



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
UNIVERSITY OF WEST ATTICA

ΤΜΗΜΑ ΒΙΟΜΗΧΑΝΙΚΗΣ ΣΧΕΔΙΑΣΗΣ ΚΑΙ ΠΑΡΑΓΩΓΗΣ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΨΗΦΙΟΠΟΙΗΣΗ ΑΝΤΙΚΕΙΜΕΝΩΝ ΜΕ ΜΟΝΑΔΙΚΑ ΤΕΚΜΗΡΙΑ

ΣΥΓΓΡΑΦΕΑΣ

ΣΤΑΜΑΤΙΟΣ ΣΠΑΓΑΔΩΡΟΣ

ΑΜ: 71444899

ΕΙΣΗΓΗΤΡΙΑ

ΕΛΕΝΗ ΑΙΚΑΤΕΡΙΝΗ ΛΕΛΙΓΚΟΥ

ΑΘΗΝΑ 2024

Η παρούσα διπλωματική εργασία εγκρίθηκε ομόφωνα από την τριμελή εξεταστική επιτροπή, η οποία ορίστηκε από την Γ.Σ. του Τμήματος Μηχανικών Βιομηχανικής Σχεδίασης και Παραγωγής του Πανεπιστημίου Δυτικής Αττικής, σύμφωνα με το νόμο και τον εγκεκριμένο Οδηγό Σπουδών του τμήματος.

Μέλη της Επιτροπής

Λεβίγκου Ελένη-Αικατερίνη (Επιβλέπουσα)

Κάντζος Δημήτριος (Μέλος)

Δρόσος Χρήστος (Μέλος)

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ/ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Ο κάτωθι υπογεγραμμένος Σταμάτιος Σπαγαδώρος του Νικολάου, με αριθμό μητρώου 71444899, φοιτητής του Πανεπιστημίου Δυτικής Αττικής της Σχολής Μηχανικών του Τμήματος Βιομηχανικής Σχεδίασης και Παραγωγής, δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της πτυχιακής/διπλωματικής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία.

Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου».

Ο Δηλών



Σταμάτιος Σπαγαδώρος

Ευχαριστίες

Ευχαριστώ την επιβλέπουσα καθηγήτρια, κυρία Ελένη Αικατερίνη Δελίγκου, για την ευκαιρία που μου έδωσε να αναλάβω την σύνθεση πτυχιακής εργασίας πάνω στον κλάδο μου, εκείνον της κατασκευής βιντεοπαιχνιδιών. Την ευχαριστώ, επίσης, για την υπομονή που έδειξε κατά το ερευνητικό κομμάτι και για την πολύτιμη βοήθεια που μου προσέφερε μέσα από την ομάδα της και την ουσιώδη συνεργασία μου με τον Σταματάκη Δημήτριο, τον οποίο προσωπικά ευχαριστώ.

Επίσης θα ήθελα να ευχαριστήσω θερμά τον Χρηστίδη Ιωάννη για την πολύτιμη εισαγωγική καθοδήγηση του στον χώρο των έξυπνων συμβολαίων με την χρήση τεχνολογίας Blockchain και για τις εξαιρετες ικανότητες του στην διαχείριση ανθρωπίνου δυναμικού και επικοινωνίας, οι οποίες δημιούργησαν ένα όμορφο παραγωγικό περιβάλλον. Αποτελώντας, τόσο έμπνευση όσο και έναν καλόκαρδο φάρο αξιών μέσα στην ομάδα.

Θα ήθελα, επίσης, να ευχαριστήσω τους γονείς μου για την πολυετή στήριξη που μου προσέφεραν, κάνοντας δυνατή την αφοσίωση μου στην επιστήμη και την μελέτη της επιστήμης υπολογιστών και της φιλοσοφίας. Όπως, και τους εγκάρδιους φίλους μου και συμφοιτητές Μάριο Τοπαρόπουλο και Μιχάλη Βάλβη, για όλες τις χαρούμενες στιγμές στο πανεπιστήμιο.

Τέλος, ευχαριστώ τον άνθρωπο που στάθηκε δίπλα μου περισσότερο από οποιονδήποτε άλλον, και με το πέρας του χρόνου με βοηθούσε όλο και περισσότερο, φωτίζοντας τις σκοτεινές μέρες που δεν έβλεπα το φως του ήλιου, την Κατερίνα.

Περίληψη

Στα πλαίσια της ερευνητικής μελέτης των λειτουργιών και των σεναρίων χρήσης των NFT μέσα από τα βιντεοπαιχνίδια, και την ανάλυση των συστημάτων αυτών προς την οικονομία και Βιομηχανία 4.0, συντάχθηκε η ακόλουθη ερευνητική αναφορά.

Κατά το θεωρητικό μέρος της έρευνας και της τεχνολογικής ανασκόπησης, παρουσιάζονται διάφορες τεχνολογίες αιχμής που πλαισιώνουν το εγχείρημα. Αρχικά, συντάσσεται μια οντολογική ανάλυση ασαφούς φύσης προς την γενικότερη έννοια των ψηφιακών οντοτήτων, τα οποία χαρακτηρίζονται με μοναδικά τεκμήρια. Ακολούθως, παρουσιάζεται η απεικόνιση των ψηφιακών οντοτήτων στα βιντεοπαιχνίδια, καθώς και αναλύονται συνοπτικά οι διάφορες τεχνολογίες και τα σημεία προστριβής ανάμεσα τους.

Ο κεντρικός πυρήνας παρουσιάζει την τεχνολογία Blockchain, με τις λειτουργίες και τα χαρακτηριστικά της, εμβαθύνοντας στην χρήση των δικτύων Blockchain προς τα βιντεοπαιχνίδια. Έπειτα, αναλύονται ανεπτυγμένες τεχνολογίες της βιομηχανίας βιντεοπαιχνιδιών, οι οποίες βρίσκουν χρήση στο κατασκευαστικό μέρος, και αναλύεται η ανάμειξη τους με τα συστήματα της Βιομηχανίας 4.0. Κύριο ρόλο διαδραματίζουν οι τρόποι αναπαράστασης και ασύγχρονης επεξεργασίας των οντοτήτων και οι τεχνολογίες που υποστηρίζουν την κατασκευή διαλειτουργικών παιχνιδιών. Ιδιαίτερη έμφαση δίνεται στα κυβερνητικά συστήματα, όπως τα κυβερνο-φυσικά συστήματα, τα οποία αναφέρονται και αναλύονται καθόλη την διάρκεια της εργασίας, και αποτελούν κύριο σημείο σύνδεσης της βιομηχανίας παιχνιδιών με την Βιομηχανία 4.0. Έπειτα, προς το τέλος του θεωρητικού μέρους, θα δούμε πως συνδυάζονται οι παραπάνω τεχνολογίες ανάμεσα στον ψηφιακό και φυσικό χώρο και θα προσδιορίσουμε το περιεχόμενο της Βιομηχανίας 4.0 και τις τεχνολογίες, αυτής.

Στο πρακτικό μέρος, παρουσιάζονται μοντέρνα εργαλεία και τεχνολογίες ψηφιοποίησης και αναπαράστασης των ψηφιακών οντοτήτων, καθώς και η τεχνολογία των έξυπνων συμβολαίων και του δικτύου Blockchain. Διαλέγονται εργαλεία αιχμής και δημιουργείται ένα πρότυπο παιχνίδι που εγκλείει ένα μεγάλο πλήθος μοντέρνων χαρακτηριστικών, αξιοποιώντας τις λειτουργίες του δικτύου Ethereum και της γλώσσας Solidity, του IPFS, της μηχανής Unreal Engine 5 και των νευρωνικών δικτύων για την δημιουργία καρτών μέσα από μοντέλα Diffusion.

Στο τέλος, παρουσιάζεται το παιχνίδι και οι δυνατότητες που προσφέρει μέσα από την συνέργεια όλων αυτών των μεμονωμένων εργαλείων και οι λειτουργίες του έξυπνου συμβολαίου.

Τέλος, μέσα από τις παρατηρήσεις, οι οποίες πήραν μέρος καθόλη την διάρκεια τέλεσης του έργου, παρουσιάζονται τα πλεονεκτήματα της χρήσης blockchain και NFT στην οικονομία και στη βιομηχανία 4.0 σε σχέση με τις υπάρχουσες λύσεις.

Λέξεις Κλειδιά:

NFT, Βιντεοπαιχνίδια, Βιομηχανία 4.0, Οικονομία, Ερευνητική μελέτη, Ψηφιακές οντότητες, Μοναδικά τεκμήρια, Τεχνολογία Blockchain, Δίκτυα Blockchain, Τεχνολογίες βιομηχανίας παιχνιδιών, Παραγωγή, Ασύγχρονη επεξεργασία, Διαλειτουργικά παιχνίδια, Κυβερνο-φυσικά συστήματα, Ενσωμάτωση ψηφιακού και φυσικού χώρου, Οντολογία, Εργαλεία ψηφιοποίησης, Έξυπνα συμβόλαια, Δίκτυο Ethereum, Γλώσσα Solidity, IPFS, Unreal Engine 5, Νευρωνικά δίκτυα, Μοντέλα διάχυσης, Ανάπτυξη πρότυπου παιχνιδιού, Συνέργεια εργαλείων, Χαρακτηριστικά παιχνιδιού, Πλεονεκτήματα Blockchain, Οικονομικά οφέλη, Οφέλη Βιομηχανίας 4.0

Abstract

Within the framework of the research study on the functions and usage scenarios of NFTs in video games, and the analysis of these systems towards the economy and Industry 4.0, the following research report was compiled.

In the theoretical part of the research and technological review, various cutting-edge technologies surrounding the project are presented. Initially, an ontological analysis towards the general concept of digital entities characterized by unique identifiers is compiled in an abstract manner. Subsequently, the representation of digital entities in video games is presented, and the various technologies and points of friction between them are briefly analyzed.

The central core of the research, examines Blockchain technology, its functions, and characteristics, delving into the use of Blockchain networks in video games. Then, advanced technologies of the video game industry, which find use in the final constructed research game, are analyzed, and their integration with Industry 4.0 systems is examined. The methods of representation and asynchronous processing of entities and the technologies supporting the creation of interoperable games sustain a major role throughout the project. Special emphasis is placed on cybernetic systems, such as cyber-physical systems, which are referenced and analyzed throughout the work and form a key connection point between the game industry and Industry 4.0. Towards the end of the theoretical part, we will see how the aforementioned technologies are combined between the digital and physical space and Industry 4.0 and its key technologies are defined.

In the practical part of the report, modern tools and technologies for the digitization and representation of digital entities are presented, as well as the technology of smart contracts and Blockchain networks. Cutting-edge tools are selected, and a prototype game is created that encompasses a large number of modern features, utilizing the functions of the Ethereum network and the Solidity language, IPFS, Unreal Engine 5, and neural networks for the creation of digital trading cards through Diffusion models. In the end, the game is presented along with the capabilities it offers through the synergy of all these individual tools and the functionalities provided by the smart contract.

Finally, by observations made throughout the project's execution, the advantages of using blockchain and NFTs in the economy and Industry 4.0 are presented, in relation to existing solutions.

Key Words:

NFTs, Videogames, Industry 4.0, Economy, Research study, Digital entities, Unique identifiers, Blockchain technology, Blockchain networks, Game industry technologies, Manufacturing, Asynchronous processing, Interoperable games, Cyber-Physical systems, Digital and physical space integration, Ontology, Digitization tools, Smart contracts, Ethereum network, Solidity language, IPFS, Unreal Engine 5, Neural networks, Diffusion models, Prototype game development, Synergy of tools, Game features, Blockchain advantages, Economy benefits, Industry 4.0 benefits

Table of Contents

Περίληψη.....	5
Abstract	7
Κεφάλαιο 1 Εισαγωγή και Προσδιορισμός.....	13
1.1 Οντολογική Ανάλυση, Προσδιορισμός και Προσέγγιση.....	13
1.2 Σκοπός της Διπλωματικής	17
1.2.1 Στόχος πρακτικού μέρους	17
1.2.2 Διατύπωση του κεντρικού ερευνητικού ερωτήματος.....	18
1.2.3 Περιγραφή των βασικών στόχων της έρευνας.....	18
1.3 Τεχνολογίες σε χρήση.....	20
1.4 Αναπαράσταση και Λειτουργικότητα των Οντοτήτων στα Βιντεοπαιχνίδια και οι Περιορισμοί αυτών	21
1.4.1 Αναπαράσταση προς τον χρήστη	21
1.4.2 Διαδικασία απεικόνισης των οντοτήτων σε ψηφιακούς κόσμους μέσα από μια μηχανή βιντεοπαιχνιδιών	22
1.4.3 Λειτουργικότητα των οντοτήτων μέσα από την μηχανή.....	23
1.5 Ψηφιοποίηση και Κυβερνητικά Συστήματα ως προς την βιομηχανία παιχνιδιών	25
1.6 Blockchain, NFT και οικονομία	26
1.7 Βιομηχανία 4.0	27
1.8 Δομή της Έρευνας	29
Κεφάλαιο 2 Μοναδικά Τεκμήρια μέσα από την τεχνολογία Blockchain και το Ethereum Network.....	31
2.1 Διαμοιρασμός δεδομένων και P2P ομότιμα δίκτυα.....	31
2.2 BLOCKCHAIN και Δυνατότητες	33
2.3 Βασικές Λειτουργίες ενός συστήματος Blockchain	36
2.4 Hashing και SHA-256	40

2.5 Δίκτυα καταναμημένων ledger με τεχνολογία Blockchain.....	42
2.6 Έξυπνη Σύμβαση/Συμβόλαιο (Smart Contract)	44
2.7 NFT και το IPFS	47
2.8 Μεταφέροντας όντα ανάμεσα σε ψηφιακούς κόσμους χρησιμοποιώντας τις τεχνολογίες Blockchain	49
2.9 Συστήματα Μεταπρογραμματισμού	50
2.10 Λειτουργίες και περιορισμοί των SMART CONTRACT για τα Βιντεοπαιχνίδια	52
2.10.1 Λειτουργίες και περιορισμοί.....	52
2.10.2 Απροσπέλαστα σημεία περιορισμού.....	53
2.11 ΤΑ ΣΤΟΙΧΕΙΑ ΕΝΟΣ BLOCKCHAIN ΣΥΣΤΗΜΑΤΟΣ ΜΕ ΛΕΙΤΟΥΡΓΙΕΣ ΜΕΤΑΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΓΙΑ ΤΑ ΒΙΝΤΕΟΠΑΙΧΝΙΔΙΑ.....	57
2.12 ETHEREUM και ETHEREUM VIRTUAL MACHINE.....	57
2.13 ΜΟΝΑΔΙΚΑ ΤΕΚΜΗΡΙΑ ΜΕΣΑ ΣΕ ΕΦΑΡΜΟΓΕΣ ΤΩΝ SMART CONTRACT	63
Κεφάλαιο 3 Μηχανές παιχνιδιών, Unreal Engine 5 και ασύγχρονες δυναμικές διαδικασίες.....	65
3.1 Μοντέρνες μηχανές βιντεοπαιχνιδιών	65
3.2 Βασικά Στοιχεία / Λειτουργίες της Unreal Engine 5.....	67
3.2.1 Προγραμματισμός στην Unreal Engine 5	67
3.2.2 Τεχνικά Δεδομένα για την Unreal Engine 5.....	69
3.2.3 Βασικά API στην Unreal Engine 5 και Βιβλιογραφία	70
3.3 Δομή παιχνιδιών	71
3.3.1 Προπαραγωγή (Pre-production).....	71
3.3.2 Παραγωγή (Production).....	71
3.3.3 Βήματα προσέγγισης κατασκευής παιχνιδιών	75
3.4 Οντότητες μέσα στην Μηχανή	78
3.5 Ο μεταπρογραμματισμός και οι δυναμικές διαδικασίες κατά την αφομοίωση νέων οντοτήτων	80

Κεφάλαιο 4 Κυβερνητικές εφαρμογές και ψηφιοποίηση αντικειμένων	88
4.1 Κυβερνητικά συστήματα	88
4.1.1 Ορισμός και εισαγωγή	88
4.1.2 Βασική δομή ενός κυβερνητικού συστήματος.....	90
4.1.3 Χρήσεις των Κυβερνητικών Συστημάτων	91
4.1.4 Οφέλη των Κυβερνητικών Συστημάτων.....	93
4.2 ΛΟΓΙΚΕΣ ΜΟΝΑΔΕΣ, ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΚΥΒΕΡΝΗΤΙΚΩΝ ΚΑΙ ΤΑ ΠΑΡΑΓΟΜΕΝΑ ΜΕΤΑΔΕΔΟΜΕΝΑ.....	94
4.3 Τα κυβερνητικά μέσα στα βιντεοπαιχνίδια.....	98
4.4 Συνδέοντας τα Δεδομένα των Κυβερνητικών Συστημάτων με Εικονικές Οντότητες στα Βιντεοπαιχνίδια και ο Ρόλος των DAO και των δικτύων Blockchain	101
Κεφάλαιο 5 Βιομηχανία 4.0 και Βιντεοπαιχνίδια	105
5.1 Βιομηχανία 4.0	105
5.1.1 Ιστορικό Πλαίσιο της Βιομηχανοποίησης και της Κατασκευής.....	105
5.1.2 Οφέλη της Βιομηχανοποίησης για την Κοινωνία και την Οικονομία	105
5.1.3 Κύρια Στοιχεία και Τεχνολογίες.....	106
5.1.4 Η Αναγκαιότητα της Βιομηχανίας 4.0 για την Κοινωνία.....	106
5.1.5 Η Συνεισφορά της Βιομηχανίας 4.0 στην Οικονομία	107
5.2 Τα συστήματα της Βιομηχανίας 4.0	108
Κεφάλαιο 6 ΠΡΑΚΤΙΚΟ ΜΕΡΟΣ	115
6.1 Παρουσίαση Κατασκευής.....	115
6.2 Κατασκευή Καρτών και Στοιχείων UI μέσα από την χρήση Γενετικής Τεχνητής Νοημοσύνης (Generative AI)	117
6.3 Κατασκευή Βιντεοπαιχνιδιού	125
6.3.1 Εργαλεία και Λειτουργικότητα	125
6.3.2 VFX και Εφέ Σωματιδίων	127

6.3.3 Πρόσθετα Ήχων.....	127
6.3.4 Στοιχεία κόσμου παιχνιδιού.....	127
6.3.5 Σύστημα διασύνδεσης παιχνιδιού με το έξυπνο συμβόλαιο.....	128
6.3.6 Επιλογή τύπου παιχνιδιού.....	131
6.3.7 Μορφοποίηση παιχνιδιού μέσα από την μηχανή Unreal Engine.....	133
6.3.8 Δημιουργία του Marketplace εντός του παιχνιδιού.....	151
6.3.9 Πώληση καρτών.....	152
6.3.10 Το κατάστημα εντός του παιχνιδιού.....	154
6.3.11 Κεντρικός πυρήνας του παιχνιδιού.....	156
6.4 Χαρακτηριστικά Καρτών και δομή JSON.....	157
6.5 Κατασκευή έξυπνου συμβολαίου.....	160
6.5.1 Ανάλυση Συμβολαίου.....	162
6.5.2 Αγοραπωλησία νομισμάτων / καρτών από τους χρήστες.....	175
6.5.3 Σύστημα επιβράβευσης «Loot» και αγοράς πακέτων καρτών.....	180
6.6 Επιπλέον λειτουργίες και επιπρόσθετος κώδικας πειραμάτων.....	183
Κεφάλαιο 7 Συμπεράσματα.....	183
Κεφάλαιο 8 Βιβλιογραφικές Αναφορές.....	187

Κεφάλαιο 1 Εισαγωγή και Προσδιορισμός

1.1 Οντολογική Ανάλυση, Προσδιορισμός και Προσέγγιση

Στον φυσικό αναλογικό χωροχρόνο ένα ον προσδιορίζεται στην ανθρωποκεντρική κλίμακα από κάποια ορισμένα σελ χαρακτηριστικών του ίδιου του όντος, τόσο φυσικών (π.χ. ύλη, αγωγιμότητα, μάζα, χώρος...), όσο και προς την χρήση (π.χ. μεταφορά πληροφορίας, αισθητική, δοχείο...), χαρακτηριζόμενο τόσο αντικειμενικά όσο και υποκειμενικά μέσα από ιδεαλιστικά κριτήρια. Αντίθετα, στον ψηφιακό χώρο ύπαρξης τα αντικείμενα δεν φέρουν τα διαστατικά χαρακτηριστικά της χωρικής διάστασης, φέροντας ούτε σώμα (χώρο) ούτε μετάλλαξη (χρόνο), αποτελώντας μόνο σύμβολα προς τα όντα του φυσικού χώρου.

Όταν τα φυσικά όντα χαρακτηρίζονται από το στιγμιαίο σώμα τους ή το συσσωμάτωμά τους μέσα από την πάροδο του χρόνου, τα ψηφιακά όντα χαρακτηρίζονται από την τέλεια και αδιάβλητη ακολουθία δυαδικών που κατά την επεξεργασία τους φέρουν πίσω τον συμβολισμό του υποτιθέμενου σώματός τους, φέροντας άπειρη αναπαραξιμότητα.

Η συνέργεια (ως synergy) των ιδιοτήτων που προσφέρει ο ψηφιακός χώρος και η αιτιότητα και συνέχεια της ύλης στον φυσικό χώρο επιτρέπουν την ζευγοποίηση παρατηρήσιμων όντων με μοναδικά ψηφιακά τεκμήρια που συμβολίζουν και αποθηκεύουν αναφορές προς την φύση/φυσιολογία των φυσικών όντων, όπως και επιτρέπουν την παρακολούθηση των φυσικών σωμάτων και την πρόσβαση μετα-πραγματικών ιδιοτήτων σε αυτά.

Τα ψηφιοποιημένα όντα εμπεριέχουν ένα σύνολο από δεδομένα που αντιστοιχούν σε χαρακτηριστικά του όντος εν ψηφιοποίηση. Αυτά τα δεδομένα μπορεί να είναι είτε ένα ψηφιακός τρισδιάστατος κλώνος ή συμβολισμοί μέσα από λέξεις, και το σύνολο των δεδομένων είναι συγκεκριμένο και απόλυτα διατυπωμένο σε bit μέσα στην μνήμη. Επειδή όμως τα δεδομένα αυτά φέρουν πολλές φορές έναν μεγάλο όγκο, επεξεργάζονται με πεπερασμένους αλγόριθμους όπως HASH functions και παράγεται ένα μοναδικό τελικό κλειδί ανά αρχείο, που μπορεί να θεωρηθεί ως ένας τελικός απόλυτος χαρακτηρισμός του ψηφιακού όντος και ένα δεύτερο, σε σειρά, σύμβολο / απόλυτος χαρακτηρισμός του «σώματος» των δεδομένων. Αλλιώς, η διεύθυνση αποθήκευσης τους μέσα στην μνήμη αποτελεί το σημείο κλήσης τους και έμμεσα το αναγνωριστικό τους σαν δεδομένα.

Κάθε διαφορετικό σύνολο χαρακτηρισμών ή μετρήσεων φέρει ένα διαφορετικό σύνολο δεδομένων, παράγοντας ένα διαφορετικό HASH και έχοντας μια διαφορετική θέση στην μνήμη του υπολογιστή. Τα ψηφιακά αυτά «αντικείμενα», θεωρούνται μοναδικά και καλούνται ανεξάρτητα από τα διάφορα προγράμματα που τα επεξεργάζονται.

Κάθε ξεχωριστά διακριτό ον επί του φυσικού χώρου, το οποίο μπορεί να παρατηρηθεί και να μετρηθεί, αν δεν βρίσκεται σε κβαντικό εναγκαλισμό ή δεν αποτελεί μέρος ενός μορφοκλασματικού συνόλου, χαρακτηρίζεται ως μοναδικό σαν σώμα. Τα όργανα που το «μετρούν» αποτελούν αισθητήρια καθώς και όργανα μέτρησης και ο βαθμός χαρακτηρισμού κρίνεται τόσο από την ακρίβεια των οργάνων μέτρησης, όσο και από την φύση του τελικού παρατηρητή και την ικανότητα παρατήρησης, κατανόησης και διατύπωσης αυτού.

Επί την παρατήρηση του, δημιουργούνται δύο όντα, ένα το σώμα εν παρατήρηση την στιγμή της παρατήρησης και ένα ψηφιακό (ον) που αποτελεί τους χαρακτηρισμούς αυτού. Λόγω αυτής τη αναπόφευκτης ασυνέχειας αυτών των δύο, απαιτούνται συστήματα εναρμόνισης και επαναψηφιοποίησης ή ενεργής παρακολούθησης του φυσικού όντος, για να βρίσκεται η ψηφιακή συμβολική αναφορά προς το φυσικό σώμα σε συνοχή με την κατάσταση του φυσικού όντος. Αυτά τα συστήματα, φέρουν χαρακτηριστικά των κυβερνητικών και αποτελούν κυβερνο-φυσικά συστήματα.

Αυτές οι ψηφιοποιημένες οντότητες αποτελούν ολογραφήματα της φυσικής πραγματικής μορφής, αλλά όντας ένα ξεχωριστό ον το καθένα από αυτά, μπορούν να εκτείνουν την φύση τους έξω από το να αποτελούν απλές απεικονίσεις. Διότι, εφόσον μια ψηφιακή οντότητα μορφοποιηθεί, ερχόμενη είτε από τον φυσικό χώρο σαν ένα ψηφιακό δίδυμο, είτε αποτελεί παράγωγο ψηφιακών λογικών αλγορίθμων και διαδικασιών (procedural generation), η ίδια (η ψηφιακή οντότητα) αποτελεί ένα μεμονωμένο ον σε κάθε instance (αντίγραφο) αυτής, και μπορεί να δεχτεί τροποποιήσεις προς μια νέα πραγματικότητα. Έτσι, το ίδιο το ψηφιακό ον μπορεί να αρχίσει να τροποποιείται λογικά ή να χρησιμοποιείται σαν στοιχείο ενός συσσωματώματος εντός του ψηφιακού κόσμου, αποκτώντας μια μορφή μετα-πραγματικότητας.

Αυτός ο ψηφιακός κόσμος, καθώς και οι νέες ψηφιακές οντότητες εντός αυτού είναι περιορισμένες υπαρξιακά από την φύση και την «φυσιολογία» του υπαρξιστικού πλαισίου στο οποίο εγκλείονται. Όπως η φυσιολογία και η ψυχή στα γραπτά του Πλάτωνα (Τίμαιος, Πολιτεία) είναι αλληλένδετα στοιχεία και περιορίζουν -ενώ παράλληλα δομούν- το ένα τη φύση του άλλου,

και όπως ο φυσικός χώρος επηρεάζει τα ίδια στοιχεία μορφολογικά, λειτουργικά, υπαρξιακά και φυσικά, έτσι περιορίζονται και δομούνται τα ψηφιακά όντα από τα συστήματα εκ των οποίων προκύπτουν και αποτελούν abstractions (αφηρημένες έννοιες) αυτών. Ταυτόχρονα, το όλον του ψηφιακού κόσμου δομείται από τα όντα τα οποία καθορίζει και μεταφράζει, από μορφή ψηφιακών δεδομένων, σε έναν ανθρωποκεντρικό συμβολισμό αυτών.

Αυτές οι μεταφράσεις, η φύση του εικονικού κόσμου, η φυσιολογία, οι προβαλλόμενες οντότητες, οι ψηφιακοί κόσμοι δεν είναι τυχαία κατασκευάσματα, αλλά λογικές πράξεις, προκαθορισμένες από τους συνειδητούς δημιουργούς της μηχανής που δημιουργεί τις «κοσμικές» αυτές εξομοιώσεις, φανταστικών μη αιτιωδών (Acasual) κόσμων με παραφυσικά χαρακτηριστικά. Αυτές οι μηχανές λέγονται «μηχανές βιντεοπαιχνιδιών» και δομούν τον χώρο «παιξίματος» ή χώρο παιχνιδιού (game world) και όλη την φυσική και τα όντα που εμπεριέχει, και το τελικό κατασκεύασμα αποτελεί ένα βιντεοπαιχνίδι.

Στα βιντεοπαιχνίδια τα μοναδικά αυτά όντα χαρακτηρίζονται με σταθερές μέσα σε Enumerations και Structures, έχοντας συγκεκριμένες θέσεις στην μνήμη μέσα σε πίνακες (arrays). Οι θέσεις αυτές και η ζευγοποίηση παίρνει μέρος στην, προαναφερθείσα μηχανή του βιντεοπαιχνιδιού, όπως η Unreal Engine 5 (Fortnite) και η Rage Engine (Grand Theft Auto). Η ίδια η μηχανή αποτελεί μια σειρά από λειτουργίες (functions) που κατασκευάζουν τον ψηφιακό χώρο και το σύνολο ενός βιντεοπαιχνιδιού ή μίας ψηφιακής διάστασης, μέσα από ένα σύνολο από πολλά ξεχωριστά ψηφιακά όντα με ειδικές λειτουργίες φτιαγμένες γύρω από την δυνατότητα αλληλεπίδρασης των όντων μεταξύ τους και του ανθρώπου προς τα όντα αυτά.

Μέσα από τις λογικά διατυπωμένες αλληλεπιδράσεις ή εκείνες που εύλογα προκύπτουν μέσα από αυτόματες διαδικασίες τα ψηφιακά όντα βρίσκουν μετάλλαξη και συνέχεια, αλλάζοντας το σώμα τους, ενώ παραμένουν εγκλωβισμένα εντός των πλαισίων του ψηφιακού κόσμου. Για την μεταφορά των όντων από έναν ψηφιακό κόσμο σε έναν δεύτερο κόσμο, είτε φυσικό είτε ψηφιακό, υπάρχουν προαπαιτήσεις και ένα πλαίσιο συστημάτων που καθιστά την μεταφορά αυτή εφικτή.

Τα ψηφιακά όντα που κρίνονται κατάλληλα για μεταφορά πρέπει να συμβολίζονται από ένα μοναδικό διακριτικό, καταχωρημένο σε μια κοινή βάση δεδομένων που λειτουργεί ως ένα μητρώο οντοτήτων. Καθώς και να ακολουθούν μια κοινή δομή χαρακτηρισμού και να

εμπεριέχουν ένα κοινό πλαίσιο λειτουργικών χαρακτηριστικών, κάνοντας εφικτή την «διαλειτουργικότητα» των λογικών στοιχείων της οντότητας και την απεικόνιση του.

Συγκεκριμένα, ο όρος «διαλειτουργικότητα» αναφέρεται σε χαρακτηριστικά που επιτρέπουν σε διάφορα συστήματα, πλατφόρμες ή προϊόντα να λειτουργούν μαζί ή να συνεργάζονται αποτελεσματικά, προωθώντας την ομαλή ανταλλαγή και χρήση πληροφοριών ή δεδομένων μεταξύ τους. Μια δυνατότητα σημαντική για τις λειτουργίες των συστημάτων της Βιομηχανίας 4.0.

Μέσα από αυτή την διαλειτουργικότητα, είναι εφικτή η μεταφορά μεταλλαγμένων οντοτήτων που μεταφέρουν στοιχεία και χαρακτηριστικά μαζί τους καθώς ταξιδεύουν ανάμεσα στους κόσμους των βιντεοπαιχνιδιών. Δίνοντας μια νέα φύση στην χρήση, λειτουργικότητα και σημασία των μεμονωμένων αυτών όντων, προς τον άνθρωπο χρήστη – παρατηρητή.

Είναι, όμως, εφικτή και η αλληλεπίδραση των όντων αυτών, που μπορεί να έχουν λάβει έναν σημαντικό βαθμό μετάστασης και να κουβαλούν έναν μεγάλο όγκο πληροφορίας μαζί τους, πίσω με τον φυσικό χώρο. Όπως τα ίδια μπορεί να εισήλθαν στον υπερφυσικό ψηφιακό χώρο μέσα από κυβερνο-φυσικά συστήματα ψηφιοποίησης – έτσι και μπορούν να αλληλεπιδράσουν ξανά με τον φυσικό χώρο μέσα από αντίστοιχα συστήματα ή και να επιστρέψουν πίσω σε αυτόν μέσα από ρομπότ και άλλες εφαρμογές αιχμής στον χώρο της μηχανικής, κυβερνητικά, ρομποτικής.

1.2 Σκοπός της Διπλωματικής

Σκοπός της διπλωματικής είναι η κατασκευή μιας ψηφιακής εμπειρίας, στην μορφή ενός βιντεοπαιχνιδιού τύπου παιχνιδιού συλλογής και ανταλλαγής καρτών. Οι κάρτες του παιχνιδιού θα αποτελούν ψηφιοποιημένες οντότητες με μοναδικά τεκμήρια σε μορφή νομισμάτων σε κάποιο δίκτυο Blockchain και θα φέρουν χαρακτηριστικά διατυπωμένα μέσα στα μεταδεδομένα τους στο IPFS. Παράλληλα με το βιντεοπαιχνίδι θα δημιουργηθεί ένα marketplace που θα επιτρέπει την συναλλαγή των καρτών αυτών.

Μέσα από τον σχεδιασμό και την κατασκευή του πρακτικού μέρους, θα απαντηθεί το κεντρικό ερευνητικό ερώτημα, εντοπίζοντας τα σενάρια χρήσης παρόμοιων συστημάτων και του πλεονεκτήματος εφαρμογής αυτών πάνω από τις υπάρχουσες λύσεις.

1.2.1 Στόχος πρακτικού μέρους

Δημιουργία ψηφιακών καρτών για παιχνίδι συλλογής και ανταλλαγής καρτών όπου θα βρίσκονται μέσα στο blockchain ως tokens. Οι κάρτες αυτές θα έχουν συγκεκριμένα χαρακτηριστικά (stats) και ιδιότητες. Στην συνέχεια και εφόσον υπάρχει η δυνατότητα, θα δημιουργηθεί ένα marketplace για τις κάρτες.

Ανάλυση των υποσυστημάτων του πρακτικού μέρους και των βασικών απαιτούμενων στοιχείων αυτών:

- Μηχανή Βιντεοπαιχνιδιών: απαραίτητη για την κατασκευή του παιχνιδιού.
 - Δυνατότητα Επικοινωνίας με πορτοφόλι διαφύλαξης κρυπτονομισμάτων ή tokens.
 - Δυνατότητα κατασκευής παιχνιδιού συλλογής και ανταλλαγής καρτών.
 - Δυναμικές δυνατότητες για τα συστήματα δικτύου και την επεξεργασία δεδομένων σε πραγματικό χρόνο.
 - Δυνατότητες ασύγχρονων υπολογιστικών διαδικασιών
- Βιντεοπαιχνίδι: το κατασκευασμένο παιχνίδι εκ της μηχανής.
 - Tokens ως Κάρτες
 - JSON-RPC talk για την επικοινωνία με τον Blockchain
 - Πρόσβαση σε smart contracts
 - Εντοιχισμένο Marketplace
- Έξυπνο Συμβόλαιο (Smart Contract):

- Εναρμονισμένο με τα συστήματα του παιχνιδιού
- Λειτουργίες Marketplace
- Διαχείριση χρηστών
- Διαχείριση στοιχείων βάσης δεδομένων

1.2.2 Διατύπωση του κεντρικού ερευνητικού ερωτήματος

Ποια είναι τα πλεονεκτήματα της χρήσης blockchain και NFT στην οικονομία και στη βιομηχανία 4.0 σε σχέση με τις υπάρχουσες λύσεις;

Αυτό το κεντρικό ερώτημα οδηγεί σε μια σειρά υποερωτημάτων:

- ❖ Τι ψηφιοποιείται σαν NFT και πως;
- ❖ Ποιες είναι οι δυνατότητες των NFT μέσα από τα Blockchain;
- ❖ Ποιες οικονομικές ευκαιρίες και χρήσεις προκύπτουν μέσα από την ένταξη NFT σε ψηφιακούς κόσμους και πως συγκρίνονται αυτά με τις υπάρχουσες λύσεις;
- ❖ Ποια είναι η επίδραση συστημάτων με NFT οντότητες στη βιομηχανία 4.0 και ποια η διαφορά τους με τις υπάρχουσες λύσεις;

1.2.3 Περιγραφή των βασικών στόχων της έρευνας

Οι βασικοί στόχοι της παρούσας έρευνας είναι οι εξής:

- ❖ **Τεχνολογική Ανάλυση:** Να πραγματοποιηθεί μια ουσιαστική παρουσίαση των διάφορων τεχνολογιών που κρίθηκαν καίριες για την χρήση μοναδικών τεκμηρίων σε ψηφιακές εμπειρίες μέσα στα πλαίσια της Βιομηχανία 4.0 και να παρουσιαστεί μια σαφής και συνοπτική εικόνα των βασικών εννοιών και εργαλείων της διαθέσιμης τεχνολογίας, όπως και η εφαρμογή τους στην Βιομηχανία.
- ❖ **Προσδιορισμός Απαιτήσεων και Προβλημάτων:** Να προσδιοριστούν τα καθοριστικά στοιχεία που δομούν ένα σύστημα που εμπεριέχει τις προαναφερόμενες τεχνολογίες και έμμεσα ή άμεσα οι τεχνολογικοί τους περιορισμοί.
- ❖ **Εύρεση Εργαλείων:** Να βρεθούν τεχνολογίες αιχμής για την πλατφόρμα του κατασκευαστικού μέρους και να παρουσιαστεί η λειτουργία τους.

- ❖ **Κατασκευή:** Να κατασκευαστεί ένα βιντεοπαιχνίδι με τεχνολογίες αιχμής που φέρει πλήρης χρήση των δυνατοτήτων που προσφέρουν τα Smart Contracts (έξυπνα συμβόλαια).
- ❖ **Εξερεύνηση Νέων Ευκαιριών και Σύγκριση με υπάρχουσες λύσεις:** Να εξερευνήσει τις νέες ευκαιρίες που προκύπτουν από την συνεργασία των δικτύων Blockchain με τα βιντεοπαιχνίδια σε σύγκριση με τις υπάρχουσες λύσεις.

1.3 Τεχνολογίες σε χρήση

Διερευνήθηκε η χρήση της Unreal Engine 5 ως μηχανή βιντεοπαιχνιδιών και εξετάστηκαν τα έξυπνα χαρακτηριστικά που εντοχίζει για την δημιουργία διαλειτουργικών στοιχείων παιχνιδιού. Ταυτόχρονα διερευνήθηκε το αν και κατά πόσο είναι εφικτή η δημιουργία προγραμμάτων με δυναμικά ασύγχρονα στοιχεία και λειτουργίες μεταπρογραμματισμού. Για τις οντότητες / κάρτες χρησιμοποιήθηκε μια μίξη συστημάτων Generative AI (Γενετική Τεχνητή Νοημοσύνη) μαζί με ένα ευρύ φάσμα μοντέλων από την ιστοσελίδα huggingface.co στο framework Stable Diffusion από την Stability AI.

Για τα μοναδικά τεκμήρια χρησιμοποιήθηκε το δίκτυο κρυπτονομισμάτων του Ethereum και δημιουργήθηκαν πολλαπλά δοκιμαστικά έξυπνα συμβόλαια και ένα τελικό στην γλώσσα Solidity. Η ανάπτυξη έγινε μέσα στο Remix IDE και έγινε deploy μέσα σε test-network της Ethereum, το Sepolia με ChainID 11155111.

Χρησιμοποιήθηκε το Διαπλανητικό Σύστημα Αρχείων IPFS (InterPlanetary File System) για την φύλαξη των στοιχείων των καρτών (tokens).

Η εργασία της επεξεργασίας των εικόνων (photomanipulation), από το Stable Diffusion, πήρε μέρος στο Adobe Photoshop και τα βίντεο κατασκευάστηκαν στο Vegas Pro 18.0 της Magix.

1.4 Αναπαράσταση και Λειτουργικότητα των Οντοτήτων στα Βιντεοπαιχνίδια και οι Περιορισμοί αυτών

1.4.1 Αναπαράσταση προς τον χρήστη

Μια οντότητα εντός του χώρου του βιντεοπαιχνιδιού μπορεί να αποτελεί οποιοδήποτε μεμονωμένο στοιχείο του διαδραστικού χώρου ή να συμβάλλει στις αλληλεπιδράσεις αυτού. Αυτό επιτρέπει στους δημιουργούς του ψηφιακών, αυτών, κόσμων την ελευθερία της έκφρασης και έναν πολύ μεγάλο βαθμό δημιουργικότητας, εξού και οι ψηφιακοί αυτοί χώροι συχνά εμπεριέχουν υπερφυσικά και φανταστικά χαρακτηριστικά. Για παράδειγμα, ένας θάμνος που είναι τοποθετημένος σε μία πίστα μπορεί να απαρτίζει μια οντότητα επειδή μπορεί να μεταλλάσσεται σε μια πέτρα όταν οι συνθήκες ενεργοποίησης αυτής της δράσης πληρούνται, αντίθετα οι υπόλοιποι θάμνοι στην πίστα αποτελούν αδιάσπαστο κομμάτι του εδάφους ή της υπόλοιπης ενιαίας βλάστησης. Σε αυτό το παράδειγμα ο θάμνος αποτελεί ένα οντόν του, τόσο λόγω παρατήρησης, όσο και λόγω αλληλεπίδρασης και διάδρασης, μέσα από την σημαντικότητα του προς το σενάριο της ιστορίας.

Μέσα στα πλαίσια των φανταστικών, αυτών, κόσμων, όπου όλα είναι εφικτά και οι δράσεις και αντιδράσεις δεν απαιτούν μια φυσική συνέχεια (κανονικότητα), τα όντα μπορούν να πάρουν ποικίλες μορφές, ακόμα και ασαφούς φύσης. Όμως υπάρχουν περιορισμοί κατά την δημιουργία εμπορικών προϊόντων βιντεοπαιχνιδιών, καθώς τα παιχνίδια εμπεριέχουν τόσο ένα σύνολο οντοτήτων, όσο και μια συνεχή ιστορία η οποία στηρίζεται και πλαισιώνεται μέσα από την χρήση οντοτήτων. Η ιστορία ενός παιχνιδιού πρέπει να είναι κατανοητή με το τελικό καταναλωτικό κοινό του βιντεοπαιχνιδιού και πρέπει να μπορεί να διαβιβάσει ξεκάθαρα το νόημα και το μήνυμα της ιστορίας, δημιουργώντας περιορισμούς ως προς τις πιθανές απεικονίσεις και τα νοήματα που μπορούν να φέρουν τα μεμονωμένα όντα.

Η αντιληπτική ικανότητα του παρατηρητή, όμως, δεν αποτελεί τον μόνο περιορισμό, καθώς τα ίδια τα συστήματα γραφικής απεικόνισης των οντοτήτων φέρουν περιορισμένες δυνατότητες γραφικά, λογικά και μέσα από το υλισμικό (hardware).

1.4.2 Διαδικασία απεικόνισης των οντοτήτων σε ψηφιακούς κόσμους μέσα από μια μηχανή βιντεοπαιχνιδιών

Απεικονιστικά, από την μεριά της μηχανής και όχι του τελικού παρατηρητή, οι οντότητες στα βιντεοπαιχνίδια αποτελούνται από συγκεκριμένα δομημένα αρχεία που ορίζουν με ακρίβεια τα οπτικά τους χαρακτηριστικά και κώδικα που περιγράφει με απόλυτη λεπτομέρεια όλες τις συμπεριφορές και τις αλληλεπιδράσεις τους μέσα στο περιβάλλον του παιχνιδιού.

Η κατασκευή του οπτικού μέρους, αυτών των οντοτήτων, ξεκινά με την ανάπτυξη ενός σκελετικού πλέγματος (skeletal mesh) – μίας εσωτερικής δομικής βάσης που μοιάζει με τον εσωτερικό σκελετό ενός έμβιου οργανισμού, καθορίζοντας το θεμελιώδες σχήμα και επιτρέποντας την άρθρωση (articulation). Στη συνέχεια, το πραγματικό mesh (πλέγμα), που αποτελεί την τελική μορφή της ψηφιακής οντότητας, σχεδιάζεται και συνδέεται σε αυτόν τον σκελετό, σχηματίζοντας την ορατή επιφάνεια της οντότητας μέσα από τρίγωνα (ή και πολύγωνα).

Αφού ολοκληρωθεί η «σκελετική δομή» και το mesh (πλέγμα), οι animators δημιουργούν τα animations που επιτρέπουν στην οντότητα να εκτελεί δυναμικές κινήσεις. Η χρήση του σκελετού ως βάση για την παροχή κίνησης επιτρέπει την δημιουργία πολύπλοκων και ρεαλιστικών κινήσεων που δεν θα ήταν αλλιώς εφικτές. Ακολουθώς, εφαρμόζονται textures και υλικά (materials) στο mesh (πλέγμα), ζωγραφίζοντας πάνω από το τρισδιάστατο μοντέλο, προσθέτοντας χρώμα και φυσικές ιδιότητες της επιφάνειας, δίνοντας στην οντότητα μια ζωντανή και αρμονική εμφάνιση σε σχέση με τα υπόλοιπα στοιχεία του παιχνιδιού.

Έπειτα, για την τελική απεικόνιση της οντότητας στην οθόνη, ακολουθεί η διαδικασία του rendering. Η οποία διαδικασία περιλαμβάνει όχι μόνο την απεικόνιση του mesh (πλέγματος) με τα textures, αλλά και εξελιγμένους γραφικούς αλγόριθμους για την ενσωμάτωση ρεαλιστικού φωτισμού, σκιών και άλλων οπτικών εφέ, εξασφαλίζοντας ότι οι οντότητες φαίνονται φυσικά τοποθετημένες μέσα στον κόσμο του παιχνιδιού. Σε ένα δεύτερο μέρος, προγραμματιστικές συναρτήσεις και πολύπλοκα συστήματα της μηχανής παιχνιδιών καθορίζουν τον τρόπο με τον οποίο αυτές οι οντότητες αλληλεπιδρούν με το περιβάλλον και ανταποκρίνονται στις ενέργειες του παίκτη, συμβάλλοντας στην παροχή μίας εξαιρετικά εμπυθιστικής εμπειρίας για τον τελικό χρήστη.

Εν τέλει, η ορθή κατασκευή αυτών των δύο ξεχωριστών λειτουργιών (του rendering και του scripting) είναι σημαντική για την ορθή οπτική απεικόνιση των οντοτήτων στα βιντεοπαιχνίδια και θέτει τα όρια της διαδικασίας της οπτικής αναπαράστασης.

1.4.3 Λειτουργικότητα των οντοτήτων μέσα από την μηχανή

Έξω από τα πλαίσια της ορθής και κατανοητής αναπαράστασης, ο βαθμός εμπύθισης και αλληλεπίδρασης σε ένα περιβάλλον παιχνιδιού – κόσμος παιχνιδιού καθορίζεται από τις πιθανές αλληλεπιδράσεις που μπορούν να πάρουν μέρος ανάμεσα στα διάφορα μεμονωμένα όντα του παιχνιδιού. Ταυτόχρονα το πλήθος και το εύρος αυτών των αλληλεπιδράσεων καθορίζει και το βαθμό ελευθερίας του χρήστη μέσα στον χώρο του βιντεοπαιχνιδιού.

Για αυτές τις αλληλεπιδράσεις, υφίστανται σημαντικά συστήματα, όπως τα spawning systems (σύστημα τοποθέτησης / εισόδου της οντότητας στον χώρο του παιχνιδιού), η μηχανή φυσικής και τα AI συστήματα, τα οποία χρησιμοποιούνται για την τέλεση των δυναμικών και ρεαλιστικών αλληλεπιδράσεων μεταξύ των οντοτήτων. Επιπλέον, κύριο ρόλο στην αλληλεπίδραση των οντοτήτων μεταξύ τους και με τον παίκτη, αποτελούν και τα collisions, triggers ή scripted events, που είναι θεμελιώδη για τη δημιουργία μιας ελκυστικής εμπειρίας παιχνιδιού. Για παράδειγμα, ένας χαρακτήρας του παίκτη μπορεί να αλληλεπιδράσει με ένα σεντούκι θησαυρού και να ενεργοποιήσει ένα event (περιστατικό) στο οποίο το σεντούκι ανοίγει, αποκαλύπτοντας αντικείμενα που μπορεί να είναι απαραίτητα για την πρόοδο της ιστορίας του παιχνιδιού.

Αυτά τα αντικείμενα / όντα που φέρουν δυνατότητες αλληλεπίδρασης προσφέρουν λειτουργικότητα προς το όλον του παιχνιδιού και είναι σημαντικά σε πολλαπλές πτυχές του gameplay. Μπορούν να λειτουργήσουν ως εμπόδια, εργαλεία ή ανταμοιβές/κλειδιά μπορεί να ξεκλειδώνουν πόρτες, όπλα να χρησιμοποιούνται για μάχη ή health packs (ζωές) να αποκαθιστούν την υγεία του παίκτη. Οι οντότητες λειτουργούν επίσης, ως στοιχεία αφήγησης, ιδιαίτερα στα παιχνίδια ρόλων (role-playing games / RPGs), όπου μπορεί να κινούν την πλοκή ή να αποκαλύπτουν σημαντικά στοιχεία του παρασκηνίου. Οι περιβαλλοντικές αλληλεπιδράσεις μέσω οντοτήτων (όπως πόρτες, διακόπτες και μοχλοί), συχνά απαιτούν από τον παίκτη να ασχοληθεί με την επίλυση γρίφων ή την εξερεύνηση προκειμένου να προχωρήσει στο παιχνίδι. Επιπλέον, πολλά RPGs διαθέτουν οικονομικά συστήματα όπου οι οντότητες μπορούν να

ανταλλάσσονται, επηρεάζοντας την οικονομία του παιχνιδιού και προσθέτοντας ένα στρώμα στρατηγικής στην εμπειρία του παίκτη.

Ταυτόχρονα, οι οντότητες μπορούν να συμβάλλουν σημαντικά στην ανάπτυξη του χαρακτήρα και στην ευρύτερη αφήγηση. Για παράδειγμα, αντικείμενα όπως πανοπλίες και όπλα όχι μόνο βελτιώνουν τα στατιστικά του χαρακτήρα, αλλά μπορούν επίσης να ενισχύσουν τις ικανότητές του, επηρεάζοντας την πρόοδό του και την αποτελεσματικότητά του στο παιχνίδι. Αυτά τα αντικείμενα συνδέονται συχνά με quests ως στόχοι ή ανταμοιβές, καθιστώντας τα αναπόσπαστα στοιχεία για την ολοκλήρωση των αποστολών και την παρακίνηση του παίκτη. Η συμπερίληψη διαφόρων αντικειμένων (όπως βιβλία, πινακίδες και αγάλματα) εμπλουτίζει την δομή και αρχιτεκτονική του κόσμου, παρέχοντας βάθος και πλαίσιο στη μυθολογία του παιχνιδιού και ενθαρρύνοντας την πιο βαθιά εξερεύνηση.

Συνολικά, οι οντότητες και τα αντικείμενα δεν είναι απλώς συμβολικές απεικονίσεις και λειτουργικά στοιχεία μέσα σε ένα παιχνίδι, αλλά είναι θεμελιώδη για τη δημιουργία ενός κόσμου παιχνιδιού που είναι απτός, διαδραστικός και ανταποκρίνεται στις ενέργειες του παίκτη. Μέσα από τις αλληλεπιδράσεις με τα διάφορα όντα, οι παίκτες νιώθουν ότι οι ενέργειές τους έχουν αντίκτυπο και οι ίδιοι αποτελούν αναπόσπαστο μέρος της εξελισσόμενης ιστορίας, ενισχύοντας έτσι την αφοσίωση και τη εμπύθιση τους στον εικονικό κόσμο. Αυτή η πλούσια αλληλεπίδραση των οντοτήτων μέσα στο περιβάλλον του παιχνιδιού εντείνει τη σημασία τους στην παροχή μιας ολοκληρωμένης και ουσιαστικής εμπειρίας.

Εν κατακλείδι, ο τρόπος απεικόνισης, των μορφολογικά ιδιαίτερων, και μη, οντοτικών δομών, μαζί με τον βαθμό πιθανής αλληλεπίδρασης του όντος με τον χρήστη, και των οντοτήτων μεταξύ τους αποτελούν κεντρικά στοιχεία που καθορίζουν τον βαθμό εμπύθισης και αλληλεπίδρασης σε ένα περιβάλλον παιχνιδιού – κόσμο παιχνιδιού. Ιδιαίτερα όταν το όλο του ψηφιακού χώρου απαρτίζεται από αυτά τα όντα.

1.5 Ψηφιοποίηση και Κυβερνητικά Συστήματα ως προς την βιομηχανία παιχνιδιών

Ο κλάδος των βιντεοπαιχνιδιών και ο κλάδος των Κυβερνητικών εντός και εκτός της Βιομηχανίας 4.0 είναι αδιάρρηκτα συνδεδεμένοι, καθώς τα βιντεοπαιχνίδια χρησιμοποιούν ένα μεγάλο πλήθος κυβερνητικών συστημάτων και συστημάτων που ακολουθούν τις αρχές των κυβερνητικών, για την λειτουργία τους αλλά και συνδέονται με υπάρχοντα συστήματα της Βιομηχανίας ή εμπλουτίζουν τα προϊόντα αυτής. Αυτό γίνεται περισσότερο εμφανές από την άνοδο των συστημάτων Επαυξημένης Πραγματικότητας AR, τα οποία ενεργά εμπεριέχουν κυβερνητικά συστήματα στον πυρήνα των βασικών λειτουργιών τους και βρίσκουν εφαρμογή στον φυσικό χώρο όπου βρίσκονται τα παράγωγα την Βιομηχανίας.

Ένα βασικό στοιχείο των κυβερνητικών είναι η ψηφιοποίηση δεδομένων και οντοτήτων προς χρήση από τα λογικά μέρη ενός κυβερνητικού συστήματος και αποτελεί κύριο σημείο τριβής ανάμεσα στην Βιομηχανία 4.0 και τα βιντεοπαιχνίδια. Μια διαδικασία που αποτελεί βασικό στοιχείο των λειτουργιών αρκετών παιχνιδιών.

Συγκεκριμένα, η ψηφιοποίηση στα κυβερνητικά συστήματα ενισχύει την αποδοτικότητα των λειτουργιών σε διάφορους τομείς, κάνοντας δυνατή την επεξεργασία δεδομένων σε πραγματικό χρόνο, την δυνατότητα παρακολούθησης των συστημάτων και την ακριβή ρύθμιση μέσα από αυτοματοποιημένους ελέγχους, απαρτίζοντας κεντρικό ρόλο στις διαδικασίες των κυβερνητικών συστημάτων. Στη βιομηχανία, αυτές οι δυνατότητες διασφαλίζουν τον ακριβή έλεγχο των λειτουργιών των μηχανημάτων και την ποιότητα των τελικών παραγόμενων προϊόντων, ενώ στη βιομηχανία των παιχνιδιών, παρόμοιες τεχνολογίες, διευκολύνουν τη δημιουργία προσαρμοστικών δυναμικών εικονικών κόσμων, ή και αντίστροφα. Αυτή η συνέργεια μεταξύ της τεχνολογικής προόδου στην βιομηχανία και στην τεχνολογία παιχνιδιών είναι ιδιαίτερα εμφανής καθώς η πρόοδος σε έναν τομέα συχνά συμπληρώνει και βελτιώνει τον άλλο. Για παράδειγμα, η χρήση προηγμένων αισθητήρων και συστημάτων ανατροφοδότησης πραγματικού χρόνου σε βιομηχανικές εφαρμογές θυμίζει τα μοντέλα αλληλεπίδρασης στα βιντεοπαιχνίδια και επιτρέπει, τόσο την σύνδεση αυτών των δύο, όσο και την ένταξη τεχνολογιών από τον έναν κλάδο στον άλλον. Επιπλέον, η ανάπτυξη νέων συστημάτων ανατροφοδότησης και αισθητηρίων μπορεί να οδηγήσει στην δυνατότητα δημιουργίας νέων, περισσότερο εμπυθιστικών εμπειριών στον κλάδο των βιντεοπαιχνιδιών. Αυτή η διασυνδεδετική χρήση της τεχνολογίας δείχνει όχι μόνο την

πολυμορφία των κυβερνητικών συστημάτων, αλλά επίσης ενισχύει την ανάπτυξη προϊόντων και τη συμμετοχή των χρηστών σε διάφορες πλατφόρμες.

Επιπλέον, η άνοδος των διαλειτουργικών (interoperable) στοιχείων (assets) παιχνιδιού στα παιχνίδια επιδεικνύει πώς οι καινοτομίες της Βιομηχανίας 4.0 εντοιχίζονται από την βιομηχανία παιχνιδιών. Η διαλειτουργικότητα στα παιχνίδια επιτρέπει τη χρήση περιουσιακών στοιχείων από ένα παιχνίδι σε άλλο, δημιουργώντας ένα ενιαίο οικοσύστημα. Αυτή η έννοια αντικατοπτρίζει πρακτικές στον βιομηχανικό τομέα, όπου η τυποποίηση σε διάφορα συστήματα επιτρέπει την αλληλεπίδραση διαφόρων συστημάτων με αποτελεσματικό τρόπο, βελτιστοποιώντας τη συνολική λειτουργικότητα της γραμμής παραγωγής. Στα βιντεοπαιχνίδια, αυτή η διαλειτουργικότητα μπορεί να επεκταθεί πέρα από τον ψηφιακό κόσμο, ενσωματώνοντας πραγματικές υπηρεσίες και προϊόντα. Για παράδειγμα, μια στολή χαρακτήρα που σχεδιάζεται σε ένα παιχνίδι θα μπορούσε να κατασκευαστεί ως πραγματικό ένδυμα, «σπάζοντας τον 4^ο τοίχο» και αφαιρώντας τα όρια μεταξύ εικονικών δημιουργιών και φυσικών προϊόντων. Αυτή η σύνδεση όχι μόνο εμπλουτίζει τις εμπειρίες των παιχνιδιών αλλά επίσης ανοίγει νέες ευκαιρίες για διαφημιστικές στρατηγικές και ενισχύει την αλληλεπίδραση του καταναλωτή με τα παραγόμενα αγαθά, συνδέοντας εικονικά περιουσιακά στοιχεία με απτά αγαθά.

1.6 Blockchain, NFT και οικονομία

Καθώς εφαρμόζονται τα συστήματα της Βιομηχανίας 4.0, η οικονομία, επηρεαζόμενη σημαντικά από αναδυόμενες τεχνολογίες όπως το blockchain και τη χρήση ψηφιακών περιουσιακών στοιχείων όπως τα NFT, αλλάζει και μεταποιείται. Αυτή η μετασχηματιστική περίοδος χαρακτηρίζεται από την ψηφιοποίηση και την αναδιάρθρωση του τρόπου με τον οποίο τα προϊόντα και οι υπηρεσίες αποκτούν αξία μέσα από την ενσωμάτωση τεχνολογιών στην παραγωγική διαδικασία. Για παράδειγμα, μια εταιρεία παραγωγής αυτοκινήτων μπορεί να χρησιμοποιεί εξελιγμένα ρομποτικά συστήματα για την αυτοματοποίηση της συναρμολόγησης, αυξάνοντας την αποδοτικότητα και μειώνοντας το κόστος, παρέχοντας ταυτόχρονα μεγαλύτερη συνέπεια και ποιότητα στο τελικό προϊόν. Σε ένα άλλο παράδειγμα, πάνω στην ψηφιακή αγορά νομισμάτων, το Ethereum χρησιμοποιεί την τεχνολογία blockchain και έξυπνων συμβολαίων για να παρέχει μια ασφαλή, διαφανή και αποκεντρωμένη μέθοδο συναλλαγών. Αυτό επιτρέπει στους χρήστες να ανταλλάσσουν αγαθά και συνάλλαγμα απευθείας, χωρίς την ανάγκη για τραπεζικούς διαμεσολαβητές, μειώνοντας το κόστος και το χρόνο των συναλλαγών. Συγκεκριμένα στο

παράδειγμα των Blockchain, η τεχνολογία αυτή ενισχύει τη διαφάνεια και την αποδοτικότητα σε αυτές τις διαδικασίες, προωθώντας μια πιο συνδεδεμένη και οργανωμένη οικονομία. Αυτές οι πρόοδοι διαταράσσουν τα παραδοσιακά οικονομικά μοντέλα, παρέχοντας αποκεντρωμένες διαδικασίες που μειώνουν την εξάρτηση από κεντρικές αρχές και διαμεσολαβητές, επαναπροσδιορίζοντας, έτσι, τις οικονομικές αλληλεπιδράσεις σε παγκόσμιο επίπεδο.

Τα NFT λύνουν ένα πλήθος μακροχρόνιων προβλημάτων στις ψηφιακές συναλλαγές, κυρίως αντιμετωπίζοντας ζητήματα που σχετίζονται με την σπανιότητα των ψηφιακών αγαθών και την ιδιοκτησία αυτών. Επιτρέποντας στα ψηφιακά αντικείμενα να είναι μοναδικά και ιδιόκτητα, τα NFT αποτρέπουν την ανεξέλεγκτη αναπαραγωγή και διανομή ψηφιακών περιουσιακών στοιχείων, διασφαλίζοντας ότι οι δημιουργοί μπορούν να διατηρούν τον έλεγχο των ψηφιακών έργων τους και να επωφελούνται οικονομικά από αυτά [23][24]. Αυτή η δυνατότητα είναι ζωτικής σημασίας σε τομείς όπως η ψηφιακή τέχνη, η μουσική και τα βιντεοπαιχνίδια, όπου η πειρατεία και η μη εξουσιοδοτημένη αντιγραφή έχουν μειώσει τα έσοδα των δημιουργών.

Στον ψηφιοποιημένο κόσμο της Βιομηχανίας 4.0, τα NFT προσφέρουν νέες οικονομικές δυνατότητες μέσα από τη δημιουργία νέων αγορών και μορφών επένδυσης. Για παράδειγμα, επιτρέπουν την εκμετάλλευση φυσικών περιουσιακών στοιχείων, όπως ακίνητα και συλλεκτικά αντικείμενα, μετατρέποντάς τα σε ψηφιακά περιουσιακά στοιχεία που μπορούν να αγοραστούν και να ανταλλαχθούν ευκολότερα και με χαμηλότερο κόστος συναλλαγών. Επιπλέον, τα NFT επιτρέπουν την υλοποίηση έξυπνων συμβολαίων, τα οποία αυτοματοποιούν και επιβάλλουν τους όρους μιας συμφωνίας μεταξύ αγοραστών και πωλητών, διευρύνοντας το φάσμα του ψηφιακού εμπορίου. Αυτές οι καινοτομίες είναι σημαντικές για την προώθηση μιας πιο περιεκτικής, αποδοτικής και ελεύθερης οικονομικής τοπογραφίας, διαμορφώνοντας τον δρόμο για νέα επιχειρηματικά μοντέλα και πηγές εσόδων στην ψηφιακή εποχή. Αυτές οι καινοτομίες είναι σημαντικές γιατί βοηθούν στη δημιουργία μιας οικονομίας που είναι πιο αποδοτική, προσβάσιμη και έτοιμη για το μέλλον. Παράλληλα, ανοίγουν τον δρόμο για τη δημιουργία νέων επιχειρηματικών μοντέλων και τρόπων κερδοφορίας στην εποχή της ψηφιακής τεχνολογίας.

1.7 Βιομηχανία 4.0

Η Βιομηχανία 4.0, γνωστή και ως η Τέταρτη Βιομηχανική Επανάσταση, αναφέρεται στην τρέχουσα τάση αυτοματοποίησης και ανταλλαγής δεδομένων ανάμεσα στα διάφορα συστήματα της Βιομηχανίας. Περιλαμβάνει κυβερνο-φυσικά συστήματα (CPS – Cyber-Physical Systems),

το Διαδίκτυο των Πραγμάτων (IoT), την υπολογιστική νέφος (Cloud Computing) και τη γνωστική υπολογιστική (cognitive computing). Στον πυρήνα της, η Βιομηχανία 4.0 ενσωματώνει τις ψηφιακές και φυσικές πτυχές της παραγωγής σε ένα ενιαίο σύστημα για να αυξήσει την αποδοτικότητα των διαδικασιών παραγωγής, να βελτιώσει τις ενδιάμεσες διαδικασίες και να συμβάλει στην δημιουργία ενός πιο δυναμικού και αλληλένδετου παραγωγικού περιβάλλοντος. Ο απώτερος σκοπός της Βιομηχανίας 4.0 είναι η δημιουργία "έξυπνων εργοστασίων", όπου μηχανές και συστήματα μπορούν να ανταλλάσσουν πληροφορίες αυτόνομα, χωρίς ανθρώπινη παρέμβαση, να εκτελούν διεργασίες και να ελέγχουν το ένα το άλλο ανεξάρτητα.

Αυτά τα νέα στοιχεία αυτοματοποίησης παρέχουν σημαντικά οφέλη στην κοινωνία και τον παραγωγικό τομέα, ενισχύοντας την παραγωγικότητα, την ευελιξία και τη λειτουργική αποδοτικότητα. Έτσι, είναι εφικτή η δημιουργία μίας πιο προσωποποιημένης και εστιασμένης παραγωγής στον πελάτη, όπου τα προϊόντα μπορούν να προσαρμοστούν στις ατομικές ανάγκες του κάθε καταναλωτή με ελάχιστο επιπλέον κόστος. Επιπλέον, η Βιομηχανία 4.0 συμβάλλει στη βιωσιμότητα μέσω της βελτιστοποίησης της χρήσης πόρων, της μείωσης των αποβλήτων και την ελαχιστοποίηση των σφαλμάτων. Ενδυναμώνει, επίσης, τους εργαζομένους με νέα εργαλεία και τεχνολογίες, οδηγώντας σε ένα πιο ασφαλές περιβάλλον εργασίας, σε νέες θέσεις εργασίας και στη δημιουργία νέων ρόλων που απαιτούν προηγμένες δεξιότητες.

Ένα σημαντικό εργαλείο για την Βιομηχανία 4.0 είναι η ενσωμάτωση της τεχνολογίας Non-Fungible Token (NFT), τα οποία καλύπτουν την ανάγκη για μοναδικούς, επαληθεύσιμους αναγνωριστικούς κωδικούς στη διαδικασία παραγωγής που δεν μπορούν να αντιγραφούν ή να παραποιηθούν και μπορούν έμπιστα και γρήγορα να συναλλαχθούν και να μεταφερθούν ανάμεσα σε συστήματα και παραγωγικές / οικονομικές οντότητες. Τα NFT προσφέρουν έναν ψηφιακό τρόπο πιστοποίησης της αυθεντικότητας και της κυριότητας σε κάθε στάδιο της αλυσίδας εφοδιασμού. Αυτό είναι ιδιαίτερα πολύτιμο σε βιομηχανίες όπου η προέλευση, η ποιότητα, το Brand και πιστοποιήσεις ISO των προϊόντων είναι ζωτικής σημασίας, όπως τα φαρμακευτικά προϊόντα, τα πολυτελή είδη και τα τρόφιμα. Χρησιμοποιώντας τα NFT, οι εταιρείες μπορούν να διασφαλίσουν ότι κάθε προϊόν, μέρος ή συστατικό έχει ένα μοναδικό, εντοπίσιμο ιστορικό, αυξάνοντας την ασφάλεια και κτίζοντας εμπιστοσύνη μεταξύ καταναλωτών και προμηθευτών. Επιπλέον, τα NFT διευκολύνουν νέες μορφές ψηφιακής κυριότητας και διαχείρισης

περιουσιακών στοιχείων, κάνοντας εφικτή την δημιουργία νέων καινοτόμων επιχειρηματικών μοντέλων και νέες ροές εσόδων, στον τομέα της παραγωγής.

1.8 Δομή της Έρευνας

Η έρευνα απαρτίζεται από 3 βασικά μέρη, το κομμάτι της αρχικής τεχνολογικής και βιβλιογραφικής έρευνας και διατύπωσης του θεωρητικού προκατασκευαστικού μέρους, της εφαρμογής του θεωρητικού μέρους σε ένα σύστημα και τέλος την δημιουργία παρατηρήσεων για την απάντηση του ερευνητικού ερωτήματος.

Κατά την τέλεση της έρευνας, υπήρξαν τεχνολογικοί περιορισμοί, καθώς η χρήση των συστημάτων Blockchain στα βιντεοπαιχνίδια δεν έχει δοκιμαστεί και εφαρμοστεί εκτενώς, για την ανάπτυξη της απαραίτητης τεχνολογίας πλήρους υποστήριξης των βιντεοπαιχνιδιών. Μέσα από την μελέτη της δομής των βιντεοπαιχνιδιών, των μηχανών βιντεοπαιχνιδιών και της επιστήμης υπολογιστών ανυψώθηκαν ερωτήματα για την ικανότητα των συστημάτων Blockchain να συναλλάξουν κώδικα παιχνιδιών, σκελετούς και 3D meshes καθώς και άλλα ακόμα πιο περίπλοκα στοιχεία – όπως μια ολόκληρη πίστα ή μια ολόκληρη διεπαφή χρήστη. Βρέθηκαν πολλοί περιορισμοί και ασυμβατότητες στην συναλλαγή στοιχείων παιχνιδιού. Μέσα από το σύνολο της έρευνας, έγινε ξεκάθαρο πως ένα από τα μέρη του συστήματος θα έπρεπε να εμπεριέχει λειτουργικά και σχεδιαστικά αρκετούς περιορισμούς για να συμβαδίζει με την τωρινή κατάσταση του μεγαλύτερου, τωρινού, δικτύου με εφαρμογή τεχνολογίας NFT, που είναι το Ethereum.

Εν τέλει, το ερευνητικό μέρος φέρει τέσσερις αριθμημένες δυσκολίες, οι οποίες δομούν το περιεχόμενο του θεωρητικού μέρους. Οι οποίες δυσκολίες είναι:

- I. Τι είναι οντότητα προς τους ψηφιακούς κόσμους και πως η φύση μιας οντότητας επηρεάζει το σύστημα κάτω από ασαφείς συνθήκες;
- II. Πρέπει να ερευνηθεί το κατά πόσο τα τωρινά συστήματα Blockchain είναι αρκετά για την κατασκευή του λειτουργικού εφαρμοσμένου μέρους και στην περίπτωση που δεν είναι, πως θα αποτυπωνόταν μια γενικευμένη προσέγγιση προς την τεχνολογία του Blockchain, χωρίς την παρουσίαση των τεχνολογιών που δεν ανταποκρίνονται στις επαυξημένες απαιτήσεις του τελικού κατασκευαστικού μέρους; Ταυτόχρονα, ποιες λειτουργίες των βιντεοπαιχνιδιών καλύπτουν, στο παρόν, τα μεγαλύτερα δίκτυα με έξυπνα συμβόλαια;

- III. Πως μεταφέρονται τα διαλειτουργικά στοιχεία και μεταφέρονται οι λειτουργίες τους ανάμεσα σε διαφορετικά παιχνίδια;
- IV. Υφίσταται μια αχνή διασύνδεση ανάμεσα στα βιντεοπαιχνίδια και τα κυβερνο-φυσικά συστήματα, διότι αυτοί οι δύο κλάδοι μοιράζονται τεχνολογίες. Ο εντοπισμός του ακριβούς σημείου τομής και των προεκτάσεων που βρίσκουν τα φυσικά κυβερνητικά συστήματα της βιομηχανίας προς τα βιντεοπαιχνίδια, μέσα από φυσικές οντότητες που χαρακτηρίζονται από μοναδικά τεκμήρια, απαιτεί ιδιαίτερη ανάλυση πολλών συστημάτων και παρουσίασε υποθέσεις και ασάφειες για έναν κλάδο που δεν έχει αναπτυχθεί ακόμα. Όμως μέσα από το σώμα του θεωρητικού μέρους και τα διαλειτουργικά στοιχεία, η σύνδεση γίνεται εφικτή. Υφίσταται μια αχνή διασύνδεση ανάμεσα στα βιντεοπαιχνίδια και τα κυβερνο-φυσικά συστήματα, διότι αυτοί οι δύο κλάδοι μοιράζονται τεχνολογίες. Ο εντοπισμός του ακριβούς σημείου τομής και των προεκτάσεων που βρίσκουν τα φυσικά κυβερνητικά συστήματα της βιομηχανίας προς τα βιντεοπαιχνίδια, μέσα από φυσικές οντότητες που χαρακτηρίζονται από μοναδικά τεκμήρια, απαιτεί ιδιαίτερη ανάλυση πολλών συστημάτων και παρουσίασε υποθέσεις και ασάφειες για έναν κλάδο που δεν έχει αναπτυχθεί ακόμα. Όμως μέσα από το σώμα του θεωρητικού μέρους και τα διαλειτουργικά στοιχεία, η σύνδεση γίνεται εφικτή.

Μία ακόμα δυσκολία που παρουσιάστηκε στο πρακτικό μέρος, ήταν η μορφοποίηση προγραμμάτων για το EVM που φέρουν λειτουργίες Back-End στον περιορισμένο επιτρεπτό μέγεθος των έξυπνων συμβολαίων. Διερευνήθηκε αρχικά η χρήση πολλαπλών συμβολαίων, αλλά ο όγκος των δεδομένων ανά χρήστη και η πολύπλοκη φύση του marketplace αποδείχτηκαν αποτρεπτικά στοιχεία. Εν τέλει ο κώδικας βελτιστοποιήθηκε και δημιουργήθηκε ένα συμβόλαιο με πολλές χρήσεις.

Εν τέλη, η προσέγγιση που ακολουθήθηκε εφαρμόζει τεχνολογίες αιχμής στα όρια χρήσης τους με τις τωρινές τεχνολογίες και γίνεται παρατήρηση και ανάλυση του τελικού κατασκευάσματος. Αυτή η προσέγγιση μέσα από τεχνολογίες αιχμής δομεί όλο το θεωρητικό μέρος.

Κεφάλαιο 2 Μοναδικά Τεκμήρια μέσα από την τεχνολογία Blockchain και το Ethereum Network

2.1 Διαμοιρασμός δεδομένων και P2P ομότιμα δίκτυα

Μία από τις σημαντικότερες λειτουργίες ενός οποιουδήποτε συστήματος που εμπεριέχει ψηφιοποιημένες οντότητες με μοναδικά τεκμήρια, είναι το σύστημα κατανομής και διαχείρισης των ψηφιακών πόρων και πακέτων που εμπεριέχουν τις ψηφιοποιημένες υπογραφές. Αναλόγως σημαντικός είναι και ο μηχανισμός έκδοσης αναγνωριστικών και τα μεταδεδομένα των μεμονομένων οντοτήτων. Τα προαναφερόμενα συστήματα, στο σύνολο τους, επιτρέπουν τον ασφαλή διαμοιρασμό των δεδομένων και την παρακολούθησή τους, καθώς αλλάζουν χέρια στο διαδίκτυο.

Υπάρχει ένα μεγάλο πλήθος πιθανών συστημάτων που χρησιμοποιούνται στη βιομηχανία για την μεταφορά πακέτων δεδομένων. Για παράδειγμα, ο τελικός χρήστης μπορεί να λάβει πακέτα μέσα από ένα δίκτυο παράδοσης περιεχομένου (CDN), είτε με άμεση λήψη από κάποιον Server ή Υπολογιστή, είτε μπορεί να λάβει αρχεία από ένα σύνολο από άλλους υπολογιστές εντός ενός peer-to-peer (ή P2P, ελληνικά: ομότιμο δίκτυο) δικτύου. Η λήψη αρχείων από στατικούς Servers όπως τα CDN είναι μια κεντροποιημένη (centralized) μέθοδος λήψης αρχείων. Αντίθετα, ένα δίκτυο P2P αποτελεί μια αποκεντροποιημένη μέθοδο λήψης αρχείων [1].

Ένα δίκτυο Peer-to-Peer (P2P) ακολουθεί την αρχιτεκτονική ενός κατανεμημένου δικτύου όπου η ισχύς επεξεργασίας, τα δεδομένα και οι εφαρμογές κατανέμονται σε πολλούς υπολογιστές, αντί να είναι μαζεμένα σε μία τοποθεσία ή διακομιστή. Σε ένα τέτοιο σύστημα, κάθε υπολογιστής (συντά αναφέρεται ως "κόμβος") μπορεί να λειτουργεί ανεξάρτητα και, σε πολλές περιπτώσεις, σε συνεργασία με άλλους κόμβους για την επίτευξη ορισμένων εργασιών ή functions ή για την διάθεση αρχείων προς λήψη [2]. Οι μεμονωμένοι κόμβοι (nodes) στο δίκτυο (που αναφέρονται ως "ομότιμοι" ή peers στα αγγλικά) λειτουργούν τόσο ως «προμηθευτές» όσο και ως «καταναλωτές» πόρων. Αυτό έρχεται σε αντίθεση με το παραδοσιακό μοντέλο πελάτη-διακομιστή, όπου ένας κεντρικός διακομιστής παρέχει και οι πελάτες καταναλώνουν πόρους. Σε ένα δίκτυο P2P, κάθε peer είναι αυτόνομο και επικοινωνεί απευθείας με άλλα peers. Κάθε νέο peer που εντάσσεται στο δίκτυο επωφελείται από τους υπάρχοντες πόρους, αλλά, ταυτόχρονα, προσθέτει στους συνολικούς διαθέσιμους πόρους. Με το πρωταρχικό πλεονέκτημα των δικτύων

P2P να βρίσκεται στην κατανεμημένη φύση τους, δεδομένου ότι δεν υπάρχει μόνο ένα σημείο ελέγχου ή αποτυχίας. Τα δίκτυα P2P είναι ανθεκτικά έναντι συστημικών αστοχιών και μπορούν να προσφέρουν δυνατότητες self-hosting στους μεμονωμένους χρήστες εντός ενός δικτύου[3][4].

Στα P2P δίκτυα, τα μεμονωμένα αρχεία που διαμοιράζονται - αποστέλλονται σαν Chunks ή Pieces, κομμάτια δηλαδή - για την ταχύτερη εύρεση λαθών σε μεγάλα αρχεία λήψης. Τα pieces ενός αρχείου που διαμοιράζεται σε ένα δίκτυο P2P συνήθως εμπεριέχουν ένα κρυπτογραφημένο κλειδί, το οποίο χρησιμοποιείται σαν hash για την επαλήθευση του αρχείου[3][4]. Η κάθε εφαρμογή χρησιμοποιεί το δικό της ξεχωριστό σύστημα που έχει δημιουργηθεί συγκεκριμένα για να καλύπτει κάποιες ανάγκες που προκαθορίζονται πριν την δημιουργία του P2P συστήματος. Ένα παράδειγμα ενός δημοφιλούς P2P συστήματος είναι το Bittorrent και η λήψη torrents. Στο συγκεκριμένο παράδειγμα, ένας torrent client παίρνει αρχεία .torrent ή υπερσυνδέσμους magnet link που φέρουν ένα σετ από μεταδεδομένα (metadata) - καθοδηγώντας τον client στην λήψη του αρχείου. Αυτό το αρχείο «καθοδηγητής» μπορεί το ίδιο να έχει εκδώσει ένα hash key για την επαλήθευση του. Μέσα στο αρχείο .torrent βρίσκονται τα κομμάτια των αρχείων - αλλιώς και pieces- που έχουν ένα συγκεκριμένο μέγεθος ανά κομμάτι (π.χ. 256 KB, 512 KB) και για κάθε ένα τέτοιο κομμάτι, το αρχείο .torrent εμπεριέχει ένα SHA-1 hash κλειδί. Αυτά τα hash χρησιμοποιούνται από τον torrent client για την επαλήθευση της ακεραιότητας κάθε ληφθέντος κομματιού. Εάν ένα torrent έχει 5.000 κομμάτια, θα υπάρχουν 5.000 SHA-1 hash keys μέσα στο αρχείο .torrent [5].

Εντός των πλαισίων ενός δικτύου P2P, είναι εφικτός ο διαμοιρασμός αρχείων, δεδομένων, μηνυμάτων αλλά και η λειτουργική περίπλεξη αυτών και η κατασκευή ενός συστήματος διαμοιρασμού εικονικών / ψηφιακών νομισμάτων ή ψηφιακών αγαθών πάσης φύσεως. Για την εκτέλεση αυτών των ηλεκτρονικών συναλλαγών χρησιμοποιούνται πολλά συστήματα και από αυτή τη συνεργατική λειτουργία, μπορεί να προκύψει η βάση ενός ψηφιακού νομίσματος, όπως το Bitcoin, ένα αποκεντρωμένο ψηφιακό νόμισμα χωρίς μεσάζοντες και ιδιοκτήτη, το οποίο δεν απαιτεί εμπιστοσύνη μεταξύ των μερών μιας συναλλαγής. Αυτά τα συστήματα αποτελούνται από ένα σύστημα που παρέχει και δημιουργεί νομίσματα από ψηφιακές υπογραφές και ένα δίκτυο peer-to-peer που χρησιμοποιεί το Proof of Work (απόδειξη εργασίας) για την καταγραφή ενός δημόσιου ιστορικού συναλλαγών, αποτελούμενο από αξιόπιστους κόμβους [6]. Αυτό αποτελεί έναν απλοποιημένο ορισμό του Bitcoin που θεωρείται η αρχή του concept του

Blockchain και το ίδιο το Bitcoin, όπως περιγράφεται από τον ίδιο τον Satoshi Nakamoto, είναι ένα Peer-to-Peer ψηφιακό/ηλεκτρονικό σύστημα χρημάτων [6].

2.2 BLOCKCHAIN και Δυνατότητες

Ο λειτουργικός πυρήνας του Bitcoin, καθώς και ο λόγος της μεγάλης και ευρείας απήχησης του στην αγορά, βρίσκεται στο αποκεντροποιημένο ηλεκτρονικό ledger (καθολικό) που κρατείται [10]. Μέσω αυτού επαληθεύονται και καταγράφονται οι συναλλαγές των μοναδικών νομισμάτων που είναι τα Bitcoin.

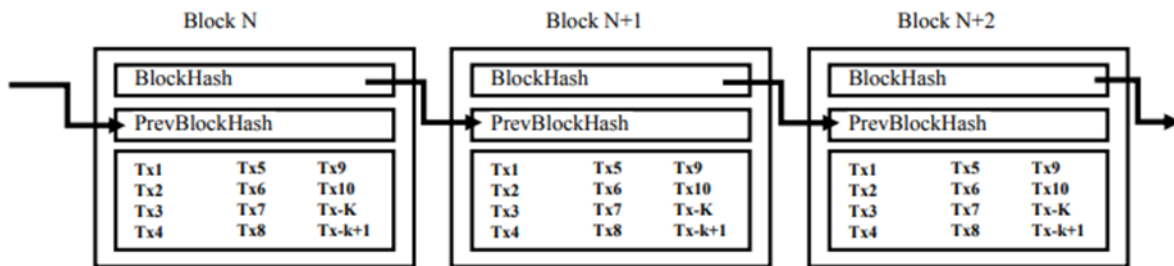
Τα συστήματα Blockchain αναπτύχθηκαν ως μέρος των λύσεων και συστημάτων που επιτρέπουν τη λειτουργία του δικτύου Bitcoin. Στην περαιτέρω ανάπτυξή τους, όμως, πλέον χρησιμοποιούνται και εκτός των πλαισίων χρήσης τους σε εφαρμογές με κρυπτονομίσματα.

Το Blockchain αποτελεί την ίδια τη δομή του ledger σε μια απλουστευμένη εφαρμογή του. Η λέξη "Blockchain" είναι απόλυτα περιγραφική της φύσης του, καθώς τα δεδομένα αποθηκεύονται σε "blocks" (μπλοκ δεδομένων) και αυτά τα μπλοκ είναι διαδοχικά "αλυσοδεμένα" (περιπλεγμένα σαν κρίκοι μιας αλυσίδας) μεταξύ τους. Τα blocks είναι ψηφιακά "δοχεία" που περιλαμβάνουν ένα σύνολο δεδομένων.

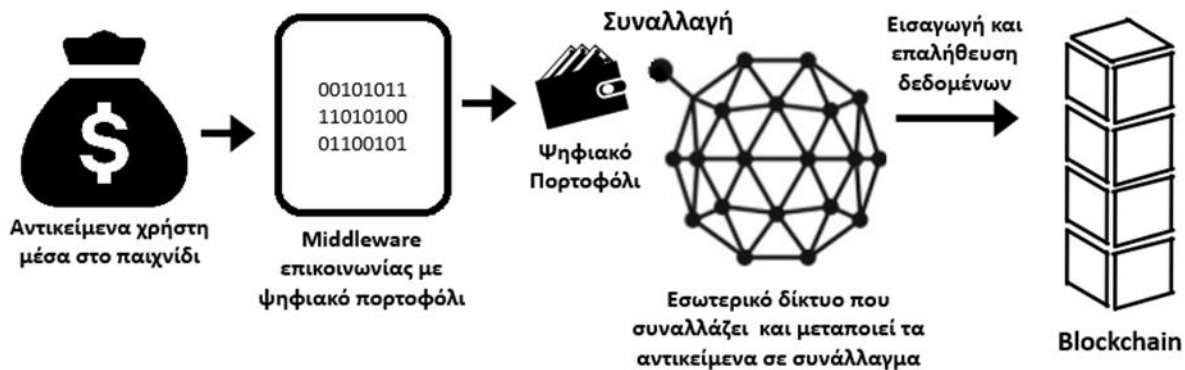
Στο πλαίσιο των κρυπτονομισμάτων, όπως το Bitcoin, αυτά τα δεδομένα στο σύνολό τους μορφοποιούν μια λίστα συναλλαγών. Κάθε μπλοκ φέρει συγκεκριμένα δεδομένα, μια χρονική σήμανση, μια αναφορά στο προηγούμενο μπλοκ μέσω ενός κρυπτογραφικού hash και το δικό του μοναδικό hash[12]. Το κομμάτι του chain προκύπτει από τη διαδοχική σύνδεση αυτών των μπλοκ.

Καθώς κάθε μπλοκ επαληθεύεται (συχνά μέσω μηχανισμών συναίνεσης όπως η απόδειξη εργασίας Proof of Work POW) και προστίθεται σε μια γραμμική, χρονολογικά διατεταγμένη αλυσίδα, η συνέχεια και η σύνδεση διατηρούνται μέσω του κρυπτογραφικού hash του προηγούμενου μπλοκ, που συμπεριλαμβάνεται στο επόμενο μπλοκ, σαν ένα αδιάσπαστο κομμάτι του. Αυτή η δομή διασφαλίζει ότι η συνέχεια θα διατηρηθεί και πως κάποιο παλαιότερο μπλοκ

δεν θα πειραχτεί από κάποια κακόβουλη οντότητα, αφού οποιαδήποτε αλλαγή οποιουδήποτε μπλοκ θα απαιτούσε τον επανυπολογισμό των hash όλων των επόμενων μπλοκ.



Γραφική αναπαράσταση αρχιτεκτονικής του Blockchain του Bitcoin. Το δεύτερο Block στην συγκεκριμένη ακολουθία από Blocks - το Block N+1 - εμπεριέχει το hash του Block N, μορφοποιώντας μια αλυσίδα όταν το hash του Block N+1 συμπεριληφθεί στο Block N+2. Πηγή: [Blockchain Technology for Industry 4.0: Secure, Decentralized, Distributed and Trusted Industry Environment 2020 Rodrigo da Rosa Righi, Antonio Marcos Alberti, Madhusudan Singh] [10]



Η εικόνα παρουσιάζει ένα απλό δίκτυο συναλλαγής και μετα-ψηφιοποίησης αντικειμένων από έναν αρχικό ψηφιακό χώρο ύπαρξης - προς ένα Blockchain, με την χρήση middleware συστήματος επικοινωνίας ανάμεσα στο παιχνίδι και μια εξωτερική τοπική εφαρμογή ψηφιακού πορτοφολιού.

Ο όρος Blockchain γενικευμένα περιγράφει μια ακολουθία πακέτων που είναι διασυνδεδεμένα με κρυπτογραφικά κλειδιά. Όμως μέσα σε αυτή την γενικευμένη έννοια υπάρχουν κάποια βασικά χαρακτηριστικά - τα οποία συνήθως, τα συστήματα με εφαρμογές Blockchains, πληρούν, για να επιφέρουν τα ίδια αξία στον χρήστη κατά την εφαρμογή και χρήση τους. Τα βασικά αυτά

χαρακτηριστικά παρουσιάζονται στην ακόλουθη λίστα με τον μορφή [Χαρακτηριστικό: Πως προσφέρεται αξία μέσα από το χαρακτηριστικό αυτό]:

Σπανιότητα: Η σπανιότητα παρέχεται μέσα από την επιβολή μιας σταθερής περιορισμένης προσφοράς ενός μετρήσιμου αγαθού, δίνοντας αξία στα αγαθά αυτά. Ένα τέτοιο σύστημα προσομοιώνει τα πολύτιμα μέταλλα προς την οικονομία (π.χ. χρυσός) τα οποία λειτουργούν ως αποθετήριο αξίας λόγω της περιορισμένης διαθεσιμότητάς τους. Η σπανιότητα διευκολύνει την προβλεψιμότητα της αξίας των περιουσιακών στοιχείων και ενθαρρύνει την επένδυση.

Προβλεψιμότητα: Η ντετερμινιστική (Αιτιοκρατική) φύση των αλγορίθμων του blockchain καθιστά τις συναλλαγές και τα αποτελέσματα των έξυπνων συμβάσεων (smart contracts) προβλέψιμα, προσθέτοντας στην ελκυστικότητα του blockchain σαν μια σταθερή και αξιόπιστη πλατφόρμα.

Ασφάλεια: Τα συστήματα Blockchain χρησιμοποιούν κρυπτογραφικές τεχνικές και έχουν σχεδιαστεί, έτσι ώστε να είναι ανθεκτικά σε προσπάθειες κακόβουλου χειρισμού του δικτύου του Blockchain ή των αποθηκευμένων στοιχείων εντός του Blockchain. Καθιστώντας τα ιδανικά για την αποθήκευση ευαίσθητων δεδομένων ή για τις περιπτώσεις όπου η διατήρηση των δεδομένων και η αποφυγή τροποποίησης τους είναι μέγιστης σημασίας, όπως αρχεία πιστοποίησης συναλλαγών.

Αποκέντρωση: Η «εξουσία» κατανέμεται μεταξύ πολλών συμμετεχόντων, καθιστώντας το σύστημα ανθεκτικό στη λογοκρισία και αποφεύγοντας την ύπαρξη ενός μοναδικού σημείου αποτυχίας. Αυτό ενισχύει την εμπιστοσύνη και καταργεί την ανάγκη για κεντρικούς διαμεσολαβητές.

Διαφάνεια: Οι συναλλαγές καταγράφονται δημοσίως, βελτιώνοντας τη λογοδοσία (ανάληψη ευθυνών) και μειώνοντας τις δραστηριότητες απάτης. Αυτή η ανοικτή επαλήθευση είναι καίρια για τη δημόσια εμπιστοσύνη.

Αντίσταση στη Λογοκρισία: Η αποκεντρωμένη φύση εξασφαλίζει ότι καμία μεμονωμένη οντότητα δεν μπορεί να ελέγξει ή να τροποποιήσει τα δεδομένα, παρέχοντας ένα υψηλό επίπεδο ελευθερίας.

Ανεκτικότητα Σφάλματος: Οι μηχανισμοί όπως το Practical Byzantine Fault Tolerance (PBFT) βοηθούν τα blockchains να λειτουργούν ακόμη και όταν μερικοί κόμβοι είναι κακόβουλοι ή εκτός λειτουργίας.

Τελικότητα: Η τελικότητα της συναλλαγής εξασφαλίζει ότι μόλις μια συναλλαγή επιβεβαιωθεί, δεν μπορεί να ανατραπεί, μειώνοντας τον κίνδυνο απάτης.

Χαμηλό Κόστος Συναλλαγών: Η αποκεντροποιημένη φύση των Blockchains εξαλείφει την ανάγκη για διαμεσολαβητές (όπως τράπεζες), μειώνοντας έτσι το κόστος των συναλλαγών.

Προγραμματιστικότητα: Οι έξυπνες συμβάσεις επιτρέπουν προγραμματιζόμενες, υπό όρους συναλλαγές, καθιστώντας το blockchain προσαρμόσιμο για διάφορες χρήσεις.

Διαλειτουργικότητα (Interoperability): Διευκολύνει την αλληλεπίδραση μεταξύ διαφορετικών blockchains, αυξάνοντας τη χρησιμότητα. Ταυτόχρονα, οι οντότητες οι οποίες χαρακτηρίζονται με μοναδικά τεκμήρια μπορούν να συναλλαχθούν ελεύθερα μέσα σε συμβατά δίκτυα Blockchain ή προς οντότητες του φυσικού χώρου με ασφάλεια.

Διακυβέρνηση: Τα κοινοτικά μοντέλα (Community-driven models) επιτρέπουν δημοκρατικές ή ημι-δημοκρατικές δομές διακυβέρνησης, καθιστώντας τις αλλαγές του πρωτοκόλλου λειτουργίας του συστήματος πιο προσαρμόσιμες και αποδεκτές.

Ιδιωτικότητα: Τα blockchains έχουν την δυνατότητα να προσφέρουν λειτουργίες όπως οι αποδείξεις μηδενικής γνώσης (Zero-Knowledge Proof) επιτρέποντας την επιβεβαίωση συναλλαγών χωρίς να αποκαλύπτονται ευαίσθητες πληροφορίες. Αυτό προσφέρει αυξημένη ιδιωτικότητα και ασφάλεια στις συναλλαγές.

Συμμόρφωση και Ρύθμιση: Πολλά Blockchain σχεδιάζονται, έτσι ώστε να συμμορφώνονται με το νομικό πλαίσιο της κάθε χώρας και να είναι συμβατά με κρατικά συστήματα και μηχανισμούς. Αυτό τα καθιστά ελκυστικά για τη βιομηχανία, επιτυγχάνοντας την ισορροπία μεταξύ καινοτομίας και νομιμότητας.

2.3 Βασικές Λειτουργίες ενός συστήματος Blockchain

Οι λειτουργίες ενός συστήματος Blockchain που προσφέρουν αξία σε ένα δίκτυο είναι πολυπληθείς και συνεχώς εξελισσόμενες. Κατά την ανάλυση κάποιων βασικών λειτουργιών της

αρχιτεκτονικής των Blockchain, είναι εφικτή η κατηγοριοποίηση αυτών σε επίπεδα για την καλύτερη κατανόηση τους.

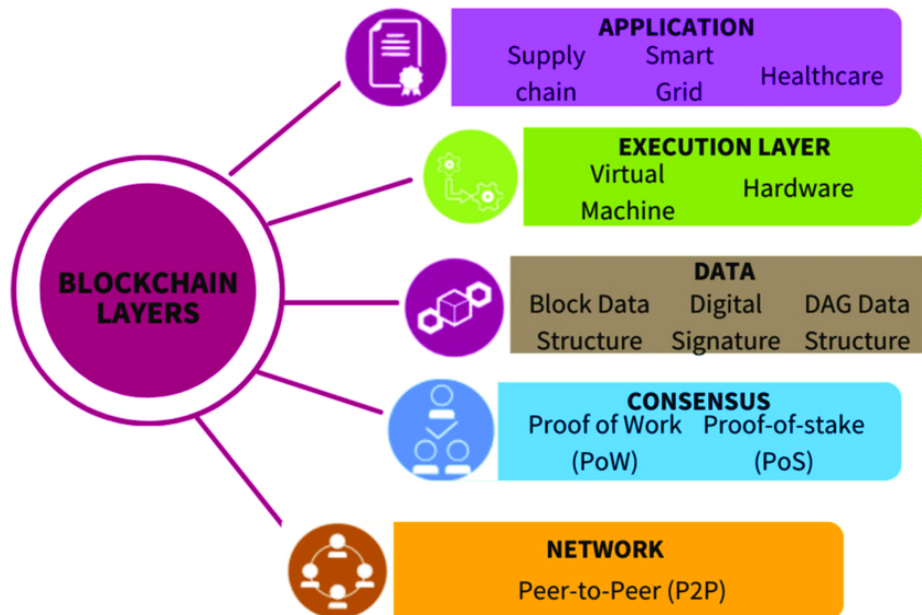
Επίπεδο 1: Στον πυρήνα της τεχνολογίας των δικτύων Blockchain βρίσκεται το υλισμικό και οι τεχνολογίες δικτύου που υποστηρίζουν την ύπαρξη του P2P ομότιμου δικτύου ανάμεσα στους μεμονωμένους διακομιστές (server) που παίρνουν μέρος στην επαλήθευση των Block.

Επίπεδο 2: Εδώ βρίσκεται η σημαντικότερη διαδικασία που προσφέρει αξία σε ένα δίκτυο Blockchain, η διαδικασία της συναίνεσης. Το επίπεδο Συναίνεσης περιλαμβάνει μηχανισμούς όπως το Proof of Work και το Proof of Stake, οι οποίοι είναι σημαντικοί για την επικύρωση των συναλλαγών και την επίτευξη συμφωνίας μεταξύ των κόμβων του δικτύου σχετικά με την κατάσταση του καθολικού, αποτρέποντας έτσι την απάτη και εξασφαλίζοντας την εμπιστοσύνη στο δίκτυο. Τα επίπεδα Δεδομένων και Συναίνεσης αποτελούν τις θεμελιώδεις τεχνολογίες που εξασφαλίζουν την ακεραιότητα και τη λειτουργικότητα του blockchain.

Επίπεδο 3: Το επίπεδο Δεδομένων εμπεριέχει δομές όπως το σύστημα των Block, τις ψηφιακές υπογραφές και το καθολικό (Ledger), τα οποία συλλογικά διατηρούν την ακεραιότητα των δεδομένων στο blockchain.

Επίπεδο 4: Περιλαμβάνει τις Εικονικές Μηχανές ή το περιβάλλον εκτέλεσης των έξυπνων συμβολαίων (Smart Contract) και το Υλισμικό (Hardware) που χρησιμοποιείται. Λειτουργεί ως η υποδομή που επιτρέπει τη λειτουργία έξυπνων συμβολαίων και αποκεντρωμένων εφαρμογών (DApps). Η Εικονική Μηχανή Ethereum (EVM) ανήκει σε αυτό το επίπεδο.

Επίπεδο 5: Είναι το επίπεδο επαφής του συστήματος Blockchain με εξωτερικά συστήματα. Υπάρχει σε αυτό το επίπεδο, ένα μεγάλο πλαίσιο εργαλείων και υποδομών που επιτρέπουν σε εφαρμογές να επικοινωνήσουν με το δίκτυο Blockchain και να μεταφέρουν δεδομένα και συμβόλαια. Αυτές οι εφαρμογές, με την σειρά τους, δείχνουν την ευελιξία της τεχνολογίας blockchain πέρα από τις χρηματοοικονομικές της ρίζες.



Πηγή: Smith, J. "Πέντε Διαφορετικά Επίπεδα του Blockchain.", Διατίθεται υπό την Άδεια Creative Commons Attribution 4.0 International. ResearchGate, 2023. Διαθέσιμο στο: [\[https://www.researchgate.net/figure/Five-Different-Layers-of-Blockchain_fig2_361234744\]](https://www.researchgate.net/figure/Five-Different-Layers-of-Blockchain_fig2_361234744)

Αυτά τα χαρακτηριστικά, που επιφέρουν αξία σε ένα δίκτυο Blockchain, προσφέρονται μέσα από ένα πλαίσιο βασικών σημαντικών λειτουργιών (key system functions) του Blockchain. Αν και τα Blockchain αποτελούν έναν ραγδαία εξελισσόμενο κλάδο και θα ήταν αδύνατο να προσδιοριστούν αυτά τα χαρακτηριστικά για ένα απροσδιόριστο πλαίσιο χρήσεων, αναλύοντας - τις μέχρι το 2023- δομές του Blockchain και του Ethereum, τα βασικά δομικά λειτουργικά στοιχεία μέχρι την δεύτερη γενιά του Blockchain [13][14] είναι τα ακόλουθα:

"Εξόρυξη" (Mining): Η «εξόρυξη» είναι η διαδικασία επικύρωσης συναλλαγών και προσθήκης τους σε ένα Block του Blockchain. Οι miners λύνουν σύνθετα μαθηματικά προβλήματα για να προσθέσουν νέα τμήματα, εξασφαλίζοντας έτσι την ασφάλεια του δικτύου.

Δημιουργία νομισμάτων/Token (Minting): Αντίθετα με την "εξόρυξη", η δημιουργία νομισμάτων χρησιμοποιείται σε blockchain τύπου Proof-of-Stake. Οι επαληθευτές επιλέγονται για να δημιουργούν νέα τμήματα με βάση τον αριθμό των νομισμάτων που κατέχουν και είναι πρόθυμοι να "στοιχηματίσουν" ως εγγύηση (collateral). Η δημιουργία νομισμάτων (Minting) συμβάλλει

στην ασφάλεια του δικτύου και μειώνει την ενεργειακή κατανάλωση που σχετίζεται με την εξόρυξη.

Κρυπτογραφία Hash: Η κρυπτογραφία hash είναι η βάση για την ακεραιότητα και την ασφάλεια των δεδομένων σε ένα σύστημα blockchain. Οι hash συναρτήσεις μετατρέπουν τις συναλλαγές σε συμβολοσειρές χαρακτήρων σταθερού μεγέθους που είναι σχεδόν αδύνατο να αναστραφούν πίσω στο αρχικό μόνωμο, πριν την κρυπτογράφηση.

Πρωτόκολλο Συναίνεσης (Consensus Protocol): Αυτό είναι ένα σύστημα που χρησιμοποιείται για να επιτευχθεί συμφωνία για μια μοναδική τιμή δεδομένων ανάμεσα σε διανεμημένα συστήματα. Είναι βασικό για τη διατήρηση της ακεραιότητας και της ομοιομορφίας του καθολικού μητρώου (ledger) [7].

Διανεμημένο Δίκτυο Ομότιμων (Distributed P2P Network): Τα δίκτυα ομότιμων (P2P) αποτελούνται από υπολογιστές που διανέμουν τα καθήκοντα μεταξύ τους. Αυτό εξασφαλίζει ότι καμία μεμονωμένη οντότητα δεν έχει έλεγχο σε ολόκληρο το blockchain.

Αμετάβλητο Καθολικό Μητρώο (Immutable Ledger): Μόλις τα δεδομένα προστεθούν στο Blockchain, δεν μπορούν να αλλαχθούν. Αυτή η λειτουργία είναι κλειδί για εφαρμογές που απαιτούν εμπιστοσύνη και καταγραφή ιστορικών στοιχείων.

Δημόσια και Ιδιωτικά Κλειδιά (Public and Private Keys): Αυτά τα κρυπτογραφικά κλειδιά είναι ουσιαστικά για την έναρξη και την επικύρωση συναλλαγών σε ένα σύστημα blockchain. Τα δημόσια κλειδιά είναι ορατά σε όλους τους συμμετέχοντες του δικτύου, ενώ τα ιδιωτικά κλειδιά παραμένουν εμπιστευτικά.

Πισίνα Συναλλαγών (Transaction Pool): Αυτή είναι μια δομή δεδομένων που "κρατά" τις συναλλαγές που περιμένουν να προστεθούν στο Blockchain. Λειτουργεί ως μηχανισμός ταξινόμησης πριν από την επικύρωση.

Μπλοκ σε Δομή Αλυσίδας (Blocks in Chain Structure): Το blockchain αποτελείται από Blocks που περιέχουν παρτίδες / πακέτα συναλλαγών. Κάθε τομέας συνδέεται με γραμμική, χρονολογική σειρά, δημιουργώντας μια αλυσίδα.

Επικύρωση (Validation): Οι επαληθευτές ή οι "miners" ελέγχουν τις συναλλαγές για την εγκυρότητά τους πριν προστεθούν σε ένα νέο Μπλοκ / Block. Η επικύρωση διασφαλίζει ότι το blockchain παραμένει ασφαλές και ελεύθερο από απάτες.

Έξυπνα Συμβόλαια (Smart Contracts): Πρόκειται για αυτόματα εκτελούμενα συμβόλαια με τους όρους άμεσα γραμμένους σε μορφή κώδικα. Τα έξυπνα συμβόλαια (Smart Contracts) επιτρέπουν αμεσότερες και διαφανείς συναλλαγές χωρίς την ανάγκη για τρίτους διαμεσολαβητές [22].

Τα παραπάνω στοιχεία και χαρακτηριστικά είναι ζωτικά για τις τεχνολογίες blockchain και τις εφαρμογές τους.

2.4 Hashing και SHA-256

Πηγές κεφαλαίου 2.4 [15] και [16].

Ένα από τα βασικά εργαλεία στο οποίο οφείλεται η ύπαρξη των Blockchain είναι το «hashing». Το «hashing» είναι μια μέθοδος μετατροπής δεδομένων εισόδου σε μοναδικούς αλφαριθμητικούς κωδικούς. Η διαδικασία είναι μονόδρομη, το οποίο σημαίνει ότι είναι πολύ δύσκολο να ανακτηθούν τα αρχικά δεδομένα από το παραγόμενο hash. Τα περισσότερα Blockchain χρησιμοποιούν κάποια έκδοση του SHA (Secure Hash Algorithm) με τα περισσότερα Blockchain να χρησιμοποιούν το SHA-2 και συγκεκριμένα το SHA-256 (256 bits hash). Το SHA-2 εμπεριέχει και το SHA-384 (384 bits hash) και SHA-512 (512 bits hash) και είναι βασισμένο στην Merkle–Damgård hash function (Η κατασκευή Merkle–Damgård ή η συνάρτηση κατακερματισμού Merkle–Damgård είναι μια μέθοδος δημιουργίας κρυπτογραφικών συναρτήσεων κατακερματισμού ανθεκτικών σε σύγκρουση από συναρτήσεις μονόδρομης συμπίεσης ανθεκτικές σε σύγκρουση [11]. Γενικότερα ο αλγόριθμος SHA-256 (Secure Hash Algorithm 256-bit) είναι ένας από τους πιο διαδεδομένους κρυπτογραφικούς αλγόριθμους για τη δημιουργία των λεγόμενων «hashes». Η διαδικασία του hashing αποτελεί μια τεχνική με την οποία δημιουργούνται μοναδικοί κωδικοί από δεδομένα εισόδου με παράγωγο hash που φέρει ένα συγκεκριμένο μέγεθος. Ο αλγόριθμος χρησιμοποιεί 256 bits στη μνήμη και το τελικό Hash έχει μήκος 64 χαρακτήρων σε δεκαεξαδική μορφή (0123456789ABCDEF) - 4 bit * 64 characters = 256bits (32-byte).

Γενικότερά για την διαδικασία του Hashing υπάρχουν πολλοί αλγόριθμοι εκτός του SHA-256. Μερικοί από αυτούς έχουν κενά ασφαλείας /είναι μη-ασφαλής όμως, όπως οι MD2, MD4, MD5,

SHA-0, SHA-1, Panama, HAVAL και Tiger και δεν επιλέγονται. Αυτοί οι αλγόριθμοι που προαναφέρθηκαν δεν χρησιμοποιούνται και θεωρούνται πως έχουν κρυπτογραφικές αδυναμίες. Συγκεκριμένα, αυτοί οι αλγόριθμοι είτε δεν προσφέρουν γενικότερα υψηλή ασφάλεια ή έχουν βρεθεί τρόποι για την δημιουργία συγκρούσεων και πολλαπλών διαφορετικών αρχείων που δίνουν το ίδιο hash. Η σύγκρουση στο hashing είναι όταν παράγεται ίδια τιμή κατακερματισμού (hash) για δύο διαφορετικές εισόδους, κάτι εξαιρετικά απίθανο για ασφαλείς αλγόριθμους παραγωγής κλειδιών hash, ειδικά για τους ανθεκτικούς-στη-σύγκρουση αλγόριθμους. Στους αλγόριθμους που δεν έχουν βρεθεί κενά ασφαλείας και συγκρούσεις ανήκουν οι SHA-2, SHA-3, BLAKE2, RIPEMD όπως και ο Whirlpool των Vincent Rijmen (συνδημιουργός του Προηγμένου Προτύπου Κρυπτογράφησης / Advanced Encryption Standard) και του Paulo S. L. M. Barreto, αλλά και ο SM3 της κινεζικής κυβέρνησης που θυμίζει τον SHA-256 και ο ρωσικός GOST. Από αυτούς οι SHA-2, SHA-3, BLAKE2, RIPEMD είναι δωρεάν για δημόσια χρήση και η ασφάλεια του κάθε αλγορίθμου ορίζεται τόσο από την δομή και την λειτουργία του ίδιου του αλγορίθμου όσο και από το μέγεθος των τελικών κλειδιών που παράγονται, με τον SHA-512 να είναι περισσότερο ασφαλής από τον SHA-256.

Έξω από τα πλαίσια των συμβατικών αλγορίθμων που προσφέρονται, μπορεί κανείς να σχεδιάσει δικό του αλγόριθμο. Ο σχεδιασμός ενός προσαρμοσμένου αλγορίθμου για hashing απαιτεί την τήρηση ορισμένων αρχών για τη διασφάλιση της αποτελεσματικότητας και της ασφάλειάς του. Παρακάτω παρατίθενται οι βασικές απαιτήσεις:

Μονόδρομη Φύση: Ένα βασικό χαρακτηριστικό ενός ασφαλούς αλγορίθμου κατακερματισμού είναι η μονόδρομη φύση του. Αφού δημιουργηθεί ο κατακερματισμός, πρέπει να είναι υπολογιστικά ανέφικτο να ανακτηθούν τα αρχικά δεδομένα από αυτόν.

Ντετερμινιστική Έξοδος: Ένας αλγόριθμος κατακερματισμού (hash algorithm) πρέπει να είναι ντετερμινιστικός, πράγμα που σημαίνει ότι τα ίδια δεδομένα θα παράγουν πάντα τον ίδιο κατακερματισμό.

Γρήγορος Υπολογισμός: Η αποτελεσματικότητα είναι ένα σημαντικό στοιχείο, ειδικά για εφαρμογές που χειρίζονται μεγάλο όγκο δεδομένων. Η συνάρτηση κατακερματισμού (hash function) πρέπει να είναι σε θέση να υπολογίζει τον κατακερματισμό (να υπολογίζει το hash) γρήγορα.

Φαινόμενο Avalanche: Μια μικρή αλλαγή στα εισαγόμενα δεδομένα πρέπει να έχει ως αποτέλεσμα ένα σημαντικά διαφορετικό hash.

Αντοχή σε Συγκρούσεις: Ο αλγόριθμος πρέπει να έχει επαρκώς μεγάλο εύρος πιθανών τιμών ώστε να ελαχιστοποιείται η πιθανότητα δύο διαφορετικών εισόδων να παράγουν τον ίδιο κατακερματισμό.

Τηρώντας αυτές τις αρχές, ένας αλγόριθμος κατακερματισμού μπορεί να επιτύχει μια ισορροπία ανάμεσα στην ταχύτητα, την ασφάλεια και τη λειτουργικότητα.

2.5 Δίκτυα κατανεμημένων ledger με τεχνολογία Blockchain

Σαν καθολικός ή γενικός ορισμός, η τεχνολογία blockchain είναι ένα λειτουργικό κομμάτι της τεχνολογίας διανεμημένου ψηφιακού καθολικού ή Ledger (Distributed Ledger Technology, DLT) [8]. Αυτό το διανεμημένο ledger μπορεί να βρίσκει εφαρμογή σε ένα πλήθος διαφορετικών δικτύων, με κάθε δίκτυο να διαφέρει ανάλογα με την εφαρμογή του [28].

Τα δίκτυα κατανεμημένων ledger με τεχνολογία Blockchain μπορούν να κατηγοριοποιηθούν κυρίως στους ακόλουθους βασικούς τύπους:

1. Δημόσια Blockchains

Προσβασιμότητα: Ανοιχτή σε όλους.

Αποκέντρωση: Υψηλού βαθμού αποκέντρωση, χωρίς την ύπαρξη κάποιου κεντρικού ή μοναδικού σημείου ελέγχου.

Παραδείγματα: Bitcoin, Ethereum.

Εφαρμογές: Κρυπτονομίσματα, αποκεντρωμένες εφαρμογές (dApps).

2. Ιδιωτικά Blockchains

Προσβασιμότητα: Συνήθως περιορισμένη σε ορισμένους χρήστες ή οργανισμούς.

Αποκέντρωση: Συνήθως είναι κεντροποιημένα εντός ενός οργανισμού ή ενός δικτύου με ιδιωτικούς Servers.

Παραδείγματα: Hyperledger, Corda.

Εφαρμογές: Επαγγελματικές συμβάσεις, διαχείριση δεδομένων για τον κλάδο της υγείας και ιατρικής.

3. Συνεταιριστικά Blockchains (Consortium)

Προσβασιμότητα: Ελεγχόμενη από ένα σύνολο οργανισμών.

Αποκέντρωση: Μερικώς αποκεντρωμένα, με τον έλεγχο να διαμοιράζεται ανάμεσα στους συμμετέχοντες οργανισμούς.

Παραδείγματα: Ripple, Quorum.

Εφαρμογές: Διαχείριση εφοδιαστικής αλυσίδας, ολιστικές δημοκρατίες, εν μέρει αποκεντροποιημένη παραγωγή κεντροποιημένων αγαθών

4. Ομοσπονδιακά Blockchains (Federated Blockchains)

Προσβασιμότητα: Ελεγχόμενη από μια ομάδα οργανισμών.

Αποκέντρωση: Μερικώς αποκεντρωμένα, καθώς η διαχείριση και ο έλεγχος μοιράζονται μεταξύ των συμμετεχόντων οργανισμών. Πιο αποκεντρωμένα από τα ιδιωτικά αλλά λιγότερο από τα δημόσια.

Παραδείγματα: R3 Corda, IBM Hyperledger Fabric.

Εφαρμογές: Τραπεζικός τομέας, διαχείριση εφοδιαστικής αλυσίδας.

5. Sidechains

Προσβασιμότητα: Εξαρτάται από την κύρια αλυσίδα.

Αποκέντρωση: Εξαρτάται, επίσης, από την κύρια αλυσίδα.

Παραδείγματα: Polygon, Gnosis Chain.

Εφαρμογές: Λύσεις ως προς την κλιμακωσιμότητα, έξυπνα συμβόλαια.

6. Υβριδικά Blockchains (Hybrid)

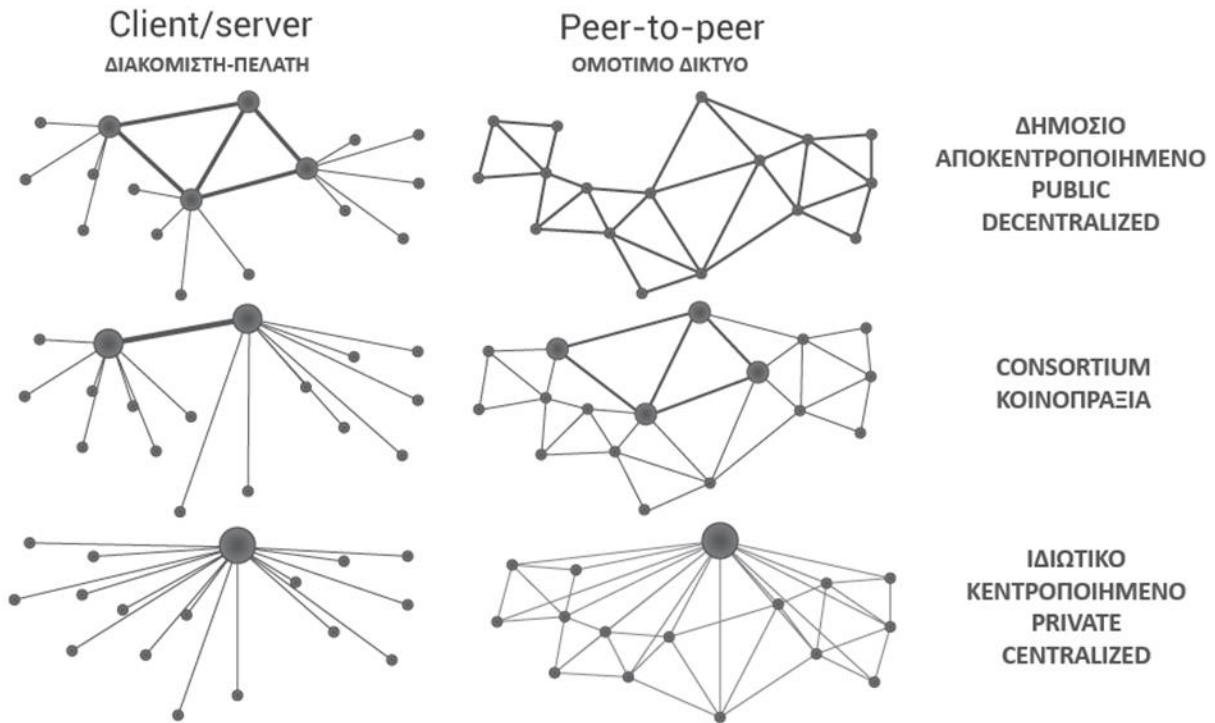
Προσβασιμότητα: Συνδυάζει χαρακτηριστικά δημόσιων και ιδιωτικών blockchains.

Αποκέντρωση: Μπορεί να ποικίλει ευρέως ανάλογα με την εφαρμογή.

Παραδείγματα: Dragonchain, Hedera Hashgraph.

Εφαρμογές: Επιχειρηματικές λύσεις, δημόσιες υπηρεσίες

Κάθε τύπος εξυπηρετεί διαφορετικές ανάγκες και προσφέρει διάφορα επίπεδα ελέγχου, ασφάλειας και προσβασιμότητας.



Διάφορα δίκτυα ανάμεσα σε χρήστες, διακομιστές και κόμβους, που προκύπτουν μέσα από τα διανεμημένα ledger.

2.6 Έξυπνη Σύμβαση/Συμβόλαιο (Smart Contract)

Ένα smart contract είναι ένα αυτό-εκτελούμενο συμβόλαιο, όπου οι όροι της συμφωνίας είναι γραμμένοι σε μορφή κώδικα. Αυτά τα συμβόλαια αναπτύσσονται σε ένα Blockchain Network, όπου εκτελούνται αυτόματα και ελέγχουν και καταγράφουν τα σχετικά γεγονότα και ενέργειες σύμφωνα με τους κωδικοποιημένους όρους. Όταν ένα smart contract δημιουργηθεί στο

blockchain, γίνεται αμετάβλητο, διασφαλίζοντας την ακεραιότητα και την ασφάλεια της συμφωνίας. Η αυτόματη εκτέλεση λογικής και η υπολογιστική φύση της σύμβασης εξαλείφει την ανάγκη για τρίτους ενδιάμεσους, μειώνοντας τα κόστη και τα πιθανά ανθρώπινα λάθη και οι συναλλαγές είναι ανιχνεύσιμες και μη αναστρέψιμες [21][22][24][25].

Η βασική δομή μιας έξυπνης σύμβασης περιλαμβάνει τα εξής στοιχεία:

1. **Δεδομένα:** Περιέχουν τις πληροφορίες των μεταβλητών και τις σταθερές που θα χρησιμοποιηθούν κατά την εκτέλεση της σύμβασης.
2. **Συνθήκες:** Οι λογικές συνθήκες που πρέπει να πληρούνται για να εκτελεστούν οι λειτουργίες της σύμβασης.
3. **Λειτουργίες:** Οι ενέργειες ή οι λειτουργίες (functions) που εκτελούνται χωρίς σφάλματα όταν οι συνθήκες πληρούνται.

Χαρακτηριστικά Έξυπνων Συμβάσεων (με βάση το Ethereum Network)

1. **Αυτό-εκτέλεση:** Εκτελούνται αυτόματα όταν πληρούνται οι προκαθορισμένες συνθήκες / όροι.
2. **Αμετάβλητες:** Μόλις απορροφηθούν / ενσωματωθούν μέσα στο blockchain, δεν μπορούν να τροποποιηθούν, εξασφαλίζοντας την ακεραιότητα των δεδομένων.
3. **Διαφάνεια:** Όλες οι δημόσιες λειτουργίες, ο κώδικας* και οι συναλλαγές των συμβάσεων είναι διαθέσιμα στο blockchain και είναι ορατά σε όλους. (*αν και όσο το επιλέξει ο προγραμματιστής)
4. **Αξιοπιστία:** Δεν υπάρχει ανάγκη για τρίτους μεσολαβητές, μειώνοντας τον κίνδυνο ανθρώπινου λάθους ή απάτης.
5. **Ασφάλεια:** Η κρυπτογράφηση και η αποκεντρωμένη φύση του blockchain προσφέρουν υψηλό επίπεδο ασφάλειας.
6. **Αποδοτικότητα:** Μειώνουν το κόστος και το χρόνο που απαιτείται σε σχέση με την διαδικασία της παραδοσιακής δημιουργίας και εκτέλεσης συμβάσεων.

Τα smart contracts έχουν ποικίλες εφαρμογές σε διάφορες βιομηχανίες. Στον τομέα των χρηματοοικονομικών, χρησιμοποιούνται για την αυτοματοποίηση συναλλαγών, όπως την μεταφορά περιουσιακών στοιχείων ή την εκκαθάριση συναλλαγών. Για παράδειγμα, στην ακίνητη περιουσία, ένα smart contract μπορεί να αυτοματοποιήσει τη διαδικασία μεταβίβασης

της ιδιοκτησίας μόλις ληφθεί η πληρωμή. Στη διαχείριση της εφοδιαστικής αλυσίδας, μπορούν να παρακολουθούν την κίνηση των αγαθών και να διασφαλίζουν ότι οι πληρωμές ολοκληρώνονται μόνο όταν πληρούνται οι προκαθορισμένοι όροι. Άλλα παραδείγματα αποτελούν τα συστήματα ψηφοφορίας και οι αποκεντρωμένες εφαρμογές (DApps) που λειτουργούν σε πλατφόρμες blockchain όπως το Ethereum.

Η ενσωμάτωση των smart contracts με την τεχνολογία blockchain είναι σημαντική για τη λειτουργικότητά τους. Το blockchain παρέχει ένα αποκεντρωμένο και διανεμημένο καθολικό (Decentralized Distributed Ledger) που καταγράφει όλες τις συναλλαγές με ασφαλή και διαφανή τρόπο. Όταν ένα smart contract αναπτυχθεί, αποθηκεύεται στο blockchain, καθιστώντας το προσβάσιμο σε όλους τους συμμετέχοντες στο δίκτυο. Κάθε φορά που εκτελείται μια συναλλαγή σύμφωνα με τους όρους του συμβολαίου, αυτή επαληθεύεται από τον μηχανισμό συναίνεσης του blockchain, διασφαλίζοντας την εμπιστοσύνη και τη διαφάνεια. Αυτή η συνεργασία μεταξύ των smart contracts και του blockchain επιτρέπει τη δημιουργία συστημάτων χωρίς την ανάγκη επίβλεψης από κάποια κεντροποιημένη οντότητα, όπου τα μεμονωμένα μέρη μπορούν να αλληλεπιδρούν άμεσα χωρίς να χρειάζεται να εμπιστεύονται ο ένας τον άλλον ή μια κεντρική αρχή.

Τα smart contracts έχουν βασικά χαρακτηριστικά που τα καθιστούν ιδιαίτερα ισχυρά και χρήσιμα. Είναι αυτόνομα, καθώς εκτελούνται αυτόματα όταν πληρούνται οι προϋποθέσεις. Είναι αμετάβλητα, εξασφαλίζοντας ότι μόλις αναπτυχθούν, ο κώδικάς τους δεν μπορεί να αλλάξει, αποτρέποντας την παραποίηση και την απάτη. Η διαφάνεια είναι ένα άλλο σημαντικό χαρακτηριστικό, καθώς οι όροι του συμβολαίου μπορούν να είναι ορατοί και επαληθεύσιμοι από όλους τους συμμετέχοντες στο blockchain. Επιπλέον, τα smart contracts προσφέρουν υψηλή ασφάλεια λόγω των κρυπτογραφικών αρχών που διέπουν την τεχνολογία blockchain, καθιστώντας τα ανθεκτικά σε hacking και άλλες κακόβουλες δραστηριότητες. Όλα αυτά τα χαρακτηριστικά τα καθιστούν άρτια για την δημιουργία ενός συστήματος διαφύλαξης και συναλλαγής οντοτήτων με μοναδικά κριτήρια, όπως τα Non Fungible Token (NFT).

Στον τομέα των μη ανταλλάξιμων Token (NFTs), τα smart contracts παίζουν καθοριστικό ρόλο. Τα NFTs είναι μοναδικά ψηφιακά περιουσιακά στοιχεία που αντιπροσωπεύουν την ιδιοκτησία ενός συγκεκριμένου αντικειμένου ή περιεχομένου, όπως τέχνη, μουσική ή συλλεκτικά αντικείμενα. Τα smart contracts μπορούν να χρησιμοποιηθούν για τη δημιουργία και τη

διαχείριση των NFTs, καθορίζοντας τους κανόνες ιδιοκτησίας και μεταφοράς που ενσωματώνονται στα μεταδεδομένα του NFT. Όταν δημιουργείται ένα NFT, το smart contract καταγράφει το μοναδικό αναγνωριστικό του στο blockchain, διασφαλίζοντας την προέλευση και την αυθεντικότητα του. Οι συναλλαγές που αφορούν τα NFTs, όπως οι πωλήσεις ή οι μεταφορές, εκτελούνται μέσω αυτών των smart contracts, παρέχοντας έναν ασφαλές και διαφανή περιβάλλον / σύστημα για τη διαχείριση της ψηφιακής ιδιοκτησίας των ψηφιακών αντικειμένων με μοναδικά τεκμήρια.

2.7 NFT και το IPFS

Τα NFT, ή Non-Fungible Tokens (Μη Ανταλλάξιμα Διακριτικά), αποτελούν ψηφιακά περιουσιακά στοιχεία που αντιπροσωπεύουν μοναδικά αντικείμενα στον ψηφιακό και εικονικό κόσμο. Σε αντίθεση με τα παραδοσιακά κρυπτονομίσματα όπως το Bitcoin ή το Ethereum (Ether), τα οποία είναι ανταλλάξιμα και έχουν την ίδια αξία, κάθε NFT είναι μοναδικό και έχει διαφορετική αξία. Αυτή η μοναδικότητα προκύπτει από τη μοναδική φύση των NFT, αφού δεν αντιστοιχούν σε χρηματικές υποδιαιρέσεις αλλά σε μια μοναδική ψηφιακή υπογραφή που αναλογεί σε μια οντότητα, όπως ψηφιακά έργα τέχνης, συλλεκτικά αντικείμενα και σε οποιοδήποτε αντικείμενο μπορεί να ψηφιοποιηθεί ή να θεωρηθεί μια μεμονωμένη οντότητα. [23][25] [24].

Η δομή ενός NFT βασίζεται στην τεχνολογία blockchain, η οποία εξασφαλίζει την ασφάλεια και την ακεραιότητα του ψηφιακού περιουσιακού στοιχείου. Κάθε NFT περιέχει μεταδεδομένα που περιγράφουν το μοναδικό αντικείμενο που αντιπροσωπεύει. Αυτά τα μεταδεδομένα μπορεί να περιλαμβάνουν το όνομα του καλλιτέχνη, την ημερομηνία δημιουργίας, τις ιδιότητες του αντικειμένου και άλλα χαρακτηριστικά που το καθιστούν μοναδικό. Τα NFT δημιουργούνται μέσω μιας διαδικασίας που ονομάζεται "minting", η οποία επιβεβαιώνει την μοναδικότητά τους και τα καταχωρεί στο blockchain.

Η κύρια λειτουργία ενός NFT είναι η παροχή απόδειξης της ιδιοκτησίας και γνησιότητας για μοναδικά ψηφιακά αντικείμενα. Όταν κάποιος αγοράζει ένα NFT, αποκτά την κυριότητα του συγκεκριμένου αντικειμένου που το NFT αντιπροσωπεύει ή την κυριότητα των ψηφιακών δεδομένων και λειτουργιών που το NFT εμπεριέχει. Η κυριότητα αυτή μπορεί να μεταφερθεί από

έναν χρήστη σε έναν άλλον μέσω του blockchain, εξασφαλίζοντας τη διαφάνεια και την ασφάλεια των συναλλαγών καθώς και την συνέχεια των οντοτήτων. Επιπλέον, τα NFT μπορούν να χρησιμοποιηθούν σε διάφορες πλατφόρμες και ψηφιακές αγορές (digital markets) για την πώληση, την αγορά και την ανταλλαγή άλλων ψηφιακών περιουσιακών στοιχείων ή οντοτήτων.

Τα NFT διαφέρουν από τα Fungible Tokens (ανταλλάξιμα διακριτικά) κυρίως λόγω της μοναδικότητάς τους. Ενώ τα Fungible Tokens, όπως το Bitcoin, είναι αμοιβαία ανταλλάξιμα και έχουν την ίδια αξία, τα NFT είναι μοναδικά και δεν μπορούν να αντικατασταθούν ίσα το ένα με το άλλο. Για παράδειγμα, ένα Bitcoin μπορεί να αντικατασταθεί με ένα άλλο Bitcoin χωρίς καμία απώλεια αξίας, ενώ ένα NFT αντιπροσωπεύει ένα μοναδικό ψηφιακό έργο τέχνης που δεν μπορεί να αντικατασταθεί με ένα άλλο έργο τέχνης χωρίς να αλλάξει η αξία του.

Τα NFT χρησιμοποιούνται για να αντιπροσωπεύουν μοναδικά ψηφιακά περιουσιακά στοιχεία, όπως ψηφιακά έργα τέχνης, μουσική, βίντεο, μοναδικά στοιχεία βιντεοπαιχνιδιών και άλλα συλλεκτικά αντικείμενα. Αυτά τα περιουσιακά στοιχεία μπορούν να αγοραστούν, να πωληθούν και να ανταλλαχθούν σε διάφορες πλατφόρμες που υποστηρίζουν την τεχνολογία NFT. Επιπλέον, τα NFT μπορούν να χρησιμοποιηθούν σαν «εισιτήρια / πάσο» για την παροχή πρόσβασης σε αποκλειστικό περιεχόμενο ή εμπειρίες, όπως εκδηλώσεις, συναυλίες ή διαδικτυακά παιχνίδια. Με αυτόν τον τρόπο, τα NFT δίνουν μια νέα μετα-διάσταση στα αγαθά που συσχετίζονται με αυτά.

Με τη χρήση των NFT για την αντιπροσώπευση μοναδικών ψηφιακών περιουσιακών στοιχείων, προκύπτει η ανάγκη για ασφαλή και αξιόπιστη αποθήκευση των δεδομένων που σχετίζονται με αυτά τα περιουσιακά στοιχεία. Τα μεταδεδομένα και το περιεχόμενο που αντιπροσωπεύουν τα NFT πρέπει να είναι προσβάσιμα και αμετάβλητα, ώστε να διατηρείται η αξία και η γνησιότητα των ψηφιακών περιουσιακών στοιχείων. Μια πιθανή και αποκεντροποιημένη λύση είναι το σύστημα IPFS (Inter Planetary File System / Διαπλανητικό Σύστημα Αρχείων), το οποίο προσφέρει μια αποκεντρωμένη και ασφαλή λύση για την αποθήκευση αυτών των δεδομένων χρησιμοποιώντας την τεχνολογία peer-to-peer (P2P).

Το Διαπλανητικό Σύστημα Αρχείων IPFS, ή Inter Planetary File System, είναι ένα αποκεντρωμένο πρωτόκολλο αποθήκευσης και κοινής χρήσης αρχείων που χρησιμοποιεί μια κατακεντρωμένη προσέγγιση (P2P) για τη διαχείριση δεδομένων [26]. Σε αντίθεση με τα παραδοσιακά συστήματα αποθήκευσης, το IPFS επιτρέπει στους χρήστες να αποθηκεύουν και να ανακτούν αρχεία από διάφορους κόμβους στο δίκτυο, εξασφαλίζοντας έτσι μεγαλύτερη

ανθεκτικότητα, ασφάλεια και ταχύτητα στη μετάδοση δεδομένων. Η δομή του IPFS βασίζεται σε ένα μοναδικό αναγνωριστικό για κάθε αρχείο, γνωστό ως "content identifier" (CID), που επιτρέπει την ακριβή ανεύρεση και ανάκτηση του αρχείου από οπουδήποτε στο δίκτυο.

Το IPFS χρησιμοποιείται συχνά σε συνδυασμό με τα NFT για την αποθήκευση των μεταδεδομένων και του περιεχομένου που αυτά αντιπροσωπεύουν. Δεδομένου ότι τα NFT αποθηκεύουν μοναδικά ψηφιακά περιουσιακά στοιχεία, είναι σημαντικό να διασφαλιστεί ότι τα δεδομένα που σχετίζονται με αυτά τα περιουσιακά στοιχεία είναι ασφαλή, προσβάσιμα και αμετάβλητα. Το IPFS προσφέρει αυτή τη δυνατότητα μέσω της κατακευκτικής φύσης του, επιτρέποντας την αποθήκευση των αρχείων σε διάφορους κόμβους και την ανάκτησή τους με βάση τον μοναδικό CID. Αυτό εξασφαλίζει ότι τα ψηφιακά περιουσιακά στοιχεία που αντιπροσωπεύονται από NFT παραμένουν αμετάβλητα.

2.8 Μεταφέροντας όντα ανάμεσα σε ψηφιακούς κόσμους χρησιμοποιώντας τις τεχνολογίες Blockchain

Η επιλογή του κατάλληλου τύπου/αρχιτεκτονικής τεχνολογίας Ledger με τεχνολογία Blockchain για την αποθήκευση στοιχείων βιντεοπαιχνιδιών και συναλλαγών εντός του παιχνιδιού είναι μια καίρια διαδικασία. Αυτό συμβαίνει επειδή τα δεδομένα και η προγραμματιστική λογική του ίδιου του Blockchain και του Ledger πρέπει να διασυνδέονται αρμονικά με τα εσωτερικά συστήματα των μεμονωμένων παιχνιδιών.

Το ίδιο το Ledger αποτελεί ένα μετα-σύστημα του κόσμου του παιχνιδιού, ενώ οι εσωτερικές λειτουργίες του δικτύου του Blockchain, των έξυπνων συμβολαίων (smart contracts) και των dApps αποτελούν στοιχεία προέκτασης του παιχνιδιού ή στοιχεία μεταπρογραμματισμού. Όταν μια οντότητα που μεταφέρεται μέσα σε ένα smart contract εμπεριέχει νέο κώδικα ή νέες λειτουργίες και το παιχνίδι πρέπει να έρθει σε θέση να τις αφομοιώσει στον κώδικά του, τότε το smart contract θεωρείται ως μεταπρόγραμμα, που μεταφέρει δεδομένα προς το παιχνίδι. Για παράδειγμα, όταν ένα βιντεοπαιχνίδι ακολουθεί τη μεθοδολογία σχεδίασης Data Driven Design και χρησιμοποιεί τα μεταδεδομένα από μια έξυπνη σύμβαση/συμβόλαιο (smart contract) για να επηρεάσει την δομή του, τότε οι διαδικασίες που εκτελούνται και μεταποιούν τα δεδομένα έξω από τον χώρο του παιχνιδιού, όπως για παράδειγμα εντός της εικονικής μηχανής/εξομοίωσης συστήματος του Ethereum, θεωρούνται συστήματα μεταπρογραμματισμού που επηρεάζουν

δυναμικά τον ψηφιακό κόσμο του παιχνιδιού [17] [19] και το παιχνίδι θεωρείται interoperable ή διαλειτουργικό. Στην απλούστερη μορφή χρήσης των blockchain στα παιχνίδια, απλά συναλλάσσονται πακέτα επικύρωσης γεγονότων παιχνιδιού (επικυρώνει gameplay events), και το Blockchain αποτελεί απλά ένα παράλληλο σύστημα [86].

Τα Blockchain με λειτουργίες smart contract που δυναμικά επηρεάζουν τις λειτουργίες του παιχνιδιού θεωρούνται αδιάσπαστο κομμάτι της φύσης του παιχνιδιού, αλλά ταυτόχρονα εξωγενή από το κεντρικό σώμα του. Σε αυτή την εφαρμογή, η φύση των smart contract ως μεταπρογραμματιστικά συστήματα επιβάλλει να φέρουν και τα λειτουργικά στοιχεία του παιχνιδιού, καθώς δεν έχουν πολύ διαφορετική φύση από το ίδιο το παιχνίδι ή τουλάχιστον πρέπει να έχουν αρκετά όμοια στοιχεία (components) για να μπορούν να επικοινωνούν με τον στατικό τελικό κώδικα πυρήνα (main core) του παιχνιδιού, που κατά βάση θα είναι γραμμένος σε μια στατική γλώσσα.

Μέσα από την έρευνα και την τεχνολογική ανασκόπηση που πραγματοποιήθηκε, έγινε σαφές πως η φύση και η αρχιτεκτονική του διανεμημένου Ledger, καθώς και το οικονομικό σύστημα που μπορεί να προκύψει από την εφαρμογή έξυπνων συμβολαίων και τεχνολογίας Blockchain, δεν μπορεί να απέχει πολύ από την ίδια την αρχιτεκτονική του παιχνιδιού. Για να υπάρχει ένας σημαντικός βαθμός αλληλεπίδρασης μεταξύ αυτών των δύο συστημάτων, και εφόσον ο μηχανισμός διασύνδεσης των δυναμικών στοιχείων με τα στατικά παιχνίδια ακολουθεί τη λογική του μεταπρογραμματισμού, πρέπει να είναι οι τεχνικές μεταπρογραμματισμού που χρησιμοποιούνται και εφαρμόζονται πάνω από την υπάρχουσα αρχιτεκτονική του παιχνιδιού αυτές που καθορίζουν την αρχιτεκτονική και τις προδιαγραφές του διανεμημένου Ledger.

2.9 Συστήματα Μεταπρογραμματισμού

Με βάση τον προσδιορισμό ενός εξωτερικού συστήματος Blockchain με smart contracts σαν σύστημα μεταπρογραμματισμού, εξετάστηκε, συνοπτικά, η δομή και η φύση των συστημάτων μεταπρογραμματισμού ως προς την γλώσσα C++, στην οποία είναι γραμμένα τα βιντεοπαιχνίδια.

Βιβλιογραφικά και πρακτικά, η γλώσσα C++ δεν υποστηρίζει από μόνη της την δυνατότητα του να προσαρμόζει τον κώδικα της δυναμικά - σε χρόνο εκτέλεσης ούτε την δυνατότητα κατασκευής κλάσεων και μηχανισμών σε χρόνο εκτέλεσης. Αυτό καθιστά την εφαρμογή

μεταπρογραμματιστικών συστημάτων περίπλοκη αφού σαν γενικός ορισμός, ο μεταπρογραμματισμός αναφέρεται στη δυνατότητα ενός προγράμματος να χειρίζεται άλλα προγράμματα ή να πραγματοποιεί αυτο-τροποποίηση και ο μεταπρογραμματισμός καλύπτει την ανάγκη για αναπαράσταση μεταπληροφορίας κατά τον χρόνο μεταγλώττισης (compilation time) ή κατά την λειτουργία του προγράμματος. Η υποστήριξη του μεταπρογραμματισμού στη C++ δεν είναι σχεδιασμένη δυνατότητα, αλλά κατάχρηση του μεταγλωττιστή (compiler hack / manipulation) της C++. Και ένα πρόγραμμα γραμμένο σε C++ θεωρείται πως έχει δυνατότητες μεταπρογραμματισμού όταν έχει εντοιχίσει τις ακόλουθες δυνατότητες (integrated the following functions):

Την δυνατότητα διαχείρισης άλλων προγραμμάτων ή του εαυτού του.

- Η εφαρμογή ενός συστήματος reflection / αντανάκλασης. Το reflection / αντανάκλαση είναι η ικανότητα ενός προγράμματος να διαχειρίζεται ως δεδομένα κάτι που αναπαριστά την κατάσταση του προγράμματος κατά τη διάρκεια της δικής του εκτέλεσης. Υπάρχουν δύο όψεις τέτοιου είδους διαχείρισης: η ενδοσκόπηση και η παρέμβαση. Και οι δύο όψεις απαιτούν έναν μηχανισμό για την κωδικοποίηση της κατάστασης εκτέλεσης ως δεδομένα, πράγμα που ονομάζεται reification (διαδικασία επαναυπόθεσης / πραγματοποίηση – η διαδικασία κατά την οποία αφηρημένες έννοιες μετατρέπονται σε δομές δεδομένων ή συγκεκριμένες αναπαραστάσεις που μπορούν να διαχειριστούν (manipulated) μέσα σε ένα πρόγραμμα).
 - Η ενδοσκόπηση είναι η ικανότητα ενός προγράμματος να παρατηρεί "τον εαυτό του" - "την κατάσταση του" και να "λογικοποιεί" αυτή, την ίδια του την κατάσταση.
 - Η παρέμβαση είναι η ικανότητα ενός προγράμματος να τροποποιεί την ίδια του την κατάσταση εκτέλεσης ή να αλλάζει την ίδια του την ερμηνεία ή έννοια.
- Την χρήση μετα-αντικειμένων ("Μετα-αντικειμενοστραφής προγραμματισμός" (Meta-Object Protocol ή MOP) / meta-objects) που αναπαριστούν μεθόδους, στοίβες εκτέλεσης (execution stacks), τον επεξεργαστή και σχεδόν όλα τα στοιχεία της γλώσσας και του περιβάλλοντος εκτέλεσης.

- Η χρήση μετα-επίπεδης αρχιτεκτονικής (metalevel), μια αρχιτεκτονική στην οποία πάνω από τον βασικό κώδικα υπάρχει ένα μεταεπίπεδο (metalevel) που παρέχει πληροφορίες σχετικά με επιλεγμένες ιδιότητες του συστήματος και καθιστά το λογισμικό αυτο-ενήμερο (self-aware), ενώ το βασικό επίπεδο περιλαμβάνει τη λογική της εφαρμογής.
 - Το βασικό επίπεδο είναι το επίπεδο όπου εκτελείται ο κύριος κώδικας και όπου υλοποιούνται οι κύριες λειτουργίες του λογισμικού. Το βασικό επίπεδο λειτουργεί υπό την παρακολούθηση και τον έλεγχο του μετα-επιπέδου.
 - Το μετα-επίπεδο μπορεί να παρακολουθεί και να ελέγχει τη συμπεριφορά του βασικού επιπέδου, επιτρέποντας προσαρμογές και βελτιώσεις σε πραγματικό χρόνο, επιτρέποντας στο λογισμικό να είναι αυτο-ενήμερο (self-aware).
- Η χρήση της τεχνικής του partial evaluation (μερική αξιολόγηση), μιας τεχνικής που αναγνωρίζει τα στατικά μέρη σε υπολογισμούς που αφορούν δεδομένα και μεταβλητές και προ-υπολογίζει το στατικό και μη δυναμικό μέρος της εξίσωσης και παράγει ειδικό κώδικα το οποίο την ώρα της εκτέλεσης μεταγλωττίζει και υπολογίζει το δυναμικό μέρος του προγράμματος [20].

Για την ορθή εφαρμογή και αφομοίωση ενός συστήματος μεταπρογραμματισμού όπως το Ethereum Network με τα smart contract, πρέπει να εφαρμοστούν μέθοδοι μεταπρογραμματισμού, όπως αυτοί που προαναφέρθηκαν, για την ανάπτυξη ασύγχρονων λειτουργιών ή την αναβάθμιση των στοιχείων του παιχνιδιού με βάση την κατάσταση της μνήμης του smart contract στο EVM.

2.10 Λειτουργίες και περιορισμοί των SMART CONTRACT για τα Βιντεοπαιχνίδια

2.10.1 Λειτουργίες και περιορισμοί

Εν έτει 2024, και κατά την έρευνα της αγοράς των βιντεοπαιχνιδιών και του κλάδου των κρυπτονομισμάτων, οι λειτουργίες των Smart Contract στα βιντεοπαιχνίδια φαίνεται να περιορίζονται σε διαδικασίες, δουλειές (workload / task) και ρουτίνες που θα έπαιρναν μέρος στο πίσω μέρος της υποδομής του διακομιστή (server back-end), χωρίς τις ταχύτητες που θα προσέφερε ένα δίκτυο από servers πίσω από GLSB Global Server Load Balancing (παγκόσμια

εξισορρόπηση φορτίου διακομιστών) και χωρίς τα ειδικά προσαρμοσμένα πρωτόκολλα για UDP sockets (υποδοχές UDP - User Datagram Protocol) που έχουν τα σύγχρονα παιχνίδια[39].

Έτσι, η χρήση των smart contract, με βάση τις -μέχρι τώρα- εφαρμογές, ανοίγει ένα πλαίσιο από ενδιαφέρουσες, όμως περιορισμένες, δυνατότητες για τις ψηφιακές εμπειρίες. Όπως η αποκεντρωμένη ιδιοκτησία αντικειμένων εντός παιχνιδιού (όπλα, κοστούμια, χαρακτήρες) μέσω του blockchain, επιτρέποντας στους παίκτες να κατέχουν πραγματικά τα αντικείμενα που κερδίζουν μέσα στις ψηφιακές εμπειρίες. Οι παίκτες μπορούν να αγοράζουν, να πωλούν και να ανταλλάσσουν αυτά τα περιουσιακά στοιχεία με ασφάλεια και διαφάνεια. Όλες οι συναλλαγές εντός παιχνιδιού καταγράφονται δημόσια, διασφαλίζοντας τη διαφάνεια και μειώνοντας την απάτη μέσω της χρήσης των smart contracts. Οι παίκτες μπορούν να δημιουργούν και να διαχειρίζονται περιεχόμενο (όπως mods – τροποποιήσεις), κερδίζοντας από αυτό, ενώ εξασφαλίζεται ότι οι δημιουργοί λαμβάνουν τα πνευματικά τους δικαιώματα. Η χρήση κρυπτονομισμάτων διευκολύνει τις αγορές και συναλλαγές εντός παιχνιδιού, επιτρέποντας στους παίκτες να λαμβάνουν αμοιβές και ανταμοιβές μέσω αυτόματων πληρωμών. Οι ανταμοιβές καταβάλλονται αυτόματα όταν ολοκληρώνονται αποστολές ή επιτυγχάνονται στόχοι, ενώ τα έπαθλα των τουρνουά διανέμονται επίσης αυτόματα στους νικητές. Τέλος, η διαχείριση συνδρομών και “premium” περιεχομένου, καθώς και η διαχείριση μικροσυναλλαγών, πραγματοποιούνται με ασφάλεια και διαφάνεια μέσω των smart contracts.

Οι μέχρι τώρα εφαρμογές επιτρέπουν είτε απλές συναλλαγές με κάποιες πράξεις να παίρνουν μέρος είτε το σύστημα να χρησιμοποιείται σαν ένα ψηφιακό ορθό που κρατάει ιστορικό από ιδιοκτησίες ψηφιακών οντοτήτων και αναφορές προς αυτές, είτε για απλή επεξεργασία δεδομένων [86]. Μελετώντας το δημοφιλέστερο σύστημα Blockchain με smart contracts, το Ethereum, γίνεται προφανές, μέσα από την έρευνα που πήρε μέρος, ότι υπάρχουν σοβαροί περιορισμοί βαθιά μέσα στην ίδια την αρχιτεκτονική του δικτύου.

2.10.2 Απροσπέλαστα σημεία περιορισμού

Απροσπέλαστα σημεία περιορισμού της εφαρμογής των αποκεντροποιημένων δημόσιων δικτύων Blockchain στα βιντεοπαιχνίδια.

- ❖ Καθυστερήσεις: Τα περισσότερα δίκτυα που απαιτούν την επαλήθευση των Block από πολλούς κόμβους δημιουργούν εργασίες (task) που μπορούν να παίρνουν από δευτερόλεπτα έως και λεπτά να επιλυθούν, χωρίς να είναι σαφές αν το «πακέτο» θα εξυπηρετηθεί από το

δίκτυο. Πολλά βιντεοπαιχνίδια εμπεριέχουν ρουτίνες που τραβάνε δεδομένα ανά tick του επεξεργαστή και η ταχύτητα είναι απόλυτα σημαντική εφόσον ένα frame είναι επιθυμητό να έχει ολοκληρωθεί σε λιγότερο από 16.7ms (για ένα παιχνίδι που τρέχει στα 60FPS) – και η επεξεργασία των δεδομένων είναι μόνο ένα μικρό κομμάτι της παραγωγής μιας εικόνας – πριν καν φτάσει στο rendering pipeline (pipeline απεικόνισης). Όταν ένα πακέτο συναλλαγής παίρνει δεκάδες δευτερόλεπτα να εξυπηρετηθεί και είναι άγνωστο το πότε και το αν θα εξυπηρετηθεί σωστά, τότε όλες οι διαδικασίες που αφορούν το δίκτυο πρέπει να είναι ασύγχρονες, και η απεικόνιση όπως και η λειτουργικότητα των στοιχείων που επικοινωνούν με το δίκτυο blockchain πρέπει να περιορίζεται σχεδιαστικά από τις καθυστερήσεις αυτές. Ο χρήστης δεν μπορεί να συναλλάξει το απόθεμα του (inventory) σε κάποιον μη-παίκτη πωλητή (NPC vendor) από τρισδιάστατο γραφικό περιβάλλον σε πραγματικό χρόνο, διότι η κίνηση του παίκτη θα παρέμενε εγκλωβισμένη στη ρουτίνα επαλήθευσης ολοκλήρωσης της συναλλαγής.

- ❖ Περιορισμένο μέγεθος των smart contract: Τα βιντεοπαιχνίδια χρησιμοποιούν περίπλοκους αλγόριθμους και παράγουν μεγάλο όγκο δεδομένων για να λειτουργήσουν, τα smart contract έχουν ένα συγκεκριμένο μέγιστο μέγεθος, που μπορούν να απαρτίσουν, το οποίο είναι μικρό. Είναι επόμενο, η πολύπλοκη λογική των παιχνιδιών να μην μπορεί να εντοιχιστεί σε τόσο μικρές προγραμματιστικές ρουτίνες. Έπειτα, τα ίδια τα παιχνίδια πολλές φορές απαιτούν την αρχικοποίηση χιλιάδων μεταβλητών και σταθερών, πολυδιάστατων πινάκων και πολύπλοκων structures – λειτουργίες οι οποίες γίνονται αδύνατες ως προς την υλοποίηση, στα περισσότερα αποκεντροποιημένα συστήματα με εφαρμογή smart contract. Οι περιορισμοί στο μέγεθος του smart contract καθιστούν δύσκολη την επέκταση και την προσθήκη νέων λειτουργιών στο παιχνίδι. Κάθε ενημέρωση ή επέκταση μπορεί να απαιτεί τη δημιουργία νέου smart contract, κάτι που περιπλέκει τη συντήρηση και την ανάπτυξη του παιχνιδιού.
- ❖ Πολλά παιχνίδια εμπεριέχουν εκατομμύρια γραμμές κώδικα και εκατοντάδες Gigabyte σε δεδομένα, αυτή η κλίμακα είναι έξω από τις δυνατότητες των δημόσιων δικτύων Blockchain και απαιτούν κεντροποιημένες εφαρμογές. Ιδιαίτερα, όταν εκατομμύρια χρήστες προσπαθούν να αποκτήσουν πρόσβαση ταυτόχρονα στα ίδια δεδομένα ή να ανεβάσουν συμβόλαια και να καλέσουν συμβόλαια με μεγάλο όγκο στην μνήμη και υπολογιστικό φόρτο εκτέλεσης.

- ❖ Η παραγωγή τυχαιότητας χωρίς oracle είναι αδύνατη και η επικοινωνία με τα oracle δημιουργεί ένα τρίτο επίπεδο ασάφειας (το δεύτερο είναι το ίδιο το σύστημα που εκτελεί τα έξυπνα συμβόλαια) πάνω από την εκτέλεση διαδικασιών εντός του παιχνιδιού, η οποία ήδη παρατείνεται σημαντικά. Ταυτόχρονα, καλώντας εξωτερικά συμβόλαια και υπηρεσίες, το κόστος εκτέλεσης του συμβολαίου αυξάνεται σημαντικά, όπως και το φόρτο του δικτύου. Το παρόν χαρακτηριστικό, μόνο του, κάνει την χρήση των μη κεντροποιημένων συστημάτων Blockchain απαγορευτική για τις περισσότερες διαδικασίες των μοντέρνων παιχνιδιών και του Generative AI, όπως έδειξε και η έρευνα που πήρε μέρος.

- ❖ Τα μοντέρνα παιχνίδια εμπεριέχουν πολύπλοκες διαδικασίες και οντότητες, οι οποίες καταλαμβάνουν εκατοντάδες Megabyte ή και πολλά Gigabyte σε αναφορές προς στοιχεία παιχνιδιού και χαρακτηριστικά αυτών. Σε δεύτερο βαθμό, τα στοιχεία παιχνιδιού που είναι παράγωγα διαδικαστικών αλγορίθμων γένεσης (procedural generation) και κατασκευάζουν την γεωμετρία τους κατά τον χρόνο εκτέλεσης, μπορούν να φέρουν πολύ μεγάλο όγκο δεδομένων των χαρακτηριστικών τους. Αν εκατομμύρια χρήστες ανεβάσουν τα δεδομένα της κατάστασης πολλαπλών παιχνιδιών με προσαρμοσμένες οντότητες στο δίκτυο, μόνο το σύστημα αναγνώρισης οντοτήτων θα απαιτούσε χώρο αποθήκευσης μεγέθους Cloud Server με την επεξεργαστική ισχύ ενός Data Center. Καθιστώντας απόλυτα ανέφικτη την χρήση ενός αποκεντροποιημένου συστήματος Blockchain με κόμβους που κρατούν αντίγραφα του όλου της κατάστασης του δικτύου. Ταυτόχρονα, οι λύσεις που δίνονται για την κλιμάκωση του δικτύου ή την εξωτερική αποθήκευση των δεδομένων περιπλέκουν ακόμη περισσότερο την διαδικασία επεξεργασίας των δεδομένων και της χρήσης τους μέσα σε παιχνίδια.

- ❖ Η βιομηχανία των παιχνιδιών αποτελεί έναν ραγδαία αναπτυσσόμενο κλάδο των 248,52 δισεκατομμυρίων δολαρίων εν έτει 2023. Τα αποκεντροποιημένα συστήματα Blockchain, αν και φέρουν μηχανισμούς ελέγχου της απάτης (double-spending) και φέρουν και την δυνατότητα ενός Hard-Fork, παρέχουν την εμπιστοσύνη μέσα από την δομή του κώδικα τους. Όμως, καμία φυσική οντότητα δεν φέρει την τελική ευθύνη σε περίπτωση κακόβουλων λειτουργιών ή σφαλμάτων που επηρεάζουν αρνητικά τα έσοδα της βιομηχανίας παιχνιδιών.

Όταν η Amazon ή η Microsoft παρέχουν υποδομή διακομιστών (server infrastructure), φέρουν και την ευθύνη ορθής λειτουργίας της υποδομής, συνεχούς uptime και ασφαλής διατήρησης των δεδομένων (έξω από τις ατομικές εφαρμογές κώδικα). Στην περίπτωση ενός καταστρεπτικού συμβάντος, οι συνήθεις νομικές διαδικασίες μπορούν να πάρουν μέρος. Κάτι τέτοιο, δεν είναι εφικτό σε ένα αποκεντροποιημένο δίκτυο.

- ❖ Η προγραμματιστική γλώσσα των περισσότερων συστημάτων Blockchain με δυνατότητες δημιουργίας smart contract δεν συμπίπτει με εκείνες των παιχνιδιών, πιο συγκεκριμένα την C++. Η εκτέλεση κώδικα του παιχνιδιού και η μεταποίηση αυτού, θα απαιτούσε την δημιουργία συστημάτων μεταποίησης και μεταγλώττισης του κώδικα. Αυτή θα ήταν, όμως, μία προβληματική διαδικασία ως προς την υλοποίηση της, με σημαντικές δυσκολίες, αφού τα op-code της εικονικής μηχανής πρέπει να συμπίπτουν με εκείνα του συστήματος εκτέλεσης του κώδικα του βιντεοπαιχνιδιού και να μεταφράζονται με την χρήση dynamic recompiler (ISA σε ISA μετάφραση) ή πρέπει να υπάρχει μια μηχανή για Emulation. Παράλληλα, για την τέλεση όλων αυτών των διαδικασιών απαιτείται η κατασκευή compiler . Στην περίπτωση που το παιχνίδι χρησιμοποιεί μια γλώσσα Script (lua, python) για τον μετα-προγραμματισμό του κώδικα του, τα ίδια προβλήματα συνεχίζουν να παρουσιάζονται εκτός και αν η γλώσσα script έχει φτιαχτεί συγκεκριμένα για την χρήση της σε κλασικούς υπολογιστές και στο EVM, χωρίς μεταποιήσεις. Τέτοια εργαλεία δεν βρέθηκαν για το EVM του Ethereum, την στιγμή τέλεσης της έρευνας.

2.11 ΤΑ ΣΤΟΙΧΕΙΑ ΕΝΟΣ BLOCKCHAIN ΣΥΣΤΗΜΑΤΟΣ ΜΕ ΛΕΙΤΟΥΡΓΙΕΣ ΜΕΤΑΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΓΙΑ ΤΑ ΒΙΝΤΕΟΠΑΙΧΝΙΔΙΑ

Μέσα από την έρευνα που πήρε μέρος [12][14][25][28], υπό τέλειες συνθήκες, ένα πιο ορθό σύστημα Blockchain φτιαγμένο για τα βιντεοπαιχνίδια θα ήταν ένα σύστημα ανά εφαρμογή - ειδικά φτιαγμένο και συμβατό στις λειτουργίες του κάθε παιχνιδιού, ξεχωριστά. Αν αυτό δεν είναι εφικτό ή χρήσιμο, τότε ένα διαλειτουργικό σύστημα που έρχεται σε επαφή με ένα πλήθος αγνώστων παιχνιδιών προ-απαιτεί την:

- native εκτέλεση κώδικα σε C++ από κεντροποιημένες υποδομές διακομιστών σε Data Center με πρόσβαση σε επεξεργαστές γραφικών για την χρήση των υπολογιστικών εργασιών, που είναι αποκλειστικά διαθέσιμες μέσα από αυτούς.
- υποστήριξη των κοινών γλωσσών Script που χρησιμοποιούν τα παιχνίδια, όπως της lua.
- παροχή της δυνατότητας υλοποίησης του πλήρους Back-End των δικτυακών λειτουργιών του παιχνιδιού εντός του συστήματος εκτέλεσης των smart contracts.
- πρόσβαση του Data Center σε υποδομές Cloud Storage / Cloud Server.
- χρήση ειδικών πρωτοκόλλων επικοινωνίας για την ελαχιστοποίηση του χρόνου απόκρισης του διακομιστή.
- ύπαρξη συστήματος Reflection και άλλων παρόμοιων δυνατοτήτων εκτέλεσης δυναμικού κώδικα για την υποστήριξη ρουτινών μεταπρογραμματισμού και την μεταποίηση αρχείων σε πραγματικό χρόνο.
- εξυπηρέτηση δεκάδων ή εκατοντάδων εκατομμυρίων συναλλαγών ταυτόχρονα από εκατομμύρια χρήστες.
- την άμεση και ταχεία πρόσβαση του διακομιστή σε υποδομή που εκτελεί λειτουργίες τεχνητής νοημοσύνης.

2.12 ETHEREUM και ETHEREUM VIRTUAL MACHINE

Το Ethereum είναι ένα αποκεντρωμένο blockchain δίκτυο που χρησιμοποιεί τεχνολογία κατανεμημένου καθολικού (distributed ledger technology - DLT) και υποστηρίζει την εκτέλεση έξυπνων συμβολαίων (smart contracts). Είναι ένα δημόσιο, ανοιχτού κώδικα blockchain που χρησιμοποιεί το μηχανισμό Proof of Stake (PoS) για την επίτευξη συναίνεσης. Οι κόμβοι (nodes)

του δικτύου, συμπεριλαμβανομένων των validators, συνεργάζονται για να διασφαλίσουν την ασφάλεια και την ακεραιότητα του δικτύου. Τα έξυπνα συμβόλαια αποθηκεύονται στο blockchain και εκτελούνται από το Ethereum Virtual Machine (EVM) [22]. Κάθε νέο μπλοκ δημιουργείται κάθε ≈ 12 δευτερόλεπτα και το δίκτυο μπορεί να υποστηρίξει **έναν περιορισμένο αριθμό συναλλαγών**. Το Ethereum 2.0 εισάγει το Beacon Chain, το οποίο διαχειρίζεται τους validators και τις επιβεβαιώσεις των μπλοκ, και το sharding, που επιτρέπει την κατανομή του φορτίου σε πολλαπλές αλυσίδες (shards) για καλύτερη κλιμάκωση. Η διαδικασία RANDAO χρησιμοποιείται για τη δημιουργία τυχαιότητας στην επιλογή των validators, εξασφαλίζοντας δίκαιη και απρόβλεπτη επιλογή. Το Ethereum επιτρέπει την ανάπτυξη αποκεντρωμένων εφαρμογών (dApps) και διαχειρίζεται αποκεντρωμένες χρηματοοικονομικές συναλλαγές (DeFi).

Τα smart contracts στο Ethereum είναι αυτοεκτελούμενα συμβόλαια γραμμένα σε γλώσσες όπως η Solidity, που μεταγλωττίζονται (compile) σε bytecode για εκτέλεση από την Ethereum Virtual Machine (EVM) [22]. Αυτά τα συμβόλαια περιλαμβάνονται στο payload των συναλλαγών (transactions) που περιλαμβάνονται στα μπλοκ του blockchain. Κατά την εκτέλεση, τα smart contracts χρησιμοποιούν τρεις κύριες μορφές αποθήκευσης: τη στοίβα (stack), τη μνήμη (memory), και την αποθήκευση (storage).

Η στοίβα χρησιμοποιείται για προσωρινές τιμές και υπολογισμούς κατά τη διάρκεια της εκτέλεσης των εντολών. Η μνήμη χρησιμοποιείται ως προσωρινός χώρος αποθήκευσης κατά την εκτέλεση των συναλλαγών, και καθαρίζεται μετά την ολοκλήρωση της συναλλαγής. Η αποθήκευση είναι ένας μόνιμος χώρος αποθήκευσης εντός του blockchain (on-chain storage), όπου αποθηκεύονται δεδομένα που χρειάζονται διαρκή αποθήκευση. Στο storage αποθηκεύονται σημαντικά δεδομένα όπως τα NFT (Non-Fungible Tokens), συνήθως μέσα από τα CID τους στο IPFS ή και ολόκληρες δομές δεδομένων με σημαντικά υψηλότερο κόστος αποθήκευσης.

Για την εκτέλεση και επαλήθευση των συναλλαγών, το Beacon Chain λειτουργεί ως ο κεντρικός συντονιστής των κόμβων και κάθε συναλλαγή τρέχει στο επιλεγμένο shard chain από επικυρωτές (validators) και full nodes του δικτύου και οι επικυρωτές (validators) κατανέμονται τυχαία στα shards μέσω της διαδικασίας RANDAO, εξασφαλίζοντας την ακεραιότητα και την ασφάλεια του δικτύου. Όταν ένας χρήστης στέλνει μια συναλλαγή, αυτή αναμεταδίδεται σε όλους τους validators του shard chain που θα εκτελέσουν το smart contract και θα επαληθεύσουν τα αποτελέσματα της εκτέλεσης μέσα από μηχανισμούς συναίνεσης. Κάθε validator του shard chain

εκτελεί το συμβόλαιο και ελέγχει την ορθότητα της εκτέλεσης, διασφαλίζοντας ότι η συναλλαγή ακολουθεί τους κανόνες του δικτύου.

Η επαλήθευση των συναλλαγών γίνεται επίσης από όλους τους validators του shard chain. Ο καθένας από αυτούς επαληθεύει ανεξάρτητα τις συναλλαγές και τα αποτελέσματα της εκτέλεσης των smart contracts. Αυτό διασφαλίζει ότι η συναλλαγή είναι έγκυρη και αποτρέπει προβλήματα όπως το double spending (διπλή δαπάνη), όπου ένας χρήστης προσπαθεί να ξοδέψει το ίδιο νόμισμα δύο φορές.

Οι full nodes (πλήρεις κόμβοι) στο Ethereum 2.0 έχουν καίριο ρόλο στη διατήρηση της ασφάλειας και της ακεραιότητας του δικτύου. Συγκεκριμένα, οι full nodes αποθηκεύουν ολόκληρο το ιστορικό και την τρέχουσα κατάσταση της Beacon Chain και επιλέγουν να αποθηκεύσουν και να επαληθεύουν δεδομένα από συγκεκριμένα shard chains. Τα full nodes διασφαλίζουν τη συνοχή και τον συγχρονισμό των δεδομένων σε ολόκληρο το δίκτυο, μεταδίδοντας συναλλαγές και μπλοκ μεταξύ των shards και της Beacon Chain. Επιπλέον, οι full nodes επαληθεύουν τις συναλλαγές και τα μπλοκ για να εξασφαλίσουν ότι τηρούν τους κανόνες του δικτύου, και συντονίζουν τις διασταυρούμενες συναλλαγές (cross-shard transactions).

Οι χρήστες πληρώνουν για την υπολογιστική ισχύ που καταναλώνει η εκτέλεση των smart contracts, μέσα από το σύστημα του gas. Το gas στο Ethereum είναι μια μονάδα μέτρησης της υπολογιστικής προσπάθειας που απαιτείται για την εκτέλεση συναλλαγών και έξυπνων συμβολαίων στο δίκτυο. Οι χρήστες ορίζουν ένα "gas limit", δηλαδή το μέγιστο ποσό gas που είναι διατεθειμένοι να πληρώσουν, και μια "gas price" σε wei, την μικρότερη μονάδα του Ether (ETH). Κάθε εντολή που εκτελείται στην Ethereum Virtual Machine (EVM) από τα smart contracts χρεώνεται σε gas, ανάλογα με την πολυπλοκότητα και τους υπολογιστικούς πόρους που απαιτεί.

Υπάρχει περίπτωση μια συναλλαγή να μην ολοκληρωθεί εάν εξαντληθεί το gas που έχει καθοριστεί για την εκτέλεση του smart contract. Σε αυτή την περίπτωση, η συναλλαγή αποτυγχάνει, αλλά το δίκτυο κρατά το gas που έχει ήδη χρησιμοποιηθεί. Επίσης, αν η συναλλαγή παραβιάζει κάποιον κανόνα (συνθήκη / solidity: require) του smart contract ή του δικτύου, θα απορριφθεί και δεν θα προστεθεί στο blockchain.

Το Ethereum ξεκίνησε το 2015 και έχει λάβει από τότε σημαντικές αλλαγές και βελτιώσεις στην αρχιτεκτονική του. Η αρχική εκδοχή, γνωστή ως Ethereum 1.0, χρησιμοποιούσε ένα σύστημα απόδειξης εργασίας (Proof of Work - PoW) για την επίτευξη συναίνεσης, το οποίο απαιτούσε μεγάλη ποσότητα υπολογιστικής ισχύος και ενέργειας. Το Ethereum 1.0 δεν χρησιμοποιούσε το Beacon Chain με τα Shard Chain, κάνοντας τα κόστη συναλλαγής και τις δυνατές πράξεις εκτέλεσης smart contract ανά μπλοκ περιορισμένες, αφού κάθε full node έπρεπε να επαληθεύει όλες τις συναλλαγές. Η ανάγκη για βελτιωμένη απόδοση και μεγαλύτερο πλήθος επεξεργασίας συναλλαγών ανά δευτερόλεπτο με χαμηλότερα κόστη gas οδήγησε στην ανάπτυξη του Ethereum 2.0, το οποίο μετατρέπει το σύστημα συναίνεσης σε απόδειξη μεριδίου (Proof of Stake - PoS).

Συνοψίζοντας όλες τις πληροφορίες μαζί, για το Ethereum (2.0), η Ethereum Virtual Machine (EVM) παίζει σημαντικό ρόλο εκτελώντας bytecode έξυπνων συμβολαίων, το οποίο συντάσσεται από γλώσσες προγραμματισμού υψηλού επιπέδου όπως η Solidity. Η EVM λειτουργεί με μια αρχιτεκτονική βασισμένη στη στοίβα. Η στοίβα λειτουργεί ως μια δομή δεδομένων τύπου LIFO (Last-In, First-Out), που κρατά προσωρινά τιμές κατά τη διάρκεια του υπολογισμού. Η μνήμη λειτουργεί ως προσωρινός χώρος αποθήκευσης για δεδομένα που χρησιμοποιούνται κατά την εκτέλεση του συμβολαίου, ο οποίος καθαρίζεται μετά την ολοκλήρωση της εκτέλεσης. Η αποθήκευση, αντιθέτως, είναι ένα μόνιμο σύστημα αποθήκευσης κλειδί-τιμή συγκεκριμένο για κάθε έξυπνο συμβόλαιο, επιτρέποντας στα δεδομένα να διατηρούνται μεταξύ των συναλλαγών. Το Gas χρησιμοποιείται ως μέτρο του υπολογιστικού κόστους που απαιτείται για την εκτέλεση λειτουργιών στην EVM. Κάθε λειτουργία έχει συγκεκριμένο κόστος σε gas, και οι συναλλαγές περιλαμβάνουν ένα όριο και τιμή gas. Η EVM χρησιμοποιεί ένα σύνολο εντολών γνωστών ως op-codes για την εκτέλεση διαφόρων εργασιών, συμπεριλαμβανομένων αριθμητικών πράξεων, ελέγχου ροής και αλληλεπίδρασης με την κατάσταση του blockchain. Υπάρχουν διάφοροι τύποι κόμβων που παίζουν ξεχωριστούς ρόλους για να διατηρήσουν το blockchain και να διασφαλίσουν την ασφάλειά του. Οι κόμβοι Beacon Chain είναι υπεύθυνοι για τη διατήρηση της Beacon Chain, η οποία είναι η βάση του μηχανισμού Proof of Stake (PoS) του Ethereum 2.0. Αυτοί οι κόμβοι αποθηκεύουν πληροφορίες για τους validators, επεξεργάζονται εγγραφές validators και διαχειρίζονται τα μηνύματα επικύρωσης από τους validators (attestations) και τις προτάσεις μπλοκ. Οι κόμβοι Shard Chain διαχειρίζονται και επικυρώνουν τα ξεχωριστά shard chains, τα οποία είναι υποσύνολα ολόκληρου του blockchain του Ethereum 2.0, για να αυξηθεί η κλιμακωσιμότητα. Οι validator nodes, ποντάρουν ETH

(δίνουν ένα ποσό για ασφάλεια – collateral) για να συμμετάσχουν στο δίκτυο, είναι υπεύθυνοι για την πρόταση και την επικύρωση μπλοκ. Επιλέγονται με βάση το μέγεθος του ποσού που έχει «πονταριστεί» σαν ασφάλεια σε ETH και μέσα από την συμμετοχή τους στη διαδικασία RANDAO. Υπάρχουν και οι light nodes, που αποθηκεύουν ελάχιστα δεδομένα όπως κεφαλίδες μπλοκ και βασίζονται σε full nodes για πρόσθετες πληροφορίες, συμμετέχουν στο δίκτυο με χαμηλότερη προσφορά πόρων, καθιστώντας τους κατάλληλους για συσκευές με περιορισμένους πόρους. [27]

Το RANDAO (Random Number Deterministic Autonomous Organization) είναι ένας σημαντικός μηχανισμός για την παραγωγή τυχαιότητας στο Ethereum 2.0, ο οποίος παίζει βασικό ρόλο στην ασφαλή και δίκαιη επιλογή των validators, καθώς και σε άλλες λειτουργίες του πρωτοκόλλου που απαιτούν τυχαιότητα[27]. Ο μηχανισμός RANDAO βασίζεται σε ένα σχήμα commit-reveal που εξασφαλίζει ότι οι παραγόμενοι τυχαίοι αριθμοί είναι αδύνατο να προβλεφθούν προτού αποκαλυφθούν επίσημα. Αναλυτικά η λειτουργία του RANDAO έχει 3 φάσεις εκτέλεσης [27]:

1. **Φάση Commit:** Στη πρώτη φάση, κάθε validator που συμμετέχει στη διαδικασία του RANDAO δημιουργεί μια τυχαία τιμή, την οποία κρατά μυστική. Στη συνέχεια, υπολογίζει το κρυπτογραφικό hash αυτής της τιμής και το υποβάλλει στο δίκτυο.
2. **Φάση Reveal:** Ακολουθεί η φάση της αποκάλυψης. Σε αυτή τη φάση, οι validators αποκαλύπτουν τις πραγματικές τους τυχαίες τιμές. Το δίκτυο ελέγχει ότι οι τιμές αντιστοιχούν στα αρχικά hash που υποβλήθηκαν.
3. **Παραγωγή Τυχαίου Αριθμού:** Το δίκτυο συνδυάζει τις τυχαίες τιμές όλων των validators για να παράγει έναν τελικό τυχαίο αριθμό.

Αυτή η τελική τυχαία τιμή επηρεάζει την επιλογή των validators και την διαδικασία συναίνεσης [27].

Η διαδικασία ανάπτυξης ενός έξυπνου συμβολαίου στο Ethereum γίνεται μέσω της αποστολής μιας συναλλαγής που περιέχει τον bytecode του συμβολαίου στο δίκτυο. Αυτός ο bytecode είναι η μεταγλωττισμένη μορφή του κώδικα που γράφεται σε γλώσσες υψηλού επιπέδου όπως η Solidity. Όταν ένα συμβόλαιο καλείται, η συναλλαγή εμπεριέχει τη διεύθυνση του συμβολαίου, τη λειτουργία που πρέπει να καλεστεί και τις απαραίτητες παραμέτρους. Η EVM εκτελεί το bytecode του συμβολαίου, χρησιμοποιώντας τη στοίβα, τη μνήμη και την μακροπρόθεσμη

αποθήκευση. Κάθε λειτουργία κατά την εκτέλεση καταναλώνει gas, και αν το όριο gas εξαντληθεί, η εκτέλεση σταματά και οι αλλαγές αναιρούνται. Η επιτυχής εκτέλεση έχει ως αποτέλεσμα την καταγραφή των αλλαγών κατάστασης στο shard chain. Στην περίπτωση που δύο χρήστες προσπαθούν να εκτελέσουν την ίδια λειτουργία ταυτόχρονα, στο ίδιο smart contract και προσπαθήσουν να κάνουν mint το ίδιο μοναδικό Token, η συναλλαγή με το υψηλότερο Gas Price θα εκτελεστεί πρώτα και η δεύτερη συναλλαγή, κοιτώντας την κατάσταση του συστήματος, θα ακυρώσει την δεύτερη «συγκρουόμενη» συναλλαγή, καθώς οι validators επιλέγουν συνήθως να επεξεργάζονται πρώτα εκείνες τις συναλλαγές που προσφέρουν το υψηλότερο gas price, επειδή αυτό μεγιστοποιεί τα κέρδη τους από τις αμοιβές συναλλαγών.

Η επαλήθευση των μπλοκ στο Ethereum 2.0 είναι μια πολύπλοκη και χρονοβόρα διαδικασία που αποσκοπεί στη διασφάλιση της ακεραιότητας και της ασφάλειας του blockchain. Οι validators επιλέγονται τυχαία για να προτείνουν νέα μπλοκ στην Beacon Chain και τις shard chains, με κάθε μπλοκ να περιέχει συναλλαγές και αναφορές σε προηγούμενα μπλοκ. Αυτή η διαδικασία περιλαμβάνει την εκπομπή των προτεινόμενων μπλοκ (που εμπεριέχουν οι συναλλαγές που έχουν επεξεργαστεί) στο δίκτυο, όπου άλλοι validators ελέγχουν τις υπογραφές, επικυρώνουν τις συναλλαγές και επιβεβαιώνουν τη συμμόρφωση με τους κανόνες του πρωτοκόλλου του δικτύου. Η οριστικοποίηση των μπλοκ στο επιτυγχάνεται μέσω του μηχανισμού Casper Friendly Finality Gadget (FFG) και θεωρούνται οριστικά αφού λάβουν επικυρώσεις σε διάφορες "εποχές". Κάθε "εποχή" είναι μια μονάδα χρόνου που περιλαμβάνει πολλαπλά slots, με κάθε slot να διαρκεί περίπου 12 δευτερόλεπτα. [27]

Στο Ethereum 2.0, οι συναλλαγές μέσα σε ένα μπλοκ και οι πληροφορίες του μπλοκ ελέγχονται πολλαπλές φορές σε διαδοχικές εποχές. Αρχικά κάθε συναλλαγή εντός ενός μπλοκ επαληθεύεται από τους validators τη στιγμή της πρότασης του μπλοκ. Έπειτα, αφού επαληθευθεί ένα μπλοκ, προστίθεται στο blockchain έχοντας όμως μόνο την προσωρινή αποδοχή του μπλοκ και όχι την τελική οριστικοποίηση του. Η οριστικοποίηση ενός μπλοκ συμβαίνει όταν πολλαπλές εποχές έχουν επικυρώσει την εγκυρότητά του μέσω του μηχανισμού Casper FFG. Δηλαδή πολλαπλές εποχές μετά την εισαγωγή του μπλοκ συνεχίζουν να επαληθεύουν τα μπλοκ που προηγήθηκαν μέχρι αυτό να θεωρηθεί οριστικά έγκυρο και αμετάκλητο.

2.13 ΜΟΝΑΔΙΚΑ ΤΕΚΜΗΡΙΑ ΜΕΣΑ ΣΕ ΕΦΑΡΜΟΓΕΣ ΤΩΝ SMART CONTRACT

Όπως έχει προαναφερθεί, τα smart contracts είναι αυτόματες συμφωνίες με τους όρους των συμβάσεων να κωδικοποιούνται σε blockchain, επιτρέποντας την αυτόματη εκτέλεση συμφωνιών χωρίς την ανάγκη για ενδιάμεσους. Η χρήση μοναδικών τεκμηρίων, όπως τα αποκεντρωμένα αναγνωριστικά (DIDs) και τα NFT μέσα στα πλαίσια ενός έμπιστου δικτύου Blockchain, ενισχύουν την ασφάλεια και τη διαφάνεια στις ψηφιακές συναλλαγές. Αυτή η τεχνολογική προσέγγιση επιτρέπει την εξασφάλιση της ακεραιότητας και της αμετάκλητης επαλήθευσης των συναλλαγών, καθιστώντας τα smart contracts ιδανικά για εφαρμογές που απαιτούν υψηλά επίπεδα αξιοπιστίας και ασφάλειας.

Ενώ υπάρχουν ιδιωτικά συστήματα που χρησιμοποιούν δικές τους βάσεις δεδομένων για να δηλώσουν και να αποθηκεύσουν μοναδικά δεδομένα που περιγράφουν μοναδικά όντα και την ιδιοκτησία τους από τρίτους, η έρευνα που πήρε μέρος εξετάζει μια πιο πολυδιάστατη μορφή αποκεντρωμένων μοναδικών τεκμηρίων, σε δημόσια συστήματα Blockchain, όπως τα νομίσματα ERC-1155 [18] στο δίκτυο Ethereum. Καθώς, η δήλωση μιας οντότητας σε ένα αποκεντροποιημένο σύστημα και η φύλαξη των δεδομένων ιδιοκτησίας σε δημόσιο καθολικό σύστημα φέρει μια σειρά από δυσκολίες και προ-απαιτήσεις, οι οποίες εν έτει 2024 καλύπτονται από τις ήδη υπάρχουσες και εφαρμοσμένες λύσεις της βιομηχανίας.

Για να εξυπηρετηθούν, λοιπόν, αυτές οι προηγμένες απαιτήσεις ψηφιοποιημένων οντοτήτων με μοναδικά τεκμήρια, το blockchain σύστημα εν χρήση πρέπει να υιοθετεί ένα αποκεντρωμένο ledger με ένα μεγάλο πλήθος ενεργών χρηστών και πολλών κόμβων, που επιτρέπει την πλήρη διαφάνεια και παρακολούθηση των συναλλαγών. Τα blockchain τύπου Ethereum, που υποστηρίζουν τα smart contracts μέσω της χρήσης της γλώσσας Solidity, είναι κατάλληλα λόγω της ευελιξίας που διαθέτουν, των συνεχών ελέγχων ασφαλείας και εξέτασης του κώδικα του συστήματος από τρίτους και λόγω των στοιχείων ασφαλείας του. Τέτοια συστήματα πρέπει να περιλαμβάνουν λειτουργίες όπως αλγόριθμοι κρυπτογράφησης για την εξασφάλιση της προστασίας δεδομένων και μηχανισμοί συναίνεσης όπως Proof of Work ή Proof of Stake για την εξασφάλιση της ακεραιότητας του δικτύου.

Ένα πρακτικό παράδειγμα της εφαρμογής μοναδικών τεκμηρίων σε blockchain είναι το σύστημα της πλατφόρμας Ethereum με τη χρήση των ERC-721 NFT tokens [18] για την ψηφιοποίηση και διαχείριση μοναδικών περιουσιακών στοιχείων. Αυτά τα tokens αντιπροσωπεύουν μοναδικά δεδομένα ή αγαθά στο blockchain και επιτρέπουν την ασφαλή ανταλλαγή και επαλήθευση της ιδιοκτησίας τους.

Αν και οι πιθανότητες είναι αμέτρητες στα πλαίσια του τι μπορεί να δημιουργηθεί προγραμματιστικά χρησιμοποιώντας το EVM και παρόμοια συστήματα, η πλατφόρμα OpenZeppelin παρέχει ένα πλαίσιο σχεδόν έτοιμων συμβολαίων smart contract, τα οποία αποτελούν το standard της βιομηχανίας όσο αφορά το EVM και το Ethereum Network. Αυτά τα πρότυπα συμβόλαια ERC (Ethereum Request for Comment) είναι ένα σύνολο προτάσεων που καθορίζουν πρότυπα για το πώς πρέπει να λειτουργούν διάφορες λειτουργίες σε συμβόλαια του Ethereum.

Χρησιμοποιώντας τις προαναφερόμενες τεχνολογίες, υπάρχουν πετυχημένα εφαρμοσμένα παραδείγματα όπως το παιχνίδι Axie Infinity, το οποίο χρησιμοποιεί ένα οικοσύστημα με βάση τα NFTs στο Ethereum blockchain για τη διαχείριση και την ανταλλαγή μοναδικών ψηφιακών πλασμάτων, τα Axies. Αυτό το παράδειγμα υποδεικνύει τον τρόπο με τον οποίο οι πλατφόρμες μπορούν να εκμεταλλευτούν την τεχνολογία blockchain για τη δημιουργία μιας οικονομίας βασισμένης σε έναν εικονικό ψηφιακό κόσμο, όπου η ύπαρξη μοναδικών τεκμηρίων μέσα από το δίκτυο Ethereum ενισχύει την ασφάλεια και την αξιοπιστία των συναλλαγών.

Κεφάλαιο 3 Μηχανές παιχνιδιών, Unreal Engine 5 και ασύγχρονες δυναμικές διαδικασίες

Εφόσον τα ψηφιακά τεκμήρια, σαν «σώμα» στον ψηφιακό χώρο, απαρτίζονται από μεταδεδομένα σε καθολικά, βάσεις δεδομένων, το IPFS, DIDs (Decentralized Identifiers), μπλοκ με δεδομένα σε Blockchain, απαιτούν ένα σύστημα παρουσίασης και διάδρασης από τον τελικό χρήστη, για να φέρουν τα ίδια λειτουργίες και χρηστικότητα προς αυτόν. Αυτό επιτυγχάνεται μέσα από το γραφικό περιβάλλον των διάφορων εφαρμογών, το οποίο μπορεί να είναι μια ιστοσελίδα, το User Interface (διεπαφή χρήστη) σε ένα executable αρχείο (εκτελέσιμο αρχείο), ή ο δισδιάστατος ή τρισδιάστατος εικονικός χώρος ενός βιντεοπαιχνιδιού. Στα πλαίσια της έρευνας χρησιμοποιήθηκαν τεχνολογίες αιχμής ή οποίες θα αναλυθούν και θα παρουσιαστούν στο ακόλουθο κείμενο.

3.1 Μοντέρνες μηχανές βιντεοπαιχνιδιών

Μια μηχανή δημιουργίας παιχνιδιών (game engine) είναι ένα ολοκληρωμένο λογισμικό πλαίσιο (framework), σχεδιασμένο για να διευκολύνει την ανάπτυξη και την λειτουργία διαδραστικών ψηφιακών εμπειριών, κυρίως βιντεοπαιχνίδια. Λειτουργεί ως στρώμα αφαίρεσης (abstraction layer) πάνω από το υλισμικό (hardware) και τις χαμηλού επιπέδου κλήσεις συστήματος (low-level system calls), προσφέροντας ένα σύνολο από υψηλού επιπέδου προγραμματιστικές διεπαφές (APIs) και εργαλεία για τις σημαντικές και βασικές εργασίες ανάπτυξης παιχνιδιών. Διαχειρίζεται, επίσης, τις υποκείμενες πολυπλοκότητες που εμπλέκονται με το rendering των γραφικών, την προσομοίωση φυσικής, τη διαχείριση εισόδων από χειριστήρια και την εκτέλεση υπολογισμών σε πραγματικό χρόνο, μεταξύ πολλών άλλων λειτουργιών. Αυτό επιτρέπει στους προγραμματιστές παιχνιδιών να επικεντρώνονται στη κατασκευή της λογικής του παιχνιδιού, τη δημιουργία περιεχομένου και άλλες υψηλού επιπέδου πτυχές της ανάπτυξης παιχνιδιών, αντί να χρειάζεται να εμβαθύνουν στις πολυπλοκότητες του προγραμματισμού χαμηλού επιπέδου και της αλληλεπίδρασης με το υλισμικό. Με άλλα λόγια, η μηχανή παιχνιδιών σαν στρώμα αφαίρεσης κρύβει τις πολύπλοκες και χαμηλού επιπέδου λειτουργίες και παρέχει στον προγραμματιστή μια πιο εύκολη και ευανάγνωστη διεπαφή για να δουλέψει [34].

Το μεγάλο πλήθος της βιομηχανίας των βιντεοπαιχνιδιών, σχεδόν αποκλειστικά χρησιμοποιεί την C++ για την κατασκευή των μηχανών βιντεοπαιχνιδιών και του βασικού κώδικα των

περισσότερων βιντεοπαιχνιδιών. Στην βιομηχανία παραγωγής AAA βιντεοπαιχνιδιών, δηλαδή μεγάλων project με προϋπολογισμούς πάνω από τα 100.000.000\$ και εκατοντάδες προγραμματιστές που δουλεύουν 4 με 5 χρόνια πάνω στο ίδιο παιχνίδι με τελευταίας γενιάς γραφικά υψηλών προδιαγραφών, η C++ χρησιμοποιείται για την κατασκευή του κεντρικού πυρήνα του παιχνιδιού, αποκλειστικά. Επίσης οι περισσότερες σύγχρονες παιχνιδομηχανές είναι γραμμένες σε C++, όπως η Unity (με C# για το scripting), Unreal Engine, Godot, CryEngine, GameMaker Studio, RPG Maker, Lumberyard της Amazon, Cocos2d-x, Anvil Engine από την Ubisoft (Assassin's Creed), Frostbite Engine από την EA DICE (Battlefield), IW Engine από την Infinity Ward (Call of Duty), RAGE (Rockstar Advanced Game Engine) από την Rockstar (Grand Theft Auto), Decima από την Guerrilla Games/Sony (Horizon Zero Dawn, God of War), Fox Engine από την Konami (Metal Gear Solid, Pro Evolution Soccer) καθώς και η μηχανή Source της Valve (Counterstrike, Dota 2, Half Life) όπως και η μηχανή της Riot Games για το "League of Legends" και η μηχανή της Blizzard για το World of Warcraft είναι γραμμένες σε C++. Επιπρόσθετα, περίπου το 48% των ανακοινωθέντων τίτλων κονσολών επόμενης γενιάς (Microsoft Xbox Series X |S, Sony PlayStation 5) κατασκευάζονται πάνω στην μηχανή Unreal Engine 5, εν έτη 2022 [29].

Μια από τις πιο τεχνολογικά προηγμένες μηχανές κατασκευής βιντεοπαιχνιδιών είναι η Unreal Engine 5, η οποία αποτελεί την τελευταία έκδοση της ευρέως αναγνωρισμένης μηχανής βιντεοπαιχνιδιών που αναπτύχθηκε από την Epic Games, γνωστή για τα προηγμένα γραφικά και την ευελιξία της. Κυκλοφόρησε στο κοινό πρόωρα το 2021 και πλήρως το 2022, και αποτελεί τεχνολογία αιχμής στη δημιουργία ψηφιακού περιεχομένου, προσφέροντας στους προγραμματιστές τα εργαλεία για τη δημιουργία γραφικά εξελιγμένων ψηφιακών εμπειριών με πρωτοφανή ρεαλισμό. Κεντρικά χαρακτηριστικά όπως τα Nanite και Lumen λειτουργούν αρμονικά για την αναπαράσταση γεωμετρίας με πολύ υψηλό αριθμό τριγώνων και δυναμικού παγκόσμιου φωτισμού (Global Illumination) σε πραγματικό χρόνο, τα οποία διευκολύνουν στη δημιουργία φωτορεαλιστικών σκηνών. Η μηχανή απευθύνεται σε προγραμματιστές παιχνιδιών, κινηματογραφιστές και πάσης φύσεως δημιουργούς οπτικού περιεχομένου, με φιλικές προς τον χρήστη διεπαφές, και δυνατή υποστήριξη για συνεργατικά έργα (group projects).

Σε σύγκριση με τις υπόλοιπες διαθέσιμες, στο κοινό, μηχανές βιντεοπαιχνιδιών η Unreal Engine 5 ξεχωρίζει από τον ανταγωνισμό λόγω των καινοτόμων χαρακτηριστικών τεχνολογίας αιχμής

που εντοιχίζει, προωθώντας τα όρια του δυνατού στη δημιουργία ψηφιακού περιεχομένου. Οι προηγμένες δυνατότητες path tracing της Unreal Engine 5 επιτρέπουν την δημιουργία ρεαλιστικού φωτισμού, δημιουργώντας σκηνές με κινηματογραφική ποιότητα. Επιπλέον, η ενσωμάτωση της τεχνητής νοημοσύνης βελτιώνει τη διαδικασία ανάπτυξης, αυτοματοποιώντας σύνθετες εργασίες και επιτρέποντας πιο ευφυή συμπεριφορά στους μη-παίκτες χαρακτήρες NPC. Οι μη-παίκτες χαρακτήρες MetaHumans είναι ένα επαναστατικό σύστημα δημιουργίας χαρακτήρων, το οποίο παρέχει στους προγραμματιστές εργαλεία για την παραγωγή υπερρεαλιστικών ψηφιακών ανθρώπων με ευκολία. Η μηχανή Chaos, προσφέρει ανεπτυγμένες δυνατότητες εξομίωσης φυσικής. Όλα αυτά, στοιχεία που λείπουν από τον ανταγωνισμό, όπως την Unity και την GoDot.

3.2 Βασικά Στοιχεία / Λειτουργίες της Unreal Engine 5

3.2.1 Προγραμματισμός στην Unreal Engine 5

Η Unreal Engine 5 υποστηρίζει τον προγραμματισμό του κώδικα και των λειτουργιών μιας εικονικής εμπειρίας μέσα από ένα ευρύ πλήθος διαφορετικών εργαλείων, τα οποία διαχωρίζονται σε 3 βασικές μεθόδους προγραμματισμού. Αυτές είναι ο προγραμματισμός σε C++, ο προγραμματισμός σε python και η χρήση του συστήματος Blueprint, το οποίο Blueprint βρίσκεται μέσα σε γραφικό περιβάλλον[31].

Η C++ είναι η κύρια γλώσσα προγραμματισμού για την Unreal Engine 5, δίνοντας στους προγραμματιστές πλήρη έλεγχο των δυνατοτήτων της μηχανής. Με την C++, οι προγραμματιστές μπορούν να γράψουν άκρως βελτιστοποιημένο (refined) κώδικα που αλληλεπιδρά άμεσα με τον πυρήνα της μηχανής. Αυτή η προσέγγιση είναι ζωτικής σημασίας για τη δημιουργία βασικών λειτουργιών (όπου η απόδοση και ο χρόνος εκτέλεσης είναι σημαντικά στοιχεία), όπως σύνθετους μηχανισμούς παιχνιδιού, προσαρμοσμένους υπολογισμούς φυσικής ή περίπλοκες συμπεριφορές AI. Το C++ API της Unreal Engine προσφέρει πρόσβαση σε ένα μεγάλο πλήθος προκατασκευασμένων λειτουργιών της μηχανής, από χαμηλού επιπέδου λειτουργίες απόδοσης μέχρι και το Reflection API που χρησιμοποιεί η Unreal, επιτρέποντας στους προγραμματιστές να επεκτείνουν και να προσαρμόσουν σχεδόν κάθε πτυχή της μηχανής. Επιπρόσθετα, ο κώδικας της Unreal Engine 5 είναι ανοιχτός (όχι, όμως, ελεύθερος) και μπορεί να τροποποιηθεί γύρω από τις ανάγκες της κάθε εφαρμογής [42].

Τα **Blueprints** παρέχουν ένα φιλικό, προς τον προγραμματιστή, σύστημα οπτικής σχεδίασης λειτουργικών ρουτινών και αποτελούν αναπόσπαστο μέρος της Unreal Engine 5. Αυτό το σύστημα επιτρέπει στους σχεδιαστές και στους προγραμματιστές να δημιουργούν πολύπλοκες αλληλεπιδράσεις και στοιχεία παιχνιδιού χωρίς την σύνθεση κώδικα σε C++. Τα Blueprints μεταγλωττίζονται σε κώδικα με βάση τα στάνταρ και τις μεθοδολογίες ανάπτυξης κώδικα της μηχανής, εξασφαλίζοντας ότι εκτελούνται αποδοτικά στο παιχνίδι. Είναι ιδιαίτερα χρήσιμα για την γρήγορη δημιουργία πρωτοτύπων, την προσαρμογή μεταβλητών παιχνιδιού σε πραγματικό χρόνο και τη διαχείριση συστημάτων βάσει συμβάντων (event driven systems / architecture), όπως αλληλεπιδράσεις UI ή χαρακτήρων και περιβάλλοντος.

Η **Python** χρησιμοποιείται κυρίως για εργασίες αυτοματοποίησης, δικτύωσης και μηχανικής μάθησης εντός της Unreal Engine 5[35]. Κώδικας σε Python μπορεί να αλληλεπιδράσει με τη μηχανή μέσω ενός εκτεταμένου API, αυτοματοποιώντας επαναλαμβανόμενες εργασίες προσφέροντας πρόσβαση στις πολυπληθείς βιβλιοθήκες της Python. Αυτή η δυνατότητα είναι ιδιαίτερα πολύτιμη σε περιβάλλοντα παραγωγής μεγάλης κλίμακας, όπου η διαχείριση μεγάλου όγκου περιεχομένου μπορεί να είναι εξαιρετικά χρονοβόρα ή κάτω από τα πλαίσια της κατασκευής πρωτότυπων συστημάτων που χρησιμοποιούν την τεχνητή Νοημοσύνη.

Τα προαναφερόμενα περιβάλλοντα προγραμματισμού υποστηρίζονται μεμονωμένα από API της Unreal Engine. Τα API επιτρέπουν την αλληλεπίδραση με τα υποσυστήματα της μηχανής—όπως γραφικά, ήχο, φυσική και δικτύωση—και επιτρέπουν την προσαρμογή της συμπεριφοράς της μηχανής, την ενσωμάτωση εξωτερικών τεχνολογιών και τη βελτιστοποίηση του περιεχομένου του παιχνιδιού. Μέσω αυτών των API, η Unreal Engine 5 διατηρεί μια ευέλικτη αρχιτεκτονική όπου μπορούν να προστεθούν νέες λειτουργίες ή να τροποποιηθούν οι υπάρχουσες ώστε να ταιριάζουν σε συγκεκριμένες ανάγκες έργων χωρίς να επηρεαστεί η απόδοση ή η επεκτασιμότητα του project. Είτε γράφοντας σύνθετη λογική παιχνιδιού σε C++, είτε κατασκευάζοντας στοιχεία παιχνιδιού μέσα από τα Blueprints, είτε αυτοματοποιώντας διαδικασίες μέσα από την Python, οι προγραμματιστές έχουν στη διάθεσή τους ένα εκτενές σύνολο εργαλείων.

3.2.2 Τεχνικά Δεδομένα για την Unreal Engine 5

- **Γλώσσες Προγραμματισμού:** C++ (κύρια γλώσσα), Blueprint (οπτική σχεδίαση κώδικα), Python (αυτοματοποίηση και προεκτάσεις).
- **Υποστηριζόμενες Πλατφόρμες:** Windows, macOS, Linux, iOS, Android, PlayStation, Xbox, Nintendo Switch, HTML5 (WebAssembly).
- **Γραφικά APIs:** DirectX 11/12, Vulkan, Metal, OpenGL.
- **Υποστηριζόμενα μορφότυπα αρχείων:** FBX, OBJ, PNG, JPEG, MP4, WAV, και άλλα.

Ακολουθεί μια συνοπτική παρουσίαση της δομής των βασικών συστημάτων της Unreal Engine 5. Επειδή, όμως, η ίδια η μηχανή αποτελεί πλατφόρμα λογισμικού επιπέδου μεγάλων επιχειρήσεων (large scale enterprise-grade software platform) με δεκάδες εκατομμύρια γραμμές κώδικα και εκατοντάδες χιλιάδες αρχεία, σε σημείο όπου η ίδια η μηχανή θεωρείται πως έχει βάση-κώδικα (codebase) σε επίπεδο/μέγεθος λειτουργικού συστήματος (Operating System), η ανάλυση μπορεί να παρουσιάσει μόνο ένα πολύ μικρό μέρος των δυνατοτήτων της Unreal Engine 5.

Διεπαφές Επεξεργασίας (Editor Interface)

- **Unreal Editor:** Η κύρια διεπαφή (interface) για τη δημιουργία και διαχείριση παιχνιδιών.
- **Content Browser:** Οργανώνει και επιτρέπει την διαχείριση αρχείων στοιχείων παιχνιδιού.
- **Viewport:** Επιτρέπει το rendering σε πραγματικό χρόνο και την αλληλεπίδραση με το παιχνίδι.
- **Blueprint Editor:** Περιβάλλον οπτικού προγραμματισμού.

Μηχανή Rendering (Rendering Engine)

- **Nanite:** Τεχνολογία Virtualized γεωμετρίας που επιτρέπει το rendering μοντέλων υψηλής λεπτομέρειας χωρίς σημαντικές απώλειες απόδοσης.
- **Lumen:** Σύστημα παγκόσμιου φωτισμού και ανακλάσεων για ρεαλιστικό φωτισμό
- **Ray Tracing:** Προηγμένη τεχνική rendering για τη δημιουργία ρεαλιστικού φωτισμού, σκιών και αντανακλάσεων.

Μηχανή Φυσικής (Physics Engine)

- **Chaos Physics and Destruction:** Προηγμένο σύστημα εξομοίωσης φυσικής για ρεαλιστικές προσομοιώσεις καταστροφών, ragdoll εφέ και άλλα.

Σύστημα Animation (Animation System)

- **Control Rig:** Εργαλείο ρύθμισης για την κίνηση χαρακτήρων.
- **Sequencer:** Εργαλείο για τη δημιουργία και διαχείριση κινηματογραφικών ακολουθιών.

Μηχανή Ήχου (Audio Engine)

- **MetaSounds:** Σύστημα παραγωγής διαδικαστικού ήχου (Procedural audio generation system).
- **Audio Mixer:** Επιτρέπει την διαχείριση ήχου και εφέ.

Εργαλεία Ανάπτυξης Παιγνιδιών (Development Tools)

- **Visual Studio Integration:** Για την δημιουργία και το debugging κώδικα C++.
- **Unreal Build System (UBT):** Διαχειρίζεται τη διαδικασία κατασκευής στα projects.
- **Automation and Testing Frameworks:** Εργαλεία για τον αυτοματισμό των δοκιμών και των build του project.

Διαχείριση Αντικειμένων (Asset Management)

- **Quixel Megascans:** Μέσα από την επέκταση (plugin) quixel Bridge, επιτρέπει την χρήση τρισδιάστατων αντικειμένων υψηλής ευκρίνειας από τρισδιάστατη σάρωση (3D scanning).

3.2.3 Βασικά API στην Unreal Engine 5 και Βιβλιογραφία

Η Unreal Engine παρέχει λεπτομερή και εκτενή εγχειρίδια οδηγιών (Documentation) για τους προγραμματιστές στην ιστοσελίδα της Epic Games [30]. Αυτή η πηγή περιλαμβάνει μια ευρεία γκάμα πληροφοριών για την βοήθεια των προγραμματιστών σε κάθε στάδιο ανάπτυξης του έργου. Τα εγχειρίδια οδηγιών καλύπτουν νέες δυνατότητες και βελτιώσεις που εισήχθησαν στην κάθε έκδοση της μηχανής, με οδηγίες ανά έκδοση.

Μέσα στο εγχειρίδιο περιλαμβάνονται και χρήσιμες πληροφορίες πάνω στα API που προσφέρει για τον προγραμματισμό των παιχνιδιών. Δίνονται οι σύνδεσμοι για την έκδοση 5.3 που τελικά χρησιμοποιήθηκε μετά από την αναβάθμιση του παιχνιδιού από την 5.1:

Διεπαφές Προγραμματισμού Εφαρμογών Παιχνιδιού (Gameplay Framework APIs):

- **Blueprint API:** https://dev.epicgames.com/documentation/en-us/unreal-engine/BlueprintAPI?application_version=5.3
- **C++ API:** https://dev.epicgames.com/documentation/en-us/unreal-engine/API?application_version=5.3
- **Python API:** https://dev.epicgames.com/documentation/en-us/unreal-engine/PythonAPI?application_version=5.3

3.3 Δομή παιχνιδιών

3.3.1 Προπαραγωγή (Pre-production)

Τεχνικός Σχεδιασμός

- **Επιλογή Χαρακτηριστικών της Unreal Engine:** Αρχικά διαλέγονται τα χαρακτηριστικά του Unreal Engine που θα χρησιμοποιηθούν (π.χ. Blueprints Visual Scripting, MetaHumans, μηχανή φυσικής).
- **Διαμόρφωση Asset Pipeline:** Καθιέρωση ροών εργασίας (workflows) για τη δημιουργία και την εισαγωγή 3D μοντέλων, textures, animation και ήχων.
- **Εξωτερικά Εργαλεία:** Επιλογή εξωτερικών εργαλείων για τις επιπρόσθετες δυνατότητες που σχεδιάζεται να έχει το κάθε παιχνίδι, όπως σύνδεση με κάποιο σύστημα Blockchain ή λήψη δεδομένων από συστήματα IOT (internet of things).

3.3.2 Παραγωγή (Production)

Βασικά Στοιχεία Παιχνιδιού μέσα στη μηχανή

- **Game Instance:** Διαχειρίζεται δεδομένα και ρουτίνες που απαιτούν να παραμένουν σταθερά (και υπαρκτά) καθ' όλη τη διάρκεια που τρέχει το παιχνίδι, ανεξάρτητα της

κατάστασης του παιχνιδιού. Χρησιμοποιείται για την προσωρινή αποθήκευση μεταβλητών που παραμένουν ακόμα και μετά την εναλλαγή πίστας.

- **Game Mode** Καθορίζει τους κανόνες του παιχνιδιού—τι συμβαίνει όταν ένας παίκτης ξεκινά ή τελειώνει το παιχνίδι, κερδίζει πόντους ή πεθαίνει. Συνήθως ελέγχει τη λογική του παιχνιδιού.
- **Game State** Αποθηκεύει πληροφορίες για την κατάσταση του παιχνιδιού, όπως σκορ, χρόνο και άλλα στατιστικά στοιχεία του παιχνιδιού που πρέπει να παρακολουθούνται.
- **PlayerState**: Παρακολουθεί τις σημαντικές λειτουργικές μεταβλητές που χαρακτηρίζουν τον χαρακτήρα του κεντρικού παίκτη, όπως την ζωή του.
- **Player Controller**: Διαχειρίζεται την συσκευή εισόδου του χρήστη HID (Human Interface Device).
- **Actor**: Η βασική κλάση για όλα τα αντικείμενα στον κόσμο του Unreal Engine που μπορούν να αλληλεπιδρούν ή να δέχονται αλληλεπίδραση.
- **Pawn**: Actors που μπορούν να χρησιμοποιηθούν σε διαδικασίες και ρουτίνες από τον παίκτη και το AI.
- **Character**: Ένα εξειδικευμένο pawn που περιλαμβάνει δυνατότητες, όπως το περπάτημα, σε οντότητες τύπου χαρακτήρα παίκτη.
- **Characters και Pawns**: «Αντικείμενα» που μπορούν να ελεγχθούν από τον παίκτη, την τεχνητή νοημοσύνη ή την λογική του παιχνιδιού προγραμματιστικά. Αλληλεπιδρούν με τον κόσμο του παιχνιδιού και μεταξύ τους.
- **Levels/Maps**: Οι φυσικοί χώροι όπου λαμβάνει μέρος το παιχνίδι, συμπεριλαμβανομένων και όλων των περιβαλλοντικών αντικειμένων / στοιχείων, σημείων εισόδου του Character στην πίστα καθώς και τα διαδραστικά στοιχεία του κόσμου.
- **Level Blueprint**: Ένα είδος Blueprint που σχετίζεται με συγκεκριμένα επίπεδα ή χάρτες και επιτρέπει την εφαρμογή scripts και λογικής που αφορούν ολόκληρο το επίπεδο ή τον χώρο παιχνιδιού.
- **HUD (Heads-Up Display)**: Διαχειρίζεται τα in-game overlays που δημιουργούν, στο σύνολο τους, την Διεπαφή Χρήστη (User Interface).
- **AIController**: Διαχειρίζεται τη συμπεριφορά του AI για τον έλεγχο των pawns.
- **Behavior Tree**: Ένα σύστημα που χρησιμοποιείται για το σχεδιασμό του AI, κυρίως για NPCs.

- **AnimBlueprint:** Διαχειρίζεται πολύπλοκες κινήσεις βάσει λογικής και αλγορίθμων.
- **Skeletal Mesh:** 3D μοντέλα που υποστηρίζουν πολύπλοκες κινήσεις και δομές οστών.
- **Static Mesh:** 3D μοντέλα χωρίς δομές οστών, που χρησιμοποιούνται για στατικά αντικείμενα.
- **Material:** Ορίζει τα στοιχεία εμφάνισης και την τελική εμφάνιση των εικόνων texture των μοντέλων, συμπεριλαμβανομένων χρωμάτων, αντανακλασιμότητας και ιδιοτήτων φωτισμού.
- **ParticleSystem:** Χρησιμοποιείται για τη δημιουργία εφέ σωματιδίων.
- **SoundCue:** Σύστημα ήχου που μπορεί να παίζει αρχεία ήχου εντός των Blueprint
- **PhysicsVolume:** Ελέγχει τις φυσικές αλληλεπιδράσεις και συμπεριφορές σε μια συγκεκριμένη περιοχή του κόσμου του παιχνιδιού.
- **Collision Mesh:** Αόρατα πλέγματα που χρησιμοποιούνται για τον εντοπισμό συγκρούσεων.
- **NavMesh:** Παρέχει λειτουργίες εύρεσης διαδρομής (pathfinding) για το AI.
- **CameraActor:** Αντιπροσωπεί μια κάμερα μέσα στον κόσμο του παιχνιδιού.
- **SceneComponent:** Χρησιμοποιείται για στοιχεία που φέρουν μια φυσική παρουσία στον κόσμο του παιχνιδιού. Είναι, ουσιαστικά, ένα component που προσφέρει φυσικότητα αντί για λειτουργικότητα – όπως τα actor component. Όμως, το scene component λειτουργεί σαν τον βασικό δομικό λίθο των φυσικών οντοτήτων που χρησιμοποιούν actor components, όπως και των πολύπλοκων λειτουργικών οντοτήτων εντός του κόσμου των παιχνιδιών.
- **WidgetBlueprint:** Χρησιμοποιείται για τη δημιουργία των στοιχείων της διεπαφής χρήστη (user interface).
- **LightActor:** Αντιπροσωπεί τύπους φωτεινών πηγών μέσα στο παιχνίδι.
- **Volume:** Εξειδικευμένα actors που χρησιμοποιούνται για τον καθορισμό χώρων (μέσα από σφαίρες ή επίπεδα /πλάνες) στον κόσμο του παιχνιδιού, όπου μέσα σε αυτούς τους καθορισμένους όγκους ισχύουν συγκεκριμένοι κανόνες ή εφέ.
- **Decal:** Προβάλλει textures πάνω σε άλλες επιφάνειες ή αντικείμενα μέσα στο περιβάλλον του παιχνιδιού.
- **BlueprintFunctionLibrary:** Μια συλλογή λειτουργιών / συναρτήσεων που μπορούν να επαναχρησιμοποιηθούν σε διαφορετικά Blueprints.

- **WorldSettings:** Διαχειρίζεται τις ρυθμίσεις κόσμου του παιχνιδιού, όπως η βαρύτητα ή ο περιβαλλοντικός φωτισμός.
- **Pak Files:** Χρησιμοποιούνται για το πακετάρισμα περιεχομένου όπως textures, μοντέλα, ήχους και άλλα στοιχεία. Τα αρχεία .pak είναι μέρος του συστήματος πακεταρίσματος στοιχείων της μηχανής και προσφέρουν την δυνατότητα της διανομής αρχείων επέκτασης παιχνιδιού (DLC) – ακόμα και ενώ το παιχνίδι τρέχει (run time) και έναν τρόπο πακεταρίσματος περιεχομένου στοιχείων του παιχνιδιού. Επίσης, βρίσκουν χρήση στη δημιουργία mods (μετατροπών), καθώς τα αρχεία .pak μπορούν να χρησιμοποιηθούν για την προσθήκη ή αντικατάσταση περιεχομένου του παιχνιδιού χωρίς την αντικατάσταση των αρχείων του κεντρικού πυρήνα του παιχνιδιού, καθιστώντας εύκολη τη διανομή των mods από τους δημιουργούς και την «απορρόφηση» τους από τα παιχνίδια.
- **Data Tables:** Χρησιμοποιούνται για την αποθήκευση δεδομένων σε μορφή δομημένου, δισδιάστατου πίνακα, τα οποία δεδομένα είναι εύκολα ανακλητά και διαχειρίσιμα από οπουδήποτε εντός του προγράμματος. Η δομή τους θυμίζει ένα φύλλο Excel και δεν αποτελούν τα ίδια δυναμικά στοιχεία, αλλά μπορούν να γίνουν μέσω τροποποιήσεων του κώδικα ή με τη χρήση προεκτάσεων. Κάθε σειρά έχει μια δομή την η οποία προκαθορίζεται από κάποιο Struct και πολλές φορές η κάθε σειρά σε έναν πίνακα DT αποτελεί ένα όν και τα χαρακτηριστικά του.
- **Enumerations (Enums):** Χρησιμοποιούνται εκτενώς σε λίστες αντικειμένων ή επιλογών μαζί με εντολές ‘switch on Enum’ για την αλλαγή της κατάστασης των στοιχείων. Στα Enums, κάθε string αντιστοιχεί σε μια προκαθορισμένη τιμή. Τα Enums κάνουν τον κώδικα πιο ευανάγνωστο και διαχειρίσιμο, βοηθώντας στην αποφυγή λαθών που θα μπορούσαν να προκύψουν από τη χρήση απλών ακέραιων τιμών. Στη μηχανή Unreal Engine, τα Enums χρησιμοποιούνται συχνά για τη διαχείριση καταστάσεων, επιλογών και κατηγοριών, προσφέροντας έναν καθαρό και σαφή τρόπο για τη διαχείριση μεγάλων λιστών αντικειμένων και ψηφιακών όντων.

Αυτά τα στοιχεία είναι κεντρικά δομικά στοιχεία για τη δημιουργία διαδραστικών ψηφιακών εμπειριών με την χρήση της Unreal Engine. Κάθε στοιχείο έχει συγκεκριμένους ρόλους και μπορεί να επεκταθεί ή να προσαρμοστεί ανάλογα με τις ανάγκες της εφαρμογής.

3.3.3 Βήματα προσέγγισης κατασκευής παιχνιδιών

Ακολουθεί μια υψηλού επιπέδου ανάλυση των βημάτων προσέγγισης κατασκευής παιχνιδιών στην Unreal Engine 5 μέσα από τα διάφορα συστήματα της:

1. Φάση Σχεδιασμού

- ❖ **GameInstance:** Ορισμός των βασικών στοιχείων που θα παραμένουν στην μνήμη ανεξαρτήτως από τις πίστες ή την κατάσταση του παιχνιδιού. Όλες οι παγκόσμιες μεταβλητές (Global Variables) πρέπει να φυλάσσονται εδώ.
- ❖ **GameMode:** Καθορισμός βασικών κανόνων του παιχνιδιού, όπως το πώς κερδίζουν ή χάνουν οι παίκτες, πώς αναγεννιούνται και άλλοι θεμελιώδεις μηχανισμοί παιχνιδιού.
- ❖ **GameState:** Χρησιμοποιείται για την παρακολούθηση των συνολικών δεδομένων του παιχνιδιού.
- ❖ **PlayerState:** Προσδιορισμός στοιχείων και μεταβλητών του παίκτη.

2. Σχεδιασμός Επιπέδων

- ❖ **WorldSettings:** Ρύθμιση των παραμέτρων του παιχνιδιού, συμπεριλαμβανομένων των παραμέτρων φυσικής και των περιβαλλοντικών παραμέτρων.
- ❖ **NavMesh:** Ορισμός πλεγμάτων πλοήγησης (navigation meshes) για την εύρεση διαδρομής από το σύστημα AI (AI pathfinding) μέσα στα επίπεδα του παιχνιδιού.
- ❖ **Volume:** Επιπρόσθεση όγκων για ειδικά εφέ ή μηχανισμούς παιχνιδιού, όπως περιοχές με διαφορετική μουσική ή ομίχλη.
- ❖ **Static Mesh:** Τοποθέτηση στατικών αντικειμένων στον κόσμο του παιχνιδιού, που δεν απαιτούν animations.
- ❖ **Dynamic Objects:** Τοποθέτηση δυναμικών αντικειμένων στον κόσμο του παιχνιδιού, με ιδιότητες φυσικής, λογική παιχνιδιού και animations.
- ❖ **Skeletal Mesh:** Τοποθέτηση χαρακτήρων και άλλων κινούμενων αντικειμένων.
- ❖ **Material:** Σχεδιασμός υλικών για διάφορες επιφάνειες και αντικείμενα / actors.
- ❖ **Collision Mesh:** Ρύθμιση των πλεγμάτων σύγκρουσης για την εφαρμογή της φυσικής στις συγκρούσεις των οντοτήτων.

3. Υλοποίηση Παιχνιδιού

- ❖ **Character:** Ανάπτυξη χαρακτήρων με δυνατότητες κίνησης και animations.
- ❖ **Pawn:** Δημιουργία άλλων ελέγξιμων αντικειμένων, όπως οχήματα.
- ❖ **AIController:** Σύνδεση των AI controllers σε μη-παίκτες χαρακτήρες (NPCs).
- ❖ **PlayerController:** Εφαρμογή σχεδιαγράμματος ελέγχου (control schemes) για την αλληλεπίδραση των χρηστών με τον κόσμο του παιχνιδιού.
- ❖ **AnimBlueprint:** Σχεδίαση σύνθετων animations για χαρακτήρες και αντικείμενα.

4. Αλληλεπιδράσεις και Διεπαφές

- ❖ **HUD:** Ανάπτυξη συστήματος απεικόνισης των ενδείξεων των λογικών λειτουργιών του παιχνιδιού πάνω στην οθόνη (Heads-Up Display) για να εμφανίζονται χρήσιμα στοιχεία του παίκτη, μηνύματα παιχνιδιού και άλλα διαδραστικά στοιχεία πάνω από το παιχνίδι.
- ❖ **WidgetBlueprint:** Δημιουργία διεπαφών χρήστη όπως: μενού, οθόνη λήξης παιχνιδιού (Game Over) και οθόνες προβολής inventory (αποθέματος).
- ❖ **CameraActor:** Ρύθμιση συστημάτων κάμερας για σκηνές ή κινηματογραφικά εφέ.
- ❖ **SceneComponent:** Χρήση των SceneComponent για τη δημιουργία προσαρμοσμένων σύνθετων αντικειμένων με δική τους συμπεριφορά.
- ❖ **Level Blueprint:** Δημιουργία λογικής και συμβάντων ειδικά για το κάθε περιβάλλον.

5. Οπτικοακουστικά Εφέ

- ❖ **ParticleSystem:** Δημιουργία οπτικών εφέ όπως εκρήξεις, καιρικά φαινόμενα ή σκόνη.
- ❖ **SoundCue:** Σχεδίαση και υλοποίηση προσαρμοζόμενων ηχητικών συστημάτων για μουσική υποβάθρου (background music) και ηχητικά εφέ.
- ❖ **Decal:** Εφαρμογή decals για εφέ όπως τρύπες από σφαίρες ή πιτσιλιές αίματος.
- ❖ **LightActor:** Ρύθμιση των στοιχείων φωτισμού.

6. Μεμονωμένα λειτουργικά στοιχεία

- ❖ **PhysicsVolume:** Χρήση όγκων φυσικής για τοπική εφαρμογή στοιχείων φυσικής με συγκεκριμένες ρυθμίσεις.
- ❖ **BlueprintFunction Library:** Ανάπτυξη βιβλιοθηκών με συναρτήσεις που μπορούν να καλεστούν από οπουδήποτε.

7. Τελική Σύνθεση και Πακετάρισμα

- ❖ **Πακετάρισμα του παιχνιδιού:** Compile και πακετάρισμα σε εκτελέσιμη μορφή από τον τελικό χρήστη.
- ❖ **Δοκιμή:** Διεξαγωγή διεξοδικών δοκιμών για την διασφάλιση της ορθής λειτουργίας των στοιχείων του παιχνιδιού, χωρίς σφάλματα.

8. Υποστήριξη μετά την Κυκλοφορία

- ❖ **Ενημέρωση και διόρθωση του παιχνιδιού** όπου και όπως είναι απαραίτητο, μέσα από την χρήση των σχολίων των παικτών.

Αυτή η δομημένη προσέγγιση εμπεριέχει ένα μεγάλο μέρος των στοιχείων που χρησιμοποιούνται για την κατασκευή ενός παιχνιδιού και επιδεικνύει κάποιες βασικές στρατηγικές και εργασίες που παίρνουν μέρος στην πολυπαραγοντική διαδικασία κατασκευής διαδραστικών ψηφιακών κόσμων.

3.4 Οντότητες μέσα στην Μηχανή

Στην Unreal Engine 5, οι οντότητες / αντικείμενα, αναφερόμενες συνήθως ως actors μέσα στον κώδικα / στην μηχανή, είναι τα θεμελιώδη αντικείμενα από τα οποία αποτελείται ο εικονικός χώρος και αλληλεπιδρούν μέσα στον κόσμο του παιχνιδιού. Τα actors, αυτά, είναι οποιαδήποτε αντικείμενα μέσα στο περιβάλλον της Unreal που μπορούν να συμπεριληφθούν εντός ενός εικονικού χώρου και αποτελούν την βάση της αναπαράστασης των μεμονωμένων στοιχείων του παιχνιδιού [32].

Στην Unreal Engine 5, τα actors είναι εξαιρετικά διαχειρίσιμα και μπορούν να αναπαριστούν ένα ευρύ φάσμα οντοτήτων, από απλά στατικά αντικείμενα όπως ένα κουτί, έως σύνθετα διαδραστικά συστήματα όπως μια πόρτα με κλειδαριές ή ένα όχημα. Κάθε οντότητα actor στο περιβάλλον της Unreal είναι μια περίπτωση (instance) της βασικής κλάσης AActor [32], η οποία παρέχει μεγάλο εύρος ιδιοτήτων και λειτουργιών, επιτρέποντάς τους να αλληλεπιδρούν με τον κόσμο του παιχνιδιού, καθώς και μεταξύ τους. Η αντικειμενοστραφής φύση των actors επιτρέπει την δημιουργία πολύπλοκων ρουτινών συμπεριφοράς μέσω κληρονομικότητας (inheritance) και σύνθεσης (composition), αξιοποιώντας την πολυμορφία για να επεκτείνουν και να προσαρμόζουν τις λειτουργίες τους για διάφορους σκοπούς.

Μια από τις βασικές λειτουργίες των actors προκύπτει μέσα από την ικανότητά τους να «κρατούν» διάφορα «υπο-αντικείμενα» που ονομάζονται «components». Τα components είναι περιπτώσεις (instances) κλάσεων που προέρχονται από το UActorComponent και χρησιμοποιούνται για την προσθήκη λειτουργικότητας στα actors. Για παράδειγμα, ένα UStaticMeshComponent μπορεί να προστεθεί σε ένα actor για να του δώσει ένα ορατό σχήμα (το σχήμα του 3D mesh) στον κόσμο του παιχνιδιού, ενώ ένα UCharacterMovementComponent μπορεί να χρησιμοποιηθεί για τη διαχείριση σύνθετης λογικής κίνησης για έναν χαρακτήρα actor. Αυτή η αρθρωτή (modular) προσέγγιση επιτρέπει την δυνατότητα κατασκευής σύνθετων actors αντικειμένων με ικανότητες που ανταποκρίνονται σε συγκεκριμένα σενάρια χρήσης αυτών και καθιστά ευκολότερη τη διαχείριση και ενημέρωση της συμπεριφοράς τους ανεξάρτητα.

Η διεπαφή της μηχανής της Unreal Engine 5 παρέχει ένα εκτεταμένο σύνολο εργαλείων για την τοποθέτηση και ρύθμιση των actors μέσα σε μια σκηνή / επίπεδο του χώρου παιχνιδιού του παιχνιδιού. Η διεπαφή χρησιμοποιείται για την χειροκίνητη τοποθέτηση των actors, για την ρύθμιση των ιδιοτήτων τους και την δημιουργία σχέσεων μεταξύ τους. Αυτή η διαχειριστικότητα

των actors καθιστά εύκολη την κατασκευή πολύπλοκων δομών. Επιπλέον, το σύστημα οπτικής Blueprint της Unreal Engine 5 ενσωματώνει τις λειτουργίες των actors, επιτρέποντας την μεταποίηση τους σε πραγματικό χρόνο. [32]

3.5 Ο μεταπρογραμματισμός και οι δυναμικές διαδικασίες κατά την αφομοίωση νέων οντοτήτων

Η C++ είναι μια ισχυρή και ευέλικτη γλώσσα που προτιμάται για τον -υψηλής ακρίβειας- έλεγχο που προσφέρει επί των πόρων του συστήματος και την απόδοση της, αλλά απαιτεί τροποποιήσεις και προσαρμογές για να χρησιμοποιηθεί στις νέες απαιτήσεις του κόσμου των συναλλασσόμενων και μεταλλασσόμενων οντοτήτων, που φέρουν τα δίκτυα Blockchain.

Οι τεχνολογίες Blockchain επιτρέπουν την συναλλαγή ψηφιακών οντοτήτων και κώδικα άγνωστης φύσης, που χαρακτηρίζεται από μεταδεδομένα. Τα ίδια τα όντα, κατά την μεταφορά τους από το ένα σύστημα στο άλλο, μπορούν να φέρουν μαζί τους κώδικα, ιδιότητες και δυνατότητες που το σύστημα που διαδρά μαζί τους δεν εμπεριέχει. Έτσι και η άνοδος του κλάδου της επαυξημένης και εικονικής πραγματικότητα, του metaverse, του gamification και των έξυπνων συμβάσεων (smart contract) δημιουργεί ένα πλαίσιο απαιτήσεων και μεταβλητών το οποίο μπορεί να διευθετηθεί μόνο μέσα από την αφομοίωση νέων και ευέλικτων χαρακτηριστικών που δεν υπάρχουν στην κλασική C++, με την οποία έχει γραφτεί η μεγάλη πλειοψηφία των μηχανών βιντεοπαιχνιδιών στην βιομηχανία.

Αυτές οι νέες τεχνολογίες δημιουργούν μεταβαλλόμενα προβλήματα αγνώστου φύσεως και απαιτούν μετατροπές στα ήδη υπάρχοντα βιντεοπαιχνίδια. Οι μεταβλητές αυτές συνθήκες, καθώς και οι νέες τεχνολογικές δυνατότητες που έχουμε εν έτη 2024, δίνουν έμφαση στο ον που συναλλάσσεται και μέσα από αυτό προσδιορίζουν την φύση του χώρου ύπαρξης του. Είναι επόμενο λοιπόν, οι προγραμματιστικοί μέθοδοι με τους οποίους συνθέτουμε κώδικα για τα βιντεοπαιχνίδια να αλλάζει μορφή από Αντικειμενοστραφή Προγραμματισμό (Object-Oriented Programming - OOP) σε Entity-Component System (ECS) μορφή σχεδιασμού με βάση τα δεδομένα (data-driven design). Παράλληλα, απαιτείται μια σειρά από δυνατότητες που επιφέρουν δυναμικά στοιχεία, όπως η δυνατότητα υποστήριξης δυναμικών διεργασιών.

Ένα εκ των δυναμικών στοιχείων είναι η δυναμική σύνταξη, η οποία απαιτεί την μεταγλώττιση κομματιών κώδικα κατά την εκτέλεση (σε αντίθεση με την κλασική πλήρη μεταγλώττιση στο compile time), καθιστώντας τις εφαρμογές πιο ευέλικτες και προσαρμοστικές. Αυτό είναι μία ουσιαστική προαπαιτήση των interoperable στοιχείων του metaverse και για τις εφαρμογές

blockchain, όπου τα περιβάλλοντα δεν είναι στατικά και μπορούν να αλλάξουν σε πραγματικό χρόνο με βάση τις αλληλεπιδράσεις των χρηστών ή τις συνθήκες του δικτύου / Blockchain.

Εδώ, είναι σημαντικό το να διατυπωθεί πως ο Μεταπρογραμματισμός αναφέρεται στην πρακτική της δημιουργίας / σύνταξης προγραμμάτων που μπορούν να χειρίζονται άλλα προγράμματα ως δεδομένα τους. Αυτό σημαίνει ότι ένα πρόγραμμα μπορεί να σχεδιαστεί για να διαβάσει, να δημιουργεί, να αναλύει ή να μετασχηματίζει άλλα προγράμματα ή να τροποποιεί τον εαυτό του δυναμικά κατά την διάρκεια της εκτέλεσης. Αυτή η δυνατότητα της «αυτό-μετάλλαξης» χρησιμοποιείται για την αύξηση της ευελιξίας και της απόδοσης. [33]

Στο πλαίσιο του metaverse, τα στοιχεία του παιχνιδιού είναι εξαιρετικά διαδραστικά και ανταποκρίνονται δυναμικά στις ενέργειες του χρήστη. Όταν εισάγονται νέα όντα μέσα σε αυτούς τους υπερ-διαδραστικούς χώρους, οι εφαρμογές πρέπει να μπορούν να τροποποιούν τους εαυτούς τους, να μεταγλωττίζουν και να ενσωματώνουν αλλαγές κώδικα σε πραγματικό χρόνο. Η δυναμική σύνταξη στην C++ μπορεί να υποστηρίξει αυτές τις ανάγκες επιτρέποντας την εφαρμογή αλλαγών χωρίς να χρειάζεται η επανεκκίνηση του προγράμματος, διατηρώντας έτσι τη συνέχεια και την αδιάσπαστη αλληλεπίδραση του χρήστη με τους εικονικούς χώρους, κάτι που είναι σημαντικό για τη διατήρηση της αίσθησης της εμπύθισης στον εικονικό κόσμο.

Ο μεταπρογραμματισμός, μια άλλη προηγμένη διαδικασία στην C++, παρέχει τη δυνατότητα δημιουργίας και διαχείρισης κώδικα κατά την διαδικασία μεταγλώττισης (compilation time) και κατά την διαδικασία εκτέλεσης (runtime). Ο μεταπρογραμματισμός επεκτείνει την ευελιξία της C++ επιτρέποντας τη δημιουργία αποδοτικών αλγορίθμων που μπορούν να προσαρμοστούν σε δεύτερο χρόνο με βάση τις συγκεκριμένες ανάγκες της εφαρμογής σε μελλοντικό χρόνο. Για την τεχνολογία blockchain, αυτό σημαίνει τη δυνατότητα δημιουργίας κώδικα που μπορεί να προσαρμοστεί δυναμικά σε διάφορα στις διάφορες απαιτήσεις των ανταλλάξιμων όντων μέσα από ένα σύστημα Blockchain. Η εξέταση και προσαρμογή του κώδικα κατά τη διάρκεια της εκτέλεσης, τις οποίες προσφέρει ο μεταπρογραμματισμός, επιτρέπει στους προγραμματιστές να δημιουργούν προγράμματα που έρχονται σε διάδραση με οντότητες που δεν ανήκουν στον χώρο του παιχνιδιού κατά την διάρκεια σχεδίασης.

Επιπλέον, στις κοινότητες του modding, όπου οι χρήστες επιδιώκουν συνεχώς βαθύτερη εξατομίκευση και επέκταση του παιχνιδιού και της διάδρασης, ο ρόλος του μεταπρογραμματισμού γίνεται ακόμα πιο σημαντικός. Οι modders μέσα από την άνοδο της

τεχνολογίας των Blockchain και των έξυπνων συμβολαίων, έχουν τώρα μια πλατφόρμα για να συναλλάσσουν την δουλειά τους και να διαδρούν με επικυρωμένες ψηφιοποιημένες οντότητες. Ήδη, η δουλειά των modders, αφορούσε διαδικασίες δυναμικού προγραμματισμού, όπως το DLL Injection ή χρησιμοποιώντας εργαλεία τροποποίησης που ακολουθούσαν τεχνικές μεταπρογραμματισμού.

Ο μεταπρογραμματισμός στην C++ επιτρέπει τη δημιουργία προσαρμοσμένου και ευέλικτου κώδικα που μπορεί να προσαρμοστεί ή να επεκταθεί σύμφωνα με τις απαιτήσεις των χρηστών ή των εισερχόμενων πακέτων μεταδεδωμένων οντοτήτων. Επιπλέον, η δυναμική σύνταξη υποστηρίζει την κοινότητα του modding επιτρέποντας αλλαγές και δοκιμές κώδικα σε πραγματικό χρόνο χωρίς την ανάγκη της συστηματικής επαναμεταγλώττισης προωθώντας μια πιο δημιουργική και ανενόχλητη διαδικασία ανάπτυξης βιντεοπαιχνιδιών.

Στην C++, λοιπόν, δεν υπάρχει φυσική υποστήριξη για τις περισσότερες από τις διαδικασίες του μετα-προγραμματισμού. Ο μεταπρογραμματισμός, χωρίς την χρήση κάποιας εξωτερικής βιβλιοθήκης ή ενός εξειδικευμένου compiler, γίνεται εφικτός μέσα από τον χειρισμό των templates, όμως αυτή η μορφή μεταπρογραμματισμού δεν επιτρέπει την ανασύνθεση μεταβλητών μετά το compile. Αν και τα templates δεν σχεδιάστηκαν αρχικά (1990) για αυτόν τον σκοπό. Αυτή η μέθοδος αποκαλείται "TMP" Template MetaProgramming και είναι κατά βάση μηχανισμός που παίρνει μέρος κατά την διαδικασία του compilation (μεταγλώττισης) του προγράμματος και επιτρέπει την δημιουργία και την βελτιστοποίηση εξειδικευμένου κώδικα κατά τη διαδικασία του compilation (μεταγλώττισης). Όταν, όμως, απαιτείται δυναμική τροποποίηση των στοιχείων του κώδικα κατά την λειτουργία του προγράμματος, ο μεταπρογραμματισμός επιτυγχάνεται δυναμικά μέσα από την χρήση των ακόλουθων προσεγγίσεων ή μια μίξη αυτών:

Βιβλιοθήκες Δυναμικού Συνδέσμου (Dynamic Link Libraries - DLLs)

Οι βιβλιοθήκες δυναμικού συνδέσμου (DLL) μπορούν να φορτωθούν κατά τη διάρκεια της εκτέλεσης του προγράμματος (runtime), επιτρέποντας την εισαγωγή νέων ενοτήτων (modules) σε μια εφαρμογή C++ χωρίς την ανάγκη επαναμεταγλώττισης (recompilation) του συνολικού κώδικα. Τα DLL χρησιμοποιούνται συχνά για την υποστήριξη του modding στα βιντεοπαιχνίδια και για την πρόσβαση των ίδιων των βιντεοπαιχνιδιών στα API του DirectX. Τα παιχνίδια χρησιμοποιούν αρχεία όπως το d3d11.dll και το d3d9.dll σαν βιβλιοθήκες με έτοιμες

συναρτήσεις που λειτουργούν ως στρώμα αφαίρεσης (abstraction layer) προς την κάρτα γραφικών GPU. Πολλά εργαλεία για μετατροπές παιχνιδιών την ώρα του τρεξίματος του παιχνιδιού (runtime) χρησιμοποιούν τεχνικές κατάληψης (hijacking) του αρχείου DLL και εισάγουν τρίτα εξωτερικά DLL ή script (γλώσσες script) αρχεία, δυναμικά, κατά το τρέξιμο του παιχνιδιού, τα οποία εμπεριέχουν κώδικα.

Ενσωμάτωση Γλωσσών Scripting (Embedding Scripting Languages)

Η ενσωμάτωση γλωσσών όπως της Python ή της Lua μπορεί να προσφέρει δυνατότητες δυναμικού scripting (κώδικα – γλώσσες script). Αυτό επιτρέπει την αλλαγή της λογικής του παιχνιδιού χωρίς επαναμεταγλώττιση (recompiling), και ένα εξωτερικό πρόγραμμα ή οι ίδιοι οι παίκτες μπορούν να εισάγουν δικά τους scripts για το modding του βιντεοπαιχνιδιού.

Αρχιτεκτονική Πρόσθετων (Plugin Architecture)

Μια εφαρμογή μπορεί να σχεδιαστεί ώστε να φορτώνει πρόσθετα (plugins) κατά τη διάρκεια της εκτέλεσης του προγράμματος (runtime). Αυτά τα πρόσθετα μπορούν να μεταγλωττιστούν (compile) ξεχωριστά και να επεκτείνουν ή να αλλάζουν δυναμικά τη λειτουργικότητα της κύριας εφαρμογής (main application).

Δυναμικός Μεταγλωτισμός – Interpreters

Οι μεταγλωττιστές όπως το Clang παρέχουν APIs για τη συντακτική ανάλυση, βελτιστοποίηση και παραγωγή κώδικα μηχανής (machine code) κατά τη διάρκεια εκτέλεσης (runtime). Μέσα από το Clang (front-end) σε συνδυασμό με το LLVM* (back-end) τμήματα κώδικα C++ (C++ code snippets) μπορούν να μεταγλωττιστούν (compiled) δυναμικά και να ενσωματωθούν σε μια εφαρμογή. Αυτό μπορεί να επιτευχθεί μέσα από διερμηνείς (interpreters) της C++, όπως το Cling που εισάγει την έννοια του "επαναπροσδιορισμού οντοτήτων" ("entity redefinition") και επιτρέπει τη δυναμική τροποποίηση των οντοτήτων και του κώδικα του παιχνιδιού [36][37][38].

*Το LLVM - Low-Level Virtual Machine "Εικονική μηχανή χαμηλού επιπέδου" είναι ένα framework μεταγλωττιστή (compiler framework) και ένα σύνολο επαναχρησιμοποιήσιμων βιβλιοθηκών (reusable libraries) για τη δημιουργία, τη βελτιστοποίηση και τη παραγωγή κώδικα εγγενούς μηχανής (machine-native code).

Μεταπρογραμματισμός με Πρότυπα (Template Metaprogramming - TMP)

Παρόλο που το TMP είναι μια δυνατότητα κατά τον χρόνο μεταγλώττισης (compile-time), μπορεί να παράγει εξαιρετικά ευέλικτες και δυναμικές δομές κώδικα. Το TMP επιτρέπει την δημιουργία κώδικα που προσαρμόζεται βάσει τύπου (type) ή ακόμη και αριθμητικών σταθερών (numerical literals), προσφέροντας μια μορφή ευελιξίας.

Ανάκληση και Ενδοσκόπηση (Reflection and Introspection)

Ορισμένος περιορισμένος βαθμός ανάκλησης μπορεί να επιτευχθεί στη C++ μέσω του RTTI (Run-time Type Information) ή προσαρμοσμένων μεταδεδομένων. Αυτό επιτρέπει δυναμική συμπεριφορά βάσει του τύπου μιας μεταβλητής. Μια παρόμοια μέθοδο χρησιμοποιείται στην μηχανή Unreal Engine 5 αλλά όχι σύστημα RTTI, καθώς το RTTI συνήθως παράγει πολύ μεγάλο όγκο μεταδεδομένων και αριθμό συνδέσμων προς το αρχείο εκτέλεσης για κάθε πολυμορφική μεταβλητή [42][43].

Έκθεση των εσωτερικών διαδικασιών, που είναι αποτυπωμένες σε κώδικα, σε γλώσσες Script

Είναι εφικτό να γίνει χρήση κάποιας Script γλώσσας (όπως Lua) για την δυναμική μορφοποίηση μεγάλου μέρους του παιχνιδιού κατά την εκτέλεση της εφαρμογής. Αυτό επιτρέπει την δυναμική συμπεριφορά με την εξωτερικοποίηση όσο το δυνατόν περισσότερης "λογικής" παιχνιδιού (συστήματα / game systems) σε εξωτερικά αρχεία, τα οποία μπορούν να φορτωθούν και να ερμηνευθούν κατά τη διάρκεια εκτέλεσης (runtime). Θυμίζει την μετατροπή κομματιών του παιχνιδιού σε mods.

Σύνταξη Άμεσης Μεταγλώττισης (Just-In-Time Compilation - JIT)

Παρόλο που είναι πιο συνηθισμένη σε γλώσσες όπως η Java, το JIT μπορεί να χρησιμοποιηθεί στη C++ για τη μεταγλώττιση τμημάτων κώδικα κατά τη διάρκεια εκτέλεσης (runtime), προσφέροντας έναν δυναμικό τρόπο για την εισαγωγή νέας λειτουργικότητας [37].

Δυναμικοί Δείκτες Συναρτήσεων (Dynamic Function Pointers)

Μπορούν να χρησιμοποιηθούν δείκτες συναρτήσεων για τη δυναμική κλήση συναρτήσεων κατά τη διάρκεια εκτέλεσης του κώδικα (runtime), προσφέροντας έναν τρόπο για την δυναμική αλλαγή της συμπεριφοράς του παιχνιδιού.

Συστήματα "Γεγονότων" (Event Systems)

Η υλοποίηση μιας αρχιτεκτονικής βασισμένης σε γεγονότα (Events) προσφέρει δυναμικά στοιχεία, κάνοντας μέρη του συστήματός ανεξάρτητα (decoupling) και επιτρέποντάς τους να αλληλεπιδρούν με ευέλικτους τρόπους.

Modding API

API που επιτρέπει την δημιουργία και την πρόσθεση τροποποιήσεων στο παιχνίδι, δίνοντας πρόσβαση στις λειτουργίες του παιχνιδιού με ασφαλή και ελεγχόμενο τρόπο.

Cppfront compiler του Herb Sutter και μια "νέα" C++

Ο Herb Sutter είναι Αρχιτέκτονας λογισμικού στην Microsoft, πρόεδρος της επιτροπής προτύπων ISO της C++. Το Cppfront είναι ένας compiler που επεκτείνει την C++ σε μια νέα και πειραματική της έκδοση που θυμίζει μια C++ με σύνταξη έκδοσης νούμερο 2 (Cpp2). Έρχεται να κάνει την C++ πιο ασφαλή σε ότι αφορά τα δεδομένα στην μνήμη, να κάνει standard μερικά functions του iostream, να ενώσει τους τύπους και πολλές άλλες αλλαγές, ανάμεσα τους το reflection σε runtime και τα metafunctions. Θεωρείται σαν μια ανανέωση της C++ που φέρει φόρτο προβλημάτων από τον καιρό της δημιουργίας της γλώσσας C. Στο συγκεκριμένο project δίνει ανατροφοδότηση και ο Tim Sweeny, ιδρυτής και Διευθύνων Σύμβουλος της Epic Games και δημιουργός της μηχανής βιντεοπαιχνιδιών Unreal Engine. Αυτό το έργο και το πλαίσιο των ανθρώπων που το προωθούν και συμβάλλουν δείχνει την πρόθεση της κατασκευής μιας γλώσσας που θα "αντικαταστήσει" ή θα επεκτείνει την C++, όπως η C++ επέκτεινε την C, και θα εμπεριέχει στοιχεία δυναμικού προγραμματισμού και code generation - διατηρώντας ταυτόχρονα την στατικότητα της C++ [40][41].

Μία λειτουργική εφαρμογή των προαναφερόμενων μεθόδων / συστημάτων μπορεί να χρησιμοποιήσει μεταδεδομένα και χαρακτηρισμούς των νομισμάτων στο IPFS για να αφομοιώσει τα διάφορα νομίσματα του χρήστη στο παιχνίδι και τις λειτουργίες τους.

Παράλληλα τα δυναμικά στοιχεία προ-απαιτούνται, ανεξάρτητα από την ικανότητα δυναμικής αναμεταγλώττισης και reflection, για την χρήση των δεδομένων των περιουσιακών στοιχείων του χρήστη στο Blockchain και την σύνταξη νέων συμβολαίων και την επικοινωνία μαζί τους μέσα από τον εικονικό κόσμο, σε πραγματικό χρόνο.

Η ίδια η Unreal Engine 5 που χρησιμοποιείται στο πρακτικό μέρος, φέρει ασύγχρονες δυναμικές διαδικασίες και μεταπρογραμματιστικά στοιχεία μέσα από ένα ευρύ φάσμα μεθόδων. Σε δεύτερο

βαθμό, επειδή ο κώδικας της μηχανής είναι ανοιχτής μορφής και βρίσκεται στο Github, είναι εφικτή η εφαρμογή οποιασδήποτε μεθόδου που προαναφέρεται έξω από τις ήδη υπάρχουσες δυνατότητες που προσφέρει η Unreal Engine 5. Όμως η διαδικασία τροποποίησης του πηγαίου κώδικα της μηχανής είναι περιπλοκή και μπορεί να δημιουργήσει πολλά κενά ασφαλείας στο τελικό πρόγραμμα.

Είναι κατανοητό πως ένα παιχνίδι που έχει σαν βασικό του στοιχείο την μετάλλαξη, πρέπει το ίδιο να εντοιχίζει την μηχανή μέσα του, ώστε να μπορεί να επιτύχει υψηλό βαθμό αφομοίωσης των τρίτων εξωτερικών οντοτήτων. Επειδή όμως αυτό δεν είναι εφικτό, ο μεταπρογραμματισμός απαιτεί τον προϋπάρχον προγραμματισμό συναρτήσεων και την κατασκευή pipeline που διαχειρίζεται τα εισερχόμενα στοιχεία στον μέγιστο βαθμό που μπορεί εκείνο.

Ταυτόχρονα, μειώνοντας τον βαθμό αφομοίωσης περαιτέρω, το παιχνίδι μπορεί να χρησιμοποιεί μια μετάφραση ή να αφομοιώνει επιλεκτικά μόνο τα μεταδεδομένα που είναι παρεμφερή με την φύση του – ή την προβολή αυτών (εικονική παρουσίαση αυτών).

Για όλες τις παραπάνω διαδικασίες η Unreal Engine 5 έχει τα ακόλουθα συστήματα και api:

Async functions / tasks (Ασύγχρονες συναρτήσεις / διαδικασίες)

Υπάρχουν διάφορες προϋπάρχουσες συναρτήσεις ή κλήσεις API μέσα από τα Blueprint που λειτουργούν ασύγχρονα (Async) και μπορούν να εκτελούν σημαντικές διαδικασίες κατά την ώρα εκτέλεσης του προγράμματος, όπως το κατέβασμα δεδομένων από το IPFS ή η εναλλαγή δεδομένων σε πίνακες Data Table. Μέσα από την χρήση αυτών των ασύγχρονων διαδικασιών μέσα στα Blueprint και έξυπνες τεχνικές προγραμματισμού με στατικό κώδικα είναι εφικτή η πλήρης αφομοίωση απλών στοιχείων παιχνιδιού από ένα σύστημα Blockchain.

Σε δεύτερο βαθμό υπάρχουν πολλά επιπρόσθετα που προσφέρουν μία σειρά από χρήσιμα έτοιμα εργαλεία με νέες σύγχρονες λειτουργίες.

Unreal Engine Reflection System

Το σύστημα reflection αποτελεί μια θεμελιώδη τεχνολογία της μηχανής Unreal Engine, υποστηρίζοντας πολλά συστήματα, όπως τα πάνελ λεπτομερειών στην διεπαφή χρήστη της εφαρμογής, τη σειριοποίηση (serialization), τη συλλογή απορριμμάτων (garbage collection), το

network replication και την επικοινωνία μεταξύ Blueprint και C++. Η Unreal διαθέτει το δικό της σύστημα reflection, καθώς η C++ δεν διαθέτει ένα τέτοιο σύστημα από μόνη της, το οποίο ονομάζεται σύστημα ιδιοτήτων (property system). Για την λειτουργία του συστήματος reflection είναι απαραίτητος ο χαρακτηρισμός των τύπων και ιδιοτήτων (types και properties) για να επεξεργαστούν από το σύστημα reflection. Αυτό το γίνεται μέσα από την χρήση μακροεντολών (macros) όπως UCLASS(), UPROPERTY() και UFUNCTION(), οι οποίες προσθέτουν μεταδεδομένα, σε κλάσεις που προέρχονται από το UObject, τα οποία μεταδεδομένα βοηθούν στην αναγνώριση και διαχείριση των κλάσεων και των ιδιοτήτων από τη μηχανή. Έπειτα, το Unreal Header Tool (UHT) συλλέγει αυτές τις πληροφορίες της διαδικασίας reflection κατά τη μεταγλώττιση (compile time)[43][43].

Dynamic Scripting

Για την πραγματική δυναμική μεταγλώττιση (recompilation) κώδικα, την ώρα εκτέλεσης της εφαρμογής (runtime), απαιτείται η εφαρμογή μιας γλώσσας script, όπως Lua ή Python (όχι όμως, μέσω της Python που ήδη συμπεριλαμβάνεται, καθώς κάνει compile μαζί με το παιχνίδι), μέσω πρόσθετων. Αυτά τα πρόσθετα επιτρέπουν τη δυνατότητα scripting που μπορεί να τροποποιηθεί κατά τη διάρκεια της εκτέλεσης[44].

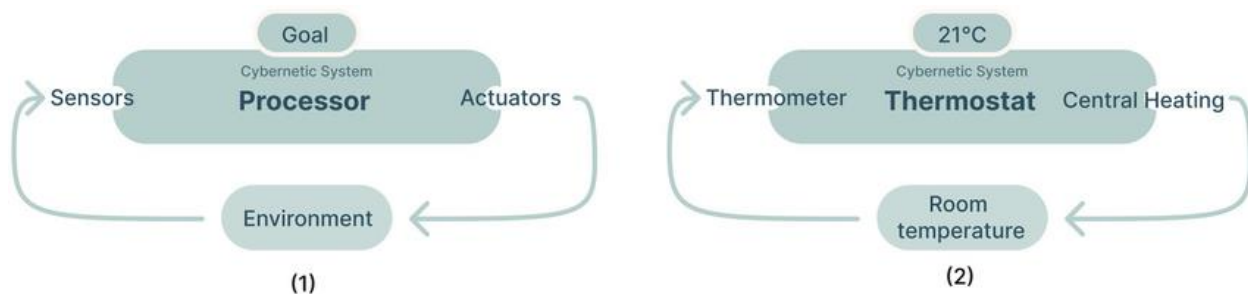
Κεφάλαιο 4 Κυβερνητικές εφαρμογές και ψηφιοποίηση αντικειμένων

4.1 Κυβερνητικά συστήματα

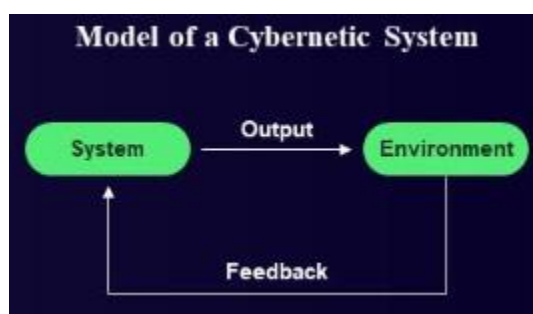
4.1.1 Ορισμός και εισαγωγή

Ο όρος "cybernetics" (Κυβερνητική) επινοήθηκε από τον Norbert Wiener τη δεκαετία του 1940, προερχόμενος από την ελληνική λέξη "κυβερνήτης [47]. Ο Wiener όρισε τα cybernetics ως την επιστήμη του ελέγχου και της επικοινωνίας στα ζώα (ζωντανούς οργανισμούς πάσης φύσεως) και στις μηχανές. Η επιστήμη της κυβερνητικής μελετά τον τρόπο με τον οποίο τα συστήματα μπορούν να ελέγχονται και να ρυθμίζονται μέσω μηχανισμών ανατροφοδότησης, όπως ένας θερμοστάτης, που διατηρεί μία καθορισμένη θερμοκρασία προσαρμόζοντας τη θέρμανση ή την ψύξη βάσει της τρέχουσας θερμοκρασίας. Η φύση των κυβερνητικών είναι διεπιστημονική και τα ίδια τα κυβερνητικά συστήματα χαρακτηρίζονται ως, αλλά και αφορούν, πολυσωματικά συμβιωτικά συστήματα και τον έλεγχό τους.

Τα cybernetics, έτσι, είναι η διεπιστημονική μελέτη των πολύπλεκτων συστημάτων, του ελέγχου και της επικοινωνίας σε ζώντες οργανισμούς, μηχανές και κοινωνικοπολιτικούς «οργανισμούς». Ως επιστήμη εστιάζει στον τρόπο με τον οποίο τα συστήματα αυτορυθμίζονται και επεξεργάζονται πληροφορίες μέσω βρόχων ανατροφοδότησης, για να διατηρήσουν τη σταθερότητα και να επιτύχουν τους στόχους τους [48] και η εφαρμογή της δημιουργεί ένα μετασύστημα ελέγχου και ανατροφοδότησης / μετουσίωσης της πληροφορίας. Σαν ενδιάμεσος "κόμβο" κλειδί για την επικοινωνία και την επεξεργασία, τα ίδια τα κυβερνητικά συστήματα ενσωματώνουν αρχές από τη μηχανική, τη βιολογία, την επιστήμη των υπολογιστών και πολλούς άλλους τομείς για την κατανόηση και το σχεδιασμό αποδοτικών και προσαρμοστικών συστημάτων [49].



Απλουστευμένη ανάλυση του συστήματος ανατροφοδότησης σε 2 απλά κυβερνητικά συστήματα
 [Πηγή: “A framework for designing AI systems that support community wellbeing”, Jan 2023
 Willem van der Maden , Derek Lomas, Paul Hekkert]



Απλουστευμένο μοντέλο του feedback loop ενός απλού κυβερνητικού συστήματος

Ένα σύστημα θεωρείται "cybernetic" (κυβερνητικό) όταν σχεδιάζεται με έμφαση στις διαδικασίες ελέγχου και επικοινωνίας ανάμεσα στα διάφορα μέρη του. Και κύρια λειτουργικά στοιχεία ενός, είναι το πώς μεταδίδονται οι πληροφορίες, πώς χρησιμοποιείται η ανατροφοδότηση για την προσαρμογή των ενεργειών και πώς το σύστημα προσαρμόζεται στις αλλαγές του περιβάλλοντός του. Ουσιαστικά, όταν εστιάζουμε στον έλεγχο, την επικοινωνία και την ανατροφοδότηση εντός ενός συστήματος πολλών σωμάτων, εξετάζουμε τις κυβερνητικές του ιδιότητες [47][48].

Σε αντίθεση, ένα σύστημα πολλών μερών που δεν είναι κυβερνητικό λειτουργεί χωρίς την έμφαση στις διαδικασίες ελέγχου και επικοινωνίας. Σε τέτοια συστήματα, οι πληροφορίες μπορεί να μην μεταδίδονται αποτελεσματικά, η ανατροφοδότηση να μην χρησιμοποιείται για την προσαρμογή των ενεργειών και το σύστημα να μην προσαρμόζεται επαρκώς στις αλλαγές του περιβάλλοντός του. Ένα μη κυβερνητικό σύστημα μπορεί να έχει λιγότερα λειτουργικά μέρη,

αλλά δεν διαθέτει τους μηχανισμούς για τον συντονισμό των διάφορων εξαρτημάτων ή μερών και την ρύθμιση τους μέσω της ανατροφοδότησης.

Συγκρίνοντας ένα σύστημα πολλών μερών που δεν είναι κυβερνητικό με ένα κυβερνητικό σύστημα, οι διαφορές είναι εμφανείς στις διαδικασίες ελέγχου και επικοινωνίας. Ένα μη κυβερνητικό σύστημα λειτουργεί χωρίς έμφαση σε αυτές τις διαδικασίες, με αποτέλεσμα οι πληροφορίες να μην μεταδίδονται αποτελεσματικά και η ανατροφοδότηση να μην χρησιμοποιείται για την προσαρμογή των ενεργειών / εργασιών του συστήματος. Από την άλλη πλευρά, ένα κυβερνητικό σύστημα αυτορυθμίζεται και προσαρμόζεται δυναμικά στις αλλαγές του περιβάλλοντός του, χρησιμοποιώντας ανατροφοδότηση για να διατηρεί τη σταθερότητα και να επιτυγχάνει τους στόχους του, καθιστώντας το πιο χρήσιμο και αποδοτικό σε απομακρυσμένα συστήματα, σε βιομηχανίες με συνθήκες ταχείας τεχνολογικής εξέλιξης και σε συστήματα που έρχονται σε άμεση επαφή με πολυπαραγοντικές περιβαλλοντικές μεταβλητές ή με συνειδητούς οργανισμούς.

Έτσι, η βιομηχανία κατευθύνεται όλο και περισσότερο προς τα κυβερνητικά συστήματα. Σύμφωνα με την έκθεση της McKinsey για τις τεχνολογικές τάσεις του 2023 [46], η επένδυση σε τεχνολογίες όπως η εφαρμοσμένη AI (τεχνητή νοημοσύνη), η προχωρημένη συνδεσιμότητα των μοντέρνων ηλεκτρικών και ηλεκτρονικών συσκευών και το cloud computing παραμένει ισχυρή, με τις συνολικές επενδύσεις να ξεπερνούν το 1 τρισεκατομμύριο δολάρια το 2022. Η ενσωμάτωση των αρχών της κυβερνητικής στις βιομηχανίες αναμένεται να αυξηθεί σημαντικά, με τον αριθμό των κυβερνητικών συστημάτων να διπλασιάζεται τα επόμενα χρόνια, καθιστώντας τα κυρίαρχα σε πολλούς τομείς.

4.1.2 Βασική δομή ενός κυβερνητικού συστήματος

Αναλύοντας βαθύτερα τα λειτουργικά μέρη ενός κυβερνητικού συστήματος, τυπικά αποτελείται από τα εξής βασικά μέρη:

1. **Αισθητήρες:** Αυτοί ανιχνεύουν αλλαγές στο περιβάλλον και συλλέγουν δεδομένα. Οι αισθητήρες λειτουργούν ως μηχανισμός εισόδου, παρέχοντας πληροφορίες σε πραγματικό χρόνο για την κατάσταση του συστήματος και τις εξωτερικές συνθήκες.
2. **Controllers:** Επεξεργαστικές μονάδες που επεξεργάζονται τα δεδομένα που λαμβάνονται από τους αισθητήρες. Οι λογικές, αυτές, μονάδες, συνήθως καθοδηγούμενες από αλγόριθμους, αξιολογούν τα δεδομένα σε σχέση με τα επιθυμητά κριτήρια απόδοσης και

καθορίζουν τις απαραίτητες προσαρμογές που πρέπει να λάβει το σύστημα μέσα από τους Actuators.

3. **Actuators:** Αυτοί είναι μηχανισμοί που υλοποιούν τις αποφάσεις της λογικής μονάδας. Εφαρμόζουν αλλαγές στο σύστημα για να επιτύχουν το επιθυμητό αποτέλεσμα, λειτουργώντας ουσιαστικά ως μηχανισμοί εξόδου.
4. **Βρόχοι ανατροφοδότησης:** Μέσα από αυτούς γίνεται εφικτή η ρύθμιση της συμπεριφοράς του συστήματος. Διότι οι βρόχοι ανατροφοδότησης επιτρέπουν στο σύστημα να συγκρίνει την τρέχουσα κατάσταση με την επιθυμητή και να κάνει συνεχή αναπροσαρμογές. Τα θετικά αποτελέσματα κατά την ανατροφοδότηση ενισχύουν τις αλλαγές που οδήγησαν σε αυτή την θετική αλλαγή, ενώ τα αρνητικά αποτελέσματα οδηγούν το σύστημα στην αναίρεση ή μείωση των αλλαγών που πήραν μέρος κατά τον βρόχο, βοηθώντας στη σταθερότητα και τον έλεγχο [47][48].

4.1.3 Χρήσεις των Κυβερνητικών Συστημάτων

Τα κυβερνητικά συστήματα χρησιμοποιούνται σε διάφορους τομείς για την επίτευξη ελέγχου, αύξηση της αποδοτικότητας και προσαρμογή των συστημάτων, ακολουθούν ορισμένα γρήγορα παραδείγματα αυτών ανά κλάδο:

Βιομηχανική Αυτοματοποίηση: Στη βιομηχανία, τα κυβερνητικά συστήματα βελτιστοποιούν τις διαδικασίες παραγωγής μέσω αυτοματοποιημένου ελέγχου των μηχανημάτων, οδηγώντας σε βελτιωμένη αποδοτικότητα, μείωση ανθρώπινων σφαλμάτων και την δημιουργία “έξυπνων” και πιο αυτόματων συστημάτων. Παραδείγματα αυτών αποτελούν:

- τα IOT (internet of things) κυβερνητικά συστήματα. Αυτά τα συστήματα ενισχύουν τις λειτουργίες ελέγχου των μηχανών μέσα από την προέκταση του δικτύου και την δυνατότητα εύκολης και γρήγορης ένταξης μικρών νέων κόμβων που δρουν σαν αισθητήρια ή νέες λογικές μονάδες. Η ενσωμάτωση του IoT επιτρέπει την παρακολούθηση σε πραγματικό χρόνο και τον προσαρμοστικό έλεγχο των βιομηχανικών διαδικασιών για τη διαχείριση υλικών και την ασφάλεια σε περιβάλλοντα Βιομηχανίας 4.0.
- τα συστήματα με Προσαρμοστική Αυτονομία: Τα έξυπνα συνεργατικά κυβερνητικά συστήματα επιδεικνύουν περισσότερα ανθρώπινα και συνειδητά χαρακτηριστικά και έτσι επιτρέπουν την δημιουργία πιο ανθρώπινων συστημάτων αυτοματισμού επιτρέποντας

προσαρμοστικά επίπεδα αυτοματισμού ανάλογα με την πολυπλοκότητα της εργασίας και τη συμμετοχή του ανθρώπου. [50]

- η ήδη υπάρχουσα τεχνολογία που ανήκει στην Βιομηχανία 4.0. Αυτές οι τεχνολογίες συνδέουν βιομηχανικά αντικείμενα με το cloud, επιτρέποντας έξυπνο αυτοματισμό και βελτιώνοντας τη συνολική λειτουργική αποδοτικότητα.
- Τα Κυβερνο-Φυσικά Συστήματα (CPS). Η υλοποίηση των οποίων παρέχει ισχυρό έλεγχο, επεξεργασία δεδομένων σε πραγματικό χρόνο και απρόσκοπτη ενσωμάτωση των φυσικών και ψηφιακών κόσμων. [51]

Βιολογικά Συστήματα: Στη βιολογία και την ιατρική, η κυβερνητική βοηθά στην κατανόηση των φυσιολογικών διαδικασιών και στον σχεδιασμό ιατρικών συσκευών που ρυθμίζουν τις σωματικές λειτουργίες, όπως οι αντλίες ινσουλίνης για τη διαχείριση του διαβήτη.

Ρομποτική: Η κυβερνητική παίζει καίριο ρόλο στη ρομποτική, όπου οι μηχανισμοί ανατροφοδότησης επιτρέπουν στα ρομπότ να προσαρμόζονται σε μεταβαλλόμενα περιβάλλοντα και να εκτελούν πολύπλοκες εργασίες αυτόνομα. Τα κυβερνητικά συστήματα είναι βασικό στοιχείο της λειτουργίας των περισσότερων ρομποτικών συστημάτων και για αυτό θεωρούνται ένα αδιάσπαστο κομμάτι των ρομπότ. Τα κυβερνητικά επιτρέπουν:

- την εφαρμογή ιεραρχικών δομών ελέγχου σε ρομποτικά συστήματα για τη διαχείριση σύνθετων εργασιών και τον συντονισμό πολλαπλών ρομπότ.
- στον σχεδιασμό και τον έλεγχο ρομποτικών συστημάτων για την επίτευξη υψηλότερων επιπέδων αυτονομίας και νοημοσύνης.
- την βελτίωση της ακρίβειας και τον καλύτερο έλεγχο των χειρουργικών μηχανημάτων ρομποτικής χειρουργικής, μέσα από τα συστήματα επαυξημένης πραγματικότητας.
- την αλληλεπίδραση Ανθρώπου-Ρομπότ. Μέσα από την δυνατότητα δημιουργίας ανθρωποειδών ρομπότ που μπορούν να αλληλεπιδρούν με ανθρώπους και να εκτελούν εργασίες σε δυναμικά περιβάλλοντα.

Διοίκηση Οργανισμών: Στη διοίκηση, οι αρχές της κυβερνητικής χρησιμοποιούνται για τον σχεδιασμό οργανωτικών δομών και διαδικασιών που μπορούν να προσαρμόζονται σε εξωτερικές αλλαγές και μεταβαλλόμενες εσωτερικές μεταβλητές, βελτιώνοντας τη λήψη αποφάσεων και την επιχειρησιακή αποδοτικότητα.

4.1.4 Οφέλη των Κυβερνητικών Συστημάτων

Τα κυβερνητικά συστήματα προσφέρουν πολλαπλά οφέλη στις επιχειρήσεις και στους οργανισμούς. Ακολουθούν 10 βασικά οφέλη:

1. **Αυξημένη Αποδοτικότητα:** Τα κυβερνητικά συστήματα μπορούν να αυτοματοποιήσουν επαναλαμβανόμενες εργασίες, μειώνοντας τον χρόνο και τα κόστη που απαιτούνται για την εκτέλεσή τους.
2. **Προσαρμοστικότητα:** Τα κυβερνητικά συστήματα μπορούν να προσαρμοστούν σε μεταβαλλόμενες συνθήκες, παρέχοντας ευελιξία στις επιχειρηματικές διαδικασίες.
3. **Διαχείριση Πόρων:** Βοηθούν στη βέλτιστη κατανομή και χρήση των πόρων, όπως το ανθρώπινο δυναμικό και τα υλικά.
4. **Συνεχής Βελτίωση:** Τα δεδομένα και οι αναλύσεις αυτών, που παρέχονται από τα κυβερνητικά συστήματα, επιτρέπουν την ταυτοποίηση των σημείων βελτίωσης και την εφαρμογή διορθωτικών ενεργειών σε πραγματικό χρόνο.
5. **Βελτιωμένη Ακρίβεια:** Η αυτοματοποίηση και οι εξελιγμένοι αλγόριθμοι μειώνουν τα σφάλματα που προκύπτουν από την ανθρώπινη παρέμβαση και από εξωτερικές μεταβλητές / περιβαλλοντικές συνθήκες.
6. **Συγκέντρωση Δεδομένων:** Διευκολύνουν τη συλλογή και ανάλυση μεγάλου όγκου δεδομένων, επιτρέποντας τη λήψη καλύτερων αποφάσεων (βελτιστοποίηση).
7. **Βελτίωση της Επικοινωνίας:** Τα κυβερνητικά συστήματα μπορούν να ενσωματωθούν σε δίκτυα, βελτιώνοντας τη συνεργασία μεταξύ των διαφόρων συστημάτων μιας επιχείρησης.
8. **Αυξημένη Ασφάλεια:** Με τη χρήση τεχνολογιών όπως η κρυπτογράφηση και τα φυσικά συστήματα ανίχνευσης εισβολέων, τα κυβερνητικά συστήματα προσφέρουν υψηλότερο επίπεδο ασφάλειας.
9. **Ενίσχυση της Παραγωγικότητας:** Επιτρέπουν στους εργαζομένους να επικεντρωθούν σε πιο σημαντικές και στρατηγικές εργασίες, αυξάνοντας τη συνολική παραγωγικότητα.
10. **Μείωση Κόστους:** Η αυτοματοποίηση διαδικασιών και η μείωση των σφαλμάτων μπορεί να οδηγήσει σε σημαντική μείωση των λειτουργικών εξόδων.

Αυτά τα οφέλη καθιστούν τα κυβερνητικά συστήματα αναπόσπαστο μέρος της σύγχρονης επιχειρηματικής λειτουργίας, προσφέροντας στρατηγικά πλεονεκτήματα σε ένα ανταγωνιστικό

περιβάλλον και κατά την ένταξή τους σε καταναλωτικά αγαθά, επιτρέπουν την παροχή πιο βελτιστοποιημένων και έξυπνων υπηρεσιών και προϊόντων στον τελικό καταναλωτή.

4.2 ΛΟΓΙΚΕΣ ΜΟΝΑΔΕΣ, ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΚΥΒΕΡΝΗΤΙΚΩΝ ΚΑΙ ΤΑ ΠΑΡΑΓΟΜΕΝΑ ΜΕΤΑΔΕΔΟΜΕΝΑ

Η Κεντρική Μονάδα Επεξεργασίας (CPU) είναι η θεμελιώδης λογική μονάδα ενός κυβερνητικού συστήματος, λειτουργώντας ως ο εγκέφαλος που ενορχηστρώνει όλες τις υπολογιστικές διαδικασίες. Η σημασία της έγκειται στην ικανότητά της να εκτελεί εντολές, να διαχειρίζεται τη ροή δεδομένων και να πραγματοποιεί σύνθετους υπολογισμούς που είναι απαραίτητοι για τη λειτουργία του συστήματος. Σε κυβερνητικά συστήματα, που συχνά περιλαμβάνουν περίπλοκους βρόχους ανάδρασης και προσαρμοστικές συμπεριφορές, ο ρόλος του CPU είναι καθοριστικός για την εξασφάλιση επεξεργασίας και απόκρισης σε πραγματικό χρόνο. Ενσωματώνει εισροές από διάφορους αισθητήρες, επεξεργάζεται τις πληροφορίες σύμφωνα με προγραμματισμένους αλγόριθμους και παράγει “εντολές” που καθοδηγούν τις ενέργειες του συστήματος.

Ο προγραμματισμός της Κεντρικής Μονάδας Επεξεργασίας (CPU) και τα παραγόμενα μεταδεδομένα έχουν εξίσου μεγάλη σημασία για τη λειτουργία ενός κυβερνητικού συστήματος. Ο προγραμματισμός του CPU επιτρέπει την εκτέλεση συγκεκριμένων εργασιών και την εφαρμογή αλγορίθμων που καθορίζουν τη συμπεριφορά του συστήματος. Με ακριβή και αποτελεσματικό κώδικα, η Κεντρική Μονάδα Επεξεργασίας μπορεί να διαχειριστεί περίπλοκες διαδικασίες, να βελτιστοποιήσει την απόδοση και να εξασφαλίσει αξιόπιστη λειτουργία υπό διαφορετικές συνθήκες.

Κατά την εκτέλεση των προγραμματισμένων διαδικασιών παράγονται μεταδεδομένα από αισθητήρια και από την ίδια την διαδικασία της επεξεργασίας εντός του CPU.

Τα παραγόμενα μεταδεδομένα από τα αισθητήρια παρέχουν χρήσιμες πληροφορίες για την παρακολούθηση και ανάλυση της απόδοσης του συστήματος. Αυτά τα δεδομένα βοηθούν στη διάγνωση προβλημάτων, στη βελτιστοποίηση διαδικασιών και στην λήψη αποφάσεων βασισμένων σε πραγματικά δεδομένα. Μέσω της ανάλυσης των μεταδεδομένων, οι προγραμματιστές και οι μηχανικοί μπορούν να κατανοήσουν καλύτερα τη συμπεριφορά του

συστήματος, να εντοπίσουν σημεία βελτίωσης και να προσαρμόσουν τους αλγορίθμους για αυξημένη αποδοτικότητα και αξιοπιστία.

Τα δεδομένα που παράγονται λογικά κατά την επεξεργασία μπορούν να αποτελούν είτε στοιχεία από εξωτερικές μεταβλητές ή χαρακτηρισμοί και ιδιότητες για το αντικείμενο το οποίο παράγει, εξετάζει ή αναλύει το κυβερνητικό σύστημα. Για παράδειγμα, μέσα από την μηχανική όραση (computer vision / υπολογιστική όραση ή τεχνητή όραση) ένα ανθρωποειδές ρομπότ μπορεί να αναγνωρίσει έναν συγκεκριμένο άνθρωπο με τον οποίο έχει ξαναέρθει σε επαφή παλαιότερα και να διαδράσει μαζί του. Σε ένα άλλο παράδειγμα, σε μια γραμμή παραγωγής με εφαρμοσμένο σύστημα Kanban, ένα αυτόματο σύστημα μπορεί να κολλάει ετικέτες με μοναδικούς κωδικούς QR ανά παραγόμενο προϊόν για την μοναδική διαχείριση ανά αγαθό από τα υπόλοιπα συστήματα του εργοστασίου και ο κωδικός αυτός μπορεί να μεταφέρει χρήσιμα στοιχεία για τα υπόλοιπα μηχανήματα του εργοστασίου. Και στις δύο περιπτώσεις, γίνεται αναγνώριση οντοτήτων μέσω του κυβερνητικού συστήματος προς την λογική μονάδα, επιτρέποντας την περαιτέρω επεξεργασία και χαρακτηρισμό του όντος που αναγνωρίζεται. Αυτά τα δεδομένα θεωρούνται παραγόμενα μεταδεδομένα μέσα από την λογική διαδικασία της επεξεργασίας και φέρουν αναφορές προς ιδιότητες κάποιου δεύτερου όντος.

Προγραμματιστικά κάθε αντικείμενο που ξεχωριστά επεξεργάζεται ένα κυβερνητικό και του προσδίδει ιδιότητες και το ανάγει σε πίνακες και δομές (structures) θεωρείται ως ένα μοναδικό entity και έχει δικές του στατικές θέσεις στην μνήμη (μέχρι να καταστραφεί – Garbage Collection). Ως προς τον ίδιο τον προ-προγραμματισμένο επεξεργαστή και τις διαδικασίες του (functions) μπορεί ο συμβολισμός του σαν μια νέα οντότητα να μην είναι εμφανής, αλλά η ίδια η φύση των μεθόδων προγραμματισμού που χρησιμοποιούμε κάτω από τις κλασσικές αρχιτεκτονικές X86-X64 και ARM, επιτρέπει στους προγραμματιστές να τα διαχειρίζονται σαν μια ξεχωριστή οντότητα. Θεωρητικά η οντοποίηση που κάνει το ψηφιακό σύστημα στις πολλές μορφές της θεωρείται ψηφιοποίηση και η οποιαδήποτε μορφή αναγωγής ιδιοτήτων ανά μοναδική θέση μνήμης το καθιστά ένα ψηφιοποιημένο ον με μοναδικά τεκμήρια και ο χρόνος ύπαρξης του προς το ψηφιακό σύστημα καθορίζεται από το μέσο αποθήκευσής του. Αυτές είναι και οι αρχές του αντικειμενοστραφή προγραμματισμού (object-oriented programming (OOP)), τα πλαίσια του οποίου προκύπτουν φυσικά μέσα από την ανάγκη του ανθρώπου να χαρακτηρίζει τον περιβάλλοντα χώρο και να διαχειρίζεται και μετρά όντα.

Όταν τα μεταδεδομένα, αυτά, θεωρούνται χαρακτηρισμοί προς ένα ον, τότε ο μοναδικός δίαυλος αναφοράς προς το ον (ακριβής θέση μνήμης) θεωρείται ως το αναγνωριστικό του στοιχείο και το ον θεωρείτε ψηφιοποιημένο με μία αναφορά σε ένα πρωταρχικό «κέλυφος» ενός ψηφιακού διδύμου σε αφηρημένη μορφή. Το σύνολο όλων αυτών των χαρακτηριστικών, μαζί με τα στοιχεία προς την θέση μνήμης που καταλαμβάνονται, αποτελούν την ψηφιακή υπογραφή ως προς το ον. Μέσα από αυτή την ψηφιακή υπογραφή, είναι δυνατή η περαιτέρω χρήση του όντος μέσα από τη μετα-επεξεργασία σε δεύτερο χρόνο ή και ασύγχρονα από την ίδια την μονάδα λογικής επεξεργασίας του κυβερνητικού συστήματος (ή από κάποια άλλη εξωτερική ή εσωτερική επεξεργαστική μονάδα, όσο η μορφή κωδικοποίησης των δεδομένων είναι κοινή).

Στις περισσότερες περιπτώσεις τα δεδομένα αυτά αποθηκεύονται σε βάσεις δεδομένων ή καταστρέφονται μέσα από της ανατροφοδότηση του κυβερνητικού συστήματος, αποκρύπτοντας την ύπαρξη τους από τον τελικό χρήστη του προϊόντος και των διάφορων κοινωνικών οργανισμών. Όταν τα δεδομένα – χαρακτηριστικά ενός όντος παρέχονται στον τελικό χρήστη, για παράδειγμα μέσα από την ακρίβεια ενός οργάνου μέτρησης, αποτελούν απλά μια λειτουργική πληροφορία ή αναφορά προς το φυσικό προϊόν. Στις περιπτώσεις όμως που προσφέρεται ένα υπολογιστικό ψηφιακό περιβάλλον με το οποίο ο χρήστης μπορεί να διαδράσει, είναι εφικτή η μετάφραση των μεταδεδομένων και χαρακτηρισμών αυτών σε μια ανθρωποκεντρική μορφή και η γραφική αναπαράσταση του όντος προς τον τελικό χρήστη, ανακοινώνοντας την ύπαρξη της οντότητας και εδραιώνοντας την υπαρκτή ή μετα-υπαρκτή φύση του όντος στον άνθρωπο.

Η χρησιμότητα των αναγνωριστικών και των μεταδεδομένων από εξωγενή συστήματα, από το ίδιο το κυβερνητικό, καθορίζεται τόσο από την ύπαρξη ενός ενδιάμεσου δικτύου μεταφοράς ή συναλλαγής αυτών, όσο και από τις κωδικοποιημένες ικανότητες που κυβερνητικού αλλά και από το μέσο γραφικής αναπαράστασης και η φύση αυτού.

Οι βασικοί παράγοντες που καθορίζουν τις δυνατότητες και την λειτουργία του παραγόμενου όντος είναι:

- Τα δεδομένα απαιτούν την πρόσβαση του κυβερνητικού, που συλλέγει και πακετάρει της πληροφορίες, σε ασφαλές δίκτυο όπου παρέχεται ένα έμπιστο περιβάλλον στους μεμονωμένους χρήστες διασφαλίζοντας την σωστή διαχείριση των δεδομένων και επεξεργασία τους, ένα τέτοιο δίκτυο θα μπορούσε να είναι το Ethereum Network.

- Απαιτείται επεξεργαστική ισχύ για την εκτέλεση όλων των functions που θα παράξουν τα μεταδεδομένα και ένα περιβάλλον για την ασύγχρονη επεξεργασία των δεδομένων αυτών.
- Όταν είναι εφικτό και αναγκαίο, το κυβερνητικό σύστημα χρειάζεται πρόσβαση σε υπηρεσίες ή λειτουργίες που επιτρέπουν την μετα-ανάλυση του όντος από κάποιο κυβερνητικό, το οποίο εφαρμόζει μεθόδους ανάλυσης χρησιμοποιώντας νευρωνικά δίκτυα.
- Ένα έμπιστο runtime περιβάλλον για την εκτέλεση των διεργασιών χωρίς τον κίνδυνο κακόβουλης αλλοίωσης των δεδομένων.
- Η αναπαράσταση απαιτεί τόσο γραφική ισχύ όσο και ένα προκαθορισμένο συμβατό περιβάλλον για να πάρει μέρος.
- Η ικανότητα του όντος να μεταλλαχθεί με τον καιρό ή να μεταφέρει λειτουργικότητα από σύστημα σε σύστημα ανάλογα με τους χαρακτηρισμούς του.

Ανάμεσα στις χρήσεις των μεταδεδομένων που συλλέγονται, στις εφαρμογές αιχμής, μπορούν να χρησιμοποιηθούν σε κάποιο δεύτερο κυβερνο-φυσικό σύστημα (Cyber-Physical System (CPS)) ή να δημιουργήσουν ένα ψηφιακό δίδυμο (Digital twin). Παράλληλα με την ραγδαία άνοδο του Gamification, της τεχνολογίας Blockchain και του μεταπρογραμματισμού, οι παραπάνω εφαρμογές είναι όλο και πιο πιθανές πάνω σε σετ μεταδεδομένων από κυβερνητικά συστήματα.

Τα μεταδεδομένα, βέβαια, δεν χρειάζονται την περαιτέρω αναγωγή τους σε κάποια οντότητα για να είναι ον από μόνα τους. Διότι τα μεταδεδομένα, όταν εξετάζονται υπό το πρίσμα της επιστήμης δεδομένων και της μηχανικής μάθησης, μπορούν να θεωρηθούν ως οντότητες:

- Data-sets για Νευρωνικά Δίκτυα: Στη μηχανική μάθηση, ειδικά στην εκπαίδευση νευρωνικών δικτύων, τα δεδομένα είναι μια θεμελιώδης οντότητα. Τα μεταδεδομένα που συλλέγονται από κυβερνητικά συστήματα περιλαμβάνουν δεδομένα αισθητήρων, σήματα ελέγχου, καταστάσεις συστήματος, μετρικές απόδοσης και καταγραφές σφαλμάτων. Αυτά τα δεδομένα είναι χρήσιμα και σημαντικά για την εκπαίδευση μοντέλων για να αναγνωρίζουν πρότυπα, να κάνουν προβλέψεις και να βελτιστοποιούν την απόδοση του συστήματος.
- Δυναμικές Πληροφορίες: Τα μεταδεδομένα δεν είναι στατικά - εξελίσσονται καθώς το σύστημα λειτουργεί, παρέχοντας δυναμικές και πλούσιες σε περιεχόμενο πληροφορίες. Αυτή η συνεχής ροή δεδομένων τα καθιστά ως μια οντότητα που μπορεί να αναλυθεί και να αξιοποιηθεί για λήψη αποφάσεων σε πραγματικό χρόνο και προσαρμοστική μάθηση.

Για το παράδειγμα των νευρωνικών δικτύων, συγκεκριμένα, δεν γίνεται αναγωγή στα δεδομένα αλλά τα ίδια τα δεδομένα σαν περιεχόμενο και σαν ένα συσσωμάτωμα έχουν αξία – δηλαδή το ίδιο το σώμα των δεδομένων φέρει την αξία τους προς τα συστήματα. Έτσι τα ίδια τα δεδομένα δρουν σαν όντα μέσα από πλήθος τρόπων και έχοντας διάφορες χρήσεις [52][53][54]:

- **Εξαγωγή Χαρακτηριστικών (Feature):** Τα μεταδεδομένα παρέχουν χαρακτηριστικά (Features) που χρησιμοποιούνται ως είσοδοι για νευρωνικά δίκτυα. Αυτά τα χαρακτηριστικά βοηθούν στον εντοπισμό μοτίβων και συσχετισμών εντός των δεδομένων, τα οποία είναι απαραίτητα για την εκπαίδευση αποτελεσματικών μοντέλων.
- **Επισημάνσεις και σχολιασμοί:** Τα μεταδεδομένα συχνά περιλαμβάνουν επισημάνσεις και σχολιασμούς που χρησιμοποιούνται για την επίβλεψη των αλγορίθμων μάθησης. Για παράδειγμα, τα δεδομένα καταγραφής της απόδοσης και των σφαλμάτων μπορούν να καθορίσουν την επιτυχία ή την αποτυχία του συστήματος, βοηθώντας το νευρωνικό δίκτυο να μάθει να προβλέπει και να βελτιώνει την απόδοση του συστήματος.
- **Εκπαίδευση Μοντέλου:** Τα πλούσια δεδομένα που παρέχονται από τα συλλεγμένα μεταδεδομένα μπορούν να διαχωριστούν σε σετ εκπαίδευσης και επικύρωσης, επιτρέποντας στο νευρωνικό δίκτυο να μαθαίνει από τα προηγούμενα δεδομένα και να επικυρώνει τις προβλέψεις του σε νέα δεδομένα.

4.3 Τα κυβερνητικά μέσα στα βιντεοπαιχνίδια

Ακολούθως, εξετάζεται η συσχέτιση και οι επιρροές των συστημάτων των κυβερνητικών στην αρχιτεκτονική των βιντεοπαιχνιδιών, τα βιντεοπαιχνίδια που εμπεριέχουν κυβερνητικά και τα βιντεοπαιχνίδια που συνεργάζονται με κυβερνητικά συστήματα. Μέσα από την βιβλιογραφία [60] [64] [62] [57] και την ανάλυση των συστημάτων και διαδικασιών που δομούν ένα παιχνίδι, όπως και μέσα από την πολυπαραγοντική φύση τους, γίνεται προφανής η αλληλένδετη φύση και οι επιρροές των κυβερνητικών στον κλάδο των παιχνιδιών. Καθώς τα κυβερνητικά συστήματα των διεργασιών ανατροφοδότησης με τις διάφορες εφαρμογές τους, χρησιμοποιούνται και διαμορφώνουν τους μηχανισμούς του παιχνιδιού (gameplay), την τεχνητή νοημοσύνη, την αλληλεπίδραση των παικτών και τις αφηγηματικές δομές. Η επιστήμη της κυβερνητικής έχει βαθιά επιρροή στα βιντεοπαιχνίδια, επηρεάζει την εσωτερική αρχιτεκτονική του συστήματός τους και αποτελεί μέρος αυτών.

Εντός των ίδιων, των πλαισίων κατασκευής και σχεδίασης σύγχρονων βιντεοπαιχνιδιών για το metaverse, βρίσκουν χρήση οι περισσότερες θεωρίες και συστήματα της κυβερνητικής. Η χρήση της Θεωρίας Αυτομάτων και της Θεωρίας Συστημάτων επιτρέπει την μοντελοποίηση και ανάλυση σύνθετων συστημάτων, βρίσκοντας χρήση στην δημιουργία εικονικών περιουσιακών στοιχείων που μπορούν να αλληλεπιδρούν αυτόνομα και να ανταποκρίνονται στις εισροές (φυσικά inputs και δεδομένα) των χρηστών, μιμούμενα τη συμπεριφορά των οντοτήτων του πραγματικού κόσμου, στο metaverse. Η δυνατότητα ύπαρξης και χρήσης διαλειτουργικών (interoperable) μεταλλάξιμων (mutable) οντοτήτων στο metaverse γίνεται εφικτή μέσα από τα πρότυπα (standards) μεταδεδομένων. Αυτά, εξασφαλίζουν την ανεμπόδιση και ομαλή ενσωμάτωση διαφορετικών συστημάτων, αντικατοπτρίζοντας τις αρχές των κυβερνο-φυσικών συστημάτων (CPS). Μέσα από την ενοποίηση διαφόρων εικονικών και υπολογιστικών συνιστωσών (στα κυβερνο-φυσικά συστήματα - CPS), παίρνει μέρος η ενσωμάτωση φυσικών και υπολογιστικών στοιχείων σε ένα σύστημα. Αντίστοιχα, στο metaverse, εικονικά και υπολογιστικά στοιχεία συνεργάζονται και δημιουργούν ένα συμβιωτικό (και παράγωγα αυτού). Η παρακολούθηση και ο έλεγχος ψηφιακών περιουσιακών στοιχείων σε πραγματικό χρόνο στο metaverse χρησιμοποιούν τις αρχές του IoT για την παροχή άμεσης ανατροφοδότησης και βελτιστοποίησης από εξωτερικούς τρίτους κόμβους που ασύγχρονα εισάγονται στο σύστημα (όπως οι κόμβοι στο Ethereum Network ή οι υπηρεσίες που επαληθεύουν τις συναλλαγές ψάχνοντας για double spending).

Περαιτέρω, τα κυβερνητικά έχουν συνεισφέρει στην ανάπτυξη των Ψηφιακών διδύμων όπως και στην Προληπτική Συντήρηση (Predictive Maintenance) και την τεχνητή νοημοσύνη / μηχανική μάθηση, στοιχεία που χρησιμοποιούνται στα μοντέρνα βιντεοπαιχνίδια 9^{ης} γενιάς. Τα ψηφιακά δίδυμα είναι εικονικές αναπαραστάσεις φυσικών οντοτήτων (ψηφιακοί κλώνοι) που χρησιμοποιούν δεδομένα σε πραγματικό χρόνο για να κρατήσουν τα όντα σε παραλληλία ή να τρέξουν εξομοιώσεις. Για την διατήρηση της παραλληλίας τα ψηφιακά δίδυμα χρησιμοποιούν μηχανισμούς ανατροφοδότησης για να ενημερώνουν συνεχώς το εικονικό μοντέλο βάσει φυσικών μεταβλητών που διέπουν την πραγματική φυσική οντότητα. Μέσα από την προληπτική συντήρηση (PdM Predictive Maintenance) ένα ψηφιακό οικοσύστημα ή ένα παιχνίδι μπορεί να προβλέπει μελλοντικά προβλήματα σε πραγματικό χρόνο μέσα από αλγόριθμους μηχανικής μάθησης και να αναδιοργανώνεται / προσαρμόζεται. Αυτά τα προβλήματα μπορεί να αφορούν τον όγκο εργασίας ενός server πριν την είσοδο μεγάλου αριθμού παικτών σε κάποιο παιχνίδι, σε

αυτήν την περίπτωση το σύστημα καλείται να προβλέψει αυτή την ανωμαλία και να ανοίξει περισσότερα server instances. [55] [56] [57] [58] [59]

Κατά την σμίξη των κυβερνητικών με τα βιντεοπαιχνίδια, ένα βιντεοπαιχνίδι μπορεί να εμπεριέχει ένα κυβερνητικό σύστημα ή απλά να «μιλάει» με ένα εξωτερικό κυβερνητικό σύστημα μέσα από δίκτυα TDP/UDP. Ένα παράδειγμα βιντεοπαιχνιδιού που χρησιμοποιεί τα μεταδεδομένα από κυβερνητικά συστήματα είναι το Microsoft Flight Simulator, το οποίο εμπεριέχει μια ρεαλιστική γραφική αναπαράσταση της γης από δεδομένα δορυφόρων και Lidar 3D Scanning. Εκτός από τα ρεαλιστικά τρισδιάστατα μοντέλα, το παιχνίδι χρησιμοποιεί δεδομένα από τρίτους παρόχους για την παρακολούθηση και εξομοίωση του καιρού σε πραγματικό χρόνο ανά γεωγραφική περιοχή και την ένταξη πραγματικών πτήσεων καθώς συμβαίνουν σε πραγματικό χρόνο, μέσα στον χάρτη. Σε αυτό το παράδειγμα όμως η επικοινωνία δεν είναι αμφίδρομη και το παιχνίδι μόνο λαμβάνει τα δεδομένα και τα εφαρμόζει.

Στο παιχνίδι Ingress της εταιρίας Niantic, μια πρώιμη έκδοση του Pokémon Go που επίσης φτιάχτηκε από την ίδια εταιρία, το παιχνίδι δρα ολόκληρο σαν ένα κυβερνητικό – χρησιμοποιώντας τον χάρτη, τις διαδράσεις των παικτών με τα διάφορα συστήματα που εφαρμόζει και την γεωτοποθεσία του παίκτη σαν τον βασικό πυλώνα μορφοποίησης του παιχνιδιού. Το Ingress αξιοποιεί γεωγραφικά δεδομένα από τον πραγματικό κόσμο και προηγμένους αλγόριθμους για να δημιουργήσει ένα δυναμικό περιβάλλον παιχνιδιού που βασισμένο στον φυσικό κόσμο, λειτουργώντας ταυτόχρονα σαν ένα ψηφιακό overlay. Οι παίκτες αλληλεπιδρούν με το παιχνίδι χρησιμοποιώντας τις κινητές συσκευές τους, οι οποίες λειτουργούν ως κυβερνητικές διεπαφές (interfaces), συνδυάζοντας αρμονικά τα εικονικά στοιχεία με τις πραγματικές τοποθεσίες. Το κυβερνητικό σύστημα του παιχνιδιού επεξεργάζεται συνεχώς δεδομένα από τους παίκτες, όπως οι κινήσεις τους και οι αλληλεπιδράσεις τους με εικονικές πύλες, και παρέχει ανάδραση σε πραγματικό χρόνο, δημιουργώντας μια άμεση και προσαρμοστική εμπειρία παιχνιδιού. Αυτός ο κύκλος ανάδρασης, χαρακτηριστικός των κυβερνητικών συστημάτων, διασφαλίζει ότι το περιβάλλον του παιχνιδιού εξελίσσεται βάσει των ενεργειών των παικτών, διατηρώντας μια κατάσταση ισορροπίας και ενδιαφέροντος. Η χρήση της τεχνολογίας γεωεντοπισμού, σε συνδυασμό με την επεξεργασία δεδομένων δικτύου, επιτρέπει στο Ingress να διευκολύνει σύνθετες αποστολές βάσει τοποθεσίας και κάνει εφικτές τις κοινοτικές αλληλεπιδράσεις, καθιστώντας το ένα πρωταρχικό παράδειγμα κυβερνητικής ενσωμάτωσης στα παιχνίδια επαυξημένης πραγματικότητας (AR). [60] [62] [64] [57]

4.4 Συνδέοντας τα Δεδομένα των Κυβερνητικών Συστημάτων με Εικονικές Οντότητες στα Βιντεοπαιχνίδια και ο Ρόλος των DAO και των δικτύων Blockchain

Για τις εφαρμογές όπου ο φυσικός χώρος είναι αδιάρρηκτα συνδεδεμένος με τον ψηφιακό μέσα από τα κυβερνητικά συστήματα, τα βιντεοπαιχνίδια, καθώς είναι μια πρόιμη μορφή ενός μελλοντικού εικονικού μεταχώρου, αποτελούν ένα άρτιο σύστημα απεικόνισης και επέκτασης της λειτουργικότητας των μεταδεδομένων που συλλέγονται από κυβερνητικά συστήματα. Αντίστροφα η ύπαρξη αυτών των μεγάλων σετ πληροφοριών προσφέρει πολύτιμη πληροφορία που θα μπορούσαν να χρησιμοποιήσουν τα τωρινά βιντεοπαιχνίδια, καθώς η διασταύρωση των κυβερνητικών συστημάτων και των βιντεοπαιχνιδιών έχει δημιουργήσει μια επαναστατική προσέγγιση στο σχεδιασμό παιχνιδιών και την αλληλεπίδραση των παικτών. Τα κυβερνητικά συστήματα, με τους βρόχους ανατροφοδότησης και τις αυτορυθμιζόμενες ικανότητές τους, παράγουν τεράστιες ποσότητες μεταδεδομένων που μπορούν να αξιοποιηθούν για τη δημιουργία δυναμικών και εμπυθιστικών εικονικών οντοτήτων μέσα στα βιντεοπαιχνίδια. Ακολουθως, εξετάζεται η χρήση των μεταδεδομένων τόσο για την απεικόνιση του όντος ψηφιακά όπως και για την ενίσχυση των περιβαλλόντων των βιντεοπαιχνιδιών, τη βελτίωση των εμπειριών των παικτών και την προώθηση καινοτόμων μηχανισμών παιχνιδιού.

Τα δεδομένα που παράγονται από τα κυβερνητικά συστήματα για τα βιντεοπαιχνίδια είναι πολυδιάστατα, περιλαμβάνοντας ακατέργαστα δεδομένα αισθητήρων, σήματα ελέγχου, καταστάσεις συστημάτων, μετρήσεις απόδοσης και αρχεία ασφαμάτων. Αυτά τα μεταδεδομένα είναι ανεκτίμητα, παρέχοντας μια συνολική εικόνα των λειτουργιών του συστήματος και των αλληλεπιδράσεων του με το περιβάλλον. Οι εικονικές οντότητες στα βιντεοπαιχνίδια, όπως οι υπολογιστικοί εικονικοί χαρακτήρες (μη-παίκτης χαρακτήρες ή NPCs), τα avatars και τα δυναμικά γραφικά στοιχεία (Procedural και generative οντότητες), μπορούν να ενισχυθούν σημαντικά μέσω των πλούσιων μεταδεδομένων που παράγονται από τα κυβερνητικά συστήματα.

Μέσω της χρήσης των δεδομένων αυτών, είναι εφικτή η προσαρμογή της εμπειρίας του παιχνιδιού στα ατομικά χαρακτηριστικά του χρήστη. Αναλύοντας δεδομένα παικτών, τα παιχνίδια μπορούν να προσαρμόσουν τα επίπεδα δυσκολίας, να παρέχουν εξατομικευμένες προκλήσεις και να προσφέρουν περιεχόμενο που ευθυγραμμίζεται με τις προτιμήσεις και τις

ικανότητες των παικτών[66]. Οι NPCs που χρησιμοποιούν κυβερνητικά δεδομένα μπορούν να επιδείξουν πιο ρεαλιστικές και ενδιαφέρουσες συμπεριφορές. Αυτοί οι NPCs μπορούν να προσαρμοστούν στις ενέργειες των παικτών, να μάθουν από τις αλληλεπιδράσεις και να παρέχουν πιο ουσιαστικές και απρόβλεπτες συναντήσεις[63]. Τα παιχνίδια μπορούν να χρησιμοποιούν κυβερνητικά δεδομένα για να δημιουργούν κόσμους που αντιδρούν και αλλάζουν βάσει των ενεργειών των παικτών. Αυτό μπορεί να περιλαμβάνει την αλλαγή του περιβάλλοντος, τη δυναμική δημιουργία γεγονότων και τη διασφάλιση ότι κάθε παιχνίδι προσφέρει μια μοναδική εμπειρία [68]. Η ενσωμάτωση προσαρμοστικών εικονικών οντοτήτων μπορεί να ενισχύσει σημαντικά την εμπύθιση του παίκτη. Οι ρεαλιστικοί NPCs, τα αντιδραστικά περιβάλλοντα παιχνιδιών και το εξατομικευμένο περιεχόμενο δημιουργούν μια βαθύτερη σύνδεση μεταξύ του παίκτη και του κόσμου του παιχνιδιού [67].

Προκλήσεις και Προβληματισμοί

Όμως η ενσωμάτωση δεδομένων από κυβερνητικά συστήματα για τη δημιουργία εικονικών οντοτήτων στα βιντεοπαιχνίδια παρουσιάζει μεγάλες δυνατότητες, αλλά και σημαντικές προκλήσεις. Μερικές από αυτές τις προκλήσεις είναι:

Ιδιωτικότητα και Ασφάλεια Δεδομένων

Η συλλογή και χρήση εκτενών μεταδεδομένων εγείρει ανησυχίες σχετικά με την ιδιωτικότητα και την ασφάλεια των δεδομένων. Η διασφάλιση ότι οι ευαίσθητες πληροφορίες προστατεύονται και χρησιμοποιούνται με ηθικό τρόπο είναι υψίστης σημασίας.

Διαλειτουργικότητα

Η ενσωμάτωση δεδομένων από διαφορετικά κυβερνητικά συστήματα απαιτεί κοινά πρότυπα διαλειτουργικότητας για να εξασφαλιστεί η απροβλημάτιστη επικοινωνία και ανταλλαγή δεδομένων.

Κλιμάκωση

Καθώς ο όγκος των δεδομένων και η πολυπλοκότητα των εικονικών οντοτήτων αυξάνεται, είναι σημαντικό να εξασφαλιστεί ότι τα συστήματα μπορούν να κλιμακωθούν αποδοτικά.

Ακρίβεια και Αξιοπιστία

Η εξασφάλιση της ακρίβειας και της αξιοπιστίας των δεδομένων και των παραγόμενων εξ αυτών εικονικών οντοτήτων είναι απαραίτητη για την αξιοπιστία του συστήματος και την εικόνα του προς τους εταίρους και τους χρήστες καθώς και την απόδοση του συστήματος.

Οι προαναφερόμενοι προβληματισμοί μπορούν να λυθούν μέσα από την χρήση αποκεντρωμένων αυτόνομων οργανισμών (DAOs) και δικτύων blockchain, αφού παρέχουν ένα ισχυρό πλαίσιο λειτουργιών για την αντιμετώπιση αυτών των προκλήσεων. Τα DAOs και τα δίκτυα blockchain μπορούν να υποστηρίξουν την ανάπτυξη και διαχείριση εικονικών οντοτήτων σε βιντεοπαιχνίδια, εξασφαλίζοντας διαφάνεια, ασφάλεια και διακυβέρνηση που καθοδηγείται από την κοινότητα.

Η τεχνολογία Blockchain, με την αποκεντρωμένη και αμετάβλητη φύση της, προσφέρει αρκετά πλεονεκτήματα για τη διαχείριση δεδομένων και αλληλεπιδράσεων στα βιντεοπαιχνίδια.

Αξιοποιώντας το blockchain, οι προγραμματιστές παιχνιδιών μπορούν να διασφαλίσουν ότι τα δεδομένα που συλλέγονται από τα κυβερνητικά συστήματα είναι ασφαλή, διαφανή και προστατευμένα από αλλοιώσεις.

Το Blockchain διασφαλίζει ότι τα δεδομένα που παράγονται από τα κυβερνητικά συστήματα αποθηκεύονται σε ένα αμετάβλητο καθολικό. Αυτό αποτρέπει την αλλοίωση και παρέχει μια διαφανή ιστορία όλων των συναλλαγών δεδομένων. Μόλις καταγραφούν τα δεδομένα στο blockchain, δεν μπορούν να τροποποιηθούν ή να διαγραφούν, εξασφαλίζοντας την ακεραιότητά τους. Το Blockchain χρησιμοποιεί κρυπτογραφικές τεχνικές για την ασφάλεια των δεδομένων, καθιστώντας τα ανθεκτικά σε μη εξουσιοδοτημένη πρόσβαση και κυβερνοεπιθέσεις [69]. Όλες οι συναλλαγές σε ένα blockchain δίκτυο είναι ορατές σε όλους τους συμμετέχοντες στο δίκτυο. Αυτή η διαφάνεια εξασφαλίζει ότι όλοι οι ενδιαφερόμενοι μπορούν να επαληθεύσουν την αυθεντικότητα και την ακεραιότητα των δεδομένων μέσα από το δημόσιο καθολικό, προωθώντας την εμπιστοσύνη και την υπευθυνότητα των μελών του δικτύου [71]. Τα έξυπνα συμβόλαια μπορούν να αυτοματοποιήσουν τους μηχανισμούς του παιχνιδιού, όπως οι συναλλαγές εντός του παιχνιδιού, εξασφαλίζοντας ότι εκτελούνται δίκαια και διαφανώς. Τέλος,

οι συμμετέχοντες μπορούν να αλληλεπιδρούν απευθείας, χωρίς την ανάγκη ενός κεντροποιημένου μεσάζοντα, μειώνοντας τον κίνδυνο απάτης [70].

Τα DAO (Αποκεντρωμένοι Αυτόνομοι Οργανισμοί) είναι οργανισμοί που διοικούνται από έξυπνα συμβόλαια στο blockchain, όπου οι αποφάσεις λαμβάνονται από την κοινότητα μέσω ενός μηχανισμού συναίνεσης, αυτό προσφέρει εμπιστοσύνη και κοινή αποδοχή στο σύστημα. Οι DAO προσφέρουν μια αποκεντρωμένη προσέγγιση στην ανάπτυξη και διακυβέρνηση παιχνιδιών, επιτρέποντας την κοινοτική λήψη αποφάσεων και την κατανομή πόρων. Τα DAO δίνουν τη δυνατότητα στους παίκτες και τους ενδιαφερόμενους να συμμετέχουν στη διαδικασία λήψης αποφάσεων. Αυτή η δημοκρατική και αποκεντροποιημένη προσέγγιση προσφέρει ένα ασφαλές περιβάλλον για επιχειρήσεις και μεμονωμένους χρήστες. Οι ενδιαφερόμενοι μπορούν να ψηφίζουν προτάσεις, επηρεάζοντας την ανάπτυξη του παιχνιδιού, τις ενημερώσεις και τις πολιτικές διακυβέρνησης, μπορούν να προτείνουν έργα ή χαρακτηριστικά, και το DAO μπορεί να κατανέμει τα κεφάλαια ανάλογα.

Εν κατακλείδι, συνδυάζοντας δεδομένα από κυβερνητικά συστήματα με την τεχνολογία blockchain και τα DAO, δημιουργείται ένα ισχυρό οικοσύστημα για τα βιντεοπαιχνίδια, ενισχύοντας την ασφάλεια, τη διαφάνεια και τη συμμετοχή της κοινότητας. Ταυτόχρονα, είναι εφικτή η επανεισαγωγή της πληροφορίας πίσω στο κυβερνητικό σύστημα για διάδραση.

Μέσα από όλα αυτά τα συστήματα που προαναφέρθηκαν, παρουσιάζονται πολλές δυνατότητες μετ-επεξεργασίας των δεδομένων και δυνατότητες για ασύγχρονες αυτόνομες εργασίες να παίρνουν μέρος σαν λειτουργίες μεταπρογραμματισμού. Οι εικονικές οντότητες στα βιντεοπαιχνίδια μπορούν να λειτουργούν με AI μοντέλα εκπαιδευμένα σε δεδομένα από κυβερνητικά συστήματα, με το blockchain να εξασφαλίζει την ακεραιότητα και την προέλευση των δεδομένων. Οι εικονικές οντότητες μπορούν να προσαρμόζονται και να εξελίσσονται με real-time δεδομένα (πραγματικού χρόνου), ασύγχρονα, με τις αλλαγές να καταγράφονται στο blockchain για διαφάνεια. Οι παίκτες μπορούν να ανταλλάσσουν περιουσιακά στοιχεία και να συμμετέχουν σε οικονομικές δραστηριότητες, μπορούν να ανταλλάσσουν περιουσιακά στοιχεία (inventory items), εντός του παιχνιδιού, σε ψηφιακές αγορές με τεχνολογία blockchain, εξασφαλίζοντας δίκαιες και ασφαλείς συναλλαγές. Οι παίκτες μπορούν να προτείνουν νέες λειτουργίες ή βελτιώσεις, και η κοινότητα μπορεί να ψηφίσει για την εφαρμογή τους.

Κεφάλαιο 5 Βιομηχανία 4.0 και Βιντεοπαιχνίδια

5.1 Βιομηχανία 4.0

5.1.1 Ιστορικό Πλαίσιο της Βιομηχανοποίησης και της Κατασκευής

Η εξέλιξη των βιομηχανικών διαδικασιών και της βιομηχανίας παραγωγής διαδραματίζει κεντρικό ρόλο στον σχηματισμό του σύγχρονου κόσμου και την μορφοποίηση της σύγχρονης δυτικής κουλτούρας. Αυτή η περίοδος, που είναι γνωστή ως βιομηχανοποίηση ή εκβιομηχάνιση, άρχισε με την πρώτη Βιομηχανική Επανάσταση στα μισά προς τέλη του 18ου αιώνα, εισάγοντας τις μηχανοκίνητες εγκαταστάσεις παραγωγής στα εργοστάσια, μέσα από την χρήση της ατμομηχανής. Αυτό σηματοδότησε την Βιομηχανία 1.0, όπου η χειρωνακτική εργασία μειώθηκε σημαντικά μέσα από την μηχανοποίηση. Η επόμενη φάση ανάπτυξης, η Βιομηχανία 2.0, ξεκίνησε στις αρχές του 20ού αιώνα με την εμφάνιση της ηλεκτρικής ενέργειας και την ένταξη των γραμμών παραγωγής σε ηλεκτροδοτημένα εργοστάσια, ενισχύοντας περαιτέρω την άμεση αποδοτικότητα των εργοστασίων και κάνοντας δυνατή την αύξηση της κλίμακας της παραγωγής. Η τρίτη επανάσταση, ή Βιομηχανία 3.0, πήρε μέρος στα τέλη του 20ού αιώνα, χαρακτηριζόμενη από την ένταξη των υπολογιστών και της αυτοματοποίησης στις διαδικασίες των εργοστασίων. Η συστηματική τεχνολογική πρόοδος και ο βαρύς αυτοματισμός στα εργοστάσια σηματοδότησε την έναρξη της Βιομηχανίας 4.0, η οποία συνθέτει στοιχεία και συστήματα υπάρχουσας τεχνολογίας, μέσω της ψηφιοποίησης και των νέων μεθόδων δικτύωσης, επεξεργασίας και σύνδεσης λογικών μονάδων επεξεργασίας. [45]

5.1.2 Οφέλη της Βιομηχανοποίησης για την Κοινωνία και την Οικονομία

Η βαριά βιομηχανοποίηση έχει φέρει πολλά οφέλη στην κοινωνία και στην οικονομία. Οικονομικά, έχει ενεργοποιήσει τη μαζική παραγωγή, η οποία μειώνει τα κόστη και καθιστά τα προϊόντα πιο προσβάσιμα σε ένα μεγαλύτερο εύρος του πληθυσμού. Αυτή η εύκολη πρόσβαση σε αγαθά έχει συμβάλει σημαντικά στη βελτίωση των βιοτικών συνθηκών και στην τόνωση των καταναλωτικών αγορών. Κοινωνικά, η βιομηχανοποίηση έχει δημιουργήσει νέους τομείς εργασίας και ευκαιριών, ενθαρρύνοντας μια μετατόπιση από αγροτικές οικονομίες σε αστικοποιημένες, βιομηχανικές κοινωνίες. Επιπλέον, η βιομηχανοποίηση έχει προωθήσει καινοτομίες στην τεχνολογία και τις υποδομές, διευκολύνοντας το εμπόριο σε παγκόσμια

κλίμακα και τις μεταφορές (logistics), που με τη σειρά τους ενισχύουν την οικονομική ανάπτυξη σε παγκόσμιο επίπεδο.

5.1.3 Κύρια Στοιχεία και Τεχνολογίες

Αυτή η μετάβαση από τη Βιομηχανία 1.0 στη 3.0 έγινε εφικτή μέσα από μια σειρά τεχνολογικών, οικονομικών και κοινωνικών αλλαγών, οι οποίες βοήθησαν την τεχνολογική ανάπτυξη και πρόοδο. Με την σειρά της, η Βιομηχανία 4.0 αντιπροσωπεύει μια σημαντική μεταμόρφωση και χαρακτηρίζεται από τη σύγκλιση των ψηφιακών τεχνολογιών με τις παραδοσιακές βιομηχανικές πρακτικές. Αυτή η νέα Βιομηχανική επανάσταση εκμεταλλεύεται το Διαδίκτυο των Πραγμάτων (IoT), την τεχνητή νοημοσύνη (AI), τα μεγάλα δεδομένα και άλλες ανερχόμενες τεχνολογίες όπως τα δίκτυα Blockchain, για να δημιουργήσει «έξυπνα εργοστάσια» με ολοκληρωμένες και αυτοματοποιημένες διαδικασίες. [45]

Η Βιομηχανία 4.0 διακρίνεται από την χρήση τεχνολογιών αιχμής που συνδέουν τον φυσικό και ψηφιακό κόσμο. Τα κύρια στοιχεία περιλαμβάνουν IoT (Internet of Things – Δίκτυο των Αντικειμένων), AI τεχνητή νοημοσύνη, μηχανική μάθηση, ρομποτική και κυβερνο-φυσικά συστήματα, μεταξύ άλλων. Αυτές οι τεχνολογίες επιτρέπουν τη συλλογή και ανάλυση μεγάλων σετ δεδομένων, τα οποία χρησιμοποιούνται για την τελειοποίηση της διαδικασίας της παραγωγής, προωθώντας την αποδοτικότητα και τη βελτιστοποίηση μέσω προληπτικής συντήρησης (predictive maintenance), ανεπτυγμένης διαχείρισης της αλυσίδας εφοδιασμού και αποδοτικότερων διαδικασιών παραγωγής. Η διαλειτουργικότητα (interoperability) μηχανημάτων και συστημάτων που διευκολύνεται από τις συσκευές IoT και τους αλγόριθμους AI επιτρέπει την δυνατότητα δημιουργίας πιο ευέλικτων και προσαρμόσιμων γραμμών παραγωγής. Επιπλέον, η επαυξημένη πραγματικότητα (AR) και η εικονική πραγματικότητα (VR) χρησιμοποιούνται για να βελτιώσουν τις διαδικασίες σχεδίασης και συντήρησης, μικραίνοντας την απόσταση ανάμεσα στις ψηφιακές προσομοιώσεις του εικονικού χώρου και τον φυσικό χώρο. [61]

5.1.4 Η Αναγκαιότητα της Βιομηχανίας 4.0 για την Κοινωνία

Η αναγκαιότητα της Βιομηχανίας 4.0 για την κοινωνία είναι πολλαπλή. Η ταχεία αύξηση του πληθυσμού και οι περιορισμένοι φυσικοί πόροι δημιουργούν την ανάγκη κατασκευής βιώσιμων και αποδοτικών συστημάτων παραγωγής τα οποία μπορούν να προσαρμόζονται σε ασταθείς συνθήκες. Η Βιομηχανία 4.0 αντιμετωπίζει αυτές τις προκλήσεις επιτρέποντας την πιο αποδοτική χρήση των πόρων, μειώνοντας τα απόβλητα και ελαχιστοποιώντας το περιβαλλοντικό

αποτύπωμα μέσα από την χρήση της τεχνολογίας υλικών και τα «έξυπνα εργοστάσια». Επιπλέον, καθώς οι παγκόσμιες αγορές γίνονται όλο και πιο ανταγωνιστικές, η ικανότητα μίας γραμμής παραγωγής ως προς την γρήγορη προσαρμογή στις μεταβαλλόμενες απαιτήσεις των καταναλωτών, είναι σημαντική. Η Βιομηχανία 4.0 διευκολύνει αυτήν την προσαρμοστική διαδικασία, επιτρέποντας την ταχεία αναπροσαρμογή των στοιχείων μίας γραμμής παραγωγής, ενισχύοντας την ταχύτητα των κύκλων παραγωγής, κάνοντας αγαθά και νέες τεχνολογίες προσβάσιμα στον καταναλωτή ταχύτερα. Επιπλέον, τα βελτιστοποιημένα και «έξυπνα» συστήματα ασφαλείας, που φέρνουν οι αυτοματοποιημένες και έξυπνες γραμμές παραγωγής, συνεισφέρουν σημαντικά στη μείωση των εργατικών ατυχημάτων, βελτιώνοντας έτσι την ποιότητα ζωής των εργαζομένων στα εργοστάσια. [9]

5.1.5 Η Συνεισφορά της Βιομηχανίας 4.0 στην Οικονομία

Η Βιομηχανία 4.0 φέρει βαθιές αλλαγές στην οικονομία και τα οικονομικά συστήματα. Προωθεί ένα νέο επίπεδο παραγωγικότητας και βιομηχανικής αποδοτικότητας, ανοίγοντας νέα επιχειρηματικά μοντέλα και οικονομικές ευκαιρίες. Τα έξυπνα εργοστάσια, που χαρακτηρίζονται από τις δυνατότητες αυτοπαρακολούθησης και αυτόνομης λήψης αποφάσεων, μειώνουν δραματικά το χρόνο αδράνειας και ενισχύουν την συνολική ποιότητα των προϊόντων, οδηγώντας σε αυξημένη οικονομική παραγωγή και ανταγωνιστικότητα σε παγκόσμιο επίπεδο, από τις οντότητες που εντοιχίζουν τις τεχνολογίες της Βιομηχανίας 4.0. Επιπλέον, η Βιομηχανία 4.0 διευκολύνει τη μετάβαση σε επιχειρηματικά μοντέλα που βασίζονται στις υπηρεσίες πέρα από τα παραδοσιακά προϊόντα, αυτό περιλαμβάνει υπηρεσίες όπως η παραγωγή κατά παραγγελία, που μπορούν να ανοίξουν νέες πηγές εσόδων. Τέλος, η αυξημένη συνδεσιμότητα, η βαριά ψηφιοποίηση των συστημάτων και η αποδοτικότητα που παρέχουν οι τεχνολογίες της Βιομηχανίας 4.0 επιτρέπουν την εκτέλεση γρηγορότερων, ασφαλέστερων και πιο ομαλών συναλλαγών μέσα από την τεχνολογία Blockchain, ενισχύοντας τις οικονομικές δραστηριότητες διασυνοριακά. [9]

5.2 Τα συστήματα της Βιομηχανίας 4.0

Στο πλαίσιο της Βιομηχανίας 4.0, ένα μεγάλο εύρος τεχνολογιών διαδραματίζει βασικούς λειτουργικούς ρόλους για την βελτίωση των διαδικασιών παραγωγής και την διευκόλυνση της ψηφιακής μετάβασης [45]. Αυτές οι τεχνολογίες όχι μόνο ενισχύουν την παραγωγή αλλά προσφέρουν επίσης διεπιστημονικές εφαρμογές ανάμεσα σε πολλαπλούς κλάδους, ιδιαίτερα στις βιομηχανίες των παιχνιδιών και του Blockchain/NFT. Ακολουθώς, παρουσιάζονται ορισμένες βασικές τεχνολογίες και κάποια παραδείγματα εφαρμογής τους:

Ρομποτική και Αυτοματισμός

Η ρομποτική και ο αυτοματισμός είναι καίρια για τη βελτίωση της ακρίβειας, της ταχύτητας και της ασφάλειας των βιομηχανικών λειτουργιών. Παράλληλα εκτιμάται πως θα διαδραματίζουν σημαντικούς ρόλους στην κοινωνία του αύριο, με εφαρμογές όπως την αυτοματοποίηση των σπιτιών και την υγειονομική φροντίδα με ρομπότ που παρέχουν υπηρεσίες φροντίδας σε ηλικιωμένους ή άτομα με ειδικές ανάγκες [75].

Αυτές οι τεχνολογίες διασυνδέονται όλο και περισσότερο με τη βιομηχανία των παιχνιδιών, όπως μέσα από τις διαδικασίες ανάπτυξης περίπλοκων συστημάτων ελέγχου και διεπαφών που θα μπορούσαν επίσης να χρησιμοποιηθούν σε περιβάλλοντα παιχνιδιού. Επίσης, ένα άλλο παράδειγμα είναι η ενσωμάτωση ρομποτικών συστημάτων σε controllers (χειριστήρια – HID), τα οποία παρέχουν απτική ανταπόκριση για να ενισχύσουν τον ρεαλισμό της εμπειρίας του παιχνιδιού.

Για τις βιομηχανίες Blockchain και NFT, τεχνολογία από τον κλάδο των αυτοματοποιημένων συστημάτων μπορεί να χρησιμοποιηθεί για την απλοποίηση της διαδικασίας δημιουργίας των NFT και να διαχειρίζονται τις υψηλού όγκου συναλλαγές που είναι χαρακτηριστικές των αγορών NFT, διασφαλίζοντας την αποδοτικότητα και την αξιοπιστία. Παράλληλα, η ασφάλεια που προσφέρει η τεχνολογία Blockchain και η τεχνολογία κατακεμημένων συστημάτων είναι χρήσιμη σε σμήνος ρομπότ. [72].

Προσθετική Κατασκευή (3D Εκτύπωση / 3D Printing)

Τεχνολογία που επιτρέπει την ταχεία παραγωγή προσαρμοσμένων αγαθών απευθείας από ψηφιακά σχέδια, κάνοντας εφικτή την κατασκευή προσωποποιημένων αγαθών και μειώνοντας τα παραγόμενα απόβλητα κατά την διαδικασία παραγωγής [76]. Αυτή η τεχνολογία

χρησιμοποιείται για την κατασκευή ιατρικών εμφυτευμάτων και προσθετικών, παρέχοντας προσιτές και εξατομικευμένες υπηρεσίες σε ασθενείς.

Στη βιομηχανία των παιχνιδιών, η 3D εκτύπωση χρησιμοποιείται για τη δημιουργία προσωποποιημένων αγαλματιδίων μέσα από στοιχεία παιχνιδιών και άλλων συλλεκτικών αντικειμένων που συσχετίζονται με τον κόσμο του παιχνιδιού και την ιστορία, προσφέροντας στους παίκτες τη δυνατότητα να φέρουν κομμάτια του ψηφιακού κόσμου στην πραγματικότητα.

Για τη βιομηχανία Blockchain και NFT, η 3D εκτύπωση προσφέρει έναν τρόπο για την παραγωγή φυσικών αντιπροσωπεύσεων ψηφιακών περιουσιακών στοιχείων, ενισχύοντας την απτή αξία των NFTs και δίνοντας τους φυσική μορφή.

Συστήματα Μηχανικής Όρασης

Αυτά τα συστήματα χρησιμοποιούν κάμερες και τεχνολογία αναγνώρισης εικόνας για τη βελτίωση του ελέγχου ποιότητας στις γραμμές παραγωγής. Στα πλαίσια της καθημερινής ζωής αυτά τα συστήματα αναλύουν και αξιολογούν το περιβάλλον για την ασφάλεια των αυτοκινητόδρομων και την παρακολούθηση της κίνησης [82].

Στα παιχνίδια, παρόμοιες τεχνολογίες χρησιμοποιούνται για την καταγραφή κινήσεων και για τη βελτίωση των διεπαφών χρηστών, ιδίως σε περιβάλλοντα επαυξημένης (Augmented Reality) και εικονικής πραγματικότητας (Virtual Reality).

Στον τομέα του Blockchain, τα προηγμένα οπτικά συστήματα που χρησιμοποιούν μηχανική όραση, μπορούν να ενισχύσουν τα μέτρα ασφαλείας παρέχοντας ασφαλείς λύσεις επαλήθευσης ταυτότητας σε συναλλαγές.

Επαυξημένη Πραγματικότητα (AR) και Εικονική Πραγματικότητα (VR)

Τα συστήματα AR και VR εμπλουτίζουν τις εκπαιδευτικές διαδικασίες και ενισχύουν τις λειτουργικές διαδικασίες, προσφέροντας σημαντικές πληροφορίες και αυξάνοντας τον βαθμό διάδρασης με τα ψηφιακά συστήματα [61].

Αυτές οι τεχνολογίες έχουν ισχυρή παρουσία στη βιομηχανία των παιχνιδιών, παρέχοντας πλούσια, εμπυθιστικά περιβάλλοντα που εμπλέκουν τους χρήστες και τον φυσικό χώρο με τον

κόσμο του παιχνιδιού. Ταυτόχρονα επιτρέπουν την παιχνιδοποίηση (gamification) διαφόρων κλάδων, όπως και επιτρέπουν την δημιουργία παιχνιδιών νέου τύπου, όπως το Beat Saber, που συνδυάζει φυσική δραστηριότητα και ρυθμό, προσφέροντας μια εντελώς νέα μορφή διασκέδασης.

Οι τομείς Blockchain και NFT αξιοποιούν την AR και την VR τεχνολογία για να προσφέρουν εικονικές γκαλερί και εμπειρίες που παρουσιάζουν ψηφιακά περιουσιακά στοιχεία και NFTs με πιο αλληλεπιδραστικούς τρόπους. Παράλληλα τα NFT μπορούν να δρουν σαν στοιχεία παιχνιδιών VR ή να ξεκλειδώνουν λειτουργίες σε χώρο AR.

Αισθητήρες και Ενεργοποιητές (Actuators)

Οι αισθητήρες είναι συσκευές που ανιχνεύουν και αποκρίνονται σε κάποιο είδος εισόδου από το φυσικό περιβάλλον, όπως η θερμοκρασία, η φωτεινότητα, ή η πίεση. Οι ενεργοποιητές, από την άλλη, είναι συσκευές που μετατρέπουν την ενέργεια σε κίνηση, επιτρέποντας τον έλεγχο ενός μηχανικού συστήματος. Μαζί, οι αισθητήρες και οι ενεργοποιητές αποτελούν βασικά στοιχεία σε αυτοματοποιημένα συστήματα και μηχανισμούς, υποστηρίζοντας την αλληλεπίδραση μεταξύ του ψηφιακού και του φυσικού κόσμου. Ένα παράδειγμα χρήσης τους, είναι τα συστήματα πυρασφάλειας, που ελέγχουν μέσα από αισθητήρια για φωτιές και ενεργοποιούν συστήματα κατάσβεσης της φωτιάς.

Η βιομηχανία των παιχνιδιών χρησιμοποιεί παρόμοιες τεχνολογίες αισθητήρων σε χειριστήρια με επιταχυνσιόμετρα και συστήματα, που προσαρμόζουν το περιβάλλον του παιχνιδιού βάσει των ενεργειών του παίκτη.

Οι εφαρμογές Blockchain περιλαμβάνουν τη χρήση δεδομένων αισθητήρων για να ενεργοποιήσουν αυτόματα έξυπνα συμβόλαια, ιδίως σε χρήσεις αλυσίδας εφοδιασμού. Ταυτόχρονα, τα μεταδεδομένα ενός νομίσματος μπορούν να εμπεριέχουν τιμές από αισθητήρια όργανα.

Βιομηχανικό Διαδίκτυο των Πραγμάτων και Διαδίκτυο των Πραγμάτων (ΠοΤ και ΙοΤ)

Το ΠοΤ συνδέει συσκευές σε ένα περιβάλλον παραγωγής, επιτρέποντας την ανεμπόδιστη ροή δεδομένων ανάμεσα σε κόμβους και την αυτοματοποίηση, το ΙοΤ εκτελεί παρόμοιες διαδικασίες αλλά με διαφορετικό απώτερο σκοπό. Μια πιθανή εφαρμοσμένη χρήση των ΙοΤ είναι η

διαχείριση πόρων και η αυτοματοποίηση των υπηρεσιών και των διαδικασιών, σε πόλεις, για πιο αποδοτική διαχείριση των αστικών υποδομών [61][89].

Μία ενδιαφέρουσα εφαρμογή του IoT στα βιντεοπαιχνίδια είναι η δημιουργία εξατομικευμένων συσκευών ελέγχου που συνδέονται με το παιχνίδι μέσω διαδικτύου. Για παράδειγμα, ένας παίκτης μπορεί να χρησιμοποιήσει μια IoT περιφερειακή συσκευή που μετράει την θερμοκρασία του δωματίου και μεταφέρει αυτά τα δεδομένα στο παιχνίδι για να επηρεάσει τα χαρακτηριστικά του κόσμου, του χαρακτήρα ή του καιρού. Ένα άλλο παράδειγμα βρίσκεται στα έξυπνα σπίτια, τα IoT συστήματα μπορούν να συνδέονται με παιχνίδια ρόλων (RPG) ή παιχνίδια εξομοίωσης (Simulation Games), όπου οι ενέργειες στο παιχνίδι μπορούν να έχουν απτές επιδράσεις στο πραγματικό περιβάλλον του χρήστη. Για παράδειγμα, σε ένα παιχνίδι με κύκλο ημέρας, όπου η ώρα σταδιακά περνάει και ο ήλιος δύει και ανατέλλει, η θέση του ήλιου και η φωτεινότητα του χώρου στον οποίο βρίσκεται ο παίκτης, μπορεί να επηρεάζει και να ρυθμίζει την ένταση των λαμπτήρων στο δωμάτιο, όπου βρίσκεται ο χρήστης.

Στο Blockchain, τα συστήματα IoT μπορούν να χρησιμοποιηθούν για την παρακολούθηση και την επαλήθευση της φυσικής κατάστασης των αντικειμένων που συνδέονται ή αποτελούν και NFTs, επαληθεύοντας την κατάσταση του περιουσιακού στοιχείου και κρατώντας το ιστορικό αυτής [90].

Edge Computing

Στο edge computing τα δεδομένα επεξεργάζονται τοπικά, δηλαδή κοντά ή ακριβώς στο σημείο όπου συλλέγονται. Κάθε κόμβος σε ένα δίκτυο edge computing λειτουργεί αυτόνομα, επεξεργάζεται δεδομένα από τους αισθητήρες ή τις συσκευές που βρίσκονται στην αμεσότητά του, και υπάρχει η δυνατότητα λήψης αποφάσεων ή εκτέλεσης ενεργειών, βάσει αυτών των δεδομένων [65]. Με την επεξεργασία δεδομένων πιο κοντά στην πηγή τους, το edge computing μειώνει την καθυστέρηση, απαραίτητη για εφαρμογές που απαιτούν την λήψη αποφάσεων σε πραγματικό χρόνο, όπως στα παιχνίδια, όπου η άμεση ανταπόκριση είναι σημαντική για την εμπειρία του χρήστη. Στην καθημερινή ζωή, το edge computing χρησιμοποιείται σε εφαρμογές υγείας, για την ταχεία επεξεργασία και ανάλυση των δεδομένων από τις ιατρικές συσκευές.

Ως προς την βιομηχανία παιχνιδιών, παρέχει συστήματα υποστήριξης για παιχνίδια VR και AR, επεξεργάζοντας τα δεδομένα στην πηγή προέλευσης τους για την μείωση της καθυστέρησης /

χρόνου επεξεργασίας, με απώτερο σκοπό την γρηγορότερη διάδραση και εν τέλει την βελτίωση της εμπειρίας του χρήστη. Ταυτόχρονα, τα συστήματα AR κάνουν εφικτή την αλληλεπίδραση στον φυσικό χώρο με μεμονωμένους κόμβους που διαδρούν ή συναλλάσσονται με τον χώρο της επαυξημένης πραγματικότητας και τον χρήστη.

Στο Blockchain, το edge computing μπορεί να αποκεντρώσει την επεξεργασία δεδομένων, μειώνοντας την εξάρτηση από έναν κεντρικό υπολογιστή ή δίκτυο, το οποίο μειώνει την καθυστέρηση στις συναλλαγές. Παράλληλα η τεχνολογία Blockchain μπορεί να χρησιμοποιηθεί για την διασφάλιση της εγκυρότητας των κόμβων και την ασφάλεια των δεδομένων.

Υπολογιστική Νέφος (Cloud Computing)

Οι υπηρεσίες υπολογιστικής νέφος παρέχουν κλιμακωτούς πόρους για τη διαχείριση μεγάλου όγκου δεδομένων, την ασύγχρονη επεξεργασία δεδομένων μέσα από ρουτίνες εκτέλεσης, την αποφόρτωση πολύπλοκων, επεξεργαστικά, υπολογισμών από τις συσκευές του χρήστη και το streaming περιβάλλοντος εργασίας και παιχνιδιών σε συσκευές χαμηλής κατανάλωσης ενέργειας [89].

Στον κλάδο των βιντεοπαιχνιδιών η νεφο-υπολογιστική χρησιμοποιείται εκτενώς, καθώς μέσα από την δημιουργία και διαχείριση εικονικών servers, εικονικών μηχανών (virtual machines) ή instances είναι εφικτή η κλιμάκωση του μεγέθους του Backend για ένα online βιντεοπαιχνίδι, όπως ένα MMO, που μπορεί να δέχεται διαφορούμενο όγκο χρηστών σε ασαφής στιγμές – και όλοι οι χρήστες πρέπει να εξυπηρετηθούν ταυτόχρονα. Παράλληλα επιτρέπει το streaming παιχνιδιών σε επεξεργαστικά αδύναμες συσκευές, όπως κινητά [90].

Στον κλάδο των Blockchain, η υπολογιστική νέφος προσφέρει υποδομές επιπέδου Enterprise για την διαχείριση, επεξεργασία και αποθήκευση πολύ μεγάλου όγκου δεδομένων. Οι Cloud Servers μπορούν να χρησιμοποιηθούν τόσο σαν Node, όσο και σαν το κέντρο / κεντρικός υπολογιστής ενός κεντροποιημένου συστήματος Blockchain [90].

Κυβερνο-Φυσικά Συστήματα (CPS)

Τα Κυβερνο-Φυσικά Συστήματα (CPS) είναι συστήματα που συνδυάζουν υπολογιστικά στοιχεία (κυβερνο-) με φυσικές διεργασίες (-φυσικά). Αυτά τα συστήματα είναι πολυμερή και τα μεμονωμένα μέρη δημιουργούν δίκτυα επικοινωνίας μέσα από τα οποία επιτυγχάνουν την αλληλεπίδραση με τον φυσικό κόσμο, πραγματοποιώντας επεξεργασίες και λαμβάνοντας

αποφάσεις με βάση τα συλλεγόμενα δεδομένα από αισθητήρια όργανα. Ουσιαστικά, παρέχουν την δυνατότητα ψηφιακής εποπτείας πάνω σε φυσικά συστήματα και φυσικές λειτουργίες, διευκολύνοντας την παρακολούθηση και τον έλεγχο των συστημάτων σε πραγματικό χρόνο. Στην καθημερινότητα, τα CPS βρίσκονται παντού τριγύρω μας, στον μοντέρνο ψηφιακό κόσμο. Για παράδειγμα, τα σύγχρονα αυτοκίνητα με τις τεχνολογίες βελτιστοποίησης της απόδοσης και της κατανάλωσης καυσίμων, όπως και τα συστήματα ασφαλείας και παρακολούθησης της οδηγικής συμπεριφοράς (lane assist, ανίχνευση σύγκρουσης), εφαρμόζουν συστήματα CPS σχεδόν σε όλες τις λειτουργίες τους [64].

Στα παιχνίδια, τα CPS μπορούν να ενισχύσουν την αλληλεπίδραση του φυσικού υλικού με το λογισμικό, βελτιώνοντας την εμπειρία παιχνιδιού. Επίσης παρέχουν πολύτιμα μεταδεδομένα για την ψηφιοποίηση των αγαθών, ως προς την συμπερίληψη τους σε εικονικούς χώρους. Ταυτόχρονα, είναι απολύτως αναγκαία για την τεχνολογία των συστημάτων AR και την δημιουργία ψηφιακών διδύμων [57][60][62].

Στο Blockchain, τα CPS υποστηρίζουν την ανάπτυξη αποκεντρωμένων εφαρμογών στον πραγματικό κόσμο, συγχρονίζοντας ψηφιακές συναλλαγές με φυσικές διαδικασίες.

Τεχνητή Νοημοσύνη (Artificial Intelligence - AI) και Μηχανική Μάθηση(Machine Learning - ML)

Αυτές οι τεχνολογίες αποτελούν κεντρικό μηχανισμό της διαδικασίας λήψης αποφάσεων σε αυτόματα συστήματα καθώς και στην διαδικασία επεξεργασίας δεδομένων και αναγνώρισης οντοτήτων, βελτιώνοντας τον βαθμό αυτόματης λειτουργίας των συστημάτων μέσα από την χρήση προηγμένης ανάλυσης δεδομένων και προσαρμοστικών αλγορίθμων. Ένα παράδειγμα εφαρμογής τους βρίσκεται στον κλάδο των αυτοκινούμενων οχημάτων, όπου συμβάλουν στην ασφάλεια στις μεταφορές και προσφέρουν αυτόματες υπηρεσίες χωρίς τον παράγοντα του ανθρώπινου λάθους, αν και αυτή η τεχνολογία βρίσκεται ακόμα στα αρχικά στάδια της [89].

Στον κλάδο των βιντεοπαιχνιδιών, όπου η «λογική» έχει κεντρικό ρόλο τόσο στην παραγωγή ρεαλιστικών παραγωγικών procedural αλγορίθμων όσο και στην διαδικασία δημιουργίας μη-παικτών χαρακτήρων (non-player NPC) που προσφέρουν ρεαλιστικές αλληλεπιδράσεις με παραγωγή λόγου και κατανόηση των συμβάντων, όπως και μαθαίνουν μέσα από τις πράξεις του παίκτη και προσαρμόζονται για να γίνουν πιο ανταγωνιστικά.

Στα συστήματα Blockchain αυτοματοποιούν την διαδικασία δημιουργίας και εκτέλεσης έξυπνων συμβολαίων και συμβάσεων, προσφέροντας αξιοπιστία και διαφάνεια. Παράλληλα, μέσα από τις τεχνολογίες αυτές είναι εφικτή η αυτόματη ανίχνευση κακόβουλων οντοτήτων και συναλλαγών.

Γνωστική Υπολογιστική (Cognitive Computing)

Η γνωστική υπολογιστική αποτελείται από συστήματα που προσομοιώνουν τη λειτουργία του ανθρώπινου εγκεφάλου μέσα σε ένα υπολογιστικό περιβάλλον, αναγνωρίζοντας μοτίβα και χρησιμοποιώντας αλγόριθμους αυτο-μάθησης. Μέσα στα πλαίσια την καθημερινότητας, η γνωστική υπολογιστική μπορεί να χρησιμοποιηθεί σε μαθησιακά προγράμματα, όπου μπορεί να παρέχει προσωποποιημένη βοήθεια στους μαθητές και να προσαρμόζεται στις ξεχωριστές μαθησιακές ανάγκες το κάθε μαθητή.

Στα βιντεοπαιχνίδια, η γνωστική υπολογιστική εντείνει τον βαθμό αλληλεπίδρασης του χρήστη με τους μη-παίκτες χαρακτήρες (NPC), μέσα από την δημιουργία ρεαλιστικών χαρακτήρων με ανθρώπινα χαρακτηριστικά που αντιδρούν στις «εισόδους» (inputs) του παίκτη σε πραγματικό χρόνο.

Στα δίκτυα Blockchain παρέχει διαδικασίες αυτόματης ανάλυσης και προσαρμοστικής αντίδρασης σε απειλές, όπως και ανεπτυγμένες λειτουργίες σε συστήματα έξυπνων συμβολαίων.

Κεφάλαιο 6 ΠΡΑΚΤΙΚΟ ΜΕΡΟΣ

6.1 Παρουσίαση Κατασκευής

Για το πρακτικό μέρος της έρευνας, κατασκευάστηκε, δοκιμαστικά, ένα ALPHA πρωτότυπο παιχνιδιού μέσα από την χρήση τεχνολογιών αιχμής, για την επίδειξη και την εξερεύνηση των δυνατοτήτων χρήσης των νομισμάτων με μοναδικά τεκμήρια NFT. Παράλληλα δημιουργήθηκαν κάρτες μέσα από αυτόματες διαδικασίες τεχνητής νοημοσύνης παρόμοιες με εκείνες που βρίσκουν χρήση σε τεχνολογίες της Βιομηχανίας 4.0. Έπειτα, η δημιουργία μοναδικών τεκμηρίων υλοποιήθηκε μέσα από το καταναμημένο καθολικό του δικτύου Ethereum και την κατάσταση (της μακροπρόθεσμης μνήμης) της εικονικής μηχανής του Ethereum (EVM), μέσα από την χρήση έξυπνων συμβολαίων γραμμένα στην γλώσσα Solidity. Τέλος, τα μεταδεδομένα των νομισμάτων – καρτών αποθηκεύονται στο IPFS, όπως και οι εικόνες ξεχωριστά, και το παιχνίδι εμπεριέχει ένα marketplace για την αγοραπωλησία καρτών από τους χρήστες.

Το ίδιο το παιχνίδι φέρει στοιχεία παιχνιδιού συλλογής και ανταλλαγής καρτών και απαιτεί την πρόσβαση στο διαδίκτυο για την εκτέλεση των διαδικασιών που αφορούν το έξυπνο συμβόλαιο και τα νομίσματα – κάρτες. Το πρωτότυπο κατασκευάστηκε στην μηχανή παιχνιδιών Unreal Engine 5, μέσα από την χρήση πολλαπλών προσθηκών (plugin) και έγινε η χρήση πολλαπλών αγορασμένων και ελεύθερων τρισδιάστατων στοιχείων για την κατασκευή και τον σχεδιασμό του εικονικού χώρου. Αγοράστηκαν τα δικαιώματα χρήσης των κομματιών ήχου και τραγουδιών και τροποποιήθηκαν ή κατασκευάστηκαν εξ' αρχής πολλά μέρη της λογικής του παιχνιδιού για να εκτελούνται σωστά οι απαιτούμενες διαδικασίες χρήσης και επικοινωνίας με Blockchain σύστημα εν χρήση.

Ακολούθως, παρουσιάζεται η διαδικασία κατασκευής όλων των επιμέρους στοιχείων του πολυσωματικού που απαρτίζει το παιχνίδι. Η παρουσίαση ξεκινάει με την δημιουργία των καρτών, όντας το βασικό στοιχείο του παιχνιδιού ως προς την χρήση του από τους παίκτες. Αντίθετα, το σύστημα Blockchain αποτελεί ένα δευτερεύον σύστημα προς την κύρια φύση του παιχνιδιού ως μια εικονική εμπειρία, παρόλο που αποτελεί βασικό θέμα της έρευνας. Έπειτα παρουσιάζεται το πως το κεντρικό αντικείμενο του παιχνιδιού (κάρτες) επηρεάζει την διαδικασία δόμησης και κατασκευής του παιχνιδιού και των υποστηριζόμενων συστημάτων, όπως το δίκτυο Ethereum. Στην συνέχεια επιδεικνύεται η μηχανή και η διαδικασία

μορφοποίησης του σώματος του παιχνιδιού και της λογικής των στοιχείων παιχνιδιού σαν ένα πρώτο μέρος και παρουσιάζεται η υποθετική δομή ενός απλού συμβολαίου, τύπου ERC-721, ειδικά φτιαγμένου αποκλειστικά για χρήση προς το συγκεκριμένο εν κατασκευή παιχνίδι. Εν συνέχεια, μετά την παρουσίαση της υποθετικής κατασκευής, αναλύονται οι περιορισμοί που θα έφερε η χρήση ενός έξυπνου συμβολαίου αυτής της φύσης και αιτιολογείται η κατασκευή και χρήση του ERC-1155, που τελικά χρησιμοποιήθηκε μέρος στο έργο. Στην συνέχεια παρουσιάζονται οι προσαρμογές που πήραν μέρος και τα εργαλεία που χρησιμοποιήθηκαν για την διασύνδεση του παιχνιδιού με το δίκτυο Ethereum, την χρήση πορτοφολιών μέσα από το Metamask και για την εκτέλεση του έξυπνου συμβολαίου. Τέλος, παρουσιάζεται το έξυπνο συμβόλαιο ERC-1155 και οι μεθοδολογίες και περιορισμοί κατασκευής του, η δομή του και πως αυτή επηρεάζει την εν τέλει λειτουργικότητα των μοναδικών τεκμηρίων και το πως προκύπτουν τα μοναδικά τεκμήρια μέσα από αυτό το σύστημα.

Στο όλον της παρουσίασης η κατασκευή διατηρεί μια «αντικειμενοστρέφια» προς το συναλλασσόμενο ψηφιοποιημένο ον με το οποίο καταπιάνεται, σε αυτήν την περίπτωση οι κάρτες. Για την καλύτερη εκτέλεση του πειράματος (της κατασκευής), στοιχεία του παιχνιδιού και του έξυπνου συμβολαίου κατασκευάζονται και δομούνται έτσι ώστε να μπορούν να διαχειριστούν ασύγχρονα, μεταβαλλόμενα στοιχεία που μπορούν να προσδιοριστούν σε δεύτερο πραγματικό χρόνο, μετά την δημιουργία του παιχνιδιού και κατά την ώρα εκτέλεσης του. Αυτά τα απροσδιόριστα στοιχεία λειτουργούν συμβολικά προς την διαλειτουργική φύση των συστημάτων της Βιομηχανίας 4.0 και των νέων απαιτήσεων που προκύπτουν μέσα από την επαυξημένη πραγματικότητα και το metaverse, όπου πολλαπλοί ψηφιακοί, και μη, χώροι έρχονται σε σύγκλιση. Ταυτόχρονα, αναλύεται η δυνατότητα των συστημάτων εν χρήση να διαχειριστούν περίπλοκα δυναμικά και μεταλλασσόμενα όντα που συναλλάσσονται προς αυτά, από τρίτα εξωτερικά συστήματα.

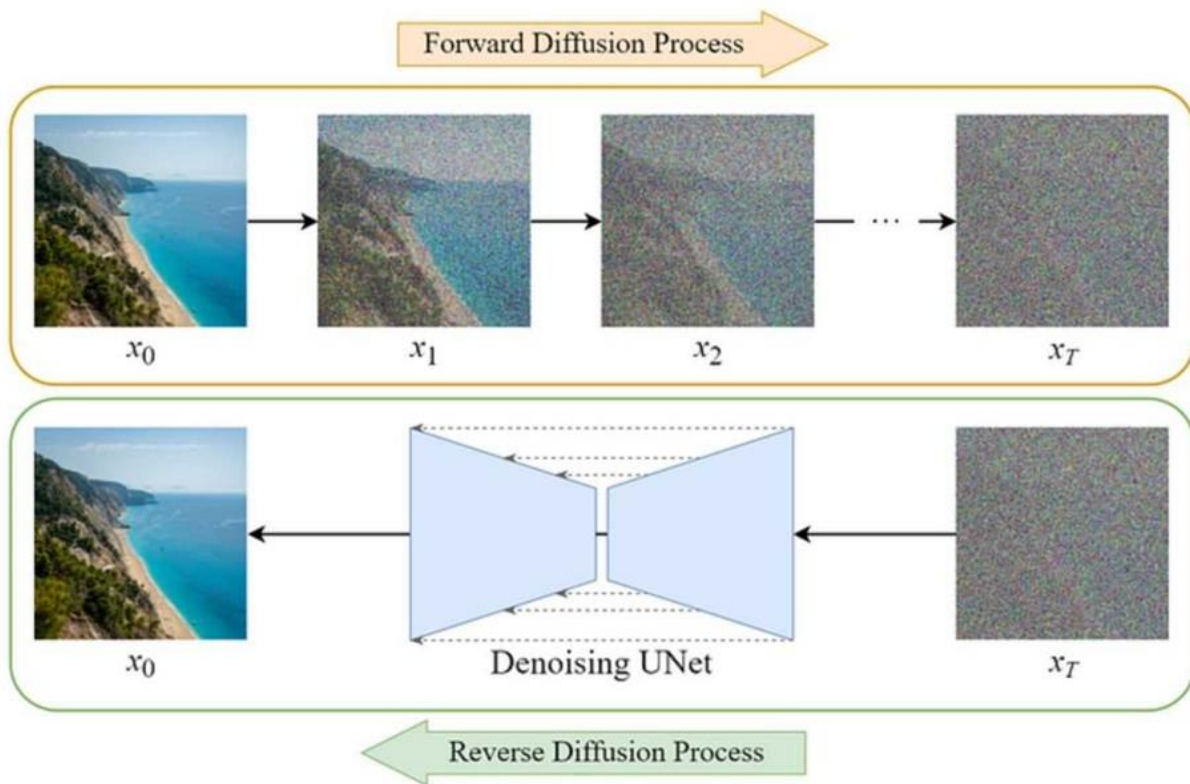
Ουσιαστικά, επιλέχθηκαν οι κάρτες ως το κύριο στοιχείο του παιχνιδιού και όχι το παιχνίδι για να προστεθεί ένας βαθμός ασάφειας και περιπλοκής στην εφαρμογή των NFT στο παιχνίδι.

6.2 Κατασκευή Καρτών και Στοιχείων UI μέσα από την χρήση Γενετικής Τεχνητής Νοημοσύνης (Generative AI)

Για την κατασκευή των καρτών χρησιμοποιήθηκε το Stable Diffusion [73], ένα μοντέλο τεχνητής νοημοσύνης της Stability AI που δημιουργεί εικόνες από περιγραφές σε μορφή κειμένου. Το Stable Diffusion χρησιμοποιεί μια τεχνική βαθιάς μάθησης (Deep Learning) που ονομάζεται 'diffusion'.

Κατά τη διαδικασία εκπαίδευσης του μοντέλου, χρησιμοποιούνται σετ εικόνων με περιγραφικά κείμενα [74]. Σε αυτές τις εικόνες εφαρμόζεται σταδιακά Gaussian θόρυβος, ώστε να μετατραπούν από καθαρές σε πλήρως θορυβώδεις. Αυτή η διαδικασία δημιουργεί δεδομένα που βοηθούν το μοντέλο να μάθει πώς να αντιστρέφει τον θόρυβο.

Έπειτα, λαμβάνει χώρα η αντίστροφη διαδικασία, κατά την οποία το μοντέλο αφαιρεί σταδιακά τον θόρυβο χρησιμοποιώντας την αρχιτεκτονική U-Net [92]. Αυτή η διαδικασία εκπαίδευσης του νευρωνικού δικτύου επιτρέπει στο μοντέλο να συνδέει τις λέξεις των περιγραφών κειμένου με τα αντικείμενα και τις θέσεις τους μέσα στις παραγόμενες εικόνες.



Επισκόπηση του Μοντέλου Diffusion Πηγή: [<https://www.ikomia.ai/blog/stable-diffusion-xl-sd-xl-model#run-stable-diffusion-xl-sd-xl-with-a-few-lines-of-code>]

Τα μοντέλα Diffusion μπορούν να εκπαιδευτούν από την αρχή ή μπορούν να λάβουν προσαρμογές και επιπλέον εκπαίδευση σε νέα σετ δεδομένων, προσθέτοντας λειτουργίες και δυνατότητες. Ταυτόχρονα μπορούν να χρησιμοποιηθούν σαν βάση για την δημιουργία νέων μοντέλων. [93]

Για την κατασκευή των κεντρικών εικόνων των καρτών χρησιμοποιήθηκαν τα μοντέλα DreamShaper XL και Juggernaut XL, τα οποία αποτελούν προηγμένες εκδόσεις του μοντέλου Stable Diffusion XL. Για τα στοιχεία της διεπαφής χρήστη χρησιμοποιήθηκε το LoRA (Low-Rank Adaptation)[78] CGgamebuttonsw από τον χρήστη songwei2698 της πλατφόρμας civitai, με το μοντέλο DreamShaper-v8.

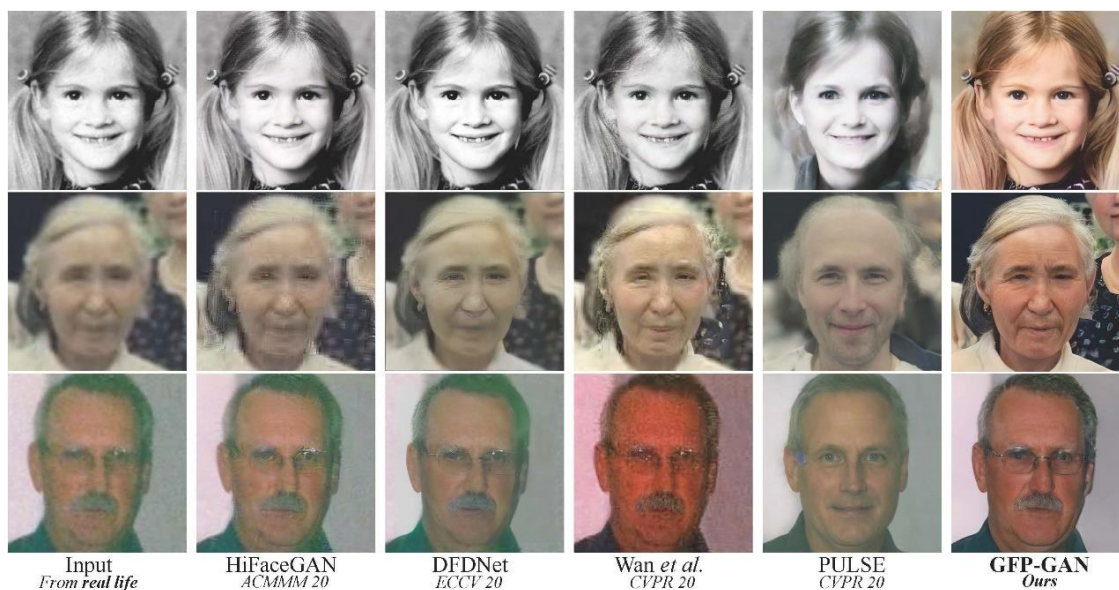
Για την ομοιομορφία των γραφικών, το ίδιο θετικό και αρνητικό κείμενο χρησιμοποιείται σε όλες τις κάρτες και αντίστοιχα σε στοιχεία του UI, με μερικές, μόνο, εναλλασσόμενες λέξεις για την αποφυγή παραγωγής παρόμοιου αποτελέσματος κάθε φορά. Αυτές οι νέες λέξεις εμπεριέχουν τα βασικά χαρακτηριστικά της εικόνας και τα στοιχεία που εμπεριέχει, ενώ οι υπόλοιπες φροντίζουν για την ομοιομορφία του στυλ.

Μετά την παραγωγή των καρτών, χρησιμοποιείται ένα ενισχυμένο μοντέλο super-resolution που βασίζεται στο ESRGAN (Enhanced Super-Resolution Generative Adversarial Networks), το οποίο είναι ένας τύπος μοντέλου βαθιάς μάθησης που χρησιμοποιεί Παραγωγικά Αντιπαλικά Δίκτυα (Generative Adversarial Networks – GAN)[87]. Το μοντέλο που χρησιμοποιείται είναι το R-ESRGAN 4x [88], το οποίο σχεδιάστηκε για την αναβάθμιση εικόνων χαμηλής ανάλυσης, δημιουργώντας, κατά την διαδικασία μεγέθυνσης, υψηλότερης ποιότητας και πιο ρεαλιστικές λεπτομέρειες σε σύγκριση με τα προηγούμενα μοντέλα που εκτελούν παρόμοιες λειτουργίες. Έπειτα, για τις φωτογραφίες που εμπεριέχουν πρόσωπα, χρησιμοποιείται το μοντέλο μηχανικής μάθησης GFP-GAN. Το μοντέλο GFP-GAN αποκαθιστά τις μικρές λεπτομέρειες του προσώπου και διορθώνει οποιαδήποτε μορφή παραμόρφωσης ή λάθους που εισάγεται κατά τη διαδικασία του upscaling και σε έναν βαθμό κατά την αρχική διαδικασία δημιουργίας της εικόνας,

βελτιώνοντας τα χαρακτηριστικά του προσώπου και μεταποιώντας τα ώστε να εμφανίζονται φυσικά και λεπτομερή[77].



Upscaling μέσα από την χρήση του RealESRGAN. Πηγή: [<https://github.com/xinntao/Real-ESRGAN>]



Σύγκριση Τεχνικών Αποκατάστασης Προσώπου παρουσιάζοντας τα αποτελέσματα της μεθόδου GFP-GAN, σε σύγκριση με τις άλλες μεθόδους. Πηγή:

[<https://github.com/TencentARC/GFPGAN>]

Συγκεκριμένα, για την μέθοδο δημιουργίας των καρτών, χρησιμοποιήθηκε το εργαλείο από τον χρήστη AUTOMATIC1111 στο Github [79]. Η διεπαφή χρήστη Web UI από τον χρήστη Automatic1111 παρέχει έναν φιλικό -προς τον χρήστη- περιβάλλον που επιτρέπει την εύκολη αλληλεπίδραση με τα μοντέλα του Stable Diffusion. Επίσης, προσφέρει εκτεταμένες επιλογές προσαρμογής, υποστηρίζει διάφορες επεκτάσεις και plugins που μπορούν να ενισχύσουν τη λειτουργικότητα. Παρέχει λειτουργίες inpainting και outpainting και μπορεί να εκτελεστεί τοπικά μέσα στο παράθυρο του προγράμματος περιήγησης του χρήστη.

Ακολούθως, παρουσιάζεται η διαδικασία δημιουργίας των εικόνων ενός νέου στοιχείου της διεπαφής χρήστη:

Η διαδικασία βασίζεται κυρίως στο μοντέλο Diffusion και στην περιγραφή κειμένου που το καθοδηγεί. Έτσι, το πρώτο βήμα για τη δημιουργία μιας εικόνας με συγκεκριμένα επιθυμητά χαρακτηριστικά είναι η χρήση ενός μοντέλου, το οποίο έχει εκπαιδευτεί πάνω σε σετ εικόνων που εμπεριέχουν τα αντικείμενα της περιγραφής.

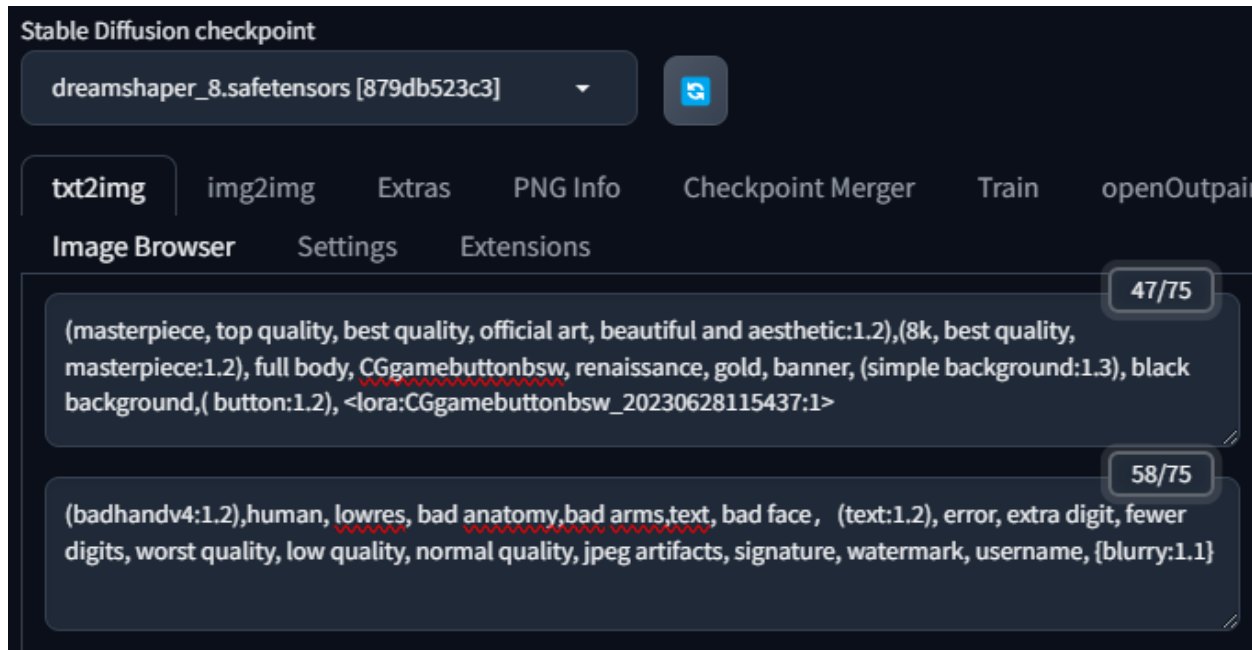
Έπειτα, πρέπει να μορφοποιηθεί η περιγραφή, καθώς η δομή της εικόνας ορίζεται από την περιγραφή σε μορφή κειμένου (positive prompt και negative prompt) που παρέχεται. Αυτό το κείμενο, αποτελεί μια λεπτομερή περιγραφή των στοιχείων που θα περιλαμβάνει η εικόνα και της θέσης τους. Το κείμενο αυτό μετατρέπεται σε tokens και το μοντέλο CLIP (Contrastive Language–Image Pre-training) μετατρέπει αυτά τα tokens, μαζί με το σημασιολογικό τους νόημα, σε αριθμητικές αναπαραστάσεις γνωστές ως embeddings – οι οποίες διατηρούν τόσο τα αντικείμενα και τα χαρακτηριστικά τους, όσο και τη σημασία του κειμένου. Τα embeddings χρησιμοποιούνται στη συνέχεια από το μοντέλο U-Net για την καθοδήγηση της διαδικασίας δημιουργίας της εικόνας.

Στην ακόλουθη εικόνα, επιδεικνύεται η διαδικασία σύνθεσης της περιγραφής κειμένου (prompt) και η αυτόματη αρίθμηση των tokens από την εφαρμογή. Η περιγραφή που δίνεται είναι:

- Περιγραφή κειμένου: (masterpiece, top quality, best quality, official art, beautiful and aesthetic:1.2),(8k, best quality, masterpiece:1.2), full body, CGgamebuttonbsw, renaissance, gold, banner, (simple background:1.3), black background,(button:1.2), <lora:CGgamebuttonbsw_20230628115437:1>

- Αρνητική περιγραφή: (badhandv4:1.2),human, lowres, bad anatomy,bad arms,text, bad face, (text:1.2), error, extra digit, fewer digits, worst quality, low quality, normal quality, jpeg artifacts, signature, watermark, username, {blurry:1.

Δίνονται τα κείμενα περιγραφής και γίνεται χρήση του CGgamebuttonbsw LoRA μέσω της συμπερίληψης της λέξης 'CGgamebuttonbsw' στην περιγραφή και της φράσης '<lora:CGgamebuttonbsw_20230628115437:1>'. Το αρνητικό prompt περιγράφει τι πρέπει να αποφύγει να παράξει το νευρωνικό δίκτυο. Οι λέξεις «renaissance», «gold», «banner» (ή, προτιμότερα, μόνο η λέξη «banner») μαζί με το Seed, αποτελούν τα μόνα στοιχεία που θα αλλάξουν κατά τη δημιουργία των εικόνων ολόκληρης της διεπαφής χρήστη (UI) για την επίτευξη ενός ομοιόμορφου και αισθητικού αποτελέσματος.



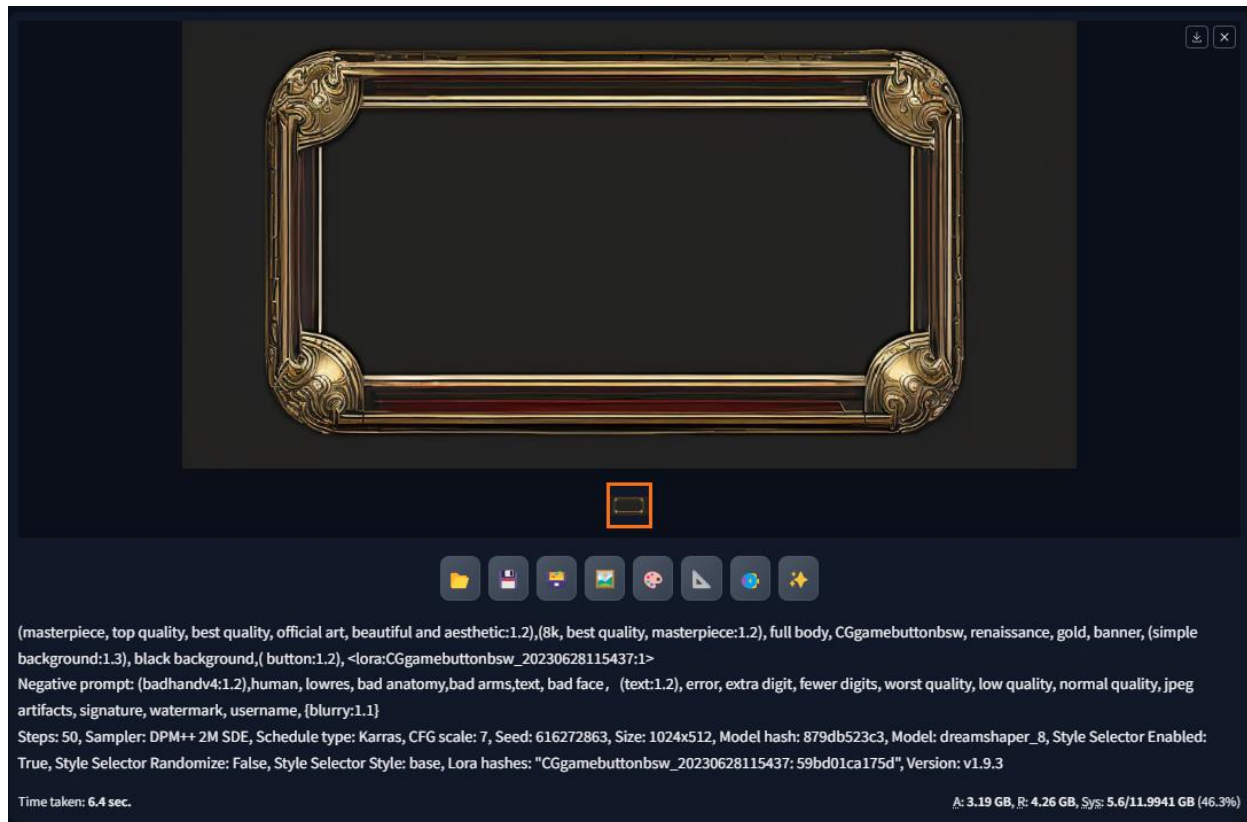
Κείμενο περιγραφής για την δημιουργία εικόνας μέσα στο WebUI του Stable Diffusion από τον χρήστη Automatic1111

Πριν την παραγωγή της εικόνας, είναι σημαντική η επιλογή των κατάλληλων ρυθμίσεων για την επίτευξη του επιθυμητού αποτελέσματος και τη σωστή παραγωγή της εικόνας. Στην περίπτωση χρήσης LoRA, μπορεί να υπάρχουν απαιτήσεις για την εισαγωγή συγκεκριμένων ρυθμίσεων για τη χρήση του.



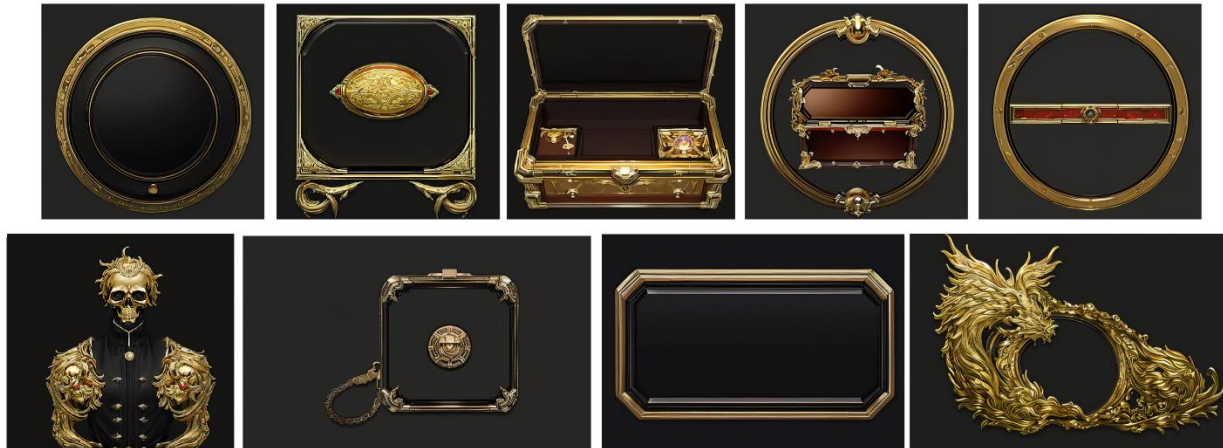
Διεπαφή Ρύθμισης για Δημιουργία Εικόνας με Stable Diffusion WebUI, επιδεικνύοντας τις ρυθμίσεις που χρησιμοποιήθηκαν

Μετά την ολοκλήρωση της διαδικασίας γένεσης, εμφανίζεται η εικόνα που δημιουργήθηκε στο παράθυρο περιήγησης, όπως παρουσιάζεται ακολούθως:



Το παράθυρο παρουσίασης της παραγόμενης εικόνας, μέσα στο WebUI του Stable Diffusion, μετά την διαδικασία δημιουργίας μαζί με το κείμενο περιγραφής που χρησιμοποιήθηκε συμπεριλαμβανομένων και όλων των πληροφοριών από τις ρυθμίσεις.

Οι ακόλουθες εικόνες δημιουργήθηκαν με την μέθοδο που προαναφέρεται, αλλάζοντας την λέξη 'banner' και παράγοντας πολλαπλές εικόνες με διαφορετικά seed. Αν οι διάσταση ή οι λεπτομέρειες των παραγόμενων εικόνων δεν είναι σωστές, τότε χρησιμοποιείται το μοντέλο R-ESRGAN 4 για την αναβάθμιση της εικόνας και Refiner για την αναδιαμόρφωση της εικόνας μέσω του εμπλουτισμού και αναδημιουργίας των λεπτομερειών.



Εικόνες διεπαφής χρήστη, που δημιουργήθηκαν μέσα από διαδικασίες μηχανικής μάθησης με την χρήση μοντέλου Diffusion

Στην συνέχεια χρησιμοποιήθηκε το Adobe Photoshop για την αφαίρεση του φόντου, την επεξεργασία των διαστάσεων και την ρύθμιση των χρωμάτων, για την επίτευξη ενός ομοιόμορφου αποτελέσματος μέσα από την χρήση μιας κοινής παλέτας ανάμεσα στα μεμονωμένα στοιχεία της διεπαφής. Παράλληλα, για την διόρθωση ατελειών και την αναδιαμόρφωση των στοιχείων των εικόνων εφαρμόστηκαν τεχνικές χειραγώγησης (photomanipulation) της φωτογραφίας εντός του Photoshop.



Εικόνες διεπαφής μετά την επεξεργασία τους στο Photoshop. Εικόνα 1 (αριστερά): Διακοσμητικό πλαίσιο για την εικόνα προφίλ χρήστη. Εικόνα 2 (κέντρο): Εικονίδιο κουμπιού για την πρόσβαση σε μενού επίδειξης των περιουσιακών στοιχείων του παίκτη. Εικόνα 3 (δεξιά): Πλαίσιο κουμπιών του κεντρικού μενού.

Για την διαδικασία κατασκευής των καρτών, χρησιμοποιήθηκαν οι προαναφερόμενες τεχνικές με τα μοντέλα DreamShaper XL και Juggernaut XL. Ταυτόχρονα χρησιμοποιήθηκε ένα μεγάλο σετ από LoRA που έχουν εκπαιδευτεί, το καθένα ξεχωριστά, στο να παράγουν εικόνες με διαφορετικά στοιχεία. Για παράδειγμα ένα LoRA παρήγαγε εικόνες γραφικών που παρουσιάζουν αγγέλους με φτερά στην πλάτη τους και ένα άλλο είχε εκπαιδευτεί στο να παράγει δαίμονες με κέρατα και φλόγες. Ακολουθεί ένα παράδειγμα με τους αγγέλους, όπου χρησιμοποιήθηκε η ίδια περιγραφή για να παράξει 4 διαφορετικές εκδόσεις της ίδιας εικόνας:



Παράδειγμα δημιουργίας φωτογραφιών με την ίδια κειμενική περιγραφή και LoRA που παράγει ήρωες με άσπρα φτερά.

6.3 Κατασκευή Βιντεοπαιχνιδιού

Η εφαρμογή που χρησιμοποιεί το σύστημα Blockchain κατασκευάστηκε ως βιντεοπαιχνίδι εντός της μηχανής Unreal Engine 5 και η δομή του ίδιου του παιχνιδιού πλαισιώνεται από ένα πλήθος από προσθήκες. Αυτές οι προσθήκες έχουν την μορφή ελεύθερου κώδικα στο GitHub ή αγορασμένων πρόσθετων από το marketplace της εταιρίας Epic Games που κατασκευάζει την ίδια την μηχανή παιχνιδιού. Αναφορικά, τα πρόσθετα που χρησιμοποιήθηκαν για δημιουργηθεί ο βασικός πυρήνας του παιχνιδιού είναι τα ακόλουθα:

6.3.1 Εργαλεία και Λειτουργικότητα

✚ **Big Integer Utilities** από τον χρήστη Ding

- Καλύπτει την ανάγκη επεξεργασίας αριθμών έξω από την τυπική κλίμακα των 32 και 64bit και επιτρέπει πράξεις επ' αυτούς, προσφέροντας ακρίβεια 32.768bit σε δεκαεξαδική μορφή.
 - ❖ Αν δεν χρησιμοποιούταν το πρόσθετο, η εναλλακτική θα ήταν να δημιουργηθεί προσαρμοσμένη λογική σε C++ που να επιτρέπει τις πράξεις σε αριθμούς με πάνω από 256bit, διότι:
 - Το Ethereum VM χρησιμοποιεί intergers των 256bit, και απαιτείται ακρίβεια των 256bit.
 - Η εγγενής υποστήριξη για ακέραιους αριθμούς 256-bit δεν είναι διαθέσιμη στην κλασική C++.
 - Οι βασικοί τύποι δεδομένων της Unreal Engine 5 δεν προσφέρουν interger με ακρίβεια 256bit.
 - Θα έπρεπε να δημιουργηθεί κώδικας για την C++ ο οποίος θα εφαρμόζει αριθμητική αυθαίρετης ακρίβειας ή αριθμητική πολλαπλής ακρίβειας μέσα από την χρήση βιβλιοθηκών.

✚ IPFS από την εταιρία 3S Game Studio

- Προσφέρει έτοιμα functions για το σύστημα με τα Blueprints της μηχανής, για την πρόσβαση σε αρχεία του Διαπλανητικού Συστήματος Αρχείων (InterPlanetary File System - IPFS) και του InterPlanetary Name System (IPNS).
 - ❖ Επιτρέπει τις ασύγχρονες διαδικασίες λήψης αρχείων χωρίς να παγώνει το πρόγραμμα αλλά είναι σημαντικά πιο περίπλοκο στη χρήση από το απλό βασικό function "Download Image" στην Unreal.

✚ JSONParser από Virtualbird UG

- Επιτρέπει την ταχεία επεξεργασία των αρχείων Json που κατεβαίνουν από το IPFS, χωρίς την δημιουργία μεγάλων και πολύπλοκων συναρτήσεων για τον χειρισμό και την συντακτική ανάλυση Json αρχείων, εντός των Blueprint.

✚ Quixel Bridge από την εταιρία Epic Games

- Προσφέρει ένα μεγάλο πλαίσιο από τρισδιάστατα στοιχεία κόσμου παιχνιδιού, τα οποία έχουν δημιουργηθεί από τεχνικές 3D Scanning και φωτογραμμετρίας.
 - ❖ Τα στοιχεία προσφέρονται σε εκδόσεις με υψηλό και χαμηλό αριθμό τριγώνων και μερικά έχουν εκδόσεις Nanite, το οποίο Nanite είναι μια τεχνολογία που επιτρέπει την

χρήση και το rendering στοιχείων με τεράστιο αριθμό τριγώνων σε πραγματικό χρόνο, μεταδίδοντας δυναμικά μόνο την απαραίτητη λεπτομέρεια σε κάθε δεδομένη στιγμή.

- Τα “Highest Quality” και “Nanite” επίπεδα λεπτομέρειας έχουν πολύ μεγάλο μέγεθος και διογκώνουν σημαντικά το τελικό μέγεθος του παιχνιδιού και τις απαιτήσεις συστήματος.

✚ **Roguelike Deckbuilder Toolkit** από τον Knut Overbye

- Βασικό στοιχείο του παιχνιδιού, προσφέροντας λογική και λειτουργικότητα παιχνιδιού καρτών. Αυτό ήταν το αρχικό πρόσθετο γύρω από το οποίο φτιάχτηκε το project εντός της Unreal.

✚ **Runtime Files Downloader** από τον Georgy Treshchev

✚ **SHA256 Hash Functions** από Blue Mountains

6.3.2 VFX και Εφέ Σωματιδίων

✚ **Blink and Dash VFX** από τον χρήστη pelengami

✚ **Niagara Appearance Effect** από τον AlexHuang

✚ **Niagara Magic VFX Bundle - Niagara Magic Projectiles - Niagara Magic AOE** από την εταιρία Sidearm Studios

✚ **Niagara Morph Effect** από τον AlexHuang

✚ **Mixed Vfx Pack 2 (Niagara)** από suppART

✚ **Infinity Blade: Effects** από την εταιρία Epic Games

6.3.3 Πρόσθετα Ήχων

✚ **Fantasy Card & Board Game Vol.1 - UI Navigation and Mechanics** από τους Cyberwave Orchestra

✚ **Magic Spells Sound Pack** από Magic Sound Effects

6.3.4 Στοιχεία κόσμου παιχνιδιού

✚ **Dark Medieval Environment Megapack + ULAT (Modular , Medieval , Medieval)** από την εταιρία Leartes Studios

✚ **Fantasy Bundle Environment Kit 3 in 1** από τον χρήστη Denys Rutkovskiy

✚ **Fantasy Castle Environment + ULAT (Fantasy, Soulslike, Castle, Fantasy)** από την εταιρία Leartes Studios

- ✚ **Gladiator Arena Environment Kit** από τον χρήστη Denys Rutkovskyi
- ✚ **Infinity Blade: Fire Lands** από την εταιρία Epic Games
- ✚ **Infinity Blade: Grass Lands** από την εταιρία Epic Games
- ✚ **Infinity Blade: Ice Lands** από την εταιρία Epic Games
- ✚ **Ithris Cemetery** από τον Rasmus Bagner
- ✚ **Matte Painting Skybox Pack: Chaotic Skies II** από Velarion
- ✚ **Medieval Street Pack (Medieval Town, Medieval Buildings, Medieval Clock Tower)** από FreshCan
- ✚ **Medieval Village Megapack (Medieval Interiors Modular Medieval Props)** από FreshCan
- ✚ **Modular Gothic/Fantasy Environment** από την εταιρία Stormcrow Studios
- ✚ **Modular Medieval Environment (Medieval Castle, Medieval Houses, Medieval Church)** από FreshCan
- ✚ **Mountain - Ancient Mountain** από Hivemind
- ✚ **Paragon: Agora and Monolith Environment** από την εταιρία Epic Games
- ✚ **Soul: Cave** από την εταιρία Epic Games
- ✚ **Temple Of The Gods - Ancient Temple - Modular Temple** από Hivemind
- ✚ **Victorian Interior Environment + ULAT** από την εταιρία Leartes Studios

6.3.5 Σύστημα διασύνδεσης παιχνιδιού με το έξυπνο συμβόλαιο

Εκτός από τα προαναφερόμενα εργαλεία, χρησιμοποιήθηκε ένα ακόμη πρόσθετο, το οποίο παρέχει τον πυρήνα της λειτουργικότητας και επικοινωνίας με το έξυπνο συμβόλαιο – αυτό είναι το WEB3.Unreal του project Game7 [80] και ανήκει στο Web3 DAO (Decentralized Autonomous Organization - Αποκεντρωμένη Αυτόνομη Οργάνωση) BitDAO σε συνεργασία με την Forte. Το Game7 είναι ένα Web3 gaming project, το οποίο έχει δημιουργηθεί για να επιταχύνει τον ρυθμό απορρόφησης τεχνολογιών blockchain από την βιομηχανία των Βιντεοπαιχνιδιών. Όπως αναγράφεται στην ιστοσελίδα του BitDAO «Το BitDAO είναι ένα επενδυτικό DAO που δημιουργήθηκε το 2021, και υποστηρίζεται από τους Peter Thiel, Founders Fund και Bybit. Διαθέτει το μεγαλύτερο ταμείο που διευθύνεται από DAO στον κόσμο, με περίπου 2,5 δισεκατομμύρια δολάρια σε διαχειριζόμενα περιουσιακά στοιχεία.» [81].

Το WEB3.Unreal είναι ένα πρόσθετο ανοιχτού κώδικα [83] για την μηχανή Unreal Engine και χρησιμοποιεί την εφαρμογή Hyperplay [84] για την πρόσβαση των παιχνιδιών στο πορτοφόλι του χρήστη και την εκτέλεση των συναλλαγών. Με την σειρά του το Hyperplay εντοιχίζει το MetaMask. Αξίζει εδώ να σημειωθεί, πως το παιχνίδι θα μπορούσε να χρησιμοποιήσει απευθείας το MetaMask SDK, όμως αυτό δεν ήταν διαθέσιμο κατά τον χρόνο δημιουργίας της εργασίας, για την Unreal Engine [85].

Το πρόσθετο WEB3.Unreal προσφέρει μια σειρά από «εργαλεία» στην μορφή ασύγχρονων συναρτήσεων που μιλάνε με έξυπνα συμβόλαια του EVM. Το HyperPlay παρέχει μια γέφυρα προς τις υποδομές πορτοφολιών και είναι native εφαρμογή του υπολογιστή που απαιτεί εγκατάσταση. Δηλαδή, για την λειτουργία του WEB3.unreal πρόσθετου είναι απαραίτητη η εγκατάσταση του HyperPlay στον υπολογιστή του χρήστη και η είσοδος του χρήστη στο MetaMask, εντός της εφαρμογής.

Συγκεκριμένα, για την υλοποίηση της «γέφυρας» διασύνδεσης των δύο native εφαρμογών, το WEB3.unreal χρησιμοποιεί την πόρτα 9680 του localhost για να επικοινωνήσει με το Hyperplay μέσα από ασύγχρονα HTTP POST requests. Στην Unreal Engine 5 η επικοινωνία με εξωτερικές εφαρμογές στον ίδιο υπολογιστή χρησιμοποιώντας HTTP requests αποτελεί έναν από τους διάφορους υποστηριζόμενους τρόπους επικοινωνίας. Συγκεκριμένα, για την υλοποίηση της «γέφυρας» διασύνδεσης των δύο native εφαρμογών, το WEB3.unreal χρησιμοποιεί την πόρτα 9680 του localhost για να επικοινωνήσει με το Hyperplay μέσα από ασύγχρονα HTTP POST requests (αξίζει, εδώ, να σημειωθεί πως υποστηρίζονται διάφοροι τρόποι επικοινωνίας ανάμεσα στην μηχανή και εξωτερικές εφαρμογές στον ίδιο υπολογιστή, η μέθοδος επικοινωνίας με τα HTTP requests είναι απλά μία από αυτές). Για την επίτευξη αυτής της επικοινωνίας, το WEB3.unreal χρησιμοποιεί το Unreal Engine HTTP module και καλεί το αρχείο "Interfaces/IHttpRequest.h" και "Runtime/Online/HTTP/Public/Http.h" της μηχανής του παιχνιδιού εντός του αρχείου Web3RequestBuilder.cpp του πηγαίου κώδικα της επέκτασης [83]. Σε εφαρμογή, όταν ενεργοποιείται το Blueprint στοιχείο "Send Contract", που αντιστοιχεί στο USendContract και είναι UObject class στο αρχείο SendContract.h της επέκτασης, εκτελείται η συνάρτηση της μεθόδου Activate στο void USendContract::Activate(). Σε σειρά καλούνται InitializeRequest(), BuildRequest() και FinalizeRequest() εντός του αρχείου

Web3RequestBuilder.cpp με το `Request = FHttpModule::Get().CreateRequest();` να δημιουργεί το HTTP request. Έπειτα, το `Request->ProcessRequest();` στέλνει το request ασύγχρονα.

```

1 // Copyright (c) 2023 Game 7 DAO Ltd. All Rights Reserved.
2 #include "Endpoints/SendContract.h"
3
4 USendContract* USendContract::SendContract(
5     const UObject* WorldContextObject,
6     FString contractAddress,
7     FString functionName,
8     FString abi,
9     TArray<FString> params,
10    int32 gasLimit,
11    FString valueInWei,
12    int32 chainId) {
13
14    USendContract* SendContractInstance = NewObject<USendContract>();
15
16    SendContractInstance->RequestBuilder.ParamsVar = params;
17    SendContractInstance->RequestBuilder.ChainID = chainId;
18    SendContractInstance->RequestBuilder.ContractAddressVar = contractAddress;
19    SendContractInstance->RequestBuilder.FunctionNameVar = functionName;
20    SendContractInstance->RequestBuilder.GasLimitVar = gasLimit;
21    SendContractInstance->RequestBuilder.ABIVar = abi;
22    SendContractInstance->RequestBuilder.ValueInWeiVar = valueInWei;
23    SendContractInstance->RequestBuilder.OnCompleteDelegate.BindUObject(SendContractInstance, &UHyperplayAsyncRequest::OnResponse);
24
25    return SendContractInstance;
26 }
27
28 void USendContract::Activate()
29 {
30     RequestBuilder.ExecuteRequest();
31 }
32
33 void USendContract::ProcessResponse(FHttpResponsePtr Response, int32 statusCode) {
34     Super::ProcessResponse(Response, statusCode);
35     const FString ResponseText = Response->GetContentAsString();
36     OnResponseOutput.Broadcast(ResponseText, statusCode);
37     OnCompleted.Broadcast(ResponseText, statusCode);
38 }

```

Αρχείο πηγαίου κώδικα επέκτασης: `web3Unreal/Private/Endpoints/SendContract.cpp`

```

void ExecuteRequest()
{
    InitializeRequest();
    BuildRequest();
    FinalizeRequest();
};

```

Αρχείο πηγαίου κώδικα επέκτασης: web3Unreal/Public/Web3RequestBuilder.h

```
void FWeb3BaseRequest::InitializeRequest()
{
    Request = FHttpModule::Get().CreateRequest();
}
```

```
void FWeb3BaseRequest::FinalizeRequest() const
{
    Request->SetURL(RequestUrl);
    Request->OnProcessRequestComplete() = OnCompleteDelegate;
    Request->SetContentAsString(HttpContentString);
    UE_LOG(LogTemp, Warning, TEXT("Processing request in web3baserequest finalize request for %s"), *HttpContentString)
    Request->ProcessRequest();
}
```

Αρχείο πηγαίου κώδικα επέκτασης: web3Unreal/Private/Web3RequestBuilder.cpp

6.3.6 Επιλογή τύπου παιχνιδιού

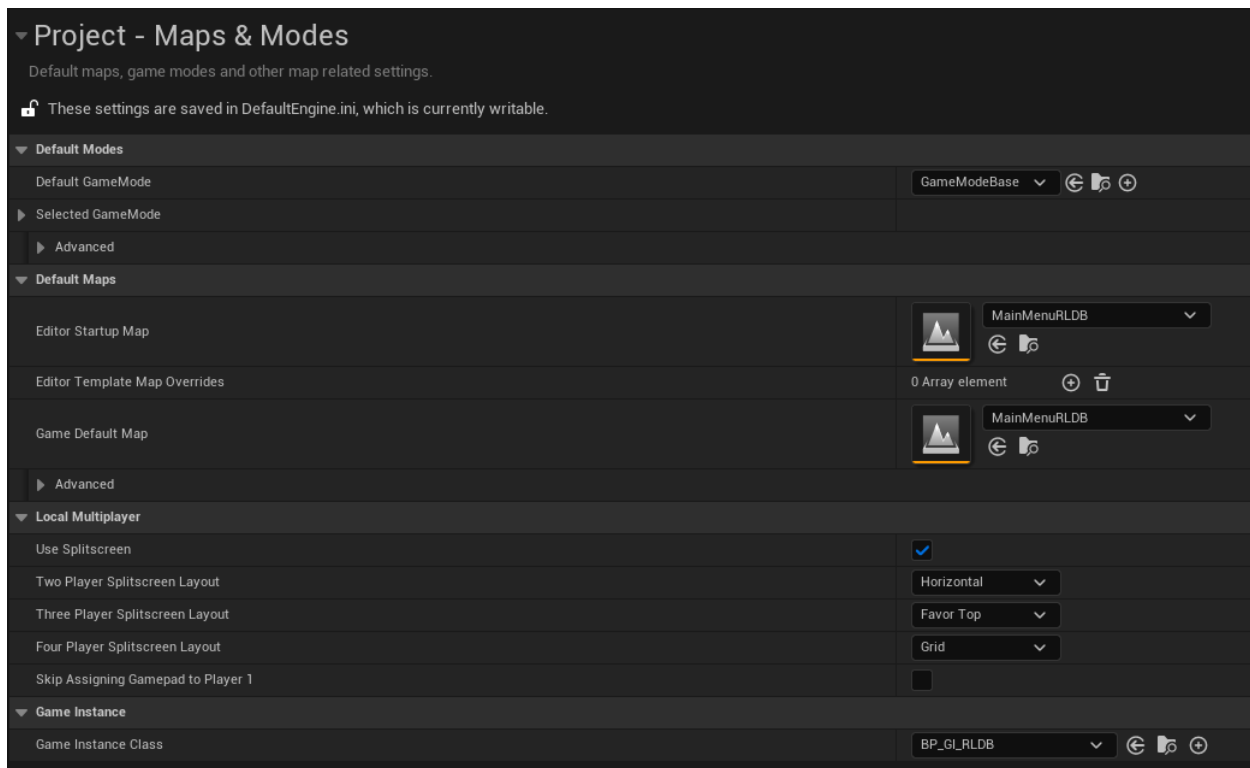
Για την κατασκευή του παιχνιδιού δημιουργήθηκε ένα project μέσα από το πρόσθετο Roguelike Deckbuilder Toolkit, το οποίο χρησιμοποιήθηκε σαν βάση για την μορφοποίηση του πυρήνα του παιχνιδιού και εν συνεχεία, την εφαρμογή του συστήματος με τα μοναδικά τεκμήρια. Πάνω σε αυτή την δομή, έγιναν πολλές μετατροπές και επιπροσθέσεις μέχρι το παιχνίδι να μπορεί να διαχειριστεί όλες τις λειτουργίες που χτίστηκαν εντός του έξυπνου συμβολαίου. Το ίδιο το παιχνίδι αποτελεί ένα Alpha μηχανικό πρωτότυπο για την επίδειξη των λειτουργιών των έξυπνων συμβολαίων, κάτω από τις αντισυμβατικές συνθήκες ενός Rogue-lite Deckbuilder παιχνιδιού.

Επιλέχθηκε το είδος Roguelite-DeckBuilder, διότι απλοποιεί την πολύπλοκη φύση του μέσου παιχνιδιού που χτίζεται στην Unreal Engine και ταυτόχρονα εμπεριέχει, η φύση του RogueLite, πολλά παράλληλα με λειτουργίες της Βιομηχανίας 4.0. Συγκεκριμένα, ένα Roguelite παιχνίδι αποτελείται από πίστες τοποθετημένες σε σειρά με στόχο την ολοκλήρωση όλων των επιπέδων σε μία ζωή. Ο θάνατος του παίκτη είναι μόνιμος και το παιχνίδι επανεκκινεί την διαδικασία ολοκλήρωσης των επιπέδων. Μέσα από κάθε θάνατο ο παίκτης κρατάει ορισμένους πόρους (οι οποίοι θυμίζουν πόρους σε παιχνίδι ρόλων), τους οποίους μπορεί να χρησιμοποιήσει στις επόμενες προσπάθειες του να φτάσει στην τελική πίστα. Οι πόροι που παραμένουν μετά τον θάνατο του παίκτη, βοηθούν τον παίκτη να κερδίσει το παιχνίδι. Το παιχνίδι όμως είναι και DeckBuilder, ο παίκτης πρέπει στρατηγικά να χτίσει μια τράπουλα, μέσα από βραβεία ολοκλήρωσης πίστας, για να δυναμώσει τον χαρακτήρα του και τα χαρακτηριστικά του.

Αυτή η επανάληψη, μεταποίηση και βελτιστοποίηση μέσα από την χρήση νέων εικονικών αγαθών και τα μετα-πραγματικά στοιχεία ενός εικονικού κόσμου που επανα-αρχικοποιείται, αλλά αφήνει πίσω του οντότητες από προηγούμενες εκδόσεις του (instances), παρέχουν τις σωστές συνθήκες για την τέλεση της έρευνας. Συγκεκριμένα, η επαναληψιμότητα και η βελτιστοποίηση μέσα από την ανάλυση και την επέκταση θυμίζουν τις δράσεις ενός κυβερνητικού συστήματος. Τα εικονικά αγαθά τα οποία εισάγονται ανά πίστα, τραβούν παράλληλα με τον ταχύ ρυθμό ένταξης μοναδικών διαλειτουργικών αγαθών σε ευφυή συστήματα. Οι οντότητες που παραμένουν μετά το πέρας του παιχνιδιού και το μεταποιούν, προσφέρουν ένα πλούσιο πλαίσιο για την έρευνα των δυνατοτήτων των δυναμικών μεταλλάξιμων παιχνιδιών με διαλειτουργικά στοιχεία. Η δυνατότητα του παιχνιδιού να μεταποιηθεί και να λειτουργήσει κάτω από νέα απροσδιόριστα δεδομένα (μια ασαφή τράπουλα) καθιστά το παιχνίδι ένα τέλειο πεδίο δοκιμών για την ένταξη ξένων οντοτήτων από το Blockchain και τις επιπτώσεις αυτών.

6.3.7 Μορφοποίηση παιχνιδιού μέσα από την μηχανή Unreal Engine

Η αρχικοποίηση ενός παιχνιδιού εντός της μηχανής παίρνει μέρος μέσα από τον προσδιορισμό των ρυθμίσεων του έργου (Edit -> Project Settings...). Μέσα στα πλαίσια ρύθμισης, ένα θεμέλιο στοιχείο του έργου αποτελεί η ρύθμιση του “Default GameMode”, του “Editor Startup Map” και ο προσδιορισμός του “Game Instance Class”.

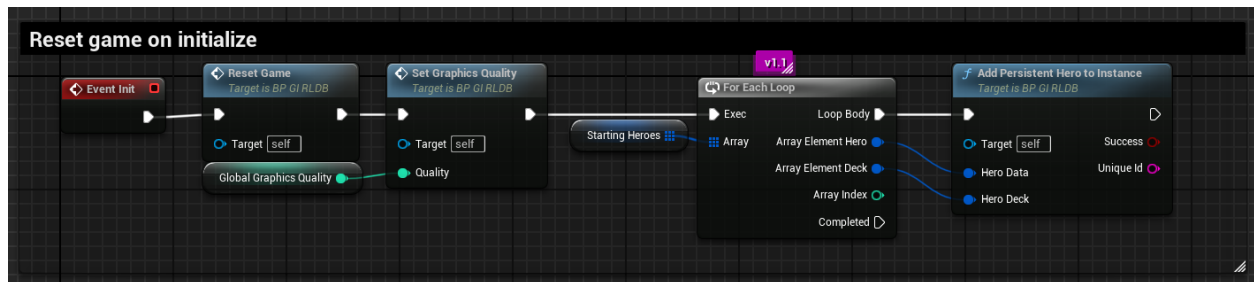


Ρυθμίσεις μηχανής παιχνιδιού και αρχικοποίηση βασικών στοιχείων παιχνιδιού

Το Editor Startup Map, αποτελεί την λειτουργική αρχή του παιχνιδιού ως προς την εμπειρία του χρήστη. Εκεί, ξεκινάει η εμπειρία του παιχνιδιού προς το παιχνίδι και τα συστήματα του και πολλές φορές αυτή η αρχή είναι το κεντρικό Menu.

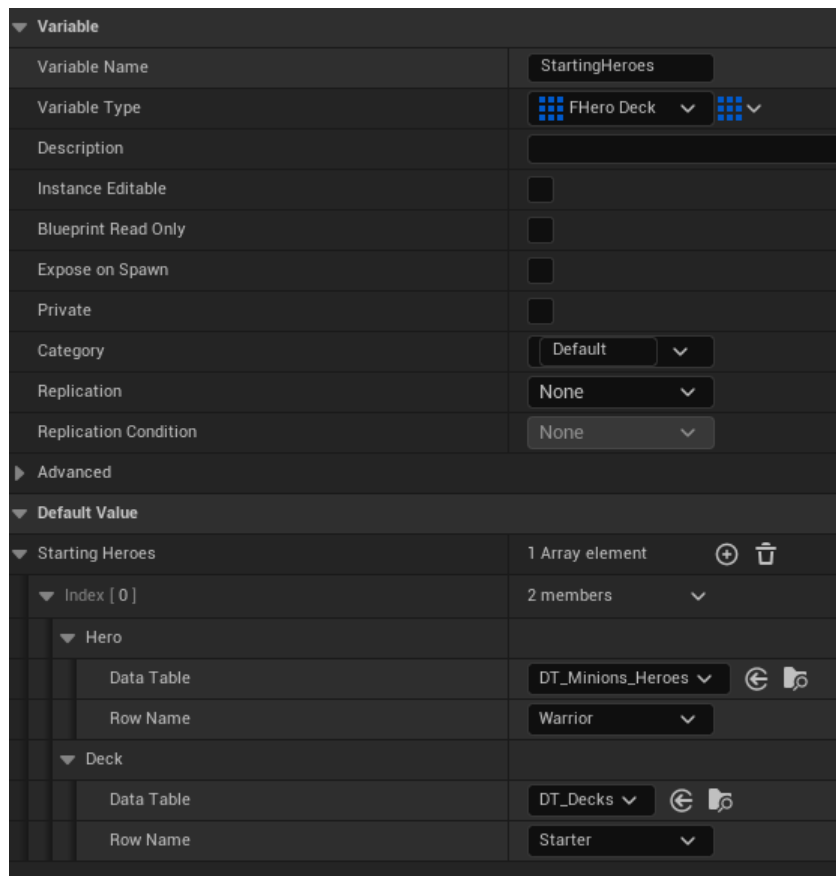
Το Game Instance Class αποτελεί τον λειτουργικό πυρήνα του παιχνιδιού και χρησιμοποιείται για να αρχικοποιήσει τις σημαντικές ρουτίνες λειτουργίας του παιχνιδιού, αλλά για την αρχικοποίηση των Global Variables μεταβλητών. Οι τιμές και τα αντικείμενα που ορίζονται εντός, διατηρούνται μέχρι τον τερματισμό της εφαρμογής και αρχικοποιούνται κατά την αρχή της εκτέλεσης της εφαρμογής. Για την πρόσβαση των τιμών απαιτείται casting των τιμών από Blueprints.

Στην περίπτωση του παιχνιδιού που κατασκευάστηκε, το Game Instance εμπεριέχει την ακόλουθη λογική ρουτίνα που ξεκινάει με το Event της αρχικοποίησης της εφαρμογής. Με μια έννοια, η ακόλουθη ρουτίνα αποτελεί την κεντρική συνάρτηση αρχικοποίησης της λογικής Gameplay.



Η ακόλουθη λογική ρουτίνα εκτελείται κατά την εκκίνηση του παιχνιδιού και ορίζει το επίπεδο γραφικών.

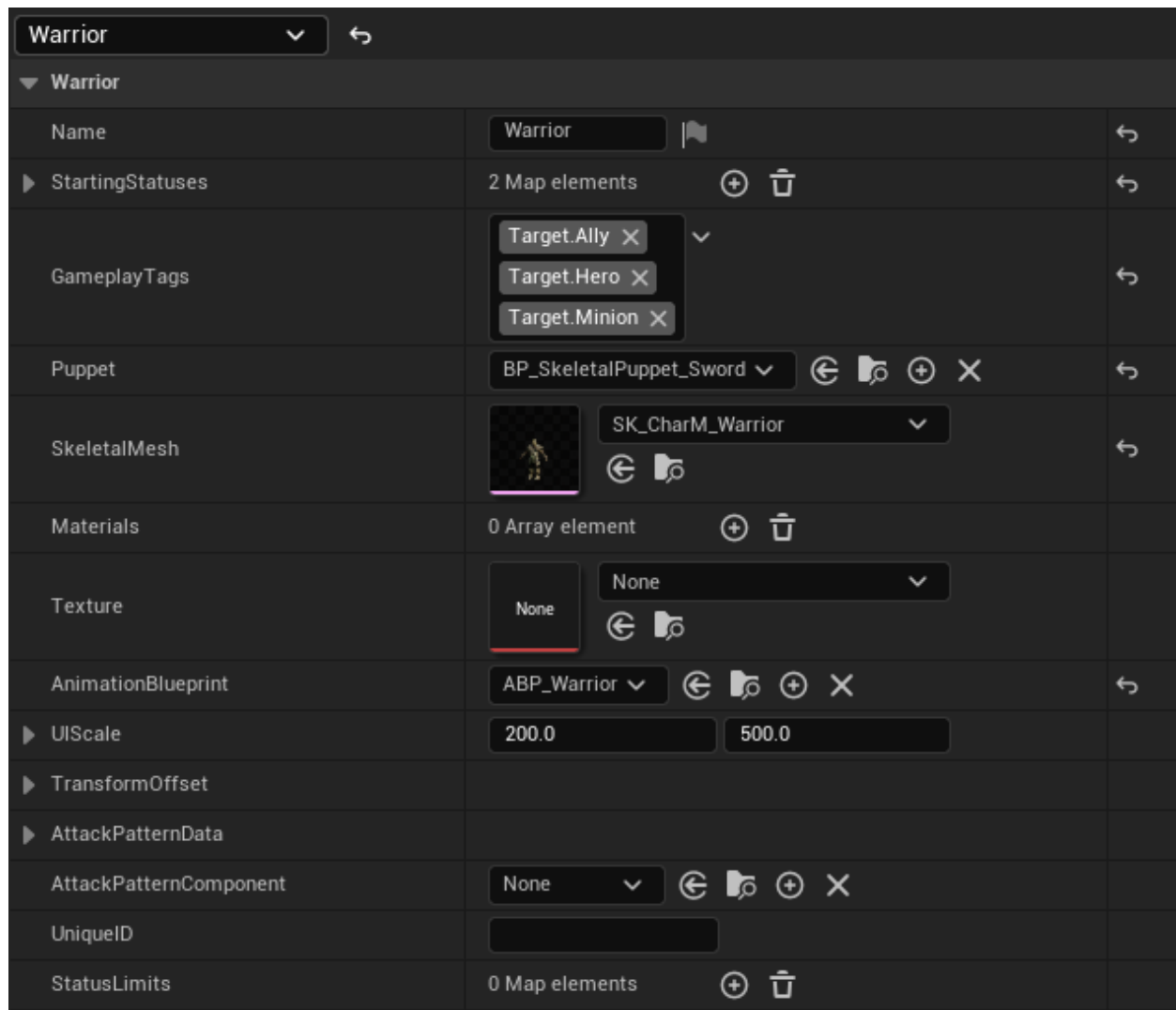
Ένα από τα βασικότερα «αντικείμενα» του παιχνιδιού, αποτελούν τα δεδομένα τα οποία εμπεριέχονται στα Data Tables DT_Minions_Heroes, DT_Decks, DT_MapEvents, DT_Cards και DT_CombatEncounters. Τα δεδομένα αυτά δομούν το όλο του παιχνιδιού, των συμβάντων και των αντικειμένων που εμπεριέχονται τόσο μέσα στο σώμα του παιχνιδιού όσο και στις κάρτες που μοιράζεται το παιχνίδι με το Blockchain.



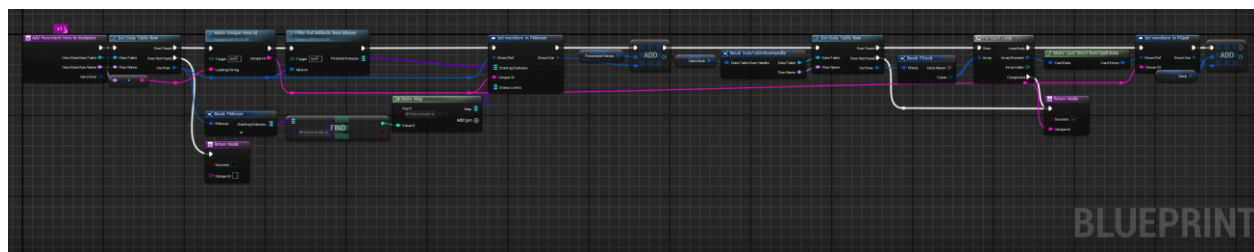
Ο πίνακας(Array) StartingHeroes μέσα από τον οποίο επιλέγεται ο χαρακτήρας και η τράπουλα.

Row No	Name	StartingStatuses	GameplayTags	Puppet	SkeletalMesh	Materials	Texture	AnimationBlueprint	UIScale	TransformOffset	AttackPatternData	AttackPat
1	Warrior	Warrior	/Script/Engine.BlueprintGer {"GameplayTags":{"Ta /Script/Engine.Blu /Script/Engine.Skelet				None	/Script/Engine.AnimBluepri {"X": 200, "Y {"Rotation": {"X": 0, "Y {"DataTable": "None", "I None				
2	Manny	Manny	/Script/Engine.BlueprintGer {"GameplayTags":{"Ta /Script/Engine.Blu /Script/Engine.Skelet				None	/Script/Engine.AnimBluepri {"X": 200, "Y {"Rotation": {"X": 0, "Y {"DataTable": "None", "I None				
3	Mage	Mage	/Script/Engine.BlueprintGer {"GameplayTags":{"Ta /Script/Engine.Blu, None				None	/Script/Engine.AnimBluepri {"X": 300, "Y {"Rotation": {"X": 0, "Y {"DataTable": "None", "I None				

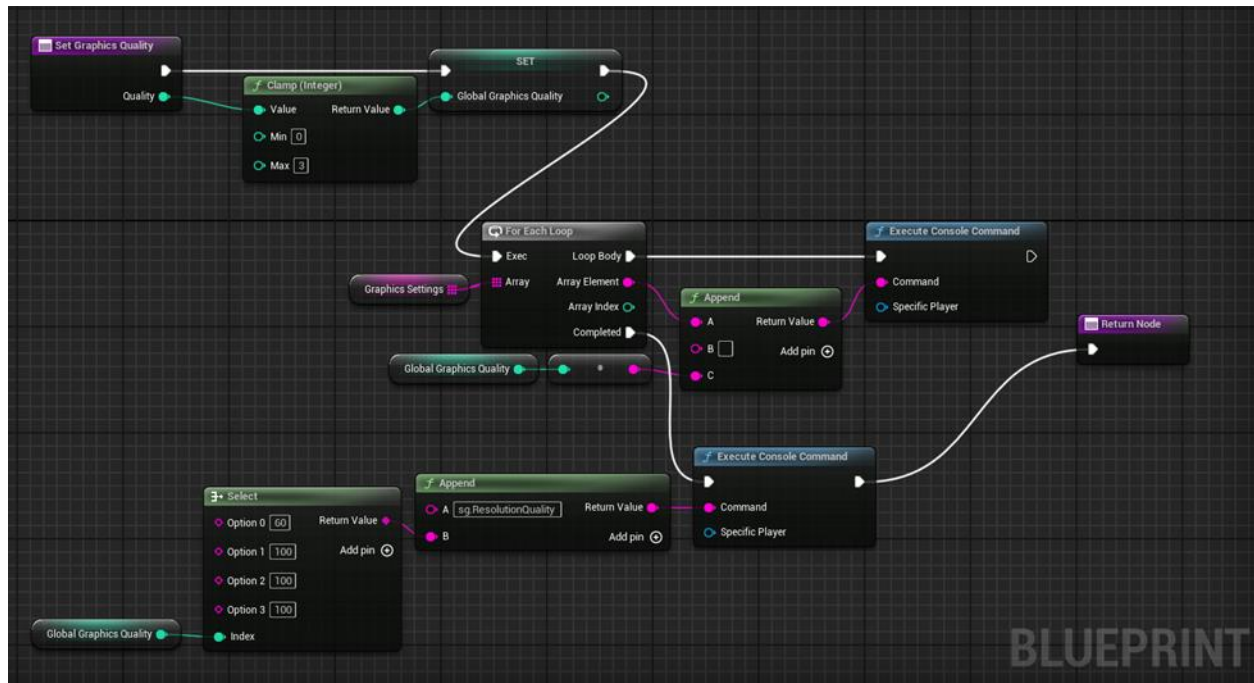
Data Table DT_Minions_Heroes



Η δομή των Data Tables, όπως το βασικό Data Table DT_Minions_Heroes.

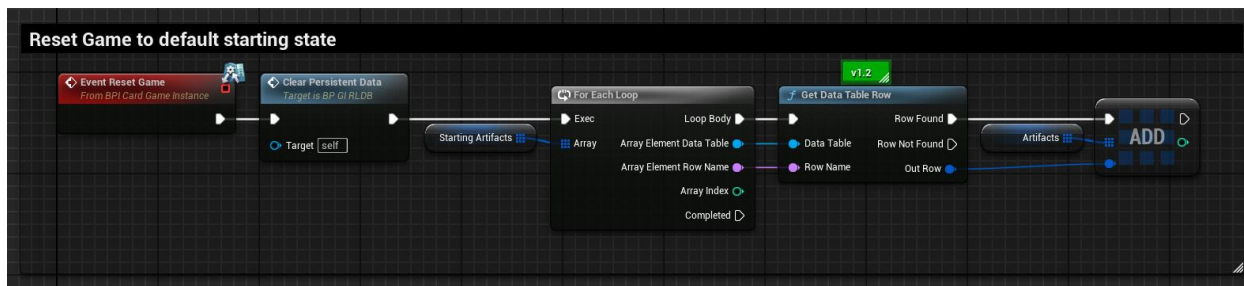


Στο Event Init εκτελείται η συνάρτηση “Add persistent hero to instance” της οποίας ο κώδικας παρουσιάζεται στην φωτογραφία. Η συνάρτηση θέτει τα στοιχεία από τα από το DT_Minions_Heroes και DT_Decks σε μεταβλητές του Game Instance.

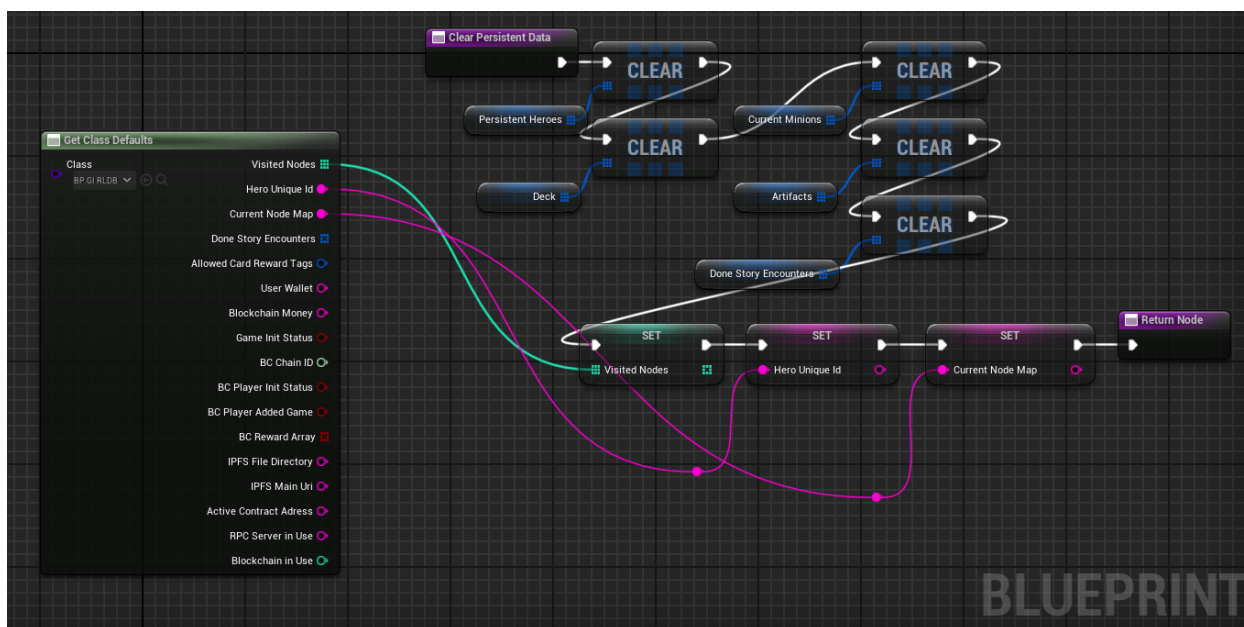


Variable	
Variable Name	GraphicsSettings
Variable Type	String
Description	
Category	Default
Advanced	
Default Value	
Graphics Settings	5 Array elements
Index [0]	sg.ShadowQuality
Index [1]	sg.ViewDistanceQuality
Index [2]	sg.AntiAliasingQuality
Index [3]	sg.PostProcessQuality
Index [4]	sg.TextureQuality

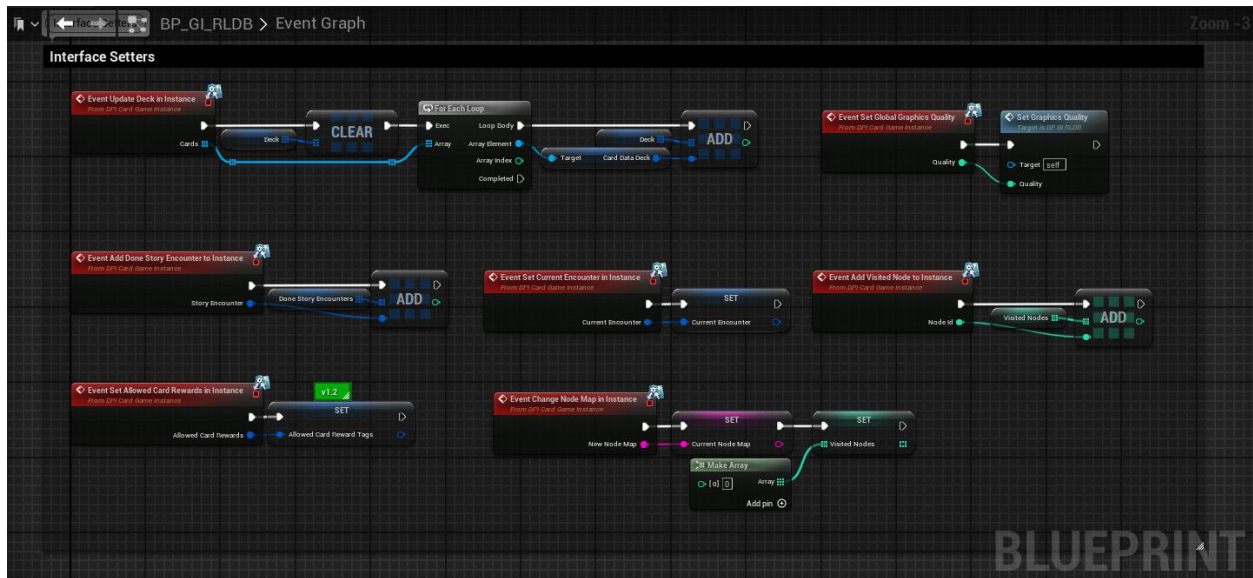
Στις φωτογραφίες παρουσιάζεται μια ακόμα συνάρτηση του Event Init του Game Instance, όπου επιδεικνύεται ο τρόπος με τον οποίο ρυθμίζονται τα γραφικά μέσα από τις μεταβλητές κονσόλας και τις CVAR εντολές του array GraphicSettings. [94].



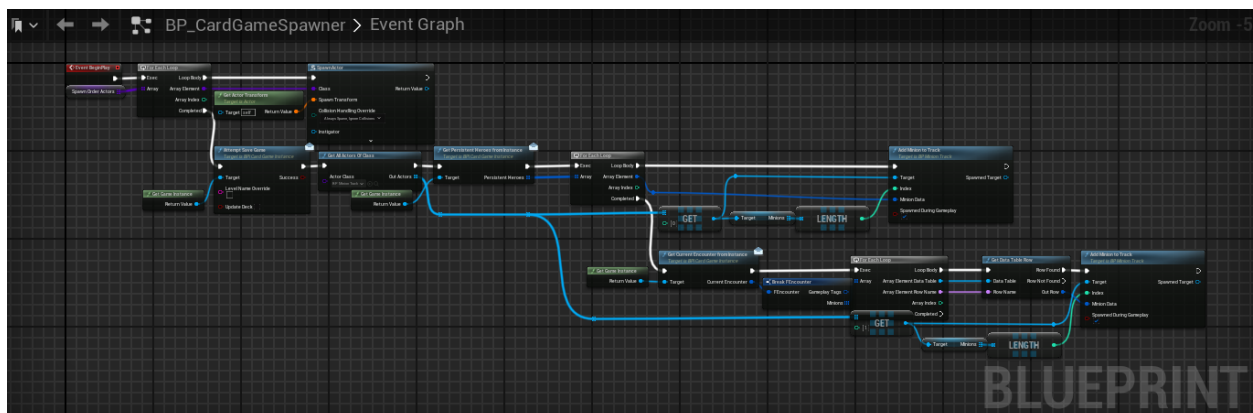
Σημαντικό στοιχείο ενός παιχνιδιού, ιδιαίτερα ενός RogueLite παιχνιδιού, αποτελεί η ιδιότητά τους να μπορούν να γυρίσουν την κατάσταση τους πίσω στο αρχικό της σημείο και να προσφέρουν την ίδια εμπειρία παιχνιδιού ξανά, σαν μια εξομοίωση, της οποίας η ύπαρξη λειτουργεί «διακοπτικά» (έρχεται και φεύγει από ύπαρξη). Αυτή η ιδιότητα της επανα-αρχικοποίησης προσφέρει αρκετές δυνατότητες δημιουργίας ανεπτυγμένων μετα-υπαρξιακών δομών εντός του metaverse. Αυτή η δυνατότητα παίρνει μέρος μέσα από ρουτίνες όπως εκείνη της φωτογραφίας.



Η λειτουργία της συνάρτησης Clear Persistent Data. Μέσα από την εκκαθάριση των τιμών του Game Instance το παιχνίδι μπορεί να ξαναρχίσει από το σημείο μηδέν. Όμως, μέσα από μια ρουτίνα Blueprint που μιλάει με το EVM και ένα ανάλογο Smart Contract, ο χρήστης μπορεί να εισάγει στοιχεία από το τέλος του παιχνιδιού στην αρχή του -μια διαδικασία παρόμοια με εκείνη της δομής των RogueLite προς τα στοιχεία «μακροχρόνιου σωσίματος».



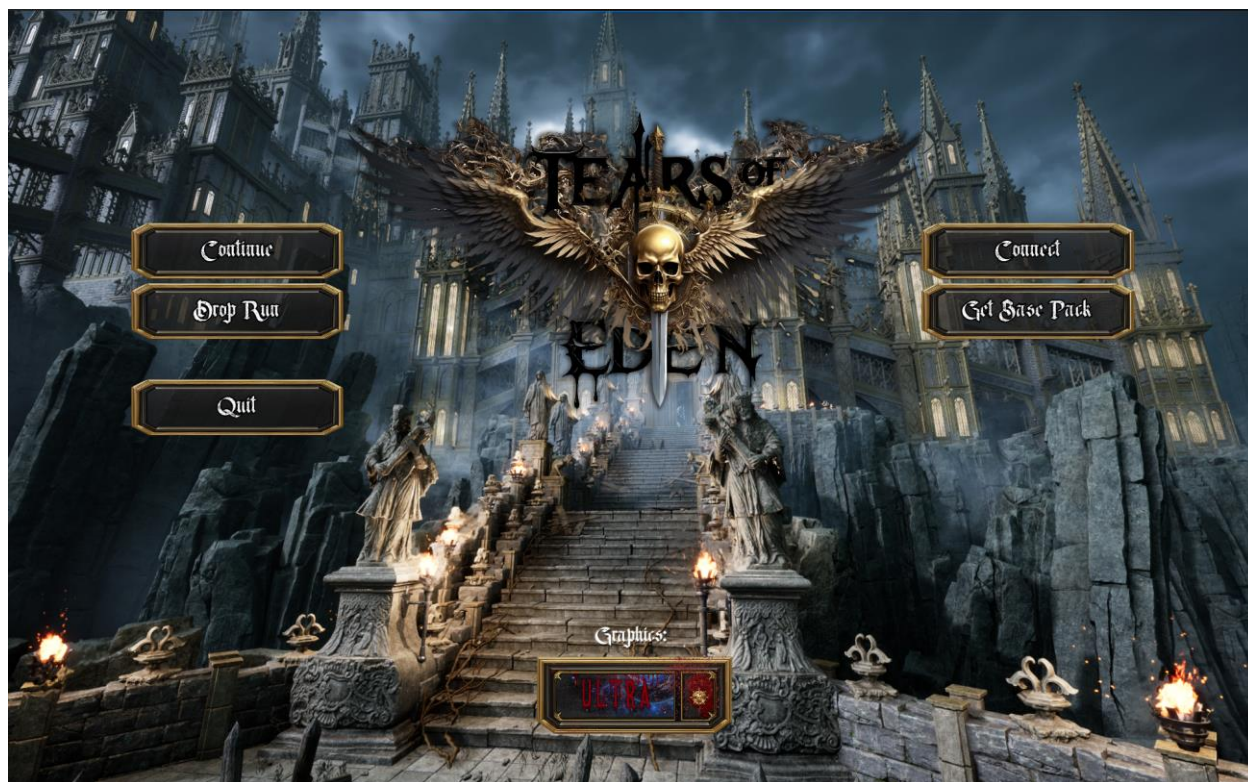
Λοιπά στοιχεία του Game Instance που χρησιμοποιούνται για την αρχικοποίηση των επιπέδων, των λειτουργικών στοιχείων του παιχνιδιού και την διασφάλιση της συνέχειας. Πρέπει να βρίσκονται στο Game Instance γιατί είναι συνδεδεμένα στοιχεία ανάμεσα σε διάφορα επίπεδα.



Σημαντικό Blueprint των λειτουργιών του παιχνιδιού καθώς λειτουργεί ως spawner.



Το παιχνίδι ξεκινάει από το επίπεδο MainMenuRLDB, το οποίο καθορίζεται εντός του Project Settings > Project – Maps & Modes. Το οποίο επίπεδο MainMenuRLDB αποτελείται μόνο από το Widget Blueprint WBP_MainMenu_RLDB. Ακολούθως παρουσιάζεται το κεντρικό μενού...

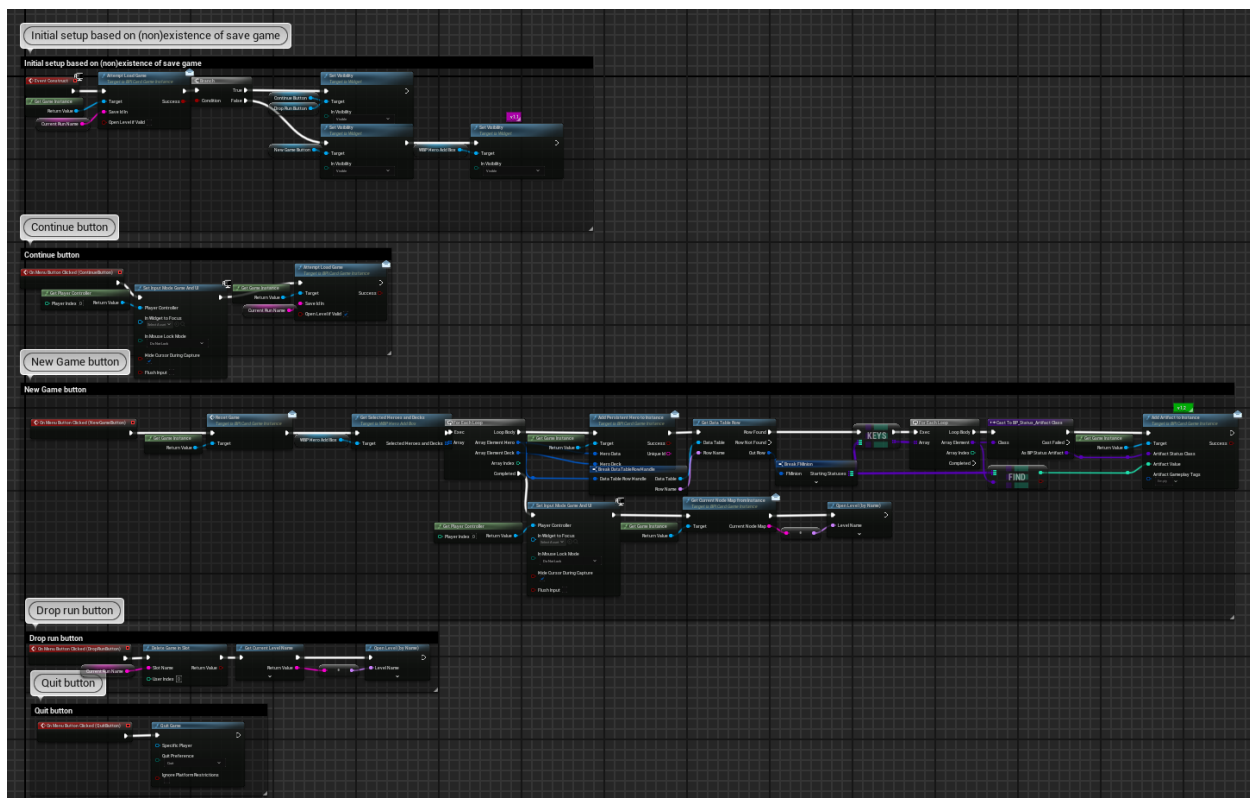


Το κεντρικό μενού του παιχνιδιού.

Το μενού αποτελείται από 3 στήλες. Στο κέντρο βρίσκονται οι ρυθμίσεις για την αλλαγή του επιπέδου γραφικών του παιχνιδιού. Στην δεξιά στήλη βρίσκονται οι απαραίτητες λειτουργίες σύνδεσης και συναλλαγής με το Blockchain, οι οποίες εμφανίζονται πριν ξεκινήσει ο παίκτης το παιχνίδι, και στην αριστερή στήλη βρίσκονται οι βασικές λειτουργίες εκτέλεσης του παιχνιδιού.

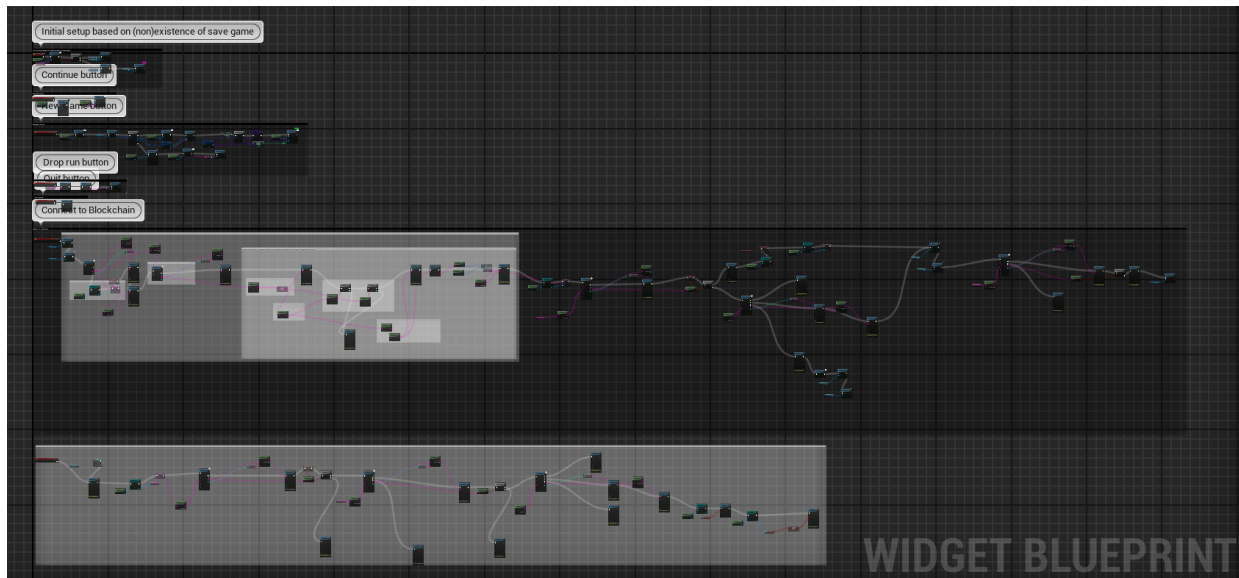
Το φόντο αποτελεί σκηνή από επίπεδο του παιχνιδιού και το λογότυπο έχει κατασκευαστεί στο Photoshop και το Stable Diffusion με inpainting και 4 διαφορετικές φωτογραφίες. Το όνομα του παιχνιδιού είναι «Tears of Eden».

Το μενού στο 'πίσω μέρος' του (Back End – Graph) έχει κάποιες βασικές λειτουργίες για την εκτέλεση του παιχνιδιού έξω από το πλαίσιο χρήσης του Blockchain



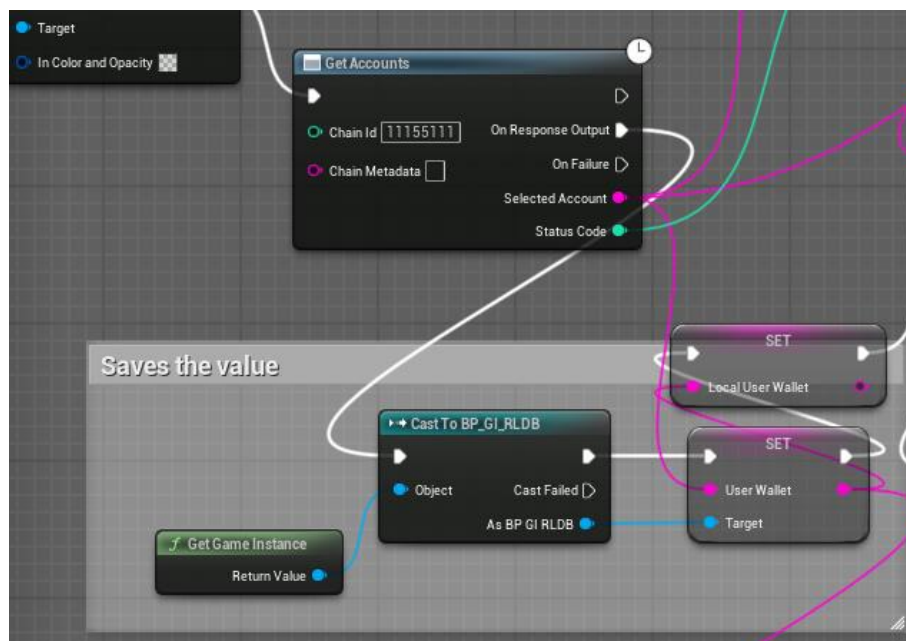
Συναρτήσεις των λειτουργιών της αριστερής διεπαφής χρήστη στο κεντρικό Menu.

Έξω, όμως, από τα πλαίσια των κλασικών λειτουργιών έχουν προστεθεί μεγάλες ρουτίνες εκτέλεσης κώδικα μέσα από τα Blueprints για την διασύνδεση με το Hyperplay και εν συνέχεια το Blockchain.

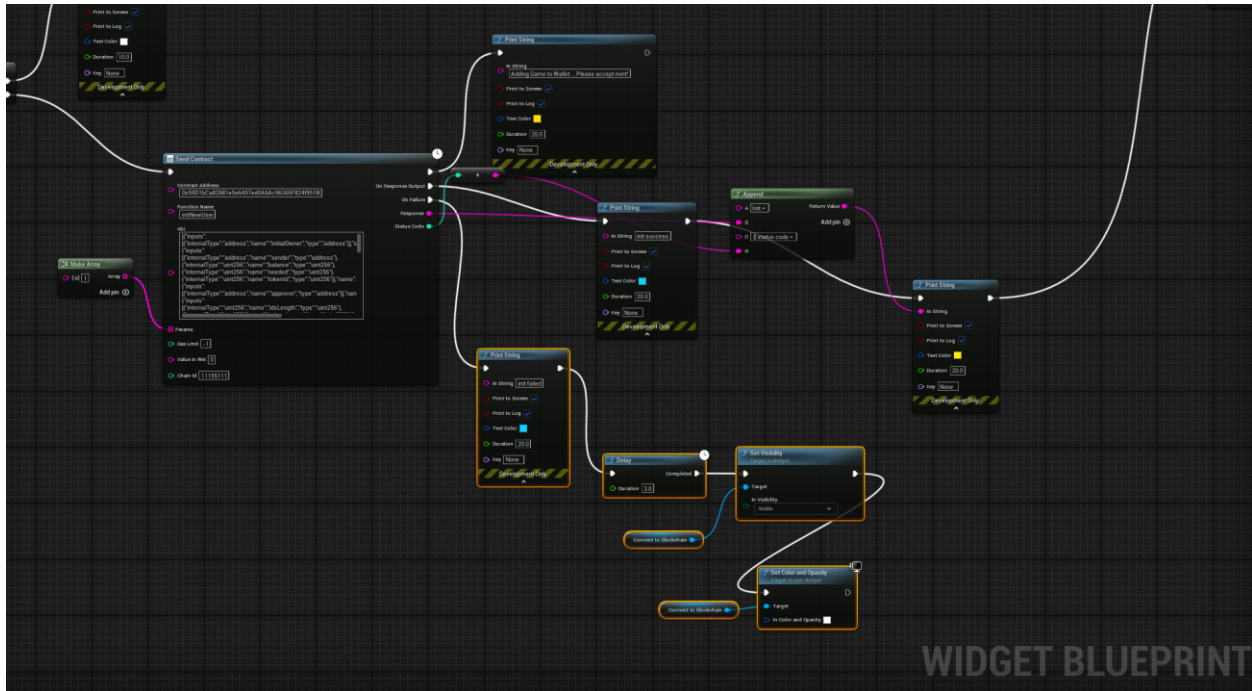


Σε αυτή την εικόνα παρουσιάζονται οι δύο μεγάλες ρουτίνες για την διασύνδεση του παιχνιδιού με το σύμβολο, την επεξεργασία του κειμένου της απάντησης HTTP response και την αρχικοποίηση των λειτουργιών του παιχνιδιού.

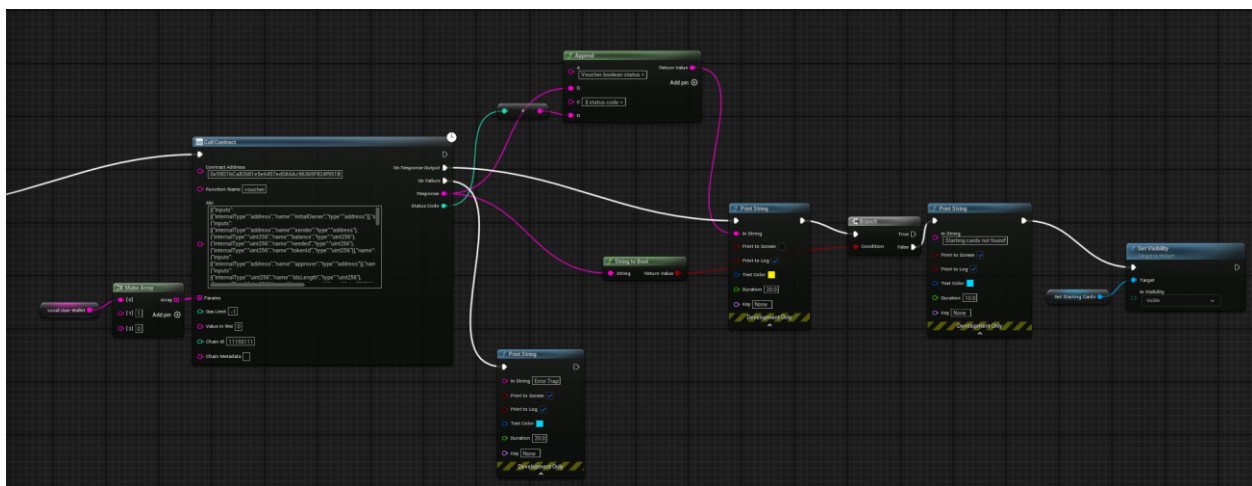
Αναλυτικά για την διασύνδεση:



[Μέρος 1] Λαμβάνεται ο λογαριασμός χρήστη από το HyperPlay.

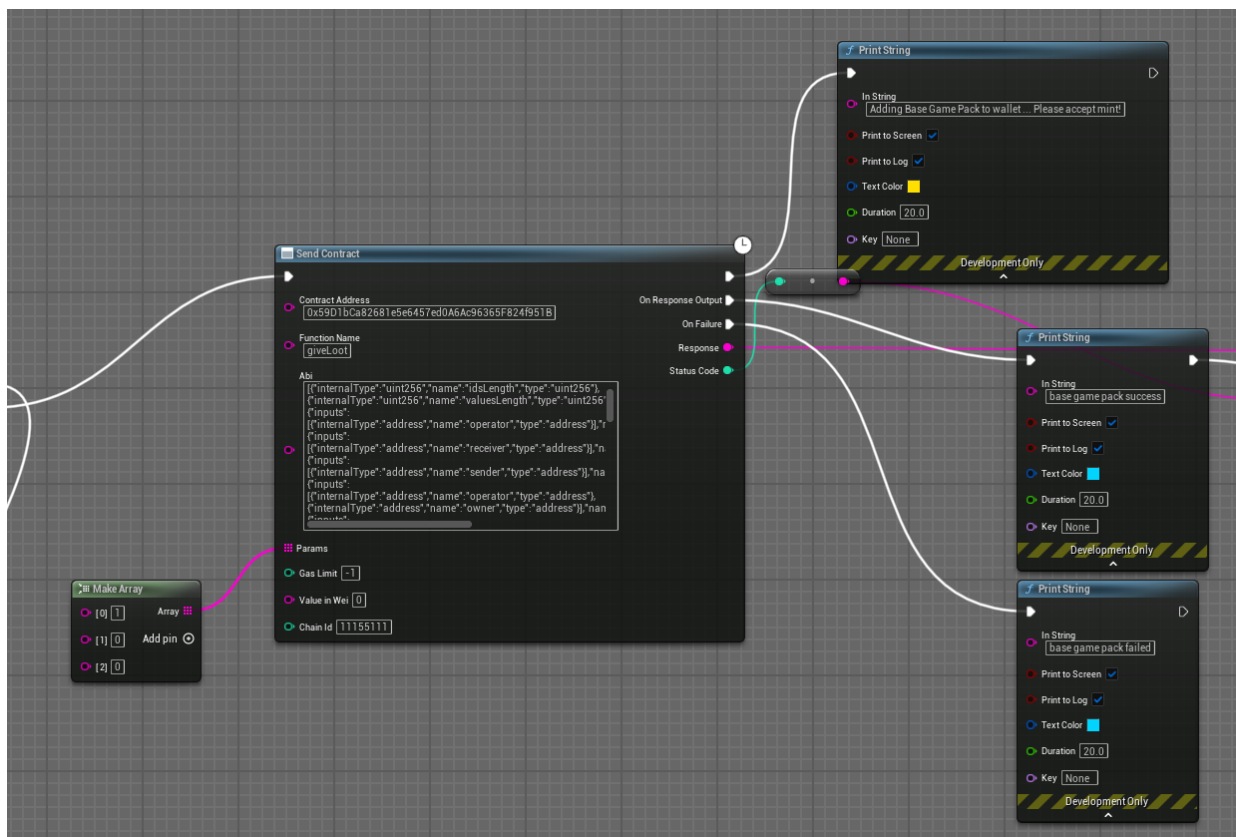


[Μέρος 5] Στην περίπτωση που ο χρήστης δεν έχει συνδεθεί, καλείται συναλλαγή από το συμβόλαιο, την οποία και πρέπει να ολοκληρώσει μέσα από το Hyperplay.



[Μέρος 6] Στην συνέχεια το παρόν σύστημα προσφέρει πολλές δυνατότητες προς ενδιαφέροντα σενάρια παιχνιδιού. Εδώ, δίνεται η δυνατότητα στον χρήστη να αποκτήσει κάποιες αρχικές «κάρτες - δώρο» με την αρχική είσοδο του στο παιχνίδι και 1000 νομίσματα «mainCurrency» από το εσωτερικό ανταλλάξιμο νόμισμα του παιχνιδιού. Τα ίδια νομίσματα χρησιμοποιούνται για να περιορίσουν την επαφή του χρήστη με το δίκτυο και ταυτόχρονα παρουσιάζουν ένα κόστος

προς τις συναλλαγές, αφού μπορεί να δοθεί η δυνατότητα στον χρήστη να αγοράσει περισσότερα, μέσα από μια μεταφορά του κεντρικού νομίσματος του Blockchain προς ένα συγκεκριμένο πορτοφόλι ή προς το σύμβολο. Στην εικόνα, ελέγχεται κατά πόσο ο χρήστης έχει εξαργυρώσει αυτό το πρωταρχικό «κουπόνι» (voucher στο smart contract) των αρχικών καρτών. Στην περίπτωση που δεν έχει εξαργυρώσει το κουπόνι αυτό, η επιλογή για να το εξαργυρώσει γίνεται ορατή στο κεντρικό μενού.



[Μέρος 7] Ο παίκτης, έπειτα, μπορεί να εξαργυρώσει το κουπόνι της μοναδικής συναλλαγής που του δίνει το αρχικό πακέτο καρτών. Αυτό παίρνει μέρος μέσα από την συνάρτηση `giveLoot` στο έξυπνο σύμβολο.

Κατά την είσοδο του χρήστη στο παιχνίδι, αφού συνεχίσει ή ξεκινήσει νέο instance του κόσμου του παιχνιδιού, ο παίκτης εισέρχεται στον κόσμο του παιχνιδιού μέσα από τον χάρτη NodeMap.



Ο χώρος παιχνιδιού εντός του χάρτη nodemap. Στην σκηνή φαίνεται ένα γραφείο βικτωριανής εποχής με έναν χάρτη πάνω του. Ο χρήστης μπορεί να διαδράσει με τον χάρτη και να εισέλθει στις επιθυμητές πίστες.

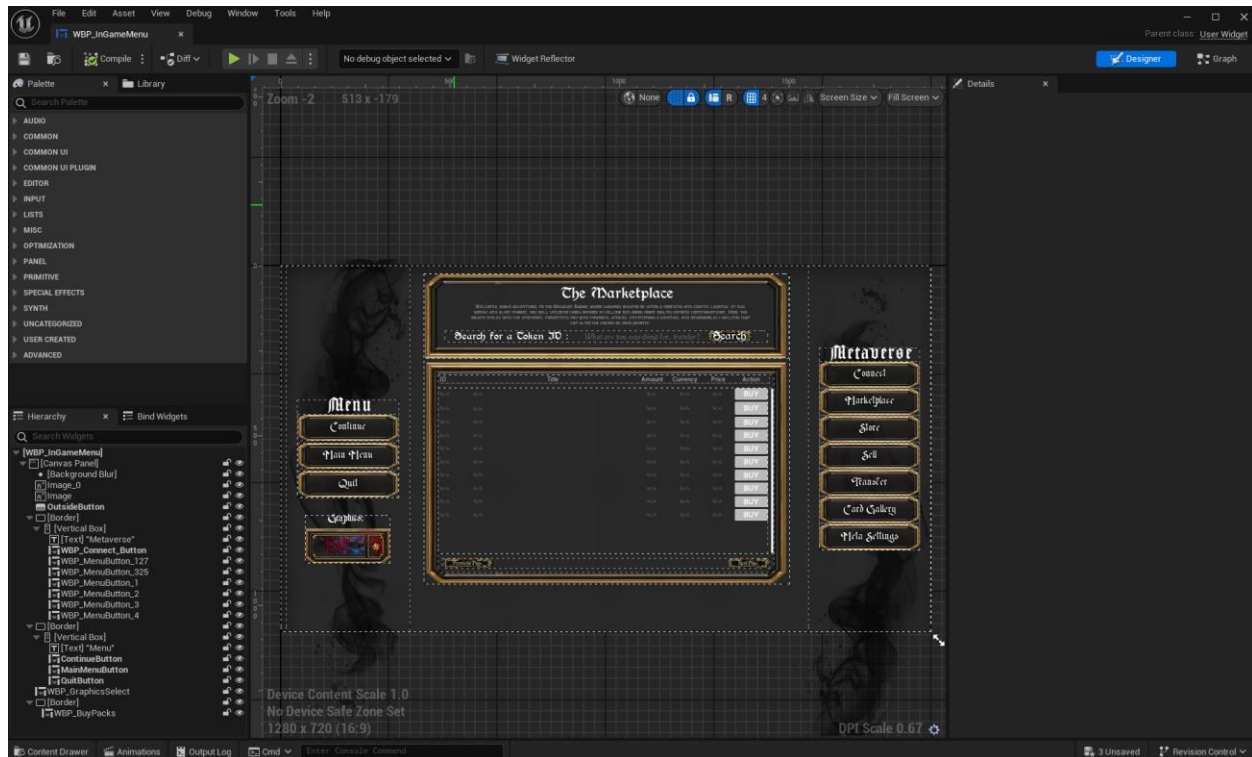
Εντός του χάρτη nodemap ο παίκτης κλείνεται να επιλέξει ένα από τα διαθέσιμα μονοπάτια. Οι προηγούμενοι «προορισμοί»/ επίπεδα από τα οποία έχει περάσει σημαδεύονται με ένα «X», για να δίνουν την αίσθηση της προόδου και να του υπενθυμίζουν τα προηγούμενα επιτεύγματα του, εντός του χώρου του παιχνιδιού. Το επίπεδο αυτό αποτελεί ενδιάμεσο ανάμεσα στις πίστες.

Εφόσον ο παίκτης δεν βρίσκεται πλέον εντός του κεντρικού Menu, η κύρια διεπαφή ρυθμίσεων (pause menu) και πρόσβασης στο έξυπνο συμβόλαιο και στο κατάστημα, μπορεί να ανοίξει.



Pause menu εντός του παιχνιδιού.

Το Menu αποτελεί μια επικάλυψη (overlay) πάνω από το παιχνίδι και ανοίγει με το κουμπί ESC – Escape του πληκτρολογίου. Στα αριστερά της διεπαφής χρήστη δίνεται πρόσβαση προς το κεντρικό μενού, τις ρυθμίσεις γραφικών και το «κλείσιμο» του overlay ή τον τερματισμό του παιχνιδιού. Στα δεξιά δίνεται πρόσβαση στα υπο-menu (submenus) που έχουν triggers ορατότητας, και μόνο ένα από αυτά μπορεί να παραμένει ορατό στο κέντρο του παραθύρου.



Διεπαφή επεξεργασίας Widget Blueprint. Το Widget της φωτογραφίας είναι το Pause menu με το Marketplace ανοικτό.

Εντός της δεξιάς στήλης Metaverse της διεπαφής προσφέρεται ένα μεγάλο πλήθος από δυνατότητες, που γίνονται εφικτές χωρίς κώδικα και χωρίς την παρουσία Back-End διακομιστή για την υποστήριξη τους. Όλες οι συναλλαγές, ο αποθηκευτικός χώρος και οι λειτουργίες δικτύου προσφέρονται μέσα από το EVM του Ethereum.

Ακολούθως, παρουσιάζεται το Marketplace για την αγορά καρτών από άλλους χρήστες. Τα αντικείμενα που πωλούνται αποτελούν γνωστά ή άγνωστα στοιχεία για τον κόσμο του παιχνιδιού και τα στοιχεία τους δυναμικά λαμβάνονται την ώρα εκτέλεσης της εφαρμογής.



Το Pause Menu ανοικτό με το Widget του Marketplace.

Η δομή του συστήματος των Token και του έξυπνου συμβολαίου διαμορφώθηκε ως εξής :

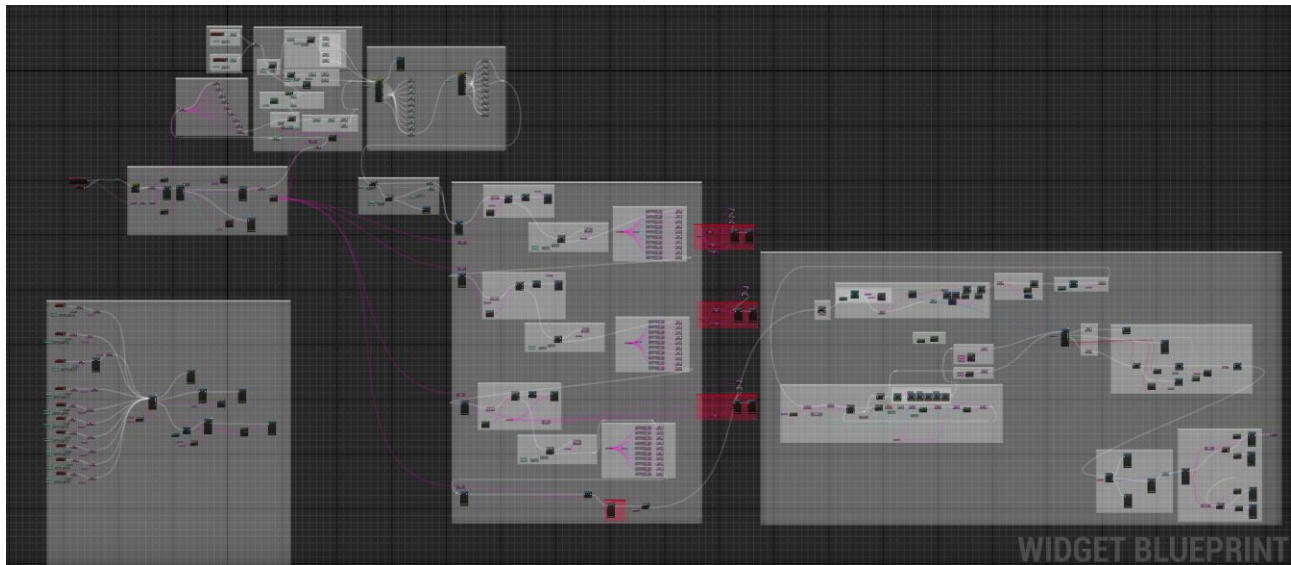
- Το σύμβολαιο υποστηρίζει την μετακίνηση και πώληση νομισμάτων ανάμεσα σε πολλαπλά παιχνίδια.
- Το παιχνίδι υποστηρίζει την δυνατότητα διαχείρισης άγνωστων νομισμάτων, για εκείνο.
- Τα στοιχεία της οντότητας λαμβάνονται μέσα από το ID της και κατεβαίνουν σε πραγματικό χρόνο από το IPFS και εξάγονται μέσα από το .json αρχείο του κάθε ξεχωριστού νομίσματος.
- Οι διαδικασίες που εκτελούνται αποτελούν δυναμικά ασαφή στοιχεία και απαιτούν μηχανισμούς δυναμικού προγραμματισμού (Reflection), οι οποίοι παρέχονται μέσα από την χρήση του συστήματος Blueprint.
- Οποιοδήποτε νόμισμα μπορεί άμεσα να συναλλαχθεί για οποιοδήποτε πλήθος άλλου νομίσματος, είτε το νόμισμα αυτό αποτελεί NFT είτε ένα κοινό νόμισμα τύπου ERC-20.
- Τα μοναδικά τεκμήρια δεν δένονται στην συναλλαγή και την ιδιοκτησία αλλά στα μοναδικά στοιχεία ή «σώμα» που φέρει το κάθε ον.
- Για τα μεταλλάξιμα όντα που μπορούν να αλλάζουν τα χαρακτηριστικά (Stats) τους και τις ιδιότητες τους, απαιτείται μηχανισμός εισόδου νέων token από τους χρήστες ο οποίος

δημιουργήθηκε σε δεύτερο συμβόλαιο (λόγω περιορισμών μεγέθους των έξυπνων συμβολαίων).

Εντός του marketplace ο χρήστης μπορεί να αναζητήσει ανά token ID όλες τις διαθέσιμες αγγελίες πώλησης και μπορεί να ανταλλάξει τα δικά του διαθέσιμα νομίσματα για νομίσματα άλλων χρηστών.

6.3.8 Δημιουργία του Marketplace εντός του παιχνιδιού

Για την δημιουργία του Marketplace ακολούθησε η εξής εκτεταμένη διαδικασία:



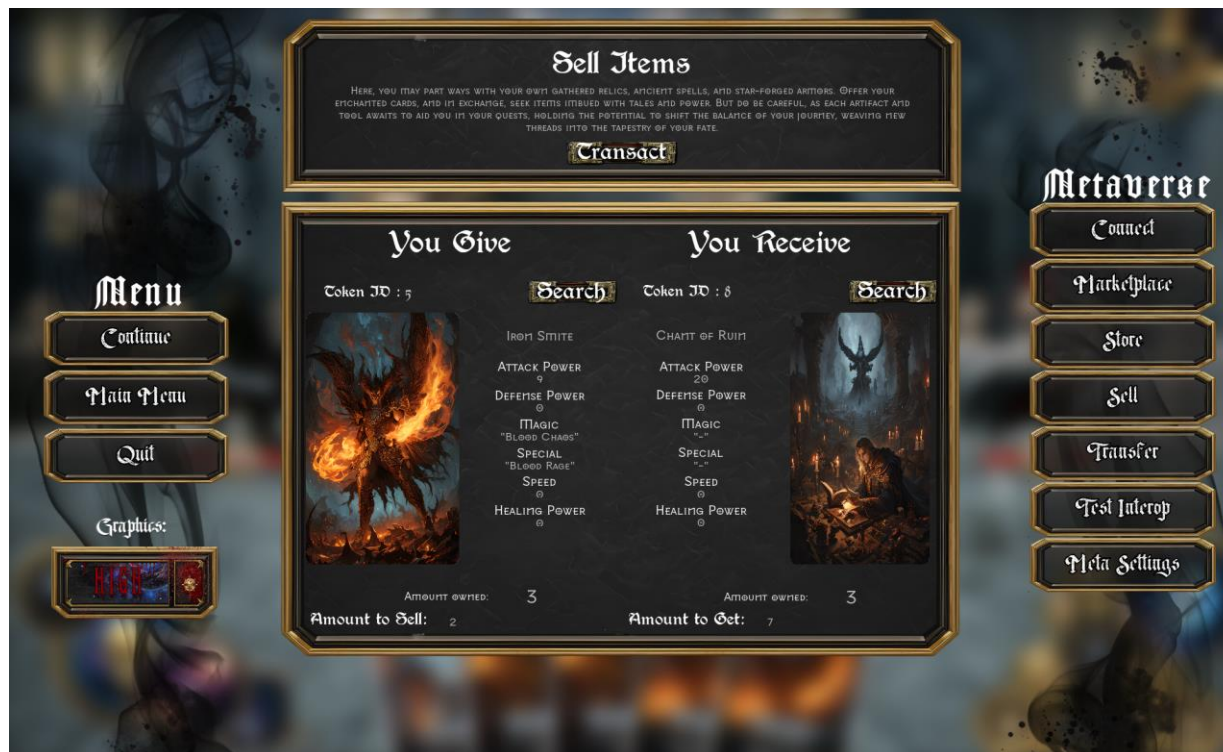
Χρησιμοποιήθηκαν τα Blueprints λόγω της δυναμικής τους φύσης. Η εικόνα επιδεικνύει μια απομάκρυνση του γραφήματος του Blueprint διεπαφής WBP_BuyPacks και διακρίνονται 2 λογικά «δέντρα» με κώδικα, το ένα κάτω αριστερά εκτελεί τις διαδικασίες αγοράς με το πάτημα του κουμπιού «BUY» και το άλλο, που εκτείνεται σε ολόκληρη την υπόλοιπη φωτογραφία:

1. Λαμβάνει την είσοδο του χρήστη.
2. Χρησιμοποιεί την συνάρτηση requestBankTokenInfo του έξυπνου συμβολαίου για να λάβει τα απαραίτητα δεδομένα για τις διαθέσιμες πωλήσεις.
3. Λαμβάνει και αποθηκεύει τα δεδομένα για να τα επεξεργαστεί
4. Αρχικοποιεί τιμές και προετοιμάζει την διαδικασία επίδειξης των τιμών μέσα από στοιχεία διεπαφής που έχουν ζευγοποιηθεί με visibility triggers (ορατότητας) τύπου ESlate Visibility Enumeration

5. Θέτει την λογική υπολογισμού και διαχείρισης των σελίδων εντός του marketplace, με ιδιαίτερη έμφαση στην επίδειξη μόνο των θέσεων με ενεργές τιμές για την πρώτη και την τελευταία σελίδα του marketplace.
6. Στο σχόλιο “Pages” διακλαδώνεται με τον μηχανισμό εναλλαγής σελίδων και τα triggers αυτού και προσαρμόζει και την ορατότητα σε αυτά ανάλογα με τον αριθμό των σελίδων.
7. Θέτει την ορατότητα των αγγελιών ανά σειρά.
8. Αφού έχει περάσει το αρχικό πέρασμα των τιμών ανά κατηγορία δεδομένων σε πίνακα, εκτελεί ένα δεύτερο πέρασμα μέσα από το οποίο τραβάει τις μεμονωμένες τιμές σε έναν δεύτερο πίνακα και τις περνάει στα ανάλογα στοιχεία της διεπαφής – ανά κατηγορία.
9. Έπειτα κατεβάζει το JSON από το IPFS, διαβάζει το αρχείο JSON και βρίσκει την τιμή χρησιμοποιώντας την επέκταση JSONparser.
10. Θέτει την τιμή αυτή σε μια μεταβλητή string την οποία διαβάζουν τα στοιχεία της διεπαφής για να βρουν το όνομα του token.

6.3.9 Πώληση καρτών

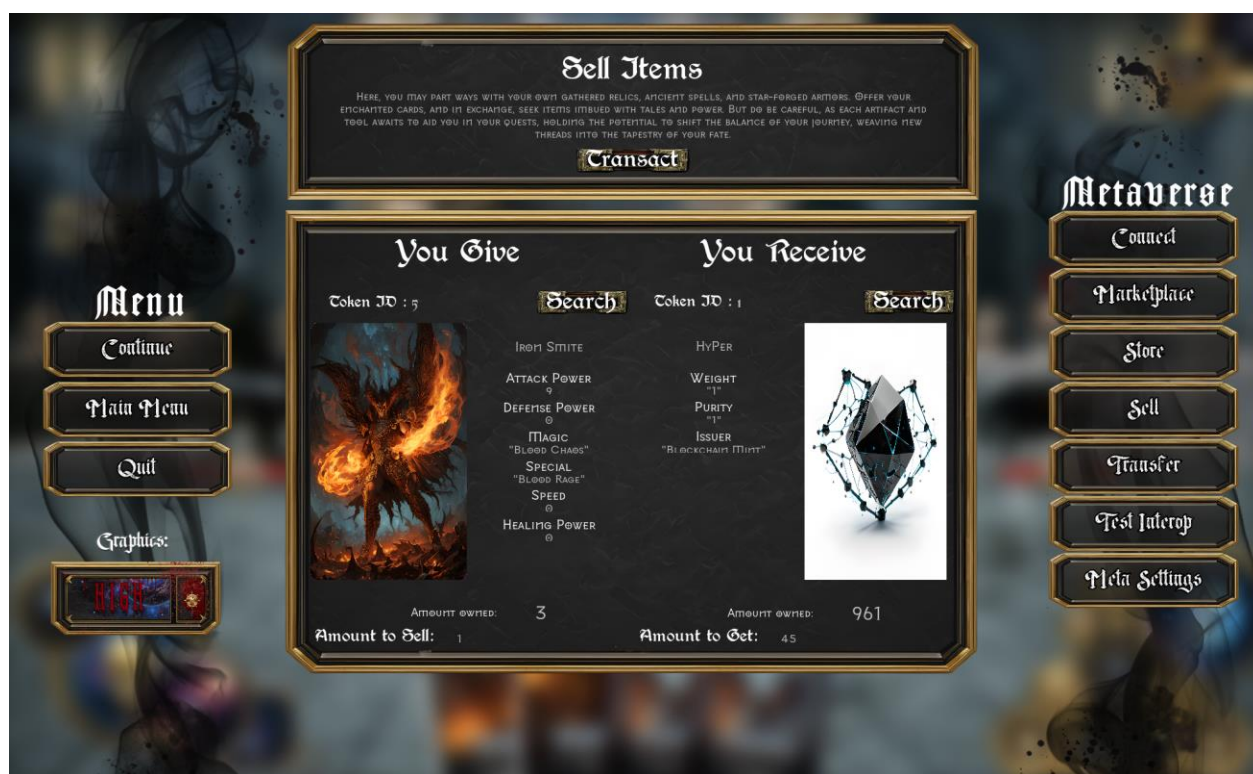
Ακολουθεί μια γρήγορη επίδειξη του μηχανισμού αγοράς καρτών από το ίδιο το συμβόλαιο.



Το menu πώλησης, αφού ο χρήστης αναζητήσει κάποιο NFT μέσα από τα ανάλογο ID.

Στο αριστερό μέρος του μενού ο χρήστης καλείται να πληκτρολογήσει την κάρτα που θέλει να πουλήσει / ανταλλάξει. Μπορεί να αναζητήσει οποιαδήποτε άλλη κάρτα εκείνος επιθυμεί – για την οποία κάρτα υπάρχει η ανάλογη καταχώρηση στα mappings από την συνάρτηση `setBatchMaxSupply`.

Μέσα από την πολυμορφικότητα του αρθρωτού (modular) σχεδιασμού της διεπαφής, ο χρήστης μπορεί να αιτήσει να ανταλλάξει τα δικά του περιουσιακά στοιχεία, τόσο για οποιουδήποτε νόμισμα οποιουδήποτε συνδεδεμένου κόσμου παιχνιδιού, όσο και για το κεντρικό νόμισμα των κόσμων των παιχνιδιών που συνδέονται στο Blockchain.



Στην φωτογραφία παρουσιάζεται η διαδικασία πώλησης της κάρτας για το βασικό νόμισμα. Τα χαρακτηριστικά που εμφανίζονται προστίθενται σε πραγματικό χρόνο από τα δεδομένα του IPFS.

“Όλες οι πληροφορίες της εικόνας έχουν κατέβει από το IPFS και μερικά από αυτά τα στοιχεία δεν υπήρχαν μέσα στο παιχνίδι πριν την αναζήτησή τους από το σύστημα. Αυτό επιτρέπει την κλήση για αγορά στοιχείων που δεν ανήκουν στο παιχνίδι, μέσα από το παιχνίδι.

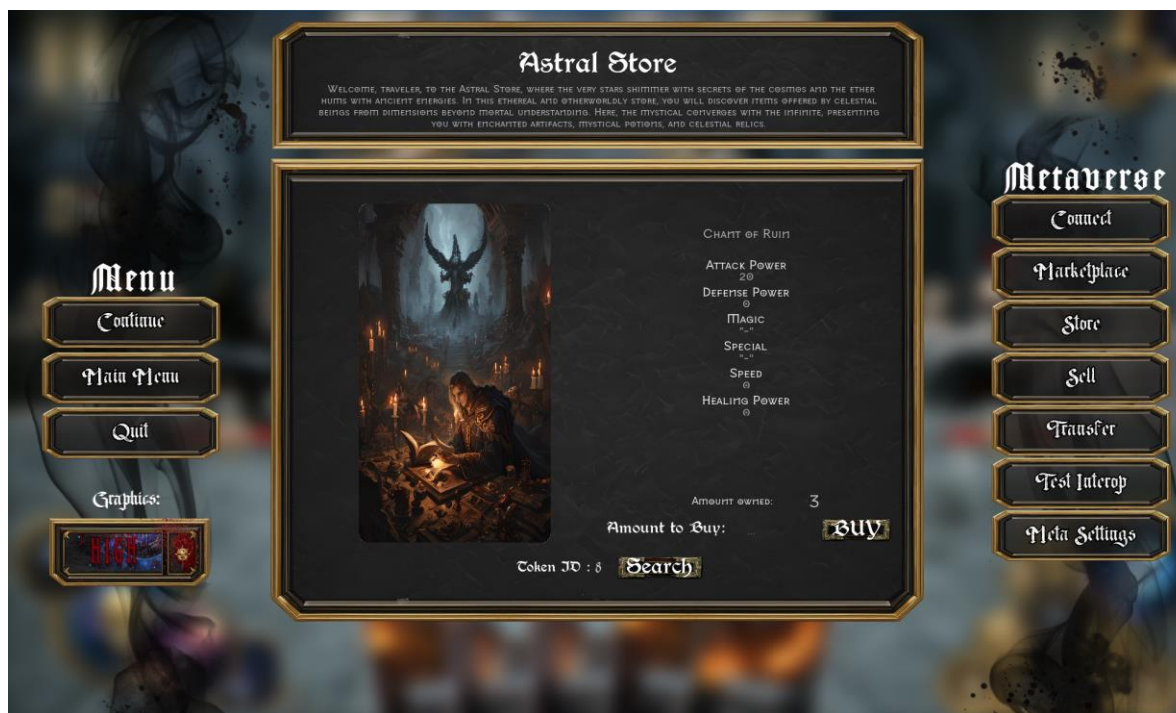
Για την ολοκλήρωση της διαδικασίας, το πρόγραμμα διαβάζει τα 4 παιδιά συμπλήρωσης κειμένου από τον χρήστη, της διεπαφής και καλεί την συνάρτηση `listAmountOfTokensForSale`.

6.3.10 Το κατάστημα εντός του παιχνιδιού

Ακολουθεί μια γρήγορη επίδειξη του μηχανισμού αγοράς καρτών από το ίδιο το σύμβολο.



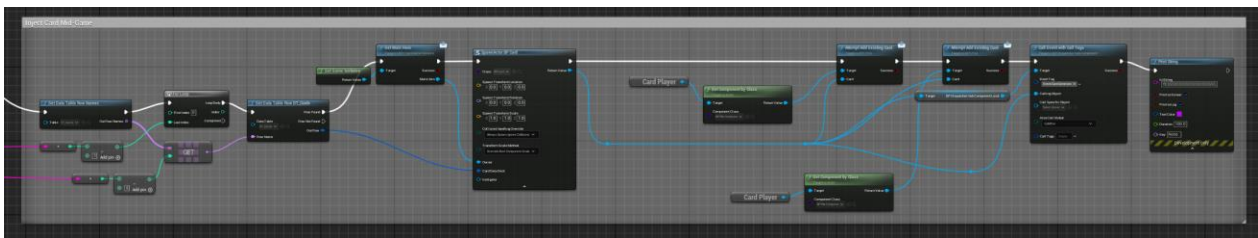
Το κατάστημα ακριβώς μετά την αρχικοποίηση του Widget,



Το κατάστημα αφού ο χρήστης αναζητήσει για κάποιο NFT μέσα από το ανάλογο ID.

Όλα τα δεδομένα που παρουσιάζονται στην εικόνα κατεβαίνουν από το IPFS. Πρώτα γίνεται λήψη του .json αρχείου με τα μεταδεδομένα και μετά επεξεργάζεται το αρχείο για να ληφθούν οι πληροφορίες της κάρτας. Έπειτα κατεβαίνει η εικόνα της κάρτας εξίσου από το IPFS και παρουσιάζεται στον χρήστη.

Ο χρήστης μπορεί να διαλέξει έναν αριθμό καρτών για να αγοράσει και όταν ολοκληρωθεί η συναλλαγή, οι κάρτες προστίθενται στο χέρι του κατά τον πραγματικό χρόνο παιχνιδιού και μπορεί επί τόπου να τις χρησιμοποιήσει. Αυτό είναι εφικτό εφόσον όλα τα Menu για την αγοραπωλησία έχουν εντειχιστεί εντός του Pause Menu (Μενού Παύσης παιχνιδιού).

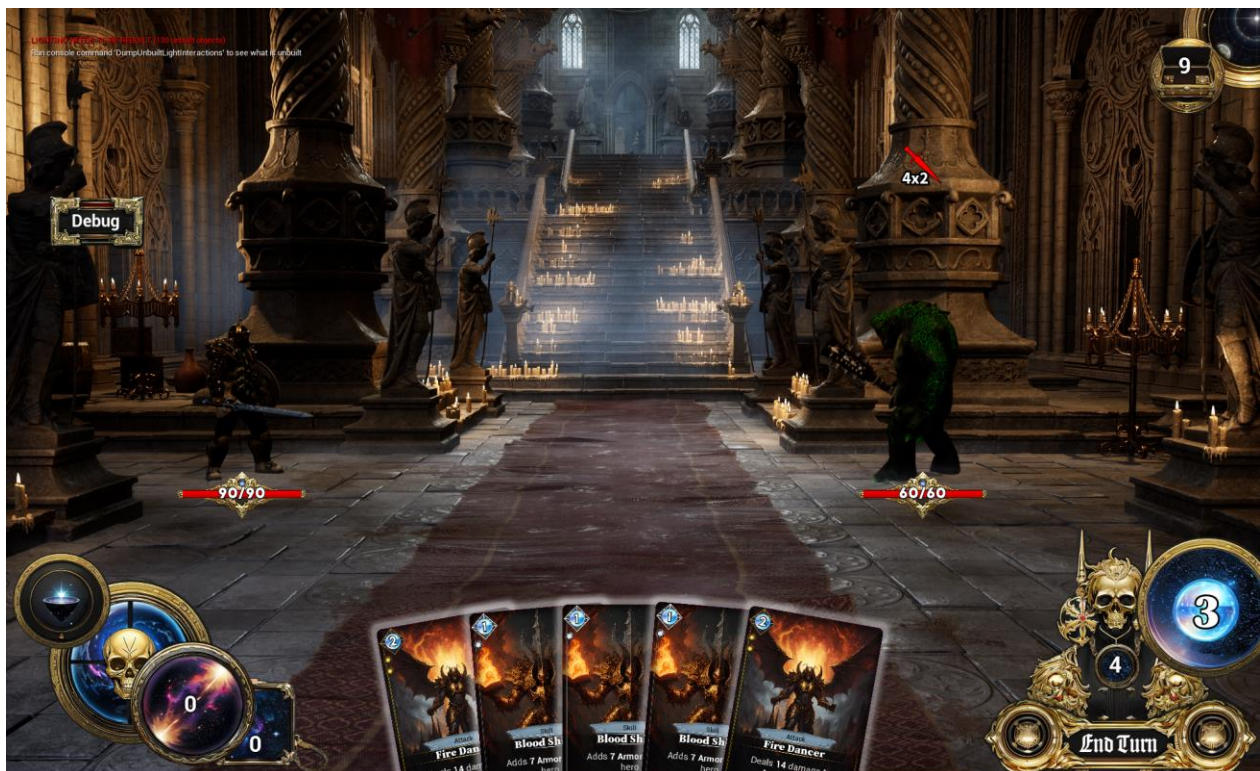
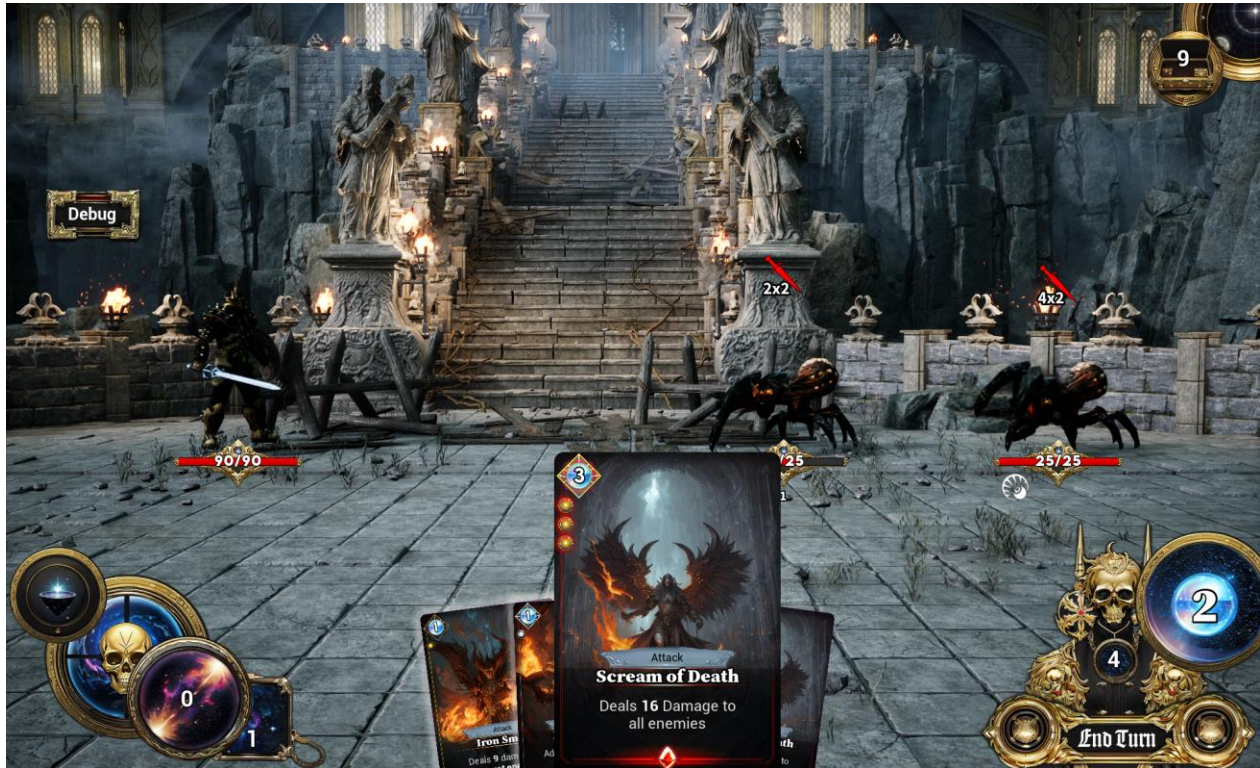


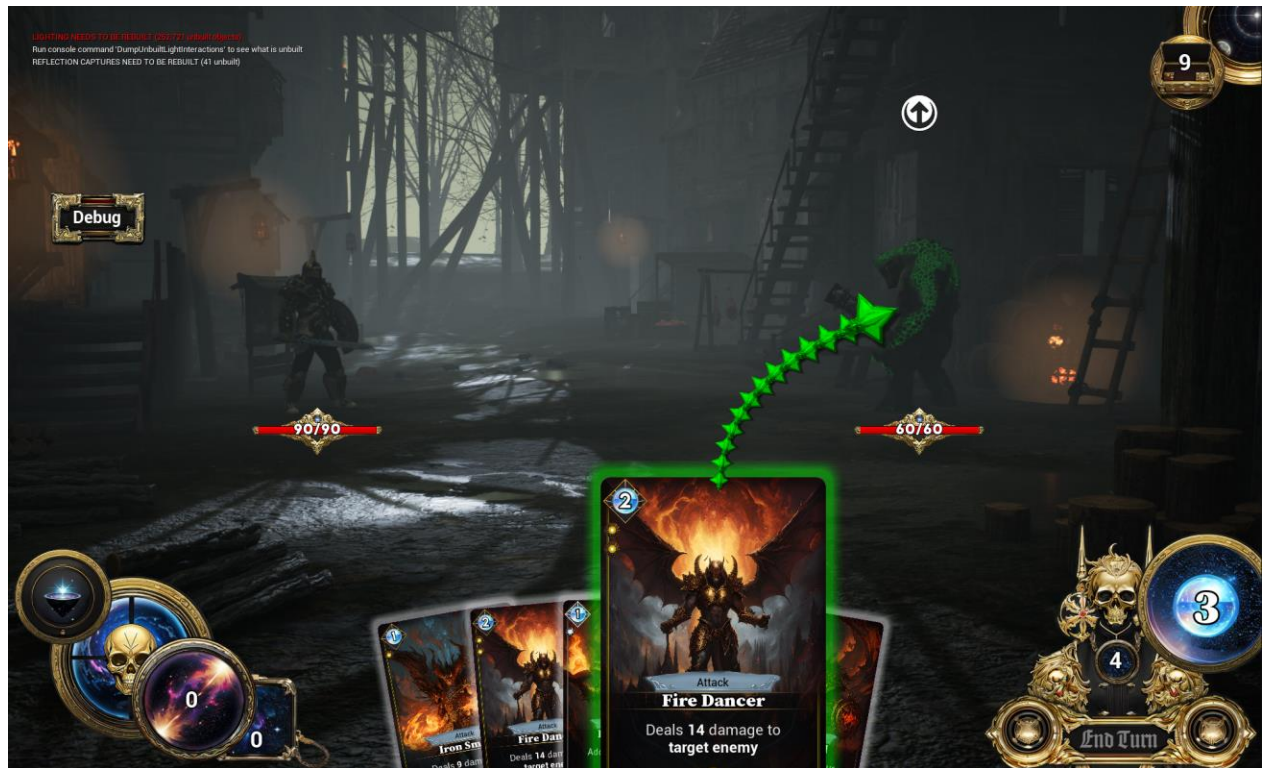
Blueprint λειτουργίας του μηχανισμού ο οποίος προσθέτει τις κάρτες στο χέρι του παίκτη. Αν δεν υπήρχαν ασύγχρονα στοιχεία και στοιχεία πραγματικού χρόνου στην μηχανή και το παιχνίδι, η λειτουργία αυτή δεν θα ήταν δυνατή να υλοποιηθεί.

Για την υλοποίηση του καταστήματος καλείται η συνάρτηση `buyFromStore` και το κατάστημα βρίσκεται εντός του γραφήματος του Widget Blueprint `WBP_Store2`.

6.3.11 Κεντρικός πυρήνας του παιχνιδιού









Ακολουθως παρουσιάζονται φωτογραφίες μέσα από το παιχνίδι κατά τον χρόνο παιχνιδιού.





6.4 Χαρακτηριστικά Καρτών και δομή JSON

Χρησιμοποιήθηκε η υπηρεσία pinata.cloud για την διαφύλαξη των αρχείων εντός του IPFS. Το CID οδηγεί σε φάκελο με .json αρχεία που περιέχουν μεταδεδομένα και έναν σύνδεσμο ο οποίος οδηγεί στην αντίστοιχη κεντρική φωτογραφία της κάθε κάρτας του παιχνιδιού. Από τις φωτογραφίες που ακολουθούν, γίνεται εμφανές πως τα χαρακτηριστικά δηλώνονται ως αντικείμενα (objects) μέσα στον πίνακα (array) "attributes".

Index of /ipfs/QmSGWFLhPhKu1p1qHC9dqWVBmoW6HtcPQLLZ86QPvRkBie		106 MB
QmSGWFLhPhKu1p1qHC9dqWVBmoW6HtcPQLLZ86QPvRkBie		
 10.png	QmaQ...SZLu	2.5 MB
 11.png	QmRe...MRHM	2.5 MB
 12.png	QmYw...oxyl	2.4 MB
 13.png	QmYX...HNBr	2.6 MB
 14.png	QmRY...tzET	2.3 MB
 15.png	QmZs...NWAw	2.8 MB
 16.png	QmfW...oxjV	2.8 MB
 17.png	QmP...i6zd	2.5 MB

Φάκελος με αρχεία μεταδεδομένων .png στο IPFS

```
1 {
2   "name": "Iron Smite",
3   "description": "Deals *var1* damage to *var2*",
4   "image": "ipfs://QmSGWFLhPhKu1p1qHC9dqWVBmow6HtcPQLLZ86QPvRkBie/5.png",
5   "tokenID": "5",
6   "GameID": "1",
7   "Variable1": "*0",
8   "Variable2": "<b>target enemy</>",
9   "Variable3": "",
10  "attributes": [
11    {
12      "trait_type": "Attack Power",
13      "value": 9
14    },
15    {
16      "trait_type": "Defense Power",
17      "value": 0
18    },
19    {
20      "trait_type": "Magic",
21      "value": "Blood Chaos"
22    },
23    {
24      "trait_type": "Special",
25      "value": "Blood Rage"
26    },
27    {
28      "trait_type": "Speed",
29      "value": 0
30    },
31    {
32      "trait_type": "Healing Power",
33      "value": 0
34    }
35  ]
36 }
```

Η δομή των μεταδεδομένων μίας κάρτας του παιχνιδιού.



Η κάρτα ROW 0 από το Data Table DT_Cards.

6.5 Κατασκευή έξυπνου συμβολαίου

Για το σύστημα των μοναδικών τεκμηρίων χρησιμοποιήθηκε το σύστημα έξυπνων συμβολαίων (smart contracts) στο EVM του Ethereum. Τα μοναδικά τεκμήρια προκύπτουν μέσα από τα Tokens που σώζονται σε δομές που θυμίζουν ζευγοποιημένους πίνακες (Arrays) – τα mapping στην γλώσσα Solidity. Η δομή του συμβολαίου ακολουθεί μια προσαρμογή του προτύπου ERC-1155 του Openzeppelin [18]. Συγκεκριμένα το κάλεσμα της συνάρτησης «_mint» είναι υπεύθυνο για την διαδικασία του minting και αυτή η διαδικασία παίρνει μέρος μέσα από την καταχώρηση των δεδομένων στο mapping_balances, όπως ακολούθως:

```
mapping(uint256 => mapping(address => uint256)) private _balances;
```

- Στο πρώτο (πράσινο) mapping η τιμή interger 256bit αναπαριστά το μοναδικό ID αναγνωριστικό του κάθε Token – μοναδικής οντότητας. Για κάθε Token ID υπάρχει ένα αντίστοιχο mapping με δεδομένα για τους χρήστες.
- Κάθε εσωτερικό mapping (μπλε), συνδέει τις μοναδικές διευθύνσεις των χρηστών με έναν αριθμό.
- Το δεξιά interger συγκρατεί το πλήθος από το κάθε νόμισμα που έχει στην ιδιοκτησία του ο κάθε χρήστης.

Σε ένα απλουστευμένο παράδειγμα, για κάθε χρήστη που έχει το token ID 1, κρατείται μια τιμή `int` με τον αριθμό του πλήθους των αντιγράφων του Token ID 1 που έχει στην κατοχή του ο κάθε χρήστης.

Σε ότι αφορά την 18-1155, έχει υβρίδια φύση νομισμάτων, όπου τα νομίσματα μπορούν να λειτουργούν σαν μοναδικά τεκμήρια αλλά και ως απλά συναλλάξιμα νομίσματα. Επιλέχθηκε, επειδή επιτρέπει στο πείραμα να εξερευνήσει το πλήρες εύρος της τεχνολογίας NFT και να πιέσει τα smart contracts στα όρια τους.

Συγκεκριμένα, κατασκευάστηκαν 8 πειραματικά έξυπνα συμβόλαια χρησιμοποιώντας τα πρότυπα ERC-20, ERC-721, ERC-1155 [18], oracles, λειτουργίες διαχείρισης χρηστών και με λειτουργίες διάδρασης του συμβολαίου με τους χρήστες. Καθοριστικός παράγοντας για την κατασκευή του τελικού συμβολαίου, ήταν το μέγιστο μέγεθος που μπορεί να έχει το κάθε συμβόλαιο που τρέχει στο EVM. Το αρχικό κεντρικό συμβόλαιο, που είχε όλες τις λειτουργίες που θα είχε το Back End ενός παιχνιδιού που προσφέρει υπηρεσίες αγοραπωλησίας αντικειμένων, ξεπερνούσε το όριο κατά 3 φορές. Στη συνέχεια το συμβόλαιο βελτιστοποιήθηκε 3 φορές, μέχρι να συμπεριλάβει τον μέγιστο αριθμό λειτουργιών από το αρχικό μεγάλο συμβόλαιο – εντός του μέγιστου επιτρεπόμενου μεγέθους συμβολαίου στο Ethereum.

Για την βελτιστοποίηση, αυτή, του μεγέθους - έπρεπε να αφαιρεθούν σχεδόν όλα τα strings κείμενα που επέστρεφαν πίσω όταν αποτύγχανε κάποιο `require`. Δημιουργώντας ασάφειες κατά την διαδικασία αποτυχίας εκτέλεσης της συνάρτησης, μέσα από τα `require`. Ταυτόχρονα, έπρεπε να αναμορφοποιηθούν οι λειτουργίες διαφόρων συναρτήσεων και η λογική να συμπυκνωθεί όσο το δυνατόν περισσότερο. Δυνατότητες κοστολόγησης συναλλαγών σε δολάρια από oracle και λήψης τυχαίων τιμών έπρεπε να αφαιρεθούν. Το σύστημα που επιτρέπει στους χρήστες να δημιουργήσουν εκείνοι δικά τους Tokens και να προσαρτηθούν μοναδικά διακριτά στα ονόματα των tokens, αφαιρέθηκε. Επίσης, αφαιρέθηκαν οι λειτουργίες διαχείρισης και banning χρηστών, μαζί με τα flags για την λειτουργία του banning συστήματος, όταν εκτελούσαν ανορθόδοξες αιτήσεις για συναλλαγές. Τα imports περιορίστηκαν στο ελάχιστο. Τέλος, βελτιστοποιήθηκε η χρήση των mapping και ο αριθμός των απαιτούμενων mapping για την «εξομοίωση» χρήσης struct εντός του συμβολαίου και την υποστήριξη εκτέλεσης περίπλοκων διαδικασιών.

6.5.1 Ανάλυση Συμβολαίου

Αρχικά, εισάγονται τα imports και προσδιορίζεται η έκδοση του compiler. Χρησιμοποιείται το πρότυπο ERC1155.

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.20;

import "@openzeppelin/contracts/token/ERC1155/ERC1155.sol";
import "@openzeppelin/contracts/access/Ownable.sol";
import "@openzeppelin/contracts/token/ERC1155/extensions/ERC1155Burnable.sol";
```

Γίνεται αρχικοποίηση του συμβολαίου. Το ipfs://Qm... είναι η θέση που μπήκε το CID του φακέλου εντός του IPFS. Το hosting των αρχείων παρέχεται από το <https://www.pinata.cloud/>. Το address **initialOwner** εμπεριέχει την διεύθυνση του πορτοφολιού του ιδιοκτήτη του συμβολαίου.

```
contract RuinsOfEden is ERC1155, Ownable, ERC1155Burnable {
    --Τα mappings και η αρχικοποίηση μεταβλητών αφαιρέθηκαν από [εδώ], για την
    --καλύτερη παρουσίαση του κώδικα μεταφέρθηκαν πάνω από τις συναρτήσεις στις
    --οποίες βρίσκουν χρήση!

    string public ContractInfo = "Creator: S. Spagadoros / v0.1 / ALPHA_01";

    constructor(address initialOwner)
        ERC1155("ipfs://Qm...../")
        Ownable(initialOwner)
    {}
}
```

Μέσα στην αρχιτεκτονική του οικονομικού συστήματος που προκύπτει από τις λειτουργίες του έξυπνου συμβολαίου, υπάρχει ένα κεντρικό νόμισμα που χρησιμοποιείται για την μορφοποίηση ενός ομοιογενούς πλαισίου συναλλαγής. Το τυπικό «Standard» νόμισμα, αυτό, χρησιμοποιείται για την τυποποίηση και απλοποίηση των οικονομικών συναλλαγών σε όλο το σύστημα.

Ταυτόχρονα βοηθάει το ίδιο στην δημιουργία χρηματοοικονομικών δεικτών για την παρακολούθηση του συστήματος. Το συμβόλαιο παρέχει λειτουργίες εναλλαγής του βασικού νομίσματος, αυτού, για να μπορεί το σύστημα να αναπροσαρμοστεί με τις τάσεις της εσωτερικής αγοράς.

```

uint256 public mainCurrency = 1;
//Το βασικό Fungible νόμισμα που χρησιμοποιεί το παιχνίδι.

function setMainCurrency(uint256 _mainCurrency) public onlyOwner {
    mainCurrency = _mainCurrency;
}

```

Η συνάρτηση **setMainCurrency** επιτρέπει την εναλλαγή του βασικού νομίσματος.

Στην μορφοποίηση του συστήματος συναλλαγών, χρησιμοποιήθηκαν λειτουργίες οι οποίες προσφέρουν τον μέγιστο βαθμό πολυπλοκότητας και δυνατοτήτων. Όλα τα νομίσματα αποτελούν συνάλλαγμα και το σύστημα φτιάχτηκε για να μιλάει με πολλαπλά παιχνίδια, για την επίδειξη της διαλειτουργικότητας (Interoperability).

Επειδή, λοιπόν, το σύστημα επιτρέπει την χρήση του smart contract σε πολλαπλά παιχνίδια. Η συνάρτηση `setupGame` είναι το πρώτο βήμα αρχικοποίησης και χρήσης ενός παιχνιδιού εντός του έξυπνου συμβολαίου.

```

mapping(uint256 => string) private gameName;
// το uint256 αναπαριστά το gameId || το string εμπεριέχει το όνομα
mapping(uint256 => bool) private gameExists;
// Δηλώνεται η ύπαρξη του παιχνιδιού, για τον γρήγορο έλεγχο της

function setupGame(uint256 ID, string memory name) public onlyOwner {
    gameName[ID] = name;
    gameExists[ID] = true;
}

```

Ένα ασαφές, λοιπόν, σύστημα, σαν εκείνο που δημιουργήθηκε, έχει στον πυρήνα του τις οντότητες που συναλλάσσονται και την φύση τους και όχι το σύστημα που εκτελεί τις συναλλαγές ή τους συναλλασσόμενους. Έχοντας, παράλληλα, κατά νου τον χαρακτήρα της αρχιτεκτονικής των παιχνιδιών ECS – Entity Component System (η οποία είναι επιθυμητή σε συστήματα με μεταλλασσόμενα όντα), η βασικότερη λειτουργία που καθορίζει το έξυπνο συμβόλαιο στα περισσότερα σημεία του, προγραμματιστικά και λειτουργικά, είναι η ικανότητα ή μη, ύπαρξης των μεμονωμένων οντοτήτων (δηλαδή των : καρτών, νομισμάτων και κλώνων αυτών). Ουσιαστικά, στον πυρήνα ενός «αντικειμενοστρεφούς» συστήματος, «υφίστανται υπαρξιακά παίγνια» τα οποία καθορίζουν και περιορίζουν την παραγωγή νέων οντοτήτων. Με απλά λόγια, το κύριο ερώτημα που απαντάται σε οποιαδήποτε διεργασία, που δημιουργεί νέα όντα, είναι το κατά πόσο τα όντα αυτά θα έπρεπε να υπάρξουν.

Σε μια περαιτέρω απλούστευση της παρατήρησης αυτής : Το συμβόλαιο στον πυρήνα του ελέγχει την ύπαρξη των οντοτήτων / νομισμάτων. Όλες οι εργασίες αφορούν το κατά πόσο θα έπρεπε να υπάρχουν τα όντα, αν θα έπρεπε να δημιουργηθούν ή να καταστραφούν και αν αυτά τα όντα μπορούν καν να έρθουν σε ύπαρξη – εξ αρχής.

Αν εγκριθεί η ύπαρξη ενός ακόμα «κλώνου» ενός νομίσματος/οντότητας, τότε και μόνο έρχεται στην ύπαρξη. Αυτή η διαδικασία είναι κεντρική, διότι το συμβόλαιο χρησιμοποιείται από εικονικούς κόσμους που εσωτερικά φέρουν πολύπλοκες οικονομίες. Η άπειρη έκδοση όλων των νομισμάτων θα κατέστρεφε την τελική εμπειρία του χρήστη και δεν θα επέτρεπε την ύπαρξη σπάνιων αγαθών, η απόκτηση των οποίων κινητοποιεί του παίκτες με διάφορους τρόπους.

Παράλληλα, ο περιορισμός της έκδοσης θυμίζει τα παραδοσιακά χρηματοοικονομικά συστήματα, τα οποία περιορίζουν την έκδοση και διάθεση ενός νομίσματος για την αποφυγή του πληθωρισμού, την σταθεροποίηση της οικονομίας, τη διατήρηση της αξίας και την οικονομική ανάπτυξη μέσα σε ένα σταθερό και έμπιστο περιβάλλον [91].

Για αυτούς του λόγους που προαναφέρθηκαν, το σύστημα χρησιμοποιεί την συνάρτηση του `setBatchMaxSupply` για τον περιορισμό της προμήθειας συγκεκριμένων νομισμάτων επιτρέποντας την δημιουργία ψηφιακών αγαθών.

```

    mapping(uint256 => uint256) public baseCostOfTokens;
// Κόστος των νομισμάτων σε mainCurrency token ID.
    mapping(uint256 => uint256) public maxStoreSupply;
// Μέγιστος αριθμός νομισμάτων που μπορούν να πωληθούν στο κατάστημα
    mapping(uint256 => uint256) public maxSupply;
// Μέγιστος αριθμός νομισμάτων που μπορούν να εκδοθούν ανά ID
    mapping(uint256 => bool) public fungible;

function setBatchMaxSupply(
    uint256[] calldata ids,
    uint256[] calldata _maxSupplies,
    bool[] calldata _Fungible,
    uint256[] calldata cost,
    uint256[] calldata _maxStoreSupply
        ) external onlyOwner {
    require(ids.length == _maxSupplies.length &&
        _maxSupplies.length == _Fungible.length);
// Απαιτείται το ίδιο πλήθος τιμών να έχει δοθεί ανά στοιχείο

    for (uint256 i = 0; i < ids.length; i++) {
        uint256 id = ids[i];
        if (_Fungible[i] == false) {
            require(_maxSupplies[i] >= inCirculation[id] &&
                _maxStoreSupply[i] >= SoldStoreSupply[id]);
// Ελέγχει πως η προσθήκη νέων τιμών σε ήδη διαθέσιμα νομίσματα δεν προκαλούν
// προβλήματα.

            maxSupply[id] = _maxSupplies[i] + _maxStoreSupply[i];
            fungible[id] = false;
            maxStoreSupply[id] = _maxStoreSupply[i];
        }
        else {
            fungible[id] = true;
            maxStoreSupply[id] = _maxStoreSupply[i];
// Αν ένα fungible νόμισμα λάβει την τιμή 1 στο maxStoreSupply, μπορεί να πωληθεί
// χωρίς περιορισμούς διαθεσιμότητας στους χρήστες μέσα από το κατάστημα
        }
        baseCostOfTokens[id] = cost[i];
    }
}

```

Συνάρτηση setBatchMaxSupply και σημαντικά mapping που χρησιμοποιούνται παντού μέσα στο συμβόλαιο για τον έλεγχο της διαθεσιμότητας.

Τα νομίσματα που μπορούν, τώρα, να «παραχθούν»:

- μπορούν να πωληθούν από το ίδιο το συμβόλαιο προς τους χρήστες.
- μπορούν να αποκτηθούν μέσα από την ολοκλήρωση ορισμένων επιτευγμάτων εντός των παιχνιδιών ή και εκτός.
- αποτελούν αγαθά με μοναδικά τεκμήρια και (θεωρητικά) «άπειρη» διάθεση (εντός της κλίμακας του Blockchain, αν αυτό είναι επιτρεπτό)
- μπορούν να αποτελούν μοναδικά ή συλλεκτικά αγαθά
- μπορούν να αποτελούν κοινά εναλλάξιμα νομίσματα

Πρακτικά, για την αρχικοποίηση του συμβολαίου και χρήση οποιουδήποτε νομίσματος είναι απαραίτητη η χρήση της `setBatchMaxSupply`, καθώς οι μεταβλητές `fungible`, `maxSupply` και `maxStoreSupply` ελέγχονται από την διαδικασία του `minting`, για να μην ξεπεράσει το νόμισμα το ορισμένο όριο διάθεσης.

Το `minting` παίρνει μέρος μέσα από την συνάρτηση `mint`, η οποία καλεί στην στάνταρ συνάρτηση `_mint` που εισάγεται από τα `import`. Για τις επιπρόσθετες λειτουργίες πώλησης καρτών και ανταμοιβής των παικτών, χρησιμοποιούνται η συναρτήσεις `mintable` και `logMint`.

Πριν την διαδικασία του `minting` γίνεται έλεγχος του υπόλοιπου διάθεσης των νομισμάτων μέσα από την ορισμένη τιμή από την συνάρτηση `setBatchMaxSupply` και την χρήση της εντολής `require(mintable(id, amount, store), "...");`.

Έπειτα, ανανεώνεται η τιμή του `mapping inCirculation` για το σωστό Token ID, για την δυνατότητα υπολογισμού των διαθέσιμων νομισμάτων προς `minting` από την συνάρτηση `giveLoot` και από το κατάσταση εντός του παιχνιδιού. Παράλληλα το `inCirculation` παρέχει γρήγορα στοιχεία, κατά την κλήση του, για την διαθεσιμότητα όλων των καρτών που έχουν εκδοθεί, ανά Token ID.

Τα διαθέσιμα νομίσματα από την συνάρτηση `giveLoot` υπολογίζονται από την εξής αφαίρεση `(maxSupply[id] - inCirculation [id]) -(maxStoreSupply[id] - SoldStoreSupply [id])`

ή

```
(maxSupply[id] - maxStoreSupply[id]) - (inCirculation [id] - SoldStoreSupply [id])
```

Η λογική της συνάρτησης logMint και το γιατί σχεδιάστηκε έτσι αναλύεται αργότερα.

```
mapping(uint256 => uint256) public inCirculation;
mapping(uint256 => uint256) public SoldStoreSupply;
// Χρησιμοποιείται για την παρακολούθηση της διάθεσης των νομισμάτων από το
// κατάσταση εντός του παιχνιδιού

function mintable(
    uint256 id,
    uint256 amount,
    //^^το πλήθος των Token στα οποία θα γίνει minting ταυτόχρονα
    bool store
// Η τιμή αυτή είναι true μόνο όταν οι συναλλαγές έρχονται από το εσωτερικό
// κατάσταση του παιχνιδιού
) internal view returns (bool canMint) {
    if (fungible[id]) {
        canMint = true;
// Τα νομίσματα που έχουν τιμή fungible ως true, είναι πάντα mintable και έχουν
// απεριόριστη διάθεση
    }
    else if (store && SoldStoreSupply[id] + amount <= maxStoreSupply[id]) {
// Τα νομίσματα δεν μπορούν να ξεπερνάνε την ορισμένη διάθεση εντός του
// maxStoreSupply ανά ID
        canMint = true;
    }
    else if (maxSupply[id] - maxStoreSupply[id] >= inCirculation[id] +
        amount && store == false) {
        canMint = true;
    }
    else {
        canMint = false;
    }
    return canMint;
// Η μεταβλητή canMint αποφασίζει αν μπορεί να ολοκληρωθεί η κάθε συναλλαγή
// δημιουργίας νομισμάτων μέσα από minting
}
```

Η συνάρτηση mintable ελέγχει την δυνατότητα έκδοσης νέων νομισμάτων από το συμβόλαιο, με βάση τις τιμές που έχουν οριστεί από την συνάρτηση setBatchMaxSupply.

```

function logMint(
  address userAdd,
  uint256 id,
  uint256 amount
) internal {
  ///////////////////////////////////////////////////////////////////
  for (uint256 i = 0; i < userTokenId[userAdd].length; i++) {
    if (userTokenId[userAdd][i] == id) {
      userTokenAmount[userAdd][i] += amount;
      return;
    }
  }
  userTokenId[userAdd].push(id);
  userTokenAmount[userAdd].push(amount);
  // Η λογική του for loop και της χρήσης του userTokenId γίνεται κατανοητή στην
  // συνάρτηση getUserTokens
  ///////////////////////////////////////////////////////////////////
}

function mint(
  uint256 id,
  uint256 amount,
  bool store
) internal {
  require(mintable(id, amount, store), "Token not mintable");
  _mint(msg.sender, id, amount, "");
  logMint(msg.sender, id, amount);
  inCirculation[id] += amount;
}

```

Η συνάρτηση logMint καταγράφει την παρουσία του νομίσματος για τον κάθε χρήστη και καλείται μέσα από την συνάρτηση mint, όπως παρουσιάζεται στην φωτογραφία.

Μια σημαντική λειτουργία που προσφέρει η χρήση των έξυπνων συμβολαίων βρίσκεται στη δυνατότητα περιορισμού τόσο της έκδοσης των μεμονωμένων οντοτήτων, όπως προαναφέρθηκε, αλλά και στη δυνατότητα περιορισμού τέλεσης των ξεχωριστών διαδικασιών ενός εξωτερικού συστήματος, οι οποίες βασίζονται σε ένα καταναλώσιμο περιορισμένο αγαθό. Ο περιορισμός επιτυγχάνεται μέσα από την ζευγοποίηση των λειτουργιών των ξεχωριστών διαδικασιών με το `_burn` νομισμάτων. Για κάθε «εκτέλεση» της διαδικασίας, εκτελείται μια συναλλαγή χρέωσης της χρήσης, παρόμοια με το «Usage Fee» στα ATM των τραπεζών.

Μέσα από τον περιορισμό των κύκλων εκτέλεσης διεργασιών και λειτουργιών, μπορεί να εφαρμοστεί ένας μηχανισμός περιορισμού παιχνιδιού (gameplay limitation mechanism). Αυτοί οι μηχανισμοί βρίσκουν χρήση συνήθως σε δωρεάν παιχνίδια, μέσα από συστήματα ενέργειας και χρονόμετρα, για την κινητοποίηση του χρήστη ως προς την αγορά περισσότερου «παιχνιδιού». Αυτοί οι μηχανισμοί αποτελούν βασικό στοιχείο του μοντέλου παιχνιδιών ως υπηρεσίες (Games as a Service - GaaS). Μέσα στο πλαίσιο του Blockchain όμως, είναι δυνατή η χρήση ενός συστήματος περιορισμού παιχνιδιού για την αποφυγή υπερφόρτωσης του δικτύου Blockchain, ταυτόχρονα αποθαρρύνουν την κατάχρηση των υπηρεσιών από τους χρήστες.

Ως προς το έξυπνο συμβόλαιο που δημιουργήθηκε, το `mainCurrency` νόμισμα δίνεται δωρεάν σε κάθε μοναδική πρώτη είσοδο σε παιχνίδι του έξυπνου συμβολαίου και λειτουργεί ως μηχανισμός περιορισμού παιχνιδιού. Ταυτόχρονα προωθεί της διαλειτουργικότητα και την δημιουργία ενός ενιαία «πολύ-κοσμικού» συστήματος ανάμεσα σε πολλά βιντεοπαιχνίδια, τύπου meta-verse.

Αξίζει όμως, εδώ, να σημειωθεί πως ο μηχανισμός που επιτρέπει σε οποιοδήποτε νόμισμα να λειτουργεί σαν fungible μέσα από το `setBatchMaxSupply`, επιτρέπει την δημιουργία μεμονωμένων οικονομιών οι οποίες μπορούν να συναλλάξουν το δικό τους κεντρικό νόμισμα για το `mainCurrency` του δικτύου και ο τρόπος ανανέωσης της ενεργείας να παίρνει μέρος μέσα από ολοκλήρωση επιτευγμάτων ή αγορά περισσότερων νομισμάτων με χρήματα.

Ως προς το παιχνίδι ο μηχανισμός περιορισμού παιχνιδιού εφαρμόζεται μέσα από την αγορά από το «Store» κατάσταση εντός του παιχνιδιού με `mainCurrency`. Καθώς ο παίκτης αγοράζει νέες κάρτες, περιορίζει το διαθέσιμο υπόλοιπο έως ότου αυτό εξαντληθεί.

Ακολούθως παρουσιάζεται η συνάρτηση `initNewUser`:

```

    mapping(address => mapping(uint256 => bool)) public userHasGame;
// Μοναδικό τεκμήριο κατοχής παιχνιδιού. Ενεργοποιείται κατά το πάτημα του
// κουμπιού CONNECT και την ολοκλήρωση της εγγραφής της πράξης στην μνήμη του EVM
    mapping(uint256 => bool) private gameExists;
// Mapping το οποίο παρακολουθεί την κατάσταση ύπαρξης του παιχνιδιού προς το
// συγκεκριμένο έξυπνο συμβόλαιο
    mapping(uint256 => address) internal userLog; //contains all users
    uint256 public totalUsers = 0;
    mapping(uint256 => bool[]) private rewardIsGameGift; // game => [voucherID
(or) rewardID]

function initNewUser(uint256 gameID) external {
    address user = msg.sender;

    require(userHasGame[user][gameID] == false && gameExists[gameID] == true,
        "Error.");
// Τα κείμενα έχουν περιοριστεί στο ελάχιστο λόγο του μικρού μέγιστου μεγέθους
//των έξυπνων συμβολαίων

    mint(mainCurrency, 1000, false);
// Δίνει 1000 νομίσματα από το νόμισμα mainCurrency ανά παιχνίδι, προωθώντας την
// διαλειτουργικότητα και την αίσθηση ύπαρξης ενός ενιαίου συστήματος
    userHasGame[user][gameID] = true;
    userLog[totalUsers] = msg.sender;
// Χρησιμοποιείται σε άλλα συστήματα παρακάτω
    totalUsers++;
// Χρησιμοποιείται σε άλλα συστήματα παρακάτω
///////////////////////////////////////////////////////////////////
    uint256 helper = rewardIsGameGift[gameID].length;
    for (uint256 i = 0; i < helper; i++) {
        voucher[msg.sender][gameID].push(false);
    }
// Χρησιμοποιείται για την υποστήριξη του συστήματος με τα reward και τα voucher
// για την δημιουργία ενός συστήματος επιβράβευσης του παίκτη.
// Γεμίζει το mapping voucher με τιμές false, αρχικοποιώντας το για το παιχνίδι.
///////////////////////////////////////////////////////////////////
}

```

Η συνάρτηση `initNewUser` παράγει ένα μοναδικό τεκμήριο κατοχής του παιχνιδιού μέσα από το mapping `userHasGame`. Η εκτέλεση της συνάρτησης μπορεί να πάρει μέρος μόνο μια φορά ανά παιχνίδι, ανά λογαριασμό. Ταυτόχρονα επιβραβεύει τον χρήστη και του παρέχει χρήσιμα νομίσματα συναλλαγής για την απόκτηση ψηφιακών οντοτήτων μέσα στο παιχνίδι, από το συμβόλαιο.

Η συνάρτηση `buyFromStore` κάνει εφικτή την αγορά νομισμάτων μέσα από το fungible νόμισμα `mainCurrency`. Αρχικά, ελέγχεται η ικανότητα του χρήστη να πληρώσει το κόστος της συναλλαγής και αν το νόμισμα, αυτό, διατίθεται προς πώληση. Τα νομίσματα μπορούν να αγοραστούν σε παρτίδες του ίδιου ID και υπολογίζεται το συνολικό κόστος τους στην μεταβλητή `realCost`. Τέλος, ο χρήστης πληρώνει το αντίτιμο της συναλλαγής μέσα από την συνάρτηση `operationFee` και διαδικασίες εικονικού `_burn`, οι οποίες βοηθούν και στον περιορισμό του παιχνιδιού. Εμβαθύνοντας, το συμβόλαιο έχει δομηθεί έτσι ώστε το `operationFee` να μεταποιεί το κόστος σε απόθεμα του συμβολαίου, για την δημιουργία ενδιαφερόντων χρηματοοικονομικών συστημάτων. Έπειτα γίνεται το `mint` για την ολοκλήρωση της

```
//only trades in mainCurrency
function buyFromStore(uint256 id, uint256 amount) public {
    require(balanceOf(msg.sender, mainCurrency) >=
        baseCostOfTokens[id] * amount && baseCostOfTokens[id] > 0);
    if (!fungible[id]) {
        require(SoldStoreSupply[id] + amount <= maxStoreSupply[id],
            "Can't exceed mint limit");
    } else {
        require(maxStoreSupply[id] == 1);
    }
    // Αν το νόμισμα είναι fungible και η τιμή στο maxStoreSupply είναι 1, τότε έχει
    // απεριόριστη διαθεσιμότητα
    }
    uint256 realCost = baseCostOfTokens[id] * amount;
    operationFee(mainCurrency, realCost);
    mint(id, amount, true);
    SoldStoreSupply[id] += amount;
}
```

Η συνάρτηση `buyFromStore` την οποία καλεί το κατάστημα `Store` εντός του παιχνιδιού.

Το αντίτιμο της συναλλαγής από το κατάστημα, είναι το ίδιο το κόστος των νομισμάτων και καταγράφεται στην μνήμη μέσα από το bankWallet. Παράλληλα, καίγονται τα νομίσματα από το πορτοφόλι του χρήστη. Αν και αυτή η μέθοδος είναι υποδεέστερη των εναλλακτικών, χρησιμοποιήθηκε λόγω έλλειψης χώρου.

```
function findUserTokenSlot(uint256 id, address userAdd)
    public
    view
    returns (uint256 slot)
{
    // Επειδή οι ταυτότητες (IDs) προωθούνται με άτακτη σειρά, πρέπει να βρίσκεται η
    // θέση τους χρησιμοποιώντας αυτήν τη συνάρτηση
    for (uint256 i = 0; i < userTokenId[userAdd].length; i++) {
        if (userTokenId[userAdd][i] == id) {
            return i;
        }
    }
}

function operationFee(uint256 id, uint256 amount) internal {
    bankWallet[id] += amount;
    // Εκτελεί την μεταφορά του κόστους των καρτών προς το συμβόλαιο
    burnUserTokens(id, amount);
    // Αφαιρεί το κόστος από τον χρήστη
}

function burnUserTokens(uint256 id, uint256 amount) public {
    _burn(msg.sender, id, amount);
    uint256 findID = findUserTokenSlot(id, msg.sender);
    // Βρίσκει την θέση του νομίσματος στο «πίνακα ταχείας καταγραφής»
    // των αγαθών του χρήστη που είναι η συνάρτηση userTokenAmount
    userTokenAmount[msg.sender][findID] -= amount;
    // Μειώνει το καταγεγραμμένο υπόλοιπο
}
```

Βοηθητικές συναρτήσεις που καλούνται κατά την διάρκεια συναλλαγών μαζί με τα σχόλια επεξήγησής τους.

Αξίζει να σημειωθεί πως η μέθοδος με το _burn δεν είναι η καλύτερη εφαρμογή ενός τέτοιου συστήματος, όμως επιλέχθηκε λόγω έλλειψης χώρου.

Μια καλύτερη προσέγγιση που δεν έλαβε μέρος λόγω περιορισμού χώρου είναι η ακόλουθη:

```
function transferNFT(address recipient, uint256 id, uint256 amount) internal {
    require(balanceOf(msg.sender, id) >= amount,
        "Caller does not have enough tokens to complete transfer");
    safeTransferFrom(msg.sender, recipient, id, amount, "");
}

function operationFee(uint256 id, uint256 amount) internal {
    bankWallet[id] += amount;
    transferNFT(0x...desiredOwnerWallet, id, amount);
// Η τιμή «0x...desiredOwnerWallet» είναι παράδειγμα διεύθυνσης πορτοφολιού
// διαχείρισης
}
```

Υποθετική εναλλακτική εφαρμογή για την διατήρηση και μεταφορά νομισμάτων κατά την πώληση.

Εναλλακτικά, η καλύτερη πιθανή λύση θα απαιτούσε την συμπερίληψη του `import "@openzeppelin/contracts/token/ERC1155/Utils/ERC1155Holder.sol";` και χρήση της συνάρτησης `sendToContract`, όπως ακολούθως:

```
function sendToContract(uint256 id, uint256 amount) public {
    require(balanceOf(msg.sender, id) >= amount, "Caller does not have enough
        tokens to complete transfer");
    safeTransferFrom(msg.sender, address(this), id, amount, "");
}

function supportsInterface(bytes4 interfaceId) public view virtual
override(ERC1155, ERC1155Holder) returns (bool) {
    return super.supportsInterface(interfaceId);
}
```

Υποθετική εναλλακτική εφαρμογή για την μεταφορά νομισμάτων προς το συμβόλαιο κατά την πώληση.

Μέσα από την χρήση του συμβολαίου, η γρήγορη λήψη σημαντικών πληροφοριών επί των δεδομένων του χρήστη, από το συμβόλαιο προς το παιχνίδι, αποτελεί μια σημαντική ανάγκη. Έτσι, για την γρήγορη εύρεση όλων των νομισμάτων τα οποία ανήκαν στον χρήστη, χρησιμοποιείται η συνάρτηση `getUserTokens`. Παράλληλα, εδώ εξηγείται η δομή του «for loop» της συνάρτησης `logMint`, καθώς ο αριθμός των συνολικών νομισμάτων μέσα στο συμβόλαιο μπορεί να είναι αστρονομικός και απαιτείται ένας γρήγορος τρόπος να βρίσκονται όλα τα νομίσματα του χρήστη – αυτή είναι η δουλειά του `userTokenId`.

Το `userTokenId[user].length` ισούται με τον αριθμό των μεμονωμένων Token ID με τα οποία έχει διαδράσει στο πορτοφόλι του, ο χρήστης.

```

mapping(address => uint256[]) public userTokenId;
// Εμπεριέχει όλα τα ID των νομισμάτων με τα οποία ο χρήστης έχει διαδράσει
function getUserTokens(address user)
    public
    view
    returns (uint256[] memory, uint256[] memory)
{
    uint256 arraySize = userTokenId[user].length;

    // Αρχικοποίηση των πινάκων με το σωστό μέγεθος
    uint256[] memory ids = new uint256[](arraySize);
    uint256[] memory amounts = new uint256[](arraySize);
    for (uint256 i = 0; i < arraySize; i++) {
        ids[i] = userTokenId[user][i];
        amounts[i] = balanceOf(user, ids[i]);
    }
    //Παράγει δύο πίνακες με τις χρήσιμες τιμές
    return (ids, amounts);
}

```

Συνάρτηση `getUserTokens` για την γρήγορη εύρεση όλων των καρτών του χρήστη.

Είναι σημαντικό, σε αυτό το κομμάτι της παρουσίασης, να γίνει μια ανασκόπηση των τρόπων μέσα από τους οποίους ο χρήστης μπορεί να λάβει νέες κάρτες/νομίσματα. Για να γίνει αυτό, παρέχονται 5 τρόποι κατά το παίξιμο του παιχνιδιού:

- Κατά την είσοδο του στο παιχνίδι, μέσα από την εκτέλεση της συνάρτησης `initNewUser`
- Μέσα από την χρήση του Pause Menu στο υπο-μενού (sub-menu) Store, με την χρήση της συνάρτησης `buyFromStore`

- Κατά την λήψη βραβείων από το σύστημα Loot του παιχνιδιού, μέσα από το σύστημα giveLoot
- Μέσα από το Marketplace, αγοράζοντας νομίσματα από άλλους χρήστες, με την χρήση της συνάρτησης buyNFTfromBank

6.5.2 Αγοραπωλησία νομισμάτων / καρτών από τους χρήστες

Για την λειτουργία του συστήματος Marketplace του παιχνιδιού, χρησιμοποιείται ένα πλήθος από συναρτήσεις που δουλεύουν συμπληρωματικά, η μία προς την άλλη, για να παρέχουν την λειτουργία της αγοραπωλησίας αγαθών. Η διαδικασία ξεκινάει με την συνάρτηση listAmountOfTokensForSale, όπου ο χρήστης δηλώνει το ID των νομισμάτων που θέλει να ανεβάσει προς πώληση και το πλήθος τους, συμπληρώνει το ID και το πλήθος του νομίσματος που θέλει να λάβει σαν πληρωμή και ανεβάζει την αγγελία, αφαιρώντας το νόμισμα / κάρτα από το πορτοφόλι του. Το νόμισμα είναι ακόμα καταγεγραμμένο στον χρήστη αλλά δεν είναι ποια διαθέσιμο για χρήση ή επίδειξη, καθώς έχει καεί (burn) προσωρινά.

Έπειτα, όσο ο χρήστης το επιλέξει, μπορεί να ακυρώσει την διαδικασία πώλησης, πριν αυτή ολοκληρωθεί. Για να πάρει μέρος η διαδικασία ακύρωσης, πρέπει να κληθεί η συνάρτηση cancelSale η οποία τοποθετεί τα Token πίσω στο πορτοφόλι του χρήστη και καλεί την συνάρτηση την συνάρτηση clearBankEntry με την σειρά της. Η clearBankEntry καθαρίζει τα «sale... mapping» στην σωστή θέση, για την αφαίρεση της αγγελίας.

Τα mapping που κάνουν εφικτή την πώληση στο Marketplace είναι τα ακόλουθα. Το πρώτο uint256 αντιπροσωπεύει το κάθε ξεχωριστό Token ID του νομίσματος και δημιουργείτε ένας πίνακας ανά Token ID. Στους αντίστοιχους πίνακες, αυτούς, σπρώχνονται τιμές ταυτόχρονα, όποτε η ίδια θέση του πίνακα για το ίδιο Token ID θα επιστρέφει δεδομένα για την ίδια αγγελία σε όλα τα mapping.

```
mapping(uint256 => uint256[]) public salePrice;
mapping(uint256 => uint256[]) public saleCurrency;
mapping(uint256 => address[]) private saleOwner;
mapping(uint256 => uint256[]) public saleAmount;
```

Τα αναφερόμενα «sale... mapping», μέσα στο κείμενο

```

    mapping(address => uint256[]) public userTokenId; // Contains all Tokens user
has interacted with
    mapping(address => uint256[]) public userTokenAmount; // Contains Wallet
Information

function listAmountOfTokensForSale(
    uint256 id,
    uint256 amount,
    uint256 currency,
    uint256 price
) external returns (string memory) {
    // external is more gas efficient
//    require(!banned[msg.sender], "Account is banned");
// Λόγο περιορισμών χώρου, το σύστημα απαγόρευσης που μπλοκάρει την πρόσβαση
// στους απαγορευμένους χρήστες όπως και το σύστημα που μαρκάρει χρήστες προς
// απαγόρευση κατά την εκτέλεση «παράνομων» πράξεων, έχει αφαιρεθεί!

    require(amount>0, "Illegal value");
    require(salePrice[id].length == saleCurrency[id].length &&
        saleCurrency[id].length == saleOwner[id].length);
// Ελέγχει πως δεν υπάρχει κάποιο καταστροφικό σφάλμα στην μνήμη

    require(balanceOf(msg.sender, id) >= amount, "out of balance");

    salePrice[id].push(price);
    saleCurrency[id].push(currency);
    saleOwner[id].push(msg.sender);
    saleAmount[id].push(amount);
// Τα mappings salePrice, saleCurrency, saleOwner και saleAmount μορφοποιούν ένα
// struct ή αλλιώς, έναν πολυδιάστατο πίνακα με τις απαραίτητες τιμές για την
// υποστήριξη του Marketplace μέσα στο παιχνίδι.
// Μέσα από τις παραπάνω εντολές, σπρώχνονται οι νέες τιμές στα Mappings
    _burn(msg.sender, id, amount);
// Το _burn το οποίο αναφαίρετε στις υπόλοιπες συναρτήσεις του Marketplace

    userTokenAmount[msg.sender][findUserTokenSlot(id, msg.sender)] -= amount;
// Επειδή το userTokenAmount και το userTokenId λειτουργούν σε συνέργεια και οι
// τιμές προωθούνται άτακτα μέσα στους πίνακες, πρέπει να βρεθεί η σωστή θέση του
// πίνακα, στην οποία βρίσκεται το συγκεκριμένο ID εντός του userTokenId mapping.
// Για αυτόν τον λόγο χρησιμοποιείται η εντολή findUserTokenSlot(id, msg.sender)
// εκεί όπου κανονικά θα βρισκόταν η θέση του πίνακα με το ποσό των νομισμάτων,
// για το σωστό token ID.
    return "Transaction Complete!";
}

```

Συνάρτηση listAmountOfTokensForSale, για την δημιουργία αγγελίας από τον χρήστη.


```

function cancelSale(uint256 id, uint256 index) external
    returns (string memory){
    require(index < salePrice[id].length, "Nothing exists here!");
    require(msg.sender == saleOwner[id][index], "Illegal request.");
// Μόνο ο δημιουργός της αγγελίας μπορεί να την ακυρώσει

    inCirculation[id] -= saleAmount[id][index];
// Απαιτείται διότι κατά την ανάρτηση της αγγελίας τα νομίσματα του χρήστη κάνουν
// _burn, όμως το inCirculation[id] δεν μειώνεται, διότι τα νομίσματα δεν
// εξαφανίζονται απλά μεταφέρονται σε νέες θέσεις την μνήμης και η καταγραφή της
// ιδιοκτησίας τους, αφού δημιουργηθεί μια αγγελία, γίνεται απλά πιο πολύπλοκη
// Αυτή η μέθοδος χρησιμοποιήθηκε λόγω έλλειψης χώρου εντός του συμβολαίου,
// αναγκάζοντας τον κώδικα να συμπυκνωθεί και να βελτιστοποιηθεί από άποψη χώρου

    mint(id, saleAmount[id][index], false);
// Κατά το mint, η τιμή του inCirculation[id] αυξάνεται, το «inCirculation[id] -=
// saleAmount[id][index];» εξασφαλίζει την αποφυγή λαθών!
    clearBankEntry(id, index);
    return "Sale canceled.";
}
}

```

Συνάρτηση cancelSale, για την ακύρωση αγγελίας από τον χρήστη.

```

function clearBankEntry(uint256 id, uint256 index) internal {
// Αφαιρεί μια αγγελία από το Marketplace.
// Αν υπήρχε χώρος, θα γινόταν και αναπροσαρμογή του πίνακα, για την αφαίρεση των
// παλιών αγγελιών.

    saleOwner[id][index] = address(0);
    saleCurrency[id][index] = 0;
    salePrice[id][index] = 0;
    saleAmount[id][index] = 0;
}

```

Βοηθητική συνάρτηση clearBankEntry.

Για την εύρεση των διαθέσιμων αγγελιών ανά Token ID, χρησιμοποιείται η συνάρτηση `requestBankTokenInfo`, όπως αυτή επιδεικνύεται ακολούθως:

```
function requestBankTokenInfo(uint256 id) public view
    returns (
        uint256[] memory,
        uint256[] memory,
        uint256[] memory,
        uint256
    ){
    require(salePrice[id].length == saleCurrency[id].length &&
        saleCurrency[id].length == saleOwner[id].length);
    // Ελέγχει πως δεν υπάρχει κάποιο καταστροφικό σφάλμα στην μνήμη

    require(salePrice[id].length > 0, "Bank Supply: 0");

    // Αρχικοποιεί τους πίνακες για να περαστούν τα δεδομένα επιστροφής
    uint256 userAdd = salePrice[id].length;
    uint256[] memory prices = new uint256[](userAdd);
    uint256[] memory currencies = new uint256[](userAdd);
    uint256[] memory amounts = new uint256[](userAdd);

    // Γεμίζει τους πίνακες με τα χρήσιμα δεδομένα
    for (uint256 i = 0; i < salePrice[id].length; i++) {
        prices[i] = salePrice[id][i];
        currencies[i] = saleCurrency[id][i];
        amounts[i] = saleAmount[id][i];
    }
    return (currencies, prices, amounts, userAdd);
}
```

Συνάρτηση `requestBankTokenInfo`.

Η αγορά των νομισμάτων ολοκληρώνεται μέσα από την συνάρτηση buyNFTfromBANK. Το id και το index που έχει σαν εισόδους, αντιστοιχούν στα «sale... mappings».

```
function buyNFTfromBANK(uint256 id, uint256 index) external
    returns (string memory){    // external is more gas efficient

//    require(banned[msg.sender] == false, "Account is banned");
// Λόγο περιορισμών χώρου, το σύστημα απαγόρευσης που μπλοκάρει την πρόσβαση
// στους απαγορευμένους χρήστες όπως και το σύστημα που μαρκάρει χρήστες προς
// απαγόρευση κατά την εκτέλεση «παράνομων» πράξεων, έχει αφαιρεθεί!

    require(index < salePrice[id].length, "Nothing exists here!");
// index specifies slot in bank struct
    require(salePrice[id].length == saleCurrency[id].length &&
        saleCurrency[id].length == saleOwner[id].length);
// Ελέγχει πως δεν υπάρχει κάποιο καταστροφικό σφάλμα στην μήμη
    require(balanceOf(msg.sender, saleCurrency[id][index]) >=
        salePrice[id][index], "Not enough Currency");
// Ελέγχει το πορτοφόλι του χρήστη για τους απαραίτητους πόρους
    require(salePrice[id][index] > 0);

    safeTransferFrom(msg.sender, saleOwner[id][index],
        saleCurrency[id][index], salePrice[id][index], "");
// Εκτελεί την «πληρωμή» προς τον πωλητή
    logMint(saleOwner[id][index], saleCurrency[id][index],
        salePrice[id][index]);
// Ανανεώνει τα δεδομένα ιδιοκτησίας, από το safeTransferFrom, για τον πωλητή
// Μια διαδικασία που δεν παίρνει μέρος για τον αγοραστή, αφού τα νομίσματα προς
// πώληση κάνουν _burn με την δημιουργεί της αγγελίας, για να σταματήσουν να
// είναι σε χρήση

    mint(id, saleAmount[id][index], false);
    clearBankEntry(id, index);
// Συνάρτηση που αφαιρεί την αγγελία μετά την ολοκλήρωση της πώλησης
    return "Transaction Complete!";
}
```

Συνάρτηση buyNFTfromBANK

6.5.3 Σύστημα επιβράβευσης «Loot» και αγοράς πακέτων καρτών

Ένα σημαντικό στοιχείο κάθε παιχνιδιού είναι η επιβράβευση των παικτών για τις πράξεις τους μέσα από «Loot» ή «Achievements». Για την υλοποίηση αυτού του μηχανισμού δημιουργήθηκε η συνάρτηση `giveLoot` και η συνάρτηση `setRewardTrigger` η οποία θέτει τα δεδομένα που χρησιμοποιεί η `giveLoot`. Κατά την ολοκλήρωση ενός επιτεύγματος καλείται η συνάρτηση `giveLoot`. Ταυτόχρονα, ο ίδιος μηχανισμός χρησιμοποιείται για την πώληση πακέτων με νομίσματα στον χρήστη, λόγω έλλειψης χώρου.

```

mapping(address => mapping(uint256 => bool[])) public voucher;
// Δομή: (Για κάθε χρήστη => για κάθε παιχνίδι => υπάρχει ένας πίνακας με όλα τα
// triggers που μπορούν να ενεργοποιηθούν για να λάβει ο παίκτης rewards
// ή loot, με την σειρά δημιουργίας τους)
// address = χρήστης || uint256 = Game ID || bool[] = Reward Received Status
// ΠΡΟΣΟΧΗ: Όταν το Voucher είναι true, η συνάρτηση giveLoot δεν ενεργοποιείται.
// Στην ουσία το Voucher σε αφήνει να τρέξεις μία μόνο φορά την giveLoot.
// Το σύστημα με τα voucher χρησιμοποιείται για την λειτουργία της giveLoot.
// Τα voucher μπορούν να είναι επιβραβεύσεις για τον παίκτη που δίνονται όταν
// αυτός ολοκληρώνει συγκεκριμένους στόχους μέσα στο παιχνίδι (tasks).
// Μπορούν να είναι achievements (με ή χωρίς νομίσματα επιβράβευσης).
// Μπορούν να είναι πακέτα με αγαθά προς πώληση από κατάστημα του παιχνιδιού.

mapping(uint256 => bool[]) private rewardIsGameGift;
// Όταν είναι true η giveLoot μπορεί να ενεργοποιηθεί μια μόνο φορά και η
// κατάσταση του voucher θα αλλάξει, έπειτα, σε true. Χρησιμοποιείται για να
// δώσει Loot στον παίκτη.

mapping(uint256 => bool[]) private rewardIsTokens;
// Όταν false δεν δίνεται reward και η giveLoot αποτυγχάνει. Μπορεί να
// αλλάξει κατάσταση μεταγενέστερα μέσα από την συνάρτηση disableRewardTrigger

mapping(uint256 => mapping(uint256 => uint256[])) private packs;
// gameID => ID πακέτου => ID καρτών μέσα στο πακέτο
mapping(uint256 => mapping(uint256 => uint256[])) private packTokenAmount;
// gameID => ID πακέτου => ποσότητα καρτών μέσα στο πακέτο
// (βοηθάει το από πάνω mapping)
mapping(uint256 => uint256[]) private packPrice;
// gameID => τιμή πακέτου (κάθε ID πακέτου αντιστοιχεί στο index του πίνακα)

mapping(uint256 => address) internal userLog; //contains all users

```

Τα mappings που χρησιμοποιούνται για την υποστήριξη των λειτουργιών της συνάρτησης `giveLoot`

Το voucher επίσης χρησιμοποιείται για τον έλεγχο των Achievements. Για παράδειγμα, για την δημιουργία ενός κλασσικού Achievement χωρίς αντικείμενα επιβράβευσης πρέπει: να κληθεί η συνάρτηση setRewardTrigger με isGameReward true και tokenID 0, όταν το giveLoot κληθεί θα κάνει την μεταβλητή voucher true αλλά δεν θα εκτελεστεί η _mint. Σε ένα άλλο παράδειγμα, isGameReward true και tokenID 87 με tokenAmount 1 μπορεί να δώσει ένα μοναδικό αντικείμενο στον χρήστη που το λαμβάνει μόνο μέσα από την ολοκλήρωση του πακέτου μία φορά.

Για την λειτουργία και την ρύθμιση των μεταβλητών του giveLoot προαπαιτείται η χρήση της συνάρτησης setRewardTrigger.

```
function setRewardTrigger(
    uint256 gameID,
    bool isGameReward,
    uint256 price,
    bool _rewardIsTokens,
    uint256[] memory tokenID,
    uint256[] memory tokenAmount
) public onlyOwner returns (uint256, uint256) {
    require(tokenID.length == tokenAmount.length);
    // Ελέγχει πως δεν υπάρχει κάποιο λάθος στις δοσμένες τιμές
    uint256 voucherID;
    rewardIsTokens[gameID].push(_rewardIsTokens);
    rewardIsGameGift[gameID].push(isGameReward);
    // Όταν είναι true -> η giveLoot μπορεί να ενεργοποιηθεί μια μόνο φορά
    packPrice[gameID].push(price);
    // Για δωρεάν πακέτα, επιβράβευση ή achievement = πρέπει να τεθεί ως 0
    voucherID = rewardIsTokens[gameID].length - 1;
    for (uint256 i = 0; i < tokenID.length; i++) {
        // parses pack reward info
        packs[gameID][voucherID].push(tokenID[i]);
        packTokenAmount[gameID][voucherID].push(tokenAmount[i]);
    }
    for (uint256 i = 0; i < totalUsers; i++) {
        // προσθέτει νέο "achievement" BOOL σε όλους τους χρήστες που έχουν το παιχνίδι
        if (userHasGame[userLog[i]][gameID] == true) {
            voucher[userLog[i]][gameID].push(false);
        }
    }
    // χρησιμοποιείται το userLog που είχε τεθεί στην συνάρτηση initNewUser
    }
    // επιστρέφει χρήσιμες πληροφορίες για τους προγραμματιστές
    return (voucherID, voucher[msg.sender][gameID].length - 1);
}
```

Συνάρτηση δημιουργίας πακέτων καρτών, επιβραβεύσεων και Achievement

```
function disableRewardTrigger(uint256 gameID, uint256 rewardID)
    public
    onlyOwner
{
    rewardIsTokens[gameID][rewardID] = false;
}
```

Συνάρτηση disableRewardTrigger η οποία χρησιμοποιείται για την απενεργοποίηση της giveLoot για συγκεκριμένες καταχωρήσεις.

Ακολούθως παρουσιάζεται η συνάρτηση giveLoot και οι λειτουργίες της.

```
function giveLoot(uint256 gameID, uint256 rewardID, uint256 packVoucherAmount
) public {
    require(gameExists[gameID] == true);
    require(userHasGame[msg.sender][gameID] == true);

    require(voucher[msg.sender][gameID][rewardID] == false);
    // Χρησιμοποιείται για επιβραβεύσεις που μπορούν να εκτελεστούν μόνο μια φορά
    require(rewardIsTokens[gameID][rewardID] == true);
    // Χρησιμοποιείται για απενεργοποίηση μιας επιβράβευσης σε δεύτερο χρόνο

    if (rewardIsGameGift[gameID][rewardID] == true) {
    // Εκτελείται η συνθήκη για επιβραβεύσεις που εκτελούνται μόνο μια φορά
        voucher[msg.sender][gameID][rewardID] = true;
        if (packs[gameID][rewardID][0] != 0) {
    // Το Token ID 0 έχει κρατηθεί για την δήλωση των Achievements που δεν δίνουν
    // πίσω κάποιο βραβείο κατά την ολοκλήρωσή τους
            givePackReward(gameID, rewardID, 1);
        }
    // Δίνει το Loot

    } else if (packVoucherAmount > 0) {
        givePackReward(gameID, rewardID, packVoucherAmount);
        operationFee(mainCurrency,
            packPrice[gameID][rewardID] * packVoucherAmount);
    // Χρησιμοποιείται για να χρεώσει
    }
}
```

Συνάρτηση giveLoot

```

function givePackReward(
  uint256 game,
  uint256 rewardID,
  uint256 amount
) internal {
  for (
    uint256 ii = 0;
    ii < packTokenAmount[game][rewardID].length;
    ii++
  ) {
    uint256 helper = packs[game][rewardID][ii];
    mint(helper, packTokenAmount[game][rewardID][ii] * amount, false);
  }
}

```

Υποστηρικτική συνάρτηση

6.6 Επιπλέον λειτουργίες και επιπρόσθετος κώδικας πειραμάτων

Υπολείπονται φωτογραφίες κώδικα που δεν χρησιμοποιήθηκε από σύστημα Banning και Entity Creation στο Smart Contract από χρήστες.

+ Blueprints με έξτρα λειτουργίες που δεν χρησιμοποιήθηκαν στο τέλος.

Κεφάλαιο 7 Συμπεράσματα

Μέσα από την θεωρητική ανάλυση των πρώτων 5 κεφαλαίων και στο μεγαλύτερο μέρος του ολικού σώματος του έργου, γίνεται εμφανές πως τα συστήματα με τεχνολογία Blockchain δεν έχουν κάποια συγκεκριμένη μορφή, αλλά παρέχουν ένα σύνολο από λειτουργίες και χαρακτηριστικά που συνήθως μοιράζονται μεταξύ τους. Το βασικότερο αυτών των χαρακτηριστικών είναι η αμετάβλητη φύση των δεδομένων που σειριακά περνάνε μέσα στα blocks και η μονιμότητα των διεργασιών που συνήθως συσχετίζονται με αυτά ή προκύπτουν μέσα από Hashing.

Για τα βιντεοπαιχνίδια, μέρος των τεχνολογιών του Blockchain (ή παρόμοια συστήματα) χρησιμοποιούνταν εδώ και αρκετά χρόνια για την επαλήθευση πακέτων, τον διαμοιρασμό αρχείων ανάμεσα στους παίκτες, την οντοποίηση, την δημιουργία ομότιμων δικτύων, τα online παιχνίδια και πολλές άλλες εφαρμογές.

Μερικά από τα οφέλη και ευκαιρίες που φέρουν αυτές οι εφαρμογές είναι ήδη εμφανή στην οικονομία. Όπως τα Gaas παιχνίδια (Games as a service), τα κυβερνο-φυσικά συστήματα όπως το Labo της Nintendo, το Just Dance με το Kinect, το Pokémon Go, παιχνίδια καμπίνας σε arcade ή σε θεματικά πάρκα με ρομποτικά συστήματα και πολλές άλλες εφαρμογές. Διότι, η τεχνολογία αιχμής πολλές φορές έβρισκε μια ενεργή αγορά στον κλάδο των βιντεοπαιχνιδιών, έτσι πάντα υπήρχε επαφή και εφαρμογή νέων τεχνολογιών στους ευφάνταστους κόσμους των παιχνιδιών.

Όμως δεν απορροφούν νέες τεχνολογίες όλοι οι κλάδοι με τον ίδιο ρυθμό που το κάνουν τα βιντεοπαιχνίδια. Τα βασικά πλεονεκτήματα της χρήσης των NFT και της τεχνολογίας Blockchain έρχονται μέσα από τις νέες δυνατότητες διασύνδεσης κλάδων, συστημάτων και υπηρεσιών με τον κλάδο των βιντεοπαιχνιδιών. Αντίστροφα, μέσα από την ένταξη λογικών μονάδων, συστημάτων τεχνητής νοημοσύνης και γενικότερα τεχνολογίες της Βιομηχανίας 4.0 έρχεται η δυνατότητα παιχνιδοποίησης (Gamification) των υπηρεσιών και διεργασιών.

Μέσα από την παιχνιδοποίηση, τα αγαθά αποκτούν συνέχεια και εξέλιξη και ο χρήστης διαδράμαζι τους για μεγαλύτερα χρονικά διαστήματα, η λειτουργία εργαλείων και η εκτέλεση εργασιών γίνεται πιο διασκεδαστική και τα συστήματα πιο ανθρώπινα και διαδραστικά.

Η τεχνολογία Blockchain και τα NFTs, προωθούν την παιχνιδοποίηση και την ένταξη περίπλοκων ψηφιακών συστημάτων σε φυσικές δομές που άλλοτε δεν είχαν καμία μορφή διασύνδεσης με τον ψηφιακό κόσμο. Μέσα από τις τεχνολογίες όπως το EVM και άλλα συστήματα που χρησιμοποιούν το Blockchain και φέρουν αρχιτεκτονική μηδενικής εμπιστοσύνης, είναι τώρα πιο εφικτό από ποτέ τα συστήματα να αποκτήσουν υπολογιστική λογική, να εκδίδουν μοναδικά τεκμήρια και να μιλάνε μεταξύ τους έμπιστα, χωρίς μεσάζοντες.

Είναι όμως, επίσης, εμφανές πως τα Blockchain έχουν περιορισμούς ως προς την χρήση τους με τα παιχνίδια. Στην περίπτωση που φέρουν αποκεντρωμένα χαρακτηριστικά η ταχύτητα απόκρισης και ολοκλήρωσης των συναλλαγών δημιουργεί προβλήματα. Αντίθετα στην περίπτωση της πλήρους κεντροποίησης δεν έχουν κάποιο ουσιαστικό πλεονέκτημα πάνω από τις υψηλά βελτιστοποιημένες και γρήγορες προγραμματιστικές ρουτίνες των μοντέρνων παιχνιδιών που τρέχουν σε Data Centers.

Αυτό που είναι τώρα εφικτό, τόσο στα φυσικά συστήματα μεταξύ τους, αλλά και στα παιχνίδια προς τα φυσικά συστήματα, είναι η δυνατότητα επικοινωνίας με κοινούς όρους ανάμεσα σε

άγνωστες οντότητες, με τους όρους των συμφωνιών απόλυτα διατυπωμένους σε κώδικα. Αυτό επιτρέπει σε οντότητες, που υπό κανονικές συνθήκες δεν θα συνέπρατταν συμφωνίες, να συνεργάζονται σε παγκόσμιο επίπεδο και να ανταλλάσσουν δεδομένα και πόρους. Συγκεκριμένα για τα παιχνίδια, όταν υπήρχε διασύνδεση στα παιχνίδια ή διαλειτουργικά στοιχεία (όπως την μεταφορά του χαρακτήρα σε παιχνίδι ρόλων στην επόμενη έκδοση του παιχνιδιού μέσα από την απορρόφηση του παλιού save file), αυτή η διασύνδεση παρέμενε μόνο εντός των παιχνιδιών του ίδιου Studio ή εταιρίας.

Πλέον, όμως, ακόμα και η μορφοποίηση ενός κοινού χώρου, που μοιράζονται όλα τα παιχνίδια, είναι δυνατή. Το metaverse με διαλειτουργικά στοιχεία είναι τώρα εντός του πλαισίου των εφικτών και υλοποιήσιμων συστημάτων. Αυτή η νέα δυνατότητα «υπερ-δικτύωσης» είναι δυνατή μέσα από το προαναφερόμενο έμπιστο σύστημα χωρίς μεσάζοντες αλλά και τα συστήματα Blockchain με δυνατότητες smart contract και NFT. Συγκεκριμένα, τα NFT προσφέρουν ένα από τα τελευταία κομμάτια του puzzle της διαλειτουργικότητας, το οποίο είναι το κοινό πλαίσιο χαρακτηρισμού οντοτήτων με μεταδεδομένα που θυμίζουν τον χαρακτηρισμό των μεταβλητών στα συστήματα μετα-προγραμματισμού με Reflection.

Τα NFT συμβολίζουν τα αντικείμενα του αντικειμενοστρεφή προγραμματισμού ή τα Entities (Actors) και Components που γεμίζουν τον ψηφιακό χώρο. Και αυτές οι ψηφιακές οντότητες μπορούν να συνδέονται και να απεικονίζουν φυσικές οντότητες.

Συγκεκριμένα μέσα από το πειραματικό παιχνίδι που δημιουργήθηκε, έγινε εφικτή η παρατήρηση διάφορων πλεονεκτημάτων που προσφέρουν μέσα από την παροχή νέων λειτουργιών και δυνατοτήτων. Μέσα από την χρήση των NFT και του συστήματος των Smart Contract ήταν δυνατή η ένταξη συστημάτων πώλησης αγαθών χωρίς την ανάληψη της ευθύνης της ασφάλειας του συστήματος και χωρίς την χρήση μεσάζοντα για την λειτουργία του συστήματος των συναλλαγών. Οι καταναλωτές είναι πιο πιθανό το να εμπιστευτούν την διαδικασία αγοράς ενός ψηφιακού αγαθού μέσα από την πλατφόρμα ενός παιχνιδιού, όταν βλέπουν το έμπιστο δίκτυο Ethereum και τα χρήματα μεταφέρονται χωρίς μεσάζοντες. Επίσης τα διαλειτουργικά στοιχεία προωθούν την αγορά και χρήση υπηρεσιών και αντικειμένων από μικρότερα Studio, καθώς μπορούν να μεταφερθούν και σε πιο δημοφιλή παιχνίδια μεταγενέστερα.

Ως προς τα όντα του παιχνιδιού, το κάθε ένα ξεχωριστά αποτελεί συνάλλαγμα και μπορεί να ανταλλαχτεί με οποιοδήποτε πλήθος άλλων οντοτήτων. Όταν αυτό το ον αποκτηθεί παραμένει στο πορτοφόλι του χρήστη για πάντα, ακόμα και αν το παιχνίδι σταματήσει την υποστήριξη του. Επίσης, η σύνδεση και η παροχή αυτών των υπηρεσιών δεν φέρει κόστος υποδομών, όπως στην περίπτωση του Ethereum, και ο χρήστης επωμίζεται όλο το κόστος λειτουργίας του δικτύου μόνο και αν το χρησιμοποιεί – αυτό σημαίνει πως δεν απαιτούνται ανοιχτοί servers και συστηματικές αναβαθμίσεις για λόγους ασφαλείας.

Πολλές από τις δυνατότητες που εισάγει η τεχνολογία Blockchain, αποτελούν εντελώς νέα στοιχεία και δεν μπορούν να συγκριθούν με παλαιότερες λύσεις. Όμως σε ότι αφορά την σύμπραξη συμφωνιών, τα συστήματα Blockchain αλλάζουν ολόκληρο το οικονομικό πεδίο. Παλαιότερα, μια συμφωνία θα απαιτούσε πολλαπλούς συμβολαιογράφους και δικηγόρους, γραφειοκρατία, τράπεζες και έμπιστους ενδιάμεσους και πάλι, η αποφυγή της απάτης ίσως να μην ήταν εφικτή. Τώρα η συμφωνία είναι απόλυτα διατυπωμένη μέσα σε μία έμπιστη οντότητα και μπορεί και να αυτό-εκτελεστεί και τα συναλλασσόμενα αγαθά να έχουν μοναδικά τεκμήρια στο δίκτυο.

Κεφάλαιο 8 Βιβλιογραφικές Αναφορές

- [1] N. Anjum, D. Karamshuk, M. Shikh-Bahaei, and N. Sastry, "Survey on peer-assisted content delivery networks," *Computer Networks*, vol. 116, pp. 79-95, 2017. DOI: 10.1016/j.comnet.2016.11.008.
- [2] Manal El Dick and Esther Pacitti, "Content Distribution in P2P Systems," *Research Report RR-7138*, INRIA, 2009. [Online]. Available: <https://inria.hal.science/inria-00439171>.
- [3] A. Oram, *Peer-to-Peer: Harnessing the Power of Disruptive Technologies*, O'Reilly Media, 2001. ISBN: 978-0596001100.
- [4] A. Mauthe and D. Hutchison, "Peer-to-Peer Computing: Systems, Concepts and Characteristics," *PIK - Praxis der Informationsverarbeitung und Kommunikation*, vol. 26, pp. 60-66, 2003. DOI: 10.1515/PIKO.2003.60.
- [5] J. Pouwelse, P. Garbacki, D. Epema, and H. Sips, "The Bittorrent P2P File-sharing System: Measurements and Analysis," in *Proceedings of the 4th International Workshop on Peer-to-Peer Systems (IPTPS 2005)*, Ithaca, NY, USA, 2005. DOI: 10.1007/11558989_17.
- [6] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2009. [Online]. Available: <https://bitcoin.org/bitcoin.pdf>.
- [7] World Bank Group, "Distributed Ledger Technology (DLT) and Blockchain," 2017. [Online]. Available: <https://documents.worldbank.org/curated/en/177911513714062215/Distributed-Ledger-Technology-DLT-and-blockchain>.
- [8] K. Solorio, R. Kanna, and D. H. Hoover, *Hands-On Smart Contract Development with Solidity and Ethereum*, O'Reilly Media, 2019. ISBN: 9781492045267.
- [9] D. Tapscott and A. Tapscott, *Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies Is Changing the World*. New York, NY, USA: Penguin Random House, 2016.
- [10] R. da R. Righi, A. M. Alberti, and M. Singh, "Blockchain Technology for Industry 4.0," 2020.

- [11] S. Goldwasser and M. Bellare, Lecture Notes on Cryptography. Cambridge, MA: MIT, July 2008.
- [12] A. Hayes, "Blockchain Facts: What Is It, How It Works, and How It Can Be Used," Investopedia, Jun. 24, 2024. [Online]. Available: <https://www.investopedia.com/terms/b/blockchain.asp>.
- [13] V. Buterin, "A next-generation smart contract and decentralized application platform," Ethereum White Paper, 2014. [Online]. Available: <https://ethereum.org/en/whitepaper/>.
- [14] P. Mukherjee and C. Pradhan, Blockchain Technology: Applications and Challenges. Springer, 2021.
- [15] S. Nakov, Practical Cryptography for Developers. Nakov.com, 2018. Available: <https://cryptobook.nakov.com>.
- [16] W. Penard and T. van Werkhoven, Cryptography in Context. Utrecht, The Netherlands: Universiteit Utrecht, 2008.
- [17] K. Czarnecki and U. Eisenecker, Generative Programming: Methods, Tools, and Applications. Boston, MA: Addison-Wesley, 2000.
- [18] "OpenZeppelin Contracts," OpenZeppelin, [Online]. Available: <https://www.openzeppelin.com/contracts>.
- [19] D. Di Gennaro, Advanced Metaprogramming in Classic C++. Berkeley, CA: Apress, 2015.
- [20] R. Anisko, "Towards Automatic Partial Evaluation for the C++ Language," Technical Report, 2002.
- [21] N. Szabo, "Smart Contracts: Building Blocks for Digital Markets," Extropy, vol. 16, 1996.
- [22] Ethereum Foundation, "What is a smart contract?" Ethereum.org, 2023.
- [23] National Archives and Records Administration, "Non-Fungible Tokens (NFTs) Briefing Paper," NARA, Apr. 2024. [Online]. Available: <https://www.archives.gov/files/records-mgmt/policy/nft-briefing-paper-april-2024.pdf>.
- [24] A.-D. Popescu, "Non-Fungible Tokens (NFT) - Innovation beyond the craze," in Proceedings of the 5th International Conference on Innovation in Business, Economics & Marketing Research, vol. 66, 2019, pp. 26-30. [Online]. Available: https://www.researchgate.net/publication/353973149_Non-Fungible_Tokens_NFT_-_Innovation_beyond_the_craze.

- [25] F. Regner, N. Urbach, and A. Schweizer, "NFTs in Practice – Non-Fungible Tokens as Core Component of a Blockchain-based Event Ticketing Application," in Proceedings of the 40th International Conference on Information Systems (ICIS), Munich, Germany, Dec. 2019. [Online]. Available: <https://aisel.aisnet.org/icis2019/1>.
- [26] "IPFS - Content Addressed, Versioned, P2P File System" Juan Benet 2014
arXiv:1407.3561
- [27] B. Edgington, "The Eth2 Book," 2023. [Online]. Available:
<https://github.com/benjaminion/upgrading-ethereum-book>
- [28] I. Bashir, *Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications*, 4th ed. Packt Publishing, 2023.
- [29] Tim Sweeney, "The State of Unreal 2022 Keynote Presentation", 2022
- [30] Epic Games, "Unreal Engine Documentation," Epic Developer Community, 2023.
[Online]. Available: <https://dev.epicgames.com/documentation/>
- [31] Epic Games, Inc., "Programming with C++ in Unreal Engine," Unreal Engine Documentation, ver. 5.3, June 2023. [Online]. Available:
https://dev.epicgames.com/documentation/en-us/unreal-engine/programming-with-cplusplus-in-unreal-engine?application_version=5.3.
- [32] Epic Games, Inc., "Actors in Unreal Engine," Unreal Engine Documentation, ver. 5.3, June 2023. [Online]. Available: https://dev.epicgames.com/documentation/en-us/unreal-engine/actors-in-unreal-engine?application_version=5.3.
- [33] D. Abrahams and A. Gurtovoy, C++ Template Metaprogramming: Concepts, Tools, and Techniques from Boost and Beyond. Boston, MA: Addison-Wesley Professional, 2004.
- [34] Epic Games, Inc., "Unreal Engine 5.3 Documentation," Epic Developer Community, 2023. [Online]. Available: <https://dev.epicgames.com/documentation/en-us/unreal-engine/unreal-engine-5-3-documentation>.
- [35] Epic Games, Inc., "Scripting the Unreal Editor using Python," Unreal Engine Documentation, 2024. [Online]. Available: https://dev.epicgames.com/documentation/en-us/unreal-engine/scripting-the-unreal-editor-using-python?application_version=5.3.
- [36] CERN, "Cling - The Interactive C++ Interpreter," 2024. [Online]. Available:
<https://root.cern/cling/>.

- [37] H. Finkel, D. Poliakoff, and D. F. Richards, "ClangJIT: Enhancing C++ with Just-in-Time Compilation," in Proceedings of the SC19: International Conference for High Performance Computing, Networking, Storage, and Analysis, Denver, CO, USA, Nov. 2019. [Online]. Available: https://sc19.supercomputing.org/proceedings/workshops/workshop_pages/ws_p3hpc107.html.
- [38] V. Vassilev, "Interactive C++ with Cling," The LLVM Project Blog, Nov. 30, 2020. [Online]. Available: <https://blog.llvm.org/posts/2020-11-30-interactive-cpp-with-cling/>.
- [39] J. Glazer and S. Madhav, Multiplayer Game Programming: Architecting Networked Games. Boston, MA, USA: Addison-Wesley, 2015.
- [40] H. Sutter, "Keynote: The Evolution of C++ - A Typescript for C++," presented at C++Now, Aspen, CO, USA, May 2023. [Online]. Available: <https://www.youtube.com/watch?v=fJvPBHErF2U>.
- [41] H. Sutter, "cppfront," GitHub, 2024. [Online]. Available: <https://github.com/hsutter/cppfront>.
- [42] Epic Games, Inc., "Unreal Engine Reflection System," Unreal Engine Developer Documentation, 2024. [Online]. Available: <https://dev.epicgames.com/documentation/en-us/unreal-engine/reflection-system-in-unreal-engine>.
- [43] M. Noland, "Unreal Property System (Reflection)," Unreal Engine Developer Blog, Mar. 2014. [Online]. Available: <https://www.unrealengine.com/en-US/blog/unreal-property-system-reflection>.
- [44] R. De Ioris, "LuaMachine," GitHub, 2024. [Online]. Available: <https://github.com/rdeioris/LuaMachine>.
- [45] J. Hamilton Ortiz, Ed., "Industry 4.0 - Current Status and Future Trends," IntechOpen, Mar. 25, 2020. doi: 10.5772/intechopen.86000.
- [46] McKinsey & Company, "McKinsey Technology Trends Outlook 2023," McKinsey Digital, Jul. 2023. [Online]. Available: <https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-top-trends-in-tech#new-and-notable>.
- [47] N. Wiener, Cybernetics: Or Control and Communication in the Animal and the Machine. Cambridge, MA: MIT Press, 1948.
- [48] W. R. Ashby, An Introduction to Cybernetics. London, U.K.: Chapman & Hall, 1956.

- [49] S. Beer, *Decision and Control: The Meaning of Operational Research and Management Cybernetics*. New York, NY: Wiley, 1966.
- [50] A. Fereidunian, M. Lehtonen, H. Lesani, C. Lucas, and M. Nordman, "Adaptive autonomy: Smart cooperative cybernetic systems for more humane automation solutions," in *2007 IEEE International Conference on Systems, Man and Cybernetics*, Montreal, QC, Canada, 2007, pp. 202-207. doi: 10.1109/ICSMC.2007.4413874.
- [51] P. Leitão, A. W. Colombo, and S. Karnouskos, "Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges," *Computers in Industry*, vol. 81, pp. 11-25, Sep. 2016, doi: 10.1016/j.compind.2015.08.004.
- [52] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA: MIT Press, 2016.
- [53] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY: Springer, 2006.
- [54] R. V. Yampolskiy, *Artificial Superintelligence: A Futuristic Approach*. Boca Raton, FL: CRC Press, 2013.
- [55] A. de Souza e Silva and R. Glover-Rijkse, Eds., *Hybrid Play: Crossing Boundaries in Game Design, Players Identities and Play Spaces*. New York, NY: Routledge, 2020. ISBN: 9781032400587.
- [56] M. Derian, *Cognitive Prosthetics*. Oxford, U.K.: Elsevier, 2018. ISBN: 978-1785482953.
- [57] R. V. Nuncio and J. M. B. Felicilda, "Cybernetics and Simulacra: The Hyperreality of Augmented Reality Games," *Kritike: An Online Journal of Philosophy*, vol. 15, no. 2, pp. 39-67, Dec. 2021.
- [58] R. Richards, "Interactivity and Video games: A Cyberarchitecture framework," 2021.
- [59] H. Rahaman and B.-K. Tan, "Interactive Space: Searching for A Dual Physical-Virtual World," in *CAADRIA 2009 Proceedings of the 14th International Conference on Computer Aided Architecture Design Research in Asia: Between Man and Machine*, Yunlin, Taiwan, 2009, pp. 675-684.
- [60] K. Salen and E. Zimmerman, *Rules of Play: Game Design Fundamentals*, Chapter 18: "Games as Cybernetic Systems". Cambridge, MA: MIT Press, 2004. ISBN: 9780262240451.

- [61] Y. N. Aldeoes, P. Gokhale, and S. Y. Sondkar, Eds., *AI, IoT, Big Data and Cloud Computing for Industry 4.0*. Cham, Switzerland: Springer, 2023.
- [62] A. Harvey, "Seeking the Embodied Mind in Video Game Theory: Embodiment in Cybernetics, Flow, and Rule Structures," *Journal of the Canadian Game Studies Association*, vol. 3, no. 4, 2009. [Online]. Available: <https://www.cgsajournal.ca/issue/view/3>.
- [63] S. Bakkes, C. T. Tan, and Y. Pisan, "Personalised gaming: A motivation and overview of literature," in *Proceedings of The 8th Australasian Conference on Interactive Entertainment*, 2012, pp. 1-10.
- [64] J. Kücklich, "The Study of Computer Games as a Second-Order Cybernetic System," in *Computer Games and Digital Cultures Conference Proceedings*, Tampere, Finland, 2002. [Online]. Available: <https://dl.digra.org/index.php/dl/article/view/18>.
- [65] B. Slusser, "Leveraging Edge Computing's Power in Industry 4.0," *Control Engineering*, 2024. [Online]. Available: <https://www.controleng.com/articles/leveraging-edge-computings-power-in-industry-4-0/>.
- [66] G. N. Yannakakis and J. Togelius, *Artificial Intelligence and Games*. Cham, Switzerland: Springer, 2018.
- [67] T. Koulouri and S. Lauria, "Exploring the role of interaction context on social dynamics in human-robot communication," *Computers in Human Behavior*, vol. 50, pp. 275-283, Sep. 2015.
- [68] M. Sailer, J. Hense, H. Mandl, and M. Klevers, "Fostering learning and motivation with educational video games: The role of the narrative context and individual differences," *Computers & Education*, vol. 113, pp. 115-129, Oct. 2017.
- [69] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," *Bitcoin White Paper*, 2008. [Online]. Available: <https://bitcoin.org/bitcoin.pdf>.
- [70] N. Szabo, "The Idea of Smart Contracts," *Nick Szabo's Blog*, 1997. [Online]. Available: <https://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html>.
- [71] V. Buterin, "A Next-Generation Smart Contract and Decentralized Application Platform," 2014. [Online]. Available: <https://ethereum.org/en/whitepaper/>.

- [72] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," arXiv preprint, arXiv:1608.00695, 2016. doi: 10.48550/arXiv.1608.00695. [Online]. Available: <https://arxiv.org/abs/1608.00695>.
- [73] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-Resolution Image Synthesis with Latent Diffusion Models," arXiv preprint arXiv:2112.10752, Dec. 2021. [Online]. Available: <https://arxiv.org/abs/2112.10752> <https://github.com/CompVis/stable-diffusion>
- [74] LAION, "Large-scale Artificial Intelligence Open Network," [Online]. Available: <https://laion.ai/>.
- [75] B. Siciliano and O. Khatib, Eds., Springer Handbook of Robotics, 2nd ed. Springer, 2016. doi: 10.1007/978-3-319-32552-1
- [76] I. Gibson, D. W. Rosen, and B. Stucker, Additive Manufacturing Technologies: 3D Printing, Rapid Prototyping, and Direct Digital Manufacturing, 2nd ed. Cham, Switzerland: Springer, 2015. doi: 10.1007/978-1-4939-2113-3.
- [77] Tencent ARC, "GFPGAN – Practical Algorithm for Real-world Face Restoration," GitHub, 2024. [Online]. Available: <https://github.com/TencentARC/GFPGAN>.
- [78] "Low-rank Adaptation for Fast Text-to-Image Diffusion Fine-tuning," GitHub, 2024. [Online]. Available: <https://github.com/cloneofsimon/lora>.
- [79] AUTOMATIC1111, "Stable Diffusion web UI," GitHub, 2024. [Online]. Available: <https://github.com/AUTOMATIC1111/stable-diffusion-webui>.
- [80] BitDAO, "BIP-5.1 G7P-1: Game7 Bootstrapping Phase," BitDAO Discourse, Aug. 2022. [Online]. Available: <https://discourse.bitdao.io/t/passed-bip-5-1-g7p-1-game7-bootstrapping-phase/2836>.
- [81] T. Gochev, "What is BitDAO?" CoinGecko Learn, 2023. [Online]. Available: <https://www.coingecko.com/learn/bitdao-what-you-need-to-know-about-the-investment-dao-with-billions-in-assets>.
- [82] R. Raut, S.-d. Krit, and P. Chatterjee, Eds., Machine Vision for Industry 4.0: Applications and Case Studies. New York, NY, USA: Routledge, 2023. ISBN: 9780367641641.
- [83] G7DAO, "web3.unreal," GitHub, 2024. [Online]. Available: <https://github.com/G7DAO/web3.unreal>.

- [84] HyperPlay, "HyperPlay - The Web3 Gaming Launcher," 2024. [Online]. Available: <https://www.hyperplay.xyz/>.
- [85] MetaMask, "How to Use MetaMask SDK in Unreal Engine," MetaMask Documentation, 2024. [Online]. Available: <https://docs.metamask.io/wallet/how-to/use-sdk/gaming/unreal-engine/>.
- [86] K. Iyer and C. Dannen, Building Games with Ethereum Smart Contracts: Intermediate Projects for Solidity Developers. Cham, Switzerland: Springer, 2018.
- [87] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Networks," in Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS), Montreal, Canada, Dec. 2014, pp. 2672-2680.
- [88] X. Wang, L. Xie, C. Dong, and Y. Shan, "Real-ESRGAN: Training Real-World Blind Super-Resolution with Pure Synthetic Data," arXiv preprint arXiv:2107.10833, 2021. [Online]. Available: <https://github.com/xinntao/ESRGAN>.
- [89] Y. N. Aldeoes, P. Gokhale, and S. Y. Sondkar, Eds., AI, IoT, Big Data and Cloud Computing for Industry 4.0. Cham, Switzerland: Springer, 2021.
- [90] L. Parthiban, T. P. Latchoumi, K. Balamurugan, K. Raja, and R. Parthiban, Eds., Integrating Blockchain and Artificial Intelligence for Industry 4.0 Innovations. Cham, Switzerland: Springer, 2021.
- [91] N. G. Mankiw, Principles of Economics, 8th ed. Boston, MA, USA: Cengage Learning, 2017. ISBN: 978-1-305-58512-6. J. Ho, A. Jain, and P. Abbeel, "Denoising Diffusion Probabilistic Models," arXiv preprint, arXiv:2006.11239, Jun. 2020. [Online]. Available: <https://arxiv.org/abs/2006.11239>.
- [92] "Fine-tune a pretrained model," Hugging Face, [Online]. Available: <https://huggingface.co/docs/transformers/en/training>.
- [93] "Unreal Engine 5 All Console Variables and Commands," Epic Developer Community Forums, [Online]. Available: <https://forums.unrealengine.com/t/unreal-engine-5-all-console-variables-and-commands/608054>.