



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

«Ανάπτυξη ηλεκτρονικού παιχνιδιού για την εκμάθηση της πλατφόρμας Unity και προγραμματισμού τεχνητής νοημοσύνης για παιχνίδια»

“Computer game development for learning Unity game engine and programming artificial intelligence for games”

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
ΑΣΗΜΟΓΛΟΥ ΜΙΧΑΗΛ

Επιβλέποντες:

Δασκαλοπούλου Ασπασία, Επίκουρος Καθηγήτρια

Τσαλαπάτα Χαρίκλεια, Ε.ΔΙ.Π

Βόλος, Ιανουάριος 2019

Περίληψη

Με την ανάπτυξη της βιομηχανίας των ηλεκτρονικών παιχνιδιών και την ανάγκη χρήσης καινούργιων μεθόδων στην εκπαιδευτική διαδικασία, ο τομέας των σοβαρών παιχνιδιών προβλέπεται να καταλάβει μια πολύ σημαντική θέση για τη μετάδοση γνώσεων και την ανάπτυξη δεξιοτήτων. Ένα σημαντικό συστατικό των περισσότερων ηλεκτρονικών παιχνιδιών, είναι οι ευφυείς πράκτορες, οι οποίοι αλληλεπιδρούν με τον παίκτη και εμπλουτίζουν τον κόσμο του παιχνιδιού.

Ο στόχος της παρούσας διπλωματικής είναι η σχεδίαση και ανάπτυξη ενός σοβαρού παιχνιδιού χρησιμοποιώντας την πλατφόρμα ανάπτυξης παιχνιδιών Unity και τον προγραμματισμό ευφύων πρακτόρων για παιχνίδια. Συνεπώς, γίνεται παρουσίαση της ιστορίας των σοβαρών παιχνιδιών και των πλεονεκτημάτων που προσφέρουν στην εκπαίδευση, των διαφόρων τεχνικών τεχνητής νοημοσύνης για τον προγραμματισμό ευφύων πρακτόρων και της πλατφόρμας Unity. Ακόμα, εξετάζεται η διαδικασία υλοποίησης του παιχνιδιού και των στοιχείων που το αποτελούν, ενώ στο τελευταίο κεφάλαιο αναπτύσσονται τα συμπεράσματα της εργασίας και προτείνονται μελλοντικές βελτιώσεις.

Abstract

With the development of the video game industry and the need to integrate new methods in the educational process, the field of serious games will most likely play a very important role in transferring knowledge and developing skills. An important component of most video games is intelligent agents who interact with the player and enrich the game world.

The aim of this diploma thesis is to design and develop a serious game using the Unity game development platform and intelligent game agents. Therefore, there is a presentation of the history of serious games, their benefits to education, various techniques of artificial intelligence used for programming intelligent agents and Unity. In addition, there is an analysis regarding the process of implementing the game and its components. Finally, the conclusions drawn as well as propositions for future improvements are examined in the last chapter of the thesis.

Ευχαριστίες

Με την ολοκλήρωση της διπλωματικής μου εργασίας θα ήθελα να ευχαριστήσω τις επιβλέπουσες κ. Δασκαλοπούλου Ασπασία και κ. Τσαλαπάτα Χαρίκλεια για την βοήθεια και την καθοδήγηση που μου προσέφεραν κατά την εκπόνηση της.

Επίσης, ένα πολύ μεγάλο ευχαριστώ στην οικογένεια και τους φίλους μου για την στήριξη τους όλα αυτά τα χρόνια.

Περιεχόμενα

Κεφάλαιο 1 - Εισαγωγή.....	1
Κεφάλαιο 2 – Σοβαρά παιχνίδια.....	3
2.1 Η εξέλιξη των σοβαρών παιχνιδιών.....	3
2.2 Τα οφέλη της χρήσης των σοβαρών παιχνιδιών στην εκπαίδευση.....	5
2.3 Διαφορετικοί μέθοδοι αλληλεπίδρασης των μαθητών με το παιχνίδι.....	7
Κεφάλαιο 3 - Τεχνητή νοημοσύνη σε παιχνίδια.....	9
3.1 Η χρήση της τεχνητής νοημοσύνης σε παιχνίδια.....	9
3.2 Τεχνικές τεχνητής νοημοσύνης για προγραμματισμό παιχνιδιών.....	10
3.2.1 Τεχνικές για τον έλεγχο της συμπεριφοράς των πρακτόρων.....	10
3.2.2 Τεχνικές για την κίνηση και την πλοήγηση των πρακτόρων.....	17
Κεφάλαιο 4 – Unity.....	20
4.1 Τι είναι η Unity.....	20
4.2 Το περιβάλλον εργασίας του Unity.....	22
Κεφάλαιο 5 – Σχεδίαση και ανάπτυξη του παιχνιδιού.....	25
5.1 Η γενική ιδέα του παιχνιδιού.....	25
5.1.1 Ο στόχος του παιχνιδιού.....	25
5.1.2 Ο ρόλος του παίκτη.....	25
5.1.3 Το είδος του παιχνιδιού.....	25
5.1.4 Στοχευμένο κοινό.....	26
5.1.5 Μοντέλο κάμερας.....	26
5.1.6 Πλατφόρμα Υλικού.....	26
5.2 Τα στοιχεία που απαρτίζουν το παιχνίδι.....	26
5.2.1 Το μενού έναρξης.....	26
5.2.2 Οι χαρακτήρες του παιχνιδιού.....	27
5.2.3 Η σκηνή του παιχνιδιού.....	31
5.2.4 Οι μηχανισμοί του παιχνιδιού.....	33
Κεφάλαιο 6 – Αποτίμηση Διπλωματικής.....	37
6.1 Απόδοση παιχνιδιού.....	37
6.2 Βελτιώσεις και προσθήκες για το παιχνίδι.....	38
6.3 Συμπεράσματα.....	39
Βιβλιογραφία.....	40

Κατάλογος Εικόνων

Illustration 1: Ψευδοκώδικας για το παιχνίδι «Pong».....	11
Illustration 2: Οπτική αναπαράσταση του δέντρου αποφάσεων για το παιχνίδι «Pong». 12	12
Illustration 3: FSM controller στην πλατφόρμα Unity.....	13
Illustration 4: Οπτική αναπαράσταση ενός δέντρου συμπεριφοράς.....	14
Illustration 5: Παράδειγμα κώδικα για ένα δέντρο συμπεριφοράς.....	15
Illustration 6: Οπτική αναπαράσταση ενός συστήματος GOAP.....	16
Illustration 7: Κώδικας για την μετακίνηση σε μια τοποθεσία-στόχο.....	17
Illustration 8: Ένας κόσμος παιχνιδιού με σημεία αναφοράς (μπλε) και τα μονοπάτια που τα ενώνουν (κόκκινες γραμμές).....	18
Illustration 9: Οπτική αναπαράσταση του πανmesh στην πλατφορμα Unity.....	19
Illustration 10: Πλατφόρμες που υποστηρίζονται από την Unity.....	20
Illustration 11: Παγκόσμιο μερίδιο αγοράς για τις πλατφόρμες σχεδίασης παιχνιδιών....	22
Illustration 12: Το περιβάλλον εργασίας της Unity.....	23
Illustration 13: Προεπισκόπηση Μενού Έναρξης.....	27
Illustration 14: Προεπισκόπηση Μενού Επιλογών.....	27
Illustration 15: Προεπισκόπηση του μοντέλου του μαθητή.....	28
Illustration 16: Προεπισκόπηση του μοντέλου της Linda.....	29
Illustration 17: Προεπισκόπηση του μοντέλου του Charlie.....	29
Illustration 18: Προεπισκόπηση του μοντέλου του ζόμπι και του FSM που ακολουθεί...30	30
Illustration 19: Κάτοψη της κύριας σκηνής του παιχνιδιού.....	31
Illustration 20: Προεπισκόπηση του εσωτερικού χώρου του σχολείου.....	31
Illustration 21: Προεπισκόπηση του χώρου εκπαίδευσης.....	32
Illustration 22: Κάτοψη του λαβυρίνθου.....	32
Illustration 23: Προεπισκόπηση του σακιδίου.....	33
Illustration 24: Προεπισκόπηση του παράθυρου διαλόγου.....	34
Illustration 25: Προεπισκόπηση της δοκιμασίας πολλαπλών απαντήσεων.....	35
Illustration 26: Προεπισκόπηση των παραθύρων με τον κώδικα του Charlie.....	35
Illustration 27: Προεπισκόπηση του μενού παύσης.....	36
Illustration 28: Benchmark για το παιχνίδι «aIcode».....	37

Κεφάλαιο 1 - Εισαγωγή

Ο όρος «βιντεοπαιχνίδι» αναφέρεται σε οποιοδήποτε παιχνίδι περιλαμβάνει την αλληλεπίδραση του χρήστη με μια ηλεκτρονική συσκευή για την δημιουργία οπτικής ανατροφοδότησης σε μια συσκευή απεικόνισης όπως μια οθόνη τηλεόρασης ή μια οθόνη υπολογιστή. Το πρώτο ηλεκτρονικό παιχνίδι δημιουργήθηκε από τον φυσικό William Higinbotham τον Οκτώβριο του 1958 και ήταν ένα πολύ απλό παιχνίδι τένις. Έκτοτε με την διάδοση των ηλεκτρονικών υπολογιστών, τις κονσόλες αλλά και την χρήση των smartphone, έχουν εξελιχθεί σε μια από τις πιο συνηθισμένες μορφές διασκέδασης τόσο για παιδιά όσο και για ενήλικες. Σύμφωνα με την Clairfield International, ένα διεθνή οργανισμό χρηματοδότησης εταιρειών, υπάρχουν πάνω από 2.5 δισεκατομμύρια παίκτες βιντεοπαιχνιδιών παγκοσμίως και το μέγεθος της παγκόσμιας αγοράς ηλεκτρονικών παιχνιδιών υπολογίζεται να ανέλθει στα περίπου 90.1 δισεκατομμύρια δολάρια το 2020 από 78.6 δισεκατομμύρια δολάρια το 2017(Clairfield International, 2018). Αν αυτός ο υπολογισμός αποδειχθεί σωστός, θα σήμαινε αύξηση περίπου 14.6% γεγονός που δείχνει ότι πλέον τα βιντεοπαιχνίδια αποτελούν ισχυρό κινητήριο μοχλό για την οικονομική ανάπτυξη.

Η χρήση των ηλεκτρονικών παιχνιδιών δεν αφορά αποκλειστικά την ψυχαγωγία και όλο και πιο συχνά δημιουργούνται παιχνίδια τα οποία έχουν ως πρωταρχικό σκοπό την εκπαίδευση του χρήστη. Στην παρούσα διπλωματική εργασία εξετάζουμε τα παιχνίδια αυτού του τύπου και συγκεκριμένα την υλοποίηση ενός παιχνιδιού το οποίο έχει ως στόχο την εκμάθηση χρήσης της μηχανής παιχνιδιών Unity και τον προγραμματισμό τεχνητής νοημοσύνης για παιχνίδια. Αρχικά, θα δούμε τον ρόλο των σοβαρών παιχνιδιών (serious games) στην εκπαίδευση και μια ιστορική αναδρομή από όταν πρωτοεμφανίστηκαν μέχρι τώρα. Στο τρίτο κεφάλαιο, αναλύεται η χρήση της τεχνητής νοημοσύνης στα βιντεοπαιχνίδια και η εξέλιξή της με το πέρασμα των χρόνων και στο επόμενο παρουσιάζεται η μηχανή Unity, που χρησιμοποιήθηκε για την υλοποίηση του παιχνιδιού. Στο πέμπτο κεφάλαιο γίνεται αναλυτική παρουσίαση του παιχνιδιού, το οποίο αναπτύχθηκε στα πλαίσια αυτής της διπλωματικής, ο σχεδιασμός και η υλοποίηση του. Τέλος, το έκτο

κεφάλαιο περιλαμβάνει τα συμπεράσματα τα οποία προέκυψαν καθώς και προτάσεις για περαιτέρω επέκταση της εργασίας.

Κεφάλαιο 2 – Σοβαρά παιχνίδια

2.1 Η εξέλιξη των σοβαρών παιχνιδιών

Με τον όρο σοβαρό παιχνίδι (serious game) εννοούμε ένα παιχνίδι σχεδιασμένο με πρωταρχικό σκοπό διαφορετικό από την ψυχαγωγία (Djaouti, Alvarez, Jessel, 2011). Τα παιχνίδια αυτά χρησιμοποιούνται από πολλούς διαφορετικούς τομείς όπως τον στρατό, την εκπαίδευση, την υγειονομική περίθαλψη και την μηχανική.

Η ιδέα ότι ένα παιχνίδι μπορεί να χρησιμοποιηθεί και για κάτι άλλο, πέρα από την διασκέδαση, είναι πολύ παλιά. Ήδη από την εποχή της αρχαίας Ελλάδας ο Πλάτωνας υποστήριζε «Ποτέ λοιπόν, καλέ μου, να μην μεταχειρίζεσαι την βία στα μαθήματα των παιδιών, αλλά φρόντιζε μάλλον παίζοντας να εκπαιδεύονται, για να είσαι και εσύ καλύτερα σε θέση να διακρίνεις την φυσική του καθενός προδιάθεση» (Πλάτωνας, Cornford, 1972). Θεωρούσε, ότι ενισχύοντας μια συμπεριφορά σε ένα παιδί μέσω ενός παιχνιδιού, θα ενισχυόταν αυτή η συμπεριφορά και ως ενήλικα. Τα πρώτα σοβαρά παιχνίδια ήταν επιτραπέζια, όπως το Chaturanga, που εμφανίστηκε στην Ινδία τον έβδομο αιώνα π.Χ. και μετέπειτα το σκάκι, τα οποία χρησιμοποιήθηκαν για να εφαρμόσουν μια μιλιταριστική μεταφορά σε ένα παιχνίδι.

Δύο δεκαετίες μετά την εφεύρεση του πρώτου ηλεκτρονικού υπολογιστή δημιουργήθηκε και το πρώτο εκπαιδευτικό βιντεοπαιχνίδι. Το «Logo» σχεδιάστηκε το 1967 από τους Wally Feurzeig, Seymour Papert και Cynthia Solomon (Abelson, Goodman, Rudolph, 1974) και συνδύαζε τα μαθηματικά και τον προγραμματισμό για να εκπαιδεύσει τους παίκτες στους βασικούς άξονες του προγραμματισμού, καλώντας τους να κατευθύνουν έναν δρομέα ποντικιού σε σχήμα χελώνας για να σχεδιάσουν γραμμές. Το 1971 αναπτύχθηκε από το Minnesota Computing Consortium ένα από τα πιο γνωστά σοβαρά παιχνίδια, το «The Oregon Trail». Ο στόχος του ήταν να διδάξει την αμερικάνικη γεωγραφία και ιστορία, δίνοντας στον παίκτη τον ρόλο ενός οδηγού άμαξας το 1847, ο οποίος έπρεπε να οδηγήσει το караβάνι από το Μισούρι στο Όρεγκον. Μέχρι και το 1977 η αγορά των εκπαιδευτικών παιχνιδιών ήταν πολύ περιορισμένη, καθώς μόνο εκπαιδευτικά ιδρύματα (σχολεία και πανεπιστήμια) ήταν σε θέση να αγοράζουν αυτά τα προγράμματα.

Με την κυκλοφορία του προσωπικού ηλεκτρονικού υπολογιστή «Apple II» το 1977 και του «Commodore 64» το 1982, που συνοδεύτηκε από την εφεύρεση των CD-ROM, η βιομηχανία των σοβαρών παιχνιδιών άνθησε. Η προσβασιμότητα που επήλθε έκτοτε είχε ως αποτέλεσμα την ίδρυση πολλών εταιρειών οι οποίες είχαν ως στόχο να παρέχουν έναν καινούργιο και διασκεδαστικό τρόπο εκμάθησης. Ένα από τα πιο δημοφιλή παιχνίδια που αναπτύχθηκαν εκείνη την περίοδο είναι το «Reader Rabbit», από την The Learning Company το 1983. Μέσα από mini-games οι μαθητές εξοικειώνονταν με το αλφάβητο και κέρδιζαν βασικές δεξιότητες ανάγνωσης και ορθογραφίας. Η επιτυχία αυτού του παιχνιδιού αντικατοπτρίζεται και από το γεγονός ότι η τελευταία του κυκλοφορία ήταν το 2013. Άλλα πολύ γνωστά παιχνίδια εκείνης της εποχής ήταν το «Where in the World is Carmen Sandiego?» που κυκλοφόρησε το 1985 από την εταιρεία Broderbund και το «Math Blaster» το 1987 από τους Davidson & Associates. Ο στόχος του πρώτου είναι η εκμάθηση γεωγραφίας, καθώς ο παίκτης καλείται να κυνηγήσει κλέφτες σε όλο το κόσμο απαντώντας ερωτήσεις γνώσεων, ενώ στο δεύτερο ο παίκτης παίρνει τον ρόλο ενός εξωγήινου που ψάχνει τον σκύλο του και πρέπει να πυροβολήσει τις σωστές απαντήσεις στις μαθηματικές εξισώσεις που εμφανίζονται στην οθόνη του.

Η δυνατότητα σύνδεσης στο διαδίκτυο μέσα από προσωπικούς υπολογιστές το 1991 άλλαξε για πάντα όχι μόνο τον τομέα των «serious games», αλλά γενικότερα των υπολογιστών. Μέσω του διαδικτύου, άτομα από διαφορετικές χώρες του κόσμου μπορούσαν πλέον να έρθουν σε επικοινωνία, να ανταλλάξουν ιδέες, γνώσεις και φυσικά να παίξουν παιχνίδια. Μέσα από αυτά, δεν αποκτούσαν πλέον μόνο γνώσεις πάνω σε συγκεκριμένους τομείς, αλλά μάθαιναν πώς να συναναστρέφονται με ανθρώπους που έχουν διαφορετική γλώσσα και πολιτισμικό υπόβαθρο. Επίσης, δημιουργήθηκαν διαδικτυακά καταστήματα καθιστώντας την απόκτηση νέων τίτλων πολύ πιο εύκολη και εμφανίστηκαν πολλά περισσότερα mini-games, τα οποία μπορούσες να παίζεις σε απευθείας σύνδεση και τα οποία συνήθως παρέχονταν δωρεάν.

Τέλος, τα τελευταία χρόνια η κυκλοφορία εξοπλισμού που επιτρέπει την χρήση ειδικής και επαυξημένης πραγματικότητας από το προσωπικό χώρο του καθενός, αναμένεται να παίξει μεγάλο ρόλο στην εξέλιξη των σοβαρών παιχνιδιών. Οι τεχνολογίες αυτές προάγουν την βαθύτερη γνωστική μάθηση, αιχμαλωτίζουν τις αισθήσεις του παίκτη

και του επιτρέπουν του να «βυθιστεί» στα σενάρια που του παρουσιάζονται, καθιστώντας τις έτσι ιδανικές για χρήση στην εκπαίδευση. Όχι μόνο προσφέρουν ένα εικονικό περιβάλλον, όπου μπορούν να αναπτυχθούν οι δεξιότητες των παικτών χωρίς τις πραγματικές συνέπειες της αποτυχίας, αλλά ταυτόχρονα ελαχιστοποιούν το κόστος (Groom, Bailenson, Nass, 2009).

Τα σοβαρά παιχνίδια έχουν αλλάξει πολύ τα τελευταία 50 χρόνια. Ακολουθούν τις τάσεις της τεχνολογίας και της καινοτομίας προσπαθώντας να δημιουργήσουν ένα μείγμα μάθησης και διασκέδασης και πολύ πιθανόν να εξελιχθούν σε ένα καταλυτικό παράγοντα στο τρόπο διάδοσης της γνώσης και δεξιοτήτων.

2.2 Τα οφέλη της χρήσης των σοβαρών παιχνιδιών στην εκπαίδευση

Τα τελευταία χρόνια η τεχνολογία έχει παρουσιάσει ραγδαία εξέλιξη και έχει διεισδύσει σε κάθε τομέα της ζωής μας. Σύμφωνα με τον Spires (2008) οι σημερινοί μαθητές γεννιούνται και μεγαλώνουν σε κοινωνικά και εκπαιδευτικά περιβάλλοντα όπου οι ψηφιακές τεχνολογίες είναι ευρέως διαδεδομένες. Ως αποτέλεσμα, υπάρχει μεγάλη εξάρτηση αυτών των νέων από την τεχνολογία σε ζητήματα που αφορούν την καθημερινή επικοινωνία, την συλλογή πληροφοριών, την διασκέδαση και την μόρφωσή τους. Για να μπορέσει λοιπόν το εκπαιδευτικό σύστημα να ανταποκριθεί στις ανάγκες αυτής της νέας γενιάς, είναι απαραίτητη η χρήση καινούργιων μεθόδων για να προσεγγιστεί η μαθησιακή διαδικασία.

Τα οφέλη που προκύπτουν από τη χρήση των παιχνιδιών στην εκπαίδευση είναι πολλά. Μπορεί να υποστηριχθεί ότι όλα τα παιχνίδια έχουν εκπαιδευτικό χαρακτήρα ακόμα και τα πιο απλά καθώς ο παίκτης καλείται να κατανοήσει και να ακολουθήσει συγκεκριμένους κανόνες για την επίτευξη ενός στόχου. Σύμφωνα με τον Howard – Jones (2011), οι γνώσεις και οι δεξιότητες που αποκτώνται μέσω της μάθησης με βάση το παιχνίδι διατηρούνται για μεγαλύτερο χρονικό διάστημα από τις πληροφορίες άλλων μεθόδων μάθησης, αυξάνοντας έτσι την ικανότητα μνήμης των παιδιών.

Ένα από τα πιο σημαντικά πλεονεκτήματα των σοβαρών παιχνιδιών είναι ότι αποτελούν ένα πολύ χρήσιμο εργαλείο για τους καθηγητές προκειμένου να κινητοποιήσουν τους μαθητές τους. Χρησιμοποιώντας παιχνίδια μέσα στην τάξη, αυξάνεται το κίνητρο για να μάθουν, να προσέχουν και να συμμετέχουν ενεργά σε εργασίες. Τα ψηφιακά παιχνίδια επιτρέπουν επίσης στους μαθητές να αλληλεπιδρούν με την ύλη του εκάστοτε μαθήματος με πρακτικό τρόπο, αντί να τους παρουσιάζονται απλά οι πληροφορίες και να τους ζητούν να τις διατηρήσουν.

Ενώ μερικά παιχνίδια είναι ατομικά, πολλά απαιτούν ομαδική εργασία. Η δυνατότητα αρμονικής συνύπαρξης και συνεργασίας με άλλους είναι μια πολύ σημαντική δεξιότητα, τόσο στα παιδιά όσο και στους ενήλικες. Για να καταφέρουν οι μαθητές να εργαστούν αποτελεσματικά ως ομάδα πρέπει να μάθουν να σέβονται τους συμπαίκτες τους, να παίζουν δίκαια και να ακούν τις γνώμες των άλλων προκειμένου να καταλήξουν σε ένα κοινό συμπέρασμα για την επίλυση των προβλημάτων που τους παρουσιάζονται. Ταυτόχρονα, θέτονται σε ένα περιβάλλον ελεγχόμενου συναγωνισμού, επιτρέποντας στον δάσκαλο να ελέγχει την ανταγωνιστικότητα μεταξύ τους και να προάγει την ευγενή άμιλλα.

Επιπρόσθετα, τα σοβαρά παιχνίδια βοηθούν στην ανάπτυξη της κριτικής σκέψης, την δημιουργικότητα και αποτελούν καλή εξάσκηση για να μάθουν οι μαθητές να επιλύουν γρήγορα προβλήματα που τους παρουσιάζονται. Σύμφωνα με τους Drigas, Karyotaki (2016) τα ηλεκτρονικά παιχνίδια προσφέρουν στους καθηγητές ένα δυναμικό περιβάλλον το οποίο μπορούν να χρησιμοποιήσουν για να φέρουν τα παιδιά αντιμέτωπα με εμπόδια τα οποία απαιτούν γρήγορη αντιμετώπιση και ανοιχτόμυαλη σκέψη. Με αυτόν τον τρόπο μπορούν να αξιολογούν τη δυνατότητα των μαθητών να καταστρώνουν στρατηγικές και να χρησιμοποιούν οποιοδήποτε εργαλείο ή πληροφορία τους παρέχεται, ώστε να ξεπεράσουν αυτές τις προκλήσεις, καθώς και να εξασκούν περαιτέρω αυτές τις ικανότητες.

Τέλος, η χρήση παιχνιδιών στην εκπαίδευση βοηθάει στην εξοικείωση των μαθητών με την τεχνολογία. Στην σημερινή κοινωνία, είναι απαραίτητο να γνωρίζει κάποιος, όχι μόνο στην προσωπική ζωή, αλλά και στο εργασιακό περιβάλλον πώς να χειρίζεται σωστά έναν υπολογιστή και άλλες ηλεκτρονικές συσκευές. Τα παιχνίδια είναι ένα ελεγχόμενο περιβάλλον, όπου τα παιδιά μπορούν να μάθουν τον σωστό και ασφαλή τρόπο

χρήσης των διαφόρων τεχνολογιών, καθώς και τους κινδύνους που κρύβονται από την λανθασμένη και αλόγιστη χρήση τους.

2.3 Διαφορετικοί μέθοδοι αλληλεπίδρασης των μαθητών με το παιχνίδι

Ένας από τους σημαντικότερους παράγοντες, όσον αναφορά τη χρήση των σοβαρών παιχνιδιών στην εκπαίδευση, είναι ο τρόπος με τον οποίο οι μαθητές αλληλεπιδρούν με αυτά. Σύμφωνα με τον Κλεισαρχάκη (2017), αυτό μπορεί γίνει με τρεις διαφορετικές τεχνικές : παίζω, δημιουργώ και μετασχηματίζω.

Με τον όρο «παίζω», εννοεί την περίπτωση, όπου ο εκπαιδευτικός χρησιμοποιεί ένα έτοιμο παιχνίδι (Commercial Off The Shelf). Ένα από τα πλεονεκτήματα αυτής της προσέγγισης είναι ότι το παιχνίδι είναι ολοκληρωμένο και δεν απαιτείται από τον εκπαιδευτικό να έχει ιδιαίτερη τεχνογνωσία. Οι μαθητές καλούνται να τερματίσουν το παιχνίδι, κερδίζοντας γνώσεις στην πορεία και ανατροφοδοτούνται για τις ενέργειες τους σύμφωνα με τον σχεδιασμό του. Το βασικό μειονέκτημα αυτής της μεθόδου είναι ότι τα παιχνίδια αυτά δεν είναι ιδανικά προσαρμοσμένα στις εκπαιδευτικές ανάγκες του κάθε μαθητή.

Στην περίπτωση του «δημιουργώ», οι εκπαιδευόμενοι καλούνται οι ίδιοι να υλοποιήσουν ένα παιχνίδι δουλεύοντας ατομικά ή σε ομάδες και με την βοήθεια του διδάσκοντα. Σε αυτή την μέθοδο, είναι απαραίτητο ο εκπαιδευτής να γνωρίζει τα στάδια σχεδίασης ενός παιχνιδιού και να έχει την τεχνογνωσία προκειμένου να καθοδηγήσει τους μαθητές του. Η προσέγγιση αυτή έχει πολλά πλεονεκτήματα, όπως την αυξημένη δημιουργικότητα, τη συνεργασία και την αντιμετώπιση διαφόρων προκλήσεων που προκύπτουν κατά την σχεδίαση και υλοποίηση. Επιπλέον, η όλη εκπαιδευτική διαδικασία αλλά και το τελικό αποτέλεσμα έχουν τη δυνατότητα να είναι προσαρμοσμένα ειδικά για τις εκπαιδευτικές ανάγκες των συμμετεχόντων.

Καταλήγοντας, στην μέθοδο «μετασχηματίζω» γίνεται χρήση του βασικού σκελετού ενός παιχνιδιού ή κάποιου που έχει ανοιχτό κώδικα. Ο στόχος των μαθητών σε αυτή την περίπτωση είναι να αλλάξουν κάποιες παραμέτρους, ώστε να παίξουν αντιμετωπίζοντας διαφορετικές καταστάσεις. Με αυτόν τον τρόπο εξασκούν την ικανότητα να ανα-

πτύσσουν διαφορετικές στρατηγικές και να προσαρμόζονται στις διαφορετικές συνθήκες που τους παρουσιάζονται. Επίσης, δεν απαιτείται πολύς χρόνος εκμάθησης κάποιου εργαλείου σχεδιασμού βιντεοπαιχνιδιών από τους μαθητές, όπως στην προηγούμενη τεχνική αλλά και σε αυτή την περίπτωση ο εκπαιδευτής πρέπει να είναι εξοικειωμένος με την υλοποίηση παιχνιδιών, ώστε να μπορεί να τους καθοδηγήσει.

Κεφάλαιο 3 - Τεχνητή νοημοσύνη σε παιχνίδια

3.1 Η χρήση της τεχνητής νοημοσύνης σε παιχνίδια

Στα βιντεοπαιχνίδια, με τον όρο τεχνητή νοημοσύνη (AI – artificial intelligence) συνήθως αναφερόμαστε σε ένα σύνολο αλγορίθμων που χρησιμοποιούνται για να καθορίσουν τη συμπεριφορά ευφυών πρακτόρων, γνωστών και ως npc (non player character). Γι' αυτό τον λόγο, η εφαρμογή του όρου «τεχνητή νοημοσύνη» μπορεί να θεωρηθεί εσφαλμένη ονομασία καθώς πολλά παιχνίδια δεν χρησιμοποιούν πραγματικές τεχνικές AI που στοχεύουν στην μηχανική μάθηση, αλλά εστιάζουν σε ένα προκαθορισμένο και περιορισμένο σύνολο αποκρίσεων, βάσει ενός περιορισμένου συνόλου εισόδων, Eaton et al (2016).

Η εφαρμογή της τεχνητής νοημοσύνης σε παιχνίδια υπάρχει από την απαρχή της. Αρχικά, οι περισσότερες έρευνες σε αυτόν τον τομέα επικεντρώνονταν σε επιτραπέζια παιχνίδια. Ο Alan Turing το 1953, επαναπροσδιόρισε τον αλγόριθμο Minimax και τον χρησιμοποίησε για να παίζει σκάκι. Έξι χρόνια αργότερα, ο Arthur Samuel εφηύρε μια από τις μορφές της μηχανικής μάθησης, την reinforcement learning, χρησιμοποιώντας ένα πρόγραμμα, το οποίο μάθαινε πώς να παίζει Ντάμα έχοντας ως αντίπαλο τον εαυτό του. Τελικά το 1997, ο υπερυπολογιστής της IBM, «Deep Blue», χρησιμοποιώντας τον αλγόριθμο Minimax και αρκετές τροποποιήσεις εξειδικευμένες στους κανόνες που διέπουν το σκάκι, κατάφερε να νικήσει τον παγκόσμιο πρωταθλητή Garry Kasparov.

Το 1970 πρωτοεμφανίστηκαν οι παιχνιδομηχανές arcade και μαζί τους παιχνίδια που χρησιμοποιούσαν αντιπάλους με AI. Στην αρχή, οι πράκτορες αυτοί δεν είχαν τη δυνατότητα να παίρνουν αποφάσεις, αλλά ακολουθούσαν συγκεκριμένα μοτίβα και προκαθορισμένες δράσεις, οι οποίες όμως κάποιες φορές επιλέγονταν τυχαία, ώστε να φαίνονται πιο «έξυπνοι» και απρόβλεπτοι. Παιχνίδια αυτής της γενιάς ήταν το «Pong» (1972), το «Qwak!» (1974) και το «Space Invaders» (1978). Με το πέρασμα των χρόνων, τα npc γίνονταν πιο πολύπλοκα, όπως στην περίπτωση του «Pac Man» (1980), όπου τέσσερα φαντάσματα κυνηγούσαν ή απέφευγαν τον παίκτη. Κάθε ένας από τους αντιπάλους συμπεριφερόταν με ξεχωριστό τρόπο, δίνοντας την ψευδαίσθηση στον παίκτη ότι αντιμετωπίζει διαφορετικούς τύπους αντιπάλων αντί για τέσσερα πανομοιότυπα αντίγραφα.

Μετά το 1980, με τη διάδοση των προσωπικών ηλεκτρονικών υπολογιστών, τις αυξημένες επιδόσεις που προσέφεραν και τη ραγδαία ανάπτυξη της βιομηχανίας των ηλεκτρονικών παιχνιδιών, η χρήση ευφών πρακτόρων σε αυτά έγινε όλο και πιο συνήθης. Σύμφωνα με τους Yannakakis, Togelius (2016), κάποια από τα παιχνίδια τα οποία αποτέλεσαν ορόσημα για τις διαφορετικές τεχνικές χρήσης της τεχνητής νοημοσύνης είναι τα εξής: το «Creatures» (Millennium Interactive 1996) που χρησιμοποίησε νευρωνικά δίκτυα για να δημιουργήσει μοντέλα συμπεριφοράς για τα πλάσματα του παιχνιδιού, το «Half-Life» (Valve, 1998) με τις προηγμένες και πολύπλοκες τακτικές που ακολουθούσαν τα npc και το «Black and White» (EA,2000), το οποίο κατάφερε για πρώτη φορά να συνδυάσει επιτυχημένα τεχνικές μηχανικής μάθησης, όπως τα perceptors (ένα μοντέλο σχεδιασμένο να προσομοιώνει την ικανότητα του εγκεφάλου να αναγνωρίζει και να διακρίνει, Engel, Broeck, 2012), δέντρα αποφάσεων, reinforcement learning και ένα γνωστικό μοντέλο πεποίθησης-επιθυμίας-πρόθεσης. Ακόμα, η σειρά «The Sims» (EA, 2000-2017) για τους πολύ ρεαλιστικούς χαρακτήρες, το «Forza Motorsport» (MS Game Studios, 2005), το οποίο χρησιμοποιεί το σύστημα Drivatar για να επεξεργαστεί δεδομένα παικτών και να εκπαιδευτεί, το «F.E.A.R.» (Sierra Entertainment, 2005) που εισήγαγε την χρήση της μεθόδου G.O.A.P. (Goal Oriented Action Planning) και το «Alien: Isolation» (Sega,2014), όπου η στρατηγική του εξωγήινου – εχθρού προσαρμόζεται σύμφωνα με τη συμπεριφορά του παίκτη. Τέλος, το «Read Dead Redemption 2» (Rockstar Games, 2018) προκάλεσε μεγάλη αίσθηση με τους εξαιρετικά πολύπλοκους και ρεαλιστικούς ευφείς πράκτορες τους, οι οποίοι παρατηρούν και την παραμικρή λεπτομέρεια στις κινήσεις και την εμφάνιση του παίκτη και αντιδρούν ανάλογα, δίνοντας την εντύπωση ότι αλληλεπιδράς με πραγματικούς ανθρώπους.

3.2 Τεχνικές τεχνητής νοημοσύνης για προγραμματισμό παιχνιδιών

3.2.1 Τεχνικές για τον έλεγχο της συμπεριφοράς των πρακτόρων

Όπως αναφέρθηκε ήδη νωρίτερα σε αυτό το κεφάλαιο, τα πρώτα βιντεοπαιχνίδια, τα οποία έκαναν χρήση της τεχνητής νοημοσύνης, ήταν πολύ απλά. Στο «Pong» για παράδειγμα, αρκούσε ένας αλγόριθμος της μορφής που φαίνεται στην εικόνα 1. Ακόμα και σε αυτή την πολύ απλή μορφή, φαίνονται τα χαρακτηριστικά που διέπουν έναν ευφύ

πράκτορα. Η ρακέτα «αισθάνεται» που βρίσκεται η μπάλα, «σκέφτεται» την απόφαση που πρέπει να λάβει και τελικά «δρα» καταλλήλως.

```
for every update {  
    if ball is left of paddle {  
        move paddle left  
    }  
    else if ball is right of the paddle {  
        move paddle right  
    }  
}
```

Illustration 1: Ψευδοκώδικας για το παιχνίδι «Pong»

Μια πολύ συνηθισμένη τεχνική για τον προγραμματισμό ηρε είναι η χρήση δέντρων αποφάσεων (decision trees). Είναι ένα σύστημα, όπου οι αποφάσεις αναπαρίστανται σε ένα διάγραμμα σχήματος δέντρου και ο αλγόριθμος το διασχίζει μέχρι να φτάσει σε κάποιο «φύλλο», το οποίο περιέχει την τελική απόφαση για την ενέργεια που πρέπει να κάνει. Κάθε κόμβος μπορεί να είναι κόμβος απόφασης (decision node), κόμβος πιθανότητας (chance node) ή τελικός κόμβος (end node). Στην εικόνα 2, φαίνεται μια αναπαράσταση ενός δέντρου αποφάσεων για τον ψευδοκώδικα που χρησιμοποιήθηκε νωρίτερα. Το γεγονός ότι κάθε απόφαση έχει μόνο μια συνθήκη και δύο πιθανά αποτελέσματα, αποτελεί και το μεγαλύτερο πλεονέκτημα των δέντρων αποφάσεων. Με την χρήση αλγορίθμων, όπως για παράδειγμα ο ID3 (Iterative Dichotomiser 3) ο οποίος εφευρέθηκε από τον Ross Quinlan (1985), μπορούν να κατασκευαστούν αυτόματα από ένα σύνολο δεδομένων, γεγονός που τα καθιστά ένα εξαιρετικά χρήσιμο και αποδοτικό εργαλείο για την ταξινόμηση καταστάσεων.

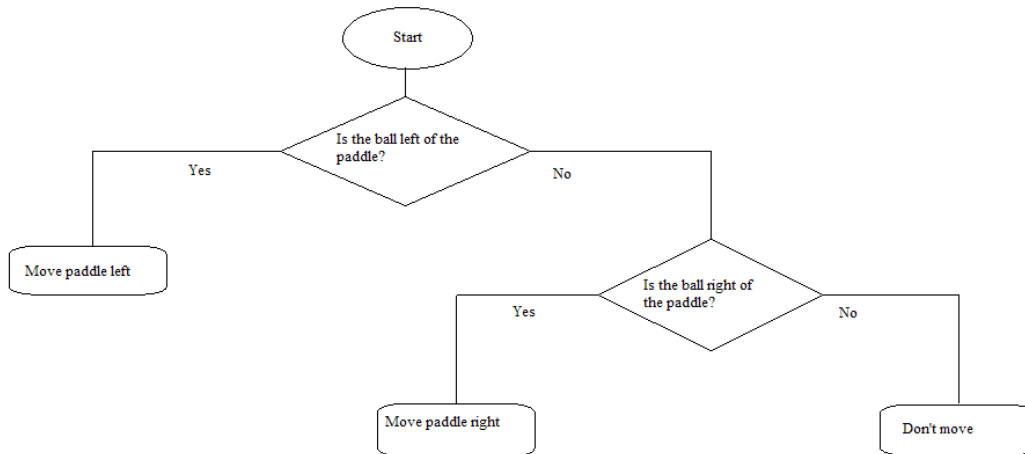


Illustration 2: Οπτική αναπαράσταση του δέντρου αποφάσεων για το παιχνίδι «Pong»

Σε ορισμένες περιπτώσεις οι τεχνικές που περιγράφηκαν παραπάνω, δεν καλύπτουν τις ανάγκες μας. Για παράδειγμα, ο πράκτορας μπορεί να χρειάζεται να λάβει μία απόφαση βάσει της κατάστασης στην οποία βρίσκεται εκείνη την στιγμή, κάτι το οποίο δεν είναι πάντα εύκολο να αναπαρασταθεί με τη χρήση συνθηκών. Άλλες φορές, μπορεί να υπάρχει υπερβολικά μεγάλος αριθμός συνθηκών για να αποδοθούν αποτελεσματικά από ένα δέντρο αποφάσεων ή μπορεί να απαιτείται να γίνει πρώτα μια εκτίμηση των αποτελεσμάτων που θα έχει μια ενέργεια, πριν αυτή εκτελεστεί. Για αυτού του είδους τα προβλήματα είναι αναγκαίες πιο πολύπλοκες λύσεις.

Μια από αυτές είναι οι μηχανές πεπερασμένων καταστάσεων ή όπως αναφέρονται συνήθως στην βιβλιογραφία FSM (Finite State Machines). Με τον όρο FSM εννοούμε μια μηχανή η οποία μπορεί να βρίσκεται σε ακριβώς μία, από ένα σύνολο πεπερασμένων καταστάσεων και χρησιμοποιείται μεταξύ άλλων για να καθορίσει τις συμπεριφορές των χαρακτήρων που ελέγχονται από τον υπολογιστή. Παρουσιάζεται ως γράφημα με κόμβους που αντιπροσωπεύουν τις καταστάσεις και μεταβάσεις που ορίζουν τις διαδρομές μεταξύ τους. Από την στιγμή που ένας χαρακτήρας βρίσκεται σε μια κατάσταση, παραμένει σε αυτήν και συμπεριφέρεται με τον τρόπο τον οποίο εκείνη ορίζει, ωστόσο να ισχύσει κάποια συνθήκη για την μετάβαση σε διαφορετική κατάσταση. Στην εικόνα 3

φαίνεται πώς αναπαριστάται μια μηχανή πεπερασμένων καταστάσεων στην πλατφόρμα Unity.

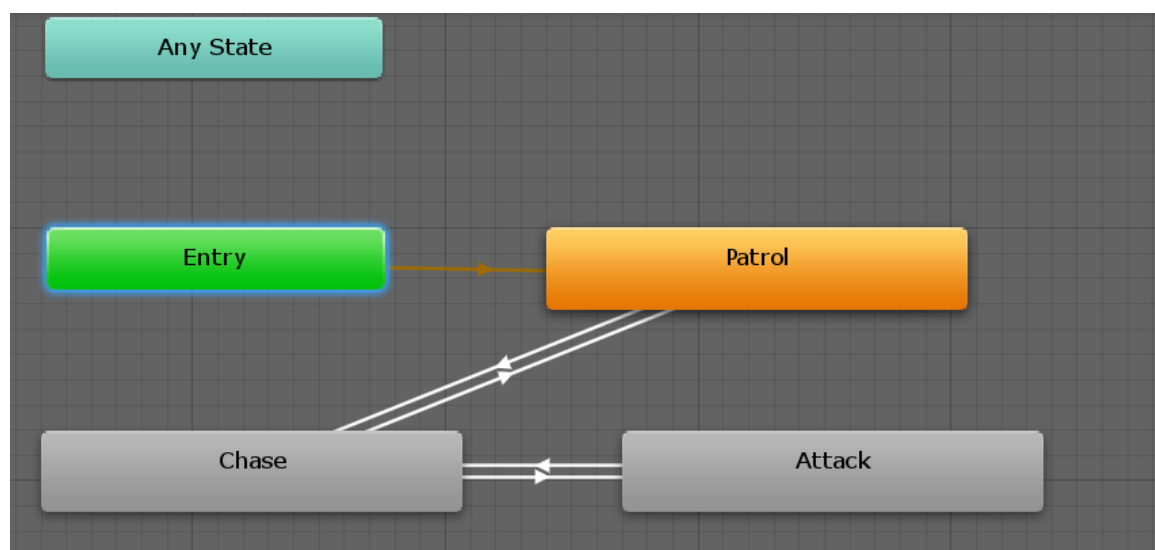


Illustration 3: FSM controller στην πλατφόρμα Unity

Το συγκεκριμένο FSM χρησιμοποιείται για να καθορίσει τη συμπεριφορά ενός φρουρού σε ένα παιχνίδι. Ξεκινάει από την κατάσταση «Entry» και μεταβαίνει απευθείας στην κατάσταση «Patrol», καθώς δεν υπάρχει κάποια συνθήκη για την διαδρομή μεταξύ τους. Όσο βρίσκεται στο «Patrol» περιπολεί την περιοχή μέχρι να βρεθεί σε μια απόσταση τριάντα μέτρων από τον παίκτη οπότε μεταβαίνει στην κατάσταση «Chase» και ξεκινάει να τον κυνηγάει. Αν η απόσταση μεταξύ τους μειωθεί κάτω από δέκα μέτρα, τότε πηγαίνει στην επόμενη και του επιτίθεται, ενώ αν αυξηθεί και πάλι σε πάνω από τριάντα ξαναγυρνάει στην προηγούμενη. Το μειονέκτημα των μηχανών πεπερασμένων καταστάσεων είναι ότι αν ο κάθε πράκτορας σε ένα παιχνίδι χρειάζεται να κάνει πολύπλοκους υπολογισμούς σε κάθε καρέ για να καθορίσει αν πρέπει να μεταβεί σε διαφορετική κατάσταση, αυξάνονται πολύ οι απαιτήσεις από τον επεξεργαστή.

Ακόμα μια τεχνική που χρησιμοποιείται για τον έλεγχο της συμπεριφοράς χαρακτήρων σε παιχνίδια είναι τα δέντρα συμπεριφοράς (behaviour trees),(Colledanchise, 2018). Μοιάζουν αρκετά με τα δέντρα αποφάσεων που αναλύθηκαν νωρίτερα σε αυτό το κεφάλαιο, αλλά έχουν κάποιες σημαντικές διαφορές. Υπάρχουν τρεις τύποι κόμβων, οι σύνθετοι (composite), οι διακοσμητικοί (decorator) και τα φύλλα (leaf). Ένας σύνθετος

κόμβος μπορεί να έχει ένα ή περισσότερα «παιδιά», ένας διακοσμητικός μπορεί να έχει μόνο ένα, ενώ τα φύλλα βρίσκονται στο τέλος κάποιου «κλαδιού» και αντιστοιχούν σε μια πραγματική ενέργεια μέσα στο παιχνίδι. Μια ομάδα κόμβων φύλλων μπορεί να αντιπροσωπεύει μια ακολουθία ενεργειών ή ένα σύνολο από το οποίο επιλέγεται μόνο η μία, αυτό εξαρτάται από τον τύπο του σύνθετου κόμβου από τον οποίο προήλθαν. Οι σύνθετοι μπορεί να είναι είτε τύπου αλληλουχίας (sequence) είτε τύπου επιλογών (selectors). Όταν γίνεται αντικείμενο επεξεργασίας ένας κόμβος αλληλουχίας, διασφαλίζει ότι κάθε «παιδί» του θα δεχτεί επίσκεψη με τη σειρά, ενώ ένας επιλογέας υποδηλώνει ότι μόνο ένα είναι δυνατόν να την δεχτεί. Στην εικόνα 4 φαίνεται με εικονική αναπαράσταση ενός τέτοιου δέντρου.

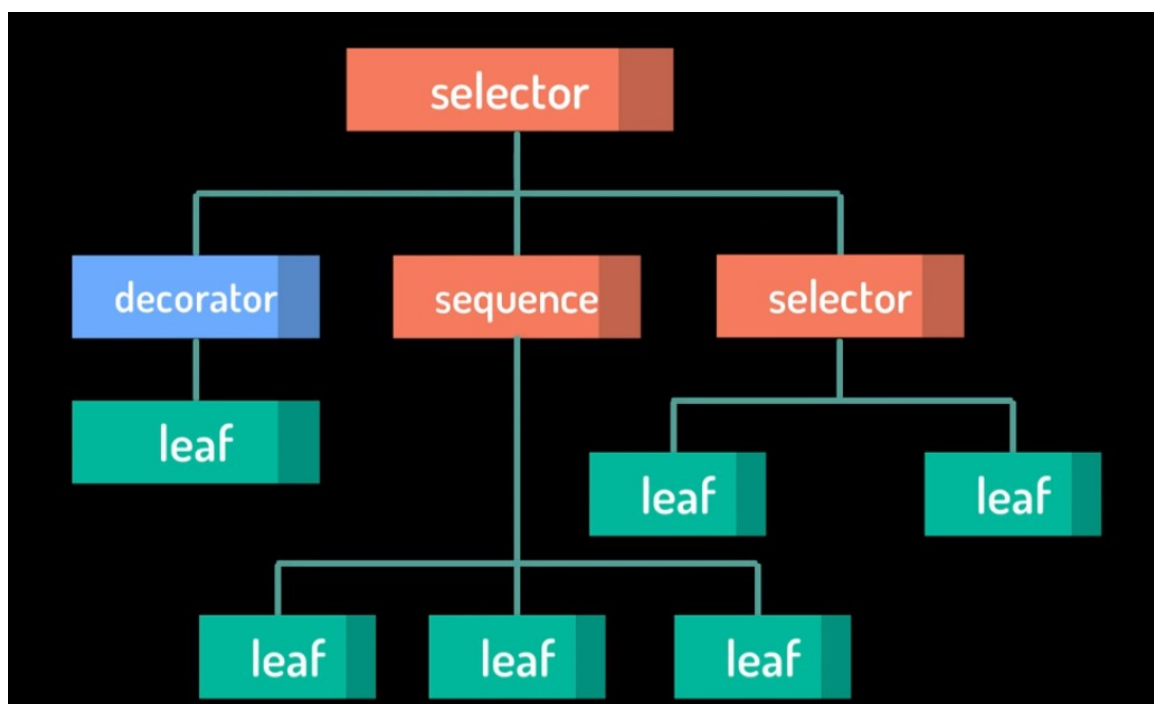


Illustration 4: Οπτική αναπαράσταση ενός δέντρου συμπεριφοράς.

Image source: www.udemy.com/artificial-intelligence-in-unity

Κάθε κόμβος-φύλλο πρέπει να αντιστοιχεί σε μια συνάρτηση η οποία εκτελείται, ενώ το ίδιο το δέντρο συμπεριφοράς γράφεται σε αυτοτελείς μονάδες (όπως φαίνεται στην εικόνα 5), οι οποίες καλούνται, όταν τρέχει κάποιο φύλλο και επιστρέφουν μία από τις τρεις τιμές: επιτυχία (succeed), αποτυχία (fail) ή εξακολουθεί να τρέχει (running). Τα

δέντρα συμπεριφοράς αποτελούν ένα πάρα πολύ ισχυρό εργαλείο και δίνουν τη δυνατότητα σχεδίασης εξαιρετικά πολύπλοκων συμπεριφορών για npc, αλλά είναι αρκετά περίπλοκα καθώς συχνά υπάρχουν πολλοί διαφορετικοί τρόποι για να συντάξουμε το δέντρο και η εξεύρεση του σωστού συνδυασμού διακοσμητή και σύνθετων κόμβων μπορεί να είναι δύσκολη.

```
1
2 tree("Root")
3     sequence
4         SeePlayer
5         TargetPlayer
6         LookAtTarget
7         Wait(0.5)
8         Fire
9
10
11
```

Illustration 5: Παράδειγμα κώδικα για ένα δέντρο συμπεριφοράς

Τέλος, μία ιδιαίτερα δημοφιλής και χρήσιμη τεχνική τεχνητής νοημοσύνης σε παιχνίδια είναι ο σχεδιασμός δράσης με προσανατολισμό στόχου ή αλλιώς GOAP. (Goal Oriented Action Planning). Σχεδιάστηκε από τον Jeff Orkin του Ινστιτούτου Τεχνολογίας της Μασαχουσέτης (MIT), στις αρχές του 2000 και εφαρμόστηκε με επιτυχία στο παιχνίδι F.E.A.R για ελέγξει την συμπεριφορά των npc. Έχει όλα τα στοιχεία μιας μηχανής πεπερασμένων καταστάσεων, αλλά λειτουργεί πολύ διαφορετικά, καθώς στο GOAP οι δράσεις και οι στόχοι αποσυνδέονται. Οι ενέργειες είναι ελεύθερα στοιχεία μέσα στο σύστημα, που συνδυάζονται για να επιτύχουν έναν στόχο, όταν αυτός παρουσιάζεται. Αντί να υπάρχει ένας καθορισμένος κατάλογος ενεργειών που πρέπει να εκτελεστούν για να επιτευχθεί ένας στόχος, όπως στα FSM, ο σχεδιασμός δράσης με προσανατολισμό στόχου επιτρέπει τη δημιουργία πολλών διαφορετικών λύσεων από όλες τις διαθέσιμες δράσεις με βάση την κατάσταση του πράκτορα και του κόσμου. Κάθε ενέργεια έχει προϋποθέσεις (preconditions) που πρέπει να τηρηθούν πριν από αυτή και αποτελέσματα (effects) ως προς την κατάσταση του πράκτορα μετά. Συνδυάζοντας τα αποτελέσματα μιας δράσης με τις προϋποθέσεις άλλων δημιουργούνται διαφορετικές αλυσίδες δράσης για την επίτευξη του στόχου. Προκειμένου να επιλέξουμε μία από όλες τις διαθέσιμες λύσεις, αναθέτουμε

ένα κόστος σε κάθε ενέργεια και η επιλογή γίνεται σύμφωνα με το συνολικό κόστος. Σε περίπτωση που τα κόστη δύο ή περισσότερων λύσεων είναι ίσα η επιλογή γίνεται τυχαία. Στην εικόνα 6 φαίνεται μια οπτική αναπαράσταση ενός συστήματος GOAP.

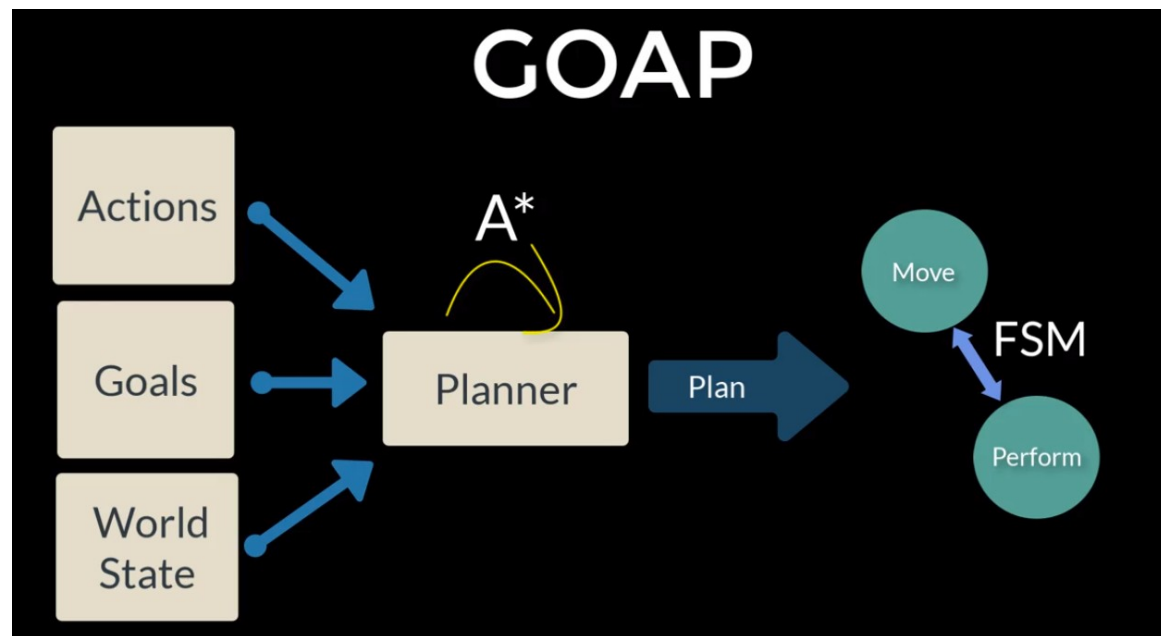


Illustration 6: Οπτική αναπαράσταση ενός συστήματος GOAP

Image source: www.udemy.com/artificial-intelligence-in-unity

Οι δράσεις, οι στόχοι και η κατάσταση του κόσμου τροφοδοτούνται σε έναν προγραμματιστή ο οποίος συνδυάζει τις δράσεις σύμφωνα με τον στόχο και τις αρχικές καταστάσεις για να καθορίσει ποια σχέδια είναι εφικτά. Στη συνέχεια χρησιμοποιεί τον αλγόριθμο A* (A star, Hart, Nilsson, Raphael, 1968) για να καταλήξει στο πιο αποδοτικό πλάνο σύμφωνα με τις τιμές κόστους που έχουν οριστεί. Μόλις δημιουργηθεί το σχέδιο, ο πράκτορας το πετυχαίνει χρησιμοποιώντας μία πολύ απλή μηχανή πεπερασμένων καταστάσεων, που τον μετακινεί στο σημείο όπου πρέπει να λάβει χώρα η δράση, η οποία στην συνέχεια εκτελείται. Πριν από κάθε ενέργεια, πραγματοποιείται έλεγχος για το αν εξακολουθεί να είναι έγκυρη. Αν δεν είναι, τότε ολόκληρο το σχέδιο απορρίπτεται και επιλέγεται κάποιο άλλο. Τα μεγαλύτερα οφέλη αυτής της τεχνικής είναι ότι μπορούν να προστεθούν περισσότερες δράσεις, χωρίς να χρειάζεται να γραφτεί πολύς επιπλέον κώδικας, καθώς αυτές λαμβάνονται υπόψιν αυτόματα από τον προγραμματιστή, ότι αντιδράει

σε αλλαγές του περιβάλλοντος πολύ καλά και ότι κάνει τη συμπεριφορά των πρακτόρων πολύ ρεαλιστική. Το βασικό του μειονέκτημα απαντάται σε περιπτώσεις που η επίτευξη ενός στόχου πρέπει να γίνει με έναν συγκεκριμένο τρόπο, καθώς αυτό απαιτεί πολύ προσεκτικό σχεδιασμό από τον προγραμματιστή και το γεγονός ότι είναι λιγότερο αποδοτικός σε σύγκριση με άλλες μεθόδους, όπως για παράδειγμα τα δέντρα συμπεριφοράς, καθώς βασίζεται στη δυναμική επίλυση προβλημάτων κατά το χρόνο εκτέλεσης.

3.2.2 Τεχνικές για την κίνηση και την πλοήγηση των πρακτόρων

Νωρίτερα σε αυτό το κεφάλαιο αναφερθήκαμε σε διάφορες τεχνικές για τον έλεγχο της συμπεριφοράς των ευφυών πρακτόρων και στα παραδείγματα που παρατέθηκαν συχνά έγινε λόγος για την κίνηση από ένα σημείο σε κάποιο άλλο. Υπάρχουν πολλοί διαφορετικοί τρόποι, με τους οποίους ένας πράκτορας αποφασίζει την πορεία του. Μερικοί από αυτούς αναλύονται στη συνέχεια.

Σε ένα πολύ βασικό επίπεδο, μπορούμε να θεωρήσουμε ότι κάθε πράκτορας έχει μια τιμή ταχύτητας, η οποία καθορίζει το πόσο γρήγορα κινείται και σε ποια κατεύθυνση. Γνωρίζοντας τη θέση του πράκτορα και του στόχου μπορούμε πολύ απλά να υπολογίσουμε την κατεύθυνση στην οποία πρέπει να κινηθεί και να το μετακινήσουμε μέχρι να φτάσει στο στόχο (Εικόνα 7).

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

public class MoveToGoal : MonoBehaviour {

    public float speed = 2.0f;
    public float accuracy = 0.01f;
    public Transform goal;

    // Use this for initialization
    void Start () {

    }

    // Update is called once per frame
    void LateUpdate () {
        this.transform.LookAt(goal.position);
        Vector3 direction = goal.position - this.transform.position;
        if(direction.magnitude > accuracy)
            this.transform.Translate(direction.normalized * speed * Time.deltaTime, Space.World);
    }
}
```

Illustration 7: Κώδικας για την μετακίνηση σε μια τοποθεσία-στόχο

Σε περιπτώσεις όπου το περιβάλλον του παιχνιδιού είναι πιο πολύπλοκο και περιλαμβάνει εμπόδια, χρειαζόμαστε έναν πιο σύνθετο αλγόριθμο, που θα τα λαμβάνει υπόψιν και θα σχεδιάζει την πορεία του πράκτορα αναλόγως. Μια πρώτη προσέγγιση σε αυτή την περίπτωση είναι η χειροκίνητη προσθήκη σημείων αναφοράς (waypoints). Κάθε τέτοιο σημείο αντιπροσωπεύει μια σημαντική θέση στον κόσμο του παιχνιδιού και χρησιμοποιώντας αλγόριθμους, όπως ο A*, μπορούμε εύκολα να βρούμε τη συντομότερη διαδρομή από ένα σημείο σε ένα άλλο, δημιουργώντας έτσι μονοπάτια ανάμεσά τους (Εικόνα 8). Αυτό είναι ένα αρκετά ευέλικτο και ισχυρό σύστημα, αλλά χρειάζεται προσοχή στην τοποθέτηση των σημείων αναφοράς, ώστε να μη δημιουργούνται διαδρομές που δεν συμφωνούν με την γεωμετρία του κόσμου.

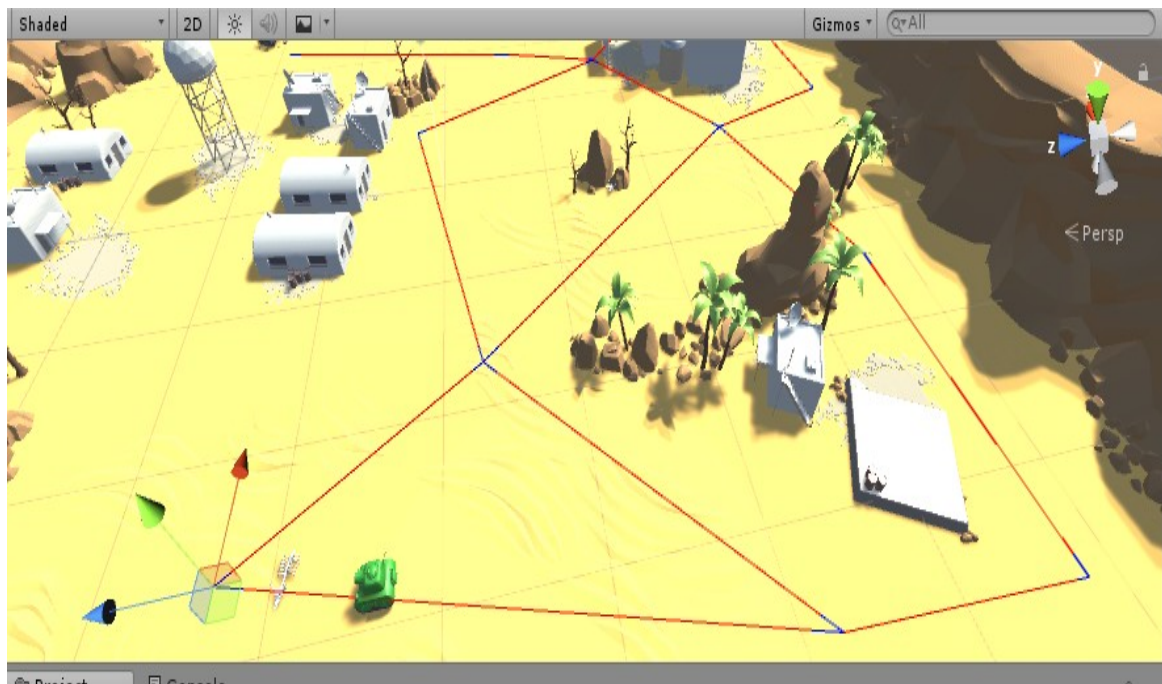


Illustration 8: Ένας κόσμος παιχνιδιού με σημεία αναφοράς (μπλε) και τα μονοπάτια που τα ενώνουν (κόκκινες γραμμές)

Τέλος, μία πάρα πολύ χρήσιμη τεχνική είναι η δημιουργία ενός πλέγματος πλοήγησης (navmesh), δηλαδή ένα δισδιάστατο πλέγμα τριγώνων που επικαλύπτει οποιοδήποτε κομμάτι του κόσμου του παιχνιδιού, στο οποίο επιτρέπεται σε έναν πράκτορα να κινηθεί. Στις περισσότερες πλατφόρμες ανάπτυξης παιχνιδιών υπάρχει η δυνατότητα, δίνοντας τα χαρακτηριστικά των πρακτόρων (ύψος και ακτίνα), να παραχθεί αυτόματα το

πλέγμα πλοήγησης (Εικόνα 9). Στη συνέχεια, χρησιμοποιώντας τον αλγόριθμο A* μπορούμε να αναζητήσουμε μια διαδρομή μέσω αυτού, η οποία θα λαμβάνει υπόψη τη γεωμετρία του χώρου, χωρίς να απαιτείται η χειροκίνητη δημιουργία σημείων αναφοράς.



Illustration 9: Οπτική αναπαράσταση του navmesh στην πλατφόρμα Unity

Κεφάλαιο 4 – Unity

4.1 Τι είναι η Unity

Η Unity είναι μια πλατφόρμα ανάπτυξης παιχνιδιών η οποία κυκλοφόρησε από την εταιρεία Unity Technologies τον Ιούνιο του 2005. Ο στόχος ήταν η δημιουργία μιας οικονομικά προσιτής πλατφόρμας ανάπτυξης παιχνιδιών, που θα περιλάμβανε επαγγελματικά εργαλεία, αλλά που θα ήταν αρκετά απλή, για να επιτρέψει τη χρήση από ερασιτέχνες, ώστε να εκδημοκρατιστεί η βιομηχανία ανάπτυξης παιχνιδιών. Όταν πρωτοκυκλοφόρησε η πλατφόρμα Unity, ήταν διαθέσιμη αποκλειστικά για Mac OS X και τα παιχνίδια τα οποία φτιάχνονταν με την χρήση της υποστήριζαν μόνο ορισμένες πλατφόρμες. Πλέον, η Unity (έκδοση 2018.3) είναι διαθέσιμη για Windows, macOS αλλά και για Linux (σε πειραματική μορφή) ενώ υποστηρίζει πάνω από 25 διαφορετικές πλατφόρμες για την εκτέλεση των παιχνιδιών της (Εικόνα 7).

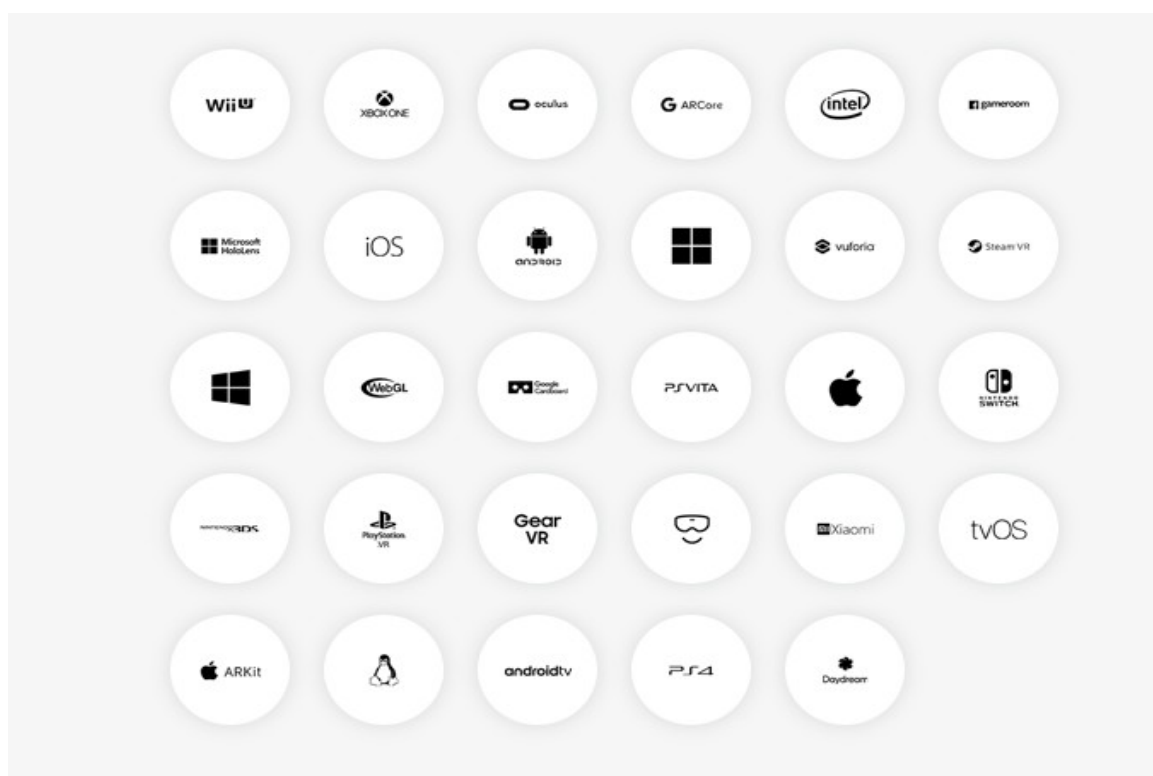


Illustration 10: Πλατφόρμες που υποστηρίζονται από την Unity

Image source : <https://unity3d.com/public-relations>

Η Unity δίνει τη δυνατότητα στους χρήστες να δημιουργούν παιχνίδια για 2D, 3D, εικονική και επαυξημένη πραγματικότητα, προγραμματίζοντας σε C# σε συνδυασμό με τις λειτουργίες drag and drop που παρέχει. Η ίδια η πλατφόρμα έχει προγραμματιστεί σε C και C ++ και υποστηρίζει τις εξής διεπαφές προγραμματισμού εφαρμογών (API): Direct3D για Windows και Xbox One, OpenGL για Linux, OpenGL ES για Android και iOS και WebGL. Ταυτόχρονα, η Unity υποστηρίζει και API χαμηλότερου επιπέδου, όπως Vulkan, iOS Metal, DirectX12, nVidia VRWorks και AMD LiquidVR επιτρέποντας στον χρήστη να επωφελείται από τις βελτιώσεις στις κάρτες γραφικών και το hardware.

Η πλατφόρμα κυκλοφορεί σε τρεις διαφορετικές εκδόσεις (μία δωρεάν και δύο με συνδρομή), τις Personal, Plus και Pro, με τις δύο τελευταίες να απευθύνονται κυρίως σε επαγγελματίες, προσφέροντας περισσότερες δυνατότητες, όπως δωρεάν αποθηκευτικό χώρο στο cloud, πρόσβαση στον πηγαίο κώδικα της Unity και χρήση εργαλείων για αναφορά επιδόσεων. Το μεγαλύτερο ίσως πλεονέκτημα της Unity είναι το γεγονός ότι συνοδεύεται από πλήρη τεκμηρίωση με παραδείγματα για ολόκληρο το API της, σε αντίθεση με άλλες πλατφόρμες που παρέχουν μόνο μερική τεκμηρίωση σε χρήστες που δεν πληρώνουν. Ακόμα, έχει δημιουργηθεί μια πολύ ενεργή διαδικτυακή κοινότητα, η οποία παρέχει βοήθεια σε νέους χρήστες. Τέλος, διαθέτει το λεγόμενο «asset store», στο οποίο ο χρήστης μπορεί να βρει χρήσιμα εργαλεία και αντικείμενα που θα τον βοηθήσουν στη δημιουργία ενός παιχνιδιού, δωρεάν ή επί πληρωμή.

Σύμφωνα με έρευνες που έχουν γίνει και την επίσημη ιστοσελίδα της Unity, η πλατφόρμα αυτή τη στιγμή επικρατεί στη βιομηχανία παραγωγής παιχνιδιών (Εικόνα 8). Πιο συγκεκριμένα, εφαρμογές και παιχνίδια σχεδιασμένα με την Unity χρησιμοποιούνται σε πάνω από 3 δισεκατομμύρια συσκευές παγκοσμίως. Το 50% των παιχνιδιών για κινητά τηλέφωνα και το 60% των παιχνιδιών εικονικής και επαυξημένης πραγματικότητας έχουν κατασκευαστεί χρησιμοποιώντας την Unity, ενώ σύμφωνα με την αναφορά του 2017 για τις ταχύτερα αναπτυσσόμενες δουλειές στην Αμερική από το LinkedIn το Unity καταλαμβάνει την 7 θέση.

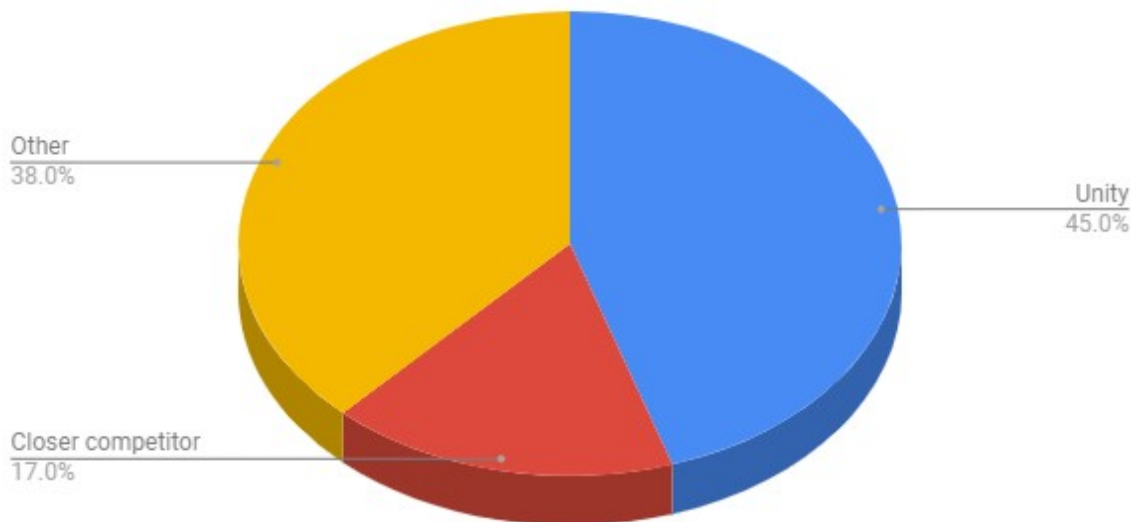


Illustration 11: Παγκόσμιο μερίδιο αγοράς για τις πλατφόρμες σχεδίασης παιχνιδιών

4.2 Το περιβάλλον εργασίας του Unity

Όπως έχει ήδη προαναφερθεί, η Unity αρχικά δημιουργήθηκε ώστε να χρησιμοποιείται από ερασιτέχνες, παρέχοντας τους επαγγελματικά εργαλεία. Γι' αυτό το λόγο η διεπαφή χρήστη της πλατφόρμας είναι αρκετά απλή στην χρήση. Τα βασικά της στοιχεία είναι η γραμμή εργαλείων (Toolbar), το παράθυρο ιεραρχίας (Hierarchy), ο περιηγητής περιεχομένων εργασίας (Project Tab), η καρτέλα επιθεώρησης (Inspector), η προβολή της σκηνής (Scene view) και η προβολή του παιχνιδιού (Game View) (Εικόνα 9). Το κύριο παράθυρο του προγράμματος αποτελείται από ξεχωριστά παράθυρα με καρτέλες τα οποία μπορούν να αναδιαμορφωθούν, να ομαδοποιηθούν, να αποσπαστούν και να συνδεθούν (Unity- Manual State Machine Basics,2018). Αυτό δίνει τη δυνατότητα στον προγραμματιστή να εξατομικεύσει το περιβάλλον εργασίας του ανάλογα με τις προσωπικές του προτιμήσεις και το είδος της εργασίας που πραγματοποιεί. Η προεπιλεγμένη ρύθμιση παρέχει πρακτική πρόσβαση στα πιο κοινά χρησιμοποιούμενα παράθυρα.

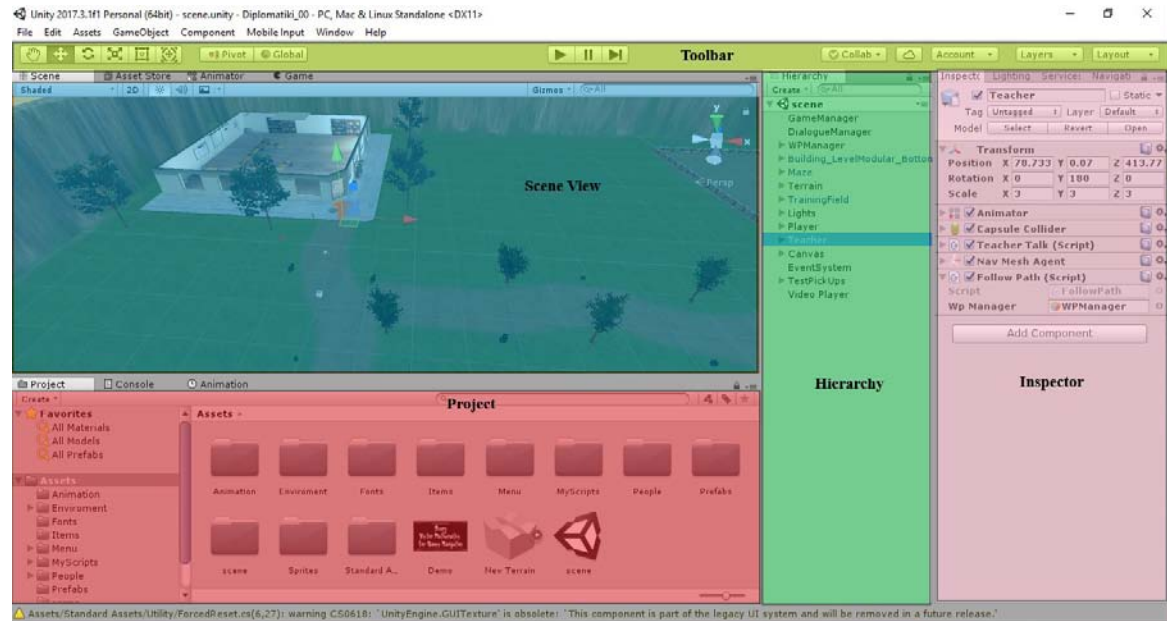


Illustration 12: Το περιβάλλον εργασίας της Unity

Η γραμμή εργαλείων παρέχει τις πιο βασικές λειτουργίες. Στα αριστερά βρίσκονται τα βασικά εργαλεία για τον χειρισμό της προβολής σκηνής και των αντικειμένων μέσα σε αυτήν. Στο κέντρο βρίσκονται τα πλήκτρα αναπαραγωγής, παύσης και βημάτων. Τα κουμπιά στο δεξί μέλος δίνουν πρόσβαση στις υπηρεσίες Unity Cloud και το λογαριασμό του χρήστη στο Unity, ακολουθούμενα από το μενού ορατότητας για τα διάφορα στρώματα που μπορεί να υπάρχουν στο παιχνίδι και το μενού διαρρύθμισης για το περιβάλλον εργασίας. Η γραμμή εργαλείων είναι το μοναδικό κομμάτι της διεπαφής χρήστη της Unity το οποίο δεν μπορεί να αναδιαταχτεί.

Η προβολή σκηνής επιτρέπει τη διαδραστική προβολή του κόσμου που δημιουργεί ο προγραμματιστής σε δισδιάστατη και τρισδιάστατη προοπτική. Μέσα από αυτή ο χρήστης μπορεί να πλοηγηθεί και να επεξεργαστεί τη σκηνή, να τοποθετήσει τοπία, χαρακτηριστές, κάμερες, φώτα και κάθε άλλου είδους αντικείμενα. Η προβολή σκηνής χρησιμοποιείται σε συνδυασμό με την προβολή παιχνιδιού, η οποία χρησιμοποιεί τις κάμερες που υπάρχουν στο παιχνίδι και αποδίδει τη μορφή του τελικού αποτελέσματος επιτρέποντας έτσι στον χρήστη να ελέγξει τι βλέπει ο παίκτης, όταν θα παίζει το παιχνίδι σε όλα τα στάδια υλοποίησης.

Το παράθυρο ιεραρχίας είναι μια ιεραρχική αναπαράσταση σε μορφή λίστας για κάθε αντικείμενο (GameObject) που υπάρχει στη σκηνή. Αποκαλύπτει τη δομή του πώς τα αντικείμενα συνδέονται μεταξύ τους και δίνει τη δυνατότητα στον προγραμματιστή να αναδιατάξει δημιουργώντας εμφωλευμένα αντικείμενα, επιτρέποντάς του έτσι να κληροδοτεί εύκολα χαρακτηριστικά από τα αντικείμενα «γονείς» στα «παιδιά» τους.

Η καρτέλα επιθεώρησης (Inspector) επιτρέπει την προβολή και την επεξεργασία όλων των ιδιοτήτων του επιλεγμένου αντικειμένου, συμπεριλαμβανομένων όλων των προσαρτημένων στοιχείων και των ιδιοτήτων τους. Καθώς υπάρχουν πολλοί διαφορετικοί τύποι αντικειμένων που ο καθένας έχει και διαφορετικές ιδιότητες, η διάταξη και τα περιεχόμενα της καρτέλας επιθεώρησης ποικίλουν. Για παράδειγμα, αν το αντικείμενο περιέχει κώδικα (script), τότε εμφανίζονται οι τιμές των δημόσιων μεταβλητών του.

Ο περιηγητής περιεχομένων εμφανίζει τη «βιβλιοθήκη» όλων των στοιχείων που περιλαμβάνονται στη συγκεκριμένη εργασία. Το αριστερό πλαίσιο εμφανίζει τη δομή των φακέλων της εργασίας σε μορφή ιεραρχικής λίστας. Επιλέγοντας έναν φάκελο και κάνοντας κλικ τα περιεχόμενα του, εμφανίζονται στο δεξί πλαίσιο ως εικονίδια που υποδεικνύουν τον τύπο τους. Επίσης υπάρχει η δυνατότητα εισαγωγής ενός καινούργιου στοιχείου κάνοντας απλά drag and drop, ενώ τέλος η τροποποίηση ενός αρχείου γίνεται χρησιμοποιώντας το κατάλληλο πρόγραμμα, από αυτά που είναι εγκατεστημένα στον υπολογιστή του χρήστη, σύμφωνα με το τύπου, του όπως για παράδειγμα το MonoDevelop για αρχεία κώδικα και το Photoshop για αρχεία εικόνας.

Κεφάλαιο 5 – Σχεδίαση και ανάπτυξη του παιχνιδιού

5.1 Η γενική ιδέα του παιχνιδιού

5.1.1 Ο στόχος του παιχνιδιού

Όπως αναφέρθηκε και στην εισαγωγή, στα πλαίσια αυτής της διπλωματικής δημιουργήθηκε ένα σοβαρό παιχνίδι, το οποίο έχει ως στόχο την εξοικείωση του χρήστη με την πλατφόρμα ανάπτυξης παιχνιδιών Unity και την παρουσίαση διαφόρων τεχνικών τεχνητής νοημοσύνης που χρησιμοποιούνται σε παιχνίδια. Το παιχνίδι ονομάστηκε «aICode» και χρησιμοποιεί τρεις διαφορετικές μεθόδους για να επιτύχει το σκοπό του. Αρχικά, γίνεται χρήση βίντεο για τη μετάδοση των επιδιωκόμενων γνώσεων και ακολούθως ελέγχεται η κατάρτιση του παίκτη πάνω στις απαιτούμενες γνώσεις με τη χρήση δοκιμασιών πολλαπλών απαντήσεων. Τέλος γίνεται οπτική αναπαράσταση των επιπτώσεων που έχουν διαφορετικές τεχνικές προγραμματισμού στη συμπεριφορά ευφυών πρακτόρων με τη χρήση ενός npc, ο οποίος υπακούει στις εντολές του παίκτη και κινείται σύμφωνα με μέθοδο που έχει επιλεγεί. Στη συνέχεια αυτού του κεφαλαίου θα αναλυθούν τα βασικά στοιχεία που αποτελούν το παιχνίδι.

5.1.2 Ο ρόλος του παίκτη

Ο παίκτης ενσαρκώνει τον ρόλο ενός μαθητή, ο οποίος εγγράφεται σε ένα κολέγιο για να εκπαιδευτεί στο σχεδιασμό και την υλοποίηση ηλεκτρονικών παιχνιδιών χρησιμοποιώντας την πλατφόρμα Unity.

5.1.3 Το είδος του παιχνιδιού

Το «aICode» πέρα από σοβαρό παιχνίδι είναι και τύπου περιπέτειας. Η κύρια δραστηριότητα του παίκτη είναι να εξερευνήσει τον κόσμο του παιχνιδιού, να αλληλεπιδράσει με τους ευφυείς πράκτορες και διάφορα αντικείμενα και να επιλύσει τις διάφορες προκλήσεις που του παρουσιάζονται.

5.1.4 Στοχευμένο κοινό

Το παιχνίδι απευθύνεται κυρίως σε παιδιά και εφήβους ηλικίας 12 – 18, που ενδιαφέρονται να μάθουν να χρησιμοποιούν την πλατφόρμα ανάπτυξης παιχνιδιών Unity.

5.1.5 Μοντέλο κάμερας

Το μοντέλο κάμερας που χρησιμοποιήθηκε είναι τρίτου προσώπου. Ο παίκτης βλέπει το μοντέλο του χαρακτήρα του μέσα στο παιχνίδι και μπορεί να την περιστρέψει ανεξάρτητα από αυτό.

5.1.6 Πλατφόρμα Υλικού

Το παιχνίδι «aICode» έχει σχεδιαστεί για να τρέχει σε υπολογιστές με λειτουργικό Windows, Linux και Mac.

5.2 Τα στοιχεία που απαρτίζουν το παιχνίδι

5.2.1 Το μενού έναρξης

Η πρώτη διεπαφή χρήστη είναι το μενού έναρξης. Αποτελεί μια δισδιάστατη σκηνή με ένα φόντο το οποίο περιέχει το όνομα του παιχνιδιού και τέσσερα κουμπιά: Play, Load, Options και Quit (Εικόνα 13). Τα κουμπιά Play και Load φορτώνουν τη βασική σκηνή στην οποία διαδραματίζεται το παιχνίδι και την οποία θα αναλύσουμε αργότερα. Με το κουμπί Options, εμφανίζεται ένα καινούργιο μενού, στο οποίο ο παίκτης μπορεί να αλλάξει βασικές ρυθμίσεις του παιχνιδιού, όπως την ένταση του ήχου με τη χρήση μιας μπάρας ολίσθησης ή να επιστρέψει στο προηγούμενο χρησιμοποιώντας το κουμπί Back (Εικόνα 14). Το κουμπί Quit τερματίζει την εφαρμογή. Τα κουμπιά είναι δυναμικά και αντιδρούν αλλάζοντας χρώμα, όταν ο κέρσορας του ποντικιού βρίσκεται πάνω τους ή όταν ο χρήστης τα επιλέγει.



Illustration 13: Προεπισκόπηση Μενού Έναρξης

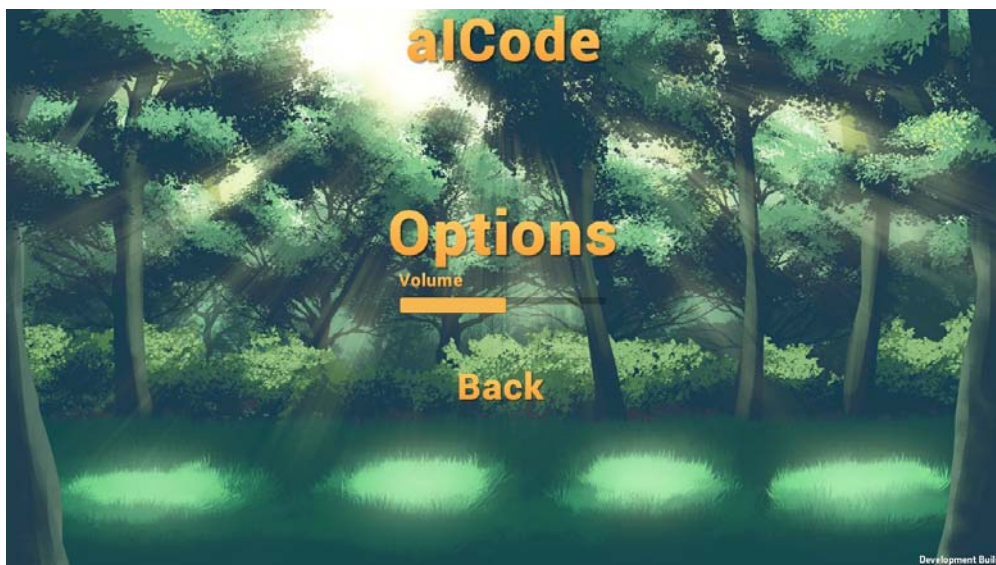


Illustration 14: Προεπισκόπηση Μενού Επιλογών

5.2.2 Οι χαρακτήρες του παιχνιδιού

Στο παιχνίδι υπάρχουν τέσσερις διαφορετικοί χαρακτήρες: ο μαθητής, η Linda, ο Charlie και ένα τέρας – ζόμπι. Ο μαθητής είναι ο χαρακτήρας που χειρίζεται ο ίδιος ο παίκτης (Εικόνα 15). Το μοντέλο που χρησιμοποιήθηκε αποκτήθηκε δωρεάν μέσω του

Unity asset store από το πακέτο «Character Pack: Free Sample» και περιλαμβάνει animations για τις διαφορετικές ενέργειες που μπορεί να επιτελέσει (να περπατήσει, να τρέξει, να πηδήξει στον αέρα και να σκύψει). Ο παίκτης τον χειρίζεται με προοπτική τρίτου προσώπου για να εξερευνήσει τον κόσμο του παιχνιδιού και να αλληλεπιδράσει με τους άλλους χαρακτήρες και αντικείμενα. Η κάμερα, μέσω της οποίας ο χρήστης βλέπει τον κόσμο του παιχνιδιού και είναι προσαρτημένη στο μοντέλο του μαθητή, μπορεί να περιστραφεί ανεξάρτητα από αυτόν, αν και υπάρχει περιορισμός στην γωνία που μπορεί να σχηματίσει με τον κάθετο άξονα (10 – 50 μοίρες), ώστε το οπτικό πεδίο του παίκτη να μην ξεφεύγει από τα όρια του κόσμου του παιχνιδιού.

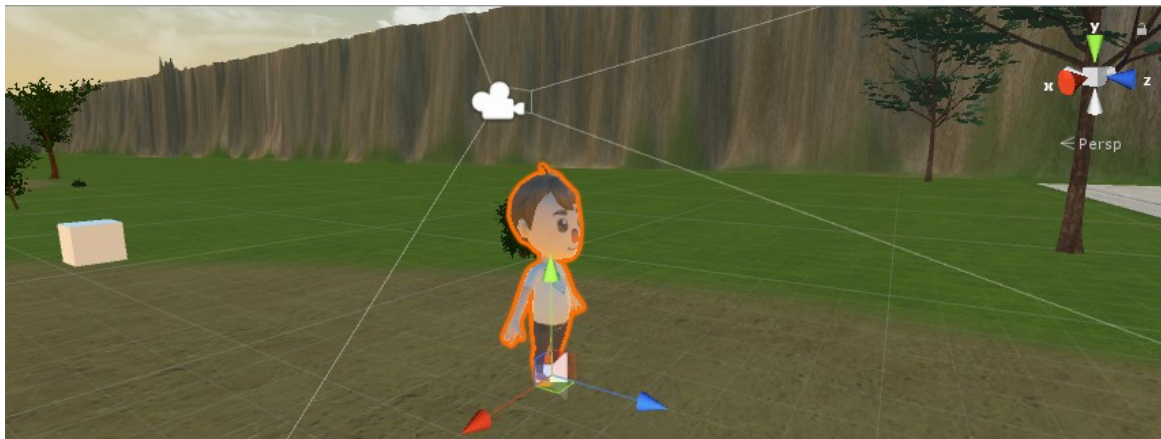


Illustration 15: Προεπισκόπηση του μοντέλου του μαθητή.

Η Linda έχει τον ρόλο της δασκάλας στο παιχνίδι (Εικόνα 16). Ο ρόλος της είναι να οδηγήσει τον παίκτη στους διάφορους χώρους του και να του επισημάνει τους τρόπους με τους οποίους μπορεί να αλληλεπιδράσει με τα διάφορα αντικείμενα που βρίσκονται γύρω του. Η ίδια είναι ένας ευφυής πράκτορας, ο οποίος κινείται στον κόσμο του παιχνιδιού χρησιμοποιώντας ένα συνδυασμό σημείων αναφοράς και του συστήματος navmesh του Unity. Όμοια με τον μαθητή, έχει animations για τις διαφορετικές καταστάσεις στις οποίες μπορεί να βρίσκεται.

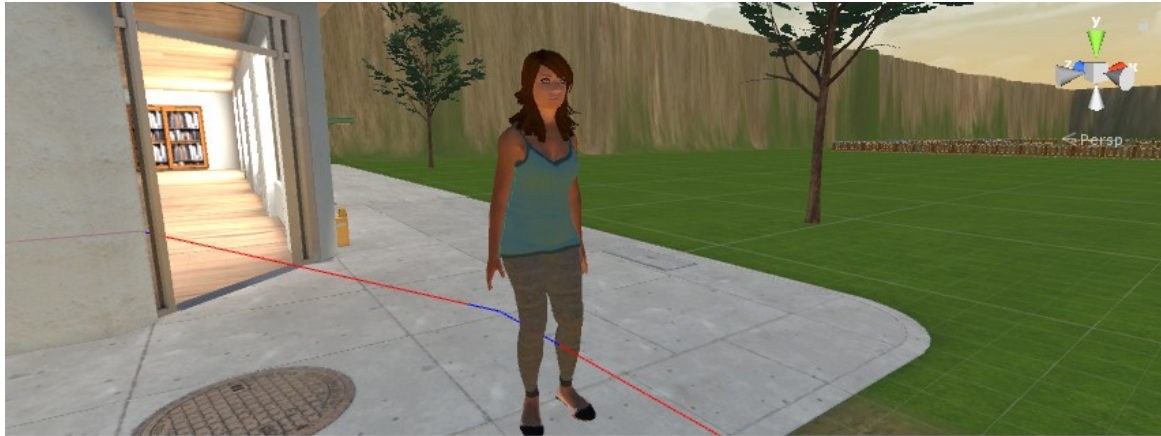


Illustration 16: Προεπισκόπηση του μοντέλου της Linda

Ο Charlie είναι ο κατοικίδιος σκύλος του μαθητή στο παιχνίδι (Εικόνα 17). Το μοντέλο του αποκτήθηκε μέσω του Unity asset store από το πακέτο «Oscar the dog» και περιλαμβάνει πληθώρα διαφορετικών animation. Ο ρόλος του μέσα στο παιχνίδι είναι ιδιαίτερα σημαντικός, καθώς αποτελεί το μέσο για να μπορέσει ο παίκτης να δει τον τρόπο που οι διαφορετικές τεχνικές προγραμματισμού ευφών πρακτόρων επηρεάζουν τη συμπεριφορά τους. Ο τρόπος με τον οποίο γίνεται αυτό περιγράφεται πιο αναλυτικά αργότερα σε αυτό το κεφάλαιο.



Illustration 17: Προεπισκόπηση του μοντέλου του Charlie

Το μοντέλο για το τέρας-ζόμπι αποκτήθηκε μέσω της ιστοσελίδας mixamo.com και είναι ένας ευφύης πράκτορας, ο οποίος λειτουργεί σύμφωνα με μια μηχανή πεπερα-

σμένων καταστάσεων (FSM) (Εικόνα 18). Οι καταστάσεις αυτές είναι: περιπολία (patrol), κυνήγι (chase) και επίθεση (attack). Αρχικά, περιπολεί μια συγκεκριμένη περιοχή του κόσμου του παιχνιδιού, αλλά, αν ο παίκτης το πλησιάσει σε απόσταση μικρότερη των τριάντα μέτρων και βρίσκεται στο οπτικό του πεδίο, τότε θα τον κυνηγήσει. Στην συνέχεια, αν βρεθεί σε απόσταση μικρότερη των πέντε μέτρων εισέρχεται στην επόμενη κατάσταση και επιτίθεται στον παίκτη, αλλιώς, αν καταφέρει ο χρήστης απομακρυνθεί από αυτό περισσότερο από τριάντα μέτρα, ξαναγυρνάει στην προηγούμενη κατάσταση. Ο ρόλος του είναι να δείξει στον παίκτη τον τρόπο με τον οποίο συμπεριφέρεται ένα npc που λειτουργεί βάση ενός FSM.

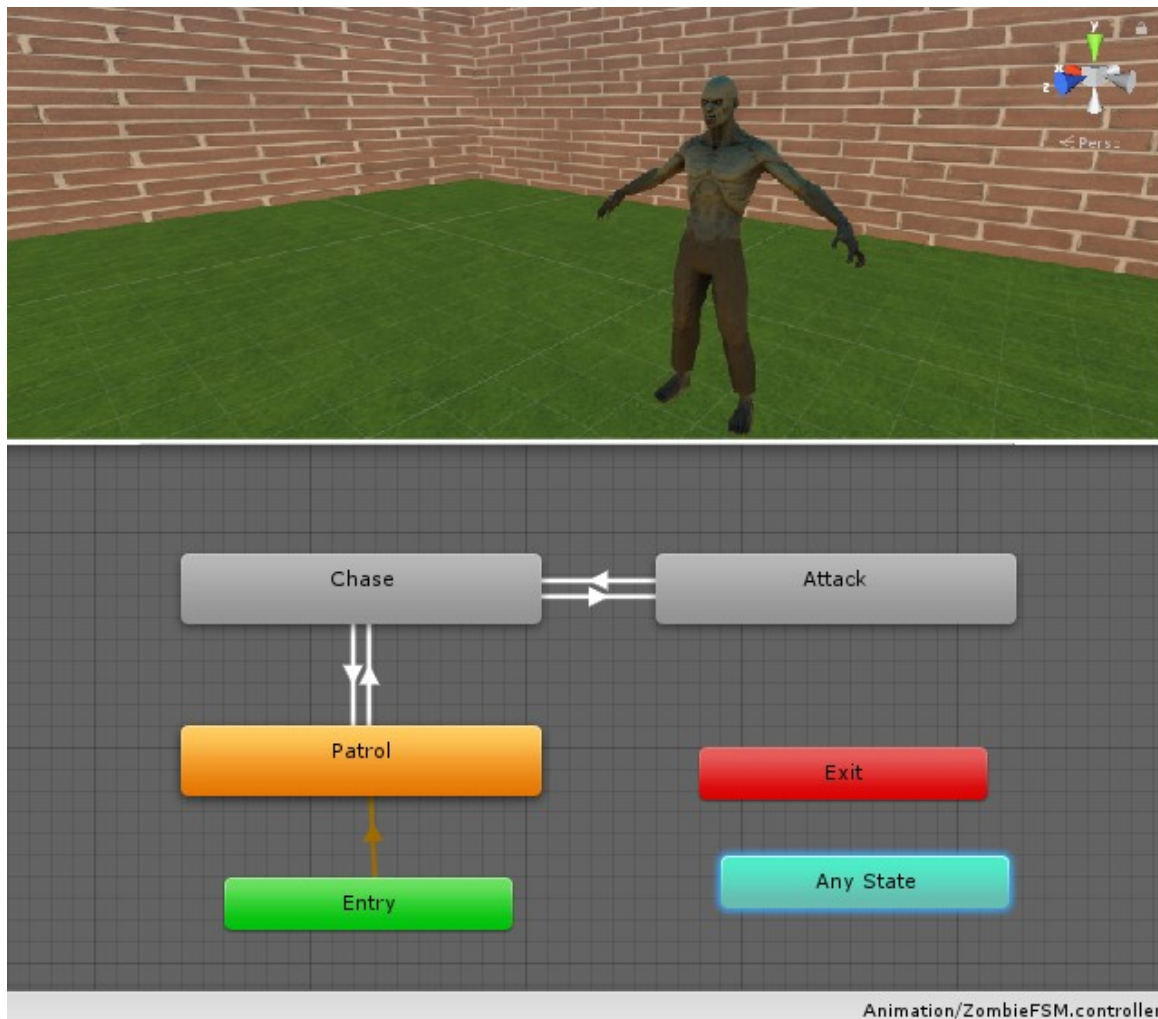


Illustration 18: Προεπισκόπηση του μοντέλου του ζόμπι και του FSM που ακολουθεί

5.2.3 Η σκηνή του παιχνιδιού

Με τη χρήση του περιβάλλοντος σχεδίασης της πλατφόρμας Unity και διαφόρων στοιχείων από το Unity asset store δημιουργήθηκε το γραφικό περιβάλλον όπου λαμβάνει χώρα το παιχνίδι. Πρόκειται για μια τρισδιάστατη σκηνή διαστάσεων 500 επί 500 units (Εικόνα 19) και η οποία περιλαμβάνει τρεις σημαντικές τοποθεσίες: το σχολείο, τον χώρο εκπαίδευσης και τον λαβύρινθο.



Illustration 19: Κάτοψη της κύριας σκηνής του παιχνιδιού

Το σχολείο είναι ο πρώτος χώρος στον οποίο καθοδηγείται ο παίκτης (Εικόνα 20). Εκεί, υπάρχει η δυνατότητα παρακολούθησης εκπαιδευτικών βίντεο, τα οποία αποσκοπούν στη μετάδοση των επιδιωκόμενων γνώσεων ακολουθούμενα από δοκιμασίες πολλαπλών απαντήσεων, ώστε να επιβεβαιωθεί ότι ο μαθητής κατανόησε το περιεχόμενο των διαλέξεων.



Illustration 20: Προεπισκόπηση του εσωτερικού χώρου του σχολείου

Ο χώρος εκπαίδευσης αφορά ένα περιφραγμένο τμήμα, που περικλείει διάφορα εμπόδια και είναι ο δεύτερος χώρος του παιχνιδιού, στον οποίο κατευθύνεται ο παίκτης (Εικόνα 21). Εκεί, έρχεται πρώτη φορά σε επαφή με τον Charlie και παρακινείται να χρησιμοποιήσει τα κουμπιά που εμφανίζονται στην οθόνη του για να παρατηρήσει πώς οι διαφορετικές τεχνικές προγραμματισμού επηρεάζουν τη συμπεριφορά του σκύλου.



Illustration 21: Προεπισκόπηση του χώρου εκπαίδευσης

Ο λαβύρινθος είναι ο χώρος μέσα στον οποίο περιπολεί το τέρας που υπάρχει στο παιχνίδι (Εικόνα 21). Ο σκοπός του είναι να δείξει στον παίκτη πώς ένας πράκτορας, που λειτουργεί με την κατάλληλη μέθοδο προγραμματισμού, μπορεί να κινηθεί σε ένα πολύπλοκο περιβάλλον με πολλά εμπόδια, ενώ ταυτόχρονα του προσφέρει μια συναρπαστική εμπειρία, καθώς καλείται να διασχίσει τον λαβύρινθο χωρίς να τον πιάσει το τέρας-ζόμπι.



Illustration 22: Κάτοψη του λαβυρίνθου

5.2.4 Οι μηχανισμοί του παιχνιδιού

Με τον όρο μηχανισμοί του παιχνιδιού εννοούμε μεθόδους, οι οποίες χρησιμοποιούνται από τους πράκτορες και είναι σχεδιασμένες για να επιτρέπουν την αλληλεπίδραση με τα διάφορα στοιχεία του, παρέχοντας έτσι το λεγόμενο *gameplay* (Sicart, 2008). Μέσα στον κόσμο του παιχνιδιού υπάρχουν χαρακτήρες και αντικείμενα με τα οποία μπορεί να αλληλεπιδράσει ο παίκτης. Γι' αυτό το λόγο δημιουργήθηκε ένα *script*, το οποίο ορίζει την κλάση «*Interactable*» και προσάπτεται σε αυτά τα στοιχεία. Μέσα από αυτό το *script* ελέγχεται αν ο παίκτης κάνει κλικ στο αντικείμενο και αν βρίσκεται εντός της ακτίνας δράσης που ορίζεται για το καθένα. Αν τηρεί αυτές τις προϋποθέσεις, εκτελείται η ενέργεια που ορίζεται για το κάθε στοιχείο.

Ο παίκτης διαθέτει ένα σακίδιο χωρητικότητας δώδεκα θέσεων, το οποίο εμφανίζεται στην οθόνη του πατώντας το πλήκτρο «B» (Εικόνα 23). Όταν το στοιχείο με το οποίο αλληλεπιδράει ο παίκτης ανήκει στην κλάση «*Equipment*», τότε αυτό προστίθεται στο σακίδιο του. Τα αντικείμενα αυτά μπορούν να ανήκουν σε μία από τις τρεις διαφορετικές κατηγορίες εξοπλισμού: *head*, *chest*, *hands*. Ανοίγοντας το σακίδιο μπορεί να επιθεωρήσει τα αντικείμενα τα οποία έχει συγκεντρώσει και είτε να τα χρησιμοποιήσει, είτε να τα καταστρέψει. Αν έχει κάνει ήδη χρήση ενός αντικειμένου, το οποίο ανήκει στην ίδια κατηγορία με αυτό που επέλεξε, τότε κρατά το καινούργιο και επανατοποθετεί το παλιό στο σακίδιο. Υπάρχει επίσης η δυνατότητα να επιστρέψει όλα τα αντικείμενα που χρησιμοποιεί εκείνη την στιγμή στο σακίδιο, πατώντας το πλήκτρο «U». Η χρήση των αντικειμένων που ανήκουν στην κλάση «*Equipment*» διαφέρει ανάλογα με το είδος τους. Για παράδειγμα, αν πρόκειται για μια μπάλα τότε μπορεί να την πετάξει χρησιμοποιώντας «T», ενώ αν είναι ένα καπέλο τότε αλλάζει την εμφάνιση του χαρακτήρα του.



Illustration 23:
Προεπισκόπηση του
σακιδίου

Ένας ακόμα μηχανισμός του παιχνιδιού προέκυψε από την ανάγκη ύπαρξης ενός συστήματος επικοινωνίας ανάμεσα στον παίκτη και στη Linda, ώστε να μπορεί να τον καθοδηγεί. Για αυτό το λόγο δημιουργήθηκε ένα σύστημα διαλόγου. Αλληλεπιδρώντας με την ίδια την Linda ή με αντικείμενα τα οποία χρήζουν επεξήγησης εμφανίζεται στην οθόνη του χρήστη ένα παράθυρο που περιέχει τον διάλογο (Εικόνα 24). Ο διάλογος εμφανίζεται σταδιακά και για συνεχιστεί, ο παίκτης πρέπει να πατήσει το πλήκτρο «Continue». Όταν ολοκληρωθεί, το παράθυρο εξαφανίζεται.



Illustration 24: Προεπισκόπηση του παράθυρου διαλόγου

Όπως έχει ήδη αναφερθεί, στο παιχνίδι γίνεται χρήση εκπαιδευτικών βίντεο. Αυτά προβάλλονται, όταν ο παίκτης αλληλεπιδράει με την οθόνη του προβολέα που βρίσκεται στον χώρο του σχολείου. Μετά την ολοκλήρωσή τους ακολουθεί μια δοκιμασία πολλαπλών απαντήσεων, όπως φαίνεται στην Εικόνα 25. Οι ερωτήσεις επιλέγονται τυχαία από μια προκαθορισμένη λίστα. Όταν ο παίκτης επιλέγει μια απάντηση, το παιχνίδι τον ενημερώνει αν ήταν σωστή ή λανθασμένη. Σε περίπτωση σωστής απάντησης εμφανίζεται η επόμενη, ενώ στην αντίθετη περίπτωση ο παίκτης πρέπει να προσπαθήσει να απαντήσει ξανά στην ίδια ερώτηση.

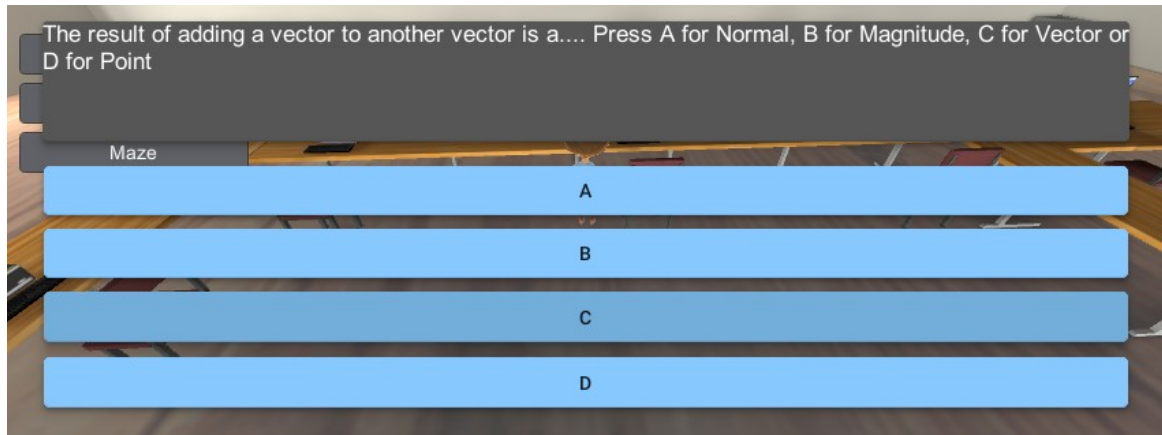


Illustration 25: Προεπισκόπηση της δοκιμασίας πολλαπλών απαντήσεων

Στη συνέχεια, αναπτύχθηκε ένα σύστημα ώστε να μπορεί ο παίκτης να βλέπει και να επιλέγει τη μέθοδο που ελέγχει τον τρόπο, με τον οποίο συμπεριφέρεται ο Charlie. Πατώντας το πλήκτρο «G» εμφανίζονται στην οθόνη δύο παράθυρα, τα οποία περιέχουν των κώδικα που καθορίζει την συμπεριφορά του σκύλου (Εικόνα 26).

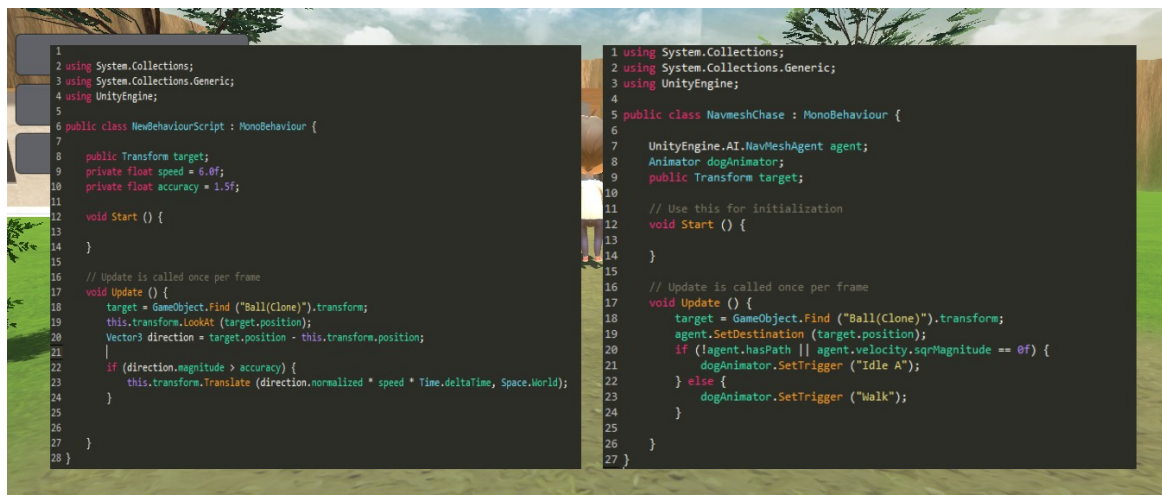


Illustration 26: Προεπισκόπηση των παραθύρων με τον κώδικα του Charlie

Κάνοντας κλικ σε ένα από τα παράθυρα, ενεργοποιείται και ο αντίστοιχος κώδικας. Αν για παράδειγμα ο παίκτης επιλέξει το πρώτο παράθυρο και πετάξει στον Charlie μια μπάλα, τότε ο σκύλος θα κινηθεί προς το μέρος της, χωρίς όμως να λαμβάνει υπόψιν του πιθανά εμπόδια που βρίσκονται μπροστά του. Αν επιλέξει το δεύτερο, τότε ο Charlie

θα κινηθεί ξανά προς την τοποθεσία της μπάλας αλλά αυτή τη φορά θα κινηθεί πιο έξυπνα λαμβάνοντας υπόψιν τη γεωμετρία του χώρου, καθώς χρησιμοποιεί το σύστημα navmesh του Unity. Αξίζει να σημειωθεί ότι, αν ο παίκτης κάνει απλά κλικ στον ίδιο τον Charlie, τότε αυτός τον ακολουθεί χρησιμοποιώντας και πάλι το σύστημα navmesh.

Τέλος, πατώντας το πλήκτρο «Esc» το παιχνίδι τίθεται σε παύση και εμφανίζονται τρία κουμπιά στην οθόνη: Resume, Menu και Quit (Εικόνα 27). Αν επιλέξει το πρώτο, τότε το παιχνίδι συνεχίζει. Το κουμπί Menu τον μεταφέρει στο μενού έναρξης ενώ το Quit τερματίζει την εφαρμογή.



Illustration 27: Προεπισκόπηση του μενού παύσης

Κεφάλαιο 6 – Αποτίμηση Διπλωματικής

6.1 Απόδοση παιχνιδιού

Χρησιμοποιώντας το εργαλείο Unity Profiler, καταγράψαμε την απόδοση του παιχνιδιού σε πραγματικό χρόνο (Εικόνα 28). Το εργαλείο αυτό μετράει τον χρόνο, ο οποίος δαπανάται στις διάφορες περιοχές του παιχνιδιού και παρουσιάζει τα αποτελέσματα σε μορφή γραφικών παραστάσεων. Με αυτό τον τρόπο μπορεί να αναλυθεί η απόδοση της CPU, της GPU, της μνήμης και του ήχου. Ο έλεγχος του παιχνιδιού «aIcode» έγινε σε ρύθμιση γραφικών «Ultra», χρησιμοποιώντας φορητό υπολογιστή με τα εξής χαρακτηριστικά:

- Λειτουργικό σύστημα: Windows 10 Home – 64 bit
- CPU: Intel Core i5 3230M @ 2.60GHz
- RAM: 8,00GB Dual-Channel DDR3 @ 798MHz
- GPU: Nvidia GeForce 710M 2047MB
- Monitor: Generic PnP Monitor (1366 x 768 @ 60Hz)

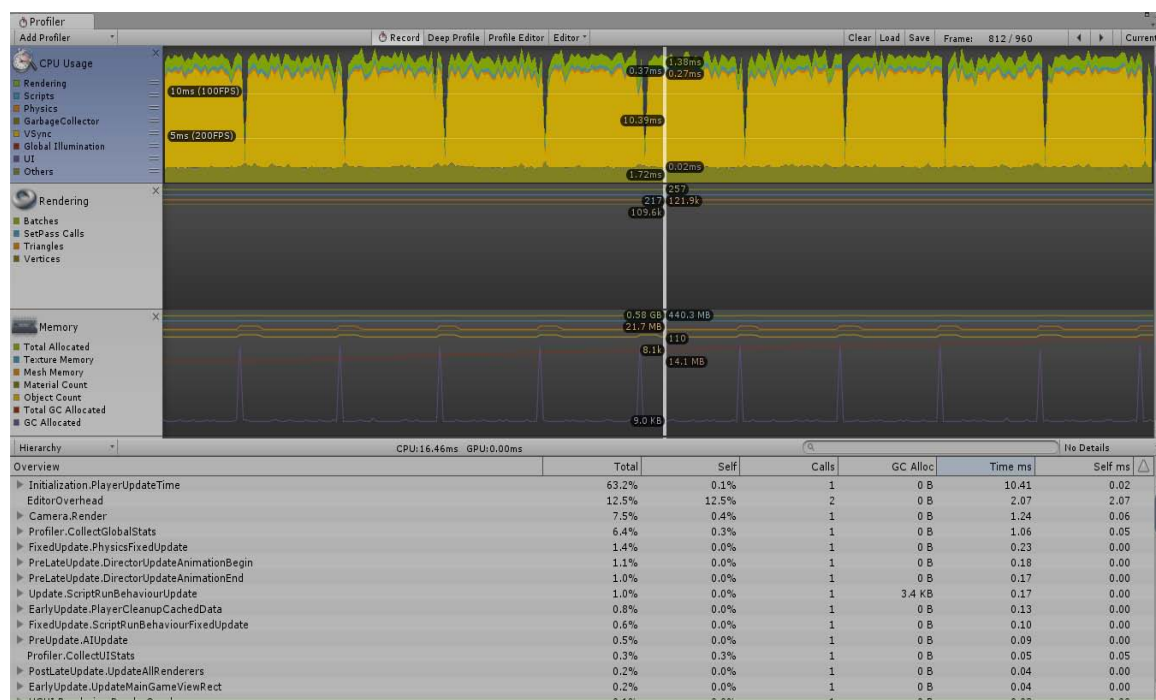


Illustration 28: Benchmark για το παιχνίδι «aIcode»

Τα αποτελέσματα των μετρήσεων δείχνουν ότι η απόδοση του παιχνιδιού είναι πολύ καλή, ακόμα και όταν τρέχει στην υψηλότερη ποιότητα δεδομένου των χαρακτηριστικών του υπολογιστή που χρησιμοποιήθηκε. Ο υπολογιστής χρειάζεται κατά μέσο όρο 16.46 ms για να σχεδιάσει ένα στιγμιότυπο (frame), το οποίο σημαίνει ότι επιτυγχάνουμε 60.75 στιγμιότυπα το δευτερόλεπτο. Επίσης από τις μετρήσεις της κύριας μνήμης φαίνεται ότι χρειάζεται ανάθεση μόλις 0.58GB συνολικά. Οι μετρήσεις αυτές δείχνουν ότι το παιχνίδι το οποίο αναπτύχθηκε κατά την εκπόνηση αυτής της διπλωματικής, μπορεί να εκτελεστεί στους περισσότερους υπολογιστές με αξιοπρεπή απόδοση.

6.2 Βελτιώσεις και προσθήκες για το παιχνίδι

Το παιχνίδι «AI Code», το οποίο υλοποιήθηκε στα πλαίσια αυτής της διπλωματικής διαθέτει τους απαραίτητους μηχανισμούς για να επιτύχει τον σκοπό του. Βέβαια, μπορούν να γίνουν βελτιώσεις και προσθήκες ώστε να γίνει καλύτερη ποιότητα του παιχνιδιού. Ένα βασικό στοιχείο το οποίο θα μπορούσε να βελτιωθεί είναι το γραφικό περιβάλλον και τα μοντέλα των χαρακτήρων. Τα γραφικά στοιχεία, τα οποία απαρτίζουν το παιχνίδι είτε δημιουργήθηκαν χρησιμοποιώντας τα εργαλεία σχεδίασης της Unity, είτε αποκτήθηκαν δωρεάν μέσω του Unity asset store, επομένως η ποιότητάς τους υστερεί σε σύγκριση με αυτά των εταιρειών. Με τη βελτίωση του γραφικού περιβάλλοντος το παιχνίδι θα γινόταν πιο ρεαλιστικό και ελκυστικό για τον παίκτη.

Η πιο σημαντική ίσως προσθήκη για το συγκεκριμένο παιχνίδι θα ήταν να έχει ο παίκτης τη δυνατότητα να γράφει και ο ίδιος κώδικα για να αλλάξει τη συμπεριφορά του Charlie, αντί να επιλέγει μόνο κάποια από τις διαθέσιμες επιλογές που του παρουσιάζονται. Αυτό, προϋποθέτει την ανάπτυξη ενός μεταγλωττιστή, ο οποίος θα επεξεργάζεται των κώδικα του παίκτη, θα του εμφανίζει τα λάθη του και θα τον θέτει σε λειτουργία. Η δημιουργία ενός μεταγλωττιστή είναι μια απαιτητική και χρονοβόρα διαδικασία, αλλά θα πρόσφερε μια πιο πρακτική προσέγγιση στον προγραμματισμό για τον μαθητή και θα προήγαγε την ανάπτυξη δεξιοτήτων, όπως τη δημιουργικότητα και την αντιμετώπιση προβλημάτων.

6.3 Συμπεράσματα

Κατά την εκπόνηση αυτής της διπλωματικής μελετήθηκε η σχετική βιβλιογραφία για τη χρήση των σοβαρών παιχνιδιών στην εκπαίδευση και τη χρήση της τεχνητής νοημοσύνης σε αυτά για τον προγραμματισμό ευφών πρακτόρων. Επιπρόσθετα, σχεδιάστηκε και υλοποιήθηκε ένα σοβαρό παιχνίδι, με στόχο την εξοικείωση του χρήστη με την πλατφόρμα ανάπτυξης παιχνιδιών Unity και με κάποιες από τις μεθόδους προγραμματισμού τεχνητής νοημοσύνης για ηpc.

Τα συμπεράσματα, τα οποία προέκυψαν από την έρευνα, ενισχύουν την άποψη ότι η χρήση των ηλεκτρονικών παιχνιδιών στην εκπαίδευση προσφέρει πολλά πλεονεκτήματα και ανταποκρίνεται στις ανάγκες των καιρών για εκσυγχρονισμό και εμπλουτισμό της εκπαιδευτικής διαδικασίας. Επίσης, προκύπτει το συμπέρασμα ότι η τεχνητή νοημοσύνη είναι ένα άρρηκτο συστατικό των ηλεκτρονικών παιχνιδιών και η γνώση της αποτελεί μια πολύ χρήσιμη δεξιότητα για όποιον θέλει να ασχοληθεί με την δημιουργία βιντεοπαιχνιδιών. Τέλος, μέσα από την ενασχόληση με την πλατφόρμα Unity αποκτήθηκαν οι βασικές δεξιότητες που απαιτούνται για την δημιουργία ενός ηλεκτρονικού παιχνιδιού, οι οποίες συνδυάζουν τομείς όπως τα γραφικά υπολογιστών και τον προγραμματισμό.

Βιβλιογραφία

- Abelson, H., Goodman, N., & Rudolph, L. N. (1974). *LOGO manual*. Massachusetts Institute of Technology Artificial Intelligence Laboratory.
- Djaouti, D., Alvarez, J., & Jessel, J. (n.d.). Classifying Serious Games. *Advances in Game-Based Learning Handbook of Research on Improving Learning and Motivation through Educational Games*, 118-136. doi:10.4018/978-1-60960-495-0.ch006
- Drigas, A., & Karyotaki, M. (2016, 04). Online and other ICT-based Assessment Tools for Problem-solving Skills. *International Journal of Emerging Technologies in Learning (iJET)*, 11(04), 56. doi:10.3991/ijet.v11i04.5339
- Eaton, E., Stone, P., Walsh, T., Wooldridge, M., Dietterich, T., Gini, M., . . . Russell, S. (2016, 01). Who speaks for AI? *AI Matters*, 2(2), 4-14. doi:10.1145/2847557.2847559
- Engel, A., & Broeck, C. V. (n.d.). Perceptron Learning – Basics. *Statistical Mechanics of Learning*, 14-32. doi:10.1017/cbo9781139164542.003
- Game Studies. (n.d.). Retrieved from <http://gamestudies.org/0802/articles/sicart>
- Groom, V., Bailenson, J. N., & Nass, C. (2009, 07). The influence of racial embodiment on racial bias in immersive virtual environments. *Social Influence*, 4(3), 231-248. doi:10.1080/15534510802643750
- Hart, P., Nilsson, N., & Raphael, B. (1968). *A formal basis for the heuristic determination of minimum cost paths*.
- Howard-Jones, P., Demetriou, S., Bogacz, R., Yoo, J. H., & Leonards, U. (2011, 02). Toward a Science of Learning Games. *Mind, Brain, and Education*, 5(1), 33-41. doi:10.1111/j.1751-228x.2011.01108.x
- Kleisarxakis, M (2017, 04). *Παιχνιδοκεντρική μάθηση*.
- LinkedIn's 2017 U.S. Emerging Jobs Report. (n.d.). Retrieved from <https://economicgraph.linkedin.com/research/LinkedIns-2017-US-Emerging-Jobs-Report>
- Plato, & Cornford, F. M. (1972). *The Republic of Plato*. Oxford University Press.

- Quinlan, J. R. (1985). *Induction of decision trees*. New South Wales Institute of Technology, School of Computing Sciences.
- Samuel, A. L. (1959, 07). Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*, 3(3), 210-229. doi:10.1147/rd.33.0210
- Sicart (Vila), M. A. (2008). Defining Game Mechanics. *Game Studies*, 8(2)
- Spires, H.A. (2008). 21st century skills and serious games: Preparing the N generation. In L.A. Annetta (Ed.), *Serious educational games*(pp. 13-23). Rotterdam, The Netherlands: Sense Publishing
- Technologies, U. (n.d.). Compare Plans. Retrieved from <https://store.unity.com/compare-plans?currency=EUR>
- Technologies, U. (n.d.). Learning the interface. Retrieved from <https://docs.unity3d.com/Manual/LearningtheInterface.html>
- Technology, media & telecom. (n.d.). Retrieved from <http://www.clairfield.com/en/sectors/technology-media-telecom/>
- Yannakakis, G. N., & Togelius, J. (2018). *Artificial intelligence and games*. Springer.