ΞΕΝΟΣ

ΓΕΩΡΓΙΟΣ

Ιούνιος 2021

ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
Τμήμα Βιομηχανικής Σχεδίασης και Παραγωγής

Διαδικτυακές εφαρμογές και τεχνολογία Blockchain

Επιβλέπουσα καθηγήτρια: Ελένη Αικατερίνη Λελίγκου

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ/ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Ο κάτωθι υπογεγραμμένος Ξένος Γεώργιος του Αθανασίου, με αριθμό μητρώου 46181 φοιτητής του Πανεπιστημίου Δυτικής Αττικής της Σχολής Μηχανικών του Τμήματος Βιομηχανικής Σχεδίασης και Παραγωγής, δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της πτυχιακής/διπλωματικής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος.

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

Ο/Η Δηλών/ούσα

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω την επιβλέπουσα καθηγήτρια κα. Ελένη Αικατερίνη Λελίγκου για την ευκαιρία που μου έδωσε μέσω της παρούσας διπλωματικής εργασίας, ώστε να κάνω κάτι πάνω στο αντικείμενο που αγαπώ, την πληροφορική και τον προγραμματισμό.

Στη συνέχεια, θα ήθελα να ευχαριστήσω θερμά τον απόφοιτο και συνάδελφο Ιωάννη Χρηστίδη, ο οποίος αποτέλεσε ένα φιλικό πρόσωπο κατά την εισαγωγή μου στον κόσμο των blockchain. Πάντα ευγενικός και ευδιάθετος, πάντα πρόθυμος, ήταν το άτομο που μου έδειξε τα «κατατόπια» και πως να κινηθώ σε αυτά.

Επίσης, θα ήθελα να ευχαριστήσω την οικογένειά μου και τους φίλους μου για τη συνεχή υποστήριξή τους μέσα σε μια αρκετά πιεστική περίοδο.

Τέλος, θα ήθελα να ευχαριστήσω ιδιαίτερα τον πατέρα μου ο οποίος αποτέλεσε σημαντικό παράγοντα στη σύλληψη της ιδέας για την παρούσα διπλωματική. Από μικρή ηλικία μου εμφύσησε αγάπη και περιέργεια ως προς την τεχνολογία και τους υπολογιστές, ενώ μέχρι σήμερα εξακολουθεί να με εμπνέει στο να εξελιχθώ περεταίρω.

ΠΕΡΙΕΧΟΜΕΝΑ

Μέλη Επιτροπής:

Λελίγκου Ελένη-Αικατερίνη (επιβλέπουσα)

Δρόσος Χρήστος

Πυρομάλης Δημήτριος

Eleni Aikaterini Leligkou

Digitally signed by Eleni Aikaterini Leligkou
Date: 2021.07.23 11:24:46 +03'00'

Christos Drosos

Digitally signed by Christos Drosos
Date: 2021.07.22 13:40:31 +03'00'

Dimitrios Pyromalis

Digitally signed by Dimitrios Pyromalis
Date: 2021.07.22 13:19:59 +03'00'

1 Περίληψη

Τα τελευταία χρόνια ο όρος “blockchain” ακούγεται όλο και περισσότερο σε παγκόσμιο επίπεδο. Ο κύριος λόγος αυτής της δημοτικότητας είναι τα κρυπτονομίσματα, μια μορφή ψηφιακού παραστατικού χρήματος η οποία είναι βασισμένη στην τεχνολογία blockchain. Σε πιο αφηρημένο επίπεδο, η τεχνολογία blockchain είναι απλά ένας τρόπος αποθήκευσης ψηφιακής πληροφορίας με τέτοιο τρόπο ώστε η παραποίησή της να καθίσταται πρακτικά αδύνατη. Αυτό επιτυγχάνεται λόγω της αποκεντρωμένης φύσης που διέπει ένα δίκτυο blockchain. Σε αντίθεση με τα παραδοσιακά συστήματα τα οποία έχουν συνήθως μια κεντρική μονάδα (server) στην οποία βρίσκεται συσσωρευμένη όλη η πληροφορία μιας οντότητας, ένα δίκτυο blockchain διαμοιράζεται σε όλους τους χρήστες του οι οποίοι αποφασίζουν από κοινού για την εξέλιξή του. Η κύρια ιδιότητα που προσδίδει ασφάλεια στην πληροφορία είναι ότι ένα blockchain ορίζεται σαν append-only, δηλαδή είναι μια δομή δεδομένων στην οποία μπορεί μόνο να προστεθεί νέα πληροφορία και όχι να αλλοιωθεί η προϋπάρχουσα. Άρα μπορούμε να συμπεράνουμε ότι ο διαμοιρασμός ενός τέτοιου δικτύου σε χρήστες δεν είναι τίποτε άλλο παρά η πρόσβασή τους στο πλήρες ιστορικό που απαρτίζει το δίκτυο, και ως κατά συνέπεια, στην παρούσα κατάσταση της πληροφορίας του.

Η σύλληψη της ιδέας των blockchains χρονολογείται από το 1982 *[1]* αλλά δεν ήταν μέχρι το 2008 που ο Satoshi Nakamoto *(ψευδώνυμο)* υλοποίησε το πρώτο blockchain στο οποίο, ένα χρόνο μετά, βασίστηκε το κρυπτονόμισμα Bitcoin. Έκτοτε, έχει ανέρθει μια πληθώρα διαφορετικών κρυπτονομισμάτων στην αγορά με αρκετά από αυτά να αποκτούν σημαντική αξία. Το δεύτερο μεγαλύτερο κρυπτονόμισμα ως προς την χρηματιστηριακή κεφαλαιοποίηση [*8*] είναι το Ethereum, το οποίο κυκλοφόρησε το 2015 [*2*] και έχει σαν κύριο χαρακτηριστικό του την αξιοποίηση των smart contracts. Τα smart contracts είναι ουσιαστικά κώδικας ο οποίος αποθηκεύεται στο blockchain και μπορεί να τρέξει μέσω των συναλλαγών που πραγματοποιούν οι χρήστες του. Επομένως, δίνεται η δυνατότητα σε ένα blockchain να φιλοξενήσει ολόκληρες εφαρμογές, των οποίων ο κώδικας όντας μέρος του blockchain είναι και αυτός αμετάβλητος, προσφέροντας έτσι αξιοπιστία ως προς τη λειτουργία τους.

Η παρούσα διπλωματική έχει ως στόχο την υλοποίηση μιας αρχιτεκτονικής η οποία θα συνδυάζει διαφορετικές εφαρμογές που θα επικοινωνούν μεταξύ τους μέσα από το ίδιο Ethereum-based blockchain. Ο σκοπός των εφαρμογών είναι να αναδημιουργήσουν το ηλεκτρονικό σύστημα συνταγογράφησης φαρμάκων της Ελλάδας έχοντας ως ειδοποιό διαφορά την ανάπτυξή τους πάνω στην τεχνολογία blockchain. Έτσι, το τελικό αποτέλεσμα προσφέρει μεγαλύτερη αξιοπιστία από το παρών σύστημα χάρη στα προαναφερόμενα πλεονεκτήματα των blockchains.

2 Abstract

Over the past years, the term “blockchain” has been gaining global traction. The main reasons behind this publicity are cryptocurrencies, a form of digital fiat currency which is based upon the blockchain technology. On an abstract layer, blockchain technology is simply an approach of storing data in such a way, that renders their ability to be tampered with practically unfeasible. This is achieved through the decentralized nature that governs a blockchain network. In contrast to traditional systems, which usually have one main server where the information of an entity is stored, a blockchain is distributed amongst its users who then decide in unison about its evolution. The major attribute granting data security is that a blockchain is defined as append-only, which means it’s a data structure where information can only be added but never changed. Therefore, we can conclude that the distribution of such a network amidst users is nothing more than granting them access to the full history that regulates the network, and consequently, the current state of its information.

The inception of blockchains can be traced back to 1982, but it wasn’t until 2008 when an individual under the alias of Satoshi Nakamoto implemented the first blockchain upon which, one year later, the cryptocurrency Bitcoin was based. Since then, there has been a plethora of different cryptocurrencies on the market with many of them gaining substantial value. The second largest cryptocurrency by market capitalization is Ethereum, which circulated in 2015 and has its focal point on smart contracts. Smart contracts are basically code stored on the blockchain, which can later be run through user transactions. On that account it enables a blockchain to host entire applications, whose code also being stored on the blockchain is itself immutable, therefore offering credibility towards their functionality.

The present thesis aims to materialize an architecture combining different applications that communicate through the same Ethereum-based blockchain. The applications serve to recreate the digital medical prescription system of Greece having the key difference of utilizing the blockchain technology. Hence, the end result offers more trust than its preexisting counterpart due to the aforementioned security advantages of blockchains.

3 Εισαγωγή

3.1 Blockchains

3.1.1 Τι είναι τα Blockchains

Τα blockchains είναι δομές δεδομένων των οποίων οι πληροφορίες βρίσκονται αποθηκευμένες μέσα σε blocks. Αυτά τα blocks είναι κατανεμημένα το ένα μετά το άλλο δημιουργώντας μια σειρά. Η κύρια διαφορά των blockchains με άλλες δομές δεδομένων είναι η append-only φύση τους που τα διέπει. Αυτό σημαίνει πως μπορεί μόνο να προστεθεί νέα πληροφορία αλλά όχι να υποστεί επεξεργασία η προϋπάρχουσα. Τα blocks, όντας υπεύθυνα για την αποθήκευση της πληροφορίας, είναι και αυτά που προστίθενται στο blockchain.

Το σύνολο των blocks ονομάζεται ledger και είναι ουσιαστικά το πλήρες ιστορικό των συναλλαγών σε ένα blockchain. Το ledger είναι και αυτό που διαμοιράζεται στους χρήστες επιτυγχάνοντας έτσι την αποκεντροποίηση, αφού πλέον η πληροφορία της δομής δεδομένων δεν βρίσκεται σε ένα μέρος αλλά σε όλους τους χρήστες του δικτύου ταυτόχρονα.

3.1.2 Blocks

Τα blocks αποτελούνται από 3 βασικά στοιχεία:

- Την πληροφορία τους.
- Το hash τους.
- Το hash του προηγούμενου block. (parent hash)

Το hash ενός block είναι το αποτέλεσμα της πληροφορίας του αφού φιλτραριστεί μέσω μιας κρυπτογραφικής συνάρτησης γνωστή και ως hash function [*4*]. Ένα hash function είναι μια μαθηματική συνάρτηση που μετατρέπει μια αριθμητική είσοδο σε μία άλλη συμπιεσμένη αριθμητική έξοδο. Η είσοδος είναι αυθαίρετου μεγέθους ωστόσο η έξοδος έχει πάντα συγκεκριμένο.

Το parent hash αποτελεί και αυτό μέρος της πληροφορίας του block οπότε και διαμορφώνει άμεσα το hash του. Έτσι επιτυγχάνεται μια αλυσίδα επαλήθευσης μεταξύ των blocks, μέσω των hash. Ας φέρουμε σαν παράδειγμα ένα blockchain το οποίο σε μια δεδομένη χρονική στιγμή αποτελείται από 100 blocks. Εάν κάποιος αλλάξει έστω και ένα μικρό τμήμα της πληροφορίας του block με αριθμό 47, τότε το hash αυτού του block θα αλλάξει επίσης. Επομένως, το block με αριθμό 48, που περιείχε και το hash του block 47, πλέον δεν δείχνει το σωστό hash. Αν πάλι ήταν κάποιος να αλλάξει και το parent hash του block 48 ώστε να είναι σωστό, αυτό με τη σειρά του θα επηρέαζε και το hash του ίδιου του block 48. Όλο αυτό δημιουργεί ένα φαινόμενο χιονοστιβάδας όπου για να θεωρηθεί ένα αλλοιωμένο block ορθό, θα πρέπει να αλλοιωθούν και τα επακόλουθά του καταλλήλως. Αυτό σημαίνει ότι το τελευταίο block θα καταλήξει να έχει και εκείνο διαφορετικό hash. Όταν λοιπόν αυτό το παραποιημένο ledger παρουσιαστεί στους υπόλοιπους χρήστες, αυτοί θα το απορρίψουν καθώς δεν θα ταιριάζει με αυτό που έχουν οι ίδιο.

3.1.3 Nodes

Όπως προαναφέρθηκε, ένα blockchain είναι διαμοιρασμένο μεταξύ των χρηστών του δικτύου του. Ο όρος «χρήστες» στη συγκεκριμένη περίπτωση διαφέρει από αυτόν που έχουμε στο μυαλό μας συνήθως. Σε ένα blockchain ορίζουμε ως χρήστες τους κόμβους, ή αλλιώς nodes, στους οποίους είναι διαμοιρασμένο το ledger. Ωστόσο δεν περιορίζονται μόνο σε αυτό. Ένας κόμβος μπορεί να κάνει περισσότερα ή και λιγότερα από την απλή αποθήκευση του ledger. Για παράδειγμα μπορεί να επικυρώνει συναλλαγές στο δίκτυο, να κρατάει ένα ιστορικό αυτών, να εμπεριέχει ολόκληρο ή τμήμα του ledger, να λειτουργεί απλά ως σημείο επικοινωνίας. [*6*]

3.1.3.1 Full Nodes

Οι κόμβοι μπορούν να κατηγοριοποιηθούν σε 2 γενικές, μητρικές κατηγορίες, τα full nodes και τα light nodes. Τα full nodes είναι ο πυλώνας ενός blockchain, και είναι υπεύθυνα για την αποθήκευση του πλήρους ιστορικού των συναλλαγών στο εκάστοτε δίκτυο. [*6*] Με άλλα λόγια τα full nodes *είναι* το blockchain. Έχοντας λοιπόν το πλήρες ιστορικό του δικτύου είναι και υπεύθυνα για την εξέλιξη του, πράγμα που επιτυγχάνεται με την πλειοψηφική τους συμφωνία στην προσθήκη ενός block. Το ποσοστό των κόμβων που ορίζεται ως πλειοψηφία δεν είναι απαραίτητα το 50% και ποικίλει από δίκτυο σε δίκτυο.

3.1.3.2 Light Nodes

Ένα light node, όπως υποδηλώνει και το όνομά του, είναι ένας «ελαφρύτερος» κόμβος που εμπεριέχει μόνο τις απαραίτητες πληροφορίες για πιο τετριμμένες λειτουργίες, όπως καθημερινές συναλλαγές. Ως κατά συνέπεια, δεν έχει τη δυνατότητα να επαληθεύει νέα blocks όπως ένα full node. Μία γενικευμένη αναλογία που θα μπορούσε να γίνει είναι να παρομοιαστεί ένα full node με έναν server ενώ ένα light node με μια τοπική αποθηκευτική μονάδα, όπως τη μνήμη cache. [*6*] Το light node αντλεί στοιχεία από το full αλλά μόνο αυτά που χρειάζεται, και τα αποθηκεύει για μελλοντική χρήση. Εάν κάποια στιγμή χρειαστεί περεταίρω δεδομένα τότε επικοινωνεί ξανά με το full node για να γίνει η κατάλληλη ενημέρωση.

Ένα blockchain ενδέχεται να εμπεριέχει δεδομένα από πολλές διαφορετικές εφαρμογές και οργανισμούς που μπορεί να μην έχουν καμία σχέση μεταξύ τους, οπότε δεν θα ήταν πρακτικό να είχαμε αποκλειστικά full nodes ενώ χρειαζόμαστε μόνο τμήμα του blockchain για μια συγκεκριμένη λειτουργία.

3.1.4 Τύποι Δικτύων Blockchain

Υπάρχουν 2 βασικοί τύποι στα δίκτυα blockchain, τα δημόσια (public) και τα ιδιωτικά (private).

3.1.4.1 Public Blockchains

Ένα public blockchain επιτρέπει στον οποιονδήποτε να συμμετάσχει χωρίς περιορισμούς. Αυτό επιτυγχάνει μέγιστη αποκεντροποίηση καθώς δεν υπάρχει ένας οργανισμός που ελέγχει το δίκτυο. [5] Ουσιαστικά ο οποιοσδήποτε, με το κατάλληλο λογισμικό, μπορεί να εγκαταστήσει έναν κόμβο στον υπολογιστή του και όχι μόνο να αποκτήσει πρόσβαση στο blockchain, αλλά να αποτελεί ενεργό μέλος αυτού και να έχει μερίδιο στην εξέλιξή του.

Τα κυριότερα πλεονεκτήματα [7] ενός δημόσιου blockchain είναι:

- Υψηλή Ασφάλεια

Δίνοντας ελεύθερη πρόσβαση στο δίκτυο το blockchain διαμοιράζεται σε πολλούς κόμβους και καθιστά τις κακόβουλες επιθέσεις από πολύ δύσκολες έως και αδύνατες. Από τη στιγμή που το blockchain υπάρχει σε πολλά μέρη ταυτόχρονα θα πρέπει να παραβιαστεί ένας μεγάλος αριθμός αυτών για να επιτύχει μια επίθεση. Επομένως η ασφάλεια ενός δικτύου αυξάνεται αναλογικά με τους συμμετέχοντες, και προφανώς ελεύθερη είσοδος σημαίνει και αυξημένος αριθμός συμμετεχόντων.

- Ανωνυμία

Σε ένα δημόσιο blockchain δεν χρειάζεται κάποιο είδος ταυτοποίησης και μέσω της κρυπτογράφησης που εφαρμόζεται στις συναλλαγές δεν μπορεί να γίνει κάποια ανίχνευση των χρηστών, τουλάχιστον όχι άμεσα. Αυτό φυσικά έχει εκμεταλλευτεί για παράνομες συναλλαγές και δραστηριότητες, συντελώντας έτσι ένα από τα μειονεκτήματα των public δικτύων.

- Μηδενικοί Κανονισμοί

Από τη στιγμή που οι κόμβοι δεν έχουν συγκεκριμένους κανονισμούς που πρέπει να ακολουθούν, δεν υπάρχει κάποιο όριο στο πως μπορεί να αξιοποιηθεί το δίκτυο.

3.1.4.2 Private Blockchains

Ένα private blockchain δίνει πρόσβαση μόνο σε συγκεκριμένους χρήστες. Τα άτομα που συμμετέχουν σε μια συναλλαγή είναι και τα μόνα που έχουν γνώση αυτής. [5] Σε αυτή την περίπτωση λοιπόν οι κόμβοι του δικτύου είναι συγκεκριμένοι και έτσι, ένα τέτοιο δίκτυο ορίζεται ως *μερικώς* αποκεντροποιημένο.

Τα κυριότερα πλεονεκτήματα [7] ενός ιδιωτικού blockchain είναι:

- Υψηλή Αποδοτικότητα

Μικρότερος αριθμός συμμετεχόντων στο δίκτυο σημαίνει μικρότερος αριθμός αιτημάτων ως προς αυτό, άρα και μικρότερος φόρτος. Η αλήθεια είναι πως παρά τα πλεονεκτήματά τους τα δημόσια blockchain είναι σχετικά αργά, συνεπώς χρειάζονται χρόνο για να επεξεργαστούν τα αιτήματα. Τα ιδιωτικά blockchain λοιπόν δεν αντιμετωπίζουν τέτοιο πρόβλημα.

- Ιδιωτικότητα

Όπως είναι αναμενόμενο, ένα ιδιωτικό δίκτυο προσφέρει... ιδιωτικότητα. Συνδυάζοντας την ασφάλεια των blockchain και τον έλεγχο πρόσβασης των χρηστών, ένα private blockchain καθίσταται ιδανικό για επιχειρήσεις που θέλουν τη μεγαλύτερη δυνατή ασφάλεια στα δεδομένα τους.

- Σταθερότητα

Γενικά στα blockchain όταν κάποιος επιθυμεί να πραγματοποιήσει μια συναλλαγή, χρειάζεται να πληρώσει ένα μικρό αντίτιμο σε κρυπτονόμισμα. Αυτό το αντίτιμο έχει ως χρήση την προτεραιοποίηση μίας συναλλαγής έναντι μίας άλλης. Όπως προαναφέρθηκε, ένα δημόσιο δίκτυο έχει καθυστερήσεις και γενικά μεγάλο αριθμό αιτημάτων και το παραπάνω αντίτιμο είναι αυτό που θα καθορίσει τη σειρά με την οποία θα εκτελεστεί το κάθε αίτημα. Σε μεγάλους αριθμούς αιτημάτων είναι αναμενόμενο να υπάρξει αύξηση του ποσού αυτού. Έχοντας λοιπόν μικρότερο αριθμό αιτημάτων σε ένα ιδιωτικό δίκτυο, δεν υπάρχει γενικά λόγος να αυξηθεί αυτό το αντίτιμο.

Συμπερασματικά, η επιλογή του κατάλληλου δικτύου για ένα blockchain είναι πολύ σημαντική και χρειάζεται ανάλυση του προβλήματος που καλείται να επιλύσει. Δεν υπάρχει αντικειμενικά ανώτερη επιλογή και το κάθε είδος καλύπτει διαφορετικές ανάγκες.

3.2 Ethereum

3.2.1 Τι είναι το Ethereum

Το Ethereum [*2*] είναι ένα blockchain ανοιχτού κώδικα με δυνατότητες φιλοξενίας smart contract. Έχει το δικό του κρυπτονόμισμα που ονομάζεται Ether, και αυτή τη στιγμή είναι το δεύτερο μεγαλύτερο στην αγορά. Το Ethereum είχε προταθεί για πρώτη φορά το 2013 από έναν Ρώσσο-Καναδό προγραμματιστή εν ονόματι Vitalik Buterin. Το 2014 η ανάπτυξή του χρηματοδοτήθηκε συλλογικά, και τελικά κυκλοφόρησε στις 30 Ιουλίου του 2015. Το Ethereum επιτρέπει σε προγραμματιστές λογισμικού να αναπτύξουν και να φιλοξενήσουν τις εφαρμογές τους στο blockchain, αξιοποιώντας έτσι την αποκεντρωμένη φύση του για να προφέρουν ασφάλεια και αξιοπιστία στους χρήστες.

3.2.2 Smart Contracts

Τα smart contracts είναι κομμάτια κώδικα που αποθηκεύονται στο blockchain και δίνουν τη δυνατότητα δημιουργίας ολόκληρων προγραμμάτων, των οποίων η λειτουργία δεν μπορεί να αλλοιωθεί όντας μέρος του blockchain.

3.2.2.1 Solidity

Η solidity [*2*] είναι αντικειμενοστραφής γλώσσα προγραμματισμού υψηλού επιπέδου και χρησιμοποιείται για τη συγγραφή των smart contracts. Φέρει ομοιότητες με τη C και τη JavaScript.

3.2.2.2 Ethereum Virtual Machine

Το Ethereum Virtual Machine [*2*] (EVM) είναι ένα περιβάλλον χρόνου εκτέλεσης για smart contracts. Ουσιαστικά είναι υπεύθυνο για το τρέξιμο του κώδικα που εμπεριέχεται στα smart contracts του Ethereum.

3.2.3 Λογαριασμοί και Διευθύνσεις

Στο Ethereum υπάρχουν 2 είδη λογαριασμών [*2*], αυτοί των χρηστών και αυτοί των smart contracts. Κάθε λογαριασμός μπορεί να περιέχει Ether και έχει την ικανότητα να στείλει αυτό το Ether σε έναν άλλον. Οι λογαριασμοί των χρηστών είναι οι μόνοι που μπορούν να δημιουργήσουν συναλλαγές ενώ αυτοί των smart contracts είναι οι μόνοι που μπορούν να περιέχουν κώδικα. Οι δεύτεροι έχουν δηλαδή μια παθητική στάση στο blockchain και περιμένουν κάποιον χρήστη να καλέσει μια από τις λειτουργίες τους.

Οι λογαριασμοί έχουν 3 βασικά χαρακτηριστικά:

- Ένα private key.
- Ένα public key.
- Μία διεύθυνση.

3.2.3.1 Private Keys

Κατά τη δημιουργία ενός Ethereum λογαριασμού δημιουργείται και ένα private key [*2*][*9*]. Το private key είναι ένα string το οποίο αποτελείται από 64 δεκαεξαδικούς χαρακτήρες. Η δημιουργία του είναι τυχαία και λειτουργεί σαν τον κωδικό ενός λογαριασμού. Συνεπώς το private key ενός λογαριασμού θα πρέπει να το γνωρίζει μόνο ο χρήστης του. Στο blockchain, όταν

επιχειρείται μια συναλλαγή από έναν λογαριασμό, απαιτείται και το private key αυτού του λογαριασμού. Εάν το private key που έδωσε ο χρήστης είναι σωστό τότε και μόνο τότε ολοκληρώνεται η συναλλαγή. Σε γενικό πλαίσιο, το private key αποδεικνύει την ιδιοκτησία του κρυπτονομίσματος που εμπεριέχεται σε έναν λογαριασμό.

3.2.3.2 Public Keys

Μετά τη δημιουργία του private key εξάγεται το public key [*2*][*9*] ενός λογαριασμού χρησιμοποιώντας έναν γνωστό αλγόριθμο που ονομάζεται Elliptic Curve Digital Signature Algorithm. Ενώ κάποιος μπορεί να βρει το public key γνωρίζοντας το private, η αντίστροφη διαδικασία δεν είναι εφικτή. Τα δύο αυτά είδη keys αποτελούν τμήμα της ασύμμετρης κρυπτογραφίας, [*10*] όπου κάθε άτομο έχει δύο τέτοια «κλειδιά». Ο αποστολέας χρησιμοποιεί το public key του παραλήπτη για να κωδικοποιήσει ένα μήνυμα. Έπειτα ο παραλήπτης μπορεί να αποκωδικοποιήσει το μήνυμα αυτό με το private key του. Όλα αυτά προφανώς δεν είναι διαδικασίες που εκτελούνται μηχανικά από τους χρήστες, αλλά αυτόματα από τους αλγορίθμους που διέπουν το εκάστοτε blockchain.

3.2.3.3 Διευθύνσεις

Αφότου έχει εξαχθεί το public key ενός λογαριασμού, περνάει μέσα από ένα hash function, παίρνονται τα τελευταία 20 ψηφία του παραγόμενου hash, προστίθεται το πρόθεμα “0x” και έτσι βρίσκεται η διεύθυνση του λογαριασμού. [*2*] Η διεύθυνση αυτή χρησιμοποιείται για την αποστολή κρυπτονομισμάτων στο λογαριασμό και έχει γενικότερα τον ρόλο «δείκτη» αυτού.

3.2.4 Wallets

Τα wallets [11] είναι εφαρμογές που επιτρέπουν στους χρήστες τους να αλληλεπιδράσουν με έναν Ethereum λογαριασμό. Μπορεί να εμφανίσουν το υπόλοιπο ενός λογαριασμού και να πραγματοποιήσουν συναλλαγές εκ μέρους του. Αυτό που κάνει επί της ουσίας ένα wallet, είναι να παρέχει το private key στο blockchain ώστε να αποδείξει την ιδιοκτησία του λογαριασμού. Για να χρησιμοποιήσει κάποιος μια Ethereum εφαρμογή χρειάζεται αναγκαστικά ένα wallet. Αυτό γίνεται επειδή ο κώδικας των smart contracts εκτελείται μέσω συναλλαγών στο δίκτυο, τις οποίες συναλλαγές μπορούν να πραγματοποιήσουν μόνο Ethereum λογαριασμοί.

3.3 Παραδείγματα εφαρμογής των Blockchain

3.3.1 Ασφαλιστικός Τομέας

Οι ασφαλιστικές εταιρίες αντιμετωπίζουν διάφορες προκλήσεις όπως χαμηλού επιπέδου πελατειακές σχέσεις, περιορισμένη ανάπτυξη σε εξελιγμένες αγορές, καθώς και τις αυξανόμενες τάσεις της ψηφιοποίησης. Η τεχνολογία blockchain προσφέρει αρκετά πλεονεκτήματα στον τομέα της ασφάλισης, συμπεριλαμβανομένης της ανάπτυξης καινοτόμων προϊόντων και υπηρεσιών, του καλύτερου εντοπισμού απάτης, της βελτιωμένης κοστολόγησης, και των μειωμένων διοικητικών δαπανών.

Τα πολυπληθή καθημερινά φαινόμενα απάτης κοστίζουν στις επιχειρήσεις τεράστια ποσά. Τα blockchain μπορούν να λύσουν το πρόβλημα αυτό αντλώντας τις πληροφορίες τους από πολλαπλές διαφορετικές πηγές, δίχως να τις μεταβάλλουν. Έτσι, οι ασφαλιστικοί μπορούν να αξιοποιήσουν τις πληροφορίες αυτές για να καταγράψουν την χρήση ενός περιουσιακού στοιχείου. Ως κατά συνέπεια, οι επιβεβαιώσεις κατοχής γίνονται γρηγορότερες, φθηνότερες, και αποδοτικότερες. [*13*]

3.3.2 Παγκόσμιο Εμπόριο

Η παραδοσιακή χρηματοδότηση του εμπορίου είναι όταν τα χρηματοπιστωτικά ιδρύματα παρέχουν πιστωτικές διευκολύνσεις στο να εγγυηθούν την ανταλλαγή αγαθών. Η διαδικασία είναι μια αιώνια βιομηχανία που δεν έχει δει πολλές αλλαγές με την αύξηση των παγκόσμιων εμπορικών ροών. Στην παραδοσιακή εμπορική διαδικασία, η αποδέσμευση κεφαλαίων εξαρτάται από την παράδοση εγγράφων και δεδομένων σε χρηματοπιστωτικά ιδρύματα, εισάγοντας την δυνατότητα για ανθρώπινα λάθη και απάτη. Σε μία αναφορά της, η Deloitte εντόπισε ορισμένες από τις προκλήσεις της παραδοσιακής χρηματοδότησης του εμπορίου, που αυξάνουν τους χρόνους συναλλαγής και παράγουν την πραγματική πιθανότητα απάτης.

- Χειροκίνητη δημιουργία και διανομή συμβάσεων.
- Σημαντικός έλεγχος έκδοσης των χρηματοδοτικών μέσων καθώς γίνονται αλλαγές.
- Καθυστέρηση πληρωμών που προκαλούνται από πολλαπλά επίπεδα επαληθεύσεων και πολλαπλούς ενδιάμεσους.
- Καθυστέρηση χρονοδιαγράμματος λόγω πολλαπλών ελέγχων από μεσάζοντες και πολυάριθμους επικοινωνιακούς κόμβους.
- Διπλή τεκμηρίωση λόγω της αδυναμίας των τραπεζών να επαληθεύσουν την δική τους αυθεντικότητα.
- Εσφαλμένη επικοινωνία και τάση απάτης λόγω της ύπαρξης πολλαπλών πλατφορμών σε διάφορες χώρες.

Η τεχνολογία Blockchain έχει τη δυνατότητα να αντιμετωπίσει ορισμένες από αυτές τις προκλήσεις και ελλείψεις. Η τεχνολογία θα μπορούσε να εξαλείψει τη μεγάλη αναποτελεσματικότητα που βιώνεται σήμερα στο διεθνές εμπόριο. Τα οφέλη του χαμηλότερου κόστους και η βελτίωση στην ασφάλεια μέσω της μείωσης των σφαλμάτων, του κινδύνου και του χρόνου, επιτρέπει σε μια εταιρεία να επιτύχει μεγαλύτερη αποτελεσματικότητα και να έχει πιο προβλέψιμο κεφάλαιο κίνησης. Η Deloitte πρότεινε μια υποδομή βασισμένη σε blockchain που έχει τη δυνατότητα να αυξήσει την αποδοτικότητα, να μειώσει τη βάση κόστους και να ανοίξει νέες ευκαιρίες εσόδων όπως νεότερα μοντέλα πίστωσης και χρηματοδότησης, αλλά και εγγυήσεις υποστήριξης του εμπορίου. Η Deloitte πρότεινε τα ακόλουθα πλεονεκτήματα για τη χρηματοδότηση συναλλαγών με βάση το blockchain:

- Επανεξέταση σε πραγματικό χρόνο: Τα οικονομικά έγγραφα, προσβάσιμα μέσω του blockchain, ελέγχονται και εγκρίνονται σε πραγματικό χρόνο. Όλα τα μέλη λειτουργούν με το ίδιο ledger, διαδικτυακά και στιγμιαία. Δεν υπάρχουν φυσικά έγγραφα ή μεταφορές. Χωρίς κίνδυνο αντιγραφής ή απώλειας.
- Διαφανής διαχείριση: Το blockchain παρέχει σε πραγματικό χρόνο μια διαφανή προβολή στα προσπελάσιμα τιμολόγια για μία καθαρή ματιά σε αυτά. Όλοι οι συνεργάτες της εφοδιαστικής αλυσίδας ενημερώνουν τις πληροφορίες σε πραγματικό χρόνο μέσα από το ίδιο σύστημα.
- Αποδιαμεσολάβηση: Το Blockchain δεν απαιτεί κάποιον αξιόπιστο διαμεσολαβητή για να αναλάβει τον κίνδυνο, εξαλείφοντας την ανάγκη για τράπεζες ανταποκριτές.
- Μειωμένος κίνδυνος αντισυμβαλλομένου: Το Blockchain παρακολουθεί τις φορτωτικές, εξαλείφοντας τη δυνατότητα διπλής δαπάνης.
- Αποκεντρωμένη εκτέλεση συμβολαίου: Ενημέρωση του blockchain σε πραγματικό χρόνο, μείωση του χρόνου και του αριθμού ατόμων που απαιτείται για την παρακολούθηση της παράδοσης των εμπορευμάτων.
- Απόδειξη ιδιοκτησίας: Το Blockchain παρέχει διαφάνεια στην τοποθεσία και ιδιοκτησία των αγαθών.
- Αυτοματοποιημένος διακανονισμός: Τα έξυπνα συμβόλαια εξαλείφουν την ανάγκη για τράπεζες ανταποκριτές και πρόσθετα τέλη συναλλαγής, μειώνοντάς τα.
- Ρυθμιστική διαφάνεια: Οι ρυθμιστικές αρχές διαθέτουν μια προβολή σε πραγματικό χρόνο για τα απαραίτητα έγγραφα.
- Αμετάβλητα δεδομένα: Επαληθεύσιμα και αμετάβλητα δεδομένα για τη μείωση του κινδύνου απάτης.

Οι πρόσφατες εμπορικές υλοποιήσεις της χρηματοδότησης του εμπορίου με βάση το blockchain ενδέχεται να είναι ένας καλός δείκτης για το τι έπεται. Το 2016, η Commonwealth Bank, η Wells Fargo, και η Brighann Cotton πρωτοστάτησαν με μία εμπορική συναλλαγή που βασίζεται σε blockchain. Η συναλλαγή αφορούσε την αποστολή βαμβακιού από το Τέξας στην Κίνα χρησιμοποιώντας την τεχνολογία blockchain. Αυτό σηματοδοτεί ένα άλλο βήμα στην αξιολόγηση τεχνολογίας η οποία, με την πάροδο του χρόνου, θα μπορούσε να υποστηρίξει την εξέλιξη της εμπορικής χρηματοδότησης. [*13*]

3.3.3 Βιομηχανία και Παραγωγή

Ενώ η πλειοψηφία των εφαρμογών της τεχνολογίας blockchain βρίσκονται στον οικονομικό τομέα, ενδιαφέρον ως προς την αξιοποίησή τους έχει εμφανιστεί και στον τομέα της παραγωγής. Τα blockchain έχουν σημαντικές προοπτικές για πληθώρα δραστηριοτήτων στον τομέα αυτόν καθώς και τη δυνατότητα να τον αλλάξουν ριζικά.

Τα διαμοιρασμένα ledgers μπορούν να χρησιμοποιηθούν για να λύσουν διάφορες προκλήσεις, ειδικά στον τομέα διαχείρησης της εφοδιαστικής αλλυσίδας, όπως την παρακολούθηση των container κατά την αποστολή τους καθώς και σημαντικών πληροφοριών σχετικά με τα προϊόντα. Οι απαιτήσεις πελατών για καλύτερη εξυπηρέτηση σημαίνει πως το να υπάρχουν τα κατάλληλα προϊόντα στα ράφια είναι ζωτικής σημασίας. Ο ατελείωτος κύκλος των αυξανόμενων κοστών ως προς τις εφοδιαστικές αλυσίδες είναι κάτι που επηρεάζει όλους. Παραγωγοί, έμποροι και διανομείς, όλοι έχουν συμφωνήσει πως η μείωση των παραπάνω κοστών είναι ένα φλέγων ζήτημα.

Ανά τα χρόνια, τεχνολογίες όπως GPS, RFID, barcodes, smart labels, location-based data, ασύρματα δίκτυα αισθητήρων, και τεχνολογίες cloud, έχουν παίξει σημαντικό ρόλο στην συγκέντρωση των πληροφοριών σε μία αλυσίδα εφοδιασμού.

Η τεχνολογία blockchain έχει ιδιαίτερες προοπτικές στο να λύσει 3 βασικά ζητήματα στον τομέα των αλυσίδων εφοδιασμού. Συγκεκριμένα, την παρακολούθηση/διαφάνεια, την πλαστογράφηση και την αποδοτικότητα. Οι σημερινές εφοδιαστικές αλυσίδες τείνουν να έχουν μία μεγάλη γκάμα από συμφωνίες και συμβολαιογραφικές υποχρεώσεις, με τις παραγγελίες να τοποθετούνται κατά μήκος πολλαπλών διαύλων επικοινωνίας. Αυτή η ανάμειξη αναλογικού και ψηφιακού παρουσιάζει αρκετά προβλήματα όπως τη μειωμένη διαφάνεια και γενικά τη μικρή αποδοτικότητα ως προς ολόκληρη την εφοδιαστική αλυσίδα. Η τεχνολογία blockchain φέρει τάξη, απλότητα, εμπιστοσύνη, ορατότητα, και αυτοματισμό σε ένα, υπό διαφορετικές συνθήκες, χαοτικό περιβάλλον. Έτσι εξαλείφεται σημαντικά η χαρτογραφία ενώ παράλληλα βελτιώνεται και η ασφάλεια. Τα blockchain μπορούν να κρατούν πληροφορίες για κάθε επιμέρους τμήμα, και να το κάνουν προσβάσιμο σε κάθε κατασκευαστή κατά τη διαδικασία της παραγωγής. Ουσιαστικά αποτελούν μία εναλλακτική έχοντας τη δυνατότητα να βελτιώσουν και να επιταχύνουν τη διάδοση της πληροφορίας, αντικαθιστώντας τη χαρτογραφία και τους μηχανικούς ελέγχους συστημάτων, που καθιστούν τις αλυσίδες εφοδιασμού ευάλωτες ως προς τις ανακρίβειες.

Η τεχνολογία θέτει εμπιστοσύνη μεταξύ των εταίρων, εξασφαλίζοντας πως κάθε συναλλαγή καταγράφεται και αποθηκεύεται σε πολλαπλές τοποθεσίες σε ολόκληρο το δίκτυο. Διαχειριστικές λειτουργίες μειώνονται ή εξαλείφονται ολοκληρωτικά χάρη στην ορατότητα των συναλαγών. Αυτό, ως κατά συνέπεια, αυξάνει την αποδοτικότητα της εφοδιαστικής αλυσίδας και μειώνει την πολυπλοκότητα του συστήματος. Οι πελάτες ξέρουν από τι είναι φτιαγμένο ένα προϊόν, από πού προέρχεται, και ποια η επιρροή του στο περιβάλλον. Οι παραγωγοί και οι έμποροι ωφελούνται από καλύτερη παρακολούθηση των εμπορευμάτων και από την ενίσχυση των πελατών τους. Επιπροσθέτως, δίνεται στους παραγωγούς/διανομείς μία πολύ καλύτερη εικόνα ως προς τις ανάγκες και τη ζήτηση των πελατών, έχοντας έτσι τη δυνατότητα να προσαρμόσουν τις όποιες επιμέρους διαδικασίες της παραγωγής, ώστε να καλύψουν αυτές τις ανάγκες και να έχουν αύξηση των εσόδων μέσω της πελατειακής ικανοποίησης.

Αυτή τη στιγμή υπάρχουν μερικές υποδείξεις επιχειρήσεων που χρησιμοποιούν την τεχνολογία blockchain για τη βελτίωση της αποδοτικότητας των εφοδιαστικών αλυσίδων τους. Για παράδειγμα, η Γερμανική εταιρία προσωπικής περιποίησης Beiersdorf, πειραματίστηκε με τα blockchain για να δημιουργήσει μία ανοιχτή συναλλαγή παλετών. Πληροφορίες σχετικά με τις παλέτες και το περιεχόμενό τους ψηφιοποιείται καθημερινά. Σαρώνοντας ένα QR κωδικό, είναι εφικτό να στείλουμε, να λάβουμε και

να διευθετήσουμε τις παλέτες. Όλες οι πληροφορίες επεξεργάζονται μηχανικά και αποθηκεύονται πλέον σε ένα έμπιστο, διαμοιρασμένο δίκτυο blockchain. Ο διαμοιρασμός πληροφορίας σε ένα έμπιστο δίκτυο γλιτώνει χρόνο και οδηγεί σε βελτιωμένη συνεργασία.

Το 2017, η EZ Lab ανέπτυξε μία πλατφόρμα για τη βιομηχανία κρασιού, με το όνομα Carto. Αυτή η πλατφόρμα είναι βασισμένη στο Ethereum. Οι γεωργοί, οι οινοποιοί και οι έμποροι μπορούν να κάνουν εγγραφή με μία κρυπτογραφημένη ψηφιακή υπογραφή έτσι ώστε οι καταναλωτές να μπορούν να επιβεβαιώσουν ότι αγοράζουν. Αυτή η πλατφόρμα βοηθάει τη βιομηχανία του κρασιού να αποφύγει την πλαστογραφία. Το παραπάνω εγχείρημα συναντήθηκε με θετικότητα από την βιομηχανία, και έχει πραγματοποιήσει συναλλαγές αξίας 200,000 δολαρίων και άνω από το 2017.

Σε ένα άλλο παράδειγμα, η Carrefour αξιοποίησε ένα Ethereum-based δίκτυο για να παρακολουθεί την εγκυρότητα και τις συνθήκες πουλερικών ελευθέρας βοσκής. Μετά την παρουσίασή του τον Ιούνιο του 2018, οι πελάτες μπορούσαν πλέον να σαρώνουν ένα QR κωδικό στη συσκευασία χρησιμοποιώντας το κινητό τους τηλέφωνο, για να αποκτήσουν πρόσβαση στην πληροφορία σχετικά με την ημερομηνία γέννησης του κοτόπουλου, καθώς και τη στιγμή που τοποθετήθηκε στο ράφι. [*13*]

3.3.4 Κυβέρνηση και Δημόσιος Τομέας

Τα blockchain μπορούν να βοηθήσουν τις κυβερνητικές δραστηριότητες σε διάφορους τομείς. Ένας από αυτούς είναι η διαχείριση αρχείων. Εθνικές, αστικές αλλά και δημοτικές κυβερνήσεις είναι υπεύθυνες για την κατοχή των εγγράφων των πολιτών που αφορούν ημερομηνίες γέννησης, θανάτου, και γενικότερες συναλλαγές ιδιοκτησίας. Μερικά από αυτά τα αρχεία εξακολουθούν να βρίσκονται σε χαρτογραφική μορφή. Η διαδικασία της τροποποίησης και της ενημέρωσης αυτών των αρχείων είναι κοπιαστική, περιττή και εκνευριστική. Η τεχνολογία blockchain μπορεί να αναδιατάξει τη διαχείριση αυτή και να κάνει τα ίδια τα αρχεία πιο ασφαλή. Τα πιστοποιητικά γάμου, θανάτου και γέννησης μπορούν να αποθηκεύονται στο δίκτυο blockchain, όπου τα αρχεία ενός θα μπορούν να ανακτηθούν με ασφάλεια.

Η αποκεντρωμένη αποθήκευση των αρχείων, όπου είναι διαμοιρασμένα σε όλο το δίκτυο, τα προστατεύει από την παραβίαση και την απώλειά τους. Πολλοί δημοτικοί, αστικοί και εθνικοί φορείς έχουν δείξει ενδιαφέρον στο να υιοθετήσουν τα blockchain ως μέσο αποθήκευσης των πληροφοριών τους. [*13*]

3.3.5 Επιστήμες Υγείας

Τα blockchain είναι ιδανικά για την οποιαδήποτε πληροφορία στην οποία υπάρχει δυνατότητα επεξεργασίας από πολλαπλά μέλη, θέλοντας παράλληλα να διέπεται από συναίνεση, εγκυρότητα και επαλήθευση. Ως δημοσίως προσβάσιμα ledgers, τα blockchain μπορούν να κάνουν κάθε είδους καταγραφή αρχείων πιο αποδοτική και να προσφέρουν μια λύση σε προβλήματα που παρουσιάζονται συγκεκριμένα στον τομέα της υγείας. Η τεχνολογία των blockchain υπολογίζεται ως καλή επιλογή για την ασφάλιση ιατρικών εγγράφων, πληροφοριών DNA, προσωπικών δεδομένων και γενικότερα ιατρικών ιστορικών. Τα μεγάλα νοσοκομεία μπορούν να αξιοποιήσουν τα blockchain ώστε να αποθηκεύσουν λεπτομέρειες της κατάστασης των ασθενών τους. Έτσι, ιατροί και ασθενείς, μπορούν να αποκτήσουν πρόσβαση σε αυτά τα αρχεία οποτεδήποτε και οπουδήποτε μέσω του δικτύου.

Ο αριθμός των παραβιάσεων σε ιατρικές βάσεις δεδομένων βρίσκεται σε έξαρση. Τα blockchain όπως είναι αναμενόμενο μπορούν να προσφέρουν πολλά έναντι αυτού του προβλήματος καθώς επιτρέπουν στα όποια μέλη του συστήματος να ανταλλάξουν πληροφορίες χωρίς να τις εκθέτουν σε κίνδυνο. Καλύτερη

επικοινωνία και διαμοιρασμός πληροφοριών μεταξύ των συνεργατών του τομέα, μπορεί να οδηγήσει σε μία γρηγορότερη και πιο στοχευμένη διάγνωση, έχοντας ως κατά συνέπεια τις πιο αποτελεσματικές και οικονομικές θεραπείες.

Εναλλακτικά, έχουμε τον φαρμακευτικό τομέα ο οποίος μπορεί να απολάβει τους καρπούς των blockchains για το ερευνητικό του τμήμα, το οποίο αξιοποιεί την ασφάλεια που του προσφέρεται για τη σταθερή ροή πληροφοριών που απαιτεί. Επίσης, τα blockchain μπορούν να φανερώσουν το πλήρες φάσμα των παρενεργειών που σχετίζονται με τις φαρμακευτικές θεραπείες, καθώς η ελεγχόμενη κατάχρηση ουσιών γίνεται ολοένα και πιο συχνή.

Όσων αφορά τις μεταμοσχεύσεις οργάνων, τα blockchain φαίνεται πάλι να χαράζουν σημαντική πορεία. Γενικά οι μεταμοσχεύσεις είναι σύνθετες. Τα όργανα «χαλάνε» πολύ γρήγορα και ένας δωρητής πρέπει να είναι συμβατός με το δέκτη όσων αφορά τον τύπο αίματος. Σύμφωνα με το κέντρο μεταμοσχεύσεων του πανεπιστημίου του Μίσιγκαν, μία καρδιά ή ένας πνεύμονας μεταμοσχεύεται συνήθως σε λιγότερο από δέκα ώρες. Χωρίς ένα αποδοτικό σύστημα σε λειτουργία, όργανα ζωτικής σημασίας καταλήγουν να πάνε χαμένα. Περισσότεροι από 120,000 άνθρωποι βρίσκονται σε λίστα αναμονής για κάποιο όργανο με 22 από αυτούς, κατά μέσο όρο, να πεθαίνουν καθημερινά. Το Organtree είναι η πρώτη εταιρία στον κόσμο που παρέχει μία αποκεντρωμένη βάση δεδομένων, βασισμένη στην τεχνολογία blockchain, για δωρεές οργάνων. Η παραπάνω συνδέει δωρητές με ασθενείς και υγειονομικές μονάδες, επιτυγχάνοντας έτσι ταχύτερη διαδικασία μεταμοσχεύσεων, καθώς και μεγαλύτερο αριθμό ανίχνευσης συμβατοτήτων.

Τα Ηνωμένα Αραβικά Εμιράτα αποτελούν την πρώτη χώρα σε παγκόσμιο επίπεδο που θα χρησιμοποιήσουν την τεχνολογία blockchain σε συνδυασμό με την τεχνητή νοημοσύνη για τις μεταμοσχεύσεις οργάνων. Ο κύριος στόχος είναι να παρέχεται ένα ασφαλέστερο και βελτιστοποιημένο περιβάλλον για τη διαδικασία των δωρεών. Επιπλέον στόχος είναι να δημιουργηθούν οι υποδομές που καθιστούν εφικτό τον καλύτερο εντοπισμό συμβατοτήτων, την καλύτερη επαλήθευσή τους και γενικότερα τις πιο αποδοτικές μεταμοσχεύσεις, μέσω της τεχνητής νοημοσύνης και των blockchain. Επίσης, είναι στα σχέδια η σύνδεση με 7 νοσοκομεία σε όλη τη χώρα, τα οποία παρέχουν χειρουργικές υπηρεσίες μεταμοσχεύσεων. Το blockchain θα εξασφαλίσει πως τα όργανα που δωρίζονται έχουν επαληθευτεί μέσω DNA το οποίο θα υπάρχει ήδη μέσα στο δίκτυο. Τα νοσοκομεία έτσι μπορούν να επιβεβαιώσουν πως το DNA του οργάνου που έχουν, ταιριάζει με το DNA ενός δωρητή που βρίσκεται μέσα στο σύστημα. [*13*]

3.3.6 Βιομηχανία των Τροφίμων

Η αστικοποίηση και οι καταναλωτικές συνήθειες της γενιάς της χιλιετίας, έχουν αυξήσει τον αριθμό ατόμων που προμηθεύεται έτοιμο φαγητό. Η παγκοσμιοποίηση έχει πυροδοτήσει μία αυξημένη ζήτηση για μεγαλύτερη γκάμα τροφίμων, έχοντας ως αποτέλεσμα μια αυξανόμενα σύνθετη και μεγαλύτερη, παγκόσμια γραμμή εφοδιασμού. Η βιομηχανοποίηση της γεωργίας και των προϊόντων ζωικής προέλευσης για την ικανοποίηση της αυξανόμενης ζήτησης, φέρει προβλήματα στον τομέα της ασφάλειας. Η κλιματική αλλαγή, μέσω της αύξησης της θερμοκρασίας, δημιουργεί προκλήσεις ως προς την παραγωγή την αποθήκευση και τη διανομή των τροφίμων. Όλο και περισσότεροι άνθρωποι απασχολούνται για τον προέλευση των τροφίμων τους. Τον Οκτώβριο του 2006, πολλαπλές πολιτείες στην Αμερική υπέστη ένα ξέσπασμα E-Coli όπου περίπου 200 άνθρωποι νόσησαν, τρείς εκ των οποίον κατέληξαν νεκροί. Οι εφοδιαστικές αλυσίδες των τροφίμων είναι ιδιαίτερα σύνθετες και μπορούν να διασχίζουν πολλές χώρες. Η αντίστροφη παρακολούθηση της πορείας ενός τρόφιμου, μέχρι και την παραγωγή του, είναι μια αρκετά αργή διαδικασία κατά τη διάρκεια της οποίας ολόκληρες βιομηχανίες μπορεί να κλείσουν. Άρτια

συνεργασία μεταξύ των κυβερνήσεων, των παραγωγών, και των καταναλωτών συμβάλλει σημαντικά στην ασφάλεια των τροφίμων.

Σύμφωνα με τον Παγκόσμιο Οργανισμό Υγείας, ένας στους 10 ανθρώπους νοσεί ετησίως λόγω κακής ποιότητας τροφίμων, ενώ σχεδόν μισό εκατομμύριο πεθαίνει ως αποτέλεσμα νόσου που προήλθε από αυτά. Επιπλέον, ενώ πολλοί άνθρωποι παγκοσμίως δεν μπορούν να τραφούν επαρκώς, σχεδόν το ένα τρίτο της παραγόμενης τροφής πάει χαμένο. Τα παιδιά κάτω των 5 ετών κουβαλάνε το 40% των ασθενειών τροφικής προέλευσης. Θεωρείται πως αυτές οι ασθένειες εμποδίζουν την κοινωνικοοικονομική ανάπτυξη λόγω του φόρτου που επιβάλλουν στο σύστημα υγείας. Παρομοίως, η ασφαλής παροχή τροφής υποστηρίζει την παγκόσμια οικονομία, το εμπόριο και τον τουρισμό.

Υπάρχουν πολλά προβλήματα με τις υπάρχουσες αλυσίδες παροχής, αλλά τα blockchain μπορούν να αποτελέσουν μια ισχυρή λύση παρακολουθώντας ολόκληρη την πορεία των τροφίμων, από την παραγωγή, μέχρι και την κατανάλωση. Η τεχνολογία blockchain, μέσω της αποτελεσματικής παρακολούθησης των τροφίμων, μπορεί να μειώσει τον χρόνο που διαφορετικά θα χανόταν μέσω της γραφειοκρατίας. Αυτό επιτυγχάνεται συνδέοντας τις πληροφορίες ενός προϊόντος ψηφιακά, όπως την φάρμα παραγωγής, τον αριθμό παρτίδας, πληροφορίες εργοστασιακών επεξεργασιών, ημερομηνίες λήξεως, θερμοκρασίες αποθήκευσης και λεπτομέρειες μεταφοράς. Κάθε επιμέρους τμήμα της πληροφορίας παρέχει χρήσιμα συμπεράσματα που θα μπορούσαν ενδεχομένως να αποκαλύψουν ελλαττωματικά προϊόντα.

Όταν μια πληροφορία προστίθεται στο blockchain μέσω συναίνεσης, γίνεται ένα μόνιμο τμήμα αυτού που δεν μπορεί να αλλοιωθεί. Αυτό βοηθά στην εξασφάλιση της ακρίβειας της πληροφορίας. Τα blockchain μπορούν επίσης να βοηθήσουν τους πωλητές ώστε να διαχειριστούν καλύτερα τον κύκλο ζωής των τροφίμων στα ράφια. Η τεχνολογία αυτή θεωρείται πως δίνει την ιδιότητα της διαφάνειας και της ανιχνευσιμότητας ενώ παράλληλα ενισχύει την ακεραιότητα των προϊόντων, μειώνει την απάτη, και μεγιστοποιεί την ασφάλεια των καταναλωτών.

Γενικά τα πλεονεκτήματα που προσφέρει η τεχνολογία blockchain στη βιομηχανία των τροφίμων είναι:

- Ασφάλεια τροφίμων
- Φρεσκάδα και ποιότητα
- Μειωμένη σπατάλη
- Καταπολέμηση απάτης
- Προώθηση υπευθυνότητας μεταξύ των παραγωγών [*13*]

3.3.7 Ασφάλεια Υπολογιστικών Συστημάτων

Οι παραβιάσεις ηλεκτρονικών συστημάτων συνεχώς αυξάνονται. Οι κακόβουλοι χρήστες γίνονται όλο και πιο ικανοί και καταστροφικοί. Οι οργανισμοί οφείλουν να ελαχιστοποιούν όσο το δυνατόν περισσότερο τις αδυναμίες στην ασφάλεια των συστημάτων τους. Η αποτελεσματική προστασία των υπολογιστικών συστημάτων είναι πιο σημαντική από ποτέ, καθώς υπέρογκα ποσά προσωπικών πληροφοριών βρίσκονται αυτή τη στιγμή αποθηκευμένα σε ιστοσελίδες και πλατφόρμες κοινωνικής δικτύωσης. Το διαδίκτυο είναι γεμάτο με εργαλεία και ικανούς hacker οι οποίοι συνεχώς προκαλούν σημαντικές ζημιές σε οργανισμούς και ατομικές μονάδες. Οι επιθέσεις γίνονται ολοένα και πιο διακριτικές, έχοντας μεγαλύτερες οικονομικές επιπτώσεις. Αναφορές από επιχειρήσεις που έχουν υποστεί τέτοιες

επιθέσεις, με ανεπανόρθωτα καταστροφικές συνέπειες στη φήμη τους, είναι αρκετά συνήθεις. Οι οργανισμοί, ενώ επενδύουν πολύ χρόνο, χρήμα, και ανθρώπινο δυναμικό ώστε να καταπολεμήσουν αυτό το φαινόμενο, εξακολουθούν να πέφτουν θύματα των επιθέσεων.

Οι παραδοσιακές μέθοδοι αντιμετώπισης έχουν αποδεδειγμένα αποτύχει. Η τεχνολογία Blockchain ωστόσο αποτελεί μία προληπτική μέθοδο η οποία είναι ικανή να ανιχνεύσει τους όποιους πιθανούς κινδύνους, που αναπόφευκτα θα ξεφύγουν από τους πατροπαράδοτους μηχανισμούς άμυνας.

Με την αυξανόμενη χρήση των οικιακών «έξυπνων» συσκευών, ανοίγουν και οι ορίζοντες για τις πιθανές επιθέσεις. Όλες αυτές οι συσκευές συνδέονται στο διαδίκτυο για την αποτελεσματική παροχή των υπηρεσιών και των δυνατοτήτων τους. Αυτό τις καθιστά εν δυνάμει εισόδους για να παραβιάσει κάποιος hacker την ιδιωτικότητα μίας κατοικίας. Υπάρχουν αρκετοί λόγοι για τους οποίους οι οικιακές αυτές συσκευές είναι ευάλωτες:

- Είναι κατασκευασμένες για διευκόλυνση. Οι κατασκευαστές είναι επιφυλακτικοί ως προς την εγκατάσταση μέτρων ασφαλείας, υπό το πρίσμα του ότι μπορεί οι καταναλωτές να θεωρήσουν τα μέτρα αυτά ενοχλητικά.
- Οι κατασκευαστές δεν είναι πάντα γνώστες του πως να εφαρμόσουν τις όποιες δικλείδες ασφαλείας αποτελεσματικά.
- Μερικοί καταναλωτές αδυνατούν να ακολουθήσου τα βήματα ασφαλείας για την ορθή σύνδεση των συσκευών τους στο διαδίκτυο.
- Οι περισσότεροι καταναλωτές δεν θεωρούν πως οι συσκευές αυτές απαιτούν την ίδια προσοχή όπως οι υπολογιστές και τα smartphone.

Η τεχνολογία blockchain μπορεί να μειώσει τους κινδύνους της ηλεκτρονικής ασφάλειας με διάφορους τρόπους. Αρχικά προσφέρει μια αεροστεγή μέθοδο ταυτοποίησης και προστατεύει τους χρήστες με τα δεδομένα τους από επιθέσεις. Επίσης παρέχει αξιοπιστία στα δεδομένα του οικονομικού τομέα, κάτι ιδιαίτερα σημαντικό για τους μέτοχους. Οι εφοδιαστικές αλυσίδες όπως η φαρμακευτική, τα ηλεκτρονικά, τα διαμάντια και τα αγαθά πολυτελείας χρειάζονται διαφάνεια για να λειτουργήσουν σωστά.

Μερικές από τις πιο αποτελεσματικές και πολλά υποσχόμενες εφαρμογές των blockchain στον τομέα της ασφάλειας του κυβερνοχώρου είναι οι εξής:

- **GuardTime** | Ένα blockchain project που έχει ως σκοπό να δημιουργήσει «keyless» συστήματα ψηφιακών υπογραφών για την ασφάλεια των ιατρικών εγγράφων σε ένα εκατομμύριο πολίτες της Εσθονίας.
- **REMME** | Επίσης ένα blockchain project το οποίο αποσκοπεί στην καλύτερη ποιότητα ασφάλειας χρηστών και επιχειρήσεων, αντικαθιστώντας τις διαδικασίες login με πιστοποιητικά SSL τα οποία αποθηκεύονται στο blockchain.
- **Blockverify** | Μία λύση έναντι της απάτης βασισμένη σε blockchain, η οποία ανιχνεύει τις πλαστογραφίες και προσφέρει ένα περιβάλλον στο οποίο δεν είναι εφικτή η δημιουργία αντιγράφων. Επίσης παρέχει την αξιοπιστία που χρειάζεται για τη λειτουργεία αλυσίδων παροχής με 100% διαφάνεια. Το Blockverify χρησιμοποιείται για ηλεκτρονικά, φαρμακευτικά προϊόντα, και είδη πολυτελείας.
- **PeerNova** | Μία τεχνολογία που παρέχει στα χρηματοοικονομικά ινστιτούτα έναν τρόπο να επαληθεύουν και να ασφαλίζουν τα δεδομένα τους, να δειαχειρίζονται τους λογιστικούς ελέγχους, καθώς και συμφωνίες. [*13*]

4 Περιγραφή και αρχιτεκτονική των εφαρμογών

4.1 Περίληψη

Η συγκεκριμένη εργασία υλοποιεί ένα σύνολο εφαρμογών σε μία. Αποσκοπεί στην αναδημιουργία του συστήματος συνταγογράφησης της Ελλάδας πάνω σε ένα Ethereum blockchain και προσφέρει τις εξής δυνατότητες:

- Καταχώρηση προφίλ για ιατρούς, φαρμακεία και ασθενείς.
- Καταχώρηση φαρμάκων.
- Καταχώρηση συνταγών.

Το περιβάλλον αλληλεπίδρασης είναι ένας ιστότοπος ειδικά διαμορφωμένος για την επικοινωνία με μια αποκεντροποιημένη εφαρμογή.

4.2 Χρήστες

Οι χρήστες της παρούσας εφαρμογής είναι αποκλειστικά ιατροί και φαρμακεία. Ο τρόπος που αλληλεπιδρούν ο καθένας είναι περιορισμένος με βάση τις αρμοδιότητές τους. Ένας ιατρός την χρησιμοποιεί για να γράψει μια συνταγή σε έναν ασθενή ενώ ένα φαρμακείο την χρησιμοποιεί για να την εκτελέσει. Οι ασθενείς δεν αλληλεπιδρούν με την εφαρμογή αλλά το προφίλ τους εμπεριέχεται σε αυτή και είναι ενημερωμένο με την κατάσταση της ασφάλισής τους. Παράλληλα υπάρχει και ένας διαχειριστής ο οποίος είναι αυτός που καταχωρεί τα προφίλ των χρηστών, των ασθενών καθώς και τα φάρμακα που θα είναι διαθέσιμα στο σύστημα. Οι χρήστες χρειάζονται ένα wallet για να αλληλεπιδράσουν με την εφαρμογή, το οποίο αντικαθιστά τη διαδικασία του login και χρησιμοποιεί έναν Ethereum λογαριασμό για τις συναλλαγές. Η εφαρμογή της εργασίας αυτής υποτίθεται πως θα ανήκει στον ΕΦΚΑ, ο οποίος θα είναι ο διαχειριστής έχοντας και αυτός έναν Ethereum λογαριασμό.

4.3 Αρχιτεκτονική

4.3.1 Τύπος Δικτύου

Με βάση τα πλεονεκτήματα του κάθε είδους δικτύου που προαναφέραμε παραπάνω, συμπεραίνουμε πως η βέλτιστη επιλογή δικτύου για την παρούσα εφαρμογή είναι ένα private δίκτυο. Αρχικά επιλέγοντας ένα ιδιωτικό δίκτυο μας δίνεται η δυνατότητα να το ορίσουμε όπως μας εξυπηρετεί. Τα κόστη συναλλαγών δεν μπορούν απλά να μειωθούν, αλλά να εξαλειφθούν εντελώς, εξασφαλίζοντας έτσι ότι κανένας χρήστης δεν πρόκειται να αποκλειστεί από την εφαρμογή λόγω έλλειψης κεφαλαίου στον λογαριασμό του. Επίσης, μη έχοντας αυτά τα πρόσθετα έξοδα, το ποσό του λογαριασμού ενός χρήστη θα αντιπροσωπεύει στο 100% τα χρήματα που του αναλογούν. Τέλος, ορίζοντας το δίκτυο όπως θέλουμε, μπορούμε να πιστώσουμε με Ether το λογαριασμό του διαχειριστή κατά την υλοποίηση του δικτύου, δίνοντάς του έτσι τη δυνατότητα να πιστώσει με τη σειρά του το contract της εφαρμογής άμεσα. Περισσότερες λεπτομέρειες ως προς τις οικονομικές παραμέτρους της εφαρμογής παρακάτω.

Ένας άλλος λόγος της παραπάνω απόφασης είναι η επιλογή των μελών του δικτύου, καθώς και η έλλειψη ανωνυμίας αυτών. Στη συγκεκριμένη περίπτωση δεν είναι επιθυμητή η ελεύθερη πρόσβαση στην πληροφορία του συστήματος, άρα δεν μπορεί ο οποιοσδήποτε να γίνει κόμβος του δικτύου. Αφενός γιατί δεν είναι έμπιστο άτομο ως προς την διαδικασία της ψηφοφορίας των blocks, και αφετέρου επειδή δεν θα έπρεπε να έχει πρόσβαση στο ledger ολόκληρου του δικτύου. Επιπροσθέτως, φιλτράροντας την είσοδο των χρηστών του δικτύου, δεν υπάρχει ανωνυμία. Αυτή η έλλειψη ανωνυμίας είναι άκρως απαραίτητη καθώς απαιτείται πλήρης διαφάνεια των ενεργειών μέσα στο δίκτυο. Έτσι, ευθύνες μπορούν να αποδοθούν σε όποιους χρήστες δράσουν εναντίων του συστήματος.

Τελευταίο, αλλά εξίσου σημαντικό, είναι η αυξημένη ταχύτητα του δικτύου. Η ταχύτητα αυτή εξαλείφει τους νεκρούς χρόνους οι οποίοι σε ένα δημόσιο δίκτυο θα αποτελούσαν πρωταρχικό πρόβλημα με κυριότερο πλήγμα την αποδοτικότητα.

4.3.2 Δομή Κόμβων

Επόμενο βήμα είναι να αποφασιστεί η δομή των κόμβων που θα διέπουν το δίκτυο. Έχοντας στο νου μας ότι ο διαχειριστής της συγκεκριμένης εφαρμογής θα είναι ο εκάστοτε υπεύθυνος ασφαλιστικός φορέας, κάνουμε μία υπόθεση εργασίας ότι ο ΕΦΚΑ θα έχει αυτόν το ρόλο. Η πιο λογική επιλογή είναι να εγκατασταθεί από ένας πλήρης κόμβος (full node) σε κάθε υποκατάστημα ΕΦΚΑ. Έτσι όλοι οι χρήστες είναι γνωστοί, έμπιστοι, και διαμοιράζοντας το blockchain μεταξύ τους δεν αρκεί η επιτυχής εισβολή στο σύστημα ενός μόνο, καθώς με την αξιοποίηση της τεχνολογίας blockchain και το σύστημα ψηφοφορίας των κόμβων, χρειάζεται πλειοψηφικός έλεγχος αυτών για την τελική αλλοίωση του συστήματος. Πρόσθετα light nodes μπορούν να υπάρξουν στο κάθε υποκατάστημα, ανάλογα με τις απαιτήσεις του δικτύου, για την ταχύτερη εξυπηρέτηση των συναλλαγών.

4.3.3 Οικονομικές Συνιστώσες

Ένα βασικό τμήμα της εφαρμογής αυτής είναι η δυνατότητα ολοκλήρωσης των χρηματικών συναλλαγών άμεσα εντός του blockchain. Ο ιδιωτικός τύπος δικτύου που προεπιλέξαμε μας δίνει τη

δυνατότητα να εισάγουμε ένα οποιοδήποτε ποσό Ether κατά τη γέννησή του, καθιστώντας έτσι άμεση τη λειτουργία της εφαρμογής και παρωχημένη τη διαδικασία του mining.

Η λογική του δικτύου είναι ένας κύκλος όπου τα κρυπτονομίσματα θα ρέουν από το λογαριασμό του διαχειριστή στο smart contract της εφαρμογής, από την εφαρμογή στους λογαριασμούς των φαρμακείων και τέλος από τους λογαριασμούς των φαρμακείων πίσω στο λογαριασμό του διαχειριστή. Η τελευταία εκ των προηγουμένων τριών συναλλαγών είναι και λίγο ιδιαίτερη. Αυτό επειδή κατά την επιστροφή των wei από τα φαρμακεία στο διαχειριστή, θα γίνεται και η πραγματική πληρωμή αυτών σε Ευρώ. Κάτι τέτοιο μπορεί να πραγματοποιηθεί μέσω τράπεζας και ειδικής συμφωνίας μεταξύ αυτής και του ΕΦΚΑ. Ο ιδιοκτήτης του εκάστοτε φαρμακείου θα πηγαίνει στην τράπεζα και θα χρησιμοποιεί ένα δικό της wallet για να στείλει το υπόλοιπο του Ethereum λογαριασμού του στο λογαριασμό του διαχειριστή. Η τράπεζα έχοντας υπάρξει μάρτυρας της συναλλαγής αυτής, θα μεταφέρει τα αντίστοιχα χρήματα από τραπεζικό λογαριασμό του ΕΦΚΑ σε αυτόν του εκάστοτε ιδιώτη.

Η αξία του Ether, και γενικότερα των κρυπτονομισμάτων, σε ένα δίκτυο καθορίζεται από τους χρήστες του. Η παρούσα εργασία και εφαρμογή, αντιστοιχίζει 1 wei σε 1 λεπτό του Ευρώ, δηλαδή 100 wei ισούνται με 1 Ευρώ. Το wei είναι υπομονάδα του Ether, συγκεκριμένα: $\mathbf{1}\ \boldsymbol{wei} = \mathbf{10^{-18}}\ \boldsymbol{Ether}$. Με άλλα λόγια 1 Ether ισούται μια 1 πεντάκις εκατομμύριο wei. Τώρα με τις παραπάνω αντιστοιχίες συμπεραίνουμε ότι 1 Ether στην παρούσα εφαρμογή, αξίζει 10 τετράκις εκατομμύρια Ευρώ… Συνεπώς, μπορούμε να αντιληφθούμε ότι με μια αρχική εισαγωγή 100 Ether στο δίκτυο, το προμηθεύουμε με υπεραρκετή ποσότητα για να υποστηρίξει την εφαρμογή πρακτικά επ’ άπειρον.

5 Εργαλεία υλοποίησης των Εφαρμογών

Ο σχεδιασμός της εφαρμογής αποτελείται από 2 βασικά τμήματα. Την συγγραφή και ανάρτηση των smart contracts σε ένα τοπικό blockchain. Την ανάπτυξη ενός ιστότοπου για να αλληλεπιδρούν οι χρήστες με την εφαρμογή.

5.1 Smart Contracts / Blockchain

5.1.1 Node.js

Το Node.js [*12*] είναι JavaScript περιβάλλον χρόνου εκτέλεσης ανοιχτού κώδικα. Δημιουργήθηκε το 2009 από τον Ryan Dahl και έχει ως σκοπό να εκτελεί server-side JavaScript κώδικα. Έχοντας event-driven αρχιτεκτονική, παρέχεται η δυνατότητα για ασύγχρονες διεργασίες που μπορούν να εκτελούνται παράλληλα χωρίς να χρειάζεται η ολοκλήρωση της πρώτης ώστε να ξεκινήσει η δεύτερη.

5.1.2 Npm

Το npm, [*12*] ή αλλιώς Node package manager, είναι ένας διαχειριστής πακέτων του Node.js. Προστέθηκε στο Node το 2010 και η λειτουργία του είναι να απλοποιεί τη διαδικασία δημοσίευσης και εγκατάστασης Node.js πακέτων, ώστε να μπορούν να ενσωματωθούν εύκολα και γρήγορα σε διάφορα projects. Το npm αντλεί από το μητρώο του όλα τα διαθέσιμα πακέτα που έχουν δημοσιευθεί από χρήστες, και έτσι μπορεί κάποιος άλλος να τα εγκαταστήσει με μια εντολή μέσω αυτού. Για τη συγκεκριμένη εργασία πολλά από τα εργαλεία που αξιοποιούνται εγκαταστάθηκαν μέσω του npm.

5.1.3 Remix IDE

Το Remix είναι ένας ιστότοπος που παρέχει χρήσιμες λειτουργίες για smart contract developers. Αποτελείται από έναν text editor, έναν compiler και έναν προσομοιωτή blockchain. Οι χρήστες μπορούν να γράψουν τα smart contract τους στον ειδικά διαμορφωμένο text editor και μετά να τα κάνουν compile, επιλέγοντας την έκδοση του compiler και έχοντας άμεσο feedback για τυχόν σφάλματα στον κώδικα. Στο τέλος τα contracts μπορούν να γίνουν deployed στο Remix, το οποίο παρέχει ένα απλό γραφικό περιβάλλον αλληλεπίδρασης με τα contracts, όπου ο χρήστης μπορεί να κάνει δοκιμές και να εξακριβώσει ότι λειτουργούν όπως αναμένεται.

5.1.4 Truffle

Το Truffle είναι ένα περιβάλλον ανάπτυξης Ethereum εφαρμογών. Προσφέρει δυνατότητες smart contract compilation, deployment και testing. Αποτελεί ουσιαστικά ένα εργαλείο που είναι υπεύθυνο για την «ανάρτηση» των smart contracts σε ένα blockchain.

5.1.5 Ganache

Το Ganache είναι ένας τοπικός προσομοιωτής blockchain. Δίνει πληθώρα δυνατοτήτων στον χρήστη ώστε να μπορεί να δοκιμάσει τις εφαρμογές του πάνω σε αυτό. Συμπεριλαμβάνει και ένα γραφικό περιβάλλον στο οποίο μπορεί κανείς εύκολα να πλοηγηθεί και να αποκτήσει πρόσβαση σε χρήσιμες

πληροφορίες για την παρούσα κατάσταση του blockchain. Κατά τη γέννηση ενός νέου δικτύου το Ganache δημιουργεί και μερικά accounts με κάποιο ποσό Ether, ώστε να μπορεί ο χρήστης να κάνει συναλλαγές και γενικότερα να αλληλεπιδρά με τις εφαρμογές του. Από αυτά τα accounts το ganache μας δίνει όλες τις πληροφορίες όπως τις διευθύνσεις τους αλλά και τα private keys τους.

5.2 Front-End

5.2.1 React.js

Η React.js είναι μία βιβλιοθήκη JavaScript ειδικά σχεδιασμένη για την ανάπτυξη διεπαφών χρήστη. Η κύρια δομή της React είναι component-based, δηλαδή το κάθε στοιχείο του UI ορίζεται ως στοιχείο και έπειτα χρησιμοποιείται. Αυτό έχει ως αποτέλεσμα ο κώδικας να είναι δομικά ευανάγνωστος και το debugging, ως κατά συνέπεια, ευκολότερο. Η React έχει το δικό της πακέτο που μπορεί να εγκατασταθεί μέσω του npm και προσφέρει στον χρήστη ένα περιβάλλον που συμπεριλαμβάνει τον δικό του debugger και εμφανίζει σφάλματα κατά την ανάπτυξη. Έτσι ο σχεδιασμός ιστοτόπων απλοποιείται και επιταχύνεται αρκετά. Συντηρείται από το Facebook.

5.2.2 React-Bootstrap

Η Bootstrap είναι μια πασίγνωστη βιβλιοθήκη HTML, CSS και JavaScript, που συναντάται σε πληθώρα ιστοσελίδων. Η React-Bootstrap είναι μια ανανεωμένη έκδοση της Bootstrap όπου ξαναδημιουργεί κάθε στοιχείο της ως ένα React component, και έτσι μπορεί να χρησιμοποιηθεί απευθείας σε ένα React project. Η React-Bootstrap είναι και αυτή στο μητρώο του npm και μπορεί να εγκατασταθεί μέσω αυτού.

5.2.3 React-Router

Η React-Router είναι μια συλλογή από components τα οποία είναι υπεύθυνα για λειτουργίες πλοήγησης. Η κύρια χρήση της είναι να εμφανίζει δυναμικά το περιεχόμενο μίας ιστοσελίδας ανάλογα με το path του URL της. Εγκατάσταση μέσω npm.

5.2.4 Web3.js

Η Web3.js είναι μια βιβλιοθήκη JavaScript με σκοπό την εύκολη επικοινωνία μεταξύ ενός Ethereum node και ενός ιστότοπου. Ουσιαστικά είναι ο κρίκος που συνδέει το γραφικό περιβάλλον μίας αποκεντροποιημένης εφαρμογής με τα smart contracts που τη διέπουν. Όπως και οι προηγούμενες βιβλιοθήκες, η Web3.js μπορεί να εγκατασταθεί μέσω του npm.

5.2.5 MDBootstrap (React)

Η MDBootstrap (Material Design for Bootstrap) είναι μια βιβλιοθήκη από components συμβατά με πολλά διαφορετικά frameworks. Η έκδοση που χρησιμοποιείται στην παρούσα εργασία είναι αυτή για React και εγκαταστάθηκε μέσω του npm. Από τη συγκεκριμένη βιβλιοθήκη αξιοποιήθηκαν κάποια Datatables, τα οποία έχουν δυνατότητες αναζήτησης και είναι υπεύθυνα για την προβολή σημαντικών πληροφοριών της εφαρμογής.

5.2.6 Moment.js

Η Moment.js είναι μια βιβλιοθήκη JavaScript η οποία παρέχει δυνατότητες επεξεργασίας ημερομηνιών και χρόνου. Χρησιμοποιήθηκε για την επεξεργασία και προβολή timestamps στην εφαρμογή ώστε να παρέχει ένα χρονικά κατανεμημένο ιστορικό.

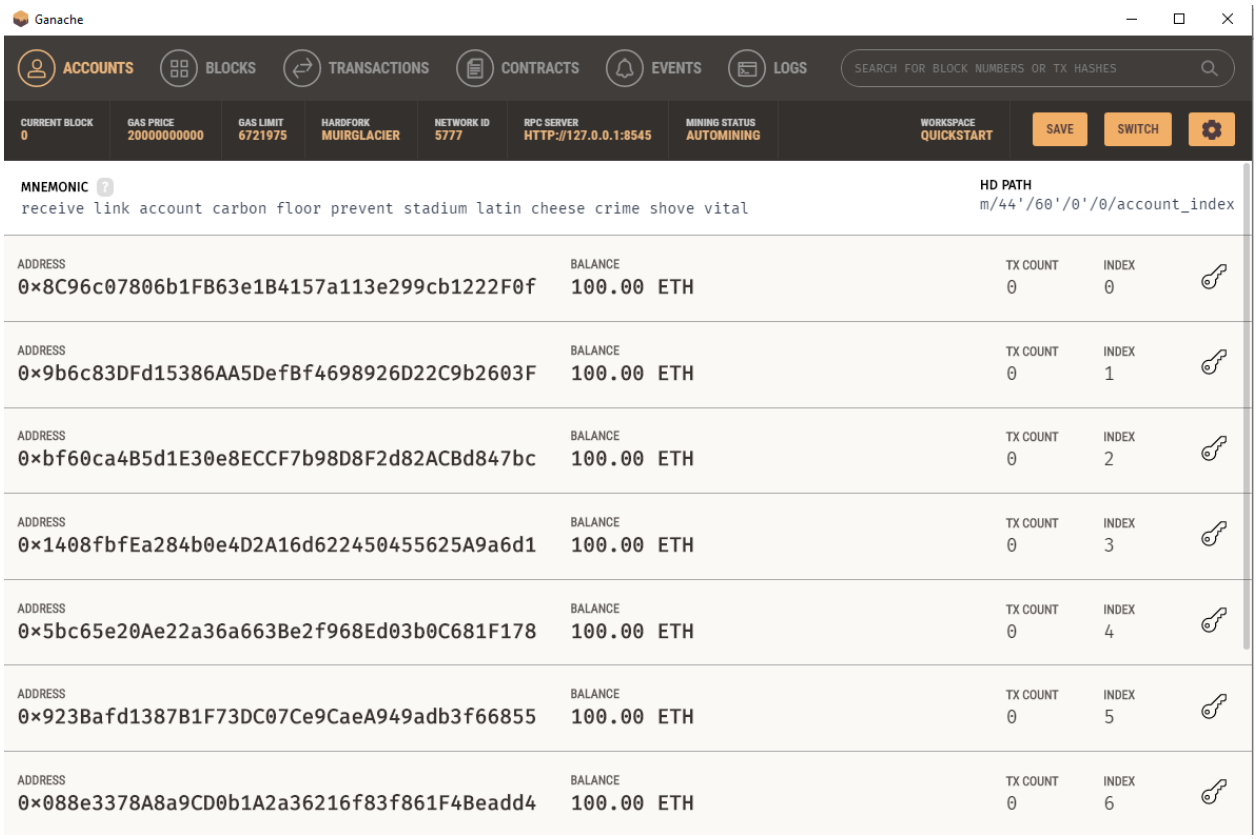
5.2.7 MetaMask

Το MetaMask είναι ένα wallet σχεδιασμένο για τη χρήση blockchain εφαρμογών. Όπως προαναφέρθηκε, ένα wallet «υπογράφει» συναλλαγές με το private key ενός λογαριασμού το οποίο εάν είναι σωστό ολοκληρώνει τη συναλλαγή. Το MetaMask μπορεί να συνδεθεί στο κύριο network του Ethereum καθώς και σε κάποια άλλα test δίκτυα. Επίσης μπορεί να συνδεθεί και σε custom δίκτυα όπως για παράδειγμα αυτό του Ganache. Τέλος, παίρνοντας τα private keys από τα accounts του Ganache, μπορούμε να τα κάνουμε import στο MetaMask και να αλληλεπιδράσουμε με το blockchain μέσω αυτών.

6 Παρουσίαση Εφαρμογών

6.1 Smart Contract Deployment

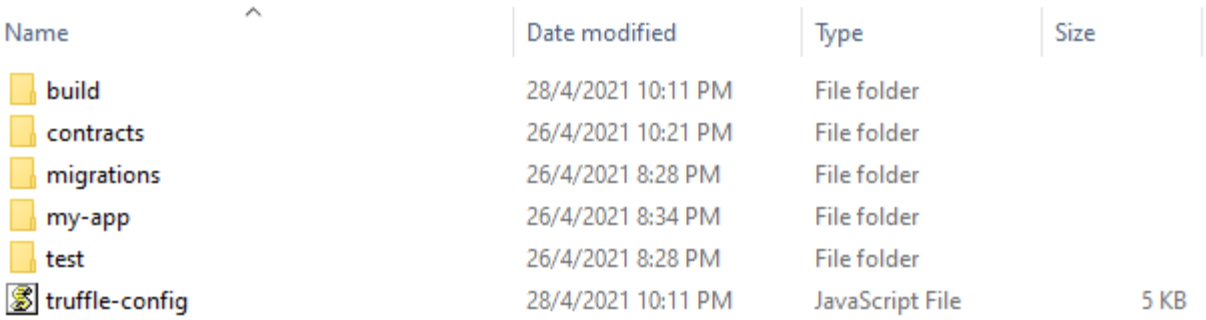
Αφότου συντάξουμε τα smart contracts θέλουμε να τα «ανεβάσουμε» στο blockchain. Όπως προαναφέρθηκε, χρησιμοποιείται το Ganache για να προσομοιώσει το blockchain αυτό. Ανοίγοντας το Ganache πατάμε στην επιλογή quickstart η οποία μας οδηγεί στην παρακάτω οθόνη:



Εικόνα 1: Ganache interface

Εδώ υπάρχουν αρκετές πληροφορίες για το blockchain αλλά θα εξερευνήσουμε τις πιο σημαντικές. Έχουμε 10 accounts προμηθευμένα με 100 Ether το καθένα και πρόσβαση σε όλες τις πληροφορίες που τα απαρτίζουν, όπως διεύθυνση και private key. Στην κορυφή υπάρχουν 2 πεδία με ονόματα NETWORK ID και RPC SERVER, αυτά θα τα χρειαστούμε στη συνέχεια για το deployment των contracts και τη σύνδεση του wallet στο δίκτυο. Είναι ουσιαστικά τα κύρια στοιχεία που αναγνωρίζουν το blockchain δίκτυο. Η παραπάνω διεύθυνση είναι η localhost, η τοπική δηλαδή του υπολογιστή.

Το επόμενο βήμα είναι να χρησιμοποιήσουμε το Truffle για να κάνει compile τα contracts και έπειτα deploy στο Ganache. Για να είναι λίγο πιο κατανοητά τα παρακάτω, ας δούμε λίγο τη δομή του project με τους φακέλους του και πως αυτοί δημιουργήθηκαν:



Name	Date modified	Type	Size
build	28/4/2021 10:11 PM	File folder	
contracts	26/4/2021 10:21 PM	File folder	
migrations	26/4/2021 8:28 PM	File folder	
my-app	26/4/2021 8:34 PM	File folder	
test	26/4/2021 8:28 PM	File folder	
truffle-config	28/4/2021 10:11 PM	JavaScript File	5 KB

Εικόνα 2: Δομή Project

Στην αρχή, μέσω της κονσόλας, έγιναν 2 αρχικοποιήσεις. Μία για το React τμήμα του project, δηλαδή το front-end, και μία για το Truffle το οποίο δημιουργεί τα απαραίτητα αρχεία που θα χρειαστεί για τις λειτουργίες του. Το truffle-config.js είναι το αρχείο που ορίζει και το deployment των contracts από το Truffle. Εκεί μπορούμε να δηλώσουμε ένα δίκτυο υπό ένα όνομα και μετά να ζητήσουμε από το Truffle να κάνει deploy τα contracts σε αυτό το όνομα δικτύου. Επομένως εδώ χρησιμοποιούμε τα στοιχεία του blockchain που πήραμε από το Ganache και μετά, μέσω της κονσόλας, λέμε στο Truffle να κάνει compile και deploy τα contracts εκεί. Από τη στιγμή που δεν προσδιορίζουμε συγκεκριμένο account, το Truffle χρησιμοποιεί το πρώτο του Ganache για το deployment.

Πλέον το μόνο πράγμα που μένει είναι να φτιάξουμε το front-end και να το κάνουμε να επικοινωνήσει με το blockchain και τα contracts. Όσων αφορά το δίκτυο όλα είναι έτοιμα, ο κώδικας είναι ανεβασμένος και περιμένει να εκτελεστεί όταν κληθεί.

6.2 Κώδικας

6.2.1 Smart Contracts

Τα αρχεία των smart contracts στη συγκεκριμένη εφαρμογή είναι 3. Θεωρητικά ολόκληρη η εφαρμογή θα μπορούσε να υλοποιηθεί μόνο με ένα αρχείο αλλά αυτό θα έκανε τον κώδικα δυσανάγνωστο και δύσκολο ως προς το debugging. Όλα τα αρχεία είναι γραμμένα σε Solidity.

6.2.1.1 Admin.sol

Το Admin.sol είναι το πρώτο και μεγαλύτερο από τα contracts. Εμπεριέχει όλες τις μεθόδους αποθήκευσης της εφαρμογής καθώς και όλες τις λειτουργίες διαχειριστή. Ακολουθεί η παρουσίασή του σε τμήματα και η σταδιακή τους επεξήγηση μαζί με αυτή των όρων.

```
pragma solidity >=0.6.12;

contract Admin {

  address private admin;

// Drug Stores
  mapping (address => DrugStore) internal drug_stores; // Key: address
  struct DrugStore {
    uint AFM;
    bool status;
  }
  function addDrugStore(uint AFM, address _address) public isAdmin{
    drug_stores[_address] = DrugStore(AFM, true);
  }
  function deactivateDrugStore(address _address) public isAdmin{
    drug_stores[_address].status=false;
  }
  function reactivateDrugStore(address _address) public isAdmin{
    require (drug_stores[_address].AFM!=0, "Drug store does not exist in the system.");
    drug_stores[_address].status=true;
  }
  modifier isDrugStore(){
    require (drug_stores[msg.sender].status==true, "This drug store is not verified by the system yet.");
    _;
  }

                                                    [...]
```

Ένα αρχείο Solidity ξεκινάει πάντα ορίζοντας την έκδοση/εκδόσεις του compiler, με τους οποίους είναι συμβατό. Αφού γίνει αυτό ορίζεται το contract και μέσα σε αυτό βρίσκονται οι λειτουργίες του. Στο παραπάνω απόσπασμα φαίνονται ορισμένες λέξεις κλειδιά:

- **Function**

  Είναι οι συναρτήσεις των contracts. Αυτές καλούνται από το front-end και εκτελούν όλες τις λειτουργίες της εφαρμογής.

- **Modifier**

  Είναι ένα άλλο είδος συνάρτησης το οποίο μπορεί να «κολλήσει» σε μια κανονική function και να κληθεί αυτομάτως πριν την κάθε της εκτέλεση.

- **Require**

  Πραγματοποιεί έναν έλεγχο μίας συνθήκης, ο οποίος εάν είναι ψευδής, ακυρώνει την εκτέλεση της συνάρτησης, επιστρέφει στο account που την κάλεσε το όποιο ποσό του είχε κοστίσει, και τέλος εμφανίζει ένα μήνυμα σφάλματος.

- **Struct**

  Τα structs θυμίζουν τις κλάσεις σε άλλες αντικειμενοστραφείς γλώσσες προγραμματισμού (πχ C++). Επομένως αφού ορίσουμε ένα struct μπορούμε να δημιουργήσουμε και αντικείμενα αυτού.

- **Mapping**

  Τα mappings αποτελούν έναν τρόπο αποθήκευσης δεδομένων. Ουσιαστικά αντιστοιχίζουν ένα τύπο δεδομένων με έναν άλλον. Ο πρώτος έχει τον ρόλο κλειδιού/pointer ενώ ο δεύτερος είναι η αποθηκευμένη πληροφορία. Στο συγκεκριμένο τμήμα κώδικα, αποθηκεύονται αντικείμενα του struct “DrugStore” στο mapping “drug_stores”, έχοντας ένα address (διεύθυνση Ethereum) ως το κλειδί τους.

  Να σημειωθεί πως πλέον όταν χρησιμοποιούμε τον όρο «κλειδί» ***δεν*** *αναφερόμαστε στα private και public keys που εξερευνήσαμε στα προηγούμενα κεφάλαια.*

Έχοντας υπόψη τα παραπάνω, το τμήμα αυτό του κώδικα αποτελείται από 3 βασικές συναρτήσεις υπεύθυνες για την καταχώρηση/ενημέρωση χρηστών «Φαρμακεία», την απενεργοποίηση και την επανενεργοποίηση των λογαριασμών τους.

```
                                                    […]

// Doctors
  mapping (address => Doctor) internal doctors; // Key: address
  struct Doctor {
    uint AFM;
    bool status;
  }
  function addDoctor(uint AFM, address _address) public isAdmin{
    doctors[_address] = Doctor(AFM, true);
  }
  function deactivateDoctor(address _address) public isAdmin{
    doctors[_address].status=false;
  }
  function reactivateDoctor(address _address) public isAdmin{
    require (doctors[_address].AFM!=0, "Doctor does not exist in the system.");
    doctors[_address].status=true;
  }
  modifier isDoctor(){
    require (doctors[msg.sender].status==true, "You are not a doctor.");
    _;
  }

                                                    […]
```

Πάλι έχουμε ακριβώς την ίδια δομή αλλά αυτή τη φορά για τους χρήστες «Ιατροί».

```
[…]

// Patients
  mapping (uint => Patient) internal patients; // Key: AMKA
  struct Patient {
    string AT;
    string name;
    string surname;
    bool insurance;
    bool flag;
  }
  function addPatient(uint AMKA, string memory AT, string memory name, string memory surname)
public isAdmin{
    patients[AMKA] = Patient(AT,name,surname,true, true);
  }
  function deactivatePatient(uint AMKA) public isAdmin{
    patients[AMKA].insurance=false;
  }
  function reactivatePatient(uint AMKA) public isAdmin{
    require (patients[AMKA].flag==true, "Patient does not exist in the system.");
    patients[AMKA].insurance=true;
  }

[…]
```

Και εδώ έχουμε την ίδια δομή με ελαφρώς τροποποιημένο struct. Το συγκεκριμένο αφορά τους ασθενείς και την κατάσταση της ασφάλισής τους. Οι ασθενείς **δεν** αποτελούν χρήστες της εφαρμογής, ωστόσο η ύπαρξή τους στο σύστημα είναι απαραίτητη για να συνταγογραφηθούν φάρμακα υπό τον ΑΜΚΑ τους.

```
                                                [...]

// Drugs
  mapping (uint => Drug) internal drugs; // Key: barcode
  struct Drug {
    uint barcode;
    string description;
    uint price;
  }
  function addDrug(uint barcode, string memory description, uint price) public isAdmin{
    drugs[barcode] = Drug(barcode, description, price);
  }
  function removeDrug(uint barcode) public isAdmin{
    delete drugs[barcode];
  }

// Orders
  mapping (uint => Order) internal orders; // Key: AMKA
  struct Order {
    Drug[] meds;
    uint[] quantity;
    bool status;
  }

                                                [...]
```

Εδώ ορίζονται οι 2 τελευταίες οντότητες της εφαρμογής, τα φάρμακα και οι παραγγελίες (συνταγές). Τα φάρμακα ορίζονται αρκετά απλά με κάποια χαρακτηριστικά, εκ των οποίων το barcode χρησιμοποιείται και σαν ID στη λογική της παρούσας εφαρμογής. Οι παραγγελίες ωστόσο είναι ελαφρώς πιο σύνθετες. Η βασική τους πληροφορία αποθηκεύεται σε 2 πίνακες. Ο πρώτος κρατάει αντικείμενα του struct “Drug” ενώ ο δεύτερος την ποσότητα του εκάστοτε. Στη συνέχεια του κώδικα, ακολουθείται τέτοια λογική όπου κατά την καταχώρηση παραγγελιών, οι 2 πίνακες γεμίζουν με το ίδιο index. Οπότε οι πίνακες είναι πάντα αντιστοιχισμένοι.

```
                                    […]

constructor() public payable{
    admin = msg.sender;
  }

  modifier isAdmin(){
    require(msg.sender == admin, "You are not the admin.");
    _;
  }

  function bank() payable public isAdmin{
  }
}
```

Τέλος ας εξηγήσουμε κάποιους τελευταίους όρους:

- Constructor

Ο constructor, γνωστός και ως συνάρτηση κατασκευαστή, είναι μία συνάρτηση που καλείται κατά τη δημιουργία ενός contract, δηλαδή κατά το deployment του σε ένα blockchain. Χρησιμοποιείται γενικά για βασικές αρχικοποιήσεις, συνήθως ζωτικής σημασίας του contract. Εδώ αποθηκεύει τον λογαριασμό από τον οποίον έγινε deployed το contract ως admin, και έτσι του δίνει πρόσβαση στις συναρτήσεις για τις διαχειριστικές λειτουργίες της εφαρμογής μέσω του από κάτω modifier.

- Msg.sender

Συναντήσαμε το msg.sender και στα προηγούμενα τμήματα του κώδικα αλλά ποτέ δεν εξηγήθηκε η χρήση του. Αυτό που κάνει είναι να επιστρέφει τη διεύθυνση του account από το οποίο καλείται η συνάρτηση τη δεδομένη εκείνη χρονική στιγμή. Στο συγκεκριμένο modifier, αλλά και στους παραπάνω, αξιοποιείται για την επαλήθευση του χρήστη καθώς κάνει ελέγχους της διεύθυνσης η οποία, εάν δεν πληροί κάποια κριτήρια, ακυρώνει την εκτέλεση της συνάρτησης μέσω του require στο οποίο εμπεριέχεται.

- Payable

Αυτό το keyword δίνει τη δυνατότητα σε συναρτήσεις να συμπεριλάβουν ένα ποσό Ether το οποίο θα αποθηκευτεί στο contract μετά την επιτυχή τους ολοκλήρωση. Το ποσό μπορεί να το συμπεριλάβει ο χρήστης κατά την κλήση της συνάρτησης μέσω του wallet του. Η συνάρτηση bank χρησιμοποιείται για την πίστωση του contract ώστε να μπορεί αυτό να πραγματοποιεί πληρωμές, όπως θα δούμε στη συνέχεια.

6.2.1.2 Doctors.sol

Το doctors.sol αφορά τις δυνατότητες των ιατρών στην εφαρμογή. Πολλές από τις εντολές που εμπεριέχει το αρχείο αναφέρονται σε οντότητες του admin.sol, οπότε στην αρχή γίνεται imported για να μπορεί να λειτουργήσει κανονικά. Με άλλα λόγια είναι σαν να έχουμε όλο το admin.sol μέσα στο doctors.sol. Το contract Doctors, μετά το import, κληρονομεί από το contract Admin μέσω του keyword "is".

```
pragma solidity >=0.6.12;
import "./admin.sol";

contract Doctors is Admin {

  event DoctorLogs(uint AFM, address adr, uint AMKA, uint barcode, string description, uint quantity, uint timestamp);

  function addOrder(uint AMKA, uint barcode, uint quantity) public isDoctor{
    require (patients[AMKA].insurance==true, "This patient is not insured.");
    require (drugs[barcode].barcode!=0, "Invalid barcode.");
    orders[AMKA].status=true;
    orders[AMKA].meds.push(drugs[barcode]);
    orders[AMKA].quantity.push(quantity);
    emit DoctorLogs(doctors[msg.sender].AFM, msg.sender, AMKA, barcode, drugs[barcode].description, quantity, now);
  }
}
```

- Event / Emit

Αυτοί είναι δύο νέοι όροι που συναντάμε. Ορίζονται με κάποιες παραμέτρους οι οποίες λαμβάνουν ένα όνομα και ένα τύπο δεδομένων. Ένα event μπορεί αργότερα να γίνει emitted και να εξάγει πληροφορίες ίδιου τύπου με αυτές που χρησιμοποιήθηκαν κατά τον ορισμό του. Αυτές οι πληροφορίες είναι προσβάσιμες από κάποιο front-end αλλά όχι από άλλα contracts. Στην παρούσα εφαρμογή τα events χρησιμοποιούνται για να κρατάμε ένα ιστορικό των συνταγών, από ποιους έγιναν, σε ποιους έγιναν, πότε έγιναν κτλ.

Η μοναδική συνάρτηση στο doctors.sol είναι η addOrder, η οποία χρησιμοποιείται για την συνταγογράφηση φαρμάκων σε ασθενείς. Αφότου ελεγχθεί η ασφάλιση του ασθενή και η ύπαρξη του φαρμάκου στο σύστημα, το πρόγραμμα συνεχίζει με την καταχώρηση της παραγγελίας, ενώ στο τέλος φροντίζει να κάνει emit και το αντίστοιχο event.

6.2.1.3 Drug_stores.sol

Με την ίδια λογική γίνεται import το doctors.sol και καταλήγει να υπάρχει μία αλυσίδα από imports όπου το ένα αντλεί από το άλλο. Οπότε στο τέλος χρειάζεται να κάνουμε deploy στο blockchain **μόνο** το drug_stores.sol, και έτσι δημιουργούνται όλα τα contracts σε ένα.

```
pragma solidity >=0.6.12;
import "./doctors.sol";

contract DrugStores is Doctors{

  event DrugStoresLogs(uint AFM, address adr, uint AMKA, uint barcode, string description, uint quantity, uint
price, uint timestamp);

  function execute(uint AMKA) public isDrugStore{
    require(orders[AMKA].status==true , "There are no pending orders under this AMKA.");

    uint amount=0;
    for(uint i=0; i<=orders[AMKA].meds.length-1; i++){
      amount=amount + (orders[AMKA].meds[i].price * orders[AMKA].quantity[i]);
    }
    require(address(this).balance>=amount, "Not enough funds to complete the payment. Try again later.");
    address payable recepient = payable(msg.sender);
    recepient.transfer(amount);

    for (uint i=0; i<=orders[AMKA].meds.length-1; i++){
        emit DrugStoresLogs(drug_stores[msg.sender].AFM, msg.sender, AMKA,
orders[AMKA].meds[i].barcode, orders[AMKA].meds[i].description, orders[AMKA].quantity[i],
orders[AMKA].meds[i].price * orders[AMKA].quantity[i], now);
    }
    delete orders[AMKA];
  }

  function size(uint AMKA) view public returns(uint){
      require(orders[AMKA].status==true, "There are no pending orders under this AMKA.");
      return orders[AMKA].meds.length;
  }

  function showOrder(uint AMKA, uint index) view public isDrugStore returns(string memory, uint){
    string memory drug = orders[AMKA].meds[index].description;
    uint quantity = orders[AMKA].quantity[index];
    return (drug, quantity);
  }}
```

Το drug_stores.sol είναι υπεύθυνο για τις λειτουργίες χρηστών τύπου «Φαρμακεία». Έχει 3 συναρτήσεις και μπορεί να εμφανίζει στον εκάστοτε φαρμακοποιό τα στοιχεία της παραγγελίας ενός ασθενή. Έπειτα αφού τον προμηθεύσει με τα φάρμακα που του αναλογούν, μπορεί να εκτελέσει την παραγγελία στο σύστημα και να πιστωθούν στο λογαριασμό του η αξία των φαρμάκων της παραγγελίας. Η πίστωση γίνεται σε Ether το οποίο μεταφέρεται από το contract στο account του φαρμακείου.

6.2.2 Front-End

Το front-end είναι μία σελίδα που αποτελείται από JavaScript κώδικα (React.js library) ο οποίος επιστρέφει HTML. Δεν θα γίνει αναλυτική περιγραφή του κώδικα καθώς είναι πάρα πολύ μεγάλος, ωστόσο θα εξηγηθούν κάποια βασικά τμήματά του για να γίνει απλά κατανοητή η δομή του.

Ας δούμε για παράδειγμα το κομμάτι που χρησιμοποιεί ο διαχειριστής για να εισάγει έναν ιατρό στο σύστημα. Είναι ουσιαστικά μια φόρμα με 2 πεδία, ένα για τον ΑΦΜ και ένα για το account address. Όταν ο χρήστης πατήσει στο κουμπί υποβολής της φόρμα, καλείται η παρακάτω συνάρτηση JavaScript με παραμέτρους τις τιμές των πεδίων.

```
async function addDoctor(AFM, address){
  const web3 = new Web3(window.ethereum);
  const netId = await web3.eth.net.getId();
  const contract = new web3.eth.Contract(MyContract.abi,MyContract.networks[netId].address);
  contract.methods.addDoctor(AFM, address).send({from:
  window.ethereum.selectedAddress}).then(() => {
    document.getElementById("d1").value='';
    document.getElementById("d2").value='';
  });
}
```

Το MetaMask αφού εγκατασταθεί, κάνει inject το API του στο ***window.ethereum***. Επομένως μέσα στην συνάρτηση JavaScript δημιουργούμε ένα instance της Web3 με στοιχεία που αντλεί από το MetaMask. Τέλος καλείται η επιθυμητή συνάρτηση του contract μέσω του Web3 instance, έχοντας το επιλεγμένο account από το MetaMask σαν τον αποστολέα της συναλλαγής.

Πανομοιότυπη λογική ακολουθείται για όλες τις διαδικασίες της σελίδας.

6.3 Τελικό Αποτέλεσμα

6.3.1 Υποδοχή

Όντας σελίδα που φιλοξενεί εφαρμογή blockchain πρέπει να βεβαιωθεί πως ο χρήστης έχει εγκαταστήσει ένα wallet. Εάν δεν ανιχνεύσει το MetaMask, τότε εμφανίζει ένα επεξηγηματικό μήνυμα μαζί με ένα σύνδεσμο για τη λήψη του.

Εικόνα 3: Μήνυμα Υποδοχής

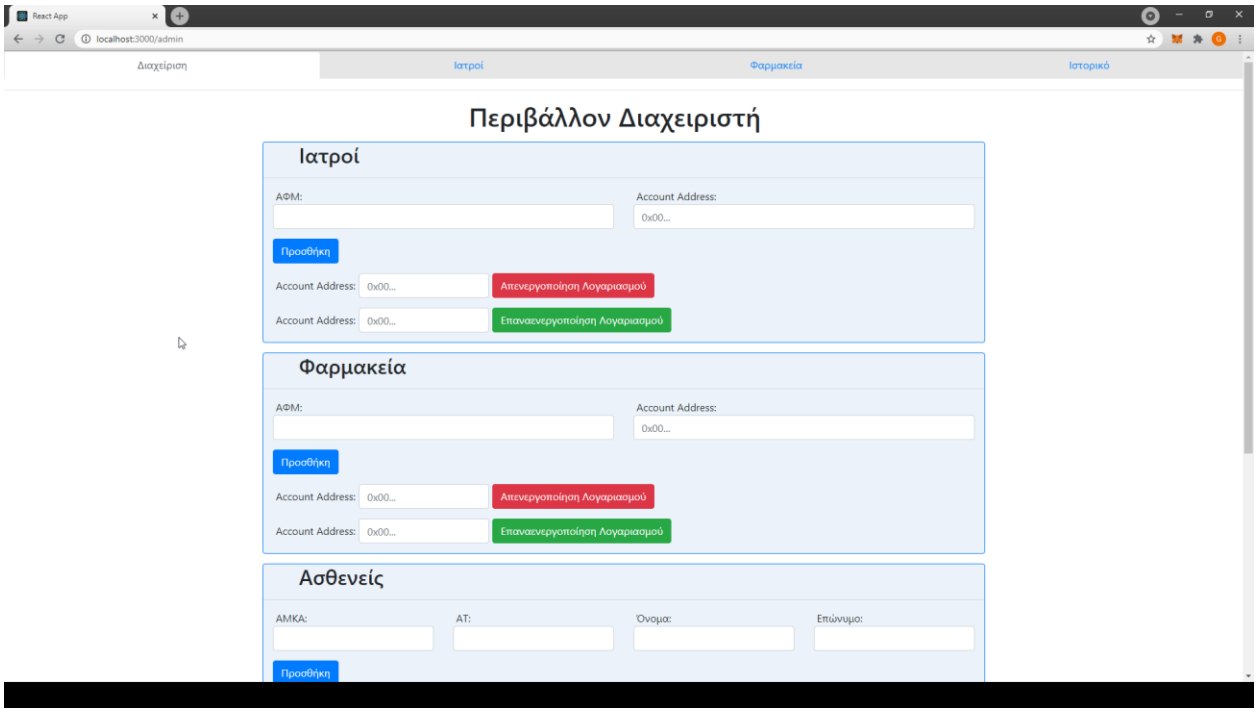
Μετά την εγκατάσταση του MetaMask ο χρήστης πρέπει να συνδεθεί στο δίκτυο. Για να το κάνει αυτό θα πρέπει να το προσθέσει στο MetaMask. Στην παρούσα εργασία θέλουμε να συνδεθούμε τοπικά στο Ganache. Όπως προαναφέρθηκε, το πρώτο account του Ganache χρησιμοποιήθηκε για το deployment των contracts, οπότε είναι και ο διαχειριστής της εφαρμογής. Ανοίγοντας το MetaMask προσθέτουμε το account αυτό, χρησιμοποιώντας το private key του που προμηθευτήκαμε από το Ganache.

Με όλες τις παραπάνω προετοιμασίες, το μήνυμα υποδοχής εξαφανίζεται. Το γραφικό περιβάλλον αποτελείται από μία γραμμή πλοήγησης στην κορυφή που αλλάζει δυναμικά το περιεχόμενο της σελίδας, επιτρέποντας στον χρήστη να περιηγηθεί στα 4 τμήματα της εφαρμογής:

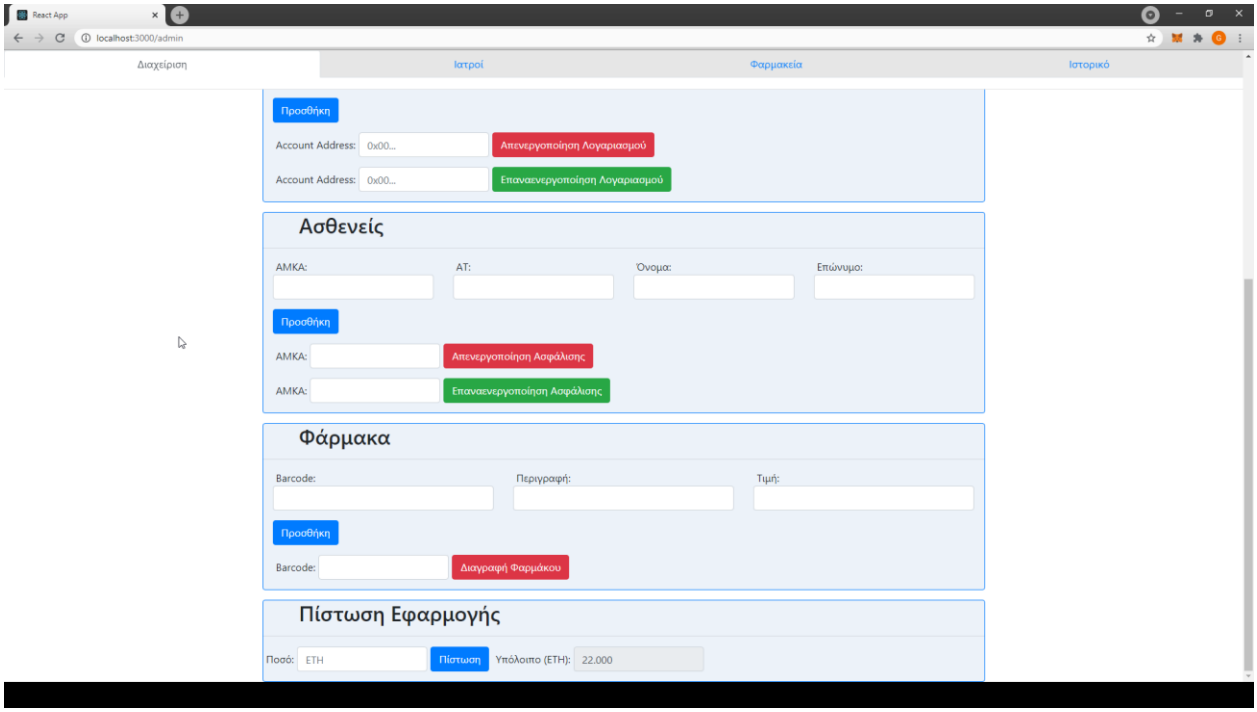
- Το περιβάλλον του διαχειριστή.
- Το περιβάλλον των ιατρών.
- Το περιβάλλον των φαρμακείων.
- Το ιστορικό της εφαρμογής.

6.3.2 Περιβάλλον Διαχειριστή

Εδώ εμπεριέχονται όλες οι λειτουργίες του διαχειριστή. Δίνεται δυνατότητα επεξεργασία των χρηστών (ιατρών/φαρμακείων), φαρμάκων κτλ.

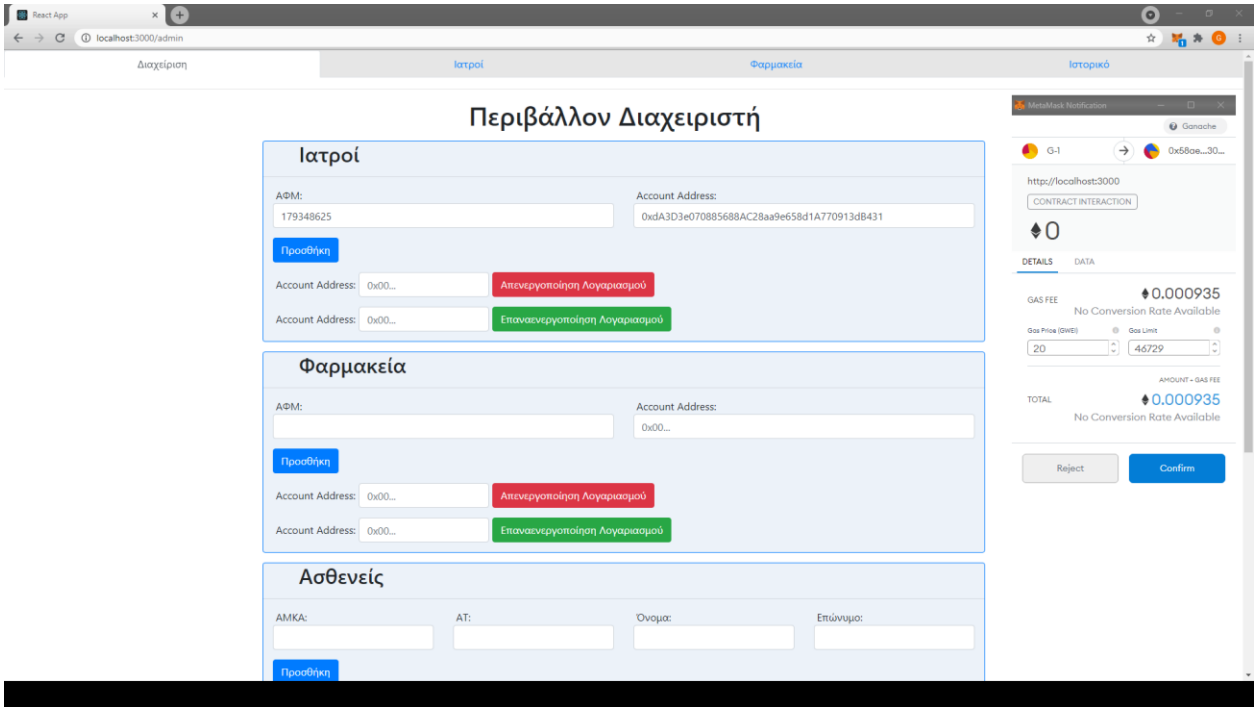


Εικόνα 4: Περιβάλλον Διαχειριστή - 1



Εικόνα 5: Περιβάλλον Διαχειριστή - 2

Εάν επιχειρήσουμε να προσθέσουμε ένα account address ως ιατρό μαζί με έναν ΑΦΜ, θα ανοίξει το MetaMask και θα ζητήσει επιβεβαίωση της συναλλαγής.



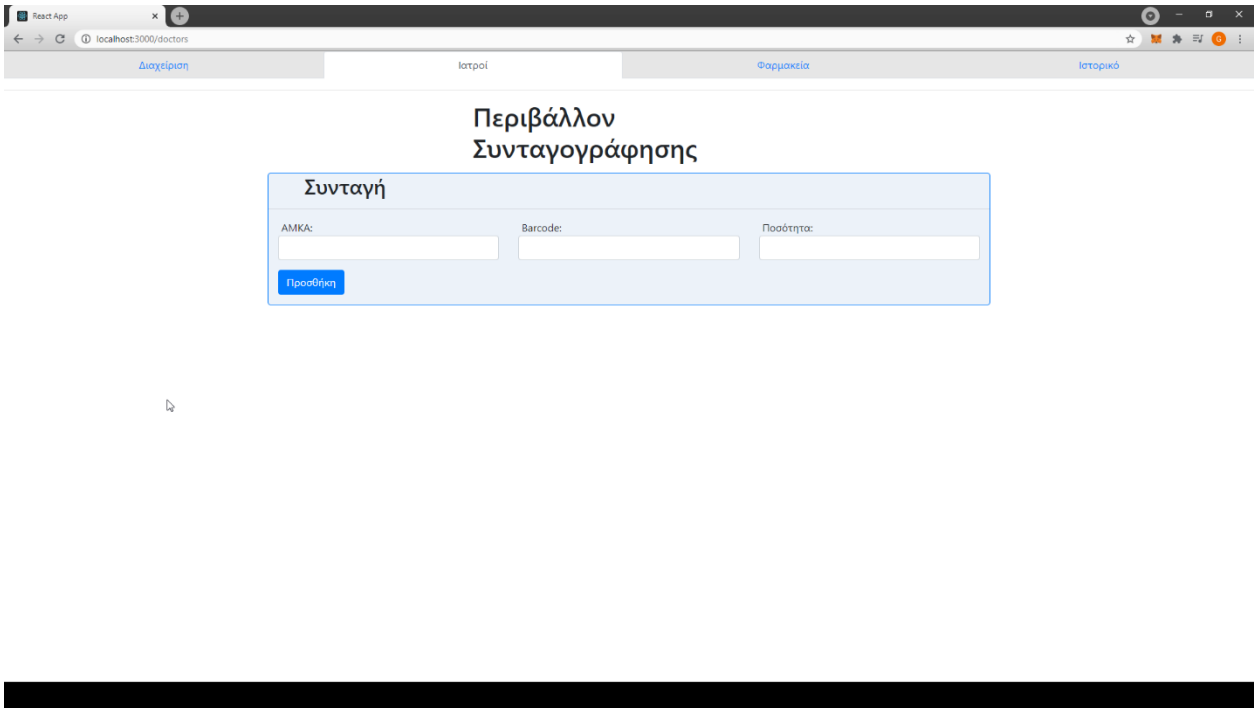
Εικόνα 6: Συναλλαγή Διαχειριστή

Στην περίπτωση που το account που επιχειρούσε τη συναλλαγή **δεν** ήταν αυτό του διαχειριστή, θα εμφανιζόταν μήνυμα σφάλματος από το MetaMask και δεν θα ολοκληρωνόταν η συναλλαγή. Αυτό γίνεται γιατί όλες οι συναρτήσεις των contracts είναι συνοδευόμενες και από έναν modifier, επομένως πάντα γίνεται έλεγχος για το αν ένας χρήστης έχει τα απαραίτητα δικαιώματα που απαιτεί η εκάστοτε λειτουργία. Αντικαθίσταται έτσι η διαδικασία login και η ταυτοποίηση γίνεται εξ ολοκλήρου μέσω των Ethereum accounts και των wallets.

Στο κάτω μέρος της σελίδας γίνεται και η πίστωση του contract, εμφανίζοντας παράλληλα και το ποσό που βρίσκεται ήδη μέσα τη δεδομένη χρονική στιγμή σε Ether.

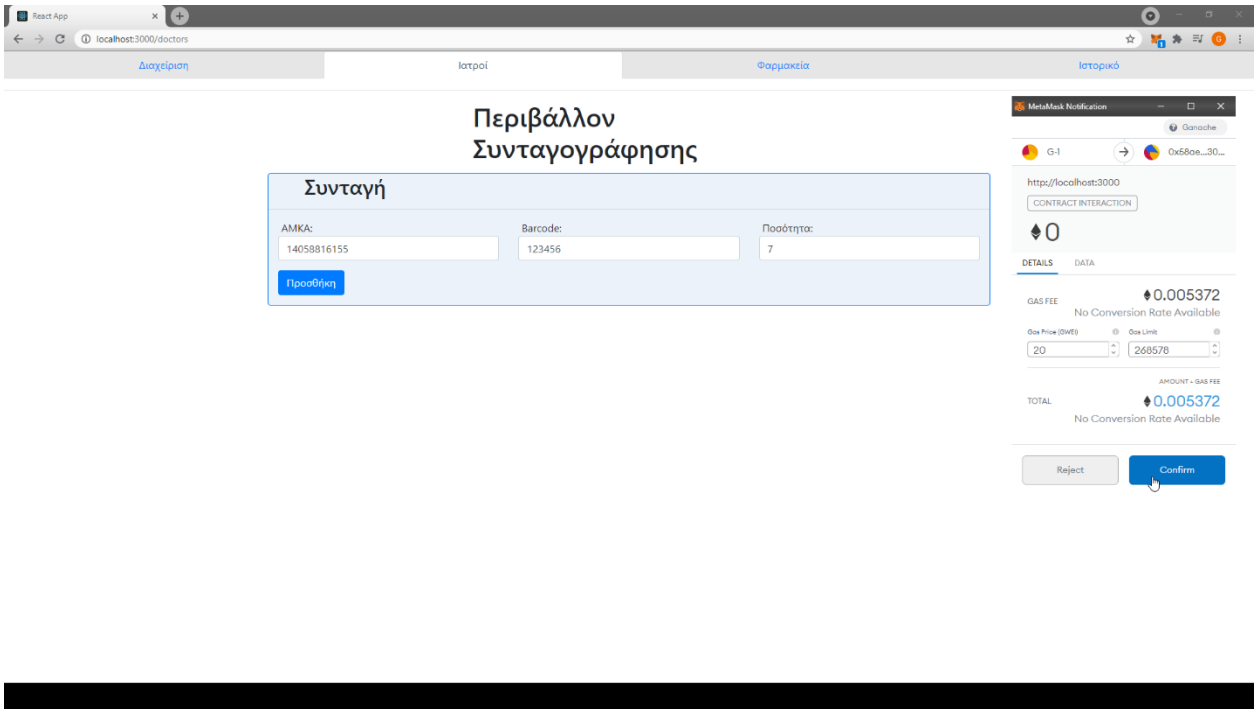
6.3.3 Περιβάλλον Ιατρών

Στο περιβάλλον των ιατρών δίνεται η δυνατότητα συνταγογράφησης. Αποτελείται από μία φόρμα που εισάγεται ο ΑΜΚΑ του ασθενή, το barcode του φαρμάκου και η ποσότητά του.



Εικόνα 7: Περιβάλλον Ιατρών

Παρακάτω βλέπου ένα παράδειγμα χρήσης της φόρμας.



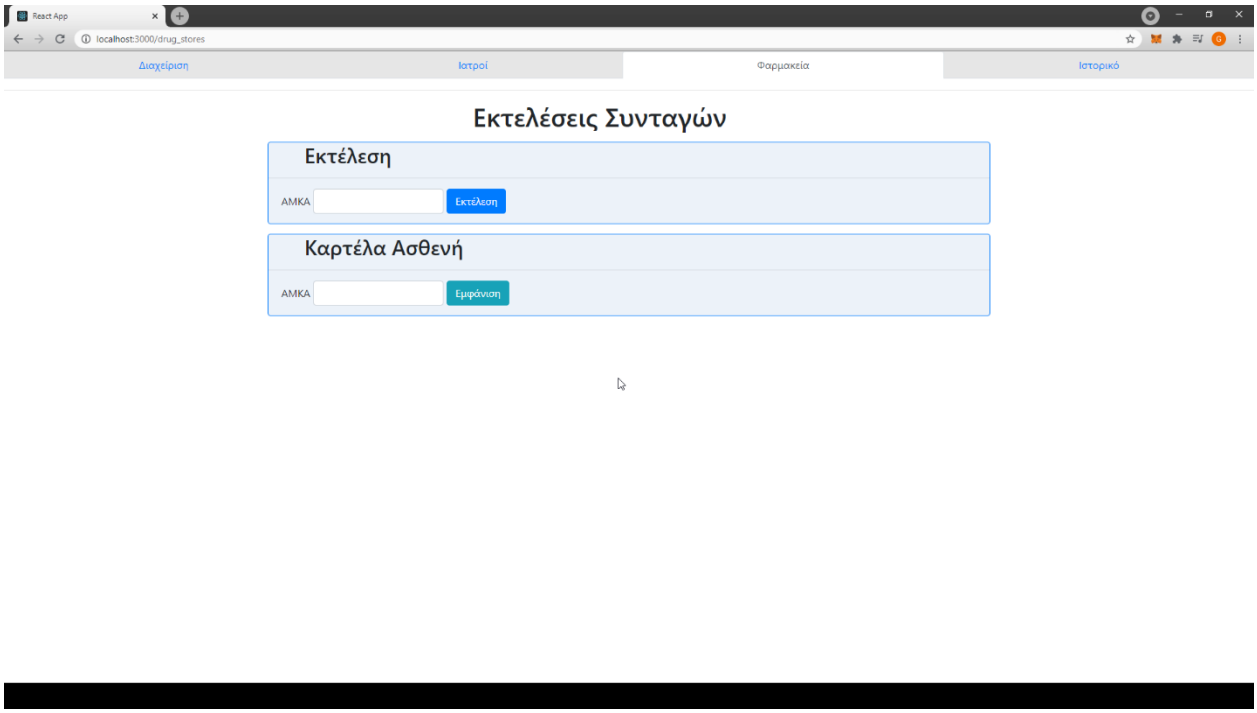
Εικόνα 8: Συναλλαγή Ιατρού

Τα παραπάνω δεδομένα έχουμε φροντίσει να τα περάσουμε από πριν στο σύστημα, όπως έναν ασθενή με ΑΜΚΑ «14058816155» και ένα φάρμακο με barcode «123456». Εάν οποιοδήποτε από τα παραπάνω στοιχεία δεν υπήρχε στο σύστημα, το MetaMask θα εμφάνιζε ανάλογο μήνυμα σφάλματος. Παρομοίως και αν ο χρήστης, δηλαδή το επιλεγμένο account από το MetaMask για τη συναλλαγή, δεν ήταν καταχωρημένο ως ιατρός.

Πριν συνεχίσουμε ας καταχωρήσουμε άλλο ένα φάρμακο στον ίδιο ΑΜΚΑ, αυτή τη φορά με barcode «654321» και ποσότητα 3.

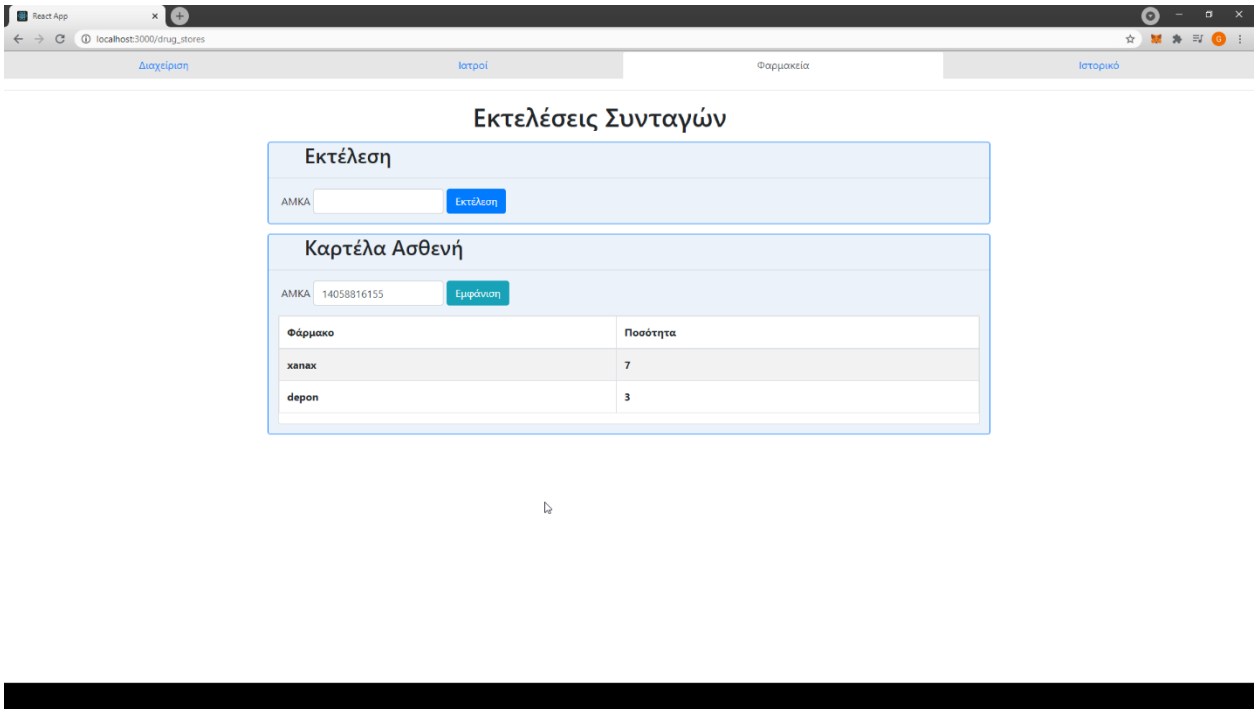
6.3.4 Περιβάλλον Φαρμακείων

Εδώ ένα φαρμακείο μπορεί να δει τις εκκρεμούσες συνταγές ενός ασθενή χρησιμοποιώντας τον ΑΜΚΑ του. Αφού δει τα προϊόντα με τα οποία οφείλει να τον προμηθεύσει, μπορεί να του τα δώσει και έπειτα να εκτελέσει την παραγγελία. Με την εκτέλεσή της διαγράφεται από το σύστημα και το φαρμακείο πληρώνεται άμεσα σε Ether, το οποίο πιστώνεται αυτομάτως στο λογαριασμό του με βάση τις τιμές και ποσότητες των φαρμάκων.



Εικόνα 9: Περιβάλλον Φαρμακείων

Ας εισάγουμε τον ΑΜΚΑ του ασθενή στον οποίο συνταγογραφήσαμε προ ολίγου.



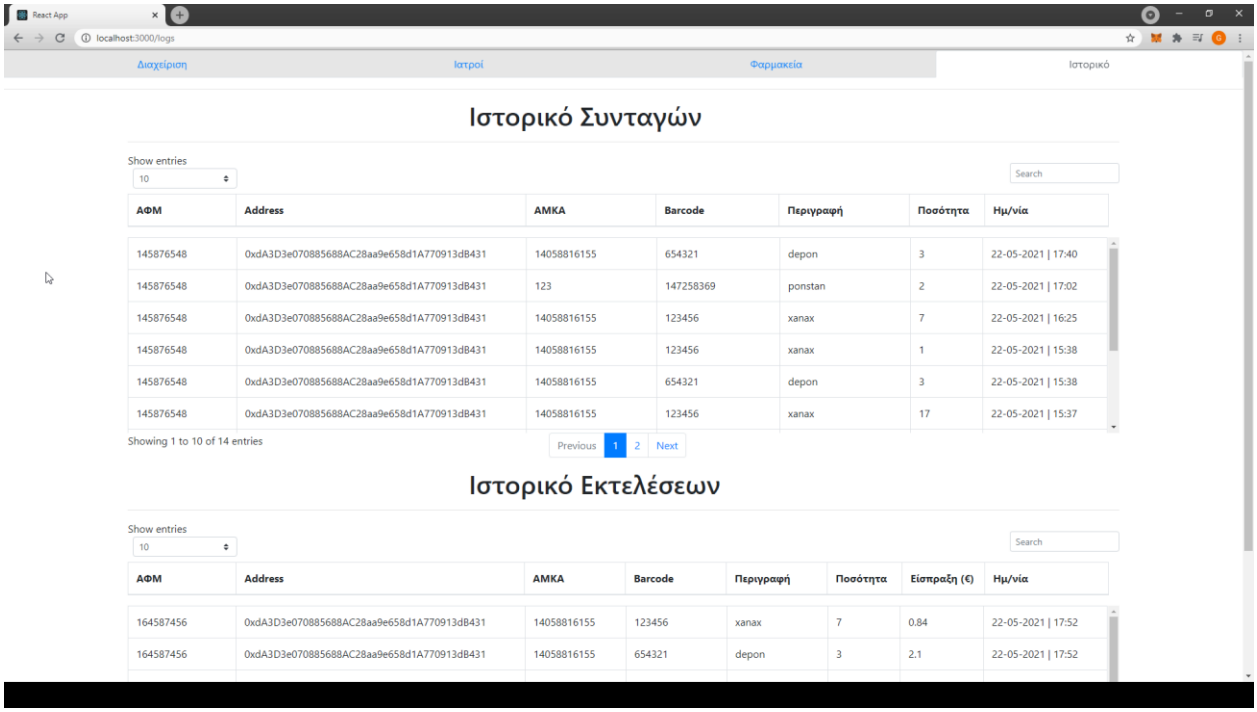
Εικόνα 10: Εμφάνιση Καρτέλας Ασθενή

Όπως παρατηρούμε εμφανίζονται δυναμικά όλες οι καταχωρημένες εγγραφές του συστήματος υπό τον αναγραφόμενο ΑΜΚΑ. Η συγκεκριμένη λειτουργία δεν ανοίγει το MetaMask γιατί η συνάρτηση του contract που καλείται δεν αλλάζει την κατάσταση του συστήματος στο blockchain, απλώς διαβάζει και εμφανίζει κάποιες πληροφορίες του. Για μία τέτοια λειτουργία δεν χρειαζόμαστε συναλλαγή, μονάχα κλήση. Η παραπάνω φόρμα εκτέλεσης ωστόσο, είναι ίδια με τις προηγούμενες που είδαμε καθώς διαγράφοντας την παραγγελία αλλάζει την κατάσταση των contracts. Με άλλα λόγια, πραγματοποιεί και αυτή συναλλαγή.

Συμπεραίνουμε ότι οι απλές κλήσεις δεν αποτελούν και μέρος των blocks, δηλαδή δεν αποτελούν μέλος της πληροφορίας που προστίθεται στο blockchain. Οι συναρτήσεις των contracts που απλά επιστρέφουν πληροφορίες χωρίς να επηρεάζουν το blockchain ορίζονται ως **view** (*βλ. drug_stores.sol*) Ως κατά συνέπεια δεν κοστίζουν τίποτα στον χρήστη, και είναι καλή πρακτική να ορίζουμε τέτοιες συναρτήσεις στα contracts όταν η λειτουργία τους δεν απαιτεί συναλλαγή. Αυτό μειώνει το συνολικό κόστος της εφαρμογής, μια παράμετρος άκρως σημαντική και μοναδική στις blockchain εφαρμογές. Παραδοσιακά applications έχουν μόνο υπολογιστικό κόστος, εδώ πέραν αυτού υπάρχει και χρηματικό κόστος μέσω του Ether των συναλλαγών.

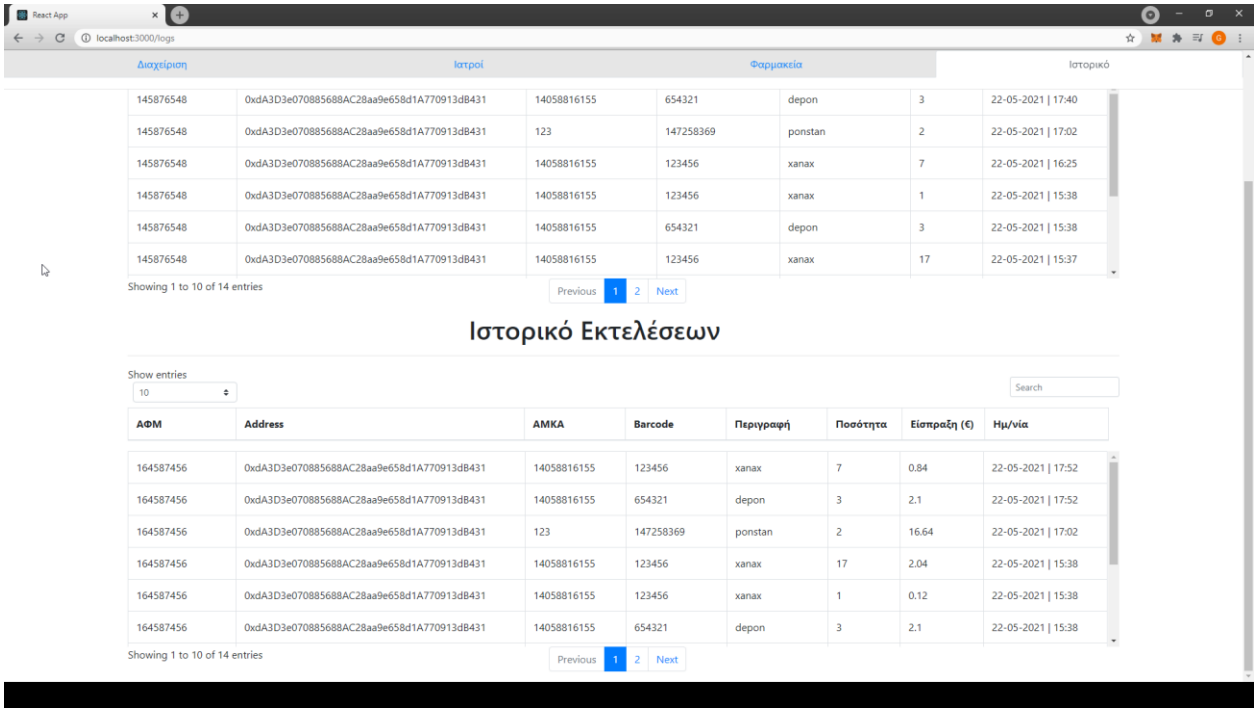
6.3.5 Ιστορικό Εφαρμογής

Σε ένα σύστημα όπως αυτό της παρούσας εργασίας, είναι απολύτως απαραίτητο να υπάρχει μία δυνατότητα καταγραφής ιστορικού. Αυτό χρειάζεται για 2 λόγους, feedback και ασφάλεια. Το πρώτο είναι ένας τρόπος να βλέπουμε τι έχει συμβεί στην εφαρμογή και να βγάζουμε πορίσματα για την παρούσα κατάστασή της. Το δεύτερο είναι ώστε να καταγράφουμε τις κινήσεις των χρηστών και να μπορούμε να τους κρατήσουμε υπεύθυνους για πιθανές κακόβουλες χρήσεις της εφαρμογής.



Εικόνα 11: Ιστορικό Συνταγών

Στο ιστορικό συνταγών φαίνονται όλες οι συνταγογραφήσεις φαρμάκων από ιατρούς. Διακρίνεται ο ΑΦΜ και το account address του ιατρού, ο ΑΜΚΑ του ασθενή στον οποίο έγινε η συνταγογράφηση, το barcode, η περιγραφή και η ποσότητα του φαρμάκου, καθώς και η ημερομηνία που πραγματοποιήθηκε. Όλη αυτή η πληροφορία αντλήθηκε μέσω των events που είδαμε προηγουμένως στον κώδικα των contracts.



Εικόνα 12: Ιστορικό Εκτελέσεων

Στο ιστορικό εκτελέσεων εμφανίζονται όλες οι συνταγές που έχουν εκτελεστεί από κάποιο φαρμακείο. Η δομή του πίνακα που τις εμφανίζει είναι πανομοιότυπη με αυτή του ιστορικού συνταγών. Οι μόνες διαφορές είναι ότι ο ΑΦΜ και το account address, αφορούν πλέον το φαρμακείο και όχι κάποιον ιατρό. Επίσης εμφανίζεται και το συνολικό ποσό που έλαβε το φαρμακείο σε ευρώ.

Τώρα, ας επιστρέψουμε στο κομμάτι της ασφάλειας παίρνοντας ένα παράδειγμα κακόβουλης χρήσης. Ένα φαρμακείο που εκτέλεσε την συνταγή ενός ασθενή χωρίς να του δώσει τα αντίστοιχα φάρμακα, θα έχει την εκτέλεση αυτή καταγεγραμμένη μαζί με τα στοιχεία του. Οπότε ο ασθενής, έχοντας ουσιαστικά χάσει τη συνταγή, θα ξεκινήσει τις απαραίτητες διαδικασίες ανάκτησής της. Μόλις βρεθεί στο ιστορικό εκτελέσεων, το φαρμακείο θα κληθεί να παρουσιάσει τα απαραίτητα αποδεικτικά στοιχεία τα οποία, αν δεν ταιριάζουν αυτά του ιστορικού, θα του επιφέρουν τις ανάλογες κυρώσεις.

7 Συμπεράσματα

Η παρούσα διπλωματική έχει ως σκοπό να παρουσιάσει την τεχνολογία blockchain ως μία εναλλακτική μέθοδο ανάπτυξης εφαρμογών, καθώς και να εξηγήσει πως μπορούν να επικοινωνήσουν μέσα από το ίδιο δίκτυο. Η ασφάλεια και η εμπιστοσύνη είναι δύο παράγοντες που όλες οι υπηρεσίες παγκοσμίως προσπαθούν να προσεγγίσουν στο μέγιστο δυνατό βαθμό, και οι διαδικτυακές εφαρμογές δεν αποτελούν εξαίρεση. Τα παραπάνω προβλήματα καλούνται να λύσουν τα blockchains. Με την αναλλοίωτη φύση τους και την αποκεντροποιημένη δομή τους, αποτελούν ένα επαναστατικό εργαλείο το οποίο ενδέχεται να αλλάξει τον τρόπο με τον οποίο πραγματοποιούμε ηλεκτρονικές συναλλαγές στο μέλλον, οικονομικές και μη.

Στο παράδειγμα που πραγματεύεται η συγκεκριμένη εργασία, παρατηρείται πως το πλεονέκτημα της ασφάλειας μπορεί να αξιοποιηθεί στο έπακρον από ένα σύστημα το οποίο, εκ φύσεως, το έχει μέγιστη ανάγκη. Ολόκληρη η συνταγογραφική εφαρμογή μίας χώρας, αποκεντροποιημένη. Φυσικά και υπάρχει πληθώρα βελτιώσεων που θα μπορούσαν να πραγματοποιηθούν. Για παράδειγμα, οι ιατροί θα μπορούσαν να έχουν ένα περιβάλλον αναζήτησης φαρμάκων ώστε να μην χρειάζεται να γνωρίζουν το barcode κατά την καταχώρηση της συνταγής. Επίσης, στα smart contracts όταν καταχωρείται ένας ασθενής, συμπεριλαμβάνεται το ονοματεπώνυμό του καθώς και ο αριθμός ταυτότητάς του. Αυτά τα στοιχεία δεν αξιοποιούνται σε κάποιο σημείο της εφαρμογής. Τέλος, θα μπορούσε να υπάρχει η δυνατότητα πολλαπλών admin, ώστε το κάθε υποκατάστημα να εισέρχεται από διαφορετικό λογαριασμό. Έτσι θα υπήρχε μεγαλύτερη διαφάνεια πίσω από τις κινήσεις των διαχειριστών, των οποίων η ορθή χρήση της εφαρμογής λαμβάνεται ως δεδομένη στην παρούσα εργασία.

Η αλήθεια ωστόσο, είναι πως η ραγδαία διάδοση της τεχνολογίας blockchain δεν σημαίνει πως όλες οι εφαρμογές, ανεξαρτήτως τύπου, συμφέρει να αναπτυχθούν πάνω στην τεχνολογία αυτή. Παρά τα σημαντικά πλεονεκτήματα που προσφέρει, εξακολουθεί να είναι ένας σχετικά υποανάπτυκτους τομέας με πολλές δυσκολίες, οι οποίες ναι μεν αξίζουν να αντιμετωπιστούν, αλλά μόνο όταν μπορούν να αξιοποιηθούν οι καρποί τους. Με σωστή κρίση όμως το μέλλον φαίνεται πολλά υποσχόμενο, τόσο στην ασφάλεια, όσο και στην αξιοπιστία.

8 Βιβλιογραφία

[1] Blockchain. (2021, May 14). Retrieved May 14, 2021, from https://en.wikipedia.org/wiki/Blockchain

[2] Ethereum. (2021, May 14). Retrieved May 14, 2021, from https://en.wikipedia.org/wiki/Ethereum

[3] Conway, L. (2020, November 17). Blockchain explained. Retrieved May 15, 2021, from https://www.investopedia.com/terms/b/blockchain.asp

[4] Cryptography hash functions. (n.d.). Retrieved May 15, 2021, from https://www.tutorialspoint.com/cryptography/cryptography_hash_functions.htm

[5] Sharma, T. (2019, August 07). Public vs. private blockchain : A comprehensive comparison. Retrieved May 16, 2021, from https://www.blockchain-council.org/blockchain/public-vs-private-blockchain-a-comprehensive-comparison/

[6] Geroni, D. (2021, April 05). Blockchain nodes: An in-depth guide. Retrieved May 16, 2021, from https://101blockchains.com/blockchain-nodes/

[7] Iredale, G. (2021, January 10). Public vs private blockchain: How do they differ? Retrieved May 16, 2021, from https://101blockchains.com/public-vs-private-blockchain/

[8] Cryptocurrency prices, charts and market capitalizations. (n.d.). Retrieved May 16, 2021, from https://coinmarketcap.com/

[9] Public and private keys. (2021, May 10). Retrieved May 16, 2021, from https://support.blockchain.com/hc/en-us/articles/360000951966-Public-and-private-keys

[10] Public-key cryptography. (2021, May 14). Retrieved May 16, 2021, from https://en.wikipedia.org/wiki/Public-key_cryptography

[11] Ethereum wallets. (n.d.). Retrieved May 16, 2021, from https://ethereum.org/en/wallets/

[12] Node.js. (2021, May 17). Retrieved May 18, 2021, from https://en.wikipedia.org/wiki/Node.js

[13] Attaran, M., & Gunasekaran, A. (2019). Applications of Blockchain Technology in Business: Challenges and Opportunities. Springer International Publishing.