



**ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ**

**ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ**

**ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ**

**Υλοποίηση Εφαρμογής Blockchain για Διαχείριση  
Χωροαναφερόμενων Χρονοσειρών Μετρητικών  
Δεδομένων με Χρήση Πρότυπων Υπηρεσιών Web**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**ΤΟΥ**

**ΠΙΤΣΟΛΗ Σ. ΓΕΩΡΓΙΟΥ**

**Επιβλέπων :** Βασίλειος Βεσκούκης  
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2022





ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

## Υλοποίηση Εφαρμογής Blockchain για Διαχείριση Χωροαναφερόμενων Χρονοσειρών Μετρητικών Δεδομένων με Χρήση Πρότυπων Υπηρεσιών Web

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ

ΠΙΤΣΟΛΗ Σ. ΓΕΩΡΓΙΟΥ

**Επιβλέπων :** Βασίλειος Βεσκούκης  
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την .....

(Υπογραφή)

.....

Βασίλειος Βεσκούκης  
Καθηγητής Ε.Μ.Π.

(Υπογραφή)

.....

Αριστείδης Παγουρτζής  
Καθηγητής Ε.Μ.Π

(Υπογραφή)

.....

Δημήτριος Φωτάκης  
Καθηγητής Ε.Μ.Π

Αθήνα, 22 Ιούλιος 2022





ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

(Υπογραφή)

.....

Πιτσόλης Γεώργιος

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

**Copyright © Πιτσόλης Γεώργιος, 2022**

Με την επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσεως υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν την χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν το συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις τους Εθνικού Μετσόβιου Πολυτεχνείου.

# Περίληψη

Αντικείμενο της παρούσας διπλωματικής εργασίας αποτελεί η ανάπτυξη ενός μοντέλου που εγγυάται την διασφάλιση χωροαναφερόμενων χρονοσειρών μετρητικών δεδομένων κάνοντας χρήση πρότυπων υπηρεσιών ιστού. Το μοντέλο αυτό θεμελιώθηκε με την υλοποίηση ενός δικτύου blockchain που συνεργάζεται με την πρότυπη υπηρεσία της πλατφόρμας του istSOS, με στόχο την διαχείριση και ασφάλεια των δεδομένων. Η ανάπτυξη του μοντέλου βασίστηκε στην αξιοποίηση πραγματικών δεδομένων που παραχωρήθηκαν από τον οργανισμό του Εθνικού Αστεροσκοπείου Αθηνών και έκανε χρήση της τεχνολογίας ενός ιδιωτικού Ethereum blockchain. Αρχικά, πραγματοποιήθηκε επεξεργασία των δεδομένων, ώστε να γίνουν συμβατά με το πρότυπο του istSOS και να τακτοποιηθούν και να διορθωθούν πιθανά σφάλματα. Η μεταφόρτωση στην πλατφόρμα του istSOS έγινε ταυτόχρονα με την αποστολή τους στο δίκτυο του blockchain. Παράλληλα, δημιουργήθηκε μία αποκεντρωμένη εφαρμογή ιστού φιλική προς τον χρήστη, ώστε να εξυπηρετείται η ομαλή συνύπαρξη των δύο τεχνολογιών. Κύριες δυνατότητες της αποκεντρωμένης εφαρμογής αποτέλεσαν η ταυτόχρονη αποστολή των δεδομένων στην πλατφόρμα του istSOS και στο blockchain, η πιστοποίηση των αρχείων του χρήστη με της αυθεντικές εγγραφές του δικτύου, η οπτικοποίηση της αλυσίδας του blockchain, η σχηματική απεικόνιση των μετρήσεων σε γράφημα με συγκεκριμένες προδιαγραφές και τέλος, η δυνατότητα λήψης των παρατηρήσεων αυτών σε μορφή JSON αρχείου. Από τα αποτελέσματα που προέκυψαν συνάγεται η κρίση πως ένα τέτοιο μοντέλο διασφαλίζει την προστασία των δεδομένων που περιλαμβάνει και μπορεί να αξιοποιηθεί μελλοντικά σε μεγαλύτερο εύρος εφαρμογών.

## Λέξεις Κλειδιά

Διασφάλιση Δεδομένων, Προστασία Δεδομένων, Γεωχωρικά Δεδομένα, Μετεωρολογικά Δεδομένα, Τεχνολογία Blockchain, Πλατφόρμα IstSOS, Αποκεντρωμένη Εφαρμογή Ιστού

# **Abstract**

The objective of this thesis is the development of a model that guarantees the assurance of spatially referenced time series of metric data using standard web services. This model foundations is the implementation of a blockchain network that cooperates with the standard service of istSOS platform, aiming optimal data management and maximum data security. This model was developed based on real data provided by the organization of the National Observatory of Athens using the technology of a private Ethereum blockchain. Initially, data were processed according to istSOS standard to ensure compatibility and identify and correct possible errors. Data uploading took place simultaneously at istSOS platform and the blockchain network. In parallel, a user-friendly decentralized web application was created to facilitate the smooth coexistence of both technologies. The decentralized application's main features were the simultaneously data uploading at istSOS platform and blockchain network, certification of the user's files with authenticated network records, visualization of the blockchain network, measurements display in a graph with specific standards and finally the ability to download these observations in JSON format file. From the results, it is concluded that such a model will ensure data protection and can be used in a wide range of applications in the future.

## **Keywords**

Data Security, Geospatial Data, Meteo Data, Blockchain, IstSOS, Decentralized Application





# Περιεχόμενα

Περίληψη .....	6
Abstract.....	7
Ευχαριστίες .....	Error! Bookmark not defined.
Περιεχόμενα.....	9
Υπότιτλοι Εικόνων .....	11
Εισαγωγή.....	13
Αρχική Ιδέα και Ανάγκη Ανάπτυξης Εφαρμογής .....	13
Οργάνωση Διπλωματικής .....	14
<b>1 Τεχνολογία Blockchain .....</b>	<b>17</b>
1.1 Δίκτυο Ομότιμων Κόμβων.....	17
1.1.1 Μορφές Δικτύων Ομότιμων Κόμβων .....	18
1.2 Κρυπτογραφία.....	18
1.2.1 Είδη Κρυπτογραφίας .....	19
1.2.2 Ασύμμετρη Κρυπτογραφία .....	20
1.3 Η Έννοια του Blockchain .....	21
1.4 Δομή και Λειτουργία .....	21
1.4.1 Δομή Block .....	21
1.4.2 Hash Puzzle – Proof of Work.....	23
1.4.2.1 Επίλυση Hash Puzzle.....	23
1.4.3 Smart Contract .....	24
1.5 Τύποι Blockchain .....	25
1.6 Εφαρμογές Blockchain .....	26
1.6.1 Αποκεντρωμένες Εφαρμογές Ιστού .....	26
<b>2 Υπηρεσίες Web για Γεωχωρικά Δεδομένα.....</b>	<b>29</b>
2.1 Τυποποίηση OGC Sensor Web Enablement .....	29
2.2 Πρότυπο OGC Sensor Observation Service .....	30
2.3 Μορφές Αναπαράστασης Δεδομένων για Υπηρεσίες Web .....	32
2.3.1 Μορφότυπος eXtensible Markup Language – XML.....	32
2.3.2 Μορφότυπος Comma Separated Values – CSV.....	33
2.3.3 Μορφότυπος JavaScript Object Notation – JSON .....	34
2.4 PostgreSQL – PostGIS .....	34
2.5 Apache HTTP Εξυπηρετητής.....	35
2.6 Αρχιτεκτονική Πλατφόρμας του IstSOS .....	35

2.7	Μετρητικά Δεδομένα με Χωρική Αναφορά.....	36
2.7.1	Κίνδυνοι Ακεραιότητας Δεδομένων .....	37
2.7.2	Μετεωρολογικά Δεδομένα.....	38
2.7.3	Φυσικές Καταστροφές, Έντονα Καιρικά Φαινόμενα και Επιπτώσεις αυτών.....	39
2.7.4	Ανάγκη Αξιοπιστίας Μετρητικών Δεδομένων .....	39
<b>3</b>	<b>Μεθοδολογία .....</b>	<b>41</b>
3.1	Συλλογή και Προεπεξεργασία Δεδομένων .....	43
3.1.1	Συλλογή Δεδομένων .....	43
3.1.2	Προεπεξεργασία Δεδομένων.....	44
3.1.3	Συμβατότητα με το Πρότυπο του IstSOS.....	47
3.1.4	Μεταφόρτωση Δεδομένων στο Πρότυπο του IstSOS .....	51
3.2	Υλοποίηση Blockchain.....	58
3.2.1	Κατασκευή Κύριας Δομής Blockchain .....	58
3.2.2	Υλοποίηση Smart Contract .....	66
3.2.2.1	Οργάνωση Περιεχομένου Smart Contract .....	67
3.2.3	Μεταφόρτωση Δεδομένων στην Αλυσίδα του Blockchain .....	72
3.3	Δημιουργία Αποκεντρωμένης Εφαρμογής .....	72
3.3.1	Δομή Αποκεντρωμένης Εφαρμογής .....	73
3.3.2	Διαμόρφωση και Λειτουργικότητα Αποκεντρωμένης Εφαρμογής.....	77
<b>4</b>	<b>Συμπεράσματα και Μελλοντική Έρευνα.....</b>	<b>89</b>
	<b>Βιβλιογραφία .....</b>	<b>91</b>

# Υπότιτλοι Εικόνων

Εικόνα 1-1 α. Αρχιτεκτονική πελάτη – εξυπηρετητή, β. Αρχιτεκτονική ομότιμων κόμβων.....	18
Εικόνα 1-2 Σχηματική απεικόνιση απλής μορφής κωδικοποίησης .....	19
Εικόνα 1-3 Σχηματική απεικόνιση συμμετρικής κρυπτογραφίας με ιδιωτικό κλειδί (μαύρο) .....	19
Εικόνα 1-4 Σχηματική απεικόνιση ασύμμετρης κρυπτογραφίας. Ιδιωτικό κλειδί (μαύρο) και δημόσιο κλειδί (λευκό).....	20
Εικόνα 1-5 Δομή block .....	22
Εικόνα 1-6 Σύνδεση block στην αλυσίδα του blockchain μέσω του hash value του προηγούμενου block .....	22
Εικόνα 1-7 Σχηματική απεικόνιση του hash puzzle.....	23
Εικόνα 1-8 Διάγραμμα Venn για blockchains χωρίς και με αδειοδότηση χρήση, αντίστοιχα.....	25
Εικόνα 2-1 Διάγραμμα ροής των πέντε βασικών κλειδιών λειτουργίας του SOS προτύπου .....	31
Εικόνα 2-2 Τυπικό SOS UML διάγραμμα για τις περιπτώσεις αλληλεπίδρασης με καταναλωτή (αριστερά) και με παραγωγό (δεξιά) .....	32
Εικόνα 2-3 Σχηματική απεικόνιση της ροής λειτουργίας της πλατφόρμας του istSOS .....	36
Εικόνα 3-1 Γενικό διάγραμμα ροής προτεινόμενης μεθοδολογίας.....	42
Εικόνα 3-2 Γενικό διάγραμμα ανάπτυξης προτεινόμενης μεθοδολογίας .....	43
Εικόνα 3-3 Τμήμα αρχείου μετεωρολογικών δεδομένων για τον σταθμό της Πεντέλης .....	44
Εικόνα 3-4 Τμήμα αρχείου μετεωρολογικών δεδομένων για τον σταθμό στο Χαροκόπειο Πανεπιστήμιο .....	45
Εικόνα 3-5 Λεξικό εύρεσης ορίων για κάθε διαφορετικό αισθητήρα .....	46
Εικόνα 3-6 Παράδειγμα συγχώνευσης μετρήσεων.....	47
Εικόνα 3-7 Πυξίδα με τα σημεία του ορίζοντα.....	49
Εικόνα 3-8 Τμήμα πίνακα με τους σταθμούς του Εθνικού Αστεροσκοπείου Αθηνών .....	50
Εικόνα 3-9 Σχηματική απεικόνιση δικτύου blockchain .....	58
Εικόνα 3-10 Σχηματική απεικόνιση σε μορφή δέντρου του περιεχομένου του φακέλου του blockchain .....	60
Εικόνα 3-11 Διάγραμμα ακολουθίας για την προσθήκη αρχείου στην πλατφόρμα του istSOS .....	75
Εικόνα 3-12 Διάγραμμα ακολουθίας για την αναπαράσταση και λήψη δεδομένων. 76	
Εικόνα 3-13 Διάγραμμα ακολουθίας για το ιστορικό του blockchain .....	77
Εικόνα 3-14 Στιγμιότυπο σύνδεσης του χρήστη στην αποκεντρωμένη εφαρμογή....	78
Εικόνα 3-15 Στιγμιότυπο επιλογής κατάλληλης γεωγραφικής περιφέρειας.....	78
Εικόνα 3-16 Στιγμιότυπο κύριας σελίδας της αποκεντρωμένης εφαρμογής .....	80
Εικόνα 3-17 Στιγμιότυπο προσθήκης αρχείων στο istSOS και στο blockchain .....	81
Εικόνα 3-18 Στιγμιότυπο μηνυμάτων συστήματος μετά την προσπάθεια εισαγωγής αρχείων .....	82

Εικόνα 3-19 Στιγμιότυπο πίνακα στοιχείων για την απεικόνιση δεδομένων σε διάγραμμα .....	83
Εικόνα 3-20 Στιγμιότυπο διαγράμματος δεδομένων θερμοκρασίας .....	84
Εικόνα 3-21 Στιγμιότυπο πίνακα ιστορικού για το blockchain .....	85
Εικόνα 3-22 Στιγμιότυπο πιστοποίησης αρχείων .....	86
Εικόνα 3-23 Στιγμιότυπο παραποίησης δεδομένων .....	86

# Εισαγωγή

## Αρχική Ιδέα και Ανάγκη Ανάπτυξης Εφαρμογής

Σήμερα, με την ραγδαία ανάπτυξη των μέσω τεχνολογίας και καινοτομιών παρουσιάζεται επιτακτική η ανάγκη διασφάλισης των δεδομένων που χρησιμοποιούνται σε κάθε είδος εφαρμογής. Όλο και περισσότεροι οργανισμοί, φορείς και υπηρεσίες εκσυγχρονίζονται και μεταφέρουν την έδρα τους στο διαδίκτυο εκτελώντας καθημερινά εκατομμύρια συναλλαγές και υπηρεσίες. Τα δεδομένα αποτελούν κομβικό κομμάτι της μετάβασης αυτής, καθώς οφείλουν να μένουν ακέραια στον άυλο δικτυακό τόπο αποφεύγοντας, ει δυνατόν, οποιαδήποτε εξωτερική παρεμβολή. Ως παράδειγμα από το θεματικό πεδίο της παρούσας εργασίας, τα μετεωρολογικά δεδομένα που καθορίζουν αποφάσεις σχετικές με την προστασία των πολιτών από ακραία καιρικά φαινόμενα και στα οποία στηρίζεται ένα ευρύ δίκτυο δράσης, προστασίας και καταστολής, χρήζουν ιδιαίτερης προσοχής. Ειδικότερα, λόγω της αξιοποίησης τους ως υλικό τεκμηρίωσης επιχειρησιακών αποφάσεων, αλλά και για τη νομική διερεύνηση τέτοιων αποφάσεων εκ των υστέρων, είναι πιθανό να αποτελέσουν στόχο κακόβουλου λογισμικού ή χρήστη.

Η προστασία αυτή των δεδομένων, για την οποία γίνεται λόγος όλο και πιο συχνά, καθιστά επιτακτική την ανάγκη χρήσης διαφόρων τεχνικών και εφαρμογών που θα εγγυώνται την διασφάλιση των δεδομένων – μετεωρολογικών, προσωπικών και γενικότερου σκοπού – σε βάθος χρόνου. Μια τέτοια τεχνολογία η οποία χρησιμοποιείται σε ευρεία κλίμακα κατά κύριο λόγο σε χρηματιστηριακού και συναλλαγματικού ενδιαφέροντος υπηρεσίες, είναι το blockchain. Κάνοντας το ντεμπούτο του στην λογική διασφάλισης συναλλαγών του κρυπτονομίσματος Bitcoin, το blockchain έχει αποτελέσει σημαντικό εργαλείο στην λογική προστασίας και διασφάλισης συναλλαγών σε κάθε πιθανό τύπο εφαρμογών και δεδομένων. Έκτοτε διάφορα οικοσυστήματα blockchain αποτελώντας λογισμικό ανοιχτού κώδικα, είναι διαθέσιμα στην κοινότητα με σκοπό την εξέλιξη των υπαρχουσών τεχνολογιών ασφαλείας σε όλα τα επίπεδα αλλά και την δημιουργία νέων.

Αντικείμενο της παρούσας διπλωματικής εργασίας αποτελεί η επίδειξη της ανάπτυξης ενός σύγχρονου blockchain για την αποθήκευση δεδομένων μετεωρολογικού ενδιαφέροντος, με σκοπό τη διασφάλιση της ακεραιότητας τους και την ταυτόχρονη αδυναμία τροποποίησης ή διαγραφής τους. Τα δεδομένα μετεωρολογικών μετρήσεων που θα αποθηκεύονται στο blockchain, θα γίνονται διαθέσιμα μέσω της πρότυπης υπηρεσίας web SOS (Sensor Observations Standard), του οργανισμού OGC (Open Geospatial Consortium) και συγκεκριμένα, μιας υλοποίησης ανοιχτού κώδικα αυτής, της διαδικτυακής πλατφόρμας istSOS, που αποτελεί περιβάλλον εξειδικευμένου περιεχομένου για διαχείριση, επεξεργασία και αξιοποίηση κάθε είδους γεωαναφερόμενων μετρητικών δεδομένων. Η αλληλεπίδραση και η διαχείριση τόσο του blockchain όσο και του istSOS θα

περατώνονται από μία αποκεντρωμένη εφαρμογή ιστού (Decentralized Application – DApp) που θα περιέχει όλα τα απαραίτητα στοιχεία μιας διεπαφής για το σκοπό αυτό.

## Οργάνωση Διπλωματικής

Η εργασία δομείται σε δύο βασικά κεντρικά μέρη:

Το **Μέρος Α: Θεωρητικό Υπόβαθρο** που αναλύει όλα τα κύρια σημεία που απαιτούνται προκειμένου να γίνει πλήρως κατανοητό το περιεχόμενο του κάθε δομικού σημείου της εργασίας. Ειδικότερα το μέρος αυτό διαρθρώνεται στα ακόλουθα δύο κεφάλαια:

- Στο **Κεφάλαιο 1** περιγράφονται συνοπτικά γενικότεροι όροι που πρέπει να είναι γνωστοί για την πορεία της εργασίας, όπως το δίκτυο ομότιμων κόμβων και η τεχνική της κρυπτογραφίας στα σύγχρονα υπολογιστικά συστήματα. Ακόμη, γίνεται περιγραφή της τεχνολογίας του blockchain. Αναλύεται η δομή και η λειτουργία του διευκρινίζοντας τα βασικά στοιχεία που το αποτελούν και γίνεται λόγος για τις εφαρμογές που το αξιοποιούν.
- Στο **Κεφάλαιο 2** γίνεται η ανάλυση της πλατφόρμας istSOS, που θα χρησιμοποιηθεί για τη διαχείριση των μετεωρολογικών δεδομένων. Περιγράφονται τα κριτήρια και οι απαιτήσεις που το διακρίνουν τόσο για το φιλτράρισμα των δεδομένων που του παρέχονται όσο και για την γόνιμη αξιοποίηση και παροχή τους στους χρήστες, κάνοντας ακόμη λόγο και για τα πρότυπα και τις τεχνολογίες που χρησιμοποιεί. Επιπλέον, γίνεται μια αναφορά σε μετρητικά μετεωρολογικά δεδομένα ως στοιχείο προβληματισμού αναφορικά με τους λόγους που τα καθιστούν δεδομένα υψηλού ενδιαφέροντος που χρήζουν προστασίας και προσοχής.

Το **Μέρος Β: Ανάπτυξη Εφαρμογής** που αποτελεί περιγραφική απεικόνιση της πορείας που ακολουθήθηκε προκειμένου να έρθει σε πέρας η υλοποίηση της εφαρμογής που πραγματεύεται η εργασία. Εκτείνεται και αυτό σε δύο κύρια κεφάλαια ως εξής:

- Στο **Κεφάλαιο 3** γίνεται εκτενής ανάλυση της μεθοδολογίας που ακολουθήθηκε μέσω διαφόρων σταδίων. Παρουσιάζεται η επεξεργασία των δεδομένων που παραχωρήθηκαν από την ομάδα meteo του Εθνικού Αστεροσκοπείου Αθηνών, η διαδικασία μεταφόρτωσής τους στην πλατφόρμα του istSOS και στο δίκτυο του blockchain, καθώς και η ανάπτυξη της αποκεντρωμένης εφαρμογής διαδικτύου που χρησιμοποιείται για την περάτωση των λειτουργιών από τον χρήστη χωρίς να απαιτείται γνώση προγραμματισμού.
- Στο **Κεφάλαιο 4** γίνεται αξιολόγηση των αποτελεσμάτων τόσο στο κομμάτι της ασφάλειας των δεδομένων όσο και σε εκείνο της ομαλής διασύνδεσης της πλατφόρμας του istSOS με το δίκτυο του blockchain. Τέλος, γίνεται

συζήτηση για τα περιθώρια μελλοντικής έρευνας και ανάπτυξης της παρούσας εφαρμογής, καθώς και σε ποια άλλα πεδία θα μπορούσε να εφαρμοστεί και να λειτουργήσει ως επιτεύξιμη εφαρμογή με ανάλογα αποτελέσματα.





# Μέρος Α

## Θεωρητικό Υπόβαθρο

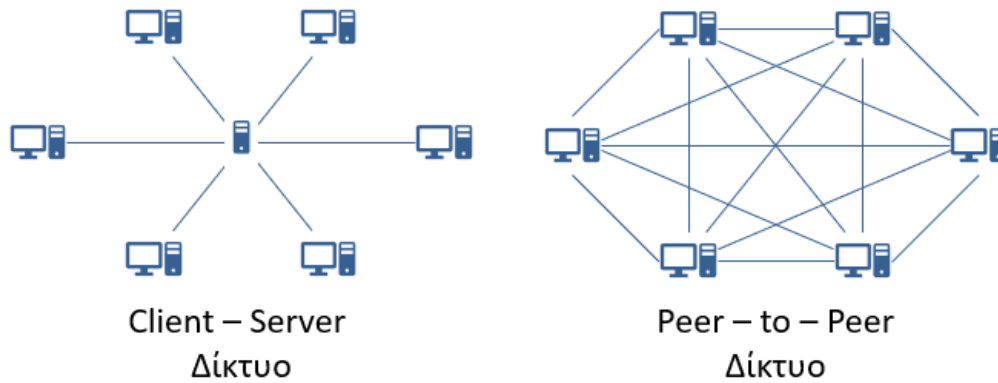
---

### 1 Τεχνολογία Blockchain

Η τεχνολογία του blockchain έθεσε κατά κύριο λόγο τις βάσεις της πάνω σε δύο βασικές τεχνικές λειτουργίες. Η αρχιτεκτονική του δικτύου ομότιμων κόμβων που χρησιμοποιείται σε διαδικτυακές εφαρμογές και η επιστήμη της κρυπτογραφίας είναι αυτές που αποτέλεσαν την αφετηρία ανάπτυξης μιας προγραμματιστικής τεχνικής που αποτελεί την μέχρι τώρα ασφαλέστερη λογική διαχείρισης δεδομένων. Τα παραπάνω περιγράφονται επαρκώς στα υποκεφάλαια που ακολουθούν και αποτελούν απαραίτητο πρόδρομο κατανόησης της συλλογιστικής πορείας της εργασίας αυτής.

#### 1.1 Δίκτυο Ομότιμων Κόμβων

Οι δικτυακές εφαρμογές σήμερα, ακολουθούν κατά κύριο λόγο δύο διαφορετικές αρχιτεκτονικές. Την αρχιτεκτονική του πελάτη – εξυπηρετητή (client – server) και την αρχιτεκτονική των ομότιμων κόμβων ( peer-to-peer / P2P ). Στην πρώτη περίπτωση, ο εξυπηρετητής, είναι ένας κεντρικός - πάντα ενεργός - υπολογιστής που διευθετεί ζητήματα και υπηρεσίες των υπόλοιπων χρηστών που βρίσκονται συνδεδεμένοι στο δίκτυο και έχει καθοριστική σημασία για την ορθή λειτουργία του δικτύου. Στην περίπτωση των δικτύων ομότιμων κόμβων, δεν υπάρχει διάκριση ανάμεσα σε εξυπηρετητή και πελάτη. Όλοι οι υπολογιστές που συμμετέχουν μοιράζονται τους πόρους ισοδύναμα, καθώς το δίκτυο χρησιμοποιεί την επεξεργαστική ισχύ, τον αποθηκευτικό χώρο και το εύρος ζώνης (bandwidth) τους, προσφέροντας τους με τον τρόπο αυτόν ίσα δικαιώματα[1], [2].



Εικόνα 1-1 α. Αρχιτεκτονική πελάτη – εξυπηρετητή, β. Αρχιτεκτονική ομότιμων κόμβων

### 1.1.1 Μορφές Δικτύων Ομότιμων Κόμβων

Τα δίκτυα ομότιμων κόμβων διακρίνονται σε τρεις κατηγορίες:

- Συγκεντρωτικά δίκτυα ομότιμων κόμβων  
Υπάρχει ένας κεντρικός υπολογιστής Index εξυπηρετητής στον οποίο βρίσκονται αποθηκευμένα όλα τα δεδομένα που χρειάζονται οι χρήστες του δικτύου. Σε αυτόν γίνεται η αναζήτηση των απαραίτητων δεδομένων από τους χρήστες που, μόλις εντοπιστούν, πραγματοποιείται σύνδεση για την μεταφορά τους.
- Αποκεντρωμένα δίκτυα ομότιμων κόμβων  
Κάθε υπολογιστής του δικτύου είναι ταυτόχρονα πελάτης και εξυπηρετητής. Η είσοδος ενός καινούργιου υπολογιστή στο δίκτυο γίνεται γνωστή σε ένα μικρό τμήμα ήδη συνδεδεμένων υπολογιστών οι οποίοι με την σειρά τους προωθούν το μήνυμα σύνδεσης σε ένα μεγαλύτερο τμήμα μέχρι εν τέλη, να καλυφθεί όλο το δίκτυο.
- Δίκτυα ομότιμων κόμβων τρίτης γενιάς  
Διαθέτουν χαρακτηριστικά ανωνυμίας, καθώς σκοπός τους είναι κανένας χρήστης να μην μπορέσει να αποκτήσει κάποια μορφή ελέγχου πάνω στα αρχεία του δικτύου. Για τον λόγο αυτό δίνουν έμφαση στην κωδικοποίηση και στον συνεχή διαμοιρασμό των αρχείων αυτών, υιοθετώντας στοιχεία αποκεντρωμένου δικτύου [1].

## 1.2 Κρυπτογραφία

Η κρυπτογραφία σαν έννοια και ανάγκη αναπτύχθηκε, για να μπορεί να προστατεύσει οποιασδήποτε μορφής δεδομένα, την πρόσβαση στα οποία προσπαθούν να επιτύχουν μη εξουσιοδοτημένοι χρήστες. Η βασική ιδέα στηριζόμενη

στις φυσικές κλειδαριές που κλειδώνουν με μοναδικά κλειδιά οδήγησε στην εισαγωγή του όρου κλειδί (key) στον τομέα της κρυπτογραφίας [2], [3].

Βασικός πυλώνας στην κρυπτογραφία είναι κωδικοποίηση και η αποκωδικοποίηση των δεδομένων. Σχηματικά, τα δεδομένα που πρέπει να προστατευτούν κωδικοποιούνται και παράγουν ένα κωδικοποιημένο κείμενο (Cypher text). Το τελευταίο αποτελείται από μία ακολουθία χαρακτήρων χωρίς καμία συνάφεια μεταξύ τους και το οποίο αποκωδικοποιείται από τον δέκτη, παράγοντας εκ νέου τα πρωταρχικά δεδομένα χωρίς καμία αλλοίωση.



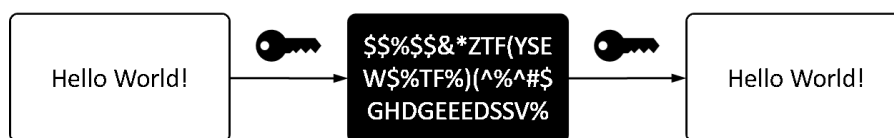
Εικόνα 1-2 Σχηματική απεικόνιση απλής μορφής κωδικοποίησης

### 1.2.1 Είδη Κρυπτογραφίας

Τα κλειδιά που μπορούν να χρησιμοποιηθούν σε ένα κωδικοποιημένο ή μη μήνυμα μπορεί να είναι είτε δημόσια (public), δηλαδή μπορεί να τα κατέχει ο οποιοδήποτε είτε ιδιωτικά (private) που ανήκουν μονάχα σε έναν χρήστη ή σε μία ομάδα χρηστών. Επομένως, με βάση την ταυτότητα των κλειδιών που χρησιμοποιούνται, η κρυπτογραφία μπορεί να διακριθεί σε δύο μεγάλες κατηγορίες:

- **Συμμετρική Κρυπτογραφία (Symmetric Cryptography)**

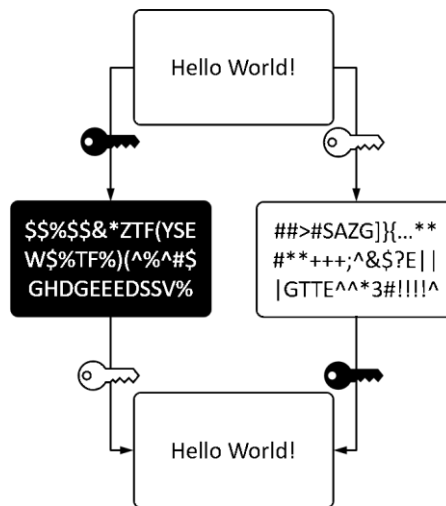
Η λειτουργία της βασίζεται μόνο σε ένα ιδιωτικό κλειδί που είναι γνωστό τόσο στον πομπό όσο και στον δέκτη. Το αρχικό μήνυμα μετατρέπεται στο κωδικοποιημένο κείμενο μέσω του κλειδιού αυτού το οποίο με την σειρά του ανάγεται στο αρχικό μήνυμα με το ίδιο πάλι κλειδί [2].



Εικόνα 1-3 Σχηματική απεικόνιση συμμετρικής κρυπτογραφίας με ιδιωτικό κλειδί (μαύρο)

- **Ασύμμετρη Κρυπτογραφία (Asymmetric Cryptography)**

Η ειδοποιός διαφορά με την συμμετρική κρυπτογραφία είναι ότι γίνεται χρήση διαφορετικού είδους κλειδιού για τις δύο φάσεις της κρυπτογράφησης ενός μηνύματος, ενός ιδιωτικού και ενός δημόσιου. Το κρυπτογραφημένο κείμενο δημιουργείται με το ένα από τα δύο κλειδιά και μπορεί να αποκωδικοποιηθεί μόνο με το άλλο.



Εικόνα 1-4 Σχηματική απεικόνιση ασύμμετρης κρυπτογραφίας. Ιδιωτικό κλειδί (μαύρο) και δημόσιο κλειδί (λευκό)

### 1.2.2 Ασύμμετρη Κρυπτογραφία

Στην ασύμμετρη κρυπτογραφία κάνουν την εμφάνισή τους και οι δύο κατηγορίες κλειδιών στην τεχνική της κωδικοποίησης, ένα ιδιωτικό και ένα δημόσιο. Όπως φαίνεται και από την παραπάνω εικόνα, οι διαδρομές που μπορούν να ακολουθηθούν είναι δύο [2], [3]:

- Δημόσια σε Ιδιωτική**  
 Οποιοσδήποτε μπορεί να δημιουργήσει το κρυπτογραφημένο κείμενο, αφού το κλειδί που απαιτείται είναι το δημόσιο. Από την μία πλευρά είναι η ύπαρξή του, που την γνωρίζουν όλοι, οπότε έχουν την ικανότητα να προχωρήσουν στην διαδικασία της κωδικοποίησης. Από την άλλη, μόνο ο δέκτης είναι αυτός που ως κάτοχος του ιδιωτικού κλειδιού είναι εκείνος που μπορεί να αποκωδικοποιήσει το κρυπτογραφημένο κείμενο και να διαβάσει το μήνυμα.
- Ιδιωτική σε Δημόσια**  
 Ακριβώς το αντίστροφο συμβαίνει στην παρούσα περίπτωση, ιδιωτική σε δημόσια. Μόνο ο πομπός, ο χρήστης δηλαδή εκείνος που είναι υπεύθυνος για την ασφάλεια, την προστασία και την κατοχή του ιδιωτικού κλειδιού, μπορεί να κωδικοποιήσει το μήνυμα και να παράγει επιτυχώς το κρυπτογραφημένο κείμενο. Όλοι οι χρήστες του δικτύου που κατέχουν το δημόσιο κλειδί μπορούν να διαβάσουν το μήνυμα αποκωδικοποιώντας πρώτα το κρυπτογραφημένο κείμενο.

### 1.3 Η Έννοια του Blockchain

Το blockchain πρόκειται για ένα δίκτυο ομότιμων κόμβων, δηλαδή έναν εκτενή κατάλογο καταχωρήσεων, που αφορούν συναλλαγές, σε ένα δημόσιο λογιστικό «βιβλίο» (ledger). Οι καταχωρήσεις αυτές ομαδοποιούνται σε μικρότερες δομές τα blocks με κάθε ένα να αντιστοιχίζεται σε μία ομάδα από καινούργιες συναλλαγές. Κάθε νέο block συνδέεται με το προηγούμενο σχηματίζοντας μια αλυσίδα, το blockchain. Κάθε block αποτελείται από τα δεδομένα συναλλαγής που πρέπει να διασφαλιστούν, ένα μοναδικό κρυπτογραφημένο ψηφιακό αποτύπωμα που προσδιορίζει το συγκεκριμένο block με το ακριβές περιεχόμενο του και το αντίστοιχο αποτύπωμα του προηγούμενου block [1].

Ένα από τα βασικότερα χαρακτηριστικά της τεχνολογίας blockchain είναι η αμεταβλητότητά του (Immutability) [2], ότι επιτρέπεται δηλαδή μόνο η προσθήκη νέων συναλλαγών αποτρέποντας αντίστοιχα οποιαδήποτε τροποποίηση ή διαγραφή ήδη αποθηκευμένων πληροφοριών. Ουσιαστικά πρόκειται για μία «αλυσίδα» συναλλαγών μεταξύ δύο ή περισσότερων χρηστών / κόμβων που βρίσκονται μέσα στο ίδιο δίκτυο. Για να καταγραφεί μία νέα συναλλαγή μεταξύ των μερών, πρέπει να επαληθευτεί η γνησιότητα και η πιστότητά της, από την πλειοψηφία των μερών του δικτύου. Μετά την καταγραφή τα δεδομένα του block δεν μπορούν να τροποποιηθούν χωρίς αλλοίωση όλων των προηγούμενων blocks, γεγονός που καθιστά το blockchain ασφαλή και ιδιαίτερα αποτελεσματικό στην διασφάλιση της ακεραιότητας των δεδομένων σε μεγάλο εύρος εφαρμογών.

### 1.4 Δομή και Λειτουργία

Όπως πιθανόν έχει γίνει κατανοητό η ιδέα του blockchain έχει συγκεντρώσει το ενδιαφέρον, από την σύλληψή της στο white paper του Bitcoin το 2008 [1] τόσο στον τομέα των κρυπτονομισμάτων και της χρηματοοικονομικής όσο και σε ένα ευρύτερο πλαίσιο συστημάτων γενικότερου σκοπού που στοχεύουν στην διασφάλιση και προστασία των δεδομένων τους.

Πρόκειται για μια δομή δεδομένων που καταγράφει συναλλαγές (transactions). Οι συναλλαγές αυτές αποτελούν την μικρότερη δυνατή μονάδα που ανταλλάσσεται μέσα σε ένα blockchain. Κάθε μία προσδιορίζει τον χρόνο που συνέβη (timestamp), τα δεδομένα που πρέπει να αποθηκευτούν στην αλυσίδα, την διεύθυνση του αποστολέα και του παραλήπτη, μια σειρά από εντολές που πρέπει να εκτελεστούν και διάφορες άλλες πληροφορίες για την ορθή λειτουργία του blockchain. Τα δεδομένα αυτά των συναλλαγών αποθηκεύονται στην βασικότερη δομή του blockchain το block.

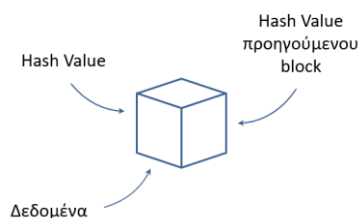
#### 1.4.1 Δομή Block

Τα blocks είναι οι «κρίκοι» της αλυσίδας του blockchain που δεσμεύουν επ' αόριστων την πληροφορία που στόχος είναι να παραμείνει αναλλοίωτη.

Κάθε block περιλαμβάνει τα ακόλουθα:

- το Hash Value

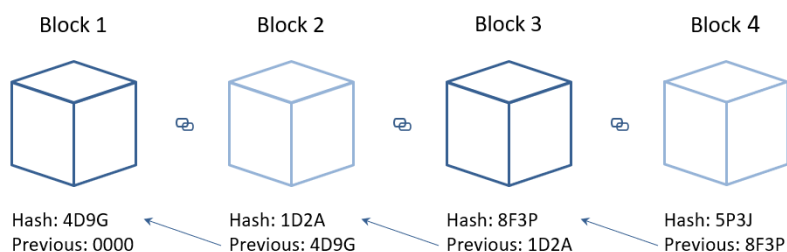
Αποτελεί το μοναδικό ψηφιακό αποτύπωμα των δεδομένων του block. Όλα τα δεδομένα που στέλνονται με κάποια συναλλαγή στο blockchain κωδικοποιούνται με διάφορους γνωστούς - μονής κατεύθυνσης - αλγόριθμους (hash συναρτήσεις: MD5, SH1, SHA256, SHA512 κλπ.) και αποθηκεύονται στο block με την μορφή ενός δέντρου Merkle [2]. Η κωδικοποίηση αυτή δίνει την δυνατότητα στους χρήστες να μπορούν ανά πάσα στιγμή να ελέγξουν αν τα δεδομένα του block είναι πιστοποιημένα ή έχουν υποστεί κάποια αλλοίωση υπολογίζοντας εκ νέου το hash value τους [4].



Εικόνα 1-5 Δομή block

- το Hash Value του προηγούμενου block

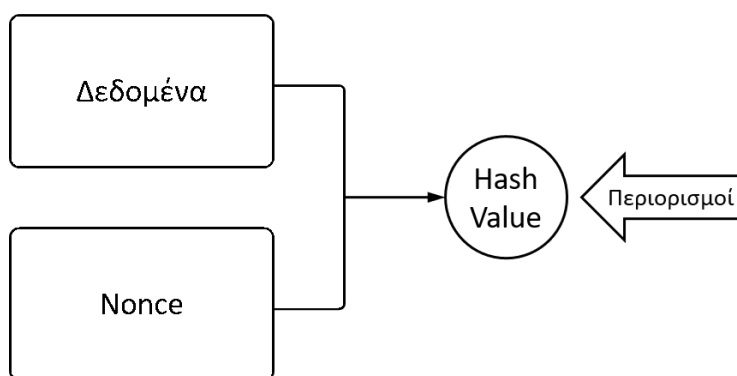
Με τον τρόπο αυτό, αν τροποποιηθεί το περιεχόμενο από ένα block θα αλλάξει και το hash value που το προσδιορίζει. Επομένως το επόμενο block θα δείχνει στο παλιό hash value του προηγούμενου block που πλέον δεν υπάρχει. Η αλυσίδα έχει σπάσει και τα δεδομένα γίνεται γνωστό ότι έχουν παραβιαστεί. Με την χρήση του hash value του προηγούμενου ακριβώς block επιτυγχάνεται μία ενιαία σύνδεση του πιο πρόσφατα προστιθέμενου block με το πρωταρχικό (initial) block του blockchain [2], [5].



Εικόνα 1-6 Σύνδεση block στην αλυσίδα του blockchain μέσω του hash value του προηγούμενου block

- και τα δεδομένα που πρέπει να αποθηκευτούν.

Η χρήση του hash value είναι εκείνη που κάνει δυνατή την πιστοποίηση του περιεχομένου του block από κάθε κόμβο (node) του blockchain. Επιπρόσθετα, στα δεδομένα, που αποστέλλονται με μία συναλλαγή και τα οποία οφείλουν να παραμένουν αμετάβλητα επ' αόριστων, προσδιορίζονται από έναν μοναδικό ακέραιο αριθμό που προσεγγίζει την δυσκολία εύρεσης του συγκεκριμένου hash value από κάποιον τρίτο, το λεγόμενο nonce [5]. Με την βοήθεια του nonce κατασκευάζεται η αλγοριθμική ακολουθία που πρέπει να ακολουθηθεί από οποιονδήποτε μέσα στο σύστημα που θέλει να προσθέσει δεδομένα, γνωστή ως Hash puzzle ή Proof-of-Work.



Εικόνα 1-7 Σχηματική απεικόνιση του hash puzzle

#### 1.4.2 Hash Puzzle – Proof of Work

Οι κόμβοι είναι συσκευές – συνήθως ηλεκτρονικοί υπολογιστές – που απαρτίζουν το δίκτυο του blockchain. Πρόκειται για ένα δίκτυο ομότιμων κόμβων οπότε όλοι οι κόμβοι είναι ισοδύναμοι μεταξύ τους, έχουν τις ίδιες ιδιότητες και κάθε ένας από αυτούς κατέχει ένα πιστό αντίγραφο του ledger, του «βιβλίου» δηλαδή των συναλλαγών που πραγματοποιούνται στο blockchain [6]. Σκοπός του συστήματος blockchain είναι οι κόμβοι να διατηρούν πιστά αντίγραφα του ledger για τις συναλλαγές που έχουν περατωθεί και οι οποίες παραμένουν αναλλοίωτες για όλη την διάρκεια ζωής του blockchain. Οι κόμβοι της αλυσίδας του blockchain είναι εκείνοι που κρίνουν αν ένας νέος κόμβος που επιθυμεί να γίνει μέλος του συστήματος είναι αξιόπιστος ή όχι καθώς επίσης και αν τα καινούργια δεδομένα που θέλουν να προστεθούν στην αλυσίδα του blockchain είναι πιστοποιημένα και αληθή. Για τον λόγο αυτό ακολουθείται μια διαδικασία που καλείται proof-of-work ( ή hash puzzle ) και είναι εκείνη στην οποία βασίζεται η λογική λειτουργίας του blockchain [1], [2].

##### 1.4.2.1 Επίλυση Hash Puzzle

Η λογική που κρύβεται πίσω από την χρησιμότητα του proof-of-work είναι η δημιουργία ενός «παζλ» αλγορίθμων που πρέπει να επιλυθούν, προκειμένου να μπορέσει να προστεθεί οτιδήποτε στο blockchain. Όταν ένας χρήστης επιθυμεί να προσθέσει μια καινούργια πληροφορία μέσα στο blockchain, θα πρέπει να κάνει μία καινούργια συναλλαγή. Η συναλλαγή αυτή χρειάζεται πρώτα να λάβει το

πιστοποιητικό γνησιότητάς της που διασφαλίζει την εγκυρότητα των δεδομένων της και έπειτα οι υπόλοιποι χρήστες του δικτύου είναι εκείνοι που ελέγχουν για την ορθότητα των δεδομένων μέσω της επίλυσης ενός πολύπλοκου μαθηματικού προβλήματος, του hash puzzle. Εφόσον η πλειοψηφία των χρηστών που καταφέρουν να επιλύσουν το πρόβλημα αυτό αναγνωρίσουν τον χρήστη και την συναλλαγή του ως έγκυρους, η καινούργια συναλλαγή λαμβάνοντας της πιστοποίησή της γίνεται μέρος της αλυσίδας του blockchain. Η διαδικασία επίλυσης του προβλήματος που απαιτείται κατά την λειτουργία του proof-of-work καλείται "mining" και οι χρήστες αντίστοιχα "miners".

Για να έχουν κίνητρο οι χρήστες να συμμετέχουν στην διαδικασία ελέγχου των καινούργιων δεδομένων, επιβραβεύονται συνήθως για την επιτυχία τους με κάποιο κρυπτονόμισμα. Η επίλυση, όμως ενός τόσο δύσκολου αλγοριθμικού προβλήματος μπορεί να αποδειχθεί αρκετά χρονοβόρα και να κοστίζει σε υπολογιστική δύναμη με απόρροια ένας κακόβουλος χρήστης από μόνος του να μην μπορεί να επιφέρει αλλαγές που θα περάσουν απαρατήρητες στο σύνολο του δικτύου. Σε αυτό συμβάλει το γεγονός πως τα προβλήματα αυτά λειτουργούν με τρόπο τέτοιο, ώστε να έχουν την δυνατότητα να επιλυθούν μόνο μέσω της τεχνικής της δοκιμής και του λάθους.

Η τεχνική αυτή βασίζεται στην υπόθεση από τον χρήστη που προσπαθεί να κάνει την πιστοποίηση του nonce που έχει χρησιμοποιηθεί για την συναλλαγή. Αρχικά, θεωρείται ένα τυχαίο nonce και υπολογίζεται το hash value των δεδομένων σε συνδυασμό με το τυχαίο nonce με την κατάλληλη συνάρτηση hash. Αν το αποτέλεσμα της συνένωσης των δεδομένων με το nonce δίνει σαν αποτέλεσμα ένα hash value που ικανοποιεί τις απαιτήσεις του συστήματος, τότε έχει βρεθεί η λύση του hash puzzle και ο χρήστης ανταμείβεται, αν έχει καταφέρει να την υπολογίσει πρώτος.

Με τον τρόπο αυτό επιτυγχάνεται μία από τις βασικότερες ιδιότητες ενός συστήματος blockchain, η συναίνεση (Consensus). Σύμφωνα με αυτήν οι χρήστες του blockchain μετά την επιτυχή υλοποίηση του proof-of-work αλγορίθμου συμφωνούν με βάση την πλειοψηφία αν μια συναλλαγή είναι έγκυρη και καταλήγουν στην τελική κατάσταση που θα πρέπει να έχει το ledger. Η νέα αυτή έκδοσή του διαμοιράζεται σε όλους τους κόμβους του δικτύου, με σκοπό να έχουν, ο καθένας χωριστά, το πιο πρόσφατο αντίγραφο του ledger. Σε γενικότερα πλαίσια η πρόσβαση στο ledger είναι ελεύθερη για όλους, ώστε να μπορούν να γνωρίζουν το ιστορικό των συναλλαγών, καθώς επίσης και την αλληλουχία των κινήσεων κατά την διάρκεια ζωής του blockchain επιτυγχάνοντας την διαφάνεια (Transparency) των συναλλαγών σε όλα τα επίπεδα.

### 1.4.3 Smart Contract

Τα έξυπνα συμβόλαια (smart contracts) είναι προγράμματα κώδικα που χρησιμοποιούνται, για να υλοποιήσουν όλες τις λειτουργίες ενός blockchain. Έκαναν την εμφάνισή τους αρκετά πριν την σύλληψη του blockchain από τον Αμερικάνο επιστήμονα στις τεχνολογίες των υπολογιστών, Nick Szabo το 1994 [7]. Πρόκειται για προγράμματα που παραμένουν αποθηκευμένα και αναλλοίωτα μέσα στο δίκτυο και

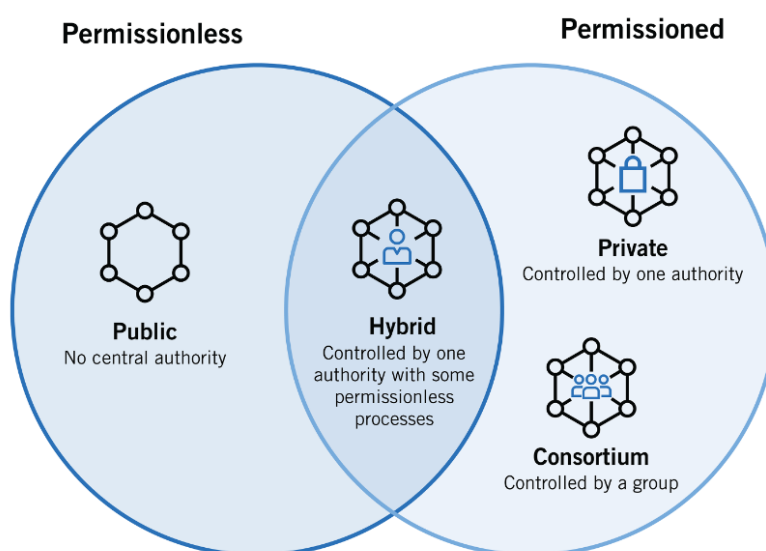


εκτελούνται όταν κάποιος χρήστης του δικτύου τα καλέσει. Περιέχουν βοηθητικές συναρτήσεις για τις λειτουργίες που πρέπει να εκτελούνται αναπόσπαστα από τους κόμβους του blockchain. Με την παρουσίαση του Bitcoin το 2009 υλοποιήθηκε και το πρώτο πρωτόκολλο έξυπνο συμβόλαιο που συμπεριλάμβανε όλες τις συνθήκες που έπρεπε να υλοποιούνται, προκειμένου να εκτελεστούν οι μεταφορές κρυπτονομισμάτων, Bitcoins, μεταξύ των χρηστών του δικτύου [7], [8].

## 1.5 Τύποι Blockchain

Κάθε blockchain μπορεί να χαρακτηριστεί είτε χωρίς αδειοδότηση χρήστη (permissionless) είτε με αδειοδότηση χρήστη (permissioned) είτε και τα δύο [9]. Στην πρώτη περίπτωση, ανήκουν τα συστήματα που επιτρέπουν σε οποιονδήποτε χρήστη να μπει στο δίκτυο χωρίς να του στερούν παράλληλα κάποιο από τα δικαιώματά του μέσα σε αυτό. Ένα τέτοιο δίκτυο μπορεί να χαρακτηριστεί ως σχετικά ανώνυμο αφού οι χρήστες δεν γνωρίζονται μεταξύ τους και μοναδικό τους κίνητρο, για να προσφέρουν την υπολογιστική τους δύναμη στο blockchain, είναι η χρηματική – μέσω κρυπτονομισμάτων – επιβράβευσή τους. Τα χωρίς αδειοδότηση χρήστη blockchain τείνουν να είναι πιο ασφαλή, καθώς διαθέτουν έναν εκτενή έλεγχο από πληθώρα κόμβων, προκειμένου μία νέα συναλλαγή να μπει στο blockchain. Αυτό οφείλεται στην έλλειψη εμπιστοσύνης μεταξύ των χρηστών και οδηγεί σε χρονοβόρες διεργασίες για την πιστότητα των συναλλαγών.

Από την άλλη πλευρά, σε ένα blockchain με αδειοδότηση η πρόσβαση νέων χρηστών είναι περιορισμένη. Ένα τέτοιο σύστημα χαρακτηρίζεται ως κλειστό ή ιδιωτικό, καθώς νέοι χρήστες μπορούν να εισέλθουν μόνο, αν προσκληθούν από κάποιον υπάρχοντα χρήστη του δικτύου, γεγονός που καθιστά την ταυτότητά τους γνωστή. Λόγω του πιο περιορισμένου αριθμού κόμβων στο δίκτυο σε σχέση με ένα χωρίς αδειοδότηση χρήστη blockchain, το με αδειοδότηση χρήστη χαρακτηρίζεται ως ταχύτερο, αλλά ελαφρώς πιο ελλιπές ως προς την ασφάλεια [9].



Εικόνα 1-8 Διάγραμμα Venn για blockchains χωρίς και με αδειοδότηση χρήστη, αντίστοιχα

Όπως φαίνεται και στην παραπάνω εικόνα, ένα δίκτυο blockchain χωρίς αδειοδότηση μπορεί να είναι δημόσιο (public), δηλαδή όλοι οι χρήστες είναι αποκεντρωμένοι και ξένοι μεταξύ τους, ενώ ένα δίκτυο blockchain με αδειοδότηση μπορεί να είναι είτε ιδιωτικό (private), όπου υπάρχει μια κεντρική μορφή ελέγχου που επιτρέπει ή όχι την είσοδο των χρηστών στο δίκτυο είτε κοινοπραξίας (consortium), όπου ο έλεγχος επιτυγχάνεται από μία ομάδα χρηστών ή οργανισμών. Τέλος, το υβριδικό δίκτυο (hybrid) αποτελεί ένα συνδυασμό των παραπάνω με μία κεντρική μονάδα ελέγχου, όπου οι χρήστες προσδιορίζουν ποιες συναλλαγές θα γίνουν δημόσιες, προσβάσιμες από όλους και ποιες θα παραμείνουν ιδιωτικές [9].

## 1.6 Εφαρμογές Blockchain

Γνωστότερη πλατφόρμα διαχείρισης blockchain κρυπτονομισμάτων είναι η ανοιχτή προς όλους – χωρίς αδειοδότηση χρήστη – πλατφόρμα Ethereum [10]. Η λειτουργία της υλοποιείται με χρήση του αλγόριθμου συναίνεσης proof-of-work και διαθέτει μέχρι σήμερα χιλιάδες έξυπνα συμβόλαια που εκτελούνται σε αυτήν. Κάνοντας χρήση του κρυπτονομίσματος Ether, που δημιουργήθηκε για τις ανάγκες της πλατφόρμας, δίνει την δυνατότητα στους προγραμματιστές να αναπτύξουν έξυπνα συμβόλαια σε διάφορες turing-complete γλώσσες προγραμματισμού με πιο γνωστή και επικρατέστερη την Solidity [8], [11], γλώσσα που αναπτύχθηκε από την ίδια την πλατφόρμα για τον σκοπό αυτό, χρησιμοποιώντας στοιχεία των γλωσσών C++, JavaScript και Python. Με το κρυπτονόμισμα αυτό ανταμείβει τους χρήστες της που κάνουν "mining" προσφέροντας την υπολογιστική τους δύναμη για τις συναλλαγές που απαιτούνται και τα δεδομένα που ανταλλάσσονται [8].

### 1.6.1 Αποκεντρωμένες Εφαρμογές Ιστού

Το λογισμικό που χρησιμοποιείται σε επίπεδο εφαρμογών, με σκοπό να αλληλεπιδράσει με ένα blockchain δίκτυο ομότιμων κόμβων στηρίζεται τόσο στην γραφή του έξυπνου συμβολαίου όσο και στον προγραμματισμό της εφαρμογής. Ο συνδυασμός και των δύο αυτών παραγόντων δημιούργησε την ανάγκη για μια νέου είδους κατηγορία εφαρμογών που θα κάνουν χρήση των υπάρχουσών τεχνολογιών με παράλληλη εκμετάλλευση των δυνατοτήτων του blockchain. Τέτοιες εφαρμογές είναι οι γνωστές εφαρμογές αποκεντρωμένης λειτουργίας ή αλλιώς DApps. Μία αποκεντρωμένη εφαρμογή μπορεί να διαθέτει διεπαφή χρήστη (user Interface), με σκοπό την πιο άμεση και φιλική αλληλεπίδρασή της με τον χρήστη και να κάνει χρήση άλλων τεχνολογιών, όπως βάσεων δεδομένων, χρηματιστηριακές εφαρμογές κλπ. [12].

Η ειδοποιός διαφορά μεταξύ μιας κεντροποιημένης εφαρμογής ιστού και μίας αποκεντρωμένης μπορεί να συνοψιστεί σε τέσσερις βασικούς πυλώνες [12]:

- Η πρώτη κάνει χρήση αρχιτεκτονικής πελάτη-εξυπηρετητή, ενώ στην δεύτερη οι χρήστες επικοινωνούν μεταξύ τους μέσα από έξυπνα συμβόλαια.

- Ο backend κώδικας μιας αποκεντρωμένης εφαρμογής τρέχει σε ένα αποκεντρωμένο δίκτυο ομότιμων κόμβων, ενώ στην εφαρμογή ιστού οι βάσεις συνδέονται πάνω στον εξυπηρετητή και μέσω αυτού αποκτούν πρόσβαση οι χρήστες.
- Από τις συνήθεις εφαρμογές διαδικτύου απαιτείται ένα κεντρικό σημείο ελέγχου που θα επιτρέπει τις διαφόρων τύπων δραστηριότητες και θα καθορίζει τους ρόλους στο δίκτυο, σε αντίθεση με τις εφαρμογές του blockchain που λειτουργούν αποκεντρωμένα.
- Στις αποκεντρωμένες εφαρμογές κυριαρχούν η ανωνυμία, η ασφάλεια και η σταθερότητα, ενώ σε μία εφαρμογή ιστού, τα στοιχεία αυτά δεν θεωρούνται δεδομένα και αποτελούν αντικείμενο που πρέπει να διασφαλιστεί.

Στην δημιουργία τέτοιου είδους εφαρμογών κάνει έντονη την παρουσία της μία εφαρμογή από την πλευρά του πελάτη, η "Web3.js" [12], [13]. Όπως όλες οι άλλες εφαρμογές της κατηγορίας της, χρησιμοποιεί HTML, CSS και JavaScript για το στήσιμο των εφαρμογών ιστού. Ταυτόχρονα περιλαμβάνει και μία σειρά από χρήσιμες βιβλιοθήκες για την ομαλή αλληλεπίδραση της αποκεντρωμένης εφαρμογής με το blockchain. Επίσης, κάνει δυνατή την κατανόηση των έξυπνων συμβολαίων και εξυπηρετεί τις συναλλαγές μέσω του κρυπτονομίσματος ether.



# 2 Υπηρεσίες Web για Γεωχωρικά Δεδομένα

Η πλατφόρμα του istSOS αποτελεί ένα σύστημα διαχείρισης δεδομένων ανοιχτού τύπου με χρήση του προτύπου SOS (Sensor Observation Service) που περιγράφεται παρακάτω [14]. Πρόκειται για μία κοινοπραξία ανοιχτών προτύπων γεωγραφικών δεδομένων (Open Geospatial Consortium – OGC) που αποτελεί έναν διεθνή οργανισμό, στον οποίο ανήκουν και συνεργάζονται περισσότεροι από 500 εθελοντικοί, κυβερνητικοί, ερευνητικοί και μη κερδοσκοπικοί οργανισμοί, επιχειρήσεις και πανεπιστήμια, με σκοπό την παροχή γεωγραφικών δεδομένων, μετρήσεων και υπηρεσιών ενθαρρύνοντας και υποβοηθώντας την ανάπτυξη εφαρμογών ανοιχτών προτύπων που στηρίζονται στα γεωγραφικά δεδομένα τόσο για την επεξεργασία τους όσο και για τον διαμοιρασμό τους [15], [16]. Ακολουθώντας την παραπάνω αρχή OGC σε συνδυασμό με το SOS πρότυπο αλλά και τις τεχνολογίες των PostgreSQL/PostGIS και Apache HTTP Server αναπτύχθηκε η πλατφόρμα του istSOS, μιας εφαρμογής αποκλειστικά γραμμένης σε Python, από το Ελβετικό Ινστιτούτο Γεωεπιστημών (IST – Istituto Scienze della Terra) το 2009 [17]. Μέσω της πλατφόρμας αυτής προσφέρεται το κατάλληλο γραφικό περιβάλλον για την αναπαράσταση γεωγραφικών δεδομένων μέσω ενός RESTful Web API για αυτοματοποίηση διαδικασιών σε καθημερινή βάση [14].

## 2.1 Τυποποίηση OGC Sensor Web Enablement

Η τυποποίηση του OGC SWE δημιουργήθηκε ούτως ώστε να υλοποιηθεί ένας παγκόσμιος ψηφιακός ιστός αισθητήρων προσδιορίζοντας την τεχνολογία, την γλώσσα και την αρχιτεκτονική που χρησιμοποιείται [14], [17]. Σκοπός της SWE είναι να επιτευχθεί ένα διεθνές δίκτυο στο οποίο η πρόσβαση θα είναι ελεύθερη και οποιοσδήποτε ειδικός ή μη θα δύναται να αναζητήσει, να έχει πρόσβαση αλλά και να επεξεργαστεί γεωγραφικά δεδομένα από ένα ποικίλο εύρος ετερογενούς προέλευσης δικτύου αισθητήρων. Ως εκ τούτου, γίνεται σαφές πως έχουν αναπτυχθεί αρκετά πρότυπα που εξυπηρετούν αυτό το σκοπό με βασικότερα και πιο συνήθη τα ακόλουθα [18]:

- Observations and Measurements (O&M)  
Πρότυπο για την κωδικοποίηση των παρατηρήσεων που συλλαμβάνουν οι αισθητήρες.
- Sensor Model Language (SensorML)  
Πρότυπο για την πλήρη περιγραφή των χαρακτηριστικών των αισθητήρων.
- Sensor Observation Service (SOS)  
Μία υπηρεσία Web που δίνει πρόσβαση στα παραπάνω πρότυπα.

## 2.2 Πρότυπο OGC Sensor Observation Service

Ως μέρος της SWE, το SOS πρότυπο αποτελεί το βασικό κανάλι αλληλεπίδρασης των παρατηρήσεων των αισθητήρων με τους χρήστες. Η υπηρεσία SOS δεν χρησιμοποιείται μόνο, για να μπορέσει κάποιος χρήστης να αντλήσει τα μετρητικά μετεωρολογικά δεδομένα που χρειάζεται, απεναντίας, παρέχει δύο βασικές λειτουργίες με πρώτη, την παροχή παρατηρήσεων με βάση χωροχρονικά κριτήρια και δεύτερη, τη δυνατότητα περιγραφής ενός ολόκληρου συστήματος αισθητήρων ανάλογα με τα κριτήρια αυτά. Το SOS πρότυπο δεν αποτελεί απλώς μία υπηρεσία ιστού για την άντληση δεδομένων μετεωρολογικού ενδιαφέροντος στους χρήστες της, αλλά ταυτόχρονα τους δίνει την δυνατότητα να φιλτράρουν τα δεδομένα αυτά χρησιμοποιώντας ένα μεγάλο εύρος περιορισμών. Τέτοιου είδους περιορισμοί προκύπτουν καθορίζοντας τον αισθητήρα που επιθυμεί να επεξεργαστεί τα δεδομένα, την γεωγραφική περιοχή που μελετά, το χρονικό πλαίσιο ή τα χρονικά διαστήματα που τους είναι απαραίτητα, καθώς επίσης και την αντιπαράθεση μιας σειράς των παραπάνω μεταξύ τους. Η επικοινωνία αυτή επιτυγχάνεται, όπως και στα περισσότερα πρότυπα του OGC, με τη συνεχή ανταλλαγή μηνυμάτων, μέσω αιτημάτων – απαντήσεων (request – response) που πραγματοποιούνται μεταξύ της υπηρεσίας, εκείνης που προσφέρει το πρότυπο και του πελάτη σύμφωνα με το HTTP πρωτόκολλο[19]. Σύμφωνα με αυτό τα αιτήματα στέλνονται είτε ως HTTP Post, για να στείλουν δεδομένα στην υπηρεσία είτε ως HTTP Get, για να πάρουν δεδομένα από αυτήν, αφού πρώτα έχουν υποβάλει ένα έγκυρο ζευγάρι κλειδιού – τιμής (KVP – key value pair), το είδος του αιτήματος και τις απαραίτητες παραμέτρους. Για να επιτύχει μία τέτοιου είδους επικοινωνία, το OGC SOS πρότυπο έκδοσης 1.0 απαιτεί κατά την αποστολή αιτήματος προς αυτό να συμπεριλαμβάνονται τουλάχιστον τα τρία βασικά αιτήματα πυρήνα του SOS προφίλ. Τα πιο βασικά και κυρίαρχα αιτήματα που είναι διαθέσιμα από την υπηρεσία του OGC SOS είναι πέντε και περιγράφονται στον παρακάτω πίνακα [17].

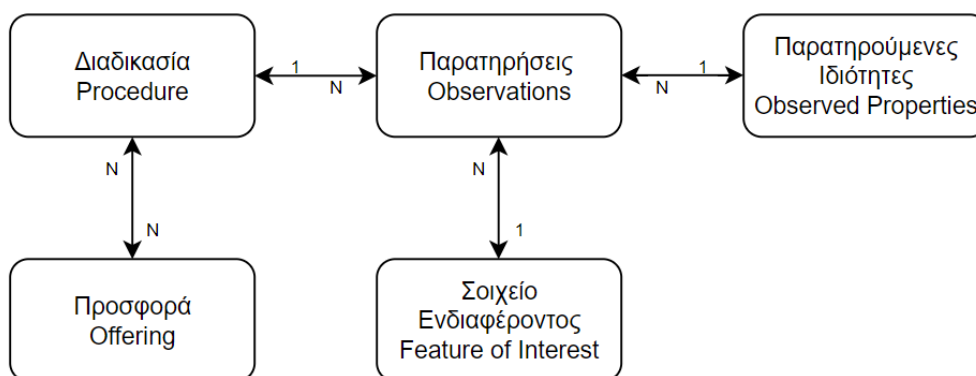
SOS αιτήματα	Προφίλ	Σύντομη περιγραφή
GetCapabilities	Πυρήνα	Επιτρέπει να γίνει περιγραφή της υπηρεσίας παρέχοντας πληροφορίες για τον διαχειριστή, τις δυνατότητες που προσφέρονται, τις ιδιότητες που παρατηρούνται κλπ.
DescribeSensor	Πυρήνα	Παρέχει πληροφορίες για κάποιο συγκεκριμένο τμήμα, σύστημα ή διεργασία σε μορφή SensorML.
GetObservation	Πυρήνα	Επιστρέφει τις παρατηρήσεις που είναι διαθέσιμες βάση των φίλτρων που έχουν εφαρμοστεί σχετικά με την ώρα, τον τόπο, τον αισθητήρα κλπ.
RegisterSensor	Συναλλαγματικό	Παρέχει την δυνατότητα να προστεθεί ένας καινούργιος αισθητήρας στην υπάρχουσα υπηρεσία.

InsertObservation	Συναλλαγματικό	Παρέχει την δυνατότητα να προστεθούν δυναμικά καινούργιες παρατηρήσεις - μετρήσεις για έναν υπάρχων αισθητήρα της υπηρεσίας.
-------------------	----------------	--

Πίνακας 4.2 Σύντομη περιγραφή των βασικότερων SOS αιτημάτων

Το πρότυπο OGC SOS έχει πέντε βασικά κλειδιά – αξίες που ακολουθεί:

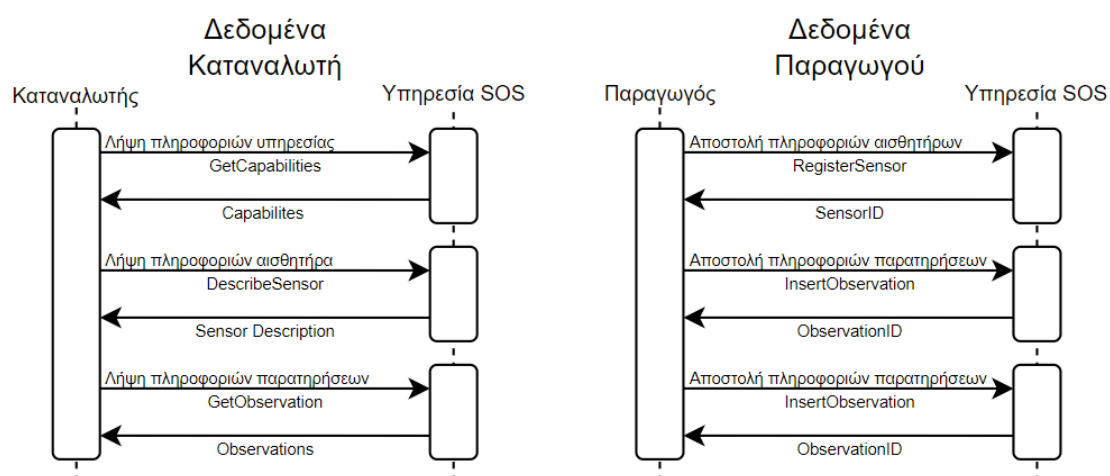
- Παρατηρήσεις (Observations)  
Αποτελούν το σημείο αναφοράς του προτύπου, καθώς αναπαριστούν τιμές δεδομένων σε συγκεκριμένο στιγμιότυπο και αναπαρίστανται με βάση το πρότυπο του O&M μοντέλου δεδομένων.
- Διαδικασία (Procedure)  
Προσδιορίζει τον πάροχο των παρατηρήσεων, όπου κατά βάση πρόκειται για έναν απλό αισθητήρα και αναπαρίσταντο σύμφωνα με το πρότυπο του SensorML μοντέλου δεδομένων.
- Παρατηρούμενες ιδιότητες (Observed Properties)  
Πρόκειται για τα φαινόμενα που είναι παρατηρήσιμα και αναπαρίστανται με ένα URI (Uniform Resource Identifier).
- Στοιχείο ενδιαφέροντος (Feature of Interest)  
Αναφέρεται στο στοιχείο εκείνο που συσχετίζει τις παρατηρήσεις μεταξύ τους.
- Προσφορά (Offering)  
Περιγράφει μια απλή ομαδοποίηση των αισθητήρων για πρακτικούς λόγους.



Εικόνα 2-1 Διάγραμμα ροής των πέντε βασικών κλειδιών λειτουργίας του SOS προτύπου

Εξειδικεύοντας, ο τρόπος με τον οποίο λειτουργεί η πλατφόρμα του istSOS μέσω του προτύπου SOS διακρίνεται ανάλογα με την όψη του πελάτη με τον οποίο συναλλάσσεται [17]. Στην περίπτωση του καταναλωτή μπορούν να χρησιμοποιηθούν τα SOS αιτήματα GetCapabilities, DescribeSensor και GetObservation, με απώτερο

σκοπό να ληφθούν δεδομένα τόσο για την υπηρεσία και τους αισθητήρες που αποτελείται όσο και για τις παρατηρήσεις που διαθέτει στην βάση της, αντίστοιχα. Από την άλλη πλευρά, στην περίπτωση του παραγωγού, του χρήστη δηλαδή που αποτελεί έναν φορέα παροχής δεδομένων, με σκοπό την διαχείρισή τους από την πλατφόρμα, η αλληλεπίδραση πραγματοποιείται με τα αιτήματα RegisterSensor και InsertObservation από τα οποία καταχωρούνται στην βάση όλες οι απαραίτητες πληροφορίες για τους αισθητήρες, καθώς και το σύνολο των παρατηρήσεων που αναμένεται να μεταφορτωθούν. Ένα σχηματικό διάγραμμα που συνοψίζει την διαφορετική αλληλεπίδραση καταναλωτή – παραγωγού με το πρότυπο SOS στην πλατφόρμα του istSOS παρουσιάζεται στην από κάτω εικόνα.



Εικόνα 2-2 Τυπικό SOS UML διάγραμμα για τις περιπτώσεις αλληλεπίδρασης με καταναλωτή (αριστερά) και με παραγωγό (δεξιά)

## 2.3 Μορφές Αναπαράστασης Δεδομένων για Υπηρεσίες Web

Οι μορφές αναπαράστασης δεδομένων είναι ένα είδος κωδικοποίησης που επιλέγεται από τις διάφορες υπηρεσίες ιστού, αλλά και εφαρμογές, προκειμένου να αποθηκεύσουν τα δεδομένα τους με τρόπο συμβατό και κατανοητό τόσο προς την μηχανή όσο και προς τον άνθρωπο. Ένας χρήστης, για να μπορέσει να αλληλεπιδράσει μέσω αιτημάτων και απαντήσεων με έναν εξυπηρετητή ιστού, απαιτείται μία κατάλληλη μορφή δεδομένων, δηλαδή τα δεδομένα αυτά να είναι διαρθρωμένα με τρόπο τέτοιο, ώστε να εξυπηρετείται η παρούσα διαδικασία όσο γίνεται ευκολότερα και πιο άμεσα. Μέχρι σήμερα έχουν αναπτυχθεί και χρησιμοποιούνται τρεις βασικές μορφές δομής δεδομένων κατάλληλων για μετάδοση: XML, CSV και JSON.

### 2.3.1 Μορφότυπος eXtensible Markup Language – XML

Ο μορφότυπος eXtensible Markup Language – XML σχεδιάστηκε το 1996 και αποτέλεσε από το 1998 βασικό πρότυπο για τις υπηρεσίες ιστού. Είναι μία γλώσσα



σήμανσης και χρησιμοποιείται για δεδομένα που θέλουν να διατηρήσουν ιεραρχική δομή μεταξύ τους έχοντας την ακόλουθη δομή:

XML	<pre> &lt;person&gt;   &lt;name&gt;     &lt;br/&gt;Eric&lt;br/&gt;   &lt;/name&gt; &lt;br/&gt; &lt;age&gt;   &lt;br/&gt;26&lt;br/&gt; &lt;/age&gt; &lt;/person&gt; </pre>	XML
-----	---	-----

Όπως φαίνεται στο ανωτέρω παράδειγμα, τα δεδομένα παρουσιάζουν μία ξεκάθαρη ιεραρχική δομή ευανάγνωστη και ευδιάκριτη τόσο από προγράμματα και εφαρμογές όσο και από τον ίδιο τον άνθρωπο. Τα πεδία "<name>" και "<age>" περιέχονται στο "<person>" και έχουν τιμές τις "<br/>Eric<br/>" και "<br/>26<br/>", αντίστοιχα. Βασικό χαρακτηριστικό της δομής αυτής είναι ο χωρισμός των δεδομένων σε οντότητες που δηλώνουν ακριβώς πού ξεκινούν (<person>) και πού τελειώνουν (</person>). Η χρήση "/" είναι εκείνη που διαχωρίζει την αρχή από το τέλος κάθε διαφορετικού πεδίου του αρχείου XML και προσδιορίζει όλο το περιεχόμενό του [20].

### 2.3.2 Μορφότυπος Comma Separated Values – CSV

Ο μορφότυπος Comma Separated Values – CSV αποτελεί την πιο συνήθη μορφή εισαγωγής και εξαγωγής δεδομένων σε λογιστικά φύλλα και βάσεις δεδομένων. Είναι ένα είδος αρχείου που αντί να αποθηκεύει τα δεδομένα σε στήλες, τα αποθηκεύει διαχωρίζοντάς τα με κόμμα. Όταν αποθηκεύονται δεδομένα – κειμένου ή αριθμητικά – σε ένα αρχείο CSV, καθιστούν την μεταφορά τους από το ένα πρόγραμμα στο άλλο αρκετά εύκολη, καθώς η πλειονότητα προγραμμάτων και εφαρμογών γνωρίζει πώς να διαχειρίζεται τέτοιου είδους αρχεία.

CSV	<pre> μέτρηση1,9.1,9.3,9.1,92,7.9,4.8,5,0.80,8.0,6,8.8 μέτρηση2,9.3,9.3,9.1,94,8.4,4.8,5,0.80,8.0,4,9.1 μέτρηση3,9.2,9.3,9.2,93,8.1,3.2,8,0.53,8.0,4,9.2 μέτρηση4,9.1,9.2,9.1,94,8.2,6.4,8,1.07,8.9,8,8.3 μέτρηση5,9.1,9.3,9.0,94,8.2,3.2,8,0.53,6.4,8,9.1 μέτρηση6,9.5,9.5,9.2,93,8.4,3.2,9,0.53,8.0,8,9.5 </pre>	CSV
-----	--	-----

Όπως φαίνεται και στο παράδειγμα παραπάνω, κάθε γραμμή του αρχείου κειμένου αποτελεί μία διαφορετική εγγραφή. Κάθε εγγραφή περιέχει δεδομένα χωρισμένα με κόμμα μεταξύ τους υποδηλώνοντας με τον τρόπο αυτό την αρχή και το τέλος του καθενός [20].

### 2.3.3 Μορφότυπος JavaScript Object Notation – JSON

Αποτελεί εναλλακτική επέκταση του μορφότυπου XML που δημιουργήθηκε το 2001 και έγινε ευρέως γνωστό από τις υπηρεσίες Yahoo και Google την διετία 2005-2006. Έχοντας ως αφετηρία του την λογική του XML τύπου αρχείων χωρίζει τα δεδομένα με την χρήση κόμματος αλλά και παρενθέσεων και αγκυλών, όπως φαίνεται παρακάτω.

JSON	<pre>{ "person":   [     {       "name": "Eric",       "age": 26,       "student": false     }, {       "name": "Andrea",       "age": 21,       "student": true     }   ] }</pre>	JSON
------	--	------

Στο παράδειγμα που παρουσιάστηκε, το JSON αντικείμενο αποτελείται από ένα πεδίο "person" που έχει σαν τιμή έναν πίνακα με δύο στοιχεία. Κάθε στοιχείο είναι ένα αντικείμενο JSON με πεδία τα "name", "age" και "student". Ενώ το μέγεθος ενός XML αρχείου είναι σχετικά μεγάλο, καθώς απαιτεί δύο επιπλέον ετικέτες για κάθε όρο, το μέγεθος ενός JSON είναι αρκετά μικρότερο. Δημιουργήθηκε, για να μπορεί να φορτώνει δεδομένα με σχετικά εύκολο τρόπο σε τοπικά JSON αντικείμενα, καθιστώντας την χρηστική για εφαρμογές διαδικτύου. Χαρακτηρίζεται ως λιγότερο φλύαρη έναντι της XML και για αυτό έχει αξιοποιηθεί ιδιαίτερα στο προγραμματιστικό περιβάλλον παρέχοντας αμεσότητα και συμβατότητα στις διαδικτυακές επικοινωνίες [20].

## 2.4 PostgreSQL – PostGIS

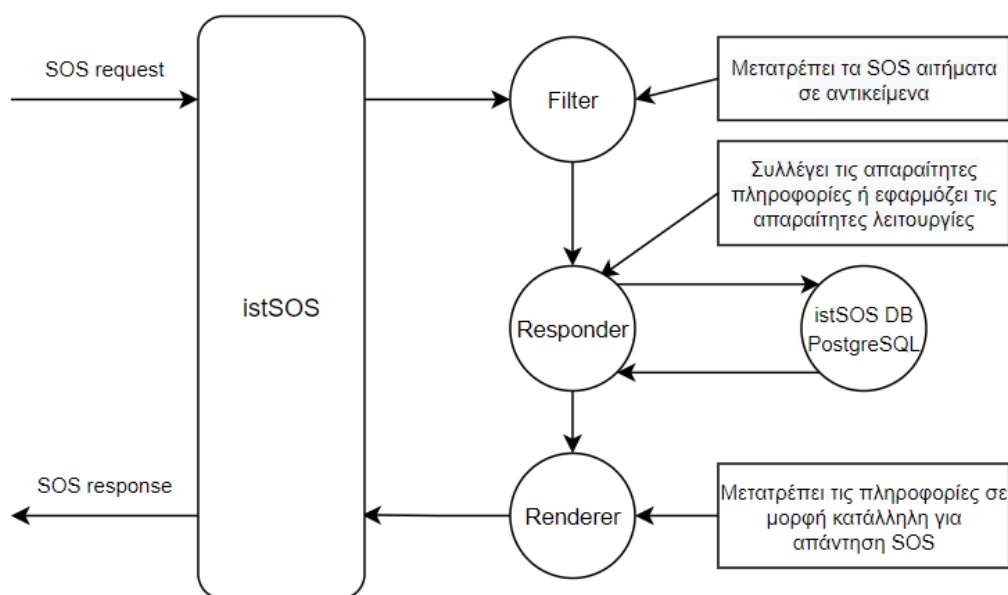
Η PostgreSQL είναι μία αντικειμενό-σχεσιακή (object-relational) βάση δεδομένων ανοιχτού κώδικα που αποτελεί επέκταση της γλώσσας προγραμματισμού SQL διασφαλίζοντας την ασφαλή αποθήκευση και διαχείριση μεγάλου όγκου δεδομένων. Αναπτύχθηκε το 1986 σαν κομμάτι της Postgres στο πανεπιστήμιο της Καλιφόρνια στο Μπέρκλεϊ. Έχει επιλεγεί από την πλατφόρμα του istSOS για την αποθήκευση όλων των γεωγραφικών δεδομένων που ανεβαίνουν σε αυτό, καθώς έχει διακριθεί για την ακεραιότητα των δεδομένων, την αξιοπιστία και την ορθή λειτουργία της. Η PostgreSQL είναι συμβατή με τα περισσότερα λειτουργικά συστήματα εκ των οποίων αναφέρονται τα Windows, τα Linux και τα UNIX [21]. Τα γεωγραφικά δεδομένα που αποστέλλονται από το istSOS αναγνωρίζονται και διαχειρίζονται από την PostgreSQL χάρις την υποστήριξη του λογισμικού ανοιχτού κώδικα PostGIS (GIS – Geographic Information System) που επιτρέπει την διατύπωση ερωτημάτων σε SQL [22].

## 2.5 Apache HTTP Εξυπηρετητής

Έχοντας ως σύνθημά του και κύριο λόγο δημιουργίας του την παροχή λογισμικού με αποστολή του το ευρύτερο καλό, ο Apache HTTP εξυπηρετητής αποτελεί τον δημοφιλέστερο εξυπηρετητή ιστού παγκοσμίως. Για να γίνει πιο κατανοητό, όταν κάποιος χρήστης επισκέπτεται έναν ιστότοπο, το πρόγραμμα πλοήγησης (browser) επικοινωνεί με τον εξυπηρετητή ιστού μέσω του HTTP/HTTPS πρωτοκόλλου προκειμένου να φορτωθούν οι αντίστοιχες ιστοσελίδες [23]. Ο εξυπηρετητής αυτός υποστηρίζεται και συντηρείται από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation – ASF).

## 2.6 Αρχιτεκτονική Πλατφόρμας του IstSOS

Συμπερασματικά, η γενικότερη λειτουργία της πλατφόρμας του istSOS, ως μίας εφαρμογής ιστού, είναι να μπορεί να δέχεται, να επεξεργάζεται και να προσφέρει χωροαναφερόμενα χρονομεταβλητά μετρητικά δεδομένα στους χρήστες του. Ένας χρήστης επικοινωνεί μαζί του μέσω του προτύπου SOS στέλνοντας αιτήματα, τα οποία προσδιορίζουν με σαφήνεια την λειτουργία που θέλει να εκτελεστεί. Το istSOS λαμβάνει αυτά τα αιτήματα, τα φιλτράρει ανάλογα με τους περιορισμούς και τις μεταβλητές που έχουν οριστεί από το χρήστη κατά την σύνταξη του αιτήματος και εν συνεχεία, κάνει κλήση στην βάση του, το PostgreSQL είτε για να κάνει λήψη από τα δεδομένα που βρίσκονται ήδη σε αυτήν είτε για να προσθέσει νέα. Τα δεδομένα αυτά πρέπει να είναι σε μορφή XML ή JSON για να μπορέσουν να γίνουν κατανοητά από την πλατφόρμα, ενώ σε κάθε άλλη περίπτωση, θα πρέπει να μετατραπούν με την χρήση βοηθητικών συναρτήσεων που παρέχονται. Έπειτα, το istSOS τα μετατρέπει σε μορφή κατάλληλη για το SOS πρότυπο και στέλνει απάντηση πίσω στον χρήστη. Όλα αυτά επιτυγχάνονται με την βοήθεια του Apache HTTP εξυπηρετητή που διαχειρίζεται όλα αυτά τα αιτήματα για την ομαλή λειτουργία του istSOS [14], [19]. Μια απλοποιημένη σχηματική αναπαράσταση της παραπάνω λειτουργίας παρουσιάζεται στην παρακάτω εικόνα [24].



Εικόνα 2-3 Σχηματική απεικόνιση της ροής λειτουργίας της πλατφόρμας του istSOS

Όπως φαίνεται στην σχηματική απεικόνιση, το λογισμικό του istSOS έχει αναπτυχθεί σε τρία ξεχωριστά τμήματα, περιγραφικά των λειτουργιών της ως εξής:

- **Φίλτρο (Filter)**  
Πρόκειται για την διεπαφή που χρησιμοποιείται στην μετατροπή των HTTP αιτημάτων που στέλνονται από τον χρήστη σε αντικείμενα ρυθιστά σύμφωνα με το πρότυπο OGC SOS.
- **Ανταποκριτής (Responder)**  
Υπεύθυνος για την περάτωση των διαδικασιών που απαιτούνται με βάση το αίτημα που στάλθηκε. Έχει πρόσβαση στην βάση δεδομένων του istSOS, από όπου και αντλεί όλες τις απαραίτητες πληροφορίες που χρειάζεται για την επίλυση των διαδικασιών αυτών.
- **Αντικείμενο Απόδοσης (Renderer)**  
Είναι το εργαλείο εκείνο που αξιοποιείται από το σύστημα του istSOS για την μετατροπή των πληροφοριών που συλλέχθηκαν από τον ανταποκριτή σε μορφή συμβατή με το πρότυπο OGC SOS, με σκοπό την αποστολή της απάντησης στον χρήστη.

## 2.7 Μετρητικά Δεδομένα με Χωρική Αναφορά

Όσο η τεχνολογία εξελίσσεται και οι σύγχρονες συνθήκες ζωής έχουν δημιουργήσει νέες απαιτήσεις για την περάτωση τεχνολογιών και εφαρμογών που παλιότερα δεν υπήρχαν, οι ανάγκες για ευρύτερη και πληρέστερη συλλογή δεδομένων πληθαίνουν. Σήμερα, συλλέγονται δεδομένα από κάθε είδους δυνατή πηγή που συναντάτε στην καθημερινότητα του ανθρώπου, έτσι ώστε να βελτιώνεται

το βιοτικό του επίπεδο και οι παροχές του σε κάθε είδους δραστηριότητα. Τα δεδομένα αυτά αξιοποιούνται από την επιστημονική κοινότητα, καθώς γίνονται αντικείμενο ερευνών, προβληματισμού και ασφάλειας. Συγκεκριμενοποιώντας, τεχνολογίες μηχανικής μάθησης και αυτοματοποίησης διαδικασιών θα θεωρούνταν ανέφικτες, αν δεν υπήρχε από πίσω μία αξιολογικά ικανοποιητική και κατάλληλα στοχευμένη βάση δεδομένων με δεδομένα κατάλληλα για την προεπεξεργασία που απαιτούν οι παραπάνω καινοτομίες. Τα δεδομένα αυτά χρήζουν διασφάλισης και επικύρωσης ιδιαίτερα στην περίπτωση των μετρητικών δεδομένων πάνω στα οποία βασίζονται επιχειρησιακές αποφάσεις, αλλά και μελέτες και έρευνες ευρύτερης κλίμακας και σημασίας.

Πέραν των τεχνολογιών που βασίζουν την αξιοπιστία τους στην βάση δεδομένων που διαθέτουν, τα δεδομένα και ως επί το πλείστον, τα μετρητικά δεδομένα είναι εκείνα στα οποία βασίζονται οι αρμόδιες αρχές προκειμένου να οδηγηθούν στην λήψη αποφάσεων σχετικά με την προάσπιση των συμφερόντων των πολιτών και την εξάλειψη πιθανών καταστάσεων που κοστίζουν ανθρώπινες ζωές.

### **2.7.1 Κίνδυνοι Ακεραιότητας Δεδομένων**

Ήδη, από την εποχή ακμής των ηλεκτρονικών υπολογιστών προτάθηκαν σημαντικά ζητήματα ασφαλείας, τα οποία εντάθηκαν με την εξέλιξη του διαδικτύου και πήραν σημαντικές διαστάσεις με την ανάπτυξη του Διαδικτύου των Πραγμάτων (Internet of Things – IoT). Όλο και περισσότεροι πολίτες και επιχειρήσεις έχουν αρχίσει να αμφισβητούν την ακεραιότητα και την ακρίβεια των δεδομένων που διαθέτουν και διαχειρίζονται. Αυτές οι αμφιβολίες δημιούργησαν την απαίτηση για ακεραιότητα των δεδομένων που διατίθενται στους χρήστες και αλληλοεπιδρούν μέσω αυτών στο διαδίκτυο. Η ακεραιότητα των δεδομένων (Data Integrity) έχει να κάνει με την αξιοπιστία που παρουσιάζουν, δηλαδή να μην μπορεί κάποιο κακόβουλο λογισμικό ή χρήστης να τα τροποποιήσει επί σκοπώ και να τα αντικαταστήσει χωρίς να γίνει αντιληπτός από το δίκτυο. Η κατά αντιστοιχία έλλειψη ακεραιότητας θέτει σημαντικά ζητήματα ασφαλείας [25].

Οι υπηρεσίες αποθήκευσης στο διαδίκτυο (cloud), που χρησιμοποιούνται σήμερα από την πλειοψηφία των ανθρώπων παγκοσμίως για αποθήκευση και διαχείριση προσωπικών και μη δεδομένων, θέτουν τα ζητήματα ασφαλείας σε πρώτο πλάνο και απαιτούν την ανάπτυξη μηχανισμών για την προστασία και ασφάλειά τους. Έτσι, έχει δημιουργηθεί η ανάγκη για σχεδιασμό ασφαλών πολιτικών στα πληροφοριακά συστήματα, απαίτηση που συνδέεται άμεσα τόσο με τεχνικές διαδικασίες και διοικητικά μέτρα όσο και με ηθικο-κοινωνικές αντιλήψεις και αρχές προφυλάσσοντας από κάθε είδους απειλή σκόπιμη ή μη. Οι διαδικασίες σχεδιασμού των πολιτικών αυτών θα πρέπει να μην επεμβαίνουν στην απρόσκοπτη λειτουργία των πληροφοριακών συστημάτων και να μπορούν να εγγυηθούν την προστασία των δεδομένων τόσο κατά την μεταφορά τους μεταξύ των χρηστών ή μεταξύ χρήστη και δικτύου όσο και κατά την παραμονή τους σε αυτό, όταν δηλαδή θεωρούνται αδρανή και παραμένουν αποθηκευμένα σε κάποια βάση προκειμένου να αξιοποιηθούν μελλοντικά.

Σύμφωνα με τα παραπάνω, παρουσιάζεται επιτακτική η ανάγκη για όλες τις επιχειρήσεις και τους οργανισμούς που διαχειρίζονται και χρησιμοποιούν δεδομένα – ιδιαίτερης σημασίας – να έχουν αναπτύξει ένα ασφαλές περιβάλλον αποθήκευσης και προώθησης τους, προκειμένου να μην μπορεί οποιοδήποτε κακόβουλο λογισμικό ή πρόσωπο να επέμβει σε αυτά για την ικανοποίηση ιδιωτικών συμφερόντων. Θα πρέπει να διαθέτουν ένα σύστημα αποτελεσματικής ταυτοποίησης των δεδομένων τους και διαφάνειας των συναλλαγών, ώστε να τεκμηριώνεται με κάθε δυνατό τρόπο οποιαδήποτε τροποποίηση ή διαγραφή.

Επίσης, ένα σύστημα που προσφέρει ακεραιότητα στα δεδομένα του μπορεί να διαλευκάνει, σε περίπτωση μεγάλων φυσικών καταστροφών, οι οποίες μόνες τους ή σε συνδυασμό με την ανθρώπινη δράση μπορεί να καταλήξουν στην απώλεια της ανθρώπινης ζωής, να προσφέρουν δεδομένα για την τεκμηρίωση των γεγονότων

### **2.7.2 Μετεωρολογικά Δεδομένα**

Δεδομένα που λαμβάνονται από αισθητήρες, για να καταγράψουν διάφορα μεγέθη χαρακτηρίζονται ως μετεωρολογικά δεδομένα. Ένα σύνολο πληροφοριών μέσω των οποίων περιγράφονται χαρακτηριστικά της ατμόσφαιρας. Ανοιχτά δεδομένα είναι εκείνα στα οποία οι πολίτες έχουν εξ ορισμού ελεύθερη πρόσβαση, καθώς η διάθεσή τους είναι ανοιχτή βάση νόμου δίνοντας τους την δυνατότητα για περαιτέρω χρήση και ανάλυση. Μια κατηγορία αυτών είναι τα πρωτογενή μετεωρολογικά δεδομένα, όπως είναι η θερμοκρασία, η υγρασία, η βροχόπτωση κλπ. που παράγονται από την πλειοψηφία των μετεωρολογικών σταθμών. Σύμφωνα με το νομοθετικό πλαίσιο που ισχύει στην Ελλάδα, οι δημόσιοι φορείς παράγουν πρωτογενή μετεωρολογικά δεδομένα και είναι υποχρεωμένοι να τα διαθέτουν ανοιχτά και με κατάλληλη αδειοδότηση που να επιτρέπει την επαναχρησιμοποίηση τους από όποιον επιθυμεί να τα αξιολογήσει, για οποιαδήποτε χρήση.

Με το πέρασμα των χρόνων έχει καταστεί επιτακτική η ανάγκη για περισσότερους μετεωρολογικούς σταθμούς λόγω της συνειδητοποίησης που υπάρχει σήμερα από την ευρύ κοινότητα για αντικειμενική συλλογή περισσότερων αντιπροσωπευτικών μετεωρολογικών χαρακτηριστικών της χώρας. Τρεις από τους πιο γνωστούς δημόσιους φορείς της Ελλάδας που παράγουν τέτοιου είδους δεδομένα και διαθέτουν ιδιόκτητους μετεωρολογικούς σταθμούς είναι οι ακόλουθοι:

- Εθνική Μετεωρολογική Υπηρεσία (EMY)
- Εθνικό Αστεροσκοπείο Αθηνών (Meteo)
- Ελληνικό Κέντρο Θαλάσσιων Ερευνών (Poseidon)

Σύμφωνα με τα μέχρι τώρα στοιχεία, τα δεδομένα που συγκεντρώνονται από τους διάφορους φορείς δεν είναι πάντοτε εύκολα προσβάσιμα, ενώ σε κάποιες περιπτώσεις η ποιότητα που διατίθενται δεν διευκολύνει την αποτελεσματική χρήση τους. Αυτό επιβάλλεται να αλλάξει και να υιοθετηθεί από όλους τους φορείς μια

ενιαία στρατηγική για την ομαλή διάθεση, λειτουργία και διαχείριση των δεδομένων αυτών.

### **2.7.3 Φυσικές Καταστροφές, Έντονα Καιρικά Φαινόμενα και Επιπτώσεις αυτών**

Εξαιτίας της κλιματικής κρίσης που κάνει σταδιακά εντονότερη την παρουσία της, τα καιρικά φαινόμενα εξελίσσονται όλο και συχνότερα σε ακραία με όλο και πιο έντονες επιπτώσεις. Χιλιάδες άνθρωποι κάθε χρόνο χάνουν τις περιουσίες τους, οι καλλιέργειες απειλούνται και τα οικοσυστήματα καταστρέφονται. Σε κάποιες περιπτώσεις τίθεται σε κίνδυνο ακόμη και η ανθρώπινη ζωή. Η Ελλάδα και γενικότερα η νότια ακτογραμμή της Ευρώπης, απειλείται ήδη με καταστροφικές πυρκαγιές, εξαφάνιση παράκτιων κοινοτήτων και ακτών, ξηρασία και ερημοποίηση· επιπτώσεις της κλιματικής αλλαγής που διαρκώς αυξάνονται [26].

Οι συνέπειες της κλιματικής κρίσης είναι πολλές και ποικίλες. Μόνο η ατμοσφαιρική ρύπανση, σύμφωνα με τον Παγκόσμιο Οργανισμό Υγείας (ΠΟΥ), εκτιμάται ότι οδηγεί στον θάνατο ετησίως επτά εκατομμύρια ανθρώπους και αποτελεί κατά αυτόν τον τρόπο μια πραγματική απειλή για την δημόσια υγεία και το κλίμα. Μετεωρολογικοί σταθμοί καταγράφουν διαρκώς τις αλλαγές αυτές στην ατμόσφαιρα και παρέχουν σημαντική βοήθεια στην επιστημονική κοινότητα που αναζητά λύσεις.

### **2.7.4 Ανάγκη Αξιοπιστίας Μετρητικών Δεδομένων**

Το νομικό υπόβαθρο που εφαρμόζεται σε μια φυσική καταστροφή ορίζεται σαφέστατα στην πλειονότητα των χωρών παγκοσμίως. Όταν συμβαίνει μία φυσική καταστροφή στην οποία μπορεί να ενυπάρχει και απώλεια ανθρωπίνων ζωών, η νομική διερεύνηση είναι μονόδρομος και αναζητά το μερίδιο της ανθρώπινης υπαιτιότητας στο τελικό αποτέλεσμα. Με βάση αυτό άνθρωποι και οργανισμοί που μοιράζονται αυτό το μερίδιο ευθύνης κινδυνεύουν, γιατί δεν ερμήνευσαν σωστά ή αδιαφόρησαν για τα δεδομένα που είχαν στην διάθεσή τους από τους αρμόδιους φορείς. Προκειμένου, λοιπόν, να αποφευχθούν περιπτώσεις παραποίησης των πρωτογενών δεδομένων απαιτείται ένα σύστημα που να προσφέρει ακεραιότητα στα δεδομένα του. Αυτό μπορεί να διαλευκάνει οποιαδήποτε νομική υπόθεση και να προσφέρει δεδομένα για την τεκμηρίωση των γεγονότων.

Στην εργασία αυτή, επιχειρείται μία υλοποίηση μιας εφαρμογής που συνδυάζει τη διαχείριση μετρητικών δεδομένων ενός μεγάλου (500+) δικτύου μετεωρολογικών σταθμών, με πρότυπες τεχνολογίες (OGC SOS), σε συνδυασμό με τη διασφάλιση της ακεραιότητας των δεδομένων και την προστασία τους από αλλοιώσεις, μέσω ενός blockchain.





# Μέρος Β

## Ανάπτυξη Εφαρμογής

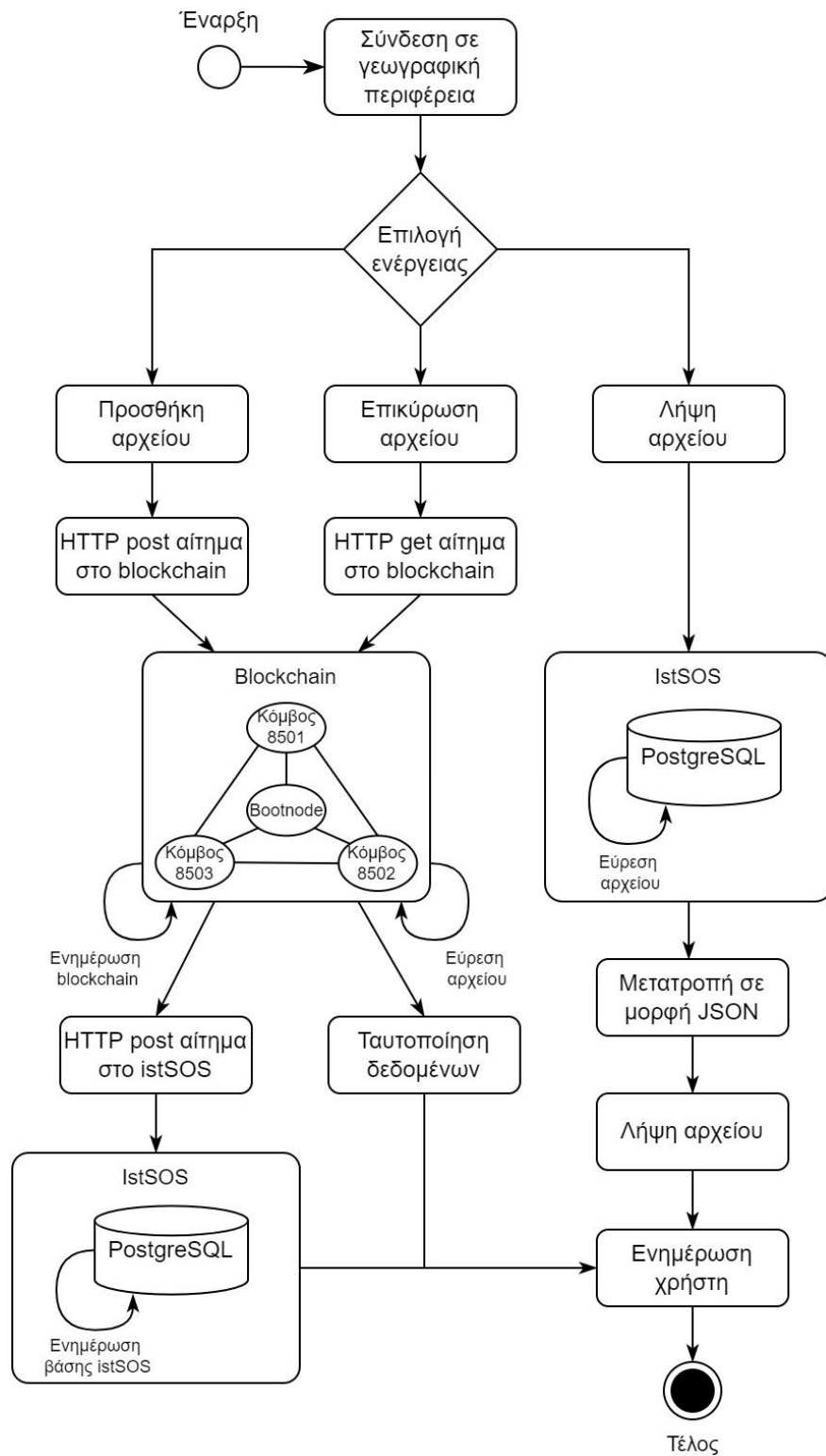
---

### 3 Μεθοδολογία

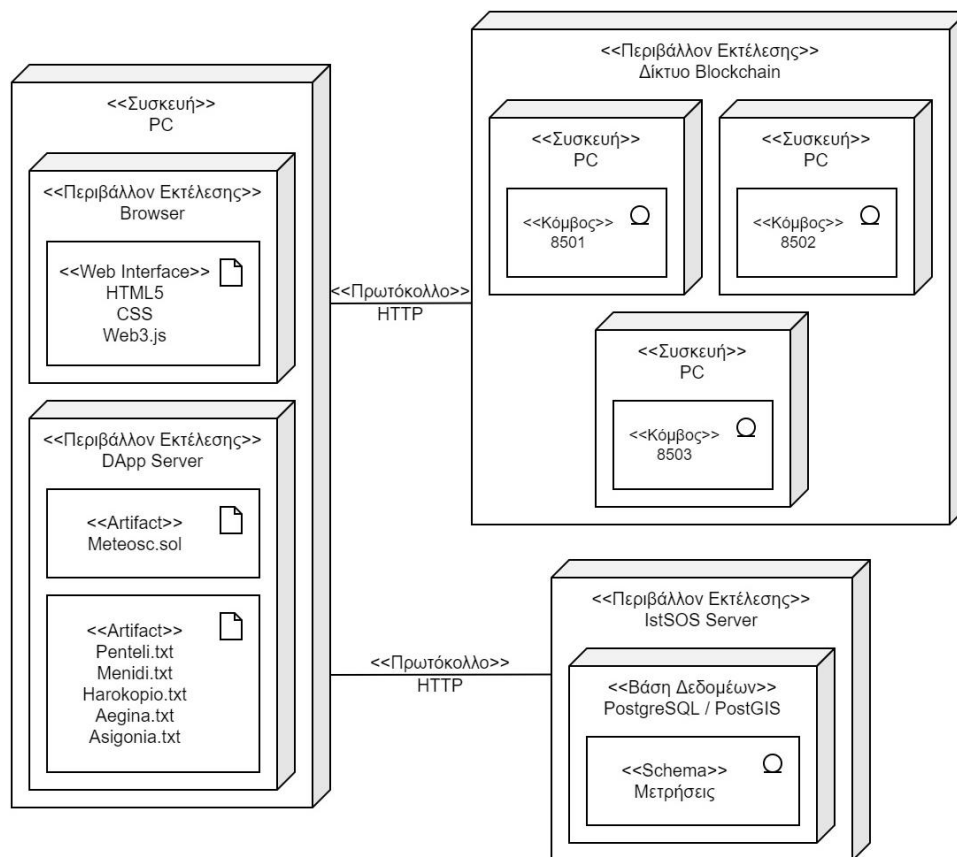
Η προστασία των δεδομένων, όπως αναφέρθηκε ήδη στο κεφάλαιο 5 «Μετρητικά Δεδομένα – Ανάγκη Αξιοπιστίας Μετρητικών Δεδομένων», αποτελεί ένα από τα βασικότερα ζητήματα που απασχολούν την διεθνή κοινότητα σήμερα, αναφορικά με το αν οι μετρήσεις που προσφέρουν και επεξεργάζονται είναι και παραμένουν αξιόπιστες. Ειδικότερα τα μετεωρολογικά δεδομένα δηλαδή τα δεδομένα πάνω στα οποία στηρίζονται οι σημαντικότερες δομές μιας κοινωνίας – πυροσβεστική, αστυνομία, ερυθρός σταυρός κλπ. – για να μπορέσουν να βρίσκονται σε ετοιμότητα, όταν οι συνθήκες το απαιτούν, οφείλουν να διέπονται από πλήρη ακεραιότητα και να μην γίνονται αντικείμενο παραποίησης ή παρερμηνείας από κανέναν.

Στην παρούσα διπλωματική εργασία, μελετάται και προτείνεται μια τεχνική διασφάλισης μετεωρολογικών δεδομένων ήδη γνωστή για την συνεισφορά της στον τομέα της ασφάλειας, αυτή του blockchain. Προτείνεται η υιοθέτηση της τεχνολογίας αυτής προκειμένου οι φορείς που διαχειρίζονται και διανέμουν ανοιχτά δεδομένα μετεωρολογικού ενδιαφέροντος να την χρησιμοποιούν έτσι ώστε να εξασφαλίζουν μέσω συναλλαγών την ακρίβεια και την αξιοπιστία των μετρήσεων. Ειδικότερα, θα δημιουργηθεί μια πρότυπη αποκεντρωμένη εφαρμογή ιστού (DApp) που θα παίρνει τα δεδομένα από τους παραπάνω φορείς, θα τα επεξεργάζεται, ώστε να μετατραπούν σε συμβατή και αναγνώσιμη μορφή και θα τα αποστέλλει ταυτόχρονα τόσο στην βάση δεδομένων της πλατφόρμας τους istSOS, προκειμένου να μπορούν μετέπειτα να αξιοποιηθούν σε μία εκτενή γκάμα εφαρμογών που παρέχει ήδη η υπηρεσία για δεδομένα τέτοιου σκοπού, όσο και στην αλυσίδα του blockchain που θα δημιουργηθεί αποκλειστικά για αυτήν την διπλωματική. Η προτεινόμενη μεθοδολογία παρουσιάζεται αναλυτικά στα παρακάτω διάγραμμα ροής και ανάπτυξης, ώστε να γίνει πιο κατανοητός ο τρόπος δομής και τα βήματα που

ακολουθήθηκαν στην διπλωματική εργασία και που αναλύονται στο παρόν κεφάλαιο.



Εικόνα 3-1 Γενικό διάγραμμα ροής προτεινόμενης μεθοδολογίας



Εικόνα 3-2 Γενικό διάγραμμα ανάπτυξης προτεινόμενης μεθοδολογίας

### 3.1 Συλλογή και Προεπεξεργασία Δεδομένων

Στην υποενότητα αυτή περιγράφεται αναλυτικά ο τρόπος με τον οποίο συλλέχθηκαν τα μετεωρολογικά δεδομένα από τους φορείς, η διαδικασία προεπεξεργασίας τους και η αποστολή τους στην πλατφόρμα του istSOS.

#### 3.1.1 Συλλογή Δεδομένων

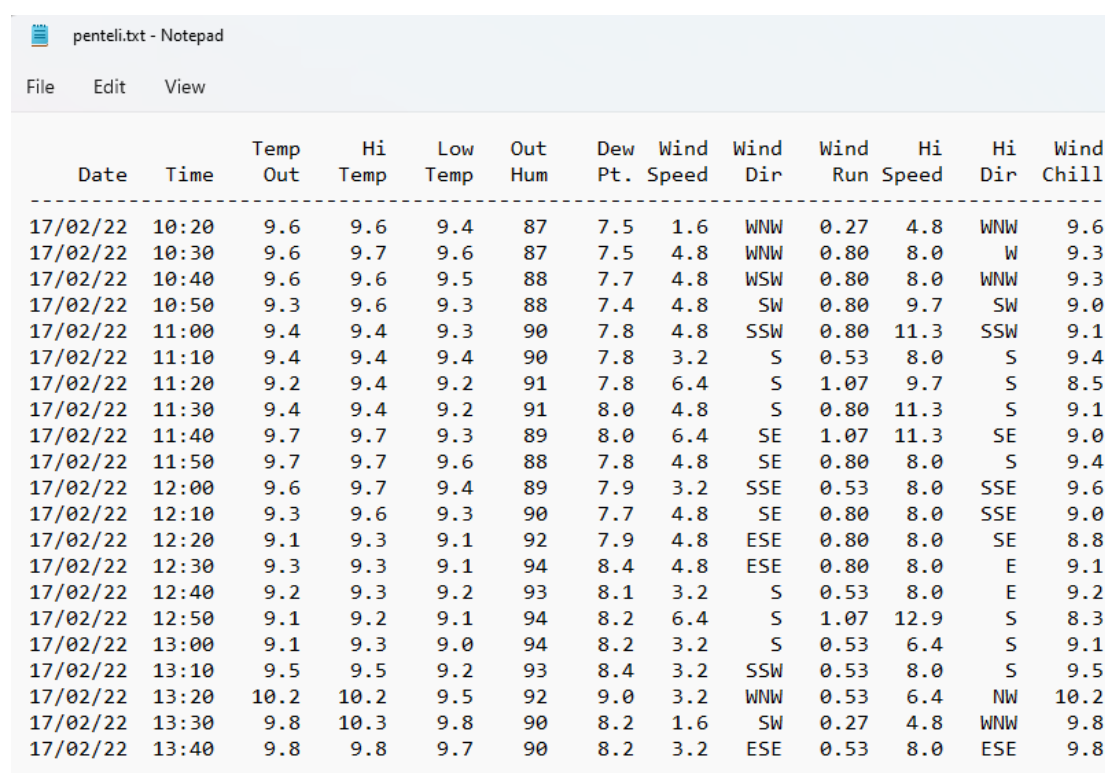
Για να μπορέσει να επιτευχθεί η κατανόηση των αναγκών και η διευθέτηση των απαραίτητων ζητημάτων που έπρεπε να ξεπεραστούν για να προχωρήσει η ανάπτυξη της εφαρμογής χρειάστηκε να βρεθούν οι αρμόδιοι φορείς που θα ήταν πρόθυμοι να προσφέρουν τα μετεωρολογικά τους δεδομένα. Αυτό επιτεύχθηκε με την συμβολή του Εθνικού Αστεροσκοπείου Αθηνών που μέσω του ιστοτόπου του meteo.gr προσφέρει ανοιχτά δεδομένα στην επιστημονική κοινότητα. Δόθηκαν, λοιπόν, δεδομένα από την καταγραφή των καιρικών φαινομένων σε μετεωρολογικούς σταθμούς πλησίον της Αττικής και ορισμένων Νησιωτικών περιοχών προκειμένου να υλοποιηθεί το πρώτο κομμάτι της εργασίας αυτής.

Οι μετρήσεις αυτές αν και στο γενικότερο σύνολο τους ήταν πλήρεις παρουσίαζαν κάποια κενά ανά διαστήματα που οφείλονταν σε αστοχίες του αισθητήρα, σε

αναβάθμιση του συστήματος μετρήσεων ή του δικτύου κλπ.. Επίσης, παρέχονταν σε μορφοποίηση απλού κειμένου (.txt) υπό την μορφή πίνακα, όπου οι στήλες αντιστοιχούσαν στα υπό μέτρηση μεγέθη και οι γραμμές στις μετρήσεις που πραγματοποιούσαν οι αισθητήρες, κάθε δέκα λεπτά. Η ανάλυση των παραπάνω δεδομένων ως μετρητικά μετεωρολογικά μεγέθη, στο πλαίσιο της εργασίας, απαιτούσε την συμμόρφωση τους με τα πρότυπα εισαγωγής δεδομένων της πλατφόρμας του istSOS και την κατά ακολουθία δημιουργία ενός νέου συνόλου δεδομένων χωρίς κενά στις μετρήσεις ή με κενά που να μπορούν να καλυφθούν, χωρίς, όμως, να αλλοιώνεται το περιεχόμενο της μέτρησης.

### 3.1.2 Προεπεξεργασία Δεδομένων

Τα αρχεία με τα δεδομένα που λήφθηκαν όπως προαναφέρθηκε ήταν σε μορφή κειμένου (.txt). Μια ενδεικτική απεικόνιση των αρχείων αυτών προβάλλεται στις παρακάτω εικόνες από τις οποίες φαίνονται μόνο οι πρώτες γραμμές και στήλες του πίνακα προς διευκόλυνση καθώς κάθε αρχείο μετεωρολογικών δεδομένων σταθμών περιείχε περίπου  $\approx 33$  διαφορετικούς αισθητήρες – στήλες,  $\approx 34$  (+1 για την χρονοσφραγίδα) – και περισσότερες από χίλιες γραμμές, διότι κάθε αρχείο περιέχει μετρήσεις για πάνω από τρεις μέρες στην σειρά.



Date	Time	Temp Out	Hi Temp	Low Temp	Out Hum	Dew Pt.	Wind Speed	Wind Dir	Wind Run	Hi Speed	Hi Dir	Wind Chill
17/02/22	10:20	9.6	9.6	9.4	87	7.5	1.6	WNW	0.27	4.8	WNW	9.6
17/02/22	10:30	9.6	9.7	9.6	87	7.5	4.8	WNW	0.80	8.0	W	9.3
17/02/22	10:40	9.6	9.6	9.5	88	7.7	4.8	WSW	0.80	8.0	WNW	9.3
17/02/22	10:50	9.3	9.6	9.3	88	7.4	4.8	SW	0.80	9.7	SW	9.0
17/02/22	11:00	9.4	9.4	9.3	90	7.8	4.8	SSW	0.80	11.3	SSW	9.1
17/02/22	11:10	9.4	9.4	9.4	90	7.8	3.2	S	0.53	8.0	S	9.4
17/02/22	11:20	9.2	9.4	9.2	91	7.8	6.4	S	1.07	9.7	S	8.5
17/02/22	11:30	9.4	9.4	9.2	91	8.0	4.8	S	0.80	11.3	S	9.1
17/02/22	11:40	9.7	9.7	9.3	89	8.0	6.4	SE	1.07	11.3	SE	9.0
17/02/22	11:50	9.7	9.7	9.6	88	7.8	4.8	SE	0.80	8.0	S	9.4
17/02/22	12:00	9.6	9.7	9.4	89	7.9	3.2	SSE	0.53	8.0	SSE	9.6
17/02/22	12:10	9.3	9.6	9.3	90	7.7	4.8	SE	0.80	8.0	SSE	9.0
17/02/22	12:20	9.1	9.3	9.1	92	7.9	4.8	ESE	0.80	8.0	SE	8.8
17/02/22	12:30	9.3	9.3	9.1	94	8.4	4.8	ESE	0.80	8.0	E	9.1
17/02/22	12:40	9.2	9.3	9.2	93	8.1	3.2	S	0.53	8.0	E	9.2
17/02/22	12:50	9.1	9.2	9.1	94	8.2	6.4	S	1.07	12.9	S	8.3
17/02/22	13:00	9.1	9.3	9.0	94	8.2	3.2	S	0.53	6.4	S	9.1
17/02/22	13:10	9.5	9.5	9.2	93	8.4	3.2	SSW	0.53	8.0	S	9.5
17/02/22	13:20	10.2	10.2	9.5	92	9.0	3.2	WNW	0.53	6.4	NW	10.2
17/02/22	13:30	9.8	10.3	9.8	90	8.2	1.6	SW	0.27	4.8	WNW	9.8
17/02/22	13:40	9.8	9.8	9.7	90	8.2	3.2	ESE	0.53	8.0	ESE	9.8

Εικόνα 3-3 Τμήμα αρχείου μετεωρολογικών δεδομένων για τον σταθμό της Πεντέλης

harokopio-athens.txt - Notepad														
File Edit View														
Date	Time	Temp Out	Hi Temp	Low Temp	Out Hum	Dew Pt.	Wind Speed	Wind Dir	Wind Run	Hi Speed	Hi Dir	Wind Chill	Heat Index	
18/02/22	0:10	9.7	9.7	9.6	93	8.6	0.0	---	0.00	0.0	---	9.7	9.9	
18/02/22	0:20	9.6	9.6	9.4	93	8.5	0.0	---	0.00	0.0	---	9.6	9.8	
18/02/22	0:30	9.4	9.5	9.3	93	8.4	0.0	---	0.00	0.0	---	9.4	9.7	
18/02/22	0:40	9.3	9.4	9.3	93	8.3	0.0	---	0.00	0.0	---	9.3	9.6	
18/02/22	0:50	9.3	9.5	9.3	94	8.4	0.0	---	0.00	0.0	---	9.3	9.6	
18/02/22	1:00	9.6	9.8	9.4	94	8.6	0.0	WNW	0.00	4.8	WNW	9.6	9.8	
18/02/22	1:10	9.9	9.9	9.8	93	8.8	0.0	---	0.00	0.0	---	9.9	10.1	
18/02/22	1:20	9.9	10.0	9.9	93	8.9	0.0	---	0.00	0.0	---	9.9	10.2	
18/02/22	1:30	9.9	9.9	9.7	93	8.8	0.0	---	0.00	0.0	---	9.9	10.1	
18/02/22	1:40	9.7	9.7	9.6	93	8.6	0.0	---	0.00	0.0	---	9.7	9.9	
18/02/22	1:50	9.5	9.6	9.3	93	8.4	0.0	---	0.00	0.0	---	9.5	9.7	
18/02/22	2:00	9.3	9.4	9.3	93	8.3	0.0	---	0.00	0.0	---	9.3	9.6	
18/02/22	2:10	9.3	9.3	9.2	93	8.2	0.0	---	0.00	0.0	---	9.3	9.5	
18/02/22	2:20	9.2	9.3	9.2	93	8.1	0.0	---	0.00	0.0	---	9.2	9.4	
18/02/22	2:30	9.3	9.4	9.2	93	8.2	0.0	---	0.00	0.0	---	9.3	9.5	
18/02/22	2:40	9.2	9.3	9.2	93	8.1	0.0	---	0.00	0.0	---	9.2	9.4	
18/02/22	2:50	9.1	9.2	9.1	93	8.0	0.0	---	0.00	0.0	---	9.1	9.3	
18/02/22	3:00	8.9	9.1	8.9	93	7.9	0.0	---	0.00	0.0	---	8.9	9.1	

Εικόνα 3-4 Τμήμα αρχείου μετεωρολογικών δεδομένων για τον σταθμό στο Χαροκόπειο Πανεπιστήμιο

Στις παραπάνω εικόνες παρουσιάζονται μερικά από τα δεδομένα που υπάρχουν μέσα στο αρχείο μετρητικών δεδομένων για τον σταθμό του Εθνικού Αστεροσκοπείου Αθηνών στην Πεντέλη (εικόνα Εικόνα 3-3 Τμήμα αρχείου μετεωρολογικών δεδομένων για τον σταθμό της Πεντέλης) και στο Χαροκόπειο Πανεπιστήμιο (εικόνα Εικόνα 3-4 Τμήμα αρχείου μετεωρολογικών δεδομένων για τον σταθμό στο Χαροκόπειο Πανεπιστήμιο).

Σε πρώτη φάση θα πρέπει να ελεγχθεί αν υπάρχουν αστοχίες όσων αφορά τα δεδομένα που αναφέρονται στο αρχείο. Τέτοιες αστοχίες είναι:

- Άγνωστες τιμές (---)
- Τιμές εκτός του εύρους μέτρησης των οργάνων και
- Συγχώνευση μετρήσεων κατά την μεταφορά τους στο αρχείο καταγραφής

### Άγνωστες Τιμές

Όπως φαίνεται και από τα παραπάνω στιγμιότυπα, οι άγνωστες τιμές δεν είναι κάτι που συναντάται σπάνια σε μετρήσεις τέτοιου είδους. Για να μπορέσει να αντιμετωπιστεί αυτό το πρόβλημα, θα πρέπει να βρεθούν όλες οι κενές προσθήκες – στην παρούσα μορφοποίηση η αδυναμία εύρεσης μέτρησης αναπαρίσταται με τρεις διαδοχικές παύλες – και να αντικατασταθούν με τον τριψήφιο δεκαδικό αρνητικό αριθμό -999.9 που είναι συνδεδεμένος στην λειτουργικότητα του istSOS ως το σημείο αναφοράς για απουσία τιμής.

Κάθε αισθητήρας στους σταθμούς του δικτύου έχει μια συγκεκριμένη αντοχή ήδη προκαθορισμένη εργοστασιακά. Προκειμένου η εφαρμογή να διασφαλίσει την ακεραιότητα των πραγματικών δεδομένων οφείλει να πραγματοποιεί ελέγχους και για την περίπτωση που έχει εισχωρήσει στις μετρήσεις ακούσια κάποια εσφαλμένη τιμή. Για να γίνει αυτό χρειάζεται να γίνουν γνωστά τα όρια μέτρησης των αισθητήρων, δηλαδή η μέγιστη και η ελάχιστη τιμή που μπορούν να καταγράψουν και τις οποίες διαθέτουν οι συγκεκριμένοι σταθμοί του φορέα. Τα δεδομένα αυτά δόθηκαν και ομαδοποιήθηκαν σε μία μορφή λεξικού για να μπορεί ο κώδικας να ανατρέχει σε αυτό κάθε φορά που θα χρειαστεί να κάνει έναν νέο έλεγχο. Πέραν των άνω και κάτω ορίων των μετρήσεων στο λεξικό συμπεριλαμβάνονται και δύο λογικές τιμές για τις περιπτώσεις στις οποίες δεν υπάρχει ένα από τα δύο όρια.

```
meteo:
  wind_tx: ["null", true, true, 1, 8]
  wind_samp: ["null", true, true, 0, -2800]
  arc_int: ["min", true, true, 1, 120]
  iss_recept: ["%", true, true, 0, 100]
  temp_out: ["°C", true, true, -40, 65]
  low_temp: ["°C", true, true, -40, 65]
  hi_temp: ["°C", true, true, -40, 65]
  out_hum: ["%", true, true, 0, 100]
  dew_pt: ["°C", true, true, -76, 54]
  wind_speed: ["km/h", true, true, 0, 320]
  wind_dir: ["null", true, true, 0, 15]
  wind_run: ["km", true, false, 0, 0]
  hi_speed: ["km/h", true, true, 0, 320]
  hi_dir: ["null", true, true, 0, 15]
  wind_chill: ["°C", true, true, -79, 57]
  heat_index: ["°C", true, true, -40, 74]
  thw_index: ["°C", true, true, -79, 74]
  thsw_index: ["°C", true, true, -68, 74]
  bar: ["hPa", true, true, 540, 1100]
  rain: ["mm", true, true, 0, 6553]
  rain_rate: ["mm/h", true, true, 0, 762]
  solar_rad: ["W/m2", true, true, 0, 1800]
  hi_solar_rad: ["W/m2", true, true, 0, 1800]
  solar_energy: ["Ly", true, true, 0, 2000]
  uv_index: ["null", true, true, 0, 16]
  hi_uv: ["null", true, true, 0, 16]
  uv_dose: ["MEDs", true, true, 0, 199]
  road_temp: ["°C", true, true, -40, 65]
  soil_temp: ["°C", true, true, -40, 65]
  soil_moist: ["cb", true, true, 0, 200]
  leaf_wet: ["null", true, true, 0, 15]
  et: ["mm", true, true, 0, 2000]
  heat_d-d: ["null", true, false, 0, 0]
  cool_d-d: ["null", true, false, 0, 0]
  in_temp: ["°C", true, true, 0, 60]
  in_hum: ["%", true, true, 0, 100]
  in_dew: ["°C", true, true, -50, 60]
  in_heat: ["°C", true, true, 0, 537]
  in_air_density: ["null", false, false, 0, 0]
  in_emc: ["null", false, false, 0, 0]
```

Εικόνα 3-5 Λεξικό εύρεσης ορίων για κάθε διαφορετικό αισθητήρα

## Συγχώνευση Μετρήσεων

Σε κάποιες περιπτώσεις παρατηρείται πως στα δεδομένα από τις μετρήσεις δύο διαφορετικών αισθητήρων υπερκαλύπτεται το κενό που υπάρχει ανάμεσα τους και

προκύπτει μία ενιαία τιμή και για τους δύο. Η συγχώνευση αυτή φαίνεται στην παρακάτω εικόνα.

Wind Run	Hi Speed	Hi Dir	Wind Chill	Heat Index	THW Index	Bar	Rain	Rain Rate	Solar Rad.	Hi Solar Rad.	UV Index
1.62	27.4	SSW	12.9	13.6	10.0	1007.0	0.0	0.0	---	---	---
1.62	29.0	W	12.8	13.4	9.9	1006.7	0.0	0.0	---	---	---
1.62	16.1	SW	12.1	12.8	9.3	1006.8	0.41	280.0	---	---	---
0.80	16.1	SW	12.4	12.4	10.6	1006.9	0.8	35.6	---	---	---
1.62	25.7	SW	11.5	12.3	8.8	1007.0	0.0	0.8	---	---	---

Εικόνα 3-6 Παράδειγμα συγχώνευσης μετρήσεων

Όπως φαίνεται από το ανωτέρω παράδειγμα που παρατηρήθηκε στις μετρήσεις του σταθμού Ασή Γωνιά στην Κρήτη οι μετρήσεις του αισθητήρα της βροχής και της μέσης συχνότητας βροχής έχουν ενωθεί στέλνοντας λανθασμένο αποτέλεσμα. Για να μπορέσει να διορθωθεί αυτό αξιοποιήθηκε το λεξικό ορίων των μονάδων μέτρησης του σταθμού. Με τον τρόπο αυτό γίνεται πλέον σαφές πού πρέπει να γίνει η διχοτόμηση των δύο τιμών με βάση την μέγιστη και την ελάχιστη τιμή που μπορούν να πάρουν. Για τον λόγο αυτό στο συγκεκριμένο παράδειγμα η τιμή 0.41280.0 μετατράπηκε αυτοματοποιημένα στις τιμές 0.41 mm για την βροχή και 280.0 mm/h για την συχνότητα βροχής.

### 3.1.3 Συμβατότητα με το Πρότυπο του IstSOS

Όπως γίνεται αντιληπτό πέραν των κλασσικών μετρήσεων που προσδιορίζονται από αριθμητικές – ακέραιες και δεκαδικές – τιμές αλλά και των λογικών διορθώσεων που έγιναν στην παραπάνω υποενότητα, υπάρχουν επιπλέον αλλαγές που πρέπει να γίνουν προκειμένου τα δεδομένα που θα αποσταλούν τελικά στην πλατφόρμα του istSOS να είναι συμβατά με τα πρότυπα που ακολουθεί για να μπορέσει να τα διαβάσει και να τα αξιοποιήσει.

Σε δεύτερη φάση, λουπόν, μελετήθηκαν οι απαιτήσεις για τα δεδομένα από την πλατφόρμα του istSOS και σημειώθηκαν οι εξής απαραίτητες για την ομαλότητα αλλαγές που πρέπει να πραγματοποιηθούν:

- Ημερομηνία και ώρα (Χρονοσφραγίδα – Timestamp)
- Μη μετρήσιμα μεγέθη, όπως κατεύθυνση ανέμου, πιο συνήθης κατεύθυνση ανέμου κλπ. και
- Προσδιορισμός χαρακτηριστικών μετεωρολογικού σταθμού

#### Ημερομηνία και Ώρα

Η ημερομηνία και η ώρα που βρίσκονται στα αρχεία είναι υπό την μορφή DD/MM/YYYY, όπου D: ημέρα (date), M: μήνας (month) και Y: έτος (year) και hh:mm,

όπου h: ώρα (hour) και m: λεπτά (minutes) αντίστοιχα. Για να μπορέσει να αναγνωριστεί ορθά η χρονοσφραγίδα της κάθε μέτρησης, θα πρέπει να προσαρμοστεί στο πρότυπο εκφράσεων χρόνου ISO 8601 που έχει καθοριστεί ως ένα διεθνές πρότυπο έκφρασης ημερομηνίας και ώρας σε μορφή τύπου δεδομένων συμβολοσειράς. Σύμφωνα με το συγκεκριμένο πρότυπο πρέπει να εφαρμοστεί και η ζώνη ώρας (time zone) στην συμβολοσειρά η οποία για τα συγκεκριμένα δεδομένα αντιστοιχεί σε "+00:00". Ένα παράδειγμα μιας τέτοιας μετατροπής παρουσιάζεται παρακάτω στην πρώτη μέτρηση που παρέχεται από τον σταθμό Ασή Γωνιά.

"17/02/2022 10:20"

⇓

"2022-02-17T10:20:00+00:00"

Στο παραπάνω παράδειγμα αξίζει να τονιστεί πως το γράμμα T συμβολίζει το πέρας της ημερομηνίας και την αρχή δήλωσης της ώρας του συμβάντος. Γενικότερα αν και επιτρέπονται και άλλες παρόμοιες εκφράσεις, η παρούσα γίνεται αποδεκτή από την πλειονότητα των εφαρμογών παγκοσμίως και γενικεύεται ως εξής:

"YYYY-MM-DDThh:mm:ss+TZD"

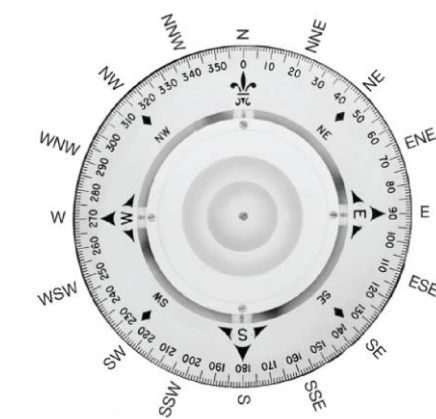
Όπου TZD η ζώνη ώρας που μπορεί να πάρει είτε θετικές είτε αρνητικές τιμές ανάλογα με το σημείο της μέτρησης και την απόκλιση αυτού με βάση το παγκόσμιο σύστημα χρόνου UTC (Coordinated Universal Time).

---

## Μη Μετρήσιμα Μεγέθη

Ένα πρόβλημα που εντοπίστηκε στα δεδομένα και έχριζε άμεσης προσαρμογής ήταν η διευθέτηση των μετρήσεων που δεν μπορούσαν να εκφραστούν εκ φύσεως σε αριθμητικές τιμές. Αυτό παρατηρήθηκε στα μετεωρολογικά δεδομένα που παρουσίαζαν μετρήσεις προσδιορισμού κατεύθυνσης. Η κατεύθυνση του ανέμου και η συχνότερη ροή ανέμου στις μετρήσεις αναπαρίστανται με λεκτικούς χαρακτήρες αντιπροσωπεύοντας συγκεκριμένες κατευθύνσεις του ορίζοντα. Οι λεκτικοί αυτοί συμβολισμοί δεν αποτελούσαν τυχαίες γραμματοσειρές αλλά συμβολικού περιεχομένου αρκτικόλεξα ως απόρροια όλων των δυνατών κατευθύνσεων που μπορούν να προσδιορίσουν προσανατολισμό κίνησης – στην παρούσα περίπτωση αέριων μαζών.





Εικόνα 3-7 Πυξίδα με τα σημεία του ορίζοντα

Όπως φαίνεται από την εικόνα παραπάνω, τα σημεία προσδιορισμού κατευθύνσεων είναι 16 και μπορούν να κωδικοποιηθούν στο δεκαδικό σύστημα με τους αριθμούς από το 0 έως το 15, όπως φαίνεται παρακάτω:

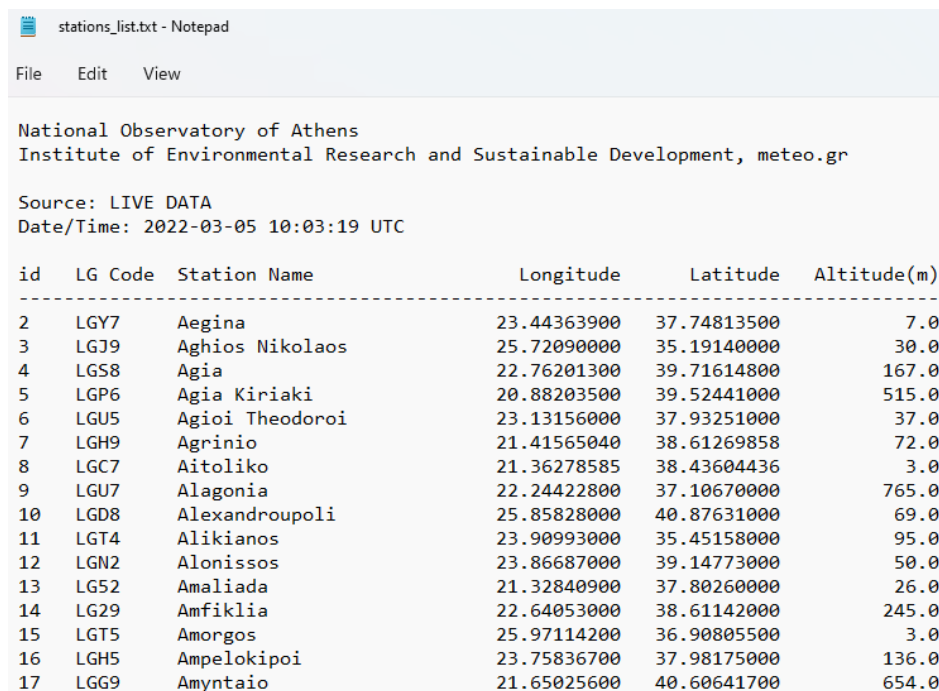
"N"	0	"S"	8
"NNE"	1	"SSW"	9
"NE"	2	"SW"	10
"ENE"	3	"WSW"	11
"E"	4	"W"	12
"ESE"	5	"WNW"	13
"SE"	6	"NW"	14
"SSE"	7	"NNW"	15

Όπου:

"N"	Βορράς ( North )
"E"	Ανατολή ( East )
"S"	Νότος ( South )
"W"	Δύση ( West )

Η κωδικοποίηση αυτή βρίσκεται μέσα στο λεξικό "wind\_dir" του κώδικα και χρησιμοποιήθηκε για να αντικατασταθούν όλοι οι λεκτικοί όροι προσδιορισμού κατεύθυνσης με αριθμητικό χαρακτήρα. Με τις μετατροπές αυτές η πλατφόρμα του istSOS θα μπορεί πλέον να διαβάσει κανονικά τις μετεωρολογικές μετρήσεις για τους συγκεκριμένους αισθητήρες, να τις αποθηκεύσει και μετέπειτα, να τις χρησιμοποιήσει για παρασκευή διαγραμμάτων, συγκριτικών μοντέλων και γενικότερη χρήση επιστημονικού ενδιαφέροντος.

Για να μπορέσει να φορτωθεί έναν καινούργιο σταθμός στο istSOS, απαιτείται από το σύστημα να εκχωρηθούν σε αυτό τα απαραίτητα προσδιοριστικά χαρακτηριστικά του σταθμού. Οι συντεταγμένες αυτού, το υψόμετρο και φυσικά, οι αισθητήρες που διαθέτει είναι μόνο κάποια από τα στοιχεία που πρέπει να αποσταλούν μέσω SOS αιτήματος στην πλατφόρμα για να γίνει η επίσημη καταγραφή του σταθμού στην βάση του istSOS. Με τον τρόπο αυτό, κάθε φορά που επιχειρεί να ανέβει ένα καινούργιο αρχείο μετεωρολογικών δεδομένων στο istSOS θα πρέπει πρώτα να ελέγχεται αν ο σταθμός αυτός υπάρχει ήδη στην βάση του istSOS, που έχει στηθεί για τον συγκεκριμένο φορέα, από κάποιο προηγούμενο αρχείο που πιθανώς έχει προστεθεί. Σε κάθε άλλη περίπτωση, πρέπει να προστεθεί ο καινούργιος, για το istSOS, σταθμός με όλα τα απαραίτητα χαρακτηριστικά που σημειώθηκαν πιο πάνω. Τα δεδομένα αυτά δόθηκαν από τον φορέα σε ένα αρχείο που να περιλαμβάνει όλους τους σταθμούς του δικτύου, απόσπασμα του οποίου ακολουθεί:



id	LG Code	Station Name	Longitude	Latitude	Altitude(m)
2	LGY7	Aegina	23.44363900	37.74813500	7.0
3	LGT9	Aghios Nikolaos	25.72090000	35.19140000	30.0
4	LGS8	Agia	22.76201300	39.71614800	167.0
5	LGP6	Agia Kiriaki	20.88203500	39.52441000	515.0
6	LGU5	Agioi Theodoroi	23.13156000	37.93251000	37.0
7	LGH9	Agrinio	21.41565040	38.61269858	72.0
8	LGC7	Aitoliko	21.36278585	38.43604436	3.0
9	LGU7	Alagonia	22.24422800	37.10670000	765.0
10	LGD8	Alexandroupoli	25.85828000	40.87631000	69.0
11	LGT4	Alikianos	23.90993000	35.45158000	95.0
12	LGN2	Alonissos	23.86687000	39.14773000	50.0
13	LG52	Amaliada	21.32840900	37.80260000	26.0
14	LG29	Amfiklia	22.64053000	38.61142000	245.0
15	LGT5	Amorgos	25.97114200	36.90805500	3.0
16	LGH5	Ampelokipoi	23.75836700	37.98175000	136.0
17	LGG9	Amyntaio	21.65025600	40.60641700	654.0

Εικόνα 3-8 Τμήμα πίνακα με τους σταθμούς του Εθνικού Αστεροσκοπείου Αθηνών

Κάθε αρχείο έχει ονομασία προσδιοριστική του ονόματος του σταθμού που αντιπροσωπεύει και της ακριβούς ημερομηνίας και ώρας της τελευταίας μέτρησης που περιλαμβάνει, για λόγους που θα αναφερθούν στην συνέχεια. Λ.χ. το αρχείο που περιλαμβάνει μετρήσεις για τον σταθμό Μενίδι – Νέα Οδός μέχρι την 17/02/2022 στις 14:20:00 ονομάζεται "MENIDI-NEAODOS\_20220217142000.txt". Με τον τρόπο αυτό, γνωρίζοντας το όνομα του σταθμού γίνεται αναζήτηση στον παραπάνω πίνακα σταθμών και γνωστοποιούνται τα χαρακτηριστικά του σταθμού. Οι αισθητήρες από την άλλη, που περιλαμβάνει κάθε σταθμός είναι ήδη γνωστοί από το περιεχόμενο του αρχείου με τις μετρήσεις. Ο τίτλος κάθε στήλης προσδιορίζει τον αισθητήρα της μέτρησης – εσωτερική, εξωτερική θερμοκρασία, βροχή, υγρασία κλπ. – επιτρέποντας

στο πρόγραμμα να συλλέξει όλους τους αισθητήρες ανά μετεωρολογικό σταθμό. Με τα παραπάνω δεδομένα συλλέγονται όλα τα απαραίτητα δεδομένα που απαιτούνται για την αποθήκευση ενός νέου σταθμού στην βάση του istSOS.

### 3.1.4 Μεταφόρτωση Δεδομένων στο Πρότυπο του IstSOS

Στο επόμενο βήμα, τα δεδομένα πρέπει να αποσταλούν, με κατάλληλη μορφοποίηση, με ένα SOS αίτημα στην βάση του istSOS. Τα δεδομένα θα πρέπει να μετατραπούν σε μορφή δεδομένων JSON, έτσι ώστε να μπορέσουν να τοποθετηθούν στο αίτημα. Κάθε γραμμή μετρήσεων αντιστοιχεί σε μία χρονοσφραγίδα, οπότε οι μετρήσεις της κάθε σειράς θα πρέπει να χωρίζονται μεταξύ τους με κόμμα, δηλαδή σε μορφή διαχωρισμού με κόμμα (CSV – Comma Separated Values), ώστε να μπουν όλες μαζί στην ίδια εγγραφή του JSON. Για παράδειγμα, αν οι μετρήσεις που αποστέλλονται είναι αυτές της εικόνας Εικόνα 3-3 τότε θα μετατραπούν ως εξής:

```
"17/02/22 10:20 9.6 9.6 9.4 87 9.4 87"
↓
"2022-02-17T10:20:00.000000+0000,9.6,9.6,9.4,87,9.4,87"
```

Τα δεδομένα χωρίζονται με κόμμα με την ίδια ακριβώς σειρά που ήταν προηγουμένως και ορίζουν την χρονοσφραγίδα τους με βάση το πρότυπο ISO 8601, όπως αναφέρθηκε προηγουμένως. Εκτός των μεγεθών αυτών, θα πρέπει να δεχθούν μια ελαφριά τροποποίηση και οι ονομασίες των αισθητήρων που υπάρχουν στην αρχή του αρχείου. Στην θέση της στήλης της ημερομηνίας θα πρέπει να μπει η συμβολοσειρά "urn:ogc:def:parameter:x-istsos:1.0:time:iso8601", με σκοπό να καθοριστεί και το πρότυπο που ακολουθείται στην δήλωση της ημερομηνίας, και πριν από κάθε όνομα αισθητήρα η συμβολοσειρά "urn:ogc:def:parameter:x-istsos:1.0:meteo:". Για παράδειγμα, αν οι μετρήσεις που αποστέλλονται είναι αυτές της εικόνας Εικόνα 3-3 τότε θα μετατραπούν ως εξής:

Date	Time	Temp Out	Hi Temp
↓			
"urn:ogc:def:parameter:x-istsos:1.0:time:iso8601, urn:ogc:def:parameter:x-istsos:1.0:meteo:temp:out, urn:ogc:def:parameter:x-istsos:1.0:meteo:hi:temp, ..."			

Για να μπορέσει το δίκτυο στην πλατφόρμα να επεξεργαστεί και να κατανοήσει τα δεδομένα θα πρέπει στο αίτημα που θα αποσταλεί να υπάρχει το πεδίο "field" που θα αναγράφει όλους τους αισθητήρες για τους οποίους επρόκειτο να μεταφορτωθούν μετρήσεις με την αντίστοιχη σειρά και ένα πεδίο "values" για τις μετρήσεις.

Το πεδίο "field" αποτελεί μία εγγραφή του αρχείου JSON που έχει ως τιμή έναν πίνακα. Ο πίνακας αυτός αποτελείται από τα μεγέθη των αισθητήρων – που αναφέρονται οι μετρήσεις του συγκεκριμένου αρχείου – ως ένα λεξικό με πρώτο όρο του το όνομα της στήλης, δεύτερο το όνομα του αισθητήρα, όπως αυτός μετατράπηκε παραπάνω και τρίτο την μονάδα μέτρησης του αισθητήρα. Οπότε το προκύπτων στιγμιότυπο του JSON για το πεδίο του "field" στο αρχείο που χρησιμοποιήθηκε στην εικόνα Εικόνα 3-3 ως παράδειγμα, είναι το ακόλουθο:

```
"elementCount": "33",
"field": [
  {
    "name": "Time",
    "definition": "urn:ogc:def:parameter:x-istsos:1.0:time:iso8601"
  },
  {
    "name": "temp:out",
    "definition": "urn:ogc:def:parameter:x-istsos:1.0:meteo:temp:out",
    "uom": "°C"
  },
  {
    "name": "hi:temp",
    "definition": "urn:ogc:def:parameter:x-istsos:1.0:meteo:hi:temp",
    "uom": "°C"
  },
  ...
  ...
]
```

Βλέπουμε λοιπόν ότι, καθώς θα σταλεί το SOS αίτημα στο istSOS, η πλατφόρμα θα αποκτήσει γνώση του ακριβή αριθμού αισθητήρων (στο παρόν περιέχονται μετρήσεις για 33 διαφορετικούς αισθητήρες, όπως φαίνεται από το πεδίο "elementCount") για τους οποίους περιέχει μετρήσεις το συγκεκριμένο αρχείο αλλά και με ποια σειρά παρουσιάζονται οι μετρήσεις σε αυτό.

Στο πεδίο περιέχονται όλες οι μετρήσεις του αρχείου ανά σειρά, δηλαδή ανά συγκεκριμένη ημερομηνία και ώρα. Η σειρά που παρουσιάζονται είναι ανάλογη της σειράς που αναγράφονται οι αισθητήρες στο πεδίο "field" προκειμένου να γίνεται κατανοητή η αντιστοιχία του μετρητικού δεδομένου με την μέτρηση που αντιπροσωπεύει. Σύμφωνα με την εικόνα Εικόνα 3-3 το πεδίο "values" διαμορφώνεται ως εξής:

```
"values": [
  [ "2022-02-17T10:20:00+00:00", 9.6, 9.6, 9.4, 87.0, 7.5, 1.6, 13.0, 0.27, 4.8, ... ],
```

```
[ "2022-02-17T10:30:00+00:00", 9.6, 9.7, 9.6, 87.0, 7.5, 4.8, 13.0, 0.8, 8.0, ... ],
[ "2022-02-17T10:40:00+00:00", 9.6, 9.6, 9.5, 88.0, 7.7, 4.8, 11.0, 0.8, 8.0, ... ],
[ "2022-02-17T10:50:00+00:00", 9.3, 9.6, 9.3, 88.0, 7.4, 4.8, 10.0, 0.8, 9.7, ... ],
...
...
]
```

Κάθε στοιχείο του πίνακα του "values" περιέχει έναν υποπίνακα με ακριβώς 33 μετρήσεις με πρώτη πάντα την χρονοσφραγίδα που έγινε η μέτρηση σε πρότυπο ISO 8601. Κατά αντιστοιχία με το πεδίο "field" μπορούμε να καταλάβουμε πως η δεύτερη θέση όλων των γραμμών αντιστοιχεί στον αισθητήρα "temp:out", η τρίτη στον αισθητήρα "hi:temp" και ούτω καθεξής.

Στο τελικό στάδιο για να αποσταλούν τα δεδομένα στο istSOS θα πρέπει πρώτα να προστεθεί – σε περίπτωση που δεν υπάρχει ήδη – στο istSOS ο σταθμός με τους αισθητήρες του μέσω του SOS αιτήματος RegisterSensor και έπειτα, τα μετρητικά μετεωρολογικά δεδομένα μέσω του SOS αιτήματος "InsertObservation". Αναλυτικότερα, η γλώσσα προγραμματισμού που χρησιμοποιήθηκε για την επεξεργασία των δεδομένων ήδη από την πρωτογενή μορφή τους είναι η Python. Μέσω της Python πραγματοποιούνται όλες οι παραπάνω μετατροπές που έχουν αναφερθεί στο κεφάλαιο αυτό. Στην περίπτωση που δεν υπάρχει ο σταθμός στην βάση δημιουργείται η μεταβλητή "stationI" (Station Information) που περιέχει όλες τις πληροφορίες του καινούργιου σταθμού μαζί με τους αισθητήρες που αναγράφονται στο αρχείο που θέλει να αποσταλεί σε μορφή JSON. Μαζί με κάθε αισθητήρα πρέπει να αναφερθεί συγκεκριμένα τόσο η μονάδα μέτρησης του όσο και τα όρια τιμών που μπορεί να δεχθεί αν αυτά έχουν δηλωθεί με σαφήνεια από τον φορέα. Αυτή η πληροφορία τοποθετείται σε έναν ειδικό δείκτη ποιότητας, αναγνωρίσιμο από το istSOS, το λεγόμενο "qualityIndex". Αν διαθέτουν και τα δύο όρια τιμών, ο δείκτης ποιότητας είναι "interval", μόνο το κάτω "min" και μόνο το πάνω "max". Για το παράδειγμα του σταθμού της Πεντέλης, που αναφέρθηκε πρωτύτερα, χρησιμοποιώντας δύο αντιπροσωπευτικούς αισθητήρες η μεταβλητή με το JSON αντικείμενο διαμορφώνεται ως εξής:

```
stationI = {
  "system_id": "PENTELI",
  "system": "PENTELI",
  "description": "weather station in PENTELI",
  "keywords": "weather, meteorological, meteo",
  "identification": [ {
    "name": "uniqueID",
    "definition": "urn:ogc:def:identifier:OGC:uniqueID",
    "value": "urn:ogc:def:procedure:x-istsos:1.0:PENTELI"
  } ],
  "classification": [ {
    "name": "System Type",
    "definition": "urn:ogc:def:classifier:x-istsos:1.0:systemType",
    "value": "insitu-fixed-point"
  }, {
    "name": "Sensor Type",
```

```

        "definition": "urn:ogc:def:classifier:x-istsos:1.0:sensorType",
        "value": "Meteo weather station"
    } ],
    "characteristics": "",
    "location": {
        "type": "Feature",
        "geometry": {
            "type": "Point",
            "coordinates": [ 23.86470000, 38.04720000, 495.0 ]
        },
        "crs": {
            "type": "name",
            "properties": { "name": "4326" }
        },
        "properties": {
            "name": "PENTELI"
        }
    },
    "inputs": [],
    "output" = [ {
        "name": "Time",
        "definition": "urn:ogc:def:parameter:x-istsos:1.0:time:iso8601",
        "uom": "iso8601",
        "description": "",
        "constraint": {}
    }, {
        "name": "temp:out",
        "definition": "urn:ogc:def:parameter:x-istsos:1.0:meteo:temp:out",
        "uom": "°C",
        "description": "",
        "constraint": {
            "role": "urn:ogc:def:classifiers:x-istsos:1.0:qualityIndex:check:reasonable",
            "interval": [ -40, 65 ] }
    }, {
        ...
    }, {
        "name": "wind:run",
        "definition": "urn:ogc:def:parameter:x-istsos:1.0:meteo:wind:run",
        "uom": "km",
        "description": "",
        "constraint": {
            "role": "urn:ogc:def:classifiers:x-istsos:1.0:qualityIndex:check:reasonable",
            "min": 0 }
    }, ... ]
}

```

Όπως φαίνεται στο JSON αντικείμενο τα σημαντικότερα πεδία αυτού, από τα οποία προσδιορίζονται με σαφήνεια όλες οι πληροφορίες που χρειάζεται το istSOS προκειμένου να δημιουργήσει έναν καινούργιο σταθμό στην βάση του είναι τα εξής:

- "system"  
Το όνομα του σταθμού που θα προστεθεί το οποίο ανάγεται από τον τίτλο του αρχείου το οποίο παραχωρήθηκε από τον φορέα.
- "value"

Το πλήρες όνομα του νέου σταθμού με την προσαύξηση που απαιτείται στην αρχή του, με σκοπό την κατανόηση αυτού από την πλατφόρμα.

- "geometry"  
Τα δεδομένα χωροταξικού προσδιορισμού του σταθμού, έτσι όπως έχουν καταγραφεί στο αρχείο μετεωρολογικών σταθμών του φορέα που του ανήκει. Τοποθετούνται στο υποπεδίο του με τίτλο "coordinates" και με την μορφή πίνακα.
- "output"  
Είναι όλοι οι αισθητήρες του σταθμού που είναι διαθέσιμοι και που μελλοντικά θα προστεθούν μετρήσεις για αυτούς από το αρχείο του σταθμού. Πρόκειται για έναν πίνακα που αποτελείται από JSON μικρότερα αντικείμενα, καθένα προσδιοριστικό του αισθητήρα που περιγράφει. Κάθε τέτοιο αντικείμενο πρέπει να περιλαμβάνει το όνομα του αισθητήρα στο πεδίο "name", τον ορισμό αυτού στο "definition" με την απαραίτητη προσθήκη που απαιτεί το istSOS, την μονάδα μέτρησης του στο "uom" και τα όρια των μετρήσεων που μπορεί να καταγράψει στο "constraint".

Μετάπειτα της προσθήκης του σταθμού στην περίπτωση που δεν υπήρχε προηγουμένως, ακολουθεί η μεταφόρτωση των μετεωρολογικών δεδομένων τα οποία αποθηκεύονται όλα μαζί σε μία μεταβλητή του κώδικα – data – που αποτελεί και αυτή αντικείμενο υπό μορφή JSON. Το τελικό περιεχόμενό της, μετά και την εισαγωγή των πεδίων "elementCount", "field" και "values", που περιέχουν όλες της μετεωρολογικές καταγραφές του αρχείου που ο χρήστης θέλει να αποθηκεύσει στην πλατφόρμα, για το παράδειγμα του σταθμού της Πεντέλης, φαίνεται παρακάτω:

```
"data" = {  
  "ObservationCollection": {  
    "description": "temporary offering to hold self-registered procedures/sensors waiting for  
      service administration acceptance",  
    "name": "temporary",  
    "member": [  
      {  
        "name": "PENTELI",  
        "samplingTime": {  
          "beginPosition": "2022-02-17T10:20:00+00:00",  
          "endPosition": "2022-02-17T13:40:00+00:00",  
          "duration": "PT3H20M"  
        },  
        "procedure": "urn:ogc:def:procedure:x-istsos:1.0:PENTELI",  
        "observedProperty": {  
          "CompositePhenomenon": {  
            "id": "comp_1",  
            "dimension": "33",  
            "name": "timeSeriesOfObservations"  
          },  
          "component": [  
            "urn:ogc:def:parameter:x-istsos:1.0:time:iso8601",  
            "urn:ogc:def:parameter:x-istsos:1.0:meteo:temp:out",  
            "urn:ogc:def:parameter:x-istsos:1.0:meteo:hi:temp",
```





που έπρεπε να μεταφορτωθεί στο istSOS. Συγκεκριμένα το καθένα περιγράφει τα κάτωθι:

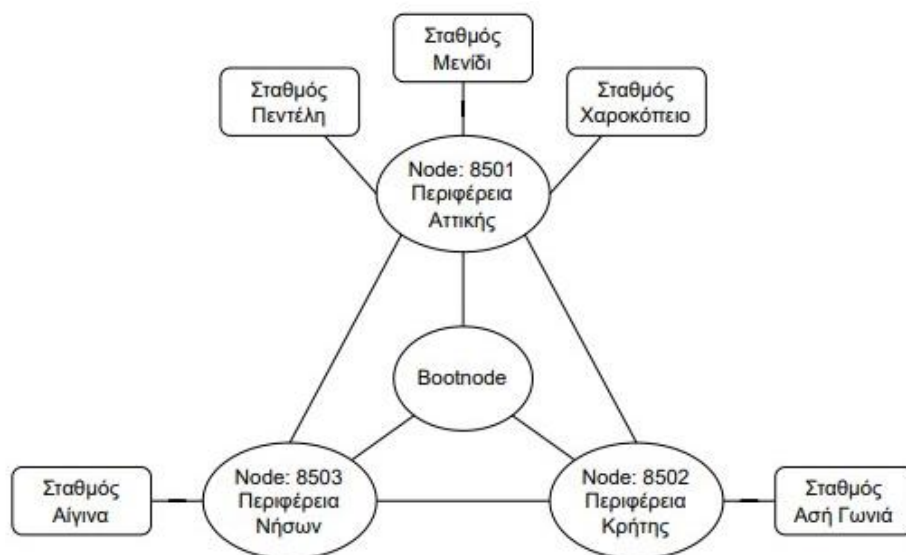
- "name"  
Το όνομα του σταθμού στον οποίο αναφέρονται οι μετρήσεις.
- "samplingTime"  
Περιλαμβάνει την ακριβή στιγμή που έγινε η πρώτη μέτρηση, την τελευταία καθώς και το διάστημα που διήρκεσαν μέσα από τα υποπεδία "beginPosition", "endPosition" και "duration", αντίστοιχα.
- "procedure"  
Το όνομα του σταθμού στον οποίο θα γίνει η προσθήκη των μετεωρολογικών δεδομένων σε συμβατή μορφή με το istSOS, δηλαδή ως εξής "urn:ogc:def:procedure:x-istsos:1.0:PENTELI" με την προσθήκη της απαραίτητης συμβολοσειράς πριν το όνομα.
- "dimension"  
Η διάσταση του σταθμού στην βάση του istSOS. Είναι ο αριθμός των αισθητήρων που υπάρχουν δηλωμένοι στην πλατφόρμα και όχι οι αισθητήρες για τους οποίους επρόκειτο να ανέβουν μετρήσεις.
- "component"  
Όπως έχει τονισθεί προηγουμένως, οι αισθητήρες που υπάρχουν αποθηκευμένοι στην βάση με το αναγνωριστικό της πλατφόρμας στην δήλωσή τους.
- "geom"  
Οι γεωγραφικές συντεταγμένες που μετεωρολογικού σταθμού που αναφερόμαστε με βάση το "procedure" συνοδευόμενες με το σύστημα συντεταγμένων που ακολουθούν (στην πλειονότητά τους οι σταθμοί χαρτογραφούνται με βάση το παγκόσμιο σύστημα συντεταγμένων EPSG:4326).
- "elementCount"  
Ο αριθμός των αισθητήρων για τους οποίους το αρχείο περιέχει μετρήσεις, με σκοπό να ανέβουν στην πλατφόρμα.
- "field"  
Οι αισθητήρες που αναφέρονται στις μετρήσεις του αρχείου αναλυτικά όπως ειπώθηκε προωτέρα.
- "values"  
Όλες οι μετρήσεις του αρχείου ταξινομημένες.

Μετά και την αποστολή των παραπάνω SOS αιτημάτων τα μετεωρολογικά δεδομένα των αρχείων που δόθηκαν από το Εθνικό Αστεροσκοπείο Αθηνών έχουν προσαρμοστεί στις απαιτήσεις και μεταφορτωθεί με επιτυχία στην πλατφόρμα του istSOS ολοκληρώνοντας το πρώτο στάδιο της παρούσας διπλωματικής. Ταυτόχρονα με την παραπάνω διαδικασία, τα αρχεία των μετρήσεων θα πρέπει να εκχωρηθούν και στην αλυσίδα του blockchain που επρόκειτο να δημιουργηθεί για να προσφέρει, χάρη στα υψηλά επίπεδα ασφαλείας που διακρίνεται, πιστοποιητικό γνησιότητας

στα μετεωρολογικά δεδομένα που μεταφορτώνονται στο istSOS και θα είναι προσβάσιμα μετέπειτα από την αποκεντρωμένη εφαρμογή ιστού.

### 3.2 Υλοποίηση Blockchain

Εν παραλλήλω της επιτυχημένης εισαγωγής των δεδομένων από τα αρχεία του φορέα πρέπει να ακολουθήσει και η ταυτόχρονη μεταφόρτωσή τους στην αλυσίδα του blockchain. Μία κίνηση που θα επιτρέψει στην εφαρμογή μελλοντικά να πραγματοποιήσει συγκρίσεις δεδομένων, με σκοπό να διαπιστώσει αν είναι γνήσια ή έχουν υποστεί κάποια μη εξουσιοδοτημένη επεξεργασία από τρίτους. Πρώτο βήμα της όλης διαδικασίας αποτελεί η κατασκευή του blockchain και έπειτα, η μεταφόρτωση των δεδομένων σε αυτό με βάση κάποια παραδοχή που θα γίνει από τον χρήστη σε συνεργασία με τον φορέα. Μετά και το πέρας της εισαγωγής των σταθμών και των αισθητήρων που τους αποτελούν στο blockchain, το δίκτυο του blockchain θα αναπαρίσταται ως ακολούθως:



Εικόνα 3-9 Σχηματική απεικόνιση δικτύου blockchain

#### 3.2.1 Κατασκευή Κύριας Δομής Blockchain

Ένα blockchain που λειτουργεί τοπικά σε έναν υπολογιστή είναι γνωστό ως ιδιωτικό (private) blockchain και είναι εκείνο που θα αξιοποιηθεί στην παρούσα διπλωματική. Με βάση αυτό πρόκειται να κατασκευαστεί ένα blockchain που θα αξιοποιεί όλες τις λειτουργίες ενός δημόσιου (public) blockchain χωρίς, όμως, να κάνει χρήση κανονικής ισοτιμίας κρυπτονομισμάτων, καθώς κάνοντας λόγο για ιδιωτικό blockchain το αντίτιμο που θα απαιτείται με κάθε νέα συναλλαγή που επρόκειτο να υλοποιηθεί σε αυτό θα είναι εικονικό. Ειδικότερα, θα κατασκευαστεί ένα ιδιωτικό Ethereum blockchain, ένα blockchain δηλαδή που προσφέρεται από την πλατφόρμα κρυπτονομισμάτων Ethereum – μιας πλατφόρμας που έχει αναπτύξει το

δικό της κρυπτονόμισμα, το "ether", για τις συναλλαγές της – το οποίο είναι ανοιχτό στο ευρύ κοινό για ανάπτυξη εφαρμογών τέτοιου είδους και η κατασκευή του οποίου είναι γνωστή και ελεύθερη. Για να μπορέσει να κατασκευαστεί το ιδιωτικό Ethereum blockchain, απαιτείται να εγκατασταθούν τα ακόλουθα [27]:

- Η γλώσσα προγραμματισμού Go της Google.  
Πρόκειται για μία γλώσσα προγραμματισμού ανοιχτού κώδικα ανεπτυγμένη από την Google, με σκοπό την αύξηση της παραγωγικότητας των προγραμματιστών. Διαθέτει πολλά εργαλεία και υλοποιήσεις, όπως η Geth που αναφέρεται στην συνέχεια. Για την εγκατάστασή της αρκεί να φορτωθεί η πιο πρόσφατη έκδοσή της και να προστεθούν τα μονοπάτια (path) που οδηγούν στον φάκελό της, προκειμένου ο υπολογιστής να γνωρίζει που βρίσκεται.
- Η υλοποίηση σε Go μιας εφαρμογής πελάτη Ethereum γνωστή ως Geth (Go-Ethereum).  
Η Geth είναι από τις βασικότερες βιβλιοθήκες που χρησιμοποιούνται από το terminal, με σκοπό να υλοποιηθούν διάφορες διεργασίες που στηρίζονται στο πρότυπο του Ethereum. Αποτελεί υλοποίηση της γλώσσας Go της Google για αυτό και απαιτείται η πρότερη εγκατάστασή της. Η υλοποίηση της Go-Ethereum είναι μία από τις τρεις πρωταρχικές υλοποιήσεις του πρωτοκόλλου του Ethereum. Αποτελεί το πιο γνωστό λογισμικό πελάτη (software client) με το οποίο μπορούν να δημιουργηθούν και να τεθούν σε λειτουργία οι κόμβοι σε ένα Ethereum δίκτυο [28]. Με την λειτουργία των κόμβων επιτρέπεται στους χρήστες να εκτελούν συναλλαγές στο blockchain και να αλληλοεπιδρούν με τα έξυπνα συμβόλαια του δικτύου.
- Η τεχνολογία Truffle.  
Η Truffle είναι μία τεχνολογία ανάπτυξης εφαρμογών Ethereum που παρέχει δυνατότητες στον χρήστη, με σκοπό την πιο εύκολη διατήρηση της δομής και της ανάπτυξης (deployment) μίας τέτοιας εφαρμογής, καθώς κάνει πολύ πιο εύκολη την διενέργεια ελέγχων και δοκιμών υποβοηθώντας το deployment και το migration που απαιτείται σε ένα έξυπνο συμβόλαιο, όπως θα περιγραφεί στη συνέχεια.

Μετά την εγκατάσταση των παραπάνω βιβλιοθηκών που είναι αναγκαίες μπορεί να δημιουργηθεί το δίκτυο του blockchain. Η βασική δομή που πρέπει να ακολουθηθεί προκειμένου στο τελικό αποτέλεσμα να υπάρχει ένα πλήρως λειτουργικό blockchain είναι η ακόλουθη:

- Δημιουργία κόμβων
- Συγγραφή Genesis αρχείου
- Σύνδεση κόμβων στο δίκτυο
- Δημιουργία Bootnode
- Εκκίνηση blockchain α. bootnode
- Εκκίνηση blockchain β. κόμβοι

Αυτά είναι τα έξη βασικά βήματα που πρέπει να εκτελεστούν για να μπορέσει να δημιουργηθεί και να λειτουργήσει ένα δίκτυο blockchain με τα χαρακτηριστικά που θα του αποδώσει το έξυπνο συμβόλαιο.

Συγκεκριμένα, η συλλογιστική πορεία που ακολουθήθηκε στην παρούσα εφαρμογή μπορεί να συνοψιστεί ως εξής. Πρώτα, δημιουργούνται οι αρχικοί κόμβοι του blockchain και έπειτα δημιουργείται το genesis αρχείο, ώστε να μπορέσουν να στηριχθούν σε αυτό και να εκκινηθούν με βάση τις προδιαγραφές που περιγράφει. Στην συνέχεια, δημιουργείται το bootnode για τον πλήρη έλεγχο των διεργασιών μεταξύ των κόμβων και του δικτύου. Εκκινούνται όλα τα συστήματα, δηλαδή οι κόμβοι και το bootnode και ξεκινάει η συγγραφή του έξυπνου συμβολαίου που θα περιλαμβάνει όλες τις απαραίτητες συναρτήσεις που χρειάζονται, ώστε να μπορεί ένας εξωτερικός χρήστης να έχει πρόσβαση στα μετεωρολογικά και μη δεδομένα που περιέχει καθώς και να προσθέτει νέα όταν γίνεται μια καινούργια εισαγωγή στο istSOS. Τέλος, το έξυπνο συμβόλαιο γίνεται compile και migrate μέσω του truffle project, ώστε να διαπιστωθεί αν όλα βαίνουν καλώς και φορτώνεται το έξυπνο συμβόλαιο στο blockchain, το οποίο πλέον είναι έτοιμο να πραγματοποιήσει συναλλαγές με τους χρήστες.

Προκειμένου να γίνει κατανοητός ο τρόπος δομής του φακέλου που φιλοξενεί το ιδιωτικό blockchain παρατίθεται παρακάτω η σχηματική απεικόνιση της τελικής του μορφής. Όλες οι εντολές που αναφέρονται ακολούθως εκτελούνται από το terminal που αντιστοιχεί στην ρίζα του δέντρου δηλαδή στο "bnet" (blockchain network).

```
└─ bnet
   ├── node1
   │   ├── geth
   │   └── keystore
   ├── node2
   │   ├── geth
   │   └── keystore
   ├── node3
   │   ├── geth
   │   └── keystore
   ├── boot.key
   └── genesis.json
```

Εικόνα 3-10 Σχηματική απεικόνιση σε μορφή δέντρου του περιεχομένου του φακέλου του blockchain

Για να ξεκινήσει η διαδικασία θα πρέπει να δημιουργηθούν οι φάκελοι (directories) bnet, node1, node2 και node3, όπου οι τελευταίοι τρεις είναι εμφωλευμένοι στον πρώτο και αντιστοιχούν στους κόμβους 1, 2 και 3 του δικτύου.

## Δημιουργία Κόμβων

Το πρώτο βήμα που θα πρέπει να γίνει για να ξεκινήσει η διαδικασία υλοποίησης του blockchain είναι η δημιουργία των πρώτων κόμβων, των κόμβων δηλαδή πάνω

στους οποίους θα στηθεί το δίκτυο και θα αποτελέσουν την αφετηρία του. Μετά την δημιουργία αυτών μπορούν να προστεθούν και άλλοι κατά την διάρκεια λειτουργίας του δικτύου χωρίς να υπάρξει κάποιο πρόβλημα συμβατότητας. Κάθε κόμβος αποκαλείται ακόμη και πορτοφόλι (wallet) και διατηρεί ένα ζευγάρι ιδιωτικού-δημόσιου κλειδιού (private – public key) που απαιτείται για την αλληλεπίδρασή του με το δίκτυο του blockchain. Με την χρήση αυτού του ζεύγους κλειδιών μπορεί να υπογράψει συναλλαγές και να προσδιορίζεται στο δίκτυο [27]. Για την συγκεκριμένη διπλωματική χρησιμοποιήθηκαν τρεις διαφορετικοί κόμβοι και όχι παραπάνω για να διατηρήσουν την πολυπλοκότητα του δικτύου και το επίπεδο δυσκολίας όσο γίνεται πιο χαμηλά. Για να δημιουργηθεί ένας νέος κόμβος, αρκεί να εκτελεστεί η παρακάτω εντολή.

```
geth --datadir nodeN/ account new
```

Όπου το "nodeN/" αντιστοιχεί στο directory του κόμβου N του οποίου πραγματοποιείται η δημιουργία. Το N παίρνει τις τιμές 1,2 και 3. Μόλις εκτελεστούν και οι τρεις εντολές, δημιουργούνται οι τρεις πρωταρχικοί κόμβοι του δικτύου και παράγεται η διεύθυνσή τους – σε δεκαεξαδική μορφή – η οποία υπάρχει αποθηκευμένη στον φάκελο που δημιουργήθηκε, "keystore", μαζί με άλλες πληροφορίες για τον κόμβο και τα χαρακτηριστικά του σε ένα αρχείο μορφής JSON με όνομα "UTC-datettime-address", όπου "datetime" η χρονοσφραγίδα που πραγματοποιήθηκε κατά την κλήση της παραπάνω εντολής και "address" η διεύθυνση του κόμβου. Επίσης, κατά την δημιουργία των κόμβων απαιτείται από το σύστημα η χρήση ενός κωδικού ασφαλείας για τον κάθε ένα χωριστά, ώστε να μπορεί στην συνέχεια να επικυρωθεί η ταυτότητα του κόμβου, που πρέπει να τεθεί σε λειτουργία, από το άτομο που αναλαμβάνει να τον εκκινήσει.

### Συγγραφή Genesis Αρχείου

Έπειτα της δημιουργίας των κόμβων, έπεται η συγγραφή του genesis αρχείου. Το αρχείο αυτό πρόκειται για ένα JSON αντικείμενο που περιέχει αναλυτικά πολλές σημαντικές πληροφορίες που πρέπει να προσδιοριστούν προτού δημιουργηθούν οι κόμβοι του δικτύου. Η δημιουργία και συμπερίληψή του στην διεύθυνση που επρόκειτο να αποθηκευτεί το blockchain στο παρόν υπολογιστικό σύστημα είναι απαραίτητη, καθώς σε αυτό στηρίζεται η δημιουργία των πρώτων κόμβων του δικτύου. Το αρχείο που χρησιμοποιήθηκε για το blockchain της εργασίας αυτής είναι το ακόλουθο:

```
{
  "config": {
    "chainId": 1515,
    "homesteadBlock": 0,
    "eip150Block": 0,
    "eip150Hash": "0x0000000000000000000000000000000000000000",
    "eip155Block": 0,
```

[illegible]

Όπως φαίνεται στον κώδικα που παρουσιάστηκε, το genesis αρχείο (genesis.json) περιέχει μια σειρά από πεδία που προσδιορίζουν ιδιότητες του blockchain, που επρόκειτο να δημιουργηθεί και να έχει την βάση του σε αυτό. Τα νούμερα που αναγράφονται στο αρχείο αυτό είναι σε δεκαεξαδική μορφή (0x) και αντιπροσωπεύουν μονάδες – εικονικού – κρυπτονομίσματος. Εκτενέστερα:

- "chainId"  
Είναι ο μοναδικός αριθμός που προσδιορίζει το συγκεκριμένο ιδιωτικό blockchain. Για την εφαρμογή αυτή ο αριθμός αυτός προσδιορίστηκε στον 1515.
- "gasLimit"  
Προσδιορίζει το ανώτατο όριο που μπορεί να κοστίσει μία συναλλαγή.
- "difficulty"

Αναφέρεται στην δυσκολία που θα απαιτείται κατά την διαδικασία του mining στην προσπάθεια επίλυσης ενός hash puzzle. Όσο μικρότερο είναι το "difficulty" τόσο πιο γρήγορα μπορεί να ολοκληρωθεί το mining σε κάποιο κόμβο, προκειμένου να παραχθούν blocks για το δίκτυο. Στα πλαίσια της ανάπτυξης της εφαρμογής αυτής για να ολοκληρώνεται πιο άμεσα η διαδικασία του mining, καθώς πρόκειται για ερευνητικής φύσης blockchain χρησιμοποιήθηκε μία αρκετά μικρή για το "difficulty".

- "extraData"  
Περιέχει όλες τις διευθύνσεις των κόμβων που συμμετέχουν στο ιδιωτικό blockchain ως μία τιμή, ενωμένες μεταξύ τους, όπως φαίνεται από την γραμμοσκιασμένη επιφάνεια στον κώδικα παραπάνω.
- "alloc"  
Είναι ένα αντικείμενο JSON που περιλαμβάνει πληροφορίες για όλους τους κόμβους που χρησιμοποιούνται στο ιδιωτικό blockchain κατά την στιγμή δημιουργίας του. Τους κόμβους δηλαδή εκείνους που δημιουργούνται πρώτοι με το genesis αρχείο και πάνω στους οποίους τρέχει το αρχικό δίκτυο. Στην παρούσα περίπτωση έχουν χρησιμοποιηθεί τρεις αρχικοί κόμβοι, όπως φαίνεται για τους οποίους έχουν αναφερθεί οι διευθύνσεις τους, καθώς και το αρχικό ποσό ισοτιμίας (gas) που θα έχουν μόλις ξεκινήσουν να «τρέχουν» στο δίκτυο. Χωρίς την αναφορά του πεδίου "alloc" στους κόμβους το σύστημα δεν θα ήταν δυνατόν να γνωρίζει ποιοι είναι οι κόμβοι που έχουν δημιουργηθεί για το συγκεκριμένο blockchain, με αποτέλεσμα να αδυνατεί να τους δώσει το απαραίτητο ποσό ισοτιμίας που θα χρειάζονται για να τεθεί σε λειτουργία το δίκτυο.

### Σύνδεση Κόμβων στο Δίκτυο

Μετά την δημιουργία του "genesis.json" και των κόμβων ακολουθεί η σύνδεσή τους στο δίκτυο, δηλαδή η αρχικοποίηση του κάθε κόμβου ξεχωριστά με το genesis αρχείο προκειμένου να μεταφερθεί όλη η πληροφορία του σε αυτούς. Η διαδικασία αρχικοποίησης και σύνδεσής τους γίνεται από την εντολή:

```
geth --datadir nodeN/ init ./genesis.json
```

Όπως και προηγουμένως, στην φάση της δημιουργίας των κόμβων, το "nodeN/" αντιστοιχεί στην διεύθυνση του φακέλου που βρίσκεται πλέον ο ήδη δημιουργημένος κόμβος N και το N παίρνει τις τιμές 1, 2 και 3. Το "./genesis.json" προσδιορίζει το genesis αρχείο που χρησιμοποιείται για να αρχικοποιηθεί ο συγκεκριμένος κόμβος.

Το bootnode πρόκειται στην ουσία για έναν γενικό κόμβο που βοηθάει τους υπόλοιπους κόμβους να εντοπίσουν ο ένας τον άλλο. Σε αντίθεση με τους υπόλοιπους κόμβους που έχουν δυναμικό IP, δηλαδή μπορούν να ενεργοποιούνται και να απενεργοποιούνται συνεχώς, το bootnode είναι έχει στατικό IP για αυτό και χρησιμοποιείται από το δίκτυο, προκειμένου να μπορεί άμεσα να δίνει πληροφορίες για το σύνολο των κόμβων του blockchain. Για να δημιουργηθεί και να αρχικοποιηθεί ο συγκεκριμένος κόμβος εκτελείται η παρακάτω εντολή:

```
bootnode -getkey boot.key
```

Με τον τρόπο αυτό δημιουργείται μια μοναδική τιμή γνωστή ως "enode" προσδιοριστική του bootnode και αποθηκεύεται στο "boot.key" αρχείο.

## Εκκίνηση Blockchain A. Bootnode

Μετά και την επιτυχή δημιουργία του bootnode, το blockchain που κατασκευάστηκε είναι έτοιμο και μπορεί να αρχίσει να λειτουργεί. Για να γίνει αυτό θα πρέπει να εκκινηθούν κατά σειρά το bootnode και έπειτα, οι κόμβοι. Αυτό γίνεται για να μπορεί να βλέπει όλους τους κόμβους που αρχίζουν να μπαίνουν στο δίκτυο, να παρέχει πληροφορίες στον χρήστη για όλες αυτές τις συνδέσεις και οι κόμβοι να μπορούν να κατευθυνθούν σε ένα "enode" που είναι λειτουργικό εκείνη την στιγμή. Για να γίνει η πρώτη εκκίνηση αρκεί η εκτέλεση της εντολής:

```
bootnode -nodekey boot.key -verbosity 9 -addr :30310
```

Όπου το "-nodekey" είναι η εντολή που προσδιορίζει σε ποιο αρχείο είναι αποθηκευμένο το "enode" του bootnode, το "-verbosity" που δίνει την δυνατότητα να εμφανίζονται τα μηνύματα μεταξύ των κόμβων βοηθώντας με αυτόν τον τρόπο τον χρήστη να καταλάβει αν οι κόμβοι συνδέθηκαν με επιτυχία στο bootnode και κατ' επέκταση στο δίκτυο και τις πληροφορίες που ανταλλάσσουν μεταξύ τους και τέλος το "-addr" που προσδιορίζει την διεύθυνση πάνω στην οποία θα «τρέχει» το bootnode και η οποία από εδώ και πέρα θα είναι δεσμευμένη για αυτό.

## Εκκίνηση Blockchain B. Κόμβοι

Ακολούθως θα πρέπει να εκκινηθούν οι κόμβοι του δικτύου ένας κάθε φορά με διάφορες προσδιοριστικές πληροφορίες στην εντολή που επρόκειτο να εκτελεστεί.



```

geth --datadir node1/
     --syncmode 'full'
     --port 30311
     --http
     --http.addr 'localhost'
     --http.port 8501
     --http.api 'personal, debug, eth, net, web3, txpool, miner, admin'
     --bootnodes
'enode://69a6177a3e000d3ee330db0b309171130e4320f48fb0c392ec16bddc57
0518ba6b15298d051789d419ac5de0b30a1e0338cffd6dbbeeaba16e1eb6d50ac7
3c9ed@127.0.0.1:0?discport=30310'
     --networkid 1515
     --miner.gasprice '0'
     --unlock '0xf3eA70434f661F2e29cDe3063F43A34Be9d1d4c8'
     --password 'node1' --mine

```

Οι προδιαγραφές αυτές αναφέρονται στην εκτέλεση του πρώτου κόμβου, που βρίσκεται αποθηκευμένος μέσα στον φάκελο node1. Το "--datadir" δείχνει τον φάκελο που είναι αποθηκευμένες οι πληροφορίες του κόμβου. Το "--syncmode 'full'" βοηθάει στην αποφυγή σφαλμάτων. Το "--port 30311" είναι η "enode" θύρα στην οποία συνδέεται ο κόμβος και πρέπει να είναι διαφορετική τόσο από την θύρα του bootnode όσο και από τις θύρες των υπολοίπων κόμβων, γιατί το δίκτυο «τρέχει» τοπικά δηλαδή στο "localhost" του υπολογιστή. Αν το σύστημα «έτρεχε» σε πραγματικό δίκτυο, τότε κάθε κόμβος θα αντιστοιχούσε σε διαφορετικό υπολογιστή και συνεπώς, θα συνδέονταν όλα στην ίδια πόρτα. Το "--http.port 8501" είναι η πόρτα που βλέπει ο HTTP εξυπηρετητής και είναι διαφορετική για κάθε κόμβο. Το "--http.api" προσδιορίζει όλα τα διαφορετικά API's που προσφέρονται για την λειτουργικότητα της διεπαφής HTTP. Το "--bootnodes" δηλώνει το "enode" του bootnode, ώστε να μπορέσει να συνδεθεί ο κόμβος μαζί του. Το "--networkid 1515" περιέχει τον μοναδικό αριθμό της αλυσίδας του blockchain, όπως αυτός έχει προσδιοριστεί ακριβώς στο genesis αρχείο. Το "--miner.gasprice '0'" το αντίτιμο που θα δίνεται για κάθε συναλλαγή και τα "--unlock", "--password" και "--mine" που διευκρινίζουν στον κόμβο που επρόκειτο να εκτελεστεί ποιον λογαριασμό πρέπει να ξεκλειδώσει, ποιος είναι ο κωδικός του και να ξεκινήσει να κάνει mining. Ο κωδικός για λόγους ευκολίας απλοποιήθηκε σε "node1", όπως φαίνεται παραπάνω για τον κόμβο 1 και με όμοιο τρόπο για τους υπόλοιπους.

Η παραπάνω εντολή εκτελείται και για τους κόμβους 2 και 3 με αντίστοιχη μορφή με μόνη διαφορά το "--unlock" και το "--password" που αναφέρονται στην διεύθυνση και στον κωδικό πρόσβασης σε αυτούς. Μετά την επιτυχή εκτέλεση και των τριών αυτών εντολών το δίκτυο θα πρέπει να λειτουργεί και αυτό θα φανεί στο terminal που «τρέχει» ο bootnode, καθώς θα τυπώνονται μηνύματα που θα δείχνουν την σύνδεση σε αυτόν, των θυρών 30311, 30312 και 30313.

### 3.2.2 Υλοποίηση Smart Contract

Μέχρι στιγμής έχει δημιουργηθεί το δίκτυο του blockchain με τρεις κόμβους και ένα bootnode που έχουν τεθεί σε λειτουργία. Παρόλα αυτά, το blockchain είναι αντικειμενικά κενό, αφού δεν έχει καμία πληροφορία μέσα του για τα μετεωρολογικά δεδομένα που πρέπει να αποθηκευτούν σε αυτό, καθώς επίσης δεν διαθέτει κανέναν προφανή, μέχρι τώρα, τρόπο για αποθήκευση και περισυλλογή των δεδομένων. Για να μπορέσει, λοιπόν, να γίνει αυτό εφικτό θα πρέπει να γραφεί ένα κατάλληλο έξυπνο συμβόλαιο που να περιέχει όλες τις απαραίτητες συναρτήσεις που απαιτούνται για την μεταφόρτωση των δεδομένων σε αυτό και όλες εκείνες που θα βοηθήσουν στην σωστή οργάνωση, δομή και αναζήτησή τους όταν ζητηθεί.

Η δομή του έξυπνου συμβολαίου εξαρτάται σε μεγάλο βαθμό από την διαμόρφωση του δικτύου ανάλογα με το υπάρχον πρόβλημα. Ειδικότερα, στην ανάπτυξη της παρούσας εργασίας που αφορά την διασφάλιση και την ακεραιότητα των μετεωρολογικών δεδομένων η συλλογιστική πορεία που ακολουθήθηκε στηρίχτηκε στην λογική του διαχωρισμού του δικτύου του blockchain σε γεωγραφικές περιφέρειες. Τα μετεωρολογικά δεδομένα είναι οργανωμένα από τον φορέα που στάλθηκαν ήδη σε αρχεία κατά σταθμό περισυλλογής. Δηλαδή όλη η πληροφορία που θα περαστεί στο istSOS και αντίστοιχα, στο blockchain για παράδειγμα για τον σταθμό του φορέα στην Πεντέλη, βρίσκονται στο αρχείο που έχει το ανάλογο όνομα. Περαιτέρω, το blockchain επρόκειτο να αντιστοιχίσει τους κόμβους του, δηλαδή τους τρεις που περιγράφηκαν πιο πάνω, σε γεωγραφικές περιοχές με το σκεπτικό ότι η προσθήκη και η επεξεργασία νέων μετεωρολογικών δεδομένων να μπορεί να πραγματοποιηθεί μόνο από σταθμούς που βρίσκονται στην ίδια γεωγραφική περιφέρεια. Οι τρεις αυτές γεωγραφικές περιοχές αποφασίστηκε να είναι η Αττική, η Κρήτη και τα νησιά λαμβάνοντας υπόψιν ότι τα δεδομένα που παραχωρήθηκαν εντάσσονται σε αυτές τις κατηγορίες. Η αντιπαραβολή των περιοχών αυτών με τις πόρτες του δικτύου που αντιστοιχούν σε κόμβους έγινε ως εξής:

Περιφέρειες	Πόρτες
Αττικής	8501
Κρήτης	8502
Νήσων	8503

Με τον τρόπο αυτό αν γίνει προσπάθεια από την γεωγραφική περιφέρεια της Κρήτης να προστεθούν μετεωρολογικά δεδομένα για τον σταθμό Χαροκόπειο στην Αττική, θα αποτύχει, καθώς ο σταθμός Χαροκόπειο ανήκει σε διαφορετική περιφέρεια επιτρέποντας μόνο μια επισκόπηση των δεδομένων αυτών. Από την άλλη πλευρά η γεωγραφική περιφέρεια του σταθμού Χαροκόπειο θα μπορεί να προσθέσει δεδομένα για τον σταθμό Μενίδι, αφού ο τελευταίος ανήκει και αυτός στην γεωγραφική περιφέρεια της Αττικής και κατά συνέπεια στον ίδιο κόμβο.

Έπειτα από την παραπάνω ανάλυση της συλλογιστικής στην οποία έθεσε τις βάσεις της η παρούσα εφαρμογή μπορεί να αναπτυχθεί και το έξυπνο συμβόλαιο, αφού πλέον είναι γνωστές οι απαιτήσεις του δικτύου και η μορφή που απαιτείται για την αρχειοθέτηση των δεδομένων σε αυτό. Για να μπορέσει όμως να τρέχει ένα

έξυπνο συμβόλαιο στο blockchain που έχει κατασκευαστεί, πρέπει να εγκατασταθεί και το Truffle Project. Το truffle είναι ένα περιβάλλον ανάπτυξης και δοκιμών για blockchains χρησιμοποιώντας εικονικές μηχανές Ethereum (Ethereum Virtual Machines – EVM). Με τον τρόπο αυτό μπορούν να δοκιμαστούν τα έξυπνα συμβόλαια που γράφονται και να διαχειριστεί το περιεχόμενο του δικτύου πολύ εύκολα και άμεσα.

### 3.2.2.1 Οργάνωση Περιεχομένου Smart Contract

Καθώς το μέγεθος των αρχείων που αποστέλλονται από τον φορέα είναι πολύ μεγάλο, – κάθε αρχείο μπορεί να περιέχει περισσότερες από χίλιες γραμμές με μετρήσεις – διαπιστώθηκε πως δεν θα μπορούσε να αποθηκεύεται αυτό καθαυτό το αρχείο στο blockchain. Ως εκ τούτου αποφασίστηκε να περνάει στο blockchain το hash value του αρχείου, το μοναδικό αυτό αποτύπωμα που παράγεται από μία μονόδρομη συνάρτηση ανάλογα με το περιεχόμενο του αρχείου, όπως έχει αναφερθεί και στο θεωρικό κομμάτι της εργασίας. Έτσι, το αρχείο με όλα τα δεδομένα του περνάει από την συνάρτηση και παράγεται το hash value. Αυτό έθεσε τις βάσεις για το έξυπνο συμβόλαιο που πρόκειται να αναπτυχθεί, καθώς υπάρχει πλέον μια ευδιάκριτη εικόνα του τι επρόκειτο να δέχεται. Με βάση τα παραπάνω το έξυπνο συμβόλαιο θα πρέπει να οργανώνει το περιεχόμενο του κάνοντας χρήση των παρακάτω δομών:

```
struct File {
    string _fullName;
    string _name;
    string _hash;
    string _firstMeasure;
    string _lastMeasure;
    address _address;
}

struct Days {
    string _fullName;
    File[] _days;
}

struct Global {
    string _name;
    Days[] _stations;
}
```

Στην πρώτη δομή "File" αποθηκεύονται όλα τα αρχεία που αποστέλλονται στο blockchain και κατά επέκταση περνάνε μέσα από το έξυπνο συμβόλαιο. Γίνεται κατανοητό πως τα αρχεία αυτά, όταν μεταφορτώνονται, πρέπει να αποστέλλουν το πλήρες όνομα του αρχείου, το όνομα του μετεωρολογικού σταθμού στον οποίο ανήκουν, το hash value που έχει δημιουργηθεί και την πρώτη και την τελευταία μέτρηση που είναι καταγεγραμμένες στο αρχείο αυτό. Το πεδίο "\_address" θα προσδιοριστεί στην συνέχεια. Τα αρχεία που είναι αποθηκευμένα σε δομές "File" ομαδοποιούνται, για να μπορέσουν να αντιπροσωπεύσουν μία ολόκληρη μέρα ενός σταθμού στην δομή "Day". Σε αυτήν αποθηκεύεται μόνο το πλήρες όνομα του σταθμού και ένας πίνακας δομών με όλα τα "Files" που αναφέρονται στον συγκεκριμένο σταθμό για μία συγκεκριμένη μέρα και ειδικότερα, αυτή που αναφέρεται στο πλήρες όνομα του αρχείου, διότι, όπως έχει αναφερθεί, τα αρχεία του φορέα ονομάζονται με το όνομα του σταθμού και την τελευταία ημερομηνία των μετρήσεων που διαθέτουν. Έπειτα, πραγματοποιείται και μία επιπλέον ομαδοποίηση, η δομή "Global" που περιλαμβάνει το απλό όνομα του σταθμού και

έναν διδιάστατο πίνακα δομών "Day", με σκοπό να βρίσκονται εκεί όλες οι μετρήσεις για έναν σταθμό ανεξαρτήτου ημερομηνίας. Η οργάνωση αυτή πραγματοποιείται με την τεχνική του mapping, μια τεχνική της γλώσσας Solidity που είναι γραμμένο το έξυπνο συμβόλαιο, που αντιστοιχίζει σε τιμές-κλειδιά και στην προκειμένη περίπτωση το πλήρες όνομα του αρχείου με την αντίστοιχη δομή "File".

```
mapping(string => mapping(string => File[])) public file;
```

### Συνάρτηση Εισαγωγής Δεδομένων

Πρώτη βασική συνάρτηση του έξυπνου συμβολαίου είναι εκείνη που προσθέτει τα αρχεία στο blockchain και η οποία αναφέρεται ως "addFile" και φαίνεται παρακάτω:

```
function addFile(
    string memory fullName,
    string memory name,
    string memory hashV,
    string memory firstMeasure,
    string memory lastMeasure
)public {
    File memory curfile;

    if (verifyRegion(name)) {
        curfile._fullName = fullName;
        curfile._name = name;
        curfile._firstMeasure = firstMeasure;
        curfile._lastMeasure = lastMeasure;
        curfile._hash = hashV;
        curfile._address = msg.sender;
        station_address[name] = msg.sender;

        if (info[name].length == 0) {
            stationNames.push(name); }

        if (file[name][fullName].length == 0) {
            info[name].push(fullName); }

        file[name][fullName].push(curfile);
    }
}
```

Όπως έχει αναφερθεί παραπάνω, οι πέντε τιμές που αποστέλλονται αντιστοιχίζονται με τις τοπικές μεταβλητές του έξυπνου συμβολαίου ενώ υπάρχει και μία έκρηξη, το "\_address" που δεν αποστέλλεται αλλά ανάγεται από την διεύθυνση του κόμβου που έκανε την προσθήκη του αρχείου αυτού. Η αιτία που το προκαλεί είναι η ανάγκη να προσδιοριστεί η γεωγραφική περιφέρεια που έστειλε το αρχείο. Για να προστεθεί

ένας καινούργιος σταθμός, αρκεί ο χρήστης να συνδεθεί από την κατάλληλη γεωγραφική περιφέρεια στην οποία επιθυμεί να τον προσθέσει και να βάλει το πρώτο του αρχείο. Μετά από αυτό μόνο χρήστες συνδεδεμένοι στην συγκεκριμένη περιφέρεια θα μπορούν να προσθέσουν περιεχόμενο για αυτόν τον σταθμό. Η συνάρτηση αυτή αποτελεί την μοναδική συνάρτηση του έξυπνου συμβολαίου για την εισαγωγή δεδομένων σε αυτό. Όλες οι υπόλοιπες που ακολουθούν εξυπηρετούν στην εξαγωγή και πιστοποίηση των δεδομένων, όταν ζητηθεί από τον χρήστη.

## Συναρτήσεις Εξαγωγής Δεδομένων

Αν ο χρήστης διαθέτει το πλήρες όνομα του αρχείου και κατά επέκταση το απλό όνομα του που μπορεί να εξαχθεί από αυτό, δηλαδή ένα αλφαριθμητικό της μορφής "aegina\_20220225100000.txt", τότε μπορεί να κάνει κλήση στις συναρτήσεις "getFile" και "getFileVersions". Με την πρώτη εντοπίζεται το συγκεκριμένο αρχείο του σταθμού και επιστρέφεται η πιο πρόσφατη έκδοση αυτού, ενώ με τη δεύτερη επιστρέφονται όλες οι διαφορετικές εκδόσεις για το συγκεκριμένο αρχείο, δηλαδή όλο το ιστορικό των συναλλαγών που πραγματοποιήθηκαν για τον συγκεκριμένο σταθμό στην ημερομηνία που αναγράφεται.

```
function getFile (
  string memory name,
  string memory fullName
) public view returns (File memory) {
  if (file[name][fullName].length != 0)
    return file[name][fullName][file[name][fullName].length - 1];
}

function getFileVersions (
  string memory name,
  string memory fullName
) public view returns (File[] memory) {
  return file[name][fullName];
}
```

Όπως φαίνεται στον κώδικα των συναρτήσεων που επισυνάφτηκε παραπάνω, γίνεται αναζήτηση στην δομή των "Files" και από εκεί εξάγεται το τελικό ζητούμενο. Στην συνέχεια, γνωστοποιώντας μόνο το όνομα του αρχείου, δηλαδή το όνομα του σταθμού που αναζητούμε, μέσω της συνάρτησης "getFileHistory" μπορεί να εντοπιστεί και να δοθεί στον χρήστη όλο το ιστορικό του σταθμού ανεξαρτήτου ημερομηνίας. Επιστρέφονται δηλαδή όλες οι εγγραφές του blockchain για τον συγκεκριμένο σταθμό από την στιγμή που προστέθηκε στην βάση μέχρι την στιγμή που καλείται η παρούσα συνάρτηση.

```
function getFileHistory(
  string memory name
) public view returns (Days[] memory) {
  uint lenInfo = info[name].length;
```

```

Days[] memory curD = new Days[](lenInfo);
for (uint i=0; i<lenInfo; i++) {
    File[] memory curFile = getFileVersions(name, info[name][i]);
    curD[i]._days = curFile;
    curD[i]._fullName = info[name][i];
}
return curD;
}

```

Ο τρόπος αναζήτησης στις συναρτήσεις αυτές λειτουργεί εμφωλευμένα, εφόσον η μία επικαλύπτει την άλλη προωθώντας την συνεργασία τους για ένα πιο γρήγορο και αποδοτικό αποτέλεσμα. Τελευταία συνάρτηση εξαγωγής δεδομένων παρουσιάζεται η "getAllFiles" ως γενικότερη όλων, καθώς δεν απαιτεί καμία παράμετρο και επιστρέφει όλες τις συναλλαγές που είναι αποθηκευμένες στο blockchain για όλους τους σταθμούς και όλες τις εκδόσεις τους οργανωμένα σε έναν πίνακα 3x3.

```

function getAllFiles (
) public view returns (Global[] memory) {
    uint lenSt = stationNames.length;
    Global[] memory curS = new Global[](lenSt);
    for (uint i=0; i<lenSt; i++) {
        Days[] memory curFile = getFileHistory(stationNames[i]);
        curS[i]._stations = curFile;
        curS[i]._name = stationNames[i];
    }
    return curS;
}

```

Όλες οι συναρτήσεις που αναφέρονται παραπάνω είναι συναρτήσεις που βοηθούν το δίκτυο να εξάγει δεδομένα στον χρήστη οργανωμένα χωρίς να απαιτεί εξειδικευμένες παραμέτρους αναζήτησης.

## Συναρτήσεις Πιστότητας

Πέραν των συναρτήσεων εισαγωγής και εξαγωγής δεδομένων στο blockchain το έξυπνο συμβόλαιο μπορεί να φιλοξενήσει και άλλου είδους βοηθητικές συναρτήσεις με σκοπό την πλήρη αξιοποίηση των λειτουργιών του δικτύου. Με βάση την λογική αυτή δημιουργήθηκαν δύο επιπρόσθετες συναρτήσεις, οι συναρτήσεις πιστότητας δεδομένων, δηλαδή συναρτήσεις που έχουν σκοπό να ελέγξουν αν τα στοιχεία που τους δίνει ο χρήστης είναι αληθή σύμφωνα πάντα με τα δεδομένα που κατέχουν. Πρώτη περίπτωση που απαιτεί πιστότητα είναι ο έλεγχος αν ένα αρχείο που διαθέτει ο χρήστης είναι γνήσιο αντίγραφο του αντίστοιχου αρχείου που υπάρχει στο blockchain. Για να επιτευχθεί αυτό ο χρήστης πρέπει να αποστείλει το πλήρες όνομα του αρχείου, το απλό όνομα και το hash value του αρχείου που επιθυμεί να ελέγξει την πιστότητα στην συνάρτηση "verifyHash" του έξυπνου συμβολαίου.

```

function verifyHash (
    string memory name,
    string memory fullName,
    string memory hashV
) public view returns (bool) {
    File memory selectedFile = getFile(name, fullName);
    if (keccak256(abi.encodePacked(selectedFile._hash)) == keccak256(abi.encodePacked(hashV))) {
        return true;
    } else {
        return false;
    }
}

```

Με τον τρόπο αυτό, το δίκτυο αναζητάει το αρχείο με το συγκεκριμένο όνομα που έλαβε και ελέγχει αν η πιο πρόσφατη έκδοση που διαθέτει από αυτό έχει το ίδιο hash value με αυτό που του έστειλε ο χρήστης. Σε κάθε άλλη περίπτωση το αρχείο του χρήστη έχει τροποποιηθεί οδηγώντας στις απαραίτητες ενέργειες που απαιτούνται από μέρος του προκειμένου να εντοπιστεί η παραποίηση. Μια ακόμη συνάρτηση πιστότητας που αναφέρεται στον κώδικα του έξυπνου συμβολαίου είναι εκείνη που ελέγχει αν ο χρήστης, που επιχειρεί να προσθέσει έναν συγκριμένο σταθμό, βρίσκεται στην σωστή γεωγραφική περιφέρεια. Σε περίπτωση που δεν έχει εξουσιοδότηση για τον σταθμό που προσπαθεί να προσθέσει, η συνάρτηση θα επιστρέψει λάθος εμποδίζοντας το σύστημα να κάνει αυτή την παράνομη συναλλαγή, ενώ σε κάθε άλλη περίπτωση επιστρέφει αλήθεια και δίνει στο δίκτυο το πράσινο φως για την προσθήκη του παρόντος αρχείου.

```

function verifyRegion (
    string memory name
) public view returns (bool) {
    if (info[name].length != 0) {
        if (station_address[name] == msg.sender) {
            return true;
        } else {
            return false;
        }
    } else {
        return true;
    }
}

```

Η πιστοποίηση βασίζεται στην σύγκριση του πεδίου "\_address", που, όπως διευκρινίστηκε παραπάνω, περιέχει τον κόμβο – την γεωγραφική περιφέρεια – που ανήκει ο συγκεκριμένος σταθμός, με την τιμή "msg.sender", μιας τιμής που παρέχεται από το Solidity και κάνει γνωστή την διεύθυνση του αποστολέα της συγκεκριμένης κλήσης.



### 3.2.3 Μεταφόρτωση Δεδομένων στην Αλυσίδα του Blockchain

Το blockchain που δημιουργήθηκε σύμφωνα με τις οδηγίες που περιγράφονται στην αρχή του κεφαλαίου αυτού – 3.2 – δε θα ήταν παρά κενός κώδικας χωρίς τα απαραίτητα δεδομένα τα οποία θα κληθεί να προστατεύσει. Σε συνέχεια, λοιπόν, της δημιουργίας της δομής του blockchain και του έξυπνου συμβολαίου μπορούν πλέον να υλοποιηθούν όλες οι απαραίτητες ενέργειες που χρειάζονται, προκειμένου να προστεθούν τα μετεωρολογικά δεδομένα και να είναι πλήρως λειτουργικό. Για να πραγματοποιηθεί η μεταφόρτωση αυτή από την πλευρά του χρήστη θα πρέπει σύμφωνα με την παραπάνω ανάλυση να δοθεί το αρχείο από τον φορέα, να προεπεξεργαστεί το περιεχόμενό του, να δημιουργηθεί το hash value με την κατάλληλη μονόδρομη συνάρτηση και τελικώς, να κληθεί η συνάρτηση "addFile" του έξυπνου συμβολαίου με παραμέτρους το πλήρες όνομα του αρχείου – αυτό δηλαδή που χρησιμοποιείται για την κατανόηση του και από την πλατφόρμα του istSOS –, το όνομα του σταθμού του αρχείου – που περιέχεται στον τίτλο του –, το hash value για ολόκληρο το περιεχόμενο του – έτσι όπως έχει προκύψει μετά την τροποποίηση του σε τυχών προσαρμογές και διορθώσεις, που πρέπει να γίνουν – και την πρώτη και τελευταία χρονοσφραγίδα που αναφέρονται σε αυτό – δηλαδή τις χρονοσφραγίδες από την πρώτη και τελευταία γραμμή του αρχείου αντίστοιχα –.

Μια χαρακτηριστική κλήση στο blockchain φαίνεται παρακάτω, όπου ένα αρχείο του φορέα κωδικοποιείται και αποστέλλεται στο blockchain ακριβώς πριν προστεθεί στο istSOS.

```
let instance = await web3Object.contracts.meteo.deployed();
let existFile = await instance.verifyHash.call(name, fullName, hash, { from: web3Object.account });

if (!existFile)
    await instance.addFile.sendTransaction ( fullName, name, hash, firstM, lastM, { from:
    web3Object.account });
else
    results.unshift('already');
```

Ο κώδικας, που παρουσιάζεται, είναι γραμμένος σε JavaScript και αποτελεί ένα απόσπασμα των όσων αναφέρθηκαν. Εστιάζοντας στις δύο συναρτήσεις του έξυπνου συμβολαίου, τις "verifyHash" και "addFile" γίνεται κατανοητό πως για να μπορέσει να προστεθεί οποιοδήποτε αρχείο στο blockchain γίνεται πρώτα έλεγχος αν το συγκεκριμένο hash υπάρχει ήδη μέσα. Στην περίπτωση που δεν υπάρχει, καλείται η συνάρτηση προσθήκης, διαφορετικά ο χρήστης ειδοποιείται ότι το αρχείο, που επιχειρεί να προσθέσει υπάρχει στην βάση και δεν παρατηρείται κάποια αλλοίωσή του.

## 3.3 Δημιουργία Αποκεντρωμένης Εφαρμογής

Η δημιουργία του δικτύου του blockchain και η μεταφόρτωση αρχείων τόσο σε αυτό όσο και στην πλατφόρμα του istSOS αναλύθηκαν και είναι πλέον σε λειτουργία.



Για να μπορέσει, όμως, ένας απλός χρήστης να έχει μια άμεση επικοινωνία με όλες αυτές τις διεργασίες που περιγράφηκαν πρέπει να δημιουργηθεί και η κατάλληλη διεπαφή χρήστη. Η διεπαφή αυτή ενσαρκώνεται με την δημιουργία μιας αποκεντρωμένης εφαρμογής ιστού, που θα επικοινωνεί άμεσα τόσο με το istSOS όσο και με το blockchain και θα είναι απλή στην χρήση και βολική για κάποιον που επιθυμεί να ελέγξει την γνησιότητα των μετεωρολογικών δεδομένων που έχει στην κατοχή του.

Για την συγκεκριμένη εφαρμογή χρησιμοποιήθηκαν διάφορες γλώσσες προγραμματισμού και εργαλεία, με σκοπό να γίνει όσο πιο φιλική και λειτουργική γίνεται προς τον χρήστη.

---

## Υλοποίηση Backend

Η γλώσσα προγραμματισμού JavaScript (JS) χρησιμοποιήθηκε στο μεγαλύτερο κομμάτι του backend της αποκεντρωμένης εφαρμογής. Με την JS υλοποιήθηκαν όλα τα λειτουργικά κομμάτια της εφαρμογής, όπως ανάδραση επιλογών, λειτουργικότητα σελίδων, επικοινωνία με τον εξυπηρετητή και άλλες δευτερεύουσες λειτουργίες. Επίσης, χρησιμοποιήθηκε και η γλώσσα προγραμματισμού Python για όλο το κομμάτι διαχείρισης των αρχείων που εισάγονται από τον χρήστη, όπως η προεπεξεργασία τους, η μορφοποίηση τους σε ενιαία μορφή – συμβατή με το istSOS και κατάλληλη για την οργάνωση τους μέσα στο δίκτυο του blockchain –. Ακόμη, έγινε χρήση της βιβλιοθήκης "Web3" για την συνεχή επικοινωνία της εφαρμογής με το blockchain και την ανταλλαγή πληροφοριών.

---

## Υλοποίηση Frontend

Για την μορφοποίηση της εφαρμογής όσον αφορά το frontend, το μέρος δηλαδή αυτής που έρχεται σε άμεση επαφή με τον χρήστη έγινε χρήση κατά κύριο λόγο της γλώσσας προγραμματισμού HTML, κυρίαρχης γλώσσας στην ανάπτυξη εφαρμογών τέτοιου τύπου. Συνεργάστηκε, όμως, για καλύτερη διαχείριση των ήδη υπαρχόντων λειτουργιών της με κώδικα γραμμένο σε Css και εισήχθησαν αντικείμενα και λειτουργικότητες από την αρκετά πλούσια σε περιεχόμενο βιβλιοθήκη του "Bootstrap". Η τελευταία πρόκειται για μία ανοιχτού κώδικα βιβλιοθήκη που παρέχει έτοιμα μορφοποιημένα αντικείμενα συμβατά με τα σημερινά πρότυπα βοηθώντας τον χρήστη να κατευθυνθεί και να αλληλεπιδράσει ομαλά με την εφαρμογή. Επίσης, για τα διαγράμματα που κάνουν την εμφάνισή τους υιοθετήθηκαν τα διαγράμματα που προσφέρει η "Chart.js" βιβλιοθήκη.

### 3.3.1 Δομή Αποκεντρωμένης Εφαρμογής

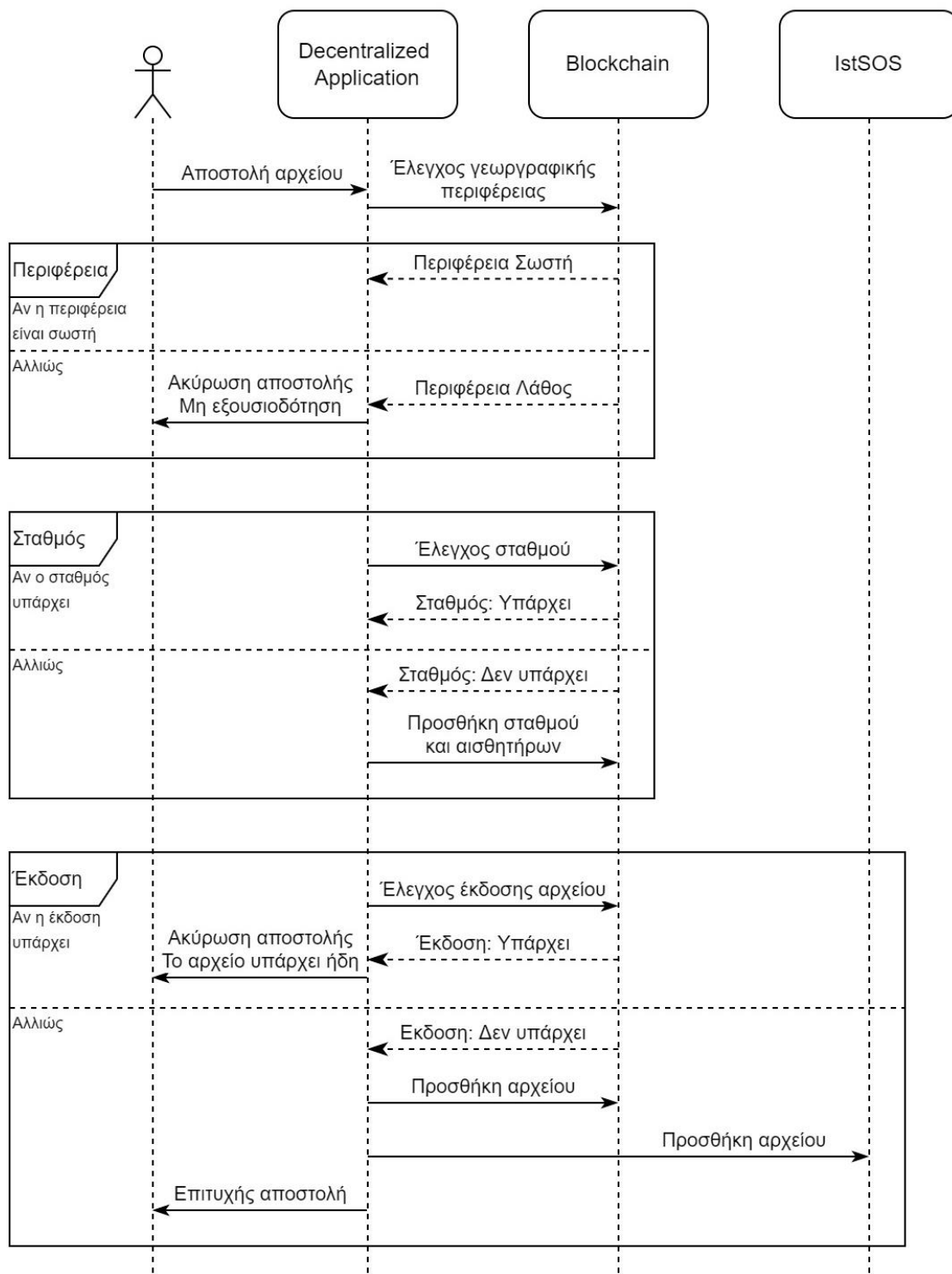
Η αποκεντρωμένη εφαρμογή που θα κατασκευαστεί θα πρέπει πρώτα να επιτρέπει στον χρήστη να συνδεθεί στην γεωγραφική περιφέρεια που επιθυμεί και σε δεύτερη φάση, να εκτελέσει κάποια ενέργεια μέσα σε αυτό. Αποφασίστηκε να μην

απαιτείται κωδικός για την σύνδεση στις γεωγραφικές περιφέρειες από τον χρήστη, αφού η παρούσα εφαρμογή αναπτύσσεται για διδακτικούς και όχι εμπορικούς λόγους.

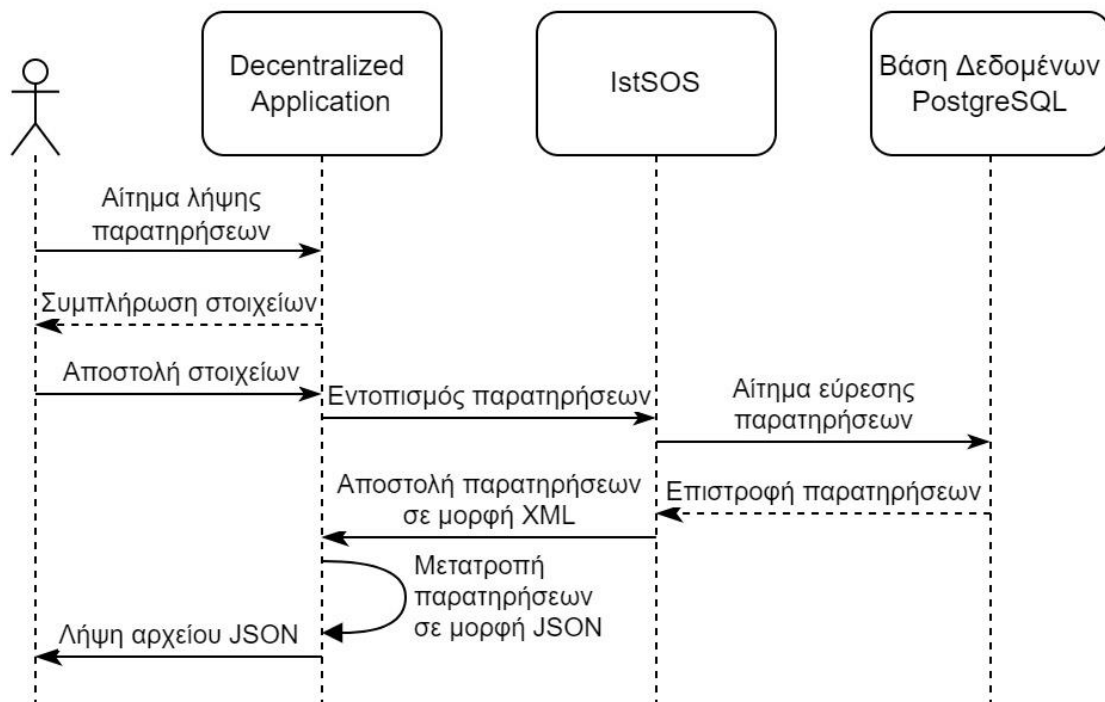
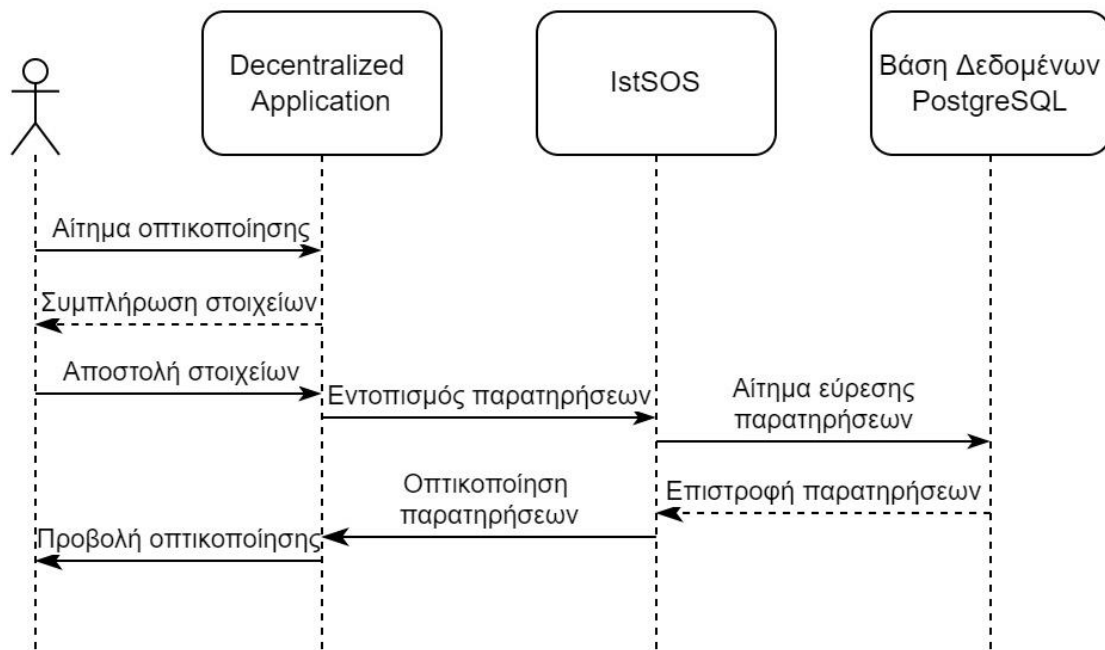
Μετά την είσοδο του χρήστη στην πλατφόρμα υπάρχουν τρεις κατευθυντήριες γραμμές που μπορεί να διαλέξει για να αλληλεπίδραση μαζί της. Επιγραμματικά αναφέρονται ως εξής:

- **Add to istSOS – Προσθήκη** αρχείου στο istSOS  
Γίνεται προσθήκη αρχείων στην πλατφόρμα του istSOS με τον ταυτόχρονο έλεγχο για ύπαρξη ή μη του αρχείου στο δίκτυο υπό την προϋπόθεση ότι ο χρήστης βρίσκεται στην σωστή γεωγραφική περιφέρεια για τα αρχεία που επιθυμεί να προσθέσει.
- **Visualize data – Αναπαράσταση** δεδομένων  
Τα δεδομένα από όλους τους σταθμούς της βάσης αναπαρίστανται σε αντίστοιχα διαγράμματα. Επίσης, δίνεται η δυνατότητα στον χρήστη για λήψη των δεδομένων οποιουδήποτε σταθμού σε μορφή JSON, με σκοπό την εύκολη μελλοντική τους χρήση και την διαλειτουργικότητα τους με άλλες εφαρμογές.
- Blockchain **history – Ιστορικό** blockchain  
Παρουσιάζονται όλα τα περιεχόμενα του blockchain οργανωμένα σε πίνακα ανά σταθμό και ανά ημερομηνία αναφοράς. Ακόμη, παρέχεται η δυνατότητα πιστότητας των δεδομένων που διαθέτει ο χρήστης με αυτά του δικτύου.

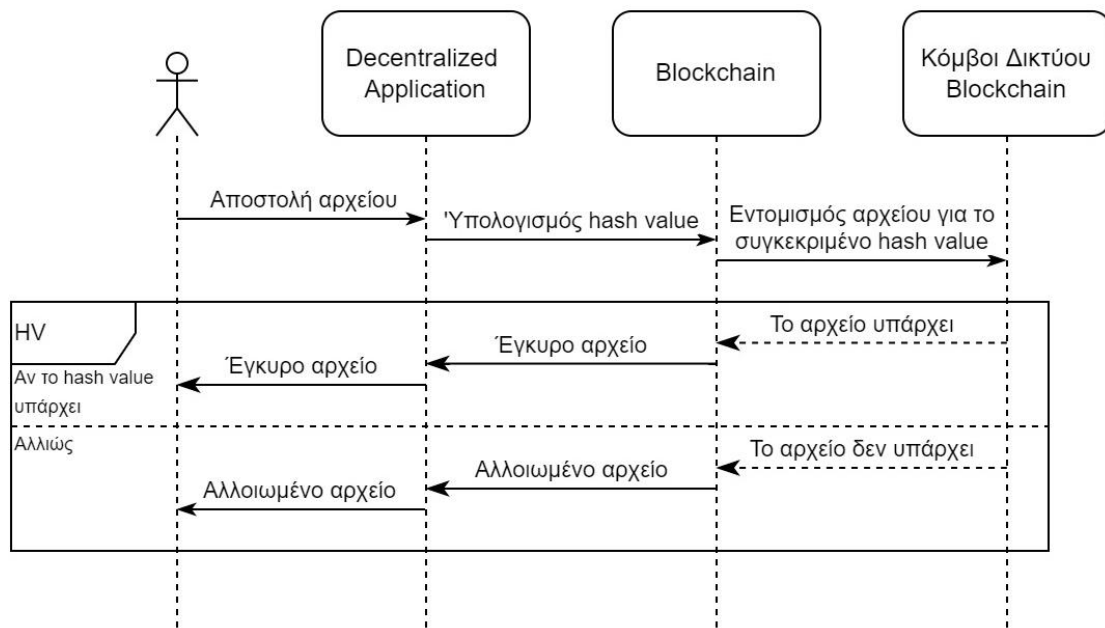
Οι τρεις παραπάνω κατευθύνσεις συντελούν στην βασική δομική οργάνωση την εφαρμογής ιστού που αναπτύχθηκε για την ομαλή συνύπαρξη μιας απαιτητικής τεχνολογίας διασφάλισης δεδομένων με την πλατφόρμα διαχείρισης γεωχωρικών μετρητικών δεδομένων. Στην συνέχεια ακολουθούν τρία διαγράμματα ακολουθίας, ένα για κάθε ξεχωριστή λειτουργία που υλοποιείται στα πλαίσια της αποκεντρωμένης εφαρμογής.



Εικόνα 3-11 Διάγραμμα ακολουθίας για την προσθήκη αρχείου στην πλατφόρμα του istSOS



Εικόνα 3-12 Διάγραμμα ακολουθίας για την αναπαράσταση και λήψη δεδομένων

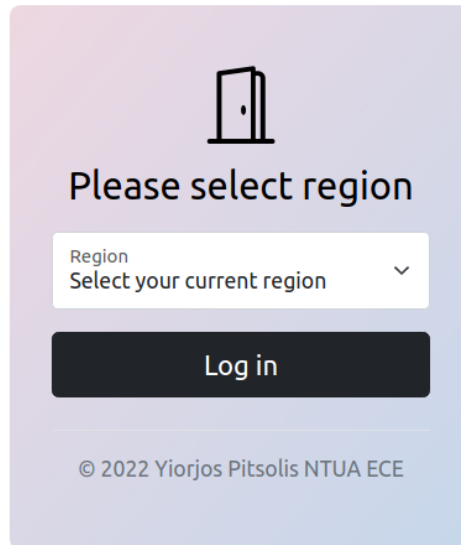


Εικόνα 3-13 Διάγραμμα ακολουθίας για το ιστορικό του blockchain

Γύρω από την δομή αυτή διαμορφώθηκε το λειτουργικό κομμάτι της αποκεντρωμένης εφαρμογής, όπως περιγράφεται ακολούθως.

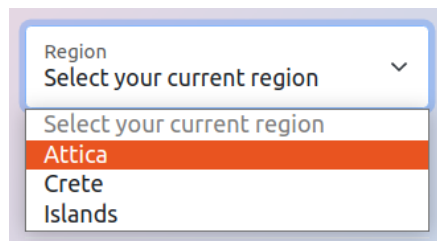
### 3.3.2 Διαμόρφωση και Λειτουργικότητα Αποκεντρωμένης Εφαρμογής

Όπως έχει ήδη ειπωθεί, ο χρήστης για να μπορέσει να αλληλεπιδράσει με οποιοδήποτε τρόπο μέσω της αποκεντρωμένης εφαρμογής θα πρέπει πρώτα, να συνδεθεί σε μία από τις διαθέσιμες γεωγραφικές περιφέρειες που είναι διαθέσιμες. Κάθε γεωγραφική περιφέρεια συνδέεται με έναν διαφορετικό κόμβο από αυτούς που διαθέτει το δίκτυο του blockchain και που έχουν σαφώς οριστεί από το σύστημα. Οι κόμβοι αυτοί δεν είναι τίποτα άλλο από συνδεδεμένους λογαριασμούς (accounts) του blockchain που, όταν πραγματοποιούν συναλλαγές με το blockchain, δεσμεύονται μέσω αυτής στο δίκτυο το οποίο αποθηκεύει στις δομές τους τον λογαριασμό – χρήση που έφερε σε πέρας την συγκεκριμένη συναλλαγή. Με τον τρόπο αυτό, ταυτίζονται οι σταθμοί με την γεωγραφική περιφέρεια που ανήκουν εμποδίζοντας οποιαδήποτε συναλλαγή προέρχεται από χρήστη συνδεδεμένο σε διαφορετική γεωγραφική περιφέρεια, δηλαδή διαφορετικό λογαριασμό. Η πρώτη σελίδα που βλέπει ο χρήστης, όταν θέσει σε λειτουργία το DApp, σύμφωνα με όσα έχουν ειπωθεί, είναι η σελίδα σύνδεσής του με το κατάλληλο account όπως φαίνεται παρακάτω.



Εικόνα 3-14 Στιγμιότυπο σύνδεσης του χρήστη στην αποκεντρωμένη εφαρμογή

Όπως φαίνεται, επιτρέπεται στον χρήστη να επιλέξει μεταξύ των διαθέσιμων γεωγραφικών περιοχών, που στην προκειμένη περίπτωση είναι τρεις: η Αττική, η Κρήτη και η περιφέρεια Νήσων.



Εικόνα 3-15 Στιγμιότυπο επιλογής κατάλληλης γεωγραφικής περιφέρειας

Μετά την επιλογή του κατάλληλου κόμβου, η αποκεντρωμένη εφαρμογή πραγματοποιεί κλήση στο blockchain επιδιώκοντας να συνδεθεί στο κατάλληλο λογαριασμό, εκείνο δηλαδή που έχει επιλέξει ο χρήστης. Αυτό επιτυγχάνεται με την βιβλιοθήκη "Web3" που χρησιμοποιήθηκε για δημιουργία, την αρχικοποίηση και την διασύνδεση της εφαρμογής με το blockchain. Αυτό πραγματοποιείται μέσα από την κλάση "w3" που γράφτηκε ώστε να παρέχει τις απαραίτητες συναρτήσεις για τις παραπάνω λειτουργίες.

```
class w3 {  
  
  constructor ( ) {  
    this.web3 = null;  
    this.web3Provider = null;  
    this.contracts = { };  
    this.account = null;  
  }  
}
```

```

async initWeb3 ( ) {
  let doc = yaml.load ( fs.readFileSync ( process.env.YAML, 'utf8' ) );
  if ( process.env.MODE == 'development' || typeof web3 === 'undefined' ) {
    this.web3Provider = new Web3.providers.HttpProvider ( doc.current.url );
  }
  else {
    this.web3Provider = web3.currentProvider;
  }
  this.web3 = new Web3( this.web3Provider );
}

async initAccount ( ) {
  this.web3.eth.getAccounts ( ( err, accounts ) => {
    if ( err ) console.log( err );
    if ( accounts == null )
      console.log( "Blockchain not running" );
    else
      this.account = accounts[0];
  } )
}

async initContractMeteo ( ) {
  const meteoArtifact = fs.readFileSync( __dirname + '/../build/contracts/Meteosc.json',
    { encoding: "utf-8" } );
  this.contracts.meteo = TruffleContract ( JSON.parse ( meteoArtifact ) );
  this.contracts.meteo.setProvider ( this.web3Provider );
}
}

```

Τα πιο σημαντικά κομμάτια του παραπάνω κώδικα είναι δύο. Το πρώτο αφορά την μεταβλητή "this.web3Provider" που συνδέει τον πάροχο του δικτύου με account που έχει επιλεγεί – το account αυτό έχει αποθηκευτεί σε ένα αρχείο Yaml, εφόσον η τιμή του μπορεί να αλλάζει συνεχώς κατά την αλληλεπίδραση του χρήστη με την εφαρμογή – και το δεύτερο αφορά την διασύνδεση του έξυπνου συμβολαίου με την αποκεντρωμένη εφαρμογή και το blockchain. Μέσα από την μεταβλητή "meteoArtifact" γίνεται φόρτωση σε αυτή του contract που έχει ειδικά κατασκευαστεί για τις απαιτήσεις της παρούσας εργασίας. Ακολούθως το έξυπνο συμβόλαιο διαβάζεται μέσω του Truffle λογισμικού και αποθηκεύεται στην μεταβλητή "this.contracts.meteo", στην οποία και γνωστοποιείται ο πάροχος, έτσι όπως έχει δηλωθεί.

Μετά και την επιλογή από τον χρήστη της περιφέρειας στην οποία θέλει να συνδεθεί μεταφέρεται στην κεντρική σελίδα της αποκεντρωμένης εφαρμογής. Σε αυτήν παρέχονται πληροφορίες για όλες τις λειτουργίες που μπορεί να εκπονήσει κάνοντας χρήση της εφαρμογής. Λόγου χάριν, άμα συνδεθεί στην περιφέρεια της Αττικής η κεντρική σελίδα, θα είναι η ακόλουθη.

## Add to istSOS

You can add new data for a station of this region, on the database on istSOS.

Add Files

## Visualize data

Select station and sensor and view data on a plot.

Make chart

## Blockchain history

All previous additions which have been applied on blockchain

Timeline

© 2022 Yiorjos Pitsolis NTUA ECE

Εικόνα 3-16 Στιγμιότυπο κύριας σελίδας της αποκεντρωμένης εφαρμογής

Μέσα από αυτήν την σελίδα ο χρήστης που βρίσκεται στην περιφέρεια της Αττικής, δηλαδή στον κόμβο 8501 του blockchain μπορεί επιλέξει μία από τις τρεις επιλογές που του δίνονται, προκειμένου να συνεχίσει την διάδρασή του στην εφαρμογή.

### Add to istSOS – Προσθήκη αρχείου




Επιλέγοντας το κουμπί "Add Files" από την πρώτη προβολή ή την ένδειξη "Add" από την επικεφαλίδα της σελίδας ο χρήστης μεταφέρεται στην σελίδα προσθήκης. Μέσα από αυτήν μπορεί να προσθέσει αρχεία στο istSOS και στο blockchain ταυτόχρονα. Επιλέγοντας τα αρχεία που επιθυμεί να προσθέσει, τα υποβάλει για μεταφόρτωση πατώντας το κουμπί "Submit" και μπορεί να παρακολουθήσει την πορεία του από την προβολή "Console Output" το οποίο προβάλλεται στην εικόνα που ακολουθεί.



**Connect to istSOS database**  

Ip chain.softlab.ntua.gr

Db blockchain

Station name	File name
ASIGONIA	asigonia_20220217153000.txt 
HAROKOPIO-ATHENS	harokopio-athens_20220218044000.txt 
PENTELI	penteli_20220217134000.txt 

Select files

Submit

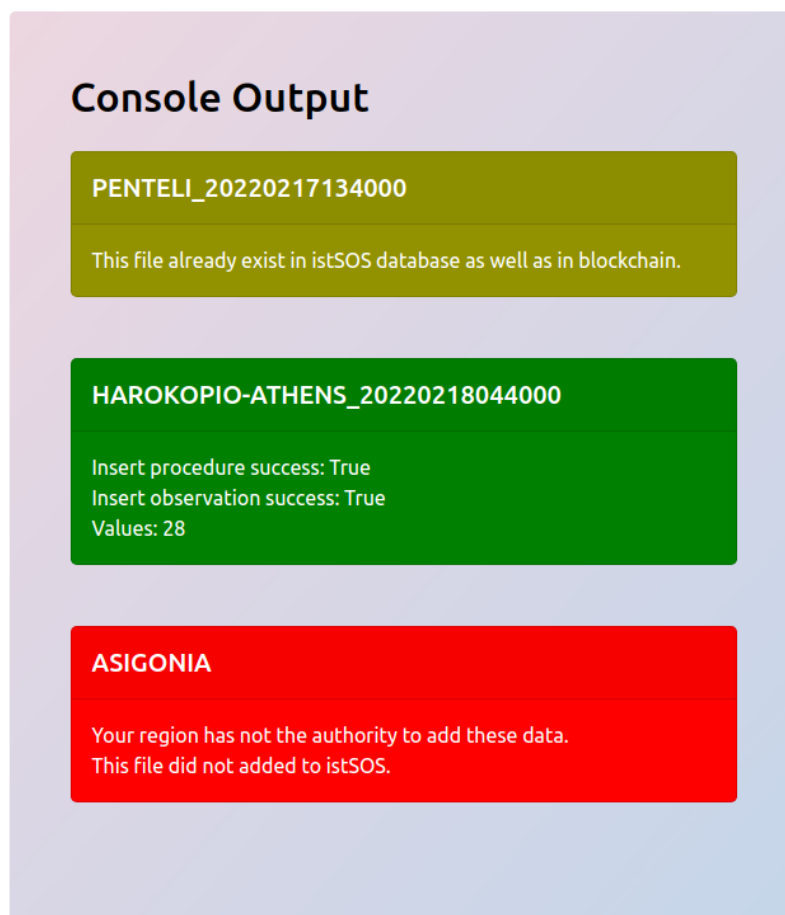
Εικόνα 3-17 Στιγμιότυπο προσθήκης αρχείων στο istSOS και στο blockchain

Στο πάνω μέρος της εικόνας αναγράφεται το "URL" του istSOS στην βάση του οποίου, όπως δηλώνεται από κάτω, επρόκειτο να προστεθούν τα επιλεγμένα αρχεία. Έχει ήδη αναφερθεί ότι, για να μπορέσει να γίνει μια επιτυχημένη μεταφόρτωση αρχείου, θα πρέπει να συναινούν τα εξής:

- Σύνδεση στην σωστή γεωγραφική περιφέρεια. Εκείνη δηλαδή στην οποία υπάγεται ο μετεωρολογικός σταθμός του οποίου το αρχείο με τις μετρήσεις επιχειρεί να προσθέσει ο χρήστης.
- Έλεγχος στο δίκτυο του blockchain, αν το συγκεκριμένο υπάρχει ήδη χωρίς κάποια τροποποίηση στο περιεχόμενό του.
- Προσθήκη στο δίκτυο του blockchain με κλήση του αντικειμένου της κλάσης "w3" που δημιουργήθηκε, όπως αναφέρθηκε παραπάνω και τέλος,
- Αποστολή αρχείου στο istSOS προς ενημέρωση της βάσης της πλατφόρμας.

Αν κάποιο από τα παραπάνω βήματα αποτύχει, η μεταφόρτωση αποτυγχάνει. Η ενημέρωση του χρήστη επιτυγχάνεται με τρία πιθανά μηνύματα που μπορεί να εμφανιστούν σε περίπτωση επιτυχίας ή μη.

- **Πράσινη ένδειξη**  
Το αρχείο έχει μεταφορτωθεί με επιτυχία τόσο στο blockchain όσο και στο istSOS. Μαζί με το μήνυμα της ορθής καταχώρησης περιλαμβάνονται και λοιπές πληροφορίες αναφορικά με το αν ο σταθμός προϋπήρχε στην βάση ή προστέθηκε πρώτη φορά μαζί με τους αισθητήρες, αν οι παρατηρήσεις στάλθηκαν επιτυχώς και πόσες γραμμές μετρήσεων – κάθε γραμμή περιλαμβάνει μία μέτρηση για κάθε αισθητήρα του σταθμού για μία συγκεκριμένη χρονοσφραγίδα – προστέθηκαν.
- **Κόκκινη ένδειξη**  
Η γεωγραφική περιφέρεια που βρίσκεται ο χρήστης δεν έχει εξουσιοδότηση για προσθήκη αρχείων του σταθμού στον οποίο αναφέρεται. Αποτέλεσμα αυτού είναι ακύρωση της συναλλαγής και η απαγόρευση του συστήματος να μεταφορτώσει το αρχείο αυτό τόσο στο blockchain όσο και στο istSOS.
- **Κίτρινη ένδειξη**  
Το παρόν αρχείο υπάρχει ήδη μέσα στην βάση στην ίδια ακριβώς έκδοση. Δεν έχει υποστεί καμία αλλαγή από τον χρήστη ο οποίος είναι αρμόδιος, οπότε η μεταφόρτωσή του δεν έχει κανένα νόημα και αναβάλλεται.



Εικόνα 3-18 Στιγμιότυπο μηνυμάτων συστήματος μετά την προσπάθεια εισαγωγής αρχείων

Στο συγκεκριμένο παράδειγμα βρισκόμαστε στην γεωγραφική περιφέρεια της Αττικής. Σε αυτήν επιχειρούμε να προσθέσουμε τρία αρχεία με μετρήσεις που αναφέρονται σε διαφορετικούς σταθμούς: στους σταθμούς Πεντέλη, Χαροκόπειο και Ασή Γωνιά. Όπως παρατηρούμε, το αρχείο της Πεντέλης υπάρχει ήδη μέσα στην βάση δεδομένων, χωρίς κάποια επιπλέον αλλαγή, και δεν προστίθεται. Το αρχείο του Χαροκόπειου αναφέρεται σε καινούργιες μετρήσεις και προστίθεται επιτυχώς αυξάνοντας την βάση δεδομένων του istSOS με 28 επιπλέον γραμμές μετρήσεων, ενώ το αρχείο της Ασή Γωνιά αποτυγχάνει, γιατί μόνο η γεωγραφική περιφέρεια Κρήτης έχει τα δικαιώματα του σταθμού και κανένας άλλος.

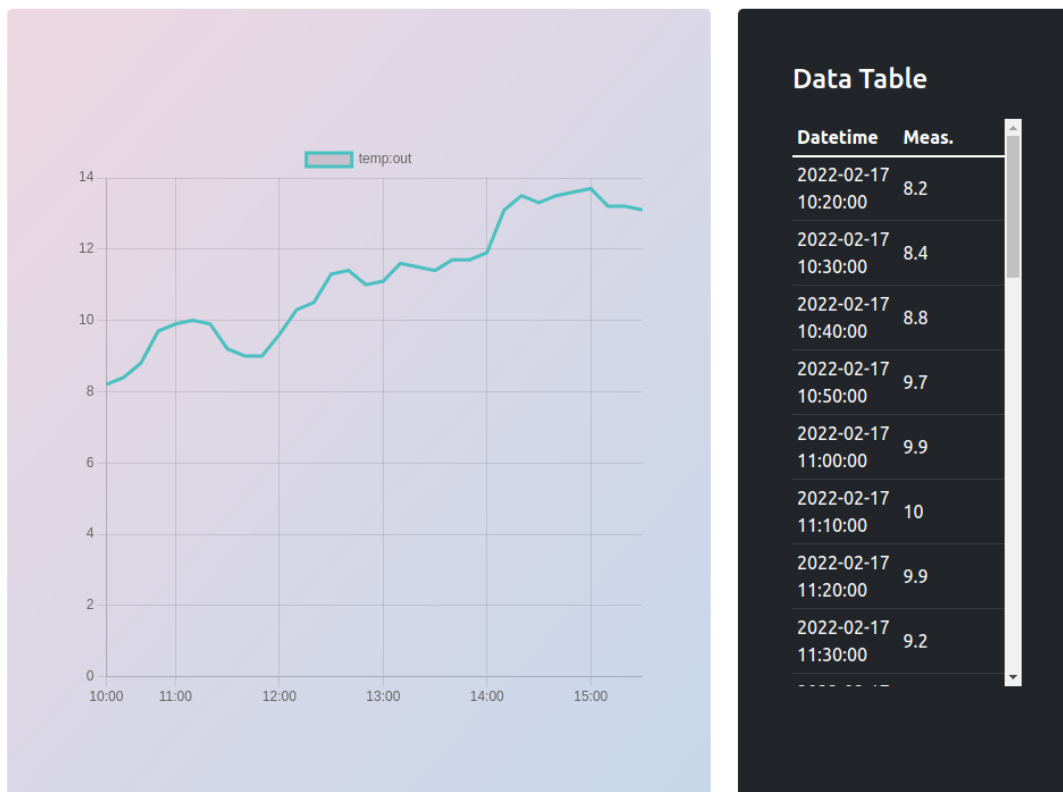
### Visualize data – Αναπαράσταση δεδομένων

Η δεύτερη σελίδα πρόκειται για την σελίδα αναπαράστασης δεδομένων στην οποία μπορεί να προηγηθεί ο χρήστης επιλέγοντας το κουμπί "Make chart" ή την δεύτερη επιλογή "Chart" από την επικεφαλίδα της κύρια σελίδας της εφαρμογής. Στην προβολή αυτή ο χρήστης μπορεί, αφού συμπληρώσει πρώτα τον πίνακα πληροφοριών, έχει την δυνατότητα να οπτικοποιήσει τα μετεωρολογικά δεδομένα που υπάρχουν στην βάση του istSOS με την χρήση διαγραμμάτων ή ακόμη να κάνει λήψη τους τοπικά στον υπολογιστή σε μορφή αρχείου JSON – κωδικοποίηση ιδιαίτερα χρήσιμη και συμβατή με μεγάλο εύρος εφαρμογών και υλοποιήσεων –.

The image shows two screenshots of a web application interface. The top screenshot is titled "Connect to istSOS database" and contains two input fields: "Ip" with the value "chain.softlab.ntua.gr" and "Db" with the value "blockchain". The bottom screenshot shows a form for selecting data. It includes a "Station" dropdown menu set to "ASIGONIA", "From" and "To" date pickers both set to "02/17/2022", and a "Sensor" dropdown menu set to "temp:out". At the bottom of this form are two green buttons: "JSON" and "Create chart".

Εικόνα 3-19 Στιγμιότυπο πίνακα στοιχείων για την απεικόνιση δεδομένων σε διάγραμμα

Στην επιλογή "Station" επιλέγεται ένας από τους υπάρχοντες σταθμούς στο δίκτυο, συμπληρώνεται η περίοδος κατά την οποία τα δεδομένα είναι επιθυμητά και τέλος, ο αισθητήρας ή οι αισθητήρες για τις μετρήσεις των οποίων θα γίνει το διάγραμμα. Με την επιλογή "JSON" γίνεται λήψη του αρχείου JSON, ενώ με την επιλογή "Create chart" κατασκευάζεται το διάγραμμα. Για την περίπτωση του αρχείου JSON υπάρχει και η επιλογή συμπερίληψης όλων των αισθητήρων του σταθμού.



Εικόνα 3-20 Στιγμιότυπο διαγράμματος δεδομένων θερμοκρασίας

Στα δεξιά του διαγράμματος υπάρχει ένας πίνακας που περιλαμβάνει όλες τις τιμές που είναι διαθέσιμες στην βάση για τον συγκεκριμένο αισθητήρα που έγινε το διάγραμμα κατά την περίοδο που δηλώθηκε από τον χρήστη.

### Blockchain history – Ιστορικό blockchain

Με την τελευταία επιλογή που είναι διαθέσιμη στην εφαρμογή μέσω του κουμπιού "Timeline" ή μέσω της τρίτης επιλογής "Blockchain", της επικεφαλίδας της κύριας σελίδας, ο χρήστης μπορεί να μεταβεί στην σελίδα που είναι αρμόδια για την οπτικοποίηση του περιεχομένου του blockchain, δηλαδή τον πίνακα ιστορικού που δείχνει όλες τις έγκυρες συναλλαγές που έλαβαν χώρα στο δίκτυο.

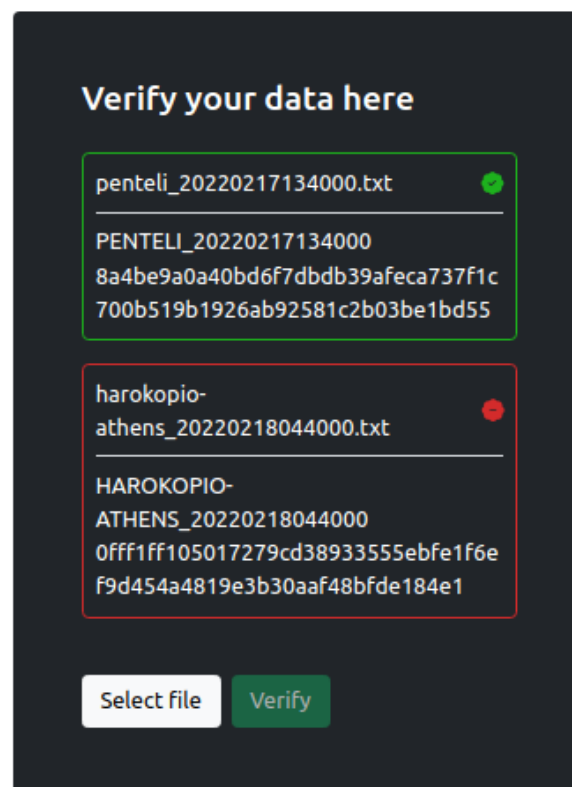
PENTELI		
PENTELI_20220217134000		
v3. PENTELI_20220217134000-3		
Hash value	First measurement	Last measurement
8a4be9a0a40bd6f7dbdb39	2022-02-	2022-02-
afeca737f1c700b519b1926	17T10:20:00+00:00	17T13:40:00+00:00
ab92581c2b03be1bd55		
v2. PENTELI_20220217134000-2		
Hash value	First measurement	Last measurement
7902d0cc520d8c5cd74803	2022-02-	2022-02-
992147087c77353e3aad2d	17T10:20:00+00:00	17T13:40:00+00:00
b6cb0d43fbb9f0ea31b1		
v1. PENTELI_20220217134000-1		
MENIDI-NEAODOS		

Εικόνα 3-21 Στιγμιότυπο πίνακα ιστορικού για το blockchain

Στο παραπάνω στιγμιότυπο φαίνονται όλοι σταθμοί που έχουν εισαχθεί μέχρι εκείνη την στιγμή στο blockchain ανεξαρτήτου γεωγραφικής περιφέρειας. Όπως εικονίζεται, υπάρχουν δύο σταθμοί στην βάση που, άμα επιλέξουμε κάποιον από αυτούς, βλέπουμε όλες τις ημερομηνίες που υπάρχουν μετρήσεις. Στον σταθμό Πεντέλη για παράδειγμα υπάρχουν μετρήσεις για μία μόνο μέρα την 17/02/2022, όπως φαίνεται και από τον τίτλο, για την οποία μέρα, όμως, υπάρχουν τρεις διαφορετικές εκδόσεις. Αυτό σημαίνει ότι το περιεχόμενο των μετρήσεων έχει τροποποιηθεί μερικώς από κάποιον εξουσιοδοτημένο χρήστη του δικτύου και οι αλλαγές αυτές είναι έγκυρες. Αναλυτικότερα, για την κάθε εισαγωγή που γίνεται μπορούμε να δούμε το hash value και την πρώτη και την τελευταία μέτρηση που περιλαμβάνει.

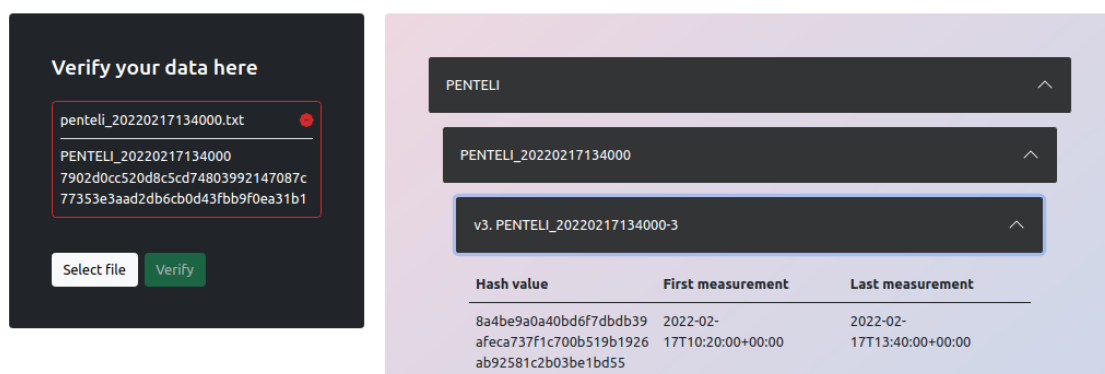
Πέραν του ιστορικού αρχείων εισαγωγής στο blockchain μέσω της σελίδας αυτής παρέχεται ακόμη, η δυνατότητα ελέγχου της γνησιότητας των αρχείων που έχει στην διάθεσή του ο χρήστης και τα οποία μπορεί να έχουν αποσταλεί από «ύποπτη» πηγή. Το μόνο που χρειάζεται να γίνει, είναι να επιλεγεί το αρχείο και η εφαρμογή κάνοντας κλήση στο συνάρτηση του έξυπνου συμβολαίου του blockchain, "verifyHash", μπορεί

να πιστοποιήσει την γνησιότητα ή μη του αρχείου, όπως περιεγράφηκε ενδελεχώς και στο υποκεφάλαιο 3.2.3.



Εικόνα 3-22 Στιγμιότυπο πιστοποίησης αρχείων

Στο παρόν παράδειγμα, ελέγχεται αν το hash value ενός αρχείου για τον σταθμό της Πεντέλης ταυτίζεται με αυτό που υπάρχει στο δίκτυο για την ίδια ημερομηνία. Από την άλλη, για το αρχείο του σταθμού Χαροκόπειο λαμβάνεται λανθασμένη σήμανση υποδηλώντας είτε πως ο συγκεκριμένος σταθμός και κατά επέκταση οποιοδήποτε αρχείο του δεν υπάρχει στο δίκτυο είτε ότι το συγκεκριμένο αρχείο είναι παραπονημένο. Παράδειγμα ενδεικτικής παραποίησης φαίνεται παρακάτω.



Εικόνα 3-23 Στιγμιότυπο παραποίησης δεδομένων

Στην αριστερή πλευρά της εικόνας φαίνεται το hash value του αρχείου που κατέχει ο χρήστης θεωρώντας γνήσιο και στην δεξιά το hash value του αρχείου που είναι αποθηκευμένο στο blockchain. Όπως γίνεται κατανοητό, διαφέρουν και ο χρήστης καταλαβαίνει ότι έχει πέσει θύμα απάτης και το αρχείο του έχει αλλοιωθεί.





# 4 Συμπεράσματα και Μελλοντική Έρευνα

Η προσπάθεια υλοποίησης μιας εφαρμογής blockchain για την διασφάλιση χωροαναφερόμενων χρονοσειρών μετρητικών δεδομένων μέσω πρότυπων υπηρεσιών Web περιεγράφηκε λεπτομερώς σε όλη την έκταση της παρούσας διπλωματικής. Ως πρότυπη υπηρεσία, χρησιμοποιήθηκε η πλατφόρμα του istSOS – μιας υπάρχουσας τεχνολογίας ανοιχτού κώδικα για την διαχείριση γεωχωρικών δεδομένων με την αξιοποίηση του SOS προτύπου – η οποία παρουσιάστηκε να συνεργάζεται αρμονικά με τις λειτουργικότητες που φέρνουν την καινοτομία των blockchain συστημάτων στην κορυφή της λίστας, με τις τεχνολογίες διασφάλισης της ακεραιότητας των δεδομένων, που την εμπιστεύονται. Για την ομαλή διάδραση της παραπάνω συνεργασίας "Blockchain – IstSOS" με τον χρήστη αναπτύχθηκε ένα Web DApp ως μία εφαρμογή μέσω της οποίας όποιος επιθυμεί θα έχει την δυνατότητα να αλληλεπιδράσει ταυτόχρονα με τα δύο άκρα της, χωρίς να διαθέτει προηγούμενες γνώσεις των αντικειμένων αυτών.

Όπως διαπιστώθηκε, ο στόχος της εργασίας, δηλαδή, η αποδοτική αποθήκευση των μετεωρολογικών δεδομένων που διατέθηκαν από το Εθνικό Αστεροσκοπείο Αθηνών για την παρούσα έρευνα, ολοκληρώθηκε με επιτυχία. Ειδικότερα, εισήχθησαν τα μετεωρολογικά δεδομένα στο istSOS και στο blockchain την ίδια στιγμή, με αποτέλεσμα να γίνεται εκμετάλλευση τόσο της εξειδικευμένης διαχείρισης τους από μία σχετική με το συγκεκριμένο αντικείμενο εφαρμογή, όπως είναι το istSOS, όσο και της ασφάλειας που τους παρέχει η κωδικοποίηση και μεταφόρτωσή τους στον δίκτυο που παράγει το blockchain. Μέσω του hash value που παράχθηκε για κάθε αρχείο δεδομένων χωριστά, σφραγίζεται η πιστότητά τους και μόνο ένας εξουσιοδοτημένος χρήστης δικαιούται να κάνει προσθήκες στα ήδη υπάρχοντα δεδομένα. Το ιδίωμα αυτό του blockchain, ως μίας τεχνολογίας μονής κατεύθυνσης, είναι που δικαιολογεί την αδυναμία των χρηστών να τροποποιήσουν, – καθώς μπορούν μόνο να προσθέσουν – υπάρχοντα δεδομένα με την παράλληλη απουσία οποιασδήποτε δυνατότητας προς τους μη εξουσιοδοτημένους «ύποπτους» χρήστες.

Ακολουθώντας την παραπάνω συλλογιστική πορεία γύρω από την οποία κατασκευάστηκε η λογική της παρούσας εργασίας, γίνεται κατανοητό ότι με την χρήση ενός δικτύου blockchain μπορεί να υπάρξει εγγύηση της ακεραιότητας του περιεχομένου του δικτύου μετεωρολογικών σταθμών μετρήσεων. Με βάση αυτό το σκεπτικό ανοίγεται δρόμος για γενικότερες υλοποιήσεις και εφαρμογές αντίστοιχου βεληνεκούς, στηριζόμενες στην ίδια βάση και εστιάζοντας στην ίδια οπτική γωνία που κυριάρχησε παραπάνω. Αξιολογώντας την χρήση μιας πλατφόρμας ανοιχτού κώδικα με δίκτυα blockchain μπορούν να φτιαχτούν δομές διαχείρισης δεδομένων πάσας φύσης και ιδιότητας. Σαν καινοτομία, η έννοια του blockchain είναι ήδη γνωστή ως βασικό εργαλείο, η χρήση της όμως, σε υπηρεσίες κλειστού κυκλώματος

όπως παρουσιάστηκε παραπάνω, δίνει την ευκαιρία για περαιτέρω προβληματισμό και αναζήτηση παρόμοιων περιπτώσεων χρήσης που θα μπορούσαν να επιλυθούν. Μια τέτοια εφαρμογή θα μπορούσε να αξιοποιηθεί για δεδομένα ιατρικού ενδιαφέροντος ή να επεκταθεί σε εφαρμογές χρήσιμες για την κοινωνία, ακόμη και για την διασφάλιση του περιεχομένου συνθηκών με ευρύτερη διάσταση.

Κλείνοντας, μπορεί να ειπωθεί, πως η παραπάνω μεθοδολογία, έπειτα από βελτιώσεις που απαιτούνται, θα μπορούσε να χρησιμοποιηθεί σαν βασικό εργαλείο στον τομέα της διασφάλισης γεωχωρικών δεδομένων υιοθετώντας την τεχνολογία του blockchain σε μια εφαρμογή εκτός του συνηθούς πεδίου δράσης του, αλλά αυτού των κρυπτονομισμάτων, βοηθώντας με τον τρόπο αυτό την επιστημονική κοινότητα να αξιοποιήσει παραγωγικά υπάρχουσες τεχνολογίες ασφάλειας με σκοπό την προστασία της ίδιας και των παρατηρήσεων της.

# Βιβλιογραφία

- [1] S. Squarepants, "Bitcoin: A Peer-to-Peer Electronic Cash System," *SSRN Electronic Journal*, 2022, doi: 10.2139/ssrn.3977007.
- [2] D. Drescher, *Blockchain basics: A non-technical introduction in 25 steps*. 2017. doi: 10.1007/978-1-4842-2604-9.
- [3] W. J. Buchanan, *Cryptography*. 2017. doi: 10.24297/ijct.v4i1a.3030.
- [4] "What is Blockchain? - Cruzlaw LLP Cruzlaw LLP." <https://cruzlaw.gi/what-is-blockchain/> (accessed Jun. 04, 2022).
- [5] A. N. Turi, "Blockchain and Distributed Ledger Technology Applications," in *Technologies for Modern Digital Entrepreneurship*, 2020. doi: 10.1007/978-1-4842-6005-0\_4.
- [6] "Blockchain Explained: How does a transaction get into the blockchain? | Euromoney Learning." <https://www.euromoney.com/learning/blockchain-explained/how-transactions-get-into-the-blockchain> (accessed Jun. 04, 2022).
- [7] "What Is a Smart Contract? » Explanation & Definition | Chainlink," Sep. 14, 2021. [https://chain.link/education/smart-contracts?utm\\_medium=paid-search&utm\\_source=google-adwords&utm\\_campaign=defi-oracle&utm\\_content=educational-page&gclid=Cj0KCQjw4uaUBhC8ARIsANUuDjVfGMHwB\\_j0c3Iyl9TJ3fxH7bHjrRwjWVuYey88hKucUU2PUR7rdQaAqmKEALw\\_wcB](https://chain.link/education/smart-contracts?utm_medium=paid-search&utm_source=google-adwords&utm_campaign=defi-oracle&utm_content=educational-page&gclid=Cj0KCQjw4uaUBhC8ARIsANUuDjVfGMHwB_j0c3Iyl9TJ3fxH7bHjrRwjWVuYey88hKucUU2PUR7rdQaAqmKEALw_wcB) (accessed Jun. 03, 2022).
- [8] "Introduction to smart contracts | ethereum.org." <https://ethereum.org/en/developers/docs/smart-contracts/> (accessed Jun. 03, 2022).
- [9] W. Kathleen E. and W. Eugenia, "Types of Blockchain: Public, Private, or Something in Between | Blogs | Manufacturing Industry Advisor | Foley & Lardner LLP." <https://www.foley.com/en/insights/publications/2021/08/types-of-blockchain-public-private-between> (accessed Jun. 02, 2022).
- [10] "Home | ethereum.org." <https://ethereum.org/en/> (accessed Jun. 03, 2022).
- [11] X. Li, P. Jiang, T. Chen, X. Luo, and Q. Wen, "A survey on the security of blockchain systems," *Future Generation Computer Systems*, vol. 107, 2020, doi: 10.1016/j.future.2017.08.020.
- [12] S. A. Renu and B. G. Banik, "Implementation of a secure ride-sharing DApp using smart contracts on ethereum blockchain," *International Journal of Safety and Security Engineering*, vol. 11, no. 2, 2021, doi: 10.18280/ijssse.110205.
- [13] D. Mohan, L. Alwin, P. Neeraja, K. D. Lawrence, and V. Pathari, "A private Ethereum blockchain implementation for secure data handling in Internet of Medical Things," *Journal of Reliable Intelligent Environments*, 2021, doi: 10.1007/s40860-021-00153-2.
- [14] "istSOS — istSOS 2.4.0-RC4 documentation." <http://istsos.org/en/latest/doc/index.html> (accessed Jun. 04, 2022).
- [15] OGC, "OGC Standards | OGC," *Open Geospatial Consortium*. 2018.
- [16] "The Home of Location Technology Innovation and Collaboration | OGC." <https://www.ogc.org/> (accessed Jun. 04, 2022).

- [17] "Introduction to the SOS standard — istSOS 2.4.0-RC4 documentation." <http://istsos.org/en/latest/doc/intro.html> (accessed Jun. 04, 2022).
- [18] B. Ventura, A. Vianello, D. Frisinghelli, M. Rossi, R. Monsorno, and A. Costa, "A methodology for heterogeneous sensor data organization and near real-time data sharing by adopting OGC SWE standards," *ISPRS International Journal of Geo-Information*, vol. 8, no. 4, 2019, doi: 10.3390/ijgi8040167.
- [19] "istSOS software — istSOS 2.4.0-RC4 documentation." <http://istsos.org/en/latest/doc/istsos.html> (accessed Jun. 05, 2022).
- [20] "CSV vs XML vs JSON – Which is the Best Response Data Format? | Digital Hospital." <https://digitalhospital.com.sg/csv-vs-xml-vs-json-which-is-the-best-response-data-format/> (accessed Jun. 12, 2022).
- [21] "PostgreSQL: About." <https://www.postgresql.org/about/> (accessed Jun. 05, 2022).
- [22] "About PostGIS | PostGIS." <http://www.postgis.net/> (accessed Jun. 05, 2022).
- [23] "Foundation Project." <https://www.apache.org/foundation/> (accessed Jun. 05, 2022).
- [24] M. Cannata, M. Antonovic, M. Molinari, and M. Pozzoni, "istSOS, a new sensor observation management system: software architecture and a real-case application for flood protection," *Geomatics, Natural Hazards and Risk*, vol. 6, no. 8, 2015, doi: 10.1080/19475705.2013.862572.
- [25] P. Yang, N. Xiong, and J. Ren, "Data Security and Privacy Protection for Cloud Storage: A Survey," *IEEE Access*, vol. 8, 2020. doi: 10.1109/ACCESS.2020.3009876.
- [26] "Έχουμε κλιματική κρίση - Greenpeace Ελλάδα." <https://www.greenpeace.org/greece/epirease/exoyme-klimatiki-krisi/> (accessed Jun. 06, 2022).
- [27] Amey, "How to build a dapp on a private Ethereum network." <https://medium.com/coinmonks/dapp-on-a-private-ethereum-network-1-c8b80695e049> (accessed Jun. 15, 2022).
- [28] "What is Geth? - ETH Gas Station." <https://legacy.ethgasstation.info/blog/what-is-geth/> (accessed Jun. 15, 2022).