Chat_gpt_3.5_turbo). Αυτό διαβάζει το κείμενο που του στάλθηκε και απαντάει ανάλογα. Δεν χρειάζεται να έχει το μηχάνημα γνώση προηγούμενων πραγμάτων για να επικοινωνήσει με τον χρήστη. Όμως για το συγκεκριμένο παιχνίδι υπάρχει στην αρχή του παιχνιδιού μια μικρή συζήτηση με το (API) την στιγμή της σύνδεσης για να εξασφαλιστεί η εγκυρότητα των απαντήσεων με βάση την ιστορία του παιχνιδιού και τον ρόλο του μέσα στο παιχνίδι!(βλέπε Εικόνα 57)

```
def __init__(self, gpt_role = "You are a magic book in a game where player builds simple python code. Don't reveal your identity, use only ascii characthers in your responses! Answer only simple python code related topics nothing else Dont reply with more than 50 words per answer, you can use less words tho! I now let you to players hands")
```

Κείμενο στον κατασκευαστή για απόκτηση γνώσης περί του θέματος απο το (API)

Αρχική συζήτηση με το (API)

Εικόνα 57

Παρακάτω ακολουθεί η μέθοδος του (module) με όνομα (openai) για τις ερωτήσεις, απαντήσεις.(βλέπε Εικόνα 58)

```
def ChatGPT_conversation(self, conversation):  
    # chek internet before the following code  
    response = openai.ChatCompletion.create(  
        model=self.model_id,  
        messages=conversation  
    )  
    # api_usage = response[ 'usage' ]  
    # print('Total token consumed: {}'.format(api_usage['total_tokens']))  
    # stop means complete  
    # print(response['choices'][0].finish_reason)  
    # print(response['choices'][0].index)  
    conversation.append({'role': response.choices[0].message.role, 'content': response.choices[0].message.content})  
    return conversation
```

Αποστολή της τρέχουσας συζήτησης στο (API) και πρόσθεση της απάντησης του στην συζήτηση

Ερωτήσεις και απάντησης στο (API) του (Chat) με την βοήθεια του (openai)

Εικόνα 58

Για να μπορέσει κάποιος να χρησιμοποιήσει το (chat) θα πρέπει να έχει δικό του λογαριασμό στο (API) της (OpenAi) και να πληρώνει το λιγότερο ένα μικρό ποσό τον χρόνο, για να το χρησιμοποιεί και να έχει το απαραίτητο κλειδί για την σύνδεση του. Αυτό θα το εισάγει στον κατασκευαστή της κλάσης (Logic) στο (module) με όνομα (chat_gpt_logic.py) μέσα στην μεταβλητή (API_KEY = “YOUR API KEY”)! Για περεταίρω πληροφορίες κατευθυνθείτε στον σύνδεσμο:(<https://platform.openai.com/docs/guides/gpt>)

4.20 ΛΕΙΤΟΥΡΓΙΑ ΜΗΧΑΝΗΣ ΓΡΑΦΙΚΩΝ ΣΥΝΟΠΤΙΚΑ

Η μηχανή γραφικών του παρόντος παιχνιδιού χρησιμοποιεί την μέθοδο (blit) του (Pygame) για να αναπαραστήσει εικόνες την μία πάνω στην άλλη με διαφάνεια και με μεγέθυνση, χωρίς περιστροφή ή αλλαγή χρώματος με κάποιο τρόπο. Βοηθητικές μέθοδοι εκτός από την παραπάνω είναι οι εξής:

- transform.scale() Για αλλαγή μεγέθους
- image.load() Και φόρτωση εικόνας

Επίσης η βασική κλάση γραφικών που παίρνει πολλά αντικείμενα εικόνας και ήχου για την δημιουργία (animation) με όνομα (Object), έχει χρησιμοποιηθεί.

Πως ακριβώς δουλεύει όμως η επαλληλία ανάλογα με το ποιο αντικείμενο είναι μπροστά ή πίσω δίνεται από την κλάση με όνομα (OrderedZList), αυτή η κλάση αρχικά κρατάει όλα τα αντικείμενα τύπου (Object), χρησιμοποιεί την βοηθητική μεταβλητή τους ως κλειδί, καθώς και την βοηθητική μέθοδο (sort) της (Python) για να κατατάξει τα αντικείμενα σε αύξουσα σειρά. Από το πιο μακρινό στο πιο κοντινό. Τέλος απεικονίζει την το κάθε αντικείμενο με αυτή την σειρά δίνοντας την εντύπωση ότι τα μακρινά αντικείμενα υπερκαλύπτονται από τα πιο κοντινά. Οι κλάσεις με ονόματα δυναμικό αντικείμενο και στατικό αντικείμενο χρησιμοποιούνται όχι ακριβώς από αυτή την μηχανή γραφικών, αλλά καλύτερα από το αντικείμενο (map) το οποίο καθορίζει πως θα εμφανίζονται τα στατικά αντικείμενα και τα δυναμικά στο χάρτη του επιπέδου. Στην ίδια κλάση με όνομα (OrderedZList) επίσης είναι και η λειτουργία του κέρσορα, για το ποιο αντικείμενο ήταν το πιο “πάνω” που πατήθηκε για να δώσει τον κωδικό του σε όλες τις άλλες διεργασίες του παιχνιδιού! Οι ορισμοί δυναμικό και στατικό αντικείμενο στην ανάλυση του παιχνιδιού είναι δική μου σύμβαση! Η σημασία τους είναι η εξής. Όταν ένα αντικείμενο της πίστας δεν αλληλοεπιδρά με τον παίκτη με κάποιο τρόπο π.χ(τα σύννεφα, ή τα δέντρα) αυτό είναι ένα στατικό αντικείμενο. Αλλιώς ένα δυναμικό και δεν αναλαμβάνει η τελική εικόνα του χάρτη να το εμπεριέχει. Πάρα σχεδιάζεται μόνο του μέσα στο επίπεδο όπως έχει γίνει ήδη φανερό για αντικείμενα όπως ο παίκτης η τα τέρατα που έχουν το δικό τους βοηθητικό αντικείμενο γραφικών με όνομα(image). Παρακάτω ακολουθεί σύνδεσμος (link) σε ακαδημαϊκό αρχείο που επεξηγεί την λειτουργία του παρόντος αλγορίθμου ονόματος (Painter’s Algorithm) με σαφήνεια:

(<https://theory.stanford.edu/~aiken/publications/papers/ppopp23.pdf>)

4.21 ΛΕΙΤΟΥΡΓΙΑ ΤΗΣ ΜΗΧΑΝΗΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΚΩΔΙΚΑ ΣΥΝΟΠΤΙΚΑ

Η επεξεργασία κώδικα σε (Python) γίνεται με την βοήθεια της κλάσης με όνομα (RunSpell) αυτή η κλάση εμπεριέχει περιορισμούς για το πως θα τρέξει η (Python) από τον παίκτη με την βοήθεια περιορισμών. Οι περιορισμοί αυτοί δεν αφήνουν την (Python) μέσω του χρήστη να τρέξει κώδικα που να επηρεάζει τον εξωτερικό κόσμο του μηχανήματος εκτός από το κομμάτι κώδικα που του ζητείται να γράψει. Τον περιορίζει ουσιαστικά σε πολύ βασικές λειτουργίες όπως π.χ(την χρήση μεθόδου (print()) για εκτύπωση). Τέλος η κλάση αποθηκεύει την έξοδο όλων των μεθόδων εκτύπωσης (print()) με ανακατεύθυνση σε μια λίστα από (strings). Η περαιτέρω λειτουργία για το πως να χρησιμοποιηθεί η κλάση μέσα στο παιχνίδι έχει ήδη προαναφερθεί. Παρακάτω ακολουθεί ο σύνδεσμος που άντλησα τις πηγές μου (link):

(<https://realpython.com/python-exec>)

4.22 ΛΕΙΤΟΥΡΓΙΑ ΤΗΣ ΑΠΟΘΗΚΕΥΣΗΣ ΠΑΙΚΤΩΝ ΣΥΝΟΠΤΙΚΑ

Η λογική πίσω από την αποθήκευση παικτών, είναι η εξής. Με την βοήθεια της κλάσης με όνομα (ExampleClass), ο παίκτης αν δεν έχει ξαναπαίξει το παρόν επίπεδο που επιλεγεί και το τερματίσει τότε γίνεται η εξής λειτουργία από την κλάση. Βλέπει το όνομα και αν υπάρχει βλέπει το επίπεδο, αν και αυτό το επίπεδο υπάρχει και έχει ξαναπαιχτεί από αυτόν ανανεώνει την εγγραφή μόνο αν υπάρχει καλύτερος χρόνος! Αν δεν υπάρχει ξανά το επίπεδο απλά εγγράφει το παρών σκορ. Τέλος αν δεν υπάρχει ούτε το όνομα πάλι εγγράφει το παρών σκορ και το νέο όνομα. Ο κώδικας σχεδιάστηκε με βάση το παρακάτω link:

(<https://www.nylas.com/blog/the-complete-guide-to-working-with-json>)

ΚΕΦΑΛΑΙΟ ΠΕΜΠΤΟ

ΑΞΙΟΛΟΓΗΣΗ

5.1 ΕΙΣΑΓΩΓΗ

Σε αυτό το κεφάλαιο θα αναλυθεί η αξιολόγηση του παιχνιδιού από ένα μικρό δείγμα οχτώ μαθητών. Τρεις από την Α Γυμνασίου, Τρεις από την Α Λυκείου και δύο από την Γ Λυκείου. Η αξιολόγηση διεξήχθη υπό την μορφή μίας συνέντευξης συγκεκριμένων ερωταπαντήσεων. Ο κάθε μαθητής μεμονωμένα και σε διαφορετικούς χρόνους από τους υπόλοιπους ερωτήθηκε μία σειρά είκοσι τεσσάρων ερωτήσεων. Αρχικά, οι μαθητές είχαν απαντήσει αν το παιχνίδι ήταν διασκεδαστικό, με τα αν το παιχνίδι τους έμαθε νέα πράγματα, δηλαδή να ήταν εκπαιδευτικό. Μετέπειτα έγιναν ερωτήσεις σχετικές με το ψηφιακό βοηθό και την όψη του μάσα στο παιχνίδι. Τέλος ερωτήθηκαν για την εικόνα του μενού και τις λειτουργίες του, καθώς και για τα γραφικά του παιχνιδιού γενικότερα. Θα παρουσιαστούν οι ερωτήσεις, η λογική με την οποία τέθηκαν, ο τρόπος διεξαγωγής της συνέντευξης και τέλος η ανάλυση των πορισμάτων της αξιολόγησης από το δείγμα των μαθητών.

5.2 ΟΙ ΕΡΩΤΗΣΕΙΣ ΑΞΙΟΛΟΓΗΣΗΣ ΤΟΥ ΠΑΙΧΝΙΔΟΥ ΑΠΟ ΤΟΥΣ ΠΑΙΚΤΕΣ(ΜΑΘΗΤΕΣ)

Μολονότι το δείγμα ήταν μικρό, οι ερωτήσεις σχεδιάστηκαν έτσι ώστε να κατατοπίζουν τους μαθητές σχετικά με τον παράγοντα στον οποίο διερωτώνται. Υπάρχει μία νοηματική συνέχεια από την μία ερώτηση στην επόμενη καθώς και οι ερωτήσεις διέπονται από σαφήνεια. Στην σχεδίαση των ερωτήσεων προσέχτηκε το περιεχόμενο τους να είναι κατανοητό από τα παιδιά καθώς και μελετήθηκαν διάφορα άλλα ερωτηματολόγια και ερωτήσεις από το Google. Μία ιστοσελίδα από την οποία μελετήθηκαν διάφορες ερωτήσεις και αντλήθηκε έμπνευση για τον σχηματισμό των ερωτήσεων για το παιχνίδι είναι η εξής:

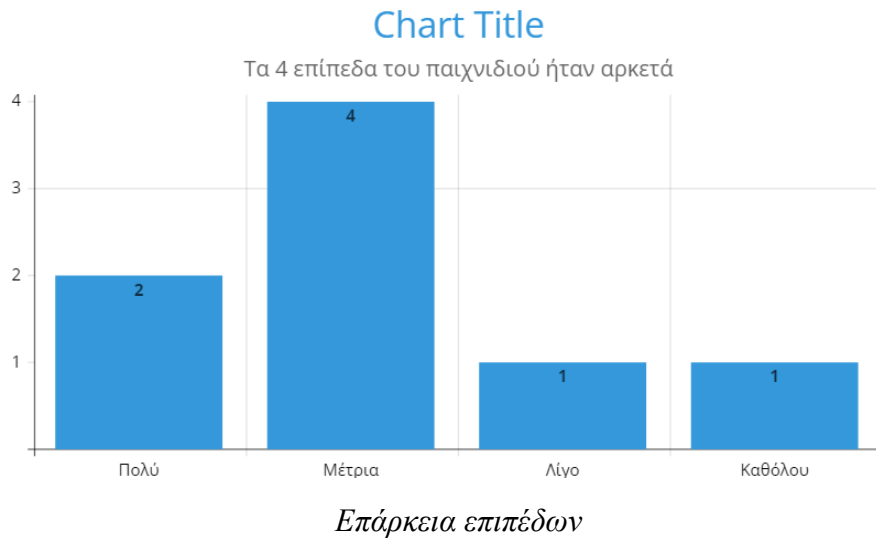
(<http://technologia.pbworks.com/w/page/33745904/Ερωτηματολόγια%20Εργασιών>)

(Οι ερωτήσεις είναι κλειστού περιεχομένου, επιλογής όπου μπορούν να επιλέξουν μία από τις τέσσερις απαντήσεις σε κάθε ερώτηση)

5.3 ΑΠΟΤΕΛΕΣΜΑΤΑ ΑΞΙΟΛΟΓΗΣΗΣ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ ΑΠΟ ΤΟΥΣ ΠΑΙΚΤΕΣ (ΜΑΘΗΤΕΣ)

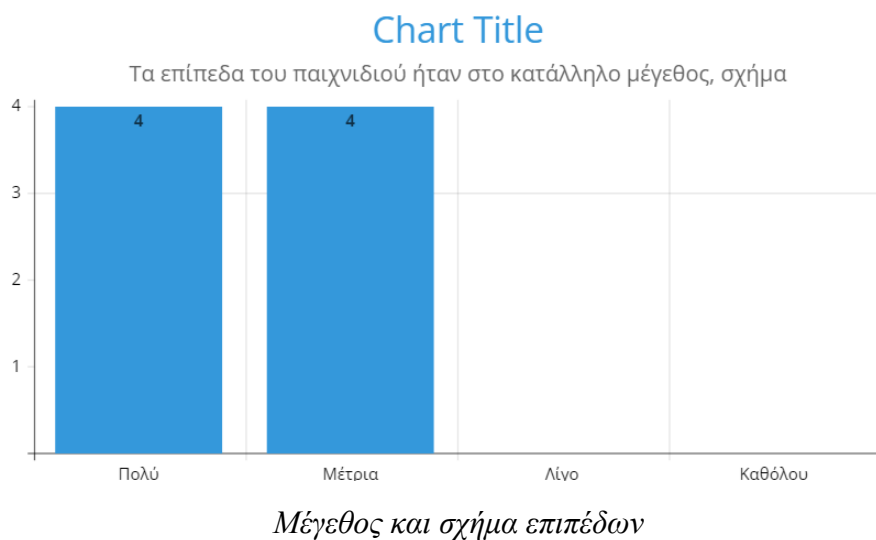
Το πρώτο ερωτηματολόγιο είχε ερωτήσεις σχετικά με το πόσο διαδραστικό φάνηκε το παιχνίδι στους παίκτες. (Σύνολο παικτών 8)

1. Τα 4 επίπεδα του παιχνιδιού ήταν αρκετά?



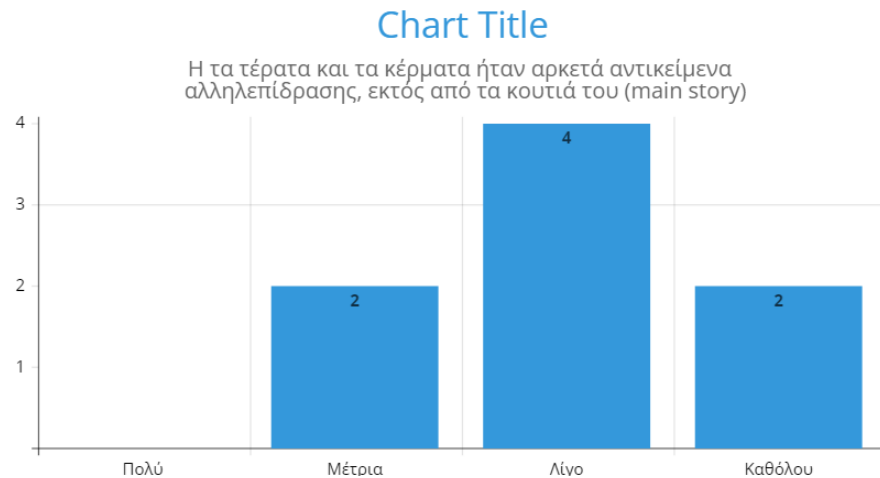
Διάγραμμα 1

2. Τα επίπεδα του παιχνιδιού ήταν στο κατάλληλο μέγεθος, σχήμα?



Διάγραμμα 2

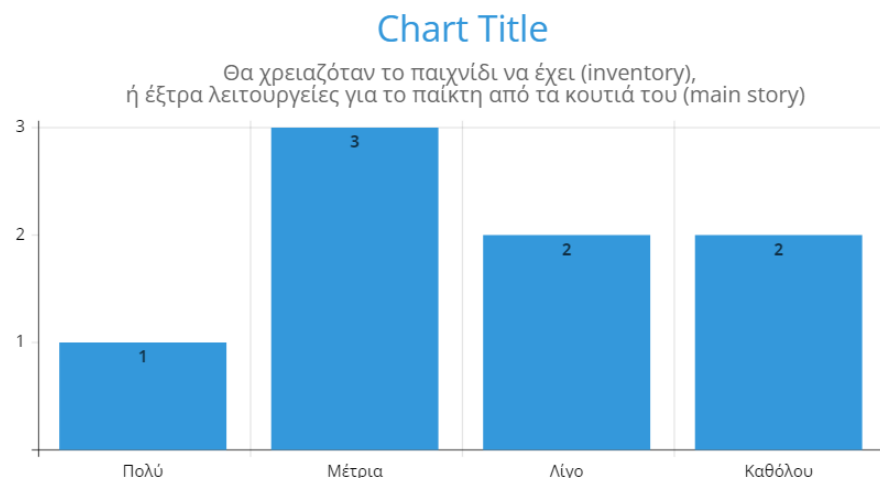
3. Η τα τέρατα και τα κέρματα ήταν αρκετά αντικείμενα αλληλεπίδρασης, εκτός από τα κουτιά του (main story)?



Επάρκεια αντικειμένων αλληλεπίδρασης παίκτη

Διάγραμμα 3

4. Θα χρειαζόταν το παιχνίδι να έχει (inventory), ή έξτρα λειτουργίες για το παίκτη?



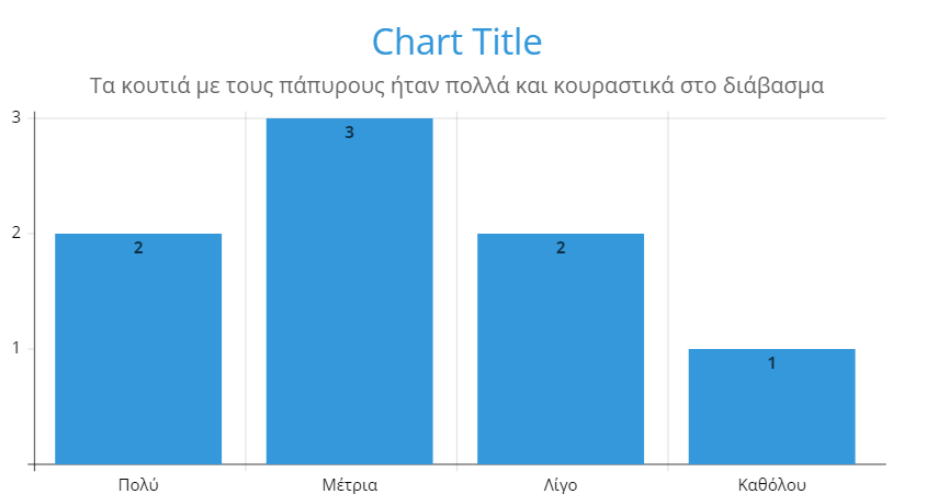
Επάρκεια λειτουργιών παιχνιδιού(gameplay)

Διάγραμμα 4

Από τα παραπάνω γραφήματα φαίνεται η εντύπωση που άφησε το παιχνίδι στους μαθητές. Φαίνεται οι περισσότεροι μαθητές να βρήκαν το παιχνίδι ουδέτερο καθώς οι λίγες πίστες και η μικρή συλλογή αντικειμένων το καθιστούσε λιγότερο διαδραστικό από όσο θα περίμεναν. Γενικά σε αυτή την κατηγορία το παιχνίδι δεν φαίνεται να πέτυχε τον στόχο του.

Το δεύτερο ερωτηματολόγιο είχε ερωτήσεις σχετικά με το πόσο εκπαιδευτικό φάνηκε το παιχνίδι στους παίκτες.

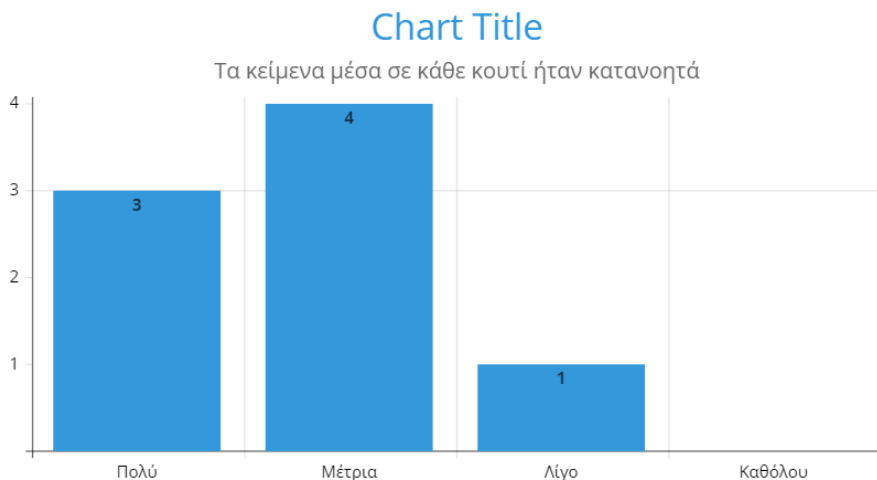
6. Τα κουτιά με τους πάπυρους ήταν πολλά και κουραστικά στο διάβασμα?



Ποσότητα κουτιών και περιεχομένου τους

Διάγραμμα 5

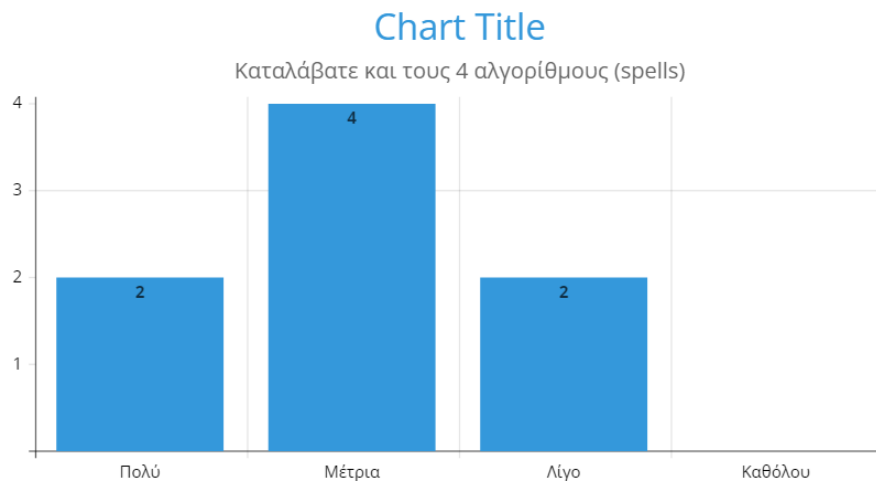
7. Τα κείμενα μέσα σε κάθε κουτί ήταν κατανοητά?



Κατανόηση κειμένων κουτιών

Διάγραμμα 6

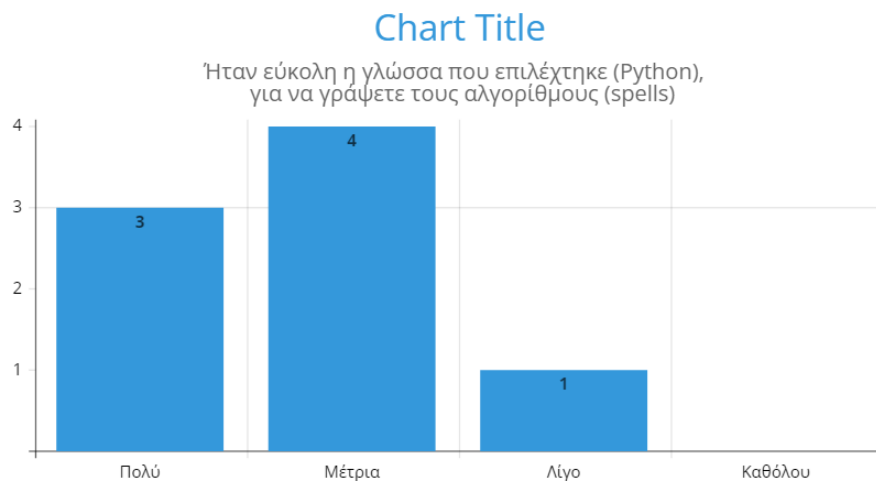
8. Καταλάβατε και τους 4 αλγορίθμους (spells)?



Κατανόηση αλγορίθμων

Διάγραμμα 7

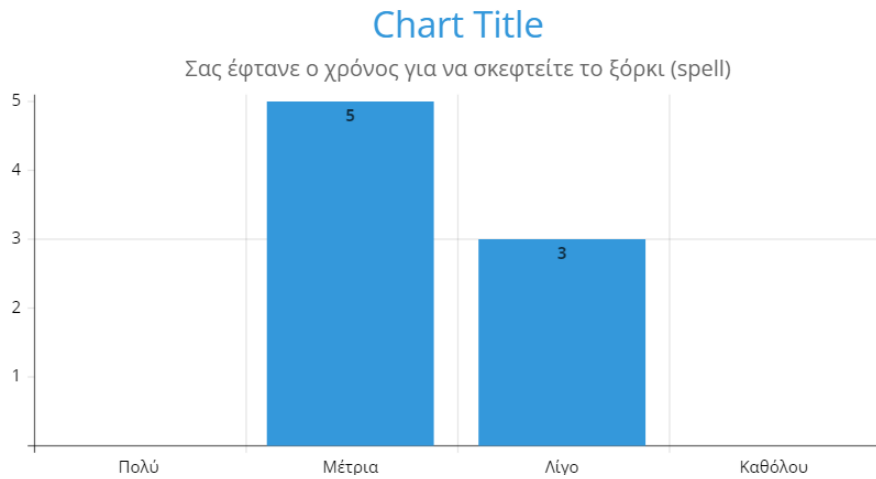
9. Ήταν εύκολη η γλώσσα που επιλέχτηκε (Python), για να γράψετε τους αλγορίθμους (spells)?



Ευκολία κατανόησης και χρήσης της γλώσσας (Python)

Διάγραμμα 8

10. Σας έφτανε ο χρόνος για να σκεφτείτε το ξόρκι (spell)?



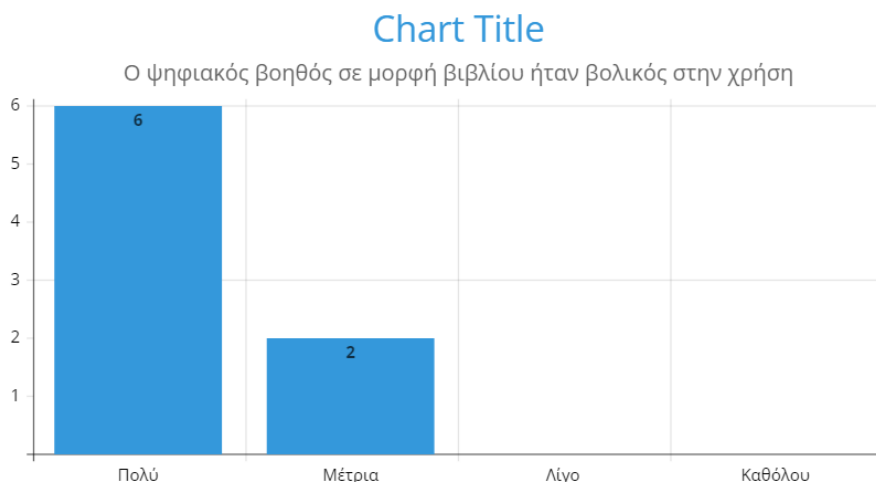
Επάρκεια χρόνου για την κατανόηση του αλγορίθμου

Διάγραμμα 9

Παρακάτω φαίνεται η εκπαιδευτική ικανότητα του παιχνιδιού. Πιο συγκεκριμένα από τους μαθητές αποσπάστηκε θετική κριτική με μόνο αρνητικό, την πολυπλοκότητα των κουτιών με πολλές πληροφορίες να κατανοήσουν σε λίγο χρόνο. Σε περίληψη, το παιχνίδι φάνηκε να επιτυγχάνει τον σκοπό της εκπαίδευσης, καθώς κατάφερε να διδάξει στα παιδιά τους αλγορίθμους σε (Python) και την λογική πίσω από αυτούς.

Το τρίτο ερωτηματολόγιο είχε ερωτήσεις σχετικά με το πόσο χρήσιμος και εύχρηστος ήταν ο ψηφιακός βοηθός με την μορφή μαγικού βιβλίου μέσα στο παιχνίδι.

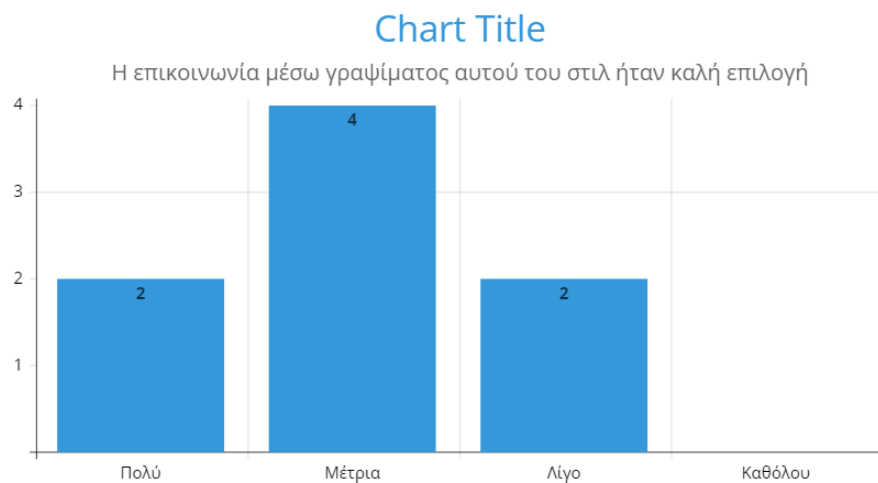
11. Ο ψηφιακός βοηθός σε μορφή βιβλίου ήταν βολικός στην χρήση?



Ευχρηστία του βιβλίου βοήθειας του παίκτη (μάγου)

Διάγραμμα 10

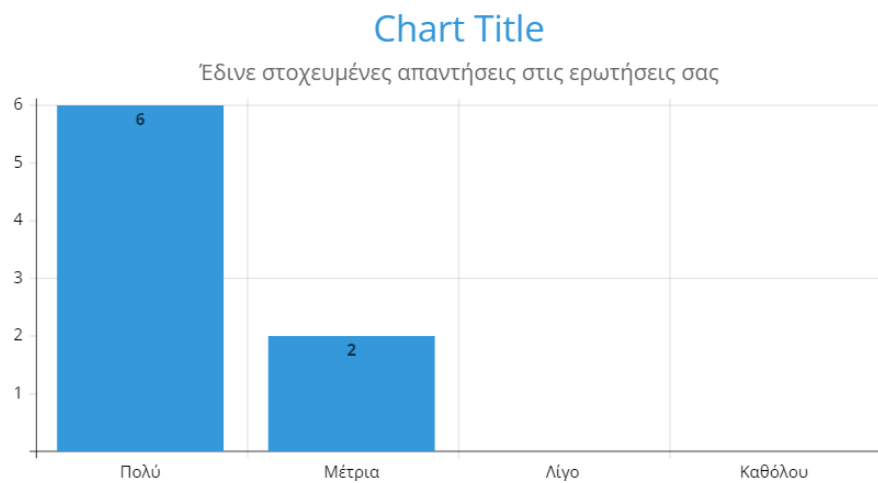
12. Η επικοινωνία μέσω γραψίματος αυτού του στιλ ήταν καλή επιλογή?



Ευκολία επικοινωνίας μέσω γραψίματος με το βιβλίο βοηθό

Διάγραμμα 11

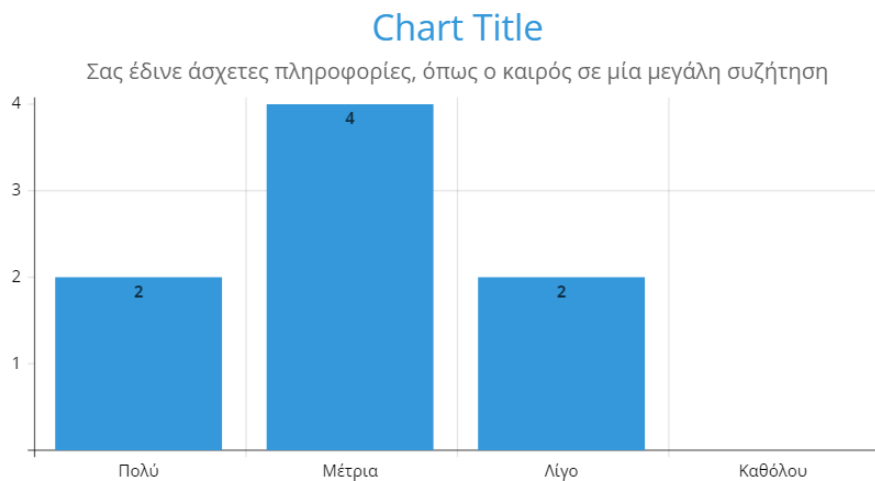
13. Έδινε στοχευμένες απαντήσεις στις ερωτήσεις σας?



Απαντήσεις εντός θέματος με το βιβλίο βοηθό

Διάγραμμα 12

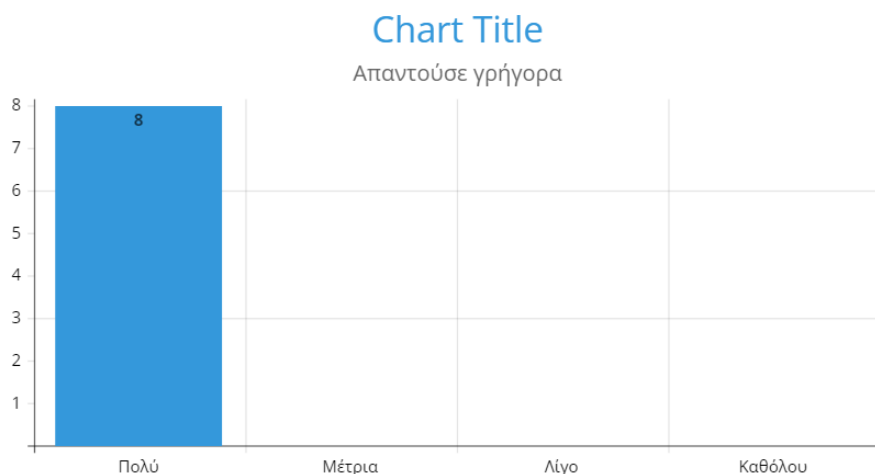
14. Σας έδινε άσχετες πληροφορίες, όπως ο καιρός σε μία μεγάλη συζήτηση?



Συζητήσεις εντός θέματος με το βιβλίο βοηθό

Διάγραμμα 13

15. Απαντούσε γρήγορα?



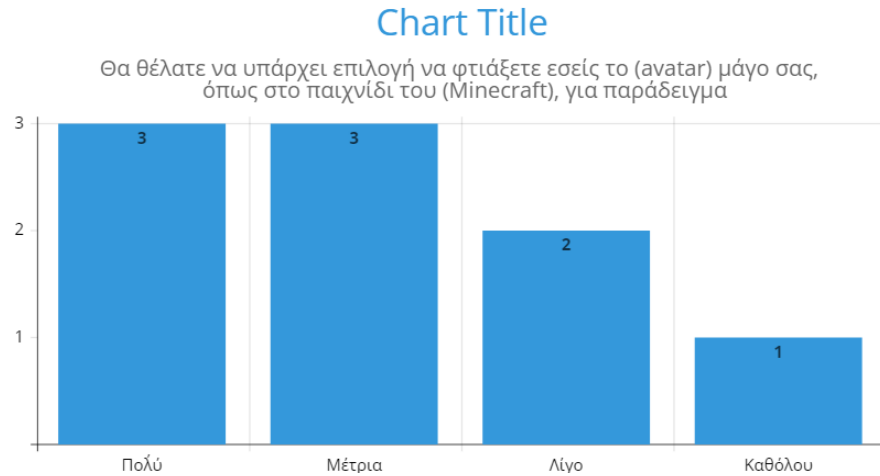
Ταχύτητα απαντήσεων βιβλίου βοηθού

Διάγραμμα 14

Ο ψηφιακός βοηθός φαίνεται να ενθουσίασε τους μαθητές καθώς σχολίασαν πολύ θετικά την ταχύτητα και την τον τρόπο με τον οποίο απαντούσε στις ερωτήσεις καθώς και το πόσο στοχευμένες ήταν οι απαντήσεις του. Τα μόνα αρνητικά ήταν πάλι η δια δραστικότητα σε μορφή κειμένου που φαίνεται να κούραζε τους μαθητές, αλλά και οι συζητήσεις εκτός θέματος μετά από κάποιο χρονικό διάστημα συνομιλώντας μαζί του. Πάλι το παιχνίδι φαίνεται να πέτυχε το σκοπό του καθώς σε μικρές συζήτησης, που ο μαθητής έκανε ερωτήσεις πάνω στο αντικείμενο του παιχνιδιού οι απαντήσεις ήταν γρήγορες και στενευμένες.

Το τέταρτο ερωτηματολόγιο είχε ερωτήσεις σχετικά με την όψη και την ευχρηστία του μενού.

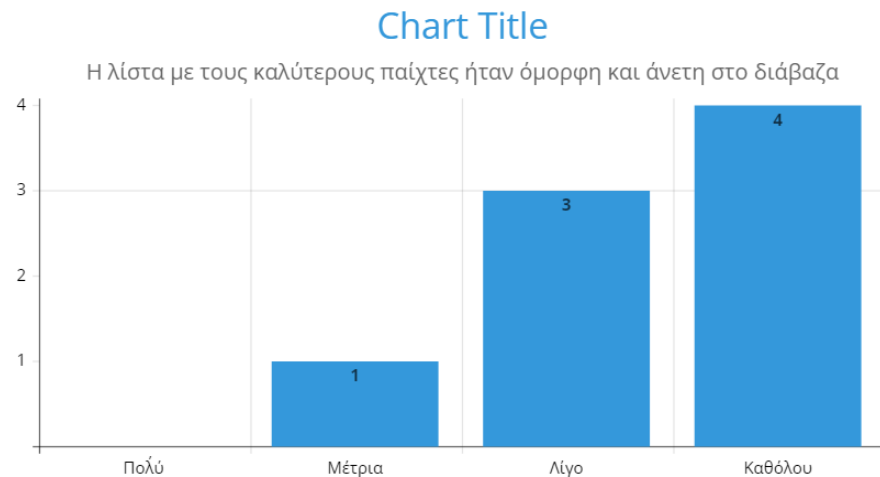
16. Θα θέλατε να υπάρχει επιλογή να φτιάξετε εσείς το (avatar) μάγο σας, όπως στο παιχνίδι του (Minecraft), για παράδειγμα?



Επιλογή αλλαγής παίκτη (μάγου)

Διάγραμμα 15

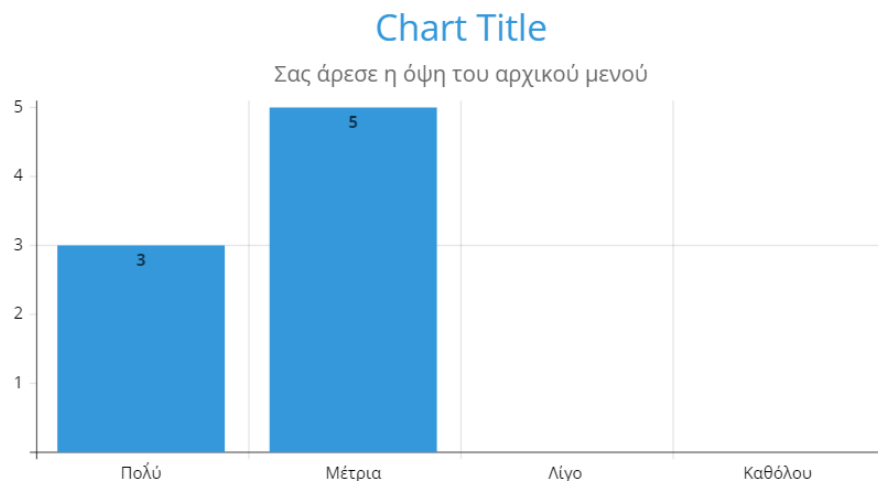
17. Η λίστα με τους καλύτερους παίκτες ήταν όμορφη και άνετη στο διάβαζα?



Ποιότητα απεικόνισης λίστας επιτυχόντων παικτών

Διάγραμμα 16

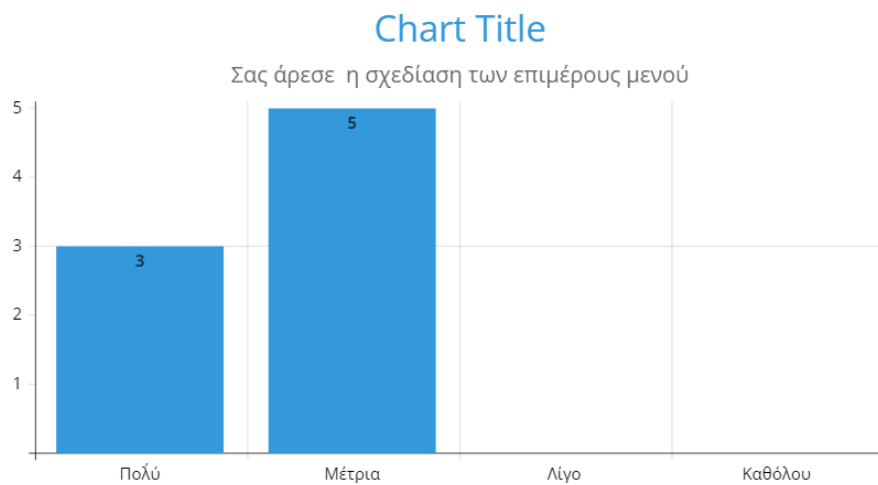
18. Σας άρεσε η όψη του αρχικού μενού?



Όψη του αρχικού μενού

Διάγραμμα 17

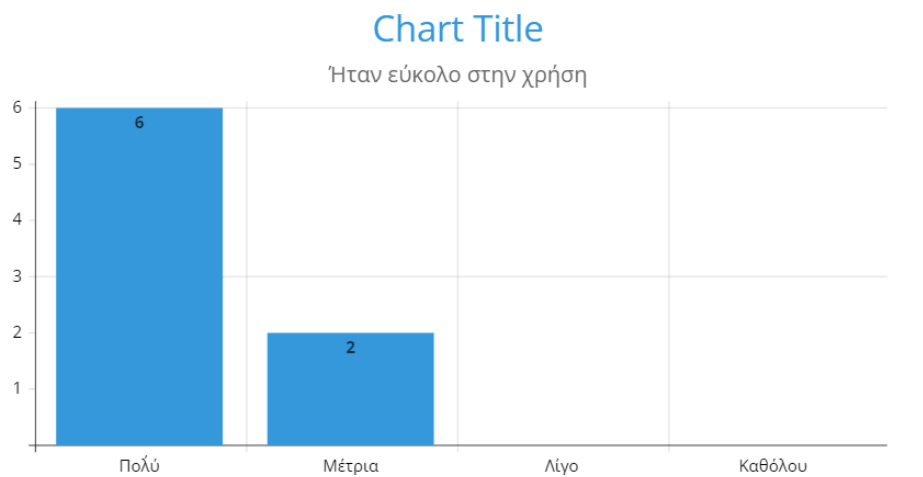
19. Σας άρεσε η σχεδίαση των επιμέρους μενού?



Όψη επιμέρους μενού

Διάγραμμα 18

20. Ήταν εύκολο στην χρήση?



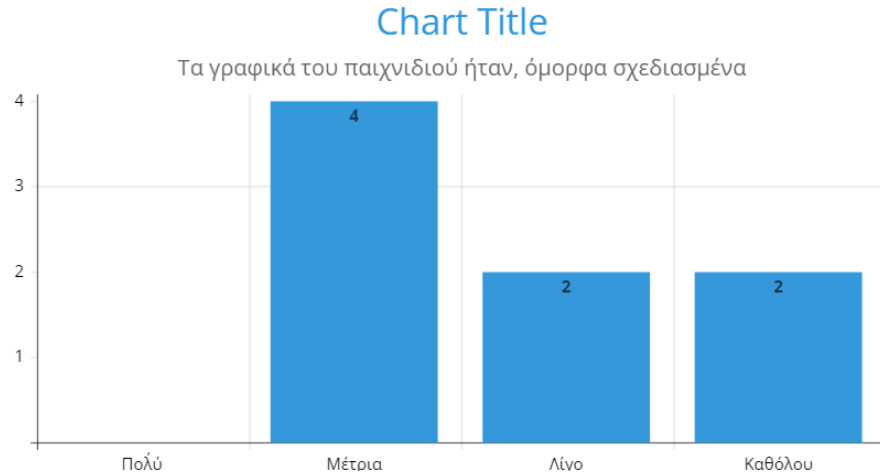
Ευκολία χρήσης του μενού

Διάγραμμα 19

Το μενού απέσπασε καλές κρητικές, αφού τα παιδιά δεν είχαν αν προσάψουν κανένα αρνητικό σχόλιο σε καμία λειτουργία του μενού. Το μόνο αρνητικό φαίνεται να ήταν η εμφάνιση της λίστας παικτών που είχαν ήδη παίξει ως δυσανάγνωστή. Άρα και σε αυτό το κομμάτι το μενού πέτυχε τον σκοπό της λειτουργίας του ακόμα και αν δεν είχε ακριβώς την ίδια όψη όπως τα περισσότερα μενού των παιχνιδιών στις μέρες μας.

Τέλος το πέμπτο ερωτηματολόγιο είχε ερωτήσεις σχετικά με γενικά χαρακτηριστικά του παιχνιδιού.

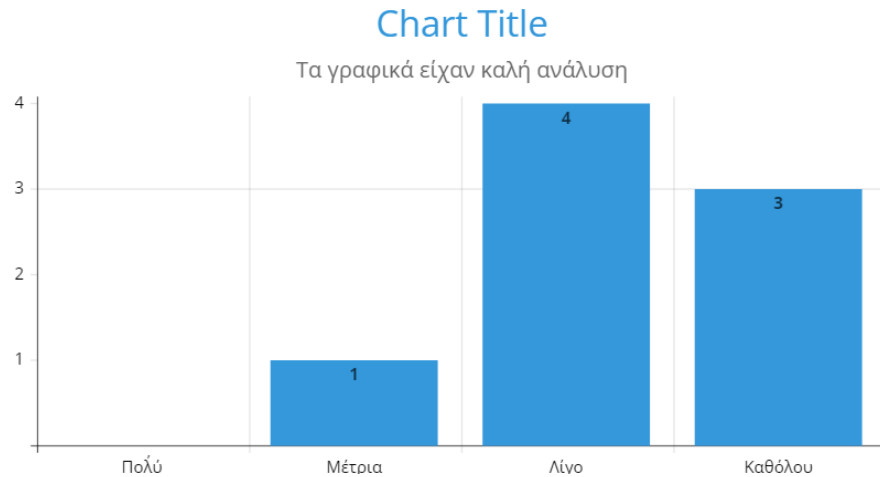
21. Τα γραφικά του παιχνιδιού ήταν, όμορφα σχεδιασμένα?



Ποιότητα γραφικών παιχνιδιού

Διάγραμμα 20

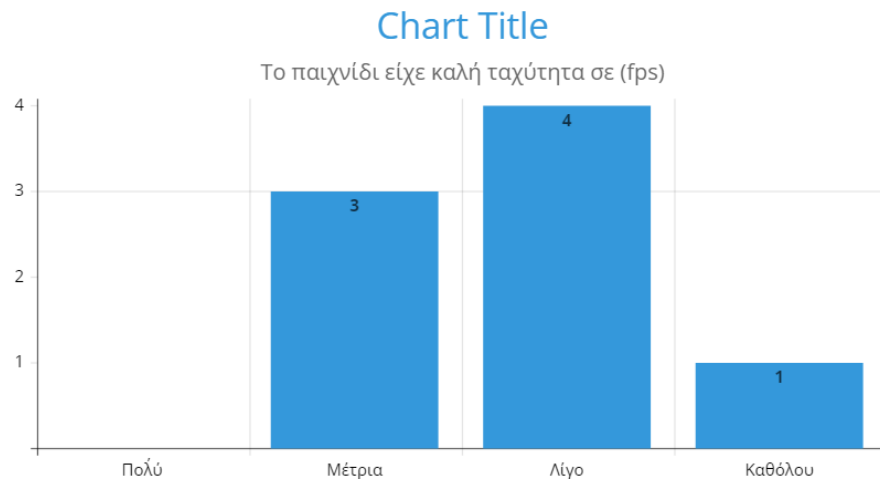
22. Τα γραφικά είχαν καλή ανάλυση?



Ανάλυση εικόνας παιχνιδιού σε (pixels)

Διάγραμμα 21

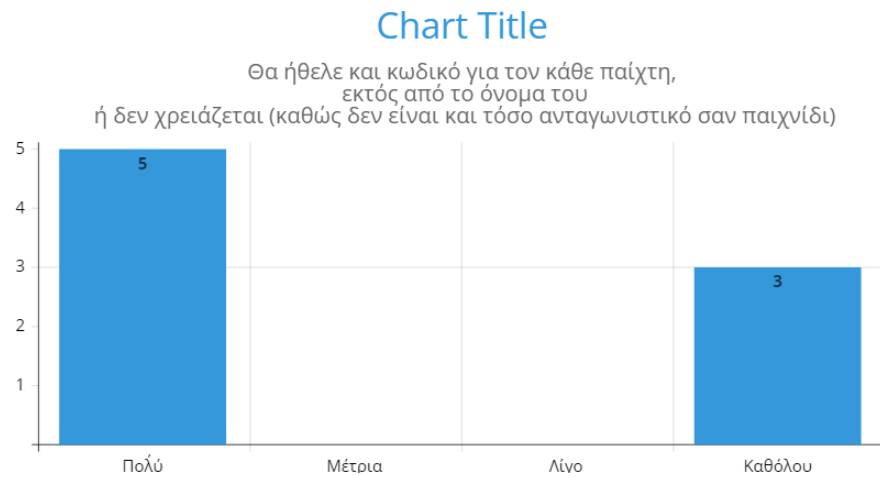
23. Το παιχνίδι είχε καλή ταχύτητα σε (fps)?



Ταχύτητα αναπαραγωγής γραφικών

Διάγραμμα 22

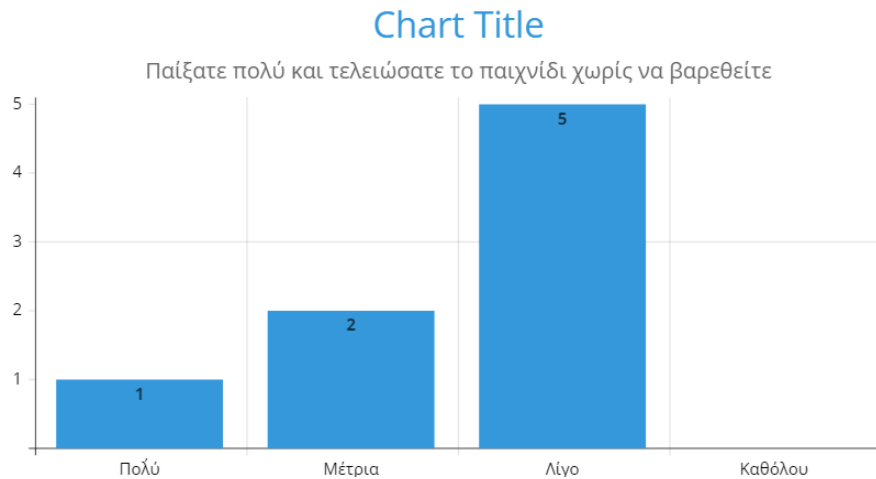
24. Θα ήθελε και κωδικό για τον κάθε παίκτη εκτός από το όνομα του ή δεν χρειάζεται (καθώς δεν είναι και τόσο ανταγωνιστικό σαν παιχνίδι)?



Επιλογή κωδικού για κάθε παίκτη

Διάγραμμα 23

25. Παίζατε πολύ και τελειώσατε το παιχνίδι χωρίς να βαρεθείτε?



Διάρκεια αλληλεπίδρασης παίκτη με το εκπαιδευτικό παιχνίδι

Διάγραμμα 24

Εδώ δυστυχώς φάνηκε το παιχνίδι να μην έχει καλά γενικά χαρακτηριστικά με τα βασικότερα πλήγματα, τα γραφικά του και η ταχύτητα διαδραστικότητας του!

5.4 ΤΕΛΙΚΟ ΠΟΡΙΣΜΑ ΑΞΙΟΛΟΓΗΣΗΣ

Μετά από μελέτη των παραπάνω βγαίνει το συμπέρασμα ότι το παιχνίδι πέτυχε τον σκοπό του ως εκπαιδευτικό, όμως δεν είναι ένα παιχνίδι που οι μαθητές θα ξετρελαίνονταν να παίζουν. Με τις σημαντικότερες σημειώσεις σε πράγματα που ευτυχώς μπορούν να διορθωθούν σε κάποια μεταγενέστερη έκδοση, όπως τα γραφικά και η ταχύτητα διαδραστικότητας του παιχνιδιού με μια έτοιμη μηχανή γραφικών, χωρίς να χρειαστεί να αλλάξει η βασική λογική του παιχνιδιού. Καθώς σε αυτή υπήρξαν θετικά αποτελέσματα, παραδείγματος χάρη η εκπαίδευση στους αλγορίθμους από τους μαθητές!

(Τα αποτελέσματα δόθηκαν με την μορφή ραβδογράμματος με την βοήθεια του παρακάτω link:

<https://charts.livegap.com/app.php?lan=en&gallery=stackedBar>)

ΚΕΦΑΛΑΙΟ ΕΚΤΟ

ΣΥΜΠΕΡΑΣΜΑΤΑ

6.1 ΕΙΣΑΓΩΓΗ ΣΥΜΠΕΡΑΣΜΑΤΩΝ

Σκοπός της εργασίας ήταν η κατασκευή ενός 2d εκπαιδευτικού παιχνιδιού με την χρήση ευφούς βοηθού για την εκμάθηση τεσσάρων βασικών αλγορίθμων στην γλώσσα (Python). Η υλοποίηση του παιχνιδιού στηρίχθηκε πάνω στο θεωρητικό υπόβαθρο που αναφέρθηκε στην ανασκόπηση της βιβλιογραφίας του καθώς και στην έρευνα άλλων εκπαιδευτικών παιχνιδιών. Η βιβλιογραφία συνοπτικά αφορούσε μεταξύ άλλων θεωριών που συντέλεσαν στην δημιουργία του παιχνιδιού, κυρίως την έρευνα γύρω από τα εκπαιδευτικά ψηφιακά παιχνίδια, τους τέσσερις προαναφερθέντες αλγόριθμους και την χρήση του προσωπικού βοηθού (chatbot). Έπειτα, αναλύθηκε η μεθοδολογία στην οποία έγκειται ο σχεδιασμός, η πλοκή και ευρύτερα η λογική του παιχνιδιού. Στην συνέχεια, διερευνήθηκε και επεξηγήθηκε η συνολική αρχιτεκτονική των δομών του παιχνιδιού και του κώδικα του. Ακόμη, παρουσιάστηκε η αξιολόγηση του από ένα μικρό δείγμα μαθητών τάξεων Γυμνασίου-Λυκείου με μορφή συνέντευξης ερωταπαντήσεων σχετικές με την ψυχαγωγική, λειτουργική και εκπαιδευτική δυνατότητα του παιχνιδιού, που βοήθησε πρόσθετα στην εκτίμηση των στόχων, της διαδικασίας καθώς και του τελικού προϊόντος. Στο κεφάλαιο αυτό θα αναλυθούν τα καταληκτικά πορίσματα από την δημιουργία του παιχνιδιού καθώς και από την αξιολόγηση του.

6.2 ΣΥΜΠΕΡΑΣΜΑΤΑ ΒΙΒΛΙΟΓΡΑΦΙΑΣ ΚΑΙ ΘΕΩΡΗΤΙΚΟΥ ΥΠΟΒΑΘΡΟΥ

Για την δημιουργία του παιχνιδιού μελετήθηκαν ευρύτερα τα εκπαιδευτικά ψηφιακά παιχνίδια και τα ψηφιακά περιβάλλοντα τους. Από αυτή την μελέτη αποκομίστηκαν χρήσιμες γνώσεις για την κατασκευή τους, τους στόχους τους αλλά και την λειτουργικότητα τους ως προς το εκπαιδευτικό κομμάτι. Αρκετοί μαθητές προτιμούν τα εκπαιδευτικά παιχνίδια από την τυποποιημένη εκμάθηση γνώσεων σε αποσπάσματα, διότι υπάρχει αυτή η δυνατότητα αλληλεπίδρασης του μαθητή με το γνωστικό αντικείμενο. Το κείμενο είναι βουβό όμως το παιχνίδι διαθέτει έντονα και γλαφυρά γραφικά, ήχο, μουσική και την δυνατότητα πολλών επιλογών στον χειρισμό. Μόνο και μόνο αυτό το γεγονός καθιστά πιο ψυχαγωγική την εκπαιδευτική διαδικασία αφού κεντρίζει το ενδιαφέρον του μαθητή αλλά συνάμα προσφέρει αυτή την εκμάθηση κανόνων δηλαδή αυτή την επίτευξη του παιδευτικού αποτελέσματος. Το παρόν παιχνίδι διαθέτει και τους τρεις παράγοντες ενός εκπαιδευτικού παιχνιδιού: αλληλεπιδραστικότητα, ψυχαγωγία και εκμάθηση γνωστικού αντικείμενου. Ακόμη αυτή η εκμάθηση ενισχύεται περισσότερο με την χρήση ψηφιακού βοηθού με την

μορφή chatbot που απαντά άμεσα σε απορίες των μαθητών ώστε να μην χάνεται κανένα γνωστικό κομμάτι εξασφαλίζοντας έτσι την μέγιστη εκμαθησιακή δυνατότητα. Δόθηκε η δέουσα προσοχή στην διαμόρφωση ενός ψηφιακού περιβάλλοντος που θα συνδύαζε όλους αυτούς τους παράγοντες και στην αλληλεπίδραση του μαθητή με το εν λόγω υλικό. Ύστερα δόθηκε η δέουσα προσοχή στην εύρεση μίας πλοκής που να ταιριάζει στον τύπο παιχνιδιών περιπέτειας με την επίλυση παζλ-γρίφων που να προκαλεί εντύπωση στους μαθητές και να διεγείρει την σκέψη τους. Τέλος με την χρήση (Python) στην (PyGame) κατασκευάστηκε το παιχνίδι.

6.3 ΣΥΜΠΕΡΑΣΜΑΤΑ ΣΤΗΝ ΑΞΙΟΛΟΓΗΣΗ

Τα συμπεράσματα της αξιολόγησης αναλύθηκαν και στο κεφάλαιο της Αξιολόγησης. Εν κατακλείδι το παιχνίδι εξετάστηκε από ένα μικρό δείγμα μαθητών που έκρινε ότι το παιχνίδι έχει επιτύχει τον εκπαιδευτικό σκοπό του. Γιατί οι μαθητές έμαθαν και τους αλγόριθμους και η εμπειρία τους διευκολύνθηκε από την εύκολη χρήση του μαγικού βιβλίου-ψηφιακού βοηθού. (chatbot). Στον αντίποδα, πιο αρνητικά κρίθηκαν τα γραφικά του παιχνιδιού όπου κατόπιν συζητήσεως με τους μαθητές όλοι συμφώνησαν πως θα προτιμούσαν να ήταν πιο ζωντανά. (πχ.3D).

6.4 ΠΡΟΤΑΣΗ ΓΙΑ ΠΕΡΑΙΤΕΡΩ ΕΡΕΥΝΑ

Πάντοτε υπάρχουν περιθώρια εξέλιξης όπως ακριβώς και σε αυτό το παιχνίδι. Πιο συγκεκριμένα, ορισμένοι παράγοντες που θα μπορούσαν να συμπληρωθούν περισσότερο είναι η δημιουργία επιπρόσθετων πιστών για την κατανόηση και άλλων αλγορίθμων. Επίσης, η δημιουργία μίας τελικής πίστας με (final boss) θα καθιστούσε το παιχνίδι πιο διασκεδαστικό και θα έδινε την ευκαιρία επανάληψης όλων των αλγορίθμων που μαθαίνει ο παίκτης. Τα γραφικά θα μπορούσαν να μετατραπούν σε 3D για μία πιο ζωντανή εμπειρία. Τέλος, μία ακόμη λειτουργία είναι αναγκαία, όπου ο κάθε παίκτης θα διασφάλιζε τον χρήστη του με κωδικό ώστε να μην επιτρέπεται όλοι οι παίκτες να έχουν πρόσβαση σε άλλους χρήστες παρά μόνο στον δικό τους.

Ακαιφαιώνοντας στο τελευταίο κεφάλαιο εξάχθηκαν τα τελικά συμπεράσματα από την κατασκευή του 2D εκπαιδευτικού παιχνιδιού με την χρήση ψηφιακού βοηθού για την εκμάθηση βασικών αλγορίθμων. Συνοψίστηκαν οι βασικές έννοιες του θεωρητικού υπόβαθρου καθώς και ο βασικός σκοπός της δημιουργίας του παιχνιδιού. Εξετάστηκαν τα συμπεράσματα από την αξιολόγηση του σε ένα μικρό δείγμα μαθητών καθώς και παρουσιάστηκαν περαιτέρω προτάσεις βελτίωσης του παιχνιδιού.

ΒΙΒΛΙΟΓΡΑΦΙΚΕΣ ΑΝΑΦΟΡΕΣ

ΕΛΛΗΝΙΚΗ ΒΙΒΛΙΟΓΡΑΦΙΑ

- «Μάθηση και διδασκαλία στην εποχή της πληροφορικής», Α.Ράπτης-Α.Ράπτη, Α τόμος, 2017
- Τεχνολογίες Πολυμέσων: Θεωρία, Υλικό, Λογισμικό, Φώτης Λαζαρίνης, 2007, εκδόσεις Κλειδάριθμος

ΞΕΝΟΓΛΩΣΣΗ ΒΙΒΛΙΟΓΡΑΦΙΑ

- Natural Language Processing in LISP, “An Introduction to Computational Linguistics, Gerald Gazdar, Chris Mellish, 1989
- Εισαγωγή στην ανάλυση και Σχεδίαση Αλγορίθμων, Anany Levitin, εκδόσεις Τζιόλα, 2008
- Analysis of Scratch Software in Scientific Production for 20 Years: Programming in Education to Develop Computational Thinking and STEAM Disciplines, by Pablo Duo Terron
- A Primer on Scientific Programming with Python, Hans Petter Langtangen
- Natural Language Processing Journal) και όνομα άρθρου (GPT-3.5, GPT-4, or BARD? Evaluating LLMs reasoning ability in zero-shot setting and performance boosting through prompts
- Wing, J. Computational Thinking: What and Why?
- Apperly, I., 2010. Mindreaders: The Cognitive Basis Of “Theory of Mind”. Psychology Press
- CLUTRR:
Διαγνωστικό σημείο αναφοράς για επαγωγικό συλλογισμό από κείμενο. Στα: Πρακτικά του 2019 Συνέδριο για τις Εμπειρικές Μεθόδους στην Επεξεργασία Φυσικής Γλώσσας και το 9ο Διεθνές Κοινό Συνέδριο για την Επεξεργασία Φυσικής Γλώσσας. EMNLP- IJCNLP.
- Dalvi et al., 2021

ΗΛΕΚΤΡΟΝΙΚΕΣ ΠΗΓΕΣ

- https://en.wikipedia.org/wiki/The_Legend_of_Zelda,<https://opengameart.org/content/zoria-tileset>
- <https://docs.fileformat.com/web/json>
- <https://www.geeksforgeeks.org/python-modules>
- <https://www.pygame.org/wiki/about>
- <https://www.libsdl.org/>
- <https://www.geeksforgeeks.org/how-to-set-up-the-game-loop-in-pygame/>
- https://en.wikipedia.org/wiki/RGBA_color_model
- https://en.wikipedia.org/wiki/Alpha_compositing
- <https://www.python.org/about/gettingstarted>
- <https://www.pygame.org>
- https://en.wikipedia.org/wiki/The_Legend_of_Zelda:_Breath_of_the_Wild
- <https://app.smartdraw.com/index.aspx?nsu=1>
- <https://www.techwithtim.net/tutorials/game-development-with-python/pygame-tutorial/pygame-tutorial-movement>
- <http://ijiet.com/wp-content/uploads/2015/02/17.pdf>
- https://pygame.readthedocs.io/en/latest/4_text/text.html
- <https://platform.openai.com/docs/guides/gpt>
- <https://theory.stanford.edu/~aiken/publications/papers/ppopp23.pdf>
- <https://realpython.com/python-exec>
- <https://www.nylas.com/blog/the-complete-guide-to-working-with-json>
- <http://technologia.pbworks.com/w/page/33745904/Ερωτηματολόγια%20Εργασιών>
- <https://charts.livegap.com/app.php?lan=en&gallery=stackedBar>
- <http://www.libpng.org/pub/png/book/chapter08.html#png.ch08.div.5.8>
- (https://en.wikipedia.org/wiki/Puzzle_video_game)
- (https://en.wikipedia.org/wiki/Virtual_camera_system#Third-person_view)
- (<https://knowablemagazine.org/article/mind/2019/video-games-educational-benefits>)

ΠΑΡΑΡΤΗΜΑ

ΕΡΩΤΗΜΑΤΟΛΟΓΙΟ ΑΞΙΟΛΟΓΙΣΗΣ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ

1) ΔΙΑΔΡΑΣΤΙΚΟΤΗΤΑ

Εδώ θα ερωτηθείτε σχετικά με το πόσο διασκεδαστικό ήταν το παιχνίδι, σε ένα σκετ με 5 ερωτήσεις:

(Κυκλώστε την απάντηση που σας ταιριάζει)

1. Τα 4 επίπεδα του παιχνιδιού ήταν αρκετά?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
2. Τα επίπεδα του παιχνιδιού ήταν στο κατάλληλο μέγεθος, σχήμα?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
3. Η τα τέρατα και τα κέρματα ήταν αρκετά αντικείμενα αλληλεπίδρασης, εκτός από τα κουτιά του (main story)?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
4. Θα χρειαζόταν το παιχνίδι να έχει (inventory), ή έξτρα λειτουργίες για το παίκτη?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
5. Τα 4 επίπεδα του παιχνιδιού ήταν αρκετά?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου

2) ΕΚΠΑΙΔΕΥΣΗ

Εδώ θα ερωτηθείτε σχετικά με το πόσο εκπαιδευτικό ήταν το παιχνίδι, σε ένα σκετ με 5 ερωτήσεις:

(Κυκλώστε την απάντηση που σας ταιριάζει)

6. Τα κουτιά με τους πάπυρους ήταν πολλά και κουραστικά στο διάβασμα?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
7. Τα κείμενα μέσα σε κάθε κουτί ήταν κατανοητά?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
8. Καταλάβατε και τους 4 αλγορίθμους (spells)?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
9. Ήταν εύκολη η γλώσσα που επιλέχτηκε (Python), για να γράψετε τους αλγορίθμους (spells)?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
10. Σας άφτανε ο χρόνος για να σκεφτείτε το ζόρκι (spell)?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου

3) ΨΗΦΙΑΚΟΣ ΒΟΗΘΟΣ

Εδώ θα ερωτηθείτε σχετικά με το πόσο χρήσιμος και εύχρηστος ήταν ο ψηφιακός βοηθός με την μορφή μαγικού βιβλίου μέσα στο παιχνίδι, σε ένα σετ με 5 ερωτήσεις:

(Κυκλώστε την απάντηση που σας ταιριάζει)

11. Ο ψηφιακός βοηθός σε μορφή βιβλίου ήταν βολικός στην χρήση?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
12. Η επικοινωνία μέσω γραψίματος αυτού του στυλ ήταν καλή επιλογή?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
13. Έδινε στοχευμένες απαντήσεις στις ερωτήσεις σας?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
14. Σας έδινε άσχετες πληροφορίες, όπως ο καιρός σε μία μεγάλη συζήτηση?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
15. Απαντούσε γρήγορα?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου

4) MENOY

Εδώ θα ερωτηθείτε σχετικά με το μενού του παιχνιδιού, σε ένα σετ με 5 ερωτήσεις:

(Κυκλώστε την απάντηση που σας ταιριάζει)

16. Θα θέλατε να υπάρχει επιλογή να φτιάξετε εσείς το (avatar) μάγο σας, όπως στο παιχνίδι του (Minecraft), για παράδειγμα?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
17. Η λίστα με τους καλύτερους παίκτες ήταν όμορφη και άνετη στο διάβαζα?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
18. Σας άρεσε η όψη του αρχικού μενού?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
19. Σας άρεσε η σχεδίαση των επιμέρους μενού?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
20. Ήταν εύκολο στην χρήση?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου

5) ΓΕΝΙΚΕΣ ΠΑΡΑΤΗΡΗΣΕΙΣ

Εδώ θα ερωτηθείτε για γενικά χαρακτηριστικά του παιχνιδιού, σε ένα σετ με 5 ερωτήσεις:

(Κυκλώστε την απάντηση που σας ταιριάζει)

21. Τα γραφικά του παιχνιδιού ήταν, όμορφα σχεδιασμένα?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
22. Τα γραφικά είχαν καλή ανάλυση?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
23. Το παιχνίδι είχε καλή ταχύτητα σε (fps)?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
24. Θα ήθελε και κωδικό για τον κάθε παίκτη εκτός από το όνομα του ή δεν χρειάζεται *(καθώς δεν είναι και τόσο ανταγωνιστικό σαν παιχνίδι)?*
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου
25. Παιζάτε πολύ και τελειώσατε το παιχνίδι χωρίς να βαρεθείτε?
 - i. Πολύ
 - ii. Μέτρια
 - iii. Λίγο
 - iv. Καθόλου