

ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Head-Tracked, Stereo Ski Simulation Implemented on a High-End Head Mounted Display



Ιορδάνης Φυτάνογλου

ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ

Αναπ. Καθ. Αικατερίνη Μανιά (Επιβλέπουσα)

Αναπ. Καθ. Μιχαήλ Γ. Λαγουδάκης

Αναπ. Καθ. Αντώνιος Δεληγιαννάκης

Χανία, Δεκέμβριος 2015

Στην οικογένειά μου

Περίληψη

Στην παρούσα διπλωματική εργασία, παρουσιάζεται η σχεδίαση και η υλοποίηση ενός παιχνιδιού εικονικής πραγματικότητας για το Oculus Rift DK2. Το παιχνίδι ανήκει στην κατηγορία προσομοίωσης/δράσης και δημιουργήθηκε κάνοντας χρήση της μηχανής παιχνιδιών Unity 3D. Πιο συγκεκριμένα, ο χρήστης εμβυθίζεται στο εικονικό περιβάλλον, καλούμενος να περάσει μια σειρά δοκιμασιών, με στόχο να φτάσει στον τερματισμό. Η αλληλεπίδραση με το εικονικό περιβάλλον γίνεται αποκλειστικά με το Oculus Rift, δίχως να απαιτείται η χρήση των κλασικών συσκευών εισόδου, όπως ποντίκι και πληκτρολόγιο. Αρχικά λοιπόν, αφού σχεδιάστηκε ένα απλό και εύχρηστο User Interface, μοντελοποιήθηκε η φυσική του σκιέρ, τα κινούμενα σχέδια, διάφορα εφέ εντυπωσιασμού, στερεοφωνικοί ήχοι, καθώς και η συμπεριφορά των αντιπάλων, η οποία είναι βασισμένη σε αρχές τεχνητής νοημοσύνης. Τέλος, η εφαρμογή αξιολογήθηκε μέσω των ερωτηματολογίων Immersive Experience Questionnaire και Simulator Sickness Questionnaire, από χρήστες και των δύο φύλων, διαφορετικής ηλικίας και επαγγελματικής κατάρτισης. Σκοπός ήταν να υπολογιστεί ο βαθμός εμβύθισης, πιθανά συμπτώματα προσομοίωσης, καθώς και να εξαχθούν συνολικά συμπεράσματα για το User Experience.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω την επιβλέπουσα καθηγήτρια κ.Κατερίνα Μανιά για την επίβλεψη και για την καθοδήγησή της στην εκπόνηση αυτής της διπλωματικής εργασίας, καθώς και για το ότι μου έδωσε την ευκαιρία να ασχοληθώ σε μεγάλο βάθος με τον τομέα της εικονικής πραγματικότητας και να υλοποιήσω την ιδέα μου.

Στη συνέχεια θα ήθελα να ευχαριστήσω θερμά τους καθηγητές που απαρτίζουν την επιτροπή, τον κ.Λαγουδάκη Μιχάλη και τον κ.Δεληγιαννάκη Αντώνιο για το χρόνο που αφιέρωσαν στη μελέτη του κειμένου της εργασίας μου. Ιδιαίτερα, ευχαριστώ τον κύριο Λαγουδάκη για την πολύτιμη βοήθειά του σε θέματα τεχνητής νοημοσύνης αναφορικά με το τελευταίο κομμάτι της πτυχιακής μου εργασίας.

Επίσης θα ήθελα να ευχαριστήσω τον Γεώργιο Αλέξανδρο Κουλιέρη για την πολύτιμη βοήθειά του καθόλη τη διάρκεια της πτυχιακής αυτής εργασίας καθώς και όλα τα παιδιά που αφιέρωσαν χρόνο να δοκιμάσουν το παιχνίδι και να συμπληρώσουν τα ερωτηματολόγια.

Τέλος ένα μεγάλο ευχαριστώ στην οικογένειά μου και στους φίλους μου που με στήριξαν όλα αυτά τα χρόνια για την ολοκλήρωση των σπουδών μου.

Περιεχόμενα

1	Εισαγωγή	12
1.1	Εισαγωγή	12
1.2	Περιγραφή Παιχνιδιού	12
1.3	Πλατφόρμα	13
1.4	Βασικά Τεχνικά Χαρακτηριστικά	14
1.4.1	Σύστημα Γραφικών	14
1.4.2	Σύστημα Φυσικής	14
1.4.3	Σύστημα Ήχου	14
1.4.4	Σύστημα Φωτισμού	14
1.4.5	Σύστημα Γραφικής Διεπαφής Χρήστη	15
1.4.6	Σύστημα Τεχνητής Νοημοσύνης	15
1.5	Δομή της εργασίας	15
2	Επισκόπηση Σχετικής Έρευνας	17
2.1	Εισαγωγή	17
2.2	Ιστορική Αναδρομή Βιντεοπαιχνιδιών	17
2.2.1	Πλατφόρμες Arcade (Arcade Platforms)	18
2.2.2	Κονσόλες Βιντεοπαιχνιδιών (Video Game Consoles)	18
2.2.3	Προσωπικοί Υπολογιστές(PCs)	19
2.2.4	Συσκευές Χειρός (Hand-held Devices)	20
2.3	Κατηγορίες Βιντεοπαιχνιδιών	20
2.3.1	Δράσης (Action Games)	21
2.3.2	Arcade Games	22
2.3.3	Πάλης (Fighting Games)	22
2.3.4	Γενικής Διασκέδασης (General Entertainment Games)	23
2.3.5	Μουσικής & Ρυθμού (Music & Rhythm Games)	23
2.3.6	Αφήγησης (Narrative Games)	23
2.3.7	Αγώνων Οδήγησης (Driving/Racing Games)	24

2.3.8	Ρόλων (RPG Games)	24
2.3.9	Βολών (Shooter Games)	25
2.3.10	Ικανοτήτων & Τύχης (Skill & Chance)	25
2.3.11	Προσομοίωσης (Simulation Games)	26
2.3.12	Αθλητισμού (Sports Games)	27
2.3.13	Στρατηγικής (Strategy Games)	27
2.4	Αγωγός Γραφικών Graphics Pipeline	29
2.5	Τρισδιάστατες Βιβλιοθήκες Γραφικών 3D APIs	29
2.5.1	Direct3D	30
2.5.2	OpenGL	30
2.6	Επιλογή της Unity 3D	31
2.7	Επικοινωνία Ανθρώπου Υπολογιστή (HCI for Games)	33
2.8	Συνέχεια Πραγματικότητας-Εικονικότητας (Reality-Virtuality Continuum)	33
2.8.1	Επαυξημένη Πραγματικότητα (Augmented Reality)	34
2.8.2	Επαυξημένη Εικονικότητα (Augmented Virtuality)	34
2.8.3	Μεικτή Πραγματικότητα (Mixed Reality)	35
2.8.4	Εικονικά Περιβάλλοντα (Virtual Environments)	35
2.8.5	Head Mounted Displays	36
2.8.6	Head Tracking	38
3	Τεχνολογική Βάση	39
3.1	Εισαγωγή	39
3.2	3DS	39
3.3	Unity 3D	40
3.4	Δομή & Αρχιτεκτονική της Unity 3D	40
3.4.1	Συντεταγμένες (Coordinates)	40
3.4.2	Τοπικό Σύστημα Συντεταγμένων (Local Space vs World Space)	41
3.4.3	Διανύσματα (Vectors)	41
3.4.4	Κάμερες (Cameras)	42
3.4.5	Πολύγωνα, Αχμές, Κορυφές & Πλέγματα (Polygons, Edges, Vertices & Meshes)	42
3.4.6	Μετασχηματισμός (Transform)	42
3.4.7	Υλικά, Υφές & Σκιαστές (Materials, Textures & Shaders)	43
3.4.8	Σωματίδια (Particle System)	43
3.4.9	Φυσική (Rigidbody Physics)	43
3.4.10	Ανίχνευση Σύγκρουσης (Collision Detection)	44
3.4.11	Φωτισμός (Lighting)	44

3.4.12	Ήχος (Audio)	46
3.4.13	Διεπαφή Χρήστη (User Interface)	46
3.4.14	Κινούμενα Σχέδια (Animations)	46
3.4.15	Πλοήγηση & Εύρεση Καλύτερου Μονοπατιού (Navigation & Pathfinding)	47
3.5	Βασικές Έννοιες της Unity 3D	47
3.5.1	(Assets)	48
3.5.2	Σκηνές (Scenes)	48
3.5.3	Αντικείμενα (GameObjects)	48
3.5.4	Συστατικά (Components)	48
3.5.5	Scripts	49
3.5.6	Prefabs	50
3.6	Oculus Rift DK2	50
3.6.1	Binocular Vision	51
3.6.2	Field of View & Scale	52
3.6.3	Rendering Techniques	53
3.6.4	Κίνηση (Motion)	54
3.6.5	Παρακολούθηση (Tracking)	54
3.6.6	Δυσφορία Προσομοίωσης (Simulator Sickness)	55
4	Σχεδιασμός Παιχνιδιού & Γραφικής Διεπαφής Χρήστη	57
4.1	Εισαγωγή	57
4.2	Σενάριο	57
4.3	Σχεδιασμός Παιχνιδιού	58
4.3.1	Σχεδιασμός Πίστας	58
4.3.2	Σχεδιασμός Χαρακτήρων	59
4.4	Γραφική Διεπαφή Χρήστη	62
4.4.1	Αρχικό μενού	63
4.4.2	Γραφικό περιβάλλον Παιχνιδιού	66
5	Υλοποίηση Παιχνιδιού	74
5.1	Εισαγωγή	74
5.2	Γραφική Διεπαφή Χρήστη	74
5.2.1	Αρχικό Μενού	75
5.2.2	Μενού Κυρίως Παιχνιδιού	78
5.3	Υλοποίηση Σκιέρ(Skier Physics)	79
5.3.1	Φυσική Σκιέρ (Skier Physics)	80
5.3.2	Ενσωμάτωση Oculus Rift DK2 & Χειρισμός Σκιέρ	81

5.4	Τεχνητή Νοημοσύνη	83
5.4.1	Αντίπαλοι Σκιέρ	85
5.4.2	Αντίπαλοι Λύκοι	86
5.5	Κινούμενα Σχέδια (Animations)	92
5.6	Σωματιδιακά τεχνάσματα εντυπωσιασμού (Particles Effects)	93
5.7	Ήχοι (Audio)	94
6	Αξιολόγηση Συστήματος	97
6.1	Εισαγωγή	97
6.2	Μέθοδος Αξιολόγησης	97
6.3	Στόχος & Αποτελέσματα	102
6.4	Συμπεράσματα	105
7	Αποτελέσματα & Μελλοντικές Επεκτάσεις	108
7.1	Εισαγωγή	108
7.2	Στόχος	108
7.3	Αποτελέσματα	109
7.4	Μελλοντικές Επεκτάσεις & Βελτιώσεις	109
7.5	Επίλογος	111
	Βιβλιογραφία	112

Κατάλογος Σχημάτων

2.1	Πλατφόρμες Arcade Αριστερά: Pac-Man, Κέντρο: Space Invaders, Δεξιά: Galaga	18
2.2	Πλατφόρμες Arcade Αριστερά: Brown Box (1967), Κέντρο: Magnavox Odyssey (1972), Δεξιά: Atari 2600 (1977)	19
2.3	Κονσόλες Βιντεοπαιχνιδιών Αριστερά: Nintendo Wii U (2012), Κέντρο: Sony PlayStation 4 (PS4) (2013), Δεξιά: Xbox One (2013)	19
2.4	Συσκευές χειρός Αριστερά: Game Boy Advance (2001), Κέντρο: Google Nexus 10 (2012), Δεξιά: iPhone5 (2012)	20
2.5	Παιχνίδια δράσης Αριστερά: Tomb Raider: Underworld (2008), Κέντρο: Lego Star Wars II: The Original Trilogy (2006), Δεξιά: God of War II (2007)	21
2.6	Παιχνίδια δράσης Αριστερά: Flow (arcade game Blockade 1976), Κέντρο: Pacman (1980), Δεξιά: Joust (1982)	22
2.7	Παιχνίδια πάλης Αριστερά: Dead or Alive 4 (2005), Κέντρο: Virtua Fighter 5 (2005), Δεξιά: WWE SmackDown vs. Raw 2007 (2006)	22
2.8	Παιχνίδια γενικής διασκέδασης Αριστερά: Bejeweled 2 (2004), Κέντρο: WarioWare: Smooth Moves(party game) (2006), Δεξιά: Bust-A-Move Bash! (2007)	23
2.9	Παιχνίδια μουσικής & ρυθμού Αριστερά: Guitar Hero II (2006), Κέντρο: Dance Dance Revolution Universe 3 (2008), Δεξιά: SingStar (2004)	23
2.10	Παιχνίδια αφήγησης Αριστερά: Condemned: Criminal Origins (2005), Κέντρο: Myst (1993), Δεξιά: King's Quest 5 (1990)	24
2.11	Παιχνίδια αγώνων οδήγησης Αριστερά: Need for Speed: Most Wanted (2005), Κέντρο: Project Gotham Racing 3 (2005), Δεξιά: Burnout Revenge (2005)	24
2.12	Παιχνίδια ρόλων Αριστερά: Final Fantasy XII (2006), Κέντρο: The Elder Scrolls IV: Oblivion (2006), Δεξιά: Untold Legends: Dark Kingdom (2006)	25
2.13	Παιχνίδια βολών Αριστερά: Call of Duty 2 (2005), Κέντρο: Half-Life 2 (2004), Δεξιά: Gears of War (2006)	25

2.14	Παιχνίδια ικανοτήτων & τύχης Αριστερά: UNO (2006), Κέντρο: World Series of Poker: Tournament of Champions (2006), Δεξιά: Catan (2008) . . .	26
2.15	Παιχνίδια προσομοίωσης Αριστερά: Flight Simulator X (2006), Κέντρο: The Sims 2 (2004), Δεξιά: SimCity 4 (2003)	27
2.16	Παιχνίδια αθλητισμού Αριστερά: Vancouver 2010 (2010), Κέντρο: Tiger Woods PGA Tour 14 (2013), Δεξιά: Football Manager 2015 (2014)	27
2.17	Παιχνίδια στρατηγικής Αριστερά: Warcraft III: Reign of Chaos (2002) Κέντρο: Command & Conquer: Red Alert 2 (2000) Δεξιά: Age of Empires III (2005)	28
2.18	Τυπική Μηχανή Γραφικών	32
2.19	R-V Continuum	34
3.1	Coordinate system	41
3.2	World Space vs Local Space	41
3.3	Materials, Textures and Shaders	43
3.4	Lighting	44
3.5	Αριστερά: Point Light Δεξιά: Spot Light	45
3.6	Directional Light	45
3.7	Area Light	45
3.8	Game Object-Components-Variables Overview	47
3.9	Απόσταση μεταξύ των καμερών (ICD)	52
4.1	Terrains Αριστερά: Junior Tour Terrain, Κέντρο: Slalom Terrain, Δεξιά: Large Hill Terrain	59
4.2	Σκηές παιχνιδιού Πάνω Αριστερά: Slalom, Πάνω δεξιά: Junior Tour, Κάτω αριστερά: Large Hill, Κάτω δεξιά: Wolves Slalom	60
4.3	Τρισδιάστατο μοντέλο σκιέρ	61
4.4	Τρισδιάστατα μοντέλα αντίπαλων σκιέρ	61
4.5	Πτώση σκιέρ (Ragdoll)	62
4.6	Τρισδιάστατο μοντέλο λύκου	62
4.7	Αρχικό μενού	63
4.8	Κυρίως μενού (Main Menu)	64
4.9	Μενού προπόνησης (Training Menu)	64
4.10	Μενού Δοκιμασιών (Challenges Menu), Πάνω Αριστερά: Δεν έχουμε επιλέξει τίποτα, Πάνω Δεξιά: Slalom Challenge, Κάτω Αριστερά: Junior Tour Challenge, Κάτω Δεξιά: Large Hill Challenge	65
4.11	Αριστερά: Τερματισμός Large Hill Challenge, Δεξιά: Τερματισμός 3 πρώτων δοκιμασιών και ξεκλείδωμα τελευταίας (Canadian Slope Challenge)	66

4.12 Πάνω Αριστερά:Slalom Loading Bar, Πάνω Δεξιά: Slalom Loading Bar, Κάτω Αριστερά:Large Hill Loading Bar, Κάτω Δεξιά: Wolves Slalom Loading Bar	67
4.13 Μενού οδηγίων	68
4.14 HUDs στο Slalom Challenge, Αριστερά: Gate counter, Δεξιά: Timer . . .	69
4.15 HUDs στο Wolves Slalom Challenge, Αριστερά: Energy, Δεξιά: Gate counter	70
4.16 Παράθυρο εκκίνησης στο Slalom Challenge	71
4.17 Παράθυρο τερματισμού πρώτου γύρου στο Large Hill Challenge	71
4.18 Παράθυρο Game Over που εμφανίζεται όταν οι λύκοι εντοπίσουν και σκοτώσουν τον σκιέρ (Wolves Slalom Challenge)	72
4.19 Προειδοποιητικό Μήνυμα Speed-Up	72
4.20 Μενού Παύσης (Pause Menu)	73
5.1 Προτεραιότητα παραθύρων	79
5.2 Wheel Collider	81
5.3 Μοντέλο σκιέρ με συγκρουστές	82
5.4 Παράμετροι NavMesh	84
5.5 NavMesh	84
5.6 Σύστημα διεπικοινωνίας αυτόνομων λύκων (A)	90
5.7 Σύστημα διεπικοινωνίας αυτόνομων λύκων (B)	91

Κεφάλαιο 1

Εισαγωγή

1.1 Εισαγωγή

Τα τελευταία χρόνια η τεχνολογία της εικονικής πραγματικότητας (Virtual Reality) έχει εξελιχθεί με ραγδαίους ρυθμούς κάνοντας επανάσταση. Το 2016 αναμένεται να κυκλοφορήσουν στην αγορά πληθώρα Head Mounted Displays και να γίνουν διαθέσιμα στο ευρύ κοινό. Κινητήριος δύναμη της ώθησης αυτής, αποτελεί το Oculus Rift το οποίο αυτή τη στιγμή είναι το πιο διαδεδομένο και δημοφιλές Head Mounted Display (HMD) στους κύκλους των προγραμματιστών με την εμπορική του έκδοση να αναμένεται να κυκλοφορήσει τους επόμενους μήνες. Στόχος, λοιπόν της παρούσας διπλωματικής εργασίας ήταν η υλοποίηση ενός διαδραστικού τρισδιάστατου παιχνιδιού που θα προσομοιώνει το άθλημα του σκι και στο οποίο θα μπορεί κανείς να χειριστεί τον παίκτη εξ' ολοκλήρου με το Oculus Rift DK2. Το εν λόγω παιχνίδι μπορεί να κατηγοριοποιηθεί στα παιχνίδια προσομοίωσης, καθώς επιδιώκει να εμβυθίσει τον χρήστη στον τρισδιάστατο χώρο και να τον κάνει να βιώσει την εμπειρία του σκι. Ο χρήστης χειρίζεται τις κινήσεις του παίκτη μέσω της κίνησης του κεφαλιού (Head Tracking) πράγμα το οποίο είναι πολύ κοντά στην πραγματικότητα. Επιπλέον το παιχνίδι περιλαμβάνει τεχνητή νοημοσύνη καθώς υπάρχουν αντίπαλοι σκιέρ και λύκοι οι οποίοι λειτουργούν αυτόνομα και συμπεριφέρονται έξυπνα ανάλογα με τις κινήσεις του παίκτη.

1.2 Περιγραφή Παιχνιδιού

Το παιχνίδι που παρουσιάζεται στην παρούσα διπλωματική εργασία είναι ένα παιχνίδι προσομοίωσης πρώτου προσώπου. Σκοπός ήταν η εκμετάλλευση του Oculus Rift μέσω του Head Tracking για την κατεύθυνση της κίνησης του παίκτη. Ο βασικός χαρακτήρας είναι ένας σκιέρ

με τον χρήστη να πρέπει να περάσει επιτυχώς τέσσερις δοκιμασίες ώστε να καταφέρει να τερματίσει το παιχνίδι. Οι τρεις πρώτες δοκιμασίες είναι ξεκλειδωτές και η τελευταία κλειδωμένη. Αφού τερματίσει τις τρεις πρώτες, ξεκλειδώνει και η τέταρτη. Αν επιτύχει και σε αυτήν, τότε έχει τερματίσει το παιχνίδι. Οι δοκιμασίες που πρέπει να τερματιστούν περιγράφονται συνοπτικά παρακάτω:

- **Slalom:** Στη δοκιμασία αυτή σημαντικό ρόλο παίζουν τα αντανακλαστικά του παίχτη και η ταχύτητά του καθώς σκοπός είναι να τερματίσει την πίστα σε χρόνο λιγότερο από αυτόν των αντιπάλων του και να περάσει έναν συγκεκριμένο αριθμό από σημαίες.
- **Junior Tour:** Στη δοκιμασία αυτή ανταγωνίζεται τρεις αντιπάλους και νικητής αναδεικνύεται αυτός που θα τερματίσει πρώτος. Στόχος του παίχτη είναι να μαζέψει διάφορα Speed Ups που θα τον βοηθήσουν να γίνει πιο αεροδυναμικός και να προσπεράσει τους αντιπάλους του. Οι αντίπαλοι, οι οποίοι έχουν τεχνητή νοημοσύνη, βρίσκουν το γρηγορότερο μονοπάτι στο βουνό για να τερματίσουν, αποφεύγοντας αντιπάλους και κάνοντας προσπεράσεις.
- **Large Hill:** Στη δοκιμασία αυτή ο παίχτης παίρνει φόρα από μια μεγάλη κατηφόρα και μέσω Head Tracking προσπαθεί να κρατήσει ευθεία πορεία, να ελέγξει τον σκιάρ στον αέρα και να καταφέρει να τον προσγειώσει στο έδαφος ομαλά χωρίς να πέσει. Απαιτεί ακρίβεια και ταχύτητα από τον παίχτη ώστε να πηδήξει πιο μακριά από τους αντιπάλους.
- **Canadian Slope:** Στη δοκιμασία αυτή ο παίχτης πραγματοποιεί κατάβαση παίρνοντας από διάσπαρτα σημειάκια με τη διαφορά ότι τώρα αγριεμένοι λύκοι του έχουν στήσει καρτέρι. Σκοπός του παίχτη είναι να αποφύγει τους λύκους, οι οποίοι κινούνται έξυπνα προσομοιώνοντας μικρές αγέλες, και να φτάσει ζωντανός στη βάση του βουνού.

1.3 Πλατφόρμα

Το παιχνίδι σχεδιάστηκε στη μηχανή παιχνιδιών Unity 3d και ο κώδικας γράφτηκε σε C# και JavaScript. Επίσης, χρησιμοποιήθηκε το Oculus Rift SDK, μέσω του οποίου ενσωματώθηκε το Oculus Rift στη Unity και χρησιμοποιήθηκε ως συσκευή εισόδου. Παρόλο που η Unity υποστηρίζει την ανάπτυξη σε πολλαπλές πλατφόρμες, το συγκεκριμένο παιχνίδι είναι σχεδιασμένο για σταθερούς υπολογιστές με λειτουργικό σύστημα Windows 7, Windows 8 ή Windows 10. Επίσης θα πρέπει να διαθέτει 2 θύρες USB καθώς και κάρτα γραφικών Nvidia GTX 600 series ή AMD Radeon HD 7000 series(τουλάχιστον) με DVI-D ή HDMI υποδοχή. Επίσης, εκτός από τις παραπάνω ελάχιστες απαιτήσεις, συνιστάται η κάρτα γραφικών να μπορεί να τρέξει τρισδιάστατα παιχνίδια στα 1080p ανάλυση με 75fps και πάνω.

1.4 Βασικά Τεχνικά Χαρακτηριστικά

Το παιχνίδι αποτελείται από ορισμένα βασικά συστήματα, τα οποία περιγράφονται παρακάτω και τα οποία αποτελούν τα δομικά συστατικά που συνδέθηκαν ώστε να δημιουργηθεί μια ενιαία εφαρμογή.

1.4.1 Σύστημα Γραφικών

Στο σύστημα αυτό περιλαμβάνονται όλες εκείνες οι διεργασίες που απαιτούνται ώστε να δημιουργηθεί το τρισδιάστατο γραφικό περιβάλλον της εφαρμογής. Περιλαμβάνονται εργαλεία για την εισαγωγή τρισδιάστατων μοντέλων, εισαγωγή εικόνων, γραμματοσειρών, χειρισμό των υφών(texture) που χρησιμοποιήθηκαν και εφαρμογής τους στα μοντέλα που έχουν εισαχθεί, μοντελοποίηση επιφανειών και ότι άλλο χρειάζεται ένα παιχνίδι για να έχει μια ολοκληρωμένη εικόνα.

1.4.2 Σύστημα Φυσικής

Το σύστημα φυσικής που χρησιμοποιήθηκε είναι το PhysX της NVIDIA το οποίο βρίσκεται ενσωματωμένο στη Unity 3D. Παρέχει στα αντικείμενα πειστική φυσική συμπεριφορά δίνοντάς τους τη δυνατότητα να δέχονται επιταχύνσεις, να έχουν συγκρουστές, βαρύτητα καθώς και άλλες δυνάμεις. Ρυθμίζοντας διάφορες παραμέτρους μέσω scripting, μπορεί κανείς να δημιουργήσει αντικείμενα που συμπεριφέρονται με αρκετά ρεαλιστικό τρόπο.

1.4.3 Σύστημα Ήχου

Το σύστημα ήχου της Unity 3D επιτρέπει να εισαχθούν τόσο στερεοφωνικά όσο και μονοφωνικά αρχεία ήχου. Έχοντας τοποθετημένο πάντοτε τον ακροατή στην ενεργή κάμερα και μέσω διαφόρων πηγών ήχου (όπως για παράδειγμα ο ήχος του αέρα, ο θόρυβος των πέδινων, ο θόρυβος σε περίπτωση πτώσης κ.ά), δημιουργήθηκε ένα λειτουργικότερο και ρεαλιστικότερο παιχνίδι.

1.4.4 Σύστημα Φωτισμού

Απαραίτητη προϋπόθεση για να είναι ορατά και ρεαλιστικά τα γραφικά, είναι ο φωτισμός τόσο των μοντέλων όσο και των υφών τους (textures). Έτσι λοιπόν, χρησιμοποιήθηκαν διάφορες πηγές τις οποίες παρέχει το σύστημα φωτισμού της Unity και οι οποίες περιγράφονται στην ενότητα [3.4.11](#).

1.4.5 Σύστημα Γραφικής Διεπαφής Χρήστη

Μέσω του συστήματος αυτού δημιουργήσαμε ένα γραφικό περιβάλλον διεπαφής χρήστη το οποίο βρίσκεται εμβυθισμένο μέσα στον χώρο. Περιλαμβάνει τόσο τα μενού κατά την έναρξη του παιχνιδιού, όσο και όλο το διαδραστικό περιβάλλον κατά το χρόνο εκτέλεσής του. Η αλληλεπίδραση του χρήστη με αυτό γίνεται με το Oculus Rift.

1.4.6 Σύστημα Τεχνητής Νοημοσύνης

Το σύστημα τεχνητής νοημοσύνης αποτελεί ένα από τα σημαντικότερα κομμάτια της πτυχιακής εργασίας καθώς δίνει διαδραστικότητα στο παιχνίδι. Χρησιμοποιείται από τους αντιπάλους του παίκτη (αντίπαλοι σκίερ, λύκοι) ώστε να μπορούν να πλοηγούνται στο χώρο, να αποφεύγουν τους όμοιούς τους και να αντιδρούν στο παιχνίδι γενικότερα σαν αυτόνομοι πράκτορες, ανάλογα πάντα με τις απαιτήσεις του σχεδιαστή.

1.5 Δομή της εργασίας

Κλείνοντας με το πρώτο αυτό κεφάλαιο το οποίο ήταν εισαγωγικό, γίνεται μια συνοπτική αναφορά των υπόλοιπων έξι που ακολουθούν:

- Στο κεφάλαιο 2 παρουσιάζεται η σχετική έρευνα πάνω σε θεωρητικά θέματα τα οποία είναι απαραίτητο να γνωρίζει κανείς ώστε να κατανοήσει την αρχιτεκτονική και τον τρόπο με τον οποίο έχει υλοποιηθεί το βιντεοπαιχνίδι. Η έρευνα αυτή πραγματοποιήθηκε πριν ξεκινήσει ο σχεδιασμός και η υλοποίησή του και περιλαμβάνει την κατηγοριοποίηση των βιντεοπαιχνιδιών, τις βασικές έννοιες των γραφικών, το HCI στα βιντεοπαιχνίδια καθώς και τα εικονικά περιβάλλοντα.
- Στο κεφάλαιο 3 περιγράφονται όλα εκείνα τα εργαλεία και οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση του παιχνιδιού, της αρχιτεκτονικής καθώς και των βασικών λειτουργιών του.
- Στο κεφάλαιο 4 περιγράφεται ο σχεδιασμός του παιχνιδιού. Γίνεται αναφορά στο σενάριο, στο γραφικό περιβάλλον της εφαρμογής, καθώς και στη γραφική διεπαφή του χρήστη.
- Στο κεφάλαιο 5 αναλύεται λεπτομερώς η υλοποίηση του παιχνιδιού, καθώς και όλες οι διαδικασίες που ακολουθήθηκαν έως την ολοκλήρωσή του.
- Στο κεφάλαιο 6 περιγράφεται η μέθοδος αξιολόγησης που χρησιμοποιήθηκε, καθώς και οι εντυπώσεις του παιχνιδιού από τους χρήστες που το δοκίμασαν. Παρουσιάζονται ορι-

σμένες διορθώσεις που έγιναν με βάση τις παρατηρήσεις των χρηστών, καθώς και συνολικά συμπεράσματα και αποτελέσματα που απορρέουν.

- Στο κεφάλαιο 7 γίνεται μια ανακεφαλαίωση της διπλωματικής εργασίας ως προς το κατά πόσο επιτεύχθηκαν οι αρχικοί στόχοι και αναφέρονται πιθανές μελλοντικές επεκτάσεις οι οποίες θα μπορούσαν να ενσωματωθούν, ώστε να διευρύνουν τη λειτουργικότητα και την απήχυσή του.

Κεφάλαιο 2

Επισκόπηση Σχετικής Έρευνας

2.1 Εισαγωγή

Το κεφάλαιο αυτό ξεκινάει με μια ιστορική αναδρομή των βιντεοπαιχνιδιών και στη συνέχεια παρουσιάζονται οι κατηγορίες στις οποίες αυτά διαχωρίζονται. Έπειτα, γίνεται μια αναφορά στον αγωγό γραφικών, καθώς και στις τρισδιάστατες βιβλιοθήκες γραφικών OpenGL και Direct3D, τις οποίες υποστηρίζει η Unity. Ακολούθως, γίνεται αναφορά στη σημασία του HCI στα βιντεοπαιχνίδια, καθώς και ανάλυση της συνέχειας από τον πραγματικό κόσμο στον εικονικό (Reality–virtuality continuum) περιγράφοντας όλα τα στάδια που μεσολαβούν. Τέλος, το κεφάλαιο κλείνει με αναφορά στα HMDs.

2.2 Ιστορική Αναδρομή Βιντεοπαιχνιδιών

Με τον όρο βιντεοπαιχνίδι εννοείται ένα ηλεκτρονικό παιχνίδι, το οποίο περιλαμβάνει αλληλεπίδραση με μια διεπαφή χρήστη, δηλαδή επικοινωνία του βιντεοπαιχνιδιού με τον άνθρωπο, με σκοπό την παραγωγή οπτικής ανάδρασης σε μια συσκευή βίντεο. Η λέξη ‘βίντεο’ στο βιντεοπαιχνίδι παραδοσιακά αναφερόταν σε μια συσκευή εμφάνισης γραφικών raster (συσκευές καθοδικού σωλήνα (CRT)). Πλέον, με τη δημοφιλή χρήση του όρου βιντεοπαιχνίδι, εννοείται ένα παιχνίδι που μπορεί να παιχτεί σε κάθε τύπο συσκευής. Η ιστορία των βιντεοπαιχνιδιών, αρχίζει στα τέλη της δεκαετίας του ‘40. Προς τα τέλη του ‘50 και στα μέσα του ‘60, στην Αμερική, αρχίζουν να μπαίνουν στην καθημερινή μας ζωή οι υπολογιστές. Από εκείνη την περίοδο, τα βιντεοπαιχνίδια έκαναν την εμφάνιση τους στις παρακάτω πλατφόρμες:

- Πλατφόρμες Arcade (Arcade Platforms)
- Κονσόλες Βιντεοπαιχνιδιών (Video Game Consoles)

- Προσωπικοί Υπολογιστές (PCs)
- Συσκευές Χειρός (Hand-held Devices)

Παρακάτω ακολουθεί μια σύντομη ιστορική αναδρομή της εξέλιξης των βιντεοπαιχνιδιών κατά τη διάρκεια των τελευταίων δεκαετιών.

2.2.1 Πλατφόρμες Arcade (Arcade Platforms)

Οι πλατφόρμες Arcade είναι μηχανήματα διασκέδασης σχετικά μεγάλων διαστάσεων (1.5μ x 0.5μ) τα οποία λειτουργούσαν με κερματοδέκτη και συνήθως μπορούσε να παίξει κανείς ένα μόνο συγκεκριμένο παιχνίδι (σχήμα 2.1). Ήταν εγκατεστημένα σε δημόσιους χώρους όπως εστιατόρια, μπαρ καθώς και σε χώρους διασκέδασης. Τα Arcade παιχνίδια ήταν δημοφιλή από τη δεκαετία του 1970 μέχρι τη δεκαετία του 1990 όπου χαρακτηρίζεται ως η χρυσή εποχή των arcade παιχνιδιών. Σταδιακά όμως η δημοτικότητά τους μειώθηκε όταν οι κονσόλες βιντεοπαιχνιδιών άρχισαν να γίνονται ευρέως διαθέσιμες στο κοινό [1].



Σχήμα 2.1: Πλατφόρμες Arcade
Αριστερά: Pac-Man, **Κέντρο:** Space Invaders,
Δεξιά: Galaga

2.2.2 Κονσόλες Βιντεοπαιχνιδιών (Video Game Consoles)

Οι κονσόλες βιντεοπαιχνιδιών είναι υπολογιστές διαδραστικής ψυχαγωγίας ή τροποποιημένα υπολογιστικά συστήματα, τα οποία παράγουν ένα σήμα βίντεο ή εικόνας που μπορεί να χρησιμοποιηθεί με μια ηλεκτρονική συσκευή απεικόνισης (τηλεόραση, οθόνη κλπ) για να εμφανισθεί ένα βιντεοπαιχνίδι. Ο όρος 'κονσόλα βιντεοπαιχνιδιού' χρησιμοποιείται για να διακρίνει ένα μηχάνημα σχεδιασμένο για τους καταναλωτές, που το αγοράζουν και το χρησιμοποιούν αποκλειστικά και μόνο για βιντεοπαιχνίδια, από έναν προσωπικό υπολογιστή, ο οποίος έχει πολλές άλλες λειτουργίες. Μια κονσόλα βιντεοπαιχνιδιού λειτουργεί όπως ένας υπολογιστής και κατασκευάζεται από τα ίδια βασικά συστατικά, συμπεριλαμβανομένης της κεντρικής μονάδας επεξεργασίας (CPU), της κάρτας γραφικών (GPU) και της τυχαίας μνήμης προσπέλασης (RAM). Η Magnavox Odyssey ήταν η πρώτη εμπορική κονσόλα βιντεοπαιχνιδιών που σχεδιάστηκε, η οποία δεν είχε ήχο και μπορούσε να προβάλλει μόνο ασπρόμαυρες εικόνες [2]. Ήταν βασισμένη

στο Brown Box που είχε σχεδιαστεί από τον Ralph Baer και τους συνεργάτες του. Μεγάλη θραύση την δεκαετία του '80 έκαναν οι κονσόλες Atari. Στις μέρες μας οι πιο δημοφιλείς κονσόλες είναι οι Nintendo Wii U, Sony PlayStation 4 (PS4) και Xbox One (σχήμα 2.2,2.3) [3].



Σχήμα 2.2: Πλατφόρμες Arcade
Αριστερά: Brown Box (1967), **Κέντρο:** Magnavox Odyssey (1972), **Δεξιά:** Atari 2600 (1977)



Σχήμα 2.3: Κονσόλες Βιντεοπαιχνιδιών
Αριστερά: Nintendo Wii U (2012), **Κέντρο:** Sony PlayStation 4 (PS4) (2013), **Δεξιά:** Xbox One (2013)

2.2.3 Προσωπικοί Υπολογιστές(PCs)

Παρατηρώντας την αγορά βιντεοπαιχνιδιών, φαίνεται ότι οι κονσόλες είναι τα πιο δημοφιλή προϊόντα στη βιομηχανία. Τα τελευταία όμως χρόνια αυτό έχει αρχίσει να αλλάζει και σύμφωνα με τους ερευνητές οι πωλήσεις που δημιουργούνται από τα παιχνίδια των υπολογιστών, αναμένεται να ξεπεράσουν αυτές των κονσολών, με την διαφορά να αυξάνεται χρόνο με τον χρόνο εξαιτίας της δημοτικότητας που κερδίζουν σε χώρες όπως η Κίνα και η Ινδία. Ένα από τα μεγαλύτερα πλεονεκτήματα που προσφέρουν οι προσωπικοί υπολογιστές, είναι ότι ο χρήστης μπορεί να παίξει τα πιο όμορφα και δημοφιλή παιχνίδια και να τα ευχαριστηθεί παίζοντάς τα στο διαδίκτυο (online). Επίσης, στις μέρες μας είναι το μοναδικό μέσο από το οποίο μπορεί κανείς να παίξει υψηλής ποιότητας εικονικής πραγματικότητας παιχνίδια (Virtual Reality Games), προσφέροντας στους χρήστες την ευκαιρία να φορέσουν ένα headset και να ταξιδέψουν σε άλλους κόσμους, αν και αυτό ενδέχεται να αλλάξει καθώς καινούρια headsets αναμένεται να κυκλοφορήσουν, τα οποία θα είναι συμβατά και με άλλες πλατφόρμες. Ο υπολογιστής δεν θα είναι μόνο η τεχνολογία στην οποία οι παίκτες ξοδεύουν χρήματα παίζοντας παιχνίδια, αλλά σύντομα θα γίνει ένα από τα βασικά πεδία εφαρμογής νέων τεχνολογιών [4].

2.2.4 Συσκευές Χειρός (Hand-held Devices)

Σε αντίθεση με τα βιντεοπαιχνίδια προσωπικών υπολογιστών και κονσολών, τα παιχνίδια συσκευών χειρός τρέχουν σε μηχανήματα μικρού μεγέθους. Έτσι, δίνεται η δυνατότητα στους ανθρώπους να τα μεταφέρουν μαζί τους οπουδήποτε και να παίξουν με αυτά οποιαδήποτε στιγμή επιθυμούν. Σε γενικές γραμμές οι μηχανές αυτές μπορούν να αναφέρονται σε ειδικά φορητές κονσόλες παιχνιδιών, προσωπικούς ψηφιακούς βοηθούς (PDAs) ή κινητά τηλέφωνα (σχήμα 2.4). Λόγω του περιορισμού τους από πλευράς hardware, τέτοιες συσκευές παιχνιδιών συχνά υποφέρουν από μικρό μέγεθος οθόνης, περιορισμένο χώρο αποθήκευσης καθώς και από μικρή διάρκεια ζωής της μπαταρίας. Αυτά τα προβλήματα δεν δημιουργούν μόνο δυσκολίες στην ανάπτυξη των παιχνιδιών χειρός, αλλά πολλές φορές και δισταγμούς στους χρήστες. Ευτυχώς, οι ελλείψεις αυτές έχουν διευθετηθεί και έχουν βρεθεί τρόποι κατά τη διάρκεια των τελευταίων ετών, με τους οποίους μπορούν να επιλυθούν [5].



Σχήμα 2.4: Συσκευές χειρός
Αριστερά: Game Boy Advance (2001), **Κέντρο:**
Google Nexus 10 (2012), **Δεξιά:** iPhone5 (2012)

2.3 Κατηγορίες Βιντεοπαιχνιδιών

Η κατηγοριοποίηση των βιντεοπαιχνιδιών γίνεται βάσει της αλληλεπίδρασης του χρήστη με το παιχνίδι (gameplay) και όχι βάσει των οπτικών διαφοροποιήσεων. Επίσης, είναι ανεξάρτητη από την ιστορία ή από το περιβάλλον το οποίο παρουσιάζει [6]. Ωστόσο, οι τρέχουσες περιγραφές βιντεοπαιχνιδιών είναι ασαφείς, απροσδιόριστες και περιέχουν πολλαπλές διαστάσεις πληροφορίας, με αποτέλεσμα ορισμένες κατηγορίες να φαίνεται ότι έχουν ασαφή όρια και να επικαλύπτονται εννοιολογικά. Παρακάτω παρουσιάζονται 13 κατηγορίες οι οποίες σύμφωνα με τους ερευνητές είναι οι βασικές [7, 8].

- Δράσης (Action Games)
- Arcade Games
- Πάλης (Fighting Games)
- Γενικής Διασκέδασης (General Entertainment Games)

- Μουσικής & Ρυθμού (Music & Rhythm Games)
- Αφήγησης (Narrative Games)
- Αγώνων Οδήγησης (Driving/Racing Games)
- Ρόλων (RPG Games)
- Βολών (Shooter Games)
- Ικανοτήτων & Τύχης (Skill & Chance)
- Προσομοίωσης (Simulation Games)
- Αθλητισμού (Sports Games)
- Στρατηγικής (Strategy Games)

Αξίζει να σημειωθεί ότι λόγω της συνεχούς και ραγδαίας εξέλιξης των βιντεοπαιχνιδιών πολλά από αυτά δύναται να είναι υβρίδια των παραπάνω βασικών κατηγοριών συνδυάζοντας χαρακτηριστικά τους με αποτέλεσμα τη δημιουργία νέων υποκατηγοριών.

2.3.1 Δράσης (Action Games)

Τα παιχνίδια δράσης επιτρέπουν στον χρήστη να προσωποποιηθεί και να ελέγχει έναν χαρακτήρα σε καταστάσεις παρόμοιες με εκείνες που υπάρχουν σε ταινίες δράσης. Κοινά χαρακτηριστικά στοιχεία αυτών των παιχνιδιών αποτελούν οι κινήσεις (τρέξιμο, άλματα, αιώρηση), οι εξερευνήσεις και οι πολλές πολεμικές μάχες, με απώτερο σκοπό πάντα να ξεπεράσουν τους εχθρούς και να τερματίσουν χωρίς να καταστρέψουν τους εαυτούς τους (σχήμα 2.5). Ο παίχτης απαιτείται να έχει γρήγορα αντανακλαστικά, ακρίβεια στις κινήσεις του και ταχύτητα. Τα περισσότερα παιχνίδια δράσης είναι πραγματικού χρόνου (real time gameplay) [9, 10].



Σχήμα 2.5: Παιχνίδια δράσης
Αριστερά: *Tomb Raider: Underworld* (2008),
Κέντρο: *Lego Star Wars II: The Original Trilogy*
 (2006), **Δεξιά:** *God of War II* (2007)

2.3.2 Arcade Games

Τα Arcade παιχνίδια είναι παρόμοια (ή και ίδια) με τα παιχνίδια που λειτουργούσαν σε Arcade πλατφόρμες (σχήμα 2.6). Ήταν και είναι δημοφιλή σε κοινό όλων των ηλικιών. Στη σημερινή εποχή τα περισσότερα (web browser based) παιχνίδια εμπίπτουν σε αυτή την κατηγορία. Τα παιχνίδια αυτά χαρακτηρίζονται από απλή διεπαφή (interface) μεταξύ χρήστη και παιχνιδιού, καθώς και απλούς κανόνες. Συχνά δίνουν έμφαση στο σκοράρισμα πόντων και στην εξέλιξη των επιπέδων [8, 11]. Επιπλέον, τα παιχνίδια αυτά αν και θεωρούνται ‘της παλιάς σχολής’ (Old School), παρόλο της μεγάλης εξέλιξης των βιντεοπαιχνιδιών και κονσολών, συνεχίζουν να είναι διαχρονικά και να βρίσκονται στην αγορά [12].



Σχήμα 2.6: Παιχνίδια δράσης
Αριστερά: Flow (arcade game Blockade 1976),
Κέντρο: Pacman (1980), **Δεξιά:** Joust (1982)

2.3.3 Πάλης (Fighting Games)

Τα παιχνίδια πάλης είναι ένα είδος βιντεοπαιχνιδιών στα οποία ο παίχτης ελέγχει έναν χαρακτήρα ο οποίος μάχεται με έναν ή περισσότερους αντιπάλους. Οι χαρακτήρες τείνουν να έχουν ίσες δυνάμεις και το παιχνίδι εξελίσσεται σε πολλούς γύρους οι οποίοι λαμβάνουν χώρα σε μια αρένα. Επιπλέον, έχουν συνήθως ιδιαίτερες ικανότητες ή επιθετικές κινήσεις που πραγματοποιούνται από το γρήγορο πάτημα μιας ακολουθίας πλήκτρων, τα γνωστά σε όλους combos (σχήμα 2.7) [11].



Σχήμα 2.7: Παιχνίδια πάλης
Αριστερά: Dead or Alive 4 (2005), **Κέντρο:** Virtua
Fighter 5 (2005), **Δεξιά:** WWE SmackDown vs. Raw
2007 (2006)

2.3.4 Γενικής Διασκέδασης (General Entertainment Games)

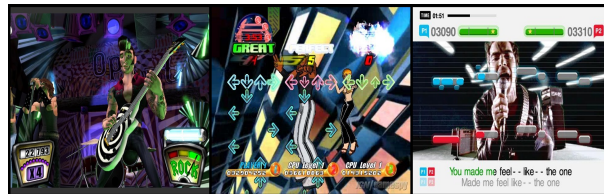
Η κατηγορία γενικής διασκέδασης χαρακτηρίζεται από παιχνίδια επίλυσης προβλημάτων (puzzle) και γνώσεων (trivia). Είναι συνήθως παιχνίδια για έναν παίκτη αλλά μερικά υποστηρίζουν και πολλαπλούς (party games). Τα παιχνίδια αυτά καταλαμβάνουν περιορισμένη περιοχή του παιχνιδιού και προβάλλονται από μια σταθερή και ακίνητη κάμερα (σχήμα 2.8) [8].



Σχήμα 2.8: Παιχνίδια γενικής διασκέδασης
Αριστερά: *Bejeweled 2* (2004), **Κέντρο:**
WarioWare: Smooth Moves(party game) (2006),
Δεξιά: *Bust-A-Move Bash!* (2007)

2.3.5 Μουσικής & Ρυθμού (Music & Rhythm Games)

Παιχνίδια στα οποία η κύρια λειτουργικότητά τους περιλαμβάνει τον χρήστη να χορεύει, να τραγουδάει ή να παίζει ένα απλουστευμένο μουσικό όργανο, ταξινομούνται ως παιχνίδια Μουσικής & Ρυθμού (σχήμα 2.9). Τα παιχνίδια αυτά προσανατολίζονται σχεδόν αποκλειστικά γύρω από τις αλληλεπιδράσεις του παίκτη με μια μουσική παρτιτούρα ή μεμονωμένα τραγούδια [8].

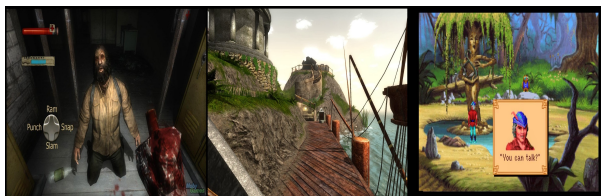


Σχήμα 2.9: Παιχνίδια μουσικής & ρυθμού
Αριστερά: *Guitar Hero II* (2006), **Κέντρο:** *Dance Dance Revolution Universe 3* (2008), **Δεξιά:** *SingStar* (2004)

2.3.6 Αφήγησης (Narrative Games)

Στα παιχνίδια αφήγησης κύρια έμφαση δίνεται στην εξέλιξη της ιστορίας. Κοινά στοιχεία των παιχνιδιών αφήγησης είναι η εξερεύνηση, τα παζλ, η συλλογή και οι μη βίαιες αλληλεπιδράσεις με άλλους χαρακτήρες (non-player character (NPC)). Πιθανά δευτερεύοντα στοιχεία των παιχνιδιών αφήγησης περιλαμβάνουν κίνηση και δράση (σχήμα 2.10). Οι μάχες είναι πολλές φορές δευτερεύουσας σημασίας στα παιχνίδια αυτά. Ο τερματισμός ενός τέτοιου παιχνιδιού, συνήθως

συμπίπτει με το τέλος της ιστορίας και την εκπλήρωση της πλοκής-υπόθεσης από τον πρωταγωνιστή [8].



Σχήμα 2.10: Παιχνίδια αφήγησης
Αριστερά: *Condemned: Criminal Origins* (2005),
Κέντρο: *Myst* (1993), **Δεξιά:** *King's Quest 5* (1990)

2.3.7 Αγώνων Οδήγησης (Driving/Racing Games)

Τα παιχνίδια αγώνων οδήγησης δίνουν τη δυνατότητα στο χρήστη να ελέγχει ένα όχημα σε μια ανταγωνιστική διοργάνωση που συμμετέχουν και άλλοι. Οι αγώνες λαμβάνουν χώρα σχεδόν πάντα σε καθορισμένες πίστες και μπορεί να περιλαμβάνουν πολλούς κύκλους. Ο χρήστης γενικά ανταγωνίζεται είτε παίκτες που ελέγχονται από άλλους χρήστες είτε παίκτες που ελέγχονται από τον υπολογιστή (NPC) με τεχνητή νοημοσύνη (artificial intelligence). Πολλά παιχνίδια αγώνων οδήγησης προσπαθούν να δημιουργήσουν αγωνιστικές συνθήκες όσο πιο ρεαλιστικές γίνεται, μέσω της οδήγησης με μεγάλη ακρίβεια και λεπτομέρεια (σχήμα 2.11) [11].



Σχήμα 2.11: Παιχνίδια αγώνων οδήγησης
Αριστερά: *Need for Speed: Most Wanted* (2005),
Κέντρο: *Project Gotham Racing 3* (2005), **Δεξιά:**
Burnout Revenge (2005)

2.3.8 Ρόλων (RPG Games)

Τα παιχνίδια αυτής της κατηγορίας έχουν την προέλευσή τους στο επιτραπέζιο παιχνίδι Dungeons & Dragons και βασικός στόχος του παίκτη είναι να ελέγχει έναν ή περισσότερους χαρακτήρες. Οι χαρακτήρες έχουν τη δυνατότητα να βελτιώνονται με την εξέλιξη του παιχνιδιού, αποκτώντας εμπειρία από μάχες και δοκιμασίες. Επιπλέον, έχουν τη δυνατότητα να βελτιώνουν τις δεξιότητες και τις ικανότητές τους. Τα παιχνίδια αυτά συνήθως κρύβουν μεγάλη ιστορία από πίσω, η οποία συχνά εξελίσσεται είτε σε φανταστικό είτε σε κόσμο της επιστημονικής

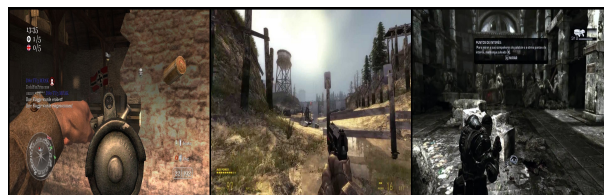
φαντασίας, ο οποίος αποτελείται από τέρατα, μαγεία ή/και προηγμένη τεχνολογία (σχήμα 2.12) [11].



Σχήμα 2.12: Παιχνίδια ρόλων
Αριστερά: *Final Fantasy XII* (2006), **Κέντρο:** *The Elder Scrolls IV: Oblivion* (2006), **Δεξιά:** *Untold Legends: Dark Kingdom* (2006)

2.3.9 Βολών (Shooter Games)

Τα παιχνίδια βολής βασίζονται στην ταχύτητα και στον χρόνο αντίδρασης του παίχτη. Ο χαρακτήρας συνήθως κρατάει κάποιο όπλο και σκοπός του είναι να χτυπήσει τους αντιπάλους και να περάσει τις δοκιμασίες δίχως να πεθάνει (σχήμα 2.13). Οι βασικές ενέργειες που κάνει ο χρήστης είναι να πυροβολεί, να κινείται, να πηδάει και να αλληλεπιδρά με το περιβάλλον [10].



Σχήμα 2.13: Παιχνίδια βολών
Αριστερά: *Call of Duty 2* (2005), **Κέντρο:** *Half-Life 2* (2004), **Δεξιά:** *Gears of War* (2006)

2.3.10 Ικανοτήτων & Τύχης (Skill & Chance)

Στην κατηγορία των παιχνιδιών ικανοτήτων και τύχης περιλαμβάνονται όλα τα παραδοσιακά παιχνίδια, όπως η ντάμα, το σκάκι και το τάβλι, καθώς και παιχνίδια καζίνο όπως πόκερ, ζάρια κ.α (σχήμα 2.14). Τα περισσότερα επιτραπέζια παιχνίδια και παιχνίδια καρτών συμπεριλαμβάνονται σε αυτή την κατηγορία με την εξαίρεση των card battle παιχνιδιών και των επιτραπέζιων στρατηγικής, όπως το Risk [11]



Σχήμα 2.14: Παιχνίδια ικανοτήτων & τύχης
Αριστερά: UNO (2006), **Κέντρο:** World Series of
 Poker: Tournament of Champions (2006), **Δεξιά:**
 Catan (2008)

2.3.11 Προσομοίωσης (Simulation Games)

Τα παιχνίδια προσομοίωσης προσπαθούν να αντιγράψουν διάφορες δραστηριότητες και εμπειρίες από την πραγματική ζωή, θέτοντας τους παίκτες σε καταστάσεις τις οποίες υπό άλλες συνθήκες δεν θα μπορούσαν να βιώσουν. Επιπλέον, ακολουθούν κανόνες και νόμους του πραγματικού κόσμου, όσο πιο πιστά γίνεται (σχήμα 2.15). Οι βασικές υποκατηγορίες στις οποίες χωρίζονται τα παιχνίδια προσομοίωσης είναι:

- **Προσομοίωση Οχήματος (Vehicle Simulation):** Τα οποία δημιουργούν εξαιρετικά ακριβείς εμπειρίες στον χρήστη και εφαρμόζονται σε αεροπλάνα, πλοία, άρματα μάχης κ.ά.
- **Προσομοίωση Ζωής (Life Simulation):** Τα οποία επιτρέπουν στους παίκτες να διαχειρίζονται κάποιον ή κάποιους χαρακτήρες και να αγωνίζονται σε διάφορα γεγονότα. Στην ούσια δίνεται η δυνατότητα στον παίκτη να παίζει σαν να συμμετέχει σε κάποιο γεγονός.
- **Προσομοίωση Διαδικασίας (Process Simulation):** Τα οποία προσομοιώνουν πολύπλοκα συστήματα και διαδικασίες όπως η κατασκευή και διαχείριση πόλεων, πολιτισμών, στρατών, σιδηροδρόμων κ.ά.

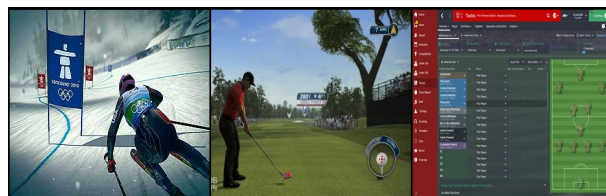
Στα παιχνίδια προσομοίωσης συνήθως δεν υπάρχουν αυστηρά καθορισμένοι στόχοι, με τους παίκτες να μπορούν να ελέγξουν ελεύθερα τον χαρακτήρα, το εκάστοτε όχημα ή την εκάστοτη διαδικασία. Όλα τα παιχνίδια είναι μια μορφή προσομοίωσης, αλλά σε αυτού του είδους έχει γίνει μεγάλη προσπάθεια, ώστε να είναι όσο το δυνατόν πιο ρεαλιστικά. Δεν έχει τόση σημασία αν είναι ευχάριστα ή δύσκολα να τα παίζει κανείς, αρκεί να δίνουν στον παίκτη την αίσθηση του πραγματικού και ρεαλιστικού. Ακόμα, η πλειονότητα των παιχνιδιών προσομοίωσης είναι εκτός-σύνδεσης (offline) εξαιτίας των πολύ υψηλών απαιτήσεων ρεαλισμού [9, 10].



Σχήμα 2.15: Παιχνίδια προσομοίωσης
Αριστερά: *Flight Simulator X* (2006), **Κέντρο:** *The Sims 2* (2004), **Δεξιά:** *SimCity 4* (2003)

2.3.12 Αθλητισμού (Sports Games)

Τα παιχνίδια αθλητισμού αποτελούν τις πρώτες κατηγορίες παιχνιδιών μοντελοποιώντας δημοφιλή αθλήματα. Από τις πρώτες μέρες της σχεδίασης ηλεκτρονικών παιχνιδιών, τότε που τα παιχνίδια δεν είχαν αποκτήσει την ταυτότητα και την πρωτοτυπία που έχουν σήμερα, οι σχεδιαστές στράφηκαν προς αυτά. Αυτό αποσκοπούσε και στο γεγονός ότι από πλευράς μάρκετινγκ ήταν καλύτερο να προσφέρει στον συντηρητικό καταναλωτή παιχνίδια με τα οποία είναι εξοικειωμένος στην καθημερινή του ζωή, παρά άγνωστα προς αυτόν. Συνεπώς, ξεκίνησε η ανάπτυξη παιχνιδιών βασισμένα σε μπάσκετ, ποδόσφαιρο, μπίτζμπολ, τένις, πυγμαχία κ.ά. Κάποια από αυτά τα παιχνίδια δίνουν έμφαση στον τρόπο με τον οποίο παίζεται το παιχνίδι και άλλα στην στρατηγική και στην οργάνωση του παιχνιδιού (σχήμα 2.16). Αποτελεί μια από τις δημοφιλέστερες κατηγορίες παιχνιδιών, καθώς τα χαρακτηριστικά των ομάδων και των παιχτών αναφέρονται στην πραγματικότητα και ανανεώνονται κάθε χρόνο [11].



Σχήμα 2.16: Παιχνίδια αθλητισμού
Αριστερά: *Vancouver 2010* (2010), **Κέντρο:** *Tiger Woods PGA Tour 14* (2013), **Δεξιά:** *Football Manager 2015* (2014)

2.3.13 Στρατηγικής (Strategy Games)

Τα παιχνίδια στρατηγικής αποτελούν μια μεγάλη κατηγορία βιντεοπαιχνιδιών. Δίνουν μεγάλη έμφαση στη σκέψη και στο σχεδιασμό για να πετύχει ο παίχτης τη νίκη και απευθύνονται κυρίως σε ώριμο κοινό. Τα παιχνίδια αυτά απαιτούν στρατηγική, τακτική και λογική σκέψη. Η προέλευση όλων των παιχνιδιών στρατηγικής είναι κλασικά παιχνίδια όπως το σκάκι, το Risk κ.ά, αλλά σε νέα παιχνίδια η εμπειρία του παιχνιδιού είναι πολύ λιγότερο αφηρημένη και η πολυπλοκότητα

συνήθως υψηλότερη ως προς το ότι έχεις να διαχειριστείς ένα περιορισμένο σύνολο πόρων για να πετύχεις έναν σκοπό. Τα περισσότερα παιχνίδια αυτής της κατηγορίας είναι στρατιωτικού χαρακτήρα. Στα παιχνίδια στρατηγικής υπάρχουν κυρίως 2 ειδών συστατικά:

- Διαχείριση Πόρων (Resource management) η οποία περιλαμβάνει την κατασκευή κτιρίων και μονάδων.
- Αποφάσεις Πόρων (Resource decisions) η οποία περιλαμβάνει τις αποφάσεις για το πως να θέσει τους πόρους σε δράσεις.

Τα παιχνίδια στρατηγικής χωρίζονται σε ανά γύρου (turn-based) και σε πραγματικού χρόνου (real-time) ανάλογα με το αν ο παίχτης περιμένει τους υπόλοιπους αντιπάλους να τελειώσουν τις κινήσεις τους ή παίζουν όλοι μαζί ταυτόχρονα. Στις μέρες μας τα παιχνίδια στρατηγικής έχουν γίνει συνώνυμα με τα πραγματικού χρόνου παιχνίδια (RTS) τα οποία έγιναν δημοφιλή με παιχνίδια όπως τα Dune II και Warcraft. Τα παιχνίδια αυτά βασίζονται στην άμεση δράση-αντίδραση μεταξύ παίχτη και αντιπάλου, δίνοντας λιγότερο χρόνο στους παίχτες να σκεφτούν στρατηγικές. Επιπλέον, τα πραγματικού χρόνου παιχνίδια θέτουν τον παίχτη υπό πίεση έτσι ώστε να πρέπει να επιμερίσει ορθολογικά τον χρόνο που έχει στη διάθεσή του, για να διαχειριστεί τους διάφορους πόρους, να χτίσει κτίρια, να κινηθεί και να επιτεθεί (σχήμα 2.17). Συνεπώς, με τον πολυδιάστατο τρόπο παιχνιδιού δίνεται άλλος βαθμός στον όρο “στρατηγική”.

Τα ανά-γύρου παιχνίδια (turn-based), παρόλο που είναι λιγότερο δημοφιλή είναι ακόμα σχετικά δυνατά στο διαδίκτυο. Βασικό τους μειονέκτημα είναι ότι ο χρήστης δεν μπορεί να ελέγξει πόσο χρόνο θέλει να παίξει και τότε, καθώς πρέπει να περιμένει μέχρι ο άλλος παίχτης να κάνει μια κίνηση ή να περάσει ο χρόνος. Ταυτόχρονα όμως, αποτελεί και πλεονέκτημα, καθώς αυτό το διάστημα αναμονής, δίνει στον παίχτη τη δυνατότητα να σκεφτεί τις στρατηγικές του [9, 10].



Σχήμα 2.17: Παιχνίδια στρατηγικής
Αριστερά: Warcraft III: Reign of Chaos (2002)
Κέντρο: Command & Conquer: Red Alert 2 (2000)
Δεξιά: Age of Empires III (2005)

2.4 Αγωγός Γραφικών Graphics Pipeline

Στα τρισδιάστατα γραφικά υπολογιστών, ο αγωγός γραφικών αναφέρεται στην ακολουθία βημάτων που χρησιμοποιούνται για την δημιουργία μιας 2D raster αναπαράστασης μια τρισδιάστατης σκηνής. Αφού δημιουργηθεί ένα τρισδιάστατο μοντέλο, για παράδειγμα σε ένα βιντεοπαιχνίδι ή σε οποιαδήποτε άλλα τρισδιάστατα κινούμενα σχέδια, ο αγωγός γραφικών αποτελεί την διαδικασία η οποία μετατρέπει αυτό το τρισδιάστατο μοντέλο σε αυτό που προβάλλει ο υπολογιστής. Οι υπολογιστές άρχισαν να δέχονται σημαντικές αλλαγές τα τελευταία χρόνια με την εισαγωγή μιας ξεχωριστής κάρτας βίντεο και την άνοδο του υλικού επιτάχυνσης γραφικών (hardware accelerated graphics). Αυτό έχει οδηγήσει στην ανάγκη για ένα προγραμματιζόμενο αγωγό γραφικών που μπορεί να χειραγωγηθεί από σκιαστές (shaders), καθιστώντας με αυτό τον τρόπο τις μη επαναπρογραμματιζόμενες υλοποιήσεις του αγωγού γραφικών απαρχαιωμένες. Παρακάτω αναφέρονται επιγραμματικά τα στάδια του αγωγού γραφικών [13]:

- Μοντελοποίηση και μετασχηματισμός (Modelling and transformation)
- Φωτισμός (Lighting)
- Μετασχηματισμός κάμερας (Camera transformation)
- Ψαλίδισμα (Clipping)
- Μετασχηματισμός προβολής (Projection transformation)
- Scan Conversion

2.5 Τρισδιάστατες Βιβλιοθήκες Γραφικών 3D APIs

Τα τρισδιάστατα γραφικά έχουν γίνει τόσο δημοφιλή, ειδικά στα βιντεοπαιχνίδια, με αποτέλεσμα να έχουν δημιουργηθεί τρισδιάστατες βιβλιοθήκες γραφικών (3D APIs) έτσι ώστε να διευκολύνουν τις διαδικασίες σε όλα τα στάδια των γραφικών. Οι βιβλιοθήκες αυτές έχουν αποδειχθεί ζωτικής σημασίας για τους κατασκευαστές υλικού γραφικών, καθώς τους παρέχουν τη δυνατότητα να έχουν πρόσβαση στο υλικό (hardware) σε ένα αφηρημένο επίπεδο. Τις περισσότερες φορές, οι μηχανές 3D ή τα συστήματα απόδοσης στις μηχανές παιχνιδιών κατασκευάζονται πάνω σε ένα API γραφικών το οποίο παρέχει μια αφαίρεση του λογισμικού της μονάδας επεξεργασίας γραφικών GPU ή της κάρτας βίντεο. Οι τρισδιάστατες βιβλιοθήκες γραφικών χωρίζονται σε χαμηλού και υψηλού επιπέδου. Στην παρούσα διπλωματική εργασία επιλέχτηκε η μηχανή παιχνιδιών Unity 3D η οποία χρησιμοποιεί την OpenGL ES και την Direct3D [14].

2.5.1 Direct3D

Η Direct3D είναι ένα 3D API, μέρος μιας ευρύτερης βιβλιοθήκης πακέτων γραφικών της DirectX της Microsoft. Χρησιμοποιείται ευρέως από προγραμματιστές για τη δημιουργία παιχνιδιών, καθώς η υψηλή απόδοση σε αυτά αποτελεί βασικό παράγοντα. Παρόλο που οι πρώτες εκδόσεις είχαν λάθη αποσπώντας αρνητικά σχόλια από την κοινότητα των προγραμματιστών, πλέον αποτελεί μια βασική βιβλιοθήκη στα τρισδιάστατα γραφικά. Δίνει τη δυνατότητα στις εφαρμογές να εκτελούνται είτε σε ολόκληρη την οθόνη είτε σε μικρό παράθυρο αν προγραμματιστούν κατάλληλα. Χρησιμοποιεί εξελιγμένες δυνατότητες γραφικών, καθώς και επιτάχυνση υλικού της κάρτας γραφικών εάν είναι διαθέσιμο από την κάρτα, επιτρέποντας την επιτάχυνση σε ολόκληρη την διοχέτευση γραφικών ή σε ένα κομμάτι της. Επιγραμματικά αναφέρονται παρακάτω ορισμένα πλεονεκτήματα και μειονεκτήματα της Direct3D [15].

- **Πλεονεκτήματα:**

- Προγραμματιζόμενοι pixel και vertex shaders (σκιαστές).
- Περιέχει συναρτήσεις για να απαριθμείτε ακριβώς τι υλικό γραφικών είναι διαθέσιμο στο σύστημα.
- Το συντακτικό και η δομή έχουν εξελιχθεί σε μια μορφή, τέτοια ώστε ο προγραμματισμός να γίνεται ευκολότερος από ότι ήταν στο παρελθόν.
- Μπορούν να γίνουν αλλαγές χωρίς να χαλάσει ο υπάρχων κώδικας.

- **Μειονεκτήματα:**

- Αναβαθμίζεται κάθε χρόνο που είναι σχετικά αργός ρυθμός αν σκεφτεί κανείς την ταχύτητα με την οποία κινείται η βιομηχανία γραφικών.
- Δεν είναι open standard καθώς μόνο η Microsoft αποφασίζει τι θα συμπεριληφθεί στην έκδοση.
- Χρειάζεται περισσότερος κώδικας για να αρχικοποιηθεί κάτι.
- Ο σχεδιασμός παιχνιδιού σε Direct3D με C είναι δύσκολος και μπορεί να οδηγήσει σε απογοήτευση.

2.5.2 OpenGL

Η OpenGL είναι μια διεπαφή λογισμικού για την κάρτα γραφικών. Αυτή η διεπαφή αποτελείται από περίπου 150 διακριτές εντολές που μπορεί κανείς να χρησιμοποιήσει για να καθορίσει τα αντικείμενα και τις λειτουργίες που απαιτούνται για την παραγωγή διαδραστικών τρισδιάστατων

εφαρμογών. Η OpenGL έχει σχεδιαστεί ως μια βελτιωμένη, ανεξάρτητη από το υλικό διεπαφή που να μπορεί να εφαρμοστεί σε πολλές διαφορετικές πλατφόρμες υλικού. Για να επιτευχθούν αυτά, καμία εντολή για την εκτέλεση ενεργειών στα παραθύρα ή στην απόκτηση εισόδου στον χρήστη δεν περιλαμβάνεται στην OpenGL. Αντ' αυτού, θα πρέπει κανείς να εργαστεί μέσω οποιουδήποτε συστήματος παραθύρων ελέγχει το συγκεκριμένο υλικό που χρησιμοποιείται. Ομοίως, η OpenGL δεν παρέχει εντολές υψηλού επιπέδου για την περιγραφή τρισδιάστατων μοντέλων. Με την OpenGL, θα πρέπει κανείς να δημιουργήσει το μοντέλο της αρεσκείας του από ένα μικρό σύνολο των γεωμετρικών primitives σημείων, γραμμών και πολυγώνων. Επιγραμματικά αναφέρονται παρακάτω ορισμένα πλεονεκτήματα και μειονεκτήματα της [15].

- **Πλεονεκτήματα:**

- Είναι πολύ φορητή και μπορεί να τρέξει σχεδόν σε οποιαδήποτε πλατφόρμα εξαιτίας του Open Standard.
- Περιέχει ένα μεγάλο εύρος χαρακτηριστικών τόσο στον πυρήνα της όσο και μέσω προεκτάσεων, επιτρέποντας της να μένει σύγχρονη με τα νέα χαρακτηριστικά υλικού.
- Χρησιμοποιείται για πολλά διαφορετικά πράγματα. Η σταθερότητα και η υποστήριξή της σε ένα ευρύ φάσμα πλατφορμών την καθιστά ελκυστική προς τους ανθρώπους των τεχνολογικών τομέων. Είναι αναγνωρισμένη ως το βιομηχανικό πρότυπο για τα γραφικά παντού εκτός από τη βιομηχανία παιχνιδιών, όπου η Direct3D αποτελεί τον βασικό ανταγωνιστή της.

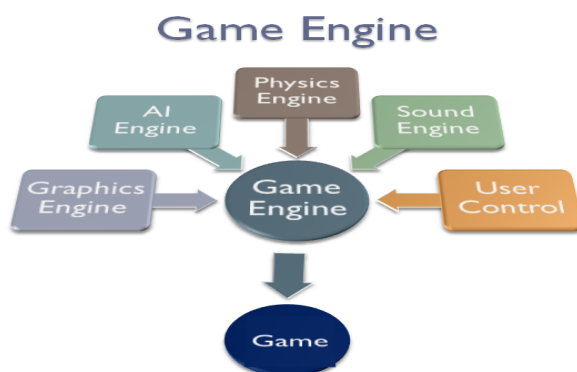
- **Μειονεκτήματα:**

- Οι προεκτάσεις (extensions) οι οποίες περιέχει, παρόλο που είναι ισχυρές, καθιστούν τον κώδικα αρκετά περίπλοκο ορισμένες φορές.
- Οι συμβάσεις ονομασίας των συναρτήσεων μπορεί να φαίνονται υπερβολικές κατά καιρούς, δεδομένου ότι πολλά IDEs έχουν θεματική βοήθεια που μπορεί να δείξει στον προγραμματιστή τις παραμέτρους που απαιτούνται. Με 12 διαφορετικά ονόματα για μια συνάρτηση μπορεί να φαίνεται παράξενο σε προγραμματιστές C++ να συνηθίσουν να λειτουργούν με τέτοια υπερφόρτωση (function overloading).

2.6 Επιλογή της Unity 3D

Για την δημιουργία ενός παιχνιδιού δεν χρειάζεται μόνο σχεδιασμός γραφικών. Χρειάζεται ήχος, μια μηχανή φυσικής, χειρισμός εισόδων και αποκρίσεων χρήστη, τεχνητή νοημοσύνη κ.ά (σχήμα 2.18). Οι μηχανές παιχνιδιών (game engines) έρχονται με ένα πρότυπο API υψηλού επιπέδου

που επιτρέπει τον χειρισμό όλων αυτών. Μας επιτρέπουν να επικεντρωθούμε σε πιο σημαντικά πράγματα, όπως το gameplay του παιχνιδιού καθώς και την διασύνδεση όλων αυτών πολύ πιο εύκολα και γρήγορα.



Σχήμα 2.18: Τυπική Μηχανή Γραφικών

Οι λόγοι που μας οδήγησαν στην επιλογή της Unity 3D έναντι άλλων μηχανών παραθέτονται παρακάτω:

- Παρέχει δωρεάν έκδοση.
- Θεωρείται από τις εύκολες μηχανές παιχνιδιών, δίνοντας την δυνατότητα στους προγραμματιστές να δημιουργήσουν οποιοδήποτε είδος παιχνιδιού.
- Υποστηρίζει πληθώρα πλατφόρμων (PC, Console, Mobile, Web).
- Δίνει τη δυνατότητα στους προγραμματιστές να συντάξουν κώδικα σε μορφή scripting σε γλώσσες JavaScript, C# και Boo.
- Έχει το Asset Store από το οποίο μπορεί κανείς είτε να βρει δωρεάν, είτε να αγοράσει αντικείμενα που τον ενδιαφέρουν, γλιτώνοντας κόπο και χρόνο.
- Δίνει τη δυνατότητα στον καθένα να φτιάξει τους δικούς του σκιαστές (shaders).
- Υποστηρίζει τις βασικές βιβλιοθήκες τρισδιάστατων γραφικών OpenGL και DirectX.
- Παρέχει ενσωματωμένο Physics Engine παρέχοντας συστατικά τα οποία μεταχειρίζονται την προσομοίωση φυσικής αυτόματα.
- Υπάρχει τεράστια κοινότητα στο forum της Unity οπότε μπορεί κανείς να βρει απαντήσεις για πολλά προβλήματα.
- Δίνει τη δυνατότητα να κάνουμε άμεσο χτίσιμο του project σε οποιαδήποτε πλατφόρμα σε αντίθεση με άλλες μηχανές.

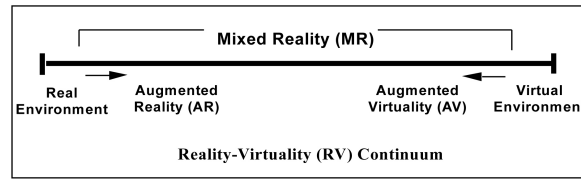
2.7 Επικοινωνία Ανθρώπου Υπολογιστή (HCI for Games)

Τα βιντεοπαιχνίδια είναι ένας πάρα πολύ δημοφιλής και επιτυχημένος τύπος διαδραστικού λογισμικού. Η αλληλεπίδραση μεταξύ χρηστών και υπολογιστών γίνεται στο επίπεδο της διεπαφής χρήστη (user interface), μέσω κατάλληλου λογισμικού και υλικού. Ένας βασικός στόχος της αλληλεπίδρασης ανθρώπου-υπολογιστή είναι να βελτιώσει την επικοινωνία μεταξύ χρηστών και υπολογιστών, μέσω της ορθής σχεδίασης εύχρηστων και εργονομικών εφαρμογών, προσαρμοσμένων στις ανθρώπινες ανάγκες. Βασικές αρχές που πρέπει να τηρούν οι σχεδιαστές διεπαφών είναι η απλότητα, η περιγραφικότητα και η ταχύτητα. Πρέπει να είναι φιλικά προς τον χρήστη, καθώς αποτελούν πόλο έλξης τόσο έμπειρων όσο και αρχάριων χρηστών. Επιπλέον, θα πρέπει να έχουν ανάδραση πληροφορίας για κάθε πιθανή ενέργεια που κάνει ο χρήστης, σταθερότητα, μικρό φόρτο απομνημόνευσης πληροφοριών προς τον χρήστη (ένας μέσος άνθρωπος μπορεί να απομνημονεύσει 7(+2) τμήματα πληροφορίας στην βραχυπρόθεσμη μνήμη του), καθώς και αποτροπή και χειρισμό πιθανών σφαλμάτων [16]. Βασική κινητήρια δύναμη στο σχεδιασμό τους είναι η απόδοση των χρηστών, η ικανοποίηση και η καινοτομία. Οι παίκτες περιμένουν καινούριες ιδέες, και ιδιαίτερα χαρακτηριστικά γνωρίσματα που δεν έχουν ξαναδεί ποτέ και τα οποία θα τους βοηθήσουν να παίξουν με πιο αποδοτικούς και ενδιαφέροντες τρόπους. Το αποτέλεσμα είναι τα παιχνίδια να έχουν γίνει οι βασικοί αποδέκτες της υιοθέτησης νέων τεχνολογιών και καινοτομιών στον τομέα του HCI [17].

2.8 Συνέχεια Πραγματικότητας-Εικονικότητας (Reality-Virtuality Continuum)

Η ευρέως διαδεδομένη άποψη ενός εικονικού περιβάλλοντος VR είναι αυτή στην οποία ο συμμετέχων παρατηρητής είναι εντελώς εμβυθισμένος σε έναν εντελώς εικονικό κόσμο, και ο οποίος μπορεί ή δεν μπορεί να μιμηθεί τις ιδιότητες ενός πραγματικού περιβάλλοντος, που είτε υπάρχει είτε είναι φανταστικό. Έτσι, δημιουργείται ένα κόσμος που υπερβαίνει τα όρια της φυσικής πραγματικότητας στο οποίο οι φυσικοί νόμοι που διέπουν τη βαρύτητα, τον χρόνο και τις ιδιότητες των υλικών δεν ισχύουν. Σε αντίθεση, ένα αυστηρά πραγματικό περιβάλλον σαφώς πρέπει να περιορίζεται από τους νόμους της φυσικής. Έτσι, λοιπόν, αντί να αναφέρουμε τις δύο έννοιες σαν αντίθετες, είναι πιο βολικό να τοποθετηθούν στις δύο άκρες του παρακάτω γραφήματος γνωστό και ως Reality-Virtuality Continuum (R-V) (σχήμα 2.19).

Το γράφημα Reality - Virtuality (R-V) Continuum αναπαριστά μια συνεχή κλίμακα ανάμεσα στο εντελώς πραγματικό και στο εντελώς εικονικό, με όλες τις δυνατές παραλλαγές των δύο να βρίσκονται στην κλίμακα της μεικτής πραγματικότητας. Έτσι, καθώς το πραγματικό πηγαίνει



Σχήμα 2.19: R-V Continuum

προς το εικονικό, με τις τεχνολογίες επαυξημένης πραγματικότητας να ενσωματώνουν ψηφιακό περιεχόμενο σε φυσικούς χώρους, έτσι και το εικονικό κινούμενο προς το πραγματικό, γίνεται ολόένα και περισσότερο συνδεδεμένο με τους φυσικούς χώρους και το παρόν [18].

2.8.1 Επαυξημένη Πραγματικότητα (Augmented Reality)

Η Επαυξημένη πραγματικότητα (Augmented Reality ή AR) είναι ένα ταχέως αναπτυσσόμενο ερευνητικό πεδίο στον χώρο της εικονικής πραγματικότητας και χρησιμοποιείται για να περιγράψει μια άμεση ή έμμεση άποψη του πραγματικού κόσμου του οποίου τα στοιχεία επαυξάνονται μέσω αισθητήριων εισόδων (sensory input), που δημιουργούνται από υπολογιστή, όπως ήχος ή γραφικά και οι οποίες συνδυάζονται με την εικόνα του κόσμου που δεχόμαστε από τις αισθήσεις μας (όραση, ακοή, αφή κλπ). Σχετίζεται με μια γενικότερη ιδέα που ονομάζεται διαμεσολαβημένη πραγματικότητα (mediated reality), κατά την οποία μια οπτική της πραγματικότητας τροποποιείται (πιθανώς και να ελλατώνεται αντί να επαυξάνεται) από υπολογιστή. Το πρώτο παράδειγμα επαύξησης της πραγματικότητας με επιπλέον πληροφορία ήταν η εφαρμογή των περιφημων HUDs (Head-up displays) στα μαχητικά αεροσκάφη, όπου ο πιλότος μπορούσε να δει πάνω στο τζάμι του χειριστηρίου του κάποιες βασικές παραμέτρους της πτήσης, όπως η ταχύτητα, το υψόμετρο, ο στόχος κ.ά., χωρίς να χρειάζεται να μετακινήσει το κεφάλι του για να κοιτάξει τα ανάλογα όργανα (εξ' ου και το όνομα head-up) και χωρίς ταυτόχρονα να χάνει την οπτική επαφή με το περιβάλλον [19]. Τομείς στους οποίους χρησιμοποιείται η επαυξημένη πραγματικότητα είναι οι: αρχαιολογία, αρχιτεκτονική, εμπόριο, εκπαίδευση, βιντεοπαιχνίδια, βιομηχανικός σχεδιασμός, ιατρική, στρατιωτική χρήση, πλοήγηση, τουρισμός κ.ά. Εφαρμογές επαυξημένης πραγματικότητας σχεδιάζονται είτε για υπολογιστές με κάποιο Head Mount Display, είτε για συσκευές χειρός όπως κινητές συσκευές, τάμπλετ ή προσωπικούς ψηφιακούς βοηθούς (PDAs)

2.8.2 Επαυξημένη Εικονικότητα (Augmented Virtuality)

Η επαυξημένη εικονικότητα (Augmented Virtuality) αναφέρεται στην συγχώνευση αντικειμένων του πραγματικού κόσμου σε εικονικούς κόσμους. Αποτελεί ενδιάμεση περίπτωση του

σχήματος 2.20, αλλά κυρίως αναφέρεται σε εικονικούς χώρους, όπου φυσικά στοιχεία όπως φυσικά αντικείμενα ή άτομα ενσωματώνονται δυναμικά και μπορούν να αλληλεπιδράσουν με τον εικονικό κόσμο σε πραγματικό χρόνο. Μια τέτοια ενσωμάτωση επιτυγχάνεται με τη χρήση διάφορων τεχνικών. Στην ουσία, η εικονική πραγματικότητα αποτελεί την αντίστροφη διαδικασία της επαυξημένης πραγματικότητας, όπου τώρα το 'φόντο' είναι ένας εικονικός κόσμος με κάποια πραγματικά αντικείμενα μέσα σε αυτόν [18].

2.8.3 Μεικτή Πραγματικότητα (Mixed Reality)

Η μεικτή πραγματικότητα (Mixed Reality) είναι η συγχώνευση πραγματικού και εικονικού κόσμου για την παραγωγή νέων περιβάλλοντων και οπτικοποιήσεων, όπου πραγματικά και ψηφιακά αντικείμενα συνυπάρχουν και αλληλεπιδρούν μεταξύ τους σε πραγματικό χρόνο. Η συγχωνευμένη πραγματικότητα δεν υπάρχει μόνο σε πραγματικό ή εικονικό περιβάλλον, αλλά είναι ένας συνδυασμός πραγματικότητας και εικονικής πραγματικότητας. Το HoloLens της Microsoft αποτελεί ένα τέλειο παράδειγμα Mixed Reality, όπου ο χρήστης μπορεί να πλοηγηθεί μέσα σε πραγματικό κόσμο με τη χρήση εικονικών αντικειμένων, αλληλεπιδρώντας μαζί τους σαν να είναι πραγματικά.

2.8.4 Εικονικά Περιβάλλοντα (Virtual Environments)

Η Εικονική Πραγματικότητα χρησιμοποιεί ηλεκτρονικούς υπολογιστές, για να δημιουργήσει και να προσομοιώσει υπαρκτά ή μη περιβάλλοντα, επιτρέποντας στον χρήστη να αλληλεπιδράσει με αυτά. Στην ουσία δημιουργείται η ψευδαίσθηση ότι βρίσκεται όντως εκεί και ότι μπορεί να κινηθεί ελεύθερα, όπως θα έκανε και στον πραγματικό κόσμο. Βασικός παράγοντας για να δημιουργηθεί η ψευδαίσθηση του πραγματικού, είναι να προσομοιωθούν όλες οι αισθήσεις μαζί. Αυτό είναι πιο δύσκολο από όσο ακούγεται, αφού οι αισθήσεις και το μυαλό μας έχουν εξελιχθεί με τέτοιο τρόπο, ώστε να μας παρέχουν μια τέλεια συγχρονισμένη εμπειρία. Έτσι λοιπόν για να είναι πετυχημένη η εμβύθιση (Immersion) ενός χρήστη σε ένα περιβάλλον εικονικής πραγματικότητας, είναι σημαντικό να απομονωθεί ο χρήστης και οι αισθήσεις του από τον πραγματικό κόσμο, επικαλύπτοντας τα ερεθίσματα του πραγματικού κόσμου με αντίστοιχα εικονικά, φτιαγμένα από το σύστημα της εικονικής πραγματικότητας. Πιο συγκεκριμένα, το σύστημα θα πρέπει να παρέχει στερεοσκοπική εικόνα, δηλαδή δύο εικόνες από διαφορετική οπτική γωνία, μία για κάθε μάτι του χρήστη, οι οποίες επικαλύπτονται, δημιουργώντας στον χρήστη την αίσθηση του βάθους στο χώρο. Παράλληλα, θα πρέπει να υπάρχει στερεοσκοπικός ήχος ο οποίος δίνει την αίσθηση στο χρήστη να αντιλαμβάνεται τον εικονικό χώρο που τον περιβάλλει με πολύ φυσικό τρόπο. Τέλος, η αφή μπορεί να χρησιμοποιηθεί με κατάλληλες συσκευές είτε για να μπορεί ο χρήστης να νιώθει τον κόσμο, είτε για να καθοδηγήσουμε το χρήστη, διευκολύνοντάς τον στην

εκτέλεση κάποιων συγκεκριμένων ενεργειών. Συμπεραίνουμε λοιπόν, ότι ο σωστός συγχρονισμός των αισθήσεων του χρήστη σε ένα σύστημα εικονικής πραγματικότητας, είναι εφικτό να δημιουργήσει την αίσθηση του ρεαλιστικού [20].

2.8.5 Head Mounted Displays

Τα Head Mounted Displays είναι συσκευές απεικόνισης, οι οποίες τοποθετούνται είτε στο κεφάλι είτε σε κάποιο κράνος και έχουν μικρές οπτικές οθόνες απεικόνισης είτε στο ένα μάτι (monocular HMD) είτε και στα δύο (binocular HMD). Διαφέρουν ως προς τον τρόπο απεικόνισης, καθώς άλλα μπορούν να προβάλλουν εικόνες που έχουν δημιουργηθεί στον υπολογιστή (computer generated image) και άλλα εικόνες από τον πραγματικό και τον εικονικό κόσμο μαζί (βλέπε augmented reality-mixed reality). Στη δεύτερη περίπτωση ο συνδυασμός πραγματικού κόσμου με εικονικό γίνεται με την προβολή της δημιουργημένης σε υπολογιστή εικόνας πάνω σε μερικώς ανακλαστικό καθρέπτη, βλέποντας απευθείας τον πραγματικό κόσμο. Επιπλέον, τα περισσότερα HMDs περιέχουν και έναν Head Tracker έτσι ώστε το σύστημα να παρακολουθεί τις κινήσεις του χρήστη. Για παράδειγμα αν ο χρήστης μετακινήσει το κεφάλι του αριστερά, οι εικόνες στην οθόνη του HMD αλλάζουν, δίνοντας την εντύπωση ότι όντως κοιτάει σε διαφορετική κατεύθυνση στο εικονικό περιβάλλον. Παρακάτω γίνεται μια σύντομη αναφορά γνωστών Head Mounted Displays που χρησιμοποιούν είτε augmented reality είτε mixed reality είτε virtual reality [21].

- **Mimic Sim AR Surgeon:** Είναι η πρώτη ρομποτική χειρουργική τεχνολογία προσομοίωσης που επιτρέπει στους εκπαιδευόμενους να χειριστούν τρισδιάστατα εικονικά ρομποτικά εργαλεία ως έναν τρόπο για να αλληλεπιδρούν με 3D ενδοσκοπικό βίντεο μιας πραγματικής χειρουργικής επέμβασης.
- **AR “Smart Helmet”:** Έχει σχεδιαστεί από τη DAQRI για χρήση από μηχανικούς, εργάτες και οποιονδήποτε εργάζεται σε βιομηχανικό περιβάλλον βελτιώνοντας την επικοινωνία, την ασφάλεια και την παραγωγικότητα σε εργοτάξια υπερθέτοντας οδηγίες και άλλο ψηφιακό περιεχόμενο πάνω σε πραγματικό βιομηχανικό εξοπλισμό.
- **Google Glass:** Σχεδιασμένο από την Google επιτρέπει στον χρήστη να ελέγχει την συσκευή, διαμέσου μιας διεπαφής που μοιάζει με χρονολόγιο και εμφανίζεται στην οθόνη. Σύροντας με το δάκτυλο προς τα πίσω εμφανίζονται τρέχοντα γεγονότα, όπως ο καιρός και σύροντας προς τα εμπρός δείχνει προηγούμενα γεγονότα, όπως τηλεφωνικές κλήσεις, φωτογραφίες, κ.ά.
- **Pinlight:** Είναι η πρώτη LCD συσκευή η οποία είναι ελαφρύτερη και χρησιμοποιεί απλούστερη τεχνολογία σε σχέση με τα συμβατικά AR HUDs. Στόχος της εταιρείας είναι να

δημιουργήσει ένα ελαφρύ, υψηλής ενεργειακής απόδοσης, καθώς και μεγαλύτερης οθόνης υπολογιστή που να φοριέται.

- **Sony SmartEyeglass:** Το SmartEyeglass προσφέρει see-through γυαλιά με ενσύρματο χειριστήριο (το οποίο συνδέεται με συμβατά smartphones μέσω Bluetooth) υπερθέτοντας στρώματα περιεχομένου, όπως ειδοποιήσεις και άλλες σχετικές πληροφορίες πάνω σε πραγματικό οπτικό περιεχόμενο μέσα στο φυσικό πεδίο όρασης του χρήστη.
- **Skully AR-1 motorcycle helmet:** Είναι σχεδιασμένο για να ενισχύσει την ασφάλεια των μοτοσικλετιστών περιλαμβάνοντας χαρακτηριστικά όπως κάμερα οπισθοπορείας, ταχύμετρο, καθώς και GPS.
- **Sony PlayStation VR(Project Morpheus):** Γνωστό με το κωδικό όνομα Project Morpheus είναι ένα headset εικονικής πραγματικότητας κατασκευασμένο από την Sony Computer Entertainment. Είναι σχεδιασμένο έτσι ώστε να είναι συμβατό με το Playstation 4 και το Playstation Vita. Η ακριβής ημερομηνία έκδοσης δεν έχει ανακοινωθεί αλλά αναμένεται να είναι μέσα στο 2016.
- **Oculus Rift:** Το Oculus Rift αποτελεί ένα headset εικονικής πραγματικότητας το οποίο δημιούργησε το 'ξέσπασμα' στην αγορά των Head Mounted Displays. Ακολουθεί τις κινήσεις του κεφαλιού και προβάλλει τρισδιάστατες εικόνες στις stereo οθόνες του.
- **Samsung Gear VR:** Θα αποτελέσει ένα από τα καλύτερα all around και φιλικά προς τον χρήστη headsets στην αγορά. Θα είναι συμβατό με Samsung Galaxy S6, S6 Edge, S6 Edge+, Note 5 και η εμπορική του έκδοση αναμένεται να κυκλοφορήσει στο τέλος του 2015.
- **Microsoft HoloLens:** Αποτελεί ένα Mixed Reality Headset το οποίο αναμιγνύει στοιχεία του πραγματικού κόσμου με εικονικά στοιχεία γνωστά ως holograms δίνοντας στο χρήστη τη δυνατότητα να αλληλεπιδράσει με αυτά.
- **Google Cardboard:** Τοποθετώντας το κινητό μέσα σε ένα cardboard και δένοντάς το στο κεφάλι, δίνει την δυνατότητα στον χρήστη να βιώσει μια φτηνή εμπειρία εικονικής πραγματικότητας με κόστος 29\$.

Μερικά ακόμα headsets τα οποία είτε κυκλοφορούν είτε αναμένεται να κυκλοφορήσουν στο εμπόριο είναι : HTC Vive, FOVE VR, Zeiss VR One, Avedant Glyph, Razer OSVR, Archos VR Headset, Freefly VR headset [22].

2.8.6 Head Tracking

Το Head Tracking χρησιμοποιείται στα εικονικά περιβάλλοντα, όπου απαιτείται ο προσανατολισμός και η θέση ενός πραγματικού φυσικού αντικειμένου. Για να καθορίσουμε ένα τρισδιάστατο σημείο απαιτείται η θέση μετάβασης, η οποία υπολογίζεται από τις καρτεσιανές συντεταγμένες x, y, z . Ωστόσο, πολλές εφαρμογές εικονικής πραγματικότητας μεταχειρίζονται ολόκληρα αντικείμενα και αυτό απαιτεί να καθοριστεί ο προσανατολισμός από τρεις γωνίες γνωστές ως pitch, roll, yaw. Συνεπώς, έξι βαθμοί ελευθερίας (DOF) είναι το ελάχιστο που απαιτείται για την πλήρη περιγραφή της θέσης ενός αντικειμένου στον τρισδιάστατο χώρο. Παρακάτω παρουσιάζονται τα βασικά χαρακτηριστικά των Head trackers, τα οποία χρησιμεύουν και σαν μετρικές για την αξιολόγηση και σύγκριση του συστήματος [23].

- **Ανάλυση (Resolution):** Μετράει την ακρίβεια με την οποία ένα σύστημα μπορεί να εντοπίσει μια θέση.
- **Ακρίβεια (Accuracy):** Η περιοχή εντός της οποίας μια θέση που αναφέρθηκε είναι σωστή.
- **Απόκριση Συστήματος (System Responsiveness):** Αποτελείται από:
 - Ρυθμός δειγματοληψίας (Sample rate): Είναι ο ρυθμός με τον οποίο οι αισθητήρες ελέγχονται για δεδομένα, συχνά εκφραζόμενος σαν συχνότητα (frequency).
 - Ρυθμός Δεδομένων (Data rate): Είναι ο αριθμός των υπολογισμένων θέσεων ανά δευτερόλεπτο, συχνά εκφραζόμενος σαν συχνότητα.
 - Ρυθμός Ανανέωσης (Update rate): Ο ρυθμός με τον οποίο το σύστημα αναφέρει νέες συντεταγμένες θέσης στον κεντρικό υπολογιστή, επίσης συνήθως αναφέρεται ως συχνότητα.
 - Λανθάνων Χρόνος (Latency): Είναι επίσης γνωστός ως υστέρηση (lag) και είναι η καθυστέρηση μεταξύ της κίνησης του ανιχνευμένου αντικειμένου και της αναφοράς της νέας θέσης. Μετριέται σε χιλιοστά του δευτερολέπτου (ms).

Κεφάλαιο 3

Τεχνολογική Βάση

3.1 Εισαγωγή

Στο κεφάλαιο αυτό αρχικά γίνεται μια σύντομη περιγραφή του σχεδιαστικού εργαλείου 3DS που χρησιμοποιήθηκε για κάποιες βασικές ενέργειες. Στη συνέχεια, γίνεται εκτενής αναφορά στη μηχανή σχεδίασης παιχνιδιών Unity 3D η οποία επιλέχθηκε να χρησιμοποιηθεί, αναλύοντας όλα τα βασικά της μέρη που είναι απαραίτητα για την δόμηση μιας ολοκληρωμένης εφαρμογής ή παιχνιδιού. Τέλος, γίνεται περιγραφή του Oculus Rift DK2 που χρησιμοποιήθηκε για την εμπειρία του εικονικού περιβάλλοντος, αναλύοντας όλα τα τεχνικά του χαρακτηριστικά.

3.2 3DS

Το 3ds Max της Autodesk είναι ένα πλήρες επαγγελματικό λογισμικό για τη δημιουργία τρισδιάστατων κινούμενων σχεδίων (3D animation), μοντέλων, παιχνιδιών και εικόνων, κατασκευασμένο από την Autodesk Media and Entertainment. Έχει δυνατότητες μοντελοποίησης και ευέλικτη αρχιτεκτονικής plugin. Είναι πολύ συχνά χρησιμοποιούμενο από σχεδιαστές βιντεοπαιχνιδιών, από εμπορικά τηλεοπτικά studios, αρχιτεκτονικά studios οπτικοποίησης καθώς επίσης χρησιμοποιείται και για εφέ ταινιών. Εκτός από τα εργαλεία μοντελοποίησης και κινούμενων σχεδίων που διαθέτει, έχει επιπλέον χαρακτηριστικά όπως σχιαστές (shaders), δυναμικές προσομοιώσεις dynamic simulation, σύστημα σωματιδίων (particle systems), αποτύπωση αντανάκλασης radiosity, normal map, global illumination, ένα προσαρμόσιμο user interface καθώς και δική του γλώσσα scripting.

Στα πλαίσια της πτυχιακής αυτής εργασίας, χρησιμοποιήθηκε το εργαλείο αυτό για να γίνει κάποια επεξεργασία στο τρισδιάστατο μοντέλο του snowboarder που βρίσκεται στο κεντρικό

μενού, καθώς και ορισμένες τροποποιήσεις στο μοντέλο του σκιέρ. Επίσης, καρέ καρέ (frame by frame) δημιουργήσαμε τα παρακάτω animations του σκιέρ:

- Αριστερή στροφή (left turn)
- Δεξιά στροφή (right turn)
- Ακίνητος (Idle)
- Σκι (Ski)

Εξαιτίας του γεγονότος ότι το παιχνίδι είναι First Person Controller, τα animations των στροφών δεν είναι ορατά κατά τη διάρκεια του παιχνιδιού. Μπορούν όμως να φανούν και να αξιοποιηθούν σε περίπτωση μελλοντικής μετατροπής του παιχνιδιού για άλλη πλατφόρμα, έχοντας τον παίχτη Third Person Controller.

3.3 Unity 3D

Η Unity 3D είναι μια ευέλικτη και ισχυρή πλατφόρμα ανάπτυξης δισδιάστατων και τρισδιάστατων παιχνιδιών καθώς και διαδραστικών εμπειριών για υπολογιστές, κινητές συσκευές, κονσόλες, καθώς και για το διαδίκτυο. Μέσω του έξυπνου και προσαρμόσιμου περιβάλλοντος εργασίας, καθώς και της πληθώρας εργαλείων που παρέχει, μπορεί κανείς να δημιουργήσει γρήγορα και παραγωγικά ροές εργασίας ελαχιστοποιώντας τον χρόνο και βοηθώντας στην αποσφαλμάτωση. Παρέχει εύκολη προσπέλαση των αντικειμένων του πρότζεκτ, ιεραρχική δομή της γεωμετρίας καθώς και άμεση προεπισκόπηση της εφαρμογής μας [24].

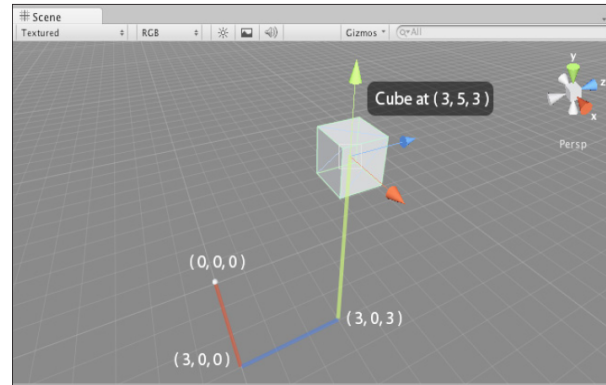
3.4 Δομή & Αρχιτεκτονική της Unity 3D

Η αρχιτεκτονική της Unity 3D βασίζεται στις υπομονάδες της. Για να μπορέσουμε να χρησιμοποιήσουμε με σωστό τρόπο τις υπομονάδες των αντικειμένων, είναι απαραίτητο να κατανοήσουμε αυτή την αρχιτεκτονική πάνω στην οποία είναι χτισμένη αυτή η μηχανή παιχνιδιών [25].

3.4.1 Συντεταγμένες (Coordinates)

Σε ένα δισδιάστατο επίπεδο, κάθε σημείο καθορίζεται μοναδικά από ένα ζεύγος αριθμητικών συντεταγμένων X(οριζόντιος άξονας), Y(κάθετος άξονας). Στις τρεις διαστάσεις και κατ'επέκταση σε όλες τις τρισδιάστατες εφαρμογές της Unity 3D υπάρχει ένας τρίτος άξονας Z, ο οποίος αναπαριστά το βάθος με αποτέλεσμα κάθε σημείο να αναπαρίσταται από μια τριάδα

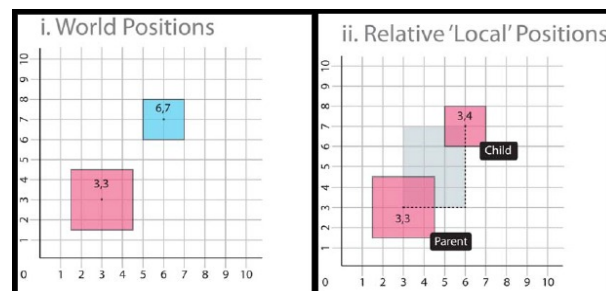
(X,Y,Z). Έτσι, ένας κύβος που βρίσκεται στο σημείο (3,5,3) στον τρισδιάστατο κόσμο, σημαίνει ότι απέχει 3 μονάδες από το σημείο 0 στον άξονα X, 5 μονάδες από τον άξονα Y και 3 μονάδες από τον άξονα Z. Αυτό το format είναι γνωστό ως καρτεσιανό σύστημα συντεταγμένων (σχήμα 3.1).



Σχήμα 3.1: Coordinate system

3.4.2 Τοπικό Σύστημα Συντεταγμένων (Local Space vs World Space)

Σε οποιονδήποτε τρισδιάστατο κόσμο, υπάρχει ένα σημείο προέλευσης, συχνά αναφερόμενο ως origin ή world zero καθώς αναπαρίσταται στη θέση (0,0,0). Όλες οι θέσεις των αντικειμένων στους τρισδιάστατους κόσμους έχουν ως σημείο αναφοράς το μηδέν. Ωστόσο, για να γίνουν τα πράγματα απλούστερα, χρησιμοποιούμε local space για να ορίσουμε θέσεις αντικειμένων σε σχέση με άλλα αντικείμενα (σχήμα 3.2).



Σχήμα 3.2: World Space vs Local Space

3.4.3 Διανύσματα (Vectors)

Όπως και στις 2 διαστάσεις, τα τρισδιάστατα διανύσματα (3D Vectors) είναι απλά γραμμές στον τρισδιάστατο χώρο οι οποίες έχουν κάποια κατεύθυνση και κάποιο μήκος. Τα διανύσματα

μπορούν να μετακινηθούν στον χώρο χωρίς να αλλάζει η τιμή και η κατεύθυνση τους. Είναι πολύ χρήσιμα στις μηχανές παιχνιδιών καθώς μας επιτρέπουν να υπολογίσουμε αποστάσεις, σχετικές γωνίες μεταξύ αντικειμένων και διευθύνσεις αντικειμένων.

3.4.4 Κάμερες (Cameras)

Οι κάμερες είναι απαραίτητες στον τρισδιάστατο χώρο καθώς δρουν ως παράθυρα για την οθόνη. Μπορούν να τοποθετηθούν σε οποιοδήποτε σημείο του χώρου, συνδέοντάς τες με χαρακτήρες ή αντικείμενα ως μέρος του σεναρίου του παιχνιδιού. Επίσης, είναι δυνατό να υπάρχουν πολλές κάμερες σε μια σκηνή, με την υπόθεση ότι μια μόνο κύρια κάμερα θα κάνει render αυτό που βλέπει ο παίχτης. Αυτός είναι και ο λόγος που η Unity παρέχει μια κύρια κάμερα με την έναρξη μιας νέας σκηνής.

3.4.5 Πολύγωνα, Ακμές, Κορυφές & Πλέγματα (Polygons, Edges, Vertices & Meshes)

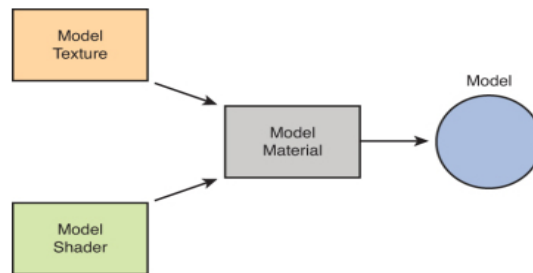
Όλα τα αντικείμενα είναι αποκλειστικά κατασκευασμένα από διασυνδεδεμένα δισδιάστατα σχήματα γνωστά ως πολύγωνα. Κατά την εισαγωγή μοντέλων από μια εφαρμογή μοντελοποίησης, η Unity μετατρέπει τα πολύγωνα σε τρίγωνα πολυγώνου polygon triangles γνωστά ως όψεις (faces). Συνδυάζοντας πολλά συνδεδεμένα πολύγωνα οι τρισδιάστατες εφαρμογές μοντελοποίησης μας επιτρέπουν να κατασκευάσουμε πολύπλοκα σχήματα γνωστά ως πλέγματα. Οι όψεις με τη σειρά τους δημιουργούνται από τις συνδεδεμένες ακμές (edges). Τα σημεία στα οποία αυτές οι ακμές συναντιούνται, είναι γνωστά ως κορυφές (vertices). Γνωρίζοντας αυτά τα σημεία, οι μηχανές παιχνιδιών μπορούν να κάνουν υπολογισμούς γνωστούς ως συγκρούσεις (collisions). Για τους προγραμματιστές είναι ζωτικής σημασίας να κατανοήσουν τη σημασία του αριθμού των πολυγώνων (polygon count), καθώς όσο μεγαλύτερος είναι, τόσο περισσότεροι πόροι του υπολογιστή καταναλώνονται.

3.4.6 Μετασχηματισμός (Transform)

Κάθε αντικείμενο σε μια σκηνή έχει έναν μετασχηματισμό. Χρησιμοποιείται για να αποθηκεύσει και να μεταχειριστεί τη θέση, την περιστροφή και την κλιμάκωση ενός αντικειμένου. Κάθε μετασχηματισμός μπορεί να έχει έναν πατέρα (parent), ο οποίος επιτρέπει στο χρήστη να μετακινήσει το αντικείμενο, να το περιστρέψει και να το κλιμακώσει ιεραρχικά [26].

3.4.7 Υλικά, Υφές & Σκιαστές (Materials, Textures & Shaders)

Η εικόνα που φαίνεται παρακάτω δείχνει τη σχέση που έχουν τα υλικά, οι υφές και οι σκιαστές μεταξύ τους. Οι υφές και οι σκιαστές εφαρμόζονται στα materials. Τα materials με τη σειρά τους εφαρμόζονται πάνω στα μοντέλα (σχήμα 3.3). Με αυτό τον τρόπο η εμφάνιση ενός μοντέλου μπορεί να αλλάχθει ή να τροποποιηθεί γρήγορα και απλά χωρίς πολύ δουλειά.



Σχήμα 3.3: Materials, Textures and Shaders

3.4.8 Σωματίδια (Particle System)

Σε ένα τρισδιάστατο παιχνίδι, οι περισσότεροι χαρακτήρες, σκηνικά και στοιχεία γενικότερα αναπαρίστανται ως πλέγματα. Τα πλέγματα είναι ο ιδανικός τρόπος για να απεικονιστούν στερεά αντικείμενα με ένα καλά καθορισμένο σχήμα. Στα παιχνίδια όμως, υπάρχουν και άλλες οντότητες που είναι ρευστές και άυλες και ως εκ τούτου είναι δύσκολο να αναπαρασταθούν με πλέγματα. Για εφέ όπως μετακίνηση υγρών, καπνό, σύννεφων, φλογών και άλλων, ακολουθείται μια διαφορετική προσέγγιση στα γραφικά γνωστή ως particle system. Για τις ανάγκες του παιχνιδιού, χρησιμοποιήθηκε το σύστημα σωματιδίων για να αναπαρασταθεί το εφέ της σκόνης χιονιού (πούδρα) που αφήνει ένας σκιερ όταν αναπτύζει ταχύτητα.

3.4.9 Φυσική (Rigidbody Physics)

Η μηχανή φυσικής της Unity δίνει τη δυνατότητα στους προγραμματιστές να προσομοιώσουν αποκρίσεις του πραγματικού κόσμου πάνω σε αντικείμενα. Η Unity χρησιμοποιεί τη μηχανή Nvidia's PhysX, η οποία είναι μια δημοφιλή και ακριβής μηχανή φυσικής. Στις μηχανές παιχνιδιών δεν υπάρχει καμία υπόθεση ότι ένα αντικείμενο θα πρέπει να επηρεάζεται από την βαρύτητα, πρώτον διότι αυτό απαιτεί πολύ επεξεργαστική ισχύ και δεύτερον επειδή δεν υπάρχει λόγος να γίνει κάτι τέτοιο. Η μηχανή φυσικής χρησιμοποιεί το δυναμικό σύστημα Rigidbody για τη δημιουργία ρεαλιστικής κίνησης. Αυτό σημαίνει ότι αντί τα αντικείμενα να είναι στατικά, μπορούν να έχουν ιδιότητες όπως μάζα, βαρύτητα, ταχύτητα και τριβή. Όσο η επεξεργαστική

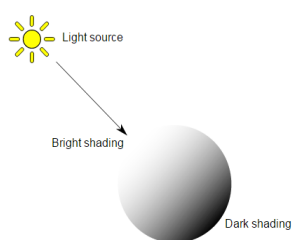
ισχύς αυξάνεται, τόσο το σύστημα Rigidbody physics εφαρμόζεται στα παιχνίδια, καθώς δίνει τη δυνατότητα για πιο ρεαλιστικές προσομοιώσεις.

3.4.10 Ανίχνευση Σύγκρουσης (Collision Detection)

Η ανίχνευση σύγκρουσης είναι ένας τρόπος να αναλύσουμε τις συγκρούσεις μεταξύ των αντικειμένων στον τρισδιάστατο κόσμο. Με την τοποθέτηση ενός Collider πάνω σε ένα αντικείμενο, δημιουργείται ένα αόρατο πλέγμα τριγύρω του, το οποίο είναι υπεύθυνο να εντοπίζει τυχόν συγκρούσεις με άλλους Colliders. Επιπλέον, υπάρχουν δύο ειδών συγκρουστές, οι πρωταρχικοί (Primitives) και τα πλέγματα (Meshes). Οι πρωταρχικοί συγκρουστές είναι απλά γεωμετρικά αντικείμενα όπως κύβοι, σφαίρες και κάψουλες και από άποψη υπολογιστικού κόστους είναι φτηνότεροι. Τα πλέγματα έχουν μεγαλύτερο υπολογιστικό κόστος, καθώς ακολουθούν ακριβώς το σχήμα του αντικειμένου στο οποίο έχουν εφαρμοστεί.

3.4.11 Φωτισμός (Lighting)

Για να υπολογιστεί η σκίαση ενός αντικειμένου, η Unity χρειάζεται να ξέρει την ένταση, τη διεύθυνση και το χρώμα του φωτός που πέφτει πάνω του (σχήμα 3.4) [27].

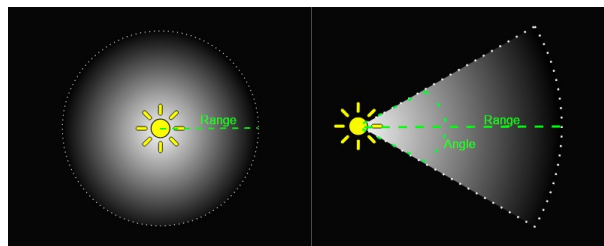


Σχήμα 3.4: Lighting

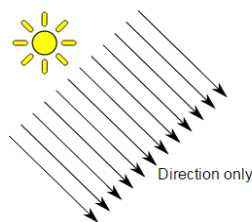
Αυτές οι ιδιότητες παρέχονται από τα αντικείμενα φωτός (Light Objects). Το βασικό χρώμα και η ένταση καθορίζονται με τον ίδιο τρόπο για όλα τα φώτα, αλλά η κατεύθυνση εξαρτάται από τον τύπο του φωτός που χρησιμοποιείτε. Επίσης, το φως μπορεί να μειωθεί με την απόσταση από την πηγή. Οι τέσσερις τύποι φωτός που είναι διαθέσιμοι στη Unity περιγράφονται παρακάτω.

- **Point Light:** Τοποθετείται σε ένα σημείο στο χώρο και στέλνει φως προς όλες τις κατευθύνσεις ομοιόμορφα. Η κατεύθυνση του φωτός που χτυπάει μια επιφάνεια είναι η γραμμή από το σημείο επαφής μέχρι το κέντρο της πηγής φωτός. Η ένταση μειώνεται με την απόσταση από την πηγή του φωτός, φτάνοντας στο μηδέν μετά από ένα συγκεκριμένο εύρος (σχήμα 3.5). Αυτό το είδος φωτός είναι ιδανικό για προσομοίωση λαμπών και άλλων τοπικών πηγών φωτός.

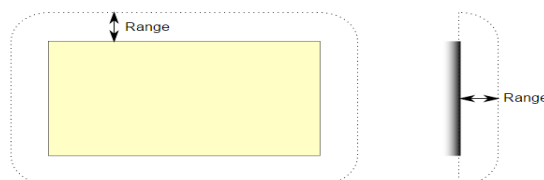
- **Spot Light:** Όπως και το Point Light έχει και αυτό συγκεκριμένη θέση και εύρος, γύρω από το οποίο η ένταση του φωτός μειώνεται. Ωστόσο, το φως αυτό περιορίζεται σε μια γωνία έχοντας ως αποτέλεσμα το φωτισμό μιας περιοχής σε σχήμα κωνικό (σχήμα 3.5). Χρησιμοποιείται γενικά για τεχνητές πηγές φωτός όπως φακούς και προβολείς.
- **Directional Light:** Εκπροσωπούν μεγάλες, απομακρυσμένες πηγές που προέρχονται από μια θέση έξω από την περιοχή του κόσμου του παιχνιδιού. Σε μια ρεαλιστική σκηνή, μπορούν να χρησιμοποιηθούν για την προσομοίωση του ήλιου ή της σελήνης. Σε ένα αφηρημένο κόσμο του παιχνιδιού, μπορεί να είναι ένας χρήσιμος τρόπος για πειστική σκίαση σε αντικείμενα χωρίς να προσδιορίζει ακριβώς από που προέρχεται το φως (σχήμα 3.6).
- **Area Light:** Το φως εκπέμπεται προς όλες τις κατευθύνσεις, αλλά μόνο από τη μία πλευρά του ορθογωνίου και μειώνεται σε ένα καθορισμένο εύρος (σχήμα 3.7). Μπορεί να χρησιμοποιηθεί για τη δημιουργία ενός ρεαλιστικού φωτός σε δρόμο καθώς και για τον εσωτερικό φωτισμό σπιτιού.



Σχήμα 3.5: *Αριστερά: Point Light Δεξιά: Spot Light*



Σχήμα 3.6: *Directional Light*



Σχήμα 3.7: *Area Light*

3.4.12 Ήχος (Audio)

Ο ήχος στη Unity αποτελείται από δύο βασικές υπομονάδες οι οποίες περιγράφονται παρακάτω.

- **Audio Source:** Αποτελεί την πηγή του ήχου και αναπαράγει ένα ηχητικό κλιπ το οποίο μπορεί να είναι είτε δισδιάστατο είτε τρισδιάστατο είτε μείγμα (SpatialBlend). Το συστατικό αυτό περιέχει ρυθμίσεις σχετικά με την ένταση του ήχου, την επεναληψιμότητά του, τον τόνο του, την προτεραιότητα του σε σχέση με άλλες πηγές, καθώς και πληθώρα άλλων ρυθμίσεων και εφέ [28].
- **Audio Listener:** Λειτουργεί στην ουσία σαν συσκευή μικροφώνου, λαμβάνοντας είσοδο από της πηγές ήχου της σκηνής και αναπαράγει ήχο από τα ηχεία του υπολογιστή. Πρέπει να υπάρχει πάντα ένας Listener ενεργοποιημένος μια δεδομένη χρονική στιγμή κατά τη διάρκεια του παιχνιδιού. Συνήθως τοποθετείται στην κύρια κάμερα καθώς από εκεί ο χρήστης ακουεί και βλέπει [29].

3.4.13 Διεπαφή Χρήστη (User Interface)

Η Unity παρέχει το νέο σύστημα διεπαφής χρήστη (UI) το οποίο επιτρέπει στον χρήστη να δημιουργήσει διαισθητικά και γρήγορα ευπαρουσιάστες γραφικές διεπαφές. Το σύστημα διεπαφής χρήστη αποτελείται από έναν καμβά (Canvas), μέσα στον οποίο τοποθετούνται όλα τα (UI) στοιχεία. Τα στοιχεία αυτά μπορεί να είναι κείμενα, εικόνες κ.ά, καθώς και πιο διαδραστικά στοιχεία όπως κουμπιά και Sliders. Υπερτερεί έναντι του παλιότερου συστήματος γραφικών GUI της Unity, καθώς μπορεί κανείς να σχεδιάσει γρηγορότερα και πιο αποδοτικά [30].

3.4.14 Κινούμενα Σχέδια (Animations)

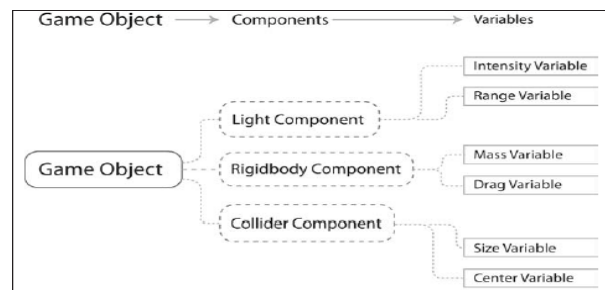
Το σύστημα κινούμενων σχεδίων (Animation) της Unity δίνει τη δυνατότητα στον χρήστη να δημιουργήσει και να διαμορφώσει τα δικά του κινούμενα σχέδια για την κατασκευή δημιουργικών συμπεριφορών. Επιπλέον η Unity επιτρέπει στο χρήστη να εισάγει κινούμενα σχέδια από κάποιο άλλο πρόγραμμα σχεδίασης γραφικών όπως το 3D Studio Max, που χρησιμοποιήθηκε στην παρούσα διπλωματική εργασία. Όταν λοιπόν εισαχθεί ένα αντικείμενο, τοποθετείται σε αυτό το συστατικό Animation, το οποίο δίνει τη δυνατότητα διαχείρισης των κινούμενων σχεδίων.

3.4.15 Πλοήγηση & Εύρεση Καλύτερου Μονοπατιού (Navigation & Pathfinding)

Το σύστημα πλοήγησης επιτρέπει στον χρήστη να δημιουργήσει χαρακτήρες που μπορούν να κινηθούν έξυπνα στον κόσμο του παιχνιδιού. Το σύστημα πλοήγησης χρησιμοποιεί πλέγματα πλοήγησης τα οποία δημιουργούνται αυτόματα από τη γεωμετρία της σκηνής. Δυναμικά εμπόδια τροποποιούν την πλοήγηση των χαρακτήρων κατά το χρόνο εκτέλεσης, και η δυνατότητα του off-mesh link μας επιτρέπει να δημιουργήσουμε συγκεκριμένες ενέργειες. Πάραλληλα μπορούν να ρυθμιστούν διάφοροι άλλοι παράμετροι που καθορίζουν τον τρόπο δημιουργίας του πλέγματος. Κατά την υλοποίηση της εργασίας αυτής υλοποιήθηκε το σύστημα πλοήγησης (Navigation & Pathfinding) της Unity για την εύρεση βέλτιστων μονοπατιών από τους αντιπάλους του παίχτη μας [31].

3.5 Βασικές Έννοιες της Unity 3D

Η Unity καθιστά τη διαδικασία παραγωγής μιας εφαρμογής απλή διαδικασία, δίνοντάς μια σειρά λογικών βημάτων για την οικοδόμηση κάθε πιθανού σεναρίου του παιχνιδιού. Με την καθιέρωση της έννοιας του αντικειμένου (GameObject), ο χρήστης είναι σε θέση να σπάσει τα μέρη του παιχνιδιού σε εύκολα διαχειρίσιμα αντικείμενα, τα οποία κατασκευάζονται από πολλά μεμονωμένα συστατικά (Components). Με την προσθήκη συστατικών προστίθεται λειτουργικότητα με ένα λογικό προοδευτικό τρόπο. Τα συστατικά με τη σειρά τους έχουν μεταβλητές (Variables) προσθέτοντας ιδιότητες στα συστατικά ή ρυθμίσεις που τα ελέγχουν. Με τη ρύθμιση αυτών των μεταβλητών, ο χρήστης έχει έναν ολοκληρωμένο έλεγχο της επίδρασης που έχουν τα συστατικά πάνω στα αντικείμενα. Το παρακάτω διάγραμμα απεικονίζει τις προαναφερθείσες έννοιες (σχήμα 3.8).



Σχήμα 3.8: Game Object-Components-Variables Overview

3.5.1 (Assets)

Αυτά αποτελούν τα δομικά στοιχεία όλων των πρότζεκτ της Unity 3D, και αναφέρεται σε όλα τα αρχεία που θα χρησιμοποιήσει κανείς για να δημιουργήσει ένα παιχνίδι όπως μοντέλα, υφές, ήχους κ.ά. Αυτός είναι ο λόγος που σε κάθε φάκελο που περιέχει πρότζεκτ της Unity, όλα τα αρχεία που χρησιμοποιούνται είναι αποθηκευμένα σε ένα φάκελο που ονομάζεται Assets.

3.5.2 Σκηνές (Scenes)

Στην Unity, οι σκηνές λειτουργούν ως επιμέρους επίπεδα, ή περιοχές του παιχνιδιού, αν και ορισμένοι προγραμματιστές δημιουργούν ολόκληρα παιχνίδια σε μία μόνο σκηνή, όπως παιχνίδια παζλ, φορτώνοντας δυναμικά περιεχόμενο μέσω κώδικα. Με την κατασκευή ενός παιχνιδιού σε πολλές σκηνές, ο προγραμματιστής είναι σε θέση να διανέμει τους χρόνους φόρτωσης και δοκιμής ξεχωριστά. Οποιαδήποτε στιγμή υπάρχει ανοιχτή κάποια σκηνή, είναι αυτή που εργαζόμαστε, καθώς δεν υπάρχει η δυνατότητα δύο σκηνές να λειτουργούν ταυτόχρονα.

3.5.3 Αντικείμενα (GameObjects)

Κάθε ενεργό αντικείμενο στην τρέχουσα ανοιχτή σκηνή ονομάζεται GameObject. Όλα τα GameObjects περιέχουν τουλάχιστον ένα συστατικό (Component) το οποίο είναι ο μετασχηματισμός (Transform). Ο μετασχηματισμός πληροφορεί την Unity για τη θέση, την περιστροφή και την κλιμάκωση ενός αντικειμένου μέσω των καρτεσιανών συντεταγμένων X,Y,Z. Μέσω κώδικα, το συστατικό μπορεί να καθορίσει τις συντεταγμένες του αντικειμένου. Έτσι, από αυτό το αρχικό συστατικό, είναι εφικτό να προστεθούν και άλλα Components στο αντικείμενο προσθέτοντας την απαιτούμενη λειτουργικότητα που επιθυμούμε.

3.5.4 Συστατικά (Components)

Τα Components υπάρχουν σε διάφορες μορφές. Χρησιμοποιούνται για τη δημιουργία συμπεριφοράς, καθορίζοντας την εμφάνιση, και επηρεάζοντας άλλες πτυχές της λειτουργίας ενός αντικειμένου στο παιχνίδι. Με την τοποθέτηση Components σε ένα αντικείμενο, μπορεί κανείς να δώσει αμέσως καινούριες λειτουργικότητες στα αντικείμενα. Κοινά συστατικά της παραγωγής παιχνιδιού έρχονται ενσωματωμένα με τη Unity, όπως το συστατικό Rigidbody, φώτα, κάμερες, πομποί σωματιδίων και πολλά άλλα. Για να κατασκευάσουμε περαιτέρω διαδραστικά στοιχεία του παιχνιδιού, γράφουμε Scripts, τα οποία αντιμετωπίζονται επίσης ως συστατικά τα οποία επεκτείνουν ή τροποποιούν την υπάρχουσα διαθέσιμη λειτουργικότητα της Unity.

3.5.5 Scripts

Μια γλώσσα σεναρίων (scripting language, script language) είναι μια γλώσσα προγραμματισμού, που επιτρέπει τον έλεγχο μιας ή περισσότερων εφαρμογών. Το Scripting αποτελεί βασικό συστατικό σε όλα τα παιχνίδια. Ακόμα και το πιο απλό παιχνίδι θα χρειαστεί σενάρια για να ανταποκριθεί στην είσοδο του χρήστη και να μεριμνήσει για τα γεγονότα του gameplay, ώστε να πραγματοποιηθούν όταν πρέπει. Από εκεί και πέρα, τα σενάρια μπορούν να χρησιμοποιηθούν για να δημιουργήσουν οπτικά εφέ, να ελέγχουν τη φυσική συμπεριφορά των αντικειμένων ή ακόμα και να υλοποιήσουν ένα προσαρμοσμένο σύστημα τεχνητής νοημοσύνης AI για τους χαρακτήρες του παιχνιδιού.

Στη Unity, τα Scripts είναι συστατικά τα οποία επεκτείνουν ή τροποποιούν τις υπάρχουσες διαθέσιμες λειτουργικότητες της. Δημιουργούνται και τοποθετούνται σαν υπομονάδες στα αντικείμενα (GameObjects) που επιθυμούμε προσδίδοντας τους λειτουργικότητα, δίχως να υπάρχει κάποιος περιορισμός στο πόσα Scripts μπορούν να χρησιμοποιηθούν. Επίσης, προσφέρει έναν πλήρη συντάκτη κώδικα scripting, το MonoDevelop, παρέχοντας τη δυνατότητα στους χρήστες να προγραμματίσουν είτε σε JavaScript είτε σε C# είτε σε Boo. Στα πλαίσια της παρούσας πτυχιακής εργασίας επιλέχθηκε η C# και η JavaScript για ορισμένα αρχεία Scripts.

Όταν θέλει κανείς να παρακολουθήσει τη λογική του παιχνιδιού και των αλληλεπιδράσεων, τα κινούμενα σχέδια, τις θέσεις της κάμερας, κλπ, υπάρχουν διάφοροι τρόποι που μπορεί να γίνει. Το κοινό μοντέλο είναι να εκτελέσουμε τις περισσότερες εργασίες στο εσωτερικό της συνάρτησης Update, ωστόσο υπάρχουν και άλλες λειτουργίες που μπορούμε να χρησιμοποιήσουμε.

- **FixedUpdate:** Η FixedUpdate συνήθως καλείται πιο συχνά από ότι η Update. Μπορεί να κληθεί πολλές φορές σε ένα καρέ, εάν ο ρυθμός καρέ (frame rate) είναι χαμηλός και μπορεί να μην κληθεί καθόλου, αν ο ρυθμός καρέ είναι υψηλός. Όλοι οι υπολογισμοί της φυσικής και οι ενημερώσεις συμβαίνουν αμέσως μετά τη FixedUpdate. Όταν πρέπει να γίνουν υπολογισμοί για την κίνηση μέσα στην FixedUpdate, δεν χρειάζεται να πολλαπλασιαστούν οι τιμές με την Time.deltaTime. Αυτό συμβαίνει επειδή η FixedUpdate καλείται ανεξάρτητα από τον ρυθμό καρέ.
- **Update:** Η Update καλείται μία φορά ανά καρέ. Είναι η κύρια συνάρτηση για τις ενημερώσεις σε κάθε καρέ.
- **LateUpdate:** Η LateUpdate καλείται μια φορά σε κάθε καρέ, αφότου η συνάρτηση Update έχει τελειώσει. Όλοι οι υπολογισμοί που εκτελούνται στην Update, θα πρέπει να έχουν ολοκληρωθεί όταν αρχίζει η LateUpdate.

Κατά τη σύνταξη του κώδικα μπορούν να υλοποιηθούν διάφορες κλάσεις οι οποίες προσδίδουν λειτουργικότητα. Η MonoBehaviour είναι η βασική κλάση από την οποία κάθε σενάριο προέρχε-

ται. Έτσι, λοιπόν, αν δημιουργηθεί κάποια κλάση η οποία θα καλεί ή θα υλοποιεί συναρτήσεις της Unity, η κλάση αυτή θα πρέπει να οριστεί ως υποκλάση της Monobehavior. Τις πιο πολλές φορές όμως, οι συναρτήσεις που παρέχει η Unity τόσο σε επίπεδο λειτουργιών όσο και σε επίπεδο γραφικών δεν επαρκούν για τη λειτουργικότητά των παιχνιδιών, με αποτέλεσμα ο σχεδιαστής να πρέπει να υλοποιήσει τις δικές του.

Για τις ανάγκες, λοιπόν της πτυχιακής αυτής εργασίας υλοποιήθηκε πληθώρα scripts και συναρτήσεων για τη διαχείριση του παίχτη (Physics Model), τη σχεδίαση της γραφικής διεπαφής χρήστη, το χειρισμό των ήχων, της φυσικής, της τεχνητής νοημοσύνης, καθώς και για την ενσωμάτωση του Oculus Rift στο παιχνίδι, ώστε να λειτουργεί το Head-Tracking.

3.5.6 Prefabs

Τα Prefabs αποτελούν προκατασκευασμένες και αποθηκευμένες εκδόσεις ενός αντικειμένου που μπορούν να επαναχρησιμοποιηθούν σε διάφορα μέρη του προγράμματός μας. Με τη χρήση των Prefabs, πολύπλοκα αντικείμενα με διάφορα στοιχεία και ρυθμίσεις μπορούν να χρησιμοποιηθούν ανά πάσα στιγμή, επιτρέποντας σε κάθε ένα ξεχωριστά να τροποποιηθεί όποτε και αν το επιθυμούμε.

3.6 Oculus Rift DK2

Μετά την υλοποίηση του βιντεοπαιχνιδιού στη Unity 3D, σειρά είχε η ενσωμάτωσή ενός HMD για τη δημιουργία ενός εικονικού περιβάλλοντος. Έτσι λοιπόν, χρησιμοποιήθηκε το Oculus Rift DK2 το οποίο υπάρχει διαθέσιμο στο εργαστήριο και δίνει τη δυνατότητα στο χρήστη, της εμβύθισης (Immersion) μέσα σε εικονικά περιβάλλοντα. Δημιουργεί την αίσθηση ότι ο χρήστης μεταφέρεται εξολοκλήρου σε ένα εικονικό (ή πραγματικά, αλλά ψηφιακά αναπαραγμένο) τρισδιάστατο κόσμο και μπορεί να παρέχει μια πολύ πιο έντονη εμπειρία από αυτή που προσφέρει μια οθόνη. Μπορεί να συγκριθεί με τη διαφορά μεταξύ του να κοιτάς ένα δωμάτιο μέσα από ένα παράθυρο, σε σχέση με το να περπατάς μέσα στο δωμάτιο και να μετακινείσαι ελεύθερα. Ο λόγος που επιλέχτηκε το Oculus Rift DK2 είναι ότι παρέχει απλή και γρήγορη ενσωμάτωση (integration) στη Unity 3D, προσφέροντας παράλληλα εντυπωσιακά αποτελέσματα.

Αν όμως οι εμπειρίες εικονικής πραγματικότητας παραβλέπουν ορισμένες βασικές πρακτικές, μπορούν να οδηγήσουν σε περίεργα συμπτώματα όπως καταπόνηση των ματιών, αποπροσανατολισμό και ναυτία. Ιστορικά, πολλά από αυτά τα προβλήματα έχουν αποδοθεί σε μη βέλτιστες μεταβλητές του υλικού, όπως για παράδειγμα στη λανθάνουσα κατάσταση του συστήματος (latency). Το Oculus Rift αντιπροσωπεύει μια νέα γενιά συσκευών εικονικής πραγματικότη-

τας, καθώς επιλύει πολλά ζητήματα των προηγούμενων συστημάτων. Αλλά ακόμη και με μια βέλτιστη υλοποίηση του υλικού, κακώς σχεδιασμένο περιεχόμενο μπορεί ακόμα να οδηγήσει σε μια δυσάρεστη εμπειρία.

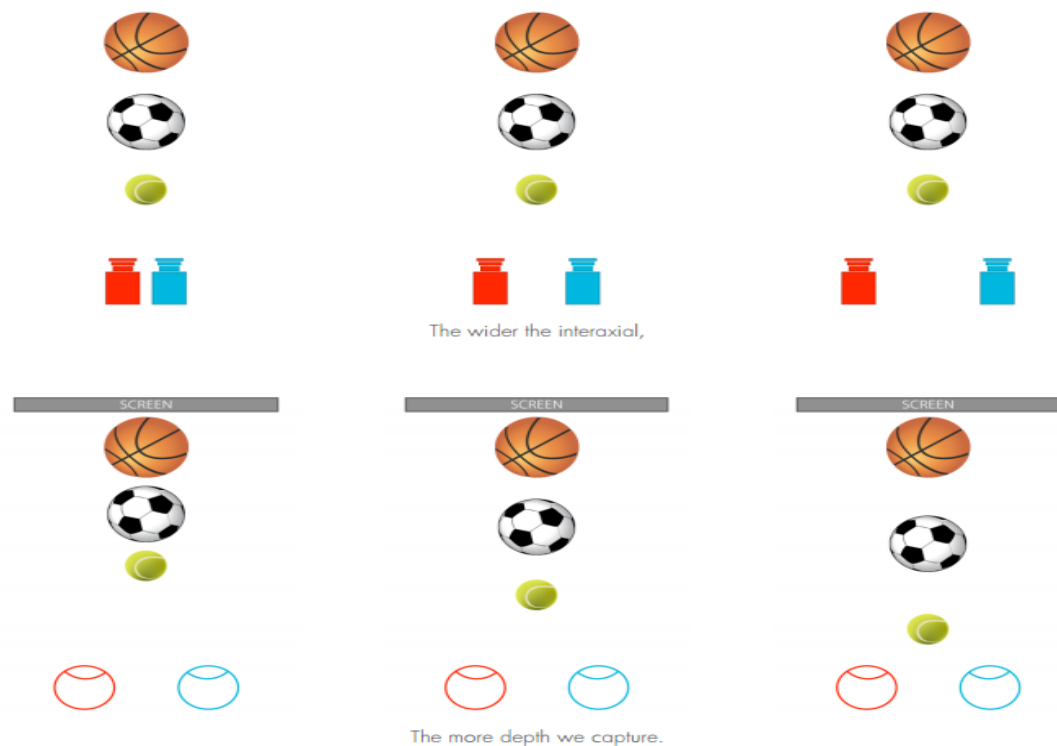
Επειδή η εικονική πραγματικότητα αποτελεί ένα πολύ εξειδικευμένο αντικείμενο, εξακολουθούν να υπάρχουν πτυχές που δεν έχουν μελετηθεί αρκετά. Οι δοκιμές από τους χρήστες είναι απολύτως ζωτικής σημασίας για το σχεδιασμό ευχάριστων και άνετων εμπειριών καθώς ως ένα δημοφιλές μέσο είναι ακόμα πολύ λιγότερο καιρό στο προσκήνιο για να έχουν καθιερώσει συμβάσεις στις οποίες μπορούμε να βασιστούμε. Στη συνέχεια περιγράφονται ορισμένα τεχνικά στοιχεία και χαρακτηριστικά του Oculus Rift DK2 [32].

3.6.1 Binocular Vision

Το Binocular Vision περιγράφει τον τρόπο με τον οποίο βλέπουμε δύο όψεις του κόσμου ταυτόχρονα. Η θέα από κάθε μάτι είναι ελαφρώς διαφορετική και το μυαλό μας τις συνδυάζει σε μια ενιαία τρισδιάστατη στερεοσκοπική εικόνα, μια εμπειρία γνωστή ως στερέοψη (stereopsis). Η διαφορά ανάμεσα σε αυτό που βλέπουμε από το αριστερό μάτι μας και αυτό που βλέπουμε από το δεξί παράγει τη διόφθαλμη διασπορά (binocular disparity).

Το Oculus Rift παρουσιάζει δύο εικόνες, μία για κάθε μάτι, οι οποίες παράγονται από δύο εικονικές κάμερες και οι οποίες χωρίζονται από μια μικρή απόσταση. Η απόσταση μεταξύ των δύο ματιών μας καλείται inter-pupilarity distance (IPD), και η απόσταση μεταξύ των δύο καμερών που συλλαμβάνουν το εικονικό περιβάλλον inter-camera distance (ICD) (σχήμα 3.9). Παρά το γεγονός ότι η IPD μπορεί να κυμαίνεται από περίπου 52 χιλιοστά έως 78 χιλιοστά, κατά μέσο όρο είναι περίπου 63,5, το ίδιο όπως η απόσταση μεταξύ των κέντρων των φακών του Rift.

Αλλαγή στην απόσταση μεταξύ των καμερών (ICD), μπορεί να επηρεάσει τους χρήστες σημαντικά. Αν η απόσταση μεταξύ των καμερών είναι αυξημένη, δημιουργεί μια εμπειρία γνωστή ως hyperstereo στην οποία το βάθος μεγαλώνει. Αν μειωθεί, το βάθος θα μικρύνει, μια κατάσταση γνωστή ως hypostereo. Η αλλαγή στην απόσταση μεταξύ των καμερών έχει δύο επιπλέον συνέπειες για το χρήστη: Πρώτον, αλλάζει ο βαθμός στον οποίο τα μάτια πρέπει να συγκλίνουν για να δουν ένα συγκεκριμένο αντικείμενο καθώς όσο αυξάνεται η απόσταση μεταξύ των καμερών, οι χρήστες πρέπει να συγκλίνουν περισσότερο τα μάτια τους για να δουν το ίδιο αντικείμενο και αυτό μπορεί να οδηγήσει σε καταπόνηση των ματιών. Δεύτερον, μπορεί να αλλάξει την αίσθηση του μεγέθους του χρήστη μέσα στο εικονικό περιβάλλον [33, 34].



Σχήμα 3.9: Απόσταση μεταξύ των καμερών (ICD)

3.6.2 Field of View & Scale

Το οπτικό πεδίο Field of View μπορεί να αναφέρεται σε διαφορετικά πράγματα που θα πρέπει πρώτα να αποσαφηνιστούν. Υπάρχουν δύο ειδών οπτικά πεδία, το display Field of View (dFOV) και το camera Field of View (cFOV). Το dFOV αναφέρεται στο φυσικό οπτικό πεδίο του χρήστη το οποίο καταλαμβάνεται από περιεχόμενο εικονικής πραγματικότητας. Είναι ένα φυσικό χαρακτηριστικό του υλικού και των οπτικών. Το cFOV αναφέρεται στο εύρος του εικονικού κόσμου, που φαίνεται από την απόδοση των καμερών σε οποιαδήποτε δεδομένη στιγμή. Όλα τα οπτικά πεδία FOVs ορίζονται από μια γωνιακή μέτρηση των κάθετων, οριζόντιων και / ή διαγώνιων διαστάσεων.

Στην εικονική πραγματικότητα, δεν υπάρχει επαφή του εξωτερικού περιβάλλοντος και ο εικονικός κόσμος γεμίζει ένα μεγάλο μέρος της περιφερειακής όρασης. Συνεπώς, είναι πολύ σημαντικό το cFOV και το dFOV να ταιριάζουν ακριβώς. Η αναλογία μεταξύ αυτών των δύο τιμών αναφέρεται ως κλίμακα, και στην εικονική πραγματικότητα η κλίμακα αυτή θα πρέπει πάντα να είναι ίση με ένα.

Στο Oculus Rift DK2, το μέγιστο dFOV καθορίζεται από την οθόνη, τους φακούς, και το πόσο κοντά ο χρήστης τοποθετεί τους φακούς στα μάτια του (σε γενικές γραμμές, όσο πιο

κοντά είναι τα μάτια στους φακούς, τόσο ευρύτερο είναι το dFOV. Το βοηθητικό πρόγραμμα διαμόρφωσης μετρά το μέγιστο dFOV που οι χρήστες μπορούν να δουν, αποθηκεύει αυτές τις πληροφορίες και το SDK θα συστήσει το κατάλληλο cFOV που ταιριάζει. Έτσι το SDK επιτρέπει τον χειρισμό του cFOV και του dFOV χωρίς αλλαγή της κλίμακας και το κάνει αυτό με την προσθήκη μαύρων περιθωρίων γύρω από την ορατή εικόνα [35].

3.6.3 Rendering Techniques

Παρακάτω αναλύονται βασικές τεχνικές απόδοσης (Rendering Techniques) καθώς και τα χαρακτηριστικά τους, τις οποίες χρησιμοποιεί το Oculus Rift για να αποδώσει με όσο το δυνατόν καλύτερο και ρεαλιστικότερο τρόπο το εικονικό περιβάλλον στο χρήστη. Αξίζει να σημειωθεί ότι στο Oculus Rift DK2 έχουν σημειωθεί σημαντικές βελτιώσεις σε σχέση με το Oculus Rift DK1 όπως μεγαλύτερη ανάλυση (από 1280x720 σε 1920x1080) καθώς και μεγαλύτερο ρυθμό ανανέωσης (από 60hz σε 75hz) ο οποίος έχει σαν αποτέλεσμα τη μείωση του θαμπώματος (motion blur) και του flickering καθώς μετακινούμε το κεφάλι μας [36].

- **Rendering resolution:** Όπως αναφέρθηκε και παραπάνω το Oculus Rift DK2 έχει ανάλυση οθόνης 1920x1080, αλλά η παραμόρφωση των φακών σημαίνει ότι η εικόνα στην οθόνη πρέπει να μετασχηματιστεί ώστε να εμφανιστεί κανονική στο θεατή. Προκειμένου να παρασχεθεί επαρκής πυκνότητα εικονοστοιχείων για τον μετασχηματισμό, κάθε μάτι απαιτεί εικόνα η οποία να είναι μεγαλύτερη από την ανάλυση του μισού της οθόνης.

Αυτό μπορεί να αποτελέσει πρόβλημα απόδοσης για ορισμένες κάρτες γραφικών και η μείωση του ρυθμού των καρτέ παραγάγει μια κακή εμπειρία εικονικής πραγματικότητας. Ρίχνοντας την ανάλυση της οθόνης, μπορεί να έχει επίδραση και να δημιουργήσει μορφώματα. Ρίχνοντας όμως την ανάλυση των ρυθμιστών στο μάτι, μπορεί να βελτιώσει τις επιδόσεις, διατηρώντας την οπτική ποιότητα.

- **Dynamically-rendered impostors/billboards:** Η αντίληψη του βάθους γίνεται λιγότερο ευαίσθητη σε μεγαλύτερες αποστάσεις από τα μάτια. Από κοντά, η στερέωση τις περισσότερες φορές μας επιτρέπει να αντιληφθούμε ποια από τα δύο αντικείμενα στο γραφείο μας είναι πιο κοντά σε εμάς. Αυτό γίνεται όμως δυσκολότερο για αντικείμενα που βρίσκονται μακριά. Αν για παράδειγμα κοιτάζουμε δύο δέντρα στο απέναντι πάρκο, μπορεί να χρειαστεί να πλησιάσουμε κοντά για να μπορέσουμε με σιγουριά να πούμε ποιο είναι πιο κοντά ή πιο μακριά. Σε ακόμα μεγαλύτερες κλίμακες, το πρόβλημα αυτό γίνεται εντονότερο. Έτσι, μπορούμε να εκμεταλευτούμε αυτή την σχετική έλλειψη ευαισθησίας για την αντίληψη του βάθους στην απόσταση, απελευθερώνοντας υπολογιστική ισχύ με τη χρήση των λεγόμενων imposter ή billboard υφών, αντικαθιστώντας το πλήρως τρισ-

διάστατο τοπίο. Με αυτόν τον τρόπο είναι δυνατόν να ξεγελαστούν τα μάτια σε εικονικά περιβάλλοντα.

- **Normal mapping vs Parallax Mapping:** Η τεχνική που είναι γνωστή ως 'κανονική χαρτογράφηση' (Normal mapping) είναι μια τεχνική που χρησιμοποιείται για να μεταφέρει το βάθος και την υφή χωρίς την προσθήκη περισσότερων πολυγόνων. Χρησιμοποιείται για να ενισχυθεί η εμφάνιση και οι λεπτομέρειες ενός μοντέλου λίγων πολυγόνων δημιουργώντας Normal map από ένα μοντέλο πολλών πολυγόνων. Αν και χρησιμοποιείται ευρέως στα σύγχρονα παιχνίδια, είναι λιγότερο συναρπαστικό όταν προβάλλεται σε στερεοσκοπικό τρισδιάστατο περιβάλλον. Επειδή το Normal mapping δεν αντιπροσωπεύει τη διόφθαλμη διασπορά (Binocular disparity), παράγει μια εικόνα που μοιάζει με μια επίπεδη υφή ζωγραφισμένη πάνω στο μοντέλο αντικειμένου.

Το Parallax mapping βασίζεται στην ιδέα του Normal mapping, αλλά αντιπροσωπεύει ενδείξεις βάθους που το Normal mapping δεν το κάνει. Έτσι, χρησιμοποιείται καλύτερα σε επιφάνειες με λεπτομέρειες που δεν επηρεάζουν την επιφάνεια σύγκρουσης, όπως οι τοίχοι από τούβλα ή τα λιθόστρωτα μονοπάτια.

3.6.4 Κίνηση (Motion)

Η κίνηση αναφέρεται σε οποιαδήποτε κίνηση συμβαίνει μέσα σε ένα εικονικό περιβάλλον, που δεν είναι το αποτέλεσμα κινήσεων σε πραγματικό περιβάλλον που έχουν αντιστοιχηθεί στο εικονικό. Η κίνηση και η επιτάχυνση πιο συχνά προέρχονται από τον εικονικό χαρακτήρα (avatar) του χρήστη που κινείται μέσα στο εικονικό περιβάλλον, ενώ παράλληλα στον πραγματικό κόσμο το σώμα του χρήστη παραμένει σταματημένο. Αυτές οι καταστάσεις μπορούν να προκαλέσουν δυσφορία διότι η οπτική του χρήστη του λέει ότι κινείται μέσα στο χώρο, αλλά οι σωματικές αισθήσεις του λένε το αντίθετο. Αυτή η ψευδαίσθηση της κίνησης, αποτελεί μια σημαντική αιτία δυσφορίας κατά τη διάρκεια της εικονικής εμπειρίας [37].

3.6.5 Παρακολούθηση (Tracking)

Το Oculus Rift περιέχει ένα γυροσκόπιο, ένα επιταχυνσιόμετρο και ένα μαγνητόμετρο και συνδυάζει αυτές τις πληροφορίες από τους αισθητήρες μέσω μιας διαδικασίας για να καθορίσει τον προσανατολισμό του κεφαλιού του χρήστη στον πραγματικό κόσμο και να συγχρονίσει την εικονική πλευρά του χρήστη σε πραγματικό χρόνο. Αυτοί οι αισθητήρες παρέχουν δεδομένα για να παρακολουθείτε με ακρίβεια η γωνιακή παρέκλιση του κεφαλιού (pitch, yaw, roll). Έτσι, λοιπόν, με τη χρήση ενός μοντέλου που αντανακλά το γεγονός ότι η κίνηση της κεφαλής σε οποιαδήποτε από τις τρεις κατευθύνσεις στρέφεται στην πραγματικότητα γύρω από ένα σημείο

περίπου στη βάση του λαιμού, είναι εφικτό με ακρίβεια να μετατραπούν οι πληροφορίες των αισθητήρων από τις κινήσεις του κεφαλιού σε κινήσεις της κάμερας.

Επιπλέον το Development Kit 2 εισάγει έξι βαθμούς ελευθερίας εντοπισμού θέσης. Κάτω από το ημιδιαφανές εξωτερικό περίβλημα του DK2 είναι μια συστοιχία υπέρυθρων micro-LEDs, που παρακολουθούνται σε πραγματικό χώρο από κάμερα υπέρυθρων. Η παρακολούθηση θέσης (Positional tracking) θα πρέπει να αντιστοιχεί πάντα ένα προς ένα με τις κινήσεις του χρήστη, ειδάλλως μπορεί να προκαλέσει δυσφορία.

Ένας βασικός παράγοντας του tracking που θα πρέπει να λαμβάνεται σοβαρά υπόψιν είναι η λανθάνουσα κατάσταση (latency), η οποία ορίζεται ως ο συνολικός χρόνος μεταξύ της κίνησης του κεφαλιού του χρήστη και της ανανεωμένης εικόνας που εμφανίζεται στην οθόνη και περιλαμβάνει τους χρόνους για την απόκριση του αισθητήρα, τη σύντηξη, την απόδοση, τη μετάδοση της εικόνας και την απόκριση προβολής. Η ελαχιστοποίηση λοιπόν, της λανθάνουσας κατάστασης είναι ζωτικής σημασίας για τον χρήστη, για μια καλή εμπειρία εμπύθισης σε εικονικό περιβάλλον [38].

3.6.6 Δυσφορία Προσομοίωσης (Simulator Sickness)

Η ζάλη ή αλλιώς ναυτία της προσομοίωσης (Simulator Sickness) συμβαίνει όταν η οπτική πληροφορία από ένα περιβάλλον προσομοίωσης δημιουργεί σύγχυση, μπερδεύοντας τον χρήστη ως προς τη θέση του, την ισορροπία του και τον τρόπο κίνησής του. Επίσης περιλαμβάνει συμπτώματα όπως καταπόνηση των ματιών και κόπωση. Ορισμένοι χρήστες είναι πιθανόν να βιώσουν κάποια συμπτώματα μετά από ένα σύντομο χρονικό διάστημα, ενώ άλλοι μπορεί να μην βιώσουν ποτέ. Οι λόγοι που προκαλούν αυτή τη δυσφορία περιγράφονται συνοπτικά παρακάτω [39].

- **Ταχύτητα Κίνησης & Επιτάχυνση (Speed of Movement & Acceleration):** Αποτελεί βασική αιτία της ζάλης και της δυσφορίας, καθώς ενώ μεταφέρεται οπτικά, δεν μεταφέρεται στο αιθουσαίο σύστημα (στο σύστημα δηλαδή που σε συνεργασία με το οπτικό σύστημα εξασφαλίζει την ισορροπία).
- **Βαθμός Ελέγχου (Degree of Control):** Προκαλείται αν αποσυνδεθούν οι κινήσεις του χρήστη από τις κινήσεις της κάμερας στο εικονικό περιβάλλον.
- **Διάρκεια (Duration):** Όσο περισσότερο παραμένουμε σε ένα περιβάλλον εικονικής πραγματικότητας, τόσο πιο πιθανό είναι να βιώσουμε ζάλη ή δυσφορία.
- **Ύψος (Altitude):** Όσο χαμηλότερη είναι η οπτική του χρήστη, τόσο πιο γρήγορα γίνονται οι αλλαγές στο επίπεδο του εδάφους προκαλώντας ζάλη και δυσφορία.

- **Διόφθαλμη Διασπορά Binocular Display:** Οι στερεοσκοπικές εικόνες μπορούν να ωθήσουν τα μάτια να συγκλίνουν σε ένα σημείο σε κάποιο βάθος, ενώ οι φακοί των ματιών να εστιάσουν σε άλλο σημείο.
- **Field of View:** Ένα μεγάλο dFOV είναι πιθανό να προκαλέσει δυσφορία για δύο λόγους που συνδέονται με την αντίληψη της κίνησης. Πρώτον, η αντίληψη της κίνησης είναι πιο ευαίσθητη στην περιφέρεια και δεύτερον, ένα μεγαλύτερο dFOV, όταν χρησιμοποιείται στο σύνολό του, παρέχει στο οπτικό σύστημα περισσότερα στοιχεία από ένα μικρότερο dFOV.
- **Λανθάνουσα Κατάσταση (Latency and Lag):** Πολλά παιχνίδια μπορεί να επιβραδυνθούν, ως αποτέλεσμα των πολλών ή πιο πολύπλοκων στοιχείων που πρέπει να κάνουν render στην οθόνη.
- **Διόρθωση Παραμόρφωσης (Distortion Correction):** Οι φακοί του Oculus Rift στρεβλώνουν την εικόνα που εμφανίζεται στην οθόνη, και αυτό διορθώνεται με την επεξεργασία που δίνεται στο SDK. Η προσοχή στις λεπτομέρειες είναι υψίστης σημασίας καθώς όλες οι τιμές διόρθωσης παραμόρφωσης πρέπει να ταιριάζουν με τα οπτικά του Rift.
- **Τρεμόσβημα (Flicker):** Μπορεί να γίνει εντονότερο από τα υψηλά επίπεδα φωτεινότητας, και γίνεται ακόμα πιο αντιληπτό στην περιφέρεια του οπτικού πεδίου. Παρά το γεγονός ότι το τρεμόπαιγμα μπορεί να γίνει λιγότερο αισθητά συνειδητό με την πάροδο του χρόνου, μπορεί ακόμα να οδηγήσει σε πονοκεφάλους και κούραση των ματιών.
- **Εμπειρία (Experience):** όσο πιο έμπειρος είναι ο χρήστης με τα εικονικά περιβάλλοντα, τόσο λιγότερες οι πιθανότητες να υποστεί συμπτώματα ζάλης.

Κεφάλαιο 4

Σχεδιασμός Παιχνιδιού & Γραφικής Διεπαφής Χρήστη

4.1 Εισαγωγή

Στο προηγούμενο κεφάλαιο έγινε εκτενής αναφορά στα λογισμικά που χρησιμοποιήθηκαν καθώς και στα τεχνικά τους χαρακτηριστικά. Στο κεφάλαιο αυτό, θα γίνει περιγραφή του τρόπου σχεδίασης του παιχνιδιού με βάση αυτά, του γραφικού περιβάλλοντος του παιχνιδιού, καθώς και της γραφικής διεπαφής του χρήστη.

4.2 Σενάριο

Το παιχνίδι που παρουσιάζεται στην παρούσα διπλωματική εργασία είναι ένα παιχνίδι προσομοίωσης πρώτου προσώπου. Σκοπός ήταν η εκμετάλλευση του Oculus Rift μέσω του Head Tracking για την κατεύθυνση της κίνησης του παίχτη. Ο παίχτης είναι ένας σκιέρ ο οποίος πρέπει να περάσει επιτυχώς τέσσερις δοκιμασίες ώστε να καταφέρει να τερματίσει το παιχνίδι. Οι τρεις πρώτες δοκιμασίες είναι ξεκλειδωτές και η τελευταία κλειδωμένη. Αφού τερματίσει τις τρεις πρώτες, ξεκλειδώνει και η τέταρτη. Αν επιτύχει και σε αυτήν, τότε έχει τερματίσει το παιχνίδι. Οι δοκιμασίες που πρέπει να επιτύχει ο παίχτης περιγράφονται συνοπτικά παρακάτω:

- **Slalom:** Στη δοκιμασία αυτή σημαντικό ρόλο παίζουν τα αντανακλαστικά του παίχτη και η ταχύτητα του, καθώς σκοπός είναι να τερματίσει την πίστα σε χρόνο λιγότερο από αυτόν των αντιπάλων του και να περάσει έναν συγκεκριμένο αριθμό από σημειάκια.
- **Junior Tour:** Στη δοκιμασία αυτή ανταγωνίζεται τρεις αντιπάλους και νικητής αναδει-

κνύεται αυτός που θα τερματίσει πρώτος. Στόχος του παίχτη είναι να μαζέψει διάφορα Speed Ups που θα τον βοηθήσουν να γίνει πιο αεροδυναμικός και να προσπεράσει τους αντιπάλους του. Οι αντίπαλοι, οι οποίοι έχουν τεχνητή νοημοσύνη, βρίσκουν το γρηγορότερο μονοπάτι στο βουνό για να τερματίσουν και αποφεύγουν τον σκιέρ κάνοντας προσπεράσεις.

- **Large Hill:** Στη δοκιμασία αυτή ο παίχτης παίρνει φόρα από μια μεγάλη κατηφόρα και μέσω Head Tracking προσπαθεί να κρατήσει ευθεία πορεία, να ελέγξει τον σκιέρ στον αέρα και να καταφέρει να τον προσγειώσει στο έδαφος χωρίς να ανατραπεί. Απαιτεί ακρίβεια και ταχύτητα από τον παίχτη ώστε να πηδήξει πιο μακριά από τους αντιπάλους. Το παιχνίδι αποτελείται από δυο γύρους, συνεπώς ο παίχτης πραγματοποιεί δυο άλματα και τα επιμέρους σκορ αθροίζονται ώστε να καθοριστεί η γενική κατάταξη.
- **Canadian Slope:** Στη δοκιμασία αυτή ο παίχτης πραγματοποιεί κατάβαση παίρνοντας από διάσπαρτα σημαία με τη διαφορά ότι τώρα αγριεμένοι λύκοι του έχουν στήσει καρτέρι. Σκοπός του παίχτη είναι να αποφύγει τους λύκους, οι οποίοι κινούνται έξυπνα προσομοιώνοντας αγέλες και να φτάσει ζωντανός στη βάση του βουνού.

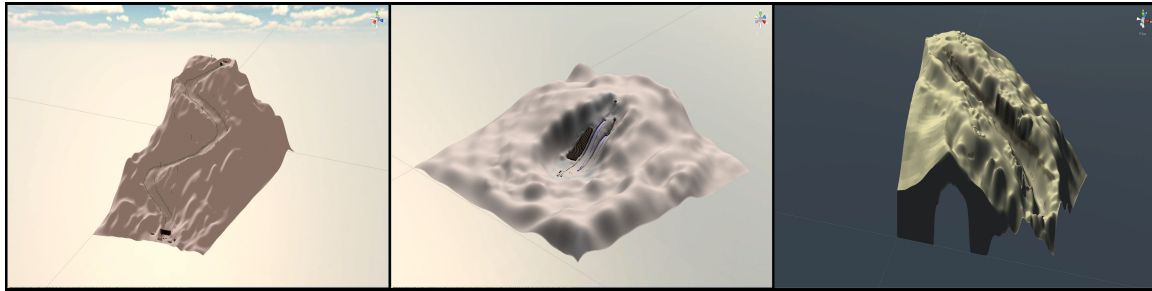
4.3 Σχεδιασμός Παιχνιδιού

Για τον σωστό σχεδιασμό του παιχνιδιού κρίθηκε αναγκαίο να μελετηθούν διάφορα άλλα παιχνίδια του ίδιου είδους, όπως για παράδειγμα το Vancouver 2010, το Shaun White Snowboarding κ.ά. Αφού βρέθηκαν κάποια κοινά χαρακτηριστικά αυτών των παιχνιδιών, αξιολογήθηκαν και μελετήθηκε αν θα ήταν ενδιαφέρον ή όχι να υλοποιηθούν σε παιχνίδι εικονικής πραγματικότητας. Στην παρούσα, λοιπόν, ενότητα γίνεται μια αναλυτική περιγραφή του τρόπου σχεδιασμού των γραφικών σε όλα τα επίπεδα.

4.3.1 Σχεδιασμός Πίστας

Οι πίστες που χρειάστηκε να σχεδιαστούν ήταν συνολικά τρεις. Κατασκευάστηκαν, λοιπόν, τρία μοντέλα βουνών τα οποία έχουν αρχική και τελική θέση, για την έναρξη και τον τερματισμό αντίστοιχα. Η πλοήγηση του παίχτη θα πρέπει να είναι ξεκάθαρη και άνετη, με φανερές κατηφόρες καθώς και σημεία τα οποία δεν μπορεί να προσπελάσει (σχήμα 4.1). Έτσι λοιπόν, ένα μοντέλο βουνού χρησιμοποιήθηκε για τις δοκιμασίες Slalom, καθώς και για την πρώτη πίστα του training. Ένα δεύτερο μοντέλο βουνού χρησιμοποιήθηκε για τις δοκιμασίες Junior Tour και Canadian Slope, διαφοροποιώντας τις ελαφρώς, αλλάζοντας ορισμένα τρισδιάστατα αντικείμενα και φωτισμό. Τέλος, ένα τρίτο μοντέλο βουνού σχεδιάστηκε για τη δοκιμασία του

Large Hill σε συνδυασμό με ένα τρισδιάστατο μοντέλο για την κατηφόρα του άλματος, το οποίο και ανακτήθηκε από το διαδίκτυο. Για την κατασκευή όλων των παραπάνω, χρησιμοποιήθηκε ο συντάκτης εδάφους (terrain editor) που παρέχει η Unity 3D.



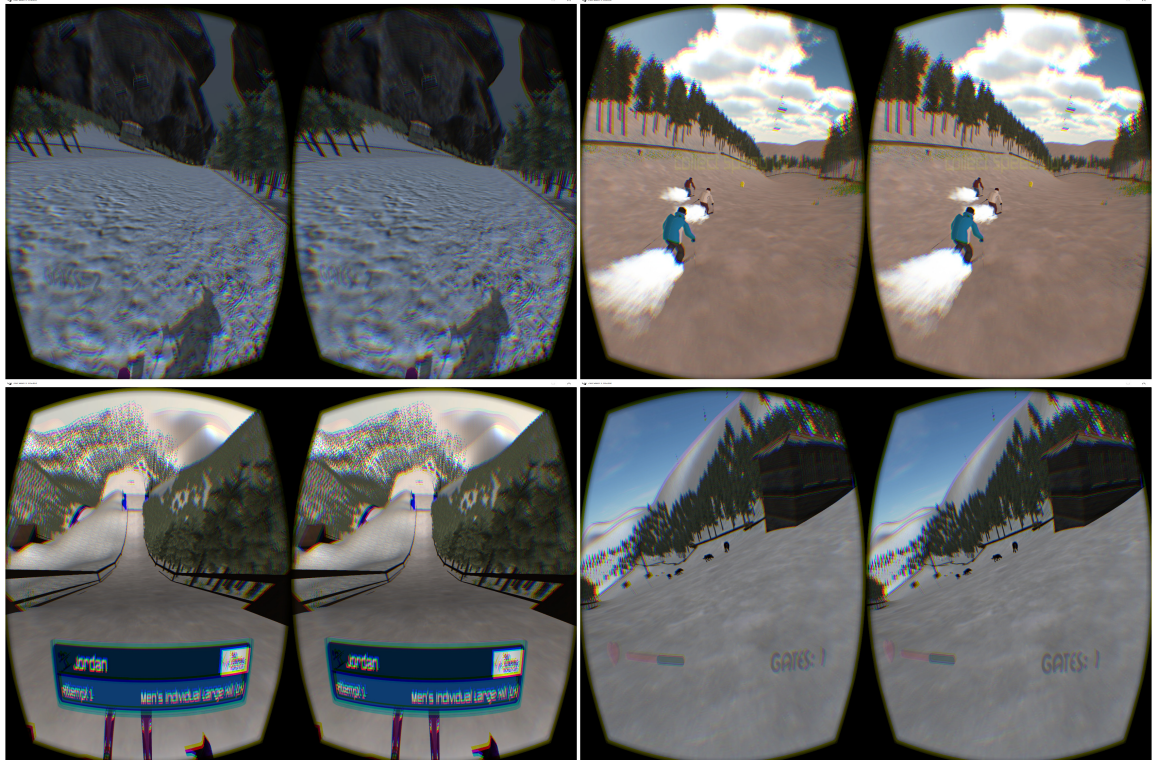
Σχήμα 4.1: *Terrains*
Αριστερά: *Junior Tour Terrain*, **Κέντρο:** *Slalom Terrain*, **Δεξιά:** *Large Hill Terrain*

Αφού ολοκληρώθηκε η κατασκευή των terrains, το επόμενο βήμα ήταν η προσθήκη διάφορων στοιχείων ώστε να γίνει το περιβάλλον πιο ρεαλιστικό. Αρχικά, με τη βοήθεια του συντάκτη εδάφους προστέθηκε υφή χιονιού, χόρτα και δέντρα. Στη συνέχεια, προστέθηκαν διάφορα τρισδιάστατα μοντέλα, ώστε ο χρήστης να νιώσει εντονότερα το συναίσθημα του Presence μέσα στο εικονικό περιβάλλον. Για το λόγο αυτό διανθίστηκε με ξύλινα σπιτάκια, ski lifts, βράχους, προστατευτικές μπάρες, διάφορα οπτικά εφέ χιονιού (spray), προειδοποιητικές πινακίδες κ.ά, εφαρμόζοντάς τους υφές (μέσω των materials) που ανακτήθηκαν επίσης από το διαδίκτυο. Στη συνέχεια ακολουθούν μερικές εικόνες από διάφορες σκηνές του παιχνιδιού (σχήμα 4.2).

4.3.2 Σχεδιασμός Χαρακτήρων

Η σχεδίαση ενός χαρακτήρα αποτελεί σημαντικό στάδιο στον σχεδιασμό των γραφικών ενός παιχνιδιού. Για να μπορέσουν να αναπαρασταθούν με ακρίβεια το σώμα και τα χαρακτηριστικά του προσώπου ενός χαρακτήρα, χρειάζεται μεγάλη λεπτομέρεια και ακρίβεια. Έτσι, τον σχεδιασμό τους τον αναλαμβάνουν σχεδιαστές που έχουν μεγαλύτερη εμπειρία σε μοντελοποίηση χαρακτήρων. Όμως, λόγω προσωπικού ενδιαφέροντος για τη διαδικασία σχεδίασης ενός χαρακτήρα, έγινε μοντελοποίηση ενός αθλητή του snowboard στο 3D Studio Max. Αφού μοντελοποιήθηκε, του προστέθηκε σκελετός (rigging) και δημιουργήθηκε ένα animation άλματος. Ο χαρακτήρας αυτός εν τέλει δεν χρησιμοποιήθηκε στο παιχνίδι. Το μοντέλο του σκιέρ ανακτήθηκε από το διαδίκτυο (σχήμα 4.3) και έγινε εισαγωγή στο 3D Studio Max όπου και έγινε κατασκευή frame by frame των τριών παρακάτω animations:

- **Idle:** Στο animation αυτό ο παίχτης δεν κινείται και κάνει κάποιες στατικές κινήσεις.
- **Left Turn:** Το animation αυτό προσομοιώνει την αριστερή στροφή του παίχτη.



Σχήμα 4.2: Σκηνές παιχνιδιού
Πάνω Αριστερά: Slalom, Πάνω δεξιά: Junior Tour,
Κάτω αριστερά: Large Hill, Κάτω δεξιά: Wolves Slalom

- **Right Turn:** Το animation αυτό προσομοιώνει την δεξιά στροφή του παίχτη.

Αφού δημιουργήθηκαν και τα animations επόμενο βήμα ήταν η εισαγωγή τους στη Unity 3D. Τελικά, τα animations για αριστερή και δεξιά στροφή δεν χρησιμοποιήθηκαν διότι το παιχνίδι είναι First Person Controller και δεν είναι σχεδόν καθόλου ορατά, καθώς στην ουσία μετατοπίζουν το σώμα του σκιέρ στο πλάι, χαλώντας την οπτική που έχουμε από το Oculus Rift. Το animation ακινησίας χρησιμοποιήθηκε κανονικά, διότι, αν ο χρήστης κατεβάσει το κεφάλι του προς τα κάτω, φαίνονται οι κινήσεις των χεριών και των ποδιών, προσδίδοντας ρεαλιστικότητα. Επίσης, το ίδιο μοντέλο σκιέρ χρησιμοποιήθηκε στη δοκιμασία Junior Tour για να αναπαραστήσει τους αντιπάλους (σχήμα 4.4). Η μόνη διαφοροποίηση που υπάρχει ανάμεσά τους είναι στα ρούχα textures τους όπου με τη βοήθεια του Photoshop τροποποιήθηκαν.

Επιπλέον, για να γίνει το παιχνίδι και η συμπεριφορά του παίχτη πιο ρεαλιστική, χρησιμοποιήθηκε το σύστημα Ragdoll της Unity. Παρέχει έναν απλό οδηγό που επιτρέπει στο χρήστη να δημιουργήσει γρήγορα συγκρουστές, Rigidbodies(βαρύτητα) και αρθρώσεις, προσδίδοντας ρεαλιστική κίνηση του χαρακτήρα κατά την πτώση. Σε περίπτωση σύγκρουσης, οι αρθρώσεις του χαρακτήρα συμπεριφέρονται άκρως ρεαλιστικά και λυγίζουν ακριβώς όπως στην πραγματικότητα (σχήμα 4.5).



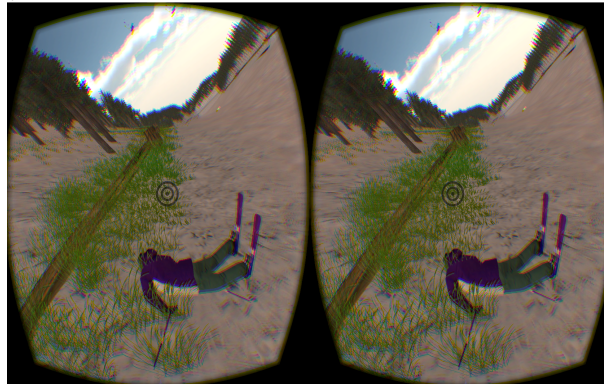
Σχήμα 4.3: Τρισδιάστατο μοντέλο σκιέρ



Σχήμα 4.4: Τρισδιάστατα μοντέλα αντίπαλων σκιέρ

Στην τελευταία δοκιμασία ο αντίπαλος πρέπει να ξεφύγει αρκετούς πεινασμένους λύκους. Τα τρισδιάστατα μοντέλα των λύκων, καθώς και τα animations τους, ανακτήθηκαν από το Asset Store της Unity 3d και ενσωματώθηκαν στο παιχνίδι (σχήμα 4.6). Τα animations που έχει ο κάθε λύκος είναι τα εξής παρακάτω:

- **Idle:** Στο animation αυτό ο λύκος δεν κινείται και κάνει κάποιες κινήσεις στατικές.
- **Run:** Το animation αυτό προσομοιώνει το τρέξιμο του λύκου και ενεργοποιείται όταν ο λύκος εντοπίσει (δει) τον σκιέρ ή κάποιος άλλος λύκος που βρίσκεται κοντά του τον ειδοποιήσει πως τον έχει δει εκείνος.
- **Attack:** Το animation αυτό προσομοιώνει την επίθεση προς τον σκιέρ και ενεργοποιείται όταν έρθει σε πολύ κοντινή απόσταση μαζί του.



Σχήμα 4.5: Πτώση σκιέρ (Ragdoll)



Σχήμα 4.6: Τρισδιάστατο μοντέλο λύκου

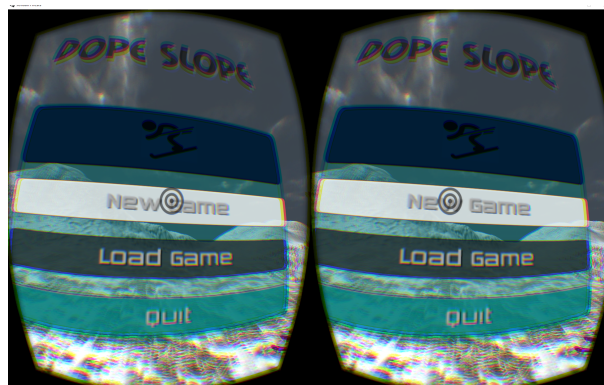
4.4 Γραφική Διεπαφή Χρήστη

Αρχικά, το παιχνίδι σχεδιάστηκε για να μπορεί ο χρήστης να το παίζει με το Oculus Rift και να περιηγείται στα μενού με το ποντίκι. Ο τρόπος αυτός απαιτούσε από τον χρήστη να βάζει και να βγάζει την κάσκα, τόσο κατά την έναρξη, όσο και κατά την λήξη ενός γύρου παιχνιδιού, μετά από παρότρυνση του συστήματος με προειδοποιητικά μηνύματα, γεγονός το οποίο μετέτρεπαι το User Experience σε μια ενοχλητική και κουραστική διαδικασία, καθώς απαιτούσε τη χρήση τριών διαφορετικών συσκευών εισόδου (headset, ποντίκι, πληκτρολόγιο). Έτσι λοιπόν, στην προσπάθεια επίλυσης του παραπάνω προβλήματος, αποφασίστηκε να τοποθετηθούν όλα τα μενού στον χώρο. Με τον τρόπο αυτό η χρήση του πληκτρολογίου και του ποντικιού δεν ήταν αναγκαία, καθώς πλέον έχοντας ο χρήστης στο κέντρο της οθόνης του πάντα έναν στόχο, μπορούσε να επιλέξει τις εκάστοτε επιλογές που του εμφανίζονταν στα μενού. Ο τρόπος επιλογής έγινε πολύ απλός και εύχρηστος, καθώς τοποθετώντας τον στόχο για λίγα δευτερόλεπτα (1-2 secs) πάνω στην επιλογή που θέλουμε γίνεται φόρτωση (loading). Το στυλ αυτό γραφικής διεπαφής διατηρήθηκε καθόλη τη διάρκεια του παιχνιδιού και για όλα τα μενού, με μοναδική εξαίρεση στο Large Hill όπου χρησιμοποιούνται δυο απλά buttons, τα Space και Ctrl

για Restart και Continue αντίστοιχα. Ο λόγος που χρησιμοποιήθηκαν αυτά τα δυο buttons γι' αυτήν αποκλειστικά τη δοκιμασία είναι ο απρόβλεπτος τρόπος που μπορεί να πέσει ο σκιέρ στο έδαφος. Τα μενού, σε περίπτωση που ο παίχτης χάσει, εμφανίζονται πάντα σε σχετική ως προς αυτόν θέση. Συνεπώς, θα έπρεπε να ελεγχθούν πολλές περιπτώσεις για τον τρόπο προσγείωσής του σκιέρ ώστε να ρυθμίζεται η σωστή θέση εμφάνισης του μενού λήξης. Στις υπόλοιπες δοκιμασίες τέτοιο πρόβλημα δεν υπάρχει, καθώς ο σκιέρ πραγματοποιεί απλή κατάβαση δίχως να χάνει την επαφή του με το έδαφος. Τέλος, χρησιμοποιήθηκε και το πλήκτρο Esc για παύση του παιχνιδιού και τα πλήκτρα Space και Ctrl και Esc για επανεκκίνηση, συνέχιση παιχνιδιού και έξοδο παιχνιδιού αντίστοιχα.

4.4.1 Αρχικό μενού

Το παιχνίδι ξεκινάει με το κυρίως μενού, το οποίο περιλαμβάνει ένα κουμπί έναρξης νέου παιχνιδιού (New Game), ένα κουμπί φόρτωσης παιχνιδιού (Load Game) το οποίο αρχικά είναι κλειδωμένο και ένα κουμπί εξόδου από το παιχνίδι (Quit). Όπως ειπώθηκε παραπάνω, τοποθετώντας τον στόχο για 120 καρέ, επιλέγεται αυτό που επιθυμεί ο χρήστης (σχήμα 4.7).



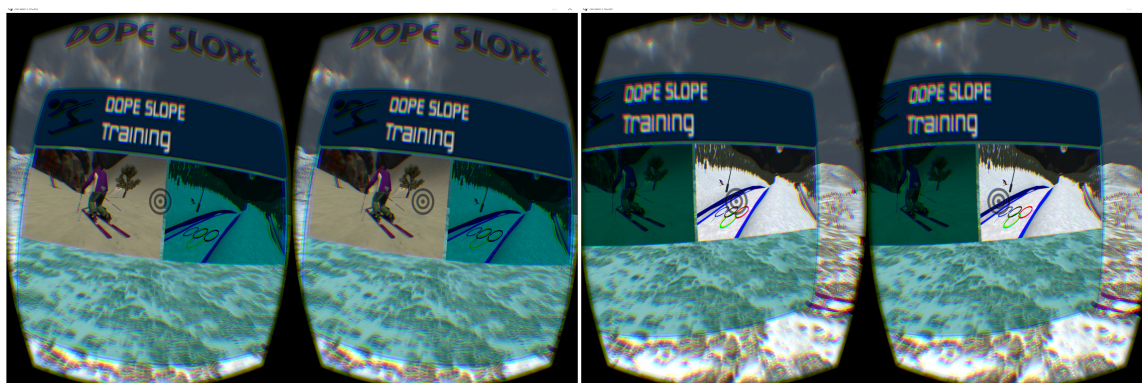
Σχήμα 4.7: Αρχικό μενού

Επιλέγοντας New Game γίνεται μεταφορά σε ένα άλλο μενού με τέσσερις διαφορετικές επιλογές (Training, Challenges, Instructions, Exit). Όπως φαίνεται και από τις εικόνες, κάνοντας hover σε μια επιλογή, αλλάζει το χρώμα και ακούγεται ένας χαρακτηριστικός ήχος (σχήμα 4.8).

- **Training:** Αν ο χρήστης επιλέξει Training, τότε εμφανίζεται μια νέα οθόνη μενού, η οποία του δίνει τη δυνατότητα να επιλέξει οποιαδήποτε από τις δυο επιλογές που φαίνονται. Στο Training ο χρήστης μπορεί να προπονηθεί σε δυο διαφορετικές πίστες κάνοντας είτε Slalom είτε άλματα. Με την επιλογή αυτή του δίνεται η δυνατότητα να προσαρμοστεί στο περιβάλλον εικονικής πραγματικότητας πριν ξεκινήσει τις αποστολές του, αποφεύγοντας με αυτό τον τρόπο να του δημιουργηθεί απότομα Simulator Sickness. Επιλέγοντας μια από τις δυο πίστες, ο χρήστης μεταβαίνει στην πίστα και το παιχνίδι ξεκινάει (σχήμα 4.9).

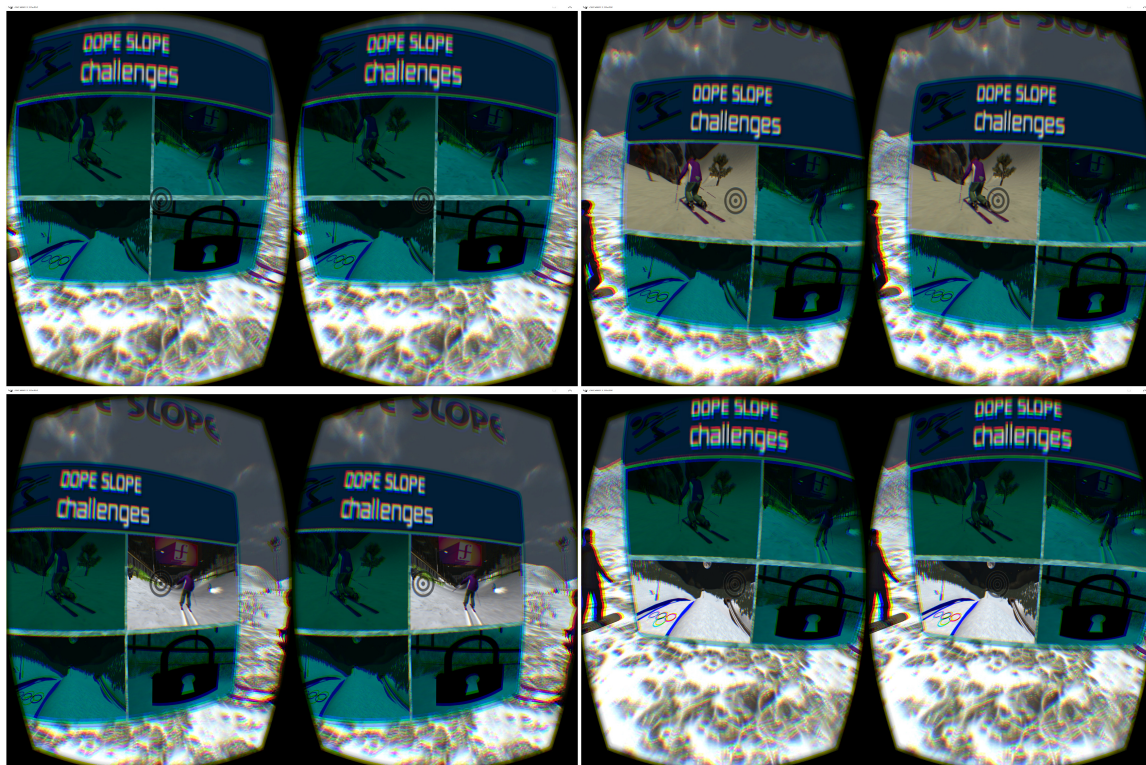


Σχήμα 4.8: Κυρίως μενού (Main Menu)



Σχήμα 4.9: Μενού προπόνησης (Training Menu)

- Challenges:** Αν ο χρήστης επιλέξει Challenges, τότε εμφανίζεται ένα παράθυρο-μενού που περιέχει όλες τις δοκιμασίες που πρέπει να ξεπεράσει, ώστε να τερματίσει το παιχνίδι. Αν ο χρήστης έχει επιλέξει New Game στο αρχικό μενού, τότε οι τρεις πρώτες δοκιμασίες είναι ξεκλειδωτές και ο χρήστης μπορεί να τις παίξει, ενώ η τελευταία είναι κλειδωμένη (σχήμα 4.10). Διαφορετικά, αν έχει επιλέξει Load Game στο αρχικό μενού, τότε εμφανίζεται μια εικόνα (πράσινο τικ) πάνω στην αντίστοιχη δοκιμασία ή δοκιμασίες που έχουν ολοκληρωθεί στο αμέσως προηγούμενο game session. Αν ο χρήστης ολοκληρώσει τις τρεις πρώτες δοκιμασίες, τότε οι επιλογές αυτές πρασινίζουν και η τελευταία πίστα ξεκλειδώνει και γίνεται διαθέσιμη προς επιλογή από το χρήστη(σχήμα 4.11). Μπορεί κανείς να έχει πρόσβαση σε οποιαδήποτε ξεκλειδωτή πίστα και να εισαχθεί σε αυτή ανεξαρτήτως σειράς μόνο μέσα από το μενού αυτό. Κάθε φορά που ο χρήστης ολοκληρώνει μια διαδικασία είτε επιτυχώς είτε ανεπιτυχώς επιστρέφει σε αυτό το μενού για να συνεχίσει το παιχνίδι. Η μόνη περίπτωση που δεν επιστρέφει, είναι όταν ολοκληρώσει όλες τις δοκιμασίες επιτυχώς, οπότε το σύστημα εμφανίζει ένα μήνυμα και ρωτάει τον χρήστη αν θέλει να ξεκινήσει όλες τις δοκιμασίες από την αρχή ή επιθυμεί να φύγει από το παιχνίδι.

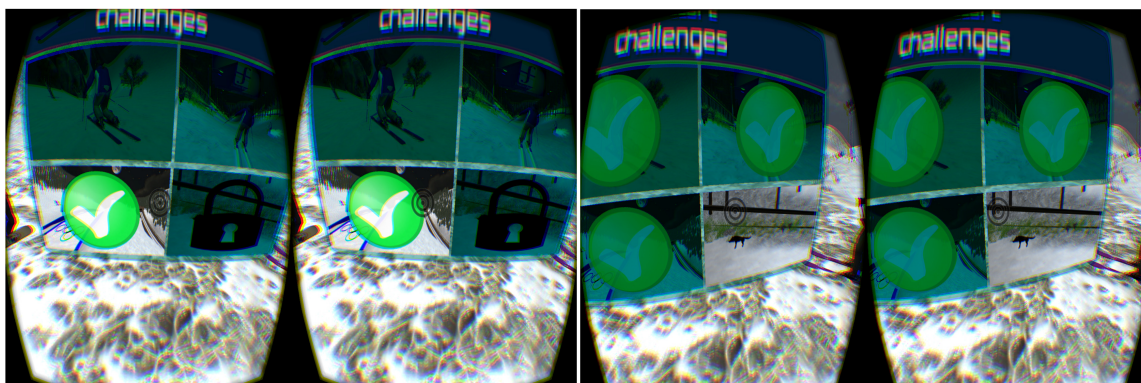


Σχήμα 4.10: Μενού Δοκιμασιών (Challenges Menu), Πάνω Αριστερά: Δεν έχουμε επιλέξει τίποτα, Πάνω Δεξιά: Slalom Challenge, Κάτω Αριστερά: Junior Tour Challenge, Κάτω Δεξιά: Large Hill Challenge

Τοποθετώντας το στόχο πάνω σε μια δοκιμασία και περιμένοντας για 1-2 δευτερόλεπτα, αλλάζει η υφή (texture) του εκάστοτε τετραγώνου (προστίθεται ένα transparent texture ενός πράσινου τικ), ακούγεται ένας χαρακτηριστικός ήχος επιλογής και εμφανίζεται μια μπάρα φόρτωσης, καθώς και γενικές πληροφορίες για την πίστα που ακολουθεί. Η μπάρα φόρτωσης διαρκεί περίπου δεκαπέντε δευτερόλεπτα, ώστε ο χρήστης να προλάβει να διαβάσει τις γενικές πληροφορίες και μόλις φτάσει στο 100%, φορτώνεται η αντίστοιχη σκηνή (σχήμα 4.12).

Μόλις τερματιστεί μια δοκιμασία, τότε η αντίστοιχη εικόνα στο μενού δοκιμασιών αλλάζει και εμφανίζεται μια ένδειξη ότι έχει ολοκληρωθεί. Στην περίπτωση αυτή δεν είναι εφικτό να την ξαναπαίξει ο χρήστης, δεδομένου ότι έχει ολοκληρωθεί επιτυχώς.

- **Instructions:** Αν ο χρήστης επιλέξει Instructions, τότε του εμφανίζεται μια οθόνη η οποία περιέχει διάφορες πληροφορίες για το παιχνίδι που πιθανόν να τον ενδιαφέρουν να διαβάσει πρώτου ξεκινήσει να παίζει. Οι οδηγίες που εμφανίζονται στην οθόνη συμπεριλαμβάνουν πληροφορίες όσον αφορά το στόχο του ήρωα, τους αντιπάλους, καθώς και τον τρόπο που χειρίζεται τον χαρακτήρα (σχήμα 4.13).
- **Exit:** Αν ο χρήστης επιλέξει Exit, τότε μεταβαίνει εκτός παιχνιδιού. Έχει υλοποιηθεί



Σχήμα 4.11: **Αριστερά:** Τερματισμός Large Hill Challenge, **Δεξιά:** Τερματισμός 3 πρώτων δοκιμασιών και ξεκλείδωμα τελευταίας (Canadian Slope Challenge)

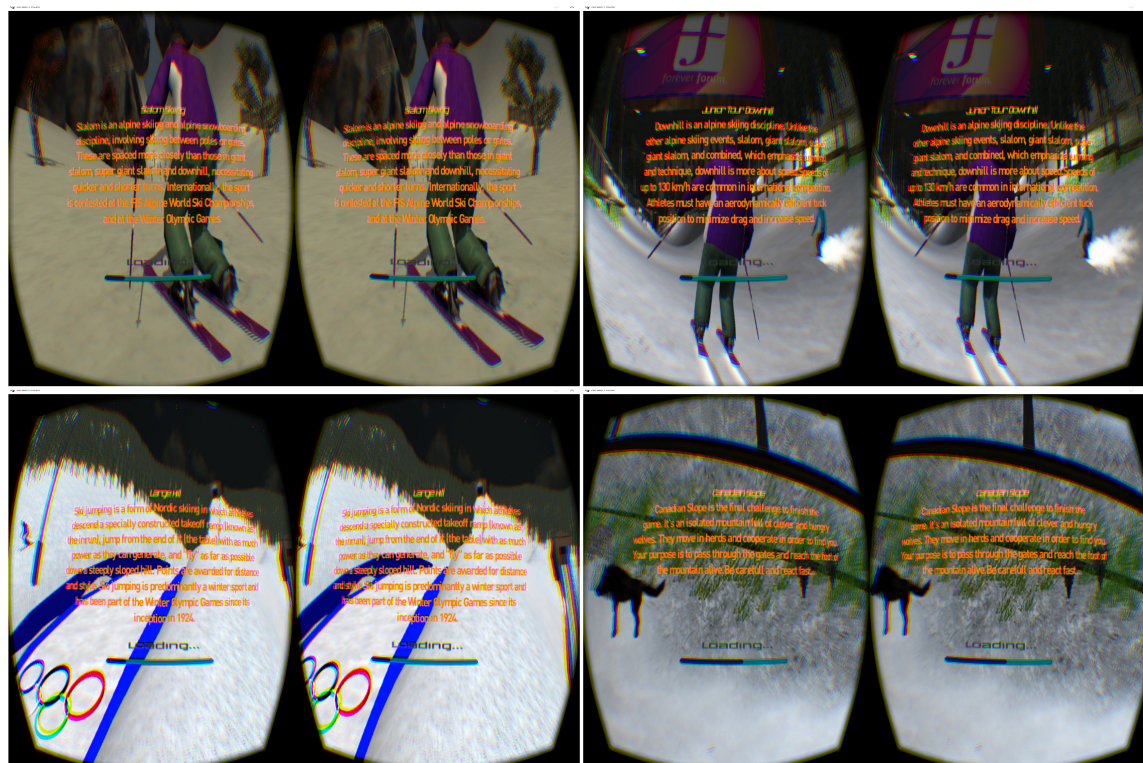
ένα πολύ απλό σύστημα αποθήκευσης-φόρτωσης, όπου ο χρήστης μπορεί να φορτώσει την κατάσταση του παιχνιδιού από το αμέσως προηγούμενο game session και να συνεχίσει. Δηλαδή, ο χρήστης μπορεί να συνεχίσει από εκεί που σταμάτησε την τελευταία φορά. Αν όμως ξεκινήσει καινούριο παιχνίδι (New Game) τα παλιά δεδομένα φόρτωσης χάνονται και αποθηκεύονται τα καινούρια.

Τέλος, όσο ο χρήστης βρίσκεται στα παράθυρα Training, Challenges και Instructions, γυρνώντας το κεφάλι του 90 μοίρες δεξιά μπορεί να δει ένα κουμπί Back, το οποίο βρίσκεται στο χώρο. Σηματοδοτώντάς το με τον στόχο και περιμένοντας λίγα δευτερόλεπτα, μεταβαίνει στο ακριβώς προηγούμενο παράθυρο που είναι το κεντρικό μενού (σχήμα 4.13).

4.4.2 Γραφικό περιβάλλον Παιχνιδιού

Πέρα από το γραφικό περιβάλλον έναρξης, τα περισσότερα παιχνίδια διαθέτουν μια γραφική διεπαφή για να δίνουν πληροφορίες στον χρήστη και να τον βοηθούν κατά τη διάρκεια του παιχνιδιού. Η γραφική αυτή διεπαφή ονομάζεται head-up display(HUD). Εκτός όμως από τα HUDs διαθέτουν επιπλέον διάφορες βοηθητικές οθόνες και μηνύματα τα οποία καθορίζουν ενέργειες του παιχνιδιού. Στην παρούσα πτυχιακή εργασία, στην προσπάθεια να υλοποιηθεί μια προσομοίωση του σκι με τη χρήση του Oculus Rift, επιλέχθηκε να μην υπερφορτωθεί η οθόνη με HUDs έτσι ώστε να μην αποσπάται η προσοχή του χρήστη και να επιτευχθεί καλύτερο user experience. Τα HUDs τα οποία χρησιμοποιήθηκαν, τοποθετήθηκαν στο κάτω μέρος της οθόνης και ήταν όλα ημιδιαφανή(transparent), έτσι ώστε να μην αποσπούν την προσοχή του χρήστη ούτε να του περιορίζουν την οπτική. Επίσης παρέμεναν στη θέση τους καθόλη τη διάρκεια της κάθε δοκιμασίας. Στη συνέχεια αναλύονται:

- **Μετρητής σημαιών(Gate Counter):** Βρίσκεται τοποθετημένος στο κάτω αριστερά μέρος της οθόνης και αποτελεί στην ουσία έναν μετρητή (σχήμα 4.14). Στις δοκιμασίες



Σχήμα 4.12: **Πάνω Αριστερά:**Slalom Loading Bar, **Πάνω Δεξιά:** Slalom Loading Bar, **Κάτω Αριστερά:**Large Hill Loading Bar, **Κάτω Δεξιά:** Wolves Slalom Loading Bar

Slalom και Wolves Slalom ο χρήστης έχει ως στόχο να περάσει έναν συγκεκριμένο αριθμό από σημαιάκια, οπότε είναι σημαντικό για αυτόν να γνωρίζει πόσα έχει περάσει την εκάστοτε χρονική στιγμή.

- **Χρονόμετρο(Timer):** Βρίσκεται τοποθετημένο στο κάτω δεξιά μέρος της οθόνης και μετράει πόσος χρόνος έχει περάσει από την εκκίνηση του σκιέρ (σχήμα 4.14). Στη δοκιμασία Slalom,ο παίχτης εκτός από το ότι πρέπει να περάσει έναν συγκεκριμένο αριθμό από σημαιάκια, πρέπει να φτάσει στον τερματισμό σε χρόνο μικρότερο από αυτόν που έφτασαν οι αντίπαλοί του.
- **Ενέργεια παίχτη(Energy):** Στην τελευταία δοκιμασία, ο παίχτης πρέπει εκτός των άλλων, να προσέχει μην του επιτεθεί και τον δαγκώσει κάποιος λύκος και χάσει ενέργεια. Έτσι λοιπόν, στο κάτω αριστερά μέρος της οθόνης εμφανίζεται μια ημιδιαφανή μπάρα που αναπαριστά την ενέργεια του παίχτη (σχήμα 4.15).

Εκτός από τα HUDs χρησιμοποιήθηκαν διάφορα μηνύματα, ειδοποιήσεις και παράθυρα τα οποία εμφανίζονται στον χρήστη είτε κατά την έναρξη, είτε κατά τη διάρκεια, είτε κατά τη λήξη του παιχνιδιού και αναλύονται στη συνέχεια:

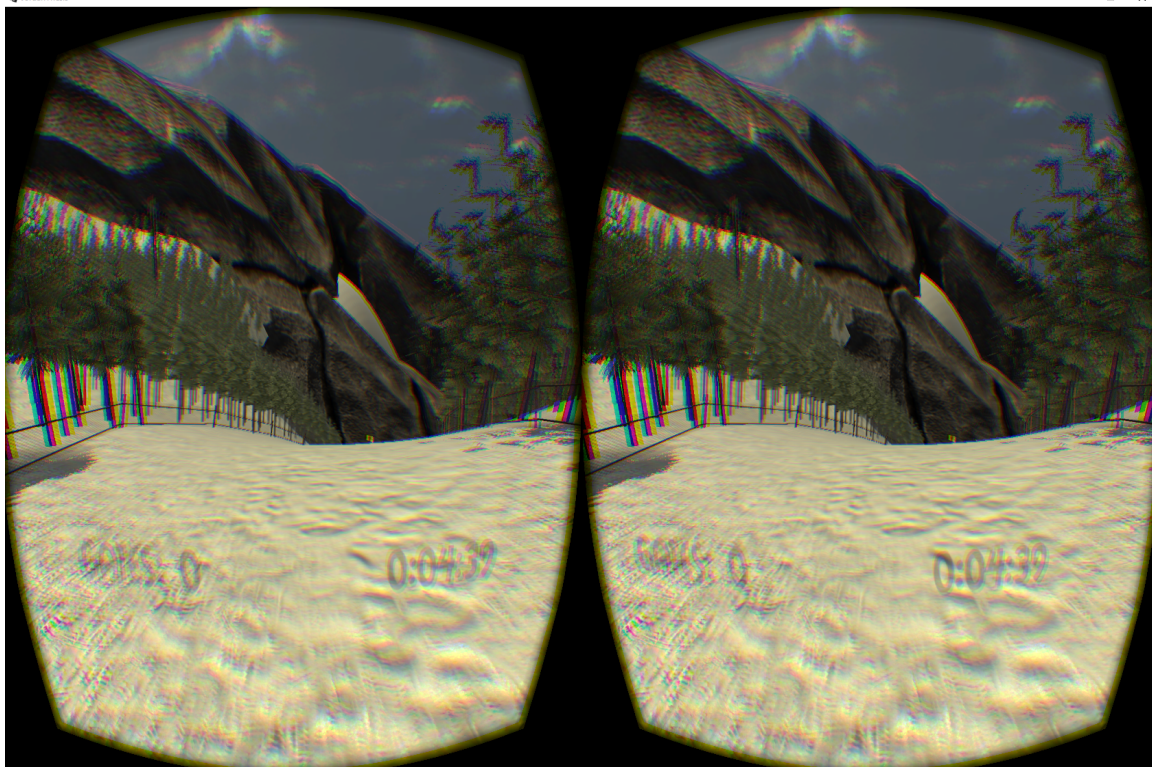
- **Παράθυρα Εκκίνησης:** Κατά την έναρξη του παιχνιδιού, σε οποιαδήποτε δοκιμασία,



Σχήμα 4.13: Μενού οδηγιών

εμφανίζεται στον παίκτη ένα αρχικό παράθυρο, το οποίο περιέχει είτε ορισμένες βασικές οδηγίες, είτε τις επιδόσεις των αντιπάλων του, ώστε να ξέρει ο παίκτης τους ακριβείς στόχους του. Στο σημείο αυτό ο χρήστης επιλέγει είτε να ξεκινήσει το παιχνίδι, είτε να επιστρέψει στο βασικό μενού σε περίπτωση που έχει μετανιώσει. Ο τρόπος με τον οποίο γίνεται η επιλογή (Exit ή Continue) είναι ο ίδιος όπως και στα αρχικά μενού. Δηλαδή, τοποθετώντας το στόχο και περιμένοντας για 1-2 δευτερόλεπτα γίνεται η επιλογή (σχήμα 4.16).

- Παράθυρα Τερματισμού:** Όταν ο παίκτης φτάσει στον τερματισμό μιας πίστας υπάρχουν δυο ενδεχόμενα. Είτε να την έχει ολοκληρώσει επιτυχώς, είτε ανεπιτυχώς. Και στις δυο περιπτώσεις το σύστημα ενημερώνει τον χρήστη με κατάλληλα μηνύματα, τόσο για τη θέση που κατέλαβε, όσο και για τα στατιστικά του παιχνιδιού. Ανάλογα με την πίστα στην οποία βρισκόμαστε αυτά διαφέρουν. Για παράδειγμα, στο (Slalom Challenge) όταν ο χρήστης τερματίσει, εμφανίζεται ένα μήνυμα που τον ενημερώνει για τη θέση του, καθώς και για τους χρόνους που έκανε, τόσο ο ίδιος, όσο και οι αντίπαλοί του. Στο (Junior Tour Challenge) ενημερώνεται απλά για τη θέση την οποία κατέλαβε, καθώς μας ενδιαφέρει μόνο ποιος θα φτάσει πρώτος στον τερματισμό. Το (Large Hill Challenge) πραγματοποιείται σε 2 γύρους. Οπότε στον πρώτο γύρο εμφανίζεται ένα παράθυρο με τις αποστάσεις των αλμάτων του παίκτη μας, καθώς και των αντιπάλων του (σχήμα 4.17), ενώ κατά τον τερματισμό του δεύτερου γύρου εμφανίζεται ένας πίνακας τελικής κατάταξης και των 2 αλμάτων συνολικά που έχει πραγματοποιήσει ο παίκτης και οι αντίπαλοί του. Τέλος, στην τελευταία πίστα, εμφανίζεται ένα παράθυρο που ενημερώνει τον χρήστη ότι κατάφερε να τερματίσει το παιχνίδι (αν έχει περάσει ανάμεσα από όλα τα σημαία και δεν τον έχουν σκοτώσει οι λύκοι) ή ότι πρέπει να ξαναπροσπαθήσει γιατί πέρασε από λιγότερα σημαία απ' όσα έπρεπε.
- Μήνυμα ότι χάσαμε (Game Over Message):** Αν ο χρήστης πέσει πάνω σε κάποιο

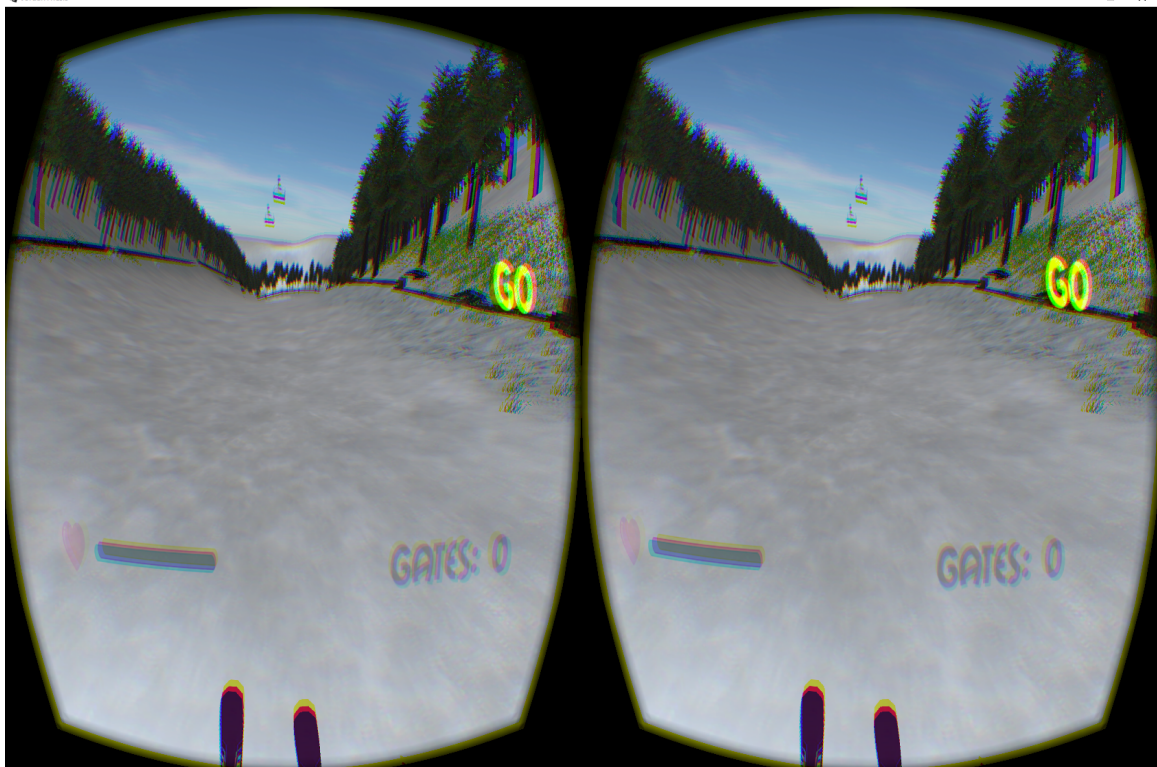


Σχήμα 4.14: HUDs στο Slalom Challenge, **Αριστερά:** Gate counter, **Δεξιά:** Timer

εμπόδιο, όπως προστατευτικές μπάρες, κολώνες των (Ski Lift), βράχους κλπ, ή προσγειωθεί άτσαλα στη δοκιμασία με τα άλματα ή τον σκοτώσουν οι λύκοι στην τελευταία πίστα (σχήμα 4.18), τότε εμφανίζεται στο χώρο ένα μήνυμα ότι έχασε. Οι επιλογές που έχει είναι είτε να επιστρέψει στο βασικό μενού, είτε να ξαναπροσπαθήσει. Η επιλογή (Restart, Continue) γίνεται με τον ίδιο τρόπο που έχει περιγραφεί προηγουμένως.

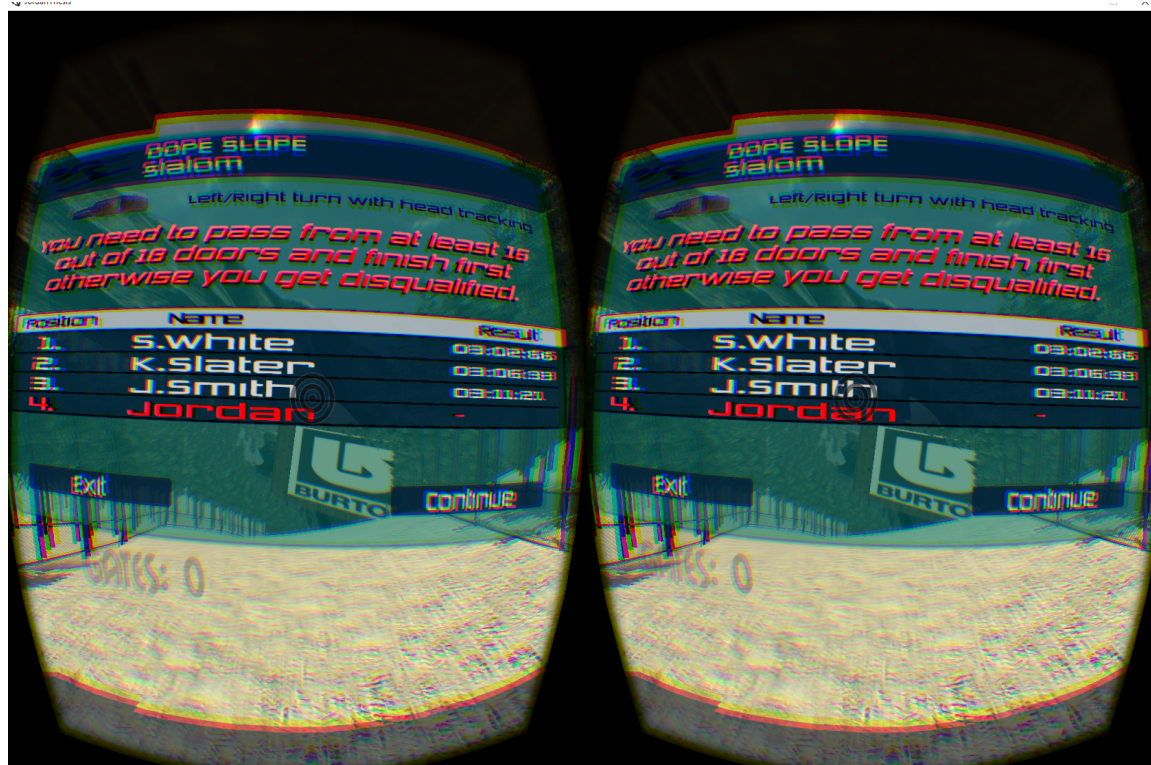
- **Μήνυμα ενημέρωσης άλματος:** Η δοκιμασία του άλματος αποτελείται από δυο γύρους. Τόσο κατά την εκκίνηση του άλματος (σχήμα 4.2 κάτω αριστερά), όσο και κατά την προσγείωση του άλματος εμφανίζεται μήνυμα που ενημερώνει σε ποιο άλμα βρίσκεται ο παίχτης και τι απόσταση πήδηξε, πριν εμφανιστεί το παράθυρο τερματισμού.
- **Μήνυμα προειδοποίησης (Speed-Up Notification):** Στην πίστα με τους αντίπαλους σκιέρ, ο παίχτης πρέπει να συλλέξει όσο πιο πολλά Speed-Ups βρίσκονται στην πίστα, ώστε να αυξήσει την ταχύτητά του. Λίγα μέτρα πριν φτάσει σ' αυτά, το σύστημα εμφανίζει ένα ημιδιαφανές μήνυμα που τον ενημερώνει ότι πλησιάζει (σχήμα 4.19).

Η μοναδική περίπτωση κατά την οποία ο χρήστης χρησιμοποιεί το πληκτρολόγιο είναι όταν θέλει να κάνει διακοπή παιχνιδιού (Pause Game) ενώ παίζει. Τα πλήκτρα που επιλέχτηκαν να χρησιμοποιηθούν είναι εύκολα στον εντοπισμό τους, ακόμα και με κλειστά μάτια, έτσι ώστε ο χρήστης να μπορεί να κάνει διακοπή, χωρίς να πρέπει πρώτα να βγάλει την κάσκα και να



Σχήμα 4.15: HUDs στο Wolves Slalom Challenge, **Αριστερά:** Energy, **Δεξιά:** Gate counter

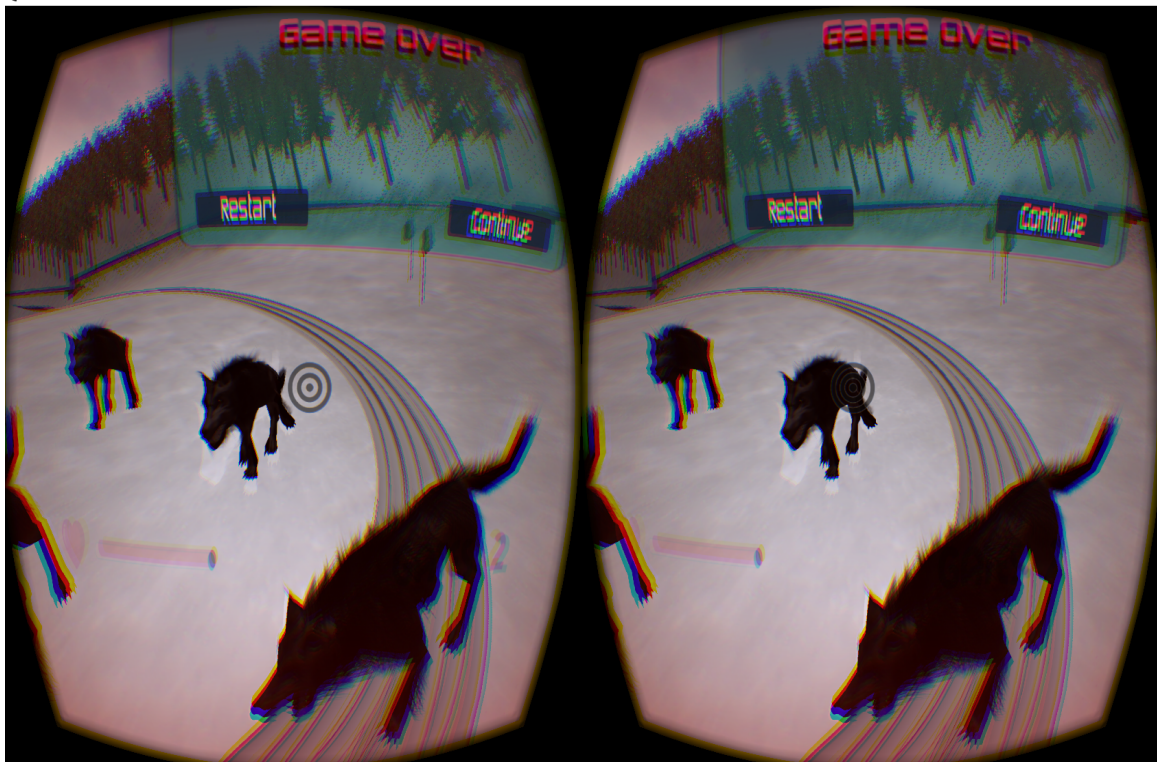
ψάξει το κουμπί διακοπής, ενώ η ροή του παιχνιδιού θα συνεχίζεται. Έτσι λοιπόν, οποιαδήποτε στιγμή ο χρήστης μπορεί να διακόψει πατώντας το πλήκτρο Esc. Οι επιλογές που του δίνονται είναι είτε να συνεχίσει το παιχνίδι πατώντας το αριστερό Ctrl, είτε να κάνει επανεκκίνηση του παιχνιδιού πατώντας Space, είτε έξοδο και επιστροφή στο βασικό μενού πατώντας εκ νέου Esc (σχήμα 4.20).



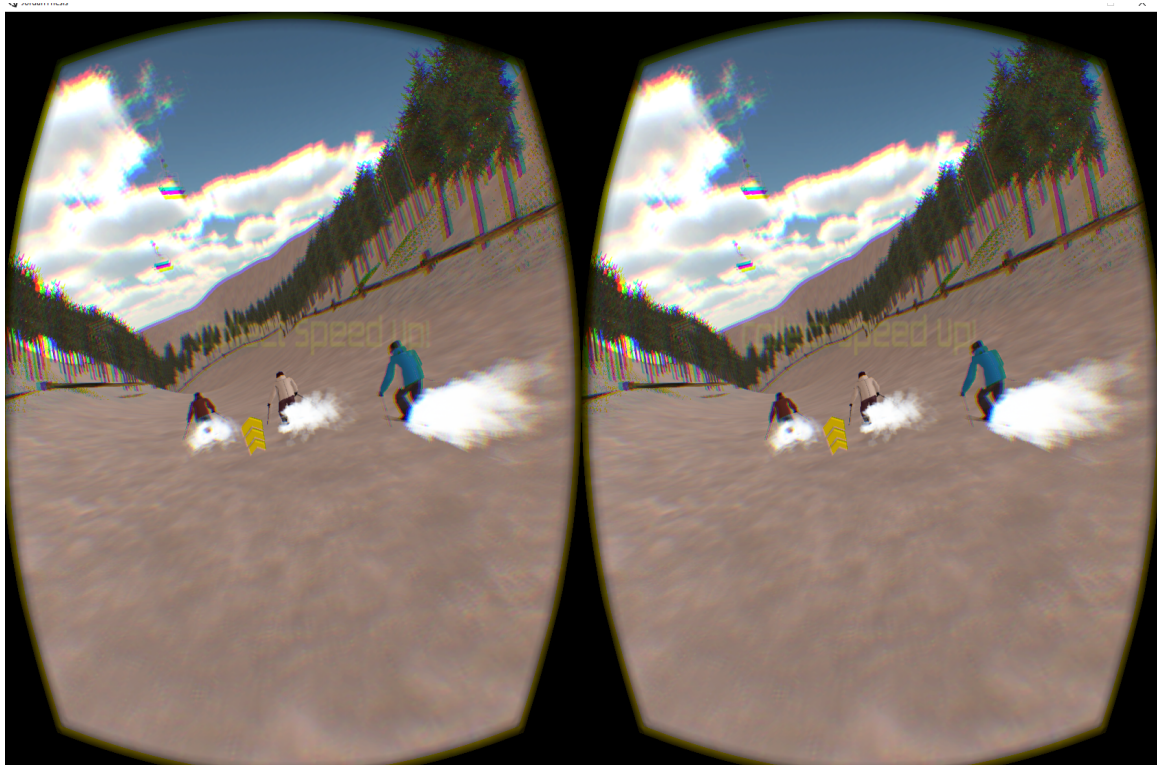
Σχήμα 4.16: Παράθυρο εκκίνησης στο Slalom Challenge



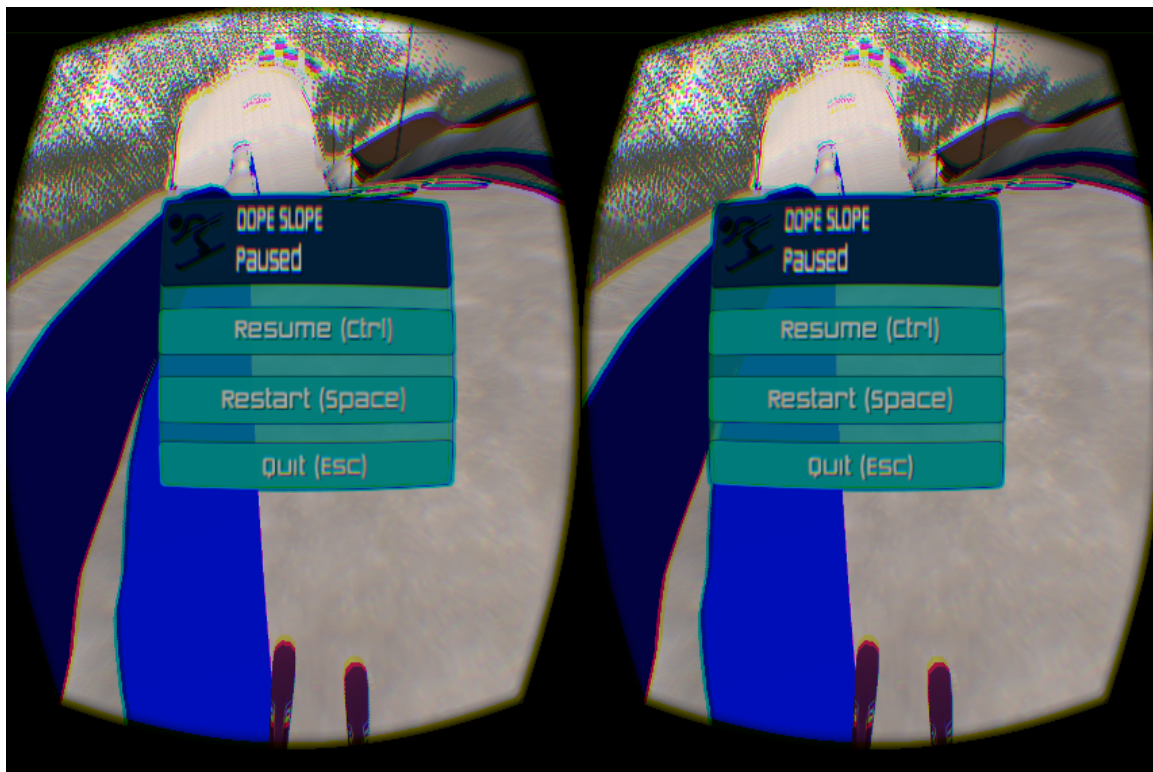
Σχήμα 4.17: Παράθυρο τερματισμού πρώτου γύρου στο Large Hill Challenge



Σχήμα 4.18: Παράθυρο *Game Over* που εμφανίζεται όταν οι λύκοι εντοπίσουν και σκοτώσουν τον σκιέρ (*Wolves Slalom Challenge*)



Σχήμα 4.19: Προειδοποιητικό Μήνυμα *Speed-Up*



Σχήμα 4.20: Μενού Παύσης (Pause Menu)

Κεφάλαιο 5

Υλοποίηση Παιχνιδιού

5.1 Εισαγωγή

Στο συγκεκριμένο κεφάλαιο θα γίνει αναλυτική περιγραφή των σταδίων υλοποίησης του παιχνιδιού. Θα γίνει αναφορά στις λειτουργίες της μηχανής παιχνιδιών και στο πως χρησιμοποιήθηκαν για να δοθεί ζωή και κίνηση στο παιχνίδι. Θα γίνει αναφορά στο προγραμματιστικό και τεχνικό κομμάτι που υλοποιήθηκε για τη δημιουργία των γραφικών, το χειρισμό του παίχτη, τη φυσική του παιχνιδιού, καθώς και την τεχνητή νοημοσύνη των αντιπάλων. Για να μπορεί να ελεγχθεί πλήρως η συμπεριφορά του παιχνιδιού και όλων των αντικειμένων, υλοποιήθηκε κώδικας γραμμένος σε C# και JavaScript.

5.2 Γραφική Διεπαφή Χρήστη

Η γραφική διεπαφή χρήστη έχει σχεδιαστεί με τέτοιο τρόπο ώστε να υπάρχει συνέπεια στα χρώματα και στη διαρρύθμισή της. Όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο, αρχικά σχεδιάστηκε με σκοπό να μπορεί ο χρήστης να την χειριστεί με το ποντίκι. Η ιδέα όμως αυτή γρήγορα εγκαταλείφθηκε, καθώς υποβαθμίζει το User Experience του χρήστη, ειδοποιώντας τον με μηνύματα για το πότε πρέπει να φορέσει την κάσκα και πότε να την βγάλει. Έτσι λοιπόν, το επόμενο βήμα ήταν να υλοποιηθεί όλη η εφαρμογή χρησιμοποιώντας αποκλειστικά και μόνο το Oculus Rift, με ελάχιστες εξαιρέσεις, όπως αναφέρθηκαν στο προηγούμενο κεφάλαιο.

Για την υλοποίηση, λοιπόν, της γραφικής διεπαφής χρησιμοποιήθηκε το σύστημα UI (User Interface) της Unity, το οποίο μας επιτρέπει να δημιουργήσουμε γρήγορα και διαισθητικά διεπαφές χρήστη. Αναπόσπαστο στοιχείο του UI είναι ο καμβάς (Canvas) μέσα στον οποίο τοποθετούνται όλα τα γραφικά στοιχεία. Δημιουργώντας ένα νέο στοιχείο UI, όπως για παράδειγμα μια

εικόνα ή ένα κουμπί, δημιουργείται αυτόματα ένας καμβάς, εάν δεν υπάρχει ήδη στη σκηνή. Το στοιχείο UI δημιουργείται πάντα ως παιδί μέσα στον καμβά.

5.2.1 Αρχικό Μενού

Μόλις το παιχνίδι ξεκινήσει, εμφανίζεται στην οθόνη ένα περιβάλλον μέσα στο οποίο υπάρχει το αρχικό παράθυρο επιλογών του παιχνιδιού (New Game, Load Game, Quit). Αρχικά λοιπόν, τοποθετήθηκαν εικόνες, γραμματοσειρές και διάφορα άλλα γραφικά στοιχεία πάνω στο παράθυρο του μενού. Στη συνέχεια, καθορίστηκε ο τρόπος επιλογής και αλληλεπίδρασης του παραθύρου αυτού με τον χρήστη, ο οποίος έγινε με ένα αρχείο script γραμμένο σε C# όπου χρησιμοποιήθηκαν οι τεχνικές του Collision Detection (ενότητα 3.4.10) και του Raycasting. Έτσι λοιπόν, πάνω σε κάθε μια επιλογή προσαρτήθηκε ένας ξεχωριστός Box Collider. Στη συνέχεια, μέσω της τεχνικής του Raycasting σχηματίζεται μια αόρατη γραμμή από το σημείο στο οποίο βρισκόμαστε (origin) και προς την κατεύθυνση που κοιτάμε, με σκοπό να εντοπίσει αν διασταυρώνεται (συγκρούεται) με κάποιον από τους Box Colliders. Σε περίπτωση σύγκρουσης, γίνεται αλλαγή χρώματος της επιλογής (άσπρο), έτσι ώστε να καταλάβει ο χρήστης ότι το έχει επιλέξει. Αν αφήσει τον στόχο για μερικά καρέ πάνω σε έναν συγκεκριμένο Box Collider, γίνεται φόρτωση.

Όπως φαίνεται και στο παρακάτω τμήμα κώδικα, η συνάρτηση Physics.Raycast(ray, out hit, 100) η οποία και χρησιμοποιήθηκε, παίρνει 3 ορίσματα, τα οποία είναι αναγκαία για να γίνει ο έλεγχος της σύγκρουσης:

- **ray:** Η αρχική θέση και διεύθυνση του σημείου από το οποίο ξεκινάει η αόρατη ακτίνα.
- **hit:** Αν είναι αληθές (true), περιλαμβάνει πληροφορίες για το σημείο στο οποίο έγινε η σύγκρουση.
- **distance:** Η μέγιστη απόσταση που η αόρατη ακτίνα μπορεί να φτάσει.

```
//Raycast Control for IntroMenu
void IntroFunction()
{
    // handle user view
    Ray ray = new Ray(cameraController.centerEyeAnchor.position,
        cameraController.centerEyeAnchor.forward);
    RaycastHit hit = new RaycastHit();

    //Check For Collision
```

```

if (IntroCube1.GetComponent<Collider>().Raycast (ray, out hit, 100.0f))
{
    if (soundplayed==false)
    {
        //Hover sound play
        HoverSound.Play();
    }
    soundplayed=true;

    //Setting counters
    Introcounter1+=1;
    Introcounter2=0;
    Introcounter3=0;

    //Waiting period before clicking
    if (Introcounter1==120)
    {
        //Click sound play
        ClickSound.Play();
        PlayerPrefs.DeleteAll();
        IntroMenu.SetActive (false);
        StartMenu.SetActive (true);
        ChallengeMenu.SetActive(false);
        menuflag=2;
    }

    //Change Button color
    IntroImg1.GetComponent<Button>().enabled=false;
}

//Check For Collision
else if (IntroCube2.GetComponent<Collider>().Raycast (ray, out hit, 100.0f))
{
    if (soundplayed==false)
    {
        HoverSound.Play();
    }
    soundplayed=true;
}

```

```

//Setting counters
Introcounter1=0;
Introcounter2+=1;
Introcounter3=0;

//Waiting period before clicking
if(Introcounter2==120)
{
    ClickSound.Play();
    IntroMenu.SetActive (false);
    StartMenu.SetActive (true);
    ChallengeMenu.SetActive(false);
    menuflag=2;
}

IntroImg2.GetComponent<Button>().enabled=false;
}
//Check For Collision
else if (IntroCube3.GetComponent<Collider>().Raycast (ray, out hit, 100.0f))
{
    if (soundplayed==false)
    {
        HoverSound.Play();
    }
    soundplayed=true;

//Setting counters
Introcounter1=0;
Introcounter2=0;
Introcounter3+=1;

//Waiting period before clicking
if(Introcounter3==120)
{
    ClickSound.Play();
    Application.Quit();
}

```

```

    }
    IntroImg3.GetComponent<Button>().enabled=false;
}
//If there is no collision (target outside of the window)
else
{
    soundplayed=false;
    Introcounter1=0;
    Introcounter2=0;
    Introcounter3=0;
    IntroImg1.GetComponent<Button>().enabled=true;
    IntroImg2.GetComponent<Button>().enabled=true;
    IntroImg3.GetComponent<Button>().enabled=true;
}
}

```

Με την ίδια ακριβώς λογική υλοποιήθηκαν και οι συναρτήσεις StartFunction(), TrainingFunction(), ChallengesFunction(), InstructionsFunction() και ExitFunction() παρέχοντας στο χρήστη μια απλή και γρήγορη αλληλεπίδραση με τα μενού που βρίσκονται εμβυθισμένα στο περιβάλλον εικονικής πραγματικότητας.

5.2.2 Μενού Κυρίως Παιχνιδιού

Η υλοποίηση της γραφικής διεπαφής του χρήστη κατά τη διάρκεια του παιχνιδιού ακολουθεί την ίδια φιλοσοφία με αυτή της σχεδίασης του αρχικού μενού που περιγράφηκε στην προηγούμενη ενότητα. Κάθε σκηνή, διαθέτει ένα script το οποίο ακολουθεί την ίδια ακριβώς λογική (Raycasting) για να επιτύχει την αλληλεπίδραση με το χρήστη. Αυτό που διαφέρει από πλευράς υλοποίησης σε σχέση με το αρχικό μενού, είναι ότι πολλές φορές τα παράθυρα ενεργοποιούνται από γεγονότα που λαμβάνουν χώρα κατά τη διάρκεια του παιχνιδιού. Έτσι, παράθυρα όπως της έναρξης της δοκιμασίας, του τερματισμού, της λήξης επειδή χάσαμε, είναι αναδυόμενα και εμφανίζονται αυτόματα ανάλογα με την εξέλιξη του παιχνιδιού. Επιπλέον, τα παράθυρα αυτά εμφανίζονται πάντα, σε σχετική ως προς τον σκιερ θέση και πάντα σε θέση που βρίσκεται ακριβώς μπροστά του.

Ο τρόπος με τον οποίο εμφανίζονται βασίζεται στους ενεργοποιητές (triggers). Στην ουσία, δηλαδή, υπάρχουν Box Colliders αόρατοι, με τους οποίους όταν ο σκιερ συγκρουστεί, έχουν σαν αποτέλεσμα την εμφάνιση συγκεκριμένων παραθύρων. Η συνάρτηση που χρησιμοποιήθηκε

για να πραγματοποιείται αυτός ο έλεγχος είναι η `OnTriggerEnter()`.

Η μόνη διαφορά από πλευράς υλοποίησης έγκειται στην περίπτωση που ο παίχτης πατήσει το `Esc` για παύση ή χάσει στην πίστα `Large Hill`. Σε αυτές τις δυο περιπτώσεις έγινε έλεγχος πλήκτρων χρησιμοποιώντας τη συνάρτηση `Input.GetKeyDown()`.

Τέλος, ένα πράγμα το οποίο είναι πάρα πολύ σημαντικό στη γραφική διεπαφή, είναι η προτεραιότητα των παραθύρων. Στην περίπτωση που πατήσουμε `Esc` για παύση παιχνιδιού και στη συνέχεια πατήσουμε εκ νέου `Esc` για έξοδο, θα εμφανιστούν 2 παράθυρα. Από κάτω θα βρισκείται το παράθυρο παύσης και από πάνω το παράθυρο επιβεβαίωσης εξόδου. Το ποιο παράθυρο θα εμφανιστεί από πάνω ή από κάτω, μπορεί να καθοριστεί από την παράμετρο `sortingOrder` που μας παρέχουν τα `Canvas` (σχήμα 5.1).



Σχήμα 5.1: Προτεραιότητα παραθύρων

5.3 Υλοποίηση Σκιέρ(Skier Physics)

Ένα σύγχρονο παιχνίδι επιβάλλεται να χρησιμοποιεί ένα ρεαλιστικό μοντέλο φυσικής. Στην παρούσα πτυχιακή εργασία έγινε προσπάθεια να προσομοιωθεί η κίνηση και η συμπεριφορά ενός σκιέρ με βάση τους νόμους της φυσικής. Έτσι λοιπόν, εκτός από τη βαρύτητα, έγινε έλεγχος συγκρούσεων μεταξύ όλων των αντικειμένων. Σε περίπτωση που δεν συνέβαινε αυτό, θα υπήρχαν αντικείμενα που το ένα θα εισχωρούσε στο άλλο χωρίς καμία αντίδραση. Συνεπώς, χρησιμοποιήθηκαν συγκρουστές σε όλα τα αντικείμενα οι οποίοι είτε είναι προσαρμοσμένοι στο αντικείμενο για μεγαλύτερη ακρίβεια υπολογισμών, είτε είναι απλά γεωμετρικά στερεά όπως σφαίρες, κύβοι, κύλινδροι κ.ά. Ύστερα από δοκιμή διαφόρων τύπων συγκρουστών στα πέδιλα του σκιέρ, αποφασίστηκε ότι ο βέλτιστος τρόπος ήταν να μοντελοποιηθεί ακριβώς όπως ένα αυτοκίνητο. Στην επόμενη υποενότητα γίνεται περαιτέρω ανάλυση.

5.3.1 Φυσική Σκιέρ (Skier Physics)

Για τη μοντελοποίηση του σκιέρ έγινε χρήση των συγκρουστών ρόδας (Wheel Colliders), οι οποίοι να μεν υπάρχουν αλλά δεν φαίνονται. Αυτό έχει σαν αποτέλεσμα ο σκιέρ να κινείται και να συμπεριφέρεται όπως ένα αυτοκίνητο με βασική όμως διαφορά ότι η μόνη κινητήριος δύναμη που του ασκείται είναι η επιτάχυνση της βαρύτητας (στα αυτοκίνητα μεταφέρεται ροπή στους άξονες, στο διαφορικό και τέλος στους τροχούς). Ο Wheel Collider αποτελεί μια ειδική κατηγορία συγκρουστών, τόσο για οχήματα, όσο και για άλλα αντικείμενα με παρόμοιες συμπεριφορές. Έχει ενσωματωμένη ανίχνευση σύγκρουσης, φυσική του τροχού και ένα μοντέλο τριβής-ολίσθησης. Αφού έγινε δοκιμή διάφορων τιμών για τις παραμέτρους, βρέθηκαν οι βέλτιστες δυνατές τιμές ώστε να υπάρχει ικανοποιητική συμπεριφορά (σχήμα 5.2). Έτσι λοιπόν, δημιουργήθηκαν τέσσερις τέτοιοι συγκρουστές και τοποθετήθηκαν σε μεγαλύτερη απόσταση από αυτή των πέδινων του σκι έτσι ώστε ο σκιέρ να είναι πιο σταθερός και να μην ανατρέπεται σε απότομες στροφές (σχήμα 5.3). Επιπλέον, για ακόμα μεγαλύτερη σταθερότητα και προς αποφυγήν ανατροπής σε περίπτωση απότομης στροφής, χαμηλώθηκε το κέντρο βάρους του με τον κώδικα που φαίνεται παρακάτω.

```
//Place the mass center under the ground
```

```
rb = GetComponent<Rigidbody>();
```

```
rb.centerOfMass= new Vector3 (0, -1, 0);
```

Στην δοκιμασία που ο παίχτης εκτελεί άλματα (Large Hill), αρχικά ήταν δύσκολο να γίνει ομαλή προσγείωση του σκιέρ με αποτέλεσμα τις περισσότερες φορές να ανατρέπεται. Για τον λόγο λοιπόν αυτό, χρησιμοποιήσαμε Coroutines. Οι Coroutines είναι εξαιρετικά χρήσιμες όταν γίνεται μοντελοποίηση μιας συμπεριφοράς κατά τη διάρκεια αρκετών καρέ. Οι Coroutines δεν έχουν σχεδόν καμία επιβάρυνση στην απόδοση και επιστρέφουν πάντα αμέσως, αλλά είναι εφικτό να κάνουν yield το αποτέλεσμα και να περιμένουν μέχρι να ολοκληρωθεί η εκτέλεση του αποτελέσματος. Έτσι λοιπόν, με την εντολή StartCoroutine("MyMethod"); γίνεται κλήση της Coroutine, ο κώδικας της οποίας φαίνεται παρακάτω.

```
//Freeze rotation
```

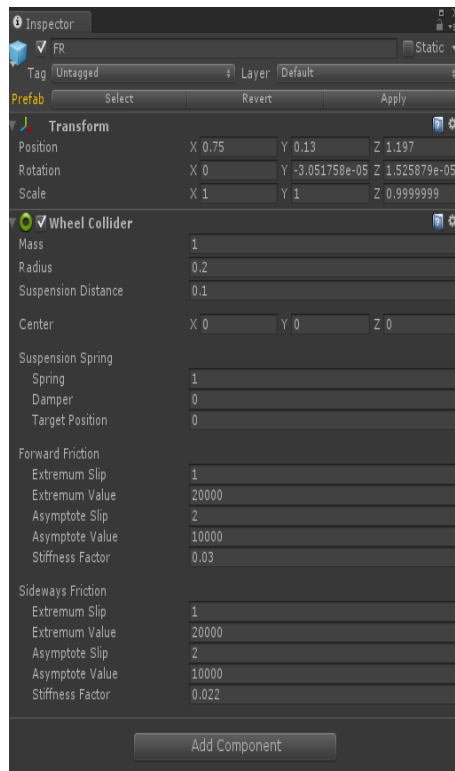
```
IEnumerator MyMethod() {
```

```
    temp99=0;
```

```
    rb.angularDrag=10000; //Freeze rotations in order to stabilize skier
```

```
    yield return new WaitForSeconds (2);
```

```
    rb.angularDrag=0; //After 2 seconds skier should be stabilized so we initialize
```



Σχήμα 5.2: Wheel Collider

```

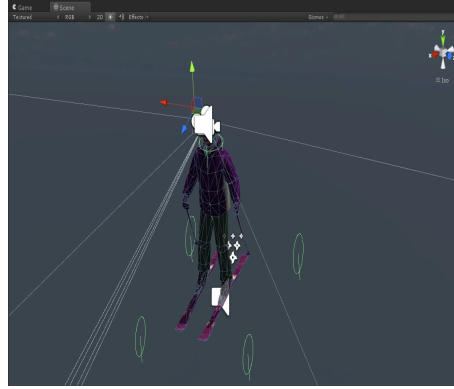
again
temp99 = 1;
}

```

Τη στιγμή της επαφής με το έδαφος παγώνουμε όλες τις περιστροφές του σκιέρ για 2 δευτερόλεπτα, ώστε να μην υπάρχουν αναπηδήσεις και να μπορέσει ο παίχτης να τον προσγειώσει. Μετά από 2 δευτερόλεπτα, αρχικοποιούμε εκ νέου την αντίσταση περιστροφής στο 0 και επανέρχεται ο φυσιολογικός έλεγχος.

5.3.2 Ενσωμάτωση Oculus Rift DK2 & Χειρισμός Σκιέρ

Αφού τοποθετήθηκαν οι συγκρουστές, επόμενο βήμα ήταν ο έλεγχος της διεύθυνσης κίνησης του σκιέρ. Όπως αναφέρθηκε και σε προηγούμενα κεφάλαια, ο έλεγχος αυτός γίνεται αποκλειστικά και μόνο με το Oculus Rift. Αρχικά λοιπόν, έγινε εισαγωγή του κατάλληλου πακέτου στη Unity, έτσι ώστε να έχουμε πρόσβαση στους μετασχηματισμούς του Oculus. Στη συνέχεια, αφαιρέθηκε η βασική κάμερα του παιχνιδιού (Main Camera) και προστέθηκε το Prefab του Oculus, το οποίο στην ουσία είναι δυο κάμερες με διαφορετική θέα, τις εικόνες των οποίων συνδυάζει το ανθρώπινο μυαλό για να δημιουργήσει μια ενιαία τρισδιάστατη στερεοσκοπική ει-



Σχήμα 5.3: Μοντέλο σκιέρ με συγκρουστές

κόννα. Έτσι, χρησιμοποιώντας την τρέχουσα τιμή περιστροφής ως προς τον άξονα Z από το γυροσκόπιο του Oculus, η οποία προκύπτει από περιστροφή του κεφαλιού ως προς τον άξονα αυτό, αντιστοιχήθηκε στη γωνία περιστροφής των μπροστινών τροχών του σκιέρ. Με τον τρόπο αυτό είχαμε άμεση αντιστοιχία μεταξύ περιστροφής κεφαλιού και διεύθυνσης σκιέρ. Επιπλέον, πολλαπλασιάστηκε επί 60 για να ρυθμίσουμε καλύτερα το turning ability του σκιέρ. Δοκιμάζοντας διάφορες τιμές θεωρήθηκε ότι η τιμή του 60 ήταν η ιδανική, καθώς σε μεγάλες ταχύτητες θέλουμε μικρότερη ικανότητα στροφής για να υπάρχει μεγαλύτερη σταθερότητα στην πορεία. Ο κώδικας ο οποίος υλοποιεί την παραπάνω λειτουργία φαίνεται στη συνέχεια.

```
wheelFL.steerAngle = -60 * oculusCamera.oculusRotation.z;
wheelFR.steerAngle = -60 * oculusCamera.oculusRotation.z;
```

Ο παραπάνω τρόπος κίνησης έχει υλοποιηθεί για την περίπτωση που ο σκιέρ βρίσκεται σε επαφή με το έδαφος. Στην περίπτωση όμως που ο σκιέρ χάσει την επαφή του με το έδαφος, όπως για παράδειγμα συμβαίνει στη δοκιμασία Large Hill, τη στιγμή που έχει φύγει από το άλμα και βρίσκετε στον αέρα, ο χειρισμός του σκιέρ διαφοροποιείται. Για το λόγο αυτό, πρώτα είναι αναγκαίος ο έλεγχος πότε βρίσκετε σε επαφή με το έδαφος και πότε όχι. Οι παρακάτω γραμμές κώδικα υλοποιούν αυτή τη διαδικασία.

```
//Check every Wheel collider if it collides with the ground
grounded1=wheelFR.GetGroundHit(out outHit);
grounded2=wheelFL.GetGroundHit(out outHit);
grounded3=wheelBR.GetGroundHit(out outHit);
grounded4=wheelBL.GetGroundHit(out outHit);
grounded = (grounded1 || grounded2 || grounded3 || grounded4);
```



```

if(grounded==true)
{
... //It includes many lines of code
}
else
{
... //It includes many lines of code
}

```

Στην περίπτωση, λοιπόν, που ο σκιέρ βρίσκεται στον αέρα, η περιστροφή του σώματός του ακολουθεί την περιστροφή του κεφαλιού (Oculus) ως προς τους άξονες X,Y,Z κάτι το οποίο ισχύει και στην πραγματικότητα όπου το σώμα ακολουθεί τις κινήσεις και τις περιστροφές του κεφαλιού. Έτσι λοιπόν, μέσω της συνάρτησης Rotate() γίνεται περιστροφή του χαρακτήρα τόσες μοίρες, όσες έχει περιστραφεί το γυροσκόπιο του Oculus Rift. Παρακάτω φαίνεται και ο αντίστοιχος κώδικας.

```

//Air control with head rotations on X,Y,Z axis
transform.Rotate(Vector3.up*rotSpeed*Time.deltaTime*(oculusCamera.oculusRotation.y));
transform.Rotate(Vector3.forward*rotSpeed*Time.deltaTime*(oculusCamera.oculusRotation.z));
transform.Rotate(Vector3.right*rotSpeed*Time.deltaTime*(oculusCamera.oculusRotation.x));

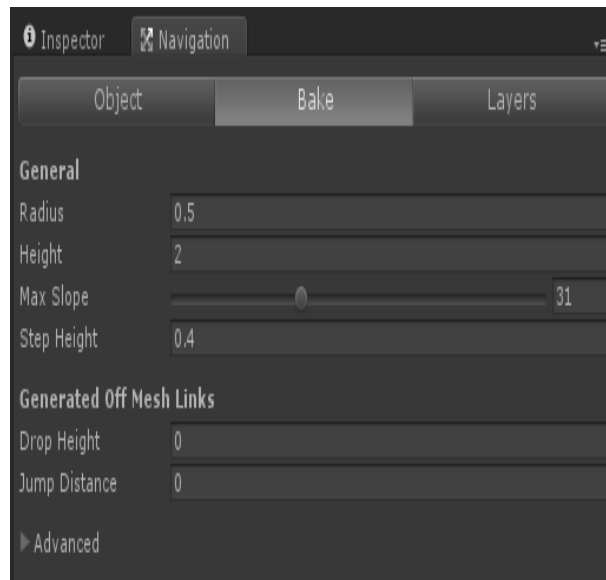
```

5.4 Τεχνητή Νοημοσύνη

Για την υλοποίηση της τεχνητής νοημοσύνης έγινε χρήση του Navigation System της Unity μέσω του οποίου γίνεται ανάγνωση του εδάφους της πίστας που μας ενδιαφέρει δημιουργώντας πλέγματα πλοήγησης, τα navigation meshes ή αλλιώς navmeshes. Στη συνέχεια, μέσω μιας διαδικασίας που ονομάζεται baking, γίνεται εισαγωγή του εδάφους (terrain) που μας ενδιαφέρει και καθορίζονται ορισμένοι παράμετροι που χρειάζεται να ληφθούν υπόψιν κατά την πλοήγηση του χαρακτήρα. Τέτοιες παράμετροι είναι οι παρακάτω (σχήμα 5.4).

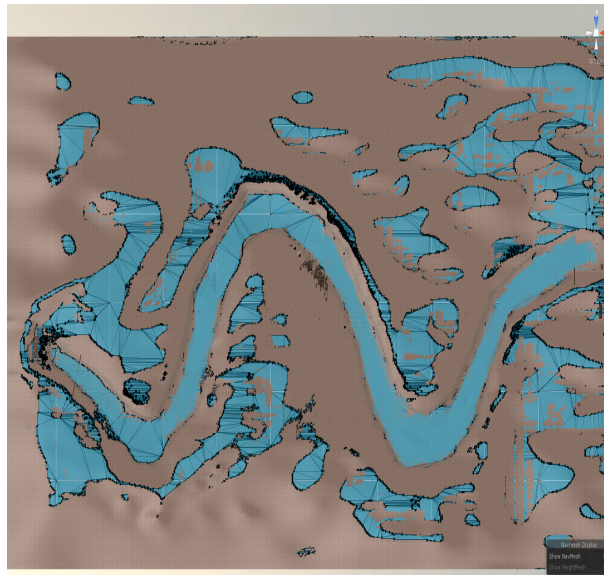
- **Ακτίνα του πράκτορα (Agent Radius):** Ορίζει πόσο κοντά μπορεί ο πράκτορας να πλησιάσει τοίχους.
- **Ύψος του πράκτορα (Agent Height):** Ορίζει μέχρι τι ύψος μπορεί ο πράκτορας να περάσει.

- **Μέγιστη κλίση (Max Slope):** Ορίζει την κλίση που μπορεί να κινηθεί.
- **Step Height:** Ορίζει πόσα ψηλά εμπόδια μπορεί να ανέβει.



Σχήμα 5.4: Παράμετροι NavMesh

Αφού καθοριστούν οι παράμετροι αυτοί ξεχωριστά για την μία πίστα που χρησιμοποιήθηκε στις δοκιμασίες Junior Tour και Wolves Slalom όπου έχουμε αυτόνομους χαρακτήρες, δημιουργήθηκε ένα πλέγμα το οποίο φαίνεται σαν μια γαλάζια διαφάνεια πάνω από το έδαφος (σχήμα 5.5).



Σχήμα 5.5: NavMesh

Για να μπορέσουν τα δεδομένα αυτά να διαβαστούν από τους χαρακτήρες μας και να μπορέσουν

να πλοηγηθούν μέσα στην πίστα ακολουθώντας το στόχο τους, πρέπει να τους εφαρμόσουμε το συστατικό NavMesh Agent. Ο NavMesh Agent είναι ένας κύλινδρος γύρω από τον χαρακτήρα, σαν Collider, ο οποίος είναι υπεύθυνος για την κίνησή τους. Στη συνέχεια, γίνεται περιγραφή των αυτόνομων αντιπάλων.

5.4.1 Αντίπαλοι Σκιέρ

Στη δεύτερη δοκιμασία ο παίχτης έρχεται αντιμέτωπος με άλλους 3 σκιέρ οι οποίοι λειτουργούν αυτόνομα. Έτσι λοιπόν, αφού τους προσαρτήθηκε ένας NavMesh Agent, δημιουργήθηκε ένα απλό script μέσα στο οποίο αρχικά χρησιμοποιήθηκε η GetComponent() για να υπάρχει πρόσβαση στο συστατικό NavMeshAgent και στη συνέχεια καθορίστηκε ο τελικός προορισμός, ο οποίος βρίσκεται στο τέλος της πίστας (Finish line), μέσω της συνάρτησης agent.SetDestination (target.position). Με τον τρόπο αυτό σχεδιάζεται αυτόματα το καλύτερο μονοπάτι προς τον προορισμό και ο πράκτορας αρχίζει να κινείται προς την κατεύθυνση αυτή.

```
using UnityEngine;
using System.Collections;

public class SampleAgentScript : MonoBehaviour {

    public Transform target;
    NavMeshAgent agent;
    // Use this for initialization
    void Start () {
        agent = GetComponent<NavMeshAgent> ();
    }

    // Update is called once per frame
    void Update () {
        //Final destination
        agent.SetDestination (target.position);
    }
}
```

Επιπλέον, δημιουργήθηκε ένα ακόμα script το AISkierControl, το οποίο ρυθμίζει την κίνηση του αυτόνομου παίχτη. Πιο συγκεκριμένα ο NavMeshAgent τον μεταφέρει από ένα σημείο σε

ένα άλλο δίχως όμως αυτός να ακολουθεί την καμπύλη και την κλίση του βουνού. Τον μετακινεί όπως θα τον μετακινούσε αν έπρεπε να τον μεταφέρει από ένα σημείο A σε ένα σημείο B πάνω στο επίπεδο. Για το λόγο, λοιπόν, αυτό σε κάθε καρέ υπολογίζουμε την κλίση του βουνού και τον περιστρέφουμε τις αντίστοιχες μοίρες, έτσι ώστε να έχει μια ομαλή πορεία συνεχώς, παράλληλη στην επιφάνεια του βουνού. Σε κάθε καρέ, λοιπόν, ο NavMeshAgent μετακινεί τον σκιέρ και το script AISkierControl, που υλοποιήθηκε τον περιστρέφει κατάλληλα. Ο τρόπος υπολογισμού της κλίσης βασίζεται σε δυο σημεία, το μπροστά και το πίσω μέρος των χιονοπέδων και η υλοποίηση φαίνεται παρακάτω. Η ταχύτητα των σκιέρ καθορίζεται στην αρχή του παιχνιδιού μέσω της συνάρτησης Random.Range(), έτσι ώστε να υπάρχει τυχαιότητα στο παιχνίδι. Οι πιθανές τιμές είναι υπολογισμένες να κυμαίνονται σε ένα συγκεκριμένο εύρος, ώστε να κάνουν το παιχνίδι σχετικά δύσκολο.

```
if (Physics.Raycast(Skier.transform.position + Skier.transform.forward,
    Vector3.down,out hit))
{
    var fHit = hit.point; // get the point where the front ray hits the ground
    if (Physics.Raycast(Skier.transform.position - Skier.transform.forward,
        Vector3.down,out hit))
    {
        var bHit = hit.point; // get the back hit point
        transform.forward = fHit - bHit; // align the object to these points
    }
}
```

5.4.2 Αντίπαλοι Λύκοι

Στην τελευταία δοκιμασία ο παίχτης πρέπει να περάσει ανάμεσα από σημαίες και να αποφύγει 20 πεντασμένους λύκους. Οι λύκοι είναι τοποθετημένοι σε μικρές αγέλες των τεσσάρων σε διάφορα σημεία της πίστας. Αρχικά, λοιπόν, προσαρτήθηκε ένας NavMesh Agent σε κάθε λύκο και στη συνέχεια δημιουργήθηκε ένα script, το AIWolves, το οποίο ελέγχει τη συμπεριφορά του κάθε λύκου. Χρησιμοποιήθηκε το ίδιο τμήμα κώδικα, όπως και για τους αντίπαλους σκιερ, ώστε ο κάθε λύκος να υπολογίζει την κλίση του terrain και να την ακολουθεί.

Ο κάθε λύκος μπορεί να είναι σε τρεις βασικές καταστάσεις:

- **Ανενεργή:** Στην κατάσταση αυτή ο λύκος στέκεται ακίνητος. Δεν έχει καμία οπτική με τον σκιέρ, ούτε έχει ακούσει κάποια ειδοποίηση από άλλο λύκο.

- **Ειδοποίησης:** Στην κατάσταση αυτή ο λύκος δεν βλέπει άμεσα τον σκιέρ (σκιέρ εκτός r2). Βρίσκεται όμως στο ορατό πεδίο κάποιου άλλου λύκου, ο οποίος και ειδοποιεί τον πρώτο λύκο ότι τον βλέπει με αποτέλεσμα ο πρώτος να κατευθύνεται προς τον δεύτερο.
- **Απειλής:** Στην κατάσταση αυτή ο λύκος βλέπει τον σκιέρ (μεταξύ τους απόσταση μικρότερη ή ίση του r1) με αποτέλεσμα να τρέχει κατά πάνω του.
- **Επίθεσης:** Ο λύκος βρίσκεται πλέον πολύ κοντά στον σκιέρ οπότε και του επιτίθεται με αποτέλεσμα ο παίχτης να χάνει ενέργεια.
- **Επιστροφής:** Στην κατάσταση αυτή ο λύκος επιστρέφει πίσω στη θέση παρακολούθησης.

Σε περίπτωση που η απόσταση μεταξύ λύκου και σκιέρ γίνει μεγαλύτερη από r1 και ο εκάστοτε λύκος δεν ειδοποιηθεί από κάποιον άλλο, επιστρέφει στη θέση παρακολούθησής του. Οι παραπάνω καταστάσεις αλλάζουν ανάλογα με την απόσταση του στόχου από τον αντίπαλο και ο τρόπος υλοποίησης τους φαίνεται στη συνέχεια.

```
void Update ()
{
    dist=agent.remainingDistance;
    if (Physics.Raycast (myTransform.position + myTransform.forward,
        Vector3.down, out hit))
    {
        var fHit = hit.point; // get the point where the front ray hits the ground
        if (Physics.Raycast (myTransform.position - myTransform.forward,
            Vector3.down, out hit))
        {
            var bHit = hit.point; // get the back hit point
            transform.forward = fHit - bHit; // align the object to these points
        }
    }

    //Distance calculation between wolf and skier
    AttackDistance = (targetAttack.position - myTransform.position).magnitude;

    //In this state the wolf has seen the skier and starts running towards him
    if ((AttackDistance>4) && (AttackDistance < dontComeCloserRange))
    {
        for(int j=0;j<StaticsScript.gameObjects.Length; j++)
        {
```

```

        if (gameObject.name==StaticsScript.gameObjects[j].name)
        {
            animation.CrossFade("Run"); //Wolf running animation
            StaticsScript . flagalerts [j] = 1; //Enables whistle for the other
            wolves
            Attack();
        }
        else
        {
            StaticsScript . flagalerts [j] = 0;
        }
    }
}
//In this state the wolf is very close to the skier and starts jump-attack
else if ((AttackDistance>0) && (AttackDistance<=4))
{
    animation.CrossFade("Jump_Attack");
    if (flag==1)
    {
        StartCoroutine("Method");
    }
}
//In this state the wolf is not able to see the skier
else
{
    // Initialize the whistle
    for (int k=0;k<StaticsScript.gameObjects.Length; k++)
    {
        if (gameObject.name==StaticsScript.gameObjects[k].name)
        {
            StaticsScript . flagalerts [k] = 0;
        }
    }
}

for (int p=0;p<StaticsScript.gameObjects.Length; p++)

```

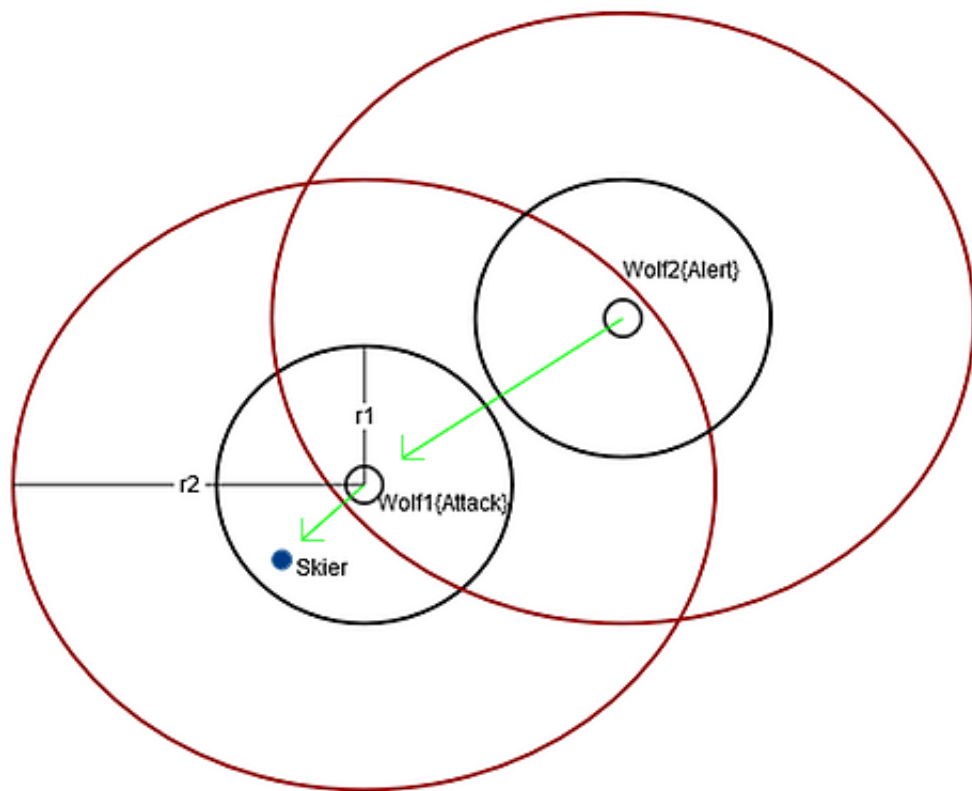
```

{
    //In this state the wolf hears the whistle from another wolf and goes
    towards him
    if (StaticsScript . flagalerts [p]==1 &&
        (((StaticsScript.transforms[p].position -
        myTransform.position).magnitude)>0 &&
        ((StaticsScript.transforms[p].position -
        myTransform.position).magnitude)<=100))
    {
        animation.CrossFade("Run");
        agent.SetDestination ( StaticsScript . transforms[p]. position);
        break;
    }
    else
    {
        if (gameObject.name==StaticsScript.gameObjects[p].name)
        {
            //In this state the wolf is on the patrol position
            if (dist==0)
            {
                animation.CrossFade("Idle");
            }
            //In this state the wolf returns back to the patrol position
            else
            {
                agent.SetDestination
                    ( StaticsScript . gameObjectsPatrol[p].transform.position);
            }
        }
    }
}
}
}
}

```

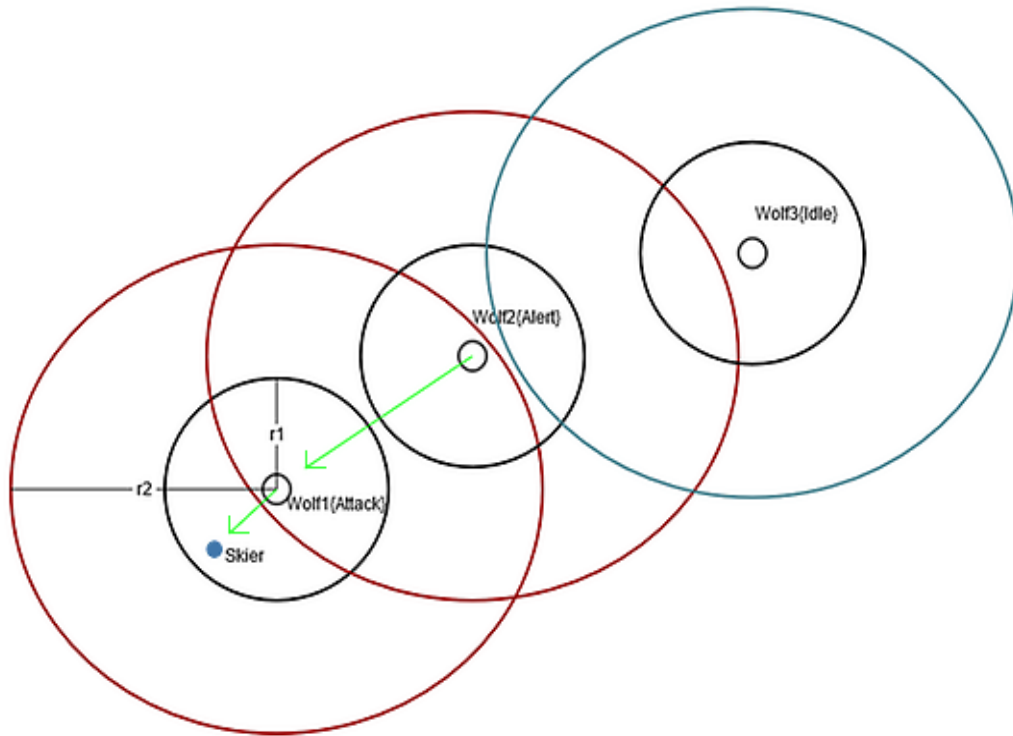
Η βασική λειτουργία του αλγόριθμου φαίνεται στο σχήμα 5.6. Όταν ο σκιέρ μπει στην περιοχή r1 του wolf1, τότε ο λύκος αρχίζει να κινείται κατά πάνω του. Παράλληλα, όσοι λύκοι βρίσκονται στην ακτίνα r2 του wolf1 ειδοποιούνται και αρχίζουν να κινούνται προς αυτόν. Όπως φαίνεται

και στην εικόνα ο wolf2 κινείται προς τον wolf1. Αν οποιαδήποτε στιγμή ο σκιέρ μπει στην ακτίνα $r1$ του wolf2, τότε αλλάζει προτεραιότητα ο wolf2 και δεν κινείται πλέον προς τον wolf1 αλλά προς τον σκιέρ. Ο τρίτος λύκος (wolf3) που φαίνεται στο σχήμα 5.7 παραμένει ακίνητος στη θέση παρακολούθησής του καθώς βρίσκεται έξω και από την ακτίνα του wolf2.



Σχήμα 5.6: Σύστημα διεπικοινωνίας αυτόνομων λύκων (A)

Με την τοποθέτηση των 20 λύκων στην πίστα μας οι υπολογισμοί ήταν πάρα πολλοί με αποτέλεσμα ο ρυθμός καρέ να πέσει κατακόρυφα και η ποιότητα του παιχνιδιού επίσης. Για το λόγο λοιπόν αυτό, χωρίστηκε η πίστα του παιχνιδιού σε 5 ζώνες τοποθετώντας 5 Box Colliders (συγκρουστές) αντίστοιχα. Στην περιοχή του κάθε συγκρουστή βρίσκοντουσαν διάσπαρτοι 4 λύκοι. Μόλις ο παίκτης ξεκινάει το παιχνίδι, όλοι οι λύκοι είναι ανενεργοί. Μόλις εντοπίσει τον πρώτο συγκρουστή, οι 4 πρώτοι λύκοι γίνονται ενεργοί και ο αλγόριθμος ξεκινάει να τρέχει. Για να μην εμφανίζονται στον παίκτη ξαφνικά μπροστά του, τοποθετήθηκαν με τέτοιο τρόπο στην πίστα ώστε μόλις ενεργοποιηθεί, να μην είναι άμεσα ορατοί. Αφού εξέλθει από την περιοχή του πρώτου συγκρουστή, οι λύκοι αυτοί καταστρέφονται για να μην καταναλώνουν περαιτέρω πόρους του επεξεργαστή κάνοντας υπολογισμούς, καθώς πρακτικά ο σκιέρ έχει απομακρυνθεί



Σχήμα 5.7: Σύστημα διεπικοινωνίας αυτόνομων λύκων (B)

από αυτούς και πλέον δεν κινδυνεύει. Στη συνέχεια, μόλις εισέλθει στο δεύτερο συγκρουστή οι επόμενοι 4 λύκοι ενεργοποιούνται και ακολουθείται ακριβώς η ίδια διαδικασία. Η λογική αυτή υλοποιήθηκε με τη συνάρτηση `OnTriggerEnter` ελέγχοντας κάθε φορά αν υπάρχει αλληλεπίδραση του συγκρουστή του σκιέρ με τον Box Collider της περιοχής.

```
void OnTriggerEnter(Collider other)
{
    if (other.gameObject.CompareTag("area1")) //If skier enters first area
    {
        areaflag=1;
        wolf1.SetActive(true);
        wolf2.SetActive(true);
        wolf3.SetActive(true);
        wolf4.SetActive(true);
        wolf5.SetActive(false);
        wolf6.SetActive(false);
        wolf7.SetActive(false);
        ...
    }
}
```

```

if (other.gameObject.CompareTag("area2")) //If skier enters second area
{
    areaflag=2;
    wolf1.SetActive(false);
    wolf2.SetActive(false);
    wolf3.SetActive(false);
    wolf4.SetActive(false);
    wolf5.SetActive(true);
    wolf6.SetActive(true);
    wolf7.SetActive(true);
    ...
}
... //Checks like the above one
}

```

5.5 Κινούμενα Σχέδια (Animations)

Για την κίνηση του παίχτη χρησιμοποιήθηκε ένα animation, το οποίο μοντελοποιεί την κίνηση του παίχτη όταν αυτός βρίσκεται ακίνητος. Το animation αυτό σχεδιάστηκε καρέ προς καρέ στο 3D Studio Max και στη συνέχεια προστέθηκε στη Unity. Το ίδιο animation χρησιμοποιήθηκε και κατά τη διάρκεια της κίνησης του παίχτη, έτσι ώστε να φαίνεται ότι υπάρχει φυσική κίνηση του σκιερ. Το animation περιλαμβάνει μικρές κινήσεις του κεφαλιού και του σώματος. Έτσι, λοιπόν χρησιμοποιώντας το Animation System της Unity και μέσω της εντολής `animation.CrossFade("SkiIdle");` έγινε αναπαραγωγή του συγκεκριμένου animation.

Σχετικά με τους αντίπαλους σκιέρ, σχεδιάστηκε στο 3D Studio Max ένα ακόμα animation, το οποίο περιγράφει την κίνηση του παίχτη όταν ολισθαίνει με μεγάλη ταχύτητα (είναι σκυφτός και με λυγισμένα πόδια). Έτσι λοιπόν, όταν περιμένουν στην εκκίνηση χρησιμοποιείται το `Ski Idle` animation, ενώ όταν αρχίσουν να κινούνται το `Ski Fast` animation. Η συνάρτηση `animation.CrossFade` χρησιμοποιήθηκε για να κάνει ομαλό blend από το ένα animation στο άλλο.

Οι αντίπαλοι λύκοι από την πλευρά τους, ανάλογα με την κατάσταση στην οποία βρίσκονται, αναπαράγουν διαφορετικό animation. Πιο συγκεκριμένα, όπως αναφέρθηκε σε προηγούμενη ενότητα οι λύκοι μπορούν να βρεθούν σε πέντε διαφορετικές καταστάσεις. Στην περίπτωση που κάποιος λύκος βρίσκεται σε ανενεργή κατάσταση, τότε αναπαράγεται το αντίστοιχο animation. Σε περίπτωση που βρεθεί είτε στην κατάσταση ειδοποίησης, είτε στην κατάσταση απειλής, είτε

στην κατάσταση επιστροφής αναπράγεται το Run animation. Τέλος, αν ο λύκος έρθει σε πολύ κοντινή απόσταση με τον σκιέρ (Κατάσταση επίθεσης), τότε αναπράγεται το κινούμενο σχέδιο της επίθεσης (Jump Attack animation). Για τη μείξη των παραπάνω κινούμενων σχεδίων χρησιμοποιήθηκε όπως και πριν η συνάρτηση animation.CrossFade με κλήση της στα κατάλληλα σημεία του κώδικα.

5.6 Σωματιδιακά τεχνάσματα εντυπωσιασμού (Particles Effects)

Τα σωματιδιακά τεχνάσματα εντυπωσιασμού χρησιμοποιούνται κατά κόρον στα σημερινά παιχνίδια και αυτό διότι καταναλώνουν σχετικά λίγους υπολογιστικούς πόρους, παρέχοντας παράλληλα ρεαλισμό. Στην παρούσα πτυχιακή εργασία χρησιμοποιήθηκαν για την αναπαράσταση της πούδρας του χιονιού κατά την κίνηση τόσο του δικού μας σκιερ, όσο και των αντιπάλων. Σχετικά με τους αντίπαλους σκιέρ, μόλις περάσουν την αρχή της εκκίνησης και αρχίσουν να κινούνται, ενεργοποιείται το αντικείμενο (GameObject) που είναι υπεύθυνο για τα τεχνάσματα εντυπωσιασμού μέσω της παρακάτω εντολής.

```
//Powder dust enable  
dust.enableEmission = true;
```

Όσον αφορά τον δικό μας χαρακτήρα, έγινε προσπάθεια να γίνουν περισσότεροι έλεγχοι ώστε να υπάρχει ρεαλιστικότερη συμπεριφορά. Αρχικά, προστέθηκε ένας Trail Renderer, ο οποίος αφήνει τη χαρακτηριστική γραμμή πίσω από το χιονοπέδιλο του σκιέρ. Η πούδρα στην πραγματικότητα παράγεται όταν ο σκιερ κάνει κάποια απότομη στροφή, ξαφνικά, δεδομένου ότι έχει κάποια ταχύτητα. Έτσι λοιπόν, γίνεται μέτρηση της ταχύτητας συνεχώς, ελέγχοντας αν βρίσκεται πάνω από ένα κατώφλι (20m/sec). Όσο είναι μικρότερη από τα 20m/sec το Particle System είναι ανενεργό. Σε περίπτωση που η ταχύτητα γίνει μεγαλύτερη, πραγματοποιείται έλεγχος ως προς τον άξονα Y του γυροσκοπίου του Oculus Rift (Head Tracking). Αν υπερβαίνει την τιμή 0.2 ή είναι μικρότερη από -0.2 τότε σημαίνει ότι ο παίχτης έχει πραγματοποιήσει μεγάλη στροφή οπότε και ενεργοποιούνται τα σωματιδιακά εφέ. Η υλοποίηση φαίνεται παρακάτω.

```
if (currentSpeed >= 0 && currentSpeed < 20)  
{  
    dust.enableEmission = false; //Powder dust deactivated  
}  
else  
{ //Check if there is head rotation
```

```

if (oculusCamera.oculusRotation.z > 0.2 || oculusCamera.oculusRotation.z < -0.2)
{
    dust.enableEmission = true; //Powder dust activated
}
else
{
    dust.enableEmission = false; //Powder dust deactivated
}
}

```

5.7 Ήχοι (Audio)

Βασικό στοιχείο το οποίο δίνει ρεαλισμό σε ένα παιχνίδι εικονικής πραγματικότητας, καθώς και την αίσθηση της εμβύθισης είναι αναμφίβολα ο ήχος. Τοποθετήθηκαν διάφοροι ήχοι τόσο στο περιβάλλον έναρξης όσο και κατά τη διάρκεια του παιχνιδιού. Τα τραγούδια καθώς και όλα τα ηχητικά εφέ προσκομήθηκαν από το διαδίκτυο και επεξεργάστηκαν για τις ανάγκες του παιχνιδιού. Οι ήχοι που χρησιμοποιήθηκαν συνολικά στο παιχνίδι φαίνονται παρακάτω:

- Μουσική του περιβάλλοντος έναρξης η οποία παίζει μέχρι να φορτωθεί κάποια δοκιμασία.
- Ήχος όταν κάνουμε hover (περνάμε από πάνω) πάνω από κάποιο κουμπί.
- Ήχος όταν επιλέγουμε κάποιο κουμπί.
- Δυο διαφορετικά ηχητικά εφέ εκκινήσεων τα οποία προειδοποιούν τον χρήστη ότι ξεκινάει.
- Ηχητικό κλιπ τερματισμού μιας πίστας.
- Ήχος πρόσκρουσης του παίχτη σε εμπόδια της πίστας, όπως περίφραξη, σπιτόκια, κολώνες και βράχοι.
- Ήχος παίχτη που αναπαράγεται σε κάθε δάγκωμα του λύκου.
- Ήχος παίχτη που αναπαράγεται όταν οι λύκοι τον σκοτώσουν.
- Ηχητικό εφέ αέρα.
- Ηχητικό εφέ γρυλίσματος λύκων.
- Ηχητικό εφέ επίθεσης λύκων.
- Ήχος από τα πόδια του σκι.

Εκτός από τους ήχους του κεντρικού μενού, όλοι οι υπόλοιποι είναι στερεοφωνικοί. Στερεοφωνικός ήχος ή , συνηθέστερα , στέρεο ήχος, είναι μια μέθοδος αναπαραγωγής ήχου που δημιουργεί την ψευδαίσθηση ότι ο ήχος προέρχεται από πολλαπλές κατευθύνσεις. Αυτό επιτυγχάνεται συνήθως με τη χρήση δύο ή περισσότερων ανεξάρτητων καναλιών ήχου μέσω της ρύθμισης δύο ή περισσότερων μεγαφώνων (ή στερεοφωνικών ακουστικών) με τέτοιο τρόπο ώστε να δημιουργούν την εντύπωση ότι ακούγεται από διάφορες κατευθύνσεις , όπως στη φυσική ακοή. Είναι συχνά σε αντίθεση με τους μονοφωνικούς ήχους του κεντρικού μενού, όπου ακούγονται ως ήχοι προερχόμενοι από μία θέση, συχνά στο κέντρο του ηχητικού πεδίου. Συνεπώς, σε ένα περιβάλλον εικονικής πραγματικότητας, όπου θέλουμε να επιτύχουμε το αίσθημα του Immersion και του Presence, ο στερεοφωνικός ήχος αποτελεί βασικό στοιχείο.

Για να δωθεί καλύτερα η αίσθηση της εμβύθισης στο εικονικό περιβάλλον, ρυθμίστηκε η ένταση του ήχου των πέλδων του σκι σε συνάρτηση με την ταχύτητα του σκιέρ. Όσο πιο γρήγορα κινείται, τόσο μεγαλύτερη η ένταση του ήχου. Ο κώδικας που υλοποιεί αυτή τη διαδικασία φαίνεται παρακάτω.

```
//Speed Calculation
```

```
currentSpeed=2*22/7*wheelBL.radius*wheelBL.rpm*60/1000;  
currentSpeed=Mathf.Round(currentSpeed);
```

```
//Volume in relation with currentSpeed
```

```
temp=currentSpeed/60;
```

```
if (temp>1)
```

```
{  
    temp=1f;
```

```
}
```

```
else
```

```
{  
    temp=temp;
```

```
}
```

```
Sound1.volume=temp;
```

Όπως φαίνεται, αρχικά υπολογίζουμε την ταχύτητα του σκιέρ, στη συνέχεια την κανονικοποιούμε και τέλος, την εκχωρούμε στην ένταση της πηγής του συγκεκριμένου ήχου.

Τέλος, για τη διαχείριση της μουσικής του παιχνιδιού, δημιουργήθηκε ένα script το GameMusicController, το οποίο σταματάει την αναπαραγωγή της μουσικής κάθε φορά που φορτώνεται κάποια σκηνή στο παιχνίδι.

```
if (Application.loadedLevelName=="Slalom")  
{  
    //music.Stop();  
}
```

Κεφάλαιο 6

Αξιολόγηση Συστήματος

6.1 Εισαγωγή

Στο κεφάλαιο αυτό παρουσιάζουμε τον τρόπο με τον οποίο αξιολογήθηκε η εφαρμογή του παιχνιδιού από διάφορους χρήστες. Ειδικότερα εξετάστηκε ο βαθμός εμπύθυνσης immersion που δημιουργεί το βιντεοπαιχνίδι, μελετώντας διάφορες παραμέτρους, καθώς και το simulator sickness που πιθανόν προκλήθηκε σε ορισμένους από αυτούς.

6.2 Μέθοδος Αξιολόγησης

Για την αξιολόγηση του συστήματος έγινε χρήση του ερωτηματολογίου χρηστικότητας Measuring and Defining the Experience of Immersion in Games[40] καθώς και του ερωτηματολογίου Simulator Sickness Questionnaire[41], ώστε να προσδιοριστεί ο βαθμός εμπύθυνσης που προκαλεί το παιχνίδι και πιθανά συμπτώματα ζάλης και δυσφορίας αντίστοιχα.

Στη διαδικασία αξιολόγησης συμμετείχαν 14 χρήστες (8 άντρες και 6 γυναίκες), ηλικίας από 21 έως 26 ετών η πλειονότητα των οποίων ήταν φοιτητές(προπτυχιακοί, μεταπτυχιακοί) του Πολυτεχνείου Κρήτης.

Το ερωτηματολόγιο περιελάμβανε 48 ερωτήσεις σύνολο, χωρισμένο σε 2 διαφορετικά υποερωτηματολόγια όπως αναφέρθηκε παραπάνω. Οι απαντήσεις δίνονταν με κλίμακα από το 1 έως το 5 για το πρώτο ερωτηματολόγιο. Για το δεύτερο, οι δυνατές απαντήσεις ήταν 4 (Καθόλου=0, Ελαφρώς=1, Μέτρια=2, Πολύ=3). Στη συνέχεια παρατίθεται το ερωτηματολόγιο που χρησιμοποιήθηκε.

ΕΡΩΤΗΜΑΤΟΛΟΓΙΟ
IMMERSION & SIMULATOR SICKNESS QUESTIONNAIRE

Αριθμός ερ/γίου: _____

Ηλικία: _____

Φύλο: Άρρεν _____ Θήλυ _____

IMMERSION QUESTIONNAIRE

Οδηγίες: Να απαντήσετε στις ακόλουθες ερωτήσεις κυκλώνοντας τον σχετικό αριθμό. Πιο συγκεκριμένα, να θυμάστε ότι αυτές οι ερωτήσεις σας ζητάνε να απαντήσετε για το πως αισθανθήκατε στο τέλος του παιχνιδιού.

1. Σε ποιο βαθμό το παιχνίδι σας τράβηξε την προσοχή;
Καθόλου 1 2 3 4 5 Πολύ
2. Σε ποιο βαθμό αισθανθήκατε ότι είστε συγκεντρωμένοι στο παιχνίδι;
Καθόλου 1 2 3 4 5 Πολύ
3. Πόση προσπάθεια καταβάλατε για να παίξετε το παιχνίδι;
Ελάχιστη 1 2 3 4 5 Πολύ
4. Αισθάνεστε ότι προσπαθήσατε όσο περισσότερο μπορούσατε;
Καθόλου 1 2 3 4 5 Πολύ
5. Σε ποιο βαθμό χάσατε την αίσθηση του χρόνου;
Καθόλου 1 2 3 4 5 Πολύ
6. Σε ποιο βαθμό αισθανθήκατε ότι δεν βρίσκεστε στον πραγματικό κόσμο ενώ παίζεται;
Καθόλου 1 2 3 4 5 Πολύ
7. Σε ποιο βαθμό ξεχάσατε τις καθημερινές σας έννοιες και προβλήματα;
Καθόλου 1 2 3 4 5 Πολύ
8. Σε ποιο βαθμό είχατε συναίσθηση του περιβάλλοντος στο οποίο βρίσκεστε;
Καθόλου 1 2 3 4 5 Πολύ
9. Σε ποιο βαθμό παρατηρήσατε γεγονότα τα οποία συνέβαιναν δίπλα σας(ενώ εσείς παίζατε);
Καθόλου 1 2 3 4 5 Πολύ

10. Αισθανθήκατε την ανάγκη οποιαδήποτε στιγμή να σταματήσετε να παίζετε και να δείτε τι συμβαίνει τριγύρω σας;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

11. Σε ποιο βαθμό αισθανθήκατε ότι αλληλεπιδράτε με το εικονικό περιβάλλον του παιχνιδιού;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

12. Σε ποιο βαθμό αισθανθήκατε ότι διαχωριστήκατε από το πραγματικό περιβάλλον;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

13. Σε ποιο βαθμό αισθανθήκατε ότι το παιχνίδι ήταν κάτι το οποίο βιώνατε, παρά κάτι το οποίο απλά κάνατε;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

14. Σε ποιο βαθμό η αίσθηση ότι βρίσκεστε μέσα στο εικονικό περιβάλλον του παιχνιδιού ήταν μεγαλύτερη από την αίσθηση ότι βρίσκεστε στο πραγματικό κόσμο;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

15. Βρήκατε τον εαυτό σας τόσο εμβυθισμένο κάποια χρονική στιγμή τόσο πολύ, που να μην μπορείτε να χειριστείτε τον χαρακτήρα;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

16. Σε ποιο βαθμό αισθανθήκατε ότι μπορείτε να κινηθείτε στο παιχνίδι σύμφωνα με τη δική σας θέληση;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

17. Σε ποιο βαθμό βρήκατε το παιχνίδι ενδιαφέρον;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

18. Υπήρχαν στιγμές κατά τη διάρκεια του παιχνιδιού που θέλατε να εγκαταλείψετε;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

19. Σε ποιο βαθμό είχατε κίνητρο να συνεχίσετε να παίζεται;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

20. Σε ποιο βαθμό βρήκατε το παιχνίδι εύκολο;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

21. Σε ποιο βαθμό αισθανθήκατε ότι κάνατε πρόοδο φτάνοντας στο τέλος τους παιχνιδιού;

Καθόλου	1	2	3	4	5	Πολύ
---------	---	---	---	---	---	------

22. Πόσο καλά πιστεύετε ότι τα πήγατε στο παιχνίδι;
Καθόλου καλά 1 2 3 4 5 Πολύ καλά
23. Σε ποιο βαθμό πιστεύεται ότι τα συναισθήματά σας επηρεάστηκαν από το παιχνίδι;
Καθόλου 1 2 3 4 5 Πολύ
24. Σε ποιο βαθμό είχατε ενδιαφέρον να δείτε πως θα εξελιχθούν τα γεγονότα του παιχνιδιού;
Καθόλου 1 2 3 4 5 Πολύ
25. Πόσο πολύ θέλατε να τερματίσετε το παιχνίδι;
Καθόλου 1 2 3 4 5 Πολύ
26. Είχατε αγωνία για το αν θα νικήσετε ή θα χάσετε στο παιχνίδι;
Καθόλου 1 2 3 4 5 Πολύ
27. Αισθανθήκατε τον εαυτό σας κάποια στιγμή τόσο εμβυθισμένο ώστε να θέλετε να "επικοινωνήσετε" απευθείας με το παιχνίδι;
Καθόλου 1 2 3 4 5 Πολύ
28. Σε ποιο βαθμό απολαύσατε τα γραφικά και τις εικόνες;
Καθόλου 1 2 3 4 5 Πολύ
29. Πόσο πιστεύεται ότι ευχαριστηθήκατε την εμπειρία προσομοίωσης του σκι;
Καθόλου 1 2 3 4 5 Πολύ
30. Όταν τελείωσε το πείραμα προσομοίωσης, απογοητευτήκατε που σταματήσατε να παίζεται;
Καθόλου 1 2 3 4 5 Πολύ
31. Θα επιθυμούσατε να παίξετε το παιχνίδι ξανά;
Όχι 1 2 3 4 5 Οπωσδήποτε Ναι

SIMULATOR SICKNESS

Οδηγίες: Κυκλώστε σε τι βαθμό το κάθε σύμπτωμα από τα παρακάτω σας έχει επηρεάσει ακριβώς μετά τη διεξαγωγή του πειράματος.

- | | | | | |
|----------------------|---------|---------|--------|------|
| 1. Γενική Δυσφορία | Καθόλου | Ελαφρώς | Μέτρια | Πολύ |
| 2. Κούραση | Καθόλου | Ελαφρώς | Μέτρια | Πολύ |
| 3. Πονοκέφαλος | Καθόλου | Ελαφρώς | Μέτρια | Πολύ |
| 4. Καταπόνηση ματιών | Καθόλου | Ελαφρώς | Μέτρια | Πολύ |

5. Δυσκολία στην εστίαση	Καθόλου	Ελαφρώς	Μέτρια	Πολύ
6. Αυξημένη έκκριση σιέλου	Καθόλου	Ελαφρώς	Μέτρια	Πολύ
7. Ίδρωμα	Καθόλου	Ελαφρώς	Μέτρια	Πολύ
8. Ναυτία	Καθόλου	Ελαφρώς	Μέτρια	Πολύ
9. Δυσκολία Στη Συγκέντρωση	Καθόλου	Ελαφρώς	Μέτρια	Πολύ
10. Πίεση στο Κεφάλι	Καθόλου	Ελαφρώς	Μέτρια	Πολύ
11. Θολή Όραση	Καθόλου	Ελαφρώς	Μέτρια	Πολύ
12. Ζάλη με τα μάτια ανοιχτά	Καθόλου	Ελαφρώς	Μέτρια	Πολύ
13. Ζάλη με τα μάτια κλειστά	Καθόλου	Ελαφρώς	Μέτρια	Πολύ
14. Τιγγος	Καθόλου	Ελαφρώς	Μέτρια	Πολύ
15. Ευαισθησία στο στομάχι	Καθόλου	Ελαφρώς	Μέτρια	Πολύ
16. Ερυγή	Καθόλου	Ελαφρώς	Μέτρια	Πολύ

Σχόλια:

6.3 Στόχος & Αποτελέσματα

Ο στόχος της παραπάνω αξιολόγησης ήταν διπλός. Πρώτον, να εξεταστεί ο βαθμός εμπύθισης στο περιβάλλον εικονικής πραγματικότητας και δεύτερον να μελετηθούν αν και σε τι βαθμό προκλήθηκαν συμπτώματα και δυσφορία στους χρήστες.

Το Immersive Experience Questionnaire (IEQ) αποτελείται από 6 διαφορετικούς τύπους ερωτήσεων, οι οποίοι περιγράφονται από τους δείκτες που φαίνονται παρακάτω. Οι ερωτήσεις βαθμολογήθηκαν από το 1 έως το 5, με το ένα να σηματοδοτεί τον ελάχιστο βαθμό και το 5 το μέγιστο.

- **Attention:** Σχετίζεται με το πόσο συγκεντρωμένος ήταν ο χρήστης (1, 2, 3, 4).
- **Temporal Disassociation:** Έλλειψη της αίσθησης του χρόνου κατά τη διάρκεια της προσομοίωσης (5, 6, 7, 8, 9, 10).
- **Temporal Transportation:** Ο βαθμός στον οποίο ο χρήστης ένιωσε ότι ήταν περισσότερο μέρος του παιχνιδιού παρά του πραγματικού περιβάλλοντος (11, 12, 13, 14, 15, 16).
- **Challenge:** Σχετίζεται με την αγωνία που είχε ο χρήστης κατά τη διάρκεια του παιχνιδιού (flow) (17, 18, 19, 20, 21, 22).
- **Emotional Attachment:** Πρόκληση συναισθημάτων, λόγω της αίσθησης ότι είναι μέρος του εικονικού περιβάλλοντος (23, 24, 25, 26, 27).
- **Enjoyment:** Ο βαθμός στον οποίο ο χρήστης ευχαριστήθηκε (28, 29, 30, 31).

Αφού λοιπόν, υπολογίστηκαν οι 6 παραπάνω δείκτες και το συνολικό Immersion για κάθε χρήστη, στη συνέχεια υπολογίστηκαν οι μέσοι όροι και τα αποτελέσματα που εξήχθησαν φαίνονται παρακάτω:

Αποτελέσματα							
Χρήστες	Mean At- tention	Mean Temporal Disasso- ciation	Mean Temporal Tran- sporta- tion	Mean Challenge	Mean E- motional Attach- ment	Mean En- joyment	Mean Immer- sion
Χρήστης 1	4.25	3	3.5	3.83	4	4.5	3.9
Χρήστης 2	4	2.83	3.5	4.17	4	3.5	3.61
Χρήστης 3	5	3	4.17	3.67	4.2	4.5	4
Χρήστης 4	4	3.83	3.83	3.33	4.8	4.25	3.97
Χρήστης 5	4.25	3.17	4	3.5	4.6	4.75	3.97
Χρήστης 6	3.25	1.83	3.83	2.67	4	4	3.2
Χρήστης 7	4.25	3	3.67	4	3.4	4.25	3.71
Χρήστης 8	3	3.5	4.17	3.83	3.4	4.75	3.77
Χρήστης 9	4.25	3.17	4	4	4.6	5	4.1
Χρήστης 10	3.25	3	4.83	4.17	5	5	4.19
Χρήστης 11	5	2.83	3.83	3.33	5	4.75	4
Χρήστης 12	3.5	2.67	3.83	3.17	5	4.5	4
Χρήστης 13	4.75	3.67	4.17	2.83	4.8	5	4.06
Χρήστης 14	4.5	3.34	3.67	3.17	4	4.5	3.77
Σύνολο	4.01	3.06	3.93	3.55	4.34	4.58	3.91

Αποτελέσματα:

Total Mean Attention= 4.01

Total Mean Temporal Disassociation= 3.06

Total Mean Temporal Transportation= 3.93

Total Mean Challenge= 3.55

Total Mean Emotional Attachment= 4.34

Total Mean Enjoyment= 4.58

Total Mean Immersion= 3.91

Το Simulator Sickness Questionnaire (SSQ) αποτελεί μέχρι σήμερα το πρότυπο για τη μέτρηση των κινδύνων που μπορεί να προκαλέσει μια προσομοίωση σε περιβάλλον εικονικής πραγματικότητας. Το SSQ περιλαμβάνει έναν κατάλογο 16 συμπτωμάτων που μπορεί να προκληθούν ύστερα από έκθεση σε προσομοιωτή. Για κάθε ένα από τα 16 συμπτώματα υπάρχουν τέσσερις

δυνατές επιλογές ανάλογα με το βαθμό που οι χρήστες τα βίωσαν και ύστερα από μερικούς υπολογισμούς παράγεται ένα τελικό σκορ συμπτωμάτων προσομοίωσης καθώς και 3 δείκτες. Οι τρεις δείκτες παρέχουν διαγνωστικές πληροφορίες σχετικά με τη Ναυτία (Nausea), το κινητικό νεύρο οφθαλμού (Oculomotor) και τον αποπροσανατολισμό (Disorientation). Η βαθμολογία κυμαίνεται ως εξής:

- Από 0 έως 235.6 για τη συνολική βαθμολογία ασθένειας Total SSQ.
- Από 0 έως 200.3 για τη ναυτία (Nausea).
- Από 0 έως 159.2 για την παράμετρο (Oculomotor).
- Από 0 έως 292.3 για τον αποπροσανατολισμό (Disorientation) .

Σε κάθε περίπτωση, το μηδέν εκπροσωπεί την απουσία συμπτωμάτων ασθένειας προσομοιωτή ενώ όσο μεγαλύτερο είναι το σκορ τόσο εντονότερα είναι και τα συμπτώματα.

Αφού συμπληρώθηκαν τα ερωτηματολόγια, έγινε υπολογισμός των Total SSQ, Nausea, Oculomotor και Disorientation για κάθε χρήστη με βάση τα βάρη και τις σχέσεις που φαίνονται στη συνέχεια:

Βάρη Συμπτωμάτων			
Συμπτώματα	Nausea	Oculomotor	Disorientation
Γενική Δυσφορία	1	1	00
Κούραση	0	1	0
Πονοκέφαλος	0	1	0
Καταπόνηση ματιών	0	1	0
Δυσκολία στην εστίαση	0	1	1
Αυξημένη έκκριση σιέλου	1	0	0
Τδρωμα	1	0	0
Ναυτία	1	0	1
Δυσκολία στη συγκέντρωση	1	1	0
Πίεση στο κεφάλι	0	0	1
Θολή Όραση	0	1	1
Ζάλη με τα μάτια ανοιχτά	0	0	1
Ζάλη με τα μάτια κλειστά	0	0	1
Τλιγγος	0	0	1
Ευαισθησία στο στομάχι	1	0	0
Ερυγή	1	0	0
Total*	[1]	[2]	[3]

Δείκτες:

Nausea= [1] x 9.54

Oculomotor= [2] x 7.58

Disorientation= [3] x 13.92

Total SSQ= ([1] + [2] + [3])* 3.74

*Total είναι το άθροισμα που προκύπτει προσθέτοντας τα σκορ των συμπτωμάτων

Η ασθένεια της προσομοίωσης είναι γενικά το αποτέλεσμα της ασυμφωνίας μεταξύ της προσομοίωσης της κίνησης μέσα στο εικονικό περιβάλλον και της αίσθησης που απορρέει από το αιθουσαίο σύστημα. Σε πολλές προσομοιώσεις, το οπτικό σύστημα λαμβάνει πληροφορίες που σχετίζονται με κίνηση (ολίσθηση σκιερ), αλλά το αιθουσαίο σύστημα ερμηνεύει μια στατική κατάσταση η οποία δε μπορεί να συγχρονιστεί με την οπτική αντίληψη της κίνησης. Αυτή η διαφορά είναι που προκαλεί τα συμπτώματα του Simulator Sickness σε πολλούς ανθρώπους.

Αφού λοιπόν υπολογίστηκαν οι 4 παραπάνω δείκτες για κάθε χρήστη, στη συνέχεια υπολογίστηκαν οι μέσοι όροι και τα αποτελέσματα που εξήχθησαν φαίνονται παρακάτω:

Αποτελέσματα:

Total Mean Nausea= 25.21

Total Mean Oculomotor= 18.95

Total Mean Disorientation= 31.82

Total Mean SSQ= 27.78

6.4 Συμπεράσματα

Βάσει των παραπάνω αποτελεσμάτων, καθώς και των σχολίων που υπήρξαν στα ερωτηματολόγια των χρηστών που δοκίμασαν το παιχνίδι, προέκυψαν τα παρακάτω συμπεράσματα.

- Immersive Experience Questionnaire: Τα αποτελέσματα που εξήχθησαν από το ερωτηματολόγιο αυτό είναι αρκετά ικανοποιητικά. Οι χρήστες σε γενικές γραμμές έμειναν πολύ ικανοποιημένοι από την εφαρμογή, γεγονός το οποίο επιβεβαιώνεται από τους παραπάνω δείκτες. Η μέση τιμή της εμπύθισης (Immersion) είναι 3.91 με ελάχιστη τιμή την 3.2 και μέγιστη 4.19. Η πλειονότητα των χρηστών ήταν συγκεντρωμένοι κατά τη διάρκεια της προσομοίωσης (4.01) και ευχαριστήθηκαν το παιχνίδι σε μεγάλο βαθμό (4.58). Επίσης, υψηλός ήταν και ο μέσος όρος του Emotional Attachment (4.34). Αυτό μπορεί να αποδωθεί στο γεγονός ότι πρόκληθηκαν συναισθήματα φόβου λόγω της αίσθησης ότι ολισθαίνουν με την επιτάχυνση της βαρύτητας πάνω σε πάγο. Χαρακτηριστικά, ένα ποσοστό των χρηστών που δοκίμασε την εφαρμογή σε όρθια στάση, αισθάνθηκε τόσο εμβυθισμένο,

που ένιωσε την ανάγκη να κρατηθεί από κάποιο αντικείμενο, φοβούμενο ότι θα πέσει. Το παραπάνω έρχεται να επιβεβαιώσει και την τιμή του Temporal Transportation (3.93). Ειδικότερα, στην πίστα με τα άλματα, οι χρήστες ένιωσαν total transportation, με αποτέλεσμα ορισμένοι να μην μπορούν να χειριστούν άνετα τον χαρακτήρα. Αυτό βέβαια, μπορεί να οφείλεται και στο γεγονός ότι το μεγαλύτερο ποσοστό των χρηστών δεν είχε ξαναβιώσει εμπειρία εικονικής πραγματικότητας. Ακόμα, σημαντικό παράγοντα αποτελεί και ο χρόνος έκθεσης στο περιβάλλον εικονικής πραγματικότητας, καθώς χρήστες που δοκίμασαν την εφαρμογή για περισσότερη ώρα, μετά από ένα σημείο ένιωθαν άνετα και το Simulator Sickness μειώθηκε αισθητά.

Χαμηλότερα ποσοστά παρατηρήθηκαν στις μέσες τιμές των Temporal Disassociation (3.06) και Challenge (3.55). Ορισμένοι χρήστες έχασαν την αίσθηση του χρόνου, ενώ άλλοι σε αρκετά μικρότερο βαθμό. Αυτό μπορεί να εξηγηθεί από το γεγονός, ότι το παιχνίδι δεν είχε πολύπλοκη ιστορία, ώστε ο χρήστης να απορροφηθεί και να χάσει την αίσθηση του χρόνου. Έχει παρατηρηθεί ότι σε παιχνίδια αφήγησης (Narratives), όπου ο χρήστης έχει μεγάλη συμμετοχή στην εξέλιξη και στην πλοκή του παιχνιδιού, είναι πιθανότερο να χάσει την αίσθηση του χρόνου. Επίσης, στα παιχνίδια αυτά υπάρχει περισσότερο άγχος και αγωνία από τον παίκτη να λύσει την υπόθεση. Συνεπώς, οι χαμηλότερες τιμές των δύο τελευταίων δεικτών, οφείλονται σε μεγάλο βαθμό στη φύση του παιχνιδιού.

- Simulator Sickness Questionnaire: Από τους 14 συμμετέχοντες όλοι ολοκλήρωσαν την προσομοίωση κανονικά, ενώ μόνο 2 δεν εμφάνισαν κανένα απολύτως σύμπτωμα. Ο λόγος που η πλειονότητα των χρηστών εμφάνισε τουλάχιστον ένα, οφείλεται κυρίως στη φύση της προσομοίωσης, καθώς περιλαμβάνει κινήσεις, απότομες στροφές και άλματα δημιουργώντας μεγάλα conflicts στο αισθητικό σύστημα. Παρόλλα αυτά, συγκριτικά με τα αποτελέσματα προσομοιωτών που υπάρχουν στη βιβλιογραφία, οι δείκτες μας κυμάνθηκαν σε λογικά, ίσως και χαμηλά επίπεδα. Τα συμπτώματα, σε κάποιους χρήστες ήταν λιγότερο και σε κάποιους άλλους περισσότερο έντονα, δίχως όμως να αποτελούν ανασταλτικό παράγοντα για την ολοκλήρωση της προσομοίωσης. Χαρακτηριστικό σύμπτωμα, το οποίο εμφανίστηκε σε μεγάλη μερίδα των χρηστών, ήταν η ναυτία, ενώ ο ίλλιγος και το ιδρώμα σε λίγο χαμηλότερα επίπεδα. Τα συμπτώματα αυτά, οφείλονται κυρίως στη φύση του παιχνιδιού, καθώς περιλαμβάνει κίνηση με μεγάλη ταχύτητα, ενώ πιθανόν να οφείλονται σε μικρότερο βαθμό και στη σχεδίασή του.

Σχόλια Χρηστών: Οι χρήστες έμειναν πολύ εντυπωσιασμένοι από την εφαρμογή. Ενθουσιάστηκαν σε μεγάλο βαθμό από την απλότητα και τη λειτουργία του User Interface καθώς δεν απαιτούσε τη χρήση πληκτρολογίου ή ποντικιού. Ορισμένα σχόλια έγιναν σχε-

τικά με τον τρόπο εμφάνισης των παραθύρων σε περίπτωση που ο παίχτης χάσει, τα οποία και διορθώθηκαν στη συνέχεια. Αξιοσημείωτα ήταν τα σχόλια και ως προς τον βαθμό εμβύθισης. Η πλειονότητα των χρηστών βρήκε τόσο τα γραφικά όσο και τους στέρεο ήχους αρκετά ρεαλιστικά, γεγονός το οποίο τους απορρόφησε μέσα στο εικονικό περιβάλλον του βουνού, χάνοντας σε ένα βαθμό της αίσθηση με το πραγματικό. Επίσης, ανάμεσα στους χρήστες, ήταν άτομα τα οποία ασχολούνται με το άθλημα του σκι, τα οποία έμειναν σε γενικές γραμμές ικανοποιημένα από την προσομοίωση. Τους φάνηκε αρκετά ρεαλιστικός ο τρόπος χειρισμού του σκιέρ με το κεφάλι, καθώς με τον ίδιο περίπου τρόπο γίνεται και στην πραγματικότητα. Άλλοι χρήστες, οι οποίοι δεν είχαν δοκιμάσει ποτέ σκι, σημείωσαν ότι πήραν μια πολύ καλή εικόνα της αίσθησης του αθλήματος.

Σχετικά με το gameplay, οι χρήστες ήταν διχασμένοι. Οι μισοί το βρήκαν αρκετά δύσκολο ενώ οι υπόλοιποι υποστήριζαν πως τα παιχνίδια οφείλουν να είναι δύσκολα στην αρχή, ώστε ο χρήστης να εθιστεί και να συνεχίσει να παίζει. Επιπλέον, 3 χρήστες επισήμαναν ότι θα μπορούσε να προστεθεί μια πίστα ελεύθερης κατάβασης χωρίς κάποιο συγκεκριμένο στόχο, ώστε ο χρήστης να είναι ελεύθερος να περιηγηθεί σε ολόκληρο το βουνό, πράγμα το οποίο δεν υλοποιήθηκε λόγω χρονικών περιορισμών.

Κεφάλαιο 7

Αποτελέσματα & Μελλοντικές Επεκτάσεις

7.1 Εισαγωγή

Στο παρόν και τελευταίο κεφάλαιο γίνεται μια σύντομη αναφορά των όσων έχουν υλοποιηθεί στα πλαίσια της διπλωματικής εργασίας. Περιλαμβάνει αναφορά των βελτιώσεων που έγιναν βάσει των αποτελεσμάτων που εξήχθησαν, ώστε να φανεί ο βαθμός επίτευξης των αρχικών στόχων. Τέλος, αναφέρονται πιθανές μελλοντικές επεκτάσεις που θα μπορούσαν να υλοποιηθούν στη συγκεκριμένη εφαρμογή.

7.2 Στόχος

Στόχος της παρούσας εργασίας, όπως έχει ήδη αναφερθεί, ήταν η σχεδίαση και η υλοποίηση ενός διαδραστικού τρισδιάστατου παιχνιδιού, που θα προσομοιώνει το άθλημα του σκι και στο οποίο θα μπορεί κανείς να χειριστεί τον παίχτη εξόλοκληρου με ένα Head Mounted Display, μέσω της κίνησης του κεφαλιού. Η υλοποίηση της εφαρμογής έγινε με τη χρήση της μηχανής παιχνιδιών Unity 3D. Το Head Mounted Display το οποίο χρησιμοποιήθηκε είναι το Oculus Rift DK2, το οποίο μέσω του SDK ενσωματώθηκε στο παιχνίδι. Ορισμένες τροποποιήσεις σε τρισδιάστατα αντικείμενα και εικόνες που χρησιμοποιήθηκαν ως υφές έγιναν στο 3D Studio Max και στο Adobe Photoshop CS3 αντίστοιχα. Τέλος, στο παιχνίδι ενσωματώθηκαν αυτόνομοι αντίπαλοι, οι οποίοι υλοποιήθηκαν μέσω τεχνητής νοημοσύνης.

7.3 Αποτελέσματα

Από τα αποτελέσματα των χρηστών και των δοκιμών που έγιναν τόσο από εμάς, όσο και από τους χρήστες που δοκίμασαν το παιχνίδι, εντοπίστηκαν ορισμένες βελτιώσεις οι οποίες ήταν εφικτό να γίνουν στα πλαίσια της πτυχιακής εργασίας οι οποίες και παρατίθενται παρακάτω:

- Η διεπαφή χρήστη αρχικά είχε σχεδιαστεί για να λειτουργεί με το ποντίκι, προβάλλοντας μηνύματα στον χρήστη για το τι θέλει να κάνει, καθώς και για το πότε πρέπει να τοποθετήσει ή να αφαιρέσει την κάσκα από το κεφάλι του. Η ανάγκη όμως για εξολοκλήρου χειρισμό του παιχνιδιού με το κεφάλι, μας οδήγησε στη δημιουργία μιας διαφορετικής προσέγγισης της διεπαφής χρήστη. Πιο συγκεκριμένα, τροποποιήσαμε όλα τα μενού, ώστε να είναι εμβυθισμένα μέσα στο τρισδιάστατο χώρο. Ο χρήστης τώρα μπορεί να επιλέξει τι θέλει να κάνει στοχεύοντας με τον στόχο που βρίσκεται στο κέντρο της οθόνης της κάσκας για περίπου δυο δευτερόλεπτα, όπου και φορτώνεται η επιλογή του.
- Κατά τη διάρκεια του παιχνιδιού, υπήρχαν σημεία στη διεπαφή χρήστη, όπου δεν ήταν προφανές για τον χρήστη, το τι πρέπει να κάνει. Για παράδειγμα, όταν έχανε έπρεπε να γυρίσει προς τα πίσω 180 μοίρες για να δει το μενού που εμφανιζόταν και να αποφασίσει τις επόμενες ενέργειές του. Έτσι λοιπόν, για να επιλυθεί αυτή η ασάφεια, προστέθηκαν κατάλληλα μηνύματα που ειδοποιούν τον χρήστη για το τι ακριβώς πρέπει να κάνει.
- Παρατηρήθηκε από τους χρήστες ότι πηγαίνοντας σε συγκεκριμένα σημεία της πίστας, τα οποία ήταν σχεδόν επίπεδα, ο σκιέρ έχανε πολύ ταχύτητα με αποτέλεσμα να ολισθαίνει πολύ αργά. Για την επίλυση αυτού του προβλήματος, αρχικά εντοπίστηκαν όσο το δυνατόν περισσότερα τέτοια σημεία στο terrain και τροποποιήθηκαν, ώστε να έχουν μεγαλύτερη κλίση. Επιπλέον, τροποποιήσαμε άλλη μια παράμετρο, που είναι η αντίσταση του αέρα. Μειώνοντάς την, ο σκιέρ είναι εφικτό να αναπτύξει μεγαλύτερη ταχύτητα.
- Τέλος, όπως και στα περισσότερα βιντεοπαιχνίδια, ύστερα από πολλές δοκιμές τόσο από εμάς, όσο και από τους χρήστες, εντοπίστηκαν bugs τα οποία οφείλονταν σε μικρολάθη στον κώδικα και τα οποία διορθώθηκαν.

7.4 Μελλοντικές Επεκτάσεις & Βελτιώσεις

Ολοκληρώνοντας το παιχνίδι και παίρνοντας ανάδραση από τους χρήστες, διαπιστώσαμε πως είναι εφικτό να γίνουν πολλές διαφορετικές τροποποιήσεις και επεκτάσεις μελλοντικά. Παρακάτω παρατίθενται για περαιτέρω μελέτη.

- Εισαγωγή πραγματικών διαδρομών heightmaps για ρεαλιστικές καταβάσεις πάνω σε πραγ-

ματικά βουνά. Το παιχνίδι αυτό προσομοίωσης, θα μπορούσε να χρησιμοποιηθεί από τα χιονοδρομικά κέντρα, δίνοντας τη δυνατότητα στους χιονοδρόμους, μέσω της εικονικής πραγματικότητας, να δοκιμάσουν καινούριες πίστες και κλίσεις που πιθανόν να μην έχουν τις ικανότητες και το επίπεδο να τις κατέβουν.

- Πολυπλότερο σύστημα Head Tracking, κάνοντας εκμετάλλευση και του επιταχυνσιόμετρου του Oculus Rift. Θα μπορούσαμε, δηλαδή, να πραγματοποιούμε στον αέρα διαφορετικές κινήσεις και φιγούρες του σκιέρ, ανάλογα με τις μοίρες περιστροφής του κεφαλιού, καθώς και την επιτάχυνση της περιστροφής αυτής. Έτσι λοιπόν, θα μπορούσαν να δημιουργηθούν καινούριες πίστες, οι οποίες θα βασίζονται στη μέγιστη συγκομιδή πόντων, ανάλογα με την πολυπλοκότητα της κίνησης του σκιέρ.
- Ανάπτυξη πολυπλοκότερης τεχνητής νοημοσύνης για τους αντίπαλους σκιέρ και λύκους. Σχετικά με τους αντίπαλους σκιέρ θα μπορούσε να προστεθεί περισσότερη τεχνητή νοημοσύνη, επιτρέποντάς τους να εκτελούν κόλπα στον αέρα και να ανταγωνίζονται τον σκιερ μας. Αυτή η βελτίωση ταιριάζει σε συνδυασμό με την κριβώς προηγούμενη. Σχετικά με τους λύκους, θα μπορούσε να προστεθεί τεχνητή νοημοσύνη, ώστε να προβλέπουν κατά μια έννοια την τροχιά που θα ακολουθήσει ο σκιέρ και να κατευθύνονται προς τα εκεί, ώστε η επίθεση να γίνεται με τον βέλτιστο τρόπο.
- Περαιτέρω βελτίωση του συστήματος φυσικής του σκιέρ, προσδίδοντάς του ακόμα ρεαλιστικότερη κίνηση.
- Προσθήκη ηχητικών εφέ, ανάλογα με την επιφάνεια πάνω στην οποία ολισθαίνει ο σκιερ για ρεαλιστικότερη εμπειρία.
- Εισαγωγή δυνατότητας παιχνιδιού για πολλαπλούς χαρακτήρες. Χρησιμοποιώντας την υπομονάδα δικτύου της Unity 3D να μπορούν να συνδεθούν περισσότεροι του ενός χρήστη που παίζει (2 Oculus Rift), παρέχοντας τη δυνατότητα πραγματοποίησης πιο διαδραστικών αγώνων σκι και αποστολών.
- Η παρούσα πτυχιακή εργασία θα μπορούσε να χρησιμοποιηθεί για περαιτέρω έρευνα πάνω σε θέματα Immersion και Presence. Θα μπορούσε πιθανόν να χρησιμοποιηθεί σε συνδυασμό με μια συσκευή καταγραφής ηλεκτροεγκεφαλογραφήματος (Electroencephalography (EEG)) για την καταγραφή της ηλεκτρικής δραστηριότητας του εγκεφάλου κατά τη διάρκεια της έκθεσης σε περιβάλλον εικονικής πραγματικότητας. Το πεδίο αυτό βρίσκεται ακόμα σε αρχικά ερευνητικά στάδια.

7.5 Επίλογος

Η επιθυμία για ενασχόληση με τεχνολογίες που θα διαδραματίσουν καθοριστικό ρόλο τα επόμενα χρόνια, με οδήγησαν στην επιλογή της υλοποίησης του συγκεκριμένου παιχνιδιού. Μέσω λοιπόν της πτυχιακής αυτής εργασίας, μου δόθηκε η ευκαιρία να μάθω να σχεδιάζω από το μηδέν ένα τρισδιάστατο παιχνίδι για υπολογιστές, χρησιμοποιώντας τεχνολογίες αιχμής, οι οποίες θα παίξουν καθοριστικό ρόλο τα επόμενα χρόνια. Χρειάστηκε να γίνει έρευνα σε διάφορους τομείς της επιστήμης των γραφικών και να αποκτηθούν γνώσεις χειρισμού πολλών επαγγελματικών προγραμμάτων, που χρησιμοποιούνται για την παραγωγή παιχνιδιών και ταινιών, όπως της Unity 3D, του 3D Studio Max και του Adobe Photoshop, καθώς επίσης και του πιο επιτυχημένου Head Mounted Display αυτή τη στιγμή, του Oculus Rift DK2. Επιπλέον, θέματα φυσικής, τεχνητής νοημοσύνης και εύρεσης καλύτερου μονοπατιού είναι τα βασικά θέματα που πραγματεύτηκε η πτυχιακή.

Η διαδικασία και η υλοποίηση του παιχνιδιού ήταν αρκετά δύσκολη και χρονοβόρα. Παρόλα αυτά, η ευχαρίστηση που προσέφερε το παιχνίδι στους χρήστες, καθώς και η ανάδραση που έλαβα από το τελικό αποτέλεσμα, είναι αδιαμφισβήτητα ένα κέρδος. Έτσι λοιπόν, η δημιουργία εικονικών περιβάλλοντων για τρισδιάστατα βιντεοπαιχνίδια, όντας ένας τομέας που απαιτεί από τη φύση του δημιουργικότητα και engineering, μου έδωσε το κίνητρο να συνεχίσω με μεταπτυχιακές σπουδές πάνω στο συγκεκριμένο αντικείμενο μετά την ολοκλήρωση των σπουδών μου από το Πολυτεχνείο Κρήτης.

Βιβλιογραφία

- [1] Qi, Wen. ,*(Serious) Games Development: The State of the Art*, Heerlen The Netherlands: Open Universiteit in the Netherlands, 2014
- [2] Winter, Dave, "Magnavox Odyssey" PONG-story, 2014, <http://www.pong-story.com/odyssey.htm>
- [3] Pierre Delforge, Noah Horowitz, *The Latest-Generation Video Game Consoles*, 2014
- [4] CNET, *Playing games on the PC is making a comeback*, 2015, <http://www.cnet.com/news/playing-games-on-the-pc-is-making-a-comeback/>
- [5] Frederick Li, "Computer Games", *Encyclopedia of Computer Science Engineering*, 2009
- [6] Apperley, T.H., *Genre and Game Studies: Toward a Critical Approach to Video Game Genres*, Simulation and Gaming, 2006
- [7] JH Lee, N Karlova, RI Clarke, K Thornton, A Perti, *Facet analysis of video game genres*, iConference 2014 Proceedings, 2014
- [8] EEDAR, *Game Genre Definitions*, <http://www.eedar.com/Uploads/Genre%20Definitions.pdf>
- [9] Egenfeldt-Nielsen, S. i, *Mapping online gaming: Genres, characteristics and revenue models*, Game-Research, 2003
- [10] Prof. Jason Fritts, *Computer Video Game Genres*, Saint Louis University
- [11] Chris Crawford, *The Art of Computer Game Design*, 1984
- [12] Ari Feldman, *Designing Arcade Computer Game Graphics*, 2000
- [13] Daniel G. Aliaga, Elisha Sacks, *Computer Graphics Pipeline in a Nutshell*, *Purdue University*, <https://www.cs.purdue.edu/homes/aliaga/cs334-08spring/lec-pipeline.pdf>

- [14] Stephan Hodes, *Concurrency in Modern 3D Graphics APIs*, <http://developer.amd.com/community/blog/2015/06/05/concurrency-in-modern-3d-graphics-apis/>
- [15] Promit Roy, *Direct3D vs. OpenGL: Which API to Use When, Where, and Why*, http://www.gamedev.net/page/resources/_/technical/graphics-programming-and-theory/direct3d-vs-opengl-which-api-to-use-when-whe-r1775
- [16] Worcester Polytechnic Institute, *Human Computer Interaction*, <http://web.cs.wpi.edu/~imgd4000/d07/slides/HCI.pdf>
- [17] Jeff Dyck, David Pinelle, Barry Brown, Carl Gutwin, *Learning from Games: HCI Design Innovations in Entertainment Software*, Graphics Interface, 2003
- [18] Paul Milgram, Haruo Takemura, Akira Utsumi, Fumio Kishino, *Augmented reality: A class of displays on the reality-virtuality continuum*, Telemanipulator and Telepresence Technologies (1994) – SPIE Vol. 2351, 1994
- [19] Dimitris Nikolaidis, *Augmented Reality*, Periscopio tis Epistimis, 2003
- [20] *What is Virtual Reality?*, <http://www.vrs.org.uk/virtual-reality/what-is-virtual-reality.html>
- [21] Sonia Schechter, *10 top Augmented Reality Virtual Reality head-up display glasses that are actually useful*, <http://www.marxentlabs.com/ar-trends-top-augmented-reality-virtual-reality-head-up-display-glasses2014/>
- [22] Paul Lamkin, *The best VR headsets*, <http://www.wareable.com/headgear/the-best-ar-and-vr-headsets>
- [23] Human Interface Technology Laboratory, Washington, D.C, *Tracking Interfaces*, <ftp://www.hitl.washington.edu/pub/scivw/publications/IDA-pdf/TRACK.PDF>
- [24] Unity, *Manual*, <http://docs.unity3d.com/Manual/index.html>
- [25] Will Goldstone, *Unity Game Development Essentials*, 2000
- [26] Unity, *Transform*, <http://docs.unity3d.com/ScriptReference/Transform.html>
- [27] Unity, *Lighting*, <http://docs.unity3d.com/Manual/Lighting.html>
- [28] Unity, *Audio Source*, <http://docs.unity3d.com/Manual/class-AudioSource.html>
- [29] Unity, *Audio Listener*, <http://docs.unity3d.com/Manual/class-AudioListener.html>

- [30] Unity, *User Interface*, <https://unity3d.com/learn/tutorials/topics/user-interface-uil>
- [31] Unity, *Navigation and Pathfinding*, <http://docs.unity3d.com/Manual/Navigation.html>
- [32] Oculus, *Introduction to Best Practices*, https://developer.oculus.com/documentation/intro-vr/latest/concepts/bp_intro/
- [33] Oculus, *Binocular Vision, Stereoscopic Imaging and Depth Cues*, https://developer.oculus.com/documentation/intro-vr/latest/concepts/bp_app_imaging/
- [34] Simon Reeve Jason Flock, *Basic Principles of Stereoscopic 3D*, http://www.sky.com/shop/_PDF/3D/Basic_Principles_of_Stereoscopic_3D_v1.pdf
- [35] Oculus, *Field of View and Scale*, https://developer.oculus.com/documentation/intro-vr/latest/concepts/bp_app_fov_scale/
- [36] Oculus, *Rendering Techniques*, https://developer.oculus.com/documentation/intro-vr/latest/concepts/bp_app_rendering/
- [37] Oculus, *Motion*, https://developer.oculus.com/documentation/intro-vr/latest/concepts/bp_app_motion/
- [38] Oculus, *Tracking*, https://developer.oculus.com/documentation/intro-vr/latest/concepts/bp_app_tracking/
- [39] Oculus, *Simulator Sickness*, https://developer.oculus.com/documentation/intro-vr/latest/concepts/bp_app_simulator_sickness/
- [40] C Jennett, AL Cox, P Cairns, S Dhoparee, A Epps, T Tijs, A Walton, *Quantifying the experience of immersion in games*, International journal of human-computer studies 66 (9), 641-661, 2008
- [41] Kennedy, Lane, Berbaum, Lilienthal, *Simulator Sickness Questionnaire*, Graphics Interface, 1993