



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΕΠΙΚΟΙΝΩΝΙΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πτυχιακή Εργασία

Τίτλος Πτυχιακής Εργασίας	(Ελληνικά) Dungeon Escape: Ανάπτυξη Βιντεοπαιχνιδιού Τρίτου Προσώπου με Unity 3D (Αγγλικά) Dungeon Escape: Development of a Third- Person Video Game Using Unity 3D
Ονοματεπώνυμο Φοιτητή	ΓΚΙΑΤΑ ΒΑΣΙΛΗ
Πατρώνυμο	ΠΑΣΚΑΛ
Αριθμός Μητρώου	Π19036
Επιβλέπων	Ευθύμιος Αλέπης, Καθηγητής

Ημερομηνία Παράδοσης Φεβρουάριος 2025

Copyright ©

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν αποκλειστικά τον συγγραφέα και δεν αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Πειραιώς.

Ως συγγραφέας της παρούσας εργασίας δηλώνω πως η παρούσα εργασία δεν αποτελεί προϊόν λογοκλοπής και δεν περιέχει υλικό από μη αναφερόμενες πηγές.

Ευχαριστίες

Θα ήθελα να εκφράσω την ειλικρινή μου ευγνωμοσύνη στον αείμνηστο καθηγητή μου, Θεμιστοκλή Παναγιωτόπουλο, για την καθοδήγηση, την υποστήριξη και τη γνώση που μου προσέφερε. Η συμβολή του στην ανάπτυξη αυτής της πτυχιακής εργασίας υπήρξε ανεκτίμητη. Αφιερώνω αυτή την εργασία στη μνήμη του, με βαθιά εκτίμηση και σεβασμό.

Επίσης, θα ήθελα να ευχαριστήσω όλους τους καθηγητές του πανεπιστημίου για τις γνώσεις που μου παρείχαν καθ' όλη τη διάρκεια των σπουδών μου.

Περίληψη

Στην εργασία αυτή θα δούμε την ανάπτυξη ενός παιχνιδιού ρολών (RPG) με κάμερα τρίτου προσώπου μέσω της μηχανής παιχνιδιών Unity. Το παιχνίδι επικεντρώνεται στην προσπάθεια ενός ήρωα να αποδράσει από ένα μπουντρούμι, αντιμετωπίζοντας εχθρούς με τη χρήση μαγικών ικανοτήτων και αναβαθμίζοντας τις ικανότητές του καθώς προχωρά στα επίπεδα. Οι κύριοι μηχανισμοί του παιχνιδιού περιλαμβάνουν στρατηγικές μάχες, σύστημα προόδου με ξόρκια και δυνατότητα αναβαθμίσεων μέσω εμπόρου που εμφανίζεται κάθε πέντε επίπεδα. Στο τελικό επίπεδο, ο παίκτης αντιμετωπίζει δύο μεγάλα bosses για να ολοκληρώσει το παιχνίδι. Τα αποτελέσματα της εργασίας περιλαμβάνουν τη δημιουργία ενός λειτουργικού πρωτοτύπου που παρέχει μια ολοκληρωμένη εμπειρία RPG.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Παιχνίδι ρολών, Unity, Μηχανισμός μάχης, Μηχανισμός αναβαθμίσεων

Abstract

In this thesis, we will examine the development of a role-playing game (RPG) with a third-person camera using the Unity game engine. The game focuses on a hero's attempt to escape a dungeon, battling enemies using magical abilities and upgrading his skills as he progresses through the levels. The main mechanics of the game include strategic combat, a progression system with spells, and the option to make upgrades through a merchant who appears every five levels. In the final level, the player faces two large bosses to complete the game. The results include the creation of a functional prototype that delivers a complete RPG experience.

KEY WORDS: Role-playing game, Unity, Combat System, Upgrading System

Πίνακας περιεχομένων

Copyright ©.....	i
Ευχαριστίες.....	i
Περίληψη	ii
Abstract	ii
Πίνακας περιεχομένων.....	iii
Κατάλογος εικόνων	vi
1 Εισαγωγή.....	1
2 Εργαλεία και τεχνολογίες ανάπτυξης της εργασίας.....	2
2.1 Unity.....	2
2.2 Microsoft Visual Studio	3
2.3 Mixamo.....	3
2.4 FL STUDIO 20.....	4
3 Ανάλυση Υλοποίησης.....	5
3.1 Ανάπτυξη χειρισμού παίχτη	5
3.2 Σχεδίαση κόσμου.....	6
3.3 Βρόχος gameplay	7
3.4 Μουσική Σχεδίαση του Παιχνιδιού	8
4 Μηχανισμοί Παιχνιδιού και Εκτέλεση Προγράμματος	9
4.1 Main Menu	9
4.1.1 Κυρίο Μενού.....	9
4.1.2 Settings Screen	10
4.1.3 Έναρξη παιχνιδιού.....	10
4.1.4 Νέο παιχνίδι.....	11
4.2 Στοιχεία UI (User Interface) και Μηχανισμοί Παιχνιδιού	12
4.2.1 Μπάρα Υγείας, Άμυνας και Μάνα.....	12
4.2.2 Πόντοι ζημίας.....	14
4.2.3 Φίλτρα ζωής και μανά	14
4.2.4 Μπάρα επιπέδου	15
4.2.5 Γραφικό Περιβάλλον Ξορκιών (κάτω μέρος της οθόνης)	15
4.3 Έμποροι (NPCs)	17
4.3.1 Κάρτες αναβαθμίσεων	18

4.3.2	Βελτιώσεις Βασικών Στατιστικών	19
4.3.3	Αναβαθμίσεις Πόρων	20
4.3.4	Καταναλώσιμα Αντικείμενα	22
4.4	Ξόρκια παίχτη.....	23
4.4.1	Ξόρκι πύρινη σφαίρας(Fireball)	23
4.4.2	Ξόρκι Μετεωριτών (AOE Comet Spell)	24
4.4.3	Ξόρκι Θεραπείας	26
4.4.4	Ξόρκι Παγωμένων Κομητών	27
4.4.5	Ξόρκι Μαγικής Ασπίδας (Shield Spell)	29
4.4.6	Ηλεκτρική Σφαίρα (Thunderball)	30
4.4.7	Ξόρκι Μαγικού Κύκλου (Magic Circle Spell).....	31
4.4.8	Ξόρκι Φυσαλίδας Παράλυσης (Stun Bubble Spell).....	33
4.5	Εχθροί	35
4.5.1	Εχθροί μάχης σώμα με σώμα	35
4.5.2	Εχθροί μάχης από απόσταση	36
4.5.3	Final bosses	37
4.6	Μερικά στιγμιότυπα του παιχνιδιού	38
4.7	Θάνατος παίχτη	42
4.8	Τέλος παιχνιδιού	43
4.9	Μενού παύσης.....	43
5	Scripts	44
5.1	Player System-Σύστημα παίχτη	45
5.1.1	PlayerStats.cs.....	45
5.1.2	PlayerStatsManager.cs.....	45
5.1.3	PlayerStatsUI.cs	46
5.1.4	PlayerInteract.cs	47
5.2	Ικανότητες παίχτη	47
5.2.1	Spells.cs (Διαχείριση λίστας ξορκιών)	47
5.2.2	Projectile.cs	48
5.2.3	AOESpell.cs (Ξόρκι Περιοχής με Επίδραση στους Εχθρούς).....	48
5.2.4	BubbleSpell.cs (Ξόρκι Παγίδευσης Εχθρού)	48
5.2.5	MagicCircle.cs	49
5.2.6	Shield.cs (Ξόρκι Ασπίδας)	49
5.2.7	Potion.cs (Διαχείριση Φίλτρων Ζωής και Μάνα).....	49

5.2.8	PlayerEffects.cs (Ξόρκι Θεραπείας – Heal Spell)	50
5.2.9	SpellCoooldown.cs (Διαχείριση Χρόνου Επαναφόρτισης Ξορκιών)	50
5.3	Κώδικας Εχθρών	50
5.3.1	EnemyAI.cs	50
5.3.2	EnemyProjectile.cs	51
5.3.3	MeleeAttack.cs	51
5.3.4	EnemyStats.cs.....	51
5.3.5	EnemyStatsUI.cs	52
5.3.6	EnemyStatsManager.cs.....	52
5.4	Συστήματα Νίκης, Ήττας και Επιλογών Παιχνιδιού	52
5.4.1	Victory.cs	52
5.4.2	GameOver.cs.....	53
5.4.3	Difficulty.cs (Διαχείριση Επιπέδου Δυσκολίας)	53
5.4.4	HelpScreen.cs	53
5.4.5	Load.cs.....	54
5.4.6	Options.cs	54
5.5	Έμποροι και αναβαθμίσεις.....	55
5.5.1	MerchantInteractable.cs	55
5.5.2	ItemStats.cs.....	55
5.5.3	CloseShop.cs	56
5.6	Σύστημα Στόχευσης, Checkpoints, Ολοκλήρωσης Επιπέδων και Χειρισμού του Παίκτη	56
5.6.1	StarterAssetsInputs.cs.....	56
5.6.2	ThirdPersonAimController.cs	57
5.6.3	Checkpoint.cs.....	58
5.6.4	CompleteLevel.cs	58
5.6.5	Spawn.cs.....	59
5.7	Λοιπά scripts	59
5.7.1	Music.cs	60
5.7.2	Billboard.cs.....	60
6	Συμπεράσματα	61
7	Πίνακας συντμήσεων-αρκτικόλεξων-ακρωνυμίων	62
8	Πίνακας ορολογίας.....	62
9	Βιβλιογραφία.....	63

Κατάλογος εικόνων

Εικόνα 2.1.Λογότυπο της Unity	2
Εικόνα 2.2.Visual Studio	3
Εικόνα 2.3. Στιγμιότυπο από την ιστοσελίδα MIXAMO	3
Εικόνα 2.4.Το project που χρησιμοποιήθηκε για την δημιουργία μουσικής.	4
Εικόνα 3.1 Δωρεάν Asset της Unity για Third Person Controller.....	5
Εικόνα 3.2 Blue Dungeon free asset.....	6
Εικόνα 3.3 Επίπεδα παιχνιδιού	6
Εικόνα 3.4 Unity Editor.....	7
Εικόνα.3.5 Gameplay loop	8
Εικόνα 4.1 Κυρίο Μενού.....	9
Εικόνα 4.2 Παράθυρο ρυθμίσεων	10
Εικόνα 4.3 Επιλογή δυσκολίας	10
Εικόνα 4.4 Εισαγωγή παιχνιδιού	11
Εικόνα 4.5 Πλήκτρα ελέγχου του παιχνιδιού.....	11
Εικόνα 4.6 Γενικό UI παιχνιδιού	12
Εικόνα 4.7 Στατιστικά του ηρώα	12
Εικόνα 4.8 Αναλώσιμα φίλτρα	14
Εικόνα 4.9 UI ξορκιών.....	15
Εικόνα 4.10 Τα διαθέσιμα ξόρκια όταν ο παίχτης είναι επίπεδο 6	15
Εικόνα 4.11 Ανεπαρκές μανά για τα ξόρκια 4 και 5	16
Εικόνα 4.12 Χρόνος επιφόρτισης σε ξόρκια	16
Εικόνα 4.13 Έμπορος με χρυσή κουκούλα.....	17
Εικόνα 4.14 UI διαθέσιμων αναβαθμίσεων.....	17
Εικόνα 4.15 Αύξηση Μέγιστων Πόντων Ζωής	19
Εικόνα 4.16 Αύξηση Μέγιστων Πόντων Άμυνας	19
Εικόνα 4.17 Αύξηση Βασικής Ζημιάς.....	20
Εικόνα 4.18 Αύξηση Μέγιστων Πόντων Μάνα	20
Εικόνα 4.19 Αύξηση Ρυθμού Αναγέννησης Μάνα.....	21
Εικόνα 4.20 Αύξηση Πόντων Εμπειρίας	21
Εικόνα 4.21 Φίλτρο Ζωής.....	22
Εικόνα 4.22 Φίλτρο Μάνα	22

Εικόνα 4.23 Στοχευση ξορκίου(Projectile)	23
Εικόνα 4.24 Εκτέλεση ξορκιού πυρός	24
Εικόνα 4.25 Στοχευση ξορκιού που επηρεάζει πολλούς εχθρούς ταυτόχρονα σε μια περιοχή.....	24
Εικόνα 4.26 Εκτέλεση ξορκιού μετεωριτών	25
Εικόνα 4.27 Εκτέλεση ξορκιού θεραπείας	26
Εικόνα 4.28 Στοχευση ξορκιού παγωμένων κομητών	27
Εικόνα 4.29 Εκτέλεση ξορκιού παγωμένων κομητών	27
Εικόνα 4.30 Οι εχθροί αποκτάνε μια μπλε απόχρωση στα γραφικά τους.....	28
Εικόνα 4.31 Εκτέλεση ξορκιού ασπίδας	29
Εικόνα 4.32 Στοχευση ξορκιού ηλεκτρικής σφαίρας.....	30
Εικόνα 4.33 Εκτέλεση ξορκιού	30
Εικόνα 4.34 Εκτέλεση ξορκιού μαγικού κύκλου	31
Εικόνα 4.35 Επίδειξη ζημίας σε εχθρό	32
Εικόνα 4.36 Εκτέλεση ξορκιού φουσαλίδας.....	33
Εικόνα 4.37 Ο εχθρός παγιδεύεται μέσα στην φουσαλίδα	33
Εικόνα 4.38 Ο εχθρός πέφτει στο πάτωμα αφού τελειώσει το ξόρκι	34
Εικόνα 4.39 Οι πρώτοι εχθροί	35
Εικόνα 4.40 Animation επίθεσης από κοντά.....	35
Εικόνα 4.41 Animation επίθεσης από κοντά με άλλο μοντέλο ιππότη.....	36
Εικόνα 4.42 Επίθεση εχθρού από απόσταση.....	36
Εικόνα 4.43 Αναζήτηση χαρακτήρα	37
Εικόνα 4.44 Τελικός εχθρός παιχνιδιού	37
Εικόνα 4.45 Επίθεση τελικού εχθρού	38
Εικόνα 4.46 Επίπεδο 1.3	38
Εικόνα 4.47 Επίπεδο 2.1	39
Εικόνα 4.48 Στιγμιότυπο που δείχνει πως φαίνονται οι πίστεις από πάνω μέσω τον Unity Editor	39
Εικόνα 4.49 Επίπεδο 5.1	39
Εικόνα 4.50 Επίπεδο 5.2	40
Εικόνα 4.51 Επίπεδο 10.3.....	40
Εικόνα 4.52 Επίπεδο 10.4.....	40
Εικόνα 4.53 Επίπεδο 10.5-Ο τελευταίος έμπορος πριν την τελική μάχη.....	41
Εικόνα 4.54 Τελική μάχη	41

Εικόνα 4.55 Μήνυμα ήττας.....	42
Εικόνα 4.56 Μήνυμα νίκης	43
Εικόνα 4.57 Μενού παύσης.....	44
Εικόνα 5.1 Τα scripts του project	44
Εικόνα 5.2 Unity Inspector του script.....	46
Εικόνα 5.3 Δομή του PlayerStatsUI στο Unity Inspector, με τα UI στοιχεία για τη ζωή, άμυνα, μάνα, εμπειρία, νομίσματα και ζημιά του παίκτη.	46
Εικόνα 5.4 Λίστα ξορκιών	47
Εικόνα 5.5 Το μοντέλο και οι μεταβλητές για το ξόρκι φουσαλίδας.....	48
Εικόνα 5.6 Το μοντέλο και οι μεταβλητές για το ξόρκι μαγικού κύκλου	49
Εικόνα 5.7 Third person Aim Controller	58
Εικόνα 5.8 Εχθροί και οι συντεταγμένες στις οποίες θα εμφανιστούν	59

1 Εισαγωγή

Τα βιντεοπαιχνίδια αποτελούν ένα από τα πιο δημοφιλή και διαρκώς εξελισσόμενα μέσα ψυχαγωγίας, συνδυάζοντας τεχνολογία, αλληλεπίδραση και αφήγηση για τη δημιουργία συναρπαστικών εμπειριών. Η ανάπτυξη ενός παιχνιδιού απαιτεί τη χρήση προηγμένων εργαλείων και τεχνικών, που περιλαμβάνουν προγραμματισμό, σχεδιασμό γραφικών, φυσική προσομοίωση και τεχνητή νοημοσύνη.

Η παρούσα πτυχιακή εργασία επικεντρώνεται στη δημιουργία ενός τρίτου προσώπου (third-person) βιντεοπαιχνιδιού δράσης, όπου ο παίκτης καλείται να εξερευνήσει τον κόσμο του παιχνιδιού, να αντιμετωπίσει εχθρούς, να αποκτήσει νέες ικανότητες και να προοδεύσει μέσω ενός μηχανισμού επιπέδων και αναβαθμίσεων. Το παιχνίδι υλοποιήθηκε με τη μηχανή παιχνιδιών Unity, αξιοποιώντας τη γλώσσα προγραμματισμού C# για την ανάπτυξη των μηχανισμών του.

Βασικός στόχος της εργασίας ήταν η υλοποίηση ενός λειτουργικού πρωτοτύπου, το οποίο να περιλαμβάνει ένα πλήρες gameplay loop, μηχανισμούς μάχης, AI εχθρών, καθώς και ένα σύστημα αναβαθμίσεων και αλληλεπίδρασης με τον κόσμο του παιχνιδιού. Παράλληλα, δόθηκε έμφαση στην ομαλή εμπειρία χειρισμού, στη δυναμική δυσκολία των εχθρών και στη γενικότερη ισορροπία του παιχνιδιού, ώστε να προσφέρει μια ευχάριστη και προκλητική εμπειρία στον παίκτη.

Η εργασία χωρίζεται σε επιμέρους κεφάλαια που περιγράφουν τη διαδικασία ανάπτυξης του παιχνιδιού. Αρχικά, γίνεται μια θεωρητική ανασκόπηση της τεχνολογίας ανάπτυξης βιντεοπαιχνιδιών και των βασικών εργαλείων που χρησιμοποιήθηκαν. Στη συνέχεια, παρουσιάζεται η μεθοδολογία ανάπτυξης, περιγράφοντας την υλοποίηση των βασικών μηχανισμών, τη σχεδίαση των επιπέδων και τη δημιουργία των εχθρών. Τέλος, αναλύονται τα αποτελέσματα και η αξιολόγηση του παιχνιδιού, καθώς και οι δυνατότητες μελλοντικής βελτίωσης και επέκτασής του.

Με αυτή την πτυχιακή εργασία, επιχειρείται η κατανόηση της διαδικασίας ανάπτυξης ενός 3D παιχνιδιού, από τη σύλληψη της αρχικής ιδέας μέχρι την υλοποίηση ενός πλήρως λειτουργικού πρωτοτύπου, καθώς και η ανάδειξη των τεχνικών προκλήσεων και των λύσεων που εφαρμόστηκαν κατά τη διάρκεια της ανάπτυξης.

2 Εργαλεία και τεχνολογίες ανάπτυξης της εργασίας

Ως ο μοναδικός δημιουργός του παιχνιδιού, ανέλαβα εξ ολοκλήρου την ανάπτυξη όλων των βασικών του στοιχείων. Αυτό περιλαμβάνει τη συγγραφή του κώδικα, τον σχεδιασμό και την υλοποίηση της κίνησης των χαρακτήρων, τη δημιουργία ή την επιλογή μουσικής επένδυσης, καθώς και την ανάπτυξη των γραφικών στοιχείων του παιχνιδιού.

Δεδομένου ότι η ανάπτυξη ενός βιντεοπαιχνιδιού απαιτεί ένα ευρύ φάσμα δεξιοτήτων και εργαλείων, στηρίχθηκα σε διάφορα λογισμικά και ιστοσελίδες που με βοήθησαν να επιταχύνω τη διαδικασία και να επιτύχω ένα ολοκληρωμένο αποτέλεσμα. Σε αυτήν την ενότητα, θα παρουσιαστούν τα εργαλεία που χρησιμοποίησα.

2.1 Unity

Η Unity είναι μία από τις κορυφαίες πλατφόρμες ανάπτυξης παιχνιδιών, προσφέροντας υποστήριξη για 2D και 3D γραφικά, scripting σε C#, και εργαλεία φυσικής και animation. Στο "Dungeon Escape", χρησιμοποιήθηκε για τη δημιουργία του πυρήνα του παιχνιδιού, όπως τη διαχείριση των επιπέδων, των εχθρών και των μαχών. Επιπλέον, η Unity παρέχει ενσωματωμένα εργαλεία για την αλληλεπίδραση και τα animations, τα οποία επέτρεψαν την υλοποίηση των μηχανισμών μάχης και των αναβαθμίσεων.

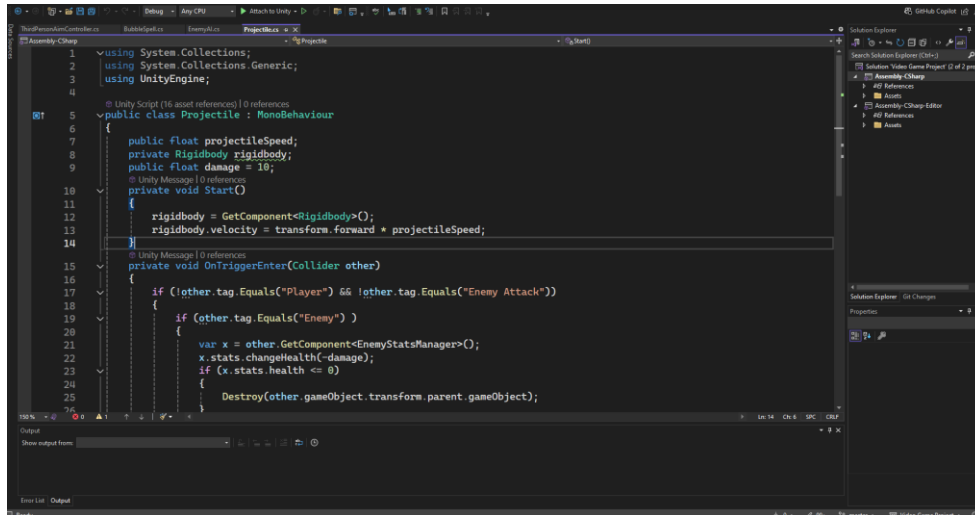


Εικόνα 2.1.Λογότυπο της Unity

Για την ανάπτυξη του παιχνιδιού χρησιμοποιήθηκε η έκδοση 2021.3.21f της Unity, η οποία είναι μια από τις πιο πρόσφατες και σταθεροποιημένες εκδόσεις, εξασφαλίζοντας μια ομαλή και αξιόπιστη διαδικασία ανάπτυξης. Τα δωρεάν assets του Unity Asset Store επίσης επέτρεψαν τη γρήγορη ανάπτυξη του παιχνιδιού, χωρίς την ανάγκη δημιουργίας μοντέλων και animations από το μηδέν, εστιάζοντας έτσι περισσότερο στη βελτίωση της μηχανικής του παιχνιδιού.

2.2 Microsoft Visual Studio

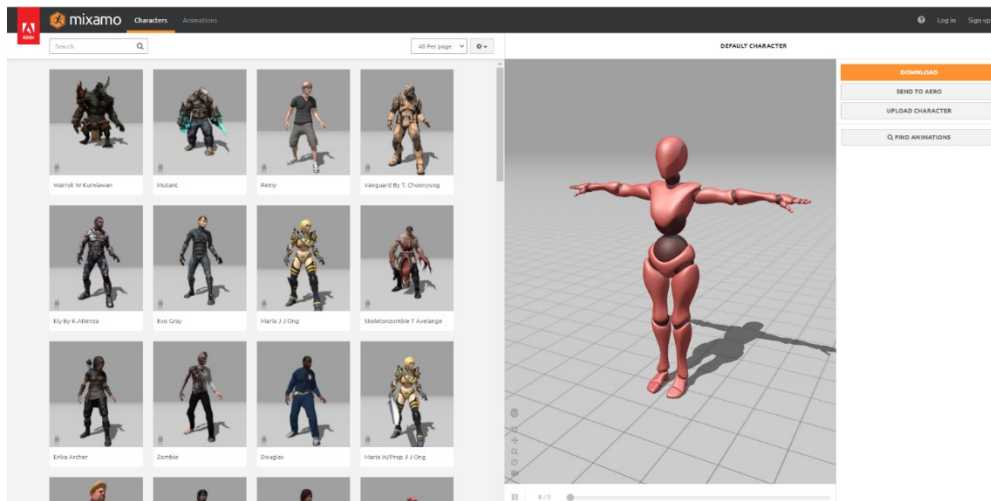
Το Visual Studio αποτελεί ένα από τα πιο ευρέως χρησιμοποιούμενα IDE για προγραμματισμό, και χρησιμοποιήθηκε για τη συγγραφή των scripts σε C#. Μέσω αυτού, αναπτύχθηκαν οι βασικοί μηχανισμοί του παιχνιδιού, όπως η τεχνητή νοημοσύνη των εχθρών, τα ξόρκια του παίκτη, και η διαχείριση της προόδου μέσα στα επίπεδα



Εικόνα 2.2. Visual Studio

2.3 Mixamo

Η Mixamo είναι μια ιστοσελίδα που παρέχει animations για ανθρωποειδή 3D μοντέλα. Χρησιμοποιήθηκε για τη δημιουργία των animations του πρωταγωνιστή και των εχθρικών χαρακτήρων του παιχνιδιού, καλύπτοντας όλες τις επιθέσεις και τις κινήσεις τους στον χώρο.



Εικόνα 2.3. Στιγμιότυπο από την ιστοσελίδα MIXAMO

2.4 FL STUDIO 20

Το FL Studio είναι ένα επαγγελματικό λογισμικό παραγωγής μουσικής (Digital Audio Workstation - DAW), που αναπτύχθηκε από την εταιρεία Image-Line. Αρχικά γνωστό ως "Fruity Loops", είναι δημοφιλές για την ευκολία χρήσης και την ισχυρή σουίτα εργαλείων που προσφέρει για τη δημιουργία, ηχογράφηση, επεξεργασία και μίξη μουσικής.



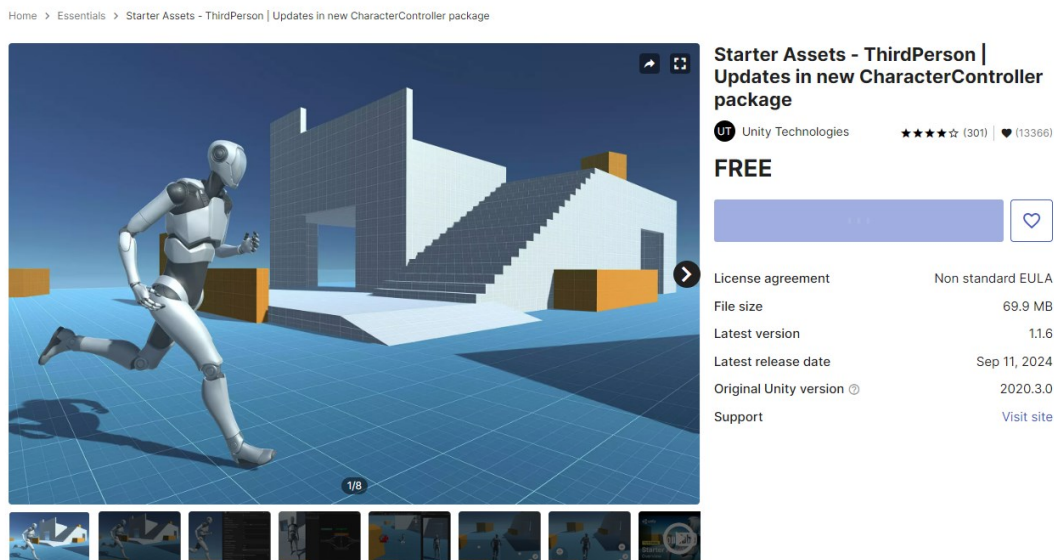
Εικόνα 2.4. Το project που χρησιμοποιήθηκε για την δημιουργία μουσικής.

3 Ανάλυση Υλοποίησης

Η ανάπτυξη του παιχνιδιού βασίστηκε στη χρήση της μηχανής παιχνιδιών Unity και της γλώσσας προγραμματισμού C#. Ο στόχος ήταν να δημιουργηθεί ένα τρίτου προσώπου (third-person) βιντεοπαιχνίδι δράσης με στοιχεία RPG, όπου ο παίκτης μπορεί να εξερευνήσει το περιβάλλον, να αλληλεπιδράσει με NPCs, να συμμετάσχει σε μάχες και να αναβαθμίσει τις ικανότητές του καθώς προοδεύει στο παιχνίδι.

3.1 Ανάπτυξη χειρισμού παίχτη

Ο χειρισμός του παίκτη είναι ένα από τα πιο κρίσιμα στοιχεία σε κάθε βιντεοπαιχνίδι, καθώς επηρεάζει άμεσα την εμπειρία και την αλληλεπίδραση του χρήστη με τον εικονικό κόσμο. Στην παρούσα υλοποίηση, ο χειρισμός βασίζεται σε ένα σύστημα τρίτου προσώπου (third-person), το οποίο επιτρέπει την ελεύθερη κίνηση του χαρακτήρα στον χώρο, τη διαχείριση της κάμερας, καθώς και τη χρήση διαφόρων ενεργειών, όπως το άλμα, η επίθεση και η στόχευση.



Εικόνα 3.1 Δωρεάν Asset της Unity για Third Person Controller

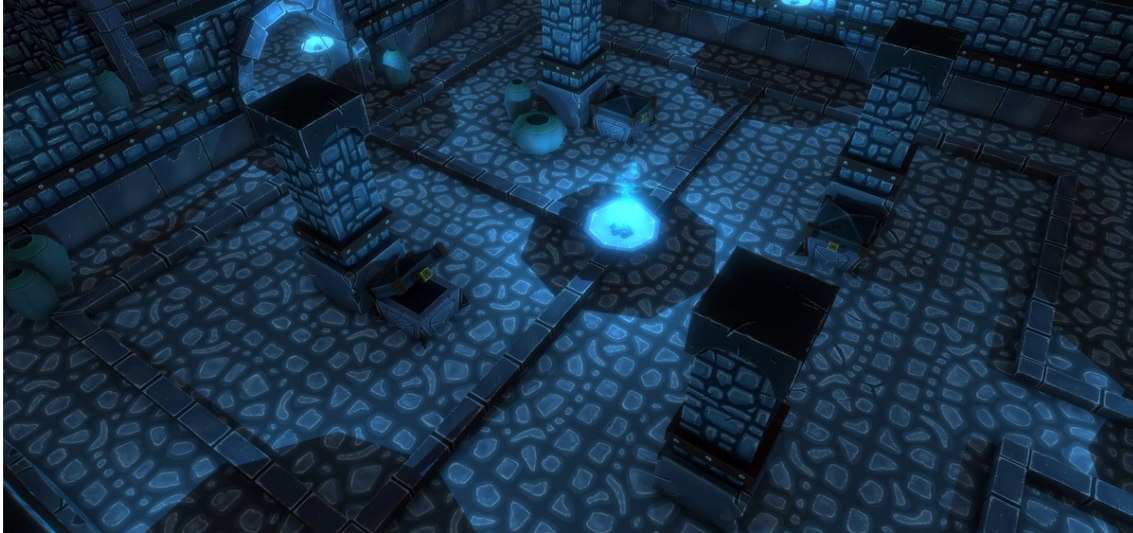
Για την ανάπτυξη του συστήματος χειρισμού, αξιοποιήθηκε το Third Person Controller της Unity, που περιλαμβάνεται στα Starter Assets. Αυτό το asset παρέχει έναν προκαθορισμένο χαρακτήρα με βασικές κινήσεις, όπως:

- Μετακίνηση στον χώρο με τη χρήση των πλήκτρων κατεύθυνσης.
- Περιστροφή και έλεγχος της κάμερας μέσω του ποντικιού.
- Άλμα για να ξεπερνάει εμπόδια ή να μετακινείται σε διαφορετικά επίπεδα.

Το Starter Assets – Third Person Controller της Unity προσφέρει έναν έτοιμο κώδικα για τον χειρισμό του παίκτη, επιτρέποντας την άμεση ενσωμάτωση βασικών κινήσεων χωρίς την ανάγκη ανάπτυξης ενός συστήματος από την αρχή. Αυτό επιταχύνει τη διαδικασία ανάπτυξης, καθώς αυτό με βοήθησε να επικεντρωθώ σε πιο σύνθετα στοιχεία του gameplay, όπως η μάχη, η χρήση ξορκιών και οι μηχανισμοί αλληλεπίδρασης.

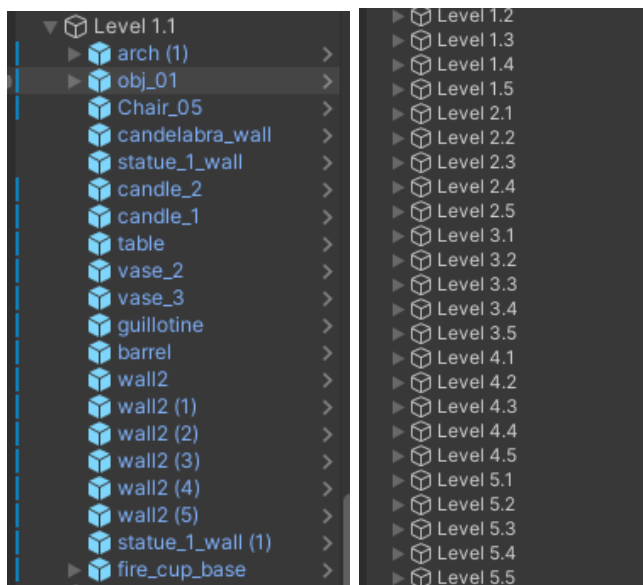
3.2 Σχεδίαση κόσμου

Για τη σχεδίαση του κόσμου και των επιπέδων χρησιμοποιήθηκαν διάφορα δωρεάν assets, τα οποία επαναχρησιμοποιούνται σε πολλαπλά επίπεδα ώστε να διατηρηθεί η συνοχή του περιβάλλοντος και η οπτική ταυτότητα του παιχνιδιού.



Εικόνα 3.2 Blue Dungeon free asset

Τα επίπεδα κατασκευάστηκαν με τη χρήση modular 3D assets, τα οποία μπορούν να προσαρμοστούν και να συνδυαστούν για τη δημιουργία μοναδικών χώρων.



Εικόνα 3.3 Επίπεδα παιχνιδιού

Για τη βέλτιστη οργάνωση και διαχείριση του κόσμου του παιχνιδιού, κάθε επίπεδο έχει ομαδοποιηθεί με βάση τα assets που το αποτελούν, επιτρέποντας μια πιο συστηματική και αποδοτική διάταξη των στοιχείων μέσα στη σκηνή.

Κάθε επίπεδο περιέχει προκαθορισμένα αντικείμενα, όπως τοίχους, έπιπλα, διακοσμητικά στοιχεία και φωτισμό, τα οποία βρίσκονται σε συγκεκριμένες θέσεις για να διαμορφώσουν τον χώρο με τρόπο που να ταιριάζει στο gameplay. Η σωστή κατανομή

και επαναχρησιμοποίηση των assets βοηθά στη δημιουργία ενός συνεκτικού περιβάλλοντος, ενώ παράλληλα μειώνει την κατανάλωση μνήμης του παιχνιδιού.



Εικόνα 3.4 Unity Editor

Σε κάθε δωμάτιο ή περιοχή, υπάρχει ένας Spawner, ένας μηχανισμός που ενεργοποιείται μόλις ο παίκτης περάσει ένα συγκεκριμένο σημείο ή εισέλθει σε έναν νέο χώρο. Μόλις ενεργοποιηθεί:

- Δημιουργεί τους αντίστοιχους εχθρούς, λαμβάνοντας υπόψη το επίπεδο του παίκτη, τη δυσκολία του παιχνιδιού και τη σχεδίαση του χώρου.
- Μπορεί να προσαρμόζει τον αριθμό και τον τύπο των εχθρών, ώστε να διατηρείται η ισορροπία της μάχης.
- Εξασφαλίζει ότι οι εχθροί δεν εμφανίζονται ξανά εφόσον ο παίκτης τους εξοντώσει, αποφεύγοντας έτσι τη μη ρεαλιστική αναπαραγωγή αντιπάλων.

3.3 Βρόχος gameplay

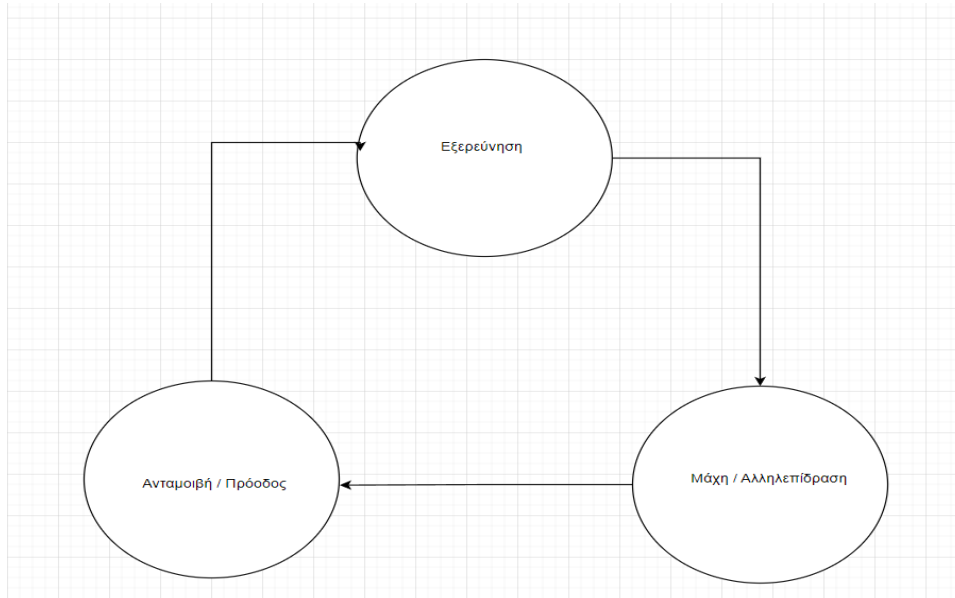
Το παιχνίδι ακολουθεί έναν δομημένο βρόχο παιχνιδιού, ο οποίος καθορίζει τη ροή του gameplay και τον τρόπο με τον οποίο ο παίκτης εξελίσσεται μέσα στον κόσμο του παιχνιδιού. Ο βρόχος βασίζεται σε τρεις κύριες φάσεις:

1. Εξερεύνηση
2. Μάχη / Αλληλεπίδραση
3. Ανταμοιβή / Πρόοδος

Η εξερεύνηση αποτελεί τη βασική δραστηριότητα του παίκτη καθώς περιπλανιέται στο παιχνίδι. Ο παίκτης κινείται στον κόσμο, ανακαλύπτει νέες περιοχές, βρίσκει εχθρούς, NPCs και προετοιμάζεται για τις προκλήσεις που θα αντιμετωπίσει. Κατά τη διάρκεια αυτής της φάσης, μπορεί να εντοπίσει χρήσιμες πηγές, checkpoints ή εμπόρους που προσφέρουν αναβαθμίσεις.

Κατά τη διάρκεια της δράσης, ο παίκτης εμπλέκεται σε μάχες με εχθρούς, χρησιμοποιώντας τα ξόρκια και τις ικανότητές του. Οι εχθροί μπορεί να είναι είτε απλοί εχθροί είτε bosses, με διαφορετικές στρατηγικές και επιθέσεις. Εκτός από τη μάχη, ο παίκτης μπορεί επίσης να αλληλεπιδρά με NPCs για τις αναβαθμίσεις του.

Μετά τη μάχη ή την ολοκλήρωση μιας σημαντικής αλληλεπίδρασης, ο παίκτης λαμβάνει ανταμοιβές, όπως XP, νομίσματα, νέα αντικείμενα ή αναβαθμίσεις. Η εμπειρία που κερδίζει μπορεί να τον βοηθήσει να ξεκλειδώσει νέα ξόρκια, να αυξήσει τη δύναμή του ή να αποκτήσει νέα ικανότητα. Αυτή η φάση προετοιμάζει τον παίκτη για την επόμενη φάση Εξερεύνησης, ξεκινώντας εκ νέου τον κύκλο του gameplay.



Εικόνα.3.5 Gameplay loop

3.4 Μουσική Σχεδίαση του Παιχνιδιού

Η μουσική και ο ηχητικός σχεδιασμός διαδραματίζουν καθοριστικό ρόλο στη δημιουργία ατμόσφαιρας και στην ενίσχυση της εμπειρίας του παίκτη. Στο Dungeon Escape, η μουσική χρησιμοποιείται τόσο για την απόδοση της ατμόσφαιρας του κόσμου όσο και για την ενίσχυση της έντασης κατά τη διάρκεια των μαχών.

Η μουσική υπόκρουση έχει σχεδιαστεί έτσι ώστε να εναλλάσσεται δυναμικά ανάλογα με την κατάσταση του παιχνιδιού:

- Κατά τη διάρκεια της εξερεύνησης, η μουσική είναι ατμοσφαιρική και μυστηριώδης, με χαμηλά έγχορδα και αργά ambient ηχητικά εφέ, ώστε να δημιουργεί ένα αίσθημα απομόνωσης και αγωνίας.
- Όταν ξεκινά μια μάχη, η μουσική αλλάζει αυτόματα σε πιο έντονους ρυθμούς, με τύμπανα, έντονες συγχορδίες και δραματικά μοτίβα, αυξάνοντας την αίσθηση της έντασης και της δράσης.
- Σε στιγμές νίκης, όπως μετά την εξολόθρευση ενός εχθρού ή την ολοκλήρωση ενός επιπέδου, ακούγονται νικητήριες μελωδίες, δίνοντας στον παίκτη ένα αίσθημα ικανοποίησης και επιβράβευσης.
- Σε περίπτωση ήττας, η μουσική γίνεται σκοτεινή και δραματική, με τη χρήση χορωδιακών και πιάνου, ενισχύοντας το συναίσθημα της απώλειας και της αποτυχίας.

4 Μηχανισμοί Παιχνιδιού και Εκτέλεση Προγράμματος

4.1 Main Menu

4.1.1 Κύριο Μενού

Παρακάτω βρίσκεται το Κύριο Μενού, το οποίο είναι το πρώτο πράγμα που εμφανίζεται όταν ανοίγουμε το εκτελέσιμο αρχείο.



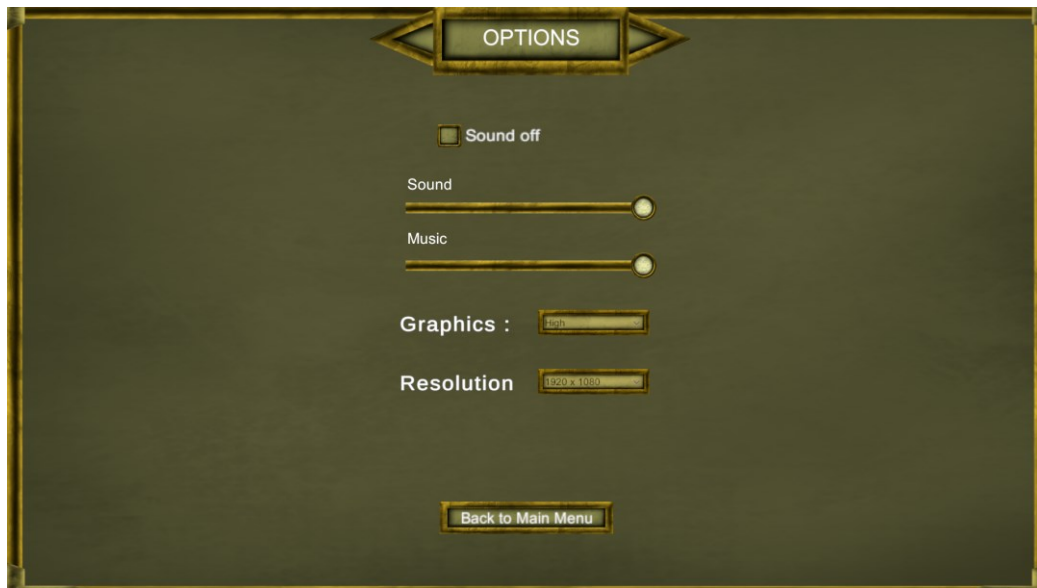
Εικόνα 4.1 Κύριο Μενού

Ο χρήστης έχει τις εξής επιλογές:

- **Start:** Δημιουργεί νέο παιχνίδι από την αρχή και του εμφανίζεται ένα νέο μενού που επιλεγεί την δυσκολία του παιχνιδιού.
- **Continue:** Συνεχίζει από το τελευταίο επίπεδο που έχει ολοκληρώσει ο παίχτης.
- **Settings:** Εμφανίζει τις ρυθμίσεις του παιχνιδιού για τα γραφικά και τον ήχο.
- **Quit:** Τερματίζει το παιχνίδι

4.1.2 Settings Screen

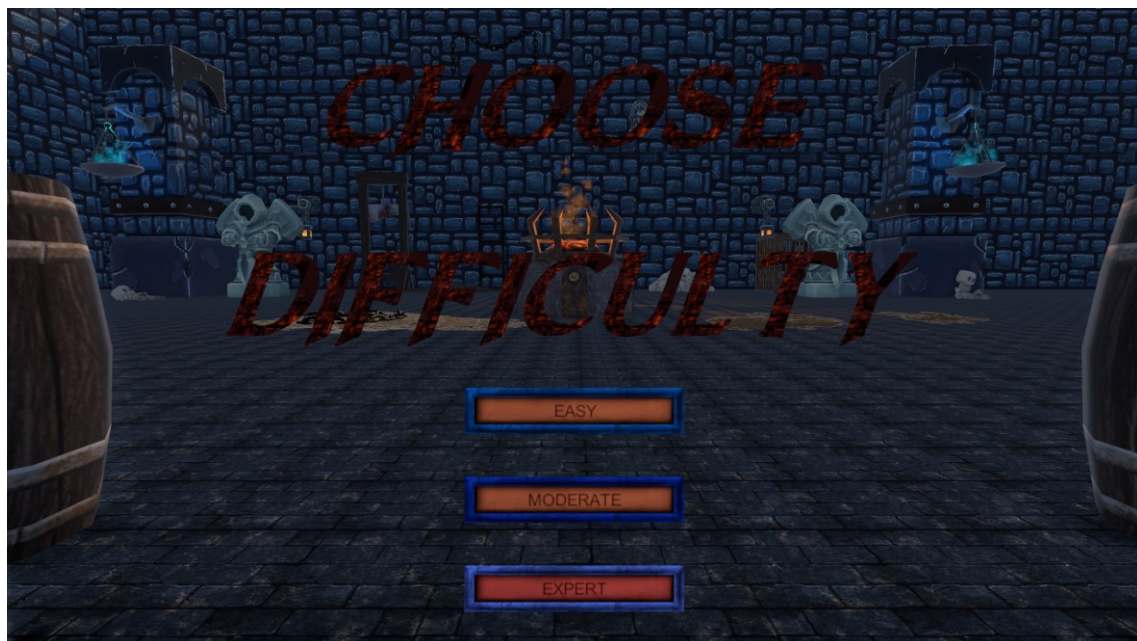
Με το κουμπί “Ρυθμίσεις” ανοίγουμε τις ρυθμίσεις του παιχνιδιού. Από εκεί μπορούμε να προσαρμόσουμε την ποιότητα των γραφικών καθώς και την ένταση του ήχου.



Εικόνα 4.2 Παράθυρο ρυθμίσεων

4.1.3 Έναρξη παιχνιδιού

Αφού πατηθεί το κουμπί Start, ο χρήστης μπορεί να επιλέξει την δυσκολία του παιχνιδιού.

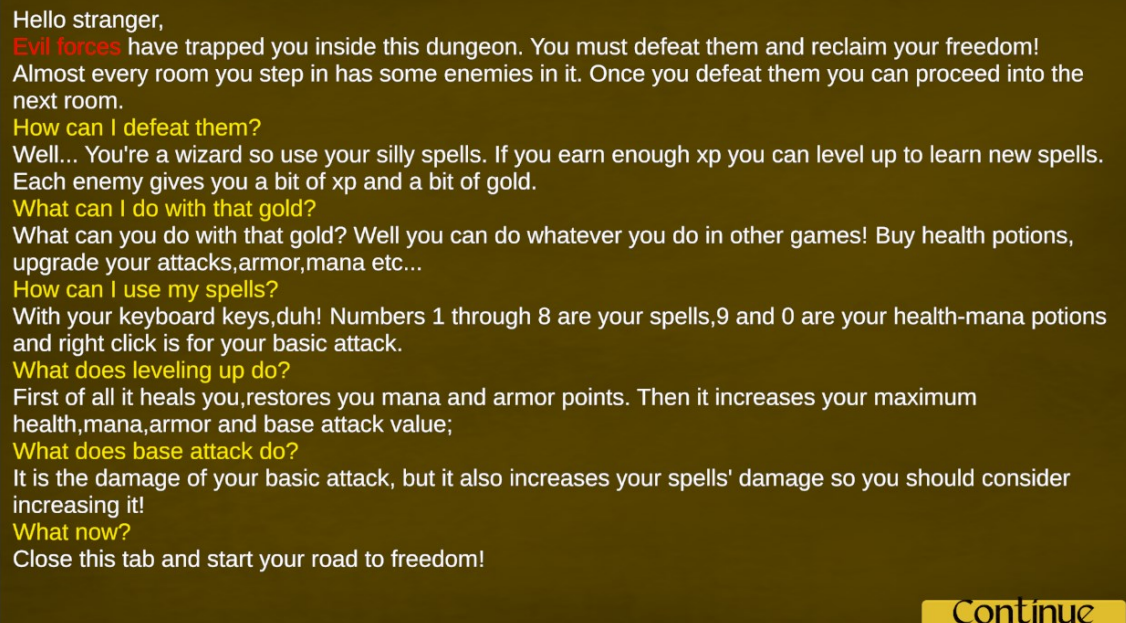


Εικόνα 4.3 Επιλογή δυσκολίας

Η δυσκολία που επιλεγεί ο χρήστης επηρεάζει την ζωή των εχθρών και την ζημιά που προκαλούν οι επιθέσεις τους.

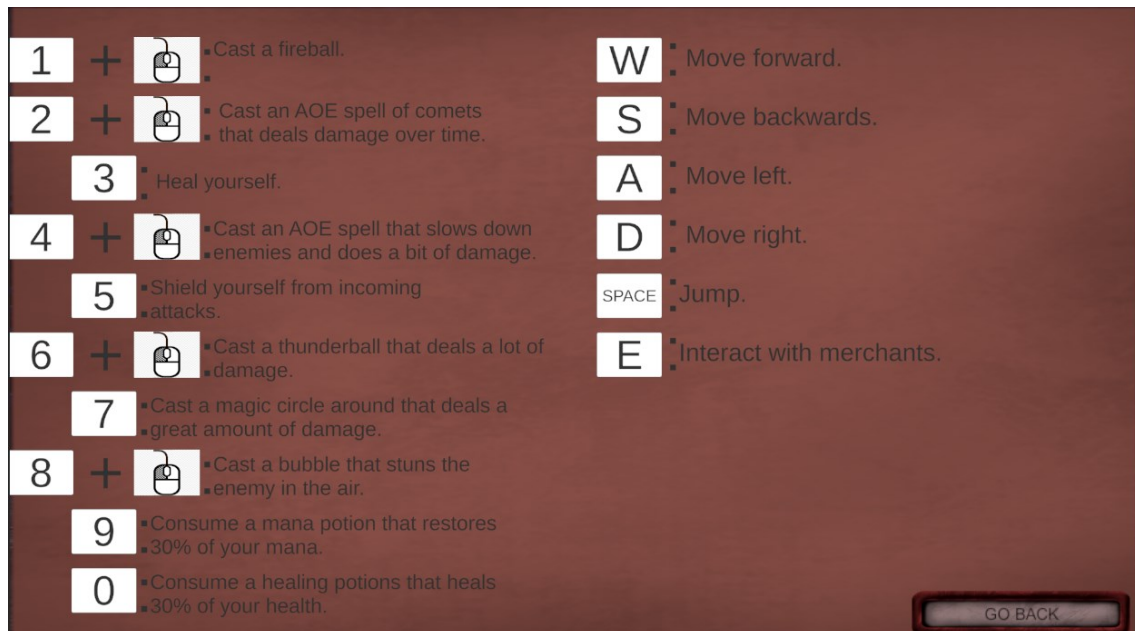
4.1.4 Νέο παιχνίδι

Εφόσον ο παίχτης ξεκινήσει νέο παιχνίδι από την αρχή, εμφανίζεται το παρακάτω παράθυρο που εξηγεί την υπόθεση του παιχνιδιού.



Εικόνα 4.4 Εισαγωγή παιχνιδιού

Έπειτα εμφανίζεται ένα παράθυρο που δείχνει όλα τα πλήκτρα και ενέργειες που μπορεί να κάνει ο χρήστης.



Εικόνα 4.5 Πλήκτρα ελέγχου του παιχνιδιού

4.2 Στοιχεία UI (User Interface) και Μηχανισμοί Παιχνιδιού

Το γραφικό περιβάλλον χρήστη (UI) του παιχνιδιού παρέχει όλες τις απαραίτητες πληροφορίες για την κατάσταση του χαρακτήρα. Από τη στιγμή που ο παίκτης εισέρχεται στο παιχνίδι, αποκτά πλήρη έλεγχο του χαρακτήρα και μπορεί να κινείται ελεύθερα στον χώρο. Όπως συμβαίνει στα περισσότερα παιχνίδια τρίτου προσώπου (third-person games), ο πρωταγωνιστής εμφανίζεται στο κέντρο της οθόνης και ακολουθεί τις κινήσεις του χρήστη.

Ο παίκτης μπορεί να ελέγχει την κάμερα με το ποντίκι, επιτρέποντάς του να περιστρέφεται και να εξερευνά τον χώρο γύρω του. Η δυνατότητα ελεύθερης κάμερας είναι απαραίτητη τόσο για την πλοήγηση στο περιβάλλον όσο και για τη στοχοποίηση των ξορκιών, καθώς τα ξόρκια εκτελούνται προς την κατεύθυνση που κοιτά ο χαρακτήρας.



Εικόνα 4.6 Γενικό UI παιχνιδιού

4.2.1 Μπάρα Υγείας, Άμυνας και Μάνα

Το γραφικό περιβάλλον χρήστη (UI) του παιχνιδιού παρέχει όλες τις απαραίτητες πληροφορίες για την κατάσταση του χαρακτήρα. Παρακάτω αναλύονται τα βασικά στοιχεία που εμφανίζονται στην οθόνη.



Εικόνα 4.7 Στατιστικά του ηρώα

Κόκκινη μπάρα: Η κόκκινη μπάρα αντιπροσωπεύει τη ζωή (HP) του χαρακτήρα. Όταν η ζωή μηδενιστεί, ο χαρακτήρας χάνει το παιχνίδι. Ωστόσο, για να μειωθεί η ζωή του παίκτη, πρέπει πρώτα να έχει μηδενιστεί η άμυνά του.

Η άμυνα λειτουργεί ως πρώτη γραμμή προστασίας, απορροφώντας τη ζημιά από τις επιθέσεις των εχθρών. Εάν ο παίκτης διαθέτει ακόμα άμυνα και δεχτεί ζημιά, τότε οι πόντοι ζημιάς της επίθεσης αφαιρούνται πρώτα από την άμυνα του χαρακτήρα. Μόνο αν η ζημιά είναι μεγαλύτερη από το διαθέσιμο επίπεδο άμυνας, η υπερβάλλουσα ζημιά αφαιρείται από την κόκκινη μπάρα ζωής.

Για παράδειγμα, αν ο παίκτης έχει 20 πόντους άμυνας και δεχτεί 30 πόντους ζημιάς, η άμυνα θα απορροφήσει τους πρώτους 20 πόντους, ενώ οι υπόλοιποι 10 πόντοι ζημιάς θα αφαιρεθούν από τη ζωή του. Αν η άμυνα έχει ήδη μηδενιστεί, κάθε επίθεση θα αφαιρείται απευθείας από τους πόντους ζωής του χαρακτήρα.

Τέλος, εάν ο παίκτης έχει πλήρη άμυνα, η ζωή του αρχίζει να ανακτάται σταδιακά χωρίς να απαιτείται η χρήση φίλτρων ή άλλων αντικειμένων.

Γαλάζια μπάρα: Αντιπροσωπεύει την **άμυνα (Armor)** του χαρακτήρα. Όταν ο χαρακτήρας δέχεται χτυπήματα, η άμυνα απορροφά μέρος της ζημιάς όπως αναφέρθηκε προηγουμένως. Επιπλέον, η άμυνα αναπληρώνεται συνεχώς με αργό ρυθμό εφόσον δεν δεχτεί ο παίκτης ζημιά από τους αντίπαλους.

Μπλε μπάρα: Αντιπροσωπεύει το **μάνα (Mana)**, το οποίο χρησιμοποιείται για την εκτέλεση ξορκιών. Κάθε ξόρκι απαιτεί ένα συγκεκριμένο ποσό μάνα για να εκτελεστεί.

Αν ο παίκτης δεν έχει αρκετό μάνα, δεν μπορεί να χρησιμοποιήσει ξόρκια μέχρι να αναπληρωθεί η μπάρα.

Το μάνα επιτρέπει στον παίκτη να επιλέξει διαφορετικές στρατηγικές μάχης, συνδυάζοντας επιθέσεις σώμα με σώμα με μαγικές ικανότητες.

Παρόμοια με την άμυνα, το μάνα αναπληρώνεται αυτόματα με την πάροδο του χρόνου. Ο ρυθμός με τον οποίο το μάνα επανέρχεται εξαρτάται από τον ρυθμό ανανέωσης του παίκτη (**mana regeneration rate**). Αυτός ο ρυθμός καθορίζει το πόσο γρήγορα η μπάρα μάνα γεμίζει, επιτρέποντας στον παίκτη να χρησιμοποιεί ξόρκια σε τακτά χρονικά διαστήματα χωρίς να βασίζεται αποκλειστικά σε φίλτρα.

Αν και ο παίκτης ξεκινά με έναν βασικό ρυθμό αναπλήρωσης, στη συνέχεια μπορεί να συναντήσει εμπόρους ή NPCs, οι οποίοι προσφέρουν αναβαθμίσεις που αυξάνουν το ποσοστό αναγέννησης του μάνα.

4.2.2 Πόντοι ζημίας

Οι πόντοι ζημίας που εμφανίζονται κάτω από τη μπάρα μάνα αντιπροσωπεύουν τη συνολική επιθετική δύναμη του χαρακτήρα και επηρεάζουν τόσο τις βασικές επιθέσεις όσο και τα ξόρκια του. Ο αριθμός των πόντων ζημίας δείχνει πόση ζημιά προκαλείται σε κάθε επίθεση, προσφέροντας μια ξεκάθαρη ένδειξη της αποτελεσματικότητας του χαρακτήρα στη μάχη.

Όταν ο παίκτης εκτελεί μια βασική επίθεση, η συνολική ζημιά που προκαλείται είναι ίση με την τρέχουσα τιμή των πόντων ζημίας. Για παράδειγμα, αν ο χαρακτήρας διαθέτει 10 πόντους ζημίας, τότε κάθε βασική επίθεση προκαλεί 10 damage στον αντίπαλο. Αυτή η τιμή μπορεί να αυξηθεί μέσω διαφόρων μηχανισμών, όπως αναβαθμίσεις ή buffs που ενισχύουν τη δύναμη του χαρακτήρα.

Εκτός από τις βασικές επιθέσεις, οι πόντοι ζημίας επηρεάζουν και τη μαγική δύναμη του χαρακτήρα, καθώς προσθέτουν επιπλέον ζημιά στα ξόρκια. Αν ένα ξόρκι έχει μια βασική τιμή ζημίας 50, και ο παίκτης έχει 10 πόντους ζημίας, τότε η τελική ζημιά του ξορκιού θα ανέρχεται σε 60 μονάδες. Αυτό το σύστημα επιτρέπει την κλιμακούμενη αύξηση της δύναμης των ξορκιών, καθιστώντας τα πιο ισχυρά όσο βελτιώνεται ο χαρακτήρας.

Οι πόντοι ζημίας μπορούν να αυξηθούν μέσα στο παιχνίδι με διάφορους τρόπους. Ο παίκτης μπορεί να ανεβάσει επίπεδο, αποκτώντας περισσότερη επιθετική δύναμη. Επιπλέον, παρόμοια με το ρυθμό ανανέωσης, μπορεί να αναβαθμίσει τους πόντους ζημίας αγοράζοντας αναβαθμίσεις από τους εμπόρους που θα συναντήσει. Το σύστημα αυτό δίνει τη δυνατότητα στον παίκτη να προσαρμόσει το στυλ μάχης του, επιλέγοντας αν θα επενδύσει περισσότερο στις φυσικές επιθέσεις ή αν θα βασιστεί στη μαγεία.

Η ισορροπημένη διαχείριση των πόντων ζημίας επιτρέπει στον παίκτη να βελτιώσει τις επιδόσεις του στη μάχη και να προσαρμοστεί ανάλογα με τις προκλήσεις του παιχνιδιού. Ένας χαρακτήρας με υψηλούς πόντους ζημίας μπορεί να εξουδετερώσει τους εχθρούς πιο γρήγορα, αλλά ίσως χρειάζεται να θυσιάσει άλλες στατιστικές, όπως άμυνα ή μάνα, για να επιτύχει αυτή την ενίσχυση.

4.2.3 Φίλτρα ζωής και μανά



Εικόνα 4.8 Αναλώσιμα φίλτρα

Τα φίλτρα ζωής και μανά είναι αντικείμενα που επιτρέπουν στον παίκτη να αναπληρώνει την κόκκινη μπάρα ζωής (HP) ή την μπλε μπάρα μάνα (Mana) αντίστοιχα. Στην οθόνη εμφανίζονται ως εικονίδια (στο πάνω δεξιά τμήμα του UI) και συνοδεύονται από έναν αριθμό που υποδηλώνει πόσα φίλτρα διαθέτει ο παίκτης.

4.2.4 Μπάρα επιπέδου

Η μπάρα επιπέδου εμφανίζεται συνήθως κάτω από τον χαρακτήρα και αποτυπώνει την πρόοδο του παίκτη προς το επόμενο επίπεδο. Στο αριστερό άκρο της μπάρας εμφανίζεται το τρέχον επίπεδο, ενώ στο δεξιό άκρο απεικονίζεται το επόμενο επίπεδο. Καθώς ο παίκτης κερδίζει εμπειρία (**EXP**) από μάχες ή αγοράζοντας εμπειρία από εμπόρους, η μπάρα γεμίζει. Όταν αυτή φτάσει στο μέγιστο, ο παίκτης ανεβαίνει επίπεδο, κερδίζοντας νέα στατιστικά και ξόρκια που βελτιώνουν τις επιδόσεις του στο παιχνίδι.

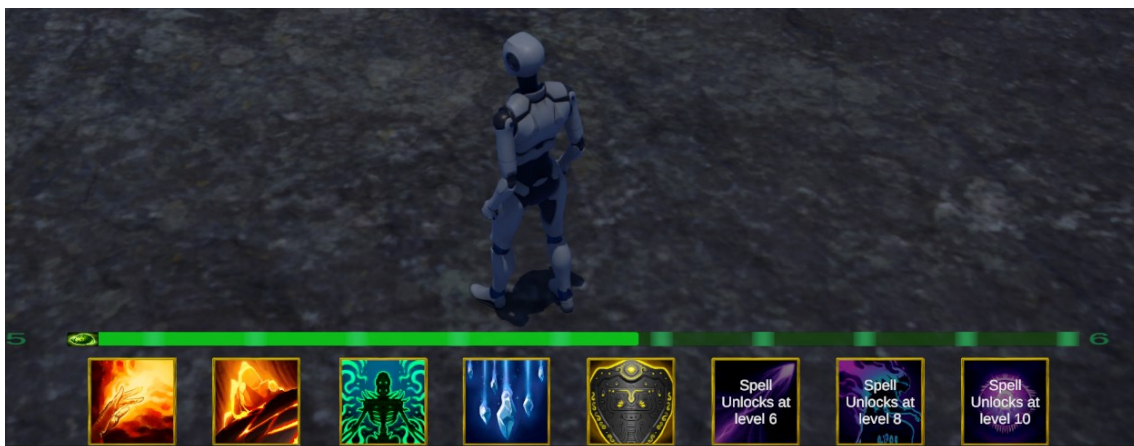


Εικόνα 4.9 UI ξορκιών

Αυτός ο μηχανισμός προσθέτει ένα αίσθημα προόδου στο gameplay, επιτρέποντας στον παίκτη να βλέπει απτά αποτελέσματα της προσπάθειάς του και να σχεδιάζει τη στρατηγική του γύρω από την ταχεία απόκτηση εμπειρίας.

4.2.5 Γραφικό Περιβάλλον Ξορκιών (κάτω μέρος της οθόνης)

Στο κάτω μέρος της οθόνης, βρίσκεται η μπάρα των ξορκιών, η οποία περιέχει όλα τα διαθέσιμα ξόρκια του παίκτη. Κάθε ξόρκι απεικονίζεται με ένα εικονίδιο, το οποίο αντιπροσωπεύει τη μαγική του ικανότητα. Τα ξόρκια είναι οργανωμένα σε μια σειρά, επιτρέποντας στον παίκτη να επιλέγει και να εκτελεί ξόρκια γρήγορα, χωρίς να διακόπτει το παιχνίδι πατώντας τα πλήκτρα 1-8.



Εικόνα 4.10 Τα διαθέσιμα ξόρκια όταν ο παίκτης είναι επίπεδο 6

Όταν ο παίκτης ξεκινά το παιχνίδι, έχει διαθέσιμο μόνο το πρώτο ξόρκι εφόσον ξεκινά στο πρώτο επίπεδο, ενώ τα πιο ισχυρά ξόρκια ξεκλειδώνονται σταδιακά καθώς ανεβαίνει επίπεδα. Τα ξόρκια που δεν είναι ακόμα διαθέσιμα εμφανίζονται κλειδωμένα, συνοδευόμενα από μια ένδειξη που αναφέρει το επίπεδο στο οποίο θα γίνουν διαθέσιμα.

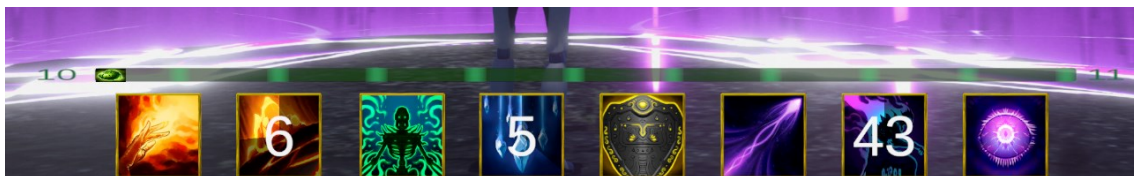
Για να χρησιμοποιήσει ένα ξόρκι, ο παίκτης πρέπει να διαθέτει αρκετό μάνα (**mana**), καθώς κάθε ξόρκι έχει συγκεκριμένο κόστος.



Εικόνα 4.11 Ανεπαρκές μάνα για τα ξόρκια 4 και 5

Εάν ο παίκτης δεν έχει επαρκές μάνα, το εικονίδιο του ξορκιού αποκτά μια έντονη μπλε απόχρωση, υποδεικνύοντας ότι δεν είναι διαθέσιμο για χρήση.

Κάθε φορά που ο παίκτης θέλει να ρίξει ένα ξόρκι, πρέπει να διαθέτει επαρκές μάνα (**mana**) και να έχει ολοκληρωθεί ο τυχόν χρόνος φόρτισης (**cooldown**).



Εικόνα 4.12 Χρόνος επιφόρτισης σε ξόρκια

Επιπλέον, κάποια ξόρκια έχουν χρόνο φόρτισης (**cooldown**), ο οποίος αποτρέπει την κατάχρησή τους. Όταν ένα ξόρκι χρησιμοποιηθεί, εμφανίζεται μια οπτική ένδειξη χρόνου επαναφόρτισης πάνω στο εικονίδιό του, με ένα σκίασμα που σταδιακά εξαφανίζεται (όπως κινούνται οι δείκτες σε ένα ρολόι), δείχνοντας πόσος χρόνος απομένει μέχρι να μπορεί να χρησιμοποιηθεί ξανά.

Αυτό το σύστημα επιτρέπει στον παίκτη να δημιουργεί συνδυασμούς ξορκιών και να αναπτύσσει στρατηγικές μάχης βασισμένες στη σωστή διαχείριση του μάνα και του **cooldown**. Η σωστή τοποθέτηση των ξορκιών στη μπάρα μπορεί να βοηθήσει τον παίκτη να έχει γρήγορη πρόσβαση στις πιο σημαντικές του ικανότητες, επιτρέποντάς του να αντιδρά άμεσα σε κάθε κατάσταση στη μάχη.

4.3 Έμποροι (NPCs)

Οι έμποροι (merchants) είναι NPCs (Non-Playable Characters) που επιτρέπουν στον παίκτη να αγοράσει αναβαθμίσεις μέσα στο παιχνίδι.



Εικόνα 4.13 Έμπορος με χρυσή κουκούλα

Κάθε 5 επίπεδα ο έμπορος εμφανίζεται και πατώντας το πλήκτρο «Ε», εμφανίζεται ένα παράθυρο με τις διαθέσιμες αναβαθμίσεις και κόστη.



Εικόνα 4.14 UI διαθέσιμων αναβαθμίσεων

Το κουμπί κλεισίματος (Close Button) βρίσκεται στο πάνω δεξί μέρος της οθόνης και επιτρέπει στον παίκτη να κλείσει το μενού και να επιστρέψει στο παιχνίδι χωρίς να κάνει αγορά.

4.3.1 Κάρτες αναβαθμίσεων

Κάθε διαθέσιμη αναβάθμιση ή αντικείμενο εμφανίζεται ως κάρτα με:

- Εικονίδιο που αντιπροσωπεύει το είδος της αναβάθμισης.
- Περιγραφή της αναβάθμισης (π.χ. προσθήκη ζωής, άμυνας, μάνα ή φίλτρων).
- Γραμμή τιμής που δείχνει πόσο κοστίζει η αναβάθμιση.
- Ποσό θα αναβαθμιστεί η κάθε ικανότητα.

Οι έμποροι δεν έχουν πάντα τις ίδιες τιμές ή τις ίδιες αναβαθμίσεις. Κάθε 5 επίπεδα ο παίκτης συναντά έναν νέο έμπορο, ο οποίος μπορεί να προσφέρει διαφορετικά boosts ή αντικείμενα.

Η τιμή μιας αναβάθμισης αλλάζει ανάλογα με την πρόοδο του παίκτη, το επίπεδό του ή τη διαθεσιμότητα των πόρων. Για παράδειγμα, ένας έμπορος μπορεί να προσφέρει +30 μάνα έναντι 100G, ενώ ένας άλλος έμπορος μπορεί να προσφέρει +40 μάνα έναντι 150G.

Επίσης, στο κάτω δεξί μέρος εμφανίζεται το διαθέσιμο χρυσό (**G**) του παίκτη. Ο παίκτης πρέπει να διαχειριστεί σωστά το υπόλοιπό του, επιλέγοντας τις αναβαθμίσεις που του ταιριάζουν καλύτερα.

Οι έμποροι λειτουργούν ως σημεία προόδου, καθώς εμφανίζονται κάθε 5 επίπεδα και επιτρέπουν στον παίκτη να βελτιώσει τις ικανότητές του. Ανάλογα με το επίπεδο του παίκτη, ορισμένες αναβαθμίσεις μπορεί να είναι πιο χρήσιμες από άλλες (π.χ. στα πρώτα επίπεδα ίσως είναι πιο χρήσιμο να αυξήσει το HP, ενώ στα υψηλότερα επίπεδα να ενισχύσει το μάνα και τη ζημιά του).

Επειδή οι τιμές και τα στατιστικά των αναβαθμίσεων διαφέρουν ανά έμπορο, ο παίκτης μπορεί να επιλέξει να αποταμιεύσει χρήματα και να περιμένει έναν επόμενο έμπορο για καλύτερες προσφορές.

Η ύπαρξη διαφορετικών εμπόρων με μεταβαλλόμενες τιμές προσθέτει στρατηγικό βάθος στο παιχνίδι, αφού ο παίκτης πρέπει να αποφασίσει αν θα ξοδέψει τα χρήματά του άμεσα ή αν θα περιμένει για πιο συμφέρουσες επιλογές στο μέλλον.

Στην εικόνα φαίνεται το μενού του εμπορού, όπου ο παίκτης μπορεί να αγοράσει διαφορετικές αναβαθμίσεις που επηρεάζουν τα στατιστικά του χαρακτήρα. Οι επιλογές αυτές χωρίζονται σε **βελτιώσεις βασικών χαρακτηριστικών, πόρων και καταναλώσιμων αντικειμένων**.

4.3.2 Βελτιώσεις Βασικών Στατιστικών

Αυτές οι αναβαθμίσεις ενισχύουν μόνιμα τα κύρια στατιστικά του χαρακτήρα, επιτρέποντάς του να γίνει πιο ανθεκτικός ή ισχυρός.



Εικόνα 4.15 Αύξηση Μέγιστων Πόντων Ζωής

Αυξάνει τη συνολική αντοχή του χαρακτήρα, επιτρέποντάς του να επιβιώνει περισσότερο στις μάχες. Ιδανικό για παίκτες που δέχονται συχνά ζημιά και θέλουν να αυξήσουν τη διάρκειά τους στη μάχη.



Εικόνα 4.16 Αύξηση Μέγιστων Πόντων Άμυνας

Ενισχύει την προστασία του χαρακτήρα, μειώνοντας τη ζημιά που δέχεται. Χρήσιμο αν ο παίκτης θέλει να αντέχει περισσότερες επιθέσεις χωρίς να χάνει ζωή.



Εικόνα 4.17 Αύξηση Βασικής Ζημιάς

Αυξάνει την ισχύ των βασικών επιθέσεων, κάνοντας τον παίκτη πιο αποτελεσματικό στη μάχη. Αυτή η αναβάθμιση κάνει τις βασικές επιθέσεις και τις επιθέσεις των ξορκιών πιο δυνατές και τείνει να είναι πιο ακριβή σε σχέση με άλλες αναβαθμίσεις.

4.3.3 Αναβαθμίσεις Πόρων

Αυτές οι επιλογές αυξάνουν τους πόρους που χρησιμοποιεί ο χαρακτήρας για ξόρκια και ειδικές ικανότητες.



Εικόνα 4.18 Αύξηση Μέγιστων Πόντων Μάνα

Αυξάνει τη δεξαμενή μάνα, επιτρέποντας στον παίκτη να ρίχνει περισσότερα ξόρκια πριν εξαντληθεί.



Εικόνα 4.19 Αύξηση Ρυθμού Αναγέννησης Μάνα

Βελτιώνει τον ρυθμό αναπλήρωσης του μάνα, μειώνοντας τον χρόνο που απαιτείται για να γεμίσει ξανά η μπάρα του μάνα. Αυτή η αναβάθμιση είναι χρήσιμη αν ο παίκτης θέλει να μειώσει την εξάρτησή του από φίλτρα μάνα, καθώς το μάνα του θα αναπληρώνεται σταδιακά κατά τη διάρκεια του παιχνιδιού.



Εικόνα 4.20 Αύξηση Πόντων Εμπειρίας

Επιτρέπει στον παίκτη να αποκτήσει επιπλέον πόντους εμπειρίας (XP), επιταχύνοντας τη διαδικασία ανόδου επιπέδου. Αυτή η αναβάθμιση είναι ιδιαίτερα χρήσιμη για παίκτες που θέλουν να ξεκλειδώσουν νέες ικανότητες γρήγορα ή να φτάσουν σε υψηλότερα επίπεδα με λιγότερη προσπάθεια.

4.3.4 Καταναλώσιμα Αντικείμενα

Αυτά τα αντικείμενα δεν είναι μόνιμες αναβαθμίσεις, αλλά αναλώσιμα που μπορούν να χρησιμοποιηθούν στη μάχη για άμεση αποκατάσταση ζωής ή μάννα.



Εικόνα 4.21 Φίλτρο Ζωής

Προσθέτει ένα επιπλέον φίλτρο ζωής στο απόθεμα του παίκτη, επιτρέποντάς του να αναπληρώσει τη ζωή του όποτε χρειάζεται.



Εικόνα 4.22 Φίλτρο Μάννα

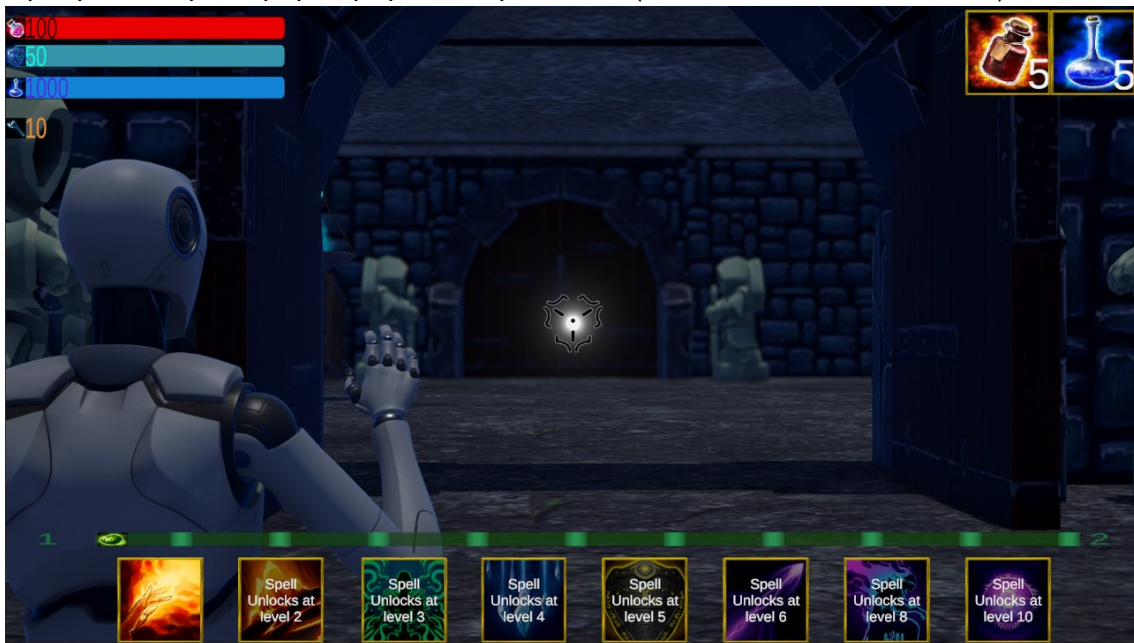
Προσθέτει ένα επιπλέον φίλτρο μάννα, το οποίο επιτρέπει την άμεση αποκατάσταση του μάννα κατά τη διάρκεια της μάχης. Συνήθως τα παραπάνω φίλτρα είναι χαμηλότερα σε κόστος εφόσον είναι αναλώσιμα.

4.4 Ξόρκια παίχτη

Σε αυτήν την ενότητα, θα παρουσιάσουμε τις ικανότητες του παίχτη που έχει διαθέσιμα στο παιχνίδι. Όπως αναφέρθηκε προηγουμένως, ο παίχτης μπορεί να πραγματοποιήσει ένα ξόρκι πατώντας έναν αριθμό (1-8). Μερικά ξόρκια πραγματοποιούνται σε συνδυασμό με το αριστερό κλικ του ποντικιού εφόσον χρειάζεται ο παίχτης να στοχεύσει προς την κατεύθυνσή του ξορκιού. Αξίζει επίσης να σημειωθεί ότι κάθε φορά που πραγματοποιείται ένα ξόρκι αναπαράγεται σχετικός ήχος ανάλογα το ξόρκι.

4.4.1 Ξόρκι πύρινη σφαίρας(Fireball)

Το ξόρκι αυτό εκτοξεύει μια σφαίρα φωτιάς που προκαλεί ζημιά σε έναν εχθρό. Κατά την ενεργοποίηση του ξορκιού, αν ο παίχτης πατήσει και κρατήσει πατημένο το πλήκτρο "1", η κάμερα αλλάζει από third-person σε first-person mode, μετατρέποντας την προοπτική σε προβολή πρώτου προσώπου (First-Person Shooter - FPS).



Εικόνα 4.23 Στόχευση ξορκιού(Projectile)

Σε αυτή τη λειτουργία, εμφανίζεται ένας δείκτης στόχευσης (**crosshair**) στο κέντρο της οθόνης, διευκολύνοντας τον παίκτη στην ακριβή στόχευση του ξορκιού. Όταν ο παίχτης έχει επιλέξει τον στόχο του, μπορεί να πατήσει το αριστερό κλικ του ποντικιού για να ρίξει τη σφαίρα φωτιάς.

Αφού πατηθεί το αριστερό κλικ του ποντικιού, ο χαρακτήρας κάνει animation που υποδεικνύει την ρήξη της σφαίρας με τα χεριά του. Η σφαίρα κινείται προς την κατεύθυνση της στόχευσης και εξαφανίζεται είτε αν συγκρουστεί με έναν εχθρό είτε αν χτυπήσει κάποιον τοίχο ή άλλο εμπόδιο.

Αν η σφαίρα βρει στόχο, προκαλεί ζημιά στον εχθρό, μειώνοντας τη μπάρα ζωής του. Σε περίπτωση που η σφαίρα χάσει τον στόχο της και προσκρούσει σε μια επιφάνεια, διαλύεται χωρίς να έχει περαιτέρω επίδραση.



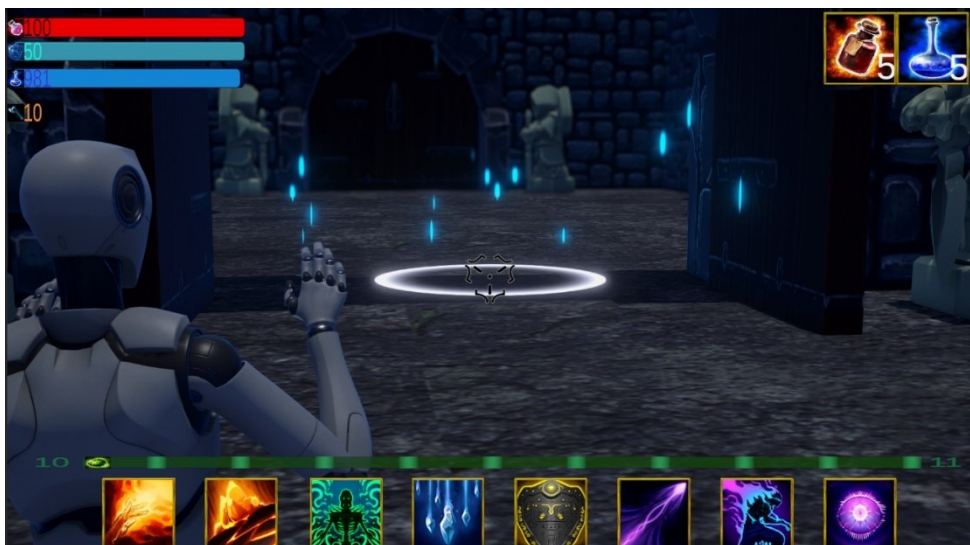
Εικόνα 4.24 Εκτέλεση ξορκιού πυρός

Αυτό το σύστημα στοχευμένης επίθεσης προσφέρει στον παίκτη μεγαλύτερο έλεγχο στις μαγικές του επιθέσεις, καθώς η δυνατότητα εναλλαγής κάμερας σε first-person mode επιτρέπει πιο ακριβείς βολές, βελτιώνοντας τη συνολική εμπειρία μάχης.

4.4.2 Ξόρκι Μετεωριτών (AOE Comet Spell)

Το δεύτερο ξόρκι, Μετεωρίτες (Comet Spell), είναι μια ισχυρή AOE (Area of Effect) ικανότητα, που επιτρέπει στον παίκτη να καλέσει μια βροχή κομητών σε μια συγκεκριμένη περιοχή, προκαλώντας συνεχή ζημιά (damage over time - DoT) σε όλους τους εχθρούς που βρίσκονται μέσα στη ζώνη επίθεσης.

Για να χρησιμοποιήσει το ξόρκι, ο παίκτης πρέπει να πατήσει και να κρατήσει πατημένο το πλήκτρο "2". Αυτή η ενέργεια αλλάζει προσωρινά την προοπτική της κάμερας, μεταφέροντας τον παίκτη σε first-person mode, επιτρέποντάς του να επιλέξει την ακριβή τοποθεσία όπου θα πέσουν οι μετεωρίτες.



Εικόνα 4.25 Στόχευση ξορκιού που επηρεάζει πολλούς εχθρούς ταυτόχρονα σε μια περιοχή

Κατά την εναλλαγή σε προβολή πρώτου προσώπου, εμφανίζεται ένας κύκλος στόχευσης στο κέντρο της οθόνης, βοηθώντας τον παίκτη να επιλέξει το σημείο ρίψης. Όταν ο παίκτης έχει στοχεύσει το επιθυμητό σημείο, μπορεί να πατήσει το αριστερό κλικ του ποντικιού για να ενεργοποιήσει την επίθεση.



Εικόνα 4.26 Εκτέλεση ξορκιού μετεωριτών

Όταν ο παίκτης πατήσει το αριστερό κλικ, ο χαρακτήρας εκτελεί ένα animation εκτέλεσης ξορκιού, σηκώνοντας το χέρι του ψηλά και στη συνέχεια κατεβάζοντάς το δυναμικά προς τα κάτω. Αυτή η κίνηση υποδηλώνει την ενεργοποίηση του ξορκιού, προσδίδοντας μια οπτική και κινητική ένδειξη της μαγικής διαδικασίας, ενισχύοντας έτσι την αίσθηση αλληλεπίδρασης και δύναμης του χαρακτήρα.

Μόλις το ξόρκι εκτελεστεί, πολλαπλοί μετεωρίτες αρχίζουν να πέφτουν στην επιλεγμένη περιοχή, χτυπώντας όλους τους εχθρούς μέσα στη ζώνη δράσης. Οι μετεωρίτες δεν προκαλούν άμεση έκρηξη, αλλά αντίθετα δημιουργούν διαρκή ζημιά σε όποιον εχθρό παραμείνει στην περιοχή του χτυπήματος.

Αν ένας εχθρός παραμείνει μέσα στην περιοχή της επίθεσης, συνεχίζει να δέχεται ζημιά για όσο διαρκεί το ξόρκι. Το ξόρκι είναι ιδιαίτερα αποτελεσματικό εναντίον ομάδων εχθρών, καθώς καλύπτει μεγάλη επιφάνεια και μπορεί να επηρεάσει πολλούς στόχους ταυτόχρονα.

Σε αντίθεση με τη σφαίρα φωτιάς (Fireball), που στοχεύει έναν μεμονωμένο εχθρό, το Comet Spell είναι ιδανικό για μάχες ενάντια σε πλήθος εχθρών (crowd control).

Επιπλέον, συνδυάζεται καλά με άλλα ξόρκια επιβράδυνσης (slow effects), καθώς οι εχθροί που δεν μπορούν να ξεφύγουν από την περιοχή των μετεωριτών θα υποστούν όλη τη ζημιά του ξορκιού.

4.4.3 Ξόρκι Θεραπείας

Το τρίτο ξόρκι, Θεραπεία (Heal), επιτρέπει στον παίκτη να ανακτήσει ένα ποσοστό της χαμένης ζωής του (HP), χωρίς να χρειάζεται να χρησιμοποιήσει φίλτρα υγείας. Σε αντίθεση με τα επιθετικά ξόρκια, τα οποία απαιτούν στόχευση με το ποντίκι, η ενεργοποίηση του ξορκιού γίνεται αυτόματα μόλις ο παίκτης πατήσει το αντίστοιχο πλήκτρο.

Κατά την εκτέλεση του ξορκιού, ο χαρακτήρας παραμένει ακίνητος, εισερχόμενος σε μια κατάσταση συγκέντρωσης, όπου πραγματοποιεί ένα animation θεραπείας. Κατά τη διάρκεια της διαδικασίας, σηκώνει ελαφρώς τα χέρια του προς το σώμα του και εκπέμπει μια πράσινη μαγική ενέργεια, η οποία περιβάλλει το σώμα του και υποδηλώνει την αποκατάσταση της ζωής του.



Εικόνα 4.27 Εκτέλεση ξορκιού θεραπείας

Η Θεραπεία διαρκεί για ένα σύντομο χρονικό διάστημα, κατά το οποίο ο χαρακτήρας δεν μπορεί να κινηθεί. Επειδή ο παίκτης παραμένει ακινητοποιημένος, είναι ευάλωτος στις επιθέσεις των εχθρών, γεγονός που μπορεί να τον εκθέσει σε κινδύνους, ειδικά αν βρίσκεται σε μάχη.

Σε σύγκριση με τα φίλτρα υγείας, τα οποία μπορούν να καταναλωθούν εν κινήσει, το ξόρκι θεραπείας απαιτεί στατική τοποθέτηση, καθιστώντας το λιγότερο ασφαλές σε καταστάσεις όπου ο παίκτης δέχεται πολλαπλές επιθέσεις.

Λόγω του ότι το Heal Spell αναγκάζει τον παίκτη να μείνει ακίνητος, είναι πιο αποτελεσματικό όταν χρησιμοποιείται σε ασφαλές περιβάλλον, όπως:

- Μετά το τέλος μιας μάχης, όταν δεν υπάρχουν ενεργές απειλές.
- Όταν οι εχθροί βρίσκονται μακριά ή έχουν ακινητοποιηθεί με άλλο ξόρκι (π.χ. Stun Bubble).
- Σε προστατευμένες περιοχές, όπως πίσω από κάλυψη ή σε έναν ασφαλή διάδρομο, όπου ο παίκτης δεν μπορεί εύκολα να δεχτεί επίθεση.

4.4.4 Ξόρκι Παγωμένων Κομητών

Το τέταρτο ξόρκι, Παγωμένοι Κομήτες (Frozen Comet Spell), είναι ένα AOE (Area of Effect) ξόρκι ελέγχου εχθρών (crowd control) που καλεί κομήτες από πάγο, οι οποίοι πέφτουν στο έδαφος, προκαλώντας μέτρια ζημιά και σημαντική επιβράδυνση στην ταχύτητα μετακίνησης των εχθρών.



Εικόνα 4.28 Στόχευση ξορκιού παγωμένων κομητών

Όταν ο παίκτης πατήσει και κρατήσει πατημένο το πλήκτρο "4", η κάμερα αλλάζει σε προβολή πρώτου προσώπου (First-Person Shooter - FPS), δίνοντάς του τη δυνατότητα να στοχεύσει την περιοχή όπου θα πέσουν οι κομήτες. Εμφανίζεται ένας δείκτης στόχευσης (crosshair) που βοηθά τον παίκτη να επιλέξει με ακρίβεια το σημείο πρόσκρουσης. Μόλις ο παίκτης πατήσει το αριστερό κλικ του ποντικιού, ξεκινά η πτώση των παγωμένων κομητών.



Εικόνα 4.29 Εκτέλεση ξορκιού παγωμένων κομητών



Εικόνα 4.30 Οι εχθροί αποκτάνε μια μπλε απόχρωση στα γραφικά τους

Καθώς οι κομήτες χτυπούν το έδαφος, παγωμένα σωματίδια και παγωμένοι κρύσταλλοι εκτοξεύονται στην περιοχή, δημιουργώντας ένα παγωμένο πεδίο που επιβραδύνει δραστικά την κίνηση όλων των εχθρών που βρίσκονται μέσα σε αυτό.

Ιδιότητες και Μηχανισμός του Ξορκιού:

- Δεν προκαλεί μεγάλη ζημιά, σε αντίθεση με άλλα επιθετικά ξόρκια, όπως η Πύρινη Σφαίρα ή ο Κεραυνός.
- Η κύρια δύναμή του έγκειται στην επιβράδυνση των εχθρών, κάνοντας τους πιο εύκολο στόχο για επόμενες επιθέσεις.
- Η επιβράδυνση ισχύει για συγκεκριμένο χρονικό διάστημα, αλλά μπορεί να επηρεάσει πολλούς εχθρούς ταυτόχρονα, καθιστώντας το ιδανικό για μάχες ενάντια σε ομάδες εχθρών.
- Αν οι εχθροί δεν βγουν από την παγωμένη περιοχή, παραμένουν επιβραδυμένοι μέχρι να λήξει η επίδραση του ξορκιού.

Το ξόρκι Παγωμένοι Κομήτες είναι εξαιρετικά χρήσιμο για έλεγχο του πεδίου μάχης, καθώς η μείωση ταχύτητας των εχθρών επιτρέπει στον παίκτη να:

- Αποφύγει εύκολα τις επιθέσεις των εχθρών, ειδικά εναντίον γρήγορων αντιπάλων που διαφορετικά θα τον πλησίαζαν γρήγορα.
- Συνδυάσει το ξόρκι με άλλα AOE damage spells, όπως το Comet Spell ή το Magic Circle, εκμεταλλευόμενος την επιβράδυνση για να διατηρήσει τους εχθρούς μέσα στην περιοχή της επίθεσης.
- Διαφύγει από επικίνδυνες καταστάσεις, καθώς οι επιβραδυμένοι εχθροί δεν μπορούν να τον ακολουθήσουν εύκολα αν αποφασίσει να υποχωρήσει.
- Στοχεύσει με μεγαλύτερη ακρίβεια με άλλα ξόρκια ή ranged επιθέσεις, καθώς οι αργοί εχθροί γίνονται ευκολότεροι στόχοι.

4.4.5 Ξόρκι Μαγικής Ασπίδας (Shield Spell)

Το πέμπτο ξόρκι, Μαγική Ασπίδα (Shield Spell), δημιουργεί ένα προστατευτικό πεδίο γύρω από τον χαρακτήρα, το οποίο τον καθιστά άτρωτο σε ζημιά για μερικά δευτερόλεπτα. Πρόκειται για ένα αμυντικό ξόρκι, σχεδιασμένο για να προσφέρει προσωρινή προστασία, δίνοντας στον παίκτη τη δυνατότητα να αντέξει ισχυρές επιθέσεις ή να προετοιμαστεί για αντεπίθεση.



Εικόνα 4.31 Εκτέλεση ξορκιού ασπίδας

Όταν ο παίκτης πατήσει το αντίστοιχο πλήκτρο, το ξόρκι ενεργοποιείται αυτόματα, χωρίς να απαιτείται στόχευση ή χρήση του ποντικιού. Κατά τη διάρκεια του ξορκιού, ο χαρακτήρας παραμένει ακίνητος, παρόμοια με το ξόρκι Θεραπείας (Heal Spell).

Ο χαρακτήρας εκτελεί ένα μοναδικό animation, όπου ανοίγει τα χέρια του και συγκεντρώνει ενέργεια, σχηματίζοντας μια ημιδιαφανή ασπίδα γύρω του. Ένα λαμπερό μαγικό πεδίο εμφανίζεται περιμετρικά του χαρακτήρα, δείχνοντας ότι το ξόρκι βρίσκεται σε ισχύ.

Ιδιότητες και Περιορισμοί:

- Κατά τη διάρκεια της ασπίδας, ο παίκτης δεν δέχεται ζημιά, ανεξάρτητα από το είδος της επίθεσης (σωματική, μαγική, βλήματα).
- Η προστασία διαρκεί για συγκεκριμένο χρονικό διάστημα, μετά το οποίο η ασπίδα εξαφανίζεται.
- Όπως και με το Heal Spell, ο χαρακτήρας παραμένει στατικός, γεγονός που τον καθιστά ευάλωτο σε τακτικές περικύκλωσης ή επιθέσεις από πολλούς εχθρούς μόλις το ξόρκι λήξει.
- Αν και ο παίκτης δεν μπορεί να κινηθεί, μπορεί να προετοιμάσει την επόμενη του κίνηση, όπως να χρησιμοποιήσει ένα επιθετικό ξόρκι μόλις η ασπίδα εξαφανιστεί.

4.4.6 Ηλεκτρική Σφαίρα (Thunderball)

Το έκτο ξόρκι, Κεραυνοφόρος Σφαίρα (Thunderball Spell), είναι ένα ισχυρό επιθετικό ξόρκι μονής στόχευσης, το οποίο εκτοξεύει μια ηλεκτρική σφαίρα που προκαλεί σημαντική ζημιά σε έναν εχθρό.



Εικόνα 4.32 Στόχευση ξορκιού ηλεκτρικής σφαίρας

Παρόμοια με το ξόρκι Fireball, Για να χρησιμοποιήσει το ξόρκι, ο παίκτης πατά και κρατά πατημένο το πλήκτρο "6", ενεργοποιώντας την προβολή πρώτου προσώπου. Εμφανίζεται ένας δείκτης στόχευσης (crosshair) στο κέντρο της οθόνης, επιτρέποντας ακριβή στόχευση σε έναν συγκεκριμένο εχθρό.



Εικόνα 4.33 Εκτέλεση ξορκιού

Όταν ο παίκτης πατήσει το αριστερό κλικ του ποντικιού, ο χαρακτήρας εκτελεί ένα animation, σηκώνοντας το χέρι του προς τα εμπρός, και εκτοξεύει μια λαμπερή ηλεκτρική σφαίρα προς την κατεύθυνση του στόχου. Σε σύγκριση με το ξόρκι Fireball, η ηλεκτρική σφαίρα εκτοξεύεται και φτάνει στον στόχο άμεσα και προκαλεί περισσότερη ζημιά. Αλλά, απαιτεί περισσότερο μανά και έχει μεγαλύτερο χρόνο φόρτισης.

4.4.7 Ξόρκι Μαγικού Κύκλου (Magic Circle Spell)

Το έβδομο ξόρκι, Μαγικός Κύκλος (Magic Circle Spell), είναι ένα εξαιρετικά ισχυρό ΑΟΕ (Area of Effect) ξόρκι, το οποίο δημιουργεί έναν τεράστιο μαγικό κύκλο γύρω από τον παίκτη. Όλοι οι εχθροί που βρίσκονται μέσα στην ακτίνα του κύκλου δέχονται πολύ μεγάλη ζημιά, καθιστώντας το ένα από τα πιο καταστροφικά ξόρκια του παιχνιδιού.



Εικόνα 4.34 Εκτέλεση ξορκιού μαγικού κύκλου

Ο παίκτης πατά το αντίστοιχο πλήκτρο για να ενεργοποιήσει το ξόρκι, χωρίς να χρειάζεται στόχευση ή χρήση του ποντικιού. Δεν μπορεί να καθορίσει που θα χτυπήσει το ξόρκι, καθώς η περιοχή δράσης του σχηματίζεται αυτόματα γύρω από τον χαρακτήρα.

Κατά τη διάρκεια του ξορκιού, ο παίκτης εκτελεί ένα θεαματικό animation, όπου σηκώνει τα χέρια του προς τα πάνω και ενεργοποιεί έναν τεράστιο, λαμπερό μαγικό κύκλο στο έδαφος. Ο χαρακτήρας παραμένει ακίνητος καθ' όλη τη διάρκεια της εκτέλεσης του ξορκιού.



Εικόνα 4.35 Επίδειξη ζημίας σε εχθρό

Οι αντίπαλοι που βρίσκονται μέσα στην περιοχή του κύκλου αρχίζουν να δέχονται τεράστια ζημιά, η οποία εφαρμόζεται άμεσα. Όταν ολοκληρωθεί το ξόρκι, ο κύκλος εξαφανίζεται, αφήνοντας πίσω του πιθανούς επιζώντες εχθρούς με ελάχιστη ζωή ή πλήρως εξουδετερωμένους αντιπάλους αναλόγως την ζωή των εχθρών και των πόντων ζημίας του παίχτη.

Όπως εμφανίζεται, καλύπτει μεγάλη περιοχή γύρω από τον παίκτη, προκαλώντας υψηλή ζημιά σε όλους τους εχθρούς που βρίσκονται κοντά του. Όμως, ο χαρακτήρας δεν μπορεί να μετακινηθεί, καθιστώντας τον ευάλωτο σε επιθέσεις από εχθρούς που βρίσκονται εκτός της περιοχής δράσης του κύκλου καθώς είναι ενεργό το ξόρκι.

Επειδή το ξόρκι δεν επηρεάζει εχθρούς που βρίσκονται εκτός της ζώνης του κύκλου, ο παίκτης μπορεί να γίνει εύκολος στόχος για ranged επιθέσεις, όπως βλήματα ή ξόρκια από απόσταση.

Είναι ιδανικό για μάχες ενάντια σε πλήθος εχθρών, καθώς η μεγάλη ζημιά και η ευρεία ακτίνα δράσης μπορούν να εξοντώσουν πολλούς αντιπάλους ταυτόχρονα.

Επιπλέον, είναι ένα από τα πιο δυνατά ξόρκια, γι' αυτό απαιτεί μεγάλη ποσότητα μάνα για να εκτελεστεί, γεγονός που σημαίνει ότι ο παίκτης θα πρέπει να έχει αρκετά αποθέματα πριν το χρησιμοποιήσει.

Τέλος, το ξόρκι έχει μεγάλο χρόνο αναμονής πριν μπορέσει να ξαναχρησιμοποιηθεί. Αυτό σημαίνει ότι ο παίκτης πρέπει να επιλέξει τη σωστή στιγμή για να το εκτελέσει, καθώς δεν μπορεί να το χρησιμοποιεί συνεχώς στη μάχη.

4.4.8 Ξόρκι Φυσαλίδας Παράλυσης (Stun Bubble Spell)

Το όγδοο ξόρκι, Φυσαλίδα Παράλυσης (Stun Bubble Spell), είναι ένα ξόρκι ελέγχου (**crowd control**) που παγιδεύει έναν ή περισσότερους εχθρούς μέσα σε μια ενεργειακή φυσαλίδα, αναγκάζοντάς τους να αιωρούνται στον αέρα και καθιστώντας τους ανίκανους να κινηθούν ή να επιτεθούν για μερικά δευτερόλεπτα. Αν και το ξόρκι δεν προκαλεί ζημιά, είναι εξαιρετικά χρήσιμο για να σταματήσει επικίνδυνους αντιπάλους και να δώσει στον παίκτη χρόνο να προετοιμάσει την επόμενη κίνησή του.



Εικόνα 4.36 Εκτέλεση ξορκιού φυσαλίδας

Ο παίκτης πατά και κρατά πατημένο το πλήκτρο "8", ενεργοποιώντας το first-person mode. Εμφανίζεται ένας δείκτης στόχευσης (crosshair), επιτρέποντας στον παίκτη να στοχεύσει έναν εχθρό ή μια περιοχή όπου θα ενεργοποιηθεί η φυσαλίδα.



Εικόνα 4.37 Ο εχθρός παγιδεύεται μέσα στην φυσαλίδα



Εικόνα 4.38 Ο εχθρός πέφτει στο πάτωμα αφού τελειώσει το ξόρκι

Όταν ο παίκτης πατήσει το αριστερό κλικ του ποντικιού, ο χαρακτήρας εκτελεί ένα animation όπου απλώνει τα χέρια του προς τον στόχο, και μια μεγάλη ενεργειακή φουσαλίδα δημιουργείται γύρω από τον εχθρό.

Ο εχθρός αιωρείται μέσα στη φουσαλίδα, παγιδευμένος στον αέρα και ανίκανος να κινηθεί ή να επιτεθεί για τη διάρκεια του ξορκιού.

Η φουσαλίδα εξαφανίζεται μετά από λίγα δευτερόλεπτα, αφήνοντας τον εχθρό να πέσει ξανά στο έδαφος και να συνεχίσει τη μάχη.

Ιδιότητες και Ιδιαίτερα Χαρακτηριστικά:

- Ισχυρό εργαλείο crowd control: Αντί να προκαλεί ζημιά, σταματά έναν ή περισσότερους εχθρούς, επιτρέποντας στον παίκτη να ελέγξει τη μάχη.
- Η φουσαλίδα μπορεί να παγιδεύσει περισσότερους από έναν εχθρούς αν βρίσκονται πολύ κοντά μεταξύ τους, καθιστώντας την εξαιρετικά χρήσιμη για ομάδες εχθρών.
- Δεν επηρεάζει bosses ή πολύ δυνατούς εχθρούς, οι οποίοι μπορεί να έχουν αντίσταση στην παράλυση.
- Δεν προκαλεί καμία ζημιά, πράγμα που σημαίνει ότι ο παίκτης πρέπει να επιτεθεί μετά το τέλος της επίδρασης για να εκμεταλλευτεί το πλεονέκτημα.
- Χαμηλό κόστος μάννα, σε σύγκριση με άλλα ξόρκια, επιτρέποντας πιο συχνή χρήση του σε μάχες.
- Το ξόρκι έχει μεγάλο χρόνο επαναφόρτισης, που σημαίνει ότι ο παίκτης δεν μπορεί να το χρησιμοποιεί συνεχώς και πρέπει να επιλέγει στρατηγικά τη στιγμή που θα το ενεργοποιήσει.

4.5 Εχθροί

Οι εχθροί έχουν διαφορά μοντέλα που τα βρήκα κυρίως από την ιστοσελίδα **Mixamo** που παρέχει και τα animations των χαρακτήρων. Στο παιχνίδι υπάρχουν τρεις βασικές κατηγορίες εχθρών:

- Εχθροί μάχης σώμα με σώμα, που βασίζονται σε κοντινές επιθέσεις και προσπαθούν να πλησιάσουν τον παίκτη.
- Εχθροί μάχης από απόσταση, οι οποίοι επιτίθενται χρησιμοποιώντας βλήματα ή μαγικά ξόρκια.
- Final Bosses, οι οποίοι εναλλάσσουν επιθέσεις από κοντά και από μακριά, καθιστώντας τους πιο απρόβλεπτους και απαιτητικούς στη μάχη.

4.5.1 Εχθροί μάχης σώμα με σώμα



Εικόνα 4.39 Οι πρώτοι εχθροί

Όπως εμφανίζεται στην εικόνα, οι αντίπαλοι εντοπίζουν τον ηρώα και κινούνται προς το μέρος του.



Εικόνα 4.40 Animation επίθεσης από κοντά



Εικόνα 4.41 Animation επίθεσης από κοντά με άλλο μοντέλο ιππότη

4.5.2 Εχθροί μάχης από απόσταση



Εικόνα 4.42 Επίθεση εχθρού από απόσταση

Οι εχθροί αυτοί ρίχνουν ξόρκια προς τον χαρακτήρα, εφόσον ο χαρακτήρας βρίσκεται στο οπτικό τους πεδίο και σε μια συγκεκριμένη απόσταση.



Εικόνα 4.43 Αναζήτηση χαρακτήρα

Παρόμοια με τους εχθρούς που επιτίθενται από κοντά, όταν ο χαρακτήρας είναι μακριά από το οπτικό τους πεδίο τον αναζητάνε και πηγαίνουν στο μέρος μέχρι να μπορούν να ρίξουν την επίθεσή τους.

4.5.3 Final bosses



Εικόνα 4.44 Τελικός εχθρός παιχνιδιού

Στην τελευταία πιστά του παιχνιδιού εμφανίζονται δυο τεράστια τέρατα που έχουν αυξημένη ζωή και με αντίθεση με τους προηγούμενους εχθρούς μπορούν να επιτεθούν από κοντά και μακριά αναλόγως την απόσταση του χαρακτήρα.



Εικόνα 4.45 Επίθεση τελικού εχθρού

4.6 Μερικά στιγμιότυπα του παιχνιδιού

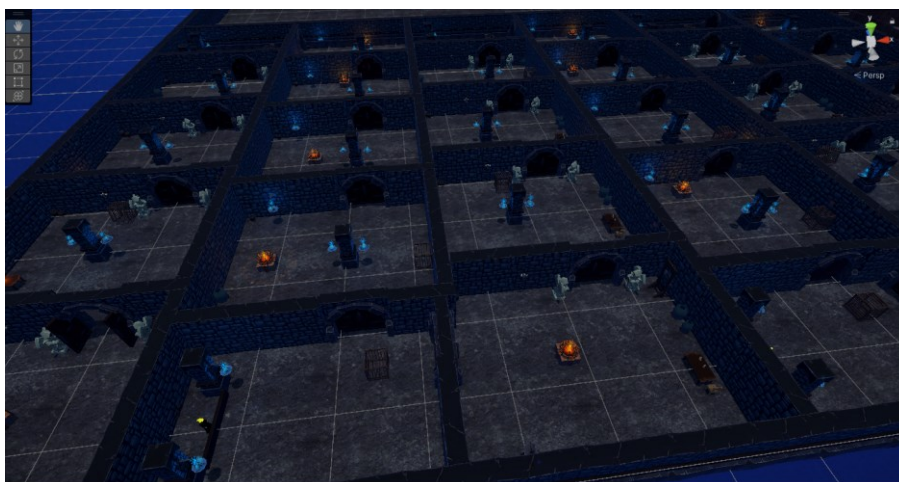
Σε αυτήν την ενότητα, θα παρουσιάσω μια σειρά από εικόνες από τα διάφορα επίπεδα του παιχνιδιού, με σκοπό να αναδείξω πώς μπορούν να δημιουργηθούν μοναδικά περιβάλλοντα χρησιμοποιώντας τα ίδια assets. Μέσα από προσεκτική τοποθέτηση αντικειμένων, φωτισμό και διαφοροποιήσεις στη σκηνοθεσία του χώρου, το παιχνίδι προσφέρει οπτική ποικιλία, αποτρέποντας τη μονοτονία και διασφαλίζοντας ότι ο παίκτης δεν βλέπει συνεχώς τα ίδια ακριβώς στοιχεία.



Εικόνα 4.46 Επίπεδο 1.3



Εικόνα 4.47 Επίπεδο 2.1



Εικόνα 4.48 Στιγμιότυπο που δείχνει πως φαίνονται οι πίστες από πάνω μέσω τον Unity Editor



Εικόνα 4.49 Επίπεδο 5.1



Εικόνα 4.50 Επίπεδο 5.2



Εικόνα 4.51 Επίπεδο 10.3



Εικόνα 4.52 Επίπεδο 10.4



Εικόνα 4.53 Επίπεδο 10.5-Ο τελευταίος έμπορος πριν την τελική μάχη



Εικόνα 4.54 Τελική μάχη

4.7 Θάνατος παίχτη

Όταν ο χαρακτήρας δεχτεί ζημιά που μειώνει τη ζωή του κάτω από το μηδέν, ενεργοποιείται animation θανάτου, όπου ο χαρακτήρας καταρρέει στο έδαφος. Την ίδια στιγμή, η κάμερα κάνει ένα ομαλό fade-out από ψηλά, δημιουργώντας μια κινηματογραφική αίσθηση της ήττας.



Εικόνα 4.55 Μήνυμα ήττας

Μετά την ολοκλήρωση του animation, εμφανίζεται ένα pop-up παράθυρο, το οποίο ενημερώνει τον παίκτη ότι ο χαρακτήρας του έχει πεθάνει. Το μήνυμα συνοδεύεται από δύο επιλογές:

- Επανεκκίνηση του επιπέδου, επιτρέποντας στον παίκτη να προσπαθήσει ξανά από την αρχή.
- Επιστροφή στο αρχικό μενού, δίνοντάς του τη δυνατότητα να αποχωρήσει από το τρέχον επίπεδο και να εξερευνήσει άλλες επιλογές.

Παράλληλα, παίζει μια ατμοσφαιρική μουσική θανάτου, συνδυάζοντας χορωδιακές φωνές και πιάνο, με στόχο να ενισχύσει τη συναισθηματική ένταση της σκηνής και να κάνει τον παίκτη να νιώσει το κλίμα της ήττας.

4.8 Τέλος παιχνιδιού

Όταν ο χαρακτήρας νικήσει όλους τους εχθρούς στο τελευταίο επίπεδο, εμφανίζεται ένα ειδικό μήνυμα επιτυχίας, το οποίο ανακοινώνει ότι όλες οι δυνάμεις του κακού έχουν ηττηθεί και ότι ο ήρωας έχει αποκτήσει την ελευθερία του.



Εικόνα 4.56 Μήνυμα νίκης

Μετά τη νίκη, ο παίκτης μπορεί να επιστρέψει στο κύριο μενού, όπου μπορεί να ξεκινήσει ένα νέο παιχνίδι από την αρχή ή να τερματίσει το πρόγραμμα.

Παράλληλα, ακούγεται μια επιική μουσική υπόκρουση, η οποία ενσωματώνει δυνατά κρουστά, ενισχύοντας το αίσθημα της νίκης και της ολοκλήρωσης. Η μουσική λειτουργεί ως μια θριαμβευτική επιβράβευση, δίνοντας στον παίκτη μια έντονη αίσθηση ικανοποίησης για την επιτυχία του.

4.9 Μενού παύσης

Τέλος, αξίζει να σημειωθεί ότι ο χρήστης μπορεί να εμφανίσει το μενού παύσης πατώντας το πλήκτρο "**Esc**", ενώ παράλληλα το παιχνίδι παγώνει, διακόπτοντας προσωρινά τη ροή του.

Μέσα από το μενού παύσης, ο χρήστης έχει στη διάθεσή του τις εξής επιλογές:

- **Resume:** Συνέχιση του παιχνιδιού από το σημείο που σταμάτησε.
- **Settings:** Πρόσβαση στο μενού ρυθμίσεων για προσαρμογή γραφικών, ήχου και άλλων παραμέτρων.
- **Controls:** Προβολή του παραθύρου με τα **key bindings**, ώστε ο παίκτης να μπορεί να δει ή να θυμηθεί τις διαθέσιμες εντολές.
- **Back to Main Menu:** Επιστροφή στο κεντρικό μενού του παιχνιδιού, διακόπτοντας την τρέχουσα πρόοδο.



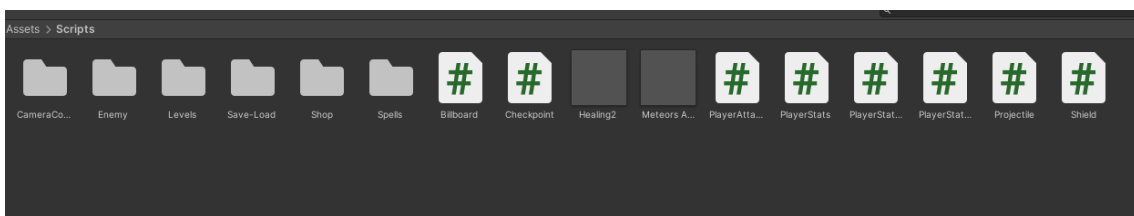
Εικόνα 4.57 Μενού παύσης

5 Scripts

Τώρα που η εκτέλεση και παρουσίαση του παιχνιδιού έχει ολοκληρωθεί, είναι η κατάλληλη στιγμή να εστιάσουμε στα τεχνικά στοιχεία της πτυχιακής εργασίας και της ανάπτυξης του παιχνιδιού, ξεκινώντας από τα scripts.

Στην Unity, ο προγραμματισμός βασίζεται στη γλώσσα C#, η οποία αποτελεί τον πυρήνα κάθε project. Ένα καλά δομημένο και αποδοτικά κωδικοποιημένο σύστημα script μπορεί να βελτιώσει σημαντικά τη διαδικασία ανάπτυξης, καθιστώντας την πιο ομαλή, αποδοτική και ευχάριστη.

Η πτυχιακή εργασία περιλαμβάνει την ανάπτυξη ενός **3D third-person παιχνιδιού** στο Unity, όπου ο παίκτης αποκτά **ξόρκια, μάχεται με εχθρούς και προοδεύει μέσω ενός RPG μηχανισμού επιπέδων**.



Εικόνα 5.1 Τα scripts του project

5.1 Player System-Σύστημα παίκτη

Η ανάπτυξη ενός τρισδιάστατου βιντεοπαιχνιδιού απαιτεί μια καλά δομημένη αρχιτεκτονική κώδικα, η οποία διαχειρίζεται τη συμπεριφορά του παίκτη, τη διεπαφή χρήστη (UI), τις μάχες, τις αλληλεπιδράσεις και τα στατιστικά του παιχνιδιού. Στην πτυχιακή αυτή εργασία, τα scripts έχουν σχεδιαστεί έτσι ώστε να λειτουργούν συνεργατικά, διαμορφώνοντας ένα δυναμικό και διαδραστικό περιβάλλον παιχνιδιού.

Τα βασικά scripts περιλαμβάνουν τη διαχείριση των στατιστικών του παίκτη (PlayerStats και PlayerStatsManager), το σύστημα επιθέσεων μέσω βλημάτων (PlayerAttack), την απεικόνιση των στατιστικών στο UI (PlayerStatsUI), καθώς και τα εφέ και τις ενέργειες που επηρεάζουν τον παίκτη (PlayerEffects). Επιπλέον, το PlayerInteract επιτρέπει στον παίκτη να αλληλεπιδρά με τον κόσμο του παιχνιδιού, ενώ τα συστήματα θεραπείας, μάνα και level-up παρέχουν μια εξελιγμένη εμπειρία προόδου.

5.1.1 PlayerStats.cs

Το PlayerStats.cs περιέχει όλα τα στατιστικά του παίκτη, καθορίζοντας τις βασικές του παραμέτρους όπως η ζωή, η άμυνα, το μάνα, η επίθεση, το XP, τα νομίσματα και ο αριθμός φίλτρων (potions).

Το script ορίζει τις μέγιστες τιμές για κάθε στατιστικό και περιλαμβάνει επιπλέον παραμέτρους όπως οι ρυθμοί αναπλήρωσης (regen rates) για ζωή και μάνα, οι οποίοι αυξάνονται με την πάροδο του χρόνου.

Αποθηκεύει επίσης τα τρέχοντα νομίσματα του παίκτη, καθώς και τη δύναμη επίθεσής του, η οποία αυξάνεται καθώς ο χαρακτήρας αποκτά επίπεδα και βελτιώνει τις ικανότητές του.

5.1.2 PlayerStatsManager.cs

Το PlayerStatsManager.cs διαχειρίζεται δυναμικά όλα τα στατιστικά του παίκτη κατά τη διάρκεια του παιχνιδιού. Ανανεώνει την κατάσταση της ζωής, του μάνα και της άμυνας, εφαρμόζει την αναπλήρωση στατιστικών όταν απαιτείται και ελέγχει αν ο παίκτης έχει πεθάνει.

Όταν ο παίκτης αποκτήσει αρκετό XP, το script τον ανεβάζει επίπεδο, αυξάνοντας αυτόματα τα στατιστικά του. Η αναβάθμιση του χαρακτήρα συνοδεύεται από αύξηση της μέγιστης ζωής, της άμυνας και του μάνα, κάνοντας τον παίκτη ισχυρότερο κάθε φορά που προοδεύει.

Το PlayerStatsManager.cs περιλαμβάνει επίσης τη διαχείριση του θανάτου του παίκτη. Όταν η ζωή του μηδενιστεί, ενεργοποιεί το αντίστοιχο animation θανάτου, σταματά την κίνηση του χαρακτήρα και "παγώνει" την κάμερα, ώστε να δώσει την αίσθηση της ήττας.



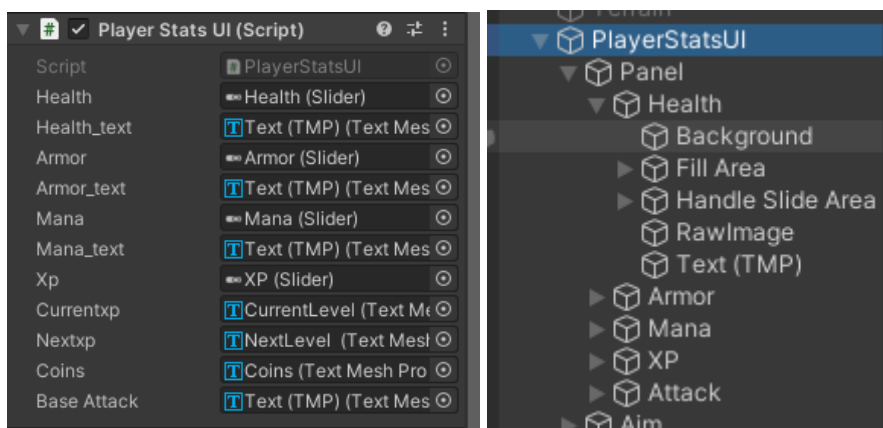
Εικόνα 5.2 Unity Inspector του script

5.1.3 PlayerStatsUI.cs

Το PlayerStatsUI.cs διαχειρίζεται την εμφάνιση των στατιστικών του παίκτη στο γραφικό περιβάλλον χρήστη (UI). Τα δεδομένα που εμφανίζονται περιλαμβάνουν τη ζωή (HP), την άμυνα (Armor), το μάνα (Mana), την εμπειρία (XP), τη ζημιά του παίκτη (Attack Power) και τα νομίσματα (Coins) που διαθέτει.

Η ενημέρωση των στατιστικών γίνεται σε πραγματικό χρόνο, λαμβάνοντας τιμές από το PlayerStatsManager και προσαρμόζοντας τα UI sliders και τα κειμενικά στοιχεία (TextMeshPro) ώστε να απεικονίζουν σωστά τα δεδομένα. Για παράδειγμα, αν ο παίκτης χάσει ζωή, η κόκκινη μπάρα του HP μειώνεται δυναμικά, ενώ αν αποκτήσει XP, η μπλε μπάρα της εμπειρίας αυξάνεται μέχρι να φτάσει στο επόμενο επίπεδο.

Το script αυτό περιλαμβάνει μεθόδους που αντλούν δεδομένα από το PlayerStatsManager, ενημερώνουν τα οπτικά στοιχεία και προσαρμόζουν αριθμητικές τιμές όπως τα τρέχοντα νομίσματα ή την επίθεση του παίκτη.



Εικόνα 5.3 Δομή του PlayerStatsUI στο Unity Inspector, με τα UI στοιχεία για τη ζωή, άμυνα, μάνα, εμπειρία, νομίσματα και ζημιά του παίκτη.

5.1.4 PlayerInteract.cs

Το PlayerInteract.cs δίνει στον παίκτη τη δυνατότητα να αλληλεπιδρά με αντικείμενα και NPCs πατώντας το πλήκτρο "E". Αυτό επιτυγχάνεται με τη χρήση Physics.OverlapSphere(), το οποίο ανιχνεύει αν υπάρχει κάποιο αντικείμενο αλληλεπίδρασης κοντά στον παίκτη.

Αν εντοπιστεί ένα αντικείμενο, όπως ένας έμπορος, τότε καλείται η μέθοδος Interact() του αντικειμένου, ενεργοποιώντας τη λειτουργία του. Για παράδειγμα, αν ο παίκτης βρεθεί μπροστά σε έναν MerchantInteractable, τότε το μενού συναλλαγών θα ανοίξει, επιτρέποντάς του να αγοράσει αναβαθμίσεις ή φίλτρα ζωής.

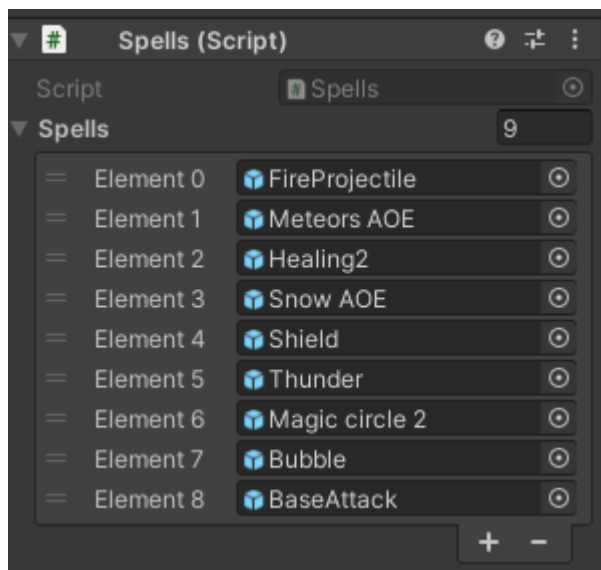
Το script αυτό θα μπορούσε στο μέλλον επίσης να υποστηρίζει άλλους τύπους αλληλεπιδράσεων, όπως το άνοιγμα θησαυρών ή η ενεργοποίηση μηχανισμών μέσα στο παιχνίδι.

5.2 Ικανότητες παίκτη

Τα scripts που αναλύονται παρακάτω ελέγχουν τη χρήση των ξορκιών, την πρόσκρουση των βλημάτων, τις ειδικές ικανότητες και τον τρόπο με τον οποίο αυτά επηρεάζουν τον παίκτη και τους εχθρούς.

5.2.1 Spells.cs (Διαχείριση λίστας ξορκιών)

Το Spells.cs είναι ένα βασικό script που διαχειρίζεται τα διαθέσιμα ξόρκια του παίκτη. Περιέχει μια λίστα από GameObjects που αναπαριστούν τα ξόρκια που μπορεί να χρησιμοποιήσει ο παίκτης. Το script επιτρέπει τη δυναμική φόρτωση και οργάνωση των ξορκιών.



Εικόνα 5.4 Λίστα ξορκιών

5.2.2 Projectile.cs

Το Projectile.cs είναι υπεύθυνο για τη διαχείριση των βλημάτων που εκτοξεύονται από τον παίκτη. Χρησιμοποιεί Rigidbody για να εφαρμόσει κίνηση στο βλήμα με μια καθορισμένη ταχύτητα (projectileSpeed).

Όταν το βλήμα χτυπήσει έναν εχθρό, προκαλεί ζημιά με βάση τη δύναμη επίθεσης του παίκτη, επιτρέποντας την κλιμάκωση της ζημιάς καθώς ο παίκτης εξελίσσεται. Αν το βλήμα συγκρουστεί με οποιοδήποτε άλλο αντικείμενο, καταστρέφεται αυτόματα.

5.2.3 AOESpell.cs (Ξόρκι Περιοχής με Επίδραση στους Εχθρούς)

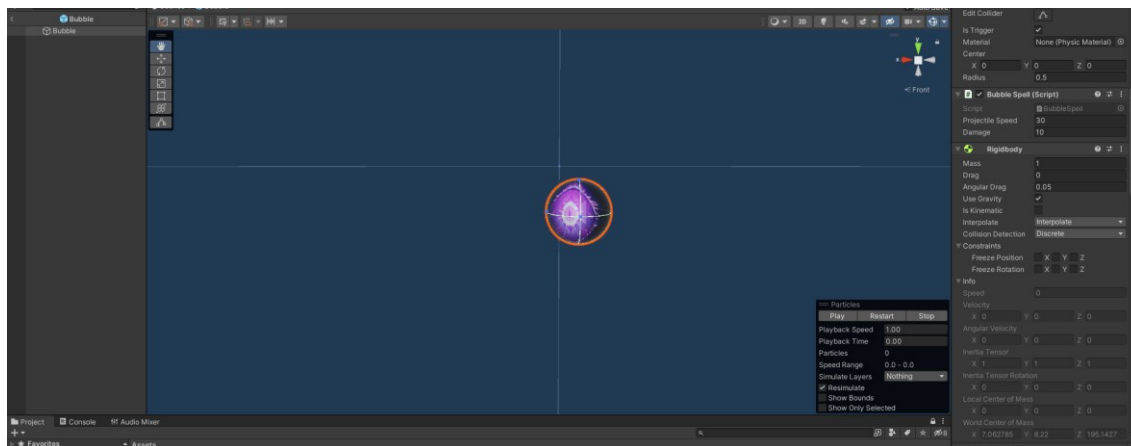
Το AOESpell.cs είναι υπεύθυνο για τη διαχείριση των ξορκιών που επηρεάζουν μια μεγάλη περιοχή. Χρησιμοποιεί Physics.OverlapSphere() για να ανιχνεύσει εχθρούς εντός της περιοχής και να τους επιφέρει ζημιά ή άλλα εφέ.

Όταν το ξόρκι ενεργοποιείται, προσθέτει ζημιά με βάση τη βασική επίθεση του παίκτη και αν το ξόρκι έχει το freeze effect, τότε μειώνει δραματικά την ταχύτητα των εχθρών χρησιμοποιώντας το NavMeshAgent τους. Το ξόρκι έχει συγκεκριμένη διάρκεια ζωής και καταστρέφεται μετά από λίγα δευτερόλεπτα.

5.2.4 BubbleSpell.cs (Ξόρκι Παγίδευσης Εχθρού)

Το BubbleSpell.cs είναι ένα αμυντικό ξόρκι που αιχμαλωτίζει έναν εχθρό μέσα σε μια μαγική φουσαλίδα. Όταν το ξόρκι χτυπά έναν εχθρό, σταματά την κίνησή του, αποτρέπει την επίθεσή του και τον κάνει απροστάτευτο για συγκεκριμένο χρονικό διάστημα.

Το script χρησιμοποιεί Rigidbody constraints για να εμποδίσει την κίνηση του εχθρού και απενεργοποιεί το AI του εχθρού, αναγκάζοντάς τον να παραμείνει ακίνητος μέχρι να ολοκληρωθεί το ξόρκι. Στο τέλος της διάρκειάς του, το spell καταστρέφεται και ο εχθρός επιστρέφει στην αρχική του κατάσταση.



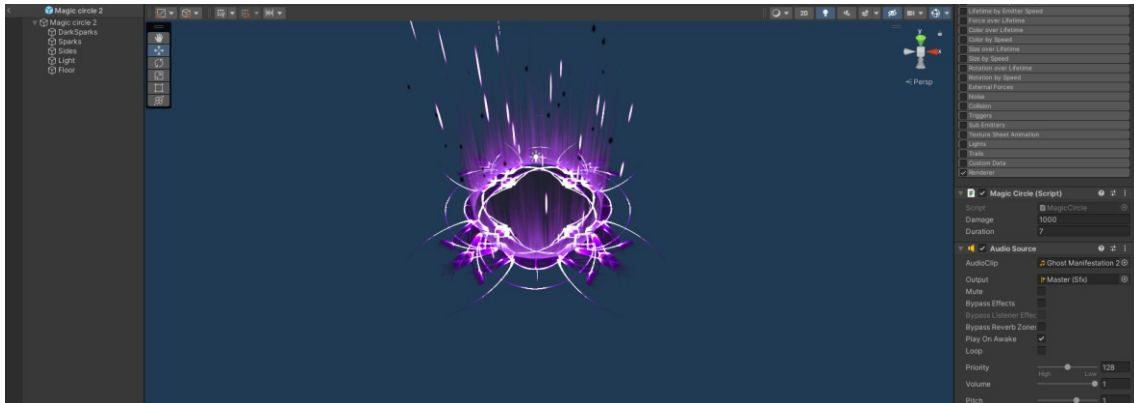
Εικόνα 5.5 Το μοντέλο και οι μεταβλητές για το ξόρκι φουσαλίδας

5.2.5 MagicCircle.cs

Το MagicCircle.cs είναι ένα ισχυρό ξόρκι περιοχής που δημιουργεί ένα μεγάλο κύκλο μαγείας γύρω από τον παίκτη και προκαλεί συνεχή ζημιά στους εχθρούς που βρίσκονται μέσα σε αυτόν.

Η τοποθέτηση του κύκλου γίνεται στο κέντρο του παίκτη, και χρησιμοποιεί Physics.OverlapSphere() για να ανιχνεύσει εχθρούς που βρίσκονται μέσα στη μαγική περιοχή. Όσο παραμένουν εντός της περιοχής, οι εχθροί δέχονται συνεχή ζημιά, η οποία υπολογίζεται με βάση τη δύναμη επίθεσης του παίκτη.

Το ξόρκι έχει περιορισμένη διάρκεια και καταστρέφεται αυτόματα όταν ολοκληρωθεί η επίδρασή του.



Εικόνα 5.6 Το μοντέλο και οι μεταβλητές για το ξόρκι μαγικού κύκλου

5.2.6 Shield.cs (Ξόρκι Ασπίδας)

Το Shield.cs δημιουργεί μια μαγική ασπίδα γύρω από τον παίκτη, η οποία τον προστατεύει από ζημιά για μερικά δευτερόλεπτα. Κατά τη διάρκεια της ενεργοποίησής του, το script αυξάνει την άμυνα (armor) και αναπληρώνει μέρος της ζωής του παίκτη.

Το ξόρκι έχει επίσης μια μοναδική λειτουργία αντανάκλαστικής δύναμης, η οποία απωθεί τους εχθρούς που προσπαθούν να επιτεθούν στον παίκτη. Χρησιμοποιεί Rigidbody και AddForce() για να εφαρμόσει μια δύναμη απώθησης όταν οι εχθροί χτυπούν την ασπίδα.

Η ασπίδα καταστρέφεται αυτόματα μετά από λίγα δευτερόλεπτα, επιστρέφοντας τον παίκτη στην κανονική του κατάσταση.

5.2.7 Potion.cs (Διαχείριση Φίλτρων Ζωής και Μάνα)

Το Potion.cs είναι υπεύθυνο για τη χρήση των φίλτρων ζωής και μάνα από τον παίκτη. Το script ελέγχει αν ο παίκτης έχει διαθέσιμα φίλτρα και αν πατηθεί το κατάλληλο πλήκτρο (πλήκτρο "9" για φίλτρα ζωής και πλήκτρο "0" για φίλτρα μάνα), τότε εφαρμόζεται η αντίστοιχη αποκατάσταση.

Η ζωή και το μάνα αναπληρώνονται ως ποσοστό των μέγιστων τιμών τους. Παράλληλα, το script ενημερώνει δυναμικά το UI, εμφανίζοντας το τρέχον διαθέσιμο απόθεμα φίλτρων.

5.2.8 PlayerEffects.cs (Ξόρκι Θεραπείας – Heal Spell)

Το PlayerEffects.cs είναι υπεύθυνο για τη θεραπεία του παίκτη μέσω του ξορκιού Heal. Όταν ο παίκτης ενεργοποιήσει το ξόρκι, δημιουργείται ένα πεδίο θεραπείας γύρω του, το οποίο αναπληρώνει τη ζωή του για ένα συγκεκριμένο χρονικό διάστημα.

Το script χρησιμοποιεί Physics.OverlapSphere() για να ανιχνεύσει αν ο παίκτης βρίσκεται εντός της περιοχής θεραπείας. Η ποσότητα της ζωής που αποκαθίσταται βασίζεται στη βασική δύναμη επίθεσης του παίκτη, επιτρέποντας έτσι έναν μηχανισμό αναβάθμισης που κάνει το ξόρκι πιο δυνατό όσο ο παίκτης ανεβαίνει επίπεδα.

Το script χρησιμοποιεί ένα **IEnumerator** για να εφαρμόσει τη θεραπεία σταδιακά ανά μικρά χρονικά διαστήματα, προσδίδοντας μια πιο ρεαλιστική αίσθηση αποκατάστασης της ζωής.

5.2.9 SpellCooldown.cs (Διαχείριση Χρόνου Επαναφόρτισης Ξορκιών)

Το SpellCooldown.cs είναι υπεύθυνο για τη διαχείριση του χρόνου επαναφόρτισης (cooldown) των ξορκιών. Κάθε ξόρκι έχει έναν χρονικό περιορισμό πριν χρησιμοποιηθεί ξανά, και το script διαχειρίζεται αυτή τη λειτουργία μέσω της παρακολούθησης του cooldown timer.

Επιπλέον, το script ενημερώνει οπτικά το UI, εμφανίζοντας μια γκρι απόχρωση στο εικονίδιο του ξορκιού και έναν αριθμό που δείχνει πόσος χρόνος απομένει πριν το ξόρκι μπορεί να ξαναχρησιμοποιηθεί.

5.3 Κώδικας Εχθρών

Το σύστημα των εχθρών αποτελεί ένα βασικό στοιχείο του παιχνιδιού, το οποίο περιλαμβάνει λειτουργίες όπως τεχνητή νοημοσύνη (AI), επίθεση από κοντά και από απόσταση, διαχείριση στατιστικών, εμφάνιση UI και αντιμετώπιση ζημιάς. Παρακάτω αναλύονται τα βασικά scripts που καθορίζουν τη συμπεριφορά και τη λειτουργία των εχθρών.

5.3.1 EnemyAI.cs

Το EnemyAI.cs ελέγχει τη συμπεριφορά των εχθρών, συμπεριλαμβανομένων των κινητικών μοτίβων, της επιθετικότητας και της ανίχνευσης του παίκτη. Χρησιμοποιεί το NavMeshAgent του Unity για να μετακινείται μέσα στον κόσμο του παιχνιδιού και να προσεγγίζει τον παίκτη όταν τον εντοπίζει.

Η βασική λογική του script λειτουργεί ως εξής:

- Ο εχθρός περιπολεί την περιοχή (Patrolling) αν ο παίκτης δεν είναι κοντά.
- Όταν ο παίκτης εισέλθει στο πεδίο όρασης του εχθρού, αυτός αλλάζει κατάσταση και τρέχει προς τον παίκτη (Chasing).
- Αν ο παίκτης βρεθεί εντός εμβέλειας επίθεσης, ο εχθρός σταματά και εκτελεί την επίθεσή του (Attacking).

Η κίνηση του εχθρού βασίζεται σε έναν τυχαίο walk point, ο οποίος υπολογίζεται κάθε φορά που ο εχθρός χρειάζεται να περιπολήσει. Ο έλεγχος αν ο παίκτης είναι κοντά γίνεται μέσω των `Physics.CheckSphere()`, τα οποία ανιχνεύουν αν υπάρχει αντικείμενο με tag "Player" εντός της `sightRange` ή της `attackRange`.

Όταν ο εχθρός επιτίθεται, χρησιμοποιεί ένα προκαθορισμένο projectile, το οποίο εκτοξεύεται από ένα συγκεκριμένο spawn point με `Instantiate()`. Η ταχύτητα του projectile ορίζεται μέσω `Rigidbody forces`, επιτρέποντας στον εχθρό να κάνει ranged επιθέσεις.

5.3.2 EnemyProjectile.cs

Το `EnemyProjectile.cs` διαχειρίζεται τα βλήματα των εχθρών, τα οποία χρησιμοποιούνται για επιθέσεις από απόσταση. Στην `Start()`, η ζημιά του projectile προσαρμόζεται με βάση τη δυσκολία και το επίπεδο του παίκτη (`damage *= player.gameDifficultyMultiplier; damage += player.level * 10;`).

Όταν το βλήμα συγκρουστεί με τον παίκτη (`OnTriggerEnter(Collider other)`), αφαιρείται ζωή λαμβάνοντας υπόψη την άμυνα του παίκτη. Εάν η άμυνα δεν είναι αρκετή, το υπόλοιπο της ζημιάς αφαιρείται από τη ζωή.

Το script περιλαμβάνει επίσης μηχανισμό καταστροφής του βλήματος μετά τη σύγκρουση (`Destroy(gameObject);`).

5.3.3 MeleeAttack.cs

Το `MeleeAttack.cs` είναι υπεύθυνο για τις επιθέσεις σώμα με σώμα (melee) των εχθρών. Αντί για βλήματα (projectiles), αυτό το script εφαρμόζει άμεση ζημιά στον παίκτη όταν ο εχθρός έρθει σε επαφή μαζί του.

Η λογική της ζημιάς βασίζεται στο επίπεδο δυσκολίας του παίκτη και την τρέχουσα πρόοδο του στο παιχνίδι. Στην `Start()` αρχικοποίηση, το script υπολογίζει τη ζημιά ως εξής:

- Βασική ζημιά (`damage = 10`)
- Πολλαπλασιαστής δυσκολίας (`damage *= player.gameDifficultyMultiplier`)
- Προσαρμογή βάσει επιπέδου (`damage += player.level * 10`)

Όταν ο εχθρός χτυπήσει τον παίκτη (`OnTriggerEnter(Collider other)`), η ζημιά υπολογίζεται λαμβάνοντας υπόψη την άμυνα του παίκτη (armor). Αν η άμυνα είναι μεγαλύτερη από τη ζημιά, τότε απορροφάται η ζημιά από την άμυνα. Αν η άμυνα δεν είναι αρκετή, το υπόλοιπο της ζημιάς αφαιρείται από τη ζωή του παίκτη (health).

5.3.4 EnemyStats.cs

Το `EnemyStats.cs` είναι ένα απλό `Serializable Class`, το οποίο αποθηκεύει τη ζωή και τη μέγιστη ζωή των εχθρών. Περιλαμβάνει μια βασική μέθοδο `changeHealth(float amount)`, η οποία επιτρέπει τη μεταβολή της ζωής του εχθρού κατά ένα συγκεκριμένο ποσό (είτε μείωση από επιθέσεις είτε αναπλήρωση από buffs).

5.3.5 EnemyStatsUI.cs

Το EnemyStatsUI.cs είναι υπεύθυνο για την εμφάνιση της ζωής των εχθρών στο UI. Κάθε εχθρός έχει μια μπάρα ζωής (Slider) πάνω από το μοντέλο του, η οποία δείχνει την τρέχουσα και τη μέγιστη ζωή του.

Στην Start(), το script αντλεί δεδομένα από το EnemyStatsManager και ενημερώνει τη μπάρα ζωής (health.value = player.stats.health). Στη συνέχεια, κατά τη διάρκεια του παιχνιδιού (Update()), η τιμή της ζωής ανανεώνεται, ώστε να είναι δυναμική και να αντανακλά σωστά την τρέχουσα κατάσταση του εχθρού.

5.3.6 EnemyStatsManager.cs

Το EnemyStatsManager.cs καθορίζει τη ζωή, τη δυσκολία και τη συμπεριφορά των εχθρών. Στην Start(), η μέγιστη ζωή (maxHealth) του εχθρού προσαρμόζεται ανάλογα με το επίπεδο του παίκτη και τον συντελεστή δυσκολίας (gameDifficultyMultiplier).

Το script περιλαμβάνει επίσης λειτουργία για την καταστροφή του εχθρού όταν η ζωή του φτάσει στο μηδέν (if (stats.health <= 0) Destroy(this.gameObject);). Όταν ένας εχθρός πεθάνει, ο παίκτης κερδίζει XP και νομίσματα (player.xp += 15 * player.level; player.coins += 10 * player.level;).

Υπάρχει επιπλέον έλεγχος για την κατάσταση "Bubble", όπου αν ο εχθρός είναι παγιδευμένος μέσα σε ένα BubbleSpell, τότε το animation αλλάζει και ο εχθρός σταματά να κινείται.

5.4 Συστήματα Νίκης, Ήττας και Επιλογών Παιχνιδιού

5.4.1 Victory.cs

Το Victory.cs διαχειρίζεται τη νίκη του παίκτη όταν ολοκληρώσει το τελευταίο επίπεδο του παιχνιδιού. Το script περιλαμβάνει ανίχνευση του animation νίκης, απενεργοποίηση του χειρισμού του παίκτη και εμφάνιση του νικητήριου μενού (Victory Canvas).

Η βασική λογική του script εκτελείται μέσα στη Update(), όπου ελέγχεται αν η τρέχουσα κατάσταση του animation είναι "Open". Όταν αυτό συμβεί:

1. Ενεργοποιείται το Victory Canvas, εμφανίζοντας το μήνυμα της νίκης.
2. Ο παίκτης παίζει το animation "Victory" μέσω του Animator.
3. Απενεργοποιείται ο χειρισμός του χαρακτήρα, ρυθμίζοντας την MoveSpeed και την SprintSpeed σε μηδέν.
4. Απενεργοποιείται ο στόχος (aiming) και η σύγκρουση του παίκτη, μέσω της απενεργοποίησης του CharacterController και του CapsuleCollider.
5. Ξεκλειδώνεται ο κέρσορας του ποντικιού, επιτρέποντας στον παίκτη να αλληλεπιδράσει με το UI.

Αυτό το script εξασφαλίζει ότι όταν ο παίκτης κερδίσει, δεν έχει πλέον έλεγχο στον χαρακτήρα του και μπορεί να επιλέξει να επιστρέψει στο κύριο μενού ή να κλείσει το παιχνίδι.

5.4.2 GameOver.cs

Το GameOver.cs διαχειρίζεται το τι συμβαίνει όταν ο παίκτης πεθάνει. Περιλαμβάνει δύο βασικές επιλογές:

- **Restart()** → Επαναφορτώνει το τρέχον επίπεδο μέσω της `SceneManager.LoadScene(scene.name)`, επιτρέποντας στον παίκτη να ξαναπροσπαθήσει.
- **MainMenu()** → Επιστρέφει τον παίκτη στο κύριο μενού, φορτώνοντας τη σκηνή `SceneManager.LoadScene(0)`.

Αυτό το script συνδέεται με το **Game Over UI**, το οποίο εμφανίζεται όταν ο παίκτης χάσει τη ζωή του.

5.4.3 Difficulty.cs (Διαχείριση Επιπέδου Δυσκολίας)

Το Difficulty.cs επιτρέπει στον παίκτη να επιλέξει τρεις διαφορετικές βαθμίδες δυσκολίας (Easy, Medium, Hard), επηρεάζοντας τον τρόπο με τον οποίο οι εχθροί και οι μηχανισμοί του παιχνιδιού προσαρμόζονται.

Όταν ο παίκτης επιλέξει ένα επίπεδο δυσκολίας, καλείται η αντίστοιχη μέθοδος:

1. **EasyLevel()** → Θέτει τον πολλαπλασιαστή δυσκολίας στο 0.75f.
2. **MediumLevel()** → Θέτει τον πολλαπλασιαστή δυσκολίας στο 1f (βασικό επίπεδο).
3. **HardLevel()** → Αυξάνει τη δυσκολία στο 1.50f, ενισχύοντας τη ζημιά και την ανθεκτικότητα των εχθρών.

Μετά την επιλογή, η μέθοδος `LoadGame()` γράφει τα δεδομένα σε ένα αρχείο "Save-File.txt", το οποίο περιέχει:

- Θέση του παίκτη.
- Στατιστικά ζωής, άμυνας, μάνα, XP.
- Επίπεδο δυσκολίας (`gameDif`).
- Διαθέσιμα φίλτρα και νομίσματα.

Στη συνέχεια, φορτώνεται το επίπεδο του παιχνιδιού μέσω της `SceneManager.LoadScene(1)`. Αυτό το script επιτρέπει δυναμική προσαρμογή της δυσκολίας και διατηρεί τις ρυθμίσεις του παίκτη κατά τη φόρτωση.

5.4.4 HelpScreen.cs

Το HelpScreen.cs διαχειρίζεται την εμφάνιση και απόκρυψη της οθόνης βοήθειας (Help Screen) όταν ο παίκτης πατήσει το πλήκτρο F1.

Όταν το μενού βοήθειας είναι ενεργό:

- Ο παίκτης δεν μπορεί να κινήσει την κάμερα, καθώς το `LockCameraPosition` τίθεται σε `true`.
- Ο κέρσορας του ποντικιού ξεκλειδώνεται, ώστε ο παίκτης να μπορεί να αλληλεπιδράσει με το UI.

Το script περιλαμβάνει επίσης διαχείριση για το πλήκτρο Escape, ώστε αν ο παίκτης βρίσκεται στο help menu, να μπορεί να το κλείσει πατώντας το.

Εάν τόσο το μενού βοήθειας όσο και το κανονικό UI είναι ενεργά ταυτόχρονα, το script διασφαλίζει ότι η οθόνη του παίκτη ξεκλειδώνεται, επιτρέποντας την ελεύθερη κίνηση του ποντικιού.

5.4.5 Load.cs

Το Load.cs επιτρέπει τη φόρτωση αποθηκευμένων δεδομένων από το αρχείο "Save-File.txt", ανακτώντας την τελευταία κατάσταση του παίκτη.

Κατά την εκτέλεση του OnEnable(), το script:

1. Απενεργοποιεί τον χαρακτήρα προσωρινά, ώστε να αποτραπούν bugs κατά τη φόρτωση των θέσεων και των στατιστικών.
2. Διαβάζει τις αποθηκευμένες τιμές από το αρχείο .txt, τις οποίες αποθηκεύει σε μια λίστα (arlist[]).
3. Μετατρέπει τις τιμές σε αριθμητικά δεδομένα (Vector3, float, int) για να αναθέσει:
 - Θέση του παίκτη.
 - Στατιστικά ζωής, άμυνας, μάνα.
 - Επίπεδο δυσκολίας.
 - Νομίσματα και φίλτρα.
4. Επαναφέρει τον παίκτη στη σωστή θέση και ενεργοποιεί τον χαρακτήρα.

Το script χρησιμοποιεί StreamReader για την ανάγνωση του αρχείου και float.Parse() για τη μετατροπή των αριθμητικών δεδομένων.

5.4.6 Options.cs

Το Options.cs επιτρέπει στον παίκτη να ρυθμίσει τον ήχο, την ανάλυση, την ποιότητα των γραφικών και την πλήρη οθόνη (fullscreen).

Διαχείριση Ήχου:

- Τα επίπεδα ήχου αποθηκεύονται σε AudioManager, επιτρέποντας στον παίκτη να ρυθμίσει την ένταση του ήχου και της μουσικής ξεχωριστά.
- Αν ο παίκτης επιλέξει να απενεργοποιήσει τον ήχο, το script μειώνει την ένταση στα -80 dB, ώστε να είναι αθόρυβο.

Ανάλυση και Ποιότητα Γραφικών:

- Το script παίρνει τις διαθέσιμες αναλύσεις της οθόνης και τις εμφανίζει σε ένα dropdown μενού.
- Ο παίκτης μπορεί να επιλέξει μεταξύ διαφορετικών ρυθμίσεων ποιότητας (QualitySettings.SetQualityLevel()), επιτρέποντας αλλαγές στη λεπτομέρεια των γραφικών.

Διαχείριση Παύσης:

- Αν ο παίκτης πατήσει Escape, το παιχνίδι σταματά (`Time.timeScale = 0f`) και εμφανίζεται το Pause Menu UI.
- Ο παίκτης μπορεί να συνεχίσει το παιχνίδι (`Resume()`), το οποίο απενεργοποιεί το Pause Menu και επαναφέρει την ταχύτητα του παιχνιδιού.

5.5 Έμποροι και αναβαθμίσεις

Το σύστημα εμπορίου στο παιχνίδι επιτρέπει στον παίκτη να αγοράζει αναβαθμίσεις και αντικείμενα, βελτιώνοντας τα στατιστικά του και αυξάνοντας τις δυνατότητές του στη μάχη. Μέσω της αλληλεπίδρασης με έναν NPC έμπορο, ο παίκτης μπορεί να ενισχύσει τη ζωή, την άμυνα, το μάνα και τη ζημιά του, καθώς και να προμηθευτεί φίλτρα που θα τον βοηθήσουν κατά τη διάρκεια της περιπέτειας.

5.5.1 MerchantInteractable.cs

Το `MerchantInteractable.cs` είναι το script που διαχειρίζεται την αλληλεπίδραση του παίκτη με τον έμπορο. Όταν ο παίκτης προσεγγίσει τον έμπορο και πατήσει το κατάλληλο κουμπί (E), το script ενεργοποιεί το παράθυρο του καταστήματος (Canvas του shop UI) και απενεργοποιεί προσωρινά την κάμερα και την κίνηση του χαρακτήρα, επιτρέποντας στον παίκτη να επιλέξει τις αναβαθμίσεις του.

Κάθε διαθέσιμο αντικείμενο στη λίστα `items` αποκτά τις σωστές τιμές μέσω της λίστας `prices[]` και `amounts[]`, οι οποίες αντιστοιχούν στο κόστος και το όφελος της αναβάθμισης. Οι τιμές των αντικειμένων εμφανίζονται δυναμικά στο UI, μετατρέποντας τους αριθμούς σε "G" (χρυσά νομίσματα).

Ο κέρσορας του ποντικιού ξεκλειδώνεται, επιτρέποντας στον παίκτη να αλληλεπιδράσει με το μενού (`Cursor.lockState = CursorLockMode.None`). Η κίνηση του παίκτη απενεργοποιείται, ώστε να μην μπορεί να κινηθεί κατά τη διάρκεια της αγοράς (`LockCameraPosition = true`).

Τέλος, το `MerchantInteractable.cs` συνεργάζεται με το `ItemStats.cs`, το οποίο καθορίζει τις αναβαθμίσεις που μπορεί να αγοράσει ο παίκτης και την αντίστοιχη κατανάλωση νομισμάτων.

5.5.2 ItemStats.cs

Το `ItemStats.cs` καθορίζει τις αναβαθμίσεις και τα αντικείμενα που μπορεί να αγοράσει ο παίκτης, διαχειρίζεται τις τιμές τους, τα ποσά που προσθέτουν στα στατιστικά και την κατανάλωση νομισμάτων.

Το script αποθηκεύει τις τιμές (`price`) και τις ποσότητες (`amount`) που προσφέρει κάθε αναβάθμιση, ενώ αντλεί δεδομένα από το `PlayerStatsManager` ώστε να εφαρμόζει σωστά τις αλλαγές στα στατιστικά του παίκτη.

Κατά την εκκίνηση, το script εντοπίζει τον `PlayerStatsManager` και ενημερώνει δυναμικά τις διαθέσιμες αναβαθμίσεις στο UI. Αν ο παίκτης έχει αρκετά νομίσματα για να αγοράσει ένα αντικείμενο, το αντίστοιχο κουμπί αγοράς γίνεται ενεργό

(`buy.interactable = true`), ενώ αν δεν έχει επαρκή ποσότητα χρυσού, το κουμπί απενεργοποιείται (`buy.interactable = false`). Αυτό διασφαλίζει ότι ο παίκτης δεν μπορεί να αγοράσει αντικείμενα για τα οποία δεν έχει αρκετούς πόρους.

Το script διαθέτει διάφορες μεθόδους που εφαρμόζουν αναβαθμίσεις στα στατιστικά του παίκτη. Για παράδειγμα, η `maxHealth()` αυξάνει τη μέγιστη ζωή του παίκτη, η `maxArmor()` ενισχύει την άμυνά του, ενώ η `manaRegen()` αυξάνει τον ρυθμό αναπλήρωσης του μάννα. Επιπλέον, ο παίκτης μπορεί να αγοράσει φίλτρα ζωής (`healthPotion()`) και μάννα (`manaPotion()`), τα οποία προστίθενται στο απόθεμά του και μπορούν να χρησιμοποιηθούν σε κρίσιμες στιγμές.

Η αγορά ενός αντικειμένου αφαιρεί την αντίστοιχη τιμή του από τα διαθέσιμα νομίσματα του παίκτη (`player.coins -= price`) και εφαρμόζει την αναβάθμιση (`player.healthMax += amount`). Σε γενικές γραμμές, το `ItemStats.cs` λειτουργεί σε συνεργασία με το `MerchantInteractable.cs`, ώστε να εμφανίζει σωστά τις αναβαθμίσεις στο κατάστημα, και με το `PlayerStatsManager`, ο οποίος εφαρμόζει τις αλλαγές στα στατιστικά του παίκτη.

5.5.3 CloseShop.cs

Το `CloseShop.cs` χειρίζεται το κλείσιμο του καταστήματος όταν ο παίκτης επιλέξει να επιστρέψει στο παιχνίδι. Πατώντας ένα κουμπί στο UI ή το πλήκτρο ESC, το μενού του καταστήματος απενεργοποιείται, επιτρέποντας στον παίκτη να συνεχίσει την περιπέτειά του.

Όταν ο παίκτης κλείνει το κατάστημα, το παιχνίδι συνεχίζει κανονικά (`Time.timeScale = 1`), η κάμερα του παίκτη ξεκλειδώνεται (`LockCameraPosition = false`) και ο κέρσορας του ποντικιού κλειδώνεται ξανά (`Cursor.lockState = CursorLockMode.Locked`). Αυτό διασφαλίζει ότι ο παίκτης επιστρέφει στο παιχνίδι χωρίς να παραμένει σε λειτουργία εμπορίου.

Το script χρησιμοποιεί το `GameObject.FindWithTag("Player")` για να εντοπίσει τον χαρακτήρα του παίκτη και να επαναφέρει τις σωστές ρυθμίσεις της κάμερας και του χειρισμού, αποτρέποντας τυχόν προβλήματα που θα μπορούσαν να προκύψουν αν το UI του καταστήματος παρέμενε ανοιχτό κατά λάθος.

5.6 Σύστημα Στόχευσης, Checkpoints, Ολοκλήρωσης Επιπέδων και Χειρισμού του Παίκτη

Το παιχνίδι διαθέτει διάφορα συστήματα που βελτιώνουν τη ροή του gameplay, όπως το σύστημα στόχευσης, τα σημεία ελέγχου (checkpoints), την ολοκλήρωση επιπέδων, την αναπαραγωγή εχθρών και τον χειρισμό του παίκτη. Παρακάτω αναλύονται τα αντίστοιχα scripts που διαχειρίζονται αυτές τις λειτουργίες.

5.6.1 StarterAssetsInputs.cs

Το `StarterAssetsInputs.cs` είναι ένα βασικό script που καταγράφει όλες τις ενέργειες του παίκτη, όπως η κίνηση, η στόχευση, το άλμα και η χρήση των ξορκιών. Το script

χρησιμοποιεί το Unity Input System για να αναγνωρίζει τα πλήκτρα που πατά ο παίκτης και να τα μεταφράζει σε ενέργειες μέσα στο παιχνίδι.

Το script περιέχει μεθόδους που καταγράφουν τα εισαγόμενα δεδομένα (InputValue), όπως:

- **MoveInput()** → Διαχειρίζεται την κίνηση του παίκτη.
- **LookInput()** → Ελέγχει την περιστροφή της κάμερας.
- **JumpInput()** → Ανιχνεύει το κουμπί για το άλμα.
- **SprintInput()** → Επιτρέπει στον παίκτη να τρέξει.
- **AimInput()** → Ενεργοποιεί τη στόχευση και καταγράφει το τρέχον ξόρκι που έχει επιλεγεί.

Το script υποστηρίζει διαφορετικά ξόρκια, καταγράφοντας το ποιο ξόρκι έχει επιλεγεί από τον παίκτη μέσω μιας μεταβλητής spellindex. Για παράδειγμα, αν ο παίκτης πατήσει Alpha3, το spell index ορίζεται στο 3, σηματοδοτώντας ότι θα εκτελέσει το αντίστοιχο ξόρκι. Επιπλέον, διαχειρίζεται και την εμφάνιση ή απόκρυψη του ποντικιού όταν ο παίκτης βρίσκεται σε κατάσταση στόχευσης ή σε κάποιο μενού.

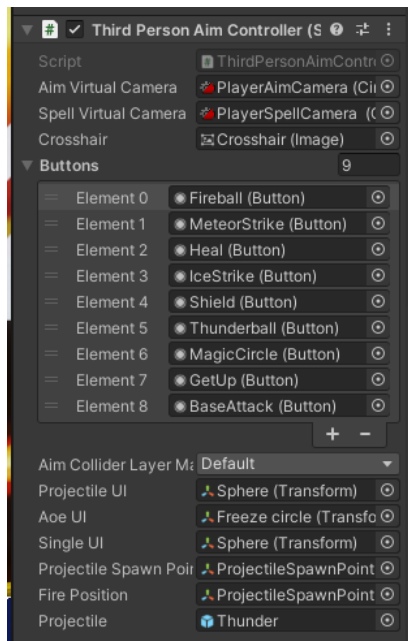
5.6.2 ThirdPersonAimController.cs

Το ThirdPersonAimController.cs διαχειρίζεται το σύστημα στόχευσης του παίκτη, επιτρέποντάς του να σημαδεύει εχθρούς και να ρίχνει ξόρκια από απόσταση. Το script χρησιμοποιεί CinemachineVirtualCamera για να ενεργοποιήσει μια ξεχωριστή κάμερα στόχευσης, η οποία εστιάζει στον στόχο του παίκτη, ενώ παράλληλα εμφανίζει ένα crosshair για να διευκολύνει τη στόχευση.

Κατά την εκκίνηση, το script συνδέεται με τα βασικά συστήματα του παιχνιδιού, όπως το StarterAssetsInputs, το οποίο καταγράφει τις ενέργειες του παίκτη, και το ThirdPersonController, το οποίο ελέγχει την κίνηση του χαρακτήρα. Μέσα στη Update(), το script εντοπίζει τη θέση του ποντικιού και δημιουργεί ένα raycast που ελέγχει αν ο παίκτης σημαδεύει κάποιο σημείο στο περιβάλλον. Αν εντοπιστεί ένα σημείο στόχευσης, μετακινεί το UI του projectile σε αυτό το σημείο, παρέχοντας μια οπτική ένδειξη.

Όταν ο παίκτης πατά το αριστερό κουμπί του ποντικιού (Mouse0), το script ελέγχει αν έχει επιλεγεί κάποιο ξόρκι και ενεργοποιεί την αντίστοιχη επίθεση, είτε ως μονό ξόρκι (single-target spell) είτε ως ξόρκι περιοχής (AOE spell). Στη συνέχεια, αναπαράγει το αντίστοιχο animation επίθεσης, ενώ αν το ξόρκι βρίσκεται σε cooldown, το script εμποδίζει τη χρήση του έως ότου είναι διαθέσιμο.

Το σύστημα στόχευσης είναι επίσης υπεύθυνο για τον περιορισμό της κίνησης του παίκτη όταν ενεργοποιείται κάποιο ισχυρό ξόρκι, όπως το ξόρκι μαγικού κύκλου ή το ξόρκι θεραπείας, ώστε να αποδίδει μια πιο στρατηγική αίσθηση στον χειρισμό των μαγικών ικανοτήτων.



Εικόνα 5.7 Third person Aim Controller

5.6.3 Checkpoint.cs

Το Checkpoint.cs επιτρέπει στον παίκτη να αποθηκεύει την πρόοδό του όταν περνά από προκαθορισμένα σημεία στον χάρτη. Όταν ο παίκτης εισέλθει σε μια περιοχή με ένα Checkpoint, το script αποθηκεύει την τρέχουσα θέση του παίκτη, ενημερώνοντας την ιδιότητα lastCheckpoint στο PlayerStatsManager.

Αυτό το script εξασφαλίζει ότι αν ο παίκτης πεθάνει, θα αναβιώσει στο τελευταίο checkpoint, αντί να ξεκινήσει από την αρχή του επιπέδου. Το σύστημα είναι απλό και λειτουργεί σε συνεργασία με το Spawn.cs, το οποίο διαχειρίζεται την αναπαραγωγή του παίκτη και των εχθρών όταν επιστρέφει στο checkpoint.

5.6.4 CompleteLevel.cs

Το CompleteLevel.cs είναι υπεύθυνο για την ανίχνευση της εξάλειψης των εχθρών σε ένα επίπεδο και το άνοιγμα της εξόδου όταν το επίπεδο ολοκληρωθεί. Όταν ξεκινήσει το παιχνίδι, το script αρχικοποιεί μια λίστα με όλους τους εχθρούς του επιπέδου και ελέγχει διαρκώς αν υπάρχουν ακόμα ενεργοί εχθροί.

Σε κάθε Update(), το script υπολογίζει πόσοι εχθροί παραμένουν στη λίστα enemies. Αν ένας εχθρός έχει πεθάνει (enemy == null), αφαιρείται από τη λίστα. Όταν όλοι οι εχθροί εξαλειφθούν, η μέθοδος OpenDoor() εκτελείται, η οποία αναπαράγει την animation "Open" της εξόδου, επιτρέποντας στον παίκτη να προχωρήσει στο επόμενο επίπεδο.

Αυτό το σύστημα εξασφαλίζει ότι ο παίκτης πρέπει να εξουδετερώσει όλους τους εχθρούς πριν συνεχίσει, διατηρώντας την ισορροπία στο gameplay.

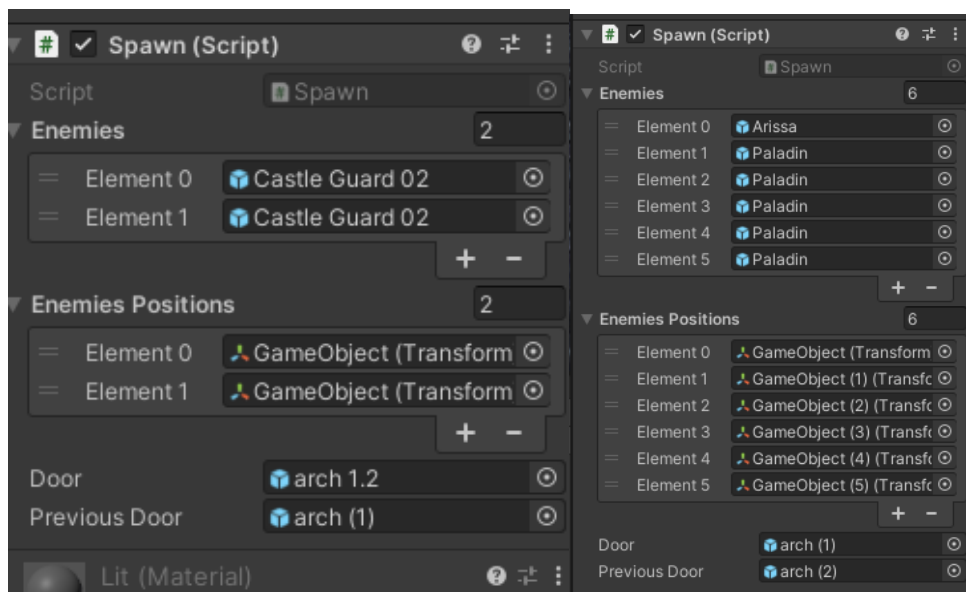
5.6.5 Spawn.cs

Το Spawn.cs είναι ένα πιο πολύπλοκο script, καθώς είναι υπεύθυνο για την αναπαραγωγή των εχθρών, το κλείσιμο και το άνοιγμα θυρών καθώς και την αποθήκευση της προόδου του παίκτη.

Όταν ο παίκτης περάσει από μια συγκεκριμένη περιοχή (OnTriggerEnter(Collider other)), το script εκτελεί τα εξής:

1. Κλείνει την προηγούμενη πόρτα, αποτρέποντας την επιστροφή στο προηγούμενο σημείο.
2. Αποθηκεύει το τελευταίο checkpoint του παίκτη, ώστε να μπορεί να επανέλθει εκεί σε περίπτωση θανάτου.
3. Δημιουργεί νέους εχθρούς, τοποθετώντας τους στις προκαθορισμένες θέσεις (Instantiate(enemies[i], enemiesPositions[i].position, enemiesPositions[i].rotation)).
4. Αποθηκεύει την πρόοδο του παίκτη σε ένα αρχείο "Save-File.txt", διασφαλίζοντας ότι τα στατιστικά του διατηρούνται σωστά.

Με αυτό το σύστημα, κάθε περιοχή του παιχνιδιού λειτουργεί ανεξάρτητα, με τους εχθρούς να αναπαράγονται μόνο όταν ο παίκτης προχωρήσει, ενώ παράλληλα καταγράφεται η πρόοδός του.



Εικόνα 5.8 Εχθροί και οι συντεταγμένες στις οποίες θα εμφανιστούν

5.7 Λοιπά scripts

Τα δύο τελευταία scripts είναι υπεύθυνα για τη διαχείριση της μουσικής ανάλογα με την κατάσταση του παιχνιδιού και την προσαρμογή συγκεκριμένων αντικειμένων στο οπτικό πεδίο της κάμερας.

5.7.1 Music.cs

Το Music.cs είναι ένα script που διαχειρίζεται τη μουσική υπόκρουση του παιχνιδιού, προσαρμόζοντάς τη δυναμικά ανάλογα με την κατάσταση του παίκτη και των εχθρών. Χρησιμοποιεί τρεις διαφορετικές ηχητικές πηγές (**AudioSource**) για να εναλλάσσει μεταξύ της χαλαρής μουσικής υπόκρουσης, της μουσικής μάχης και της μουσικής θανάτου.

Κατά την εκκίνηση, το script αποθηκεύει τρεις ηχητικές πηγές:

- **background:** Η βασική μουσική του παιχνιδιού που παίζει όταν ο παίκτης εξερευνά τον κόσμο.
- **battle:** Η μουσική μάχης που ενεργοποιείται όταν ο παίκτης βρίσκεται κοντά σε εχθρούς.
- **death:** Η μουσική που παίζει όταν ο παίκτης πεθαίνει.

Το Music.cs ελέγχει συνεχώς την κατάσταση του παιχνιδιού μέσω της μεθόδου `Update()`. Σε κάθε frame, εκτελείται μια αναζήτηση, η οποία ελέγχει αν υπάρχουν εχθροί στη σκηνή. Αν υπάρχουν ενεργοί εχθροί, το script απενεργοποιεί τη βασική μουσική (`background.enabled = false`) και ενεργοποιεί τη μουσική μάχης (`battle.enabled = true`). Αν δεν υπάρχουν εχθροί, το αντίθετο συμβαίνει, και η μουσική επιστρέφει στη βασική υπόκρουση.

Το script περιλαμβάνει επίσης έλεγχο για τον θάνατο του παίκτη. Αν η ζωή του παίκτη φτάσει στο μηδέν (`playerStats.health <= 0`), τότε και οι δύο προηγούμενες μουσικές σταματούν (`background.enabled = false`, `battle.enabled = false`), και η μουσική θανάτου ενεργοποιείται (`death.enabled = true`).

5.7.2 Billboard.cs

Το Billboard.cs είναι ένα script που χρησιμοποιείται για να διασφαλίσει ότι συγκεκριμένα αντικείμενα, όπως οι μπάρες ζωής, τα floating UI elements και οι ενδείξεις εχθρών, είναι πάντα στραμμένα προς την κάμερα. Αυτό είναι σημαντικό για την ορατότητα των στοιχείων του UI, καθώς σε ένα 3D περιβάλλον, τα επίπεδα UI αντικείμενα μπορεί να εμφανίζονται σε λάθος γωνίες αν δεν περιστρέφονται δυναμικά προς την κάμερα.

Κατά την εκκίνηση (`Awake()`), το script εντοπίζει την κύρια κάμερα του παιχνιδιού (`GameObject.FindGameObjectWithTag("MainCamera").transform`) και αποθηκεύει τη μεταβλητή της (`cam`). Στη συνέχεια, σε κάθε frame (`LateUpdate()`), το script ρυθμίζει τη θέση του αντικειμένου ώστε να κοιτάζει πάντα προς την ίδια κατεύθυνση με την κάμερα, χρησιμοποιώντας την `LookAt()`.

Με αυτόν τον τρόπο, τα στοιχεία που χρησιμοποιούν το Billboard.cs γίνονται πιο ευδιάκριτα και ευανάγνωστα, βελτιώνοντας την εμπειρία του παίκτη.

6 Συμπεράσματα

Κατά τη διαδικασία ανάπτυξης, δόθηκε έμφαση στην εξερεύνηση, τη στρατηγική μάχη και τη συνεχή αναβάθμιση των ικανοτήτων του παίκτη, δημιουργώντας έτσι μια διαδραστική εμπειρία που κρατά το ενδιαφέρον του χρήστη. Η εισαγωγή μηχανισμών όπως τα checkpoints, οι έμποροι, το σύστημα εμπειρίας (XP) και η τεχνητή νοημοσύνη (AI) των εχθρών συνέβαλε σημαντικά στην ολοκληρωμένη λειτουργικότητα του παιχνιδιού.

Ωστόσο, υπήρξαν ορισμένες προκλήσεις και περιορισμοί που αντιμετωπίστηκαν κατά τη διάρκεια της ανάπτυξης:

- **Βελτιστοποίηση της απόδοσης:** Ορισμένα επίπεδα περιείχαν μεγάλο αριθμό αντικειμένων και εχθρών, γεγονός που επηρέαζε την απόδοση. Αυτό αντιμετωπίστηκε χρησιμοποιώντας γραφικά με λιγότερη λεπτομέρεια και λιγότερα αντικείμενα.
- **Ισορροπία δυσκολίας:** Αρχικά, το παιχνίδι ήταν είτε πολύ δύσκολο είτε πολύ εύκολο σε ορισμένα σημεία. Μέσω δοκιμών και αναπροσαρμογών, επιτεύχθηκε μια πιο ομαλή καμπύλη δυσκολίας.
- **Περιορισμοί στα animations:** Τα animations των χαρακτήρων προέρχονται κυρίως από έτοιμα assets και όχι από custom δημιουργία, κάτι που περιορίσει την ποικιλία των κινήσεων.

Η εργασία αυτή ανοίγει τον δρόμο για μελλοντική βελτίωση και εξέλιξη του παιχνιδιού, προσφέροντας δυνατότητες για:

- **Προσθήκη περισσότερων επιπέδων και εχθρών** με μοναδικά μοτίβα συμπεριφοράς.
- **Δημιουργία ενός πιο προηγμένου AI συστήματος** για εχθρούς με adaptive AI που αλλάζει στρατηγική κατά τη διάρκεια της μάχης.
- **Υλοποίηση multiplayer mode**, όπου οι παίκτες μπορούν να συνεργάζονται ή να μάχονται μεταξύ τους.
- **Ενίσχυση της ιστορίας του παιχνιδιού** με περισσότερα NPCs, διαλόγους και αποστολές.
- **Βελτιωμένα animations και οπτικά εφέ** για μεγαλύτερη ρεαλιστικότητα.

Η εξέλιξη του **Dungeon Escape** μπορεί να οδηγήσει στη δημιουργία ενός **πιο πλούσιου και πολυδιάστατου RPG**, το οποίο θα επεκτείνει την εμπειρία του παίκτη και θα προσφέρει μεγαλύτερη ποικιλία gameplay.

Η ανάπτυξη ενός τρίτου προσώπου RPG παιχνιδιού μέσω της Unity υπήρξε μια απαιτητική, αλλά και δημιουργική διαδικασία. Μέσα από ανάλυση, σχεδιασμό, ανάπτυξη και δοκιμές, το παιχνίδι εξελίχθηκε σε ένα λειτουργικό πρωτότυπο που συνδυάζει εξερεύνηση, μάχη, προσαρμογή χαρακτήρα και προκλήσεις.

Η πτυχιακή αυτή εργασία αποδεικνύει τη δυνατότητα ανάπτυξης ολοκληρωμένων παιχνιδιών από ένα άτομο, αξιοποιώντας σύγχρονα εργαλεία και τεχνολογίες. Μελλοντικές βελτιώσεις μπορούν να επεκτείνουν το project, καθιστώντας το ακόμα πιο ενδιαφέρον και ανταγωνιστικό στον χώρο των ανεξάρτητων (indie) παιχνιδιών.

7 Πίνακας συντμήσεων-αρκτικόλεξων-ακρωνυμίων

RPG	Role Playing Game
NPC	Non-Player Character
HP	Health Points
EXP or XP	Experience Points
AOE	Area of effect
dB	Decibel
UI	User Interface
AI	Artificial Intelligence
3D	Three-Dimensional
FPS	First Person Shooter

8 Πίνακας ορολογίας

Canvas	Καμβάς
Asset	Ψηφιακό αντικείμενο (Μοντέλο, Ήχος, Κώδικας)
Checkpoint	Σημείο Ελέγχου
Projectile	Οποιοδήποτε αντικείμενο που εκτοξεύεται στο παιχνίδι και ακολουθεί συγκεκριμένη τροχιά, όπως βέλη, σφαίρες, μαγικές σφαίρες ή ενεργειακά κύματα.
Non-Playable Character	Μη Παίξιμος Χαρακτήρας
Multiplayer	Πολλαπλοί Παίκτες
Animation	Κινούμενη ακολουθία / Κίνηση χαρακτήρων και αντικειμένων

9 Βιβλιογραφία

- Starter Assets - ThirdPerson:
<https://assetstore.unity.com/packages/essentials/starter-assets-thirdperson-updates-in-new-charactercontroller-pa-196526>
- Mixamo: <https://www.mixamo.com/>
- Γραφικά μπουντρουμιού το Asset Store της Unity:
 - ο <https://assetstore.unity.com/packages/3d/environments/dungeons/modular-castle-121360>
 - ο <https://assetstore.unity.com/packages/3d/environments/dungeons/blue-dungeon-106912>
 - ο <https://assetstore.unity.com/packages/3d/environments/dungeons/the-red-prison-40198>
- Βίντεο που χρησιμοποιήθηκε για την ανάπτυξη του Third Person Controller: <https://www.youtube.com/watch?v=FbM4CkqtOuA>
- UI Pack: <https://assetstore.unity.com/packages/2d/gui/icons/free-ui-pack-170878>
- Εικονίδια για τα ξόρκια και στατιστικά :
<https://assetstore.unity.com/packages/2d/gui/icons/free-rpg-fantasy-spell-icons-200511>
- Βίντεο εφέ για την εκτέλεση ξορκιών:
<https://assetstore.unity.com/packages/vfx/particles/spells/magic-effects-free-247933>
- Γραφικά για τον ουρανό :
<https://assetstore.unity.com/packages/2d/textures-materials/sky/allsky-free-10-sky-skybox-set-146014>