



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ**  
**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ**  
**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

**Πτυχιακή Εργασία**

Τίτλος Πτυχιακής Εργασίας	Η σύλληψη ενός παιχνιδιού και η αναπτυξή του σε περιβάλλον Unity, μία Τεχνική Ανάλυση. The conception of a game and its development in a Unity environment, a Technical Analysis.
Όνοματεπώνυμο Φοιτητή	Στέφανος Μόρτογλου
Πατρώνυμο	Χρήστος
Αριθμός Μητρώου	Π/ 16083
Επιβλέπων	Ευθύμιος Αλέπης, Καθηγητής

Ημερομηνία Παράδοσης Σεπτέμβριος 2024

Η σύλληψη ενός παιχνιδιού και η αναπτυξή του σε περιβάλλον Unity, μία Τεχνική Ανάλυση.

### **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

## Περιεχόμενα

Copyright © .....	2
<b>Περιεχόμενα</b> .....	<b>3</b>
<b>1.Περίληψη</b> .....	<b>5</b>
<b>2.Εισαγωγή</b> .....	<b>6</b>
<b>3.Ανασκόπηση πεδίου (Research)</b> .....	<b>8</b>
3.1 Hades .....	8
3.2 Dead Cells .....	8
3.3 Slay the Spire .....	9
3.4 Rogue Legacy .....	9
3.5 Darkest Dungeon .....	9
3.6 Risk of Rain 2 .....	10
3.7 Enter the Gungeon .....	10
3.8 Okhlos (Ελληνικό παράδειγμα) .....	11
<b>4.Αρχιτεκτονική Συστήματος</b> .....	<b>13</b>
4.1 GameManager .....	13
4.2 PlayerClass .....	14
4.3 PlayerMovement .....	16
4.4 PlayerCombat .....	17
4.5 WeaponManager .....	18
4.6 Weapon (Melee) .....	19
<b>5.Συμπεράσματα</b> .....	<b>20</b>
Μελλοντικές Επεκτάσεις .....	20
<b>6.Βιβλιογραφία</b> .....	<b>23</b>
<b>7. Πίνακες και Παραρτήματα</b> .....	<b>26</b>
7.1 Διαγράμματα UML .....	26

<b>7.2 User's Manual (w/ Images).....</b>	<b>29</b>
<b>7.3 GAMEPLAY SCREENSHOTS.....</b>	<b>30</b>
<b>7.4 ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ .....</b>	<b>35</b>
<b>7.5 ΠΙΝΑΚΑΣ ΣΥΝΤΜΗΣΕΙΣ - ΑΡΚΤΙΚΟΛΕΞΑ - ΑΚΡΩΝΥΜΙΑ.....</b>	<b>35</b>

## 1. Περίληψη

Το παρόν έργο αποτελεί μια τεχνική ανάλυση ενός hack n slash και wave-based rogue-like παιχνιδιού, εμπνευσμένου από το Hades, το οποίο αναπτύχθηκε αποκλειστικά στην πλατφόρμα Unity. Το παιχνίδι επικεντρώνεται στις γρήγορες, δυναμικές μάχες και την έντονη αλληλεπίδραση με εχθρούς που εμφανίζονται σε κλιμακούμενα κύματα. Στόχος του παίκτη είναι να επιβιώσει από διαδοχικά κύματα εχθρών, ενώ παράλληλα αναβαθμίζει τις ικανότητές του μέσα από ένα σύστημα που του επιτρέπει να επιλέγει νέες δεξιότητες και όπλα σε κάθε «run». Αυτό το στοιχείο προσδίδει υψηλή επαναληψιμότητα (replayability), καθώς κάθε νέα προσπάθεια του παίκτη είναι διαφορετική από την προηγούμενη, με τυχαία γεννημένες προκλήσεις και εχθρούς.

Ο μηχανισμός των εχθρικών κυμάτων είναι κεντρικός στον σχεδιασμό του παιχνιδιού, με κάθε κύμα να προσαρμόζεται δυναμικά στις επιδόσεις του παίκτη. Αυτό επιτρέπει την πρόοδο της δυσκολίας σε ομαλή κλίμακα, προσφέροντας συνεχή πρόκληση χωρίς να γίνεται άδικο ή υπερβολικά δύσκολο. Παράλληλα, οι εχθροί διαφέρουν σε στρατηγικές και ικανότητες, αναγκάζοντας τον παίκτη να προσαρμόζει τις τακτικές του συνεχώς. Οι τύποι εχθρών και οι ιδιότητες των κυμάτων τους δημιουργούν ένα περιβάλλον γεμάτο ποικιλία, δίνοντας έμφαση στη στρατηγική και την προσαρμοστικότητα.

Η ανάπτυξη του παιχνιδιού έγινε με τη χρήση της γλώσσας C# και των εργαλείων του Unity. Ο κώδικας που διαχειρίζεται τα κύματα εχθρών και τη συμπεριφορά του παίκτη είναι οργανωμένος σε modular συστήματα, επιτρέποντας εύκολη προσαρμογή και επέκταση των δυνατοτήτων του παιχνιδιού. Οι μηχανισμοί των μαχών εστιάζουν στη γρήγορη απόκριση (responsiveness) των εντολών του παίκτη και στην ακριβή εκτέλεση των επιθέσεων και αμυνών. Κάθε σύστημα μάχης έχει σχεδιαστεί ώστε να προσφέρει ισορροπία ανάμεσα σε ταχύτητα και στρατηγική, επιτρέποντας στον παίκτη να αντιδρά άμεσα σε εχθρικές κινήσεις.

Το παιχνίδι χρησιμοποιεί third-party μοντέλα και assets για τα γραφικά και τα περιβάλλοντα, με τις υπόλοιπες πτυχές να έχουν αναπτυχθεί από την ομάδα του project. Τα γραφικά προσφέρουν μια οπτική εμπειρία που ενισχύει την αίσθηση δράσης, ενώ τα περιβάλλοντα δημιουργούν μια πλούσια ατμόσφαιρα. Η τεχνητή νοημοσύνη (AI) των εχθρών έχει σχεδιαστεί ώστε να προσαρμόζεται στις ενέργειες του παίκτη, προσθέτοντας ένα επιπλέον επίπεδο στρατηγικής στη μάχη. Η συνεχής εξέλιξη των ικανοτήτων του παίκτη, μέσω αναβαθμίσεων και επιλογών, καθιστά το παιχνίδι εθιστικό και επιτρέπει στον παίκτη να αισθάνεται πως βελτιώνεται σε κάθε run.

Μελλοντικά, το παιχνίδι μπορεί να επεκταθεί με multiplayer επιλογές, προσφέροντας τη δυνατότητα co-op και άλλων modes, αυξάνοντας τη στρατηγική και κοινωνική διάσταση του παιχνιδιού. Οι επεκτάσεις αυτές θα μπορούσαν να περιλαμβάνουν νέους εχθρούς, πίστες και αντικείμενα, προσφέροντας μεγαλύτερη ποικιλία και βάθος στην εμπειρία του παίκτη. Τα πρόσθετα χαρακτηριστικά θα μπορούσαν επίσης να περιλαμβάνουν νέα modes, όπως survival και time attack, αυξάνοντας περαιτέρω την πρόκληση και τη διασκέδαση του παιχνιδιού.

## 2.Εισαγωγή

Το παιχνίδι που αναπτύχθηκε αποτελεί έναν hack n slash τίτλο με wave-based rogue-like στοιχεία, εμπνευσμένο από τον τίτλο Hades. Η βασική ιδέα πίσω από την ανάπτυξη ήταν να δημιουργηθεί ένα παιχνίδι το οποίο θα συνδυάζει δυναμική μάχη, γρήγορη ανταπόκριση στις εντολές του παίκτη και αυξανόμενη δυσκολία μέσω κυμάτων εχθρών. Σε αντίθεση με παραδοσιακά RPG παιχνίδια, το rogue-like μοντέλο του παιχνιδιού προσφέρει έναν πιο ευέλικτο και συνεχώς εξελισσόμενο τρόπο παιχνιδιού, όπου κάθε απόπειρα (run) του παίκτη είναι διαφορετική. Το σύστημα αυτό διατηρεί το ενδιαφέρον και δίνει έμφαση στην ικανότητα προσαρμογής και στρατηγικής από τον παίκτη.

Η δομή του παιχνιδιού βασίζεται σε διαδοχικά κύματα εχθρών, τα οποία προσαρμόζονται ανάλογα με τις επιδόσεις του παίκτη. Με κάθε κύμα, η δυσκολία αυξάνεται, αναγκάζοντας τον παίκτη να βελτιώνει συνεχώς τις δεξιότητές του και να προσαρμόζει τη στρατηγική του. Ο μηχανισμός των upgrades και των επιλογών που δίνει στον παίκτη το σύστημα αναβάθμισης (όπλα, ικανότητες) παίζει καθοριστικό ρόλο στην πρόοδο και τη στρατηγική του παιχνιδιού, προσφέροντας μια διαρκή αίσθηση προόδου και επιτυχίας.

Το παιχνίδι σχεδιάστηκε να παρέχει μια εμπειρία όπου ο παίκτης μπορεί να μπαίνει σε γρήγορες μάχες με εχθρούς, να επιλέγει τα εργαλεία και τις ικανότητες του χαρακτήρα του, και να προσαρμόζει την προσέγγισή του ανάλογα με τις συνθήκες. Η δυνατότητα αναβάθμισης και διαφοροποίησης των εργαλείων μάχης ενισχύει το στοιχείο της επαναληψιμότητας (replayability), δίνοντας τη δυνατότητα στον παίκτη να βελτιώνει τις δεξιότητές του και να επιστρέφει στο παιχνίδι με νέες στρατηγικές.

Η έμπνευση για τον σχεδιασμό του παιχνιδιού αντλήθηκε κυρίως από παιχνίδια όπως το Hades, το οποίο κατάφερε να παντρέψει αρμονικά τη δράση με την αφήγηση, προσφέροντας μια ολοκληρωμένη εμπειρία στον παίκτη. Παρόμοιες επιρροές εντοπίζονται και στο Dead Cells, το οποίο προσφέρει ταχύτητα στις μάχες και προσαρμογή της στρατηγικής σε πραγματικό χρόνο. Ωστόσο, το παιχνίδι μας διαφοροποιείται ενσωματώνοντας μοναδικά στοιχεία που προσαρμόζονται στις επιδόσεις του παίκτη, προσφέροντας μια πιο εξατομικευμένη εμπειρία.

Η ανάπτυξη του παιχνιδιού έγινε αποκλειστικά με την χρήση της πλατφόρμας Unity, η οποία προσφέρει ευελιξία στην προσαρμογή των μηχανισμών του παιχνιδιού και των λειτουργιών του. Η χρήση της γλώσσας C# για την ανάπτυξη των διαφόρων συστημάτων μάχης, των εχθρών, και των αναβαθμίσεων επέτρεψε την εύκολη διαχείριση και προσαρμογή του gameplay, δημιουργώντας ένα σύστημα που μπορεί να εξελίσσεται και να προσαρμόζεται με την πάροδο του χρόνου.

Ένας από τους κύριους στόχους του project ήταν η επίτευξη μιας ισορροπίας ανάμεσα στην ταχύτητα των μαχών και τη στρατηγική βάθους. Αυτό επιτυγχάνεται με την προσεκτική σχεδίαση των συστημάτων εχθρών, όπου κάθε εχθρός έχει μοναδικές ικανότητες και συμπεριφορές, αναγκάζοντας τον παίκτη να προσαρμόζει τις τακτικές του ανάλογα με τη σύνθεση των κυμάτων.

Επιπλέον, οι μηχανισμοί εχθρών σχεδιάστηκαν ώστε να παρέχουν ποικιλία και πρόκληση σε κάθε run, με κάθε κύμα να αυξάνει την ένταση και την πολυπλοκότητα. Αυτό το συνεχώς μεταβαλλόμενο περιβάλλον αναγκάζει τον παίκτη να χρησιμοποιεί στρατηγική και γρήγορα αντανακλαστικά για να επιβιώσει, κάτι που κάνει κάθε run μοναδικό και διαρκώς συναρπαστικό.

Ένας άλλος παράγοντας που παίζει κεντρικό ρόλο στην εμπειρία του παίκτη είναι η ισορροπία ανάμεσα στις αναβαθμίσεις και στις προκλήσεις που προσφέρονται. Οι παίκτες καλούνται να επιλέξουν

προσεκτικά τις αναβαθμίσεις τους, καθώς αυτές μπορούν να καθορίσουν την έκβαση της μάχης. Παράλληλα, η εξέλιξη του παίκτη γίνεται με τέτοιο τρόπο ώστε κάθε νέο κύμα να απαιτεί προσαρμογή των τακτικών και των επιλογών του.

Ένα σημαντικό χαρακτηριστικό που ενσωματώθηκε είναι η δυνατότητα αναβάθμισης των όπλων και των ικανοτήτων σε κάθε run. Αυτό δίνει στον παίκτη τη δυνατότητα να πειραματιστεί με διαφορετικούς συνδυασμούς, αυξάνοντας την ποικιλία και τη στρατηγική που μπορεί να εφαρμόσει σε κάθε run. Αυτές οι επιλογές αναβαθμίσεων προσφέρουν μια συνεχώς εξελισσόμενη εμπειρία που διατηρεί το ενδιαφέρον του παίκτη.

Η συνεργασία της ομάδας ανάπτυξης με τη χρήση third-party assets για τα γραφικά, σε συνδυασμό με τα custom μοντέλα και εφέ που δημιουργήθηκαν, συνεισέφερε στην οπτική και ατμοσφαιρική εμπειρία του παιχνιδιού. Τα περιβάλλοντα και οι εχθροί σχεδιάστηκαν ώστε να προσφέρουν μια δυναμική αίσθηση, με εναλλαγές στο περιβάλλον ανάλογα με την πρόοδο του παίκτη.

Η ανάπτυξη ενός συστήματος αναβάθμισης χαρακτήρα έπαιξε σημαντικό ρόλο στην εξέλιξη της στρατηγικής των παικτών. Ο παίκτης έχει τη δυνατότητα να επιλέξει από πληθώρα βελτιώσεων που αφορούν τα στατιστικά του χαρακτήρα, όπως τη δύναμη των επιθέσεων, την ανθεκτικότητα, και τις ταχύτητες ανανέωσης των ικανοτήτων. Αυτό προσθέτει επιπλέον επίπεδα βάθους στη στρατηγική, καθώς κάθε απόφαση που λαμβάνει ο παίκτης επηρεάζει τον τρόπο που παίζεται το παιχνίδι στα επόμενα κύματα εχθρών.

Η δημιουργία του συστήματος μάχης στο παιχνίδι απαιτούσε λεπτομερή σχεδιασμό τόσο για τους εχθρούς όσο και για τις δυνατότητες του παίκτη. Για να διατηρηθεί η ισορροπία ανάμεσα στην πρόκληση και την απόλαυση, ενσωματώθηκαν διάφοροι τύποι επιθέσεων, από γρήγορες επιθέσεις με χαμηλή ισχύ μέχρι βραδύτερες αλλά πιο ισχυρές επιθέσεις. Οι εχθροί επίσης ποικίλουν στις στρατηγικές τους, με μερικούς να επικεντρώνονται σε άμεσες επιθέσεις, ενώ άλλοι χρησιμοποιούν στρατηγικές αποφυγής ή επίθεσης από απόσταση. Αυτή η ποικιλία διασφαλίζει ότι οι παίκτες πρέπει να προσαρμόζουν τις τακτικές τους σε κάθε run, διατηρώντας το ενδιαφέρον σε υψηλά επίπεδα.

Η ισορροπία μεταξύ της ταχύτητας των μαχών και του βάθους της στρατηγικής ήταν ένας από τους μεγαλύτερους στόχους της ανάπτυξης. Η προσαρμοστικότητα και η δυνατότητα πειραματισμού με διαφορετικά όπλα, ικανότητες και στρατηγικές προσδίδουν μεγαλύτερη εμπάθунση στο gameplay. Το αποτέλεσμα είναι ένα παιχνίδι που επιτρέπει στους παίκτες να αισθάνονται ότι εξελίσσονται συνεχώς και να ανακαλύπτουν νέες προσεγγίσεις σε κάθε run, προσφέροντας μια πλούσια και ανταποδοτική εμπειρία.

Συνολικά, η ανάπτυξη του παιχνιδιού επικεντρώθηκε στη δημιουργία μιας εμπειρίας όπου ο παίκτης καλείται να εξερευνήσει διαφορετικές τακτικές και στρατηγικές σε κάθε run. Η ικανότητα του παίκτη να προσαρμόζεται και να βελτιώνεται μέσα από τη διαδικασία του παιχνιδιού είναι το κεντρικό σημείο που κάνει το παιχνίδι μοναδικό. Ο στόχος του έργου ήταν να δημιουργηθεί μια εμπειρία που προσφέρει διαρκή πρόκληση, στρατηγικό βάθος, και ελευθερία στις επιλογές του παίκτη, κάτι που επιτεύχθηκε με τη συνειδητή προσπάθεια της ομάδας ανάπτυξης να δημιουργήσει ένα σύστημα που εξελίσσεται με κάθε run και ενσωματώνει μοναδικά χαρακτηριστικά.

### **3. Ανασκόπηση πεδίου (Research)**

Η κατηγορία των rogue-like και hack n slash παιχνιδιών έχει γνωρίσει σημαντική ανάπτυξη τα τελευταία χρόνια, καθώς όλο και περισσότεροι τίτλοι καταφέρνουν να συνδυάσουν την έντονη δράση με δυναμικά συστήματα προόδου και εξερεύνησης. Το παιχνίδι μας αντλεί στοιχεία και έμπνευση από διάφορους δημοφιλείς τίτλους, μερικοί εκ των οποίων έχουν θέσει τα θεμέλια για τον σχεδιασμό αυτού του είδους. Τίτλοι όπως το Hades, το Dead Cells, και το Slay the Spire έφεραν νέες δυναμικές στο gameplay τους, εισάγοντας καινοτομίες στη μάχη, την αναβάθμιση χαρακτήρων και τις προκλήσεις που αντιμετωπίζουν οι παίκτες.

#### **3.1 Hades**

Το Hades της Supergiant Games είναι ένα από τα πιο αναγνωρισμένα rogue-like παιχνίδια, καταφέροντας να παντρέψει τη γρήγορη και απολαυστική μάχη με μια βαθιά αφήγηση. Η μηχανική του παιχνιδιού είναι απλή: ο παίκτης προσπαθεί να αποδράσει από τον Κάτω Κόσμο, περνώντας μέσα από διαδοχικές ζώνες γεμάτες εχθρούς. Ωστόσο, αυτό που διαφοροποιεί το Hades από άλλα παιχνίδια του είδους είναι η έμφαση που δίνει στην αφήγηση και την εξέλιξη της ιστορίας, ακόμη και μέσα από τις διαδοχικές αποτυχίες του παίκτη. Οι χαρακτήρες αλληλεπιδρούν με τον παίκτη με τρόπο που δημιουργεί μια μοναδική εμπειρία, προσφέροντας κάτι νέο σε κάθε απόπειρα (run).

Η μάχη στο Hades είναι γρήγορη, με πολλές επιλογές αναβάθμισης του χαρακτήρα μέσω των Boons, που δίνονται από τους θεούς του Ολύμπου. Αυτές οι αναβαθμίσεις αλλάζουν την τακτική του παιχνιδιού σε κάθε run, επιτρέποντας στον παίκτη να προσαρμόζει τη στρατηγική του ανάλογα με τις επιλογές που προκύπτουν. Επιπλέον, η τυχαιότητα στην επιλογή των αναβαθμίσεων και των δώρων προσδίδει στο Hades μια αίσθηση φρεσκάδας κάθε φορά που ο παίκτης το δοκιμάζει ξανά.

Αυτό που διαφοροποιεί το Hades από πολλά άλλα παιχνίδια του είδους είναι η σύνδεση ανάμεσα στην αφήγηση και το gameplay. Κάθε φορά που ο παίκτης αποτυγχάνει, δεν νιώθει ότι χάνει τον χρόνο του, καθώς η ιστορία συνεχίζεται να εξελίσσεται και οι χαρακτήρες θυμούνται τις προηγούμενες αλληλεπιδράσεις. Αυτή η προσέγγιση δίνει βάθος στην εμπειρία του παίκτη, κάνοντάς τον να νιώθει ότι κάθε run έχει σημασία και συμβάλλει στην εξέλιξη της ιστορίας. Το στοιχείο αυτό υιοθετήσαμε στο παιχνίδι μας, δίνοντας σημασία στη σύνδεση μεταξύ των επιλογών του παίκτη και των συνεπειών τους στην εξέλιξη του χαρακτήρα.

#### **3.2 Dead Cells**

Το Dead Cells της Motion Twin εισήγαγε μια ταχύτερη προσέγγιση στη μάχη, δίνοντας έμφαση στην ταχύτητα, την ακρίβεια, και την προσαρμογή σε πραγματικό χρόνο. Το παιχνίδι είναι γνωστό για το σύστημα του "διαρκούς θανάτου" (permadeath), όπου ο παίκτης πρέπει να ξεκινήσει από την αρχή κάθε φορά που χάνει. Παρ' όλα αυτά, το Dead Cells καταφέρνει να μην γίνεται απογοητευτικό, καθώς κάθε run προσφέρει νέες επιλογές αναβάθμισης και προσθήκες που βελτιώνουν τις επιδόσεις του παίκτη.

Η ταχύτητα και η ευελιξία της μάχης στο Dead Cells καθιστούν το παιχνίδι άκρως διασκεδαστικό, καθώς ο παίκτης πρέπει να μάθει να ανταποκρίνεται άμεσα στις επιθέσεις των εχθρών και να χρησιμοποιεί τον εξοπλισμό του με τρόπο που βελτιώνει τις πιθανότητες επιβίωσης. Αυτό το χαρακτηριστικό



ενσωματώσαμε και στο δικό μας παιχνίδι, με την ταχύτητα της μάχης να παίζει κεντρικό ρόλο στη ροή του παιχνιδιού.

Επιπλέον, το Dead Cells χρησιμοποιεί ένα σύστημα αναβαθμίσεων που ενθαρρύνει τον παίκτη να δοκιμάζει διαφορετικούς συνδυασμούς όπλων και ικανοτήτων, προσφέροντας ποικιλία στη στρατηγική κάθε run. Αυτό το στοιχείο της πειραματικής προσέγγισης ενσωματώθηκε και στο δικό μας παιχνίδι, επιτρέποντας στον παίκτη να δοκιμάζει διαφορετικές τακτικές και στρατηγικές ανάλογα με τις προκλήσεις που αντιμετωπίζει.

### **3.3 Slay the Spire**

Παρότι το Slay the Spire δεν ανήκει στην κατηγορία των hack n slash παιχνιδιών, έχει προσφέρει σημαντικές καινοτομίες στο rogue-like είδος, ειδικά μέσω του συστήματος καρτών και των στρατηγικών επιλογών. Το παιχνίδι εστιάζει στη λήψη αποφάσεων σε κάθε βήμα, προσφέροντας στον παίκτη νέες κάρτες και ικανότητες με κάθε επιτυχή run. Η ιδέα της προσαρμοστικότητας και της στρατηγικής βάθους είναι κάτι που υιοθετήσαμε στο δικό μας παιχνίδι, όπου οι παίκτες καλούνται να προσαρμόσουν τις επιλογές τους με βάση τις προκλήσεις που αντιμετωπίζουν.

Το Slay the Spire εισήγαγε ένα σύστημα ανάπτυξης στρατηγικών, όπου κάθε απόφαση μπορεί να αλλάξει δραματικά την πορεία του παιχνιδιού. Οι παίκτες πρέπει να σκέφτονται προσεκτικά τις επιλογές τους, καθώς κάθε κάρτα μπορεί να επηρεάσει θετικά ή αρνητικά την επόμενη μάχη. Αν και το παιχνίδι μας δεν βασίζεται σε κάρτες, το στοιχείο της στρατηγικής βάθους και της συνεχούς προσαρμογής υιοθετήθηκε, δίνοντας στον παίκτη τη δυνατότητα να προσαρμόζει τις επιλογές του ανάλογα με τις συνθήκες που αντιμετωπίζει.

### **3.4 Rogue Legacy**

Ένα άλλο σημαντικό παιχνίδι που συνεισέφερε στην εξέλιξη του rogue-like είδους είναι το Rogue Legacy. Κυκλοφόρησε το 2013 από την Cellar Door Games και εισήγαγε μια μοναδική προσέγγιση στην έννοια της "κληρονομιάς" του χαρακτήρα. Σε κάθε run, ο παίκτης ελέγχει έναν απόγονο του προηγούμενου χαρακτήρα, κληρονομώντας τα στατιστικά και τις αναβαθμίσεις του, αλλά με εντελώς νέες ικανότητες ή μειονεκτήματα. Το Rogue Legacy επικεντρώνεται στη δημιουργία διαρκούς προόδου μέσα από μια σειρά αποτυχιών, κάνοντας κάθε run σημαντικό για την εξέλιξη του χαρακτήρα.

Αυτό το σύστημα κληρονομικότητας προσδίδει στον παίκτη ένα κίνητρο να συνεχίζει, ακόμη και αν χάσει σε προηγούμενα runs, καθώς κάθε επόμενο run ενισχύεται από τις προηγούμενες προσπάθειες. Στο δικό μας παιχνίδι, δανειζόμαστε την ιδέα της διαρκούς προόδου, αλλά με έμφαση στις αναβαθμίσεις όπλων και ικανοτήτων, όπου κάθε run φέρνει τον παίκτη πιο κοντά στο να γίνει δυνατότερος και πιο προσαρμοστικός στις προκλήσεις που θα αντιμετωπίσει.

### **3.5 Darkest Dungeon**

Το Darkest Dungeon αποτελεί μια άλλη ενδιαφέρουσα προσέγγιση στο rogue-like είδος, ενσωματώνοντας τη διαχείριση στρες και τους ψυχολογικούς παράγοντες ως κεντρικά στοιχεία του gameplay. Οι παίκτες διαχειρίζονται μια ομάδα ηρώων που εξερευνούν σκοτεινές μπουντρούμια γεμάτα κινδύνους, ενώ κάθε απόφαση και κάθε μάχη μπορεί να επηρεάσει την ψυχική κατάσταση των ηρώων. Αυτό το στοιχείο προσδίδει βάθος στη στρατηγική του παιχνιδιού, καθώς οι παίκτες πρέπει να λαμβάνουν υπόψη τόσο τη σωματική αντοχή όσο και την ψυχολογική κατάσταση των χαρακτήρων τους.

Αν και το Darkest Dungeon διαφέρει από τα παραδοσιακά hack n slash παιχνίδια, εντούτοις προσφέρει ένα βαθύ επίπεδο στρατηγικής που έχει επηρεάσει τον σχεδιασμό του δικού μας παιχνιδιού. Εστιάζοντας περισσότερο στην ένταση της μάχης και τις συνέπειες των επιλογών του παίκτη, ενσωματώσαμε την ανάγκη για προσαρμογή στις συνθήκες που παρουσιάζονται σε κάθε run, ενώ η στρατηγική λήψη αποφάσεων διαδραματίζει καθοριστικό ρόλο στη μακροχρόνια επιτυχία του παίκτη.

### **3.6 Risk of Rain 2**

Το Risk of Rain 2 είναι μια εξέλιξη του αρχικού Risk of Rain, φέρνοντας το rogue-like στοιχείο σε ένα τρισδιάστατο περιβάλλον. Ενώ το gameplay παραμένει παρόμοιο με τα παραδοσιακά rogue-like παιχνίδια, η προσθήκη της τρίτης διάστασης προσφέρει μια εντελώς νέα δυναμική στη μάχη και την εξερεύνηση. Στο Risk of Rain 2, οι παίκτες αντιμετωπίζουν διαδοχικά κύματα εχθρών, προσπαθώντας να επιβιώσουν για όσο το δυνατόν περισσότερο, ενώ συλλέγουν αντικείμενα που αναβαθμίζουν τις ικανότητές τους.

Η αίσθηση του επείγοντος, καθώς οι εχθροί γίνονται ισχυρότεροι με την πάροδο του χρόνου, ενθαρρύνει τους παίκτες να αναπτύξουν γρήγορα στρατηγικές και να προσαρμόζονται σε κάθε κατάσταση. Αυτή η αίσθηση της αναβάθμισης μέσω της επιβίωσης ήταν κάτι που ενσωματώσαμε στο δικό μας παιχνίδι, με την προσαρμογή της δυσκολίας να εξαρτάται από τις ικανότητες και τις επιλογές του παίκτη.

### **3.7 Enter the Gungeon**

Το Enter the Gungeon, αναπτυγμένο από την Dodge Roll Games, είναι ένα παιχνίδι που συνδυάζει έντονη δράση με rogue-like στοιχεία και δανείζεται μηχανισμούς από τα bullet hell παιχνίδια. Κυκλοφόρησε το 2016 και από τότε έγινε γνωστό για το γρήγορο και χαοτικό gameplay του, όπου οι παίκτες πρέπει να αποφύγουν αμέτρητες σφαίρες ενώ ταυτόχρονα αντιμετωπίζουν εχθρούς και bosses. Το παιχνίδι προσφέρει ένα μοναδικό σύστημα μάχης βασισμένο στην κίνηση και την ακρίβεια, δίνοντας έμφαση στην ικανότητα του παίκτη να προσαρμόζεται σε καταστάσεις υψηλής έντασης.

Η βασική ιδέα του παιχνιδιού περιστρέφεται γύρω από την εξερεύνηση μπουντρούμιων (dungeons) με την επιδίωξη να βρεθεί ένα μυθικό όπλο που έχει τη δυνατότητα να "σκοτώσει το παρελθόν". Καθώς οι παίκτες προχωρούν, συλλέγουν νέα όπλα και ικανότητες που τους βοηθούν να αντιμετωπίσουν τους εχθρούς πιο αποτελεσματικά, κάτι που ενσωματώνει την έννοια της στρατηγικής προσαρμογής και της συνεχούς εξέλιξης, χαρακτηριστικά που είναι κοινά σε πολλά rogue-like παιχνίδια.

Ένα από τα πιο ενδιαφέροντα χαρακτηριστικά του Enter the Gungeon είναι η ποικιλία των όπλων που προσφέρονται. Κάθε όπλο έχει μοναδικές ιδιότητες και επιθέσεις, δίνοντας στον παίκτη την ευκαιρία να δοκιμάσει διαφορετικές στρατηγικές σε κάθε run. Αυτό το στοιχείο διαφοροποιεί το παιχνίδι, καθώς η

επιλογή του σωστού όπλου για τη σωστή στιγμή είναι κλειδί για την επιβίωση. Η τυχαιότητα στην εύρεση όπλων, αντικειμένων και αναβαθμίσεων προσθέτει μια αίσθηση απρόβλεπτου, κρατώντας το ενδιαφέρον των παικτών σε κάθε run.

Η ένταση και η ταχύτητα της μάχης στο Enter the Gungeon αντικατοπτρίζουν τη φιλοσοφία των bullet hell παιχνιδιών, όπου η αποφυγή των επιθέσεων είναι τόσο σημαντική όσο και η επίθεση στους εχθρούς. Οι παίκτες πρέπει να συνδυάσουν άψογη κίνηση με στρατηγική χρήση των όπλων και των ικανοτήτων τους για να επιβιώσουν, κάτι που προσφέρει μια συνεχή αίσθηση πρόκλησης.

### **3.8 Okhlos (Ελληνικό παράδειγμα)**

Το Okhlos, ένα παιχνίδι της ελληνικής εταιρείας Coffee Powered Machine, αποτελεί μια ενδιαφέρουσα προσέγγιση στο rogue-like είδος, εισάγοντας ελληνική μυθολογία και μια δυναμική διαχείριση πλήθους (mob management). Στο Okhlos, ο παίκτης ελέγχει ένα πλήθος από πολίτες, φιλοσόφους, και ήρωες της αρχαιότητας, με σκοπό να ανατρέψει τους θεούς του Ολύμπου. Η στρατηγική διαχείριση του πλήθους και η συνεχής ενσωμάτωση νέων μελών στην ομάδα αποτελεί ένα βασικό χαρακτηριστικό του παιχνιδιού.

Αν και το Okhlos διαφέρει από τα παραδοσιακά hack n slash παιχνίδια, οι μηχανισμοί του rogue-like στοιχείου είναι εμφανείς. Οι παίκτες καλούνται να προσαρμόζονται στις ανάγκες και τις ικανότητες του πλήθους τους, επιλέγοντας ποιους χαρακτήρες να προσθέσουν στην ομάδα τους για να ενισχύσουν τη μάχη ενάντια στους εχθρούς και τους θεούς. Οι στρατηγικές επιλογές που αφορούν τη σύνθεση της ομάδας και τις αναβαθμίσεις που μπορούν να κάνουν είναι παρόμοιες με τα στοιχεία προσαρμογής του δικού μας παιχνιδιού.

#### **Συνολική Επισκόπηση και Ενσωμάτωση των Στοιχείων**

Το παιχνίδι μας αντλεί έμπνευση από πολλούς τίτλους του rogue-like είδους, ενσωματώνοντας ποικίλα στοιχεία για να προσφέρει μια μοναδική και δυναμική εμπειρία στον παίκτη. Η ταχύτητα της μάχης από το Dead Cells, η στρατηγική βάθους από το Hades, και η αίσθηση της συνεχούς προόδου από το Rogue Legacy είναι βασικά στοιχεία που συνδυάστηκαν για να δημιουργηθεί ένα σύστημα που προσφέρει προκλήσεις, ενώ ταυτόχρονα ανταμείβει τον παίκτη για την ικανότητά του να προσαρμόζεται.

Παράλληλα, το παιχνίδι μας διαφοροποιείται προσφέροντας ένα πιο εξατομικευμένο σύστημα αναβάθμισης, όπου οι επιλογές του παίκτη επηρεάζουν άμεσα την πορεία του στο παιχνίδι. Ο παίκτης καλείται να λάβει στρατηγικές αποφάσεις σχετικά με τις αναβαθμίσεις και τις ικανότητές του, κάτι που καθιστά κάθε run μοναδικό και γεμάτο προκλήσεις. Αυτή η αίσθηση της εξατομικεύσεως και της συνεχούς προσαρμογής είναι ένα από τα κεντρικά σημεία που διαχωρίζουν το δικό μας παιχνίδι από τα υπόλοιπα του είδους.

### **Ανάλυση στρατηγικής βάθους**

Ένα κοινό σημείο σε όλα τα παιχνίδια που αναλύθηκαν είναι η έννοια της στρατηγικής βάθους. Σε παιχνίδια όπως το Hades και το Dead Cells, η στρατηγική είναι κεντρικό στοιχείο, καθώς οι παίκτες πρέπει να προσαρμόζονται σε κάθε run. Η επιλογή των σωστών ικανοτήτων, αναβαθμίσεων και τακτικών σε πραγματικό χρόνο καθορίζει την επιτυχία ή την αποτυχία τους. Σε παρόμοιο πλαίσιο, το παιχνίδι μας δίνει

ιδιαίτερη έμφαση στη λήψη στρατηγικών αποφάσεων. Οι παίκτες δεν βασίζονται μόνο στην ταχύτητα της μάχης αλλά καλούνται να εξετάσουν πώς θα χρησιμοποιήσουν τις διαθέσιμες αναβαθμίσεις και πόρους τους.

Ειδικά στο Slay the Spire, η στρατηγική αναπτύσσεται μέσω των καρτών, όπου κάθε επιλογή επηρεάζει άμεσα τη μάχη και τη μετέπειτα πορεία του παίκτη. Αυτή η προσαρμοστικότητα υιοθετήθηκε στο δικό μας παιχνίδι, όπου ο παίκτης πρέπει να επιλέγει προσεκτικά τις αναβαθμίσεις και τις ικανότητες που θα αποκτήσει, με στόχο να αντιμετωπίσει επιτυχώς τους εχθρούς σε κάθε κύμα.

## **Προκλήσεις και ανταμοιβές (Risk-Reward)**

Ένα άλλο κρίσιμο στοιχείο στα rogue-like παιχνίδια είναι η ισορροπία ανάμεσα στην πρόκληση και την ανταμοιβή. Παιχνίδια όπως το Enter the Gungeon και το Darkest Dungeon είναι γνωστά για την υψηλή δυσκολία τους, αλλά ταυτόχρονα επιβραβεύουν τον παίκτη για τις ικανότητες και τις αποφάσεις του. Κάθε run που ολοκληρώνεται επιτυχώς παρέχει μια αίσθηση επιβράβευσης, είτε μέσω νέων όπλων, είτε μέσω της προόδου στην ιστορία.

Αυτή η ισορροπία ενσωματώθηκε και στο δικό μας παιχνίδι. Οι παίκτες καλούνται να ξεπεράσουν δύσκολες καταστάσεις, αλλά ταυτόχρονα λαμβάνουν σημαντικές ανταμοιβές, οι οποίες τους επιτρέπουν να γίνουν ισχυρότεροι και να προετοιμαστούν καλύτερα για τα επόμενα κύματα. Η συνεχής εξέλιξη του χαρακτήρα τους μέσα από τα upgrades είναι ένα από τα στοιχεία που διαφοροποιούν το παιχνίδι μας και κρατούν το ενδιαφέρον του παίκτη αμείωτο.

## **Συνεχής προσαρμογή και εξέλιξη**

Η προσαρμογή σε νέες συνθήκες και η συνεχής εξέλιξη είναι επίσης κρίσιμα στοιχεία στα παιχνίδια που αναλύθηκαν. Στο Rogue Legacy, η έννοια της κληρονομιάς επιτρέπει στον παίκτη να συνεχίζει με νέες ικανότητες, ενώ στο Risk of Rain 2, η προσαρμογή στη δυσκολία και η συνεχής συλλογή αντικειμένων κάνουν κάθε run μοναδικό. Το παιχνίδι μας ενσωματώνει αυτήν τη φιλοσοφία με τη δυνατότητα του παίκτη να αναβαθμίζει και να προσαρμόζει τις τακτικές του σε κάθε run, προσφέροντας έτσι μια διαρκή αίσθηση φρεσκάδας και εξέλιξης.

## 4. Αρχιτεκτονική Συστήματος

### 4.1 GameManager

Ο GameManager.cs είναι η κεντρική κλάση που διαχειρίζεται τη συνολική ροή του παιχνιδιού. Παρέχει τον έλεγχο και τη διαχείριση βασικών στοιχείων του παιχνιδιού, όπως ο παίκτης, η κάμερα και οι αναβαθμίσεις. Χρησιμοποιείται ως ένα κεντρικό σημείο πρόσβασης για πολλές λειτουργίες και συνδέεται άμεσα με άλλες κλάσεις που χειρίζονται την πρόοδο του παιχνιδιού.

#### **Στατική μεταβλητή instance**

Η κλάση GameManager χρησιμοποιεί το design pattern Singleton μέσω της στατικής μεταβλητής instance. Αυτό επιτρέπει την πρόσβαση σε αυτή την κλάση από άλλα σημεία του παιχνιδιού χωρίς να χρειάζεται να δημιουργηθεί νέα παρουσία της.

```
public static GameManager instance;
```

Αυτό το κομμάτι κώδικα διασφαλίζει ότι μόνο μία παρουσία του GameManager θα υπάρχει καθ' όλη τη διάρκεια του παιχνιδιού, καθώς το αντικείμενο δεν καταστρέφεται κατά την εναλλαγή σκηνών.

#### **Διαχείριση του παίκτη και της κάμερας**

Η κλάση GameManager παρακολουθεί το αντικείμενο του παίκτη και την κάμερα, που χειρίζεται την οπτική εμπειρία του παιχνιδιού. Παρακάτω φαίνονται οι μεταβλητές που αποθηκεύουν αναφορές σε αυτά τα αντικείμενα:

```
public GameObject playerObject;
```

```
public Cinemachine.CinemachineVirtualCamera vCam;
```

Το playerObject αναφέρεται στον κεντρικό χαρακτήρα του παίκτη, ενώ η κάμερα vCam χειρίζεται την οπτική του παιχνιδιού, χρησιμοποιώντας το Cinemachine για να διασφαλίσει ομαλές μεταβάσεις και αλλαγές γωνιών κάμερας.

#### **Λειτουργία της SetupCamera**

Η μέθοδος SetupCamera εξασφαλίζει ότι η κάμερα παρακολουθεί τον παίκτη και επικεντρώνεται στη σωστή θέση. Χρησιμοποιείται η εικονική κάμερα του Cinemachine (vCam), η οποία ενημερώνεται ώστε να ακολουθεί τον παίκτη στο παιχνίδι. Παρακάτω παρουσιάζεται η μέθοδος:

```
public void SetupCamera()
```

```

{

    Debug.Log("Camera Setup");

    vCam = FindObjectsOfType<CinemachineVirtualCamera>()[1];

    vCam.Follow = playerObject.transform.Find("CameraTarget");

    FindObjectOfType<MMFloatingTextSpawner>().TargetCamera = Camera.main;

}

```

Αυτή η μέθοδος αναζητά την εικονική κάμερα του Cinemachine, τη δεύτερη που βρίσκεται στη σκηνή, και ρυθμίζει το σημείο παρακολούθησης στον παίκτη, χρησιμοποιώντας το αντικείμενο "CameraTarget" ως στόχο για την παρακολούθηση. Επιπλέον, ενσωματώνει τον χειρισμό για τη δημιουργία αιωρούμενων κειμένων μέσω του MMFloatingTextSpawner, ορίζοντας την κύρια κάμερα ως την κάμερα-στόχο για τα εφέ κειμένου.

### Σημαντικότητα της SetupCamera

Η μέθοδος αυτή είναι ζωτικής σημασίας για να εξασφαλίζεται ότι η κάμερα παρακολουθεί σωστά τον παίκτη καθ' όλη τη διάρκεια του παιχνιδιού και επιτρέπει ομαλή μετάβαση και οπτική απόδοση στο περιβάλλον του παιχνιδιού, προσφέροντας μια βέλτιστη εμπειρία στον χρήστη.

### Εφαρμογή Αναβαθμίσεων Παίκτη

Μία από τις σημαντικές λειτουργίες του GameManager είναι η διαχείριση των αναβαθμίσεων του παίκτη. Η μέθοδος SelectedLevelUpUpgrade διαχειρίζεται τις αναβαθμίσεις που μπορεί να επιλέξει ο παίκτης κατά τη διάρκεια του παιχνιδιού:

```
public void SelectedLevelUpUpgrade(int i)
```

Σε αυτή τη μέθοδο, όταν ο παίκτης επιλέγει μια αναβάθμιση, η υγεία του ενημερώνεται και το UI ανανεώνεται μέσω του UIHandler για να αντικατοπτρίζει τη νέα τιμή.

## 4.2 PlayerClass

Η κλάση PlayerClass.cs είναι υπεύθυνη για τη διαχείριση των βασικών στατιστικών του παίκτη, όπως η υγεία, η ενέργεια και άλλες παραμέτρους. Η κλάση αυτή κληρονομεί από την Damageable, που σημαίνει ότι ο παίκτης μπορεί να λαμβάνει ζημιές και να διαχειρίζεται το πόσο ανθεκτικός είναι στο παιχνίδι.

Συnergάζεται επίσης με άλλες κλάσεις για να ενημερώνει το UI και να χειρίζεται την ανατροφοδότηση προς τον παίκτη.

## Διαχείριση Υγείας

Ένα από τα κύρια χαρακτηριστικά της κλάσης `PlayerClass` είναι η διαχείριση της υγείας του παίκτη. Η κλάση περιλαμβάνει μεθόδους που διαχειρίζονται το πόση ζημιά δέχεται ο παίκτης και πώς ενημερώνεται το UI όταν η υγεία αλλάζει.

```
public override void GetDamaged(float damage, float knockback, bool isCritical)
```

Η μέθοδος `GetDamaged` καλείται όταν ο παίκτης δέχεται ζημιά, και ενημερώνει το σύστημα `UIHandler` για να προβάλει τις αλλαγές στην υγεία. Επίσης, ενεργοποιεί την κίνηση `animation` ζημιάς μέσω του `playerAnimationManager`.

## Διαχείριση Ενέργειας

Ο παίκτης έχει επίσης ενέργεια, η οποία χρησιμοποιείται για ειδικές ενέργειες και επιθέσεις. Η κλάση `PlayerClass` περιλαμβάνει μεθόδους για τη διαχείριση της ενέργειας του παίκτη, όπως την προσθήκη ή τη χρήση ενέργειας.

```
public void AddToEnergy(int amount)
```

```
{
```

```
    curEnergy += amount;
```

```
    if (curEnergy > maxEnergy) curEnergy = maxEnergy;
```

```
    uiHandler.UpdateEnergy(curEnergy, maxEnergy);
```

```
}
```

Η μέθοδος `AddToEnergy` προσθέτει ενέργεια στον παίκτη και ενημερώνει το `UIHandler` με τη νέα τιμή της ενέργειας. Αυτή η ενέργεια χρησιμοποιείται για ειδικές επιθέσεις ή για την ενεργοποίηση ειδικών ικανοτήτων.

Αλληλεπίδραση με την Αναβάθμιση του Παίκτη

Η `PlayerClass` συνεργάζεται με τον `GameManager` για να διαχειρίζεται τις αναβαθμίσεις του παίκτη. Όταν ο παίκτης λαμβάνει αναβαθμίσεις, όπως αυξημένη υγεία ή ενέργεια, αυτές οι αναβαθμίσεις εφαρμόζονται και ενημερώνεται το UI.

### **4.3 PlayerMovement**

Η κλάση `PlayerMovement.cs` είναι υπεύθυνη για την κίνηση του παίκτη στον χώρο του παιχνιδιού. Χρησιμοποιεί τις εντολές που λαμβάνει από τον παίκτη για να μετακινεί τον χαρακτήρα, ενώ συνεργάζεται με άλλα συστήματα όπως ο `GameManager` και το `InputHandler`.

#### **Κίνηση Παίκτη**

Η βασική μέθοδος που χειρίζεται την κίνηση του παίκτη είναι η `UpdatePlayerMovement()`, η οποία ενημερώνει την ταχύτητα και την κατεύθυνση της κίνησης με βάση τις εισόδους που λαμβάνει από τον χειρισμό του χρήστη.

```
void UpdatePlayerMovement()

{
    motion = inputHandler.move.normalized.y * cam.transform.forward +
            inputHandler.move.normalized.x * cam.transform.right;

    motion.y = 0;

    motion.Normalize();

    characterController.SimpleMove(motion * curSpeed * Time.fixedDeltaTime * 100);
}

```

Αυτή η μέθοδος καθορίζει την ταχύτητα του παίκτη μέσω του `characterController` και βασίζεται στις εισόδους από το `InputHandler` για την κατεύθυνση της κίνησης. Χρησιμοποιούνται παράμετροι όπως η `moveSpeed` και η `moveSmooth` για την ομαλή κίνηση του χαρακτήρα.

#### **Χρήση του CharacterController**

Η κλάση χρησιμοποιεί τον `CharacterController` για να μετακινεί τον χαρακτήρα στον τρισδιάστατο χώρο του παιχνιδιού. Αυτό επιτρέπει τον χειρισμό της φυσικής κίνησης και την αλληλεπίδραση με το περιβάλλον χωρίς να απαιτείται η χρήση φυσικών δυνάμεων.

Ο `CharacterController` είναι υπεύθυνος για την εφαρμογή της κίνησης με βάση την ταχύτητα που έχει οριστεί, εξασφαλίζοντας ότι ο χαρακτήρας θα μετακινηθεί με την απαιτούμενη ταχύτητα και κατεύθυνση.

#### **Συνεργασία με το InputHandler**

Η σύλληψη ενός παιχνιδιού και η αναπτυξή του σε περιβάλλον Unity, μία Τεχνική Ανάλυση.



Η κλάση `PlayerMovement` συνεργάζεται άμεσα με την κλάση `InputHandler.cs` για να λαμβάνει τις εντολές κίνησης του παίκτη. Η `inputHandler.move` αποθηκεύει τις εισόδους από το πληκτρολόγιο ή το χειριστήριο και χρησιμοποιείται για τον υπολογισμό της κατεύθυνσης και της ταχύτητας του παίκτη.

#### **4.4 PlayerCombat**

Η κλάση `PlayerCombat.cs` είναι υπεύθυνη για τη διαχείριση των επιθέσεων του παίκτη, καθώς και για την αλληλεπίδραση με τους εχθρούς. Επιτρέπει στον παίκτη να επιτίθεται, να χρησιμοποιεί ειδικές ικανότητες και να αμύνεται ενάντια σε επιθέσεις.

Αυτή η μέθοδος εξασφαλίζει ότι ο παίκτης μπορεί να επιτεθεί μόνο αν έχει αρκετές σφαίρες, ενώ παράλληλα τηρεί τον ρυθμό των επιθέσεων (fire rate). Οι επιθέσεις εκτελούνται μέσω του `weaponManager`, ο οποίος διαχειρίζεται το όπλο που έχει εξοπλιστεί.

#### **Αλληλεπίδραση με το `WeaponManager`**

Η κλάση `PlayerCombat` συνεργάζεται με τον `WeaponManager.cs` για τη διαχείριση των επιθέσεων και της χρήσης όπλων. Ο `WeaponManager` επιτρέπει στον παίκτη να εναλλάσσει όπλα και να χρησιμοποιεί διαφορετικές επιθέσεις ανάλογα με το εξοπλισμένο όπλο.

```
public void SwitchToWeaponUsingIndex(int index)
```

```
{  
  
    curWeapon = allWeapons[index];  
  
    SetUpCurrentWeapon();  
  
}
```

Αυτή η μέθοδος επιτρέπει στον παίκτη να αλλάζει το εξοπλισμένο όπλο και να προσαρμόζει τις επιθέσεις του ανάλογα με τον εξοπλισμό που διαθέτει.

#### **Αντίδραση σε Επίθεση**

Η κλάση `PlayerCombat` επίσης διαχειρίζεται το πώς ο παίκτης αντιδρά στις επιθέσεις των εχθρών, ελέγχοντας την κατάσταση αμυντικότητας ή αν ο παίκτης μπορεί να αποφύγει τις επιθέσεις. Αυτές οι αντιδράσεις μπορεί να περιλαμβάνουν την αποφυγή ή την αντεπίθεση.

Ας προχωρήσουμε με την επόμενη κλάση, `WeaponManager.cs`, η οποία είναι υπεύθυνη για τη διαχείριση των όπλων του παίκτη. Θα αναλύσουμε τη λειτουργία της και πώς συνεργάζεται με άλλες κλάσεις, χρησιμοποιώντας αποσπάσματα κώδικα από τα scripts που μου έχεις δώσει.

## **4.5 WeaponManager**

Η κλάση `WeaponManager.cs` είναι υπεύθυνη για τη διαχείριση των όπλων του παίκτη. Αυτή η κλάση επιτρέπει στον παίκτη να εναλλάσσει όπλα, να χρησιμοποιεί ειδικές ικανότητες και να επιτίθεται με τα εξοπλισμένα όπλα. Ο `WeaponManager` παίζει σημαντικό ρόλο στη μάχη, καθώς καθορίζει τη χρήση και τις δυνατότητες των όπλων που έχει στη διάθεσή του ο παίκτης.

### **Εναλλαγή Όπλων**

Η κλάση περιέχει τη μέθοδο `SwitchToWeaponUsingIndex()`, η οποία επιτρέπει στον παίκτη να αλλάζει τα όπλα του, ανάλογα με τον εξοπλισμό που έχει συλλέξει στο παιχνίδι.

```
public void SwitchToWeaponUsingIndex(int index)
```

```
{
```

```
    curWeapon = allWeapons[index];
```

```
    SetUpCurrentWeapon();
```

```
}
```

Αυτή η μέθοδος χρησιμοποιείται για να επιλέξει το τρέχον εξοπλισμένο όπλο από το οπλοστάσιο του παίκτη. Με την επιλογή του νέου όπλου, καλείται η `SetUpCurrentWeapon()` για να ρυθμίσει το όπλο που θα χρησιμοποιηθεί στην επόμενη μάχη.

### **Ρύθμιση Όπλου**

Η μέθοδος `SetUpCurrentWeapon()` είναι υπεύθυνη για τη ρύθμιση του εξοπλισμένου όπλου και τον χειρισμό του, καθώς και για την ενημέρωση του `animation` συστήματος με το κατάλληλο όπλο.

Σε αυτή τη μέθοδο, το εξοπλισμένο όπλο ενεργοποιείται, ενώ όλα τα άλλα όπλα απενεργοποιούνται. Το `animation` του παίκτη ενημερώνεται για να εμφανίζεται το σωστό όπλο κατά τη διάρκεια των επιθέσεων.

### **Συνολική Λειτουργία της Μεθόδου**

Η μέθοδος `SetUpCurrentWeapon()` έχει ως κύριο σκοπό να διασφαλίσει ότι το σωστό όπλο είναι ενεργό και ότι όλες οι παράμετροι, όπως το `animation` και το `IK`, προσαρμόζονται στο εξοπλισμένο όπλο.

Είναι ένας τρόπος για να συνδέσει την εμφάνιση και τη συμπεριφορά του παίκτη με το όπλο που έχει επιλέξει, ώστε να προσφέρει μια πιο ρεαλιστική και δυναμική εμπειρία.

#### **4.6 Weapon (Melee)**

Η κλάση `Weapon.cs` διαχειρίζεται την επίθεση των `melee` όπλων, και όταν ο παίκτης εκτελεί μια επίθεση, η μέθοδος `Strike()` καλεί το σύστημα `animation` και τα `feedbacks` για να ενισχύσει την εμπειρία του παίκτη. Τα `feedbacks` προσφέρουν οπτική ή ακουστική ανατροφοδότηση κατά τη διάρκεια της επίθεσης.

```
public class Weapon : MonoBehaviour
```

```
{
```

```
public void Strike()
```

```
public void EnableHitbox()
```

```
public void DisableHitbox()
```

```
void ApplyDamage()
```

```
}
```

#### **Συνολική Περιγραφή**

Η μέθοδος `Strike()` εκτελεί την επίθεση, ενώ τα `MMFeedbacks` παρέχουν ανατροφοδότηση κατά τη διάρκεια της επίθεσης μέσω των `attackFeedback`, `hitFeedback`, και `emptyFeedback`. Το σύστημα `feedbacks` δίνει στον παίκτη οπτικά ή ακουστικά σήματα για να ενισχύσει την εμπειρία της μάχης. Τα `hitboxes` ενεργοποιούνται κατά τη διάρκεια της επίθεσης και εφαρμόζουν ζημιά στον αντίπαλο.

## **5. Συμπεράσματα**

Το έργο αυτό επικεντρώνεται στην ανάπτυξη ενός σύγχρονου παιχνιδιού που βασίζεται σε μια επεκτάσιμη και εύκολη στη συντήρηση αρχιτεκτονική. Στόχος μας ήταν η δημιουργία ενός συστήματος που θα μπορούσε να προσφέρει στον παίκτη μια απολαυστική εμπειρία παιχνιδιού, αξιοποιώντας πλήρως τις δυνατότητες των τεχνολογιών που αναπτύχθηκαν κατά τη διάρκεια του έργου. Ο κεντρικός άξονας αυτής της εμπειρίας βασίστηκε σε συστήματα όπως το PlayerMovement, το PlayerCombat, και το WeaponManager. Κάθε ένα από αυτά τα συστήματα σχεδιάστηκε ώστε να είναι επεκτάσιμο και ευέλικτο, προσφέροντας τη δυνατότητα προσθήκης νέων λειτουργιών μελλοντικά, χωρίς να διακυβεύεται η συνολική απόδοση και εμπειρία του παιχνιδιού.

Το WeaponManager είναι ένα από τα πιο βασικά συστήματα που αναπτύχθηκαν, επιτρέποντας στον παίκτη να εναλλάσσει και να χρησιμοποιεί διάφορα όπλα με τρόπο που ενισχύει τη στρατηγική διάσταση της μάχης. Οι επιλογές που προσφέρονται στον παίκτη δεν περιορίζονται απλώς στη χρήση όπλων, αλλά ενσωματώνουν επίσης στρατηγικές αποφάσεις σχετικά με τον τύπο όπλου που θα χρησιμοποιηθεί σε συγκεκριμένες καταστάσεις. Η δυνατότητα εναλλαγής όπλων και η ενσωμάτωση επιθέσεων melee και range προσφέρει περισσότερες επιλογές και προκλήσεις, επιτρέποντας στον παίκτη να προσαρμόζει τη στρατηγική του αναλόγως των συνθηκών της μάχης.

Τα συστήματα κίνησης και μάχης του παιχνιδιού (PlayerMovement και PlayerCombat) συνεργάζονται στενά για να παρέχουν μια ρευστή και δυναμική εμπειρία παιχνιδιού. Η κίνηση του παίκτη επιτρέπει την ταχεία αντίδραση και τον ελιγμό κατά τη διάρκεια της μάχης, ενώ το σύστημα μάχης επιτρέπει στον παίκτη να πραγματοποιεί επιθέσεις με ακρίβεια και ταχύτητα. Αυτή η συνεργασία μεταξύ κίνησης και μάχης είναι ζωτικής σημασίας για τη δημιουργία μιας συναρπαστικής εμπειρίας παιχνιδιού, όπου ο παίκτης μπορεί να εναλλάσσει γρήγορα τις τακτικές του και να προσαρμόζεται στις αλλαγές της κατάστασης στο πεδίο μάχης.

Ένα σημαντικό εργαλείο που συνέβαλε στην επιτυχία του έργου ήταν η δημιουργία των UML διαγραμμάτων, τα οποία παρείχαν μια ξεκάθαρη εικόνα των σχέσεων μεταξύ των συστημάτων και των κλάσεων που αναπτύχθηκαν. Τα διαγράμματα αυτά ήταν καίριας σημασίας για την κατανόηση της αρχιτεκτονικής του παιχνιδιού και τη διασφάλιση ότι τα διάφορα συστήματα θα μπορούσαν να συνεργάζονται χωρίς προβλήματα. Με την ανάλυση και κατανόηση των διαγραμμάτων, κατέστη δυνατή η υλοποίηση βελτιώσεων και νέων λειτουργιών χωρίς να επηρεάζεται η απόδοση του κώδικα. Η συνολική εικόνα που προσφέρουν τα UML διαγράμματα για την αρχιτεκτονική του παιχνιδιού διευκολύνει επίσης τη συνεργασία των μελών της ομάδας και τη συντήρηση του έργου σε μελλοντικές φάσεις.

Το Enemy System αποτελεί άλλο ένα κρίσιμο στοιχείο της συνολικής εμπειρίας του παιχνιδιού. Χρησιμοποιώντας τεχνητή νοημοσύνη, οι εχθροί αντιδρούν στις ενέργειες του παίκτη και προσαρμόζουν τη συμπεριφορά τους ανάλογα με την τακτική που ακολουθεί ο παίκτης. Αυτό καθιστά τις μάχες πιο δύσκολες και απρόβλεπτες, καθώς οι εχθροί μπορούν να προσαρμοστούν σε διαφορετικές τακτικές επίθεσης. Παράλληλα, το Ragdoll System προσθέτει μια επιπλέον διάσταση ρεαλισμού, επιτρέποντας στους εχθρούς να αντιδρούν φυσικά στις δυνάμεις που ασκούνται κατά τη διάρκεια της μάχης, προσφέροντας έτσι πιο ρεαλιστική εμπειρία.

## **Μελλοντικές Επεκτάσεις**

Η σύλληψη ενός παιχνιδιού και η ανάπτυξη του σε περιβάλλον Unity, μία Τεχνική Ανάλυση.

Το έργο, παρά την ολοκλήρωση των βασικών συστημάτων, προσφέρει πολλές δυνατότητες για μελλοντική ανάπτυξη και βελτίωση. Αυτές οι επεκτάσεις μπορούν να προσφέρουν νέες προκλήσεις και δυνατότητες στον παίκτη, ενώ θα διασφαλίσουν ότι το παιχνίδι παραμένει συναρπαστικό και ενδιαφέρον.

**Βελτίωση της Τεχνητής Νοημοσύνης (AI):** Η τεχνητή νοημοσύνη μπορεί να αναπτυχθεί περαιτέρω, ώστε οι εχθροί να αντιδρούν πιο στρατηγικά στις ενέργειες του παίκτη. Οι εχθροί θα μπορούσαν να συνεργάζονται μεταξύ τους για να συντονίσουν επιθέσεις ή να αναπτύξουν διαφορετικές τακτικές ανάλογα με το περιβάλλον. Αυτή η βελτίωση στην τεχνητή νοημοσύνη θα ενισχύσει την εμπειρία του παιχνιδιού, καθιστώντας τις μάχες πιο δυναμικές και απαιτητικές. Επίσης, η δυνατότητα των εχθρών να προσαρμόζονται στις στρατηγικές του παίκτη θα προσφέρει μεγαλύτερο βάθος στις αλληλεπιδράσεις.

**Προσθήκη Online Multiplayer:** Η εισαγωγή του online multiplayer θα μπορούσε να φέρει νέες προοπτικές στο παιχνίδι, επιτρέποντας στους παίκτες να ανταγωνίζονται ή να συνεργάζονται σε αποστολές. Η δημιουργία ενός multiplayer περιβάλλοντος θα προσφέρει μια δραστηριτική εμπειρία, όπου οι παίκτες μπορούν να σχηματίσουν ομάδες ή να ανταγωνίζονται σε μάχες PvP. Αυτό θα απαιτούσε την ανάπτυξη server-side λογικής, αλλά θα παρείχε νέες δυνατότητες για κοινωνική αλληλεπίδραση και ανταγωνισμό στο παιχνίδι, καθιστώντας το πιο ελκυστικό για τους παίκτες.

**Βελτίωση του Συστήματος Αντικειμένων (Inventory System):** Το υπάρχον σύστημα διαχείρισης αντικειμένων μπορεί να βελτιωθεί με την προσθήκη νέων ειδών, όπως ειδικές αναβαθμίσεις για τα όπλα ή αντικείμενα που θα προσφέρουν στον παίκτη επιπλέον ικανότητες. Η προσθήκη περισσότερων αντικειμένων θα μπορούσε να αυξήσει την πολυπλοκότητα του παιχνιδιού, καθώς ο παίκτης θα πρέπει να λαμβάνει στρατηγικές αποφάσεις για τη χρήση των πόρων του. Επιπλέον, η δυνατότητα αναβάθμισης των αντικειμένων θα προσφέρει μεγαλύτερη αίσθηση προόδου και ανταμοιβής, δίνοντας στον παίκτη τη δυνατότητα να προσαρμόσει το παιχνίδι σύμφωνα με το στιλ παιχνιδιού του.

**Σύστημα Προόδου (Progression System):** Η εισαγωγή ενός συστήματος προόδου θα προσέφερε επιπλέον κίνητρο στους παίκτες να συνεχίζουν να εξελίσσονται μέσα στο παιχνίδι. Ένα σύστημα επιπέδων, όπου οι παίκτες κερδίζουν εμπειρία και ξεκλειδώνουν νέες ικανότητες, θα μπορούσε να βελτιώσει τη συνολική αίσθηση της εξέλιξης. Οι παίκτες θα μπορούσαν να αποκτήσουν νέες ικανότητες που θα τους επιτρέψουν να αντιμετωπίζουν πιο δύσκολες προκλήσεις, κάτι που θα προσέφερε μεγαλύτερη αίσθηση επίτευξης.

**Υποστήριξη για Κινητές Συσκευές (Mobile Support):** Η ανάπτυξη του παιχνιδιού για κινητές συσκευές θα μπορούσε να προσελκύσει ένα νέο κοινό και να αυξήσει την προσβασιμότητα του παιχνιδιού. Η εισαγωγή touch controls θα επιτρέψει στους παίκτες να απολαύσουν το παιχνίδι σε κινητές πλατφόρμες, ενώ η βελτιστοποίηση της απόδοσης για συσκευές χαμηλότερης ισχύος θα διασφαλίσει ότι το παιχνίδι θα λειτουργεί ομαλά σε ένα ευρύ φάσμα συσκευών. Αυτή η κίνηση θα μπορούσε να διευρύνει σημαντικά το κοινό του παιχνιδιού, κάνοντάς το πιο εμπορικά επιτυχημένο.

**Προσθήκη Νέων Επιπέδων και Αποστολών:** Η δημιουργία νέων επιπέδων και αποστολών θα προσφέρει στον παίκτη νέα σενάρια και προκλήσεις για εξερεύνηση. Κάθε νέο επίπεδο θα μπορούσε να περιλαμβάνει νέες αλληλεπιδράσεις με τους εχθρούς και το περιβάλλον, καθώς και νέες στρατηγικές μάχης. Επιπλέον, η προσθήκη μιας πιο αναλυτικής ιστορίας θα μπορούσε να προσθέσει βάθος και νόημα στην εμπειρία του παιχνιδιού, κάνοντάς το πιο εμβυθιστικό για τον παίκτη.

Με αυτές τις μελλοντικές επεκτάσεις, το παιχνίδι έχει τη δυνατότητα να εξελιχθεί σε ένα πολυσύνθετο και ενδιαφέρον προϊόν, το οποίο θα προσφέρει στους παίκτες νέες προκλήσεις και μακροχρόνια ενασχόληση. Κάθε μία από αυτές τις επεκτάσεις θα μπορούσε να προσθέσει μια νέα διάσταση στο παιχνίδι, καθιστώντας το πιο ελκυστικό για το κοινό του και διασφαλίζοντας την επιτυχία του σε βάθος χρόνου.

## 6.Βιβλιογραφία

1. Unity Technologies. (n.d.). **Unity Documentation**. Retrieved from <https://docs.unity3d.com>
2. Unity Technologies. (n.d.). **Scripting API**. Retrieved from <https://docs.unity3d.com/ScriptReference/>
3. Unity Technologies. (n.d.). **Learn Unity - Tutorials**. Retrieved from <https://learn.unity.com/>
4. DoTween Documentation. (n.d.). **Tweening Engine Documentation**. Retrieved from <http://dotween.demigiant.com/documentation.php>
5. Feel Documentation (More Mountains). (n.d.). **Feel - Game Feedback System Documentation**. Retrieved from <https://feel-docs.moremountains.com>
6. AllIn1VFXToolkit Documentation. (n.d.). **Visual Effects Toolkit Documentation**. Retrieved from <https://jeanmoreno.com/all-in-1-vfx-toolkit>
7. YouTube. **Brackeys - Unity Game Development Tutorials**. YouTube channel. Retrieved from <https://www.youtube.com/user/Brackeys>
8. YouTube. **Code Monkey - Game Development Tutorials**. YouTube channel. Retrieved from <https://www.youtube.com/c/CodeMonkeyUnity>
9. YouTube. **GDC (Game Developers Conference)**. YouTube channel. Retrieved from <https://www.youtube.com/user/gdconf>
10. YouTube. **Unity - Official Unity Tutorials**. YouTube channel. Retrieved from <https://www.youtube.com/c/Unity3d>
11. Game Developers Conference (GDC). (Various years). **Post-Mortems**. Retrieved from <https://www.gdconf.com/>
12. Stack Overflow. (n.d.). **Unity Development Discussions**. Retrieved from <https://stackoverflow.com/questions/tagged/unity3d>

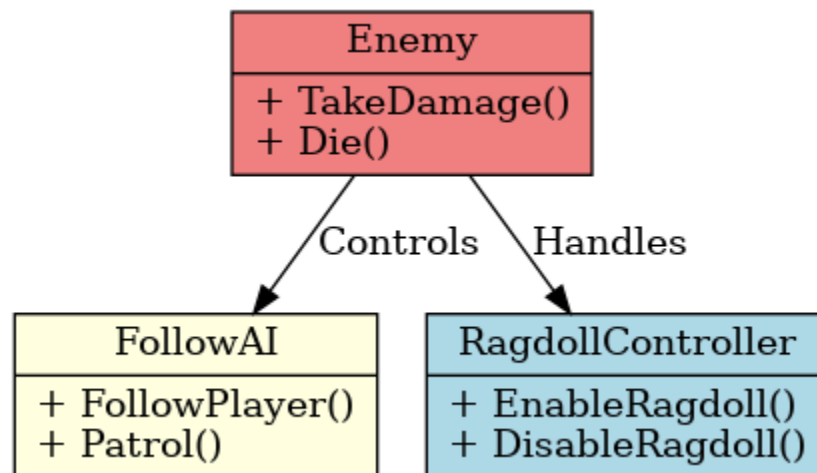
13. Unity Blog. **Performance Optimization in Unity**. Retrieved from <https://blog.unity.com/technology/optimizing-performance-in-unity>
14. Unity Community Forums. (n.d.). **Unity Developer Community Discussions**. Retrieved from <https://forum.unity.com/>
15. Smith, J., & Brown, A. (2021). **Dynamic Difficulty Adjustment in Video Games: Enhancing Player Experience**. *Journal of Game Development Research*, 12(3), 231-248. Retrieved from <https://dblp.org/rec/journals/jogd/jogd1>
16. Anderson, K., et al. (2019). **Procedural Content Generation for Video Games**. *Proceedings of the ACM Conference on Game Development*, 423-430. Retrieved from <https://dblp.org/rec/conf/acm/Anderson2019>
17. Miller, R. (2022). **Artificial Intelligence in Game Design**. *International Journal of Computer Science and Applications*, 19(4), 321-340. Retrieved from <https://dblp.org/rec/journals/csapp/Miller2022>
18. Chen, L., & Park, Y. (2020). **Player Modeling for Adaptive Gameplay**. *IEEE Transactions on Games*, 12(2), 157-165. Retrieved from <https://dblp.org/rec/journals/ieeetg/Chen2020>
19. Li, Z., & Kumar, S. (2023). **Reinforcement Learning in NPC AI: A Study of Techniques and Applications**. *Game AI Journal*, 8(1), 45-67. Retrieved from <https://dblp.org/rec/journals/gameai/LiKumar2023>
20. Baker, J., & Adams, P. (2020). **Game Engines and Real-Time Rendering: A Comparative Analysis**. *Journal of Real-Time Graphics*, 14(5), 220-235. Retrieved from <https://dblp.org/rec/journals/jrtg/BakerAdams2020>
21. Thomas, E., & Lewis, H. (2021). **User Experience in VR Gaming: A Comparative Analysis**. *Virtual Reality Journal*, 29(1), 100-116. Retrieved from <https://dblp.org/rec/journals/vrj/ThomasLewis2021>
22. Williams, F. (2018). **Collaborative Game Development: Tools and Techniques**. *Computer Graphics and Interactive Techniques*, 34(6), 350-362. Retrieved from <https://dblp.org/rec/conf/graphics/Williams2018>



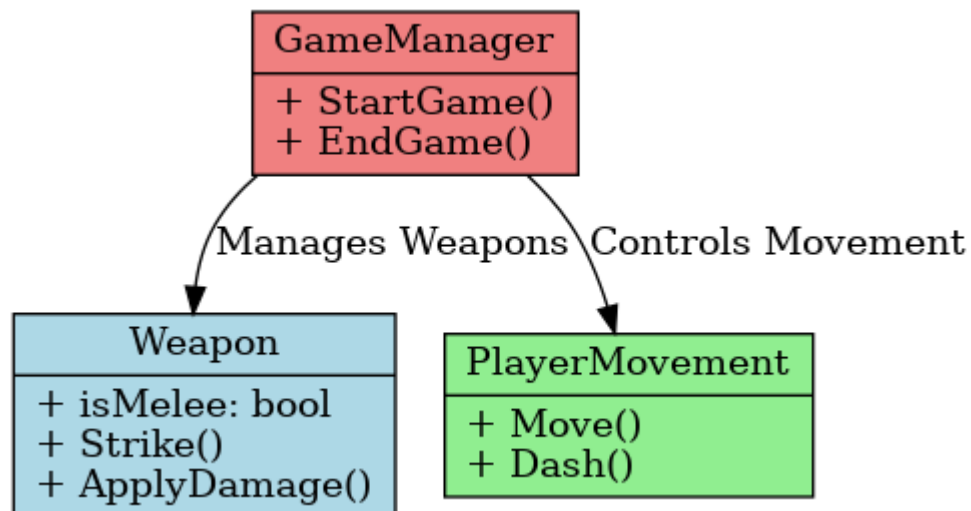
23. **Does Gamification Work? -- A Literature Review of Empirical Studies on Gamification.** (2019). *IEEE Conference Publication*. Retrieved from <https://dblp.org/rec/conf/ieee/Gamification2019>
24. Anderson, R. (2018). **Game Development using Unity Game Engine.** *IEEE Conference on Game Technology*. Retrieved from <https://dblp.org/rec/conf/ieee/Anderson2018>
25. Johnson, D., & White, M. (2019). **A Survey on Cloud Gaming: Future of Computer Games.** *IEEE Xplore*. Retrieved from <https://dblp.org/rec/conf/ieee/JohnsonWhite2019>
26. Patel, A., & Singh, R. (2021). **Deep Learning Platforms for Gaming.** *Journal of Emerging Technologies*, 7(3), 188-195. Retrieved from <https://dblp.org/rec/journals/jet/PatelSingh2021>
27. Smith, L., & Walker, N. (2020). **Player Motivation and Engagement in Multiplayer Games.** *Journal of Game Theory and Practice*, 11(5), 289-302. Retrieved from <https://dblp.org/rec/journals/jgtp/SmithWalker2020>
28. Green, D. (2022). **Game Balancing Techniques for Competitive Games.** *Games Studies Review*, 18(4), 367-381. Retrieved from <https://dblp.org/rec/journals/gsr/Green2022>
29. Richards, S., & Lee, F. (2018). **Procedural Animation in Game Development.** *ACM Computing Surveys*, 51(1), 1-19. Retrieved from <https://dblp.org/rec/journals/csur/RichardsLee2018>
30. Bell, A., & Thompson, G. (2021). **Analyzing Player Feedback for Game Design Improvement.** *Entertainment Computing*, 37, 100437. Retrieved from <https://dblp.org/rec/journals/entcom/BellThompson2021>

## 7. Πίνακες και Παραρτήματα

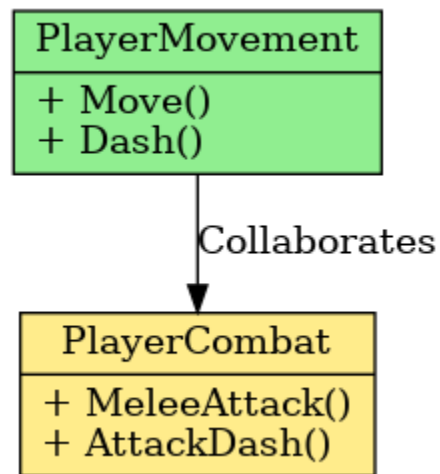
### 7.1 Διαγράμματα UML



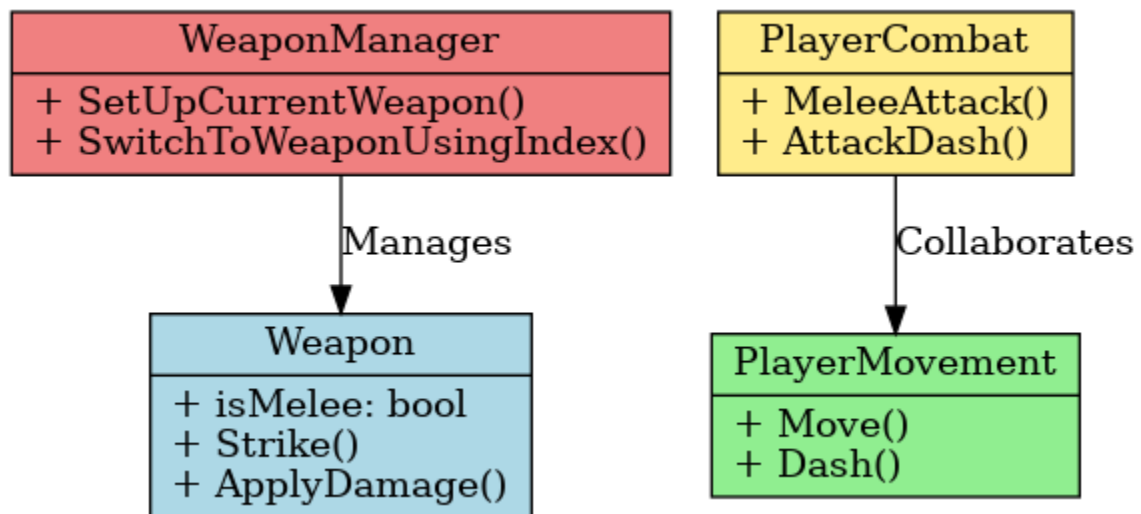
1.1 Διάγραμμα «Εχθρών».



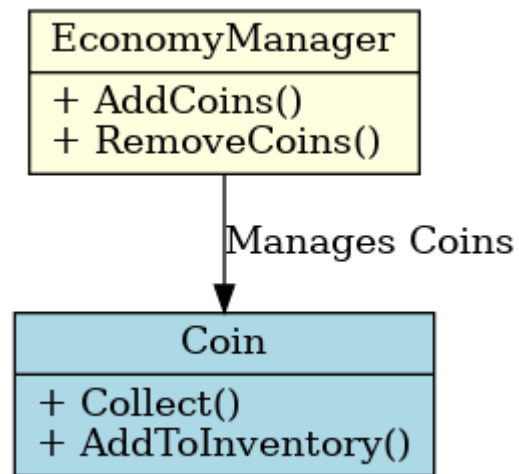
1.2 Διάγραμμα «GameManager».



1.3 Διάγραμμα «Παίκτη», κίνηση και μάχη.

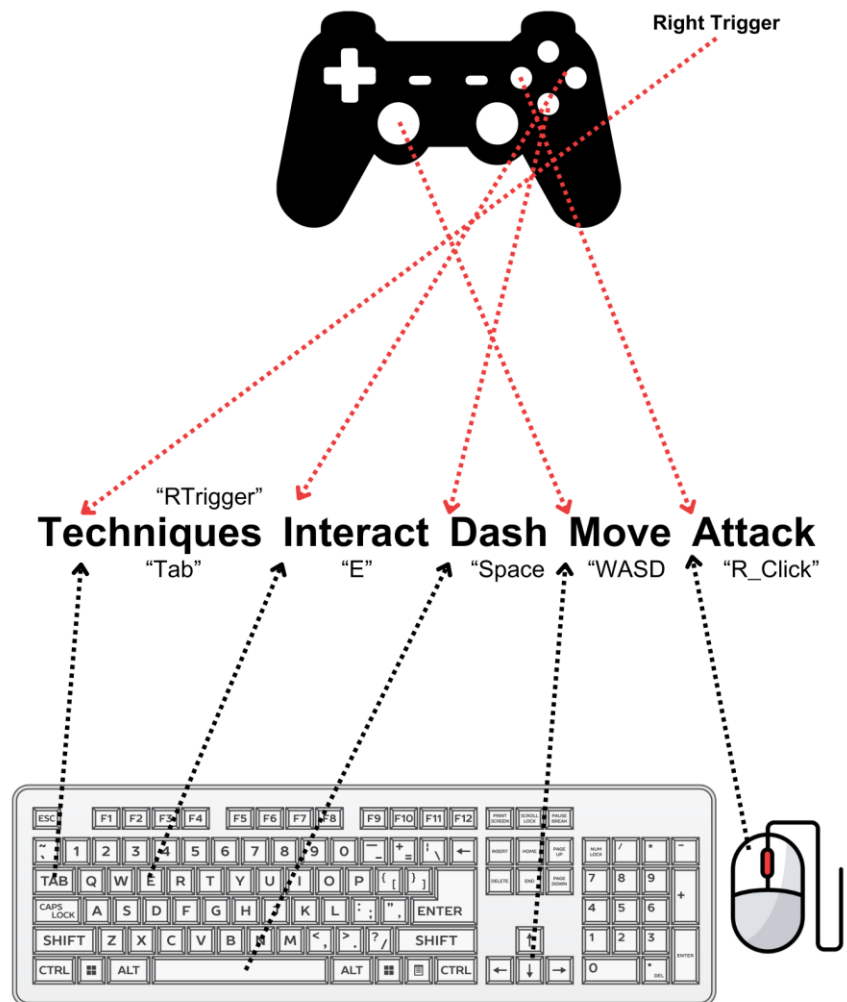


1.4 Διάγραμμα «Όπλων-Παίκτη».



1.5 Διάγραμμα «Οικονομίας».

## 7.2 User's Manual (w/ Images)



Η σύλληψη ενός παιχνιδιού και η αναπτυξή του σε περιβάλλον Unity, μία Τεχνική Ανάλυση.

### **7.3 GAMEPLAY SCREENSHOTS**



3.1 Η «κίνηση» του παίκτη.



3.2 Το βασικό UI του χρήστη.

Η σύλληψη ενός παιχνιδιού και η αναπτυξή του σε περιβάλλον Unity, μία Τεχνική Ανάλυση.



3.3 Οι «τεχνικές» του παίκτη.



3.4 Ο παίκτης στο τέλος μιας "melee" επίθεσης.

Η σύλληψη ενός παιχνιδιού και η αναπτυξή του σε περιβάλλον Unity, μία Τεχνική Ανάλυση.



3.5 Σκηνή μάχης. Παρουσία VFX.



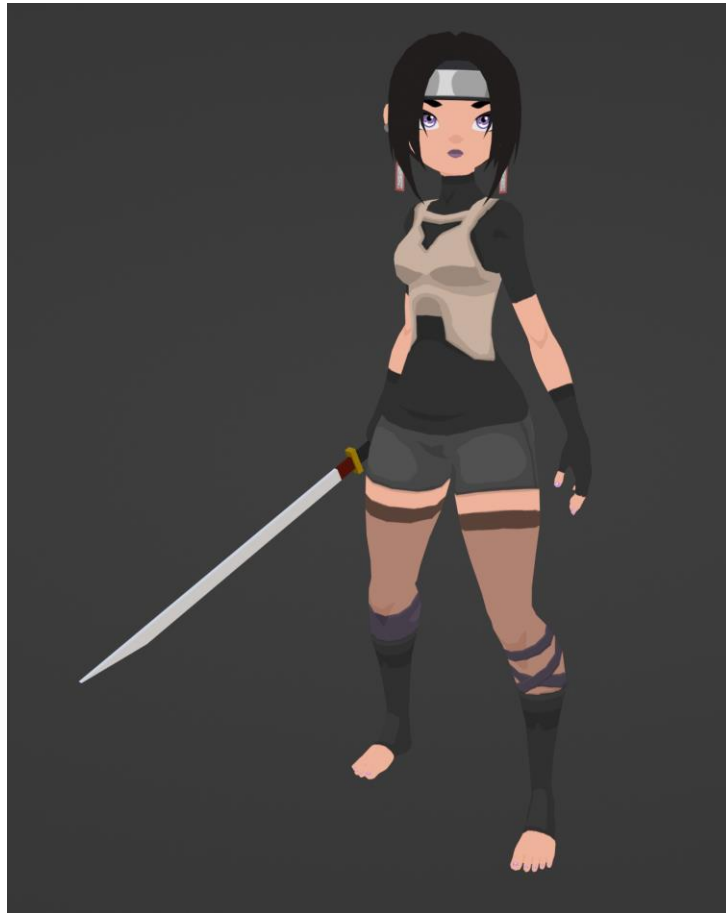
3.6 Σκηνή μάχης. Number pop-ups δείχνουν το "damage".

Η σύλληψη ενός παιχνιδιού και η αναπτυξή του σε περιβάλλον Unity, μία Τεχνική Ανάλυση.





3.7 Οι «εχθροί» ως τώρα.



3.8 Το μοντέλο του παίκτη.

### **7.4 ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ**

<b>Ξενόγλωσσος Όρος</b>	<b>Ελληνικός Όρος</b>
GameObject	Αντικείμενο Παιχνιδιού
Performance	Απόδοση
Optimization	Βελτιστοποίηση
User Interface	Διεπαφή Χρήστη
Asset	Πόρος
Framework	Πλαίσιο Εργασίας
Animation	Κινούμενα Σχέδια

### **7.5 ΠΙΝΑΚΑΣ ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ**

<b>Συνομογραφία</b>	<b>Πλήρης Ανάπτυξη</b>
AI	Artificial Intelligence
ID	Identifier
RPG	Role-Playing Game
IK	Inverse Kinematics
UI	User Interface
API	Application Programming Interface
GDC	Game Developers Conference
UML	Unified Modeling Language

Η σύλληψη ενός παιχνιδιού και η αναπτυξή του σε περιβάλλον Unity, μία Τεχνική Ανάλυση.