

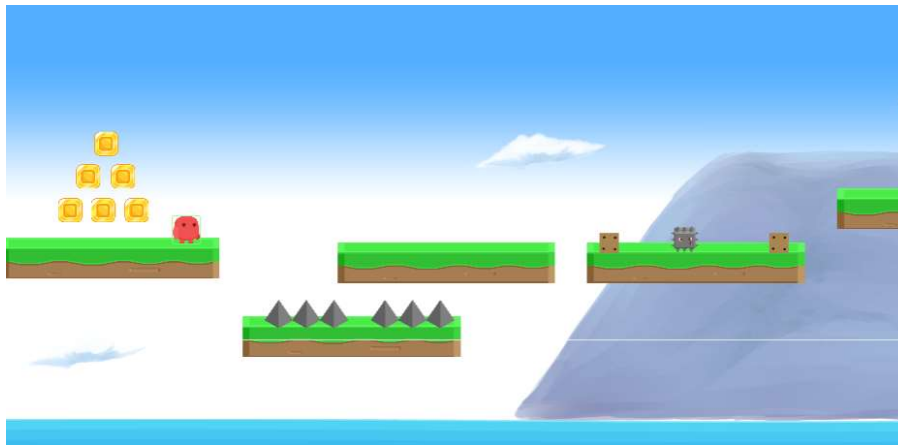


ΔΙΕΘΝΕΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΤΗΣ ΕΛΛΑΔΟΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«Σχεδίαση και ανάπτυξη δισδιάστατου παιχνιδιού
για κινητές συσκευές android»



Της φοιτήτριας

Βαρελτζή Χριστίνας

Αριθμός Μητρώου: 123832

Επιβλέπων Καθηγητής

Αδαμίδης Παναγιώτης

Θεσσαλονίκη 2022

Τίτλος Π.Ε. Σχεδίαση και ανάπτυξη δισδιάστατου παιχνιδιού για κινητές
συσκευές android

Κωδικός Π.Ε. 20210

Ονοματεπώνυμο φοιτήτριας Χριστίνα Βαρελτζή

Ονοματεπώνυμο εισηγητή Παναγιώτης Αδαμίδης

Ημερομηνία ανάληψης Π.Ε. 22/10/2020

Ημερομηνία περάτωσης Π..Ε. 29/01/2022

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

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

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

«Για τους γονείς μου»

Περίληψη

Στόχος της πτυχιακής εργασίας είναι ο σχεδιασμός και η ανάπτυξη παιχνιδιού για λειτουργικό σύστημα Android. Μέσω αυτού επιτυγχάνεται και η γνωριμία με το Unity, την μηχανή ανάπτυξης διδιάστατων και τρισδιάστατων ηλεκτρονικών παιχνιδιών συμβατών με πολλές πλατφόρμες, όπως υπολογιστές, κινητές συσκευές με λογισμικό Android ή iOS κλπ. Ως γλώσσα προγραμματισμού χρησιμοποιείται η C#. Στο πρακτικό μέρος της πτυχιακής εργασίας παρουσιάζεται η διαδικασία δημιουργίας ενός παιχνιδιού 2D από τα πρώτα στάδια της σχεδίασης των μοντέλων και του περιβάλλοντος, μέχρι τον προγραμματισμό και την δημιουργία του εκτελέσιμου αρχείου. Γίνεται χρήση αλγορίθμων για τη δημιουργία του συστήματος κίνησης, παρουσιάζεται ο τρόπος χρήσης μοντέλων με animations και η δημιουργία διαφόρων μενού. Επίσης σκοπός της εργασίας είναι η περαιτέρω εξοικείωση με την λογική του αντικειμενοστραφούς προγραμματισμού καθώς και τη διαδικασία της σύνθεσης ενός πλήρους ηλεκτρονικού παιχνιδιού υπολογιστή, η οποία συμπεριλαμβάνει την δημιουργία C# scripts και τον χειρισμό του προγράμματος Unity editor.

Design and development of a 2D game for android mobile devices

Vareltzi Christina

Abstract

The topic of the dissertation is about Game Design and Development for Android operating systems using Unity which is a development engine, about two-dimensional or three-dimensional games, compatible with many platforms such as computers, mobile devices with Android or iOS software, etc. The programming language that is used is C#. The second part of the dissertation follows the process of creating a 2D game from the first stages of designing the models and the environment, to programming and creating the executable file. Algorithms are used to create the motion system along with the animation models and various menus are presented. The goal of the project is to get further acquainted with object-oriented programming as well as the process of composing a complete computer game which includes the creation of C # scripts and the management of the Unity editor program.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή μου κύριο Αδαμίδα Παναγιώτη για την συνεργασία, την καθοδήγηση, την υπομονή του και το χρόνο που αφιέρωσε για την διεκπεραίωση της εργασίας. Ακόμα θα ήθελα να ευχαριστήσω την αδερφή μου, Δήμητρα για την κατανόηση και ψυχολογική υποστήριξη που μου πρόσφερε τους τελευταίους μήνες καθώς και όλους τους φίλους και συγγενείς που με βοήθησαν μέσα από τα θετικά και ενθαρρυντικά τους σχόλια.

Περιεχόμενα

Περίληψη	iv
Abstract	v
Ευχαριστίες	vi
Περιεχόμενα	vii
Κατάλογος Σχημάτων	ix
Κεφάλαιο 1 – Εισαγωγή.....	1
Κεφάλαιο 2 – Ηλεκτρονικά Παιχνίδια.....	3
2.1 Εισαγωγή.....	3
2.2 Η εξέλιξη των ηλεκτρονικών παιχνιδιών μέσα στο χρόνο.....	3
2.3 Είδη ηλεκτρονικών παιχνιδιών	4
2.3.1 Παιχνίδια δράσης (Action Games).....	4
2.3.2 Παιχνίδια Περιπέτειας (Adventure Games)	7
2.3.3 Παιχνίδιαδράσης – περιπέτειας (Action – Adventure)	7
2.3.4 Παιχνίδια ρόλων (Role Playing Games)	8
2.3.5 Παιχνίδια στρατηγικής (StrategyGames).....	9
2.3.6 Παιχνίδια προσομοίωσης (SimulationGames).....	9
2.3.7 Εκπαιδευτικά παιχνίδια (Educational Games)	10
2.3.8 Παιχνίδια γρίφων (Puzzle Games)	10
Κεφάλαιο 3 – Ανάπτυξη Παιχνιδιών	12
3.1 Εισαγωγή.....	12
3.2 Γραφικά Υπολογιστών.....	12
3.3 Εργαλεία Ανάπτυξης Γραφικών.....	13
3.3.1 AdobePhotoshop	13
3.3.2 AdobeAnimate.....	13
3.3.3 Autodesk 3DSMax.....	14
3.3.4 Houdini.....	14
3.3.5 Blender	15
3.4 Μηχανές ανάπτυξης παιχνιδιών.....	15
Κεφάλαιο 4ο – Unity	17
4.1 Εισαγωγή.....	17
4.2 Unity	17
4.3 Πλεονεκτήματα	17

4.4 Περιβάλλον Εργασίας Unity	19
4.4.1 UnityEditor.....	19
4.4.2 Menu	20
4.5 Σημαντικά Παράθυρα και Εργαλεία	22
Κεφάλαιο 5 - Προγραμματισμός.....	24
5.1 Γλώσσα Προγραμματισμού C#.....	24
5.2 Από τη C στηC#.....	24
5.3 .NET Framework	24
5.4 Monodevelop.....	25
5.5 C# στη Unity -MonoBehaviour.....	26
5.5.1 Αλληλεπίδραση Αντικειμένων.....	27
Κεφάλαιο 6 – Υλοποίηση	28
6.1 Assets	28
6.2 Σκηνές (Scenes)	28
6.2.1 LogInScene	28
6.2.2 Menu Scene.....	29
6.2.3 LoadLevel Scene.....	30
6.2.4 LevelScene	31
6.2.5 EndScene.....	33
6.3 Εξαγωγή Παιχνιδιού	33
Κεφάλαιο 7 – Συμπεράσματα	35
Βιβλιογραφία	36

Κατάλογος Σχημάτων

Σχήμα 2.1: First-Person Shooter View	5
Σχήμα 2.2: Third-Person Shooter View	5
Σχήμα 2.3: Fighting Game	6
Σχήμα 2.4: Platform Game	6
Σχήμα 2.5: Beat Em Up Game	7
Σχήμα 2.6: Adventure games - Hidden Objects	7
Σχήμα 2.7: Role Playing Games	8
Σχήμα 2.8: Massively Multiplayer Online Role-Playing Game	8
Σχήμα 2.9: Real Time Strategy Games	9
Σχήμα 2.10: Real-life Simulation Games	10
Σχήμα 2.11: Education Games	10
Σχήμα 2.12: Puzzle Games	11
Σχήμα 3.1: Περιβάλλον Εργασίας Adobe Photoshop	13
Σχήμα 3.2: Περιβάλλον Εργασίας Adobe Animate	13
Σχήμα 3.3: Περιβάλλον Εργασίας Autodesk 3DS Max	14
Σχήμα 3.4: Περιβάλλον Εργασίας Houdini	14
Σχήμα 3.5: Περιβάλλον Εργασίας Blender	15
Σχήμα 4.1: Logo Unity	17
Σχήμα 4.2: Free VS Pro Version Unity	18
Σχήμα 4.3: Unity Editor	20
Σχήμα 4.4: Βασικό Menu	20
Σχήμα 4.5: Transform tools Menu	21
Σχήμα 4.6: Game View Control Menu	21

Σχήμα 4.7: Player Run Animation	22
Σχήμα 4.8: Player Animator.....	22
Σχήμα6.1: LogIn Scene.....	29
Σχήμα6.2: Menu Scene Hierarchy	30
Σχήμα6.3: LoadingLevel1 Hierarchy.....	31
Σχήμα6.4: Level1 Scene Hierarchy.....	32
Σχήμα6.5: Ιδιότητες Αντικειμένου Player	32
Σχήμα6.6: End Scene	33
Σχήμα6.7: Android Build Support	34

Κεφάλαιο 1 – Εισαγωγή

Η παρούσα πτυχιακή εργασία έχει σαν στόχο την σχεδίαση και την ανάπτυξη ενός παιχνιδιού για android κινητές συσκευές. Η μηχανή που χρησιμοποιήθηκε είναι η Unity Game Engine και ως γλώσσα προγραμματισμού η C#.

Στα πλαίσια της παρούσας πτυχιακής εργασίας αναπτύχθηκε ένα παιχνίδι δύο διαστάσεων κατηγορίας platform. Το παιχνίδι αποτελείται από τρία επίπεδα με το καθένα να έχει σταδιακά αυξανόμενη δυσκολία από το προηγούμενο.

Για την δημιουργία του παιχνιδιού χρειάστηκαν εννιά σκηνές. Στην πρώτη σκηνή δίνεται η δυνατότητα στον χρήστη να κάνει log in συμπληρώνοντας κάποιο email και δημιουργώντας ένα κωδικό πρόσβασης. Τα στοιχεία αυτά αποθηκεύονται με τη βοήθεια της Google στην βάση Firebase. Μετά την ολοκλήρωση του log in εμφανίζεται η δεύτερη σκηνή που προσφέρει το κύριο μενού του παιχνιδιού. Μέσα από αυτό επιλέγοντας την επιλογή "Play" ξεκινάει η ροή του παιχνιδιού.

Ο παίκτης κινείται ελεύθερα στο χώρο του παιχνιδιού με στόχο να ξεπερνά τα εμπόδια που προσφέρει κάθε πίστα και να νικά τους εκάστοτε εχθρούς. Ακόμα το κύριο μενού δίνει την ευκαιρία στο χρήστη να κάνει έξοδο από το παιχνίδι. Οι σκηνές τέσσερα, έξι και οχτώ αποτελούν τις πίστες του παιχνιδιού ενώ η τρίτη, η πέμπτη και η έβδομη σκηνή είναι οι σκηνές μετάβασης των επιπέδων. Τέλος η ένατη και τελευταία σκηνή ανακοινώνει την λήξη του παιχνιδιού στον χρήστη και τον οδηγεί στην έξοδο από αυτό.

Σε κάθε επίπεδο υπάρχει ένα Pause Menu, όπου ο χρήστης μπορεί να χρησιμοποιήσει ώστε να βάλει το παιχνίδι σε λειτουργία παύσης ή και να οδηγηθεί στην έξοδο από αυτό όποτε το επιθυμεί. Για να γίνουν πραγματικότητα όλα αυτά χρησιμοποιήθηκαν πολλά εργαλεία της Unity. Δημιουργήθηκαν και χρησιμοποιήθηκαν διάφορα scripts με τη βοήθεια της C#. Τέλος τα αντικείμενα που χρησιμοποιήθηκαν ήταν δωρεάν και εύκολα προσβάσιμα μέσω του Asset Store της Unity.

Στο δεύτερο κεφάλαιο, παρουσιάζεται η εξέλιξη των ηλεκτρονικών παιχνιδιών μέσα στο χρόνο. Από το 1952 που γεννήθηκε το πρώτο βιντεοπαιχνίδι ως χρυσή εποχή των Arcade την δεκαετία του 1980 και φυσικά μέχρι σήμερα όπου κυκλοφορεί η έβδομη γενιά παιχνιδομηχανών. Ακόμα περιγράφονται τα πιο γνωστά είδη των ηλεκτρονικών παιχνιδιών όπως τα παιχνίδια δράσης περιπέτειας, τα παιχνίδια ρόλων και στρατηγικής.

Στο τρίτο κεφάλαιο, αναφέρονται τα είδη των γραφικών καθώς και πέντε από τις πιο γνωστές μηχανές ανάπτυξης γραφικών. Επιπλέον περιγράφονται δημοφιλείς μηχανές ανάπτυξης ηλεκτρονικών παιχνιδιών όπως η Unreal και πολλές άλλες που κυκλοφόρησαν από την Amazon, το Facebook και την Apple.

Στο κεφάλαιο τέσσερα, γίνεται μια μικρή ιστορική αναδρομή για την Unity Game Engine. Στη συνέχεια αναφέρονται η λόγοι που προτιμήθηκε για την υλοποίηση της πτυχιακής εργασίας καθώς και τα πλεονεκτήματά της. Αναλύεται εκτενώς το περιβάλλον εργασίας, περιγράφοντας τα διάφορα παράθυρα που προσφέρει καθώς και τα μενού που κάνουν εύχρηστη τη λειτουργία του λογισμικού.

Στο πέμπτο κεφάλαιο γίνεται αναφορά στην γλώσσα προγραμματισμού C#. Συγκεκριμένα αναφέρονται μερικά ιστορικά στοιχεία της καθώς και στοιχεία που αφορούν την εξέλιξη της. Επιπλέον αναφέρεται ο λόγος όπου είναι συμβατή με τη Unity και περιγράφονται διάφορες συναρτήσεις και κλάσεις που χρησιμοποιεί η Unity.

Στο έκτο κεφάλαιο περιγράφεται το παιχνίδι που αναπτύχθηκε για την πτυχιακή εργασία. Περιγράφονται οι διάφορες σκηνές που απαρτίζουν το παιχνίδι, σημαντικές λειτουργίες του παίκτη, τα assets που χρησιμοποιήθηκαν και τα menu που δημιουργήθηκαν για την ομαλή λειτουργία του παιχνιδιού. Ακόμα περιγράφεται ο τρόπος εξαγωγής του παιχνιδιού σε κινητές android συσκευές και πως μέσω του Google Play γίνεται δυνατή η διανομή και η πώληση της εφαρμογής.

Στο έβδομο κεφάλαιο γίνεται μια ανακεφαλαίωση της πτυχιακής εργασίας και αναφέρονται τα συμπεράσματα της.

Κεφάλαιο 2 – Ηλεκτρονικά Παιχνίδια

2.1 Εισαγωγή

Το 1952 ήταν το έτος που δημιουργήθηκε το πρώτο ηλεκτρονικό παιχνίδι (video game). Λόγω της περιορισμένης τεχνολογίας που διέθετει η εποχή, οι δυνατότητες του ήταν περιορισμένες και η χρήση του επιτρεπτή από λίγους. Από τότε έως και σήμερα, φυσικά, η ανάπτυξη που έχει παρουσιάσει η βιομηχανία των ηλεκτρονικών παιχνιδιών είναι πραγματικά εκθετική καθώς τα video games πλέον δεν περιορίζονται μόνο σε υπολογιστές και κονσόλες αλλά και σε κινητές συσκευές όπως smartphones και tablets [2].

Η κατασκευή και η ανάπτυξη των πρώτων ηλεκτρονικών παιχνιδιών ήταν μια ιδιαίτερα επίπονη διαδικασία καθώς οι προγραμματιστές δημιουργούσαν τα πάντα από το μηδέν. Πλέον, υπάρχουν διάφορα εργαλεία ανάπτυξης παιχνιδιών που μπορούν να χρησιμοποιηθούν από έμπειρους έως και αρχάριους προγραμματιστές. Σήμερα, κατά την ανάπτυξη των παιχνιδιών, δίνεται ιδιαίτερη προσοχή στον ήχο, τα γραφικά και τα εφέ ώστε το τελικό προϊόν να είναι εντυπωσιακό [3].

2.2 Η εξέλιξη των ηλεκτρονικών παιχνιδιών μέσα στο χρόνο

Η εμπειρία του χρήστη δεν ήταν ο κύριος στόχος κατά την ανάπτυξη και τη κατασκευή των πρώτων παιχνιδιών καθώς αυτά ήταν ελάχιστα ψυχαγωγικά. Για την λειτουργία τους, τα παιχνίδια απαιτούσαν ισχυρούς υπολογιστές οι οποίοι ταυτόχρονα ήταν και ογκώδεις. Η τρίλιζα (OXO, 1952), το tennis for two (1958) και το spacewar (1961) ήταν από τα πρώτα παιχνίδια που δημιουργήθηκαν ποτέ. Για το πρώτο χρησιμοποιήθηκε μια συσκευή ψηφιακής προβολής ως μέσο αναπαράστασης ενώ για το δεύτερο ένας παλμογράφος [1].

Το 1978 η εταιρία Taito δίνοντας στο κοινό το παιχνίδι Space Invader, πυροδότησε την λεγόμενη χρυσή εποχή των arcades. Η μεγάλη επιτυχία που σημείωσε το παιχνίδι αυτό έγινε η αιτία έμπνευσης και για άλλους κατασκευαστές ώστε να μουν στην αγορά για να δημιουργήσουν τα δικά τους παιχνίδια. Έτσι, λοιπόν το 1979 κάνει την εμφάνιση του ένα από τα πιο δημοφιλή παιχνίδια όλων των εποχών το Pac-Man της Namco. Τα arcade παιχνίδια ήταν πλέον τοποθετημένα σε πολυσύχναστους δημόσιους χώρους όπως εμπορικά κέντρα, εστιατόρια, χώρους ψυχαγωγίας, ξενοδοχεία και μπαρ. Κάθε κονσόλα ήταν αυτόνομη, λειτουργούσε με κέρματα και πρόσφερε ένα έγχρωμο παιχνίδι ψυχαγωγίας. Ο τεράστιος όγκος τους αλλά και το μεγάλο κόστος τους τα καθιστούσε μη επιτρεπτά για οικιακή χρήση. Τα παιχνίδια arcade άρχισαν να χάνουν την δημοτικότητά τους σταδιακά μέσα στην δεκαετία του 1990 [1][2].

Στην συνέχεια, έκαναν την εμφάνισή τους στην αγορά οι μηχανές παιχνιδιών για οικιακή χρήση πρώτης γενιάς. Όπως ήταν φυσικό στη βιομηχανία τότε, δεν υπήρχαν ιδιαίτερες πρωτοποριακές ιδέες όσον αφορά τη σχεδίαση των παιχνιδιών και αυτό είχε σαν αποτέλεσμα να υπάρχουν πολλά παρόμοια παιχνίδια. Ταυτόχρονα μέσω των ηλεκτρονικών υπολογιστών που εμφανίστηκαν στη αγορά, δίνεται η δυνατότητα σε μεμονωμένα άτομα να προγραμματίσουν τα δικά τους παιχνίδια. Τα παιχνίδια αυτά ήταν απλά και η παραγωγή τους εύκολη και γρήγορη. Λόγω της εποχής υπήρχαν περιορισμοί στη μνήμη και στα γραφικά που εμπόδιζαν την ανάπτυξη πολύπλοκων παιχνιδιών. Εταιρείες ανέθεταν όλο και πιο συχνά σε μικρές ομάδες προγραμματιστών την ανάπτυξη παιχνιδιών. Συνεπώς οι υπολογιστές ήταν στο επίκεντρο των πωλήσεων. Λίγο αργότερα οι κονσόλες ξανά εμφανίζονται δυναμικά στην αγορά, όταν η Nintendo προσφέρει στο κοινό μια 8-bit συσκευή (NES) συνοδευόμενη από το παιχνίδι Super Mario Bros [1].

Αυτό που χαρακτηρίζει κυρίως τη δεκαετία του 1990 είναι οι νέες ιδέες που κατέκλυσαν τον τομέα των βιντεοπαιχνιδιών. Νέες κατηγορίες εμφανίστηκαν στην αγορά όπως το FPS (First- Person Shooter), παιχνίδια στρατηγικής και διαδικτύου. Σημαντική φυσικά θεωρείται και η μετάβαση των γραφικών από 2D σε 3D. Επιπλέον κυκλοφόρησαν φορητές κονσόλες παιχνιδιών (gameboy) που ήταν ιδιαίτερα αγαπητές στις νεαρές ηλικίες, ανοίγοντας έτσι το δρόμο για τις κονσόλες πέμπτης γενιάς (32 64 bit). Τέλος οι κασέτες έπαψαν να χρησιμοποιούνται σταδιακά και τη θέση τους πήραν τα cd (compact discs) [1].

Το 1998 η εταιρεία Nokia πρωτοπόρησε βάζοντας στα κινητά της τηλέφωνα το αγαπημένο σε όλους παιχνίδι φιδάκι (snake). Με αυτήν την κίνηση μετατρέπει έτσι τα κινητά σε μικρές πλατφόρμες ηλεκτρονικών παιχνιδιών. Αυτό φυσικά διαδόθηκε πολύ γρήγορα και σε άλλες εταιρείες κινητής τηλεφωνίας που πρόσθεταν παιχνίδια στα μοντέλα τους. Κατά το τέλος της δεκαετίας 1990 τα arcade παιχνίδια που είχαν προσφέρει αμέτρητες αναμνήσεις σε μικρούς και μεγάλους, φτάνουν δυστυχώς στο τέλος τους. Αντιθέτως οι επεξεργαστές και τα γραφικά των υπολογιστών ολοένα και αυξάνουν τις δυνατότητες τους. Η συνεχής εξέλιξη των προηγούμενων σε συνδυασμό με το διαδίκτυο, που εισβάλλει πλέον στην καθημερινότητα των ανθρώπων, καθιστά τους υπολογιστές αρκετά δημοφιλείς. Στη δεκαετία 2000 νέες εξελιγμένες κονσόλες έκαναν την εμφάνισή τους και ο ανταγωνισμός ανάμεσα τους συνεχώς αυξάνεται. Διάφορα νέα παιχνίδια κυκλοφόρησαν άλλα συμβατά για φορητές παιχνιδομηχανές και άλλα για ηλεκτρονικούς υπολογιστές. Η βιομηχανία των ηλεκτρονικών παιχνιδιών σταδιακά μεγαλώνει και τα κέρδη της αυξάνονται. Εταιρείες άρχισαν να εκμεταλλεύονται τις ευρυζωνικές συνδέσεις προς όφελος τους, δημιουργώντας παιχνίδια που πολλοί χρήστες ταυτόχρονα θα μπορούσαν να συνδεθούν σε αυτά και να παίξουν μεταξύ τους (MMO) [1].

Τέλος, σήμερα βρίσκεται σε κυκλοφορία η έβδομη γενιά φορητών παιχνιδομηχανών (playstation 5, Nintendo wii), με το κύριο χαρακτηριστικό τους να είναι η πρόσβαση στο διαδίκτυο αλλά και φυσικά τα εντυπωσιακά γραφικά τους. Η τεχνολογία έχει πραγματοποιήσει τεράστια άλματα που πλέον τα κινητά τηλέφωνα χρησιμοποιούνται όχι μόνο σαν μέσα επικοινωνίας αλλά και ως μέσα ψυχαγωγίας. Οι δυνατότητες τους φτάνουν αυτές ενός μικρού υπολογιστή. Η ασύρματη πρόσβαση στο internet, οι επεξεργαστές, οι κάρτες γραφικών αλλά και η μεγάλη χωρητικότητα στα μέσα αποθήκευσης τα καθιστά απολύτως ικανά να φιλοξενήσουν διάφορα παιχνίδια [1].

2.3 Είδη ηλεκτρονικών παιχνιδιών

Σήμερα υπάρχουν διάφορα είδη ηλεκτρονικών παιχνιδιών. Ο κύριος τρόπος με τον οποίον διαχωρίζονται είναι η εμπειρία παιχνιδιού που προσφέρουν στον χρήστη. Φυσικά, κάθε κατηγορία με τη σειρά της, χωρίζεται και αυτή σε υποκατηγορίες. Ο αριθμός των ατόμων που συμμετέχουν ταυτόχρονα σε ένα παιχνίδι, το θέμα του και η χρήση δύο ή τριών διαστάσεων είναι μερικά από τα κριτήρια διαχωρισμού τους. Είναι σημαντικό να σημειωθεί ότι ένα παιχνίδι μπορεί να ανήκει ταυτόχρονα σε δύο κατηγορίες.

2.3.1 Παιχνίδια δράσης (Action Games)

Τα παιχνίδια δράσης είναι μια κατηγορία βιντεοπαιχνιδιών που δίνει έμφαση στις σωματικές δοκιμασίες, όπως τον συγχρονισμό χεριών-ματιών του χρήστη καθώς και την υψηλή ταχύτητα αντίδρασης του. Μερικές από τις προκλήσεις που θα συναντήσει ο παίκτης μπορεί να είναι ακόμα η λύση γρίφων, η συλλογή διαφόρων αντικειμένων προκειμένου να ολοκληρωθεί το παιχνίδι κλπ, χωρίς αυτές να αποτελούν το κυρίαρχο είδος προκλήσεων. Συνήθως το χρονικό περιθώριο που έχει κάθε πίστα δεν είναι αρκετό ώστε να υλοποιηθεί κάποιο σχέδιο επίθεσης, που θα βοηθήσει στην ολοκλήρωση του παιχνιδιού. Μερικές από τις υποκατηγορίες αυτού του είδους είναι:

- Σκόπευσης (Shooter) (Σχήμα 2.1, 2.2): Στα παιχνίδια αυτά ο παίκτης είναι πάντα οπλισμένος και κουβαλάει διάφορα πυρομαχικά. Ο παίκτης πρέπει να διασχίσει την κάθε πίστα πυροβολώντας τους εχθρούς που συναντάει ώστε να φτάσει στο τέλος του παιχνιδιού. Στις πίστες υπάρχουν διασκορπισμένα αντικείμενα που μπορεί να συλλέξει, όπως αλεξίσφαιρα, κιάλια νυχτερινής όρασης, ή κάποια είδος αναβάθμιση που αφορά το όπλο του. Πρόκειται για ένα είδος με πολλές υποκατηγορίες με κυριότερες τα παιχνίδια στόχευσης πρώτου και τρίτου προσώπου (First Person Shooter, Third-Person Shooter), τα παιχνίδια τακτικής (Tactical Shooter) και τα Shoot'em Up [6].



Σχήμα 2.1 First-Person Shooter



Σχήμα 2.2 Third-Person Shooter

- Μάχης (Fighting) (Σχήμα 2.3): Δύο παίκτες, ο ένας εκ των οποίων χειραγωγείται από τον υπολογιστή, μάχονται μεταξύ τους. Ο παίκτης, που χειρίζεται ο χρήστης, επιτίθεται στον αντίπαλο βάσει των κινήσεων που εισάγει στο χειριστήριο. Για να είναι η κίνηση αποτελεσματική και να προκαλέσει τη μέγιστη δυνατή ζημιά στον αντίπαλο πρέπει ο χρήστης να ακολουθήσει κάποιο συγκεκριμένο συνδυασμό πλήκτρων με ακρίβεια και ταχύτητα [6].



Σχήμα 2.3 Fighting

- Πλατφόρμας (Platform) (Σχήμα 2.4): Συνήθως πρόκειται για δισδιάστατα παιχνίδια. Ο παίκτης κινείται πλαγίως αποφεύγοντας τα εμπόδια και τους εκάστοτε εχθρούς της κάθε πίστας. Επιπλέον έχει την ικανότητα να ανεβαίνει σκάλες και να υπερπηδά. Είναι από τα παιχνίδια που έχουν εξελιχθεί επιτυχημένα σε τρισδιάστατα μέσα στα χρόνια. Ακόμα και σήμερα εξακολουθούν να κυκλοφορούν παιχνίδια πλατφόρμας παρόλο που έχουν χάσει τη δημοτικότητα τους [6].



Σχήμα 2.4 Platform

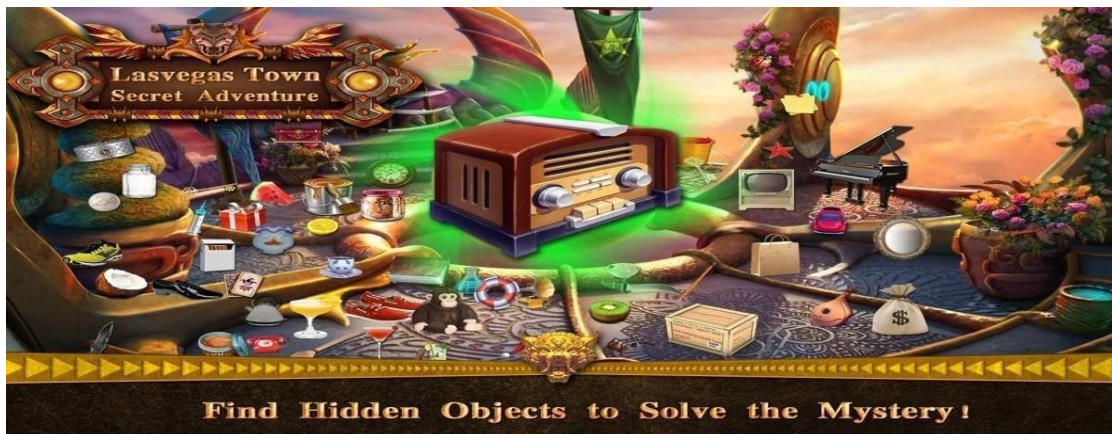
- Beat'em Up (Σχήμα 2.5): Ο παίκτης έρχεται αντιμέτωπος με έναν ατελείωτο στρατό από αντιπάλους χαμηλότερης δυναμικής. Ο παίκτης ελέγχεται από τον χρήστη και μπορεί να μετακινηθεί κατά μήκος και πλάτος της πίστας. Κύριος στόχος του παιχνιδιού είναι ο παίκτης να φτάσει στο τέλος του επιπέδου και να νικήσει τον τελικό εχθρό ώστε να περάσει στην επόμενη πίστα. Η δυσκολία στα επίπεδα αυξάνεται σταδιακά. Το παιχνίδι ολοκληρώνεται είτε αν ο παίκτης νικήσει είτε αν χάσει (game over). Ανεξάρτητα από το λόγο λήξης του παιχνιδιού, αποθηκεύεται πάντα το τελικό σκορ του χρήστη που λειτουργεί σαν πρόκληση για το επόμενο [6].



Σχήμα 2.5 Beat Em Up

2.3.2 Παιχνίδια Περιπέτειας (Adventure Games)

Τα παιχνίδια περιπέτειας (Σχήμα 2.6) δίνουν ιδιαίτερη έμφαση στην αφήγηση και την επίλυση γρίφων ενώ αντίθετα η δράση που προσφέρουν είναι ελάχιστη. Οι έξυπνοι γρίφοι και η εντυπωσιακή αφήγηση είναι απαραίτητα στοιχεία ώστε να προχωρήσει ομαλά η σεναριακή πλοκή του παιχνιδιού. Ο παίκτης είναι πάντα ο πρωταγωνιστής αυτών των ιστοριών. Οι γρίφοι είναι διαφόρων ειδών και συνήθως σχετίζονται με την αλληλεπίδραση του παίκτη με τους χαρακτήρες και αντικείμενα του περιβάλλοντος [5]. Όπως αναφέρθηκε και προηγουμένως τα παιχνίδια περιπέτειας περιέχουν πληθώρα γρίφων όπως αποκωδικοποίηση μηνυμάτων, εύρεση και χρήση στοιχείων που οδηγούν σε άνοιγμα κλειδαριών και θυρών ή εύρεση και εξερεύνηση νέων τοποθεσιών. Η επίλυση ενός γρίφου μπορεί να οδηγήσει σε νέες περιοχές στον κόσμο του παιχνιδιού που με τη σειρά τους να αποκαλύψουν νέα στοιχεία. Παράδειγμα ενός γρίφου είναι η εύρεση διαφόρων αντικειμένων μέσα σε ένα χώρο, τα οποία είναι κρυμμένα. Στο τέλος της δοκιμασίας αποκαλύπτεται το στοιχείο που βοηθάει στην εξέλιξη της ιστορίας.



Σχήμα 2.6 Adventure games - hidden objects

2.3.3 Παιχνίδια δράσης – περιπέτειας (Action – Adventure)

Η συγκεκριμένη κατηγορία συνδυάζει αρμονικά στοιχεία δράσης και περιπέτειας. Πρόκειται για παιχνίδια όπου ο παίκτης εξερευνεί το περιβάλλον της πίστας και επιλύει σχετικά εύκολους γρίφους (στοιχείο περιπέτειας). Παράλληλα, το παιχνίδι προσφέρει σκηνές δράσης – μάχης καθώς ο παίκτης τίθεται να πολεμήσει τις συνεχείς απειλές που εμφανίζονται (στοιχείο δράσης). Το είδος αυτό τα τελευταία χρόνια έχει γνωρίσει τεράστια ανάπτυξη και έχει ιδιαίτερη απήχηση στο κοινό. Αποτελεί επίσης την κατηγορία που κατατάσσεται κάποιο παιχνίδι το οποίο δεν εμπίπτει εύκολα σε κάποια άλλη γνωστή κατηγορία [4].

2.3.4 Παιχνίδια ρόλων (Role Playing Games)

Τα παιχνίδια ρόλων (Σχήμα 2.7), επίσης γνωστά ως RPG (Role Playing Games), είναι ιδιαίτερα συναρπαστικά καθώς βασίζονται σε επιτραπέζια παιχνίδια ρόλων τα γνωστά και ως pen and paper role playing games. Ο παίκτης επιλέγει έναν χαρακτήρα του παιχνιδιού ο οποίος έχει κάποιες συγκεκριμένες ιδιότητες όπως μαγικές ικανότητες ή σωματική δύναμη. Καθώς διαδραματίζεται το παιχνίδι, ο παίκτης πρέπει να φέρει εις πέρας διάφορες αποστολές νικώντας τους εκάστοτε αντιπάλους. Μέσω αυτής της διαδικασίας συγκεντρώνονται πόντοι εμπειρίας που αποσκοπούν στην αύξηση του επιπέδου του χαρακτήρα. Ο κόσμος που εξερευνούν αυτά τα παιχνίδια είναι φανταστικός και είναι ένα από τα κύρια χαρακτηριστικά που επιβίωσαν κατά τη μετάβασή τους από το χαρτί στις διάφορες κονσόλες. Το κάθε παιχνίδι αυτής της κατηγορίας διηγείται μια ιστορία στο δικό του πλήρες αναπτυγμένο κόσμο με τους δικούς του ορισμένους κανόνες. Το είδος αυτό χωρίζεται σε κατηγορίες βάση τη θεματολογία, την προοπτική απεικόνισης, τον αριθμό των παικτών και τη σύνδεση στο διαδίκτυο. Τα παιχνίδια ρόλων πολλών χρηστών (Massively Multiplayer Online Role Playing Game - MMORPG, Σχήμα 2.8) έχουν ιδιαίτερη απήχηση στο κοινό καθώς επιτρέπουν τη συμμετοχή εκατοντάδων χρηστών ταυτόχρονα, οι οποίοι είτε συνεργάζονται είτε ανταγωνίζονται ο ένας τον άλλον σε πραγματικό χρόνο [4].



Σχήμα 2.7 Role PlayingGames



Σχήμα 2.8 Massively Multiplayer Online Role-Playing Game

2.3.5 Παιχνίδια στρατηγικής (Strategy Games)

Πρόκειται για παιχνίδια όπου ο παίκτης διαχειρίζεται την διοίκηση ενός στρατού. Στα παιχνίδια αυτά, για να επιτευχθεί η νίκη, απαιτείται προσεκτική σκέψη και σχεδιασμός. Ο παίκτης έχει μια συνολική άποψη του κόσμου που διαδραματίζεται το παιχνίδι δίνοντας ένα κύριο πλεονέκτημα ως προς την τοποθέτηση των μονάδων του. Βασικές αρμοδιότητές του είναι η οργάνωση, ανάπτυξη, εκπαίδευση και καθοδήγηση του στρατού κατά τη διάρκεια της μάχης. Ταυτόχρονα η σωστή διαχείριση των πόρων του παιχνιδιού δίνει τη δυνατότητα στον παίκτη να αποκτήσει επιπλέον στρατό ή να αναβαθμίσει τον ήδη υπάρχον. Είναι σημαντικό να σημειωθεί ότι όλοι οι συμμετέχοντες έχουν τον ίδιο στόχο και πρόσβαση στους ίδιους πόρους. Συνήθως τα παιχνίδια αυτά χωρίζονται βάση τον χρόνο που έχει κάθε παίκτης για να ολοκληρώσει τις ενέργειες του, κάνοντας τα παιχνίδια στρατηγικής πραγματικού χρόνου (real time strategy, Σχήμα 2.9) τα πιο δημοφιλή μέσα σε αυτή την κατηγορία. Η δράση είναι συνεχής και οι παίκτες πρέπει να παίρνουν συνεχώς αποφάσεις και να προσαρμόζουν τις ενέργειές τους. Ανεξάρτητα της παραπάνω κατηγοριοποίησης, τα παιχνίδια αυτά παίζονται είτε στο διαδίκτυο, είτε χωρίς να απαιτείται σύνδεση σε αυτό. Άλλη κατηγοριοποίηση είναι με βάση την προοπτική απεικόνισης [4].



Σχήμα 2.9 Real Time Strategy Games

2.3.6 Παιχνίδια προσομοίωσης (Simulation Games)

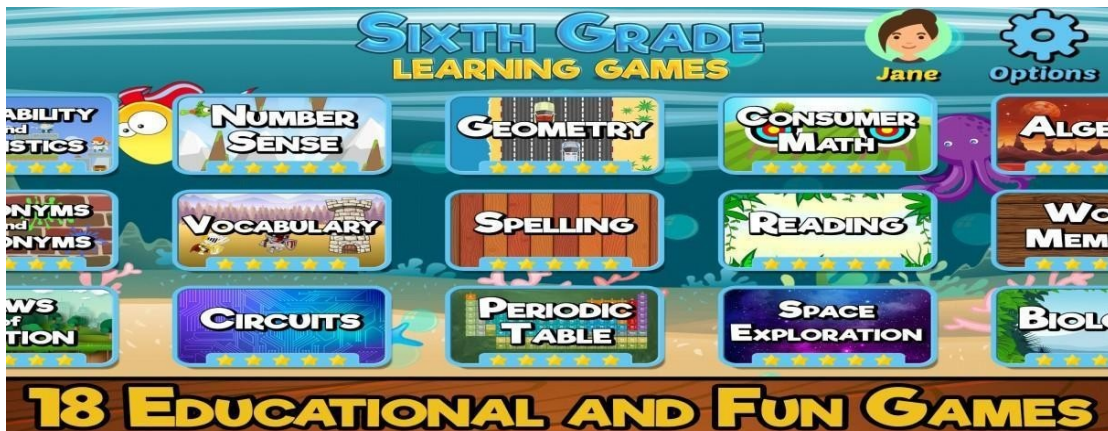
Τα παιχνίδια αυτά στοχεύουν στην προσομοίωση δραστηριοτήτων. Ένα διαδομένο παράδειγμα παιχνιδιού προσομοίωσης είναι η ανάπτυξη και διαχείριση μιας πόλης, λαμβάνοντας πάντα υπόψη τους κανόνες του πραγματικού κόσμου (Σχήμα 2.10). Δίνεται τεράστια σημασία στην ακρίβεια, το ρεαλισμό και την αυθεντικότητα με αποτέλεσμα αυτά να αποτελούν χαρακτηριστικά στοιχεία της εμπειρίας που προσφέρει το παιχνίδι. Η επιτυχία στο παιχνίδι καθορίζεται από τη σωστή διαχείριση των πόρων, που επιτρέπει την ανάπτυξη της πόλης και κατά επέκταση την ευημερία των πολιτών της. Τα παιχνίδια αγώνων ταχύτητας (racing games) και τα παιχνίδια άθλησης (sport games) μπορούν να θεωρηθούν επίσης παιχνίδια προσομοίωσης [4].



Σχήμα 2.10 Real- life Simulation Game

2.3.7 Εκπαιδευτικά παιχνίδια (Educational Games)

Είναι παιχνίδια με εκπαιδευτικό χαρακτήρα που έχουν σαν στόχο να βοηθήσουν τον χρήστη να αποκτήσει νέες γνώσεις και να διεκρινεί ή να αξιολογήσει τις ήδη υπάρχουσες. (Σχήμα 2.11). Παράλληλα αναπτύσσουν και εξασκούν τις δεξιότητες των παικτών με έναν ευχάριστο, διασκεδαστικό και παραγωγικό τρόπο. Υπάρχει μεγάλη ποικιλία σε αυτό το είδος παιχνιδιών τα οποία διαφέρουν σε θέμα και δυσκολία. Χρησιμοποιούνται ευρέως από εκπαιδευτικούς φορείς αλλά και γονείς ώστε μέσα από την διαδραστικότητα να διδάξουν επίτευξη στόχων, κανόνες, προσαρμοστικότητα καθώς και επίλυση προβλημάτων. Απευθύνονται σε όλους ανεξάρτητα από την ηλικία ή το μορφωτικό επίπεδο του παίκτη [5].



Σχήμα 2.11 Educational Games

2.3.8 Παιχνίδια γρίφων (Puzzle Games)

Πρόκειται για παιχνίδια όπου στόχος του παίκτη είναι να επιλύσει τους γρίφους που προσφέρει το κάθε παιχνίδι (Σχήμα 2.12). Η ποικιλία των γρίφων είναι τεράστια καθώς αποτελούνται από λέξεις, σχέδια, εικόνες, αριθμούς ή και το συνδυασμό αυτών. Οι γρίφοι είναι δημιουργημένοι έτσι ώστε να δοκιμάσουν πολλές δεξιότητες του παίκτη όπως τη μνήμη, τη λογική, τη παρατηρητικότητα και τη ταχύτητα των αντιδράσεων του. Παράλληλα δοκιμάζει ικανότητες όπως αναγνώριση προτύπων, επίλυση ακολουθιών, αντιστοίχιση σχημάτων, χωρική αναγνώριση και ολοκλήρωση λέξεων [5].



Σχήμα 2.12 Puzzle Games

Κεφάλαιο 3 – Ανάπτυξη Παιχνιδιών

3.1 Εισαγωγή

Σε αυτό το κεφάλαιο περιγράφονται οι διάφορες κατηγορίες με βάση τις οποίες μπορούν να χωριστούν τα γραφικά των υπολογιστών. Κυρίως διαχωρίζονται σε γραφικά δύο ή τριών διαστάσεων αλλά επίσης μπορούν να χωριστούν με βάση την ανάλυση και την απόδοση τους. Επιπλέον σε αυτό το κεφάλαιο αναφέρονται μερικές από τις πιο γνωστές μηχανές ανάπτυξης γραφικών όπως το Blender, το PhotoShop κ.α. Στη συνέχεια περιγράφονται μηχανές και πλατφόρμες ανάπτυξης παιχνιδιών όπως η Unity Engine η οποία χρησιμοποιήθηκε για την υλοποίηση της παρούσας πτυχιακής.

3.2 Γραφικά Υπολογιστών

Ο επιστημονικός κλάδος της πληροφορικής που ασχολείται με την αλγοριθμική σύνθεση εικόνων σε ηλεκτρονικούς υπολογιστές ονομάζεται γραφικά υπολογιστών.

Πρόκειται για έναν τεράστιο και πρόσφατα αναπτυγμένο τομέα της επιστήμης των υπολογιστών. Τα γραφικά υπολογιστών είναι υπεύθυνα για την αποτελεσματική και ουσιαστική προβολή δεδομένων τέχνης και εικόνας στον χρήστη. Χρησιμοποιείται επίσης στην επεξεργασία δεδομένων εικόνας που λαμβάνονται από τον φυσικό κόσμο, όπως περιεχόμενο φωτογραφιών και βίντεο. Η ανάπτυξη γραφικών υπολογιστών είχε σημαντικό αντίκτυπο σε πολλούς τύπους μέσων και γενικά έχει φέρει επανάσταση σε κινούμενα σχέδια, ταινίες, διαφημίσεις, βιντεοπαιχνίδια. Ένας υπολογιστής είναι ικανός να απεικονίζει εικόνες δύο ή τριών διαστάσεων.

Τα διςδιάστατα γραφικά χωρίζονται σε δύο κατηγορίες και είναι αυτά που απεικονίζονται σε δύο διαστάσεις στην οθόνη μιας ψηφιακής συσκευής. Η πρώτη κατηγορία είναι τα διανυσματικά γραφικά, στην οποία κατατάσσονται αυτά που χρησιμοποιούνται για τη δημιουργία εικόνων και ψευδο τρισδιάστατων σχημάτων. Τα γραφικά ψηφίδων είναι η δεύτερη κατηγορία και σε αυτήν ανήκουν εκείνα που δημιουργούνται από τη ψηφιοποίηση υπαρκτών αντικειμένων. Η βασική διαφορά ανάμεσα στις δύο κατηγορίες είναι η ανάλυση τους. Τα γραφικά ψηφίδων είναι σημαντικό να αποτελούνται από όσο το δυνατόν περισσότερες και μικρότερες ψηφίδες ώστε να απεικονίζουν ένα πιο ακριβές τελικό αποτέλεσμα. Αντίθετα τα διανυσματικά γραφικά είναι ανεξάρτητα ανάλυσης και έτσι οι εικόνες που απεικονίζονται με τη βοήθεια τους μπορούν να μεγεθυνθούν/σμικρυνθούν χωρίς να αλλοιώνεται η ποιότητα τους [16].

Τα τρισδιάστατα γραφικά υπολογιστών περιγράφουν γραφικά τριών διαστάσεων στην οθόνη μιας ψηφιακής συσκευής. Επειδή, η οθόνη του υπολογιστή είναι δύο διαστάσεων, τα τρισδιάστατα μοντέλα υφίστανται ειδική επεξεργασία με σκοπό την μετατροπή τους σε διςδιάστατες εικόνες. Κάθε τρισδιάστατο αντικείμενο αποτελείται από χιλιάδες μικρά πολύγωνα, τα οποία αλλάζουν τις γωνίες τους για να δημιουργήσουν κίνηση. Μέσω σύνθετων μαθηματικών συναρτήσεων που υπολογίζονται από 3D επιταχυντές, καθορίζεται το χρώμα και η ορατότητα ενός αντικειμένου [17].

Τα γραφικά επιπλέον μπορούμε να τα διακρίνουμε σε στατικά γραφικά υπολογιστών και σε γραφικά υπολογιστών πραγματικού χρόνου. Αυτός ο διαχωρισμός γίνεται με βάση τον χρόνο στον οποίο γίνεται η απόδοσή τους. Η πρώτη κατηγορία αφορά τα αντικείμενα γραφικών που δημιουργούνται αρχικά και έπειτα αποθηκεύονται και αναπαράγονται σε αρχείο μορφής βίντεο χωρίς να είναι αλληλεπιδραστικά. Αντίθετα στη δεύτερη ανήκουν αυτά που αποδίδονται σε πραγματικό χρόνο κάθε φορά που εκτελείται ένα πρόγραμμα και είναι αλληλεπιδραστικά. Για τη δημιουργία των δεύτερων απαιτείται μηχανή απόδοσης γραφικών πραγματικού χρόνου [17].

3.3 Εργαλεία Ανάπτυξης Γραφικών

3.3.1 Adobe Photoshop

Το Adobe Photoshop (Σχήμα 3.1) είναι το πιο διαδεδομένο πρόγραμμα επεξεργασίας ψηφιακών εικόνων και γραφικών για τροποποίηση, βελτίωση ή ακόμα και δημιουργία εικόνων από το μηδέν. Αναπτύχθηκε και κυκλοφόρησε από την Adobe Systems. Πρόκειται για ένα εύχρηστο και ευέλικτο πρόγραμμα, το οποίο διαθέτει ένα πάρα πολύ φιλικό περιβάλλον για χρήστη καθώς και το πληρέστερο σετ από εργαλεία επεξεργασίας φωτογραφίας. Η Extended έκδοση που κυκλοφορεί εξασφαλίζει τη δημιουργία και τη επεξεργασία ποσοτικών δεδομένων εικόνας και τρισδιάστατων αντικειμένων, παρέχοντας προηγμένες δυνατότητες ανάλυσης εικόνας. Η πιο πρόσφατη έκδοση του Adobe Photoshop είναι η Adobe Photoshop CS6, που κυκλοφόρησε τον Μάιο του 2012 [20].



Σχήμα 3.1 Περιβάλλον εργασίας Adobe Photoshop

3.3.2 Adobe Animate

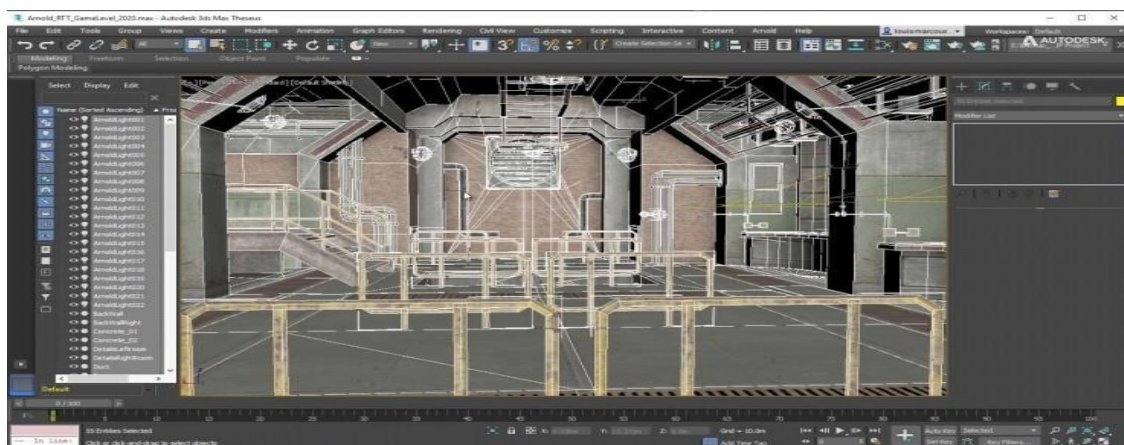
Μια ακόμα εφαρμογή της οικογένειας Adobe Systems, το Adobe Animate (Σχήμα 3.2) χρησιμοποιεί διανυσματικά γραφικά και προσφέρει υποστήριξη για ενσωμάτωση εικόνας, ήχου και κειμένου. Διαθέτει ισχυρά εργαλεία που επιτρέπουν την δημιουργία περιβάλλοντος παιχνιδιού, τον σχεδιασμό οθονών εκκίνησης κ.α. Ένα άλλο χαρακτηριστικό του είναι οι βούρτσες οι οποίες ενθαρρύνουν την δημιουργία χαρακτήρων που ζωντανεύουν εντυπωσιακά μπροστά στα μάτια των χρηστών [13].



Σχήμα 3.2 Περιβάλλον εργασίας Adobe Animate

3.3.3 Autodesk 3DSMax

Μια δημοφιλής επιλογή ανάμεσα στους προγραμματιστές παιχνιδιών είναι το Autodesk 3ds Max (Σχήμα 3.3). Πρόκειται για ένα πρόγραμμα δημιουργίας γραφικών για υπολογιστές με βελτιωμένες δυνατότητες μοντελοποίησης και μια ευέλικτη αρχιτεκτονική προσθηκών. Επίσης, επιτρέπει την ανάπτυξη παιχνιδιών κινουμένων σχεδίων 3D επαγγελματικής ποιότητας χάρη στον ενσωματωμένο επεξεργαστή χρωμάτων, τα εργαλεία επεξεργασίας χαρακτήρων κινουμένων σχεδίων και του βασικού καρέ. Τα εντυπωσιακά εφέ και διακοσμητικά στοιχεία καθώς και τα διαδικαστικά εργαλεία κίνησης δημιουργούν πιο ρεαλιστικούς χαρακτήρες και περιβάλλοντα παιχνιδιών [19].



Σχήμα 3.3 Περιβάλλον εργασίας Autodesk 3ds Max

3.3.4 Houdini

Το Houdini (Σχήμα 3.4) είναι ένα πρόγραμμα εμπλουτισμένο με χαρακτηριστικά για μοντελοποίηση και δημιουργία τρισδιάστατων χαρακτήρων κινουμένων σχεδίων. Χρησιμοποιείται για γραφικά υψηλής ποιότητας και έντονες, αλλά προσβάσιμες εμπειρίες 3D. Αυτός ο δημιουργός κινουμένων σχεδίων έρχεται ως λύση για τους προγραμματιστές παιχνιδιών που απαιτείται να επεξεργάζονται τεράστιες ποσότητες περιεχομένου χωρίς να αλλοιώνονται τα οπτικά εφέ. Χρησιμοποιώντας κινούμενα γραφικά ακόμη και εργαλεία εικονικής πραγματικότητας, το πρόγραμμα επιτρέπει τη δημιουργία συναρπαστικών κινουμένων σχεδίων που παρέχουν στο χρήστη διαδραστικό παιχνίδι και μια μοναδική εμπειρία παιχνιδιού [21].



Σχήμα 3.4 Περιβάλλον εργασίας Houdini

3.3.5 Blender

Το Blender (Σχήμα 3.5) είναι μια εφαρμογή που προσφέρει ποιοτικά εφέ με χαρακτηριστικά όπως μοντελοποίηση, προσομοίωση, rigging και rendering. Διαθέτει βούρτσες κάλυψης και υφής για υποστήριξη τρισδιάστατης ζωγραφικής, CPU, GPU και VR. Αυτό το λογισμικό κινούμενων σχεδίων είναι επίσης πλήρως προσαρμόσιμο χάρη στην ευέλικτη ελεγχόμενη διεπαφή Python [14].



Σχήμα 3.5 Περιβάλλον εργασίας Blender

3.4 Μηχανές ανάπτυξης παιχνιδιών

Μια μηχανή παιχνιδιού είναι ένα σύστημα λογισμικού σχεδιασμένο για τη δημιουργία και την ανάπτυξη βιντεοπαιχνιδιών. Υπάρχουν πολλές μηχανές παιχνιδιών οι οποίες είναι σχεδιασμένες ώστε να λειτουργούν σε διάφορες κονσόλες και διάφορα λειτουργικά συστήματα υπολογιστών. Η κεντρική λειτουργικότητα που παρέχεται από μια μηχανή παιχνιδιού αποτελείται από μια μηχανή φωτοαπόδοσης (renderer) για δισδιάστατα ή τρισδιάστατα γραφικά, μια μηχανή φυσικής ή εντοπισμού συγκρούσεων (collision detection, καθώς και collision response), ήχο, scripting, animation, τεχνητή νοημοσύνη, δικτύωση, streaming, διαχείριση μνήμης, νήματα (threading), υποστήριξη εντοπισμού θέσης (localization), και ένα γράφο σκηνής (scene graph). Οι μηχανές παιχνιδιών συνήθως επιτρέπουν το ίδιο παιχνίδι να τρέχει σε διάφορες πλατφόρμες συμπεριλαμβανομένων των κονσόλων και των προσωπικών υπολογιστών με μικρές αλλαγές στον πηγαίο κώδικα του παιχνιδιού [40].

Πριν από τις μηχανές παιχνιδιών, τα παιχνίδια τυπικά γραφόταν ως μοναδικές οντότητες. Έτσι, ένα παιχνίδι έπρεπε να σχεδιαστεί από το μηδέν για να κάνει τη βέλτιστη χρήση του υλικού εμφάνισης. Άλλες πλατφόρμες είχαν περισσότερα περιθώρια αλλά το μεγαλύτερο μέρος του κώδικα δεν επαναχρησιμοποιείται μετά, καθώς οι μεταγενέστερες γενιές παιχνιδιών θα χρησιμοποιούσαν τελείως διαφορετικές σχεδιάσεις παιχνιδιών οι οποίες εκμεταλλεύονταν επιπλέον πόρους. Έτσι, οι περισσότερες σχεδιάσεις παιχνιδιών τη δεκαετία του 1980 αναπτύσσονταν διαμέσου αυστηρών κανόνων, μικρής ποσότητας δεδομένων επιπέδου και γραφικών. Οι μηχανές παιχνιδιών πρωτοεμφανίστηκαν στα μέσα της δεκαετίας του 1990. Επαναχρησιμοποιήσιμες μηχανές κάνουν την ανάπτυξη συνεχών παιχνιδιών γρηγορότερη και ευκολότερη, το οποίο είναι ένα βασικό πλεονέκτημα στην ανταγωνιστική βιομηχανία βιντεοπαιχνιδιών. Παράλληλα η τεχνολογία έχει αναπτυχθεί τόσο πολύ, με αποτέλεσμα να κυκλοφορούν δεκάδες μηχανές ανάπτυξης παιχνιδιών. Φυσικά, η καθεμία από αυτές έχει διαφορετικά δικαιώματα και δυνατότητες [40].

Μερικές από τις μηχανές που χρησιμοποιούν σήμερα έμπειροι προγραμματιστές και μη ώστε να δημιουργήσουν παιχνίδια τόσο μέσω scripting γλώσσας όσο και μέσω βιβλιοθηκών είναι :

- ❖ Το **Construct 2** που πρόκειται για μια πλατφόρμα HTML5 που χρησιμοποιείται για τη δημιουργία παιχνιδιών 2D. Είναι πολύ εύκολη στη χρήση και απευθύνεται κυρίως σε μη προγραμματιστές. Επιτρέπει την δημιουργία οπτικά ελκυστικών παιχνιδιών σε σύντομο χρονικό διάστημα χωρίς τη χρήση κώδικα [12].
- ❖ Το **AppGameKit** προσφέρει λύσεις για όλους τους τύπους προγραμματιστών από αρχάριους έως και προχωρημένους. Είναι μια φιλική προς το χρήστη εφαρμογή που χρησιμοποιεί τη δική της γλώσσα προγραμματισμού η οποία είναι παρόμοια με γλώσσα προγραμματισμού C++ [9].
- ❖ Το **Corona SDK "The 2D Game Engine"** είναι μια πλατφόρμα που χρησιμοποιεί τη γλώσσα προγραμματισμού Lua και χρησιμοποιείται για την ανάπτυξη παιχνιδιών 2D. Φημίζεται για τη σαφή τεκμηρίωση και την ενεργή υποστηρικτική κοινότητά της [12].
- ❖ Άλλη μια εφαρμογή που απευθύνεται σε όλους τους χρήστες είναι η **Unreal Engine** η οποία επιτρέπει να γίνουν αλλαγές στο παιχνίδι χωρίς προγραμματισμό. Ένα κύριο πλεονέκτημα της είναι η δυνατότητα δοκιμής του παιχνιδιού μέσα στην ίδια την εφαρμογή. Η **Unreal Engine** χρησιμοποιείται για τη δημιουργία τρισδιάστατων παιχνιδιών και προσφέρει εντυπωσιακά γραφικά [11].
- ❖ Η Apple ξεκίνησε τη δική της μηχανή παιχνιδιών τη **SpriteKit** για να δημιουργήσει 2D παιχνίδια. Αυτή η πλατφόρμα υποστηρίζει τις γλώσσες Swift και Objective-C και προσφέρει στους προγραμματιστές ένα ευρύ φάσμα δυνατοτήτων που θα τους βοηθήσει στη δημιουργία παιχνιδιών 2D υψηλής απόδοσης [12].
- ❖ Το **Marmalade SDK** είναι κατάλληλη για την ανάπτυξη παιχνιδιών για κινητά και επιτραπέζιους υπολογιστές. Χρησιμοποιεί την γλώσσα προγραμματισμού C++ και μπορεί να υποστηρίξει την δημιουργία 2D και 3D παιχνιδιών [10].
- ❖ Το **Amazon Lumberyard** είναι η μηχανή ανάπτυξης παιχνιδιών που ξεκίνησε από την Amazon. Προσφέρει δωρεάν εργαλεία για την δημιουργία παιχνιδιών και μπορεί να κυκλοφορήσει σε διαφορετικές συσκευές χάρη στη συμβατικότητα μεταξύ πλατφορμών [8].
- ❖ Το **Cocos2D-x** είναι ένα πακέτο εργαλείων ανάπτυξης παιχνιδιών 2D και κυκλοφόρησε από το Facebook. Ένα από τα καλύτερα χαρακτηριστικά του είναι ότι το **Cocos 2D-x** επιτρέπει την δημιουργία πολύ μικρού μεγέθους παιχνιδιών [10].
- ❖ Για την σχεδίαση και ανάπτυξη του παιχνιδιού της πτυχιακής προτιμήθηκε η **Unity Game Engine**. Πρόκειται για μια δωρεάν ολοκληρωμένη μηχανή ανάπτυξης παιχνιδιών ανοιχτού κώδικα, με ένα μεγάλο Forum, Fan Club και άμεση ανταπόκριση για βοήθεια. Επίσης, αναγνωρίζεται ως η πιο διαδεδομένη μηχανή ανάπτυξης σε ότι αφορά δημιουργία παιχνιδιών σε Smartphones και Tablet, παρέχει δυνατότητα για απόδοση 3D γραφικών και τα Script τρέχουν πιο γρήγορα από άλλες μηχανές. Το Unity Game Engine χαρακτηρίζεται για την ιδιαίτερη φιλικότητα ως προς το χρήστη, ενώ είναι αρκετά εύκολο στην εκμάθηση για κάποιον αρχάριο και παράλληλα προσφέρει πολλές δυνατότητες σε κάποιον έμπειρο χρήστη.

Κεφάλαιο 4ο – Unity

4.1 Εισαγωγή

Σε αυτό το κεφάλαιο αναφέρεται επί συντομία η ιστορία της Unity σημειώνοντας κάποιες σημαντικές ημερομηνίες ως προς την ίδρυση της. Εν συνέχεια παρουσιάζονται κάποια σημαντικά πλεονεκτήματα που προωθούν τον χρήστη να χρησιμοποιήσει τη Unity και όχι μία άλλη μηχανή ανάπτυξης. Ακόμα παρουσιάζεται εκτενώς το λογισμικό από τα διάφορα παράθυρα έως τα διάφορα μενού που βοηθάνε στη εύκολη χρήση και την ομαλή λειτουργία της μηχανής.

4.2 Unity

Η Unity (Σχήμα 4.1) ιδρύθηκε στην Κοπεγχάγη από τους Nicholas Francis, Joachim Ante και David Helgason. Η ιστορία ξεκίνησε όταν ο πρώτος εξέφρασε την επιθυμία του να φτιάξει μία μηχανή ανάπτυξης παιχνιδιών (Game Engine) σε ένα φόρουμ τον Μάιο του 2002. Αναζητούσε για συνεργάτες προγραμματιστές που ήταν εξειδικευμένοι στη δημιουργία παιχνιδιών που βασίζονται σε Mac όπως ο ίδιος.



Σχήμα 4.1 Logo Unity

Τελικά η εταιρία ιδρύθηκε το 2004 στη Δανία από τους τρεις παραπάνω προγραμματιστές. Η Unity Game Engine ανακοινώθηκε για πρώτη φορά και κυκλοφόρησε τον Ιούνιο του 2005 ως αποκλειστική μηχανή παιχνιδιών Mac OS X. Λίγο αργότερα γίνεται εφικτή η συμβατότητα της και με την Microsoft Windows. Η μηχανή γίνεται αγαπητή από το κοινό καθώς βοηθάει ανεξάρτητους Game Developers να φτιάξουν το δικό τους παιχνίδι όταν οι ίδιοι δεν είναι σε θέση να δημιουργήσουν τη δική τους μηχανή ανάπτυξης παιχνιδιών. Όμως η μεγάλη επιτυχία έγινε όταν ήρθε στην επιφάνεια το Iphone και το Appstore, διότι η μηχανή ήταν ήδη έτοιμη για την συγκεκριμένη πλατφόρμα. Σήμερα πλέον, η πλειοψηφία των προγραμματιστών που ασχολείται με την δημιουργία παιχνιδιών σε iOS και Android χρησιμοποιεί το Unity [22][23].

4.3 Πλεονεκτήματα

Αυτό που ξεχωρίζει το Unity από τα υπόλοιπα εργαλεία ανάπτυξης παιχνιδιών είναι η δυνατότητα που προσφέρει στον χρήστη να δημιουργήσει ένα παιχνίδι για οποιαδήποτε πλατφόρμα επιθυμεί. Συγκεκριμένα η έκδοση Unity 2020.1.9f1, που χρησιμοποιήθηκε για την πτυχιακή, προσφέρει τις πλατφόρμες : iOS, Android, Windows Phones, Macs, Windows PCs, Linux, Playstation, Xbox και Wii. Ακόμα, η μεταφορά του παιχνιδιού από την πλατφόρμα που δημιουργήθηκε σε μία άλλη γίνεται εύκολα και χωρίς να χρειάζονται αλλαγές ή κάποιες ιδιαίτερες ρυθμίσεις.

Με την πάροδο του χρόνου ολοένα και περισσότεροι προγραμματιστές χρησιμοποιούν το Unity, φτάνοντας τους εκατομμύρια χρήστες μέχρι και σήμερα. Αυτό έχει σαν αποτέλεσμα την δημιουργία μιας μεγάλης ενεργής και συνεχής αναπτυσσόμενης κοινότητας, η οποία προσφέρει σημαντική βοήθεια σε όλους τους χρήστες ανεξαρτήτως το επίπεδό τους. Απλοί χρήστες του λογισμικού δημιουργούν video tutorials και τα μοιράζονται δωρεάν με οποιοδήποτε επιθυμεί να ξεκινήσει την δική του εμπειρία στην ανάπτυξη βιντεοπαιχνιδιών. Επίσης το Unity παρέχει δωρεάν ζωντανά σεμινάρια σε απευθείας σύνδεση. Στα online αυτά μαθήματα τα οποία γίνονται από ειδικούς της Unity, παρουσιάζονται ολοκληρωμένα project και αναλύονται οι διάφορες ενότητες που τα απαρτίζουν. Μέσα από αυτά προσφέρονται κομμάτια κώδικα, διάφορα χρήσιμα tips και φυσικά η

δημιουργία ενός παιχνιδιού από την αρχή έως το τέλος.

Ακόμα η Unity παρέχει τη κοινότητα Forum and Answers , όπου στην οποία μπορεί ελεύθερα να εκφράσει κάθε χρήστης τις απορίες και τις δυσκολίες που μπορεί να αντιμετωπίζει κατά τη δημιουργία του παιχνιδιού. Μέσα στο Forum αναπτύσσονται συζητήσεις περί των αποριών που έχουν αναρτηθεί. Έτσι η γραπτή αλληλεπίδραση μεταξύ των μελών οδηγεί σε λύσεις και χρήσιμες συμβουλές που μπορεί να υιοθετήσει ο καθένας. Αντίθετα η ενότητα Answers, είναι για συγκεκριμένες ερωτήσεις που έχουν να κάνουν για οτιδήποτε αφορά θέματα developing. Συγκεκριμένα είναι μία βάση δεδομένων Q&A στην οποία αποθηκεύονται μαζικά όλες οι ερωτήσεις και στη συνέχεια είναι προσβάσιμες προς όλους είτε είναι εγγεγραμμένα μέλη είτε όχι. Με αυτόν το τρόπο ο καθένας μπορεί να απαντήσει στις ερωτήσεις και φυσικά να αναζητήσει λύση στις απορίες του σε μια πιθανόν ήδη απαντημένη ερώτηση.

Αξίζει φυσικά να σημειωθεί και το Asset Store της Unity. Πρόκειται για μία αυξανόμενη online βιβλιοθήκη που περιέχει δωρεάν και εμπορεύσιμα αντικείμενα που δημιουργήθηκαν τόσο από την Unity Technologies όσο και από τα μέλη της κοινότητας. Διαθέτει μία μεγάλη ποικιλία αντικειμένων, καλύπτοντας τα πάντα, από υφές, μοντέλα και κινούμενα σχέδια μέχρι κομμάτια κώδικα ή και ολόκληρα παραδείγματα παιχνιδιών. Η βιβλιοθήκη αυτή είναι διαθέσιμη σε όλους και επιτρέπει στον καθένα να προσκομίσει δωρεάν είτε με πληρωμή διάφορα αντικείμενα για τα παιχνίδια του. Η πρόσβαση στο Asset Store είναι εύκολη και γρήγορη μέσα από το Unity Editor και τα αντικείμενα εισάγονται απευθείας στο project που αναπτύσσεται. Οι χρήστες χρησιμοποιούν το Asset Store καθώς το περιεχόμενό του βοηθά να βελτιώσουν το παιχνίδι τους και να μειώσουν τον απαιτούμενο φόρτο εργασίας για τη δημιουργία μοντέλων ή εργαλείων.

Ένα ακόμη πλεονέκτημα του Unity είναι ότι δίνει τη δυνατότητα στο χρήστη να ενσωματώσει ήδη υπάρχοντα scripts σε διάφορα αντικείμενα. Έτσι, τα αντικείμενα αυτά αποκτούν ξεχωριστές ιδιότητες εύκολα και γρήγορα. Τα scripts είναι γραμμένα σε C# ή Javascript.

Τέλος, η Unity προσφέρει μια πλήρως λειτουργική δωρεάν έκδοση που μπορεί να χρησιμοποιηθεί για την ανάπτυξη ενός παιχνιδιού. Φυσικά, υπάρχει και η Pro Version, όπου αντίθετα με την κανονική είναι επί πληρωμή. Και οι δύο εκδόσεις είναι εύκολες και απλές στην χρήση τους αλλά η Pro Version διαθέτει επιπλέον χαρακτηριστικά και πιο ισχυρές δυνατότητες. Έτσι, μπορούν να δημιουργηθούν πιο σύνθετα, πιο εξελιγμένα και πιο όμορφα σε γραφικό τομέα παιχνίδια. Οι διαφορές μεταξύ των δύο εκδόσεων μπορούν να παρατηρηθούν στην Σχήμα 4.2.

UNITY 5		PERSONAL EDITION	PROFESSIONAL EDITION
What's included			
Engine with all features	ⓘ	✓	✓
Royalty-free	ⓘ	✓	✓
All platforms (limitations apply)	ⓘ	✓	✓
Customizable Splash Screen		✗	✓
Unity Cloud Build Pro - 12 Months	ⓘ	✗	✓
Unity Analytics Pro	ⓘ	✗	✓
Team License	ⓘ	✗	✓
Prioritized bug handling	ⓘ	✗	✓
Game Performance Reporting	ⓘ	✗	✓
Beta access	ⓘ	✗	✓
+ MORE FEATURES			
		FREE DOWNLOAD	FROM \$75/MONTH

Σχήμα 4.2 Free VS Pro Version Unity

4.4 Περιβάλλον Εργασίας Unity

4.4.1 Unity Editor

Ο Editor του Unity (Σχήμα 4.3) αποτελείται από διάφορα παράθυρα τα οποία κρατάνε το Project οργανωμένο και εύχρηστο. Φυσικά ο κάθε χρήστης έχει τη δυνατότητα να διαμορφώσει τη μορφή του Editor όπως επιθυμεί διαλέγοντας ένα από τα Layouts και τα παράθυρα που θέλει να φαίνονται. Τα σημαντικότερα παράθυρα είναι :

Scene View

Είναι ο χώρος στον οποίο γίνεται η κατασκευή του παιχνιδιού. Εκεί είναι τοποθετημένη αυτόματα η κάμερα και κατά την ανάπτυξη του παιχνιδιού μπαίνουν τα υπόλοιπα αντικείμενα που το απαρτίζουν. Οποιαδήποτε ρύθμιση στην οπτική γωνία της κάμερας και η επιλογή, ο χειρισμός καθώς και η τροποποίηση κάποιου αντικειμένου πραγματοποιείται μέσα στο Scene View.

Game View

Στο παράθυρο αυτό γίνεται η αναπαράσταση του παιχνιδιού. Ο χρήστης έχει τη δυνατότητα να τρέξει το παιχνίδι σε πραγματικό χρόνο και να ελέγξει βήμα βήμα ότι έχει υλοποιησει.

Hierarchy

Το παράθυρο Hierarchy περιέχει όλα τα αντικείμενα που βρίσκονται στη σκηνή αναλυτικά με τα ονόματα τους. Από προεπιλογή, τα αντικείμενα παρατίθενται στο παράθυρο αυτό με τη σειρά που δημιουργήθηκαν. Τα παράθυρα Hierarchy και Scene View είναι άμεσα συνδεδεμένα μεταξύ τους και όταν ένα αντικείμενο προστεθεί ή αφαιρεθεί από το ένα τότε το ίδιο συμβαίνει ταυτόχρονα και στο άλλο.

Project

Μέσα από αυτό, παρέχεται στο χρήστη η άμεση πρόσβαση και διαχείριση των Assets που έχει εισάγει στο project. Επομένως εκεί εμφανίζονται τα διάφορα στοιχεία που χρησιμοποιούνται για το παιχνίδι όπως prefabs, ήχοι, animations, textures κ.α. Ακόμα στο χώρο αυτό φαίνονται τα scripts που χαρίζουν διάφορες λειτουργίες στο σκηνικό και στα αντικείμενα.

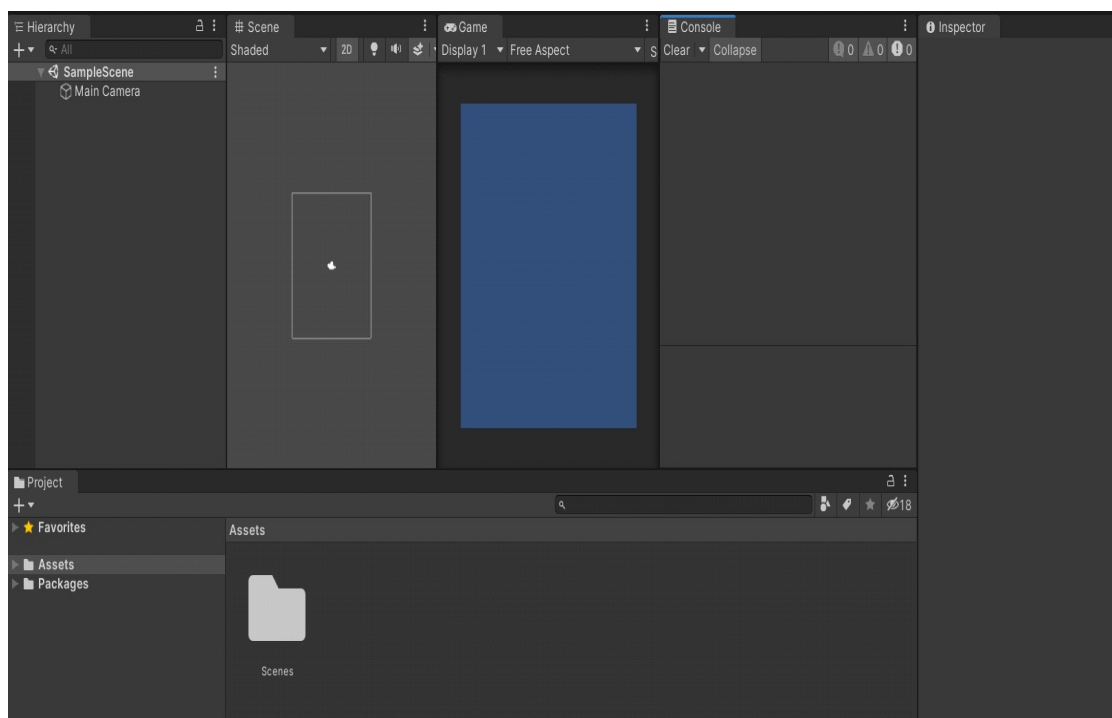
Inspector

Το παράθυρο Inspector χρησιμοποιείται για την προβολή και την επεξεργασία των ιδιοτήτων και των ρυθμίσεων των αντικειμένων. Κατά την επιλογή κάποιου στοιχείου εμφανίζονται τα βασικά χαρακτηριστικά του όπως η θέση (position), το μέγεθος (scale) και η περιστροφή (rotation). Επιπλέον ο χρήστης μπορεί να εισάγει και άλλες ιδιότητες στο αντικείμενο ώστε να το κάνει πιο ενδιαφέρον όπως:

- Box Collider 2D*: Ορίζει ένα αόρατο πλέγμα περιφερειακά ενός αντικειμένου. Σκοπός του είναι να επιτρέπει τις φυσικές συγκρούσεις μεταξύ αντικειμένων. Εάν ενεργοποιηθεί η λειτουργία trigger τότε μέσω κώδικα υπάρχει η δυνατότητα να οριστούν διάφορες λειτουργίες ως προς την αλληλεπίδραση των αντικειμένων.
- RigidBody 2D*: Πρόκειται για τη φυσική υπόσταση ενός αντικειμένου. Με τον τρόπο αυτόν το αντικείμενο αποκτά μάζα και μπορεί να κινείται στο χώρο βάση των νόμων της φυσικής.
- Animator*: Ρυθμίζει την ομαλή εναλλαγή κινήσεων των animations. Αυτές συνδέονται μεταξύ τους με μια παράμετρο ως συνθήκη αλλαγής, η οποία μπορεί να αλλάξει εύκολα μέσω κώδικα.
- Script*: Αρχεία κώδικα σε C#. Ορίζουν τη συμπεριφορά του αντικειμένου μέσα στο χώρο καθώς και την αλληλεπίδραση του με τα άλλα αντικείμενα.

Console

Το παράθυρο Console εμφανίζει μηνύματα σφαλμάτων αλλά και προειδοποιήσεων που προκύπτουν κατά την εκτέλεση των scripts.



Σχήμα 4.3 Unity Editor

4.4.2 Menu

Το βασικό μενού (Σχήμα 4.4) περιέχει όλες τις επιλογές για το περιβάλλον ανάπτυξης ενός παιχνιδιού στο Unity και περιέχει τις εξής καρτέλες:

File Edit Assets GameObject Component Window Help

Σχήμα 4.4 Βασικό Μενού

File: Από την καρτέλα File ο χρήστης μπορεί να δημιουργήσει, να ανοίξει και να αποθηκεύσει μία σκηνή ή ένα project. Ακόμα από αυτήν την καρτέλα γίνεται η επιλογή της πλατφόρμας στην οποία θα τρέχει το παιχνίδι.

Edit: Η καρτέλα Edit προσφέρει την δυνατότητα αναίρεσης-επαναφοράς μιας ενέργειας, την αποκοπή, την αντιγραφή, την επικόλληση και την διαγραφή ενός αντικειμένου. Επίσης μέσα από αυτήν μπορεί να γίνει η επιλογή συγκεκριμένων σκηνών και η έναρξη, η παύση και η συνέχιση του παιχνιδιού. Καθιστά εφικτή την αναζήτηση, την επιλογή όλων των αντικειμένων στην σκηνή, την διαμόρφωση της λειτουργίας και της εμφάνισης του Unity. Επιπλέον ο χρήστης έχει πρόσβαση στο λογαριασμό του μέσα από την καρτέλα Edit, κάτι που κάνει ιδιαίτερα εύκολη τη χρήση του ηλεκτρονικού καταστήματος.

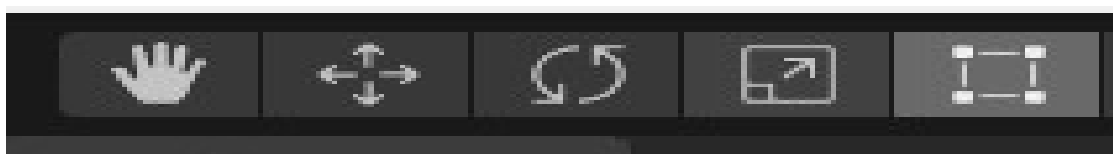
Asset: Αυτή καρτέλα βρίσκεται επίσης και στο παράθυρο Project του Editor. Παρέχει την δυνατότητα εισαγωγής καινούργιων στοιχείων στο project καθώς και την εισαγωγή πακέτων με στοιχεία τα οποία παρέχει η ίδια η Unity μέσα από το Asset Store.

GameObject: Η καρτέλα GameObject μπορεί να χαρακτηριστεί ως η βιβλιοθήκη των αντικειμένων. Σε αυτήν υπάρχουν μαζικά τα στοιχεία από όπου μπορεί να διαλέξει ο χρήστης και να τοποθετήσει

στο σκηνικό του. Μερικά από αυτά είναι τα αντικείμενα 2D ή 3D, ο φωτισμός, η κάμερα, ο ήχος κτλ. Τα αντικείμενα αυτά μπορούν να σταθούν στο χώρο αυτόνομα ή να έχουν μία σχέση child-parent με άλλα αντικείμενα.

Component: Αυτή καρτέλα παρέχει στον χρήστη διάφορες ιδιότητες που μπορεί να ενσωματώσει στα αντικείμενα του παιχνιδιού ώστε να γίνουν πιο ρεαλιστικά. Οι συγκρούσεις, οι καταστροφές και οποιαδήποτε άλλη αλληλεπίδραση μεταξύ των αντικειμένων γίνεται εφικτή με τη χρήση της ιδιότητας *colliders*. Ταυτόχρονα η ιδιότητα *RigidBody* προσφέρει βαρύτητα και τη ψευδαίσθηση ενός φυσικού σώματος. Τέλος, μέσω του *navigation* ο χρήστης αντιλαμβάνεται ποια μέρη στο παιχνίδι είναι προσβάσιμα καθώς κινείται ο χαρακτήρας ώστε να μην υπάρχουν προβλήματα γραφικών και κίνησης.

Οι επιλογές του μενού διαχείρισης αντικειμένων (Σχήμα 4.5) μπορούν να χρησιμοποιηθούν κατά τη λειτουργία του *SceneView*.



Σχήμα 4.5 Transform tools Menu

Hand Tool: Χρησιμοποιείται για την μετακίνηση της κάμερας και κατά συνέπεια του σκηνικού. Μία ακόμα χρήση του είναι για ζουμ προκειμένου να πραγματοποιηθούν οι κατάλληλες αλλαγές σε μέρη ή αντικείμενα του χώρου.

Translate Tool: Με αυτό το εργαλείο επιλέγονται και τοποθετούνται τα αντικείμενα στο σκηνικό.

Rotate Tool: Μετά την επιλογή ενός αντικειμένου, χρησιμοποιείται για τη μετακίνηση του ως προς τους άξονες x, y, z.

Scale Tool: Επιλέγοντας το εργαλείο αυτό αλλάζουν οι διαστάσεις ενός αντικειμένου πατώντας απλά στο κέντρο του.

Rect Tool: Χρησιμοποιείται για την αλλαγή της κλίμακας ενός αντικειμένου.

Το μενού ελέγχου προβολής παιχνιδιού (Σχήμα 4.6) χρησιμοποιείται κατά την λειτουργία του *Game View*.



Σχήμα 4.6 Game View Controls Menu

Play Button: Το κουμπί *Play* επιτρέπει την αναπαραγωγή της τρέχουσας σκηνής. Το παιχνίδι εκτελείται στην τελική του μορφή και λειτουργούν όλες οι κινήσεις, τα *animations*, οι ήχοι και τα εφέ. Η εκτέλεση του παιχνιδιού σταματάει κάνοντας ξανά κλικ στο κουμπί *Play*.

Pause Button: Το κουμπί *Pause* διακόπτει την εκτέλεση της τρέχουσας προβολής παιχνιδιού. Το παιχνίδι θα διατηρήσει την κατάστασή του και θα συνεχίσει ακριβώς εκεί που βρισκόταν σε παύση. Κάνοντας ξανά κλικ στο κουμπί *Pause* θα συνεχίσει να τρέχει το παιχνίδι.

Step Button: Το κουμπί *Step* λειτουργεί ενώ η προβολή Παιχνιδιού βρίσκεται σε παύση και προκαλεί στο παιχνίδι να εκτελέσει ένα μόνο πλαίσιο του παιχνιδιού. Αυτό επιτρέπει στο χρήστη να προχωρήσει το παιχνίδι αργά και να εντοπίσει τυχόν προβλήματα που ενδέχεται να έχει. Κάνοντας ξανά κλικ στο κουμπί *Step* το παιχνίδι διακόπτεται.

4.5 Σημαντικά Παράθυρα και Εργαλεία

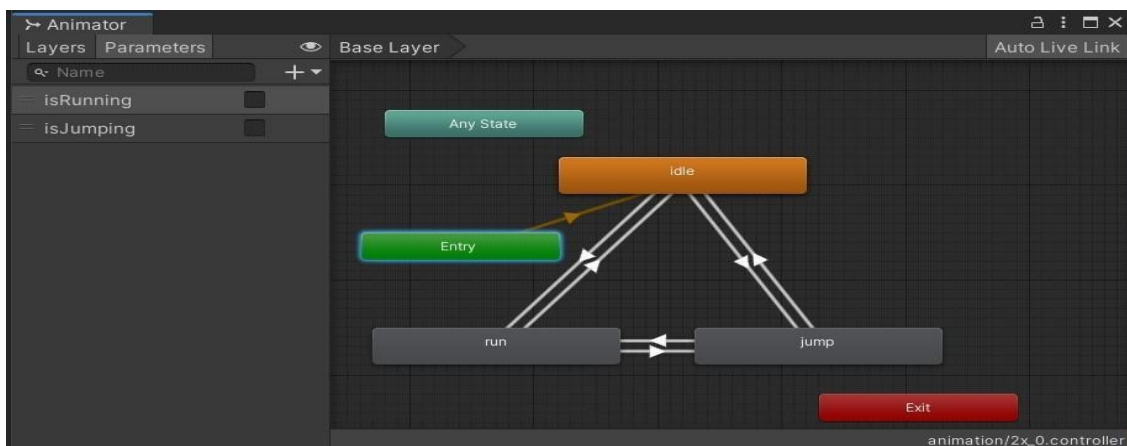
Sprite Editor: Ο Sprite Editor χρησιμοποιείται για να τεμαχίσει σε ξεχωριστά διαστάσιμα αντικείμενα γραφικών (Sprites) μία μεγαλύτερη Σχήμα που αποτελείται από άλλες μικρότερες (Sprite sheet). Συχνά τα sprites χρησιμοποιούνται ώστε να συνθέσουμε ένα animation.

Animation Window: Ένα animation είναι μια αλληλουχία από διαφορετικά frames, με το κάθε frame να ορίζει μια διαφορετική κατάσταση του αντικειμένου, με αποτέλεσμα τη ψευδαίσθηση της κίνησης και της αλλαγής. Σε 2D ένας τρόπος για να επιτευχθεί ένα τέτοιο αποτέλεσμα είναι μέσω ενός sprite sheet του οποίου το κάθε sprite αναπαριστά τον χαρακτήρα σε διαφορετικό σημείο της κίνησής του. Είναι προφανές πως όσο πιο πολλά frames υπάρχουν σε ένα animation, τόσο πιο λεπτομερές και όμορφο θα είναι. Τα frames μπορούν να δημιουργηθούν μέσα από το παράθυρο Animation του Unity (Σχήμα 4.7) και μπορούν να επηρεάζουν πολλά και διαφορετικά properties ενός αντικειμένου. Τα 2D animations μπορούν επίσης να φτιαχτούν μεταφέροντας πολλαπλά sprites από το παράθυρο του project μέσα στη σκηνή.



Σχήμα 4.7 Player Run Animation

Animator Window: Ο τρόπος διαχείρισης όλων των animation ενός αντικειμένου ρυθμίζεται μέσα από το παράθυρο Animator (Σχήμα 4.8). Κάθε animation αναπαρίσταται από ένα state. Ο χαρακτήρας θα ξεκινήσει εκτελώντας το default state το οποίο είναι το πρώτο animation που φτιάχτηκε, εκτός αν οριστεί διαφορετικά. Για τη μετάβαση από ένα animation σε ένα άλλο χρησιμοποιούνται τα transitions, τα οποία ενεργοποιούνται βάσει κάποιων παραμέτρων. Οι παράμετροι αυτοί μπορούν να είναι αριθμητικές τιμές, Boolean τιμές ή triggers. Η ενεργοποίηση ή μεταβολή των παραμέτρων γίνεται μέσω κώδικα, συγκεκριμένα με τις συναρτήσεις της κλάσης Animator: SetFloat, SetBool, SetTrigger.



Σχήμα 4.8 Player Animator

Prefabs: Τα Prefab είναι ένας ειδικός τύπος στοιχείου που επιτρέπει την πλήρη αποθήκευση αντικειμένων στο Project ώστε να χρησιμοποιηθούν ξανά αργότερα. Αυτά τα στοιχεία μπορούν στη συνέχεια να μοιραστούν μεταξύ σκηνών ή ακόμη και άλλων έργων χωρίς να χρειάζεται να διαμορφωθούν ξανά.

Tags: Ένα tag μπορεί να χρησιμοποιηθεί για την αναγνώριση ενός αντικειμένου στο παιχνίδι και μέσα από scripts να δοθούν διάφορες ικανότητες στο αντικείμενο αυτό.

Render Layer: Χρησιμοποιείται για την ρύθμιση του επιπέδου απεικόνιση ενός γραφικού. Τα γραφικά που ανήκουν σε κατώτερο επίπεδο καλύπτονται από αυτά που βρίσκονται σε ανώτερο.

Κεφάλαιο 5 - Προγραμματισμός

5.1 Γλώσσα Προγραμματισμού C#

Η C# είναι μία multi-paradigm γλώσσα προγραμματισμού η οποία έχει κοινά χαρακτηριστικά με την C, καθώς προέρχεται από αυτή. Σχεδιάστηκε και λειτουργεί από την Microsoft στα πλαίσια της πλατφόρμας .Net. Εδώ και δύο δεκαετίες έχει αναγνωριστεί και επισήμως από την Emca και την ISO.

Πρόκειται για μία απλή, σύγχρονη, γενικής χρήσης αντικειμενοστραφής γλώσσα προγραμματισμού. Μερικά στοιχεία που την χαρακτηρίζουν είναι ότι είναι στατική, δυναμική, ισχυρή, ασφαλής και ονομαστική. Δημιουργήθηκε το 1999, από μία ομάδα προγραμματιστών με επικεφαλής τον Anders Hejlsberg. Στις 10 Νοεμβρίου 2020 κυκλοφόρησε η έκδοση 9.0 η οποία είναι η πιο πρόσφατη μέχρι και σήμερα.

Επιπλέον είναι ειδικά σχεδιασμένη για την ανάπτυξη λογισμικού σε .Net Framework. Οτιδήποτε εμπεριέχεται στην C# είναι αντικείμενα. Αξιοσημείωτο είναι το γεγονός ότι προσφέρει απευθείας πρόσβαση σε τεράστιες βιβλιοθήκες κλάσεων του .Net Framework [29].

5.2 Από τη C στη C#

Η C είναι μία εύκολη, με μικρό πλήθος εντολών γλώσσα προγραμματισμού που ξεκίνησε ως στενός συνεργάτης του Unix. Από αυτήν γεννήθηκαν οι γλώσσες C++, JAVA και C#. Η C++ αρχικά σχεδιάστηκε ως μία βελτίωση της C και για αυτό το λόγο κληρονομεί όλα τα πλεονεκτήματα της αλλά ταυτόχρονα έχει επιπλέον χαρακτηριστικά, όπως ενθυλάκωση, κληρονομικότητα και πολυμορφισμό. Σημαντικό μειονέκτημα της γλώσσας αυτής είναι η αδυναμία να μεταφερθεί σε διαφορετικά περιβάλλοντα. Λύση σε αυτό δίνει η Java, η οποία είναι μια καθαρά αντικειμενοστραφής γλώσσα προγραμματισμού ανοιχτού κώδικα που μπορεί να εκτελεστεί σε οποιοδήποτε περιβάλλον διαθέτει την εικονική μηχανή JAVA. Παρόλο που μπορεί να εκτελεστεί στα Windows δεν συνδέεται με αυτά και ο κώδικας που παράγεται δεν συνεργάζεται με κώδικα δεύτερης γλώσσας ώστε να δημιουργηθούν μεγάλα καταναμημένα συστήματα. Για τον λόγο αυτό, δημιουργήθηκε η C#, η οποία είναι ένας συνδυασμός όλων των άλλων γλωσσών προγραμματισμού σε σχεδόν τέλεια ισορροπία [36][37][38].

5.3 .NET Framework

Το .NET Framework είναι ένα πλαίσιο λογισμικού (software framework) που προορίζεται για την πλατφόρμα των Windows. Αποτελείται από μια μεγάλη βιβλιοθήκη κλάσεων και υποστηρίζει δεκάδες γλώσσες προγραμματισμού με τη δυνατότητα η μια να μπορεί να χρησιμοποιηθεί από την άλλη. Τα προγράμματα που γράφονται για το .NET Framework εκτελούνται σε ένα περιβάλλον εκτέλεσης γνωστό ως Common Language Runtime (CLR), ενός ειδικού λογισμικού, σχεδιασμένου να υποστηρίζει την εκτέλεση προγραμμάτων. Το CLR περιέχει μια εικονική μηχανή που διαχειρίζεται την εκτέλεση ενός προγράμματος και παρέχει μια σειρά σημαντικών υπηρεσιών όπως ασφάλεια, διαχείριση μνήμης και διαχείριση εξαιρέσεων. Τα προγράμματα που γράφονται για το .NET χρησιμοποιούν τη βιβλιοθήκη κλάσεων του .NET, η οποία δίνει πρόσβαση στο περιβάλλον εκτέλεσής του. Βασικές λειτουργίες όπως οι γραφικές διεπαφές χρηστών (Graphical User Interfaces – GUIs), η επικοινωνία με βάσεις δεδομένων, η κρυπτογραφία, η ανάπτυξη web εφαρμογών και οι δικτυακές επικοινωνίες παρέχονται μέσω του Application Programming Interface (API) του .NET και μπορούν να συνδυαστούν με κώδικα από τους προγραμματιστές για τη δημιουργία ολοκληρωμένων εφαρμογών. Όσο ένα πρόγραμμα περιορίζεται στη χρήση της βιβλιοθήκης κλάσεων του .NET, μπορεί να τρέχει οπουδήποτε υπάρχει εγκατεστημένο το περιβάλλον εκτέλεσης του .NET. Η γλώσσα C# είναι μια από τις γλώσσες που υποστηρίζει το .NET. Ο μεταγλωττιστής της C# στοχεύει ειδικά στο .NET κάτι που

που σημαίνει ότι τα προγράμματα γραμμένα σε C# θα τρέχουν πάντα στο .NET Framework [34][35].

5.4 Monodevelop

Οι δυνατότητες που παρέχει η Unity είναι πραγματικά εκπληκτικές. Ωστόσο χρειάζεται κώδικας για να ολοκληρωθεί το παιχνίδι και να βγει το τελικό αποτέλεσμα εντυπωσιακό. Αυτός ο κώδικας δημιουργείται ώστε να καλύψει τις διάφορες απαιτήσεις του παιχνιδιού. Το πρόγραμμα που προσφέρει στο χρήστη τη δυνατότητα να γράψει τον κώδικα του παιχνιδιού ώστε να ανταπεξέλθει στις απαιτήσεις του είναι το MonoDevelop [30].

Ο κώδικας μπορεί να προστεθεί στα επιθυμητά αντικείμενα του εκάστοτε Project είτε προσθέτοντας ως Component κάποιο Script ή μέσω δημιουργίας ενός script από το παράθυρο Project του editor. Έπειτα το script ανοίγει κατ' επιλογήν με το πρόγραμμα Monodevelop για να γίνει δυνατή η επεξεργασία του.

Η Unity σε συνδυασμό με το Monodevelop υποστηρίζει τρεις γλώσσες προγραμματισμού για την κατασκευή των παιχνιδιών της. Αυτές είναι:

- C#: Είναι μία αντικειμενοστραφής γλώσσα προγραμματισμού που αναπτύχθηκε από τη Microsoft.
- JavaScript: Είναι μια διερμηνευμένη γλώσσα προγραμματισμού επηρεασμένη από τη σύνταξη της C και δημιουργημένη από τη Netscape.
- Boo: Είναι μία στατικά δακτυλογραφημένη, γλώσσα προγραμματισμού γενικού σκοπού εμπνευσμένη από τη σύνταξη της Python και αναπτυγμένη από την Unity Technologies.

Οι τρεις παραπάνω γλώσσες είναι αντικειμενοστραφείς και είναι ο κύριος λόγος που έχουν επιλεγεί από την Unity για την ανάπτυξη των παιχνιδιών της.

Το MonoDevelop είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (Integrated Development Environment – IDE) ανοιχτού κώδικα για λειτουργικά συστήματα Linux, MacOS και Windows. Πρωταρχικός στόχος της είναι η ανάπτυξη project που χρησιμοποιούν Mono και .NET Framework. Το MonoDevelop ενσωματώνει χαρακτηριστικά παρόμοια με αυτά των NetBeans και Microsoft Visual Studio, όπως αυτόματη ολοκλήρωση κώδικα, έλεγχο πηγής, γραφικό περιβάλλον εργασίας χρήστη (Graphical User Interface -GUI) και σχεδιασμό ιστοχώρων [30].

5.5 C# στη Unity - MonoBehaviour

Η Unity μπορεί να χρησιμοποιήσει ένα σύνολο script, τα οποία αλληλεπιδρούν μεταξύ τους αλλάζοντας μεταβλητές. Κατά τη δημιουργία ενός script, μετά τη δήλωση της κλάσης, η οποία έχει το ίδιο όνομα με το όνομα του script by default, δηλώνονται όλες τις μεταβλητές που θέλει ο χρήστης να χρησιμοποιήσει για το αντικείμενο στο οποίο θα επισυνάψει το script. Ένα script μπορεί να περιέχει πολλές κλάσεις, καθώς και υποκλάσεις οι οποίες κληρονομούν ιδιότητες από άλλες κλάσεις του ίδιου script ή του πηγαίου κώδικα της Unity [32].

Η MonoBehaviour αποτελεί τη βασική κλάση από την οποία προέρχεται το κάθε script της C# και μέσω αυτής γίνεται χρήση διαφόρων συναρτήσεων. Οι συναρτήσεις Void Start() και Void Update() είναι δύο από τις βασικές συναρτήσεις της MonoBehaviour. Υπάρχουν πάντα by default σε όλα τα scripts που δημιουργούνται και χρησιμοποιούνται για τις περισσότερες λειτουργίες των αντικειμένων. Εκτός από αυτές τις δύο βασικές συναρτήσεις, η MonoBehaviour προσφέρει ακόμα τις Void Awake(), Void FixedUpdate(), Void LateUpdate(), Void OnGUI(), Void OnDisable() και Void OnEnable() [32].

Η Void Start() καλείται στο frame όταν ενεργοποιηθεί ένα script, λίγο πριν κληθεί μια μέθοδος update για πρώτη φορά. Και η συνάρτηση Void Awake() και η Void Start() καλείται ακριβώς μία φορά στη διάρκεια ζωής του script. Ωστόσο, η Void Awake() καλείται όταν το αντικείμενο του script αρχικοποιείται, ανεξάρτητα από το αν το σενάριο είναι ενεργοποιημένο ή όχι. Η συνάρτηση Void Awake() καλείται σε όλα τα αντικείμενα της σκηνής πριν από την κλήση της συνάρτησης Void Start() οποιουδήποτε αντικειμένου. Όταν τα αντικείμενα δημιουργούνται κατά τη διάρκεια του παιχνιδιού, η συνάρτηση Void Awake() καλείται μετά την ολοκλήρωση των λειτουργιών της Void Start().

Η Void Update() συχνά περιλαμβάνει το περισσότερο κομμάτι του κώδικα ενός script και τον εκτελεί σε κάθε frame, δηλαδή πολλές φορές το δευτερόλεπτο, ανάλογες με το frame rate, κατά την διάρκεια του παιχνιδιού. Για αυτό το λόγο στην Void Update() προσθέτουμε λειτουργίες με παραμέτρους που θέλουμε να αλλάζουν σε πραγματικό χρόνο όσο λειτουργεί το παιχνίδι. Για παράδειγμα, τις κινήσεις και τις αλληλεπιδράσεις των αντικειμένων.

Η Void FixedUpdate() χρησιμοποιείται για να αντικαταστήσει την Void Update(), όταν στο πρόγραμμα χρησιμοποιείται η Rigidbody, η οποία ελέγχει την θέση ενός αντικειμένου με βάση τους νόμους της φυσικής.

Η Void LateUpdate() καλείται σε κάθε frame μετά την κλήση όλων των λειτουργιών της Update. Αυτό είναι χρήσιμο καθώς οργανώνει τη σειρά εκτέλεσης των συναρτήσεων. Για παράδειγμα, η κάμερα θα πρέπει πάντα να εφαρμόζεται στη Void LateUpdate() επειδή παρακολουθεί αντικείμενα που ενδέχεται να έχουν μετακινηθεί μέσα στην συνάρτηση Update.

Η Void OnGUI() καλείται για απόδοση και διαχείριση συμβάντων GUI (Graphical user interface). Η υλοποίηση του OnGUI ενδέχεται να καλείται πολλές φορές ανά frame. Εάν η ιδιότητα MonoBehaviour έχει οριστεί σε false, το OnGUI () δεν θα κληθεί.

Η Void OnDisable() καλείται όταν ένα αντικείμενο καταστρέφεται και μπορεί να χρησιμοποιηθεί για οποιονδήποτε κώδικα εκκαθάρισης. Αντίθετα η Void OnEnable() καλείται όταν ένα αντικείμενο ενεργοποιηθεί. Όλες αυτές οι συναρτήσεις βοηθούν τον χρήστη να έχει τον πλήρη έλεγχο σε όσα έχει σχεδιάσει.

5.5.1 Αλληλεπίδραση αντικειμένων

Ένα σημαντικό στοιχείο της διαδραστικότητας στα παιχνίδια είναι η δυνατότητα να εκτελούνται γεγονότα με βάση το αν ο παίκτης βρίσκεται σε κάποια συγκεκριμένη περιοχή ή αν ήρθε σε σύγκρουση με κάποιο αντικείμενο.

Για το σκοπό αυτό χρησιμοποιείται το component του collider, το οποίο χρησιμοποιείται για να ορίσει μια περιοχή γύρω από ένα αντικείμενο η οποία συμπεριφέρεται σαν εμπόδιο για τον παίκτη. Εναλλακτικά, ορίζοντας μέσα από τον editor τον collider ως trigger, η περιοχή δεν εμποδίζει με κάποιο τρόπο τον παίκτη.

Είτε το component λειτουργεί σαν collider είτε σαν trigger, υπάρχει η δυνατότητα μέσα από τον κώδικα να αναγνωρίζεται τότε κάποιο αντικείμενο εισέρχεται σε ένα trigger ή συγκρούεται με ένα collider και να διατηρούνται κάποιες πληροφορίες σχετικά με το γεγονός αυτό.

Οι συναρτήσεις δεν καλούνται κάπου, αλλά ορίζονται από τον προγραμματιστή μέσα στο script και η Unity τις καλεί ανάλογα με την περίπτωση. Σημειώνεται πως ενώ οι συναρτήσεις OnCollision παίρνουν ως όρισμα ένα Collision (κλάση που περιέχει πληροφορίες ως προς τη σύγκρουση αντικειμένων), οι συναρτήσεις OnTrigger παίρνουν ως όρισμα ένα Collider. Συγκεκριμένα τον collider που ανήκει στο αντικείμενο που εισήλθε/εξήλθε στη περιοχή που ορίζει το trigger. Σημειώνεται πως για να κληθούν επιτυχώς οι συναρτήσεις OnCollision θα πρέπει τουλάχιστον ένα από τα αντικείμενα που συμμετέχουν να έχει non kinematic Rigidbody component. Αντίθετα στις συναρτήσεις OnTrigger χρειάζεται τουλάχιστον ένα από τα συμμετέχοντα αντικείμενα να έχει ένα Rigidbody component.

`Void OnCollisionEnter()`: Εκτελείται όταν το αντικείμενο που έχει σαν component το script με τη συγκεκριμένη συνάρτηση έρθει σε επαφή με ένα άλλο αντικείμενο.

`Void OnTriggerEnter()`: Εκτελείται όταν ένα αντικείμενο εισέρχεται στη περιοχή του trigger του αντικειμένου που έχει σαν component το script με τη συγκεκριμένη συνάρτηση.

`Void OnCollisionStay()`: Εκτελείται μία φορά ανά frame για κάθε αντικείμενο που συγκρούεται με το αντικείμενο που έχει σαν component το script με τη συγκεκριμένη συνάρτηση

`Void OnTriggerStay()`: Εκτελείται όσο ένα άλλο αντικείμενο βρίσκεται εντός της περιοχής του trigger του αντικειμένου που έχει σαν component το script με τη συγκεκριμένη συνάρτηση.

`Void OnCollisionExit()`: Εκτελείται όταν το αντικείμενο που έχει σαν component το script με τη συγκεκριμένη συνάρτηση δεν είναι πλέον σε επαφή με ένα άλλο αντικείμενο.

`Void OnTriggerExit()`: Εκτελείται όταν ένα αντικείμενο δεν είναι πλέον μέσα στη περιοχή του trigger, του αντικειμένου που έχει σαν component το script με τη συγκεκριμένη συνάρτηση.

Επειδή συχνά επιθυμείτε η ενεργοποίηση των συναρτήσεων μόνο από συγκεκριμένα αντικείμενα χρησιμοποιούνται τα tags για έλεγχο. Τα tags σε κάθε αντικείμενο ορίζονται μέσα από τον editor.

Κεφάλαιο 6 – Υλοποίηση

Στο κεφάλαιο αυτό παρουσιάζεται ολόκληρη η ανάπτυξη και η εκτέλεση του παιχνιδιού που αφορά την συγκεκριμένη πτυχιακή εργασία. Η υλοποίηση του παιχνιδιού πραγματοποιήθηκε με την βοήθεια της Unity Game Engine και η γλώσσα προγραμματισμού που χρησιμοποιήθηκε είναι η C#.

Το παιχνίδι που αναπτύχθηκε είναι δύο διαστάσεων (2D), υπόκειται στην γενική κατηγορία παιχνιδιών δράσης και ειδικότερα στην υποκατηγορία των παιχνιδιών πλατφόρμας. Σε αυτό το παιχνίδι ο παίκτης κινείται ελεύθερα μέσα στη πίστα, έχοντας σαν στόχο την αποφυγή των εμποδίων και των εχθρών που προσφέρει το εκάστοτε επίπεδο.

6.1 Assets

Όλες οι σκηνές που αποτελούν το παιχνίδι, δημιουργήθηκαν με τη βοήθεια του Asset-Store που προσφέρει η Unity. Συγκεκριμένα,

- Ο παίκτης
- οι διάφοροι εχθροί
- τα εδάφη που αποτελούν τις πλατφόρμες σε όλα τα επίπεδα
- η θάλασσα

είναι αντικείμενα που μπορεί ο καθένας να βρει στο δωρεάν asset pack "Free Platform Game Assets".

Τα διάφορα background, σύννεφα και βουνά που χρησιμοποιήθηκαν για όλες τις σκηνές του παιχνιδιού είναι προσβάσιμα από όλους από το επίσης δωρεάν asset pack "Free 2D Adventure Beach Pack". Για την επεξεργασία και την εμφάνιση κειμένου στις σκηνές των Main Menu, Pause Menu και LoadingLevel χρησιμοποιήθηκε το "Textmesh Pro".

6.2 Σκηνές (Scenes)

6.2.1 LogInScene

Η σκηνή "LogIn" (Σχήμα 6.1) έχει οριστεί ως η πρώτη σκηνή του παιχνιδιού. Αποτελείται από δύο πεδία (InputField) και ένα κουμπί. Στο πρώτο πεδίο ο χρήστης συμπληρώνει ένα email και στο δεύτερο ορίζει ένα κωδικό. Στη συνέχεια πατώντας το κουμπί LogIn τα δεδομένα αποθηκεύονται στη πλατφόρμα Firebase και ξεκινάει η ροή του παιχνιδιού.

Το Firebase είναι μια πλατφόρμα που αναπτύχθηκε από την Google για τη δημιουργία εφαρμογών για κινητά αλλά και τον ιστό. Συγκεκριμένα μέσω της ιστοσελίδας <https://firebase.google.com/docs/auth/unity/password-auth> και πατώντας την επιλογή "Gotoconsole" δίνεται η δυνατότητα δημιουργίας ενός project. Κατά την δημιουργία του project πρέπει να επιλεγεί η επιλογή "Register as an Android App" και το ίδιο όνομα που θα οριστεί στην επιλογή "Android package" να συμπληρωθεί αργότερα μέσα στο Unity από την καρτέλα "Build and run – Player Settings". Για να μπορέσει να λειτουργήσει η Firebase με τη Unity ακόμα πρέπει να γίνει εισαγωγή των αρχείων "google-services.json" και "FirebaseUnitySDK" μέσα στο project της Unity.



Σχήμα 6.1 LogIn Scene

6.2.2 Menu Scene

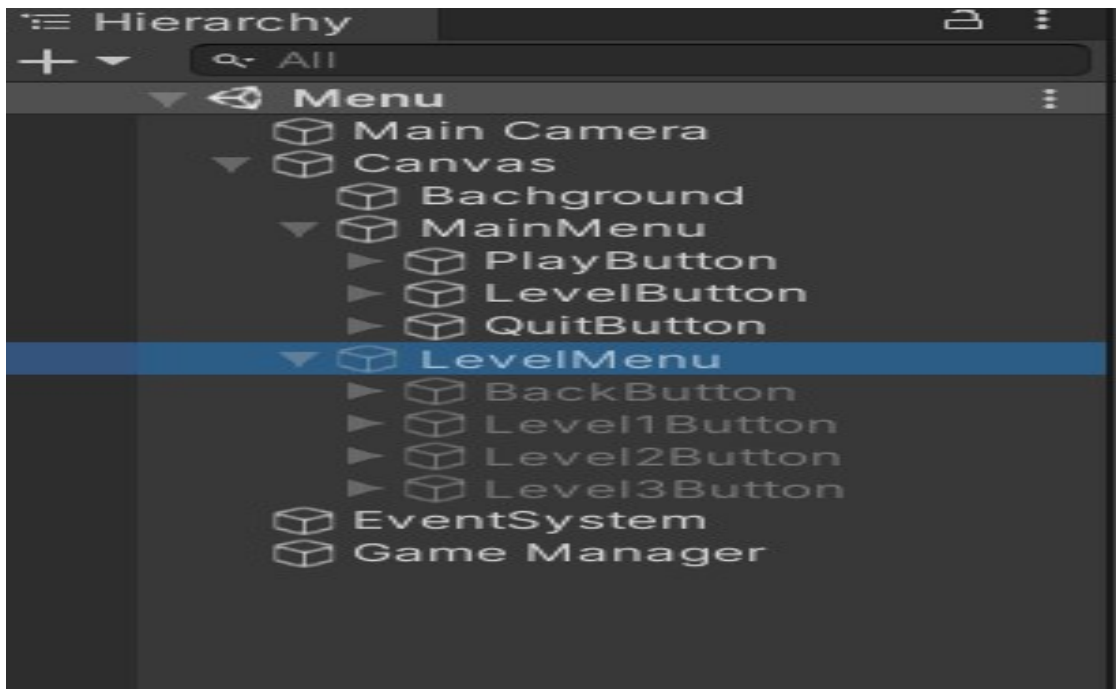
Η σκηνή "Menu" (Σχήμα 6.2) αποτελείται από δύο menu, το MainMenu και το LevelMenu. Στην πραγματικότητα αυτά τα menu είναι δύο gameobject, που περιλαμβάνουν κουμπιά με συγκεκριμένες λειτουργίες οι οποίες πραγματοποιούνται μέσω scripts.

Το παιχνίδι ξεκινάει με ενεργοποιημένο το MainMenu το οποίο αποτελείται από τρία κουμπιά το PlayButton, το LevelButton και το QuitButton.

Εάν ο χρήστης επιλέξει το κουμπί PlayButton, ξεκινάει η ροή του παιχνιδιού. Μετά το πέρας της πίστας ξεκλειδώνεται η επόμενη. Η συνάρτηση που χρησιμοποιείται είναι η SceneManager.LoadScene() και για να μπορεί να λειτουργήσει πρέπει να γίνει η χρήση της βιβλιοθήκης UnityEngine.SceneManagement.

Εάν ο χρήστης επιλέξει το κουμπί LevelButton, απενεργοποιείται το MainMenu και ενεργοποιείται το LevelMenu. Μέσα από αυτό ο παίκτης μπορεί να μεταβεί σε όποια πίστα θέλει πατώντας απλά το κουμπί του επιπέδου. Τα επίπεδα στην αρχή είναι κλειδωμένα και είναι μόνο ενεργό το κουμπί του επιπέδου ένα. Ακόμα υπάρχει το κουμπί BackButton που επιτρέπει στο χρήστη να επανέλθει στο κεντρικό μενού.

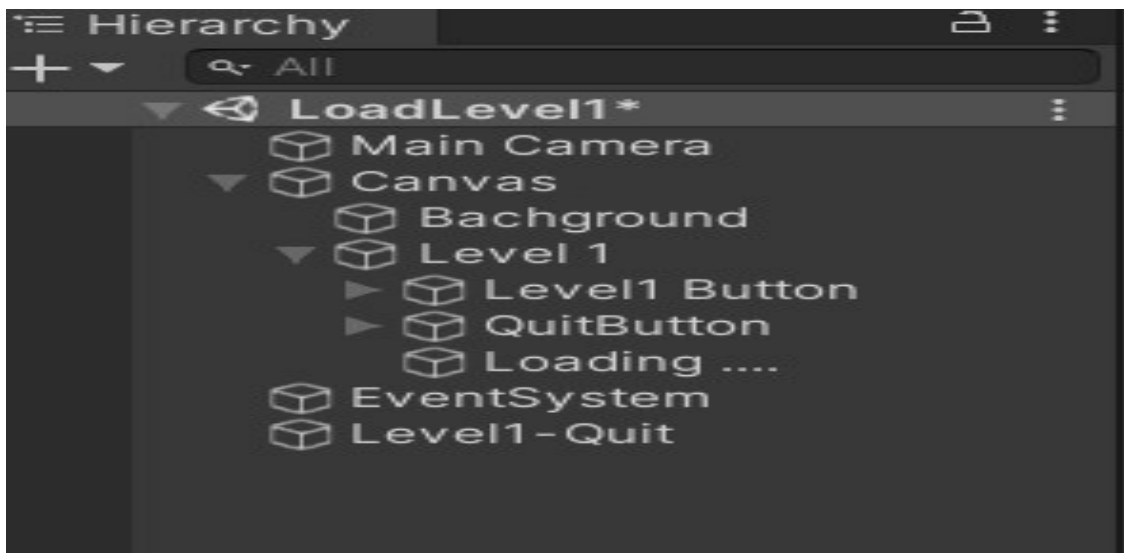
Τέλος, στο κεντρικό μενού υπάρχει το QuitButton που όταν το επιλέξει ο χρήστης το παιχνίδι κλείνει. Η συνάρτηση που χρησιμοποιείται για να πραγματοποιηθεί αυτή η ενέργεια είναι η Application.Quit().



Σχήμα 6.2 Menu Scene Hierarchy

6.2.3 LoadLevel Scene

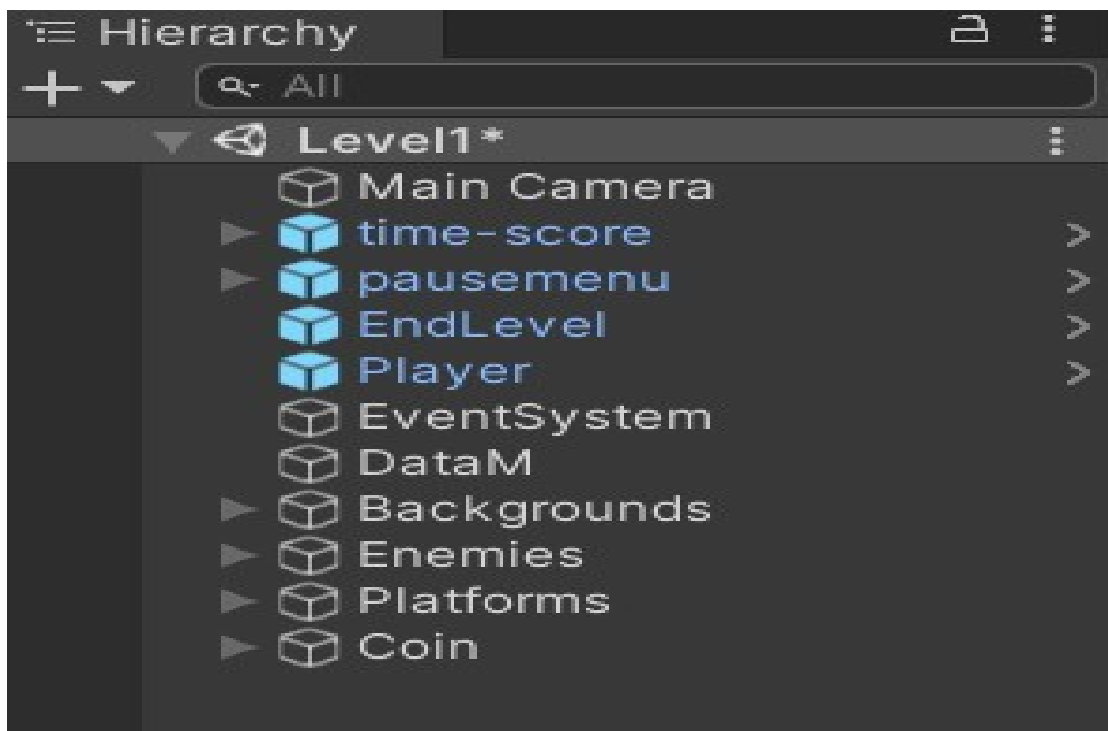
Αυτή η σκηνή εμφανίζεται στο παιχνίδι πάντα πριν από κάθε επίπεδο. Συγκεκριμένα η σκηνή LoadLevel1 (Σχήμα 6.3) περιέχει το κουμπί Level1 Button το οποίο πρέπει να επιλέξει ο χρήστης ώστε να ξεκινήσει η πρώτη πίστα. Ακόμα η σκηνή αυτή εμφανίζεται όταν ο παίκτης πεθάνει ή όταν έρθει σε επαφή με το αντικείμενο EndLevel σηματοδοτώντας την ολοκλήρωση του επιπέδου. Οι συναρτήσεις που χρησιμοποιήθηκαν είναι η SceneManager.LoadScene() και Application.Quit() και έγινε η χρήση της βιβλιοθήκης UnityEngine.SceneManagement.



Σχήμα 6.3 LoadLevel1 Scene Hierarchy

6.2.4 LevelScene

Η δράση του παιχνιδιού διαδραματίζεται στις σκηνές που αποτελούν τα επίπεδα του παιχνιδιού. Μέσα σε αυτές βρίσκονται και αλληλεπιδρούν όλα τα αντικείμενα. Συγκεκριμένα, τα αντικείμενα του πρώτου επιπέδου αναφέρονται στην Σχήμα 6.4.



Σχήμα 6.4 Level1Scene Hierarchy

Main Camera:

Η κάμερα βρίσκει και ακολουθεί το αντικείμενο με tag "Player" μέσα στην πίστα. Μέσα στον κώδικα έχουν οριστεί public μεταβλητές που ορίζουν την ελάχιστη και τη μέγιστη τιμή που μπορεί να ακολουθήσει στους άξονες x,y η κάμερα τον παίκτη μέσα στην πίστα. Έτσι δίνεται μεγαλύτερος έλεγχος στα όρια του επιπέδου.

Player:

Η κίνηση του παίκτη στους άξονες πραγματοποιείται με την συνάρτηση `Input.GetAxis()`. Ο παίκτης αποτελείται από τρία animation που αναπαριστούν τις τρεις καταστάσεις που έχει καθόλου τη διάρκεια του παιχνιδιού. Όταν ξεκινάει το παιχνίδι ο παίκτης βρίσκεται σε idle κατάσταση ενώ όταν τρέχει ή αναπηδά η κατάσταση του αλλάζει σε run και jump αντίστοιχα. Η αρμονική εναλλαγή των animation πραγματοποιείται με την βοήθεια του animator και της συνάρτησης `GetComponent<Animator>().SetBool()`.

Ο παίκτης έχει δύο βασικές ιδιότητες (Σχήμα 6.5) ώστε να μπορεί να αλληλεπιδρά όσο πιο ρεαλιστικά γίνεται με τα υπόλοιπα αντικείμενα. Η ιδιότητα `RigidBody 2D` προσφέρει την μηχανή φυσικής σε ένα αντικείμενο και είναι αυτή που του επιτρέπει στον παίκτη να μπορεί να κινείται στους άξονες x, y. Ταυτόχρονα η `BoxCollider 2D` χρησιμοποιεί ένα αόρατο σχήμα με σκοπό τον χειρισμό φυσικών συγκρούσεων μεταξύ αντικειμένων. Ένα collider δεν χρειάζεται να έχει ακριβώς το ίδιο σχήμα με τα αντικείμενα για τα οποία θα χρησιμοποιηθεί.

Με την ιδιότητα `BoxCollider 2D`, λοιπόν ο παίκτης έχει την ικανότητα να πεθαίνει όταν έρχεται σε επαφή με κάποιον εχθρό, να συλλέγει νομίσματα και να μεταφέρεται σε επόμενη πίστα. Επιπλέον ο παίκτης αποκτά πόντους όταν συλλέγει νομίσματα και σκοτώνει τους εχθρούς. Όλες αυτές οι λειτουργίες γίνονται δυνατές με τη χρήση του `trig.gameObject.name == ""` και της συνάρτησης `Destroy(trig.gameObject)`. Η τελευταία χρησιμοποιείται συγκεκριμένα όταν ο εχθρός σκοτώνει τον παίκτη ή όταν ο παίκτης συλλέγει νομίσματα.



Σχήμα 6.5 Ιδιότητες αντικειμένου Player

Canvas:

Στον καμβά αναγράφεται το χρονόμετρο της πίστας και οι συνολικοί πόντοι του παιχνιδιού. Η παρουσία ενός χρονομέτρου μέσα στο παιχνίδι υποδεικνύει το χρόνο που απομένει. Με την έναρξη του παιχνιδιού, ξεκινά η αντίστροφη μέτρηση και όταν ο χρόνος λήξει, τότε το παιχνίδι τελειώνει αυτόματα. Όταν αρχίζει το παιχνίδι το σκορ είναι μηδέν και καθώς ο παίκτης σκοτώνει ή συλλέγει νομίσματα τότε το σκορ αυξάνεται κατά δέκα ή δεκαπέντε πόντους αντίστοιχα. Στον κώδικα αυτών των λειτουργιών έγινε χρήση της βιβλιοθήκης `UnityEngine.UI`.

PauseMenu:

Ένα ακόμα σημαντικό interface είναι το `PauseMenu`. Μέσα από αυτό, ο παίκτης μπορεί όπως είναι προφανές να σταματήσει το παιχνίδι όποια στιγμή θέλει, να κάνει επαναφορά σε αυτό όταν το θελήσει, και να κάνει έξοδο από αυτό. Ο κώδικας που χρειάστηκε για αυτές τις λειτουργίες είναι αυτός του `Script PauseMenu`. Σε αυτόν υπάρχουν τρεις συναρτήσεις που υποστηρίζουν τις λειτουργίες που αναφέρθηκαν. Η `Resume()` από την οποία απενεργοποιείται το Interface της παύσης και γίνεται ψευδές το `Boolean pause`. Η `Pause()` που έχει την ίδια λειτουργία, κάνοντας όμως τις παραπάνω τιμές αληθείς, και η `Quit()` που πραγματοποιεί την έξοδο του παιχνιδιού. Οι συναρτήσεις έχουν ρυθμιστεί να καλούνται μέσα από τα `Components` του Unity όταν γίνεται κλικ στο παραπάνω Interface της παύσης. Τέλος η συνάρτηση `Update()`, χρησιμοποιώντας το `Boolean pause` σε μια δομή ελέγχου, όταν το πλήκτρο `escape` πατηθεί, αν ο χρήστης παίζει το παιχνίδι εμφανίζεται το μενού παύσης, ενώ αν βρίσκεται ήδη εκεί γίνεται επαναφορά στο παιχνίδι.

Platform:

Κάθε παιχνίδι χρειάζεται έδαφος ώστε να μπορούν να σταθούν όλα τα αντικείμενα και οι χαρακτήρες πάνω στο χώρο. Πολλά δάπεδα μαζί ορίζουν το αντικείμενο platforms το οποίο έχει ένα component RigidBody 2D έτσι ώστε να είναι συμπαγές και να μην πέφτουν τα αντικείμενα και οι χαρακτήρες στο κενό.

Background:

Το αντικείμενο background περιέχει όλα τα αντικείμενα που αποτελούν το περιβάλλον του παιχνιδιού, όπως το φόντο, τα σύννεφα, τα βουνά και το νερό.

Coins:

Βρίσκονται σκορπισμένα μέσα την πίστα. Όταν ο παίκτης έρθει σε επαφή μαζί τους μαζεύει πόντους και τα αντικείμενα εξαφανίζονται/καταστρέφονται από την οθόνη. Για αυτή την λειτουργία χρησιμοποιήθηκε η συνάρτηση Destroy(trig.gameObject) και ενεργοποιήθηκε η ιδιότητα trigger στο αντικείμενο.

Enemy:

Περιέχει όλους τους εχθρούς της πίστας. Οι εχθροί μπορούν να κινηθούν παράλληλα και στους δύο άξονες. Επιπλέον υπάρχουν εμπόδια που συμπεριλαμβάνονται στην ίδια κατηγορία. Όταν παίκτης έρθει σε επαφή με κάποιο από αυτά οδηγείται στον θάνατο και η πίστα ξεκινάει από την αρχή.

6.2.5 EndScene

Μετά την ολοκλήρωση της τελευταίας πίστας, το παιχνίδι οδηγεί στον χρήστη στην τελευταία σκηνή (Σχήμα 6.6). Εκεί υπάρχει ένα ευχαριστήριο μήνυμα ως προς το χρήστη και ένα κουμπί Exit. Το κουμπί τερματίζει την εκτέλεση του παιχνιδιού με τη βοήθεια της συνάρτησης Application.Quit().



Σχήμα 6.6EndScene

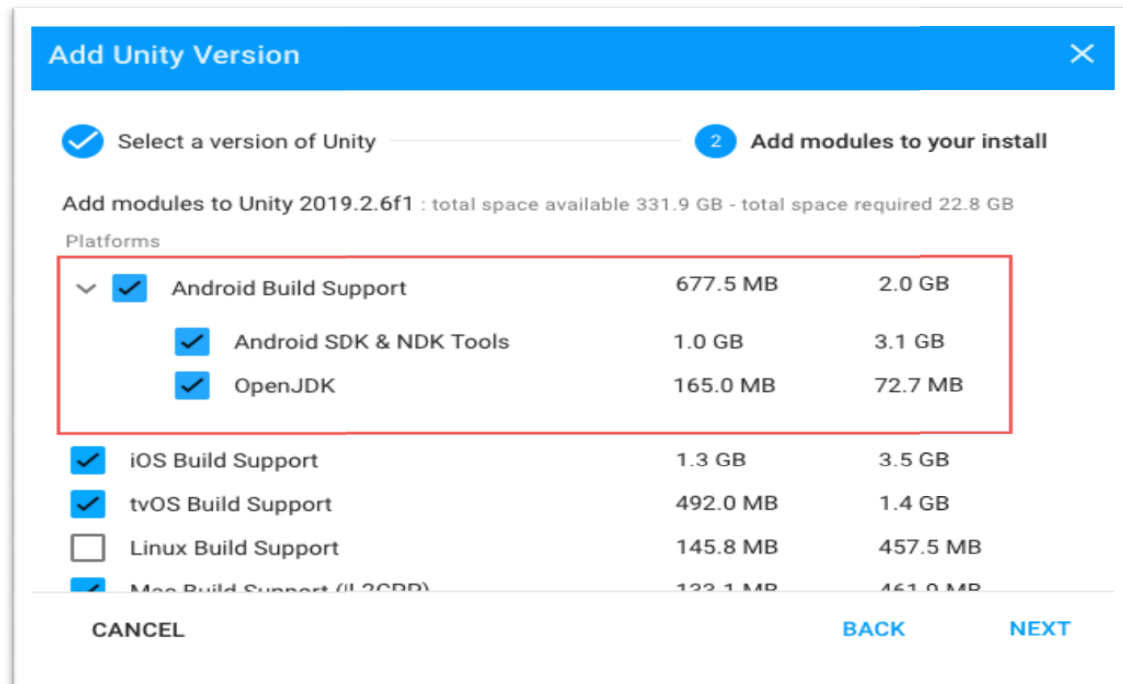
6.3 Εξαγωγή Παιχνιδιού

Το Unity δίνει την δυνατότητα εξαγωγής του παιχνιδιού σε διάφορες πλατφόρμες μέσα από το παράθυρο Build Settings. Η μεταφορά γίνεται εύκολα, γρήγορα και χωρίς πολλές αλλαγές και ρυθμίσεις.

Το παιχνίδι που αναπτύχθηκε στα πλαίσια της συγκεκριμένης πτυχιακής εργασίας υλοποιήθηκε σε Windows και σε Android. Για την εξαγωγή μιας εφαρμογής από το Unity σε συσκευές που υποστηρίζουν λογισμικό Android προαπαιτείται η εγκατάσταση του προγράμματος Java SE Development Kit και του Android SDK. Το μόνο που χρειάζεται είναι κατά την εγκατάσταση του Unity να επιλεγεί η επιλογή AndroidBuildSupport (Σχήμα 6.7). Αν δεν προεπιλεγεί από την αρχή και ο χρήστης σε δεύτερο χρόνο εγκαταστήσει τα παραπάνω προγράμματα, τότε πιθανότατα η εξαγωγή του παιχνιδιού να μην είναι επιτυχής καθώς τα προγράμματα μπορεί να μην είναι συμβατά με την συγκεκριμένη έκδοση Unity που χρησιμοποιείται.

Αφότου επιβεβαιωθεί η εγκατάσταση του Android Support, μέσα από την καρτέλα File και την επιλογή Build Settings πρέπει να γίνει η αλλαγή της πλατφόρμας από PC σε Android. Έπειτα πρέπει να προστεθούν όλες οι σκηνές που απαρτίζουν το παιχνίδι στο παράθυρο Scenes In Building ώστε να δοθεί δομή στο παιχνίδι. Στη συνέχεια, μέσα από τη καρτέλα Edit – Preferences – External Tools πρέπει να οριστεί το path των SDK, JDK και NDK. Συνήθως αυτό γίνεται αυτόματα με την εγκατάσταση του Android Build Support. Τέλος μέσα από το File – Build Settings – Player Settings μπορεί να οριστεί το όνομα του παιχνιδιού και να γίνουν διάφορες ρυθμίσεις ως προς την εμφάνιση του παιχνιδιού. Η εξαγωγή του παιχνιδιού ολοκληρώνεται με το κουμπί Build. Το αρχείο που προκύπτει είναι της μορφής .apk και είναι το παιχνίδι στη τελική του μορφή.

Οι Android συσκευές δεν επιτρέπουν την εγκατάσταση εφαρμογών που δεν προέρχονται από το Play Store για λόγους ασφάλειας. Για να εγκατασταθεί λοιπόν η εφαρμογή θα πρέπει να ενεργοποιηθεί η επιλογή εγκατάστασης εφαρμογών που δεν προέρχονται από το Google Market. Την παραπάνω επιλογή την παρέχει εύκολα η κινητή συσκευή κατά την εγκατάσταση της εφαρμογής.



Σχήμα 6.7 Android Build Support

Κεφάλαιο 7 – Συμπεράσματα

Ανακεφαλαιώνοντας, στο δεύτερο κεφάλαιο παρουσιάστηκαν τα πρώτα βήματα της βιομηχανίας των παιχνιδιών. Μέσα από την ιστορική ανάδρομή που έγινε, τονίστηκαν διάφορα σημαντικά σημεία της ιστορίας, όπως το 1952 όπου δημιουργήθηκε το πρώτο βιντεοπαιχνίδι και φυσικά η χρυσή εποχή των Arcade την δεκαετία 1980. Στη συνέχεια αναλύθηκαν διάφορα είδη παιχνιδιών με βάση την εμπειρία που προσφέρουν στον χρήστη. Στο τρίτο κεφάλαιο δόθηκε ιδιαίτερη σημασία στα γραφικά και τα είδη υπάρχουν. Επιπλέον παρουσιάστηκαν γνωστές μηχανές ανάπτυξης γραφικών και μηχανές ανάπτυξης παιχνιδιών. Το τέταρτο κεφάλαιο αφιερώθηκε στη Unity ξεκινώντας με μία μικρή ιστορική αναδρομή. Αναλύθηκαν οι λόγοι για τους οποίους επιλέχθηκε η Unity και τις δυνατότητες που προσφέρει σε σχέση με τις υπόλοιπες μηχανές. Ακόμη παρουσιάστηκε εκτενώς το περιβάλλον εργασίας, τονίζοντας τα διάφορα παράθυρα και menu που προσφέρει. Στο πέμπτο κεφάλαιο γίνεται αναφορά στην γλώσσα προγραμματισμού C#. Συγκεκριμένα αναφέρονται μερικά ιστορικά στοιχεία της καθώς και στοιχεία που αφορούν την εξέλιξη της. Επιπλέον αναφέρεται ο λόγος που είναι συμβατή με την Unity και περιγράφονται διάφορες συναρτήσεις και κλάσεις που χρησιμοποιεί η Unity. Στο έκτο και τελευταίο κεφάλαιο παρουσιάζεται η δημιουργία του παιχνιδιού και περιγράφεται ο τρόπος εξαγωγής του παιχνιδιού σε κινητές android συσκευές.

Η υλοποίηση της παρούσας πτυχιακής εργασίας, ήταν η πρώτη προσπάθεια δημιουργίας ενός παιχνιδιού δύο διαστάσεων χρησιμοποιώντας την μηχανή ανάπτυξης Unity και φυσικά η πρώτη επαφή με το εργαλείο. Μέσα από αυτή την εμπειρία που αποκτήθηκε διαπιστώθηκαν, πέρα από την θεωρητική ανάλυση και στην πράξη, τα χαρακτηριστικά και οι ευκολίες που προσφέρει η Unity για κάποιον που βρίσκεται σε ένα πρώιμο στάδιο πάνω στον τομέα της ανάπτυξης βιντεοπαιχνιδιών. Αυτές οι ευκολίες ήταν κυρίως η πληθώρα παραδειγμάτων και tutorials που υπάρχουν, η ευχρηστία του User Interface και η ύπαρξη γλώσσας αντικειμενοστραφούς προγραμματισμού στον τομέα της ανάπτυξης. Η εφαρμογή μπορεί να βελτιωθεί μελλοντικά έτσι ώστε να υπάρχει συνεχές ενδιαφέρον και ανταγωνισμός.

Μπορούν να σχεδιαστούν νέες πίστες με μεγαλύτερο βαθμό δυσκολίας και εχθροί που θα καταδιώκουν τον παίκτη με διαφορετική ταχύτητα ο καθένας. Γρίφοι θα υπάρχουν σε διάφορα σημεία του παιχνιδιού και ο παίκτης θα καλείται να τους λύσει ώστε να κερδίσει πόντους εμπειρίας τους οποίους θα μπορεί να χρησιμοποιήσει για να αναβαθμίσει τα όπλα του. Ακόμα θα μπορούσε να προστεθεί ήχος καθ' ολη την διάρκεια του παιχνιδιού καθώς και εφέ για μεγαλύτερο ρεαλισμό. Τέλος, μέσω του Google Play θα μπορούσε να γίνει προώθηση του παιχνιδιού ή και πώληση ώστε να υπάρξει κέρδος.

Η φύση της εργασίας προσφέρει σημαντική εμπειρία και έδωσε τη δυνατότητα να εφαρμοστούν οι γνώσεις που αποκτήθηκαν κατά τη διάρκεια των σπουδών.

Βιβλιογραφία

- [1] En.wikipedia.org. 2021. *History of video games - Wikipedia*. [online] Available at: <https://en.wikipedia.org/wiki/History_of_video_games> [Accessed 2 August 2021].
- [2] KingsIsle Blog. n.d. The Early History of Video Games. [online] Available at: <<https://kingsisleblog.com/2016/03/08/the-early-history-of-video-games/>> [Accessed 2 August 2021].
- [3] The Strong Museum of Play. n.d. Video Games Archive - The Strong Museum of Play. [online] Available at: <https://www.museumofplay.org/video_games/> [Accessed 2 August 2021].
- [4] Focus on the Family. 2021. *Types of Video Games*. [online] Available at: <<https://www.focusonthefamily.com/parenting/types-of-video-games/>> [Accessed 2 August 2021].
- [5] Posts, F., 2021, B., teens, U., John!, I., two?, S., math), 2., Python, 1., 8 easy, e., needs, W. and success, C., 2021. *Ultimate List of Different Types of Video Games | 49 Genres & Subcategories*. [online] iD Tech. Available at: <<https://www.idtech.com/blog/different-types-of-video-game-genres>> [Accessed 2 August 2021].
- [6] En.wikipedia.org. 2021. *List of types of games - Wikipedia*. [online] Available at: <https://en.wikipedia.org/wiki/List_of_types_of_games> [Accessed 2 August 2021].
- [7] Gamedesigning.org. 2021. *Essential Software for all Aspiring Video Game Developers (Reviewed)*. [online] Available at: <<https://www.gamedesigning.org/career/software/>> [Accessed 2 August 2021].
- [8] The Verge. 2021. *11 tools to get you started making video games*. [online] Available at: <<https://www.theverge.com/2020/4/14/21219609/video-game-tools-editor-developer-make-price-free-programming>> [Accessed 2 August 2021].
- [9] "Tools and Software Used for Designing and Developing Video Games | NEIT", NEIT Public Website, 2021. [Online]. Available: <https://www.neit.edu/blog/game-development-software-tools>. [Accessed: 03- Aug- 2021]
- [10] Gamedesigning.org. 2021. 10 Game Development Tools: Free Gear for Better Game Making. [online] Available at: <<https://www.gamedesigning.org/gaming/game-development-tools/>> [Accessed 3 August 2021].
- [11] "8 Free Game Development Software Tools to Make Your Own Games", MUO, 2021. [Online]. Available: <https://www.makeuseof.com/tag/five-free-game-development-tools-make-your-own-games/>. [Accessed: 03- Aug- 2021]
- [12] "The 10 Video Game Engines Pros Use (With Infographic)", Gamedesigning.org, 2021. [Online]. Available: <https://www.gamedesigning.org/career/video-game-engines/>. [Accessed: 03- Aug- 2021]
- [13] "5 best software for game animation to use right now", Windows Report - Error-free Tech Life, 2021. [Online]. Available: <https://windowsreport.com/game-animation-software/>. [Accessed: 03- Aug- 2021]
- [14] "12 Essential Graphic Design Software For Creatives (Recommended by Digital Artists)", Gamedesigning.org, 2021. [Online]. Available: <https://www.gamedesigning.org/graphic/design-software/>. [Accessed: 03- Aug- 2021]

- [15] "ΚΑΤΗΓΟΡΙΕΣ", Video Games, 2021. [Online]. Available: <https://videogameeffects.weebly.com/kappaalphatauetagammaomicronrhoiotaepsilonsigma.html>. [Accessed: 03- Aug- 2021]
- [16] Johantsili1981.wixsite.com, 2021. [Online]. Available: <https://johantsili1981.wixsite.com/graphics2015/2d3d>. [Accessed: 03- Aug- 2021]
- [17] "Computer graphics - Wikipedia", En.wikipedia.org, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Computer_graphics. [Accessed: 03- Aug- 2021]
- [18] "5 online εργαλεία για να δημιουργήσετε απίστευτα γραφικά!", Skillbox.gr, 2021. [Online]. Available: <https://www.skillbox.gr/news/view/Nea/7795/5-online-ergaleia-gia-na-dimioyrgisete-apisteyta-grafika>. [Accessed: 03- Aug- 2021]
- [19] "Δωρεάν Online Εργαλεία Σχεδιασμού Γραφικών – Πλατφόρμες ηλεκτρονικού εμπορίου", Ecommerce Platforms, 2021. [Online]. Available: <https://ecommerce-platforms.com/el/articles/free-online-graphic-design-tools>. [Accessed: 03- Aug- 2021]
- [20] "Top 10 Graphic Design Software for Game Devs - Buildbox | Game Maker | Video Game Software", Buildbox | Game Maker | Video Game Software, 2021. [Online]. Available: <https://www.buildbox.com/top-10-graphic-design-software/>. [Accessed: 03- Aug- 2021]
- [21] "The best graphic design software in 2021 – including paid-for and free options", Creative Bloq, 2021. [Online]. Available: <https://www.creativebloq.com/graphic-design/free-graphic-design-software-8134039>. [Accessed: 03- Aug- 2021]
- [22] "Unity Technologies - Wikipedia", En.wikipedia.org, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Unity_Technologies. [Accessed: 03- Aug- 2021]
- [23] "History of Unity Technologies in Timeline - Popular Timelines", populartimelines.com, 2021. [Online]. Available: <https://populartimelines.com/timeline/Unity-Technologies>. [Accessed: 03- Aug- 2021]
- [24] U. Technologies, "Unity - Manual: The Inspector Window", Docs.huihoo.com, 2021. [Online]. Available: <https://docs.huihoo.com/unity/5.6/Documentation/Manual/UsingTheInspector.html>. [Accessed: 03- Aug- 2021]
- [25] "Getting to Know the Unity Editor | Introduction to Unity | InformIT", InformIT.com, 2021. [Online]. Available: <https://www.informit.com/articles/article.aspx?p=2464010&seqNum=2>. [Accessed: 03- Aug- 2021]
- [26] "Learning Transform tools – Unity Tutorial - Gyanendu Shekhar's Blog", Gyanendu Shekhar's Blog, 2021. [Online]. Available: <http://gyanendushekhar.com/2017/08/20/learning-transform-tools-unity-tutorial/>. [Accessed: 03- Aug- 2021]
- [27] "Guide to Extending Unity Editor's Menus", Medium, 2021. [Online]. Available: <https://blog.redbluegames.com/guide-to-extending-unity-editors-menus-b2de47a746db>. [Accessed: 03- Aug- 2021]
- [28] "JavaScript - Wikipedia", En.wikipedia.org, 2021. [Online]. Available: <https://en.wikipedia.org/wiki/JavaScript>. [Accessed: 03- Aug- 2021]
- [29] "C Sharp (programming language) - Wikipedia", En.wikipedia.org, 2021. [Online]. Available: [https://en.wikipedia.org/wiki/C_Sharp_\(programming_language\)](https://en.wikipedia.org/wiki/C_Sharp_(programming_language)). [Accessed: 03- Aug- 2021]
- [30] "MonoDevelop - Wikipedia", En.wikipedia.org, 2021. [Online]. Available: <https://en.wikipedia.org/wiki/MonoDevelop>. [Accessed: 03- Aug- 2021]

- [31] "Boo (programming language) - Wikipedia", En.wikipedia.org. [Online]. Available: [https://en.wikipedia.org/wiki/Boo_\(programming_language\)](https://en.wikipedia.org/wiki/Boo_(programming_language)). [Accessed: 03- Aug- 2021]
- [32] U. Technologies, "Unity - Scripting API: MonoBehaviour", Docs.unity3d.com, 2021. [Online]. Available: <https://docs.unity3d.com/ScriptReference/MonoBehaviour.html>. [Accessed: 03- Aug- 2021]
- [33] U. Technologies, "Unity - Scripting API: Collider", Docs.unity3d.com, 2021. [Online]. Available: <https://docs.unity3d.com/ScriptReference/Collider.html>. [Accessed: 03- Aug- 2021]
- [34] ".NET Framework - Wikipedia", En.wikipedia.org, 2021. [Online]. Available: https://en.wikipedia.org/wiki/.NET_Framework. [Accessed: 03- Aug- 2021]
- [35] "What is .NET Framework? A software development framework.", Microsoft, 2021. [Online]. Available: <https://dotnet.microsoft.com/learn/dotnet/what-is-dotnet-framework>. [Accessed: 03- Aug- 2021]
- [36] "Java to C#—C# to Java", Medium, 2021. [Online]. Available: <https://betterprogramming.pub/java-to-c-c-to-java-f766c9f659c4>. [Accessed: 03- Aug- 2021]
- [37] "Comparison of C Sharp and Java - Wikipedia", En.wikipedia.org, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Comparison_of_C_Sharp_and_Java. [Accessed: 03- Aug- 2021]
- [38] "Programming language - Wikipedia", En.wikipedia.org, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Programming_language. [Accessed: 03- Aug- 2021]
- [39] En.wikipedia.org. 2021. Game engine - Wikipedia. [online] Available at: https://en.wikipedia.org/wiki/Game_engine [Accessed 3 August 2021].
- [40] "Lets Make A Game Together", Youtube.com, 2021. [Online]. Available: https://www.youtube.com/channel/UCHhzc_ukN0BxB2aVQBKt89Q. [Accessed: 03- Aug- 2021]
- [41] "The Game Dev_", Youtube.com, 2021. [Online]. Available: <https://www.youtube.com/channel/UCHfilkvh6OafPoMaR6MbGLw>. [Accessed: 03- Aug- 2021]
- [42] "Brackeys", Youtube.com, 2021. [Online]. Available: https://www.youtube.com/channel/UCYbK_tjZ2OrIZFBvU6CCMiA. [Accessed: 03- Aug- 2021]
- [43] "Alexander Zotov", Youtube.com, 2021. [Online]. Available: <https://www.youtube.com/channel/UCYgUFRfYfr5IyL8HfTTiWhA>. [Accessed: 03- Aug- 2021]