ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ
ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

Έλεγχος συναλλαγών στο blockchain μέσω κατανεμημένης off-chain μηχανικής μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Πέτρος Σκούφης

Επιβλέπων : Θεοδώρα Βαρβαρίγου
Καθηγήτρια Ε.Μ.Π.

Αθήνα, Μάιος 2022

Αυτή η σελίδα σκοπίμως αφέθηκε κενή

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ
ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

Έλεγχος συναλλαγών στο blockchain μέσω κατανεμημένης off-chain μηχανικής μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Πέτρος Σκούφης

Επιβλέπων : Θεοδώρα Βαρβαρίγου
Καθηγήτρια Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 25η Μαΐου 2022.

.....	.....	.....
Θεοδώρα Βαρβαρίγου	Συμεών Παπαβασιλείου	Εμμανουήλ Βαρβαρίγος
Καθηγήτρια Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.

Αθήνα, Μάιος 2022

Αυτή η σελίδα σκοπίμως αφέθηκε κενή

.....
Πέτρος Σκούφης

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Αυτή η σελίδα σκοπίμως αφέθηκε κενή

Περίληψη

Η εποχή που ζούμε χαρακτηρίζεται σε μεγάλο βαθμό από την κατακόρυφη αύξηση της παραγωγής δεδομένων. Είναι αρκετές οι φορές που τα πιο πλούσια σε αξία δεδομένα είναι και αυτά που διαφέρουν από την κανονικότητα του συνόλου, είναι δηλαδή έκτοπα, και σηματοδοτούν ένα αξιόλογο γεγονός. Η έγκαιρη ταυτοποίηση των έκτοπων αυτών δειγμάτων από έξυπνα συστήματα τεχνητής νοημοσύνης μπορεί να φανεί ιδιαιτέρως χρήσιμη στην αντιμετώπιση σύγχρονων προβλημάτων, όπως το φαινόμενο της απάτης στις ηλεκτρονικές συναλλαγές και ο προσδιορισμός μεταλλάξεων μίας ασθένειας στο κομμάτι της υγείας. Παρ ’όλα αυτά και στις δύο περιπτώσεις τα δεδομένα είναι ευαίσθητα και γεωγραφικώς κατανεμημένα, άρα προσβάσιμα από μικρή μερίδα οργανισμών. Η τεχνολογία του blockchain μπορεί να υποστηρίξει κατανεμημένα δίκτυα αποφάσεων που διασφαλίζουν την προστασία των δεδομένων που διακινούνται στο εσωτερικό τους. Η παρούσα διπλωματική προτείνει μία ολοκληρωμένη αρχιτεκτονική συνεργασίας οργανισμών, βασισμένη στο Ethereum blockchain, με σκοπό την αντιμετώπιση του προβλήματος αναγνώρισης έκτοπων δεδομένων. Οι συμμετέχοντες οργανισμοί μπορούν μέσω oracles και με τη βοήθεια του IPFS να προσκομίσουν ογκώδη δεδομένα στο δίκτυο και να παρέχουν εκτιμήσεις πάνω σε αυτά με τα μοντέλα μηχανικής μάθησης που έχουν στη διάθεσή τους. Μέσω δοκιμών σε σενάρια διαπιστώνεται πως η αρχιτεκτονική αυτή μπορεί να παράξει συνολικά καλύτερες εκτιμήσεις ως προς τη μετρική “f1-macro-avg”, από την πλειονότητα των οργανισμών που την απαρτίζουν.

Λέξεις – κλειδιά : Blockchain, μηχανική μάθηση, τεχνητή νοημοσύνη, oracles, IPFS, αναγνώριση ανωμαλιών

Αυτή η σελίδα σκοπίμως αφέθηκε κενή

Abstract

The era that we live in is characterized in a great extent by the enormous increase in data production. There are many times when the richest in value data are those that differ from the dataset's normality, and as a result can be considered outliers, and they act as signals of significant events. The early identification of those outlier samples from smart artificial intelligence systems can be proven very helpful in combating contemporary problems, such as the issue of fraud in online transactions and the identification of mutation of a virus in the field of health. However, in both those situations data are private and geographically distributed and as a result accessible to a small group of organizations. Blockchain technology can support distributed decision networks that secure the protection of sensitive data that are exchanged within them. This diploma thesis suggests a complete architecture of cooperation between organizations, based on the Ethereum blockchain, in an effort to combat the issue of outlier samples detection. The participant organizations, with the aid of oracles and the IPFS, can bring heavy data on the network and provide predictions over them, using the machine learning models that they have at their disposal. Through testing in different realistic scenarios, it is found that the proposed architecture produces, as a whole, better predictions, as far as the "f1-macro-avg" metric is concerned, compared to the majority of the organizations that constitute the network individually.

Keywords : Blockchain, machine learning, artificial intelligence, oracles, IPFS, anomaly detection

Ευχαριστίες

Με την περάτωση της παρούσας διπλωματικής ολοκληρώνεται το ταξίδι μου ως προπτυχιακός φοιτητής στη Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου, ένα ταξίδι γεμάτο εμπειρίες, χαμόγελα και γνώσεις.

Θα ήθελα από καρδιάς να ευχαριστήσω την Καθηγήτρια κ. Βαρβαρίγου για την υποστήριξή της και την εμπιστοσύνη που μου έδειξε στο θέμα της διπλωματικής μου εργασίας, δίνοντάς μου την ελευθερία να μελετήσω ένα θέμα για το οποίο είχα τεράστιο ενδιαφέρον. Παράλληλα, θα ήθελα να ευχαριστήσω τον ερευνητή κύριο Αντώνιο Λίτκε, για το ενδιαφέρον και τη συνεχή και πολύτιμη υποστήριξή του στην ερευνητική μου αυτή προσπάθεια, όποτε και αν τον χρειάστηκα.

Θα ήθελα να πω ένα μεγάλο ευχαριστώ στην οικογένειά μου, που και σε αυτό, όπως και σε όλα μου τα όνειρα, δεν έπαψε στιγμή να με υποστηρίζει και πιστεύει σε μένα. Για όλους τους φίλους και συμφοιτητές, που στάθηκαν στυλοβάτες στο ταξίδι μου, που μοχθήσαμε και γελάσαμε μαζί και που έδωσαν χρώμα στα φοιτητικά μου χρόνια, υπάρχει μία ξεχωριστή θέση στην καρδιά μου.

Τέλος, ξεχωριστή μνεία θα ήθελα να κάνω στην θεία μου Ελένη και στον καθηγητή μου Διονύση, που με πάθος μου καλλιέργησαν από τη νεαρή μου ηλικία την αγάπη για την επιστήμη και με βοήθησαν να βρω το πεδίο που μου ταιριάζει και αγαπώ.

Περιεχόμενα

Ευρετήριο Εικόνων

Ευρετήριο Διαγραμμάτων

Ευρετήριο Πινάκων

Κεφάλαιο 1

Εισαγωγή

Το blockchain και η τεχνητή νοημοσύνη αποτελούν δύο ακμάζοντες κλάδους της τεχνολογίας, με τις εφαρμογές τους να εξελίσσουν καθημερινά την ανθρώπινη ζωή. Το blockchain αποτελεί ένα μέσο αποθήκευσης δεδομένων και εκτέλεσης συναλλαγών με στιβαρά συγκριτικά πλεονεκτήματα, όπως η ασφάλεια, η ιχνηλασιμότητα και η ακεραιότητα των δεδομένων στο εσωτερικό του δικτύου. Από την άλλη η μηχανική μάθηση ως κομμάτι του ευρύτερου πεδίου της τεχνητής νοημοσύνης παρουσιάζει τρομερές δυνατότητες ως προς την εκμετάλλευση των δεδομένων και την εξαγωγή έξυπνων αποφάσεων από αυτά.

Η αλληλεπίδραση των δύο τεχνολογιών σε κοινές εφαρμογές αποτελεί ήδη πραγματικότητα, η οποία δημιουργεί ένα πολύ δυνατό μίγμα συγκριτικών πλεονεκτημάτων, το οποίο έρχεται να αντιμετωπίσει σύνθετα και πολυπαραγοντικά προβλήματα του σύγχρονου κόσμου. Στον τομέα των έξυπνων πόλεων η τεχνολογία blockchain επιλύει τα θέματα ιδιωτικότητας που προκύπτουν από τη συλλογή δεδομένων από αισθητήρες IoT, δημιουργώντας ασφαλή δίκτυα διακίνησης δεδομένων μεταξύ των σημείων συλλογής, τα οποία τροφοδοτούν μοντέλα έξυπνων αποφάσεων που βασίζονται στη μηχανική μάθηση, στις πόλεις αυτές. Ακόμα εταιρίες κολοσσοί στον τομέα της πληροφορικής, όπως η IBM, έχουν αναπτύξει λύσεις που στον τομέα της οικονομίας προχωρούν σε ανάλυση δεδομένων χρηστών μέσω μοντέλων μηχανικής μάθησης, τα οποία υπολογίζουν το ρίσκο δανεισμού αυτών, χωρίς όμως να εκθέτουν ευαίσθητα για αυτούς δεδομένα, αφού αυτά διατηρούνται στο blockchain. Στην ίδια κατεύθυνση και η εταιρία αυτοκινήτων Porsche, χρησιμοποιεί το blockchain για να διακινεί ασφαλώς τα δεδομένα των χρηστών της και να τα χρησιμοποιεί για να τους παρέχει έξυπνες λειτουργίες στα αυτοκίνητά τους.

Αυτό οφείλεται σε μεγάλο βαθμό και σε υπηρεσίες γέφυρες ανάμεσα στις δύο αυτές τεχνολογίες, όπως τα oracles, τα οποία ενώνουν το στεγανό ανεξάρτητο κόσμο της αλυσίδας του blockchain, με εξωτερικά δεδομένα παραγόμενα από μοντέλα μηχανικής μάθησης, έπειτα από υπολογιστικά απαιτητικές προβλέψεις.

Στα πλαίσια της διπλωματικής αυτής θα προταθεί μία συνολική αρθρωτή αρχιτεκτονική που συνδυάζει μεταξύ άλλων τις τεχνολογίες του blockchain και της μηχανικής μάθησης για να αντιμετωπίσει το πρόβλημα της αναγνώρισης ανωμαλιών. Παράλληλα, θα πραγματοποιηθεί μία έρευνα ως προς τους τύπους των προβλημάτων αναγνώρισης ανωμαλιών στα οποία η προτεινόμενη αρχιτεκτονική ανταποκρίνεται με βέλτιστο τρόπο.

1.1. Αντικείμενο της διπλωματικής εργασίας

Ένα από τα κυριότερα χαρακτηριστικά της εποχής μας είναι η ύπαρξη μίας πληθώρας συσκευών οι οποίες παράγουν καθημερινά εκατομμύρια δεδομένα. Η αξιοποίηση αυτών των δεδομένων, με σκοπό τη βελτίωση της ανθρώπινης ζωής, αποτελεί ένα σύγχρονο επιστημονικό στοίχημα, το οποίο προσπαθούν να επιλύσουν νέες ακμάζουσες τεχνολογίες, όπως η τεχνητή νοημοσύνη και το blockchain.

Στην προσπάθεια αυτή της επιστήμης για αποτελεσματική διαχείριση και εξαγωγή συμπερασμάτων από τα δεδομένα, καταλυτικός παράγοντας επιτυχίας αποτελεί η ύπαρξη συνεργασίας μεταξύ φορέων και ιδρυμάτων που ασχολούνται με το αντικείμενο. Η δημιουργία, λοιπόν, ενός πλαισίου μέσα στο οποίο οι φορείς αυτοί μπορούν να συνεργαστούν και να μοιραστούν τα δεδομένα και τα συμπεράσματά τους πάνω σε αυτά, με τρόπο που διασφαλίζει την ιδιωτικότητα των προσωπικών δεδομένων των εμπλεκόμενων αλλά και την εμπιστοσύνη στο τελικό αποτέλεσμα, θα αποτελούσε μία τεράστια νίκη της επιστήμης.

Στα πλαίσια της παρούσας διπλωματικής εργασίας, προτείνεται ένα ολοκληρωμένο αρθρωτό πληροφοριακό σύστημα, το οποίο ακολουθεί την παραπάνω προσέγγιση. Η προτεινόμενη λύση χρησιμοποιεί το Ethereum blockchain, ως μέσο για την επικοινωνία μεταξύ κόμβων, τη γνωστοποίηση αιτημάτων και την αποθήκευση αναγκαίων δεδομένων. Παράλληλα, το blockchain αποσυμφορείται από το μεγάλο όγκο δεδομένων, ο οποίος αποθηκεύεται στο IPFS. Οι αποφάσεις των κόμβων, οι οποίες καταγράφονται στο blockchain, λαμβάνονται από μοντέλα μηχανικής μάθησης, τα οποία είναι φορτωμένα σε ένα εξωτερικό ως προς το δίκτυο blockchain τμήμα του προτεινόμενου πληροφοριακού συστήματος και επικοινωνούν με αυτό μέσω oracles. Το σύστημα υποστηρίζει επιπλέον διεπαφές command line και web frontend, οι οποίες κάνουν δυνατή την αξιοποίησή του από χρήστες διαφόρων κατηγοριών. Τέλος, όλα τα επιμέρους τμήματα του συστήματος επικοινωνούν μεταξύ τους μέσω αρχιτεκτονικών REST API, πράγμα που διασφαλίζει ότι οι αλλαγές σε ένα τμήμα του συστήματος δε θα επηρεάζουν το σύστημα ως ολότητα.

Η προτεινόμενη αυτή λύση ελέγχεται ως προς την αποτελεσματικότητά της στα πλαίσια της διπλωματικής, πάνω σε ένα πρόβλημα αναγνώρισης ανωμαλιών σε δεδομένα μεγάλου όγκου, όπως είναι οι εικόνες. Μέσα από τα διαφορετικά σενάρια που ελέγχονται, εξετάζεται η αποτελεσματικότητα της λύσης, σε καθεστώτα προβλημάτων όπου επιμέρους μοντέλα μηχανικής μάθησης έχουν εκπαιδευθεί στην αναγνώριση μίας κοινής ανωμαλίας πάνω στα δεδομένα εισόδου, αλλά και στο πιο ενδιαφέρον πρόβλημα όπου καθένα από τα μοντέλα έχει εκπαιδευθεί ώστε να αναγνωρίζει μία ξεχωριστή ανωμαλία. Παράλληλα, μεταβάλλεται το μέγεθος τους συνόλου δεδομένων εκπαίδευσης και η σύσταση των ανωμαλιών σε αυτά. Στόχος είναι να καταδειχθεί ότι οι κοινή απόφαση που λαμβάνεται από τους κόμβους που έχουν εκπαιδευθεί σε ανεξάρτητα μεταξύ τους σύνολα δεδομένων προσεγγίζει ή/και ξεπερνάει σε περιπτώσεις την απόδοση καθενός από τα επιμέρους κόμβους και να διερευνηθούν οι συγκεκριμένοι τύποι προβλημάτων, στους οποίους η αρχιτεκτονική συνολική απόφασης υπερτερεί έναντι της ατομικής απόφασης.

1.2. Οργάνωση της διπλωματικής εργασίας

Το κείμενο της διπλωματικής εργασίας οργανώνεται ως εξής:

Το Κεφάλαιο 1 περιλαμβάνει μία εισαγωγή στις τεχνολογίες blockchain και μηχανικής μάθησης και τη μεταξύ τους αλληλεπίδραση σε εφαρμογές. Επιπλέον, γίνεται αναφορά στο αντικείμενο της παρούσας διπλωματικής και στην έρευνα που θα διενεργηθεί σε αυτή.

Το Κεφάλαιο 2 περιλαμβάνει ολόκληρο το θεωρητικό υπόβαθρο, στο οποίο στηρίζονται οι προτάσεις της παρούσας διπλωματικής. Περιγράφονται τα κύρια συστατικά όλων των τεχνολογιών που πρόκειται να χρησιμοποιηθούν, όπως η τεχνητή νοημοσύνη, το blockchain, το IPFS, τα oracles και η αρχιτεκτονική REST. Τέλος, γίνεται παρουσίαση σύγχρονων εφαρμογών των παραπάνω τεχνολογιών και ιδίως εφαρμογών που συνδυάζουν παραπάνω από μία από αυτές, με σκοπό να αντιμετωπίσουν σύγχρονα προβλήματα.

Το Κεφάλαιο 3 αναφέρεται στην αρχιτεκτονική του πληροφοριακού συστήματος που προτείνεται στα πλαίσια της παρούσας διπλωματικής. Παρουσιάζονται λειτουργικές και μη λειτουργικές απαιτήσεις του συστήματος, διαγραμματική απεικόνιση του συνολικού συστήματος και των επιμέρους υποσυστημάτων, καθώς και η κύρια λειτουργικότητα καθενός από αυτά. Επιπλέον, γίνεται συνοπτική παρουσίαση των τεχνολογιών που θα χρησιμοποιηθούν και επεξήγηση της επιλογής τους, όπου αυτό κρίνεται αναγκαίο.

Το Κεφάλαιο 4 περιλαμβάνει την παρουσίαση της λειτουργίας της εφαρμογής ως σύνολο και των επιμέρους υποσυστημάτων της ατομικά. Για καθένα από τα υποσυστήματα παρουσιάζεται μία απλή ροή εργασιών, μέσω της οποίας διαφαίνεται η ομαλή λειτουργία του, όπως επίσης και η συμβολή του στο συνολικό σύστημα.

Το Κεφάλαιο 5 περιλαμβάνει ένα πειραματικό μέρος, όπου μέσω δύο σεναρίων ελέγχεται η αποτελεσματικότητα της προτεινόμενης αρχιτεκτονικής σε διάφορες παραλλαγές προβλημάτων αναγνώρισης ανωμαλιών.

Το Κεφάλαιο 6 περιλαμβάνει μία παρουσίαση των συμπερασμάτων και των περιθωρίων για περαιτέρω έρευνα που γεννώνται μετά την ανάλυση που πραγματοποιείται στην παρούσα διπλωματική.

Κεφάλαιο 2

Θεωρητικό υπόβαθρο και σχετικές εργασίες

2.1. Blockchain

2.1.1. Κατανεμημένη υπολογιστική

Η κατανεμημένη υπολογιστική αποτελεί ένα μεγάλο πεδίο της επιστήμης των υπολογιστών, το οποίο εξετάζει τα κατανεμημένα υπολογιστικά συστήματα. Ως κατανεμημένο σύστημα σύμφωνα με τον Tanenbaum [1] ορίζουμε ένα σύστημα τα συστατικά του οποίου βρίσκονται σε διαφορετικούς υπολογιστές, οι οποίοι είναι συνδεδεμένοι σε δίκτυο και επικοινωνούν και συντονίζονται περνώντας μηνύματα ο ένας στον άλλον από οποιοδήποτε σύστημα. Τα συστατικά ενός κατανεμημένου συστήματος αλληλεπιδρούν μεταξύ τους με σκοπό να πετύχουν έναν κοινό στόχο.

2.1.2. Ορισμός blockchain και κύρια χαρακτηριστικά

Το blockchain είναι ένας όρος, ο οποίος και αυτός έχει μία κατανεμημένη φύση. Σύμφωνα με τους Χριστίδη και Δεβετσικιώτη [2], το blockchain είναι μία κατανεμημένη δομή δεδομένων, η οποία αναπαράγεται και διαμοιράζεται μεταξύ των μελών ενός δικτύου.

Συγκεκριμένα το blockchain, είναι ένα δημόσιο ledger το οποίο μπορεί να λειτουργήσει στη μορφή ενός log, που αποθηκεύει τις συναλλαγές που γίνονται με χρονολογική σειρά, διασφαλίζοντάς τες με ένα μηχανισμό συμφωνίας (consensus mechanism) και εξασφαλίζοντας πως το log αυτό δε μπορεί να μεταβληθεί (immutability) [3]. Το ιδιαίτερα χαρακτηριστικά του blockchain, τα οποία το κάνουν να ξεχωρίζει ως δομή αποθήκευσης δεδομένων είναι τα παρακάτω:

- **Αδυναμία μεταβολής (immutability**): δεν είναι δυνατό για καμία οντότητα να χειραγωγήσει, να αλλάξει ή να πλαστογραφήσει τα δεδομένα που έχουν αποθηκευτεί στο δίκτυο.
- **Μη αναστρεψιμότητα (irreversibility)**: όταν μία συναλλαγή εκτελεστεί επιτυχώς δεν υπάρχει κανένας τρόπος να αντιστραφεί η εκτέλεσή της.
- **Αποκέντρωση (decentralization)**: ο έλεγχος του δικτύου και της λήψης αποφάσεων σε αυτό μεταφέρεται από μία κεντρική οντότητα σε ένα κατανεμημένο δίκτυο ομότιμων κόμβων.
- **Διατηρισιμότητα (persistence)**: κάθε συναλλαγή μεταδίδεται σε όλο το δίκτυο κάνοντας την αλλοίωσή της πρακτικά αδύνατη.

- Ανω**νυμία (anonymity)**: η πλοήγηση του χρήστη μέσα στο δίκτυο γίνεται μέσω μίας ή περισσότερων διευθύνσεων, οι οποίες δεν εκθέτουν δεδομένα που μπορούν να αποκαλύψουν την πραγματική του ταυτότητα.

2.1.3. Εισαγωγή στο Bitcoin

Η έννοια του blockchain εμφανίστηκε για πρώτη φορά από τον Satoshi Nakamoto [4] το 2008, όταν αυτός παρουσίασε το κρυπτονόμισμα Bitcoin. Το Bitcoin αποτελεί, σύμφωνα με τον ιδρυτή του, μία εξολοκλήρου peer-to-peer εκδοχή ηλεκτρονικού χρήματος, η οποία επιτρέπει στις ηλεκτρονικές πληρωμές να γίνονται άμεσα, χωρίς να απαιτείται η μεσολάβηση από ένα χρηματοπιστωτικό ίδρυμα. Το κύριο πρόβλημα όταν δεν υπάρχει κάποιος μεσάζων για να εγγυηθεί την εγκυρότητα της συναλλαγής, είναι πως στην περίπτωση του ηλεκτρονικού χρήματος δεν υπάρχουν εγγυήσεις πως ο αγοραστής δεν θα επαναχρησιμοποιήσει το νόμισμα το οποίο έδωσε ως αντίτιμο σε μία συναλλαγή, ένα πρόβλημα γνωστό και ως πρόβλημα διπλού-εξόδου (double spending). Στο άρθρο του αυτό ο Nakamoto προτείνει ένα peer-to-peer δίκτυο ως μία λύση για την επίλυση του προβλήματος διπλού-εξόδου.

Η έννοια του blockchain, όπως εισάγεται από τον Nakamoto ξεφεύγει σε χρηστικότητα από το πεδίο των οικονομικών συναλλαγών και βρίσκει εφαρμογή σε δεκάδες πεδία, όπου υπάρχει ανάγκη για ένα δίκτυο που να υποστηρίζει έμπιστες συναλλαγές κάθε είδους.

2.1.4. Η έννοια του consensus και τα βασικά πρωτόκολλα

Μία από τις κύριες έννοιες στις οποίες βασίζεται το blockchain είναι η έννοια του consensus [5]. Ως consensus ορίζουμε το γεγονός πως όλοι οι κατανεμημένοι κόμβοι που απαρτίζουν ένα δίκτυο blockchain πρέπει να κατέχουν το ίδιο κατανεμημένο ledger. Αν και στα παραδοσιακά υπολογιστικά συστήματα, τα οποία χρησιμοποιούν έναν ή περισσότερους κοινούς server για την άντληση των δεδομένων το πρόβλημα αυτό φαντάζει απλό, στα κατανεμημένα δίκτυα, στα οποία συγκαταλέγεται και το blockchain, ο κάθε κόμβος εκτελεί ταυτόχρονα χρέη host και server και συνεπώς πρέπει να επικοινωνεί με τους υπόλοιπους κόμβους, ώστε το αντίγραφό του να διατηρείται σε consensus με το δίκτυο. Η διαθεσιμότητα των κόμβων και η πιθανή κακόβουλη συμπεριφορά τους, αποτελούν τροχοπέδη στην ομαλή λειτουργία του δικτύου και το consensus του αποτελέσματος. Παράλληλα, το πρωτόκολλο consensus πρέπει να ταιριάζει και με τον τύπο του blockchain που χρησιμοποιείται.

Ένα βασικό ζήτημα που συναντάται στα κατανεμημένα συστήματα είναι η απουσία ύπαρξης ενός ιδανικού πρωτόκολλου consensus. Το ζήτημα αυτό διατυπώθηκε από τον Brewer [6], μέσω του θεωρήματός του που έμεινε γνωστό ως «CAP Theorem» και αναφέρει πως τα κατανεμημένα συστήματα μπορούν να προσφέρουν εγγυήσεις για μόνο δύο από τα τρία χαρακτηριστικά που αναφέρονται παρακάτω ταυτόχρονα:

- **Συνοχή (Consistency):** Κάθε απόπειρα ανάγνωσης διαβάζει, είτε την πιο πρόσφατη εγγραφή, είτε δέχεται σφάλμα.

- **Διαθεσιμότητα (Availability):** Κάθε αίτημα λαμβάνει μία απάντηση, όχι σφάλμα, χωρίς την εγγύηση ότι περιέχει την πιο πρόσφατη εγγραφή.
- **Ανοχή στην διαίρεση (Partition tolerance):** Το κατανεμημένο σύστημα εξακολουθεί να λειτουργεί παρά το γεγονός ότι ένας πεπερασμένος αριθμός μηνυμάτων χάνεται ή καθυστερεί ανάμεσα στους επιμέρους κόμβους του.

Απόρροια του παραπάνω θεωρήματος αποτελεί το γεγονός πως μόλις συμβαίνει η διαίρεση ενός δικτύου λόγω αποτυχίας (network partition failure), πρέπει να γίνει επιλογή ανάμεσα στις δύο παρακάτω εναλλακτικές για το δίκτυο:

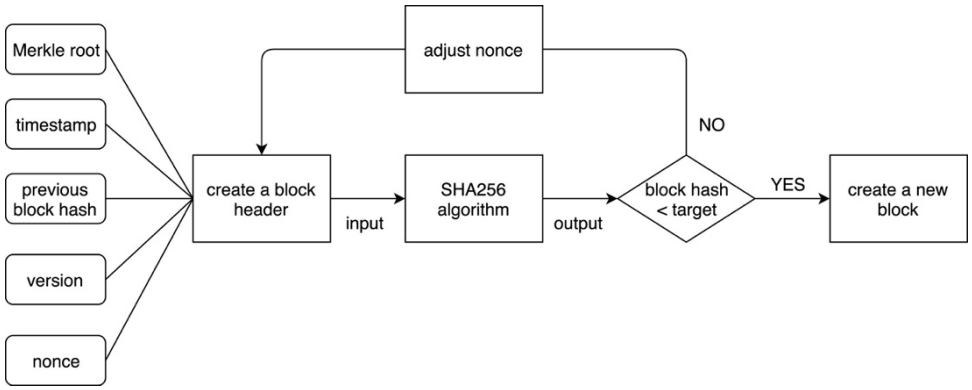
1. διακοπή της λειτουργίας του δικτύου που οδηγεί σε μείωση της διαθεσιμότητάς του αλλά διασφαλίζει την συνοχή των δεδομένων.
2. συνέχεια της λειτουργίας του δικτύου πετυχαίνοντας τη διαθεσιμότητά του υπό την απειλή όμως ασυνέπειας δεδομένων.

Μία επιπλέον απειλή που πρέπει να διαχειριστεί το πρωτόκολλο consensus, είναι αυτή του Προβλήματος των Βυζαντινών Στρατηγών [7]. Το πρόβλημα αυτό, το οποίο έχει διατυπωθεί από τους Lamport, Shostak και Pease, αναφέρεται στην ύπαρξη τμημάτων ενός κατανεμημένου υπολογιστικού συστήματος, στην περίπτωση του blockchain κόμβων, τα οποία, είτε λόγω δυσλειτουργίας, είτε λόγω κακόβουλων κινήτρων παρέχουν αντικρουόμενες πληροφορίες σε διαφορετικά τμήματα του δικτύου.

Σύμφωνα με τους Zhang και Lee [5] τα κυριότερα πρωτόκολλα consensus στο blockchain, τα οποία προσπαθούν να αντιμετωπίσουν με διάφορους μηχανισμούς τα προαναφερθέντα ζητήματα είναι τα παρακάτω:

- **Proof of Work (PoW):** Το πρωτόκολλο PoW εκτελεί έναν διαγωνισμό ισχύος μεταξύ των υποψήφιων κόμβων σε κάθε γύρο της διεργασίας consensus με σκοπό να επιλέξει τον κόμβο αυτόν που θα δημιουργήσει το νέο block. Για να συμμετάσχουν στο διαγωνισμό οι κόμβοι πρέπει να επιλύσουν ένα κρυπτογραφικό puzzle. Ο κόμβος όποιος καταφέρνει πρώτος να επιλύσει το puzzle έχει τη δυνατότητα να δημιουργήσει ένα νέο μπλοκ. Για την επίλυση του puzzle οι κόμβοι πρέπει συνεχώς να επεξεργάζονται την τιμή του nonce μέχρι να επιτύχουν το σωστό αποτέλεσμα διεργασία η οποία απαιτεί μεγάλο υπολογιστικό κόστος. Είναι δυνατόν για κακόβουλους κόμβους να εισάγουν ένα block στην αλυσίδα, αλλά όσο η αλυσίδα μεγαλώνει τόσο το φορτίο αυξάνεται, αυξάνοντας παράλληλα και τις υπολογιστικές απαιτήσεις για να επιτευχθεί κάτι τέτοιο. Το PoW αποτελεί ένα probabilistic-finality consensus πρωτόκολλο, πράγμα που σημαίνει πως εγγυάται eventual consistency. Το πρωτόκολλο αυτό έχει υιοθετηθεί από τα δημοφιλέστερα σύγχρονα

blockchains, όπως το Bitcoin και το Ethereum.



Διάγραμμα 1: Διαγραμματική απεικόνιση του πρωτοκόλλου Proof of Work [5]

- **Proof of Stake (PoS):** Στο πρωτόκολλο αυτό ο κόμβος που επιλέγεται για να δημιουργήσει ένα νέο block, εξαρτάται από το stake το οποίο κατέχει αντί της υπολογιστικής του δύναμης, Ως stake, ορίζεται ένα εσωτερικό νόμισμα, το οποίο χρησιμοποιείται ως κίνητρο στο δίκτυο με σκοπό να αντικαταστήσει την χρήση υπολογιστικής ισχύος και συνεπώς να παράξει ένα ενεργειακά φιλικό πρωτόκολλο.
- **Delegeted Proof of Stake (DPoS)**
- **Practical Byzantine Fault Tolerance (PBFT)**

2.1.5. Οι τύποι blockchain με βάση την πρόσβαση και τα δικαιώματα

Οι τρεις τύποι blockchain που υπάρχουν, ανάλογα με την πρόσβαση και τα δικαιώματα των χρηστών και των κόμβων σε αυτά είναι τα δημόσια blockchains (public blockchains), τα consortium blockchains και τα εξολοκλήρου ιδιωτικά blockchains (fully-private blockchains) [8]. Καθένας από αυτούς τους τύπους δικτύων έχει συγκεκριμένα χαρακτηριστικά και πλεονεκτήματα και ενδείκνυται για εφαρμογές με διαφορετικές απαιτήσεις. Πιο συγκεκριμένα:

- **Δημόσια blockchains:** τα δημόσια blockchains αποτελούν την εξολοκλήρου κατανεμημένη (fully-decentralized) εκδοχή του blockchain. Στα δημόσια blockchain μπορεί οποιοσδήποτε στον κόσμο να διαβάσει το περιεχόμενό τους, να στείλει σε αυτά συναλλαγές και να αναμένει αυτές οι συναλλαγές να συμπεριληφθούν στο blockchain, εφόσον κριθούν έγκυρες. Επιπλέον, ο οποιοσδήποτε έχει το δικαίωμα να συμμετάσχει στην διαδικασία consensus, η οποία αποφασίζει για το ποια blocks θα εισαχθούν στο blockchain. Ο τύπος αυτός αποτελεί μία ουσιαστική εναλλακτική για τα δίκτυα που βασίζονται σε έναν κεντρικό φορέα για την παροχή εμπιστοσύνης. Στα δημόσια blockchains η εμπιστοσύνη επιτυγχάνεται με ένα συνδυασμό από οικονομικά κίνητρα και κρυπτογραφική επιβεβαίωση χρησιμοποιώντας μία πληθώρα από consensus πρωτόκολλα. Ο συνδυασμός αυτός περιορίζει κατά κάποιο τρόπο την έκταση της επιρροής που μπορεί να έχουν μεμονωμένοι κόμβοι στην διαδικασία consensus, έτσι ώστε αυτή να είναι ανάλογη με τους πόρους, οικονομικούς και μη που προσφέρει αυτός στο δίκτυο. Με αυτό τον τρόπο αυξάνεται η προστασία του δικτύου έναντι κακόβουλων ή προβληματικών κόμβων.
- **Consortium blockchains:** Σε αυτό τον τύπο blockchains, η διαδικασία του consensus ελέγχεται από μία προκαθορισμένη ομάδα κόμβων με αυξημένα δικαιώματα. Στο

παράδειγμα των ηλεκτρονικών συναλλαγών, η ομάδα αυτή μπορεί να αποτελείται από 20 προκαθορισμένα τραπεζικά ιδρύματα, από τα οποία σύμφωνα με το πρωτόκολλο consensus ένας αριθμός, για παράδειγμα 15, πρέπει να υπογράψει κάθε block, ώστε αυτό να θεωρηθεί έγκυρο. Το σχήμα αυτό το οποίο είναι «μερικώς κατανεμημένο» (partially decentralized), μπορεί να υπάρχει με αρκετές διαφοροποιήσεις ως προς το βαθμό της δημοσιότητας που έχει το περιεχόμενό του, με άλλες παραλλαγές του να επιτρέπουν τη δημόσια ανάγνωση και άλλες να την περιορίζουν στους προκαθορισμένους κόμβους με αυξημένα δικαιώματα. Επιπλέον, αυτού του είδους τα blockchains, συνδυάζονται συχνά με αρχιτεκτονικές API, οι οποίες παρέχουν τη δυνατότητα στο κοινό να αποκτήσει πρόσβαση σε όλα τα δεδομένα ή σε μερικά κομμάτια αυτών, βάζοντας παράλληλα περιορισμούς ως προς τον αριθμό και τη συχνότητα των κλήσεων.

- **Εξολοκλήρου ιδιωτικά blockchains:** Στον τύπο αυτό τα δικαιώματα για εγγραφή στο blockchain είναι αυστηρώς περιορισμένα σε έναν οργανισμό, πράγμα που συγκρούεται εμμέσως με την αποκεντρωμένη φύση του blockchain, ως έννοιας. Τα δικαιώματα ανάγνωσης μπορεί να διαφέρουν, όπως και στα consortium blockchains. Ο τύπος αυτός ουσιαστικά λειτουργεί σαν υποκατάστατο μίας βάσης δεδομένων, η οποία απολαμβάνει τα πλεονεκτήματα των blockchains, ως προς τα χαρακτηριστικά της και ανάλογα με το εύρος των δικαιωμάτων ανάγνωσης μπορεί να χρησιμοποιηθεί ως τρόπος εσωτερικής και εξωτερικής λογοδοσίας και ελέγχου οργανισμών.

2.1.6. Το Ethereum blockchain

Το Ethereum αποτελεί ένα δημόσιο blockchain, το οποίο βασίζεται πάνω στο κρυπτονόμισμα ETH και προσφέρεται για τη δημιουργία κατανεμημένων εφαρμογών (Dapps), μέσω έξυπνων συμβολαίων που αναπτύσσονται σε κώδικα Ethereum Virtual Machine (EVM). Ο κώδικας αυτός είναι χαμηλού επιπέδου και αποτελείται από μία σειρά bytes, καθένα από τα οποία αντιστοιχεί σε μία πράξη.

Σύμφωνα με το whitepaper του συγκεκριμένου blockchain, όπως αυτό συντάθηκε από τον εφευρέτη του, V. Buterin [9], αυτό προτάθηκε έχοντας ως σκοπό να παρέχει ένα εναλλακτικό πρωτόκολλο για κατανεμημένες εφαρμογές μεγάλες κλίμακας. Το πρωτόκολλο αυτό υπόσχεται να εστιάζει στο να παρέχει ένα πλαίσιο όπου είναι εφικτή η ταχύτατη ανάπτυξη, χωρίς να θυσιάζεται η ασφάλεια και η αποδοτικότητα τον εφαρμογών που το χρησιμοποιούν. Το Ethereum αποτελεί ένα blockchain με μεγάλο επίπεδο αφαίρεσης, παρέχοντας μία Turing-πλήρη γλώσσα, η οποία επιτρέπει στους προγραμματιστές να δημιουργούν εφαρμογές με μεγάλη ευελιξία και ελευθερία.

Ένα βασικό συστατικό του Ethereum, είναι οι λογαριασμοί (accounts), οι οποίοι προσδιορίζονται από μία διεύθυνση 20-byte και περιέχουν τα εξής τέσσερα πεδία:

- **Nonce**: Είναι ένας μετρητής που χρησιμοποιείται με σκοπό να διασφαλίσει πως κάθε συναλλαγή μπορεί να εκτελεστεί μόνο μία φορά.
- **Ether balance:** Είναι το τρέχον υπόλοιπο του λογαριασμού στο κρυπτονόμισμα που χρησιμοποιεί το συγκεκριμένο blockchain.

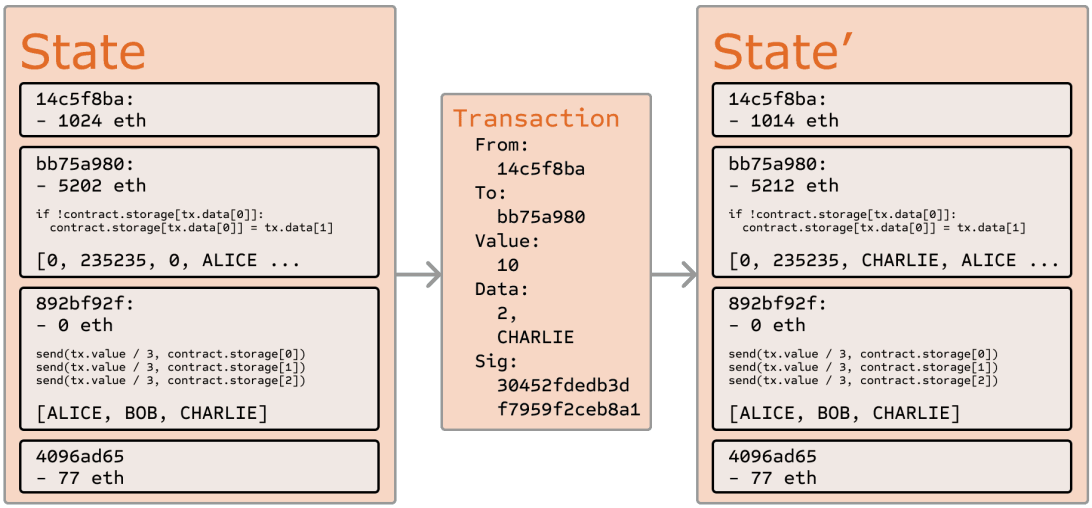
- **Κώδικας συμβολαίου (Contract code):** Το πεδίο αυτό περιέχει τον κώδικα του συμβολαίου, όταν ο λογαριασμός αντιστοιχεί σε κάποιο έξυπνο συμβόλαιο. Κάθε φορά που ο συγκεκριμένος λογαριασμός δέχεται ένα μήνυμα ο κώδικας αυτός εκτελείται.
- **Χώρος αποθήκευσης (Storage):** Το πεδίο αυτό είναι άδειο και χρησιμοποιείται σε λογαριασμούς που αντιστοιχούν σε συμβόλαια, αντιστοιχώντας στο χώρο που τα έξυπνα συμβόλαια γράφουν και διαβάζουν κατά την εκτέλεσή τους.

Όταν οι λογαριασμοί δεν αντιστοιχούν σε συμβόλαια, τότε είναι εξωτερικοί λογαριασμοί χρηστών, οι οποίοι ελέγχονται από προσωπικά ιδιωτικά κλειδιά (private keys). Στο Ethereum η «κατάσταση» (state) αποτελείται από τους λογαριασμούς αυτούς και τις μεταφορές αξίας και πληροφοριών μεταξύ τους.

Η ανταλλαγή αυτή συντελείται μέσω συναλλαγών και μηνυμάτων. Οι συναλλαγές αποτελούν υπογεγραμμένα πακέτα δεδομένων, τα οποία περιέχουν ένα μήνυμα που στέλνεται από έναν εξωτερικό λογαριασμό. Τα κύρια χαρακτηριστικά των συναλλαγών είναι:

- Ο παραλήπτης (recipient): ο λογαριασμός για τον οποίο προορίζεται το μήνυμα που αποστέλλεται.
- Η υπογραφή (signature) του αποστολέα: το προσδιοριστικό το οποίο πιστοποιεί την ταυτότητα του αποστολέα.
- Η ποσότητα ether που μεταφέρονται από τον αποστολέα στον παραλήπτη.
- Ένα προαιρετικό πεδίο δεδομένων (data field): το πεδίο αυτό χρησιμοποιείται κατά κόρον από τα έξυπνα συμβόλαια, ώστε να δεχθούν δεδομένα για να καταγράψουν στην αλυσίδα.
- Μία τιμή που αναφέρεται ως STARTGAS: η τιμή αυτή αναπαριστά το μέγιστο αριθμό υπολογιστικών βημάτων που επιτρέπεται να διαρκέσει η εκτέλεση της συναλλαγής.
- Μία τιμή που αναφέρεται ως GASPRICE: η τιμή αυτή αναπαριστά το αντίτιμο που καλείται ο αποστολέας να καταβάλει για κάθε υπολογιστικό βήμα που συντελείται.

Η διαδικασία της μετάβασης από μία κατάσταση στην επόμενη συνοψίζεται στην επόμενη εικόνα.



Διάγραμμα 2: Διάγραμμα μετάβασης καταστάσεων στο Ethereum [9]

Η συνάρτηση που εφαρμόζεται για να μεταβεί το δίκτυο του Ethereum από μία αρχική κατάσταση S σε μία νέα κατάσταση S' μέσω μιας συναλλαγής TX αναπαρίσταται ως:

$$APPLY(S,TX) -> S'$$

και περιγράφεται από τα παρακάτω βήματα:

1. Έλεγχος της συναλλαγής ως προς την εγκυρότητα της μορφής της, όπως αυτή αντικατοπτρίζεται από τον αριθμό των τιμών που περιλαμβάνει. Επιπλέον, συντελείται έλεγχος της εγκυρότητας της υπογραφής του αποστολέα, καθώς επίσης επιβεβαιώνεται πως ο μετρητής nonce της συναλλαγής ταυτίζεται με αυτόν του λογαριασμού του αποστολέα. Σε αντίθετη περίπτωση η συναλλαγή αποτυγχάνει και τερματίζει επιστρέφοντας σφάλμα.
2. Υπολογισμός του αντιτίμου της συναλλαγής σύμφωνα με τη φόρμουλα:

$$STARTGAS * GASPRICE$$

και προσδιορισμός της διεύθυνσης του αποστολέα, όπως αυτή προκύπτει από την υπογραφή στη συναλλαγή. Το αντίτιμο που υπολογίστηκε αφαιρείται από το λογαριασμό του αποστολέα, ενώ παράλληλα αυξάνεται ο μετρητής nonce του λογαριασμού του, ώστε να συμπεριληφθεί σε αυτόν και η νέα συναλλαγή που πρόκειται να εκτελεστεί. Στην περίπτωση που ο αποστολέας δεν έχει στο λογαριασμό του επαρκείς πόρους για να καλύψει το αντίτιμο της συναλλαγής, η συναλλαγή αποτυγχάνει και τερματίζει επιστρέφοντας σφάλμα.

3. Αρχικοποίηση της τιμής GAS ως εξής:

$$GAS = STARTGAS$$

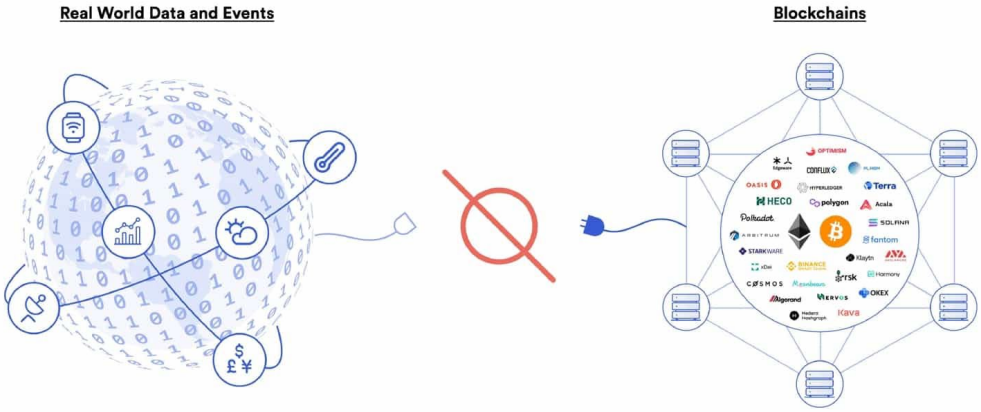
και αφαίρεση μιας συγκεκριμένης τιμής αντιτίμου ανά byte για να καταβληθεί για τα bytes της συναλλαγής.

4. Μεταφορά της αξίας της συναλλαγής από το λογαριασμό του αποστολέα στο λογαριασμό του παραλήπτη. Σε περίπτωση που ο λογαριασμός του παραλήπτη δεν υπάρχει κατά τη στιγμή της μεταφοράς, τότε γίνεται εκείνη τη στιγμή η δημιουργία του. Επιπλέον, στην περίπτωση που ο λογαριασμός του παραλήπτη αντιστοιχεί σε λογαριασμό έξυπνου συμβολαίου, τότε εκκινείται η εκτέλεση του κώδικα του έξυπνου συμβολαίου, η οποία τερματίζει είτε κατά την ολοκλήρωσή της, είτε σε περίπτωση που η εκτέλεση ξεμείνει από διαθέσιμους πόρους, gas.
5. Στην περίπτωση που η μεταφορά αξίας απέτυχε, είτε καθώς ο αποστολέας δεν κατείχε επαρκείς πόρους, είτε γιατί η εκτέλεση του κώδικα ξέμεινε από πόρους, τότε όλες οι αλλαγές κατάστασης αναστρέφονται, πλην της πληρωμής των αντιτίμων (fees) στους miners, τα οποία προστίθενται κανονικά στους λογαριασμούς τους,.
6. Σε αντίθετη περίπτωση, όσοι πόροι (gas) απέμειναν επιστρέφονται στον αποστολέα και το αντίτιμο που καταβλήθηκε για τους πόρους που καταναλώθηκαν μεταφέρεται στους miners.

2.1.7. Η έννοια των blockchain oracles

Ένα από τα κυριότερα εμπόδια τα οποία δυσχεραίνουν την ευρεία υιοθέτηση του blockchain σε μία πληθώρα πεδίων, παρά τα αδιαμφισβήτητα πλεονεκτήματά του είναι η εξ ορισμού απουσία επικοινωνίας με εξωτερικά συστήματα και συνεπώς εξωτερικές πηγές δεδομένων. Τα έξυπνα συμβόλαια που εκτελούνται στο blockchain δεν μπορούν να αλληλεπιδρούν με πηγές δεδομένων εξωτερικές του δικτύου, οι οποίες για το λόγο αυτό είναι γνωστές και ως «εκτός-αλυσίδας» (off-chain), έχοντας πρόσβαση μόνο στα δεδομένα που βρίσκονται ήδη στο blockchain, είναι δηλαδή «εντός-αλυσίδας». Η απομόνωση του blockchain από τον εξωτερικό κόσμο, λειτουργεί ως εγγύηση για τα χαρακτηριστικά του blockchain που παρουσιάστηκαν στην προηγούμενη ενότητα.

Η ανάγκη για εισαγωγή επικαιροποιημένων δεδομένων και σύνθετων υπολογισμών μέσα στα έξυπνα συμβόλαια καλύπτεται με τη χρήση των oracles. Τα oracles αποτελούν εξωτερικούς πράκτορες δεδομένων, οι οποίοι παρακολουθούν τα γεγονότα στον πραγματικό κόσμο και τα αναφέρουν πίσω στο blockchain, με σκοπό αυτά να χρησιμοποιηθούν από τα έξυπνα συμβόλαια [10]. Τα oracles αποτελούν, λοιπόν, έμπιστες οντότητες, οι οποίες γεφυρώνουν το blockchain με τον εξωτερικό κόσμο. Το γεγονός πως τα δεδομένα που τα oracles φέρνουν στο blockchain μπορούν να καθορίσουν άμεσα την έκβαση ενός έξυπνου συμβολαίου κάνει ξεκάθαρη την ανάγκη που υπάρχει να λειτουργούν με έναν σωστό μηχανισμό, παρέχοντας έγκυρες και συνεπείς πληροφορίες, ώστε να βεβαιώσουν τη συνέπεια και την εγκυρότητα της εκτέλεσης των έξυπνων συμβολαίων [11].



Εικόνα 1: Έλλειψη επικοινωνίας μεταξύ blockchain και πραγματικού κόσμου [12]

Ανάλογα με την πηγή, την κατεύθυνση και την εμπιστοσύνη τα oracles μπορούν να διαχωριστούν σε επιμέρους κατηγορίες [13]. Οι κατηγορίες αυτές δεν είναι αμοιβαίως αποκλειόμενες, με μία συγκεκριμένη υπηρεσία oracle να μπορεί να ανήκει ταυτόχρονα σε παραπάνω από μία τέτοιες κατηγορίες. Η διάκριση συντελείται με σκοπό να κατατάξει τις υπηρεσίες αυτές με βάση τα επιμέρους χαρακτηριστικά τους, τα οποία τις καθιστούν πιο συμβατές με συγκεκριμένους τύπους κατανεμημένων εφαρμογών. Οι κύριοι τύποι oracles παρουσιάζονται στον παρακάτω πίνακα [Πίνακας 1].

Ονομασία	Κριτήριο Διάκρισης	Περιγραφή
Oracles Λογισμικού (Software Oracles)	Πηγή πληροφοριών το διαδίκτυο.	Ο πιο συνήθης τύπος oracles. Αναμεταδίδει

		πληροφορίες από διάφορες πηγές στο διαδίκτυο στο έξυπνο συμβόλαιο και αντιστρόφως.
Oracles Υλικού (Hardware Oracles)	Πηγή πληροφοριών ο πραγματικός κόσμος.	Ο τύπος αυτός μεταφράζει γεγονότα του πραγματικού κόσμου, όπως αυτά συλλέγονται από συσκευές IoT (π.χ. αισθητήρες), σε δεδομένα τα οποία γίνονται διαθέσιμα για τα έξυπνα συμβόλαια. Βρίσκει κυρίως εφαρμογή στην εφοδιαστική αλυσίδα.
Ανθρώπινα Oracles (Human Oracles)	Οι πληροφορίες καταχωρούνται από ανθρώπινο παράγοντα.	Σε αυτό τον τύπο oracles, η καταχώρηση των πληροφοριών συντελείται από ανθρώπους. Ο μη αυτοματοποιημένος τρόπος προσκόμισης δεδομένων στην αλυσίδα του έξυπνου συμβολαίου επιτρέπει ευελιξία στα δεδομένα και χειρισμό πιο περίπλοκων αιτημάτων που πιθανώς δε μπορεί να επεξεργαστεί μία μηχανή.
Υπολογιστικά Oracles (Computation Oracles)	Οι πληροφορίες έχουν μεγάλο υπολογιστικό κόστος.	Ο τύπος αυτός των oracles χρησιμοποιείται για να αποφορτίσει το blockchain από υπολογιστικά βαριές διεργασίες, οι οποίες είναι είτε αδύνατες είτε ακριβές και ασύμφορες όταν συντελούνται εντός αλυσίδας.
Εισερχόμενα/Εξερχόμενα Oracles (inbound/Outbound	Κριτήριο η κατεύθυνση των πληροφοριών.	Τα oracles που μεταδίδουν πληροφορίες

Oracles)		του εξωτερικού ως προς το blockchain κόσμου προς την αλυσίδα ονομάζονται εισερχόμενα oracles. Αντιστοίχως, αυτά που κοινοποιούν πληροφορίες των έξυπνων συμβολαίων προς τον έξω κόσμο ονομάζονται εξερχόμενα oracles.
Oracles Συγκεκριμένου Συμβολαίου (Contract-specific Oracles)	Το oracle σχεδιάζεται για ένα συγκεκριμένο έξυπνο συμβόλαιο.	Ο εξειδικευμένος αυτός τύπος oracles, κατασκευάζεται ώστε να εξυπηρετεί πολύ συγκεκριμένες περιπτώσεις χρήσεις, επιτρέπει μεγάλη ευελιξία και εστίαση ως προς το περιεχόμενο των διεργασιών που επιτελεί, αλλά αντισταθμίζεται από μεγάλο κόστος ανάπτυξης και συντήρησής του, τόσο σε χρόνο όσο και σε πόρους, αφού λειτουργεί με ένα μόνο έξυπνο συμβόλαιο κάθε φορά.
Oracles Βασισμένα σε Consensus (Consensus-based Oracles)	Το πλήθος των πηγών των πληροφοριών.	Ο τύπος αυτός των oracles χρησιμοποιεί πολλαπλές πηγές για να προσκομίσει πληροφορίες στην αλυσίδα, προσπαθώντας να αυξήσει την ασφάλεια των πληροφοριών που παρέχονται και να εξαλείψει κενά ασφαλείας και διαθεσιμότητας που πηγάζουν από τη χρήση μοναδικών πηγών

		πληροφοριών. Η πληροφορία καθορίζεται από ένα σχήμα πλειοψηφίας, το οποίο αν και καθυστερεί καθώς απαιτείται η επίτευξη πλειοψηφίας, αυξάνει την εμπιστοσύνη στην τελική πληροφορία που καταγράφεται στην αλυσίδα.

Πίνακας 1: Ταξινόμηση τύπων oracles [10]

Ο Caldarelli [14] πραγματοποιεί στο άρθρο του μία εκτενή βιβλιογραφική ανασκόπηση των oracles, έχοντας ως σκοπό να εστιάσει στο πως οι διαφορετικοί τομείς στους οποίους βρίσκουν εφαρμογές τα oracles αντιμετωπίζουν τα προβλήματα που απορρέουν από τη χρήση τους. Το βασικό πρόβλημα της χρήσης oracles, το οποίο έχει καθιερωθεί να αναφέρεται και ως "The Oracle Problem", συνοψίζεται στο γεγονός πως η αυτόματη και «εύπιστη» λογική εκτέλεσης των έξυπνων συμβολαίων αντίκειται με τις επιφυλάξεις που υπάρχουν για την ασφάλεια, την αυθεντικότητα και την εμπιστοσύνη στις πληροφορίες που παρέχονται από εξωτερικές ως προς το blockchain πηγές, όπως τα oracles [15]. Το ζήτημα αυτό απαιτεί ιδιαίτερο χειρισμό,, καθώς μπορεί να επιφέρει κατάρρευση των εγγυήσεων που προσδίδουν στο blockchain τα συγκριτικά του πλεονεκτήματα.

Προτού αναφερθεί στα προβλήματα που δημιουργούνται από τα oracles, η έρευνά του Caldarelli [14] λειτουργεί ως ένας χρήσιμος χάρτης που απεικονίζει κλάδους στους οποίους υπάρχει τάση υιοθέτησης των oracles. Οι κυριότεροι κλάδοι, λοιπόν, που βρίσκουν εφαρμογές τα oracles είναι αυτοί των Αποκεντρωμένων Αυτόνομων Οργανισμών (Decentralized Autonomous Organizations – DAOs), της οικονομίας (finance), της διαχείρισης ενέργειας (energy), της νομικής και της προστασίας δικαιωμάτων, αλλά και της διαχείρισης εφοδιαστικής αλυσίδας και της ιχνηλασιμότητας σε αυτή.

Ένα χαρακτηριστικό παράδειγμα της χρήσης oracles στο πεδίο της οικονομίας και συγκεκριμένα της Αποκεντρωμένης Οικονομίας (Decentralized Finance – DeFi) είναι το δίκτυο MakerDAO και συγκεκριμένα ο τρόπος διαχείρισης του stablecoin "Dai" που στηρίζει το δίκτυο. Η ίδια η φύση των stablecoins εστιάζει στην ελαχιστοποίηση της αστάθειας της τιμής, μέσω της χρήσης κατάλληλων μηχανισμών. Το MakerDAO, το οποίο έχει χτιστεί πάνω στο Ethereum blockchain, χρησιμοποιεί ένα μηχανισμό που προχωρά σε αυτόματη διαχείριση δανείων μεταξύ του "Dai" και του πιο ισχυρού "ETH" με σκοπό να διατηρήσει μία όσο το δυνατόν πιο σταθερή ισοτιμία μεταξύ του "Dai" και του δολαρίου. Επιπλέον, σε καταστάσεις που το "ETH" πραγματοποιεί «επικίνδυνες» πτώσεις στην τιμή του, με αποτέλεσμα τα δάνεια "ETH" να παύουν να μπορούν να εγγυηθούν για το σύνολο του "Dai" και τη σταθερότητα του δικτύου, υπάρχει ένας μηχανισμός που προχωρά σε άμεση ρευστοποίηση αυτών των δανείων. Για να συντελεστεί επιτυχώς αυτό, η πλατφόρμα χρησιμοποιεί oracles, τα οποία πρέπει σε πραγματικό χρόνο να παρακολουθούν τις εξελίξεις στις ισοτιμίες των νομισμάτων και κρυπτονομισμάτων που την ενδιαφέρουν στα

ανταλλακτήρια και να ενημερώνει άμεσα τα έξυπνα συμβόλαια που έχουν αναλάβει τη διαχείριση αυτού του μηχανισμού ασφαλείας για τις τιμές τους, ώστε να λαμβάνουν άμεσα αποφάσεις. Μία καθυστέρηση λίγων δευτερολέπτων στην ενημέρωση σε ένα τέτοιο γεγονός το Μάρτιο του 2020, οδήγησε την πλατφόρμα σε ζημία της τάξης των 4 εκατομμυρίων δολαρίων.

Στον τομέα της ενέργειας η χρήση oracles, προτείνεται ως ένας τρόπος ενσωμάτωσης απαιτητικών υπολογιστικών υπηρεσιών στην διαχείριση smart grids [16]. Στην πολυεπίπεδη αρχιτεκτονική που παρουσιάζεται στο άρθρο υπάρχει ένα ξεχωριστό επίπεδο που διασφαλίζει την Υπολογιστική Κλιμάκωση (Computational Scaling) του συστήματος. Μέσω της χρήσης μίας υπηρεσίας oracle, τα δεδομένα που έχουν ληφθεί και ενσωματωθεί στην αλυσίδα μέσω των υπόλοιπων επιπέδων τροφοδοτούνται σε μοντέλα πρόβλεψης, τα οποία εκτελώντας απαιτητικές υπολογιστικά μεθόδους πρόβλεψης εκτός αλυσίδας, παρέχουν στο σύστημα προβλέψεις ενεργειακής κατανάλωσης και παραγωγής για πολλαπλούς χρονικούς ορίζοντες. Με βάση αυτά τα δεδομένα καθορίζονται οι τιμές προσφοράς και ζήτησης στην peer to peer (P2P) ενεργειακή αγορά.

2.1.8. Το InterPlanetary File System (IPFS) ως μέσο αποθήκευσης δεδομένων

Ένας από τους κυριότερους στενωπούς στην ανάπτυξη κατανεμημένων εφαρμογών που χρησιμοποιούν blockchains ως αποθηκευτικό χώρο για τα δεδομένα τους είναι ο όγκος των δεδομένων. Όταν τα blockchains επωμίζονται με μεγάλο όγκο δεδομένων προς αποθήκευση, παρατηρείται αισθητή μείωση της αποδοτικότητάς τους, τόσο στα πλαίσια μετρικών όπως η απόκριση και η ταχύτητά τους, όσο και σε μετρικές όπως η ενεργειακή κατανάλωση του δικτύου. Μία από τις πιο αποτελεσματικές λύσεις αποθήκευσης μεγάλου όγκου δεδομένων, η οποία συνεργάζεται πολύ καλά με τα blockchains είναι το InterPlanetary File System (IPFS) [17].

Το IPFS αποτελεί ένα πρωτόκολλο κατανεμημένης αποθήκευσης αρχείων. Το πρωτόκολλο αυτό βασίζεται στη διευθυνσιοδότηση των αρχείων προς αποθήκευση με ένα μοναδικό hash, το οποίο παράλληλα μπορεί να χρησιμοποιηθεί για την ανάκτησή τους. Το πρωτόκολλο χρησιμοποιεί έναν έξυπνο μηχανισμό διπλότυπων (duplication mechanism), ο οποίος επιτυγχάνει την εξυπηρέτηση των αιτημάτων στο δίκτυο, χωρίς την ανάγκη ύπαρξης ενός κεντρικού εξυπηρετητή (central server). Για δεδομένα, τα αιτήματα των οποίων εμφανίζονται με μεγάλη συχνότητα, το πρωτόκολλο δημιουργεί διπλότυπα σε όλη την έκταση της αλυσίδας του αιτήματος, έτσι ώστε το επόμενο αίτημα από την αλυσίδα να μπορεί να εξυπηρετηθεί τοπικά.

Το IPFS, λόγω των χαρακτηριστικών του αποτελεί το ιδανικό μέσο για ασφαλή, high throughput αποθήκευση μεγάλου όγκου δεδομένων εξασφαλίζοντας ακόμα μεγάλη διαθεσιμότητα για παράλληλη πρόσβαση. Αποθηκεύοντας το μεγάλο όγκο των δεδομένων τους στο IPFS και μόνο το τελικό IPFS hash στην αλυσίδα του blockchain, δίνεται η δυνατότητα στις εφαρμογές που έχουν ως βάση το blockchain να επιφορτίζουν την αλυσίδα μόνο με το άκρως απαραίτητο βάρος χωρίς να κάνουν «εκπτώσεις» στην λεπτομέρεια των

δεδομένων που αποθηκεύουν, ή να κινδυνεύουν να χάσουν αυτά κάποια από τις εγγυήσεις που τους προσφέρει το blockchain ως μέσο αποθήκευσης (πχ. immutability) [18].

2.2. Τεχνητή νοημοσύνη και μηχανική μάθηση

2.2.1. Ορισμός τεχνητής νοημοσύνης και επιμέρους τομείς της

Ένας επιπλέον τομέας, ο οποίος στην εποχή μας επιτυγχάνει αλματώδη εξέλιξη είναι αυτός της λήψης αποφάσεων, κυρίως λόγω της υιοθέτησης της τεχνητής νοημοσύνης μέσα στις διαδικασίες αυτές.

Σύμφωνα με τους Collins et. al [19] η ιστορία της τεχνητής νοημοσύνης στην πρώιμή της μορφή, έχει τις ρίζες της στην αρχαία Ελλάδα και βασίζεται στην επιστήμη και στην τεχνολογία. Η σύγχρονη εκδοχή της τεχνητής νοημοσύνης εισάγεται από τον Alan Turing, το 1950, ο οποίος προτείνει την κατασκευή μίας αυτόματης μηχανής, η οποία απαντά στο ερώτημα «Μπορούν οι μηχανές να σκεφτούν;» [19], ενώ το ίδιο ερώτημα πραγματεύεται και η σύγχρονή του McCorduck στο βιβλίο της με τίτλο «Μηχανές που σκέφτονται.» [20]. Ένας πρώτος ορισμός για την τεχνητή νοημοσύνη έρχεται από τον John McCarthy, ο οποίος την χαρακτηρίζει ως «την επιστήμη και μηχανική της κατασκευής ευφυών μηχανών» [21] .

Στη σύγχρονη εκδοχή της τεχνητής νοημοσύνης, η οποία βασίζεται στις πρωταρχικές αυτές έννοιες έχει επιτευχθεί μία μικρή άρση της γενίκευσης των πρώτων ορισμών. Ως τεχνητή νοημοσύνη πλέον, ορίζουμε την ικανότητα μιας μηχανής να εκτελεί εργασίες που τυπικά είναι συνδεδεμένες με ευφυή όντα. Μέσα σε αυτή τη γενική έννοια υπάρχουν επιμέρους πεδία τα οποία πραγματεύονται το καθένα από ένα πιο συγκεκριμένο φάσμα προβλημάτων που απαιτούν ευφυία. Αναλυτικότερα [19]:

- **Expert Systems:** Το πεδίο αυτό της τεχνητής νοημοσύνης ασχολείται με τη δημιουργία συστημάτων τα οποία προσπαθούν να μιμηθούν τη συμπεριφορά και της ικανότητες των ανθρώπων στην επίλυση προβλημάτων.
- **Μηχανική Μάθηση:** Το πεδίο αυτό εστιάζει στη χρήση δεδομένων και προηγμένων αλγορίθμων με σκοπό να μιμηθεί την διαδικασία της ανθρώπινης μάθησης και να βελτιώνει σταδιακά την ακρίβειά (accuracy) του [22].
- **Ρομποτική**
- **Επεξεργασία Φυσικής Γλώσσα (Natural Language Processing)**
- **Όραση Υπολογιστών (Computer Vision)**
- **Αναγνώριση Φωνής (Speech Recognition)**

Παρά το γεγονός πως η τεχνητή νοημοσύνη αποτελεί μία έννοια που έχει διατυπωθεί εδώ και δεκαετίες, η ωριμότητά της στην εποχή μας καθιστά την υιοθέτησή της ιδιαιτέρως εφικτή και αποτελεσματική σε μία πληθώρα εφαρμογών και πεδίων, τόσο σε ερευνητικό όσο και σε πρακτικό επίπεδο.

2.2.2. Μηχανική μάθηση και αναγνώριση ανωμαλιών

Μία συγκεκριμένη κατηγορία προβλημάτων στην οποία είναι αρκετά συχνή η χρήση μηχανικής μάθησης είναι αυτή της αναγνώρισης ανωμαλιών (anomaly detection) στα δεδομένα, γνωστή και ως αναγνώριση έκτοπων δεδομένων (outlier detection). Από στατιστικής άποψης ένα έκτοπο δείγμα σε ένα σύνολο δεδομένων είναι ένα δείγμα το οποίο φαίνεται να μη συμφωνεί με τα υπόλοιπα δεδομένα στο σύνολο αυτό [23].

Η απόκλιση αυτή από την κανονικότητα του συνόλου δεδομένων για ένα δείγμα, μπορεί να σηματοδοτεί είτε την ιδιαίτερη σημασία του, είτε την συσχέτισή του με κάποιο είδος εκούσιου ή ακούσιου σφάλματος. Σε κάθε περίπτωση είναι αδιαμφισβήτητη η ανάγκη για έγκαιρη αναγνώριση και εξέταση αυτών των δειγμάτων στα σύνολα δεδομένων. Αυτό μπορεί να γίνει ιδιαιτέρως αντιληπτό στο παράδειγμα των οικονομικών συναλλαγών, όπου μία τέτοια έκτοπη συναλλαγή μπορεί να συσχετίζεται με την απόπειρα απάτης και συνεπώς είναι αναγκαία η αναγνώριση και απόρριψή της από τα συστήματα ελέγχου των τραπεζικών ιδρυμάτων.

Η αυξημένη ανάγκη για την επιτυχημένη αναγνώριση των ανωμαλιών στα δεδομένα έχει οδηγήσει σε μεγάλη επιστημονική έρευνα στο πεδίο και κατά προέκταση πολλές διαφορετικές προσεγγίσεις. Οι μέθοδοι αυτές μπορεί να χωριστούν στις με βάση το εργαλείο που χρησιμοποιούν για να εκτελέσουν τον διαχωρισμό σε κανονικά δεδομένα και ανωμαλίες, στις κατηγορίες που παρουσιάζονται στις παρακάτω υποενότητες [24].

2.2.2.1. Πιθανοτικές μέθοδοι

Οι αλγόριθμοι που βασίζονται στις πιθανότητες για την εύρεση ανωμαλιών σε σύνολα δεδομένων ακολουθούν την εξής λογική. Αρχικά υπολογίζουν τη συνάρτηση πυκνότητας πιθανότητας του συνόλου δεδομένων X, συμπεραίνοντας τις παραμέτρους θ του μοντέλου. Έπειτα αναγνωρίζουν ως ανωμαλίες τα δείγματα εκείνα, στα οποία η πιθανότητα $P(X|\theta)$ ελαχιστοποιείται. Ο τύπος αυτός μοντέλων έχει το χαρακτηριστικό να παρουσιάζει ταυτόχρονα «εξέλιξη» και «μνήμη», υπό την έννοια πως τα νέα δεδομένα οδηγούν το μοντέλο στο να προσαρμοστεί, χωρίς όμως να «ξεχνά» παλαιότερα δεδομένα.

Χαρακτηριστικό παράδειγμα μεθόδου που χρησιμοποιεί τις πιθανότητες ως βάση είναι τα Μοντέλα Γκαουσιανών Μειγμάτων (Gaussian Mixture Models – GMM). Η μέθοδος αυτή εφαρμόζει μία ή περισσότερες γκαουσιανές κατανομές πάνω στο σύνολο δεδομένων. Κατά την εκπαίδευσή του το μοντέλο χρησιμοποιεί τον αλγόριθμο Expectation-Maximization (EM), ο οποίος επαναληπτικά εκτιμά τις παραμέτρους των κατανομών του μοντέλου, έχοντας ως γνώμονα την μεγιστοποίηση της a-posteriori πιθανότητας [25]. Αν και ο τύπος αυτός μοντέλων είναι ιδιαίτερα διαδεδομένος και αποτελεσματικός σε αρκετά προβλήματα, εμπόδιο στην αποτελεσματική χρήση του αποτελεί η διαδικασία της εύρεσης του πλήθους των κατανομών που πρέπει να χρησιμοποιηθούν και ο προσδιορισμός των παραμέτρων τους.

2.2.2.2. Μέθοδοι βασισμένες σε απόσταση

Οι μέθοδοι αυτές βασίζονται στον υπολογισμό της απόστασης των χαρακτηριστικών των δεδομένων στο σύνολο δεδομένων, είτε από γείτονες, είτε από κάποια γενικά πρότυπα που αναπαριστούν ένα κανονικό δείγμα του συνόλου δεδομένων με σκοπό να προσδιορίσουν τα έκτοπα δείγματα. Παραδείγματα τέτοιων αποστάσεων είναι η απόσταση Hamming, η Ευκλείδεια απόσταση, η απόσταση Manhattan και η απόσταση Minkowski.

2.2.2.3. Μέθοδοι βασισμένες σε γειτονιές

Μία από τις βασικότερες μεθόδους προσδιορισμού έκτοπων δειγμάτων, η οποία βασίζεται στην εξέταση γειτονιών δεδομένων, προσδιορίζοντας ως έκτοπα δείγματα τα δεδομένα που απέχουν πολύ από τη γειτονιά τους, είναι ο Τοπικός Παράγοντας Έκτοπου (Local Outlier Factor – LOF) [26]. Η μέθοδος αυτή χρησιμοποιεί την Ευκλείδεια απόσταση με σκοπό για καθένα από τα δεδομένα να προσδιορίσει μία μετρική που ορίζει ως *outlying degree,* δηλαδή το βαθμό κατά τον οποίο το συγκεκριμένο δείγμα είναι έκτοπο. Η μέθοδος αυτή λαμβάνει υπόψιν τόσο την απόσταση του σημείου που ελέγχεται από τη γειτονιά του, όσο και την απόσταση των υπόλοιπων σημείων της γειτονιάς μεταξύ τους.

2.2.2.4. Μέθοδοι βασισμένες σε γεωμετρικούς τόπους (domain-based)

Μία άλλη κατηγορία μεθόδων που εκτελούν αναγνώριση έκτοπων δειγμάτων σε σύνολα δεδομένων βασίζεται στην κατασκευή ενός υπερεπιπέδου που διαχωρίζει τα δεδομένα που θεωρούνται κανονικά από αυτά που αποτελούν ανωμαλίες. Κάθε δεδομένα το οποίο βρίσκεται έξω από τα όρια της περιοχής των κανονικών δεδομένων αυτομάτως ορίζεται ως ανωμαλία.

Ένας κλασικός αλγόριθμος, ο οποίος υλοποιεί την παραπάνω τεχνική είναι ο αλγόριθμος Μηχανής Διανυσμάτων Υποστήριξης Μίας Κλάσης (One Class SVM). Ο συγκεκριμένος αλγόριθμος υπολογίζει ένα υπερεπίπεδο που διαχωρίζει τα κανονικά δεδομένα από τις ανωμαλίες, σε ένα χώρο υψηλής διαστατικότητας. Αυτό επιτυγχάνεται με χρήση συναρτήσεων πυρήνα (kernels), οι οποίοι υπολογίζουν εσωτερικά γινόμενα μεταξύ σημείων των δεδομένων εισόδου. Το τελικό σύνορο που διαχωρίζει το υπερεπίπεδο είναι τέτοιο ώστε να μεγιστοποιείται η απόσταση μεταξύ των κανονικών σημείων και του συνόρου. Ο αλγόριθμος αυτός εισάγει και ένα συντελεστή χαλάρωσης υ, ο οποίος επιτρέπει σε ένα μικρό ποσοστό υ, του συνόλου των κανονικών σημείων να βρίσκεται εκτός του συνόρου, με σκοπό να αποφύγει φαινόμενα overfitting που μπορούν να επηρεάσουν αρνητικά την απόδοση των μοντέλων.

2.2.2.5. Επιπλέον μέθοδοι αναγνώρισης ανωμαλιών

Στο πεδίο της αναγνώρισης ανωμαλιών χρησιμοποιούνται παράλληλα και άλλες μέθοδοι που βασίζονται σε άλλες δομικές αρχές. Η θεωρία πληροφορίας αποτελεί τη βάση για αρκετές μεθόδους, όπως η απόκλιση Kullback-Leibler, οι οποίες εκτελούν αναγνώριση έκτοπων δειγμάτων. Την ίδια στιγμή τα νευρωνικά δίκτυα αποτελούν ένα τεράστιο πεδίο, με μεθόδους όπως τα δίκτυα Kohonen και οι Self Organized Maps να ξεχωρίζουν στην

διεργασία αυτή. Τέλος, στο πεδίο αυτό χρησιμοποιούνται και μέθοδοι απομόνωσης, όπως τα Δάση Απομόνωσης (Isolation Forests), οι οποίες χρησιμοποιούν τυχαίες δενδρικές δομές, οι οποίες εκτελούν αναδρομικές τυχαίους διαχωρισμούς σε χαρακτηριστικά των δεδομένων με σκοπό να υπολογίσουν ένα σκορ απομόνωσης (isolation score), που υποδεικνύει κατά πόσο ένα δείγμα είναι έκτοπο ή όχι.

Ένα μεγάλο κομμάτι από εργασίες στον τομέα προσεγγίζει το πρόβλημα από τη σκοπιά της στατιστικής επιστήμης [27]. Αυτού του είδους οι προσεγγίσεις βασίζονται κυρίως στη χρήση κατανομών των δεδομένων, πιθανοτήτων, καθώς και τις μπεϋζιανής θεωρίας (Bayesian Theory) με σκοπό να προσδιορίσουν ένα μοντέλο για τις ανωμαλίες. Η προσέγγιση αυτή περιορίζεται γενικά σε δεδομένα με χαρακτηριστικά σε μία ή μερικές μόνο διαστάσεις, πράγμα που την αποκλείει από την εφαρμογή στα περισσότερα σύγχρονα προβλήματα, τα οποία παρουσιάζουν μεγάλη διαστατικότητα ως προς τα χαρακτηριστικά τους.

2.2.3. Ανάλυση Κυρίαρχων Συνιστωσών (Principal Component Analysis – PCA)

Η τεχνική PCA, είναι μια γραμμική τεχνική πολλαπλών μεταβλητών, η οποία αναλύει σύνολα δεδομένων, στα οποία οι παρατηρήσεις περιγράφονται από πληθώρα αλληλεξαρτώμενων ποσοτικών μεταβλητών, μειώνοντας τη διαστατικότητά τους [28]. Στόχος της PCA είναι:

- Η εξαγωγή των πιο σημαντικών πληροφοριών από το σύνολο δεδομένων.
- Η συμπίεση του συνόλου δεδομένων κρατώντας μόνο τις πιο πλούσιες σε σημασία πληροφορίες.
- Η απλοποίηση της περιγραφής του συνόλου δεδομένων.
- Η ανάλυση της δομής των παρατηρήσεων και των μεταβλητών που τις περιγράφουν.

Για να το επιτύχει αυτό η PCA υπολογίζει νέες μεταβλητές, οι οποίες καλούνται κύριες συνιστώσες, οι οποίες προέρχονται από γραμμικούς συνδυασμούς των αρχικών μεταβλητών. Η πρώτη από αυτές τις κυρίαρχες συνιστώσες υπολογίζεται με τέτοιο τρόπο, ώστε να έχει τη μεγαλύτερη διασπορά και να περιέχει το μεγαλύτερο μέρος της πληροφορίας του συνόλου δεδομένων. Η δεύτερη υπολογίζεται με σκοπό να είναι ορθογώνια στην πρώτη, με αποτέλεσμα να μη διατηρεί τις πληροφορίες που έχουν εισαχθεί στο νέο σύνολο δεδομένων μέσω της πρώτης συνιστώσας και να είναι ασυσχέτιστη με αυτή. Η μέθοδος αυτή είναι εξαιρετικά αποτελεσματική στη συμπίεση δεδομένων, καθώς οι πρώτες συνιστώσες επιτυγχάνουν να διατηρήσουν περίπου το 90% της αρχικής πληροφορίας.

Παράλληλα, το γεγονός πως η ανάλυση PCA αναλύει τα σύνολα δεδομένων, σε σύνολα δεδομένων μικρότερων διαστάσεων, όπου τα χαρακτηριστικά αποτελούν γραμμικούς συνδυασμούς των αρχικών, μπορεί να λειτουργήσει και ως μία μέθοδος για την προστασία ευαίσθητων δεδομένων. Με την κατάλληλη επεξεργασία, τα νέα χαρακτηριστικά μπορεί να διατηρούν την αξία της αρχικής πληροφορίας σε μεγάλο βαθμό, το κάνουν όμως με τέτοιο τρόπο που αποκρύπτει την αρχική «ευάλωτη» φύση των δεδομένων. Μία τέτοια μέθοδος, η

οποία χρησιμοποιεί την PCA, σε συνδυασμό με έναν συντελεστή μετατόπισης (shift factor), για επιπλέον προστασία περιγράφεται στο [29].

2.3. Η αρχιτεκτονική Representational State Transfer (REST)

Η αρχιτεκτονική Representational State Transfer (REST) προτάθηκε για πρώτη φορά από τον Roy T. Fielding στη διδακτορική διατριβή του [30]. Βασικά στοιχεία της αρχιτεκτονικής αυτής είναι τα [31]:

- **Πόρος (Resource):** μπορεί να είναι οτιδήποτε, από ένα πραγματικό αντικείμενο ως μία αφαιρετική έννοια, από τη στιγμή που είναι αρκετά σημαντικό, ώστε να υπάρχουν αναφορές προς αυτό. Στα πλαίσια της αρχιτεκτονικής REST, συνήθως αναπαριστά κάτι το οποίο μπορεί να αποθηκευτεί σε έναν ηλεκτρονικό υπολογιστή.
- **Αναπαράσταση (Representation):** είναι οποιαδήποτε χρήσιμη πληροφορία σχετικά με την κατάσταση ενός πόρου. Ένας πόρος μπορεί κατά προέκταση να έχει παραπάνω από μία διαφορετικές αναπαραστάσεις.
- **Κατάσταση (State):** στην αρχιτεκτονική REST υπάρχουν δύο διακριτοί τύποι καταστάσεων. Ο πρώτος είναι η κατάσταση ενός πόρου, ο οποίος περιέχει πληροφορίας σχετικά με έναν πόρο, και ο δεύτερος είναι η κατάσταση εφαρμογής, στην οποία περιλαμβάνονται όλες εκείνες οι πληροφορίες που προσδιορίζουν το μονοπάτι που ακολούθησε ένας χρήστης (client) εντός μίας εφαρμογής.

Έχοντας αυτά τα δομικά στοιχεία ως βάση, καθώς επίσης και ένα σύνολο από αρχιτεκτονικούς περιορισμούς, η αρχιτεκτονική REST προτείνει ένα πρωτόκολλο διαδικτυακής επικοινωνίας μεταξύ εξυπηρετητών (servers) και πελατών (clients), μέσω HTTP πακέτων και APIs.

Διάγραμμα 3: Επεξήγηση λειτουργίας της αρχιτεκτονικής REST [31]

Στο παραπάνω διάγραμμα [Διάγραμμα 3] απεικονίζεται συνοπτικά ο τρόπος που χρησιμοποιώντας μεθόδους HTTP, οι οποίες προορίζονται να χρησιμοποιηθούν σε HTTP APIs, συμπεριλαμβανομένων και των RESTful APIs. Οι βασικές μέθοδοι είναι οι παρακάτω [32]:

- GET: λαμβάνει την αναπαράσταση της κατάστασης του πόρου στόχου.

- POST: αφήνει τον πόρο στόχο να επεξεργαστεί την αναπαράσταση που περικλείεται στο αίτημα.
- PUT: δημιουργεί ή αλλάζει την κατάσταση του πόρου στόχου με την κατάσταση που περικλείεται στην αναπαράσταση που περιέχεται στο αίτημα.
- DELETE: διαγράφει την κατάσταση του πόρου στόχου.

2.4. Εφαρμογές που συνδυάζουν blockchain και τεχνητή νοημοσύνη

Όπως είναι φυσιολογικό, όντας δύο από τις πιο ακμάζουσες τεχνολογίες στην εποχή μας, έχουν γίνει αρκετές προσπάθειες να συνδυαστούν σε ολοκληρωμένα συστήματα το blockchain και η τεχνητή νοημοσύνη.

2.4.1. Η συνδρομή του blockchain στην τεχνητή νοημοσύνη

Σύμφωνα με την έρευνα του [33] υπάρχουν συγκεκριμένοι τομείς που τα χαρακτηριστικά του blockchain μπορούν να συνδράμουν στις μεθόδους της τεχνητής νοημοσύνης. Διαχωρίζοντάς με βάση το περιεχόμενό τους προκύπτουν οι παρακάτω κατηγορίες, όπως αυτές παρουσιάζονται στις υποενότητες που ακολουθούν.

2.4.1.1. Διαχείριση δεδομένων βασισμένη στο blockchain

Ένα από τα βασικά ζητήματα που αποτελούν στενωπό στην διασφάλιση της ιδιωτικότητας των δεδομένων των εφαρμογών τεχνητής νοημοσύνης και της γενικότερης ασφάλειας είναι ο κεντρικός τρόπος διαχείρισης και αποθήκευσης των δεδομένων [34]. Το πρόβλημα αυτό διογκώνεται όταν οι εφαρμογές πραγματεύονται ευαίσθητα προσωπικά δεδομένα. Οι βασικές ιδιότητες του blockchain, όπως είναι η κατανεμημένη φύση του επιτρέπουν στις εφαρμογές τεχνητής νοημοσύνης που χρησιμοποιούν τέτοιους τρόπους αποθήκευσης να ξεπερνούν αυτό το πρόβλημα.

2.4.1.2. Διαχείριση marketplaces βασισμένη στο blockchain

Ο αποκεντρωμένος αυτός τρόπος διαχείρισης των marketplaces, όπου η λειτουργία του marketplace συντελείται μέσω ενός κατανεμημένου δικτύου ομότιμων κόμβων υπόσχεται μειωμένα κόστη συναλλαγών, ενώ παράλληλα αυξάνει την δικαιοδοσία που έχουν οι χρήστες πάνω στα δεδομένα τους, γεννώντας ευκαιρίες ως προς την πιο ελεύθερη διακίνηση δεδομένων, τα οποία μπορούν να χρησιμοποιηθούν προς το όφελος της τεχνητής νοημοσύνης [35]. Συγκεκριμένα στο άρθρο του Subramanian [35], πραγματοποιείται μία ενδελεχής σύγκριση μεταξύ των παραδοσιακών μορφών ηλεκτρονικών marketplaces και των αποκεντρωμένων marketplaces που βασίζονται σε εφαρμογές του blockchain, η οποία παρουσιάζει τα συγκριτικά πλεονεκτήματα των πρώτων έναντι των δεύτερων σε πολλαπλούς τομείς [Πίνακας 2]. Επιπλέον, γίνεται μία πρόβλεψη της πιθανότητας

υιοθέτησης τέτοιων αποκεντρωμένων μορφών marketplaces σε διάφορους τομείς με συνοπτική παρουσίαση του κύριου λόγου που οδηγεί το συγγραφέα σε αυτό το συμπέρασμα [Πίνακας 3].

Χαρακτηριστικό Marketplace	Αποκεντρωμένο Marketplace Βασισμένο στο Blockchain	Παραδοσιακό E-Marketplace
Εμπιστοσύνη μέσω υποχρεωτικής συμμόρφωσης με το συμβόλαιο	Κατανεμημένη επικύρωση της συμφωνίας μέσω μηχανισμών proof-of-work και proof-of-stake. Το δίκτυο υποχρεώνει την εκτέλεση του συμβολαίου για τους συμβαλλόμενους και επικυρώνει την αξιολόγησή τους.	Τρίτες οντότητες, όπως τράπεζες, εγγυώνται για το συμβόλαιο. Συχνά η ευθύνη αναλαμβάνεται από τη διαχειριστική οντότητα του marketplace. Υπάρχει πιθανότητα για αλλαγές.
Διάρκεια συναλλαγής	Μέχρι και άμεση λόγω της ταχύτητας της επικύρωσης μέσω του δικτύου. Καθυστερήσεις μπορεί να ευθύνονται στους μηχανισμούς consensus που χρησιμοποιούνται.	Γραφειοκρατικοί μηχανισμοί και διαδικασίες, πχ. letter of credit, που περιέχουν μεγάλη καθυστέρηση.
Διαχείριση αξίας	Επιβράβευση των συμμετεχόντων στο δίκτυο με tokens.	Τραπεζικά συστήματα
Ιδιωτικότητα και ασφάλεια	Η ταυτότητα των εμπλεκομένων δεν φανερώνεται στο δίκτυο. Η ιχνηλασιμότητα των συναλλαγών είναι δυνατόν να συντελεστεί αν και με κάποια δυσκολία. Οι λεπτομέρειες των συναλλαγών είναι δυνατό να κρύβονται πίσω από επίπεδα κρυπτογράφησης. Το κόστος αλλοίωσης του μηχανισμού επικύρωσης του δικτύου είναι πάρα	Η ταυτότητα του συναλλασσόμενου είναι εντελώς φανερή μέσα στο marketplace. Η ασφάλεια του marketplace είναι άμεσα εξαρτώμενη από την ασφάλεια των υποσυστημάτων που το απαρτίζουν.

	πολύ μεγάλο.	

Πίνακας 2: Σύγκριση αποκεντρωμένων e-marketplaces με παραδοσιακά e-marketplaces [35]

E-Marketplace	Πιθανότητα Αποκέντρωσης	Λόγοι
Φυσικά προϊόντα	Μερική αποκέντρωση	Δυνατότητα αποκέντρωσης τμημάτων όπως η υποστήριξη B2B, η λογιστική, η πληρωμή και η διαχείριση της φήμης (reputation)
Ψηφιακά προϊόντα	Πολύ πιθανή	Πλήρης διαδικτυακή πληρωμή και παράδοση προϊόντων
Marketplaces περιεχομένου παρεχόμενου από τους χρήστες	Πολύ πιθανή	Διαδικτυακό περιεχόμενο και διαδικτυακή φήμη χρηστών
Πρόβλεψη αγορών	Πολύ πιθανή	Επικύρωση μέσω blockchain για επιβολή συμβολαίων
Crowdfunding	Πολύ πιθανή	Ευκολότερη επικύρωση και λειτουργικότητα παρεχόμενη από το blockchain
Ανταλλακτήρια νομισμάτων και σύνθετα οικονομικά συμβόλαια	Πολύ πιθανή	Εύκολη δημιουργία σύνθετων έξυπνων συμβολαίων και χαμηλά κόστη συναλλαγής

Πίνακας 3: Αποκέντρωση σε διαφορετικούς τύπους marketplaces [35]

Σε αυτή την κατεύθυνση προορίζεται το άρθρο [36], στο οποίο προτείνεται ένα συνολικό σύστημα διαχείρισης και αγοραπωλησίας δεδομένων παραγόμενων από αισθητήρες IoT, βασισμένο πάνω στο blockchain. Στο άρθρο αυτό προτείνεται η χρήση έξυπνων συμβολαίων αναπτυγμένων σε γλώσσα Solidity, τα οποία είναι διαθέσιμα, μέσω του δικτύου Alastria, το οποίο είναι ένα public-permissioned network. Η συμμετοχή στο δίκτυο είναι ελεύθερη και προστατευμένη, ενώ παράλληλα λειτουργεί και ένα σχήμα επιβράβευσης με ένα εσωτερικό token, το Msec token. Για την διαφύλαξη της ιδιωτικότητας περιήγησης στο marketplace και την ύπαρξη παράλληλα ελέγχου ταυτότητας κατά την είσοδο χρησιμοποιείται ένα τρίτο υποσύστημα ελέγχου ταυτότητας μέσω OpenID και Oauth2 που αναφέρεται ως Security Manager. Η πιλοτική εφαρμογή του συστήματος αυτού σε έξυπνες πόλεις Ευρώπη και Ιαπωνία δύναται να συνεισφέρει στην παραγωγή και διάθεση χρήσιμων δεδομένων από τις έξυπνες αυτές πόλεις, προς όφελος ανάπτυξης έξυπνων λύσεων που διαχειρίζονται τα δεδομένα αυτά για να λάβουν έξυπνες αποφάσεις.

2.4.1.3. Αρχιτεκτονικές τεχνητής νοημοσύνης βασισμένες στο blockchain

Σε αυτό το πεδίο των προτεινόμενων λύσεων, οι έξυπνοι πράκτορες συνθέτουν ένα δίκτυο κατανεμημένης τεχνητής νοημοσύνης, όπου η αίτησης για δεδομένα και η παροχής αποτελεσμάτων συντελείται μέσω έξυπνων συμβολαίων [37]. Συγκεκριμένα, οι Montes και Goertzel, προτείνουν το SingularityNET, ένα δίκτυο το οποίο και αυτό βασίζεται σε ένα κατανεμημένο marketplace δεδομένων. Στην πλατφόρμα αυτή λειτουργούν πράκτορες τεχνητής νοημοσύνης, οι οποίοι ανεξαρτήτως της εσωτερικής δομής τους προορίζονται για την επίλυση συγκεκριμένων προβλημάτων. Η πλατφόρμα παρέχει στους πράκτορες τη δυνατότητα outsourcing έργου σε τρίτους, ανταλλαγής πληροφοριών, διακανονισμού πληρωμών και βαθμολογίας πρακτόρων, η οποία επηρεάζει την φήμη τους στην πλατφόρμα. Για να συντελεστεί αυτό η πλατφόρμα παρέχει ένα εσωτερικό token, το AGIX token, το οποίο χρησιμοποιείται ως συνάλλαγμα για τις εσωτερικές συναλλαγές στην πλατφόρμα, αλλά και ως μέσο επιβράβευσης.

2.4.1.4. Συστήματα σμήνους (swarm systems) βελτιωμένα μέσω blockchain

Στο πεδίο αυτό της συλλογικής λήψης αποφάσεων από κατανεμημένους πράκτορες, το blockchain μπορεί να συνδράμει στην εγκαθίδρυση ενός αφαλούς πλαισίου επικοινωνίας μεταξύ των πρακτόρων και την προστασία του συνόλου από κακόβουλα μέλη [38]. Συγκεκριμένα, στο άρθρο του ο Ferrer εστιάζει σε τέσσερις παράγοντες στους οποίους το blockchain μπορεί να βελτιώσει την υπάρχουσα πραγματικότητα στα συστήματα σμήνους: την ασφάλεια, την κατανεμημένη λήψη αποφάσεων, τη διαφοροποίηση στη συμπεριφορά και την παροχή νέων επιχειρηματικών μοντέλων. Στο κομμάτι της ασφάλειας προτείνεται η χρήση δημόσιων και ιδιωτικών κλειδιών και η κρυπτογράφηση των μηνυμάτων με βάση αυτά, έτσι ώστε να διασφαλίζεται η ταυτότητα του αποστολέα και του παραλήπτη και η προστασία του περιεχομένου του μηνύματος ακόμα και σε δημόσια κανάλια επικοινωνίας, τα οποία μπορεί να παρακολουθούν και εξωτερικοί παρατηρητές. Στο κομμάτι της λήψης αποφάσεων, το άρθρο προτείνει ένα διαφανή τρόπο λήψης αποφάσεων από τα μέλη του σμήνους. Κάθε φορά που ένα μέλος βρίσκεται σε μία κατάσταση όπου πρέπει να λάβει μία απόφαση, ξεκινά μία ειδική συναλλαγή, δημιουργώντας μία διεύθυνση για καθεμία από τις επιλογές που έχει. Τα υπόλοιπα μέλη του σμήνους ψηφίζουν, μεταφέροντας tokens στην επιλογή που πιστεύουν ως καλύτερη μέχρι να επιτευχθεί πλειοψηφία. Ο μηχανισμός αυτός προσφέρει έναν ασφαλή και ελεγχόμενο τρόπο απόφασης, αφού όλα τα μέλη μπορούν να παρακολουθήσουν το υπόλοιπο των διευθύνσεων που συμμετείχαν στη διαδικασία ψηφοφορίας.

2.4.1.5. Αύξηση διαφάνειας αποφάσεων τεχνητής νοημοσύνης μέσω blockchain

Ένα από τα βασικά μειονεκτήματα των μοντέλων τεχνητής νοημοσύνης είναι η έλλειψη ερμηνείας των αποφάσεών τους, σε σημείο που αυτές μπορεί να χάνουν την αξία τους. Η χρήση του blockchain στο πεδίο, όχι μόνο μπορεί να αυξήσει την εμπιστοσύνη του κοινού στα μοντέλα τεχνητής νοημοσύνης, αλλά και να παρέχει έναν διαφανή τρόπο εξερεύνησης της διαδικασίας που ακολουθούν τα μοντέλα αυτά για να λάβουν την τελική τους απόφαση [39]. Η διαφάνεια στις διαδικασίες που ακολουθούν τα μοντέλα για να οδηγηθούν στη λήψη αποφάσεων, η οποία αναφέρεται συχνά με τον όρο Explainable AI (XAI), δύναται πέραν της αύξησης της εμπιστοσύνης στην. τεχνητή νοημοσύνη, να φέρει στην επιφάνεια τα πλεονεκτήματα και τις αδυναμίες των διαδικασιών αυτών αλλά και να δώσει μία αίσθηση της μελλοντικής συμπεριφοράς τέτοιων συστημάτων [40].

2.4.2. Η συνδρομή της τεχνητής νοημοσύνης στο blockchain

Αντίστοιχη βελτίωση και προσφορά αξίας μπορεί να επιφέρει και η υιοθέτηση της τεχνητής νοημοσύνης σε εφαρμογές blockchain. Η τεχνητή νοημοσύνη μπορεί να επέμβει και να βελτιώσει δομικά στοιχεία των δικτύων blockchain, όπως τα έξυπνα συμβόλαια και η διαδικασία της εξόρυξης (mining) των blocks.

2.4.2.1. Τεχνητή νοημοσύνη στα έξυπνα συμβόλαια

Στο πεδίο των έξυπνων συμβολαίων μεταξύ άλλων συστήματα τεχνητής νοημοσύνης θα μπορούσαν να αναλύουν τη συμπεριφορά εμπλεκόμενων για να τους προτείνουν τις βέλτιστες συνθήκες ώστε αυτοί να έρθουν σε συμφωνία, είτε να προσδιορίζουν τη βέλτιστη στιγμή για την εκτέλεση ενός έξυπνου συμβολαίου, που προχωρά για παράδειγμα σε μία αγορά την κατάλληλη τιμολογιακά στιγμή [41,42].

Συγκεκριμένα στο άρθρο [41] προτείνεται η χρήση της τεχνητής νοημοσύνης με σκοπό την κατασκευή έξυπνων συμβολαίων, τα οποία μαθαίνουν μόνα τους, είναι δηλαδή self-learned, και αλλάζουν τη συμπεριφορά τους και την επιχειρησιακή τους λογική με σκοπό να ανταποκρίνονται σε αλλαγές στο δίκτυο. Σύμφωνα με το άρθρο, ορίζονται αρχικά κανόνες, οι οποίοι καλύπτουν διάφορα σενάρια της εφαρμογής, τα οποία μπορεί να μην είναι ενεργά κατά την εκκίνηση του συστήματος αλλά προορίζονται να ενεργοποιηθούν υπό συγκεκριμένες προϋποθέσεις. Οι προϋποθέσεις ενεργοποίησης των κανόνων αυτών προκύπτουν από ανάλυση των δεδομένων της αλυσίδας και σηματοδοτούνται συνήθως από την εκπομπή γεγονότων. Για παράδειγμα η αύξηση των συμμετεχόντων στο δίκτυο θα μπορούσε να οδηγήσει στην εκπομπή ενός γεγονότος, το οποίο θα «πυροδοτούσε» με τη σειρά του μια προκαθορισμένη αντίδραση, όπως ένα νέο έξυπνο συμβόλαιο ή μία νέα λειτουργία στη λογική της αλυσίδας. Με τον τρόπο αυτό αναλύεται η συμπεριφορά των συμμετεχόντων στο δίκτυο και καθορίζεται ένα νέο σύνολο κανόνων, το οποίο βασίζεται σε αυτές τις συμπεριφορές για να πετύχει συγκεκριμένους στόχους.

2.4.2.2. Τεχνητή νοημοσύνη στη διαδικασία εξόρυξης

Στο πεδίο της διαδικασίας της εξόρυξης, η τεχνητή νοημοσύνη μπορεί να χρησιμοποιηθεί είτε για να εκτιμήσει το χρονικό διάστημα μέχρι την επόμενη επιβεβαίωση συναλλαγής, είτε για να βελτιώσει τους υπάρχοντες μηχανισμούς consensus, αυξάνοντας την ενεργειακή τους απόδοση και την ασφάλειά τους [43,44]. Συγκεκριμένα στο άρθρο [43], ερευνάται η χρήση μηχανικής μάθησης με σκοπό την πρόβλεψη του χρόνου επιβεβαίωσης συναλλαγής στο Ethereum blockchain. Ελέγχονται τρία μοντέλα, οι Naive Bayes Classifiers, τα Random Forests και τα Multi-Layer Perceptrons, ως προς την ικανότητά τους να προβλέψουν αν και πότε μία συναλλαγή θα εγκριθεί. Τα ευρήματα του άρθρου συνιστούν ως βέλτιστο τύπο τα Random Forests, αν και ο συγγραφέας εκφράζει την ανάγκη για μελέτη και άλλων νευρωνικών δομών, καθώς επίσης και για έρευνα στον τρόπο που μπορούν οι εκτιμήσεις αυτές να κάνουν τις συναλλαγές στο δίκτυο οικονομικότερες και πιο γρήγορες για ένα χρήστη στο μέλλον.

Παράλληλα, στο άρθρο [44] προτείνεται ένας εναλλακτικός μηχανισμός consensus για τους κόμβους του blockchain, ο οποίος αναφέρεται ως Proof of Artificial Intelligence (PoAI). Ο μηχανισμός αυτός κατατάσσει αρχικά όλους τους κόμβους του δικτύου μέσω ενός Συνελικτικού Νευρωνικού Δικτύου (Convolutional Neural Network – CNN), σύμφωνα με διάφορα χαρακτηριστικά που αντικατοπτρίζουν την ποιότητά τους ως κόμβους του δικτύου. Οι κορυφαίοι των κόμβων αυτών έχουν το δικαίωμα να συμμετάσχουν στη διαδικασία εξόρυξης, όπου ο κόμβος εξόρυξης επιλέγεται σύμφωνα με ένα μηχανισμό περιστροφής. Ο υπολογισμός πολλών χαρακτηριστικών κατά την κατάταξη των κόμβων αποφορτίζει τους κόμβους από την ανάγκη συναγωνισμού για υπολογιστική ισχύ, εξοικονομώντας ενέργεια και διασφαλίζοντας μεγαλύτερη δικαιοσύνη και αποκέντρωση στη διαδικασία επιλογής, πράγμα που διαφαίνεται και στα αποτελέσματα της έρευνας.

2.4.3. Εφαρμογές όπου τεχνητή νοημοσύνη και blockchain αλληλοσυμπληρώνονται

Τέλος, σημαντικές είναι και οι εφαρμογές, όπου οι δύο αυτές τεχνολογίες συνδυάζονται ισότιμα αλληλοσυμπληρώνοντας η μία την άλλη και δημιουργώντας ολοκληρωμένες λύσεις που χαίρουν των πλεονεκτημάτων και των δύο. Μία τέτοια εφαρμογή προτείνεται από το Μαρκόπουλο [45], όπου το blockchain και η τεχνητή νοημοσύνη συνδυάζονται για να κατασκευάσουν ένα Δημοκρατικό Μοντέλο Ομαδοποίησης (Democratic Teaming Model), το οποίο υπόσχεται να αυτοματοποιήσει και να εκδημοκρατίσει τη διαδικασία ομαδοποίησης του ανθρωπίνου δυναμικού μέσα σε οργανισμούς. Συγκεκριμένα, στο άρθρο προτείνεται η χρήση της τεχνητής νοημοσύνης, μέσω expert models, για την παροχή προτάσεων ομαδοποίησης του προσωπικού, όπως προκύπτουν από τη διαρκή παρακολούθηση των δραστηριοτήτων, της συμπεριφοράς, των ενδιαφερόντων και των εμπειριών του προσωπικού από το σύστημα αυτό. Η προστασία και η ροή των ευαίσθητων αυτών πληροφοριών διασφαλίζεται από το blockchain, με τα δεδομένα να αποθηκεύονται εξωτερικά του οργανισμού. Έτσι το σύστημα μπορεί αντικειμενικά να αναλύει τα δεδομένα που του παρέχονται και να βελτιώνει την ανάλυσή του αυτή ανάλογα με την εξέλιξη του έργου για το οποίο χρειάζεται να σχηματιστεί η ομάδα.

Επιπλέον, στο [46] προτείνεται ένα ολοκληρωμένο σύστημα που επιτρέπει σε ιδρύματα να μοιραστούν ευαίσθητα δεδομένα χωρίς τον κίνδυνο έκθεσής τους, με σκοπό αυτά να τροφοδοτηθούν και να συνδράμουν στην εκπαίδευση μοντέλων τεχνητής νοημοσύνης. Συγκεκριμένα, ο συγγραφέας παρατηρεί τη δυσκολία που συναντούν οι οργανισμοί να μοιραστούν μεταξύ τους δεδομένα, λόγω της ανάγκης προστασίας της ιδιωτικότητας. Τονίζεται όμως πως τα δεδομένα αυτά έχουν μεγάλη αξία, καθώς το μοντέλα μηχανικής μάθησης μπορούν να επωφεληθούν αρκετά από την ύπαρξη μεγαλύτερου συνόλου δεδομένων για εκπαίδευση. Οι συγγραφείς προτείνουν μία λύση βασισμένη στο blockchain με σκοπό να ξεπεράσουν το πρόβλημα της ιδιωτικότητας και να επιτρέψουν στα μοντέλα να εκπαιδευθούν στα μεγάλα αυτά σύνολα δεδομένων, χωρίς όμως να κινδυνεύουν τα ευαίσθητα δεδομένα. Στην προτεινόμενη λύση το blockchain αναλαμβάνει τον έλεγχο της μηχανής εκπαίδευσης. Η μηχανή εκπαίδευσης αποτελεί ένα ανεξάρτητο, ασφαλές και «στεγανό» περιβάλλον στο οποίο τροφοδοτούνται ως είσοδοι τα δεδομένα εκπαίδευσης και το ανεκπαίδευτο μοντέλο μηχανικής μάθησης. Η μηχανή εκπαίδευσης προχωρά εσωτερικά στην εκπαίδευση του μοντέλου και επιστρέφει ως έξοδο το εκπαιδευμένο μοντέλο. Μετά την εσωτερική εκπαίδευση τα δεδομένα διαγράφονται άμεσα, ώστε να μην υπάρχει κίνδυνος να εκτεθούν σε τρίτους.

Κεφάλαιο 3

Αρχιτεκτονική πληροφοριακού συστήματος

3.1. Απαιτήσεις συστήματος

3.1.1. Λειτουργικές απαιτήσεις

Το συνολικό σύστημα που προτείνεται μέσω της παρούσας διπλωματικής, παρά την αρθρωτή φύση του έχει σκοπό να λειτουργεί ως μία ολότητα και να ικανοποιεί ένα σύνολο λειτουργικών και μη λειτουργικών απαιτήσεων, όπως αυτές περιγράφονται ακολούθως. Οι απαιτήσεις αυτές είναι ρεαλιστικές και σχετικές με ένα ευρύ φάσμα εφαρμογών στις οποίες θα μπορούσε να χρησιμοποιηθεί το παρόν σύστημα.

- Επικοινωνία και λειτουργία της εφαρμογής με το Ethereum blockchain.
- Αρθρωτή κατασκευή και επικοινωνία υποσυστημάτων μέσω REST API.
- Επικοινωνία των υποσυστημάτων υποβολής αιτημάτων και λήψης αποφάσεων με το IPFS.
- Μηδενική τοπική επιβάρυνση των υποσυστημάτων με όγκο αρχείων. Αποθήκευση εξολοκλήρου στο IPFS και διατήρηση δεδομένων αυστηρά και μόνο για το χρόνο που αυτά απαιτούνται.
- Πλήρης δυνατότητα παραμετροποίησης διεπαφών μέσω αρχείων .config.
- Υποστήριξη μεγάλου εύρους μοντέλων μηχανικής μάθησης.
- Διατήρηση της ανωνυμίας των χρηστών κατά τη χρήση της εφαρμογής. Η πλοήγηση σε όλες της διεπαφές γίνεται με το ανώνυμο πορτοφόλι του χρήστη.
- Πλήρης αυτοματοποίηση της διαδικασίας απόφασης από τους κόμβους απόφασης. Τα γεγονότα αιτημάτων ανιχνεύονται αυτόματα, όπως αυτόματα διεκπεραιώνεται και η παροχή εκτίμησης.

3.1.2. Μη λειτουργικές απαιτήσεις

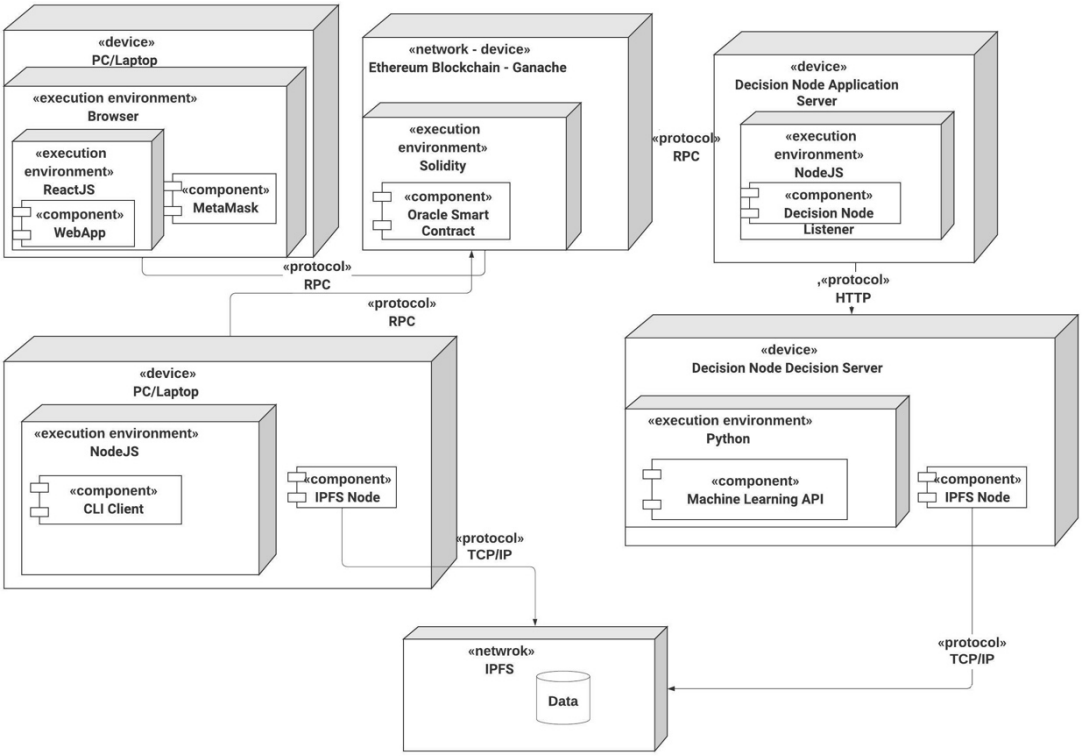
- Εύχρηστες και λειτουργικές διεπαφές για χρήστες διαφορετικών απαιτήσεων και εξοικείωσης με την τεχνολογία.
- Υψηλά επίπεδα ασφάλειας σε όλα τα υποσυστήματα της εφαρμογής.

- Αυτονομία υποσυστημάτων και δυνατότητα επέκτασης μεμονωμένα χωρίς να επηρεάζεται το υπόλοιπο σύστημα.
- Προστασία των ευαίσθητων δεδομένων μέσω κατανεμημένης αποθήκευσής τους.
- Έλλειψη μοναδικών σημείων αποτυχίας (single point of failure) για εξασφάλιση μέγιστης δυνατής διαθεσιμότητας.

3.2. Αρχές επιλογής επιμέρους τεχνολογιών και επεξήγηση λειτουργίας επιμέρους συστημάτων

3.2.1. Συνολική προτεινόμενη αρχιτεκτονική

Στην παρακάτω εικόνα [Διάγραμμα 4] αναπαρίσταται μία συνολική άποψη της αρχιτεκτονικής του προτεινόμενου συστήματος. Απεικονίζονται τα επιμέρους συστήματα και τα συστατικά τους, καθώς επίσης και τα κύρια κανάλια επικοινωνίας με τα πρωτόκολλα που τις διέπουν. Τα επιμέρους υποσυστήματα πρόκειται να κατασκευαστούν σύμφωνα με τα όσα υπαγορεύει το παραπάνω διάγραμμα, βεβαιώνοντας πως παρά τις όποιες ιδιαιτερότητες κατά την εσωτερική ανάπτυξή τους πληρούν τις ελάχιστες προϋποθέσεις για επικοινωνία με το υπόλοιπο σύστημα, καθώς και φιλοξενία στους ανάλογους εξυπηρετητές.



Διάγραμμα 4: Διάγραμμα UML Component της συνολικής αρχιτεκτονικής

3.2.2. Ethereum blockchain – Ganache

3.2.2.1. Κριτήρια επιλογής

Ως πλατφόρμα υλοποίησης του δικτύου στο blockchain επιλέχθηκε η χρήση του Ethereum. Η ευρεία χρήση του συγκεκριμένου δημόσιου blockchain στον προγραμματιστικό κόσμο μας παρέχει μεγάλη ευελιξία ως προς τα εργαλεία που έχουν στη διάθεσή μας για την ανάπτυξη του προτεινόμενου πληροφοριακού συστήματος. Ιδιαίτερα αν ληφθεί υπόψιν το γεγονός πως στα πλαίσια του συστήματος συνεργάζονται πολλαπλά υποσυστήματα διαφορετικών τεχνολογιών, η χρήση μιας λύσης blockchain για την οποία έχουν αναπτυχθεί ήδη βιβλιοθήκες που τη γεφυρώνουν με συστήματα βασικών τεχνολογιών αποτελούσε βασική απαίτηση.

Στα πλαίσια της ανάπτυξης του πληροφοριακού συστήματος, χρησιμοποιήθηκε το πρόγραμμα Ganache, με σκοπό να προσομοιωθεί ένα δίκτυο ελέγχου Ethereum blockchain, προσφέροντας παράλληλα επιπλέον λειτουργικότητα, όπως διεπαφή επισκόπησης, η οποία βοηθά κατά τη φάση της ανάπτυξης της εφαρμογής.

3.2.2.2. Έξυπνα συμβόλαια (smart contracts)

Ένα από τα βασικά συστατικά της προτεινόμενης λύσης αποτελεί το έξυπνο συμβόλαιο (smart contract), το οποίο έχει αναπτυχθεί στην προγραμματιστική γλώσσα Solidity. Το συμβόλαιο αποτελεί την καρδιά της κατανεμημένης εφαρμογής, αφού περικλείει όλη την επιχειρησιακή λογική (business logic) που αντιμετωπίζει το ζήτημα της λήψης έξυπνων αποφάσεων, μέσω δεδομένων από εξωτερικά συστήματα. Για να το επιτύχει αυτό, το έξυπνο συμβόλαιο ορίζει δομές δεδομένων που περιγράφουν τα δεδομένα ενός αιτήματος, εκπέμπει γεγονότα για να επιτρέπει σε εξωτερικά συστήματα να αντιλαμβάνονται την ανάγκη να εισάγουν στην αλυσίδα χρήσιμα εξωτερικά δεδομένα, παρέχει μεθόδους διεπαφής με τα δεδομένα της αλυσίδας και ορίζει με άμεσο τρόπο κόμβους αυξημένων δικαιωμάτων και αρμοδιοτήτων. Τα χαρακτηριστικά αυτά του έξυπνου συμβολαίου θα αναλυθούν στις υποενότητες που ακολουθούν.

3.2.2.3. Τύποι δικαιωμάτων κόμβων

Όπως προκύπτει και από την αρχική περιγραφή του συστήματος οι εμπλεκόμενοι στο πληροφοριακό σύστημα διαχωρίζονται σε δύο κατηγορίες:

- **Απλούς Χρήστες:** Έχουν τη δυνατότητα να προσπελάσουν τα δεδομένα του blockchain και να υποβάλουν νέα αιτήματα στην εφαρμογή. Δε συμμετέχουν στη λήψη απόφασης με υποβολή εκτιμήσεων πάνω στα δεδομένα.
- **Χρήστες με μοντέλα εκτίμησης:** Έχουν όλες τις δυνατότητες του απλού χρήστη και επιπλέον συμμετέχουν στη λήψη αποφάσεων προσκομίζοντας στο blockchain τις εκτιμήσεις των μοντέλων μηχανικής μάθησης που έχουν, μέσω των υπηρεσιών NodeJS Oracle Service.

Η επιλογή ενός public permissionless τύπου blockchain, όπως το Ethereum, για την κατασκευή της εφαρμογής δεν παρέχει τη δυνατότητα να καθορίσουμε κατά την κατασκευή των κόμβων, κόμβους διαφορετικών δικαιωμάτων, όπως θα συνέβαινε σε τύπους private ή

consortium. Παρ ‘όλα αυτά, μέσω του σώματος του έξυπνου συμβολαίου δίνεται η δυνατότητα παράκαμψης της αδυναμίας αυτής του Ethereum. Συγκεκριμένα, γνωρίζοντας εκ των προτέρων τις διευθύνσεις των κόμβων που θέλουμε να έχουν αυξημένα δικαιώματα και να συμμετέχουν στη διαδικασία απόφασης, τους ορίζουμε απευθείας στο έξυπνο συμβόλαιο, εισάγοντας παράλληλα έναν έλεγχο διεύθυνσης πριν εκτελεστεί οποιαδήποτε διαδικασία που απαιτεί αυξημένα δικαιώματα. Με τον τρόπο αυτό βεβαιώνεται πως στο τμήμα αυτό του συμβολαίου εισέρχονται μόνο οι κόμβοι αυξημένων δικαιωμάτων, ενώ παράλληλα στο ίδιο σημείο του κώδικα γίνεται και έλεγχος, ώστε κάθε κόμβος απόφασης να συμμετέχει κατά μέγιστο μία φορά στη λήψη της απόφασης για ένα συγκεκριμένο αίτημα, αποφεύγοντας έτσι το φαινόμενο της «διπλής-ψήφου».

Η τροποποίηση αυτή του έξυπνου συμβολαίου, ώστε να επιτελεί αυτή την επιπλέον λειτουργία της διαχείρισης δικαιωμάτων επιτάσσει την αυξημένη προσοχή κατά τον ορισμό των διευθύνσεων αυτών των κόμβων. Επιπλέον, δημιουργείται μία νέα διαρκής ανάγκη για ανανέωση του έξυπνου συμβολαίου κάθε φορά που απαιτείται προσθήκη, αφαίρεση ή αλλαγή διεύθυνσης ενός κόμβου αυξημένων δικαιωμάτων.

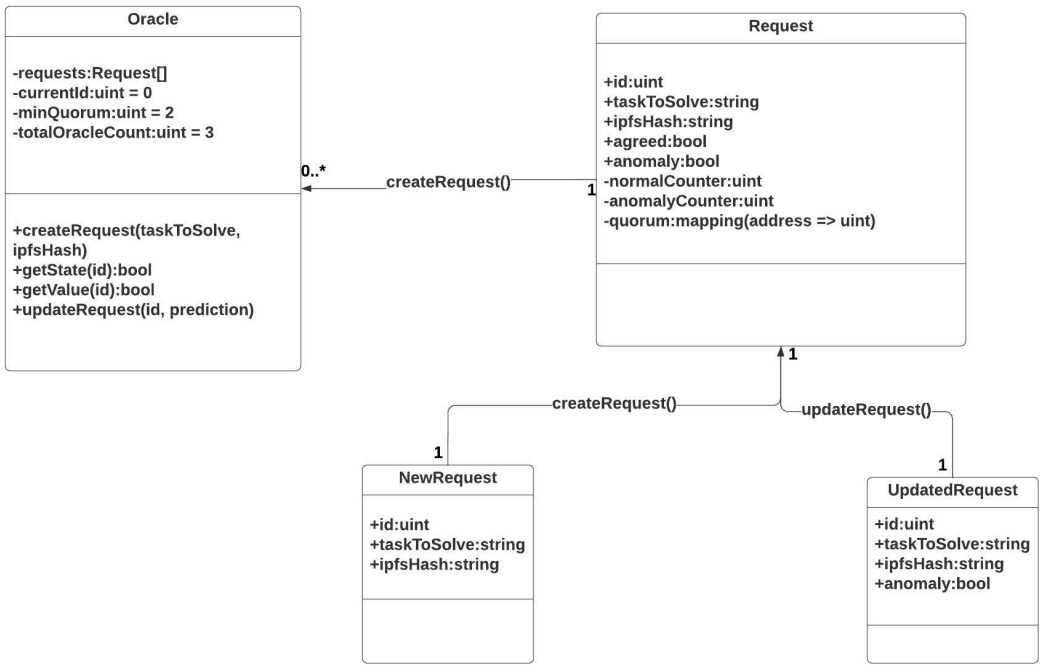
3.2.2.4. Εκπομπή γεγονότων

Το έξυπνο συμβόλαιο χρησιμοποιεί τα γεγονότα (events) ως μηχανισμό ενημέρωσης του εξωτερικού στο blockchain κόσμου και των παρατηρητών που λειτουργούν σε αυτό για την λήψη αποφάσεων και την ανάγκη παροχής δεδομένων από εξωτερικά συστήματα. Πιο συγκεκριμένα, το έξυπνο συμβόλαιο εκπέμπει δύο είδη γεγονότων που παρουσιάζονται παρακάτω:

- Γεγονός **NewRequest**: Το γεγονός αυτό εκπέμπεται κατά την εισαγωγή ενός νέου αιτήματος προς εκτίμηση στο blockchain. Ανιχνεύεται από εξωτερικού παρατηρητές, όπως το NodeJS Oracle Service, οι οποίοι είναι εγκατεστημένοι στους κόμβους απόφασης και σηματοδοτεί την ανάγκη για παροχή της εκτίμησής των εξωτερικών μοντέλων πάνω στα δεδομένα που παρέχει στο σώμα του.
- Γεγονός **UpdatedRequest**: Το γεγονός αυτό σηματοδοτεί τη λήψη μίας απόφασης για ένα συγκεκριμένο αίτημα από τους κόμβους απόφασης του blockchain. Το γεγονός αυτό εκπέμπεται μόλις επιτευχθεί συμφωνία μεταξύ των κόμβων για τη συλλογική εκτίμησή τους, σύμφωνα πάντα με το πρωτόκολλο συμφωνίας που χρησιμοποιείται. Αν και υπάρχουν εναλλακτικοί τρόποι εύρεσης αν έχει επιτευχθεί συμφωνία για ένα συγκεκριμένο αίτημα, η εκπομπή γεγονότος μπορεί να χρησιμοποιηθεί από εφαρμογές οι οποίες έχουν ανάγκη από λήψη των αποτελεσμάτων με την ελάχιστη δυνατή καθυστέρηση, όπως για παράδειγμα τραπεζικές εφαρμογές που διαχειρίζονται πληρωμές σε πραγματικό χρόνο.

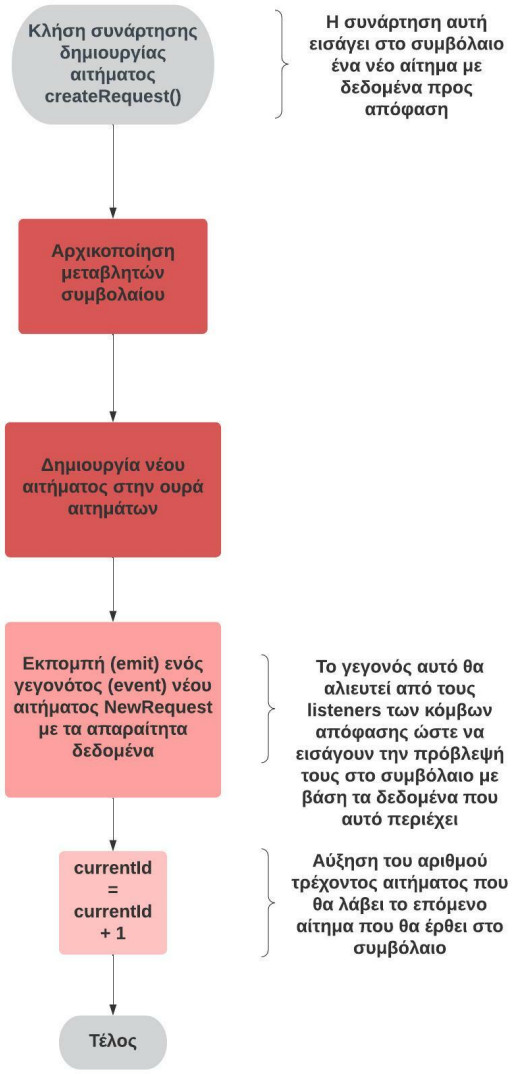
3.2.2.5. Συνολική άποψη δομών και λογικής συναρτήσεων έξυπνου συμβολαίου

Στο παρακάτω διάγραμμα παρουσιάζεται μία απεικόνιση των δομών που υπάρχουν στο έξυπνο συμβόλαιο και των συναρτήσεων που χρησιμοποιούνται για εγγραφή και ανάγνωση [Διάγραμμα 5].

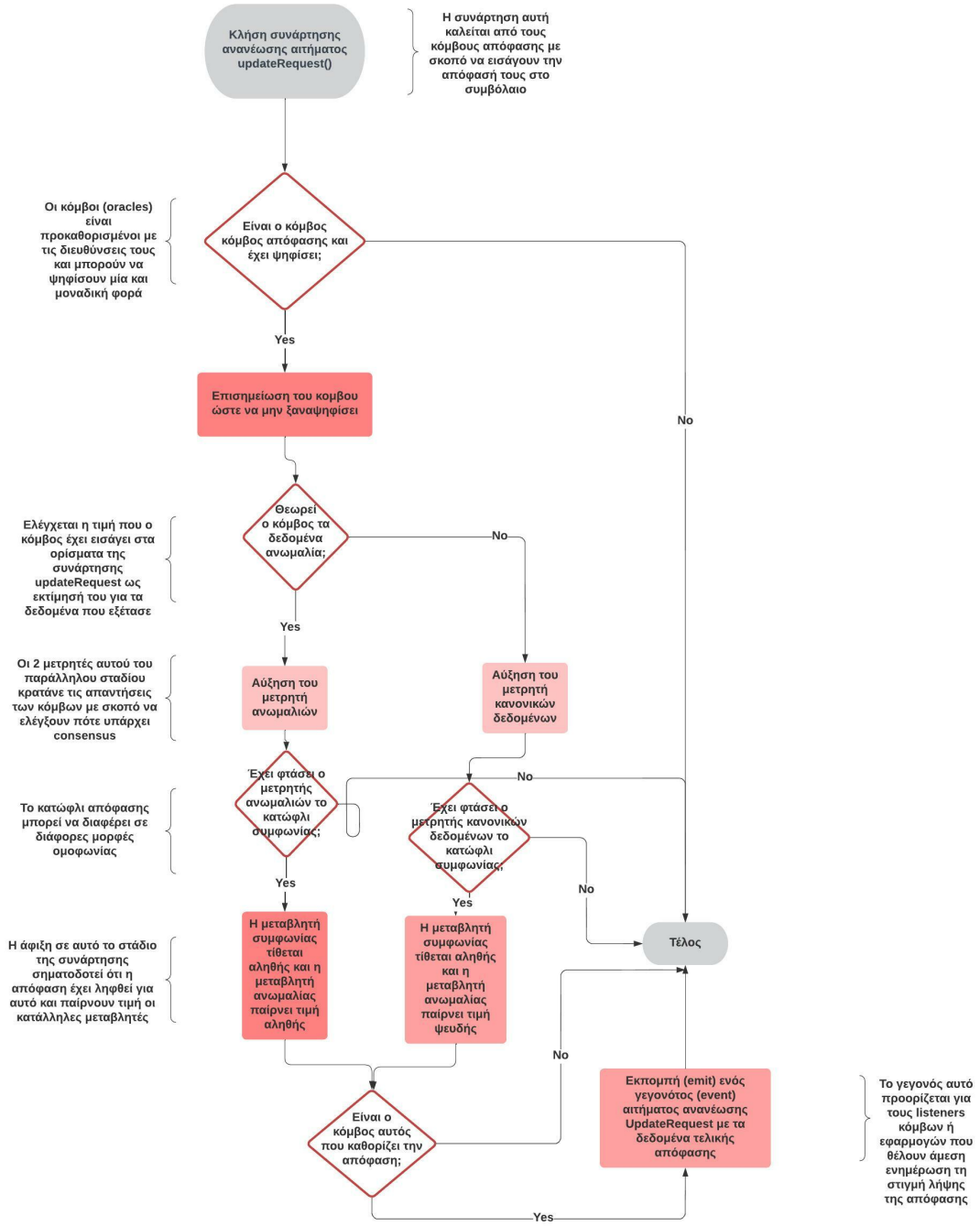


Διάγραμμα 5: Διάγραμμα UML Class δομών έξυπνου συμβολαίου

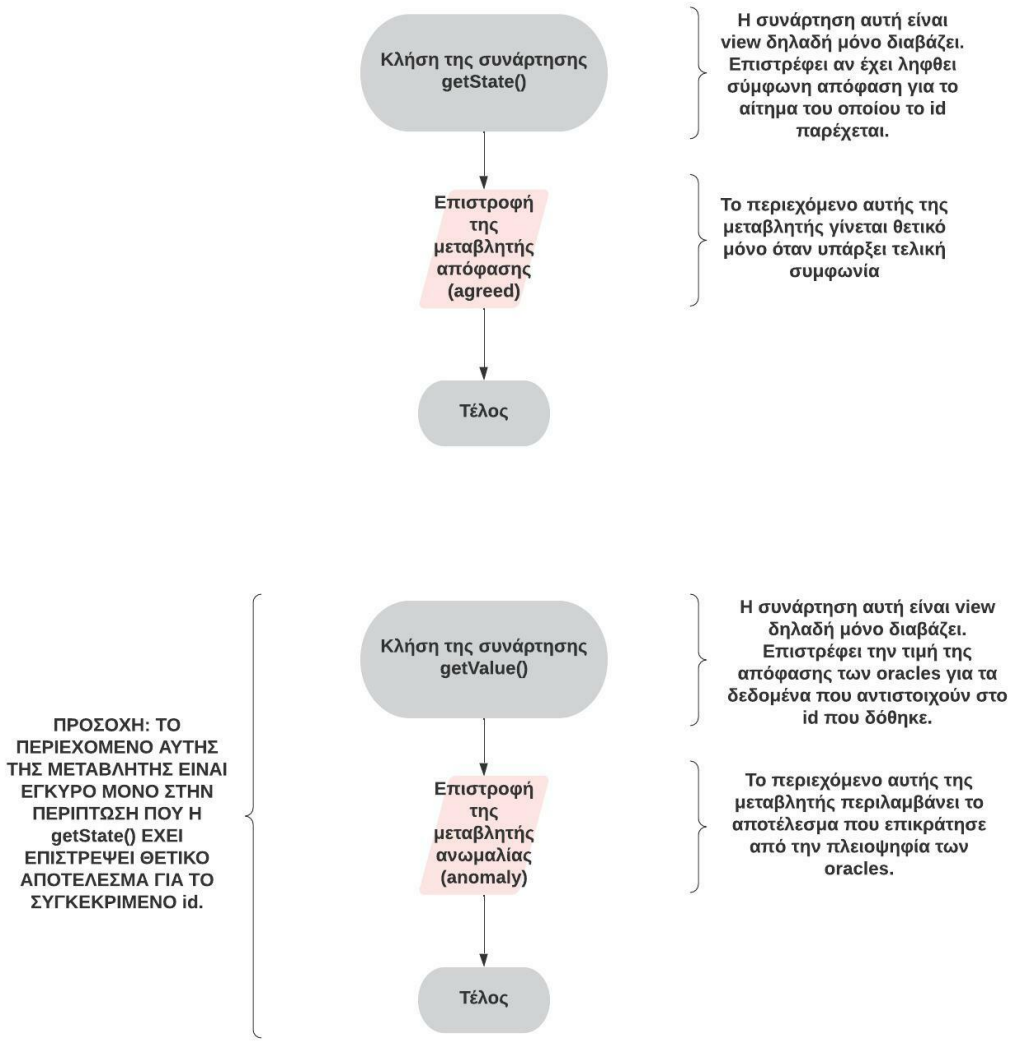
Επιπλέον, στα παρακάτω διαγράμματα [Διάγραμμα 6-8] παρουσιάζεται συνολικά η επιχειρησιακή λογική πάνω στην οποία πρόκειται να δομηθεί το έξυπνο συμβόλαιο και οι συναρτήσεις που θα χρησιμοποιούνται για να εγγράφουν νέα δεδομένα στην αλυσίδα, αλλά και να προσπελάσουν τα ήδη υπάρχοντα. Η επιχειρησιακή λογική αυτή αποτελεί μία απλή περιγραφή του αλγορίθμου, ο οποίος μετατρεπόμενος σε γλώσσα Solidity θα αποτελέσει το έξυπνο συμβόλαιο και τη βάση του πληροφοριακού συστήματος.



Διάγραμμα 6: Διάγραμμα UML Flow με την επιχειρησιακή λογική πίσω από τη συνάρτηση δημιουργία αιτήματος του έξυπνου συμβολαίου



Διάγραμμα 7: Διάγραμμα UML Flow με την επιχειρησιακή λογική πίσω από τη συνάρτηση ανανέωσης αιτήματος του έξυπνου συμβολαίου



Διάγραμμα 8: Διάγραμμα UML Flow με την επιχειρησιακή λογική πίσω από τις συναρτήσεις ανάγνωσης του έξυπνου συμβολαίου

3.2.3. Μοντέλα μηχανικής μάθησης

Τα μοντέλα μηχανικής μάθησης αποτελούν ένα βασικό συστατικό της εφαρμογής, αφού παρέχουν τις εκτιμήσεις και ουσιαστικά εισάγουν τον παράγοντα της έξυπνης απόφασης στο σύστημα. Ως μέρος του συστήματος τα μοντέλα φορτώνονται έτοιμα και προεκπαιδευμένα μέσω ενός .pkl αρχείου στον εκτιμητή που υπάρχει στο Python REST API Service κάθε κόμβου απόφασης. Το σύστημα παρέχει ουσιαστικά τεράστια ευελιξία επιτρέποντας τη φόρτωση οποιουδήποτε μοντέλου μπορεί να υλοποιήσει τη συνάρτηση predict().

Όπως αναφέρεται και παραπάνω, κατά τη λειτουργία του συστήματος φορτώνονται έτοιμα προεκπαιδευμένα μοντέλα. Η διαδικασία της εκπαίδευσης επιτελείται εκτός του συστήματος με μια διαδικασία που περιγράφεται παρακάτω. Τα μοντέλα κατασκευάζονται και εκπαιδεύονται μέσω ενός αυτοματοποιημένου script σε γλώσσα Python. Ως βάση τους έχουν έναν απλό εκτιμητή, ο τύπος του οποίου διευκρινίζεται από το εκάστοτε σενάριο που

ελέγχεται κάθε φορά. Κατά τη διαδικασία της εκπαίδευσης, καθένα από τα μοντέλα δέχεται ένα προκαθορισμένο αριθμό labeled δειγμάτων, τα οποία προέρχονται από ένα κοινό σύνολο δεδομένων αλλά είναι διαφορετικά για καθέναν από τα μοντέλα, με σκοπό να προσομοιώσουν ένα ρεαλιστικό σενάριο κατανεμημένα εκπαιδευμένων μοντέλων.

Κατά τη διαδικασία της εκπαίδευσης δεν χρησιμοποιούνται τεχνικές αναζήτησης του βέλτιστου εκτιμητή, όπως αναζήτηση βέλτιστων παραμέτρων, καθώς σκοπός της παρούσας εργασίας είναι να ελέγξει τη γενική συμπεριφορά μοντέλων εκπαιδευμένων με κατανεμημένο τρόπο όταν καλούνται να λάβουν συλλογικές αποφάσεις, όπως αυτές στα πλαίσια του παρόντος πληροφοριακού συστήματος. Για να υπάρχει η δυνατότητα αναπαραγωγής και επικύρωσης των αποτελεσμάτων της εργασίας, χρησιμοποιούνται συγκεκριμένα ορίσματα όπου τα μοντέλα αυτά δέχονται τυχαίες καταστάσεις (random states) με σκοπό την άρση της τυχαιότητας.

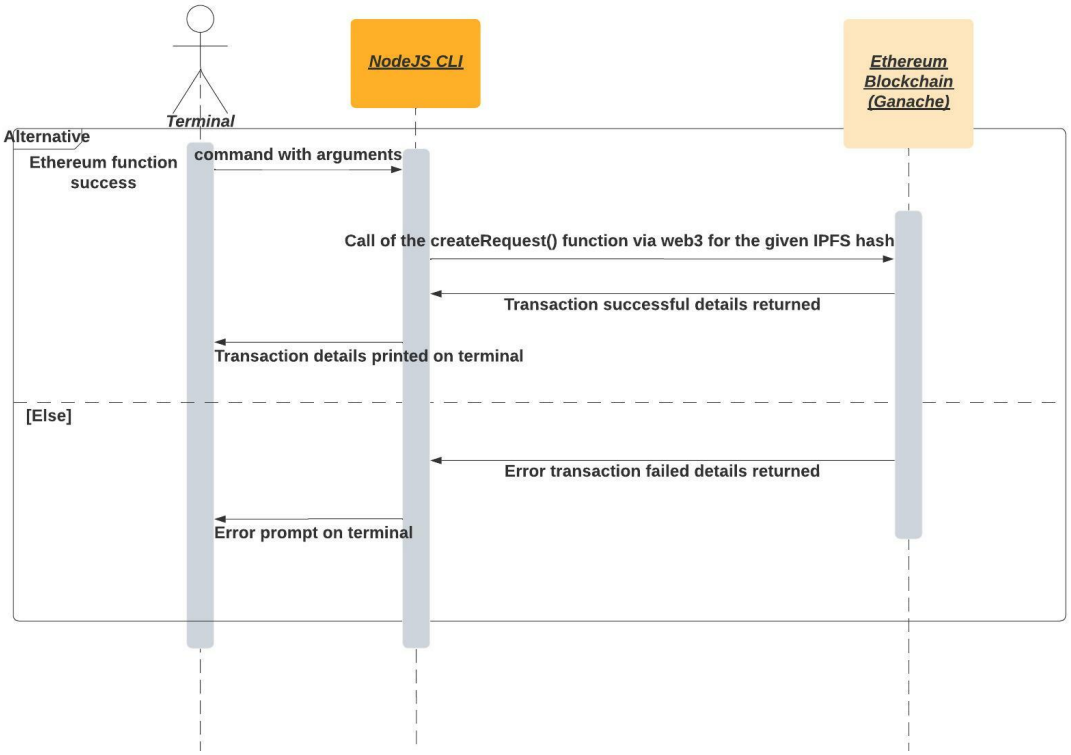
3.2.4. Η υπηρεσία διεπαφής τερματικού NodeJS CLI Client

Μία από τις κύριες διεπαφές χρήσης του συστήματος είναι η διεπαφή τερματικού NodeJS CLI Client, η οποία επιτρέπει σε απλούς χρήστες να αλληλοεπιδρούν με το σύστημα, υποβάλλοντας δεδομένα προς εκτίμηση και ελέγχοντας τα αποτελέσματα της συλλογικής εκτίμησης των κόμβων απόφασης σε δεδομένα που έχουν προηγουμένων υποβληθεί.

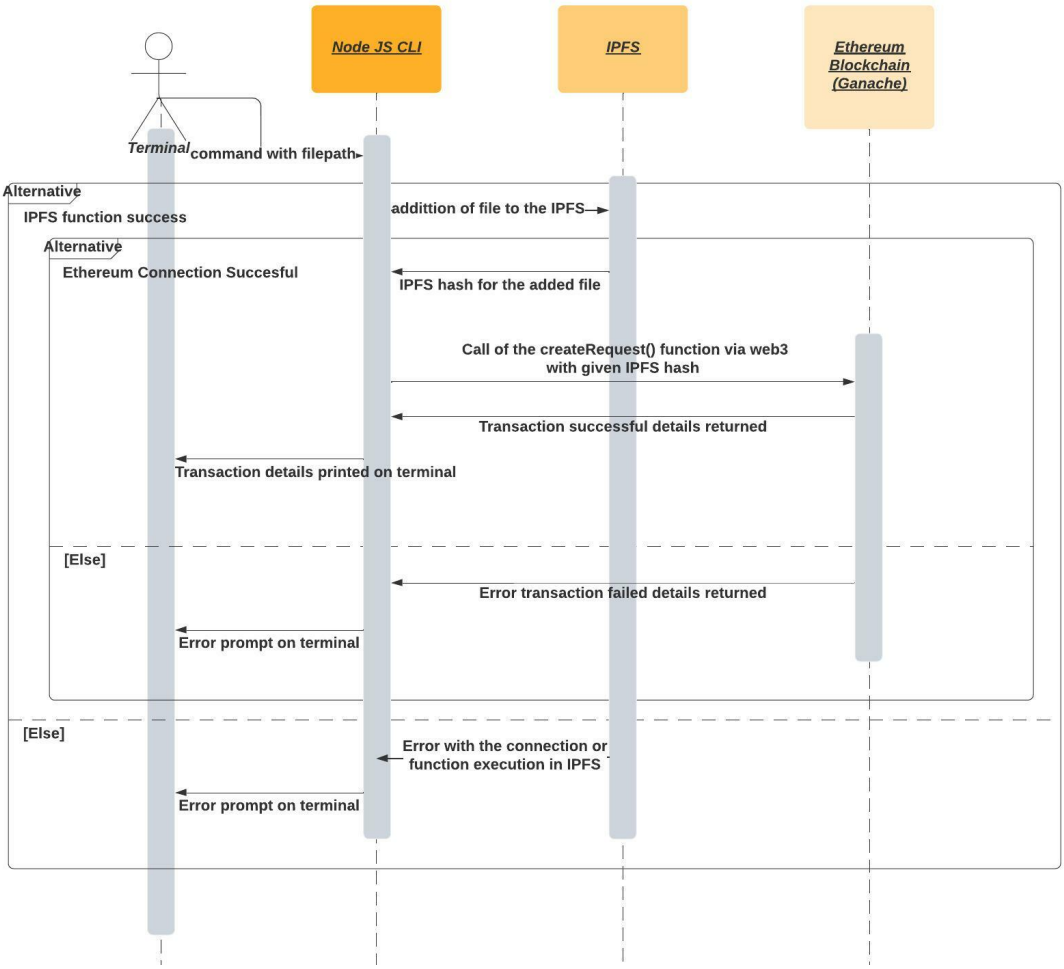
Η διεπαφή αυτή προορίζεται για χρήστες με πιο αυξημένες απαιτήσεις, όντας λιγότερο φιλική προς το μέσο χρήστη αλλά παρέχοντας αυξημένες δυνατότητες. Συγκεκριμένα, μέσω του αρχείου .config, δίνεται η δυνατότητα πλήρους παραμετροποίησης των διευθύνσεων του έξυπνου συμβολαίου, του πορτοφολιού του χρήστη και των παρόχων της επικοινωνίας με το blockchain και το IPFS. Επιπλέον, λόγω του γεγονότος πως η εφαρμογή εκτελείται μέσω τερματικού, παρέχεται η δυνατότητα εκτέλεσης πολλαπλών εντολών ταυτόχρονα, μέσω της αυτοματοποίησης με ένα bash script. Αυτή τη δυνατότητα εκμεταλλευτήκαμε κατά την εκτέλεση των δοκιμών, όπου τα σενάρια ελέγχου αποτυπώθηκαν σε έναν τέτοιο τύπο αρχείου.

3.2.4.1. Διαγράμματα απεικόνισης λειτουργιών της διεπαφής τερματικού

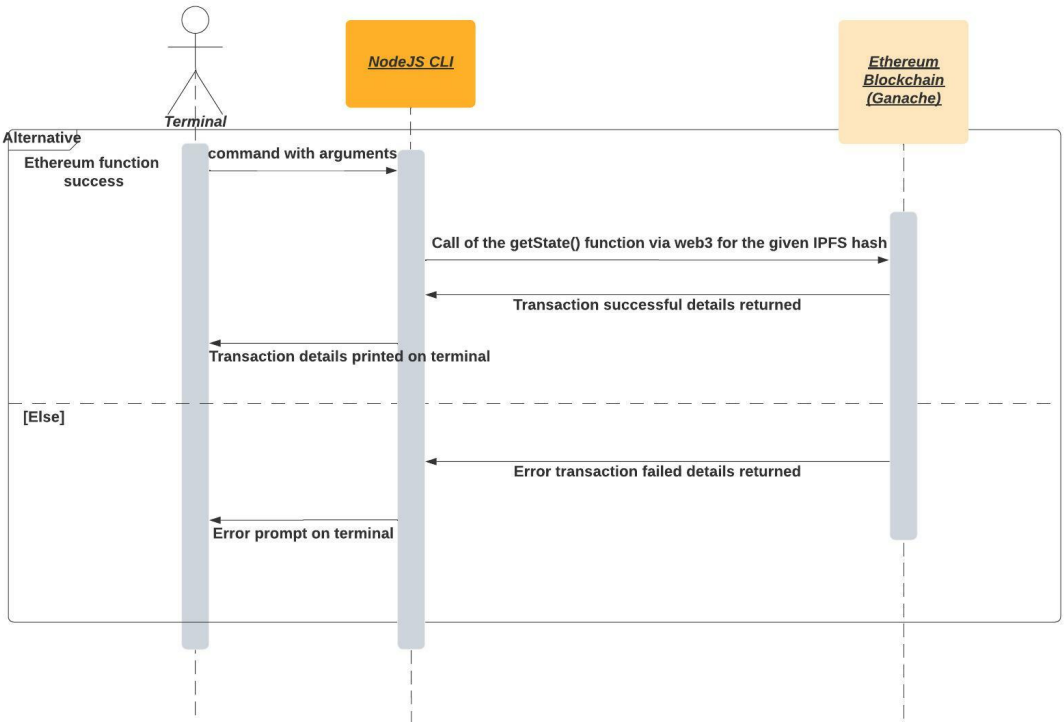
Στα παρακάτω διαγράμματα UML Sequence [Διάγραμμα 9-12] παρουσιάζεται η λογική πίσω από τις λειτουργίες που πρόκειται να παρέχει η συγκεκριμένη διεπαφή στους χρήστες. Απεικονίζονται οι αλληλεπιδράσεις του υποσυστήματος με άλλα υποσυστήματα του πληροφοριακού συστήματος κατά την εκτέλεση των λειτουργιών, η ροή της πληροφορίας, καθώς και οι προϋποθέσεις επιτυχούς εκτέλεσης των λειτουργιών που παρέχει η διεπαφή.



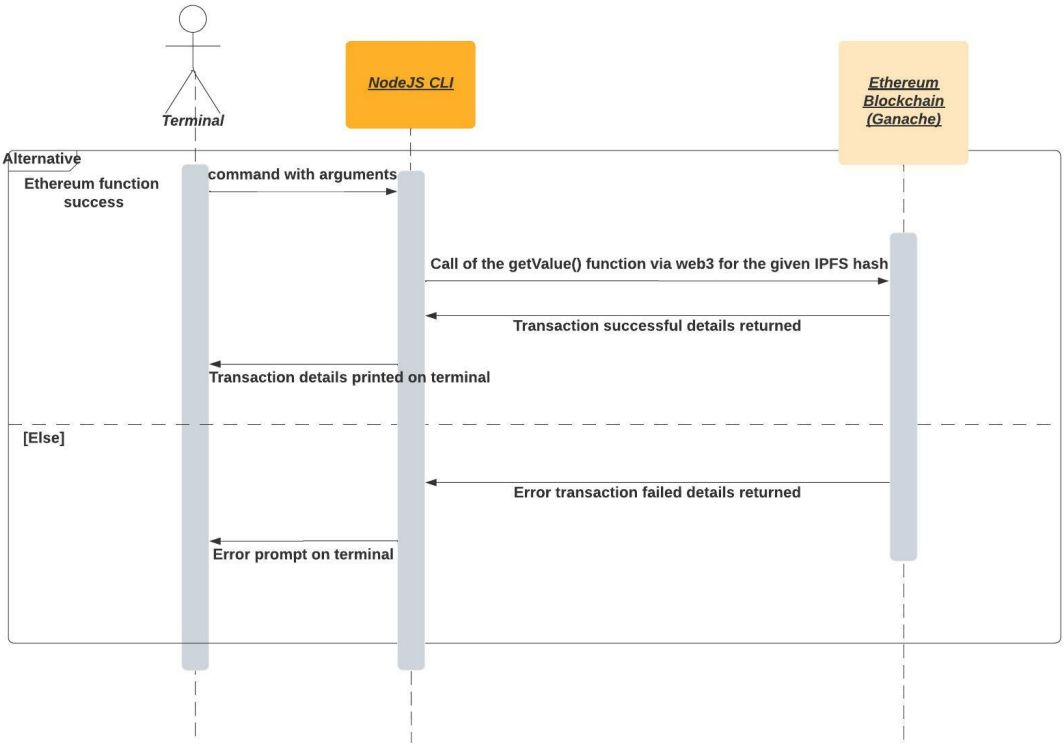
Διάγραμμα 9: Διάγραμμα UML Sequence δημιουργίας νέου αιτήματος με γνωστό IPFS hash



Διάγραμμα 10: Διάγραμμα UML Sequence δημιουργίας νέου αιτήματος με όρισμα τη θέση του αρχείου και ανέβασμα στο IPFS



Διάγραμμα 11: Διάγραμμα UML Sequence ανάγνωσης κατάστασης αιτήματος



Διάγραμμα 12: Διάγραμμα UML Sequence ανάγνωσης τιμής αιτήματος

3.2.5. Η υπηρεσία διεπαφής φυλλομετρητή ReactJS Frontend Client

Η διεπαφή των απλών χρηστών, οι οποίοι έχουν περιορισμένες απαιτήσεις από το πληροφοριακό σύστημα, συντελείται μέσω ενός εύχρηστου περιβάλλοντος διαδικτυακής εφαρμογής, αναπτυγμένο στο framework React.js της Javascript. Η διεπαφή αυτή, εκμεταλλεύεται το μεγάλο πλήθος βιβλιοθηκών της React.js, με σκοπό να προσφέρει στους χρήστες μία ολοκληρωμένη εμπειρία από τις υπηρεσίες του πληροφοριακού συστήματος, εξασφαλίζοντας παράλληλα τη λιγότερη δυνατή ανάγκη για προηγμένες γνώσεις από μέρους τους. Τους παρέχει διαδραστικά γραφικά, εύκολη σύνδεση με το προσωπικό τους πορτοφόλι και επικοινωνία με το blockchain ώστε να μπορούν να εκμεταλλευτούν στο έπακρο όλες τις δυνατότητες τους πληροφοριακού συστήματος. Παρακάτω παρουσιάζονται αναλυτικότερα τα κύρια χαρακτηριστικά της εφαρμογής και οι βιβλιοθήκες που χρησιμοποιούνται για την ανάπτυξή τους.

3.2.5.1. Σύνδεση με το προσωπικό πορτοφόλι του χρήστη μέσω ethers.js και MetaMask

Κύρια απαίτηση της αλληλεπίδρασης του χρήστη με τις υπηρεσίες της εφαρμογής, οι οποίες έχουν ως κορμό το blockchain, είναι η χρήση ενός μοναδικού προσδιοριστικού κατά την αλληλεπίδρασή του με το έξυπνο συμβόλαιο. Το μοναδικό προσδιοριστικό αυτό είναι η διεύθυνση του προσωπικού του πορτοφολιού, το οποίο μάλιστα χρησιμοποιεί ώστε

να «πληρώσει» το αντίτιμο που απαιτείται για συναλλαγές που δεν είναι δωρεάν, όπως αυτή της δημιουργίας νέου αιτήματος εκτίμησης.

Η διαχείριση της ταυτότητας και των συναλλαγών του χρήστη στα πλαίσια της διπλωματικής εργασίας παραχωρείται σε μία από τις πιο δημοφιλής εφαρμογές στο πεδίο, το MetaMask, η οποία αποτελεί plugin για την εφαρμογή browser. Ακολουθώντας μία απλή διαδικασία ο χρήστης μπορεί να εισάγει εκεί το λογαριασμό του για το δίκτυο της εφαρμογής, ώστε να είναι διαθέσιμο στις εφαρμογές που τρέχουν στον browser.

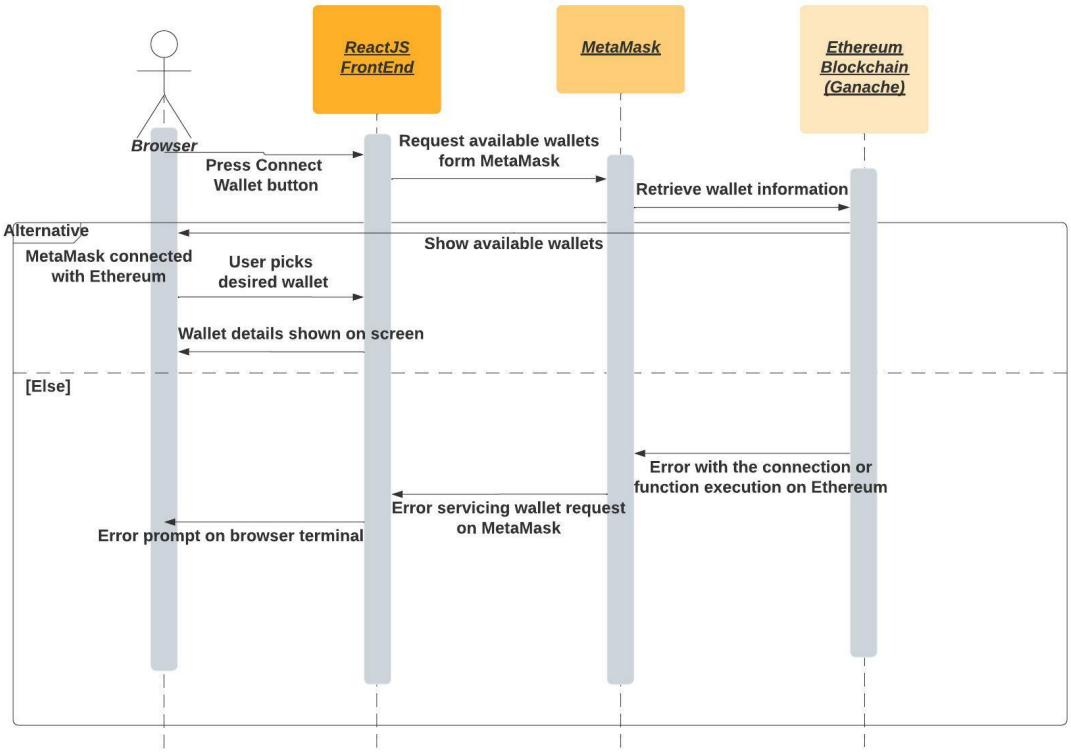
Για να επιτελέσει τη σύνδεση με το προσωπικό πορτοφόλι του χρήστη η εφαρμογή χρησιμοποιεί τη βιβλιοθήκη ethers.js. Με αυτή ανιχνεύει την ύπαρξη συνδεδεμένων λογαριασμών στον browser και επιτρέπει στο χρήστη εγκρίνοντας ένα αναδυόμενο παράθυρο του MetaMask να παραχωρήσει δικαιώματα στην εφαρμογή να χρησιμοποιεί το πορτοφόλι του. Οποιαδήποτε συναλλαγή απαιτεί έξοδα από το πορτοφόλι που έχει συνδέσει ο χρήστης, απαιτεί και επιπλέον έγκριση η οποία εμφανίζεται πάλι σε μορφή αναδυόμενου παραθύρου από το MetaMask.

3.2.5.2. Διαδραστικό περιβάλλον διεπαφής μέσω της MUI

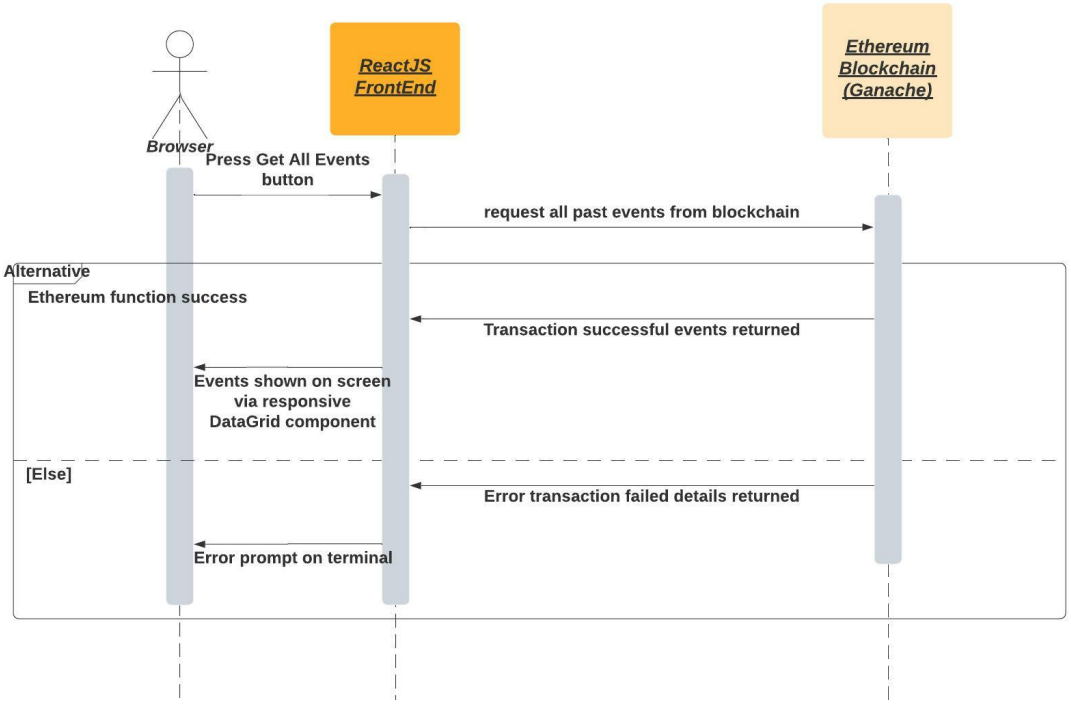
Ένας από τους κυριότερους παράγοντες που καθιστούν την εφαρμογή εύχρηστη και φιλική προς το μέσο χρήστη είναι η ίδια η διαδραστικότητα της React.js και της βιβλιοθήκης MUI. Η React.js δίνει τη δυνατότητα ανανέωσης του περιεχομένου που προβάλλεται στην οθόνη του χρήστη, καθώς οι πληροφορίες μεταβάλλονται, χωρίς να απαιτεί την ανανέωση του παραθύρου, επιτρέποντας έτσι μία ομαλή εμπειρία χρήσης. Την ίδια στιγμή, η βιβλιοθήκη MUI, παρέχει έτοιμα components με φιλικά προς το χρήστη χαρακτηριστικά, αλλά και προηγμένες δυνατότητες διαχείρισης δεδομένων, όπως η αναζήτηση και το φιλτράρισμά τους.

3.2.5.3. Διαγράμματα απεικόνισης λειτουργιών της διεπαφής φυλλομετρητή

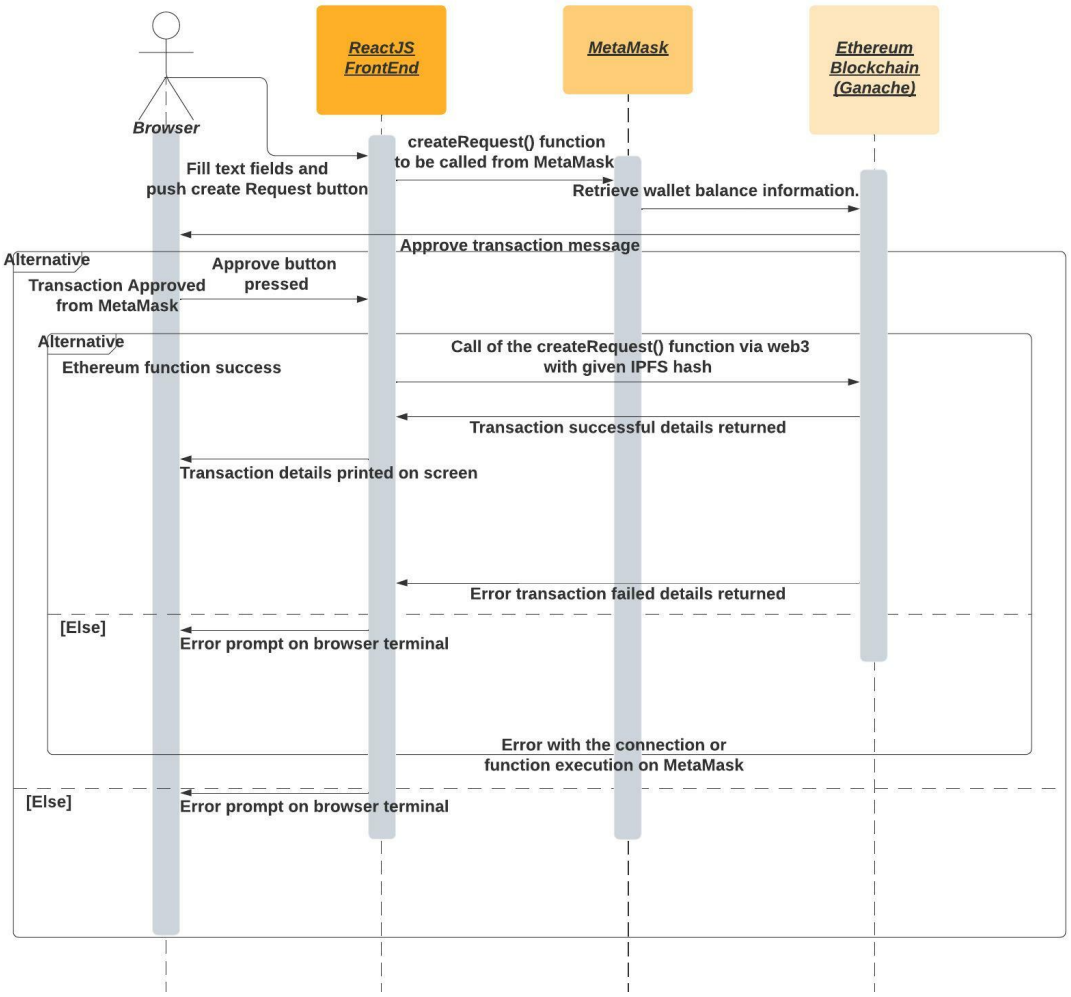
Στα παρακάτω διαγράμματα UML Sequence [Διάγραμμα 13-15] παρουσιάζεται η λογική πίσω από τις λειτουργίες που πρόκειται να παρέχει η συγκεκριμένη διεπαφή στους χρήστες. Απεικονίζονται οι αλληλεπιδράσεις του υποσυστήματος με άλλα υποσυστήματα του πληροφοριακού συστήματος κατά την εκτέλεση των λειτουργιών, η ροή της πληροφορίας, καθώς και οι προϋποθέσεις επιτυχούς εκτέλεσης των λειτουργιών που παρέχει η διεπαφή.



Διάγραμμα 13: Διάγραμμα UML Sequence σύνδεσης πορτοφολιού μέσω MetaMask



Διάγραμμα 14: Διάγραμμα UML Sequence προβολής γεγονότων μέσω DataGrid



Διάγραμμα 15: Διάγραμμα UML Sequence δημιουργίας αιτήματος

3.2.6. Η υπηρεσία διασύνδεσης NodeJS Oracle Service

Ένα βασικό κομμάτι του συστήματος το οποίο επιτελεί τη σύνδεση των δύο βασικών τεχνολογιών του blockchain και της μηχανικής μάθησης είναι η υπηρεσία oracle που έχει υλοποιηθεί σε NodeJS. Η υπηρεσία αυτή βρίσκεται εγκατεστημένη σε όλους τους κόμβους, οι οποίοι έχουν επωμιστεί την ευθύνη της παροχής στο blockchain εκτιμήσεων, μέσω των μοντέλων μηχανικής μάθησης που έχουν στη διάθεσή τους και τα αποτελέσματα των οποίων παρέχουν μέσω ενός REST API.

Οι βασικές αρμοδιότητες της υπηρεσίας αυτής είναι οι παρακάτω:

- Παρακολούθηση του blockchain για γεγονότα NewRequest
- Υποβολή αιτημάτων προς εκτίμηση στα REST API των μοντέλων μηχανικής μάθησης
- Παροχή των αποτελεσμάτων των εκτιμήσεων πίσω στο blockchain

3.2.6.1. Παρακολούθηση του blockchain για γεγονότα NewRequest

Η ανάγκη για παροχή δεδομένων τα οποία βρίσκονται εκτός-αλυσίδας προς το blockchain γνωστοποιείται από το ίδιο μέσω της εκπομπής ενός γεγονότος NewRequest. Η χρήση γεγονότων επιτρέπει σε υπηρεσίες, όπως το NodeJS Oracle Service, οι οποίες παρακολουθούν το blockchain, να ενημερωθούν για την ανάγκη που υπάρχει να προσκομίσουν εξωτερικά δεδομένα εντός-αλυσίδας με σκοπό να ληφθεί μία απόφαση. Το αίτημα αυτό περιλαμβάνει τις βασικές πληροφορίες που απαιτεί το εξωτερικό σύστημα, ώστε να προσπελάσει τα δεδομένα και να εκτελέσει την εκτίμησή του.

3.2.6.2. Υποβολή αιτημάτων προς εκτίμηση στα REST API των μοντέλων μηχανικής μάθησης

Μετά την αναγνώριση ενός γεγονότος NewRequest, η υπηρεσία NodeJS Oracle Service, έχει την ευθύνη της μεταβίβασης του αιτήματος στην τοπική υπηρεσία REST API του μοντέλου μηχανικής μάθησης. Η υπηρεσία αυτή γνωρίζει μέσω των ορισμάτων που τις παρέχονται κατά την εκκίνησή της το κατάλληλο endpoint, στο οποίο πρέπει να υποβάλει τα αιτήματά της και το οποίο μπορεί να είναι εσωτερικό ή εξωτερικό του συστήματος. Η υπηρεσία μετασχηματίζει τα περιεχόμενα του γεγονότος NewRequest, το οποίο έχει αναγνωρίσει, σε ένα POST αίτημα με τα κατάλληλα περιεχόμενα και το διαβιβάζει προς εξυπηρέτηση στην υπηρεσία REST API του μοντέλου μηχανικής μάθησης. Έπειτα αναμένει την απάντησή του, ώστε να ενημερώσει καταλλήλως το blockchain.

3.2.6.3. Παροχή αποτελεσμάτων των εκτιμήσεων πίσω στο blockchain

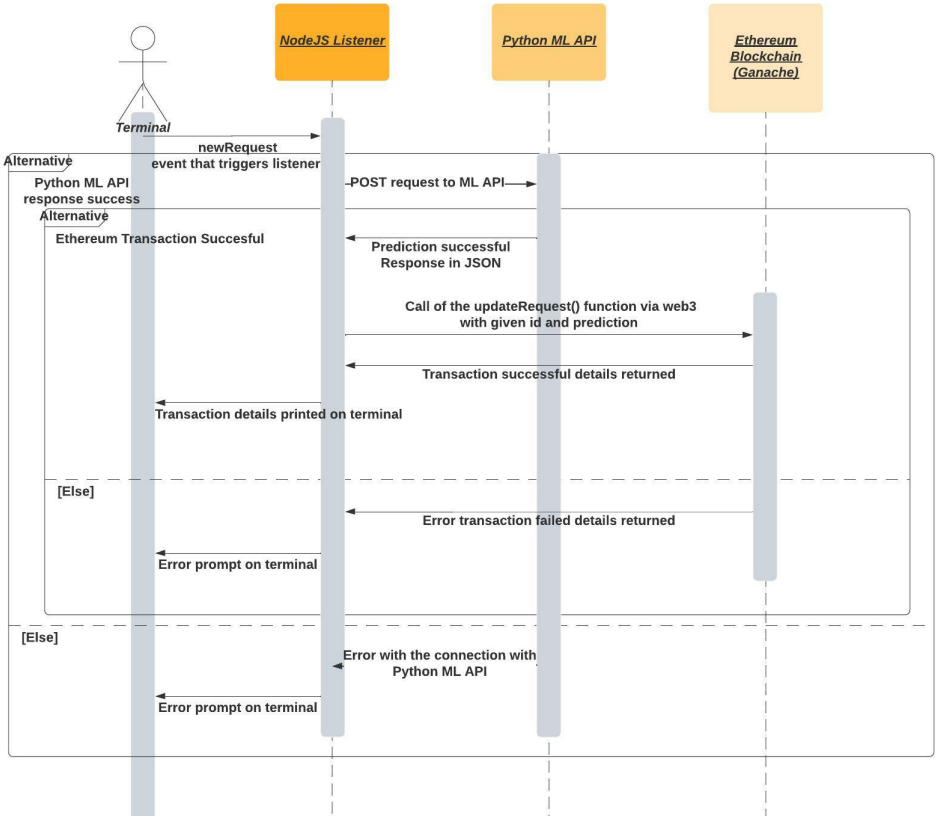
Μετά της απάντησης από την υπηρεσία REST API του μοντέλου μηχανικής μάθησης, η υπηρεσία NodeJS Oracle Service έχει την ευθύνη του ελέγχου του αποτελέσματος αυτού και της διαβίβασής του στο blockchain, εφόσον αυτό είναι έγκυρο. Σε πρώτη φάση λοιπόν, η υπηρεσία ελέγχει αν η απάντηση στο αίτημα έχει κωδικό έγκυρης απάντησης ή κωδικό σφάλματος. Σε περίπτωση σφάλματος η υπηρεσία δεν επαναλαμβάνει το αίτημα, ούτε ενημερώνει το blockchain για την έκβαση. Η αντιμετώπιση τέτοιων καταστάσεων συντελείται μέσω της χαλαρότητας του πρωτοκόλλου consensus του έξυπνου συμβολαίου, το οποίο επιτρέπει την επίτευξη συμφωνία ακόμα και με την απουσία ή την αντίθετη άποψη ενός μικρού αριθμού κόμβων απόφασης.

Σε περίπτωση λήψης μίας έγκυρης απάντησης, η υπηρεσία ελέγχει μόνο πως τα πεδία της απάντησης αυτής είναι του τύπου που αναμένεται από το blockchain. Σε θετική περίπτωση, προχωρά στην ενημέρωση του blockchain για τα εξωτερικά δεδομένα, μέσω της κλήσης της συνάρτησης updateRequest() που παρέχεται από το έξυπνο συμβόλαιο για αυτή τη χρήση.

3.2.6.4. Διάγραμμα απεικόνισης λειτουργίας της υπηρεσίας NodeJS Oracle Service

Όπως περιγράφηκε και στις ανωτέρω ενότητες, η υπηρεσία NodeJS Oracle Service, λειτουργεί αδιάκοπα στους κόμβους απόφασης παρακολουθώντας το blockchain για νέα

γεγονότα και επικοινωνώντας με τα συστήματα απόφασης των κόμβων, όταν αυτό είναι αναγκαίο. Στην παρακάτω εικόνα [Διάγραμμα 16] απεικονίζεται το UML Sequence διάγραμμα της υπηρεσίας αυτής, στο οποίο αποτυπώνονται οι αλληλεπιδράσεις του υποσυστήματος με άλλα υποσυστήματα του πληροφοριακού συστήματος κατά την εκτέλεση της λειτουργίας του, η ροή της πληροφορίας, καθώς και οι προϋποθέσεις επιτυχούς εκτέλεσης της λειτουργίας που παρέχει η υπηρεσία.



Διάγραμμα 16: Διάγραμμα UML Sequence της λειτουργίας του υποσυστήματος NodeJS Oracle Service

3.2.7. Το υποσύστημα παροχής εκτιμήσεων Python Machine Learning API

Ένα από τα βασικά χαρακτηριστικά του συστήματος είναι το γεγονός εμπλουτίζει τις αποφάσεις που λαμβάνονται στο blockchain με βάση τα συμπεράσματα που εξάγουν τα κατανεμημένα μοντέλα μηχανικής μάθησης από τα δεδομένα. Τα μοντέλα αυτά λειτουργούν ανεξάρτητα από το υπόλοιπο σύστημα και παρέχουν τις εκτιμήσεις τους μόνο όποτε τους ζητείται.

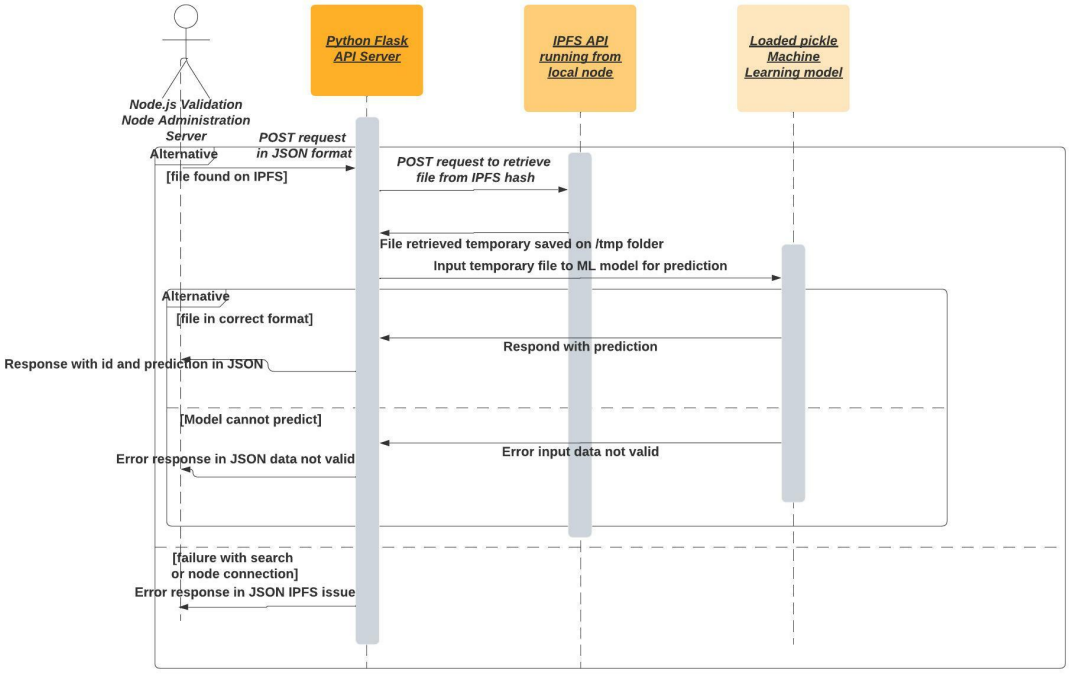
Για να επιτευχθεί αυτό, με τρόπο που εξασφαλίζει την ελάχιστη δυνατή έκθεση των μοντέλων στο υπόλοιπο σύστημα, αλλά και την ανεξάρτητη με τα μοντέλα λειτουργία του συστήματος, το τμήμα αυτό έχει υλοποιηθεί με την μορφή μίας REST API υπηρεσίας, η οποία στα πλαίσια της διπλωματικής εργασίας λειτουργεί τοπικά. Παρ ’όλα αυτά η αρχιτεκτονική που χρησιμοποιείται, επιτρέπει την λειτουργία αυτού του συστήματος και σε ένα εξωτερικό περιβάλλον, για παράδειγμα σε ένα περιβάλλον cloud, το οποίο θα μπορούσε

να του εξασφαλίσει μεγαλύτερη υπολογιστική ισχύ, σε πιθανόν πιο απαιτητικά προβλήματα. Η μόνες προϋπόθεση που υπάρχει για τη λειτουργία του συστήματος είναι να επιτρέπεται στο NodeJS Oracle Service να επικοινωνεί με το υποσύστημα αυτό, ώστε να κάνει αιτήματα και να διαβιβάζει τα αποτελέσματα στο blockchain, καθώς και η ανάγκη για επικοινωνία με έναν κόμβο IPFS, ώστε να αντλούνται από εκεί τα δεδομένα μεγάλου όγκου.

Οι λειτουργίες που επιτελεί το τμήμα αυτό του πληροφοριακού συστήματος παρατίθενται παρακάτω.

1. Φόρτωση και χρήση μοντέλου μηχανικής μάθησης: Το υποσύστημα αυτό έχει ως βάση του ένα μοντέλο μηχανικής μάθησης, το οποίο πραγματοποιεί τις εκτιμήσεις πάνω στα δεδομένα που έχει αιτηθεί το NodeJS Oracle Service. Το μοντέλο φορτώνεται κατά την εκκίνηση του υποσυστήματος από ένα αρχείο .pkl και πρέπει να είναι τέτοιας μορφής, ώστε να μπορεί να υλοποιεί τη συνάρτηση .predict( ).
2. Λειτουργία ενός REST API Server: Ολόκληρη η επικοινωνία του υποσυστήματος με το NodeJS Oracle Service, συντελείται μέσω μίας REST API υπηρεσίας. Το υποσύστημα μέσω του πακέτου Flask της Python εκθέτει συγκεκριμένα endpoints, με σκοπό να παρέχει τη δυνατότητα σε εξωτερικά συστήματα να στέλνουν αιτήματα προς εξυπηρέτηση. Τα αιτήματα αυτά περιορίζονται σε αιτήματα εκτίμησης, πάνω σε δεδομένα τα οποία περιέχει το σώμα του αιτήματος. Με τον τρόπο αυτό το μοντέλο συντελεί στην απόφαση που λαμβάνεται στο blockchain, χωρίς όμως να εκθέτει οποιαδήποτε πληροφορία για τον τρόπο με τον οποίο λαμβάνει την εκτίμησή του.
3. Επικοινωνία με το IPFS και ανάκτηση δεδομένων: Μία από τις βασικές αρχές της προτεινόμενης αρχιτεκτονικής είναι η αποφυγή μεταφοράς μεγάλου όγκου δεδομένων μεταξύ υποσυστημάτων. Η ανάκτηση των δεδομένων μεγάλου όγκου συντελείται στο σημείο εκείνο όπου υπάρχει πραγματικά η ανάγκη για χρήση τους, έτσι ώστε να περιορίζεται η επιβάρυνση του συστήματος στην απολύτως αναγκαία. Δεδομένου ότι το υποσύστημα αυτό είναι εκείνο που εκτελεί την πρόβλεψη πάνω στα δεδομένα, πρέπει να του παρέχεται σύνδεση με έναν κόμβο IPFS, ώστε να μπορεί να εκτελέσει αιτήματα ανάκτησης δεδομένων. Η ανάκτηση των δεδομένων είναι προσωρινή και τα δεδομένα διαγράφονται άμεσα μετά την εκτέλεση της πρόβλεψης.

Στην παρακάτω εικόνα [Διάγραμμα 17] φαίνεται μέσω ενός διαγράμματος UML Sequence, ο τρόπος που ανταποκρίνεται το υποσύστημα με σκοπό να εξυπηρετήσει ένα νέο αίτημα για εκτίμηση που καταφθάνει σε αυτό. Απεικονίζονται η αλληλεπίδρασή του με τρίτα συστήματα, η ροή των πληροφοριών και οι προϋποθέσεις επιτυχούς εκτέλεσης της διαδικασίας πρόβλεψης.



Διάγραμμα 17: Διάγραμμα UML Sequence της εξυπηρέτησης αιτήματος για εκτίμηση από την υπηρεσία εκτίμησης του κόμβου απόφασης

3.2.8. IPFS

Ένας από τους κυριότερους στόχους της αρχιτεκτονικής που προτείνεται είναι να αποφορτίσουμε το κομμάτι του blockchain από τον κύριο όγκο των δεδομένων, διατηρώντας παράλληλα έναν κατανεμημένο τρόπο λειτουργίας. Για το λόγο αυτό, ο κύριο όγκος των δεδομένων αποθηκεύεται στο IPFS, το οποίο όπως αναλύθηκε προηγουμένων αποτελεί έναν απολύτως κατανεμημένο τρόπο αποθήκευσης δεδομένων μεγάλου όγκου. Τα δεδομένα που αποθηκεύονται στο IPFS, συσχετίζονται με τις συναλλαγές που πραγματοποιούνται στο blockchain μέσω του ipfs hash, το οποίο τα προσδιορίζει μοναδικά. Με τον τρόπο αυτό ανά πάσα στιγμή μπορεί οποιοσδήποτε κατέχει τον κωδικό αυτό κατακερματισμούυ να τα αντλήσει, ενώ την ίδια στιγμή είναι πρακτικά αδύνατον για κάποιον να τα προσπελάσει χωρίς να κατέχει αυτό το hash.

3.2.9. Το σύνολο δεδομένων MNIST

Βασικό συστατικό της παρούσας διπλωματικής εργασίας αποτελεί το σύνολο δεδομένων που χρησιμοποιήθηκε με σκοπό να προσομοιώσει μία απλή, αλλά ρεαλιστική περίπτωση χρήσης της προτεινόμενης αρχιτεκτονικής. Το σύνολο δεδομένων που χρησιμοποιήθηκε είναι το «MNIST handwritten digits dataset». Το σύνολο αυτό αποτελείται από ένα σύνολο εκπαίδευσης 60.000 δειγμάτων και ένα σύνολο ελέγχου 10.000 δειγμάτων. Τα δείγματα είναι εικόνες που απεικονίζουν σαρωμένες χειρόγραφες απεικονίσει ψηφίων από το 0 έως το 9.

Αν και το συγκεκριμένο dataset αποτελείται από εικόνες, θέλαμε να κάνουμε το πρόβλημα ανεξάρτητο από το πεδίο της όρασης υπολογιστών, για αυτό και επιτελέσαμε

πάνω στο σύνολο του συνόλου δεδομένων PCA, κρατώντας τις 100 κύριες συνιστώσες. Η μείωση της διαστατικότητας αποτελεί μία τεχνική η οποία χρησιμοποιείται συχνά σε σύνολα δεδομένων με ευαίσθητα δεδομένα, έχοντας ως σκοπό να διατηρήσει μεγάλο κομμάτι της αξίας της πληροφορίας για τα μοντέλα, εξαλείφοντας όμως τον κίνδυνο να εκτεθούν οι raw πληροφορίες του χρήστη. Έτσι το τελικό σύνολο δεδομένων παρά την αρχική του φύση παρουσίαζε μεγάλη ομοιότητα με σύνολα δεδομένων στα οποία εκτελείται αναγνώριση απάτης.

Κεφάλαιο 4

Παρουσίαση υλοποίησης και λειτουργικότητας εφαρμογής

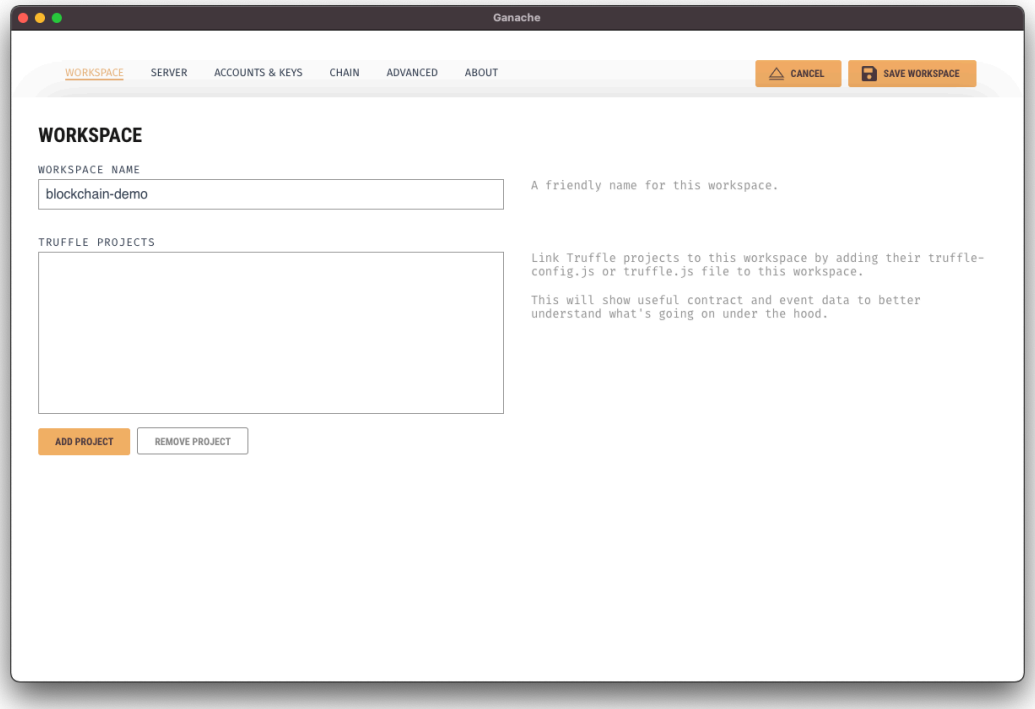
Σε αυτό το τμήμα της εργασίας θα πραγματοποιηθεί μία βήμα προς βήμα παρουσίαση της εγκατάστασης και της λειτουργίας του πληροφοριακού συστήματος, εστιάζοντας και επεξηγώντας παράλληλα σημεία του κώδικα ανάπτυξης που είναι καίριας σημασίας.

Το σύνολο του κώδικα που αναπτύχθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας μαζί με οδηγίες για την εγκατάσταση και εκτέλεση των υποσυστημάτων μπορεί να βρεθεί στο ακόλουθο αποθετήριο στο GitHub [47]:
https://github.com/pskoufis13/diploma-thesis-ntua

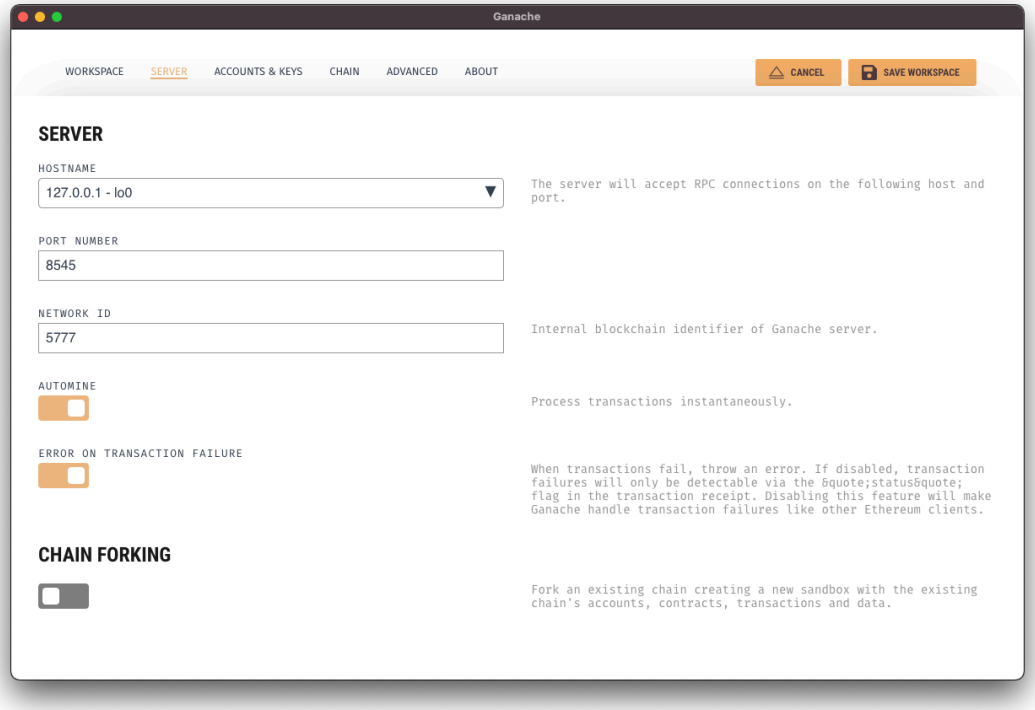
4.1. Δημιουργία δικτύου blockchain (Ethereum – Ganache)

Πρωταρχικό βήμα της δημιουργίας της εφαρμογής είναι η δημιουργία και εκκίνηση ενός δικτύου Ethereum, πάνω στο οποίο θα λειτουργεί το προτεινόμενο σύστημα. Η δημιουργία ενός τέτοιου δικτύου στα πλαίσια της ανάπτυξης και του ελέγχου του συστήματος θα γίνει μέσω της εφαρμογής Ganache.

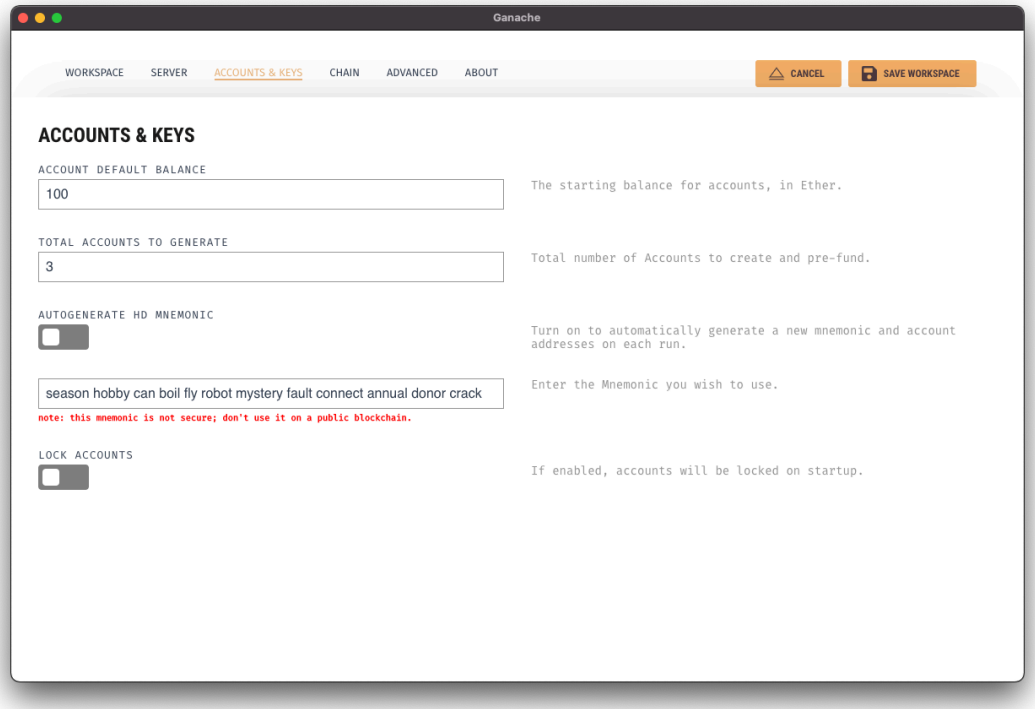
Η διαδικασία δημιουργίας ενός νέου περιβάλλοντος εργασίας παρουσιάζεται στις παρακάτω εικόνες. Επιλέγουμε το περιβάλλον εργασίας να τρέχει στην πόρτα «*8545*» του τοπικού δικτύου «*127.0.0.1*» και να δημιουργήσουμε τρεις λογαριασμούς. Η δημιουργία του μηνμονικού γίνεται χειροκίνητα με σκοπό να μπορεί να αναπαραχθεί το δίκτυο και οι διευθύνσεις των λογαριασμών αν χρειαστεί. Τέλος, σημειώνεται πως είναι δυνατή η προσθήκη νέων λογαριασμών αν αυτό απαιτηθεί μετά την εκκίνηση του περιβάλλοντος.



Εικόνα 2: Κατασκευή νέου περιβάλλοντος με όνομα "blockchain-demo"

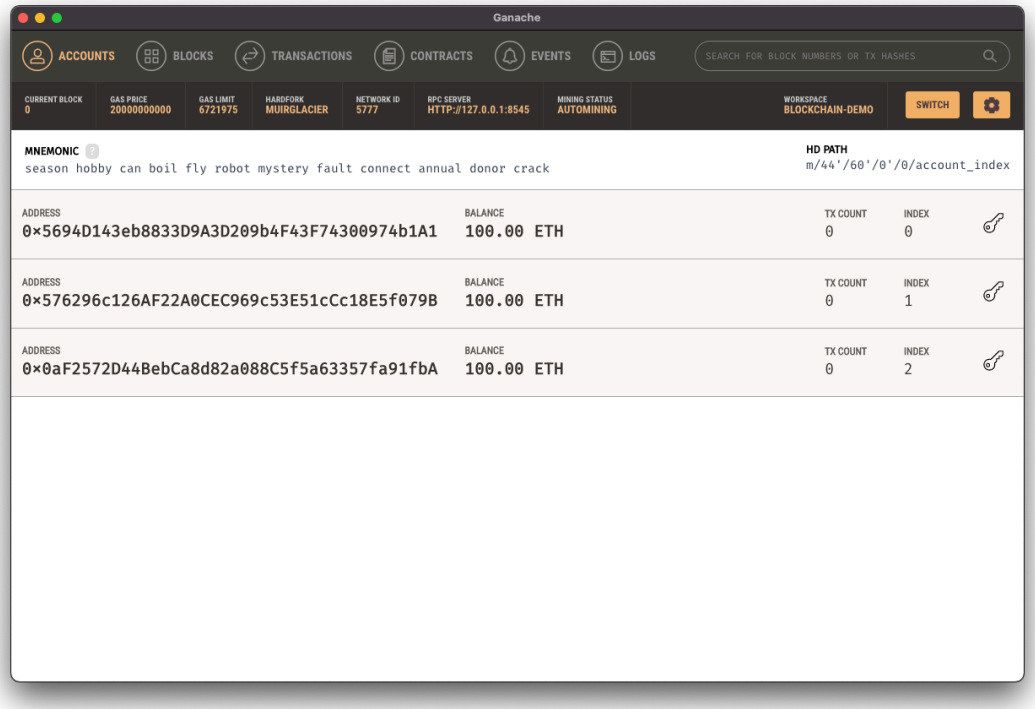


Εικόνα 3: Ορισμός παραμέτρων δικτύου - hostname και αριθμού πόρτας

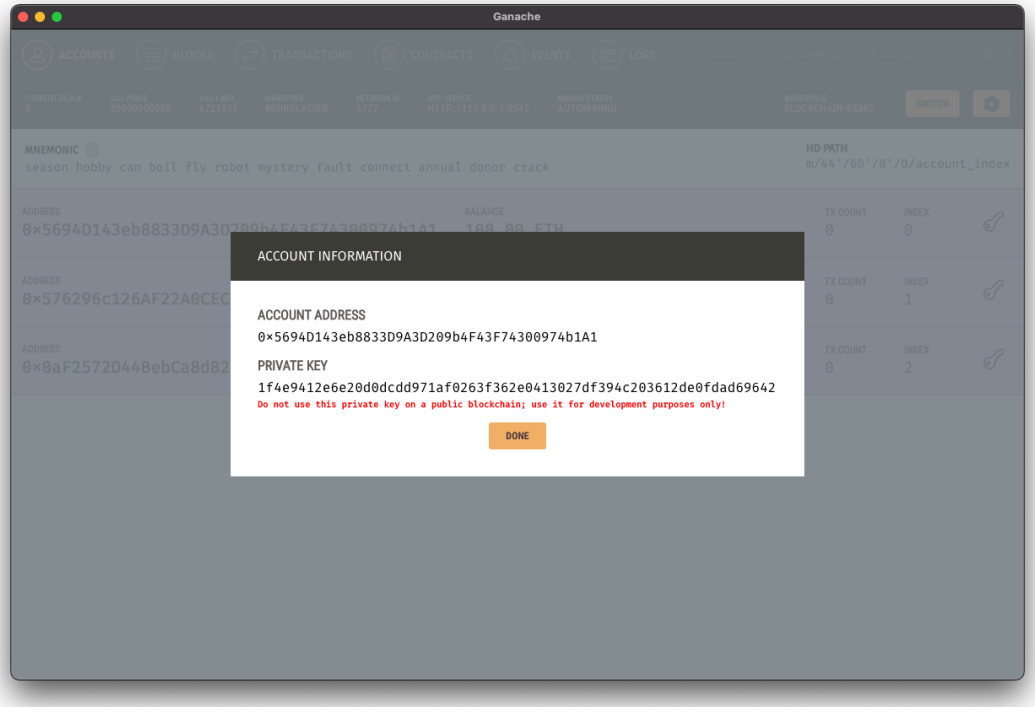


Εικόνα 4: Ορισμός πλήθους λογαριασμών και μνημονικού

Κατά την ολοκλήρωση του ορισμού των πεδίων, πατώντας το πλήκτρο *"Save Workspace"* δημιουργείται το νέο περιβάλλον ανάπτυξης blockchain. Στο επόμενο στιγμιότυπο φαίνονται οι λογαριασμοί που δημιουργήθηκαν με τα χαρακτηριστικά τους (διεύθυνση - δημόσιο κλειδί). Επιπλέον, από τη διεπαφή αυτή μπορούμε να προσπελάσουμε και τα ιδιωτικά χαρακτηριστικά κάθε λογαριασμού, δηλαδή το ιδιωτικό κλειδί τους [Εικόνα 5 – 6].



Εικόνα 5: Κεντρική όψη επισκόπησης λογαριασμών και των δημόσιων χαρακτηριστικών τους



Εικόνα 6: Όψη εμφάνισης ιδιωτικού κλειδιού κάθε λογαριασμού

4.2. Ανάπτυξη έξυπνου συμβολαίου, μεταγλώττιση, deployment και έλεγχος

Για το τμήμα της ανάπτυξης του έξυπνου συμβολαίου, της μεταγλώττισής του deployment του στο τοπικό δίκτυο και της παρακολούθησης των συναλλαγών θα χρησιμοποιήσουμε τη διαδικτυακή εφαρμογή IDE “Remix IDE”.

4.2.1. Ανάπτυξη του έξυπνου συμβολαίου

Το έξυπνο συμβόλαιο θα αναπτυχθεί στη γλώσσα Solidity μέσα από τον editor της εφαρμογής. Παρακάτω παρουσιάζονται και επεξηγούνται τα κύρια τμήματα του έξυπνου συμβολαίου και ο αντίστοιχος κώδικας.

4.2.1.1. Η δομή του αιτήματος

Το κύριο κομμάτι του έξυπνου συμβολαίου είναι το αίτημα (Request). Ένα αίτημα αντικατοπτρίζει ουσιαστικά ένα δείγμα δεδομένων ενός προβλήματος, για το οποίο ένας κόμβος έχει ζητήσει μία εκτίμηση από το δίκτυο για το αν τα δεδομένα αυτά αποτελούν ανωμαλία ή όχι. Αυτό στα πλαίσια του έξυπνου συμβολαίου αναπαρίσταται με μία δομή (struct) [Εικόνα 7].

```
//defines request contents
struct Request {
  uint id;                          //request id
  string taskToSolve;               //the problem for which the request applies
  string ipfsHash;                  //the IPFS hash to retrieve the data to decide on
  bool agreed;                      //a boolean signaling whether orbits have reached consensus
  bool anomaly;                     //a boolean containing the decision of the oracles
  uint normalCounter;               //a counter for the decision of oracles indicating normal data
  uint anomalyCounter;              //a counter for the decision of oracles indicating outlier data
  mapping(address => uint) quorum;  //oracles which will query the answer (1=oracle hasn't voted, 2=oracle has voted)
}
```

Εικόνα 7: Ορισμός της δομής Request του έξυπνου συμβολαίου

Ο τύπος και το περιεχόμενο των μεταβλητών της δομής επεξηγούνται αναλυτικά στον πίνακα που ακολουθεί [Πίνακας].

uint id	Το μοναδικό προσδιοριστικό κάθε αιτήματος. Αυξάνεται αυτόματα για κάθε νέο αίτημα και αποτελεί ουσιαστικά ένα κύριο κλειδί ανάκτησης του αιτήματος.
string taskToSolve	Το πεδίο αυτό χρησιμοποιείται, ώστε να προσδιορίσει ο αποστολέας του αιτήματος σε ποιο πρόβλημα αναφέρονται τα δεδομένα που παρέχει, με σκοπό η υπηρεσία oracle που πραγματοποιεί τη σύνδεση του blockchain με τον εξωτερικό κόσμο να τροφοδοτήσει τα δεδομένα στο κατάλληλο σύστημα.
string ipfsHash	Το πεδίο αυτό περιλαμβάνει το IPFS hash το οποίο αντιστοιχεί στα δεδομένα στα οποία αναφέρεται το αίτημα. Όπως αναφέρθηκε και νωρίτερα το κύριο μέρος των δεδομένων αποθηκεύεται εκτός αλυσίδας με σκοπό να μην την επιβαρύνει.

bool agreed	Το πεδίο αυτό είναι βοηθητικό και χρησιμοποιείται ώστε να ορίσει αν οι κόμβοι απόφασης έχουν επέλθει σε συμφωνία για τα δεδομένα.
bool anomaly	Το πεδίο αυτό είναι βοηθητικό και χρησιμοποιείται ώστε να ορίσει το αποτέλεσμα της συμφωνίας των κόμβων για τα δεδομένα. Αν το χαρακτηριστικό ***agreed*** δεν έχει ορισθεί, το περιεχόμενο αυτής της μεταβλητής είναι άκυρο και δεν πρέπει να λαμβάνεται υπόψιν.
uint normalCounter	Το πεδίο αυτό είναι βοηθητικό και χρησιμοποιείται ώστε να καταμετρήσει τους κόμβους που έχουν αποφανθεί πως τα δεδομένα αυτά αποτελούν κανονικά δεδομένα.
uint anomalyCounter	Το πεδίο αυτό είναι βοηθητικό και χρησιμοποιείται ώστε να καταμετρήσει τους κόμβους που έχουν αποφανθεί πως τα δεδομένα αυτά αποτελούν έκτοπα δεδομένα (ανωμαλία).
mapping(address => uint) quorum)	Το πεδίο αυτό είναι βοηθητικό και χρησιμοποιείται ώστε να εντοπίζονται οι κόμβοι που έχουν ήδη συμμετάσχει με την ψήφο τους στη διαδικασία απόφασης για ένα συγκεκριμένο δείγμα δεδομένων, με σκοπό να αποφεύγεται το φαινόμενο της «διπλής-ψήφου».

Πίνακας 4: Επεξήγηση μεταβλητών της δομής Request

4.2.1.2. Το πρωτόκολλο συμφωνίας

Ένα άλλο βασικό τμήμα του έξυπνου συμβολαίου, το οποίο μάλιστα θα μεταβάλλεται κατά την εκτέλεση των σεναρίων – πειραμάτων που θα ακολουθήσουν του κεφαλαίου αυτού, είναι το πρωτόκολλο συμφωνίας μεταξύ των κόμβων απόφασης. Αυτό βασίζεται στην επίτευξη είτε απλής, είτε ενισχυμένης πλειοψηφίας και στον κώδικα αποτυπώνεται με αρκετά άμεσο τρόπο, όπως και οι διευθύνσεις των κόμβων απόφασης.

```
uint minQuorum = 2; //minimum number of responses to receive before declaring final result
uint totalOracleCount = 3; // Hardcoded oracle count
```

Εικόνα 8: Ορισμός μεταβλητών πρωτοκόλλου απόφασης

Στον πίνακα που ακολουθεί επεξηγείται ο ρόλος των μεταβλητών που συμμετέχουν στη διαδικασία απόφασης [Πίνακας 5].

uint totalOracleCount	Η μεταβλητή αυτή περιλαμβάνει το συνολικό πλήθος των κόμβων απόφασης που συμμετέχουν στη διαδικασία.
uint minQuorum	Η μεταβλητή αυτή περιλαμβάνει το ελάχιστο πλήθος των κόμβων απόφασης που απαιτείται να συμφωνήσουν με σκοπό η απόφαση να θεωρηθεί ειλημμένη.

Πίνακας 5: Επεξήγηση μεταβλητών διαδικασίας απόφασης

4.2.1.3. Τα γεγονότα ενημέρωσης του εξωτερικού περιβάλλοντος

Όπως αναφέρθηκε και κατά την ανάλυση της αρχιτεκτονικής, ως μηχανισμός ενημέρωσης των εξωτερικών παρατηρητών για αλλαγές στο blockchain που απαιτούν τη συνδρομή τους ή ενδέχεται να τους ενδιαφέρουν, χρησιμοποιούνται τα γεγονότα (events). Στο έξυπνο συμβόλαιο συναντάμε δύο είδη γεγονότων, το νέο αίτημα (NewRequest) και το ανανεωμένο αίτημα (UpdatedRequest) [Εικόνα 9].

```
//event that triggers oracle outside of the blockchain
event NewRequest (
  uint id,
  string taskToSolve,
  string ipfsHash
);

//triggered when there's a consensus on the final result
event UpdatedRequest (
  uint id,
  string taskToSolve,
  string ipfsHash,
  bool anomaly
);
```

Εικόνα 9: Υλοποίηση των δύο τύπων γεγονότων

Το γεγονός NewRequest προορίζεται για να ενημερώσει τους εξωτερικούς παρατηρητές oracles των κόμβων απόφασης για την ανάγκη να προωθήσουν τα περιεχόμενα του αιτήματος στα συστήματα έξυπνης απόφασης μέσω μηχανικής μάθησης που κατέχουν και να ενημερώσουν το blockchain για το αποτέλεσμα, φέροντας τα δεδομένα απόφασης εντός αλυσίδας.

uint id	Το μοναδικό προσδιοριστικό κάθε αιτήματος. Αποτελεί ουσιαστικά ένα κύριο κλειδί ανάκτησης του αιτήματος.
string taskToSolve	Το πεδίο αυτό χρησιμοποιείται, ώστε να προσδιορίσει ο αποστολέας του αιτήματος σε ποιο πρόβλημα αναφέρονται τα δεδομένα που παρέχει, με σκοπό η υπηρεσία oracle που πραγματοποιεί τη σύνδεση του blockchain με τον εξωτερικό κόσμο να τροφοδοτήσει τα δεδομένα στο κατάλληλο σύστημα.
string ipfsHash	Το πεδίο αυτό περιλαμβάνει το IPFS hash το οποίο αντιστοιχεί στα δεδομένα στα οποία αναφέρεται το αίτημα. Χρησιμοποιείται από τα εξωτερικά συστήματα ώστε να προσπελάσουν, μέσω του IPFS το περιεχόμενο των δεδομένων για να εκτελέσουν την εκτίμησή τους.

Πίνακας 6: Επεξήγηση μεταβλητών γεγονότος NewRequest

Το γεγονός UpdatedRequest προορίζεται ώστε να ενημερώσει εξωτερικούς παρατηρητές πραγματικού χρόνου για την λήψη απόφασης πάνω σε ένα δείγμα δεδομένων, ώστε να μπορούν να προχωρήσουν τυχόν διαδικασίες τους που εξαρτώνται από το αποτέλεσμα με την ελάχιστη καθυστέρηση.

uint id	Το μοναδικό προσδιοριστικό κάθε αιτήματος. Αποτελεί ουσιαστικά ένα κύριο κλειδί ανάκτησης του αιτήματος.
string taskToSolve	Το πεδίο αυτό χρησιμοποιείται, ώστε να προσδιορίσει ο αποστολέας του αιτήματος σε ποιο πρόβλημα αναφέρονται τα δεδομένα που παρέχει, με σκοπό η υπηρεσία oracle που πραγματοποιεί τη σύνδεση του blockchain με τον εξωτερικό κόσμο να τροφοδοτήσει τα δεδομένα στο κατάλληλο σύστημα.
string ipfsHash	Το πεδίο αυτό περιλαμβάνει το IPFS hash το οποίο αντιστοιχεί στα δεδομένα στα οποία αναφέρεται το αίτημα. Χρησιμοποιείται από τα εξωτερικά συστήματα ώστε να προσπελάσουν, μέσω του IPFS το περιεχόμενο των δεδομένων για να εκτελέσουν την εκτίμησή τους.
bool anomaly	Το πεδίο αυτό περιέχει το αποτέλεσμα της εκτίμησης του εξωτερικού συστήματος απόφασης, όσον αφορά τα δεδομένα τα οποία περιγράφονται από το παραπάνω id και ipfsHash, για το συγκεκριμένο πρόβλημα taskToSolve.

Πίνακας 7: Επεξήγηση μεταβλητών γεγονότος UpdatedRequest

4.2.1.4. Συναρτήσεις αλληλεπίδρασης με το έξυπνο συμβόλαιο

Η διάδραση με το έξυπνο συμβόλαιο, τόσο για τους απλούς χρήστες όσο και για τους κόμβους απόφασης συντελείται μέσω παρεχόμενων συναρτήσεων [Εικόνα 10].

4.2.1.4.1. Η συνάρτηση δημιουργίας αιτήματος createRequest

Η συνάρτηση createRequest παρέχει τη δυνατότητα στους χρήστες να δημιουργήσουν ένα νέο αίτημα.

```
function createRequest (
  string memory _taskToSolve,
  string memory _ipfsHash
)
public
{
  uint lenght = requests.push(Request(currentId, _taskToSolve, _ipfsHash, false, false, 0, 0));
  Request storage r = requests[lenght-1];

  // Hardcoded oracles address
  r.quorum[address(0x5694D143eb8833D9A3D209b4F43F74300974b1A1)] = 1;
  r.quorum[address(0x576296c126AF22A0CEC969c53E51cCc18E5f079B)] = 1;
  r.quorum[address(0x0aF2572D44BebCa8d82a088C5f5a63357fa91fbA)] = 1;

  // launch an event to be detected by oracle outside of blockchain
  emit NewRequest (
    currentId,
    _taskToSolve,
    _ipfsHash
  );

  // increase request id
  currentId++;
}
```

Εικόνα 10: Υλοποίηση συνάρτησης δημιουργίας νέου αιτήματος

Τα ορίσματα που χρειάζεται να παρέχουν είναι μόνο τα ***taskToSolve*** *και* ***ipfsHash***, με περιεχόμενο που έχει επεξηγηθεί νωρίτερα. Αποτελεί ευθύνη των χρηστών και των διεπαφών διάδρασης με το πληροφοριακό σύστημα που παρέχουμε η εγκυρότητα των στοιχείων αυτών, καθώς αποτελεί καίρια προϋπόθεση για την ομαλή και έγκυρη λήψη της απόφασης.

Στο σώμα της συνάρτησης κατασκευάζεται ένα νέο αίτημα με τα χαρακτηριστικά που δόθηκαν από το χρήστη, το οποίο αποθηκεύεται στη λίστα με τα αιτήματα που έχουν υποβληθεί στο έξυπνο συμβόλαιο. Σημαντική λεπτομέρεια αποτελεί το γεγονός πως οι διευθύνσεις των κόμβων απόφασης αποτελούν κομμάτι του κώδικα, είναι δηλαδή "hardcoded", πράγμα το οποίο αποτελεί μία πρακτική που γενικά απαιτεί μεγάλη προσοχή και δυσχεραίνει την επέκταση και αναβάθμιση του συμβολαίου. Σε κάθε προσθήκη, αφαίρεση ή αλλαγή διεύθυνσης κόμβου απόφασης τμήματα του κώδικα πρέπει να συντηρούνται με σκοπό την εύρυθμη λειτουργία του συστήματος. Παρόλα αυτά ο συμβιβασμός αυτός επιτρέπει την προσομοίωση των χαρακτηριστικών ενός consortium ή private blockchain σε ένα δημόσιο blockchain όπως το Ethereum.

Στο τέλος της συνάρτησης εκπέμπεται ένα νέο γεγονός νέου αιτήματος, με σκοπό να εκκινήσει τη διαδικασία απόφασης από τους κόμβους απόφασης και τα oracles τους. Επιπλέον, αυξάνεται ο μετρητής αιτημάτων που εκχωρεί σε κάθε νέο αίτημα το id του με σκοπό να διασφαλίζονται οι προϋποθέσεις για να αποτελεί αυτό κύριο κλειδί.

string taskToSolve	Το πεδίο αυτό χρησιμοποιείται, ώστε να προσδιορίσει ο αποστολέας του αιτήματος σε ποιο πρόβλημα αναφέρονται τα δεδομένα που παρέχει, με σκοπό η υπηρεσία oracle που πραγματοποιεί τη σύνδεση του blockchain με τον εξωτερικό κόσμο να τροφοδοτήσει τα δεδομένα στο κατάλληλο σύστημα.

string ipfsHash	Το πεδίο αυτό περιλαμβάνει το IPFS hash το οποίο αντιστοιχεί στα δεδομένα στα οποία αναφέρεται το αίτημα. Χρησιμοποιείται από τα εξωτερικά συστήματα ώστε να προσπελάσουν, μέσω του IPFS το περιεχόμενο των δεδομένων για να εκτελέσουν την εκτίμησή τους.

Πίνακας 8: Επεξήγηση ορισμάτων συνάρτησης createRequest

4.2.1.4.2. Η συνάρτηση ανανέωσης αιτήματος updateRequest

Η συνάρτηση updateRequest παρέχει τη δυνατότητα στους χρήστες να προσκομίσουν στην αλυσίδα τα δεδομένα απόφασης για κάποιο από τα αιτήματα [Εικόνα 11].

Σημαντικό είναι το γεγονός πως αν και η συνάρτηση μπορεί να κληθεί από οποιονδήποτε, οι διαδοχικοί έλεγχοι που βρίσκονται στο σώμα της διασφαλίζουν πως επιπλέον δεδομένα στη λήψη της απόφασης θα μπορέσουν να προσκομίσουν μόνο κόμβοι που είναι προκαθορισμένοι ως κόμβοι απόφασης για αυτό το αίτημα και μάλιστα δεν έχουν ξανασυμμετάσχει στη διαδικασία. Σε περίπτωση που τα νέα δεδομένα που προσκομίζονται μέσω της συνάρτησης οδηγούν σε συμφωνία και λήψη απόφασης εκπέμπεται και σχετικό γεγονός, ώστε να ενημερωθούν καταλλήλως οι εξωτερικοί παρατηρητές.

```
//called by the oracle to record its answer
function updateRequest (
  uint _id,
  bool _oracleDecision
) public {

  Request storage currRequest = requests[_id];

  //check if oracle is in the list of trusted oracles
  //and if the oracle hasn't voted yet
  if(currRequest.quorum[address(msg.sender)] == 1){

    //marking that this address has voted
    currRequest.quorum[msg.sender] = 2;

    if(_oracleDecision == true){
      currRequest.anomalyCounter++;
      if(currRequest.anomalyCounter == minQuorum){
        currRequest.agreed = true;
        currRequest.anomaly = true;
        emit UpdatedRequest (
          currRequest.id,
          currRequest.taskToSolve,
          currRequest.ipfsHash,
          currRequest.anomaly
        );
      }
    } else {
      currRequest.normalCounter++;
      if(currRequest.normalCounter == minQuorum){
        currRequest.agreed = true;
        currRequest.anomaly = false;
        emit UpdatedRequest (
          currRequest.id,
          currRequest.taskToSolve,
          currRequest.ipfsHash,
          currRequest.anomaly
        );
      }
    }
  }
 }
}
```

Εικόνα 11: Υλοποίηση της συνάρτησης ανανέωσης αιτήματος

uint id	Το μοναδικό προσδιοριστικό κάθε αιτήματος. Αποτελεί ουσιαστικά ένα κύριο κλειδί ανάκτησης του αιτήματος.
bool oracleDecision	Το πεδίο αυτό περιέχει το αποτέλεσμα της εκτίμησης του εξωτερικού συστήματος απόφασης, όσον αφορά τα δεδομένα τα οποία περιγράφονται από το παραπάνω id και ipfsHash, για το συγκεκριμένο πρόβλημα taskToSolve.

Πίνακας 9: Επεξήγηση ορισμάτων συνάρτησης updateRequest

4.2.1.4.3. Η συνάρτηση λήψης κατάστασης αιτήματος getState

Πέρα από τις συναρτήσεις, οι οποίες επιτρέπουν στους χρήστες να γράφουν νέα στοιχεία στο blockchain, προσφέρονται παράλληλα και κάποιες συναρτήσεις view, οι οποίες δεν απαιτούν την χρήση Ethers ως αντίτιμο για την κλήση τους. Οι συναρτήσεις

αυτές χρησιμοποιούνται ώστε να προβάλουν βασικές πληροφορίες σχετικά με τα δεδομένα που βρίσκονται εντός αλυσίδας.

Η πρώτη από τις δύο συναρτήσεις προβολής που προσφέρονται στο έξυπνο συμβόλαιο είναι η συνάρτηση λήψης κατάστασης getState. Η συνάρτηση αυτή επιτρέπει στο χρήστη που την καλεί να ελέγξει σε τι κατάσταση βρίσκεται η διαδικασία λήψης απόφασης από τους κόμβους απόφασης για ένα συγκεκριμένο αίτημα, προσκομίζοντας το περιεχόμενο της μεταβλητής "*agreed*" του αιτήματος. Θετική επιστροφή συνεπάγεται πως οι κόμβοι έχουν λάβει απόφαση σύμφωνα με το πρωτόκολλο απόφασης, ενώ αρνητική σημαίνει πως δεν έχει επέλθει ακόμα συμφωνία. Βασικό είναι να αναφέρουμε πως σε περίπτωση αρνητικής απάντησης δεν διασφαλίζεται ότι μπορεί κάποια στιγμή να επέλθει συμφωνία, καθώς υπάρχουν περιπτώσεις όπου η διαφωνία των κόμβων οδηγεί σε αδυναμία συμφωνίας. Επιπλέον, σημειώνεται πως για όσο διάστημα το αποτέλεσμα επιστροφής αυτής της συνάρτησης είναι αρνητικό, δεν πρέπει να λογίζεται ως έγκυρο το αποτέλεσμα της συνάρτησης getValue.

```
//called to fetch the state of the decision
function getState(
  uint _id
) public view returns(bool) {
  Request storage currRequest = requests[_id];
  return currRequest.agreed;
}
```

Εικόνα 12: Υλοποίηση της συνάρτησης getState

uint id	Το μοναδικό προσδιοριστικό κάθε αιτήματος. Αποτελεί ουσιαστικά ένα κύριο κλειδί ανάκτησης του αιτήματος.
bool RETURN	Η κατάσταση στην οποία βρίσκεται η διαδικασία απόφασης για το ζητούμενο αίτημα. True σε περίπτωση ειλημμένης απόφασης, False σε κάθε άλλη περίπτωση.

Πίνακας 10: Επεξήγηση ορίσματος και επιστροφής getState

4.2.1.4.4. Η συνάρτηση λήψης τιμής αιτήματος getValue

Η δεύτερη από τις δύο συναρτήσεις προβολής που προσφέρονται στο έξυπνο συμβόλαιο είναι η συνάρτηση λήψης τιμής getValue. Η συνάρτηση αυτή επιτρέπει στο χρήστη που την καλεί να ελέγξει την τιμή της απόφασης από τους κόμβους απόφασης για ένα συγκεκριμένο αίτημα, προσκομίζοντας το περιεχόμενο της μεταβλητής "*anomaly*" του αιτήματος. Θετική επιστροφή συνεπάγεται πως οι κόμβοι θεωρούν πως τα συγκεκριμένα δεδομένα αποτελούν έκτοπα δεδομένα (ανωμαλία), ενώ αρνητική συνεπάγεται πως τα δεδομένα είτε είναι κανονικά είτε δεν έχει επέλθει ακόμα συμφωνία. Η υλοποίηση έχει γίνει

με τέτοιο τρόπο, ώστε να συνάδει με τη λογική του τεκμηρίου αθωότητας, δηλαδή λογίζει κάθε δεδομένο ως φυσιολογικό αν δεν υπάρξει ομόφωνη απόφαση για το αντίθετο.

```
//called to fetch result
function getValue(
  uint _id
) public view returns(bool) {
  Request storage currRequest = requests[_id];
  return currRequest.anomaly;
}
```

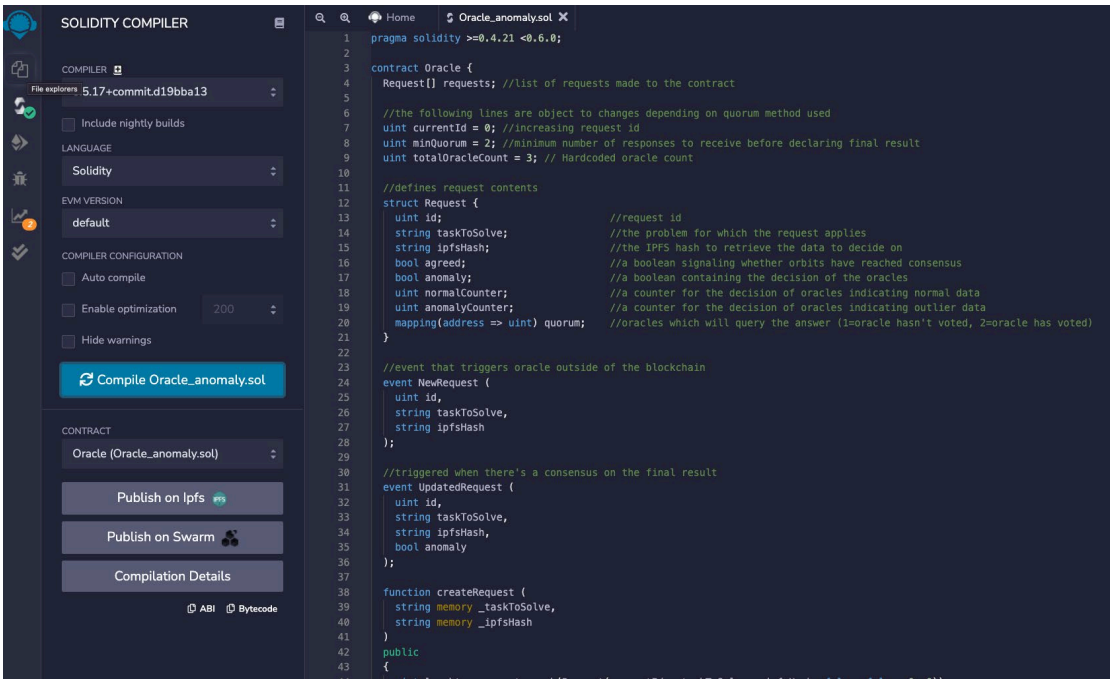
Εικόνα 13: Υλοποίηση της συνάρτησης getValue

uint id	Το μοναδικό προσδιοριστικό κάθε αιτήματος. Αποτελεί ουσιαστικά ένα κύριο κλειδί ανάκτησης του αιτήματος.
bool RETURN	Η τρέχουσα εκτίμηση των κόμβων για το συγκεκριμένο αίτημα.True σε περίπτωση ειλημμένης απόφασης πως αποτελεί έκτοπο δείγμα, False σε κάθε άλλη περίπτωση.

Πίνακας 11: Επεξήγηση ορίσματος και επιστροφής συνάρτησης getValue

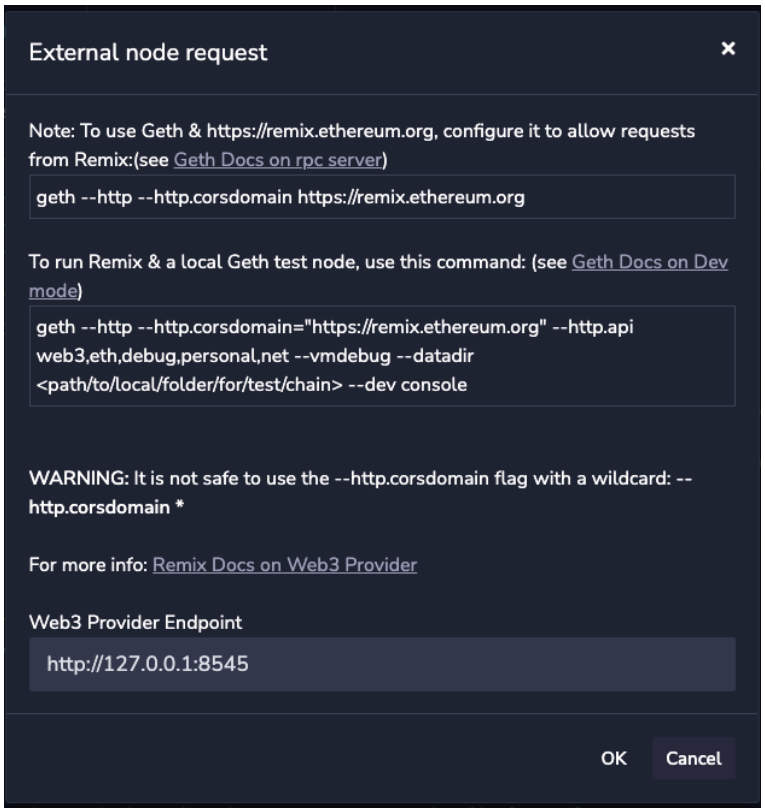
4.2.2. Μεταγλώττιση και deployment έξυπνου συμβολαίου

Για τη μεταγλώττιση του έξυπνου συμβολαίου θα χρησιμοποιηθεί η έκδοση «0.5.17» της Solidity. Χρησιμοποιώντας το περιβάλλον του Remix IDE η μεταγλώττιση ολοκληρώνεται χωρίς κάποιο σφάλμα.



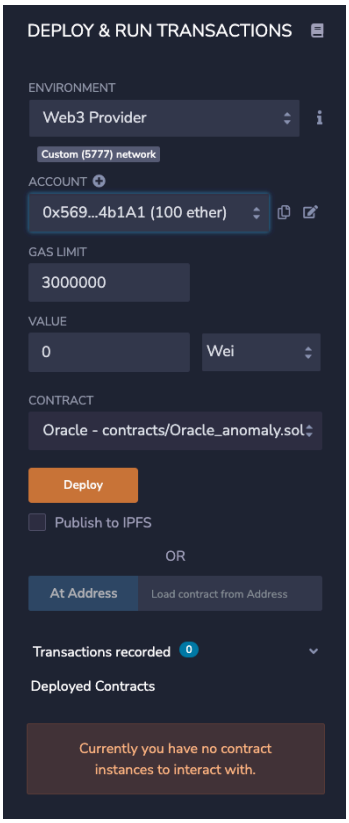
Εικόνα 14: Μεταγλώττιση συμβολαίου από το περιβάλλον του Remix IDE

Για να προχωρήσουμε στο deployment στο τοπικό περιβάλλον Ethereum που έχουμε δημιουργήσει μέσω του Ganache. Για να επιτευχθεί αυτό καταφεύγουμε στο αντίστοιχο κομμάτι της εφαρμογής Remix IDE, και επιλέγουμε στο πεδίο «environment» τον τοπικό πάροχο Web3 (local web3 provider). Για να καταφέρει να επιτευχθεί η σύνδεση, πρέπει ο τοπικός πάροχος να μπορεί να δέχεται αιτήματα στο endpoint «http://127.0.0.1:8545».



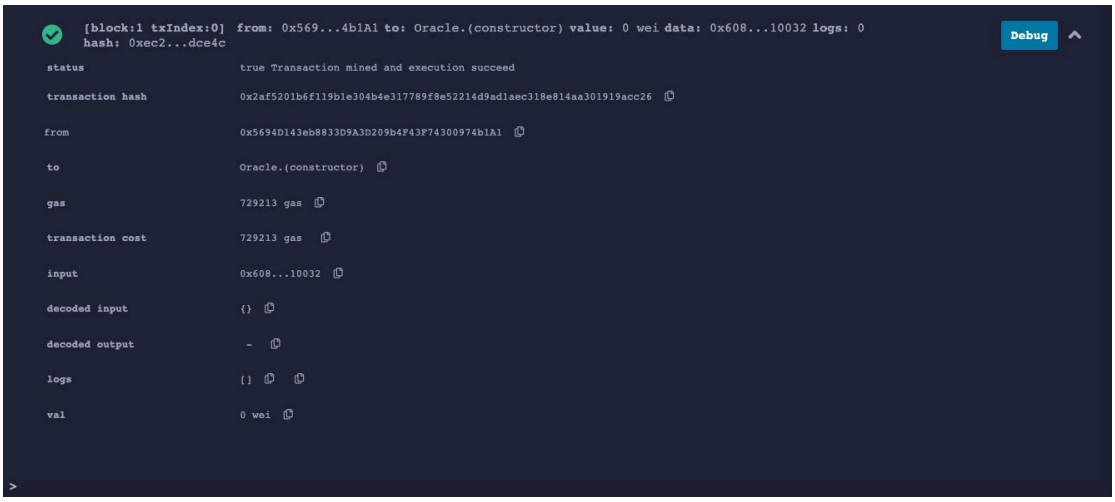
Εικόνα 15: Μήνυμα πληροφοριών σύνδεσης Remix IDE και Ganache

Επιβεβαιώνοντας τα στοιχεία αυτά, τα οποία έχουμε συμπληρώσει και νωρίτερα κατά την κατασκευή του περιβάλλοντος στο Ganache, μας επιτρέπεται το deployment του μεταγλωττισμένου συμβολαίου, χρησιμοποιώντας έναν από τους λογαριασμούς που έχουν δημιουργηθεί στο Ganache και αυτόματα φορτωθεί και στο περιβάλλον του Remix IDE.



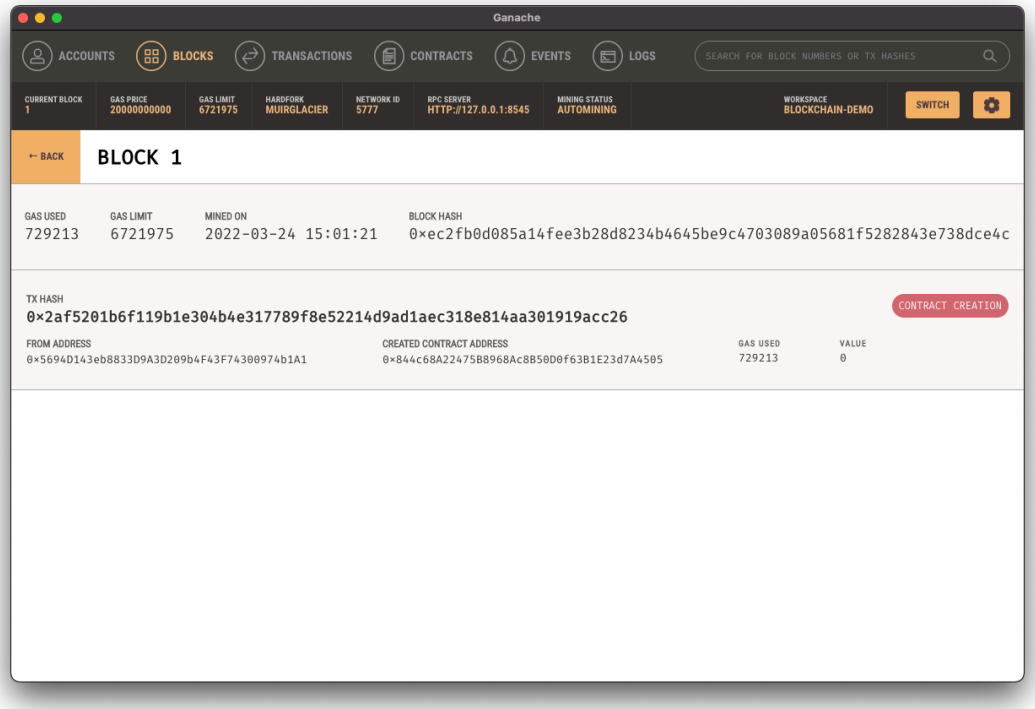
Εικόνα 16: Πεδίο επιλογής λεπτομερειών deployment

Επιλέγοντας τον πρώτο λογαριασμό και πατώντας το πλήκτρο “Deploy”, παρατηρούμε πως το έξυπνο συμβόλαιο έχει γίνει επιτυχώς deployed στο τοπικό περιβάλλον Ethereum του Ganache. Αυτό μπορούμε να το επιβεβαιώσουμε τόσο από το τερματικό του Remix IDE, στο οποίο εμφανίζεται η συναλλαγή ως έγκυρη [Εικόνα 17].



Εικόνα 17: Μήνυμα επιτυχούς deployment στο τερματικό του Remix IDE

Επιπλέον, η δημιουργία του block μπορεί να επιβεβαιωθεί και από το αντίστοιχο περιβάλλον του Ganache, όπως φαίνεται παρακάτω [Εικόνα 18].

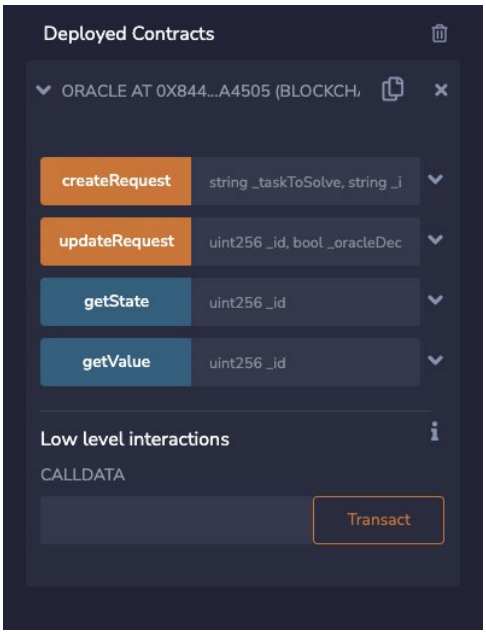


Εικόνα 18: Block deployment του συμβολαίου στο περιβάλλον του Ganache

4.2.3. Έλεγχος έξυπνου συμβολαίου

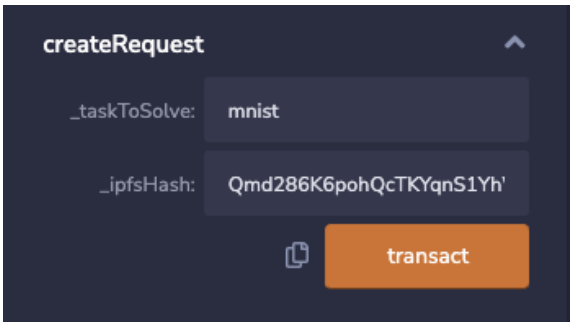
Ολοκληρώνοντας το deployment του έξυπνου συμβολαίου, μας δίνεται πλέον η δυνατότητα πλήρους εκτέλεσης όλων των συναρτήσεων και προσομοίωσης της λειτουργίας της, μέσω του διαχειριστικού περιβάλλοντος που προσφέρει το Remix IDE [Εικόνα 19]. Πρόκειται, λοιπόν, να ελέγξουμε σε αυτό το στάδιο την ορθή λειτουργία της επιχειρησιακής λογικής του έξυπνου συμβολαίου, προτού προχωρήσουμε με την ανάπτυξη των υπόλοιπων τμημάτων του πληροφοριακού συστήματος.

Σημειώνεται πως θα προσομοιωθεί η συμπεριφορά όλων των τμημάτων του πληροφοριακού συστήματος, όπως των oracles και του IPFS, με χειροκίνητη εισαγωγή δεδομένων, τα οποία ανταποκρίνονται ως προς τη μορφή τους στην πραγματικότητα.



Εικόνα 19: Διαχειριστικό περιβάλλον κλήσης συναρτήσεων Remix IDE

Για την εκκίνηση του ελέγχου θα δημιουργήσουμε ένα νέο αίτημα, χρησιμοποιώντας τη συνάρτηση createRequest, εκτελώντας τη συναλλαγή από τον πρώτο λογαριασμό. Η συμπλήρωση των πεδίων φαίνεται στο παρακάτω στιγμιότυπο [Εικόνα 20].



Εικόνα 20: Κλήση της συνάρτησης createRequest με αληθοφανή δεδομένα

Εκτελώντας τη συναλλαγή πατώντας το πλήκτρο “transact” παρατηρούμε από το τερματικό πως αυτή εκτελείται επιτυχώς [Εικόνα 21].

Εικόνα 21: Μήνυμα επιτυχούς εκτέλεσης της συνάρτησης createRequest

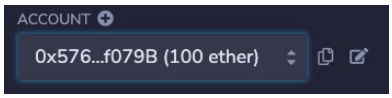
Εκτός της επιτυχούς εκτέλεσης μας ενδιαφέρει να ελέγξουμε τα περιεχόμενα της συναλλαγής. Συγκεκριμένα, όπως φαίνεται παρακάτω, τόσο τα ορίσματα γίνονται κανονικά δεκτά στη συναλλαγή, όσο και το γεγονός που πρέπει να εκπέμπεται από τη συνάρτηση εκπέμπεται κανονικά [Εικόνα 22].

Εικόνα 22: Περιεχόμενα συναλλαγής createRequest

Έπειτα από την επιτυχή εκτέλεση της δημιουργίας ενός αιτήματος, θα αλλάξουμε λογαριασμό και από τον δεύτερο κατά σειρά λογαριασμό θα ελέγξουμε τις δύο συναρτήσεις προβολής. Όπως φαίνεται παρακάτω και οι δύο επιστρέφουν τα αναμενόμενα αποτελέσματα, δηλαδή η και οι δύο "false" [Εικόνα 23-24]. Επιπλέον, όπως ήταν αναμενόμενο παρατηρούμε πως δεν χρεώθηκε με κάποιο αντίτιμο ο λογαριασμός του χρήστη, αφού η κλήση των συναρτήσεων προβολής δεν κοστίζει [Εικόνα 25].

Εικόνα 23: Αποτέλεσμα κλήσης getState σε αίτημα που δεν έχει ολοκληρωθεί η λήψη απόφασης

Εικόνα 24: Αποτέλεσμα κλήσης getValue σε αίτημα που δεν έχει ολοκληρωθεί η λήψη απόφασης



Εικόνα 25: Αμετάβλητο υπόλοιπο λογαριασμού μετά από κλήση συνάρτησης προβολής

Στο σημείο αυτό θα προσομοιώσουμε τη λειτουργία των oracles, χρησιμοποιώντας τη συνάρτηση updateRequest, ώστε να προσκομίσουμε στο blockchain δεδομένα που αναπαριστούν τις εκτιμήσεις που παράγονται από τα μοντέλα των κόμβων αποφάσεων. Χρησιμοποιώντας λοιπόν τον δεύτερο και τρίτο κατά σειρά λογαριασμό, θα χρησιμοποιήσουμε τη συνάρτηση αυτή προορίζεται για χρήση από τις υπηρεσίες oracle των κόμβων απόφασης και θα εισάγουμε και από τους δύο κόμβους μία εκτίμηση πως τα δεδομένα που υπέβαλε ο πρώτος λογαριασμός για εξέταση αποτελούν ανωμαλία.

Εικόνα 26: Επιτυχημένη εισαγωγή απόφασης μέσω της updateRequest

Εικόνα 27: Επιτυχής εκπομπή γεγονότος μετά την επίτευξη συμφωνίας

Παρατηρούμε πως οι κόμβοι καταφέρνουν με επιτυχία να εισάγουν τις εκτιμήσεις τους στο έξυπνο συμβόλαιο, ενώ επιπλέον κατά την επίτευξη της συμφωνίας εκπέμπεται κανονικά το γεγονός τύπου UpdatedRequest, το οποίο χρησιμοποιείται με αυτό το σκοπό. Ακόμα, εξετάζοντας τα δεδομένα της αλυσίδας με τις συναρτήσεις getValue και getState παρατηρούμε πως το περιεχόμενο των μεταβλητών έχει ανανεωθεί επιτυχώς, ώστε να αντικατοπτρίζει την πραγματικότητα.

Στο στάδιο αυτό θα ολοκληρώσουμε προσωρινά τον έλεγχο του έξυπνου συμβόλαιο μέσω της διεπαφής του Remix IDE. Έλεγχος σε περισσότερα σενάρια θα διενεργηθεί μέσω των διεπαφών που θα κατασκευαστούν στα πλαίσια της διπλωματικής εργασίας.

4.3. Ανάπτυξη διεπαφής τερματικού NodeJS CLI Service

Η κύρια διεπαφή χρήσης των υπηρεσιών του πληροφοριακού συστήματος, η οποία παρέχει πλήρη πρόσβαση σε όλες τις λειτουργίες της εφαρμογής, είναι η διεπαφή τερματικού NodeJS CLI Service. Η διεπαφή αυτή χρησιμοποιείται τόσο ώστε να υποβάλει κανείς αιτήματα στο blockchain, όσο και στο να προσπελάσει δεδομένα που βρίσκονται εντός αλυσίδας και να παρακολουθήσει γεγονότα που εκπέμπονται από την εφαρμογή.

4.3.1. Επικοινωνία με το έξυπνο συμβόλαιο και εκτέλεση συναρτήσεων

Η διεπαφή χρησιμοποιεί τη βιβλιοθήκη "web3-eth-contract" με σκοπό να επικοινωνήσει με το έξυπνο συμβόλαιο και να εκτελέσει συναρτήσεις. Για την επίτευξη της επικοινωνίας αυτής, είναι ανάγκη να παρέχονται οι διευθύνσεις του έξυπνου συμβολαίου και του πορτοφολιού του χρήστη, καθώς επίσης και το ABI του έξυπνου συμβολαίου, το οποίο μπορεί να ανακτηθεί από το Remix IDE. Όπως είναι εμφανές προτιμήθηκε να αποφευχθεί η απευθείας χρήση των διευθύνσεων στο κύριο σώμα του κώδικα και αντί αυτού χρησιμοποιήθηκαν μεταβλητές περιβάλλοντος που ορίζονται με αρχεία .config, πράγμα που θα επεξηγηθεί παρακάτω.

```
var Contract = require('web3-eth-contract');
const MyContract = require('./contract_test.json')

// set provider for all later instances to use
Contract.setProvider(process.env.WEB3_PROVIDER);

const myAddress = process.env.WALLET_ADDRESS;

var contract = new Contract(MyContract, process.env.CONTRACT_ADDRESS, {
  from: myAddress,
  gas: 5000000
});
```

Εικόνα 28: Τμήμα ορισμού βασικών μεταβλητών για την επικοινωνία με το έξυπνο συμβόλαιο

Η κλήση των συναρτήσεων που παρέχει το έξυπνο συμβόλαιο γίνεται μέσω δύο διαφορετικών μεθόδων που παρέχει η βιβλιοθήκη "web3-eth-contract". Για τις συναρτήσεις που αλλάζουν το περιεχόμενο της αλυσίδας και κατά προέκταση απαιτούν έξοδα από το λογαριασμό του χρήστη, χρησιμοποιείται η μέθοδος *send()* [Εικόνα 29]. Οι συναρτήσεις αυτές είναι οι createRequest και updateRequest.

```
function createNewRequest(taskToSolve, ipfsHash, result) {
  contract.methods.createRequest(taskToSolve, ipfsHash).send({
    from: myAddress})
```

Εικόνα 29: Απόσπασμα κώδικα κλήσης της μεθόδου send

Αντιστοίχως, οι συναρτήσεις προβολής που είναι τύπου view, καλούνται χρησιμοποιώντας τη μέθοδο *call()* που παρέχεται από την ίδια βιβλιοθήκη [Εικόνα 30].

```
function getContractState(id, result) {
  contract.methods.getState(id).call({
```

Εικόνα 30: Απόσπασμα κώδικα κλήσης της μεθόδου call

Η υλοποίηση των συναρτήσεων δε θα επεξηγηθεί περαιτέρω, καθώς μετέπειτα σε αυτό το κεφάλαιο θα εστιάσουμε και θα επεξηγήσουμε εκτενώς τις εντολές που παρέχει η επαφή τερματικού, οι οποίες βρίσκονται σε αντιστοιχία με αυτές τις συναρτήσεις.

4.3.2. Επικοινωνία με το IPFS και προσθήκη αρχείων

Ένα βασικό κομμάτι της λειτουργικότητας που προσφέρει η διεπαφή τερματικού έγκειται στην ικανότητά της να αυτοματοποιήσει όλη τη ροή υποβολής δεδομένων προς εκτίμηση στο πληροφοριακό σύστημα, συμπεριλαμβανομένου και του ανεβάσματος των δεδομένων προς εκτίμηση στο IPFS. Για να επιτευχθεί αυτό χρησιμοποιείται η βιβλιοθήκη "ipfs-http-client", η οποία παρέχει τη δυνατότητα δημιουργίας σύνδεσης με έναν κόμβο του IPFS, μέσω της συνάρτησης *create(),* καθώς επίσης και τη δυνατότητα προσθήκης ενός αρχείου στο IPFS μέσω της μεθόδου *add().*

4.3.3. Ορισμός μεταβλητών περιβάλλοντος μέσω .config αρχείων

Η δυνατότητα παραμετροποίησης της διεπαφής, ώστε να μπορεί να τρέχει με διαφορετικά configurations γίνεται με την χρήση αρχείων .config και της βιβλιοθήκης "custom-env". Εντός του αρχείου .config ορίζουμε βασικές μεταβλητές περιβάλλοντος, οι οποίες περιλαμβάνουν διευθύνσεις υπηρεσιών που τρέχουν στο σύστημα και με τις οποίες επικοινωνεί η εφαρμογή, καθώς και τις διευθύνσεις του έξυπνου συμβολαίου και του πορτοφολιού του χρήστη [Εικόνα 31].

```
1  WEB3_PROVIDER=ws://localhost:8545
2  WEB3_URL=http://localhost:8545
3  CONTRACT_ADDRESS=0x76296d075E8c403767a8959f939d1f087A1975F5
4  ML_API_ENDPOINT=http://127.0.0.1:5000/testFiles
5  WALLET_ADDRESS=0xb5346CF224c02186606e5f89EACC21eC25398077
6  IPFS_ADDRESS=http://localhost:5001
```

Εικόνα 31: Παράδειγμα .config αρχείου και των περιεχομένων του

Κατά την εκκίνηση της εφαρμογής, μπορούμε να εναλλάσσουμε το τρέχον περιβάλλον, μέσω διαφορετικών .config αρχείων, ώστε να χρησιμοποιούμε διαφορετικούς τέτοιους συνδυασμούς μεταβλητών χωρίς αλλαγές στο βασικό μέρος του κώδικα.

4.3.4. Εντολές που παρέχονται από τη διεπαφή τερματικού

Παρακάτω επεξηγούνται όλες οι εντολές που μπορεί να εκτελέσει ο χρήστης μέσω της διεπαφής τερματικού. Για την υλοποίηση της διεπαφής τερματικού χρησιμοποιήθηκε η βιβλιοθήκη "yargs", η οποία παρέχει εύχρηστες συναρτήσεις για αντιστοίχιση συναρτήσεων σε εντολές τερματικού, έλεγχο ορισμάτων και παροχή βοήθειας στους χρήστες. Ακολουθούν οι εντολές της διεπαφής στην παρακάτω μορφή:

εντολή [όρισμα1] [όρισμα2]

4.3.4.1. getState [id]

Η εντολή αυτή αντιστοιχεί στη συνάρτηση getState του έξυπνου συμβολαίου. Ως όρισμα απαιτεί υποχρεωτικά το *id* του αιτήματος, του οποίου την κατάσταση ο χρήστης θέλει να ελέγξει. Σε περίπτωση επιτυχούς εκτέλεσης το αποτέλεσμα τυπώνεται στο τερματικό, ενώ σε αντίθετη περίπτωση τυπώνεται μήνυμα σφάλματος ή αποτυχίας με ανάλογη επεξήγηση.

uint id	Το μοναδικό προσδιοριστικό κάθε αιτήματος. Αποτελεί ουσιαστικά ένα κύριο κλειδί ανάκτησης του αιτήματος.

Πίνακας 12: Επεξήγηση ορισμάτων εντολής getState

4.3.4.2. getValue [id]

Η εντολή αυτή αντιστοιχεί στη συνάρτηση getValue του έξυπνου συμβολαίου. Ως όρισμα απαιτεί υποχρεωτικά το *id* του αιτήματος, του οποίου την τιμή ο χρήστης θέλει να ελέγξει. Σε περίπτωση επιτυχούς εκτέλεσης το αποτέλεσμα τυπώνεται στο τερματικό, ενώ σε αντίθετη περίπτωση τυπώνεται μήνυμα σφάλματος ή αποτυχίας με ανάλογη επεξήγηση.

uint id	Το μοναδικό προσδιοριστικό κάθε αιτήματος. Αποτελεί ουσιαστικά ένα κύριο κλειδί ανάκτησης του αιτήματος.

Πίνακας 13: Επεξήγηση ορισμάτων εντολής getValue

4.3.4.3. newRequest [task] [hash]

Η εντολή αυτή αντιστοιχεί στη συνάρτηση createRequest του έξυπνου συμβολαίου. Ως ορίσματα απαιτεί υποχρεωτικά το IPFS *hash* των δεδομένων προς εκτίμηση, καθώς επίσης και το πρόβλημα *task* στο οποίο αναφέρονται τα δεδομένα προς εκτίμηση. Σε περίπτωση επιτυχούς εκτέλεσης το αποτέλεσμα τυπώνεται στο τερματικό, ενώ σε αντίθετη περίπτωση τυπώνεται μήνυμα σφάλματος ή αποτυχίας με ανάλογη επεξήγηση.

string task	Το πεδίο αυτό χρησιμοποιείται, ώστε να

	προσδιορίσει ο αποστολέας του αιτήματος σε ποιο πρόβλημα αναφέρονται τα δεδομένα που παρέχει, με σκοπό η υπηρεσία oracle που πραγματοποιεί τη σύνδεση του blockchain με τον εξωτερικό κόσμο να τροφοδοτήσει τα δεδομένα στο κατάλληλο σύστημα.
string hash	Το πεδίο αυτό περιλαμβάνει το IPFS hash το οποίο αντιστοιχεί στα δεδομένα στα οποία αναφέρεται το αίτημα. Χρησιμοποιείται από τα εξωτερικά συστήματα ώστε να προσπελάσουν, μέσω του IPFS το περιεχόμενο των δεδομένων για να εκτελέσουν την εκτίμησή τους.

Πίνακας 14: Επεξήγηση ορισμάτων εντολής newRequest

4.3.4.4. uploadAndRequest [task] [filepath]

Η εντολή αυτή υλοποιεί μία σύνθετη ροή εργασιών. Αρχικά προσθέτει στο IPFS το αρχείο με τα δεδομένα προς εκτίμηση, το οποίο βρίσκεται τοπικά αποθηκευμένο στη διεύθυνση *filepath.* Σε περίπτωση επιτυχούς προσθήκης του αρχείου, ανακτά το IPFS hash, το οποίο χρησιμοποιεί μαζί με το όρισμα *task* ώστε να υποβάλει νέα δεδομένα προς εκτίμηση στο έξυπνο συμβόλαιο μέσω της συνάρτησης createRequest. Σε περίπτωση επιτυχούς εκτέλεσης το αποτέλεσμα τυπώνεται στο τερματικό, ενώ σε αντίθετη περίπτωση τυπώνεται μήνυμα σφάλματος ή αποτυχίας με ανάλογη επεξήγηση.

string task	Το πεδίο αυτό χρησιμοποιείται, ώστε να προσδιορίσει ο αποστολέας του αιτήματος σε ποιο πρόβλημα αναφέρονται τα δεδομένα που παρέχει, με σκοπό η υπηρεσία oracle που πραγματοποιεί τη σύνδεση του blockchain με τον εξωτερικό κόσμο να τροφοδοτήσει τα δεδομένα στο κατάλληλο σύστημα.
string filepath	Το πεδίο αυτό περιλαμβάνει τη διεύθυνση του αρχείου, το οποίο ο χρήστης επιθυμεί να ανεβάσει στο IPFS με σκοπό να το υποβάλει προς εκτίμηση, στο τοπικό σύστημα.

Πίνακας 15: Επεξήγηση ορισμάτων εντολής uploadAndRequest

4.3.4.5. updateRequest [id] [prediction]

Η εντολή αυτή αντιστοιχεί στη συνάρτηση updateRequest του έξυπνου συμβολαίου. Ως ορίσματα απαιτεί υποχρεωτικά *id* των δεδομένων για τα οποία ο κόμβος απόφασης πρόκειται να υποβάλει την εκτίμησή του, καθώς και το περιεχόμενο *prediction* της εκτίμησης αυτής. Σε περίπτωση επιτυχούς εκτέλεσης το αποτέλεσμα τυπώνεται στο τερματικό, ενώ σε αντίθετη περίπτωση τυπώνεται μήνυμα σφάλματος ή αποτυχίας με

ανάλογη επεξήγηση. Σημαντικό είναι να αναφερθεί πως η εντολή αυτή δε συμμετέχει στη γενική ροή εργασιών του πληροφοριακού συστήματος, καθώς σε καθεστώς κανονικής λειτουργίας την ευθύνη για να προσκομίσει τις εκτιμήσεις των κόμβων στην αλυσίδα κατέχει η ξεχωριστή υπηρεσία NodeJS Oracle.

uint id	Το μοναδικό προσδιοριστικό κάθε αιτήματος. Αποτελεί ουσιαστικά ένα κύριο κλειδί ανάκτησης του αιτήματος.
bool prediction	Το πεδίο αυτό περιλαμβάνει την εκτίμηση του κόμβου για τα δεδομένα, στα οποία αναφέρεται το αίτημα το ο οποίο προσδιορίζεται μοναδικά από το id.

Πίνακας 16: Επεξήγηση ορισμάτων εντολής updateRequest

4.3.5. Παρουσίαση και έλεγχος ορθής λειτουργίας διεπαφής τερματικού

Στο κομμάτι αυτό εκτελέστηκε μία πλήρης ροή εργασιών που περιλάμβανε την υποβολή αιτημάτων και των έλεγχο όλων των συναρτήσεων που παρουσιάστηκαν στις υποενότητες 4.3.4.1. – 4.3.4.5. . Η συμπεριφορά της διεπαφής βρέθηκε να συνάδει απόλυτα με τα αναμενόμενα, για το λόγο αυτό και θα χρησιμοποιηθεί ως μέσο εκτέλεσης των σεναρίων ελέγχου που ακολουθούν στο Κεφάλαιο 5. Η εκτενής παρουσίαση της διαδικασίας ελέγχου δεν κρίθηκε απαραίτητη σε αυτό το στάδιο καθώς, όχι μόνο έχει γίνει σε βάθος παρουσίαση της υλοποίησης των συναρτήσεων, αλλά επίσης πρόκειται να γίνει και εκτενής χρήση του υποσυστήματος αυτού στα σενάρια που ακολουθούν.

4.4. Υπηρεσία NodeJS Oracle Service

Η υπηρεσία NodeJS Oracle Service αποτελεί το συνδετικό κρίκο μεταξύ του έξυπνου συμβολαίου και των μοντέλων λήψης έξυπνων αποφάσεων των κόμβων απόφασης. Παρακολουθεί το δίκτυο για γεγονότα τα οποία αιτούνται δεδομένα που βρίσκονται εκτός αλυσίδας και μόλις τα αντιληφθεί διαβιβάζει αμέσως τα αιτήματα για εξυπηρέτηση στους τοπικούς εξυπηρετητές των κόμβων απόφασης, οι οποίοι διαχειρίζονται τα μοντέλα έξυπνων αποφάσεων μέσω μηχανικής μάθησης.

4.4.1. Επικοινωνία με το έξυπνο συμβόλαιο και εκτέλεση συναρτήσεων

Για την επικοινωνία του με το έξυπνο συμβόλαιο και την εκτέλεση των σχετικών συναρτήσεων η παρούσα υπηρεσία χρησιμοποιεί την ίδια λογική με τη διεπαφή τερματικού, βασιζόμενη στη βιβλιοθήκη “web3-eth-contract”. Τα κύρια στοιχεία της υλοποίησης ταυτίζονται για το λόγο αυτό δε θα εξηγηθούν περαιτέρω.

Μόνη προσθήκη – διαφοροποίηση του παρόντος τμήματος είναι η χρήση μίας επιπλέον μεθόδου πάνω στο έξυπνο συμβόλαιο, η οποία επιτρέπει την παρακολούθηση του δικτύου για γεγονότα. Η μέθοδος αυτή είναι η *events.AllEvents(),* η οποία μεταξύ άλλων

επιτρέπει να χρησιμοποιηθούν επιπλέον φίλτρα ως προς τον τύπο και το περιεχόμενο των γεγονότων που ανιχνεύονται. Μετά την επιτυχή ανίχνευση ενός γεγονότος, η υπηρεσία oracle εκτελεί μία σειρά από ενέργειες με αιτήματα σε εξωτερικούς προς το blockchain εξυπηρετητές και ξανά στο blockchain με σκοπό να προσκομίσει τα δεδομένα πίσω στην αλυσίδα.

4.4.2. Επικοινωνία με τους εξυπηρετητές αποφάσεων των κόμβων

Οι εκτιμήσεις των κόμβων πάνω στα δεδομένα ενός αιτήματος παρέχονται στην υπηρεσία oracle μέσω του REST API που προσφέρουν τα συστήματα αυτά. Για τη δημιουργία και αποστολή αιτημάτων χρησιμοποιήθηκε η βιβλιοθήκη “axios”. Η υπηρεσία με ένα POST αίτημα, συγκεκριμένου περιεχομένου στο κατάλληλο API endpoint, αιτείται την παροχή μίας εκτίμησης για τα δεδομένα, χωρίς να γνωρίζει περαιτέρω πληροφορίες για τη λογική λήψης αποφάσεων που χρησιμοποιεί το μοντέλο λήψης έξυπνων αποφάσεων. Στην απάντηση που δέχεται αν εξυπηρετηθεί επιτυχώς το αίτημα λαμβάνει την εκτίμηση την οποία και διαβιβάζει στην αλυσίδα.

4.4.3. Ορισμός μεταβλητών περιβάλλοντος μέσω .config αρχείων

Όπως και στη διεπαφή τερματικού έτσι και σε αυτό το τμήμα του πληροφοριακού συστήματος έγινε χρήση αρχείων .config με σκοπό τον ορισμό βασικών μεταβλητών διευθύνσεων. Στο παρακάτω στιγμιότυπο παρουσιάζεται το σύνολο αυτών των μεταβλητών για την παρούσα υπηρεσία [Εικόνα 32].

```
1  WEB3_PROVIDER=ws://localhost:8545
2  WEB3_URL=http://localhost:8545
3  CONTRACT_ADDRESS=0x8cF33f15d22B2b8929508c5A879EA78600EA2eE0
4  ML_API_ENDPOINT=http://127.0.0.1:5000/testFiles
5  WALLET_ADDRESS=0x5694D143eb8833D9A3D209b4F43F74300974b1A1
```

Εικόνα 32: Παράδειγμα .config αρχείου Oracle

4.4.4. Λειτουργία υπηρεσίας

Η εκκίνηση της υπηρεσίας απαιτεί μόνο την εκκίνηση του εκτελέσιμου της. Ο χρήστης οφείλει να έχει ορίσει σωστά τις διευθύνσεις στο αρχείο .config με σκοπό την εύρυθμη λειτουργία της. Επιπλέον, παρέχεται η δυνατότητα χρήσης φίλτρων για τα γεγονότα που ανιχνεύει η υπηρεσία, μέσω παροχής ορισμάτων.

Μετά την εκκίνησή της η εφαρμογή λειτουργεί αενάως, εωσότου ο χρήστης προχωρήσει στη χειροκίνητη διακοπή της. Κατά την ανίχνευση ενός γεγονότος γίνεται εκτύπωση ανάλογου μηνύματος στο τερματικό, το οποίο ενημερώνει καταλλήλως για τα περιεχόμενα του γεγονότος που ανιχνεύθηκε. Επιπλέον, κατά τη λήψη της απόφασης από τους εξυπηρετητές λήψης έξυπνων αποφάσεων και κατά την υποβολή των δεδομένων των εκτιμήσεων στην αλυσίδα, τυπώνονται επίσης κατάλληλα μηνύματα στο τερματικό.

4.5. Εξυπηρετητές Python Machine Learning API

Οι έξυπνες αποφάσεις παράγονται και παρέχονται στο υπόλοιπο πληροφοριακό σύστημα μέσω του εξυπηρετητή Python Machine Learning και του API που αυτός εκθέτει. Το τμήμα αυτό του πληροφοριακού συστήματος πραγματοποιεί την εξυπηρέτηση αιτημάτων στα endpoints που παρέχει, την παραγωγή εκτιμήσεων μέσω των μοντέλων μηχανικής μάθησης που είναι φορτωμένα σε αυτό, την ανάκτηση δεδομένων από το IPFS αλλά και την επεξεργασία τους, ώστε να είναι σε μορφή που να μπορούν τα μοντέλα να εκτελέσουν εκτιμήσεις πάνω σε αυτά. Οι επιμέρους λειτουργίες αυτές εξηγούνται παρακάτω.

4.5.1. Δημιουργία εξυπηρετητή στα πρότυπα του REST API

Το παρόν σύστημα πρέπει να λειτουργεί ως ένας εξυπηρετητής, αναμένοντας συνεχώς αιτήματα για να παρέχει εκτιμήσεις πάνω σε δεδομένα, οι οποίες πρέπει να εισαχθούν στα δεδομένα της αλυσίδας. Για το λόγο αυτό χρησιμοποιείται η βιβλιοθήκη "flask" της Python. Η βιβλιοθήκη αυτή επιτρέπει τη δημιουργία ενός εξυπηρετητή (server) σε συγκεκριμένη διεύθυνση, καθώς επίσης και τη δημιουργία επιμέρους endpoints τα οποία χρησιμοποιούνται από εξωτερικά συστήματα για να υποβάλουν τα αιτήματά τους.

4.5.2. Φόρτωση και χρήση μοντέλου μηχανικής μάθησης

Την «καρδιά» του εξυπηρετητή αποτελεί το μοντέλο μηχανικής μάθησης, το οποίο παράγει τις εκτιμήσεις. Το μοντέλο αυτό είναι προεκπαιδευμένο και παρέχεται κατά την εκκίνηση του εξυπηρετητή συμπιεσμένο, σε μορφή .pkl. Για το λόγο αυτό χρησιμοποιείται η βιβλιοθήκη "pickle" για τη φόρτωσή του. Η διεύθυνση του αρχείου .pkl το οποίο περιέχει το μοντέλο παρέχεται ως όρισμα κατά την εκκίνηση του εξυπηρετητή. Από το σημείο αυτό και έπειτα το μοντέλο χρησιμοποιεί τη βασική μέθοδο *predict()*, ώστε να παρέχει τις εκτιμήσεις του όποτε κάτι τέτοιο απαιτείται.

4.5.3. Επικοινωνία με το IPFS και ανάκτηση δεδομένων

Αναγκαία προϋπόθεση για να παράσχει ο εξυπηρετητής και τα μοντέλα τις εκτιμήσεις τους αποτελεί η επιτυχής ανάκτηση του περιεχομένου τους από το IPFS, μέσω του *ipfs_hash* που παρέχεται στα περιεχόμενα του αιτήματος. Η επικοινωνία του υποσυστήματος αυτού με τον τοπικό κόμβο IPFS που τρέχει στο τοπικό δίκτυο, συντελείται αυτή τη φορά μέσω του REST API που παρέχει ο κόμβος και όχι μέσω κάποιας ειδικής βιβλιοθήκης. Για την αποστολή των αιτημάτων χρησιμοποιείται η βιβλιοθήκη "requests" της Python, η οποία επιτρέπει στο υποσύστημα να αποστέλλει αιτήματα POST σε συγκεκριμένη διεύθυνση του κόμβου IPFS.

Η ανάκτηση γίνεται μέσω της χρήσης του endpoint *get* με κατάλληλα ορίσματα, όπως φαίνεται παρακάτω. Τα ανακτημένα δεδομένα αποθηκεύονται προσωρινά τοπικά στο υποσύστημα, αυστηρά και μόνο για το διάστημα που επεξεργάζονται από τα μοντέλα. Μετά το πέρας της επεξεργασίας τους και της εξαγωγής των εκτιμήσεων συντελείται αυτόματη διαγραφή τους, με σκοπό να μην επιβαρύνεται η μνήμη του συστήματος με περιττά δεδομένα.

4.5.4. Επεξεργασία μορφής δεδομένων

Η παροχή εκτιμήσεων από το μοντέλο μηχανικής μάθησης απαιτεί αυτά να βρίσκονται σε συγκεκριμένη μορφή. Για να επιτευχθεί αυτό συντελείται μία σειρά από τροποποιήσεις, ώστε τα αρχικά συμπιεσμένα δεδομένα που βρίσκονται σε μορφή .tar να μετατραπούν σε έναν πίνακα "pandas", στα δεδομένα του οποίου μπορεί να εκτελέσει εκτιμήσεις το μοντέλο. Το πρώτο βήμα είναι η αποσυμπίεση του αρχείο .tar, μέσω της χρήσης της βιβλιοθήκης "tarfile" της Python. Το αρχείο .csv, το οποίο περιέχεται στο συμπιεσμένο αρχείο μετονομάζεται κατάλληλα και φορτώνεται σε έναν πίνακα μέσω της συνάρτησης *read_csv()* της βιβλιοθήκης "pandas".

4.6. Διεπαφή φυλλομετρητή ReactJS Frontend Client

Για να είναι δυνατή η χρήση του πληροφοριακού συστήματος από χρήστες με βασική εξοικείωση στα πληροφοριακά συστήματα, αναπτύχθηκε επιπλέον και μία φιλική διεπαφή σε μορφή διαδικτυακής εφαρμογής, στο framework της React.js. Η διεπαφή αυτή επιτρέπει σε απλούς χρήστες να συνδέσουν το προσωπικό τους πορτοφόλι μέσω της εφαρμογής MetaMask με την εφαρμογή και να εκτελέσουν τις συναρτήσεις του έξυπνου συμβολαίου που προορίζονται για αυτούς. Συγκεκριμένα, έχουν τη δυνατότητα δημιουργίας αιτημάτων και προβολής της κατάστασής τους σε ευανάγνωστη μορφή.

4.6.1. Επικοινωνία με το έξυπνο συμβόλαιο και εκτέλεση συναρτήσεων

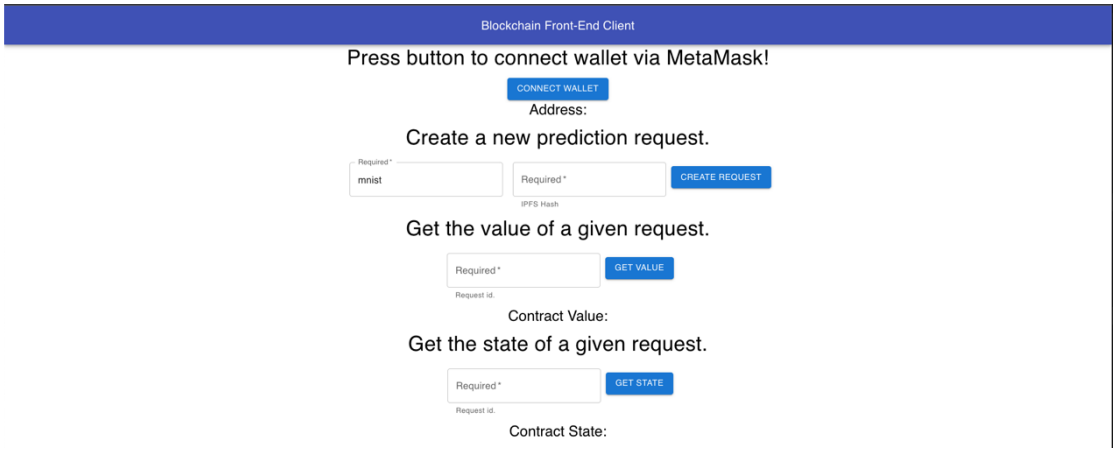
Απαραίτητη προϋπόθεση για τη χρήση της εφαρμογής από τους χρήστες και την εκτέλεση των συναρτήσεων του έξυπνου συμβολαίου αποτελεί η σύνδεση με το προσωπικό τους πορτοφόλι. Για να επιτευχθεί αυτό με τρόπο που διασφαλίζει παράλληλα τη μέγιστη δυνατή προστασία των στοιχείων του χρήστη χρησιμοποιείται η βιβλιοθήκη "ethers" της React.js και η εφαρμογή MetaMask.

Η βιβλιοθήκη "ethers" επιχειρεί να συνδέσει την εφαρμογή με το πορτοφόλι του χρήστη, το οποίο βρίσκεται σε περιβάλλον browser. Σε αυτό το περιβάλλον το πορτοφόλι του χρήστη, καθώς και όλες τις συναλλαγές που εκτελούνται από αυτό διαχειρίζεται η εφαρμογή MetaMask. Κατά την πρώτη σύνδεση του χρήστη στην εφαρμογή, πατώντας το αντίστοιχο κουμπί δίνεται η δυνατότητα επιλογής του πορτοφολιού προς σύνδεση, εφόσον αυτό έχει φορτωθεί πρώτα στο MetaMask και σύνδεσής του με την εφαρμογή. Η έγκριση του χρήστη απαιτείται παράλληλα κάθε φορά που ενδέχεται να εκτελεστεί από το

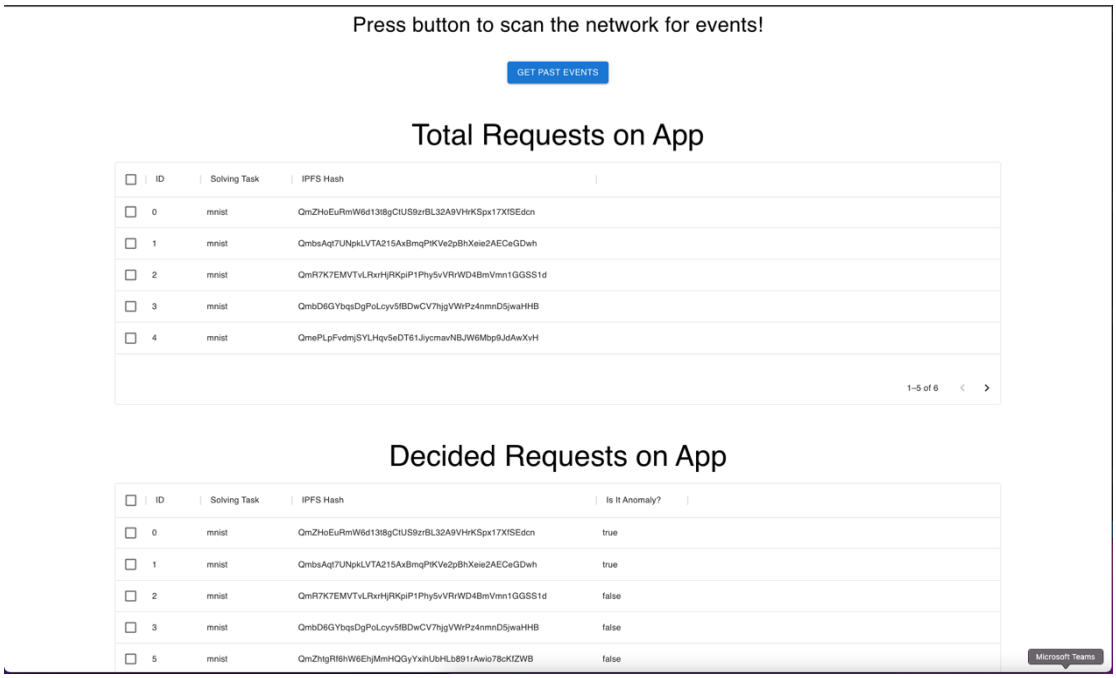
πορτοφόλι του κάποια συναλλαγή send(), η οποία πιθανώς να οδηγήσει στη χρέωσή του. Σε αυτή την περίπτωση η εφαρμογή MetaMask εμφανίζει αναδυόμενο παράθυρο, το οποίο απεικονίζει λεπτομερώς τη χρέωση με την οποία θα επιβαρυνθεί ο χρήστης και τις λοιπές λεπτομέρειες της συναλλαγής.

4.6.2. Λειτουργίες που παρέχονται από τη διαδικτυακή διεπαφή

Σε αυτό το τμήμα παρουσιάζονται αναλυτικά οι λειτουργίες που παρέχει η διαδικτυακή διεπαφή. Τονίζεται πως για την υλοποίηση των γραφικών της διεπαφής έχει χρησιμοποιηθεί η βιβλιοθήκη "material ui", η οποία παρέχει έτοιμα components με εύχρηστη λειτουργικότητα.



Εικόνα 33: Οθόνη της εφαρμογής για τις λειτουργίες σύνδεσης με πορτοφόλι και εκτέλεσης συναρτήσεων έξυπνου συμβολαίου



Εικόνα 34: Οθόνη της εφαρμογής για προεπισκόπηση γεγονότων

4.6.2.1. Σύνδεση με το πορτοφόλι του χρήστη

Η πρώτη λειτουργία που παρέχεται στο χρήστη είναι η σύνδεσή της εφαρμογής με το προσωπικό του πορτοφόλι, μέσω της επέκτασης φυλλομετρητή MetaMask. Το βήμα αυτό αποτελεί απαραίτητη προϋπόθεση για να μπορούν να εκτελεστούν όλες οι υπόλοιπες λειτουργίες της εφαρμογής, αφού απαιτούν μία προσωπική διεύθυνση του χρήστη για να επικοινωνούν με το blockchain, να εκτελούν τις συναρτήσεις του έξυπνου συμβολαίου και να αντλούν δεδομένα.

4.6.2.2. Υποβολή αιτήματος με γνωστό IPFS hash

Η διαδικτυακή εφαρμογή παρέχει στους χρήστες τη δυνατότητα υποβολής αιτημάτων προς εκτίμηση στο έξυπνο συμβόλαιο, χρησιμοποιώντας της συνάρτησή του έξυπνου συμβολαίου *createRequest*. Η εφαρμογή παρέχει τη δυνατότητα αυτή στο χρήστη μέσω της κατάλληλης φόρμας, στην οποία ο χρήστης καλείται να συμπληρώσει τη διεργασία "task", στην οποία αναφέρονται τα δεδομένα υπό εκτίμηση, καθώς και το "IPFS hash" που αντιστοιχεί στα δεδομένα. Η εφαρμογή παρέχει και στο χρήστη βοήθεια κατά τη συμπλήρωση, έχοντας προσυμπληρωμένο το πεδίο "task" με τη λέξη "mnist" που αντιστοιχεί στο πρόβλημα που πραγματεύεται η παρούσα διπλωματική. Επιπλέον, παρέχει βοήθεια έχοντας στο πεδίο "IPFS Hash" μία επεξήγηση με το περιεχόμενο που πρέπει να έχει το κελί αυτό.

4.6.2.3. Ανάκτηση κατάστασης και τιμής εκτίμησης αιτήματος

Οι χρήστες της διαδικτυακής εφαρμογής έχουν επιπλέον τη δυνατότητα να χρησιμοποιούν της δύο συναρτήσεις προβολής που παρέχει το έξυπνο συμβόλαιο. Συμπληρώνοντας ένα από τα δύο πανομοιότυπα πεδία με το αναγνωριστικό "id" του αιτήματος του οποίου τα δεδομένα θέλουν να προσπελάσουν και πατώντας το αντίστοιχο πλήκτρο μπορούν να καλέσουν τις συναρτήσεις *getState* και *getValue* του έξυπνου συμβολαίου. Τα αποτελέσματα προβάλλονται στο ανάλογο πεδίο έπειτα από την ανάκτησή τους.

4.6.2.4. Συνολική επισκόπηση αιτημάτων και ληφθέντων αποφάσεων

Η τελευταία δυνατότητα που δίνεται στους χρήστες είναι αυτή της πλήρους επισκόπησης όλων των αιτημάτων που έχουν υποβληθεί στο σύστημα προς εκτίμηση, καθώς και όλων των αποφάσεων του συστήματος. Πατώντας το αντίστοιχο πλήκτρο η εφαρμογή γεμίζει τις δύο έξυπνες δομές Dataframes με όλα τα στοιχεία των αιτημάτων αυτών και των αποφάσεων. Για να επιτευχθεί αυτό, η εφαρμογή χρησιμοποιεί επιτυχώς τα γεγονότα τα οποία εκπέμπονται κατά τη δημιουργία αιτημάτων και λήψη αποφάσεων. Με το πάτημα του πλήκτρου, αλιεύονται από το έξυπνο συμβόλαιο όλα τα γεγονότα που έχουν

εκπεμφθεί μέχρι τη στιγμή του πατήματος και γεμίζουν τα αντίστοιχα Dataframes. Οι δομές αυτές πέρα από την ευκολία στην προβολή παρέχουν και άλλες έξυπνες δυνατότητες, όπως η ταξινόμηση και η αναζήτηση με βάση συγκεκριμένα χαρακτηριστικά των δεδομένων που γεμίζουν τους πίνακες.

Κεφάλαιο 5

Έλεγχος απόδοσης αρχιτεκτονικής σε πραγματικά σενάρια

5.1. Σενάριο 1

5.1.1. Γενική περιγραφή σεναρίου

Το σενάριο αυτό θα εκτελεστεί έχοντας ως dataset αποκλειστικά τα δεδομένα των ψηφίων 5 και 6, τα οποία και παρουσιάζουν τη μεγαλύτερη ομοιότητα μεταξύ τους από τα δεδομένα του dataset, με σκοπό το πρόβλημα να διατηρεί έναν ικανοποιητικό για εξαγωγή συμπερασμάτων βαθμό δυσκολίας.

Στο σενάριο αυτό τα δείγματα του ψηφίου 5 θα θεωρείται πως αποτελούν φυσιολογικά δείγματα (inliers) και τα δείγματα που αντιστοιχούν στον αριθμό 6 ανωμαλίες (outliers). Το πείραμα θα εκτελεστεί για διάφορα μεγέθη δειγμάτων εκπαίδευσης (μικρό – 250 δείγματα, μεσαίο – 500 δείγματα, μεγάλο – 1000 δείγματα) και για διάφορες αναλογίες outliers (1%, 2.5%, 5%, 10%).

5.1.2. Στόχος του σεναρίου

Στόχος του παρόντος σεναρίου είναι να διερευνηθεί η απόδοση της κατανεμημένης αρχιτεκτονικής αναγνώρισης έκτοπων δεδομένων σε μοντέλα τα οποία έχουν εκπαιδευθεί με ανεξάρτητα αλλά παρόμοια ως προς τον τύπο των ανωμαλιών που διαχειρίζονται δεδομένα.

Μέσω του παρόντος πειράματος αποσκοπείται ο προσδιορισμός του τύπου αυτού των προβλημάτων στα οποία η παρούσα αρχιτεκτονική παρουσιάζει βέλτιστη απόδοση. Επιπλέον, σκοπός του πειράματος αυτού είναι να αναδειχθεί η τάση που δείχνει η συνολική απόφαση να προσεγγίζει τη συμπεριφορά των βέλτιστων μοντέλων που συμμετέχουν στη διαδικασία απόφασης.

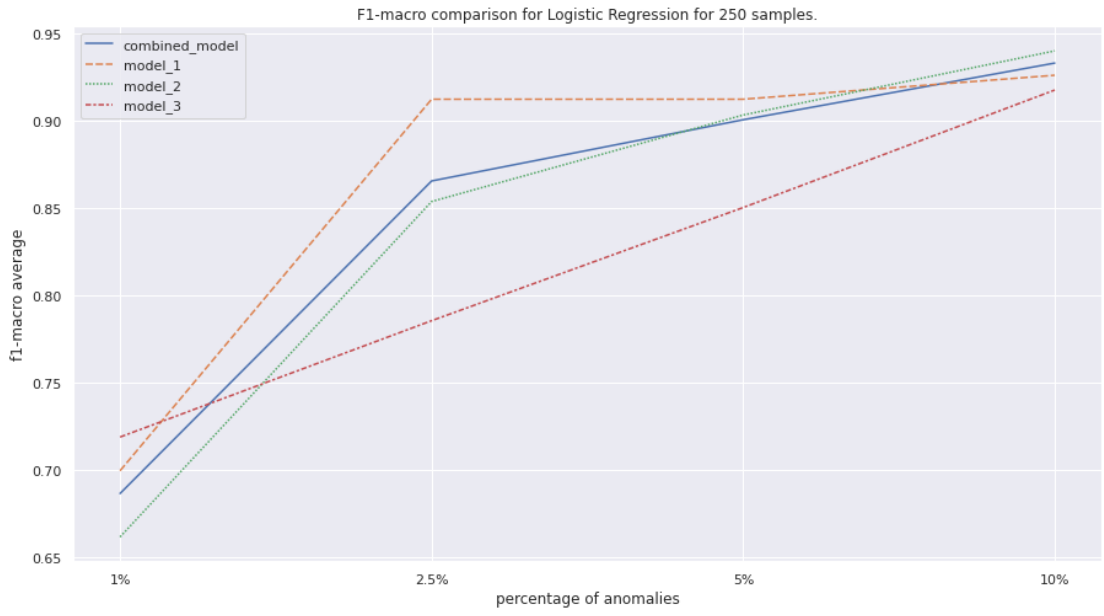
5.1.3. Περιγραφή διαδικασίας εκτέλεσης σεναρίου

Για την εκτέλεση του παρόντος σεναρίου θα χρησιμοποιηθούν σε μεγάλο βαθμό scripts που αυτοματοποιούν τη διαδικασία. Συγκεκριμένα η ροή της διαδικασίας είναι η εξής:

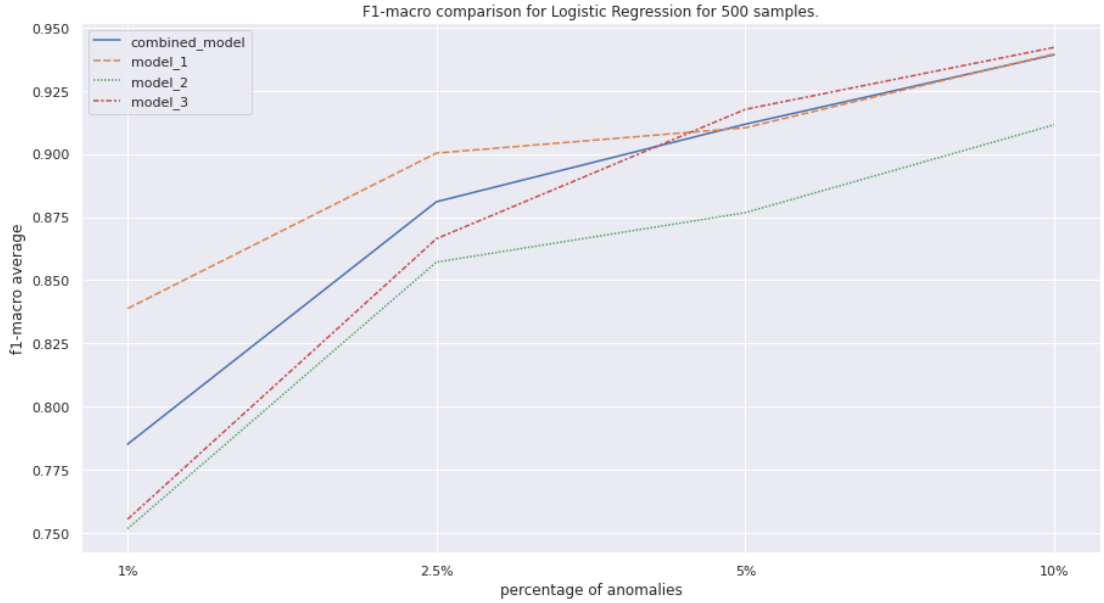
- Δημιουργία ενός νέου έξυπνου συμβολαίου στο δίκτυο.
- Εκπαίδευση νέων μοντέλων στο σύνολο δεδομένων σύμφωνα με τις προδιαγραφές του εκάστοτε σεναρίου και εξαγωγή του συνόλου δεδομένων ελέγχου και των εκτιμήσεων των μοντέλων ατομικά.
- Φόρτωση των μοντέλων στο σύστημα.
- Εκτέλεση πολλαπλών αιτημάτων createRequest από ένα τερματικό για το σύνολο των δεδομένων ελέγχου.
- Εκτέλεση πολλαπλών αιτημάτων getValue από ένα τερματικό για το σύνολο των αποφάσεων του συστήματος πάνω στα δεδομένα ελέγχου και ανακατεύθυνση της εξόδου σε αρχείο κειμένου.
- Εξαγωγή της classification report από το αρχείο κειμένου συγκρίνοντας τα αποτελέσματα με τις πραγματικές ετικέτες.
- Σύγκριση της κοινής απόφασης με τις αποφάσεις των επιμέρους μοντέλων.

5.1.4. Παρουσίαση και ανάλυση αποτελεσμάτων

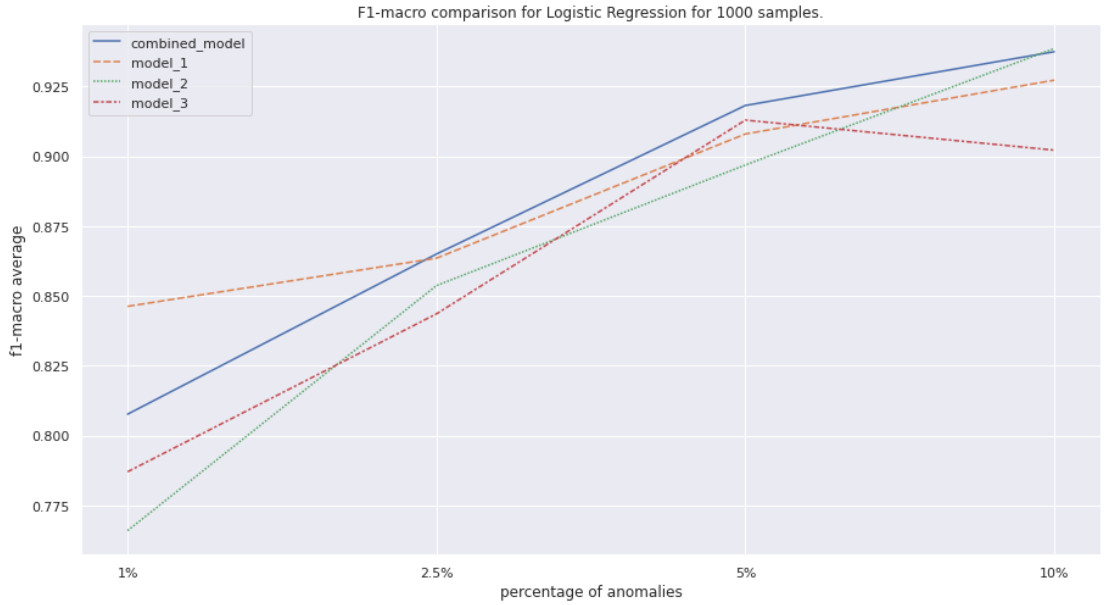
Παρακάτω παρουσιάζονται τρία διαδοχικά διαγράμματα που απεικονίζουν την απόδοση των τριών επιμέρους μοντέλων, καθώς και του συνδυαστικού μοντέλου στην αναγνώριση ανωμαλιών στο σύνολο δεδομένων που περιγράφηκε νωρίτερα, υπό διαφορετικές κάθε φορά συνθήκες [Διάγραμμα 18-20]. Η μπλε γραμμή αντιστοιχεί στο συνδυαστικό μοντέλο, ενώ τα άλλα τρία χρώματα στα επιμέρους μοντέλα. Καθένα από τα διαγράμματα απεικονίζει την απόδοση στη μετρική “f1-macro average” των μοντέλων λογιστικής παλινδρόμησης με τον άξονα των y να αντιστοιχεί στην αντίστοιχη επίδοση και τον άξονα x στο ποσοστό των ανωμαλιών στο οποίο είχαν προεκπαιδευθεί τα μοντέλα.



Διάγραμμα 18: Απεικόνιση απόδοσης μοντέλων λογιστικής παλινδρόμησης εκπαιδευμένων σε σύνολο εκπαίδευσης 250 δειγμάτων



Διάγραμμα 19: Απεικόνιση απόδοσης μοντέλων λογιστικής παλινδρόμησης εκπαιδευμένων σε σύνολο εκπαίδευσης 500 δειγμάτων



Διάγραμμα 20: Απεικόνιση απόδοσης μοντέλων λογιστικής παλινδρόμησης εκπαιδευμένων σε σύνολο εκπαίδευσης 1000 δειγμάτων

Από την ανάλυση των παραπάνω αποτελεσμάτων καταλήγουμε στα εξής συμπεράσματα.

Αρχικά η συμπεριφορά του συνδυαστικού μοντέλου φαίνεται να είναι τέτοια που να εκμεταλλεύεται τα καλύτερα κάθε φορά μοντέλα με σκοπό να παράξει τις καλύτερες εκτιμήσεις. Αυτό είναι εμφανές από το γεγονός πως σε καμία από τις περιπτώσεις που ελέγχθηκαν το συνδυαστικό μοντέλο δεν πέτυχε συνολικά χειρότερη απόδοση από το

χειρότερο εκ των τριών μοντέλων. Μάλιστα στην πλειονότητα των περιπτώσεων το συνδυαστικό μοντέλο είτε προσεγγίζει το βέλτιστο μοντέλο, είτε το ξεπερνά.

Παράλληλα από τα διαγράμματα παρατηρούνται και κάποιες τάσεις ως προς τη συμπεριφορά του συνδυαστικού μοντέλου. Συγκεκριμένα παρατηρείται πως καθώς το μέγεθος του συνόλου εκπαίδευσης αυξάνεται, αυξάνεται και η κατάταξη του συνδυαστικού μοντέλου ως προς τα υπόλοιπα μοντέλα. Επιπλέον, ανάλογη συμπεριφορά αύξησης της απόδοσης και συνεπώς και της κατάταξης του συνδυαστικού μοντέλου παρατηρείται και κατά την αύξηση του ποσοστού των ανωμαλιών στο σύνολο δεδομένων.

Οι παραπάνω παρατηρήσεις μας οδηγούν στο συμπέρασμα πως η προτεινόμενη αρχιτεκτονική συμπεριφέρεται βέλτιστα σε καταστάσεις προβλημάτων, όπου τα επιμέρους μοντέλα βρίσκονται σε μία ώριμη από πλευράς εκπαίδευσης φάση, έχοντας παράλληλα ικανοποιητική εμπειρία από έκτοπα δεδομένα.

5.2. Σενάριο 2

5.2.1. Γενική περιγραφή σεναρίου

Στο σενάριο αυτό το σύνολο δεδομένων αποτελείται από τέσσερα διακριτά ψηφία, τα ψηφία 3,5,6 και 8. Το ψηφίο 8, όντας εκείνο που παρουσιάζει τις περισσότερες κοινές πλευρές με όλα τα υπόλοιπα, αφού ουσιαστικά επικαλύπτει όλες τις πλευρές τους θα αποτελέσει το ψηφίο που λογίζεται ως κανονικό δεδομένο. Τα άλλα τρία ψηφία θα αποτελέσουν τις ανωμαλίες, οι οποίες όμως θα είναι διαφορετικές για καθένα από τα σύνολα εκπαίδευσης.

Στο τελικό σύνολο δεδομένων ελέγχου θα υπάρχουν δείγματα και από τα τρία έκτοπα ψηφία, με σκοπό να ελεγχθεί η απόδοση της προτεινόμενης αρχιτεκτονικής σε ένα σενάριο όπου κάποια από τα μοντέλα καλούνται για πρώτη φορά να εκτελέσουν εκτιμήσεις πάνω σε μία νέα για αυτά μορφή έκτοπων δεδομένων.

Το πείραμα θα εκτελεστεί για διάφορα μεγέθη δειγμάτων εκπαίδευσης (μικρό – 250 δείγματα, μεσαίο – 500 δείγματα, μεγάλο – 1000 δείγματα) και για διάφορες αναλογίες outliers (1%, 2.5%, 5%, 10%).

5.2.2. Στόχος του σεναρίου

Το σενάριο αυτό έχει σκοπό να αναπαραστήσει τύπους ανωμαλιών που παρουσιάζουν μία γεωγραφική εξάπλωση, όπως για παράδειγμα ένα είδος απάτης ή μία μετάλλαξη κάποιας ασθένειας. Στόχος είναι να ελεγχθεί η συμπεριφορά της προτεινόμενης αρχιτεκτονικής στον εναλλακτικό αυτό τύπου προβλήματος αναγνώρισης ανωμαλιών, ο οποίο ανταποκρίνεται σε μια συγκεκριμένη κατηγορία πραγματικών προβλημάτων, πιο απαιτητική του Σεναρίου 1.

Μέσω της εκτέλεσης του παρόντος σεναρίου αποσκοπείται να βρεθεί η σύσταση των παραγόντων του προβλήματος (μέγεθος συνόλου δεδομένων – ποσοστό ανωμαλιών στο σύνολο δεδομένων), στα οποία η αρχιτεκτονική αυτή φαίνεται να παρουσιάζει

καλύτερη απόδοση, παρέχοντας συνολικά καλύτερες προβλέψεις ως προς τις ατομικές προβλέψεις της πλειοψηφίας των κόμβων απόφασης και των αντίστοιχων μοντέλων τους.

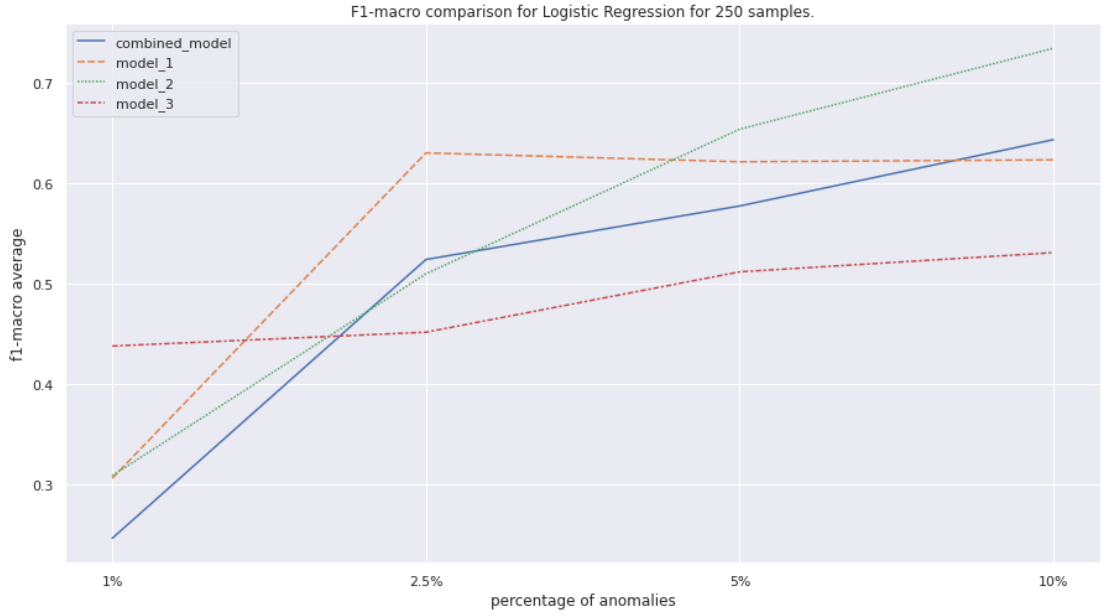
5.2.3. Περιγραφή διαδικασίας εκτέλεσης σεναρίου

Για την εκτέλεση του παρόντος σεναρίου θα χρησιμοποιηθούν σε μεγάλο βαθμό scripts που αυτοματοποιούν τη διαδικασία. Συγκεκριμένα η ροή της διαδικασίας είναι η εξής:

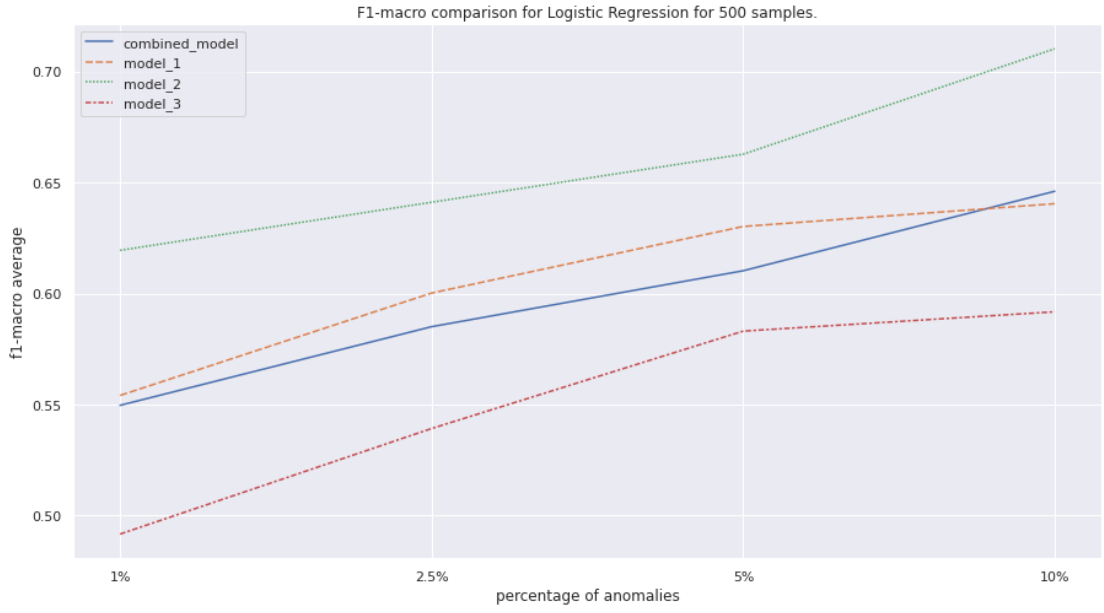
- Δημιουργία ενός νέου έξυπνου συμβολαίου στο δίκτυο.
- Εκπαίδευση νέων μοντέλων στο σύνολο δεδομένων σύμφωνα με τις προδιαγραφές του εκάστοτε σεναρίου και εξαγωγή του συνόλου δεδομένων ελέγχου και των εκτιμήσεων των μοντέλων ατομικά.
- Φόρτωση των μοντέλων στο σύστημα.
- Εκτέλεση πολλαπλών αιτημάτων createRequest από ένα τερματικό για το σύνολο των δεδομένων ελέγχου.
- Εκτέλεση πολλαπλών αιτημάτων getValue από ένα τερματικό για το σύνολο των αποφάσεων του συστήματος πάνω στα δεδομένα ελέγχου και ανακατεύθυνση της εξόδου σε αρχείο κειμένου.
- Εξαγωγή της classification report από το αρχείο κειμένου συγκρίνοντας τα αποτελέσματα με τις πραγματικές ετικέτες.
- Σύγκριση της κοινής απόφασης με τις αποφάσεις των επιμέρους μοντέλων.

5.2.4. Παρουσίαση και ανάλυση αποτελεσμάτων

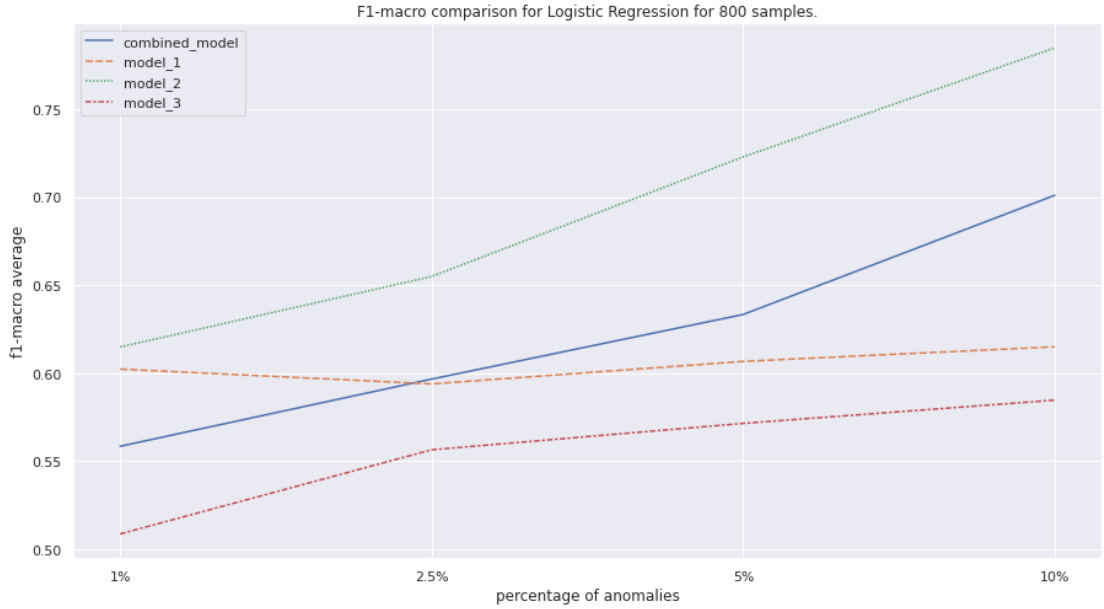
Παρακάτω παρουσιάζονται πέντε διαδοχικά διαγράμματα που απεικονίζουν την απόδοση των τριών επιμέρους μοντέλων, καθώς και του συνδυαστικού μοντέλου στην αναγνώριση ανωμαλιών στο σύνολο δεδομένων που περιγράφηκε νωρίτερα, υπό διαφορετικές κάθε φορά συνθήκες [Διάγραμμα 21-25]. Η μπλε γραμμή αντιστοιχεί στο συνδυαστικό μοντέλο, ενώ τα άλλα τρία χρώματα στα επιμέρους μοντέλα. Καθένα από τα διαγράμματα απεικονίζει την απόδοση στη μετρική “f1-macro average” των μοντέλων λογιστικής παλινδρόμησης με τον άξονα των y να αντιστοιχεί στην αντίστοιχη επίδοση και τον άξονα x στο ποσοστό των ανωμαλιών στο οποίο είχαν προεκπαιδευθεί τα μοντέλα.



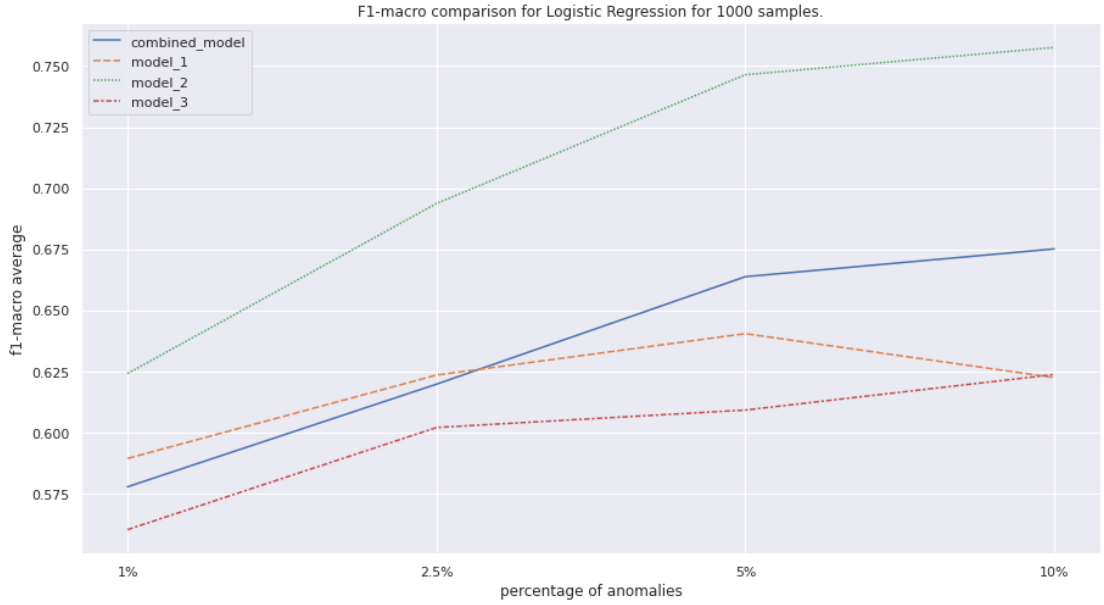
Διάγραμμα 21: Απεικόνιση απόδοσης μοντέλων λογιστικής παλινδρόμησης εκπαιδευμένων σε σύνολο εκπαίδευσης 250 δειγμάτων σε σενάριο πολλαπλών τύπων ανωμαλιών



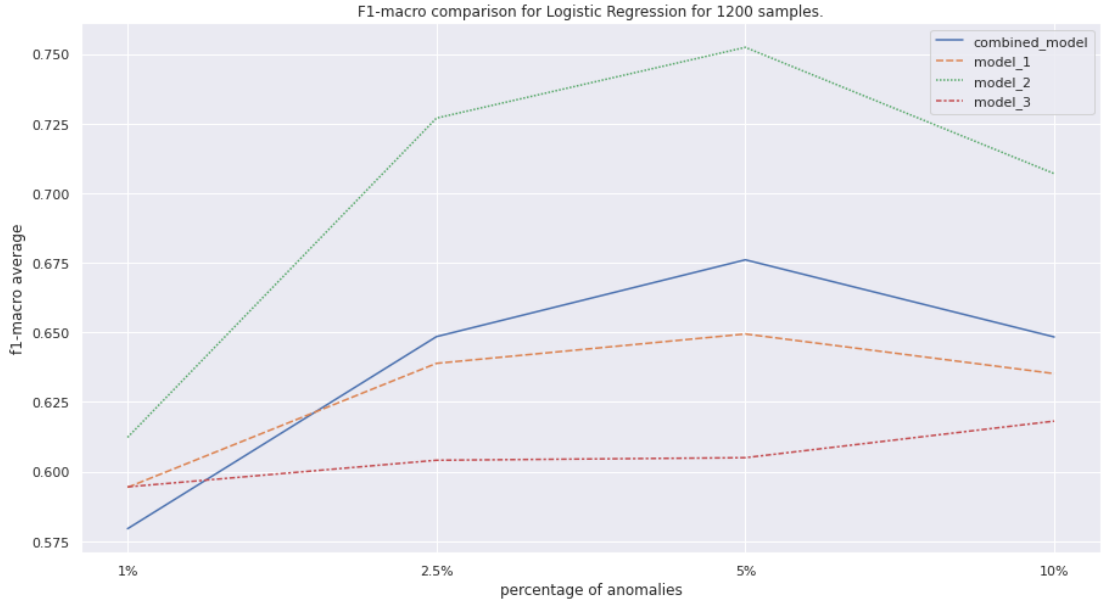
Διάγραμμα 22: Απεικόνιση απόδοσης μοντέλων λογιστικής παλινδρόμησης εκπαιδευμένων σε σύνολο εκπαίδευσης 500 δειγμάτων σε σενάριο πολλαπλών τύπων ανωμαλιών



Διάγραμμα 23: Απεικόνιση απόδοσης μοντέλων λογιστικής παλινδρόμησης εκπαιδευμένων σε σύνολο εκπαίδευσης 800 δειγμάτων σε σενάριο πολλαπλών τύπων ανωμαλιών



Διάγραμμα 24: Απεικόνιση απόδοσης μοντέλων λογιστικής παλινδρόμησης εκπαιδευμένων σε σύνολο εκπαίδευσης 1000 δειγμάτων σε σενάριο πολλαπλών τύπων ανωμαλιών



Διάγραμμα 25: Απεικόνιση απόδοσης μοντέλων λογιστικής παλινδρόμησης εκπαιδευμένων σε σύνολο εκπαίδευσης 1200 δειγμάτων σε σενάριο πολλαπλών τύπων ανωμαλιών

Όπως αποδεικνύεται από τη γενική επίδοση των μοντέλων στο συγκεκριμένο σενάριο, το πρόβλημα αναγνώρισης πολλαπλών τύπων ανωμαλιών είναι πιο απαιτητικό από αυτό της συγκεκριμένης ανωμαλίας, ιδιαίτερα όταν το κάθε επιμέρους μοντέλο έχει εκπαιδευθεί σε έναν τύπο ανωμαλιών.

Όσον αφορά την επίδοση του συνδυαστικού μοντέλου, στο σενάριο αυτό παρατηρείται πως η συνδυαστική απόφαση υστερεί της ατομικής απόφασης των μοντέλων σε περιπτώσεις μικρών συνόλων εκπαίδευσης και ιδιαίτερα στην περίπτωση χαμηλού ποσοστού ανωμαλιών στα σύνολα εκπαίδευσης. Παρ ‘όλα αυτά εξακολουθεί να παρατηρείται αύξηση της απόδοσης του συνδυαστικού μοντέλου, τόσο κατά απόλυτο βαθμό όσο και όσον αφορά την κατάταξή του συγκριτικά με τα υπόλοιπα, καθώς αυξάνεται το ποσοστό ανωμαλιών που συμπεριλαμβάνονται στο σύνολο εκπαίδευσης.

Ιδιαίτερα ενδιαφέρον αποτελεί το γεγονός πως τόσο η απόδοση του συνδυαστικού μοντέλου, όσο και των επιμέρους μοντέλων μειώνεται στην περίπτωση του μεγάλου συνόλου δεδομένων με τα 1200 δείγματα, όταν αυτό περιλαμβάνει ανωμαλίες σε ποσοστό 10%. Αυτό μπορεί να εξηγηθεί από το γεγονός πως χρησιμοποιούνται γραμμικά μοντέλα, όπως αυτό της λογιστικής παλινδρόμησης για το διαχωρισμό κανονικών και έκτοπων δεδομένων, τα οποία φαίνεται να προσαρμόζονται υπερβολικά στο διαχωρισμό των δύο τύπων δεδομένων που λαμβάνουν κατά την εκπαίδευση και να είναι λιγότερο δεκτικά σε νέου τύπου ανωμαλίες. Η ιδιαιτερότητα αυτή στη συμπεριφορά των μοντέλων έγκειται στην ίδια τη φύση τους και πιθανόν να μην παρατηρείται κατά τη χρήση εναλλακτικών μοντέλων απόφασης που εκτελούν το διαχωρισμό με διαφορετικό μηχανισμό. Επιπλέον, πιθανή λύση για την εξάλειψη αυτού του παράδοξου φαινομένου θα μπορούσε να αποτελέσει η χρήση ενός διαφορετικού τύπου απόφασης για το συνδυαστικό μοντέλο που θα προσμετρούσε το επίπεδο βεβαιότητας του μοντέλου για την εκτίμηση που παρέχει.

Συνοψίζοντας την ανάλυση του Σεναρίου 2, παρατηρούμε πως το συνδυαστικό μοντέλο που προτείνεται στην παρούσα διπλωματική πετυχαίνει ικανοποιητικές επιδόσεις και στο πρόβλημα των πολλαπλών τύπων ανωμαλιών,, ιδιαίτερα σε καθεστώτα συνόλων

εκπαίδευσης μεσαίου μεγέθους (500 – 1000 δείγματα) που περιέχουν ανωμαλίες σε ποσοστό 2.5% - 5%.

Κεφάλαιο 6

Συμπεράσματα και περιθώρια για περαιτέρω έρευνα

Κατά την ολοκλήρωση της παρούσας διπλωματικής εργασίας αφιερώνεται ένα τμήμα, ώστε να αναλυθούν οι ευκαιρίες για έρευνα που δημιουργούνται μετά το πέρας της παρούσας έρευνας και της ανάλυσης των αποτελεσμάτων της. Άλλωστε, η αρθρωτή αρχιτεκτονική του πληροφοριακού συστήματος επιτρέπει την παρέμβαση σε επιμέρους τμήματα με ελάχιστες προσαρμογές, ευνοεί συνεπώς την έρευνα.

6.1. Εξερεύνηση εναλλακτικών τύπων blockchains

Η αντικατάσταση του Ethereum με έναν διαφορετικό τύπο blockchain μπορεί να επιφέρει σημαντικές αλλαγές στο σύστημα χρησιμοποιώντας έναν πιο κλειστό ως προς τα δικαιώματα πρόσβασης τύπο blockchain, όπως ένα permissioned blockchain, δύναται να αυξήσει την ιδιωτικότητα και την ασφάλεια των δεδομένων που εισάγονται στο blockchain και παράλληλα να αποφορτίσει το έξυπνο συμβόλαιο από την ευθύνη του ελέγχου των δικαιωμάτων των κόμβων κατά την εκτέλεση των συναρτήσεων των έξυπνων συμβολαίων.

Ένα άλλο πεδίο στο οποίο υστερεί το Ethereum έναντι πιθανών εναλλακτικών είναι το ενεργειακό του αποτύπωμα. Ο όγκος των αναμενόμενων αιτημάτων αν το πληροφοριακό σύστημα τεθεί σε ισχύ σε μία εφαρμογή που διασυνδέει πληθώρα τραπεζικών συστημάτων υπαγορεύει την ανάγκη αναζήτησης εναλλακτικών τύπων blockchains που ελαχιστοποιούν το ενεργειακό αποτύπωμα του πληροφοριακού συστήματος.

6.2. Εξερεύνηση εναλλακτικών πρωτοκόλλων συμφωνίας

Στα πλαίσια της παρούσας διπλωματικής ελέγχθηκαν σενάρια, στα οποία οι κόμβοι είχαν ισότιμη και ακέραιη ψήφο. Συνεπώς δεν υπήρχε κάποια ένδειξη του επιπέδου της βεβαιότητας του κάθε κόμβου για την εκτίμηση που αυτός παρείχε στο δίκτυο, μέσω του συστήματος μηχανικής μάθησης που είχε στη διάθεσή του. Περαιτέρω έρευνα θα μπορούσε να εξερευνήσει τη χρήση διαφορετικών πρωτοκόλλων συμφωνίας, στα οποία ο κόμβος μπορεί να εισάγει στο δίκτυο μαζί με την εκτίμησή του και το ποσοστό βεβαιότητάς του για την εκτίμηση αυτή και αυτό να συνυπολογίζεται κατά την παραγωγή της τελικής συνολικής εκτίμησης.

6.3. Εξερεύνηση της απόδοσης πιο σύνθετων μοντέλων μηχανικής μάθησης σε πιο απαιτητικά σενάρια

Η παρούσα διπλωματική εστίασε στον έλεγχο της συνολικής αρχιτεκτονικής με απλά μοντέλα μηχανικής μάθησης σε ένα σχετικά απλό σενάριο ελέγχου. Η εξερεύνηση της χρήσης διαφόρων τύπων μοντέλων απόφασης σε πιθανώς πιο απαιτητικά και ρεαλιστικά σενάρια μπορεί να αποτελέσει αντικείμενο έρευνας που θα πιστοποιήσει περαιτέρω τις συνολικές δυνατότητες της προτεινόμενης αρχιτεκτονικής.

6.4. Συμπεράσματα

Με την ολοκλήρωση της παρούσας διπλωματικής και της έρευνας που συντελέστηκε στα πλαίσια της φθάνουμε στα ακόλουθα συμπεράσματα ως προς τις προτάσεις που περιλαμβάνει.

Αρχικά, η συνολική προτεινόμενη αρχιτεκτονική φαίνεται να λειτουργεί επιτυχημένα και να ανταποκρίνεται στην επίλυση του είδους των προβλημάτων για το οποίο προορίζεται. Η επικοινωνία των συστημάτων συντελείται ορθά και ταχύτατα, ενώ στα πλαίσια του ελέγχου που συντελέστηκε κατά την ανάπτυξη δεν παρατηρήθηκε κάποιο υποσύστημα στενωπός, ούτε και ιδιαίτερη επιβάρυνση κάποιου από τα υποσυστήματα με ογκώδη δεδομένα. Έχουμε λοιπόν κάθε λόγο να αναμένουμε πως το σύνολο της προτεινόμενης αρχιτεκτονικής θα είναι ικανό να ανταποκριθεί και σε ρεαλιστικά, μη ερευνητικά, περιβάλλοντα σε συνθήκες αντιμετώπισης πραγματικών προβλημάτων.

Όσον αφορά τις επιδόσεις της προτεινόμενης αρχιτεκτονικής και των μοντέλων που χρησιμοποιήθηκαν στα σενάρια ελέγχου, επιβεβαιώθηκε η αρχική εκτίμηση, ότι η απόφαση του συνόλου συγκλίνει στις βέλτιστες, παρά στις χείριστες, ατομικές αποφάσεις των μοντέλων που συνδράμουν σε αυτή. Παράλληλα, καταλήξαμε στο συμπέρασμα πως ακόμα και στη γενική κατηγορία των προβλημάτων αναγνώρισης ανωμαλιών, υπάρχουν ειδικές υποκατηγορίες, όσον αφορά το μέγεθος των διαθέσιμων δεδομένων προς εκπαίδευση και τη σύστασή τους σε αναλογία κανονικών δεδομένων και ανωμαλιών, στα οποία η προτεινόμενη αρχιτεκτονική ταιριάζει περισσότερο, πετυχαίνοντας αποτελέσματα που υπερτερούν καθενός από τα ατομικά μοντέλα.

Βιβλιογραφία

1. A. S. Tanenbaum and M. van Steen, *Distributed systems: principles and paradigms*. Upper Saddle River, N.J: Prentice Hall, 2002.
2. K. Christidis and M. Devetsikiotis, “Blockchains and Smart Contracts for the Internet of Things,” *IEEE Access*, vol. 4, pp. 2292–2303, 2016, doi: 10.1109/ACCESS.2016.2566339.
3. D. Puthal, N. Malik, S. P. Mohanty, E. Kougianos, and G. Das, “Everything You Wanted to Know About the Blockchain: Its Promise, Components, Processes, and Problems,” *IEEE Consumer Electron. Mag.*, vol. 7, no. 4, pp. 6–14, 2018, doi: 10.1109/MCE.2018.2816299.
4. S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system.” 2009.
5. S. Zhang and J.-H. Lee, “Analysis of the main consensus protocols of blockchain,” *ICT Express*, vol. 6, no. 2, pp. 93–97, 2020, doi: 10.1016/j.icte.2019.08.001.
6. S. Gilbert and N. Lynch, “Brewer’s conjecture and the feasibility of consistent, available, partition-tolerant web services,” *SIGACT News*, vol. 33, no. 2, pp. 51–59, 2002, doi: 10.1145/564585.564601.
7. L. Lamport, R. Shostak, and M. Pease, “The Byzantine Generals Problem,” *ACM Trans. Program. Lang. Syst.*, vol. 4, no. 3, pp. 382–401, 1982, doi: 10.1145/357172.357176.
8. E. Foundation, “On Public and Private Blockchains.” https://blog.ethereum.org/2015/08/07/on-public-and-private-blockchains/ (accessed May 19, 2022).
9. “Ethereum Whitepaper,” *ethereum.org*. https://ethereum.org (accessed May 19, 2022).
10. H. Al-Breiki, M. H. U. Rehman, K. Salah, and D. Svetinovic, “Trustworthy Blockchain Oracles: Review, Comparison, and Open Research Challenges,” *IEEE Access*, vol. 8, pp. 85675–85685, 2020, doi: 10.1109/ACCESS.2020.2992698.
11. *Business Process Management: Blockchain and Robotic Process Automation Forum*. Accessed: May 19, 2022. [Online]. Available: https://link.springer.com/book/10.1007/978-3-030-58779-6
12. “What Is an Oracle in Blockchain? » Explained | Chainlink.” https://chain.link/education/blockchain-oracles (accessed May 19, 2022).
13. A. Beniiche, “A Study of Blockchain Oracles,” *arXiv:2004.07140 [cs]*, Jul. 2020, Accessed: May 19, 2022. [Online]. Available: http://arxiv.org/abs/2004.07140
14. G. Caldarelli, “Real-world blockchain applications under the lens of the oracle problem. A systematic literature review,” in *2020 IEEE International Conference on Technology Management, Operations and Decisions (ICTMOD)*, Marrakech, Morocco, Nov. 2020, pp. 1–6. doi: 10.1109/ICTMOD49425.2020.9380598.

15. G. Caldarelli, “Understanding the Blockchain Oracle Problem: A Call for Action,” *Information*, vol. 11, no. 11, p. 509, Oct. 2020, doi: 10.3390/info11110509.
16. T. Cioara, C. Pop, R. Zanc, I. Anghel, M. Antal, and I. Salomie, “Smart Grid Management Using Blockchain: Future Scenarios and Challenges,” in *2020 19th RoEduNet Conference: Networking in Education and Research (RoEduNet)*, Bucharest, Romania, Dec. 2020, pp. 1–5. doi: 10.1109/RoEduNet51892.2020.9324874.
17. “IPFS Powers the Distributed Web.” https://ipfs.io/#why (accessed May 19, 2022).
18. Q. Zheng, Y. Li, P. Chen, and X. Dong, “An Innovative IPFS-Based Storage Model for Blockchain,” in *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, Santiago, 2018, pp. 704–708. doi: 10.1109/WI.2018.000-8.
19. A. M. Turing, “I.—COMPUTING MACHINERY AND INTELLIGENCE,” *Mind*, vol. LIX, no. 236, pp. 433–460, Oct. 1950, doi: 10.1093/mind/LIX.236.433.
20. P. McCorduck and C. Cfe, *Machines Who Think: A Personal Inquiry into the History and Prospects of Artificial Intelligence*. CRC Press, 2004.
21. De Spiegeleire, S., Maas, M. and Sweijs, T., *Artificial Intelligence and the Future of Defense*, 2017. [online] Google Books. Available at: <https://books.google.com/books/about/Artificial_Intelligence_and_the_Future_o.html?id=xZUnDwAAQBAJ> [Accessed 20 May 2022].
22. “What is Machine Learning?” https://www.ibm.com/cloud/learn/machine-learning (accessed May 19, 2022).
23. V. Barnett and T. Lewis, *Outliers in statistical data*, 3rd ed. Chichester ; New York: Wiley, 1994.
24. R. Domingues, M. Filippone, P. Michiardi, and J. Zouaoui, “A comparative evaluation of outlier detection algorithms: Experiments and analyses,” *Pattern Recognition*, vol. 74, pp. 406–421, Feb. 2018, doi: 10.1016/j.patcog.2017.09.037.
25. T. K. Moon, “The expectation-maximization algorithm,” *IEEE Signal Process. Mag.*, vol. 13, no. 6, pp. 47–60, Nov. 1996, doi: 10.1109/79.543975.
26. O. Alghushairy, R. Alsini, T. Soule, and X. Ma, “A Review of Local Outlier Factor Algorithms for Outlier Detection in Big Data Streams,” *BDCC*, vol. 5, no. 1, p. 1, Dec. 2020, doi: 10.3390/bdcc5010001.
27. Escalante, H., *A comparison of outlier detection algorithms for machine learning.* 2005. [online] Programming and Computer Software. Available at: <https://www.researchgate.net/profile/Hugo-Jair-Escalante/publication/228728521_A_comparison_of_outlier_detection_algorithms_for_machine_learning/links/0912f50b777c20ab5e000000/A-comparison-of-outlier-detection-algorithms-for-machine-learning.pdf> [Accessed 20 May 2022].
28. H. Abdi and L. J. Williams, “Principal component analysis: Principal component analysis,” *WIREs Comp Stat*, vol. 2, no. 4, pp. 433–459, 2010, doi: 10.1002/wics.101.
29. R. V. Banu and N. Nagaveni, "Preservation of Data Privacy Using PCA Based Transformation," 2009 International Conference on Advances in Recent Technologies in Communication and Computing, 2009, pp. 439-443, doi: 10.1109/ARTCom.2009.159.

30. “Architectural Styles and the Design of Network-based Software Architectures.” https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm (accessed May 19, 2022).
31. Xinyang Feng, Jianjing Shen and Ying Fan, "REST: An alternative to RPC for Web services architecture," 2009 First International Conference on Future Information Networks, 2009, pp. 7-10, doi: 10.1109/ICFIN.2009.5339611.
32. R. T. Fielding and J. Reschke, “Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content,” Internet Engineering Task Force, Request for Comments RFC 7231, Jun. 2014. Accessed: May 19, 2022. [Online]. Available: https://datatracker.ietf.org/doc/rfc7231/
33. E. Karger, “Combining Blockchain and Artificial Intelligence – Literature Review and State of the Art,” *ICIS 2020 Proceedings*, Dec. 2020, [Online]. Available: https://aisel.aisnet.org/icis2020/blockchain_fintech/blockchain_fintech/6
34. K. Salah, M. H. U. Rehman, N. Nizamuddin, and A. Al-Fuqaha, “Blockchain for AI: Review and Open Research Challenges,” *IEEE Access*, vol. 7, pp. 10127–10149, 2019, doi: 10.1109/ACCESS.2018.2890507.
35. H. Subramanian, “Decentralized blockchain-based electronic marketplaces,” *Commun. ACM*, vol. 61, no. 1, pp. 78–84, Dec. 2017, doi: 10.1145/3158333.
36. G. Palaiokrassas *et al.*, “Combining Blockchains, Smart Contracts, and Complex Sensors Management Platform for Hyper-Connected SmartCities: An IoT Data Marketplace Use Case,” *Computers*, vol. 10, no. 10, p. 133, Oct. 2021, doi: 10.3390/computers10100133.
37. D. Chalmers, N. G. MacKenzie, and S. Carter, “Artificial Intelligence and Entrepreneurship: Implications for Venture Creation in the Fourth Industrial Revolution,” *Entrepreneurship Theory and Practice*, vol. 45, no. 5, pp. 1028–1053, 2021, doi: 10.1177/1042258720934581.
38. E. C. Ferrer, “The blockchain: a new framework for robotic swarm systems,” 2016, doi: 10.48550/ARXIV.1608.00695.
39. F. Corea, *Applied artificial intelligence: where AI can be used in business*. New York, NY: Springer Berlin Heidelberg, 2018.
40. A. Rai, “Explainable AI: from black box to glass box,” *J. of the Acad. Mark. Sci.*, vol. 48, no. 1, pp. 137–141, Jan. 2020, doi: 10.1007/s11747-019-00710-5.
41. A. S. Almasoud, M. M. Eljazzar, and F. Hussain, “Toward a Self-Learned Smart Contracts,” in *2018 IEEE 15th International Conference on e-Business Engineering (ICEBE)*, Xi’an, 2018, pp. 269–273. doi: 10.1109/ICEBE.2018.00051.
42. Nguyen H. and S. Hafid, “Use of Artificial Intelligence for Smart Contracts and Blockchains,” in *FinTechLaw Report : E-Banking, Payments and Commerce in the Mobile World*, 2018.
43. H. J. Singh and A. S. Hafid, “Prediction of Transaction Confirmation Time in Ethereum Blockchain Using Machine Learning,” in *Blockchain and Applications*, vol. 1010, J. Prieto, A. K. Das, S. Ferretti, A. Pinto, and J. M. Corchado, Eds. Cham: Springer International Publishing, 2020, pp. 126–133. doi: 10.1007/978-3-030-23813-1_16.

44. J. Chen, K. Duan, R. Zhang, L. Zeng, and W. Wang, “An AI Based Super Nodes Selection Algorithm in BlockChain Networks,” 2018, doi: 10.48550/ARXIV.1808.00216.
45. E. Markopoulos, I. S. Kirane, D. Balaj, and H. Vanharanta, “Artificial Intelligence and Blockchain Technology Adaptation for Human Resources Democratic Ergonomization on Team Management,” in *Human Systems Engineering and Design II*, vol. 1026, T. Ahram, W. Karwowski, S. Pickl, and R. Taiar, Eds. Cham: Springer International Publishing, 2020, pp. 445–455. doi: 10.1007/978-3-030-27928-8_68.
46. A. Ladia, “Privacy Centric Collaborative Machine Learning Model Training via Blockchain,” in *Blockchain and Applications*, vol. 1010, J. Prieto, A. K. Das, S. Ferretti, A. Pinto, and J. M. Corchado, Eds. Cham: Springer International Publishing, 2020, pp. 62–70. doi: 10.1007/978-3-030-23813-1_8.
47. P. Skoufis (2022) diploma-thesis-ntua [Source code] https://github.com/pskoufis13/diploma-thesis-ntua